HITACHI

ノンストップデータベース

HiRDB Version 10 コマンドリファレンス（Windows(R)用）

文法書

3020-6-559-C0

前書き

■ 対象製品

●適用 OS：Windows Server 2019，Windows Server 2022，Windows Server 2025，Windows 10 Pro (x64)，Windows 10 Enterprise (x64)，Windows 11

P-2962-91A4 HiRDB Server Version 10 10-12

P-2962-7PA4 HiRDB Accelerator Version 10 10-00

P-2962-7HA4 HiRDB Non Recover Front End Server Version 10 10-00

P-2962-7JA4 HiRDB Advanced High Availability Version 10 10-00

●適用 OS：Windows Server 2019，Windows Server 2022，Windows Server 2025，Windows 10，Windows 11

P-2662-11A4 HiRDB/Run Time Version 10 10-12

P-2662-12A4 HiRDB/Developer's Kit Version 10 10-12

P-2662-32A4 HiRDB Developer's Suite Version 10 10-12

●適用 OS：Windows Server 2019，Windows Server 2022，Windows Server 2025，Windows 10 Home (x64)，Windows 10 Pro (x64)，Windows 10 Enterprise (x64)，Windows 11

P-2962-11A4 HiRDB/Run Time Version 10(64) 10-12

P-2962-12A4 HiRDB/Developer's Kit Version 10(64) 10-12

これらのプログラムプロダクトのほかにもこのマニュアルをご利用になれる場合があります。詳細は「リリースノート」でご確認ください。

■ 輸出時の注意

本製品を輸出される場合には、外国為替および外国貿易法の規制ならびに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。

なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ 発行

2026 年 9 月　3020-6-559-C0

■ 著作権

All Rights Reserved. Copyright (C) 2018, 2026, Hitachi, Ltd.

変更内容

変更内容(3020-6-559-C0)　HiRDB Version 10 10-12

追加・変更内容	関連するコマンド	変更個所
クライアントマシンが複数の IP アドレスを持つ環境で，接続時の送信元 IP アドレスを明示的に指定する機能をサポートしました。 これによって，IP アドレスによる接続制限機能や，NAT やファイアウォールなど，送信元 IP アドレスを認識する機能を制御できます。	−	1.5.5
ディクショナリ搬出入ユティリティ（pdexp），及び pddefrev（定義系 SQL の生成）コマンドを使用して，次の定義を移行できるようにしました。 • プラグインが提供する抽象データ型を含む表定義（CREATE TABLE） • インデクス型を指定したインデクス定義（CREATE INDEX） • 手続きのパラメタのデータ型として抽象データ型を指定している手続き（CREATE PROCEDURE） • 手続き定義の中からほかの手続き，関数を呼び出している手続き（CREATE PROCEDURE） また，エクスポート・インポートによる表定義情報と表データの移行で，次の定義を移行できるようにしました。 • プラグインが提供する抽象データ型を含む表定義 • インデクス型を指定したインデクス定義	pdexp pddefrev	2.28.1(6)，9.3.2，9.3.3，9.3.5，表 9-1

単なる誤字・脱字などはお断りなく訂正しました。

変更内容(3020-6-559-B0)　HiRDB Version 10 10-11

追加・変更内容	関連するコマンド
pdinfoget コマンドで収集する情報を拡充しました。 また，初期情報（-e オプション）だけ取得できるようになりました。 これによって，HiRDB での障害発生時に早期に対応できるようになります。	pdinfoget
次の機能をサポートしたことによって，セキュリティの強化，ロール機能の利便性が向上しました。 • 監査対象イベントの定義で，オブジェクト種別（ROLE），セッション種別（ROLE）を指定できるようにしました。 • CURRENT_ROLE 値関数によって，現在使用しているロール名を検索できるようにしました。 • 埋込み型 UAP，ODBC から SET ROLE 文を発行できるようにしました。	pdaudput

変更内容(3020-6-559-A0)　HiRDB Version 10 10-10

追加・変更内容	関連するコマンド
表のアクセス権限を一括で管理できるロール機能をサポートしました。これによって，権限の管理や移行が容易になるだけでなく，アクセス制御の一貫性が向上し，セキュリティの強化にもつながります。	pdmod
HiRDB の適用 OS に次の OS を追加しました。 • Windows Server 2025	−

変更内容(3020-6-559-90)　HiRDB Version 10 10-09

追加・変更内容	関連するコマンド
データディクショナリ用 RD エリアの計算式を変更しました。	pdinit
ディクショナリ表に次の表識別子を追加しました。 • SQL_ROLES • SQL_ROLE_PRIVILEGES	pdrorg

はじめに

このマニュアルは，プログラムプロダクト　ノンストップデータベース　HiRDB Version 10 で使用するコマンドの文法について説明したものです。なお，ここに記載されていない前提情報については，マニュアル「HiRDB Version 10 解説」を参照してください。

■ 対象読者

HiRDB Version 10（以降，**HiRDB** と表記します）のコマンドを使用する方を対象にしています。

このマニュアルは次に示す知識があることを前提に説明しています。

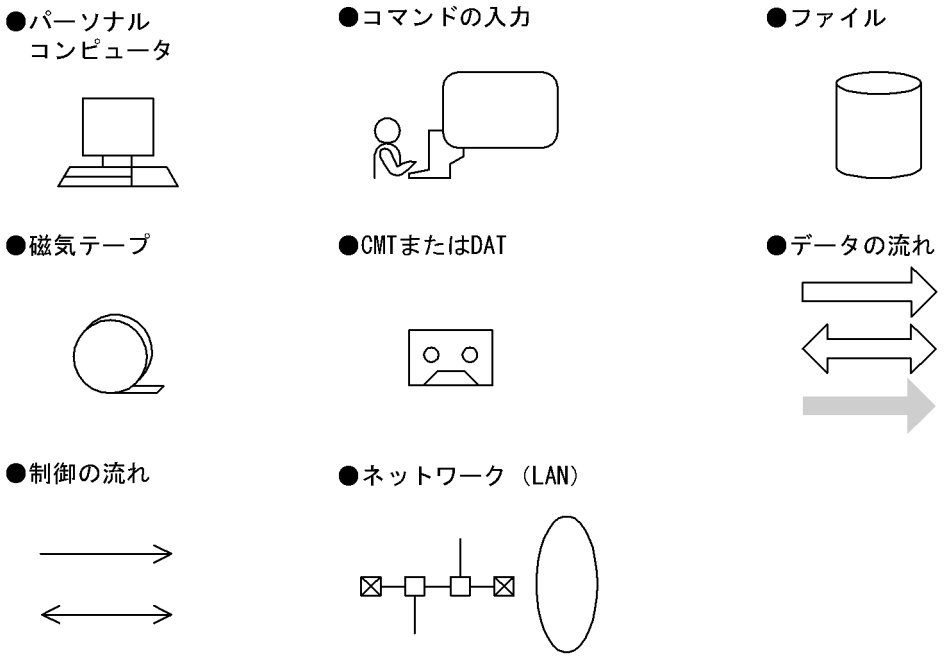
- Windows のシステム管理の基礎的な知識
- SQL の基礎的な知識

なお，このマニュアルは次に示すマニュアルを前提としていますので，あらかじめお読みいただくことをお勧めします。

- 「HiRDB Version 10 システム導入・設計ガイド」
- 「HiRDB Version 10 システム運用ガイド」

■ 図中で使用している記号

このマニュアルの図中で使用している記号を，次のように定義します。



■ このマニュアルで使用している記号

形式および説明で使用している記号を次に示します。ここで説明する文法記述記号は，説明のための記号なので実際には記述しないでください。

文法記述記号	意味
〔　〕	この記号で囲まれている項目は省略できます。 （例）pdfbkup 〔-y〕 　これは，pdfbkup と指定するか，または pdfbkup -y と指定することを示します。
…	この記号直前の項目を繰り返して指定できます。 （例）pdclose -r RD エリア名 〔，RD エリア名〕… 　これは，-r オプションの RD エリア名を繰り返し指定できることを示します。
｜	この記号で区切られた項目を選択できます。 （例）pdloginit -d sys ｜ spd 　これは，-d オプションに sys か spd のどちらかを指定できることを示します。
{　}	この記号で囲まれている複数の項目のうちから，一つを選択できます。 （例）pdcat -｛a ｜ b ｜ ab｝ 　これは，-a，-b，-ab の三つのオプションのうち，どれか一つを指定することを示します。 （例）pdstart 〔{-x ホスト名｜-s サーバ名}〕 　これは，-x ホスト名，-s サーバ名の二つのオプションのうち，どちらか一つを指定することを示します。
____（下線）	この記号で示す項目は，省略時の仮定値です。 （例）pdcat 〔-｛a ｜ b ｜ ab｝〕 　これは，オプションの指定を省略した場合，-ab オプションが仮定されることを示します。
～	この記号のあとにユーザ指定値の属性を示します。
《　》	ユーザが指定しなかった場合に仮定される値を示します。
＜　＞	ユーザ指定値の構文要素記号を示します。
((　))	ユーザ指定値の指定範囲を示します。

■ このマニュアルで使用している構文要素記号

このマニュアルで使用している構文要素記号を次に示します。

構文要素記号	意味
＜英字＞	アルファベット（A～Z，a～z）と下線（_）
＜英字記号＞	アルファベット（A～Z，a～z）と＃，@，￥
＜英数字＞	英字と数字（0～9）

構文要素記号		意味
＜英数字記号＞		英字記号と数字
＜符号なし整数＞		数字
＜ 16 進数字＞		数字と（A～F，a～f）
＜識別子＞		先頭がアルファベットの英数字
＜識別子＞	RD エリア名を指定する場合の識別子	先頭が英字記号で始まる英数字記号，下線（_），ハイフン（-），および空白 空白を含む場合は，引用符（“）で囲んでください。
＜識別子＞	ホスト名を指定する場合の識別子	アルファベット（A～Z，a～z），数字，ピリオド（.），ハイフン（-），下線（_），および@で構成される文字列
＜識別子＞	グローバルバッファ名を指定する場合の識別子	先頭が英字記号で始まる英数字記号
＜識別子＞	インメモリデータバッファ名を指定する場合の識別子	先頭が下線（_）で始まる英数字記号
＜記号名称＞		先頭が英字記号の英数字記号
＜文字列＞		任意の文字の配列
＜パス名＞※		¥，英数字，ピリオド（.），空白，丸括弧，#，@
＜ HiRDB ファイル名＞		アルファベット（A～Z，a～z），数字（0～9），ピリオド（.），下線（_），および@で構成される文字列（最大 30 文字）

注

すべて半角文字を使用してください。また，英字の大文字と小文字は区別されますが，一部区別されない名称もあります。

注※

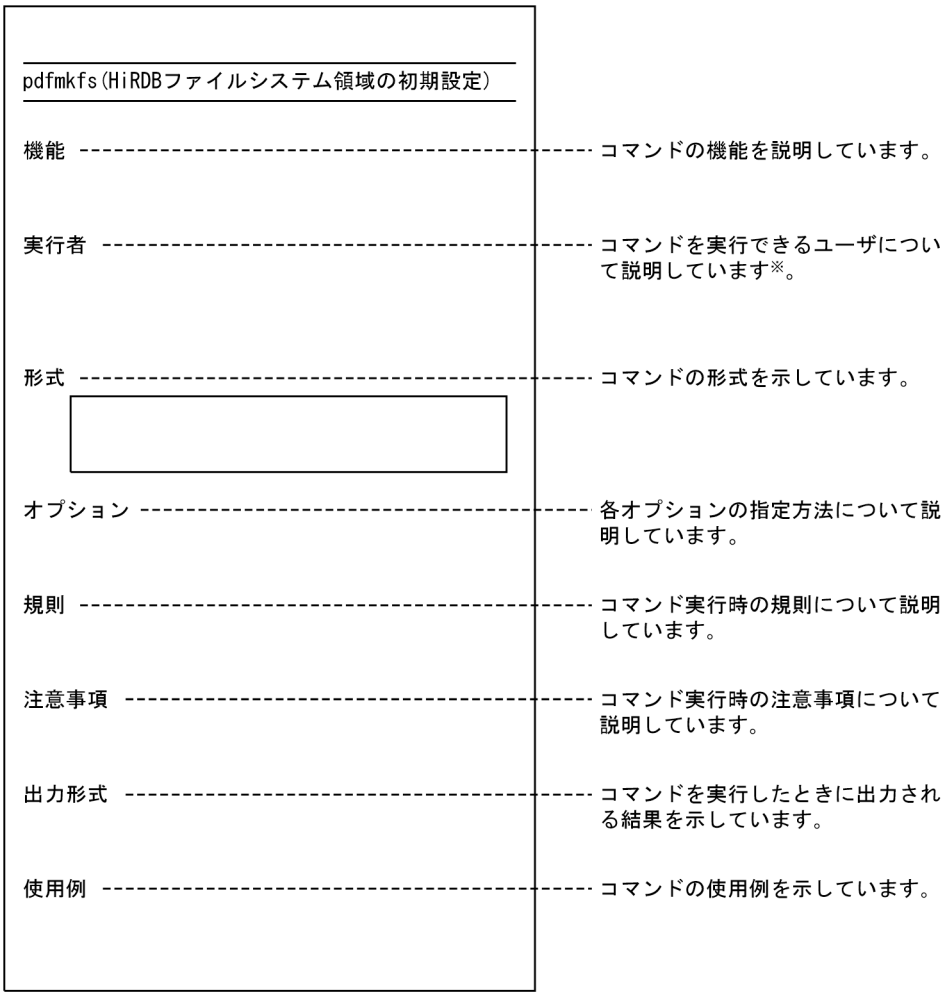
パス名は，使用している OS に依存します。

パス名に空白，または丸括弧を含む場合は，前後を引用符（"）で囲んでください。

また，ドライブ名にはコロン（:）を使用できます。

■ コマンドの記述構成

このマニュアルで説明しているコマンドの記述構成を次に示します。ただし，コマンドによって多少異なります。



注※

各コマンドの実行者については，表「運用コマンド一覧」および表「ユティリティー一覧」を参照してください。

目次

前書き　2
変更内容　3
はじめに　5

1　コマンドの概要　31
1.1　概要　32
1.1.1　コマンドの入力方法　32
1.1.2　コマンドの記述形式　33
1.1.3　コマンドの別名での利用　35
1.1.4　オプション，及び制御文を指定するときの注意事項　35
1.2　運用コマンド一覧　37
1.3　ユティリティー覧　44
1.4　ユティリティの入出力ファイル　47
1.4.1　「ファイル」と「ボリューム」の概念　47
1.4.2　ユティリティで使用できるファイルの形態　48
1.4.3　テープ装置アクセス機能　50
1.5　コマンド実行時の留意事項　51
1.5.1　コマンド実行中の割り込みによる強制終了　51
1.5.2　運用コマンド，ユティリティでの RD エリアの指定　51
1.5.3　コマンド実行結果を DAT 形式で出力する場合の規則　55
1.5.4　OS ログインユーザの簡易認証機能使用時の規則　57
1.5.5　環境変数に設定したクライアント環境変数の制限事項　57
1.5.6　一般ユーザによるコマンド実行時の注意事項　60
1.6　HiRDB の付加 PP を導入すると使用できる機能　61

2　運用コマンド　62
2.1　pdacunlck（連続認証失敗アカウントロック状態の解除）　63
2.1.1　pdacunlck の形式と規則　63
2.2　pdadmvr（HiRDB の製品情報の取得）　65
2.2.1　pdadmvr の形式と規則　65
2.3　pdaudatld（監査証跡表の自動データロード機能の制御）　68
2.3.1　pdaudatld の形式と規則　68
2.4　pdaudbegin（監査証跡の取得開始）　72
2.4.1　pdaudbegin の形式と規則　72
2.5　pdaudend（監査証跡の取得停止）　74

2.5.1 pdaudend の形式と規則 74
2.6 pdaudrm（閉塞中の監査証跡ファイルの削除） 76
2.6.1 pdaudrm の形式と規則 76
2.7 pdaudswap（現用の監査証跡ファイルのスワップ） 78
2.7.1 pdaudswap の形式と規則 78
2.8 pdbkupls（バックアップファイルの情報表示） 80
2.8.1 pdbkupls の形式と規則 80
2.9 pdbufls（グローバルバッファ情報の表示） 87
2.9.1 pdbufls の形式と規則 87
2.10 pdbufmod（グローバルバッファの動的変更） 99
2.10.1 pdbufmod の形式と規則 99
2.11 pdcancel（UAP，ユティリティ処理の強制終了） 109
2.11.1 pdcancel の形式と規則 109
2.12 pdcat（ファイルの内容表示） 113
2.12.1 pdcat の形式と規則 113
2.13 pdcat【-d sts】（ステータスファイルの内容表示） 114
2.13.1 pdcat【-d sts】の形式と規則 114
2.14 pdcat【-d 省略】（メッセージログファイルの内容表示） 117
2.14.1 pdcat【-d 省略】の形式と規則 117
2.15 pdchgconf（システム構成変更コマンド） 122
2.15.1 pdchgconf の形式と規則 122
2.16 pdchpathf（ファイルパス変更オフラインコマンド） 127
2.16.1 pdchpathf の形式と規則 127
2.17 pdchpathn（ファイルパス変更オンラインコマンド） 130
2.17.1 pdchpathn の形式と規則 130
2.18 pdchprc（サーバプロセスの起動本数変更） 132
2.18.1 pdchprc の形式と規則 132
2.19 pdclibsync（C ライブラリファイルの操作） 136
2.19.1 pdclibsync の形式と規則 136
2.20 pdclose（RD エリアのクローズ） 141
2.20.1 pdclose の形式と規則 141
2.21 pdclttrc（SQL トレースの動的取得） 144
2.21.1 pdclttrc の形式と規則 144
2.22 pdcmt（トランザクションのコミット） 148
2.22.1 pdcmt の形式と規則 148
2.23 pdconfchk（システム定義のチェック） 150
2.23.1 pdconfchk の形式と規則 150
2.24 pdcspool（トラブルシュート情報及び作業用一時ファイルの削除） 158
2.24.1 pdcspool の形式と規則 158

2.25 pddbfrz（ユーザ LOB 用 RD エリアの満杯 HiRDB ファイルの更新凍結） 162
2.25.1 pddbfrz の形式と規則 162
2.26 pddbls（RD エリアの状態表示） 165
2.26.1 pddbls の形式と規則 165
2.27 pddelfw.bat（Windows ファイアウォールの例外リスト削除） 174
2.27.1 pddelfw.bat の形式と規則 174
2.28 pddefrev（定義系 SQL の生成） 175
2.28.1 pddefrev の形式と規則 175
2.29 pdextfunc（拡張システム定義スカラ関数の定義，削除） 179
2.29.1 pdextfunc の形式と規則 179
2.30 pdfbkup（HiRDB ファイルシステムのバックアップ） 181
2.30.1 pdfbkup の形式と規則 181
2.31 pdfchfs（HiRDB ファイルシステム領域の管理情報変更） 184
2.31.1 pdfchfs の形式と規則 184
2.32 pdffsck（HiRDB ファイルシステム領域の整合性の検証及び修復） 188
2.32.1 pdffsck の形式と規則 188
2.33 pdfgt（トランザクションの強制終了） 193
2.33.1 pdfgt の形式と規則 193
2.34 pdfls（HiRDB ファイルシステムの内容表示） 196
2.34.1 pdfls の形式と規則 196
2.35 pdfmkfs（HiRDB ファイルシステム領域の初期設定） 201
2.35.1 pdfmkfs の形式と規則 201
2.36 pdfrm（HiRDB ファイルの削除） 212
2.36.1 pdfrm の形式と規則 212
2.37 pdfrstr（HiRDB ファイルシステムのリストア） 214
2.37.1 pdfrstr の形式と規則 214
2.38 pdfstatfs（HiRDB ファイルシステム領域の状態表示） 217
2.38.1 pdfstatfs の形式と規則 217
2.39 pdfzeroinit（HiRDB ファイルシステム領域内の使用済み領域の初期化） 223
2.39.1 pdfzeroinit の形式と規則 223
2.40 pdhold（RD エリアの閉塞） 225
2.40.1 pdhold の形式と規則 225
2.41 pdinfoget（障害情報の取得と容量見積もり） 235
2.41.1 pdinfoget の形式と規則 235
2.42 pdjarsync（JAR ファイルの操作） 246
2.42.1 pdjarsync の形式と規則 246
2.43 pdkill（プロセスの強制停止） 251
2.43.1 pdkill の形式と規則 251
2.44 pdlistls（リスト定義情報の表示） 252

2.44.1 pdlistls の形式と規則 252
2.45 pdlogadpf（ログ関係のファイルの割り当て） 256
2.45.1 pdlogadpf の形式と規則 256
2.46 pdlogatul（自動ログアンロード機能の制御） 259
2.46.1 pdlogatul の形式と規則 259
2.47 pdlogchg（ログ関係のファイルのステータス変更） 264
2.47.1 pdlogchg の形式と規則 264
2.48 pdlogcls（ログ関係のファイルのクローズ） 268
2.48.1 pdlogcls の形式と規則 268
2.49 pdloginit（ログ関係のファイルの初期設定） 270
2.49.1 pdloginit の形式と規則 270
2.50 pdlogls（ログ関係のファイルの情報表示） 276
2.50.1 pdlogls の形式と規則 276
2.51 pdlogls【-d spd】（シンクポイントダンプファイルの情報表示） 277
2.51.1 pdlogls【-d spd】の形式と規則 277
2.52 pdlogls【-d sys】（システムログファイルの情報表示） 280
2.52.1 pdlogls【-d sys】の形式と規則 280
2.53 pdlogopen（ログ関係のファイルのオープン） 294
2.53.1 pdlogopen の形式と規則 294
2.54 pdlogrm（ログ関係のファイルの削除） 296
2.54.1 pdlogrm の形式と規則 296
2.55 pdlogswap（ログ関係のファイルのスワップ） 299
2.55.1 pdlogswap の形式と規則 299
2.56 pdlogsync（シンクポイントダンプの取得） 302
2.56.1 pdlogsync の形式と規則 302
2.57 pdlogucat（アンロードログファイルの情報表示） 304
2.57.1 pdlogucat の形式と規則 304
2.58 pdlogunld（ログ関係のファイルのアンロード） 308
2.58.1 pdlogunld の形式と規則 308
2.59 pdls（HiRDB システムの状態表示） 313
2.59.1 pdls の形式と規則 313
2.60 pdls【-d act】（ユーザの状態表示） 316
2.60.1 pdls【-d act】の形式と規則 316
2.61 pdls【-d aud】（監査証跡ファイルの状態表示） 318
2.61.1 pdls【-d aud】の形式と規則 318
2.62 pdls【-d ha】（HiRDB の系の状態表示） 321
2.62.1 pdls【-d ha】の形式と規則 321
2.63 pdls【-d lck】（サーバの排他制御の状態表示） 329
2.63.1 pdls【-d lck】の形式と規則 329

2.64 pdls【-d mem】（サーバの共用メモリの状態表示） 338
2.64.1 pdls【-d mem】の形式と規則 338
2.65 pdls【-d prc】（サーバのプロセスの状態表示） 341
2.65.1 pdls【-d prc】の形式と規則 341
2.66 pdls【-d rpc】（サーバの通信制御情報の表示） 346
2.66.1 pdls【-d rpc】の形式と規則 346
2.67 pdls【-d rpl】（HiRDB Datareplicator 連携の状態表示） 351
2.67.1 pdls【-d rpl】の形式と規則 351
2.68 pdls【-d scd】（サーバのスケジュールの状態表示） 356
2.68.1 pdls【-d scd】の形式と規則 356
2.69 pdls【-d stj】（ユニット及びサーバごとの統計情報出力種別の指定有無の表示） 359
2.69.1 pdls【-d stj】の形式と規則 359
2.70 pdls【-d sts】（ステータスファイルの状態表示） 363
2.70.1 pdls【-d sts】の形式と規則 363
2.71 pdls【-d svr】（ユニット及びサーバの状態表示） 367
2.71.1 pdls【-d svr】の形式と規則 367
2.72 pdls【-d trn】（サーバのトランザクションの状態表示） 371
2.72.1 pdls【-d trn】の形式と規則 371
2.73 pdls【-d ust】（ユニットの稼働状態の表示） 379
2.73.1 pdls【-d ust】の形式と規則 379
2.74 pdmemdb（インメモリデータ処理に関するコマンド） 383
2.74.1 pdmemdb の形式と規則 383
2.75 pdmstchk（マスタディレクトリ用 RD エリアの整合性チェック） 390
2.75.1 pdmstchk の形式と規則 390
2.76 pdntenv（HiRDB の動作環境の設定） 394
2.76.1 pdntenv の形式と規則 394
2.77 pdobils（SQL オブジェクト用バッファの統計情報表示） 406
2.77.1 pdobils の形式と規則 406
2.78 pdobjconv（64 ビットモードの HiRDB への SQL オブジェクトの移行） 437
2.78.1 pdobjconv の形式と規則 437
2.79 pdopen（RD エリアのオープン） 441
2.79.1 pdopen の形式と規則 441
2.80 pdparaload（パラレルローディング） 443
2.80.1 pdparaload の形式と規則 443
2.81 pdpfresh（サーバプロセスのリフレッシュ） 452
2.81.1 pdpfresh の形式と規則 452
2.82 pdplgrgst（プラグインの登録・削除） 456
2.82.1 pdplgrgst の形式と規則 456
2.83 pdprfed（PRF トレース情報の編集出力） 460

2.83.1 pdprfed の形式と規則 460
2.84 pdprflevel（PRF トレース取得レベルの表示，変更） 463
2.84.1 pdprflevel の形式と規則 463
2.85 pdprgcopy（修正版 HiRDB のコピー） 465
2.85.1 pdprgcopy の形式と規則 465
2.86 pdprgrenew（修正版 HiRDB の入れ替え） 467
2.86.1 pdprgrenew の形式と規則 467
2.87 pdrbk（トランザクションのロールバック） 470
2.87.1 pdrbk の形式と規則 470
2.88 pdrdrefls（関連する RD エリアの情報の表示） 472
2.88.1 pdrdrefls の形式と規則 472
2.89 pdrels（RD エリアの閉塞解除） 480
2.89.1 pdrels の形式と規則 480
2.90 pdrplstart（HiRDB Datareplicator 連携の開始） 483
2.90.1 pdrplstart の形式と規則 483
2.91 pdrplstop（HiRDB Datareplicator 連携の終了） 485
2.91.1 pdrplstop の形式と規則 485
2.92 pdsetacl（ファイルセキュリティ強化機能の設定） 488
2.92.1 pdsetacl の形式と規則 488
2.93 pdsetfw.bat（Windows ファイアウォールの例外リスト登録） 493
2.93.1 pdsetfw.bat の形式と規則 493
2.94 pdstart（HiRDB システム・ユニット・サーバの開始） 495
2.94.1 pdstart の形式と規則 495
2.95 pdstbegin（統計情報の出力開始） 502
2.95.1 pdstbegin の形式と規則 502
2.96 pdstend（統計情報の出力停止） 515
2.96.1 pdstend の形式と規則 515
2.97 pdstjswap（統計ログファイルの切り替え） 520
2.97.1 pdstjswap の形式と規則 520
2.98 pdstjsync（統計ログファイルへの統計ログバッファの反映） 522
2.98.1 pdstjsync の形式と規則 522
2.99 pdstop（HiRDB システム・ユニット・サーバの終了） 526
2.99.1 pdstop の形式と規則 526
2.100 pdstscls（ステータスファイルのクローズ） 532
2.100.1 pdstscls の形式と規則 532
2.101 pdstsinit（ステータスファイルの初期設定） 535
2.101.1 pdstsinit の形式と規則 535
2.102 pdstsopen（ステータスファイルのオープン） 541
2.102.1 pdstsopen の形式と規則 541

2.103 pdstsrm（ステータスファイルの削除） 544
2.103.1 pdstsrm の形式と規則 544
2.104 pdstsswap（ステータスファイルのスワップ） 547
2.104.1 pdstsswap の形式と規則 547
2.105 pdsvhostname（サーバのホスト名表示） 549
2.105.1 pdsvhostname の形式と規則 549
2.106 pdtrndec（未決着トランザクションの強制自動決着） 551
2.106.1 pdtrndec の形式と規則 551
2.107 pdtrnqing（トランザクションキューイング機能の開始，解除） 555
2.107.1 HiRDB/シングルサーバ又は FES ユニットに pdtrnqing コマンドを実行する場合 555
2.107.2 影響分散スタンバイレス型系切り替え機能使用時に pdtrnqing コマンドを実行する場合 558
2.108 pdvrup（HiRDB バージョンアップ） 563
2.108.1 pdvrup の形式と規則 563

3 データベース初期設定ユティリティ（pdinit） 564
3.1 概要 565
3.1.1 機能 565
3.2 コマンドの形式 566
3.2.1 形式 566
3.2.2 オプション 566
3.3 制御文 568
3.3.1 define system 文 568
3.3.2 create rdarea 文 569
3.3.3 注意事項 580
3.4 規則及び注意事項 582
3.4.1 規則 582
3.4.2 注意事項 582
3.5 使用例 584

4 データベース定義ユティリティ（pddef） 594
4.1 概要 595
4.1.1 機能 595
4.1.2 実行者 595
4.2 コマンドの形式 596
4.2.1 形式 596
4.2.2 オプション 596
4.3 継続行の扱い 598
4.3.1 CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER 以外の SQL の場合 598
4.3.2 CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER の場合 598

4.3.3 SQL 文の途中で EOF を検知した場合 599
4.4 規則及び注意事項 600
4.4.1 規則 600
4.4.2 注意事項 600
4.4.3 SQL エラー時に出力される情報 601
4.4.4 注釈（コメント）の扱い 602
4.4.5 CONNECT 後の動作 603
4.4.6 SQL 文中の制御文字の扱い 603
4.4.7 BOM を持つファイルの使用 603
4.5 使用例 604

5 データベース作成ユティリティ（pdload） 607
5.1 機能 608
5.1.1 データロードとは 608
5.1.2 LOB 列がある表へのデータロード 610
5.1.3 プラグインが提供する抽象データ型列（LOB 属性）がある表へのデータロード 611
5.1.4 文字集合を定義した表へのデータロード 612
5.1.5 自動採番機能を使用したデータロード 613
5.1.6 同期点指定のデータロード 617
5.1.7 エンディアンが異なるプラットフォーム間のデータ移行 617
5.1.8 データロードする表が格納されている RD エリア 621
5.1.9 pdload 実行時のログ取得方式 621
5.1.10 実行者 621
5.2 代表的なデータロードの例 623
5.2.1 インデクスが定義されている表へのデータロード 623
5.2.2 LOB 列がある表へのデータロード 625
5.2.3 プラグインが提供する抽象データ型列がある表へのデータロード 626
5.3 目的ごとの参照先一覧 629
5.3.1 必ず指定する項目 629
5.3.2 データロードする表の属性によって指定する項目 630
5.3.3 入力データファイルの形態によって指定する項目 631
5.3.4 データロードの方法によって指定する項目 633
5.4 コマンドの形式 635
5.4.1 形式 635
5.4.2 オプション 635
5.4.3 source 文（入力データファイルの情報の記述） 665
5.4.4 index 文（インデクス情報ファイルの情報の記述） 672
5.4.5 idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述） 673
5.4.6 sort 文（ソート用ワークディレクトリの情報の記述） 674

5.4.7 lobdata 文（LOB 入力ファイルの情報の記述） 676
5.4.8 lobcolumn 文（列単位 LOB 入力ファイルの情報の記述） 677
5.4.9 lobmid 文（LOB 中間ファイルの情報の記述） 679
5.4.10 srcuoc 文（UOC 格納ライブラリの情報の記述） 682
5.4.11 array 文（繰返し列を含む表の場合の配列データ形式の記述） 684
5.4.12 extdat 文（拡張 DAT 形式の情報の記述） 687
5.4.13 src_work 文（分割入力データファイルの出力先の記述） 689
5.4.14 constraint 文（検査保留状態に関する記述） 694
5.4.15 option 文（データ処理に関する情報の記述） 699
5.5 入力データファイル 717
5.5.1 DAT 形式の場合 717
5.5.2 バイナリ形式の場合 725
5.5.3 固定長データ形式の場合 733
5.5.4 繰返し列を含む表の配列データ形式 742
5.5.5 列単位 LOB 入力ファイル 753
5.5.6 pdrorg 出力バイナリ形式ファイル 757
5.6 エラー情報ファイル 758
5.6.1 エラー情報ファイルの内容 758
5.6.2 エラー情報ファイルの出力例 760
5.6.3 エラー情報参照時の注意事項 761
5.6.4 source 文の errdata オペランド指定時の出力規則 762
5.7 列構成情報ファイル 765
5.7.1 列名文 766
5.7.2 スキップデータ文 787
5.7.3 追加データ文 790
5.7.4 記述例 791
5.8 ナル値・関数情報ファイル 796
5.8.1 列番号文 796
5.9 処理性能情報ファイル 798
5.10 データロード時のナル値，及び既定値 799
5.11 UOC を利用したデータロード 804
5.11.1 概要 804
5.11.2 UOC インタフェース 805
5.11.3 UOC の使用例 813
5.11.4 ダイナミックリンクライブラリの作成 815
5.12 規則及び注意事項 816
5.12.1 規則 816
5.12.2 注意事項 820
5.12.3 特定機能使用時のオプション，及び制御文の指定可否 826

5.12.4 BOM を持つファイルの使用 828
5.12.5 制御文で指定しない場合のファイル出力先ディレクトリ 829
5.13 エラー時のデータベースの状態とその回復方法 830
5.14 使用例 837
5.14.1 HiRDB/シングルサーバの場合 838
5.14.2 HiRDB/パラレルサーバの場合 857

6 会話型 SQL 実行ユティリティ 888
6.1 概要 889
6.1.1 機能 889

7 データベース構成変更ユティリティ（pdmod） 890
7.1 データベース構成変更ユティリティの機能 891
7.2 コマンドの形式 893
7.2.1 コマンドの形式とオプション 893
7.3 RD エリアの追加 895
7.3.1 機能 895
7.3.2 使用例 896
7.3.3 制御文（create rdarea 文） 900
7.3.4 注意事項 909
7.4 RD エリアの拡張 911
7.4.1 機能 911
7.4.2 使用例 912
7.4.3 制御文（expand rdarea 文） 913
7.4.4 注意事項 916
7.5 RD エリアの再初期化 917
7.5.1 機能 917
7.5.2 使用例 919
7.5.3 制御文（initialize rdarea 文） 921
7.5.4 再初期化後の表，及びインデクスの状態 929
7.5.5 実行結果リストの出力 934
7.5.6 注意事項 935
7.6 RD エリアの削除 939
7.6.1 機能 939
7.6.2 使用例 939
7.6.3 制御文（remove rdarea 文） 940
7.6.4 注意事項 941
7.7 RD エリアの移動 942
7.7.1 機能 942

7.7.2 使用例 947
7.7.3 制御文（move rdarea 文） 947
7.7.4 実行結果リストの出力 949
7.7.5 注意事項 950
7.8 ディクショナリ表の属性定義変更 952
7.8.1 機能 952
7.8.2 使用例 952
7.8.3 制御文（alter system 文） 953
7.8.4 注意事項 954
7.9 RD エリアの属性変更 955
7.9.1 機能 955
7.9.2 使用例 955
7.9.3 制御文（alter rdarea 文） 956
7.10 HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更 960
7.10.1 機能 960
7.10.2 使用例 961
7.10.3 制御文（alter HiRDB mode to parallel 文） 963
7.10.4 注意事項 964
7.11 監査人の登録（セキュリティ監査機能） 965
7.11.1 機能 965
7.11.2 使用例 966
7.11.3 制御文（create auditor 文） 966
7.11.4 注意事項 967
7.12 監査証跡表の作成（セキュリティ監査機能） 968
7.12.1 機能 968
7.12.2 使用例 968
7.12.3 制御文（create audit table 文） 968
7.13 規則及び注意事項 970
7.13.1 規則 970
7.13.2 注意事項 970

8 データベース再編成ユティリティ（pdrorg） 972
8.1 概要 973
8.1.1 pdrorg の機能 973
8.1.2 表の再編成 975
8.1.3 表のアンロード 978
8.1.4 表へのリロード 981
8.1.5 ディクショナリ表の再編成 983
8.1.6 インデクスの一括作成 984

8.1.7 インデクスの再作成 984
8.1.8 インデクスの再編成 985
8.1.9 ディクショナリ表のメンテナンス 986
8.1.10 pdrorg 実行時のログ取得方式 986
8.1.11 処理対象の表，インデクスが格納されている RD エリア 987
8.1.12 抽象データ型列がある表に対する pdrorg の実行可否 987
8.1.13 実行者 988
8.2 表の再編成 989
8.2.1 使用例 989
8.2.2 目的別の参照先一覧 1003
8.3 表のアンロード 1006
8.3.1 使用例 1006
8.3.2 目的別の参照先一覧 1024
8.3.3 データベース作成ユティリティの入力用ファイルの形式 1026
8.4 表へのリロード 1042
8.4.1 使用例 1042
8.4.2 目的別の参照先一覧 1042
8.4.3 アンロード元の表とリロード先の表の表定義が異なる場合のリロード実行可否 1044
8.4.4 表の再編成時のリロード処理でエラーとなった場合の運用方法 1049
8.5 ディクショナリ表の再編成 1059
8.5.1 使用例 1059
8.5.2 目的別の参照先一覧 1060
8.6 インデクスの一括作成 1063
8.6.1 使用例 1063
8.6.2 目的別の参照先一覧 1069
8.7 インデクスの再作成 1071
8.7.1 使用例 1071
8.7.2 目的別の参照先一覧 1078
8.8 インデクスの再編成 1081
8.8.1 使用例 1081
8.8.2 目的別の参照先一覧 1083
8.9 ディクショナリ表のメンテナンス 1085
8.9.1 実行手順 1085
8.9.2 メンテナンス処理中にエラーが発生した場合の対処方法 1085
8.9.3 メンテナンスする前の状態に戻す方法 1086
8.9.4 目的別の参照先一覧 1086
8.10 コマンドの形式 1088
8.10.1 形式 1088
8.10.2 オプション 1089

8.10.3 unload 文（アンロードデータファイルの情報の記述） 1122
8.10.4 index 文（インデクス情報ファイルの情報の記述） 1126
8.10.5 idxname 文（インデクスの情報の記述） 1127
8.10.6 idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述） 1129
8.10.7 sort 文（ソート用ワークディレクトリの情報の記述） 1130
8.10.8 lobunld 文（LOB データのアンロードファイルの情報の記述） 1131
8.10.9 unlduoc 文（UOC 格納ライブラリの情報の記述） 1135
8.10.10 tblname 文（別表への表データ再登録の情報の記述） 1136
8.10.11 array 文（繰返し列の行データ出力形式の記述） 1137
8.10.12 unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述） 1138
8.10.13 reld_func 文（コンストラクタ関数の情報の記述） 1140
8.10.14 constraint 文（検査保留状態に関する記述） 1141
8.10.15 option 文（データ処理に関する情報の記述） 1146
8.10.16 blobtovarchar 文（データ変換をする列の記述） 1157
8.10.17 fixtext_option 文（固定長データ形式の入力データファイル生成時の出力データに関する記述） 1158
8.11 処理性能情報ファイル 1162
8.12 UOC を利用した再編成 1163
8.12.1 再編成時の UOC の利用方法 1163
8.12.2 オプションと制御文の関係 1167
8.12.3 UOC の処理概要 1170
8.12.4 UOC インタフェース 1174
8.12.5 UOC の使用例 1190
8.13 特殊な表に対して pdrorg を実行する場合の注意事項 1193
8.13.1 抽象データ型がある表 1193
8.13.2 改竄防止表 1196
8.14 規則及び注意事項 1198
8.14.1 規則 1198
8.14.2 注意事項 1203
8.14.3 BOM を持つファイルの使用 1210
8.15 エラー時のデータベースの状態とその回復方法 1211
8.15.1 アンロード時 1212
8.15.2 リロード時 1212
8.15.3 インデクスの一括作成，及びインデクスの再作成実行時 1214
8.15.4 インデクスの再編成実行時 1215
8.15.5 LOB データのアンロード時 1216
8.15.6 LOB データのリロード時 1216
8.15.7 ディクショナリ表のメンテナンス実行時 1217

9 ディクショナリ搬出入ユティリティ（pdexp） 1218

9.1 概要 1219
9.1.1 表定義情報の搬出入 1221
9.1.2 トリガ定義情報の搬出入 1222
9.1.3 ストアドプロシジャ情報の搬出入 1222
9.1.4 定義系 SQL の生成 1222
9.1.5 実行者 1223
9.2 コマンドの形式 1224
9.2.1 形式 1224
9.2.2 オプション 1224
9.3 規則及び注意事項 1229
9.3.1 規則 1229
9.3.2 表定義情報，トリガ定義情報，及びストアドプロシジャ情報の搬出入についての規則 1230
9.3.3 定義系 SQL の生成についての規則 1232
9.3.4 参照制約を定義した表の搬出入の規則について 1234
9.3.5 注意事項 1235
9.3.6 搬入時に作成される定義系 SQL が 2,000,000 バイトを超えた場合の修正例 1236
9.3.7 BOM を持つファイルの使用 1238
9.3.8 単純注釈を使用する場合の搬出入 1239
9.4 使用例 1240

10 リバランスユティリティ（pdrbal） 1244

10.1 概要 1245
10.1.1 リバランスユティリティの機能 1245
10.1.2 リバランスユティリティの動作モード 1246
10.1.3 実行環境 1248
10.1.4 実行者 1249
10.2 使用例 1250
10.2.1 共有モードの場合 1250
10.2.2 占有モードの場合 1252
10.3 コマンドの形式 1256
10.3.1 形式 1256
10.3.2 オプション 1256
10.3.3 index 文（インデクス情報ファイルの情報の記述） 1267
10.3.4 idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述） 1268
10.3.5 sort 文（ソート用ワークディレクトリの情報の記述） 1269
10.3.6 execstop 文（pdrbal 実行時間の情報の記述） 1270
10.3.7 unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述） 1271
10.3.8 reld_func 文（コンストラクタ関数の情報の記述） 1273

10.3.9 report 文（処理結果ファイルの情報の記述） 1274
10.3.10 option 文（オプション機能の記述） 1276
10.4 リバランス中の FIX ハッシュ分割表に対して SQL を実行する場合の注意事項 1278
10.5 リバランス中の SQL 実行可否 1279
10.6 注意事項 1281
10.7 エラー時の対処方法 1284

11 空きページ解放ユティリティ（pdreclaim） 1285
11.1 概要 1286
11.1.1 空きページとは 1286
11.1.2 pdreclaim の概要 1287
11.1.3 pdreclaim の機能 1287
11.1.4 pdrorg との機能差 1293
11.1.5 実行環境 1295
11.1.6 実行者 1295
11.1.7 pdreclaim と他機能との同時実行 1295
11.2 使用例 1301
11.2.1 HiRDB/シングルサーバの場合 1301
11.2.2 HiRDB/パラレルサーバの場合 1302
11.3 コマンドの形式 1306
11.3.1 形式 1306
11.3.2 オプション 1307
11.3.3 idxname 文（インデクスの情報の記述） 1321
11.3.4 option 文（オプションの情報の記述） 1323
11.4 pdreclaim の処理結果 1325
11.4.1 pdreclaim（-j 及び-a なし）の処理結果（表の使用中空きページ解放） 1325
11.4.2 pdreclaim（-j 又は-a あり）の処理結果（表の使用中空きセグメント解放） 1326
11.4.3 pdreclaim（-j 及び-a なし）の処理結果（インデクスの使用中空きページ解放） 1326
11.4.4 pdreclaim（-j 又は-a あり）の処理結果（インデクスの使用中空きセグメント解放） 1327
11.4.5 処理結果についての注意事項 1328
11.5 注意事項 1330

12 グローバルバッファ常駐化ユティリティ（pdpgbfon） 1333
12.1 概要 1334
12.1.1 pdpgbfon の概要 1334
12.1.2 pdpgbfon の機能 1335
12.1.3 実行環境 1337
12.1.4 実行者 1337
12.1.5 pdpgbfon と他機能との同時実行 1337

12.2 使用例 1340
12.2.1 HiRDB/シングルサーバの場合 1340
12.2.2 HiRDB/パラレルサーバの場合 1341
12.3 コマンドの形式 1345
12.3.1 形式 1345
12.3.2 オプション 1346
12.3.3 idxname 文（インデクスの情報の記述） 1348
12.3.4 option 文（オプションの情報の記述） 1349
12.4 pdpgbfon の処理結果 1351
12.4.1 pdpgbfon の処理結果の詳細 1351
12.5 注意事項 1353

13 整合性チェックユティリティ（pdconstck） 1355
13.1 概要 1356
13.1.1 前提条件 1356
13.1.2 pdconstck 実行時の認可識別子 1357
13.2 pdconstck の機能 1359
13.2.1 整合性チェック機能 1359
13.2.2 検査保留状態の強制変更機能 1367
13.3 使用例 1371
13.3.1 表単位での整合性チェック 1371
13.3.2 制約単位での整合性チェック 1372
13.3.3 表単位での検査保留状態の強制設定 1373
13.3.4 制約単位での検査保留状態の強制解除 1374
13.4 コマンドの形式 1376
13.4.1 形式 1376
13.4.2 オプション 1376
13.5 処理結果 1380
13.5.1 pdconstck の処理結果 1380
13.5.2 整合性チェックの処理結果 1380
13.5.3 検査保留状態情報 1383
13.6 注意事項 1385

14 統計解析ユティリティ（pdstedit） 1386
14.1 概要 1387
14.1.1 統計解析ユティリティ（pdstedit）の概要 1387
14.2 コマンドの形式 1391
14.2.1 形式 1391
14.2.2 オプション 1391

14.2.3 各オプション指定時の留意事項 1399
14.3 統計情報の詳細 1403
14.3.1 統計情報の出力形式 1403
14.3.2 システムの稼働に関する統計情報 1408
14.3.3 UAP に関する統計情報 1423
14.3.4 SQL に関する統計情報 1435
14.3.5 グローバルバッファプールに関する統計情報 1438
14.3.6 データベース操作に関する HiRDB ファイルの統計情報 1443
14.3.7 デファードライト処理に関する統計情報 1445
14.3.8 インデクスに関する統計情報 1447
14.4 DAT 形式ファイルへの統計情報出力 1452
14.4.1 DAT 形式ファイルのデータ格納形式 1452
14.4.2 DAT 形式ファイルの単位 1452
14.4.3 DAT 形式ファイルのレコード形式 1453
14.5 注意事項 1498
14.6 使用例 1499

15 データベース状態解析ユティリティ（pddbst） 1502
15.1 概要 1503
15.1.1 DB 状態解析機能 1503
15.1.2 状態解析結果蓄積機能 1506
15.1.3 再編成時期予測機能 1507
15.2 コマンドの形式 1511
15.2.1 形式 1511
15.2.2 オプション 1512
15.2.3 predict 文 1520
15.2.4 workdir 文 1521
15.2.5 option 文 1522
15.3 DB 状態解析機能の出力形式 1524
15.3.1 RD エリア単位の状態解析（論理的解析） 1524
15.3.2 RD エリア単位の状態解析（物理的解析） 1551
15.3.3 表単位又はインデクス単位の状態解析 1561
15.3.4 クラスタキー及びクラスタリングデータページの格納状態解析 1577
15.4 再編成時期予測機能の実行結果 1581
15.4.1 実行結果の出力形式 1581
15.4.2 CSV 形式での出力形式 1589
15.5 基準値定義ファイル（再編成時期予測機能） 1594
15.5.1 形式および規則 1594
15.6 予測レベル 2 での状態解析結果蓄積機能の実行方法 1601

15.6.1 インターバル解析 1601
15.6.2 マージ解析 1601
15.6.3 選択基準 1602
15.7 規則及び注意事項 1605
15.7.1 規則 1605
15.7.2 注意事項 1605
15.7.3 BOM を持つファイルの使用 1606
15.7.4 コミットしていない削除データに排他を掛ける場合の注意事項 1606
15.8 使用例 1607
15.8.1 DB 状態解析機能 1607
15.8.2 状態解析結果蓄積機能 1607
15.8.3 再編成時期予測機能 1607

16 最適化情報収集ユティリティ（pdgetcst） 1608
16.1 概要 1609
16.1.1 検索による最適化情報の収集 1609
16.1.2 最適化情報パラメタファイルによる最適化情報の登録 1610
16.1.3 最適化情報パラメタファイルへの最適化情報の退避 1611
16.1.4 ユティリティの最大同時実行数 1612
16.2 コマンドの形式 1613
16.2.1 形式 1613
16.2.2 オプション 1614
16.3 最適化情報 1618
16.3.1 取得の有無 1618
16.3.2 取得する最適化情報 1618
16.3.3 最適化情報収集レベル 1622
16.4 最適化情報パラメタファイル 1624
16.4.1 最適化情報パラメタファイルの項目 1624
16.4.2 最適化情報パラメタファイルの項目の指定方法 1628
16.4.3 最適化情報パラメタファイルの項目の出力内容 1633
16.4.4 最適化情報パラメタファイルの例 1635
16.4.5 最適化情報パラメタファイルの記述規則 1636
16.5 規則及び注意事項 1638
16.5.1 規則 1638
16.5.2 注意事項 1638
16.5.3 BOM を持つファイルの使用 1639
16.6 出力形式 1640
16.6.1 検索による最適化情報の収集 1640
16.6.2 最適化情報パラメタファイルによる最適化情報の登録 1642

16.7 使用例 1645
16.7.1 最適化情報を収集する場合 1645
16.7.2 最適化情報を収集し，処理結果をファイルに出力する場合 1645
16.7.3 最適化情報をディクショナリ表から削除する場合 1645
16.7.4 最適化情報パラメタファイルの最適化情報を登録する場合 1645
16.7.5 最適化情報を最適化情報パラメタファイルに退避する場合 1645

17 アクセスパス表示ユティリティ（pdvwopt） 1646
17.1 概要 1647
17.1.1 機能 1647
17.2 コマンドの形式 1648
17.2.1 形式 1648
17.2.2 オプション 1648
17.2.3 引数 1648
17.3 実行手順 1650
17.4 規則及び注意事項 1651
17.4.1 規則 1651
17.4.2 注意事項 1651
17.5 出力形式 1653
17.5.1 集合演算処理情報 1656
17.5.2 問合せ処理情報 1659
17.5.3 結合処理情報 1662
17.5.4 実表検索処理情報 1668
17.5.5 結合方法の種類 1689
17.5.6 検索方法の種類 1691
17.5.7 転送方法の種類 1692
17.5.8 アクセスパスの種類 1695
17.5.9 グループ分け処理方式の種類 1696
17.5.10 プラグイン提供関数の種別 1698
17.5.11 サーチ条件 1699
17.5.12 キー条件 1717
17.5.13 ORDER BY 処理方式の種類 1723
17.6 チューニングの考え方 1726
17.6.1 チューニング手順の例 1726
17.6.2 インデクス定義について 1729
17.6.3 インデクス検索について 1737
17.6.4 結合検索について 1743

18 データベース複写ユティリティ（pdcopy） 1745

18.1 データベース複写ユティリティの機能 1746
18.2 バックアップを取得するために必要な知識 1747
18.2.1 バックアップの取得単位 1747
18.2.2 バックアップ取得モード 1747
18.2.3 バックアップを取得するタイミング 1748
18.2.4 同時にバックアップを取得する必要がある RD エリア 1749
18.2.5 バックアップファイルを格納するサーバマシン 1749
18.2.6 アンロードレスシステムログ運用をしている場合 1749
18.2.7 差分バックアップ機能を使用してバックアップを取得する場合 1750
18.2.8 インメモリデータ処理を使用している場合 1750
18.2.9 データベース複写ユティリティを実行するときの HiRDB の状態について 1750
18.3 使用例 1751
18.3.1 システム単位にバックアップを取得する場合 1751
18.3.2 RD エリア単位にバックアップを取得する場合 1751
18.3.3 サーバ単位にバックアップを取得する場合 1752
18.3.4 DAT 上のファイルへバックアップを取得する場合 1752
18.3.5 差分バックアップ機能を使用してバックアップを取得する場合 1753
18.3.6 NetBackup を使用してバックアップを取得する場合 1753
18.4 コマンドの形式 1754
18.4.1 オプションの形式 1754
18.4.2 オプションの説明 1756
18.5 注意事項 1772
18.6 データベース複写ユティリティの処理結果リスト 1774
18.6.1 処理結果出力ファイルの内容 1774
18.6.2 差分バックアップの履歴情報ファイルの内容 1778
18.7 バックアップファイルの形式 1780

19 データベース回復ユティリティ（pdrstr） 1783

19.1 データベース回復ユティリティの機能 1784
19.1.1 データベースの回復 1784
19.1.2 ログポイント情報ファイルの再作成 1785
19.2 データベースを回復するために必要な知識 1786
19.2.1 データベースの回復単位 1786
19.2.2 回復対象の RD エリアの状態 1786
19.2.3 バックアップ取得モードとの関係 1787
19.2.4 HiRDB の開始モードとの関係 1787
19.2.5 マスタディレクトリ用 RD エリアを回復する場合 1787
19.2.6 構成変更前に取得したバックアップファイルを使用して，バックアップだけの回復をする場合 1788

19.2.7 最新の状態への回復 1788
19.2.8 最新の状態以外に回復する場合 1788
19.2.9 アンロードログファイル又はシステムログファイルを使用してデータベースを回復した場合 1789
19.2.10 アンロードレスシステムログ運用の場合の回復 1789
19.2.11 差分バックアップ機能を使用してバックアップからの回復 1789
19.2.12 範囲指定の回復 1790
19.2.13 データベース回復ユティリティを実行するときの HiRDB の状態について 1790
19.3 使用例 1791
19.3.1 システム単位の回復をする場合 1791
19.3.2 RD エリア単位の回復をする場合 1792
19.3.3 サーバ単位の回復をする場合 1793
19.3.4 差分バックアップ機能を使用したバックアップから回復をする場合 1793
19.3.5 NetBackup を使用したバックアップから RD エリアを回復する場合 1793
19.3.6 ログポイント情報ファイルを再作成する場合 1794
19.4 コマンドの形式 1795
19.4.1 オプションの形式 1795
19.4.2 オプションの説明 1797
19.5 注意事項 1813
19.6 データベース回復ユティリティの処理結果リスト 1815

20 レジストリ機能初期設定ユティリティ（pdreginit） 1818
20.1 概要 1819
20.1.1 機能 1819
20.2 コマンドの形式 1820
20.2.1 形式 1820
20.2.2 オプション 1820
20.3 制御文 1822
20.3.1 create rdarea 文 1822
20.3.2 注意事項 1825
20.4 規則及び注意事項 1826
20.4.1 規則 1826
20.4.2 注意事項 1826
20.5 使用例 1828

21 JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput） 1830
21.1 概要 1831
21.1.1 前提条件及び動作環境 1831
21.1.2 JP1/NETM/Audit 用監査ログ出力ユティリティで使用するファイル 1832
21.1.3 JP1/NETM/Audit 用監査ログ出力ファイルの内容 1834

21.1.4 監査証跡の出力に関する留意事項 1845
21.2 コマンドの形式 1846
21.2.1 形式 1846
21.2.2 オプション 1846
21.3 指定できるオプションの組み合わせ 1850
21.4 リターンコード一覧 1851

付録 1852

付録 A コマンド一覧 1853
付録 B コマンド実行時の排他制御モード 1859
付録 B.1 運用コマンドの排他制御モード 1859
付録 B.2 ユティリティの排他制御モード 1860
付録 C コマンド実行時の RD エリアの状態 1879
付録 C.1 RD エリアの状態遷移 1879
付録 C.2 RD エリアの状態によるユティリティ及び UAP の実行可否 1881
付録 C.3 検査保留状態の設定可否 1892
付録 D ユティリティの最大同時実行数 1895
付録 D.1 各ユティリティの最大同時実行数 1895
付録 D.2 複数のユティリティの同時実行数 1896
付録 D.3 注意事項 1897
付録 E pdload 及び pdrorg で使用する UOC の作成方法と注意事項 1898
付録 E.1 ダイナミックリンクライブラリの作成方法 1898
付録 E.2 UOC を作成する場合の注意事項 1899
付録 F コマンドの同時接続数 1903
付録 G コマンドのリターンコード一覧 1907
付録 H コマンドが出力するファイル一覧 1924
付録 I バージョンアップによって省略値が変更，又は指定不要になったオプション及び制御文 1929
付録 I.1 省略値が変更，又は指定不要になったオプション，制御文 1929

索引 1934

1 コマンドの概要

この章では，コマンドの入力方法，及び記述形式について説明します。

1.1 概要

HiRDB で使用するコマンドには，運用コマンドとユティリティとがあります。入力方法及び記述形式については，どちらも同じです。

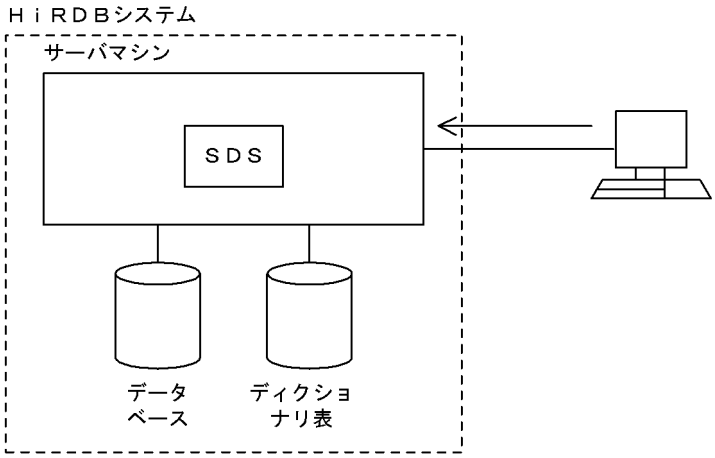
1.1.1 コマンドの入力方法

HiRDB のコマンドは，コマンドプロンプトから入力します。

コマンドを実行する場合，HiRDB サーバをインストールしたドライブに移動してから実行するようにしてください。例えば，HiRDB サーバを D ドライブにインストールした場合は，コマンドプロンプトで D ドライブに移動した後にコマンドを実行してください。

コマンドの実行形態（HiRDB/シングルサーバの場合）を次の図に示します。

図 1–1　コマンドの実行形態（HiRDB/シングルサーバの場合）



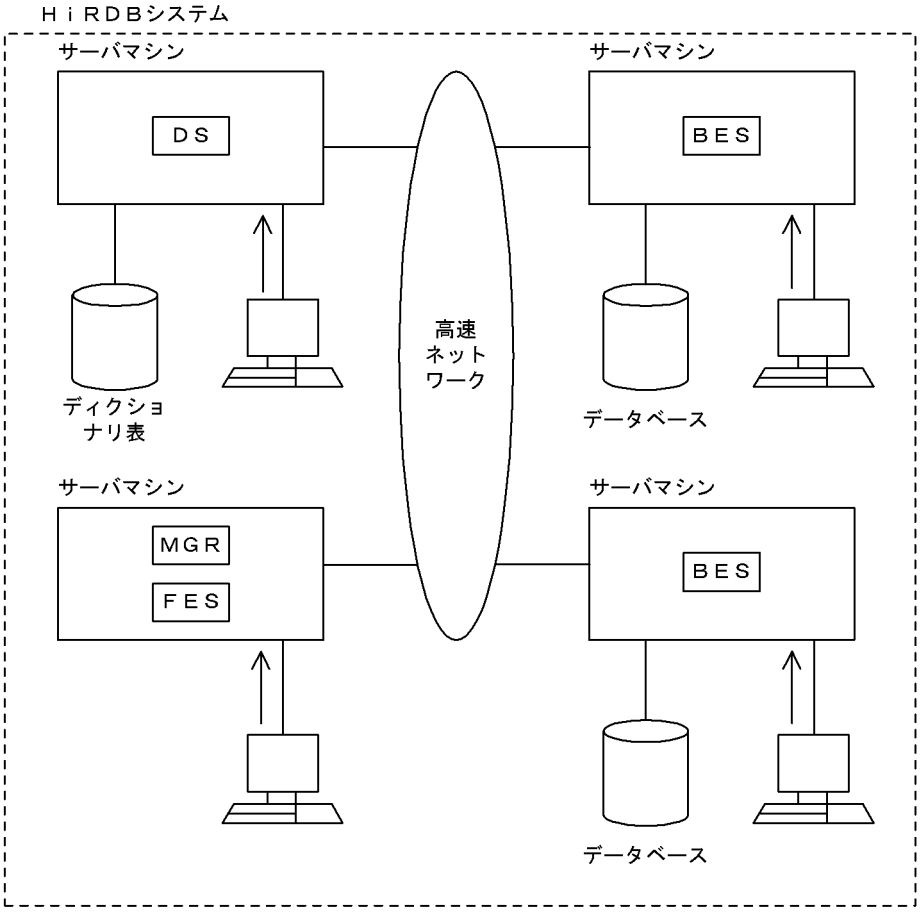
（凡例）　ＳＤＳ：シングルサーバ

[説明]

シングルサーバがあるサーバマシンでコマンドを入力します。

コマンドの実行形態（HiRDB/パラレルサーバの場合）を次の図に示します。

図 1–2　コマンドの実行形態（HiRDB/パラレルサーバの場合）



（凡例）ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＤＳ　：ディクショナリサーバ
ＢＥＳ：バックエンドサーバ

[説明]

システムマネジャがあるサーバマシンでコマンドを入力します。

なお，コマンドによってはシステムマネジャ以外のサーバマシンから実行する場合もあります。各コマンドを実行するサーバマシンについては，表「運用コマンド一覧」，表「ユティリティー一覧」，又は各コマンドの説明箇所を参照してください。

1.1.2　コマンドの記述形式

コマンドの記述形式を次に示します。

```
コマンド名　オプション　コマンド引数
```

(1)　コマンド名

コマンド名は，実行するコマンドのファイル名称です。

(2) オプション

オプションで指定するのは，コマンドの操作そのものを修飾するか，又は操作の対象物を特定するために，対象物を修飾するかのどちらかです。

次の説明中で使用する cmd は，コマンド名称を示します。

- オプションはマイナス記号（−）で始まる文字列で，フラグ引数を指定しないか，又は 1 種類のフラグ引数を指定します。

  オプションの記述形式を次に示します。

  −オプションフラグ

  　又は

  −オプションフラグ　フラグ引数

  （凡例）

  オプションフラグ：

  　1 文字の英数字（英大文字と英小文字は区別されます）。

  フラグ引数：

  　オプションフラグに対する引数（コンマ（,）で区切って複数指定できます）。
- フラグ引数のないオプションフラグは，一つのマイナス記号の後ろにまとめて指定できます。

  （例）次の二つは同じ意味です。

  　cmd -a -b -c

  　cmd -abc
- フラグ引数を必要とするオプションフラグのフラグ引数は，省略できません。
- フラグ引数中に空白を含める場合は，前後を引用符（"）で囲まなければなりません。

  （例）1 2 を引数に持つオプション f は，次のように記述します。

  　cmd -f "1 2"

  なお，空白を含むパス名のフラグ引数を指定する場合も，同様にフラグ引数全体の前後を引用符で囲んでください。
- 同じオプションフラグを 2 回以上指定すると，最後に指定したものが有効になります。

  （例）次のように入力すると，-a 2 が有効となります。

  　cmd -a 1 -a 2
- オプションは，コマンド引数より前に指定しなければなりません。

  （例）オプションフラグ a が，フラグ引数をとらない場合，次のように入力すると，file と-b はコマンド引数とみなされます。

  　cmd -a file -b
- 二つのマイナス記号（−−）は，オプションの終わりを示します。

（例）次のように入力すると，-b はコマンド引数とみなされます。

cmd -a -- -b

- マイナス記号だけのオプションは，入力できません。

（例）次のように入力すると，–はコマンド引数とみなされます。

cmd -

(3) コマンド引数

コマンド引数は，コマンド操作の直接の対象となるものを指定します。対象物を修飾指定する場合，オプションで対象物を指定します。

(4) 注意事項

制御文ファイルに，空白を含むパス名を指定する場合は，パス名の前後を引用符（"）で囲む必要があります。

（例）pdload の source 文に空白を含むパス名を指定する場合

source host1:"c:¥hirdb data¥data1","c:¥hirdb data¥data2"

1.1.3 コマンドの別名での利用

OS やほかのプログラムが提供しているコマンド名称と HiRDB のコマンド名称が同一であると，HiRDB のコマンドを実行できないことがあります。そのような場合，次の回避策があります。

- 環境変数の設定を HiRDB のコマンド優先にします。
- 絶対パスを指定してコマンドを実行します。

しかし，上記の二つの回避策がとれない場合，HiRDB のコマンドを任意の名称で実行できる方法があります。コマンドを別名で実行するためのバッチファイルの作成方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

1.1.4 オプション，及び制御文を指定するときの注意事項

1. コマンドのオプションで，認可識別子やパスワードなどを小文字で指定する場合，「¥""」と「"¥"」で囲んで指定してください。

   （例）

   pddbst -u ¥""hitachi"¥" -p ¥""hitachi"¥"と指定した場合，HiRDB では pddbst -u "hitachi" -p "hitachi"と解釈されます。

2. 制御文などの編集にメモ帳などを使用する場合，「ファイル名を付けて保存」でファイルの種類はテキストファイルを選択してください。そのほかの種類を選択した場合，文字コード体系が変更されるため，ユティリティ実行時などに制御文エラーとなることがあります。
3. 制御文を作成する場合，行の最後に必ず「改行」を入れてください。最後の行に改行なしで EOF となっている場合，HiRDB は制御文を正しく認識できません。

1.2　運用コマンド一覧

HiRDB の運用コマンド一覧を次の表に示します。なお，運用コマンドによっては，「運用コマンドの実行者」「HiRDB 稼働中での実行」「実行するサーバマシン」に，さらに条件が付くことがあります。詳細は，各運用コマンドの説明を参照してください。

表 1–1　運用コマンド一覧

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
pdacunlck	連続認証失敗アカウントロック状態を解除します。	HiRDB 管理者	○	SDS 又は MGR
pdadmvr	HiRDB の製品情報を表示します。	HiRDB 管理者	◎	各サーバマシン
pdaudatld	監査証跡表の自動データロード機能を制御します。	HiRDB 管理者及び監査人※1	○	SDS 又は MGR
pdaudbegin	監査証跡の取得を開始します。	HiRDB 管理者	○	SDS 又は MGR
pdaudend	監査証跡の取得を停止します。	HiRDB 管理者	○	SDS 又は MGR
pdaudrm	監査証跡ファイルを削除します。	HiRDB 管理者	○	SDS 又は MGR
pdaudswap	監査証跡ファイルをスワップします。	HiRDB 管理者及び監査人※1	○	SDS 又は MGR
pdbkupls	pdcopy で取得したバックアップファイルの情報を表示します。	HiRDB 管理者	◎	バックアップファイルがあるホストのサーバマシン
pdbufls	グローバルバッファの状態を表示します。	HiRDB 管理者	○	SDS 又は MGR
pdbufmod	HiRDB 稼働中にグローバルバッファの追加，削除，又は変更をします。	HiRDB 管理者	○	SDS 又は MGR
pdcancel	UAP 又はユティリティの処理をする HiRDB 側の処理プロセスを，UAP 又はユティリティから強制的に切り離し，終了します。	HiRDB 管理者	○	任意のサーバマシン
pdcat	ステータスファイルの内容を表示します。	HiRDB 管理者	◎	任意のサーバマシン
	メッセージログファイルの内容を表示します。	一般ユーザ	○	任意のサーバマシン
pdcbl	COBOL 言語で記述した UAP のプリプロセスをします※2。	一般ユーザ	◎	任意のサーバマシン
pdchgconf	HiRDB の稼働中に，システム定義を変更します。	HiRDB 管理者	○	SDS 又は MGR
pdchpathf	HiRDB ファイルに登録しているパス情報を変更します。	HiRDB 管理者	×	SDS 又は DS

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
pdchpathn	ディクショナリ表に登録している HiRDB ファイルシステム領域のパスを変更します。	HiRDB 管理者	○	SDS 又は MGR
pdchprc	サーバプロセスの常駐プロセス数と最大起動プロセス数を変更します。また，サーバプロセスの情報も表示します。	HiRDB 管理者	○	SDS 又は MGR
pdclibsync	C ライブラリファイルを操作します。	HiRDB 管理者	◎	任意のサーバマシン
pdclose	RD エリアをクローズします。	HiRDB 管理者	○	SDS 又は MGR
pdclttrc	SQL 実行中のクライアントの，SQL トレースを取得します。	HiRDB 管理者	○	SDS 又は MGR
pdcmt	グローバルトランザクションを構成しているトランザクションブランチが，トランザクションを決着できないときに，トランザクションをコミットします。	HiRDB 管理者	○	各サーバマシン
pdconfchk	HiRDB システム定義の整合性をチェックします。	HiRDB 管理者	◎	SDS 又は MGR
pdcpp	C 言語で記述した UAP のプリプロセスをします※2。	一般ユーザ	◎	任意のサーバマシン
pdcspool	トラブルシュート情報及び作業用一時ファイルを削除します。	HiRDB 管理者	◎	各サーバマシン
pddbfrz	ユーザ LOB 用 RD エリアを構成する HiRDB ファイルで，データが満杯となった HiRDB ファイルを，更新凍結状態にします。	HiRDB 管理者	○	SDS 又は MGR
pddbls	RD エリアの状態を表示します。	HiRDB 管理者	○	SDS 又は MGR
pddefrev	定義系 SQL を生成します。	DBA 権限保持者，又は表の所有者	○	SDS 又は MGR
pddelfw.bat	Windows ファイアウォールの例外リストから HiRDB を削除します。	HiRDB 管理者	×	各サーバマシン
pdextfunc	拡張システム定義スカラ関数の定義又は削除をします。	HiRDB 管理者	○	任意のサーバマシン
pdfbkup	HiRDB ファイルシステム領域のバックアップを取得します。	HiRDB 管理者	◎	HiRDB ファイルシステム領域があるサーバマシン
pdfchfs	HiRDB ファイルシステム領域の管理情報を変更します。	HiRDB 管理者	◎	HiRDB ファイルシステム領域があるサーバマシン

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
pdffsck	HiRDB ファイルシステム領域の整合性の検証及び修復をします。	HiRDB 管理者	◎	HiRDB ファイルシステム領域があるサーバマシン
pdfgt	グローバルトランザクションを構成している各トランザクションブランチが何らかの要因でトランザクションを終了できない場合，強制的にトランザクションを終了します。	HiRDB 管理者	○	各サーバマシン
pdfls	HiRDB ファイルシステム領域内の HiRDB ファイルの内容を表示します。	一般ユーザ	◎	HiRDB ファイルシステム領域があるサーバマシン
pdfmkfs	ファイルを HiRDB ファイルシステム領域用に初期設定します。	HiRDB 管理者	◎	HiRDB ファイルシステム領域を初期設定するサーバマシン
pdfrm	HiRDB ファイルシステム領域から HiRDB ファイルを削除します。	HiRDB 管理者	◎	HiRDB ファイルシステム領域があるサーバマシン
pdfrstr	pdfbkup コマンドで取得したバックアップを HiRDB ファイルシステム領域にリストアします。	HiRDB 管理者	×	HiRDB ファイルシステム領域があるサーバマシン
pdfstatfs	HiRDB ファイルシステム領域の状態を表示します。	一般ユーザ	◎	HiRDB ファイルシステム領域があるサーバマシン
pdfzeroinit	指定した HiRDB ファイルシステム領域内にある使用済み領域を初期化します。	HiRDB 管理者	◎	HiRDB ファイルシステム領域があるサーバマシン
pdhold	RD エリアを閉塞します。	HiRDB 管理者	○	SDS 又は MGR
pdinfoget	HiRDB システムで障害が発生した場合，障害の原因を調査するために障害情報を取得します。また，取得する情報の容量見積もりをします。	HiRDB 管理者	◎	任意のサーバマシン
pdjarsync	JAR ファイルの登録，再登録，削除，又は一覧表示をします。	HiRDB 管理者	◎	SDS 又は MGR
pdkill	プロセスを強制的に終了します。	HiRDB 管理者	○	各サーバマシン
pdlistls	リスト定義情報を表示します。	HiRDB 管理者	○	SDS 又は MGR

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
pdlogadpf	HiRDB システム定義で指定したファイルグループに物理ファイルを割り当てます。	HiRDB 管理者	○	SDS 又は MGR
pdlogatul	自動ログアンロード機能の再開始，中断，又は動作状態の表示をします。	HiRDB 管理者	○	SDS 又は MGR
pdlogchg	ファイルグループのステータスを変更します。	HiRDB 管理者	◎	SDS 又は MGR
pdlogcls	ファイルグループを構成するオープン中の物理ファイルをクローズし，稼働中の HiRDB で使用しないようにします。	HiRDB 管理者	○	SDS 又は MGR
pdloginit	HiRDB ファイルシステム領域にログ関係の物理ファイルを作成し，稼働中の HiRDB で使用できるように初期設定します。	HiRDB 管理者	◎	SDS 又は MGR
pdlogls	シンクポイントダンプファイルの情報を表示します。	HiRDB 管理者	○	SDS 又は MGR
	システムログファイルの情報を表示します。	HiRDB 管理者	◎	SDS 又は MGR
pdlogopen	ファイルグループを構成するクローズ中の物理ファイルをオープンし，稼働中の HiRDB で使用できるようにします。	HiRDB 管理者	○	SDS 又は MGR
pdlogrm	HiRDB ファイルシステム領域にあるログ関係のファイル※3 を削除します。	HiRDB 管理者	◎	SDS 又は MGR
pdlogswap	ログ関係のファイル※3 をスワップします。	HiRDB 管理者	○	SDS 又は MGR
pdlogsync	シンクポイントダンプを取得します。	HiRDB 管理者	○	SDS 又は MGR
pdlogucat	アンロードログファイルの情報を表示します。	HiRDB 管理者	◎	アンロードログファイルがあるホストのサーバマシン
pdlogunld	アンロードログファイルを作成し，ファイルグループをアンロード済み状態にします。	HiRDB 管理者	◎	SDS 又は MGR
pdls	次に示す状態を表示します。 • ユーザの状態 • 監査証跡ファイルの状態 • 系切り替え機能を使用しているときの系の状態 • サーバの排他の状態 • サーバの共用メモリの情報 • サーバのプロセスの状態 • コネクションが確立しているサーバの通信制御情報 • HiRDB Datareplicator 連携の状態 • サーバのスケジュールの状態	-d オプションの指定によって異なります。	-d オプションの指定によって異なります。	-d オプションの指定によって異なります。

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
	• ユニット及びサーバごとの統計情報出力種別の指定有無 • ステータスファイルの状態 • ユニット及びサーバの状態 • サーバのトランザクションの状態 • ユニットの稼働状態			
pdmemdb	インメモリデータ処理に関する操作をします。	HiRDB 管理者	○	SDS 又は MGR
pdmstchk	マスタディレクトリ用 RD エリアとディクショナリ表の整合性をチェックします。	HiRDB 管理者	○	SDS 又は DS
pdntenv	HiRDB の動作環境を表示，又は設定します。	HiRDB 管理者	◎	各サーバマシン
pdobils	SQL オブジェクト用バッファに格納されている SQL オブジェクトの利用状況を表示します。	HiRDB 管理者	○	任意のサーバマシン
pdobjconv	64 ビットモードの HiRDB へ SQL オブジェクトを移行します。	DBA 権限保持者	○	SDS 又は MGR
pdocb	OOCOBOL 言語で記述した UAP のプリプロセスをします※2。	一般ユーザ	◎	任意のサーバマシン
pdocc	C++言語で記述した UAP のプリプロセスをします※2。	一般ユーザ	◎	任意のサーバマシン
pdopen	RD エリアをオープンします。	HiRDB 管理者	○	SDS 又は MGR
pdosinfoget	HiRDB システムで障害が発生した場合，障害の原因を調査するために OS の情報を取得します。	スーパユーザ	○	任意のサーバマシン
pdparaload	パラレルローディングを実行します。	pdload と同じ	○	SDS 又は MGR
pdpfresh	現在稼働中のサーバプロセスを終了させ，代わりに新しいサーバプロセスを開始します。	HiRDB 管理者	○	SDS 又は MGR
pdplgrgst	プラグインの登録，削除，又はバージョンアップをします。	DBA 権限保持者	○	任意のサーバマシン
pdprfed	PRF トレース情報ファイル（バイナリ形式）を CSV 形式に変換して出力します。	HiRDB 管理者	◎	任意のサーバマシン
pdprflevel	PRF トレース取得レベル（pd_prf_level オペランドの指定値）を表示又は変更します。	HiRDB 管理者	○	任意のサーバマシン
pdprgcopy	修正版 HiRDB を入れ替え用ディレクトリ下にコピーします。	HiRDB 管理者	◎	SDS 又は MGR
pdprgrenew	稼働中の HiRDB を，修正版 HiRDB に入れ替えます。	HiRDB 管理者	○	SDS 又は MGR
pdrbk	グローバルトランザクションを構成しているトランザクションブランチがトランザクションを	HiRDB 管理者	○	各サーバマシン

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
	決着できないときに，トランザクションをロールバックします。			
pdrdrefls	関連する RD エリアの情報の表示，又は RD エリアの関連性チェックをします。	HiRDB 管理者	○	SDS 又は MGR
pdrels	RD エリアの閉塞状態を解除します。	HiRDB 管理者	○	SDS 又は MGR
pdrplstart	HiRDB Datareplicator 連携を開始します。	HiRDB 管理者	○	SDS 又は MGR
pdrplstop	HiRDB Datareplicator 連携を終了します。	HiRDB 管理者	○	SDS 又は MGR
pdsetacl	ファイルセキュリティ強化機能の設定をします。	Administrators 権限保持者	×	各サーバマシン
pdsetfw.bat	Windows ファイアウォールの例外リストに HiRDB を登録します。	HiRDB 管理者	×	各サーバマシン
pdstart	HiRDB システムを開始します。	HiRDB 管理者	×	SDS 又は MGR
	ユニット又はサーバを開始します。	HiRDB 管理者	○	SDS 又は MGR
pdstbegin	統計ログファイルへの統計情報の出力を開始します。	HiRDB 管理者	○	SDS 又は MGR
pdstend	統計ログファイルへの統計情報の出力を停止します。	HiRDB 管理者	○	SDS 又は MGR
pdstjswap	統計ログファイルをスワップします。	HiRDB 管理者	○	SDS 又は MGR
pdstjsync	統計ログファイルに統計ログバッファの情報を出力します。	HiRDB 管理者	○	任意のサーバマシン
pdstop	HiRDB システム，ユニット，又はサーバを終了します。	HiRDB 管理者	○	SDS 又は MGR
pdstscls	オープン状態のステータスファイルをクローズします。	HiRDB 管理者	○	SDS 又は MGR
pdstsinit	ステータスファイル（論理ファイル）を構成する，物理ファイルを初期設定します。	HiRDB 管理者	◎	SDS 又は MGR
pdstsopen	初期設定したステータスファイル，又はクローズしたステータスファイルをオープンします。	HiRDB 管理者	○	SDS 又は MGR
pdstsrm	ステータスファイルを削除します。	HiRDB 管理者	◎	SDS 又は MGR
pdstsswap	ステータスファイルをスワップします。	HiRDB 管理者	○	SDS 又は MGR
pdsvhostname	サーバのホスト名を DAT 形式で表示します。	HiRDB 管理者	○	SDS 又は MGR
pdtrndec	未決着のトランザクションがある場合，そのトランザクションを強制的に決着します。	HiRDB 管理者	○	SDS 又は MGR
pdtrnqing	トランザクションキューイングの開始，又は解除をします。	HiRDB 管理者	○	任意のサーバマシン

運用コマンド	内容	運用コマンドの実行者	HiRDB 稼働中での実行	実行するサーバマシン
pdvrup	HiRDB をバージョンアップしたときに，マスタディレクトリ用 RD エリア及びデータディクショナリ用 RD エリアを保守します。	HiRDB 管理者	○	SDS 又は MGR

（凡例）

◎：HiRDB の稼働に関係なく実行できます。

○：HiRDB が稼働中のときだけ実行できます。

×：HiRDB が停止中のときだけ実行できます。

SDS：HiRDB/シングルサーバの場合に，シングルサーバがあるサーバマシンで実行することを示します。

MGR：HiRDB/パラレルサーバの場合に，システムマネジャがあるサーバマシンで実行することを示します。

DS：HiRDB/パラレルサーバの場合に，ディクショナリサーバがあるサーバマシンで実行することを示します。

注 1

運用コマンド又はユティリティの実行時に指定したファイルを，ほかの運用コマンド，ユティリティ，又は UAP で同時に使用しないでください。同じファイルを運用コマンド，ユティリティ，又は UAP で同時に使用した場合は，処理結果が不正となったりエラーが発生したりすることがあります。

注 2

運用コマンドは，HiRDB のサービスの状態が開始（OS によって表記は異なります）となっているときに実行してください。

注※1

HiRDB 管理者でコマンドを実行し，HiRDB へアクセスする際の認証には監査人の認可識別子とパスワードを指定してください。

注※2

詳細は，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

注※3

システムログファイル及びシンクポイントダンプファイルを示します。

1.3　ユティリティー覧

HiRDB のユティリティー覧を次の表に示します。なお，ユティリティによっては，「ユティリティの実行者」「HiRDB 稼働中での実行」「実行するサーバマシン」に，さらに条件が付くことがあります。詳細は，各ユティリティの説明を参照してください。

表 1–2　ユティリティー覧

ユティリティ	内容	ユティリティの実行者	HiRDB 稼働中での実行	実行するサーバマシン
データベース初期設定ユティリティ（pdinit）	データベースの初期設定（RD エリアの作成）をします。	HiRDB 管理者	○	SDS 又は MGR
データベース定義ユティリティ（pddef）	定義系 SQL を実行します。	CONNECT 権限保持者	○	任意のサーバマシン
データベース作成ユティリティ（pdload）	ユーザが用意したデータを表に格納します。	動作モードによって異なります。	○	SDS 又は MGR
会話型 SQL 実行ユティリティ	会話形式で SQL を実行できます。	CONNECT 権限保持者	○	任意のサーバマシン
データベース構成変更ユティリティ（pdmod）	RD エリアの追加，拡張，又は再初期化など，データベースの物理構成を変更します。	HiRDB 管理者	○	SDS 又は MGR
データベース再編成ユティリティ（pdrorg）	表の再編成，インデクスの再作成など，表のメンテナンスをします。	機能によって異なります。	○	SDS 又は MGR
ディクショナリ搬出入ユティリティ（pdexp）	表定義情報又はストアドプロシジャ情報を移行します。	DBA 権限保持者，又は表の所有者	○	SDS 又は MGR
リバランスユティリティ（pdrbal）	リバランス表に RD エリアを追加したときに発生する，データの偏りを修正します。	DBA 権限保持者，又は SELECT 権限，INSERT 権限，DELETE 権限保持者	○	SDS 又は MGR
空きページ解放ユティリティ（pdreclaim）	オンライン中に空きページ（使用中空きページ）を解放します。	DBA 権限保持者，又は INSERT 権限，DELETE 権限保持者	○	SDS 又は MGR

ユティリティ	内容	ユティリティの実行者	HiRDB 稼働中での実行	実行するサーバマシン
グローバルバッファ常駐化ユティリティ（pdpgbfon）	表のページ情報をグローバルバッファに読み込みます。	DBA 権限保持者，SELECT 権限保持者，又はスキーマの所有者	○	SDS 又は MGR
整合性チェックユティリティ（pdconstck）	参照制約又は検査制約が定義された表に対して，整合性チェック及び検査保留状態の操作（設定又は解除）をします。	DBA 権限保持者，又は表の所有者	○	SDS 又は MGR
統計解析ユティリティ（pdstedit）	アンロードデータファイル，及びシステムログファイルから情報を入力し，統計情報を編集します。	HiRDB 管理者	◎	任意のサーバマシン
データベース状態解析ユティリティ（pddbst）	ユーザ用 RD エリアの表とインデクス，又はデータディクショナリ用 RD エリアの格納状態を解析し，解析結果を表示します。	DBA 権限保持者	○	SDS 又は MGR
最適化情報収集ユティリティ（pdgetcst）	コストベースの最適化のための最適化情報を取得し，ディクショナリ表に最適化情報を登録します。	SELECT 権限保持者	○	SDS 又は MGR
アクセスパス表示ユティリティ（pdvwopt）	SQL 最適化処理によって決定されたアクセスパスを表示します。	一般ユーザ	◎	SDS 又は FES
データベース複写ユティリティ（pdcopy）	• データベースのバックアップを取得します。 • ログポイント情報の取得，及びログポイント情報ファイルの作成をします。	HiRDB 管理者	○	SDS 又は MGR
データベース回復ユティリティ（pdrstr）	• 障害が発生した場合，データベースのバックアップとバックアップ取得以降のログを基にして，データベースを回復します。 • ログポイント情報ファイルを再作成します。	HiRDB 管理者	○	SDS 又は MGR
レジストリ機能初期設定ユティリティ（pdreginit）	レジストリ機能を管理，操作するためのレジストリ用 RD エリア，レジストリ LOB 用 RD エリア，及びレジストリ管理表を作成します。	HiRDB 管理者	○	SDS 又は MGR
JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput）	JP1/NETM/Audit 用監査ログ出力ファイルに，監査証跡表のデータを出力します。	HiRDB 管理者及び監査人※	○	SDS 又は MGR

（凡例）

◎：HiRDB が稼働中でも停止中でも実行できます。

○：HiRDB が稼働中のときだけ実行できます。

SDS：HiRDB/シングルサーバの場合に，シングルサーバがあるサーバマシンで実行することを示します。

MGR：HiRDB/パラレルサーバの場合に，システムマネジャがあるサーバマシンで実行することを示します。

FES：HiRDB/パラレルサーバの場合に，フロントエンドサーバがあるサーバマシンで実行することを示します。

注 1

運用コマンド又はユティリティの実行時に指定したファイルを，ほかの運用コマンド，ユティリティ，又は UAP で同時に使用しないでください。同じファイルを運用コマンド，ユティリティ，又は UAP で同時に使用した場合は，処理結果が不正となったりエラーが発生したりすることがあります。

注 2

ユティリティは，HiRDB のサービスの状態が開始（OS によって表記は異なります）となっているときに実行してください。

注※

HiRDB 管理者でコマンドを実行し，HiRDB へアクセスする際の認証には監査人の認可識別子とパスワードを指定してください。

1.4　ユティリティの入出力ファイル

1.4.1 「ファイル」と「ボリューム」の概念

HiRDB では，「ファイル」及び「ボリューム」を次のように定義しています。

ファイル：

ユティリティからみた論理的な単位

ボリューム：

物理的な格納単位

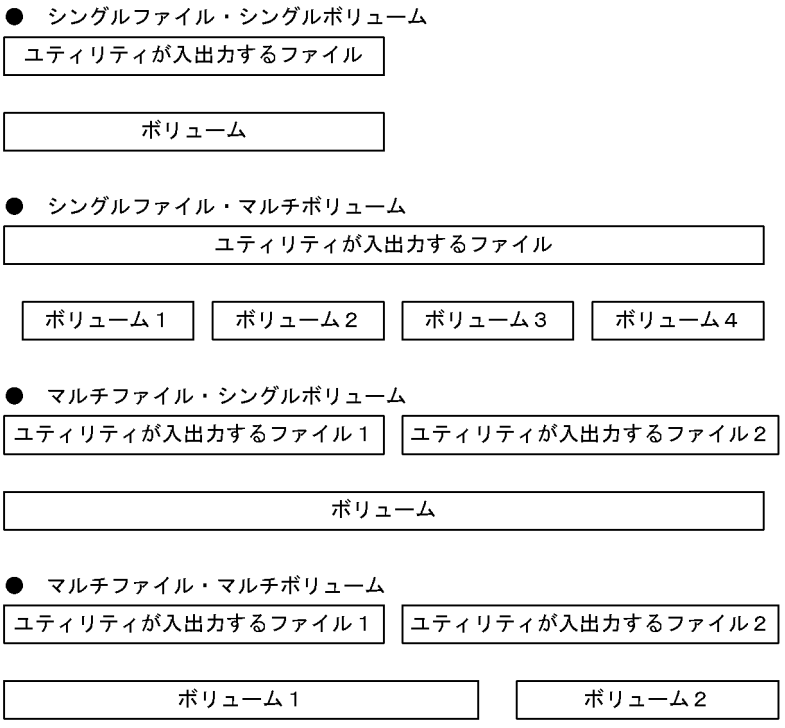
ファイルの場合，一つのファイルを指します。テープ装置を扱う場合は，各テープ媒体のことを指します。 HiRDB ファイルシステムでは，一つの HiRDB ファイルを指します。

通常一つの「ファイル」は一つの「ボリューム」の中に格納されます（シングルファイル・シングルボリューム）。しかし，「ファイル」が「ボリューム」に比べて大きい場合，「ファイル」を複数のボリュームに分割して格納することになります（シングルファイル・マルチボリューム）。それとは逆に「ファイル」が「ボリューム」に比べて小さい場合，フォーマット上できるならば，複数の「ファイル」を一つの「ボリューム」上にまとめて格納することになります(マルチファイル・シングルボリューム）。

また，「マルチファイル・シングルボリューム」に「ファイル」を追加してゆくと最後の「ファイル」が次のボリュームにわたって格納されることがあります（マルチファイル・マルチボリューム）。

「ファイル」と「ボリューム」の概念図を次の図に示します。

図 1–3 「ファイル」と「ボリューム」の概念図



1.4.2　ユティリティで使用できるファイルの形態

HiRDB の各ユティリティの入出力ファイルは，特に明記しないかぎりファイルを対象とし，その形態は「シングルファイル・シングルボリューム」となります（2^{63}バイトを超えるファイルは操作できません）。

また，ネットワークドライブのファイルは，ユティリティの入出力ファイルとして使用できません。ローカルドライブ上のファイルを使用してください。

ユティリティによっては「シングルファイル・シングルボリューム」以外の形態のファイルを扱えます。ファイル「シングルファイル・シングルボリューム」以外の入出力ができるユティリティの一覧を次の表に示します。

表 1–3　ファイル「シングルファイル・シングルボリューム」以外の入出力ができるユティリティ一覧

<table>
<tr><th>ユティリティ名</th><th>ファイル種別</th><th>シングルファイルマルチボリューム</th><th>テープ装置</th><th>HiRDB ファイル</th></tr>
<tr><td>データベース作成ユティリティ（pdload）</td><td>入力データファイル，LOB 入力ファイル，列単位 LOB 入力ファイル</td><td rowspan="4">○</td><td>△※</td><td>×</td></tr>
<tr><td>データベース再編成ユティリティ（pdrorg）</td><td>アンロードデータファイル，LOB データのアンロードファイル</td><td rowspan="4">○※</td><td>○
-f hirdb</td></tr>
<tr><td>データベース複写ユティリティ（pdcopy）</td><td rowspan="2">バックアップファイル</td><td rowspan="3">○
-k i</td></tr>
<tr><td>データベース回復ユティリティ（pdrstr）</td></tr>
<tr><td>ディクショナリ搬出入ユティリティ（pdexp）</td><td>搬出ファイル</td><td>×</td><td>×</td></tr>
</table>

（凡例）

○：

使用できます。コマンドラインでの指定が必要な場合は，○の下にオプションフラグを記述してあります。その場合，このコマンドオプション以外にも各ファイルに固有の情報の指定が必要となります。詳細は，各コマンドの該当オプションフラグの説明を参照してください。

△：

可変長ブロックモードは使用できます，固定長ブロックモードでは DAT 形式が使用できます。バイナリ形式は使用できません。

×：

未対応です。

注※

64 ビットモードの場合は使用できません。

(1)　テープ装置

ユティリティで使用できるテープ装置を次の表に示します。

表 1–4　ユティリティで使用できるテープ装置

装置名称	ユティリティでの直接入出力
CMT	×
DAT	○
OMT	×
CGMT	×
8mm	×
DLT	○
LTO	○

（凡例）

○：使用できます。

×：使用できません。

ユティリティがテープ装置に対して直接入出力をする場合，シングルボリュームとして扱うため，装置タイプは単体の Tape Drive としてアクセスします。Autoloader や Tape Library の制御はしません。

テープ装置では可変長ブロックモードで使用することをお勧めします。

固定長ブロックモードでは 512 バイト単位で入出力をしますが，可変長ブロックモードでは 32 キロバイト単位で入出力をするため，入出力回数が削減できます。固定長ブロックモードを使用する場合は，ブロックサイズが 512 バイトであることを確認してください。

各テープ装置については，OS のマニュアルを参照してください。

pdrorg，pdload，又は pdrstr でテープ装置を使用する場合，デバイスドライバのマイナー番号に，クローズ時に巻き戻さない指定はしないでください。巻き戻しをしないでユティリティを実行した場合，ユティリティ終了後に mt コマンド（磁気テープ操作プログラム）などでテープを巻き戻す必要があります。

一つの媒体に複数のバックアップを保存したい場合は，いったんユティリティでバックアップをファイルに出力して，それから一つの媒体へコピーするなどしてください。

(2)　HiRDB ファイル

入出力ファイルとして HiRDB ファイルを使用する場合は，事前に pdfmkfs コマンドで-k UTL を指定して，HiRDB ファイルシステムを作成し，HiRDB ユティリティ実行時に表「ファイル「シングルファイル・シングルボリューム」以外の入出力ができるユティリティ一覧」のオプションを指定する必要があります。

HiRDB ユティリティは HiRDB ファイルを次の形態で使用します。

- シングルファイル・シングルボリューム
- シングルファイル・マルチボリューム

HiRDB ファイルについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

1.4.3　テープ装置アクセス機能

DAT，DLT，又は LTO 上にあるファイルをテープ装置アクセス機能を使用してアクセスできます。

(1)　テープ装置アクセス機能を適用できるファイル

次に示すファイルに対してテープ装置アクセス機能が適用されます。

- 入力データファイル（pdload の source 文に指定する入力データファイル）
- アンロードデータファイル（pdrorg の unload 文に指定するアンロードデータファイル）
- LOB データのアンロードファイル（pdrorg の lobunld 文に指定する LOB データのアンロードファイル）
- バックアップファイル（pdcopy，及び pdrstr の-b オプションに指定するバックアップファイル）

(2)　ファイルの指定方法

DAT，DLT，又は LTO 上のファイルを指定する場合は，ファイル名に「¥¥.¥tape[n]」を指定してください。n はテープドライブ番号を示しています。 n を省略した場合は，0 が仮定されます。

［管理ツール］−［コンピュータの管理］を起動して，［記憶域］−［リムーバブル記憶域］−［物理的な場所］にあるテープ装置のプロパティの［デバイス情報］で確認できます。

(3)　ハードウェア圧縮機能

DAT，DLT，又は LTO アクセス時のハードウェア圧縮機能の使用を指定できます。ただし，この設定はハードウェア圧縮をサポートしている DAT，DLT，又は LTO に限り有効です。

指定値の参照及び更新は，pdntenv コマンドを使用します。インストール直後は圧縮をする指定（on）が設定されています。

1.5　コマンド実行時の留意事項

1.5.1　コマンド実行中の割り込みによる強制終了

コマンド実行中は，シグナル割り込み（例えば，CTL ＋ C，CTL ＋¥，pdkill コマンドなど）を使用してコマンドを強制終了させないでください。タイミングによっては，HiRDB が終了することがあります。ユティリティについては，pdcancel コマンドで終了させてください。

1.5.2　運用コマンド，ユティリティでの RD エリアの指定

運用コマンド，ユティリティでの RD エリアの指定について説明します。

(1)　RD エリアの指定

運用コマンド，及びユティリティで RD エリアを指定する場合，次の方法が選択できます。

- RD エリア名を指定する（直接指定）
- 複数の RD エリアをグループ化して指定する（一括指定）
- 運用コマンド又はユティリティの引数に ALL を指定する（ALL 指定）

(a)　RD エリア名を指定する（直接指定）

RD エリア名をコマンドラインで直接指定できます。

(b)　複数の RD エリアをグループ化して指定する（一括指定）

運用コマンドで指定する RD エリアは，次の方法で一括指定できます。これによって，複数の RD エリアをグループ化して指定できます。

表 1–5　RD エリアを一括指定する方法

項目	パターン文字列の例	意味	指定例
前方一致	nnn*	先頭部分が nnn の RD エリアが対象となります。	RDUSER* （RDUSER で始まる RD エリアが対象となります）
	nnn*mmm*	先頭部分が nnn，任意の部分に mmm を含む RD エリアが対象となります。	RD*USER* （RD で始まり，USER を含む RD エリアが対象となります）
	nnn*mmm	先頭部分が nnn，最後の部分が mmm の RD エリアが対象となります。	RD*USER （RD で始まり，USER で終わる RD エリアが対象となります）

項目	パターン文字列の例	意味	指定例
中間一致	*nnn*	任意の部分に nnn を含む RD エリアが対象となります。	*RDUSER* （RDUSER を含む RD エリアが対象となります）
	*nnn*mmm*	任意の部分に nnn と mmm を含む RD エリアが対象となります。	*RD*USER* （RD と USER を含む RD エリアが対象となります）
後方一致	*nnn	最後の部分が nnn の RD エリアが対象となります。	*RDUSER （RDUSER で終わる RD エリアが対象となります）
	*mmm*nnn	任意の部分に mmm を含み，最後の部分が nnn の RD エリアが対象となります。	*RD*USER （RD を含み，USER で終わる RD エリアが対象となります）
	.nnn	.の後に nnn で終わる RD エリアが対象となります。pdcopy 及び pdrstr の場合だけ指定できます。	..RDUSER （任意の 2 文字の後が RDUSER で終わる，8 文字の RD エリアが対象となります）

(c)　運用コマンド又はユティリティの引数に ALL を指定する（ALL 指定）

ALL を指定すると，すべての RD エリアに対して処理を実行します。

ALL は，引数の先頭に記載してください。また，ALL を指定したときは，同時に直接指定と一括指定を指定することはできません。

(2)　RD エリアを指定する場合の規則

1. 運用コマンド，又はユティリティによって，直接指定，一括指定，ALL 指定の指定可否が異なります。直接指定，一括指定，ALL 指定の指定可否を次の表に示します。

表 1–6　直接指定，一括指定，ALL 指定の指定可否

運用コマンド，及びユティリティ名	指定方法		
	直接指定	一括指定	ALL 指定
pdhold	○	○	○
pdrels	○	○	○
pdopen	○	○	○
pdclose	○	○	○
pddbls	○	○	○
pddbfrz	○	○	○

運用コマンド，及びユティリティ名		指定方法		
		直接指定	一括指定	ALL 指定
pdmemdb		○	○	○
pdbufmod		○	×	×
pdrdrefls		○	○	○
pdrorg		○	×	×
pdreclaim		○	×	×
pdpgbfon		○	×	×
pddbst	再編成時期予測	×	×	○
	状態解析結果蓄積	○	○	○
	RD エリア論理解析	○	○	○
	RD エリア物理解析	○	○	○
pdcopy		○	○	×
pdrstr		○	○	×

（凡例）

○：指定できます。

×：指定できません。

2. RD エリアを指定する場合，文字列長の上限は，OS やバッチファイルの指定文字列長の制限に従います。その値を超えるとエラーとなり，運用コマンド及びユティリティは実行されません。
3. RD エリア名を重複して指定した場合，重複排除して処理されます。ただし，次の運用コマンド又はユティリティを指定した場合は，重複排除しないでエラーとなります。
   - pdrorg
   - pdreclaim
   - pdpgbfon
4. 指定できる RD エリア数の上限値を次の表に示します。重複排除された結果の値と処理対象として指定できる RD エリア数です。なお，重複排除された RD エリアの数が上限値を超えるとエラーとなります。

表 1–7　指定できる RD エリア数の上限値

運用コマンド，及びユティリティ名	重複排除された RD エリア数の上限値	処理できる RD エリア数の 上限値
pdhold	4,096	上限なし
pdrels	4,096	上限なし
pdopen	4,096	上限なし
pdclose	4,096	上限なし

運用コマンド，及びユティリティ名		重複排除された RD エリア数の上限値	処理できる RD エリア数の 上限値
pddbls		4,096	上限なし
pddbfrz		4,096	上限なし
pdmemdb		4,096	上限なし
pdbufmod		4,096	4,096
pdrdrefls		上限なし	上限なし
pdrorg		1	1
pdreclaim		1	1
pdpgbfon		1	1
pddbst	再編成時期予測	1	上限なし
	状態解析結果蓄積	4,096	上限なし
	RD エリア論理解析	4,096	上限なし
	RD エリア物理解析	4,096	上限なし
pdcopy		上限なし	上限なし
pdrstr		上限なし	上限なし

(3) 注意事項

1. パターン文字列中に指定する「*」及び「.」についての注意事項を示します。
   - pdcopy，及び pdrstr の場合
     パターン文字列中に指定する「*」及び「.」は，任意の位置で任意の個数指定できます。ただし，「*」だけのパターン文字列，「.」だけのパターン文字列，及び「*」と「.」だけのパターン文字列は指定しないでください。
   - pdcopy，及び pdrstr 以外の場合
     パターン文字列中に指定する「*」は，任意の位置で任意の個数指定できます。ただし，「*」だけのパターン文字列は指定しないでください。
2. パターン文字列中には，「%」を含めないでください。
3. RD エリア名，及びパターン文字列中の英字の大文字と小文字を区別する場合，全体を「¥""」と「"¥"」で囲んでください。「¥""」と「"¥"」で囲まない場合はすべて大文字として扱われます。また，空白を含む場合も「¥""」と「"¥"」で囲んでください。
   ただし，pdcopy，及び pdrstr にパターン文字列を指定した場合は，「¥""」と「"¥"」の有無に関係なく，文字を大文字と小文字に区別します。
4. 一括指定でグループ化された RD エリアと，ほかの RD エリア名の指定（ほかの一括指定でグループ化された RD エリア名も含む）が重複した場合，複数回処理をするかどうかについては運用コマンド及び

ユティリティによって異なります。RD エリア名が重複した場合の運用コマンド及びユティリティの処理方法を次に示します。

表 1–8　RD エリア名が重複した場合の運用コマンド及びユティリティの処理方法

<table>
<tr><th colspan="2">運用コマンド，及びユティリティ名</th><th>重複した場合の処理</th></tr>
<tr><td colspan="2">pdhold</td><td>重複排除しないで複数回処理を実行します。</td></tr>
<tr><td colspan="2">pdrels</td><td rowspan="4">重複排除しないで複数回処理を実行します。</td></tr>
<tr><td colspan="2">pdopen</td></tr>
<tr><td colspan="2">pdclose</td></tr>
<tr><td colspan="2">pddbls</td></tr>
<tr><td colspan="2">pddbfrz</td><td>重複排除しないで複数回処理を実行します。</td></tr>
<tr><td colspan="2">pdmemdb</td><td>重複排除します。</td></tr>
<tr><td colspan="2">pdbufmod</td><td>該当しません。</td></tr>
<tr><td colspan="2">pdrdrefls</td><td>重複排除します。</td></tr>
<tr><td colspan="2">pdrorg</td><td rowspan="5">該当しません。</td></tr>
<tr><td colspan="2">pdreclaim</td></tr>
<tr><td colspan="2">pdpgbfon</td></tr>
<tr><td rowspan="4">pddbst</td><td>再編成時期予測</td></tr>
<tr><td>状態解析結果蓄積</td><td rowspan="5">重複排除します。</td></tr>
<tr><td>RD エリア論理解析</td></tr>
<tr><td>RD エリア物理解析</td></tr>
<tr><td colspan="2">pdcopy</td></tr>
<tr><td colspan="2">pdrstr</td></tr>
</table>

5. 指定するパターン文字列の文字数の合計が指定可能な RD エリア名称の長さ（30 文字）を超えないようにしてください。

1.5.3　コマンド実行結果を DAT 形式で出力する場合の規則

コマンドの実行結果を DAT 形式で出力する場合の規則を次の表に示します。

表 1–9　コマンド実行結果を DAT 形式で出力する場合の規則

コマンド	オプション	各要素をコンマ（,）で区切って表示する	文字要素を引用符（"）で囲んで表示する	数値要素をそのまま表示する	空要素（文字要素）を二つの連続した引用符（"）で表示する	空要素（数値要素）の値を表示しない
pdbufls	−	○	○	○	○	○
pddbls	−	○	○	○	○	○
pddbst	-k logi	○	○※1	○	○	○
	-k phys	○	○※1	○	−※2	○
	-k pred	○	○※1	○	−※2	○
pdbkupls	−	○	−※3	○	−※2	○
pdfstatfs	−	○	○	○	○	○
pdlogls	-d sys	○	○	○	○	○
pdlogucat	−	○	−※3	○	−※2	○
pdls	-d prc	○	○	○	○	○
	-d svr	○	○	−※4	○	−※4
	-d trn	○	○	○	○	○
	-d ust	○	○	−※4	○	−※4
pdobils	−	○※5	−	○	−	○
pdstedit	−	○	○※1	○	○	○
pdsvhostname	−	−※6	○	−※4	−※2	−※4

（凡例）

○：該当します。

−：該当しません。

注

要素，文字要素，数値要素，及び空要素について説明します。

要素：要素には，文字要素，数値要素，及び空要素があります。

文字要素：出力形式が文字列の要素を表します。

数値要素：出力形式が数値の要素を表します。

空要素：情報がない要素を表します。

注※1　先頭にタイトル行を出力しますが，タイトル行自体は引用符（"）で囲まれません。

注※2　数値要素と同じで，値は表示されません。

注※3　数値要素と同じで，そのまま表示されます。

注※4　数値要素がないため該当しません。

注※5　-C オプションの区切り文字にコンマを指定した場合，各要素がコンマで区切られて表示されます。

注※6　要素が一つだけのため該当しません。

1.5.4　OS ログインユーザの簡易認証機能使用時の規則

OS ログインユーザの簡易認証機能を使用する場合の規則を示します。

(1)　HiRDB 接続時の規則

HiRDB に接続する場合，環境変数 PDUSER の認可識別子及びパスワードに簡易認証キーワードを指定し，-u オプションは指定しないでください。

(2)　認可識別子を省略した場合の規則

コマンドライン及び制御文に指定する表名，インデクス名，抽象データ型名，関数名，順序数生成子で認可識別子を省略した場合，仮定される認可識別子は，ログインウィンドウのユーザ名になります。

(3)　制限事項

OS ログインユーザの簡易認証機能を使用した場合，次の表のとおりエラーとなります。なお，プラグインユティリティについては，各種プラグイン製品のマニュアルを参照してください。

表 1–10　OS ログインユーザの簡易認証機能使用時にエラーとなるユティリティ一覧

ユティリティ	OS ログインユーザの簡易認証機能による実行時の動作
pdrorg を用いたエクスポート	KFPX18430-E でエラー終了
pdgetcst	KFPN00104-E でエラー終了
pdexp	KFPX28448-E でエラー終了
pddefrev	
pdplgrgst	KFPY02230-E でエラー終了

1.5.5　環境変数に設定したクライアント環境変数の制限事項

OS 又はユーザの環境変数に設定したクライアント環境変数については，実行する運用コマンド，又はユティリティによって次に示すクライアント環境変数だけ，指定を有効とします。ほかのクライアント環境変数については，指定があっても無効として動作します。クライアント環境変数については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

■指定を有効とするクライアント環境変数（その 1）

- 対象ユティリティ名
  pdaudatld
  pdaudput
  pdaudswap

pdconstck
pddbst
pddefrev
pdexp
pdextfunc
pdgetcst
pdload
pdls(-d aud)
pdorbegin
pdorcheck
pdorcreate
pdorend
pdparaload
pdpgbfon
pdrbal
pdrdrefls
pdreclaim
pdreginit
pdrorg
プラグインが提供するコマンド

- 指定を有効とするクライアント環境変数
  PDUSER
  PDLANG
  PDPLUGINNSUB
  PDDBBUFLRU
  PDCLTBINDLOOPBACKADDR
  PDTIMEDOUTRETRY
  PDDDLDEAPRPEXE
  PDLCKWAITTIME
  PDBINDRETRYCOUNT
  PDBINDRETRYINTERVAL
  PDCLTSNDADDR

■指定を有効とするクライアント環境変数（その 2）

- 対象ユティリティ名
  pdplgrgst

- 指定を有効とするクライアント環境変数
  PDUSER
  PDLANG
  PDPLUGINNSUB
  PDHOST
  PDNAMEPORT
  PDCLTGRP
  PDDBBUFLRU
  PDCLTBINDLOOPBACKADDR
  PDCWAITTIME
  PDSWAITTIME
  PDSWATCHTIME
  PDTIMEDOUTRETRY
  PDDDLDEAPRPEXE
  PDLCKWAITTIME
  PDBINDRETRYCOUNT
  PDBINDRETRYINTERVAL
  PDAUTHTYPE
  PDIPCFILEDIR
  PDCLTSNDADDR

■指定を有効とするクライアント環境変数（その 3）

- 対象ユティリティ名
  pddef
- 指定を有効とするクライアント環境変数
  PDUSER
  PDLANG
  PDPLUGINNSUB
  PDHOST
  PDNAMEPORT
  PDCLTRCVADDR
  PDDBLOG
  PDCLTGRP
  PDDBBUFLRU
  PDCLTBINDLOOPBACKADDR
  PDCWAITTIME

PDSWAITTIME
PDSWATCHTIME
PDTIMEDOUTRETRY
PDDDLDEAPRPEXE
PDLCKWAITTIME
PDDELRSVWDFILE
PDBINDRETRYCOUNT
PDBINDRETRYINTERVAL
PDCNSTRNTNAME
PDTMPTBLRDAREA
PDAUTHTYPE
PDIPCFILEDI
PDCLTSNDADDR

1.5.6　一般ユーザによるコマンド実行時の注意事項

HiRDB 管理者以外のユーザがユティリティを実行する場合，ユティリティのオプション，及び制御文で指定するファイル出力先ディレクトリに，ユティリティ実行ユーザ又は HiRDB グループに対するフルコントロールを許可するアクセス権を事前に付与してください。権限が付与されていない場合，次の表に示すファイルが残ることがあります。該当するファイルについては，HiRDB 管理者が削除してください。

表 1–11　一般ユーザがコマンドを実行した場合に残る可能性があるファイル

実行したユティリティ	一時ファイル出力先	残る可能性があるファイルの名称
定義系 SQL の生成（pddefrev）	システム定義の pd_tmp_directory オペランドに指定，又は仮定されるワークファイル出力先ディレクトリ	EXP 文字列 （例：EXPxOsUpvb）
ディクショナリ搬出入ユティリティ（pdexp）	システム定義の pd_tmp_directory オペランドに指定，又は仮定されるワークファイル出力先ディレクトリ	EXP 文字列 （例：EXPxOsUpvb）
データベース状態解析ユティリティ（pddbst）	制御文ファイル（workdir 文）に指定，又は仮定されるワークファイル出力先ディレクトリ	数値.dbst.stor 数値.dbst.data 数値.dbst.msg 数値.dbst.head （例：19663.dbst.stor）

1.6　HiRDB の付加 PP を導入すると使用できる機能

HiRDB の付加 PP を次の表に示します。HiRDB の付加 PP を導入すると次の表に示す機能を使用できます。

表 1–12　HiRDB の付加 PP

付加 PP の製品名	付加 PP を導入すると使用できる機能
HiRDB Advanced High Availability	• 1:1 スタンバイレス型系切り替え機能 • 影響分散スタンバイレス型系切り替え機能 • グローバルバッファの動的変更（pdbufmod コマンド） • システム構成変更コマンド（pdchgconf コマンド） • 表のマトリクス分割 • 分割格納条件の変更（ALTER TABLE）
HiRDB Non Recover FES	回復不要 FES
HiRDB Accelerator	インメモリデータ処理によるバッチ高速化機能

2 運用コマンド

この章では，運用コマンドの文法について説明します。

2.1 pdacunlck（連続認証失敗アカウントロック状態の解除）

2.1.1 pdacunlck の形式と規則

(1) 機能

設定したアカウントロック期間を経過していない認可識別子，又は無期限の連続認証失敗アカウントロック状態となっている認可識別子の，連続認証失敗アカウントロック状態を解除します。

pdacunlck コマンドを実行すると，ディクショナリ表 SQL_USERS の連続認証失敗アカウントロック日時がナル値でない認可識別子の，連続認証失敗アカウントロック日時，及び連続認証失敗回数をナル値に設定します。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdacunlck {認可識別子〔, 認可識別子〕… | ALL}
          〔-W 実行監視時間〕
```

(4) コマンド引数

(a) 認可識別子〔，認可識別子〕...

連続認証失敗アカウントロック状態を解除する認可識別子を指定します。認可識別子は，最大 128 個指定できます。

(b) ALL

連続認証失敗アカウントロック状態となっているすべての認可識別子を解除する場合に指定します。

ALL を指定して pdacunlck コマンドを実行した場合，ユーザ権限情報用バッファに CONNECT 権限の情報がなく，かつ連続認証失敗アカウントロック日時がナル値でないユーザは，コマンド実行中は HiRDB に接続できません。

(c) -W 実行監視時間 ～<符号なし整数>((0～3600))

pdacunlck コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5) 規則

1. pdacunlck コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdacunlck コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. データディクショナリ用 RD エリアが閉塞している場合，pdacunlck コマンドは実行できません。

(6) 注意事項

1. 認可識別子を指定した場合，ディクショナリ表 SQL_USERS の行ごとに排他モードで排他を掛けます。また，ALL を指定した場合，連続認証失敗アカウントロック日時がナル値でないユーザに対して，ディクショナリ表 SQL_USERS の行ごとに排他モードで排他を掛けます。
2. pdacunlck コマンド実行時のリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
3. 認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

2.2 pdadmvr（HiRDB の製品情報の取得）

2.2.1 pdadmvr の形式と規則

(1) 機能

HiRDB の製品情報を表示します。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdadmvr 〔-s | -c〕
```

(4) オプション

(a) -s

バージョン情報にリビジョン以下のコードを表示させる場合に指定します。バージョン情報の表示例を次に示します。

- **オプションなしの場合：**09-00
- **-s オプションありの場合：**09-00-01

(b) -c

オプション指定なし又は-s オプション指定時に表示される情報に加えて，文字コード種別の情報が表示されます。

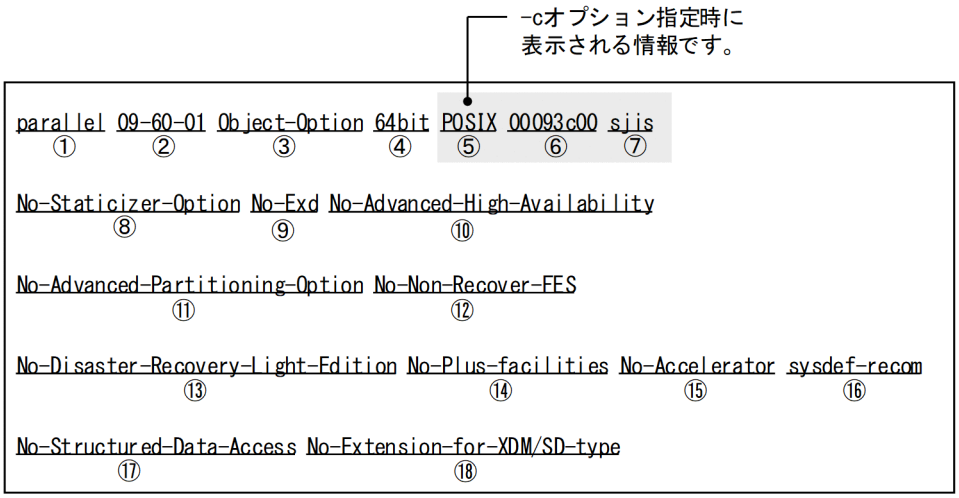
(5) 規則

1. pdadmvr コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdadmvr コマンドは，各サーバマシンで実行してください。

(6) 注意事項

1. pdadmvr コマンド実行時のリターンコードは，0 の場合は正常終了，8 の場合は異常終了（オプション指定不正）となります。

(7)　出力形式



図中の丸付き数字が項番に対応しています。

項番	表示項目	表示内容と意味
1	HiRDB の種類	• **single**：HiRDB/シングルサーバ • **parallel**：HiRDB/パラレルサーバ
2	HiRDB のバージョン	HiRDB のバージョン オプションの指定によって表示形式が異なります。 • オプション指定なしの場合 リビジョンまでを表示します。 （例）09-00 • -s 又は-c オプション指定の場合 リビジョン以下のコードまでを表示します。 （例）09-00-01
3	Object-Option	常に Object-Option が表示されます。
4	HiRDB のアドレッシングモード	常に 64bit が表示されます。 • **64bit**：64 ビットアドレッシングモード
5	HiRDB の内部情報	常に POSIX が表示されます。
6	HiRDB の内部情報	－
7	文字コード種別	• **sjis**：シフト JIS 漢字コード • **chinese**：EUC 中国語漢字コード • **chinese-gb18030**：中国語漢字コード（GB18030） • **utf-8**：Unicode（UTF-8） • **utf-8_ivs**：Unicode（IVS 対応 UTF-8） • **lang-c**：単一バイト文字コード
8	No-Staticizer-Option	常に No-Staticizer-Option が表示されます。

項番	表示項目	表示内容と意味
9	HiRDB External Data Access のセットアップ有無※1	常に No-Exd が表示されます。
10	HiRDB Advanced High Availability のセットアップ有無	• **Advanced-High-Availability**：あり • **No-Advanced-High-Availability**：なし
11	HiRDB Advanced Partitioning Option のセットアップ有無	• **Advanced-Partitioning-Option**：あり • **No-Advanced-Partitioning-Option**：なし HiRDB Advanced Partitioning Option の機能は HiRDB Advanced High Availability に包含されるため，HiRDB Advanced High Availability のセットアップ有無と同じになります。
12	HiRDB Non Recover FES のセットアップ有無※2	• **Non-Recover-FES**：あり • **No-Non-Recover-FES**：なし
13	No-Disaster Recovery Light Edition	常に No-Disaster-Recovery-Light-Edition が表示されます。
14	HiRDB Data Convert Type1 Option のセットアップ有無	• **No-Plus-facilities**：なし • **Plus-facilities**：あり
15	HiRDB Accelerator のセットアップ有無	• **Accelerator**：あり • **No-Accelerator**：なし
16	オペランド省略時動作	• **sysdef-recom**：推奨モード • **sysdef-v0904**：0904 互換モード
17	HiRDB Structured Data Access Facility かどうか	• **No-Structured-Data-Access**：HiRDB Structured Data Access Facility ではない • **Structured-Data-Access**：HiRDB Structured Data Access Facility である
18	HiRDB Structured Data Access Facility Extension for XDM/SD type のセットアップ有無	• **Extension-for-XDM/SD-type**：あり • **No-Extension-for-XDM/SD-type**：なし

注※1　HiRDB/パラレルサーバの場合，又は HiRDB/シングルサーバで-c オプションを指定した場合に出力されます。

注※2　HiRDB/パラレルサーバの場合に出力されます。

2.3 pdaudatld（監査証跡表の自動データロード機能の制御）

2.3.1 pdaudatld の形式と規則

(1) 機能

監査証跡表の自動データロード機能を制御します。次の制御ができます。

- 停止した監査証跡表の自動データロード機能を再開始します。
- 実行中の監査証跡表の自動データロード機能を停止します。
- 監査証跡表の自動データロード機能の動作状態と，自動データロード対象の監査証跡ファイルがあるユニットの一覧を表示します。

(2) 実行者

HiRDB 管理者でコマンドを実行し，HiRDB へアクセスする際の認証には監査人の認可識別子とパスワードを指定してください。

(3) 形式

```
pdaudatld 〔-U 認可識別子〕 〔-b | -t | -i〕
```

(4) オプション

(a) -U 認可識別子

監査人の認可識別子を指定します。このオプションを指定すると，パスワード入力要求の応答付きメッセージが出力されます。

＜規則＞

1. このオプションに指定した認可識別子と，応答付きメッセージに入力したパスワードで HiRDB に接続し，その認可識別子に監査権限があるかどうかをチェックします。
2. このオプションは，応答付きメッセージに応答できる環境で pdaudatld コマンドを実行する場合に指定できます。
3. このオプションを省略する場合は，環境変数 PDUSER を設定しておく必要があります。このオプションを省略した場合，pdaudatld コマンド実行時の環境変数 PDUSER の設定値が仮定されます。このオプションを省略し，かつ環境変数 PDUSER も設定していない場合，pdaudatld コマンドは実行できません。指定例を次に示します。

   （例）

   PDUSER=¥""認可識別子"¥"/¥""パスワード"¥"

(b) -b

監査証跡表の自動データロード機能を再開始します。データロード待ち状態になっていた監査証跡ファイルのデータロードを順次実行します。

(c) -t

監査証跡表の自動データロード機能を停止します。このオプションを指定して pdaudatld コマンドを実行すると，監査証跡ファイルの世代がスワップしても，監査証跡表の自動データロードは行いません。

このオプションを指定して pdaudatld コマンドを実行した時に監査証跡表の自動データロードが実行中だった場合，そのデータロードが完了するまで待ってから自動データロード機能を停止します。詳細については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(d) -i

監査証跡表の自動データロード機能の動作状態と，自動データロード対象の監査証跡ファイルがあるユニットの一覧を表示します。

(5) 規則

1. pdaudatld コマンドは，HiRDB システムが稼働中の場合だけ実行できます。ただし，pdstart -r で HiRDB を起動している場合は実行できません。
2. pdaudatld コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdaudatld コマンドは，システム定義に pd_aud_file_name オペランドを指定していない場合，又は pd_aud_auto_loading オペランドに Y 以外を指定している場合，コマンドエラーとなります。
4. pdaudatld -b 又は-t コマンド実行中に pdaudbegin，pdaudend，pdaudrm，pdaudswap，pdaudatld -b 又は-t コマンドを実行すると，コマンドエラーとなります。
5. オプションをすべて省略した場合，pdaudatld は-i オプションが指定されたものとして動作します。

(6) 注意事項

1. pdaudatld コマンドのリターンコードを次に示します。

   0：正常終了

   4：警告終了

   pdaudatld -t コマンドで自動データロード処理が完了するのを待っているときに自動データロードでエラーが発生したため，自動データロード完了とみなして自動データロード機能を停止したことを示します。

   8：異常終了

(7)　出力形式

```
Hostname : aa....aa (bb....bb)
Auto-Loading-Service-Status : cc....cc
unit  Status    Dataload-Request-Time  Count  Dataload-Start-Time
dddd  ee....ee  ffff/gg/hh ii:jj:kk      lll  mmmm/nn/oo pp:qq:rr
dddd  ee....ee  ffff/gg/hh ii:jj:kk      lll
   :                :          :          :
dddd  ee....ee  ffff/gg/hh ii:jj:kk      lll
```

[説明]

aa....aa：

pdaudatld コマンドを実行したユニットの現用系ホスト名称（32 文字）

bb....bb：

コマンド実行時刻（年（西暦 4 けた）/月/日 時:分:秒の形式）

cc....cc：

監査証跡表の自動データロード機能の動作状態（16 文字以内）

ENABLE：有効

DISABLE（WAIT）：停止処理中

DISABLE：停止

dddd：

自動データロード対象の監査証跡ファイルがあるユニットのユニット識別子（4 文字）

ee....ee：

監査証跡表の自動データロード機能によるデータロードの実行状態

loading：自動データロード中

standby：データロード待ち状態

skip：ユニット停止中のため，自動データロードをスキップした状態

ffff/gg/hh ii:jj:kk：

自動データロード対象になった時刻（年（西暦 4 けた）/月/日 時:分:秒の形式）

lll：

自動データロードの要求を受けた回数（3 文字）

自動データロード対象になった時点では 1 です。ee....ee が「standby」又は「skip」で，同一ユニットから自動データロード要求を受けると 1 ずつ増加します。システムマネジャがあるユニットを停止して再起動した場合は，1 に戻ります。

mmmm/nn/oo pp:qq:rr：

自動データロード機能でデータロードを開始した時刻（年（西暦 4 けた）/月/日 時:分:秒の形式）

ee....ee に「loading」が表示されている場合だけ表示します。

注

- 自動データロード対象の監査証跡ファイルが 1 件もない場合，先頭 1 行だけ表示します。
- dddd から mmmm/nn/oo pp:qq:rr までは，自動データロード中及びデータロード待ち状態のユニットの数だけ表示します。複数行表示されている場合，自動データロードはこの表示順に実行します。自動データロードが終了したユニットは，これらの項目が表示されなくなります。
- データロード待ち状態のユニットで，新たに監査証跡ファイルのスワップが発生した場合，自動データロード時にまとめて監査証跡表にデータロードします。
- 自動データロード中のユニットで，新たに監査証跡ファイルのスワップが発生した場合，同一ユニットで「loading」と「standby」の 2 行を表示することがあります。このとき，実行中のデータロードが KFPS05716-W メッセージを表示してリトライしたり，KFPS05713-E メッセージを表示して自動データロードを停止すると，表示は 1 行になります。自動データロードの要求を受けた回数（lll）には，2 行の値が合算された値を表示します。表示例を次に示します。

  ＜例＞

  「loading」と「standby」の 2 行が表示されています。

```
unit  Status    Dataload-Request-Time  Count  Dataload-Start-Time
unt1  loading   2007/03/01 12:34:56        2  2007/03/01 12:35:01
unt1  standby   2007/03/01 12:38:02        1
```

  実行中のデータロードが KFPS05716-W メッセージを表示してリトライすると，次のように表示が 1 行になります。

```
unit  Status    Dataload-Request-Time  Count  Dataload-Start-Time
unt1  standby   2007/03/01 12:34:56        3
```

2.4　pdaudbegin（監査証跡の取得開始）

2.4.1　pdaudbegin の形式と規則

(1)　機能

監査証跡の取得を開始します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdaudbegin
```

(b)　HiRDB/パラレルサーバの場合

```
pdaudbegin 〔-u ユニット識別子〕
```

(4)　オプション

(a)　-u ユニット識別子　～＜識別子＞((4 文字))

監査証跡の取得を開始するユニットのユニット識別子を指定します。

-u オプションを省略した場合，すべてのユニットで監査証跡の取得を開始します。

(5)　規則

1. pdaudbegin コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdaudbegin コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdaudbegin コマンド実行中に，pdaudbegin，pdaudend，pdaudrm，及び pdaudswap コマンドを実行するとコマンドエラーになります。
4. pdaudbegin コマンドは，システム定義に pd_aud_file_name オペランドを指定していない場合，又は pd_aud_file_name オペランドに指定している監査証跡ファイルがない場合，コマンドエラーとなります。

5. HiRDB/パラレルサーバで，-u オプションを省略して pdaudbegin コマンドを実行した場合，コマンド処理中にエラーが発生したユニット，及び監査証跡の取得が既に開始されているユニットについては，エラーメッセージを出力します。エラーにならなかったユニットについては監査証跡の取得を開始します。

(6) 注意事項

1. pdaudbegin コマンドの結果は，pdls -d aud コマンド，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：正常終了（監査証跡取得開始済みのユニットがあり，処理をスキップしました）

   スキップしたユニット以外は，正常監査証跡取得を開始できているため，対処は必要ありません。

   8：異常終了（オプション指定不正，rsh 失敗など）

   監査証跡の取得開始ができていません。

   その状態で業務を継続すると監査証跡が取得されない問題が発生することがあります。

   メッセージに従って対処した後，再度コマンドを実行してください。

2. 監査証跡を取得中かどうかについては，pdls -d aud コマンドを実行して確認できます。

2.5　pdaudend（監査証跡の取得停止）

2.5.1　pdaudend の形式と規則

(1)　機能

監査証跡の取得を停止します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdaudend
```

(b)　HiRDB/パラレルサーバの場合

```
pdaudend 〔-u ユニット識別子〕
```

(4)　オプション

(a)　-u ユニット識別子　～＜識別子＞((4 文字))

監査証跡の取得を停止するユニットのユニット識別子を指定します。

-u オプションを省略した場合，すべてのユニットで監査証跡の取得を停止します。

(5)　規則

1. pdaudend コマンドは，HiRDB が稼働中で，かつ監査証跡の取得が開始されている場合に実行できます。
2. pdaudend コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdaudend コマンド実行中に，pdaudbegin，pdaudend，pdaudrm，及び pdaudswap コマンドを実行すると，コマンドエラーとなります。
4. pdaudend コマンドは，システム定義に pd_aud_file_name オペランドを指定していない場合，又は pd_aud_file_name オペランドに指定している監査証跡ファイルがない場合，コマンドエラーとなります。
5. HiRDB/パラレルサーバで，-u オプションを省略して pdaudend コマンドを実行した場合，コマンド処理中にエラーが発生したユニット，及び監査証跡の取得が既に停止しているユニットについては，エ

ラーメッセージを出力します。エラーとならなかったユニットについては，監査証跡の取得を停止します。

6. 監査証跡の取得を開始している場合は，pdaudend コマンドを実行しないかぎり，ユニットが停止するまで監査証跡を取得し続けます。

(6) 注意事項

1. pdaudend コマンドの結果は，pdls -d aud コマンド，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：正常終了（監査証跡取得停止済みのユニットがあり，処理をスキップしました）

   スキップしたユニット以外は，正常監査証跡取得を停止できているため，対処は必要ありません。

   8：異常終了（オプション指定不正，rsh 失敗など）

   監査証跡の取得停止ができていません。

   その状態で業務を継続すると次の問題が発生することがあります。

   pd_aud_no_standby_file_opr の指定が無い又は forcewrite を指定

   再実行しないで監査証跡を取得し続けた場合，使用できる監査証跡ファイルが無くなると，古い監査証跡から破棄します。

   pd_aud_no_standby_file_opr に down を指定

   再実行しないで監査証跡を取得し続けた場合，使用できる監査証跡ファイルが無くなると，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。

   メッセージに従って対処した後，再度コマンドを実行してください。

2.6　pdaudrm（閉塞中の監査証跡ファイルの削除）

2.6.1　pdaudrm の形式と規則

(1)　機能

閉塞中の監査証跡ファイルを削除します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdaudrm -g 監査証跡ファイル名 〔-f〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdaudrm -u ユニット識別子 -g 監査証跡ファイル名 〔-f〕
```

(4)　オプション

(a)　-u ユニット識別子　～＜識別子＞((4 文字))

削除する監査証跡ファイルがあるユニットのユニット識別子を指定します。

(b)　-g 監査証跡ファイル名

削除する監査証跡ファイルの名称を指定します。

(c)　-f

-g オプションで指定した監査証跡ファイルが，データロード待ちの状態であっても削除する場合に指定します。

監査証跡ファイルについては，pdls -d aud コマンドで確認できます。

(5)　規則

1. pdaudrm コマンドは，HiRDB 稼働中のときだけ実行できます。
2. pdaudrm コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

3. pdaudrm コマンド実行中に，pdaudbegin，pdaudend，pdaudrm，及び pdaudswap コマンドを実行すると，コマンドエラーとなります。
4. pdaudrm コマンドは，システム定義に pd_aud_file_name オペランドを指定していない場合，又は pd_aud_file_name オペランドに指定している監査証跡ファイルがない場合，コマンドエラーとなります。
5. 現用の監査証跡ファイルは削除できません。

(6) 注意事項

1. pdaudrm コマンドの結果は，pdls -d aud コマンド，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了となります。
2. 容量不足で生成に失敗した監査証跡ファイルを削除しようとすると，KFPS05752-E メッセージ（理由コード：FILE_ACCESS）でエラーとなりますが，対処は不要です。HiRDB を一度正常終了してから正常開始すると，生成に失敗した監査証跡ファイルは pdls -d aud コマンドで表示されなくなります。

2.7　pdaudswap（現用の監査証跡ファイルのスワップ）

2.7.1　pdaudswap の形式と規則

(1)　機能

現用の監査証跡ファイルをスワップします。

(2)　実行者

HiRDB 管理者でコマンドを実行し，HiRDB へアクセスする際の認証には監査人の認可識別子とパスワードを指定してください。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdaudswap 〔-U 認可識別子〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdaudswap -u ユニット識別子 〔-U 認可識別子〕
```

(4)　オプション

(a)　-u ユニット識別子　～＜識別子＞((4 文字))

スワップする現用の監査証跡ファイルがあるユニットの，ユニット識別子を指定します。

(b)　-U 認可識別子

監査人の認可識別子を指定します。このオプションを指定すると，パスワード入力要求の応答付きメッセージが出力されます。

＜規則＞

1. このオプションに指定した認可識別子と，応答付きメッセージに入力したパスワードで HiRDB に接続し，その認可識別子に監査権限があるかをチェックします。
2. このオプションは，応答付きメッセージに応答できる環境で pdaudswap コマンドを実行する場合に指定できます。
3. このオプションを省略した場合，pdaudswap コマンド実行時の環境変数 PDUSER の設定値が仮定されます。このオプションを省略し，かつ環境変数 PDUSER も設定していない場合，pdaudswap

コマンドは実行できません。このオプションを省略する場合は，環境変数 PDUSER を設定しておく必要があります（指定例を次に示します）。

（例）

```
PDUSER=¥""認可識別子"¥"/¥""パスワード"¥"
```

(5) 規則

1. pdaudswap コマンドは，HiRDB 稼働中のときだけ実行できます。ただし，pdstart -r コマンドで HiRDB を開始している場合は実行できません。
2. pdaudswap コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdaudswap コマンド実行中に，pdaudbegin，pdaudend，pdaudrm，及び pdaudswap コマンドを実行すると，コマンドエラーとなります。
4. pdaudswap コマンドは，システム定義に pd_aud_file_name オペランドを指定していない場合，又は pd_aud_file_name オペランドに指定している監査証跡ファイルがない場合，コマンドエラーとなります。
5. スワップ先にできるファイルの世代がない場合，及び監査証跡ファイルが 1 世代も生成されていない場合，コマンドエラーとなります。
6. スワップ先にできるファイルの世代の，監査証跡ファイルのオープンに失敗した場合，自動的にほかの世代へスワップします。
7. システム定義に pd_aud_no_standby_file_opr = down を指定していて，かつ使用できる世代数が残り 1 世代以下の場合，pdaudswap コマンドを実行するとコマンドエラーになります。

(6) 注意事項

1. pdaudswap コマンドの結果は，pdls -d aud コマンド，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了となります。

2.8 pdbkupls（バックアップファイルの情報表示）

2.8.1 pdbkupls の形式と規則

(1) 機能

pdcopy で取得したバックアップの情報（RD エリアの一覧，バックアップ取得日時など）を表示します。

pdbkupls コマンドは，次のような場合に使用します。

- 取得したバックアップの内容を調査したいとき（例えば，以前取得したバックアップに何が入っているかを調べたいなど）
- 取得したバックアップの情報を履歴として残しておきたいとき

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdbkupls -b {バックアップファイル名〔,バックアップファイル名〕…
          |ポリシー名}
      〔-k {u|i|n}〕
      〔-o バックアップ情報出力ファイル名〕 〔-c〕
      〔-a〕 〔-s 経過メッセージ出力間隔〕
      〔-U {バックアップ検索条件開始時刻,バックアップ検索条件終了時刻
            |,バックアップ検索条件終了時刻}〕
```

(4) オプション

(a) -b {バックアップファイル名〔,バックアップファイル名〕...|ポリシー名}

バックアップファイルの名称を指定します。

バックアップファイル名〔,バックアップファイル名〕...の場合：

～<パス名>((1023 バイト以内))

ファイル名，又は HiRDB ファイル名を指定します。

ファイル名の場合は同時に-k u を，HiRDB ファイル名の場合は同時に-k i を指定してください。

ポリシー名の場合：

～＜識別子＞((128 バイト以内))

NetBackup のポリシー名を指定します。このとき，同時に-k n を指定してください。

＜規則＞

1. ホスト名を指定した場合，エラーとなります。
2. 複数個にわたっているバックアップだけ複数同時に指定できます。複数個にわたっていないバックアップを複数同時に指定した場合は無視されます。また，複数個にわたっているバックアップを一つだけ指定した場合，-a オプションを指定しているときはエラー終了します。
3. バックアップは，pdcopy コマンドを実行してコマンドの戻り値が 0 又は 4 で取得したバックアップを指定してください。

(b)　-k　{u｜i｜n}

バックアップファイルの種別を指定します。

u：

バックアップファイルが通常のファイルの場合に指定します。

i：

バックアップファイルが HiRDB ファイルの場合に指定します。

n：

バックアップファイルが NetBackup を使用して取得している場合に指定します。

(c)　-o バックアップ情報出力ファイル名　～＜パス名＞((1023 バイト以内))

バックアップファイルの情報をファイルに出力する場合に指定します。省略した場合，標準出力に出力します。

(d)　-c

CSV 形式で出力する場合に指定します。省略した場合，標準形式で出力します。

(e)　-a

すべての情報を出力する場合に指定します。

省略した場合，HiRDB 識別子，バックアップ取得モード，バックアップ取得時刻，及びログポイント情報ファイルの指定有無を出力します。

(f)　-s 経過メッセージ出力間隔　～＜符号なし整数＞((1～1024))

pdbkupls コマンドの実行中に経過メッセージを出力する場合に指定します。経過メッセージ出力間隔には，バックアップファイルのデータ処理量をギガバイト単位で指定します。例えば，「-s 100」と指定した場合，100 ギガバイト処理するごとに経過メッセージを出力します。

なお，このオプションは，-a オプションを指定しているときだけ有効となります。

(g) -U {バックアップ検索条件開始時刻，バックアップ検索条件終了時刻｜，バックアップ検索条件終了時刻}

同一ポリシー名で取得したバックアップ中の，任意の時点で取得したバックアップを対象とする場合に指定します。このオプションは，NetBackup 連携機能使用時（-k n 指定時）に指定します。それ以外の場合に指定しても無効となります。

バックアップ検索条件開始時刻，及びバックアップ検索条件終了時刻の両方を指定した場合，指定した期間内での最新のバックアップが対象となります。バックアップ検索条件終了時刻だけを指定した場合，終了時刻時点でのバックアップが対象となります。

＜規則＞

- このオプションを省略した場合，同一ポリシー名で取得したバックアップ中の最新のバックアップを使用します。
- バックアップの取得日時は，bpimagelist コマンド（-policy オプション指定）で参照できます。
- バックアップ検索条件開始時刻及びバックアップ検索条件終了時刻は，日付と時刻を_（アンダースコア）で結んで指定します。形式を次に示します。時刻を省略した場合，開始時刻は 000000 が，終了時刻は 235959 が仮定されます。

```
-U YYYYMMDD〔_hhmmss〕, YYYYMMDD〔_hhmmss〕
```

YYYY：年　～＜符号なし整数＞((1990～2037))

西暦で指定します。

MM ：月　～＜符号なし整数＞((01～12))

DD ：日　～＜符号なし整数＞((01～31))

hh ：時　～＜符号なし整数＞((00～23))

mm ：分　～＜符号なし整数＞((00～59))

ss ：秒　～＜符号なし整数＞((00～59))

終了時刻だけを指定する場合は，コンマ（,）に続けて終了時刻を指定します。

(5) 規則

1. pdbkupls コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdbkupls コマンドは，バックアップファイルがあるホストのサーバマシンで実行してください。なお，NetBackup を使用する場合は，pdcopy 実行時のバックアップファイル出力先のホストがあるサーバマシンで実行してください。
3. pdbkupls コマンドは，pdcopy で取得したバックアップファイルだけ処理できます。それ以外のバックアップファイルを指定した場合，エラーとなります。

4. NetBackup を使用する場合，JP1/VERITAS NetBackup Agent for HiRDB License が必要となります。JP1/VERITAS NetBackup Agent for HiRDB License については，マニュアル「JP1/VERITAS NetBackup v4.5 Agent for HiRDB License」を参照してください。
5. 差分バックアップファイルは指定できません。指定した場合はエラーとなります。

(6) 注意事項

1. pdbkupls コマンドのリターンコードは，0 の場合は正常終了，12 の場合は異常終了となります。12 の場合は，その前に出力されているメッセージを参照してエラー要因を取り除いてください。
2. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdbkupls で出力されるファイルには BOM は付きません。

(7) 出力形式

出力形式についての規則を次に示します。

1. (改行)には改行コード（CR+LF）が入ります。
2. 出力情報がない項目については，標準形式の場合は「*」を出力します。CSV 形式の場合は，該当部分の情報は出力しません（コンマが 2 個続きます）。
3. rr...r～uu...u は，RD エリア数分繰り返します。繰り返す場合，出力順は RD エリア ID の昇順となり，(改行)と<<RDAREA information>>も繰り返し出力します。
4. バックアップ情報出力ファイルにデータを出力した場合，最終データの次の行に EOF が入ります。
5. 出力されるデータは，1,024 バイト（改行コードを含まない）ごとに改行されます。
6. CSV 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

出力形式を次に示します。

(a) -a オプションを省略した場合

標準形式：

```
<<System information>>(改行)
<HiRDB system id> : aaaa(改行)
(改行)
<<Backup file information>>(改行)
<Backup mode> : dd...d(改行)
<Backup start Time> : ee...e(改行)
<Logpoint information Y/N> : g(改行)
```

CSV 形式：

```
aaaa,dd...d,ee...e,g(改行)
```

[説明]

aaaa～g については，「-a オプションを指定した場合」の説明を参照してください。

(b)　-a オプションを指定した場合

標準形式：

```
<<System information>>(改行)
<HiRDB system id> : aaaa(改行)
(改行)
<<Backup file information>>(改行)
<Backup file count> : bbbb(改行)
<Backup file name> : cc...c(改行)
<Backup mode> : dd...d(改行)
<Backup start time> : ee...e(改行)
<Backup end time> : ff...f(改行)
<Logpoint information Y/N> : g(改行)
<Get RDAREA's count> : hhhh(改行)
<Get RDAREA's list> : ii...i(改行)
<Errskip Y/N> : j(改行)
<Replica RDAREA generation No> : kk(改行)
<Server name> : ll...l(改行)
<Server RUNID> : mm...m(改行)
<Server group name> : nn...n(改行)
<Server block No> : oo...o(改行)
<Server generation No> : pp(改行)
<Server Log start time> : qq...q(改行)
(改行)
<<RDAREA information>>(改行)
<RDAREA name> : rr...r(改行)
<RDAREA id> : ss...s(改行)
<RDAREA kind> : tt...t(改行)
<RDAREA last LSN> : uu...u(改行)
<RDAREA last in-memory db-sync time> : vv...v(改行)
        :
```

CSV 形式：

```
aaaa,bbbb,cc...c,dd...d,ee...e,ff...f,g,hhhh,ii...i,j,kk,ll...l,mm...m,nn...n,
oo...o,pp,qq...q,rr...r,ss...s,tt...t,uu...u …
```

[説明]

aaaa：

バックアップを取得した HiRDB の HiRDB 識別子です（4 文字以内）。

bbbb：

-b オプションに指定したバックアップファイルの個数です（10 進数の 1～4 けた）。

cc...c：

-b オプションに指定したバックアップファイル名です（1,023 文字以内）。pdcopy の-b オプションに指定した順番で出力します。複数ある場合は，コンマで区切って出力します。

dd...d：

バックアップ取得時の-M オプションの指定値です（10 文字以内）。

-M x 又は-M r の場合は EXCLUSIVE，-M s の場合は SHARED となります。

ee...e：

バックアップ取得時に最初のレコードを書き込んだ時間です（YYYY-MM-DD hh:mm:ss）。

インメモリ RD エリアがバックアップ対象に含まれている場合，インメモリデータバッファとインメモリ RD エリアの同期を取った時刻（一度も同期を取っていない場合はインメモリ化した時刻）を出力します。インメモリ RD エリアが複数ある場合は，その中で一番古い時刻が出力されます。

ff...f：

バックアップ取得時に最後のレコードを書き込んだ時間です（YYYY-MM-DD hh:mm:ss）。

g：

バックアップ取得時にログポイント情報ファイルを作成していたかどうかの情報です（1 文字）。

ログポイント情報ファイルを作成していた場合は Y，作成していない場合は N となります。

hhhh：

バックアップファイル中に格納されいてる RD エリアの個数です（10 進数 1～4 けた）。

ii...i：

バックアップファイル中に格納されいてる RD エリアの名称です（30 文字以内）。複数ある場合はコンマで区切ります。また，バックアップ取得時の RD エリア ID の昇順で出力します。

j：

バックアップ取得時に-J オプションを指定していたかどうかの情報です（1 文字）。

-J オプションを指定していた場合は Y，指定していない場合は N となります。

kk：

常に 0 を表示します。

ll...l：

RD エリアがあるサーバの名称です（8 文字以内）。

mm...m：

ログサーバラン ID（システムが使用する情報）です（16 進数 8 けた）。

nn...n：

ロググループ名（システムが使用する情報）です（8 文字以内）。

oo...o：

ブロック番号です（16 進数 8 けた）。

pp：

世代番号です（10 進数 1～2 けた）。

qq...q：

システムログファイルの使用開始時刻です（YYYY-MM-DD hh:mm:ss）。

rr...r：

RD エリア名です（30 文字以内）。

ss...s：

RD エリア ID です（10 進数 1～8 けた）。

tt...t：

RD エリアの種別です（15 文字以内）。

MASTERDIRECTORY：マスタディレクトリ用 RD エリア

DATADIRECTORY：データディレクトリ用 RD エリア

DATADICTIONARY：データディクショナリ用 RD エリア

SYSTEM_LOB：データディクショナリ LOB 用 RD エリア

USER：ユーザ用 RD エリア

USER_LOB：ユーザ LOB 用 RD エリア

REG：レジストリ用 RD エリア

REG_LOB：レジストリ LOB 用 RD エリア

uu...u：

RD エリアの更新通番（システムが使用する情報）です（17 文字）。

vv...v：

インメモリ RD エリアがバックアップ対象に含まれている場合，インメモリデータバッファとインメモリ RD エリアの同期を取った時刻（一度も同期を取っていない場合はインメモリ化した時刻）を，YYYY-MM-DD hh:mm:ss の形式で出力します。

インメモリ RD エリアがバックアップ対象に含まれていない場合は，'*'が出力されます。

なお，この項目は標準形式のときだけ出力されます。

2.9 pdbufls（グローバルバッファ情報の表示）

2.9.1 pdbufls の形式と規則

(1) 機能

グローバルバッファの状態を表示します。

インメモリ RD エリアがある場合は，インメモリデータバッファの状態も表示できます。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdbufls 〔-k 出力種別〕 〔-d〕 〔-x 〔-y〕〕〔-M〕〔-N〕
        〔 {-s サーバ名〔, サーバ名〕…
           |-a グローバルバッファ名〔, グローバルバッファ名〕…} 〕
        〔-W 実行監視時間〕
```

(4) オプション

(a) -k 出力種別 ～≪sts≫

表示する情報の種別を指定します。

def：

グローバルバッファの定義情報を表示します。

sts：

グローバルバッファの統計情報を表示します。表示する統計情報は，前回の pdbufls コマンド実行時点からの累計です。最初に pdbufls コマンドを実行する場合，又は-d オプションを指定した場合は，HiRDB 開始時点からの累計を表示します。

all：

グローバルバッファの定義情報及び統計情報の両方を表示します。

(b) -s サーバ名 ～＜識別子＞((1～8))

サーバ内にあるすべてのグローバルバッファの情報を表示したい場合，そのサーバ名を指定します。指定できるサーバ名の数は最大 128 個です。129 個以上指定すると，129 個目以降のサーバは無視されます。

(c)　-a グローバルバッファ名　～＜識別子＞((1～16))

表示するグローバルバッファの名称を指定します。グローバルバッファ名を重複指定した場合は，重複排除して処理されます。指定できるグローバルバッファ名の数は最大 128 個です。129 個以上指定すると，129 個目以降のグローバルバッファは無視されます。

このオプションにインメモリデータバッファ名を指定すると，インメモリデータバッファの情報が表示されます。

(d)　-d

HiRDB 開始時点からのグローバルバッファの統計情報を表示したい場合に指定します。なお，-k def を指定した場合にこのオプションを指定しても無視されます。

(e)　-x

DAT 形式で情報を表示する場合に指定します。-k all の場合，このオプションは指定できません。

(f)　-y

-x オプション指定時に，ヘッダを付ける場合に指定します。

(g)　-M

インメモリデータバッファの情報を表示する場合に指定します。このオプションの指定例を次に示します。

＜指定例 1 ＞

pdbufls -k def -M

この場合，グローバルバッファとインメモリデータバッファの両方の定義情報が表示されます。

＜指定例 2 ＞

pdbufls -k sts -M -a インメモリデータバッファ名

この場合，-a オプションに指定したインメモリデータバッファの統計情報が表示されます。

インメモリデータバッファの情報もグローバルバッファと同じように表示されます。そのため，このオプションを指定してインメモリデータバッファの情報を表示した場合は，pdbufls コマンドの説明文中の「グローバルバッファ」を「インメモリデータバッファ」に読み替えてください。

(h)　-N

OTHER 用グローバルバッファに割り当てている RD エリアの名称を表示する場合に指定します。-k オプションに sts を指定した場合，又は省略した場合はこのオプションは無視されます。

(i)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdbufls コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

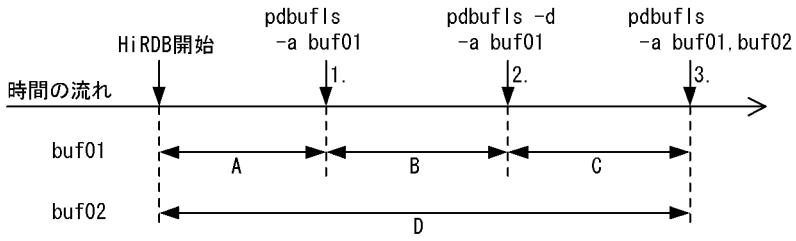
なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5) 規則

1. pdbufls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdbufls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. -s オプション及び-a オプションの両方を省略した場合は，すべてのサーバのグローバルバッファの情報を表示します。
4. グローバルバッファの統計情報の取得範囲を次の図に示します。

図 2–1 グローバルバッファの統計情報の取得範囲



[説明]

1. 表示される情報は，HiRDB 開始から 1 までの A の期間となります。このとき，前回統計情報を取得した時刻は表示されません。
2. 表示される情報は，-d オプションを指定しているので，HiRDB 開始から 2 までの A ＋ B の期間となります。このとき，前回統計情報を取得した時刻は表示されません。また，この時点のグローバルバッファの情報は，次回の pdbufls コマンド実行には反映されません。
3. buf01 で表示される情報は，1 から 3 までの B ＋ C の期間となります。buf02 で表示される情報は，HiRDB 開始から 3 までの D の期間となります。このとき，buf01 には 1 で統計情報を取得した時刻が表示されます。また，buf02 には，前回統計情報を取得した時刻は表示されません。

(6) 注意事項

1. pdbufls コマンドのリターンコードを次に示します。

   0：正常終了

   4：警告終了（一部のサーバの処理がエラー終了）

   8：異常終了

   12：異常終了（エラーメッセージを表示できない事象が発生）

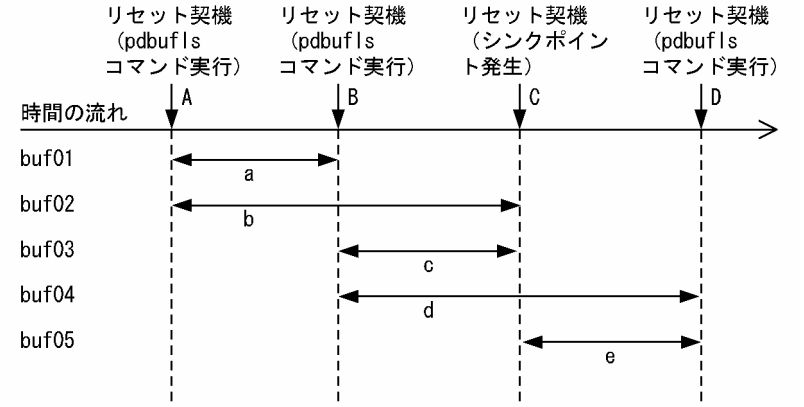
   リターンコードが 12 の場合は，シングルサーバ又はディクショナリサーバがあるホストのイベントログのエラーメッセージを参照し，エラーの要因を取り除いた後，再度 pdbufls コマンドを実行してください。イベントログにエラーメッセージが出力されていない場合は，保守員に連絡してください。

2. 参照 GET 回数と更新 GET 回数の合計が 4,294,967,295 回を超えるグローバルバッファが検出されると，該当するグローバルバッファの統計情報はリセットされます。リセット契機を次に示します。

   - pdbufls -k sts，又は pdbufls -k all を実行したとき（JP1/PFM-Agent for HiRDB が，内部的に pdbufls -k sts，又は pdbufls -k all を実行したときを含む）
   - pdstbegin -k buf で統計ログを取得中にシンクポイントが発生したとき

   リセット契機と，リセット時に得られる統計情報について，次の図に示します。

   図 2–2　グローバルバッファの統計情報のリセット契機と，リセット時に得られる統計情報



[説明]

- A 時点でリセットされた場合

  A 時点で表示される情報のうち，次の項目以外は 0 となります。

  ・カレントの参照バッファ数

  ・カレントの更新バッファ数

  ・デファードライトトリガ時の出力契機となる更新バッファ数

  B 時点で表示される情報は，a の期間となります。

  C 時点で表示される情報は，C 時点で出力される統計ログのうち，b の期間となります。

- B 時点でリセットされた場合

  B 時点で表示される情報のうち，次の項目以外は 0 となります。

  ・カレントの参照バッファ数

  ・カレントの更新バッファ数

  ・デファードライトトリガ時の出力契機となる更新バッファ数

  C 時点で表示される情報は，C 時点で出力される統計ログのうち，c の期間となります。

  D 時点で表示される情報は，d の期間となります。

- C 時点でリセットされた場合

  C 時点で出力される統計ログのうち，次の項目以外は 0 となります。

  ・バッファ面数

  ・カレントの参照バッファ数

  ・カレントの更新バッファ数

・デファードライトトリガ時の出力契機となる更新バッファ数
・バッファ管理領域の排他処理での，排他競合待ちとなった回数
・バッファの排他処理での，排他競合待ち発生率
・スリープ処理実行での，バッファの排他処理のスリープ処理実行回数の平均値
・スピン処理実行での，バッファの排他処理発生率
・バッファの排他処理中の，スピンループの実行回数の平均値
・シンクポイント時のバッファプール排他占有時間
・シンクポイント時のバッファプール排他占有時間内の処理バッファ数

D 時点で表示される情報は，e の期間となります。

(7) 出力形式

(a) -k def を指定した場合

```
DEFINE OF GLOBAL BUFFER
  EDIT TIME aa...a
  BUFNAME SVID    TYPE   SIZE  NUM    WRATIO RDAREA/INDEX NAME
                  PRMAX  PRNUM CSIZE  MAPS
  bb...b  cc...c  d      eeK   ff...f ggg    hh...h
                  kk...k lll   mm...m nn
      [NOT FOUND ii...i=jj...j]
```

[説明]

aa...a：

pdbufls コマンドを実行した時刻

年月日　時：分：秒の形式

bb...b：

グローバルバッファ名称（16 文字以内）

cc...c：

サーバ名（8 文字以内）

d：

グローバルバッファ種別

R：RD エリア用

I：インデクス用

B：LOB 用

O：OTHER 用

OTHER 用のグローバルバッファとは，システム共通定義の pdbuffer オペランドで-o オプションを指定したグローバルバッファのことです。

M：インメモリデータバッファ

ee：

バッファ長（10 進数 2 けた，単位はキロバイト）。

ff...f：

バッファ面数（32 ビットモードの場合 10 進数 6 けた以内，64 ビットモードの場合 10 進数 10 けた以内）。

ggg：

デファードライトトリガ時の更新ページ出力比率（10 進数 3 けた，単位は%）。システム共通定義の pdbuffer オペランドの-w オプションの指定値。

hh...h：

RD エリア名又はインデクス名（65 文字以内）。

同一グローバルバッファに複数 RD エリアが割り当てられている場合，割り当てられている 1RD エリアを 1 行とし，複数行で表示されます。既に削除されている RD エリアは表示されません。

割り当てられていたインデクス又はすべての RD エリアを削除している場合，この項目は表示されません。

また，OTHER 用のグローバルバッファでは，次のどちらかに該当する場合，この項目は表示されません。

- -N オプションを指定していない
- OTHER 用グローバルバッファに割り当てられた RD エリアが存在しない

出力形式は次のようになります。

RD エリア名の場合："RD エリア名"

インデクス名の場合："認可識別子"."インデクス識別子"

ii...i：

-s オプションを指定した場合，「SERVER」と表示されます。-a オプションを指定した場合，「BUFFER POOL」と表示されます。

jj...j：

-s オプションを指定した場合，サーバ名（8 文字以内）が表示されます。-a オプションを指定した場合，グローバルバッファ名（16 文字以内）が表示されます。

kk...k：

同時実行最大プリフェッチ数（10 進数 5 けた）。システム共通定義の pdbuffer -m オプションの指定値です。

lll　：

一括入力最大ページ数（10 進数 3 けた）。システム共通定義の pdbuffer -p オプションの指定値です。

mm...m：

システムが使用する内部情報。「******」固定です。

nn　：

システムが使用する内部情報。「**」固定です。

(b)　-k sts を指定した場合

```
STATISTICS OF GLOBAL BUFFERRC
EDIT TIME aa...a
BUFFNAME SVID   HIT(REF,UPD)  RFGET  READ   RFFLS  REFBUF          WAITL
         LAST_EXEC_TIME       UPGET  WRITE  UPFLS  UPBUF(TRG)      SYNC
                              PRRED  PRHIT  PRINS  PRREQ           INSB
                              LRREQ  LWREQ  LRPAG  LWPAG
                              CINSM  CFMAX  CFAVG
bb...b   cc...c eee(fff,ggg)  hh...h jj...j ll...l nn...n          qq...q
         dd...d               ii...i kk...k mm...m oo...o(pp...p) rr...r
                              uu...u vvv    ww...w xx...x          yy...y
                              zz...z AA...A BB...B CC...C
                              DD...D EE...E FF...F
       [NOT FOUND ss...s=tt...t]
```

[説明]

aa...a：

pdbufls コマンドを実行した時刻

年月日　時：分：秒の形式

bb...b：

グローバルバッファ名（16 文字以内）

cc...c：

サーバ名（8 文字以内）

dd...d：

個々のグローバルバッファに対する，前回統計情報を取得した時刻

年月日　時：分：秒の形式

前回統計情報が取得されていない場合，又は-d オプションを指定している場合には，「****-**-** **:**:**」が表示されます。

eee：

グローバルバッファのヒット率（10 進数 3 けた，単位は%）

fff：

参照要求のヒット率（10 進数 3 けた，単位は%）

ggg：

更新要求のヒット率（10 進数 3 けた，単位は%）

hh...h：

参照 GET 回数※

ii...i：

更新 GET 回数※

jj...j：

実 READ 回数※

kk...k：

実 WRITE 回数※

ll...l：

参照バッファフラッシュ回数※

mm...m：

更新バッファフラッシュ回数※

nn...n：

カレントの参照バッファ数（32 ビットモードの場合 10 進数 6 けた以内，64 ビットモードの場合 10 進数 10 けた以内）

oo...o：

カレントの更新バッファ数（32 ビットモードの場合 10 進数 6 けた以内，64 ビットモードの場合 10 進数 10 けた以内）

pp...p：

デファードライトトリガ時の出力契機となる更新バッファ数（32 ビットモードの場合 10 進数 6 けた以内，64 ビットモードの場合 10 進数 10 けた以内）

qq...q：

バッファ排他待ち発生回数※

rr...r：

DB シンクポイント発生回数※

ss...s：

-s オプションを指定した場合，「SERVER」と表示されます。-a オプションを指定した場合，「BUFFER POOL」と表示されます。

tt...t：

-s オプションを指定した場合，サーバ名（8 文字以内）が表示されます。-a オプションを指定した場合，グローバルバッファ名（16 文字以内）が表示されます。

uu...u：

プリフェッチ入力ページ数※

**vvv　**：

プリフェッチヒット率（10 進数 3 けた）

ww...w：

プリフェッチバッファ不足発生回数※

xx...x：

プリフェッチ READ 要求回数※

yy...y：

バッファ不足発生回数※

zz...z：

LOB バッファ READ 要求回数※

AA...A：

LOB バッファ WRITE 要求回数※

BB...B：

LOB バッファ一括入力ページ数※

CC...C：

LOB バッファ一括出力ページ数※

DD...D：

システムが使用する内部情報

EE...E：

システムが使用する内部情報

FF...F：

システムが使用する内部情報

注※　表示形式を次に示します。

数値の範囲	表示形式
0～999	0～999
1,000～9,994	1.00K～9.99K※
9,995～99,949	10.0K～99.9K※
99,950～999,499	100K～999K※
999,500～9,994,999	1.00M～9.99M※
9,995,000～99,949,999	10.0M～99.9M※
99,950,000～999,499,999	100M～999M※
999,500,000～4,294,967,295	1.00G～4.29G※

（凡例）

K：キロ

M：メガ

G：ギガ

注※

表示けた数の次の位が，四捨五入されます。

(c)　DAT 形式での出力形式（-x 及び-y オプション指定時）

- **-k def**

```
”DEFINE_OF_GLOBAL_BUFFER”[CR]
”EDIT_TIME”[CR]
”aa...a”[CR]
”BUFFNAME”,”SVID”,”TYPE”,”SIZE”,”NUM”,”WRATIO”,”RDAREA/INDEX_NAME”,”PRMAX”,
”PRNUM”,”CSIZE”,”MAPS”[CR]
”bb...b”,”cc...c”,”d”,ee,ff...f,ggg,”hh...h”,kk...k,lll,mm...m,nn[CR]
```

- **-k sts**

```
”STATISTICS_OF_GLOBAL_BUFFER”[CR]
”EDIT_TIME”[CR]
”aa...a”[CR]
”BUFFNAME”,”SVID”,”LAST_EXEC_TIME”,”HIT”,”REFHIT”,”UPDHIT”,”RFGET”,”UPGET”,
”READ”,”WRITE”,”RFFLS”,”UPFLS”,”REFBUF”,”UPBUF”,”TRG”,”WAITL”,”SYNC”,
”PRRED”,”PRHIT”,”PRINS”,”PRREQ”,”INSB”,”LRREQ”,”LWREQ”,”LRPAG”,”LWPAG”,
”CINSM”,”CFMAX”,”CFAVG”,”RFGETR”,”UPGETR”,”RFHITR”,”UPHITR”,”READR”,”WRITER”,
”RFFLSR”,”UPFLSR”,”WAITLR”,”SYNCR”,”PRREDR”,”PRHITR”,”PRINSR”,”PRREQR”,”INSBR”,
”LRREQR”,”LWREQR”,”LRPAGR”,”LWPAGR”[CR]
”bb...b”,”cc...c”,”dd...d”,eee,fff,ggg,”hh...h”,”ii...i”,
”jj...j”,”kk...k”,”ll...l”,”mm...m”,nn...n,oo...o,pp...p,”qq...q”,”rr...r”,
”uu...u”,vvv,”ww...w”,”xx...x”,”yy...y”,”zz...z”,”AA...A”,”BB...B”,”CC...C”,
”DD...D”,”EE...E”,”FF...F”,”GG...G”,”HH...H”,”II...I”,”JJ...J”,”KK...K”,”LL...L”,
”MM...M”,”NN...N”,”OO...O”,”PP...P”,”QQ...Q”,”RR...R”,”SS...S”,”TT...T”,”UU...U”,
”VV...V”,”WW...W”,”XX...X”,”YY...Y”[CR]
```

注

[CR]は，改行を表します。

[説明]

aa...a から **FF...F** については，「出力形式」で説明した内容を参照してください。なお，次に説明する項目は，DAT 形式表示の場合だけ表示されます。

GG...G：

参照 GET 回数の実データ（10 進数 10 けた以内）

HH...H：

更新 GET 回数の実データ（10 進数 10 けた以内）

II...I：

参照要求ヒット回数の実データ（10 進数 10 けた以内）

JJ...J：

更新要求ヒット回数の実データ（10 進数 10 けた以内）

KK...K：

実 READ 回数の実データ（10 進数 10 けた以内）

LL...L：

実 WRITE 回数の実データ（10 進数 10 けた以内）

MM...M：

参照バッファフラッシュ回数の実データ（10 進数 10 けた以内）

NN...N：

更新バッファフラッシュ回数の実データ（10 進数 10 けた以内）

OO...O：

バッファ排他待ち発生回数の実データ（10 進数 10 けた以内）

PP...P：

DB シンクポイント発生回数の実データ（10 進数 10 けた以内）

QQ...Q：

プリフェッチ入力ページ数の実データ（10 進数 10 けた以内）

RR...R：

プリフェッチヒット回数の実データ（10 進数 10 けた以内）

SS...S：

プリフェッチバッファ不足発生回数の実データ（10 進数 10 けた以内）

TT...T：

プリフェッチ READ 要求回数の実データ（10 進数 10 けた以内）

UU...U：

バッファ不足発生回数の実データ（10 進数 10 けた以内）

VV...V：

LOB バッファ READ 要求回数の実データ（10 進数 10 けた以内）

WW...W：

LOB バッファ WRITE 要求回数の実データ（10 進数 10 けた以内）

XX...X：

LOB バッファ一括入力ページ数の実データ（10 進数 10 けた以内）

YY...Y：

LOB バッファ一括出力ページ数の実データ（10 進数 10 けた以内）

［DAT 形式での出力規則］

1. DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

2. 1 行に 1 グローバルバッファの情報を表示します。ただし，同一グローバルバッファに複数 RD エリアが割り当てられている場合，グローバルバッファの定義情報は，割り当てられている 1RD エリアを 1 行で表示します。このとき，RD エリア名以外の項目はすべて繰り返して表示します。
3. グローバルバッファの統計情報の，単位を含めた表示は文字要素として扱います。
4. pdbufls コマンドの実行時刻は，タイトル行の一部となります。そのため，タイトル行の出力指定（-y オプション）がある場合にだけ表示します。
5. 次の項目の情報が取得できない場合は，空要素として表示します。
   ・グローバルバッファの定義情報のバッファ長，及びバッファ面数
   ・グローバルバッファの統計情報の前回情報を取得した時刻
   また，システムが使用する内部情報は，空又は「""」となります。
6. DAT 形式で表示する場合，次のエラーメッセージは出力しません。
   [NOT FOUND ss...s=tt...t]

2.10　pdbufmod（グローバルバッファの動的変更）

2.10.1　pdbufmod の形式と規則

(1)　機能

HiRDB の稼働中に，グローバルバッファの追加，削除，及び変更をします。

pdbufmod の変更は，ユニット又はサーバの正常終了後は引き継がれません。したがって，次回の HiRDB の正常開始までに，システム定義の pdbuffer オペランドに pdbufmod での変更を反映しておく必要があります。なお，pdchgconf コマンドを使用すると，HiRDB を終了しなくてもシステム定義を変更できます。

pdbufmod コマンドは，HiRDB Advanced High Availability を組み込んでいる場合に使用できます。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdbufmod -k 変更種別 -a グローバルバッファ名
        〔{-r RDエリア名〔, RDエリア名〕…|-b RDエリア名〔, RDエリア名〕…
          |-o|-i 認可識別子.インデクス識別子}〕
        〔-n バッファ面数 〔-l バッファサイズ〕〕
        〔-m 同時実行最大プリフェッチ数〕 〔-p 一括入力最大ページ数〕
        〔-w デファードライトトリガ時の更新ページ出力比率〕
        〔-y デファードライトトリガ契機の更新バッファ面数〕
        〔-W 実行監視時間〕
```

(4)　オプション

(a)　-k 変更種別

変更種別を指定します。

add：

グローバルバッファを新規に追加する場合，又は既存のグローバルバッファに新たにグローバルバッファを追加する場合に指定します。

新規にグローバルバッファを追加する場合，-n オプションを必ず指定してください。また，新規にグローバルバッファを追加する場合に-r，-b，又は-i オプションを指定するときは，指定する RD エリア，又はインデクスは先に定義されていなければなりません。

既存の共用メモリセグメントに空きがない場合，新規のグローバルバッファは，新規の共用メモリセグメントに作成されます。

del：

グローバルバッファを削除する場合，又はグローバルバッファの特定の RD エリアを削除したい場合に指定します。

特定の RD エリアを削除する場合は，-r 又は-b オプションでその RD エリアを指定します。

削除するグローバルバッファに割り当てられている RD エリア，及びグローバルバッファから削除する RD エリアは，事前に閉塞クローズ状態にしておく必要があります。

インメモリ RD エリアに割り当てているグローバルバッファは削除できません。このグローバルバッファを削除する場合は，インメモリ化を解除した後にグローバルバッファを削除してください。

upd：

グローバルバッファの定義を変更する場合に指定します。グローバルバッファの定義とは，バッファ面数，バッファサイズ，同時実行最大プリフェッチ数，一括入力最大ページ数，デファードライトトリガ時の更新ページ出力比率，及びデファードライトトリガ契機の更新バッファ面数を指します。

グローバルバッファの定義を変更した後は，グローバルバッファのキャッシュ内容は破棄されます。

グローバルバッファの定義を変更して，グローバルバッファのメモリサイズが増加した場合，既存の共用メモリセグメントに空きがないと，定義を変更したグローバルバッファは新規の共用メモリセグメントに作成されます。

変更種別とそのほかのオプションとの関係を次に示します。

そのほかのオプション	-k 変更種別		
	add	del	upd
-a	◎	◎	◎
-r	○	○	×
-b	○	○	×
-o	○※1	×	×
-i	○※1	×	×
-n	○※1※2	×	○
-l	○※1	×	○
-m	○※1	×	○
-p	○※1	×	○
-w	○※1	×	○

そのほかのオプション	-k 変更種別		
	add	del	upd
-y	○※1	×	○

（凡例）

◎：必ず指定します。

○：任意に指定できます。

×：指定できません。

注※1

新規にグローバルバッファを追加する場合に指定できます。

注※2

新規にグローバルバッファを追加する場合，必ず指定してください。

(b)　-a グローバルバッファ名　～＜識別子＞((1～16))

追加，削除，又は変更をするグローバルバッファの名称を指定します。

-k add（既存のグローバルバッファへの RD エリアの追加），-k del，及び-k upd の場合は，既に定義されているグローバルバッファの名称を指定してください。-k add（新規のグローバルバッファの追加）の場合は，サーバ内で一意となるグローバルバッファの名称を指定してください。

このオプションにはインメモリデータバッファを指定できません。

(c)　-r RD エリア名〔，RD エリア名〕...　～＜識別子＞((1～30))

グローバルバッファに割り当てる RD エリア，又はグローバルバッファから削除する RD エリアの名称を指定します。

＜規則＞

1. RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。
2. 同時に-n オプションでバッファ面数を指定して，一つのグローバルバッファに複数の RD エリアを指定した場合，RD エリアのページサイズの中で最も大きいページサイズがバッファ長となります。そのため，ページサイズが近い RD エリアを同じグローバルバッファに割り当てることをお勧めします。ただし，同じページサイズでも用途の異なる表がある RD エリア，ランダムアクセスが多い RD エリア，及び順次アクセスが多い RD エリアは，別グローバルバッファとして割り当てた方がバッファリングの効果が向上します。
3. ユーザ用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   ・オープン状態，かつ閉塞解除状態
   ・オープン状態，かつコマンド閉塞状態

(d)　-b RD エリア名〔，RD エリア名〕...　～＜識別子＞((1～30))

グローバルバッファに割り当てる LOB 用 RD エリア，又はグローバルバッファから削除する LOB 用 RD エリアの名称を指定します。

＜効果的な指定方法＞

次に示すような LOB 用 RD エリアを指定します。

- プラグインインデクスを格納する LOB 用 RD エリア
- データ量が少なく，アクセス頻度が高いデータを格納する LOB 用 RD エリア

なお，一つの LOB 用グローバルバッファには，一つの LOB 用 RD エリアを割り当てることをお勧めします。

＜規則＞

1. RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。
2. 指定できる RD エリアは，データディクショナリ LOB 用 RD エリア，ユーザ LOB 用 RD エリア，及びレジストリ LOB 用 RD エリアです。これ以外の RD エリアを指定した場合はエラーとなります。
3. -b オプションを指定した場合，-m，-p，及び-w オプションは指定できません。
4. -o オプション指定のグローバルバッファを割り当てていない場合，-r に指定している LOB 用 RD エリアは，-b にも指定する必要があります。
5. ユーザ LOB 用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   ・オープン状態，かつ閉塞解除状態
   ・オープン状態，かつコマンド閉塞状態

(e)　-o

-r オプションで指定していないすべての RD エリアに，グローバルバッファを割り当てる場合に指定します。-o オプションは１回だけ指定できます。２回以上指定した場合は最初の指定を有効とします。

(f)　-i 認可識別子.インデクス識別子

認可識別子：～＜識別子＞((1～30))

インデクス識別子：～＜識別子＞((1～30))

インデクス用グローバルバッファを割り当てるインデクスの名称（認可識別子.インデクス識別子）を指定します。

＜効果的な指定方法＞

使用頻度の高いインデクスを指定してください。使用頻度の高いインデクスにグローバルバッファを割り当てると，インデクスページのメモリ常駐度が上がるため，入出力回数が削減できます。

特に，クラスタキー又はユニークキーに定義したインデクスをグローバルバッファに割り当てると効果が大きいです。なお，クラスタキーのインデクス識別子は HiRDB が決めるため，表を定義した後にディクショナリ表（SQL_INDEXES 表の INDEX_NAME 列）を検索してインデクス識別子を確認してください。

また，すべてのインデクスをインデクス用のグローバルバッファに割り当てると，グローバルバッファ全体の使用効率が悪くなります。したがって，メモリ量に応じてグローバルバッファに割り当てるインデクスを厳選してください。

＜規則＞

1. 認可識別子及びインデクス識別子を「¥""」と「"¥"」で囲んだ場合は，英字の大文字と小文字が区別されますが，「¥""」と「"¥"」で囲まない場合はすべて大文字として扱われます。また，インデクス識別子に空白を含む場合は，「¥""」と「"¥"」で囲んでください。
2. 認可識別子.インデクス識別子を指定する場合，データディクショナリ用 RD エリアは次のどちらかの状態でなければなりません。
   ・オープン状態，かつ閉塞解除状態
   ・オープン状態，かつコマンド閉塞状態

(g) -n バッファ面数

32 ビットモードの場合：～＜符号なし整数＞((4～460000))

64 ビットモードの場合：～＜符号なし整数＞((4～1073741824))

グローバルバッファの面数を指定します。

＜指定値の目安＞

- 必要以上に大きな値を指定すると，入出力回数は減るがバッファを検索するためのオーバヘッドが増えます。
- グローバルバッファは共用メモリに確保されます。そのため，必要以上に大きくするとほかのメモリの使用時にページングが多発し，性能が悪くなります。
- シンクポイント取得のタイミング，及びグローバルバッファ内の更新ページの比率によっては，データベースへの出力が集中します。そのため，入出力回数のバランスも考慮する必要があります。
- バッファ面数が大き過ぎると，共用メモリを確保できないことがあるため，注意してください。
- グローバルバッファの面数は，次の表から求めてください。

条件		グローバルバッファ面数の求め方
-r 又は-o オプション指定のグローバルバッファの場合	HiRDB/シングルサーバの場合	同時に発生する SQL 処理要求の数×1SQL で使用するページ数（3～6 程度）
	HiRDB/パラレルサーバの場合	同時実行ユーザ数×1 トランザクション内の平均同時アクセス表数×3×n※1
-b オプション指定のグローバルバッファの場合		LOB 用 RD エリアの総ページ数×常駐度※2(%)
-i オプション指定のグローバルバッファの場合		インデクスの格納ページ数×常駐度※3(%)

注※1

n=1 として見積もったバッファ面数では，バッファのヒット率を高くできません。このため，係数としてどれくらいの余裕をバッファに持たせるか（バッファのヒット率をどのくらいにするか）を検討して指定してください。

注※2

常駐度は 1(100%)が望ましいが，メモリ容量，データのアクセス頻度などを考慮して設定します。ユーザ LOB 用 RD エリアの総ページ数，及びレジストリ用 RD エリアの総ページ数については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

注※3

常駐度は 1（100%）が望ましいが，メモリ容量，インデクスの重要度などを考慮して設定します。インデクスの格納ページ数については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

- バッファのヒット率が HiRDB 稼働業務で 80%以上となるように設定してください。バッファのヒット率は次に示す方法で確認できます。

  ・統計解析ユティリティ（グローバルバッファに関する統計情報）の「更新バッファヒット率」及び「参照バッファヒット率」で確認できます。

  ・pdbufls コマンドのヘッダ名 HIT で確認できます。

(h)　- l バッファサイズ　～<符号なし整数>((4～30 の偶数))

グローバルバッファのバッファサイズをキロバイト単位で指定します。このオプションは-n オプションと一緒に指定します。

<指定値の目安>

通常，このオプションは省略してください。このオプションを省略すると，このグローバルバッファに割り当てた RD エリアの最大ページサイズが，バッファサイズとして仮定されます。ただし，次に示す場合は指定値の変更を考えてください。

- 将来，このオプションで指定した値を超えるページ長の RD エリアを追加したり，再初期化して RD エリアのページ長を大きくしたりすることがある場合，このオプションの指定値を大きめに設定しておくことをお勧めします。ただし，HiRDB を終了できる場合は，HiRDB の次回開始時にはバッファサイズに最大ページ長が仮定されるため，このオプションの指定値を大きめに設定しておく必要はありません。

<規則>

1. このオプションの指定値が RD エリアの最大ページサイズより小さい場合，RD エリアの最大ページサイズがバッファサイズとなります。
2. このオプションに奇数を指定した場合，バッファサイズは指定した値に 1 を加算した値となります。

(i)　-m 同時実行最大プリフェッチ数　～<符号なし整数>((0～95000))《0》

同時にプリフェッチ機能を使用する最大数を指定します。プリフェッチ機能は，rawI/O 機能を使用して大量検索をする場合に入出力時間を削減する効果があります。

このオプションに 0 を指定するか，又は省略した場合は，プリフェッチ機能は動作しません。プリフェッチ機能を動作させるには，1 以上を指定してください。

＜指定値の目安＞

指定値の設定は，このグローバルバッファに割り当てられた RD エリアの表の中で，プリフェッチ機能が適用される SQL を同時に実行する回数を指定します。プリフェッチ機能が適用される SQL を次に示します。ただし，3.に関しては実行数を 2 として算出します。

1. インデクスを使用しない SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）
2. インデクスを使用した昇順検索※をする SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）
3. クラスタキーを使用した昇順検索※をする SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）

注※　複数列インデクスの場合はインデクス定義で指定した順序

＜規則＞

プリフェッチ機能を使用する場合，グローバルバッファ用の共用メモリからグローバルバッファとは別に一括入力専用のバッファが確保されるため，再度共用メモリを見積もる必要があります。

プリフェッチ機能，及びグローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(j)　-p 一括入力最大ページ数　～＜符号なし整数＞((2～256))《32》

プリフェッチ機能の一括入力最大ページ数を指定します。この指定は，-m オプションで 1 以上を指定したときだけ有効になります。

＜指定値の目安＞

共用メモリ量と削減効果のコストパフォーマンスから，指定の目安として次に示す計算式を満足するように指定します。

```
a×b＝64～128　　（単位：キロバイト）

a：プリフェッチ対象表のデータ，又はインデクスが格納されているRDエリアのページ長
b：一括入力最大ページ数
```

(k)　-w デファードライトトリガ時の更新ページ出力比率

～＜符号なし整数＞((0～100))《20》

デファードライトトリガでの更新ページ出力比率をパーセントで指定します。デファードライトトリガについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

このオペランドに 0 を指定した場合は，デファードライトトリガでの更新ページ出力をしません。

＜指定値の目安＞

- グローバルバッファごとの入出力回数，及び更新ページのヒット率などを統計解析ユティリティで確認し，指定値を決めてください。更新ページのヒット率が高いグローバルバッファでは出力比率を低く，ヒット率の低いグローバルバッファでは出力比率を高く指定してください。

- 必要以上に大きい値を指定すると，更新の頻度が多い場合，入出力が多発します。また，遅延書き込み時にデータベースに書き込むページ数が増え，スループットが低下します。反対に，少な過ぎる値を指定すると，シンクポイントダンプ出力時にデータベースに書き出すページ数が多くなることがあります。このため，できるだけ，シンクポイントダンプ出力間隔の間に同じページがデータベースに書き出される回数が少なくなるようにしてください。
- すべてのページがグローバルバッファ上にあり，複数トランザクションで同一ページの更新が多く発生する場合は 0 を指定してください。0 を指定すると，シンクポイント間隔内でデータベースに書き出す総ページ数を減らすことができます。

(l)　-y デファードライトトリガ契機の更新バッファ面数

～＜符号なし整数＞((2～2147483647))

デファードライト処理を実行するときの，更新バッファ面数を指定します。更新バッファ面数が指定値に達すると，デファードライトトリガによってディスクに書き込まれます。

なお，-y オペランドの指定を省略した場合，デファードライトトリガ契機の更新バッファ面数は，pd_dbbuff_rate_updpage オペランド，pdbuffer -y オペランドの組み合わせによって次のようになります。

pd_dbbuff_rate_updpage オペランドの指定有無	pdbuffer -y オペランドの指定有無	デファードライトトリガ契機の更新バッファ面数
あり	あり	pdbuffer -y オペランド指定値
	なし	グローバルバッファ面数×pd_dbbuff_rate_updpage オペランド指定比率
なし	あり	pdbuffer -y オペランド指定値
	なし	HiRDB が自動計算します。

＜指定値の目安＞

通常はこのオペランドを省略してください。デファードライト処理がシンクポイント間隔内で終了しない場合など，更新バッファ面数を減らし，更新バッファヒット率を多少抑えてでも書き込み時間を短縮したいときに，このオペランドを指定します。指定する場合は，50%（HiRDB が設定する初期値）を目安にし，マニュアル「HiRDB システム運用ガイド」の「デファードライト処理適用時のシンクポイント処理時間のチューニング」を参照して，オペランドの値を決定してください。

＜規則＞

-y オプションにグローバルバッファ面数より大きい値を指定した場合は，グローバルバッファ面数と同じ値が適用されます。

＜注意事項＞

-y オプションの指定値が小さ過ぎると，デファードライト処理の回数が増えてしまい，デファードライト処理による負荷を高めるおそれがあります。このため，指定値の目安に従った設定をお勧めします。

(m)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdbufmod コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pdbufmod コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdbufmod コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdbufmod コマンドを実行すると，関連する RD エリア及び RD エリア状態に対して EX モードで排他を掛けます。そのため，関連する RD エリアを他トランザクション又はユティリティがアクセスする場合は待ち状態になります。
4. pdbufmod コマンドは，次の場合は実行できません。
   - 高速系切り替え機能，又はスタンバイレス型系切り替え機能を使用している場合
   - システム定義で pd_dbbuff_modify=Y を指定していない場合
5. pdbufmod コマンドは，複数同時には実行できません。
6. 定義できるグローバルバッファ数，及び新たに追加できる共用メモリセグメント数の上限は，システム定義の pd_max_add_dbbuff_no 及び pd_max_add_dbbuff_shm_no オペランドの指定値で決まります。
7. pdbufmod コマンドは，pdbuffer オペランドに-D を指定しているグローバルバッファを操作できません。

(6)　注意事項

1. グローバルバッファの削除に伴う共用メモリセグメントの削除は，コマンドの終了契機によって遅延したりします。すぐに削除したい場合は，pdpfresh コマンドを実行してください。
2. pdbufmod コマンド実行中に，ユニットが終了，又は HiRDB を強制終了した場合は，再開始後に pdbufls -k def コマンドで前回実行していた操作が完了しているか確認してください。完了していない場合は，再度 pdbufmod コマンドを実行してください。
3. pdbufmod コマンドのリターンコードを次に示します。

   0：正常終了

   4：警告終了（一部のサーバの処理が異常終了）

   8：異常終了

   12：異常終了（エラーメッセージを出力できない事象が発生）

リターンコードが 12 の場合は，シングルサーバ又はディクショナリサーバがあるホストのイベントログのエラーメッセージを参照し，エラーの要因を取り除いた後，再度 pdbufmod コマンドを実行してください。イベントログにエラーメッセージが出力されていない場合は，保守員に連絡してください。

(7) 使用例

例１：

グローバルバッファ（gbuf01）を追加して，そのグローバルバッファに追加した RD エリア（RDAREA1）を割り当てます。

```
pdbufmod -k add   .................1
         -a gbuf01   ..............2
         -r RDAREA1   .............3
         -n 1000   ................4
```

[説明]

1. グローバルバッファを追加する場合に指定します。
2. 追加するグローバルバッファの名称を指定します。
3. 割り当てる RD エリアを指定します。
4. グローバルバッファのバッファ面数を指定します。

例２：

RD エリアを割り当てていないグローバルバッファ（gbuf01）を削除します。

```
pdbufmod -k del   .................1
         -a gbuf01   ..............2
```

[説明]

1. グローバルバッファを削除する場合に指定します。
2. 削除するグローバルバッファの名称を指定します。

例３：

グローバルバッファ（gbuf01）のバッファ面数を 1000 から 2000 に変更します。

```
pdbufmod -k upd   .................1
         -a gbuf01   ..............2
         -n 2000   ................3
```

[説明]

1. グローバルバッファの定義を変更する場合に指定します。
2. 定義を変更するグローバルバッファの名称を指定します。
3. 変更後のバッファ面数を指定します。

2.11　pdcancel（UAP，ユティリティ処理の強制終了）

2.11.1　pdcancel の形式と規則

(1)　機能

UAP 又はユティリティを処理する HiRDB 側の処理プロセスを，UAP 又はユティリティから強制的に切り離し，終了します。

UAP が異常終了などで終了しているのに，HiRDB 側の処理プロセスがまだ処理途中の場合は，このコマンドを実行して終了します。このコマンドは，UAP と接続していた HiRDB 側のシングルサーバプロセス，又はフロントエンドサーバプロセス及びバックエンドサーバプロセスを強制終了します。

また，実行中のユティリティの処理を中断します。このコマンドを実行すると，対象の指定によって KFPS05080-Q，KFPS05081-Q 又は KFPS05082-Q メッセージのどれかを表示して強制終了を実行するかどうかを問い合わせます。"y"を応答すると，強制終了を実行します。"y"以外を応答すると，強制終了を中止してコマンドを終了します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdcancel　{-U | -u UAP識別子〔-i プロセスID〕 | -i プロセスID -d}
```

(b)　HiRDB/パラレルサーバの場合

```
pdcancel　{-x ホスト名 | -X ユニット識別子}
            {-U | -u UAP識別子　〔-i プロセスID〕 | -i プロセスID -d}
```

(4)　オプション

(a)　UAP の場合

-x ホスト名　～＜識別子＞((1～32))

該当する UAP が接続しているフロントエンドサーバがあるホストの名称を指定します。ホスト名は pdls コマンドで確認できます。

-X ユニット識別子　～＜識別子＞((4 文字))

該当する UAP が接続しているフロントエンドサーバがあるユニットの，ユニット識別子を指定します。

-U

指定したホスト名のユニットで動作している，すべての UAP を強制終了する場合に指定します。

-u UAP 識別子　～＜文字列＞((1～30))

強制終了する HiRDB 側の処理プロセスに対応する，クライアント UAP の UAP 識別子（PDCLTAPNAME）を指定します。該当するユニットで同一識別子を持つ UAP が複数動作している場合は，-i オプションを指定する必要があります。

-i プロセス ID　～＜符号なし整数＞((1～10))

強制終了するプロセスのプロセス ID を指定します。-u オプションと同時に指定した場合は，シングルサーバ，又はフロントエンドサーバのプロセス ID を指定してください。-d オプションと同時に指定した場合は，シングルサーバプロセス，ディクショナリサーバプロセス，フロントエンドサーバプロセス，又はバックエンドサーバプロセスを指定してください。プロセス ID は，pdls コマンド（-d オプションに prc を指定）で表示されます。したがって，該当する UAP のサービス要求受け付け時刻を基にして，強制終了するプロセスのプロセス ID を知ることができます。

-d

HiRDB 側の処理プロセスを強制終了する場合に指定します。プロセスがトランザクション処理中の場合は，トランザクション関係にあるディクショナリサーバプロセス，フロントエンドサーバプロセス，及びバックエンドサーバプロセスを強制終了します。

(b)　ユティリティの場合

-x ホスト名　～＜識別子＞((1～32))

システムマネジャがあるホストの名称を指定します。ただし，データベース初期設定ユティリティ（pdinit）及びデータベース構成変更ユティリティ（pdmod）を強制終了する場合は，ディクショナリサーバがあるホストの名称を指定してください。

ホスト名は pdls コマンドで確認できます。

-X　ユニット識別子　～＜識別子＞((4 文字))

ユティリティが動作しているユニットのユニット識別子を指定します。ただし，データベース初期設定ユティリティ及びデータベース構成変更ユティリティを強制終了する場合は，ディクショナリサーバがあるユニットのユニット識別子を指定してください。

ユニット識別子は pdls -d prc コマンドで確認できます。

-u UAP 識別子　～＜識別子＞((1～8))

ユティリティの処理をしているサーバの名称を指定します。サーバ名は，pdls -d prc -a コマンドで表示される，"0m"で始まる名称です。例えば，データベース作成ユティリティ（pdload）の場合，サーバ名は"0mload0"と表示されます。ただし，データベース構成変更ユティリティを強制終了する場合は，UAP 識別子として pdmod を指定します。また，pdrrpref コマンドをキャンセルする場合，サーバ名は"pdrrpref"を指定します。

各ユティリティのサーバの名称を次に示します。

ユティリティ	サーバの名称
データベース初期設定ユティリティ（pdinit）	0minitx※
データベース作成ユティリティ（pdload）	0mloadx※
データベース構成変更ユティリティ（pdmod）	pdmod
データベース再編成ユティリティ（pdrorg）	0mrorgx※
ディクショナリ搬出入ユティリティ（pdexp）	0mexpx※
リバランスユティリティ（pdrbal）	0mrbalx※
空きページ解放ユティリティ（pdreclaim）	0mrorgx※
グローバルバッファ常駐化ユティリティ（pdpgbfon）	0mrorgx※
データベース状態解析ユティリティ（pddbst）	0mdbstx※
最適化情報収集ユティリティ（pdgetcst）	0mgcstx※
データベース複写ユティリティ（pdcopy）	0bcpyx※
データベース回復ユティリティ（pdrstr）	0brstr

注※

末尾の x は，0，1，2...と数値文字になります。

-i プロセス ID　～<符号なし整数>((1～10))

強制終了するユティリティのプロセス ID を指定します。ユティリティの場合，このオプションは必ず指定してください。

-d

ユティリティを強制終了する場合に指定します。ユティリティがトランザクション処理中の場合は，トランザクション関係にあるシングルサーバプロセス，ディクショナリサーバプロセス，及びバックエンドサーバプロセスを強制終了します。

(5)　規則

1. pdcancel コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdcancel コマンドは，任意のサーバマシンから実行できます。
3. pdcancel コマンド（-d オプション指定なし）で UAP と接続していた HiRDB 側の処理プロセスを強制終了する場合，強制終了の対象となるプロセスはトランザクション処理中のものだけです。このため，トランザクション開始以前，又はトランザクション終了後であるのに，シングルサーバプロセス，又はフロントエンドサーバプロセスからの応答が UAP に返ってこない場合は，-d オプションを指定して pdcancel コマンドを実行するか，pdkill コマンドでシングルサーバプロセス，若しくはフロントエンドサーバプロセスを終了してください。該当する UAP に対応するプロセスがトランザクション処理中かどうかは，pdls コマンド（-d オプションに trn -a を指定）で表示される情報で判別できます。この

場合，該当する UAP に関する情報が表示されればトランザクション処理中であり，表示されなければトランザクション処理中でないことになります。

4. UAP が異常終了などで終了しているのに，HiRDB 側の処理プロセスがまだ処理途中の場合に pdcancel コマンドを実行しなかったときは，該当する UAP のクライアント環境定義（PDSWAITTIME）で指定した時間でプロセスは強制終了されます。

(6) 注意事項

1. pdcancel コマンドの結果は，pdls コマンド（-d prc 指定）で確認できます。
2. クライアント環境定義の PDCLTAPNAME に英数字以外の文字を指定した場合，pdcancel コマンドを実行できない場合があります。
3. ユティリティをキャンセルする場合，不当にキャンセルしないように注意してください。
4. pdcancel コマンド実行時のリターンコードは，0 の場合は正常終了，8 の場合は異常終了（オプション指定不正，rsh 失敗など）となります。

2.12　pdcat（ファイルの内容表示）

2.12.1　pdcat の形式と規則

(1)　共通機能

指定した表示対象種別ごとに HiRDB システムのファイルの内容を表示します。

(2)　共通形式

```
pdcat 〔-d 表示対象種別〕 〔表示対象種別ごとのオプション〕
```

注

表示対象種別ごとのオプションは，指定する表示対象種別によって異なります。

(3)　共通オプション

(a)　-d 表示対象種別

内容を表示するファイルの種別を指定します。

sts：

ステータスファイルの内容を表示します。

省略時：

メッセージログファイルの内容を表示します。

(4)　共通注意事項

1. pdcat コマンド実行時のリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. システム定義の pd_tmp_directory オペランドを指定しないで，環境変数 TMP だけを指定している場合，pdcat コマンド，又はほかの HiRDB のコマンドを実行中に中断すると，環境変数 TMP に設定したディレクトリ下に"pdcmd"で始まる名称のファイルが残ることがあります。pdcat コマンド，又はほかの HiRDB のコマンドが終了した後も，"pdcmd"で始まる名称のファイルが残っている場合は，エクスプローラなどで削除してください。

2.13　pdcat【-d sts】（ステータスファイルの内容表示）

2.13.1　pdcat【-d sts】の形式と規則

(1)　機能

ステータスファイルの内容を標準出力に出力します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　ユニット用ステータスファイル

```
pdcat -d sts 〔 {-x ホスト名 | -u ユニット識別子} 〕
      -f 物理ファイル名 〔-v〕 〔-e〕
```

(b)　サーバ用ステータスファイル

```
pdcat -d sts -s サーバ名 -f 物理ファイル名 〔-v〕 〔-e〕
```

(4)　オプション

(a)　{-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。省略した場合，HiRDB システム全体の情報を表示します。

-x ホスト名　～＜識別子＞((1～32))

ユニット用ステータスファイルに対応するホストのホスト名を指定します。

-u ユニット識別子　～＜識別子＞((4 文字))

ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

(b)　-s サーバ名　～＜識別子＞((1～8))

サーバ用ステータスファイルに対応するサーバ名を指定します。HiRDB/シングルサーバでは HiRDB サーバは一つですが，サーバ用ステータスファイルが操作対象であることを識別するために必ず指定してください。

(c)　-f 物理ファイル名　～<パス名>((167 文字以内))

物理ファイルの名称を，絶対パス名で指定します。物理ファイル名は，HiRDB ファイルシステム領域名¥HiRDB ファイル名で指定してください。

(d)　-v

ステータスファイルに関する次の内容をチェックします。

- レコード(先頭，終端)整合性チェック番号によるレコード破壊チェック
- pdstsinit コマンド実行時のファイル管理情報によるレコード長，レコード数チェック
- ステータスファイル管理情報によるファイル更新完了チェック
- ステータスファイル管理情報内のレコード数，レコード種別チェック

チェック結果が不正の場合，指定したステータスファイルの内容を表示した後に，エラーメッセージを出力します。

このオプションの指定を省略すると，ステータスファイルはチェックされません。

(e)　-e

ステータスファイルを排他的にオープンします。このため，ユニット，又はサーバの開始時にこのオプションを指定して pdcat コマンドを実行すると，HiRDB が排他エラーで異常終了する場合があります。

このオプションの指定を省略すると，ステータスファイルは排他的にオープンされません。このため，指定したステータスファイルを HiRDB で使用中の場合，正しい状態が表示されないことがあります。

(5)　規則

1. pdcat コマンドは，HiRDB の稼働に関係なく実行できます。ただし，HiRDB 開始処理中及び終了処理中は実行できません。
2. pdcat コマンドは，任意のサーバマシンから実行できます。

(6)　出力形式

```
Path name:aa...a
    Initial_time Current_time  R_le  R_co   factor Available Manage  Assign-Seq
    bb...b       cc...c        ddddd ee...e fff%   gg...g    hh...h      ii...i
```

［説明］

aa...a：

物理ファイルの名称(63 けた以内)

bb...b：

初期設定年月日と時刻

年(西暦の下 2 けた)/月/日　時:分:秒の形式

cc...c：

現用決定年月日と時刻

年(西暦の下 2 けた)/月/日　時:分:秒の形式

現用ファイルとして使用されていない場合，「--/--/--　--：--：--」と表示されます。

ddddd：

レコード長(10 進数)

ee...e：

レコード数(10 進数)

fff：

ファイル内のレコード使用率（%）

gg...g：

ファイル内の連続空きレコード数(10 進数)

hh...h：

ファイル内の管理レコード数(10 進数)

ii...i：

システムが使用する内部情報(10 進数)

なお，障害が発生したファイルや破壊されたファイルに対する表示内容は，不正となる場合があります。

2.14　pdcat【-d 省略】（メッセージログファイルの内容表示）

2.14.1　pdcat【-d 省略】の形式と規則

(1)　機能

メッセージログファイル中のメッセージの内容を表示します。

(2)　実行者

権限がないユーザでも実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdcat 〔- {a | b | ab} 〕 〔-c 重要度コード〔, 重要度コード〕…〕
      〔-y yyyymmdd〕 〔-t hhmmss〕 〔-T hhmmss〕 〔-n レコード数〕
      〔ファイル名〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdcat 〔- {a | b | ab} 〕 〔-c 重要度コード〔, 重要度コード〕…〕
      〔 {-x ホスト名 | -u ユニット識別子} 〕 〔-y yyyymmdd〕
      〔-t hhmmss〕 〔-T hhmmss〕 〔-n レコード数〕
      〔ファイル名〕
```

(4)　オプション

(a)　- {a ｜ b ｜ ab}

-a：

メッセージ ID 及びテキスト以外の情報を表示します。

-b：

メッセージ ID 及びテキストを表示します。

-ab：

メッセージのすべての情報を表示します。

(b)　-c 重要度コード

重要度コードでメッセージを選択する場合，その重要度コードを指定します。省略した場合，すべてのメッセージを表示します。

重要度コードについては，マニュアル「HiRDB メッセージ」を参照してください。

(c)　{-x ホスト名｜-u ユニット識別子}

ホストごと又はユニットごとにメッセージを表示する場合，そのホストのホスト名又はそのユニットのユニット識別子を指定します。省略した場合，すべてのホスト又はすべてのユニットのメッセージログを出力します。

-x ホスト名　～＜識別子＞((1～32))

ホストごとにメッセージを表示する場合，そのホストのホスト名を指定します。pdunit オペランドの-x オプションにホスト名を指定している場合は，そのホスト名を指定します。pdunit オペランドの-x オプションを省略している場合は，標準ホスト名を指定してください。このオプションを省略した場合は，すべてのホストのメッセージログを出力します。

なお，影響分散スタンバイレス型系切り替え機能を適用している場合，サーバに関するメッセージを出力するためには，サーバが実行系となっているホストのホスト名を指定する必要があります。系切り替えをしたサーバに関するメッセージは，サーバが実行系となっているホストのメッセージログに出力されるためです。

-u ユニット識別子　～＜識別子＞((4 文字))

ユニットごとにメッセージを表示する場合，そのユニットのユニット識別子を指定します。影響分散スタンバイレス型系切り替え機能を適用している場合，サーバに関するメッセージを出力するためには，サーバが実行系となっているユニットのユニット識別子を指定する必要があります。系切り替えをしたサーバに関するメッセージは，サーバが実行系となっているユニットのメッセージログに出力されるためです。

(d)　-y yyyymmdd

メッセージログの発生日付でメッセージを選択する場合，表示を開始するメッセージログの発生日付を，年月日で指定します。年は 4 けたの西暦で，月と日は 2 けたで指定します。省略した場合，すべてのメッセージログを表示します。

(e)　-t hhmmss

メッセージログの発生時刻でメッセージを選択する場合，表示を開始するメッセージログの発生時刻を，時分秒で指定します。省略した場合，すべてのメッセージログを表示します。

(f)　-T hhmmss

メッセージログの発生時刻でメッセージを選択する場合，表示を終了するメッセージログの発生時刻を，時分秒で指定します。省略した場合，すべてのメッセージログを表示します。

(g)　-n レコード数　～＜符号なし整数＞((1～32767))

表示するメッセージの数を限定したい場合，表示するメッセージログのレコード数を指定します。省略した場合，すべての情報を表示します。

(5)　コマンド引数

(a)　ファイル名　～＜パス名＞

表示するメッセージログファイルの名称を指定します。メッセージログファイルを複写したファイルも指定できます。

省略した場合，次のメッセージログファイルの作成日付の古い順にマージして表示します。

- %PDDIR%¥spool¥pdlog1
- %PDDIR%¥spool¥pdlog2

(6)　規則

1. pdcat コマンドは，HiRDB の稼働に関係なく実行できます。ただし，HiRDB 開始処理中及び終了処理中は実行できません。
2. pdcat コマンドは，任意のサーバマシンから実行できます。
3. HiRDB/パラレルサーバで，8 文字以上の名称のホストがある場合，-u オプションを指定してユニットごとに情報を出力してください。

(7)　注意事項

1. HiRDB/パラレルサーバの場合，システム定義の pd_mlg_msg_log_unit オペランドの指定値によって，メッセージログファイルの格納場所が次のようになります。
   - pd_mlg_msg_log_unit=manager の場合
     システムマネジャがあるユニット
   - pd_mlg_msg_log_unit=local の場合
     すべてのユニット
2. pdcat コマンドを実行した場合，pd_mlg_msg_log_unit=local を指定しているときは，該当ユニットのメッセージログファイルの内容を表示します。pd_mlg_msg_log_unit=manager を指定している場合，又は pd_mlg_msg_log_unit オペランドを省略している場合，システムマネジャがあるユニットでは該当ユニットのメッセージログファイルの内容を表示し，それ以外のユニットでは何も表示されません。
3. 8 文字以上の名称のホストがある場合，-x オプションに 8 文字以上のホスト名を指定すると，ホスト名の先頭 8 文字が同じホストの情報をすべて出力します。

(8) 出力形式

```
aa...a bb...b cc...c dddd ee...e ff...f gg...g hhhh ii...i
jjj kk...k ll...l
```

［説明］

項目ごとに 1 文字以上の半角空白を空けて表示します。

aa...a：

メッセージ通番（10 進数 7 けた以内）

bb...b：

プロセス ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，プロセス内メッセージ通番以降の項目の表示位置は右にシフトします。

cc...c：

プロセス内メッセージ通番（10 進数 7 けた以内）

dddd：

HiRDB 識別子（4 文字）

ee...e：

年月日（年/月/日の形式）（10 進数 10 けた）

ff...f：

時分秒（時:分:秒の形式）（10 進数 8 けた）

gg...g：

要求元ホスト名（先頭 8 文字）

影響分散スタンバイレス型系切り替え機能を適用している場合，サーバが実行系となっているホストのホスト名となります。

hhhh：

ユニット識別子（4 文字）

影響分散スタンバイレス型系切り替え機能を適用している場合，サーバが実行系となっているユニットのユニット識別子となります。

ii...i：

メッセージ出力要求元サーバ名（8 文字以内）

jjj：

システムが使用する内部情報（3 文字）

kk...k：

メッセージ ID（10 進数 11 けた以内）

ll...l：

メッセージテキスト（223 文字以内）

なお，一部のメッセージ及び，HiRDB の停止中などには，HiRDB 識別子，要求元のユニット識別子，又は要求元のサーバ名が設定されていない場合があります。

＜出力例＞

プロセス ID が 100000 以上の表示を含む場合

```
       12 38095       1 PRDB 2014/04/07 17:18:46 HOST0001 UNT0 _rdm
rdm KFPS05220-I HiRDB system termination process started. mode = NORMAL
 … プロセスIDが99999以下
       13 108221       0 PRDB 2014/04/07 17:18:47 HOST0001 UNT0 _pdustop
 adm KFPS01840-I Now terminating HiRDB unit UNT0
 … プロセスIDが100000以上
       14  8101       0 PRDB 2014/04/07 17:18:47 HOST0001 UNT0 _trn
trn KFPS00958-I Unit terminating; stops receiving new transactions
 … プロセスIDが99999以下
```

2.15　pdchgconf（システム構成変更コマンド）

2.15.1　pdchgconf の形式と規則

(1)　機能

HiRDB の稼働中に，システム定義を変更します。

pdchgconf コマンドを実行する前に，変更後の定義内容の妥当性を確認してください（pdconfchk -d chgconf コマンドを実行してください）。

pdchgconf コマンドは，HiRDB Advanced High Availability を組み込んでいる場合に使用できます。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdchgconf
```

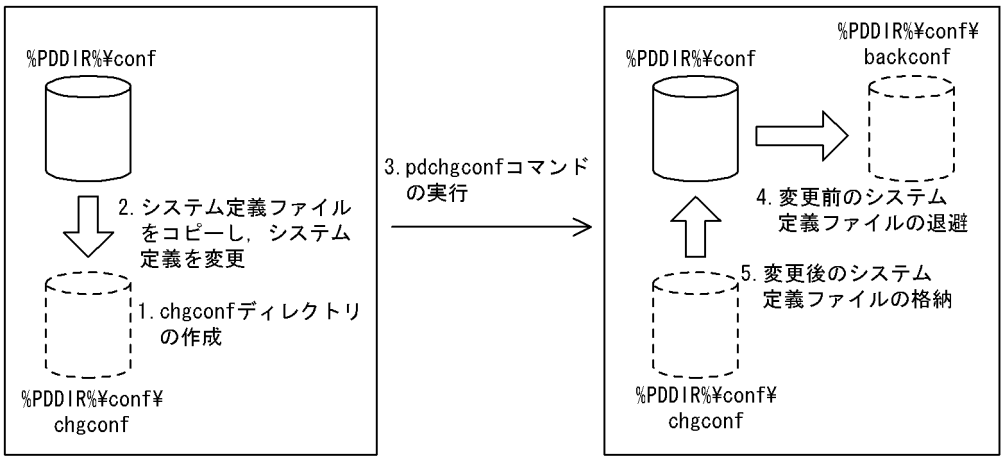
(4)　オプション

なし。

(5)　変更手順

システム定義の変更の概要を次の図に示します。

図 2–3　システム定義の変更の概要（pdchgconf コマンド）



［説明］

1. pdchgconf コマンドを実行する前に，%PDDIR%¥conf に変更後のシステム定義ファイルを格納するディレクトリ chgconf を作成してください。HiRDB/パラレルサーバの場合は，すべてのユニットに作成してください。
2. %PDDIR%¥conf にあるすべてのシステム定義ファイルを chgconf ディレクトリにコピーした後，システム定義の変更をしてください。

   変更できるシステム定義については，マニュアル「HiRDB システム運用ガイド」を参照してください。
3. pdchgconf コマンドを実行します。
4. pdchgconf コマンドが終了すると，変更前のシステム定義ファイルは%PDDIR%¥conf の backconf ディレクトリに退避されます。
5. %PDDIR%¥conf には変更後のシステム定義ファイルが格納されます。

(6) 規則

1. pdchgconf コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdchgconf コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(7) 注意事項

1. pd_system_id，及び pd_master_file_name オペランドは，pdchgconf コマンドでは変更できません。また，現用のシステムファイル，及び上書きできないシステムファイルに対しても削除はできません。これらの変更，又は削除をするとエラーになります。
2. pdchgconf コマンドを実行すると，システムログファイルが切り替わります。そのため，pdchgconf コマンドを実行する前に，変更後の構成で稼働するための容量が，システムログファイルにあるかどうかを確認してから実行してください。スワップ先にできるシステムログファイルがない状態で pdchgconf コマンドを実行すると，KFPS01256-E メッセージを出力し，HiRDB は停止します（アボートコード Psjnf07 又は Psjn381 を出力）。HiRDB が停止した場合，スワップ先にできるシステムログファイルを準備してから，HiRDB を開始してください。スワップ先にできるシステムログファイルがない場合，システムログファイルを追加する必要があります。スワップ先にできるファイルがない場合の対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
3. pdchgconf コマンドを実行すると，メッセージログファイルが切り替わります。ただし，メッセージログファイルの切り替えを示すメッセージは出力されません。そのため，メッセージログファイルのメッセージログが必要な場合は，pdchgconf コマンドを実行する前にメッセージログファイルのバックアップを取得してください。
4. 次の条件に該当する場合，pdchgconf コマンドは実行できないことがあります。
   - HiRDB/パラレルサーバの場合で，一部のユニット又はサーバが終了しているとき（縮退起動も含む）。なお，回復不要 FES のユニットが停止している場合，pdchgconf コマンドを実行できることがあります。詳細については，「回復不要 FES がある場合の注意事項」を参照してください。

- HiRDB/パラレルサーバの場合で，ユニット間のネットワークに通信障害が発生しているとき
- pdrplstop コマンドを実行中の場合
- pdchgconf コマンド実行後に，15 分間以上トランザクション，又はユティリティが動作した場合
- pdstart -r コマンドで HiRDB を開始している場合
- インメモリ RD エリアがある場合
- HiRDB Datareplicator 連携中に，システム定義の pd_rpl_init_start オペランドの内容と，実際のデータ抽出モードが異なっている場合。この場合の詳細を次に示します。

変更前の pd_rpl_init_start オペランドの内容	HiRDB Datareplicator 連携の状態	変更後の pd_rpl_init_start オペランドの内容	pdchgconf コマンドの実行可否
Y	実行中	Y	○
		N 又は省略	×
	pdrplstop コマンド実行で停止中	Y	×
		N 又は省略	○
N 又は省略	停止中	Y	×
		N 又は省略	○
	pdrplstart コマンド実行で実行中	Y	○
		N 又は省略	×

（凡例）

○：実行できます。

×：実行できません。

- 次に示す条件をすべて満たす場合

  (i)pd_mode_conf オペランドに AUTO を指定している

  (ii)回復不要 FES を定義したユニットが停止している

  (iii)システム定義の変更後に，上記(ii)のユニットの回復不要 FES の適用をやめる

5. pdchgconf コマンドの実行中は，次の運用に制限が生じます。
   - UAP のレスポンスが遅くなることがあります。pdchgconf コマンドの実行時間の目安は，「HiRDB を正常終了する時間＋ HiRDB を正常開始する時間」です。
   - ほかの運用コマンド及びユティリティは実行しないでください。
   - 系切り替え機能は使用できません。Hitachi HA Toolkit Extension を使用した系切り替え機能の場合は，pdchgconf コマンドを実行する前に待機系を停止して，pdchgconf コマンドが正常終了した後に変更後のシステム定義に入れ替え，待機系を再開始してください。
6. HiRDB Datareplicator 連携中に，ユニット又はサーバの追加，削除をする場合，HiRDB Datareplicator の抽出環境の再構築が必要となります。HiRDB Datareplicator の抽出環境の再構築中に，HiRDB のト

ランザクションが動作して抽出側 DB の追加，更新，及び削除がされると，反映側 DB との不整合が発生します。この場合，pdchgconf コマンドでの構成変更はしないでください。

7. pdchgconf コマンドの実行が失敗した場合，%PDDIR%¥conf 下の backconf ディレクトリにある変更前のシステム定義ファイルを使用して，自動的にシステム定義変更前の状態に戻します。戻す処理が失敗した場合，HiRDB が終了することがあります。この場合，エラーメッセージを参照し対処してください。
8. pdchgconf コマンド実行中，HiRDB の開始，終了に伴うエラーメッセージ及びアボートコードが出力されることがあります。この場合，変更後のシステム定義，又は環境に問題があることがあります。エラーメッセージ，及びアボートコードの対処方法に従って対処してください。
9. pdchgconf コマンドのリターンコードを次に示します。

   0：正常終了

   8：異常終了

   リターンコードが 8 の場合は，HiRDB が停止している可能性があるため，マニュアル「HiRDB システム運用ガイド」の「HiRDB の稼働中に HiRDB システム定義を変更する方法（システム構成変更コマンド）」を参照して対処してください。
10. 回復不要 FES のユニットが停止している場合，次に示す状態では pdchgconf コマンドを実行できないことがあります。
    - システム定義の pd_mode_conf オペランドの指定値が AUTO の場合
    - システム定義変更後に，停止しているユニットのフロントエンドサーバが回復不要 FES でなくなる場合

(8) 回復不要 FES がある場合の注意事項

1. 回復不要 FES が正常停止していて，該当するユニットが稼働している場合，pdchgconf コマンドは実行できません。
2. 回復不要 FES ユニットのサーバマシンがダウンしている場合，pdchgconf コマンドの終了に時間が掛かることがあります。
3. サーバマシンの起動後に，回復不要 FES ユニットを一度も開始していない場合，pdchgconf コマンドがエラー終了することがあります。
4. 回復不要 FES のユニット以外のユニットがすべて稼働していて，かつ単独停止中のサーバがない場合，回復不要 FES のユニットが停止していても，pdchgconf コマンドを実行できます。ただし，注意事項の 5.～9.に該当する場合は実行できません。

   pdchgconf コマンドは，停止しているユニットの%PDDIR%¥conf 下にあるシステム定義ファイルを，%PDDIR%¥conf¥chgconf 下のシステム定義ファイルに置き換えます。また，停止しているユニットは，変更後のシステム定義を反映するために再度開始します。ただし，マシンダウンやネットワーク障害などによって，pdchgconf コマンドは，停止しているユニットのシステム定義ファイルの置き換えや開始をしないことがあります。その場合，KFPS04665-W メッセージが表示されます。KFPS04665-W メッセージに表示されたユニットに対しては，ユニットを再開始する前に，次の(i)，(ii)の操作を必ず行ってください。行わなかった場合，HiRDB がシステム定義の変更を正しく受け付けられなくなり，

不正な動作をすることがあります。この場合，すぐに該当ユニットを停止し，(i)，(ii)の操作を行ってください。

(i) %PDDIR%¥conf 下，%PDCONFPATH%下のシステム定義ファイルを，変更後のシステム定義ファイルに置き換える。

(ii) pdconfchk コマンドで，ユニット間のシステム定義に差異がないことを確認する。

2.16　pdchpathf（ファイルパス変更オフラインコマンド）

2.16.1　pdchpathf の形式と規則

(1)　機能

RD エリアの構成ファイルを格納している HiRDB ファイルシステム領域を移動するために，HiRDB ファイルに登録されているパス情報を変更します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdchpathf 制御文ファイル パス変更情報ファイル
```

(4)　オプション

(a)　制御文ファイル　～＜パス名＞

制御文ファイルには変更するファイルの情報を次の形式で記述してください。HiRDB/パラレルサーバの場合，一つの制御文ファイルに複数の制御文を記述できます。制御文にはコメントは記述できません。

制御文

```
オペランド
change path
 〔host ホスト名〕
area "HiRDBファイルシステム領域名"
 〔area "HiRDBファイルシステム領域名"
  〕…
;
```

host ホスト名：

～＜識別子＞((1～32))

【HiRDB/シングルサーバの場合】

指定不要です。指定しても無視されます。

【HiRDB/パラレルサーバの場合】

HiRDB ファイルシステム領域を定義しているユニットのホスト名を指定してください。システム定義 pdunit の-x 又は-c に指定しているホスト名を指定してください。同一のホスト名を複数の制御文に指定することはできません。

area "HiRDB ファイルシステム領域名"：

～＜パス名＞((165 文字以内))

変更する HiRDB ファイルシステム領域名のパス名称を指定します。

HiRDB ファイルシステム領域名は引用符(")で囲んで指定してください。指定した HiRDB ファイルシステム領域に格納している最大の HiRDB ファイル名長と，HiRDB ファイルシステム領域名長が合わせて 166 文字以内になるようにしてください。

複数の HiRDB ファイルシステム領域を変更する場合，それぞれの変更前後の HiRDB ファイルシステム領域名が重ならないようにしてください。重なった場合，コマンドがエラーになります。

<問題のある変更の例>

- HiRDB ファイルシステム領域名 A のファイルパスを B に変更し，HiRDB ファイルシステム領域名 B のファイルを A に変更する（ファイル名の入れ替え）。
- HiRDB ファイルシステム領域名 A のファイルを B に変更し，HiRDB ファイルシステム領域名 B のファイルを C に変更する（ファイル名の玉突き）。

<注意事項>

マスタディレクトリ用 RD エリアの構成ファイルのファイルパスを変更する場合は，最初の制御文に指定してください。最初の制御文に指定しない場合，コマンドがエラー終了します。

(b)　パス変更情報ファイル　～＜パス名＞

パス変更情報ファイルは，pdchpathn コマンドの入力ファイルです。指定したファイルがある場合はコマンドがエラー終了します。パス変更情報ファイルのサイズを次に示します。

```
パス変更情報ファイルサイズ(単位：バイト）＝ 20＋ｂ×384
```

[説明]

b：

制御文に指定した HiRDB ファイルシステム領域に作成している HiRDB ファイル数の合計。

(5)　規則

1. pdchpathf コマンドは，HiRDB が停止中のときだけ実行できます。pdchpathf コマンド実行後は，HiRDB を起動して pdchpathn コマンドを実行してください。
2. pdchpathf コマンドは，シングルサーバ又はディクショナリサーバがあるサーバマシンで実行してください。
3. pdchpathf コマンド実行後は，pdchpathn コマンドの実行が完了するまで再実行できません。変更する HiRDB ファイルシステム領域が複数ある場合は，制御文ファイルにすべて記載してください。
4. マスタディレクトリ用 RD エリアのファイルを変更する場合，先頭の制御文に記載する必要があります。
5. pdchpathf コマンドは，HiRDB を正常停止した状態で実行してください。強制終了，異常終了状態で実行した場合，再開始時の更新ログによる回復中に DB 破壊となります。

(6) 注意事項

1. pdchpathf コマンドのリターンコードを次に示します。

   0：正常終了

   4：異常終了（回復不要）

   8：異常終了（回復要）

   pdchpathf コマンドのリターンコードが 4 の場合は，制御文の情報に問題があります。この場合，制御文ファイルを修正して再実行できます。リターンコードが 8 の場合は，ファイル更新中にエラーが発生しているため，実行前のバックアップから回復が必要です。なお，リターンコードは KFPH27072-I メッセージで確認できます。

2. リアルタイム SAN レプリケーションを使用し，リモートサイトへのデータ反映方式にログ同期方式を適用している場合，pdchpathn コマンド実行後に業務サイトとログ適用サイトの HiRDB 構成を同じにしてから，システムログ適用化を実施してください。

2.17　pdchpathn（ファイルパス変更オンラインコマンド）

2.17.1　pdchpathn の形式と規則

(1)　機能

pdchpathf コマンドが出力したパス変更情報ファイルを基に，ディクショナリ表で管理している HiRDB ファイルシステム領域のパスを変更します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdchpathn 〔-W 実行監視時間〕 パス変更情報ファイル
```

(4)　オプション

(a)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdchpathn コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。

このオプションに 0 を指定した場合は，監視を行いません。

このオプションを省略した場合は，pd_cmd_exec_time オペランドの指定値が仮定されます。詳細はマニュアル「HiRDB システム定義」の pd_cmd_exec_time オペランドを参照してください。

なお，pdchpathn コマンドの実行時間は，パスを変更する HiRDB ファイル数に依存します。そのため，変更する HiRDB ファイル数が多いほど，実行時間が長くなります。

(5)　コマンド引数

(a)　パス変更情報ファイル　～＜パス名＞

pdchpathf コマンドが出力したパス変更情報ファイルのパスを指定します。

SDS 又は DS があるホスト上でアクセスできるファイルを指定してください。

HiRDB/パラレルサーバの場合で，MGR と DS が別のユニットにある場合は，DS があるホスト上での絶対パスを指定してください。

(6) 規則

1. pdchpathn コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdchpathn コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdchpathn コマンドを実行する場合，データディクショナリ用 RD エリア及びマスタディレクトリ用 RD エリアがオープンかつ閉塞なし状態である必要があります。

(7) 注意事項

1. pdchpathn コマンドのリターンコードを次に示します。

   0：正常終了

   8：異常終了

   異常終了時は，出力したエラーメッセージを参照して，エラー要因を取り除き再度コマンドを実行してください。
2. pdchpathn コマンドを実行する前に，マニュアル「HiRDB システム運用ガイド」の「ファイルパス変更コマンドを使用した移動手順」の注意事項を参照してください。
3. pdchpathn コマンドは，ディクショナリ表を更新するため，システムファイルに更新ログを出力します。マニュアル「HiRDB システム導入・設計ガイド」の「ファイルパス変更オンラインコマンド（pdchpathn）実行時に出力されるシステムログ量」を参照して，更新ログ量を見積ってください。
4. pdchpathf コマンド実行前に取得したバックアップは使用できません。pdchpathn コマンドの正常終了後，全 RD エリアのバックアップを取得してください。
5. リアルタイム SAN レプリケーションを使用し，リモートサイトへのデータ反映方式にログ同期方式を適用している場合，pdchpathn コマンド実行後に業務サイトとログ適用サイトの HiRDB 構成を同じにしてから，システムログ適用化を実施してください。

2.18　pdchprc（サーバプロセスの起動本数変更）

2.18.1　pdchprc の形式と規則

(1)　機能

サーバプロセスの常駐プロセス数と最大起動プロセス数を変更します。また，サーバプロセスの常駐プロセス数と最大起動プロセス数を表示します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdchprc 〔-p常駐プロセス数〔，最大起動プロセス数〕〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdchprc {-a | -sサーバ名}
          〔-p常駐プロセス数〔，最大起動プロセス数〕〕
```

(4)　オプション

(a)　-p 常駐プロセス数〔，最大起動プロセス数〕　～＜符号なし整数＞

- HiRDB/シングルサーバの場合：((0～3000))
- HiRDB/パラレルサーバの場合：((0～12000))

サーバプロセスの常駐プロセス数及び最大起動プロセス数を指定します。最大起動プロセス数を省略した場合，常駐プロセス数と同じ値が仮定されます。また，-p オプションを省略した場合，pdchprc コマンド実行時のサーバプロセスの常駐プロセス数，及び最大起動プロセス数を表示します。

プロセス数には，0～HiRDB のシステム定義で指定したサーバプロセスの最大起動プロセス数までの値を指定できます。サーバプロセスの最大起動プロセス数を次に示します。

- シングルサーバ又はフロントエンドサーバの場合
  システム共通定義の pd_max_users オペランドの値
- バックエンドサーバの場合

サーバ共通定義又はバックエンドサーバ定義の pd_max_bes_process オペランドの指定値

- ディクショナリサーバの場合
  サーバ共通定義又はディクショナリサーバ定義の pd_max_dic_process オペランドの指定値

(b) -a

pdchprc コマンドを実行したときに起動している，すべてのサーバのサーバプロセスの常駐プロセス数，最大起動プロセス数を表示します。-p オプションと同時に指定した場合は，起動中のすべてのフロントエンドサーバとバックエンドサーバの，サーバプロセスの常駐プロセス数，最大起動プロセス数を変更します。

(c) -s サーバ名 ～<識別子>((1～8))

サーバプロセスの常駐プロセス数，最大起動プロセス数の変更又は表示をする，サーバの名称を指定します。

(5) 規則

1. pdchprc コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdchprc コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. -p オプションに指定した常駐プロセス数の値が最大起動プロセス数より大きい場合，エラーとなります。
4. -p オプションに，HiRDB のシステム定義で指定したサーバプロセスの最大起動プロセス数より大きな値を指定した場合は，警告メッセージが表示され，HiRDB のシステム定義で指定したサーバプロセスの最大起動プロセス数に値を変更します。
5. -p オプションでの常駐プロセス数，最大起動プロセス数に値が変更された場合，HiRDB はプロセス数を変更した旨のメッセージをサーバごとに表示します。
6. pdchprc コマンドで変更したサーバプロセスの常駐プロセス数及び最大起動プロセス数は，該当するサーバが停止するまで有効となります。HiRDB が正常終了，強制終了，又は計画停止した後の再開始では，変更した情報は引き継がれません。

(6) 注意事項

1. -s オプションに，停止中又は停止処理中のサーバを指定した場合，pdchprc コマンドはエラー終了します。また，-a オプションを指定した場合，停止中又は停止処理中のサーバに対しては，サーバプロセスの常駐プロセス数，最大起動プロセス数の変更及び表示はできません。
2. ディクショナリサーバのサーバプロセスの常駐プロセス数，最大起動プロセス数を変更する場合は，-s オプションにディクショナリサーバのサーバ名を指定してください。
3. サーバプロセスの最大起動プロセス数を 0 に変更した場合，pdchprc コマンド実行後に発生したトランザクション処理は，すべてエラーとなります。
4. ディクショナリサーバ又はバックエンドサーバの最大起動プロセス数を，HiRDB のシステム定義で指定した値より小さくした場合，そのサーバにトランザクション処理が集中したときは，処理しきれなかったトランザクションがエラーとなることがあります。

5. HiRDB/パラレルサーバで，システムマネジャとフロントエンドサーバが同じユニットの場合，フロントエンドサーバの最大起動プロセス数を 0 に変更すると，次に示すユティリティは実行できなくなります。
   - データベース作成ユティリティ（pdload）
   - データベース構成変更ユティリティ（pdmod）
   - データベース再編成ユティリティ（pdrorg）
   - データベース状態解析ユティリティ（pddbst）
   - 最適化情報収集ユティリティ（pdgetcst）
6. 1：1 スタンバイレス型系切り替え機能を適用する構成で，系切り替えによって代替 BES が正規 BES の処理を代行している（代替中）場合，正規 BES に対して pdchprc コマンドを実行すると次のようになります。
   - -s オプションを指定して pdchprc コマンドを実行すると，コマンドはエラー終了します。
   - 正規 BES のサーバプロセスの常駐プロセス数及び最大起動プロセス数の変更及び表示はできません。

   1：1 スタンバイレス型系切り替え機能を適用する場合のサーバプロセス数については，マニュアル「HiRDB システム運用ガイド」の「HiRDB システム定義の作成」の「系切り替え後のサーバプロセスの割り当て」を参照してください。
7. 影響分散スタンバイレス型系切り替え機能を適用している場合，pdchprc コマンドでサーバプロセスの常駐プロセス数及び最大起動プロセス数を表示させると，サーバプロセス割り当て数によってはバックエンドサーバプロセスの常駐プロセス数及び最大起動プロセス数が変更されて表示されることがあります。また，-p オプションを指定して常駐プロセス数及び最大起動プロセス数を変更しても，サーバプロセス割り当て数によっては，バックエンドサーバプロセスの常駐プロセス数及び最大起動プロセス数が変更されないことがあります。影響分散スタンバイレス型系切り替え機能使用時のサーバプロセス割り当て数については，マニュアル「HiRDB システム運用ガイド」を参照してください。
8. pdchprc コマンドのリターンコードを次に示します。

   0：正常終了

   4：一部のユニットが正常終了

   一部のユニットでサーバプロセス数を変更できていないおそれがあります。

   その状態で業務を継続すると同時に実行可能なトランザクション数が減るなどの問題が発生することがあります。

   メッセージに従って対処した後に再度コマンドを実行してください。

   8：異常終了（オプション指定不正，rsh 失敗など）

   サーバプロセス数を変更できていないおそれがあります。

   その状態で業務を継続すると同時に実行可能なトランザクション数が減るなどの問題が発生することがあります。

   メッセージに従って対処した後に再度コマンドを実行してください。

(7)　出力形式

-p オプションを省略したときだけ，表示されます。

```
aa...a          bb...b,cc...c          dd...d,ee...e
aa...a          bb...b,cc...c          dd...d,ee...e
```

［説明］

aa...a：

サーバ名（8 文字以内）

bb...b：

HiRDB のシステム定義で指定したサーバプロセスの常駐プロセス数（10 進数 5 けた以内で右詰め表示）

cc...c：

HiRDB のシステム定義で指定したサーバプロセスの最大起動プロセス数（10 進数 5 けた以内で右詰め表示）

dd...d：

pdchprc コマンドで変更したサーバプロセスの常駐プロセス数（10 進数 5 けた以内で右詰め表示）

bb...b と同じ場合は****を表示します。

ee...e：

pdchprc コマンドで変更したサーバプロセスの最大起動プロセス数（10 進数 5 けた以内で右詰め表示）

cc...c と同じ場合は****を表示します。

＜出力例＞

```
dic1          10, 100   ****,****          … プロセス数が10000未満の場合
fes1          10, 100   ****,****          … プロセス数が10000未満の場合
bes1       5000,10000     500,****          … プロセス数が10000以上の場合
bes2       10000,12000    500,10000         … プロセス数が10000以上の場合
```

2.19　pdclibsync（C ライブラリファイルの操作）

2.19.1　pdclibsync の形式と規則

(1)　機能

C ライブラリファイルに対して次に示す操作ができます。

- C ライブラリファイルの新規登録
- C ライブラリファイルの再登録
- C ライブラリファイルの削除
- C ライブラリファイルの一覧表示
- C ライブラリファイルのホスト間の再登録（HiRDB/パラレルサーバ限定）

このコマンドは，C ストアドプロシジャ又は C ストアドファンクションを使用するときに使用します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdclibsync 〔 {-I | -R | -D | -L} 〕
           〔-u 認可識別子〔, 認可識別子〕…〕
           〔-f Cライブラリファイル名〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdclibsync 〔 {-I | -R | -D | -L | -S} 〕
           〔-x ホスト名〔, ホスト名〕…〕
           〔-u 認可識別子〔, 認可識別子〕…〕
           〔-f Cライブラリファイル名〕
```

(4) オプション

(a) {-I ｜-R ｜-D ｜-L ｜-S}

pdclibsync コマンドで実行する処理を指定します。

-I

C ライブラリファイルを HiRDB に新規登録します。C ライブラリファイルは，pd_c_library_directory オペランドで指定したディレクトリ下に格納されます。

このオプションを指定した場合は，-u 及び-f オプションを必ず指定してください。

-f オプションで指定した C ライブラリファイルが既に登録されている場合に，このオプションを指定してコマンドを実行するとエラーになります。

-R

C ライブラリファイルを HiRDB に再登録します（登録済みの C ライブラリファイルを上書きします）。

このオプションを指定した場合は，-u 及び-f オプションを必ず指定してください。

-D

C ライブラリファイルを削除します。

このオプションを指定した場合は，-u 及び-f オプションを必ず指定してください。

-f オプションで指定した C ライブラリファイルが HiRDB に登録されていない場合に，このオプションを指定してコマンドを実行するとエラーになります。

-L

HiRDB に登録されている C ライブラリファイルの一覧を表示します。

-S

このオプションは，HiRDB/パラレルサーバ限定のオプションです。

-x オプションで指定したサーバマシンの C ライブラリファイルをコピー元とし，HiRDB/パラレルサーバを構成する全サーバマシンにその C ライブラリファイルを再登録します。-x オプションを省略した場合，ディクショナリサーバが定義されているサーバマシンの C ライブラリファイルをコピー元とします。

例えば，サーバマシンを追加したときにこのオプションを使うと，追加したサーバマシンに C ライブラリファイルを登録できます。

pdclibsync コマンドに指定するオプションの関係を次の表に示します。

表 2–1　pdclibsync コマンドに指定するオプションの関係

C ライブラリファイルの操作種別のオプション	ほかのオプション		
	-x	-u	-f
-I	○	◎	◎
-R	○	◎	◎

C ライブラリファイルの操作種別のオプション	ほかのオプション		
	-x	-u	-f
-D	○	◎	◎
-L	○	○	○
-S	○※	○	○

（凡例）

◎：必ず指定してください。

○：任意に指定できます。

注※

-x オプションを指定する場合は，ホスト名を一つだけ指定してください。

(b)　-x ホスト名〔，ホスト名〕...

このオプションは，HiRDB/パラレルサーバ限定のオプションです。

・-S オプションを指定した場合

コピー元の C ライブラリファイルがあるサーバマシンのホスト名を指定します。ホスト名を二つ以上指定することはできません。

このオプションを省略した場合は，ディクショナリサーバが定義されているサーバマシンのホスト名が仮定されます。

・-S オプション以外を指定した場合

pdclibsync コマンドの処理対象となるホストを限定する場合に指定します。

例えば，-D オプションを指定して C ライブラリファイルを削除する場合に，-x オプションを指定すると，指定したホストにある C ライブラリファイルだけが削除されます。

(c)　-u 認可識別子〔，認可識別子〕...

認可識別子を指定します。認可識別子に MASTER 又は HiRDB を指定することはできません。また，指定した文字列の英大文字と英小文字は区別されます。

-I，-R，及び-D オプションを指定した場合は，このオプションを必ず指定してください。

・-I オプションを指定した場合

C ライブラリファイルの所有者となる認可識別子を指定します。C ストアドプロシジャ又は C ストアドファンクションの所有者となる認可識別子を指定してください。

HiRDB 管理者とここで指定した認可識別子のユーザだけが，登録した C ライブラリファイルを操作できます。

・-I オプション以外を指定した場合

指定した認可識別子のユーザが所有する C ライブラリファイルがコマンドの処理対象になります。

(d)　-f C ライブラリファイル名

操作対象の C ライブラリファイルの名称を指定します。なお，英大文字と英小文字は同一文字として扱われます。

-I，-R，及び-D オプションを指定した場合は，このオプションを必ず指定してください。

-S，-L オプション指定時にこのオプションを省略した場合は，登録されているすべての C ライブラリファイルが操作対象になります。

(5)　規則

1. pdclibsync コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdclibsync コマンドは，任意のサーバマシンから実行できます。
3. pdclibsync コマンドは，複数同時に実行しないでください。

(6)　リターンコード

pdclibsync コマンドのリターンコードを次に示します。

0：正常終了

4：異常終了

C ライブラリファイルに対する処理を実行する前に異常終了したときに返されるリターンコードです。

例えば，新規登録を行った場合に C ライブラリファイルが既に登録されている，削除を行った場合に該当する C ライブラリファイルがないときにこのリターンコードが返されます。

8：異常終了

C ライブラリファイルに対する処理の実行中に異常終了したときに返されるリターンコードです。

この場合，出力されたメッセージを参照し，障害原因を取り除いた後にコマンドを再実行してください。なお，このリターンコードが返された場合，C ライブラリファイルが壊れている可能性があるため，C ライブラリファイルの一覧を表示してください。ファイルサイズが通常と異なる C ライブラリファイルがある場合，そのファイルが壊れている可能性があります。この場合，C ライブラリファイルを再登録するなどの対処を取ってください。

(7)　注意事項

1. トランザクションの実行中に，そのトランザクション中で実行される C ストアドルーチンが使用する C 関数を含む C ライブラリファイルを再登録及び削除しないでください。

(8)　出力形式

pdclibsync コマンドのオプションに-L を指定，又はオプションを省略した場合は，次の情報が表示されます。

```
HOSTNAME(aabbcc)  DATE        TIME       SIZE    OWNER   FILE
dd...d            eeee/ff/gg  hh:ii:jj   kk...k  ll...l  mm...m
   :                :           :          :       :       :
```

［説明］

aabbcc：

pdclibsync コマンドを実行した時刻が表示されます。

dd...d：

pdclibsync コマンドを実行したホスト名が表示されます。

eeee/ff/gg：

C ライブラリファイルの日付※が表示されます。

hh:ii:jj：

C ライブラリファイルの時刻※が表示されます。

kk...k：

C ライブラリファイルの大きさが表示されます（単位：バイト）。

ll...l：

C ライブラリファイルの所有者名が表示されます。

mm...m：

C ライブラリファイルの名称が表示されます。

注※

SQL 文（INSTALL CLIB，REPLACE CLIB）で C ライブラリファイルを新規登録又は再登録した場合は，SQL 文を実行した日付及び時刻となります。pdclibsync コマンドで C ライブラリファイルを新規登録又は再登録した場合は，新規登録又は再登録した C ライブラリファイルの日付及び時刻となります。

2.20　pdclose（RD エリアのクローズ）

2.20.1　pdclose の形式と規則

(1)　機能

指定した RD エリアをクローズします。pdhold コマンドによって閉塞した RD エリア，又は障害閉塞状態の RD エリアを指定します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdclose -r　{RDエリア名〔, RDエリア名〕…|ALL}　〔-W 実行監視時間〕
```

(4)　オプション

(a)　-r {RD エリア名〔，RD エリア名〕...｜ALL}

RD エリア名　～＜識別子＞((1～30))

クローズする RD エリア名を指定します。

ALL

マスタディレクトリ用 RD エリアを除くすべての RD エリアをクローズします。

＜規則＞

1. RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。
2. 参照可能バックアップ閉塞（更新 WAIT モード），更新可能バックアップ閉塞の RD エリア，及びマスタディレクトリ用 RD エリアはクローズできません。
3. ユーザ用 RD エリア又はユーザ LOB 用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，かつ pdhold コマンドによる閉塞状態
4. データディクショナリ用 RD エリアとユーザ用 RD エリア，又はユーザ LOB 用 RD エリアを同時に指定する場合は，データディクショナリ用 RD エリア名を最後に指定してください。

(b)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdclose コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pdclose コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdclose コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdclose コマンドは，RD エリアに対してモード EX で排他制御をします。そのため，指定した RD エリアにほかのトランザクションがアクセスしている場合は，そのトランザクションの終了まで pdclose コマンドは待ち状態となります。
4. 共用 RD エリアに対して pdclose コマンドを実行した場合，全バックエンドサーバに排他を掛けます。このため，該当する RD エリアへのアクセスを複数同時に実行している場合，グローバルデッドロックが発生し，タイムアウトする可能性があります。グローバルデッドロックが発生した場合，再度 pdclose コマンドを実行してください。
5. DB 非同期状態又は RD エリア障害状態の場合に pdclose コマンドを実行すると，インメモリデータバッファ上のデータをディスクに書き込み，インメモリデータバッファと RD エリアの同期を取ります（DB 同期状態になります）。

(6)　注意事項

1. pdclose コマンドの結果は，pddbls コマンドで確認できます。
2. pdclose コマンドのリターンコードと基本的な対処方法を次に示します。

   0 以外の場合，一部の RD エリアがクローズしていることがあります。その状態では業務を継続できないおそれがあるため，pdclose コマンドに指定した RD エリアを pdrels コマンド（-o オプション指定）でオープンしてください。

   0：正常終了

   4：警告終了

   一部の RD エリアの処理がエラー終了しました。

   8：異常終了

   一部の RD エリアの処理がエラー終了していることがあります。

   12：異常終了

   エラーメッセージを表示できない事象が発生しました。

一部の RD エリアの処理がエラー終了していることがあります。シングルサーバ又はディクショナリサーバがあるホストのイベントログ中のエラーメッセージを参照して，エラーの要因を取り除いた後，再度コマンドを実行してください。イベントログ中にエラーメッセージが出力されていない場合は保守員に連絡してください。

2.21　pdclttrc（SQL トレースの動的取得）

2.21.1　pdclttrc の形式と規則

(1)　機能

SQL 実行中のクライアントの，SQL トレースを取得します。SQL トレースは，クライアントに出力されます。

SQL トレースを取得するサーバのサーバ名とプロセス ID は，pdls -d prc コマンドの実行結果の，クライアントの IP アドレス，UAP の識別情報から特定できます。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdclttrc 〔-s サーバ名〕 〔-p プロセスID〕 〔-e〕 〔-l PDUAPREPLVL指定値〕
        〔-m ?パラメタ及び検索データの最大データ長〕
        〔-n 出力オペレーションコード数〕 〔-o SQLトレースファイルのサイズ〕
```

(4)　オプション

(a)　-s サーバ名　～＜識別子＞((1～8))

SQL トレースを取得するクライアントが接続している，サーバの名称を指定します。シングルサーバ又はフロントエンドサーバのサーバ名を指定してください。

このオプションを省略した場合，HiRDB のすべてのサーバ（シングルサーバ又はフロントエンドサーバ）に接続しているプロセスの SQL トレースを取得します。

シングルサーバの場合，不正なサーバ名を指定してもエラーにはなりません。

(b)　-p プロセス ID　～＜符号なし整数＞((1～10))

SQL トレースを取得するクライアントが接続している，サーバのプロセス ID を指定します。

HiRDB/パラレルサーバの場合，必ず-s オプションと同時に指定してください。HiRDB/シングルサーバの場合，-s オプションの指定は任意です。

このオプションを省略した場合，シングルサーバ又はフロントエンドサーバに接続しているすべてのプロセスの SQL トレースを取得します。

(c)　-e

SQL トレースの取得を停止します。

＜規則＞

1. 既に接続が切断されていたり，-n オプションの指定値に達して SQL トレースの取得が停止したりしている状態で，このオプションを指定してもエラーにはなりません。
2. このオプションだけ指定した場合，HiRDB のすべてのサーバ（シングルサーバ又はフロントエンドサーバ）に接続しているプロセスの SQL トレース取得を停止します。
3. 同時に-s オプションを指定した場合，指定したサーバに接続しているすべてのプロセスの SQL トレース取得を停止します。
4. 同時に-p オプションを指定した場合，指定したプロセスの SQL トレース取得を停止します。
5. クライアント側への停止要求は次回 SQL で実行されます。その間に内部的なオペレーションが実行された場合，-e オプションで停止した後も SQL トレースに情報が出力されることがあります。内部的なオペレーションについては，「DAT 形式ファイルのレコード形式」のオペレーションコードを参照してください。

(d)　-l PDUAPREPLVL 指定値

このオプションは，SQL トレース情報以外に，UAP 統計レポート情報も出力する場合に指定します。UAP 統計レポートの出力情報（クライアント環境定義 PDUAPREPLVL の指定値）を指定します。指定値，情報の出力先などについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

プロセス間メモリ通信機能を使用している場合，このオプションを指定しても無視されます。このとき，プロセス間メモリ通信中のメッセージが出力されます。

-e オプションと同時に指定した場合，このオプションの指定は無効となります。

(e)　-m ?パラメタ及び検索データの最大データ長　～＜符号なし整数＞((4～32008))

SQL トレースに出力する?パラメタ情報，及び検索データの最大データ長を，バイト単位で指定します。

このオプションは，クライアント環境定義 PDPRMTRC に INOUT を指定した場合と同じ機能になります。

-e オプションと同時に指定した場合，このオプションの指定は無効となります。

(f)　-n 出力オペレーションコード数　～＜符号なし整数＞((0～10000))《0》

SQL トレースファイルに出力するオペレーションコード数を指定します。指定したオペレーションコード数分情報を出力した場合，SQL トレース取得は停止します。

このオプションを省略した場合，-e オプション指定の pdclttrc コマンドを実行するまで，又は DISCONNECT されるまで SQL トレースを出力します。

-e オプションと同時に指定した場合，このオプションの指定は無効となります。

(g) -o SQL トレースファイルのサイズ ～<符号なし整数>((0，4096～2000000000)) 《32768》

SQL トレースファイルのサイズをバイト単位で指定します。

0 を指定した場合，ファイルの最大サイズとなります。

-e オプションと同時に指定した場合，このオプションの指定は無効となります。

(5) 規則

1. pdclttrc コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdclttrc コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdclttrc コマンドは，接続しているクライアントのバージョンが 07-01 以降の場合に，SQL トレースを取得します。
4. SQL トレースの取得は，pdclttrc コマンド実行時に処理中の SQL から取得を開始します。ただし，SQL トレース取得開始時に処理中の SQL については，次の情報は出力されません。
   - SQL 開始時刻
   - クライアント環境定義 PDSQLEXECTIME=YES 指定時に出力される SQL 実行時間
   - UAP 統計レポートに出力される SQL 単位情報の SQL 実行時間と差分情報
5. SQL トレースファイル及び UAP 統計レポートの出力先は，-l オプション指定，クライアント環境定義 PDCLTPATH，及び PDREPPATH の指定で決まります。

   SQL トレースファイルの出力先を次に示します。ただし，クライアント側で Type4 JDBC ドライバを使用し pdtrcmgr コマンドによる取得中にこのコマンドを実行した場合の出力先は，クライアント環境定義 PDTRCPATH の指定先になります。

PDCLTPATH の指定	PDREPPATH の指定	SQL トレースファイルの出力先	
		-l オプション指定なし	-l オプション指定あり
なし	なし	カレントディレクトリ	カレントディレクトリ
あり	なし	PDCLTPATH の指定ディレクトリ	PDCLTPATH の指定ディレクトリ
なし	あり	カレントディレクトリ	PDREPPATH の指定ディレクトリ
あり	あり	PDCLTPATH の指定ディレクトリ	

6. -l オプションを指定した場合，UAP 単位の情報はコマンド開始からの情報であるため，コマンド実行以前の情報については出力されません。
7. このコマンド実行時のファイル名称については，マニュアル「HiRDB UAP 開発ガイド」の「SQL トレース機能」を参照してください。
8. クライアント側で既に SQL トレースが取得されているときにこのコマンドを実行した場合の動作を次に示します。

   クライアントライブラリを使用している場合

   このコマンドを実行しても無効となります。

   Type4 JDBC ドライバを使用している場合

   SQL トレース取得をやめ，このコマンドによる SQL トレースの取得を開始します。
9. pdclttrc コマンド実行中に再度 pdclttrc コマンドを実行した場合，現在 SQL トレースを取得しているプロセスについては，取得中である旨のメッセージが出力されます。
10. このコマンドで取得する SQL トレースの出力形式は，クライアント環境変数 PDSQLTRCFMT の指定値に従います。PDSQLTRCFMT 省略時は出力形式 2 で出力します。

(6) 注意事項

1. 通信エラーや DISCONNECT などで接続が切断された場合，又は-n オプションで指定したオペレーションコード数分 SQL トレースを出力した場合，自動的に SQL トレースの取得を停止します。
2. クライアント環境定義 PDHSICOPTIONS で UAPREP 又は ALL を指定している場合，pdclttrc コマンドを実行しないでください。PDHSICOPTIONS の詳細は，マニュアル「HiRDB UAP 開発ガイド」の「クライアント環境定義（環境変数の設定）」を参照してください。
3. pdclttrc コマンドのリターンコードを次に示します。

   0：正常終了

   8：エラー終了
4. -p オプションを指定しないで実行した pdclttrc コマンドがエラー終了となった場合，一部のプロセスについてだけ SQL トレース取得状態となることがあります。エラーとなった原因を対策してから再実行するか，-e オプションを指定して SQL トレースの取得を停止してください。

2.22　pdcmt（トランザクションのコミット）

2.22.1　pdcmt の形式と規則

(1)　機能

pdcmt コマンドは，グローバルトランザクションを構成している各トランザクションブランチが，何らかの要因（通信障害など）でトランザクションを決着できない場合に，強制的にトランザクションをコミットするときに使用します。この場合，ほかのトランザクションとの不整合を発生させないために，グローバルトランザクション内のほかのトランザクションもコミットしなければなりません。

pdcmt コマンドは，「pdls -d trn」コマンドを実行して表示された情報中のトランザクション第１状態が READY でトランザクション第２状態が p のトランザクションを，強制的にコミットします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdcmt 〔 {-x ホスト名 | -u ユニット識別子} 〕 〔-s サーバ名〕
        {-A | -t トランザクション識別子}
```

(b)　HiRDB/パラレルサーバの場合

```
pdcmt {-x ホスト名 | -u ユニット識別子} 〔-s サーバ名〕
        {-A | -t トランザクション識別子}
```

(4)　オプション

(a)　{-x ホスト名｜-u ユニット識別子}

該当するトランザクションがあるホストのホスト名，又はユニットのユニット識別子を指定します。

-x ホスト名　～＜識別子＞((1～32))

該当するトランザクションがあるホストのホスト名を指定します。

-u ユニット識別子　～＜識別子＞((4 文字))

該当するトランザクションがあるユニットのユニット識別子を指定します。

(b)　-s サーバ名　～＜識別子＞((1～8))

ユニットに複数のサーバがある場合，サーバを特定するときにサーバ名を指定します。省略した場合，対象となるホスト内のすべてのサーバを対象とします。

(c)　-A

-s オプションで指定したサーバ内のトランザクションで，READY 状態で回復処理待ち状態のすべてのトランザクションをコミットする場合に指定します。

(d)　-t トランザクション識別子　～＜英数字＞((16 文字))

READY 状態で回復処理待ち状態のトランザクションをコミットする場合に，トランザクション識別子を指定します。

(5)　規則

1. pdcmt コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdcmt コマンドは，サーバマシンごとに実行してください。

(6)　注意事項

1. pdcmt コマンドの結果は，pdls コマンド（-d trn 指定），コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   1：異常終了（オプション指定不正，rsh 失敗など）

   コミットができていないトランザクションが残っていることがあります。

   メッセージに従って対処した後，再度コマンドを実行してください。
2. 回復不要 FES に接続して実行したトランザクションは，トランザクション第 1 状態が READY，トランザクション第 2 状態が p の場合でも，強制的にコミットできないことがあります。この場合，トランザクションを自動決着してください。未決着状態のトランザクションを決着する方法の詳細については，マニュアル「HiRDB システム運用ガイド」の「未決着状態のトランザクションを決着する方法」を参照してください。
3. 現在は停止しているクライアント製品からの要求で発生した未決着状態のトランザクションを強制的にコミットすると，この後でクライアント製品を開始したときに，HiRDB とクライアント製品との間でこのトランザクションの同期が取れないことがあります。

2.23 pdconfchk（システム定義のチェック）

2.23.1 pdconfchk の形式と規則

(1) 機能

HiRDB を開始するために必要な，システム定義の内容をチェックします。

システム定義が誤っている場合は，誤りの箇所及び詳細を表示します。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdconfchk 〔-d サブディレクトリ名〕 〔-n〕
```

(4) オプション

(a) -d サブディレクトリ名 ～＜パス名＞

チェック対象の HiRDB システム定義ファイルを格納したサブディレクトリを，%PDDIR%¥conf 又は%PDCONFPATH%からの相対パス名で指定します。%PDDIR%¥conf とは，ユニット制御情報定義ファイルを格納しているディレクトリを意味します。%PDCONFPATH%とは，それ以外の HiRDB システム定義ファイルを格納しているディレクトリを意味します。

チェックする HiRDB システム定義ファイルを格納する場所を，次に示します。

HiRDB システム定義ファイルの種類	-d オプションあり	-d オプションなし
ユニット制御情報定義ファイル	%PDDIR%¥conf¥サブディレクトリ¥	%PDDIR%¥conf¥
上記以外	%PDCONFPATH%¥サブディレクトリ¥	%PDCONFPATH%¥

＜適用基準＞

運用している環境の HiRDB システム定義を変更する前に，変更した HiRDB システム定義をサブディレクトリ下に格納してチェックできます。これによって，変更した HiRDB システム定義を，事前にチェックできます。

＜規則＞

1. 指定するサブディレクトリは，事前に作成しておいてください。

2. サブディレクトリ下には，変更した HiRDB システム定義ファイルだけでなく，すべての HiRDB システム定義ファイルを格納してください。
3. HiRDB/シングルサーバの場合，%PDDIR%¥conf と%PDCONFPATH%には，同じパス名が設定されます。
4. HiRDB/パラレルサーバの場合，%PDDIR%¥conf と%PDCONFPATH%が指すパス名が異なることがあります（例えば，HiRDB システム定義ファイルを共用化した場合など）。その場合，両方のパスに同一名称でサブディレクトリを作成してください。

   また，ユニットごとに%PDDIR%¥conf，%PDCONFPATH%の設定が異なる場合は，ユニットごとに%PDDIR%¥conf 下と%PDCONFPATH%下にサブディレクトリを作成し，HiRDB システム定義ファイルを格納してください。
5. %PDDIR%¥conf¥サブディレクトリ，及び%PDCONFPATH%¥サブディレクトリは，最大 220 バイトです。名称が 220 バイトを超えないようにしてください。
6. pdchgconf コマンドで変更するシステム定義をチェックする場合は，%PDDIR%¥conf¥chgconf に変更後のシステム定義ファイルを格納し，pdconfchk コマンド実行時に-d chgconf を指定してください。
7. 系切り替え機能を使用している場合，現用系の定義をチェックするため，サブディレクトリは現用系に作成してください。また，すべての実行系ユニットを現用系に切り替えてから pdconfchk コマンドを実行してください。なお，システムマネジャがあるユニットがオフラインのときは，現用系のシステムマネジャがあるユニットで pdconfchk -d コマンドを実行してください。
8. HiRDB/シングルサーバの場合，サブディレクトリ下の HiRDB システム定義ファイルのオペランド指定値は，%PDDIR%¥conf 又は%PDCONFPATH%下のオペランド指定値と一致させる必要があります。
   - pdunit オペランドの-x オプションに指定するホスト名
   - pd_hostname オペランドに指定するホスト名
9. HiRDB/パラレルサーバの場合，サブディレクトリ下の HiRDB システム定義ファイルのオペランド指定値は，%PDDIR%¥conf 又は%PDCONFPATH%下のオペランド指定値と一致させる必要があります。
   - システムマネジャがあるユニットの pdunit オペランドの-x オプションに指定するホスト名
   - システムマネジャがあるユニットの pd_hostname オペランドに指定するホスト名

(b) -n

システムファイルのチェックと，アクセス権限のチェックをスキップしたい場合に指定します。pdfmkfs コマンドや pdloginit コマンドでシステムファイルを作成していないときなどに利用します。

(5) チェックする内容

- **文法のチェック**

文法チェックの結果は標準出力又は標準エラー出力に出力されます。

- **システムファイルのチェック**

システムログファイル，シンクポイントダンプファイル，及びステータスファイルの有無をチェックします。

ただし，pdconfchk コマンドで-n オプションを指定した場合はファイルをチェックしません。

- **アクセス権限のチェック**

次のことをチェックします。

- HiRDB 管理者が HiRDB ファイルをアクセスできるかどうか
  ただし，pdconfchk コマンドで-n オプションを指定した場合はアクセス権限をチェックしません。
- システム定義の pd_tmp_directory オペランドで指定したディレクトリの有無，及び HiRDB 管理者がアクセスできるかどうか
  ただし，pd_tmp_directory オペランドを省略した場合はチェックしません。

- **システムファイルの重複指定チェック**

同一ユニット，又は同一サーバ定義内に指定した以下の値が重複していないかどうかをチェックします。

- システムログファイルのシステムログファイルパス名
- シンクポイントダンプファイルのシンクポイントダンプファイルパス名
- ステータスファイルの論理ファイル名，及びサーバステータスファイルパス名

- **ホストのチェック**

ホスト名が HOSTS ファイルに記述されているかどうかをチェックします。

- **サーバマシン間のチェック（HiRDB/パラレルサーバの場合）**

システムマネジャのサーバマシンを基準に，サーバマシン間のチェックをします。

pdconfchk コマンドでチェックできるシステム定義のオペランドについては，マニュアル「HiRDB システム定義」の「pdconfchk コマンドでチェックできるオペランド」を参照してください。

(6) 規則

1. pdconfchk コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdconfchk コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(7) 注意事項

1. 正常終了時は，KFPS05007-I return code=0 を出力し終了します。正常終了以外の場合には，KFPS05007-I return code=4（警告），又は KFPS05007-I return code=8（エラー）の終了メッセージの前に原因となるメッセージを出力します。

2. サーバマシン間のチェックで，コメント部分（#以降の部分），及びコメントだけの行はエラー検知の対象としません。

3. サーバマシン間のチェックでエラーがあった場合，システムマネジャがあるサーバマシンのシステム定義の内容も見直してください。システムマネジャがあるサーバマシンのシステム定義の内容が誤っていることもあります。

4. サーバマシン間のチェックで，定義文の前方及び文中にあるタブとスペースはチェック対象となります。定義文の後方にあるタブとスペースはチェック対象外となります。

5. サーバマシン間のチェックで，システムマネジャとシステムマネジャ以外の定義の順番が同じでない場合，順番の異なる定義を誤りとします。

```
（例）
システムマネジャの定義        システムマネジャ以外の定義
    A                             B
    B                             A
```

   この場合，システムマネジャの定義 B とシステムマネジャ以外の定義 B を，Different definition として表示します。

6. pdconfchk コマンドで KFPY01012-E メッセージが出力された場合，pdconfchk コマンドは KFPS05007-I return code=0 で終了することがあります。この場合，エラーメッセージの原因を取り除いてください。

7. pdconfchk コマンドを実行して KFPS05062-W メッセージが出力された場合でも，pdconfchk コマンドは KFPS05007-I（return code=0）で終了します。

8. pdconfchk コマンドが正常に動作した場合でも，KFPO00107-E メッセージが出力されることがあります。pdconfchk コマンドでの定義チェックの結果は KFPS05007-I メッセージで確認してください。

9. 系切り替え機能を使用している場合，指定しているオプションによって pdconfchk コマンド実行時にチェック対象となる系が異なります。

   -d オプション以外を指定している場合

   実行系がチェック対象となります。ただし，HiRDB システムが停止している又は定義エラーなどによって実行系かどうかを判別できないときは現用系がチェック対象になります。

   -d オプションを指定している場合

   現用系がチェック対象となります。

(8) 出力形式

```
＜システムファイルの有無チェック，アクセス権限のチェック＞
 [aa...a]
  bbbb    cc...c    dd...d    ee...e
＜システムファイルの重複指定チェック＞
 [aa...a]
  bbbb    cc...c    dd...d    ee...e
  bbbb    cc...c    dd...d    ee...e
＜ホストのチェック＞
 [aa...a]
  bbbb    cc...c    dd...d    ff...f
＜サーバマシン間のチェック(HiRDB/パラレルサーバの場合)＞
```

```
 [aa...a]
 <gggg:hhhh>
  gggg;  cc...c;  line = iii <- hhhh;  cc...c;  nothing
  gggg;  cc...c;  nothing    -> hhhh;  cc...c;  line = iii
```

[説明]

aa...a：

チェックした項目のエラーの内容

File not found：

ファイルがありません。次のどちらかの原因が考えられます。

- パス名に使用できない文字を指定しています。
- HiRDB ファイル名の長さが制限（30 文字）を超えました。

Directory not found：

ディレクトリがありません。次のどちらかの原因が考えられます。

- パス名に使用できない文字を指定しています。
- システム定義の pd_tmp_directory オペランドに指定したディレクトリがありません。

File invalid：

ファイルが不正です。

Directory invalid：

ディレクトリが不正です。ドライブ直下のディレクトリ（例：C:¥）を指定している場合は，別のディレクトリを指定してください。

No Permission：

権限がありません。

Duplicate filename：

ファイル名が重複しています。

Hosts name not matched：

ホスト名が一致しません。

Different definition：

システムマネジャとシステムマネジャ以外のユニットで異なる定義があります。

bbbb：

ユニット識別子

cc...c：

システム定義のファイル名

dd...d：

オペランドの指定内容

ee...e：

dd...d のオペランドで指定しているファイル名（絶対パス名）

ff...f：

dd...d が pdstart，pdunit，又は pd_hostname の場合に，定義されているホスト名

gggg：

システムマネジャがあるユニットのユニット識別子

hhhh：

システムマネジャ以外のユニットのユニット識別子

iii：

システム定義のファイルの行番号

＜出力例＞

```
[File not found]
 unt1   pdutsys   pd_syssts_file_name_1              c:¥hirdb¥rdsys011¥ut1sts1a ....1
[File invalid]
 unt1   pdutsys   pd_syssts_file_name_3              c:¥hirdb¥yyy¥pddir ............2
[No permission]
 unt1   bes1      pd_sts_file_name_1                 c:¥hirdb¥rdsys011¥bes1sts1a ...3
[Duplicate filename]
 unt1   fes1      pdlogadpf -d sys -g fes1log1 -a    c:¥hirdb¥rrdsys011¥fes1log1 ...4
 unt1   fes1      pdlogadpf -d sys -g fes1log1 -b    c:¥hirdb¥rrdsys011¥fes1log1 ...4
 unt1   bes1      pdlogadpf -d spd -g bes1spd1 -a    c:¥hirdb¥rrdsys011¥fes1log1 ...4
[Hosts name not matched]
 unt1   pdsys     pdstart -x                         dcm3500 .......................5
[Different definition]
<unt1:unt2>
 unt1   pdsys    line = 2   -> unt2   pdsys   nothing ..............................6
 unt1   pdsys    nothing    <- unt2   pdsys   line = 5 ..............................7
```

［説明］

1. unt1 の pdutsys にある pd_syssts_file_name_1 オペランドに指定したファイルが見付かりません。
2. unt1 の pdutsys にある pd_syssts_file_name_3 オペランドに指定したファイルが不正です。
3. unt1 の bes1 にある pd_sts_file_name_1 オペランドに指定したファイルにアクセス権限がありません。
4. unt1 の fes1 にある pdlogadpf -d sys -g fes1log1 オペランドの-a に指定しているファイル名と，次のファイル名が重複しています。
   - unt1 の fes1 にある pdlogadpf -d sys -g fes1log1 オペランドの-b に指定しているファイル名
   - unt1 の bes1 にある pdlogadpf -d spd -g bes1spd1 オペランドの-a に指定しているファイル名
5. unt1 のホスト名と，unt1 の pdsys にある pdstart -x dcm3500 オペランドで指定しているホスト名の設定が異なっています。
6. unt1 の pdsys の 2 行目の定義が，unt2 の pdsys にはありません。
7. unt2 の pdsys の 5 行目の定義が，unt1 の pdsys にはありません。

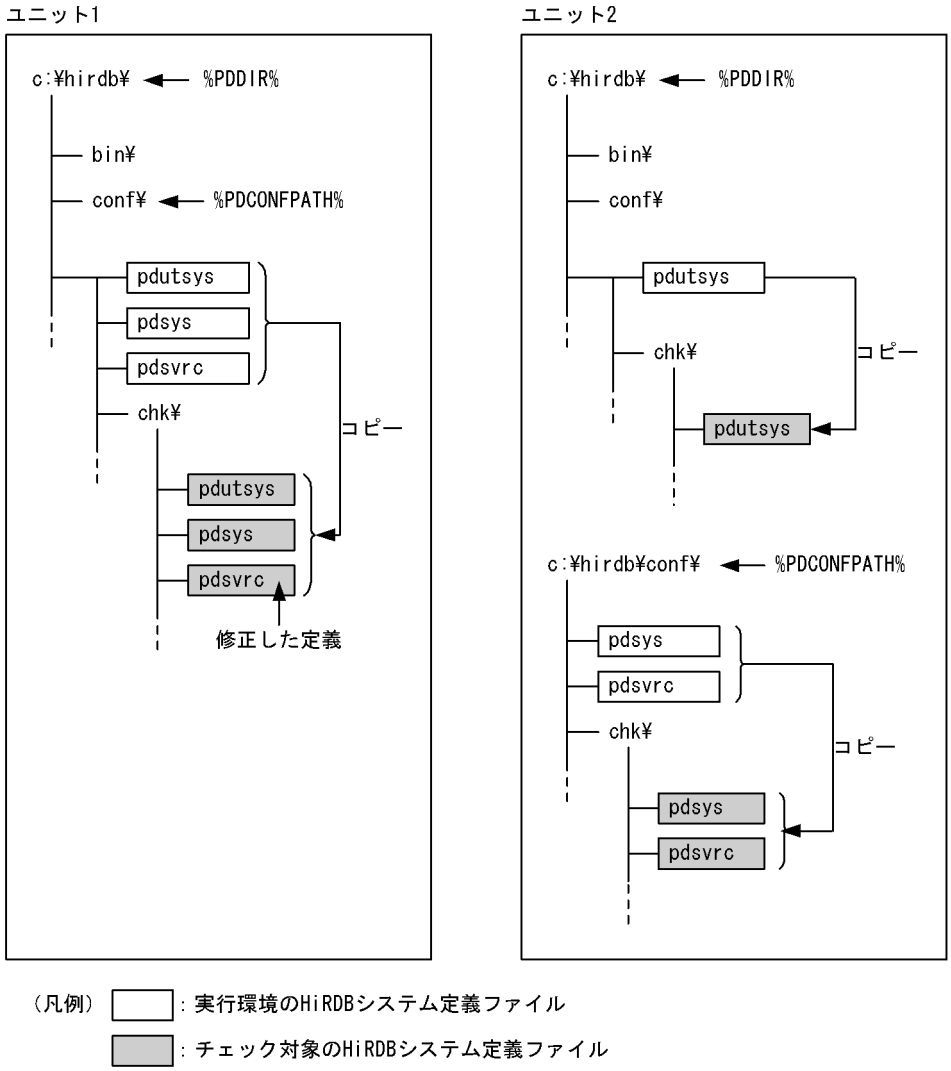
(9) 使用例

運用中の HiRDB/パラレルサーバ（ユニット 1，ユニット 2 の構成）のシステム定義を変更します。変更する HiRDB システム定義ファイルはユニット 1 の pdsvrc とします。

1. ユニット 1 とユニット 2 の%PDDIR%¥conf 下と%PDCONFPATH%下にサブディレクトリ「chk」を作成します。
2. 各ユニットで，%PDDIR%¥conf 下の pdutsys を%PDDIR%¥conf¥chk¥にコピーします。
3. 各ユニットで，%PDCONFPATH%下の HiRDB システム定義ファイルを%PDCONFPATH%¥chk¥にコピーします。
4. ユニット 1 の%PDCONFPATH%¥chk¥pdsvrc の定義内容を変更します。
5. 「pdconfchk -d chk」を実行します。※
6. 内容に問題がなければ，pdstop コマンドで HiRDB を正常終了又は計画停止します。ただし，計画停止の場合は，変更できない定義があるため注意してください。
7. pdlogunld コマンドで，「アンロード待ち状態」のシステムログファイルをアンロードします。
8. 4.で変更した HiRDB システム定義ファイルを%PDCONFPATH%下にコピーします。
9. pdstart コマンドで HiRDB を正常開始します。

注※

「pdconfchk -d chk」実行時に，チェックする HiRDB システム定義ファイルの場所は次のようになります。


ユニット1
c:¥hirdb¥ ← %PDDIR%
bin¥
conf¥ ← %PDCONFPATH%
pdutsys
pdsys
pdsvrc
chk¥
コピー
pdutsys
pdsys
pdsvrc
修正した定義
ユニット2
c:¥hirdb¥ ← %PDDIR%
bin¥
conf¥
pdutsys
chk¥
コピー
pdutsys
c:¥hirdb¥conf¥ ← %PDCONFPATH%
pdsys
pdsvrc
chk¥
コピー
pdsys
pdsvrc
（凡例）
：実行環境のHiRDBシステム定義ファイル
：チェック対象のHiRDBシステム定義ファイル

2.24　pdcspool（トラブルシュート情報及び作業用一時ファイルの削除）

2.24.1　pdcspool の形式と規則

(1)　機能

HiRDB が作成した，次のファイルを削除します。

- スプール（%PDDIR%¥spool ディレクトリ）下のトラブルシュート情報
- 作業用一時ファイル格納ディレクトリ（%PDDIR%¥tmp ディレクトリ）下の作業用一時ファイル
- システム定義の pd_tmp_directory オペランドに指定したディレクトリ下のファイル

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdcspool 〔-i〕〔-d 日数〕〔-k 種別〕
```

(4)　オプション

(a)　-i

トラブルシュート情報が格納された各ファイルを削除するときに，標準出力にプロンプトを返し，削除するかどうかを確認します（作業用一時ファイルの場合はプロンプトを返しません）。プロンプトに y を応答すると，該当ファイルが削除されます。省略した場合は，プロンプトを出力しないで該当ファイルを削除し，削除したファイル名が標準出力に出力されます。

(b)　-d 日数　～＜符号なし整数＞((0～24855))≪1≫

pdcspool コマンドを実行した時刻から換算して「24 時間×指定した日数」以前に作成されたファイルを削除対象とします。例えば，日数に 10 を指定した場合，240 時間前の時点より，以前に作成されたファイルが削除対象となります。

0 を指定した場合は，スプール下のトラブルシュート情報，及び作業用一時ファイル格納ディレクトリ下の作業用一時ファイルをすべて削除対象とします。ただし，0 を指定した場合でも，次に示すファイルについては，ファイルの更新日付がこのコマンドの起動時刻から 3600 秒以内のものは削除対象となりません。ファイルが残っている場合は，3600 秒後に再度コマンドを実行してください。

- %PDDIR%¥tmp ディレクトリ下の作業用一時ファイル

省略した場合は，24 時間以前に作成されたスプール下のトラブルシュート情報，及び作業用一時ファイル格納ディレクトリ下の作業用一時ファイルが削除対象となります。

pdcopy，pdrstr など，処理が長時間になるコマンドを定期的に実行するような運用の場合，実行完了に要する時間以上の値を指定してください。

(c) -k 種別

削除するトラブルシュート情報の範囲を指定します。

省略した場合，アクセスパス情報以外のトラブルシュート情報が削除されます。

all：

アクセスパス情報を含め，すべてのトラブルシュート情報及び作業用一時ファイルが削除されます。ただし，システム定義の pd_tmp_directory オペランドの指定を省略した場合，削除されません。

dump：

HiRDB が内部的に取得するファイル（デッドロック・タイムアウト情報ファイル，及びアクセスパス情報ファイルを除くファイル）が削除されます。

(5) 規則

1. pdcspool コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdcspool コマンドは，各サーバマシンで実行してください。

(6) 注意事項

1. pdcspool コマンドの結果は，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：正常終了（削除対象となるファイルがありません）

   トラブルシュート情報及び作業用一時ファイルの削除を行いませんでした。このため，対処は必要ありません。

   12：異常終了（オプション指定不正など）

   トラブルシュート情報及び作業用一時ファイルの削除を行いませんでした。メッセージに従って対処した後，再度コマンドを実行してください。
2. HiRDB 管理者以外のユーザが実行したコマンド又はユティリティによって，出力されたトラブルシュート情報ファイルは削除されないことがあります。この場合，トラブルシュート情報ファイルの削除権限を持つユーザがファイルを削除してください。このとき，作業用一時ファイル格納ディレクトリ（%PDDIR%¥tmp ディレクトリ）下の次のディレクトリ及びファイルは削除しないでください。
   - home

- pdommenv
- ファイル名が"CMb"で始まるファイル

3. 環境変数 TMP が設定されている場合，作業用一時ファイルの出力先は%PDDIR%¥tmp ディレクトリではなく，環境変数に設定されたディレクトリになります。環境変数 TMP に設定されたディレクトリ下の作業用一時ファイルは pdcspool コマンドでの削除対象とはなりません。また，パイプファイル(FIFO)も削除しません。このため，これらのファイルは，エクスプローラなどで直接削除する必要があります。
4. pdcspool コマンドは，HiRDB が次のディレクトリ下に作成したすべてのファイルを削除の対象とします。

   そのため，該当ディレクトリ下にファイルを作成した場合，一緒に削除されます。削除対象ディレクトリの一覧を，次に示します。

ディレクトリ名		出力ファイルの内容	-k オプション			備考
上位ディレクトリ	下位ディレクトリ		all	dump	省略時	
%PDDIR%¥tmp（作業用一時ファイル格納ディレクトリ）	−	作業用一時ファイル	○	○	○	-d オプションに 0 を指定した場合，このコマンドの起動時刻から 3600 秒以上経過した更新日付のファイルが削除されます。
ユニット制御情報定義の pd_tmp_directory オペランドに指定したディレクトリ	−	作業用一時ファイル	○	○	○	
%PDDIR%¥spool（トラブルシュート情報）	pdsqldump	アクセスパス情報	○	×	×	クライアント環境定義 PDVWOPTMODE に 1 以上を指定して，SQL 文を実行した場合に出力されます。
	pdlckinf	デッドロック・タイムアウト情報	○	×	○	排他制御でエラーが発生したときに出力されます。
		排他資源管理テーブル情報	○	×	○	排他資源管理テーブル不足が発生したときに出力されます。
		デッドロック SQL 情報	○	×	○	デッドロックが発生したときに出力されます。
	save	退避コアファイルなど	○	○	○	プロセスが異常終了したときに出力されます。
	pdshmdump	共用メモリダンプファイル	○	○	○	プロセス又はユニットが異常終了したときに出力されます。
	pdsysdump(1,2)	簡易ダンプファイル	○	○	○	なし。
	pdsdsdump(1,2)		○	○	○	HiRDB/シングルサーバの場合，出力されます。

ディレクトリ名		出力ファイルの内容	-k オプション			備考
上位ディレクトリ	下位ディレクトリ		all	dump	省略時	
	pdfesdump(1,2) pddicdump(1,2) pdbesdump(1,2)		○	○	○	HiRDB/パラレルサーバの場合，出力されます。
	pdjnlinf	システムログファイルの状態情報ファイル	○	×	○	pdsnap.で始まるファイルだけ削除されます。
	tmp	作業用一時ファイル	○	○	○	なし。

（凡例）

○：削除されます。

×：削除されません。

－：該当しません。

ただし，次のディレクトリ下のディレクトリ，又はファイルは削除されません。

上位ディレクトリ	削除対象外のディレクトリ名，又はファイル名	出力ファイルの説明
%PDDIR%¥spool （トラブルシュート情報）	パイプファイル	任意のファイル名称を持つすべてのパイプファイル
%PDDIR%¥tmp （作業用一時ファイル格納ディレクトリ）	パイプファイル	任意のファイル名称を持つすべてのパイプファイル
	home	HiRDB が起動するプロセスのカレントワーキングディレクトリ
	pdommenv	共有メモリ情報ファイル
	ファイル名称が"CMb"で始まるファイル	pdbufls コマンドの差分情報ファイル
ユニット制御情報定義の pd_tmp_directory オペランドに指定したディレクトリ	パイプファイル	任意のファイル名称を持つすべてのパイプファイル
	home	HiRDB が起動するプロセスのカレントワーキングディレクトリ
	pdommenv	共有メモリ情報ファイル
	ファイル名称が"CMb"で始まるファイル	pdbufls コマンドの差分情報ファイル

2.25　pddbfrz（ユーザ LOB 用 RD エリアの満杯 HiRDB ファイルの更新凍結）

2.25.1　pddbfrz の形式と規則

(1)　機能

ユーザ LOB 用 RD エリアを構成する HiRDB ファイルで，データが満杯（ファイルの最終ページまで使用中）となった HiRDB ファイルを，更新及び削除できない状態にします。これを**更新凍結状態**といいます。更新凍結状態となった HiRDB ファイルに格納されている LOB データは，更新及び削除ができません。

このコマンドは，pdcopy 以外の方法でバックアップを取得する場合に使用してください。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pddbfrz 〔-d〕 -r {RDエリア名〔, RDエリア名〕… | ALL} 〔-W 実行監視時間〕
```

(4)　オプション

(a)　-d

HiRDB ファイルの更新凍結状態を解除する場合に指定します。

省略した場合は，データが満杯となっている HiRDB ファイルを更新凍結状態にします。ただし，RD エリアを構成する先頭の HiRDB ファイルは更新凍結状態にできません。

(b)　-r {RD エリア名〔，RD エリア名〕... | ALL}

RD エリア名　～＜識別子＞((1～30))

更新凍結状態にする RD エリア名，又は更新凍結状態を解除する RD エリア名を指定します。RD エリア名には，ユーザ LOB 用 RD エリアだけ指定できます。

ALL

すべてのユーザ LOB 用 RD エリアを更新凍結状態にします。-d オプション指定時は，すべてのユーザ LOB 用 RD エリアの更新凍結状態を解除します。

＜規則＞

1. RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。

2. ユーザ LOB 用 RD エリア以外の RD エリア，プラグインインデクスが格納されているユーザ LOB 用 RD エリア，及び未使用のユーザ LOB 用 RD エリアを指定した場合はエラーとなります。

(c)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pddbfrz コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pddbfrz コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pddbfrz コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pddbfrz コマンドを実行する場合，ユーザ LOB 用 RD エリアが次のどちらかの状態でなければなりません。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，かつ pdhold コマンドによる閉塞状態
4. pddbfrz コマンドを実行すると，RD エリアに対して排他モードで排他を掛けます。そのため，ほかのトランザクションがアクセスしている RD エリアに対して pddbfrz コマンドを実行すると，そのトランザクションが終了するまで pddbfrz コマンドは待ち状態となります。
5. RD エリアを構成する先頭の HiRDB ファイルには，HiRDB の管理レコードがあるため，pddbfrz コマンドを実行しても更新凍結状態にはなりません。
6. 更新凍結状態にできる HiRDB ファイルは，コマンド実行時にデータが満杯となっているファイルだけです。そのため，コマンド実行時に満杯ではなかった HiRDB ファイルがその後のデータ登録などで満杯になったとしても，更新凍結状態に切り替わることはありません。

(6)　注意事項

1. pddbfrz コマンドの結果は，pddbst で確認できます。
2. 更新凍結状態の HiRDB ファイルのデータを更新したい場合，又は pdrorg の再編成などで RD エリアを更新する必要がある場合，pddbfrz -d コマンドで更新凍結状態を解除してください。なお，RD エリアの更新後は，RD エリアのすべての HiRDB ファイルのバックアップを取得し直す必要があります。
3. pddbfrz コマンドのリターンコードと基本的な対処方法を次に示します。

   0 以外の場合，一部のユーザ LOB 用 RD エリアの HiRDB ファイルが更新凍結状態になっていることがあります。その状態では業務を継続できないおそれがあるため，pddbfrz コマンドに指定した RD エリアを pddbfrz コマンド（-d オプション指定）で更新凍結状態を解除してください。

0：正常終了

4：警告終了

一部の RD エリアの処理がエラー終了しました。

8：異常終了

一部の RD エリアの処理がエラー終了していることがあります。

12：異常終了

エラーメッセージを表示できない事象が発生しました。

一部の RD エリアの処理がエラー終了していることがあります。シングルサーバ又はディクショナリサーバがあるホストのイベントログ中のエラーメッセージを参照して，エラーの要因を取り除いた後，再度コマンドを実行してください。イベントログ中にエラーメッセージが出力されていない場合は保守員に連絡してください。

2.26　pddbls（RD エリアの状態表示）

2.26.1　pddbls の形式と規則

(1)　機能

RD エリアの状態を表示します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
  pddbls  {-r RDエリア名〔, RDエリア名〕…|-r ALL}
                〔 {-l|-a} 〕  〔-b〕  〔-x 〔-y〕〕〔-M〕〔-D〕〔-T〕〔-P〕〔-W 実行監視時
間〕
```

(b)　HiRDB/パラレルサーバの場合

```
  pddbls  {-r RDエリア名〔, RDエリア名〕…|-r ALL
                |-s サーバ名〔, サーバ名〕…}
                 〔 {-l|-a} 〕  〔-b〕  〔-m〕  〔-x 〔-y〕〕〔-M〕〔-D〕〔-T〕〔-P〕〔-W 実
行監視時間〕
```

(4)　オプション

(a)　-r RD エリア名〔，RD エリア名〕...　～＜識別子＞((1～30))

表示する RD エリア名を指定します。

(b)　-r ALL

すべての RD エリアに関する情報を表示します。

＜規則＞

RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。

(c) -s サーバ名〔,サーバ名〕... ～<識別子>((1～8))

サーバ内のすべての RD エリアに関する情報を表示したい場合，表示するサーバ名を指定します。指定したサーバのすべての RD エリアに関する情報を表示します。

指定できるサーバ数は最大 128 個です。129 個以上指定すると，129 個目以降のサーバは無視されます。

(d) -{l | a}

-l：

RD エリアの状態を表示します。

-a：

RD エリアに関するすべての情報を表示します。ただし，次に示す情報は表示されません。

- 共用 RD エリアに関する情報
- インメモリデータバッファに関する情報

共用 RD エリアに関する情報を表示する場合は-m オプションを，インメモリデータバッファに関する情報を表示する場合は-M 及び-D オプションを指定してください。

また，-m オプションを指定した場合，全バックエンドサーバの共用 RD エリア情報が表示されます。ただし，共用 RD エリア内のセグメント使用情報については，更新可能バックエンドサーバだけが表示されます。

(e) -b

閉塞状態の RD エリアの情報だけを表示します。サーバ名を指定した場合，そのサーバの閉塞状態の RD エリアについてだけ，情報を表示します。省略した場合，指定したすべての RD エリアの情報を表示します。

(f) -m

共用 RD エリアの情報を表示する場合に指定します。

-r オプションに共用 RD エリアを指定し，かつ-m オプションを指定した場合，全バックエンドサーバの該当する RD エリアの RD エリア情報と共用 RD エリア情報を表示します。

(g) -x

DAT 形式で情報を表示する場合に指定します。

(h) -y

-x オプション指定時に，ヘッダを付ける場合に指定します。

(i) -M

このオプションはインメモリデータ処理に関するオプションです。

-r オプションで指定したインメモリ RD エリアに割り当てられたインメモリデータバッファの状態を表示します。

(j) -D

このオプションはインメモリデータ処理に関するオプションです。

-r オプションで指定したインメモリ RD エリアについて，次に示す情報を表示します。

- インメモリ化した時刻
- インメモリデータバッファとの同期を取った時刻

(k) -T

一時表用 RD エリアの属性を表示する場合に指定します。

(l) -P

pdchpathf コマンドの実行時刻を表示します。DAT 形式では表示しません。

(m) -W 実行監視時間 ～<符号なし整数>((0～3600))

pddbls コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5) 規則

1. pddbls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pddbls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. RD エリア名に，ユーザ用 RD エリア又はユーザ LOB 用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，pdhold コマンドによる閉塞状態

(6) 注意事項

1. pddbls コマンドのリターンコードを次に示します。
   0：正常終了
   4：警告終了

8：異常終了

12：異常終了（エラーメッセージを表示できない事象が発生）

12 の場合，シングルサーバ又はディクショナリサーバがあるホストのイベントログ中のエラーメッセージを参照して，エラーの要因を取り除いた後，再度コマンドを実行してください。イベントログ中にエラーメッセージが出力されていない場合は保守員に連絡してください。

(7)　出力形式

(a)　-x 及び-y オプション省略時

```
STATE OF RDAREA
RDAREA ID     STATUS  [SERVER] TYPE
              OPNMODE [SHARED][MEMORY-STATUS]
aa...a bb...b cc...c  dd...d  eeee
              jj...j  ppp     qq...q
 [SEGMENT   ff...f/gg...g]
 [UNABLE TO DISPLAY CODE=hh]
    :    :   :
 [NOT FOUND ii...i=aa...a]
 [RDAREA NOT FOUND]
 [DB-STAY-TIME  rr...r]
 [DB-SYNC-TIME  ss...s]
 [RDAREA_FOR_TEMPORARY_TABLE = tt...t]
 [PATH CHANGE = vv...v]
```

[説明]

aa...a：

RD エリア名（30 文字以内）

bb...b：

RD エリア番号（10 進数 11 けた）

cc...c：

RD エリアの状態（17 文字以内）。RD エリアの状態を次に示します。

RD エリアの状態	意味	セグメント情報
CLOSE	クローズ	△
CLOSE HOLD	クローズ，障害閉塞中	－
CLOSE HOLD（INQ）	クローズ，参照可能閉塞中	△
CLOSE HOLD（CMD）	クローズ，コマンド閉塞中	－
CLOSE HOLD（BU）	クローズ，更新可能バックアップ閉塞中	○
CLOSE HOLD（BU I）	クローズ，参照可能バックアップ閉塞中	○
CLOSE HOLD（BU W）	クローズ，更新可能バックアップ閉塞中（WAIT モード）	○

RD エリアの状態	意味	セグメント情報
CLOSE HOLD（BU IW）	クローズ，参照可能バックアップ閉塞中（更新 WAIT モード）	○
CLOSE ACCEPT-HOLD	クローズ，pdhold コマンド受付状態	−
HOLD	障害閉塞中	−
HOLD（INQ）	参照可能閉塞中	○※
HOLD（CMD）	コマンド閉塞中	○※
HOLD（BU）	更新可能バックアップ閉塞中	○
HOLD（BU I）	参照可能バックアップ閉塞中	○
HOLD（BU W）	更新可能バックアップ閉塞中（WAIT モード）	○
HOLD（BU IW）	参照可能バックアップ閉塞中（更新 WAIT モード）	○
ACCEPT-HOLD	pdhold コマンド受け付け状態	○※
OPEN	オープン	○

（凡例）

○：表示します。

△：RD エリアのオープン契機が DEFER 又は SCHEDULE の場合に表示されます。

−：表示しません。

注※

クローズ処理中は，セグメント情報を表示しません。

dd...d：

HiRDB/パラレルサーバの場合，サーバ名（8 文字以内）

eeee：

RD エリアの種別

MAST：

マスタディレクトリ用 RD エリア

DDIR：

データディレクトリ用 RD エリア

DDIC：

データディクショナリ用 RD エリア

DLOB：

データディクショナリ LOB 用 RD エリア

USER：

ユーザ用 RD エリア

ULOB：

ユーザ LOB 用 RD エリア

LIST：

リスト用 RD エリア

RGST：

レジストリ用 RD エリア

RLOB：

レジストリ LOB 用 RD エリア

ff...f：

RD エリア内の未使用セグメント数（10 進数 11 けた）

LOB 用 RD エリアの場合，次のセグメントが未使用セグメント数に含まれますが，同期点に達するまで再使用はできません。同期点に達した後は再使用できるようになります。

- DELETE 文で削除された BLOB 型のデータが格納されていたセグメント
- UPDATE 文で更新された BLOB 型のデータについて，更新前のデータが格納されていたセグメント
- DROP PROCEDURE，DROP SCHEMA などの，手続き更新によって削除された BLOB 型のディクショナリデータが格納されていたセグメント

gg...g：

RD エリア内の全セグメント数（10 進数 11 けた）

hh ：

リターンコード。リターンコードを次に示します。

リターンコード	意味	対策方法
01	該当する RD エリアが，情報を表示できない状態です。RD エリアの状態については，cc...c を参照してください。	該当しません。
02	該当する RD エリアは，ほかのトランザクションで占有されているため，情報を表示できない状態です。	占有しているトランザクションの実行完了後に，再度実行してください。
03	該当する RD エリアには，グローバルバッファが割り当てられていません。	グローバルバッファを割り当てて，再度実行してください。
04	HiRDB ファイルアクセスで異常が発生しました。	対応するエラーメッセージが出力されるので，そのメッセージから対策方法を検討してください。
05	排他処理でエラーが発生しました。	
101	RD エリアの種別が不正です。	該当する RD エリアに対して誤った運用をしていないか確認してください。誤りがない場合には，保守員に連絡してください。

ii...i：

RD エリア指定の場合，「RDAREA」と表示されます（6 文字以内）。

サーバ指定の場合，「SERVER」と表示されます。

jj...j：

HiRDB ファイルシステム領域のオープン契機ユーザ指定値。システム共通定義，データベース初期設定ユティリティ，又はデータベース構成変更ユティリティで指定した HiRDB ファイルシステム領域のオープン契機（INITIAL，DEFER，又は SCHEDULE のどれか）を表示します。なお，系切り替え機能を使用している場合は，実行系でコマンドを実行したときにオープン契機を表示します。ただし，次のどれかの系切り替え機能を使用している場合，系切り替えが発生するとオープン契機が変更されることがあります。

- 高速系切り替え機能
- 1：1 スタンバイレス型系切り替え機能
- 影響分散スタンバイレス型系切り替え機能

詳細は，マニュアル「HiRDB システム運用ガイド」を参照してください。

ppp：

共用 RD エリア情報

SUP：共用 RD エリア（更新可能バックエンドサーバ）

SRD：共用 RD エリア（参照専用バックエンドサーバ）

NON：共用 RD エリアではない RD エリア

-m オプションを省略した場合，又は HiRDB/シングルサーバの場合は，ヘッダ，及び共用 RD エリア情報は表示されません。

qq...q：

インメモリデータバッファの状態

表示される情報	説明	
Y (SYNC)	この RD エリアはインメモリ RD エリアになっています。	インメモリデータバッファとインメモリ RD エリアの同期が取れている状態（**DB 同期状態**）です。
Y (ASYNC)		インメモリデータバッファとインメモリ RD エリアの同期が取れていない状態（**DB 非同期状態**）です。
Y (OBST-MEM)		インメモリデータバッファに障害が発生している状態（**バッファ障害状態**）です。
Y (OBST-DB)		インメモリ RD エリアに障害が発生している状態（**RD エリア障害状態**）です。
Y (OBST-ALL)		インメモリデータバッファと RD エリアの両方に障害が発生している状態（**バッファ障害状態**かつ **RD エリア障害状態**）です。
N	この RD エリアはインメモリ RD エリアではありません。	

-M オプションを省略した場合は，この情報は表示されません（ヘッダも表示されません）。

rr...r：

RD エリアをインメモリ化した時刻

最後にインメモリ化した時刻を，年（西暦 4 けた）/月/日 時：分：秒の形式で表示します。インメモリ RD エリアがない場合，又は時刻を表示できない場合は，****/**/** **:**:**が表示されます。

-D オプションを省略した場合は，この情報は表示されません（ヘッダも表示されません）。

ss...s：

インメモリデータバッファとインメモリ RD エリアの同期を取った時刻

最後に同期を取った時刻を，年（西暦 4 けた）/月/日 時：分：秒の形式で表示します。一度も同期を取っていない場合，又は時刻を表示できない場合は，****/**/** **:**:**が表示されます。

-D オプションを省略した場合は，この情報は表示されません（ヘッダも表示されません）。

tt...t：

一時表用 RD エリアの属性

OCCUPIED：特定 SQL セッション占有属性の一時表用 RD エリア

SHARED：SQL セッション間共有属性の一時表用 RD エリア

NONE：上記以外

uu...u：

システムで使用する情報

注

- 指定した RD エリアがない場合は，「NOT FOUND RDAREA=」が表示されます。
- 指定したサーバがない場合は，「NOT FOUND SERVER=」が表示されます。
- -b オプションを指定したときに閉塞状態の RD エリアがない場合，又は-s オプションで指定したサーバに RD エリアがない場合は，「RDAREA NOT FOUND」が表示されます。
- -a を指定したときに情報が表示されない場合は，「UNABLE TO DISPLAY CODE=hh」が表示されます。

vv...v：

pdchpathf コマンドの実行時刻

YYYY/MM/DD hh:mm:ss の形式で表示します。

pdchpathf コマンドを実行していない場合，又は pdchpathf コマンド実行後に pdchpathn コマンドを実行した場合，"NO"と表示します。

(b)　DAT 形式での出力形式（-x 及び-y オプション指定時）

```
”RDAREA”,”ID”,”STATUS”,”UNUSED_SEGMENT”,”TOTAL_SEGMENT”,”SERVER”,”TYPE”,”OPNMODE”,”ORIGINAL-
RDAREA”,
”ORIGINAL-ID”,”GENERATION-NUMBER”,”REPLICA-STATUS”,”REPLICA-COUNTER”,”SHARED”,”MEMORY-STATUS
”,
”DB-STAY-TIME”,”DB-SYNC-TIME”,”RDAREA_FOR_TEMPORARY_TABLE”,”DATAMODEL” [CR]
”aa...a”,bb...b,”cc...c”,ff...f,gg...g,”dd...d”,”eeee”,”jj...j”,””,
,,””,,”ppp”,”qq...q”,”rr...r”,”ss...s”,”tt...t”,”uu...u” [CR]
```

注

[CR]は，改行を表します。

[DAT 形式での出力規則]

1. DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。
2. 1 行に 1RD エリアの情報を表示します。
3. オプションの指定，サーバ種別などによって取得できない情報については，空要素として表示します。
4. -a オプション指定時にセグメント情報を取得できない状態の場合，RD エリア内の未使用セグメント数に-1，RD エリア内の全セグメント数にリターンコードを表示します。
5. DAT 形式で表示する場合，次のエラーメッセージは出力しません。

   [NOT FOUND ii...i=aa...a]

   [RDAREA NOT FOUND]

2.27　pddelfw.bat（Windows ファイアウォールの例外リスト削除）

2.27.1　pddelfw.bat の形式と規則

(1)　機能

Windows ファイアウォールの例外リストから HiRDB を削除します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pddelfw.bat
```

(4)　オプション

なし。

(5)　規則

pddelfw.bat コマンドは，HiRDB を停止した状態で実行してください。

(6)　注意事項

1. pddelfw.bat コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，それ以外の場合は異常終了です。出力される KFPC00406-E メッセージの詳細コードに従って対処してください。

2.28　pddefrev（定義系 SQL の生成）

2.28.1　pddefrev の形式と規則

(1)　機能

HiRDB システム内に定義してある表，インデクス，ビュー表，ストアドプロシジャ，及びトリガの定義系 SQL を生成します。

生成された定義系 SQL は，データベース定義ユティリティの入力ファイルとして利用できます。pddefrev コマンドは，内部的にディクショナリ搬出入ユティリティ（pdexp）を起動して定義系 SQL を生成します。

(2)　実行者

DBA 権限を持つユーザ，又は表の所有者が実行できます。

(3)　形式

```
pddefrev -f 制御文ファイル名
         -o 定義系SQL出力ファイル名
         〔-w ワークディレクトリ名〕
         〔-W 実行監視時間〕
```

(4)　オプション

(a)　-f 制御文ファイル名　～＜パス名＞((1023 バイト以内))

定義系 SQL を生成したい表やストアドプロシジャを指定した制御文ファイルを指定します。制御文ファイルの内容については，「ディクショナリ搬出入ユティリティ（pdexp）」を参照してください。

(b)　-o 定義系 SQL 出力ファイル名　～＜パス名＞((1023 バイト以内))

生成した定義系 SQL を出力するファイルの名称を，絶対パス名で指定します。指定したファイルが既にある場合は上書きし，ない場合は新規に作成します。

(c)　-w ワークディレクトリ名　～＜パス名＞

pddefrev コマンドの内部でディクショナリの搬出をするときに作成する搬出ファイルを，一時的に格納するディレクトリを絶対パス名で指定します。省略した場合は，次に示すディレクトリにファイルを作成します。

<table>
<tr><td rowspan="4">-w オプションの指定</td><td colspan="3">システム定義の pd_tmp_directory オペランドの指定</td></tr>
<tr><td rowspan="3">あり</td><td colspan="2">なし</td></tr>
<tr><td colspan="2">環境変数 TMP の指定</td></tr>
<tr><td>あり</td><td>なし</td></tr>
<tr><td>あり</td><td colspan="3">-w オプションに指定したディレクトリ</td></tr>
<tr><td>なし</td><td>pd_tmp_directory に指定したディレクトリ</td><td>TMP に指定したディレクトリ</td><td>%PDDIR%¥tmp ディレクトリ</td></tr>
</table>

なお，作成される搬出ファイルの名称は次のようになります。

```
pddefrev.exp.HHMMSS
```

HHMMSS には，ローカル時刻が設定されます。

(d)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pddefrev コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pddefrev コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pddefrev コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pddefrev コマンドを実行する場合，定義系 SQL 出力ファイルとワークディレクトリのアクセス権限を，HiRDB 管理者に対して与えておく必要があります。

(6)　注意事項

1. pddefrev コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。pddefrev コマンド実行時のリターンコードは，0 の場合は正常終了，8 又は 12 の場合は異常終了となります。
2. pddefrev コマンドは，同時に複数実行できません。また，ディクショナリ搬出入ユティリティとは同時に実行できません。
3. 生成される定義系 SQL を次に示します。

種別	インデクス定義	コメント定義	生成される定義系 SQL
実表	あり	あり	CREATE TABLE CREATE INDEX COMMENT
		なし	CREATE TABLE CREATE INDEX
	なし	あり	CREATE TABLE COMMENT
		なし	CREATE TABLE
ビュー表	−	あり	CREATE VIEW COMMENT
		なし	CREATE VIEW
ストアドプロシジャ	−	−	CREATE PROCEDURE
トリガ	−	−	CREATE TRIGGER

（凡例）−：該当しません。

4. 定義系 SQL の生成順序は，制御文ファイルでの指定順になります。
5. 生成された定義系 SQL には，データベース定義ユティリティの入力ファイルとして利用できるように，行の最後にセミコロン（；）が付きます。ただし，CREATE PROCEDURE 及び CREATE TRIGGER については，end_proc;が付きます。
6. 次に示す表，手続き，及び関数の定義系 SQL は生成されません。
   - ユーザが定義した抽象データ型を含む表定義（CREATE TABLE）
   - 部分構造インデクス定義（CREATE INDEX）
   - 関数定義（CREATE FUNCTION，及びシステム定義スカラ関数）
   - 抽象データ型内で宣言されている手続き（CREATE TYPE）
7. CREATE PROCEDURE を生成する場合，SQL 最適化オプション及び SQL 拡張最適化オプションの指定値は，10 進数に変換された形式で生成されます。
8. 生成される定義系 SQL は，条件によって改行されます。改行条件については，「定義系 SQL の生成についての規則」を参照してください。
9. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pddefrev の制御文ファイルには，BOM を持つファイルを使用できます。なお，pddefrev の制御文ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pddefrev で出力されるファイルには BOM は付きません。
10. 文字集合 UTF-16 を指定した CHAR 型の列を，キーレンジ分割又はマトリクス分割（境界値指定のキーレンジ分割）の分割キーとして指定した表の場合，分割キーの長さが列の定義長を超えるときは，その表の定義系 SQL は生成できません。

11. DEFAULT 句の ON ROW EXISTS 指定がある表の定義系 SQL を生成する場合は，ON ROW EXISTS 指定なしの表を生成します。

2.29 pdextfunc（拡張システム定義スカラ関数の定義，削除）

2.29.1 pdextfunc の形式と規則

(1) 機能

拡張システム定義スカラ関数の定義又は削除をします。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdextfunc {-c | -e〔force〕}
```

(4) オプション

(a) -c

拡張システム定義スカラ関数を定義する場合に指定します。

(b) -e

拡張システム定義スカラ関数を削除する場合に指定します。ただし，拡張システム定義スカラ関数を使用している関数，手続き又はトリガがあるときは，削除できません。そのときは，force オプションを指定してください。

(c) force

拡張システム定義スカラ関数を使用している関数，手続き又はトリガがある場合で，拡張システム定義スカラ関数を削除するときに指定します。

(5) 規則

1. pdextfunc コマンドは，HiRDB が稼働中のときだけ実行できます。
2. HiRDB サーバへの要求から応答まで 1 分以上経過した場合は，タイムアウトエラーになります。
3. pdextfunc コマンドを実行する前に，次の環境変数を設定してください。
   - クライアント環境定義の PDUSER に DBA 権限を持つユーザの認可識別子及びパスワードを設定してください。

- LANG 又はクライアント環境定義の PDLANG に，HiRDB サーバの文字コード種別に対応した値を設定してください。

詳細は，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

(6) 注意事項

1. pdextfunc コマンドの実行結果は，コマンド実行時のリターンコード又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，4 の場合は警告終了，8 の場合は異常終了です。

2.30 pdfbkup（HiRDB ファイルシステムのバックアップ）

2.30.1 pdfbkup の形式と規則

(1) 機能

HiRDB ファイルシステムをファイルに退避（バックアップ）します。また，HiRDB ファイル単位にも退避できます。

退避するとき，HiRDB ファイルシステムの属性（pdfmkfs コマンドで指定した属性）も一緒に退避します。なお，属性は引き継がれますが，HiRDB ファイルシステムを拡張した後，pdfrstr コマンドで拡張前のバックアップをリストアしても，拡張前の大きさに戻ることはありません。

HiRDB ファイルシステム領域の状態を表示（pdfstatfs コマンドを入力）した時，増分回数，又は不連続な空き領域の総数が算出された場合，次の処置をすると，増分された領域，又は不連続な空き領域を統合し，HiRDB ファイルシステム領域を有効に利用できます。

- HiRDB ファイルシステムの退避（pdfbkup コマンド）
- HiRDB ファイルシステム領域の初期設定（pdfmkfs コマンド）
- 退避した HiRDB ファイルシステムの，初期設定された HiRDB ファイルシステムへの格納（pdfrstr コマンド）

このコマンドは，通常，HiRDB 停止中に実行してください。HiRDB 稼働中に実行する場合は，マニュアル「HiRDB システム運用ガイド」の「バックアップ閉塞を使用してバックアップを取得する場合（pdcopy コマンド以外でバックアップを取得する場合）」を参照してください。このとき，SCHEDULE 属性でない RD エリアを含む HiRDB ファイルシステム領域に対しては，HiRDB の常駐プロセスなどが HiRDB ファイルをオープンしていることがあるため，強制的に退避する-f オプションを指定してください。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdfbkup 〔-y〕 〔-r〕 〔-i〕 〔- {c | l | f} 〕
          HiRDBファイルシステム領域名〔¥HiRDBファイル名〕
          バックアップファイル名
```

(4) オプション

(a) -y

HiRDB ファイルシステムを退避するときに，ユーザの確認なしで実行する場合に指定します。省略した場合，確認してから実行します。

(b) -r

指定したバックアップファイルが既にある場合，削除して再作成するときに指定します。省略した場合，バックアップファイルが既にあるときにはエラーとなります。

(c) -i

HiRDB ファイルシステムの退避中にエラーが発生した場合に，その HiRDB ファイルの退避をやめ，次のファイルから処理を続行するときに指定します。省略した場合は，エラー終了します。

(d) -{c｜l｜f}

他プロセス又は他ユーザが，対象となる HiRDB ファイルシステム領域にアクセスしている場合の処理を指定します。

-c：

他プロセス又は他ユーザが，対象となる HiRDB ファイルシステム領域にアクセスしている場合は，退避処理をしません。

-l：

他プロセス又は他ユーザが，対象となる HiRDB ファイルシステム領域にアクセスしている場合，アクセス中の HiRDB ファイルは退避しません。

-f：

他プロセス又は他ユーザが，対象となる HiRDB ファイルシステム領域にアクセスしている場合でも，強制的に退避します。

(5) コマンド引数

(a) HiRDB ファイルシステム領域名〔¥HiRDB ファイル名〕

退避する HiRDB ファイルシステム領域名と HiRDB ファイル名を指定します。

HiRDB ファイルシステム領域名 ～＜パス名＞((165 文字以内))

HiRDB ファイルシステム領域名を指定します。

HiRDB ファイル名 ～＜ HiRDB ファイル名＞

HiRDB ファイル名を指定します。

(b)　バックアップファイル名

退避先のファイル名を指定します。

(6)　規則

1. pdfbkup コマンドは，HiRDB の稼働に関係なく実行できます。なお，稼働中に実行する場合は-f オプションを指定してください。
2. pdfbkup コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。
3. 一つのファイルには，一つの HiRDB ファイルシステムしか退避できません。そのため，複数の HiRDB ファイルシステムを退避する場合は，その数分のファイルが必要になります。
4. HiRDB ファイルシステムの退避中にエラーが発生した場合，バックアップファイルは作成しません。
5. -y オプションを省略した場合は，確認メッセージが表示されます。このとき，確認メッセージに対してＧ以外の文字を指定した場合は，HiRDB ファイルシステムは退避されません。
6. 退避する HiRDB ファイルのレコード長は，pdfls コマンドで確認しておいてください。レコード長が，回復先 HiRDB ファイルシステム領域のセクタ長の整数倍でない場合，退避したファイルは回復できません。
7. 共用 RD エリアとして使用している HiRDB ファイルシステム領域に対して実行する場合，更新可能バックエンドサーバがあるサーバマシンから実行してください。

(7)　注意事項

1. pdfbkup コマンドの実行結果は，コマンド実行時のリターンコード又はエラーメッセージの有無で確認できます。
2. pdfbkup コマンドのリターンコードを次に示します。

   0：正常終了

   1：警告終了（一部の HiRDB ファイルの処理がエラー終了）

   255 又は-1：異常終了

   リターンコードが 0 以外の場合，バックアップファイルが正常に作成できていないおそれがあります。エラーの要因を取り除いた後，バックアップファイルを削除し，再度コマンドを実行してください。

   なお，使用しているシェルによっては-1 になります。
3. pdfbkup コマンドが異常終了した場合，バックアップファイルが作成されることがありますが，ファイルの内容は保証できません。

2.31　pdfchfs（HiRDB ファイルシステム領域の管理情報変更）

2.31.1　pdfchfs の形式と規則

(1)　機能

HiRDB ファイルシステム領域の最大増分回数（pdfmkfs の-e オプションの指定値）を変更します。

プロセス固有領域不足が発生した際，pdfmkfs コマンドの-e オプションに見積もり値以上の不当に大きな値を指定していた場合は，これを適切な値に変更することで，プロセス固有領域のメモリ所要量を削減できます。

pdfchfs コマンドは，次のどちらかに該当するときに実行します。

- 大量のデータを格納しており，HiRDB ファイルシステム領域の再作成に伴うデータの回復作業に時間を要する場合
- HiRDB ファイルシステム領域の再作成のために HiRDB を停止できない場合

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdfchfs -e 最大増分回数 HiRDBファイルシステム領域名
```

(4)　オプション

(a)　-e 最大増分回数 ～＜符号なし整数＞((0～60000))

pdfmkfs コマンドの-e オプションで指定した，HiRDB ファイルシステム領域の最大増分回数を変更します。最大増分回数は，pdfstatfs コマンド実行結果の[available expand count]に表示される値で確認できます。ただし，pdfmkfs コマンドの-e オプションの指定値を超える値は指定できません。

pdfmkfs コマンドの-e オプションに指定していた値が見積もりで算出した値以上の場合，最大増分回数を小さくすることで，各 HiRDB サーバプロセスが使用するプロセス固有領域のメモリ所要量を削減できます。

メモリ所要量の削減量の目安を次に示します。

- pdfmkfs コマンドの-n オプション指定値が 2048 以上の場合：

  ```
  ↑｛（24×a）÷b｝↑×b　［単位：バイト］
  ```

- pdfmkfs コマンドの-n オプション指定値が 2047 以下の場合：

```
↑｛（16×a）÷b｝↑×b　［単位：バイト］
```

可変記号の説明

a：pdfmkfs -e オプションの指定値

b：pdfmkfs -s オプションの指定値（pdfstatfs -b コマンド実行結果の[sector size]に表示される値。省略した場合，ダイレクトディスクアクセス（raw I/O）のときは 1024，ファイルのときは 512）

サーバプロセスが使用するプロセス固有領域の詳細は，マニュアル「HiRDB システム導入・設計ガイド」の「メモリ所要量の計算式」を参照してください。

＜注意事項＞

1. 指定値が次の条件を満たさない場合，コマンドは KFPI21604-E を出力しエラー終了します。
   - 下限：現時点での使用増分回数
   - 上限：pdfmkfs コマンド実行時の-e オプション指定値
2. 変更前と同じ値を指定した場合，コマンドは正常終了します。
3. 次の値より小さい値を指定した場合，HiRDB ファイルの作成又は拡張時に，HiRDB ファイルの増分限界に達する前に領域の最大増分回数に達してエラーになるおそれがあります。

   ```
   23×増分を使用するHiRDBファイル数の累計
   ```

   HiRDB ファイルシステム領域の使用目的ごとに，「pdfmkfs（HiRDB ファイルシステム領域の初期設定）」を参照してください。

(5)　コマンド引数

(a)　HiRDB ファイルシステム領域名　～＜パス名＞((165 文字以内))

HiRDB ファイルシステム領域名を指定します。

＜注意事項＞

1. 次のすべての条件を満たす HiRDB ファイルシステム領域に対しては実行できません。実行した場合，コマンドは KFPI21603-E を出力しエラー終了します。
   - pdfmkfs コマンドに-a オプションを指定して，HiRDB ファイルシステム領域を自動的に拡張する指定をしている場合

     自動的に拡張する指定をしているかどうかは pdfstatfs コマンド実行結果の[area auto expand]に表示される値で確認できます。
   - pdfmkfs コマンドの-k オプションで指定した，HiRDB ファイルシステム領域の使用目的が DB，SDB，又は SYS の場合

     使用目的は pdfstatfs コマンド実行結果の[initialize area kind]に表示される値で確認できます。

(6) 規則

1. pdfchfs コマンドは，HiRDB の稼働状況に関係なく実行できます。ただし，HiRDB ファイルシステム領域の使用目的が SVR の場合を除きます。

   HiRDB 停止中に実行する場合の制限はありません。HiRDB 稼働中に実行する場合は，pdfmkfs コマンドの-k オプションで指定した，HiRDB ファイルシステム領域の使用目的ごとに，次に示す手順で実行してください。

項番	使用目的	pdfchfs コマンド実行時の手順
1	DB 又は SDB	1. 対象 RD エリアを閉塞かつクローズ状態にします。コマンド実行対象の HiRDB ファイルシステム領域に格納している HiRDB ファイルを構成ファイルとするすべての RD エリアが対象です。 2. プロセスのリフレッシュ（pdpfresh）をします。 3. pdfchfs コマンドを実行します。 4. RD エリアの閉塞かつクローズ状態を解除します。 **＜注意事項＞** • マスタディレクトリ用 RD エリアの構成ファイルを格納している HiRDB ファイルシステム領域は適用できません。適用するには，HiRDB を停止して実行する必要があります。 • プロセスをリフレッシュする際，HiRDB ファイルにアクセスしているプロセスを強制的にリフレッシュしてよい場合には pdpfresh コマンドの-f オプションを指定して実行します。そうでない場合にはこのプロセスが実行中のジョブが終了するまで待つか，そのジョブを終了させてからコマンドを実行してください。HiRDB ファイルにアクセスしているプロセスの確認方法及び，pdpfresh の使用方法については，「pdpfresh（サーバプロセスのリフレッシュ）」を参照してください。
2	SYS	1. 対象の HiRDB ファイルをクローズします。コマンド実行対象の HiRDB ファイルシステム領域に格納している HiRDB ファイルすべてが対象です。 2. pdfchfs コマンドを実行します。 3. HiRDB ファイルをオープンします。 **＜注意事項＞** • HiRDB ファイルシステム領域内に現用の HiRDB ファイルが存在する場合，別領域の HiRDB ファイルにスワップした後に実行してください。 • 全世代を同じ HiRDB ファイルシステム領域に格納している場合は適用できません。適用するには，HiRDB を停止して実行してください。HiRDB/パラレルサーバでサーバ用 HiRDB ファイルだけを格納している場合は，サーバ単位の終了でも実行できます。 • 監査証跡ファイルを格納している HiRDB ファイルシステム領域は適用できません。適用するには，HiRDB を停止して実行してください。
3	WORK	1. 作業表を使用するトランザクションの運用をいったん停止します。 2. pdfchfs コマンドを実行します。 **＜留意事項＞** コマンド実行中にトランザクションを実行した場合，pdfchfs コマンド終了までトランザクションは実行待ちになり，pdfchfs コマンド実行後に動作を再開します。
4	UTL	1. 対象 HiRDB ファイルシステム領域を指定するユティリティの運用をいったん停止します。

項番	使用目的	pdfchfs コマンド実行時の手順
		2. pdfchfs コマンドを実行します。 **＜留意事項＞** コマンド実行中にユティリティを実行した場合，pdfchfs コマンド終了までユティリティは実行待ちになり，pdfchfs コマンド実行後に動作を再開します。
5	SVR	HiRDB 稼働中に実行はできません。HiRDB を停止してから実行してください。

この手順に従わないで pdfchfs コマンドを実行した場合，対象の HiRDB ファイルシステム領域を使用中のプロセスが存在すると，コマンドは KFPI21528-E を出力してエラー終了します。エラー終了した場合はこの手順に従ってコマンドを再実行してください。

2. pdfchfs コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。
3. 共用 RD エリアとして使用している HiRDB ファイルシステム領域に対して実行する場合，更新可能バックエンドサーバがあるサーバマシンから実行してください。

(7) 注意事項

1. pdfchfs コマンドでは，pdfmkfs コマンド実行時に作成した管理情報のサイズは変更できません。管理部に使用しない無駄な領域ができるため，pdfmkfs コマンド再実行による変更を推奨します。

2.32　pdffsck（HiRDB ファイルシステム領域の整合性の検証及び修復）

2.32.1　pdffsck の形式と規則

(1)　機能

HiRDB ファイルシステム領域の整合性を検証して，不整合があれば修復します。また，不整合を検知した場合は，検知したファイル又は領域を表示します。

pdffsck コマンドは，次のどちらかに該当する場合に実行します。

- pdfls コマンド実行時に KFPI21586-W メッセージが表示された場合
- HiRDB ファイルシステム領域が，本来の容量より少ない容量しか使用できなくなった場合（例えば，100 ギガバイト使えるはずが 80 ギガバイトまでしか使えないなど）

整合性の検証内容と修復方法を次に示します。

対象	検証内容	原因	修復方法
HiRDB ファイル	使用できない HiRDB ファイルの有無	ファイルの作成，又は削除処理の中断	未使用状態として管理します。また，割り当てられていた領域は空き領域として管理します。
HiRDB ファイル内の領域	空き領域リスト，及びファイル管理テーブルのどちらからも参照されない領域の有無	ファイルの拡張（自動増分）処理の中断	孤立している領域を空き領域として管理します。
HiRDB ファイルシステム領域の管理情報	HiRDB ファイルシステム領域の管理領域に格納されている統計情報の矛盾	ファイルの作成，又は削除処理の中断	各検証が終了した状態で，ファイル数の統計情報などを修正します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdffsck 〔-c〕 HiRDBファイルシステム領域名
```

(4)　オプション

(a)　-c

HiRDB ファイルシステム領域の検証だけをするときに指定します。-c オプションを指定した場合，HiRDB ファイルシステム領域の修復はしません。

(5) コマンド引数

(a) HiRDB ファイルシステム領域名 ～＜パス名＞((165 文字以内))

HiRDB ファイルシステム領域の名称を指定します。

(6) 規則

1. pdffsck コマンドは，-c オプションありの場合は，HiRDB の稼働に関係なく実行できます。-c オプションなしの場合は，HiRDB ファイルシステム領域の使用目的ごとに次のように実行してください。
   - -k DB の HiRDB ファイルシステム領域の場合

     HiRDB ファイルシステム領域内のすべての RD エリアを，pdhold -c コマンドでコマンド閉塞かつクローズ状態にして，pdpfresh コマンドで稼働中のプロセスを終了させた後に実行してください。また，pdffsck コマンドを実行した後は，pdrels コマンドですべての RD エリアの閉塞状態を解除してください。
   - -k SYS の HiRDB ファイルシステム領域の場合

     pdlogswap コマンドでシステムログファイルをスワップして，pdlogcls コマンドでクローズ状態にした後に実行してください。また，pdffsck コマンドを実行した後は，pdlogopen コマンドでシステムログファイルを稼働中の HiRDB で使用できるようにしてください。
   - -k UTL の HiRDB ファイルシステム領域の場合

     ユティリティで使用中の場合は，ユティリティの処理が終了した後に実行してください。
   - -k WORK の HiRDB ファイルシステム領域の場合

     HiRDB 稼働中には修復できないため，HiRDB を停止してから実行してください。
2. -c オプションありの pdffsck コマンドは，他プロセス又は他ユーザがアクセスしている HiRDB ファイルシステム領域に対して実行できます。-c オプションなしの pdffsck コマンドは，他プロセス又は他ユーザがアクセスしている HiRDB ファイルシステム領域に対しては実行できません。
3. pdffsck コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。
4. -k SDB の HiRDB ファイルシステム領域に対して pdffsck コマンドを実行する場合，更新可能バックエンドサーバがあるサーバマシンから実行してください。

(7) 注意事項

1. pdffsck コマンドのリターンコードを次に示します。

   0：正常終了

   1：警告終了（複数ディスクの操作でエラーが発生）

   255 又は-1：異常終了

   リターンコードが 0 以外の場合，処理が正常に完了していないおそれがあります。エラーの要因を取り除いた後，再度コマンドを実行してください。

   なお，使用しているシェルによっては-1 になります。

2. I/O エラーが発生した場合は処理を中止します。
3. 対象とする HiRDB ファイルシステム領域に作成されている HiRDB ファイルは，pdfls コマンドで確認できます。
4. RD エリアの自動増分機能適用時に HiRDB ファイルシステム領域を検証する場合は，検証処理が終了するまで自動増分はしないで，検証処理終了後に自動増分を実行します。対象となる HiRDB ファイルシステム領域に，RD エリアの自動増分機能を適用しているかどうかは，pdfstatfs コマンドで確認できます。
5. pdffsck コマンドの実行中には，pdmod 及び pdrstr は実行しないでください。実行すると，データベースを破壊するおそれがあります。
6. pdffsck コマンドでのファイル修復処理中にプロセスが打ち切られた場合は，コマンドを再実行して修復を完了させてください。
7. バージョン 07-02 以前の HiRDB では，管理領域の情報が破壊されていて pdffsck コマンドでは修復できないことがあります。この場合は，KFPI21585-E メッセージが出力されるので，出力メッセージに従って，対象となる HiRDB ファイルシステム領域をバックアップファイルから回復してください。詳細については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(8) 出力形式

領域の不整合が検知されて KFPI21559-W メッセージが表示された場合だけ，検知した不整合の状態によって，次のどれかの情報が出力されます。なお，不整合が検知されないで，KFPI21558-I メッセージが表示された場合には，次の情報は出力されません。

(a) 利用できない HiRDB ファイルがある場合（ファイルの作成，削除処理の中断）

利用できない HiRDB ファイルは，ファイルに割り当てられていた領域が利用できる状態とできない状態で区別します。領域が利用できない状態のファイルについては，すべてのファイルのファイル名，ファイル開始位置，及びファイルサイズが表示されます。利用できないファイルのすべてが，領域を利用できない状態とは限らないため，出力されるファイルカウントとファイルリストの数が一致しないこともあります。

```
unusable file count                 aa...a
file list
     FILE NAME                   OFFSET[kB]    SIZE[kB]
     bb...b                         cc...c       dd...d
       :                               :            :
total unusable area size[kB]              ee...e
```

[説明]

aa...a：

利用できない HiRDB ファイルの数（10 進数 4 けた以内）

bb...b：

領域が利用できない状態の HiRDB ファイル名（30 文字以内）

cc...c：

HiRDB ファイルの開始オフセット（10 進数 10 けた以内）

dd...d：

HiRDB ファイルのサイズ（単位：キロバイト）（10 進数 10 けた以内）

ee...e

領域が利用できない状態の HiRDB ファイルの合計サイズ（修復実行によって空き領域として再利用できるようになるサイズ）（単位：キロバイト）（10 進数 10 けた以内）

(b) 利用できない領域がある場合（ファイルの拡張処理の中断）

利用できない領域の開始位置とサイズが表示されます。該当するすべての領域が出力されます。出力される領域カウントと領域リストの数は常に一致します。

```
unusable area count                aa...a
area list
     OFFSET[kB]                      SIZE[kB]
     bb...b                            cc...c
       :                                  :
total unusable area size[kB]           dd...d
```

[説明]

aa...a：

利用できない領域数（10 進数 4 けた以内）

bb...b：

利用できない領域開始オフセット（10 進数 10 けた以内）

cc...c：

利用できない領域サイズ（単位：キロバイト）（10 進数 10 けた以内）

dd...d：

利用できない領域の合計サイズ（修復実行によって空き領域として再利用できるようになるサイズ）（単位：キロバイト）（10 進数 10 けた以内）

(c) 利用できない HiRDB ファイル，及び利用できない領域が共にある場合

```
unusable file count                 aa...a
file list
     FILE NAME                  OFFSET[kB]    SIZE[kB]
     bb...b                        cc...c       dd...d
       :                              :            :
unusable area count                 ff...f
area list
     OFFSET[kB]                      SIZE[kB]
     gg...g                            hh...h
       :                                  :
total unusable area size[kB]           ee...e
```

［説明］

aa...a：

利用できない HiRDB ファイルの数（10 進数 4 けた以内）

bb...b：

領域が利用できない状態の HiRDB ファイル名（30 文字以内）

cc...c：

HiRDB ファイルの開始オフセット（10 進数 10 けた以内）

dd...d：

HiRDB ファイルのサイズ（単位：キロバイト）（10 進数 10 けた以内）

ee...e：

領域が利用できない状態の HiRDB ファイル，及び利用できない領域すべての合計サイズ（修復実行によって空き領域として再利用できるようになるサイズ）（単位：キロバイト）（10 進数 10 けた以内）

ff...f：

利用できない領域数（10 進数 4 けた以内）

gg...g：

利用できない領域開始オフセット（10 進数 10 けた以内）

hh...h：

利用できない領域サイズ（単位：キロバイト）（10 進数 10 けた以内）

2.33 pdfgt（トランザクションの強制終了）

2.33.1 pdfgt の形式と規則

(1) 機能

グローバルトランザクションを構成している各トランザクションブランチが，何らかの要因（通信障害など）でトランザクションを終了できない場合に強制的にそのトランザクションを終了します。

例えば，次の場合はトランザクションが終了しません。

- OLTP システムと HiRDB の間でトランザクションの同期合わせをする前に，通信障害が発生した場合
- OLTP システムと HiRDB の間でトランザクションの同期合わせをする前に，OLTP システムが異常終了した場合

このような場合に pdfgt コマンドを使用して，トランザクションを強制的に終了します。また，上記以外の障害が発生し，トランザクションを強制的に終了するときにも pdfgt コマンドを使用できます。

pdfgt コマンドは，pdls コマンド（-d trn 指定）で表示された情報のトランザクション第 1 状態が FORGETTING でトランザクション第 3 状態が w のトランザクションを強制的に終了します。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

(a) HiRDB/シングルサーバの場合

```
pdfgt 〔 {-x ホスト名 | -u ユニット識別子} 〕 〔-s サーバ名〕
      {-A | -t トランザクション識別子}
```

(b) HiRDB/パラレルサーバの場合

```
pdfgt {-x ホスト名 | -u ユニット識別子} 〔-s サーバ名〕
      {-A | -t トランザクション識別子}
```

(4) オプション

(a) {-x ホスト名｜-u ユニット識別子}

該当するトランザクションがあるホストのホスト名，又はユニットのユニット識別子を指定します。

-x ホスト名 ～＜識別子＞((1～32))

該当するトランザクションがあるホストのホスト名を指定します。

-u ユニット識別子 ～＜識別子＞((4 文字))

該当するトランザクションがあるユニットのユニット識別子を指定します。

(b) -s サーバ名 ～＜識別子＞((1～8))

ユニットに複数のサーバがある場合，サーバを特定するときにサーバ名を指定します。省略した場合，対象となるホスト内のすべてのサーバを対象とします。

(c) -A

-s オプションで指定したサーバ内のトランザクションのうち，FORGETTING 状態で同期合わせ待ち状態のすべてのトランザクションを終了する場合に指定します。

(d) -t トランザクション識別子 ～＜識別子＞((16 文字))

FORGETTING 状態で同期合わせ待ち状態のトランザクションを終了する場合に，トランザクション識別子を指定します。

(5) 規則

1. pdfgt コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdfgt コマンドは，各サーバマシンで実行してください。

(6) 注意事項

1. pdfgt コマンドの結果は，pdls コマンド（-d trn 指定），コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   1：異常終了（オプション指定不正，rsh 失敗など）

   対象のトランザクションが終了できていないことがあります。

   メッセージに従って対処した後，再度コマンドを実行してください。
2. 回復不要 FES に接続して実行したトランザクションは，トランザクション第 1 状態が FORGETTING，トランザクション第 3 状態が w の場合でも，強制的に終了できないことがあります。この場合，トランザクションを自動決着してください。未決着状態のトランザクションを決着する方法の詳細について

は，マニュアル「HiRDB システム運用ガイド」の「未決着状態のトランザクションを決着する方法」を参照してください。

3. 現在は停止しているクライアント製品からの要求で発生した未決着状態のトランザクションを強制的に終了すると，この後でクライアント製品を開始したときに，HiRDB とクライアント製品との間でこのトランザクションの同期が取れないことがあります。

2.34　pdfls（HiRDB ファイルシステムの内容表示）

2.34.1　pdfls の形式と規則

(1)　機能

HiRDB ファイルシステム領域内の HiRDB ファイルの内容を表示します。コマンド引数に HiRDB ファイルシステム領域名だけを指定した場合には，指定した HiRDB ファイルシステム領域内にあるすべての HiRDB ファイルについて出力します。HiRDB ファイル名も指定した場合は，指定した HiRDB ファイルについて出力します。

(2)　実行者

権限がないユーザでも実行できます。

(3)　形式

```
  pdfls  { 〔 〔-H〕  〔-L | -S〕  〔- {t | u} 〕 〕  HiRDBファイルシステム領域名〔¥HiRDBファイル名〕

         | 〔-x〕  HiRDBファイルシステム領域名〔¥HiRDBファイル名〕}
```

(4)　オプション

(a)　-H

表示する情報にヘッダを付けて，HiRDB ファイル名のアルファベット順に表示します。

(b)　-L

HiRDB ファイルの排他制御の状態を，HiRDB ファイル名のアルファベット順に縦方向で表示します。

(c)　-S

HiRDB ファイルの分割格納状態（エクステントの状態）を，HiRDB ファイル名のアルファベット順に縦方向で表示します。

(d)　- {t | u}

-t：

HiRDB ファイルシステム領域の内容を，更新クローズ日時が最新のものから順に表示します。

更新クローズ日時とは，ファイルオープン後 1 回以上書き込みをしたときの日時です。書き込みをしていない HiRDB ファイルについては，ファイルの作成日時を更新クローズ日時とみなします。なお，RD

エリアを構成する HiRDB ファイルについては，更新クローズがあっても更新クローズ日時を利用しないことがあります。

-u：

HiRDB ファイルシステム領域の内容を，クローズ日時が最新のものから順に表示します。

クローズ日時は，オープン後 1 回以上書き込み，読み込みをしたときの日時です。書き込み，読み込みをしていない HiRDB ファイルについては，ファイルの作成日時をクローズ日時とみなします。なお，RD エリアを構成する HiRDB ファイルについては，クローズがあってもクローズ日時を利用しないことがあります。

-t 又は-u オプションと，-H 又は-L オプションを同時に指定した場合は，-t 又は-u オプションの指定が有効となります。

(e) -x

HiRDB ファイル名だけをアルファベット順に横方向で表示したい場合に指定します。省略した場合，縦方向に表示します。

(5) コマンド引数

(a) HiRDB ファイルシステム領域名 ～＜パス名＞((165 文字以内))

HiRDB ファイルシステム領域名を指定します。

(b) HiRDB ファイル名 ～＜ HiRDB ファイル名＞

HiRDB ファイル名を指定します。

(6) 規則

1. pdfls コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdfls コマンドは，HiRDB ファイルシステムがあるサーバマシンごとに実行してください。
3. 共用 RD エリアとして使用している HiRDB ファイルシステム領域に対して実行する場合，更新可能バックエンドサーバがあるサーバマシンから実行してください。

(7) 注意事項

1. pdfls コマンド実行時のリターンコードは，0 の場合は正常終了，255 の場合は異常終了となります。リターンコード 255 は，使用しているシェルによっては-1 になります。

(8) 出力形式

各項目は，1 文字以上の半角空白を空けて表示します。

(a)　-H オプションを指定した場合

```
MODE    UID    GID    RSIZE  RNUM   TIME   FILE
aabbcc  dd...d ee...e ff...f gg...g hh...h ii...i
```

[説明]

aabbcc：システムが使用する情報（rwrw--固定）

dd...d：

HiRDB ファイルを作成した OS 上のユーザ名

ee...e：

HiRDB ファイルを作成した OS 上のユーザが属するグループ名

常に 0 が表示されます。

ff...f：

レコード長（10 進数 5 けた）

gg...g：

レコード数（10 進数 8 けたで右詰め表示）

hh...h：

更新クローズ日時

時：分　月　日　年（西暦）の形式で表示されます。

2005 年 1 月 7 日(金)9 時 5 分の場合の出力例を次に示します。

09:05 Jan 07 2005

ii...i：

HiRDB ファイル名（30 文字以内）

(b)　-H 及び-L オプションを指定した場合

```
MODE    UID    GID    PID    L TIME   FILE
aabbcc  dd...d ee...e pp...p q hh...h ii...i
```

[説明]

aabbcc：システムが使用する情報（rwrw--固定）

dd...d：

HiRDB ファイルを作成した OS 上のユーザ名

ee...e：

HiRDB ファイルを作成した OS 上のユーザが属するグループ名

常に 0 が表示されます。

pp...p：

排他を掛けているプロセスがある場合は*が，排他を掛けているプロセスがない場合は–が表示されます。

q：

排他状態識別フラグ

E：

排他が掛けられています。

-：

排他は掛けられていません。

hh...h：

更新クローズ日時

時：分　月　日　年（西暦）の形式で表示されます。

2005 年 1 月 7 日(金)9 時 5 分の場合の出力例を次に示します。

09:05 Jan 07 2005

ii...i：

HiRDB ファイル名（30 文字以内）

(c)　-H 及び-S オプションを指定した場合

```
MODE    UID     GID     RSIZE  RNUM    TIME    FILE
aabbcc  dd...d  ee...e  ff...f gg...g  hh...h  ii...i
                extent size    jj...j
```

[説明]

aabbcc：システムが使用する情報（rwrw--固定）

dd...d：

HiRDB ファイルを作成した OS 上のユーザ名

ee...e：

HiRDB ファイルを作成した OS 上のユーザが属するグループ名

常に 0 が表示されます。

ff...f：

レコード長（10 進数 5 けた）

gg...g：

レコード数（10 進数 8 けたで右詰め表示）

hh...h：

更新クローズ日時

時：分　月　日　年（西暦）の形式で表示されます。

2005 年 1 月 7 日(金)9 時 5 分の場合の出力例を次に示します。

09:05 Jan 07 2005

ii...i：

HiRDB ファイル名

jj...j：

HiRDB ファイルの１エクステント当たりのレコード数。HiRDB ファイルの１エクステントが複数ある場合，エクステントごとに表示されます。

2.35　pdfmkfs（HiRDB ファイルシステム領域の初期設定）

2.35.1　pdfmkfs の形式と規則

(1)　機能

指定したファイルを HiRDB ファイルシステム領域用に初期設定します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdfmkfs -n HiRDBファイルシステム領域サイズ 〔-l 最大ファイル数〕
        〔-k 使用目的〕 〔-e 最大増分回数〕 〔-s セクタ長〕 〔-i〕 〔-r〕 〔-a〕
         ファイル名
```

(4)　オプション

(a)　-n HiRDB ファイルシステム領域サイズ　～＜符号なし整数＞

HiRDB ファイルシステム領域として割り当てる容量をメガバイト単位で指定します。このオプションは，必ず指定してください。指定範囲を次に示します。

ファイルの種類	指定範囲
ファイル（NTFS）	1～1,048,575
ダイレクトディスクアクセス（raw I/O）	1～1,048,575

このオプションと同時に-k DB -a，-k SDB -a，-k SYS -a，又は-k WORK -a を指定すると，このオプションで指定した HiRDB ファイルシステム領域の容量を使い切ったときに自動的に領域を拡張します。

なお，HiRDB ファイルシステム領域の管理で使用する管理領域も，-n オプションの指定値に含まれます。したがって，実際に HiRDB ファイルに割り当てられる容量は，（-n オプションの指定値）−（管理領域の容量）となります。管理領域の容量の求め方については，注意事項を参照してください。実際に HiRDB ファイルに割り当てられる容量は，pdfstatfs コマンドで確認できます。

(b)　-l　最大ファイル数　～＜符号なし整数＞((1～4096))

HiRDB ファイルシステム領域内に作成する HiRDB ファイル数の上限を指定します。

領域内の HiRDB ファイル数は，少ない方が性能が向上します。

-a オプションを指定しない場合は，このオプションを必ず指定してください。

なお，ここで指定した値は，pdfmkfs コマンド実行後には変更できません。したがって，使用する HiRDB ファイル数と，今後の拡張予定などを考慮して値を決める必要があります。最大ファイル数の計算式については，マニュアル「HiRDB システム導入・設計ガイド」の最大ファイル数の見積もり（pdfmkfs -l コマンド）を参照してください。

このオプションと同時に-k DB -a，-k SDB -a，又は-k SYS -a は指定できません。

(c)　-k 使用目的　～《SVR》

HiRDB ファイルシステム領域の使用目的を指定します。

DB：

RD エリア用の HiRDB ファイルシステム領域として初期化します。

SDB：

共用 RD エリア用の HiRDB ファイルシステム領域として初期化します。SDB は，HiRDB/パラレルサーバの場合で，かつ HiRDB ファイルシステム領域としてダイレクトディスクアクセスを使用する場合に指定できます。通常の RD エリアは，使用目的 DB の HiRDB ファイルシステム領域を使用してください。

使用目的 SDB で初期化する場合，更新可能バックエンドサーバがあるサーバマシンで実行してください。参照可能バックエンドサーバからは実行しないでください。

SYS：

システムログファイル，シンクポイントダンプファイル，ステータスファイル，又は監査証跡ファイル用の HiRDB ファイルシステム領域として初期化します。

SYS を指定した場合，-e オプションは指定しないでください。

WORK：

リスト用 RD エリア，又は作業表用ファイルの HiRDB ファイルシステム領域として初期化します。

UTL：

ユティリティ用の HiRDB ファイルシステム領域として初期化します。

ユティリティ用の HiRDB ファイルシステム領域には，次のファイルを作成できます。

- アンロードデータファイル
- LOB データのアンロードファイル
- インデクス情報ファイル（プラグインの遅延一括作成の場合）
- バックアップファイル
- 差分バックアップファイル
- 差分バックアップ管理ファイル
- アンロードログファイル

Windows のキャッシュを使用しないでアクセスをします。

バージョン 09-50 より前の HiRDB で作成した HiRDB ファイルシステム領域について

- NUTL で作成した HiRDB ファイルシステム領域は，バージョン 09-50 以降も動作に変更はありません。
- UTL で作成した HiRDB ファイルシステム領域は，バージョン 09-50 以降は Windows のキャッシュを使用しないでアクセスするようになります。

NUTL：

バージョン 09-50 より前の HiRDB との互換のための指定です。NUTL を指定すると UTL を指定したものとみなします。

SVR：

ユティリティ用を除くすべての目的に使用できる HiRDB ファイルシステム領域として初期化します。

テスト用のシステムなどで，一つの HiRDB ファイルシステム領域に，異なる使用目的の HiRDB ファイルを混在させても問題ない場合に指定します。

通常は信頼性の面から，使用目的によって HiRDB ファイルシステム領域を分けるようにしてください。

系切り替え機能で使用する HiRDB ファイルシステム領域を作成する場合，SVR を指定しないでください。SVR を指定して作成した HiRDB ファイルシステム領域に対して，HiRDB は Windows のファイルキャッシュに書き込んだ時点で処理を続行します。このため，ディスク上にデータが書き込まれる前に異常が発生するとファイルの整合性がとれなくなります。

(d)　-e　最大増分回数　～＜符号なし整数＞((0～60000))≪0≫

このオプションを指定すると，HiRDB ファイルの容量が不足した場合，自動的に HiRDB ファイルの容量が増分されます。このオプションには容量を増分する回数の上限値を指定します。

HiRDB ファイルシステム領域内の全 HiRDB ファイルの容量増分回数がカウント対象になります。例えば，HiRDB ファイルシステム領域内に二つの HiRDB ファイルがあり，それぞれ増分した回数が 5 回，3 回の場合，増分回数は 8 回とカウントされます。

使用目的が SYS の場合，指定した値は無効となり 0 が仮定されます。増分回数に応じて HiRDB ファイルシステム領域を使用するプロセスのメモリ所要量が増加するため，指定値の求め方に記載した値より大きい値を指定しないでください。指定値の求め方に記載した値より大きい値を指定した場合，メモリ不足の原因になります。例えば，最大値 60000 と最小値 0 では，メモリ所要量が約 1.8 メガバイト異なります。メモリ所要量の計算式については，マニュアル「HiRDB システム導入・設計ガイド」の「メモリ所要量の計算式」を参照してください。

なお，1HiRDB ファイルの容量増分回数の上限が 23 回のため，このオプションに最大ファイル数（-l オプションの指定値）×23 以上の値を指定しても無効になります。この場合，最大増分回数には最大ファイル数×23 が仮定されます。1 領域 1 ファイルにする場合（-l の指定値に 1 を指定する場合）は，上限が 23 回という条件は適用されません。領域内で，増分によってファイル構成データが分断されることがないため，1 を指定すれば HiRDB ファイルシステム領域サイズ（-n 指定値）まで増分ができます。ただし，HiRDB ファイルの上限サイズを超えることはできません。

このオプションと同時に-k DB -a，-k SDB -a，又は-k SYS -a は指定できません。

＜適用基準＞

1. 作業表用ファイルを作成する HiRDB ファイルシステム領域の場合，作業表用ファイルの容量見積もりの結果が次の値を超えるときは，必ず増分回数を指定してください。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
   ・HiRDB/シングルサーバの場合
   128 キロバイト
   ・HiRDB/パラレルサーバの場合
   512 キロバイト
2. 容量見積もりの結果が上記の値を超えない場合，及び作業表用ファイル以外の HiRDB ファイルシステム領域の場合でも，増分回数を指定することをお勧めします。
3. -e オプションで RD エリアを自動増分する場合は，必ず増分回数を指定してください。
4. RD エリアの削除，再初期化（割り当てサイズの縮小，又は with reconstruction 指定），及び統合では，割り当て済み HiRDB ファイルの削除やサイズ縮小が発生するため，HiRDB ファイルシステム領域内に断片化した空きが発生します。このような運用をする場合には，増分回数を指定することをお勧めします。RD エリアの追加，拡張，及び再初期化で新しく作成する HiRDB ファイルは，連続した空きを確保できない場合，断片化した空きを複数個使用します。この場合，使用する「断片化した空き数−1」の増分回数が必要になります。

＜指定値の求め方＞

指定する増分回数は，次の式から求めてください。

-k オプション	増分回数の指定値
DB※1	Min（60000，23×増分対象ファイル数の累計※2） なお，RD エリアの削除，再初期化（割り当てサイズの縮小，又は with reconstruction 指定），及び統合を頻繁に行う場合は，次の値を指定することをお勧めします。 Min（60000，23×RD エリアの構成ファイル数の累計）
SDB※1	
SYS※1	0
WORK	Min（60000，23×-l オプションの指定値）
UTL	
SVR	Min（60000，23×増分を適用するファイル数の累計） 増分を適用するファイルは，作業表用ファイル，及び増分対象ファイルです。※2

注 1

pdfmkfs 実行後は，増分回数の増加はできないため，将来の拡張を見込んで値を決めてください。なお，pdfchfs コマンドで，増分回数の削減はできます。

注 2

計算値より少ない値を指定した場合，HiRDB ファイルの増分の限界に達する前に，HiRDB ファイルシステム領域の限界になります。

注※1

-a オプションを指定する場合，指定できません。

注※2

増分対象ファイルについては，マニュアル「HiRDB システム運用ガイド」の「RD エリアの自動増分」を参照してください。

(e)　-s セクタ長　～＜符号なし整数＞((1024，2048，4096))

省略値

ダイレクトディスクアクセスを使用する場合：1024

通常ファイルの場合：512

DVD-RAM 装置などの，物理セクタ長が 2,048 バイト及び 4,096 バイトの媒体を扱う装置で HiRDB ファイルシステム領域を使用する場合，最小入出力単位をセクタ長で指定します。媒体の物理セクタ長については，媒体のマニュアルを参照してください。

また，論理セクタ長が 4,096 バイトのディスクを次の方法で使用する場合，4,096 を指定してください。

- ダイレクトディスクアクセスを使用する
- 使用目的に UTL を指定する
- pd_ntfs_cache_disable オペランドに Y を指定し，通常ファイルに作成する

＜規則＞

1. セクタ長に 2,048 を指定した場合，-k オプションには DB 又は UTL のどちらか一方を指定できます。
2. ダイレクトディスクアクセスを使用する場合だけ，セクタ長に 1,024 及び 2,048 を指定できます。
3. pdwork に指定した HiRDB ファイルシステム領域のセクタ長を変更する場合は，HiRDB が停止した状態で行ってください。
4. RD エリア用の HiRDB ファイルシステム領域として使用する場合，RD エリアを作成するときの pdinit 又は pdmod の制御文のページ長には，-s オプションで指定したセクタ長の整数倍の値を指定してください。

   媒体の物理セクタ長が 4,096 バイトの場合，次の指定をしてください。

   ・システムログファイル用の HiRDB ファイルシステム領域として使用する場合，システム定義 pd_log_rec_leng オペランドに 4,096 を指定してください。

   ・ステータスファイル用の HiRDB ファイルシステム領域として使用する場合，pdstsinit コマンドの-l を省略するか，4,096 の倍数を指定してください。
5. 通常ファイルや論理セクタ長が 512 バイトのディスクにも，4,096 を指定できます。

(f)　-i

-n オプションで指定した HiRDB ファイルシステム領域サイズ分の領域を最初から初期化する場合に指定します。ただし，-a オプションを指定して，HiRDB ファイルシステム領域を自動的に拡張する場合は，

ファイル種別や-k オプションの指定値などによって増分領域の初期化可否が異なります。詳細は，-a オプションの説明を参照してください。

-i オプションを省略した場合，HiRDB ファイルシステム領域の管理情報だけを作成します（最大約 3.5 メガバイトの領域が確保されます）。実際に HiRDB ファイルシステム領域を使用したときに，-n オプションで指定したサイズまで拡張していきます。また，ダイレクトディスクアクセス（raw I/O）で，HDP 機能を使用していない場合，-n オプションに指定した容量の領域が保証されるため，ディスクが容量不足になりません。

系切り替え機能で使用する HiRDB ファイルシステム領域を作成する場合，-i オプションを指定してください。-i オプションを指定しないと，HiRDB ファイルシステム領域の拡張とサーバマシンの電源異常が同時に発生した場合にファイルが破壊されることがあります。

＜注意事項＞

- -n オプションで指定した HiRDB ファイルシステム領域サイズが大きいと，初期化するのに時間が掛かります。
- ファイル（NTFS）で-i オプションを省略した場合，-n オプションで指定したサイズまでデータを書き込む前に，ディスク（OS のファイルシステム）が容量不足になることがあります。HDP 機能を使用していない，及びディスクの容量不足を回避したいときは，-i オプションを指定することをお勧めします。なお，ディスクの容量不足が発生すると，HiRDB は次の状態になります。

  ・-k オプションに DB 又は SDB を指定した領域のとき：

  RD エリアが障害閉塞するおそれがあります。

  ・-k オプションに SYS を指定した領域のとき：

  スワップ先があればスワップしますが，スワップ先がないと HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。
- HDP 機能を使用して HiRDB ファイルシステム領域を作成する場合は，-i オプションを指定しないでください。-i オプションを指定すると，-n オプションに指定した容量を初期化して実ディスクを割り当てるため，ボリューム容量を仮想化していない状態になります。

(g) -r

ファイルを初期設定する場合，確認メッセージを表示しないときに指定します。省略した場合，指定したファイルが既にあるときは，初期設定するかどうかの確認メッセージが表示されます。確認メッセージに対して g 以外の文字を入力した場合は，初期設定されません。

ダイレクトディスクアクセス（raw I/O）を使用する場合，-r オプションは指定できません（確認メッセージも表示されません）。

(h) -a

RD エリアの自動増分や作業表を使用する SQL の実行などで HiRDB ファイルシステム領域内の空き領域がなくなった場合，HiRDB ファイルシステム領域を自動的に拡張したいときに指定します。このオプションを指定すると，HiRDB ファイルシステム領域サイズが-n オプションで指定した値に達したとき，自動的に

HiRDB ファイルシステム領域を拡張します。また，システムログファイルの自動拡張機能を適用するシステムログファイルの HiRDB ファイルシステム領域を作成する場合もこのオプションを指定します。RD エリアの自動増分及びシステムログファイルの自動拡張機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

なお，自動的に拡張した HiRDB ファイルシステム領域の初期化可否を，次の表に示します。

表 2–2　自動的に拡張した HiRDB ファイルシステム領域の初期化可否

項番	ファイル種別	-k オプションの指定値	pd_rdarea_expand_format オペランドの指定値※	増分領域の初期化可否
1	ファイル（NTFS）	DB 又は SDB	Y	○
2			N	×
3		WORK	−	×
4		SYS		○
5	ダイレクトディスクアクセス（raw I/O）	DB 又は SDB	−	×
6		WORK		
7		SYS		

（凡例）

○：増分領域を初期化します。

×：増分領域を初期化しません。

−：該当しません。

注※

RD エリアの自動増分が発生した場合に増分領域を初期化するかどうかをシステム共通定義の pd_rdarea_expand_format オペランドで指定します。

Y：増分領域を初期化する

N：増分領域を初期化しない

pd_rdarea_expand_format オペランドについては，マニュアル「HiRDB システム定義」を参照してください。

＜規則＞

1. このオプションは，-k DB，-k SDB，-k SYS，又は-k WORK のときだけ指定できます。
2. -k DB，-k SDB，又は-k SYS の場合，HiRDB ファイルシステム領域内に作成できる HiRDB ファイル数は 1 ファイルです。そのため，-l オプションと-e オプションは指定できません。
3. -k DB，-k SDB，又は-k WORK の場合，-i オプションの指定有無に関係なく，拡張した領域の初期化は行いません。ただし，-k SYS の場合，ファイルの種類がファイル（NTFS）であれば，拡張した領域を初期化し，ファイルの種類がダイレクトディスクアクセス（raw I/O）であれば初期化しません。

4. -a オプションを指定する場合の HiRDB ファイルシステム領域の上限は次のとおりです。

表 2–3　HiRDB ファイルシステム領域の上限

-k オプションの指定値	HiRDB ファイルシステム領域の上限（単位：メガバイト）
DB	65536
SDB	
SYS	102400
WORK	「-n HiRDB ファイルシステム領域サイズ」で示した値を参照してください。

(5)　コマンド引数

(a)　ファイル名　～＜パス名＞((165 文字以内))

初期化するファイルの名称を指定します。ここで指定した名称が，HiRDB ファイルシステム領域名となります。

HiRDB ファイルシステム領域内に作成する HiRDB ファイルは，ファイル名として 30 文字まで指定できます。HiRDB ファイルシステム領域名¥HiRDB ファイル名の上限は 167 文字のため，HiRDB ファイルシステム領域名が 136 文字を超える場合は，HiRDB ファイル名の最大長が小さくなるので注意してください。

HiRDB が自動的に生成する HiRDB ファイルについては，次の HiRDB ファイルシステム領域の最大長を超えないようにしてください。

HiRDB が自動的に生成する HiRDB ファイル	HiRDB ファイルシステム領域の最大長（単位：文字）
作業表用ファイル（システム定義の pdwork オペランドで指定）	141
監査証跡ファイル（システム定義の pd_aud_file_name オペランドで指定）	150
プラグインインデクスの遅延一括作成でのインデクス情報ファイル（システム定義の pd_plugin_ixmk_dir オペランドで指定）	136
システムログの自動ログアンロード機能でのアンロードログファイル（システム定義の pd_log_auto_unload_path オペランドで指定）	136

また，パス名はシンボリックリンクで指定できます。ただし，OS の制限によって，ダイレクトディスクアクセス（raw I/O）を使用する HiRDB ファイルシステム領域に対しては，シンボリックリンクを使用できません。なお，シンボリックリンクは，マニュアル「HiRDB システム運用ガイド」の「RD エリアを格納している HiRDB ファイルシステム領域を移動する方法」を行う場合だけ使用できます。シンボリックリンクを指定できるかどうかを次の表に示します。

表 2–4　シンボリックリンクの指定可否

ファイル種別	領域の使用目的					
	DB	SDB	SYS	WORK	UTL	SVR
ファイル（NTFS）	○	−	×	×	×	×
ダイレクトディスクアクセス（raw I/O）	×	×	×	×	×	×

（凡例）

○：シンボリックリンクを指定できます。

×：シンボリックリンクを指定できません。

−：該当しません。

(6)　規則

1. pdfmkfs コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdfmkfs コマンドは，HiRDB ファイルシステム領域を初期設定するサーバマシンごとに実行してください。

(7)　注意事項

1. pdfmkfs コマンドのリターンコードを次に示します。

   0：正常終了

   1：警告終了（エラーメッセージが出力されたデバイスの HiRDB ファイルシステム領域の作成でエラーが発生）

   255 又は-1：異常終了

   リターンコードが 0 以外の場合，初期設定が正常に完了していないおそれがあります。エラーの要因を取り除いた後，再度コマンドを実行してください。

   なお，使用しているシェルによっては-1 になります。
2. 初期化のとき，ディスクボリューム又はパーティションの容量より大きい値を指定すると，そのパーティションに物理的に続くパーティションを破壊することがあります。したがって，ディスクボリューム又はパーティションの容量より大きな値は指定しないでください。
3. HiRDB ファイルシステム領域名，HiRDB ファイル名，及び物理ファイル名の関係を次に示します。



4. pdfmkfs コマンドの結果は，pdfstatfs コマンドで確認できます。

5. -n，-l，及び-e オプションに指定する値は，使用目的によって次のマニュアルを参照してください。
   - 使用目的が SYS の場合
     マニュアル「HiRDB システム導入・設計ガイド」の「システムファイルの容量の見積もり」
   - 使用目的が DB の場合
     マニュアル「HiRDB システム導入・設計ガイド」の「RD エリアの容量の見積もり」
   - 使用目的が WORK の場合
     マニュアル「HiRDB システム導入・設計ガイド」の「作業表用ファイルの容量の見積もり」，及び
     マニュアル「HiRDB システム定義」の「pdwork オペランド」
   - 使用目的が UTL の場合
     「データベース再編成ユティリティ（pdrorg）」，及び「データベース複写ユティリティ（pdcopy）」
   - 使用目的が SVR の場合
     HiRDB ファイルシステム領域に作成する HiRDB ファイルの使用目的によって，上記マニュアルを参照してください。
6. 使用目的が WORK の場合，システム定義の pd_watch_resource と pdwork_wrn_pnt の指定によって，初期化した HiRDB ファイルシステム領域に対する使用率警告メッセージを出力できます。
7. HiRDB 管理者のシステム資源の制限値は，-n オプション指定値より大きな値，又は無制限に設定してください。
8. HiRDB ファイルシステム領域の所有者及びアクセス権の変更については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(8)　HiRDB ファイルシステム領域の管理領域サイズ

HiRDB ファイルシステム領域には，管理領域が割り当てられます。割り当てる管理領域の大きさは，OS の種別，-n，-l，-e，-s オプション，使用するファイル種別などによって変わります。管理領域の計算式を次に示します。なお，固定管理部と可変管理部は，該当する HiRDB ファイルシステム領域を使用している間は，メモリに読み込まれるため，各プロセスのメモリ消費量が増えることがあります。

```
管理領域（バイト）
＝固定管理部＋可変管理部＋HiRDBファイル管理部×a
```

- -n 2047 以下，かつ-a 指定なしの場合
  固定管理部＝ 4096
  可変管理部＝↑(20 ＋ 4×a)÷c↑×c ＋↑(48 ＋ 16×(a ＋ b))÷c↑×c
  HiRDB ファイル管理部＝↑320÷c↑×c
- -n 2048 以上又は-k WORK -a 指定の場合
  固定管理部＝ 4096
  可変管理部＝↑(20 ＋ 4×a)÷c↑×c ＋↑(56 ＋ 24×(a ＋ b))÷c↑×c
  HiRDB ファイル管理部＝↑480÷c↑×c

可変記号の説明

a：-l オプションの指定値

b：-e オプションの指定値（省略時は 0）

c：-s オプションの指定値（省略した場合は 512）

なお，-k DB，-k SDB，又は-k SYS の場合に-a オプションを指定すると，割り当てられる管理領域の大きさは固定となります。管理領域の値を次に示します。

表 2–5　割り当てられる管理領域の大きさ（-k DB，-k SDB，又は-k SYS の場合）

ファイルの種類	セクタ長（-s 指定値）	管理領域サイズ（単位：キロバイト）
通常のファイル	512	5.5
ダイレクトディスクアクセス（raw I/O）	1024	7
	2048	10
	4096	16

(9)　使用例

次の HiRDB ファイルシステム領域を初期設定します。

- c:¥sysarea¥rdsys011

```
pdfmkfs -n 25   .....................................1
        -l 10   .....................................2
        -e 5    .....................................3
        c:¥sysarea¥rdsys011   .......................4
```

[説明]

1. HiRDB ファイルシステム領域サイズ　**25 メガバイト**
2. 最大ファイル数　**10 個**
3. 最大増分回数　**5 回**
4. ファイル名　**c:¥sysarea¥rdsys011**

2.36　pdfrm（HiRDB ファイルの削除）

2.36.1　pdfrm の形式と規則

(1)　機能

指定した HiRDB ファイルを HiRDB ファイルシステム領域から削除します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdfrm 〔-i〕 HiRDBファイルシステム領域名¥HiRDBファイル名
```

(4)　オプション

(a)　-i

HiRDB ファイルを削除する前に，確認のメッセージを表示したい場合に指定します。確認のメッセージが表示された場合，y を入力してください。

(5)　コマンド引数

(a)　HiRDB ファイルシステム領域名¥HiRDB ファイル名

削除する HiRDB ファイルがある HiRDB ファイルシステム領域名と，HiRDB ファイルを指定します。

HiRDB ファイルシステム領域名　～＜パス名＞((165 文字以内))

HiRDB ファイルシステム領域名を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞

HiRDB ファイル名を指定します。

(6)　規則

1. pdfrm コマンドは，削除対象の HiRDB ファイルがある HiRDB ファイルシステム領域の使用目的が UTL 以外の場合，その HiRDB ファイルがある HiRDB ファイルシステム領域を，稼働中の HiRDB が使用中のとき（その中にある HiRDB ファイルを一度でもオープンしたことがあるとき）は実行できません。使用目的が UTL の場合は，HiRDB ファイルシステム領域の使用に関係なく実行できますが，指定した HiRDB ファイルを稼働中の HiRDB が使用中（ほかのプロセスでオープン中）のときは実行できません。

2. pdfrm コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。
3. 指定した HiRDB ファイルが使用中（ほかのプロセスでオープン中）の場合は，削除できません。
4. -i オプションを指定した場合，確認のメッセージに対して y 以外の文字を入力したときは，HiRDB ファイルは削除されません。
5. 共用 RD エリアとして使用している HiRDB ファイルシステム領域に対して実行する場合，更新可能バックエンドサーバがあるサーバマシンから実行してください。

(7) 注意事項

1. pdfrm コマンドのリターンコードを次に示します。

   0：正常終了

   255 又は-1：異常終了

   リターンコードが 0 以外の場合，HiRDB ファイルを正常に削除できていないおそれがあります。エラーの要因を取り除いた後，指定した HiRDB ファイルシステム領域名¥HiRDB ファイル名が正しいか pdfls コマンドで確認し，再度コマンドを実行してください。

   なお，使用しているシェルによっては-1 になります。
2. pdfrm コマンドの結果は，pdfls コマンドで確認できます。
3. pdfrm コマンドは使用目的が UTL の場合を除き，HiRDB ファイルを強制的に削除する非常用のコマンドです。通常はその使用目的に応じてデータベース構成変更ユティリティ，pdlogrm コマンド，pdstsrm コマンドなどの上位のコマンドを使用して HiRDB ファイルを削除してください。
4. pdfrm コマンドは RD エリアを構成している HiRDB ファイルでも強制的に削除できるため，実行するときには十分注意してください。

2.37 pdfrstr（HiRDB ファイルシステムのリストア）

2.37.1 pdfrstr の形式と規則

(1) 機能

pdfbkup コマンドでバックアップした HiRDB ファイルシステムを格納（リストア）します。また，HiRDB ファイル単位にも格納できます。

HiRDB ファイルシステム領域の状態を表示（pdfstatfs コマンドを入力）した時，増分回数，又は不連続な空き領域の総数が算出された場合，次の処置をすると，増分された領域，又は不連続な空き領域を統合し，HiRDB ファイルシステム領域を有効に利用できます。

- HiRDB ファイルシステムの退避（pdfbkup コマンド）
- HiRDB ファイルシステム領域の初期設定（pdfmkfs コマンド）
- 退避した HiRDB ファイルシステムの，初期設定された HiRDB ファイルシステムへの格納（pdfrstr コマンド）

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdfrstr 〔-y〕 〔- {t | o | r} 〕
          バックアップファイル名〔¥HiRDBファイル名〕
          HiRDBファイルシステム領域名
```

(4) オプション

(a) -y

HiRDB ファイルシステムをリストアするときに，ユーザの確認なしで実行する場合に指定します。省略した場合，確認してから実行します。

(b) - {t | o | r}

格納先の HiRDB ファイルシステム領域に，リストアするバックアップファイル中にある HiRDB ファイルと同じ名称の HiRDB ファイルがある場合の処理を指定します。

-t：

タイムスタンプを比較し，格納先の HiRDB の HiRDB ファイルが更新されている場合は，格納しません。

-o：

バックアップファイル中の HiRDB ファイルを格納しません。

-r：

タイムスタンプに関係なく，バックアップファイル中の HiRDB ファイルを格納します。OS の時刻を遅らせたことによって，バックアップファイル中にある HiRDB ファイルのタイムスタンプが，格納先の HiRDB ファイルシステム領域にある HiRDB ファイルより古くなっていても，データ自体はバックアップファイル中にある HiRDB ファイルの方が新しいことを保証されている場合などに使用します。

(5) コマンド引数

(a) バックアップファイル名〔¥HiRDB ファイル名〕

pdfbkup コマンドで退避したバックアップファイル名を指定します。HiRDB ファイル単位に格納する場合は，HiRDB ファイル名も指定します。

バックアップファイル名

バックアップファイル名を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞

HiRDB ファイル名を指定します。

(b) HiRDB ファイルシステム領域名　～＜パス名＞((165 文字以内))

格納先の HiRDB ファイルシステム領域名を指定します。

(6) 規則

1. pdfrstr コマンドは，HiRDB が停止中のときだけ実行できます。
2. pdfrstr コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。
3. 格納先の HiRDB ファイルシステム領域にバックアップファイルにない HiRDB ファイルがある場合，その HiRDB ファイルは pdfrstr コマンド実行後も残ります。
4. -y オプションを省略した場合は，確認メッセージが表示されます。このとき，確認メッセージに対してＧ以外の文字を指定した場合は，HiRDB ファイルシステムは格納されません。
5. pdfrstr コマンド実行中に異常終了した場合，格納先の HiRDB ファイルシステム領域が，破壊される可能性があります。このため，格納先の HiRDB ファイルシステム領域内に HiRDB ファイルがある場合，pdfbkup コマンドで退避しておいてください。
6. 退避した HiRDB ファイルのレコード長が，回復先の HiRDB ファイルシステム領域のセクタ長の整数倍でない場合，HiRDB ファイルは回復できません。HiRDB ファイルのレコード長は，退避する前に pdfls

コマンドで確認しておいてください。また，回復先の HiRDB ファイルシステム領域のセクタ長は，pdfstatfs コマンドで確認できます。

(7) 注意事項

1. pdfrstr コマンドの実行結果は，コマンド実行時のリターンコード又はエラーメッセージの有無で確認できます。
2. pdfrstr コマンドのリターンコードを次に示します。

   0：正常終了

   1：警告終了（一部の HiRDB ファイルの処理がエラー終了）

   255 又は-1：異常終了

   リターンコードが 0 以外の場合，バックアップした HiRDB ファイルシステムの格納が正常に完了していないおそれがあります。エラーの要因を取り除いた後，再度コマンドを実行してください。

   なお，使用しているシェルによっては-1 になります。
3. ユティリティ用 HiRDB ファイルシステム領域に格納する場合，ユティリティ実行中は格納できません。ユティリティ終了後，実行してください。
4. OS の時刻を遅らせる場合の pdfrstr コマンドの実行への影響については，マニュアル「HiRDB システム運用ガイド」を参照してください。

2.38　pdfstatfs（HiRDB ファイルシステム領域の状態表示）

2.38.1　pdfstatfs の形式と規則

(1)　機能

HiRDB ファイルシステム領域の状態を表示します。

(2)　実行者

権限がないユーザでも実行できます。

(3)　形式

```
pdfstatfs　{〔- {c｜d}〕〔 {-S｜-x 〔-y〕}〕　〔-b〕 HiRDBファイルシステム領域名
          ｜-A 〔-x 〔-y〕〕 HiRDBファイルシステム領域名}
```

(4)　オプション

(a)　- {c ｜ d}

c：

HiRDB ファイルシステム領域中に割り当てられた領域の最大使用量，最大使用ファイル数，及び最大使用増分回数を 0 にします。HiRDB ファイルシステム領域の使用目的が WORK，及び UTL の場合に有効です。

d：

HiRDB ファイルシステム領域中に割り当てられた領域の最大使用量，最大使用ファイル数，及び最大使用増分回数を表示します。

(b)　-S

HiRDB ファイルシステム領域中の，空き領域の状態を表示します。

(c)　-x

DAT 形式で情報を表示する場合に指定します。

(d)　-y

-x オプション指定時に，ヘッダを付ける場合に指定します。

(e)　-b

HiRDB ファイルシステム領域のセクタ長を表示します。

(f)　-A

HiRDB ファイルシステム領域のすべての情報を表示します。

-d，-S，及び-b オプション指定時に表示する情報に加えて，次の情報を表示します。

- HiRDB ファイルシステム領域の最大増分回数を変更する際の上限値
- HiRDB ファイルシステム領域の初期化有無
- HiRDB ファイルシステム領域の作成先のファイル種別

(5)　コマンド引数

(a)　HiRDB ファイルシステム領域名　～＜パス名＞((165 文字以内))

HiRDB ファイルシステム領域名を指定します。

(6)　規則

1. pdfstatfs コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdfstatfs コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。
3. 共用 RD エリアとして使用している HiRDB ファイルシステム領域に対して実行する場合，更新可能バックエンドサーバがあるサーバマシンから実行してください。

(7)　注意事項

1. pdfstatfs コマンド実行時のリターンコードは，0 の場合は正常終了，255 の場合は異常終了となります。リターンコード 255 は，使用しているシェルによっては-1 になります
2. -c オプションで最大使用量，最大使用ファイル数，及び最大使用増分回数を 0 にしてから，その後-d オプションで表示すると，その間に使用した HiRDB ファイルシステム領域の最大使用量，最大使用ファイル数，及び最大使用増分回数を取得できます。
3. -c オプションで最大使用量，最大使用ファイル数，及び最大使用増分回数を 0 にできるのは，HiRDB ファイルシステム領域の使用目的が WORK，及び UTL の場合だけです。

   HiRDB ファイルシステム領域の使用目的が SYS，DB，及び SVR の場合，-c オプションで最大使用量，最大使用ファイル数，及び最大使用増分回数を 0 にはできません。ただし，pdfzeroinit コマンドで使用済み領域の初期化を行うと最大使用量，最大使用ファイル数，及び最大使用増分回数は 0 になります。

   したがって，最大使用量，最大使用ファイル数，及び最大使用増分回数は pdfmkfs コマンド実行時点又は pdfzeroinit コマンド実行時点からの累積を表示します。

4. HDP 機能を使用している場合は，-c オプションを指定しないでください。-c オプションを指定すると，pdfzeroinit コマンドで使用済みの領域が初期化できなくなります。

(8)　出力形式

(a)　-x 及び-y オプション省略時

```
 user area capacity         aa...a[kB]
 remain user area capacity  bb...b[kB]
[peak capacity              mm...m[kB]]
 available file size        cc...c[kB]
 available file count       dd...d
 current file count         ee...e
 remain file count          ff...f
 free area count            gg...g
 available expand count     hh...h
 current expand count       ii...i
[sector size                qq...q[Byte]]
 initialize area kind       jj...j
 initialize user id         kk...k
 initialize time            ll...l
 area auto expand           rr...r
[peak file count            ss...s]
[peak expand count          tt...t]
[limit expand count         uu...u]
[clear option               v]
[file system type           www]
[area type                  xx...x]
[***** HiRDB file system area space information *****]
[offset[kB]     size[kB]  used/unused]
[   nn...n      oo...o    pp...p]
```

[説明]

aa...a：

HiRDB ファイルシステム領域中で，ユーザに割り当てられた領域の総容量（単位：キロバイト）（10 進数 10 けた以内で右詰め表示）

bb...b：

ユーザに割り当てられた領域の中で，未使用領域（HiRDB ファイルとして割り当てられていない領域）の容量（単位：キロバイト）（10 進数 10 けた以内で右詰め表示）

cc...c：

一つのエクステント内に確保できる 1HiRDB ファイルの最大容量（単位：キロバイト）（10 進数 10 けた以内で右詰め表示）

dd...d：

pdfmkfs コマンドの-l オプションで指定した，作成できるファイルの上限数（10 進数 4 けた以内で右詰め表示）

ee...e：

作成済みの HiRDB ファイルの数（10 進数 4 けた以内で右詰め表示）

ff...f：

作成できる HiRDB ファイルの数（最大作成可能ファイル数−作成済みファイル数）（10 進数 4 けた以内で右詰め表示）

gg...g：

不連続な空き領域の総数（10 進数 5 けた以内で右詰め表示）

hh...h：

pdfmkfs コマンドの-e オプションで指定した，HiRDB ファイルシステム領域の増分回数の上限値（10 進数 5 けた以内で右詰め表示）

ただし，pdfchfs コマンドで増分回数の上限値を変更していた場合は，pdfchfs コマンドの-e オプションで指定した値が表示されます。

ii...i：

HiRDB ファイルシステム領域の増分回数合計値（10 進数 5 けた以内で右詰め表示）

jj...j：

pdfmkfs コマンドの-k オプションで指定した，HiRDB ファイルシステム領域の使用目的（9 文字以内）

使用目的が DB の場合，その領域が LOB 用 RD エリアを構成する HiRDB ファイルを作成したことがあるかどうかが表示されます。

DB：LOB 用 RD エリアを構成する HiRDB ファイルを作成したことがある HiRDB ファイルシステム領域

DB(NOLOB)：LOB 用 RD エリアを構成する HiRDB ファイルを作成したことがない HiRDB ファイルシステム領域

バージョン 09-50 より前の HiRDB で，かつ使用目的 NUTL で作成した HiRDB ファイルシステム領域の場合，NUTL が表示されます。

kk...k：

HiRDB ファイルシステム領域を初期化したユーザのログイン名称

ll...l：

HiRDB ファイルシステム領域を初期化した時刻（24 文字）

曜日　月　日　時：　分：　秒　年（西暦）の形式で表示されます。

2005 年 1 月 7 日(金)9 時 5 分 8 秒の場合の出力例を次に示します。

initialize time Fri Jan 07 09:05:08 2005

mm...m：

現時点でのユーザ最大使用量（10 進数 10 けた以内で右詰め表示）。-d 又は-A オプションを指定したときだけ表示されます。

nn...n：

HiRDB ファイルシステム領域中の，使用中領域又は未使用領域の先頭オフセット（右詰め表示）。-S 又は-A オプションを指定したときだけ表示されます。

oo...o：

HiRDB ファイルシステム領域中の，使用中又は未使用の領域サイズ（右詰め表示）。-S 又は-A オプションを指定したときだけ表示されます。

pp...p：

使用中又は未使用の表示。

used：使用中領域

unused：未使用領域

-S 又は-A オプションを指定したときだけ表示されます。

qq...q：

HiRDB ファイルシステム領域のセクタ長（10 進数 4 けた以内で右詰め表示）。

ファイルの場合は 512，ダイレクトディスクアクセス（raw I/O）の場合は pdfmkfs コマンドの-s オプションで指定したセクタ長（省略時は 1,024）となります。

-b 又は-A オプションを指定したときだけ表示されます。

rr...r：

HiRDB ファイルシステム領域を自動的に拡張する機能の適用，非適用の表示。

RD エリアの自動増分時に HiRDB ファイルシステム領域内の空き領域がなくなった場合，HiRDB ファイルシステム領域を自動的に拡張するかどうかを表示します。

USE：適用

NOUSE：非適用

USE が表示されている場合，user area capacity の値は変動します。

ss...s：

現時点でのユーザ最大使用ファイル数（10 進数 4 けた以内で右詰め表示）。-d 又は-A オプションを指定したときだけ表示されます。

tt...t：

現時点でのユーザ最大使用増分回数（10 進数 5 けた以内で右詰め表示）。-d 又は-A オプションを指定したときだけ表示されます。

uu...u：

HiRDB ファイルシステム領域の最大増分回数を変更する際の上限値（10 進数 5 けた以内で右詰め表示）。

pdfmkfs コマンドの-e オプションで指定した最大増分回数です。

pdfchfs コマンドの-e オプションで指定できる上限値となります。

-A オプションを指定したときだけ表示されます。

v：

HiRDB ファイルシステム領域の初期化有無。

pdfmkfs コマンドを実行した際の-i オプション指定の有無を表します。

Y：-i オプション指定あり

N：-i オプション指定なし

バージョン 09-50 より前の HiRDB で作成した HiRDB ファイルシステム領域の場合は，"-"が表示されます。

-A オプションを指定したときだけ表示されます。

www：

HiRDB ファイルシステム領域の作成先のファイル種別。

RAW：ダイレクトディスクアクセス（rawI/O）

REG：ファイル

バージョン 09-50 より前の HiRDB で作成した HiRDB ファイルシステム領域の場合は，"---"が表示されます。

-A オプションを指定したときだけ表示されます。

xx...x：

HiRDB ファイルシステム領域種別。

NORMAL：通常の HiRDB ファイルシステム領域

UNSUPPORTED：未サポートの HiRDB ファイルシステム領域

-A オプションを指定したときだけ表示されます。

(b)　DAT 形式での出力形式（-x 及び-y オプション指定時）

```
”USER_AREA_CAPACITY”,”REMAIN_USER_AREA_CAPACITY”,”PEAK_CAPACITY”,
”AVAILABLE_FILE_SIZE”,”AVAILABLE_FILE_COUNT”,”CURRENT_FILE_COUNT”,
”REMAIN_FILE_COUNT”,”FREE_AREA_COUNT”,”AVAILABLE_EXPAND_COUNT”,
”CURRENT_EXPAND_COUNT”,”SECTOR_SIZE”,”INITIALIZE_AREA_KIND”,
”INITIALIZE_USER_ID”,”INITIALIZE_TIME”,”AREA_AUTO_EXPAND”,
”PEAK_FILE_COUNT”,”PEAK_EXPAND_COUNT”,”LIMIT_EXPAND_COUNT”,
”CLEAR_OPTION”,”FILE_SYSTEM_TYPE”,”AREA_TYPE” [CR]
aa...a,bb...b,mm...m,
cc...c,dd...d,ee...e,
ff...f,gg...g,hh...h,
ii...i,qq...q,”jj...j”,
”kk...k”,”ll...l”,”rr...r”,
ss...s,tt...t,uu...u,
”v”,”www”,”xx...x” [CR]
```

注

[CR]は，改行を表します。

［DAT 形式での出力規則］

1. DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。
2. セクタ長，現時点でのユーザ最大使用量，現時点でのユーザ最大使用ファイル数，及び現時点でのユーザ最大使用増分回数は，各情報取得用のオプションを組み合わせて指定した場合だけ表示します。各情報取得用のオプションを指定しなかった場合は，該当する項目は表示しないで，空要素として表示します。

2.39　pdfzeroinit（HiRDB ファイルシステム領域内の使用済み領域の初期化）

2.39.1　pdfzeroinit の形式と規則

(1)　機能

HiRDB ファイルシステム領域内の使用済み領域を初期化します。

なお，pdfzeroinit コマンドは，HDP 機能を使用して HiRDB ファイルシステム領域を仮想化しているときに使います。

HDP 機能を使用した HiRDB ファイルシステム領域の仮想化については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdfzeroinit  HiRDBファイルシステム領域名
```

(4)　コマンド引数

(a)　HiRDB ファイルシステム領域名　～＜パス名＞((1～165 バイト))

使用済み領域を初期化する HiRDB ファイルシステム領域名を指定してください。

(5)　規則

1. pdfzeroinit コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdfzeroinit コマンドは，HiRDB ファイルシステム領域があるサーバマシンごとに実行してください。共用 RD エリアとして使用している HiRDB ファイルシステム領域に対して実行する場合，更新できるバックエンドサーバがあるサーバマシンから実行してください。
3. 指定した HiRDB ファイルシステム領域内に HiRDB ファイルがない場合に実行できます。HiRDB ファイルがある場合は，警告終了します。
4. pdfzeroinit コマンドで初期化する範囲は，HiRDB ファイルシステム領域の管理領域の終端から現時点でのユーザ最大使用量までの範囲です。
5. 指定した HiRDB ファイルシステム領域内にある使用済み領域を初期化した場合，現時点でのユーザ最大使用量，現時点でのユーザ最大使用ファイル数，及び現時点でのユーザ最大使用増分回数は 0 になります。

(6) 注意事項

1. pdfzeroinit コマンドのリターンコードを次に示します。

   0：正常終了

   1：警告終了（使用済み領域の初期化がエラー終了）

   255 又は-1：異常終了

   リターンコードが 0 以外の場合，初期化が正常に完了していないおそれがあります。エラーの要因を取り除いた後，再度コマンドを実行してください。

   なお，使用しているシェルによっては-1 になります。

2. pdfzeroinit コマンドは初期化するサイズによって，書き込み処理に時間が掛かります。なお，業務でアクセス中の HiRDB ファイルシステム領域に対しては，pdfzeroinit コマンドを実行しないでください。業務に影響を与えることがあります。

3. HiRDB ファイルシステム領域の種別が WORK の HiRDB ファイルシステム領域に対して pdfzeroinit コマンドを実行する場合は，HiRDB が停止した状態で実行してください。HiRDB 稼働中に実行すると，作業表用ファイルを必要とする SQL 及びユティリティに影響を与えることがあります。

4. 初期化対象の領域に対して，pdfstatfs -c コマンドを実行しないでください。pdfzeroinit コマンドで使用済みの領域を初期化できなくなります。

2.40　pdhold（RD エリアの閉塞）

2.40.1　pdhold の形式と規則

(1)　機能

指定した RD エリアを閉塞します。ただし，RD エリアを使用中の場合は，使用の完了を待って閉塞します。

-r 及び-W 以外のオプションを指定しない閉塞を，**コマンド閉塞**といいます。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdhold  -r  {RDエリア名〔, RDエリア名〕…|ALL}
        〔-c|-i|-b〔-w〕〔-u〕〕〔-W 実行監視時間〕
```

(4)　オプション

(a)　-r {RD エリア名〔，RD エリア名〕...|ALL}

RD エリア名　～＜識別子＞((1～30))

閉塞する RD エリア名を指定します。

ALL

すべての RD エリアを閉塞する場合に指定します。ただし，ほかのオプションとの組み合わせによって，閉塞される RD エリアは次のようになります。

ほかのオプション	閉塞される RD エリア
-b	リスト用 RD エリア及び一時表用 RD エリアを除くすべての RD エリア
上記以外	マスタディレクトリ用 RD エリアを除くすべての RD エリア

＜規則＞

1. RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。
2. -b オプションを指定した場合はマスタディレクトリ用 RD エリアを指定できますが，それ以外の場合は指定できません。また，-b オプション指定時は，リスト用 RD エリア及び一時表用 RD エリアは指定できません。

3. ユーザ用 RD エリア又はユーザ LOB 用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，かつ pdhold コマンドによる閉塞状態
4. -c オプションを指定して，RD エリア名にデータディクショナリ用 RD エリアとユーザ用 RD エリア又はユーザ LOB 用 RD エリアを同時に指定する場合は，データディクショナリ用 RD エリアを最後に指定してください。

(b) -c

閉塞状態にした後，RD エリアをクローズします。これを「クローズ，コマンド閉塞」といいます。

このオプションを指定しなかった場合は，RD エリアの閉塞だけをします。

(c) -i

閉塞した RD エリアに対して，ユーザからの参照を許可する場合に指定します。これを「参照可能閉塞」といいます。

このオプションを指定しなかった場合は，閉塞した RD エリアへの参照はできません。

(d) -b〔-w〕〔-u〕

バックアップ閉塞にする場合に指定します。

バックアップ閉塞とは，オンライン中でもバックアップを取得できる閉塞状態のことをいいます。次の場合に RD エリアをバックアップ閉塞にします。

- pdcopy の更新可能モードでバックアップを取得する場合
- pdcopy 以外の汎用バックアップツールや，OS のコマンドなどでバックアップを取得する場合
- 二重化された論理ボリュームを切り離す場合

バックアップ閉塞には，次の四つの状態があります。

1. 参照可能バックアップ閉塞（-b 指定）
2. 参照可能バックアップ閉塞（更新 WAIT モード）（-b，-w 指定）
3. 更新可能バックアップ閉塞（-b，-u 指定）
4. 更新可能バックアップ閉塞（WAIT モード）（-b，-w，-u 指定）

1 と 2 を**データベースの静止化**といいます。

各バックアップ閉塞の説明を次に示します。

参照可能バックアップ閉塞：

閉塞解除するまで更新トランザクションをエラーにしてもよい場合，参照可能バックアップ閉塞にします。

＜留意事項＞

1. 参照可能バックアップ閉塞の場合，更新トランザクションとのデッドロックは発生しません。
2. この状態で取得したバックアップは，システムログを使用しなくても，バックアップ取得時点にデータベースを回復できます。また，バックアップ閉塞直前からのシステムログを使用すると，障害発生時点の状態にデータベースを回復できます。
3. 参照可能バックアップ閉塞の RD エリアは参照はできますが，閉塞解除するまでその RD エリアへの更新は SQL エラー（−920）となります。特に，ログレスモードの更新トランザクションがエラーで終了した場合，更新済みの RD エリアがあると，その RD エリアは障害閉塞となるため，注意してください。

参照可能バックアップ閉塞（更新 WAIT モード）：

閉塞解除するまで更新トランザクションを待たせてもよい場合に，参照可能バックアップ閉塞（更新 WAIT モード）にします。

＜留意事項＞

1. この状態で取得したバックアップは，システムログを使用しなくても，バックアップ取得時点にデータベースを回復できます。また，バックアップ閉塞直前からのシステムログを使用すると，障害発生時点の状態にデータベースを回復できます。
2. 参照可能バックアップ閉塞（更新 WAIT モード）の RD エリアは参照できますが，閉塞解除するまでその RD エリアへの更新は，バックアップ閉塞の RD エリアの排他で待ち状態となります。したがって，クライアント環境定義の PDLCKWAITTIME（省略時はシステム定義の pd_lck_wait_timeout），及び PDCWAITTIME には，バックアップ閉塞中の時間以上を設定する必要があります。タイムアウトとなった場合，更新トランザクションは SQL エラー（-770 や-732）となります。
3. 参照可能バックアップ閉塞（更新 WAIT モード）にした場合，更新トランザクションとの間で，デッドロックが発生することがあります。システム定義に pd_deadlock_priority_use=Y を指定し，更に pd_command_deadlock_priority オペランドにデッドロックプライオリティ値を指定することで，デッドロック発生時に更新トランザクションと運用コマンドのどちらを優先させるかを選択できます。
4. デッドロックが発生した場合，pdhold コマンドが取得したバックアップ閉塞の排他をすべて解除し，再度同じバックアップ閉塞処理を行います。このリトライ処理は，最大 5 回繰り返します。5 回目のリトライ処理でもデッドロックが発生する場合は，pdhold コマンドが掛けたバックアップ閉塞をすべて解除し，エラー終了します。
5. ログレスモードの更新トランザクションがタイムアウト又はデッドロックでエラー終了した場合は，既に更新した RD エリアがあると，その RD エリアは障害閉塞となるため，注意してください。

更新可能バックアップ閉塞：

すぐにバックアップ閉塞にしたい場合で，かつバックアップ閉塞中もその RD エリアに対してアクセスしたいときに更新可能バックアップ閉塞にします。ほかのコマンドが実行中の場合，そのコマンドが完了した後，更新可能バックアップ閉塞にする pdhold コマンドが実行できるようになります。

＜留意事項＞

1. 更新可能バックアップ閉塞にした場合，その RD エリアに対して参照及び更新ができます。
2. 更新可能バックアップ閉塞は，閉塞解除するまで更新バッファのデータベース反映時に，ページの物理ログが出力されます。このため，次の点に注意してください。

   (a) システムログファイルの容量が不足していないか確認してください。

   (b) 大量更新中には，更新可能バックアップ閉塞にしないでください（プラグインインデクスの更新も大量更新となります）。

   (c) バックアップが完了したら，すぐに更新可能バックアップ閉塞を解除してください。
3. 更新トランザクション実行中でも，pdhold コマンドを実行すると更新可能バックアップ閉塞状態にできます。
4. この状態で取得したバックアップを使用してデータベースを回復するときは，バックアップ及び前回のシンクポイント以降のシステムログが必要になります。ただし，前回のシンクポイント以降に更新前ログ取得モード又はログレスモードの更新トランザクションを実行している場合は，このバックアップでは回復できません。
5. 更新可能バックアップ閉塞中に共用 RD エリアを更新すると，物理ログが出力されます。そのため，更新可能バックアップ閉塞状態以外の共用 RD エリアに対する更新に比べ，出力されるログの量が増加します。更新可能バックアップ閉塞状態の共用 RD エリアのバックアップを取得する場合は，更新業務を実行しない，又は，更新量を減らすことをお勧めします。
6. 更新可能バックアップ閉塞中の RD エリアに対する更新トランザクションが更新前ログ取得モード又はログレスモードの場合，バックアップ閉塞の RD エリアの排他で待ち状態となります。この場合，クライアント環境定義の PDLCKWAITTIME（省略時はシステム定義の pd_lck_wait_timeout），及び PDCWAITTIME には，バックアップ閉塞中の時間以上を設定する必要があります。タイムアウトとなった場合，更新前ログ取得モード又はログレスモードの更新トランザクションはエラーとなります。
7. ログレスモードの更新トランザクションがタイムアウトでエラー終了した場合，更新済みの RD エリアがあると，その RD エリアは障害閉塞となるため，注意してください。

更新可能バックアップ閉塞（WAIT モード）：

更新トランザクションの完了を待ってバックアップ閉塞にしてもよい場合で，かつバックアップ閉塞中にその RD エリアにアクセスしたいときに更新可能バックアップ閉塞（WAIT モード）にします。

＜留意事項＞

1. 更新可能バックアップ閉塞（WAIT モード）中に，更新バッファの内容が RD エリアに反映された場合，バックアップ閉塞解除時に警告メッセージ（KFPH00157-W）を出力します。
2. 更新可能バックアップ閉塞（WAIT モード）にした場合，その RD エリアに対しての参照及び更新ができます。

3. 更新可能バックアップ閉塞（WAIT モード）は，閉塞解除するまで更新バッファのデータベース反映時に，ページの物理ログが出力されます。このため，次の点に注意してください。

   (a) システムログファイルの容量が不足していないか確認してください。

   (b) 大量更新中には，更新可能バックアップ閉塞にしないでください（プラグインインデクスの更新も大量更新となります）。

   (c) バックアップが完了したら，すぐに更新可能バックアップ閉塞を解除してください。

4. 更新可能バックアップ閉塞（WAIT モード）中に取得したバックアップを使用してデータベースを回復する場合，バックアップ閉塞解除時に警告メッセージ（KFPH00157-W）が出力されていなければ，システムログがなくてもバックアップ閉塞時点に回復できます。更新可能バックアップ閉塞（WAIT モード）を解除した以降の任意の時点に回復する場合，RD エリア間で整合性がとれた状態に回復する場合，又は警告メッセージ（KFPH00157-W）が出力された場合は，バックアップ閉塞直前からのシステムログを使用して回復する必要があります。ただし，-M s 指定の pdcopy でバックアップを取得した場合，警告メッセージ（KFPH00157-W）の有無に関係なく，前回のシンクポイント以降のシステムログが必要となります。

5. 更新可能バックアップ閉塞中の RD エリアに対する更新トランザクションが更新前ログ取得モード又はログレスモードの場合，バックアップ閉塞の RD エリアの排他で待ち状態となります。この場合，クライアント環境定義の PDLCKWAITTIME（省略時はシステム定義の pd_lck_wait_timeout），及び PDCWAITTIME には，バックアップ閉塞中の時間以上を設定する必要があります。タイムアウトとなった場合，更新前ログ取得モード又はログレスモードの更新トランザクションはエラーとなります。

6. 更新可能バックアップ閉塞（WAIT モード）にした場合，更新前ログ取得モード又はログレスモードの更新トランザクションとの間でデッドロックが発生することがあります。システム定義に pd_deadlock_priority_use=Y を指定し，更に pd_command_deadlock_priority オペランドにデッドロックプライオリティ値を指定することで，デッドロック発生時に更新トランザクションと運用コマンドのどちらを優先させるかを選択できます。

7. デッドロックが発生した場合，pdhold コマンドが取得したバックアップ閉塞の排他をすべて解除し，再度同じバックアップ閉塞処理を行います。このリトライ処理は，最大 5 回繰り返します。5 回目のリトライ処理でもデッドロックが発生する場合は，pdhold コマンドが掛けたバックアップ閉塞をすべて解除し，エラー終了します。

8. ログレスモードの更新トランザクションがタイムアウト又はデッドロックでエラー終了した場合，更新済みの RD エリアがあると，その RD エリアは障害閉塞となるため，注意してください。

9. 更新可能バックアップ閉塞中に共用 RD エリアを更新すると，物理ログが出力されます。そのため，更新可能バックアップ閉塞状態以外の共用 RD エリアに対する更新に比べ，出力されるログの量が増加します。更新可能バックアップ閉塞状態の共用 RD エリアのバックアップを取得する場合は，更新業務を実行しない，又は，更新量を減らすことをお勧めします。

-w

参照可能バックアップ閉塞（更新 WAIT モード），又は更新可能バックアップ閉塞（WAIT モード）にする場合に指定します。

-u

更新可能バックアップ閉塞にする場合に指定します。

更新可能バックアップ閉塞の RD エリアに対しては，閉塞中でも参照及び更新ができます。ただし，ログレスモードの UAP，ユティリティなどの更新は，閉塞解除されるまで待ち状態となります。

このオプションを省略した場合は，参照可能バックアップ閉塞となります。

＜バックアップ閉塞に共通する留意事項＞

1. バックアップ閉塞中は，pdmod（RD エリアの拡張，再初期化，属性変更），pdload，及び pdrorg（リロード）は実行できません。
2. 参照可能バックアップ閉塞（更新 WAIT モード），更新可能バックアップ閉塞，及び更新可能バックアップ閉塞（WAIT モード）の RD エリアは，リランのときに状態は引き継ぎません。
3. ログレスモードでの運用時に更新可能バックアップ閉塞にする場合は，事前に pdlogswap -d sys -w コマンドでシステムログファイルのスワップ及びシンクポイントの有効化を必ずしてください。しなかった場合に障害が発生したときは，pdrstr で範囲指定で回復することになります。
4. 更新可能バックアップ閉塞でのバックアップ取得中に HiRDB を終了又は再開始した場合，取得していたバックアップは保証されなくなります。この場合，再度バックアップを取得し直してください。
5. 参照可能バックアップ閉塞（更新 WAIT モード）の場合は，pdhold コマンドと更新トランザクションとの間で，また，更新可能バックアップ閉塞の場合は，pdhold コマンドとログレスモードの更新トランザクションとの間でデッドロックが発生することがあります。デッドロックが発生した場合，RD エリアが既に参照可能バックアップ閉塞（更新 WAIT モード），又は更新可能バックアップ閉塞となっていれば，その RD エリアを pdrels コマンドで閉塞解除した後，再度 pdhold コマンドを実行してください。RD エリアが参照可能バックアップ閉塞（更新 WAIT モード），又は更新可能バックアップ閉塞になっていない場合，時間を空けてから再度 pdhold コマンドを実行してください。
6. 参照可能バックアップ閉塞，参照可能バックアップ閉塞（更新 WAIT モード），又は更新可能バックアップ閉塞（WAIT モード）にする場合，更新トランザクションが実行中のときは，その更新トランザクションが終了するまで pdhold コマンドは待ちになります。この待ちでは排他待ち時間を監視しないで，実行中の更新トランザクションの終了を待ち続けます。この待ちの間，同一 RD エリアに対する後続のトランザクションは排他待ちになり，待ち時間がクライアント環境定義の PDLCKWAITTIME（省略時はシステム定義の pd_lck_wait_timeout），又は PDCWAITTIME 以上になるとタイムアウトします。pdhold で実行時間を監視する場合は，-W オプションを指定してください。待ち状態となっている pdhold コマンドを終了したい場合は，pdcancel コマンドでシングルサーバプロセス，又はバックエンドサーバプロセスを終了してください。pdcancel の-u オプションの UAP 識別子には文字列 pdhold を指定します。
7. バックアップ閉塞にする pdhold コマンドが，ほかのコマンドが掛けた資源「RD エリア状態」の排他によって待ち状態になった場合，システム定義の pd_lck_wait_timeout オペランドでのタイムアウトにはなりません。この場合，ほかのコマンドでの排他が解除されるまで待ち状態となります。待ち状態となっている pdhold コマンドを終了したい場合は，pdcancel コマンドでシングルサーバプロセス，又はバックエンドサーバプロセスを終了してください。

8. 参照可能閉塞になっている RD エリアは，バックアップ閉塞にはできません。RD エリアの状態は，pddbls コマンドで確認できます。
9. RD エリアの自動増分を適用している場合，その RD エリアを更新可能バックアップ閉塞，又は更新可能バックアップ閉塞（WAIT モード）にすると，RD エリアの自動増分を抑止します。RD エリアの自動増分の抑止状態を解除したい場合は，pdrels コマンドを実行して，更新可能バックアップ閉塞，及び更新可能バックアップ閉塞（WAIT モード）を解除してください。
10. RD エリアの自動増分を適用している場合，その RD エリアを更新可能バックアップ閉塞，又は更新可能バックアップ閉塞（WAIT モード）にするときは，事前にデータベース状態解析ユティリティで十分な未使用領域があることを確認してください。

(e)　-W 実行監視時間　～<符号なし整数>((0～3600))

pdhold コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pdhold コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdhold コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdhold コマンドは，RD エリアに対してモード EX で，-i 又は-b オプション指定の場合はモード PR で排他制御をします。そのため，指定した RD エリアにほかのトランザクションがアクセスしている場合は，そのトランザクションの終了まで pdhold コマンドは待ち状態となります。ただし，-i 又は-b オプション指定の場合は，更新系トランザクションがアクセスしているときだけ待ち状態となります。RD エリアの閉塞状態とトランザクションとの関係を次の表に示します。

表 2–6　RD エリアの閉塞状態とトランザクションとの関係

RD エリアの閉塞状態	トランザクション		
	参照※1	更新※2	更新ログレス※3
コマンド閉塞	待ち	待ち	待ち
	エラー	エラー	エラー
参照可能閉塞	待ちなし	待ち	待ち
	待ちなし※4	エラー	エラー
参照可能バックアップ閉塞	待ちなし	待ち※5	待ち※5
	待ちなし※4	エラー	エラー

RD エリアの閉塞状態	トランザクション		
	参照※1	更新※2	更新ログレス※3
参照可能バックアップ閉塞（更新 WAIT モード）	待ちなし	待ち※5	待ち※5
	待ちなし※4	待ち	待ち
更新可能バックアップ閉塞	待ちなし	待ちなし	待ちなし
	待ちなし	待ちなし	待ち
更新可能バックアップ閉塞（WAIT モード）	待ちなし	待ち※5	待ち※5
	待ちなし※4	待ちなし	待ち

注

上段は，トランザクションが先に RD エリアに排他を掛けた場合を示しています。

下段は，pdhold コマンドが先に RD エリアに排他を掛けた場合を示しています。

例えば，参照 UAP が先に RD エリアに排他を掛けた場合，その RD エリアを pdhold コマンドでコマンド閉塞にしようとすると，pdhold コマンドが待ちとなります。また，pdhold コマンド（コマンド閉塞）が先に RD エリアに排他を掛けた場合，その RD エリアへの参照 UAP を実行しようとすると，参照 UAP がエラーとなります。

注※1

参照トランザクションとは，RD エリアに SR 又は PR の排他が掛かる SQL を実行するトランザクションのことをいいます。例えば，WITH SHARE 指定のある SELECT 文は SR の排他を掛けるため，参照トランザクションとなります。

注※2

更新トランザクションとは，RD エリアに SU，PU，又は EX の排他が掛かる SQL を実行するトランザクションのことをいいます。例えば，WITH EXCLUSIVE 指定のある SELECT 文は SU の排他を掛けるため，更新トランザクションとなります。

注※3

クライアント環境定義の PDDBLOG に NO を指定した，ログレスモードのトランザクションのことをいいます。

注※4

更新トランザクションによってコマンドが待ち状態の場合は，後続の参照トランザクションも待ちになります。

注※5

排他待ち時間を監視しないで，実行中の更新トランザクションの終了を待ち続けます。この待ちの間，同一 RD エリアに対する後続のトランザクションは排他待ちになり，待ち時間がクライアント環境定義の PDLCKWAITTIME（省略時はシステム定義の pd_lck_wait_timeout），又は PDCWAITTIME 以上になるとタイムアウトします。pdhold で実行時間を監視する場合は，-W オプションを指定してください。待ち状態となっている pdhold コマンドを終了したい場合は，pdcancel

コマンドでシングルサーバプロセス，又はバックエンドサーバプロセスを終了してください。pdcancel の-u オプションの UAP 識別子には文字列 pdhold を指定します。

4. 閉塞処置中のトランザクションの扱いを次の表に示します。なお，閉塞処理中とは，ACCEPT-HOLD（pddbls コマンドでの実行結果の RD エリアの状態）を指します。

表 2–7　閉塞処置中のトランザクションの扱い

閉塞処理中の受付内容	参照トランザクション	更新トランザクション
コマンド閉塞	エラー	エラー
参照可能閉塞	待ち	エラー
参照可能バックアップ閉塞	待ち	エラー
参照可能バックアップ閉塞（更新 WAIT モード）	待ち	待ち
更新可能バックアップ閉塞	待ちなし※	待ちなし※
更新可能バックアップ閉塞（WAIT モード）	待ち	待ち

注※

更新可能バックアップ閉塞は，RD エリアに排他を掛けないため，待ちにはなりません。

5. データ未完状態の改竄防止表が格納されている RD エリアは，参照可能閉塞（-i 指定），及びバックアップ閉塞（-b 指定）にはできません。データ未完状態の改竄防止表に対して，pdrorg で表へのリロードを実行すると，データ未完状態は解除されます。解除された後に，参照可能閉塞，又はバックアップ閉塞にしてください。

6. 共用 RD エリアに対して pdhold コマンドを実行した場合，全バックエンドサーバに排他を掛けます。このため，該当する RD エリアへのアクセスを複数同時に実行している場合，グローバルデッドロックが発生し，タイムアウトする可能性があります。グローバルデッドロックが発生した場合，再度 pdhold コマンドを実行してください。

7. インメモリ RD エリアに pdhold コマンドを実行した場合の HiRDB の処理を次の表に示します。

表 2–8　インメモリ RD エリアに pdhold コマンドを実行した場合の HiRDB の処理

pdhold コマンドの種類	pdhold コマンド実行時の HiRDB の処理
pdhold -b（参照可能バックアップ閉塞）	インメモリデータバッファとインメモリ RD エリアの同期処理を行います（DB 同期状態になります）。 このとき，インメモリ RD エリアは，各オプションの指定に従った閉塞状態になります。
pdhold -b -w（参照可能バックアップ閉塞（更新 WAIT モード））	
pdhold -c	
pdhold -b -u（更新可能バックアップ閉塞）	pdhold コマンドがエラーになります。
pdhold -b -u -w（更新可能バックアップ閉塞（WAIT モード））	
pdhold	インメモリデータバッファとインメモリ RD エリアの同期処理を行いません。
pdhold -i	

pdhold コマンドの種類	pdhold コマンド実行時の HiRDB の処理
	インメモリ RD エリアは，各オプションの指定に従った閉塞状態になります。

(6) 注意事項

1. pdhold コマンドの結果は，pddbls コマンドで確認できます。
2. pdhold コマンドのリターンコードと基本的な対処方法を次に示します。

   0 以外の場合，一部の RD エリアが閉塞していることがあります。その状態では業務を継続できないおそれがあるため，pdhold コマンドに指定した RD エリアを pdrels コマンドで閉塞解除してください。

   0：正常終了

   4：警告終了

   一部の RD エリアの処理がエラー終了しました。

   8：異常終了

   一部の RD エリアの処理がエラー終了していることがあります。

   12：異常終了

   エラーメッセージを表示できない事象が発生しました。

   一部の RD エリアの処理がエラー終了していることがあります。シングルサーバ又はディクショナリサーバがあるホストのイベントログ中のエラーメッセージを参照して，エラーの要因を取り除いた後，再度コマンドを実行してください。イベントログ中にエラーメッセージが出力されていない場合は保守員に連絡してください。
3. 次に示す閉塞状態は，HiRDB（HiRDB/パラレルサーバの場合はユニット）が強制停止した後の再開始時には引き継ぎません。
   - 参照可能バックアップ閉塞（更新 WAIT モード）
   - 更新可能バックアップ閉塞
   - 更新可能バックアップ閉塞（WAIT モード）

2.41　pdinfoget（障害情報の取得と容量見積もり）

2.41.1　pdinfoget の形式と規則

(1)　機能

HiRDB システムで障害が発生した場合，障害の原因を調査するために障害情報の取得，及び取得する情報の容量見積もりをします。

障害情報取得

HiRDB で障害が発生した場合，pdinfoget コマンドを実行することで，コマンドを実行したホストの HiRDB や OS の障害情報を取得します。取得した情報は，初期情報ファイル又は詳細情報ファイルに出力し，実行状況を実行ログに出力します。

障害情報容量見積もり

障害情報を取得する場合，障害情報の出力先に十分なディスクの空き容量がないと，必要な障害情報が取得できない，指定したディスクの構成によってはディスク容量を圧迫してシステムの性能に影響を及ぼすなどのおそれがあります。これを防ぐために，事前に取得する障害情報に必要な容量をメガバイト単位で見積もります。ただし，ここで表示する値は，障害情報の取得実行中に必要なディスク容量の最大値であり，取得後の障害情報の容量とは異なります。

(2)　実行者

HiRDB 管理者が実行できます。なお，HiRDB が出力する情報ファイルや OS の情報ファイルへの参照権限が必要です。

(3)　形式

(a)　障害情報取得

```
pdinfoget -e 初期情報ファイル出力先ディレクトリ名
        〔-d 詳細情報ファイル出力先ディレクトリ名〕
        〔-w ワークディレクトリ名〕
        〔-n〕
        〔-t ネットワーク情報取得監視時間〕
        〔-s イベントログファイル名〕

        〔-c クラッシュダンプファイル名〕
```

(b)　障害情報容量見積もり

```
pdinfoget -m
        〔-n〕
        〔-s イベントログファイル名〕

        〔-c クラッシュダンプファイル名〕
```

(4)　オプション

(a)　-e 初期情報ファイル出力先ディレクトリ名

初期情報ファイル（障害原因の切り分けに必要な情報をまとめたファイル）を出力する既存のディレクトリ名を絶対パスで指定します。ディレクトリ名の長さの上限値を次に示します。

ディレクトリ名の長さ（単位：バイト）

OS 上のパス名の上限値−初期情報ファイル名（最大：66）

指定したディレクトリが存在しない，又は書き込み権限がない場合は，エラーメッセージを出力して処理を打ち切ります。

初期情報ファイル出力先ディレクトリには，初期情報ファイルと実行ログが出力されます。初期情報ファイルはアーカイブファイルであり，複数のファイルが格納されています。それぞれのファイル名を次に示します。

初期情報ファイル：pdinfoget_ホスト名称_init_日時.zip※

ホスト名称の長さが 32 バイト以上の場合，先頭から 32 バイト分をファイル名とします。

実行ログ：pdinfoget_comlog_日時.txt※，

注※

日時は「MMDDhhmmss」（MM：月，DD：日，hh：時，mm：分，ss：秒）です。

ファイルの容量が 2 ギガバイトを超える場合，ファイルを複数に分割して出力します。このとき，ファイル名の末尾に「.」及び連番「.0，.1，.2，.3，.4，……」が付加されます。弊社問い合わせ窓口にすべてのファイルを送付してください。

初期情報ファイルに取得される障害情報については，表「pdinfoget コマンドで取得する障害情報一覧」を参照してください。

(b)　-d 詳細情報ファイル出力先ディレクトリ名

詳細情報ファイル（障害原因の切り分け後，さらに詳細な調査に必要な情報をまとめたファイル）を出力する既存のディレクトリ名を絶対パスで指定します。ディレクトリ名の長さの上限値を次に示します。

ディレクトリ名の長さ（単位：バイト）

OS 上のパス名の上限値−詳細情報ファイル名（最大 68）

指定したディレクトリが存在しない，又は書き込み権限がない場合は，エラーメッセージを出力して処理を打ち切ります。

詳細情報ファイルはアーカイブファイルであり，複数のファイルが格納されています。詳細情報ファイル名を次に示します。

詳細情報ファイル名：pdinfoget_ホスト名称_detail_日時.zip※

ホスト名称の長さが 32 バイト以上の場合，先頭から 32 バイト分をファイル名とします。

注※

日時は「MMDDhhmmss」（MM：月，DD：日，hh：時，mm：分，ss：秒）です。

ファイルの容量が 2 ギガバイトを超える場合，ファイルを複数に分割して出力します。このとき，ファイル名の末尾に「.」及び連番「.0，.1，.2，.3，.4，……」が付加されます。弊社問い合わせ窓口にすべてのファイルを送付してください。

詳細情報ファイルに取得される障害情報については，表「pdinfoget コマンドで取得する障害情報一覧」を参照してください。

(c)　-w ワークディレクトリ名

pdinfoget は障害情報を取得した後，HiRDB・OS コマンドを実行して障害情報をアーカイブ，又は圧縮します。このとき，一時的に作成する作業用のファイルを格納する領域として，既存のディレクトリ名を絶対パスで指定します。ディレクトリ名の長さの上限値を次に示します。

ディレクトリ名の長さ（単位：バイト）

OS 上のパス名の上限値−作業用のファイル名（最大 53）

指定したディレクトリが存在しない，又は書き込み権限がない場合は，エラーメッセージを出力して処理を打ち切ります。

作業用のファイルは，指定したディレクトリ下にワークディレクトリ「pdinfoget_work_日時」を作成し，そこに格納します。既に「pdinfoget_work_日時」ディレクトリが存在する場合は，エラー終了します。

これらのファイル容量を事前に見積もり，十分なディスク容量を用意する必要があります。見積もり式を次に示します。

Max（A，B）

A：-m オプションで表示される初期情報ファイルの見積もり値

B：-m オプションで表示される詳細情報ファイルの見積もり値

このオプションを省略した場合，次に示すディレクトリが仮定されます。

-w オプションの指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	-w オプションに指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※　コマンド実行環境の環境変数の指定

なお，このオプションを省略した場合，障害情報のアーカイブ，又は圧縮時に一時的に作成するファイルは次に示すディレクトリに格納されます。出力先ディレクトリは，それぞれ 1 の優先順位が一番高くなります。

処理	出力先ディレクトリ	ファイル名
アーカイブ	1. 環境変数 TMP で指定したディレクトリ 2. 環境変数 TEMP で指定したディレクトリ 3. 初期情報ファイル出力先ディレクトリ，又は詳細情報ファイル出力先ディレクトリ	AFI*
圧縮	1. 環境変数 TMP で指定したディレクトリ 2. 初期情報ファイル出力先ディレクトリ，又は詳細情報ファイル出力先ディレクトリ	cab*

これらのファイルは，処理の終了時に削除されます。

ただし，処理中にエラーが発生した場合，ファイルが削除されないことがあります。この場合は，ファイルを手動で削除する必要があります。

(d)　-s イベントログファイル名

pdinfoget がデフォルトで取得すると設定しているイベントログファイルと異なるイベントログファイルを取得する必要がある場合，ファイル名を絶対パスで指定します。イベントログファイルを指定するときは，イベントビューアなどでファイルに保存したイベントログを指定します。

pdinfoget がデフォルトで取得するイベントログファイルは，pdinfoget を実行した時点のイベントログからバックアップした次のファイルです。

- アプリケーション：AppEvtBkup.日時.evt※
- システム：SysEvtBkup.日時.evt※

注※

日時は，バックアップ時点の時間が「MMDDhhmmss」（MM：月，DD：日，hh：時，mm：分，ss：秒）の形式で設定されます。

-s オプションで取得するファイルを指定する必要があると考えられるのは，次のような場合です。

- pdinfoget を実行する時期が遅く，イベントログファイルのスワップが行われ，障害が発生した時間帯と，デフォルトで取得対象としているイベントログファイルの時間帯が異なっている場合
  この場合，障害が発生した時間帯と一致するログを持つファイルを，-s オプションで指定します。

(e)　-c クラッシュダンプファイル名

詳細情報として，クラッシュダンプファイルを取得する場合，そのファイル名を絶対パスで指定します。

このオプションが指定されていないと，クラッシュダンプファイルは取得しません。

(f)　-m

取得する障害情報に必要な容量の見積もり値を出力する場合に指定します。見積もり値は標準出力に次の形式で出力されます。

ファイルの種類	表示形式（単位：メガバイト）
初期情報ファイル	init_directory = xxx Mbyte
詳細情報ファイル	detail_directory = yyyy Mbyte
PRF トレース情報ファイル	prf_directory = wwww Mbyte※

（凡例）xxx，yyyy，zzzz，wwww：数値

注※

次に該当する場合は，"0 Mbyte"と表示されます。

- PRF トレース情報ファイルを取得しない場合

-s 又は-c オプションを指定して障害情報を取得する場合，それらの容量も見積もり値に含めるため，-s 又は-c オプションを同時に指定する必要があります。なお，-s 及び-c 以外のオプションが同時に指定されている場合は，エラー終了します。

(g)　-n

PRF トレース情報ファイルの容量の見積もり，取得を行わない場合に指定します。次の場合に指定してください。

- 弊社問い合わせ窓口から指示があった場合（障害の原因調査に支障が出ない場合）
- PRF トレース情報ファイルを別途取得する場合

(h)　-t ネットワーク情報取得監視時間 ((1～3600))《10》

初期情報で収集するネットワーク情報の取得監視時間を秒単位で指定します。次の場合に指定してください。

- 弊社問い合わせ窓口から指示があった場合

表 2–9　pdinfoget コマンドで取得する障害情報一覧

取得する情報項目				取得の有無	初期情報	詳細情報
HiRDB コマンド	HiRDB の状態を知るため，HiRDB コマンドを実行して取得した情報	ユニット，及び内部コンポーネントの状態		○	○	×
		サーバプロセスの状態		○	○	×
		HiRDB サーバのスケジュール状態		○	○	×
		HiRDB サーバの設定情報		○	○	×
		ユニット内の HiRDB サーバの状態		○	○	×
		実行中トランザクション情報		○	○	×
		ユーザ識別子情報		○	○	×
		排他情報		○	○	×
		排他制御用プール使用状況		○	○	×
		内部ロック情報		○	○	×
		セマフォ情報		○	○	×
		メッセージキュー情報		○	○	×
		ユニット及びサーバの状態		○	○	×
		共用メモリ情報		○	○	×
		HiRDB のバージョン情報		○	○	×
		サーバプロセスの通信制御情報		○	○	×
		グローバルバッファの使用情報		○	○	×
		RD エリアの状態表示		○	○	×
		SQL オブジェクト情報		○	○	×
		HiRDB セットアップ情報		○	○	×
		系切り替え機能の情報		○	○	×
		シンクポイントダンプの情報		○	○	×
		システムログファイルの情報		○	○	×
HiRDB 情報ファイル	HiRDB が参照する定義や，結果として出力したファイル	%PDDIR%¥spool 下の全ディレクトリとファイル※1		○	×	○
		%PDDIR%下の構成情報		○	○	×
		%PDDIR%¥spool 下のファイル	コマンドログファイル	○	○	×
			メッセージログファイル	○	○	×

取得する情報項目				取得の有無	初期情報	詳細情報
			エラーログファイル	○	○	×
			リモート系コマンド情報ファイル	○	○	×
			系切り替え機能情報ファイル	○	○	×
			障害時スナップ情報ファイル	○	○	×
			アボート情報ファイル	○	○	×
			cwaitover 情報ファイル	○	○	×
			PRF トレース情報ファイル	○	×	○
		システム定義情報（conf/*，%PDCONFPATH%）		○	○	×
		簡易セットアップツール情報（pdistup/*）	bin	×	×	×
			conf	×	×	×
			ini	×	×	×
			sample	○	○	×
			tmp	○	○	×
			spool	○	○	×
			pdi_log.txt	○	○	×
		HiRDB データベース環境情報（%PDDIR%¥.dbenv¥*）		○	○	×
		HiRDB 内部用ワークディレクトリ下（%PDDIR%¥tmp¥*）		○	×	○
		クライアントトレース情報（%PDCLTPATH%下のファイル）	SQL トレース情報	○	※2	※2
			エラーログファイル	○	○	×
			pdess*ファイル	○	○	×
			UAP 統計情報	○	○	×
		仕様差吸収ライブラリの全情報（%PDDIR%¥UXPLDIR¥*）		○	×	○
		仕様差吸収ライブラリ情報ディレクトリ下のファイル	仕様差吸収ライブラリエラー情報	○	○	×

取得する情報項目			取得の有無	初期情報	詳細情報
		PRF トレース情報ファイル（pd_prf_output_directory オペランドに指定したディレクトリに出力されたファイル）	○	×	○
		PRF トレース情報ファイルの構成情報（pd_prf_output_directory オペランドに指定したディレクトリの構成情報）	○	○	×
OS コマンド	OS 下の状況を知るため，コマンドを実行して取得した情報	プロセス情報	○	○	×
		ディスク情報	○	○	×
		プロセス間通信機能のステータス情報	○	○	×
		CPU 利用率，及びディスク状況	○	○	×
		ホスト名称	○	○	×
		IP configuration	○	○	×
		OS のパフォーマンス情報	○	○	×
		IP 構成情報	○	○	×
		ネットワーク接続やルーティングテーブル インターフェースの統計情報	○	○	×
		レジストリ情報（システム，ユーザ環境変数）	○	○	×
		グループポリシー情報	○	○	×
		Windows Defender 除外設定	○	○	×
		HiRDB サービスの登録情報	○	○	×
		全サービスの稼働情報	○	○	×
		共有ファイル状態情報	○	○	×
		Windows ファイアウォール設定情報	○	○	×
		HiRDB 管理者情報	○	○	×
OS 情報ファイル	OS が参照する定義や，結果として出力したファイル	OS のバージョン情報	○	○	×
		マシン情報	○	○	×
		プロセッサ情報	○	○	×
		メモリ実装情報	○	○	×
		レジストリ情報	○	○	×
		環境変数情報	○	○	×
		ネットワーク情報	○	○	×
		イベントログファイル	○	○	×

取得する情報項目			取得の有無	初期情報	詳細情報
		クラッシュダンプファイル	○	×	○
		Windows ダンプファイル	○	×	○

（凡例）

○：この情報を取得します。

×：この情報を取得しません

注※1

次のディレクトリは取得しません。

・olkfifs

・olkrsfs

注※2

SQL トレース情報（pdsql1.trc，pdsql2.trc，pd2sql1.trc，及び pd2sql2.trc）のファイル容量の合計が 100 メガバイト以下の場合，初期情報ファイルに出力します。100 メガバイトを超える場合，詳細情報ファイルに出力します。

(5) 規則

1. pdinfoget コマンドを実行するためには，次の環境変数が設定されている必要があります。

   PDDIR

   HiRDB 運用ディレクトリを指定します。

   PDCONFPATH

   HiRDB システム定義ファイルを格納するディレクトリを指定します。指定がない場合は，%PDDIR%¥conf が仮定されます。

2. pdinfoget コマンドは，HiRDB の稼働に関係なく実行できます。
3. 初期情報ファイル出力先ディレクトリと詳細情報ファイル出力先ディレクトリは，同じディレクトリを指定しないでください。同じディレクトリ名を指定していると，エラーメッセージを出力して処理を終了します。
4. ディレクトリ名を指定するオプションで，上限値を超えた長さのディレクトリ名を指定すると，ファイルやディレクトリの作成又はコピーでエラーが発生します。このとき，pdinfoget コマンドのリターンコードに 4 又は 8 が返されます。
5. 初期情報ファイル出力先ディレクトリ，又は詳細情報ファイル出力先ディレクトリに，次の表に示すディレクトリが既に存在する場合，エラーメッセージを出力して処理を終了します。

   表 2–10　出力先ディレクトリ下にあるとエラーになるディレクトリ

出力先ディレクトリ	出力先ディレクトリ下にあるとエラーになるディレクトリ
初期情報ファイル出力先ディレクトリ	CLTDIR

出力先ディレクトリ	出力先ディレクトリ下にあるとエラーになるディレクトリ
	PDDIR spool conf .dbenv pdistup uxpldir OSFILE SYSLOG COREINF PRF
詳細情報ファイル出力先ディレクトリ	PDDIR spool tmp uxpldir CLTDIR COREINF PRF

この場合，次のどちらかの対処をしてから再度実行してください。

- 初期情報ファイル出力先ディレクトリ又は詳細情報ファイル出力先ディレクトリに別のディレクトリを指定する
- 指定したディレクトリ内の情報が不要な場合，すべて削除する

6. pdinfoget コマンド実行中でも，次のメッセージを出力した時点で該当する初期情報ファイルの障害情報は取得を完了しているため，取り出すことができます。詳細情報が多く，取得に時間が掛かる場合は，pdinfoget コマンドが終了する前に初期情報ファイルだけを取り出し，使用できます。

```
KFPN10403-I init information file output, file=初期情報ファイルパス名
```

(6) 注意事項

1. pdinfoget コマンドの結果は，%ERRORLEVEL%に設定されます。その意味と対処方法を次に示します。

%ERRORLEVEL%	終了状態	意味と対処方法
0	正常終了	正常に終了しました。 KFPN10451-I が出力されている場合，障害情報のアーカイブ，又は圧縮が完了しませんでしたが，そのまま障害情報として取得してください。
4	警告終了	1. 障害情報取得の場合 一部の情報を取得できませんでした。 環境設定などによって取得できなかった情報がありますが，そのまま障害情報として取得してください。 取得の際，HiRDB コマンドがメッセージログやイベントログに出力したメッセージの対処は不要です。

%ERRORLEVEL%	終了状態	意味と対処方法
		2. 障害情報容量見積もりの場合 -m オプションを指定して，障害情報容量見積もりを出力するときに，エラーが発生しました。
8	異常終了	エラーが発生したため，処理を打ち切りました。 1. オプションの指定方法が誤っています。 オプションを正しく指定して，再度実行してください。 2. 初期情報ファイル出力先ディレクトリと詳細情報ファイル出力先ディレクトリに同じディレクトリを指定しています。 出力先ディレクトリには，異なるディレクトリを指定して，再度実行してください。 3. 初期情報ファイル出力先ディレクトリ，又は詳細情報ファイル出力先ディレクトリのディレクトリ作成に失敗しました。又は，アクセス権限がありません。 指定する出力先ディレクトリに書き込み権限を付加するか，書き込み権限がある別のディレクトリを指定して，再度実行してください。 4. 初期情報ファイル出力先ディレクトリ，又は詳細情報ファイル出力先ディレクトリ下に表「出力先ディレクトリ下にあるとエラーになるディレクトリ」に示すディレクトリがあります。 指定したディレクトリの内容を確認し，情報が不要な場合は削除して再度実行してください。情報が必要な場合は別のディレクトリを指定して再度実行してください。 5. HiRDB 運用ディレクトリ（%PDDIR%）が設定されていません。 HiRDB 運用ディレクトリ（%PDDIR%）を設定して，再度実行してください。

2. 初期情報ファイル及び詳細情報ファイルの出力先には，十分なディスク容量を用意する必要があります。特に，HiRDB の稼働中に，障害情報の出力先ディレクトリに%PDDIR%下のディレクトリを指定して pdinfoget コマンドを実行すると，容量不足のため HiRDB が異常終了することがあるため，注意してください。なお，障害情報取得時に初期情報ファイル，又は詳細情報ファイルの出力先の容量が見積もりより少ない場合は，KFPN00407-W メッセージを出力します。容量不足が発生して圧縮が行われなかった場合，圧縮前のファイルを取得してください。メッセージに出力した空き容量を確保できる出力先がほかにある場合は，障害情報の出力先を変更して，再実行してください。
3. pdinfoget コマンド実行中，アーカイブ及び圧縮対象となるファイルの容量とファイル数を KFPN00451-I メッセージに出力します。ファイルの容量，ファイル数が多い場合は，処理時間が長くなります。
4. pdinfoget コマンドで出力された初期情報ファイル，又は詳細情報ファイルは，ZIP64 フォーマットで圧縮されます。解凍する場合は，ZIP64 フォーマットに対応したツールなどを使用してください。

2.42　pdjarsync（JAR ファイルの操作）

2.42.1　pdjarsync の形式と規則

(1)　機能

JAR ファイルの操作をします。

JAR ファイルの操作には，次のものがあります。

- JAR ファイルのホスト間の再登録（HiRDB/パラレルサーバ限定）
- JAR ファイルの登録
- JAR ファイルの再登録
- JAR ファイルの削除
- JAR ファイルの一覧表示

このコマンドは，Java ストアドプロシジャ又は Java ストアドファンクションを使用するときに使うコマンドです。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdjarsync 〔 {-I | -R | -D | -L} 〕
          〔-u 認可識別子〔, 認可識別子〕…〕
          〔-f JARファイル名〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdjarsync 〔 {-S | -I | -R | -D | -L} 〕
          〔-x ホスト名〔, ホスト名〕…〕
          〔-u 認可識別子〔, 認可識別子〕…〕
          〔-f JARファイル名〕
```

(4) オプション

(a) {-S ｜-I ｜-R ｜-D ｜-L}

JAR ファイルの操作種別を指定します。

-S

HiRDB/パラレルサーバの場合に，JAR ファイルのホスト間の再登録をします。

-x オプションで指定したホストにある JAR ファイルを，HiRDB 環境下のサーバマシンに再登録します。-x オプションを省略した場合，ディクショナリサーバがあるサーバマシンの JAR ファイルを，HiRDB 環境下のサーバマシンに再登録します。

HiRDB サーバの構成を変更した場合（例えば，サーバマシンを増やしたとき），系切り替えをした場合など，-S オプションを指定してください。

-I

指定した JAR ファイルを登録します。

このオプションを指定した場合，-u，-f オプションを必ず指定してください。

指定した JAR ファイルが既に登録されている場合，エラーとなります。

-R

指定した JAR ファイルを再登録します。

指定した JAR ファイルが既に登録されている場合には，登録されている JAR ファイルに上書きします。

このオプションを指定した場合，-u，-f オプションを必ず指定してください。

-D

指定した JAR ファイルを削除します。

このオプションを指定した場合，-u，-f オプションを必ず指定してください。

指定した JAR ファイルが HiRDB に登録されていない場合，エラーとなります。

pdjarsync コマンドが異常終了した場合，書き込み中の JAR ファイルが残ることがあります。この場合，-D オプションを指定して，残った JAR ファイルを削除してください。

-L

登録されている JAR ファイルの一覧を表示します。

JAR ファイルの操作種別と他オプションとの関係を次の表に示します。

表 2–11　JAR ファイルの操作種別と他オプションとの関係

JAR ファイルの操作種別	他オプション		
	-x	-u	-f
-S	○※	○	○
-I	○	◎	◎
-R	○	◎	◎

JAR ファイルの操作種別	他オプション		
	-x	-u	-f
-D	○	◎	◎
-L	○	○	○

（凡例）

◎：必ず指定してください。

○：任意に指定できます。

注※

-x オプションを指定する場合は，ホスト名を一つだけ指定してください。

(b)　-x ホスト名〔，ホスト名〕...

-S オプションを指定した場合，再登録の起点となるホスト名を一つだけ指定します。省略した場合は，ディクショナリサーバのホストが仮定されます。

-S オプション以外を指定した場合，操作するホスト名を指定します。省略した場合は，HiRDB 環境下のすべてのホストが仮定されます。

(c)　-u 認可識別子〔，認可識別子〕...

JAR ファイルの操作をする，ユーザの認可識別子を指定します。なお，英大文字と英小文字は同一文字として扱われます。

-I，-R，及び-D オプションを指定した場合は，このオプションを必ず指定してください。

認可識別子には，'MASTER'及び'HiRDB'は指定できません。

(d)　-f JAR ファイル名

操作をする JAR ファイルの名称を指定します。なお，英大文字と英小文字は同一文字として扱われます。

-I，-R，及び-D オプションを指定した場合は，このオプションを必ず指定してください。

-S，-L オプション指定時にこのオプションを省略した場合は，登録されているすべての JAR ファイルが仮定されます。

JAR ファイルは，システム定義の pd_java_archive_directory オペランドで指定したディレクトリに格納されます。

(5)　規則

1. pdjarsync コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdjarsync コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

3. pdjarsync コマンドは，複数同時に実行しないでください。

(6) 注意事項

1. SQL の INSTALL JAR，REPLACE JAR，又は REMOVE JAR を実行した場合，内部的に pdjarsync コマンドが実行されます。この場合，pdjarsync コマンドは，シングルサーバ又はフロントエンドサーバから実行されます。
2. pdjarsync コマンドのリターンコードを次に示します。

   0：正常終了

   4：異常終了（登録，再登録の場合は JAR ファイルが登録済み，削除の場合は JAR ファイルが登録されていないことを示します）

   8：異常終了

(7) 出力形式

pdjarsync コマンドのオプションに-L を指定，又はオプションを省略した場合に，次の情報が出力されます。

```
HOSTNAME(aabbcc)  DATE        TIME       SIZE    OWNER   FILE
dd...d            eeee/ff/gg  hh:ii:jj   kk...k  ll...l  mm...m
   :                :           :          :       :       :
```

[説明]

aabbcc：

pdjarsync コマンドを実行した時刻

dd...d：

ホスト名

eeee/ff/gg：

JAR ファイルの日付※

hh:ii:jj：

JAR ファイルの時刻※

kk...k：

JAR ファイルの大きさ（単位：バイト）

ll...l：

JAR ファイルの所有者名

mm...m：

JAR ファイルの名称

注※

SQL 文（INSTALL JAR，REPLACE JAR），又は JDBC のメソッドで JAR ファイルの登録又は再登録をした場合，SQL 文又は JDBC のメソッドを実行した日付及び時刻となります。

pdjarsync コマンドで JAR ファイルの登録又は再登録をした場合，登録又は再登録した JAR ファイルの日付及び時刻となります。

2.43　pdkill（プロセスの強制停止）

2.43.1　pdkill の形式と規則

(1)　機能

指定したプロセスを強制的に終了します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdkill プロセスID
```

(4)　コマンド引数

(a)　プロセス ID　～＜符号なし整数＞((1～10))

強制終了するプロセスのプロセス ID を指定します。プロセス ID は，pdls -d prc コマンドで確認できます。

(5)　規則

1. pdkill コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdkill コマンドは，各サーバマシンで実行してください。

(6)　注意事項

1. pdkill コマンドの結果は，pdls コマンド（-d prc 指定）で確認できます。
2. システムサーバ，またはクリティカル状態のプロセスを強制終了した場合，HiRDB が異常終了します。なお，プロセスがクリティカル状態かどうかは pdls -d rpc コマンドで確認できます。
3. pdkill コマンド実行時のリターンコードは，0 の場合は正常終了，-1 の場合は異常終了となります。

2.44　pdlistls（リスト定義情報の表示）

2.44.1　pdlistls の形式と規則

(1)　機能

リスト定義情報を表示します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdlistls 〔-d {1|2}〕
        〔-u リスト所有者名〕
        〔-t リスト基表所有者名.リスト基表名〕
        〔-W 実行監視時間〕
```

(4)　オプション

(a)　-d {1 | 2}

表示するリスト定義情報を指定します。

-d オプション	表示する情報					
	システム定義の値※	リストの所有者名	リスト名	リスト数	基表の所有者名	基表名
1	○	○	×	○	×	×
2	○	○	○	×	○	○

（凡例）

○：表示されます。

×：表示されません。

注※

pd_max_list_users 及び pd_max_list_count オペランドの指定値を示します。

(b)　-u リスト所有者名　～＜識別子＞((1～30))

特定のリスト所有者のリスト定義情報だけを表示する場合，そのリスト所有者名を指定します。省略した場合，すべてのリスト所有者が仮定されます。

リスト所有者名は，「¥""」と「"¥"」で囲まないかぎり大文字として扱われます。

(c)　-t リスト基表所有者名.リスト基表名

特定のリスト基表のリスト定義情報だけを表示する場合，そのリスト基表所有者名，リスト基表名を指定します。

リスト基表所有者名は 1～30 文字，リスト基表名は 1～30 文字の識別子で指定してください。また，リスト基表所有者名，リスト基表名は，「¥""」と「"¥"」で囲まないかぎり大文字として扱われます。また，空白を含む場合も「¥""」と「"¥"」で囲んでください。

(d)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdlistls コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pdlistls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdlistls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6)　注意事項

1. pdlistls コマンド実行時のリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(7)　出力形式

(a)　-d 1 の場合

```
 STATE OD LIST DEFINITION
 #       OWNER NAME    LIST COUNT
[aa...a  bb...b        cc...c]
           :
[NOT FOUND dd...d]
pd_max_list_users : ee...e
pd_max_list_count : ff...f
```

[説明]

aa...a：

表示上の通番

bb...b：

リストの所有者名

cc...c：

リストが使用しているセグメント数

dd...d：

見付からなかった情報

-t オプションを指定した場合：TABLE

-u オプションを指定した場合：USER

ee...e：

システム定義の pd_max_list_users オペランドの値（シングルサーバ又はディクショナリサーバでの値）

ff...f：

システム定義の pd_max_list_count オペランドの値（シングルサーバ又はディクショナリサーバでの値）

(b)　-d 2 の場合

```
  STATE OD LIST DEFINITION
  #       OWNER NAME  LIST NAME  BASE TABLE OWNER  BASE TABLE NAME
 [aa...a  bb...b      cc...c     dd...d            ee...e]
            :
 [NOT FOUND ff...f]
 pd_max_list_users : gg...g
 pd_max_list_count : hh...h
```

[説明]

aa...a：

表示上の通番

bb...b：

リストの所有者名

cc...c：

リスト名

dd...d：

リストの基表の所有者名

ee...e：

リストの基表名

ff...f：

見付からなかった情報

-t オプションを指定した場合：TABLE

-u オプションを指定した場合：USER

gg...g：

システム定義の pd_max_list_users オペランドの値（シングルサーバ又はディクショナリサーバでの値）

hh...h：

システム定義の pd_max_list_count オペランドの値（シングルサーバ又はディクショナリサーバでの値）

2.45　pdlogadpf（ログ関係のファイルの割り当て）

2.45.1　pdlogadpf の形式と規則

(1)　機能

HiRDB 稼働中に，HiRDB システム定義で指定したファイルグループに物理ファイルを割り当てます。物理ファイルは，pdlogadpf コマンド実行前に pdloginit コマンドで作成しておく必要があります。

pdlogadpf コマンド実行後，pdlogopen コマンドでオープンすると，ファイルグループは稼働中の HiRDB で使用できる状態になります。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogadpf -d spd 〔-s サーバ名〕 -g ファイルグループ名
          -a A系の物理ファイル名 〔-b B系の物理ファイル名〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogadpf -d spd -s サーバ名 -g ファイルグループ名
          -a A系の物理ファイル名 〔-b B系の物理ファイル名〕
```

(4)　オプション

(a)　-d spd

シンクポイントダンプファイルに物理ファイルを割り当てます。

(b)　-s サーバ名　～＜識別子＞((1～8))

ファイルグループに対応するサーバの名称を指定します。

(c)　-g ファイルグループ名　～＜識別子＞((1～8))

物理ファイルを割り当てるファイルグループの名称を指定します。シングルサーバ定義，フロントエンドサーバ定義，バックエンドサーバ定義，又はディクショナリサーバ定義で指定したファイルグループ名を指定してください。

(d)　-a A 系の物理ファイル名　～＜パス名＞((167 文字以内))

ファイルグループに割り当てる A 系の物理ファイル名を，絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。

(e)　-b B 系の物理ファイル名　～＜パス名＞((167 文字以内))

ファイルグループに割り当てる B 系の物理ファイル名を，絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。

このオプションは，シンクポイントダンプファイルを二重化している場合に必ず指定してください。二重化していない場合に指定したときはエラーとなります。

シンクポイントダンプファイルを二重化している場合，-a，及び-b オプションを指定し，片方の系でファイルの割り当てが失敗したときは，コマンドエラーとなります。

(5)　規則

1. pdlogadpf コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdlogadpf コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6)　注意事項

1. 既に物理ファイルを割り当てているファイルグループに対して，pdlogadpf コマンドを実行するとエラーになります。
2. pdlogadpf コマンドの結果は，コマンド実行時のリターンコード，エラーメッセージの有無，又はメッセージログから確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，4 の場合は異常終了，8 の場合は異常終了（オプション指定不正，rsh 失敗など）です。

(7)　使用例

システム定義で指定したファイルグループに物理ファイルを割り当てます。

- ファイルグループ名　spdfgf15
- 物理ファイル名　c:¥sysarea¥rdsys011¥spdf15

【HiRDB/シングルサーバの場合】

```
pdlogadpf -d spd   ..........................1
          -g spdfgf15   ....................2
          -a c:¥sysarea¥rdsys011¥spdf15   ...3
```

[説明]

1. ログ関係のファイルの種別　**spd（シンクポイントダンプファイル）**
2. ファイルグループ名　**spdfgf15**
3. 物理ファイル名　**c:¥sysarea¥rdsys011¥spdf15**

【HiRDB/パラレルサーバの場合】

```
pdlogadpf -d spd   ..........................1
          -s bes1   .........................2
          -g spdfgf15   ....................3
          -a c:¥sysarea¥rdsys011¥spdf15   ...4
```

[説明]

1. ログ関係のファイルの種別　**spd（シンクポイントダンプファイル）**
2. ファイルグループに対応するサーバの名称　**bes1**
3. ファイルグループ名　**spdfgf15**
4. 物理ファイル名　**c:¥sysarea¥rdsys011¥spdf15**

2.46　pdlogatul（自動ログアンロード機能の制御）

2.46.1　pdlogatul の形式と規則

(1)　機能

自動ログアンロード機能を制御します。次の制御ができます。

- HiRDB 稼働中に停止した自動ログアンロード機能を再開始します。
- 実行中の自動ログアンロード機能を中断します。
- 自動ログアンロード機能の動作状態を表示します。

自動ログアンロード機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogatul -d sys 〔-s サーバ名〕 〔 {-i | -b | -t 〔-w〕 } 〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogatul -d sys -s サーバ名 〔 {-i | -b | -t 〔-w〕 } 〕
```

(4)　オプション

(a)　-d sys

システムログファイルに対して操作することを示します。

(b)　-s サーバ名　～＜識別子＞((1～8))

操作するサーバの名称を指定します。

(c)　-i

自動ログアンロード機能の動作状態を表示する場合に指定します。

(d) -b

自動ログアンロード機能を再開始する場合に指定します。

再開始直後のアンロードログファイル作成ディレクトリは，前回動作時の情報が引き継がれます。

(e) -t

自動ログアンロード機能を中断する場合に指定します。

アンロード処理中の場合は，処理が中断されます。

(f) -w

-t オプションを指定して自動ログアンロード機能を中断する場合に，現在自動アンロード実行中のログ世代に対するアンロード処理が完了するまで，コマンドの終了を遅延させるときに指定します。このオプションは，-t オプションを指定したときだけ有効となります。なお，-t オプションを指定し，-w オプションを省略した場合は，pdlogatul コマンドを実行した時点で自動ログアンロード機能を中断し，実行中のアンロード処理を強制的に終了します。

このオプションを指定すると，アンロード処理完了の待ち時間は約 9 時間（32768 秒）となります。この時間を経過してもアンロード処理が完了しない場合は，その時点で自動ログアンロード機能を中断し，実行中のアンロード処理を強制終了します。

このオプションを指定すると，不完全な状態のアンロードログファイル（通常自動ログアンロード機能を中断した場合に残ることがある）がなくなるため，pdrstr でその時点のアンロードログファイルをすべて入力できます。

(5) 規則

1. pdlogatul コマンドは，HiRDB が稼働中の場合に実行できます。なお，-i 又は-t オプションを指定した場合は，HiRDB の停止処理中（pdstop コマンド実行中）でも実行できます。ただし，pdstart -r コマンドで HiRDB が稼働している場合（HiRDB の開始処理中及び終了処理中を含む）は，-i 又は-t オプションの指定に関係なく，pdlogatul コマンドを実行できません。

   -b，-t 又は-t -w オプションを指定できる，HiRDB の稼働状態と自動ログアンロード機能の動作状態の組み合わせを次に示します。

項番	HiRDB の稼働状態	自動ログアンロード機能の動作状態	オプション		
			-b	-t	-t -w
1	稼働中	ACTIVE	×	○	○
2		STOP	○	×	×
3		STOP（E）	○	○	○
4		STOPPING	○	○	×
5		−	×	×	×

項番	HiRDB の稼働状態	自動ログアンロード機能の動作状態	オプション		
			-b	-t	-t -w
6	停止処理中	ACTIVE	×	○	○
7		STOP	×	×	×
8		STOP（E）	×	○	○
9		STOPPING	×	○	×
10		–	×	×	×
11	その他	すべての状態	×	×	×

（凡例）

○：指定できます。

×：指定できません。

–：自動ログアンロード機能を使用できない状態です。

2. pdlogatul コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 現用世代のログファイルグループ情報は，現用ログ世代がある場合に出力されます。
4. pdlogatul -t -w コマンド実行時にリターンコードが 4 になった場合は，不完全な状態のアンロードログファイルが残っている可能性があります。この場合，出力されたエラーメッセージからエラー要因を調査し対策した後，再度 pdlogatul -b コマンドを実行してください。その後，pdlogatul -t -w コマンドを実行してください。

(6) 注意事項

1. pdlogatul コマンドのリターンコードを次に示します。

0：正常終了

4：次のどちらかの状態です。

- -t 及び-w オプションを指定して実行した場合，アンロード処理完了の待ち時間（32768 秒）が経過してもアンロード処理が完了しなかったため，アンロード処理を強制終了し，自動ログアンロード機能を中断しました。
- -t 及び-w オプションを指定して実行した場合，アンロード処理完了監視中に次のどれかの要因で自動ログアンロード機能が中断しました。
  - ・すべてのアンロードログ作成先ディレクトリが使用できません。
  - ・アンロードログファイルの名称が重複しています。
  - ・資源不足です（メモリ，最大起動プロセス数超過など）。

8：コマンドエラー

(7) 出力形式

```
HOSTNAME : aa...a(bb...b)
SERVER_NAME : cc...c
AUTO_LOG_UNLOAD  NOW_UNLOAD_LOG_GROUP  CREATE_DIR
              d                ee...e  ff...f
CURRENT LOG GENERATION INFO.
LOG_GROUP  GEN_NO.  SERVER_RUN_ID  RUN_ID  UNLOAD_FILE_NAME
gg...g     hh...h   ii...i         jj...j  kk...k
```

［説明］

aa...a：

ホスト名

bb...b：

pdlogatul を実行した時刻（時：分：秒）

cc...c：

サーバ名

d：

自動ログアンロード機能の動作状態

ACTIVE：動作中

STOP：停止中

STOP（E）：障害停止中

STOPPING：停止時のアンロード処理待ち監視中

－：自動ログアンロード機能を使用できない状態

ee...e：

現在アンロード中のシステムログファイルのファイルグループ名

自動ログアンロード機能が動作中でない場合，****が表示されます。

ff...f：

使用中のアンロードログファイル作成ディレクトリ名

gg...g：

現用世代のシステムログファイルのファイルグループ名

hh...h：

現用世代のシステムログ世代番号

ii...i：

現用世代のサーバのラン ID

jj...j：

現用世代のシステムログのラン ID

kk...k：

現用世代を自動ログアンロードしたときの生成ファイル名称

2.47　pdlogchg（ログ関係のファイルのステータス変更）

2.47.1　pdlogchg の形式と規則

(1)　機能

指定したファイルグループのステータスを，強制的にアンロード済み状態，又は HiRDB Datareplicator の抽出完了状態にします。現用ファイルグループのステータス変更はできません。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogchg {-d sys 〔-s サーバ名〕 -g ファイルグループ名 〔-R〕
            |-z ログポイント情報ファイル名}
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogchg {-d sys -s サーバ名 〔-u ユニット識別子〕 -g ファイルグループ名 〔-R〕
            |-z ログポイント情報ファイル名 〔-x ホスト名〕}
```

(4)　オプション

(a)　-d sys

システムログファイルのステータスを変更します。

(b)　-s サーバ名　～＜識別子＞((1～8))

ファイルグループに対応するサーバの名称を指定します。

(c)　-u ユニット識別子　～＜識別子＞((4 文字))

影響分散スタンバイレス型系切り替え機能を適用している場合，ステータスを変更するシステムログファイルを作成したディスクがあるホストの，ユニット識別子を指定します。

該当するサーバ，及びシステムマネジャが稼働中の場合，-u オプションの指定は無視されます。

次のどれかに該当する場合に-u オプションを指定すると，エラーとなります。

- 指定するユニットが，-s オプションで指定したサーバの移動先となる HA グループに属していない。
- -s オプションで指定したサーバが，HA グループ内のサーバではない。
- 影響分散スタンバイレス型系切り替え機能を適用していない。

-u オプションの指定要否を次に示します。

サーバの状態	ファイル作成先のディスクがあるホスト	-u オプションの指定
稼働中	実行系バックエンドサーバがあるホスト	省略できます。
停止中	現用系バックエンドサーバがあるホスト	
	上記以外のホスト	必ず指定します。

(d)　-g ファイルグループ名　～＜識別子＞((1～8))

ステータスを変更するファイルグループの名称を指定します。

(e)　-z　ログポイント情報ファイル名　～＜パス名＞

ログポイント情報ファイルに記述されているログポイントより古いすべてのファイルグループのステータスを変更します。

このログポイント情報ファイルには，次に示すファイルのどちらかを指定してください。

- pdcopy コマンドで-z オプションを指定して作成したファイル
- pdrstr コマンドで-z オプションを指定して作成したファイル

(f)　-x　ホスト名　～＜識別子＞((1～32))

ステータスを変更するログファイルグループが属するサーバが稼働するホスト名を指定します。

省略した場合，コマンドを入力したホストが仮定されます。

(g)　-R

HiRDB Datareplicator 連携を実行している場合に，抽出未完了状態のファイルグループを，強制的に抽出完了状態に変更するときに指定します。このオプションを指定すると，抽出状態だけを抽出未完了状態から抽出完了状態に変更します。

(5)　規則

1. pdlogchg コマンドの実行条件を次に示します。
   - システム定義で pd_log_unload_check=N を指定している場合，HiRDB が稼働中で，かつ操作対象のファイルグループがオープン状態のときは，-R オプションの指定がないと実行できません。

- 上記以外の場合，HiRDB の稼働に関係なく実行できます（ただし，HiRDB 開始処理中及び HiRDB 終了処理中は除きます）。

2. pdlogchg コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6) 注意事項

1. pdlogchg コマンドのリターンコードを次に示します。

   0：正常終了

   4：異常終了

   8：異常終了（オプション指定不正，rsh 失敗など）

   12：異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）

2. pdlogchg コマンドは，HiRDB 開始処理中に実行しないでください。HiRDB 開始処理中とは，KFPS01800-I メッセージの出力から KFPS05210-I メッセージの出力までの間のことをいいます。実行した場合，システムログファイルの状態が HiRDB に反映されないことがあります。このような場合には，HiRDB 開始処理完了後に pdlogcls コマンドで該当するファイルグループをクローズし，その後 pdlogopen コマンドでオープンすると，システムログファイルの状態が反映されます。

3. pdlogchg コマンドは，HiRDB システム定義ファイルの内容を参照して，処理をします。そのため，システムログを取得した稼働中の HiRDB の HiRDB システム定義ファイルと，pdlogchg コマンド実行時の HiRDB システム定義ファイルの内容が一致していない場合は，ステータスを変更できないことがあります。

4. pdlogchg コマンドがエラーとなった場合でも，正しく終了することがあります。正しく終了しているかどうかについては，終了コードが 0 であるか，又は pdlogls コマンドでステータスが変更になっているかを確認することで分かります。

5. pdlogchg コマンドの結果は，次の方法で確認できます。

   - HiRDB 稼働中の場合

     pdlogls コマンド

   - HiRDB 停止中の場合

     コマンド実行時のリターンコード，エラーメッセージの有無，又は pdlogls コマンド。

6. HiRDB が稼働していないときに pdlogchg コマンドを実行し，コマンドが終了する前に HiRDB を開始するとコマンド実行エラーとなります。その場合，HiRDB の稼働状態を確認してから，再度 pdlogchg コマンドを実行してください。

7. pdstop -s コマンドで個別に終了したサーバでは，そのサーバで pd_log_unload_check=N を指定している場合，-R オプションの指定がないと pdlogchg コマンドは実行できません。

8. -R オプションは，HiRDB Datareplicator 連携で，システム共通定義の pd_log_rpl_no_standby_file_opr=stop を指定している場合に次に示す三つの条件が満たされるような障害が発生して，データ連動をいったん停止するときに，強制的にファイルグループのステータスを変更するために実行します。

   - すべてのシステムログファイルが抽出未完了状態になったため，HiRDB が強制終了

- HiRDB の運用上の理由で定義の変更不可
- HiRDB Datareplicator で障害が発生して，HiRDB Datareplicator が稼働不可

このような場合には，-R オプションを指定した pdlogchg コマンドをすべてのシステムログファイルに対して実行し，抽出未完了状態を抽出完了状態に変更してください。このとき，抽出側データベースの更新情報の一部を消失してしまうため，反映側 DB に反映できなくなり，抽出側と反映側のデータベースが不整合となる可能性があります。そのため，データ連動の再開始をする前に反映側 DB を再作成する必要があります。データベースの再作成の手順については，マニュアル「HiRDB データ連動機能 HiRDB Datareplicator」を参照してください。

2.48　pdlogcls（ログ関係のファイルのクローズ）

2.48.1　pdlogcls の形式と規則

(1)　機能

指定したファイルグループを構成する，オープン中の物理ファイルをクローズし，稼働中の HiRDB で使用しないようにします。ただし，次のファイルグループを構成する物理ファイルはクローズできません。

- 現用，及び上書きできない状態のシステムログファイル
- 書き込み中，及び上書きできない状態のシンクポイントダンプファイル

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogcls {-d {sys | spd} 〔-s サーバ名〕 -g ファイルグループ名 〔-a〕 〔-b〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogcls {-d {sys | spd} -s サーバ名 -g ファイルグループ名 〔-a〕 〔-b〕
```

(4)　オプション

(a)　-d sys ｜ spd

クローズするログ関係のファイルを指定します。

sys：
　システムログファイル

spd：
　シンクポイントダンプファイル

(b)　-s サーバ名　～＜識別子＞((1～8))

ファイルグループに対応するサーバの名称を指定します。

(c) -g ファイルグループ名 ～<識別子>((1～8))

クローズするファイルのファイルグループ名を指定します。

(d) -a

システムログファイル又はシンクポイントダンプファイルをクローズする場合，A 系の物理ファイルをクローズ対象とするときに指定します。

(e) -b

システムログファイル又はシンクポイントダンプファイルをクローズする場合，B 系の物理ファイルをクローズ対象とするときに指定します。二重化していない場合にこのオプションを指定すると，コマンドエラーとなります。

-a オプション及び-b オプションを省略した場合，次のようになります。

- 二重化していない場合は，-a オプションを指定したときと同様に，A 系の物理ファイルをクローズ対象とします。
- 二重化している場合は，-a オプション及び-b オプションを指定したときと同様に，A 系及び B 系の物理ファイルをクローズ対象とします。

(5) 規則

1. pdlogcls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdlogcls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. クローズ対象となる物理ファイルが，すべてクローズ済みのときは，警告メッセージ（KFPS01283-W 又は KFPS02168-W）が出力されます。

(6) 注意事項

1. pdlogcls コマンドのリターンコードを次に示します。
   0：正常終了
   4：異常終了
   8：異常終了（オプション指定不正，rsh 失敗など）
2. ログ関係のファイルのスワップ処理中，又はほかの運用コマンドの処理中に pdlogcls コマンドを実行すると，エラーになることがあります。
3. pdlogcls コマンドの結果は，pdlogls コマンドで確認できます。
4. システム定義の pd_log_remain_space_check=safe 指定時，pdlogcls コマンドでシステムログファイルをクローズした場合に容量不足となるときは，KFPS01280-E メッセージ（理由コード：712）を出力し，エラーとなります。この場合，システムログファイルの空き容量を増やした後，pdlogcls コマンドを再度実行してください。システムログファイルの空き容量監視機能の運用方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

2.49　pdloginit（ログ関係のファイルの初期設定）

2.49.1　pdloginit の形式と規則

(1)　機能

HiRDB ファイルシステムにログ関係の物理ファイルを作成し，稼働中の HiRDB で使用できるように初期設定をします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdloginit {-d sys 〔-s サーバ名〕
             -f 物理ファイル名 -n HiRDBレコード数 〔-l レコード長〕
          |-d spd 〔-s サーバ名〕
             -f 物理ファイル名 -n HiRDBレコード数}
```

(b)　HiRDB/パラレルサーバの場合

```
pdloginit {-d sys -s サーバ名 〔-u ユニット識別子〕
             -f 物理ファイル名 -n HiRDBレコード数 〔-l レコード長〕 〔-D〕
          |-d spd  -s サーバ名 〔-u ユニット識別子〕
             -f 物理ファイル名 -n HiRDBレコード数 〔-D〕 }
```

(4)　オプション

(a)　-d sys ｜ spd

初期設定をするログ関係のファイルを指定します。

sys：

システムログファイル

spd：

シンクポイントダンプファイル

(b)　-s サーバ名　～＜識別子＞((1～8))

ファイルグループに対応するサーバの名称を指定します。

(c)　-f 物理ファイル名　～＜パス名＞((167 文字以内))

初期設定をする物理ファイルの名称を絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。HiRDB ファイル名は，HiRDB ファイルシステム領域内で重複しないようにしてください。既に初期設定されている HiRDB ファイル名を指定すると，エラーとなります。また，HiRDB ファイル名に pl で始まる名称は指定できません。

(d)　-n HiRDB レコード数　～＜符号なし整数＞((12～104857600))

初期設定をする HiRDB ファイルのレコード数を指定します。なお，ここでいうレコードとは，HiRDB ファイルシステム上の次に示すサイズの領域のことです。

- -d sys の場合は，-l オプションで指定したレコード長
- -d spd の場合は，4,096 バイト

＜指定範囲＞

HiRDB レコード数には，次に示す範囲の値を指定してください。

-d sys の場合

- 上限

  107374182400÷-l オプションの指定値
- 下限

  ↑(32000 ＋ 368 ＋ 3×4096)÷-l オプションの指定値↑＋ 1

-d spd の場合

12～262144

＜指定値の目安＞

HiRDB レコード数の見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

- -d sys の場合

  見積もったレコード数は総容量なので，作成するシステムログファイルの個数で割った値×1.2 を目安にしてください。
- -d spd の場合

  見積もったレコード数×1.2 を目安にしてください。

(e)　-l レコード長　～＜符号なし整数＞((1024～4096))

システムログファイルとして初期設定する HiRDB ファイルのレコード長を指定します。

通常はこのオプションを指定しないでください。

このオプションは，システム定義変更前にシステムログファイルを作っておきたい場合にだけ指定します。

＜規則＞

1. レコード長は，HiRDB システム定義の pd_log_rec_leng オペランドと同じ値にしてください。設定される値は次のようになります。
   - 指定値が 1,024 のときは 1,024 に設定されます。
   - 指定値が 1,025～2,048 のときは 2,048 に設定されます。
   - 指定値が 2,049～4,096 のときは 4,096 に設定されます。

   上記以外の値を指定するとエラーになります。
2. このオプションを省略すると，HiRDB システム定義の pd_log_rec_leng オペランドに指定した値で初期設定します。pd_log_rec_leng オペランドと異なるレコード長で初期設定したファイルは，システムログファイルとしてオープンできません。
3. シンクポイントダンプファイルとして初期設定する場合，このオプションを指定するとエラーになります。また，HiRDB ファイルシステムのセクタ長未満のレコード長を指定した場合もエラーになります。

(f)　-u ユニット識別子　～＜識別子＞((4 文字))

影響分散スタンバイレス型系切り替え機能を適用している場合，ファイル作成先のディスクがあるホストの，ユニット識別子を指定します。

該当するサーバが稼働中の場合，-u オプションの指定は無視されます。

次のどれかに該当する場合に-u オプションを指定すると，エラーとなります。

- 指定するユニットが，-s オプションで指定したサーバの移動先となる HA グループに属していない。
- -s オプションで指定したサーバが，HA グループ内のサーバではない。
- 影響分散スタンバイレス型系切り替え機能を適用していない。

-u オプションの指定要否を次に示します。

サーバの状態	ファイル作成先のディスクがあるホスト	-u オプションの指定
稼働中	実行系バックエンドサーバがあるホスト	省略できます。
停止中	現用系バックエンドサーバがあるホスト	
	上記以外のホスト	必ず指定します。

(g) -D

pdchgconf コマンドでユニットの追加，又はサーバの追加をする場合に指定します。

ユニットの追加，又はサーバの追加をする場合に必要となる，システムログファイル又はシンクポイントダンプファイルの初期化をします。このオプションを指定した場合，サーバ名のチェックはしません。

このオプションを指定して pdloginit コマンドを実行する場合，作成するシステムログファイル又はシンクポイントダンプファイルがあるユニットで実行してください。

(5) 規則

1. pdloginit コマンドは，HiRDB の稼働に関係なく実行できます（ただし，HiRDB 開始処理中及び HiRDB 終了処理中は除きます）。
2. pdloginit コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdloginit コマンド実行時，環境変数の PDDIR に HiRDB ディレクトリの環境変数と同じものが設定されていないと，コマンドエラーになったときにメッセージが出力されません。
4. システムログファイルのレコード長は，各サーバのすべてのファイルで，HiRDB システム定義の pd_log_rec_leng オペランドと同じ値を指定してください。

(6) 注意事項

1. システムログファイルを二重化する場合，一つのログファイルグループを構成する A 系と B 系の物理ファイルは，レコード長とレコード数が同じになるように初期設定してください。
2. システムログファイルを二重化する場合，A 系と B 系の物理ファイルは障害発生に備えて，なるべく別々のハードディスクに作成してください。
3. pdloginit コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：異常終了

   8：異常終了（オプション指定不正，rsh 失敗など）

   12：異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）
4. セクタ長が 4,096 バイトを超える HiRDB ファイルシステムには，ファイルを初期設定できません。

(7) 使用例

例 1：

システムログファイルを作成します。

【HiRDB/シングルサーバの場合】

```
pdloginit -d sys   ..........................1
          -f c:¥sysarea¥rdsys011¥logf11   ...2
```

```
          -n 64   ..........................3
pdloginit -d sys  ..........................1
          -f c:¥sysarea¥rdsys012¥logf12   ...2
          -n 64   ..........................3
```

[説明]

1. 初期設定をするログ関係のファイル種別　**sys（システムログファイル）**
2. 物理ファイル名　**c:¥sysarea¥rdsys011¥logf11，c:¥sysarea¥rdsys012¥logf12**
3. 物理ファイルのレコード数　**64**

【HiRDB/パラレルサーバの場合】

```
pdloginit -d sys  ..........................1
          -s bes1   ........................2
          -f c:¥sysarea¥rdsys011¥logf11   ...3
          -n 64   ..........................4
pdloginit -d sys  ..........................1
          -s bes1   ........................2
          -f c:¥sysarea¥rdsys012¥logf12   ...3
          -n 64   ..........................4
```

[説明]

1. 初期設定をするログ関係のファイル種別　**sys（システムログファイル）**
2. ファイルグループに対応するサーバ名　**bes1**
3. 物理ファイル名　**c:¥sysarea¥rdsys011¥logf11，c:¥sysarea¥rdsys012¥logf12**
4. 物理ファイルのレコード数　**64**

例 2：

シンクポイントダンプファイルを作成します。

【HiRDB/シングルサーバの場合】

```
pdloginit -d spd  ..........................1
          -f c:¥sysarea¥rdsys011¥spdf11   ...2
          -n 64   ..........................3
pdloginit -d spd  ..........................1
          -f c:¥sysarea¥rdsys012¥spdf12   ...2
          -n 64   ..........................3
```

[説明]

1. 初期設定をするログ関係のファイル種別　**spd（シンクポイントダンプファイル）**
2. 物理ファイル名　**c:¥sysarea¥rdsys011¥spdf11，c:¥sysarea¥rdsys012¥spdf12**
3. 物理ファイルのレコード数　**64**

【HiRDB/パラレルサーバの場合】

```
pdloginit -d spd  ..........................1
          -s bes1   ........................2
          -f c:¥sysarea¥rdsys011¥spdf11   ...3
          -n 64   ..........................4
```

```
pdloginit -d spd   ..........................1
          -s bes1   .........................2
          -f c:¥sysarea¥rdsys012¥spdf12   ...3
          -n 64   ...........................4
```

［説明］

1. 初期設定をするログ関係のファイル種別　**spd（シンクポイントダンプファイル）**
2. ファイルグループに対応するサーバ名　**bes1**
3. 物理ファイル名　**c:¥sysarea¥rdsys011¥spdf11，c:¥sysarea¥rdsys012¥spdf12**
4. 物理ファイルのレコード数　**64**

2.50　pdlogls（ログ関係のファイルの情報表示）

2.50.1　pdlogls の形式と規則

(1)　共通機能

ログ関係のファイルの情報を表示します。

(2)　共通形式

```
pdlogls -d表示対象種別 〔表示対象種別ごとのオプション〕
```

(3)　共通オプション

(a)　-d 表示対象種別

情報を表示するログ関係のファイルを指定します。

spd：

シンクポイントダンプファイル

sys：

システムログファイル

(4)　共通注意事項

1. pdlogls コマンドのリターンコードを次に示します。

   0：正常終了

   4：異常終了

   8：異常終了（オプション指定不正，rsh 失敗など）

   12：異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）

2. システム定義の pd_tmp_directory オペランドを指定しないで，環境変数 TMP だけを指定している場合，pdlogls コマンド，又はほかの HiRDB のコマンドを実行中に中断すると，環境変数 TMP に設定したディレクトリ下に"pdcmd"で始まる名称のファイルが残ることがあります。pdlogls コマンド，又はほかの HiRDB のコマンドが終了した後も，"pdcmd"で始まる名称のファイルが残っている場合は，エクスプローラなどで削除してください。

2.51　pdlogls【-d spd】（シンクポイントダンプファイルの情報表示）

2.51.1　pdlogls【-d spd】の形式と規則

(1)　機能

シンクポイントダンプファイルの情報を表示します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogls -d spd 〔-s サーバ名〕 〔-g ファイルグループ名〕 〔-e〕 〔-l〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogls -d spd -s サーバ名 〔-g ファイルグループ名〕 〔-e〕 〔-l〕
```

(4)　オプション

(a)　-s サーバ名　～＜識別子＞((1～8))

情報を表示するサーバの名称を指定します。

(b)　-g ファイルグループ名　～＜識別子＞((1～8))

特定のファイルグループの情報を表示する場合に，そのファイルグループ名を指定します。省略した場合，指定したサーバの，すべてのファイルグループの情報が表示されます。

(c)　-e

要素ファイルの情報を表示する場合に指定します。省略した場合，要素ファイルの情報は表示されません。

(d)　-l

シンクポイントダンプ有効化のスキップ回数の上限値を表示する場合に指定します。省略した場合，シンクポイントダンプ有効化のスキップ回数の上限値は表示されません。また，シンクポイントダンプ有効化のスキップ回数監視機能を使用していない場合は 0 が表示されます。

(5) 規則

1. pdlogls コマンドは，-d spd の場合は HiRDB が稼働中のときだけ実行できます。
   ただし，次の場合は実行できません。
   - HiRDB 開始処理中のとき
   - HiRDB 終了処理中のとき
   - pdstart -r コマンドで HiRDB が稼働中のとき
   - -s オプションで指定したサーバのユニットが停止中のとき
   - -s オプションで指定したサーバが停止中，又は停止処理中のとき
2. pdlogls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6) 注意事項

1. ファイルグループの状態の変更中（スワップ中，又はファイルグループを操作する運用コマンドの処理中など）に，pdlogls コマンドを実行すると，正しい状態が表示されないことがあります。
2. 1 レコードは，HiRDB ファイルシステム領域上の 4,096 バイトのことです。

(7) 出力形式

(a) pdlogls -d spd -s サーバ名 -e -l を指定した場合

```
HOSTNAME: aa...a(bbccdd)
pd_spd_syncpoint_skip_limit: ee...e
Server  Group   Generation Status Log     Log
name            number            group   block
ff...f  gg...g  hh...h     [i]    jj...j  kk...k
       Status Path
 Sys-A [l]    mm...m
 Sys-B [n]    oo...o
```

[説明]

aa...a：

ホスト名（32 文字以内）

bbccdd：

時分秒（pdlogls コマンド実行時刻）

ee...e：

シンクポイントダンプ有効化のスキップ回数の上限値（10 進数）。シンクポイントダンプ有効化のスキップ回数監視機能を使用していない場合は 0 が表示されます。

ff...f：

サーバ名（8 文字以内）

gg...g：

ファイルグループ名（8 文字以内）

hh...h：

世代番号（16 進数 8 けた）

i：

世代状態

a：上書きできない状態

u：書き込み中，又は上書きできる状態

r：予約の状態

jj...j：

HiRDB が再開始するときに読み込むシステムログのファイルグループ名（8 文字以内）

kk...k：

HiRDB が再開始するときに読み込むログのブロック番号（16 進数 8 けた）

l：

A 系の物理ファイルの状態

o：オープン状態

c：クローズ状態

h：閉塞状態

mm...m：

A 系の物理ファイルのパス名

n：

B 系の物理ファイルの状態

o：オープン状態

c：クローズ状態

h：閉塞状態

oo...o：

B 系の物理ファイルのパス名

2.52　pdlogls【-d sys】（システムログファイルの情報表示）

2.52.1　pdlogls【-d sys】の形式と規則

(1)　機能

システムログファイルの情報を表示します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogls -d sys 〔-s サーバ名〕 〔-g ファイルグループ名〕
            〔-e〕 〔-C 〔-H〕〕 〔-E〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogls -d sys -s サーバ名 〔-u ユニット識別子〕 〔-g ファイルグループ名〕
            〔-e〕 〔-C 〔-H〕〕 〔-E〕
```

(4)　オプション

(a)　-s サーバ名　～＜識別子＞((1～8))

情報を表示するサーバの名称を指定します。

(b)　-g ファイルグループ名　～＜識別子＞((1～8))

特定のファイルグループの情報を表示したい場合に，そのファイルグループの名称を指定します。省略した場合，指定したサーバのすべてのファイルグループの情報が表示されます。

(c)　-e

要素ファイルの情報を表示したいときに指定します。省略した場合，要素ファイルの情報は表示されません。

(d)　-u ユニット識別子　～＜識別子＞((4 文字))

影響分散スタンバイレス型系切り替え機能を適用している場合，情報を表示するシステムログファイルを作成したディスクがあるホストの，ユニット識別子を指定します。

該当するサーバ，及びシステムマネジャが稼働中の場合，-u オプションの指定は無視されます。

次のどれかに該当する場合に-u オプションを指定すると，エラーとなります。

- 指定するユニットが，-s オプションで指定したサーバの移動先となる HA グループに属していない。
- -s オプションで指定したサーバが，HA グループ内のサーバではない。
- 影響分散スタンバイレス型系切り替え機能を適用していない。

u オプションの指定要否を次に示します。

サーバの状態	ファイル作成先のディスクがあるホスト	-u オプションの指定
稼働中	実行系バックエンドサーバがあるホスト	省略できます。
停止中	現用系バックエンドサーバがあるホスト	
	上記以外のホスト	必ず指定します。

(e)　-C

システムログファイルの情報を，DAT 形式で出力する場合に指定します。

(f)　-H

-C オプションを指定した場合に，タイトル行を付けて出力するときに指定します。

(g)　-E

システムログファイルの詳細情報を表示する場合に指定します。

(5)　規則

1. pdlogls コマンドは，-d sys の場合は HiRDB の稼働に関係なく実行できます。ただし，次の場合は実行できません。
   - HiRDB 開始処理中のとき
   - HiRDB 終了処理中のとき
   - pdstart -r コマンドで HiRDB が稼働中のとき
   - -s オプションで指定したサーバのユニットが稼働中で，かつ指定したサーバが停止中，又は停止処理中のとき
2. pdlogls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6) 注意事項

1. ファイルグループの状態の変更中（スワップ中，又はファイルグループを操作する運用コマンドの処理中など）に，pdlogls コマンドを実行すると，正しい状態が表示されないことがあります。
2. アンロード（状態変更）しても，アンロード待ち状態と表示される場合は，以前に入力した pdlogunld コマンド，又は pdlogchg コマンドでメモリ中の状態を更新できない障害が発生している可能性があります。ただし，この場合アンロード処理自体は正常に終了しています。このような場合には，該当するファイルグループを pdlogcls コマンドでクローズし，その後 pdlogopen コマンドでオープンしてください。
3. システム定義の pdlogadpf -d sys オペランドの指定上限数を超えた分については，開始時には無視して HiRDB を開始しますが，停止時に実行する pdlogls コマンドの結果には含めて表示されます。

(7) 出力形式

(a) pdlogls -d sys -s サーバ名 -e -E を指定した場合

```
HOSTNAME : aa...a(bbccdd)
***** Off-line Information *****   ....................................1
Group  Type Server Gen No. Status  Run ID Block No.     Ex-Status
ee...e fff  gg...g hh...h  ijklmno pp...p qq...q rr...r JKLMNOPQ   .......2
      Element  Status   Record-count   Sys-A  Sys-B※ Record-size
      ss...s   tuvwxy   AA...A BB...B  CDEFGH CDEFGH      IIII   .........3
      AutoExp-A   AutoExp-B※
      R SS...S    R SS...S   ........................................4
```

注※

システムログファイルを二重化している場合に表示されます。

[説明]

1 はオフライン（対応サーバが停止状態）のログファイル情報の場合に表示します※1。2 はファイルグループの状態，3 は要素ファイルの状態，4 は要素ファイルの詳細情報を示します。2〜4 は，ファイルグループの数だけ繰り返し表示されます。なお，R と SS...S は，-E を指定したときだけ表示されます。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdlogls コマンドを実行した時刻（時分秒）

ee...e：

ファイルグループ名（8 文字以内）

fff：

ファイル種別（3 文字）

sys：システムログファイル

gg...g：

サーバ名（8 文字以内）

hh...h：

世代番号（16 進数 8 けた以内）

ijklmno：

ファイルグループの状態（7 文字）

i：

ファイルグループのオープン状態

o：オープン中

c：クローズ中

使用しているファイルグループがクローズ状態の場合は，pdlogopen コマンドでファイルグループをオープンしてください。

j：

ファイルグループの現在の状態

c：現用

s：待機中

n：予約

使用しているファイルグループが予約の場合は，pdlogopen コマンドで待機中にしてください。

k：

ファイルグループのアンロード状態

なお，アンロード待ち状態のファイルグループは，過去に現用として使用時，スワップしたため現用でなくなり，アンロードする必要があるログを含んでいます。

●オフライン状態（*** Off-line Information *****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）で，HiRDB Datareplicator 連携をしていない場合**

u：

アンロード待ち状態

-：

アンロード済み状態，又はアンロードできない※10

●オフライン状態（*** Off-line Information *****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）で，HiRDB Datareplicator 連携をしている場合**

a：次のどちらかの状態です※6。

・アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※9

・アンロード待ち，かつオフライン時に HiRDB Datareplicator によるデータ連動情報の抽出が完了した状態

u：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

e：次のどちらかの状態です※6。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※9

・アンロード済み，かつオフライン時に HiRDB Datareplicator によるデータ連動情報の抽出が完了した状態

-：次のどちらかの状態です。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

・アンロード又は HiRDB Datareplicator によるデータ連動情報の抽出ができない※10

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）※7で，HiRDB Datareplicator 連携をしていない場合**

u：

アンロード待ち状態※8

-：

アンロード済み状態，又はアンロードできない※10

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）※7で，HiRDB Datareplicator 連携をしている場合**

a：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※8，※9

u：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態※8

e：

アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※8，※9

-：次のどちらかの状態です。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

・アンロード又は HiRDB Datareplicator によるデータ連動情報の抽出ができない※10

l：

ファイルグループが上書きできるか，できないかの状態

d：

上書きできません（HiRDB システムの回復に必要なログを含んでいます）

サーバ内のすべてのシステムログファイルがこの状態の場合，システムログファイルを新規追加してください。

-：

上書きできます（HiRDB システムの回復に必要なログを含んでいません）

m：

ファイルグループの HiRDB での状態

b：

pdlogchg 若しくは pdlogunld コマンドで使用中，又は回復処理で使用中の状態です。

-：

pdlogchg 若しくは pdlogunld コマンドで使用していないか，又は回復処理で使用していない状態です。

n：

ファイルグループの使用履歴

c：

次のどちらかの状態のファイルグループです。HiRDB が停止中のときだけ表示されます。

- 現在，現用として使用している
- 過去に HiRDB に障害が発生したときに現用ログとして使用していて，システムログファイル内の管理情報がそのままとなっている

-：

過去に現用として使用していて，正しく処理されたファイルグループです。

o：

システムが使用する内部情報

pp...p：

ラン ID（ファイルが使用されたときのログサーバプロセスのラン ID）（16 進数 8 けた）

qq...q：

先頭ブロック番号（16 進数 8 けた以内）

rr...r：

最終ブロック番号（16 進数 8 けた以内）

ss...s：

要素ファイル名（8 文字以内）

tuvwxy：

要素ファイルの状態（6 文字）

t：

要素ファイルのオープン状態

o：

オープン状態（該当する要素ファイルを構成する物理ファイルがオープンされています）

c：

クローズ状態（該当する要素ファイルを構成する物理ファイルがクローズされています）

u：

要素ファイルが使用できるかどうかの状態

n：

要素ファイルが，HiRDB の稼働中に使用できる物理ファイルで構成されていません

u：

要素ファイルが，HiRDB の稼働中で使用できる物理ファイルで構成されています

v：

要素ファイルのアンロード状態※2

なお，アンロード待ち状態の要素ファイルは，過去に現用として使用時，スワップしたため現用でなくなり，アンロードする必要があるログを含んでいます。

●オフライン状態（*** Off-line Information *****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）で，HiRDB Datareplicator 連携をしていない場合**

u：

アンロード待ち状態

-：

アンロード済み状態，又はアンロードできない※10

●オフライン状態（*** Off-line Information *****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）で，HiRDB Datareplicator 連携をしている場合**

a：次のどちらかの状態です※6。

・アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※9

・アンロード待ち，かつオフライン時に HiRDB Datareplicator によるデータ連動情報の抽出が完了した状態

u：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

e：次のどちらかの状態です※6。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※9

・アンロード済み，かつオフライン時に HiRDB Datareplicator によるデータ連動情報の抽出が完了した状態

-：次のどちらかの状態です。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

・アンロード又は HiRDB Datareplicator によるデータ連動情報の抽出ができない※10

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）※7で，HiRDB Datareplicator 連携をしていない場合**

u：

アンロード待ち状態※8

-：

アンロード済み状態，又はアンロードできない※10

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）※7で，HiRDB Datareplicator 連携をしている場合**

a：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※8，※9

u：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態※8

e：

アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※8，※9

-：次のどちらかの状態です。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

・アンロード又は HiRDB Datareplicator によるデータ連動情報の抽出ができない※10

w：

システムが使用する内部情報

x：

要素ファイルの閉塞状態

h：

閉塞中（該当する要素ファイルを構成する，物理ファイルに対するログの出力，又はログファイルヘッダの入出力で障害が発生し，以後ログを出力していない物理ファイルを含む要素ファイルです）

-：

閉塞中ではありません（正常な要素ファイルです）。

y：

要素ファイルが読み込みできるか，できないかの状態

r：

該当する要素ファイルを構成する，物理ファイルが読み込みできない状態です。

-：

該当する要素ファイルを構成する，物理ファイルが読み込みできる状態です。

AA...A：

該当するファイルでの使用済みレコード数（16 進数 8 けた以内）※3※5。

ログを出力した HiRDB ファイルシステム領域上のレコード数です。

BB...B：

該当するファイル内の全レコード数（16 進数 8 けた以内）※4※5。

ログを出力した HiRDB ファイルシステム領域上のレコード数です。

CDEFGH：

物理ファイルの状態（6 文字）

C：

物理ファイルのオープン状態

o：

オープン中（該当する物理ファイルをオープンしています）

c：

クローズ中（該当する物理ファイルをクローズしています）

D：

物理ファイルの状態

c：

現用

ファイルグループの状態が現用の場合は，実際にログの出力対象になっていることを示します。ファイルグループの状態が現用でない場合は，過去に現用であったときに障害が発生し，回復していないためにログファイルヘッダにステータスが残っている状態です。pdlogunld コマンド又は pdlogchg コマンドで状態を回復するか，又は pdloginit コマンドで初期設定しないと再使用できません。

s：

待機

E：

物理ファイルのアンロード状態※2

なお，アンロード待ち状態の物理ファイルは，過去に現用として使用時，スワップしたため現用でなくなり，アンロードする必要があるログを含んでいます。

●オフライン状態（*** Off-line Information *****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）で，HiRDB Datareplicator 連携をしていない場合**

u：

アンロード待ち状態

-：

アンロード済み状態，又はアンロードできない※10

●オフライン状態（*** Off-line Information *****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）で，HiRDB Datareplicator 連携をしている場合**

a：次のどちらかの状態です※6。

・アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※9

・アンロード待ち，かつオフライン時に HiRDB Datareplicator によるデータ連動情報の抽出が完了した状態

u：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

e：次のどちらかの状態です※6。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※9

・アンロード済み，かつオフライン時に HiRDB Datareplicator によるデータ連動情報の抽出が完了した状態

-：次のどちらかの状態です。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

・アンロード又は HiRDB Datareplicator によるデータ連動情報の抽出ができない※10

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）※7で，HiRDB Datareplicator 連携をしていない場合**

u：

アンロード待ち状態※8

-：

アンロード済み状態，又はアンロードできない※10

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）※7で，HiRDB Datareplicator 連携をしている場合**

a：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※8，※9

u：

アンロード待ち，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態※8

e：

アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出待ち状態※8，※9

-：次のどちらかの状態です。

・アンロード済み，かつ HiRDB Datareplicator によるデータ連動情報の抽出済み状態

・アンロード又は HiRDB Datareplicator によるデータ連動情報の抽出ができない※10

F：

システムが使用する内部情報

G：

物理ファイルの閉塞状態

h：

閉塞中です（ログの出力，又はログファイルヘッダの入出力で障害が発生し，以後ログを出力していません）。

-：

閉塞中ではありません（ログを出力できます）。

H：

物理ファイルを読み込みできるか，できないかの状態

r：

読み込みできない状態（全面回復，部分回復しようとしても，ログが消去（初期化）されていて，読み込みできません。システムログファイルに障害が発生し，障害の要因を取り除いた後に物理ファイルを再作成してオープンした場合に，この状態になります）

-：

読み込みできる状態

IIII：

該当するファイルのレコード長（10 進数 4 けた）※5

ログを出力できる HiRDB ファイルシステム上のレコード長です。

JKLMNOPQ：

ファイルグループの拡張状態（8 文字）

J：

常に - を表示します。

K：

ファイルグループの自動拡張状態

●オフライン状態（*** Off-line Information*****が表示されている，又は DAT 形式出力時は UU...U に OFFLINE が表示されている）の場合**

-：

システムログファイルの自動拡張機能が動作していません。

●オンライン状態（*** Off-line Information *****が表示されていない，又は DAT 形式出力時は UU...U に OFFLINE が表示されていない）の場合**

-：

システムログファイルの自動拡張機能を使用していません。

u：

システムログファイルの自動拡張機能の拡張対象です。

e：

システムログファイルの拡張中です。

x：

システムログファイルの自動拡張機能の拡張対象外です。

なお，システムログファイルの自動拡張機能を使用している場合，現用ファイルは，拡張中のときは e，拡張中以外のときは x が表示されます。

L：

ファイルグループの自動拡張可否

-：

システムログファイルの自動拡張機能で拡張できます。

x：

システムログファイルの自動拡張機能で拡張できません。

M：

システムが使用する内部情報

N：

システムが使用する内部情報

O：

システムが使用する内部情報

P：

システムが使用する内部情報

Q：

システムが使用する内部情報

R：

要素ファイルの自動拡張可否

-：システムログファイルの自動拡張機能で拡張できます。

f：システムログファイルの容量の上限に達したため，システムログファイルの自動拡張機能で拡張できません。

c：pd_log_auto_expand_size オペランドで指定した拡張上限サイズに達したため，システムログファイルの自動拡張機能で拡張できません。

e：システムログファイルの自動拡張機能で拡張時にエラーが発生したため，拡張できません。

x：システムログファイルが pdfmkfs コマンドに-a オプションを指定しないで作成した HiRDB ファイルシステム領域に作成されているため，システムログファイルの自動拡張機能で拡張できません。

SS...S：

要素ファイルの使用不可領域のレコード数（16 進数 8 けた以内右詰め）※5

システムログファイルの自動拡張機能で発生した，使用できない領域をレコード数で表示します。

注※1

オフライン（対応サーバが停止状態）での状態表示（ファイルグループ，要素ファイル，物理ファイルの A/B 系）は，k，v，及び E で示すアンロード状態だけが有効な情報となります。

注※2

ファイルグループがアンロード済み状態の場合は，要素ファイル及び物理ファイルがアンロード待ち状態であっても，pdlogunld コマンド又は pdlogchg コマンドを実行する必要はありません。これは，pdlogunld コマンド又は pdlogchg コマンド実行時に障害が発生してステータスが書き換えられなかったためです。

注※3

0 が表示されている場合は，ログが 1 件も出力されていない場合です。全面回復完了後，現用以外の状態の場合は，使用済みレコード数の内容が正しくないことがあります。

注※4

ファイル内の全レコード数は，ログブロックを出力できる HiRDB ファイルのレコード数であり，pdloginit コマンドの-l オプションで指定したレコード長によって次のようになります。

- -l 1024：pdloginit コマンドの-n オプションで指定したレコード数−13
- -l 2048：pdloginit コマンドの-n オプションで指定したレコード数−7
- -l 4096：pdloginit コマンドの-n オプションで指定したレコード数−4

注※5

1 レコードは，ログサーバプロセスで管理する HiRDB ファイルシステム上の次に示すサイズの領域のことです。

- pdloginit コマンドの-d オプションで sys を指定した場合は，-l オプションで指定したレコード長
- pdloginit コマンドの-d オプションで spd を指定した場合は，4,096 バイト

注※6

該当するファイルグループで HiRDB Datareplicator によるデータ連動情報の抽出が完了したかどうかを確認するためには，pdls -d rpl コマンドによって出力される抽出側 HiRDB Datareplicator の抽出状況を参照してください。

注※7

HiRDB の起動時にオープンされないファイルグループ（pdlogadfg オペランドに ONL 指定がないファイルグループ）の状態は，「-」と表示されます。正しい状態を表示するには，pdlogopen コマンドで該当するファイルグループをオープンした後で，再度 pdlogls コマンドを実行してください。

注※8

オンライン中にオープンされているファイルグループをクローズした場合，そのファイルグループの状態表示はクローズ前の状態が出力されます。次の場合でも出力される状態は変更されません。

- アンロードが完了した
- HiRDB Datareplicator によるデータ連動情報の抽出が完了した
- pdlogchg でファイルグループの状態を変更した

正しい状態を表示するには，pdlogopen コマンドで該当するファイルグループをオープンした後で，再度 pdlogls コマンドを実行してください。

注※9

HiRDB Datareplicator と連携していて，HiRDB Datareplicator が未抽出のシステムログファイルを含んでいます。

注※10

次に示す理由で，アンロード，又は HiRDB Datareplicator によるデータ連動情報の抽出ができません。

- 使用できるシステムログファイルがない，又はシステムログファイルが存在しない
- 入出力エラーが発生した
- ほかのサーバで使用したシステムログファイルである

(b) pdlogls -d sys -s サーバ名 -e -C -H -E を指定した場合

```
"EDIT-TIME","HOSTNAME","INFORMATION","GROUP","TYPE","SERVER","GEN NO.","G-STATUS1","G-
STATUS2","G-STATUS3","G-STATUS4","G-STATUS5","G-STATUS6","G-STATUS7","RUN ID","BLOCK N
O.(FIRST)","BLOCK NO.(LAST)","EX-STATUS1","EX-STATUS2","EX-STATUS3","EX-STATUS4","EX-S
TATUS5","EX-STATUS6","EX-STATUS7","EX-STATUS8","ELEMENT","E-STATUS1","E-STATUS2","E-ST
ATUS3","E-STATUS4","E-STATUS5","E-STATUS6","RECORD-COUNT(USED)","RECORD-COUNT(ALL)","A
-STS1","A-STS2","A-STS3","A-STS4","A-STS5","A-STS6","B-STS1","B-STS2","B-STS3","B-STS4
","B-STS5","B-STS6","RECORD-SIZE","A-AUTOEXP-STS","A-UNUSED-SIZE","B-AUTOEXP-STS","B-U
```

```
NUSED-SIZE”
”TT...T”,”aa...a”,”UU...U”,”ee...e”,”fff”,”gg...g”,hh...h,”i”,”j”,”k”,”l”,”m”,”n”,”o”,
pp...p,qq...q,rr...r,”J”,”K”,”L”,”M”,”N”,”O”,”P”,”Q”,”ss...s”,”t”,”u”,”v”,”w”,”x”,”y”,
AA...A,BB...B,”C”,”D”,”E”,”F”,”G”,”H”[,”C”,”D”,”E”,”F”,”G”,”H”],IIII,”R”,SS...S[,”R”,S
S...S]
```

[説明]

[ ]内は，システムログファイルを二重化している場合に表示されます。

aa...a～SS...S については，pdlogls -d sys -s サーバ名 -e -E を指定した場合の説明を参照してください。

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

1 行には，1 ファイルグループの情報を表示します。

TT...T：

pdlogls コマンドを実行した時刻を YYYY/MM/DD hh:mm:ss の形式で表示します。

YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒

UU...U：

オフライン（対応サーバが停止状態）のログファイル情報の場合，OFFLINE を表示します。オンラインの場合は空要素を表示します。

2.53　pdlogopen（ログ関係のファイルのオープン）

2.53.1　pdlogopen の形式と規則

(1)　機能

指定したファイルグループを構成するクローズ中の物理ファイルをオープンし，稼働中の HiRDB で使用できるようにします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogopen {-d {sys|spd} 〔-s サーバ名〕 -g ファイルグループ名 〔-a〕 〔-b〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogopen {-d {sys|spd} -s サーバ名 -g ファイルグループ名 〔-a〕 〔-b〕
```

(4)　オプション

(a)　-d sys ｜ spd

オープンするログ関係のファイルを指定します。

sys：

システムログファイル

spd：

シンクポイントダンプファイル

(b)　-s サーバ名　～＜識別子＞((1～8))

ファイルグループに対応するサーバの名称を指定します。

(c)　-g ファイルグループ名　～＜識別子＞((1～8))

オープンするファイルのファイルグループ名を指定します。

(d) -a

システムログファイル又はシンクポイントダンプファイルをオープンする場合，A 系の物理ファイルをオープン対象とするときに指定します。

(e) -b

システムログファイル又はシンクポイントダンプファイルをオープンする場合，B 系の物理ファイルをオープン対象とするときに指定します。二重化していない場合にこのオプションを指定すると，コマンドエラーとなります。

-a オプション及び-b オプションを省略した場合，次のようになります。

- 二重化していない場合は，-a オプションを指定したときと同様に，A 系の物理ファイルをオープン対象とします。
- 二重化している場合は，-a オプション及び-b オプションを指定したときと同様に，A 系及び B 系の物理ファイルをオープン対象とします。

(5) 規則

1. pdlogopen コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdlogopen コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. オープン対象となる物理ファイルが，すべてオープン済みのときは，警告メッセージ（KFPS01283-W 又は KFPS02167-W）が出力されます。

(6) 注意事項

1. pdlogopen コマンドのリターンコードを次に示します。
   0：正常終了
   4：異常終了
   8：異常終了（オプション指定不正，rsh 失敗など）
2. ファイルグループの状態の変更中（スワップ処理中，又はほかの運用コマンドの処理中など）に pdlogopen コマンドを実行すると，エラーとなることがあります。
3. pdlogopen コマンドの結果は，pdlogls コマンドで確認できます。

2.54　pdlogrm（ログ関係のファイルの削除）

2.54.1　pdlogrm の形式と規則

(1)　機能

HiRDB ファイルシステム下のログ関係の物理ファイルを削除します。物理ファイルがオープン中の場合は，削除できません。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogrm -d sys | spd 〔-s サーバ名〕 -f 物理ファイル名 〔-u〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogrm -d sys | spd -s サーバ名 〔-X ユニット識別子〕 -f 物理ファイル名 〔-u〕 〔-D〕
```

(4)　オプション

(a)　-d sys ｜ spd

削除するログ関係のファイルを指定します。

sys：

システムログファイル

spd：

シンクポイントダンプファイル

(b)　-s サーバ名　～＜識別子＞((1～8))

物理ファイルで構成される，ファイルグループに対応するサーバの名称を指定します。

(c)　-f 物理ファイル名　～＜パス名＞((167 文字以内))

削除する物理ファイル名を絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。

(d)　-u

-f オプションで指定した物理ファイルがシステムログファイルの場合，そのファイルが次の状態であっても強制的にファイルを削除するときに指定します。

- アンロード待ち状態
- HiRDB Datareplicator 連携時の抽出未完了状態

このオプションを省略した場合，物理ファイルのアンロード状態がチェックされ，ファイルが上記のどれかの状態の場合はコマンドエラーとなります。

シンクポイントダンプファイルの場合は，指定しても無視されます。

このオプションは，何らかの障害によって通常の削除（-u 指定なしの pdlogrm コマンド）ができない場合に指定します。さらに，それでも削除できない場合には，pdfrm コマンドで削除します。

(e)　-X ユニット識別子　～＜識別子＞((4 文字))

影響分散スタンバイレス型系切り替え機能を適用している場合，削除ファイルのディスクがあるホストの，ユニット識別子を指定します。

該当するサーバが稼働中の場合，-X オプションの指定は無視されます。

次のどれかに該当する場合に-X オプションを指定すると，エラーとなります。

- 指定するユニットが，-s オプションで指定したサーバの移動先となる HA グループに属していない。
- -s オプションで指定したサーバが，HA グループ内のサーバではない。
- 影響分散スタンバイレス型系切り替え機能を適用していない。

-X オプションの指定要否を次に示します。

サーバの状態	ファイル作成先のディスクがあるホスト	-X オプションの指定
稼働中	実行系バックエンドサーバがあるホスト	省略できます。
停止中	現用系バックエンドサーバがあるホスト	
	上記以外のホスト	必ず指定します。

(f)　-D

このオプションは，pdchgconf コマンドでユニットの削除，又はサーバの削除をする場合に指定します。

ユニットの削除，又はサーバの削除をする場合，そのユニット又はサーバのシステムログファイル，シンクポイントダンプファイルを削除します。このオプションを指定した場合，サーバ名のチェックはしません。

このオプションを指定して pdlogrm コマンドを実行する場合，削除するシステムログファイル又はシンクポイントダンプファイルがあるユニットで実行してください。

(5) 規則

1. pdlogrm コマンドは，HiRDB の稼働に関係なく実行できます（ただし，HiRDB 開始処理中及び HiRDB 終了処理中は除きます）。
2. pdlogrm コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6) 注意事項

1. pdlogrm コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：異常終了

   8：異常終了（オプション指定不正，rsh 失敗など）

   12：異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）

2.55　pdlogswap（ログ関係のファイルのスワップ）

2.55.1　pdlogswap の形式と規則

(1)　機能

ログ関係のファイルをスワップします。スワップ先のファイルがない場合は，コマンドエラーとなります。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogswap -d sys 〔-s サーバ名〕 〔-w〔-t 有効化待ち監視時間〕〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogswap -d sys -s サーバ名 〔-w〔-t 有効化待ち監視時間〕〕
```

(4)　オプション

(a)　-d sys

システムログファイルをスワップします。

(b)　-s サーバ名　～＜識別子＞((1～8))

スワップするファイルグループに対応するサーバの名称を指定します。

(c)　-w

システムログファイルのスワップ後に取得するシンクポイントダンプの有効化完了まで，コマンド実行の終了を遅延させる場合に指定します。

アンロードレスシステムログ運用の場合に，データベースのバックアップ取得直前にこのオプションを指定してシステムログファイルをスワップしておくと，次に示す運用上の利点があります。

- データベースのバックアップ取得後にシステムログファイルをアンロード済みにする場合，バックアップ取得直前までに使用していたすべてのシステムログファイルをアンロード済みにできます。

- データベースの回復時，データベース回復ユティリティの入力としなければならないシステムログファイルの数を最小限にできます。

(d)　-t 有効化待ち監視時間　～＜符号なし整数＞((10～32768))《180》

-w オプションを指定してシンクポイントが有効化されるまでコマンド実行の終了を遅延させる場合に，シンクポイントダンプの有効化を監視する時間を秒単位で指定します。このオプションは，-w オプションを指定したときだけ有効となります。

(5)　規則

1. pdlogswap コマンドは，HiRDB が稼働中のときだけ実行できます。
   ただし，次の場合は実行できません。
   - HiRDB 開始処理中のとき
   - HiRDB 終了処理中のとき
   - pdstart -r コマンドで HiRDB が稼働中のとき
2. pdlogswap コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdlogswap コマンド実行中の場合は，再度 pdlogswap コマンドを実行できません。ただし，-w オプションを指定した pdlogswap コマンドがシンクポイントダンプの有効化待ちの状態のときには，-w オプションを指定しない pdlogswap コマンドを実行できます。

(6)　注意事項

1. pdlogswap コマンドの結果は，pdlogls コマンド，エラーメッセージ（KFPS01280-E（理由コードが 105 又は 116））の有無，又はコマンド実行時のリターンコードで確認できます。コマンド実行時のリターンコードは， echo %errorlevel%によって求められます。コマンド実行時のリターンコードを次に示します。

   0：

   正常終了しました。

   4：

   システムログファイルはスワップしましたが，-t オプションでの有効化待ち監視時間を経過してもシンクポイントダンプが有効化されなかったので，強制的にコマンド処理を終了しました。

   8：

   異常終了しました。
2. -w オプションを指定して pdlogswap コマンドを実行し，-t オプションでの有効化待ち監視時間を経過してもシンクポイントダンプが有効化されない場合には，pdlogswap コマンドは強制的に処理を終了します。
3. pd_log_remain_space_check オペランドに safe 又は warn を指定している場合，次のどちらかに該当するときは，システムログファイルの空き容量監視機能によってシステムログファイルの空き容量不足と判定されることがあります。

- スワップ先にできる状態，かつ待機の状態のファイルグループの容量が，現用のファイルグループの容量より少ない状態で，pdlogswap コマンドを実行した場合
- pdlogswap コマンドを連続して実行した場合

システムログファイルの空き容量監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

2.56　pdlogsync（シンクポイントダンプの取得）

2.56.1　pdlogsync の形式と規則

(1)　機能

シンクポイントダンプを取得します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogsync -d sys 〔-s サーバ名〕 〔-w〔-t タイムアウト時間〕〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogsync -d sys -s サーバ名 〔-w〔-t タイムアウト時間〕〕
```

(4)　オプション

(a)　-d sys

シンクポイントダンプを取得します。

(b)　-s サーバ名　～＜識別子＞((1～8))

シンクポイントダンプを取得するサーバの名称を指定します。

(c)　-w

シンクポイントダンプが有効化されるまで，pdlogsync コマンドの終了を待ち合わせる場合に指定します。このオプションを指定すると，pdlogsync コマンド実行時のリターンコードから，シンクポイントダンプが有効化されたかどうかが分かります。リターンコードが 0 の場合は，シンクポイントダンプが有効化されています。

バックアップ取得用のバッチコマンドを作成する場合，このオプションを指定します。

(d)　-t タイムアウト時間　～＜符号なし整数＞((10～32768))《180》

このオプションは，-w オプションを指定したときだけ有効となります。pdlogsync コマンドのタイムアウト時間を秒単位で指定します。

ここで指定した時間までにシンクポイントダンプが有効化されない場合は，pdlogsync コマンドを終了します。ただし，シンクポイントダンプの有効化処理は続いています。

シンクポイントダンプが有効化されると，メッセージ KFPS02183-I が出力されます。ただし，システム定義の pd_spd_assurance_msg オペランドで N を指定すると，メッセージ KFPS02183-I は出力されません。

(5)　規則

1. pdlogsync コマンドは，HiRDB が稼働中のときだけ実行できます。
   ただし，次の場合は実行できません。
   - HiRDB 開始処理中のとき
   - HiRDB 終了処理中のとき
   - pdstart -r コマンドで HiRDB が稼働中のとき
2. pdlogsync コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6)　注意事項

1. pdlogsync コマンドのリターンコードを次に示します。
   0：正常終了
   4：異常終了
   8：異常終了（オプション指定不正，rsh 失敗など）

2.57　pdlogucat（アンロードログファイルの情報表示）

2.57.1　pdlogucat の形式と規則

(1)　機能

アンロードログファイルの情報を表示します。

作成したアンロードログファイルの内容を調べたい場合に pdlogucat コマンドを使用します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdlogucat 〔-k {std|csv}〕 アンロードログファイル名
```

(4)　オプション

(a)　-k {std | csv}

情報の出力形式を指定します。

std：

固定フィールド＋ヘッダ情報の形式で出力します。

csv：

CSV 形式で出力します。

(5)　コマンド引数

(a)　アンロードログファイル名　～＜パス名＞

アンロードログファイルの名称を絶対パス名で指定します。

なお，HiRDB ファイルシステム領域に作成したアンロードログファイルを指定する場合，パス名は最大 167 文字となります。

(6)　規則

1. pdlogucat コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdlogucat コマンドは，アンロードログファイルがあるホストのサーバマシンで実行してください。

(7)　注意事項

1. pdlogunld コマンド，又は自動ログアンロード機能で作成中のアンロードログファイルに対しては，pdlogucat コマンドを実行しないでください。実行した場合，不正な情報が表示されることがあります。
2. pdlogucat コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(8)　出力形式

(a)　-k std の場合（出力情報は固定長）

```
Hostname: aa....aa(bbccdd)
File_type  Format_ver  System_id  Unit_id  Serv_name  Run_id    User_run_id  Server_name
Group     Svs_start_time       Gen_num   Unload_time          First_block_num  Buff_leng
Status
eee        fffff       gggg       hhhh     ii....ii   jjjjjjjj  kkkkkkkk     ll....ll
mm....mm  nnnn/oo/pp qq:rr:ss  tt....tt  uuuu/vv/ww xx:yy:zz  AA....AA         BB....BB
CC....CC
```

注

ヘッダの行は 1 行で表示され，その後にファイル情報が 1 行で表示されます。

表示できる情報がない項目には*が入ります。

バージョン 07-02 より前のアンロードログファイルを指定した場合，サーバラン ID，及びファイルグループ名は*となります。

aa....aa：

ホスト名（32 文字以内）

bbccdd：

pdlogucat コマンドを実行した時刻（時分秒）

eee：

ファイル種別

sys：システムログファイル

fffff：

システムが使用する内部情報

gggg：

HiRDB 識別子

hhhh：

ユニット識別子

ii....ii：

システムが使用する内部情報

jjjjjjj：

システムログラン ID（16 進数）

kkkkkkkk：

サーバラン ID（16 進数）

ll....ll：

サーバ名（8 文字以内）

mm....mm：

ファイルグループ名（8 文字以内）

nnnn/oo/pp qq:rr:ss：

現用のシステムログファイルの割り当て時刻（年/月/日 時:分:秒）

tt....tt：

世代番号（16 進数 8 文字以内）

uuuu/vv/ww xx:yy:zz：

アンロードログファイル作成開始時刻（年/月/日 時:分:秒）

AA....AA：

先頭ログブロック番号（16 進数 8 文字以内）

BB....BB：

システムが使用する内部情報

CC....CC：

システムログのアンロードの完了状態

completed：正しく作成されたアンロードログファイル

failed：作成途中で中断されたアンロードログファイル

(b)　-k csv の場合（出力情報は可変長）

```
aaa,bbbbb,cccc,dddd,ee....ee,ffffffff,gggggggg,hh....hh,ii....ii,jjjjjjjj,kk....kk,
llllllll,mm....mm,nnnnnn,o
```

注

表示できる情報がない場合，その項目は空要素として表示されます。例えば，システムログラン ID が表示されない場合は「aaa,bbbbb,cccc,dddd,ee....ee,,gggggggg,...」となります。

バージョン 07-02 より前のアンロードログファイルを指定した場合，サーバラン ID，及びファイルグループ名は空要素として表示されます。

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

aaa：

ファイル種別

sys：システムログファイル

bbbbb：

システムが使用する内部情報

cccc：

HiRDB 識別子

dddd：

ユニット識別子

ee....ee：

システムが使用する内部情報

ffffffff：

システムログラン ID（16 進数）

gggggggg：

サーバラン ID（16 進数）

hh....hh：

サーバ名（8 文字以内）

ii....ii：

ファイルグループ名（8 文字以内）

jjjjjjj：

現用のシステムログファイルの割り当て時刻（16 進数通算秒）

kk....kk：

世代番号（16 進数 8 文字以内）

lllllll：

アンロードログファイル作成開始時刻（16 進数通算秒）

mm....mm：

先頭ログブロック番号（16 進数 8 文字以内）

nnnnnn：

システムが使用する内部情報

o：

システムログのアンロードの完了状態

0：正しく作成されたアンロードログファイル

1：作成途中で中断されたアンロードログファイル

2.58　pdlogunld（ログ関係のファイルのアンロード）

2.58.1　pdlogunld の形式と規則

(1)　機能

指定したファイルグループを出力先のファイルへアンロードし，ファイルグループをアンロード済み状態にします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdlogunld -d sys 〔-s サーバ名〕 -g ファイルグループ名
                 〔-o 出力先ファイル名〕 〔-n〕 〔-f〕
                 〔-k 出力先ファイル種別〕
                 〔-i 出力先ファイルの初期容量〔，増分容量〕〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdlogunld -d sys -s サーバ名 〔-u ユニット識別子〕 -g ファイルグループ名
                 〔-o 出力先ファイル名〕 〔-n〕 〔-f〕
                 〔-k 出力先ファイル種別〕
                 〔-i 出力先ファイルの初期容量〔，増分容量〕〕
```

(4)　オプション

(a)　-d sys

システムログファイルをアンロードします。

(b)　-s サーバ名　～＜識別子＞((1～8))

ファイルグループに対応するサーバの名称を指定します。

(c)　-g ファイルグループ名　～＜識別子＞((1～8))

アンロードするファイルのファイルグループ名を指定します。

(d)　-o 出力先ファイル名　～＜パス名＞((-k hirdb 指定時は 167 文字以内))

出力先のアンロードログファイルの名称を指定します。

出力先ファイルは，-s オプションに指定したサーバがあるホストに作成されます。省略した場合，標準出力へ出力します。

-k hirdb を指定した場合，出力先ファイル名は「HiRDB ファイルシステム領域名¥HiRDB ファイル名」の形式で指定してください。

(e)　-n

ファイルグループの状態を変更しないで，アンロード待ちの状態を継続する場合に指定します。このオプションを指定することで，同一世代のファイルを繰り返してアンロードでき，複数のアンロードファイルを作成できます。

(f)　-f

ファイルグループの状態をチェックしないで，アンロードする場合に指定します。このオプションを指定した場合，現用ファイルグループ又はアンロード済みのファイルグループでもアンロードできます。ただし，ファイルグループの状態は変更されません。

(g)　-k 出力先ファイル種別

-o オプション指定時に，出力先のアンロードログファイルの種別を指定します。省略した場合，regular が仮定されます。

regular：

ファイルに出力する場合に指定します。

hirdb：

HiRDB ファイルに出力する場合に指定します。系切り替え機能を使用する場合に指定してください。

-k regular を指定する場合は，アンロードログファイルを格納するディレクトリを作成しておく必要があります。また，作成するディレクトリには，HiRDB 管理者に対して読み込み権限，書き込み権限，及び実行権限を与えてください。

-k hirdb を指定する場合は，pdfmkfs コマンド（-k UTL）で HiRDB ファイルシステム領域を作成しておく必要があります。

(h)　-i 出力先ファイルの初期容量〔，増分容量〕

～＜符号なし整数＞((1～1048574))

-k hirdb を指定した場合に，出力先ファイルの初期容量と増分容量（初期容量を超えた場合に確保する容量）をメガバイト単位で指定します。

省略した場合は，次の値が仮定されます。

-i の指定値	システムログファイルサイズ	
	2400MB 未満	2400MB 以上
出力先ファイルの初期容量	100MB	システムログファイルサイズ÷24
増分容量	100MB	（システムログファイルサイズ−出力ファイルの初期容量）÷23

＜規則＞

1. アンロードログファイルの最大サイズはシステムログファイルの最大サイズと同じ（100 ギガバイト）であるため，通常は-i オプションを指定する必要はありません。ただし，システムログファイルが 100 メガバイトより小さい場合は，そのシステムログファイルのサイズを-i オプションに指定してください。
2. 2 ギガバイト以上の HiRDB ファイルシステム領域使用時は，-i オプションを指定しないでください。
3. 初期容量は，pdfmkfs コマンドで HiRDB ファイルシステム領域を作成したときの-n オプションの指定値より小さくしてください。
4. 次の場合には，出力先ファイルの増分はできません。
   - HiRDB ファイルシステム領域の増分回数指定を超えて拡張しようとした場合

     HiRDB ファイルシステム領域の増分回数指定は pdfmkfs コマンドの-e オプションで指定します。-k hirdb を指定する場合は，マニュアル「HiRDB システム導入・設計ガイド」の「最大増分回数の見積もり」を参照して-e オプションに指定する値を見積もってください。
   - HiRDB ファイルが，個々の HiRDB ファイルの増分回数の上限（23 回）を超えて増分しようとした場合

(i)　-u ユニット識別子　～＜識別子＞((4 文字))

影響分散スタンバイレス型系切り替え機能を適用している場合，アンロードするシステムログファイルを作成したディスクがあるホストの，ユニット識別子を指定します。

該当するサーバ，及びシステムマネジャが稼働中の場合，-u オプションの指定は無視されます。

次のどれかに該当する場合に-u オプションを指定すると，エラーとなります。

- 指定するユニットが，-s オプションで指定したサーバの移動先となる HA グループに属していない。
- -s オプションで指定したサーバが，HA グループ内のサーバではない。
- 影響分散スタンバイレス型系切り替え機能を適用していない。

-u オプションの指定要否を次に示します。

サーバの状態	ファイル作成先のディスクがあるホスト	-u オプションの指定
稼働中	実行系バックエンドサーバがあるホスト	省略できます。
停止中	現用系バックエンドサーバがあるホスト	
	上記以外のホスト	必ず指定します。

(5) 規則

1. pdlogunld コマンドの実行条件を次に示します。
   - システム定義で pd_log_unload_check=N を指定して HiRDB が稼働中の場合で，かつ操作対象のファイルグループがオープン状態のときは，-f オプションの指定がないと実行できません。
   - 上記以外の場合，HiRDB の稼働に関係なく実行できます（ただし，HiRDB 開始処理中及び HiRDB 終了処理中は除きます）。
2. pdlogunld コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 現用ファイルグループ，及びアンロード済みのファイルグループは-f オプションを指定しないとアンロードできません。

(6) 注意事項

1. pdlogunld コマンドのリターンコードを次に示します。
   0：正常終了
   4：異常終了
   8：異常終了（オプション指定不正，rsh 失敗など）
   12：異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）
2. スワップ先にできる状態のファイルグループを，-f オプションを指定してアンロードすると，アンロード中に現用ファイルグループとなることがあります。アンロード中に現用ファイルグループとなった場合には，コマンドエラーとなります。そのため，スワップ先にできる状態のファイルグループは，-f オプションを指定してアンロードする場合，アンロード前に pdlogcls コマンドでクローズする必要があります。
3. pdlogunld コマンドは，HiRDB 開始処理中に実行しないでください。HiRDB 開始処理中とは，KFPS01800-I メッセージの出力から KFPS05210-I メッセージの出力までの間のことをいいます。実行した場合，システムログファイルの状態が HiRDB に反映されないことがあります。このような場合には，HiRDB 開始処理完了後に pdlogcls コマンドで該当するファイルグループをクローズし，その後 pdlogopen コマンドでオープンすると，システムログファイルの状態が反映されます。
4. pdlogunld コマンドは，HiRDB システム定義ファイルの内容を参照して，処理をします。そのため，ログを取得した稼働中の HiRDB の HiRDB システム定義ファイルと，pdlogunld コマンド実行時の HiRDB システム定義ファイルの内容が一致していない場合は，アンロードできないことがあります。
5. pdlogunld コマンドの結果は，次の方法で確認できます。
   - HiRDB 稼働中の場合

pdlogls コマンド

- HiRDB 停止中の場合

  完了メッセージ

6. HiRDB が稼働していないときに pdlogunld コマンドを実行し，コマンドが終了する前に次に示すどれかの状態になると，コマンド実行エラーとなります。

   - HiRDB 開始中
   - HiRDB オンライン中
   - HiRDB 停止処理中

   このとき作成されたアンロードログファイルはデータベースの回復処理に使用できません。HiRDB の稼働状態を確認してから，再度 pdlogunld コマンドを実行してください。

7. pdstop -s コマンドで個別に終了したサーバでは，そのサーバで pd_log_unload_check=N を指定している場合，-f オプションの指定がないと pdlogunld コマンドは実行できません。

8. pdlogunld コマンドで HiRDB ファイルシステム領域に作成したアンロードログファイルを，pdfbkup コマンド及び pdfrstr コマンドで別の HiRDB ファイルシステム領域に移動する場合，移動するアンロードログファイルのレコード長は，pdfrstr コマンド実行時に使用する HiRDB ファイルシステム領域のセクタ長の整数倍でなければなりません。アンロードログファイルのレコード長と，pdfrstr コマンド実行時に使用する HiRDB ファイルシステム領域のセクタ長の関係を次に示します（単位：バイト）。

アンロードログファイルのレコード長	pdfrstr コマンド実行時に使用する HiRDB ファイルシステム領域のセクタ長		
	1,024	2,048	4,096
1,024	○	×	×
2,048	○	○	×
4,096	○	○	○

（凡例）

○：pdfrstr コマンドを実行できます。

×：pdfrstr コマンドは実行できません。

アンロードログファイルのレコード長は，pdfls コマンドで確認できます。また，現在のセクタ長は pdfstatfs コマンドで確認できます（HiRDB ファイルシステム領域のセクタ長は，pdfmkfs コマンド実行時に-s オプションで指定した値となります）。

2.59　pdls（HiRDB システムの状態表示）

2.59.1　pdls の形式と規則

(1)　共通機能

指定した表示対象種別ごとに HiRDB システムの状態を表示します。

(2)　共通実行者

HiRDB 管理者が実行できます。

(3)　共通形式

```
pdls 〔-d 表示対象種別 〔表示対象種別ごとのオプション〕〕
```

注

表示対象種別ごとのオプションは，指定する表示対象種別によって異なります。

(4)　共通オプション

(a)　-d 表示対象種別　～《svr》

状態を表示する対象種別を指定します。

act：

ユーザの状態を表示します。

aud：

監査証跡ファイルの状態を表示します。

ha：

系切り替え機能を使用しているときの，系の状態を表示します。

lck：

サーバの排他制御の状態を表示します。

mem：

サーバの共用メモリの情報を表示します。

prc：

サーバのプロセスの状態を表示します。

rpc：

サーバの通信制御情報を表示します。

rpl：

HiRDB Datareplicator 連携の状態を表示します。

scd：

サーバのスケジュールの状態を表示します。

stj：

ユニット，及びサーバごとの統計情報出力種別の指定有無を表示します。

sts：

ステータスファイルの状態を表示します。

svr：

ユニット及びサーバの状態を表示します。

trn：

サーバのトランザクションの状態を表示します。

ust：

ユニットの稼働状態を表示します。

(5) 共通規則

1. -d オプションに aud，ha，rpl，及び ust を指定した pdls コマンドは，HiRDB の稼働に関係なく実行できます。それ以外のオプションを指定した pdls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdls コマンドは，-d ha，lck，svr，trn，及び ust の場合，任意のサーバマシンから実行できます。それ以外の場合，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 表示対象種別ごとのオプションに，ほかの表示対象種別で指定できるオプションを指定した場合は，そのオプションは無視されます。

(6) 共通注意事項

1. システム定義の pd_tmp_directory オペランドを指定しないで，環境変数 TMP だけを指定している場合，pdls コマンド，又はほかの HiRDB のコマンドを実行中に中断すると，環境変数 TMP に設定したディレクトリ下に"pdcmd"で始まる名称のファイルが残ることがあります。pdls コマンド，又はほかの HiRDB のコマンドが終了した後も，"pdcmd"で始まる名称のファイルが残っている場合は，エクスプローラなどで削除してください。
2. 影響分散スタンバイレス型系切り替え機能を適用している場合，系切り替えでほかのユニットに移動したサーバは，移動先のユニットのサーバとして扱います。
3. コマンドを実行するユニットが次の場合，各ユニットの稼働状態を正しく確認できない場合があります。
   - 系切り替え処理中の場合

- ユニットの状態を検知するまでの間（システム定義の pd_host_watch_interval オペランドに指定した時間）

2.60　pdls【-d act】（ユーザの状態表示）

2.60.1　pdls【-d act】の形式と規則

(1)　機能

ユーザの状態を表示します。

(2)　形式

```
pdls -d act 〔-s サーバ名〕
            〔{-x ホスト名〔，ホスト名〕… | -u ユニット識別子〔，ユニット識別子〕…}〕
```

(3)　オプション

(a)　-s サーバ名　～＜識別子＞((1～8))

指定したサーバのユーザの状態を表示します。省略した場合，HiRDB システム内のすべてのサーバに関する，ユーザの状態を表示します。

(b)　-x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-x オプションの指定は無視されます。また，複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(c)　-u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-u オプションの指定は無視されます。また，複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4)　注意事項

1. pdls -d act コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。8 の場合は，その前に出力されているメッセージを参照してエラー要因を取り除いてください。

(5) 出力形式

```
HOSTNAME : aa...a(bbccdd)
SVID    PID    ACTID                 LOCK-STATE
ee...e  ff...f gg...g-hh...h-ii...i  j
```

[説明]

SVID「サーバ名」から LOCK-STATE「UNTIL DISCONNECT 指定の排他保持状態」までの各項目は，1 文字以上の半角空白を空けて表示します。

aa...a ：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e ：

サーバ名（8 文字以内）

ff...f ：

プロセス ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，UID（ユーザ ID）以降の項目の表示位置は右にシフトします。

ユニットの再開始で復元したユーザ情報の場合は，0 を表示します。

gg...g-hh...h-ii...i：

ユーザ識別通番（22 文字以内）

20 文字以上の場合，LOCK-STATE（UNTIL DISCONNECT 指定の排他保持状態）以降の項目の表示位置が右にシフトします。

j：

UNTIL DISCONNECT 指定の排他保持状態

Y ：排他を保持しています

N ：排他を保持していません

＜出力例＞

プロセス ID が 100000 以上の表示を含む場合

```
HOSTNAME : HOST0001(144031)
SVID      PID     ACTID             LOCK-STATE
SERVER01  38095   16-1-1            N          … プロセスIDが99999以下
SERVER01  108221   16-1-2            N         … プロセスIDが100000以上
SERVER01   8101   16-1-3            N          … プロセスIDが99999以下
```

2.61　pdls【-d aud】（監査証跡ファイルの状態表示）

2.61.1　pdls【-d aud】の形式と規則

(1)　機能

監査証跡ファイルの状態を表示します。

(2)　形式

```
pdls -d aud 〔-u ユニット識別子〕 〔-U 認可識別子〕
```

(3)　オプション

(a)　-u ユニット識別子　～＜識別子＞((4 文字))

情報を表示するユニットのユニット識別子を指定します。

-u オプションは，HiRDB/パラレルサーバの場合に指定できます。また，-u オプションを省略した場合，すべてのユニットの情報を表示します。

(b)　-U 認可識別子

監査人の認可識別子を指定します。このオプションを指定すると，パスワード入力要求の応答付きメッセージが出力されます。

このオプションは，HiRDB が稼働中の場合の pdls -d aud コマンド実行時に有効となります。

＜規則＞

1. HiRDB 管理者がコマンドを実行すると，パスワード入力要求のメッセージが表示されます。このとき，監査人が認可識別子とパスワードを入力してください。なお，HiRDB が停止中の場合，HiRDB にアクセスするときに認証を行いません。
2. このオプションに指定した認可識別子と，応答付きメッセージに入力したパスワードで HiRDB に接続し，その認可識別子に監査権限があるかをチェックします。
3. このオプションは，応答付きメッセージに応答できる環境で pdls -d aud コマンドを実行する場合に指定できます。
4. このオプションを省略した場合，pdls -d aud コマンド実行時の環境変数 PDUSER の設定値が仮定されます。このオプションを省略し，かつ環境変数 PDUSER も設定していない場合，pdls -d aud コマンドは実行できません。このオプションを省略する場合は，環境変数 PDUSER を設定しておく必要があります（指定例を次に示します）。

   （例）

PDUSER=¥""認可識別子"¥"/¥""パスワード"¥"

(4) 注意事項

1. pdls -d aud コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. pdls -d aud コマンドは，ユニットの開始処理途中，終了処理途中，及び pdstart -r で HiRDB を起動したときは実行できません。
3. システム定義に pd_aud_file_name オペランドを指定していない場合，又は pd_aud_file_name オペランドに指定している監査証跡ファイルがない場合に，pdls -d aud コマンドを実行するとコマンドエラーになります。
4. HiRDB/パラレルサーバの場合に，-u オプションを省略して pdls -d aud コマンドを実行したときは，コマンド処理中にエラーが発生したユニットについてはエラーメッセージを出力し，エラーが発生していないユニットについては監査証跡ファイルの情報を表示します。
5. ユニット停止中にユニット用ステータスファイルを初期化してからユニットを開始した場合，障害が発生して閉塞中となっている世代の監査証跡ファイルのデータロード状態の表示は，–となります。

(5) 出力形式

```
HOSTNAME     : aa...a(bbccdd)
UNITID       : eeee
Audit-Status : ff...f
File-Name    : gg...g
Generation-Name    Status
hh...h             ijk
   :                :
```

すべてのユニットの情報を表示する場合は，ユニット数分上記の情報が表示されます。

[説明]

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

eeee：

ユニット識別子（4 文字）

ff...f：

監査証跡の取得状態（8 文字以内）

ACTIVE：取得中です。

INACTIVE：停止中です。

**********：HiRDB が稼働中でないため，監査証跡の取得状態を表示できません。

gg...g：

監査証跡ファイル用の HiRDB ファイルシステム領域名（絶対パス名 117 文字以内）

監査証跡ファイル名が 117 文字を超える場合は，先頭から 117 文字を出力します。

hh...h：

監査証跡ファイルの世代ごとのファイル名（16 文字）

監査証跡ファイルが 1 世代未満の場合は表示されません。

ijk：

監査証跡ファイルの世代ごとの状態（3 文字）

監査証跡ファイルが 1 世代未満の場合は表示されません。

i：監査証跡ファイルの現在の状態

c：現用。現時点の監査証跡の出力対象世代です。監査証跡の取得が停止している場合は，次回の監査証跡の取得開始時に現用となることを示しています。

s：待機中。現時点では監査証跡の出力対象世代ではありません。障害が発生して閉塞中となった監査証跡ファイルについては，ユニット停止中は s が表示されます。

j：監査証跡ファイルのデータロード状態

d：データロード待ち。該当する世代の監査証跡ファイルは，監査証跡表にデータロードされていません。

-：データロード済み。該当する世代の監査証跡ファイルは，監査証跡表にデータロード済みです。障害が発生して閉塞中の監査証跡ファイルについては，ユニット停止中は-が表示されます。

k：監査証跡ファイルの閉塞状態

h：閉塞中。監査証跡ファイルに障害が発生し，以後監査証跡が出力されていません。

-：閉塞していません。

2.62　pdls【-d ha】（HiRDB の系の状態表示）

2.62.1　pdls【-d ha】の形式と規則

(1)　機能

IP アドレスを引き継がない系切り替え機能，又はスタンバイレス型系切り替え機能を使用しているときの，HiRDB の系の状態を表示します。

(2)　形式

```
pdls -d ha 〔 {-u ユニット識別子 | -s サーバ名} 〕 〔-a〕
```

(3)　オプション

(a)　-u ユニット識別子　～＜識別子＞((4 文字))

系の状態を表示するユニットのユニット識別子を指定します。省略した場合，すべてのユニットについて系の状態を表示します。

影響分散スタンバイレス型系切り替え機能を適用しているユニットのユニット識別子を指定した場合，そのユニット内のホスト BES とゲスト BES だけ状態表示の対象となります。

(b)　-s サーバ名　～＜識別子＞((1～8))

影響分散スタンバイレス型系切り替え機能を適用している場合，状態表示をするサーバのサーバ名を指定します。影響分散スタンバイレス型系切り替え機能を適用していないユニットのサーバ名を指定するとエラーになります（KFPS05041-E メッセージ）。

(c)　-a

影響分散スタンバイレス型系切り替え機能を適用している場合，実行状態だけではなく，詳細情報（待機状態，受け入れ可能状態，及び実行系起動待ち状態）も表示するときに指定します。

＜規則＞

1. 影響分散スタンバイレス型系切り替え機能を適用していない場合に-a オプションを指定しても無視されます。
2. -a オプションと-u オプションを指定した場合，指定したユニット内のホスト BES とゲスト BES の詳細情報が表示されます。
3. -a オプションと-s オプションを指定した場合，指定したサーバの HA グループ内の各ユニットでの詳細情報が表示されます。

4. -a オプションを省略した場合，実行中のサーバの情報が表示されます。

5. 受け入れ可能状態のサーバを確認したい場合は，-a オプションを指定してください。

6. -a，-u，及び-s オプションの組み合わせによって，表示される情報は次のようになります。

オプション			稼働しているサーバ			停止しているサーバ
-a	-u	-s	実行状態	待機状態及び受け入れ可能状態	実行系起動待ち状態	
あり	あり	あり	×	×	×	×
		なし	○	○	○	○
	なし	あり	○	○	○	○
		なし	○	○	○	○
なし	あり	あり	×	×	×	×
		なし	○	−	−	−
	なし	あり	○	−	−	○※
		なし	○	−	−	○※

（凡例）

○：表示されます。

−：表示対象外です。

×：エラーとなります。

注※

HA グループ内の全ユニットの，実行系でないサーバの状態が表示されます。

(4) 注意事項

1. pdls -d ha コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. 系切り替え機能を使用していない場合，又は IP アドレスを引き継ぐ系切り替え機能を使用している場合に，このコマンドを実行したときはメッセージを表示してコマンドの処理を終了します。

(5) 出力形式

(a) 影響分散スタンバイレス型系切り替え機能を適用していない場合

```
UNITID  INITIAL-HOST [STATUS]  RESERVED-HOST [STATUS]
aaaa    bb...bb      [ccc]     dd...dd       [eee]
  :       :            :         :             :
```

[説明]

aaaa：

ユニット識別子（4 文字）

bb...b：

現用系のホスト名（32 文字以内）

ccc：

現用系のホストの状態（3 文字）

モニタモードの場合：

ONL：実行中（オンライン）

SBY：待機中（停止も含みます）

STP：両系とも停止

サーバモードの場合：

ONL：実行中（オンライン）

SBY：待機中※

STP：停止

系切り替え対象外のホストの場合：

ONL：実行中（オンライン）

STP：停止

dd...d：

予備系のホスト名（32 文字以内）

該当するユニットが系切り替えの対象でない場合，又は IP アドレスを引き継ぐ系切り替え機能を使用している場合，**********を表示します。

eee：

予備系のホストの状態（3 文字）

モニタモードの場合：

ONL：実行中（オンライン）

SBY：待機中（停止も含みます）

STP：両系とも停止

*******：系切り替え対象外，又は IP アドレスを引き継ぐ系切り替え機能使用時

サーバモードの場合：

ONL：実行中（オンライン）

SBY：待機中※

STP：停止

*******：系切り替え対象外，又は IP アドレスを引き継ぐ系切り替え機能使用時

注※

1:1 スタンバイレス型系切り替え機能を使用している場合に代替 BES が SBY となるのは，代替 BES が代替準備を完了した状態であり，代替 BES ユニットが開始した状態ではありません。

出力形式についての補足説明：

1. 系の状態と表示結果を次に示します。

系の状態		表示結果					
		モニタモード		サーバモード		系切り替え対象外	
現用系	予備系	現用系	予備系	現用系	予備系	現用系	予備系
停止	停止	STP	STP	STP	STP	STP	***
停止	待機中※	−	−	STP	SBY	STP	−
稼働中	停止	ONL	SBY	ONL	STP	ONL	***
稼働中	待機中	ONL	SBY	ONL	SBY	ONL	−
停止	稼働中	SBY	ONL	STP	ONL	−	−
待機中	稼働中	SBY	ONL	SBY	ONL	−	−
待機中※	停止	−	−	SBY	STP	−	−

（凡例）−：該当しません。

注※

待機系だけの起動はできませんが，Hitachi HA Toolkit Extension 使用時は両系起動中に実行系を正常終了した場合，待機系は自動停止しないため，待機系だけが起動している状態になります。

2. 現用系ホスト名，及び予備系ホスト名の長さが１行に収まらない場合，予備系ホストに関する情報を次の行に表示します。例を次に示します。

```
UNITID  INITIAL-HOST [STATUS]  RESERVED-HOST [STATUS]
aaaa    bb.....................................................bb      [ccc]
        dd.....................................................dd      [eee]
```

3. モニタモード，サーバモードと，pdls -d ha コマンドの表示結果の対応を次の図に示します。

図 2–4　モニタモード，サーバモードと，pdls -d ha コマンドの表示結果の対応

●現用系で開始して，系切り替えをした後に予備系で正常終了した場合

・系の状態遷移

現用系	予備系
停止	停止
稼働中	：
：	：
：	待機中
	：
系切り替え	系切り替え
停止	稼働中
：	：
：	：
	停止

・pdls -d haコマンドでの表示結果

モニタモードの場合		サーバモードの場合	
現用系	予備系	現用系	予備系
STP	STP	STP	STP
ONL	SBY	ONL	：
：	：	：	：
：	：	：	SBY
			：
			：
SBY	ONL	STP	ONL
：	：	：	：
：	：	：	：
STP	STP		STP

●現用系で開始して，系切り替えをしないで現用系で正常終了した場合

・系の状態遷移

現用系	予備系
停止	停止
稼働中	：
：	：
：	待機中
	：
停止	停止

・pdls -d haコマンドでの表示結果

モニタモードの場合		サーバモードの場合	
現用系	予備系	現用系	予備系
STP	STP	STP	STP
ONL	SBY	ONL	：
：	：	：	：
：	：	：	SBY
			：
STP	STP	STP	STP

4. 1:1 スタンバイレス型系切り替え機能を使用している場合，現用系，予備系，実行中，及び待機中のユニットの意味は次のようになります。

   現用系：正規 BES ユニット

   予備系：代替 BES ユニット

   実行中：処理を実行しているユニット（非代替中の正規 BES ユニット又は代替中の代替 BES ユニット）

   待機中：待機しているユニット（非代替中の代替 BES ユニット又は代替中の正規 BES ユニット）

(b)　影響分散スタンバイレス型系切り替え機能を適用している場合

```
UNITID  INITIAL-HOST [STATUS]  RESERVED-HOST [STATUS]
aaaa    bb...bb      [ccc]     dd...dd       [eee]
  :       :            :         :             :

SVID    UNIT  [STATUS] INITIAL-UNIT
ff...f  gggg  [hhh]    iiii
  :       :     :        :
```

[説明]

aaaa～eee の内容は，影響分散スタンバイレス型系切り替え機能を適用していない場合と同じです。

ff...f～iiii は，サーバごとに，サーバがあるユニットのユニット識別子でソートして表示されます。

ff... f：

サーバ名（8 文字以内）

影響分散スタンバイレス型系切り替え機能を適用したユニットのサーバです。

gggg：

サーバがあるユニットのユニット識別子（4 文字）

ff...f のサーバがあるユニットのユニット識別子です。

-u オプションを指定している場合は，指定したユニット識別子となります。

-a オプションを指定して，-u オプションを省略している場合，ff...f のサーバの，移動先 HA グループの全ユニットのうち，表示対象となるユニットのユニット識別子となります。

-a オプションと-u オプションを省略している場合，ff...f のサーバが実行系となっているユニットのユニット識別子となります。

ff...f のサーバが HA グループの全ユニットで実行状態となっていない場合，****が表示されます。

hhh：

サーバのステータス（3 文字）

ONL：

実行状態（オンライン）

SBY：

ホスト BES の場合は待機状態，ゲスト BES の場合は受け入れ可能状態

-a オプションを省略している場合，表示されません。

STP：

-a，-u，及び-s オプションを省略している場合，HA グループ内の全ユニットでの実行状態以外の状態であることを示します。

-a オプションを指定している場合，gggg のユニットで停止していることを示します。

WIT：

実行系起動待ち状態

-a オプションを省略している場合は表示されません。

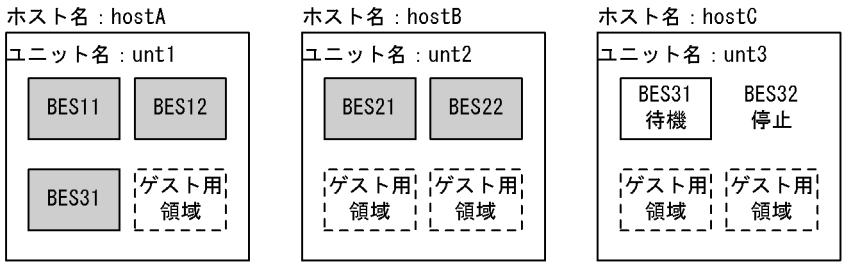
iiii：

定義先ユニットのユニット識別子（4 文字）

ff...f のサーバの，システム定義の pdstart -u で指定したユニット識別子です。

影響分散スタンバイレス型系切り替え機能を適用した場合の表示例：

影響分散スタンバイレス型系切り替え機能を適用した場合の構成例を次に示します。



（凡例）：稼働中のサーバを示します。

上記の構成の場合に pdls -d ha コマンドを実行すると，次のように情報が表示されます。

- pdls -d ha の場合

```
SVID  UNIT  [STATUS]  INITIAL-UNIT
BES11 unt1  [ONL]     unt1
BES12 unt1  [ONL]     unt1
BES31 unt1  [ONL]     unt3
BES21 unt2  [ONL]     unt2
BES22 unt2  [ONL]     unt2
BES32 ****  [STP]     unt3
```

- pdls -d ha -a の場合

```
SVID  UNIT  [STATUS]  INITIAL-UNIT
BES11 unt1  [ONL]     unt1
BES12 unt1  [ONL]     unt1
BES21 unt1  [SBY]     unt2
BES22 unt1  [SBY]     unt2
BES31 unt1  [ONL]     unt3
BES32 unt1  [STP]     unt3
BES11 unt2  [SBY]     unt1
BES12 unt2  [SBY]     unt1
BES21 unt2  [ONL]     unt2
BES22 unt2  [ONL]     unt2
BES31 unt2  [SBY]     unt3
BES32 unt2  [STP]     unt3
BES11 unt3  [SBY]     unt1
BES12 unt3  [STP]     unt1
BES21 unt3  [SBY]     unt2
BES22 unt3  [SBY]     unt2
BES31 unt3  [SBY]     unt3
BES32 unt3  [STP]     unt3
```

- pdls -d ha -u unt1 の場合

```
SVID  UNIT  [STATUS]  INITIAL-UNIT
BES11 unt1  [ONL]     unt1
BES12 unt1  [ONL]     unt1
BES31 unt1  [ONL]     unt3
```

- pdls -d ha -u unt1 -a の場合

```
SVID  UNIT  [STATUS]  INITIAL-UNIT
BES11 unt1  [ONL]     unt1
```

```
BES12 unt1  [ONL]     unt1
BES21 unt1  [SBY]     unt2
BES22 unt1  [SBY]     unt2
BES31 unt1  [ONL]     unt3
BES32 unt1  [STP]     unt3
```

- pdls -d ha -s BES12 の場合

```
SVID  UNIT  [STATUS]  INITIAL-UNIT
BES12 unt1  [ONL]     unt1
```

- pdls -d ha -s BES12 -a の場合

```
SVID  UNIT  [STATUS]  INITIAL-UNIT
BES12 unt1  [ONL]     unt1
BES12 unt2  [SBY]     unt1
BES12 unt3  [STP]     unt1
```

2.63　pdls【-d lck】（サーバの排他制御の状態表示）

2.63.1　pdls【-d lck】の形式と規則

(1)　機能

サーバの排他制御の状態（占有情報及び待ち情報），排他資源管理テーブルの使用率，又は排他制御用プールパーティションの統計情報を表示します。

(2)　形式

```
pdls -d lck　{〔-a〕〔-s サーバ名〕｜〔-p〔-e〕〕｜〔-T〔-R〕〕}
            〔{-x ホスト名〔，ホスト名〕…｜-u ユニット識別子〔，ユニット識別子〕…}〕
```

(3)　オプション

サーバの排他制御の状態（占有情報及び待ち情報），排他資源管理テーブルの使用率，又は排他制御用プールパーティションの統計情報を表示するかを指定します。オプションを省略した場合，待ち情報だけが表示されます。

(a)　-a

占有情報及び待ち情報を表示したい場合に指定します。

(b)　-s サーバ名　～＜識別子＞((1～8))

指定したサーバに関する待ち情報だけを表示します。省略した場合，すべてのサーバの情報を表示します。

(c)　-p

各サーバの排他資源管理テーブルの使用率を表示したい場合に指定します。

(d)　-e

各サーバの排他制御用プールパーティションの詳細情報を表示したい場合に指定します。

(e)　-T

排他制御用プールパーティションの統計情報を表示したい場合に指定します。表示された情報は，排他制御用プールパーティションの数をチューニングするときに使用できます。排他制御用プールパーティションのチューニング方法は，マニュアル「HiRDB システム運用ガイド」の「排他制御用プールを分割する方法」を参照してください。

-T オプションを指定する場合に出力される情報の中で，次に示す情報は，表示できるけた数を超えると 0 に戻ります。0 に戻らないようにするためには，-R オプションも同時に指定してください。

- 排他要求回数
- 排他制御用プールパーティションのスレッド間ロック要求回数

(f) -R

前回のコマンド実行時から，コマンド実行時点までの統計情報を集計したい場合に指定します。前回との差分の統計情報を取得するため，このコマンドを多重実行しないでください。なお，HiRDB を開始してから最初にこのコマンドを実行する場合，HiRDB 開始時からコマンド実行時点までの統計情報を集計します。

省略した場合，HiRDB を開始したときからコマンドを実行した時点までの統計情報を集計します。

(g) -x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-x オプションの指定は無視されます。また，複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(h) -u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-u オプションの指定は無視されます。また，複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4) 注意事項

1. pdls -d lck コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(5) 出力形式

(a) オプション省略時，又は-a，-s オプションを指定した場合

```
HOSTNAME : aa...a(bbccdd)
SVID         PID    TID KIND  RESOURCE
 WAIT  MODE  TRNBID             ACTID               LOCK-STATE  DPRIO
ee...e    ff...f gg...g hhhh  ii...i
jj...j kk    ll...l             mm...m-nn...n-oo...o p            qqq   ...1
                        :
```

[説明]

1 は，占有及び待ち情報の数だけ繰り返し表示されます。

1 で表示する各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e：

サーバ名（8 文字以内）

ff...f：

プロセス ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，TID（スレッド ID）以降の項目の表示位置は右にシフトします。

gg...g：

スレッド ID（10 進数 5 けた）

hhhh：

資源種別（16 進数 4 けた）

ii...i：

資源名称（16 進数 28 けた）

jj...j：

待ち時間（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示，単位は秒）

100000 以上の場合，MODE（排他制御モード）以降の項目の表示位置は右にシフトします。

占有情報の場合は*****を表示

kk：

排他制御モード（2 文字）

ll...l：

トランザクション識別子（16 文字）

p（UNTIL DISCONNECT 指定の排他保持状態）に Y が表示されている場合，このトランザクションブランチは既に終了していることがあります。そのため，タイミングによっては pdls -d trn コマンドで表示されないときがあります。

mm...m-nn...n-oo...o：

ユーザ識別通番（22 文字以内）

19 文字以上の場合，LOCK-STATE（UNTIL DISCONNECT 指定の排他保持状態）以降の項目の表示位置は右にシフトします。

p：

UNTIL DISCONNECT 指定の排他保持状態

Y：排他を保持しています

N：排他を保持していません

qqq：

デッドロックプライオリティ値（10 進数 3 けた）

HiRDB 内部で取得する排他の占有情報及び待ち情報の場合，-1 を出力する場合があります。

＜出力例＞

プロセス ID が 100000 以上の表示を含む場合

```
HOSTNAME : HOST0001(145717)
SVID       PID     TID  KIND  RESOURCE                          WAIT  MODE
 TRNBID              ACTID              LOCK-STATE  DPRIO
SERVER01   38095   1    0001  03000000000000000000000000000000  *****  SR
 PRDBUNT000010003    16-1-2             N              64  … プロセスIDが99999以下
SERVER01   108221   1    0002  03000200000000000000000000000000  *****  SR
  PRDBUNT000010003    16-1-2             N              64 … プロセスIDが100000以上
SERVER01    8101   1    0007  000300000e450000030002000000  *****  PR
 PRDBUNT000010003    16-1-2             N              64  … プロセスIDが99999以下
```

(b)　-p オプションを指定し，かつ-e オプションを指定しない場合

```
HOSTNAME : aa...a(bbccdd)
SVID       TABLE-KIND   TOTAL    USED    RATE
ee...e       RESOURCE  ff...f  gg...g    hhh%   ...1
                    :
```

[説明]

1 は，サーバの数だけ繰り返し表示されます。

1 で表示する各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e：

サーバ名（8 文字以内）

ff...f：

管理できる排他資源の最大数（32 ビットモードの場合 10 進数 8 けた，64 ビットモードの場合 10 進数 10 けた）

gg...g：

現在管理中の排他資源数（32 ビットモードの場合 10 進数 8 けた，64 ビットモードの場合 10 進数 10 けた）

hhh：

排他資源管理テーブル使用率（%）

(c)　-p オプション及び-e オプションを指定した場合

```
HOSTNAME : aa...a(bbccdd)
SVID : ee...e   ...1
TABLE-NAME       TABLE-KIND        TOTAL       USED RATE-C RATE-T   ...1
ff...f           gg...g           hh...h     ii...i   jjj%   kkk%   ...1, 2
  :                 :                :          :       :      :
SVID : ee...e   ...1
TABLE-NAME       TABLE-KIND        TOTAL       USED RATE-C RATE-T   ...1
ff...f           gg...g           hh...h     ii...i   jjj%   kkk%   ...1, 2
  :                 :                :          :       :      :
```

[説明]

HOSTNAME に表示されるホスト上に存在するユニットごとに表示されます。HiRDB サーバがないユニットは，HOSTNAME だけ表示されます。HiRDB サーバがないユニットとは，システムマネジャだけを配置したユニット又は影響分散スタンバイレス型系切り替え機能を適用しているユニットで，バックエンドサーバが稼働していないユニットが該当します。

1 はサーバごとに，2 は排他制御用プールパーティションごとに表示されます。

2 で表示する各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e：

サーバ名（8 文字以内）

ff...f：

HiRDB が独自に生成した，排他制御用プールパーティション名称（16 文字以内）

gg...g：

テーブル種別（8 文字以内）

RESOURCE：排他資源を管理する排他資源管理テーブル

hh...h：

管理可能な排他資源の最大数（符号なし整数 9 けた以内）

ii...i：

現在管理中の排他資源数（符号なし整数 9 けた以内）

jjj：

現在の使用率（小数点以下は切り捨て）（符号なし整数 3 けた以内）

kkk：

HiRDB 開始時から現在までの使用率の最高値（小数点以下は切り捨て）（符号なし整数 3 けた以内）

(d)　-T オプション及び-R オプションを指定した場合

```
 HOSTNAME : aa...a(bbccdd)
 SVID : nn...n   ...1
 RANGE : eeee/ff/gg hh:ii:jj - kkkk/ll/mm bb:cc:dd...1
 TABLE-NAME       TABLE-KIND   RATE-T            L-CNT            T-CNT W-RATE Q-LEN W-TIME
W-TIME-T...1
 oo...o           pp...p         qqq%          rr...r          ss...s ttt.t%  uuuu vvvv.v
  wwww.w...1,2
    :               :             :               :               :      :      :    :
    :
 SVID : nn...n   ...1
 RANGE : eeee/ff/gg hh:ii:jj - kkkk/ll/mm bb:cc:dd...1
 TABLE-NAME       TABLE-KIND   RATE-T            L-CNT            T-CNT W-RATE Q-LEN W-TIME
W-TIME-T...1
 oo...o           pp...p         qqq%          rr...r          ss...s ttt.t%  uuuu vvvv.v
  wwww.w...1,2
    :               :             :               :               :      :      :    :
    :
```

[説明]

HOSTNAME に表示されるホスト上に存在するユニットごとに表示されます。HiRDB サーバがないユニットは，HOSTNAME だけ表示されます。HiRDB サーバがないユニットとは，システムマネジャだけを配置したユニット又は影響分散スタンバイレス型系切り替え機能を適用しているユニットで，バックエンドサーバが稼働していないユニットが該当します。

1 はサーバごとに，2 は排他制御用プールパーティションごとに表示されます。

2 で表示する各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

eeee/ff/gg hh:ii:jj：

前回 pdls -d lck -T -R コマンドを実行した日付と時刻（年月日時分秒）

HiRDB を開始してから最初にこのコマンドを実行する場合，HiRDB を開始した日付と時刻。

kkkk/ll/mm bb:cc:dd：

pdls -d lck -T -R コマンドを実行した日付と時刻（年月日時分秒）

nn...n：

サーバ名（8 文字以内）

oo...o：

HiRDB が独自に生成した，排他制御用プールパーティション名称（16 文字以内）

pp...p：

テーブル種別（8 文字以内）

RESOURCE：排他資源を管理する排他資源管理テーブル

qqq：

排他資源使用率の最高値（小数点以下は切り捨て）

rr...r：

排他要求回数（符号なし整数 16 けた以内）

ss...s：

排他制御用プールパーティションのスレッド間ロック要求回数（符号なし整数 16 けた以内）

ttt.t：

排他制御用プールパーティションのスレッド間ロック要求回数のうち，スレッド間ロック待ちが発生した回数の割合（小数点 2 位以下は切り捨て）

uuuu：

排他制御用プールパーティションのスレッド間ロック待ち数の平均値

システムの稼働に関する統計情報の排他制御用プールパーティションのスレッド間ロック待ち数を排他制御用プールパーティションごとに集計した値（小数点以下は切り捨て）。

vvvv.v：

スレッド間ロック待ちが発生したときのスレッド間ロック待ち時間の平均値（単位：ミリ秒）※1※2

wwww.w：

スレッド間ロック待ちが発生したときのスレッド間ロック待ち時間の最大値（単位：ミリ秒）※1※2

注※1

スレッド間ロック待ち時間の統計情報を取得しているかどうかで表示が異なります。

統計情報を取得している場合：

- スレッド間ロック待ち時間の平均値が 10 秒以上のときは，vvvv.v は 9999.9 と表示されます。
- スレッド間ロック待ち時間の最大値が 10 秒以上のときは，wwww.w は 9999.9 と表示されます。
- 平均値及び最大値が 10 秒未満のときは，取得した時間が表示されます。

統計情報を取得していない場合：

****.*が表示されます。

注※2

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

注意事項

-w オプションを指定して pdstbegin コマンドを実行した後，最初に-T オプション及び-R オプションを指定する場合，次に示す値は実際の値より小さな値が出力されます。

- スレッド間ロック待ちが発生したときのスレッド間ロック待ち時間の平均値
- スレッド間ロック待ちが発生したときのスレッド間ロック待ち時間の最大値

そのため，スレッド間ロック待ち時間に関する統計情報を取得する場合は，-w オプションを指定して pdstbegin コマンドを実行した後に，一度-T オプション及び-R オプションを指定して pdls -d lck コマンドを実行してください。

(e) -T オプションを指定し，かつ-R オプションを指定しない場合

```
HOSTNAME : aa...a(bbccdd)
SVID : nn...n   ...1
TABLE-NAME       TABLE-KIND   RATE-T           L-CNT           T-CNT W-RATE Q-LEN...1
oo...o           pp...p         qqq%         rr...r         ss...s ttt.t%  uuuu...1, 2
   :                :            :              :              :     :       :
SVID : nn...n   ...1
TABLE-NAME       TABLE-KIND   RATE-T           L-CNT           T-CNT W-RATE Q-LEN...1
oo...o           pp...p         qqq%         rr...r         ss...s ttt.t%  uuuu...1, 2
   :                :            :              :              :     :       :
```

[説明]

HOSTNAME に表示されるホスト上に存在するユニットごとに表示されます。HiRDB サーバがないユニットは，HOSTNAME だけ表示されます。HiRDB サーバがないユニットとは，システムマネジャだけを配置したユニット又は影響分散スタンバイレス型系切り替え機能を適用しているユニットで，バックエンドサーバが稼働していないユニットが該当します。

1 はサーバごとに，2 は排他制御用プールパーティションごとに表示されます。

2 で表示する各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

nn...n：

サーバ名（8 文字以内）

oo...o：

HiRDB が独自に生成した，排他制御用プールパーティション名称（16 文字以内）

pp...p：

テーブル種別（8 文字以内）

RESOURCE：排他資源を管理する排他資源管理テーブル

qqq：

排他資源使用率の最高値（小数点以下は切り捨て）

rr...r：

排他要求回数（符号なし整数 16 けた以内）

ss...s：

排他制御用プールパーティションのスレッド間ロック要求回数（符号なし整数 16 けた以内）

ttt.t：

排他制御用プールパーティションのスレッド間ロック要求回数のうち，スレッド間ロック待ちが発生した回数の割合（小数点 2 位以下は切り捨て）

uuuu：

排他制御用プールパーティションのスレッド間ロック待ち数の平均値

システムの稼働に関する統計情報の排他制御用プールパーティションのスレッド間ロック待ち数を排他制御用プールパーティションごとに集計した値（小数点以下は切り捨て）。

2.64　pdls【-d mem】（サーバの共用メモリの状態表示）

2.64.1　pdls【-d mem】の形式と規則

(1)　機能

サーバの共用メモリの情報を表示します。ホストに対応するユニットで使用中の共用メモリサイズを，ユニットごとに表示します。

(2)　形式

```
pdls -d mem 〔 {-x ホスト名〔, ホスト名〕… | -u ユニット識別子〔, ユニット識別子〕…} 〕
```

(3)　オプション

(a)　-x ホスト名〔, ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(b)　-u ユニット識別子〔, ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4)　注意事項

1. pdls -d mem コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(5)　出力形式

```
HOSTNAME : aa...a(bbccdd)
SHM-ID  GET-SIZE  ACT-SIZE SHM-OWNER          POOL-ID
ee...e    ff...f    gg...g    hh...h           ii...i
  :         :         :         :                :
```

[説明]

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e：

共用メモリセグメントの識別子（10 進数 10 けた以内）

6 けたで表示できる値を超えたときは，以降の表示はその分だけ右側にずれます。

ff...f：

確保サイズ（10 進数 10 けた以内）（単位：バイト）

9 けたで表示できる値を超えたときは，以降の表示はその分だけ右側にずれます。

gg...g：

使用サイズ（10 進数 10 けた以内）（単位：バイト）

9 けたで表示できる値を超えたときは，以降の表示はその分だけ右側にずれます。

hh...h：

共用メモリを使用するプロセスの属性（8 文字以内）

共用メモリの種類		hh...h の内容
ユニットコントローラ用共用メモリ		MANAGER
グローバルバッファ用共用メモリ	影響分散スタンバイレス型系切り替え機能を適用するユニットの場合	ユニット名を（）で囲んだ文字列
	影響分散スタンバイレス型系切り替え機能を適用しないユニットの場合	この共用メモリを使用するサーバ名
インメモリデータバッファ用共用メモリ		この共用メモリを使用するサーバ名
ユティリティ用共用メモリ		UTILITY
セキュリティ監査情報用バッファ用共用メモリ		AUDDEF

ii...i：

プール識別子（16 文字以内）

共用メモリの種類		ii...i の内容
ユニットコントローラ用共用メモリ		表示されません。
グローバルバッファ用共用メモリ	影響分散スタンバイレス型系切り替え機能を適用するユニットの場合	ユニット内での通番
	影響分散スタンバイレス型系切り替え機能を適用しないユニットの場合	この共用メモリを使用するサーバ内での通番
インメモリデータバッファ用共用メモリ		この共用メモリを使用するサーバ内での通番※
ユティリティ用共用メモリ		一意な文字列
セキュリティ監査情報用バッファ用共用メモリ		なし

注

ff...f（確保サイズ）と gg...g（使用サイズ）は，常に一致した値（確保サイズの値）が表示されます。

注※

通番は，次に示す表で説明する通番の最小値から昇順に割り当てられます。

条件			通番の最小値
グローバルバッファの動的変更機能の使用有無	pd_max_add_dbbuff_no の指定	pd_max_add_dbbuff_shm_no の指定	
使用している pd_dbbuff_modify = Y	指定あり	指定あり	a ＋ 513
		指定なし	a ＋ 1
	指定なし	指定あり	a ＋ 513
		指定なし	1513
使用していない pd_dbbuff_modify = N	−	−	513

（凡例）

−：該当しない

a：pd_max_add_dbbuff_shm_no の値

pd_max_add_dbbuff_shm_no オペランドの省略値については，マニュアル「HiRDB システム定義」で確認してください。

2.65　pdls【-d prc】（サーバのプロセスの状態表示）

2.65.1　pdls【-d prc】の形式と規則

(1)　機能

サーバのプロセスの状態を表示します。

(2)　形式

```
pdls -d prc 〔 {-a | -s サーバ名} 〕 〔-c〕 〔-C 〔-H〕〕
            〔 {-x ホスト名〔, ホスト名〕… | -u ユニット識別子〔, ユニット識別子〕…} 〕
```

(3)　オプション

(a)　{-a ｜ -s サーバ名}

プロセスの状態を表示する場合に，HiRDB システムのすべてのサーバ（HiRDB が使用する内部的なサーバも含まれます）の情報を表示するか，指定したサーバの情報を表示するかを指定します。省略した場合，すべてのサーバ（HiRDB が使用する内部的なサーバは含まれません）の情報が表示されます。

-a

HiRDB システムのすべてのサーバ（HiRDB が使用する内部的なサーバも含まれます）の，プロセスの状態を表示します。

-s サーバ名　～＜識別子＞((1～8))

プロセスの状態を表示するサーバの名称を指定します。

(b)　-c

プロセスの動作状態（プロセスの停止予定の有無）を表示する場合に指定します。

このオプションは，pdpfresh コマンドでプロセスのリフレッシュを開始した後，プロセスの停止状況を確認するために使用します。

(c)　-C

サーバプロセスの状態を，DAT 形式で出力する場合に指定します。

(d)　-H

-C オプションを指定した場合に，タイトル行を付けて出力するときに指定します。

(e)　-x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-x オプションの指定は無視されます。また，複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(f)　-u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-u オプションの指定は無視されます。また，複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4)　注意事項

1. pdls -d prc コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(5)　出力形式

(a)　-C オプションを省略した場合

```
HOSTNAME : aa...a(bbccdd)
STATUS PID    UID    GID    SVID   TIME   PROGRAM C-PID          C-GRP
e      ff...f gg...g hh...h ii...i jjkkll mm...m  nn...n(oo...o) pp
:         :      :      :      :       :       :
```

［説明］

STATUS「サーバの稼働状態」から C-GRP「サーバのプロセスと接続しているクライアントのクライアントグループ名称」までの各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

e：

サーバの稼働状態

D：サーバの開始処理中又は終了処理中

L：サーバ稼働中

C：プロセス停止処理中，又は停止予定あり（-c オプション指定時に表示）

ff...f：

プロセス ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，UID（ユーザ ID）以降の項目の表示位置は右にシフトします。

gg...g：

ユーザ ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，GID（グループ ID）以降の項目の表示位置は右にシフトします。

hh...h：

グループ ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，SVID（サーバ名）以降の項目の表示位置は右にシフトします。

ii...i：

サーバ名（8 文字以内）

jjkkll：

サービス要求受け付け時刻（時分秒）

UAP と接続していないサーバの場合は 999999 を表示

mm...m：

UAP の識別情報（30 文字以内）

プロセスがユーザサーバの場合は，次に示す値が表示されます。接続していないプロセスの場合は何も表示されません（DAT 形式出力時は空要素が表示されます）。また，DAT 形式の場合，PDCLTAPNAME の末尾に半角空白を指定していると，末尾の半角空白は削除された状態で表示されます。

- クライアント環境定義 PDCLTAPNAME の指定がある

  PDCLTAPNAME に指定した値を表示
- クライアント環境定義 PDCLTAPNAME の指定がない

  「Unknown」を表示

また，次のどちらかのプロセスでは，「********」と表示します。

- 起動・停止途中のプロセス
- システムサーバプロセス

nn...n：

サーバのプロセスと接続していたクライアント（UAP 又はユティリティ）のプロセス ID（10 進数 10 けた以内で左詰め表示）。

クライアントと接続していないサーバプロセスの場合，プロセス ID は表示されません（DAT 形式出力時は空要素が表示されます）。

なお，次に示す場合，プロセス ID に 0 が表示されます。

- UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合
- Type4 JDBC ドライバから接続している場合

oo...o：

サーバのプロセスと接続していたクライアント（UAP 又はユティリティ）の IP アドレス（15 文字以内）。

クライアントと接続していないサーバプロセスの場合，IP アドレスは両端の()を含めて表示されません（DAT 形式出力時は空要素が表示されます）。

pp：

サーバのプロセスと接続しているクライアントのクライアントグループ名称（2 文字以内）。

クライアントと接続していないサーバプロセスの場合，クライアントグループ名称は表示されません（DAT 形式出力時は空要素が表示されます）。

1 文字の場合はユーザが任意に設定したクライアントグループ名称を，2 文字の場合は HiRDB のクライアント種別ごとのクライアントグループ名称を示します。HiRDB のクライアント種別ごとのクライアントグループ名称を次に示します。

クライアントグループ名称	クライアント種別
XA	X/Open XA インタフェース
PC	PC クライアント
WS	WS クライアント

注

サーバプロセス起動直後には，UAP の識別情報に********が表示されたり，サーバ名に情報が表示されない（DAT 形式出力時には空要素を表示する）場合があります。HiRDB のコマンドプロセスの状態を表示する場合，ユーザ ID 及びグループ ID には 0 が表示されます。

＜出力例＞

プロセス ID が 100000 以上の表示を含む場合

```
HOSTNAME : HOST0001(151654)
STATUS  PID   UID   GID    SVID       TIME  PROGRAM C-PID          C-GRP
  L    38095  1108   700   SERVER01 151643  uap0001 7040(10.1.0.1) WS
… プロセスIDが99999以下
  L    108221  1108   700   SERVER01 999999
… プロセスIDが100000以上
  L     8101  1108   700   SERVER01 999999
… プロセスIDが99999以下
```

(b)　-C -H オプションを指定した場合

```
"EDIT-TIME","HOSTNAME","STATUS","PID","UID","GID","SVID","TIME","PROGRAM","C-PID","C-ADDR","
C-GRP"
"qq...q","aa...a","e",ff...f,gg...g,hh...h,"ii...i","jjkkll","mm...m",nn...n,"oo...o","pp"
```

[説明]

aa...a～pp については，-C オプションを省略した場合の説明を参照してください。

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

1 行には，1 サーバのプロセス情報を表示します。

qq...q：

pdls コマンドを実行した時刻を YYYY/MM/DD hh:mm:ss の形式で表示します。

YYYY：年（西暦）　MM：月　DD：日　hh：時　mm：分　ss：秒

2.66　pdls【-d rpc】（サーバの通信制御情報の表示）

2.66.1　pdls【-d rpc】の形式と規則

(1)　機能

コネクションが確立しているサーバの通信制御情報を表示します。このコマンドは，UAP の動作中に無応答状態になった場合に，処理の制御が HiRDB 側にあるのか，クライアント側にあるのかを知りたい場合に有効です。

(2)　形式

```
pdls -d rpc 〔-a〕
```

(3)　オプション

(a)　-a

HiRDB システム内のすべてのプロセスの通信制御情報を表示します。より詳細な通信制御情報を知りたい場合に有効です。

(4)　注意事項

1. pdls -d rpc コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. -a オプションを指定した場合，サーバ種別が F 及び S でクライアントが接続しているときに，次の情報が表示されます。
   - クライアント OS 種別
   - クライアントグループ名称
   - XA インタフェース使用有無
3. -a オプションを指定した場合，クライアント種別が C でクライアントが接続しているときに，次の情報が表示されます。
   - クライアントのプロセス ID
   - クライアントの IP アドレス
   - クライアントの UAP 識別情報

(5) 出力形式

(a) オプションを省略した場合の表示（UAP が接続されている場合）

```
[aaaa]
  Client Information :
    PID      ADDR      NAME     OS   GROUP   XA
    bb...b   cc...c    dd...d   e    ff      g
  Server Information :
    PID     SVID    STATUS  SYS_EVENT USR_EVENT  CRITICAL(MASK)  SERVICE  SYS_CALL
    hh...h  ii...i  jj...j  0xkk...k  0xll...l   m(0xnn...n)     oo...o   0xtt...t
       :       :       :         :         :           :            :         :
[aaaa]
  Client Information :
  :
```

(b) オプションを省略した場合の表示（ユティリティが接続されている場合）

```
[aaaa]
  Server Information :
    PID     SVID     STATUS   SYS_EVENT USR_EVENT    CLITICAL(MASK) SERVICE  SYS_CALL
    hh...h  ii...i   jj...j   0xkk...k  0xll...l     m(0xnn...n)    oo...o   0xtt...t
       :       :        :         :         :             :            :         :
[aaaa]
  Server Information :
  :
```

(c) -a オプションを指定した場合の表示

```
HOSTNAME : ss...s
PID     SVID    STATUS   SYS_EVENT USR_EVENT CLITICAL(MASK) SERVICE OS GROUP XA  ..1
hh...h  ii...i  jj...j   0xkk...k  0xll...l  m(0xnn...n)    oo...o  e  ff    g  ...2
CLTKIND SVRKIND CLTPID CLTADDR CLTNAME ACTID    SYS_CALL  ........................3
p       q       bb...b cc...c  dd...d  0xrr...r 0xtt...t  ........................4
:        :        :       :       :         :       :
```

[説明]

1～4 は，各々 1 行で表示されます。2 及び 4 は，プロセスの数だけ繰り返し表示します。

aaaa：

項番

bb...b：

サーバのプロセスと接続しているクライアントのプロセス ID（10 進数 10 けた以内）

なお，次に示す場合，プロセス ID は表示されません。

- クライアントと接続していないサーバプロセスの場合
- UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合
- Type4 JDBC ドライバから接続している場合

cc...c：

サーバのプロセスと接続しているクライアントの IP アドレス（15 文字以内）

なお，UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合は，IP アドレスは表示されません。

dd...d：

サーバのプロセスと接続しているクライアントの UAP 識別情報（30 文字以内）

クライアント環境変数の PDCLTAPNAME に指定した UAP の識別情報を表示します。PDCLTAPNAME が指定されていない場合，"Unknown"を表示します。

なお，UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合は，UAP 識別情報は表示されません。

e：

クライアント OS 種別

W：WS 系クライアント

P：PC 系クライアント

なお，UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合は，クライアント OS 種別は表示されません。

ff：

サーバのプロセスと接続しているクライアントのクライアントグループ名称（2 文字以内）

1 文字の場合はユーザが任意に設定したクライアントグループ名称を，2 文字の場合は HiRDB のクライアント種別ごとのクライアントグループ名称を示します。

なお，次の場合はクライアントグループ名称が表示されません。

- UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合
- クライアントグループ名称の接続枠保証機能を使用した場合に，自由利用枠で接続したとき
- クライアントグループ名称の接続枠保証機能を使用していない場合

HiRDB のクライアント種別ごとのクライアントグループ名称を次に示します。

クライアントグループ名称	クライアント種別
XA	X/Open XA インタフェース
PC	PC クライアント
WS	WS クライアント

g：

X/Open XA インタフェース使用有無

Y：使用しています

N：使用していません

なお，UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合は，X/Open XA インタフェース使用有無は表示されません。

hh...h：

サーバのプロセス ID（10 進数 10 けた以内）

ii...i：

サーバ名（8 文字以内）

jj...j：

プロセス状態表示（12 文字以内）

ACTIVE：処理動作中

SUSPEND（QUE）：処理中断中（サービス要求待ち状態：キュー待ち）

SUSPEND（CLT）：処理中断中（次サービス要求待ち状態）

SUSPEND（SVR）：処理中断中（そのほかの受信待ち状態）

kk...k：

プロセス状態詳細表示（26 文字）

システム内部コードを表示します。

ll...l：

プロセス状態詳細表示（8 文字）

システム内部コードを表示します。

m：

クリティカル状態表示

Y：クリティカル中です。

N：クリティカル中ではありません。

nn...n：

クリティカル情報（16 進数 8 けた）

oo...o：

サービス名称表示（32 文字以内）

システム内部情報を表示します。

p：

クライアント種別

C：クライアント

U：ユティリティ

空白：上記以外

q：

サーバ種別（ただし，-a オプション指定時は，クライアント種別が C のときだけ表示します）

F：フロントエンドサーバ（FES）

B：バックエンドサーバ（BES）

D：ディクショナリサーバ（DIC）

S：シングルサーバ（SDS）

rr...r：

アクティビティ ID（18 文字）

ただし，アクティビティ ID を保持していないプロセスの場合は表示しません。

ss...s：

ホスト名（32 文字以内）

tt...t：

システムが使用する内部情報（8 文字）

(6) 情報確認例

SQL が応答されない場合

STATUS が ACTIVE となっていれば，そのサーバは処理中です。

現在の制御がクライアントにあるのかサーバにあるのかを調べる場合

フロントエンドサーバ(シングルサーバ)の STATUS に表示されている情報で判断できます。フロントエンドサーバ(シングルサーバ)の STATUS が SUSPEND(CLT)であれば，制御がクライアントにあります。

2.67　pdls【-d rpl】（HiRDB Datareplicator 連携の状態表示）

2.67.1　pdls【-d rpl】の形式と規則

(1)　機能

HiRDB Datareplicator 連携の状態を表示します。また，システムログファイルでの抽出側 HiRDB Datareplicator のシステムログの抽出状況を表示します。

(2)　形式

```
pdls -d rpl 〔-j〕 〔-u ユニット識別子〕 〔-s サーバ名〕
```

(3)　オプション

(a)　-j

システムログファイルでの抽出側 HiRDB Datareplicator のシステムログの抽出状況を表示する場合に指定します。このオプションを指定した場合，次の情報を表示します。

- HiRDB Datareplicator 連携機能を使用しているかどうか
- 抽出側 HiRDB Datareplicator のシステムログファイルからのシステムログの抽出状況
- システムが使用する内部情報

省略した場合，HiRDB Datareplicator 連携機能を使用しているかどうかだけ表示されます。システムログの抽出状況及び内部情報は表示されません。

(b)　-u ユニット識別子　～＜識別子＞((4 文字))

特定のユニットの，抽出側 HiRDB Datareplicator のシステムログの抽出状況を表示する場合に，そのユニット識別子を指定します。

省略した場合，システム内のすべてのユニットが表示対象となります。-j オプションを省略して，このオプションを指定した場合，システム及び指定したユニットが HiRDB Datareplicator 連携機能を使用しているかどうかだけが表示されます。

(c)　-s サーバ名　～＜識別子＞((1～8))

特定のサーバの，抽出側 HiRDB Datareplicator のシステムログの抽出状況を表示する場合に，そのサーバ名を指定します。

省略した場合，システム内のすべてのサーバが表示対象となります。-j オプションを省略して，このオプションを指定した場合，システム及び指定したサーバがあるユニットが HiRDB Datareplicator 連携機能を使用しているかどうかだけが表示されます。

(4) 注意事項

1. pdls -d rpl コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. -d rpl を指定した pdls コマンドで，-j オプションを指定して，-u 及び-s オプションを省略した場合，システム内のすべてのユニットの，すべてのサーバについて HiRDB Datareplicator 連携の状態を表示します。
3. -d rpl を指定した pdls コマンドは，サーバが停止している状態でも実行できます。しかし，システム定義の pd_rpl_hdepath に抽出側 HiRDB Datareplicator 運用ディレクトリ名が指定されていない場合，又はデータ連動用連絡ファイルがない場合には，エラーとなります。

(5) 出力形式

(a) -j オプションを省略した場合

```
SYSTEMID         :aaaa(bbccdd)   ※1
Data Replication :e              ※1
UNITID           :ffff(bbccdd)   ※2
Data Replication :e              ※2
```

注※1

HiRDB/パラレルサーバの場合で，かつ-u オプションを指定した場合，指定したユニットがシステムマネジャのサーバマシンのときだけ出力されます。

注※2

HiRDB/パラレルサーバの場合，システム内のユニットの数だけ出力されます。また，-u オプションを指定した場合，指定したユニットについてだけ出力されます。

［説明］

aaaa：

HiRDB 識別子（4 文字）

bbccdd：

pdls コマンドを入力した時刻（時分秒）

e：

HiRDB Datareplicator 連携に関する情報（1 文字）

この情報は，HiRDB システム及びユニットとして HiRDB Datareplicator 連携機能を使用しているかどうかを表示します。そのため，システム内及びユニット内のバックエンドサーバすべてに連動の対象と

なるデータベースがない場合でも，HiRDB Datareplicator 連携機能を使用していれば，「Y（HiRDB Datareplicator 連携機能を使用しています）」が表示されます。

Y：

HiRDB Datareplicator 連携機能を使用しています。

N：

HiRDB Datareplicator 連携機能を使用していません。

＊：

システムマネジャが停止中のため，システムの連携状態を表示できません。

それぞれのユニットの状態を確認して，HiRDB Datareplicator 連携機能を使用していないユニットが一つでもあれば，システムとしては HiRDB Datareplicator 連携機能を使用していないことになります。

-：

このユニットには，HiRDB Datareplicator 連携の対象となるサーバがありません。連携の対象となるサーバは，HiRDB/パラレルサーバの場合はバックエンドサーバ，HiRDB/シングルサーバの場合はシングルサーバです。

ffff：

ユニット識別子（4 文字）

(b) -j オプションを指定した場合

```
SYSTEMID         :aaaa(bbccdd)          ※1
Data Replication :e                     ※1
UNITID           :ffff(bbccdd)          ※2
Data Replication :e                     ※2
SERVER NAME      :gg...g                ※2※3
Extract Database :h                     ※2※3
Extract Status   :I                     ※2※3
System Log Extract Point  :             ※2※3※4
Run ID   Group    Gen No.   Block No.   ※2※3※4
jj...j   kk...k   ll...l    mm...m      ※2※3※4
System Log Sync info      :             ※2※3※4
Run ID   Group    Gen No.   Block No.   ※2※3※4
nn...n   oo...o   pp...p    qq...q      ※2※3※4
```

注※1

HiRDB/パラレルサーバの場合で，かつ-u オプションを指定した場合，指定したユニットがシステムマネジャのサーバマシンのときだけ出力されます。

注※2

HiRDB/パラレルサーバの場合は，システム内のユニットの数だけ出力されます。

注※3

HiRDB/パラレルサーバの場合は，ユニット内のサーバの数だけ出力されます。-s オプションを指定した場合は，指定したサーバについてだけ出力されます。また，システム定義の pd_rpl_hdepath オペランドを指定していない場合，又は連動連絡ファイルが不正な状態（ファイルがないなど）の場合，出力されません。

注※4

指定した HiRDB サーバに抽出の対象となるデータベースがある場合だけ表示されます。また，-u オプションを指定した場合，指定したユニットについてだけ出力されます。

［説明］

aaaa：

HiRDB 識別子（4 文字）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

e：

HiRDB Datareplicator 連携に関する情報（1 文字）

この情報は，HiRDB システム及びユニットとして HiRDB Datareplicator 連携機能を使用しているかどうかを表示します。そのため，システム内及びユニット内のバックエンドサーバすべてに連動の対象となるデータベースがない場合でも，HiRDB Datareplicator 連携機能を使用していれば，「Y（HiRDB Datareplicator 連携機能を使用しています）」が表示されます。

Y：

HiRDB Datareplicator 連携機能を使用しています。

N：

HiRDB Datareplicator 連携機能を使用していません。

＊：

システムマネジャが停止中のため，システムの連携状態を表示できません。

それぞれのユニットの状態を確認して，HiRDB Datareplicator 連携機能を使用していないユニットが一つでもあれば，システムとしては HiRDB Datareplicator 連携機能を使用していないことになります。

-：

このユニットには，HiRDB Datareplicator 連携の対象となるサーバがありません。連携の対象となるサーバは，HiRDB/パラレルサーバの場合はバックエンドサーバ，HiRDB/シングルサーバの場合はシングルサーバです。

ffff：

ユニット識別子（4 文字）

gg...g：

サーバ名（8 文字以内）

h：

抽出の対象となるデータベースの有無

Y：

g で表示された HiRDB サーバに抽出の対象となるデータベースがあります。

N：

g で表示された HiRDB サーバに抽出の対象となるデータベースがありません。

i：

システムが使用する内部情報（1 文字）

HiRDB Datareplicator 連携機能を使用している場合は，次の情報が表示されます。

・抽出側 HiRDB Datareplicator のシステムログの抽出状況

jj...j：

ラン ID（ファイルが使用されたときのログサーバプロセスのラン ID）（16 進数 8 けた）

kk...k：

現在抽出中のシステムログファイルのグループ名（8 文字以内）

ll...l：

現在抽出中のシステムログファイルの世代番号（16 進数 8 けた）

mm...m：

現在抽出中のシステムログブロック番号（16 進数 8 けた）

・抽出側 HiRDB のシステムログの内部情報

nn...n：

システムで使用する内部情報（8 文字）

oo...o：

システムで使用する内部情報（8 文字以内）

pp...p：

システムで使用する内部情報（8 文字以内）

qq...q：

システムで使用する内部情報（8 文字以内）

注

j～q の項目は，データ連動用連絡ファイルを初期化した後に，HiRDB Datareplicator 連携を開始していない場合，及び HiRDB Datareplicator 連携を実行中にデータ連動用連絡ファイルを初期化した場合には，空白，又は「0」が表示されます。

2.68　pdls【-d scd】（サーバのスケジュールの状態表示）

2.68.1　pdls【-d scd】の形式と規則

(1)　機能

サーバのスケジュールの状態を表示します。

(2)　形式

```
pdls -d scd 〔 {-a | -s サーバ名} 〕
            〔 {-x ホスト名〔，ホスト名〕… | -u ユニット識別子〔，ユニット識別子〕…} 〕
```

(3)　オプション

(a)　-a

pdls コマンドを実行した時点で，起動中のすべてのサーバの，サービスのスケジュール状態を表示します。

(b)　-s サーバ名　～＜識別子＞((1～8))

スケジュールの状態を表示するサーバの名称を指定します。

(c)　-x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-x オプションの指定は無視されます。また，複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(d)　-u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-u オプションの指定は無視されます。また，複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4)　注意事項

1. pdls -d scd コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(5) 出力形式

```
HOSTNAME : aa...a(bbccdd)
total server count ee...e   ..................................................1
SVID   ST QUECNT MAXCNT USED_POL FREE_POL MAX_UPOL MAX_FPOL LAST_RECV_TIME
ff...f g  hh...h ii...i jj...j   kk...k   ll...l   mm...m   nnooppqqrrss   .....2
   :   :     :       :       :         :        :        :
```

[説明]

1 は，-s オプション指定の場合だけ表示されます。また，-a オプション指定の場合で，スケジューラの下で動作するサーバが複数あるときは，サーバの数だけ 2 が繰り返し表示されます。スケジューラとは，要求されたサービスのスケジューリングとプロセスの制御をするものです。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e：

スケジューラの下で動作するサーバの数（10 進数 6 けた）

ff...f：

サーバ名（8 文字以内）

g：

該当するサーバの稼働状態

S：サーバ準備中

A：スケジューリングできる状態

E：サーバ終了処理中

hh...h：

現在のサービス要求キューイング数（10 進数 5 けた）

ii...i：

サービス要求の最大キューイング数（10 進数 5 けた）

jj...j：

転送データ用バッファの現在の使用サイズ（10 進数 8 けた）

kk...k：

転送データ用バッファの現在の未使用サイズ（10 進数 8 けた）

ll...l：

転送データ用バッファの最大使用サイズ（10 進数 8 けた）

mm...m：

転送データ用バッファの現在の最大連続未使用サイズ（10 進数 8 けた）

nnooppqqrrss：

最後にメッセージキューからメッセージを取り出した日時（年月日時分秒）

1 度もメッセージを取り出していない場合は 000000000000 が表示されます。

注

g（該当するサーバの稼働状態）が E の場合，hh...h から mm...m には 0 が表示されることがあります。サーバ以外にユティリティの状態も表示されます。

2.69 pdls【-d stj】（ユニット及びサーバごとの統計情報出力種別の指定有無の表示）

2.69.1 pdls【-d stj】の形式と規則

(1) 機能

次に示す情報を表示します。

- ユニット及びサーバごとの pdstbegin コマンド及びシステム共通定義 pdstbegin による統計情報出力種別の指定有無
- システム共通定義 pd_statistics によるシステムの稼働に関する統計情報出力の指定有無

(2) 形式

```
pdls -d stj 〔-s サーバ名〕
            〔 {-x ホスト名〔, ホスト名〕… | -u ユニット識別子〔, ユニット識別子〕…} 〕
```

(3) オプション

(a) -s サーバ名 ～＜識別子＞((1～8))

統計情報出力種別の指定有無を表示するサーバ名を指定します。省略した場合，すべてのサーバが表示対象になります。

(b) -x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-x オプションの指定は無視されます。また，複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(c) -u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-u オプションの指定は無視されます。また，複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4) 注意事項

1. pdls -d stj コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

2. 統計情報出力中にファイル障害が発生した場合，pdls -d stj コマンドによる統計情報出力指定有無の表示は，統計情報出力が指定されていない表示になります。
3. HiRDB/シングルサーバの場合，SQL オブジェクト転送に関する統計情報は表示されません。
4. スレッド間ロック待ち時間を出力する統計情報種別については，「pdstbegin（統計情報の出力開始）(4)オプション」を参照してください。

(5) 出力形式

```
HOSTNAME : aa...a(bbccdd)
UNITID      sys
eeee         h
INTERVAL : ff...f     DIO-INTERVAL : uu...u
 SVID       sys uap sql buf fil dfw idx sop dop pcd sqh obj dio
 gg...g      i   j   k   l   m   n   o   p   q   r   s   t   v
   :                             :
```

[説明]

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls -d stj コマンド実行時刻（時分秒）

eeee：

ユニット識別子（4 文字）

ff...f：

システムの稼働に関する統計情報出力の時間間隔を分単位で表示します。システムの稼働に関する統計情報出力が指定されていない場合には 0 を表示します（10 進数 4 けた以内）。

gg...g：

サーバ名（8 文字以内）

ユニット内にサーバがない場合には，"********"を表示します。

次に示す h～t，及び v で，該当する統計情報の出力が指定されている場合には"*"を表示します。指定されていない場合には空白を表示します。ただし，1:1 スタンバイレス型系切り替え機能を使用している代替 BES ユニットの場合は，h に"-"を表示します。スレッド間ロック待ち時間を取得する場合は，i に"w"を表示します。UAP に関する統計情報やグローバルバッファプールに関する統計情報のスレッド間ロック待ち時間も取得するときは，i に"w"を表示すると共に，j 及び l に"*"を表示します。

データベースの入出力に関する統計情報を取得する場合は，システム共通定義の pdstbegin オペランド又は pdstbegin コマンドの-D オプション指定値によって，v に表示する内容が異なります。v に表示する内容を次に示します。

everyio	everyfile	表示内容
×	×	*

everyio	everyfile	表示内容
×	○	F
○	×	I
○	○	A

（凡例）

○：指定あり

×：指定なし

h：

ユニット単位のシステムの稼働に関する統計情報

i：

サーバ単位のシステムの稼働に関する統計情報

j：

UAP に関する統計情報

k：

SQL に関する統計情報

l：

グローバルバッファプールに関する統計情報

m：

データベース操作に関する HiRDB ファイルの統計情報

n：

デファードライト処理に関する統計情報

o：

インデクスに関する統計情報

p：

SQL 静的最適化に関する統計情報

q：

SQL 動的最適化に関する統計情報

r：

SQL オブジェクト実行に関する統計情報

s：

SQL 文の履歴に関する統計情報

t：

SQL オブジェクト転送に関する統計情報

uu...u：

データベースの入出力に関する統計情報出力の時間間隔を秒単位で表示します。データベースの入出力に関する統計情報出力が指定されていない場合には 0 を表示します（10 進数 4 けた以内）。

v：

データベースの入出力に関する統計情報

2.70　pdls【-d sts】（ステータスファイルの状態表示）

2.70.1　pdls【-d sts】の形式と規則

(1)　機能

ステータスファイルの状態を表示します。

(2)　形式

(a)　HiRDB/シングルサーバの場合

ユニット用ステータスファイル

```
pdls -d sts 〔 {-x ホスト名 | -u ユニット識別子} 〕
            〔 {-n 論理ファイル名 | -f 物理ファイル名 | -a | -l | -p} 〕
```

サーバ用ステータスファイル

```
pdls -d sts -s サーバ名
            〔 {-n 論理ファイル名 | -f 物理ファイル名 | -a | -l | -p} 〕
```

(b)　HiRDB/パラレルサーバの場合

ユニット用ステータスファイル

```
pdls -d sts {-x ホスト名 | -u ユニット識別子}
            〔 {-n 論理ファイル名 | -f 物理ファイル名 | -a | -l | -p} 〕
```

サーバ用ステータスファイル

```
pdls -d sts -s サーバ名
            〔 {-n 論理ファイル名 | -f 物理ファイル名 | -a | -l | -p} 〕
```

(3)　オプション

(a)　{-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルの状態を表示する場合，ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。HiRDB/シングルサーバの場合，自ホスト又は自ユニットに固定されているので省略できます。

-x ホスト名　～＜識別子＞((1～32))

ユニット用ステータスファイルに対応するホストのホスト名を指定します。

-u ユニット識別子　～＜識別子＞((4 文字))

ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

(b)　-s サーバ名　～＜識別子＞((1～8))

サーバ用ステータスファイルの状態を表示する場合，サーバ用ステータスファイルに対応するサーバの名称を指定します。

(c)　{-n 論理ファイル名｜-f 物理ファイル名｜-a ｜-l ｜-p}

ステータスファイルの状態をどのように表示するかを指定します。省略した場合，ユニット又はサーバが使用中のすべてのステータスファイルの状態が表示されます。

-n 論理ファイル名　～＜識別子＞((1～8))

ステータスファイルの状態を論理ファイルごとに表示したい場合，状態を表示する論理ファイルの名称を指定します。

-f 物理ファイル名　～＜パス名＞((167 文字以内))

ステータスファイルの状態を物理ファイルごとに表示したい場合，状態を表示する物理ファイルの名称を，絶対パス名で指定します。物理ファイル名は，HiRDB ファイルシステム領域名¥HiRDB ファイル名で指定してください。

-a

ユニット又はサーバが使用中の，すべてのステータスファイルの状態を短縮形式で表示します。

-l

ユニット又はサーバが使用中の，すべての論理ファイルごとの状態を表示します。

-p

ユニット又はサーバが使用中の，すべての物理ファイルごとの状態を表示します。

(4)　注意事項

1. pdls -d sts コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. HiRDB が動作中のときに，論理ファイルの状態が CLOSE，NONE，又は BLOCKADE のステータスファイルに対して，pdstsinit コマンド又は pdstsrm コマンドを実行しても，pdls コマンドで表示されるステータスファイルの状態は変わりません。この場合，pdls コマンドを実行する前に pdstsopen コマンドを実行すると，現在のステータスファイルの状態が表示されます。

(5)　出力形式

```
HOSTNAME : aa...a(bbccdd)
 Logical file L_Status Factor Available Manage   ...........2
 ee...e       ff...f   ggg    hh...h    ii...i   ...........2
```

```
 Sy  P_status R_length R_count P_name   .....................1,2
  j  kk...k   ll...l   mm...m  nn...n   .....................1,2
  j  kk...k   ll...l   mm...m  nn...n   .....................1,2
```

[説明]

1 は物理ファイルの状態表示部分，2 は論理ファイルの状態表示部分です。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

ee...e：

論理ファイル名（8 文字以内）

ff...f：

論理ファイルの状態（8 文字以内）

ACTIVE：現用ファイル

BLOCKADE：障害閉塞ファイル

CLOSE：予約ファイル

NONE：論理ファイルに対応する物理ファイルがない状態

STANDBY：予備ファイル

ggg：

ファイル内のレコード使用率（%）

hh...h：

ファイル内の連続空きレコード数（10 進数 8 けた）

ii...i：

ファイル内の管理レコード数（10 進数 8 けた）

j：

物理ファイルが A 系か B 系かの表示

A：A 系

B：B 系

kk...k：

物理ファイルの状態を最大 11 通りの組み合わせで表示します。11 文字以内の場合は右詰めで表示します（11 文字以内）。

a：現用ファイル

b：障害閉塞ファイル

c：クローズファイル

i：初期設定状態

l：論理エラー

n：論理ファイルに対応する物理ファイルがない状態

o：オープンファイル

p：物理エラー

r：障害によるファイル回復状態

s：予備ファイル

u：使用済みファイル

ll...l：

レコード長（10 進数 8 けた）

mm...m：

レコード数（10 進数 8 けた）

nn...n：

物理ファイル名（63 文字以内）

2.71　pdls【-d svr】（ユニット及びサーバの状態表示）

2.71.1　pdls【-d svr】の形式と規則

(1)　機能

ユニット及びサーバの状態を表示します。

(2)　形式

```
pdls -d svr 〔-a 〔-b〕〕
            〔{-x ホスト名〔，ホスト名〕… | -u ユニット識別子〔，ユニット識別子〕…}〕
```

(3)　オプション

(a)　-a

ユニット及びサーバの状態を DAT 形式で表示する場合に指定します。

(b)　-b

-a オプション指定時に，タイトル行を付けて表示する場合に指定します。

(c)　-x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。複数のホスト名を指定する場合，コンマの前後には空白を含めないでください。

(d)　-u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。複数のユニット識別子を指定する場合，コンマの前後には空白を含めないでください。

(4)　注意事項

1. pdls -d svr コマンドのリターンコードは，0 の場合は正常終了，4 の場合は警告終了，8 の場合は異常終了となります。8 の場合は，その前に出力されているメッセージを参照してエラー要因を取り除いてください。
2. HiRDB/パラレルサーバの場合，システムマネジャ以外のユニットが異常終了又は強制終了したとき，終了したユニットをシステムマネジャが検知するまでの間は，そのユニットが終了する前の状態を表示することがあります。

3. 影響分散スタンバイレス型系切り替え機能を適用している場合，系切り替えで他ユニットに移動したサーバは，移動先のユニットに属するサーバとして表示します。
4. -a 〔-b〕オプションは，-d svr 省略時も指定できます（指定例：pdls -a -b）。
5. 影響分散スタンバイレス型系切り替え機能を適用している場合，ユニットがサーバの系切り替えを検知するまでの間は，系切り替え実行サーバが系切り替え元と系切り替え先の，両方のユニットで表示されることがあります。
6. ホスト，及びユニットの状態によっては，ユニット，及びサーバの情報が表示されないことがあります（すべてのホスト，及びユニットが障害状態の場合，-b オプション指定時はタイトル行だけ表示します）。
7. エラーメッセージが表示された場合でも，リターンコードにエラー値が設定されないことがあります。したがって，pdls -d svr -a 〔-b〕コマンドの実行結果は，エラーメッセージの出力有無と，リターンコードに正常値以外が返されているかどうかの両方を確認する必要があります。
8. pdstart -r コマンドで HiRDB を開始した状態で pdls -d svr コマンドを実行した場合，リターンコード 4 で終了します。
9. HiRDB 開始後に，pdstop -s コマンドですべてのフロントエンドサーバを終了した状態で pdls -d svr コマンドを実行した場合，リターンコード 0 で終了します。

(5)　出力形式

(a)　-a オプション省略時

```
HOSTNAME(aabbcc) UNITID SVID   STATUS  STARTTIME
dd...d           eeee   ff...f gg...g  hhiijj
   :               :      :      :       :
```

(b)　-a オプション指定時

```
"dd...d","eeee","ff...f","gg...g","hhiijj"
   :       :       :        :        :
```

(c)　-a 及び-b オプション指定時

```
"HOSTNAME","UNITID","SVID","STATUS","STARTTIME"
"dd...d","eeee","ff...f","gg...g","hhiijj"
```

［説明］

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

aabbcc：

pdls コマンドを実行した時刻（時分秒）

dd...d：

ホスト名（32 文字以内）

eeee：

ユニット識別子（4 文字）

ff...f：

サーバ名（8 文字以内）

ユニットの状態を表示している場合は＊＊＊＊＊＊＊＊を表示

gg...g：

ユニット又はサーバのステータス情報（8 文字以内）

ACTIVE：

稼働中

STOP：

停止状態

STOP(N)：

停止状態

HiRDB システム起動時，又はシステムマネジャユニットの系切り替え時に起動していないユニットについても表示されることがあります。

STOP(F)：

pdstop コマンド（-f オプション指定）で強制停止

STOP(A)：

次のどれかの状態です。

- システム稼働中に異常終了を検出した
- ユニット bbbb への通信，又はユニット bbbb からの受信ができない
- ユニット bbbb へのクライアント接続ができない

ユニット bbbb で ps コマンドを実行して，HiRDB プロセスが起動中かどうかを確認してください。

HiRDB プロセスが起動中の場合：

通信障害，又はユニット bbbb でサービスのスローダウンが発生していることが考えられます。この場合，KFPS05289-E メッセージが出力される前に出力されたメッセージから原因を調査して，pdstop -z コマンドでユニットを停止した後，再度起動してください。

HiRDB プロセスが停止中の場合：

異常終了している可能性があります。異常終了時のメッセージに従って対策してください。再度起動する場合は，pdstart -x コマンドを実行してください。

START(I)：

データベース初期設定ユティリティ実行待ちの状態（ディクショナリ未初期化）

SUSPEND：

フロントエンドサーバがデータディクショナリ用 RD エリアの回復待ちやディクショナリサーバの起動待ちなどでウェイトしている状態（ウェイトしている原因を取り除いた後，pdstart -a コマンドで起動してください）。

STARTING：

開始準備中

STOPPING：

停止準備中

TRNPAUSE：

新規トランザクションのスケジューリング抑止中。

該当するサーバで新規トランザクションのスケジューリングを抑止しています。KFPS01160-E メッセージが出力されている場合は，システムログの空き容量が不足した原因を取り除いてください。システムログの空き容量が不足した原因の調査方法及び対処方法については，マニュアル「HiRDB システム運用ガイド」の「空き率が警告値未満になったときの HiRDB 管理者の処置」を参照してください。

hhiijj：

起動時刻（時分秒）

2.72　pdls【-d trn】（サーバのトランザクションの状態表示）

2.72.1　pdls【-d trn】の形式と規則

(1)　機能

サーバのトランザクションの状態を表示します。

(2)　形式

```
pdls -d trn 〔-t トランザクション識別子〕 〔- {a | c} 〕 〔-s サーバ名〕 〔-C 〔-H〕〕
          〔 {-x ホスト名〔, ホスト名〕… | -u ユニット識別子〔, ユニット識別子〕…} 〕
```

(3)　オプション

(a)　-t トランザクション識別子　～＜英数字＞((16 文字))

指定したトランザクション識別子のトランザクションに関する情報を表示します。省略した場合，すべてのトランザクションに関する情報を表示します。

(b)　- {a ｜ c}

-a：

トランザクションに関するすべての情報を表示します。

-c：

トランザクションブランチ数を表示します。

省略した場合，トランザクションに関するすべての情報（一部表示されない情報があります）が表示されます。

(c)　-s サーバ名　～＜識別子＞((1～8))

フロントエンドサーバ名又はバックエンドサーバ名を指定します。

省略した場合，該当するユニットで管理しているすべてのトランザクションに関する情報が表示されます。

フロントエンドサーバのサーバ名を指定した場合，指定したフロントエンドサーバが管理しているすべてのグローバルトランザクションに関する情報が表示されます。バックエンドサーバのサーバ名を指定した場合，指定したバックエンドサーバが管理している，グローバルトランザクションから分岐したすべてのトランザクションに関する情報が表示されます。

(d) -C

サーバのトランザクションの状態を，DAT 形式で出力する場合に指定します。

(e) -H

-C オプションを指定した場合に，タイトル行を付けて出力するときに指定します。

(f) -x ホスト名〔，ホスト名〕...

特定のホストについて情報を表示したい場合，そのホスト名を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-x オプションの指定は無視されます。また，複数のホスト名を指定する場合は，コンマの前後に空白を含めないでください。

(g) -u ユニット識別子〔，ユニット識別子〕...

特定のユニットについて情報を表示したい場合，そのユニット識別子を指定します。

-s オプションと同時に指定した場合は，-s オプションの指定が有効となり，-u オプションの指定は無視されます。また，複数のユニット識別子を指定する場合は，コンマの前後に空白を含めないでください。

(4) 注意事項

1. pdls -d trn コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(5) 出力形式

(a) -a オプションを指定した場合

```
HOSTNAME : aa...a(bbccdd)
TRNGID          TRNBID          STATUS                PID  SVID    B-SVID
eeeeffffgg...g  eeeeffffhh...h  jj...j(k,l)       mm...mm  ii...i  nn...n
                PROGRAM                       TIME   SYSINF
                oo...o                        ppqqrr ss...s
                S-PID=tt...t C-PID=uu...u(vv...v)
                TMID=wwww XID=xx...x,yy...y ENVGRP=zzzz
                XDS=BB...B NODE-ID=CCCC
                READING-BLOCK-NO=DD...D
```

[説明]

トランザクションが複数ある場合，トランザクションの数だけ繰り返し表示されます。-a 省略時は，PROGRAM，TIME，SYSINF，及び S-PID～ENVGRP の情報が表示されません。

TRNGID「トランザクション識別子」から B-SVID「分岐元サーバ名」まで，及び PROGRAM「UAP の識別子情報」から SYSINF「システムが使用する内部情報」までの各項目は，1 文字以上の半角空白を空けて表示します。

aa...a：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

eeeeffffgg...g 及び **eeeeffffhh...h**：

トランザクション識別子（16 文字）

eeee：

HiRDB 識別子（4 文字）

ffff：

ユニット識別子（4 文字）

gg...g：

グローバルトランザクション番号（16 進数 8 けた）

hh...h：

トランザクションブランチ番号（16 進数 8 けた）

ii...i：

サーバ名（8 文字以内）

jj...j：

トランザクション第 1 状態（16 文字以内）

BEGINNING：

トランザクション開始処理中

ACTIVE：

トランザクション実行中

SUSPEND：

トランザクション中断中

IDLE：

同期点処理へ移行中

PREPARE：

コミット（1 相目）処理中

READY：

コミット（2 相目）処理中

H_COMMIT：

システムが独自に決定したコミットの処理中

H_ROLLBACK：

システムが独自に決定したロールバックの処理中

COMMIT：

コミット処理中

ROLLBACK_ACTIVE：

ロールバック処理待ち状態

ROLLBACK：

ロールバック処理中

H_FORGETTING：

システムが独自に決定したトランザクションの終了処理中

FORGETTING：

トランザクション終了処理中

注　H_COMMIT，H_ROLLBACK，及び H_FORGETTING については，障害時にユーザからの決着指示を待たないで，HiRDB が独自に決着方法を決定します。

k：

トランザクション第 2 状態

u：

データベースアクセス処理をするプロセスで，データアクセス実行中

r：

トランザクション回復プロセスで，トランザクションの回復処理実行中

p：

トランザクション回復プロセスで，トランザクションの回復処理待ち状態

l：

トランザクション第 3 状態

s：

通信中（サーバ間でのトランザクション決着指示の通信中）

r：

通信待ち中（サーバ間でのトランザクション決着指示の通信待ち中）

w：

同期合わせ中（トランザクションマネジャとサーバ間でのトランザクション同期合わせ中）

n：

通信中でない（s，r の状態でない）

mm...m：

プロセス ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，SVID（サーバ名）と B-SVID（分岐元サーバ名）の表示位置は右にシフトします。

ただし，トランザクションの回復処理待ち状態のトランザクションなど，トランザクションを実行しているプロセスがない場合は，0 を表示します。

nn...n：

分岐元サーバ名（8 文字以内）

ただし，自サーバが要求元の場合は********を表示

oo...o：

UAP の識別情報（30 文字以内）

クライアント環境定義の PDCLTAPNAME に指定した UAP の識別名称を表示します。PDCLTAPNAME が設定されていない場合は Unknown を表示します。

ユティリティの場合は，ユティリティのコマンド名を表示します。ただし，ユティリティサーバプロセス上で動作していた場合や，障害発生によって UAP 識別情報が取得できなかった場合など，情報を表示しない（DAT 形式出力時は空要素を表示する）ことがあります。また，DAT 形式の場合，PDCLTAPNAME の末尾に半角空白を指定していると，末尾の半角空白は削除された状態で表示されます。

ppqqrr：

サービス要求受け付け時刻（時分秒）

ss...s ：

システムが使用する内部情報（32 文字以内）

tt...t：

UAP 又はユティリティの処理の延長で，トランザクション処理を受け付けたサーバのプロセス ID（99999 以下の場合，10 進数 5 けた以内で右詰め表示，100000 以上の場合は，10 進数 10 けた以内で左詰め表示）

100000 以上の場合，C-PID（サーバのプロセスと接続していたクライアント（UAP 又はユティリティ）のプロセス ID と IP アドレス）の表示位置は右にシフトします。

ユティリティサーバプロセス上で動作していた場合や，障害発生によってプロセス ID が取得できなかった場合など，0 を表示することがあります。

uu...u：

サーバのプロセスと接続していたクライアント（UAP 又はユティリティ）のプロセス ID（10 進数 10 けた以内で左詰め表示）

なお，次に示す場合は 0 が表示されます。

- クライアントと接続していないサーバプロセスの場合
- UAP とリンケージしているクライアントライブラリのバージョンが 04-00 より前の場合
- Type4 JDBC ドライバから接続しているサーバプロセスの場合

vv...v：

サーバのプロセスと接続していたクライアント（UAP 又はユティリティ）の IP アドレス（15 文字以内）

クライアントと接続していたサーバプロセスだけ有効です。クライアントと接続していないサーバプロセスの場合は 0.0.0.0 と表示します。

wwww：

接続している OLTP の OLTP 識別子（4 文字以内）

クライアント環境変数 PDTMID を省略した場合は，"****"を表示します。

接続元が XDS クライアントの場合は，"HRDB"と表示します。

xx...x,yy...y：

OLTP 又は XDS から与えられたトランザクション識別子（33 文字以内）

zzzz：

OLTP システムから与えられた環境変数グループ識別子（4 文字）

複数接続機能を使用している場合に表示されます。複数接続機能を使用していない場合は，"****"を表示します。

BB...B：

XDS サーバ名（8 文字以内）

接続元が XDS の場合にかぎり表示します。

CCCC：

ノード識別子（4 文字固定）

接続元が XDS の場合にかぎり表示します。

接続元 XDS の XDS サーバ定義の pdq_node_id オペランドに指定したノード識別子を表示します。

DD...D：

入力中のシステムログのブロック番号（8 文字以内）

ロールバック処理でシステムログの入力処理を実行中の場合だけ表示（DAT 形式出力時は空要素を表示）します。ロールバック処理が進行中であることを確認する方法は，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜出力例＞

プロセス ID が 100000 以上の表示を含む場合

```
HOSTNAME : HOST0001(152813)
TRNGID           TRNBID           STATUS          PID  SVID     B-SVID
PRDBUNT001000005 PRDBUNT000010004 ACTIVE(u,n)   38095 SERVER01 ********
                 PROGRAM                 TIME   SYSINF
                 uap001                  152659 p_f_sqa_csetup
                 S-PID= 6888 C-PID=7063(10.1.0.1)
… プロセスIDが99999以下
PRDBUNT001000006 PRDBUNT000010005 ACTIVE(u,n)   108221 SERVER01 ********
                 PROGRAM                 TIME   SYSINF
                 uap002                  152755 p_f_sqa_csetup
                 S-PID= 7022 C-PID=7068(10.1.0.1)
… プロセスIDが100000以上
PRDBUNT001000007 PRDBUNT000010006 ACTIVE(u,n)    8101 SERVER01 ********
                 PROGRAM                 TIME   SYSINF
                 uap003                  152810 p_f_sqa_csetup
                 S-PID= 6887 C-PID=7069(10.1.0.1)
… プロセスIDが99999以下
```

(b)　-a -C -H オプションを指定した場合

```
"EDIT-TIME","HOSTNAME","TRNGID","TRNBID","STATUS1","STATUS2","STATUS3","PID","SVID","B
-SVID","PROGRAM","TIME","SYSINF","S-PID","C-PID","C-ADDR","TMID","XID","ENVGRP","XDS",
"NODE-ID","READING-BLOCK-NO"
"AA...A","aa...a","eeeeffffgg...g","eeeeffffhh...h","jj...j","k","l",mm...m,"ii...i","
nn...n","oo...o","ppqqrr","ss...s",tt...t,uu...u,"vv...v","wwww","xx...x,yy...y","zzzz",
"BB...B","CCCC","DD...D"
```

[説明]

aa...a～DD...D については，-a オプションを指定した場合の説明を参照してください。

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

1 行には，1 トランザクションの情報を表示します。

AA...A：

pdls コマンドを実行した時刻を YYYY/MM/DD hh:mm:ss の形式で表示します。

YYYY：年（西暦）　MM：月　DD：日　hh：時　mm：分　ss：秒

(c)　-c オプションを指定した場合

```
HOSTNAME : aa...a(bbccdd)
TRNGID          SVID   B-SVNUM
eeeeffffgg...g  hh...h ii...i
```

[説明]

トランザクションが複数ある場合，トランザクションの数だけ繰り返し表示されます。

TRNGID「トランザクション識別子」から B-SVNUM「トランザクションブランチ数」までの各項目は，1 文字以上の半角空白を空けて表示します。

aa...a ：

ホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

eeeeffffgg...g：

トランザクション識別子（16 文字）

eeee ：

HiRDB 識別子（4 文字）

ffff ：

ユニット識別子（4 文字）

gg...g：

グローバルトランザクション番号（16 進数 8 けた）

hh...h ：

サーバ名（8 文字以内）

-s オプションを省略した場合は，"********"を表示します。

ii...i ：

トランザクションブランチ数（10 進数 5 けた）

-s オプションを指定した場合は，該当するサーバが管理している，トランザクション識別子が同一のトランザクションブランチ数となります。-s オプションを省略した場合は，該当するユニットが管理している，トランザクション識別子が同一のトランザクションブランチ数となります。

(d)　-c -C -H オプションを指定した場合

```
"EDIT-TIME","HOSTNAME","TRNGID","SVID","B-SVNUM"
"jj...j","aa...a","eeeeffffgg...g","hh...h",ii...i
```

[説明]

aa...a～ii...i については，-c オプションを指定した場合の説明を参照してください。

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

1 行には，1 トランザクションの情報を表示します。

jj...j：

pdls コマンドを実行した時刻を YYYY/MM/DD hh:mm:ss の形式で表示します。

YYYY：年（西暦）　MM：月　DD：日　hh：時　mm：分　ss：秒

2.73　pdls【-d ust】（ユニットの稼働状態の表示）

2.73.1　pdls【-d ust】の形式と規則

(1)　機能

自ユニットの稼働状態を表示します。

このコマンドを実行したユニットの稼働状態が表示されます。

(2)　形式

```
pdls -d ust 〔-C 〔-H〕〕 〔-a〕
```

(3)　オプション

(a)　-C

ユニットの稼働状態を，DAT 形式で出力する場合に指定します。

(b)　-H

-C オプションを指定した場合に，タイトル行を付けて出力するときに指定します。

(c)　-a

ユニットの開始状態を表示する場合に指定します。

(4)　注意事項

1. pdls -d ust コマンドのリターンコードを次に示します。バージョン 09-50 以降の HiRDB では，リターンコードに 1 を追加しています。

   0：

   ユニット稼働中（コマンド実行ユニットにフロントエンドサーバがある場合は，フロントエンドサーバに接続できます）

   1：

   エラー終了（HiRDB の稼働状態を判別できないため，コマンドがエラー終了したことを示しています。コマンドエラー時に出力されているメッセージを基に対処してください）

   4：

   ユニット開始途中（コマンド実行ユニットにフロントエンドサーバがありますが，フロントエンドサーバに接続できません），又はユニット停止途中

8：

プロセスサーバプロセスの再起動中断状態

12：

ユニットの停止状態

16：

OS からの登録が削除された状態

2. ユニットの状態が"PAUSE"の場合，フロントエンドサーバの状態が"SUSPEND"のままで"ONLINE"にならない場合，又はセットアップ状態が"UNSETUP"の場合は，次のどれかの対処をしてください。
   - ユニットの状態が"PAUSE"の場合は障害が発生しています。KFPS00715-I メッセージ，及びそれ以前にイベントログに出力されたメッセージを参照して，障害要因を取り除いた後，サービスを再起動してください。
   - フロントエンドサーバの状態が"SUSPEND"のままで"ONLINE"にならない場合は，障害が発生している可能性があります。イベントログに出力されたメッセージを参照して，障害要因を取り除いた後，pdstart -a コマンドでフロントエンドサーバの SUSPEND 状態を解除してください。
   - セットアップ状態が"UNSETUP"の場合はサービスが停止しています。HiRDB を開始する前に，サービスを開始してください。

(5) 出力形式

(a) -C オプションを省略した場合

```
HOSTNAME  : aa...a(bbccdd)
SYSTEMID  : eeee
UNITID    : ffff
ENTRYHOST : gg...g
PAIRHOST  : hh...h
UNIT-STAT FES-STAT  SETUP-STAT
iiiiiiii  jjjjjjjj  kkkkkkk
START-OPT : llllllll
```

[説明]

aa...a：

現用系のホスト名（32 文字以内）

bbccdd：

pdls コマンドを実行した時刻（時分秒）

eeee：

HiRDB 識別子（4 文字）

ffff：

ユニット識別子（4 文字）

gg...g：

pdls コマンドを実行したユニットのホスト名（32 文字以内）

hh...h：

IP アドレス引き継ぎなしのスタンバイ型系切り替え機能，又は 1:1 スタンバイレス型系切り替え機能を適用している場合，相手系ホスト名（32 文字以内）。

上記以外の場合，情報は表示されません（DAT 形式出力時は空要素が表示されます）。

iiiiiiii：

ユニットの状態（8 文字）

PAUSE△△△：プロセスサーバプロセスの再起動中断状態

STOP△△△△：停止状態

STARTING：開始途中，又は系切り替え機能を使用している場合でユニットが待機中

ONLINE△△：稼働状態

STOPPING：停止途中

△は，半角の空白文字を表します。なお，DAT 形式出力時は△は表示されません。

jjjjjjj：

フロントエンドサーバの状態（8 文字）

STOP△△△△：停止状態，又は系切り替え機能を使用している場合で，ユニットが待機中

STARTING：開始途中

SUSPEND△：サスペンド状態

ONLINE△△：稼働状態

STOPPING：停止途中

************：ユニットにフロントエンドサーバなし

△は，半角の空白文字を表します。なお，DAT 形式出力時は△は表示されません。

HiRDB/シングルサーバの場合は，常に********となります。

kkkkkkk：

セットアップ状態（7 文字）

SETUP△△：サービス稼働中

UNSETUP：サービス停止中

△は，半角の空白文字を表します。なお，DAT 形式出力時は△は表示されません。

lllllll：

ユニットの開始状態（8 文字）

-a オプションを指定したときだけ表示されます。

RECOVERY：pdstart -r コマンドでユニットを開始しています。

NORMAL△△：上記以外の方法で開始しています。

△は，半角の空白文字を表します。なお，DAT 形式出力時は△は表示されません。また，UNIT-STAT が PAUSE 及び STOP の場合は，********が表示されます。

(b)　-C -H オプションを指定した場合

```
"EDIT-TIME","HOSTNAME","SYSTEMID","UNITID","ENTRYHOST","PAIRHOST","UNIT-STAT","FES-ST
AT","SETUP-STAT","START-OPT"
"mm...m","aa...aa","eeee","ffff","gg...g","hh...h","iiiiiiii","jjjjjjjj","kkkkkkk","ll
llllll"
```

[説明]

aa...a～llllllll については，-C オプションを省略した場合の説明を参照してください。

DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

1 行には，自ユニットの稼働状態を表示します。

mm...m：

pdls コマンドを実行した時刻を YYYY/MM/DD hh:mm:ss の形式で表示します。

YYYY：年（西暦）　MM：月　DD：日　hh：時　mm：分　ss：秒

2.74　pdmemdb（インメモリデータ処理に関するコマンド）

2.74.1　pdmemdb の形式と規則

(1)　機能

pdmemdb コマンドで次に示すことができます。

- RD エリアのインメモリ化
- インメモリ化の解除
- インメモリデータバッファへのデータの再読み込み
- インメモリデータバッファ上のデータの破棄

なお，pdmemdb コマンドの実行結果は，pddbls -M コマンドで確認できます。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdmemdb -k 処理種別　{-r RDエリア名〔, RDエリア名〕…|-r ALL}
                      〔-p ページ指定種別〕〔-d〕〔-W 実行監視時間〕
```

(4)　オプション

(a)　-k 処理種別

pdmemdb コマンドで実行する処理を指定します。

stay：

-r オプションで指定した RD エリアをインメモリ化する場合に指定します。

rels：

-r オプションで指定したインメモリ RD エリアのインメモリ化を解除します。

reload：

-r オプションで指定したインメモリ RD エリア内のデータをインメモリデータバッファに再読み込みします。

cancel：

-r オプションで指定したインメモリ RD エリアに割り当てられているインメモリデータバッファ上の更新情報を破棄します。

各オプションの指定によって，pdmemdb コマンド実行後のインメモリデータバッファ及び RD エリアの状態が異なります。詳細については，表「pdmemdb コマンド実行時の前提条件（インメモリデータバッファと対象 RD エリアの状態）」を参照してください。

(b)　-r RD エリア名〔，RD エリア名〕...　～＜識別子＞((1～30))

処理対象の RD エリアの名称を指定します。

次に示す RD エリアを指定した場合，pdmemdb コマンドがエラーになります。

- ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアを指定した場合
- RD エリア名の一括指定時，ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアが対象になった場合
- オープン属性が SCHEDULE の RD エリアを指定した場合
- 共用 RD エリアを指定した場合
- グローバルバッファを割り当てていない RD エリアを指定した場合

＜規則＞

RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。

(c)　-r ALL

このオプションを指定した場合，-k オプションで指定した処理を実行できるすべての RD エリアを処理対象にします。

全インメモリ RD エリアのインメモリ化を一度に解除するときなどに，このオプションを指定します。

なお，RD エリアをインメモリ化するときにこのオプションを指定すると，必要のない RD エリアをインメモリ化するおそれがあるため，インメモリ化するときはこのオプションを指定しないことをお勧めします。

注意事項

監査証跡表及び改竄防止表に定義したインデクスがある場合，-r ALL 指定をしないでください。-r ALL 指定をすると，監査証跡表又は改竄防止表を格納している RD エリアもインメモリ化されます。監査証跡表又は改竄防止表を格納している RD エリアをインメモリ化してはいけない理由については，注意事項を参照してください。

次に示す RD エリアは，-r ALL 指定時に処理対象となりません。

- ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリア
- オープン属性が SCHEDULE の RD エリア
- 共用 RD エリア

- グローバルバッファを割り当てていない RD エリア
- インメモリデータバッファ及び RD エリアの状態が正しくない RD エリア（正しい状態については，表「pdmemdb コマンド実行時の前提条件（インメモリデータバッファと対象 RD エリアの状態）」を参照）

(d)　-p ページ指定種別《free》

このオプションは-k オプションに stay を指定した場合に指定できます。

インメモリデータバッファが使用する共用メモリを実メモリ上にページ固定するかどうかを指定します。

free：

インメモリデータバッファが使用する共用メモリを実メモリ上にページ固定しません。

fixed：

インメモリデータバッファが使用する共用メモリを実メモリ上にページ固定します。なお，Windows のバージョンによってはページ固定できない場合があります。使用している Windows がページ固定できるかどうかについては，pdntenv -os コマンドで確認できます。

インメモリデータバッファが使用する共用メモリを実メモリ上にページ固定（fixed 指定）すると，共用メモリのページングを防止できるため，共用メモリに対するアクセス性能が向上します。

インメモリデータバッファが使用する共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。実メモリに比べて大きなメモリをページ固定すると，ページングの多発や仮想メモリ不足の原因になります。したがって，実メモリや全仮想メモリに対する共用メモリプールの占める割合を検討する必要があります。インメモリデータバッファが使用する共用メモリの計算式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

ページ固定は，実メモリサイズからページ固定する共用メモリサイズを減算したサイズが，スワップ領域サイズからページ固定する共用メモリサイズを減算したサイズの 1/2 以上にならないことを目安にしてください。

fixed を指定してインメモリデータバッファが使用する共用メモリを実メモリ上にページ固定すると，共用メモリのページサイズが通常のページサイズ（4KB）からラージページサイズ（2MB）に拡大されます。ページサイズが拡大すると，仮想アドレスを実アドレスに変換するための管理領域（PTE：Page Table Entry）を通常よりも小さくできるため，メモリ不足を防止できます。また，仮想アドレスと物理アドレスの変換回数が少なくなり，ページフォルトの発生を抑えることができます。これによって，トランザクション性能が向上することがあります。

(e)　-d

このオプションは-k オプションに rels を指定した場合に指定できます。

インメモリ化を強制解除する場合にこのオプションを指定します。インメモリ化を強制解除すると，インメモリデータバッファ上の更新データはインメモリ RD エリアに反映されません。

(f)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdmemdb コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　リターンコード

pdmemdb コマンドのリターンコードを次に示します。

- 0：正常終了
- 4：警告

  -r オプションで複数の RD エリアを指定した場合，一部の RD エリアの処理が正常に終了しなかったときのリターンコードです。
- 8：異常終了
- 12：異常終了

  エラーメッセージを表示できない場合にリターンコード 12 が表示されます。この場合，イベントログのエラーメッセージを参照してください。HiRDB/パラレルサーバの場合はディクショナリサーバを定義したサーバマシンのイベントログを参照してください。エラーの要因を取り除いた後に，再度 pdmemdb コマンドを実行してください。イベントログにエラーメッセージが出力されていない場合は，保守員に連絡してください。

(6)　規則

1. HiRDB が稼働中のときに pdmemdb コマンドを実行できます。
2. HiRDB/パラレルサーバの場合，システムマネジャを定義したサーバマシンで pdmemdb コマンドを実行してください。
3. データディクショナリ用 RD エリアが次に示すどちらかの状態のときに，pdmemdb コマンドを実行できます。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，かつコマンド閉塞状態
4. pdmemdb コマンドは，HiRDB Accelerator が正しくインストールされていて，かつ pd_max_resident_rdarea_no オペランドに 1 以上が指定されている場合に実行できます。
5. pdmemdb コマンドを実行する前に，インメモリデータバッファと対象 RD エリアの状態を確認してください。正しい状態になっていないと pdmemdb コマンドを実行できません。pdmemdb コマンド実行時の前提条件（インメモリデータバッファと対象 RD エリアの状態）を次の表に示します。

表 2–12　pdmemdb コマンド実行時の前提条件（インメモリデータバッファと対象 RD エリアの状態）

<table>
<tr><th colspan="2" rowspan="2">pdmemdb コマンドの-k オプションの指定</th><th colspan="2">コマンド実行時の前提条件</th><th colspan="2">コマンド実行後の状態</th></tr>
<tr><th>インメモリデータバッファの状態</th><th>RD エリアの状態</th><th>インメモリデータバッファの状態</th><th>RD エリアの状態</th></tr>
<tr><td colspan="2">stay</td><td>未使用状態</td><td>コマンド閉塞かつクローズ状態</td><td rowspan="2">DB 同期状態</td><td rowspan="2">コマンド閉塞かつクローズ状態</td></tr>
<tr><td colspan="2">reload</td><td>バッファ障害状態</td><td>障害閉塞かつクローズ状態</td></tr>
<tr><td rowspan="5">rels</td><td>-d オプション指定なし</td><td>DB 同期状態</td><td>コマンド閉塞かつクローズ状態</td><td rowspan="5">未使用状態（インメモリ化の解除）</td><td>コマンド閉塞かつクローズ状態</td></tr>
<tr><td rowspan="4">-d オプション指定あり</td><td>DB 非同期状態</td><td>コマンド閉塞かつオープン状態</td><td rowspan="4">障害閉塞かつクローズ状態</td></tr>
<tr><td>バッファ障害状態</td><td>障害閉塞かつクローズ状態</td></tr>
<tr><td>RD エリア障害状態</td><td>コマンド閉塞かつオープン状態</td></tr>
<tr><td>バッファ障害状態かつ RD エリア障害状態</td><td>障害閉塞かつクローズ状態</td></tr>
<tr><td colspan="2">cancel</td><td>DB 非同期状態</td><td>コマンド閉塞かつオープン状態</td><td>バッファ障害状態</td><td>障害閉塞かつオープン状態</td></tr>
</table>

(7)　注意事項

1. RD エリアの容量に比例して pdmemdb コマンドの処理時間が長くなります。その間，シンクポイントが有効化されないため，注意してください。
2. -k オプションに stay，reload，rels のどれかを指定してコマンドを実行したときにコマンドプロセスを kill すると，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了することがあるため，コマンドプロセスを kill しないでください。
3. インメモリ RD エリアがある場合に HiRDB が強制終了又は異常終了すると，インメモリデータバッファ上の更新情報はインメモリ RD エリアに反映されません。このとき，インメモリ RD エリアが障害閉塞することがあります。
   - ユーザ用 RD エリア，及びユーザ LOB 用 RD エリアの場合

     DB 同期状態以外のときは，ユーザ用 RD エリア，及びユーザ LOB 用 RD エリアが障害閉塞します。
   - リスト用 RD エリアの場合

     RD エリア障害状態又は両系障害状態（バッファ障害状態かつ RD エリア障害状態）のときは，リスト用 RD エリアが障害閉塞します。

このため，インメモリ化を行った場合は，必ずバックアップを取得してください。バックアップは，次の場合に必要となります。

- 両系障害状態（バッファ障害状態かつ RD エリア障害状態）となった場合
- 最新の状態に回復する場合
- 関連する RD エリアをインメモリ化していない場合（インメモリ RD エリアに対して定義系 SQL を実行する場合を含む）

バックアップ取得の運用については，マニュアル「HiRDB バッチ高速化機能」を参照してください。

4. インメモリ RD エリアがあるディスクのペアボリューム操作を行わないことをお勧めします。インメモリデータバッファとインメモリ RD エリアの同期状態（DB 同期状態なのか DB 非同期状態なのか）を正しく認識しないでペアボリューム操作を行うと，データの不整合が発生するおそれがあります。
5. インメモリ化の処理中（pdmemdb コマンドの実行からインメモリ化を完了するまでの間）は，pdstop，pdchgconf，又は pdprgrenew コマンドを実行しないでください。実行して両方のコマンドが正常終了した場合，インメモリデータバッファを割り当てた共用メモリセグメントが解放されない可能性があります。この場合は，OS の機能を使って共用メモリセグメントの使用状況を調べて，必要に応じて共用メモリセグメントの解放を行ってください。
6. HiRDB が強制終了又は異常終了すると，インメモリデータバッファ上の更新情報がインメモリ RD エリアに反映されません。このため，監査証跡表又は改竄防止表を格納している RD エリアをインメモリ RD エリアにしないでください。

   監査証跡表又は改竄防止表も，ほかの表と同様にデータベース回復ユティリティを使用すれば，更新情報を反映してインメモリ RD エリアを最新の状態に回復できますが，セキュリティ上の観点から，インメモリデータ処理の適用はお勧めしません。

   また，監査証跡表の自動データロード機能は更新前ログ取得モードで動作するため，自動データロード機能を使用している場合は，監査証跡表に更新情報を反映できなくなるおそれがあります。
7. pdmemdb コマンドは，指定した RD エリアに対して EX モードで排他制御をします。そのため，指定した RD エリアにほかのトランザクションがアクセス中の場合は，そのトランザクションの終了まで pdmemdb コマンドは待ち状態になります。
8. -r オプションで複数の RD エリアを指定した場合（複数の RD エリアに対して一度に処理を行う場合），一部の RD エリアに対する処理でエラーが発生すると，すべての RD エリアの処理を無効化します。

   なお，HiRDB/パラレルサーバの場合は，バックエンドサーバごとにこの仕組みが適用されます。HiRDB/パラレルサーバのときの例を次に示します。

   （例）

   ```
   pdmemdb -k stay -r RD01,RD02,RD03,RD04
   ```

   RD01 と RD02 はバックエンドサーバ 1 で，RD03 と RD04 はバックエンドサーバ 2 で管理されています。RD02 のインメモリ化に失敗した場合，RD01 もインメモリ化されません。RD03 と RD04 はインメモリ化されます。

   なお，このとき，pdmemdb コマンドのリターンコードに 4 が返されます。

参考

この規則は，一括指定の場合や，ALL 指定の場合も適用されます。

直接指定及び一括指定の場合と，ALL 指定の場合の，エラー処理の差異を次に示します。

pdmemdb コマンド実行時の条件又はエラー発生時の条件	直接指定及び一括指定の場合	ALL 指定の場合
指定した RD エリアに，共用 RD エリアなどのインメモリデータ処理の適用外となる RD エリアがある場合	pdmemdb コマンドがエラーになります。指定した RD エリアは一つもインメモリ化されません。	インメモリデータ処理の適用外となる RD エリアはインメモリ化されません。残りの RD エリアはすべてインメモリ化されます。
指定した RD エリアのうち，共用メモリ不足などで一部の RD エリアに対する処理でエラーが発生した場合	注意事項 8 で説明した処理が行われます。	

2.75　pdmstchk（マスタディレクトリ用 RD エリアの整合性チェック）

2.75.1　pdmstchk の形式と規則

(1)　機能

マスタディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディレクトリ用 RD エリアの整合性を確認します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdmstchk 〔-k チェック対象〕 〔-u 認可識別子〕
```

(4)　オプション

(a)　-k チェック対象

コマンド実行時にチェック対象となる資源を指定します。

all：

以下のすべての資源をチェックします。

rdarea：

RD エリアの情報をチェックします。

table：

表の情報をチェックします。

index：

インデクスの情報をチェックします。

view：

ビュー表の情報をチェックします。

plugin：

プラグインの情報をチェックします。

(b)　-u 認可識別子

ディクショナリ表を検索するユーザの認可識別子を指定します。

CONNECT 権限もしくは DBA 権限を持つユーザを指定してください。

このオプションを指定すると，パスワード入力要求の応答付きメッセージが出力されますので，応答付きメッセージが出力されたらパスワードを入力してください。

認可識別子を「"¥"」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「"¥"」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

パスワードを「"」で囲んだ場合は，英大文字と英小文字が区別されます。「"」で囲まない場合は，すべて英大文字として扱われます。

このオプションを省略した場合，環境変数 PDUSER の設定値を仮定します。このオプションを省略し，かつ環境変数 PDUSER も設定していない場合は，このコマンドは実行できません。

(5) 規則

1. pdmstchk コマンドは，HiRDB 稼働中のときだけ実行できます。
2. pdmstchk コマンドは，シングルサーバ又はディクショナリサーバを定義しているユニットで実行してください。
3. pdmstchk コマンド実行時は，環境変数 PDHOST，PDNAMEPORT を設定してください。

(6) リターンコード

pdmstchk コマンドのリターンコードを次に示します。

リターンコード	内容
0	正常終了（整合性に問題なし）
4	正常終了（整合性に問題あり）
8	異常終了（コマンド実行中にエラーが発生）

(7) 整合性に問題あるときの対処

pdmstchk コマンドを実行し，不整合を検知した場合，次のどちらかの対処をしてください。

- 不整合になる操作をする前のバックアップから回復する
- HiRDB を再構築する

(8) 注意事項

1. 次のどれかを更新する操作と，pdmstchk コマンドを同時に実行すると，不整合を検知することがあります。このため，同時に実行しないでください。
   - マスタディレクトリ用 RD エリア

- データディレクトリ用 RD エリア
- ディクショナリ表

2. pdchpathf を実行した場合，pdchpathn を実行してから，このコマンドを実行してください。
   pdchpathn コマンドを実行する前に pdmstchk コマンドを実行すると，不整合を検知します。

(9)　出力形式

```
MASTER DIRECTORY RDAREA CHECK(aa....aa)    YYYY/MM/DD hh:mm:ss
[bb....bb]                    ...1
ID=cc....cc, dd....dd         ...1, 2
   :                          ...1, 2
All Data matched              ...3
```

[説明]

1 は，不整合を検知したデータ種別ごとに表示します。同一のデータ種別が複数回表示されることがあります。

2 は，1 のデータ種別の中で不整合になったデータをデータの ID ごとに繰り返し出力します。

3 は，不整合のデータがない場合に出力します。

aa....aa：

チェック対象。-k オプションの指定値を表示します。省略時は all を表示します。

YYYY/MM/DD hh:mm:ss：

コマンド実行日時（年月日時分秒）

bb....bb：

データ種別

表示内容	データ種別
rdarea	RD エリア
table	表
index	インデクス
view	ビュー表
replica	レプリカ RD エリア
plugin	プラグイン

cc....cc：

不一致になったデータの ID

dd....dd：

不一致になった理由を表示します。

理由	説明
Data exists only in MASTER DIRECTORY	この資源はマスタディレクトリ用 RD エリアにだけあります。
Data exists only in DICTIONARY TABLE	この資源はディクショナリ表にだけあります。
Data unmatch between MASTER DIRECTORY and DICTIONARY TABLE	マスタディレクトリとディクショナリ表の情報が異なります。
Data unmatch between MASTER DIRECTORY and DATA DIRECTORY	マスタディレクトリとデータディレクトリの情報が異なります。

2.76　pdntenv（HiRDB の動作環境の設定）

2.76.1　pdntenv の形式と規則

(1)　機能

HiRDB の動作環境を表示，又は設定します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdntenv 〔 {-hc {on｜off} ｜-pn サービスポート名称｜-a｜-os｜-sc {auto｜manual}
            ｜-c 文字コード種別｜-ro {on｜off} ｜-shmfile {regular｜page}
            ｜-sr {リソース名 指定値 〔リソース名 指定値〕…
                  ｜リソース名 〔リソース名〕… -d}
            ｜-wd 〔プロセス名〕｜-sw サービス開始遅延時間｜-v オペランド省略時動作} 〕
```

(4)　オプション

(a)　-hc {on ｜ off}

DAT アクセス時にハードウェア圧縮をするかどうかを指定します。

on（初期値）：

ハードウェア圧縮をします。

off：

ハードウェア圧縮をしません。

DAT 装置が接続しているサーバマシンの指定が有効になります。

この指定は DAT ドライブがハードウェア圧縮をサポートしている場合に限り有効です。

同一 DAT に対する書き込み時と読み出し時の設定は統一してください。

(b)　-pn サービスポート名称

リモート系コマンドのサービスポート名称を指定します。サービスポート名称は，プリフィクスが pdrshsrv に識別子（空白を含まない半角英数字 4 文字以内）を付けた形式で指定します。

（例）pdrshsrvUNT1

%windir%¥system32¥drivers¥etc¥SERVICES ファイルに指定したサービスポート名称とポート番号を指定してください。

次回のサービス開始から指定値が有効になります。

初期値は「pdrshsrv 識別子」です。

(c) -a

HiRDB が複数インストールされている場合（マルチ HiRDB の場合），すべての HiRDB の情報を表示します。

(d) -os

共用メモリのページ固定化に対応した OS かどうかを表示します。ページ固定に対応している Windows の場合，Windows の Large Page でのページサイズも表示します。

(e) -sc {auto | manual}

HiRDB/シングルサーバを pdstop コマンドで終了する場合，同時に Windows のサービスも停止するかどうかを指定します。

auto：

HiRDB を終了する場合に，同時に Windows のサービスも停止するときに指定します。この場合，次回 HiRDB を開始する前に，Windows のサービスを開始する必要があります。

manual（初期値）：

HiRDB を終了する場合に，Windows のサービスを停止しないときに指定します。この場合，次回 HiRDB を開始するときは，Windows のサービスを開始する必要がありません。manual を指定した場合，Windows のサービス制御マネージャを占有する時間を短縮できます。

＜規則＞

1. このオプションを省略した場合，バージョン 06-00 より前の HiRDB のときは auto，バージョン 06-00 以降の HiRDB のときは manual が仮定されます。また，HiRDB を上書きインストールした場合，このオプションの指定値が引き継がれます。
2. このオプションの指定値は，次回のサービス開始から有効となります。
3. HiRDB/パラレルサーバの場合はこのオプションを指定できません。
4. pdchgconf コマンド，又は pdprgrenew コマンドを使用する場合は，必ず manual を指定してください。auto を指定した環境で pdchgconf コマンド，又は pdprgrenew コマンドを実行するとエラーとなり，HiRDB が停止することがあります。

(f) -c 文字コード種別

HiRDB で使用する各国文字データの文字コード種別を指定します。

sjis（初期値）：シフト JIS 漢字コード

lang-c：単一バイト文字コード

chinese：EUC 中国語漢字コード

chinese-gb18030：中国語漢字コード（GB18030）

utf-8：Unicode（UTF-8）

ISO/IEC 10646 の規格では，1 文字当たり 1〜4 バイトの範囲に文字が割り当てられ，5，6 バイトの範囲は将来の規格のために予約されています。HiRDB では 1 文字当たり 1〜6 バイトの範囲まで使用できますが，文字が割り当てられていない 5，6 バイトの範囲を使用する場合，将来発生するおそれのある問題については保証できません。

utf-8_ivs：Unicode（IVS 対応 UTF-8）

ISO/IEC 10646 では，文字が割り当てられているのは 1 文字 1〜4，7 及び 8 バイトの範囲です。5，6，9 及び 10 バイトの範囲は将来の規格のために予約されており，文字が割り当てられていません。したがって，文字が割り当てられていない 1 文字 5，6，9 及び 10 バイトの範囲を使用した場合は，将来発生するかもしれない問題について保証できません。

(g) -ro {on | off}

リモート系コマンドを使用するかどうかを指定します。このオプションは HiRDB/シングルサーバの場合に有効となります。

on：

リモート系コマンドを使用する場合に指定します。次に示す場合にリモート系コマンドを使用します。

- HiRDB/シングルサーバで IP アドレスを引き継がない系切り替え機能を使用する場合
- 簡易セットアップツールを使用して，別マシンの HiRDB/シングルサーバの環境を構築する場合

なお，on を指定する場合は，別途-pn オプションも指定してください。

off（初期値）：

リモート系コマンドを使用しません。

このオプションの指定値は，次回のサービス開始から有効となります。また，HiRDB/パラレルサーバの場合はこのオプションを指定できません。

(h) -shmfile {regular | page}

HiRDB で使用する共用メモリの割り当て先を指定します。省略した場合，前回の割り当て先のままとなります。

regular（初期値）：

共用メモリを HiRDB 運用ディレクトリ下のファイルに割り当てます。

page：

共用メモリをページングファイル（仮想メモリ）に割り当てます。

各指定値の詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「仮想メモリの確認」を参照してください。

共用メモリの割り当て先を変更する手順を次に示します。

＜割り当て先変更手順＞

共用メモリの割り当て先を変更する手順を次に示します。

1. HiRDB を正常終了し，HiRDB のサービスを停止します。
2. pdntenv コマンドの-shmfile オペランドに，共用メモリの割り当て先を指定して実行します。
3. ページングファイル容量（仮想メモリ容量）を拡張し，Windows をリブートします。※
4. HiRDB，及び HiRDB のサービスを再開始します。

注※

ページングファイル容量を見積もった結果，十分なサイズがある場合は，この手順は必要ありません。

＜規則＞

1. regular を指定した場合，HiRDB 運用ディレクトリがあるドライブの容量に，HiRDB で使用する共用メモリ量※を加算する必要があります。
2. page を指定した場合，ページングファイル（仮想メモリ）の最大使用量に，HiRDB で使用する共用メモリ量※を加算する必要があります。このとき，ページングファイルの共用メモリ分は固定的に使用されるため，注意してください。

注※

HiRDB で使用する共用メモリ量については，マニュアル「HiRDB システム導入・設計ガイド」のメモリ所要量の計算式を参照してください。

(i)　-sr {リソース名 指定値 〔リソース名 指定値〕...｜リソース名 〔リソース名〕... -d}

●リソース名 指定値 〔リソース名 指定値〕...

ユニットごとにリソース数及びリソース容量を設定する場合に指定します。

この項目は HiRDB が自動計算して設定するため，通常，指定する必要はありません※。ただし，次の条件に該当する場合は，指定が必要となることがあります。

- HiRDB が自動計算する際に想定している構成よりもリソース数を多く必要とする構成の場合

詳細は，マニュアル「HiRDB システム導入・設計ガイド」の「リソース数に関連する環境変数の見積もり」を参照してください。

なお，リソースの設定方法には，システム環境変数に設定する方法もあります。

設定方法の選択基準については，マニュアル「HiRDB システム導入・設計ガイド」の「リソース数に関連する環境変数の見積もり」を参照してください。

注※

pdntenv コマンドで設定内容を表示した際に，各リソース値の設定方法として”auto”が表示されていない場合は自動計算が有効になっていません。この項目で設定済みか，又はシステム環境変数で設定していることが考えられます。この場合は，リソースの設定をすべて削除し，自動計算を有効にしてください。削除方法については，次の「●リソース名〔リソース名〕... -d」の説明を参照してください。なお，削除にあたっては，マニュアル「HiRDB システム導入・設計ガイド」の「リソース数に関連する環境変数の見積もり」の内容を十分確認してください。

リソース名：

設定するリソース名を指定します。リソース名に指定できる値を次に示します。

msgmni（メッセージキュー識別子数）

msgtql（メッセージキューテーブル数）

semmax（セマフォ識別子数）

shmmax（共用メモリ使用数）

同一リソース名を重複して指定した場合，及びここに示したリソース名以外のものを指定した場合はエラーとなります。

指定値：

リソース数又はリソース容量を指定します。指定値の見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」の「リソース数に関連する環境変数の見積もり」を参照してください。

●リソース名〔リソース名〕... -d

ユニットごとにリソース数及びリソース容量の設定を削除する場合に指定します。ただし，システム環境変数での設定を実施している場合は，システム環境変数を削除，及び OS を再起動しないと，システム環境変数での指定値が有効となります。

リソース名：

削除する（未設定状態にする）リソース名を指定します。

(j) -wd〔プロセス名〕

Windows のダンプ出力に関する情報を表示します。

プロセス名：

ダンプ情報を表示する HiRDB のプロセス名を拡張子付きで指定します。長さが 21 バイト以上のプロセス名は指定できません。

ダンプ出力に関する情報の見出しを表示した後，ダンプ出力対象の HiRDB のプロセスの情報を表示します。このオプションで表示する情報を次に示します。

項番	情報の見出し	表示内容	表示する最大長
1	Process※1	ダンプ出力対象の HiRDB のプロセス名	20 バイト※2
2	Folder	ダンプの出力先フォルダ	259 バイト

注※1

プロセス名に"LocalDumps"が表示されたときは，指定したプロセスのダンプはその行に表示したダンプ出力先フォルダに出力されます。

注※2

プロセス名の長さが 20 バイトを超える場合は，21 バイト目以降は表示しません。

＜規則＞

1. プロセス名を省略した場合は，ダンプ出力対象に登録されている実行ファイル（.exe）のダンプ出力情報を表示します。Windows Error Report 機能のダンプ出力対象として登録されているプロセスの情報をすべて表示します。なお，プロセス名に"LocalDumps"が表示されたときは，ダンプ出力情報として出力されたプロセス以外のプロセスのダンプはその行に表示したダンプ出力先フォルダに出力されます。
2. 指定したプロセス名がダンプ出力対象として登録されていない場合は，KFPC00105-I メッセージが出力されます。また，プロセス名を省略した場合でも，ダンプ出力対象にプロセスが一つも登録されていないときは KFPC00105-I メッセージが出力されます。
3. このコマンド実行時にレジストリが更新されていた場合，レジストリ更新中の不完全な値が表示されることがあります。

注

ダンプ出力については，マニュアル「HiRDB システム導入・設計ガイド」の「Windows ダンプ出力」を参照してください。

(k)　-sw サービス開始遅延時間

HiRDB サービスの起動開始を遅延させる場合に指定します。

サービス開始遅延時間：

HiRDB サービスの起動開始を遅延させる秒数を指定します。

秒数は 0～900 の間で指定してください。

次回の HiRDB サービス起動開始から，指定した時間 HiRDB サービスの起動開始が遅延されます。

HiRDB サービスの起動開始を遅延しない場合は，0 を指定してください。

このオプションを設定していない場合，サービス開始遅延時間を 0 秒と仮定します。

(l)　-v {recom｜v0904}

HiRDB ではバージョンごとにシステム定義オペランドの省略値を変更しています。このオプションにオペランド省略時動作を指定します。

recom：推奨モード

v0904：0904 互換モード

＜規則＞

1. オペランド省略時動作は，HiRDB システム内のすべてのユニットで同じ値を指定してください。
2. オペランド省略時動作の変更は，HiRDB をセットアップした直後，又は HiRDB を正常終了した後のオフライン状態で実行してください。正常終了していない場合，HiRDB の起動に失敗することがあります。

(m)　オプション指定なし

現在の設定内容を表示します。

(5)　規則

1. pdntenv コマンドは，オプションごとに実行条件が異なります。実行条件を次に示します。

オプション	実行条件
-hc，-pn，-sc，-ro，-sw	HiRDB の状態に関係なく，実行できます。次回のサービス開始から指定値が有効になります。
-a	HiRDB の状態に関係なく，実行できます。サーバマシン内の最新バージョンの HiRDB で実行してください。
-os，-wd，オプション指定なし	HiRDB の状態に関係なく，実行できます。
-c，-shmfile，-v	HiRDB のサービスが停止しているときだけ実行できます。
-sr	HiRDB の状態に関係なく，実行できます。次回の HiRDB システム開始，又は HiRDB ユニット開始から指定値が有効になります。

2. pdntenv コマンドは，各サーバマシンで実行してください。

(6)　注意事項

1. pdntenv コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。
2. HiRDB の文字コードを変更する場合，変更後にデータベース初期設定ユティリティを実行してデータベースを作り直す必要があります。ただし，次のケースについては作り直しの必要はありません。
   - IVS 非対応 UTF-8 から IVS 対応 UTF-8 へ変更
   - IVS 対応 UTF-8 から IVS 非対応 UTF-8 へ変更

   なお，上記変更時は，ALTER ROUTINE ALL を実行し，関数，手続き，及びトリガの SQL オブジェクトを再作成してください。引数がすべて定数のスカラ関数を指定している関数，手続き，及びトリガの場合，SQL オブジェクトの再作成を行わないと，期待している結果を得られないことがあります。

   また，システム共通定義の pd_substr_length オペランドの値も見直してください。pd_substr_length オペランドについては，マニュアル「HiRDB システム定義」を参照してください。

(7)　出力形式

(a)　オプションなしの場合

```
***** HiRDB setup information *****
Program Name         = HiRDB/SingleServer
Version              = 09-50
Character Code       = sjis
Dat Hard Compress    = on
Setup ID             = ----
Rsh Port Name        = pdrshsrv
Path Name            = C:¥win32app¥hitachi¥hirdb_s
Service Complete     = manual
Rsh Online           = on
ShmFile              = regular
Install Guid         =
Firewall Profile     = public
msgmni               = 50(unit/cmd)
msgtql               = 80(machine/env)
semmax               = 64(unit/cmd)
shmmax               = 4096(unit/cmd)
Service Wait Time    = 60
File Security        = on
Setup Acl            = on
Service Account      = admin
HiRDB Administrator = HiRDBADM
HiRDB Group          = HiRDBGRP
Default Option       = recom
```

[説明]

Program Name：HiRDB のプログラム名

Version：HiRDB のバージョン

Character Code：HiRDB の文字コード種別

sjis：シフト JIS 漢字コード

lang-c：単一バイト文字コード

chinese：EUC 中国語漢字コード

chinese-gb18030：中国語漢字コード（GB18030）

utf-8：Unicode（UTF-8）

utf-8_ivs：Unicode（IVS 対応 UTF-8）

Dat Hard Compress：DAT アクセス時のハードウェア圧縮の指定

on：圧縮

off：非圧縮

Setup ID：セットアップ識別子（標準セットアップ指定でのインストール時は----を表示）

Rsh Port Name：リモート系コマンドのサービスポート名称

Path Name：HiRDB のインストールディレクトリ

Service Complete：HiRDB のサービス開始完了時期

auto：KFPS01001-I メッセージ出力後，HiRDB のサービスが開始完了となります。

manual：pdstart 実行待ち時点で HiRDB のサービスが開始完了となります。

空白：該当しません。

Rsh Online：リモート系コマンドの指定

on：リモート系コマンドを使用します。

off：リモート系コマンドを使用しません。

空白：該当しません。

Shmfile：HiRDB で使用する共用メモリの割り当て先

regular：HiRDB 運用ディレクトリ下のファイルに割り当てます。

page：ページングファイルに割り当てます。

空白：該当しません。

Install Guid：インストール時にインストーラが生成する GUID（グローバル固有 ID）

空白：該当しません。

Firewall Profile：［セキュリティが強化されたファイアウォール］で例外リストに登録したプロファイル

any：すべてのプロファイルが登録されています。

private：プライベートプロファイルが登録されています。

public：パブリックプロファイルが登録されています。

domain：ドメインプロファイルが登録されています。

ただし，複数のプロファイルに対して例外登録されている場合は，組み合わせをコンマで区切って表示します。

（例）private,public,domain

--------：［セキュリティが強化されたファイアウォール］での例外リスト登録が未対応の OS であるか，例外登録していません。

msgmni：メッセージキュー識別子数の設定値（設定範囲／設定方法）※

msgtql：メッセージキューテーブル数の設定値（設定範囲／設定方法）※

semmax：セマフォ識別子数の設定値（設定範囲／設定方法）※

shmmax：共用メモリセグメント数の設定値（設定範囲／設定方法）※

注※　設定範囲及び設定方法の表示項目：

- *******：デフォルト値，又は自動計算した設定値の場合に表示します。
- **unit**：ユニット単位に設定した場合に表示します。
- **machine**：OS 単位に設定した場合に表示します。

- **cmd**：pdntenv コマンドの-sr オプションで設定した場合に表示します。
- **env**：システム環境変数で設定した場合に表示します。
- **auto**：自動計算によって設定した場合に表示します。

Service Wait Time：HiRDB サービス開始を遅延させる時間

空白：設定していません。

File Security：ファイルセキュリティ強化機能の使用有無

on：ファイルセキュリティ強化機能を使用します。

off：ファイルセキュリティ強化機能を使用しません。

空白：対応していません。

Setup Acl：アクセス権限設定機能の使用有無

on：アクセス権限設定機能を使用します。

off：アクセス権限設定機能を使用しません。

空白：対応していません。

Service Account：HiRDB サービスのログオンアカウント設定

local：ローカルシステムアカウントを使用します。

admin：HiRDB 管理者を使用します。

空白：対応していません。

HiRDB Administrator：HiRDB 管理者ユーザ名

空白：対応していません。

HiRDB Group：HiRDB グループ名

空白：対応していません。

Default Option：オペランド省略時動作の設定値

recom：推奨モード

v0904：互換モード

(b)　-a オプションを指定した場合（マルチ HiRDB の場合）

```
***** HiRDB setup information *****
Program Name        = HiRDB/SingleServer
Version             = 09-04-12
Character Code      = sjis
Dat Hard Compress   = on
Setup ID            = UNT1
Rsh Port Name       = pdrshsrvUNT1
Path Name           = C:¥win32app¥hitach¥hirdb_sUNT1
Service Complete    = manual
Rsh Online          = on
ShmFile             =
Install Guid        = {CCBBADFC-91C5-4303-AB47-EEAD6C20C237}
Firewall Profile    = public
msgmni              = ***(unit/auto)
```

```
msgtql              = ***(unit/auto)
semmax              = ***(unit/auto)
shmmax              = ***(unit/auto)
Service Wait Time   =
File Security       =
Setup Acl           =
Service Account     =
HiRDB Administrator =
HiRDB Group         =
Default Option      =
***** HiRDB setup information *****
Program Name        = HiRDB/ParallelServer
Version             = 09-50
Character Code      = sjis
Dat Hard Compress   = on
Setup ID            = UNT2
Rsh Port Name       = pdrshsrvUNT2
Path Name           = C:¥win32app¥hitach¥hirdb_pUNT2
Service Complete    = manual
Rsh Online          = on
ShmFile             = regular
Install Guid        = {84921B0F-B033-412D-9A5F-768B842C08E1}
Firewall Profile    = public
msgmni              = 50(unit/cmd)
msgtql              = 80(machine/env)
semmax              = 64(unit/cmd)
shmmax              = 4096(unit/cmd)
Service Wait Time   = 300
File Security       = on
Setup Acl           = on
Service Account     = admin
HiRDB Administrator = HiRDBADM
HiRDB Group         = HiRDBGRP
Default Option      = recom
```

[説明]

オプションなしの場合を参照してください。

(c) -os オプションを指定した場合

```
*** Operating system information ***
Fixed Shared Memory  : supported
Large Page Size      : 8192
```

[説明]

Fixed Shared Memory：共用メモリのページ固定対応情報

supported：共用メモリのページ固定に対応している Windows です。

unsupport：共用メモリのページ固定に対応していない Windows です。

Large Page Size：Windows の Large Page でのページサイズ（10 進数 20 けた以内）（単位：バイト）

共用メモリのページ固定に対応していない Windows の場合は----を表示します。

(d)　-wd オプションを指定した場合

```
*** HiRDB Dump information ***
Process      Folder
LocalDumps   C:¥Users¥Administrator¥AppData¥Local¥CrashDumps
     :                              :
pdsds.exe    C:¥Users¥Administrator¥AppData¥Local¥CrashDumps¥pdsds.exe
     :                              :
```

［説明］

Process：

プロセス名を示します（最大長 20 バイト）。

プロセス名を指定した場合

指定した HiRDB のプロセス名又は"LocalDumps"

プロセス名を省略した場合

ダンプ出力対象の HiRDB のプロセス名又は"LocalDumps"

Folder：

ダンプ出力フォルダを絶対パスで示します（最大長 259 バイト）。

2.77　pdobils（SQL オブジェクト用バッファの統計情報表示）

2.77.1　pdobils の形式と規則

(1)　機能

SQL オブジェクト用バッファに格納されている SQL オブジェクトの統計情報を表示します。この情報から，処理時間の掛かる SQL や入出力の多い SQL を特定できたり，SQL オブジェクトの再利用状況を把握できたりします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdobils 〔-s サーバ名〕 〔-R｜-r〕 〔-C 〔区切り文字〕 〔-H〕〕 〔-e〕
        〔-U〕 〔-A 〔ACTID〔,ACTID〕…〕〕 〔-NR〕
        〔-N SQLオブジェクト番号〔,SQLオブジェクト番号〕…〕
        〔-O 出力ファイル名〕
```

(4)　オペランド

(a)　-s サーバ名　～＜識別子＞((1～8))

特定サーバの情報を表示する場合，又は特定サーバの統計情報のカウンタを初期化する場合，そのサーバ名（シングルサーバ又はフロントエンドサーバ）を指定します。

(b)　-R ｜-r

統計情報のカウンタを初期化する場合に指定します。例えば，統計情報のカウンタがオーバフローした場合※など，このオプションを指定します。

-R：

SQL オブジェクトバッファ統計情報を出力した後に，カウンタを初期化する場合に指定します。一定間隔で統計情報を繰り返し取得する場合は，このオプションを指定することをお勧めします。

-r：

SQL オブジェクトバッファ統計情報のカウンタの初期化だけを行う場合に指定します。カウンタが初期化される項目については，出力形式を参照してください。

注※

SQL オブジェクト用バッファに格納されている各 SQL オブジェクトのカウンタは，SQL オブジェクトを再利用し続けると，オーバフローすることがあります。カウンタがオーバフローした場合，統計情報の表示項目に*が表示されます。

(c) -C〔区切り文字〕 ～＜文字列＞((1～10))

統計情報を DAT 形式で出力する場合に指定します。

要素を区切って出力したい場合は，区切り文字を指定します。省略した場合，タブ記号が区切り文字になります。なお，区切り文字の先頭にハイフン（-）は指定できません。

(d) -H

-C オプション指定時に，1 行目にタイトル行を出力する場合に指定します。

(e) -e

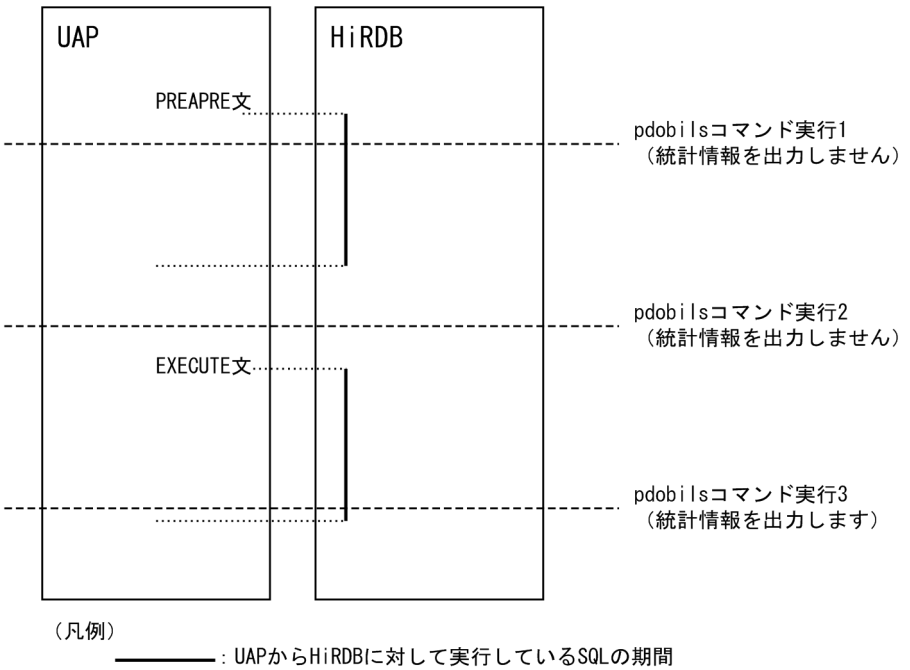
実行回数（出力形式の EXECUTE COUNT の項目）が 1 以上の統計情報だけを出力する場合に指定します。

(f) -U

pdobils コマンド実行時に，実行中の SQL の SQL オブジェクト情報と，その SQL を実行している UAP の情報を出力する場合に指定します。

pdobils コマンド実行時の実行タイミングと出力される統計情報の関係を次の図に示します。

図 2–5 -U オプションを指定した pdobils コマンド実行と，実行中の SQL との関係による統計情報の出力有無



〔説明〕

pdobils コマンドの実行タイミングによって，統計情報の出力有無が変わります。各実行時点での統計情報の出力有無は次のようになります。

pdobils コマンド実行時点	実行している SQL の統計情報出力有無	備考
1	×	SQL 文の前処理が完了し，UAP に結果が返るまでの間にコマンドを実行すると，統計情報が出力される場合があります。
2	×	なし。
3	○	SQL 文実行中に前処理結果が無効※になると，統計情報が出力されない場合があります。

（凡例）

○：実行している SQL の統計情報を出力します。

×：実行している SQL の統計情報を出力しません。

注※

トランザクション決着で無効になります。

(g) -A〔ACTID〔,ACTID〕...〕

ユーザ識別子ごとに使用している次の表に示す状態の SQL オブジェクト情報と，その SQL を実行している UAP の情報を出力する場合に指定します。

状態	出力有無
ACTIVE	○
LRU	△
TEMPORARY	×
COMPILE/TRANSFER	×
PROCESS	×
RELEASE	×

（凡例）

○：出力あり

△：出力あり（コマンド実行中にステータスが ACTIVE から LRU に遷移した場合）

×：出力なし（クライアント環境定義 PDVWOPTMODE 又は PDUAPREPLVL で，SQL オブジェクトの作成を行う指定をした場合など，SQL オブジェクト情報が出力されないステータスがあります。ステータスの詳細については，「一般出力形式」の［説明］の「ステータス」の項目を参照してください。）

ACTID は，特定のユーザ識別子の情報を出力する場合に指定します。-s オプションを指定し，かつマルチフロントエンドサーバを利用している場合は，ACTID に指定したユーザ識別子の存在するフロントエンドサーバのサーバ名を-s オプションに指定してください。なお，指定できるユーザ識別子の数は最大 64 個です。

(h)　-NR

SQL 文中の改行コード（0x0A），及び復帰コード（0x0D）を，空白コード（0x20）に置き換える場合に指定します。

(i)　-N SQL オブジェクト番号〔,SQL オブジェクト番号〕...

SQL オブジェクトに関する保守情報を出力する場合，出力する SQL オブジェクトの SQL オブジェクト番号を指定します※。

SQL オブジェクト番号は，「出力形式」の「6. SQL オブジェクトの番号」の項目に表示される値を指定してください。

ただし，コマンド実行時に指定された SQL オブジェクト番号の SQL オブジェクトがバッファに存在しない場合，その SQL オブジェクトに関する保守情報は出力されません。

注※

マルチフロントエンドサーバを利用している場合は，出力したい SQL オブジェクトが存在するフロントエンドサーバのサーバ名を-s オプションで指定してください。マルチフロントエンドサーバを利用している場合にサーバ名を省略すると，KFPN91001-I メッセージが表示されます。

(j)　-O 出力ファイル名

標準出力，及び標準エラー出力をファイルに出力する場合に指定します。指定した出力ファイルが既に存在している場合は，その出力ファイルに追加書きします。

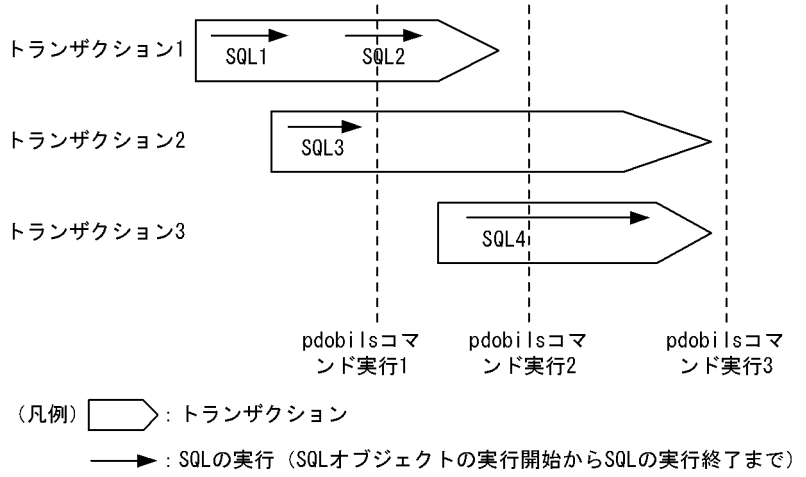
(5)　規則

1. pdobils コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdobils コマンドは，任意のサーバマシンで実行してください。

(6)　注意事項

1. pdobils コマンドの結果は，コマンド実行時のリターンコードで確認できます。リターンコードが 0 の場合は正常終了，4 の場合は異常終了です。
2. pdobils コマンドは，SQL オブジェクト用バッファを操作するため，操作のオーバヘッドによって HiRDB に負荷が掛かることがあります。
3. pdobils コマンドを強制終了した場合，SQL オブジェクト用バッファを操作する SQL の処理，及びプロセス後の処理が最大 1 分待たされます。pdobils コマンドは強制終了しないでください。
4. 統計情報を取得するタイミングは，システム定義の pd_sqlobject_stat_timing オペランドの指定値によって異なります。
5. SQL オブジェクトの実行時間は，トランザクション決着，又は SQL オブジェクトの解放時に，SQL オブジェクト用バッファに反映されます。pdobils コマンド実行時に表示される SQL オブジェクトの実行時間を次の図に示します。

図 2–6　pdobils コマンド実行時に表示される SQL オブジェクトの実行時間



〔説明〕

pdobils コマンドの実行タイミングによって，表示される SQL オブジェクトの実行時間は変わります。各実行時点で表示される時間は次のようになります。

pdobils コマンド実行時点	SQL1	SQL2	SQL3	SQL4
1	×	×	×	×
2	○	○	×	×
3	○	○	○	○

（凡例）

○：表示される実行時間に，該当する SQL の時間が反映されます。

×：表示される実行時間に，該当する SQL の時間は反映されません。

注 1

カーソルを使用した場合の実行時間は，カーソルのオープンからクローズまでにサーバが行った処理時間の合計となります。

注 2

pdobils コマンドの実行中でも，SQL オブジェクト用バッファはほかの UAP などから利用されます。そのため，pdobils コマンド実行中にトランザクションが決着した SQL オブジェクトがある場合，その情報が残っていれば出力情報に反映されます。

注 3

実行時間は，推奨モード又は互換モードによって次のとおり変更となります。

動作モード	実行時間
推奨モード	マイクロ秒単位
v09-04 互換モード	秒単位（秒未満切り捨て）

(7) 出力形式

pdobils コマンドの出力形式は，指定されたオプションの組み合わせによって，次に示す形式で出力されます。

オプションの指定			-C オプションの指定	
-U	-A	-N	なし	あり
なし	なし	なし	一般出力形式	一般出力 DAT 形式
あり	なし	なし	実行中情報出力形式	実行中情報出力 DAT 形式
なし	あり	なし	使用中情報出力形式	使用中情報出力 DAT 形式
なし	なし	あり	保守情報出力形式	－

（凡例）

－：組み合わせはありません。

(a) 一般出力形式

```
pdobils VV-RR-ZZ [70]
<< SQL OBJECT CACHE INFORMATION >> DATE: 2004/01/01 TIME: 00:00:00 [1]
HOST NAME                    : node01 [2]
SERVER NAME                  : fes01 [3]
CACHE SIZE(KB)               : 2048 [4]
SQL OBJECT TOTAL SIZE(B)     : 1932556 [5]

<< SQL OBJECT LIST >>
*SQL OBJECT NO               : 1 [6]
 STATUS                      : ACTIVE [7]
 TYPE                        : STATIC SQL [8]
 SIZE                        : 7792 [9]
 EXECUTE COUNT               : 10 [10]
 # EXECUTE TIME AVG(s)       : 1.234567 [11]
 # EXECUTE TIME MAX(s)       : 9.000000 [12]
 # SERVICE NAME              : [13]
 # UAP NAME                  : Unknown [14]
 # CONNECT NO                : 123 [15]
 # SQL NO                    : 456 [16]
 # RECORD DATE/TIME          : 2004/01/01 00:00:00 [17]
 DB REFERENCE GET COUNT      : 55500 [28] AVG 5550 [29] MAX 5600 [30]
 DB UPDATE GET COUNT         : 0 [31] AVG 0 [32] MAX 0 [33]
 DB READ COUNT               : 800 [34] AVG 80 [35] MAX 85 [36]
 DB WRITE COUNT              : 0 [37] AVG 0 [38] MAX 0 [39]
 LOB REFERENCE GET COUNT     : 0 [40] AVG 0 [41] MAX 0 [42]
 LOB UPDATE GET COUNT        : 0 [43] AVG 0 [44] MAX 0 [45]
 LOB READ COUNT              : 0 [46] AVG 0 [47] MAX 0 [48]
 LOB WRITE COUNT             : 0 [49] AVG 0 [50] MAX 0 [51]
 LIST REFERENCE GET COUNT    : 0 [52] AVG 0 [53] MAX 0 [54]
 LIST UPDATE GET COUNT       : 0 [55] AVG 0 [56] MAX 0 [57]
 LIST READ COUNT             : 0 [58] AVG 0 [59] MAX 0 [60]
 LIST WRITE COUNT            : 0 [61] AVG 0 [62] MAX 0 [63]
 WKFILE READ COUNT           : 0 [64] AVG 0 [65] MAX 0 [66]
 WKFILE WRITE COUNT          : 0 [67] AVG 0 [68] MAX 0 [69]
```

```
 PREPROCESSOR USER          : user1 [18]
 PREPROCESSOR SOURCE        : abc.ec [19]
 SECTION NO                 : 123 [20]
 ISOLATION LEVEL            : 2 [21]
 OPTIMIZE LEVEL             : 10 [22]
 ADDITIONAL OPTIMIZE LEVEL  : 0 [23]
 DEFAULT SCHEMA             : [24]
 ROUTINE ID                 : 123 [25]
 ACCESS TYPE SSCT# n        : user1.ZAIKO (T:0x0002007b) INDEX SCAN (I:0x00030005)[71]
 SQL                        : SELECT GNO,GNAME,KIKAKU,TANKA,SURYO,GENKA FROM ZAIKO [26]

<< SQL OBJECT STATUS COUNT >> [27]
STATUS     | TYPE      STATIC SQL        DYNAMIC SQL       ROUTINE           TOTAL
ACTIVE               :  0                 0                 0                 0
LRU                  :  0                 0                 0                 0
TEMPORARY            :  0                 0                 0                 0
COMPILE/TRANSFER     :  0                 0                 0                 0
PROCESS              :  -                 -                 -                 0
RELEASE              :  -                 -                 -                 0
TOTAL                :  0                 0                 0                 0
```

注 1

出力される情報は，サーバの種別，及びステータスによって異なります。

注 2

6～26，及び 28～69，及び 71 の項目は，SQL オブジェクトの数分繰り返されます。また，タイプ，及びステータスによって，出力される情報が変わります。詳細については，〔注意事項〕を参照してください。

注 3

-s オプションを指定しなかった場合，1～69，及び 71 の情報がシングルサーバ又はフロントエンドサーバの数分繰り返されます。

注 4

-R オプション，又は-r オプションを指定した場合に初期化される項目は，10～17，及び 28～69 です。

〔説明〕

1. 統計情報の表示時刻

   統計情報を表示した時刻が表示されます。

2. ホスト名

   統計情報を取得したホスト名が表示されます。

3. サーバ名

   統計情報を取得したサーバ名が表示されます。

4. SQL オブジェクト用バッファのサイズ

   SQL オブジェクト用バッファのサイズが表示されます（単位：キロバイト）。

5. SQL オブジェクトの合計サイズ

SQL オブジェクト用バッファに格納している SQL オブジェクトの合計サイズが表示されます（単位：バイト）。なお，pdobils コマンド実行中に，SQL オブジェクト用バッファがほかの UAP などから利用された場合，各 SQL オブジェクトのサイズの合計と，SQL オブジェクトの合計サイズが一致しないことがあります。

6. SQL オブジェクトの番号

   SQL オブジェクト用バッファ中の管理番号が表示されます。

7. ステータス

   SQL オブジェクトの状態が表示されます。SQL オブジェクトの状態を次に示します。

状態	説明
ACTIVE	SQL オブジェクトを利用している UAP があることを示します。
LRU	SQL オブジェクト用バッファに SQL オブジェクトが格納されているが，それを使用している UAP がないことを示します。
TEMPORARY	プロセス間で共有しない SQL オブジェクトを示します。次のどちらかに該当する場合，SQL オブジェクトはプロセス間で共有しません。 • クライアント環境定義 PDVWOPTMODE 又は PDUAPREPLVL で，SQL オブジェクトの作成を行う指定をした場合 • 動的 SQL の SQL オブジェクトを，SQL オブジェクト用バッファに格納中である場合
COMPILE/TRANSFER	静的 SQL の SQL オブジェクトを作成中，又はフロントエンドサーバからバックエンドサーバ又はディクショナリサーバに SQL オブジェクトを転送中であることを示します。
PROCESS	SQL オブジェクトをプロセスメモリで管理していることを示します。
RELEASE	SQL オブジェクトを解放中，又は無効になったことを示します。

8. タイプ

   SQL オブジェクトのタイプを表示します。

   STATIC SQL：静的 SQL の SQL オブジェクトです。

   DYNAMIC SQL：動的 SQL の SQL オブジェクトです。

   ROUTINE：ルーチンの SQL オブジェクトです。

9. サイズ

   SQL オブジェクトのサイズが表示されます（単位：バイト）。

10. 実行回数※4

    SQL オブジェクトを実行した回数が表示されます。

11. 平均実行時間

    SQL オブジェクトの平均実行時間が表示されます（単位：秒）。

12. 最大実行時間

    SQL オブジェクトの最大実行時間が表示されます（単位：秒）。

13. サービス名

    OLTP 環境下で UAP を実行した場合，最大実行時間を記録したサービス名が表示されます。

14. UAP 名

    最大実行時間を記録した UAP 名が表示されます。UAP 名が特定できない場合は表示されません。

15. CONNECT 通番※4

    最大実行時間を記録した UAP の CONNECT 通番が表示されます。

16. SQL 番号※4

    最大実行時間を記録した SQL 番号が表示されます。

17. 記録した時刻

    最大実行時間を記録した時刻が表示されます。

18. プリプロセスしたユーザの認可識別子

    UAP のソースプログラムのプリプロセスを実行したユーザの認可識別子が表示されます。なお，この項目は，静的 SQL の SQL オブジェクトの場合に表示されます。

19. プリプロセスしたソースプログラム名

    プリプロセスをしたソースプログラム名が表示されます。なお，この項目は，静的 SQL の SQL オブジェクトの場合に表示されます。

20. セクション番号

    セクション番号が表示されます。なお，この項目は，静的 SQL の SQL オブジェクトの場合に表示されます。

21. データ保証レベル

    データ保証レベルが表示されます。なお，この項目は，静的 SQL，又は動的 SQL の SQL オブジェクトの場合に表示されます。

22. SQL 最適化オプション

    SQL 最適化オプションが表示されます。なお，この項目は，静的 SQL，又は動的 SQL の SQL オブジェクトの場合に表示されます。

23. SQL 拡張最適化オプション

    SQL 拡張最適化オプションが表示されます。なお，この項目は，静的 SQL，又は動的 SQL の SQL オブジェクトの場合に表示されます。

24. デフォルトスキーマ名

    SQL 文で認可識別子を省略した場合，仮定される認可識別子が表示されます。SQL 文で認可識別子を指定している場合は表示されません。なお，この項目は，動的 SQL の SQL オブジェクトの場合に表示されます。

25. ルーチンのオブジェクト ID

    ルーチンのオブジェクト ID が表示されます。なお，この項目は，ルーチンの SQL オブジェクトの場合に表示されます。

26. SQL 文

    SQL 文が表示されます。なお，この項目は，静的 SQL，又は動的 SQL の SQL オブジェクトの場合に表示されます。

27. タイプとステータスごとの SQL オブジェクト数

    SQL オブジェクト用バッファで管理している SQL オブジェクトの，タイプとステータスごとの数を表示します。-e オプション指定時は，EXECUTE COUNT が 1 以上の SQL オブジェクトだけを集計対象とします。

28. データページ，インデクスページ，及びディレクトリページの参照回数（合計）※２※４

    SQL オブジェクトの利用によって，グローバルバッファに対して参照のためのバッファ取得を要求した合計回数です。

29. データページ,インデクスページ,及びディレクトリページの参照回数（平均）※１※２※４

    SQL オブジェクトの利用によって，グローバルバッファに対して参照のためのバッファ取得を要求した，SQL の 1 回当たりの平均回数です。

30. データページ,インデクスページ,及びディレクトリページの参照回数（最大）※２※４

    SQL オブジェクトの利用によって，グローバルバッファに対して参照のためのバッファ取得を要求した，SQL の 1 回当たりの最大回数です。

31. データページ,インデクスページ,及びディレクトリページの更新回数（合計）※２※４

    SQL オブジェクトの利用によって，グローバルバッファに対して更新のためのバッファ取得を要求した合計回数です。

32. データページ,インデクスページ,及びディレクトリページの更新回数（平均）※１※２※４

    SQL オブジェクトの利用によって，グローバルバッファに対して更新のためのバッファ取得を要求した，SQL の 1 回当たりの平均回数です。

33. データページ,インデクスページ,及びディレクトリページの更新回数（最大）※２※４

    SQL オブジェクトの利用によって，グローバルバッファに対して更新のためのバッファ取得を要求した，SQL の 1 回当たりの最大回数です。

34. データページ,インデクスページ,及びディレクトリページの実 READ 回数（合計）※２※３※４

    SQL オブジェクトの利用によって，データページ,インデクスページ,及びディレクトリページに対して入力した回数の合計です。この回数には，プリフェッチ機能で入力した回数は含まれますが，非同期 READ 機能で入力した回数は含まれません。

35. データページ,インデクスページ,及びディレクトリページの実 READ 回数（平均）※１※２※３※４

    SQL オブジェクトの利用によって，データページ,インデクスページ,及びディレクトリページに対して入力した回数の，SQL の 1 回当たりの平均回数です。この回数には，プリフェッチ機能で入力した回数は含まれますが，非同期 READ 機能で入力した回数は含まれません。

36. データページ,インデクスページ,及びディレクトリページの実 READ 回数（最大）※２※３※４

    SQL オブジェクトの利用によって，データページ,インデクスページ,及びディレクトリページに対して入力した回数の，SQL の 1 回当たりの最大回数です。この回数には，プリフェッチ機能で入力した回数は含まれますが，非同期 READ 機能で入力した回数は含まれません。

37. データページ,インデクスページ,及びディレクトリページの実 WRITE 回数（合計）※２※３※４

SQL オブジェクトの利用によって，データページ,インデクスページ,及びディレクトリページに対して出力した回数の合計です。この回数には，デファードライト処理で出力した回数は含まれません。

38. データページ,インデクスページ,及びディレクトリページの実 WRITE 回数(平均)※1※2※3※4

    SQL オブジェクトの利用によって，データページ,インデクスページ,及びディレクトリページに対して出力した回数の，SQL の 1 回当たりの平均回数です。この回数には，デファードライト処理で出力した回数は含まれません。

39. データページ,インデクスページ,及びディレクトリページの実 WRITE 回数（最大）※2※3※4

    SQL オブジェクトの利用によって，データページ,インデクスページ,及びディレクトリページに対して出力した回数の，SQL の 1 回当たりの最大回数です。この回数には，デファードライト処理で出力した回数は含まれません。

40. LOB 列のデータページの参照回数（合計）※4

    SQL オブジェクトの利用によって，LOB 列のデータページを参照するためにバッファ取得を要求した回数の合計です。

41. LOB 列のデータページの参照回数（平均）※1※4

    SQL オブジェクトの利用によって，LOB 列のデータページを参照するためにバッファ取得を要求した，SQL の 1 回当たりの平均回数です。

42. LOB 列のデータページの参照回数（最大）※4

    SQL オブジェクトの利用によって，LOB 列のデータページを参照するためにバッファ取得を要求した，SQL の 1 回当たりの最大回数です。

43. LOB 列のデータページの更新回数（合計）※4

    SQL オブジェクトの利用によって，LOB 列のデータページを更新するためにバッファ取得を要求した回数の合計です。

44. LOB 列のデータページの更新回数（平均）※1※4

    SQL オブジェクトの利用によって，LOB 列のデータページを更新するためにバッファ取得を要求した，SQL の 1 回当たりの平均回数です。

45. LOB 列のデータページの更新回数（最大）※4

    SQL オブジェクトの利用によって，LOB 列のデータページを更新するためにバッファ取得を要求した，SQL の 1 回当たりの最大回数です。

46. LOB 列のデータページの実 READ 回数（合計）※4

    SQL オブジェクトの利用によって，LOB 列のデータページに対して発生した実 READ 回数の合計です。

47. LOB 列のデータページの実 READ 回数（平均）※1※4

    SQL オブジェクトの利用によって，LOB 列のデータページに対して発生した実 READ 回数の，SQL の 1 回当たりの平均値です。

48. LOB 列のデータページの実 READ 回数（最大）※4

SQL オブジェクトの利用によって，LOB 列のデータページに対して発生した実 READ 回数の，SQL の 1 回当たりの最大値です。

49. LOB 列のデータページの実 WRITE 回数（合計）※4

    SQL オブジェクトの利用によって，LOB 列のデータページに対して発生した実 WRITE 回数の合計です。

50. LOB 列のデータページの実 WRITE 回数（平均）※1※4

    SQL オブジェクトの利用によって，LOB 列のデータページに対して発生した実 WRITE 回数の，SQL の 1 回当たりの平均値です。

51. LOB 列のデータページの実 WRITE 回数（最大）※4

    SQL オブジェクトの利用によって，LOB 列のデータページに対して発生した実 WRITE 回数の，SQL の 1 回当たりの最大値です。

52. リストページの参照回数（合計）※4

    SQL オブジェクトの利用によって，グローバルバッファに対してリストページ参照のためのバッファ取得を要求した合計回数です。

53. リストページの参照回数（平均）※1※4

    SQL オブジェクトの利用によって，グローバルバッファに対してリストページ参照のためのバッファ取得を要求した，SQL の 1 回当たりの平均回数です。

54. リストページの参照回数（最大）※4

    SQL オブジェクトの利用によって，グローバルバッファに対してリストページ参照のためのバッファ取得を要求した，SQL の 1 回当たりの最大回数です。

55. リストページの更新回数（合計）※4

    SQL オブジェクトの利用によって，グローバルバッファに対してリストページ更新のためのバッファ取得を要求した合計回数です。

56. リストページの更新回数（平均）※1※4

    SQL オブジェクトの利用によって，グローバルバッファに対してリストページ更新のためのバッファ取得を要求した，SQL の 1 回当たりの平均回数です。

57. リストページの更新回数（最大）※4

    SQL オブジェクトの利用によって，グローバルバッファに対してリストページ更新のためのバッファ取得を要求した，SQL の 1 回当たりの最大回数です。

58. リストページの実 READ 回数（合計）※4

    SQL オブジェクトの利用によって，リストページに対して発生した実 READ 回数の合計です。

59. リストページの実 READ 回数（平均）※1※4

    SQL オブジェクトの利用によって，リストページに対して発生した実 READ 回数の，SQL の 1 回当たりの平均値です。

60. リストページの実 READ 回数（最大）※4

SQL オブジェクトの利用によって，リストページに対して発生した実 READ 回数の，SQL の 1 回当たりの最大値です。

61. リストページの実 WRITE 回数（合計）※4

    SQL オブジェクトの利用によって，リストページに対して発生した実 WRITE 回数の合計です。

62. リストページの実 WRITE 回数（平均）※1※4

    SQL オブジェクトの利用によって，リストページに対して発生した実 WRITE 回数の，SQL の 1 回当たりの平均値です。

63. リストページの実 WRITE 回数（最大）※4

    SQL オブジェクトの利用によって，リストページに対して発生した実 WRITE 回数の，SQL の 1 回当たりの最大値です。

64. 作業表用ファイルの READ 回数（合計）※4

    SQL オブジェクトの利用によって，作業表用ファイルに対して発生した READ 回数の合計です。

65. 作業表用ファイルの READ 回数（平均）※1※4

    SQL オブジェクトの利用によって，作業表用ファイルに対して発生した READ 回数の，SQL の 1 回当たりの平均値です。

66. 作業表用ファイルの READ 回数（最大）※4

    SQL オブジェクトの利用によって，作業表用ファイルに対して発生した READ 回数の，SQL の 1 回当たりの最大値です。

67. 作業表用ファイルの WRITE 回数（合計）※4

    SQL オブジェクトの利用によって，作業表用ファイルに対して発生した WRITE 回数の合計です。

68. 作業表用ファイルの WRITE 回数（平均）※1※4

    SQL オブジェクトの利用によって，作業表用ファイルに対して発生した WRITE 回数の，SQL の 1 回当たりの平均値です。

69. 作業表用ファイルの WRITE 回数（最大）※4

    SQL オブジェクトの利用によって，作業表用ファイルに対して発生した WRITE 回数の，SQL の 1 回当たりの最大値です。

70. HiRDB のバージョン

    ZZ がない場合は VV-RR だけを表示します。

71. アクセス表と検索方法

    ACCESS TYPE SSCT# n : aa...a (T:bb...b) cc...c (I:dd...d)

    n：システム固有情報を表示します。

    aa...a：アクセスする表の表名を表示します。ただし，ビュー表の基表の場合は「*.*」を表示します。

    bb...b：アクセスする表の ID を表示します。

    cc...c：アクセスする表の検索方法を表示します。検索方法については「検索方法の種類」を参照してください。

dd...d：インデクスの ID を表示します。ただし，cc...c が「TABLE SCAN」の場合は(I:dd...d)を表示しません。

注※1

平均値の小数点以下は切り捨てられます。ただし，合計がオーバフローした場合，平均値の箇所に「****」を表示します。

注※2

表やインデクスなどの定義情報が格納されているディクショナリ表へのアクセス回数は含まれません。

注※3

デファードライト処理とは，グローバルバッファ上で更新されたページを COMMIT 文が発行されてもディスクに書き込まないで，更新ページ数がある一定の値に達した時点でディスクに書き込む処理のことです。デファードライト処理によって，DB 処理プロセスと非同期に HiRDB ファイルへ出力されます。

プリフェッチ機能とは，DB 処理プロセスがディスクボリューム上の表データのページを，複数ページ一括して読み込む機能です。

非同期 READ 機能とは，DB 処理プロセスと非同期にディスクボリューム上の表データのページを，複数ページ一括して読み込む機能です。

注※4

カウンタがオーバフローすることがあります。オーバフローしたカウンタには，先頭に「*」が表示されます。オーバフローした場合でも，カウンタはカウントし続けます。

〔注意事項〕

タイプとステータスの組み合わせによって，6～26 の項目は表示される場合と表示されない場合とがあります。タイプとステータスの組み合わせによる各項目の表示有無を次の表に示します。なお，番号は出力形式の番号と対応しています。

表 2–13　タイプとステータスの組み合わせによる各項目の表示有無（静的 SQL の場合）

番号	タイプ																	
	STATIC SQL						DYNAMIC SQL						ROUTINE					
	ステータス						ステータス						ステータス					
	A	L	T	C	P	R	A	L	T	C	P	R	A	L	T	C	P	R
6	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
7	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
8	○	○	－	－	－	－	○	○	－	－	－	－	○	○	－	－	－	－
9	○	○	－	－	－	－	○	○	－	－	－	－	○	○	－	－	－	－
10	○	○	－	－	－	－	○	○	－	－	－	－	○	○	－	－	－	－
11	△	△	－	－	－	－	△	△	－	－	－	－	△	△	－	－	－	－
12	△	△	－	－	－	－	△	△	－	－	－	－	△	△	－	－	－	－

番号	タイプ																	
	STATIC SQL						DYNAMIC SQL						ROUTINE					
	ステータス						ステータス						ステータス					
	A	L	T	C	P	R	A	L	T	C	P	R	A	L	T	C	P	R
13	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
14	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
15	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
16	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
17	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
18	○	○	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
19	○	○	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
20	○	○	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−	−
21	○	○	−	−	−	−	○	○	−	−	−	−	×	×	−	−	−	−
22	○	○	−	−	−	−	○	○	−	−	−	−	×	×	−	−	−	−
23	○	○	−	−	−	−	○	○	−	−	−	−	×	×	−	−	−	−
24	×	×	−	−	−	−	○	○	−	−	−	−	−	−	−	−	−	−
25	−	−	−	−	−	−	−	−	−	−	−	−	○	○	−	−	−	−
26	○	○	−	−	−	−	○	○	−	−	−	−	−	−	−	−	−	−
28	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
29	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
30	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
31	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
32	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
33	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
34	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
35	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
36	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
37	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
38	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
39	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
40	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
41	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−

番号	タイプ																	
	STATIC SQL						DYNAMIC SQL						ROUTINE					
	ステータス						ステータス						ステータス					
	A	L	T	C	P	R	A	L	T	C	P	R	A	L	T	C	P	R
42	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
43	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
44	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
45	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
46	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
47	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
48	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
49	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
50	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
51	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
52	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
53	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
54	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
55	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
56	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
57	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
58	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
59	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
60	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
61	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
62	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
63	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
64	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
65	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
66	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
67	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
68	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−
69	△	△	−	−	−	−	△	△	−	−	−	−	△	△	−	−	−	−

番号	タイプ																	
	STATIC SQL						DYNAMIC SQL						ROUTINE					
	ステータス						ステータス						ステータス					
	A	L	T	C	P	R	A	L	T	C	P	R	A	L	T	C	P	R
71	○	○	−	−	−	−	○	○	−	−	−	−	−	−	−	−	−	−

（凡例）

○：表示されます。

△：番号 10 の項目が 1 以上の場合に表示されます。0 の場合，該当する情報は表示されません。

×：表示されません。

−：該当しません。

A：ACTIVE

L：LRU

T：TEMPORARY

C：COMPILE/TRANSFER

P：PROCESS

R：RELEASE

(b)　実行中情報出力形式

```
pdobils VV-RR-ZZ [70]
<< SQL OBJECT CACHE INFORMATION >> DATE: 2004/01/01 TIME: 00:00:00 [1]
HOST NAME                    : node01 [2]
SERVER NAME                  : fes01 [3]
CACHE SIZE(KB)               : 2048 [4]
SQL OBJECT TOTAL SIZE(B)     : 1932556 [5]

<< UAP INFORMATION >> [72]
 PROCESS ID                  : 12345678 [73]
 TRANSACTION ID(GID/BID)     : aa...a/bb...b [74]
 SERVICE NAME                : [75]
 UAP NAME                    : pdsql [76]
 CONNECT NO                  : 234 [77]

<< CURRENT SQL OBJECT >>
*SQL OBJECT NO               : 1 [6]
 STATUS                      : ACTIVE [7]
 TYPE                        : STATIC SQL [8]
 SIZE                        : 7792 [9]
 EXECUTE COUNT               : 10 [10]
 # EXECUTE TIME AVG(s)       : 1.234567 [11]
 # EXECUTE TIME MAX(s)       : 9.000000 [12]
 # SERVICE NAME              : [13]
 # UAP NAME                  : Unknown [14]
 # CONNECT NO                : 123 [15]
 # SQL NO                    : 456 [16]
 # RECORD DATE/TIME          : 2004/01/01 00:00:00 [17]
```

```
 DB REFERENCE GET COUNT     : 55500 [28] AVG 5550 [29] MAX 5600 [30]
 DB UPDATE GET COUNT        : 0 [31] AVG 0 [32] MAX 0 [33]
 DB READ COUNT              : 800 [34] AVG 80 [35] MAX 85 [36]
 DB WRITE COUNT             : 0 [37] AVG 0 [38] MAX 0 [39]
 LOB REFERENCE GET COUNT    : 0 [40] AVG 0 [41] MAX 0 [42]
 LOB UPDATE GET COUNT       : 0 [43] AVG 0 [44] MAX 0 [45]
 LOB READ COUNT             : 0 [46] AVG 0 [47] MAX 0 [48]
 LOB WRITE COUNT            : 0 [49] AVG 0 [50] MAX 0 [51]
 LIST REFERENCE GET COUNT   : 0 [52] AVG 0 [53] MAX 0 [54]
 LIST UPDATE GET COUNT      : 0 [55] AVG 0 [56] MAX 0 [57]
 LIST READ COUNT            : 0 [58] AVG 0 [59] MAX 0 [60]
 LIST WRITE COUNT           : 0 [61] AVG 0 [62] MAX 0 [63]
 WKFILE READ COUNT          : 0 [64] AVG 0 [65] MAX 0 [66]
 WKFILE WRITE COUNT         : 0 [67] AVG 0 [68] MAX 0 [69]
 PREPROCESSOR USER          : user1 [18]
 PREPROCESSOR SOURCE        : abc.ec [19]
 SECTION NO                 : 123 [20]
 ISOLATION LEVEL            : 2 [21]
 OPTIMIZE LEVEL             : 10 [22]
 ADDITIONAL OPTIMIZE LEVEL  : 0 [23]
 DEFAULT SCHEMA             : [24]
 ROUTINE ID                 : 123 [25]
 ACCESS TYPE SSCT# n        : user1.ZAIKO (T:0x0002007b) INDEX SCAN (I:0x00030005)[71]
 SQL                        : SELECT GNO,GNAME,KIKAKU,TANKA,SURYO,GENKA FROM ZAIKO [26]

<< SQL OBJECT STATUS COUNT >> [27]
STATUS     | TYPE      STATIC SQL        DYNAMIC SQL       ROUTINE           TOTAL
ACTIVE               :  0                 0                 0                 0
LRU                  :  0                 0                 0                 0
TEMPORARY            :  0                 0                 0                 0
COMPILE/TRANSFER     :  0                 0                 0                 0
PROCESS              :  -                 -                 -                 0
RELEASE              :  -                 -                 -                 0
TOTAL                :  0                 0                 0                 0
```

〔説明〕

1〜71 の項目は一般出力形式と同じです。

72. SQL を実行している UAP 情報

    SQL を実行している UAP の情報を表示します。複数の UAP が SQL を実行している場合は，UAP と SQL オブジェクトの情報（6〜26，及び 71〜77 の項目）を繰り返し表示します。

73. SQL を実行している UAP が接続しているサーバのプロセス ID

    SQL を実行している UAP が接続しているフロントエンドサーバ又はシングルサーバのプロセス ID を表示します。

74. SQL を実行している UAP のトランザクション ID

    TRANSACTION ID(GID/BID) : aa...a/bb...b

    SQL を実行している UAP のトランザクション ID を表示します。

    aa...a：トランザクションのグローバル識別子

    bb...b：トランザクションのブランチ識別子

75. SQL を実行している UAP のサービス名

    SQL が OLTP 環境下で実行されている場合に UAP のサービス名を表示します。

76. SQL を実行している UAP 名

    SQL を実行している UAP 名を表示します。

    UAP 名が特定できない場合，UAP 名を表示しません。

77. SQL を実行している UAP のコネクト通番

    SQL を実行している UAP のコネクト通番を表示します。

(c) 使用中情報出力形式

```
pdobils VV-RR-ZZ [70]
<< SQL OBJECT CACHE INFORMATION >> DATE: 2004/01/01 TIME: 00:00:00 [1]
HOST NAME                   : node01 [2]
SERVER NAME                 : fes01 [3]
CACHE SIZE(KB)              : 2048 [4]
SQL OBJECT TOTAL SIZE(B)    : 1932556 [5]

<< UAP INFORMATION >> [72]
 PROCESS ID                 : 12345678 [73]
 TRANSACTION ID(GID/BID)    : aa...a/bb...b [74]
 SERVICE NAME               : [75]
 UAP NAME                   : pdsql [76]
 CONNECT NO                 : 234 [77]
 ACTID                      : 1-1-4 [79]


<< USING SQL OBJECT LIST >>
*SQL OBJECT NO              : 1 [6]
 STATUS                     : ACTIVE [7]
 TYPE                       : STATIC SQL [8]
 SIZE                       : 7792 [9]
 EXECUTE COUNT              : 10 [10]
 # EXECUTE TIME AVG(s)      : 1.234567 [11]
 # EXECUTE TIME MAX(s)      : 9.000000 [12]
 # SERVICE NAME             : [13]
 # UAP NAME                 : Unknown [14]
 # CONNECT NO               : 123 [15]
 # SQL NO                   : 456 [16]
 # RECORD DATE/TIME         : 2004/01/01 00:00:00 [17]
 DB REFERENCE GET COUNT     : 55500 [28] AVG 5550 [29] MAX 5600 [30]
 DB UPDATE GET COUNT        : 0 [31] AVG 0 [32] MAX 0 [33]
 DB READ COUNT              : 800 [34] AVG 80 [35] MAX 85 [36]
 DB WRITE COUNT             : 0 [37] AVG 0 [38] MAX 0 [39]
 LOB REFERENCE GET COUNT    : 0 [40] AVG 0 [41] MAX 0 [42]
 LOB UPDATE GET COUNT       : 0 [43] AVG 0 [44] MAX 0 [45]
 LOB READ COUNT             : 0 [46] AVG 0 [47] MAX 0 [48]
 LOB WRITE COUNT            : 0 [49] AVG 0 [50] MAX 0 [51]
 LIST REFERENCE GET COUNT   : 0 [52] AVG 0 [53] MAX 0 [54]
 LIST UPDATE GET COUNT      : 0 [55] AVG 0 [56] MAX 0 [57]
 LIST READ COUNT            : 0 [58] AVG 0 [59] MAX 0 [60]
 LIST WRITE COUNT           : 0 [61] AVG 0 [62] MAX 0 [63]
 WKFILE READ COUNT          : 0 [64] AVG 0 [65] MAX 0 [66]
 WKFILE WRITE COUNT         : 0 [67] AVG 0 [68] MAX 0 [69]
```

```
 PREPROCESSOR USER          : user1 [18]
 PREPROCESSOR SOURCE        : abc.ec [19]
 SECTION NO                 : 123 [20]
 ISOLATION LEVEL            : 2 [21]
 OPTIMIZE LEVEL             : 10 [22]
 ADDITIONAL OPTIMIZE LEVEL  : 0 [23]
 DEFAULT SCHEMA             : [24]
 ROUTINE ID                 : 123 [25]
 ACCESS TYPE SSCT# n        : user1.ZAIKO (T:0x0002007b) INDEX SCAN (I:0x00030005)[71]
 SQL                        : SELECT GNO,GNAME,KIKAKU,TANKA,SURYO,GENKA FROM ZAIKO [26]

<< SQL OBJECT STATUS COUNT >> [27]
STATUS     | TYPE      STATIC SQL        DYNAMIC SQL       ROUTINE           TOTAL
ACTIVE               :  0                 0                 0                 0
LRU                  :  0                 0                 0                 0
TEMPORARY            :  0                 0                 0                 0
COMPILE/TRANSFER     :  0                 0                 0                 0
PROCESS              :  -                 -                 -                 0
RELEASE              :  -                 -                 -                 0
TOTAL                :  0                 0                 0                 0
```

〔説明〕

1〜71 の項目は一般出力形式と同じです。

72. SQL オブジェクトを使用している UAP 情報

    SQL オブジェクトを使用している UAP の情報を表示します。複数の UAP が SQL オブジェクトを使用している場合は，UAP と SQL オブジェクトの情報（6〜26，71〜77，及び 79 の項目）を繰り返し表示します。

73. SQL オブジェクトを使用している UAP が接続しているサーバのプロセス ID

    SQL オブジェクトを使用している UAP が接続しているフロントエンドサーバ，又はシングルサーバのプロセス ID を表示します。

74. SQL オブジェクトを使用している UAP のトランザクション ID

    TRANSACTION ID(GID/BID) : aa...a/bb...b

    SQL オブジェクトを使用している UAP のトランザクション ID を表示します。

    aa...a：トランザクションのグローバル識別子

    bb...b：トランザクションのブランチ識別子

75. SQL オブジェクトを使用している UAP のサービス名

    SQL オブジェクトが OLTP 環境下で使用されている場合に UAP のサービス名を表示します。

76. SQL オブジェクトを使用している UAP 名

    SQL オブジェクトを使用している UAP 名を表示します。

77. SQL オブジェクトを使用している UAP のコネクト通番

    SQL オブジェクトを使用している UAP のコネクト通番を表示します。

79. SQL オブジェクトを使用しているユーザ識別子

    SQL オブジェクトを使用しているユーザ識別子を表示します。

(d)　保守情報出力形式

```
pdobils VV-RR-ZZ [70]
<< SQL OBJECT CACHE INFORMATION >> DATE: 2004/01/01 TIME: 00:00:00 [1]
HOST NAME                    : node01 [2]
SERVER NAME                  : fes01 [3]
CACHE SIZE(KB)               : 2048 [4]
SQL OBJECT TOTAL SIZE(B)     : 1932556 [5]

<< SQL OBJECT #N 12345/67890 >> [78]※
0001: abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789xx
0002: abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789xx
```

注※

指定された SQL オブジェクト番号の数分，繰り返し出力します。

指定された SQL オブジェクト番号の SQL オブジェクトが存在しない場合は，表示されません。

〔説明〕

4〜5，及び 70 の項目は，一般出力形式と同じです。

1. SQL オブジェクトバッファ統計情報表示時間

   保守情報を表示した時間を表示します。

2. ホスト名

   保守情報を取得したホスト名を表示します。

3. サーバ名

   保守情報を取得したサーバ名を表示します。

78. SQL オブジェクトの番号とサイズ

   SQL OBJECT #N aa...a/bb...b

   N：SQL オブジェクト用バッファ中の管理番号を表示します。

   aa...a：保守情報のサイズをバイト単位で表示します。

   bb...b：SQL オブジェクトのサイズをバイト単位で表示します。

(e)　一般出力 DAT 形式

一般出力 DAT 形式で出力する項目を，先頭から区切り文字列で区切って標準出力に出力します。

出力形式の SQL には，SQL 文に含まれている TAB（X'09'），NL（X'0a'），CR（X'0d'）などを，そのまま出力します。また，行末には改行文字（0x0d0a）が入ります。

一般出力 DAT 形式で出力する項目を次の表に示します。

表 2–14　一般出力 DAT 形式で出力する項目

出力する項目名	内容
DATE	[1]の DATE 部分

出力する項目名	内容
TIME	[1]の TIME 部分
HOST NAME	[2]
SERVER NAME	[3]
CACHE SIZE(KB)	[4]
SQL OBJECT TOTAL SIZE(B)	[5]
SQL OBJECT NO	[6]
STATUS	[7]
TYPE	[8]
SIZE	[9]
EXECUTE COUNT	[10]
EXECUTE TIME AVG(s)	[11]
EXECUTE TIME MAX(s)	[12]
SERVICE NAME	[13]
UAP NAME	[14]
CONNECT NO	[15]
SQL NO	[16]
RECORD DATE	[17]の DATE 部分
RECORD TIME	[17]の TIME 部分
DB REFERENCE GET COUNT	[28]
DB REFERENCE GET COUNT(AVG)	[29]
DB REFERENCE GET COUNT(MAX)	[30]
DB UPDATE GET COUNT	[31]
DB UPDATE GET COUNT(AVG)	[32]
DB UPDATE GET COUNT(MAX)	[33]
DB READ COUNT	[34]
DB READ COUNT(AVG)	[35]
DB READ COUNT(MAX)	[36]
DB WRITE COUNT	[37]
DB WRITE COUNT(AVG)	[38]
DB WRITE COUNT(MAX)	[39]
LOB REFERENCE GET COUNT	[40]

出力する項目名	内容
LOB REFERENCE GET COUNT(AVG)	[41]
LOB REFERENCE GET COUNT(MAX)	[42]
LOB UPDATE GET COUNT	[43]
LOB UPDATE GET COUNT(AVG)	[44]
LOB UPDATE GET COUNT(MAX)	[45]
LOB READ COUNT	[46]
LOB READ COUNT(AVG)	[47]
LOB READ COUNT(MAX)	[48]
LOB WRITE COUNT	[49]
LOB WRITE COUNT(AVG)	[50]
LOB WRITE COUNT(MAX)	[51]
LIST REFERENCE GET COUNT	[52]
LIST REFERENCE GET COUNT(AVG)	[53]
LIST REFERENCE GET COUNT(MAX)	[54]
LIST UPDATE GET COUNT	[55]
LIST UPDATE GET COUNT(AVG)	[56]
LIST UPDATE GET COUNT(MAX)	[57]
LIST READ COUNT	[58]
LIST READ COUNT(AVG)	[59]
LIST READ COUNT(MAX)	[60]
LIST WRITE COUNT	[61]
LIST WRITE COUNT(AVG)	[62]
LIST WRITE COUNT(MAX)	[63]
WKFILE READ COUNT	[64]
WKFILE READ COUNT(AVG)	[65]
WKFILE READ COUNT(MAX)	[66]
WKFILE WRITE COUNT	[67]
WKFILE WRITE COUNT(AVG)	[68]
WKFILE WRITE COUNT(MAX)	[69]
PREPROCESSOR USER	[18]
PREPROCESSOR SOURCE	[19]

出力する項目名	内容
SECTION NO	[20]
ISOLATION LEVEL	[21]
OPTIMIZE LEVEL	[22]
ADDITIONAL OPTIMIZE LEVEL	[23]
DEFAULT SCHEMA	[24]
ROUTINE ID	[25]
ACCESS TYPE	[71]
SQL	[26]

(f)　実行中情報出力 DAT 形式

実行中情報出力 DAT 形式で出力する項目を，先頭から区切り文字列で区切って標準出力に出力します。

出力形式の SQL には，SQL 文に含まれている TAB（X'09'），NL（X'0a'），CR（X'0d'）などを，そのまま出力します。また，行末には改行文字（0x0d0a）が入ります。

実行中情報出力 DAT 形式で出力する項目を次の表に示します。

表 2–15　実行中情報出力 DAT 形式で出力する項目

出力する項目名	内容
DATE	[1]の DATE 部分
TIME	[1]の TIME 部分
HOST NAME	[2]
SERVER NAME	[3]
CACHE SIZE(KB)	[4]
SQL OBJECT TOTAL SIZE(B)	[5]
PROCESS ID	[73]
TRANSACTION ID(GID/BID)	[74]
SERVICE NAME	[75]
UAP NAME	[76]
CONNECT NO	[77]
SQL OBJECT NO	[6]
STATUS	[7]
TYPE	[8]

出力する項目名	内容
SIZE	[9]
EXECUTE COUNT	[10]
EXECUTE TIME AVG(s)	[11]
EXECUTE TIME MAX(s)	[12]
SERVICE NAME	[13]
UAP NAME	[14]
CONNECT NO	[15]
SQL NO	[16]
RECORD DATE	[17]の DATE 部分
RECORD TIME	[17]の TIME 部分
DB REFERENCE GET COUNT	[28]
DB REFERENCE GET COUNT(AVG)	[29]
DB REFERENCE GET COUNT(MAX)	[30]
DB UPDATE GET COUNT	[31]
DB UPDATE GET COUNT(AVG)	[32]
DB UPDATE GET COUNT(MAX)	[33]
DB READ COUNT	[34]
DB READ COUNT(AVG)	[35]
DB READ COUNT(MAX)	[36]
DB WRITE COUNT	[37]
DB WRITE COUNT(AVG)	[38]
DB WRITE COUNT(MAX)	[39]
LOB REFERENCE GET COUNT	[40]
LOB REFERENCE GET COUNT(AVG)	[41]
LOB REFERENCE GET COUNT(MAX)	[42]
LOB UPDATE GET COUNT	[43]
LOB UPDATE GET COUNT(AVG)	[44]
LOB UPDATE GET COUNT(MAX)	[45]
LOB READ COUNT	[46]
LOB READ COUNT(AVG)	[47]
LOB READ COUNT(MAX)	[48]

出力する項目名	内容
LOB WRITE COUNT	[49]
LOB WRITE COUNT(AVG)	[50]
LOB WRITE COUNT(MAX)	[51]
LIST REFERENCE GET COUNT	[52]
LIST REFERENCE GET COUNT(AVG)	[53]
LIST REFERENCE GET COUNT(MAX)	[54]
LIST UPDATE GET COUNT	[55]
LIST UPDATE GET COUNT(AVG)	[56]
LIST UPDATE GET COUNT(MAX)	[57]
LIST READ COUNT	[58]
LIST READ COUNT(AVG)	[59]
LIST READ COUNT(MAX)	[60]
LIST WRITE COUNT	[61]
LIST WRITE COUNT(AVG)	[62]
LIST WRITE COUNT(MAX)	[63]
WKFILE READ COUNT	[64]
WKFILE READ COUNT(AVG)	[65]
WKFILE READ COUNT(MAX)	[66]
WKFILE WRITE COUNT	[67]
WKFILE WRITE COUNT(AVG)	[68]
WKFILE WRITE COUNT(MAX)	[69]
PREPROCESSOR USER	[18]
PREPROCESSOR SOURCE	[19]
SECTION NO	[20]
ISOLATION LEVEL	[21]
OPTIMIZE LEVEL	[22]
ADDITIONAL OPTIMIZE LEVEL	[23]
DEFAULT SCHEMA	[24]
ROUTINE ID	[25]
ACCESS TYPE	[71]
SQL	[26]

(g)　使用中情報出力 DAT 形式

使用中情報出力 DAT 形式で出力する項目を，先頭から区切り文字列で区切って標準出力に出力します。

出力形式の SQL には，SQL 文に含まれている TAB（X'09'），NL（X'0a'），CR（X'0d'）などを，そのまま出力します。また，行末には改行文字（0x0d0a）が入ります。

使用中情報出力 DAT 形式で出力する項目を次の表に示します。

表 2–16　使用中情報出力 DAT 形式で出力する項目

出力する項目名	内容
DATE	[1]の DATE 部分
TIME	[1]の TIME 部分
HOST NAME	[2]
SERVER NAME	[3]
CACHE SIZE(KB)	[4]
SQL OBJECT TOTAL SIZE(B)	[5]
PROCESS ID	[73]
TRANSACTION ID(GID/BID)	[74]
SERVICE NAME	[75]
UAP NAME	[76]
CONNECT NO	[77]
ACTID	[79]
SQL OBJECT NO	[6]
STATUS	[7]
TYPE	[8]
SIZE	[9]
EXECUTE COUNT	[10]
EXECUTE TIME AVG(s)	[11]
EXECUTE TIME MAX(s)	[12]
SERVICE NAME	[13]
UAP NAME	[14]
CONNECT NO	[15]
SQL NO	[16]
RECORD DATE	[17]の DATE 部分

出力する項目名	内容
RECORD TIME	[17]の TIME 部分
DB REFERENCE GET COUNT	[28]
DB REFERENCE GET COUNT(AVG)	[29]
DB REFERENCE GET COUNT(MAX)	[30]
DB UPDATE GET COUNT	[31]
DB UPDATE GET COUNT(AVG)	[32]
DB UPDATE GET COUNT(MAX)	[33]
DB READ COUNT	[34]
DB READ COUNT(AVG)	[35]
DB READ COUNT(MAX)	[36]
DB WRITE COUNT	[37]
DB WRITE COUNT(AVG)	[38]
DB WRITE COUNT(MAX)	[39]
LOB REFERENCE GET COUNT	[40]
LOB REFERENCE GET COUNT(AVG)	[41]
LOB REFERENCE GET COUNT(MAX)	[42]
LOB UPDATE GET COUNT	[43]
LOB UPDATE GET COUNT(AVG)	[44]
LOB UPDATE GET COUNT(MAX)	[45]
LOB READ COUNT	[46]
LOB READ COUNT(AVG)	[47]
LOB READ COUNT(MAX)	[48]
LOB WRITE COUNT	[49]
LOB WRITE COUNT(AVG)	[50]
LOB WRITE COUNT(MAX)	[51]
LIST REFERENCE GET COUNT	[52]
LIST REFERENCE GET COUNT(AVG)	[53]
LIST REFERENCE GET COUNT(MAX)	[54]
LIST UPDATE GET COUNT	[55]
LIST UPDATE GET COUNT(AVG)	[56]
LIST UPDATE GET COUNT(MAX)	[57]

出力する項目名	内容
LIST READ COUNT	[58]
LIST READ COUNT(AVG)	[59]
LIST READ COUNT(MAX)	[60]
LIST WRITE COUNT	[61]
LIST WRITE COUNT(AVG)	[62]
LIST WRITE COUNT(MAX)	[63]
WKFILE READ COUNT	[64]
WKFILE READ COUNT(AVG)	[65]
WKFILE READ COUNT(MAX)	[66]
WKFILE WRITE COUNT	[67]
WKFILE WRITE COUNT(AVG)	[68]
WKFILE WRITE COUNT(MAX)	[69]
PREPROCESSOR USER	[18]
PREPROCESSOR SOURCE	[19]
SECTION NO	[20]
ISOLATION LEVEL	[21]
OPTIMIZE LEVEL	[22]
ADDITIONAL OPTIMIZE LEVEL	[23]
DEFAULT SCHEMA	[24]
ROUTINE ID	[25]
ACCESS TYPE	[71]
SQL	[26]

(8)　統計情報の見方

統計情報の見方について説明します。

- 番号 27 の PROCESS について

  STATUS が PROCESS の TOTAL 数がカウントされている場合，SQL オブジェクト用バッファに SQL オブジェクトが格納できないため，プロセスごとに SQL オブジェクトを管理していることを示します。

  常にカウントされている場合は，SQL オブジェクト用バッファを再度見積もり直すことをお勧めします。
- 番号 27 の DYNAMIC について
- その 1

STATUS が TEMPORARY の STATIC SQL 数がカウントされている場合，クライント環境定義 PDUAPREPLVL が設定されている UAP を実行していることを示します。

PDUAPREPLVL が不要な UAP の場合，PDUAPREPLVL の指定を削除して UAP を実行することをお勧めします。

- その 2

  STATUS が TEMPORARY の DYNAMIC SQL 数が常にカウントされている場合，クライント環境定義 PDUAPREPLVL，及び PDVWOPTMODE が設定されている UAP を実行していることを示します。

  PDUAPREPLVL，及び PDVWOPTMODE が不要な UAP の場合，PDUAPREPLVL，及び PDVWOPTMODE の指定を削除して UAP を実行することをお勧めします。

- 実行中 SQL 文の特定について

  -U オプションを指定することで，<<CURRENT SQL OBJECT>>の「SQL」から，実行している SQL を特定できます。-U オプションを指定しない場合でも，SQL オブジェクト用バッファが十分にある場合，次の手順で統計情報を参照すると，実行中の SQL 文を特定できます。なお，実行中の SQL 文を特定する場合，PDUAPREPLVL 及び PDVWOPTMODE を指定しないで，再利用ありを指定する必要があります。

  1. STATUS が ACTIVE の SQL オブジェクトを特定します。
  2. 1.の SQL オブジェクトの SQL を参照します。なお，静的 SQL の場合は，更に SQL を実行している UAP も特定できます。SQL オブジェクトの PREPROCESSOR USE，及び PREPROCESSOR SOURCE を参照し，そのプリプロセスをしたソースを使用している UAP が，SQL 文を実行している UAP となります。

- チューニングの対象となる SQL について

  次の項目の値を参照し，値が大きい SQL 文がチューニング対象となります。

  - SQL オブジェクトの実行回数
  - SQL オブジェクトの平均実行時間
  - SQL オブジェクトの最大実行時間
  - データページ，インデクスページ，及びディレクトリページの参照回数
  - データページ，インデクスページ，及びディレクトリページの更新回数
  - データページ，インデクスページ，及びディレクトリページの実 READ 回数
  - データページ，インデクスページ，及びディレクトリページの実 WRITE 回数
  - 作業表用ファイルの READ 回数
  - 作業表用ファイルの WRITE 回数

  アクセス表と検索方法を確認し，意図した検索方法で検索されていない場合は，SQL チューニングを行うことを検討してください。

  チューニングの詳細については，マニュアル「HiRDB システム運用ガイド」の実行時間が長い SQL のチューニングを参照してください。

- ユーザ識別子ごとの SQL 文の特定方法について

-A オプションを指定することで，<<USING SQL OBJECT LIST>>の「SQL」から使用している SQL を特定できます。

また，定期的に-A オプションを指定して使用中 SQL オブジェクトの情報を取得しておくことで，デッドロック発生時に出力されるデッドロック情報のユーザ識別子と突き合わせて，SQL 文を特定できる場合があります。

2.78 pdobjconv（64 ビットモードの HiRDB への SQL オブジェクトの移行）

2.78.1 pdobjconv の形式と規則

(1) 機能

HiRDB を 32 ビットモードから 64 ビットモードへ移行するときに，pdobjconv コマンドを実行します。このコマンドを実行すると，ディクショナリ中の次に示す情報が 64 ビットモード対応になります。このコマンドを実行しないと，64 ビットモードに移行後，ビュー表，手続き，及び関数が使用できなくなります。

- ビュー解析情報
- 手続きの SQL オブジェクト
- 関数の SQL オブジェクト
- トリガの SQL オブジェクト

このコマンドは，32 ビットモードの HiRDB から 64 ビットモードの HiRDB へバージョンアップするときに（pdvrup コマンド実行後に）HiRDB が自動的に起動します。したがって，通常時は実行することはほとんどありません。

(2) 実行者

DBA 権限を持っているユーザが実行できます。

(3) 形式

```
pdobjconv
```

(4) オプション

なし。

(5) 規則

1. pdobjconv コマンドは，HiRDB が稼働中のときに実行できます。
2. pdobjconv コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdobjconv コマンドを実行するには，システム用 RD エリアがオープン状態であり，かつ閉塞解除状態である必要があります。

(6) 注意事項

1. ユーザが pdobjconv コマンドを実行した場合は，カレントディレクトリ下に SQL オブジェクト移行情報が作成されます。したがって，pdobjconv コマンドの実行者にファイルの作成権限が必要になります。
2. pdobjconv コマンドは，同時に複数実行しないでください。
3. 次の条件をどちらも満たす場合，pdobjconv コマンド実行前後で，ユーザが定義したビュー表の検索結果が異なることがあります。
   - 移行前に，ビュー表を次の順序で定義している。

     1.ユーザが関数を定義する。

     2.1.で定義した関数を指定してビュー表を定義する。

     3.1.と同一のユーザが，同一名称で，かつ引数の数が一致する関数を定義する。
   - 上記の条件を満たすビュー表を定義した後，pdobjconv コマンドを実行している。
4. pdobjconv コマンドの結果は，次のコマンド実行時のリターンコードで確認できます。

   0：

   pdobjconv コマンドが正常終了しました。

   4：

   警告レベルのエラーはありますが，pdobjconv コマンドを正常終了します。

   8：

   一部の SQL オブジェクトの移行に失敗しました。メッセージ又は SQL オブジェクト移行情報を参照して，エラーとなった要因を調査して取り除いてください。

   又は，pdobjconv コマンド実行上のエラーが発生しました。

   12：

   pdobjconv コマンドが異常終了しました。メッセージ又は SQL オブジェクト移行情報を参照して，エラーとなった要因を調査して取り除いてください。

   pdcancel コマンドで pdobjconv コマンドをキャンセルしたり，又は pdobjconv コマンドのプロセスで異常が発生したりすると，リターンコードが 12 になります。

(7) 出力形式

pdobjconv コマンドを実行すると，実行結果（SQL オブジェクト移行情報）が次に示すファイルに出力されます。

- HiRDB が自動的に pdobjconv コマンドを起動したとき

  $PDDIR/spool/pdobjconv.log
- ユーザが pdobjconv コマンドを起動したとき

  カレントディレクトリ下の pdobjconv.log

SQL オブジェクト移行情報の出力例を次に示します。

```
   << SQL OBJECT Converter >>                [15]START TIME : YYYY/MM/DD HH:MM:SS

[1]VIEW DEF COUNT               : 1
[2]PROCEDURE COUNT              : 12
[3]FUNCTION COUNT               : 1
[4]TRIGGER COUNT                : 1
 [5]            [6]      [7]      [8]
*VIEW*         INVALID : USER01.USER_VIEW01
KFPX21103-I...

 [5]            [6]      [7]      [8]
*PROCEDURE*    VALID   : MASTER.REGISTRY_DELETE_CONTEXT  (REGISTRY_DELETE_CONTEXT)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_DELETE_KEY  (REGISTRY_DELETE_KEY)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_GET_CONTEXT  (REGISTRY_GET_CONTEXT)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_GET_LVALUE  (REGISTRY_GET_LVALUE)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_GET_SVALUE  (REGISTRY_GET_SVALUE)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_GET_VALUE_SIZE  (REGISTRY_GET_VALUE_SIZE)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_NEW_CONTEXT  (REGISTRY_NEW_CONTEXT)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_RESET_LVALUE  (REGISTRY_RESET_LVALUE)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_RESET_SVALUE  (REGISTRY_RESET_SVALUE)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_SET_LVALUE  (REGISTRY_SET_LVALUE)
KFPX21102-I...

*PROCEDURE*    VALID   : MASTER.REGISTRY_SET_SVALUE  (REGISTRY_SET_SVALUE)
KFPX21102-I...

*PROCEDURE*    INVALID : USER01.USER_PROC01  (USER_PROC01)
KFPX21103-I...
   [5]            [6]      [7]      [8]                  [9]
*FUNCTION*     INVALID : USER01.USER_FUNC01  (FUSER_FUNC010000000028)
KFPX21103-I...

   [5]            [6]      [7]      [8]
*TRIGGER*      INVALID : USER01.(TRIG2002080112300000) ((TRIG2002080112300000))
KFPX21103-I...

   << RESULT REPORT >>                      [16]ENDED TIME : 1999/08/13 10:46:17
      [10]          [11]          [12]         [13]          [14]
  = RESOURCE == = SUCCESS === = FAILURE === = PASS ====== = TOTAL =====
```

```
*VIEW*                    1           0            0            1
*PROCEDURE*               1           0           11           12
*FUNCTION*                1           0            0            1
*TOTAL*                   3           0           11           14
```

〔説明〕

1. ビュー表の数
2. 手続きの数
3. 関数の数
4. トリガの数
5. SQL オブジェクトの識別文字列
   VIEW：ビュー表
   PROCEDURE：手続き
   FUNCTION：関数
   TRIGGER：トリガ
6. SQL オブジェクトの状態
   INVALID：無効
   VALID：有効
7. 認可識別子
8. ビュー表名，手続き名，又は関数名
9. 特定名
10. SQL オブジェクトの識別文字列
    VIEW：ビュー表
    PROCEDURE：手続き
    FUNCTION：関数
    TRIGGER：トリガ
11. 正常に再登録された SQL オブジェクトの数
12. 再登録に失敗した SQL オブジェクトの数
13. 再登録をしなかった SQL オブジェクトの数
14. SQL オブジェクトの合計
15. pdobjconv コマンド開始時間
16. pdobjconv コマンド終了時間

2.79　pdopen（RD エリアのオープン）

2.79.1　pdopen の形式と規則

(1)　機能

指定した RD エリアをオープンします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdopen -r　{RDエリア名〔, RDエリア名〕…|ALL}　〔-W 実行監視時間〕
```

(4)　オプション

(a)　-r　{RD エリア名〔，RD エリア名〕...｜ ALL}

RD エリア名　～＜識別子＞((1～30))

オープンする RD エリア名を指定します。

ALL

マスタディレクトリ用 RD エリアを除くすべての RD エリアをオープンします。

＜規則＞

1. RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。
2. 既にオープンしている RD エリアについては，再度オープンできません。
3. ユーザ用 RD エリア又はユーザ LOB 用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，かつ pdhold コマンドによる閉塞状態
4. データディクショナリ用 RD エリア，ユーザ用 RD エリア，及びユーザ LOB 用 RD エリアを同時に指定する場合は，データディクショナリ用 RD エリア名を最後に指定してください。

(b)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdopen コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pdopen コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdopen コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdopen コマンドは，RD エリアに対してモード EX で排他制御をします。そのため，指定した RD エリアにほかのトランザクションがアクセスしている場合は，そのトランザクションの終了まで pdopen コマンドは待ち状態となります。
4. 共用 RD エリアに対して pdopen コマンドを実行した場合，全バックエンドサーバに排他を掛けます。このため，該当する RD エリアへのアクセスを複数同時に実行している場合，グローバルデッドロックが発生し，タイムアウトする可能性があります。グローバルデッドロックが発生した場合，再度 pdopen コマンドを実行してください。

(6)　注意事項

1. pdopen コマンドの結果は，pddbls コマンドで確認できます。
2. pdopen コマンドのリターンコードと基本的な対処方法を次に示します。

   0 以外の場合，一部の RD エリアがオープンしていないことがあります。その状態では業務を継続できないおそれがあるため，再度 pdopen コマンドでオープンしてください。

   0：正常終了

   4：警告終了

   一部の RD エリアの処理がエラー終了しました。

   8：異常終了

   一部の RD エリアの処理がエラー終了していることがあります。

   12：異常終了

   エラーメッセージを表示できない事象が発生しました。

   一部の RD エリアの処理がエラー終了していることがあります。シングルサーバ又はディクショナリサーバがあるホストのイベントログ中のエラーメッセージを参照して，エラーの要因を取り除いた後，再度コマンドを実行してください。イベントログ中にエラーメッセージが出力されていない場合は保守員に連絡してください。

2.80　pdparaload（パラレルローディング）

2.80.1　pdparaload の形式と規則

(1)　機能

一つの入力データファイルから横分割表を構成する複数の RD エリアに対して，データロード（pdload コマンド）を並列実行します（パラレルローディング機能）。

(2)　実行者

pdload コマンドと同じ実行権限を持つユーザが実行できます。

(3)　形式

```
pdparaload 〔pdloadコマンドオプション〕 〔-I 実行間隔〕 〔認可識別子.〕表識別子
           pdparaload制御文ファイル名
```

(4)　オプション

(a)　pdload コマンドオプション

pdload コマンドのオプションを指定します。指定されたコマンドのオプションは，pdparaload コマンドが pdload コマンドを実行するときに使用します。

pdload コマンドのオプションの中で pdparaload コマンドに指定できないオプションがあります。pdparaload コマンドでの pdload コマンドオプションの指定可否を次に示します。

なお，指定できないオプションを指定すると，データロード実行中にエラー終了します。

表 2–17　pdload コマンドオプションの指定可否

項番	オプション	pdparaload への指定可否	説明
1	-d	○	pdload コマンドを作成モードで実行する場合に指定します。 なお，表に非横分割インデクスが定義されているときは，表の既存データ削除時に非横分割インデクスの排他を取得します。表の分割数や既存データ量によっては削除処理に時間が掛かり，排他待ちで pdload がエラー終了するおそれがあるため，以下の対処を推奨します。 • -I オプションを指定して，RD エリアごとの pdload の実行間隔を空けて実行する

項番	オプション	pdparaload への指定可否	説明
			• 事前に PURGE TABLE 文で該当表のデータを削除した後に，-d オプションを指定しないで pdparaload を実行する
2	-a ｜-b ｜-U	○ （-w all は×）	入力データファイルの形式を指定します。 なお，-w all（表移行用アンロードファイルを使用し，表，及びインデクス定義を伴うデータロード指定）は指定できません。
3	-i	○	インデクス作成方法を指定します。
4	-l	○	pdload コマンド実行時のデータベースの更新ログの取得方法を指定します。
5	-k	○	LOB データを LOB 列に格納する場合，及び抽象データ型列に格納する値を生成するコンストラクタ関数の引数が LOB パラメタのときに，そのデータの入力方法を指定します。 pdparaload コマンドでは，-k f オプションでのファイルの作成方法を推奨します。 このオプションを指定すると，一つの LOB データごとにファイルを作成するため，並列実行時に LOB データを重複して読み込むことがありません。そのため，入出力処理を削減できます。ただし，このオプションを指定すると，各 RD エリア単位でのデータロード中に，LOB 中間ファイルが作成されます。作成された LOB 中間ファイルは，pdparaload コマンド終了後も削除されないため，pdparaload コマンド終了後にユーザが削除する必要があります。
6	-c	○	列構成情報ファイルの名称を指定します。
7	-v	○	ナル値・関数情報ファイルの名称を指定します。
8	-n	○	ローカルバッファを使用するときにバッファ面数を指定します。
9	-u	×	pdload コマンドを実行するユーザの認可識別子は指定できません。
10	-x	○	入力データがクラスタキーの昇順又は降順になっているかどうかをチェックしない場合に指定します。
11	-s	○	入力データファイルが DAT 形式の場合，各データ間のセパレータ文字を変更するときに指定します。
12	-e	○	入力データの不正を検出した場合に，処理を中断するときに指定します。
13	-r	○	入力データファイルの 1 行目からデータを入力しないで，指定した行からデータを入力する場合に指定します。
14	-z	○	可変長文字列，可変長各国文字列，及び可変長混在文字列に対して，長さ 0 のデータを格納する場合に指定します。

項番	オプション	pdparaload への指定可否	説明
15	-y	○	データロード時，未使用ページを使い切った場合に，使用中ページの未使用領域へデータを格納したいときに指定します。
16	-o	×	インデクス一括作成が正常終了した後，index 文に指定したインデクス情報ファイルを自動的に削除できません。
17	-m	○	処理の実行状況を出力するメッセージの出力間隔を指定します。
18	-X	○	障害を検知するために，コマンドで実施されるディクショナリ操作についての通信の応答時間を指定します。
19	-K	○	XML 型のパラメタに対する入力データ値の形式を指定します。
20	-G	○	入力データファイルとして指定した XML データのデータ種別（XML 文書又は ESIS-B 形式）を指定します。
21	-F	○	FLOAT 型及び SMALLFLT 型の入力データ値が OS の制限値に満たない場合，又は OS の制限値を超えている場合に，入力データ値を OS の制限値内に補正するときに指定します。

（凡例）

○：pdparaload コマンドに指定できます。

×：pdparaload コマンドに指定できません。

(b)　-I 実行間隔　～＜符号なし整数＞((10～600000))《1000》（単位：ミリ秒）

pdparaload コマンドは，RD エリア単位でデータロードを行います。このオプションでは，各データロードの実行間隔を指定します。

実行間隔を空けることで，複数の pdload コマンドが同時に実行されないため，ディクショナリ表にアクセスが集中し，実行待ちになることを防止できます。

＜指定値の目安＞

通常は，デフォルト値で運用します。

pdload の前処理時間が指定値を超える場合は，1000 ずつ加算して pdparaload コマンドを再実行してください。

(c)　〔認可識別子.〕表識別子　～＜識別子＞((1～30 バイト))

pdparaload コマンドを使用して，データロードする表の表識別子を指定します。指定時の規則は，pdload コマンドと同じです。pdload コマンドの〔認可識別子.〕表識別子については，「〔認可識別子.〕表識別子」を参照してください。

(d)　pdparaload 制御文ファイル名　～＜パス名＞((1～1023 バイト))

pdparaload 制御文ファイルのパスを指定します。pdparaload 制御文ファイルとは，pdparaload コマンドによって実行される，pdload コマンドの制御文を記述したファイルです。pdload コマンドの制御文については，「コマンドの形式」を参照してください。ただし，pdload コマンドの制御文の中で pdparaload 制御文ファイルに指定できない制御文があります。pdparaload 制御文ファイルでの pdload コマンドの制御文の指定可否を次に示します。

なお，指定できない制御文を指定すると，pdparaload は制御文をチェックしないため，データロード実行時に制御文エラーとなります。

表 2–18　pdload コマンド制御文の指定可否

項番	制御文	オプション	pdparaload での指定可否	説明
1	mtguide	−	×	テープ装置は使用できません。
2	emtdef	−	×	
3	source	RD エリア名	×	RD エリア名は，pdparaload コマンドが指定するため，ユーザは指定できません。
4		サーバ名｜ホスト名	HiRDB/パラレルサーバの場合：◎ HiRDB/シングルサーバの場合：△	HiRDB/パラレルサーバの場合，必ず指定してください。指定しないでデータロードを実行すると，各 RD エリアが存在するサーバ上のファイルを入力データファイルとして処理してしまいます。
5		入力データファイル名	◎	入力データが入っている入力データファイルの名称を絶対パス名で指定します。
6		(uoc)	△	UOC を使用する場合，入力データファイルの入出力を UOC 側でするときに指定します。
7		error	△	エラー情報を出力するファイルの名称を絶対パス名で指定します。 なお，データロードの実行時に pdparaload コマンドがファイル名に RD エリア名，_RD エリア ID，""を付加します。そのため，指定するパス名の長さに注意してください。詳細については，注意事項の 4.を参照してください。
8		errdata	△	入力データがエラーとなった場合，そのエラーとなった行データを出力するときに指定します。 なお，データロードの実行時に pdparaload コマンドがファイル名に RD エリア名，_RD エリア ID，""を付加します。そのため，指定するパス名の長さに注意してください。詳細については，注意事項の 4.を参照してください。
9		errwork	△	errdata オペランド指定時に，エラーデータファイル作成用のバッファサイズをキロバイト単位で指定します。

項番	制御文	オプション	pdparaload での指定可否	説明
10		maxreclen	△	入力データファイルが DAT 形式，拡張 DAT 形式，バイナリ形式，及び pdrorg 出力バイナリ形式の場合に入力データ長を指定します。
11	index	−	×	インデクス情報ファイルの情報は指定できません。
12	idxwork	サーバ名	△	インデクス情報ファイルを作成するサーバの名称を指定します。
13		ディレクトリ名	○	インデクス情報ファイルを作成するディレクトリの名称を絶対パス名で指定します。
14	sort	サーバ名	△	ソート用ワークファイルを作成するサーバの名称を指定します。
15		ディレクトリ名	○	ソート用ワークファイルを作成するディレクトリの名称を，絶対パス名で指定します。
16	lobdata	LOB 入力ファイル名	×※	LOB 列がある表へデータロードする場合，又はコンストラクタ関数への入力パラメタとして LOB データを入力する場合に，LOB の情報を指定します。 対象表に BLOB 列，又は BLOB パラメタがある抽象データ型列を定義し，-k オプションに f，又は v を指定する場合は，必ず lobdata 文を指定してください。指定しないと，基表だけがデータロードされ，BLOB データはデータロードされません。
17		LOB 入力ファイルのディレクトリ名	△※	
18	lobcolumn	−	×	列単位 LOB 入力ファイルは使用できません。
19	lobmid	−	×	LOB 中間ファイルは指定できません。 LOB 中間ファイルは，次のファイル名で作業用ディレクトリ下に作成します。 ¥作業用ディレクトリ¥LOBMID-xxxxxxxxx xxxxxxxxx は，各実行プロセスの実行時間，及びプロセス ID を含む名称です。各 RD エリア単位にファイルを作成します。
20	srcuoc	−	△	UOC を利用して，データを編集してからデータベースに格納する場合，UOC の情報を指定します。
21	array	−	△	繰返し列を含む表の場合，入力データファイルに記述する配列データ形式とナル値の扱いを指定します。
22	extdat	−	△	DAT 形式の入力データファイルで，拡張機能を使用する場合に指定します。
23	src_work	−	×	分割入力データファイルは作成できません。
24	constraint	−	△	検査保留状態に関する設定をします。

項番	制御文	オプション	pdparaload での指定可否	説明
25	option	spacelvl	△	入力データに対して，空白変換をするかどうかを指定します。
26		tblfree	△	データロード時に，CREATE TABLE で指定した空き領域比率（PCTFREE の指定値）を変更してデータを格納したい場合に指定します。
27		idxfree	△	インデクス一括作成モードでインデクスを作成する場合に，CREATE INDEX で指定した空き領域比率（PCTFREE の指定値）を変更してインデクスを作成したいときに指定します。
28		job	×	同期点指定のデータロードはできません。
29		cutdtmsg	△	DAT 形式の入力データファイルのデータロード時に切り捨てが発生した場合，エラー情報ファイルに警告メッセージを出力するかどうかを指定します。
30		nowait	×	NOWAIT 検索はできません。
31		bloblimit	△	pdrorg 出力バイナリ形式の入力データファイルでデータ変換をしてデータロードする場合に，データを保持する領域の大きさを指定します。
32		exectime	△	pdload の実行時間を監視する場合に，その監視時間を分単位で指定します。
33		null_string	△	DEFAULT 句を指定している表に対してデータロードする場合，入力データがナル値（"*"又は記述しない場合）のときに，DEFAULT 句の既定値を格納するか，又はナル値を格納するかを指定します。
34		dataerr	△	入力データエラー（論理エラー）を検知した場合，データ格納処理を無効（ロールバック）にするときに指定します。 RD エリア単位でデータロードを行うため，非分割キーインデクスがあり，-i s オプション（インデクス更新モード）以外の指定がある場合は，各データロード実行時に制御文エラーとなります。
35		lengover	△	DAT 形式（拡張 DAT 形式を含む）の入力データファイルの，データ型が CHAR，VARCHAR，NCHAR，NVARCHAR，MCHAR，MVARCHAR，又は BINARY の列に格納する入力データ長が列定義長より長い場合，入力データエラーとして扱うときに指定します。
36		divermsg	×	divermsg は，pdparaload コマンドが指定するため，ユーザは指定できません。

項番	制御文	オプション	pdparaload での指定可否	説明
37		srcendian	△	エンディアンが異なるプラットフォーム間で pdrorg 出力バイナリ形式ファイルを使用してデータを移行する場合に指定します。
38		allspace	△	固定長データ形式の入力データファイルを使用してデータロードする場合，数値データ型（INTEGER，SMALLINT，DECIMAL，FLOAT，SMALLFLT）の列に格納する文字データを「0」と変換して格納するときに指定します。
39		whitespace	△	XML 型の列に対して XML 文書をデータロードする場合に，XML 文書に含まれる空白類の取り扱い方法を指定します。
40		seq_range	△	自動採番機能を使用してデータロードする場合，順序番号の取得方式を指定します。
41		file_buff_size	△	入力バッファに入力データファイルのデータを読み込む場合に，使用する入力バッファ用のメモリサイズを指定します。
42		charset	△	UTF-8 の環境で定義した表に対して，UTF-16 の入力データファイルでデータロードする場合，入力データファイルのエンディアンを指定します。

（凡例）

◎：必ず指定してください。

○：制御文を指定した場合は，指定してください。

△：指定は任意です。

×：指定できません。

−：該当しません。

注※

対象表に BLOB 列，又は BLOB パラメタがある抽象データ型列を定義し，-k オプションに f，又は v を指定する場合は，lobdata 文を指定する必要があります。詳細を次の表に示します。

項番	-k オプション指定	BLOB 列	BLOB 型の入力パラメタを持つ抽象データ型の列
1	d	×	×
2	f	○	○
3	v	○	×

（凡例）

○：lobdata 文が必要です。

×：lobdata 文は必要ありません。

(5) 規則

1. pdparaload コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdparaload コマンドでデータロードできる表は，横分割表だけです。

   ただし，フレキシブルハッシュ分割を定義した表は対象外とします（マトリクス分割にフレキシブルハッシュ分割を定義している場合も含む）。フレキシブルハッシュ分割を定義した表を指定した場合，pdparaload コマンドはエラー終了します。
3. 横分割表を構成する RD エリアが，データベース作成ユティリティ（pdload）を実行できる状態である必要があります。RD エリアの状態については，「RD エリアの状態によるユティリティ及び UAP の実行可否」を参照してください。
4. pdparaload コマンドを起動するには，RD エリア単位でデータロードを実行するために使用される排他資源数を合計した数分の排他資源が必要です。RD エリア単位データロードが必要とする排他資源数については，「注意事項」を参照してください。
5. 同じ表には，pdparaload コマンドを同時に実行できません。異なる表であれば，pdparaload コマンドを同時に実行することができます。ただし，各表を構成する RD エリアに同じ RD エリアがないことが前提となります。pdparaload コマンドは，内部で RD エリア単位にデータロードを実行するため，RD エリア単位に排他を掛けています。同じ RD エリアに格納した表に同時に pdparaload コマンドを実行した場合，データロードが排他エラーで終了します。

   RD エリア単位のデータロードの排他制御モードについては，「ユティリティの排他制御モード」を参照してください。

(6) 注意事項

1. pdparaload コマンドのリターンコードを次に示します。

   0：正常終了

   4：正常終了（一部の入力データに論理エラーが発生したため，データベースへの格納処理をスキップしました）

   8：異常終了
2. pdparaload コマンドは，次の規則で pdload 制御文ファイル名を生成します。

   LOD_CTL_認可識別子_表識別子_RD エリア ID

   上記と同じ名称のファイルが既に存在している場合は，エラー終了します。

   なお，認可識別子，表識別子が「¥」を含む場合は，これらは「(」（左括弧）に置き換えて生成されます。
3. 次の制御文から pdparaload コマンドが生成するファイル名と同じ名称のファイルが既に存在している場合は，既存のファイルを上書きします。
   - source 文の error で指定したエラー情報ファイル名
   - source 文の errdata で指定したエラーデータファイル名
4. pdparaload コマンドは，source 文で指定するファイル名に RD エリア名，_RD エリア ID，""を付加します。そのため，指定するパス名，及びファイル名の長さは，次のように指定してください。

- source 文の場合

  source 文の error，及び errdata オペランドに指定するパス名，及びファイル名の長さは，次の計算式の値以下にしてください。

  ＜ error オペランドの場合＞

  ・パス名の長さ（単位：バイト）

  OS 上のパス名の上限値−（RD エリア名称長＋ 9）

  ・ファイル名の長さ（単位：バイト）

  OS 上のファイル名の上限値−（RD エリア名称長＋ 9）

  ＜ errdata オペランドの場合＞

  ・パス名の長さ（単位：バイト）

  （OS 上のパス名の上限値−8）−（RD エリア名称長＋ 9）

  ・ファイル名の長さ（単位：バイト）

  （OS 上のファイル名の上限値−8）−（RD エリア名称長＋ 9）

  また，1 行の source 文の長さが 1023 バイトに収まるように指定する必要があります。1023 バイトを超えた場合はエラー終了します。

  source 文で付加される RD エリア名と""の規則を次に示します。

  なお，RD エリア名が「¥」を含む場合は，これらは「(」（左括弧）に置き換えて生成されます。

```
[RDエリア名の付加前]
  source△入力データファイル名△error=エラー情報ファイル名△errdata=エラー情報ファイル名

[RDエリア名の付加後]
  source△"RDエリア名"△入力データファイル名△
  error="エラー情報ファイル名_RDエリア名_RDエリアID"△
  errdata="エラー情報ファイル名_RDエリア名_RDエリアID"
```

（凡例）△：半角空白

5. pdparaload コマンドで分割キー構成列に対して順序数生成子を指定した実行はできません。実行した場合は処理を続行します。その結果，入力データファイルのデータがデータベースに格納できない場合があります。

2.81　pdpfresh（サーバプロセスのリフレッシュ）

2.81.1　pdpfresh の形式と規則

(1)　機能

サーバプロセスをリフレッシュします。リフレッシュとは，稼働中のプロセスを終了させ，代わりに新しいプロセスを起動することをいいます。

リフレッシュ対象となるのは，シングルサーバ，バックエンドサーバ，及びディクショナリサーバです。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdpfresh 〔-s サーバ名〕 〔 {-f | -c} 〕
```

(4)　オプション

(a)　-s サーバ名　～＜識別子＞((1～8))

リフレッシュをするサーバの名称を指定します。

HiRDB/パラレルサーバの場合は，バックエンドサーバ又はディクショナリサーバ（フロントエンドサーバは指定できません）を指定できます。省略した場合，すべてのサーバ（フロントエンドサーバを除く）を対象とします。

HiRDB/シングルサーバの場合は，-s オプションの指定にかかわらず，シングルサーバを対象とします。

(b)　-f

強制的にリフレッシュする場合に指定します。

-f オプションの指定有無によるプロセス終了契機，及びエラー発生有無の違いを次の表に示します。

表 2–19　-f オプションの指定有無によるプロセス終了契機，及びエラー発生有無の違い（pdpfresh コマンド）

プロセス	-f オプションの指定	
	なし	あり
運用コマンド又はユティリティの内部処理中のシングルサーバ，バックエンドサーバ，及びディクショナリサーバのプロセス	運用コマンド又はユティリティの実行中に pdpfresh コマンドを実行した場合，サーバプロセスのリフレッシュはされないで，エラー終了（リターンコード=6）します。この場合，運用コマンド又はユティリティの動作に影響はありません。	トランザクション終了時点で強制終了します。この場合，運用コマンド又はユティリティが異常終了することがあります。
次のサーバプロセス • AP 単位のローカルバッファを使用しているバックエンドサーバのプロセス • SQL セッション固有一時表を使用しているバックエンドサーバ	サーバプロセスは強制終了しません。これらのプロセスは，左記の条件を満たさなくなった場合，又は次回の DISCONNECT 時に終了します。シングルサーバの場合は，HiRDB クライアントと HiRDB サーバとの間で接続エラーになることがあります。バックエンドサーバの場合はエラーは発生しません。 左記のプロセスがあるかぎり，サーバプロセスのリフレッシュは完了しないため，注意してください。	トランザクション終了時点で強制終了します。以降，SQL を実行すると，新たにプロセスに接続して処理を実行します。
次のサーバプロセス • UNTIL DISCONNECT の排他が掛かっているシングルサーバ • ホールダブルカーソルを使用しているシングルサーバ，又はバックエンドサーバ • SQL セッション固有一時表を使用したシングルサーバ		トランザクション終了時点で強制終了します。以降，SQL を実行すると接続エラーになります。
バックエンドサーバ接続保持機能を使用しているバックエンドサーバのプロセス	トランザクション実行中の場合，トランザクション終了後にプロセスがリフレッシュされます。トランザクション中でない場合は，次回のトランザクション終了後にプロセスがリフレッシュされます。	トランザクション終了時点で強制終了します。以降，SQL を実行すると，新たにプロセスに接続して処理を実行します。

(c)　-c

pdpfresh コマンドでプロセス終了要求を受けたプロセスが，すべて終了しているかどうかを調べる場合に指定します。バッチファイルを使用してプロセスのリフレッシュ状態を監視したい場合に指定してください。

結果は，pdpfresh コマンドのリターンコードで分かります（KFPS00730-I メッセージにも表示されます）。

プロセスごとのリフレッシュ状態を調べたい場合は，pdls -d prc -c コマンドを実行してください。

(5)　規則

1. pdpfresh コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdpfresh コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

3. pdpfresh コマンドは，サーバプロセスのリフレッシュの完了を待たないで終了します。そのため，リフレッシュ中にエラーが発生した場合，pdpfresh コマンドがリターンコード 0 で終了した後に，エラーメッセージが出力されることがあります。
4. pdpfresh コマンド実行後，プロセスのリフレッシュ完了前に再度 pdpfresh コマンドを実行した場合，再実行時の稼働中プロセスがリフレッシュ対象となります。
5. HiRDB/パラレルサーバでフロントエンドサーバの名称を指定した場合，リフレッシュをしないで処理を正常終了します。
6. トランザクション実行中のサーバプロセスは，すぐには終了しないで，トランザクションが完了した時点で終了します。
7. HiRDB/シングルサーバの場合，リフレッシュ実行中に HiRDB クライアントと HiRDB サーバとの間で接続エラーが発生することがあります。これを回避したい場合は，UAP に，接続エラー発生時に接続処理を再実行する処理を追加してください。また，自動再接続機能を使用すると，このエラーが UAP に返却されないで，自動的に再接続がされます。自動再接続機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

(6) 注意事項

1. pdpfresh コマンドの実行結果は，pdls -d prc コマンド，及びコマンド実行時のリターンコードで確認できます。コマンド実行時のリターンコードを次に示します。

リターンコード	実行結果		備考
0	-c オプションあり	リフレッシュするプロセスはありません。	−
	-c オプションなし	リフレッシュ処理を開始しました。	
1	-c オプションあり	リフレッシュ中のプロセスがあります。	-c オプション指定時にだけ出力されます。
	-c オプションなし	−	
4	警告終了しました（一つ以上のサーバでエラーが発生）。※		HiRDB/パラレルサーバの場合にだけ出力されます。
6	運用コマンド又はユティリティが実行中のため，リフレッシュ処理をしないで終了しました。		-f オプションを指定していない場合にだけ出力されます。
8	異常終了		−

（凡例）−：該当しません。

注※

何らかの要因で一つ以上のサーバから応答がなかった場合，ほかのサーバに対して処理を続行し，リターンコード 4 で終了します。この場合，エラーが発生したサーバの状態を確認してください。

エラーが発生したサーバが稼働中の場合，そのサーバプロセスが RD エリアにアクセス中のままとなっている可能性があります。エラーが発生したサーバが稼働中の場合は，-s オプションにそのサーバ名を指定して，再度 pdpfresh コマンドを実行してください。

2. pdpfresh コマンドは，サーバプロセスのリフレッシュ完了を待たないで終了します。

3. -c オプション指定なしの pdpfresh コマンドがリターンコード 0 となった後，-c オプション指定ありの pdpfresh コマンドがリターンコード 0 になれば，リフレッシュ完了です。
4. -c オプション指定ありの pdpfresh コマンドのリターンコード 0 は，コマンド実行時点でリフレッシュ中のプロセスがないことを意味します（リフレッシュ処理が実行され，それが完了したことを意味するものではありません）。したがって，-c オプション指定なしの pdpfresh コマンドを実行しないで-c オプションありの pdpfresh コマンドを実行すると，リターンコード 0 となりますが，リフレッシュ処理はされていないので注意してください。
5. pdpfresh コマンドでサーバプロセスをリフレッシュした場合，KFPS01820-E，KFPS04620-I，及び KFPS01819-I メッセージは出力されません。
6. リフレッシュ中は，常駐プロセス数を一時的に 0 にします。したがって，リフレッシュ中のサーバに対して，pdchprc コマンドは実行しないでください。実行した場合，リフレッシュが完了しないことがあります。この場合，再度 pdpfresh コマンドを実行して，リフレッシュが完了したら，pdchprc コマンドを実行してください。
7. リフレッシュ対象のサーバプロセスの数が多い場合，すべてのプロセスのリフレッシュが完了するまでに時間が掛かることがあります。
8. pdpfresh コマンド（-f 及び-c オプション指定なし）と運用コマンド又はユティリティの実行順序によって，実行結果が異なります。

pdpfresh コマンドと運用コマンド又はユティリティの実行順序	運用コマンド又はユティリティ実行によって割り当てられるプロセス	pdpfresh コマンドの実行結果
運用コマンド又はユティリティ実行後に pdpfresh コマンドを実行	−	エラー終了します（リターンコード=6）。
pdpfresh コマンド実行後に運用コマンド又はユティリティを実行	リフレッシュ対象のプロセス※1	正常終了します（リターンコード=0）。
	リフレッシュされたプロセス※2	

（凡例）−：該当しません。

注※1

運用コマンド又はユティリティが終了した後にリフレッシュします。リフレッシュが完了するまで，pdpfresh -c コマンドはリターンコード 1 で終了します。

注※2

運用コマンド又はユティリティはリフレッシュされたプロセスを割り当てて実行するため，pdpfresh -c コマンドはリフレッシュが完了しているプロセスとして扱います。

また，pdpfresh コマンド（-f 及び-c オプション指定なし）と運用コマンド又はユティリティを同時に実行，又は連続して実行すると，実行結果が不定となります。この現象を回避したい場合は，次に示すどちらかの方法を実行してください。

- pdpfresh コマンド（-f 及び-c オプション指定なし）実行した後，pdpfresh -c コマンドでプロセスのリフレッシュが完了するのを確認してから運用コマンド又はユティリティを実行する
- 運用コマンド又はユティリティが終了するのを待って，pdpfresh コマンド（-f 及び-c オプション指定なし）を実行する

2.82　pdplgrgst（プラグインの登録・削除）

2.82.1　pdplgrgst の形式と規則

(1)　機能

●プラグインを HiRDB に登録します。

登録するプラグインのプラグイン定義ファイル，及び PIC ファイルを指定します。

pdplgrgst コマンドを実行すると，プラグインに関する定義情報がディクショナリに登録されます。抽象データ型定義，又はインデクス型定義も同時に登録されます。

一回の pdplgrgst コマンドの実行で，一つのプラグインが登録できます。

●プラグインを HiRDB から削除します。

次に示すものを削除します。

- データディクショナリに登録されているプラグインの定義情報
- プラグインが提供している関数，抽象データ型，及びインデクス型

●プラグインをバージョンアップします。

既に登録されているプラグインの，定義情報，関数，抽象データ型定義，及びインデクス型定義を更新します。

(2)　実行者

DBA 権限を持っているユーザが実行できます。

(3)　形式

```
pdplgrgst 〔- {d | a} 〕〔-u〕
          プラグイン定義ファイル名 PICファイル名
```

(4)　オプション

(a)　- {d ｜ a}

プラグインを削除又はバージョンアップする場合に指定します。省略した場合はプラグインを HiRDB に登録します。

-d：

プラグインを削除する場合に指定します。

-a：

プラグインをバージョンアップする場合に指定します。

(b) -u

プラグインの所有者を MASTER ではなく，pdplgrgst コマンドの実行者（クライアント環境定義の PDUSER に指定した認可識別子）にする場合に指定します。

プラグインの所有者を MASTER 以外にする場合，次に示すことを注意してください。

1. pdplgrgst コマンド実行者のスキーマが既に定義されている必要があります。
2. プラグインが抽象データ型及びインデクス型の両方を提供している場合は，必ず同じ所有者にしてください。
3. プラグインの削除又はバージョンアップは，プラグインの所有者しかできません。また，pdplgrgst コマンドに-u オプションを指定してプラグインを削除又はバージョンアップしてください。
4. プラグイン所有者のスキーマを削除すると，プラグインも同時に削除されます。このとき，システム共通定義から pdplugin オペランドを削除する必要があります。

(5) コマンド引数

(a) プラグイン定義ファイル名

プラグイン定義ファイルのファイル名を指定します。

コマンドを入力するときのカレントディレクトリにあるファイル名，又は絶対パスを含めたファイル名を指定します。

(b) PIC ファイル名

PIC ファイルのファイル名を指定します。

コマンドを入力するときのカレントディレクトリにあるファイル名，又は絶対パスを含めたファイル名を指定します。

(6) 規則

1. pdplgrgst コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdplgrgst コマンドは，任意のサーバマシンで実行してください。
3. pdplgrgst コマンドを実行する場合，事前にクライアント環境定義 PDHOST，PDNAMEPORT，及び PDUSER を設定しておく必要があります。なお，PDUSER には DBA 権限のあるユーザの認可識別子及びパスワードを設定してください。

4. 通信情報ファイルディレクトリ変更機能を使用している場合，クライアント環境定義 PDIPCFILEDIR に値を設定する必要があります。指定値については，マニュアル「HiRDB UAP 開発ガイド」の「クライアント環境定義の設定内容」を参照してください。

(7) 注意事項

1. pdplgrgst コマンドの実行結果は，コマンド実行時のリターンコードで確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了です。
2. pdplgrgst コマンドを実行する場合，必ず HiRDB の利用環境すべてでプラグインをインストールしておいてください。
3. pdplgrgst コマンドで HiRDB に登録したプラグインを使用するには，システム共通定義に使用するプラグインの名称を記述した後，HiRDB を再起動してください。
4. プラグインの登録中は，そのスキーマに対してほかの処理はできません。
5. プラグイン登録時に pdplgrgst コマンドが異常終了した場合，型定義が残ることがあります。このときは，pdplgrgst コマンドを-d オプション指定で実行してプラグインを削除し，再度登録してください。
6. プラグインを削除する場合，pdplgrgst コマンド実行前に次のものを削除しておいてください。
   - 削除するプラグインを使用した表，インデクス，及びビュー表
   - 削除するプラグインが提供している抽象データ型，インデクス型，及び関数を利用した，関数，手続き，及び抽象データ型
7. データ型プラグインを削除する場合，そのデータ型のインデクス機能を提供するインデクス型プラグインも登録されているときは，インデクス型プラグインを先に削除してください。詳細は各プラグインのマニュアルを参照してください。
8. プラグインを削除した場合，システム共通定義 pdplugin の記述を削除してください。
9. プラグインをバージョンアップする場合，pdplgrgst コマンドを実行する前に，マスタディレクトリ用 RD エリア，データディクショナリ用 RD エリア，データディレクトリ用 RD エリア，及びデータディクショナリ LOB 用 RD エリアのバックアップを取得してください。
10. 登録されているプラグインよりもバージョンの古いプラグインに戻すことはできません。
11. データ型プラグインと，そのデータ型のインデクス機能を提供するインデクス型プラグインの両方をバージョンアップする場合は，データ型プラグインを先にバージョンアップしてください。
12. データ型プラグインとインデクス型プラグインのバージョンを合わせてください。
13. プラグインをバージョンアップした場合，データディクショナリ表の SQL_PLUGIN_ROUTINES 表及び SQL_PLUGIN_ROUTINE_PARAMS 表の SPECIFIC_NAME 列は更新されます。
14. データベース構成変更ユティリティ（pdmod）で HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成に移行した場合，登録しているプラグインによってはプラグイン提供の関数が無効になることがあります。その場合は，-a オプションを指定して再度 pdplgrgst コマンドを実行してください。
15. 複数のプラグインで関数名とパラメタの数が同じ関数を提供している場合，次の条件を満たすときは，プラグインの登録がエラーとなります。

- 一方のプラグイン登録後にそのプラグイン関数を呼び出す関数をユーザが定義し，もう一方のプラグイン登録時にユーザが定義した関数のパラメタ又は戻り値のデータ型に抽象データ型を使用し，かつその関数をビュー定義に使用している。

2.83　pdprfed（PRF トレース情報の編集出力）

2.83.1　pdprfed の形式と規則

(1)　機能

PRF トレース情報ファイル（バイナリ形式）を CSV 形式に変換して標準出力に出力します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdprfed 〔-T 開始時刻，終了時刻〕
        〔-p プロセスID〔，プロセスID〕…〕　〔-e イベントID〔，イベントID〕…〕
        〔-H〕
        PRFトレース情報ファイル名
```

(4)　オプション

(a)　-T 開始時刻，終了時刻

指定された時刻内に取得されたトレース情報だけを CSV 出力します。時刻は，YYYYMMDDHHMMSS 形式で指定します。

(b)　-p プロセス ID〔，プロセス ID〕...

指定されたプロセス ID の情報だけを CSV 出力します。複数のプロセス ID を指定した場合は，すべてに当てはまるトレース情報だけ出力します。

(c)　-e イベント ID〔，イベント ID〕...

指定されたイベント ID の情報だけを CSV 出力します。複数のイベント ID を指定した場合は，すべてに当てはまるトレース情報だけ出力します。

(d)　-H

PRF トレース情報を，ヘッダを付けないで CSV 出力します。

(5)　コマンド引数

(a)　PRF トレース情報ファイル名

PRF トレース情報が格納されているファイル名を指定します。

(6)　規則

1. pdprfed コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdprfed コマンドは，任意のサーバマシンで実行できます。

(7)　注意事項

1. pdprfed コマンドのリターンコードを次に示します。

   0：正常終了

   8：異常終了

   12：異常終了（内部矛盾）
2. このコマンドを指定すると，時刻順ではなく，PRF トレース情報ファイル（バイナリ形式）に格納された順番に PRF トレースが CSV 出力されます。PRF トレースファイルの内容を時刻順にソートしたり，複数ファイルの内容をマージしたりする必要がある場合は，このコマンドで出力した CSV データを編集してください。
3. PRF トレース情報ファイル（バイナリ形式）内の情報が一部出力されていなかったり，不正な情報だったりした場合も，そのまま CSV 出力されます。その場合，リターンコードは正常終了（0）となります。
4. PRF トレース情報ファイル（バイナリ形式）の一部の項目だけが破損していた場合は，「出力形式」とは異なる形式で結果が出力されることがあります。
5. PRF トレース情報ファイルには，同じプラットフォームで取得したファイルを指定してください。異なるプラットフォームで取得したファイルを指定した場合，不当な結果が出力されます。

(8)　出力形式

出力例を次に示します。

```
time,svname,pid,tid,eid,no,ip,len,c_ip,c_pid,c_port,c_conn,c_no,r_id,r_user,binary,text
2012/12/04 12:13:06.902202,f001,9974,1,0x10022002,13,10.197.12.141,104,10.197.12.141,12862,,
2,,unt100000002,,02004F50 4E320000,..OPN2..
2012/12/04 12:13:06.902684,f001,9974,1,0x13012503,14,10.197.12.141,124,10.197.12.141,12862,,
2,,unt100000002,,64000000 62303131 00202020 00000000 E42E0000 01000000 00000000,d...b011.
................
2012/12/04 12:13:06.902769,f001,9974,1,0x11019101,15,10.197.12.141,92,,,,,,unt100000002,,0AC
50C8D 0200E42E 00000000 AA2A0000,............*..
2012/12/04 12:13:06.902797,f001,9974,1,0x11019100,16,10.197.12.141,72,,,,,,unt100000002,,,
2012/12/04 12:13:06.903019,b011,10064,1,0x11029101,1,10.197.12.141,60,,,,,,,,
2012/12/04 12:13:06.903476,b011,10064,1,0x11009600,2,10.197.12.141,168,,,,,,unt100000002,,2F
350E00 20000000 30000000 05000000 F6260000 48524442 756E7431 02010100 00000000 00000000 0200
0000 03000000 05000000 04000000 90000000 48524442 756E7431 66303031 00000000 00000000 F62600
```

```
00 02000000 05000000,/5.. ...0........&..HRDBunt1.............................HRDBunt1f00
1.........&.........
2012/12/04 12:13:06.903503,b011,10064,1,0x13022003,3,10.197.12.141,112,0.0.0.0,0,,2,,unt1000
00002,,64000000 0000A03A 00000000 01000000,d......:........
2012/12/04 12:13:06.904324,f001,9974,1,0x10012002,17,10.197.12.141,108,10.197.12.141,12862,,
2,,unt100000002,,02004F50 4E320000 00000000,..OPN2......
```

〔説明〕

出力される情報について次の表に示します。

ヘッダ	内容	出力形式との対応
time	トレース取得時刻	2010/02/10 13:40:36.201194
svname	サーバ名，プロセス名※	fes1
pid	プロセス ID	30089
tid	スレッド ID	1
eid	イベント ID	0x10012002
no	PRF トレース通番	125
ip	IP アドレス	10.208.175.34
len	保守情報	100
c_ip	クライアントの IP アドレス	10.208.175.35
c_pid	クライアントのプロセス ID	31884
c_port	クライアントのポート番号	30142
c_conn	クライアントのコネクト通番	3
c_no	保守情報	
r_id	PRF くくり付け ID※	unt10000043E
r_user	Cosminexus と連携した場合： ルートアプリケーション情報 上記以外： 保守情報	41424300 4AE40000 012E0000
binary	保守情報	02004155 49320000 00000000
text	保守情報	..AUI2.....

（凡例）

空欄：出力されません。

注※

PRF トレース情報ファイル（バイナリ形式）の欠損によって，文字データが不正なバイトは「.」で表示されます。

2.84　pdprflevel（PRF トレース取得レベルの表示，変更）

2.84.1　pdprflevel の形式と規則

(1)　機能

PRF トレース取得レベル（pd_prf_level オペランドの指定値）を表示又は変更します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdprflevel 〔-l PRFトレース取得レベル〕
```

(4)　オプション

(a)　-l PRF トレース取得レベル

PRF トレース取得レベルを表示又は変更します。

PRF トレース取得レベル：

変更する PRF トレース取得レベルを指定します。PRF トレース取得レベルは，00000007（最小レベル），0000001f（標準レベル），0000007f（詳細レベル），000001ff（保守レベル），00000000（抑止レベル）で指定します。詳細は，マニュアル「HiRDB システム定義」の pd_prf_level オペランドを参照してください。

省略時：

PRF トレース取得レベルを 16 進数で表示します。

(5)　規則

1. pdprflevel コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdprflevel コマンドは，各サーバマシンで実行してください。

(6)　注意事項

1. pdprflevel コマンドのリターンコードを次に示します。
   0：正常終了
   8：異常終了

2. このコマンドは，システム定義の pd_prf_trace に Y を指定又は省略し，トラブルシュート情報取得プロセス（pdprfd プロセス）の起動中に実行してください。pdprfd プロセスが起動していないときにコマンドを実行すると，コマンドがエラー終了します。pdprfd プロセスは，OS のタスクマネージャなどで，起動状態を確認できます。
3. このコマンドで PRF トレース取得レベルを変更した場合，次の PRF トレース取得ポイントから変更が有効になります。PRF トレース取得ポイントについては，マニュアル「HiRDB システム運用ガイド」を参照してください。
4. このコマンドで PRF トレース取得レベルを変更した後，HiRDB を終了後，再度 HiRDB を開始すると，PRF トレース取得レベルは pd_prf_level オペランドの指定値に戻ります。
5. このコマンドで PRF トレース取得レベルを変更した後に系切り替えが発生した場合，変更後の PRF トレース取得レベルは待機系 HiRDB へは引き継がれません。サーバモードで系切り替えを運用している場合は，待機系でも同じ PRF トレース取得レベルを指定し，pdprflevel コマンドを実行してください。モニタモードで系切り替えを運用している場合は，系切り替え後に PRF トレース取得レベルを変更してください。

2.85　pdprgcopy（修正版 HiRDB のコピー）

2.85.1　pdprgcopy の形式と規則

(1)　機能

pdprgrenew コマンドを実行する前に，HiRDB のプログラムを修正版 HiRDB のインストールディレクトリから入れ替え用ディレクトリ（%PDDIR%¥renew）にコピーします。修正版 HiRDB への入れ替えについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

入れ替え用ディレクトリにコピーするファイルを次に示します。

修正版 HiRDB のインストールディレクトリ（コピー元）	入れ替え用ディレクトリ（コピー先）	内容
インストールディレクトリ¥bin	%PDDIR%¥renew¥bin	HiRDB の運用コマンド及びユティリティ
インストールディレクトリ¥lib	%PDDIR%¥renew¥lib	HiRDB のライブラリ，メッセージテキストファイル，及びシステム定義の定義解析用ファイル
インストールディレクトリ¥client	%PDDIR%¥renew¥client	HiRDB クライアントのライブラリ，運用コマンド，及びユティリティ
インストールディレクトリ¥include	%PDDIR%¥renew¥include	UAP 作成時に使用するヘッダ情報
インストールディレクトリ¥HiRDEF	%PDDIR%¥renew¥HiRDEF	HiRDB 定義支援
インストールディレクトリ¥sample	%PDDIR%¥renew¥sample	サンプルデータベース機能
インストールディレクトリ¥pdistup	%PDDIR%¥renew¥pdistup	HiRDB セットアップツール

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
修正版HiRDBのインストールディレクトリ¥bin¥pdprgcopy HiRDB運用ディレクトリ
```

(4)　コマンド引数

(a)　HiRDB 運用ディレクトリ　～＜パス名＞((1～200))

修正版 HiRDB に入れ替える HiRDB の，HiRDB 運用ディレクトリを指定します。

(5) 規則

1. pdprgcopy コマンドは，HiRDB の稼働に関係なく実行できます。
2. pdprgcopy コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 入れ替え前の HiRDB と修正版 HiRDB とで，次の項目が一致していないと pdprgcopy コマンドは実行できません。
   - HiRDB の種別（HiRDB/シングルサーバ，又は HiRDB/パラレルサーバ）
   - バージョン，リビジョン
   - アドレッシングモード（32 ビットモード，又は 64 ビットモード）

(6) 注意事項

1. pdprgcopy コマンドは，入れ替え用ディレクトリ（%PDDIR%¥renew）を作成するため，同一名称のディレクトリがある場合は，あらかじめ別のディレクトリ名に変更しておいてください。
2. pdprgcopy コマンドを実行する前に，HiRDB 運用ディレクトリに空き容量があるか確認してください。
3. pdprgcopy コマンドを実行する前に，あらかじめ修正版 HiRDB をすべてのユニットにインストールしておく必要があります。
4. 入れ替え前の HiRDB と修正版 HiRDB では，文字コード種別は同じになります。
5. pdprgcopy コマンドの結果は，コマンド実行時のリターンコードで確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了です。8 の場合は，その前に出力されているメッセージを参照してエラー要因を取り除いてください。

2.86　pdprgrenew（修正版 HiRDB の入れ替え）

2.86.1　pdprgrenew の形式と規則

(1)　機能

稼働中の HiRDB を，修正版 HiRDB（あらかじめ pdprgcopy コマンドでコピーしておいた入れ替え用ディレクトリ（%PDDIR%¥renew）の HiRDB）に入れ替えます。修正版 HiRDB への入れ替えについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

稼働中の HiRDB の，次の HiRDB 運用ディレクトリのファイルを入れ替えます。

ディレクトリ	内容
%PDDIR%¥renew¥bin	HiRDB の運用コマンド及びユティリティ
%PDDIR%¥renew¥lib	HiRDB のライブラリ，メッセージテキストファイル，及びシステム定義の定義解析用ファイル
%PDDIR%¥renew¥client	HiRDB クライアントのライブラリ，運用コマンド，及びユティリティ
%PDDIR%¥renew¥include	UAP 作成時に使用するヘッダ情報
%PDDIR%¥renew¥HiRDEF	HiRDB 定義支援
%PDDIR%¥renew¥sample	サンプルデータベース機能
%PDDIR%¥renew¥pdistup	HiRDB セットアップツール

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
修正版HiRDBのインストールディレクトリ¥bin¥pdprgrenew
        〔-b〕 HiRDB運用ディレクトリ
```

(4)　オプション

(a)　-b

pdprgrenew コマンドが KFPS04646-I（リターンコード=12）で異常終了した場合など，pdprgrenew コマンド実行前の状態に HiRDB を戻すときに指定します。

(5) コマンド引数

(a) HiRDB 運用ディレクトリ ～＜パス名＞ ～＜パス名＞((1～200))

修正版 HiRDB に入れ替える HiRDB の，HiRDB 運用ディレクトリを指定します。

(6) 規則

1. pdprgrenew コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdprgrenew コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdprgrenew を実行する場合，あらかじめ pdprgcopy コマンドで入れ替え用ディレクトリ（%PDDIR%¥renew）を作成しておく必要があります。作成していない場合は実行できません。なお，入れ替え用ディレクトリ（%PDDIR%¥renew）は，pdprgrenew コマンドが正常終了したときに削除されます。

(7) 注意事項

1. pdprgrenew コマンドは，バックアップ用ディレクトリ（%PDDIR%¥renew_bak）を作成するため，同一名称のディレクトリがある場合は，あらかじめ別のディレクトリ名に変更しておいてください。
2. pdprgrenew コマンドを実行する前に，次のプログラムが起動されている場合は終了してください。
   - HiRDB コマンドプロンプト
   - イベントビューア
3. pdprgrenew コマンドを実行すると，システムログファイルが切り替わります。そのため，pdprgrenew コマンドを実行する前に，入れ替え後の HiRDB で稼働するための容量が，システムログファイルにあるかどうかを確認してから実行してください。スワップ先にできるシステムログファイルがない状態で pdprgrenew コマンドを実行すると，KFPS01256-E メッセージを出力し，HiRDB は停止します（アボートコード Psjnf07 又は Psjn381 を出力）。HiRDB が停止した場合，スワップ先にできるシステムログファイルを準備してから，HiRDB を開始してください。スワップ先にできるシステムログファイルがない場合，システムログファイルを追加する必要があります。スワップ先にできるファイルがない場合の対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
4. pdprgrenew コマンドを実行すると，メッセージログファイルが切り替わります。ただし，メッセージログファイルの切り替えを示すメッセージは出力されません。そのため，メッセージログファイルのメッセージログが必要な場合は，pdprgrenew コマンドを実行する前にメッセージログファイルのバックアップを取得してください。
5. 稼働中の HiRDB が次の状態の場合，HiRDB の入れ替えができないことがあります。
   - HiRDB/パラレルサーバの場合，一部のサーバ又はユニットが停止しているとき（縮退起動によって開始していないユニットも含む）
   - HiRDB/パラレルサーバの場合，ユニット間のネットワークに通信障害が発生しているとき
   - pdprgrenew コマンド実行後，15 分間以上トランザクション，又はユティリティが動作している場合

- インメモリ RD エリアがある場合
- pdrplstop コマンドを実行中の場合
- HiRDB Datareplicator のデータ抽出機能を使用している場合に，pd_rpl_init_start オペランドの指定と実際のデータ抽出モードが異なっているとき

  pd_rpl_init_start オペランドの値と pdprgrenew コマンドの実行可否を次に示します。

pd_rpl_init_start オペランドの値	HiRDB Datareplicator のデータ抽出機能の実行状況	pdprgrenew コマンドの実行可否
Y	実行中	○
	pdrplstop コマンドによって停止中	×
N（省略値）	停止中	○
	pdrplstart コマンドによって実行中	×

（凡例）

○：実行できます。

×：実行できません。

6. pdprgrenew コマンド実行中は，次の運用に制限が生じます。
   - UAP のレスポンスが，pdprgrenew コマンドの実行時間だけ遅くなります。
   - 運用コマンド，及びユティリティを実行した場合，pdprgrenew コマンドの実行が失敗したり，HiRDB が停止している旨のエラーメッセージを表示したりすることがあります。
7. HiRDB 運用ディレクトリ下のファイルは，HiRDB 運用ディレクトリと同じ HiRDB ファイルシステム領域になければなりません。
8. pdprgrenew コマンドの結果は，コマンド実行時のリターンコードで確認できます。

   pdprgrenew コマンドのリターンコードを次に示します。

   0：正常終了

   8：異常終了（入れ替え前の HiRDB の状態）

   標準エラー出力やイベントログに出力されたエラーメッセージを参照してエラーの原因を取り除いてください。その後，コマンドを再実行してください。

   12：異常終了（HiRDB が停止状態）

   マニュアル「HiRDB システム導入・設計ガイド」の「修正版 HiRDB への入れ替え」を参照して対処してください。
9. pdprgrenew コマンド実行中，HiRDB の開始，終了に伴うエラーメッセージ及びアボートコードが出力されることがあります。この場合，入れ替え後の HiRDB を開始するために必要とするリソース不足など，環境に問題があることがあります。エラーメッセージ，及びアボートコードの対処方法に従って対処してください。

2.87　pdrbk（トランザクションのロールバック）

2.87.1　pdrbk の形式と規則

(1)　機能

pdrbk コマンドは，グローバルトランザクションを構成している各トランザクションブランチが，何らかの要因（通信障害など）でトランザクションを決着できない場合に，強制的にトランザクションをロールバックするときに使用します。この場合，ほかのトランザクションとの不整合を発生させないために，グローバルトランザクション内のほかのトランザクションもロールバックしなければなりません。

pdrbk コマンドは，「pdls -d trn」コマンドを実行して表示された情報中のトランザクション第１状態が READY でトランザクション第２状態がｐのトランザクションを，強制的にロールバックします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdrbk 〔 {-x ホスト名 | -u ユニット識別子} 〕 〔-s サーバ名〕
        {-A | -t トランザクション識別子}
```

(b)　HiRDB/パラレルサーバの場合

```
pdrbk {-x ホスト名 | -u ユニット識別子} 〔-s サーバ名〕
        {-A | -t トランザクション識別子}
```

(4)　オプション

(a)　{-x ホスト名｜-u ユニット識別子}

該当するトランザクションがあるホストのホスト名，又はユニットのユニット識別子を指定します。

-x ホスト名　～＜識別子＞((1～32))

該当するトランザクションがあるホストのホスト名を指定します。

-u ユニット識別子　～＜識別子＞((4 文字))

該当するトランザクションがあるユニットのユニット識別子を指定します。

(b)　-s サーバ名　～＜識別子＞((1～8))

ユニットに複数のサーバがある場合，サーバを特定するときにサーバ名を指定します。省略した場合，対象となるホスト内のすべてのサーバを対象とします。

(c)　-A

-s オプションで指定したサーバ内のトランザクションで，READY 状態で回復処理待ち状態のすべてのトランザクションをロールバックする場合に指定します。

(d)　-t トランザクション識別子　～＜英数字＞((16 文字))

READY 状態で回復処理待ち状態のトランザクションをロールバックする場合に，トランザクション識別子を指定します。

(5)　規則

1. pdrbk コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdrbk コマンドは，各サーバマシンで実行してください。

(6)　注意事項

1. pdrbk コマンドの結果は，pdls コマンド（-d trn 指定），コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   1：異常終了（オプション指定不正，rsh 失敗など）

   ロールバックできていないトランザクションが残っていることがあります。

   メッセージに従って対処した後，再度コマンドを実行してください。
2. 回復不要 FES に接続して実行したトランザクションは，トランザクション第 1 状態が READY，トランザクション第 2 状態が p の場合でも，強制的にロールバックできないことがあります。この場合，トランザクションを自動決着してください。未決着状態のトランザクションを決着する方法の詳細については，マニュアル「HiRDB システム運用ガイド」の「未決着状態のトランザクションを決着する方法」を参照してください。
3. 現在は停止しているクライアント製品からの要求で発生した未決着状態のトランザクションを強制的にロールバックすると，この後でクライアント製品を開始したときに，HiRDB とクライアント製品との間でこのトランザクションの同期が取れないことがあります。

2.88　pdrdrefls（関連する RD エリアの情報の表示）

2.88.1　pdrdrefls の形式と規則

(1)　機能

関連する RD エリアの情報を表示します。

pdrdrefls コマンドには，次の二つの機能があります。

1. 関連する RD エリアの情報の表示
2. RD エリアの関連性のチェック

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　関連する RD エリアの情報の表示

```
pdrdrefls 〔-k dsp〕 -e org
          {-r RDエリア名〔, RDエリア名〕…|-r ALL
           |-t 〔認可識別子.〕表識別子}
          〔-s サーバ名〔, サーバ名〕…〕 〔-l 〔-d 区切り文字〕〕 〔-a〕 〔-c 制約種別〕
```

(b)　RD エリアの関連性のチェック

```
pdrdrefls -k chk -e org {-r RDエリア名〔, RDエリア名〕…|-r ALL}
          〔-s サーバ名〔, サーバ名〕…〕 〔-c 制約種別〕
```

(4)　オプション

(a)　-k {dsp | chk}

実行する pdrdrefls コマンドの機能を指定します。

dsp：

表を格納するすべての RD エリア，及び RD エリア内にある資源（表，インデクス，及び LOB 列）について，関連のある RD エリアの情報を表示します。

chk：

RD エリアの関連性をチェックします。関連する RD エリアが不足している場合，不足している RD エリアの情報をメッセージで出力します。

(b)　-e org

固定のオプションです。

(c)　-r RD エリア名〔，RD エリア名〕...　～＜識別子＞((1～30))

処理対象の RD エリアの名称を指定します。ただし，指定できるのはユーザ用 RD エリア（一時表用 RD エリアを除く）及びユーザ LOB 用 RD エリアです。

RD エリアを指定する場合の規則については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。

(d)　-r ALL

すべてのユーザ用 RD エリア（一時表用 RD エリアを除く），及びユーザ LOB 用 RD エリアを処理対象とする場合に指定します。

(e)　-t〔認可識別子.〕表識別子　～＜識別子＞((認可識別子：1～30，表識別子：1～30))

処理対象の表の名称を指定します。

認可識別子を省略した場合，pdrdrefls コマンド実行時の環境変数 PDUSER の認可識別子が仮定されます。環境変数 PDUSER を設定していない場合，及び環境変数 PDUSER の認可識別子とパスワードの両方に簡易認証キーワードを指定している※場合，ログインウィンドウのユーザ名が仮定されます。-t オプションの指定と環境変数 PDUSER の設定によって有効となる認可識別子について次に示します。

-t オプションの認可識別子	環境変数 PDUSER の設定内容	有効な認可識別子
指定あり	－	-t オプションに指定された認可識別子
省略	認可識別子[/パスワード]	環境変数 PDUSER に指定された認可識別子
	設定なし	ログインウィンドウのユーザ名
	認可識別子とパスワードの両方に簡易認証キーワードを指定	

注※

OS ログインユーザの簡易認証機能の有効，無効（システム共通定義の pd_os_authenticate オペランドの指定値）にかかわらず，環境変数 PDUSER の設定内容で判断します。簡易認証キーワードについては，マニュアル「HiRDB システム運用ガイド」の「OS ログインユーザの簡易認証機能」を参照してください。

認可識別子及び表識別子を「¥""」と「"¥"」で囲んだ場合は，英字の大文字と小文字が区別されますが，「¥""」と「"¥"」で囲まない場合はすべて大文字として扱われます。

(f)　-s サーバ名〔，サーバ名〕...　～＜識別子＞((1～8))

処理対象のサーバの名称を指定します。指定できるサーバの数は，最大 128 個です。

-k dsp の場合，関連する RD エリアのうち，指定したサーバに定義されている情報だけを表示します。

-k chk の場合，-r オプションに指定したすべての RD エリアが，指定したサーバに定義されているかどうかをチェックします。-r オプションに指定したすべての RD エリアがその条件を満たしていない場合，その RD エリア名をメッセージで出力し，エラー終了します。

指定するサーバ名は，pddbst の RD エリア単位の状態解析（論理的解析），又は表単位の状態解析で確認できます。

(g)　-l

出力情報の各情報を改行しないで表示する場合に指定します。

(h)　-d 区切り文字

-l オプション指定時の各情報の区切り文字を，半角 1 文字で指定します。

省略した場合は空白が仮定されます。

区切り文字としてアポストロフィ（'）や引用符（"）などの特殊文字を指定する場合，「¥'」，「¥"」と指定してください。エスケープ文字を区切り文字として指定する場合は，「¥¥」と指定してください。

(i)　-a

すべての関連情報（サーバ名，RD エリア名，RD エリア種別，及び資源情報）を表示する場合に指定します。省略した場合，RD エリア名だけが表示されます。

(j)　-c 制約種別

制約定義の「関連する RD エリアの情報」を表示又はチェックする場合に指定します。

ref：

参照制約の「関連する RD エリアの情報」を表示又はチェックします。参照制約の運用方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(5)　規則

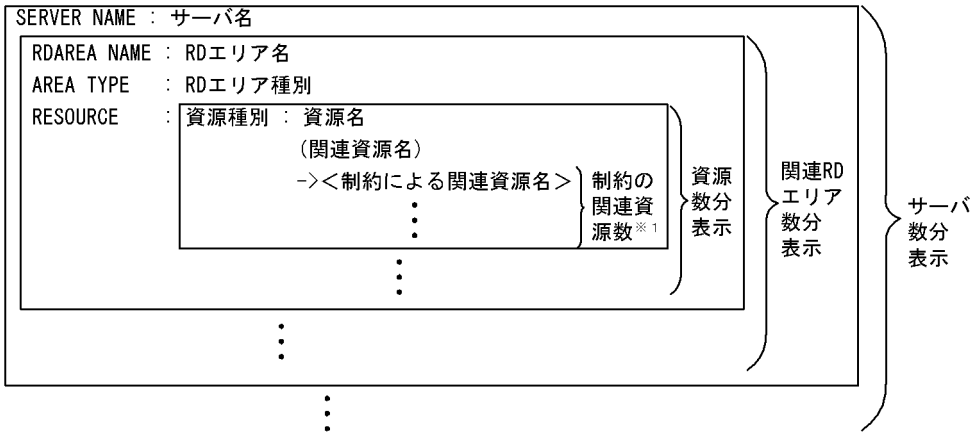
1. pdrdrefls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdrdrefls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6) 注意事項

1. pdrdrefls コマンドの結果は，コマンド実行時のリターンコードで確認できます。リターンコードが 0 の場合は正常終了，4 の場合は警告終了（解析対象の資源がありません），8 の場合は異常終了です。
2. pdrdrefls コマンドの処理対象となる RD エリアに対して，表，インデクスの追加，削除，及び定義変更はしないでください。追加，削除，及び定義変更をした場合は，結果不正や異常終了となることがあります。

(7) 出力形式

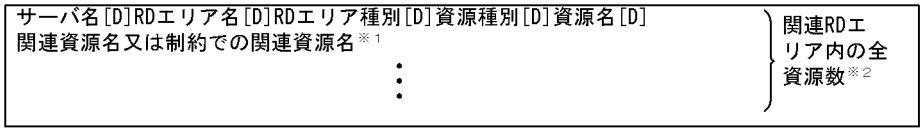
●-l指定なし，-a指定ありの場合



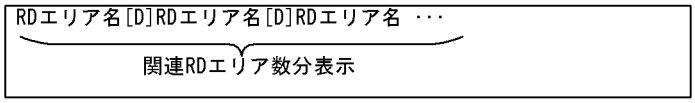
●-l指定なし，-a指定なしの場合



●-l指定あり，-a指定ありの場合



●-l指定あり，-a指定なしの場合



（凡例）[D]：区切り文字
注※1　-cオプションを指定した場合に表示されます。
注※2　複数の関連資源又は制約関連資源がある資源は，その分の行が表示されます。

〔説明〕

サーバ名（最大 8 バイト）：

関連する RD エリアがあるサーバの名称です。

RD エリア名（最大 32 バイト）：

関連する RD エリアの名称です。RD エリア名は引用符（"）で囲んで表示されます。

RD エリア種別（最大 8 バイト）：

RD エリアの種別です。

USER：ユーザ用 RD エリア

USER_LOB：ユーザ LOB 用 RD エリア

資源種別（3 バイト）：

資源の種別です。

TBL：表

IDX：インデクス

LOB：LOB 列

資源名（最大 65 バイト）：

該当する RD エリアに格納されている資源の名称です。資源名は引用符で囲んで表示されます。

資源種別が TBL の場合：認可識別子.表識別子

資源種別が IDX の場合：認可識別子.インデクス識別子

資源種別が LOB の場合：LOB 列名，又は LOB 属性の抽象データ型列名

関連資源名（最大 67 バイト）：

資源種別が IDX 又は LOB の資源の関連資源名です。関連資源名は（認可識別子.表識別子）の形式で表示され，引用符で囲まれます。

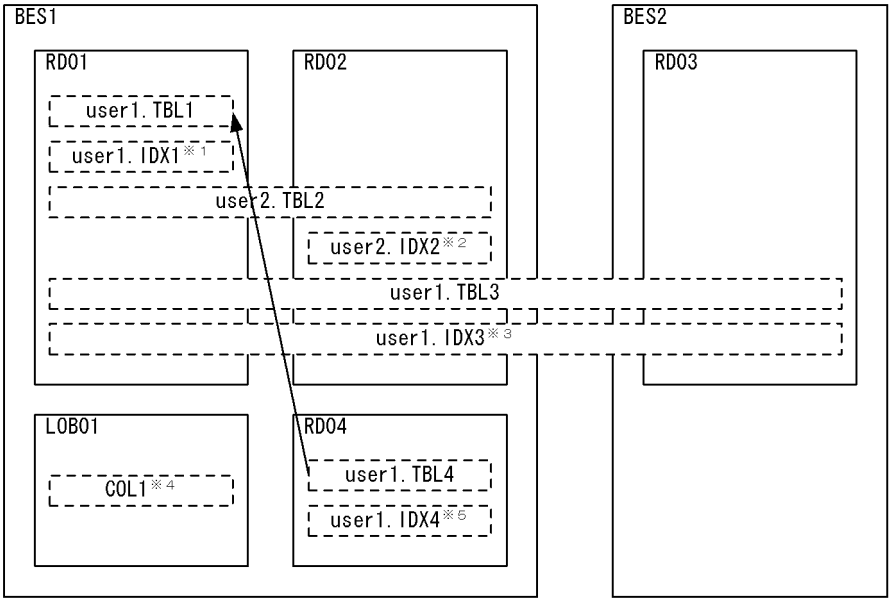
制約による関連資源名（最大 69 バイト）：

資源種別が TBL の資源に対して，制約定義によって関連する資源の名称です。この情報は，-c オプションを指定した場合だけ表示されます。-c ref を指定した場合は，披参照表名称が表示されます。制約による関連資源名は，引用符（"）で囲んで表示されます。

(8)　使用例

関連する RD エリアの情報を表示します。なお，-d オプションを省略しています。また，実行結果の△は 1 バイトの空白を表します。

- **RD エリアの構成**



（凡例）

→：参照制約です。矢印の元が参照表，先が被参照表です。

注※1　user1.TBL1に定義しているインデクスです。
注※2　user2.TBL2に定義しているインデクスです（非分割キーインデクス）。
注※3　user1.TBL3に定義しているインデクスです（分割キーインデクス）。
注※4　LOB列構成基表はuser1.TBL1です。
注※5　user1.TBL4に定義しているインデクスです。

- **実行結果**

- pdrdrefls -e org -r RD01 -a の場合

```
SERVER NAME : BES1
  RDAREA NAME : "RD01"
  AREA TYPE : USER
    RESOURCE  : TBL : "user1"."TBL1"
                      "user2"."TBL2"
                      "user1"."TBL3"
                IDX : "user1"."IDX1"
                      ("user1"."TBL1")
                      "user1"."IDX3"
                      ("user1"."TBL3")
  RDAREA NAME : "RD02"
    AREA TYPE : USER
    RESOURCE  : TBL : "user2"."TBL2"
                      "user1"."TBL3"
                IDX : "user2"."IDX2"
                      ("user2"."TBL2")
                      "user1"."IDX3"
                      ("user1"."TBL3")
  RDAREA NAME : "RD04"                      ┐
    AREA TYPE : USER                        │
    RESOURCE  : TBL : "user1"."TBL4"        ├ ※
                      -><"user1"."TBL1">    │
                IDX : "user1"."IDX4"        │
                     ("user1"."TBL4")       ┘
  RDAREA NAME : "LOB01"
    AREA TYPE : USER_LOB
```

```
    RESOURCE  : LOB : "COL1"
                    ("user1"."TBL1")
```

注※　-c ref を指定した場合だけ表示されます。

- pdrdrefls -e org -r RD01 の場合

```
  RDAREA NAME : "RD01"
  RDAREA NAME : "RD02"
  RDAREA NAME : "RD04" ※
  RDAREA NAME : "LOB01"
```

注※　-c ref を指定した場合だけ表示されます。

- pdrdrefls -e org -r RD01 -l -a の場合

```
BES1△"RD01"△USER△TBL△"user1"."TBL1"
BES1△"RD01"△USER△TBL△"user2"."TBL2"
BES1△"RD01"△USER△TBL△"user1"."TBL3"
BES1△"RD01"△USER△IDX△"user1"."IDX1"△"user1"."TBL1"
BES1△"RD01"△USER△IDX△"user1"."IDX3"△"user1"."TBL3"
BES1△"LOB01"△USER_LOB△LOB△"COL1"△"user1"."TBL1"
BES1△"RD02"△USER△TBL△"user2"."TBL2"
BES1△"RD02"△USER△TBL△"user1"."TBL3"
BES1△"RD02"△USER△IDX△"user2"."IDX2"△"user2"."TBL2"
BES1△"RD02"△USER△IDX△"user1"."IDX3"△"user1"."TBL3"
BES1△"RD04"△USER△TBL△"user1"."TBL4"△
"user1"."TBL1" ※
BES1△"RD04"△USER△IDX△"user1"."IDX4"△
"user1"."TBL4" ※
```

注※　-c ref を指定した場合だけ表示されます。

- pdrdrefls -e org -r RD01 -l の場合

```
"RD01"△"RD02"△"LOB01"△"RD04" ※
```

注※　-c ref を指定した場合だけ表示されます。

- pdrdrefls -e org -r RD02 -a の場合

```
SERVER NAME : BES1
  RDAREA NAME : "RD02"
    AREA TYPE : USER
    RESOURCE  : TBL : "user2"."TBL2"
                      "user1"."TBL3"
                IDX : "user2"."IDX2"
                      ("user2"."TBL2")
                      "user1"."IDX3"
                      ("user1"."TBL3")
  RDAREA NAME : "RD01"
    AREA TYPE : USER
    RESOURCE  : TBL : "user1"."TBL1"
                      "user2"."TBL2"
                      "user1"."TBL3"
                IDX : "user1"."IDX1"
                      ("user1"."TBL1")
                      "user1"."IDX3"
                      ("user1"."TBL3")
```

```
  RDAREA NAME : "RD04"                    ┐
    AREA TYPE : USER                      │
    RESOURCE  : TBL : "user1"."TBL4"      ├ ※
                      -><"user1"."TBL1"> │
                IDX : "user1"."IDX4"      │
                     ("user1"."TBL4")     ┘
  RDAREA NAME : "LOB01"
    AREA TYPE : USER_LOB
    RESOURCE  : LOB : "COL1"
                      ("user1"."TBL1")
```

注※　-c ref を指定した場合だけ表示されます。

- pdrdrefls -e org -r RD03 -a の場合

```
SERVER NAME : BES2
  RDAREA NAME : "RD03"
    AREA TYPE : USER
    RESOURCE  : TBL : "user1"."TBL3
                IDX : "user1"."IDX3"
                      ("user1"."TBL3")
```

- pdrdrefls -e org -r RD04 -a の場合

```
SERVER NAME : BES1
  RDAREA NAME : "RD04"
    AREA TYPE : USER
    RESOURCE  : TBL : "user1"."TBL4"
                      -><"user1"."TBL1"> ※
                IDX : "user1"."IDX4"
                      ("user1"."TBL4")
  RDAREA NAME : "RD01"                    ┐
    AREA TYPE : USER                      │
    RESOURCE  : TBL : "user1"."TBL1"      │
                      "user2"."TBL2"      │
                      "user1"."TBL3"      │
                IDX : "user1"."IDX1"      │
                      ("user1"."TBL1")    │
                      "user1"."IDX3"      │
                      ("user1"."TBL3")    │
  RDAREA NAME : "RD02"                    ├ ※
    AREA TYPE : USER                      │
    RESOURCE  : TBL : "user2"."TBL2"      │
                      "user1"."TBL3"      │
                IDX : "user2"."IDX2"      │
                      ("user2"."TBL2")    │
                      "user1"."IDX3"      │
                      ("user1"."TBL3")    │
  RDAREA NAME : "LOB01"                   │
    AREA TYPE : USER_LOB                  │
    RESOURCE  : LOB  : "COL1"             │
                ("user1"."TBL1")          ┘
```

注※　-c ref を指定した場合だけ表示されます。

2.89　pdrels（RD エリアの閉塞解除）

2.89.1　pdrels の形式と規則

(1)　機能

指定した RD エリアの閉塞状態を解除します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdrels -r　{RDエリア名〔, RDエリア名〕…|ALL}　〔-o〕〔-W 実行監視時間〕
```

(4)　オプション

(a)　-r {RD エリア名〔，RD エリア名〕...|ALL}

RD エリア名　～＜識別子＞((1～30))

閉塞状態を解除する RD エリア名を指定します。

ALL

マスタディレクトリ用 RD エリアを除くすべての RD エリアの閉塞解除をします。

＜規則＞

1. バックアップ閉塞中でないマスタディレクトリ用 RD エリアは指定できません。
2. ユーザ用 RD エリア又はユーザ LOB 用 RD エリアを指定する場合は，データディクショナリ用 RD エリアが次のどちらかの状態でなければなりません。
   - オープン状態，かつ閉塞解除状態
   - オープン状態，かつ pdhold コマンドによる閉塞状態

(b)　-o

閉塞状態を解除した後，RD エリアをオープンします。省略した場合，RD エリアの閉塞解除だけをします。

(c)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

pdrels コマンドの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(5)　規則

1. pdrels コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdrels コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdrels コマンドは，RD エリアに対してモード PR で排他制御をします（更新可能バックアップ閉塞の解除時を除く）。そのため，指定した RD エリアにほかの更新系トランザクションがアクセスしている場合は，そのトランザクションの終了まで pdrels コマンドは待ち状態となります。
4. 共用 RD エリアに対して pdrels コマンドを実行した場合，全バックエンドサーバに排他を掛けます。このため，該当する RD エリアへのアクセスを複数同時に実行している場合，グローバルデッドロックが発生し，タイムアウトする可能性があります。グローバルデッドロックが発生した場合，再度 pdrels コマンドを実行してください。

(6)　注意事項

1. pdrels コマンドの結果は，pddbls コマンドで確認できます。
2. 更新可能バックアップ閉塞（WAIT モード）中に，更新バッファの内容が RD エリアに反映された場合，閉塞解除するときに警告メッセージ（KFPH00157-W）が出力されます。
3. pdrels コマンドのリターンコードと基本的な対処方法を次に示します。

   0，1 以外の場合，一部の RD エリアが閉塞解除していないことがあります。その状態では業務を継続できないおそれがあるため，再度 pdrels コマンドで閉塞解除してください。

   0：正常終了

   1：警告終了

   更新可能バックアップ閉塞（WAIT モード）時に，更新バッファの内容が RD エリアに反映されました。閉塞解除処理は正常終了しています。

   4：警告終了

   一部の RD エリアの処理がエラー終了しました。

   8：異常終了

   一部の RD エリアの処理がエラー終了していることがあります。

12：異常終了

エラーメッセージを表示できない事象が発生しました。

一部の RD エリアの処理がエラー終了していることがあります。シングルサーバ又はディクショナリサーバがあるホストのイベントログ中のエラーメッセージを参照して，エラーの要因を取り除いた後，再度コマンドを実行してください。イベントログ中にエラーメッセージが出力されていない場合は保守員に連絡してください。

4. 障害閉塞した RD エリアを回復していない場合，閉塞解除はできません。ただし，書き込み時の入出力エラー，ログレスモードの UAP，及びログレスモードのユティリティによる障害閉塞は解除できます。
5. データ未完状態の改竄防止表が格納されている RD エリアがコマンド閉塞している場合，その RD エリアの閉塞解除はできません。この場合，データ未完状態の改竄防止表に対して pdrorg の表へのリロードを実行すると，データ未完状態は解除されます。解除された後に，閉塞解除をしてください。

2.90　pdrplstart（HiRDB Datareplicator 連携の開始）

2.90.1　pdrplstart の形式と規則

(1)　機能

HiRDB Datareplicator 連携を開始します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdrplstart
```

(4)　オプション

なし。

(5)　規則

1. pdrplstart コマンドは，HiRDB システム内のすべてのユニット，及びすべてのサーバが稼働中のときだけ実行できます。例えば，HiRDB/パラレルサーバの場合で，一部のユニット又はサーバが停止しているときや，停止処理中のときは実行できません。ただし，停止中のユニットが縮退起動によって開始していないユニットだけの場合は，pdrplstart コマンドは実行できます。
2. pdrplstart コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdrplstart コマンドを実行する場合は，環境変数 PDDIR に HiRDB 運用ディレクトリ名が設定されている必要があります。設定されていない場合は，コマンドエラーになったときに原因を示すメッセージが出力されません。
4. HiRDB Datareplicator 連携を実行しているときに，再度 pdrplstart コマンドを入力するとエラーになります。
5. pdrplstart コマンドを実行した後，コマンド処理中の間は pdrplstart コマンド，pdrplstop コマンド，及び pdstop コマンド（-f，-z オプションは除く）は実行しないでください。
6. データベース更新中に pdrplstart コマンドを実行すると，意図したタイミングでの更新データの反映ができないことがあります。pdrplstart コマンドは，すべてのトランザクションが終了してから実行してください。また，pdrplstart コマンドが終了するまでの間，新しいトランザクションは実行しないでください。

(6) 注意事項

1. pdrplstart コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了です。
2. pdrplstart コマンドを実行しているときに通信障害が発生すると，一部のユニットが HiRDB Datareplicator 連携を開始したままの状態となり，コマンド実行時のリターンコードが 8 で異常終了することがあります。このような場合には，pdrplstop -f コマンドを実行して，すべてのユニットの HiRDB Datareplicator 連携の停止を確認してから，再度 pdrplstart コマンドを実行してください。
3. pdrplstart コマンドを実行するときには，抽出側と反映側のデータベースの整合性が保たれていなければなりません。整合性が保たれていない状態で HiRDB Datareplicator 連携を開始した場合，反映側のデータベースの内容は保証できません。
4. pdrplstart コマンドを入力してから，コマンドの処理が終了するまでは，pdrplstart コマンド及び pdrplstop コマンドを入力しないでください。
5. HiRDB Datareplicator との連動ができない状態で pdrplstart コマンドを実行すると，データ連動を開始できないユニットが，アボートコード Psjiz09 で異常終了することがあります。そのため，HiRDB Datareplicator との連動ができない状態では，pdrplstart コマンドを実行しないでください。

2.91 pdrplstop（HiRDB Datareplicator 連携の終了）

2.91.1 pdrplstop の形式と規則

(1) 機能

HiRDB Datareplicator 連携を終了します。

(2) 実行者

HiRDB 管理者が実行できます。

(3) 形式

```
pdrplstop 〔-f〕
```

(4) オプション

(a) -f

HiRDB Datareplicator のシステムログの抽出状況に関係なく，HiRDB Datareplicator 連携を強制終了する場合に指定します。停止中のユニットがある場合でも，HiRDB Datareplicator 連携を終了します。

(5) 規則

1. pdrplstop コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdrplstop コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdrplstop コマンドを実行する場合には，環境変数 PDDIR に HiRDB 運用ディレクトリ名が設定されている必要があります。設定されていない場合は，コマンドエラーになったときに原因を示すメッセージが出力されません。
4. オプションなしの pdrplstop コマンドを実行する場合，HiRDB Datareplicator でシステムログの抽出が完了しているときに HiRDB Datareplicator 連携を終了します。停止中のユニットがある場合はエラーとなります。ただし，停止中のユニットが，縮退起動によって開始されていないユニットだけの場合は，稼働中のユニットの HiRDB Datareplicator 連携を終了します。縮退起動によって開始されていないユニットは，次回ユニットを開始したときにシステムログの抽出が完了するのを待って，HiRDB Datareplicator 連携を終了します。
5. 抽出側 HiRDB が HiRDB/パラレルサーバの場合に pdrplstop -f コマンドを実行するときは，システムマネジャがあるユニットが稼働中であれば，ほかのユニットが停止中でも実行できます。この場合，pdrplstop -f コマンドの実行時に停止していたユニットは，次回ユニットを開始したときに HiRDB Datareplicator 連携を中止した状態になります。

6. HiRDB Datareplicator 連携を実行していないときに，pdrplstop コマンドを入力するとエラーになります。
7. pdrplstop コマンドを実行して HiRDB Datareplicator との連携を終了すると，抽出未完了状態のシステムログファイルは抽出完了状態に変わります。

(6) 注意事項

1. pdrplstop コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードを次に示します。

リターンコード	終了状態	内容
0	正常終了	• -f オプションなしの場合 HiRDB Datareplicator 連携を終了しました。 • -f オプションありの場合 HiRDB Datareplicator 連携の終了を指示しました。
4	異常終了	次のどちらかの理由で HiRDB Datareplicator 連携を終了できませんでした。 • 一部のユニットが停止中です（縮退起動によって開始されていないユニットは除く）。 • HiRDB Datareplicator 連携で未抽出ログがあります。
8	異常終了	何らかの障害が発生したために，HiRDB Datareplicator 連携を終了できませんでした。又は，pdrplstop コマンドを実行できない状態のときに pdrplstop コマンドを実行しました。

2. pdrplstop コマンドでは，HiRDB Datareplicator 連携対象の HiRDB サーバにトランザクションがあるかどうかを確認しません。そのため，トランザクションがないことを確認してから pdrplstop コマンドを実行してください。
3. pdrplstop コマンドの実行処理中，又は終了後に，抽出側 DB で更新をした場合，抽出側 DB と反映側 DB との間に不整合が発生します。また，pdrplstop -f コマンドを実行した場合，すべてのトランザクションが終了していないとき，又は抽出側 DB で未抽出ログがあるときには，抽出側 DB と反映側 DB との間に不整合が発生します。

   抽出側 DB と反映側 DB との間に不整合が発生した場合，pdrplstop コマンドを実行し，HiRDB Datareplicator 連携を終了してください。pdls -d rpl コマンドですべてのユニットの HiRDB Datareplicator 連携が終了したことを確認してから，HiRDB Datareplicator 連携を再開始する前に，反映側 DB を再作成してください。反映側 DB を再作成する運用手順については，マニュアル「HiRDB データ連動機能 HiRDB Datareplicator」を参照してください。
4. pdrplstop コマンドを入力してから，コマンドの処理が終了するまでは，pdrplstart コマンド，pdrplstop コマンド，及び pdstop コマンドを入力しないでください。
5. 複数のユニットで構成されている HiRDB/パラレルサーバでは，pdrplstop -f コマンドが正常終了していても，すべてのユニットの HiRDB Datareplicator 連携が停止しているとは限りません。そのため，pdrplstop コマンドが正常終了した後は，pdls コマンド（-d rpl）ですべてのユニットの HiRDB Datareplicator 連携が停止したことを確認してください。

6. HiRDB システム定義の誤りなどの理由で，HiRDB Datareplicator 連携を実行しているのにデータ連動の対象となるデータベースがないという場合でも，いったん開始した HiRDB Datareplicator 連携を中止するときには，必ず pdrplstop コマンドを実行してください。
7. HiRDB Datareplicator 連携を再開することを前提に pdrplstop コマンドを実行する場合，HiRDB Datareplicator 連携停止中に抽出側 DB の更新はしないでください。更新すると，DB 間の不整合が発生します。ただし，DB に論理的な更新を伴わないデータベースの再編成は，HiRDB Datareplicator 連携停止中に実行しても DB 間の不整合にはなりません。
8. pdrplstop コマンド実行時の条件と実行可否を次に示します。DB 間に不整合が発生した場合は，反映側 DB を再作成する必要があります。

pdrplstop コマンド実行時の条件				実行可否	
シングルサーバ又はシステムマネジャがあるユニットの状態	稼働中のバックエンドサーバに未抽出ログがあるか	停止中のバックエンドサーバがあるか	停止中のバックエンドサーバがすべて縮退起動によって開始されていないユニットにあるか	-f なし	-f あり
稼働中	ない	ない	–	○※3	○※3
		ある	ある	○※1※3	△※2
			ない	×	△※2
	ある	–	–	×	△
停止中	–	–	–	×	×

（凡例）

–：該当しません（条件が HiRDB の処理に影響を与えません）。

○：不整合を発生させないで実行できます。

△：実行できますが，未抽出システムログがあると不整合が発生します。

×：実行できません（不整合も発生しません）。

注※１

縮退起動によって開始していないユニットは，次回開始時に未抽出システムログの抽出を待ってから HiRDB Datareplicator 連携を終了します。

注※２

開始していないユニットは，次回開始時に未抽出システムログの抽出を待たないで HiRDB Datareplicator 連携を終了します。

注※３

pdrplstop コマンドの実行処理中，又は終了後に，抽出側 DB で更新をした場合，不整合が発生します。

2.92　pdsetacl（ファイルセキュリティ強化機能の設定）

2.92.1　pdsetacl の形式と規則

(1)　機能

ファイルセキュリティ強化機能に関する設定の変更と，設定変更に伴うディレクトリ・ファイルのアクセス権の変更を行います。

このコマンドの機能を次に示します。

- ファイルセキュリティ強化機能を使用するかどうかの設定
- HiRDB 管理者，HiRDB グループの設定
- HiRDB サービスのログオンアカウントの設定
- HiRDB 運用ディレクトリ，及び%PDDIR%¥tmp への推奨アクセス権の設定
- 指定したディレクトリ下のファイルへのアクセス権の設定
- システム定義の pd_tmp_directory オペランドに指定したパスへのアクセス権設定

(2)　実行者

Administrators 権限を持つユーザが実行できます。

なお，複数のサーバが存在する場合，このコマンドは各サーバマシンで実行する必要があります。

(3)　形式

```
pdsetacl   { -f on -u HiRDB管理者ユーザ名 -g HiRDBグループ名
             〔アクセス権設定対象パス名 〔アクセス権設定対象パス名〕…〕
           |-f off 〔-u HiRDB管理者ユーザ名〕
           |アクセス権設定対象パス名 〔アクセス権設定対象パス名〕…
           |-c }
```

(4)　オプション

(a)　-f {on | off}

ファイルセキュリティ強化機能を使用するかどうかを指定します。

on：ファイルセキュリティ強化機能を使用します。

off：ファイルセキュリティ強化機能を使用しません。

(b)　-u HiRDB 管理者ユーザ名

HiRDB 管理者を設定又は変更する場合に，HiRDB 管理者になるユーザの名称を「ユーザ名」又は「ドメイン名¥ユーザ名」の形式で指定します。

このオプションを指定した場合，HiRDB 管理者のパスワード入力を求める次のプロンプトを表示します。

```
HiRDB管理者名△Password:△
```

（凡例）

△：半角空白

(c)　-g HiRDB グループ名

HiRDB グループを設定又は変更する場合に，HiRDB グループの名称を「グループ名」又は「ドメイン名¥グループ名」の形式で指定します。

(d)　アクセス権設定対象パス名

アクセス権を変更するディレクトリ・ファイルを，絶対パス又は相対パスで指定します。変更する対象が複数存在する場合は，対象数分指定します。パスに空白を含む場合，ダブルクォーテーションで囲む必要があります。

なお，FAT ファイルシステム上のパスは指定できません。

設定するアクセス権の内容，及び継承関係の詳細を次の表に示します。

表 2–20　アクセス権設定方法

対象種別	アクセス権設定方法
ファイル	指定ファイルに HiRDB 運用ディレクトリのアクセス権と同じ権限を設定します。 ファイルが HiRDB 運用ディレクトリ下にあるかどうかによって，アクセス権の設定方法が異なります。アクセス権の設定方法を次に示します。 • ファイルが HiRDB 運用ディレクトリ下にある場合：上位ディレクトリのアクセス権を継承します。 • ファイルが HiRDB 運用ディレクトリ外にある場合：HiRDB 運用ディレクトリのアクセス権と同じアクセス権を設定し，上位ディレクトリとの継承関係を削除します。
ディレクトリ	指定ディレクトリ，及び指定ディレクトリ内のサブディレクトリ・ファイルに，HiRDB 運用ディレクトリのアクセス権と同じアクセス権を設定します。 指定ディレクトリのアクセス権は，次のように設定します。 • HiRDB 運用ディレクトリ下にある場合：上位ディレクトリの権限を継承します。 • HiRDB 運用ディレクトリ外にある場合：HiRDB 運用ディレクトリのアクセス権を上書きし，上位ディレクトリとの継承関係を削除します。 指定ディレクトリ内のサブディレクトリ・ファイルについては，上位ディレクトリのアクセス権を継承するように設定します。

なお，次のパスには，コマンド実行者に対してフルコントロールが与えられている必要があります。

1. アクセス権設定対象パス
   ディレクトリの場合，ディレクトリ内のサブディレクトリ，ファイルを含みます。
2. HiRDB 運用ディレクトリ
3. 1 と 2 の上位ディレクトリ

(e)　-c

システム定義に指定したパスのアクセス権限だけを変更する場合に指定します。-c 以外のオプションが指定されている場合，このオプションを省略していても，指定されたものとして仮定します。

変更対象とするシステム定義及び設定するアクセス権限の詳細は，マニュアル「HiRDB システム導入・設計ガイド」の「ファイルセキュリティ強化機能」の「設定するアクセス権」を参照してください。ただし，システム定義に指定されたパスが，HiRDB 運用ディレクトリ，又はそのサブディレクトリであった場合，コマンドはアクセス権限を設定しないで，HiRDB 運用ディレクトリの権限を継承します。

なお，システム定義に指定されたパスには，コマンド実行者に対してフルコントロールが与えられている必要があります。

(5)　規則

このコマンドで指定するオプションと，実行する機能の組み合わせを次の表に示します。

表 2–21　指定オプションと実行する機能の組み合わせ

オプション指定					実行する機能					
-f	-u	-g	アクセス権設定対象パス名	-c	HiRDB 管理者登録	HiRDB グループ登録※1	HiRDB サービスのログオンアカウント設定	運用ディレクトリのアクセス権設定	アクセス権設定対象パスのアクセス権設定	pd_tmp_directory のアクセス権設定
on	○	○	−	−	◎	◎	HiRDB 管理者	◎	×	◎※2
on	○	○	○	−	◎	◎	HiRDB 管理者	◎	◎	◎※2
off	○	−	−	−	◎	×	HiRDB 管理者	×	×	×
off	−	−	−	−	×	×	システム	×	×	×
−	−	−	○	−	×	×	×	×	ファイルセキュリティ強化機能使用時：◎ ファイルセキュリティ強化機能未使用時：×	ファイルセキュリティ強化機能使用時：◎※2 ファイルセキュリティ強化機能未使用時：×

オプション指定					実行する機能					
-f	-u	-g	アクセス権設定対象パス名	-c	HiRDB 管理者登録	HiRDB グループ登録※1	HiRDB サービスのログオンアカウント設定	運用ディレクトリのアクセス権設定	アクセス権設定対象パスのアクセス権設定	pd_tmp_directory のアクセス権設定
−	−	−	−	○	×	×	×	×	×	ファイルセキュリティ強化機能使用時：◎※2 ファイルセキュリティ強化機能未使用時：×

（凡例）

on：on を指定することを示します。

off：off を指定することを示します。

○：該当するオプションを指定していることを示します。

−：該当するオプションを指定していないことを示します。

◎：該当する機能を実行します。

×：該当する機能を実行しません。

HiRDB 管理者：HiRDB 管理者をログオンアカウントとして設定※3 することを示します。

システム：ローカルシステムアカウントをログオンアカウントとして設定することを示します。

注※1

HiRDB グループの登録時，-g オプションで指定したグループに対して必要なユーザ権利を付与します。付与するユーザ権利については，マニュアル「HiRDB システム導入・設計ガイド」の「インストール前に必要な作業」を参照してください。

注※2

システム定義が存在しないために pd_tmp_directory の情報を取得できない場合，エラーメッセージが出力されます。この場合，システム定義を作成した後，再度コマンドを実行してください。

注※3

ログオンアカウントの設定時，-u オプションで指定したアカウントに対して必要なユーザ権利を付与します。付与するユーザ権利については，マニュアル「HiRDB システム導入・設計ガイド」の「インストール前に必要な作業」を参照してください。

(6) リターンコード

コマンド実行時のリターンコードを次に示します。

0：正常終了

4：警告

一部のディレクトリ・ファイルのアクセス権設定に失敗したときのリターンコードです。

8：異常終了

コマンドが正常終了しなかった場合は，出力されたメッセージに従って対処をしてください。

2.93　pdsetfw.bat（Windows ファイアウォールの例外リスト登録）

2.93.1　pdsetfw.bat の形式と規則

(1)　機能

Windows ファイアウォールの例外リストに HiRDB を登録します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdsetfw.bat 〔”セットアップ識別子”〕
            〔-fwpro {any | プロファイル {プロファイル} …} 〕
```

(4)　オプション

(a)　“セットアップ識別子”

インストール時に指定したセットアップ識別子を引用符付きで指定します。標準セットアップを指定してインストールをした場合は，省略します。

指定したセットアップ識別子は，Windows ファイアウォールに次の形式で表示されます。

“製品名 セットアップ識別子 （～.exe)”

製品名には”HiRDB/Single Server”又は”HiRDB/Parallel Server”と表示されます。

(b)　-fwpro｛any｜プロファイル｛プロファイル｝…}

“セキュリティが強化された Windows ファイアウォール”の例外リストに登録するプロファイルの範囲を指定します。現在，OS が使用しているプロファイルの範囲と異なる例外登録を実行する場合に指定します。なお，省略すると，OS が現在使用しているプロファイルの範囲に対して，例外登録します。

any：

すべてのプロファイルに対して，例外登録します。Windows Server2008 及び Windows Server2012 の場合は，any を選択します。選択しない場合，タイムアウトや異常終了するときがあります。

プロファイル｛プロファイル｝…：

プロファイルには次のどれかを選択します。

pub：

パブリックプロファイルに対して例外登録します。OS のプロファイルを変更する予定がある場合に，事前に登録するとき，変更後のプロファイルを選択します。ただし，プロファイルを変更後，HiRDB を開始してください。

dom：

ドメインプロファイルに対して例外登録します。OS のプロファイルを変更する予定がある場合に，事前に登録するとき，変更後のプロファイルを選択します。ただし，プロファイルを変更後，HiRDB を開始してください。

pri：

プライベートプロファイルに対して例外登録します。OS のプロファイルを変更する予定がある場合に，事前に登録するとき，変更後のプロファイルを選択します。ただし，プロファイルを変更後，HiRDB を開始してください。

(5) 規則

pdsetfw.bat コマンドは，HiRDB を停止した状態で実行してください。

(6) 注意事項

pdsetfw.bat コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，それ以外の場合は異常終了です。出力される KFPC00406-E メッセージの詳細コードに従って対処してください。

2.94　pdstart（HiRDB システム・ユニット・サーバの開始）

2.94.1　pdstart の形式と規則

(1)　機能

HiRDB システム，ユニット，又はサーバを開始します。また，再開始する場合もこのコマンドを実行します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdstart 〔 {-i | -r | dbdestroy} 〕
```

(b)　HiRDB/パラレルサーバの場合

HiRDB システムの開始

```
pdstart 〔 {-i | -r | dbdestroy} 〕
```

ユニットの開始

```
pdstart {-x ホスト名 | -u ユニット識別子}　〔 {-r | dbdestroy} 〕
```

サーバの開始

```
pdstart -s サーバ名
```

フロントエンドサーバの SUSPEND 状態を解除する場合の開始

```
pdstart -a〔-sフロントエンドサーバ名〕
```

各ユニットからのユニットの開始

```
pdstart -q 〔-r〕
```

1:1 スタンバイレス型系切り替え機能使用時の代替部の開始（代替部を待機状態にする）

```
pdstart -q -c
```

影響分散スタンバイレス型系切り替え機能使用時の起動先ユニットを指定したサーバの開始

```
pdstart 〔 {-x ホスト名 | -u ユニット識別子} 〕 -s サーバ名
```

pdstart -r コマンドで HiRDB/パラレルサーバを開始した後，異常終了したシステムマネジャのユニットの強制的な開始

```
pdstart -R
```

(4) オプション

(a) {-i ｜ -r ｜ dbdestroy}

HiRDB システムの開始モードを指定します。省略した場合，前回の終了モードが正常終了であれば正常開始，前回の終了モードが強制終了，又は異常終了であれば再開始となります。再開始時には，データベースの回復処理をします。

-i：

データベース初期設定ユティリティ（pdinit）でデータベースの初期設定が終了している場合に，再度データベースを初期設定したいときに指定します。

HiRDB システムを開始すると，KFPS05050-Q メッセージを表示して初期設定モードで開始するかどうかを問い合わせます。"y"を応答すると，初期設定モードでの開始を続行します。"y"以外を応答すると，開始をキャンセルして終了します。

-r：

データベース複写ユティリティ（pdcopy）でマスタディレクトリ用 RD エリアのバックアップを取得する場合（-M x 指定）に，指定します。また，マスタディレクトリ用 RD エリアの障害などで HiRDB を開始できない場合，データベース回復ユティリティ（pdrstr）を実行するときに指定します。

dbdestroy：

再開始(pdstart オプション指定なし）で HiRDB が再開始できない場合に，強制開始するときに指定します。HiRDB システムを強制開始しようとするとき，KFPS05049-Q メッセージを最大 3 回表示して強制開始をするかどうかを問い合わせます。3 回のメッセージにすべて"y"を応答すると，強制開始をします。"y"以外を応答すると強制開始をキャンセルして終了します。このオプション指定時は，すべての RD エリア（システム用 RD エリアも含まれます）を破壊しますので，注意してください。

(b) {-x ホスト名｜-u ユニット識別子} 〔-r ｜ dbdestroy〕

ユニットを開始したい場合に指定します。

-x ホスト名 ～＜識別子＞((1～32))

ユニットを開始する場合，開始するユニットがあるホストの名称を指定します。-i オプションとは同時に指定できません。

影響分散スタンバイレス型系切り替え機能を適用している場合，指定したホストのユニットにあるホスト BES とゲスト BES を開始します。クラスタソフトウェアで実行系と指示されたサーバは実行系として開始し，待機系と指示されたサーバは受け入れ可能状態として開始します。実行系として開始したサーバは，HA グループ内のほかのユニットから見ると，受け入れ可能状態となります。

-u ユニット識別子　～＜識別子＞((4 文字))

ユニットを開始する場合，開始するユニットのユニット識別子を指定します。ただし，システムマネジャがあるユニットのユニット識別子は指定できません。また，-i オプションとは同時に指定できません。

影響分散スタンバイレス型系切り替え機能を適用している場合，指定したユニットにあるホスト BES とゲスト BES を開始します。クラスタソフトウェアで実行系と指示されたサーバは実行系として開始し，待機系と指示されたサーバは受け入れ可能状態として開始します。実行系として開始したサーバは，HA グループ内のほかのユニットから見ると，受け入れ可能状態となります。

-r 及び dbdestroy については，{-i ｜-r ｜ dbdestroy} の説明を参照してください。

(c)　-s サーバ名　～＜識別子＞((1～8))

サーバを開始する場合，開始するサーバの名称を指定します。

影響分散スタンバイレス型系切り替え機能を適用している場合の注意事項：

-u オプションを指定していない場合，HA グループ内の稼働中の全ユニットで，指定したサーバを開始します。ただし，実行系として開始するのは，クラスタソフトウェアで指示された 1 サーバだけであり，残りのサーバは受け入れ可能状態として開始します。

-u オプションを指定している場合，そのユニットの指定したサーバを開始します。クラスタソフトウェアで実行系と指示されたサーバは実行系として開始し，待機系と指示されたサーバは受け入れ可能状態として開始します。開始したサーバが実行系の場合，HA グループ内のほかのユニットから見ると，そのサーバは受け入れ可能状態となります。

影響分散スタンバイレス型系切り替え機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(d)　-a〔-s フロントエンドサーバ名〕

フロントエンドサーバの SUSPEND 状態（データディクショナリ用 RD エリアの回復待ちや，ディクショナリサーバの起動待ちなどの状態）の原因を取り除いた後に，フロントエンドサーバを起動するときに指定します。-s の指定の有無に関係なく，障害が発生したすべてのフロントエンドサーバが起動されます。

(e)　-q 〔-r〕

HiRDB/パラレルサーバでユニットを開始する場合に指定します。また，スタンバイ型系切り替え機能を適用している場合で，待機ユニットを開始する場合に指定します。

-q オプションを指定する場合，開始したいユニットに対応するホスト（システム共通定義の pdunit の-x 又は-c オペランドに指定したホスト）に直接ログインして実行します。

-q オプションを指定した場合，システム定義の pd_reduced_check_time オペランドの値（省略時は 20 分）までに，すべてのユニットを開始する必要があります。ユニットが開始できない場合，HiRDB の開始処理は中止されます。

-r については，{-i ｜-r ｜ dbdestroy} の説明を参照してください。

(f)　-q -c

1:1 スタンバイレス型系切り替え機能を使用する場合，代替部を待機状態にするときに指定します。次の場合に指定してください。

- クラスタソフトのコマンドで計画系切り替えを実行した場合，代替部は停止状態になります。この場合，障害発生に備えるために，pdstart -q -c コマンドで代替部を待機状態にします。
- 正規 BES ユニットを pdstop -u コマンドなどで停止して，対応する代替 BES ユニットの代替部が連動停止した後，再度正規 BES ユニットを pdstart -q コマンドで開始した場合，代替部は停止状態になります。この場合，障害発生に備えるために，pdstart -q -c コマンドで代替部を待機状態にします。

pdstart -q -c コマンドを実行する場合，待機状態にする代替部のホスト（システム定義の pdunit オペランドの-x に指定したホスト）に，直接ログインして実行してください。

(g)　-R

pdstart -r コマンドで HiRDB/パラレルサーバを開始した後，システムマネジャのユニットに障害が発生して異常終了した場合に使用します。このオプションを使用して HiRDB/パラレルサーバを回復する場合の対処については，マニュアル「HiRDB システム運用ガイド」の「pdstart -r コマンドで開始した場合に障害が発生したときの対処」を参照してください。

pdstart -R コマンドを実行すると，HiRDB はシステムマネジャのユニットをいったん強制的に開始し，コマンド実行時に稼働していたすべてのユニットを強制終了します。pdstart -R コマンドを実行した場合，システム構成と各ユニットの状態によって HiRDB の動作が異なります。システム構成と各ユニットの状態ごとの HiRDB の動作とリターンコードを次に示します。

<table>
<tr><th colspan="2">システム構成</th><th colspan="2">ユニットの状態</th><th rowspan="2">HiRDB の動作</th><th rowspan="2">pdstart のリターンコード</th></tr>
<tr><th>種別</th><th>ユニット数</th><th>システムマネジャ</th><th>システムマネジャ以外</th></tr>
<tr><td>HiRDB/シングルサーバ</td><td>－</td><td>－</td><td>－</td><td>pdstart コマンドはエラー終了します（KFPS05005-I メッセージを出力します）。</td><td>8</td></tr>
<tr><td rowspan="4">HiRDB/パラレルサーバ</td><td>1</td><td>－</td><td>－</td><td>pdstart コマンドはエラー終了します（KFPS05813-E メッセージの ONE_UNIT を出力します）。</td><td>8</td></tr>
<tr><td rowspan="3">2 以上</td><td rowspan="3">停止中</td><td>pdstart -r で稼働中</td><td>すべてのユニットが強制終了します。</td><td>0</td></tr>
<tr><td>pdstart -r 以外で稼働中</td><td>• システムマネジャ以外のユニットは稼働中のまま，KFPS05237-E メッセージを出力します。
• システムマネジャのユニットは強制終了します。</td><td>8</td></tr>
<tr><td>停止中</td><td>• システムマネジャ以外のユニットは停止中のままです。</td><td>0</td></tr>
</table>

システム構成		ユニットの状態		HiRDB の動作	pdstart のリターンコード
種別	ユニット数	システムマネジャ	システムマネジャ以外		
				• システムマネジャのユニットは強制終了します。	
		稼働中	−	pdstart コマンドは終了します（KFPS01853-W メッセージの NOT OFFLINE を出力します）。	8

（凡例）

−：関係しません。

(5) 規則

1. pdstart コマンドは，HiRDB システムの開始の場合は HiRDB が停止中のときだけ実行でき，ユニット又はサーバの開始の場合はシステムマネジャが稼働中のときだけ実行できます。
2. pdstart コマンドは，シングルサーバがあるサーバマシン，又はシステムマネジャがあるサーバマシンで実行してください。
3. HiRDB/パラレルサーバでシステムマネジャが異常終了した場合，pdstart コマンド（オプション指定なし）で再開始してください。
4. -i オプションを指定した場合，データベース初期設定ユティリティ以外のユティリティ及び運用コマンドは，データベース初期設定ユティリティが正常終了し HiRDB システムが開始処理完了の状態になった後に実行してください。HiRDB システムは，データベース初期設定ユティリティの処理が終了する前に実行される，ほかのユティリティ及び運用コマンドのチェックはしません。実行した場合は，その後の HiRDB システムの動作は保証されません。
5. -i オプションを実行する場合，全ユニットが停止していることを確認してください。pdstart -i コマンドを実行した際に，全ユニットが停止していないと，一部のユニットの起動を認識しないことがあります。pdls -d ust コマンドで各ユニットの稼働状態を確認し，ユニットが稼働している場合は，pdstop -z コマンドで停止してください。
6. 1:1 スタンバイレス型系切り替え機能を使用していて，代替部を待機状態にする pdstart コマンドの場合，-i オプションは指定できません。
7. -r オプションを指定した場合，データベース複写ユティリティ又はデータベース回復ユティリティを実行するだけのために，HiRDB システム又はユニットを開始します。このため，-r オプション指定で開始した場合は，データベース複写ユティリティ，データベース回復ユティリティ，pdls コマンド，及び pdstop コマンド以外は実行しないでください。実行した場合は，その後の HiRDB システムの動作は保証されません。
8. dbdestroy オプションを指定する場合，前回の HiRDB の開始後又は pdclose コマンド入力以降に更新した RD エリア（システム用 RD エリアも含まれます）はすべて破壊されます。このオプションで HiRDB を強制開始する時には，データベース回復ユティリティで破壊された RD エリアをすべて回復するか，又はすべてのデータベースを再作成する必要があります。dbdestroy オプションを指定して強制開始，

及び RD エリアの回復する場合の手順については，マニュアル「HiRDB システム運用ガイド」を参照してください。

9. dbdestroy オプションを実行する場合，全ユニットが停止していることを確認してください。pdstart dbdestroy コマンドを実行した際に，全ユニットが停止していないと，一部のユニットの起動を認識しないことがあります。pdls -d ust コマンドで各ユニットの稼働状態を確認し，ユニットが稼働している場合は，pdstop -z コマンドで停止してください。
10. 系切り替え機能を適用している場合に-r オプションを指定するときは，pdstart コマンド実行前に影響分散スタンバイレス型系切り替え機能を適用したユニット以外のユニットで，ディスクの活性化及び IP アドレスの起動をする必要があります。このとき，pdstart コマンドの-q オプションの指定によって，対処が異なります。
    - -q オプションを指定しない場合

      ディスクの接続状態及び IP アドレスの稼働状態を確認し，現用系でディスクの活性化及び IP アドレスの起動をしてください。
    - -q オプションを指定する場合

      ディスクの接続状態及び IP アドレスの稼働状態を確認し，pdstart コマンドを実行する系でディスクの活性化及び IP アドレスの起動をしてください。
11. HiRDB/パラレルサーバで，IP アドレスを引き継がない系切り替え機能を適用したユニットを pdstart -u コマンドで開始した場合，現用系ユニットが実行系として起動します。

    予備系のユニットを実行系として開始する場合は，予備系ユニットで，pdstart -q コマンドを実行してください。
12. 系切り替え機能を適用している場合に pdstart -r コマンドを実行するときは，系切り替え対象のユニット（影響分散スタンバイレス型系切り替え機能を適用したユニットは除く）に対して，あらかじめディスクの活性化及び IP アドレスの起動をしてください。

    HiRDB/パラレルサーバの場合，全ユニットを現用系で開始するかどうかで，次のように HiRDB の開始方法が変わります。なお，サーバマシン障害などで現用系が開始できない場合は，「一部又はすべてのユニットを予備系で開始する場合」でユニットごとに開始してください。
    - 全ユニットを現用系で開始する場合

      ディスクの接続状態を確認し，ディスクが現用系で活性化していない場合は，現用系でディスクの活性化をしてください。IP アドレスを引き継ぐ系切り替え機能を適用したユニットの場合は，現用系で IP アドレスが起動しているか確認し，IP アドレスが起動していないときは，現用系で IP アドレスを起動してください。すべての現用系でディスクの活性化及び IP アドレスの起動を確認した後，システムマネジャのあるユニットで pdstart -r コマンドを実行してください。
    - 一部又はすべてのユニットを予備系で開始する場合

      ディスクの接続状態を確認し，開始する系のディスクが活性化していない場合は，開始する系でディスクの活性化をしてください。IP アドレスを引き継ぐ系切り替え機能を適用したユニットの場合は，開始する系で IP アドレスが起動しているか確認し，IP アドレスが起動していないときは，開始する系で IP アドレスを起動してください。すべてのユニットでディスクの活性化及び IP アドレスの起動を確認した後，それぞれのユニットに対して pdstart -q -r コマンドを実行してください。

13. pdstart コマンドを実行した場合，データベース初期設定ユティリティ（pdinit -d コマンド）でディクショナリの初期化が必要なときは，KFPS05201-Q メッセージを表示してディクショナリの初期化完了を待ちます。
14. pdstart コマンドを実行した場合，pdvrup コマンドで HiRDB バージョンアップが必要なときは，KFPS05203-Q メッセージを表示して HiRDB バージョンアップの完了を待ちます。
15. 障害などで開始できないユニットがある場合，pd_start_skip_unit オペランドの指定値に従い KFPS05048-Q メッセージを表示して縮退起動してよいかどうかの応答を待ちます。"y"を応答すると，縮退起動します。"y"以外を応答すると，HiRDB の開始処理を中止します。

(6) 注意事項

1. pdstart コマンドのリターンコードを次に示します。

   0：正常終了。

   4：異常終了（タイムアウト）。

   8：異常終了（オプション指定不正，又は HiRDB 起動処理失敗）。
2. HiRDB 開始処理の完了有無を確認する方法については，マニュアル「HiRDB システム運用ガイド」の「開始処理完了の確認」を参照してください。

2.95　pdstbegin（統計情報の出力開始）

2.95.1　pdstbegin の形式と規則

(1)　機能

ホスト名で指定したサーバマシンにあるユニットの統計情報を，統計ログファイルに出力します。統計情報の出力を停止したい場合は，pdstend コマンドを実行します。

統計情報は，各ホストの pdstj1，又は pdstj2 という名称の統計ログファイルに出力されます。

出力した統計情報は，統計解析ユティリティの入力となります。統計解析ユティリティについては，「統計解析ユティリティ（pdstedit）」を参照してください。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdstbegin 〔-k 統計情報種別〔，統計情報種別〕…〕　〔-m システム稼働統計情報出力時間間隔〕
          〔-I データベースの入出力統計情報出力時間間隔〕
          〔-D 〔データベースの入出力統計情報オプション
            〔，データベースの入出力統計情報オプション〕〕〕
          〔 {-a | -s サーバ名〔，サーバ名〕…} 〕
          〔-w〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdstbegin 〔-k 統計情報種別〔，統計情報種別〕…〕　〔-m システム稼働統計情報出力時間間隔〕
          〔-I データベースの入出力統計情報出力時間間隔〕
          〔-D 〔データベースの入出力統計情報オプション
            〔，データベースの入出力統計情報オプション〕〕〕
          〔 {-x ホスト名 | -u ユニット識別子} 〕
          〔 {-a | -s サーバ名〔，サーバ名〕…} 〕
```

```
〔-w〕
```

(4) オプション

(a) -k 統計情報種別 ～《sys》

出力する統計情報の種別を指定します。

sys：

システムの稼働に関する統計情報

uap：

UAP に関する統計情報

sql：

SQL に関する統計情報

sqh：

SQL 文の履歴に関する統計情報（発行された操作系 SQL，定義系 SQL，及び LOCK 文と SQL に関する統計情報）

この統計情報は，SQL に関する統計情報出力が指定されている場合に出力されます。

buf：

グローバルバッファプールに関する統計情報

fil：

データベース操作に関する HiRDB ファイルの統計情報

dfw：

デファードライト処理に関する統計情報

idx：

インデクスに関する統計情報

sop：

SQL 静的最適化に関する統計情報

dop：

SQL 動的最適化に関する統計情報

pcd：

SQL オブジェクト実行に関する統計情報

obj：

SQL オブジェクト転送に関する統計情報（HiRDB/パラレルサーバの場合だけ有効）

dio：

データベースの入出力に関する統計情報

all：

sqh，pcd，及び obj を除く上記すべての統計情報

(b) -m システム稼働統計情報出力時間間隔 ～＜符号なし整数＞((1～1440))《10》（単位：分）

システムの稼働に関する統計情報（-k sys 指定）を統計ログファイルに出力する場合，情報を出力する時間間隔を分単位で指定します。

(c) -I データベースの入出力統計情報出力時間間隔 ～＜符号なし整数＞((10～3600))《60》（単位：秒）

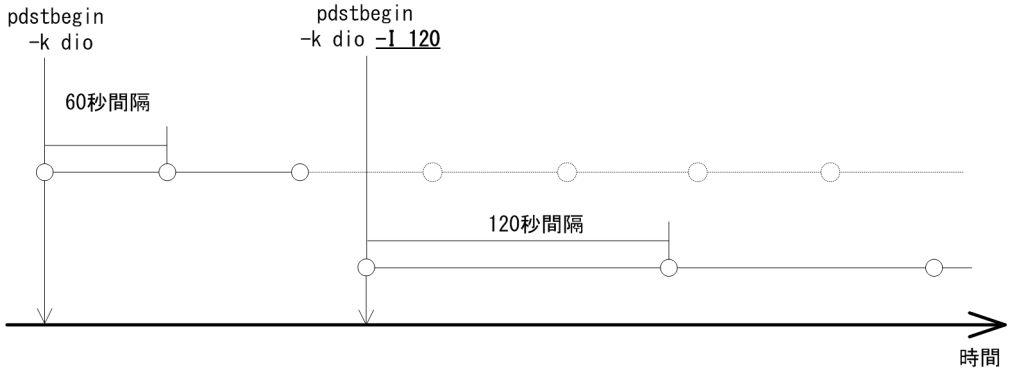
データベースの入出力に関する統計情報（-k dio 指定）を統計ログファイルに出力する場合，情報を出力する時間間隔を秒単位で指定します。

指定時間間隔でデータベースの入出力が 1 回も発生しなかった場合は，統計ログファイルを出力しません。

-k dio の指定がない場合は，このオプションを無視します。

データベースの入出力統計情報出力中に pdstbegin -I コマンドを実行すると，コマンド実行時点を新たな時間間隔の開始時刻として設定します。情報を出力する時間間隔を次の図に示します。

図 2–7 -I オプションで指定する時間間隔



統計情報出力時間間隔が短くなるに従って統計ログ出力量は増え，統計ログバッファのスワップ頻度が増すため，システム負荷が高くなります。このため，詳細な調査が必要な場合にだけ省略値より短い出力時間間隔へ変更し，情報取得後は変更前の値に戻してください。

(d) -D 〔データベースの入出力統計情報オプション〔,データベースの入出力統計情報オプション〕〕

データベースの入出力に関する統計情報の計測動作を制御するオプションです。-k dio の指定がない場合は，このオプションを無視します。

このオプションでは，次の項目を制御できます。

- 入出力時間の計測頻度

- 統計情報の出力量

各項目に対応するオプション値指定有無による動作差異を次の表に示します。

表 2–22　入出力時間の計測頻度

指定値	動作	メリット	デメリット
指定なし	RD エリアを構成する各 HiRDB ファイルに発生した入力と出力に対して，それぞれ発生順に秒間 100 回の入出力時間を計測します。	性能影響を最小限に抑えられます。	RD エリアを構成する各 HiRDB ファイルに発生した入力と出力に対して，それぞれ秒間 101 回目以降については入出力時間を計測しません。
everyio	RD エリアを構成する HiRDB ファイルに発生した入出力に対して，すべての入出力時間を計測します。	入出力情報を漏れなく取得できます。	入出力時間計測のオーバヘッドによって，トランザクション性能が劣化します。

everyio 指定時は，トランザクション実行中の RD エリアに発生するすべての入出力の時間を計測するため，オーバヘッドが増加します。事前にテストをして，本番業務に性能上問題がないと判断した場合だけ指定してください。

表 2–23　統計情報の出力量

指定値	動作	メリット	デメリット
指定なし	• 出力時間間隔内にアクセスのあった RD エリアを構成する HiRDB ファイルの入出力情報のうち，統計情報として取得する次の項目ごとに，それぞれのユニットで上位 10 ファイルの情報を統計ログに出力します。 ・read 最大回数 ・read 最大時間 ・read 合計時間 ・write 最大回数 ・write 最大時間 ・write 合計時間 • 同じファイルが複数の項目で出力対象となった場合，重複排除しないで出力します。 • 同順となるレコードが複数存在して 10 件を超える場合は，次の順序が早いものを優先します。 1.RD エリア定義順 2.HiRDB ファイル指定順	1 回の出力契機の統計ログサイズが固定であるため，ディスク消費量が少なく，統計ログバッファのスワップ処理多発による性能影響が少なくなります。	それぞれのユニットで上位 10 ファイルの情報しか出力しないため，RD エリアを構成する HiRDB ファイルすべての入出力の情報は取得できません。
everyfile	出力時間間隔内にアクセスのあった RD エリアを構成する HiRDB ファイルの入出力情報をすべて出力します。	すべての RD エリア構成ファイルの入出力傾向を漏れなく出力できます。	1 回の出力契機の統計ログサイズが固定でないため，ディスク消費量が多く，バッファ

指定値	動作	メリット	デメリット
			のスワップ処理多発による性能への影響が大きくなります。

everyfile 指定時は，1 回の出力契機での統計ログの出力量が固定ではありません。システム共通定義の pd_stj_file_size オペランドには，マニュアル「HiRDB システム定義」の「統計ログファイル（pd_stj_file_size）の見積もり式」で見積もった値を指定してください。

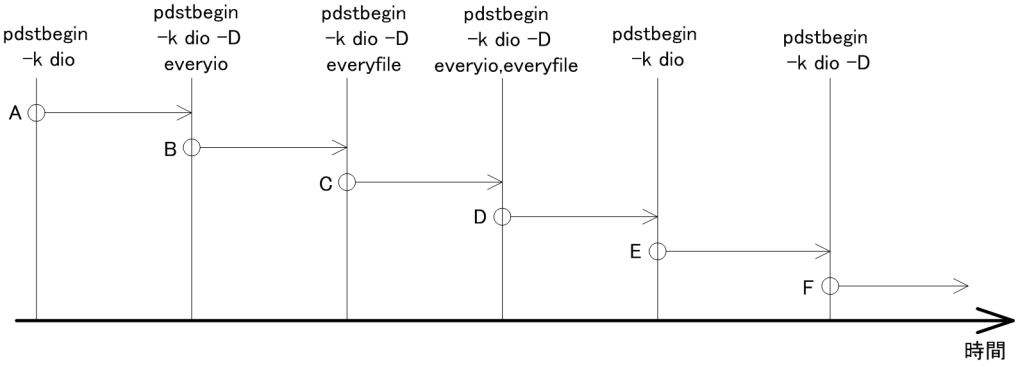
システム共通定義の pd_stj_buff_size オペランドには，マニュアル「HiRDB システム定義」の「pd_stj_buff_size オペランド」で示す見積もり式で見積もった値を指定してください。

データベースの入出力に関する統計情報出力中に計測動作を変更する場合は，「pdstbegin -k dio」コマンドを-D オプション指定で実行します。このとき，-D オプションの各項目のうち，指定しない項目は省略時の動作になります。

なお，-D オプションを省略した場合は，それまでの動作を引き継ぎます。

-D オプション指定によって計測動作を変更する例を次の図に示します。

図 2–8　-D オプションで指定する入出力統計情報の計測動作



〔説明〕

各区間の計測動作を次の表に示します。

表 2–24　各区間の計測動作

区間	各項目の動作	
	入出力時間の計測頻度	統計情報の出力量
A	指定なし	指定なし
B	everyio	指定なし
C	指定なし	everyfile
D	everyio	everyfile
E	everyio	everyfile
F	指定なし	指定なし

(e) {-x ホスト名｜-u ユニット識別子}

統計情報を出力するホストのホスト名，又はユニットのユニット識別子を指定します。省略した場合，HiRDB システム全体の統計情報を統計ログファイルに出力します。

-x ホスト名 ～＜識別子＞((1～32))

統計情報を出力するホストのホスト名を指定します。なお，1:1 スタンバイレス型系切り替え機能を使用している場合に代替中のときは，正規 BES のホスト名，又は代替 BES のホスト名のどちらを指定しても，両方のホストの統計情報出力を開始します。

-u ユニット識別子 ～＜識別子＞((4 文字))

統計情報を出力するユニットのユニット識別子を指定します。なお，1:1 スタンバイレス型系切り替え機能を使用している場合に代替中のときは，正規 BES のユニット識別子，又は代替 BES のユニット識別子のどちらを指定しても，両方のユニットの統計情報出力を開始します。

(f) {-a ｜-s 〔サーバ名〔，サーバ名〕...〕} ～《-a》

どのサーバから統計情報を出力するかを指定します。省略した場合，サーバの種別によって出力する統計情報が異なります。出力する統計情報については，表「出力される統計情報と-s オプションとの関係（HiRDB/シングルサーバの場合）」又は表「出力される統計情報と-s オプションとの関係（HiRDB/パラレルサーバの場合）」を参照してください。

-a

システム全体及び全サーバの統計情報を出力します。-s オプションと同時に指定できません。

-s 〔サーバ名〔，サーバ名〕...〕 ～＜識別子＞((1～8))

統計情報を出力するサーバの名称を指定します。-a オプションと同時に指定できません。なお，1:1 スタンバイレス型系切り替え機能を使用している場合に代替中のときは，正規 BES，又は代替 BES のどちらを指定しても，両方のサーバの統計情報出力を開始します。

(g) -w

統計情報にスレッド間ロック待ち時間の情報を取得する場合に指定します。スレッド間ロック待ち時間は，スレッド間ロック要求が待ち状態になってから，スレッド間ロックを取得するまでの時間です。

スレッド間ロック待ち時間の情報は，次の統計情報で取得できます。

- システムの稼働に関する統計情報（sys）
- UAP に関する統計情報（uap）
- グローバルバッファプールに関する統計情報（buf）

スレッド間ロック待ち時間の情報を取得する場合は，システムの稼働に関する統計情報の取得が前提条件となります。そのため，システムの稼働に関する統計情報を取得していない場合は，システムの稼働に関する統計情報を取得してください。

より正確な情報を取得するために，このコマンドを実行する直前又は直後に，pdstjsync コマンドを実行してください。詳細は，「pdstjsync（統計ログファイルへの統計ログバッファの反映）」を参照してください。

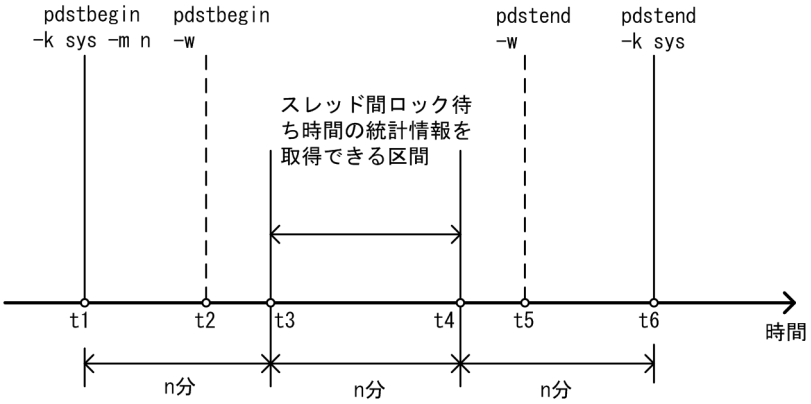
なお，スレッド間ロック待ち時間の統計情報の取得状況は，pdls -d stj コマンドを実行して確認してください。詳細は，「pdls【-d stj】（ユニット及びサーバごとの統計情報出力種別の指定有無の表示）」を参照してください。

統計情報種別ごとに，スレッド間ロック待ち時間の統計情報を取得できる条件を次に示します。

システムの稼働に関する統計情報（sys）

システムの稼働に関する統計情報とスレッド間ロック待ち時間の統計情報を取得できる区間を次の図に示します。

図 2–9　スレッド間ロック待ち時間の統計情報を取得できる区間（sys）

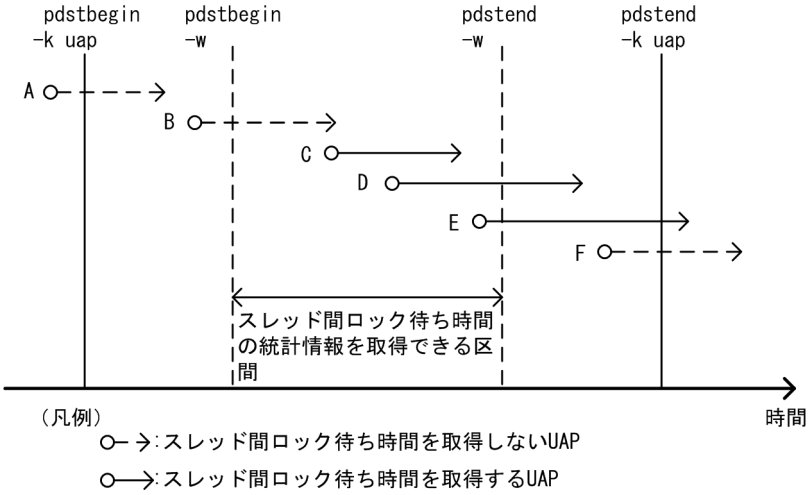


スレッド間ロック待ち時間の統計情報を正確に取得できるのは，t3～t4 の区間です。

UAP に関する統計情報（uap）

UAP に関する統計情報とスレッド間ロック待ち時間の統計情報を取得できる区間を次の図に示します。

図 2–10　スレッド間ロック待ち時間の統計情報を取得できる区間（uap）

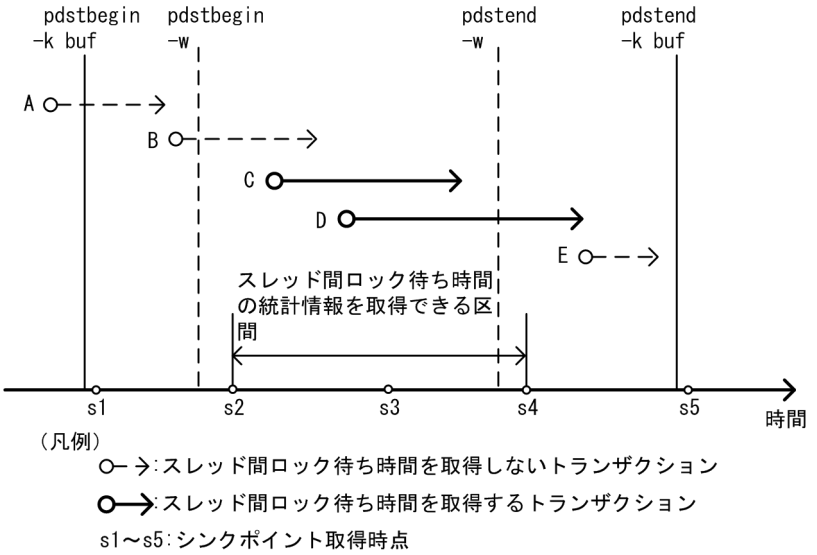


pdstbegin -w コマンド実行後から，pdstend -w コマンドが実行されるまでの間に開始された UAP のスレッド間ロック待ち時間の統計情報を取得できます。

グローバルバッファプールに関する統計情報（buf）

グローバルバッファプールに関する統計情報とスレッド間ロック待ち時間の統計情報を取得できる区間を次の図に示します。

図 2–11　スレッド間ロック待ち時間の統計情報を取得できる区間（buf）



pdstbegin -w コマンドを実行後，スレッド間ロック待ち時間の統計情報を正確に取得できるのは，直後のシンクポイントの S2 から，pdstend -w コマンド実行直後のシンクポイントの S4 までの区間です。また，トランザクション D の途中で pdstend -w コマンドが実行されていますが，トランザクションが決着するまでスレッド間ロックの待ち時間の統計情報を取得します。

ユニット内にスレッド間ロック待ち時間の統計情報を取得していないサーバがあると，システムの稼働に関する統計情報のユニットごとの編集結果は実際よりも小さい値が出力されます。そのため，システムの稼働に関する統計情報を取得するサーバを限定した場合は，サーバごとの編集結果から統計情報を評価してください。

(5)　規則

1. pdstbegin コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstbegin コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 統計情報を出力している場合は，pdstend コマンドを実行しないと，HiRDB が停止するまで統計情報を出力し続けます。
4. 出力される統計情報と-s オプションとの組み合わせによって，出力される統計情報が異なります。出力される統計情報と-s オプションとの関係（HiRDB/シングルサーバの場合）を表「出力される統計情報と-s オプションとの関係（HiRDB/シングルサーバの場合）」，（HiRDB/パラレルサーバの場合）を表「出力される統計情報と-s オプションとの関係（HiRDB/パラレルサーバの場合）」に示します。

表 2–25　出力される統計情報と-s オプションとの関係（HiRDB/シングルサーバの場合）

統計情報種別	-s オプションで指定したサーバ（シングルサーバ）
sys	○

統計情報種別	-s オプションで指定したサーバ（シングルサーバ）
uap	○
sql	○
sqh	○
buf	○
fil	○
dfw	○
idx	○
sop	○
dop	○
pcd	○
obj	×
dio	○

（凡例）

○：統計情報を出力します。

×：統計情報を出力しません。

表 2–26　出力される統計情報と-s オプションとの関係（HiRDB/パラレルサーバの場合）

統計情報種別	-s オプションで指定したサーバ		
	フロントエンドサーバ	ディクショナリサーバ	バックエンドサーバ
sys	○	○	○
uap	○	×	×
sql	○	×	×
sqh	○	×	×
buf	×	○	○
fil	×	○	○
dfw	×	○	○
idx	×	○	○
sop	○	×	×
dop	○	×	×
pcd	○	○	○
obj	×	○	○
dio	×	○	○

（凡例）

○：統計情報を出力します。

×：統計情報を出力しません。

5. 次の場合，すべてのサーバのスレッド間ロックの待ち時間の統計情報の取得を開始します。

- -a オプションを指定した場合
- -s オプション及び-a オプションの指定を省略して，-w オプションを指定した場合

(6) 注意事項

1. pdstbegin コマンドの結果は，pdls -d stj コマンド，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：異常終了（一部ユニットの統計情報取得開始に失敗している）

   一部のユニットで統計情報の出力を開始できていません。

   その状態で業務を継続すると一部のユニットで統計情報が取得されません。

   メッセージに従って対処した後，再度コマンドを実行してください。

   8：異常終了（オプション指定不正，rsh 失敗など）

   統計情報の出力を開始できていません。

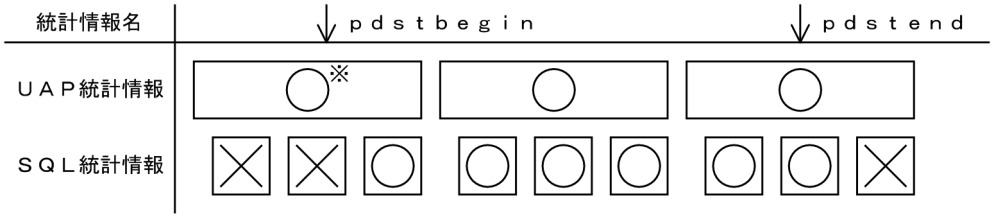
   その状態で業務を継続すると統計情報が取得されません。

   メッセージに従って対処した後，再度コマンドを実行してください。

2. pdstbegin コマンドは，コマンド名 pdststart でも実行できます。
3. ユニット内のサーバがすべて停止している場合，ユニットの統計情報は取得されません。
4. HiRDB/パラレルサーバでシステムマネジャだけのユニット（同じユニットにフロントエンドサーバ，バックエンドサーバ，又はディクショナリサーバがない）の場合，このユニットの統計情報は取得されません。
5. 影響分散スタンバイレス型系切り替え機能を適用したユニットで，ユニット内に実行系として稼働しているサーバが存在しない場合，このユニットの統計情報は取得されません。
6. サーバの sys（システムの稼働に関する統計情報）の取得を開始すると，ユニットの sys の取得も開始します。
7. HiRDB を終了し，再度開始した場合，統計情報の出力は引き継がれません。したがって，HiRDB 開始後に統計情報を出力したい場合は，再度 pdstbegin コマンドを実行してください。
8. サーバを一度終了すると，サーバの正常開始時には統計情報の出力は引き継がれません。一度終了したサーバの統計情報を出力したい場合は，サーバの正常開始後に pdstbegin コマンドを実行してください。
9. 統計情報が出力中かどうかについては，pdls -d stj コマンドを実行してください。

10. UAP に関する統計情報は，pdstbegin コマンドの入力から pdstend コマンドの入力までに実行していた UAP についてだけ出力します。また，OLTP 環境下で実行している UAP では，pdstbegin コマンドの実行から pdstend コマンドの実行までに実行したトランザクションについてだけ，UAP に関する統計情報を出力します。
11. SQL に関する統計情報は，pdstbegin コマンドの入力から pdstend コマンドの入力までに実行した SQL についてだけ出力されます。
12. UAP に関する統計情報と SQL に関する統計情報は，pdstbegin 及び pdstend コマンドを入力するタイミングによっては情報が一致しない場合があります。出力される統計情報と，pdstbegin，pdstend コマンドの入力との関係を次の図に示します。

図 2–12　出力される統計情報と，pdstbegin，pdstend コマンドの入力との関係



（凡例）　○：出力します。
　　×：出力しません。

注※ 次の場合は出力されません。
　・ＯＬＴＰ環境下のＵＡＰ

13. 各統計情報の出力契機を次に示します。

sys：

-m オプションに指定した時間間隔ごとに情報を出力します。

uap：

HiRDB への接続時及び HiRDB からの切離し時に情報を出力します。

sql：

SQL の実行開始時及び SQL の実行終了時に情報を出力します。

sqh：

PREPARE 時及び埋込み型 SQL 文の実行時に情報を出力します。

buf，fil，idx：

シンクポイント時に情報を出力します。

dfw：

シンクポイント時，及びデファードライトトリガ時に情報を出力します。

sop：

動的 SQL の前処理時，及び静的 SQL の前処理時に，SQL オブジェクトが SQL オブジェクト用バッファにヒットしなかった場合に情報を出力します。

dop：

FETCH 文及び CLOSE 文以外の SQL 文を実行したときに情報を出力します。

pcd：

SQL オブジェクトの実行時に情報を出力します。

obj：

ディクショナリサーバ，及びバックエンドサーバで SQL オブジェクトの実行要求時に情報を出力します。

dio：

-I オプションに指定した時間間隔（秒）ごとにデータベースへの入出力が発生した場合に情報を出力します。

14. 次に示す統計情報は，シンクポイント時に取得され，シンクポイント間の情報が編集されます。このため，シンクポイントが 2 回以上発生しないと統計情報が取得されません。
    - グローバルバッファプールに関する統計情報
    - データベース操作に関する HiRDB ファイルの統計情報
    - インデクスに関する統計情報

    これらの統計情報を取得するためには，pdstbegin コマンド実行直後と，pdstend コマンド実行直前に，pdlogswap コマンドを実行してシンクポイントを発生させる必要があります。
15. SQL 文の履歴に関する統計情報は，SQL に関する統計情報出力が指定されている場合に出力されます。
16. 影響分散スタンバイレス型系切り替え機能を適用している場合，pdstbegin コマンド実行後で，かつ pdstend 実行前に再開始が発生すると，再開始後も再開始前と同じ統計情報が出力されます。また，pdstbegin コマンド実行後で，かつ pdstend 実行前に系切り替えが発生すると，障害が発生したユニットで実行系として稼働していたサーバについては，切り替え先の受け入れユニットでも切り替え前と同じ統計情報が出力されます。統計情報の出力を停止する場合は，pdstend コマンドを実行してください。
17. 影響分散スタンバイレス型系切り替え機能を適用している場合，-x，又は-u オプションを指定するときは，系切り替えの前後で pdstbegin コマンドの実行対象が異なります。pdstbegin コマンドの実行対象は次のようになります。

サーバの種別	サーバの状態	実行対象
ホスト BES	実行中	○
	待機中	×
ゲスト BES	実行中	○
	受け入れ可能状態	×

（凡例）

○：実行できます。

×：実行できません。KFPS01882-E メッセージを表示します。

18. 影響分散スタンバイレス型系切り替え機能を適用している場合，-x，又は-u オプションと，-s オプションを同時に指定したときの注意事項を次に示します。
    - pdstbegin コマンド実行時点で，-x，又は-u オプションに指定したユニットの，-s オプションで指定したバックエンドサーバが稼働中の場合，統計情報の取得を開始します。

- -s オプションで指定したバックエンドサーバが-x，又は-u オプションで指定したユニットで稼働していない場合，KFPS01882-E メッセージを表示して，エラー終了します。

  バックエンドサーバが稼働しているユニットを pdls -d svr コマンドで確認した後，pdstbegin コマンドを再実行してください。

19. pdstbegin -w コマンドで，スレッド間ロックの待ち時間の統計情報を取得すると，システム全体の性能に影響を与えるおそれがあります。そのため，通常は指定しないでください。

2.96　pdstend（統計情報の出力停止）

2.96.1　pdstend の形式と規則

(1)　機能

ホスト名（又はユニット識別子）で指定したサーバマシンにあるユニットの，指定した統計情報の出力を停止します。pdstend コマンドを実行しないと，HiRDB が停止するまで統計情報を出力し続けます。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdstend 〔-k 統計情報種別 〔,統計情報種別〕…〕
          〔-a | -s サーバ名 〔サーバ名〕…〕
          〔-w〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdstend 〔-k 統計情報種別 〔,統計情報種別〕…〕
          〔 {-x ホスト名 | -u ユニット識別子} 〕
          〔-a | -s サーバ名 〔サーバ名〕…〕
          〔-w〕
```

(4)　オプション

(a)　-k 統計情報種別

出力を停止する統計情報の種別を指定します。このオプションを省略すると，すべての統計情報の出力を終了します。ただし，-w オプションを指定している場合は，スレッド間ロック待ち時間の統計情報だけ取得を停止します。

sys：

システムの稼働に関する統計情報

スレッド間ロック待ち時間を取得している場合は，スレッド間ロック待ち時間の統計情報の取得も停止します。

uap：

UAP に関する統計情報

sql：

SQL に関する統計情報

sqh：

SQL 文の履歴に関する統計情報（発行された操作系 SQL，定義系 SQL，及び LOCK 文と SQL に関する統計情報）

この統計情報は，SQL に関する統計情報出力が指定されている場合に出力されます。

buf：

グローバルバッファプールに関する統計情報

fil：

データベース操作に関する HiRDB ファイルの統計情報

dfw：

デファードライト処理に関する統計情報

idx：

インデクスに関する統計情報

sop：

SQL 静的最適化に関する統計情報

dop：

SQL 動的最適化に関する統計情報

pcd：

SQL オブジェクト実行に関する統計情報

obj：

SQL オブジェクト転送に関する統計情報（HiRDB/パラレルサーバの場合だけ有効）

dio：

データベースの入出力に関する統計情報

all：

sqh，pcd，及び obj を除く上記すべての統計情報

(b) {-x ホスト名｜-u ユニット識別子}

統計情報を出力するホストのホスト名，又はユニットのユニット識別子を指定します。省略した場合，HiRDB システム全体の統計情報の出力を停止します。

-x ホスト名　～＜識別子＞((1～32))

統計情報を出力するホストのホスト名を指定します。なお，1:1 スタンバイレス型系切り替え機能を使用している場合に代替中のときは，正規 BES のホスト名，又は代替 BES のホスト名のどちらを指定しても，両方のホストの統計情報出力を停止します。

-u ユニット識別子　～＜識別子＞((4 文字))

統計情報を出力するユニットのユニット識別子を指定します。なお，1:1 スタンバイレス型系切り替え機能を使用している場合に代替中のときは，正規 BES のユニット識別子，又は代替 BES のユニット識別子のどちらを指定しても，両方のユニットの統計情報出力を停止します。

(c)　{-a｜-s〔サーバ名〔，サーバ名〕...〕}　～《-a》

統計情報の出力を停止するサーバを指定します。省略した場合，-a が仮定されます。

-a

HiRDB システム及び全サーバの統計情報の出力を停止します。-s オプションと同時に指定できません。

-s〔サーバ名〔，サーバ名〕...〕　～＜識別子＞((1～8))

統計情報の出力を停止するサーバの名称を指定します。-a オプションと同時に指定できません。なお，1:1 スタンバイレス型系切り替え機能を使用している場合に代替中のときは，正規 BES，又は代替 BES のどちらを指定しても，両方のサーバの統計情報出力を停止します。

(d)　-w

スレッド間ロック待ち時間の統計情報取得を停止します。-k オプションを省略し，-w オプションを指定した場合，スレッド間ロック待ち時間の統計情報の取得は停止しますが，システムの稼働に関する統計情報は取得され続けます。また，-k オプションに sys，若しくは all を指定するか，又は-k オプションを省略した場合，-w オプションの指定有無に関係なく，システムの稼働に関する統計情報の取得を停止し，スレッド間ロック待ち時間の統計情報の取得も停止します。

スレッド間ロック待ち時間の統計情報の取得状況は，pdls -d stj コマンドを実行して確認してください。システムの稼働に関する統計情報とスレッド間ロック待ち時間の統計情報を取得できる区間は，「pdstbegin（統計情報の出力開始）(4)オプション」を参照してください。

より正確な情報を取得するために，このコマンドを実行する直前又は直後に，pdstjsync コマンドを実行してください。詳細は，「pdstjsync（統計ログファイルへの統計ログバッファの反映）」を参照してください。

(5)　規則

1. pdstend コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstend コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 統計情報を出力している場合は，pdstend コマンドを実行しないと，HiRDB が停止するまで統計情報を出力し続けます。

4. スレッド間ロック待ち時間の統計情報の取得を停止する前に，統計情報の取得を停止する場合は，統計情報の取得停止に合わせてスレッド間ロック待ち時間の統計情報の取得を停止してください。
5. 次に示す場合には，すべてのサーバのスレッド間ロック待ち時間の統計情報の取得を停止します。
   - -a オプションを指定した場合
   - -s オプション又は-a オプションを指定しないで，-w オプションを指定した場合

(6) 注意事項

1. pdstend コマンドの結果は，pdls -d stj コマンド，エラーメッセージの有無，及びコマンド実行時のリターンコードで確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：異常終了（一部ユニットの統計情報の出力停止に失敗している）

   一部のユニットで統計情報の停止に失敗しており，統計情報の取得が継続していることがあります。

   メッセージに従って対処したあと再度コマンドを実行してください。

   8：異常終了（オプション指定不正，rsh 失敗など）

   統計情報の停止に失敗しています。

   メッセージに従って対処した後，再度コマンドを実行してください。
2. サーバの sys（システムの稼働に関する統計情報）の取得をすべて停止すると，ユニットの sys の取得も停止します。
3. SQL 文の履歴に関する統計情報は，SQL に関する統計情報出力が指定されている場合に出力されます。
4. 影響分散スタンバイレス型系切り替え機能を適用している場合，pdstend 実行後に再開始が発生すると，再開始後は統計情報が出力されません。また，pdstend 実行後に系切り替えが発生すると，障害が発生したユニットで実行系として稼働していたサーバについては，切り替え先の受け入れユニットで統計情報は出力されません。統計情報の出力を再開する場合は，pdstbegin コマンドを実行してください。
5. 影響分散スタンバイレス型系切り替え機能を適用している場合，-x，又は-u オプションを指定するときは，系切り替えの前後で pdstend コマンドの実行対象が異なります。pdstend コマンドの実行対象は次のようになります。

サーバの種別	サーバの状態	実行対象
ホスト BES	実行中	○
	待機中	×
ゲスト BES	実行中	○
	受け入れ可能状態	×

（凡例）

○：実行できます。

×：実行できません。KFPS01882-E メッセージを表示します。

6. 影響分散スタンバイレス型系切り替え機能を適用している場合，-x，又は-u オプションと，-s オプションを同時に指定したときの注意事項を次に示します。

   - pdstend コマンド実行時点で，-x，又は-u オプションに指定したユニットの，-s オプションで指定したバックエンドサーバが稼働中の場合，統計情報の出力を停止します。
   - -s オプションで指定したバックエンドサーバが-x，又は-u オプションで指定したユニットで稼働していない場合，KFPS01882-E メッセージを表示して，エラー終了します。

     バックエンドサーバが稼働しているユニットを pdls -d svr コマンドで確認した後，pdstend コマンドを再実行してください。

2.97　pdstjswap（統計ログファイルの切り替え）

2.97.1　pdstjswap の形式と規則

(1)　機能

統計ログファイルをスワップし，障害が発生した統計ログファイルの管理情報を回復します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

```
pdstjswap 〔 {-x ホスト名 | -u ユニット識別子} 〕
```

(b)　HiRDB/パラレルサーバの場合

```
pdstjswap {-x ホスト名 | -u ユニット識別子}
```

(4)　オプション

(a)　-x ホスト名　～＜識別子＞((1～32))

統計ログファイルがあるユニットのホスト名を指定します。

(b)　-u ユニット識別子　～＜識別子＞((4 文字))

統計ログファイルがあるユニットのユニット識別子を指定します。

(5)　規則

1. pdstjswap コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstjswap コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6)　注意事項

1. pdstjswap コマンドのリターンコードを次に示します。
   0：正常終了
   8：異常終了（オプション指定不正，rsh 失敗など）

2. pdstjswap コマンドを実行すると，1 世代前の統計ログファイルが上書きされます。1 世代前の統計ログファイルの情報が必要な場合は，pdstjswap コマンドを実行する前に統計ログファイルのバックアップを取得してください。

2.98　pdstjsync（統計ログファイルへの統計ログバッファの反映）

2.98.1　pdstjsync の形式と規則

(1)　機能

統計ログバッファに保存されている統計情報（統計ログレコード）を統計ログファイルに出力します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdstjsync 〔-m〕
```

(4)　オプション

(a)　なし

統計ログバッファに保存されている統計情報を統計ログファイルに出力します。

また，現在取得中のシステムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報を統計ログバッファに出力し，その延長で統計ログファイルに出力します。

(b)　-m

統計ログバッファに保存されている統計情報を統計ログファイルに出力します。

なお，現在取得中のシステムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報は，統計ログバッファに出力されません。統計ログファイルにも出力されません。

オプション指定なしの場合と，-m オプションを指定した場合の処理の違いを次に示します。

統計情報の条件	HiRDB の処理	
	オプション指定なしの場合	-m オプションを指定した場合
統計ログバッファに保存されている統計情報の扱い	統計ログバッファに保存されている統計情報を統計ログファイルに出力します。	
現在取得中のシステムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報の扱い	現在取得中のシステムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報を統計ログバッファに出力し，その延長で統計ログファイルに出力します。	現在取得中のシステムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報は，統計ログバッファに出力されません。統計ログファイルにも出力されません。

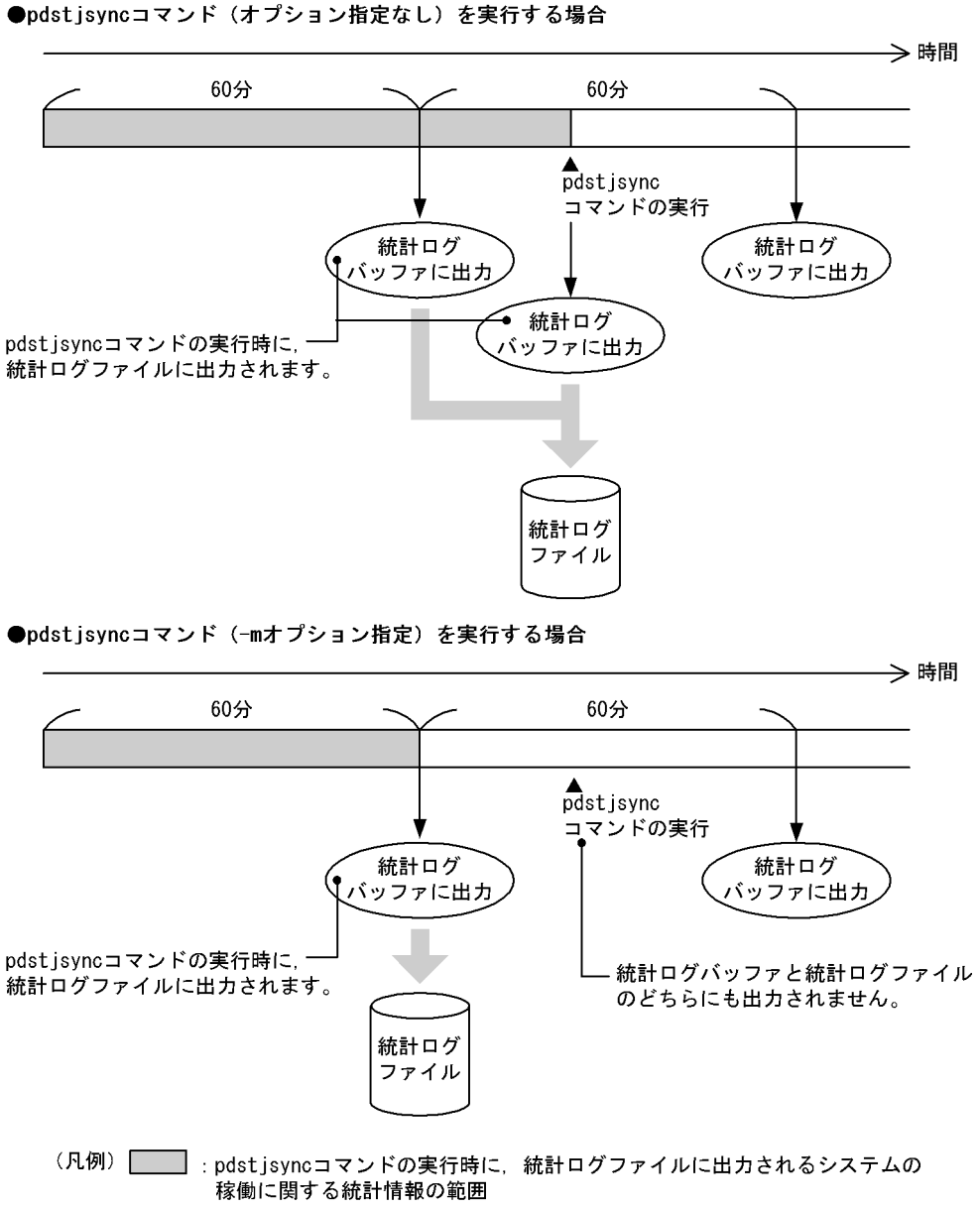
通常，システムの稼働に関する統計情報は，pdstbegin コマンドの-m オプションに指定した間隔で統計ログバッファに出力されます。例えば，pdstbegin コマンドの-m オプションに 60 分を指定した場合，60 分ごとにシステムの稼働に関する統計情報が統計ログバッファに出力されます。

オプション指定なしの pdstjsync コマンドを実行すると，この間隔とは別にシステムの稼働に関する統計情報を統計ログバッファに出力し，その延長で統計ログファイルに出力します。

-m オプションを指定して pdstjsync コマンドを実行すると，システムの稼働に関する統計情報は統計ログバッファに出力されないため，統計ログファイルにも出力されません。この場合，pdstbegin コマンドの-m オプションに指定した間隔で統計ログバッファに出力されます。

システムの稼働に関する統計情報の統計ログファイルへの出力の仕組みを次の図に示します。なお，この図では，pdstbegin コマンドの-m オプションに 60 分を指定したとしています。

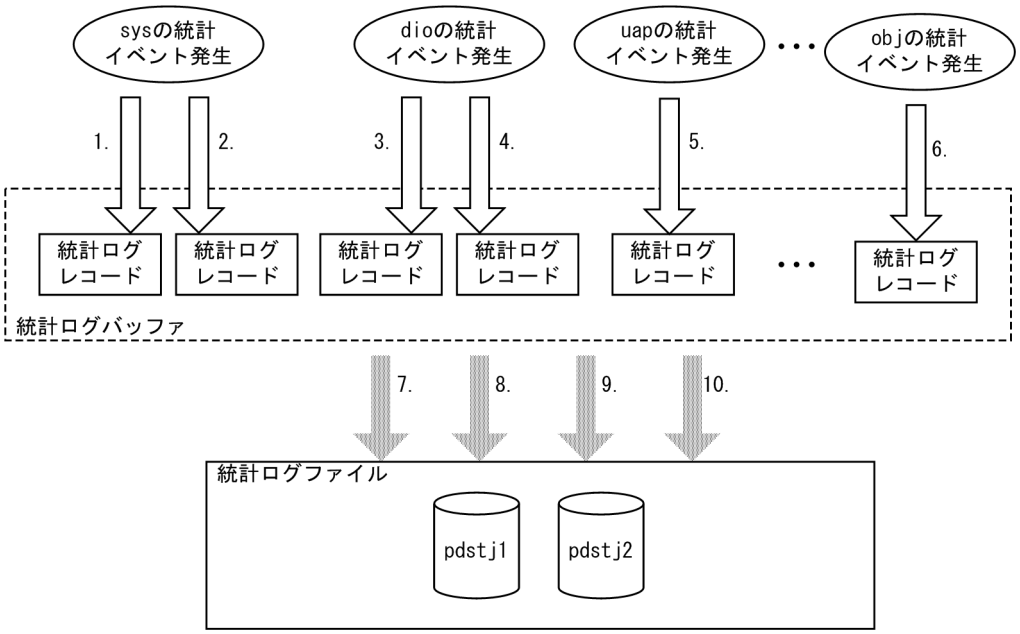
図 2–13　システムの稼働に関する統計情報の統計ログファイルへの出力の仕組み



(5) 規則

1. pdstjsync コマンドは，HiRDB が稼働中で，かつ統計情報取得中のときだけ実行できます。
2. pdstjsync コマンドは，任意のサーバマシン（統計ログバッファの情報を統計ログファイルに出力したいユニットのサーバマシン）で実行してください。
3. システムの稼働に関する統計情報，又はデータベースの入出力に関する統計情報を取得中の場合，pdstjsync コマンド実行時点のこれらの統計情報についても統計ログファイルに出力します。統計レコード出力タイミング，及び統計ログファイルへの出力タイミングを次の図に示します。

図 2–14　統計レコード出力タイミング，及び統計ログファイルへの出力タイミング



（凡例）

sys：システムの稼働に関する統計情報

dio：データベースの入出力に関する統計情報

uap：UAP に関する統計情報

obj：SQL オブジェクト転送に関する統計情報

［説明］

1.pdstbegin -m コマンドで指定した時間間隔ごとに，統計ログバッファに統計ログレコードを出力します。

2.pdstjsync コマンド実行時に，統計ログバッファに統計ログレコードを出力します。ただし，-m オプションを指定した場合はこの処理を行いません。

3.pdstbegin -l コマンドで指定した時間間隔ごとに，統計ログバッファに統計ログレコードを出力します。

4.pdstjsync コマンド実行時に，統計ログバッファに統計ログレコードを出力します。ただし，-m オプションを指定した場合はこの処理を行いません。

5.HiRDB への接続時，及び HiRDB からの切り離し時に，統計ログバッファに統計ログレコードを出力します。

6.SQL オブジェクトの実行要求時に，統計ログバッファに統計ログレコードを出力します。

7.統計ログバッファ満杯時に，統計ログバッファの情報を統計ログファイルに出力します。

8.統計情報取得の停止時（pdstend コマンド実行時）に，統計ログバッファの情報を統計ログファイルに出力します。

9.pdstop コマンド実行時（正常停止，及び計画停止）に，統計ログバッファの情報を統計ログファイルに出力します。

10.pdstjsync コマンド実行時に，統計ログバッファの情報を統計ログファイルに出力します。

(6) 注意事項

1. システムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報の取得中に pdstjsync コマンド（オプション指定なし）を実行すると，統計ログファイルに出力する統計ログの量が増えます。そのため，システム定義の pd_stj_file_size オペランドの指定値を見積もる場合には，pdstjsync コマンドの実行回数も考慮する必要があります。なお，システムの稼働に関する統計情報，及びデータベースの入出力に関する統計情報以外の統計情報については，統計ログの量が増えることはありません。
2. pdstjsync コマンドを実行しても，pdstedit の統計情報の編集には影響ありません。
3. pdstjsync コマンドのリターンコードは，0 の場合は正常終了，8 の場合は異常終了（オプション指定不正，又は統計情報未取得）となります。

2.99　pdstop（HiRDB システム・ユニット・サーバの終了）

2.99.1　pdstop の形式と規則

(1)　機能

HiRDB システム，ユニット，又はサーバを終了します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　正常終了

HiRDB/シングルサーバの場合

```
pdstop 〔-d〕
```

HiRDB/パラレルサーバの場合

```
pdstop 〔 {-x ホスト名 | -u ユニット識別子 | -s サーバ名
          | -u ユニット識別子 -s サーバ名} 〕 〔-d〕
```

(b)　計画停止

```
pdstop -P 〔-d〕
```

(c)　強制終了

HiRDB/シングルサーバの場合

```
pdstop -f 〔 {-d | -q} 〕
```

HiRDB/パラレルサーバの場合

```
pdstop {-f 〔 {-x ホスト名 | -u ユニット識別子 | -s サーバ名
              | -u ユニット識別子 -s サーバ名} 〔-d〕 〕
        | -z 〔 {-d | -q | -c | -s サーバ名} 〕 }
```

(4) オプション

(a) -P

計画停止の場合に指定します。

計画停止と正常終了の違いを次に示します。

HiRDB への CONNECT	トランザクション	正常終了	計画停止
なし	なし	○	○
	あり	×	×
あり	なし	×	○
	あり	×	×

（凡例）

○：実行できます。

×：実行できません。

(b) -f

強制終了する場合に指定します。ただし，-s オプションと同時に指定している場合は，影響分散スタンバイレス型系切り替え機能を適用しているユニットに配置したバックエンドサーバ以外のサーバ名を指定すると，正常終了となります。

(c) {-x ホスト名｜-u ユニット識別子｜-s サーバ名}

ユニット又はサーバを終了する場合に指定します。省略した場合，HiRDB システムを終了します。

-x ホスト名 ～＜識別子＞((1～32))

ユニットを終了する場合，終了するユニットがあるホストの名称を指定します。ただし，システムマネジャがあるユニットのホスト名は指定できません。

影響分散スタンバイレス型系切り替え機能を適用している場合，指定したホストのユニットにあるホスト BES とゲスト BES を終了します。終了したサーバが実行系の場合，HA グループ内のほかのユニットからの受け入れ可能状態が解除されます。

-u ユニット識別子 ～＜識別子＞((4 文字))

ユニットを終了する場合，終了するユニットのユニット識別子を指定します。ただし，システムマネジャがあるユニットのユニット識別子は指定できません。

影響分散スタンバイレス型系切り替え機能を適用している場合，指定したユニットにあるホスト BES とゲスト BES を終了します。終了したサーバが実行系の場合，HA グループ内のほかのユニットからの受け入れ可能状態が解除されます。

-s サーバ名　～＜識別子＞((1～8))

サーバを終了する場合，終了するサーバの名称を指定します。

影響分散スタンバイレス型系切り替え機能を適用している場合：

-u オプションを指定していない場合，HA グループ内の稼働中の全ユニットで，指定したサーバを終了します。

-u オプションを指定している場合，そのユニットの指定したサーバを終了します。終了したサーバが実行系の場合，HA グループ内のほかのユニットからの受け入れ可能状態が解除されます。

影響分散スタンバイレス型系切り替え機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(d)　- z

HiRDB/パラレルサーバで，システムマネジャが異常終了して再開始できないときに，システムマネジャ以外のユニットに直接ログインし，そのユニットを強制終了する場合に指定します。

HiRDB システム開始中，又は正常終了中に，誤って-z オプションを指定した pdstop コマンドを実行した場合は，ユニットが異常終了したり，コマンドが実行されなかったりします。

影響分散スタンバイレス型系切り替え機能を適用している場合，-s オプションを指定すると，該当するユニットの指定したサーバを強制終了できます。

(e)　-d

終了と同時に共用メモリダンプを出力する場合に指定します。ただし，同時に-s オプションを指定した場合，このオプションの指定は無視されます。共用メモリダンプは，%PDDIR%¥spool¥pdshmdump¥shmdump に出力します。

(f)　-q

系切り替え機能を使用している場合に強制終了するときに指定します。このオプションは，システム定義の pd_ha オペランドに use を指定していて，かつ pd_ha_unit オペランドを省略している場合に指定してください。

(g)　-c

1:1 スタンバイレス型系切り替え機能を使用している場合に，代替部を強制終了するときに指定します。次の場合に指定してください。

- Hitachi HA Toolkit Extension を使用している場合，代替中に系を戻すときに指定します（Hitachi HA Toolkit Extension の termcommand オペランドに pdstop -q -c コマンドを指定します）。
- 代替 BES ユニットを停止しないで，代替部だけを強制終了する場合に指定します。なお，代替中の場合は，システムマネジャがあるユニットから「pdstop -f -u 正規 BES ユニットのユニット識別子」又は「pdstop -f -x 正規 BES ユニットのホスト名」を実行しても，代替部を強制終了できます。

pdstop -q -c コマンドを実行する場合，強制終了する代替部のホスト（システム定義の pdunit オペランドの-x に指定したホスト）に，直接ログインして実行してください。

(5) 規則

1. pdstop コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstop コマンドは，シングルサーバ，又はシステムマネジャがあるサーバマシンで実行してください。
3. pdstart コマンドに-r オプションを指定して開始した場合は，-x オプションでユニット単位に終了できません。この場合は，オプションを指定しないか，又は-f オプションを指定して HiRDB システムを終了してください。
4. pdstart コマンドに-i オプションを指定して開始した場合，データベース初期設定ユティリティの処理の終了後であれば，-x オプションでユニット単位に終了できます。ただし，データベース初期設定ユティリティの処理の終了前の場合は，-x オプションでユニット単位に終了できません。この場合は，-f オプションを指定して HiRDB システムを終了してください。
5. -z オプション指定時は，該当するユニットの強制停止だけをして，そのほかのオプションは無視されます。
6. pdstop コマンド，又は pdstop -P コマンドで HiRDB を停止している場合，pdstop -f -x ホスト名，又は pdstop -f -u ユニット識別子による強制終了はできません。
7. pdrplstart，及び pdrplstop コマンド実行中は，pdstop コマンド（-f，及び-z オプション指定は除く）は実行しないでください。
8. 運用コマンド，又はユティリティを実行している場合，pdstop コマンドは正常終了できないことがあります。なお，pdstop コマンドでの計画停止及び強制終了はできます。

(6) 注意事項

1. pdstop コマンドのリターンコードを次に示します。

   0：正常終了

   4：異常終了（計画停止の通信タイムアウト）

   ユティリティ，又はトランザクションが実行中のため，pdstop コマンドが異常終了しました。pdstop コマンドは異常終了しましたが，計画停止処理は継続しています。

   ユティリティ，又はトランザクションの終了を待ち合わせるか，キャンセルしてください。

   8：異常終了（オプション指定不正）
2. HiRDB 停止処理の完了有無を確認する方法については，マニュアル「HiRDB システム運用ガイド」の「停止処理完了の確認」を参照してください。
3. HiRDB 開始処理中，HiRDB 終了処理中，又はトランザクション処理中に pdstop コマンド（-f 指定）を実行すると，ユニットが異常終了することがあります。
4. システムマネジャの正常動作中に，システムマネジャ以外のユニットに-z オプション指定の pdstop コマンドが入力された場合，そのユニットが異常終了したことになります。

5. HiRDB が OLTP システムと連携して動作している場合は，先に OLTP 側を終了させてから HiRDB を終了してください。先に OLTP 側を終了しないと，OLTP 側のトランザクションが処理できないで，OLTP の正常終了又は計画停止とならない場合があります。
6. pdstop -P コマンド実行後，ユティリティ実行中やトランザクション実行中のために 15 分経過しても計画停止処理が終了しない場合，計画停止処理に時間が掛かっていることを示す KFPS05072-W メッセージが表示されます。その後，計画停止処理は続行されます。
7. システム定義の pd_log_auto_unload_path オペランドを指定していて，かつ自動ログアンロード機能動作中に pdstop コマンドを実行した場合，すべてのシステムログファイルのアンロードが完了するまで HiRDB は停止できません。すぐに HiRDB を停止したい場合は，pdstop コマンド実行前に pdlogatul -t コマンドで自動ログアンロード機能を停止させてください。
8. pdstart -r コマンドで HiRDB/パラレルサーバを開始した後，システムマネジャのユニットに障害が発生して異常終了した場合，次のどちらかの方法で，一度すべてのシステムマネジャ以外のユニットを終了してください。
   - システムマネジャのユニットで，pdstart -R コマンドを実行します。
   - すべての稼働中のシステムマネジャ以外のユニットで pdstop -z コマンドを実行します。

   対処については，マニュアル「HiRDB システム運用ガイド」の「pdstart -r コマンドで開始した場合に障害が発生したときの対処」を参照してください。
9. システムの正常停止（HiRDB のサービス停止も含む）又は計画停止時に監査証跡表の自動データロード実行中だった場合，KFPS05719-W メッセージを表示して，自動データロードが完了するのを待ちます※。

   監査証跡表の自動データロード処理が完了すると，KFPS05720-I メッセージを表示して，停止処理を継続します。早急に HiRDB を停止したい場合は，pdstop の前に pdaudatld -t コマンドを実行して，監査証跡表の自動データロード機能を停止してください。既に自動データロードが実行中の場合は，pdcancel コマンドでキャンセルしてください。
10. 監査証跡表の自動データロード完了を待っている間に自動データロードがエラー終了した場合，KFPS05721-W メッセージを表示してから，停止処理を継続します。
11. ユニットの正常停止時に監査証跡表の自動データロードが実行中だった場合，コマンドエラーになることがあります。監査証跡表を配置した BES があるユニット，又は自動データロード中の監査証跡ファイルがあるユニットを正常停止しようとすると，KFPS05070-E 又は KFPS05234-E メッセージを表示します。確実にユニットを正常停止したい場合は，pdaudatld -t コマンドを実行して，監査証跡表の自動データロード機能を停止してから pdstop コマンドを実行してください。
12. サーバの正常停止時に監査証跡表の自動データロードが実行中だった場合，コマンドエラーになることがあります。監査証跡表を配置した BES を正常停止しようとすると，KFPS05071-E 又は KFPS05235-E メッセージを表示します。確実に BES を正常停止したい場合は，pdaudatld -t コマンドを実行して，監査証跡表の自動データロード機能を停止してから pdstop コマンドを実行してください。
13. インメモリ RD エリアがある場合，HiRDB を正常終了又は計画停止できません。そのため，インメモリ化を解除した後に，HiRDB を正常終了又は計画停止してください。

なお，インメモリ RD エリアがある場合でも強制終了はできますが，このとき，インメモリデータバッファ上の更新データはインメモリ RD エリアに反映されません。

14. pdstop コマンドが強制終了した場合には，OS の syslogfile に KFPS04671-I メッセージが出力されているか確認してください。KFPS04671-I メッセージが出力されている場合は，強制終了処理で異常が発生した後に，HiRDB が自動で後処理を行っていますので強制終了処理は完了しています。

注※

pdstop コマンドはシステム正常停止（オプションなし）のとき，UAP 又はユティリティが 15 分以上動作し続けると停止処理を打ち切ります。また，システム計画停止（-P オプション）のとき，トランザクション又はユティリティが 15 分以上動作し続けると KFPS05072-W メッセージを表示して警告終了します。しかし，監査証跡表の自動データロード完了を待つ場合の待ち時間はこの 15 分には含みません。

2.100　pdstscls（ステータスファイルのクローズ）

2.100.1　pdstscls の形式と規則

(1)　機能

オープン状態のステータスファイルをクローズします。このコマンドでクローズできるファイルは，HiRDB システム定義で定義したステータスファイルだけで，HiRDB が稼働中のときに有効です。現用のファイルはクローズできません。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

ユニット用ステータスファイル

```
pdstscls 〔 {-x ホスト名 | -u ユニット識別子} 〕
          {-n 論理ファイル名 | -f 物理ファイル名}
```

サーバ用ステータスファイル

```
pdstscls -s サーバ名
         {-n 論理ファイル名 | -f 物理ファイル名}
```

(b)　HiRDB/パラレルサーバの場合

ユニット用ステータスファイル

```
pdstscls {-x ホスト名 | -u ユニット識別子}
          {-n 論理ファイル名 | -f 物理ファイル名}
```

サーバ用ステータスファイル

```
pdstscls -s サーバ名
         {-n 論理ファイル名 | -f 物理ファイル名}
```

(4) オプション

(a) {-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルをクローズする場合，ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。HiRDB/シングルサーバの場合，自ホスト又は自ユニットに固定されているので省略できます。

-x ホスト名 ～＜識別子＞((1～32))

ユニット用ステータスファイルに対応するホストのホスト名を指定します。

-u ユニット識別子 ～＜識別子＞((4 文字))

ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

(b) -s サーバ名 ～＜識別子＞((1～8))

サーバ用ステータスファイルをクローズする場合，サーバ用ステータスファイルに対応するサーバの名称を指定します。

(c) -n 論理ファイル名 ～＜識別子＞((1～8))

論理ファイルごとにステータスファイルをクローズする場合，クローズする論理ファイルの名称を指定します。指定した論理ファイルを構成する A 系と B 系のファイルを両方同時にクローズします。ただし，A 系及び B 系のファイルが，次の状態でないとクローズできません。

- A 系ファイル，B 系ファイルのどちらかが BLOCKADE 又は STANDBY
- A 系ファイル，B 系ファイルの両方とも BLOCKADE 又は STANDBY

(d) -f 物理ファイル名 ～＜パス名＞((167 文字以内))

物理ファイルごとにステータスファイルをクローズする場合，クローズする物理ファイルの名称を絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。ステータスファイルに障害が発生した場合に，閉塞状態になった系のファイルではなく，STANDBY 状態の系のファイルをクローズするときに指定します。

(5) 規則

1. pdstscls コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstscls コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6) 注意事項

1. pdstscls コマンドのリターンコードを次に示します。
   0：正常終了
   8：異常終了（オプション指定不正，rsh 失敗など）

2. pdstscls コマンドの結果は，pdls コマンド（-d sts 指定）で確認できます。

2.101　pdstsinit（ステータスファイルの初期設定）

2.101.1　pdstsinit の形式と規則

(1)　機能

HiRDB ファイルシステムにステータスファイル（論理ファイル）を構成する物理ファイルを作成し，稼働中の HiRDB で使用できるように初期設定をします。稼働中の HiRDB で使用している（現用又は予備の状態）ステータスファイルの初期設定はできません。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

ユニット用ステータスファイル

```
pdstsinit 〔 {-x ホスト名 | -u ユニット識別子} 〕
          -f 物理ファイル名 〔-l レコード長〕 〔-c レコード数〕
```

サーバ用ステータスファイル

```
pdstsinit -s サーバ名
         -f 物理ファイル名 〔-l レコード長〕 〔-c レコード数〕
```

(b)　HiRDB/パラレルサーバの場合

ユニット用ステータスファイル

```
pdstsinit {-x ホスト名 | -u ユニット識別子}
          -f 物理ファイル名 〔-l レコード長〕 〔-c レコード数〕 〔-D〕
```

サーバ用ステータスファイル

```
pdstsinit -s サーバ名 〔 {-x ホスト名 | -u ユニット識別子} 〕
         -f 物理ファイル名 〔-l レコード長〕 〔-c レコード数〕 〔-D〕
```

(4) オプション

(a) {-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルの初期設定をする場合，ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。HiRDB/シングルサーバの場合，自ホスト又は自ユニットに固定されているので省略できます。

-x ホスト名　～＜識別子＞((1～32))

- ユニット用ステータスファイルの場合

  ユニット用ステータスファイルに対応するホストのホスト名を指定します。

- サーバ用ステータスファイルの場合

  影響分散スタンバイレス型系切り替え機能を適用している場合，ファイル作成先となるディスクがあるホストのホスト名を指定します。このホストには，-s オプションで指定したサーバの移動先となる，HA グループにあるユニットがなければなりません。

  -s オプションで指定したサーバ，及びシステムマネジャが稼働中の場合は，このオプションを指定しても無視されます。

  影響分散スタンバイレス型系切り替え機能を適用していない場合，このオプションと-s オプションを同時に指定すると，このオプションに指定したホストと-s オプションに指定したサーバの現用系のホストが一致しているか，チェックをします。ホスト名が一致しない場合，エラーとなります。

-u ユニット識別子　～＜識別子＞((4 文字))

- ユニット用ステータスファイルの場合

  ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

- サーバ用ステータスファイルの場合

  影響分散スタンバイレス型系切り替え機能を適用している場合，ファイル作成先となるディスクがあるホストのユニット識別子を指定します。このユニットは，-s オプションで指定したサーバの移動先となる，HA グループにあるユニットでなければなりません。

  -s オプションで指定したサーバ，及びシステムマネジャが稼働中の場合は，このオプションを指定しても無視されます。

  影響分散スタンバイレス型系切り替え機能を適用していない場合，このオプションと-s オプションを同時に指定すると，このオプションに指定したユニットの現用系のホストと，-s オプションに指定したサーバの現用系のホストが一致しているか，チェックをします。ホスト名が一致しない場合，エラーとなります。

(b) -s サーバ名　～＜識別子＞((1～8))

サーバ用ステータスファイルの初期設定をする場合，サーバ用ステータスファイルに対応するサーバの名称を指定します。

(c)　-f 物理ファイル名　～＜パス名＞((167 文字以内))

初期設定をする物理ファイルの名称を絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。また，HiRDB ファイル名に pl で始まる名称は指定できません。既に初期設定している物理ファイル名を指定した場合は，再度初期設定されます。

(d)　-l レコード長　～＜符号なし整数＞((1024～32768))《4096》

ステータスファイルのレコード長をバイト単位で指定します。レコード長は，短いとステータスファイルの入出力回数は増加しますが，使用効率が良くなります。逆に，長いとステータスファイルの入出力回数は減少しますが，使用効率が悪くなります。これらの点を考慮して，レコード長を指定してください。

(e)　-c レコード数　～＜符号なし整数＞((32～2096107))《256》

ステータスファイルのレコード数を指定します。

レコード数の見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。なお，見積もったレコード数×1.2 を目安にしてください。

(f)　-D

pdchgconf コマンドでユニットの追加，又はサーバの追加をする場合に指定します。

ユニットの追加，又はサーバの追加をする場合に必要となる，ステータスファイルの初期化をします。このオプションを指定した場合，ユニット識別子及びサーバ名のチェックはしません。

このオプションを指定して pdstsinit コマンドを実行する場合，作成するステータスファイルがあるユニットで実行してください。

(5)　規則

1. pdstsinit コマンドは，HiRDB の稼働に関係なく実行できます（ただし，HiRDB 開始処理中及び HiRDB 終了処理中は除きます）。
2. pdstsinit コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 一つの論理ファイルを構成する A 系と B 系の物理ファイルは，レコード長とレコード数が同じになるように初期設定をしてください。論理ファイルごとであれば，レコード数は同じでなくてもかまいません。ただし，レコード長はすべて同じにする必要があります。
4. ステータスファイルのレコード数は HiRDB システムの構成に依存します。ステータスファイルのレコード数は，pdls コマンド（-d sts 指定）で確認できます。そのときにレコード使用率も表示されるので，レコード使用率を参考にしてレコード数を変更してください。例えば，レコード使用率が高くなった場合，現用のファイルよりレコード数の多いステータスファイルを予備のファイルとして初期設定をしておく方法などがあります。

(6) 注意事項

1. pdstsinit コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了（オプション指定不正，rsh 失敗など），12 の場合は異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）です。
2. 影響分散スタンバイレス型系切り替え機能を適用している場合の，-u，及び-s オプションの指定方法を次に示します。

初期設定対象	ユニット又はサーバの状態	ファイル作成先のディスクがあるホスト	-u，及び-s オプションの指定方法
ユニット用ステータスファイル	稼働中	ユニットがあるホスト	どちらか一方を指定します。
	停止中		
サーバ用ステータスファイル	稼働中	実行系バックエンドサーバがあるホスト	省略できます。
	停止中	現用系バックエンドサーバがあるホスト	
		上記以外のホスト	どちらか一方を指定します。

(7) 使用例

ユニット用ステータスファイル及びサーバ用ステータスファイルを作成します。

【HiRDB/シングルサーバの場合】

```
pdstsinit -f c:¥sysarea¥rdsys011¥systs01a   ...1
          -c 512   ............................2
pdstsinit -f c:¥sysarea¥rdsys012¥systs01b   ...1
          -c 512   ............................2
pdstsinit -s sds1   ............................3
          -f c:¥sysarea¥rdsys011¥stsf01a   ....4
          -c 512   ............................5
pdstsinit -s sds1   ............................3
          -f c:¥sysarea¥rdsys012¥stsf01b   ....4
          -c 512   ............................5
```

［説明］

1. ユニット用ステータスファイルを構成する物理ファイルの名称
   c:¥sysarea¥rdsys011¥systs01a，c:¥sysarea¥rdsys012¥systs01b
2. 物理ファイルのレコード数　**512**
3. サーバ用ステータスファイルに対応するサーバの名称　**sds1**
4. サーバ用ステータスファイルを構成する物理ファイルの名称
   c:¥sysarea¥rdsys011¥stsf01a，c:¥sysarea¥rdsys012¥stsf01b
5. 物理ファイルのレコード数　**512**

【HiRDB/パラレルサーバの場合】

```
pdstsinit -x host1  .........................1
          -f c:¥sysarea¥rdsys011¥systs11a  ...2
          -c 512  .........................3
pdstsinit -x host1  .........................1
          -f c:¥sysarea¥rdsys012¥systs11b  ...2
          -c 512  .........................3
pdstsinit -x host2  .........................1
          -f c:¥sysarea¥rdsys013¥systs21a  ...2
          -c 512  .........................3
pdstsinit -x host2  .........................1
          -f c:¥sysarea¥rdsys014¥systs21b  ...2
          -c 512  .........................3
pdstsinit -x host3  .........................1
          -f c:¥sysarea¥rdsys015¥systs31a  ...2
          -c 512  .........................3
pdstsinit -x host3  .........................1
          -f c:¥sysarea¥rdsys016¥systs31b  ...2
          -c 512  .........................3
pdstsinit -s fes1  ..........................4
          -f c:¥sysarea¥rdsys011¥stsf11a  ....5
          -c 512  .........................6
pdstsinit -s fes1  ..........................4
          -f c:¥sysarea¥rdsys012¥stsf11b  ....5
          -c 512  .........................6
pdstsinit -s ds1  ..........................4
          -f c:¥sysarea¥rdsys013¥stsf21a  ....5
          -c 512  .........................6
pdstsinit -s ds1  ..........................4
          -f c:¥sysarea¥rdsys014¥stsf21b  ....5
          -c 512  .........................6
pdstsinit -s bes1  .........................4
          -f c:¥sysarea¥rdsys015¥stsf31a  ....5
          -c 512  .........................6
pdstsinit -x bes1  .........................4
          -f c:¥sysarea¥rdsys016¥stsf31b  ....5
          -c 512  .........................6
```

[説明]

1. ユニット用ステータスファイルに対応するホスト名　**host1,host2,host3**
2. ユニット用ステータスファイルを構成する物理ファイルの名称

   c:¥sysarea¥rdsys011¥systs11a

   c:¥sysarea¥rdsys012¥systs11b

   c:¥sysarea¥rdsys013¥systs21a

   c:¥sysarea¥rdsys014¥systs21b

   c:¥sysarea¥rdsys015¥systs31a

   c:¥sysarea¥rdsys016¥systs31b
3. 物理ファイルのレコード数　**512**
4. サーバ用ステータスファイルに対応するサーバ名　**fes1,ds1,bes1**

5. サーバ用ステータスファイルを構成する物理ファイルの名称

   c:¥sysarea¥rdsys011¥stsf11a

   c:¥sysarea¥rdsys012¥stsf11b

   c:¥sysarea¥rdsys013¥stsf21a

   c:¥sysarea¥rdsys014¥stsf21b

   c:¥sysarea¥rdsys015¥stsf31a

   c:¥sysarea¥rdsys016¥stsf31b

6. 物理ファイルのレコード数　**512**

2.102　pdstsopen（ステータスファイルのオープン）

2.102.1　pdstsopen の形式と規則

(1)　機能

pdstsinit コマンドで初期設定したステータスファイル，又は pdstscls コマンドでクローズしたステータスファイルをオープンします。オープンできるステータスファイルは，HiRDB システム定義で定義したステータスファイルだけで，HiRDB が稼働中のときだけ有効です。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

ユニット用ステータスファイル

```
pdstsopen 〔 {-x ホスト名 | -u ユニット識別子} 〕
            {-n 論理ファイル名 | -f 物理ファイル名}
```

サーバ用ステータスファイル

```
pdstsopen -s サーバ名
          {-n 論理ファイル名 | -f 物理ファイル名}
```

(b)　HiRDB/パラレルサーバの場合

ユニット用ステータスファイル

```
pdstsopen {-x ホスト名 | -u ユニット識別子}
            {-n 論理ファイル名 | -f 物理ファイル名}
```

サーバ用ステータスファイル

```
pdstsopen -s サーバ名
          {-n 論理ファイル名 | -f 物理ファイル名}
```

(4) オプション

(a) {-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルをオープンする場合，ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。HiRDB/シングルサーバの場合，自ホスト又は自ユニットに固定されているので省略できます。

-x ホスト名 ～＜識別子＞((1～32))

ユニット用ステータスファイルに対応するホストのホスト名を指定します。

-u ユニット識別子 ～＜識別子＞((4 文字))

ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

(b) -s サーバ名 ～＜識別子＞((1～8))

サーバ用ステータスファイルをオープンする場合，サーバ用ステータスファイルに対応するサーバの名称を指定します。

(c) -n 論理ファイル名 ～＜識別子＞((1～8))

論理ファイルごとにステータスファイルをオープンする場合，オープンする論理ファイルの名称を指定します。指定した論理ファイルを構成する A 系及び B 系のファイルを両方同時にオープンします。ただし，A 系及び B 系のファイルが，次の状態でないとオープンできません。

- A 系ファイル，B 系ファイルのどちらかが CLOSE，NONE，又は BLOCKADE
- A 系ファイル，B 系ファイルの両方とも CLOSE，NONE，又は BLOCKADE

(d) -f 物理ファイル名 ～＜パス名＞((167 文字以内))

物理ファイルごとにステータスファイルをオープンする場合，オープンする物理ファイルの名称を絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。ステータスファイルに障害が発生して閉塞状態になったステータスファイル，及び容量を拡張するために pdstsinit コマンドで初期設定した物理ファイルをオープンする場合に指定します。

(5) 規則

1. pdstsopen コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstsopen コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 現用のステータスファイルが片系運転の場合，閉塞状態の系の物理ファイルを pdstsinit コマンドで初期設定した後，pdstsopen コマンドを実行すると，現用のステータスファイルとして使用できます。

(6) 注意事項

1. pdstsopen コマンドのリターンコードを次に示します。

0：正常終了

8：異常終了（オプション指定不正，rsh 失敗など）

2. pdstsopen コマンドの結果は，pdls コマンド（-d sts 指定）で確認できます。

2.103　pdstsrm（ステータスファイルの削除）

2.103.1　pdstsrm の形式と規則

(1)　機能

HiRDB ファイルシステム下のステータスファイルを構成する物理ファイルを削除します。稼働中の HiRDB で使用している（現用又は予備の状態）の物理ファイルは削除できません。このコマンドは，障害が発生して閉塞状態になった物理ファイルを削除する場合に実行します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

ユニット用ステータスファイル

```
pdstsrm 〔 {-x ホスト名 | -u ユニット識別子} 〕 -f 物理ファイル名
```

サーバ用ステータスファイル

```
pdstsrm -s サーバ名 -f 物理ファイル名
```

(b)　HiRDB/パラレルサーバの場合

ユニット用ステータスファイル

```
pdstsrm {-x ホスト名 | -u ユニット識別子} -f 物理ファイル名 〔-D〕
```

サーバ用ステータスファイル

```
pdstsrm -s サーバ名 〔 {-x ホスト名 | -u ユニット識別子} 〕 -f 物理ファイル名 〔-D〕
```

(4)　オプション

(a)　{-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルを削除する場合，ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。HiRDB/シングルサーバの場合，自ホスト又は自ユニットに固定されているので省略できます。

-x ホスト名　～＜識別子＞((1～32))

- ユニット用ステータスファイルの場合

  ユニット用ステータスファイルに対応するホストのホスト名を指定します。

- サーバ用ステータスファイルの場合

  影響分散スタンバイレス型系切り替え機能を適用している場合，削除するステータスファイルを作成したディスクがあるホストの，ホスト名を指定します。このホストには，-s オプションで指定したサーバの移動先となる，HA グループにあるユニットがなければなりません。

  -s オプションで指定したサーバ，及びシステムマネジャが稼働中の場合は，このオプションを指定しても無視されます。

  影響分散スタンバイレス型系切り替え機能を適用していない場合，このオプションと-s オプションを同時に指定すると，このオプションに指定したホストと-s オプションに指定したサーバの現用系のホストが一致しているか，チェックをします。ホスト名が一致しない場合，エラーとなります。

-u ユニット識別子　～＜識別子＞((4 文字))

- ユニット用ステータスファイルの場合

  ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

- サーバ用ステータスファイルの場合

  影響分散スタンバイレス型系切り替え機能を適用している場合，削除するステータスファイルを作成したディスクがあるホストの，ユニット識別子を指定します。このユニットは，-s オプションで指定したサーバの移動先となる，HA グループにあるユニットでなければなりません。

  -s オプションで指定したサーバ，及びシステムマネジャが稼働中の場合は，このオプションを指定しても無視されます。

  影響分散スタンバイレス型系切り替え機能を適用していない場合，このオプションと-s オプションを同時に指定すると，このオプションに指定したユニットの現用系のホストと，-s オプションに指定したサーバの現用系のホストが一致しているか，チェックをします。ホスト名が一致しない場合，エラーとなります。

(b)　-s サーバ名　～＜識別子＞((1～8))

サーバ用ステータスファイルを削除する場合，サーバ用ステータスファイルに対応するサーバの名称を指定します。

(c)　-f 物理ファイル名　～＜パス名＞((167 文字以内))

削除する物理ファイルの名称を絶対パス名で指定します。物理ファイル名は，**HiRDB ファイルシステム領域名¥HiRDB ファイル名**で指定してください。

(d)　-D

pdchgconf コマンドでユニットの削除，又はサーバの削除をする場合に指定します。

ユニットの削除，又はサーバの削除する場合，そのユニット又はサーバのステータスファイルを削除します。このオプションを指定した場合，ユニット識別子及びサーバ名のチェックはしません。

このオプションを指定して pdstsrm コマンドを実行する場合，作成するステータスファイルがあるユニットで実行してください。

(5) 規則

1. pdstsrm コマンドは，HiRDB の稼働に関係なく実行できます（ただし，HiRDB 開始処理中及び HiRDB 終了処理中は除きます）。
2. pdstsrm コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. HiRDB 稼働中に使用していた現用のステータスファイルは削除しないでください。

(6) 注意事項

1. pdstsrm コマンドの結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，8 の場合は異常終了（オプション指定不正，rsh 失敗など），12 の場合は異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）です。
2. 影響分散スタンバイレス型系切り替え機能を適用している場合の，-u，及び-s オプションの指定方法を次に示します。

<table>
<tr><th>初期設定対象</th><th>ユニット又はサーバの状態</th><th>ファイル格納先ディスクがあるホスト</th><th>-u，及び-s オプションの指定方法</th></tr>
<tr><td rowspan="2">ユニット用ステータスファイル</td><td>稼働中</td><td rowspan="2">ユニットがあるホスト</td><td rowspan="2">どちらか一方を指定します。</td></tr>
<tr><td>停止中</td></tr>
<tr><td rowspan="3">サーバ用ステータスファイル</td><td>稼働中</td><td>実行系バックエンドサーバがあるホスト</td><td rowspan="2">省略できます。</td></tr>
<tr><td rowspan="2">停止中</td><td>現用系バックエンドサーバがあるホスト</td></tr>
<tr><td>上記以外のホスト</td><td>どちらか一方を指定します。</td></tr>
</table>

2.104　pdstsswap（ステータスファイルのスワップ）

2.104.1　pdstsswap の形式と規則

(1)　機能

ステータスファイルをスワップします。現用のステータスファイルを予備にして，予備のステータスファイルを現用にします。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

(a)　HiRDB/シングルサーバの場合

ユニット用ステータスファイル

```
pdstsswap 〔 {-x ホスト名 | -u ユニット識別子} 〕
```

サーバ用ステータスファイル

```
pdstsswap -s サーバ名
```

(b)　HiRDB/パラレルサーバの場合

ユニット用ステータスファイル

```
pdstsswap {-x ホスト名 | -u ユニット識別子}
```

サーバ用ステータスファイル

```
pdstsswap -s サーバ名
```

(4)　オプション

(a)　{-x ホスト名｜-u ユニット識別子}

ユニット用ステータスファイルをスワップする場合，ユニット用ステータスファイルに対応するホストのホスト名，又はユニットのユニット識別子を指定します。HiRDB/シングルサーバの場合，自ホスト又は自ユニットに固定されているので省略できます。

-x ホスト名　～＜識別子＞((1～32))

ユニット用ステータスファイルに対応するホストのホスト名を指定します。

-u ユニット識別子　～＜識別子＞((4 文字))

ユニット用ステータスファイルに対応するユニットのユニット識別子を指定します。

(b)　-s サーバ名　～＜識別子＞((1～8))

サーバ用ステータスファイルをスワップする場合，サーバ用ステータスファイルに対応するサーバの名称を指定します。

(5)　規則

1. pdstsswap コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdstsswap コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(6)　注意事項

1. pdstsswap コマンドのリターンコードを次に示します。

   0：正常終了

   8：異常終了（オプション指定不正，rsh 失敗など）
2. pdstsswap コマンドの結果は，pdls コマンド（-d sts 指定）で確認できます。

2.105　pdsvhostname（サーバのホスト名表示）

2.105.1　pdsvhostname の形式と規則

(1)　機能

指定したサーバのホスト名を DAT 形式で表示します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdsvhostname -s サーバ名 〔-b〕
```

(4)　オプション

(a)　-s サーバ名　～＜識別子＞((1～8))

ホスト名を表示するサーバの名称を指定します。

＜規則＞

1. HiRDB で定義されていないサーバ名を指定した場合，KFPS05041-E メッセージを出力して終了します。
2. 指定したサーバが停止中の場合（ユニットの停止も含む），ホスト名を表示しません。
3. 指定できるサーバは，シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，及びバックエンドサーバです。

(b)　-b

タイトル行を表示する場合に指定します。

省略した場合，タイトル行は表示しないで，ホスト名だけ表示します。

(5)　出力形式

- **-b 省略時**

  ```
  ”aa...a”
  ```

- **-b 指定時**

```
"HOSTNAME"
"aa...a"
```

[説明]

aa...a：

-s オプションで指定したサーバが現在稼働しているホスト名（32 文字以内）。

次のどれかのホスト名を表示します。

- システム定義の pdunit オペランドの-x，又は-c オペランドに指定したホスト名
- IP アドレス引き継ぎなしの系切り替え機能を使用している場合，現用系で稼働しているときは-x オペランドに指定したホスト名，予備系で稼働しているときは-c オペランドに指定したホスト名
- 影響分散スタンバイレス型系切り替え機能を使用している場合，指定したサーバが稼働中のホスト名
- 1:1 スタンバイレス型系切り替え機能を使用している場合，指定したサーバが代替部で稼働中のときは，代替部の稼働中のホスト名

(6) 規則

1. pdsvhostname コマンドは，HiRDB が稼働中のときだけ実行できます。停止中に実行した場合は，KFPS01853-W メッセージを出力して終了します。
2. pdsvhostname コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。システムマネジャ以外のサーバマシンから実行した場合，KFPS05037-E を出力して終了します。
3. DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

(7) 注意事項

1. pdsvhostname コマンドの実行タイミングによっては，HiRDB が停止中であっても正常終了することがあります。次のどちらかの場合に発生します。
   - システムマネジャの共用メモリが残っている。
   - サーバが停止したとき，停止したことをユニットが検知していない。
2. pdsvhostname コマンドのリターンコードを次に示します。

   0：正常終了です。

   1：指定したサーバが停止中です。

   4：HiRDB が停止中です。

   8：エラー終了（メモリ不足も含む）※，又は指定したサーバが HiRDB にありません。

   注※

   イベントログ，又は標準エラー出力に出力されているメッセージを参照して，エラー原因を取り除いた後，再度 pdsvhostname コマンドを実行してください。

2.106　pdtrndec（未決着トランザクションの強制自動決着）

2.106.1　pdtrndec の形式と規則

(1)　機能

障害発生後の HiRDB の再開始時，又は HiRDB の正常終了時に，未決着状態のトランザクションがある場合（KFPS00992-E メッセージが出力されます），このトランザクションを強制的に決着します。

未決着状態のトランザクションがあるときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdtrndec -i 入力ファイル名〔，入力ファイル名〕…
        〔-r ロールバックスクリプトファイル名〕 〔-o 出力先ディレクトリ名〕
```

(4)　オプション

(a)　-i 入力ファイル名〔，入力ファイル名〕...　～＜パス名＞((1～4096))

イベントログをテキストファイルで保存したファイルを，絶対パス名で指定します。

＜規則＞

1. ファイルを複数指定する場合は，コンマを含めて 4,096 文字以内で指定してください。
2. イベントログを保存したファイル以外のファイルを指定した場合，コマンドの実行結果は保証されません。
3. 入力ファイル名には空白を含んでいてはいけません。
4. 入力ファイル中の該当するトランザクションの情報が欠落している場合（不当な入力ファイルを指定した場合も含む），トランザクションは決着しないで決着保留となります。

(b)　-r ロールバックスクリプトファイル名　～＜パス名＞((1～4107))

事前に実行した pdtrndec コマンドの出力ファイル（pdtrnrbk.bat）を指定します。

このオプションを指定した場合は，ロールバックスクリプトファイル中で決着対象としているトランザクションだけを強制的に決着します。pdtrndec コマンドの出力ファイル以外のファイルを指定した場合，コマンドの実行結果は保証されません。

(c)　-o 出力先ディレクトリ名　～＜パス名＞((1～4096))

pdtrndec コマンドを実行した場合に出力される実行履歴（pdtrndecout），及びバッチファイル（pdtrnrbk.bat）の出力先ディレクトリ名を，絶対パス名で指定します。

＜規則＞

1. このオプションを省略した場合は，カレントディレクトリが出力先となります。
2. 指定した出力先ディレクトリに既に実行履歴，及びバッチファイルがある場合，実行履歴は追加書きされ，バッチファイルは上書きされます。
3. 同時に-r オプションを指定した場合，バッチファイルは出力されません。
4. 出力先ディレクトリ名には空白を含んでいてはいけません。

(5)　規則

1. pdtrndec コマンドは，HiRDB が稼働中のときだけ実行できます。
2. pdtrndec コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdtrndec コマンドは，障害発生後の HiRDB の再開始時，又は HiRDB の正常終了時に未決着状態のトランザクションがある場合に実行するものです。したがって，未決着状態のトランザクションがあるかどうか分からない状態で実行すると，不当にトランザクションを決着してしまう可能性があります。
4. pdtrndec コマンドは，複数同時実行しないでください。同時実行した場合，実行結果は保証されません。

(6)　注意事項

1. pdtrndec コマンドを実行する場合，回復不要 FES を除く HiRDB の全サーバが稼働中であることを確認してください。稼働していないサーバがある場合，決められた手順でサーバを開始してください。
2. pdtrndec コマンドを実行する場合，HiRDB のクライアント製品（OLTP システム，HiRDB Datareplicator など）が正常に稼働していることを確認してください。稼働していない場合は，各製品で決められた手順で開始してください。ただし，開始していないクライアント製品がある状態で強制的に決着する場合は，次の点に注意してください。
   - コマンド実行後に出力されたバッチファイル（pdtrnrbk.bat）を実行すると，コマンド実行後に決着保留としたトランザクションを強制的にロールバックします。したがって，この後に，開始していないクライアント製品を開始すると，HiRDB とクライアント製品の間で同期が取れなくなる可能性があります。
3. 入力ファイルを指定して pdtrndec コマンドを実行しても未決着状態のトランザクションが残っている場合は，バッチファイル（pdtrnrbk.bat）を指定して実行し，未決着状態のトランザクションを決着してください。

4. pdtrndec コマンドは，トランザクションのコミットコマンド，ロールバックコマンドなどの処理完了と同期するため，時間が掛かることがあります。
5. pdtrndec コマンド実行後，pdtrndec コマンドで指定した出力先ディレクトリ下のファイルは削除してください。
6. pdtrndec コマンドの結果は，pdls コマンド（-d trn 指定），コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。

   コマンド実行時のリターンコードを次に示します。

   0：正常終了

   4：警告終了（一つ以上の未決着状態のトランザクションがあります）

   メッセージに従って対処した後，再度コマンドを実行してください。

   8：異常終了

   メッセージに従って対処した後，再度コマンドを実行してください。
7. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdtrndec の入力ファイルには，BOM を持つファイルを使用できます。なお，pdtrndec の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdtrndec で出力されるファイルには BOM は付きません。

(7) 出力形式

- **実行履歴（pdtrndecout）**

```
pdtrndec result                    [1]START TIME:2002/09/05 17:20:08
[2]HOSTNAME [3]TRNGID        [4]TRNBID        [5]SVID   [6]DECISION [7]TIME
host1       HRD1unt100020b4d HRD1unt100020034 bes1      COMMIT      17:20:08
host1       HRD1unt100020b22 HRD1unt100020035 bes1      ********    --:--:--
host1       HRD1unt100020b51 HRD1unt100020038 bes1      FORGET      17:20:09
pdtrndec result                    [1]START TIME:2002/09/05 17:36:47
[2]HOSTNAME [3]TRNGID        [4]TRNBID        [5]SVID   [6]DECISION [7]TIME
host1       HRD1unt100020b22 HRD1unt100020035 bes1      ********    --:--:--
```

〔説明〕

1. コマンド開始日時（年/月/日 時:分:秒）
2. 未決着トランザクションがあったホスト名（32 文字以内）
3. 未決着トランザクションがあったグローバルトランザクション ID（16 文字）
4. 未決着トランザクションがあったブランチトランザクション ID（16 文字）
5. 未決着トランザクションがあったサーバ名（8 文字以内）
6. 未決着トランザクションが決着した種別（8 文字以内）

   種別を次に示します。

種別	意味	処置
COMMIT	強制コミット決着	なし。

種別	意味	処置
ROLLBACK	強制ロールバック決着	なし。
FORGET	強制終了	なし。
FAIL	強制コミット決着，強制ロールバック決着，又は強制終了がエラー	エラー原因を取り除いた後，再度 pdtrndec コマンドを実行してください。
********	決着保留	• HiRDB/シングルサーバの場合，及び HiRDB/パラレルサーバのシステムマネジャユニットの入力ファイルで pdtrndec コマンドを実行した場合 入力ファイルを変更して，再度 pdtrndec コマンドを実行してください。 • HiRDB/パラレルサーバのシステムマネジャがないユニットの入力ファイルで pdtrndec コマンドを実行した場合 バッチファイル（pdtrnrbk.bat）を指定して，再度 pdtrndec コマンドを実行してください。

7. 強制決着を実行した時刻（時:分:秒）

- **バッチファイル（pdtrnrbk.bat）**

```
[1]pdrbk -x host1 -s bes1 -t HRD1unt100020b22
```

〔説明〕

1. トランザクションをロールバック決着するためのコマンドフォーマット

2.107　pdtrnqing（トランザクションキューイング機能の開始，解除）

pdtrnqing コマンドは次に示す実行環境で使用できます。

- HiRDB/シングルサーバの場合，又は HiRDB/パラレルサーバで FES ユニットがある場合
- 影響分散スタンバイレス型系切り替え機能を使用している場合

実行環境によって pdtrnqing コマンドの仕様が異なるため，実行環境ごとにコマンドの説明を行います。

2.107.1　HiRDB/シングルサーバ又は FES ユニットに pdtrnqing コマンドを実行する場合

(1)　機能

トランザクションキューイングを開始又は解除します。

なお，HiRDB/パラレルサーバの場合は，次に示す条件を満たすユニットに対してだけトランザクションキューイングを適用できます（pdtrnqing コマンドを実行できます）。

1. フロントエンドサーバだけで構成されているユニット
2. システムマネジャとフロントエンドサーバだけで構成されているユニット

この条件を満たしていないユニットにはトランザクションキューイングを適用できません（pdtrnqing コマンドを実行できません）。

なお，これらの条件を満たすユニットをここでは FES ユニットと表記します。

(2)　このコマンドの目的

pdtrnqing コマンドは，スタンバイ型系切り替え機能の計画系切り替えを行う前に，新規トランザクションをキューイングしておくために使います。pdtrnqing コマンドを実行したときの HiRDB の処理をシステム構成ごとに説明します。

- **HiRDB/シングルサーバの場合**

  pdtrnqing コマンドを実行すると，新規トランザクションのキューイングが開始されて，新規トランザクションは待ち状態になります（実行中のトランザクションは処理されます）。新規トランザクションがキューイングされるため，計画系切り替えを行うときに，トランザクションエラーが発生しません。
- **HiRDB/パラレルサーバの場合**

  pdtrnqing コマンドを実行すると，コマンド実行対象ユニット（FES ユニット）の新規トランザクションのキューイングが開始されて，新規トランザクションは待ち状態になります（実行中のトランザクションは処理されます）。FES ユニットの新規トランザクションがキューイングされるため，FES ユニットの計画系切り替えを行うときに，トランザクションエラーが発生しません。

注意事項

計画系切り替えを実行する場合は，クライアント環境定義の PDAUTORECONNECT オペランドに YES を指定して，自動再接続機能を適用しておく必要があります。自動再接続機能を適用しないと，計画系切り替え中に HiRDB に接続しようとした UAP が CONNECT エラーになることがあります。

(3) 実行者

HiRDB 管理者が実行できます。

(4) 形式

```
pdtrnqing 〔-d | -t 待ち合わせ時間〕
```

(5) オプション

(a) -d

トランザクションキューイングを解除する場合にこのオプションを指定します。

(b) -t 待ち合わせ時間 ～<符号なし整数>((1～3600))《180》（単位：秒）

pdtrnqing コマンドは，実行中のトランザクション（HiRDB/パラレルサーバの場合は FES ユニットで実行中のトランザクション）が終了するまで処理を待ち合わせます。この待ち合わせ時間を秒単位で指定します。実行中のトランザクションがここで指定した時間を経過しても終了しない場合，トランザクションキューイングを中止して pdtrnqing コマンドをエラー終了します。

(6) 規則

1. HiRDB/パラレルサーバの場合，トランザクションキューイングを適用（開始又は解除）するユニットで pdtrnqing コマンドを実行します。pdtrnqing コマンドを実行したユニットだけにトランザクションキューイングが適用されます。
2. 停止中のユニットで pdtrnqing コマンドを実行できません。
3. 次に示す場合，pdtrnqing コマンドは正常終了します。この場合，トランザクションキューイングの適用状態は変わりません。
   - トランザクションキューイングの適用中に，pdtrnqing コマンドを実行して再度トランザクションキューイングを開始しようとした場合
   - トランザクションキューイング中でないときに，pdtrnqing -d コマンドを実行してトランザクションキューイングを解除しようとした場合

(7) 注意事項

1. ほかのコマンドの実行中に pdtrnqing コマンドを実行しないでください。実行すると，ほかのコマンドがエラー終了することがあります。
2. 次に示す場合に pdtrnqing コマンドを実行すると，pdtrnqing コマンドがエラー終了します。
   - 修正版 HiRDB の入れ替え（pdprgrenew コマンド）を実行している場合
   - システム構成変更コマンド（pdchgconf コマンド）を実行している場合
3. 次に示す場合に pdtrnqing コマンドでトランザクションキューイングを開始しようとすると，pdtrnqing コマンドがエラー終了します。
   - ユティリティを実行している場合
   - トランザクションキューイングの開始処理中の場合
4. 実行中のトランザクションの終了を pdtrnqing コマンドが待ち合わせている間に，次の操作を行った場合，次のコマンドの処理を優先します。このとき，pdtrnqing コマンドはキャンセルされてエラー終了します。
   - HiRDB の正常終了又は計画停止（pdstop コマンド）
   - HiRDB の強制終了（pdstop コマンド）※
   - 修正版 HiRDB の入れ替え（pdprgrenew コマンド）
   - システム構成変更コマンドの実行（pdchgconf コマンド）
   - トランザクションキューイングの解除（pdtrnqing -d コマンド）

   注※　HiRDB を強制終了する場合，pdtrnqing コマンドは強制停止（kill）されます。
5. 次に示す操作をした場合，トランザクションキューイングは自動的に解除されます。
   - トランザクションキューイングを行っているユニットを再開始した場合（ユニットを系切り替えした場合も含む）
   - システム，ユニット，サーバを終了した場合（pdstop コマンド）※1※2
   - 修正版 HiRDB の入れ替え（pdprgrenew コマンド）
   - システム構成変更コマンドの実行（pdchgconf コマンド）

   注※1

   トランザクションキューイング中のユニット，又はユニット内のサーバを終了した場合にトランザクションキューイングが解除されます。

   注※2

   pdstop コマンドがエラー終了した場合でも，トランザクションキューイングが解除されることがあります。
6. UAP のトランザクションがキューイングの対象になりますが，次に示すトランザクションはキューイングの対象になりません。
   - 定義系 SQL を実行するトランザクション

- ホールダブルカーソルを使用しているトランザクション
- ASSIGN LIST 文を実行するトランザクション
- UNTIL DISCONNECT 指定の LOCK 文を実行するトランザクション
- X/Open XA インタフェースを使用したトランザクション
- コマンド又はユティリティのトランザクション

(8) リターンコード

pdtrnqing コマンドのリターンコードを次に示します。

0：正常終了

4：警告

8：エラー終了

12：エラー終了

リターンコードの 8 及び 12 は，トランザクションキューイングの開始又は解除が実行できなかったときに返されるリターンコードです。このとき，イベントログ又は標準エラー出力にメッセージが出力されます。

実行環境の不正や，操作誤りなどの場合はリターンコード 8 で終了し，それ以外の場合はリターンコード 12 で終了します。

2.107.2 影響分散スタンバイレス型系切り替え機能使用時に pdtrnqing コマンドを実行する場合

(1) 機能

影響分散スタンバイレス型系切り替え機能のバックエンドサーバでトランザクションキューイングを開始又は解除します。

(2) このコマンドの目的

pdtrnqing コマンドは，影響分散スタンバイレス型系切り替え機能の計画系切り替えを行う前に，新規トランザクションをキューイングしておくために使います。

pdtrnqing コマンドを実行すると，コマンド実行対象のバックエンドサーバに対する新規トランザクションのキューイングが開始されて，新規トランザクションは待ち状態になります（実行中のトランザクションは処理されます）。

(3) 前提条件

pdtrnqing コマンドを実行する場合は，pd_ha_transaction オペランドに queuing を指定しておく必要があります。

pd_ha_transaction オペランドに error（省略値）を指定している場合は，トランザクションキューイングの開始又は解除をしないで pdtrnqing コマンドを終了します。この場合のリターンコードは 4 になります。

(4) 実行者

HiRDB 管理者が実行できます。

(5) 形式

```
pdtrnqing 〔-d 〔-f〕〕〔-t 待ち合わせ時間〕-s バックエンドサーバ名
```

(6) オプション

(a) -d

トランザクションキューイングを解除する場合にこのオプションを指定します。

(b) -f

-d オプションを指定して，トランザクションキューイングを解除する場合にこのオプションを指定できます。次に示す場合にこのオプションを指定してください。

- pdtrnqing コマンドの実行対象となるバックエンドサーバがあるユニットが停止中の場合にこのオプションを指定してください。ユニットが停止中の場合でも pdtrnqing コマンドが実行できます。
- 停止中のユニット（例えば，システムマネジャがあるユニットなど）から pdtrnqing コマンドを実行できます。

(c) -t 待ち合わせ時間 ～<符号なし整数>((1～3600))（単位：秒）

- **トランザクションキューイングを開始するときに指定する場合**
  pdtrnqing コマンドは，実行中のトランザクション（pdtrnqing コマンドの実行対象となるバックエンドサーバで実行中のトランザクション）が終了するまで処理を待ち合わせます。この待ち合わせ時間を秒単位で指定します。実行中のトランザクションがここで指定した時間を経過しても終了しない場合，トランザクションキューイングを中止して pdtrnqing コマンドをエラー終了します。
- **トランザクションキューイングを解除するときに指定する場合**
  pdtrnqing コマンドの実行対象となるバックエンドサーバが待機中又は開始中の場合，バックエンドサーバの起動完了を待ち合わせる時間を秒単位で指定します。バックエンドサーバの起動処理が，ここで指定した時間を経過しても完了しない場合，トランザクションキューイングを解除しないで pdtrnqing コマンドをエラー終了します。

なお，ここで指定する待ち合わせ時間は，pd_ha_trn_queuing_wait_time オペランドの指定値（デフォルト 180 秒）以下にしてください。pd_ha_trn_queuing_wait_time オペランドの指定値以上の時間を指定しても無効になり，pd_ha_trn_queuing_wait_time オペランドの値が有効になります。

(d)　-s バックエンドサーバ名　～＜識別子＞((1～8))

pdtrnqing コマンドの実行対象となるバックエンドサーバを指定します。影響分散スタンバイレス型系切り替え機能を適用しているバックエンドサーバを指定できます。

(7)　規則

1. pdtrnqing コマンドは，-s オプションに指定するバックエンドサーバがあるユニットで実行できます。
2. 次に示す場合，pdtrnqing コマンドは正常終了します。この場合，トランザクションキューイングの適用状態は変わりません。
   - トランザクションキューイングの適用中に，pdtrnqing コマンドを実行して再度トランザクションキューイングを開始しようとした場合
   - トランザクションキューイング中でないときに，pdtrnqing -d コマンドを実行してトランザクションキューイングを解除しようとした場合
3. pdtrnqing コマンドを実行する場合，コマンドの実行対象となるユニットが起動している必要があります。また，コマンドの実行対象となるバックエンドサーバが次に示す状態である必要があります。

pdtrnqing コマンドに指定するオプション	pdtrnqing コマンドの実行対象となるバックエンドサーバの状態				
	停止	待機中	開始中	稼働中	停止中
なし	×	×	×	○※1	×
-d	×	△※2※3	○※2	○	×
-d -f	○	○	○	○	○

（凡例）

○：pdtrnqing コマンドが実行できます。

×：pdtrnqing コマンドが実行できません。

△：バックエンドサーバの開始処理の完了を待ち合わせます。pd_ha_trn_queuing_wait_time オペランドで指定した時間内に，バックエンドサーバの開始処理が完了した場合，トランザクションキューイングが解除されます。開始処理が完了しない場合は，pdtrnqing コマンドがエラーになります。

注※1

-s オプションに指定したバックエンドサーバで処理中のトランザクションが終了するまで，pdtrnqing コマンドを待ち合わせます。pd_ha_trn_queuing_wait_time オペランドで指定した待ち合わせ時間を超えてもトランザクションが終了しない場合は，pdtrnqing コマンドがエラーになります。

注※2

-s オプションに指定したバックエンドサーバが待機中又は開始中の場合，バックエンドサーバの開始処理の完了を待ち合わせます。pd_ha_trn_queuing_wait_time オペランドで指定した待ち合わせ時間を経過してもバックエンドサーバの開始処理が完了しない場合は，pdtrnqing コマンドがエラーになります。

注※3

クラスタソフトウェアのコマンドでバックエンドサーバを実行系に切り替えた直後から，HiRDB が管理するサーバステータスを「開始中」にするまでの間に pdtrnqing -d コマンド実行するとエラーになります。このため，バックエンドサーバが待機中のときに，バックエンドサーバの開始処理を待ち合わせます。

(8) 注意事項

1. トランザクションキューイングの解除時，バックエンドサーバが待機中の場合はバックエンドサーバの開始処理の完了を待ち合わせます。そのため，対象とするバックエンドサーバを間違った場合（待機中のままのバックエンドサーバを対象とした場合），pdtrnqing コマンドはすぐにエラーにはなりません。この場合，-t オプションで指定した待ち合わせ時間が経過したら pdtrnqing コマンドがエラー終了します。
2. 次に示す操作をした場合，トランザクションキューイングは自動的に解除されます。
   - トランザクションキューイングを行っているバックエンドサーバを再開始した場合（バックエンドサーバが系切り替えした場合も含む）
   - システム，ユニット，サーバを終了した場合（pdstop コマンド）※1※2
   - 修正版 HiRDB の入れ替え（pdprgrenew コマンド）
   - システム構成変更コマンドの実行（pdchgconf コマンド）

注※1

トランザクションキューイング中のユニット，又はユニット内のサーバを終了した場合にトランザクションキューイングが解除されます。

注※2

pdstop コマンドがエラー終了した場合でも，トランザクションキューイングが解除されることがあります。

(9) リターンコード

pdtrnqing コマンドのリターンコードを次に示します。

0：正常終了

4：警告終了

pd_ha_transaction オペランドに queuing が指定されていないため，トランザクションキューイングができないときに返されるリターンコードです。

8：エラー終了

12：エラー終了

リターンコードの 8 及び 12 は，トランザクションキューイングの開始又は解除が実行できなかったときに返されるリターンコードです。このとき，イベントログ又は標準エラー出力にメッセージが出力されます。

実行環境の不正や，操作誤りなどの場合はリターンコード 8 で終了し，それ以外の場合はリターンコード 12 で終了します。

2.108　pdvrup（HiRDB バージョンアップ）

2.108.1　pdvrup の形式と規則

(1)　機能

HiRDB をバージョンアップした場合，データベースの整合性を新規バージョンに合わせるために，マスタディレクトリ用 RD エリア及びデータディクショナリ用 RD エリアを保守します。

なお，このコマンドは，システム定義で pd_auto_vrup=N を指定しているときだけ実行できます。pd_auto_vrup=Y の場合は，HiRDB が自動的に pdvrup コマンドを実行します。

(2)　実行者

HiRDB 管理者が実行できます。

(3)　形式

```
pdvrup
```

(4)　オプション

なし。

(5)　規則

1. pdvrup コマンドを実行する契機は，新しいバージョンの HiRDB をインストールした後に HiRDB を開始したときです。このとき，システム定義に pd_auto_vrup=N を指定している場合に，pdvrup コマンドの入力を要求する KFPS05203-Q メッセージが出力されます。それ以外のときに実行すると，エラーとなります。バージョンアップの手順については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
2. pdvrup コマンドは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdvrup コマンド実行中は，UAP 及びユティリティは実行できません。
4. pdvrup コマンド実行時に，ディクショナリ表に対して空き領域の再利用機能を適用します。

(6)　注意事項

1. pdvrup コマンドの実行結果は，コマンド実行時のリターンコード，又はエラーメッセージの有無で確認できます。コマンド実行時のリターンコードが 0 の場合は正常終了，4 の場合はバージョンアップは正常終了（この場合 HiRDB を再起動してください），12 の場合は異常終了です。

3 データベース初期設定ユティリティ（pdinit）

この章では，各ファイルを HiRDB のデータベースとして使用するための物理構成を設定するデータベース初期設定ユティリティ（pdinit）について説明します。

3.1　概要

実行者　HiRDB 管理者が実行できます

3.1.1　機能

データベース初期設定ユティリティは，各ファイルを HiRDB のデータベースとして使用するために，物理構成の設定をするユティリティです。HiRDB のデータベースは，各種の RD エリアの集合で構成されています。また，RD エリアは一つ又は複数の HiRDB ファイルで構成されています。

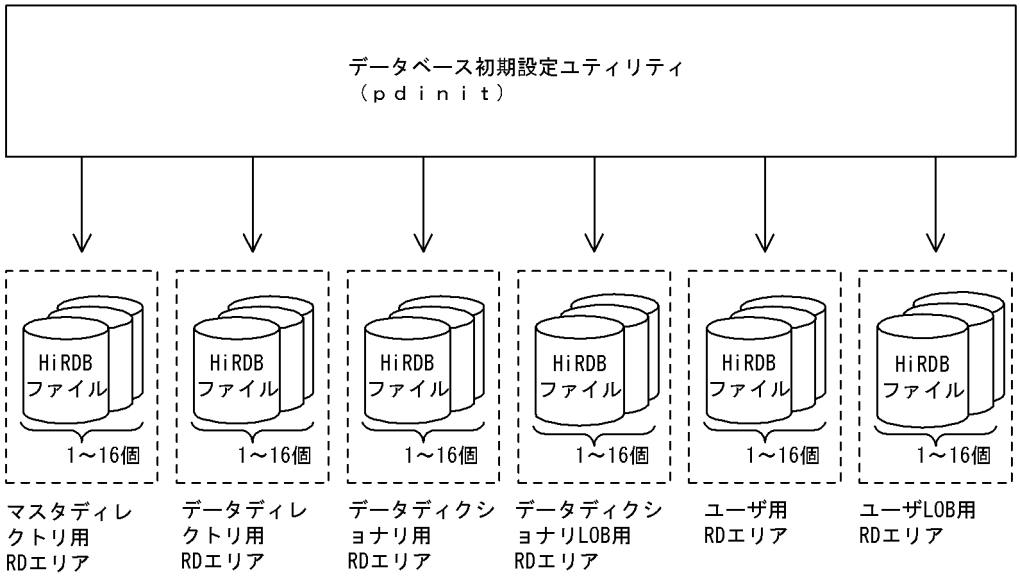
データベース初期設定ユティリティを実行する場合は，あらかじめ HiRDB ファイルを作成するための HiRDB ファイルシステム領域を初期設定しておく必要があります。HiRDB ファイルシステムの初期設定については，pdfmkfs コマンドを参照してください。

データベース初期設定ユティリティには，次の二つの機能があります。

- 各 RD エリアを初期化し，同時にマスタディレクトリ及びデータディレクトリを生成します。さらに，データディクショナリ用 RD エリアに HiRDB 固有の資源，及び各種権限を管理するためのディクショナリ表も作成します。
- データベースのシステム定義をします。

データベース初期設定ユティリティ（pdinit）の概要を次の図に示します。

図 3–1　データベース初期設定ユティリティ（pdinit）の概要



3.2　コマンドの形式

3.2.1　形式

```
pdinit -d 制御文ファイル名　〔-u 認可識別子〔-p パスワード〕〕
〔-W 実行監視時間〕
```

3.2.2　オプション

●　**-d 制御文ファイル名**　～＜パス名＞((255 文字以内))

【HiRDB/シングルサーバの場合】

データベース初期設定ユティリティの制御文があるファイルの名称を絶対パス名で指定します。

【HiRDB/パラレルサーバの場合】

データベース初期設定ユティリティの制御文があるファイルの名称を絶対パス名で指定します。制御文ファイルは，ディクショナリサーバがあるサーバマシンに作成してください。

●**-u 認可識別子**　～＜先頭が英字記号で始まる英数字記号＞((30 文字以内))

ここで指定した認可識別子に対して DBA 権限を与えて HiRDB に登録します。

- **このオプションを省略した場合**

  このオプションを省略した場合，root がこのオプションに指定されたと仮定します。

- **オプション指定時の規則**

  指定する認可識別子に小文字がある場合は，「¥""」と「"¥"」で囲んでください。「¥""」と「"¥"」で囲まないと大文字として扱われます。

- **備考**

  HiRDB を使用するオープンミドルウェアの延長で，HiRDB がインストールされる場合にもこのオプションは指定できます。

●**-p パスワード**　～＜識別子＞((30 文字以内))

-u オプションに指定した認可識別子に対応するパスワードを指定します。

指定するパスワードに小文字がある場合は，「¥""」と「"¥"」で囲んでください。「¥""」と「"¥"」で囲まないと大文字として扱われます。

なお，このオプションを省略した場合，-u オプションに指定した認可識別子と同じ文字列がパスワードとして登録されます。

-u オプションと-p オプションの指定値と，HiRDB に登録される認可識別子とパスワードの関係を次に示します。

-u オプション の指定	-p オプション の指定	HiRDB に登録される 認可識別子	HiRDB に登録される パスワード
HIR001	sytg0421	HIR001	sytg0421
HIR001	なし	HIR001	HIR001
なし	なし	root	root

●-W 実行監視時間　～＜符号なし整数＞((0～3600)) ≪0≫

データベース初期設定ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，システム共通定義の pd_cmd_exec_time オペランドに指定した実行監視時間は適用されません。このオプションを省略した場合は，0 を仮定します。

3.3 制御文

3.3.1 define system 文

```
define system  [dicinf  {unlimited | limited} ]
               [dictionary datatype mchar  {use | nouse} ]
  ;
```

(1) dicinf {unlimited | limited}

ディクショナリ表をアクセスする場合に，ユーザが持っている権限によって検索できる情報を限定するかどうかを指定します。

unlimited

ディクショナリ表をアクセスする場合に，すべての情報を検索できます。

limited

ディクショナリ表をアクセスする場合に，ユーザが持っている権限によって検索できる情報を限定します。DBA 権限を持っているユーザは，すべての情報を検索できます。DBA 権限を持っていないユーザは，利用できる情報だけ検索できます。

limited を指定した場合，ディクショナリ表を検索する UAP のアクセス表数が増大するので，同時アクセス可能実表数（pd_max_access_tables の値）を大きくする必要があります。一つのディクショナリ表につき，同時アクセス可能実表数は５必要です。

ディクショナリ表の参照権限の設定については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(2) dictionary datatype mchar {use | nouse}

ディクショナリ表のデータに混在文字データを使用するかどうかを指定します。

use：

ディクショナリ表の列のデータ型が MVARCHAR になります。

nouse：

ディクショナリ表の列のデータ型が VARCHAR になります。

MVARCHAR にするか，又は VARCHAR にするかの選択基準を次に示します。

- 表名又は列名にマルチバイト文字を使用している場合は，MVARCHAR を選択してください。
- DABroker を使用していて，表名又は列名に漢字を使用している場合は，MVARCHAR を選択してください。
- 上記以外の場合は，VARCHAR を選択してください。

3.3.2　create rdarea 文

```
create 〔shared〕
            rdarea RDエリア名
            for {masterdirectory | datadirectory | datadictionary 〔of routines〕 | user used
by {PUBLIC | 認可識別子〔, 認可識別子〕…} | LOB used by {HiRDB（ディクショナリ表の名称） |
PUBLIC | 認可識別子〔, 認可識別子〕…} | list}
             〔server name サーバ名〕
             〔open attribute {INITIAL | DEFER | SCHEDULE} 〕
             〔page ページ長 characters〕
             〔storage control segment セグメントサイズ pages〕
             〔max entries 最大リスト登録数〕
             〔extension {use 増分セグメント数 segments | nouse} 〕
             〔temporary table {use {shared | occupied} | nouse} 〕
            file name ”HiRDBファイルシステム領域名¥HiRDBファイル名”
                initial HiRDBファイル内セグメント数 segments
             〔file name ”HiRDBファイルシステム領域名¥HiRDBファイル名”
                  initial HiRDBファイル内セグメント数 segments〕…
  ;
```

注

file name オペランド，initial オペランドは，必ず create rdarea 文の最後に指定してください。

(1)　shared

このオペランドは，HiRDB/パラレルサーバの場合に指定できます。

RD エリアを共用 RD エリアとして作成する場合に指定します。

＜規則＞

1. shared を指定する場合，システム定義の pd_shared_rdarea_use オペランドに Y を指定する必要があります。指定していない場合はエラーになります。
2. HiRDB/シングルサーバの場合，shared を指定するとエラーになります。
3. 共用 RD エリアとして作成できるのはユーザ用 RD エリアだけです。
4. 共用 RD エリアは，使用目的が SDB の HiRDB ファイルシステム領域に作成してください。
5. 共用 RD エリアとして作成した RD エリアは，server name オペランドで指定したサーバ以外のサーバからも参照できます（すべてのバックエンドサーバから参照できます）。

(2)　RD エリア名

～＜識別子＞((1～30))

HiRDB で使用する RD エリアの名称を指定します。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。

（例）

- **create rdarea "pdbuser01" for ...**
  この場合，RD エリア名は pdbuser01 で扱われます。
- **create rdarea pdbuser01 for ...**
  この場合，RD エリア名は PDBUSER01 で扱われます。

(3) for {masterdirectory | datadirectory | datadictionary〔of routines〕 | user used by {PUBLIC | 認可識別子〔，認可識別子〕...} | LOB used by {HiRDB（ディクショナリ表の名称） | PUBLIC | 認可識別子〔，認可識別子〕...} | list}

RD エリアの種類を指定します。

masterdirectory

RD エリアがマスタディレクトリ用 RD エリアの場合に指定します。

datadirectory

RD エリアがデータディレクトリ用 RD エリアの場合に指定します。

datadictionary

RD エリアがデータディクショナリ用 RD エリアの場合に指定します。

of routines

ストアドプロシジャ，ストアドファンクション，及びトリガに関するディクショナリ表を，ほかのディクショナリ表と別の RD エリアに格納する場合に指定します。

このオペランドは，一つのデータディクショナリ用 RD エリアにだけ指定できます。また，このオペランドを指定した場合，データディクショナリ LOB 用 RD エリアを必ず指定してください。

user used by {PUBLIC | 認可識別子〔，認可識別子〕...}

RD エリアがユーザ用 RD エリアの場合に指定します。

PUBLIC

ユーザ用 RD エリアを公用 RD エリアとする場合に指定します。

認可識別子〔，認可識別子〕... ～＜先頭が英字記号で始まる英数字記号＞((1～30))

ユーザ用 RD エリアを私用 RD エリアにする場合，ユーザ用 RD エリアを利用できるユーザの認可識別子を指定します。

＜規則＞

1. 認可識別子を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として HiRDB システムに登録されます。
2. 認可識別子を指定すると，その私用 RD エリアに対して RD エリア利用権限を持つことができます。
3. 認可識別子は，最大 255 個指定できます。

4. 認可識別子に PUBLIC，MASTER，HiRDB，及び ALL は指定できません。

LOB used by ｛HiRDB（ディクショナリ表の名称）｜PUBLIC｜認可識別子〔，認可識別子〕...｝

RD エリアがデータディクショナリ LOB 用 RD エリア又はユーザ LOB 用 RD エリアの場合に指定します。

HiRDB（ディクショナリ表の名称）

RD エリアがデータディクショナリ LOB 用 RD エリアの場合に指定します。ディクショナリ表の名称には，SQL_ROUTINES を指定してください。

＜規則＞

1. ストアドプロシジャ，又はストアドファンクションを使用する場合には，必ずこの RD エリアを定義してください。
2. この RD エリアには，ストアドプロシジャ及びストアドファンクションの定義ソース文を格納する RD エリアと，ストアドプロシジャ及びストアドファンクションのオブジェクトを格納する RD エリアがあるため，指定する場合は二つする必要があります（この RD エリアを指定した create rdarea 文を 2 個指定します）。一つ目の RD エリアにはストアドプロシジャ及びストアドファンクションの定義ソース文が格納され，二つ目の RD エリアにはストアドプロシジャ及びストアドファンクションのオブジェクトが格納されます。
3. データベース初期設定ユティリティ実行時にこの RD エリアを定義していない場合に，ストアドプロシジャ及びストアドファンクションを使用するときには，データベース構成変更ユティリティでこの RD エリアを追加してください。

PUBLIC

ユーザ LOB 用 RD エリアを公用 RD エリアとする場合に指定します。

認可識別子〔，認可識別子〕... ～＜先頭が英字記号で始まる英数字記号＞((1～30))

ユーザ LOB 用 RD エリアを私用 RD エリアにする場合，ユーザ LOB 用 RD エリアを利用できるユーザの認可識別子を指定します。

＜規則＞

1. 認可識別子を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として HiRDB システムに登録されます。
2. 認可識別子を指定すると，その私用 RD エリアに対して RD エリア利用権限を持つことができます。
3. 認可識別子は，最大 255 個指定できます。
4. 認可識別子に PUBLIC，MASTER，HiRDB，及び ALL は指定できません。

list

RD エリアがリスト用 RD エリアの場合に指定します。

(4) server name サーバ名

～＜識別子＞((1～8))

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は server name オペランドを指定しないでください。

【HiRDB/パラレルサーバの場合】

RD エリアを管理するサーバの名称を指定します。

マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディクショナリ LOB 用 RD エリアの場合，ディクショナリサーバのサーバ名を指定してください。ディクショナリサーバ以外のサーバ名を指定した場合でも，ディクショナリサーバのサーバ名が仮定されます。

ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアの場合，バックエンドサーバのサーバ名を指定してください。バックエンドサーバ以外のサーバ名を指定した場合は，エラーとなります。

共用 RD エリアを作成する場合，このオペランドで指定したサーバが更新可能バックエンドサーバとなります。

一つのサーバで複数の RD エリアを管理する場合，同一サーバ名の create rdarea 文は続けて指定する必要があります。例を次に示します。

＜例 1 ＞

```
create rdarea A … server name BES1 … ;
create rdarea B … server name BES1 … ;
create rdarea C … server name BES2 … ;
```

同一サーバ名（BES1）の create rdarea 文を続けて指定しているので，エラーにはなりません。

＜例 2 ＞

```
create rdarea A … server name BES1 … ;
create rdarea B … server name BES2 … ;
create rdarea C … server name BES1 … ;
```

同一サーバ名（BES1）の create rdarea 文を続けて指定していないので，エラーになります。

(5) open attribute {INITIAL | DEFER | SCHEDULE}

RD エリアのオープン契機を指定します。

＜適用基準＞

特殊な運用形態でない場合は，通常 INITIAL を指定してください。

RD エリア数が多いために HiRDB の開始に時間が掛かるなどの不都合を回避したい場合は，DEFER を指定してください。

＜規則＞

1. このオペランドは，pd_rdarea_open_attribute_use オペランドに Y を指定しているときだけ有効となります。
2. このオペランドはユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアの場合に指定できます。ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアに対してこのオペランドを指定しても，無視されます（ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアは，INITIAL 固定です）。

3. このオペランドを省略した場合，pd_rdarea_open_attribute オペランドの指定値が仮定されます。
4. 該当する RD エリアに割り当てた HiRDB ファイルシステム領域が，ほかの RD エリアに共有される場合，共有する RD エリアのオープン契機はすべて同じにする必要があります。同じでない場合，HiRDB ファイルシステム領域を共有する RD エリアは，期待通りの効果が得られないことがあります。

INITIAL：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンして，RD エリア情報をメモリ上に常駐させるので，最初の SQL から高速な運用ができます。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はオープンで，それ以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. INITIAL の場合，クローズ状態の RD エリアはアクセスできません。

DEFER：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，RD エリアを最初にアクセスするときに HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させるので，2 回目以降のアクセスから高速な運用ができます。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，各 RD エリアに対して最初にアクセスするときに，該当する RD エリアをオープンします。以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. DEFER の場合，クローズ状態の RD エリアでもアクセスできます。

SCHEDULE：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，HiRDB 開始後，各トランザクション内で RD エリアに対して最初にアクセスするときに，HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させます。ただし，トランザクション終了時にそのトランザクション内でオープンした HiRDB ファイルシステム領域をクローズするため，2 回目以降のアクセス時でもオープン以降の処理をして，トランザクションに掛かる負荷は増加します。

＜注意事項＞

1. HiRDB 開始時の R D エリアの初期状態はクローズで，アクセスした RD エリアをトランザクション内でだけオープンし，トランザクション終了時にトランザクション内でオープンしたすべての RD エリアをクローズします。また，pdopen コマンドを実行することで，閉塞クローズ状態になるまでの間はオープン状態にできます。そのほかの運用コマンドを使用すれば，RD エリアの状態を任意に遷移させることもできます。なお，障害事象を検知した場合には，障害閉塞となります。
2. SCHEDULE の場合，クローズ状態の RD エリアでもアクセスできます。

(6) page ページ長 characters

～＜符号なし整数＞((4096～30720))《4096 又は 8192》

RD エリアを構成する HiRDB ファイルのページ長を，2,048 の倍数のバイト単位で指定します。HiRDB ファイルシステム領域作成時にセクタ長を指定した場合，2,048 の倍数かつセクタ長の倍数を指定してください。

＜マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリアの場合＞

4,096 を指定してください。4,096 以外の値を指定した場合でも，4,096 が仮定されます。

＜データディクショナリ LOB 用 RD エリア，ユーザ LOB 用 RD エリアの場合＞

8,192 を指定してください。8,192 以外の値を指定した場合でも，8,192 が仮定されます。

そのほかの RD エリアのページ長については，マニュアル「HiRDB システム導入・設計ガイド」を参照して適切な値を指定してください。

ここで指定したページ長は，HiRDB が RD エリアに対して入出力する最小の単位となります。

(7) storage control segment セグメントサイズ pages

～＜符号なし整数＞((1～16000))《1 又は 50》

1 セグメントの大きさを，ページ数で指定します。

＜マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリアの場合＞

50 を指定してください。50 以外の値を指定した場合でも，50 が仮定されます。

＜データディクショナリ LOB 用 RD エリア，ユーザ LOB 用 RD エリアの場合＞

1 を指定してください。1 以外の値を指定した場合でも，1 が仮定されます。

そのほかの RD エリアのセグメントサイズについては，マニュアル「HiRDB システム導入・設計ガイド」を参照して適切な値を指定してください。

(8) max entries 最大リスト登録数

～＜符号なし整数＞((500～50000，かつ 500 の倍数))≪500≫

リスト用 RD エリアに作成できるリストの最大数を，500 の倍数で指定します。500 の倍数でない値を指定した場合，500 の倍数に切り上げた値が仮定されます。

このオペランドをリスト用 RD エリア以外の RD エリアに指定した場合，エラーとなります。

(9) extension {use 増分セグメント数 segments | nouse}

RD エリアに自動増分を適用するかどうかを指定します。

RD エリアの自動増分については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜前提条件＞

RD エリアがある HiRDB ファイルシステム領域に-e オプション又は-a オプションが指定されている必要があります。

use 増分セグメント数 segments：

RD エリアの自動増分を適用します。

RD エリアの自動増分は，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，ユーザ用 RD エリア，及びユーザ LOB 用 RD エリアの場合に有効となります。ほかの RD エリアに use を指定しても，nouse が仮定されます。

自動増分の契機は RD エリア内の空きセグメント数が自動増分契機に達したときです。
pd_rdarea_extension_timing オペランドで指定できます。

増分セグメント数　～＜符号なし整数＞((1～64000))

増分セグメント数を指定します。

- pdfmkfs -e を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルのエクステント数は最大 24 で，これを超えるとエラーとなります。また，HiRDB ファイルシステム領域の空きが不足したり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

  HiRDB ファイルシステム領域当たりの最大増分回数は，HiRDB ファイルシステム領域作成時の指定値で決まるため，HiRDB ファイルシステム領域内のファイル数，増分頻度を考慮し最大増分回数を定義しておく必要があります。

- pdfmkfs -a を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルを格納しているディスクに，必要な HiRDB ファイルシステム領域を確保する容量が不足していたり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

- 指定した増分セグメント数を適用した場合に，RD エリアを構成する HiRDB ファイルの最終ファイルのサイズが 64 ギガバイトを超える増分セグメント数は指定できません（KFPX24026-E エラーとなります）。なお，このチェックは，最終 file オペランドの initial に指定したセグメント数に，指定した増分セグメント数を 1 回分加算して算出したファイルサイズで判定します。

nouse：

RD エリアの自動増分機能を適用しません。

＜注意事項＞

1. 新たに追加される未使用セグメントは，RD エリアを構成する HiRDB ファイルの最終ファイルに追加されます。
2. HiRDB ファイルシステム領域の容量不足によって，未使用セグメントを確保できない場合，RD エリアの拡張又は再初期化をするか，データベース再編成ユティリティで表の再編成をしてください。
3. エクステントの数が上限値を超えた場合は，RD エリアがある HiRDB ファイルシステム領域のエクステントを統合するか，又は RD エリアに別の HiRDB ファイルシステム領域の HiRDB ファイルを追加してください。エクステントを統合するには，pdfbkup でバックアップを取得し，pdfmkfs で HiRDB ファイルシステム領域の初期設定をし，pdfrstr でバックアップから回復します。
4. RD エリア自動増分処理の開始から終了までの間，最終ファイルに対して排他が掛かります。

(10) temporary table {use {shared | occupied} | nouse}

ユーザ用 RD エリアを一時表用 RD エリアとして使用するかどうかを指定します。一時表用 RD エリアについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

＜前提条件＞

作成する RD エリアが公用 RD エリア（user used by PUBLIC 指定）である必要があります。それ以外の場合に temporary table オペランドを指定すると，エラーになります。また，shared オペランドと同時に指定したときも，エラーになります。なお，この前提条件は，nouse を指定したときには該当しません。

use shared：

ユーザ用 RD エリアを，SQL セッション間共有属性の一時表用 RD エリアとして使用します。

use occupied：

ユーザ用 RD エリアを，特定 SQL セッション占有属性の一時表用 RD エリアとして使用します。

nouse：

ユーザ用 RD エリアを一時表用 RD エリアとして使用しません。

＜注意事項＞

RD エリア作成後に一時表用 RD エリアの使用有無，及び一時表用 RD エリアの属性（SQL セッション間共有属性又は特定 SQL セッション占有属性）を変更する場合，データベース構成変更ユティリティの remove rdarea 文で該当する RD エリアを削除し，create rdarea 文で属性を変更して再作成してください。

(11) file name "HiRDB ファイルシステム領域名¥HiRDB ファイル名"

～((167 文字以内))

RD エリアに対して割り当てる，HiRDB ファイルシステム領域の名称と HiRDB ファイルの名称を指定します。

HiRDB ファイルシステム領域名¥HiRDB ファイル名は引用符（"）で囲んで指定してください。

＜規則＞

1. 引用符内には改行文字を含まないようにしてください。
2. 一つの RD エリアに対して，最大 16 個の HiRDB ファイルを割り当てられます。
3. HiRDB ファイルシステム領域名¥HiRDB ファイル名は，HiRDB システム内で重複しないようにしてください。
4. リスト用 RD エリアの場合は，使用目的が WORK の HiRDB ファイルシステム領域を使用してください。
5. マスタディレクトリ用 RD エリアを複数の HiRDB ファイルで構成する場合，先頭の HiRDB ファイル名はシステム定義の pd_master_file_name オペランドに指定してある名称と同じにしてください。

6. "HiRDB ファイルシステム領域名¥HiRDB ファイル名"の HiRDB ファイルシステム領域名までは大文字，小文字を区別しませんが，HiRDB ファイル名は大文字，小文字を区別します。

HiRDB ファイルシステム領域名　～＜パス名＞

HiRDB ファイルシステム領域の名称を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞((1～30 文字))

HiRDB ファイルの名称を指定します。pl で始まる名称は指定できません。

(12)　initial HiRDB ファイル内セグメント数 segments

HiRDB ファイルのセグメント数を指定します。

ただし，HiRDB ファイルの大きさが 64 ギガバイトを超えるようなセグメント数は指定できません。各 RD エリアを構成する HiRDB ファイルの総セグメント数は次の計算式より大きい値を指定してください。

RD エリアの種別	最小セグメント数
マスタディレクトリ用 RD エリア	4 ＋↑すべての RD エリア数÷800↑
データディレクトリ用 RD エリア	1
データディクショナリ用 RD エリア※1	221※3
データディクショナリ LOB 用 RD エリア※2	1
ユーザ用 RD エリア	1
ユーザ LOB 用 RD エリア	1
リスト用 RD エリア	1

注※1

データディクショナリ用 RD エリアが複数ある場合，次に示す表の順に，セグメント数の多い RD エリアから 1RD エリアに１ディクショナリ表ずつ格納されます。ディクショナリ表のインデクスは，対応するディクショナリ表と同じ RD エリアに格納されます。したがって，データディクショナリ用 RD エリアが複数ある場合は，格納されるディクショナリ表及びそのディクショナリ表のインデクスの数が最小セグメント数となります。

格納順序	ディクショナリ表の名称	インデクス数
1	SQL_PHYSICAL_FILES	2
2	SQL_RDAREAS	4
3	SQL_TABLES	2
4	SQL_COLUMNS	3
5	SQL_INDEXES	3
6	SQL_USERS	2
7	SQL_RDAREA_PRIVILEGES	2

格納順序	ディクショナリ表の名称	インデクス数
8	SQL_TABLE_PRIVILEGES	2
9	SQL_DIV_TABLE	3
10	システムが使用する表	2
11	SQL_INDEX_COLINF	2
12	SQL_TABLE_STATISTICS	1
13	SQL_COLUMN_STATISTICS	1
14	SQL_INDEX_STATISTICS	2
15	SQL_VIEW_TABLE_USAGE	3
16	SQL_VIEWS	2
17	システムが使用する表	2
18	SQL_DIV_INDEX	2
19	SQL_DIV_COLUMN	2
20	SQL_REFERENTIAL_CONSTRAINTS	3
21	SQL_ALIASES	1
22	SQL_DATATYPES	3
23	SQL_DATATYPE_RESOURCES	2
24	SQL_TABLE_RESOURCES	3
25	SQL_PLUGINS	3
26	SQL_PLUGIN_ROUTINES	4
27	SQL_PLUGIN_ROUTINE_PARAMS	3
28	SQL_INDEX_TYPES	2
29	SQL_INDEX_RESOURCES	2
30	SQL_INDEX_DATATYPE	2
31	SQL_INDEX_FUNCTION	2
32	SQL_TYPE_RESOURCES	3
33	SQL_INDEX_TYPE_FUNCTION	1
34	SQL_EXCEPT	3
35	システムが使用する表	0
36	システムが使用する表	0
37	SQL_TRIGGERS	4
38	システムが使用する表	2

格納順序	ディクショナリ表の名称	インデクス数
39	SQL_TRIGGER_COLUMNS	2
40	SQL_TRIGGER_DEF_SOURCE	2
41	SQL_TRIGGER_USAGE	4
42	SQL_PARTKEY	1
43	SQL_PARTKEY_DIVISION	1
44	SQL_AUDITS	3
45	システムが使用する表	2
46	SQL_KEYCOLUMN_USAGE	2
47	SQL_TABLE_CONSTRAINTS	2
48	SQL_CHECKS	2
49	SQL_CHECK_COLUMNS	2
50	SQL_DIV_TYPE	1
51	SQL_SYSPARAMS	1
52	システムが使用する表	1
53	SQL_INDEX_XMLINF	2
54	SQL_SEQUENCES	2
55	システムが使用する表	1
56	システムが使用する表	3
57	システムが使用する表	4
58	システムが使用する表	5
59	システムが使用する表	4
60	システムが使用する表	3
61	SQL_ACCESS_SECURITY	3
62	SQL_ROLES	2
63	SQL_ROLE_PRIVILEGES	1
64	SQL_ROUTINES	4
65	SQL_ROUTINE_RESOURCES	4
66	SQL_ROUTINE_PARAMS	3

注※2

データディクショナリ LOB 用 RD エリアを指定していない場合，SQL_ROUTINES，SQL_ROUTINE_RESOURCES，及び SQL_ROUTINE_PARAMS は作成されません。また，ストアドプロ

シジャ及びストアドファンクションに関するディクショナリ表を格納するデータディクショナリ用 RD エリア（of routines を指定した RD エリア）がある場合，SQL_ROUTINES，SQL_ROUTINE_RESOURCES，及び SQL_ROUTINE_PARAMS はそのデータディクショナリ用 RD エリアに格納されます。

注※3

HiRDB が作成するディクショナリ表数及びディクショナリ表のインデクス数の合計を示します。

セグメント数は，データページ部分の総ページ数÷セグメントサイズで求めてください。データページ部分の総ページ数の見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

3.3.3 注意事項

1. 制御文に注釈を付けたい場合は，スラント，アスタリスク（/＊）とアスタリスク，スラント（＊/）で囲んで指定してください。
2. define system 文と create rdarea 文を同時に指定する場合は，define system 文を先に指定してください。
3. create rdarea 文の指定順序と指定できる数を次に示します。

指定順序	RD エリアの種類	指定できる数
1	マスタディレクトリ用 RD エリア	1
2	データディレクトリ用 RD エリア	1
3	データディクショナリ用 RD エリア	1～ディクショナリ表の数※
4	データディクショナリ LOB 用 RD エリア	0 又は 2
5	ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア	任意

注 1

RD エリアの数は，システム共通定義の RD エリアの最大数（pd_max_rdarea_no の値）を超えないようにしてください。また，RD エリアを構成する HiRDB ファイルの数も，RD エリアを構成する HiRDB ファイルの最大数（pd_max_file_no の値）を超えないようにしてください。

注 2

共用 RD エリアを作成する場合，その RD エリアはすべてのバックエンドサーバに作成したものとしてカウントされます。したがって，どれかのバックエンドサーバが RD エリアの最大数，又は RD エリアを構成する HiRDB ファイルの最大数を超える場合，共用 RD エリアは作成できません。

注 3

0904 互換モードを適用している場合で，一時表用 RD エリアを作成するときは，システム共通定義の一時表用 RD エリアの最大数（pd_max_tmp_table_rdarea_no の値）を超えないようにしてください。

一時表用 RD エリアの最大数を超える場合は，HiRDB を終了してからシステム共通定義を変更する必要があります。

注※

of routines を指定したデータディクショナリ用 RD エリアがある場合は，指定できるデータディクショナリ用 RD エリアの数は（ディクショナリ表の数−2）になります。

3.4　規則及び注意事項

3.4.1　規則

1. データベース初期設定ユティリティは，HiRDB が初期設定終了待ち状態（KFPS05201-Q メッセージを出力し，待っている状態）のときだけ実行できます。初期設定終了待ち状態とは，マスタディレクトリ用 RD エリアが初期化されていない状態，又は pdstart コマンドで-i オプションを指定して HiRDB を開始した状態をいいます。
2. データベース初期設定ユティリティは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. データベース初期設定ユティリティ実行時にエラーが発生した場合，エラーが発生した時点で処理を打ち切ります。この場合，標準エラー出力，及びメッセージログファイルに出力されたエラーの原因を取り除き，再度データベース初期設定ユティリティを実行する必要があります。
4. データベース初期設定ユティリティが異常終了（アボート，及びプロセスダウン）した場合，%PDDIR%¥spool に初期設定ステータスファイル（pdinit）が残ることがあります。データベース初期設定ユティリティを再度実行する場合には，このファイルを削除する必要があります。
5. 1:1 スタンバイレス型系切り替え機能を使用していて代替 BES を待機状態にしている場合，pdinit は実行できません。1:1 スタンバイレス型系切り替え機能を使用している場合に pdinit を実行するときは，すべての正規 BES を実行系として開始してください。
6. データベース初期設定ユティリティを実行する場合は，-u オプション及び-p オプションを指定してください。指定していない場合は，HiRDB は固有名称の認可識別子及びパスワードに DBA 権限を与えるため，DBA 権限が他のユーザに使用されるおそれがあります。
7. データベース初期設定ユティリティ実行時に，ディクショナリ表に対して空き領域の再利用機能を適用します。

3.4.2　注意事項

1. pdinit のリターンコードを次に示します。

   0：正常終了

   4：正常終了（警告レベルのエラーがありましたが，処理は正常に終了しました）

   8：正常終了（初期化は正常に終了しましたが，初期化終了の通信がエラーになりました）

   12：異常終了
2. pdinit の結果は，次の方法で確認できます。

   pddbls コマンド（-r ALL -a 指定）

   ディクショナリ表の検索

   データベース状態解析ユティリティ

3. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，制御文ファイル中に BOM が存在しても問題ありません。ただし，制御文ファイル中のコメントには，ASCII コード以外の文字コードを使用できません。ASCII コード以外の文字コードを使用した場合，pdinit が誤動作するおそれがあるため，使用しないでください。

3.5 使用例

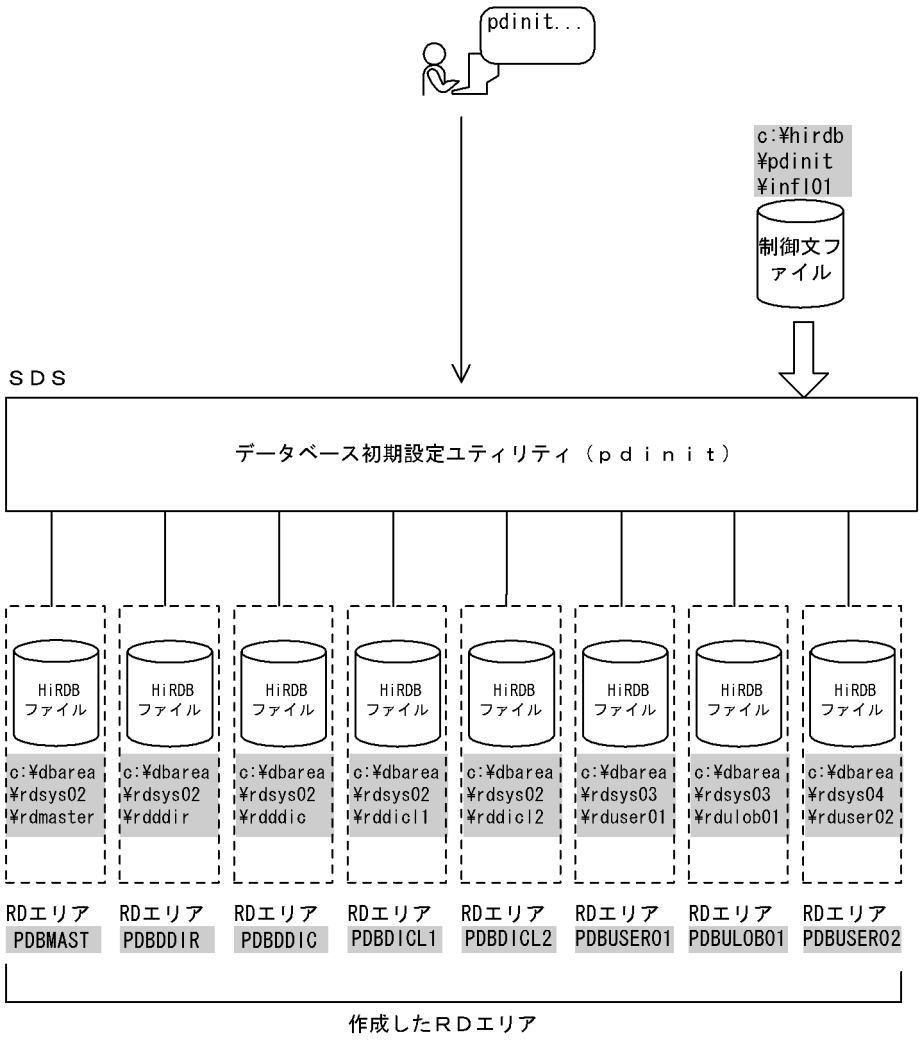
データベース初期設定ユティリティの使用例を，例 1 に示します。

例 1

次に示すRDエリアを作成します。
・PDBMAST（マスタディレクトリ用RDエリア）
・PDBDDIR（データディレクトリ用RDエリア）
・PDBDDIC（データディクショナリ用RDエリア）
・PDBDICL1（データディクショナリLOB用RDエリア）
・PDBDICL2（データディクショナリLOB用RDエリア）
・PDBUSER01（ユーザ用RDエリア）
・PDBULOB01（ユーザLOB用RDエリア）
・PDBUSER02（ユーザ用RDエリア）
なお，既に次のHiRDBファイルシステム領域が作成されているものとします。
・c:¥dbarea¥rdsys02
・c:¥dbarea¥rdsys03
・c:¥dbarea¥rdsys04

【HiRDB/シングルサーバの場合】

[概要]



（凡例）　：データベース初期設定ユティリティでの指定値
ＳＤＳ：シングルサーバ

[コマンド実行例]

```
pdinit -d c:¥hirdb¥pdinit¥infl01
```

制御文ファイル（c:¥hirdb¥pdinit¥infl01）の内容

```
/* マスタディレクトリ用RDエリアの定義 */
create rdarea PDBMAST   ..............................1
              for masterdirectory   ..................2
              page 4096 characters   .................3
              storage control segment 50 pages   .....4
              file name "c:¥dbarea¥rdsys02¥rdmaster"
                  initial 10 segments ;   ............5

/* データディレクトリ用RDエリアの定義 */
create rdarea PDBDDIR   ..............................6
              for datadirectory   ....................7
              page 4096 characters   .................8
              storage control segment 50 pages   .....9
              file name "c:¥dbarea¥rdsys02¥rdddir"
                  initial 5 segments ;   ............10
```

```
/* データディクショナリ用RDエリアの定義 */
 create rdarea PDBDDIC   ..............................11
               for datadictionary   ....................12
               page 4096 characters   ..................13
               storage control segment 30 pages   ......14
               file name "c:¥dbarea¥rdsys02¥rdddic"
                   initial 150 segments ;   ............15

/* データディクショナリLOB用RDエリアの定義 */
 create rdarea PDBDICL1   .............................16
               for LOB used by HiRDB(SQL_ROUTINES)   ...17
               page 8192 characters   ..................18
               storage control segment 1 pages   .......19
               file name "c:¥dbarea¥rdsys02¥rddicl1"
                   initial 200 segments ;   ............20

/* データディクショナリLOB用RDエリアの定義 */
 create rdarea PDBDICL2   .............................21
               for LOB used by HiRDB(SQL_ROUTINES)   ...22
               page 8192 characters   ..................23
               storage control segment 1 pages   .......24
               file name "c:¥dbarea¥rdsys02¥rddicl2"
                   initial 2000 segments ;   ...........25

/* ユーザ用RDエリアの定義 */
 create rdarea PDBUSER01   ............................26
               for user used by PUBLIC   ..............27
               page 4096 characters   ..................28
               storage control segment 150 pages   .....29
               file name "c:¥dbarea¥rdsys03¥rduser01"
                   initial 50 segments ;   ............30

/* ユーザLOB用RDエリアの定義 */
 create rdarea PDBULOB01   ............................31
               for LOB used by PUBLIC   ...............32
               page 8192 characters   ..................33
               storage control segment 1 pages   .......34
               file name "c:¥dbarea¥rdsys03¥rdulob01"
                   initial 50 segments ;   ............35

/* ユーザ用RDエリアの定義 */
 create rdarea PDBUSER02   ............................36
               for user used by USERID   ..............37
               page 4096 characters   ..................38
               storage control segment 50 pages   ......39
               file name "c:¥dbarea¥rdsys04¥rduser02"
                   initial 100 segments ;   ...........40
```

[説明]

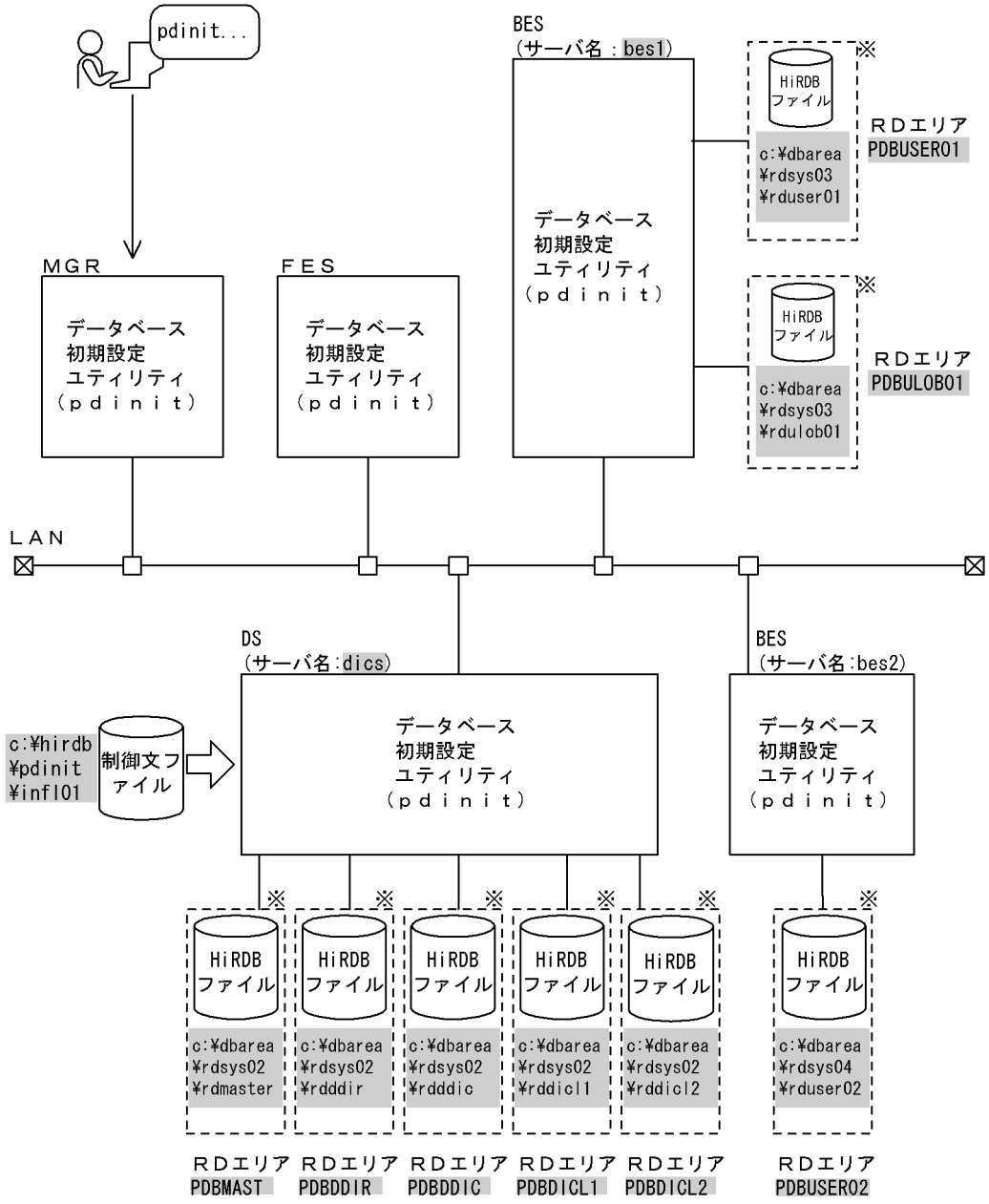
1. RD エリアの名称　**PDBMAST**
2. RD エリアの種類　**マスタディレクトリ用 RD エリア**
3. ページ長　**4096 バイト**
4. セグメントサイズ　**50 ページ**

5. RD エリアを構成する HiRDB ファイルの指定
   名称　**c:¥dbarea¥rdsys02¥rdmaster**
   セグメント数　**10**
6. RD エリアの名称　**PDBDDIR**
7. RD エリアの種類　**データディレクトリ用 RD エリア**
8. ページ長　**4096 バイト**
9. セグメントサイズ　**50 ページ**
10. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rdddir**
    セグメント数　**5**
11. RD エリアの名称　**PDBDDIC**
12. RD エリアの種類　**データディクショナリ用 RD エリア**
13. ページ長　**4096 バイト**
14. セグメントサイズ　**30 ページ**
15. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rdddic**
    セグメント数　**150**
16. RD エリアの名称　**PDBDICL1**
17. RD エリアの種類　**データディクショナリ LOB 用 RD エリア**（ストアドプロシジャの定義ソース文を格納）
18. ページ長　**8192 バイト**
19. セグメントサイズ　**1 ページ**
20. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rddicl1**
    セグメント数　**200**
21. RD エリアの名称　**PDBDICL2**
22. RD エリアの種類　**データディクショナリ LOB 用 RD エリア**（ストアドプロシジャのオブジェクトを格納）
23. ページ長　**8192 バイト**
24. セグメントサイズ　**1 ページ**
25. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rddicl2**
    セグメント数　**2000**
26. RD エリアの名称　**PDBUSER01**

27. RD エリアの種類　**ユーザ用 RD エリア（公用）**
28. ページ長　**4096 バイト**
29. セグメントサイズ　**150 ページ**
30. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys03¥rduser01**
    セグメント数　**50**
31. RD エリアの名称　**PDBULOB01**
32. RD エリアの種類　**ユーザ LOB 用 RD エリア（公用）**
33. ページ長　**8192 バイト**
34. セグメントサイズ　**1 ページ**
35. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys03¥rdulob01**
    セグメント数　**50**
36. RD エリアの名称　**PDBUSER02**
37. RD エリアの種類　**ユーザ用 RD エリア（私用）**
    認可識別子　**USERID**
38. ページ長　**4096 バイト**
39. セグメントサイズ　**50 ページ**
40. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys04¥rduser02**
    セグメント数　**100**

【HiRDB/パラレルサーバの場合】

[概要]



（凡例）　：データベース初期設定ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＤＳ　：ディクショナリサーバ
ＢＥＳ：バックエンドサーバ
注※　作成したＲＤエリアを示します。

[コマンド実行例]

```
pdinit -d c:¥hirdb¥pdinit¥infl01
```

制御文ファイル（c:¥hirdb¥pdinit¥infl01）の内容

```
/* マスタディレクトリ用RDエリアの定義 */
create rdarea PDBMAST   ..........................................1
              for masterdirectory   ..............................2
              server name dics   .................................3
              page 4096 characters   .............................4
```

```
              storage control segment 50 pages   .................5
              file name "c:¥dbarea¥rdsys02¥rdmaster"
                  initial 10 segments ;   .......................6

/* データディレクトリ用RDエリアの定義 */
 create rdarea PDBDDIR   ..........................................7
              for datadirectory   ..............................8
              server name dics   ...............................9
              page 4096 characters   ..........................10
              storage control segment 50 pages   .............11
              file name "c:¥dbarea¥rdsys02¥rdddir"
                  initial 5 segments ;   .......................12

/* データディクショナリ用RDエリアの定義 */
 create rdarea PDBDDIC   ..........................................13
              for datadictionary   ............................14
              server name dics   ...............................15
              page 4096 characters   ..........................16
              storage control segment 30 pages   .............17
              file name "c:¥dbarea¥rdsys02¥rdddic"
                  initial 150 segments ;   .....................18

/* データディクショナリLOB用RDエリアの定義 */
 create rdarea PDBDICL1   .........................................19
              for LOB used by HiRDB(SQL_ROUTINES)   ...........20
              server name dics   ...............................21
              page 8192 characters   ..........................22
              storage control segment 1 pages   ..............23
              file name "c:¥dbarea¥rdsys02¥rddicl1"
                  initial 200 segments ;   .....................24

/* データディクショナリLOB用RDエリアの定義 */
 create rdarea PDBDICL2   .........................................25
              for LOB used by HiRDB(SQL_ROUTINES)   ...........26
              server name dics   ...............................27
              page 8192 characters   ..........................28
              storage control segment 1 pages   ..............29
              file name "c:¥dbarea¥rdsys02¥rddicl2"
                  initial 2000 segments ;   ....................30

/* ユーザ用RDエリアの定義 */
 create rdarea PDBUSER01   ........................................31
              for user used by PUBLIC   .......................32
              server name bes1   ...............................33
              page 4096 characters   ..........................34
              storage control segment 150 pages   .............35
              file name "c:¥dbarea¥rdsys03¥rduser01"
                  initial 50 segments ;   ......................36

/* ユーザLOB用RDエリアの定義 */
 create rdarea PDBULOB01   ........................................37
              for LOB used by PUBLIC   ........................38
              server name bes1   ...............................39
              page 8192 characters   ..........................40
              storage control segment 1 pages   ..............41
              file name "c:¥dbarea¥rdsys03¥rdulob01"
                  initial 50 segments ;   ......................42
```

```
/* ユーザ用RDエリアの定義 */
 create rdarea PDBUSER02   ....................................43
               for user used by USERID   ......................44
               server name bes2   .............................45
               page 4096 characters   .........................46
               storage control segment 50 pages   .............47
               file name "c:¥dbarea¥rdsys04¥rduser02"
                   initial 100 segments ;   ...................48
```

[説明]

1. RD エリアの名称　**PDBMAST**
2. RD エリアの種類　**マスタディレクトリ用 RD エリア**
3. RD エリアを管理するサーバの名称　**dics**
4. ページ長　**4096 バイト**
5. セグメントサイズ　**50 ページ**
6. RD エリアを構成する HiRDB ファイルの指定
   名称　**c:¥dbarea¥rdsys02¥rdmaster**
   セグメント数　**10**
7. RD エリアの名称　**PDBDDIR**
8. RD エリアの種類　**データディレクトリ用 RD エリア**
9. RD エリアを管理するサーバの名称　**dics**
10. ページ長　**4096 バイト**
11. セグメントサイズ　**50 ページ**
12. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rdddir**
    セグメント数　**5**
13. RD エリアの名称　**PDBDDIC**
14. RD エリアの種類　**データディクショナリ用 RD エリア**
15. RD エリアを管理するサーバの名称　**dics**
16. ページ長　**4096 バイト**
17. セグメントサイズ　**30 ページ**
18. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rdddic**
    セグメント数　**150**
19. RD エリアの名称　**PDBDICL1**
20. RD エリアの種類　**データディクショナリ LOB 用 RD エリア**（ストアドプロシジャの定義ソース文を格納）

21. RD エリアを管理するサーバの名称　**dics**
22. ページ長　**8192 バイト**
23. セグメントサイズ　**1 ページ**
24. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rddicl1**
    セグメント数　**200**
25. RD エリアの名称　**PDBDICL2**
26. RD エリアの種類　**データディクショナリ LOB 用 RD エリア**（ストアドプロシジャのオブジェクトを格納）
27. RD エリアを管理するサーバの名称　**dics**
28. ページ長　**8192 バイト**
29. セグメントサイズ　**1 ページ**
30. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys02¥rddicl2**
    セグメント数　**2000**
31. RD エリアの名称　**PDBUSER01**
32. RD エリアの種類　**ユーザ用 RD エリア（公用）**
33. RD エリアを管理するサーバの名称　**bes1**
34. ページ長　**4096 バイト**
35. セグメントサイズ　**150 ページ**
36. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys03¥rduser01**
    セグメント数　**50**
37. RD エリアの名称　**PDBULOB01**
38. RD エリアの種類　**ユーザ LOB 用 RD エリア（公用）**
39. RD エリアを管理するサーバの名称　**bes1**
40. ページ長　**8192 バイト**
41. セグメントサイズ　**1 ページ**
42. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys03¥rdulob01**
    セグメント数　**50**
43. RD エリアの名称　**PDBUSER02**
44. RD エリアの種類　**ユーザ用 RD エリア（私用）**

認可識別子　**USERID**

45. RD エリアを管理するサーバの名称　**bes2**
46. ページ長　**4096 バイト**
47. セグメントサイズ　**50 ページ**
48. RD エリアを構成する HiRDB ファイルの指定

    名称　**c:¥dbarea¥rdsys04¥rduser02**

    セグメント数　**100**

4 データベース定義ユティリティ（pddef）

この章では，スキーマの定義や内容の変更などをするデータベース定義ユティリティ（pddef）について説明します。

4.1 概要

4.1.1 機能

データベース定義ユティリティ（pddef）は，スキーマの定義や内容の変更などをするユティリティです。

標準入力から入力した定義系 SQL を実行し，スキーマの定義や内容の変更などをしてから，その実行結果を標準出力に出力します。また，定義情報はディクショナリ表に格納されます。

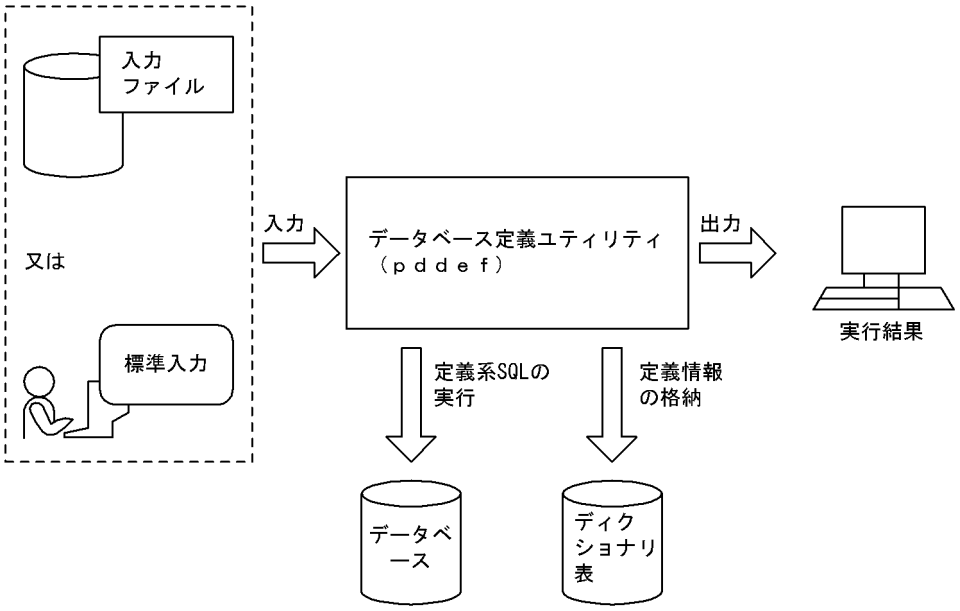
なお，データベース定義ユティリティを実行する前に，クライアント環境定義を設定しておく必要があります。最低限，PDHOST，PDNAMEPORT，及び PDUSER は指定してください。指定例を次に示します。

```
PDHOST=host1
PDNAMEPORT=20001
PDUSER=¥""k916321"¥"/¥""k916321"¥"
        :
```

クライアント環境定義については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

データベース定義ユティリティ（pddef）の概要を次の図に示します。

図 4–1　データベース定義ユティリティ（pddef）の概要



4.1.2 実行者

CONNECT 権限を持っているユーザが実行できます。ただし，実行する SQL によって必要な権限が異なります。SQL 実行時の権限については，マニュアル「HiRDB SQL リファレンス」を参照してください。

4.2　コマンドの形式

4.2.1　形式

```
pddef 〔-e〕 〔-s〕 〔-R〕 {定義系SQL | 入力ファイル}
```

4.2.2　オプション

- **-e**

SQL 文の終了指定として，END_FUNC;（CREATE FUNCTION の場合），END_TYPE;（CREATE TYPE の場合），又は END_TRIG（CREATE TRIGGER の場合）を追加する場合に指定します。

CREATE FUNCTION の場合：

1 行の最後に END_PROC;又は END_FUNC;があれば，そこまでを SQL 文とみなします。

CREATE TYPE の場合：

1 行の最後に END_PROC;又は END_TYPE;があれば，そこまでを SQL 文とみなします。

CREATE TRIGGER の場合：

1 行の最後に END_PROC;又は END_TRIG;があれば，そこまでを SQL 文とみなします。

SQL 文の継続行の扱いについては，「継続行の扱い」を参照してください。

- **-s**

定義系 SQL を直接入力する場合，入力待ちのプロンプトを表示するときに指定します。

プロンプトは，入力状況に応じて次の二つが表示されます。

COMMAND ?：通常の入力待ちを示します。

NEXT ?：SQL 文の継続行の入力待ちを示します。

表示例を次に示します。

```
pddef -s
COMMAND ? +----2----+----3----+----4----+----5----+----6----+----7----+

CREATE TABLE T1 (
NEXT ?    +----2----+----3----+----4----+----5----+----6----+----7----+

C1 INT,
NEXT ?    +----2----+----3----+----4----+----5----+----6----+----7----+

C2 CHAR(64)
```

```
NEXT ?    +----2----+----3----+----4----+----5----+----6----+----7----+

);
KFPA12000-I Processing of SQL completed
COMMAND ? +----2----+----3----+----4----+----5----+----6----+----7----+
```

- **{定義系 SQL ｜入力ファイル}**

定義系 SQL を記述します。

定義系 SQL を直接入力する方法と，定義系 SQL を記述した入力ファイルを指定する方法があります。定義系 SQL については，マニュアル「HiRDB SQL リファレンス」を参照してください。

定義系 SQL を直接入力する場合：

pddef を実行すると SQL 文の入力待ちになります。そのときに，定義系 SQL を入力してください。pddef の後に続けて SQL 文を入力できません。それぞれの SQL 文の最後にはセミコロン（；）を指定してください。

pddef を終了する場合，＜コントロール＞＋＜Ｚ＞を押して，Enter キーを押してください。

入力ファイルを指定する場合：

あらかじめメモ帳などで定義系 SQL を記述した入力ファイルを作成します。リダイレクションによって標準入力の入力元を，その入力ファイルに変更します。

（例）`pddef <infile`

- **-R**

実行した SQL の一部又はすべてで SQLCODE が負の場合（HiRDB 接続時の異常終了も含む），pddef のリターンコードを 4 にするときに指定します。このオプションは，pddef 実行時の SQL 中のエラーを検知したい場合に指定します。

4.3 継続行の扱い

4.3.1 CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER 以外の SQL の場合

1 行の最後にセミコロン（；）があれば，そこまでを SQL 文とみなして処理します（セミコロンの後ろの空白及び制御文字は無視されます）。セミコロンがない場合は，その行は次の行に継続しているものとして扱われます。

（例）

```
CREATE TABLE T1(C1 INT,      (継続)
            C2 CHAR(10),    (継続)
            C3 CHAR(10));   (SQL文の終了)
```

SQL 文中に引用符（''）で囲まれたセミコロンが行の最後に記述してある場合は，SQL 文の終了とみなされます。

4.3.2 CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER の場合

CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER での SQL 文の終了指定を次の表に示します。1 行の最後に次の表に示す文字列があれば，そこまでを SQL 文とみなします（セミコロンの後ろの空白及び制御文字は無視されます）。

表 4–1 CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER での SQL 文の終了指定

SQL 文	-e オプション	
	指定なし	指定あり
CREATE PROCEDURE	END_PROC;	END_PROC;
CREATE FUNCTION	END_PROC;	END_PROC;又は END_FUNC;
CREATE TYPE	END_PROC;	END_PROC;又は END_TYPE;
CREATE TRIGGER	END_PROC;	END_PROC;又は END_TRIG;

CREATE PROCEDURE，CREATE FUNCTION，CREATE TYPE，及び CREATE TRIGGER の場合は，1 行の最後にセミコロンがあっても次の行に継続しているものとして扱われます。

（例）

```
CREATE PROCEDURE P1()                    (継続)
BEGIN                                    (継続)
  INSERT INTO T1(C1) VALUES(10);         (継続)
  UPDATE T1 SET C1 = 0 WHERE C1 < 0;     (継続)
END                                      (継続)
END_PROC;                                (SQL文の終了)
```

SQL 文中に引用符（''）で囲まれた END_PROC；が行の最後に記述してある場合は，SQL 文の終了とみなされます。

4.3.3 SQL 文の途中で EOF を検知した場合

SQL 文の途中（セミコロンがない状態）で EOF を検知した場合，コマンドの終了とみなされ，それまでに入力した SQL 文が実行されます。

（例）

```
CREATE TABLE T1(C1 INT,                  (継続)
                C2 CHAR(10)              (継続)
EOF                                      (コマンドの終了)
```

4.4　規則及び注意事項

4.4.1　規則

1. データベース定義ユティリティは，HiRDB が稼働中のときだけ実行できます。
2. データベース定義ユティリティは，任意のサーバマシンから実行できます。
3. 定義系 SQL を実行すると，1SQL ごとにコミット処理を実行します。
4. 実行できる SQL 文の長さは，最大 2,000,000 バイトです。
5. 1 行に複数の定義系 SQL は記述できません。
6. データベース定義ユティリティで CREATE INDEX を実行する場合，インデクスを作成する表に行データが格納されていると，CREATE INDEX を実行するときにインデクスを一括作成します。このとき，クライアント環境定義に PDDBLOG=NO を指定していると，データベースの更新ログ取得しないでインデクスを一括作成します。したがって，この場合 HiRDB 管理者はログレスモードで運用する必要があります。ログレスモードの運用方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
7. データ連動の対象となる抽出側データベースに対して，ALTER TABLE 又は DROP TABLE は実行しないでください。実行した場合，整合性が保証されません。ただし，対象となるデータベース中の表又は列が更新中でなく，かつ既に更新情報がすべて反映側に転送されている場合には，HiRDB 側の定義を変更してください。そして，それに合わせて HiRDB Datareplicator 側での抽出となる表又は列の抽出定義を hdeprep コマンドで再度解析してから HiRDB Datareplicator を再起動すると，整合性が保証されます。これらの手順については，マニュアル「HiRDB データ連動機能 HiRDB Datareplicator」を参照してください。
8. システム定義の pd_sql_simple_comment_use オペランドが Y の場合，N の場合と制御文字の扱いが異なるため，制御文字を含む SQL の動作が次のように変わります。
   - 改行コードを含め，2,000,000 バイトを超える SQL がエラーになります。
   - 字句中に制御文字を含む SQL がエラーになります。又はオペランドの値によって結果が異なります。
9. pddef を実行する場合，文字コードは定義を登録する HiRDB サーバと同じになるように設定し，設定した文字コードで入力してください。

4.4.2　注意事項

1. データベース定義ユティリティの結果は，ユティリティ実行時のリターンコード，又はディクショナリ表を検索することで確認できます。リターンコードの意味を次に示します。
   - -R オプションを省略した場合

リターンコード	意味	対策方法	HiRDB の動作
0	正常終了（SQL がエラーとなった場合も含む）	該当しません。	該当しません。
8	メモリ不足	ほかのプロセスが終了してから再度実行するか，又はメモリを増やしてください。	処理を中断して，pddef を終了します。
	オプション指定エラー	オプションを正しく指定してください。	
	内部通信（オペレーションコード SINF）エラー	出力されるメッセージに従って対処してください。	

- -R オプションを指定した場合

リターンコード	意味	対策方法	HiRDB の動作
0	正常終了	該当しません。	該当しません。
4	HiRDB 接続時のエラー	設定している環境変数を見直してください。	処理を中断して，pddef を終了します。
	SQL エラー	実行した SQL を見直してください。	処理を続行します。
8	メモリ不足	ほかのプロセスが終了してから再度実行するか，又はメモリを増やしてください。	処理を中断して，pddef を終了します。
	オプション指定エラー	オプションを正しく指定してください。	
	内部通信（オペレーションコード SINF）エラー	出力されるメッセージに従って対処してください。	

注

リターンコード 4 が発生し，その後にリターンコード 8 が発生した場合，処理を中断し，最終リターンコード 8 で終了します。

2. HiRDB 接続時のパスワード秘匿化機能を使用している場合，必要に応じて，クライアント環境変数 PDAUTHTYPE を設定し，HiRDB へ接続するときの認証方式を指定してから実行してください。指定値についてはマニュアル「HiRDB システム運用ガイド」の「HiRDB 接続時のパスワード秘匿化機能の設定方法」を参照してください。
3. 通信情報ファイルディレクトリ変更機能を使用している場合，クライアント環境定義 PDIPCFILEDIR に値を設定する必要があります。指定値については，マニュアル「HiRDB UAP 開発ガイド」の「クライアント環境定義の設定内容」を参照してください。

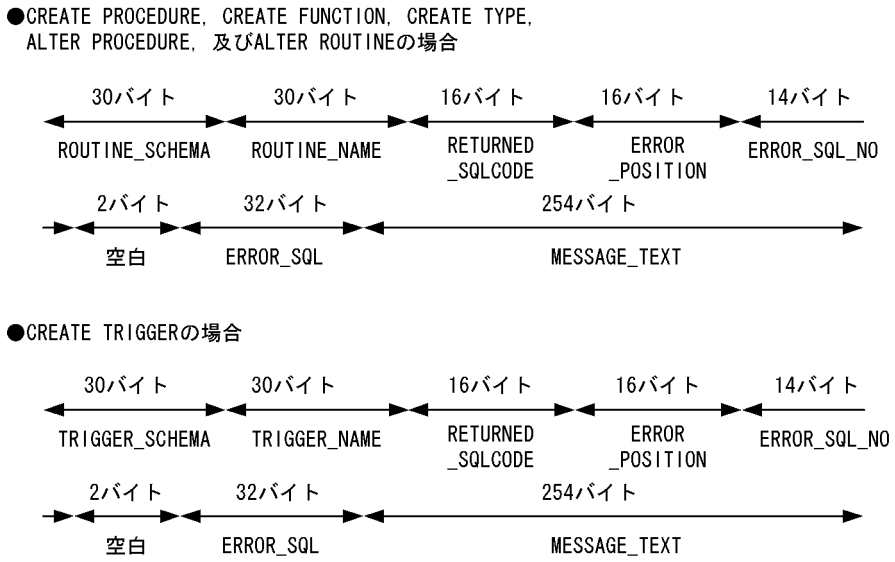
4.4.3 SQL エラー時に出力される情報

次の SQL 文でエラーが発生した場合，GET DIAGNOSTICS 文が実行され，エラー情報が出力されます。ただし，SQL 文中に注釈がある場合，ERROR_POSITION には注釈を除いた値が表示されます。

- CREATE PROCEDURE
- CREATE FUNCTION
- CREATE TYPE
- ALTER PROCEDURE
- ALTER ROUTINE
- CREATE TRIGGER

エラー情報の形式を次の図に示します。

図 4–2　エラー情報の形式



4.4.4　注釈（コメント）の扱い

SQL 文の最初に現れた「/*」以降を注釈とみなします。それ以降，指定したすべての文字（セミコロンを含む）を注釈として扱います。また，注釈の開始を示す「/*」以降，最初に現れた「*/」を注釈の終了とみなします。この注釈を囲み注釈といいます。ただし，システム定義の pd_sql_simple_comment_use オペランドが Y の場合は，SQL 文中の「--」から，それ以降の最初の改行コードまでのすべての文字（セミコロンを含む）を注釈とみなします。この注釈を単純注釈といいます。

例を次に示します。なお，下線部が注釈とみなされます。

```
 （例1）
CREATE TABLE T1 (C1 INT) ; /* CU ADDRESS TABLE */   …注釈の開始・終了

 （例2）
CREATE TABLE T1 /* CU ADDRESS TABLE */ (C1 INT) ;   …注釈の開始・終了

 （例3）
CREATE TABLE T1 /* CU ADDRESS   …注釈の開始・継続
TABLE */ (C1 INT) ;   ……注釈の終了
```

```
 (例4)
CREATE TABLE T1 /* CU ADDRESS TABLE * (C1 INT) ;    …注釈の開始・継続

 (例5)
CREATE TABLE T1 /* CU /* ADDRESS */ TABLE */ (C1 INT) ;  …注釈の開始・終了

 (例6)
CREATE TABLE T1 -- CU ADDRESS  …注釈の開始・終了
(C1 INT) ;
```

引用符（''）やアポストロフィ（'）で囲まれた「/* ～ */」及び「-- ～ 改行コード」は注釈とみなされません。注釈の記述方法については，マニュアル「HiRDB SQL リファレンス」の SQL の記述形式を参照してください。また，注釈の扱いは会話型 SQL 実行ユティリティ（HiRDB SQL Executer）とは異なるので注意してください。

4.4.5　CONNECT 後の動作

CONNECT 後にサーバと内部通信します（オペレーションコード SINF）。この内部通信がエラーになると，SQL エラーが出力され，処理を終了します。

4.4.6　SQL 文中の制御文字の扱い

データベース定義ユティリティを実行した場合，条件によって SQL 文中の制御文字の扱いが異なります。条件ごとの制御文字の扱いを次に示します。

表 4–2　SQL 文中の制御文字の扱い

項番	システム定義の pd_sql_simple_comment_use オペランドの値	改行コードの削除	改行コード以外の制御文字の空白変換
1	N	○	○
2	Y	×	×

（凡例）

○：削除又は変換します。

×：削除又は変換しません。

4.4.7　BOM を持つファイルの使用

pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pddef の入力ファイルには，BOM を持つファイルを使用できます。なお，pddef の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。

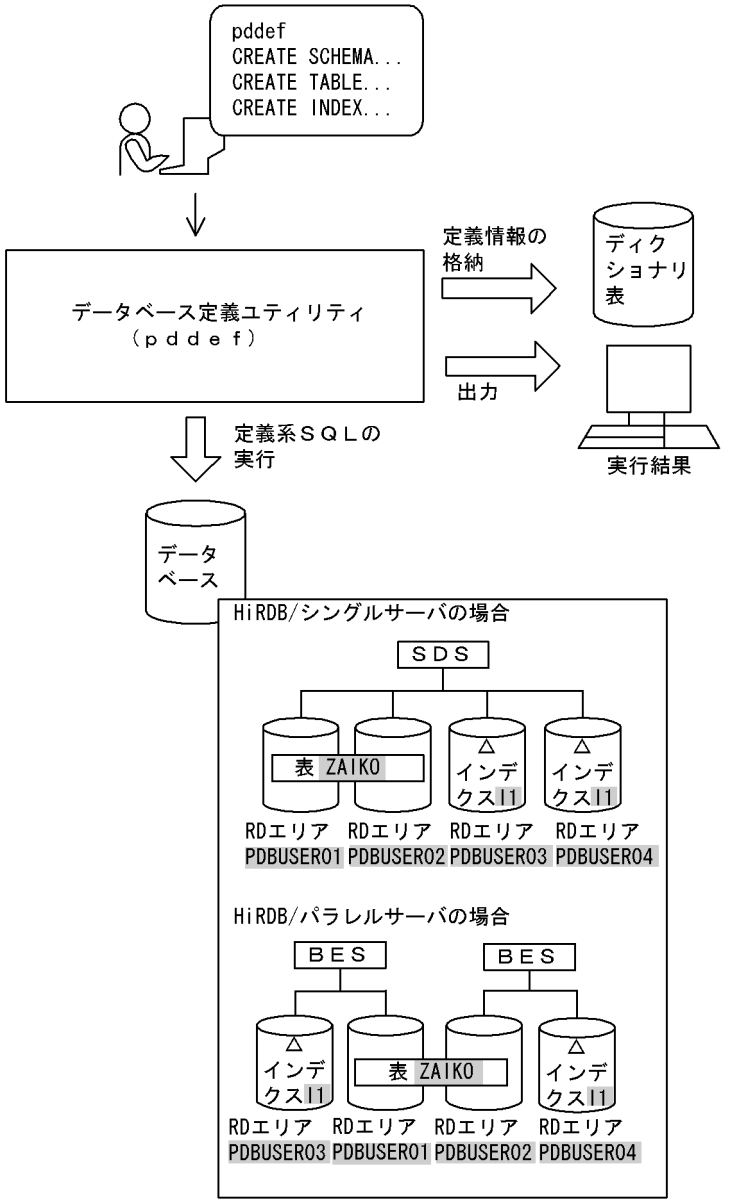
4.5 使用例

データベース定義ユティリティの使用例を，例 1 に示します。

例 1

次に示す表及びインデクスを定義します。
　・表　ZAIKO
　・インデクス　I1
なお，次に示すRDエリアが作成されているものとします。
　・PDBUSER01
　・PDBUSER02
　・PDBUSER03
　・PDBUSER04

[概要]



（凡例） ：データベース定義ユティリティの定義系ＳＱＬでの指定値
ＳＤＳ：シングルサーバ
ＢＥＳ：バックエンドサーバ

[コマンド実行例]

・定義系 SQL を直接入力する場合

```
pddef
  CREATE SCHEMA;   ....................................1
  CREATE TABLE ZAIKO(SCODE CHAR(4),SNAME NCHAR(8),
                     COL NCHAR(1),TANKA INTEGER,
                     ZSURYO INTEGER)
                IN((PDBUSER01) SCODE<'302S',
                   (PDBUSER02) SCODE>='302S');   .........2
  CREATE INDEX I1 ON ZAIKO(SCODE)
                IN ((PDBUSER03),(PDBUSER04));   ..........3
<<コントロール>＋<Ｚ>, Enterキーで終了>
```

[説明]

1. スキーマの定義
2. 表（ZAIKO）の定義
3. インデクス（I1）の定義

・入力ファイルを指定する場合

```
pddef < c:¥hirdb¥pddef¥crfile
```

[説明]

入力ファイルの指定　**c:¥hirdb¥pddef¥crfile**

入力ファイル（**c:¥hirdb¥pddef¥crfile**）の内容

```
CREATE SCHEMA;   ......................................1
CREATE TABLE ZAIKO(SCODE CHAR(4),SNAME NCHAR(8),
                   COL NCHAR(1),TANKA INTEGER,
                   ZSURYO INTEGER)
             IN((PDBUSER01) SCODE<'302S',
                (PDBUSER02) SCODE>='302S');   ............2
CREATE INDEX I1 ON ZAIKO(SCODE)
             IN ((PDBUSER03),(PDBUSER04));   ............3
```

[説明]

1. スキーマの定義
2. 表（ZAIKO）の定義
3. インデクス（I1）の定義

5 データベース作成ユティリティ（pdload）

この章では，ユーザが用意したデータを表に格納するデータベース作成ユティリティ（pdload）について説明します。

なお，IVS 対応 UTF-8 文字コード及び IVS 対応 UTF-16 文字集合を使用する場合は，この章の記載を次のように読み替えてください。

UTF-8 -> IVS 対応 UTF-8

utf-8 -> utf-8_ivs

UTF-16 -> IVS 対応 UTF-16

utf-16be -> utf-16ibe

utf-16le -> utf-16ile

5.1　機能

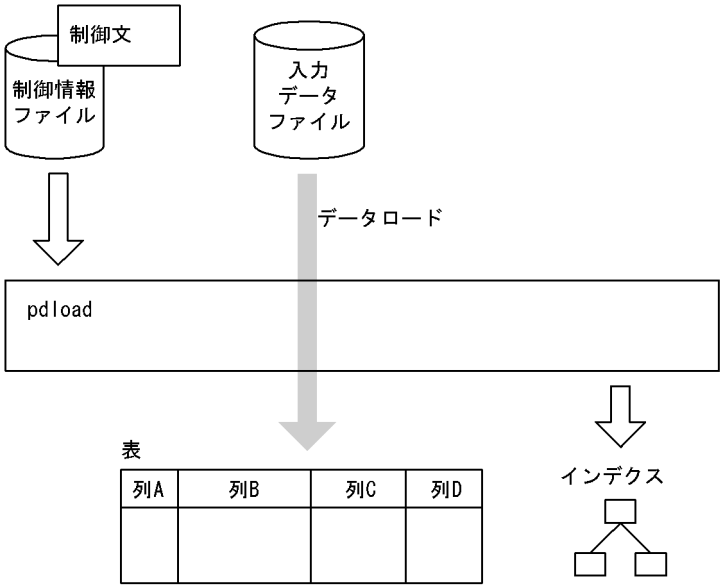
データベース作成ユティリティ（pdload）は，ユーザが用意したデータを表へデータロードするユティリティです。

5.1.1　データロードとは

ユーザが用意したデータを入力し，表へデータを格納します。これを**データロード**といいます。

データロードの概要を次の図に示します。

図 5–1　データロードの概要



(1)　入力データファイル

入力データファイルとは，ユーザが用意したデータを格納したファイルのことです。

使用できる形式としては，全部で 4 種類あります。基本的には次の 2 種類のうち，どちらかの形式を使用します。

DAT 形式：

列データを文字で記述する形式です。一般的には，CSV 形式と呼ばれている形式です。

- **pdload が列データを HiRDB 内部形式に変換して格納するため，HiRDB 以外のシステムのデータを基に表を作成する場合に適しています。**

DAT 形式の記述例：

```
中村,36,1958-10-15,yokohama
          :
```

また，DAT 形式には，更に拡張機能が使用できる**拡張 DAT 形式**もあります。形式は DAT 形式と基本的に同じですが，囲み文字の変更などの拡張機能を使用できます。

バイナリ形式：

列データを HiRDB 内部形式で記述する形式です。

- **HiRDB 内部形式への変換処理が不要となるため，性能面で優れています。大容量のデータを格納する場合，性能を要求される場合に適しています。**

バイナリ形式の記述例：

```
928691ba8140814081402400000003f0080000800796f6b6f68616d61
<----------------><------><------><----------------->
        中村           36        1          yokohama
           :

注　上段は実際のデータを，下段はそのデータの内容を示しています。
```

また，次のように，1 行固定で記述する形式，及び pdrorg が出力する形式も使用できます。

固定長データ形式：

1 行の長さ，及び各列データの開始位置（行の先頭からのオフセット）を，すべての行が同じになるように記述します。記述する形式としては，DAT 形式，又はバイナリ形式のどちらかで記述します。

- **セパレータ文字で区切られていないテキストデータや，データ格納順序などを変更したいバイナリ形式のデータを基に表を作成する場合に適しています。**

pdrorg 出力バイナリ形式：

pdrorg を使用してバイナリ形式で出力したアンロードデータファイルです。

- **他 HiRDB システムからデータを移行するときなどに使用します。**

(2)　制御情報ファイル

制御情報ファイルには，pdload の制御文を記述します。

制御文には，入力データファイルの指定，インデクス情報の指定，LOB 列の情報の指定，ファイル出力先ディレクトリの指定などを記述します。なお，制御文にファイル出力先ディレクトリを指定しない場合，「制御文で指定しない場合のファイル出力先ディレクトリ」に示すディレクトリにファイルを出力します。

(3)　表

表は，あらかじめ定義されている必要があります。

(4)　表の属性によるデータロード方法

(a)　表にインデクスが定義されている場合

同時にインデクスも作成できます。

また，データロード時にはインデクス情報だけ出力しておいて，後から pdrorg でインデクスを作成することもできます。

(b)　横分割表の場合

表単位，又は RD エリア単位にデータロードできます。

(c)　監査証跡表の場合

セキュリティ監査機能を使用する場合，監査証跡ファイルに出力された監査証跡を監査証跡表にデータロードできます。

通常の表（ユーザが定義する表）に対するデータロードか，監査証跡表に対するデータロードかは，制御情報ファイルの srcuoc 文で指定します。srcuoc 文については，「srcuoc 文（UOC 格納ライブラリの情報の記述）」を参照してください。

5.1.2　LOB 列がある表へのデータロード

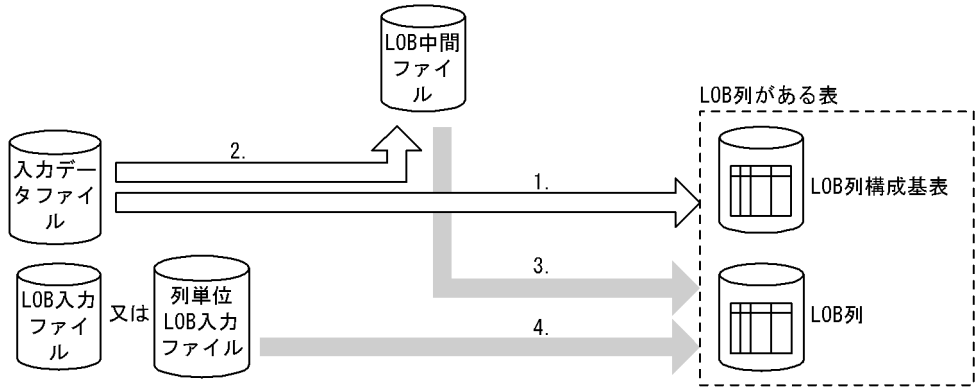
(1)　データロード方法

LOB 列がある表の場合，LOB 列構成基表と LOB 列へ同時にデータロードできます。また，別々にデータロードすることもできます。**LOB 列構成基表**とは，LOB 列がある表の LOB 列を除いた部分のことをいいます。また，**LOB 列**とは，データ型が BLOB の列のことをいいます。

LOB 列だけを後からデータロードするのは，表に複数の LOB 列がある場合に，LOB 列へのデータロードを，列単位に分けて複数同時実行するための機能です。ただし，LOB 列構成基表のデータロードだけ完了した状態では，LOB 列へのデータロードが未完というデータベースとしては不安定な状態なので，該当する表の更新は PURGE TABLE 文を含めて絶対にしないでください。別々にデータロードする場合は，他ユーザからの更新を抑止するために，RD エリアをコマンド閉塞にしておくことをお勧めします。

LOB 列がある表へのデータロード方法を次の図に示します。

図 5–2　LOB 列がある表へのデータロード方法



［説明］
●LOB列構成基表とLOB列へ，同時にデータロードする場合
1回のデータロードの実行で，pdloadが1～4を行います。
●LOB列構成基表とLOB列へ，別々にデータロードする場合
実行順序は，必ず(a)，(b)の順にしてください。
(a) LOB列構成基表へのデータロード（1，及び2）
LOB列構成基表へのデータロードを実行します。この場合，まずLOB列構成基表にデータロードし，更に入力データファイル中のLOBデータに関する情報をLOB中間ファイルに出力します。
(b) LOB列へのデータロード（3，及び4）
LOB列へのデータロードを実行します。この場合，(a)で出力されたLOB中間ファイルからLOBデータに関する情報もデータロードします。

LOB 入力ファイルとは，1 列 1 行ごとに用意するファイルです。したがって，LOB 列数×行数分のファイルが必要となります。

列単位 LOB 入力ファイルとは，1 列ごとに用意するファイルです。したがって，LOB 列数分のファイルが必要となります。

(2)　入力データ

LOB 列がある表へデータロードする場合，入力データは次のどちらかのパターンで用意します。

入力データファイル	LOB 入力ファイル
LOB 列以外のデータ，LOB 入力ファイル名を記述	格納データ数分の LOB 入力ファイルを用意
LOB 列以外のデータ，LOB データを記述	不要

5.1.3　プラグインが提供する抽象データ型列（LOB 属性）がある表へのデータロード

(1)　データロード方法

プラグインが提供する抽象データ型列（LOB 属性）がある表へデータロードする場合，LOB 列構成基表と LOB 列へ同時にデータロードします。

(2) 入力データ

プラグインが提供する抽象データ型列（LOB 属性）がある表へデータロードする場合，入力データは次のどちらかのパターンで用意します。

入力データファイル	LOB パラメタのファイル
LOB 属性の抽象データ型列以外のデータ，LOB パラメタのファイル名を記述	格納データ数分の LOB パラメタのファイルを用意
LOB 属性の抽象データ型列以外のデータ，LOB パラメタを記述	不要

LOB パラメタとは，抽象データ型列に格納する値を生成する，コンストラクタ関数の入力パラメタとなる LOB データのことをいいます。

5.1.4 文字集合を定義した表へのデータロード

文字集合の列がある表にデータロードする場合，入力データ中の文字コードを列の文字集合に指定した文字コードに変更します。入力データファイルの形式とデータロードする表の組み合わせによって，文字コード変換の有無は異なります。文字コード変換の有無を次の表に示します。

表 5–1 文字コード変換の有無

入力データファイルの形式			データロード対象表の文字集合定義列の有無	
			あり	なし
DAT 形式			○	×
バイナリ形式			×※1	
固定長データ形式			○	
pdrorg が出力したファイル	DAT 形式		○	
	バイナリ形式	pdload 用アンロードファイル※3	○	
		pdrorg 用アンロードファイル※3	○※2	
		表移行用アンロードファイル※3	○	

（凡例）

○：文字コード変換をします。

×：文字コード変換をしません。

注※1

データベースには文字コード変換しないで格納するため，格納データは保証しません。

注※2

pdrorg 実行時は-k unld 指定でアンロードを行い，既定義型の文字コードで出力したアンロードデータファイルを使用してください。-k rorg 指定で再編成を行い，文字集合が定義された列のコードで出力したアンロードデータファイルを使用するとKFPL15047-E メッセージを出力して処理を打ち切ります。

注※3

詳細については，「pdrorg 出力バイナリ形式ファイル」を参照してください。

文字集合の列がある表にユーザが作成したバイナリ形式の入力データファイルでデータロードする場合，pdload は文字コード変換をしないため，入力データファイルのデータを文字集合定義列の文字コードにしておく必要があります。入力データファイルを作成するときに使用する文字コードを次の表に示します。

表 5–2　入力データファイルを作成するときに使用する文字コード

入力データファイルの形式	データロード対象表の文字集合定義列の有無	
	あり	なし
DAT 形式	既定文字集合	既定文字集合
バイナリ形式	列に指定した文字コード	
固定長データ形式	既定文字集合	

(1)　注意事項

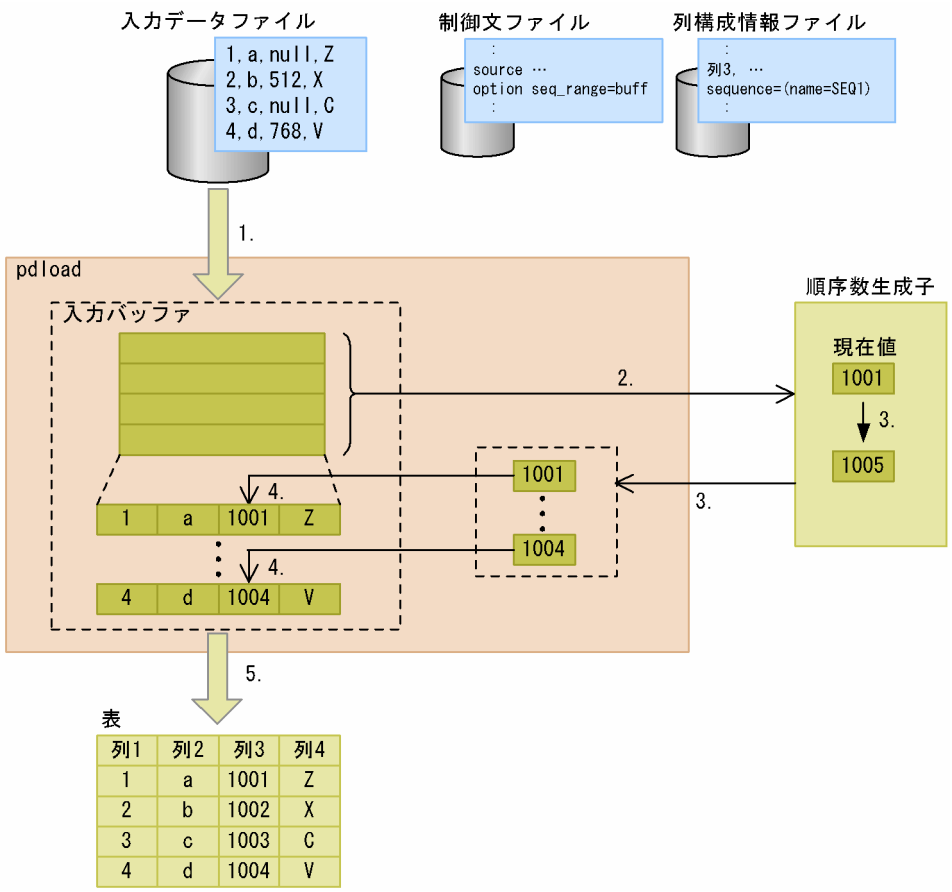
- pdrorg で-W オプションを指定して FIX 表のデータをアンロードした場合，出力されたアンロードデータファイルの形式はユーザが作成したバイナリ形式ファイルと同じです。このため，文字集合の列がない表へは pdload 実行時に-W オプションを省略しても正しくデータロードできます。しかし，文字集合の列がある表にデータロードするとき，アンロードデータファイルは既定文字集合の文字コードで作られているため，-W オプションを省略すると文字コード変換をしないで文字集合定義列にデータロードしてしまいます。必ず-W を指定してください。
- 文字集合の列を分割キーとした場合は，文字コード変換後の値で分割条件の判定を行います。

5.1.5　自動採番機能を使用したデータロード

自動採番機能を使用してデータロードすると，表の列に，順序数生成子が生成する順序番号を格納できます。

自動採番機能を使用したデータロードを次の図に示します。

図 5–3　自動採番機能を使用したデータロード



[説明]

1. 入力データファイルから，入力バッファにデータを読み込みます（ここでは，順序番号の取得方式としてバッファ単位取得方式を使用していると仮定します）。
2. 読み込んだ行数分の順序番号を確保するため，順序数生成子をコールします。
3. 順序数生成子では，確保した順序番号を返し，次に割り当てる値を現在値に設定します。
4. pdload は，入力バッファからデータを読み込みながら，取得した順序番号を含めて行データを組み立てます。
5. 組み立てた行データをデータロードします。

自動採番機能を使用してデータロードする場合，入力データファイルは次のどれかの形式でなければなりません。

- DAT 形式
- 拡張 DAT 形式
- 固定長データ形式
- バイナリ形式
- pdrorg 出力バイナリ形式（pdload 用アンロードファイル）

(1)　順序番号の格納方式

自動採番機能を使用してデータロードする場合，入力データファイル内の，順序番号を格納する列の扱いを選択できます。順序番号の格納方式には次の 3 種類があります。

列データ全置換：

順序番号を格納する列に対して，入力データファイル中の該当する列データをすべて順序番号に置き換えます。

列データー部置換：

順序番号を格納する列に対して，入力データファイル中の該当する列データのうち，指定した置換条件に一致するデータだけ順序番号に置き換えます。

列データ追加：

順序番号を格納する列に対応するデータが入力データファイル中にない場合，順序番号を入力データとして追加します。なお，入力データファイルがバイナリ形式の場合，この方式は指定できません。

順序番号の格納方式は，列構成情報ファイル又はナル値・関数情報ファイルで指定します。

(2)　順序番号の取得方式

自動採番機能を使用してデータロードする場合，順序数生成子からの順序番号の取得タイミングと取得単位を選択できます。順序番号の取得方式には次の 3 種類があります。

全数一括取得方式：

データロード完了後に一括して順序数生成子の値を使用した順序番号にします。

指定単位取得方式：

指定した単位ごとに順序番号を取得しながらデータロードします。

バッファ単位取得方式：

入力データファイルの形式が固定長データ形式，バイナリ形式，又は pdload 用アンロードファイルの場合，入力バッファに読み込める行数分の順序番号を取得しながらデータロードします。入力データファイルの形式が DAT 形式，又は拡張 DAT 形式の場合，1 行ごとに順序番号を取得しながらデータロードします。

(3)　順序番号を格納できるデータ型

順序番号は次のデータ型の列に格納できます。

- INTEGER
- SMALLINT
- DECIMAL
- FLOAT
- SMALLFLT

- CHAR
- VARCHAR
- MCHAR
- MVARCHAR

これらのデータ型と順序数生成子のデータ型が異なる場合，pdload がデータ変換してデータロードします。変換後のデータが列のデータ型として格納できない場合，データロードはエラーとなります。

(4)　エラー時の処理

(a)　論理エラー発生時

入力データで論理エラーが発生した場合，エラーが発生する前に割り当てられた順序番号は欠番となります。

(b)　データ変換エラー発生時

「順序番号を格納できるデータ型」に示すデータ変換エラーが発生した場合，pdload は-e オプションの指定有無に関係なく，エラー終了します。この場合，option 文の dataerr オペランドの指定によって，それまでデータロードしたデータをコミットするか，又はロールバックするかを決定します。ロールバック指定の場合は，データロード中に使用した順序番号は欠番となります。

(5)　注意事項

自動採番機能を使用したデータロード実行時の注意事項を次に示します。

- 1 表の複数列に順序番号を格納する場合，各列に異なる順序数生成子を指定したときは，それぞれ指定した順序数生成子が生成した順序番号を格納します。また，各列に同じ順序数生成子を指定した場合は，同じ行の各列に同じ値の順序番号を格納します。
- UOC を利用して，ユティリティが入力デ−タファイルを入力してデータロードする場合，該当する列に対する順序番号を置換又は追加する前のデータを UOC に渡します。pdload は，UOC が編集したデータを受け取った後，該当する列に対して順序番号の置換又は追加をします。
- 繰返し列に対しては，自動採番機能を使用したデータロードはできません（エラーとなります）。繰返し列に順序番号を格納する場合は，次のどちらかの方法で対処してください。
  - HiRDB SQL Executer や UAP などを使用して，入力データに順序番号を含めて INSERT 文を実行する。
  - 自動採番機能を使用しないでデータロードし，HiRDB SQL Executer や UAP などを使用して，順序番号を格納する繰返し列に対して UPDATE 文を実行する。
- 自動採番機能を使用したデータロードでは，順序数生成子に対して排他を掛けます。特に，順序番号の取得方式が全数一括取得方式の場合はデータロード処理が終了するまで排他を掛けるため，ほかのユティリティや UAP が排他待ちとなることがあります。
- 順序番号を割り当てた後で論理エラーが発生した場合，入力データの形式が DAT 形式，拡張 DAT 形式，又は固定長データ形式のときは，エラーデータファイルに入力データの値が出力されます。

入力データの形式がバイナリ形式のときは，エラーデータファイルに順序番号が出力されます。

5.1.6　同期点指定のデータロード

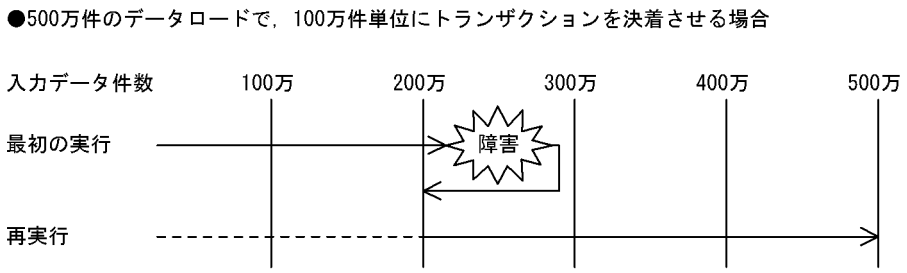
通常，データロードをするときは，すべてのデータを格納した後にトランザクションを決着します。ユティリティが途中で異常終了した場合は，最初の時点までロールバックしてしまいます。このため，再実行時は最初からし直すことになります。

同期点指定のデータロードの場合，任意のデータ件数を格納するごとにトランザクションを決着できます。これによって，ユティリティが異常終了した場合でもロールバック時間及びデータロード再実行時間を短縮できるので，大量のデータロードをするときに適しています。

なお，インデクスの作成処理（インデクス一括作成モード），及び LOB 列へのデータロード処理の場合は，同期点指定のデータロードは適用されません。この場合は，すべての処理が完了してからトランザクションが決着します。

同期点指定のデータロードをする場合，option 文（job オペランド）を指定します。同期点指定のデータロードの概要を次の図に示します。

図 5–4　同期点指定のデータロードの概要



[説明]

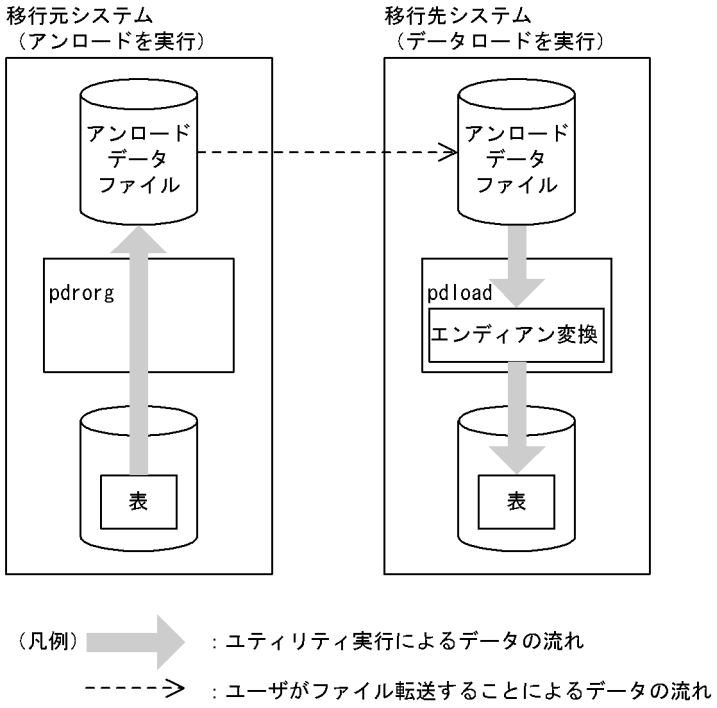
最初の実行では，障害発生後にロールバックします。200 万件の時点でトランザクションが決着しているため，ロールバックするのは 200 万件の時点までです。

再実行では，200 万件までのデータ格納処理をスキップし，それ以降のデータ格納処理をします。

5.1.7　エンディアンが異なるプラットフォーム間のデータ移行

既存の HiRDB システムを別の HiRDB システムに移行する場合，既存システムのデータを別システムに移行する必要があります。データ移行時，移行元システムと移行先システムでエンディアンが異なる場合，pdload が入力データファイルのエンディアンを変換して移行先システムにデータロードできます。このとき，入力データファイルには，pdrorg を使用してバイナリ形式でアンロードしたファイル（pdrorg 出力バイナリ形式ファイル）を指定します。エンディアンが異なるプラットフォーム間のデータ移行の概要を次の図に示します。

図 5–5　エンディアンが異なるプラットフォーム間のデータ移行の概要



[説明]

移行元システムで pdrorg を使用してデータをアンロードします。アンロードデータファイルを移行先システムに転送し，pdload でエンディアンを変換してデータロードします。

pdrorg 出力バイナリ形式ファイルについては，「pdrorg 出力バイナリ形式ファイル」を参照してください。

(1)　エンディアン変換の有無

入力データの制御情報中の数値データと数値型データをエンディアン変換します。FIX 表と非 FIX 表の入力データの変換有無をそれぞれ次の表に示します。

表 5–3　エンディアン変換の有無（FIX 表）

データ型	エンディアン変換の有無
INTEGER	○
SMALLINT	○
DECIMAL	×
FLOAT※1	○
SMALLFLT※1	○
CHAR	×※2
NCHAR	×※2
MCHAR	×※2

データ型	エンディアン変換の有無
DATE	×
TIME	×
TIMESTAMP	×
INTERVAL YEAR TO DAY	×
INTERVAL HOUR TO SECOND	×

（凡例）

○：エンディアン変換します。

×：エンディアン変換しません。

注※1

浮動小数点数はハードウェア表現によって精度が異なり，プラットフォーム間移行によってけた落ちが発生するおそれがありますが，これに関しては保証しません。

注※2

データ中に数値が入っていても，データ中のエンディアン変換はしません。

表 5–4　エンディアン変換の有無（非 FIX 表）

箇所	データ型	エンディアン変換の有無
基本行データの長さ及びオフセット部	–	○
既定義型の列データ値	INTEGER	○
	SMALLINT	○
	DECIMAL	×
	FLOAT※1	○
	SMALLFLT※1	○
	CHAR	×※2
	VARCHAR	△※2
	NCHAR	×※2
	NVARCHAR	△※2
	MCHAR	×※2
	MVARCHAR	△※2
	DATE	×
	TIME	×
	TIMESTAMP	×

箇所	データ型	エンディアン変換の有無
	INTERVAL YEAR TO DAY	×
	INTERVAL HOUR TO SECOND	×
	BINARY	△※2
	BLOB	△※2
コンストラクタ関数の引数のデータ値	INTEGER	○
	SMALLINT	○
	FLOAT※1	○
	SMALLFLT※1	○
	CHAR	×※2
	VARCHAR	△※2
	NCHAR	×※2
	NVARCHAR	△※2
	MCHAR	×※2
	MVARCHAR	△※2
	BINARY	△※2
	BLOB	△※2
繰返し列の要素数	−	○
繰返し列のナル値フラグ	−	×
繰返し要素値	−	※3
BINARY 引数のデータ値	−	×
BINARY 列のデータ値	−	×
BLOB 引数のデータ値	−	×※2
BLOB 列のデータ値	−	×※2

（凡例）

○：エンディアン変換します。

△：実データ長部分だけをエンディアン変換します。

×：エンディアン変換しません。

−：該当しません。

注※1

浮動小数点数はハードウェア表現によって精度が異なり，プラットフォーム間移行によってけた落ちが発生するおそれがありますが，これに関しては保証しません。

注※2

データ中に数値が入っていても，データ中のエンディアン変換はしません。

注※3

繰返し列の要素値は既定義型の列データ値で該当するデータ型を参照してください。

5.1.8　データロードする表が格納されている RD エリア

データロードする表を UAP がアクセスして不当に待ち状態にならないように，データロードの対象となる表が格納されている RD エリア（インデクスが定義されている場合はインデクスが格納されている RD エリアも含む）は，pdhold コマンドで閉塞しておくことをお勧めします。

なお，横分割表に RD エリア単位にデータロードを並列実行すると，表の占有時間は短くなります（データロードの処理時間は短縮されます）。この場合，RD エリア単位の入力データファイルが用意できていない（表単位の入力データファイルしかない）ときは，src_work 文を指定してください。src_work 文を指定すると，横分割表に RD エリア単位にデータロードを並列実行するときに必要となる，RD エリア単位の入力データファイルが作成されます。この作成されるファイルを**分割入力データファイル**といいます。分割入力データファイルを作成する場合のオプション，及び制御文の指定可否については，「特定機能使用時のオプション，及び制御文の指定可否」を参照してください。

5.1.9　pdload 実行時のログ取得方式

pdload を実行する場合，ログ取得方式は更新前ログ取得モード（-l p）にすることをお勧めします。

更新前ログ取得モードの場合，更新後のデータベース更新ログを取得しないため，その分の処理時間が短縮されます。したがって，ログ取得モードを適用したときに比べてユティリティの実行時間を短縮できます。

5.1.10　実行者

(1)　コネクト権限

pdload を実行するユーザ（-u オプション，又は PDUSER 環境変数に設定した認可識別子）は，HiRDB に対するコネクト権限が必要です（pdload はデータロード対象となる表の定義情報をディクショナリから検索するとき，HiRDB に接続するため）。

(2)　アクセス権限

基本的には，データロードする表の所有者が実行します。ほかのユーザが実行する場合は，データロードを実行するモードによって，次の権限が必要となります。ただし，監査証跡表へデータロードする場合，

HiRDB 管理者がコマンドを実行すると，パスワード入力要求のメッセージが表示されます。このとき，監査人がパスワードを入力してください。

追加モード（-d オプション指定なし）の場合：

INSERT 権限を持っているユーザが実行できます。

作成モード（-d オプション指定あり）の場合：

INSERT 権限，及び DELETE 権限を持っているユーザが実行できます。

(a) 注意事項

- 表移行用アンロードファイルを使用する場合（-b -W -w 指定時）は，データロードする表を所有するユーザで実行する必要があります。
- 表をインポートする場合（-b -W -w all 指定時）は，スキーマを所有するユーザで実行する必要があります。

5.2　代表的なデータロードの例

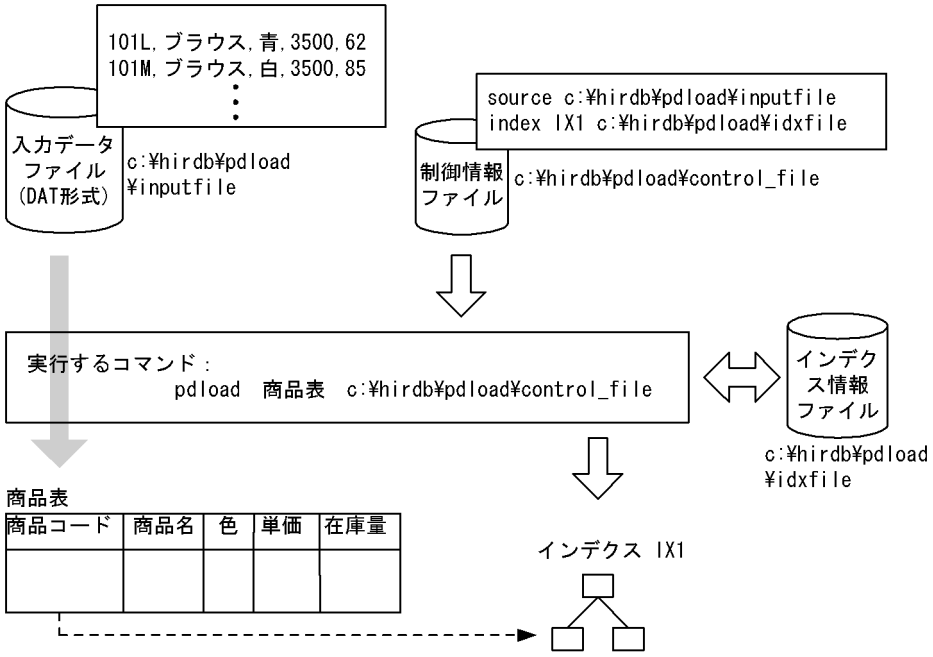
ここでは，代表的なデータロードの例として次の三つについて説明します。

- インデクスが定義されている表へのデータロード
- LOB 列がある表へのデータロード
- プラグインが提供する抽象データ型列がある表へのデータロード

5.2.1　インデクスが定義されている表へのデータロード

インデクスが定義されている表へのデータロードを次の図に示します。

図 5–6　インデクスが定義されている表へのデータロード



(1)　実行するコマンド

pdload で指定しているオプションについて説明します。

商品表：

　データロードする表の，表識別子です。

c:¥hirdb¥pdload¥control_file：

　制御情報ファイルのファイル名です。

その他，省略しているオプションには次の仮定値が適用されます。

-d 省略：追加モード

{-a ｜-b} 省略：入力データファイルが DAT 形式

-i 省略：インデクス一括作成モード

-l 省略：更新前ログ取得モード

(2) 制御情報ファイル

(a) source 文

c:¥hirdb¥pdload¥inputfile という名称の入力データファイルを指定しています。

(b) index 文

c:¥hirdb¥pdload¥idxfile という名称のインデクス情報ファイルを指定しています。

(3) 入力データファイル

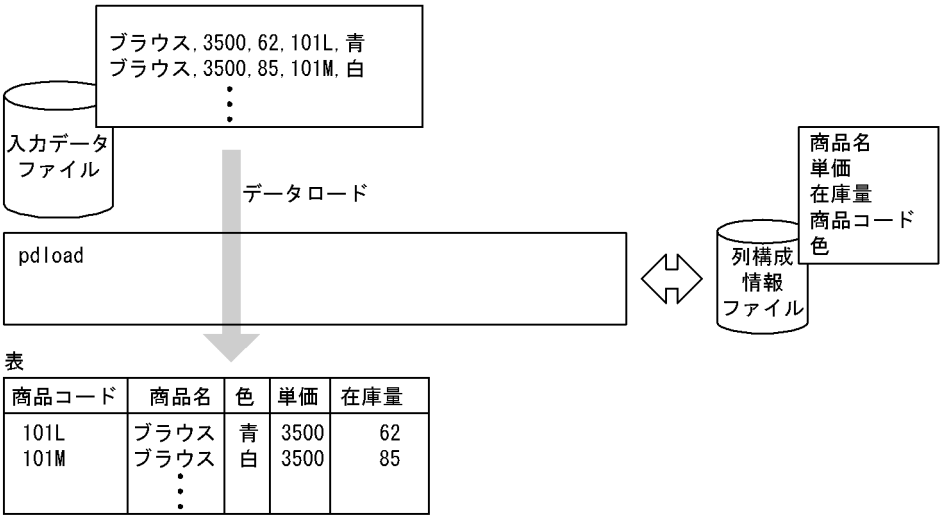
この例では，入力データファイルは DAT 形式で記述しています。

(4) 入力データと表の列との並び順が異なる場合

入力データファイルのデータの並びと，表の列の並びが異なる場合，**列構成情報ファイル**を指定して入力データと表の列の並びを対応させることができます。

列構成情報ファイルは，入力データファイルが DAT 形式，固定長データ形式，又は pdrorg 出力バイナリ形式（非 FIX のときだけ）の場合に使用できます。

例を次に示します。

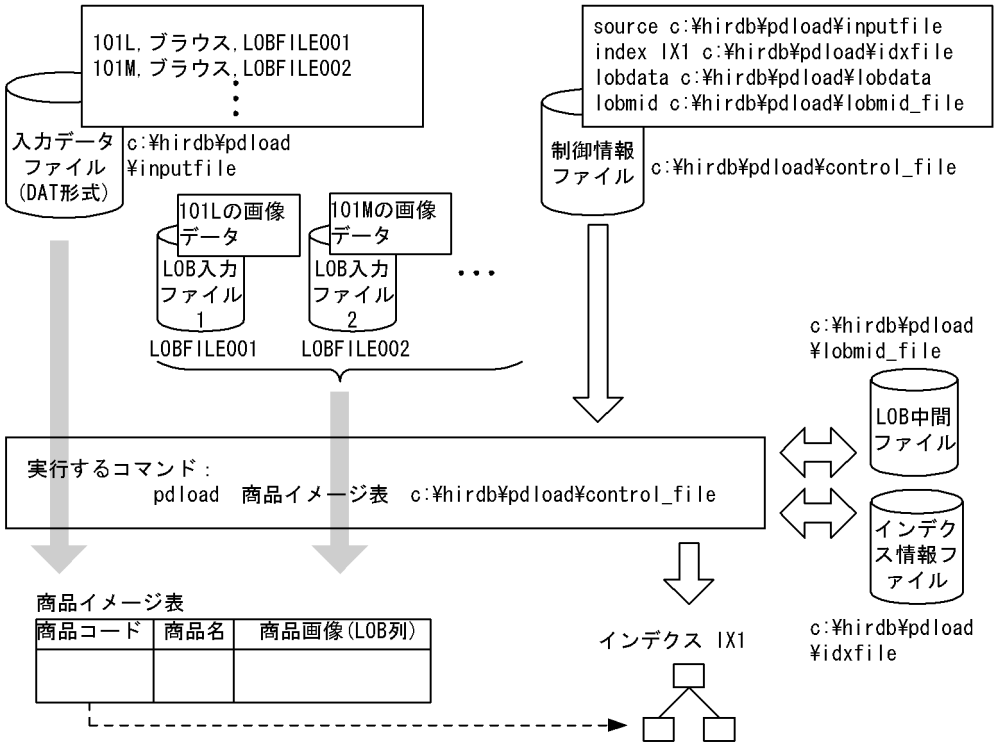


［説明］
入力データの並びと，表の列の並びが異なっています。
このような場合，列構成情報ファイルに，入力データ順に列名を記述します。

5.2.2　LOB 列がある表へのデータロード

LOB 列がある表へのデータロードを次の図に示します。

図 5–7　LOB 列がある表へのデータロード



(1)　実行するコマンド

pdload で指定しているオプションについて説明します。

商品イメージ表：

　データロードする表の，表識別子です。

c:¥hirdb¥pdload¥control_file：

　制御情報ファイルのファイル名です。

その他，省略しているオプションには次の仮定値が適用されます。

-d 省略：追加モード

{-a ｜-b} 省略：入力データファイルが DAT 形式

-i 省略：インデクス一括作成モード

-l 省略：更新前ログ取得モード

-k 省略：f（LOB 入力ファイルを LOB 列×行数分用意し，入力データファイル中にその LOB 入力ファイルのファイル名を記述）

(2)　制御情報ファイル

(a)　source 文

c:¥hirdb¥pdload¥inputfile という名称の入力データファイルを指定しています。

(b)　index 文

c:¥hirdb¥pdload¥idxfile という名称のインデクス情報ファイルを指定しています。

(c)　lobdata 文

c:¥hirdb¥pdload¥lobdata という名称の LOB 入力ファイルのディレクトリを指定しています。ここで指定したディレクトリ名と，入力データファイル中の LOB 入力ファイル名を合わせて，絶対パス名となるようにします。

(d)　lobmid 文

c:¥hirdb¥pdload¥lobmid_file という名称の LOB 中間ファイルを指定しています。

(3)　入力データファイル

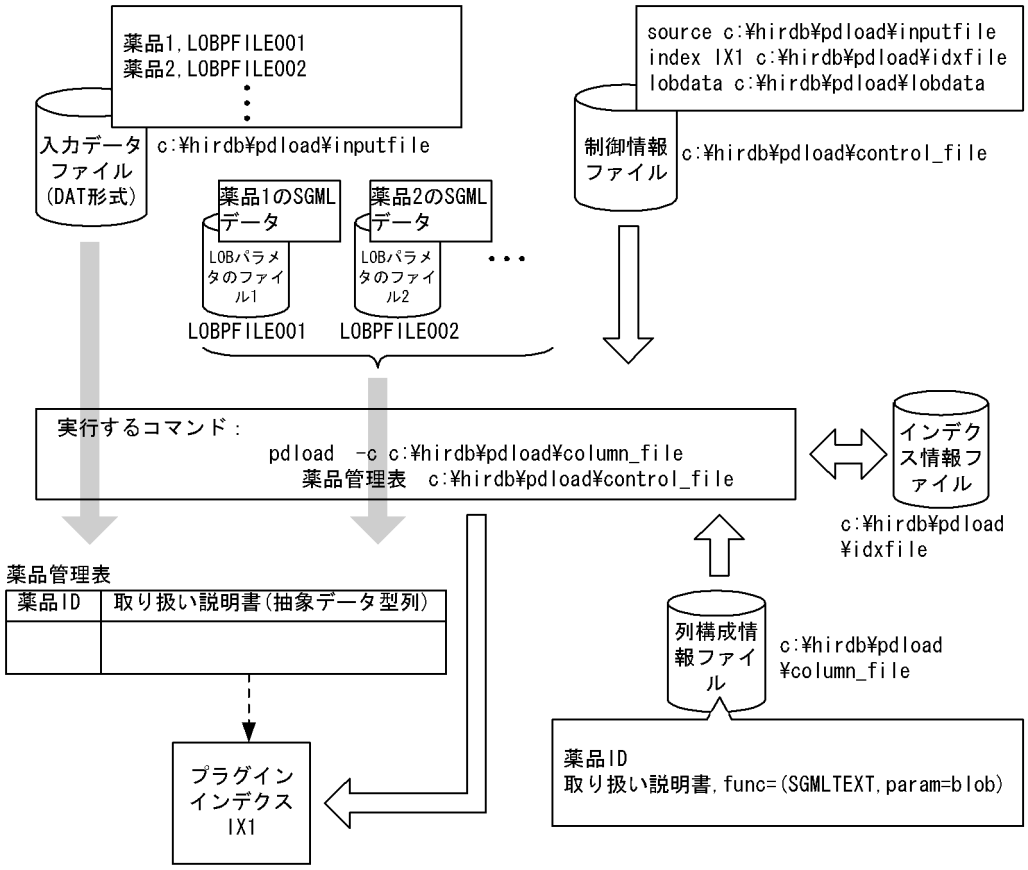
この例では，入力データファイルは DAT 形式で記述しています。

また，LOB 入力ファイルは別に用意し，入力データファイル中に LOB 入力ファイルのファイル名を記述しています。

5.2.3　プラグインが提供する抽象データ型列がある表へのデータロード

プラグインが提供する抽象データ型がある表へのデータロードを次の図に示します。なお，ここでは HiRDB Text Search Plug-in を例にして説明します。

図 5–8　プラグインが提供する抽象データ型がある表へのデータロード



(1)　実行するコマンド

pdload で指定しているオプションについて説明します。

-c c:¥hirdb¥pdload¥column_file：

列構成情報ファイルのファイル名です。

薬品管理表：

データロードする表の，表識別子です。

c:¥hirdb¥pdload¥control_file：

制御情報ファイルのファイル名です。

その他，省略しているオプションには次の仮定値が適用されます。

-d 省略：追加モード

{-a ｜-b} 省略：入力データファイルが DAT 形式

-i 省略：インデクス一括作成モード

-l 省略：更新前ログ取得モード

-k 省略：f（LOB パラメタのファイルを用意し，入力データファイル中にその LOB パラメタのファイルのファイル名を記述）

(2)　制御情報ファイル

(a)　source 文

c:¥hirdb¥pdload¥inputfile という名称の入力データファイルを指定しています。

(b)　index 文

c:¥hirdb¥pdload¥idxfile という名称のインデクス情報ファイルを指定しています。

(c)　lobdata 文

c:¥hirdb¥pdload¥lobdata という名称の LOB パラメタのファイルのディレクトリを指定しています。ここで指定したディレクトリ名と，入力データファイル中の LOB パラメタのファイル名を合わせて，絶対パス名となるようにします。

(3)　入力データファイル

この例では，入力データファイルは DAT 形式で記述しています。

また，LOB パラメタのファイルは別に用意し，入力データファイル中に LOB パラメタのファイルの名称を記述しています。

(4)　列構成情報ファイル

c:¥hirdb¥pdload¥column_file という名称の列構成情報ファイルを指定しています。

薬品 ID：

入力データファイルの 1 番目にあるデータの列名

取り扱い説明書：

入力データファイルの 2 番目にあるデータの列名

func=(SGMLTEXT,param=blob)：

抽象データ型列へ格納する値を生成する，コンストラクタ関数の情報を指定します。

SGMLTEXT：コンストラクタ関数の名称

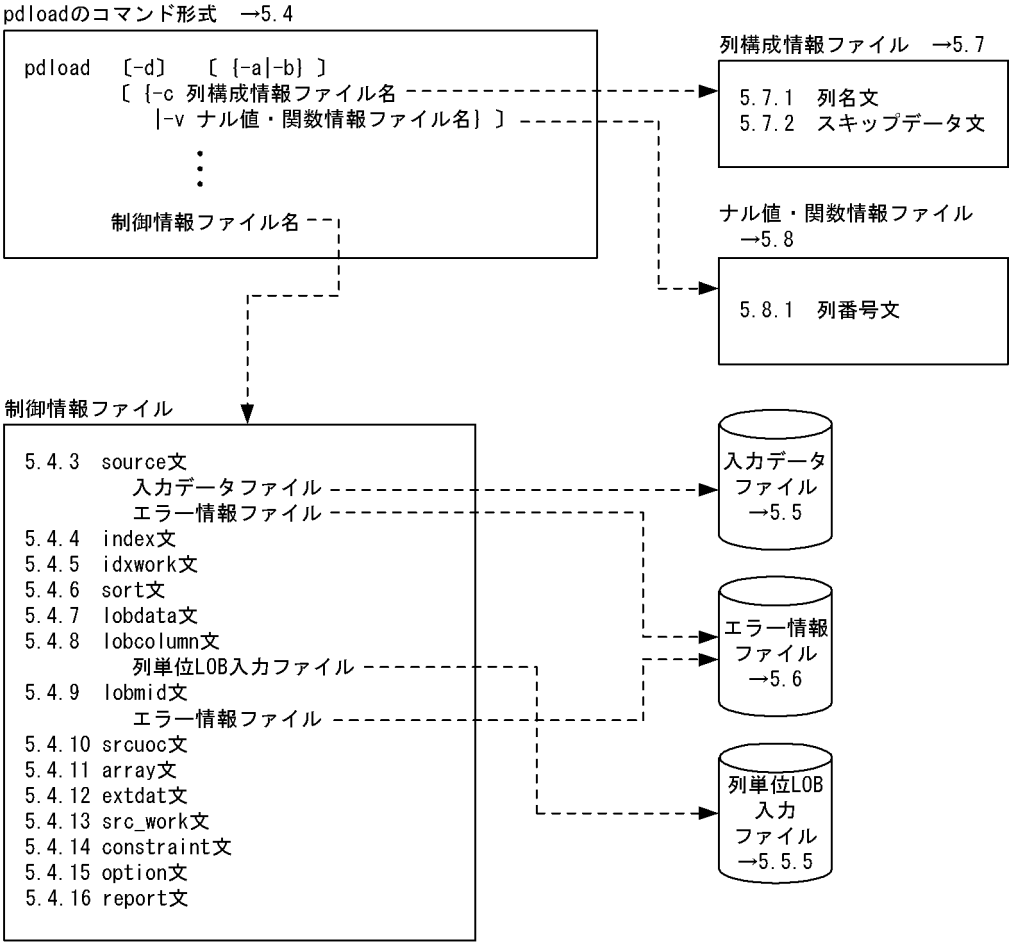
blob：コンストラクタ関数に渡す引数の型

5.3　目的ごとの参照先一覧

データロードの形態（どのような表にどのようなデータロードをするか）によって，指定するオプション，制御文，及び各ファイルが異なります。

指定するオプション，制御文，及び各ファイルの関連を次の図に示します。

図 5-9　オプション，制御文，及び各ファイルの関連



注　pdloadのコマンド形式，制御文，及びファイルの詳細については，→の節，項，又は制御文の前に記述されている項を参照してください。

5.3.1　必ず指定する項目

pdload を実行する場合には，次の項目を必ず指定するので，該当する参照先を参照してください。

項目	参照先	
	オプション	制御文，又はファイル
データロードする表の名称	〔認可識別子.〕表識別子	source 文
制御文を記述したファイル	制御情報ファイル名	

項目		参照先	
		オプション	制御文，又はファイル
入力データファイルの形式	DAT 形式	省略	入力データファイル
	バイナリ形式	-b	入力データファイル
	バイナリ形式（監査証跡ファイル）	-b	入力データファイル
		-W	
	pdrorg 出力バイナリ形式	-b	入力データファイル
		-W	
		-w	
		-U	
	固定長データ形式	-a	入力データファイル
		–	列名文

（凡例）–：該当しません。

5.3.2　データロードする表の属性によって指定する項目

次の項目が該当する場合，その参照先を参照してください。

項目		参照先	
		オプション	制御文，又はファイル
インデクスが定義されている場合		-i	index 文
		-x	idxwork 文
		-o	sort 文
LOB 列がある場合	LOB 入力ファイルを使用するとき	-k	lobdata 文
			lobmid 文
	列単位 LOB 入力ファイルを使用するとき	-k	lobcolumn 文
			lobmid 文
			列単位 LOB 入力ファイル
プラグインが提供する抽象データ型列（LOB 属性）がある場合		-k	lobdata 文
XML 型の列が定義されている場合		-K	option 文
		-G	
繰返し列がある場合		–	array 文

項目	参照先	
	オプション	制御文，又はファイル
		入力データファイル
可変長文字列があり，その列に対する入力データがない場合に，長さ 0 のデータを格納するとき※	-z	−
制約定義がある場合	−	constraint 文
分割格納条件を変更した場合	−	option 文
監査証跡表の場合	−	source 文
		srcuoc 文

（凡例）−：該当しません。

注※

表定義時に WITH DEFAULT を指定した場合は，1 バイト，又は 1 文字の空白を格納します。指定していない場合は，ナル値を格納します。

5.3.3　入力データファイルの形態によって指定する項目

(1)　DAT 形式の場合

次の項目が該当する場合，その参照先を参照してください。

項目	参照先	
	オプション	制御文
入力データと表の並びが対応していない場合	-c	列名文
入力データ中に表へ格納しないデータがある場合	-c	スキップデータ文
抽象データ型列に格納する値を生成するコンストラクタ関数の情報を記述する場合	-c	列名文
入力データ中のセパレータ文字がコンマ（,）以外の場合	-s	−
入力データ中の各国文字列型，及び混在文字列型のデータの，空白の文字コードを変換する場合	−	option 文
文字列型，及び BINARY 型の列に格納する入力データ長が列定義長より長い場合に，入力データエラーとして扱うとき	−	option 文
DAT 形式の入力データファイルが次のどちらかの条件を満たす場合	−	extdat 文

項目	参照先	
	オプション	制御文
• 格納するデータ中に，改行記号又は NULL 文字がある。 • 文字データを囲む文字が，引用符（"）以外である。		
INTEGER 及び SMALLINT の列に，小数点があるデータを格納する場合	−	extdat 文

（凡例）−：該当しません。

(2) バイナリ形式の場合

次の項目が該当する場合，その参照先を参照してください。

項目	参照先	
	オプション	制御文
入力データと表の並びが対応していない場合※	-c	列名文
入力データ中に表へ格納しないデータがある場合※	-c	スキップデータ文
入力データをナル値とみなす値に変換する場合，又は抽象データ型列に格納する値を生成するコンストラクタ関数の情報を記述する場合	-v	列番号文
入力データ中の各国文字列型，及び混在文字列型のデータの，空白の文字コードを変換する場合	−	option 文

（凡例）−：該当しません。

注※

pdrorg が出力した入力データファイルで，かつ非 FIX 表にデータロードする場合に指定できます。

(3) 固定長データ形式の場合

次の項目が該当する場合，その参照先を参照してください。

項目	参照先	
	オプション	制御文
入力データと表の並びが対応していない場合	-c	列名文
入力データ中に表へ格納しないデータがある場合	-c	スキップデータ文
入力データをナル値とみなす値に変換する場合，又は抽象データ型列に格納する値を生成するコンストラクタ関数の情報を記述する場合	-c	列名文
引用符（"）を入力データとして格納する場合	−	列名文

項目	参照先	
	オプション	制御文
入力データ中の各国文字列型，及び混在文字列型のデータの，空白の文字コードを変換する場合	−	option 文

（凡例）−：該当しません。

5.3.4　データロードの方法によって指定する項目

次の項目が該当する場合，その参照先を参照してください。

項目		参照先	
		オプション	制御文，又はファイル
データベース更新ログを取得する場合，又は取得しない場合		-l	−
大量のデータロードをする場合に，同期点を指定するとき		−	option 文
pdload の実行時間を監視する場合		−	option 文
DEFAULT 句を指定している表に対して，DEFAULT 句の既定値を格納する場合		−	option 文
UOC を使用する場合		−	srcuoc 文
入力データの不正を検知したときに，処理を途中で中断する場合※1		-e	エラー情報ファイル
		-r	option 文
入力データの不正を検知したとき，データロード処理を無効にする場合		−	option 文
自動採番機能を使用してデータロードする場合		−	option 文
UTF-8 の環境で定義した表に対して，UTF-16 の入力データファイルでデータロードする場合		−	option 文
表の空き領域比率を変えてデータロードする場合※2	未使用領域へデータを格納するとき	-y	−
	空き領域比率を指定して未使用領域へデータを格納するとき	−	option 文
データロード実行状態のメッセージの出力間隔を，10万行単位以外にする場合		-m	−
インフォメーションメッセージの出力を抑止する場合			
サーバ間通信の応答時間を監視する場合		-X	−

項目	参照先	
	オプション	制御文，又はファイル
グローバルバッファを使用しないで，ローカルバッファを使って一括出力のデータロードをする場合※3	-n	−
データロードを実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更する場合※4	-u	−
表単位の入力データファイルを，RD エリア単位にデータロードできる入力データファイルに分割する場合	−	src_work 文
FLOAT 型及び SMALLFLT 型の入力データ値を OS の制限値内に補正してデータロードする場合	-F	−

（凡例）−：該当しません。

注※1

例えば，DAT 形式の入力データファイルを使用して繰返し列がある表にデータロードする場合，入力データは VV 形式で作成しているのに，elmtype オペランドを省略すると FF 形式と扱われるために，すべての入力データがエラーとなり，無駄なエラー情報ファイルやエラーデータファイルが作成されてしまいます。このような場合に，エラーを検知した時点で処理を途中で中断できれば，無駄なファイルは作成されなくなり，また，無駄な動作もなくなります。

注※2

通常，追加データロードをする場合，表の空き領域比率に従って未使用ページにデータを格納します。このとき，格納できる未使用ページがなくなった場合，エラーとなり処理がロールバックされます。未使用ページがなくなった場合は，データベース再編成ユティリティでの表の再編成，又はデータベース構成変更ユティリティで RD エリアの追加，若しくは拡張をしてから，再度データロードしなければなりません。このような場合，これらの作業をしなくても，データロード時に表の空き領域比率を変えれば，データロードの処理を完了できることがあります。

注※3

一括出力ページ数を指定すると，指定したページ数単位で一括出力をするので，入出力回数を削減できます。

注※4

何も指定しない場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

5.4　コマンドの形式

5.4.1　形式

```
pdload 〔-d〕
       〔 {-a｜-b〔-W 〔-w {all｜data} 〔,rdaomit〕 〔,csetnck〕 〕 〕 ｜-U} 〕
       〔-i インデクス作成方法〕
       〔-l ログ取得方式〕
       〔-k LOB作成種別〕
       〔 {-c 列構成情報ファイル名
           ｜-v ナル値・関数情報ファイル名} 〕
       〔-n 〔一括出力用ローカルバッファ面数〔,div〕｜一括出力用ローカルバッファ面数,〔div〕,ランダムアクセス用ローカルバッファ面数｜,,ランダムアクセス用ローカルバッファ面数} 〕
       〔-u 認可識別子〕
       〔-x〕
       〔-s セパレータ文字〕
       〔-e〕
       〔-r 入力開始行〕
       〔-z〕
       〔-y〕
       〔-o〕
       〔-m〔 経過メッセージ出力間隔〕〔,インフォメーションメッセージ出力抑止レベル〕〕
       〔-X サーバ間通信の応答監視時間〕
       〔-K{ i｜f }〕
       〔-G{ xml｜esisb }〕
       〔-F〕
       〔認可識別子.〕表識別子
       制御情報ファイル名
```

注

背景色付き太字表示のオプションは，重要な項目，又は必ず指定する項目を表しています。また，〔認可識別子.〕表識別子，及び制御情報ファイル名は，必ずオプションの最後に指定してください。

5.4.2　オプション

(1)　-d

～《追加モード》

データベース作成ユティリティを作成モードで実行する場合に指定します。省略した場合は，追加モードが仮定されます。

作成モードとは，表の既存のデータをすべて削除してから，入力データを格納するモードのことをいいます。

追加モードとは，表の既存のデータを残して，入力データを追加するモードのことをいいます。

(a) 規則

1. 追加モードでデータロードした場合，追加データは既存のデータの最終格納ページ以降のページに格納されます。このため，クラスタキーを定義した表に対して追加モードでデータロードすると，追加データはクラスタキー順に格納されません。
2. 追加モードでデータロードをするときは，必ず未使用ページにデータを格納します。未使用ページを確保できなかった場合にはエラーとなります。既存のデータが格納されている使用中ページに空きがあっても利用しません。ただし，-y オプション指定時には，既存データが格納されている使用中ページを利用します。
3. RD エリア単位のデータロードを作成モードで実行した場合，削除される既存の表データは，指定した RD エリア部分だけです。
4. データ削除中は，option 文に nowait=yes を指定していても NOWAIT 検索はできません。
5. 同期点指定のデータロードで異常終了した後の再実行の場合，作成モードを指定しても無視されます。
6. 更新凍結状態の RD エリアに格納されている LOB 列，又は LOB 属性の抽象データ型列がある表に対してデータロードする場合，作成モードを指定するとデータを削除できないためエラーとなります。なお，LOB 列構成基表にだけデータロードする場合でも，作成モードを指定するとエラーになります。
7. 改竄防止表の場合，作成モードのデータロードはできません。

(2) {-a | -b〔-W 〔-w {all | data}〔,rdaomit〕〔,csetnck〕〕 | -U}

～《DAT 形式》

入力データファイルの形式を指定します。入力データファイルについては，「入力データファイル」を参照してください。

省略：DAT 形式

-a：固定長データ形式

-b：バイナリ形式

監査証跡ファイルはバイナリ形式なので，監査証跡表にデータロードする場合は-b オプションを指定する必要があります。

-Wオプションと同時に指定することで，pdload 用アンロードファイルとして扱います。

-W：pdload 用アンロードファイル

このオプションを指定する場合は，同時に-b オプションも指定してください。アンロードした表とデータロードする表の間で，列データ型などの表定義が異なる場合はデータロードできません。ただし，非 FIX 表に限り，列構成情報ファイルを指定することで，列定義順序と列数を変更したデータロードができます。なお，UAP や UOC で入力データファイルの内容を変更する場合，行長，列データオフセット，及び行データの間は，間隔を空けないで詰めた形式で編集する必要があります。

監査証跡表にデータロードする場合はこのオプションを指定する必要があります。

-w：表移行用アンロードファイル

このオプションを指定する場合は，同時に-b -W オプションも指定してください。

{all ｜ data}

all：表，インデクス定義，及び表データを格納します。

data：表データだけを格納します。

rdaomit

このオプションは，移行元と移行先の格納 RD エリア名が一致していない場合に指定します。ただし，次の条件を満たしている表に限ります。

- インポートする表に BLOB 列がない
- 非分割表である

csetnck

表移行用アンロードファイル中の文字コードと，インポート先の HiRDB で使用する文字コードが同じかどうかのチェックを省略します。

-U：pdrorg 用アンロードファイル

このオプションを指定してデータロードする場合，アンロードした表の列定義とデータロードする表の列定義を合わせる必要があります。SUPPRESS 指定の列を持つ表をアンロードするときに-S オプションを指定（空白データを圧縮）して作成したアンロードデータファイル，及びスキーマ単位のアンロードデータファイルは使用できません。また，使用できるアンロードデータファイルの媒体はファイルだけです（テープや HiRDB ファイルは不可）。アンロードしたデータが複数ファイルに出力された場合は，pdrorg 実行時にデータ出力があったアンロードデータファイル名だけを，unload 文に指定した順に pdload の source 文に記述してください。指定順を誤ると，正しくデータロードできないことがあります。

なお，データロード対象表が次のどれかの条件に該当する場合にこのオプションを指定すると，制御文エラーになります。

- 抽象データ型の列を持つ表
- BINARY 型の列を持つ表
- 繰返し列を持つ表
- FIX 表

これらの表にデータロードするときは，pdload 用アンロードファイルを使用してください。

(3) -i インデクス作成方法

～《c》

インデクスの作成方法を指定します。インデクスの作成方法には次の四つがあります。

c：

インデクス一括作成モードを示します。このモードを指定した場合，表の作成処理に引き続き，インデクスを一括作成します。

行データの格納中はインデクスを作成しないで，インデクス作成情報をインデクス情報ファイルに出力します。行データの格納処理が完了した後，引き続きインデクスを作成します。

＜適用基準＞

大量のデータロードをする場合に，高速にインデクスが作成できます。

＜注意事項＞

1. 追加モードのデータロードする場合，インデクス情報ファイルには，既存データと追加データを合わせたすべてのインデクス情報が出力されるため，全件再作成となります。なお，プラグインインデクスの追加データロードの場合は，追加データのインデクス情報だけが出力されます。そのため，プラグインインデクスは，追加した件数だけ作成します。
2. インデクス一括作成モードで同期点指定のデータロードをすると，次の理由によって処理性能が悪くなります。

   ・B-tree インデクスの場合，すべてのデータを格納した後，再度データページを検索してインデクス情報ファイルの作成をしてしまう。

   ・プラグインインデクスの場合，トランザクションが決着するごとにインデクス作成処理を開始するため，データロード処理とインデクス作成処理を交互に実行することになる。

n：

インデクス情報出力モードを示します。このモードを指定した場合，インデクス情報だけをインデクス情報ファイルに出力します。

行データの格納中はインデクスを作成しないで，インデクス作成情報をインデクス情報ファイルに出力します。

＜適用基準＞

出力されたインデクス情報ファイルを使用して，データベース再編成ユティリティでインデクスの一括作成を複数同時に実行することで，高速にインデクスを作成できます。特に，HiRDB/パラレルサーバで複数のサーバに分割格納されている表の場合に効果があります。

＜注意事項＞

1. RD エリア単位のデータロードで，サーバ内で複数の RD エリアに対して表を分割格納している場合に，分割表の一部を作成したとき，サーバ内横分割をしていない非分割キーインデクスはインデクス情報出力モードとして処理されます。
2. プラグインインデクスが定義されている表の場合，インデクス情報出力モードは指定できません。
3. インデクス情報出力モードを指定した場合，データロード完了後にデータベース再編成ユティリティでインデクスの一括作成をする必要があります。

s：

インデクス更新モードを示します。このモードを指定した場合，行を格納するたびにインデクスを更新します。

＜適用基準＞

既存データが大量にある表に対して，少量の追加モードのデータロードをする場合や，ユニークキーインデクス又は主キーインデクスが定義されていて，かつ格納するデータのキー値が重複している可能性がある場合に指定してください。

x：

インデクス情報出力抑止モードを示します。このモードを指定した場合，インデクスは更新しません。また，インデクス情報をインデクス情報ファイルに出力しません。

＜適用基準＞

データロードを数回に分けて実行する場合に指定します。

例えば，MT が n 巻あるため，n 回データロードを実行する場合にこのモードを指定します。

＜注意事項＞

1. -i x でデータロードする場合は，最後の実行で-i c（インデクス一括作成モード）を指定するか，又は-i n（インデクス情報出力モード）を指定してください。-i n を指定した場合は，後でデータベース再編成ユティリティでインデクスの一括作成（-k ixmk）をする必要があります。

   また，すべて-i x でデータロードした場合は，後でデータベース再編成ユティリティでインデクスの再作成（-k ixrc）をする必要があります。

2. プラグインインデクスが定義されている表の場合，インデクス情報出力抑止モードは指定できません。

3. -i x でデータロードするとインデクスを更新しないため，インデクスは未作成状態となります。

(a) インデクス作成上の注意事項

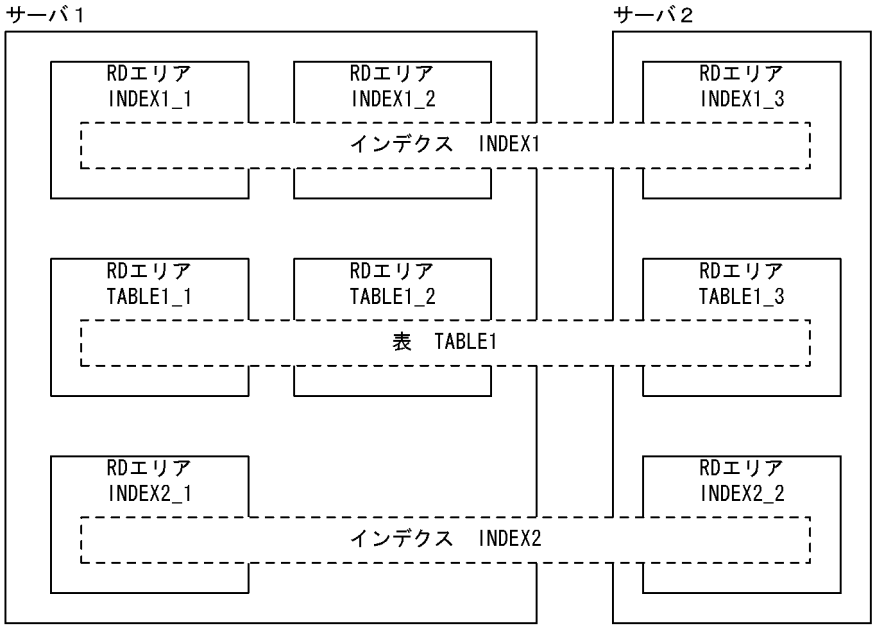
1. インデクス更新モード（-i s）以外で pdload を実行すると，行データをすべて表へ格納した後に一括してインデクスを作成することになります。このとき，インデクス作成時にキー重複エラーやキー重複の上限数を超えるエラーが発生すると，表とインデクスが不整合な状態となります。このため，次に示す場合は，ログ取得モードに関係なく，ユティリティ実行前にバックアップを取得してください。
   - ユニークキーインデクス又は主キーインデクスを定義している表の場合
   - 繰返し列にインデクスを定義している表の場合
2. 横分割インデクスと非横分割インデクスでのインデクス作成方法を次の図に示します。

   1 サーバ内の複数 RD エリアに表を分割格納している場合，表格納用 RD エリアと同じ数のインデクス格納用 RD エリアに格納する横分割インデクスと，表格納用 RD エリアの数とは関係なく一つのインデクス格納用 RD エリアに格納する非横分割インデクスがあります。

   横分割インデクスは，サーバ内で分割されている表に対して，同じように分割格納します（図中の RD エリア INDEX1_1，INDEX1_2）。

   非横分割インデクスは，表格納用 RD エリアの数に関係なく，サーバ内の 1RD エリアに格納します（図中の RD エリア INDEX1_3，INDEX2_1，INDEX2_2）。

図 5–10　分割キーインデクスと非分割キーインデクスでのインデクス作成方法



[説明]

データロード完了後は，-i オプションの指定によってインデクスが作成されます。

例えば，RD エリア単位に TABLE1_1 にデータロードをした場合，横分割インデクス INDEX1_1 が作成されます（INDEX1_2 は影響を受けません）。しかし，非横分割インデクス INDEX2_1 は，TABLE1_1 及び TABLE1_2 の両方の情報がないと作成できないため，インデクスは作成されません（TABLE1_1 のインデクス情報ファイルだけ作成されます）。

非横分割インデクスを作成するためには，TABLE1_2 に対してもデータロードをして，TABLE1_2 のインデクス情報ファイルを作成します。そして，作成済みの TABLE1_1 のインデクス情報ファイルと TABLE1_2 のインデクス情報ファイルを使用して，pdrorg のインデクスの一括作成をします。

3. 表データの格納が終了してから，インデクスの一括作成が終了するまでの間は，該当するインデクスは未完状態となり，使用できません。

4. -i c 又は-i n を指定して，入力データファイルにデータがなかった場合，インデクス情報ファイルには既存データのインデクス情報だけ出力されます。これを pdrorg の入力とすることで，インデクスの再作成ができます。この場合，pdload は必ず追加モード（-d オプション省略）で実行してください。作成モード（-d オプション指定）で実行した場合，pdload は先に表データを削除してしまうので，インデクス情報もなくなってしまいます。

5. -i n でデータロードをした後に，pdrorg のインデクスの一括作成（-k ixmk）を実行していない状態で，ALTER TABLE で RD エリアを追加すると，-i n で出力されたインデクス情報ファイルは使用できなくなります。ALTER TABLE で RD エリアを追加する場合は，先に pdrorg のインデクスの一括作成を終了させてください。なお，先に ALTER TABLE で RD エリアを追加してしまった場合は，pdrorg のインデクスの再作成（-k ixrc）をする必要があります。

6. -i c，又は-i n を指定すると，「インデクス数×表格納用 RD エリア数」のインデクス情報ファイルを作成します。このファイルは，データロード時に同時にオープンするため，ファイル数が多いと 1 プロセ

スで使用できるファイル数の上限を超えることがあります。上限を超えると，性能が劣化するおそれがあります。そのため，RD エリアごとにデータロードすることを推奨します。

なお，1 プロセスで使用できるファイルディスクリプタ数の上限値は，2048 です。

7. インデクス情報出力モード（-i n）を指定した場合，インデクスの一括作成（-k ixmk）が完了するまでは ALTER INDEX でインデクスの名称を変更しないでください。インデクスの名称を変更すると，インデクスの一括作成ができなくなります。なお，インデクスの一括作成が完了する前にインデクスの名称を変更してしまった場合は，ALTER INDEX で元のインデクス名称に戻してからインデクスの一括作成をし，その後でインデクスの名称を変更してください。
8. インデクス情報出力モード（-i n）を指定した場合，インデクスの一括作成（-k ixmk）が完了するまでは，インデクスを格納した RD エリアの名称を変更しないでください。インデクスを格納した RD エリアの名称を変更すると，インデクスの一括作成ができなくなります。なお，インデクスの一括作成が完了する前にインデクスを格納した RD エリアの名称を変更してしまった場合は，pdmod で元の RD エリア名称に戻してからインデクスの一括作成をし，その後で RD エリアの名称を変更してください。
9. インデクス一括出力モード（-i c）又はインデクス情報出力モード（-i n）を指定して，分割数の多い分割表のインデクスを作成した場合は，RD エリアごとにデータロードを実行してください。

(b)　プラグインインデクス作成上の注意事項

1. プラグインインデクスが定義されている場合は，インデクス情報出力モード，及びインデクス情報出力抑止モードは指定できません。したがって，インデクス一括作成モード，又はインデクス更新モードを指定してください。
2. B-tree 構造のインデクスの場合はインデクス一括作成モードでデータロードを実行するとすべてのインデクスを作成しますが，プラグインインデクスの場合は追加分のデータしかインデクスを作成しません。このため，0 件のデータロードを実行した場合，B-tree 構造のインデクスはインデクスを再作成しますが，プラグインインデクスは再作成しません。しかし，プラグインインデクスの場合も，インデクス格納 RD エリアの再初期化後に 0 件のデータロードを実行すると，インデクスを再作成します。これは，インデクス作成時の異常終了やディスク障害などでインデクスが破壊された状態になった場合に，インデクスを復旧するために利用します。

(c)　-i c 又は-i n 指定時の注意事項

-i c 又は-i n 指定時に index 文，及び idxwork 文を省略すると，既定の出力先ディレクトリにインデクス情報ファイルを出力します。pdload が異常終了すると，このインデクス情報ファイルが不完全な状態で残ります。このため，pdload の異常終了を何度も繰り返すと不完全なインデクス情報ファイルが増加し，ディスク容量など OS の資源を圧迫することになります。したがって，既定の出力先ディレクトリに出力した必要のないインデクス情報ファイルについては削除するようにしてください。

index 文を省略して idxwork 文だけを指定した場合に，idxwork 文に指定したディレクトリ下に出力するインデクス情報ファイルについても同様です。

既定の出力先ディレクトリについては「規則及び注意事項」の「制御文で指定しない場合のファイル出力先ディレクトリ」を参照してください。

また，インデクス情報ファイルは，次の名称規則で出力します。

```
INDEX-インデクス名※-インデクス格納用RDエリア名※-一意となる文字列
```

注※

インデクス名又はインデクス格納 RD エリア名に「¥」が含まれている場合は，「¥」を「(」に置き換えます。

(4)　-l ログ取得方式

～《p》

pdload 実行時のデータベースの更新ログの取得方法を指定します。

なお，pdload 実行中に異常終了した場合，更新ログを取得していても，データベースは pdload 実行前の状態には戻りません。pdload が異常終了した場合の対処方法については，「エラー時のデータベースの状態とその回復方法」を参照してください。

a：

ログ取得モードを示します。ロールバック及びロールフォワードに必要なデータベースの更新ログを取得します。

＜適用基準＞

このモードは，少量のデータをデータロードする場合に適しています。

ログ取得モードで実行した場合，pdload の実行前及び実行後にバックアップを取得する必要はありませんが，処理性能はほかのモードに比べて劣ります。

p：

更新前ログ取得モードを示します。ロールバックに必要なデータベースの更新ログは取得しますが，ロールフォワードに必要なデータベースの更新ログは取得しません。

＜適用基準＞

このモードは，大量のデータをデータロードする場合に適しています。

更新前ログ取得モードで実行した場合，ログ取得モードに比べると実行時間は短縮できます。ただし，以後の媒体障害などに備えて，pdload の実行後に，表及びインデクスを格納した RD エリアのバックアップを取得する必要があります。

pdload 実行中に障害が発生した場合，pdload が障害発生直前に同期を取った時点まで回復します。そのため，RD エリアがログレス閉塞になることは防げますが，処理対象表は pdload 実行前の状態には回復されません。

＜注意事項＞

プラグインが出力するログについて，プラグインが提供する抽象データ型を定義した表にデータロードをする場合，p（更新前ログ取得モード）の指定が有効かどうかは，各種プラグインの実装によって決まります。p の指定が有効にならないプラグインの場合は，a（ログ取得モード）が仮定されます。

n：

ログレスモードを示します。データベースの更新ログを取得しません。

＜適用基準＞

このモードは，処理対象となる RD エリアにデータロード対象表とその表のインデクスだけ定義している場合に，大量のデータをデータロードするときに適しています。

ログレスモードで実行した場合，ほかのモードに比べると実行時間は短縮できます。ただし，ユティリティ実行中の障害でバックアップから回復する場合や，以後の媒体障害などの発生によって，データベースのバックアップ及びログから回復する場合に備えて，ユティリティの実行前及び実行後に，表及びインデクスを格納した RD エリアのバックアップを取得する必要があります。

障害発生時には，バックアップ取得時点までしか回復できません。

＜注意事項＞

1. pdload をログレスモードで実行していた場合にエラーが発生したときは，エラーによって閉塞した RD エリアはバックアップから回復するか，又は再初期化をする必要があります。

(a) 注意事項

1. n の場合，障害発生時に以前取得したバックアップとログ，又は入力データ（データロードする表にデータが格納されていない場合で，かつ該当する RD エリアにその表だけ格納する場合）から回復できるときは，pdload 実行前のバックアップ取得は不要です。
2. p，及び n の場合の運用方法（データベースの更新ログを取得しないときの運用）については，マニュアル「HiRDB システム運用ガイド」を参照してください。
3. -l オプションの指定に関係なく，pdload が生成するトランザクション（T）によってトランザクションログは必ず取得されます。1 サーバ当たり，次のトランザクションログを生成するため，pdload 実行時のログ量の計算式は次のようになります。

```
ログ量＝（1328＋176×3）×T＋A　（単位：バイト）

T＝（x×2）＋（y×2）＋（z×2）
  x：表の数※
  y：LOB列（LOB属性）格納用RDエリアの数
  z：インデクス数×インデクス格納用RDエリアの数（-i sの場合は不要）
```

注※

同期点指定のデータロードの場合は同期点の数，それ以外の場合は 1 となります。

A ＝データベース操作に応じて出力されるシステムログ量（データベースの更新ログ量）。この値は-l オプションの指定値で変わります。システムログ量の求め方については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

また，p 又は n を指定した場合，排他制御によって ENQ ログを取得します。1 サーバ当たり次の ENQ ログを取得します。

```
ENQログ＝（p＋q＋r）×T

  p：表格納用RDエリアの数
```

```
q：LOB列（LOB属性）格納用RDエリアの数
r：インデクス格納用RDエリアの数
```

したがって，「システムログファイルのレコード長×ENQ ログ」の分だけシステムログが出力されます。

(5) -k LOB 作成種別

～《f 又は d》

LOB データを LOB 列に格納する場合，及び抽象データ型列に格納する値を生成するコンストラクタ関数の引数が LOB パラメタの場合に，そのデータの入力方法を指定します。このオプションを省略した場合は f が仮定されますが，option 制御文に job オペランドの指定がなく，-W オプションを指定している場合は d が省略時仮定値となります。

f：

LOB データごとにファイルを用意する場合に指定します。これらのファイルを LOB 入力ファイルと呼びます。

c：

一つの LOB 列にある複数の LOB データを，一つのファイルに用意する場合に指定します。このファイルを列単位 LOB 入力ファイルと呼びます。

列単位 LOB 入力ファイルは，何らかのプログラムを使用して他データベースからデータを移行するときなどに作成したりします。

抽象データ型の列に格納するデータを生成する関数の入力パラメタとして指定する LOB データの場合は，このオプションは指定できません。したがって，LOB を入力パラメタとする抽象データ型の列がある表の場合は，抽象データ型の列でない LOB 列があるときに，その列に対しても列単位 LOB ファイルは使用できません。

v：

LOB 列への格納データは格納する LOB データごとに LOB 入力ファイルを用意し，抽象データ型列に格納する値を生成するコンストラクタ関数の入力パラメタとなるデータ（LOB パラメタ）は入力データファイルに格納する場合に指定します。

この指定は，入力データファイルがバイナリ形式の場合だけできます。

d：

LOB データ，及び抽象データ型列の LOB パラメタデータを，入力データファイルに直接記述する場合に指定します。

この指定は，入力データファイルがバイナリ形式の場合だけ指定できます。

-k d を指定した場合，UOC を利用したデータロードはできません。

(a) 適用基準

-k オプションは，次に示す LOB 列，抽象データ型列の LOB パラメタの有無，及び入力データファイルの形式によって指定値を決めてください。

LOB 列	入力データファイルの形式	抽象データ型列		
		なし	あり	
			LOB パラメタなし	LOB パラメタあり
なし	DAT 形式	−	−	-k f
	バイナリ形式			-k v 又は-k d
	固定長データ形式			-k f
	pdrorg 出力バイナリ形式	−	−	-k d
あり	DAT 形式	-k f 又は-k c	-k f 又は-k c	-k f
	バイナリ形式	-k f，-k c，又は-k d	-k f，-k c，又は-k d	-k v 又は-k d
	固定長データ形式	-k f 又は-k c	-k f 又は-k c	-k f
	pdrorg 出力バイナリ形式	-k d	-k d	-k d

（凡例）

−：-k オプションは指定する必要がありません。

(b) 規則

1. LOB 列構成基表と LOB 列を別々に作成する場合，LOB 列作成時は LOB 列構成基表作成時と同じオプションを指定してください。
2. -k オプションの指定値によって，入力データファイル，及び LOB 入力ファイルの内容は次のようになります。

-k オプション	入力データファイル		LOB 入力ファイル	
	LOB 列格納データ	抽象データ型 LOB パラメタ	LOB 列格納データ	抽象データ型 LOB パラメタ
f	LOB 入力ファイルのファイル名を記述	LOB 入力ファイルのファイル名を記述	格納データ数分のファイルを用意	格納データ数分のファイルを用意
c	ダミーデータを記述	−	列数分のファイルを用意	−
v	LOB 入力ファイルのファイル名を記述	LOB パラメタの内容を記述	格納データ数分のファイルを用意	不要
d	LOB データの内容を記述	LOB パラメタの内容を記述	不要	不要

（凡例）−：該当しません。

3. 同期点指定のデータロードの場合，-k オプションには d を指定してください。f，c，及び v は指定できません。

(6)　-c 列構成情報ファイル名

列構成情報ファイルの名称を指定します。

列構成情報ファイルについては，「列構成情報ファイル」を参照してください。

(a)　規則

1. 列構成情報ファイルは，入力データファイルが固定長データ形式の場合は必ず指定してください。
   DAT 形式，又は pdrorg 出力バイナリ形式の場合は任意に指定できます。ただし，pdrorg 出力バイナリ形式の場合は，非 FIX 表に対してデータロードするときに指定できます。FIX 表に対してデータロードした場合にはエラーとなります。FIX 表の場合に列構成情報ファイルを指定して列構成を変更したいときは，該当するファイルを固定長データ形式の入力データファイルとして扱うと，同様の処理ができます。
2. DAT 形式の入力データファイルの場合，次の条件をすべて満たすときは，列構成情報ファイルの指定は不要です。
   - 入力データの列の並び順と表の列定義の順序が同じとき
   - 入力データの列数と表の列数が同じとき
   - 抽象データ型に格納するデータを生成するコンストラクタ関数の名称が，列データ型の名称と同じで，かつ該当する関数が一つだけのとき
   - 繰返し列に格納する入力データの要素数が，表定義時に指定した最大要素数と同じとき
3. 列構成情報ファイルは，データベース作成ユティリティを実行する（pdload コマンドを入力する）ホストに用意してください。
4. 列構成情報ファイルは，ナル値・関数情報ファイルと同時に指定できません。

(7)　-v ナル値・関数情報ファイル名

ナル値・関数情報ファイルの名称を指定します。

ナル値・関数情報ファイルは，入力データをナル値とみなす値に変更するとき，又は抽象データ型列に格納する値を生成するコンストラクタ関数の情報を記述するときに使用します。

ナル値・関数情報ファイルについては，「ナル値・関数情報ファイル」を参照してください。

(a)　規則

1. ナル値・関数情報ファイルは，入力データファイルがバイナリ形式の場合に指定できます。DAT 形式，及び固定長データ形式の場合には指定できません。
2. ナル値・関数情報ファイルは，列構成情報ファイルと同時に指定できません。

(8) -n {一括出力用ローカルバッファ面数〔,div〕|一括出力用ローカルバッファ面数,〔div〕,ランダムアクセス用ローカルバッファ面数|,,ランダムアクセス用ローカルバッファ面数}

表へデータロードする場合に，データベースに対して書き込みを行うときに使用するローカルバッファの面数を指定します。このオプションを指定した際に，一括出力用ローカルバッファ面数，又はランダムアクセス用ローカルバッファ面数を指定していない場合は，該当ローカルバッファは使用しません。なお，このオプション自体を省略した場合はローカルバッファを使用しません。この場合，グローバルバッファを使用して 1 ページ単位に出力します。

一括出力用ローカルバッファ面数　～<符号なし整数>((2～4096))

一括出力用ローカルバッファの面数を指定します。一括出力用ローカルバッファは，データページに対して使用します。

一括出力用ローカルバッファ面数には，16～32 面を指定することをお勧めします。184 キロバイト÷ページ長を上限に指定値を決めてください。

div

次の条件をすべて満たす場合に div を指定してください。次の条件をすべて満たす場合に div を指定しないと，入出力回数が増加し，性能が劣化することがあります。

- 横分割表に対して，表単位にデータロードをする
- ハッシュ関数（HASH0～HASH6，HASHZ）を使用した横分割表に対してデータロードをする，又はキーレンジ分割の表に格納する入力データのキー値の順序がランダムになっている
- 表を格納する RD エリアが 1 サーバ内に複数ある

なお，div を指定すると，サーバ内表分割数分のバッファを確保するため，メモリ消費量が増えます。

ランダムアクセス用ローカルバッファ面数　～<符号なし整数>((4～125000))

ランダムアクセス用ローカルバッファの面数を指定します。ランダムアクセス用ローカルバッファは，インデクスページに対して使用します。このオペランドは，インデクス更新モードでインデクスを作成する場合に指定します。インデクス更新モードでは，行を格納するたびに B-tree 構造のインデクスページを参照しますが，ランダムアクセス用ローカルバッファを使用すれば，これらのインデクスページを該当するバッファに読み込むため，グローバルバッファ競合によるオンライン処理の性能劣化を軽減できます。

推奨する-n オプションの指定を次の表に示します。

表 5–5　推奨する-n オプションの指定（pdload）

表種別	列定義	サーバ内での表の分割	
		あり	なし
FIX 表	－	-n x，div	-n x

表種別	列定義	サーバ内での表の分割	
		あり	なし
非 FIX 表	列長が 256 バイトを超える可変長文字列，又は BINARY 列を定義している場合	-n の指定を省略	
	上記以外	-n x，div	-n x

（凡例）

x：一括出力用ローカルバッファ面数

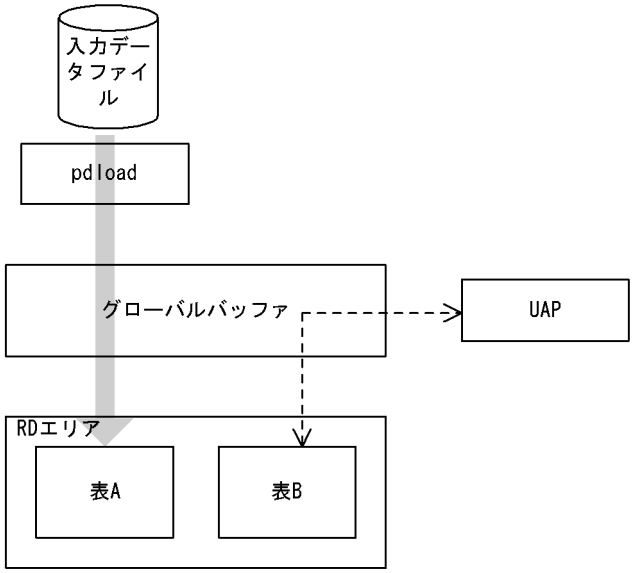
−：該当なし

(a)　pdload が使用するバッファについて

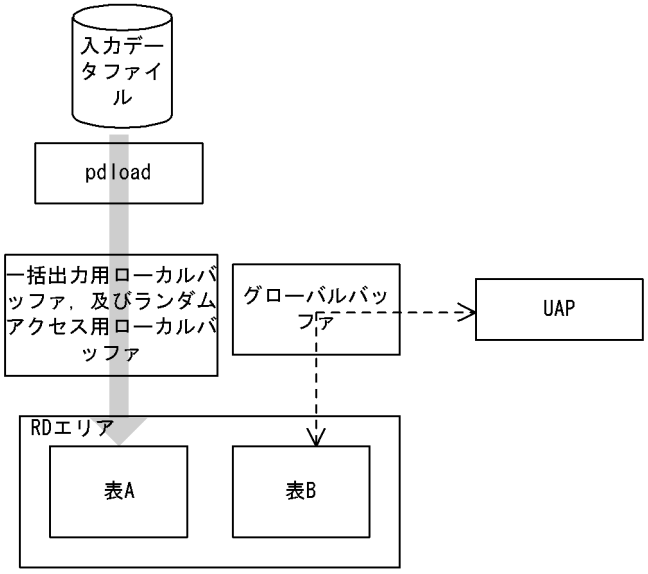
1. -n オプションを省略した場合，グローバルバッファを使用します。この場合，データロード時に大量のグローバルバッファを使用するため，そのグローバルバッファを使用する UAP のトランザクション性能が劣化します。-n オプションを指定すると，このようなバッファ競合はなくなります。pdload とバッファの関係を次の図に示します。

図 5–11　pdload とバッファの関係

●グローバルバッファだけを使用する場合



●ローカルバッファとグローバルバッファを使用する場合



〔説明〕

グローバルバッファだけを使用する場合（-n オプションを省略した場合），pdload と UAP とでバッファの競合が発生します。

ローカルバッファとグローバルバッファを使用する場合（-n オプションを指定した場合），pdload と UAP とのバッファの競合は発生しません。ただし，LOB 列のデータロードの場合は，-n オプションを指定してもグローバルバッファを使用します。

2. -n オプションに一括出力用ローカルバッファ面数だけを指定した場合，一つの RD エリアに対して一つの一括出力用ローカルバッファを使用します。RD エリアが n 個ある場合でも，一つの一括出力用ローカルバッファを使用するため，バッファ競合が発生します。バッファ競合が発生すると，逆に入出力回数が増加して性能劣化を引き起こします。このような場合には div を指定してください。div を指定す

ると，RD エリア数分の一括出力用ローカルバッファを確保するため（一つの RD エリアに対して一つの一括出力用ローカルバッファを確保する），バッファの競合はなくなります。

3. 一括出力用ローカルバッファを指定しても，条件によってはグローバルバッファを使用することがあります。各条件と使用されるバッファとの関係を次の表に示します。

表 5–6　各条件と使用されるバッファとの関係（pdload）

条件				ランダムアクセス用ローカルバッファの指定				
				なし		あり		
				グローバルバッファ	一括出力用ローカルバッファ	グローバルバッファ	一括出力用ローカルバッファ	ランダムアクセス用ローカルバッファ
データページ	ユーザ用 RD エリア	FIX 表	クラスタリング指定の表で，行が所定のセグメントに格納できなくなった場合	○	−	○	−	−
			上記以外	−	○	−	○	−
		非 FIX 表	列長が 256 バイトを超える可変長文字列，又は BINARY 列を定義している場合	○	−	○	−	−
			抽象データ型列又は繰返し列を定義していて，かつ 1 行のデータが 1 ページに入らない場合	○	−	○	−	−
			クラスタリング指定の表で，行が所定のセグメントに格納できなくなった場合	○	−	○	−	−
			上記以外	−	○	−	○	−
		表からキー値検索する場合（-i c 又は n 指定時）		−	○	−	○	−
	ユーザ LOB 用 RD エリア	LOB 列格納用 RD エリア		○	−	○	−	−
		LOB 属性格納用 RD エリア		○	−	○	−	−
インデクスページ	インデクスの同時作成をする場合（-i s 指定時）			○	−	−	−	○
	インデクスの一括作成をする場合（-i c 指定時）			○	−	−	−	○
ディレクトリページ				○	−	○	−	−

（凡例）

○：使用します。

－：使用しません。

(b)　規則

1. リバランス表の場合，グローバルバッファが十分に割り当てられていないと，極端に性能が劣化することがあります。したがって，定義長が 256 バイトを超える列がないときは一括出力ページ数を指定するようにしてください。

   定義長が 256 バイト以上の列があるなどの理由で，ローカルバッファ指定が有効にならない場合は，次の面数以上のグローバルバッファを割り当ててください。

   1RD エリアに必要なバッファ面数

   ＝ 1024÷(表格納用 RD エリア数)×2 ＋ 3

2. FIX ハッシュ分割のリバランス表の場合，ハッシュグループごとに指定したページ数のバッファを確保するため，横分割していない表やほかの条件で分割している表よりも，メモリを多量に消費します。

(9)　-u 認可識別子

データベース作成ユティリティを実行するユーザの認可識別子を指定します。

省略時仮定値については，「省略時仮定値」を参照してください。

このオプションを指定した場合，パスワード入力要求の応答付きメッセージが表示されます。パスワードが不要な場合は，応答付きメッセージに対して空送信してください。そこで応答した認可識別子/パスワードで HiRDB システムに接続し，その認可識別子について表へのアクセス権限があるかどうかがチェックされます。

(a)　適用基準

環境変数 PDUSER に設定されている認可識別子ではない，別の認可識別子で実行したい場合に指定します。

(b)　省略時仮定値

省略した場合，次のように認可識別子/パスワードが仮定されます。

1. データベース作成ユティリティ実行時の環境変数 PDUSER の設定値が仮定されます。パスワードの応答ができない環境で実行するときは，PDUSER を必ず設定してください。環境変数 PDUSER の指定例を次に示します。

   （例）

   パスワードありの場合　PDUSER=¥””認可識別子”¥”/¥””パスワード”¥”

   パスワードなしの場合　PDUSER=¥””認可識別子”¥”

2. 環境変数 PDUSER が設定されていない場合，ログインウィンドウのユーザ名が仮定されます。パスワードについては，入力要求の応答付きメッセージが表示されるので，そこで応答してください。パスワードが不要な場合は，応答付きメッセージに対して空送信してください。

(c) 規則

1. このオプションは，パスワードが応答できない環境では指定しないでください。
2. 認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(10) -x

表にクラスタキーが定義してある場合，入力データがクラスタキーの昇順又は降順になっているかどうかをチェックしないで，入力順にデータロードする場合に指定します。

このオプションを省略した場合は，クラスタキー順のチェックをして，クラスタキー順でない入力データはエラーとなります。

(a) 適用基準

入力データの順序がクラスタキー順であることが分かっている場合，このオプションを指定することでクラスタキーチェックのためのオーバヘッドを削減できます。

(b) 注意事項

1. UNIQUE 指定のクラスタキーが定義されている表の場合，重複キーのチェックができなくなるため，重複キーのデータを格納しようとするとエラー（ロールバック）となります。
2. クラスタキーが定義されている表に対して，クラスタキー順でない入力データを格納できますが，クラスタリングの効果はなくなります。

(11) -s セパレータ文字

～<文字列>

入力データファイルが DAT 形式の場合，各データ間のセパレータ文字を変更する場合に指定します。省略した場合，コンマ（,）が仮定されます。

バイナリ形式，及び固定長データ形式の場合は，指定しても無視されます。

(a) 適用基準

入力データ中の各データ間のセパレータ文字が，コンマ以外の場合に指定します。

(b) 規則

1. 入力データ中にない文字コードであれば，セパレータ文字としてタブやパイプ記号（|）を指定できます。ただし，これらの特殊な文字はセパレータ文字を引用符（"）で囲むなどして指定する必要があります。
2. 次に示す文字は，セパレータ文字として指定できません。
   - 英大文字（A～Z），英小文字（a～z）

- 数字（0～9）

さらに，次に示す文字もユティリティが特別な意味で使用する予約文字なので，セパレータ文字として指定できません。

- アスタリスク（＊），引用符（"），下線（_）

3. 次に示す文字は，入力データのコードと重なる可能性があるのでセパレータ文字に適していません。なお，セパレータ文字は 1 バイトのため，2 バイトコードは指定できません。
   - 数値データ入力時の符号（＋，−）
   - 日本語入力時に文字コードが重なる文字（｜，￥，[,]，(,)，{,}，￣）
   - 日付データ入力時のハイフン（−）
   - 時刻データ入力時のコロン（：）
   - 日間隔，時間隔データ入力時のピリオド（.）

(12) -e

入力データの不正を検出した場合に，処理を中断するときに指定します。

処理を中断した後，それまでの格納処理を無効にしたい（ロールバックしたい）場合は，option 文の dataerr オペランドを指定します。

-e オプション，及び option 文の dataerr オペランドの両方を省略した場合，入力データの不正を検出しても処理を続行して，正常なデータだけを DB に格納します。

-e オプションを指定した場合，最初にエラーとなったデータの要因だけをエラー情報ファイルに出力します。-e オプションを指定しなかった場合，エラーとなったすべてのデータの要因をエラー情報ファイルに出力します。

(a) 適用基準

入力データに不正があった場合に，最初のエラーの時点で処理を中断させることができます。

例えば，DAT 形式の入力データファイルで繰返し列に対してデータロードする場合，入力データは VV 形式で作成しているのに array 文の elmtype オペランドを省略すると FF 形式と扱われるために，すべての入力データがエラーとなります。そして，無駄なエラー情報ファイル，及びエラーデータファイルが作成されてしまいます。このような場合に，-e オプションを指定しておくと，最初にエラーを検知した時点で処理を中断するので，無駄にファイルを作成したりすることはなくなります。

(b) 規則

1. 次の条件の場合は，このオプションの指定に関係なく，処理を中断してロールバックします。

条件	理由
バイナリ形式の入力データファイルで，可変長データの長さ部の領域が負の値の場合	行の区切りがないバイナリ形式では，列長でデータの切り出しをするため，行データを認識する必要があります。しかし，負の値では，データの切り出しができないため，処理を続行できません。
option 文に dataerr オペランドを指定した場合	dataerr オペランドを指定した場合，処理をロールバックするため，処理を継続しても意味がありません。

2. エラー情報ファイルの名称は，制御情報ファイル中で指定しますが，指定がなければデータベース作成ユティリティが入力データファイルと同じホストのディレクトリ下にファイルを作成します。ファイルの名称はデータベース作成ユティリティが決定し，そのファイル名称はメッセージに出力されます。エラー情報ファイル名については，「error ＝エラー情報ファイル名」を参照してください。
3. HiRDB/パラレルサーバで，入力データファイルを用意したサーバ（source 文に指定したサーバ）とデータロード対象表があるサーバが異なる場合，次のエラーデータを検出した時点で処理は中断できません。
   - ユニークキーインデクス，又は主キーインデクスのキー重複エラー
   - 抽象データ型列に格納する値の不正

   特に，HiRDB/パラレルサーバでは，複数サーバに分割された表に対して表単位のデータロードをする場合，-e オプションを指定すると処理中に上記のエラーが発生しても入力した行データがどこまで表に格納されたか保証できません。そのため，次の条件をすべて満たす場合は，-e オプションは指定しないでください。
   - HiRDB/パラレルサーバ
   - 複数サーバに分割された表への表単位のデータロード
   - ユニークキーインデクス，又は主キーインデクスを定義した表，又は抽象データ型列を持つ表
4. option 文の dataerr オペランドを指定した場合にエラーデータを検知したときは，-e オプションを省略していても処理を中断します。

(13)　-r 入力開始行

～＜符号なし整数＞((2～4294967295))《1》

入力データファイルの 1 行目からデータを入力しないで，指定した行からデータを入力する場合に指定します。

(a)　適用基準

-e オプションを指定したデータロードが中断した場合に，再度データロードをするときに指定します。

(b)　注意事項

1. UOC（UOC が入力データファイルを入力する方法）を利用してデータロードする場合，-r オプションを指定しても無視されます。

2. 同期点指定のデータロードが異常終了しロールバックした後に，-r オプションを指定して同期点指定のデータロードを再実行すると，再実行時の入力開始行は次のようになります。

状態	再実行時の入力開始行
-r 入力開始行 ＞同期点指定のデータロードで保存された行	-r 入力開始行
-r 入力開始行 ＝同期点指定のデータロードで保存された行	同期点指定のデータロードで保存された行※
-r 入力開始行 ＜同期点指定のデータロードで保存された行	同期点指定のデータロードで保存された行※

注※

pdload の再実行を示す KFPL00810-I メッセージが出力されます。

(14) -z

可変長文字列，可変長各国文字列，及び可変長混在文字列に対して，長さ 0 のデータを格納する場合に指定します。

(a) 適用基準

可変長文字列，可変長各国文字列，及び可変長混在文字列に対して，ナル値，又は 1 バイトの空白（1 文字の空白）を格納したくない場合に指定します。

(b) 注意事項

1. DAT 形式の場合の，長さ 0 のデータの扱いを次に示します。

   …，""，…（長さ 0 のデータとして扱われます）

   …,，…　　（ナル値として扱われます）

2. バイナリ形式の場合は，実データ長に 0 が設定され，実データ部がない形式を長さ 0 のデータとして扱います。

(15) -y

データロード時，未使用ページを使い切った場合に，使用中ページの未使用領域へデータを格納したいときに指定します。このオプションを指定した場合，未使用領域にデータを格納する前に KFPH26010-I メッセージを出力します。

(a) 適用基準

通常，データロード時に未使用ページを使い切った場合はエラーとなり，処理がロールバックされます。これを避けるため，使用中ページの未使用領域にもデータを格納して，とりあえずデータロードを完了させたい場合に指定します。

(b) 規則

1. -l オプションに p（更新前ログ取得モード）は指定できません。
2. 未使用ページを使い切った後のデータの格納には，ページ内空き領域比率の指定値は適用されません。
3. 未使用ページを使い切った後，-n オプションの指定は適用されません。

(16) -o

インデクス一括作成が正常終了した後，index 文に指定したインデクス情報ファイルを自動的に削除する場合に指定します。なお，idxwork 文に指定したディレクトリ，及び pdload が自動的に作成した%PDDIR%¥tmp ディレクトリのインデクス情報ファイルは，このオプションの指定に関係なく，インデクスの一括作成が正常終了した後に削除されます。

(a) 適用基準

特にプラグインインデクスを作成するときのインデクス情報ファイルは大容量となり，インデクス情報ファイルを削除しないでおくと，ディスク容量を圧迫することになります。このインデクス情報ファイルの削除し忘れに備え，インデクス一括作成終了後に自動的にインデクス情報ファイルを削除する場合に，このオプションを指定します。

(b) 規則

RD エリア単位にデータロードする場合，次の条件に該当するインデクスについては，データロード後に出力したインデクス情報ファイルを使用して pdrorg（-k ixmk）を実行する必要があるため，-o オプションを指定しても無視されます。

- 表格納 RD エリアは 1 サーバ内に複数あるが，対応するインデクス格納 RD エリアは一つだけの場合

(17) -m〔経過メッセージ出力間隔〕〔,インフォメーションメッセージ出力抑止レベル〕

経過メッセージ出力間隔　～<符号なし整数>((1～1000))《10》

処理の実行状況を出力するメッセージの出力間隔を，1 万行単位で指定します。なお，このメッセージは，-m オプションに lvl2 を指定すると，トラブルシュート情報を出力するワークファイルにも出力されます。

＜適用基準＞

省略時仮定値の 10 万行間隔出力が長い場合，又は短い場合に，任意に指定してください。

＜注意事項＞

インデクス一括作成中の場合，このオプションの指定は無効となり，インデクス作成の開始時，及び終了時にだけメッセージが出力されます。

インフォメーションメッセージ出力抑止レベル　～《lvl0》

実行した端末の標準出力，イベントログファイル，及びメッセージログファイルへのインフォメーションメッセージの出力を抑止します。標準出力，イベントログファイル，及びメッセージログファイルのすべての出力を抑止する場合，トラブルシュート情報はワークファイルに出力します。指定できるオプションとメッセージの出力先を次の表に示します。

表 5–7　オプションの指定値とメッセージの出力先

オプションの指定値	意味	出力先		
		標準出力	イベントログ及びメッセージログファイル	ワークファイル
lvl0	出力を抑止しない	○	○	×
lvl1	標準出力への出力を抑止する	×	○	×
lvl2	標準出力，イベントログ及びメッセージログファイルへの出力を抑止する	×	×	○

（凡例）

○：メッセージを出力します。

×：メッセージを出力しません。

ワークファイルにメッセージを出力するのはトラブルシュート目的（ユティリティの動作を確認するため）であり，通常，メッセージを確認する必要がない場合は lvl1 を指定することをお勧めします。

＜ワークファイル＞

ワークファイルは，-m オプションに lvl2 を指定すると，ユティリティを実行するごとに各サーバに作成されます。このため，pdcspool コマンドで HiRDB によって定期的に削除されるように，pd_tmp_directory オペランドにディレクトリを指定しておくことをお勧めします。ワークファイルの出力先を次の表に示します。

表 5–8　ワークファイルの出力先

システム定義の pd_tmp_directory オペランドの指定	環境変数 TMP の指定	ワークファイルの出力先
あり	–	pd_tmp_directory オペランドに指定したディレクトリ
なし	あり	環境変数 TMP に指定したディレクトリ
	なし	%PDDIR%¥tmp ディレクトリ

（凡例）

–：該当しません。

なお，ファイルオープンエラーや I/O エラーなどのためにインフォメーションメッセージがワークファイルへ出力できない場合は，エラーメッセージが出力されますが，処理は続行します。

ワークファイルは，HiRDB が自動で作成します。ファイル名称は，PDLOAD-aa-bb の形式で決められます。それぞれの意味について説明します。

- PDLOAD：pdload が出力したことを示す固定のプリフィクスです。

- aa：プロセスのサーバ名称です。
- bb：ファイル作成時間，及びプロセス ID です。

(18)　-X サーバ間通信の応答監視時間

～＜符号なし整数＞　((1～65535))　《300》

コマンドを実行したサーバで通信障害などの障害が発生した場合，コマンドは無応答となり，業務が停止するおそれがあります。pdload では障害を検知するために，コマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が，-X オプションに設定した時間を超えた場合は，pdload はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。pdload の監視時間には，このほかに pdload の実行監視時間があります。サーバ間通信の応答監視時間と pdload の実行監視時間との関係については，「option 文（データ処理に関する情報の記述）」の「exectime=pdload の実行監視時間」を参照ください。

＜適用基準＞

- 通信障害やユニットダウンによってサーバから応答が返らない状態となった場合に，300 秒より早く障害を検知したいとき，-X オプションに 300 より小さい値を指定してください。
- 系切り替え機能を使用している場合，系切り替えが完了しているにもかかわらず，コマンドが応答を待ち続けることがあります。その場合は，監視時間を短く設定することで，コマンドを即時終了できます。
- 同時実行するアプリケーションやユティリティの多重度が高い場合など，ディクショナリからの応答が遅延し，-X オプションのデフォルトである 300 秒以内にユティリティの前処理が完了しないときは，監視時間でタイムアウトするおそれがあります。このような環境では，-X オプションには 300 より大きい値を指定してください。

(19)　-K{ i ｜ f }

表に定義したすべての XML 型の列について，XML 型のパラメタに対する入力データ値の形式を指定します。表に XML 型の列が定義されていない場合，このオプションを指定するとエラーになります。

i：

XML 型のパラメタに対する入力データ値を入力データファイルに直接記述します。

f：

XML 型のパラメタに対する入力データ値ごとにファイルを用意し，そのファイル名称を入力データファイルに記述します。

(a)　入力データ値の形式

XML 型の列を定義した表へデータロードする場合に選択できる入力データ値の形式を次の表に示します。入力データ値の形式は，-K オプションで指定します。

表 5–9　XML 型の列を定義した表へデータロードする場合の入力データ値の形式

入力データ形式	説明	備考
埋め込み形式（-K i 指定）	ほかの列のデータと同様の入力ファイル形式に従って，入力データファイル内に XML 文書を埋め込みます。	入力データファイル中のデータは BINARY 型の形式で記述します。
ファイルパス指定形式（-K f 指定）	XML データを格納したファイルを個々に用意し，入力データファイル内に対応するファイルパス名を記述します。	入力データファイル中のファイルパス名は BINARY 型の形式，又は CHAR 型の形式で記述します。

(b)　指定できる入力データファイルの形式

XML 文書，及び ESIS-B 形式のデータをデータロードするときに指定できる，入力データファイルの形式を次の表に示します。

表 5–10　指定できる入力データファイルの形式

入力データファイルの形式	入力データ値の形式（-K オプション指定値）と入力データ種別（-G オプション指定値）			
	埋め込み形式（-K i）		ファイルパス指定形式（-K f）	
	ESIS-B 形式（-G esisb）	XML 文書（-G xml）	ESIS-B 形式（-G esisb）	XML 文書（-G xml）
DAT 形式	×	×	○	○
拡張 DAT 形式	×	×	○	○
固定長データ形式（CHAR 型指定）	×	×	○	○
固定長データ形式（BINARY 型指定）	○	○	○	○
バイナリ形式	○	○	×	×
pdrorg 出力バイナリ形式	○	×	×	×

（凡例）

○：pdload で指定できます。

×：pdload で指定できません。

(20)　-G{ xml ｜ esisb }

表に定義したすべての XML 型の列について，入力データファイルとして指定した XML データのデータ種別（XML 文書又は ESIS-B 形式）を指定します。表に XML 型の列が定義されていない場合，このオプションを指定するとエラーになります。

xml：

入力データファイル中の XML 型のパラメタとして指定したデータが，XML 文書であることを示します。

esisb：

入力データファイル中の XML 型のパラメタとして指定したデータが，ESIS-B 形式のデータであることを示します。

(a) 入力データ種別

XML 型の入力データとして，次のデータを選択できます。入力データ種別は，-G オプションで指定します。

- XML 文書

  ユーザが作成した XML のテキストデータです。

- ESIS-B 形式のデータ

  XML 変換コマンド（phdxmlcnv）又は XML 変換ライブラリを使用して，XML 文書を解析した形式のデータです。

(b) XML 型のデータロード時に使用する列構成情報ファイル，及びナル値・関数情報ファイルの指定方法

XML 型の列を定義した表へのデータロード時に，列構成情報ファイル，又はナル値・関数情報ファイルを使用する場合，次のコンストラクタ関数を指定する必要があります。

列構成情報ファイルを使用する場合のコンストラクタ関数は次のとおりです。

入力データファイルの形式	入力データ種別ごとの列構成情報ファイルへの指定方法	
	ESIS-B 形式（-G esisb）	XML 文書（-G xml）
DAT 形式 拡張 DAT 形式	列名, func=(XML,param=binary) [,filedir=パス名]	列名, func=(XMLPARSE,param=integer, param=binary,param=integer) [,filedir=パス名]
固定長データ形式 （CHAR 型指定）	func=(XML,param=binary, type=char()) [,filedir=パス名]	列名, func=(XMLPARSE,param=integer, type=integer,param=binary,type=char() [,null=(先頭位置,[{c ｜ x}]'比較値')] [,filldata ＝埋め字データ] ,param=integer,type=integer) [,filedir=パス名]
固定長データ形式 （BINARY 型指定）	func=(XML,param=binary ,type=binary()) [,filedir=パス名]	列名, func=(XMLPARSE,param=integer, type=integer,param=binary,type=binary() [,null=(先頭位置,[{c ｜ x}]'比較値')] ,param=integer,type=integer)

入力データファイルの形式	入力データ種別ごとの列構成情報ファイルへの指定方法	
	ESIS-B 形式（-G esisb）	XML 文書（-G xml）
		[,filedir=パス名]
バイナリ形式	−	−
pdrorg 出力バイナリ形式	列名, func=(XML,param=binary)	−

（凡例）

−：指定できません。

注

入力データ種別が XML 文書の場合，入力データには，XMLPARSE 関数の第 2 引数のデータだけ指定してください。データの指定方法は，「DAT 形式の場合」，又は「固定長データ形式の場合」を参照してください。第 1 引数と第 3 引数は，pdload がデータ値を設定します。

ナル値・関数情報ファイルを使用する場合のコンストラクタ関数は次のとおりです。

入力データファイルの形式	入力データ種別ごとの列構成情報ファイルへの指定方法	
	ESIS-B 形式（-G esisb）	XML 文書（-G xml）
DAT 形式	−	−
拡張 DAT 形式		
固定長データ形式（CHAR 型指定）		
固定長データ形式（BINARY 型指定）		
バイナリ形式	func=(XML,param=binary [,null=(先頭位置,[{c ｜ x}]'比較値')])	func=(XMLPARSE,param=integer ,param=binary [,null=(先頭位置,[{c ｜ x}]'比較値')] ,param=integer)
pdrorg 出力バイナリ形式	−	−

（凡例）

−：指定できません。

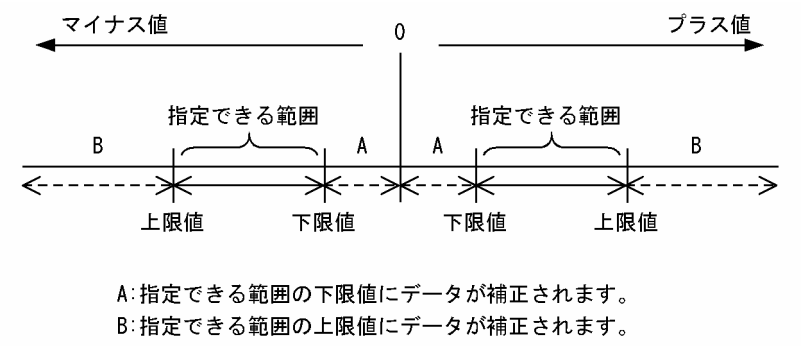
注

入力データ種別が XML 文書の場合，入力データには，XMLPARSE 関数の第 2 引数のデータだけ指定してください。データの指定方法は，「バイナリ形式の場合」を参照してください。第 1 引数と第 3 引数は，pdload がデータ値を設定します。

(21)　-F

FLOAT 型及び SMALLFLT 型の入力データ値が OS の制限値に満たない場合，又は OS の制限値を超えている場合に，入力データ値を OS の制限値内に補正するときに指定します。-F オプション指定時に入力データ値が補正される範囲を次の図に示します。

図 5–12　-F オプション指定時に入力データ値が補正される範囲



(a)　規則

- -F オプションを指定していない場合，入力データ値が OS の制限値に満たない場合は値がそのままとなり，OS の制限値を超える場合はエラーになります。
- -F オプションは，次のデータ型の場合に有効となります。
  - DAT 形式
  - 拡張 DAT 形式
  - 固定長データ形式
  - DAT 形式の pdload 用アンロードファイル
  - 拡張 DAT 形式の pdload 用アンロードファイル
  - 固定長データ形式の pdload 用アンロードファイル

(22)〔認可識別子.〕表識別子

データロードする表の表識別子を指定します。認可識別子を省略した場合，HiRDB に接続したときのユーザの名称が仮定されます。

(a)　規則

認可識別子又は表識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(23)　制御情報ファイル名

データベース作成ユティリティの制御文を記述した制御情報ファイルの名称を指定します。

制御情報ファイル中に記述できる制御文を次に示します。各制御文については，5.4.3〜5.4.16 を参照してください。なお，制御情報ファイル中にコメントは記述できません。

- source 文（入力データファイルの情報の記述）
- index 文（インデクス情報ファイルの情報の記述）
- idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述）

- sort 文（ソート用ワークディレクトリの情報の記述）
- lobdata 文（LOB 入力ファイルの情報の記述）
- lobcolumn 文（列単位 LOB 入力ファイルの情報の記述）
- lobmid 文（LOB 中間ファイルの情報の記述）
- srcuoc 文（UOC 格納ライブラリの情報の記述）
- array 文（繰返し列を含む表の場合の配列データ形式の記述）
- extdat 文（拡張 DAT 形式の情報の記述）
- src_work 文（分割入力データファイルの出力先の記述）
- constraint 文（検査保留状態に関する記述）
- option 文（データ処理に関する情報の記述）

(a)　指定できる制御文の個数

制御情報ファイルに指定できる制御文の個数を次に示します。

制御文	指定できる制御文の最大数
source 文	1
index 文	インデクス数×格納 RD エリア数
idxwork 文	インデクス格納サーバ数
sort 文	インデクス格納サーバ数
lobdata 文	1
lobcolumn 文	LOB 列数
lobmid 文	1
srcuoc 文	1
array 文	1
extdat 文	1
src_work 文	1
constraint 文	1
option 文	1

(b)　制御文とオプションの関係

制御文とオプションの関係を次に示します。

- **インデクスを作成する場合**
  インデクスを作成する場合は，-i オプションの指定によって次の制御文を指定します。

制御文	-i オプションの指定			
	c	n	s	x
source 文	◎	◎	◎	◎
index 文	○	○	－	－
idxwork 文	○	○	－	－
sort 文	○	－	－	－

（凡例）

◎：必要

○：任意

－：不要

- **LOB データ，LOB パラメタを入力する場合**

  LOB データ，LOB パラメタを入力する場合は，LOB 列構成基表，及び LOB 列を同時に作成するか，別に作成するかで，指定する制御文が異なります。

制御文	データロード単位							
	LOB 列構成基表，及び LOB 列へ同時にデータロード				LOB 列構成基表へのデータロード		LOB 列へのデータロード	
	-k f	-k c	-k v	-k d	-k f	-k c	-k f	-k c
source 文	◎	◎	◎	◎	◎	◎	－	－
lobdata 文	◎	◎	◎	－	－	－	◎	○
lobcolumn 文	－	◎	－	－	－	－	－	◎
lobmid 文	◎※	◎	◎※	－	◎※	◎	◎※	◎

（凡例）

◎：必要

○：任意

－：不要

注※

抽象データ型の列に格納する，データを生成するコンストラクタ関数への入力パラメタとする LOB の場合は，指定しても無視されます。

(c)　制御文で記述するファイル，及びディレクトリについて

制御文で記述するファイル，及びディレクトリの規則を次に示します。

1. HiRDB 管理者に対してアクセス権限を与えておく必要があります。

2. パス名は大文字，小文字を区別しません。このため，大文字，小文字が異なるだけの同一パス名を指定すると，重複エラーとなります。
3. ネットワークドライブに接続するパス名は指定できません。

5.4.3　source 文（入力データファイルの情報の記述）

入力データファイルの情報を指定します。

＜適用基準＞

LOB 列がある表の LOB 列にだけデータロードするとき以外は，必ず指定します。

＜規則＞

source 文は必ず 1 行で記述してください（1 行の長さは 1023 バイト以内）。また，source 文は制御情報ファイルの中に一つだけ指定できます。

(1)　形式

```
source 〔RDエリア名〕 〔｛サーバ名｜ホスト名｝：〕
        {入力データファイル名〔，入力データファイル名〕…｜（uoc）}
        〔error＝エラー情報ファイル名〕
        〔errdata＝エラーデータファイル名〔，出力行数〕〕
        〔errwork＝エラーデータファイル作成用バッファサイズ〕
        〔maxreclen=入力データ長〕
```

(2)　説明

(a)　RD エリア名

～＜識別子＞((1～30))

横分割表に対して，RD エリア単位にデータロードをする場合，データロードをする RD エリアの名称を指定します。

＜規則＞

1. キーレンジ分割，又は FIX ハッシュ分割の横分割表の場合，該当する RD エリアに格納するデータかどうかを HiRDB がチェックし，格納範囲不正のときはエラーとします。ただし，フレキシブルハッシュ分割の横分割表の場合，該当する RD エリアに格納するデータかどうかはチェックしないで，そのままデータを格納します。

2. RD エリア名を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

(b) {サーバ名｜ホスト名}：

入力データファイルがあるサーバ名，又はホスト名を指定します。

サーバ名 ～＜識別子＞((1～8))

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。

【HiRDB/パラレルサーバの場合】

入力データファイルがあるフロントエンドサーバ，又はバックエンドサーバの名称を指定します。監査証跡表にデータロードする場合は，監査証跡ファイルがあるユニットの任意のサーバ名称を指定します。なお，影響分散スタンバイレス型系切り替えを適用している場合，サーバ名にゲストBES（切り替え対象サーバ）を指定すると，正規ユニット（そのサーバが実行系として動作しているユニット）に入力データファイルがあるものとして動作します。

RD エリア名を指定した場合は省略できます。省略した場合，指定した RD エリアのあるサーバ名が仮定されます。RD エリア名を省略した場合は，必ず指定してください。

ホスト名 ～＜識別子＞((1～32))

【HiRDB/シングルサーバの場合】

入力データファイルがあるホストの名称を指定します。シングルサーバがあるホスト名を指定してください。

RD エリア名の指定に関係なく省略できます。省略した場合，データベース作成ユティリティ（pdloadコマンド）を入力したホスト名が仮定されます。なお，系切り替え機能を使用している場合は，現用系のホスト名を指定してください。

【HiRDB/パラレルサーバの場合】

HiRDB/パラレルサーバの場合は指定しないでください。

(c) 入力データファイル名

～＜パス名＞

入力データが入っている入力データファイルの名称を絶対パス名で指定します。

入力データファイルが複数ある場合は，コンマ（,）で区切って指定できます。入力データファイルについては，「入力データファイル」を参照してください。

また，入力データファイルに対してはテープ装置アクセス機能を使用できます。テープ装置アクセス機能については，「テープ装置アクセス機能」を参照してください。

【HiRDB/シングルサーバの場合】

シングルサーバがあるサーバマシンに作成してください。

【HiRDB/パラレルサーバの場合】

フロントエンドサーバ，又はバックエンドサーバに作成してください。

＜規則＞

1. 指定した入力データファイルに対して，データロード処理をする前にアクセス可否のチェックをします。この場合，アクセス不可のときにはデータロード処理は実行されません。
2. BOM（Byte Order Mark）を持つファイルの使用可否については，「規則及び注意事項」の「BOM を持つファイルの使用」を参照ください。

(d) （uoc）

UOC を使用する場合，入力データファイルの入出力を UOC 側でするときに指定します。なお，監査証跡表にデータロードする場合，システムが提供する UOC を使用するため，このオペランドを指定してください。

ユーザが作成する UOC については，「UOC を利用したデータロード」を参照してください。

(e) error ＝エラー情報ファイル名

～＜パス名＞

エラー情報を出力するファイルの名称を絶対パス名で指定します。

入力データファイルがあるサーバのファイル名を指定してください。指定したファイルが存在しない場合，指定したファイル名でユティリティがファイルを作成するため，事前にファイルを作成しておく必要はありません。なお，error オペランドを省略すると，次に示す形式でユティリティがユニークな名称のファイルを作成します。

```
¥ディレクトリ名¥ERROR-xxxxxxxxx
```

ディレクトリ名：表「pdload がファイルを出力するディレクトリ」に示すディレクトリ

ERROR-：エラー情報ファイルのプリフィクス

xxxxxxxxx：ファイル作成時間とプロセス ID を文字列に変換した値

ユティリティが作成したファイルの名称は，KFPL00709-I メッセージに出力されます。

また，-e オプションを指定している場合は，入力データにエラーがあるときだけエラー情報ファイルが作成されます。

なお，HiRDB/パラレルサーバの場合，BES が検知した入力データのエラー情報はエラー情報ファイル作成用一時ファイルとして一時的に作成されます。このファイルは，次に示す形式で各 BES に作成されます。

```
¥ディレクトリ名¥ERRTMP-xxxxxxxxx
```

ディレクトリ名：表「pdload がファイルを出力するディレクトリ」に示すディレクトリ

ERRTMP-：エラー情報ファイルのプリフィクス

xxxxxxxxx：一意なファイル名，ファイル作成時間，及びプロセス ID を文字列に変換した値

(f)　errdata=エラーデータファイル名〔，出力行数〕

入力データがエラーとなった場合，そのエラーとなった行データを出力するときに指定します。

エラーデータファイルに出力された行データを修正すれば，再度入力データファイルとしてデータロードできます。

-e オプションを指定した場合は，エラーデータファイルを指定しても無効となります。

エラーデータファイル名　～＜パス名＞

エラーとなった行データを出力するファイルの名称を絶対パス名で指定します。

このファイルは，入力データファイルと同じサーバマシンに作成する必要があります。省略した場合，エラーの行データは出力されません。

出力行数　～＜符号なし整数＞((1～4294967295))《100》

エラーとなった行データの最大出力行数を指定します。

エラーの行データが，ここで指定した行数より多い場合でも処理は続行されます。ただし，指定した行数までしかエラーの行データは出力されません。

＜規則＞

1. エラーデータファイルを指定するときは，ファイル名，及びパス名の長さは，次の計算式の値以下にしてください。

   ＜パス名の長さ＞（単位：バイト）

   OS 上のパス名の上限値−8

   ＜ファイル名の長さ＞（単位：バイト）

   OS 上のファイル名の上限値−8

   なお，エラーデータファイルを使用する場合，エラーデータを格納するためのエラーデータファイル作成用一時ファイルが一時的に作成されることがあります。エラーデータファイル作成用一時ファイルが作成される条件とファイル名については，「エラー情報参照時の注意事項」を参照してください。

2. エラーデータファイルの出力結果には，次の制限があります。

   ＜ DAT 形式の入力データファイルの場合＞

   maxreclen オペランドの指定値よりデータ長が大きい場合は出力対象となりません。

   maxreclen オペランドを省略した場合，32 キロバイト以上のデータがあるときは，出力対象となりません。

＜バイナリ形式の入力データファイルの場合＞

可変長文字列データの長さ部が不正なために 1 行に編集できなかったデータは，編集された途中までしか出力されません。また，固定長の列データが定義長に満たない場合は，出力対象となりません。

＜列単位 LOB 入力ファイルを使用した場合＞

エラーデータファイルには，エラーとなった入力データファイル中の行データだけが出力されます。そのため，入力データファイルを修正して再度データロードする場合，列単位 LOB 入力ファイルを入力データファイルの出力順序に合わせて修正する必要があります。

3. インデクスキー値の重複エラーの場合，次の条件のときはエラーであってもエラーデータファイル出力の対象とはなりません。
   - インデクス一括作成モードの場合
   - HiRDB/パラレルサーバの場合，インデクス更新モードで，かつ表格納用 RD エリアがあるサーバとは別のサーバに入力データファイルがあり，エラーデータファイル作成用のバッファが不足したとき。
4. LOB 列にデータロードする場合，及び pdrorg 出力バイナリ形式の入力データファイルを使用する場合は，エラーデータファイル出力の対象とはなりません。
5. バイナリ形式の入力データファイルに可変長文字列がある場合，その可変長文字列にエラーがあっても，エラーの列の直前までしか出力対象となりません。エラーデータファイルを参照する場合には注意してください。

(g) errwork ＝エラーデータファイル作成用バッファサイズ

～((0～2097152))

errdata オペランド指定時に，エラーデータファイル作成用のバッファサイズをキロバイト単位で指定します。

このオペランドは，errdata オペランドを指定してデータロードしたときに KFPL25222-W メッセージが出力され，一部のエラーデータがエラーデータファイルに出力されなかった表に対して，再度データロードする場合に指定します。

なお，errwork オペランドに 0 を指定すると，キー重複エラー及び抽象データ型列への値不正エラーは出力されませんが，データロードの実行性能は良くなります。

KFPL25222-W メッセージが出力されるのは，次の三つの条件をすべて満たす場合です。

1. HiRDB/パラレルサーバで source 文に指定したサーバ名と表格納 RD エリアが定義されているバックエンドサーバ名が異なる場合，HiRDB/パラレルサーバで複数のバックエンドサーバに横分割されている表の場合。
2. データロード対象の表に，ユニークキーインデクス又は主キーインデクスが定義されている場合，又は抽象データ型の列がある場合。
3. 次のどれかに該当する場合

- 入力データファイル中にデータベースに格納しないデータ（列構成情報ファイルで skipdata 文を指定したデータ，列定義長よりも長いデータ，ナル比較値と一致してナル値となるデータ）がある場合
- VARCHAR，NVARCHAR，MVARCHAR，BINARY，又は LOB 列へのデータの格納の場合
- コンストラクタ関数への入力パラメタ型が，VARCHAR，NVARCHAR，MVARCHAR，BINARY，又は LOB の抽象データ型へデータを格納する場合

＜見積もり式＞

エラーデータファイル作成用バッファサイズの見積もり式を次に示します。見積もり式の単位はバイトなので，指定するときはキロバイトに補正（端数切り上げ）してください。

```
バッファサイズ＝｛↑Ｘ÷（データベース格納平均行長＋Ｙ）↑｝
                  ×入力データ平均行長×２×
                  表格納用RDエリアが定義されているサーバ数

    Ｘ：システム共通定義pd_utl_buff_sizeの指定値×１０２４
    Ｙ：FIX表の場合…２４
        非FIX表の場合…（列数＋1）×４＋２４
```

データベース格納平均行長（表の格納ページ数の計算方法）については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

なお，上記見積もり式は行長の平均値で計算しているので，行データの並びによってはバッファが不足することがあります。メモリに余裕がある場合は，見積もり式の値を次の値に変更することで，確実にエラーデータファイルに出力できます。

- 非 FIX 表の場合，データベース格納平均行長を 0 にします。
- 入力データ平均行長を，入力データの最大長にします。

(h)　maxreclen=入力データ長

入力データファイルが DAT 形式，拡張 DAT 形式，バイナリ形式，及び pdrorg 出力バイナリ形式の場合に指定できます。固定長データ形式の場合，及びストリーミングテープ装置を使用する場合にこのオペランドを指定しても無視されます。

入力データファイルが DAT 形式又は拡張 DAT 形式の場合　～＜符号なし整数＞((0，32～524288))≪32≫

入力データファイルが DAT 形式の場合で，1 行のデータ長が 32 キロバイトを超えるとき，入力データファイル中の 1 行の最大データ長をキロバイトで指定します。

このオペランドを省略した場合に 1 行のデータ長が 32 キロバイト以上のデータがあるとき，又はオペランドの指定値よりも長いデータがあると，処理を打ち切ります。

＜規則＞

1. このオペランドを省略する場合，又はストリーミングテープ装置を使用する場合は，入力データファイル（DAT 形式）の 1 行のデータ長は 32 キロバイト以下でなければなりません。
2. pdrorg で-W dat オプションを指定して出力した DAT 形式のアンロードデータファイルを使用する場合は，KFPL22222-I メッセージに最大長が表示されます。

3. このオペランドに 0 を指定すると，pdload が次の基準で処理対象の表の定義から，行長を計算します。なお，計算値は表定義を利用しているため，入力データ中に表に格納しないデータがある場合や，拡張 DAT 形式で囲み文字（"）が二重に出力される場合には，計算値が入力データ長と一致しなくなるのでエラーとなります。この場合，入力データ長には 0 以外の値を指定してください。

データ型，又はパラメタ型	計算値（単位：バイト）	計算値の算出式
INTEGER	12	符号＋けた数＋セパレータ文字
SMALLINT	7	符号＋けた数＋セパレータ文字
DECIMAL（m,n）	m ＋ 3	符号＋けた数＋小数点＋セパレータ文字
FLOAT	24	符号＋可数部＋小数点＋ e ＋符号＋指数部＋セパレータ文字
SMALLFLT	17	符号＋可数部＋小数点＋ e ＋符号＋指数部＋セパレータ文字
DATE	11	指定形式＋セパレータ文字
TIME	9	指定形式＋セパレータ文字
INTERVAL YEAR TO DAY	11	符号＋けた数＋コンマ＋セパレータ文字
INTERVAL HOUR TO SECOND	9	符号＋けた数＋コンマ＋セパレータ文字
TIMESTAMP(p)	19 ＋(p ＋ 1)＋ 1	指定形式＋セパレータ文字
CHAR(n)，MCHAR(n)，VARCHAR(n)，又は MVARCHAR(n)	n ＋ 3	文字数＋引用符＋セパレータ文字
NCHAR(n)，又は NVARCHAR(n)	(n×2)＋ 3	文字数＋引用符＋セパレータ文字
BINARY(n)	n ＋ 3	データ数＋引用符＋セパレータ文字
BLOB	1025	パス名最大値＋セパレータ文字

入力データファイルがバイナリ形式又は pdrorg 出力バイナリ形式の場合　～＜符号なし整数＞((0，32～2097152))≪0≫

BINARY 列がある表にデータロードする場合，入力データ中の行の長さの最大長をキロバイト単位で指定します。BINARY 列がない表にデータロードする場合にこのオペランドを指定しても無視されます。

＜規則＞

1. このオペランドを省略，又は 0 を指定した場合，pdload が表の定義から最大行長を算出し，その値を使用して処理します。ただし，入力データ長を指定した場合でも，pdload が表の定義から算出した値の方が小さければ，算出した値を使用して処理します。
2. pdload が表の定義から算出する最大行長より，実際の最大データ長が小さい場合，確保する領域を最小限にするためにこのオペランドを指定します。

5.4.4　index 文（インデクス情報ファイルの情報の記述）

インデクス一括作成モード，又はインデクス情報出力モードでデータロードする場合に，インデクス情報を出力するインデクス情報ファイルの情報を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，index 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないよう，できるだけ指定するようにしてください。

インデクスの数，又はインデクスの格納 RD エリア数が多い場合は，idxwork 文を指定すると便利です。

＜規則＞

1. index 文は，（インデクス数×インデクス格納 RD エリア数）分指定します。
2. index 文を省略した場合，ユティリティがファイル名称を決定し，インデクス情報ファイルを作成します。作成先ディレクトリは，idxwork 文を省略している場合，インデクス格納 RD エリアがあるサーバの，表「pdload がファイルを出力するディレクトリ」に示すディレクトリになります。

   ユティリティが名称を決定してインデクス情報ファイルを作成した場合，インデクス一括作成完了後，ユティリティがファイルを削除しますが，index 文を指定した場合，-o オプションを指定しないとインデクス一括作成完了後にファイルを削除しません。そのため，ユーザがファイルを削除する必要があります。
3. index 文と idxwork 文を同時に指定した場合，index 文の指定が有効となります。
4. 処理対象表にインデクスが定義されていない場合，index 文は指定しないでください。

(1)　形式

```
index インデクス識別子 〔RDエリア名〕 インデクス情報ファイル名
```

(2)　説明

(a)　インデクス識別子

インデクスの識別子を指定します。

インデクス識別子を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，インデクス識別子に空白が含まれる場合は，引用符で囲んでください。

(b)　RD エリア名

～＜識別子＞((1～30))

横分割表の場合，インデクスを格納している RD エリアの名称を指定します。

RD エリア名を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

(c)　インデクス情報ファイル名

～＜パス名＞

インデクス情報を出力するインデクス情報ファイルの名称を，絶対パス名で指定します。

インデクス情報ファイルは，インデクス格納 RD エリアがあるシングルサーバ，又はバックエンドサーバを指定してください。

5.4.5　idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述）

index 文を省略した場合に，自動的にインデクス情報ファイルが作成されるディレクトリの名称を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，idxwork 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないよう，できるだけ指定するようにしてください。

＜規則＞

1. index 文，及び idxwork 文を省略した場合，インデクス格納 RD エリアがあるサーバの，表「pdload がファイルを出力するディレクトリ」に示すディレクトリにインデクス情報ファイルが作成されます。
2. idxwork 文は，次の数だけ指定できます。

   【HiRDB/シングルサーバの場合】

   一つだけ idxwork 文を指定します。

   【HiRDB/パラレルサーバの場合】

   分割格納された表のインデクスの場合，分割格納先のサーバの数だけ idxwork 文を指定します。分割格納されていない表や，分割格納された表を RD エリア単位にデータロードする場合には一つだけ指定します。指定するサーバ名は，インデクス格納 RD エリアがあるサーバの名称を指定します。
3. idxwork 文と index 文を同時に指定した場合，index 文の指定が有効となり，idxwork 文の指定は無視されます。

(1)　形式

```
idxwork 〔サーバ名〕 ディレクトリ名
```

(2)　説明

(a)　サーバ名

～＜識別子＞((1～8))

インデクス情報ファイルを作成するサーバの名称を指定します。

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。シングルサーバの名称を指定しても無視されます。

【HiRDB/パラレルサーバの場合】

インデクス情報ファイルを作成するサーバの名称を指定します。

(b)　ディレクトリ名

～＜パス名＞((1～255))

インデクス情報ファイルを作成するディレクトリの名称を絶対パス名で指定します。

(3)　注意事項

自動作成されるインデクス情報ファイルの名称を次に示します。

```
ディレクトリ名¥INDEX-インデクス名-インデクス格納用RDエリア名
--一意となる文字列
```

（例）

idxwork 文のディレクトリ名が c:¥hirdb¥pdload，インデクス名が IDX1，インデクス格納用 RD エリア名が USER01 の場合，作成されるインデクス情報ファイルの名称は次のようになります。

```
c:¥hirdb¥pdload¥INDEX-IDX1-USER01-aaaa00001
```

注意事項

インデクス名又はインデクス格納 RD エリア名に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

5.4.6　sort 文（ソート用ワークディレクトリの情報の記述）

B-tree インデクスがある表に対して，インデクス一括作成モードでデータロードをする場合，ソート用ワークファイルの情報（ファイルの作成ディレクトリなど）を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，sort 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないよう，できるだけ指定するようにしてください。

＜規則＞

1. sort 文を省略した場合，ソート用ワークファイルはインデクス格納 RD エリアがあるサーバの，表「pdload がファイルを出力するディレクトリ」に示すディレクトリに作成されます。
2. 表にプラグインインデクスだけが定義されている場合，指定しても無視されます。
3. ソート用ワークファイルは，pdload の処理終了後に自動的に削除されます。
4. sort 文は，次の数だけ指定できます。

   【HiRDB/シングルサーバの場合】

   一つだけ sort 文を指定します。

   【HiRDB/パラレルサーバの場合】

   分割格納された表のインデクスの場合，分割格納先のサーバの数だけ sort 文を指定します。分割格納された表のインデクスの場合でも，RD エリア単位にデータロードする場合には，一つだけ sort 文を指定します。また，分割格納しないインデクスの場合も，一つだけ sort 文を指定します。

(1) 形式

```
sort 〔サーバ名〕 ディレクトリ名
```

(2) 説明

(a) サーバ名

～＜識別子＞((1～8))

ソート用ワークファイルを作成するサーバの名称を指定します。

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。シングルサーバの名称を指定しても無視されます。

【HiRDB/パラレルサーバの場合】

ソート用ワークファイルを作成するサーバの名称を指定します。

(b) ディレクトリ名

～＜パス名＞

ソート用ワークファイルを作成するディレクトリの名称を，絶対パス名で指定します。

5.4.7　lobdata 文（LOB 入力ファイルの情報の記述）

LOB 列がある表へデータロードする場合，又はコンストラクタ関数への入力パラメタとして LOB データを入力する場合に，LOB の情報を指定します。

＜適用基準＞

lobdata 文は，LOB 列がある表へデータロードする場合に指定してください（ただし，LOB 列構成基表だけをデータロードするときには不要です）。

また，抽象データ型列がある表に対して，コンストラクタ関数へ入力パラメタとして LOB データを入力してデータロードする場合も，lobdata 文を指定できます。

＜規則＞

1. lobdata 文は，-k f，c，又は v の場合に指定できます。
2. データロードで BLOB 列を含む表に対しデータを挿入する場合は，次の 2 つの方法があります。

   ・BLOB 列データと LOB 列構成基表を 1 回の pdload 実行で同時に挿入します。

   この際，source 文と，lobdata 文の指定が必要です。

   ・BLOB 列データと LOB 列構成基表を 2 回の pdload 実行で別々に挿入します。

   この際，1 回目の pdload では，source 文と lobmid 文を指定し，2 回目の pdload では 1 回目に指定した lobmid 文と lobdata 文の指定が必要です。

   なお，2 回目の pdload を実行しないで，UPDATE 文を使って BLOB 列データを挿入することはできません。UPDATE 文を使用して BLOB 列データを挿入したい場合は，1 回目の pdload の入力データファイルの BLOB 列にはナル値を挿入する指定をしてください。
3. LOB 列及び LOB パラメタ（コンストラクタ関数の入力パラメタとしての LOB データ）ごとに LOB 入力ファイルを用意する場合，入力データファイル中の LOB 列に該当するデータ部にはファイル名を記述します。

   しかし，そのファイル名を絶対パス名で記述すると，入力データファイルの容量が増大し，記述も面倒です。

   LOB 入力ファイルが，あるディレクトリ下にまとめて用意されている場合，ディレクトリ名は lobdata 文に，ファイル名は入力データファイル中に記述することで，指定したディレクトリ名とファイル名を合わせて絶対パス名として認識できます。ただし，ディレクトリ名の指定は，入力データファイルに LOB 入力ファイルのファイル名を記述する場合だけ有効で，入力データファイルに LOB データを直接記述する場合は無視されます。

(1)　形式

```
lobdata 〔LOB入力ファイルのディレクトリ名〕
```

(2) 説明

(a) LOB 入力ファイルのディレクトリ名

～＜パス名＞

LOB 入力ファイルがあるディレクトリ名を指定します。

LOB 入力ファイルが一つのディレクトリ下にまとまっている場合に，そのディレクトリ名を絶対パス名で指定します。

ここで指定したパス名と，source 文で指定した入力データファイル中の LOB 列のファイル名称，又は lobcolumn 文の列単位 LOB 入力ファイルの名称を合わせて，LOB 入力ファイルの絶対パス名となるように指定してください。

＜注意事項＞

1. 作成した絶対パス名の長さは最大 1,023 バイトです。
2. source 文で指定した入力データファイル中の LOB 入力ファイルの名称，又は lobcolumn 文の列単位 LOB 入力ファイルの名称を絶対パス名で指定している場合は，LOB 入力ファイルのディレクトリ名は省略できます。

(3) 注意事項

LOB 列，LOB 属性の抽象データ型の列，及び lobdata 文の有無によるデータロードの方法を次に示します。なお，LOB 属性の抽象データ型として SGMLTEXT 型を例に説明します。

<table>
<tr><th colspan="2" rowspan="2">表定義</th><th colspan="2">lobdata 文の指定</th></tr>
<tr><th>あり</th><th>なし</th></tr>
<tr><td rowspan="2">LOB 列あり</td><td>SGMLTEXT 型の列あり</td><td>LOB 列構成基表，SGMLTEXT 型の列，及び LOB 列へ同時にデータロードします。</td><td>LOB 列構成基表，及び SGMLTEXT 型の列へ同時にデータロードし，後から LOB 列へデータロードします。</td></tr>
<tr><td>SGMLTEXT 型の列なし</td><td>LOB 列構成基表，及び LOB 列へ同時にデータロードします。</td><td>LOB 列構成基表へ先にデータロードし，後から LOB 列へデータロードします。</td></tr>
<tr><td rowspan="2">LOB 列なし</td><td>SGMLTEXT 型の列あり</td><td colspan="2">抽象データ型列構成基表，及び SGMLTEXT 型の列へ同時にデータロードします。</td></tr>
<tr><td>SGMLTEXT 型の列なし</td><td>エラーとなります。</td><td>LOB 列なし，SGMLTEXT 型の列なしでデータロードします。</td></tr>
</table>

5.4.8 lobcolumn 文（列単位 LOB 入力ファイルの情報の記述）

LOB 列がある表へデータロードする場合に，列単位 LOB 入力ファイルを使用するとき，そのファイルの情報を指定します。

＜適用基準＞

LOB 列がある表に対して，列単位 LOB 入力ファイルを使用してデータロードする場合に指定します。

列単位 LOB 入力ファイルは，何らかのプログラムを使用して他データベースからデータを移行するときなどに作成したりします。

＜規則＞

1. lobcolumn 文は，-k c の場合にだけ指定できます。
2. 作成する表に LOB 列が複数ある場合は，その列数分 lobcolumn 文を指定する必要があります。ただし，LOB 列を RD エリア単位に作成する場合は，一つだけ指定できます。

   また，LOB 列構成基表の入力データファイルが DAT 形式の場合で，そのファイルにすべての列の記述がなく，対応する列を列構成情報ファイルで指定しているときは，記述のない LOB 列に対する lobcolumn 文は省略できます。指定した場合は，KFPL16325-W メッセージを出力し，指定した lobcolumn 文の内容は無視され，表定義時の LOB 列の指定に従ってナル値又はデフォルト値として LOB データを作成します。
3. LOB 列構成基表だけデータロードする場合には，指定する必要はありません。
4. 抽象データ型の列を含む表に対しての指定はできません。
5. 処理対象表に LOB 列がない場合，lobcolumn 文は指定しないでください。

(1) 形式

```
lobcolumn 〔LOB列の列名〕
            列単位LOB入力ファイル名〔，列単位LOB入力ファイル名〕…
```

(2) 説明

(a) LOB 列の列名

データロードする表の LOB 列の列名を指定します。

LOB 列が一つしかない場合，又は LOB 列を RD エリア単位にデータロードする場合には省略できます。

＜規則＞

1. 表に LOB 列が複数ある場合は必ず指定してください。
2. LOB 列の列名を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，LOB 列の列名に空白が含まれる場合は，引用符で囲んでください。

(b) 列単位 LOB 入力ファイル名〔，列単位 LOB 入力ファイル名〕...

～＜パス名＞

列単位 LOB 入力ファイルの名称を指定します。

指定したファイル名と lobdata 文で指定したディレクトリ名を合わせて，絶対パス名になるように指定してください。lobdata 文のディレクトリ名を省略している場合は，ここで指定するファイル名を絶対パス名にしてください。

＜規則＞

1. 列単位 LOB 入力ファイルのファイル名の長さは，最大 1,023 です。
2. ファイル名を複数指定する場合は，LOB 列構成基表の入力データファイルの列の順番と合わせて指定してください。
3. 列単位 LOB 入力ファイルは次の場所に用意してください。

【HiRDB/シングルサーバの場合】

列単位 LOB 入力ファイルは，シングルサーバのホストに準備してください。ホスト名は，lobmid 文で指定します。

【HiRDB/パラレルサーバの場合】

列単位 LOB 入力ファイルは，フロントエンドサーバ又はバックエンドサーバに準備してください。サーバ名は，lobmid 文で指定します。

5.4.9 lobmid 文（LOB 中間ファイルの情報の記述）

LOB 列がある表へデータロードする場合，LOB 中間ファイルの情報を指定します。

LOB 中間ファイルには，LOB 列構成基表作成時に LOB 列のデータロードに必要な情報が出力されます。

＜適用基準＞

LOB 列がある表へデータロードする場合で，-k d 以外のときは必ず指定してください。

ただし，抽象データ型の列に格納する値を生成する，コンストラクタ関数への入力パラメタの LOB の場合は，指定しても無視されます。

＜規則＞

1. -k f，c，及び v の場合に指定してください。
2. LOB 中間ファイルの内容は，-k オプションの指定によって異なります。LOB 列構成基表と LOB 列を別々に作成する場合は，- k オプションの指定を同じにする必要があります。
3. LOB 列構成基表にデータロードする場合，lobmid 文の指定を省略すると，入力データファイルが存在するサーバに，次に示す形式でユティリティがユニークな名称の LOB 中間ファイルを作成します。

```
¥ディレクトリ名¥LOBMID-xxxxxxxxx
```

ディレクトリ名：表「pdload がファイルを出力するディレクトリ**」**に示すディレクトリ

LOBMID-：LOB 中間ファイルのプリフィクス

xxxxxxxx：ファイル作成時間とプロセス ID を文字列に変換した値

4. LOB 列構成基表と LOB 列を同時に作成する場合は source 文の指定が優先されるため，サーバ名，ホスト名，及びエラー情報ファイル名は指定する必要はありません。指定した場合は無視されます。

(1) 形式

```
lobmid 〔RDエリア名〕 {サーバ名|ホスト名} :
        LOB中間ファイル名〔,LOB中間ファイル名〕…
       〔error=エラー情報ファイル名〕
```

(2) 説明

(a) RD エリア名

～＜識別子＞((1～30))

ユーザ LOB 用 RD エリアの名称を指定します。

省略した場合，LOB 中間ファイルのすべての内容を，データロードの対象とします。

RD エリア名を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

(b) {サーバ名|ホスト名}：

LOB 中間ファイルがあるサーバ名，又はホスト名を指定します。

サーバ名 ～＜識別子＞((1～8))

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。

【HiRDB/パラレルサーバの場合】

LOB 中間ファイルがあるフロントエンドサーバ，又はバックエンドサーバの名称を指定します。

RD エリア名を指定した場合は省略できます（RD エリア名，又はサーバ名のどちらかを必ず指定してください）。

サーバ名を省略した場合，指定した RD エリアのあるサーバ名が仮定されます。

ホスト名 ～＜識別子＞((1～32))

【HiRDB/シングルサーバの場合】

LOB 中間ファイルがあるホストの名称を指定します。

シングルサーバがあるホスト名を指定してください。

RD エリア名の指定に関係なく省略できます。省略した場合，データベース作成ユティリティ（pdload コマンド）を入力したホスト名が仮定されます。

【HiRDB/パラレルサーバの場合】

HiRDB/パラレルサーバの場合は指定しないでください。

(c)　LOB 中間ファイル名

～＜パス名＞

LOB 中間ファイルの名称を絶対パス名で指定します。

LOB 列構成基表作成時は，この LOB 中間ファイルに LOB 列のデータロード時に必要な情報が出力されます。そして，LOB 列のデータロード時には，LOB 中間ファイルから必要な情報を取り出します。

(d)　error=エラー情報ファイル名

～＜パス名＞

エラー情報を出力するファイルの名称を絶対パス名で指定します。

LOB 中間ファイルと同じユニットのファイルを指定してください。指定したファイルが存在しない場合，指定したファイル名でユティリティがファイルを作成するため，事前にファイルを作成しておく必要はありません。なお，error オペランドを省略すると，次に示す形式でユティリティがユニークな名称のファイルを作成します。

```
¥ディレクトリ名¥ERROR-xxxxxxxxx
```

ディレクトリ名：表「pdload がファイルを出力するディレクトリ**」**に示すディレクトリ

ERROR-：エラー情報ファイルのプリフィクス

xxxxxxxx：ファイル作成時間とプロセス ID を文字列に変換した値

(3)　LOB 中間ファイルに関する注意事項

LOB 列構成基表と LOB 列を別々に作成する場合，LOB 列作成時には次のことに注意する必要があります。

1. 同じ入力データファイルを使用して複数回の pdload で LOB 列構成基表を作成した場合，それぞれの pdload で出力された LOB 中間ファイルは，LOB 列作成時の lobmid 文に同時に指定できます。ただし，同時に指定する場合は，LOB 列構成基表の入力データファイルの行の順番と対応するように LOB 中間ファイルの指定順を考慮する必要があります。

   LOB 中間ファイルの指定順が不正な場合，列単位 LOB 入力ファイルとの対応がとれなくなり，エラーとなります。

2. 異なる入力データファイルを使用して複数回の pdload で LOB 列構成基表を作成した場合，それぞれの pdload で出力された LOB 中間ファイルは，LOB 列作成時の lobmid 文に同時に指定できません。

同時に指定した場合，列単位 LOB 入力ファイルとの対応がとれなくなり，エラーとなります。したがって，LOB 列作成時には，それぞれの入力データファイルに対応する列単位 LOB 入力ファイルを準備し，それぞれ別の pdload で作成する必要があります。

5.4.10　srcuoc 文（UOC 格納ライブラリの情報の記述）

UOC を利用して，データを編集してからデータベースに格納する場合，UOC の情報を指定します。UOC については，「UOC を利用したデータロード」を参照してください。

監査証跡表にデータロードする場合，監査証跡ファイルの入力と編集はシステムが提供する UOC（**監査証跡データ登録 UOC**）が行います。そのため，監査証跡表にデータロードする場合は srcuoc 文の指定が必要になります。

＜適用基準＞

UOC を利用してデータロードする場合，又は監査証跡表にデータロードする場合に指定します。

(1)　形式

```
ユーザが作成したUOCの場合
  srcuoc ライブラリ名称 entry=関数名 〔param='ユーザパラメタ'〕

監査証跡表の場合（監査証跡データ登録UOCの場合）
  srcuoc param='監査証跡データ登録UOC用パラメタ'
```

(2)　説明

(a)　ライブラリ名称

～＜パス名＞

UOC を格納しているダイナミックリンクライブラリの名称を絶対パス名で指定します。

(b)　entry=関数名

ライブラリ中のコールする関数名を指定します。

(c)　param='ユーザパラメタ'

～＜文字列＞((1～1023 バイト))

UOC に対してユーザはコマンドラインなどを通して直接パラメタを渡すことができないため，UOC にパラメタを渡したい場合にその内容をユーザパラメタに指定します。

このオペランドに記述した内容は，インタフェース領域を介して指定した文字列のまま UOC に渡されます。

なお，ユーザパラメタに指定する文字列には空白，又はタブを指定しないでください。

(d) param='監査証跡データ登録 UOC 用パラメタ'

～<文字列>((1～4095 バイト))

データロード対象となる監査証跡ファイルの情報を次の形式で指定します。

```
param’dir=監査証跡ファイルが存在するHiRDBファイル名
        , file= { (データロード対象となる監査証跡ファイル名
                〔, データロード対象となる監査証跡ファイル名…〕 ) | all}
      〔, mode= {normal | force} 〕’
```

dir=監査証跡ファイルが存在する HiRDB ファイル名

監査証跡ファイルが存在する HiRDB ファイル名を指定します。

file= {(データロード対象となる監査証跡ファイル名〔, ...〕) | all}

データロード対象とする監査証跡ファイル名を指定します。

dir オペランドに指定した HiRDB ファイル内のすべての監査証跡ファイルをデータロード対象とする場合は，all を指定します。

mode= {normal | force}

データロード対象とする監査証跡ファイルの状態によってデータロードするかどうかを指定できます。mode オペランドの指定値とデータロードの結果を次に示します。

監査証跡ファイルの状態		mode 指定値	データロードの結果
閉塞有無	データロード状態		
閉塞していない	済み	normal	×
	待ち		○
	済み	force	○※
	待ち		○※
閉塞している	済み	normal	×
	待ち		×
	済み	force	×※
	待ち		×※
現用	−	normal	×
		force	×

(凡例)

○：データロードします。

×：データロードしません。

−：該当しません。

注※

file=all を mode=force と組み合わせて指定すると，制御文エラーになります。

なお，監査証跡ファイルの名称や状態は，pdls -d aud コマンドで確認できます。

5.4.11　array 文（繰返し列を含む表の場合の配列データ形式の記述）

繰返し列を含む表の場合，入力データファイルに記述する配列データ形式とナル値の扱いを指定します。

＜適用基準＞

繰返し列を含む表にデータロードする場合に指定します。

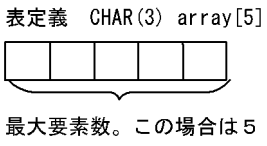
この制御文を省略した場合，配列データ形式は DAT 形式ファイル，及び固定長データ形式ファイルのときは FF 形式，バイナリ形式ファイルのときは VV 形式が仮定されます。また，ナル値オプションは c が仮定されます。

＜規則＞

1. この制御文は表に含まれるすべての繰返し列に対して指定します。このため，列ごとに指定を変更する場合は，列構成情報ファイル，又はナル値・関数情報ファイルで指定してください。
2. 繰返し列を含む表にデータロードする場合の説明に，次に示す用語を使用します。

最大要素数：

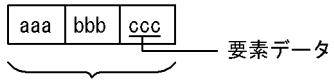
表定義時に該当する列に指定した要素数です。



最大要素数。この場合は５

指定要素数：

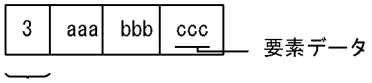
最大要素数のうち，入力データファイルに指定する要素数です。列構成情報ファイル，ナル値・関数情報ファイルで指定できます。



指定要素数。この場合は３

実要素数：

入力データファイル中に指定した有効な要素数です。



実要素数。この場合は３

(1) 形式

```
array 〔elmtype=配列データ形式〕 〔nullset=ナル値オプション〕
```

(2) 説明

(a) elmtype=配列データ形式

ff，fv，又は vv を指定します。

elmtype オペランドを省略した場合，DAT 形式ファイル，及び固定長データ形式ファイルのときは FF 形式，バイナリ形式ファイルのときは VV 形式となります。

配列データ形式については，「繰返し列を含む表の配列データ形式」を参照してください。

ff

FF 形式（Fixed field Fixed element）です。

入力データ中に実要素数（入力データファイル中に指定した有効な要素数）は指定しないで，要素データを常に一定個数指定します。

＜適用基準＞

FF 形式は，入力データ中に実要素数を指定しないで繰返し列へデータを格納したい場合に有効です。

＜規則＞

1. 入力データに指定する要素データの数は，表定義で指定した要素数と同じか，又はそれ以下にしてください。
2. FF 形式は，次に示す入力データ形式に対して指定できます。
   - DAT 形式
   - バイナリ形式
   - 固定長データ形式
3. 指定した要素データをデータベースに格納するかどうかは，次の条件を満たす場合にユティリティが決定します。
   - 指定されたデータがナル値かどうか
   - ナル値の取り扱いの指定内容

fv

FV 形式（Fixed field Variable element）です。

入力データ中に要素データとともに実要素数を指定します。ただし，入力データには，表定義，又は列構成情報ファイル，ナル値・関数情報ファイルで指定した要素数分の要素データを常に指定する必要があります。

＜適用基準＞

FV 形式は，固定長でデータを処理できるため，UAP でデータを作成，加工しやすくなりますが，ファイル容量が大きくなります。

＜規則＞

1. FV 形式は，次に示す入力データ形式に対して指定できます。
   ・バイナリ形式
   ・固定長データ形式
2. FV 形式は入力データを固定長データ型で指定する形式のため，可変長データ型（VARCHAR，MVARCHAR，NVARCHAR）の場合には，各要素データの先頭に有効データ長を指定します。さらに，有効データ以降は，表の定義長までの部分を任意のデータで埋め字を設定し，常に表の定義長分の長さで指定する必要があります。

vv

VV 形式（Variable field Variable element）です。

入力データ中に要素データとともに実要素数を指定します。要素データを実要素数分だけ指定します。

＜適用基準＞

VV 形式は，指定するデータは最小の長さでよいため，FV 形式よりファイル容量を小さくできます。ただし，実要素数分のデータしか用意しないため，FF 形式や FV 形式に比べて UAP でデータを加工しにくくなります。

＜規則＞

1. 可変長データ型の場合には，各要素データの先頭に指定した有効データ長分のデータだけ指定します。
2. VV 形式は，次に示す入力データ形式に対して指定できます。
   ・DAT 形式
   ・バイナリ形式

上記の配列データ形式は，指定する単位によって次のどれかのファイルに指定します。

指定する単位		指定するファイル	内容
表単位		制御情報ファイル	array 文の elmtype オペランドで指定します。
列単位	DAT 形式，又は固定長形式	列構成情報ファイル	elmtype オペランドで指定します。
	バイナリ形式	ナル値・関数情報ファイル	elmtype オペランドで指定します。

(b)　nullset=ナル値オプション

～《c》

配列データ形式に FF を指定した場合に，ある要素から指定要素の最後まですべてにナル値を指定した場合のナル値格納の扱いを指定します。

FF 形式以外の場合に指定しても無視します。

ナル値オプションには，次のどちらかを指定します。

c：該当する要素に何も格納しません。

e：該当する要素にナル値を格納します。

5.4.12　extdat 文（拡張 DAT 形式の情報の記述）

DAT 形式の入力データファイルで，次の拡張機能を使用する場合に指定します。

なお，囲み文字の変更，及び INTEGER，SMALLINT の列への小数点データの格納以外の拡張機能は，オペランドの指定有無に関係なく extdat 文を指定すれば使用できます。

- 次の文字の入力
  - 文字データ型，各国文字データ型，混在文字データ型へのナル文字（0x00）
  - BINARY 型へのナル文字（0x00）
  - 改行記号（0x0a）
  - 復帰記号（0x0d0a）
- 数データ型，日間隔データ型，及び時間隔データ型のデータ型に囲み文字（"）が指定されているデータの入力
- 数データ型，日付データ型，日間隔データ型，時刻データ型，及び時間隔データ型の列の入力データ部の，半角空白及びタブの削除
- 囲み文字の変更
- 囲み文字そのものを入力（囲み文字を 2 個連続させると，1 個のデータとして扱われます）
- INTEGER，SMALLINT の列への小数点データの格納

囲み文字とは，extdat 文の enclose オペランドに指定した文字のことをいいます。オペランドを省略した場合は，引用符（"）が囲み文字となります。

セパレータ文字の直前，直後のデータの前後を囲み文字で囲んだ場合，囲み文字は入力の対象にはなりません。ただし，囲み文字で囲まれた文字列内の囲み文字は，入力の対象となります。

なお，DAT 形式の場合は，引用符が囲み文字となります（囲み文字の変更はできません）。

＜適用基準＞

入力データファイルが DAT 形式で，上記の拡張機能を使用したい場合に指定します。

(1) 形式

```
extdat 〔enclose=囲み文字〕
       〔decin= {yes | no} 〕
```

(2) 説明

(a) enclose=囲み文字

～＜文字列＞((1 バイト))

各入力データの列値を囲んでいる囲み文字を指定します。

＜規則＞

1. セパレータ文字の直前，直後のデータの前後を囲み文字で囲んでいる場合，その囲み文字は入力の対象にはなりません。
2. 囲み文字と同じ値をデータとして扱う場合は，その囲み文字を 2 個連続で記述してください。例えば，アポストロフィ（'）が囲み文字の場合は，「''」と 2 個連続で記述します。
3. DAT 形式と拡張 DAT 形式では，囲み文字の扱いに違いがあるので注意してください。文字データの入力と囲み文字の扱いについては，「DAT 形式の場合」を参照してください。
4. 次の文字は囲み文字として指定できません。
   ・空白，タブ，アスタリスク（*）
   ・セパレータ文字と同じ文字
5. シフト JIS 漢字コードの場合，次の文字は入力データのコードと重複する可能性があるので，囲み文字には適していません。
   ・入力データファイルのデータにある文字
   ・日本語入力時に文字コードが重なる文字
   ([，¥,]，^，_，'，{，｜,}，￣)
6. 囲み文字は 1 バイトのため，2 バイトコードは指定できません。

(b) decin= {yes | no}

INTEGER 及び SMALLINT の列に，小数点があるデータを格納するかどうかを指定します。

yes：

小数点があるデータを格納します。

yes を指定した場合，最初の小数点までをデータとして扱い，それ以降は無視してフォーマットのチェックもしません。格納される値は，小数点以下を切り捨てた整数値となります。

no：

小数点があるデータを格納しません。

no を指定した場合，小数点があるデータは INTEGER 及び SMALLINT の列には格納しないで，エラーデータとします。

(3)　拡張機能の内容

(a)　文字データ型，各国文字データ型，混在文字データ型，及び BINARY 型へのナル文字（0x00），改行記号（0x0a），及び復帰記号（0x0d0a）の入力

文字データ型，各国文字データ型，混在文字データ型，及び BINARY 型の入力データが囲み文字で囲まれている場合，入力データ中にあるナル文字（0x00），改行記号（0x0a），及び復帰記号（0x0d0a）をデータとして扱います。囲み文字で囲まれていない場合，ナル文字，改行記号，及び復帰記号はデータの区切り，又は行の終端として扱われます。

例えば，"tokyo¥0shibuyaku"という CHAR(16)のデータがあった場合，extdat 文を指定すれば「tokyo¥0shibuyaku」が入力されますが，extdat 文を指定しないとこのデータは入力できません。

(b)　数データ型，日間隔データ型，及び時間隔データ型のデータ型に囲み文字（"）が指定されているデータの入力

数データ型，日間隔データ型，及び時間隔データ型の入力データに，囲み文字（"）が指定されていてもデータの入力ができます。なお，DAT 形式の場合，データの入力はできません。

(c)　数データ型，日付データ型，日間隔データ型，時刻データ型，時間隔データ型，及び時刻印データ型の列の入力データ部の，半角空白及びタブの削除

数データ型，日付データ型，日間隔データ型，時刻データ型，時間隔データ型，及び時刻印データ型の列の入力データ部の，半角空白及びタブを削除します。データがすべて半角空白又はタブの場合については，「データロード時のナル値，及び既定値」を参照してください。

(d)　囲み文字の変更

enclose オペランドに指定した文字を囲み文字として扱います。

(e)　囲み文字そのものを入力

囲み文字を 2 個連続して記述すると，1 個のデータとして扱います。なお，この規則は，マルチバイトコードも同様です。そのため，外字に割り当てたコードなどが囲み文字と一致した場合，不当に入力データエラーと扱われることがあります。この場合，囲み文字を変更する必要があります。

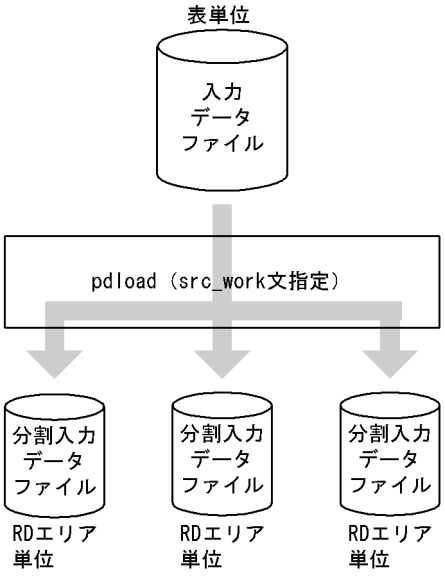
5.4.13　src_work 文（分割入力データファイルの出力先の記述）

RD エリア単位のデータロードを実行する場合に，表単位の入力データファイルから RD エリア単位の入力データファイル（分割入力データファイル）を作成するときに指定します。src_work 文は，データロード

時に指定するのではなく，データロードの前準備として分割入力データファイルを用意するために指定するものです。

分割入力データファイルの作成の概要を次の図に示します。

図 5–13　分割入力データファイルの作成の概要



＜適用基準＞

横分割表に RD エリア単位にデータロードを並列実行する場合で，かつ RD エリアごとに入力データファイルを準備していないときに指定します。

src_work 文を使用すると，ユーザが RD エリアごとの入力データファイルを用意する必要がなくなります。

＜前提条件＞

1. 入力データファイルは，DAT 形式ファイル（拡張 DAT 形式を含む），及び固定長データ形式ファイルを使用できます。分割入力データファイルも同じ形式で出力します。
2. 分割入力データファイルを作成する場合，表に対する INSERT 権限が必要となります。
3. 処理対象となる表は横分割表でなければなりません（分割方法，及び列構成には制限なし）。
4. 分割入力データファイルは，シングルボリュームでなければなりません。ファイルが既にある場合は上書きされ，ない場合は新規に作成されます。
5. 分割入力データファイルは，入力データファイルがあるホストと同一ホストに作成されます。
6. 分割入力データファイル作成時は，データをチェックして出力しますが，次のデータエラーについてはチェックされません。
   - クラスタキー順序不正
   - キー重複エラー
   - LOB 列の格納データの不正※1
   - 抽象データ型の格納データの不正※2

注※1

次のどちらかです。

・LOB 入力ファイルにアクセスできない。

・LOB 列のデータ長が LOB 列の定義長より長い。

注※2

次のどちらかです。

・プラグイン関数のデータチェックでエラーとなる不正データがある。

・BLOB 型の引数エラー（注※1 のエラーと同様）。

7. 分割入力データファイルは次の名称で作成されます。

【HiRDB/シングルサーバの場合】

「src_work 文に指定するディレクトリ名」＋「入力データファイル名」＋「RD エリア名」

【HiRDB/パラレルサーバの場合】

「src_work 文に指定するディレクトリ名」＋「入力データファイル名」＋「サーバ名」＋「RD エリア名」

上記の絶対パス名やファイル名の長さは，OS の制限を超えないように注意してください。

入力データファイルを複数指定した場合，「入力データファイル名」は先頭のファイル名となります。

「RD エリア名」は，表格納用 RD エリア名となります。

なお，RD エリア名に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

8. src_work 文指定の pdload の実行では，処理対象表にアクセスすることはありません（処理対象表に排他は掛かりません）。

(1) 形式

```
src_work 分割入力データファイルの出力先ディレクトリ名
```

(2) 説明

(a) 分割入力データファイルの出力先ディレクトリ名

分割入力データファイルを出力するディレクトリ名を絶対パス名で指定してます。

指定するディレクトリは，入力データファイルがあるサーバ又はホストでなければなりません。また，指定するディレクトリには，HiRDB 管理者に対してアクセス権限を与えておく必要があります。

(3) 注意事項

1. 分割入力データファイルの名称について

同じ表格納用 RD エリアに定義された複数の表に対して，同じ入力データファイル名で pdload を複数回実行すると，分割入力データファイルの名称が同じになり，ファイルの内容が保証できなくなりま

す。したがって，「src_work 文に指定するディレクトリ名」，又は「入力データファイル名」がユニークな名称となるようにしてください。

2. LOB 列がある表について

   分割入力データファイルの作成は，入力データファイルを分割するだけなので，LOB 入力ファイルにはアクセスしません。したがって，LOB 列がある表の場合，LOB 入力ファイルは pdload を実行するホストから参照できる場所に配置する必要があります。

3. 固定長データ形式の入力データファイルの DECIMAL 型データの扱い

   DECIMAL 型の符号正規化機能を使用している場合，分割入力データファイルに出力する DECIMAL 型の値は正規化します。

4. 該当 RD エリアに格納するデータがない場合，0 バイトの分割入力データファイルが作成されます。

5. 分割入力データファイルを出力する場合，自動採番機能は使用できません。

6. src_work 文指定の pdload の実行では，データの圧縮はしません。そのため，圧縮列の列データは圧縮しないで分割入力データファイルに出力します。

7. 表格納 RD エリア数が 1024 を超える表に対して実行された場合，1024RD エリア分の分割ファイルを作成する部分実行を複数回繰り返して，すべての RD エリアの分割入力データファイルを作成します。そのため，次のメッセージが n 回※出力されます。pdload 実行時に-m オプションを指定すると，これらのメッセージの出力数を削減できます。詳細は，「-m」を参照してください。

   - KFPL00700-I
   - KFPL00701-I
   - KFPL00702-I
   - KFPL00705-I
   - KFPL00709-I

   また，source 制御文に error オペランドを指定していない場合，エラー情報ファイルが n 個※作成されます。エラー情報ファイルを複数作成させたくない場合は，source 制御文に error オペランドを指定してください。詳細は，「error ＝エラー情報ファイル名」を参照してください。

   注※

   n ＝表格納 RD エリア数÷1024

   なお，小数点以下は整数値に切り上げてください。

(4) 使用例

(a) 分割入力データファイルを作成して RD エリア単位のデータロードをする例

横分割表（T1）に RD エリア単位にデータロードを並列実行します。このとき，入力データファイル（c:¥hirdb¥pdload¥inputfile）は表単位であるため，RD エリア単位に実行するための分割入力データファイルを作成します。その後，RD エリア単位にデータロードを並列実行します。

1.表定義

```
CREATE FIX TABLE T1 (C1 DEC,
                     C2 CHAR(10))
                    IN ((RDAREA1) C1 > 1000,(RDAREA2) C1 < -1000,(RDAREA3));
```

2.分割入力データファイルの作成

- **pdload のコマンド**

```
pdload T1 c:¥hirdb¥pdload¥control_file
```

[説明]

T1：処理対象表の名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

- **制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容**

```
source FES:c:¥hirdb¥pdload¥inputfile error=c:¥hirdb¥pdload¥errfile    [1]
src_work c:¥hirdb¥divwork    [2]
```

[説明]

1. 入力データファイル，及びエラー情報ファイルの名称を指定します。
2. 分割入力データファイルを作成するディレクトリ名を指定します。

3.RD エリア単位のデータロードの並列実行

「2.分割入力データファイルの作成」の結果，次の三つの分割入力データファイルが作成されます。

- c:¥hirdb¥divwork¥inputfile_BES1_RDAREA1
- c:¥hirdb¥divwork¥inputfile_BES2_RDAREA2
- c:¥hirdb¥divwork¥inputfile_BES3_RDAREA3

上記のファイルを入力データファイルにして，RD エリア単位にデータロードを並列実行します。

(b)　全列固定長のバイナリ形式の入力データファイルを使用する例

分割入力データファイルを作成する場合，バイナリ形式の入力データファイルは使用できません。ただし，すべての列が固定長のデータ型※で構成される表であれば，固定長データ形式として扱うことができます。

注※

次のデータ型です。

INTEGER，SMALLINT，DECIMAL，FLOAT，SMALLFLT，DATE，TIME，INTERVAL YEAR TO DAY，INTERVAL HOUR TO SECOND，CHAR，NCHAR，MCHAR，TIMESTAMP

1.表定義

```
CREATE FIX TABLE T2 (C1 DEC(7),
                     C2 CHAR(10)) FIX HASH HSAH6 BY C1
                    IN (RDAREA1,RDAREA2);
```

2.分割入力データファイルの作成

- **pdload のコマンド**

```
pdload -a -c c:¥hirdb¥pdload¥column_file T2 c:¥hirdb¥pdload¥control_file
```

[説明]

-a：入力データファイルが固定長データ形式の場合の指定

T2：処理対象表の名称

c:¥hirdb¥pdload¥column_file：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

- **列構成情報ファイル（c:¥hirdb¥pdload¥column_file）の内容**

```
C1,type=dec(7)
C2,type=char(10)
```

[説明]

表の構成列 C1，C2 の並びと，データ型を指定します。

- **制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容**

```
source FES:c:¥hirdb¥pdload¥inputfile error=c:¥hirdb¥pdload¥errfile    [1]
src_work c:¥hirdb¥divwork    [2]
```

[説明]

1. 入力データファイル，及びエラー情報ファイルの名称を指定します。
2. 分割入力データファイルを作成するディレクトリ名を指定します。

3.RD エリア単位のデータロードの並列実行

「2.分割入力データファイルの作成」の結果，次の二つの分割入力データファイルが作成されます。

- c:¥hirdb¥divwork¥inputfile_BES1_RDAREA1
- c:¥hirdb¥divwork¥inputfile_BES2_RDAREA2

上記のファイルを入力データファイルにして，RD エリア単位にデータロードを並列実行します。

5.4.14　constraint 文（検査保留状態に関する記述）

検査保留状態に関する設定をします。

検査制約又は参照制約を定義した表にデータロードする場合，pdload ではデータの整合性チェックをしないため，制約の整合性が保証できないデータを格納する可能性があります。このとき，pdconstck で整合性を検証するまで，表へのデータ操作を制限させるため，データロード対象表やその表に関係する参照表を検査保留状態に設定します。

なお，ユーザ自身が制御文を指定して，表の検査保留状態を変更しないこともできます。検査保留状態については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

＜適用基準＞

次のどちらかに該当する場合，constraint 文を指定します。

- 検査制約又は参照制約を定義した表にデータロードする場合，その表の検査保留状態を変更（設定）しないとき
- 被参照表にデータロードする場合，その表に関係する参照表の検査保留状態を変更（設定）しないとき

＜規則＞

1. この制御文は，システム定義の pd_check_pending オペランドの値が USE の場合だけ有効となります。次のどれかに該当する場合，constraint 文の指定は無視され，表の検査保留状態を変更しません。
   ・システム定義の pd_check_pending オペランドの値が NOUSE の場合
   ・表に制約が定義されていない場合
   ・分割入力データファイルを作成する場合（src_work 文を指定する場合）

(1) 形式

```
constraint 〔pending=no〕
           〔ref_pending=no〕
```

(2) 説明

(a) pending=no

参照表又は検査制約表にデータロードする場合，その表の検査保留状態を変更しないときに指定します。このオペランドを省略した場合の仮定値は，「検査保留状態の設定範囲」を参照してください。

＜適用基準＞

次のどちらかに該当する場合，pending=no を指定します。

- 参照表又は検査制約を定義した表にデータロードする場合，列に格納するデータがユーザによる事前チェックによって整合性が保たれている保証があるとき
- 参照表又は検査制約を定義した表に対して，インデクスの再作成をするためにデータ件数 0 件でデータロードをする場合

(b) ref_pending=no

被参照表にデータロードする場合，その表に関係する参照表の検査保留状態を変更しないときに指定します。このオペランドを省略した場合の仮定値は，「検査保留状態の設定範囲」を参照してください。

＜適用基準＞

次のどちらかに該当する場合，ref_pending=no を指定します。

- 被参照表に対して作成モードでデータロードする場合，被参照表のデータと参照表のデータがユーザの事前チェックによって整合性が保たれている保証があるとき
- 被参照表に対して，インデクスの再作成をするために追加モードでデータ件数 0 件のデータロードをする場合

(3)　注意事項

1. 被参照表へデータロードする場合，その表に関係する参照表を検査保留状態に設定するときの注意事項については，「コマンド実行時の RD エリアの状態」を参照してください。
2. 検査保留状態を設定する場合，関連する資源に対して排他が掛かります。また，検査保留状態の設定が完了したとき，掛かっていた排他が解除されます。排他制御については，「コマンド実行時の排他制御モード」を参照してください。

(4)　検査保留状態の設定範囲

検査制約の検査保留状態の設定範囲と参照制約の検査保留状態の設定範囲を次の表に示します。

表 5–11　検査制約の検査保留状態の設定範囲

データロード実行条件		検査制約の検査保留状態		
システム定義の pd_check_pending オペランドの値	実行単位	ディクショナリ表		RD エリア中の表情報
		SQL_TABLES 表 CHECK_PEND2 列	SQL_CHECKS 表 CHECK_PEND2 列	検査制約の状態
USE	表	●	●	◎
	RD エリア	●	●	○
NOUSE	表	×	×	×
	RD エリア	×	×	×

（凡例）

◎：表を格納するすべての RD エリア中の表情報（検査制約の状態）を設定します。

●：検査保留状態を設定します。

○：データロード対象の RD エリア中の表情報（検査制約の状態）を設定します。

×：検査保留状態を変更しません（現在の状態を維持します）。

表 5–12　参照制約の検査保留状態の設定範囲

データロード実行条件			データロード対象表					
システム定義の pd_check_pending オペランドの値	-d オプションの指定	実行単位	被参照表			参照表		
			被参照表に関連する参照表の検査保留状態			参照表の検査保留状態		
			ディクショナリ表		RD エリア中の表情報	ディクショナリ表		RD エリア中の表情報
			SQL_TABLES 表 CHECK_PEND 列	SQL_REFERENTIAL_CONSTRAINTS 表 CHECK_PEND 列	参照制約の状態	SQL_TABLES 表 CHECK_PEND 列	SQL_REFERENTIAL_CONSTRAINTS 表 CHECK_PEND 列	参照制約の状態
USE	あり	表	●	●	◎	●	●	◎
		RD エリア	●	●	◎	●	●	○
	なし	表	×	×	×	●	●	◎
		RD エリア	×	×	×	●	●	○
NOUSE	あり	表	×	×	×	×	×	×
		RD エリア	×	×	×	×	×	×
	なし	表	×	×	×	×	×	×
		RD エリア	×	×	×	×	×	×

（凡例）

◎：表を格納するすべての RD エリア中の表情報（検査制約の状態）を設定します。

●：検査保留状態を設定します。

○：データロード対象の RD エリア中の表情報（検査制約の状態）を設定します。

×：検査保留状態を変更しません（現在の状態を維持します）。

(5)　検査保留状態の設定可否の例

次に示す表 T1～T5 について，constraint 文の pending 及び ref_pending オペランドを指定した場合の，表の検査保留状態の設定可否について説明します。

（表の説明）

T1：主キーを持つ表（T2 に対する被参照表）

T2：T1 の主キーを参照する外部キーを持ち，さらに自身が主キーを持つ表（T1 への参照表であり，かつ T3 に対する被参照表である表）

T3：T2 の主キーを参照する外部キーを持つ表（T2 への参照表）

T4：検査制約を定義した表

T5：制約定義がない表

表 T1～T5 に対してデータロードを実行した場合の表の検査保留状態を次の表に示します。

表 5–13　データロードを実行した場合の表の検査保留状態

データロード対象表と制約定義有無				constraint 文のオペランドの指定有無		表の検査保留状態				
表名	参照制約		検査制約	pending	ref_pending	T1	T2	T3	T4	T5
	被参照表	参照表								
T1	○	×	×	なし	なし	–	P	–	–	–
					あり	–	N	–	–	–
				あり	なし	–	P	–	–	–
					あり	–	N	–	–	–
T2	○	○	×	なし	なし	–	P	P	–	–
					あり	–	P	N	–	–
				あり	なし	–	N	P	–	–
					あり	–	N	N	–	–
T3	×	○	×	なし	なし	–	–	P	–	–
					あり	–	–	P	–	–
				あり	なし	–	–	N	–	–
					あり	–	–	N	–	–
T4	×	×	○	なし	なし	–	–	–	P	–
					あり	–	–	–	P	–
				あり	なし	–	–	–	N	–
					あり	–	–	–	N	–
T5	×	×	×	なし	なし	–	–	–	–	–
					あり	–	–	–	–	–
				あり	なし	–	–	–	–	–
					あり	–	–	–	–	–

（凡例）

○：定義あり。

×：定義なし。

P：検査保留状態を設定します。

N：検査保留状態を変更しません（現在の状態を維持します）。

−：該当しません（pending 及び ref_pending オペランドの指定を無視します）。

(6)　検査保留状態を設定するタイミング

データロード対象表やその表に関係する参照表の検査保留状態を設定するタイミングは，データロードのモードによって異なります。データロードのモードと検査保留状態に設定するタイミングの関係を次の表に示します。

表 5–14　検査保留状態に設定するタイミング

データロードのモード	検査保留状態を設定するタイミング
作成モード	データ削除処理を開始する前
追加モード	データロードを開始する前

なお，表の検査保留状態を設定した後，エラーの発生などによって pdload がロールバックした場合でも，表の検査保留状態の設定は完了（コミットが完了）しているため，表は検査保留状態となります。

5.4.15　option 文（データ処理に関する情報の記述）

データロードするときの，データ処理に関するオプション機能について指定します。

＜適用基準＞

次のどれかのオプション機能を使用したい場合に指定します。

- 空白変換レベルの指定（spacelvl オペランド）
- データロード時の，空き領域比率の変更（tblfree オペランド）
- インデクス作成時の，空き領域比率の変更（idxfree オペランド）
- 同期点指定のデータロード（job オペランド）
- データの切り捨てが発生した場合の警告メッセージ出力有無（cutdtmsg オペランド）
- データロードする表に対する NOWAIT 検索の実行（nowait オペランド）
- pdrorg 出力バイナリ形式の入力データファイルに LOB 列又は LOB パラメタがある場合の指定（bloblimit オペランド）
- pdload の実行時間の監視（exectime オペランド）
- DEFAULT 句を指定している表へのデータロードで，入力データがナル値の場合のデータの格納方法（null_string オペランド）
- 入力データエラーを検知した場合，それまでの格納処理を無効にする指定（dataerr オペランド）

- DAT 形式の入力データファイルの場合，入力データ長が列定義長より長いときに入力データエラーとする指定（lengover オペランド）
- エラーデータ情報の抑止（divermsg オペランド）
- エンディアンが異なるプラットフォーム間で pdrorg 出力バイナリ形式ファイルを使用してデータを移行する場合の指定（srcendian オペランド）
- 入力データファイルが固定長データ形式の場合で，数値データ型の列に文字データを格納するときの格納方法（allspace オペランド）
- XML 文書に含まれる空白類の取り扱い方法（whitespace オペランド）
- 自動採番機能を使用したデータロード（seq_range，file_buff_size オペランド）
- UTF-8 の環境で定義した表に対して，UTF-16 の入力データファイルでデータロードする場合の指定（charset オペランド）

(1) 形式

```
option  〔spacelvl= {0 | 1 | 3} 〕
        〔tblfree= {未使用領域比率 | （〔未使用領域比率〕, セグメント内空きページ比率）} 〕
        〔idxfree=未使用領域比率〕
        〔job= {ジョブ名〔, 同期点行数〕〔, CLR〕 |, , CLR} 〕
        〔cutdtmsg= {on | off} 〕
        〔nowait= {yes | no} 〕
        〔bloblimit=領域確保サイズ〕
        〔exectime=pdloadの実行監視時間〕
        〔null_string= {null | default} 〕
        〔dataerr=rollback〕
        〔lengover=err〕
        〔divermsg=off〕
        〔srcendian= {big | little} 〕
        〔allspace=zero〕
        〔whitespace= {preserve | strip} 〕
        〔seq_range= {buff | 順序番号取得単位 | all} 〕
        〔file_buff_size=バッファ長〕
        〔charset= {utf-16be | utf-16le} 〕
```

(2) 説明

(a) spacelvl= {0 | 1 | 3}

入力データに対して，空白変換をするかどうかを指定します。

0，1，及び 3 のことを，空白変換レベルといいます。

spacelvl オペランドを省略した場合，システム共通定義の pd_space_level オペランドの指定値に従って空白変換をします。

0：

入力データの空白変換をしません。

1，及び 3：

入力データの空白変換をします。1 と 3 のどちらを指定しても意味は同じです。

空白変換をする場合，データロード対象の表の列が各国文字列型，又は混在文字列型のとき，対応する入力データを次のように変換してデータベースに格納します。

- 各国文字列型の列の場合

  入力データ中の連続する半角空白 2 バイトを，全角空白 1 文字に変換します。ただし，先頭から 2 バイト単位で変換します。

- 混在文字列型の列の場合

  入力データ中の全角空白 1 文字を，半角空白 2 バイトに変換します。

  文字コードが utf-8 の場合，全角空白 1 文字（3 バイト）を半角空白 2 バイトに変換します。MCHAR の場合は，データ長が短くなった分，定義長まで後ろから半角空白が埋められます。MVARCHAR の場合は，データ長が短くなったままとなります。

抽象データ型の列に対してコンストラクタ関数を介してデータを格納する場合，対応するコンストラクタ関数の引数のデータ型が各国文字列型，又は混在文字列型のときは，入力データを引数としてコンストラクタ関数に渡す前に空白変換をします。

＜適用基準＞

表データ中の空白文字を統一したい場合に指定します。

表データ中の空白文字を統一する方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜注意事項＞

spacelvl オペランドを指定して空白変換をする場合，及びシステム共通定義の指定によって空白変換をする場合は，次のことに注意してください。

1. クラスタキー順のチェックはデータ変換後の値でするため，変換前のデータがクラスタキー順でも順序エラーとなることがあります。この場合，-x オプションを指定して，クラスタキー順のチェックをしないようにしてください。クラスタキー順に格納したい場合は，データロード後に再編成をしてください。
2. ユニークキーインデクス又は主キーインデクスを構成する列が空白変換対象の場合，そのインデクスに対して重複キーが発生することがあります。この場合，次のように対処しておく必要があります。

   ・ユニークキーインデクスの場合

   事前にインデクスのユニーク指定を外すか，又は入力データ中の該当するデータを変更してください。

   ・主キーインデクスの場合

   入力データ中の該当するデータを変更してください。
3. 列構成情報ファイル，及びナル値・関数情報ファイルで指定するナル値の比較値については，空白変換はしません。この場合，ナル値の比較値は，空白変換前の入力データと比較します。
4. 入力データの長さのチェックは，空白変換前の入力データと比較します。

(b)　tblfree=｛未使用領域比率｜（〔未使用領域比率〕，セグメント内空きページ比率）｝

データロード時に，CREATE TABLE で指定した空き領域比率（PCTFREE の指定値）を変更してデータを格納したい場合に指定します。

未使用領域比率

0～99 の値を指定できます。

セグメント内空きページ比率

0～50 の値を指定できます。

＜適用基準＞

表定義時に空き領域比率に 0 以外を指定した表の場合，データロード中に定義時の空き領域比率が適用されたことによって，データ格納時に RD エリアの容量不足が発生することがあります。このような場合に指定すると，暫定的に RD エリアを拡張しないで，データロードを完了できます。

＜注意事項＞

このオペランドを指定した場合，-y オプションを指定して空き領域へデータ格納するよりも，次の点で効率が良くなります。

- 空き領域にデータを格納する場合も一括入出力が有効となります。
- -l オプションに p を指定できます。

(c)　idxfree=未使用領域比率

インデクス一括作成モードでインデクスを作成する場合に，CREATE INDEX で指定した空き領域比率（PCTFREE の指定値）を変更してインデクスを作成したいときに指定します。

指定できる値は，0～99 です。

＜適用基準＞

インデクス定義時に空き領域比率に 0 以外を指定したインデクスの場合，インデクス一括作成処理中に定義時の空き領域比率が適用されたことによって，インデクス格納時に RD エリアの容量不足が発生することがあります。このような場合に指定すると，暫定的に RD エリアを拡張しないで，インデクス一括作成処理を完了できます。

(d)　job=｛ジョブ名〔，同期点行数〕〔，CLR〕｜，，CLR｝

同期点指定のデータロードをする場合に指定します。

同期点指定のデータロードとは，任意の件数のデータを格納するごとに，トランザクションを決着するデータロードのことをいいます。ログ取得方式がログ取得モード（-l a）又は更新前ログ取得モード（-l p）の場合，データの格納途中で障害が発生したときでも，データを最初から格納し直す必要がなく，短時間でデータベースを回復できます。この場合の対処方法については，マニュアル「HiRDB システム導入・設計ガイド」の「同期点指定のデータロード実行中にユティリティが異常終了したときの対処方法」を参照してください。

なお，監査証跡表にデータロードする場合，自動的に監査証跡ファイルの世代ごとに COMMIT を行うため，件数単位に COMMIT を指示するこのオペランドは指定できません。指定すると，制御文エラーとなります。

ジョブ名　～〈英数字〉((1～3))

同期点指定のデータロードを実行する場合のジョブ名を指定します。

ここで指定したジョブ名は，データロード実行中にユティリティが異常終了した場合，後から再実行するときの名称となります。また，このジョブ名はほかの pdload のジョブ名と重複すると，再実行時に誤動作する原因になります。したがって，ジョブ名はユニークになる名称を指定してください。

同期点行数　～((1～1000))《100》

何万件格納するごとにトランザクションを決着するか，データ件数を指定します。例えば，100 を指定すると，100 万件ごとに同期点が取得されます。

ここで小さな値を指定した場合，少ない件数でトランザクションが決着するので，異常終了時のロールバック処理は短くなります。しかし，トランザクションの生成，決着，及びプロセスの再起動が頻繁に発生するため，オーバヘッドが大きくなります。一方，大きな値を指定した場合，処理性能の劣化は少なくなりますが，障害発生時の回復時間は長くなります。

ジョブ名を指定しないで同期点行数だけを指定した場合，エラーとなります。

CLR

pdload を再実行する必要がない場合，又は現在データベース中にある同期点情報をクリアして pdload を再実行する場合に指定します。

ユティリティが異常終了した場合，同期点情報がデータベースに残ります。この情報が残ったままだと，次に実行したユティリティが正常に動作しない場合があります。

CLR を指定した場合，ほかのユーザやユティリティが設定した同期点情報でもクリアするため，注意して使用してください。

データベース中に同期点情報が設定されているかどうかは，データベース状態解析ユティリティ（pddbst）の表単位の状態解析で確認できます。

＜適用基準＞

大量のデータロードをする場合など，格納時間が長くなるときに有効となります。ただし，同期点処理が実行される分，処理性能は悪くなります。また，データページについては，同期点のたびに新たなページから格納を開始するため，データ格納に必要なページ数が通常のデータロードよりも多くなります。

＜ job オペランドの指定例＞

- 100 万件単位に同期点指定のデータロードをする場合

  job=JOB,100　又は　job=JOB

- 現在設定されている同期点情報をクリアして，新たに 100 万件単位の同期点指定のデータロードをする場合

  job=JOB,100,CLR　又は　job=JOB,,CLR

- 現在設定されている同期点情報をクリアして通常のデータロードをする場合

job=,,CLR

＜注意事項＞

1. 同期点情報は，ジョブ名で管理されます。例えば，同期点指定のデータロードを実行していた pdload(A)が異常終了した場合，その表に対して同じジョブ名を指定した pdload(B)を実行すると，pdload(A)が残した同期点情報を引き継いで pdload(B)が動作してしまいます。このため，ジョブ名が重複しないように管理する必要があります。
2. 同期点指定のデータロードが異常終了した場合，そのときのジョブ名の同期点指定のデータロードを再実行するまでは，その表に対して通常のデータロード（同期点指定でないデータロード）も実行できません。
3. 同期点情報は RD エリア単位に管理されます。このため，横分割した表の場合，次の条件では表全体の同期点情報が不一致となります。

   ・一部の RD エリアだけ再初期化された場合

   ・RD エリア単位の同期点指定のデータロードでエラーが発生した場合

   各 RD エリアが管理している同期点情報は，ジョブ名が同じ場合でも行数が不一致のときは，再実行できません。しかし，同期点情報を保持している RD エリアと保持していない RD エリアが混在する場合は，保持されている同期点情報で再実行できます。同期点情報のジョブ名が同じで行数が異なる場合の動作を次に示します。

保持している同期点情報（同期点行数）			表単位の同期点指定のデータロードをした場合の入力開始行
RD エリア 1	RD エリア 2	RD エリア 3	
0	0	0	0 件目
0	n	0	n 件目
n	0	m	実行できません。
n	m	p	実行できません。

（凡例）

0：同期点情報を保持していない状態

n，m，p：保持している同期点行数

4. 同期点指定のデータロードが異常終了した場合，再実行時のインデクス作成方法（-i オプション）は変更できません。
5. 同期点指定のデータロードが異常終了してから再実行するまでに，RD エリアの再初期化，及び PURGE TABLE 文を実行すると，設定されていた同期点情報が削除されます。
6. 同期点指定のデータロードで，プラグインインデクスの一括作成処理中に異常終了した場合，「プラグインインデクス一括作成部分回復機能」を使用できるプラグインのプラグインインデクスであれば，エラーとなったプラグインインデクスを pdrorg で作成した後に，pdload を再実行してください。pdrorg を実行しないで pdload を再実行した場合，プラグインインデクスが全件作り直しになります。

7. 同期点指定のデータロードで制御できるデータ件数の上限は 4,294,960,000 件です。これ以上のデータ件数のデータロードをする場合は，入力データファイルを複数に分けて，複数回 pdload を実行してください。

(e) cutdtmsg=｛on ｜ off｝

DAT 形式の入力データファイルのデータロード時に，次のデータ型の列に対して文字データを格納する場合，入力データのデータ長が表の列定義長より長いときは，表の列定義長以上のデータ部分は切り捨てられます。

- 文字データ型（CHAR，VARCHAR）
- 各国文字データ型（NCHAR，NVARCHAR）
- 混在文字データ型（MCHAR，MVARCHAR）
- BINARY 型

この切り捨てが発生したときに，エラー情報ファイルに警告メッセージ（KFPL31090-W）を出力するかどうかを指定します。

on：警告メッセージを出力します。

off：警告メッセージを出力しません。

＜規則＞

1. -e オプションを指定している場合は，cutdtmsg=off が仮定されます。
2. DAT 形式以外の入力データファイルに対しては，このオペランドを指定しても無視されます。
3. cutdtmsg=on を指定して切り捨てが発生すると，すべてのデータが格納されても，pdload のリターンコードは 4 となります。
4. 警告メッセージは行数ではなく，データの切り捨てを検知したデータ件数分（列数，要素数分）出力されます。したがって，大量に出力される可能性があるので，cutdtmsg=on を指定する場合はディスク容量に注意してください。
5. エラー情報ファイルに警告メッセージを出力しても，エラーデータファイルには該当する入力データを出力しません。

(f) nowait=｛yes ｜ no｝

データロード中の表に対して，NOWAIT 検索をするかどうかを指定します。NOWAIT 検索とは，排他オプションに NOWAIT を指定して検索する SQL のことをいいます。

yes：NOWAIT 検索を実行する場合に指定します。

no：NOWAIT 検索を実行しない場合に指定します。

＜ nowait=yes の場合の規則＞

1. インデクス一括作成モード（-i c）は指定できません。インデクス更新モード（-i s）を指定してください。
2. ローカルバッファ（-n オプション）は使用できません。
3. RD エリア単位のデータロードはできません。
4. 作成モード（-d オプション指定）の場合，データ削除中はその表に対して NOWAIT 検索はできません（排他待ちとなります）。データ格納中は，NOWAIT 検索ができます。
5. プラグインが提供する抽象データ型列がある表の場合，プラグインが無排他検索機能を提供しているかどうかで実行可否が決まります。無排他検索機能を提供しているかどうかについては，各プラグインマニュアルを参照してください。
6. LOB 列がある表の LOB 列にだけデータロードをする場合，このオペランドの指定に関係なく，LOB 列構成基表は参照でき，LOB 列は参照できなくなります。LOB 列を参照したい場合には，LOB 列構成基表と LOB 列を同時にデータロードしてください。
7. pdload が異常終了してロールバックしても，データロード処理でいったん使用中となったセグメントは解放されません。解放されないセグメントは，pdload を再実行するときに再利用しますが，UAP の INSERT 文，UPDATE 文では再利用しません。このため，pdload 再実行前に，UAP の INSERT 文，UPDATE 文を実行すると，該当する RD エリアの未使用セグメントがなくなるまで，pdload 及び UAP で再利用できないセグメントが残ってしまいます。このセグメントを解放するには，pdrorg で表の再編成を実行する必要があります。この現象を回避するためには，RD エリアを参照可能閉塞状態にして pdload を実行します（pdload が異常終了しても，UAP からその表に対してアクセスできません）。
8. 共用表に対して yes を指定しても無視されます（no が仮定されます）。

(g)　bloblimit=領域確保サイズ

～＜符号なし整数＞((1～2097152))

pdrorg 出力バイナリ形式の入力データファイルで，次のデータ変換をしてデータロードする場合，データをメモリ上に保持しなければならないときがあります。

- 抽象データ型の BLOB パラメタのデータ→BLOB 型の列
- BLOB データ→BLOB パラメタがある抽象データ型の列

このとき，データを保持する領域の大きさをキロバイト単位で指定します。ただし，BLOB パラメタ又は BLOB 型の定義長と，このオペランドの指定値を比べて，小さい方の大きさで領域を確保します。

なお，pdrorg 用アンロードファイルを使用して BLOB 型の列を持つ表へデータロードする場合は，BLOB 型の定義長ではなく，実際データベースに格納されている BLOB データの実長に合わせた領域を確保するときにこのオペランドを指定します。BLOB データの最大長はアンロードした表から SQL で求めることができます。バッファは複数の BLOB 型の列で共用します。実際に確保する領域の大きさを次に示します。

MIN（表の全BLOB型の定義長の合計，bloblimitオペランドの指定値）

確保した領域より入力した BLOB データの合計長が長くなった場合は処理を打ち切ります。

＜適用基準＞

次のどちらかの条件に該当する場合，このオペランドを指定してください。

- 抽象データ型の BLOB パラメタのデータ→BLOB 型の列の場合，その BLOB パラメタのデータよりも後に，BLOB パラメタがある抽象データ型列へ格納するデータがあるとき
- BLOB データ→BLOB パラメタがある抽象データ型の列の場合，その BLOB データよりも前に，BLOB 型の列へ格納するデータがあるとき

＜規則＞

1. このオペランドは，pdload 用アンロードファイルを使用するときだけ有効となります。
2. 指定した大きさを超えるデータがある場合，エラーとなります。

(h)　exectime=pdload の実行監視時間

～＜符号なし整数＞((0～35791394))

pdload の実行時間を監視する場合に，その監視時間を分単位で指定します。0 を指定した場合は，実行時間の監視はしません。指定した監視時間内に pdload の処理が終了しなかった場合，pdload 制御プロセスを強制終了し，無応答原因の調査用に障害情報を取得します。

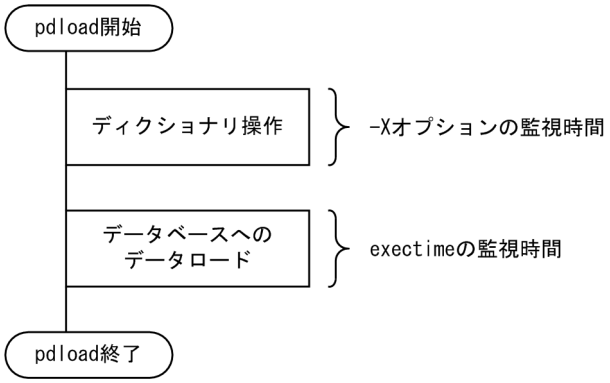
＜ pdload の実行時間の監視＞

pdload 実行時，次の 2 つの処理に対して実行時間を監視できます。

- データロードの前準備で，ディクショナリを操作する時間（-X オプション）
- 実際にデータを読み込み，データベースにデータロードする時間（option 文の exectime オペランド）

-X オプション，及び option 文の exectime で設定する監視時間の関係については，次の図を参照してください。

図 5–14　pdload 実行時の監視時間範囲の関係



このオペランドを省略した場合は，次の優先順位でシステム定義の指定値に従います。

1. pd_utl_exec_time
2. pd_cmd_exec_time

＜適用基準＞

夜間バッチなどで監視時間を設定しておけば，pdload 実行中に何らかの障害（通信障害（一時的な障害を含む），ディスク障害など）が発生し，pdload が無応答状態になった場合は異常終了するため，早期に異変を検知できて，早期に回復処理に着手できます。

＜指定値の目安＞

このオペランドは無応答障害検知を目的とするもので，長大トランザクション実行時間を監視するものではありません。したがって，指定値は，該当する表の処理時間に余裕をもたせた値にする必要があります。例えば，7，8 分程度で終了する pdload の実行時間を監視する場合は，exectime=10 と指定しないで，exectime=20 と指定するようにしてください。また，追加モードのデータロードをする場合，入力データファイルのデータ件数が一定でも，母体のデータ件数によって実行時間は変化するため，更に余裕のある値を指定してください（データ量が単調に増加する表では，適時指定値を見直す必要があります）。

(i)　null_string= {null ｜ default}

DEFAULT 句を指定している表に対してデータロードする場合，入力データがナル値（"*"又は記述しない場合）のときに，DEFAULT 句の既定値を格納するか，又はナル値を格納するかを指定します。

このオペランドは，入力データファイルが DAT 形式の場合に指定できます。バイナリ形式，固定長データ形式，及び pdrorg 生成バイナリ形式の場合に指定するとエラーになります。

null：ナル値を格納します。

default：DEFAULT 句の既定値を格納します。

(j)　dataerr=rollback

入力データエラー（論理エラー）を検知した場合，データ格納処理を無効（ロールバック）にするときに指定します。

-e オプションと dataerr オペランドの関係と dataerr オペランドとほかのオプション，制御文を同時に指定する場合の留意事項を次の表に示します。

表 5–15　-e オプションと dataerr オペランドの関係

-e オプション	option 文の dataerr オペランド	入力データエラーを検知した場合の pdload の動作	トランザクション決着方法	pdload のリターンコード
指定なし	指定なし	エラーデータの DB 格納処理をスキップして，処理を続行します。	コミット	4
	指定あり	エラーデータを検知した時点で処理を打ち切ります。	ロールバック	8

-e オプション	option 文の dataerr オペランド	入力データエラーを検知した場合の pdload の動作	トランザクション決着方法	pdload のリターンコード
指定あり	指定なし	エラーデータを検知した時点で処理を打ち切ります。	コミット	4
	指定あり	エラーデータを検知した時点で処理を打ち切ります。	ロールバック	8

表 5–16　dataerr オペランドとほかのオプション，制御文を同時に指定する場合の留意事項

オプション及び制御文	dataerr オペランドと同時に指定する場合の留意事項
-d	同時に指定すると制御文エラーとなります。
-l	a 又は p を指定してください。
-i	次の条件をすべて満たす場合，ロールバック後はインデクス未完状態となるため，同時に指定すると制御文エラーとなります。 • RD エリア単位のデータロード • 非分割キーインデクスを格納した RD エリア数が表格納 RD エリア数と一致しない • -i オプションが s 以外 上記の条件をすべて満たさないように，pdload のオプション，制御文の変更，又はインデクス定義の変更を検討してください。
	キー重複エラーの可能性があるデータロードの場合，s を指定してください。s 以外を指定した場合，キー重複エラーを検知しても，表に格納したデータはロールバックされません。
-k	f，v，又は c を指定した場合，ロールバック後に不要な LOB 中間ファイルが残ります。pdload の再実行前には，この不要な LOB 中間ファイルを削除してください。
	f，v，又は c を指定した場合，LOB 列へのデータロード中にエラーを検知したときは，dataerr オペランドの指定に関係なく LOB 列構成基表はコミット，LOB 列はロールバックとなります。この場合，LOB データを修正して再度データロードしてください。
-e	オプションの指定に関係なく，エラーデータを検知した時点で処理を打ち切り，ロールバックします。
上記以外のオプション	特にありません。
option 文の job オペランド	同時に指定すると制御文エラーとなります。
source 文の RD エリア名	次の条件をすべて満たす場合，ロールバック後はインデクス未完状態となるため，同時に指定すると制御文エラーとなります。 • RD エリア単位のデータロード • 非分割キーインデクスを格納した RD エリア数が表格納 RD エリア数と一致しない • -i オプションが s 以外 上記の条件をすべて満たさないように，pdload のオプション，制御文の変更，又はインデクス定義の変更を検討してください。
source 文の error オペランド	検知したエラーの情報を出力した後，ロールバックします。
source 文の errdata オペランド	同時に指定すると制御文エラーとなります。
上記以外の制御文	特にありません。

(k)　lengover=err

DAT 形式（拡張 DAT 形式を含む）の入力データファイルの，次のデータ型の列に格納する入力データ長が列定義長より長い場合，入力データエラーとして扱うときに指定します。

- CHAR
- VARCHAR
- NCHAR
- NVARCHAR
- MCHAR
- MVARCHAR
- BINARY

入力データ長，列定義長，及び lengover オペランドの関係と lengover オペランドとほかのオプション，制御文を同時に指定する場合の留意事項を次の表に示します。

表 5–17　入力データ長，列定義長，及び lengover オペランドの関係

入力データ長と列定義長の関係	lengover 指定なし	lengover 指定あり
m=n	入力データをそのまま格納します。	
m<n	入力データの m+1 以降のデータは切り捨てて格納します。	入力データエラーとして扱います。
m>n	CHAR，NCHAR，及び MCHAR の場合： 入力データの n+1 以降に空白を埋めて格納します。 VARCHAR，NVARCHAR，MVARCHAR，及び BINARY の場合： 入力データをそのまま格納します。	

表 5–18　lengover オペランドとほかのオプション，制御文を同時に指定する場合の留意事項

オプション及び制御文	lengover オペランドと同時に指定する場合の留意事項
-a，-b	lengover オペランドは DAT 形式の場合に有効なオペランドのため，同時に指定すると制御文エラーとなります。
上記以外のオプション	特にありません。
option 文の cutdtmsg オペランド	cutdtmsg オペランドを指定しなくても KFPL31090-E メッセージが出力されます。
option 文の dataerr オペランド	ロールバック対象の入力データエラーとして扱います。
extdat 文	拡張 DAT 形式の入力データファイルでも有効となります。
source 文の error オペランド	KFPL31090-W/E メッセージの出力回数が次のように変わります。

オプション及び制御文	lengover オペランドと同時に指定する場合の留意事項
	lengover オペランド指定あり： 　データ長不正を検知した最初の列だけメッセージを出力します。 lengover オペランド指定なし： 　データ長不正を検知したすべての列のメッセージを出力します。
source 文の errdata オペランド	エラーデータファイルに該当データを出力します。
上記以外の制御文	特にありません。

(l)　divermsg=off

分割格納条件を変更した表に対して，RD エリア単位にデータロードする場合，入力データ中に RD エリアの格納条件と一致しない行データがあると，エラーデータ情報を出力します。このエラーデータ情報の出力を抑止する場合，このオペランドを指定します。

このオペランドは，データロード対象表が横分割表で，かつ RD エリア単位にデータロードする場合に有効となります。これ以外の条件でこのオペランドを指定した場合，指定は無視されます。

分割格納条件と一致しない行データがある場合の，divermsg オペランドと各項目との関係を次の表に示します。

表 5–19　分割格納条件と一致しない行データがある場合の，divermsg オペランドと各項目との関係

項目	divermsg オペランド	
	指定あり	指定なし
pdload のリターンコード	0※	4
エラー情報ファイル	KFPL31009-E メッセージを出力しません。	KFPL31009-E メッセージを出力します。
エラーデータファイル	エラーデータを出力しません。	エラーデータを出力します。
-e オプション	無効	有効

注※

　データ形式エラーがあった場合，リターンコードは 4 となります。

(m)　srcendian=｛big｜little｝

エンディアンが異なるプラットフォーム間で pdrorg 出力バイナリ形式ファイルを使用してデータを移行する場合，移行先で入力データファイルのエンディアンを指定し，移行先のエンディアンに変換します。移行元と移行先のプラットフォームのエンディアンが同じ場合は，このオペランドの指定は不要です。

big：入力データファイルのエンディアンはビッグエンディアンです。

little：入力データファイルのエンディアンはリトルエンディアンです。

移行元と移行先でエンディアンが異なる場合のオペランド指定を次の表に示します。

表 5–20　移行元と移行先でエンディアンが異なる場合のオペランド指定

移行元	移行先	
	ビッグエンディアン	リトルエンディアン
ビッグエンディアン	–※1	srcendian=big
リトルエンディアン	srcendian=little	–※2

注※1

移行元と移行先でエンディアンが同じのため指定は不要です。ただし，srcendian=big の指定は可能です。

注※2

移行元と移行先でエンディアンが同じのため指定は不要です。ただし，srcendian=little の指定は可能です。

＜規則＞

- エンディアンが異なるプラットフォーム間でデータを移行する場合，pdrorg が出力したバイナリ形式のファイルを入力とするため，srcendian オペランドを指定するときは，-W と-b オプション，又は-U オプションを指定する必要があります。-W と-b オプションを両方指定しない場合，又は-U オプションを指定しない場合，srcendian オペランドは指定できません。指定すると，KFPL25002-E 又は KFPL24991-E メッセージを出力してエラー終了します。
- -b -W -w を指定した場合は，エンディアンの指定を無視し，入力ファイル中のエンディアンの設定に従います。

＜注意事項＞

srcendian オペランドの指定値を誤った場合，格納データは保証しません。次の現象が発生するおそれがあります。

- 入力データエラー
- プログラムの異常終了
- データベースに不正な値を格納

(n)　allspace=zero

固定長データ形式の入力データファイルを使用してデータロードする場合，数値データ型（INTEGER，SMALLINT，DECIMAL，FLOAT，SMALLFLT）の列に格納する文字データを「0」と変換して格納するときに指定します。なお，この場合の文字データとは，文字形式で指定（type=char(n)，type=adec(n,m)）し，かつその指定値がすべて空白（0x20），タブ（0x09），ナル文字（0x00）というデータです。

このオペランドは，-a オプションを指定した固定長データ形式の入力データファイルの場合だけ有効となります。そのほかの形式の場合にこのオペランドを指定した場合，指定は無視されます。

allspace オペランドの指定と，数値データ型の列に文字データを格納する場合の pdload の動作を次の表に示します。

表 5–21　allspace オペランドの指定と pdload の動作

allspace オペランド	pdload の動作
指定なし	データ変換エラーとし，KFPL31004-E メッセージをエラー情報ファイルに出力して，DB 格納処理をスキップします。
指定あり	該当する列のデータ型の「0」を DB に格納します。

(o)　whitespace=｛preserve ｜ strip｝

XML 型の列に対して XML 文書をデータロードする場合に，XML 文書に含まれる空白類の取り扱い方法を指定します。

preserve：すべての空白類を保持します。

strip：テキストノードに含まれる空白類に対して次の処理を行います（xml:space="preserve"属性を持つ要素の子孫のテキストノードは除きます）。

- テキストノードの先頭及び末尾の空白類を除去する
- 連続する空白類を一つの空白に置き換える

(p)　seq_range=｛buff ｜順序番号取得単位｜ all｝

自動採番機能を使用してデータロードする場合，順序番号の取得方式を指定します。

buff

順序番号をバッファ単位取得方式で取得します。この方式は，入力バッファ上に読み込まれた行数分の順序番号を取得してからデータロードします。

入力バッファの大きさを変更する場合は，option 文の file_buff_size オペランドを指定します。

なお，DAT 形式ファイル，又は拡張 DAT 形式ファイルの場合は入力バッファに読み込まれた行数が不定になるため，常に 1 行単位で順序番号を取得します。そのため，大量のデータロードをする場合に buff を指定することはお勧めしません。

順序番号取得単位　～＜符号なし正数＞((1～2147483647))

順序番号を指定単位取得方式で取得します。この方式は，指定した行数分の順序番号を取得してからデータロードします。

all

順序番号を全数一括取得方式で取得します。

順序数生成子循環オプションに cycle を指定していない順序数生成子の場合，順序番号の範囲を超える値は取得できません。そのため，all を指定した場合は，順序番号が取得範囲を超えた時点でエラー終了します。また，all 以外を指定した場合は，順序番号が取得範囲を超えて，指定した取得単位の順序番号を取得できなかったときにエラー終了します。

(q) file_buff_size=バッファ長

～<符号なし整数>((32～512000)) (単位：KB)

このオペランドは seq_range=buff の場合に指定できます。入力バッファに入力データファイルのデータを読み込む場合に，使用する入力バッファ用のメモリサイズを指定します。

このオペランドを省略した場合は，1,024 となります。

(r) charset={utf-16be | utf-16le}

UTF-8 の環境で定義した表に対して，UTF-16 の入力データファイルでデータロードする場合，入力データファイルのエンディアンを指定します。このオペランドは，入力データファイルが次の形式の場合に指定できます。

- 拡張 DAT 形式
- 固定長データ形式
- pdrorg 出力バイナリ形式（拡張 DAT 形式又は固定長データ形式の pdload 用アンロードファイル）

utf-16be：エンディアンがビッグエンディアンの場合に指定します。

utf-16le：エンディアンがリトルエンディアンの場合に指定します。

このオペランドを省略した場合，入力データファイルの文字コードは UTF-8 と判断してデータロードをします。

また，このオペランドを指定した場合，UTF-8 の列へ UTF-16 のデータを格納するため，pdload が文字コード変換をします。文字コード変換が発生するデータ型を次に示します。

項番	データ型		文字コード変換と charset オペランド指定の関係		
			指定なし	指定あり	
				utf-16be	utf-16le
1	CHAR，VARCHAR	文字集合指定なし	×	○	○
2		文字集合 EBCDIK	◎	×	×
3		文字集合 UTF-16	○	×	◎
4	NCHAR，NVARCHAR		×	×	×
5	MCHAR，MVARCHAR		×	○	○
6	BINARY 型		×	×	×
7	BLOB	データ	×	×	×
8		ファイル名※	×	○	○
9	抽象データ型	CHAR，VARCHAR	×	○	○

項番	データ型			文字コード変換と charset オペランド指定の関係		
				指定なし	指定あり	
					utf-16be	utf-16le
10		NCHAR，NVARCHAR		×	×	×
11		MCHAR，MVARCHAR		×	○	○
12		BINARY 型	データ	×	×	×
13			ファイル名※	×	○	○
14		BLOB	データ	×	×	×
15			ファイル名※	×	○	○
16		上記以外		×	×	×
17	上記以外			×	×	×

（凡例）

○：文字コードを変換します（変換後にデータ長が変わります）。

◎：文字コードを変換します（変換後もデータ長は変わりません）。

×：文字コードを変換しません。

注※

ファイル名は列のデータ型に関係なく，OS が処理できる文字コードに変換します。

変換前と変換後でデータ長が変わる場合，pdload がデータ長を調整します。列のデータ型と変換後データ長の組み合わせによるデータ長の調整方法を次の表に示します。

表 5–22　列のデータ型と変換後データ長の組み合わせによるデータ長の調整方法

列のデータ型	変換後データ長	
	変換後データ長≦列定義長	変換後データ長＞列定義長
CHAR，及び MCHAR	データを左詰めにし，足りない部分に空白を埋めて格納します。	データを左詰めにし，列定義長を超える部分を切り捨てて格納します。※
VARCHAR，及び MVARCHAR	変換後データ長でデータを格納します。	

注※

データの切り捨てが発生した場合，option 文の lengover オペランド，及び cutdtmsg オペランドの指定値によってその後の処理が変わります。lengover オペランド及び cutdtmsg オペランドの指定値と，データの切り捨てが発生した場合の処理を次に示します。

lengover オペランド	cutdtmsg オペランド	データの切り捨てが発生した場合の処理
省略	省略又は off	そのまま処理を続行します。

lengover オペランド	cutdtmsg オペランド	データの切り捨てが発生した場合の処理
	on	エラー情報ファイルに警告メッセージを出力し，処理を続行します。
err	−	異常終了します。

（凡例）−：該当しません。

5.5　入力データファイル

この節では，入力データファイルについて説明しています。

- **DAT 形式**
  列データを文字で指定する形式です。
  詳細については，「DAT 形式の場合」を参照してください。
- **バイナリ形式**
  HiRDB 内部形式に沿った形式です。
  詳細については，「バイナリ形式の場合」を参照してください。
- **固定長データ形式**
  DAT 形式のように文字でも記述でき，バイナリ形式のように HiRDB 内部形式でも記述できます。
  詳細については，「固定長データ形式の場合」を参照してください。
- **繰返し列を含む表の配列データ形式**
  データロードする表に繰返し列がある場合，入力データファイルのデータも繰返し列に合わせて配列データ形式で記述します。形式としては，DAT 形式，バイナリ形式，又は固定長データ形式のどれかで記述することになります。
  詳細については，「繰返し列を含む表の配列データ形式」を参照してください。
- **列単位 LOB 入力ファイル**
  列単位 LOB 入力ファイルには，入力行数分の LOB データを記述します。
  詳細については，「列単位 LOB 入力ファイル」を参照してください。
- **pdrorg 出力バイナリ形式ファイル**
  pdrorg を使用してバイナリ形式でアンロードしたファイルです。詳細については，「pdrorg 出力バイナリ形式ファイル」を参照してください。

5.5.1　DAT 形式の場合

DAT 形式の場合，入力データ，及びコンストラクタ関数への入力パラメタは文字データで記述します。

(1)　記述形式

列データと列データの間をセパレータ文字で区切って指定します。

各データ型に対する列データ，及びコンストラクタ関数のパラメタの記述形式（DAT 形式）を次の表に示します。

表 5–23　各データ型に対する列データ・パラメタの記述形式（DAT 形式）

データ型		列データ・パラメタ記述形式	
		一般データ	ナル値又は既定値※1 の指定方法
数データ	INTEGER	数値を文字で記述します。 マイナスは，"-"符号を付けます。 小数点は，整数．小数で記述します。 浮動小数点は，仮数部ｅ指数部で記述します。 整数（例：-1234 … 0 … 1234） 小数点（例：-1.56 … 0 … 1.56） 浮動小数点（例：-2.4e ＋ 9 … 0e0 … 2.4e ＋ 9）	’＊’又は記述しません。
	SMALLINT		
	DECIMAL		
	FLOAT		
	SMALLFLT		
文字データ	CHARACTER	文字，又は前後を囲み文字（"）で囲んである文字で記述します。 （例：abcdef や ABCDEF など） （例："abcd"や"ABCD"など）	記述しません。
	VARCHAR		
各国文字データ	NCHAR		
	NVARCHAR		
混在文字データ	MCHAR		
	MVARCHAR		
日付データ	DATE	数値 yyyy-mm-dd の形式で記述します。 yyyy：西暦　mm：月　dd：日 （例：1995 年 1 月 1 日 → 1995-01-01）	’＊’又は記述しません。
日間隔データ	INTERVAL YEAR TO DAY	数値〔-〕yyyymmdd．の形式で記述します。マイナスは"-"符号を付けます。 （例：1111 年 1 か月 1 日間 → 11110101．）	
時刻データ	TIME	数値 hh：mm：ss の形式で記述します。 hh：時　mm：分　ss：秒 （例：12 時 1 分 1 秒 → 12：01：01）	
時間隔データ	INTERVAL HOUR TO SECOND	数値〔-〕hhmmss．の形式で記述します。マイナスは"-"符号を付けます。 （例：1 時間 1 分 1 秒 → 010101．）	
時刻印データ	TIMESTAMP	数値 YYYY-MM-DD hh:mm:ss〔.nnnnnn〕の形式で記述します。 小数秒部は，0〜6 けたの値を指定します。小数部が定義したけた数に満たない場合は 0 を仮定し，長ければ切り捨てます。ただし，けた数が 7 以上の場合はエラー（KFPL31002-E）となります。※2	
バイナリデータ	BINARY	バイナリデータ，又はバイナリデータを囲み文字で囲んで記述します。 （例：abc 123 又は"abc 123"）	記述しません。

データ型		列データ・パラメタ記述形式	
		一般データ	ナル値又は既定値※1 の指定方法
	XML 型の BINARY 属性パラメタ	-K f の場合に，XML 文書，又は ESIS-B 形式のデータを格納したファイルの名称を記述します。 （例：¥user¥xml_dir¥xml001.txt）	
長大データ	BLOB	＜-k　f の場合＞ LOB 入力ファイルの名称を記述します。 （例：lobfile01） ＜-k　c の場合＞ セパレータ文字を含まない 1～1023 バイトの文字列で記述します。なお，この文字列は LOB 列作成時には無視されます。 （例：@）	

注

各データ型でのナル値の扱いは次のようになります。

- 固定長の文字データ型，各国文字データ型，混在文字データ型，バイナリデータ型，及び長大データ型の場合，次の記述をナル値又は既定値として扱います。

  …,,…　　…,""，…

- -z オプション指定ありの可変長の文字データ型，各国文字データ型，混在文字データ型，及びバイナリデータ型の場合，次の記述をナル値又は既定値として扱います。

  …,,…

  なお，…,""，…は長さ 0 のデータとして扱います。

- そのほかのデータ型の場合，次の記述をナル値又は既定値として扱います。

  …,,…　　…,""，…　　…，＊，…

extdat 文で enclose オペランドを指定した場合，引用符（"）は enclose オペランドに指定した囲み文字となります。

注※1

option 文の null_string オペランドの指定値によって，格納される値が決まります。

注※2

TIMESTAMP 型の小数秒部のけた数を指定した場合の格納方法を次に示します。

定義けた数	入力データの小数秒部のけた数							
	0	1	2	3	4	5	6	7 以上
0	そのまま格納します。※1	定義けた数以降を切り捨てます。						エラー（KFPL31002-E）

定義けた数	入力データの小数秒部のけた数							
	0	1	2	3	4	5	6	7 以上
2	定義けた数まで 0 を埋めます。※2		そのまま格納します。※1	定義けた数以降を切り捨てます。				
4	定義けた数まで 0 を埋めます。※2				そのまま格納します。※1	定義けた数以降を切り捨てます。		
6	定義けた数まで 0 を埋めます。※2						そのまま格納します。※1	

注※1

入力データを TIMESTAMP 型に変換して，そのまま格納します。

注※2

実データはバイト単位のため，定義けた数が奇数の場合，定義けた＋ 1 の箇所まで 0 を埋めます。

(2) 規則

- ファイルの 1 行に対して，表の 1 行のデータを記述します（行の終わりには改行文字が必要です）。
- コンストラクタ関数に渡す入力パラメタを記述する場合，パラメタをセパレータ文字で区切ります。
- データ中のナル文字（0x00），及び改行記号（0x0a），及び行の終端の復帰改行（0x0d0a）は，データとして扱われません。これらをデータとして扱いたい場合は，extdat 文を指定して囲み文字で入力データを囲んでください。
- 1 行は最大 32 キロバイトです。ただし，source 文に maxreclen オペランドを指定した場合は，32 キロバイト以上でもデータロードできます。
- 囲み文字で列の入力値が始まると，囲み文字とセパレータ文字が連続した位置を列の終わりと扱います。そのため，囲み文字とセパレータ文字が連続した値を列の入力値とする場合は，-s オプションでセパレータ文字を変更してください。
- セパレータ文字の直前，直後の文字データの前後を囲み文字（"）で囲んだ場合，その囲み文字は入力対象にはなりません。また，囲み文字で囲まれた文字列内のセパレータ文字は，データとして扱われます。文字データの入力と囲み文字の扱いを次の表に示します。

表 5–24　文字データの入力と囲み文字の扱い

入力データファイル内の文字データ	表へ格納されるデータ	
	DAT 形式	拡張 DAT 形式
～,ABCD,～	ABCD	ABCD
～,AB"CD,～	AB"CD	入力データエラー
～,AB""CD,～	AB""CD	AB"CD

入力データファイル内の文字データ	表へ格納されるデータ	
	DAT 形式	拡張 DAT 形式
～,"ABCD",～	ABCD	ABCD
～,"AB"CD",～	AB"CD	入力データエラー
～,"AB""CD",～	AB""CD	AB"CD
～,""ABCD"",～	"ABCD"	入力データエラー
～,""""ABCD"""",～	""ABCD""	"ABCD"
～,,～	ナル値又は既定値※	ナル値又は既定値※
～,"",～	ナル値又は既定値※	ナル値又は既定値※
～,""""",～	""	"
～,"AB,CD",～	AB,CD	AB,CD
～,AB¥0CD,～	入力データエラー	入力データエラー
～,AB¥nCD,～	入力データエラー	入力データエラー
～,"AB¥0CD",～	入力データエラー	AB¥0CD
～,"AB¥nCD",～	入力データエラー	AB¥nCD

（凡例）

「,」：セパレータ文字

「"」：囲み文字

「¥0」：ナル文字（0x00）

「¥n」：改行コード

注※

option 文の null_string オペランドの指定値によって，格納される値が決まります。

- 数値型データに変換する文字データの記述形式を次の表に示します。なお，入力データで数値データ型に文字データを指定する場合の制限については，マニュアル「HiRDB SQL リファレンス」の数定数の使用上の制限を参照してください。

表 5–25　数値型データに変換する文字データの記述形式

文字型データで表す数値	表記法	
	指定形式	指定例（指定→格納値）
整数	△〔{＋｜−}〕自然数△ 自然数は，0～9 の数字の並びを示します。	＋ 10△　→　10 −009　→　−9 △△△0△　→　0

文字型データで表す数値	表記法	
	指定形式	指定例（指定→格納値）
10 進数	△〔整数〕.〔小数〕△ 小数は，符号なし整数を示します。整数又は小数のどちらかは必ず指定してください。	△．05△　→　0.05 −0.8△　→　−0.8 1.0△△　→　1.0 ＋1.△　→　1.0
浮動小数点数※	△仮数 e〔指数〕△ 仮数は，整数又は 10 進数を示します。e は，E と指定してもよいです。指数は，1 又は 2 けたの整数を示します。	.3e　→　0.3×10 2.4e−02　→　2.4×10−2 67e−5△　→　67×10−5

（凡例）△：空白を指定してもしなくてもよい箇所を示します。

注※

仮数部が 17 桁を超える値を指定した場合，表へ格納されるデータが，バージョン 10-05 以前と異なることがあります。詳細はマニュアル「HiRDB システム導入・設計ガイド」の「バージョン 10-05 以前との FLOAT 型・SMALLFLT 型の差異について」を参照してください。

- 入力データのデータ長と表のデータ長が異なるときの格納データの扱いを次の表に示します。

表 5–26　入力データのデータ長と表のデータ長が異なるときの格納データの扱い

表のデータ長（m）	入力データのデータ長（n）	
	固定長文字データ	可変長文字データ，又はバイナリデータ
m=n の場合	入力データはそのまま格納されます。	
m<n の場合	option 文の lengover オペランドを指定していない場合，入力データの(m ＋ 1)以降のデータは切り捨てて格納されます。※ option 文の lengover オペランドを指定している場合，入力データエラーとして扱うため，データは格納されません。	
m>n の場合	入力データの(n ＋ 1)以降に空白が埋められた形式で格納されます。	入力データはそのまま格納されます。

注※

option 文に cutdtmsg=on を指定すると，データの切り捨てが発生した場合にエラー情報ファイルに警告メッセージを出力します。

- 入力データのデータ部分にセパレータ文字（セパレータ文字と同じコードも含む）がある場合，そのデータ全体を囲み文字（"）で囲まないとそのデータはセパレータ文字として扱われます。
- extdat 文に enclose="を指定した場合に，格納されるデータ及びナル値の指定方法を次の表に示します。

表 5–27　extdat 文に enclose="を指定した場合に，格納されるデータ及びナル値の指定方法

データ型		入力データ	表へ格納されるデータ	ナル値又は既定値の指定方法※
数データ	INTEGER	"1234"	1234	*，""，"*"，又は記述しません。
	SMALLINT	"1234"	1234	
	DECIMAL(6,1)	"000001"	00001.0	
	FLOAT	"2.225075e-308"	+2.225075000000000E-308	
	SMALLFLT	"1.175495e-38"	+1.175495051471520E-38	
文字データ	CHAR	"ABCDE"	ABCDE	""，又は記述しません。
	VARCHAR	"ABCDE"	ABCDE	
各国文字データ	NCHAR	"あいうえお"	あいうえお	
	NVARCHAR	"あいうえお"	あいうえお	
混在文字データ	MCHAR	"ABCDE"	ABCDE	
	MVARCHAR	"ABCDE"	ABCDE	
日付データ	DATE	"2000-05-31"	2000-05-31	*，""，"*"，又は記述しません。
日間隔データ	INTERVAL YEAR TO DAY	"+00000000."	00000000	
時刻データ	TIME	"09:00:00"	09:00:00	
時間隔データ	INTERVAL HOUR TO SECOND	"+000000."	000000	
時刻印データ	TIMESTAMP(6)	"2002-08-08△20:25:43.39"	2002-08-08 20:25:43.390000	
バイナリデータ	BINARY	"ABCDE"	ABCDE	""，又は記述しません。
	XML 型の BINARY 属性パラメタ	"XML 文書ファイルパス名"	XML 文書ファイルの内容	
長大データ	BLOB	"LOB 入力ファイル名"	LOB 入力ファイルの内容	

（凡例）

△：半角の空白

注※

option 文の null_string オペランドの指定値によって，格納される値が決まります。

- extdat 文に enclose="を指定した場合の空白削除の例を次の表に示します。

表 5–28　extdat 文に enclose="を指定した場合の空白削除の例

データ型		入力データ	表へ格納されるデータ
数データ	INTEGER	"△1234△"	1234
	SMALLINT	"△"	ナル値
	DECIMAL(6,1)	"△000001"	00001.0
	FLOAT	"△2.225075e-308"	+2.225075000000000E-308
	SMALLFLT	"1.175495e-38△"	+1.175495051471520E-38
日付データ	DATE	"△2000-05-31"	2000-05-31
日間隔データ	INTERVAL YEAR TO DAY	"+△00000000."	00000000
時刻データ	TIME	"△09:00:00"	09:00:00
時間隔データ	INTERVAL HOUR TO SECOND	"△+△000000."	000000
時刻印データ	TIMESTAMP	"△2002-08-08△△20:25:43.39△"	2002-08-08 20:25:43.39

（凡例）

△：半角の空白

- 拡張 DAT 形式の文字集合として UTF-16 又は IVS 対応 UTF-16 を使用する場合，2 バイト単位に入力データを解析します。そのため，拡張 DAT 形式の入力データファイルに，奇数バイト長の BINARY 列データ，及び奇数バイト長の BINARY 属性パラメタデータを記述する場合は，次のどちらかの対処をしてください。
  - 文字コードを UTF-8 にする。
  - 入力データファイルをバイナリ形式又は固定長データ形式にする。

  上記の対処をしなかった場合，pdload はセパレータ文字や改行文字を正しく認識できないため，次の現象が発生します。
  - KFPL31005-E エラー
  - 改行を認識できないため，後続の行を同一列のデータと誤認します。そのため，BINARY 列又は BINARY 属性パラメタを持つ抽象データ型列へ不当に長いデータが格納されます。
- 入力データファイルに予備列のデータを記述しないでください。入力データファイルに予備列のデータが記述された場合，論理エラー番号の 5 を出力します。論理エラー番号については，「エラー情報ファイルの内容」を参照してください。

(3)　記述例

DAT 形式のデータ記述例を次の図に示します。

図 5–15　DAT 形式のデータ記述例

入力データファイル

```
中村,36,1958-10-15,yokohama
鈴木,25,1968-01-01,kamakura
田中,41,1953-12-31,shibuya
佐藤,36,1963-05-30,kawasaki
小田,46,1948-04-01,shinjuku
```

注　それぞれのデータ型は，ＮＣＨＡＲ，ＩＮＴＥＧＥＲ，ＤＡＴＥ，ＶＡＲＣＨＡＲの順です。

5.5.2　バイナリ形式の場合

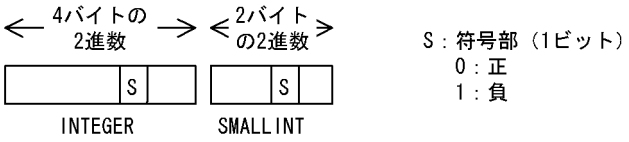
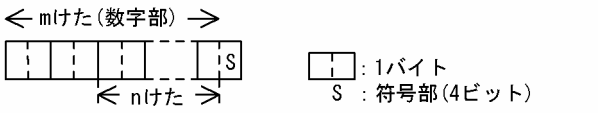
バイナリ形式の場合，入力データはデータベースへ格納される形式で記述します。記述方法を次に示します。

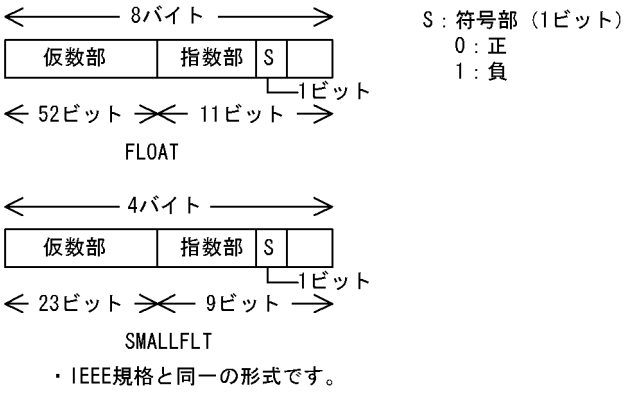
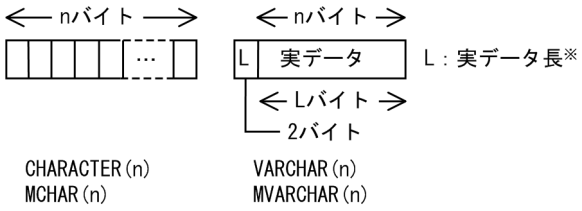
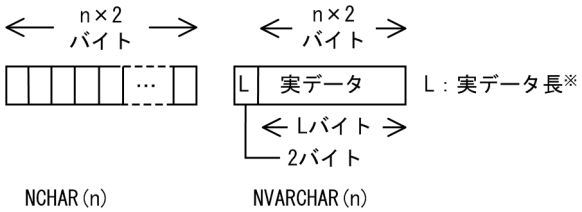
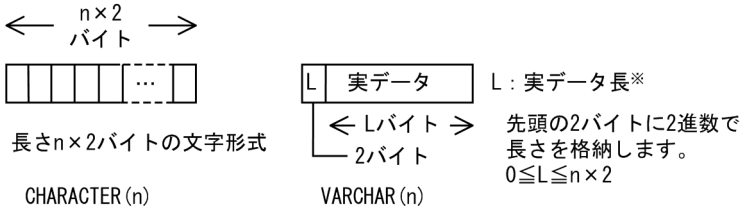
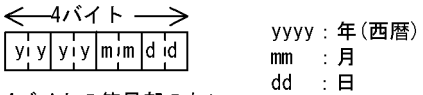
(1)　記述形式

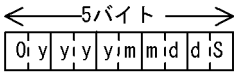
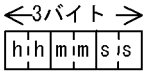
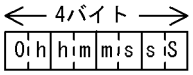
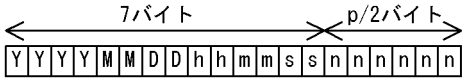
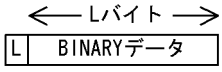
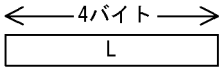
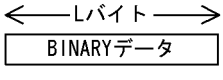
(a)　列データの記述形式

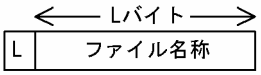
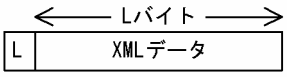
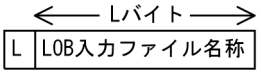
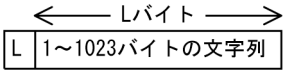
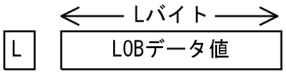
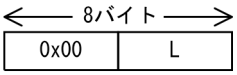
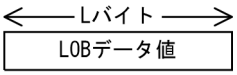
各データ型に対する列データ・パラメタの記述形式（バイナリ形式）を次の表に示します。

表 5–29　各データ型に対する列データ・パラメタの記述形式（バイナリ形式）

データ型・パラメタ型		列データ記述形式
数データ	INTEGER，SMALLINT	
	DECIMAL(m,n) 1≦m≦38 0≦n≦38 m≧n	 小数点の位置 ・パック形式は，1バイトで2けたの数字を表します。 ・符号部は，aefはcに，bはdに正規化されます。

データ型・パラメタ型		列データ記述形式
	FLOAT，SMALLFLT	 ・IEEE規格と同一の形式です。
文字データ，各国文字データ，混在文字データ※1	CHARACTER(n)， VARCHAR(n)，MCHAR(n)， MVARCHAR(n)	 注※ -zオプションを指定した場合にだけ実データ長に0が入り，実データ部がない形式を長さ0のデータとして扱います。
	NCHAR(n)， NVARCHAR(n)	 注※ -zオプションを指定した場合にだけ実データ長に0が入り，実データ部がない形式を長さ0のデータとして扱います。
文字データ※2	CHARACTER(n)， VARCHAR(n)	 注※ -zオプションを指定した場合にだけ実データ長に0が入り，実データ部がない形式を長さ0のデータとして扱います。
日付データ	DATE	 4バイトの符号部のないパック形式10進数 ・パック形式は，1バイトで2けたの数字を表します。

データ型・パラメタ型		列データ記述形式
日間隔データ	INTERVAL YEAR TO DAY	 5バイトの パック形式10進数 yyyy：年（西暦） mm　：月 dd　：日 S　　：符号部（4ビット） ・パック形式は，1バイトで2けたの数字を表します。 ・符号部は正規化されません。 ・mmに12以上を指定した場合は，年に繰り上がります。
時刻データ	TIME	 3バイトの符号部のない パック形式10進数 hh：時 mm：分 ss：秒 ・パック形式は，1バイトで2けたの数字を表します。
時間隔データ	INTERVAL HOUR TO SECOND	 4バイトの パック形式10進数 hh：時 mm：分 ss：秒 S　：符号部（4ビット） ・パック形式は，1バイトで2けたの数字を表します。 ・符号部は正規化されません。 ・mm，ssに60以上を指定した場合は，それぞれ時，分に繰り上がります。
時刻印データ	TIMESTAMP(p)	 最大10バイトの符号部のない パック形式10進数 ・小数秒は，定義によって0，2，4，又は6となります。 YYYY：年 MM：月 DD：日 hh：時 mm：分 ss：秒 nnnnnn：小数秒
BINARYデータ，BINARY属性データ	BINARY	＜データベース作成ユティリティの入力データファイルの場合＞  先頭4バイトに2進数でデータの長さを格納します。 ＜データベース再編成ユティリティの-W指定の入力データファイルの場合＞ （基本データ部）  列データ部の4バイトに2進数でデータ長を格納します。 （BINARY列データ部及びBINARY属性データ部）  データは，すべての列データ（LOB列構成基表データ）の後に出力します。 BINARYデータ，及びBINARY属性データは，すべての列データ（LOB列構成基表データ）の後に，LOB属性データ及びLOBデータよりも前，かつBINARY属性データ，BINARYデータの順で，それぞれ列定義順に出力します。

データ型・パラメタ型		列データ記述形式
	XML 型の BINARY 属性パラメタ	＜ファイルパス指定形式の場合＞  先頭4バイトに2進数でファイル名称の長さを格納します。 固定長データ形式で，-K fオプションを指定したときに記述できます。 ＜埋め込み形式の場合＞  先頭4バイトに2進数でデータの長さを格納します。 -K iオプションを指定したときに記述できます。
LOB データ，LOB 属性データ	BLOB	＜-k fの場合＞  先頭4バイトに2進数でLOB入力ファイル名称の長さを格納します。 ＜-k cの場合＞  先頭4バイトに2進数で指定した文字列の長さを格納します。この文字列はLOBデータ作成時には無視されます。 ＜-k v，又は-k dの場合＞※3  ＜データベース再編成ユティリティの-W指定の入力データファイルの場合＞ （基本データ部）  列データ部8バイトの領域の先頭4バイトに0x00を格納し，後ろ4バイトにデータ長を2進数で格納します。 （BLOB列データ部及びBLOB属性データ部）  データは，すべての列データ（LOB列構成基表データ）の後に出力します。 LOB属性データは，LOB列構成基表のデータの後ろで，LOBデータよりも前に出力されます。

注※1

文字コード変換なし又は変換後もデータ長が変化しない文字コード変換をする場合

注※2

変換後にデータ長が変化する文字コード変換をする場合

注※3

詳細は，「-k オプションに v，又は d を指定した場合の LOB データ，及び LOB パラメタの記述方法」を参照してください。

(b)　入力パラメタの記述形式

入力パラメタの記述方法を次に示します。

- 抽象データ型に格納する値を生成するコンストラクタ関数の入力パラメタを，関数のパラメタ型と同じ構造で記述します。
- 入力パラメタの順序とコンストラクタ関数の引数の順序は同じにしてください。
- 格納先の列が抽象データ型の場合，コンストラクタ関数は抽象データ型と同じ名称の関数を使用します。引数が異なる同一名称のコンストラクタ関数が複数ある場合や，抽象データ型の名称とは異なるコンストラクタ関数を使用する場合は，使用する関数名をナル値・関数情報ファイルに指定してください。
- 入力パラメタが複数ある場合，パラメタ値とパラメタ値の間は，空白を入れないで詰めます。

(2) 規則

- 磁気テープ装置の場合に，可変長ブロックモードで入力データを作成するときは，ブロック長を 32 キロバイト以下にしてください。
- 入力データの列順序と表定義の列順序は同じにしてください。
- 列データと列データの間は，空白を入れないで詰めます。可変長文字列の場合で定義長に実際のデータが満たない場合は，実長でデータを記述し，次のデータを詰めます。
- 入力データに次のデータがある場合，同期点指定のデータロードはお勧めできません。次のデータがあると，不正が検知されても，同期点までのデータが格納されてしまいます。
  - 繰返し列
  - 可変長データ（VARCHAR，NVARCHAR，MVARCHAR，BINARY，及び BLOB）
  - 抽象データ型の引数が可変長データ
- バイナリ形式は，入力データを直接記述するフォーマットのため，デフォルト値を格納できません。デフォルト値を格納する場合は，入力データファイル作成時に直接デフォルト値を記述してください。
- 入力データファイルに予備列のデータを記述しないでください。入力データファイルに予備列のデータが記述された場合，pdload は，次のどちらかの動作をします。なお，pdrorg 出力バイナリ形式の入力データに予備列のデータが記述された場合も同様です。
  - エラーが発生し，論理エラー番号の 2，4，5，7，8 又は 9 を出力します。
    論理エラー番号については，「エラー情報ファイルの内容」を参照してください。
  - リターンコード 0 で正常終了します。
    ただし，KFPL00703-I メッセージで示した格納行数が，入力したデータ行数より多くなります。

(3) 記述例

バイナリ形式のデータ記述例を次の図に示します。

図 5–16　バイナリ形式のデータ記述例

入力データファイル

```
928691ba8140814081400000000243f8000000008796f6b6f68616d61
 中　村                    36        1      yokohama
8db293a18140814081400000001440000000000773686962757961
 佐　藤                    20        2      shibuya
97e996d88140814081400000029c059999a0005746f6b796f
 鈴　木                    41     -3.4      tokyo
```

注

1. 入力データの上段はデータを，下段は格納される値を示します。
2. それぞれのデータ型は，NCHAR(5)，INTEGER，SMALLFLT，VARCHAR の順です。
3. 各国文字データはシフト JIS コードです。

(4)　-k オプションに v，又は d を指定した場合の LOB データ，及び LOB パラメタの記述方法

(a)　-k v の場合

-k オプションに v を指定した場合，LOB データの入力ファイル名と LOB パラメタのデータは，入力データファイルに直接記述します。

この方法は，-k オプションに f を指定した場合に比べて，LOB 入力ファイルの入出力時間が削減できます。

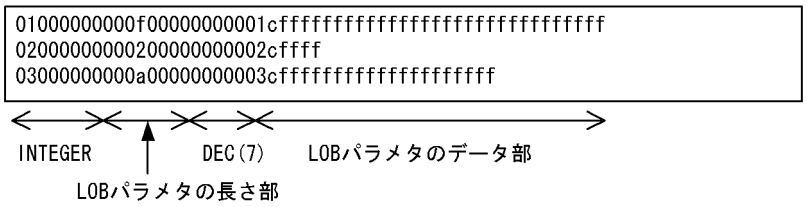
- **規則**

1. LOB データ及び LOB パラメタは，行データ中の該当列に 4 バイトの長さ部を持ち，行データの直後にデータ部を持つ形式で指定します。指定順序は行データの列順です。
2. LOB データ又は LOB パラメタがナル値の場合は，長さ部に-1 を指定します。ナル値・関数情報ファイル及び関数情報ファイルによるナル比較値は無視されます。データ部は指定しません。
3. LOB データ又は LOB パラメタが 0 バイトの場合は，長さ部に 0 を指定します。データ部は指定しません。

- **記述例**

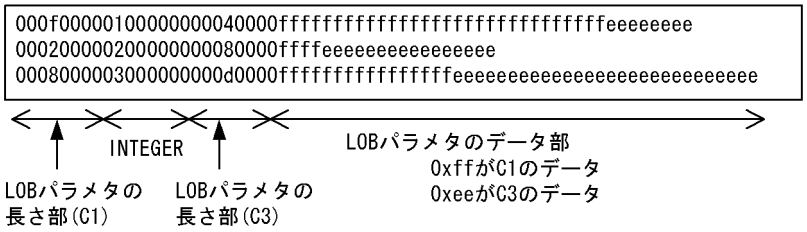
- **BLOB を入力とする抽象データ型の列を一つ持つ表の場合**

```
表定義：CREATE TABLE T1(C1 INTEGER,C2 SGMLTEXT,C3 DEC(7,0))
```



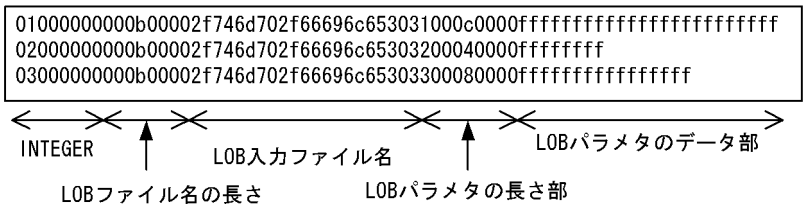
- **BLOB を入力とする抽象データ型の列を二つ持つ表の場合**

表定義：CREATE TABLE T2(C1 SGMLTEXT,C2 INTEGER,C3 SGMLTEXT)



- **LOB 列と BLOB を入力とする抽象データ型の列を持つ表の場合**

表定義：CREATE TABLE T3(C1 INTEGER,C2 BLOB(100),C3 SGMLTEXT)



(b)　-k d の場合

-k オプションに d を指定した場合，入力データファイルには，LOB データ格納ファイル名称でなく，データそのものを記述します。

- **規則**

1. LOB データ，LOB パラメタデータは行データ中の該当列に 4 バイトの長さ部を保持し，行データの直後にデータ部を持つ形式で指定してください。
2. 入力データファイルの指定順序は次に示す規則に従ってください。

```
[行データ] [LOBパラメタデータ] [LOBデータ]
    1                2              3
```

[説明]

1. 長さ部は行データ中に列定義順で指定します。
2. LOB パラメタデータは，行データの直後に列定義順で指定します。
3. LOB データは「2.」に続いて列定義順に指定します。

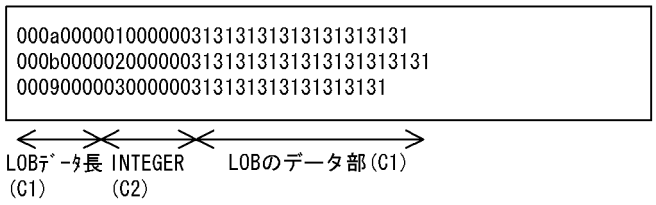
3. LOB データ，LOB パラメタデータがナル値の場合は，4 バイトの長さ部に-1 を指定してください。ナル値・関数情報ファイルによるナル比較値は無視します。なお，この場合データ部はありません。

4. LOB データ，LOB パラメタデータが 0 バイトの場合は，長さ部に 0 を指定してください。なお，この場合データ部はありません。

- **記述例**

- **LOB 列を一つ持つ表の場合**

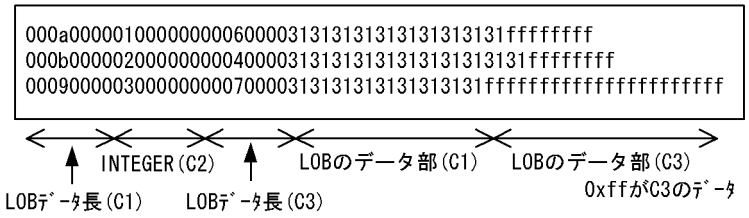
```
表定義：CREATE TABLE T1(C1 BLOB(100),C2 INTEGER);
```



［指定方法］
1．LOBデータの長さ部（４バイト）を行データの前に指定します。
2．LOBデータを「１．」に続いて指定します。

- **LOB 列を二つ持つ表の場合**

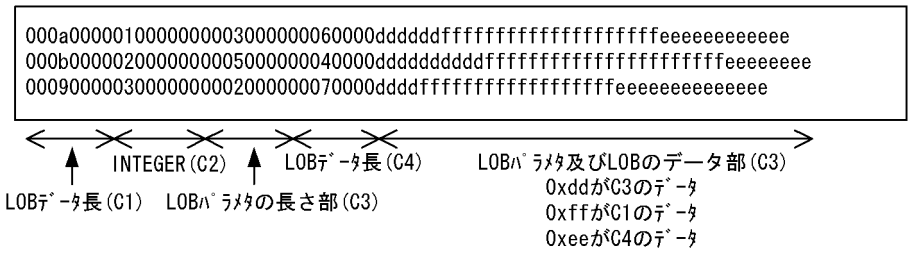
```
表定義：CREATE TABLE T2(C1 BLOB(100),C2 INTEGER,C3 BLOB(100));
```



［指定方法］
1．LOBデータの長さ部（４バイト）を行データ中に列定義順（この場合，LOBデータ長(C1)，LOBデータ長(C3)の順）に指定します。
2．LOBデータを「１．」に続いて，列定義順（この場合，LOBデータ(C1)，LOBデータ(C3)の順）に指定します。

- **LOB 列を二つと LOB を入力とする抽象データ型の列を持つ表の場合**

```
表定義：CREATE TABLE T3(C1 BLOB(100),C2 INTEGER,C3
              SGMLTEXT,C4 BLOB(100));
```



［指定方法］

1．LOBデータ及びLOBパラメタの長さ部（４バイト）を行データ中に列定義順（この場合，LOBデータ長(C1)，LOBパラメタの長さ部(C3)，LOBデータ長(C4)）の順）に指定します。
2．LOBパラメタデータを行データの直後に列定義順（この場合，LOBパラメタデータ(C4)だけ）に指定します。
3．LOBデータを「２．」に続いて，列定義順（この場合，LOBデータ(C1)，LOBデータ(C4)の順）に指定します。

5.5.3　固定長データ形式の場合

固定長データ形式の場合，入力データは文字，又はデータベースに格納される形で記述します。

抽象データ型に格納するデータをコンストラクタ関数で生成する場合は，入力パラメタを文字データ，又は関数の引数の型で記述します。

なお，文字列形式の数値データ，及び文字列形式の文字データとは，それぞれ次の表に示すデータ型の列に格納する文字列データのことです。

表 5–30　文字列形式の数値データと文字列形式の文字データ

文字列形式の数値データ	文字列形式の文字データ
• INTEGER • SMALLINT • DECIMAL • FLOAT • SMALLFLT • DATE • INTERVAL YEAR TO DAY • TIME • INTERVAL HOUR TO SECOND • TIMESTAMP	• CHAR • VARCHAR • MCHAR • MVARCHAR • NCHAR • NVARCHAR

(1)　記述形式

入力データを文字で記述する場合は，「DAT 形式の場合」を参照してください。データベースに格納される形で記述する場合は，「バイナリ形式の場合」を参照してください。

(2) 規則

- 1 行の長さは固定です。
- 各列値の開始位置は統一してください。また，数値に変換して格納する文字列は左詰めでも右詰めでもよいですが，文字のまま格納するときは左詰めにしてください（可変長文字列の場合は，空白などを埋めて次の列の開始位置を統一してください）。
- ナル値は列構成情報ファイルの指定に従います。
- CHAR，VARCHAR，NCHAR，NVARCHAR，MCHAR，及び MVARCHAR に対して固定長の文字列データを入力した場合（type=char(n)を指定した場合），先頭 1 バイトを除いて，列値の終端から連続する半角空白，タブ，及び¥0 を削除します。その後，ナル比較値の比較，及びデータベースへのデータの格納をします。

  そのため，タブと¥0 は CHAR 及び MCHAR には格納できません。半角空白，タブ，及び¥0 は，VARCHAR，NCHAR，NVARCHAR，及び MVARCHAR には格納できません。また，半角文字を入力できないデータ型（NCHAR 及び NVARCHAR）に対して半角空白，タブ，及び¥0 があるデータを入力した場合，次のようになります。

  (a) すべて半角空白，タブ，又は¥0 のデータの場合

  半角空白，タブ，及び¥0 を取り除いた結果，先頭 1 バイトだけが残ります。ただし，NCHAR 型，及び NVARCHAR 型は，データ長が必ず偶数バイトでなければならないため，データ長が不正なデータとしてエラーになります。

  (b) 半角空白，タブ，又は¥0 と，それ以外のデータが混在する場合

  列値の終端から連続する半角空白，タブ，及び¥0 を削除した結果，データ長が奇数バイトの場合，(a)と同様にデータ長が不正なデータとしてエラーになります。半角空白，タブ，及び¥0 を削除した結果，データ長が偶数バイトの場合はデータベースにデータを格納します。

  以上の点から，NCHAR 型，及び NVARCHAR 型へ格納する固定長文字列の入力データの後方を埋める文字は，半角空白，タブ，及び¥0 ではなく，全角空白を使用するようにしてください。

  また，表のデータ型が NCHAR の場合，入力対象外としたデータのために格納する入力データが表の定義長に満たない部分には，全角空白が仮定されます。
- 入力データファイルの列データの形式を，必ず列構成情報ファイルに指定してください。なお，固定長データ形式の場合，ほかの入力データファイルにはない固有の入力データ型があります。詳細については，「固定長データ形式固有の入力データ型」を参照してください。
- 入力データ側の文字データの長さと格納表の定義長が異なる場合，DAT 形式ファイルの固定長データの場合と同様の扱いとなります。ただし，可変長文字データの長さ部が列構成情報ファイルに指定した定義長を超える場合，データは格納されません。また，可変長文字データの長さ部が負の値でも処理は続行します。
- データベースに格納されるデータ形式で入力データを記述する場合，バイナリ形式のフォーマットで記述してください。バイナリ形式のフォーマットについては，「バイナリ形式の場合」を参照してください。
- 文字列形式の数値データでデータを記述する場合，格納先の列のデータ型に変換可能な文字列でデータを記述してください。

- 列名文の mode オペランドの指定が text の場合だけ，文字列形式の文字データでは囲み文字が使用できます。入力データの先頭 1 バイト目が「"」で始まり，末尾が「"」で終わっているとき，「"」を囲み文字と判断し，囲まれているデータを入力データとして扱います。
- 文字列形式で入力データを記述する場合の空白（0x20）と囲み文字（0x22）の扱いと，文字列形式で入力データを記述した場合に，データベースに格納されるデータを次の表に示します。

表 5–31　文字列形式で入力データを記述する場合の空白（0x20）と囲み文字（0x22）の扱い（固定長データ形式）

表の列のデータ型，又はコンストラクタ関数のパラメタのデータ型	列名文及び option 文の指定内容	空白（0x20）		囲み文字（"）
		囲み文字で囲まれていない場合	囲み文字で囲まれている場合	
INTEGER，SMALLINT，DECIMAL，FLOAT，及び SMALLFLT	option 文 allspace オペランドの指定あり	先頭から後方へ連続する空白，及び終端から前方へ連続する空白を除いた値が入力データとなります。数値と数値の間や，符号と数値の間の空白は除かないため，数値や符号の途中に空白を指定しないでください。データがすべて空白の場合は指定値「0」となります。	入力データエラーとなります。	
	option 文 allspace オペランドの指定なし	先頭から後方へ連続する空白，及び終端から前方へ連続する空白を除いた値が入力データとなります。数値と数値の間や，符号と数値の間の空白は除かないため，数値や符号の途中に空白を指定しないでください。データがすべて空白の場合は入力データなしとなり，入力データエラーとなります。		
CHAR 及び MCHAR	mode=text	終端から前方へ連続する空白を除いた値が入力データとなります。	空白を含めた値が入力データとなります。	「"」を囲み文字として扱います。「"」を除いた値が入力データとなります。
	mode=bin	空白を含めた値が入力データとなります。		「"」を囲み文字として扱いません。「"」を含めた値が入力データとなります。

表の列のデータ型，又はコンストラクタ関数のパラメタのデータ型	列名文及び option 文の指定内容	空白（0x20）		囲み文字（"）
		囲み文字で囲まれていない場合	囲み文字で囲まれている場合	
NCHAR	−	終端から前方へ連続する空白を除いた値が入力データとなります。	空白を含めた値が入力データとなります。	「"」を囲み文字として扱います。「"」を除いた値が入力データとなります。
VARCHAR 及び MVARCHAR	enclose_del=yes	終端から前方へ連続する空白を除いた値が入力データとなります。データがすべて空白の場合は 1 バイトの空白データとなります。	空白を含めた値が入力データとなります。	「"」を囲み文字として扱います。「"」を除いた値が入力データとなります。連続する囲み文字の「""」は 0 長データとなります。
	enclose_del=no			「"」を囲み文字として扱いません。「"」を含めた値が入力データとなります。
NVARCHAR	−	終端から前方へ連続する空白を除いた値が入力データとなります。データがすべて空白の場合は 1 バイトの空白（奇数バイト）となり，入力データエラーとなります。	空白を含めた値が入力データとなります。	「"」を囲み文字として扱います。「"」を除いた値が入力データとなります。連続する囲み文字の「""」は 0 長データとなります。
DATE，INTERVAL YEAR TO DAY，TIME，INTERVAL HOUR TO SECOND，及び TIMESTAMP	−	先頭から後方へ連続する空白，及び終端から前方へ連続する空白を除いた値が入力データとなります。	入力データエラーとなります。	
BLOB	−	終端から前方へ連続する空白を除いた値が入力データとなります。	空白を含めた値が入力データとなります。	「"」を囲み文字として扱いません。「"」を含めた値が入力データとなります。
BINARY(7)	−	空白を含めた値が入力データとなります。		「"」を囲み文字として扱いません。「"」を含めた値が入力データとなります。
XML 型の BINARY 属性パラメタ（ファイルパスを指定する場合）	−	終端から前方へ連続する空白を除いた値がファイルパス名となります。	終端から前方へ連続する空白を除いた値がファイルパス名となります。囲み文字もファイルパス名の一部とみなされます。	「"」を囲み文字として扱いません。「"」を含めた値がファイルパス名となります。

（凡例）－：該当しません。

表 5–32　入力データを文字列形式で記述した場合の格納データ（固定長データ形式）

表の列のデータ型，又はコンストラクタ関数のパラメタのデータ型	列名文及び option 文の指定内容		入力データファイルの記述データ	格納データ
INTEGER，SMALLINT，DECIMAL，FLOAT，及び SMALLFLT	type=char(3)		1△△	各データ型の 1
			△△1	
			△1△	
			+△1	入力データエラー
			1△1	
	type=char(3)	option 文 allspace オペランドの指定あり	△△△	各データ型の「0」
		option 文 allspace オペランドの指定なし	△△△	入力データエラー
CHAR(7)及び MCHAR(7)	type=char(7)又は type=char(7),mode=text		ABC△△△△	ABC△△△△
			"ABC"△△	
			"ABC△△"	
			""ABC""	"ABC"△△
			△△△△△△△	△△△△△△△
			""△△△△△	
			△"△A△"△	△"△A△"△
	type=char(7),mode=bin		ABC△△△△	ABC△△△△
			"ABC"△△	"ABC"△△
			"ABC△△"	"ABC△△"
			""ABC""	""ABC""
			△△△△△△△	△△△△△△△
			""△△△△△	""△△△△△
			△"△A△"△	△"△A△"△
NCHAR(3)	type=char(6)		あ▲▲	あ▲▲
			"あ"▲	"あ"▲
			"あ▲"	あ▲▲
			""あ""	"あ"▲

表の列のデータ型，又はコンストラクタ関数のパラメタのデータ型	列名文及び option 文の指定内容	入力データファイルの記述データ	格納データ
		▲▲▲	▲▲▲
		""▲▲	""▲▲
VARCHAR(7)及び MVARCHAR(7)	type=char(7)又は type=char(7),enclose_del=yes	ABC△△△△	ABC（3 バイト）
		"ABC"△△	
		"ABC△△"	ABC△△（5 バイト）
		""ABC""	"ABC"（5 バイト）
		△△△△△△△	△（1 バイト）
		""△△△△△	（0 バイト）
		△"△A△"△	△"△A△"
	type=char(7),enclose_del=no	ABC△△△△	ABC（3 バイト）
		"ABC"△△	"ABC"（5 バイト）
		"ABC△△"	"ABC△△"（7 バイト）
		""ABC""	""ABC""（7 バイト）
		△△△△△△△	△（1 バイト）
		""△△△△△	""（2 バイト）
		△"△A△"△	△"△A△"△
NVARCHAR(3)	type=char(6)	あ▲▲	あ▲▲（6 バイト）
		"あ"▲	"あ"▲（6 バイト）
		"あ▲"	あ▲（4 バイト）
		""あ""	"あ"（4 バイト）
		▲▲▲	▲▲▲（6 バイト）
		""▲▲	""▲▲（6 バイト）
DATE	type=char(12)	2003-12-31△△	2003-12-31
		△△2003-12-31	
		△2003-12-31△	
INTERVAL YEAR TO DAY	type=char(11)	00010101.△△	+00010101.
		△△00010101.	
		△00010101.△	
TIME	type=char(10)	08:45:00△△	08:45:00

表の列のデータ型，又はコンストラクタ関数のパラメタのデータ型	列名文及び option 文の指定内容	入力データファイルの記述データ	格納データ
		△△08:45:00	
		△08:45:00△	
INTERVAL HOUR TO SECOND	type=char(9)	010101.△△	+010101.
		△△010101.	
		△010101.△	
TIMESTAMP	type=char(21)	1991-01-01△11:22:33△△	1991-01-01△11:22:33
		△△1991-01-01△11:22:33	
		△1991-01-01△11:22:33△	
BLOB	type=char(13)	c:¥pdload¥file01△△	c:¥pdload¥file01 のデータ
		△△c:¥pdload¥file01	エラー
		△c:¥pdload¥file01△	
BINARY(7)	type=char(7)	△△△△△△△	△△△△△△△

（凡例）

△：半角の空白（0x20）

▲：全角の空白

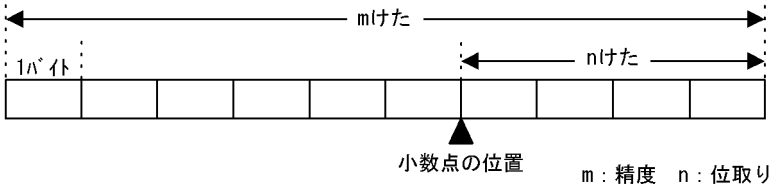
- 入力データファイルに予備列のデータを記述しないでください。入力データファイルに予備列のデータが記述された場合，pdload は，次のどちらかの動作をします。
  - エラーが発生し，論理エラー番号の 1，2，4，5，7，8，9 又は 10 を出力します。
    論理エラー番号については，「エラー情報ファイルの内容」を参照してください。
  - リターンコード 0 で正常終了します。
    ただし，KFPL00703-I メッセージで示した格納行数が，入力したデータ行数より多くなります。

(3)　固定長データ形式固有の入力データ型

固定長データ形式の入力データファイルは，表の列データ型と同じデータ形式，又は列データ型にデータ変換可能な文字列で作成しますが，DECIMAL 型の列に格納する入力データ型として，ADEC（ASCII CODE DECIMAL）型と BDEC（BINARY DECIMAL）型を使用できます。

(a)　ADEC 型（ASCII CODE DECIMAL）

ADEC 型は，文字列形式で数値データを記述します。データ中には小数点を持たないで，列構成情報ファイルで小数点の位置を指定するデータ型です。ADEC 型の形式を次に示します。



＜規則＞

- 使用できる文字を次に示します。

  ＋，−，空白（0x20），タブ（0x09），ナル文字（0x00），0〜9

  ただし，データ中に 1 バイト以上の空白，タブ，又はナル文字を含むと，小数点を持たない整数値と解釈します。そのため，type オペランドで指定した精度と位取りで小数点位置を解釈するデータを作成する場合は，「0」でパディングする必要があります。
- 符号（＋又は−）だけの入力データは，入力データエラーとなります。
- option 文の allspace オペランド指定ありの場合だけ，空白だけの入力データを許可します。この場合，0 が格納されます。allspace オペランド指定なしの場合は入力データエラーとなります。
- 入力データ中には小数点は記述しません。列構成情報ファイルの type オペランドで精度と位取りを指定します。精度(m)と位取り(n)の関係を次に示します。

  1≦m≦38　　0≦n≦38　　m≧n
- 入力データに符号（＋又は−）を持つ場合，精度(m)には符号部 1 バイトを含めたけた数を指定します。
- type オペランドで指定した精度と位取りは，格納する列の DECIMAL 型にデータ変換可能（けた落ちが発生しない）値を指定します。データ変換不可の場合，制御文エラーとなります。

＜データの格納例＞

次の場合のデータの格納例を次の表に示します。

- 格納する表の列定義：DECIMAL(10,4)
- 入力データ：ADEC 型
- 列構成情報ファイルの指定：type=adec(10,4)

表 5–33　ADEC 型で記述したデータを DECIMAL(10,4)に格納した場合の例

入力データファイルに記述したデータ	DECIMAL(10,4)に格納するデータ	説明
1△△△△△△△△△	+000001.0000	1 バイトでも空白を含むと，type オペランドに指定した精度と位取りは無視され，小数点がない値と解釈します。
△△△△△△△△△1	+000001.0000	
△△△△△1△△△△	+000001.0000	
△△△△△10000	+010000.0000	
000001△△△△	+000001.0000	

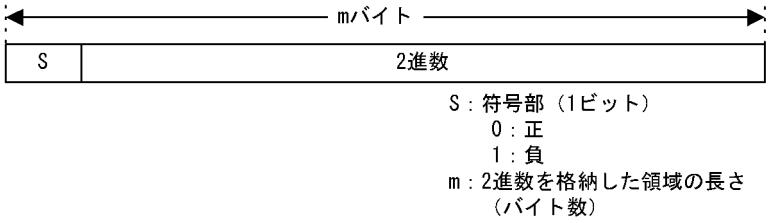
入力データファイルに記述したデータ	DECIMAL(10,4)に格納するデータ	説明
1000000000	+100000.0000	0～9 だけの場合，type オペランドに指定した精度と位取りで小数点の位置を決定します。
0000000001	+000000.0001	
0000010000	+000001.0000	
+000010000	+000001.0000	0～9 の値と符号部の場合，小数点の位置はデータの後方から解釈します。
-000010000	-000001.0000	

（凡例）

△：空白（0x20）

(b) BDEC 型（BINARY DECIMAL）

BDEC 型は，2 進数で記述されたバイナリデータを 10 進数で記述し，10 進数のけた数に合わせて，表の列データ型から小数点の位置を決定するデータ型です。BDEC 型は先頭 1 ビットを符号として扱います。形式を次に示します。



＜規則＞

- 列構成情報ファイルの type オペランドでバイナリデータの長さを指定します。
- BDEC 型のけた数(m)を次に示します。
  32 ビットモードの場合：1～4 バイト
  64 ビットモードの場合：1～8 バイト
- BDEC 型に実際に指定できる値を次に示します。
  32 ビットモードの場合：-2147483648～2147483647
  64 ビットモードの場合：-9223372036854775808～9223372036854775807
- DECIMAL 列に実際に格納できる値を次に示します。
  32 ビットモードの場合：精度は 10 けたまで
  64 ビットモードの場合：精度は 19 けたまで

＜データの格納例＞

次の場合のデータの格納例を次の表に示します。

- 入力データ：4 バイトの 2 進数
- 列構成情報ファイルの指定：type=bdec(4)

表 5–34　BDEC 型で記述したデータを DECIMAL 列に格納した場合の例

入力データファイルに記述したデータ	格納先の列定義	格納するデータ
0xff 0xff 0xfc 0xf7	DECIMAL(10,0)	-0000000777
	DECIMAL(10,2)	-00000007.77
	DECIMAL(10,10)	-0.0000000777

このように，同じ値でも格納先の DECIMAL 列の精度と位取りによって，格納するデータが異なります。

(4)　記述例

固定長データ形式のデータ記述例を次の図に示します。

図 5–17　固定長データ形式のデータ記述例

・テキストファイルの場合

```
ｎａｋａｍｕｒａ      10      北海道      00:00:00
ＫＡＷＡＧＵＣＨＩ     2      愛媛        13:01:27
ｏｄａ                -1      広島        06:33:14
ｔａｎａｋａ          36      神奈川      20:00:01
ｍａｓｕｔａｎｉ       0      大分        17:15:58
  CHARACTER          INTEGER   NCHAR       TIME
```

・バイナリファイルの場合

```
6e616b616d7572612020000036c0800796f6b6f68616d612020
ｎａｋａｍｕｒａ        ３６    ｙｏｋｏｈａｍａ
7361746f752020202020000020c0700736869627579612020 20
ｓａｔｏｕ              ２０    ｓｈｉｂｕｙａ
73757a756b69202020200000041c0500746f6b796f2020202020
ｓｕｚｕｋｉ            ４１    ｔｏｋｙｏ
  CHARACTER          DECIMAL   VARCHAR
```

5.5.4　繰返し列を含む表の配列データ形式

繰返し列を含む表の場合に，入力データファイルに記述する配列データの指定方法について説明します。

入力データファイルの形式は，入力データ形式(DAT 形式，バイナリ形式，固定長データ形式)，配列データ形式(FF,FV,VV)，及び可変長データ型かどうかによって異なります。

(1)　記述形式

繰返し列を含む表の場合の入力データファイルに記述する配列データの指定方法を次の表に示します。

表 5–35　繰返し列を含む表の場合の入力データファイルに記述する配列データの指定方法

内容	ファイル形式	配列データ形式		
		FF 形式	FV 形式	VV 形式
入力データ中に実要素数を指定するかどうか	−	指定しません。	指定します。	
実要素数の指定方法	−	−	DAT 形式の場合，文字で指定します。固定長データ形式及びバイナリ形式の場合，2 バイトの数値(16 進数)で指定します。	
入力データ中に指定する要素データの数	−	表定義で指定した要素数以下の数。詳細は「入力データ中の要素データ数と表定義で指定した要素数の関係」を参照してください。		実要素数分
DB に格納する要素数	−	ナル値の取り扱い方法によって異なります。詳細は「列のナル値と要素のナル値の扱い」を参照してください。	実要素数分	
入力データ形式と指定可否	DAT 形式	指定できます。	指定できません。	指定できます。
	バイナリ形式	可変長は指定できません。		指定できます。
	固定長データ形式	指定できます。		行データ全体が固定長のため指定できません。
ナル値の指定方法(要素としてのナル値)	DAT 形式	通常の列と同様に区切り文字の連続などで指定します。詳細は「DAT 形式の場合」を参照してください。	−	
	バイナリ形式	通常の列と同様にナル値・関数情報ファイルでナル値とみなす値を指定します。		
	固定長データ形式	通常の列と同様に列構成情報ファイルでナル値とみなす値を指定します。		−
ナル値の指定方法(列としてのナル値)	−	要素にすべてナル値を指定します。ただし，全要素にナル値を設定する場合と列としてナル値を設定する場合があります。詳細は「列のナル値と要素のナル値の扱い」を参照してください。	実要素数に 0 を指定します。	
DB に格納する要素数	−	ナル値の扱いの指定によって異なります。詳細は「列	指定された実要素数分のデータを格納します。	

内容	ファイル形式	配列データ形式		
		FF 形式	FV 形式	VV 形式
		のナル値と要素のナル値の扱い」を参照してください。		
可変長データ型の指定方法	DAT 形式	通常の列と同じです。	−	通常の列と同じです。
	バイナリ形式	指定できません。		有効データ長分指定します(埋め字なし)。
	固定長データ形式	表の定義長分指定し，有効データより後ろに埋め字をします。		−
DAT 形式ファイルの場合の要素データの区切り文字	−	列の区切り文字と同じです。	−	列の区切り文字と同じです。
要素データのエラーチェック	−	通常の列と同様に，データ型に応じたチェックをするとともに FF 形式及び VV 形式の場合，実要素数のチェックをします。また，FF 形式及び VV 形式の場合，有効要素データ以外の部分に対してデータを指定しますが，有効要素以降の部分についてはチェックしません。		

(凡例)−：該当しません。

(a)　DAT 形式の場合

DAT 形式ファイルには，FF 形式，又は VV 形式の配列データ形式で記述できます。

FF 形式のフォーマット：

指定要素数分を区切り文字で区切ってデータを記述します。列構成情報ファイルを省略，又は列構成情報ファイルで指定要素数を省略した場合，最大要素数が指定要素数になります。

VV 形式のフォーマット：

要素データの先頭に実要素数 0 ～最大要素数で指定し，実要素数分を区切り文字でデータを区切って記述します。

(b)　バイナリ形式の場合

バイナリ形式ファイルには，FF，FV，及び VV 形式の配列データ形式で記述できます。

FF 形式のフォーマット：

指定要素数分のデータを表「各データ型に対する列データ・パラメタの記述形式（バイナリ形式）」に示す形式で記述します。ナル値・関数情報ファイル自体又はナル値・関数情報ファイルで指定要素数を省略した場合，最大要素数が指定要素数になります。

FV 形式のフォーマット：

要素データの先頭に実要素数を 2 バイトの 2 進数で指定し，指定要素数分のデータを表「各データ型に対する列データ・パラメタの記述形式（バイナリ形式）」に示す形式で記述します。ナル値・関数情報

ファイル自体又はナル値・関数情報ファイルで指定要素数を省略した場合，最大要素数が指定要素数になります。

VV 形式のフォーマット：

要素データの先頭に実要素数を 2 バイトの 2 進数で指定し，実要素数分のデータを表「各データ型に対する列データ・パラメタの記述形式（バイナリ形式）」に示す形式で記述します。

(c)　固定長データ形式の場合

固定長データ形式ファイルには，FF 形式又は FV 形式の配列データ形式で記述できます。

列構成情報ファイル，及びナル値・関数情報ファイルを省略，又は列構成情報ファイル，及びナル値・関数情報ファイルで指定要素数を省略した場合，最大要素数が指定要素数になります。

なお，バイナリ形式でデータを記述した場合，VARCHAR，NVARCHAR，MVARCHAR で指定する各要素のデータ長が定義長に満たない場合は，定義長まで埋め字をする必要があります。

FF 形式のフォーマット：

指定要素数分のデータを表「各データ型に対する列データ・パラメタの記述形式（バイナリ形式）」に示す形式で記述します。

FV 形式のフォーマット：

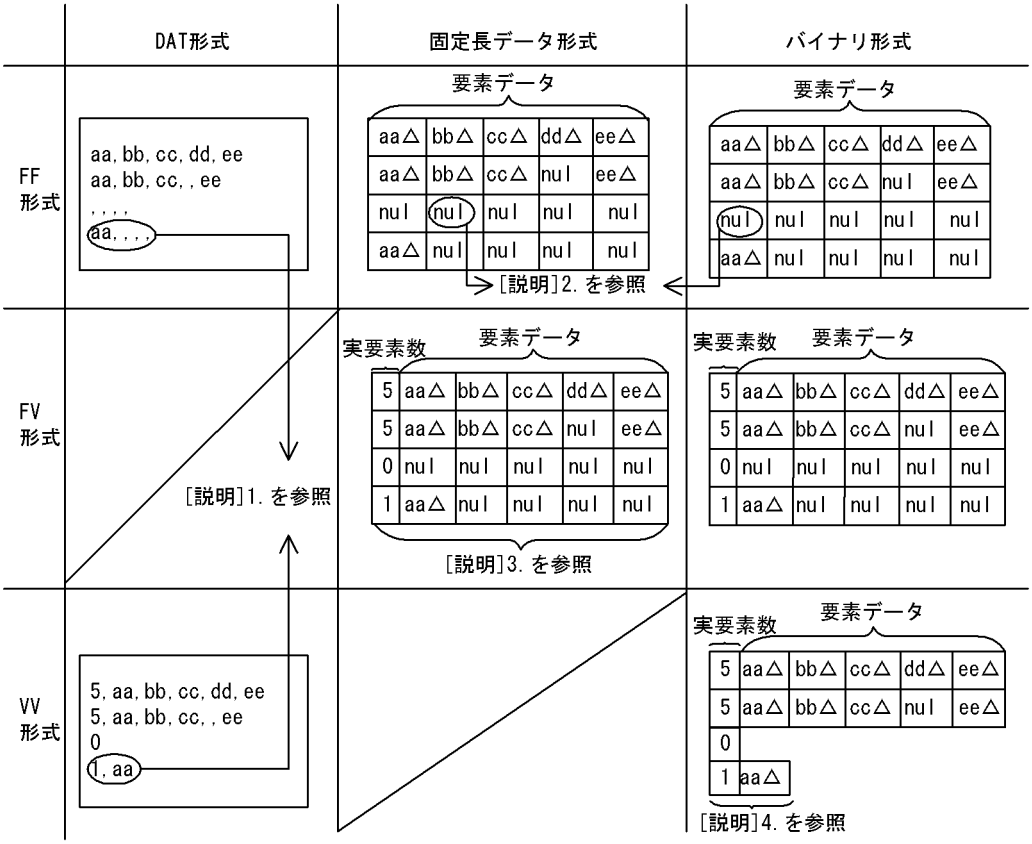
要素データの先頭に実要素数を 2 バイトの 2 進数で指定し，指定要素数分のデータを表「各データ型に対する列データ・パラメタの記述形式（バイナリ形式）」に示す形式で記述します。

(d)　固定長データ型の場合の配列データの指定例

固定長データ型の場合の配列データの指定例を次の図に示します。

なお，CHAR(3) ARRAY[5]の列の場合の例です。ナル値には３バイトの「nul」という文字を指定しています。

図 5-18　固定長データ型の場合の配列データの指定例



（凡例）△：スペースを表します。

注　DAT形式ファイルの場合は，入力データは１行ごとに改行しますが，固定長データ形式ファイル，及びバイナリ形式ファイルの場合は，行データを連続して指定します。

［説明］
１．DAT形式ファイルの場合，FF形式では，常に一定数の区切り文字を指定しますが，VV形式では実要素数分しか指定しません。
２．ナル値は，列構成情報ファイルやナル値情報ファイルで指定したナル値を指定します。
３．FV形式では，実要素数を指定します。ただし，指定するデータの数は固定です。
４．VV形式では，実要素数を指定し，指定した数だけ要素データを指定します。

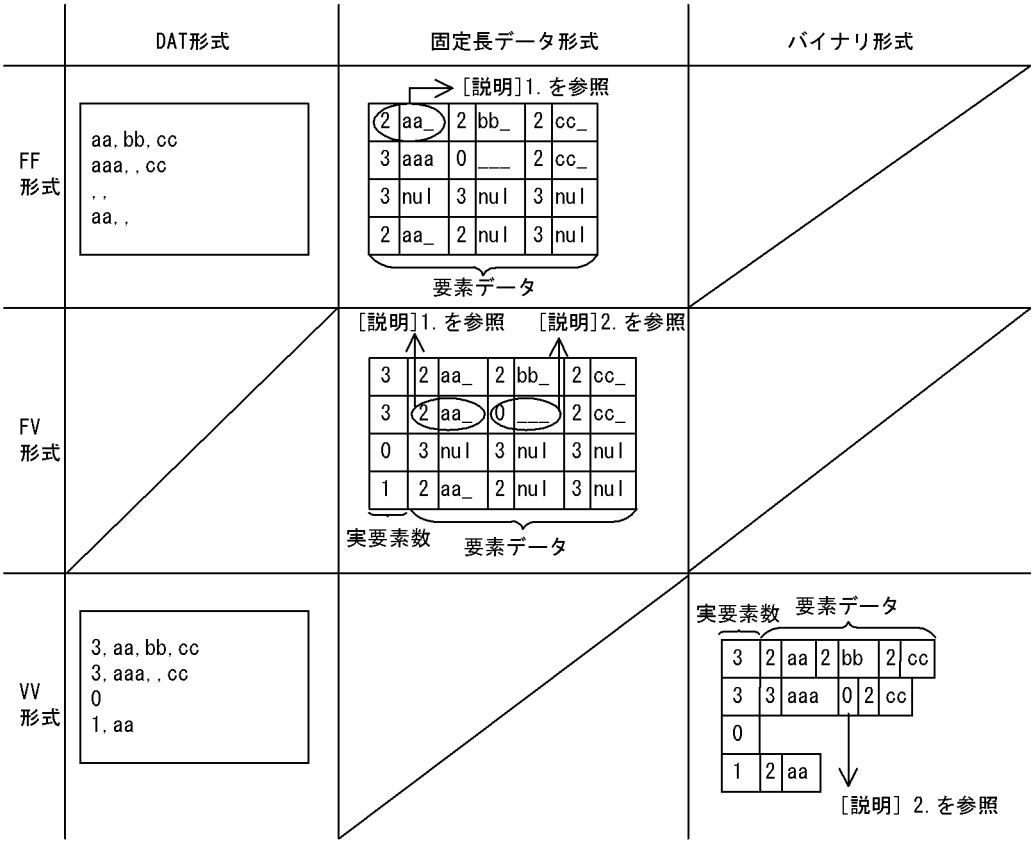
(e)　可変長データ型の場合の配列データの指定例

固定長データ形式，及びバイナリ形式の場合で，可変長データ型(VARCHAR，MVCHAR，NVARCHAR)のときは，FF 形式及び FV 形式ではデータは常に表の定義長分指定します。

有効なデータより後方には任意の埋め字をします。VV 形式の場合は埋め字をしません。

可変長データ型の場合の配列データの指定例を次の図に示します。なお，VARCHAR(3) ARRAY[3]の列の場合の例です。ナル値には３バイトの「nul」という文字を指定しています。

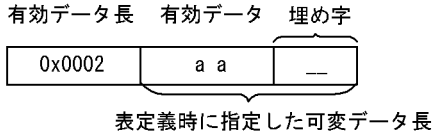
図 5-19　可変長データ型の場合の配列データの指定例



（凡例）_：埋め字を表します。

注　DAT形式ファイルの場合は，入力データは１行ごとに改行しますが，固定長データ形式ファイル，及びバイナリ形式ファイルの場合は，行データを連続して指定します。

［説明］
１．FF,FV形式の場合，可変長の要素データは，次に示すフォーマットで指定します。



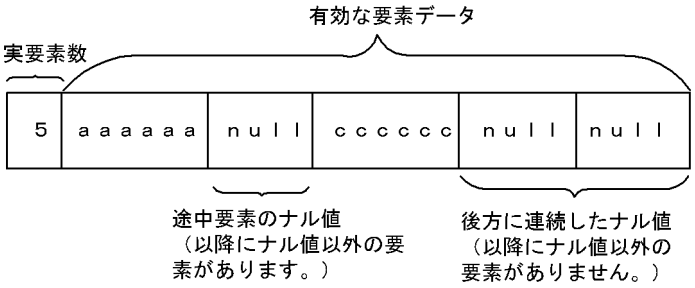
２．０バイトのVARCHARを指定します。ただし，VV形式の場合は埋め字をしません。

(2)　列のナル値と要素のナル値の扱い

繰返し列には列としてのナル値と，要素としてのナル値があります。データベース作成ユティリティでは，どちらもデータベースに格納できます。

繰返し列に対してデータロードする場合，入力データファイル中の要素のナル値の並び方には次に示す場合があります。繰返し列への入力データとナル値の種類を次の図に示します。

図 5–20　繰返し列への入力データとナル値の種類



この図に示すような場合，ナル値の扱いの指定(array 文のナル値オプション，列構成情報ファイル及びナル値・関数情報ファイルの nullset オペランドでの指定)によって，すべての要素データがナル値の場合，列としてナル値としたり，要素データの後方に連続したナル値を有効データとしないで，データベースに格納しない(要素数に含まない)こともできます。

繰返し列へデータロードする場合のデータベース作成ユティリティ（pdload）でのナル値の扱いについて次の表に示します。

表 5–36　繰返し列へデータロードする場合のデータベース作成ユティリティ（pdload）でのナル値の扱い

項目		ナル値の扱い	
		FF 形式	FV 形式及び VV 形式
ナル値の判定方法	DAT 形式	通常の列と同様(例えば，区切り文字が連続して入力された場合)に判定し，その要素をナル値とみなします。	
	バイナリ形式	通常の列と同様(ナル値とみなす値を列構成情報ファイルで指定します)に判定し，その要素をナル値とみなします。	
	固定長データ形式	通常の列と同様(ナル値とみなす値をナル値・関数情報ファイルで指定します)に判定し，その要素をナル値とみなします。	
ナル値のデータベースへの格納	列としてのナル値の指定方法	途中のナル値については，データベースに格納します。後方に連続したナル値，及び指定した要素がすべてナル値の場合は，「FF 形式でのナル値の扱い」を参照してください。	指定された要素数が 0 の場合，列としてナル値を格納します。要素データがすべてナル値でも，列としてのナル値とはしないで，実要素数分のナル値を設定します。
	要素としてのナル値の指定方法	途中のナル値については，データベースに格納します。後方に連続したナル値，及び指定した要素がすべてナル値の場合は，「FF 形式でのナル値の扱い」を参照してください。	途中のナル値，及び後方に連続したナル値のどちらも入力データ中に指定した実要素数の範囲内であれば，データベースにナル値を格納します。

(a)　FF 形式でのナル値の扱い

FF 形式の場合，入力データ中には実要素数を指定しません。

このため，後方に連続したナル値がある場合，次に示す指定によって無効な要素データとみなし，データベースに格納しないようにできます。

- 制御情報ファイル array 文の nullset オペランド
- 列構成情報ファイルの nullset オペランド
- ナル値・関数情報ファイルの nullset オペランド

なお，列構成情報ファイル及びナル値・関数情報ファイルでは列単位の指定，制御情報ファイルの array 文では表単位の指定です。三つとも指定した場合，列構成情報ファイル及びナル値・関数情報ファイルでの指定が優先されます。

また，nullset オペランドの指定を省略した場合は nullset=c が仮定されます。

ナル値オプション指定値とデータベースへの格納方法を次の表に示します。

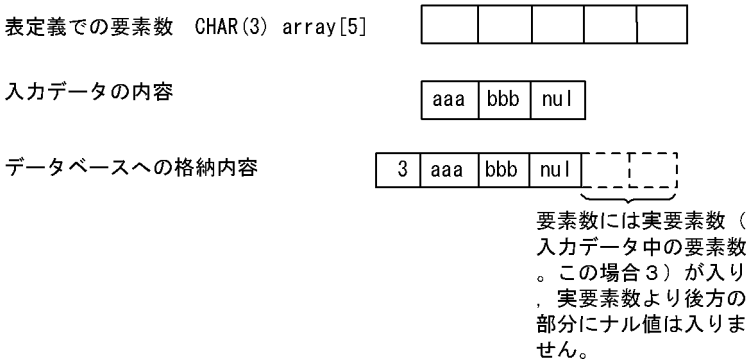
表 5–37　FF 形式でのナル値オプション指定値とナル値のデータベースへの格納方法

ナル値の種類	ナル値オプション指定値	
	c（デフォルト）	e
途中のナル値	該当する要素にナル値を格納します。	該当する要素にナル値を格納します。
後方の連続したナル値(以前に有効要素データが一つでもある場合)	該当する要素(後方の連続したナル値)は，無効データとみなし，データベースに格納しません。	該当する要素(後方の連続したナル値)にナル値を格納します。※
後方の連続したナル値(要素データがすべてナル値の場合)	列をナル値とします。	全要素にナル値を格納します。※

注

FF 形式の場合，入力データファイル中の要素データの数は，表定義で指定した要素数よりも少ないときがあります。

この場合，入力データファイル中の実要素数より後方の部分は，データが指定されていません。これは，ナル値ではなく，要素データがないということであるため，nullset=e が指定されてもナル値は格納しません。



注※

nullset=e を指定した場合，入力データファイルの要素データ数分のナル値がデータベースに格納されます。このため，データベースの格納効率を考慮すると，nullset=c を指定することをお勧めします。nullset=e を指定するのは，ナル値の数をそろえたい場合です。

(3)　入力データ中の要素データ数と表定義で指定した要素数の関係

FF 形式，及び FV 形式の場合，入力データ中に指定する要素データの数を表定義で指定した最大要素数以下の任意の個数にできます。

入力データ中に指定した要素データの個数（実要素数）は，次のように指定します。

DAT 形式，及び固定長データ形式の場合：

列構成情報ファイルの element オペランド

バイナリ形式の場合：

ナル値・関数情報ファイルの element オペランド

なお，入力データ中の要素数と，表定義で指定した要素数が同じ場合は指定不要です。この場合，表定義で指定した要素数が仮定されます。

入力データ中に指定した繰返し列の要素データ数と，表定義で指定した要素数が異なる場合のデータベースへの格納可否を次の表に示します。

表 5–38　入力データ中に指定した繰返し列の要素データ数と，表定義で指定した要素数が異なる場合のデータベースへの格納可否

入力データ中の繰返し要素データ数：表定義で指定した繰返し要素数	繰返し列への配列データ形式		
	FF 形式	FV 形式（データ中に指定した実要素数）	VV 形式（データ中に指定した実要素数）
入力データ要素数＞表定義要素数	エラー	エラー	エラー
入力データ要素数＝表定義要素数	正常にデータを格納	正常にデータを格納（入力データ中に指定した要素数分の要素データをデータベースに格納します)	正常にデータを格納
入力データ要素数＜表定義要素数	正常にデータを格納(element オペランドで指定した要素数分のデータを格納し，以降の要素は指定がないため，データベースに格納しません）	正常にデータを格納（入力データ中に指定した要素数分の要素データをデータベースに格納します)	正常にデータを格納

(4)　記述例

(a)　DAT 形式の場合

DAT 形式のデータ記述例を次の図に示します。

```
表定義：CREATE TABLE T1(C1 INTEGER,C2 CHAR(1) ARRAY[10])
```

図 5-21　DAT 形式のデータ記述例（繰返し列の場合）

入力データファイル
（FF形式，指定要素数７を指定，ナル値オプションを省略した場合）

```
1，a，b，c，d，e，f，g                                    1
2，a，b，c，d，e，，                                       2
3，a，b，，d，e，，g                                       3
4，，，，，，，                                            4
```

各行の１番目は，C1のデータ，繰返し列C2のデータは２番目から８番目までのデータです。指定要素数を指定しなかった場合は，最大要素数分（10個）のデータを指定する必要があります。

［説明］
1. １～７番目の要素に指定した値を格納します。
2. １～５番目の要素に指定した値を格納します。６番目，７番目の要素はナル値オプションが省略されているため，データベースには何も格納しません。
3. １～７番目の要素に指定した値を格納します。それ以降の要素には何も格納しません。３番目，６番目の要素にはナル値を格納します。
4. 指定要素すべてにナル値を指定した場合です。ナル値オプションを省略しているため，列にナル値を格納します。

入力データファイル（VV形式の場合）

```
1，7，a，b，c，d，e，f，g                                 1
2，5，a，b，c，d，e，，                                    2
3，7，a，b，，d，e，，g                                    3
4，7，，，，，，，                                         4
5，0                                                      5
```

各行の１番目は，C1のデータ，繰返し列C2のデータは２番目から８番目までのデータです。VV形式のため，C2のデータの先頭には実要素数を指定します。指定要素数を指定しなかった場合は，最大要素数分（10個）のデータを指定する必要があります。

［説明］
1. 実要素数が７のため，１～７番目の要素に指定した値を格納します。
2. 実要素数が５のため，１～５番目の要素に指定した値を格納します。
3. 実要素数が７のため，１～７番目の要素に指定した値を格納します。３番目，６番目の要素にはナル値を格納します。
4. 実要素数が７のため，１～７番目の要素にナル値を格納します。
5. 実要素数が０のため，列にナル値を格納します。

(b)　バイナリ形式の場合

バイナリ形式のデータ記述例を次の図に示します。

```
表定義：CREATE TABLE T1(C1 CHAR(4),C2 CHAR(10),C3 CHAR(5) ARRAY[10])
```

図 5–22　バイナリ形式のデータ記述例（繰返し列の場合）

入力データファイル（VV形式の場合の概念図）

C1(CHAR(4))	C1(CHAR(10))	C3 (CHAR(5) ARRAY[10])					
データ	データ	要素数	要素ﾃﾞｰﾀ1	要素ﾃﾞｰﾀ2	要素ﾃﾞｰﾀ3	要素ﾃﾞｰﾀ4	要素ﾃﾞｰﾀ5
0001	WWWWWWWWW	2	AAAAA	BBBBB			
0002	XXXXXXXXX	5	AAAAA	BBBBB	CCCCC	DDDDD	EEEEE
0003	YYYYYYYYY	0					
0004	ZZZZZZZZZ	4	AAAAA	BBBBB	CCCCC	DDDDD	

入力データファイル（VV形式，ナル比較値'EEEEE'の場合の実例）

```
303030315757575757575757570200414141414242424242
 0 0 0 1 W W W W W W W W W W  2  A A A A A B B B B B
3030303258585858585858585805004141414142424242434343434444444444454545454
 0 0 0 2 X X X X X X X X X X  5  A A A A A  B B B B B C C C C C D D D D D E E E E E
303030335959595959595959590000
 0 0 0 3 Y Y Y Y Y Y Y Y Y Y  0
303030345a5a5a5a5a5a5a5a5a040041414141414242424242434343434444444444
 0 0 0 4 Z Z Z Z Z Z Z Z Z Z  4  A A A A A B B B B B C C C C C D D D D D
```

1
1
2
2
3
3
4
4

CHAR(4)　　CHAR(10)　　CHAR(5) ARRAY[10]

注１　入力データの上段はデータを，下段は格納される値を示します。
注２　各行の先頭から４バイト目まではC1のデータ，５～14バイト目まではC2のデータ，繰返し列C3のデータは15バイト目以降のデータです。VV形式のため，最大要素数分（10個）まで指定できます。

［説明］
1. １，２番目の要素に指定した値を格納します。
2. １～５番目の要素に指定した値を格納します。ナル比較値が'EEEEE'のため，５番目の要素はナル値を格納します。
3. 実要素数に０を指定しているため，列にナル値を格納します。
4. １～４番目の要素に指定した値を格納します。

(c)　固定長データ形式の場合

固定長データ形式のデータ記述例を次の図に示します。

```
表定義：CREATE TABLE T1(C1 CHAR(4),C2 CHAR(10),C3 VARCHAR(5) ARRAY[10])
```

図 5–23　固定長データ形式のデータ記述例（繰返し列の場合）

入力データファイル（FV形式の場合の概念図）

C1(CHAR(4))	C1(CHAR(10))	C3 (VARCHAR(5) ARRAY[10])						
データ	データ	要素数	要素ﾃﾞｰﾀ1		要素ﾃﾞｰﾀ2		要素ﾃﾞｰﾀ3	
0001	WWWWWWWWW	2	3	AAA	5	BBBBB		
0002	XXXXXXXXX	3	2	AA	2	BB	4	CCCC
0003	YYYYYYYYY	1	4	AAAA				
0004	ZZZZZZZZZ	0						

入力データファイル（FV形式，指定要素数に３を指定，ナル比較値'CCCC'の場合の実例）

```
                                                                                  
303030315757575757575757570200030041414100000500424242424200000000000000        1
 0 0 0 1 W W W W W W W W W W  2  3  A A A    5  B B B B B                          1
303030325858585858585858580300020041410000000200424200000004004343434300        2
 0 0 0 2 X X X X X X X X X X  3  2  A A       2  B B       4  C C C C              2
303030335959595959595959590100040041414141000000000000000000000000000000        3
 0 0 0 3 Y Y Y Y Y Y Y Y Y Y  1  4  A A A A                                        3
303030345a5a5a5a5a5a5a5a5a5a000000000000000000000000000000000000000000000000    4
 0 0 0 4 Z Z Z Z Z Z Z Z Z Z  0                                                   4
```

CHAR(4)　CHAR(10)　VARCHAR(5) ARRAY[10]

注１　入力データの上段はデータを，下段は格納される値を示します。
注２　各行の先頭から４バイト目まではC1のデータ，５～14バイト目まではC2のデータ，繰返し列C3のデータは15バイト目以降のデータです。FV形式のため，指定要素数分（３個）まで指定する必要があります。指定要素数を省略すると，最大要素数分（10個）まで指定する必要があります。

［説明］

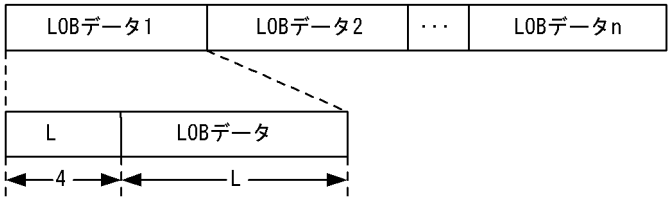
1. 実要素数２のため，１，２番目の要素に指定した値を格納します。１番目の要素はデータ長が定義長に満たないため，満たない分を０で埋め字をしています。また，実要素数２のため，３番目の要素分も０で埋め字をしています。
2. 実要素数３のため，１～３番目の要素に指定した値を格納します。ナル比較値が'CCCC'のため，３番目の要素はナル値を格納します。また，１，２，３番目の要素はデータ長が定義長に満たないため，満たない分を０で埋め字をしています。
3. 実要素数１のため，１番目の要素に指定した値を格納します。１番目の要素はデータ長が定義長に満たないため，満たない分を０で埋め字をしています。
4. 実要素数に０を指定しているため，列にナル値を格納します。

5.5.5　列単位 LOB 入力ファイル

(1)　列単位 LOB 入力ファイルの形式

列単位 LOB 入力ファイルの形式を次の図に示します。

図 5-24　列単位 LOB 入力ファイルの形式



（凡例）L：LOBデータ長（0≦L≦定義長）

図内の形式で LOB 列構成基表の入力データ順に入力行数分の LOB データを作成します。

格納する LOB データによっては，次の指定をする必要があります。

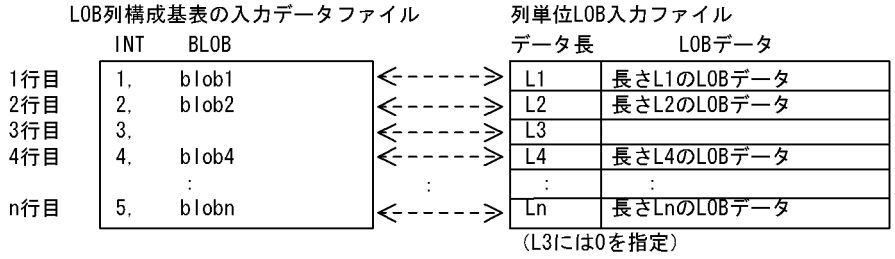
- **0 バイト以上の LOB データを格納する場合**
  図内の形式に従います。ただし，L=0 の場合は LOB データ長の 4 バイトだけを指定します。
- **デフォルト値，及びナル値を格納する場合**
  L=0 として，LOB データ長の 4 バイトだけを指定します（LOB データがナル値かどうかは，対応する LOB 列構成基表の行データの指定によって決まります）。

LOB 列構成基表の行データと列単位 LOB 入力ファイルの LOB データは，LOB データの指定順によって対応させます。すなわち，LOB 列構成基表の n 行目のデータは列単位 LOB 入力ファイルの n 件目のデータに対応します。

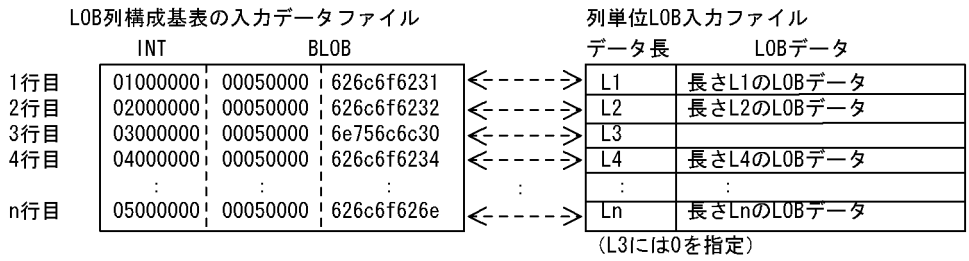
LOB 列構成基表の行データと列単位 LOB 入力ファイルの LOB データの対応例を次の図に示します。なお，例では 3 行目の LOB の値をナル値としています。

図 5-25　LOB 列構成基表の行データと列単位 LOB 入力ファイルの LOB データの対応例

・DAT形式の場合



・バイナリ形式の場合



ナル値情報ファイルの指定例

```
2,null=(1,c'null0')
```

(2) 列単位 LOB 入力ファイルに関する注意事項

(a) LOB 列構成基表の入力データファイルの指定内容と，列単位 LOB 入力ファイルの LOB データの対応

LOB 列構成基表の入力データファイルの指定内容と列単位 LOB 入力ファイルの対応する LOB データの指定内容が一致しない場合，エラーとなります。

エラーとなるケースを次に示します。

- **LOB 列構成基表の入力データファイルが DAT 形式の場合**

  LOB 列構成基表の入力データファイルの指定がナル値，又はデフォルト値の場合に，列単位 LOB 入力ファイルの対応する LOB データが 1 バイト以上指定しているときにエラーとなります。

- **LOB 列構成基表の入力データファイルがバイナリ形式の場合**

  LOB 列構成基表の入力データファイルの指定と，ナル値・関数情報ファイルの指定が一致しナル値とする場合に，列単位 LOB 入力ファイルの対応する LOB データが 1 バイト以上指定しているときはエラーとなります。

(b) LOB 列構成基表の入力データファイルのデータ件数と，列単位 LOB 入力ファイルのデータ件数に関する注意事項

列単位 LOB 入力ファイルを使用して LOB 列を作成する場合，LOB 列作成時に列単位 LOB 入力ファイル中のデータ件数をデータベース作成ユティリティが確認します。ただし，オプションの指定と入力データファイル中のエラーデータの有無によってデータ件数の確認をしない場合があります。データ件数の確認をしない場合は，KFPL16328-W メッセージが出力されます。

オプションの指定と入力データファイル中のエラーデータの有無の関係を次の表に示します。

表 5–39　オプションの指定と入力データファイル中のエラーデータの有無の関係

オプション		LOB 列構成基表の入力データファイル中のエラーデータ有無	
		なし	あり
-e オプションの指定	なし	○	○※1
	あり	○	×※2

（凡例）

○：列単位 LOB 入力ファイル中のデータ件数の確認をします。

×：列単位 LOB 入力ファイル中のデータ件数の確認をしません。

注※1

LOB 列構成基表作成時に，エラーにならなかった行に対応する入力データファイル中の LOB 列の情報を LOB 中間ファイルに出力し，その後列単位 LOB 入力ファイルのデータ件数の確認をします。

エラーデ−タファイルなどを使用して，エラーとなった LOB 列構成基表の行データを後から作成する場合は，その後から作成した LOB 列構成基表の入力デ−タファイルに対応する列単位 LOB 入力ファイルを準備する必要があります。最初に準備した列単位 LOB 入力ファイルはそのまま使用できません。

注※2

LOB 列構成基表の入力デ−タファイル中のエラーとなる前の行までの，対応する LOB 列の情報を LOB 中間ファイルに出力します。

このため，LOB 列構成基表の入力デ−タファイル中のエラーとなる前の行までの，LOB 列構成基表の入力デ−タファイルに対応する列単位 LOB 入力ファイル中の LOB データがずれている場合（不正なデータがあった場合，又は必要なデータがなかった場合なども含まれます）でも，そのまま LOB 列の情報を LOB 中間ファイルに出力します。

LOB 列構成基表の入力デ−タファイル中のエラーデータを修正後，エラーとなった行データ以降の LOB 列構成基表を-r オプション指定の追加モードで作成した場合，LOB 列構成基表の入力デ−タファイルの最終行まで作成した後，LOB 列作成時に初めて列単位 LOB 入力ファイルのデータ件数の確認をします。

データ件数の確認でエラーとなった場合，エラーの原因が列単位 LOB 入力ファイル中のどこにあるかによって，次のどれかの対策が必要となります。

(a)　エラーの原因が，作成しようとした列単位 LOB ファイル中のデータの範囲内にある場合

列単位 LOB 入力ファイル中の LOB データのエラー原因を取り除くことによって，LOB 列を作成できます。

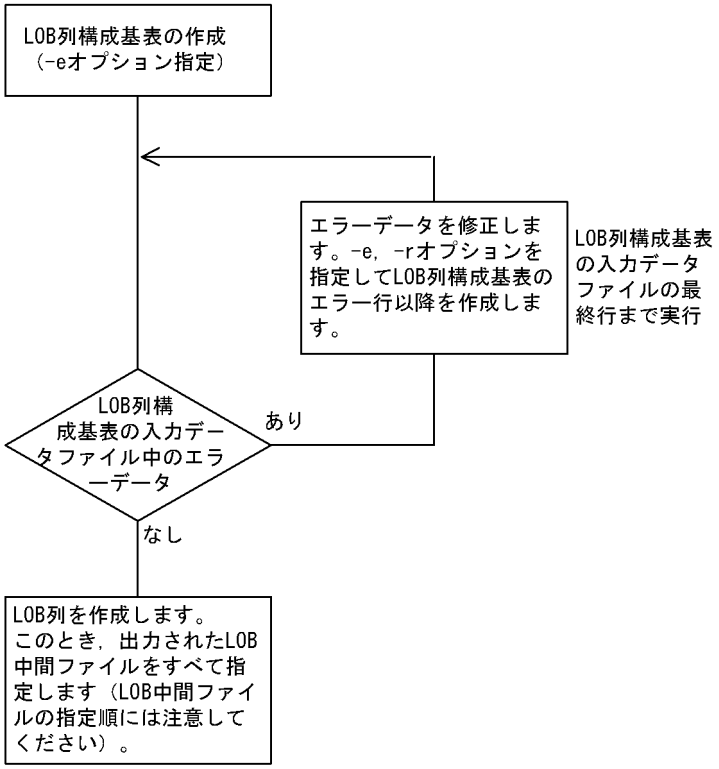
(b)　エラーの原因が，作成しようとした列単位 LOB 入力ファイル中のデータの範囲内にない場合

列単位 LOB 入力ファイルの LOB データのエラー原因を取り除き，作成しようとした列単位 LOB 入力ファイル中の LOB データだけでなく，既に作成されている LOB データも含めて作成する必要があります。

このような場合には，作成する LOB 列構成基表が分割表か非分割表かを確認し，更に LOB 列構成基表の作成単位が表単位か RD エリア単位かを確認します。そして，エラーとなったユーザ LOB 用 RD エリアだけ，又はエラーとなった LOB 列に対応するユーザ LOB 用 RD エリアすべてを，再初期化，又はバックアップからの回復をしてから，LOB 列を作成します。

これらの，列単位 LOB 入力ファイルのデータ件数エラー時の対策を容易にするための手順を，次の図に示します。

図 5–26　列単位 LOB 入力ファイルのデータ件数エラー時の対策を容易にするための手順



5.5.6　pdrorg 出力バイナリ形式ファイル

pdrorg 出力バイナリ形式ファイルを次に示します。

(1)　pdload 用アンロードファイル

pdrorg 実行時に-W bin オプションを指定して，pdload の入力データファイルとして使用するために出力したアンロードデータファイルです。このファイルを入力データファイルとして pdload を実行する場合，-W オプションと-b オプションを指定する必要があります。

(2)　pdrorg 用アンロードファイル

pdrorg 実行時に-W オプションを指定しないで出力したアンロードデータファイルです。このファイルを入力データファイルとして pdload を実行する場合，-U オプションを指定する必要があります。

なお，pdrorg 用アンロードファイルの場合，UOC を利用したデータロードはできません。

(3)　表移行用アンロードファイル

pdrorg 実行時に-W bin -w オプションを指定して，pdload の入力データファイルとして使用するために出力したアンロードデータファイルです。このファイルを入力データファイルとして pdload を実行する場合，-W オプション，-b オプション，及び-w オプションを指定する必要があります。

5.6　エラー情報ファイル

エラー情報ファイルには，入力データにエラーがあった場合にエラー情報が出力されます。

5.6.1　エラー情報ファイルの内容

エラー情報ファイルには，エラーとなった要因を示す論理エラー番号が出力されます。なお，論理エラー番号 XX は，KFPL310XX メッセージに対応しています。

エラー発生時の論理エラー番号とその内容を次の表に示します。

表 5–40　論理エラー番号とその内容

論理エラー番号	論理エラーの内容
1	NCHAR，NVARCHAR に入力するデータが不正です。 • 入力データ長が奇数です。
2	• パック形式 10 進数データの数値部又は符号部が不正です。 • 日付，時刻，時刻印データの数値部が不正です。又は，時刻印データの小数部の長さが不正です。 • 日間隔，時間隔データの数値部又は符号部が不正です。
3	可変長データ（VARCHAR，NVARCHAR，MVARCHAR，BLOB，及び BINARY）の長さ部が不正です。
4	データ変換エラーです。
5	• 入力データの列数が，格納する表の列数と一致しません。 • 引用符（"）で囲まれていないため，列が認識できません。 • テキスト（DAT 形式）ファイルのデータ中にナル文字（0x00）があります。
6	ナル値を禁止している表又は列に対して，ナル値を指定しています。 • FIX 属性の表にナル値を指定しています。 • クラスタキー構成列にナル値を指定しています。 • 非ナル値制約の列にナル値を指定しています。 • 主キー構成列にナル値を指定しています。
7	入力レコードがクラスタキー順に並んでいません。
8	クラスタキー又はインデクスを定義している表で，次のような場合にはエラーとなります。 • キーの重複禁止が指定されているが，キーが重複しています。
9	作成表（処理対象の表）の横分割条件と一致しない列値があります。
10	バイナリ形式の入力データの行長が不正です。
11	LOB データが不正です。 • LOB 列構成基表の入力データファイルで指定した，LOB 入力ファイルのファイル名の長さが不正です。 • 列単位 LOB 入力ファイル又はバイナリ形式ファイルで指定した LOB データ長が不正です。

論理エラー番号	論理エラーの内容
	• 列単位 LOB 入力ファイルでの，LOB データ長 4 バイト取得時，又は LOB データ長で指定した LOB データ取得時に，不正な EOF を検知しました。
12	指定した LOB 入力ファイルがありません。
13	LOB データの長さが定義長を超えています。
14	LOB 列構成基表の指定内容と LOB 入力ファイル中の指定内容が矛盾しています。 • ナル値指定に対して，1 バイト以上のデータを記述しています。 • デフォルト値に対して，1 バイト以上のデータを記述しています。
15	LOB 列構成基表と列単位 LOB 入力ファイルのデータ件数が一致していません。
16	コンストラクタ関数への入力パラメタが不正です。
17	LOB 入力ファイルに対するアクセス権限がありません。
18	抽象データ型に格納するデータフォーマットが不正です。
19	繰返し列に格納するデータの要素数が不正です。
20	不正な位置に囲み文字があります。
21	指定されたファイルパス名が不正です（入力データをファイルから読み込む場合）。 • 入力ファイルで指定したファイルパス名称長が不正です。
22	指定されたファイルが存在しません（入力データをファイルから読み込む場合）。又は，指定されたファイルがレギュラーファイルではありません（入力データをファイルから読み込む場合）。
23	指定されたファイルのデータ長が定義長を超えています（入力データをファイルから読み込む場合）。
24	指定されたファイルに対するアクセス権限がありません（入力データをファイルから読み込む場合）。
90	入力データ長が表の列定義長より大きいです。 lengover オペランド指定なしの場合： 列定義長を超えた部分を切り捨ててデータを格納します。 lengover オペランド指定ありの場合： 入力データエラーとして扱うため，データは格納しません。

注 1

-e オプションを指定しない場合，可変長データの長さが不正なときは指定した分の長さを不正データとみなし，該当する行データを読み飛ばして処理を続行します。

注 2

バイナリ形式の入力データファイル（データベース再編成ユティリティが出力した入力データファイルは除きます）がエラーの場合，最初にエラーとなった入力データのダンプイメージリストを 16 進形式及び文字形式（2 バイトのコードは無視されます）で出力します。なお，含まれるデータによって出力形式は次のようになります。

- BINARY 型の列，又は BINARY 属性の抽象データ型列のデータの場合

データ長が 1,024 バイトを超える場合，長さ部 4 バイトと先頭 1,024 バイトまでのデータ部を出力します。改行した後，先頭に「：」を出力して，次の行から続きの列のデータを出力します。出力例を次に示します。

```
 **** First error data [CODE:4] ****
41424344 00000001 61620001 0000001c    ABCD....ab......
ffffffff 00002000 00010002 00030004    ................
00050006 00070008 0009000a 000b000c    ................

                    ⋮

03fd03fe 03ff0400                      ................
   :
46364                                   ....cd
```

- BLOB 型の列のデータの場合

  出力対象外となります。

5.6.2　エラー情報ファイルの出力例

エラー情報ファイルの出力例を次に示します。

このダンプイメージリストでは，データ変換エラーとなった 10 行目の行データを出力しています。

```
pdload (03-00) **** DB LOAD **** 1996-05-07 12:34:56   single   ........1
KFPL31004-E Data conversion error occurred,
            line=10, column=4   ...................................2
KFPL31002-E Invalid pack format value,
            line=77, column=5   ...................................2
KFPL30000-I Server name=SDS   .....................................3
KFPL30001-I Rows successfully loaded=98   .........................4

**** First error data [CODE:4] ****   .............................5
41424344 00000001 61620001 0000001c  ABCD ... ab ..
ffffffff 6364                        ... cd   .....................6
```

[説明]

1. ヘッダ部
2. 入力データのエラーメッセージ
3. サーバ名
4. 格納行数
5. サブヘッダ（コードの番号は論理エラー番号）
6. 1 件目のエラーデータのダンプイメージリスト（16 進数，アスキー文字）

-e オプション指定有無によるエラー情報ファイルへの出力内容を次に示します。

-e オプションの指定	入力データエラー	エラー情報ファイルの出力内容
あり	なし	−
	あり	1〜6
なし	なし	1，3，4
	あり	1〜6

（凡例）

−：出力されません。

注

エラー情報ファイルの出力内容の番号は，エラー情報ファイルの出力例の番号と対応しています。

5.6.3　エラー情報参照時の注意事項

- データベース再編成ユティリティが出力した非 FIX 表のバイナリ形式の入力データファイルの場合は，エラーデータであってもエラー情報ファイルへのダンプイメージリスト出力，及びエラーデータファイル出力の対象にはなりません。
- バイナリ形式ファイルの場合，次に示す入力データエラーがあるとエラー該当列の直前までしか出力対象になりません。そのため，エラーデータファイルを参照する場合は注意してください。
  - 可変長文字列，又は BINARY 列にエラーがある場合
  - 入力データの列長が列定義長に満たない場合
- バイナリ形式ファイルの場合，エラー情報ファイルのダンプイメージリストは，LOB 列，抽象データ型の列，又は繰返し列がある表では出力されません。また，可変長文字列又は BINARY 列の長さ部が不正な場合は，該当する可変長文字列又は BINARY 列の直前のデータまでしか出力されません。
- バイナリ形式ファイル中に入力データの長さ部が負の値となるデータや，繰返し列の要素数が負の値となるデータがある場合，1 行分のデータとして認識できないため，-e オプション省略時は出力処理を打ち切り，-e オプション指定時はエラー情報ファイルを作成しません。この場合，KFPL24500-E メッセージを参照して対策してください。
- DAT 形式の入力データファイルの場合，source 文の maxreclen オペランドの指定値よりも大きいデータ（source 文の maxreclen オペランド省略時は，行長が 32 キロバイトを超えるデータ）は，エラーデータファイル出力の対象にはなりません。
- エラーデータファイル出力順序は，入力データファイル中のデータ順ではなく，エラー情報ファイルに出力されたエラーメッセージの順序に対応します。
- エラーデータファイルに出力されなかったデータがある場合，その件数に関係なく 1 回の警告メッセージが出力されます。
- 通信処理中，データ格納時にデータの論理エラー（キー重複エラー，又はプラグイン関数が検知したエラー）が発生した場合，入力データファイルのあるサーバ上にエラーデータファイル作成用一時ファイルが作成されます。作成されるファイル名は次のとおりです。

エラーデータファイル名＋データ格納サーバ名

ただし，パス名の長さが 1023 バイトを超える場合は，「データ格納サーバ名」は追加しないで，エラーデータファイル名の先頭を「データ格納サーバ名」で上書きし，パス名の長さが 1023 バイトを超えないように作成されます。

なお，このファイルは，エラーデータファイルと同じディレクトリ上に作成され，通信処理終了後に削除されます。

- 入力データファイルの形式が，ユーザが作成したバイナリ形式，又は固定長データ形式の場合，入力データエラーを検知すると，エラー情報ファイルに最初にエラーを検知した入力データのダンプイメージリストが出力されます。ただし，ダンプイメージリストを出力するために必要な作業領域（メモリ）が確保できない場合は，KFPL25222-W メッセージが出力されます。このとき，ダンプイメージリストは出力されないで，データロード処理が続行されます。

  入力データエラー検知時のダンプイメージリストの出力有無について次の表に示します。

表 5–41　入力データエラー検知時のダンプイメージリストの出力有無

入力データファイルの形式	ダンプイメージリストの出力	作業用のメモリの確保	
		確保できる	確保できない
DAT 形式	×	−	−
拡張 DAT 形式	×	−	−
固定長データ形式	○	ダンプイメージリストを出力し，データロード処理を続行します。	ダンプイメージリストは出力しないで，データロード処理を続行します。
バイナリ形式	○		
pdrorg 出力バイナリ形式	×	−	−

（凡例）

○：出力します。

×：出力しません。

−：該当しません。

5.6.4　source 文の errdata オペランド指定時の出力規則

source 文の errdata オペランドを指定した場合の，エラーデータファイル出力規則，及び該当データのエラー情報ファイルへのダンプイメージリスト出力規則を次に示します。

- HiRDB/パラレルサーバで source 文に記述したサーバ名と表格納 RD エリアがあるバックエンドサーバが異なる場合，HiRDB/パラレルサーバで表格納 RD エリアが複数のバックエンドサーバにある場合のファイル出力可否を次に示します。

エラーデータファイル作成用ワークバッファとの関連	入力データファイルの形式	入力データエラー種別	
		キー重複エラー又は抽象データ型に格納する値不正	左記以外のエラー
バッファが不足した場合，又はerrwork=0 の場合	DAT 形式	×	△
	固定長データ形式	×	○
	バイナリ形式	×	○
	pdrorg が出力した非 FIX 表，及び文字集合を指定した FIX 表のバイナリ形式の入力データファイル	×	×
バッファが十分な場合	DAT 形式	△	△
	固定長データ形式	△	○
	バイナリ形式	△	○
	pdrorg が出力した非 FIX 表，及び文字集合を指定した FIX 表のバイナリ形式の入力データファイル	×	×

（凡例）

○：エラーデータファイル，エラー情報ファイルのダンプイメージリストを出力します。

△：エラーデータファイルを出力し，エラー情報ファイルのダンプイメージリストは出力しません。

×：エラーデータファイル，エラー情報ファイルのダンプイメージリストを出力しません。

注

バッファが不足してエラーデータファイルに一部のエラーデータが出力されなかった場合は，エラー情報ファイルのエラーメッセージに(No output for error data)を付けて出力します。その場合は，errwork オペランドに十分なバッファ容量を指定して再実行すればエラーデータは出力されます。

- HiRDB/パラレルサーバで source 文に記述したサーバ名と表格納 RD エリアがあるバックエンドサーバが同じ場合のファイル出力可否を次に示します。

入力データファイルの形式	入力データエラー種別	
	キー重複エラー又は抽象データ型に格納する値不正	左記以外のエラー
DAT 形式	△	△
固定長データ形式	○	○
バイナリ形式	○	○
pdrorg が出力した非 FIX 表，及び文字集合を指定した FIX 表のバイナリ形式の入力データファイル	×	×

（凡例）

○：エラーデータファイル，エラー情報ファイルのダンプイメージリストを出力します。

△：エラーデータファイルを出力し，エラー情報ファイルのダンプイメージリストは出力しません。
×：エラーデータファイル，エラー情報ファイルのダンプイメージリストを出力しません。

5.7　列構成情報ファイル

列構成情報ファイルは，次のような場合に指定します。

項目	入力データファイルの形式			説明
	DAT 形式	固定長データ形式	pdload 用アンロードファイル	
入力データファイルのデータの列の並び順と，表の列の並び順が異なっているデータを格納する。	○	○	○	1
入力データファイルのデータの列数と，表の列数が異なっているデータを格納する。	○	○	○	2
入力データファイルのデータを，表の列のデータ型に変換して格納する。	−	○	×	3
表の列定義長と，入力データファイルの文字データ型の値が異なる場合，文字データ型を拡張して格納する。	−	○	○	3
ナル比較値と一致した入力データファイルのデータの列の値を，ナル値として格納する。	−	○	○	4
抽象データ型に格納する値を生成するコンストラクタ関数を指定する。	○	○	○	5
入力データファイルの繰返し列の要素数が，表定義時の最大要素数より少ない場合に指定する。	○	○	−※	6
順序数生成子から取得した順序番号を入力データとして格納する。	○	○	○	7

（凡例）

○：指定できます。

×：指定できません。

−：該当しません。

注※

配列データ形式が VV 形式の場合，要素数は指定できません。

[説明]

1. 入力データの並び順に対応する列名文を記述します。
2. 入力データ中に表に格納しないデータがある場合は，その対応する箇所にスキップデータ文を記述します。また，入力データ中に表に格納するデータがない場合は，列名文を省略します。
3. 表の列のデータ型を対応する列名文に記述します。
4. ナル値にしたい比較値（入力データ）を列名文に記述します。
5. コンストラクタ関数名，及びパラメタの型を列名文に記述します。抽象データ型に格納するデータを生成するコンストラクタ関数の名称が，列データ型の名称と異なる場合，又は同じ名称の関数が複数ある場合は，必ず指定します。固定長データ形式の場合は，入力パラメタのデータ型，及びナル比較値も記述します。

6. 入力データの要素数を列名文に記述します。列名文については「列名文」を，スキップデータ文については「スキップデータ文」を参照してください。
7. 使用する順序数生成子，及び入力データの置換条件を列名文に指定します。また，列に対応するデータが入力データ中にない場合は，追加データ文を指定します。

- **規則**

1. 列名文，及びスキップデータ文は 1 行で記述してください。ただし，1 行で書ききれない場合，指定の最後の文字がオペランドを区切るコンマ（,）のときには，そのコンマと継続を示す文字（¥）の間に 1 バイト以上の空白又はタブを記述して，行の末尾に継続文字を指定することで，複数行に分けて記述できます。
2. LOB 列構成基表と LOB 列を別々に作成する場合，LOB 列構成基表作成時に列構成情報ファイルを指定しているときには，LOB 列作成時にも同じ内容の列構成情報ファイルを指定する必要があります。指定しなかった場合，LOB 列作成時に不当にエラーとなることがあります。

5.7.1 列名文

(1) 形式

```
列名〔 { 〔, type=データ型
            〔, null= {’数値比較値’ | (先頭位置, {c|x} ’比較値’) } 〕〕
            〔, element=要素数〕〔, elmtype=配列データ形式〕
            〔, nullset=ナル値オプション〕〔, mode= {text | bin} 〕
            〔, enclose_del= {yes | no} 〕〔, filldata=埋め字データ〕
            〔, sequence=(〔認可識別子.〕順序数生成子識別子〔, replace=
              {’数値比較値’ | (先頭位置, {c|x} ’比較値’) | null | force} 〕)〕
       | 〔, func= (〔認可識別子.〕関数名
           , param=引数の型〔, type=データ型
            〔, null= {’数値比較値’|(先頭位置, {c|x} ’比較値’)} 〕〕
            〔, mode= {text | bin} 〕〔, enclose_del= {yes | no} 〕
            〔, filldata=埋め字データ〕
            〔, param=引数の型〔, type=データ型
              〔, null= {’数値比較値’|(先頭位置, {c|x} ’比較値’)} 〕〕
              〔, mode= {text | bin} 〕〔, enclose_del= {yes | no} 〕
              〔, filldata=埋め字データ〕〕…) 〕
        } 〕
         〔, filedir=パス名〕
```

列名文のオペランドは，入力データファイルの形式によって指定できるものとできないものがあります。形式ごとのオペランド指定可否を次に示します。

指定列の型	オペランド	入力データ形式		
		DAT 形式	固定長データ形式	pdrorg 出力バイナリ形式
既定義型	列名	◎	◎	◎

指定列の型	オペランド	入力データ形式		
		DAT 形式	固定長データ形式	pdrorg 出力バイナリ形式
（列指定）	type	×	◎	○※3※7
	null	×	○	○
	element	○	○	×
	elmtype	○	○	×
	nullset	○	○	×
	mode	×	○※4	×
	enclose_del	×	○※5	×
	filldata	×	○	×
	sequence	○	○	○
抽象データ型（関数指定）	列名	◎	◎	◎
	func	○※1	○※1	○※1
	param	○※2	○※2	○※2
	type	×	○※2	○※3
	null	×	○	○
	mode	×	○※4	×
	enclose_del	×	○※5	×
	filldata	×	○	×
	filedir	○※6	○※6	×

（凡例）

◎：必ず指定します。

○：指定できます。

×：指定できません。

注※1

関数指定時は，必ず指定します。

注※2

関数指定で func オペランド指定時は，必ず指定します。

注※3

列のデータ型，又は関数指定の param に指定したデータ型が，CHAR，MCHAR，NCHAR，BLOB，又は BINARY の場合に指定できます。次のどれかの条件に該当する場合は，必ず指定してください。

- BLOB 列のアンロードデータを，BINARY パラメタ又は BLOB パラメタがある抽象データ型の列へデータロードする場合
- 抽象データ型の BLOB パラメタのアンロードデータを，BLOB 型又は BINARY 型の列へデータロードする場合
- BLOB 列のアンロードデータを，BINARY 列へデータロードする場合
- 抽象データ型の BLOB パラメタのアンロードデータを，BINARY パラメタがある抽象データ型の列へデータロードする場合

注※4

type オペランドの指定値が char で，表の定義が CHAR 又は MCHAR の場合にだけ指定できます。

注※5

type オペランドの指定値が char で，表の定義が VARCHAR 又は MVARCHAR の場合にだけ指定できます。

注※6

列のデータ型が XML 型で，-K f オプションを指定しているときだけ指定できます。

注※7

列のデータ型の param に指定したデータ型が，CHAR，MCHAR，NCHAR，BLOB，又は BINARY の場合に指定できます。列のデータ形式が固定長である文字データ型を，データロード先の定義長に拡張してデータロードする場合は，必ず指定してください。

(2) オペランドの説明

(a) 列名

入力データを格納する表の列の名称を記述します。ただし，予備列の列の名称は記述できません。記述した場合，pdload は KFPL25103-E メッセージを出力して異常終了します。

列名を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，列名に空白が含まれる場合は，引用符で囲んでください。

(b) type=データ型

入力データの列のデータ型を記述します。データ型及び引数の型の記述方法と指定可否について次の表に示します。

表 5–42 データ型及び引数の型の記述方法と指定可否

記述方法	データ型の指定可否		記述規則
	固定長データ形式	pdload 用アンロードファイル	
integer	○	×	なし。
smallint	○	×	

記述方法	データ型の指定可否		記述規則
	固定長データ形式	pdload 用アンロードファイル	
char(n)※1※2	○	○	n には入力データのバイト数，m には入力データの文字数を指定します※6。 n の最大は CHAR，MCHAR の最大バイト数×2（ただし，格納対象列のデータ型が BINARY 型の場合は，BINARY 型の最大バイト数），m の最大は NCHAR のけた数で，格納先の表の列定義長と一致する必要はありません。また，数値型に変換する場合，文字列中にナル文字（¥ 0）は含めないでください。
nchar(m)※1※2	○	○	
mchar(n)※1※2	○	○	
varchar(n)	○	×	n は入力データのバイト数，m には入力データの文字数を指定します※6。 n の最大は VARCHAR，MVARCHAR の最大バイト数×2，m の最大は NVARCHAR の最大けた数で，格納先の表の列定義長と一致する必要はありません。
nvarchar(m)	○	×	
mvarchar(n)	○	×	
float	○	×	なし。
smallflt	○	×	
dec（精度［，位取り］）	○	×	なし。
date	○	×	なし。
interval-year-to-day	○	×	
time	○	×	
interval-hour-to-second	○	×	
yy-mm-dd※3	○	×	文字で記述された，西暦の下 2 けたのデータを DATE 型に格納するときだけ指定できます。年，月，日を区切る 1 バイトの文字は任意に指定できます（例：yy-mm-dd，yy/mm/dd など）。
yymmdd※3	○	×	符号なしパック形式で記述された，西暦の下 2 けたのデータを DATE 型に格納するときだけ指定できます。
timestamp(p)	○	×	p は小数秒部分のけた数です（0，2，4，又は 6）。
binary(n)	○	○※5	pdrorg 出力バイナリ形式の type オペランドに指定する場合は，(n)は省略します。 n は BINARY 型の最大バイト数です。
binaryprm	×	○	なし。
blob(n)	○	○※4	• 固定長データ形式の type オペランドに指定する場合

記述方法	データ型の指定可否		記述規則
	固定長データ形式	pdload 用アンロードファイル	
			n は LOB 入力ファイル名の最大長で，LOB 列の定義とは関係ありません。先頭 4 バイトに長さを持つバイナリ形式のときに指定します。 • pdrorg 出力バイナリ形式の type オペランドに指定する場合 (n)は省略します。
blobprm	×	○	なし。
adec（精度［，位取り］）	○	×	DECIMAL 列に格納するデータを文字列形式の 10 進数で記述する入力データ形式です。データ中には小数点を持たないため，type オペランドで小数点位置を精度と位取りの形で指定します。adec のデータ形式については，「ADEC 型（ASCII CODE DECIMAL）」を参照してください。
bdec(n)	○	×	DECIMAL 列に格納するデータをバイナリデータ（2 進数）で記述する入力データ形式です。n にはバイナリデータの長さを指定します。データ中には小数点を持たないため，列のデータ型に合わせて小数点位置を決定します。bdec のデータ形式については，「BDEC 型（BINARY DECIMAL）」を参照してください。

（凡例）

○：指定できます。

×：指定できません。

注※1

列値を左詰めで記述してください。

注※2

数値に変換して格納する場合には，左詰めでなくてもよいです。

注※3

西暦の下 2 けたのデータを，19yy に拡張して格納します。

注※4

対応する列のデータ型が BLOB 型の場合は指定できません。type 文を指定しないでください。

注※5

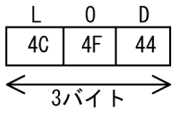
対応する列のデータ型が BINARY 型の場合は指定できません。type 文を指定しないでください。

注※6

指定例を次に示します。

入力データの文字コードが UTF-8 の場合

入力データの内容
例）文字列が”LOD”の場合

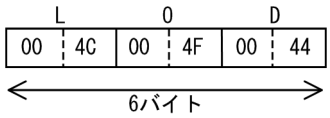


- 列構成情報ファイルの指定内容

```
C1,type=char(3)
```

入力データの文字コードが UTF-16 の場合

入力データの内容
例）文字列が”LOD”の場合



- 列構成情報ファイルの指定内容

```
C1,type=char(6)
```

列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否については，「列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否」を参照してください。

(c)　null=｛’数値比較値’｜（先頭位置，{c ｜ x}’比較値’)}

表の対応する列にナル値を格納する場合に指定します。

入力データと比較をするため，type オペランドに指定したデータ型に対応する比較値を指定してください。

入力データのデータ型によって，比較できる方法が異なります。入力データの各データ型の比較方法を次に示します。

入力データのデータ型	比較方法		
	数値比較	文字比較	16 進数比較
DECIMAL	○	×	○
DATE	○	×	○
INTERVAL YEAR TO DAY	○	×	○
TIME	○	×	○
INTERVAL HOUR TO SECOND	○	×	○
TIMESTAMP	○	×	○
CHAR	×	○	○

入力データのデータ型	比較方法		
	数値比較	文字比較	16 進数比較
VARCHAR	×	○	○
NCHAR	×	○	○
NVARCHAR	×	○	○
MCHAR	×	○	○
MVARCHAR	×	○	○
BINARY	×	○	○
BLOB	×	○	○
INTEGER	○	×	○
SMALLINT	○	×	○
FLOAT	○	×	○
SMALLFLT	○	×	○
yymmdd 形式	○	×	○
yy-mm-dd 形式	×	○	○
ADEC	×	○	○
BDEC	×	×	○

（凡例）

○：指定できます。

×：指定できません。

＜規則＞

1. 入力データと，このオペランドで指定した値が同じ列にはナル値を格納します。
2. 非ナル値制約の列，クラスタキー構成列，及び主キー構成列には指定できません。
3. pdrorg 出力バイナリ形式の入力データファイルの場合，LOB 列，及び BINARY 列のデータ，並びに抽象データ型の LOB，及び BINARY パラメタは，オフセットに 0 を設定することでナル値を指定するため，これらの列に対してこのオペランドは指定できません。なお，これらのデータ型が実データを持つ場合，オフセットに 0 を設定してナル値と扱うためには，対応する入力データ中の実データ部分を削除しておく必要があります（対応するオフセットを 0 に編集するだけでは，正しく処理がされません）。
4. 固定長データ形式の入力データファイルの場合，列の切り出し後に比較をするため，後方の空白を削除してからデータを比較します。

数値比較値　　～((255 文字以内))

入力データが数値属性の場合，比較する数値を記述します。

＜規則＞

1. 記述方法は DAT 形式での数値データと同じですが，空白は含めないでください（TIMESTAMP 型データの日時間の空白を除く）。
2. 数値属性とは，INTEGER，SMALLINT，FLOAT，SMALLFLT，DEC，DATE，TIME，INTERVAL YEAR TO DAY，INTERVAL HOUR TO SECOND，TIMESTAMP のデータ型，及び yymmdd 形式を示します。
3. TIMESTAMP 型の数値を指定する場合，小数部のけた数が定義の指定に満たないときは，0 が仮定されます。

＜記述例＞

- DATE の場合→yyyy-mm-dd の形式
  1995 年 1 月 1 日→1995-01-01
- TIME の場合→hh:mm:ss の形式
  12 時 1 分 1 秒→12:01:01
- INTERVAL YEAR TO DAY の場合→yyyymmdd.の形式
  1900 年間 1 か月 1 日→19000101.
- INTERVAL HOUR TO SECOND の場合→hhmmss.の形式
  12 時間 1 分 1 秒→120101.
- TIMESTAMP の場合→yyyy-mm-dd hh:mm:ss.nnnn の形式
  1900 年 1 月 1 日 1 時 10 分 30 秒 45
  →1900-01-01 01:10:30.4500 又は 1900-01-01 01:10:30.45

（先頭位置，{c ｜ x}’比較値’）

比較値が文字の場合は，入力データが文字属性，BINARY，又は BLOB のときに指定します。文字属性とは，CHAR，NCHAR，MCHAR，VARCHAR，NVARCHAR，及び MVARCHAR のデータ型，並びに yy-mm-dd 形式を示します。

比較値が 16 進数の場合は，入力データが数値属性でも比較できます（すべてのデータ型と比較できます）。

先頭位置　～＜符号なし整数＞((1～32000))

ナル値として扱うかどうかを判定する対象列の格納データの，比較開始先頭位置をバイト数で指定します。

指定した比較開始位置が次の条件に当てはまる場合，制御文エラーになります。

- DAT 形式の場合
  先頭位置に指定した比較開始位置＞ 32,000
- 固定長形式，及びバイナリ形式の場合
  先頭位置に指定した比較開始位置＞該当する列の定義長（各国文字列の場合は定義長×2）

指定した先頭位置以降の入力データと，比較値に指定した値を比較して，比較が完全一致した入力データをナル値として扱います。例を次に示します。

CHAR(5)の入力データ	null=(2,'XYZ')との比較結果	入力データの扱い
XYZ	比較値と完全一致します。	ナル値として扱います。
XYZ**	比較値は一致しますが，先頭位置から一致しません。	入力データとして扱います。
*XY**	比較値と一致しません。	

なお，option 文に charset オペランドを指定している場合，先頭位置に指定する比較開始位置は文字コードの文字長を考慮してください。

文字コードが UTF-8 の比較値'XYZ'と，文字コードが UTF-16 の入力データの 2 文字目以降を比較する場合，比較開始位置には 3 を指定する必要があります（null=(3,c'XYZ')など）。

c：

比較値が文字の場合に指定します。なお，入力データファイルを HiRDB の規定文字コード以外の文字コードで作成している場合でも，HiRDB の規定文字コード列を指定します（内部で pdload が入力データファイルの文字コードに変換します）。

指定した比較値が次の条件に当てはまる場合，制御文エラーになります。

- 比較値のバイト長＞ 255

  なお，比較値のバイト長は，入力データファイルの文字コードが HiRDB の既定文字コード以外の場合は，変換後の値です。
- 先頭位置に指定した比較開始位置＋比較値のバイト長−1 ＞該当する列の定義長（各国文字列の場合は定義長×2）

x：

比較値が 16 進数の場合に指定します。なお，入力データファイルを HiRDB の規定文字コード以外の文字コードで作成している場合，HiRDB の規定文字コード列で指定します。

指定した 16 進数の文字列（0〜f）を 2 文字で 1 バイトと扱います。

指定する文字は，大文字及び小文字のどちらでもかまいません。

指定した文字が 16 進数の文字列でない場合，又は指定した比較値が次の条件に当てはまる場合，制御文エラーになります。

- 比較値のバイト長＞ 254
- 先頭位置に指定した比較開始位置＋比較値のバイト長−1 ＞該当する列の定義長（各国文字列の場合は定義長×2）

数値型の列の場合，不当に比較値が入力データと同じになることがあります。この場合，16 進数で比較値を指定してください。指定例を次に示します。

列のデータ型	ナル値として扱う値	null=以降の指定方法
DECIMAL	「NULL」という文字列	(1,x'4e554c4c')
DATE	0000 年 00 月 00 日	(1,x'00000000')
INTERVAL YEAR TO DAY	先頭 1 バイトが 0xff	(1,x'ff')

列のデータ型	ナル値として扱う値	null=以降の指定方法
TIME	99 時 99 分 99 秒	(1,x'999999')
INTERVAL HOUR TO SECOND	2 バイト目が 0xff	(2,x'ff')
TIMESTAMP	2 バイト目と 3 バイト目が 0xff	(2,x'ffff')
INTEGER※	「NULL」という文字列	(1,x'4e554c4c')
SMALLINT※	「NL」という文字列	(1,x'4e4c')
FLOAT※	「**NULL**」という文字列	(1,x'2a2a4e554c4c2a2a')
SMALLFLT※	「NULL」という文字列	(1,x'4e554c4c')

注※

16 進数の比較値を指定しても，何らかの値と一致してしまう可能性があります（例えば，SMALLINT の指定例の場合，ビッグエンディアンの 20044 と一致します）。このため，ナル値として扱う値は，格納されない値にする必要があります。

比較値　～((255 文字以内))

入力データと比較する文字を指定します。

x を指定した場合，文字数は偶数で指定してください（最大指定長は 254 文字となります）。

(d)　element=要素数

～＜ 1～表定義時に該当する列に指定した要素数＞

繰返し列の場合に，入力データファイルに記述する要素数を指定します。

element オペランドは，表定義時に該当する列に指定した要素数よりも入力データファイルに指定する要素数が少ない場合に指定します。

element オペランド省略時は，表定義時に該当する列に指定した最大要素数が仮定されます。

＜規則＞

1. 配列データ形式が VV 形式の場合は，入力データファイルに指定する要素数が指定する要素の数になるため，このオペランドを省略しても無視します。
2. pdrorg 出力バイナリ形式の入力データファイルの場合はエラーとなります。

(e)　elmtype=配列データ形式

繰返し列の場合に，配列データ形式を指定します。

配列データ形式

次のどれかを指定します。

- ff：FF 形式

- fv：FV 形式
- vv：VV 形式

配列データ形式については，「繰返し列を含む表の配列データ形式」を参照してください。

＜規則＞

1. elmtype オペランドを省略した場合の配列データ形式は，制御情報ファイルの array 文を指定していないときは次の形式が仮定されます。
   - DAT 形式ファイル：FF 形式
   - バイナリ形式ファイル：VV 形式
   - 固定長データ形式ファイル：FF 形式
2. pdrorg 出力バイナリ形式の入力データファイルの場合はエラーとなります。

(f)　nullset=ナル値オプション

配列データ形式が FF 形式で，ある要素から指定要素の最後までナル値を指定した場合のナル値格納の扱いを指定します。

ナル値オプション

c：該当する要素に何も格納しません。

e：該当する要素にナル値を格納します。

＜規則＞

1. nullset オペランドを省略した場合の仮定値は次のようになります。
   - 制御情報ファイルの array 文を指定していない場合は c が仮定されます。
   - 制御情報ファイルの array 文を指定している場合は array 文の nullset オペランドの指定値が仮定されます。
2. pdrorg 出力バイナリ形式の入力データファイルの場合はエラーとなります。

(g)　mode= {text ｜ bin}

入力データが固定長文字列（type=char(n)）で，格納先の列データ型が CHAR，又は MCHAR の場合，入力データの格納方法を指定します。なお，このオペランドは，固定長データ形式の場合に指定できます。

＜適用基準＞

次の文字を入力データとして格納したい場合，bin を指定します。

- 入力データの後方から連続する，半角空白（0x20），タブ（¥t），及びナル文字（¥0）
- 入力データの両端の引用符（"）

入力データ中の，上記の文字を削除してかまわない場合は，text（省略時仮定値）を指定します。

text：

次の入力データの編集をして，データを格納する場合に指定します。

- 入力データの後方から連続する半角空白（0x20），タブ（¥t），及びナル文字（¥0）を削除します。
- 入力データの両端の引用符（"）を削除します。
- 入力データ長が格納列の定義長より短い場合，半角空白を埋めます。

bin：

次の入力データの編集だけをして，データを格納する場合に指定します。

- 入力データ長が格納列の定義長より短い場合，半角空白を埋めます。

なお，bin を指定した場合，空白変換はできません（option 文に spacelvl オペランドは指定できません）。

(h) enclose_del= {yes | no}

入力データが固定長文字列（type=char(n)）で，格納先の列データ型が VARCHAR，又は MVARCHAR の場合，入力データの格納方法を指定します。なお，このオペランドは，固定長データ形式の場合に指定できます。

＜適用基準＞

入力データの両端の引用符（"）を，入力データとして格納したい場合，no を指定します。

入力データの両端の引用符を削除してかまわない場合は，yes（省略時仮定値）を指定します。

yes：

次の入力データの編集をして，データを格納する場合に指定します。

- 入力データの後方から連続する半角空白（0x20），タブ（¥t），及びナル文字（¥0）を削除します。
- 入力データの両端の引用符（"）を削除します。

no：

次の入力データの編集だけをして，データを格納する場合に指定します。

- 入力データの後方から連続する半角空白（0x20），タブ（¥t），及びナル文字（¥0）を削除します。

なお，半角空白，タブ，及びナル文字を入力データとして格納したい場合は，引用符で囲んでください。

(i) filldata=埋め字データ

固定長データ形式の場合，BINARY 型のデータを CHAR 型で記述するときに，後方の埋め字データを記述します。埋め字データは，1 バイトのデータで，0～255 までの 16 進数表記（X'00'～X'ff'）で記述します。このオペランドを省略した場合は 0 が仮定されます。

＜規則＞

1. CHAR データから BINARY データに変換する場合，後方にある連続した埋め字を削除したデータを，BINARY データのデータ部として扱います。

2. 固定長データ形式以外の入力データファイルで指定した場合，及び BINARY 型以外の列に指定した場合は制御文エラーとなります。
3. BINARY 型パラメタ以外のパラメタに対して指定した場合，制御文エラーとなります。
4. BINARY 型パラメタに対して BINARY データを格納したファイルを指定する場合，埋め字として filldata，半角空白，タブ，及び¥0 を使用できます。ただし，option 文に charset オペランドを指定している場合，filldata は使用できません（指定しても無視されます）。

(j) sequence=(〔認可識別子.〕順序数生成子識別子〔，replace=｛'数値比較値'｜（先頭位置，{c ｜ x}'比較値'）｜ null ｜ force}〕)

順序数生成子から取得した順序番号を入力データとして格納する場合に指定します。

〔認可識別子.〕順序数生成子識別子

順序番号を取得する順序数生成子の識別子を指定します。

認可識別子を省略した場合は，pdload 実行ユーザ（-u オプション又は PDUSER 環境変数に設定した認可識別子）が仮定されます。

replace=｛'数値比較値'｜（先頭位置，{c ｜ x}'比較値'）｜ null ｜ force}

入力データを，順序数生成子から取得した順序番号で置き換える場合，その条件を指定します。

入力データとこのオペランドの指定値が一致した場合，該当する列のデータを順序数生成子から取得した順序番号に置き換えます。入力データのデータ型で値を比較するため，比較値は type オペランドに指定したデータの属性に合わせて指定してください。

'数値比較値'

入力データが数値属性の場合に，比較する数値を指定します。

指定方法については，null オペランドの説明を参照してください。

（先頭位置，{c ｜ x}'比較値'）

入力データが文字属性の場合に，比較する文字列を先頭位置と比較値で指定します。

指定方法については，null オペランドの説明を参照してください。

null

入力データの値が null の場合に順序番号で置き換えます。入力データファイルの形式が DAT 形式又は拡張 DAT 形式の場合に指定できます。

force

無条件に置き換えます。

＜規則＞

1. pdparaload を使用する場合，又は RD エリア単位にデータロードする場合は，分割キーの列に対してこのオペランドを指定しないでください。指定すると，分割キーの値が順序数生成子から取得した順序番号になるため，該当する分割キーの値が指定した RD エリアへの格納対象外の値となり，データを格納できないおそれがあります。

(k)　filedir=パス名

-K f オプションを指定して，該当する XML 型の列のパラメタ属性データを，1 データ 1 ファイルとして読み込む場合に，パラメタ属性のデータファイルを格納しているディレクトリを指定します。

＜規則＞

1. ここで指定したパス名と，入力データファイル中のファイルパス名を組み合わせて，パラメタ属性データとして読み込むファイルのファイルパス名とみなします。組み合わせたファイルパス名が 1,023 バイト以下となるように指定してください。
2. XML 型の列がない場合にこのオペランドを指定すると，エラーになります。
3. このオペランドを省略する場合は，入力データファイルに，パラメタ属性データとして読み込むファイルの絶対パスを記述してください。
4. このオペランドで，列ごとに格納ディレクトリへの絶対パスを指定する場合，入力データファイルには，このオペランドで指定したディレクトリからの相対のファイルパス名だけを記述してください。

＜記述例＞

列 2 に XML 型を定義している場合の，DAT 形式の入力データファイルの記述例を示します。

- 入力データファイルのデータ

```
 列1    列2         列3
00001,xml001.txt,2007-10-01
00002,xml002.txt,2007-10-02
```

- 列構成情報ファイルの記述

```
列1
列2,filedir=/user/xml_data
列3
```

- pdload が参照するファイル

```
/user/xml_data/xml001.txt
/user/xml_data/xml002.txt
```

(l)　func=（〔認可識別子.〕関数名，param=引数の型〔，type=データ型〔，null=｛'数値比較値'｜（先頭位置，{c｜x}'比較値'）｝〕〕〔，mode=｛text｜bin｝〕〔，enclose_del=｛yes｜no｝〕〔，filldata=埋め字データ〕〔，...〕）

指定する列名のデータ型が抽象データ型の場合に，格納する値を生成するコンストラクタ関数の情報を記述します。

列構成情報ファイルに指定した入力パラメタのデータ型と，呼び出すコンストラクタ関数の引数の型が異なる場合のデータベースへの格納可否については，「列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否」を参照してください。

〔認可識別子.〕関数名

格納するデータをプラグインの提供するコンストラクタ関数を呼び出して生成する場合に，その関数名称を指定します。

関数の戻り値が格納対象の列の抽象データ型と異なる関数，及び引数に抽象データ型を持つ関数は指定できません。

認可識別子を省略した場合は，抽象データ型を定義したユーザが仮定されます。

param=引数の型

コンストラクタ関数に渡すパラメタのデータ型を指定します。

入力となるパラメタが複数ある場合は，関数の入力形式に合わせた順序で指定します。

コンストラクタ関数に渡すパラメタのデータ型の記述方法及びデータ型の指定可否について次の表に示します。

表 5–43　コンストラクタ関数に渡すパラメタのデータ型の記述方法及びデータ型の指定可否

パラメタのデータ型		データ型の記述方法	データ型の指定可否		
分類	データ型		DAT 形式	固定長データ形式	pdload 用アンロードファイル
数データ	INTEGER	integer	○	○	○
	SMALLINT	smallint	○	○	○
	DECIMAL	dec()	○	○	○
		adec()	○	○	○
		bdec()	○	○	○
	FLOAT	float	○	○	○
	SMALLFLT	smallflt	○	○	○
文字データ	CHARACTER	char	○	○	○
	VARCHAR	varchar	○	○	○
各国文字データ	NCHAR	nchar	○	○	○
	NVARCHAR	nvarchar	○	○	○
混在文字データ	MCHAR	mchar	○	○	○
	MVARCHAR	mvarchar	○	○	○
日付データ	DATE	date	○	○	○
		yy-mm-dd	×	×	×
		yymmdd	×	×	×
時刻データ	TIME	time	○	○	○
時刻印データ	TIMESTAMP	timestamp	○	○	○

パラメタのデータ型		データ型の記述方法	データ型の指定可否		
分類	データ型		DAT 形式	固定長データ形式	pdload 用アンロードファイル
日間隔データ	INTERVAL YEAR TO DAY	interval-year-to-day	○	○	○
時間隔データ	INTERVAL HOUR TO SECOND	interval-hour-to-second	○	○	○
長大データ	BLOB	blob	○	○	○
		blobprm	×	×	×
バイナリデータ	BINARY	binary	○	○	○
		binaryprm	×	×	×

（凡例）

○：指定できます。

×：指定できません。

type=データ型

入力データファイルが固定長データ形式の場合に，入力となるパラメタのデータ型を指定します。

入力となるパラメタが複数ある場合は，関数の入力形式に合わせた順序で指定します。

データ型の記述方法及びデータ型の指定可否について次の表に示します。

表 5–44　データ型の記述方法及びデータ型の指定可否

分類	データ型	データ型の記述方法	データ型の指定可否	
			固定長データ形式	pdload 用アンロードファイル
数データ	INTEGER	integer	○	×
	SMALLINT	smallint	○	×
	DECIMAL	dec（精度［，位取り］）	○	×
		adec（精度［，位取り］）	○	×
		bdec(n)	○	×
	FLOAT	float	○	×
	SMALLFLT	smallflt	○	×
文字データ	CHARACTER	char(n)	○	×
	VARCHAR	varchar(n)	○	×
各国文字データ	NCHAR	nchar(m)	○	×
	NVARCHAR	nvarchar(m)	○	×

分類	データ型	データ型の記述方法	データ型の指定可否	
			固定長データ形式	pdload 用アンロードファイル
混在文字データ	MCHAR	mchar(n)	○	×
	MVARCHAR	mvarchar(n)	○	×
日付データ	DATE	date	○	×
		yy-mm-dd	×	×
		yymmdd	×	×
時刻データ	TIME	time	○	×
時刻印データ	TIMESTAMP〔(p)〕	timestamp(p)	○	×
日間隔データ	INTERVAL YEAR TO DAY	interval-year-to-day	○	×
時間隔データ	INTERVAL HOUR TO SECOND	interval-hour-to-second	○	×
長大データ	BLOB	blob(n)※	○	○
		blobprm	×	○
バイナリデータ	BINARY	binary(n)※	○	○
		binaryprm	×	○

（凡例）

○：指定できます。

×：指定できません。

注※

pdload 用アンロードファイルの場合は，(n)を省略できます。

null=｛'数値比較値'｜（先頭位置，{c ｜ x}'比較値'）｝

入力データファイルが固定長データ形式の場合に，入力となるパラメタをナル値としたい場合に指定します。

比較は入力パラメタのデータ型となります。指定方法については，null オペランドの説明を参照してください。

mode=｛text ｜ bin｝

指定方法については，mode オペランドの説明を参照してください。

enclose_del=｛yes ｜ no｝

指定方法については，enclose_del オペランドの説明を参照してください。

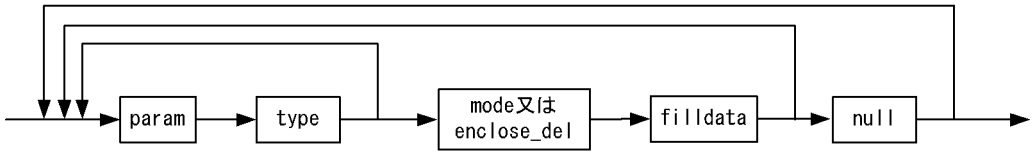
filldata=埋め字データ

指定方法については，filldata オペランドの説明を参照してください。

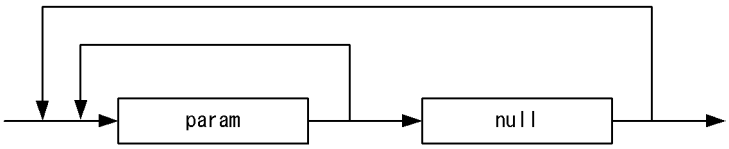
＜func オペランド内の指定順序＞

func オペランド内に指定する，param，type，null，filldata，mode，及び enclose_del の指定順序を次に示します

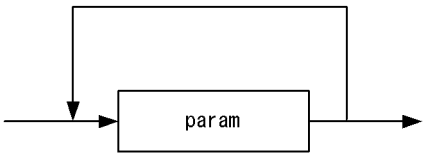
- 固定長データ形式の場合



- バイナリ形式の場合



- DAT 形式の場合



(3) 列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否

列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否を，次の表に示します。

表 5–45 列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否（1/2）

列構成情報ファイルに指定したデータ型	表のデータ型									
	INTEGER	SMALLINT	CHAR	NCHAR	MCHAR	VARCHAR	NVARCHAR	MVARCHAR	FLOAT	SMALLFLT
integer	○	×	×	×	×	×	×	×	×	×
smallint	×	○	×	×	×	×	×	×	×	×
char	○	○	○※1	○	○	○	○	○	○	○
nchar	×	×	×	○※1	×	×	×	×	×	×
mchar	×	×	×	×	○※1	×	×	×	×	×
varchar	×	×	×	×	×	○※2	×	×	×	×

列構成情報ファイルに指定したデータ型	表のデータ型									
	INTEGER	SMALLINT	CHAR	NCHAR	MCHAR	VARCHAR	NVARCHAR	MVARCHAR	FLOAT	SMALLFLT
nvarchar	×	×	×	×	×	×	○	×	×	×
mvarchar	×	×	×	×	×	×	×	○※2	×	×
float	×	×	×	×	×	×	×	×	○	×
smallflt	×	×	×	×	×	×	×	×	×	○
dec	×	×	×	×	×	×	×	×	×	×
date	×	×	×	×	×	×	×	×	×	×
interval-year-to-day	×	×	×	×	×	×	×	×	×	×
time	×	×	×	×	×	×	×	×	×	×
interval-hour-to-second	×	×	×	×	×	×	×	×	×	×
timestamp	×	×	×	×	×	×	×	×	×	×
binary	×	×	×	×	×	×	×	×	×	×
yy-mm-dd	×	×	×	×	×	×	×	×	×	×
yymmdd	×	×	×	×	×	×	×	×	×	×
blob	×	×	×	×	×	×	×	×	×	×
adec	×	×	×	×	×	×	×	×	×	×
bdec	×	×	×	×	×	×	×	×	×	×

（凡例）

○：格納できます。

×：格納できません。

注※1

固定長の文字データ型の入力データ長と格納先の表の定義長に応じて，次のようにデータが格納されます。

項番	格納先列への文字集合の指定	n と m の関係	データの格納方法
1	なし	n ＝ m	そのまま DB に格納します。
2		n ＞ m	エラーになります。
3		n ＜ m	（n ＋ 1）以降を空白文字で拡張して格納します。
4	あり	n'＝ m	そのまま DB に格納します。

項番	格納先列への文字集合の指定	n と m の関係	データの格納方法
5		n'＞ m	文字コード変換後の入力データを，格納先の定義長に切り詰めてから DB に格納します。詳細は，表「入力データ長，列定義長，及び lengover オペランドの関係」を参照してください。
6		n'＜ m	（n'＋ 1）以降を空白文字で拡張して格納します。

注※2

可変長の文字データ型の入力データ長と格納先の表の定義長に応じて，次のようにデータが格納されます。

項番	格納先列への文字集合の指定	n と m の関係	データの格納方法
1	なし	n≦m	そのまま DB に格納します。
2		n ＞ m	入力データを格納先の定義長に切り詰めてから DB に格納します。詳細は，表「入力データ長，列定義長，及び lengover オペランドの関係」を参照してください。
3	あり	n'≦m	そのまま DB に格納します。
4		n'＞ m	文字コード変換後の入力データを，格納先の定義長に切り詰めてから DB に格納します。詳細は，表「入力データ長，列定義長，及び lengover オペランドの関係」を参照してください。

（凡例）

n：入力データ長

m：格納先表の定義長

n'：文字コード変換後のデータ長

表 5–46　列構成情報ファイルに指定したデータ型と，表のデータ型が異なる場合のデータベースへの格納可否（2/2）

列構成情報ファイルに指定したデータ型	表のデータ型							
	DECIMAL	DATE	INTERVAL YEAR TO DAY	TIME	INTERVAL HOUR TO SECOND	TIMESTAMP	BINARY	BLOB
integer	×	×	×	×	×	×	×	×
smallint	×	×	×	×	×	×	×	×
char	○	○	○	○	○	○	○	○
nchar	×	×	×	×	×	×	×	×
mchar	×	×	×	×	×	×	×	×

列構成情報ファイルに指定したデータ型	表のデータ型							
	DECIMAL	DATE	INTERVAL YEAR TO DAY	TIME	INTERVAL HOUR TO SECOND	TIMESTAMP	BINARY	BLOB
varchar	×	×	×	×	×	×	×	×
nvarchar	×	×	×	×	×	×	×	×
mvarchar	×	×	×	×	×	×	×	×
float	×	×	×	×	×	×	×	×
smallflt	×	×	×	×	×	×	×	×
dec	○	×	×	×	×	×	×	×
date	×	○	×	×	×	×	×	×
interval-year-to-day	×	×	○	×	×	×	×	×
time	×	×	×	○	×	×	×	×
interval-hour-to-second	×	×	×	×	○	×	×	×
timestamp	×	×	×	×	×	○	×	×
binary	×	×	×	×	×	×	○	×
yy-mm-dd	×	○	×	×	×	×	×	×
yymmdd	×	○	×	×	×	×	×	×
blob	×	×	×	×	×	×	○※	○
blobprm	×	×	×	×	×	×	○※	○※
adec	○	×	×	×	×	×	×	×
bdec	○	×	×	×	×	×	×	×

（凡例）

○：格納できます。

×：格納できません。

注※

pdrorg 出力バイナリ形式の入力データファイルの場合，格納できます。

(4) pdrorg 出力バイナリ形式の入力データファイルに対する列名文の type オペランドの指定例

pdrorg 出力バイナリ形式の入力データファイルに対する列名文の type オペランドの指定例を次に示します。

入力データ型	変換後データ型			
	BLOB 型	BLOB パラメタ	BINARY 型	BINARY パラメタ
BLOB 型	−	func=(F_BLOB, param=blob)	type=blob	func=(F_BIN, param=binary, type=blob)
BLOB パラメタ	type=blobprm	func=(F_BLOB, param=blob)	type=blobprm	func=(F_BIN, param=binary, type=blobprm)
BINARY 型	×	×	−	×
BINARY パラメタ	×	×	×	func=(F_BIN, param=binary)

（凡例）

−：指定する必要はありません。

×：該当する入力データ型は，変換後データ型に変換できません。

F_BIN：
　BINARY 型をパラメタとする，プラグインが提供する抽象データ型のコンストラクタ関数です。

F_BLOB：
　BLOB 型をパラメタとする，プラグインが提供する抽象データ型のコンストラクタ関数です。

5.7.2　スキップデータ文

(1)　形式

```
＊skipdata＊　〔 {, length=バイト数 | , type=データ型 〔, element=要素数〕 } 〕
```

(2)　オペランドの説明

(a)　length=バイト数

～<符号なし整数>((1～32000))

固定長データ形式の入力データファイルの場合に，スキップする列のバイト数を指定します。

＜規則＞

1. DAT 形式の場合には指定する必要はありません。

2. 改行文字をスキップする場合，改行記号（￥ n）と復帰記号（￥ r）で改行文字が構成されるときには，length=2 と指定してください。

(b) type=データ型

pdload 用アンロードファイルの入力データファイルの場合，スキップする列のデータ型を指定します。

データ型の記述方法及びデータ型の指定可否について次の表に示します。

表 5–47 データ型の記述方法及びデータ型の指定可否

パラメタのデータ型		データ型の記述方法	pdload 用アンロードファイルの指定可否
分類	データ型		
数データ	INTEGER	integer	○
	SMALLINT	smallint	○
	DECIMAL	dec（精度［，位取り］）	○
		adec（精度［，位取り］）	×
		bdec(n)	×
	FLOAT	float	○
	SMALLFLT	smallflt	○
文字データ	CHARACTER	char(n)	○
	VARCHAR	varchar(n)	○
各国文字データ	NCHAR	nchar(m)	○
	NVARCHAR	nvarchar(m)	○
混在文字データ	MCHAR	mchar(n)	○
	MVARCHAR	mvarchar(n)	○
日付データ	DATE	date	○
		yy-mm-dd	×
		yymmdd	×
時刻データ	TIME	time	○
時刻印データ	TIMESTAMP	timestamp(p)	○
日間隔データ	INTERVAL YEAR TO DAY	interval-year-to-day	○
時間隔データ	INTERVAL HOUR TO SECOND	interval-hour-to-second	○
長大データ	BLOB	blob	○
		blobprm	×
バイナリデータ	BINARY	binary(n)	○

パラメタのデータ型		データ型の記述方法	pdload 用アンロードファイルの指定可否
分類	データ型		
		binaryprm	×

（凡例）

○：指定できます。

×：指定できません。

抽象データ型の列を持つ表を pdrorg のアンロード処理でバイナリ形式の入力データファイルに出力した場合，ファイル中の抽象データ型列のデータは，アンロード時に指定したコンストラクタパラメタ逆生成関数によって，コンストラクタ関数の引数のデータ型に変換されて出力されています。そのため，type オペランドに指定するデータ型は，コンストラクタパラメタ逆生成関数が生成するデータ型にしてください。コンストラクタパラメタ逆生成関数が生成するデータ型については，各プラグインマニュアルを参照してください。

(c)　element=要素数

pdrorg 出力バイナリ形式の入力データファイルの場合，スキップする列が繰返し列のときに，その繰返し列の最大要素数を指定します。

このオペランドは，必ず type オペランドの後に指定してください。

(3)　注意事項

1. 固定長データ形式の入力データファイルがテキストファイルの場合，行の終端の改行文字を格納しないようにするときにスキップデータ文を指定します。
2. スキップデータ文は列に対して指定するため，関数の引数に対しては指定できません。
3. 入力データファイルごとの，オペランドの指定可否を次に示します。

オペランド	入力データファイルの形式		
	DAT 形式	固定長データ形式	pdrorg 出力バイナリ形式
length	×	○	×
type	×	×	○

（凡例）

○：必ず指定します。

×：指定できません。

5.7.3　追加データ文

(1)　形式

```
＊adddata＊
列名，sequence=(〔認可識別子.〕順序数生成子識別子)
```

(2)　オペランドの説明

(a)　列名

順序数生成子の順序番号を格納する列名を指定します。

(b)　sequence=(〔認可識別子.〕順序数生成子識別子)

順序番号を取得する順序数生成子の識別子を指定します。

認可識別子を省略した場合は，pdload 実行ユーザ（-u オプション又は PDUSER 環境変数に設定した認可識別子）が仮定されます。

(3)　注意事項

列構成情報ファイルの 1 行目に追加データ文を指定して，順序数生成子が生成する順序番号だけをデータロードすることはできません。順序番号だけを表に格納する場合は，該当する行の行数文のデータを用意し，sequence オペランドに replace を指定してデータロードしてください。順序数生成子だけを表に格納する場合の例を次に示します。

（例）

表定義は次のように指定してください。

```
 create table TABLE_C (
   col01 integer
);
```

入力データファイルは，DAT 形式で行数分の改行データを用意してください。col01 列に順序数生成子 USER01.SEQ_C01 が生成する順序番号を格納する場合，列構成情報ファイルは次のように記述してください。

```
col01,sequence=(USER01.SEQ_C01) , replace=force
```

pdparaload を使用する場合，又は RD エリア単位にデータロードする場合は，分割キーの列に対してこのオペランドを指定しないでください。指定すると，分割キーの値が順序数生成子から取得した順序番号になるため，該当する分割キーの値が指定した RD エリアへの格納対象外の値となり，データを格納できないおそれがあります。

5.7.4　記述例

列構成情報ファイルの記述例を次に示します。

(1)　DAT 形式

(a)　入力データファイル中に不要なデータが含まれる場合

表定義：

```
CREATE TABLE T1 (COL1 INTEGER,COL2 DATE,COL3 CHAR(10));
```

入力データファイル及び列構成情報ファイル：

入力データファイル（COL2，COL1，不要データ，COL3のデータを持つ場合）

```
1997-06-26,-1,Y,abc
1964-09-13,0,Y,ABC
1999-12-31,999,N,abcdefg
```

列構成情報ファイル

```
COL2
COL1
*skipdata*
COL3
```

(b)　順序数生成子を使用した場合

順序数生成子（USER01.SEQ_B01）から順序番号を生成し，COL03 に格納する例について示します。

表定義：

```
CREATE TABLE T1 (COL1 INTEGER,
                 COL2 CHAR(10),
                 COL3 DECIMAL(10),
                 COL4 INTEGER);
```

入力データファイル：

```
1,ABC,2500
2,DEF,30000
3,GHI,10
   :
```

列構成情報ファイル：

```
COL1
COL2
COL4
*adddata*
COL3,sequence=(USER01.SEQ_B01)
```

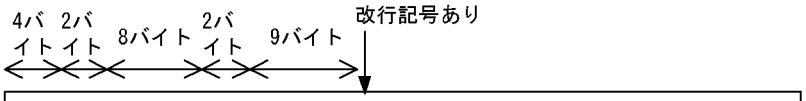
(2)　固定長データ形式

表定義：

```
CREATE TABLE T1 (COL1 INTEGER,COL2 DATE,COL3 CHAR(10));
```

入力データファイル及び列構成情報ファイル：

入力データファイル（COL1，COL2，COL3のデータを持つ場合）




```
 -1     97/06/26     abc
  0     64/09/13     ABC
999     99/12/31     abcdefg
```

列構成情報ファイル

```
COL1,type=char(6)                    1
COL2,type=yy-mm-dd
*skipdata*,length=2                  2
COL3,type=char(9)                    3
*skipdata*,length=1                  4
```

［説明］

1　4バイトの文字と2バイトの空白の場合，"COL1,type=char(4)"と"*skipdata*,length=2"に分けて記述しなくても，この指定のようにまとめて記述できます。

2　列COL3へ格納するデータは文字のため，左詰めをするスキップデータ文を指定しています。

3　列構成情報ファイルに指定した列長の合計が1行の長さになります。したがって，表定義がCHAR(10)でも入力データは9バイトのため，この指定となります。

4　テキストファイルの場合は，改行記号を含んでいるのでスキップデータ文を指定します。つまり，この固定長データ形式ファイルの1行（1レコード）の長さはデータ部の25バイト＋改行記号となり，26バイトと判断されます。

(3)　pdrorg 出力バイナリ形式

LOB 属性を含む SGMLTEXT 型の列がある表 T1 と，BLOB 型の列がある表 T2 を例にして，表 T1 のアンロードデータを表 T2 にデータロードする場合について説明します。なお，説明中の表定義は，列名及びデータ型の指定部分だけで，ほかの指定部分は省略しています。

(a)　T1 と T2 の列定義の順序が異なる場合

- アンロード時の表定義

```
T1(C1 INT, C2 SGMLTEXT, C3 BLOB)
```

- データロード時の表定義

```
T2(C4 BLOB ,C5 INT, C6 SGMLTEXT)
```

- 列構成情報ファイルの内容

```
C5
C6, func=(SGMLTEXT, param=blob)
C4
```

(b) T2 の列が T1 より少ない場合

- アンロード時の表定義

```
T1(C1 INT, C2 SGMLTEXT, C3 BLOB)
```

- データロード時の表定義

```
T2(C4 INT ,C5 BLOB)
```

- 列構成情報ファイルの内容

```
C4
*skipdata*,type=blobprm
C5
```

(c) T2 の列が T1 より多い場合

- アンロード時の表定義

```
T1(C1 INT, C2 BLOB)
```

- データロード時の表定義

```
T2(C4 INT, C5 BLOB, C6 SGMLTEXT)
```

- 列構成情報ファイルの内容

```
C4
C5
```

(d) 列データを入れ替える場合

T1 の SGMLTEXT 型列の LOB 属性のデータ(C2)を T2 の LOB 列(C6)に，T1 の LOB 列のデータ(C3)を T2 の SGMLTEXT 型列の LOB 属性(C5)に，それぞれ入れ替えます。

- アンロード時の表定義

```
T1(C1 INT, C2 SGMLTEXT, C3 BLOB)
```

- データロード時の表定義

```
T2(C4 INT ,C5 SGMLTEXT, C6 BLOB)
```

- 列構成情報ファイルの内容

```
C4
C6, type=blobprm
C5, func=(sgmltext, param=blob)
```

(e)　列のデータ型を変更する場合

SGMLTEXT 型列の LOB 属性のデータ(C2)を SGMLBIN 型列の BINARY 属性(C5)に，LOB 列のデータ(C3)を BINARY 型列(C6)にデータロードします。

- アンロード時の表定義

```
T1(C1 INT, C2 SGMLTEXT, C3 BLOB)
```

- データロード時の表定義

```
T2(C4 INT ,C5 SGMLBIN, C6 BINARY)
```

- 列構成情報ファイルの内容

```
C4   ..............................................1
C5, func=(sgmlbin, param=binary, type=blobprm)   ....2
C6, type=blob   ...................................3
```

[説明]

1. T1 の C1 データを，T2 の C4 に格納することを意味します。C1 と C4 のデータ型は同じであるため，データ型の指定は省略します。
2. T1 の C2 データを，T2 の C5 に格納することを意味します。プラグインが提供する抽象データ型列 SGMLBIN にデータを格納するため，コンストラクタ関数 sgmlbin と属性のデータ型 binary を func オペランドで指定します。入力データは，プラグインが提供する抽象データ型列 SGMLTEXT の LOB 属性のデータのため，type オペランドで LOB 属性であること（blobprm）を指定します。
3. T1 の C3 データを T2 の C6 に格納することを意味します。C3 のデータ型と C6 のデータ型が異なるため，type オペランドに入力データのデータ型（blob）を指定します。

(f)　T1 と T2 の列定義が異なる場合

CHARACTER 型の定義長を，10 から 21 に拡張してデータロードします。

- アンロード時の表定義

```
T1(C1 CHAR(10), C2 BLOB)
```

- データロード時の表定義

```
T2(C1 CHAR(21), C2 BLOB)
```

- 列構成情報ファイルの内容

```
C1,type=char(10)
C2
```

(g)　データ変換をする場合に必要となる領域

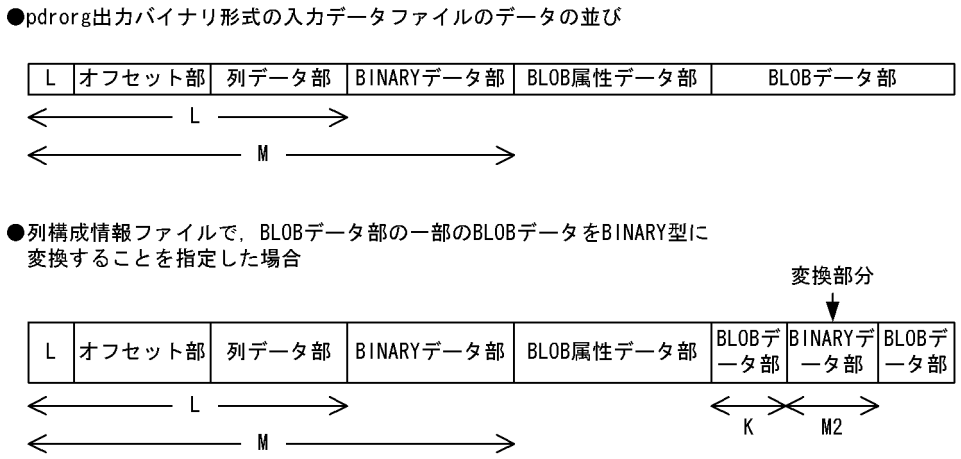
データ変換をする場合，データ変換後の BINARY 型データ，抽象データ型の BINARY パラメタデータ，及び抽象データ型の BLOB パラメタデータのデータの間に BLOB データが入るような並びになったときは，

その BLOB データ保持するためにメモリを確保する必要があります。メモリを確保できない場合は，メモリ確保エラーで終了します。次のような場合に，メモリの確保が必要となります。

- BLOB データを，BLOB 属性の抽象データ型列にデータロードする場合，その BLOB データよりも前に BLOB 型の列へデータロードするデータがあるとき
- 抽象データ型の BLOB パラメタデータを BLOB 型の列にデータロードする場合，その抽象データ型の BLOB パラメタデータよりも後に BLOB 属性の抽象データ型列へデータロードするデータがあるとき
- BLOB データ，又は抽象データ型の BLOB パラメタデータを，BINARY 型の列，又は BINARY 属性の抽象データ型列にデータロードする場合，対象となる BLOB データ，又は抽象データ型の BLOB パラメタデータより前に，ほかの BLOB 型の列へデータロードするデータがあるとき

BINARY 型の列，BLOB 属性の抽象データ型列，及び BLOB 型の列を定義した表から出力した，pdrorg 出力バイナリ形式の入力データファイルのデータの並びを次の図に示します。

図 5–27　pdrorg 出力バイナリ形式の入力データファイルのデータの並び



[説明]

pdload では，入力データ領域として少なくとも M の領域を確保します。領域は，格納先の表定義から M の最大長を算出して確保します（source 文の maxreclen オペランドで，任意にこの長さを指定することもできます）。

データがこの並びのままデータロードする場合は，データを順番に処理するため，pdload はこれ以上入力データを保持するための領域は確保しません。

列構成情報ファイルでデータの変換を指定した場合，pdload は M ＋ M2 ＋ K の領域を確保します。入力データ領域として，M ＋ M2 の領域を確保しますが，更に BINARY 型に変換した部分より前の BLOB データ部分も保持する必要があるため，K の領域も確保します。K の部分の領域は，option 文の bloblimit オペランドで指定できます。格納先の表定義から算出した値，又は option 文の bloblimit オペランドの指定値のうち，小さい方の値で領域を確保します。

5.8　ナル値・関数情報ファイル

ナル値・関数情報ファイルの比較値と入力データを比較して一致した場合，表にナル値を格納するときに指定します。また，抽象データ型の列に格納する値を生成するコンストラクタ関数の名称が，列のデータ型の名称と異なる場合や，該当する関数が複数あるときに関数情報を指定する場合に指定します。

このナル値・関数情報ファイルに指定する文を，列番号文といいます。

ナル値・関数情報ファイルは，入力データがバイナリ形式の場合に指定できます。

列番号文は，必ず 1 行で記述してください。複数列分の列番号文を記述したい場合は，複数行記述してください。

5.8.1　列番号文

(1)　形式

```
列番号 {, null＝ {（先頭位置, 〔 {c | x} 〕’ 比較値’） |’ 数値比較値’}
          〔, element=要素数〕〔, elmtype=配列データ形式〕〔, nullset=ナル値オプション〕
        |, func=（関数名, param=引数の型
          〔, null＝ {（先頭位置, 〔 {c | x} 〕’ 比較値’） |’ 数値比較値’} 〕〔, …〕）
        |, sequence=（〔認可識別子.〕順序数生成子識別子
          〔, replace= {’ 数値比較値’ |（先頭位置, {c|x} ’ 比較値’） | null | force} 〕）
         }
```

(2)　オペランドの説明

(a)　列番号

～<符号なし整数>((1～30000))

入力データを格納する表の列の番号を，定義順（昇順）の列番号で指定します。予備列に対応する列番号は指定できません。指定した場合，KFPL25103-E メッセージが出力され，異常終了します。

(b)　列番号以外のオペランド

列番号以外のオペランドの説明については，「列名文」の該当するオペランドを参照してください。

(3)　記述例

ナル値・関数情報ファイルの記述例を次に示します。ここでは，バイナリ形式を例にしています。

- **表定義**

```
CREATE TABLE T1 (C1 INT,C2 DATE,C3 VARCHAR(10),C4 BLOB(100k));
```

- **ナル値・関数情報ファイル**

```
1,null='-1'
2,null='0000-00-00'
3,null=(5,x'20')
4,null=(1,'@')
```

［説明］

各列に対応するデータが次の値の場合，ナル値が格納されます。

C1：0xffffffff

C2：0x00000000

C3：0x000a61626364202020202020

C4：0x0000000140

(4)　注意事項

列に対応するデータがない入力データファイルをデータロードした場合，表の列には WITH DEFAULT の既定値，又はナル値が格納されます。詳細については，「データロード時のナル値，及び既定値」を参照してください。

5.9　処理性能情報ファイル

%PDDIR%¥spool¥utlrpt ディレクトリ下に内部的な情報を出力します。

5.10　データロード時のナル値，及び既定値

データロード時のナル値，及び既定値は，表定義時のナル値，既定値の指定，及び pdload 実行時の制御文の指定によって格納値が変わります。データロード時に格納されるナル値，及び既定値を次の表に示します。

表 5–48　データロード時に格納されるナル値，及び既定値

<table>
<tr><th rowspan="2">入力データファイルの形式</th><th colspan="3">表定義時の指定値</th><th rowspan="2">pdload 実行時の option 文の null_string の指定値</th><th colspan="3">データロード時の格納値</th></tr>
<tr><th>非ナル値制約指定</th><th>WITH DEFAULT</th><th>DEFAULT 句</th><th>入力データがナル値の場合※1</th><th>列名文を省略した場合※2</th><th>非ナル値の場合</th></tr>
<tr><td rowspan="16">DAT 形式</td><td rowspan="4">NULL</td><td rowspan="4">－</td><td rowspan="2">あり</td><td>default</td><td>DEFAULT 句の既定値</td><td rowspan="2">DEFAULT 句の既定値</td><td rowspan="8">入力データの値</td></tr>
<tr><td>null</td><td rowspan="3">ナル値</td></tr>
<tr><td rowspan="2">なし</td><td>default</td><td rowspan="2">ナル値</td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="4">NOT NULL</td><td rowspan="2">あり</td><td rowspan="2">－</td><td>default</td><td rowspan="2">WITH DEFAULT の既定値</td><td rowspan="2">WITH DEFAULT の既定値</td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="4">なし</td><td rowspan="2">あり</td><td>default</td><td>DEFAULT 句の既定値</td><td rowspan="2">DEFAULT 句の既定値</td></tr>
<tr><td>null</td><td rowspan="3">ナル値（エラーとなります）</td></tr>
<tr><td rowspan="2">なし</td><td>default</td><td rowspan="2">×</td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="16">拡張 DAT 形式</td><td rowspan="4">NULL</td><td rowspan="4">－</td><td rowspan="2">あり</td><td>default</td><td>DEFAULT 句の既定値</td><td rowspan="2">DEFAULT 句の既定値</td><td rowspan="16">入力データの値</td></tr>
<tr><td>null</td><td rowspan="3">ナル値</td></tr>
<tr><td rowspan="2">なし</td><td>default</td><td rowspan="2">ナル値</td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="4">NOT NULL</td><td rowspan="2">あり</td><td rowspan="2">－</td><td>default</td><td rowspan="2">WITH DEFAULT の既定値</td><td rowspan="2">WITH DEFAULT の既定値</td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="8">なし</td><td rowspan="4">※3</td><td rowspan="2">あり</td><td>default</td><td>DEFAULT 句の既定値</td><td rowspan="2">DEFAULT 句の既定値</td></tr>
<tr><td>null</td><td rowspan="3">ナル値（エラーとなります）</td></tr>
<tr><td rowspan="2">なし</td><td>default</td><td rowspan="2">×</td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="4">※4</td><td rowspan="4">あり</td><td>default</td><td>DEFAULT 句の既定値</td><td rowspan="4">DEFAULT 句の既定値</td></tr>
<tr><td>null</td><td>WITH DEFAULT の既定値</td></tr>
</table>

<table>
<tr><th rowspan="2">入力データファイルの形式</th><th colspan="3">表定義時の指定値</th><th rowspan="2">pdload 実行時の option 文の null_string の指定値</th><th colspan="3">データロード時の格納値</th></tr>
<tr><th>非ナル値制約指定</th><th>WITH DEFAULT</th><th colspan="2">DEFAULT 句</th><th>入力データがナル値の場合※1</th><th>列名文を省略した場合※2</th><th>非ナル値の場合</th></tr>
<tr><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2">なし</td><td>default</td><td rowspan="2"></td><td rowspan="2">×</td><td rowspan="2"></td></tr>
<tr><td>null</td></tr>
<tr><td rowspan="4">バイナリ形式</td><td rowspan="2">NULL</td><td rowspan="2">－</td><td colspan="2">あり</td><td rowspan="4">－</td><td rowspan="2">ナル値</td><td rowspan="4">×</td><td rowspan="4">入力データの値</td></tr>
<tr><td colspan="2">なし</td></tr>
<tr><td rowspan="2">NOT NULL</td><td colspan="2">あり</td><td colspan="2">－</td><td rowspan="3">×</td></tr>
<tr><td>なし</td><td colspan="2">あり</td></tr>
<tr><td colspan="2">なし</td></tr>
<tr><td rowspan="6">固定長データ形式</td><td rowspan="3">NULL</td><td rowspan="3">－</td><td colspan="2">あり</td><td rowspan="6">－</td><td rowspan="3">ナル値</td><td>DEFAULT 句の既定値</td><td rowspan="6">入力データの値</td></tr>
<tr><td colspan="2">なし</td><td>ナル値</td></tr>
<tr><td rowspan="3">NOT NULL</td><td colspan="2">あり</td><td rowspan="3">×</td><td>WITH DEFAULT の既定値</td></tr>
<tr><td rowspan="2">なし</td><td colspan="2">あり</td><td>DEFAULT 句の既定値</td></tr>
<tr><td colspan="2">なし</td><td>×</td></tr>
<tr><td rowspan="6">pdrorg 出力バイナリ形式</td><td rowspan="3">NULL</td><td rowspan="3">－</td><td colspan="2">あり</td><td rowspan="6">－</td><td rowspan="3">ナル値</td><td>DEFAULT 句の既定値</td><td rowspan="6">入力データの値</td></tr>
<tr><td colspan="2">なし</td><td>ナル値</td></tr>
<tr><td rowspan="3">NOT NULL</td><td colspan="2">あり</td><td rowspan="3">ナル値（エラーとなります）</td><td>WITH DEFAULT の既定値</td></tr>
<tr><td rowspan="2">なし</td><td colspan="2">あり</td><td>DEFAULT 句の既定値</td></tr>
<tr><td colspan="2">なし</td><td>×</td></tr>
</table>

（凡例）

DEFAULT 句の既定値：詳細については表「データロード時に格納される DEFAULT 句の既定値」を参照してください。

WITH DEFAULT 句の既定値：詳細については表「データロード時に格納される WITH DEFAULT の既定値」を参照してください。

－：該当しません。

×：該当する表定義時の指定値，及び pdload 実行時の指定値の組み合わせでは，列名文を指定又は省略できないことを示します。

注

WITH DEFAULT，又は DEFAULT 句を指定した列に UNIQUE 指定のクラスタキー又は主キーが定義されている場合，既定値を格納するとキー重複エラーとなります。この場合，必ず入力データを指定してください。

注※1

入力データが次のデータの場合，入力データがナル値となります。

ナル値となる入力データの詳細については，「入力データファイル」を参照してください。

DAT 形式の場合：

- "*"となっているデータ
- 記述がないデータ

拡張 DAT 形式の場合：

- "*"となっているデータ
- 記述がないデータ
- 数データ型，日付データ型，日間隔データ型，時刻データ型，時間隔データ型，又は時刻印データ型の列の，入力データがすべて半角空白又はタブのデータ

バイナリ形式の場合：

ナル値・関数情報ファイルの列番号文の null オペランドに指定した比較値と一致するデータ

固定長データ形式の場合：

列構成情報ファイルの列名文の null オペランドに指定した比較値と一致するデータ

pdrorg 出力バイナリ形式の場合：

列データ位置オフセットが 0 のデータ

注※2

列構成情報ファイルで，該当する列に対する列名文を省略している場合を指します。

注※3

次の列を指します。

- UNIQUE を指定したインデクスの構成列
- UNIQUE を指定したクラスタキーの列
- PRIMARY を指定したクラスタキーの列
- 主キーの列
- 分割キーの構成列

注※4

注※3 以外の列を指します。

表 5–49　データロード時に格納される DEFAULT 句の既定値

<table>
<tr><th colspan="2" rowspan="2">データ型</th><th colspan="7">格納する列の DEFAULT 句の指定</th></tr>
<tr><th>定数</th><th>USER</th><th>CURRENT_TIME，又は CURRENT TIME</th><th>CURRENT_DATE，又は CURRENT DATE</th><th>CURRENT_TIMESTAMP，又は CURRENT TIMESTAMP</th><th>NULL</th><th>省略</th></tr>
<tr><td rowspan="2">数データ</td><td rowspan="2">INTEGER
SMALLINT
DECIMAL
FLOAT
SMALLFLT</td><td rowspan="14">設定値</td><td rowspan="2">–</td><td rowspan="2">–</td><td rowspan="2">–</td><td rowspan="2">–</td><td rowspan="14">ナル値</td><td rowspan="14">WITH DEFAULT の既定値と同じ</td></tr>
<tr><td rowspan="2">文字データ</td><td>CHARACTER</td><td rowspan="4">行を挿入した実行ユーザの認可識別子</td><td>pdload 実行時の時刻※1</td><td>pdload 実行時の日付※2</td><td>pdload 実行時の日時※3</td></tr>
<tr><td>VARCHAR</td><td rowspan="6">–</td><td rowspan="4">–</td><td rowspan="6">–</td></tr>
<tr><td rowspan="2">混在文字データ</td><td>MCHAR</td></tr>
<tr><td>MVARCHAR</td></tr>
<tr><td rowspan="2">各国文字データ</td><td>NCHAR</td><td rowspan="12">–</td></tr>
<tr><td>NVARCHAR</td></tr>
<tr><td>日付データ</td><td>DATE</td><td>pdload 実行時の日付</td></tr>
<tr><td>日間隔データ</td><td>INTERVAL YEAR TO DAY</td><td rowspan="10">–</td></tr>
<tr><td>時刻データ</td><td>TIME</td><td>pdload 実行時の時刻</td></tr>
<tr><td>時間隔データ</td><td>INTERVAL HOUR TO SECOND</td><td rowspan="3">–</td></tr>
<tr><td>時刻印データ</td><td>TIMESTAMP</td><td>pdload 実行時の日時※4</td></tr>
<tr><td>バイナリデータ</td><td>BINARY</td><td rowspan="2">–</td></tr>
<tr><td>長大データ</td><td>BLOB</td><td>–</td></tr>
</table>

（凡例）–：設定できません。

注※1

hh:mm:ss の形式で格納されます。

注※2

yyyy-mm-dd の形式で格納されます。

注※3

yyyy-mm-dd hh:mm:ss〔.000000〕の形式で格納されます。

注※4

小数秒部分は 0 を格納します。

表 5–50　データロード時に格納される WITH DEFAULT の既定値

データ型		格納する値
数データ	INTEGER SMALLINT DECIMAL FLOAT SMALLFLT	0
文字データ	CHARACTER	空白
	VARCHAR	1 バイトの空白
各国文字データ	NCHAR	空白
	NVARCHAR	1 文字の空白
混在文字データ	MCHAR	空白
	MVARCHAR	1 バイトの空白
日付データ	DATE	現在の日付
日間隔データ	INTERVAL YEAR TO DAY	0 か年 0 か月 0 か日
時刻データ	TIME	現在の時刻
時間隔データ	INTERVAL HOUR TO SECOND	0 時間 0 分 0 秒
時刻印データ	TIMESTAMP	現在の日時
バイナリデータ	BINARY	長さ 0 のデータ
長大データ	BLOB	長さ 0 のデータ

注

現在の日付，現在の時刻，及び現在の日時とは，データベース作成ユティリティを実行した日付及び時刻を示します。

5.11 UOC を利用したデータロード

5.11.1 概要

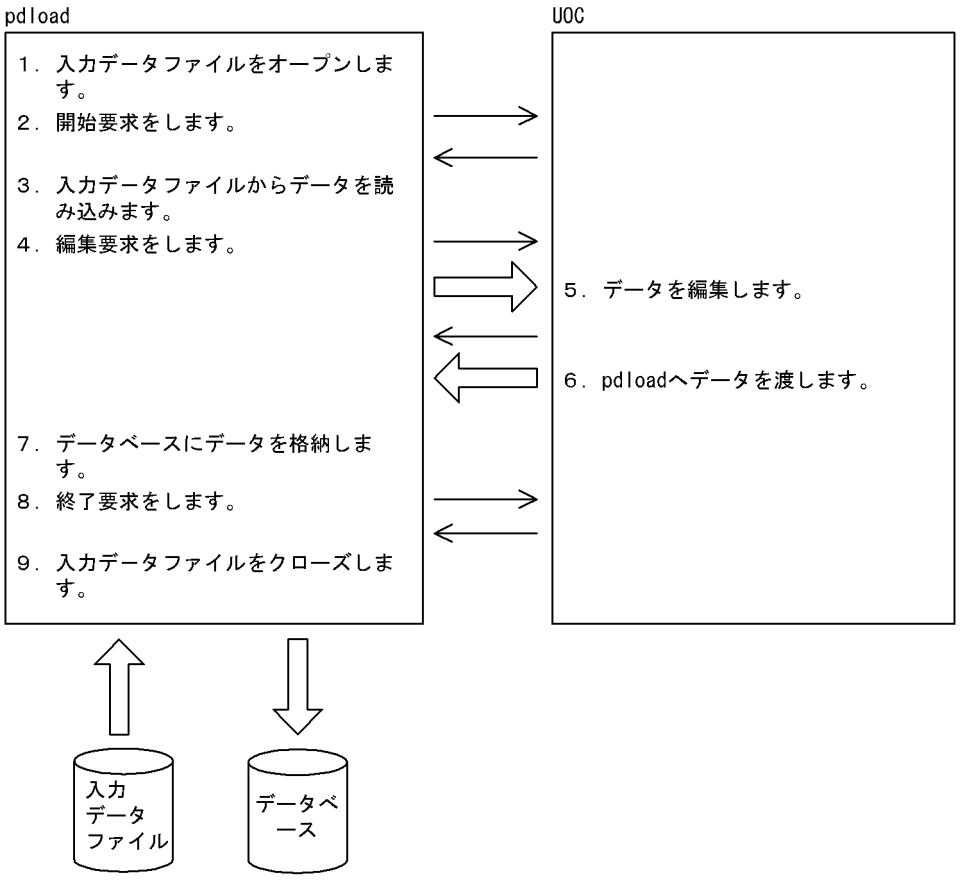
入力データファイルをそのままデータベースに格納するのではなく，ユーザが作成したプログラムでデータを編集してから格納したい場合があります。このような場合に UOC（User Own Coding）を使用します。UOC は C 言語で作成します。

UOC を利用して表を作成する場合，次の２種類の方法があります。

- ユティリティが入力データファイルを入力する方法（source 文にファイル名を指定）
- UOC が入力データファイルを入力する方法（source 文に(uoc)を指定）

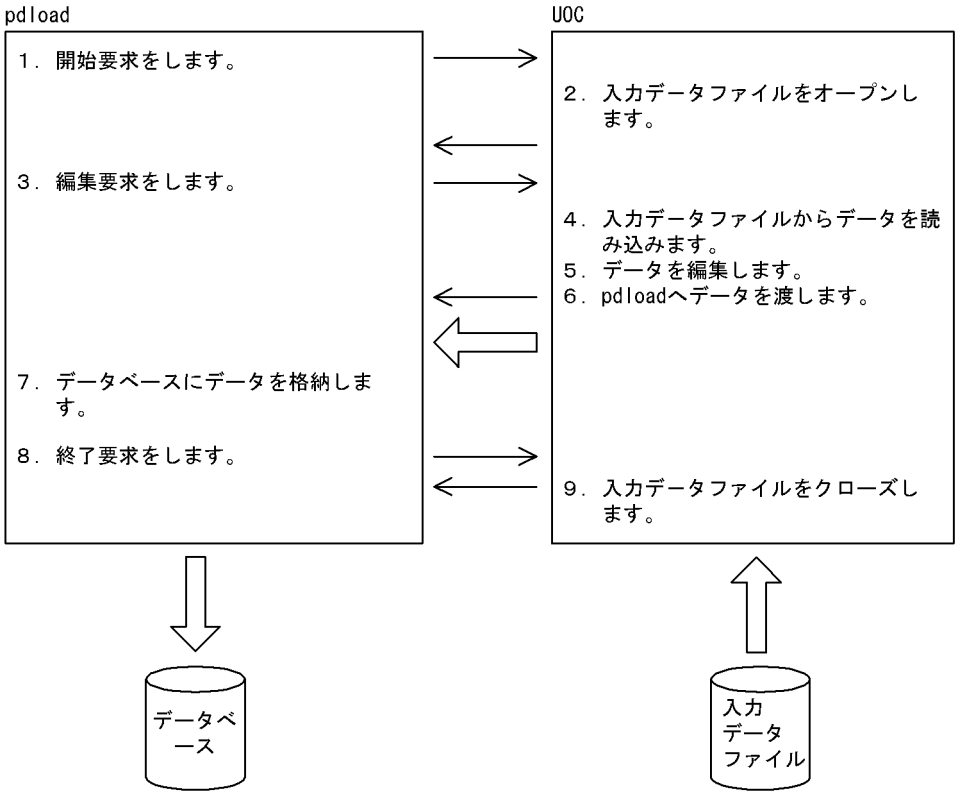
それぞれの概要を次の図に示します。

図 5–28 UOC を利用した場合の表の作成方法（ユティリティが入力データファイルを入力する方法）



［説明］
入力データファイルのデータがなくなるまで，３～７を繰り返します。

図 5–29　UOC を利用した場合の表の作成方法（UOC が入力データファイルを入力する方法）



［説明］
入力データファイルのデータがなくなるまで，３～７を繰り返します。

5.11.2　UOC インタフェース

データベース作成ユティリティと UOC 間の連絡は，インタフェース領域を使用して情報の入出力をします。

この領域はデータベース作成ユティリティが確保します。UOC は，関数の第 1 引数にこの領域のアドレスを受け取って参照，更新をします。

UOC インタフェース領域の構造を次の図に示します。また，UOC インタフェース領域の内容を表「UOC インタフェース領域の内容」に示します。

図 5–30　UOC インタフェース領域の構造（32 ビットモードの場合）

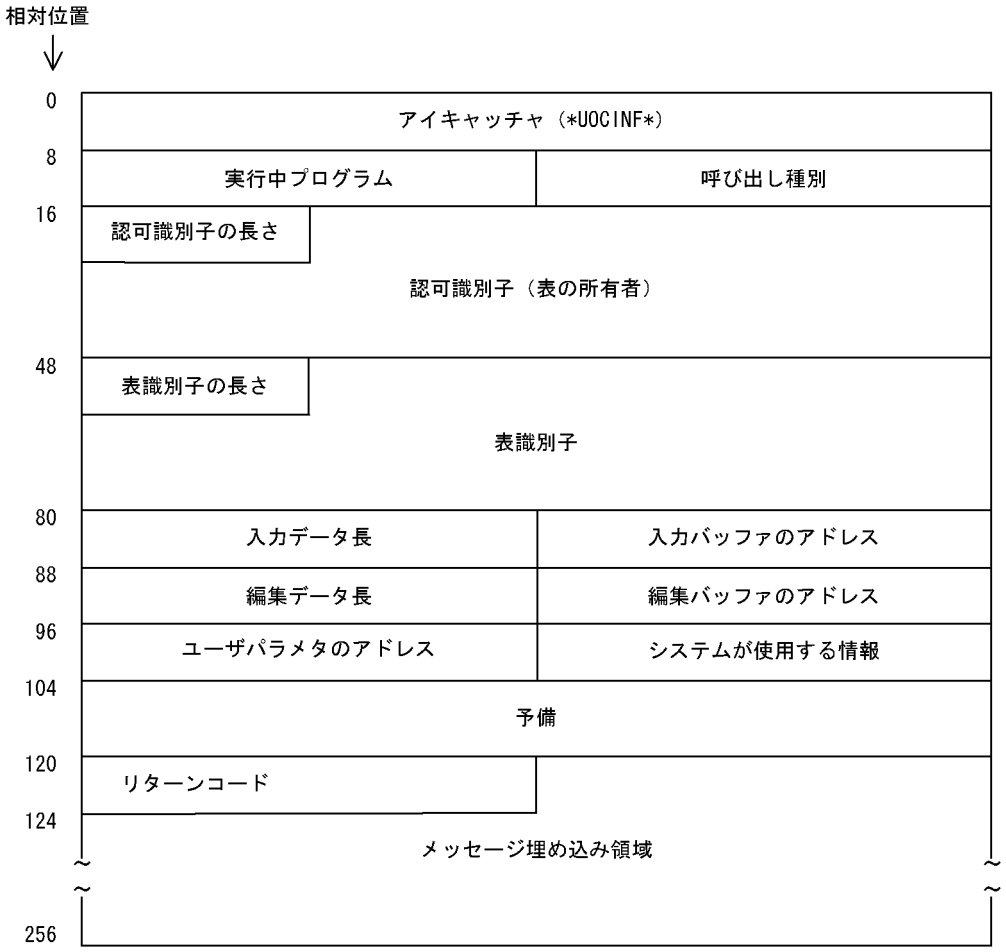


図 5–31　UOC インタフェース領域の構造（64 ビットモードの場合）

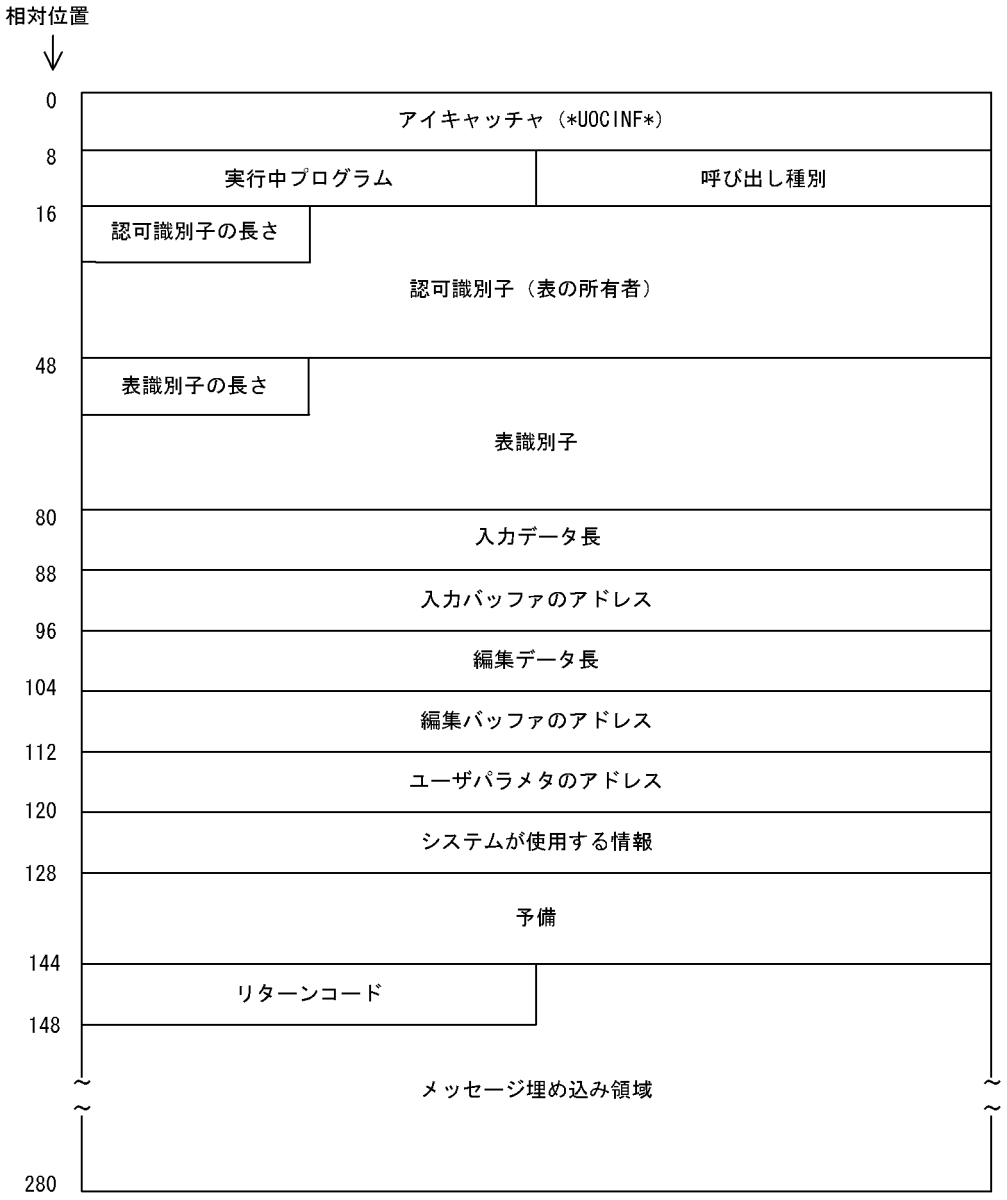


表 5–51　UOC インタフェース領域の内容

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
0	0	アイキャッチャ	8	8	char	pdload	インタフェース領域（’＊UOCINF＊’）
8	8	実行中プログラム	4	4	long	pdload	制御が渡っているプログラム 0：pdload 1：UOC
12	12	呼び出し種別	4	4	long	pdload	UOC に対する処理要求種別

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
							o：開始要求 処理開始の準備要求をします。 入力データファイルのオープンなど，処理の開始準備をします。 e：編集要求※1 行データの編集要求をします。 （1）の方法では，pdloadが入力した1行分のデータが入力バッファアドレスの指す入力バッファに設定されています。 （2）の方法では，UOCが入力データファイルの読み込みをします。 その後，（1），（2）共にデータを編集した結果を，編集バッファアドレスの指す編集バッファに設定します。 このコールは，データがなくなるか，又はUOCからのリターンコードが0，4以外になるまで繰り返します。 c：終了要求 処理の終了要求をします。 入力データファイルのクローズなど，終了処理をします。 （1）の方法ではpdloadが入力するデータがなくなった場合，（2）の方法ではUOCからリターンコード16を受け取った場合に要求をします。 t：停止要求 処理の強制終了要求をします。 pdload側でエラーが発生した場合，又はUOCからリターンコード0，4，16以外を受け取った場合に，

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
							終了要求の代わりにこのコールをします。 開始要求をした場合，正常時は終了要求，異常時は停止要求をします。終了要求又は停止要求以降は，UOC がコールされることはありません。
16	16	認可識別子の長さ	2	2	short	pdload	表の所有者名の長さ
18	18	認可識別子	30	30	char	pdload	表の所有者名
48	48	表識別子の長さ	2	2	short	pdload	表の名称の長さ
50	50	表識別子	30	30	char	pdload	表の名称
80	80	入力データ長※2	4	8	long	pdload	**(1)** バイナリ形式及び拡張 DAT 形式の入力データファイルの場合，入力バッファ※3に設定したデータの長さ（全長）を設定します。 **(2)** 無視されます。
84	88	入力バッファのアドレス	4	8	void*	pdload	**(1)** pdload が入力したデータを格納するバッファのアドレスを設定します。 **(2)** 無視されます。
88	96	編集データ長※2 ※5	4	8	long	UOC	バイナリ形式及び拡張 DAT 形式の入力データファイルの場合，編集バッファ※4に設定したデータの長さ（全長）を設定します。 **(1)** データを格納しない場合は−1 を設定します。
92	104	編集バッファのアドレス	4	8	void*	UOC	UOC で編集したデータを格納するバッファのアドレス
96	112	ユーザパラメタのアドレス	4	8	void*	pdload	srcuoc 文の param オペランドに指定した文字列のアドレスを設定します。param オペランドを省略した場合はナル値が設定されます。
100	120	システムが使用する情報	4	8	void*	pdload	システムが使用する情報（UOC 側では使用しないこと）
104	128	予備	16	16	char	pdload	予備領域（UOC 側では使用しないこと）。

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
							なお，最初の 1 バイトはシステムが使用する領域です。
120	144	リターンコード	4	4	long	UOC	UOC の処理結果を通知するリターンコードです。pdload から制御を受け取った UOC は，次の基準でリターンコードを戻してください。この領域は pdload が UOC をコールするごとにクリア（−1 を設定）しているので，毎回設定する必要があります。 0： 開始・終了要求に対する処理が正常に実行された場合，又は編集要求に対して編集バッファに 1 行分のデータを用意できた場合に設定します。 4： デバッグなどで，正常時もメッセージを出力したい場合に設定します。 内容は 0 と同じです。 リターンコードに 4 を設定した場合，メッセージは 3 回まで出力されます。メッセージが 3 回出力された後に 4 を設定しても無視され，メッセージは出力されません。 8： UOC の処理中にエラーが発生した場合，それまでの入力処理を有効にして（コミットして）処理を打ち切るときに設定します。 ただし，HiRDB 側で内部エラーが発生した場合は，ロールバックされます。コミットされたか，又はロールバックされたかについては，メッセージログに出力されたメッセージで確認できます。

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
							16： UOC が入力デ−タファイルを読み込む場合，データがなくなったことを pdload に通知するときに設定します。 また，pdload が入力デ−タファイルを読み込む場合に，今回受け取ったデータ以降のデータを受け取らないときにも指定できます。 このリターンコードを受け取ると，pdload は終了処理をします。 20： UOC の処理中にエラーが発生した場合，それまでの入力処理を無効にして（ロールバックして）処理を打ち切るときに設定します。 リターンコードに 0，16 以外の値を設定すると，メッセージ埋め込み領域の内容を標準出力及びメッセージログに出力します。メッセージは，末尾に（￥ 0）を設定して 131 バイト以内の文字列としてください。
124	148	メッセージ埋め込み領域	132	132	char	UOC	標準出力及びログに出力するメッセージの格納領域

（凡例）

32：32 ビットモードの HiRDB の相対位置，及び長さを示します。

64：64 ビットモードの HiRDB の相対位置，及び長さを示します。

注

内容の（1），（2）の意味を次に示します。

（1）：pdload が入力デ−タファイルを入力する場合

（2）：UOC が入力デ−タファイルを入力する場合

注※1

-e オプション指定時は，エラーデータを検知すると UOC の処理も打ち切ります。また，pdload が入力データファイルを入力する場合は次のようになります。

- -r オプションが指定されている場合は，指定した行データまでは渡されません。
- パーソナルコンピュータで作られた DAT 形式ファイルなどで，改行記号（￥ n）の前に復帰記号（￥ r）があるデータは，復帰記号は削除された形で渡されます。
- pdrorg が出力する-W オプション指定のバイナリ形式の入力デ−タファイルの場合，LOB 列及び BINARY 列の編集はできません。
- バイナリ形式の入力デ−タファイルを処理する場合，可変長文字列の長さが不正なデータは渡されません。

注※2

該当するデータを入力対象外にする場合に-1 を設定します。

64 ビットモードの HiRDB の場合，処理上有効なデータ長の最大値は 2 ギガ−1 バイトです。

注※3

入力バッファアドレスのフィールドを指す領域を，入力バッファと呼びます。入力バッファは pdload が入力デ−タファイルを読み込む場合に，pdload が確保します。入力バッファには，pdload が入力したデータが 1 行ごとに格納されます。入力バッファの形式は次のようになります。

・DAT 形式の入力デ−タファイルの場合

テキストデータを改行記号まで読み込んだものが，そのまま格納されます。末尾は改行記号（￥ n）＋ナル文字（￥ 0）となります。最後尾のナル文字以降を参照，更新しないでください。

・拡張 DAT 形式の入力デ−タファイルの場合

囲み文字で囲まれていないナル文字又は改行記号までのテキストデータを読み込んだものが，そのまま格納されます。格納されたデータの長さは入力データ長に設定されます。入力データ長に設定された長さ以上の領域を参照，更新しないでください。

・バイナリ形式の入力デ−タファイルの場合

表定義から計算した行長分のデータが格納されます。格納されたデータの長さは入力データ長に設定されます。入力データ長に設定された長さ以上の領域を参照，更新しないでください。

・pdrorg が出力した入力デ−タファイルの場合

入力データ長には，FIX 表の場合はデータ部だけの長さ，非 FIX 表の場合はオフセット部とデータ部の合計値が設定されます。入力データ長に設定された長さ以上の領域を参照，更新しないでください。なお，LOB 列及び BINARY 列は UOC では扱えないため，LOB 列及び BINARY 列がある表は，pdrorg の UOC で編集してください。

・固定長データ形式の入力データファイルの場合

列構成情報ファイルに指定した列情報から計算された，行長分のデータが格納されます。したがって，テキストファイルの場合は，改行記号（￥ n）及び復帰記号（￥ r）が付いたままの形となります。

注※4

編集バッファアドレスのフィールドを指す領域を，編集バッファと呼びます。編集バッファは UOC が確保します。ただし，pdload が入力デ−タファイルを読み込む場合，設定されたデータ長を超えて入力バッファの更新をしないときには，入力バッファをそのまま編集バッファとして利用できます。バイナリ形式の場合，編集データ長のフィールドにデータの長さを設定して，編集バッファの何バイト目までがデータなのかを指定します。DAT 形式の場合，長さを設定する必要はありませんが，編集バッファのデータの末尾は改行記号（￥ n）＋ナル文字（￥ 0）とします。

注※5

バイナリ形式及び拡張 DAT 形式の入力データファイルの場合，編集バッファに用意したデータの長さを指定してください。

拡張 DAT 形式の入力データファイルの場合，編集したデータの長さが source 文の maxreclen オペランドの指定値を超えないようにしてください。そのほかの形式の入力データファイルの場合，初期値の 0 にしてください。

5.11.3　UOC の使用例

ここでは，UOC を使用したデータロードの例について説明します。

なお，使用例中の UOC コーディング例は，サンプルデータベースとして提供されています。格納ディレクトリは，%PDDIR%¥sample¥sampleUOC です。

(1)　UOC を使用したデータロード例

2 けた表示の年号を持つ入力デ−タファイルを，2000 年対応のデータベースに格納するために 4 けた表示の年号に変更します。なお，入力デ−タファイルは DAT 形式で，pdload が入力デ−タファイルを入力します。

(a)　データベースの表定義

```
CREATE TABLE 名簿（社員番号 INTEGER,
                  生年月日 DATE,
                  出身地 CHAR(10),
                  氏名 CHAR(16));
```

(b)　入力デ−タファイルの形式

```
10001,68/04/30,KAWASAKI,YUUJI IKEDA
10002,64/09/13,HIROSHIMA,YOSHIO ODA
20001,70/11/02,YOKOHAMA,MASAMI KAWAGUCHI
```

(c)　コマンド形式

```
pdload 名簿 制御情報ファイル
```

(d)　制御情報ファイルの内容

```
source 入力データファイル名
srcuoc ライブラリ名 entry=date_change_func
```

(e)　UOC コーディング例

（ファイル名：sample1.c）

```
/*****************************************************************************
**                                                                          **
**  HiRDB sample Data input User Own Coding (for CSV file)                  **
**                          ~    ~   ~                                       **
**  name    :   date_change_func                                            **
**                                                                          **
**  func    :   YY/MM/DD  ===>  YYYY-MM-DD                                  **
**                                                                          **
**  i/o     :   none                                                        **
**                                                                          **
**  return  :   none                                                        **
**                                                                          **
*****************************************************************************/
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <errno.h>
#include <pdutluoc.h>                                                      1

#define DATELEN         10                  /* DATE length               */

    static  void    sub_date_change(char *,char *);

void date_change_func(
    UTL_UOC_INF    *uocinf                  /* A(UOC interface area)     */    2
){
    long            wk_leng;                /* length                    */
    char            date_data[10];          /* DATE (HiRDB)              */
    char           *date_ptr;               /* A(birth date)             */
    char           *address_p;              /* A(native)                 */
    static  char    buff[512];              /* buffer                    */

    switch(uocinf->req_cd){                                                3
    case UTL_UOC_START:
/*-------------------------------------------------------------------------*/
/*  START                                                                  */
/*-------------------------------------------------------------------------*/
        uocinf->edit_adr = buff;
        uocinf->rtn_code = UTL_UOC_NML;
        break;
    case UTL_UOC_EDIT:
/*-------------------------------------------------------------------------*/
/*  EDIT                                                                   */
/*-------------------------------------------------------------------------*/
        date_ptr = strchr(uocinf->data_adr,',');
        if (date_ptr == NULL){
            strcpy(uocinf->err_msg,"Invalid data");
```

```
            goto OWARI;
        }
        sub_date_change(++date_ptr,date_data);
        wk_leng = (long)date_ptr - (long)uocinf->data_adr;
        strncpy(buff,uocinf->data_adr,wk_leng);
        strncpy((char*)((long)buff + wk_leng),date_data,DATELEN);
        wk_leng += DATELEN;
        address_p = strchr(date_ptr,',');
        strcpy((char*)((long)buff + wk_leng),address_p);
        uocinf->rtn_code = UTL_UOC_NML;
        break;
    case UTL_UOC_END:
    case UTL_UOC_TERM:
/*-------------------------------------------------------------------------*/
/*  END                                                                    */
/*-------------------------------------------------------------------------*/
        uocinf->rtn_code = UTL_UOC_NML;
        break;
    default:
        strcpy(uocinf->err_msg,"Invalid request code");
        goto OWARI;
    }
    return;
OWARI:
    uocinf->rtn_code = UTL_UOC_ERR;
    return;
}

static void sub_date_change(
    char    *year_two,                  /* YY/MM/DD                    */
    char    *year_four                  /* YYYY-MM-DD                  */
){
    strcpy(year_four,"19");
    strncat(year_four,year_two,8);
    year_four[4] = '-';
    year_four[7] = '-';
    return;
}
```

[説明]

1. HiRDB が提供する，UOC 作成用のヘッダを引き込みます。
2. UOC インタフェース領域のアドレスを引数として受け取ります。
3. 呼び出し種別を判定してそれぞれの処理をします。

5.11.4　ダイナミックリンクライブラリの作成

UOC のコーディングが終了したら，ダイナミックリンクライブラリを作成します。ダイナミックリンクライブラリの作成方法については，「ダイナミックリンクライブラリの作成方法」を参照してください。

5.12　規則及び注意事項

5.12.1　規則

(1)　ユティリティの実行について

- pdload は，HiRDB が稼働中のときだけ実行できます。
- pdload は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

(2)　ユティリティの実行可否について

データロードする表，インデクス，及び LOB 列が格納されている R D エリアのオープン属性，RD エリアの状態によって，pdload の実行可否が変わります。pdload の実行可否については，「コマンド実行時の RD エリアの状態」を参照してください。

(3)　ユティリティの最大同時実行数

pdload の最大同時実行数は，システム共通定義の pd_utl_exec_mode オペランドの指定値によって異なります。

pd_utl_exec_mode=0 の場合：

最大同時実行数は 32 です。

pd_utl_exec_mode=1 の場合：

最大同時実行数は pd_max_users オペランドの指定値です。

(4)　ユティリティで処理中の表，インデクスについて

- pdload で作成中の表やインデクスは，ほかの UAP やユティリティからアクセスできません。ただし，option 文に nowait=yes を指定した場合，NOWAIT 検索はできます。NOWAIT 検索とは，排他オプションに NOWAIT を指定して検索する SQL のことをいいます。

  データロードする表が格納されている RD エリアを，pdhold コマンドで閉塞しておけば，UAP が閉塞エラーとなります。このため，UAP の実行によってユティリティの実行が排他待ちになることや，ユティリティの実行によって UAP の実行が排他待ちになることを防止できます。
- データロード中の表に対して，定義系 SQL は実行しないでください。実行すると，pdload がアボートコード Pu02004 で異常終了します。データロード中の定義系 SQL 実行を防止するためには，データロードする表及びインデクスが格納されている RD エリアを，pdhold コマンドで閉塞状態にしておく必要があります。なお，データロード中に同一 RD エリア内の別の表を参照する場合には，pdhold コマンドで参照可能な閉塞状態にしておきます。
- 定義系 SQL を実行中の RD エリアにある表やインデクスに対して，pdload を実行すると，排他待ちとなります。

- 入力データファイルが DAT 形式，又は固定長データ形式の場合，表の列に WITH DEFAULT の指定がなく，非ナル値制約の列に対して入力データの指定がないときはエラーとなります。

(5) CREATE TABLE，CREATE INDEX の PCTFREE との関係

- **データ格納時のページ内空き領域比率，及びセグメント内空きページ比率について**

  データロードのときのデータは，表定義（CREATE TABLE の PCTFREE オペランド）で指定したページ内空き領域比率，及びセグメント内空きページ比率に従ってデータが格納されます。

  インデクスについては，インデクス一括作成モードを指定した場合，又は作成モードで新しくデータを作成した場合には，インデクス定義（CREATE INDEX の PCTFREE オペランド）で指定したページ内空き領域比率に従ってデータが格納されます。

  ただし，-y オプションを指定した場合には，表定義時，及びインデクス定義時の空き領域比率に関係なく，使用中ページに空きがあればデータを格納します。また，option 文の tblfree オペランド，及び idxfree オペランドを指定した場合，表定義時，及びインデクス定義時の空き領域比率を変更してデータを格納できます。

- **空き領域のサーチ**

  pdload 実行時は，常に未使用ページにデータを格納します（未使用ページを確保できなかった場合はエラーとなります）。追加モード（-d 指定なし）でデータロードする場合，既存データを格納した使用中ページに空きがあっても利用しません。これは，既存データの使用中ページに空きがあっても，その空きをサーチするために処理時間が掛かってしまうためです（特にデータロードは，まとまったデータの追加をするため，処理時間が大幅に増大する可能性があります）。ただし，-y オプションを指定した場合には，未使用ページを確保できなかったときでもエラーにはなりません（空いている使用中ページにデータを格納します）。

  なお，SEGMENT REUSE 指定の表にデータロードする場合，SEGMENT REUSE の指定は無効となります。

(6) データロード中のシンクポイントについて

通常のデータロード時はシンクポイントを取得しません。このため，データロード中にほかの UAP などを実行してシステム障害が発生した場合など，再開始に要する時間が増大するため，できるだけデータロードとほかの UAP などは同時に実行しないようにしてください。

同期点指定のデータロードの場合は，任意の行数単位にシンクポイントを取得できます。このため，通常のシンクポイントを取得しないデータロードに比べると，異常終了時の再開始に要する時間は短くなります。

(7) 各国文字データのチェックについて

各国文字データとして格納する入力データの値については，データがマルチバイトコードかどうかのチェックはしません。

(8) DECIMAL 型の符号正規化機能について

DECIMAL 型の符号正規化機能を使用している場合，DECIMAL 型の符号部は次のように扱われます。
DECIMAL 型の符号正規化機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

- DECIMAL 型の符号部の仕様

HiRDB での，DECIMAL 型の符号部の仕様を次に示します。

符号部	意味
X'C'	正の値を示しています。
X'D'	負の値を示しています。
X'F'	正の値を示しています。

- DECIMAL 型の符号部の変換規則

DECIMAL 型の符号正規化機能を使用すると，データを挿入したときに DECIMAL 型の符号部を次の規則に従って変換します。

- DECIMAL 型の符号部の変換規則（0 データ以外の場合）

埋込み変数のデータの符号部	正規化しない場合	正規化する場合
X'A'	エラー	X'C'に変換
X'B'	エラー	X'D'に変換
X'C'	無変換	無変換
X'D'	無変換	無変換
X'E'	エラー	X'C'に変換
X'F'	無変換	X'C'に変換
X'0'～X'9'	エラー	エラー

- DECIMAL 型の符号部の変換規則（0 データの場合）

0 データの符号部	正規化しない場合	正規化する場合
X'A'	エラー	X'C'に変換
X'B'	エラー	
X'C'	無変換	
X'D'	無変換	
X'E'	エラー	
X'F'	無変換	

このように，0 データの正規化を実施すると，−0 のデータが＋ 0 に変換されます。

なお，文字データを変換する DAT 形式の入力データファイルの場合は，システム定義の指定に関係なく，正規化された値（正の値，0 データの場合は符号部が X'C'，又は負の値の場合は符号部が X'D'）を格納します。

- エラー情報ファイル，及びエラーデータファイルを参照する場合の注意事項

DECIMAL 型の符号正規化機能を使用した場合，エラー情報ファイルに出力されるエラーデータのダンプイメージリスト，及びエラーデータファイルに出力される入力データ中の DECIMAL 列は，pdload がエラーを検知した列までは正規化された値で出力されます。

(9)　抽象データ型がある表へのデータロード

プラグインが提供する抽象データ型列がある表の場合は，コンストラクタ関数を呼び出して，該当する列へ格納する値を生成しデータロードできます。ただし，コンストラクタ関数が共用ライブラリで作成されていない場合はデータロードできません。

ユーザが定義した抽象データ型列がある表の場合は，該当する列へ格納する値の生成ができないため，データロードできません。

抽象データ型列がある表にバイナリ形式の入力データファイルを使ってデータロードする場合は，UOC を利用したデータロードはできません。

(10)　リバランス表へのデータロード

リバランス表に RD エリアを追加した場合，そのリバランス表に対して pdrbal を実行（リターンコード=0）していないと，追加した RD エリアへのデータロード（RD エリア単位のデータロード）はできません。

また，フレキシブルハッシュ分割表にデータロードする場合，ハッシュグループを無視して格納するため，pdrbal で再配置するデータがハッシュキー値とリンクしません。

(11)　改竄防止表へのデータロード

改竄防止表にデータロードする場合，次の注意事項があります。

- -d オプションは指定できません。
- データロード対象となる RD エリアがデータ未完状態の場合，データロードできません。データ未完状態とは，改竄防止表に対する表の再編成中にエラーなどが発生し，表の再編成処理が完了していない状態のことをいいます。
- 挿入履歴保持列（SYSTEM GENERATED 指定の列）には，pdload の実行を開始した日時（pdload を実行したホストの日時）が格納されます。
- 挿入履歴保持列に対して入力データを指定しても無視されます。

(12)　トリガが定義されている表へのデータロード

pdload の実行でトリガは動作しません。pdload を実行する前に，表に定義されているトリガ定義の内容を確認してください。

(13)　一時表へのデータロード

一時表に対して，データロードはできません。一時表を指定して実行すると，pdload は KFPL15031-E メッセージを出力して，エラー終了します。

5.12.2　注意事項

(1)　ユティリティ実行後の結果について

pdload の結果は，pddbst で確認できます（UAP の実行，又は pdrorg でのアンロードでも確認できます）。また，pdload の処理が終了した場合は，リターンコードが返ります。pdload のリターンコードと，監査証跡表に pdload を実行した場合のリターンコードを次の表に示します。

表 5–52　pdload のリターンコード

リターンコード	意味	対処方法
0	正常終了。 入力データファイル中のすべてのデータをデータロードしました。	なし。
	正常終了。 入力データファイルの全データを，分割入力データファイルに出力しました。	分割入力データファイルを使用して，RD エリア単位にデータロードをしてください。
4	入力データのエラー。 入力データファイル中の一部のデータにエラーがあるため，データロードの処理をスキップしました。	エラー情報ファイルを参照し，入力データファイルのエラーデータを修正してから，再度データロードを実行してください。
	入力データのエラー。 入力データファイル中の正常なデータは分割入力データファイルに出力しましたが，一部のエラーデータは分割入力データファイルに出力していません（-e オプション指定時は，エラーデータ検知時点まで分割入力データファイルに出力）。	エラー情報ファイルを参照し，入力データファイルのエラーデータを修正してから，再度分割入力データファイルを作成してください。
8	入力データエラー。 option 文の dataerr オペランドを指定した場合は，入力データファイル中にエラーがあるため，データロード処理をロールバックしました。	エラーの入力データを修正して，再度データロードを実行してください。
	キー重複エラー。 インデクスの一括作成中にキー重複エラーを検知しました。	バックアップから DB を回復して，エラーの入力データを修正してください。その後，再度データロードを実行してください。
	異常終了。 エラーが発生したため，データロードの処理を途中で終了しました。	エラー原因を取り除いて，再度データロードを実行してください。

リターンコード	意味	対処方法
	異常終了。 入力データファイルのデータを，分割入力データファイルに出力できませんでした。	エラーメッセージを参照してエラーの原因を取り除いて，再度分割入力データファイルを作成してください。

表 5–53　pdload のリターンコード（監査証跡表の場合）

リターンコード	意味	対処方法
0	正常終了。 srcuoc 文に指定した監査証跡ファイルの全世代のデータを監査証跡表にデータロードしました。	なし。
4	警告終了。 srcuoc 文に指定した監査証跡ファイルの中で，データロード待ち状態のファイルは監査証跡表にデータロードしましたが，閉塞又はデータロード済み状態のファイルはデータロードの処理をスキップしました。	監査証跡ファイルが閉塞状態でスキップした場合，閉塞の原因を取り除いて，再度データロードを実行してください。 監査証跡ファイルがデータロード済み状態でスキップした場合，通常はデータロード済みのため対処は不要です。あえてデータロードする必要があるときは，srcuoc 文の mode オペランドに force を指定して再度実行してください。
8	異常終了。 エラーが発生したため，データロードの処理を途中で終了しました。	エラー原因を取り除いて，再度データロードを実行してください。

(2)　ユティリティ実行中の処理の中断について

pdload 実行中に処理を中断したい場合は，pdcancel コマンドを使用します。なお，pdload が無応答障害（定例ジョブなど，一定時間でデータロード処理が終わるはずなのに終わらない）と判断して強制終了させる場合は，pdls コマンド（-d rpc -a 指定）の表示結果をリダイレクトでファイルに取得してください。

この場合，作成モード（-d オプション指定）で pdload を実行しているときは，表の格納データはすべて削除されます。また，追加モード（-d オプションなし）で pdload を実行しているときは，処理がロールバックされます。

再編成時期予測機能を使用している場合，pdload をシグナル割り込みで強制終了すると，運用履歴表が更新できません。再編成時期予測機能を使用している場合に pdload を終了させるときは，必ず pdcancel コマンドを使用してください。

(3)　LOB 列構成基表及び LOB 列の RD エリアとバッファなどの資源について

LOB 列がある表にデータロードする場合，LOB 列構成基表，及び LOB 列の RD エリアとバッファなどの資源は，すべて準備しておく必要があります。LOB 列構成基表だけを作成する場合も同様です。

(4)　ユティリティ実行時に使用できるファイルの媒体について

pdload 実行時に使用できるファイルの媒体を次に示します。

ファイル	ファイル	固定長ブロックテープ	可変長ブロックテープ
入力データファイル	○	○※1※2	○※2
制御情報ファイル	○	×	×
列構成情報ファイル	○	×	×
ナル値・関数情報ファイル	○	×	×
エラー情報ファイル	○	×	×
エラーデータファイル	○	×	×
LOB 入力ファイル	○	×	○
LOB 中間ファイル	○	×	×
インデクス情報ファイル	○	×	×
ソート用ワークファイル	○	×	×
処理結果ファイル	○	×	×

（凡例）

○：使用できます。

×：使用できません。

注※1

バイナリ形式，及び固定長データ形式の場合は，使用できません。

注※2

pdrorg 用アンロードファイルの場合は，使用できません。

(5)　データ連動の対象となる抽出側データベースへのユティリティ実行について

データ連動の対象となる抽出側データベースに対して pdload を実行する場合，次の点に注意してください。

- 既にある表に対して，作成モードで実行しないでください（-d オプションを指定しないでください）。
- 既にある表に対して，追加モードでデータを追加する場合には，ログ取得モードで実行してください（-l オプションには a を指定してください）。
- HiRDB Datareplicator 連携の対象となる表を新規に追加する場合，ログ取得モードで実行してください（-l オプションには a を指定してください）。

(6) 行の最後がコンマ（,）で終わっている DAT 形式の入力データファイルをデータロードする場合について

行の最後がコンマ（,）で終わっている DAT 形式の入力データファイルをデータロードする場合に，「列数が不一致」などの理由でデータロードできないときがあります。このような場合は，列構成情報ファイルに表の列名をすべて記述して，最後にスキップデータ文を記述します。この場合，入力データの修正は不要です。

なお，表の列数が多く列構成情報ファイルの記述が大変なときは，次の方法で作成できます。

1. SQL Executer を使用して表の列名をファイルに出力します。

   該当する表の列名をファイルに出力する場合，列定義順に出力する必要があるため，SQL 文の記述は次のようにしてください。

   ```
   PUTFILE TO ファイル名 SELECT COLUMN_NAME FROM MASTER.SQL_COLUMNS
   WHERE TABLE_NAME='表名' ORDER BY COLUMN_ID;
   ```

2. 1.で編集した結果を出力したファイル名の最後の行に，スキップデータ文を追記します。

(7) プラグインインデクスが定義してある表へのデータロードについて

- プラグインインデクスが定義してある表にログレスモードでデータロードする場合，及びインデクス一括作成モードで追加データロードをする場合は，ユティリティの異常終了に備え，実行前に必ず処理対象の表が格納されている RD エリア（LOB 用も含む）とインデクス格納 RD エリアの両方のバックアップを取得してください。バックアップを取得しておかないと，障害が発生したときに RD エリアの回復ができないため，初期状態からの再作成が必要になります。
- インデクス一括作成機能（-i c オプション指定）を使用してプラグインインデクスを作成する場合，「プラグインインデクス一括作成部分回復機能」をプラグインが提供していれば，ユティリティの異常終了時に取得したログを使用してロールバックによるインデクスの回復ができます。この機能が提供されていない場合，ロールバックでの回復はできないため，インデクスは全件作り直しとなります。
- プラグインインデクス遅延一括作成機能を使用する場合，クライアント環境定義 PDPLGIXMK に YES を指定して UAP を実行した表に対しては，データベース再編成ユティリティでインデクスを一括作成した後でないと，追加モードでの pdload は実行できません。ただし，作成モードでの pdload は実行できます。

  また，この機能で使用するインデクス情報ファイル出力ディレクトリ（pd_plugin_ixmk_dir 指定ディレクトリ）を，pdload が使用するファイル出力領域として使用しないでください。

(8) HiRDB Text Search Plug-in の差分インデクス機能を使用している場合について

HiRDB Text Search Plug-in の差分インデクス機能を使用している場合，クライアント環境定義 PDPLUGINNSUB の指定によって，pdload が更新するインデクスを次に示します。

既存データの有無	PDPLUGINNSUB の指定		
	Y	N	設定なし
既存データなし（作成モードのデータロード）	M	M	M
既存データあり（追加モードのデータロード）	S	M	S

（凡例）

M：MASTER インデクスを更新

S：差分インデクスを更新

(9) ユニークキーインデクス又は主キーインデクスを定義した表へのデータロードについて

ユニークキーインデクス又は主キーインデクスを定義した表にデータロードをする場合は，次に示す注意が必要です。

- **入力データファイル中にキー値が重複するデータがある場合，インデクス更新モード以外のモードでデータロードをしないでください。**

  インデクス一括作成モード，インデクス情報出力モード，又はインデクス情報出力抑止モードでデータロードをすると，データを表に格納し，インデクスのキー情報をインデクス情報ファイルに出力します。

  この段階ではキー値の重複チェックはされません。キー値の重複チェックは，その後のインデクスデータの格納段階でされます。

  キー値が重複していると，インデクスの作成処理はロールバックされますが，データは既に格納されています（コミットされて元に戻りません）。この場合，バックアップを使用して RD エリアを回復する必要があります。

  したがって，**キー値の重複データがある入力データファイルを使用してデータロードをする場合は，必ずインデクス更新モードを指定してください。**このモードはデータを格納するごとにインデクスを更新するのでキー値の重複をすぐに検知し，該当するデータはデータベースに格納しません。

  なお，**デフォルト値がインデクス一括作成モードになっているので注意してください。**

(10) データロード対象の表を基にリストを作成している場合について

データロード対象の表を基にリストを作成している場合，データロード後にリストを使用して検索したときには，次のような現象が発生します。

- 誤った行を取り出す。
- 検索した行が見付からない。

このような場合，リストを使用して検索する前に，リストを再作成する必要があります。

(11)　コマンド実行画面に出力されるメッセージ

pdload は，標準出力に経過メッセージを出力しながら処理をします。また，エラー発生時は，標準エラー出力にエラーメッセージを出力します。標準出力や標準エラー出力への出力が抑止されている環境で pdload を実行した場合，メッセージ出力待ちで pdload が無応答状態となったり，メッセージログファイルに KFPL20003-E メッセージを出力して pdload が異常終了したりすることがあります。そのため，標準出力や標準エラー出力への出力ができない環境では pdload を実行しないでください。なお，標準出力，及び標準エラー出力に出力するメッセージの順序や数は，メッセージログファイルやイベントログと一致しないことがあります。正確にメッセージを把握したい場合は，メッセージログファイル又はイベントログを参照してください。

(12)　共用表へのデータロード

共用表にデータロードする場合，共用表，及び処理対象の表に定義した共用インデクスを格納している RD エリアに対して，EX モードで排他を掛けます。そのため，該当する RD エリアに定義したほかの表やインデクスも参照，及び更新ができません。共用表に対してデータロードする場合の排他制御モードについては，「ユティリティの排他制御モード」を参照してください。

(13)　参照制約又は検査制約を定義した表へのデータロード

参照制約又は検査制約を定義した表に対してデータロードする場合，pdload はデータの整合性が保たれているかどうかのチェックをしません。そのため，データロードする場合は，pdconstck を使用して整合性を確認する必要があります。表の整合性確認手順については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(14)　うるう秒を含んだ時刻データ又は時刻印データのデータロード

うるう秒を含んだ時刻データ又は時刻印データをデータロードする場合，pd_leap_second オペランドを Y にしてください。このオペランドを Y にすることによって，時刻データ，又は時刻印データの秒で指定できる範囲が 0〜61 秒になります。

(15)　再編成時期予測機能との関係

作成モード（-d 指定）のデータロードをすると，再編成時期予測機能の結果に表やインデクスのデータを削除した運用履歴が反映されます。

また，pdload が異常終了した状態のまま，pddbst の状態解析結果蓄積機能を実行すると，データロード完了後の状態で再編成時期を予測できないため，予測結果が不正※となります。したがって，pdload が異常終了した場合は，pdload を再実行して正常終了させた状態で，pddbst の状態解析結果蓄積機能を実行してください。

注※

例えば，データロード中に異常終了してロールバックした状態の DB の情報を蓄積すると，空の DB を基に予測してしまいます。

(16) 分割キー構成列に対して順序数生成子を指定した RD エリア単位データロード

分割キー構成列に対して順序数生成子を指定した RD エリア単位でのデータロードはできません。実行した場合は処理を続行します。その結果，入力データファイルのデータがデータベースに格納できない場合があります。

5.12.3 特定機能使用時のオプション，及び制御文の指定可否

pdload の特定機能を使用する場合，オプション，及び制御文の指定可否が変わります。特定機能とは次の機能をいいます。

- 監査証跡表へのデータ登録
- 分割入力データファイルの作成

特定機能使用時のオプション，及び制御文の指定可否を次の表に示します。なお，監査証跡表へのデータ登録については，マニュアル「HiRDB システム運用ガイド」を参照してください。

表 5–54 特定機能使用時のオプションの指定可否

オプション	指定可否	
	監査証跡表へのデータ登録	分割入力データファイルの作成
-d	○	×
-a	×	○（入力データファイルが固定長データ形式の場合-a を指定）
-b	◎（-b を指定）	×
-W	◎	×
-w	×	×
-U	×	×
-i	○	×
-l	○	×
-k	×	×
{-c ｜-v}	○（-v を指定）	○（入力データファイルが固定長データ形式の場合-c を指定）
-n	○	×
-u	○	○
-x	○	×

オプション	指定可否	
	監査証跡表へのデータ登録	分割入力データファイルの作成
-s	×	○
-e	×	○
-r	×	×
-z	○（省略した場合でも指定したと仮定）	○
-y	○	×
-o	○	×
-m	○	○
-X	○	○
-K	×	×
-G	×	×

（凡例）

◎：必ず指定します。

○：任意に指定できます。

×：指定できません。

表 5–55　特定機能使用時の制御文の指定可否

制御文	オペランド	指定可否	
		監査証跡表へのデータ登録	分割入力データファイルの作成
source	−	◎（(uoc)を指定）	◎（入力データファイルを指定）
index	−	○	×
idxwork	−	○	×
sort	−	○	×
lobdata	−	×	×
lobcolumn	−	×	×
lobmid	−	×	×
srcuoc	−	◎	×
array	−	○	○
extdat	−	×	○
src_work	−	×	◎
option	spacelvl	○	○

制御文	オペランド	指定可否	
		監査証跡表へのデータ登録	分割入力データファイルの作成
	tblfree	○	×
	idxfree	○	×
	job	×	×
	cutdtmsg	×	×
	nowait	○	×
	bloblimit	×	×
	exectime	○	○
	null_string	×	×
	divermsg	×	×
	dataerr	×	×
	lengover	×	○

（凡例）

◎：必ず指定します。

○：任意に指定できます。

×：指定できません。

－：該当しません。

5.12.4　BOM を持つファイルの使用

pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdload の入力ファイルには，BOM を持つファイルを使用できることがあります。pdload での BOM を持つファイルの使用可否を次の表に示します。なお，pdload の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdload で出力されるファイルには BOM は付きません。

表 5–56　pdload での BOM を持つファイルの使用可否（UTF-8 の場合）

オプション又は制御文	入力ファイル		BOM を持つファイルの使用
-c	列構成情報ファイル		○
-v	ナル・関数情報ファイル		○
－	制御情報ファイル		○
source	入力データファイル	DAT	○
		拡張 DAT	○
		バイナリ	×

オプション又は制御文	入力ファイル		BOM を持つファイルの使用
		固定長	×
		pdrorg -W 生成	×
index	インデクス情報ファイル		×
lobdata	LOB 入力ファイル		×
lobcolumn	列単位 LOB 入力ファイル		×
lobmid	LOB 中間ファイル		×
−	SQL 定義ファイル		×

（凡例）

○：使用できます。

×：使用できません。

−：該当しません。

5.12.5　制御文で指定しない場合のファイル出力先ディレクトリ

pdload 実行時，制御情報ファイルでファイル出力先を指定しない場合，次の表に示すディレクトリにファイルを出力します。

なお，SQL 定義ファイルは制御情報ファイルで出力先を指定できません。SQL 定義ファイルの出力先は，次の表に示す，制御文の指定をしない場合と同様です。

表 5–57　pdload がファイルを出力するディレクトリ

制御文※1 の指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※2 の指定	
		あり	なし
あり	制御文に指定したディレクトリ又はファイル		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※1

source 文（error オペランド），index 文，idxwork 文，sort 文，lobmid 文，又は lobmid 文（error オペランド）

注※2

システム環境変数の指定

5.13　エラー時のデータベースの状態とその回復方法

データロード中にエラーが発生した場合の，データベースの状態とその回復方法を次の表に示します。

- 表「データベースの状態とその回復方法（データロード中にエラーが発生した場合）(1/3)」
- 表「データベースの状態とその回復方法（データロード中にエラーが発生した場合）(2/3)」
- 表「データベースの状態とその回復方法（データロード中にエラーが発生した場合）(3/3)」

また，LOB 列へのデータロード中にエラーが発生した場合の，データベースの状態とその回復方法を表「データベースの状態とその回復方法（LOB 列へのデータロード中にエラーが発生した場合）」に示します。また，入力データエラー時の，データベースの状態とその回復方法を表「データベースの状態とその回復方法（入力データエラーの場合）」に示します。

表中の区分の状態とは，エラー発生時のデータベースの状態を示します。また，回復とは，回復方法を示します。

回復方法の実施は，発生したエラー原因を取り除いた後に行ってください。

表 5–58　データベースの状態とその回復方法（データロード中にエラーが発生した場合）(1/3)

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		1.	2.	3.
			なし（制御文エラーは除く）	KFPL00721-I 行データ削除	KFPL00702-I データロード開始
a 又は p	s	状態	実行前の状態	空	空
		回復	再実行	再実行	再実行
	c	状態	実行前の状態	空	空
		回復	再実行	再実行	再実行
	n	状態	実行前の状態	空	空
		回復	再実行	再実行	再実行
	x	状態	実行前の状態	空	空
		回復	再実行	再実行	再実行
n	s	状態	保証なし	保証なし	保証なし
		回復	DB 回復後再実行	DB 回復後再実行	DB 回復後再実行
	c	状態	保証なし	保証なし	保証なし
		回復	DB 回復後再実行	DB 回復後再実行	DB 回復後再実行
	n	状態	保証なし	保証なし	保証なし
		回復	DB 回復後再実行	DB 回復後再実行	DB 回復後再実行

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		1.	2.	3.
			なし（制御文エラーは除く）	KFPL00721-I 行データ削除	KFPL00702-I データロード開始
	x	状態	保証なし	保証なし	保証なし
		回復	DB 回復後再実行	DB 回復後再実行	DB 回復後再実行

表 5–59　データベースの状態とその回復方法（データロード中にエラーが発生した場合）（2/3）

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		4.	5.	6.
			KFPL00800-I 同期点通知	KFPL15225-E ロールバック	KFPL24510-E エラーデータ検知
a 又は p	s	状態	KFPL00800-I に出力された行まで格納済み	3.又は 4.の状態	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	再実行	再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※
	c	状態	KFPL00800-I に出力された行まで格納済み	3.又は 4.の状態	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	再実行	再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※
	n	状態	KFPL00800-I に出力された行まで格納済み	3.又は 4.の状態	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	再実行	再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		4.	5.	6.
			KFPL00800-I 同期点通知	KFPL15225-E ロールバック	KFPL24510-E エラーデータ検知
	x	状態	KFPL00800-I に出力された行まで格納済み	3.又は 4.の状態	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	再実行	再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※
n	s	状態	−	保証なし	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	−	DB 回復後再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※
	c	状態	−	保証なし	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	−	DB 回復後再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※
	n	状態	−	保証なし	KFPL15225-E が出力されている場合は 5.の状態。 出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	−	DB 回復後再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※
	x	状態	−	保証なし	KFPL15225-E が出力されている場合は 5.の状態。

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		4.	5.	6.
			KFPL00800-I 同期点通知	KFPL15225-E ロールバック	KFPL24510-E エラーデータ検知
					出力されていない場合は，KFPL24509-E に出力された行の直前の行まで格納済み。
		回復	−	DB 回復後再実行	5.の状態の場合は再実行。 途中まで行が格納されている場合は，-r オプション指定で再実行。※

（凡例）

−：該当するオプションを指定した場合，該当するメッセージは出力されないことを示します。

注※

-r オプション指定で再実行する場合，-d オプションは指定しないでください。

表 5–60　データベースの状態とその回復方法（データロード中にエラーが発生した場合）（3/3）

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		7.	8.	9.
			KFPL00703-I データロード完了	KFPL00715-I インデクスの作成開始	KFPL00716-I インデクスの作成完了
a 又は p	s	状態	表及びインデクス作成完了	−	−
		回復	不要※1	−	−
	c	状態	表作成完了	表作成完了	表及びインデクス作成完了
		回復	pdrorg -k ixmk で，インデクスの一括作成を実行	pdrorg -k ixmk で，インデクスの一括作成を実行※2	不要※1※2
	n	状態	表作成完了	−	−
		回復	pdrorg -k ixmk で，インデクスの一括作成を実行※1	−	−
	x	状態	表作成完了	−	−
		回復	pdrorg -k ixmk で，インデクスの一括作成を実行※1	−	−
n	s	状態	表作成完了	−	−
		回復	pdrorg -k ixmk で，インデクスの一括作成を実行※1	−	−
	c	状態	表作成完了	表作成完了	表及びインデクス作成完了

オプション		区分	エラー発生直前の出力メッセージ		
-l	-i		7.	8.	9.
			KFPL00703-I データロード完了	KFPL00715-I インデクスの作成開始	KFPL00716-I インデクスの作成完了
		回復	DB 回復後再実行※3	DB 回復後再実行※3	不要※1
	n	状態	表作成完了	−	−
		回復	pdrorg -k ixmk で，インデクスの一括作成を実行※1	−	−
	x	状態	表作成完了	−	−
		回復	pdrorg -k ixmk で，インデクスの一括作成を実行※1	−	−

（凡例）

−：該当するオプションを指定した場合，該当するメッセージは出力されないことを示します。

注※1

-k d 以外で LOB 列にデータロードしている場合，表「データベースの状態とその回復方法（LOB 列へのデータロード中にエラーが発生した場合）」も参照してください。

注※2

インデクス格納用 RD エリアが複数ある場合，KFPL00716-I メッセージが出力されたインデクスが作成済みですが，出力されていないインデクスは未作成となります。未作成のインデクスに対してインデクスの一括作成を実行してください。

注※3

表格納用 RD エリアとインデクス格納用 RD エリアが異なる場合，表格納用 RD エリアのログレス閉塞を解除した後，インデクス格納用 RD エリアを再初期化して pdrorg -k ixrc を実行してもかまいません。

表 5–61　データベースの状態とその回復方法（LOB 列へのデータロード中にエラーが発生した場合）

制御文	-l オプション	区分		エラー発生直前の出力メッセージ	
				KFPL00707-I LOB 列へのデータロード開始	KFPL00708-I LOB 列へのデータロード完了
LOB 列構成基表と LOB 列を同時にデータロードする場合： source 文 lobmid 文 lobdata 文	a 又は p	状態	USER	LOB 列構成基表とインデクス作成完了	表作成完了
			ULOB	-d オプション指定の場合は空。なしの場合は実行前状態。	LOB 列作成完了
		回復	USER	不要	不要
			ULOB	LOB 列にだけデータロード※2	不要※2
	n	状態	USER	ログレス閉塞	表作成完了

制御文	-l オプション	区分		エラー発生直前の出力メッセージ	
				KFPL00707-I LOB 列へのデータロード開始	KFPL00708-I LOB 列へのデータロード完了
			ULOB	保証なし	LOB 列作成完了
		回復	USER	DB 回復後再実行※1	不要
			ULOB	DB 回復後再実行※1	不要
LOB 列にだけデータロードする場合： lobmid 文 lobdata 文	a 又は p	状態	USER	−	−
			ULOB	実行前状態（-d オプションは無視されます）	LOB 列作成完了
		回復	USER	−	−
			ULOB	再実行	不要
	n	状態	USER	−	−
			ULOB	保証なし	LOB 列作成完了
		回復	USER	−	−
			ULOB	DB 回復後再実行	不要

（凡例）

−：該当するオプションを指定した場合，該当するメッセージは出力されないことを示します。

USER：ユーザ用 RD エリア

ULOB：ユーザ LOB 用 RD エリア

注※1

ログレスモードの場合，LOB 列へのデータロード中に異常終了したときでも，表（LOB 列構成基表，LOB 列，LOB 属性の抽象データ型列，及びインデクス）をバックアップから回復した後，再度，LOB 列構成基表，及び LOB 列へデータロードする必要があります。

注※2

ユーザ LOB 用 RD エリアが複数ある場合，KFPL00708-I メッセージが出力されている LOB 列は作成済みとなります。出力されていない LOB 列は未作成のため，未作成の LOB 列に対してデータロードを実行してください。

表 5–62　データベースの状態とその回復方法（入力データエラーの場合）

-e オプション	source 文の errdata オペランド	区分	ロールバック不要エラー (リターンコード 4 で終了)	ロールバック要エラー (リターンコード 8 で終了)
なし	なし	状態	エラー情報ファイルに出力された行以外は DB に格納済み	-l n 以外の場合は pdload 実行前状態。 -l n の場合は保証なし。

-e オプション	source 文の errdata オペランド	区分	ロールバック不要エラー (リターンコード 4 で終了)	ロールバック要エラー (リターンコード 8 で終了)
		回復	入力データファイル中の該当するデータを別ファイルにして修正し，再度データロード	-l n 以外の場合は，入力データを修正して再度データロード。 -l n の場合は，DB 回復後，入力データを修正して再度データロード。
	あり	状態	エラー情報ファイルに出力された行以外は DB に格納済み	-l n 以外の場合は pdload 実行前状態。 -l n の場合は保証なし。
		回復	エラーデータファイルに出力されたデータを修正し，再度データロード	-l n 以外の場合は，入力データを修正して再度データロード。 -l n の場合は，DB 回復後，入力データを修正して再度データロード。
あり	該当しません	状態	KFPL24509-E に出力された行の直前まで DB に格納済み	-l n 以外の場合は pdload 実行前状態。 -l n の場合は保証なし。
		回復	KFPL24509-E に出力された行のデータを修正して，-r オプション指定で再度データロード	-l n 以外の場合は，入力データを修正して再度データロード。 -l n の場合は，DB 回復後，入力データを修正して再度データロード。

注

ロールバック要エラーには，LOB 列格納データエラー，可変長文字列の長さ部が負の値，及び繰返し列の要素数が負の値があります。

インデクス更新モード（-i s）以外でのキー値重複エラーは，入力データエラーではなく，表とインデクスが不一致となる DB 破壊エラーとなります。回復方法は，DB を pdload 実行前の状態に戻して，入力データを修正し，再度データロードをしてください。

5.14　使用例

データベース作成ユティリティの使用例を次の表に示します。

表 5–63　データベース作成ユティリティの使用例

使用例	内　　容	区分
例 1	非横分割表へのデータロード • 入力データファイルが DAT 形式の場合と固定長データ形式の場合とで説明しています。 • 列構成情報ファイルを使用します。	S
例 2	横分割表への表単位のデータロード	S
例 3	横分割表への RD エリア単位のデータロード	S
例 4	LOB 列がある表へのデータロード • LOB 列構成基表と LOB 列へ同時にデータロードします。	S
例 5	LOB 列がある表へのデータロード • LOB 列構成基表だけデータロードします。	S
例 6	LOB 列がある表へのデータロード • LOB 列だけデータロードします。	S
例 7	監査証跡表へのデータロード • 監査証跡表に対して，データロード待ちの監査証跡ファイルをデータロードします。	S
例 8	自動採番機能を使用したデータロード	S
例 9	非横分割表へのデータロード • 入力データファイルはフロントエンドサーバにあるものとします。	P
例 10	横分割表への表単位のデータロード • 入力データファイルはフロントエンドサーバにあるものとします。	P
例 11	横分割表への RD エリア単位のデータロード • 入力データファイルは，それぞれのバックエンドサーバにあるものとします。	P
例 12	LOB 列がある表へのデータロード • LOB 列構成基表と LOB 列へ同時にデータロードします。 • 表は横分割表で，RD エリア単位にデータロードします。 • 入力データファイルは，データロード対象の RD エリアがあるバックエンドサーバにあるものとします。	P
例 13	LOB 列がある表へのデータロード • 列単位 LOB 入力ファイルを使用します。 • 表は横分割表で，表単位にデータロードします。 • 入力データファイルはフロントエンドサーバにあるものとします。	P
例 14	繰返し列がある表へのデータロード • 表は横分割表で，表単位にデータロードします。	P

使用例	内　　容	区分
	• 表にはクラスタキーインデクスが定義されていて，入力データの順序がクラスタキー順になっていることが分かっているため，クラスタキー順のチェックをしないオプションを指定します。	
例 15	プラグイン使用時の表へのデータロード • 表には SGMLTEXT 型の列があります。 • 入力データファイルは DAT 形式です。 • コンストラクタ関数の情報を指定するため，列構成情報ファイルを使用します。	P
例 16	プラグイン使用時の表へのデータロード • 表には GEOMETRY 型の列があります。 • 入力データファイルは固定長データ形式です。 • 入力データファイルが固定長データ形式である，また，コンストラクタ関数の情報を指定するため，列構成情報ファイルを使用します。	P
例 17	プラグイン使用時の表へのデータロード • 表には SGMLTEXT 型の列と，LOB 列があります。 • LOB パラメタは入力データファイル中に直接記述し，LOB 入力ファイルはファイル名を入力データファイル中に記述して LOB 入力ファイルを別に用意するものとします。	P
例 18	プラグイン使用時の表へのデータロード • 表には SGMLTEXT 型の列と，LOB 列があります。 • 先に LOB 列構成基表だけデータロードします。その後，各バックエンドサーバで並列に LOB 列へデータロードします。 • LOB パラメタは入力データファイル中に直接記述し，LOB 入力ファイルはファイル名を入力データファイル中に記述して LOB 入力ファイルを別に用意するものとします。	P
例 19	監査証跡表へのデータロード • 監査証跡表に対して，データロード待ちの監査証跡ファイルをデータロードします。	P

（凡例）

S：HiRDB/シングルサーバ

P：HiRDB/パラレルサーバ

5.14.1　HiRDB/シングルサーバの場合

(1)　表へのデータロード

例 1

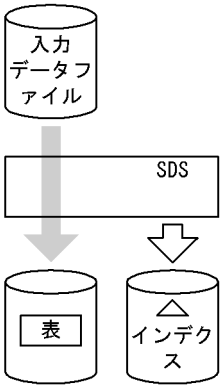
表（TABLE1）にデータロードします。この例では，入力データファイルがDAT形式の場合と，固定長データ形式の場合について説明します。
なお，次の表及びインデクスが定義されているものとします。
表の定義

```
  CREATE TABLE TABLE1(C1 INT,C2 CHAR(8),C3 INT) IN PDB USER01
```

インデクスの定義

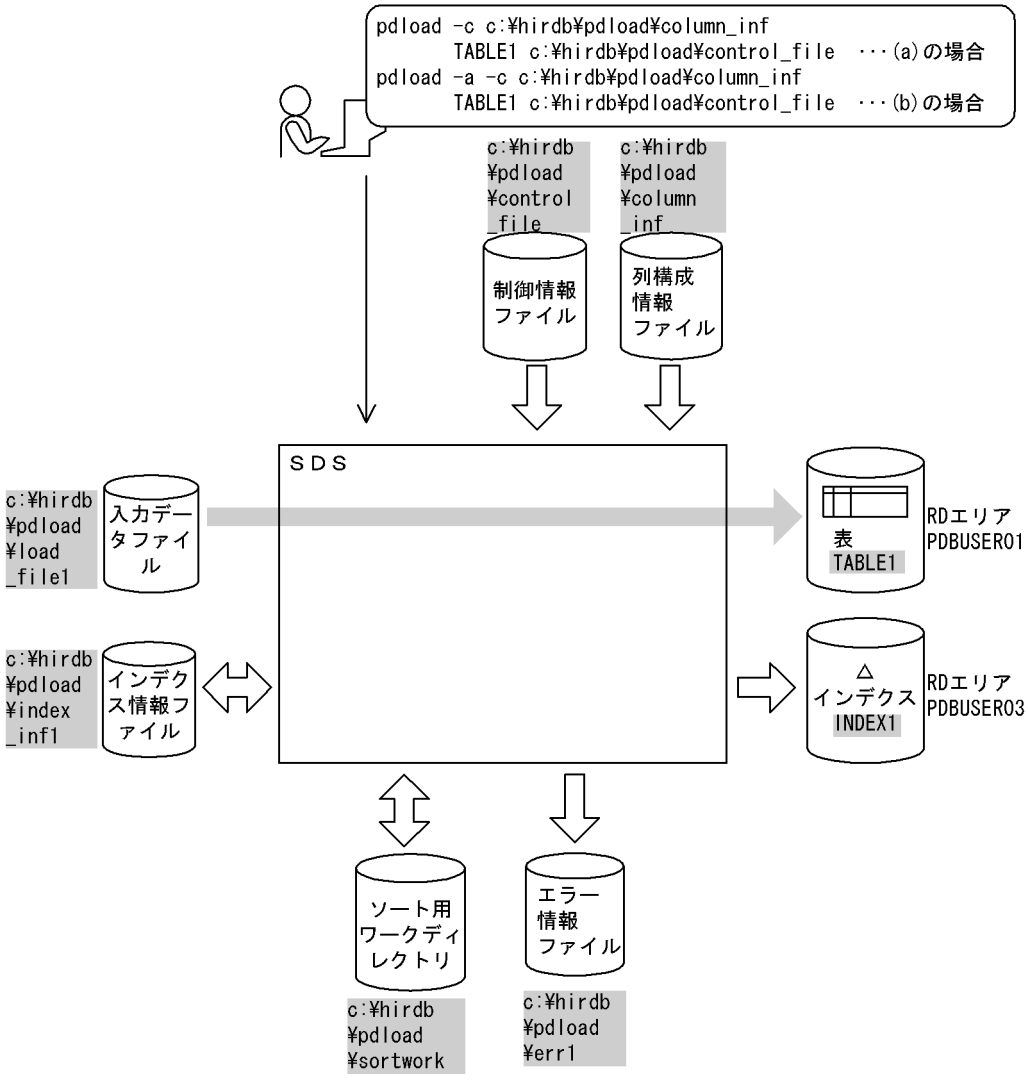
```
  CREATE INDEX INDEX1 ON TABLE1(C1) IN PDBUSER03
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

(a)　入力データファイルが DAT 形式の場合

[入力データファイルの内容]

```
1,ABCDEFG,99999,
2,a,-1,
3,XYZ,100,
    :
```

[実行するコマンドの説明]

表（TABLE1）へデータロードします。

c:¥hirdb¥pdload¥column_inf：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[列構成情報ファイル（c:¥hirdb¥pdload¥column_inf）の内容]

```
C1                  1
C2                  1
C3                  1
*skipdata*          2
```

[説明]

1. 入力データを TABLE1 の列（C1，C2，C3）の順に格納
2. 行の末尾のセパレータ文字を無視させるためのスキップデータ文

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1   1
index INDEX1 c:¥hirdb¥pdload¥index_inf1                        2
sort c:¥hirdb¥pdload¥sortwork                                  3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称
2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX1：インデクス識別子

   c:¥hirdb¥pdload¥index_inf1：インデクス情報ファイルの名称
3. ソート用ワークディレクトリの指定

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(b)　入力データファイルが固定長データ形式の場合

[入力データファイルの内容]

```
1  ABCDEFG  99999
2  a        -1
```

```
3  XYZ      100
      :
```

注　行末に改行記号があります。

[実行するコマンドの説明]

表（TABLE1）へデータロードします。

-a：入力データファイルが固定長データ形式の場合の指定

c:¥hirdb¥pdload¥column_inf：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[列構成情報ファイル（c:¥hirdb¥pdload¥column_inf）の内容]

```
C1,type=char(3)              1
C2,type=char(8)              1
C3,type=char(6)              1
*skipdata*,length=1          2
```

[説明]

1. 入力データの列のデータ型を記述
2. 改行記号を格納しないようにするためのスキップデータ文

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1   1
index INDEX1 c:¥hirdb¥pdload¥index_inf1                        2
sort c:¥hirdb¥pdload¥sortwork                                  3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX1：インデクス識別子

   c:¥hirdb¥pdload¥index_inf1：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(c)　入力データファイルが固定長データ形式固有のデータ形式の場合

表（TABLE2）にデータロードします。この例では，入力データファイルが固定長データ形式固有のデータ形式の場合について説明します。

表の定義

```
  CREATE TABLE TABLE2(C1 DECIMAL(7,2),
                      C2 CHAR(10),
                      C3 DECIMAL(10,4));
```

[入力データファイル（/usr/input_file）の内容]

バイナリエディタで参照した場合の入力データファイルの内容です。16 進数で表現されています。データの先頭から 7 バイトが C1 列に格納されるデータです。8 バイト～17 バイトが C2 列に，18 バイト～22 バイトが C3 列に格納されます。

```
303030303130304142434445464748494a00000001
202020202020206162636465666768696a00000002
303030303330302020202020202020202000000003
```

[実行するコマンドライン]

```
pdload -a -c c:¥hirdb¥pdload¥column_inf TABLE2 c:¥hirdb¥pdload¥control_file
```

[実行するコマンドの説明]

表（TABLE2）へデータロードします。

-a：入力データファイルが固定長データ形式の場合の指定

c:¥hirdb¥pdload¥column_inf：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[列構成情報ファイル（c:¥hirdb¥pdload¥column_inf）の内容]

```
C1,type=adec(7,2)           1
C2,type=char(10)            2
C3,type=bdec(4)             3
```

[説明]

1. 文字列形式の数値で記述した，小数点の無いデータの長さは 7 バイト，後方から 2 バイトは小数点以下の値として DECIMAL(7,2)に格納
2. 長さ 10 バイトのデータを CHAR(10)に格納
3. 長さ 4 バイトの領域に格納した 2 進数のデータを DECIMAL(10,4)に格納

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
option allspace=zero                                        1
source c:¥hirdb¥pdload¥input_file error=c:¥hirdb¥pdload¥err  2
```

[説明]

1. DECIMAL 列に格納する入力データが空白の場合，「0」を格納する指定
2. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

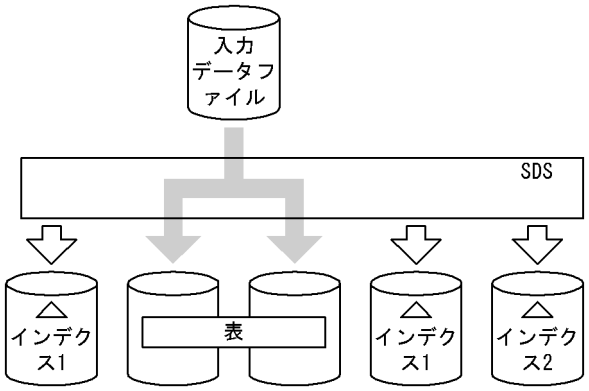
(2) 横分割表への表単位のデータロード

例 2

```
横分割表（TABLE1）に，表単位にデータロードします。
なお，次の表及びインデクスが定義されているものとします。
表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
```

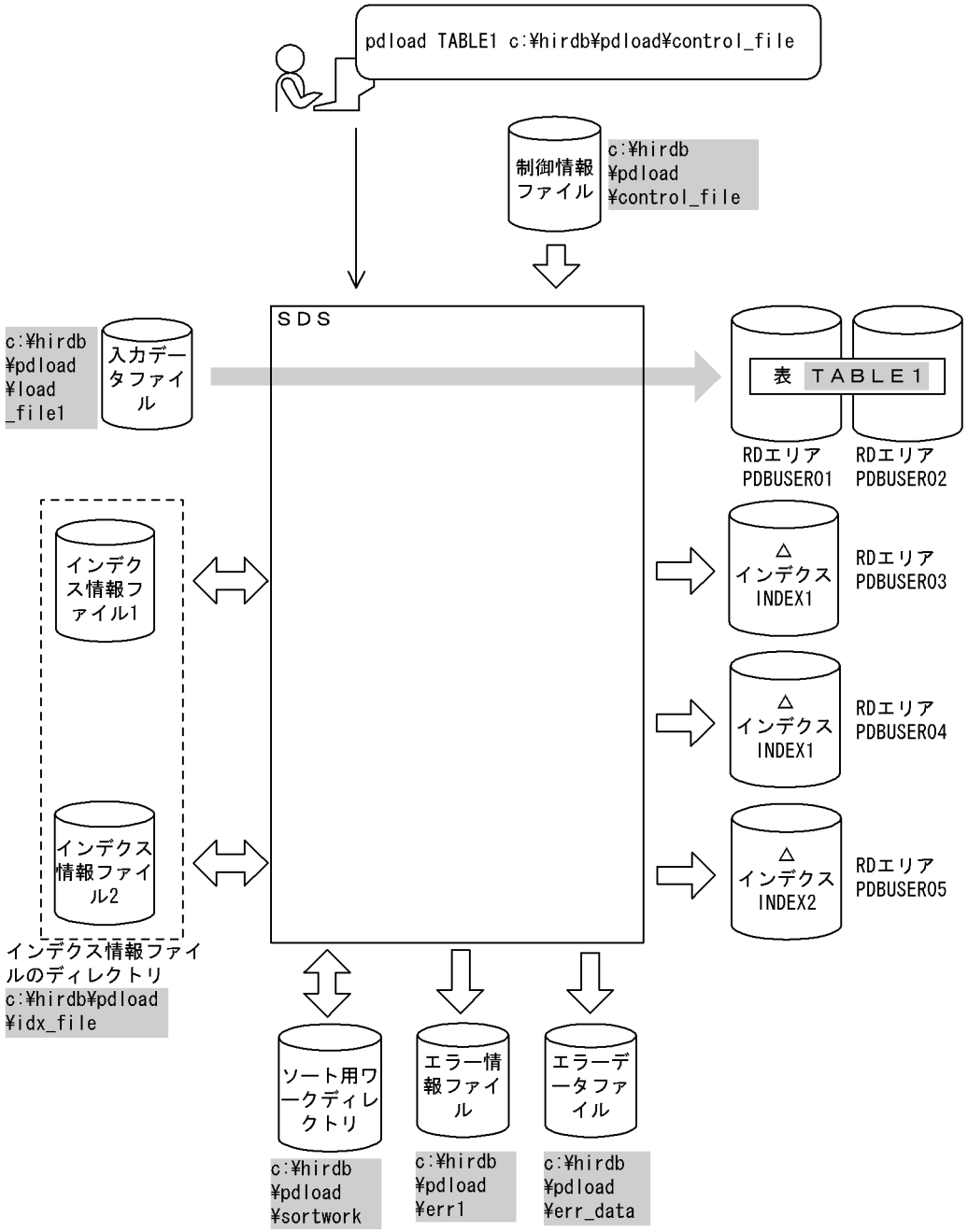
```
インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER04))
インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN (PDBUSER05)
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

横分割表（TABLE1）へ表単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1
       errdata=c:¥hirdb¥pdload¥err_data,1000                    1
idxwork c:¥hirdb¥pdload¥idx_file                                2
sort c:¥hirdb¥pdload¥sortwork                                   3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合のエラー情報とエラーとなった行データを出力するファイルの指定（ただし，入力データエラーの出力は 1000 件までとします）

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

   c:¥hirdb¥pdload¥err_data：エラーデータファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   c:¥hirdb¥pdload¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

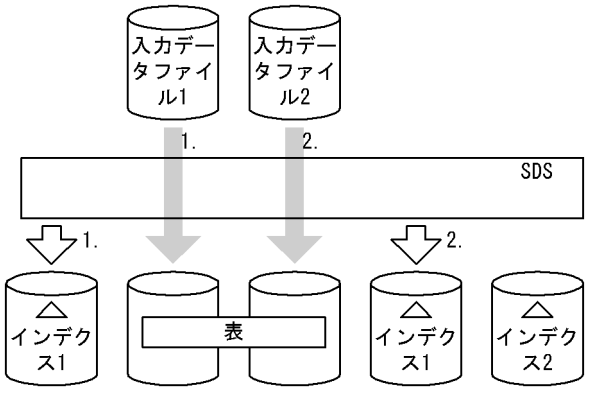
   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(3) 横分割表への RD エリア単位のデータロード

例 3

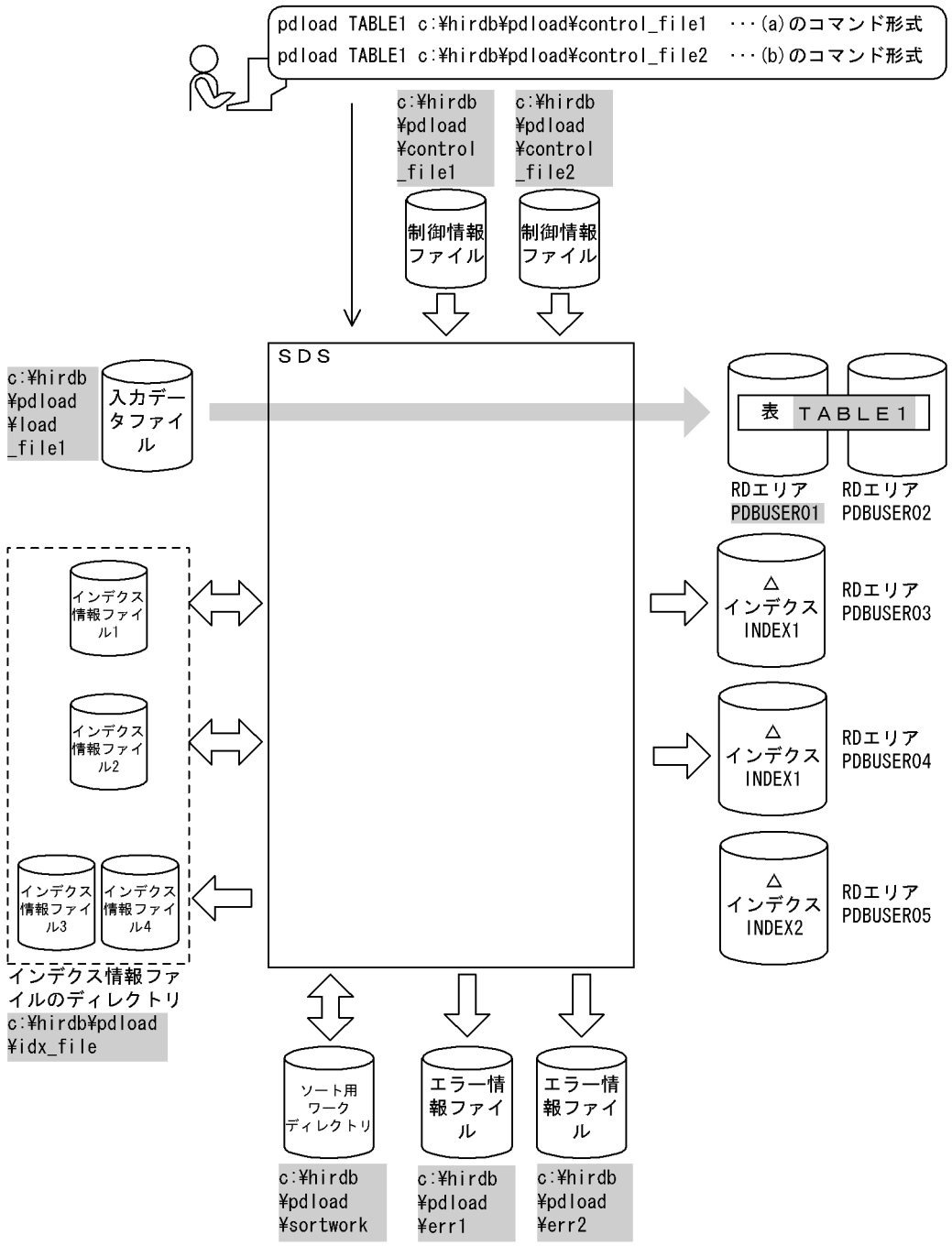
```
横分割表（TABLE1）に，RDエリア単位にデータロードします。PDBUSER01，及びPDBUSER02に対して，2回実行します。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER04))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN (PDBUSER05)
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

(a)　PDBUSER01 へのデータロード

[実行するコマンドの説明]

横分割表（TABLE1）へ RD エリア（PDBUSER01）単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source PDBUSER01 c:¥hirdb¥pdload¥load_file1
       error=c:¥hirdb¥pdload¥err1                    1
idxwork c:¥hirdb¥pdload¥idx_file                     2
sort c:¥hirdb¥pdload¥sortwork                        3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   PDBUSER01：入力データを格納する表がある RD エリアの名称

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   c:¥hirdb¥pdload¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(b) PDBUSER02 へのデータロード

[実行するコマンドの説明]

横分割表（TABLE1）へ RD エリア（PDBUSER02）単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source PDBUSER02 c:¥hirdb¥pdload¥load_file2
       error=c:¥hirdb¥pdload¥err2                    1
idxwork c:¥hirdb¥pdload¥idx_file                     2
sort c:¥hirdb¥pdload¥sortwork                        3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   PDBUSER02：入力データを格納する表がある RD エリアの名称

   c:¥hirdb¥pdload¥load_file2：入力データファイルの名称

   c:¥hirdb¥pdload¥err2：エラー情報ファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   c:¥hirdb¥pdload¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(c) INDEX2 の作成

INDEX2 はこのままでは使用できません。

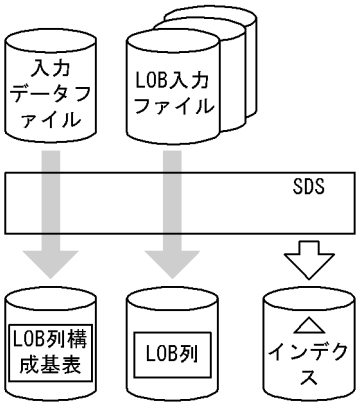
インデクス情報を出力したインデクス情報ファイル 3，及びインデクス情報ファイル 4 を使用して，pdrorg でインデクスの一括作成（-k ixmk）をしてください。

(4)　LOB 列構成基表と LOB 列のデータロード

例 4

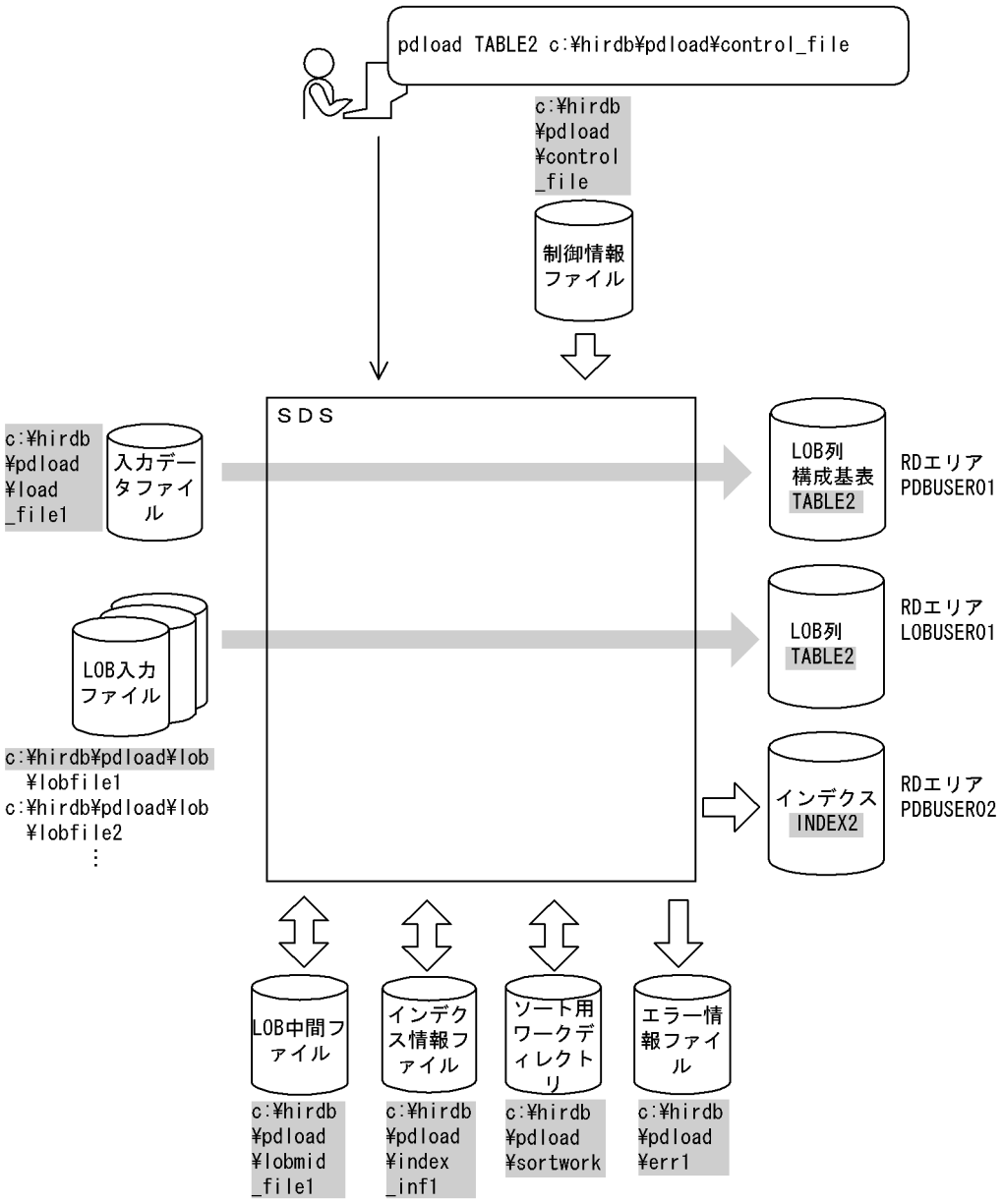
```
LOB列がある表（TABLE2）に対して，LOB列構成基表とLOB列を同時にデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT,C2 BLOB IN LOBUSER01) IN PDBUSER01
・インデクスの定義
  CREATE INDEX INDEX2 ON TABLE2(C1) IN PDBUSER02
```

［概要］



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

LOB 列がある表（TABLE2）へ表単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1   1
index INDEX2 c:¥hirdb¥pdload¥index_inf1                        2
sort c:¥hirdb¥pdload¥sortwork                                  3
lobdata c:¥hirdb¥pdload¥lob                                    4
lobmid c:¥hirdb¥pdload¥lobmid_file1                            5
```

[説明]

1. 入力データファイル，及び入力データにエラーがあった場合にエラーを出力するファイルの指定

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   入力データファイルの内容（DAT 形式）

   ```
   1,lobfile1
   2,lobfile2
   3,lobfile3
   ```

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   c:¥hirdb¥pdload¥index_inf1：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. LOB 入力ファイルのディレクトリの指定

   c:¥hirdb¥pdload¥lob：LOB 入力ファイルのパス名

5. LOB 列構成基表データロード時に LOB 列のデータロードに必要な情報を出力するファイルの指定（これを LOB 列のデータロード時にそのまま使用します）

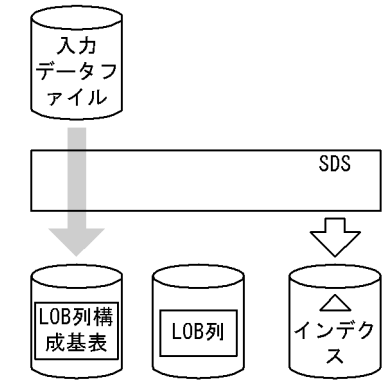
   c:¥hirdb¥pdload¥lobmid_file1：LOB 中間ファイルの名称

(5) LOB 列構成基表だけのデータロード

例 5

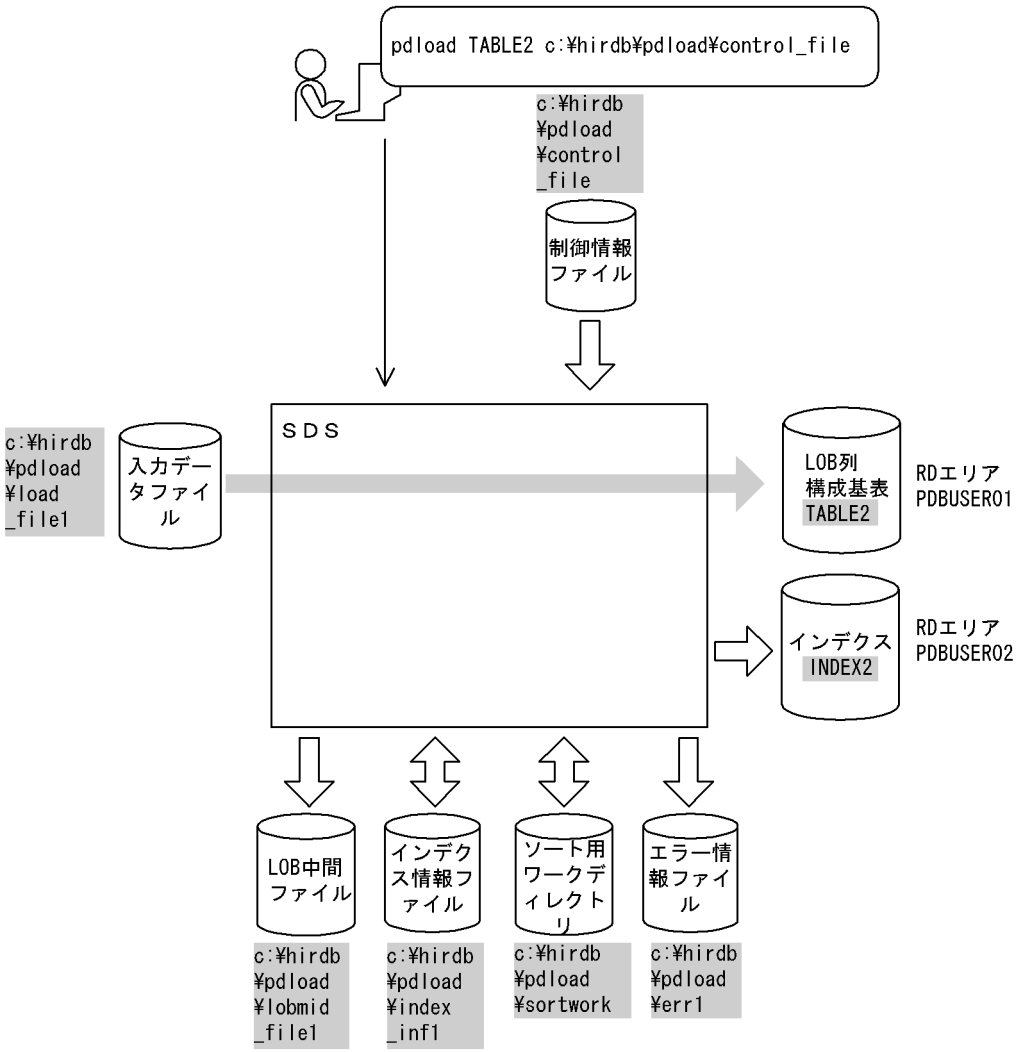
```
LOB列がある表（TABLE2）に対して，LOB列構成基表だけをデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT,C2 BLOB IN LOBUSER01) IN PDBUSER01
・インデクスの定義
  CREATE INDEX INDEX2 ON TABLE2(C1) IN PDBUSER02
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

LOB 列がある表（TABLE2）へ表単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1   1
index INDEX2 c:¥hirdb¥pdload¥index_inf1                        2
sort c:¥hirdb¥pdload¥sortwork                                  3
lobmid c:¥hirdb¥pdload¥lobmid_file1                            4
```

[説明]

1. 入力データファイル，及び入力データにエラーがあった場合にエラーを出力するファイルの指定

c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

入力データファイルの内容（DAT 形式）

```
1,lobfile1
```

2,lobfile2

3,lobfile3

c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   c:¥hirdb¥pdload¥index_inf1：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. LOB 列構成基表データロード時に LOB 列のデータロードに必要な情報を出力するファイルの指定

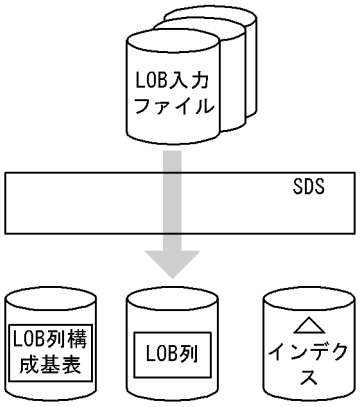
   c:¥hirdb¥pdload¥lobmid_file1：LOB 中間ファイルの名称

(6) LOB 列だけのデータロード

例 6

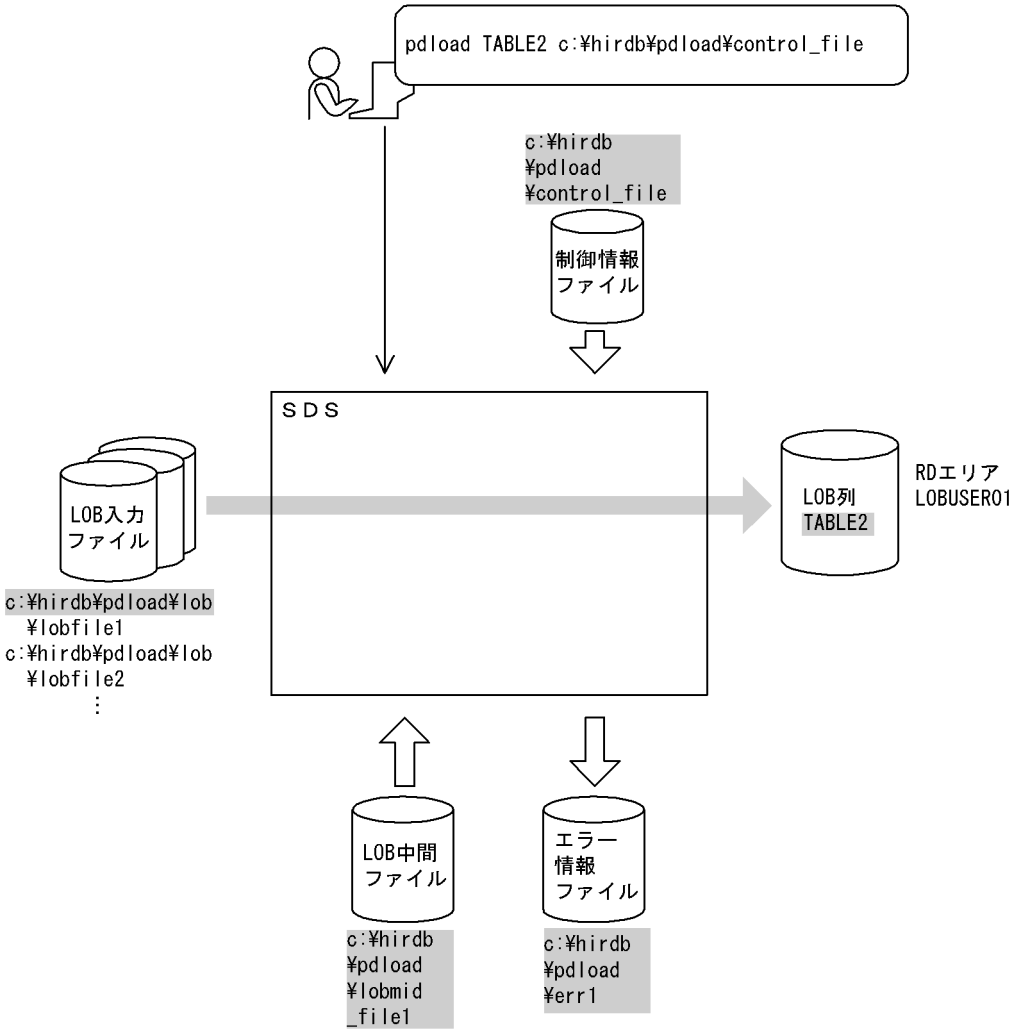
```
LOB列がある表（TABLE2）に対して，LOB列だけをデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT,C2 BLOB IN LOBUSER01) IN PDBUSER01
・インデクスの定義
  CREATE INDEX INDEX2 ON TABLE2(C1) IN PDBUSER02
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

LOB 列がある表（TABLE2）へ表単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
lobdata c:¥hirdb¥pdload¥lob                                          1
lobmid c:¥hirdb¥pdload¥lobmid_file1 error=c:¥hirdb¥pdload¥err1   2
```

[説明]

1. LOB 入力ファイルのディレクトリの指定

   c:¥hirdb¥pdload¥lob：LOB 入力ファイルのパス名

   LOB 列構成基表データロード時に次の入力データファイルを使用したものとします。

   入力データファイルの内容（DAT 形式）

   ```
   1,lobfile1
   ```

2,lobfile2

3,lobfile3

2. LOB 列構成基表データロード時に LOB 列のデータロードに必要な情報を出力したファイル，及び LOB 入力エラーがあった場合にエラーを出力するファイルの指定

c:¥hirdb¥pdload¥lobmid_file1：LOB 中間ファイルの名称

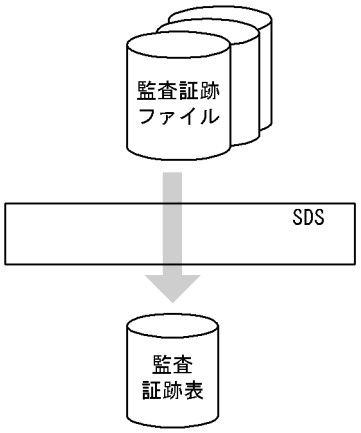
c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

(7) 監査証跡表へのデータロード

例 7

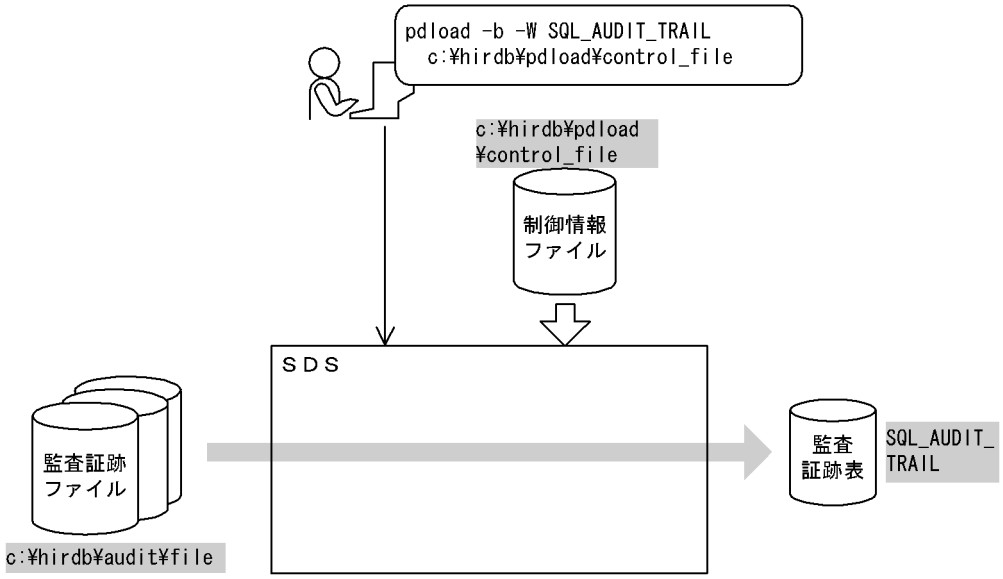
監査証跡表（SQL_AUDIT_TRAIL）に対して，データロード待ちの監査証跡ファイルをデータロードします。なお，監査証跡表にはインデクスが定義されていないものとします。

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース作成ユティリティでの指定値
ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

監査証跡表（SQL_AUDIT_TRAIL）に対して，データロード待ちの監査証跡ファイル（c:¥hirdb¥audit¥file）をデータロードします。

-b 及び-W：監査証跡表へデータロードする場合に必ず指定するオプションです。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source (uoc)                                          1
srcuoc param='dir=c:¥hirdb¥audit¥file,file=all'       2
```

[説明]

1. 監査証跡データ登録 UOC を使用する場合の指定
2. 監査証跡データ登録 UOC に渡すパラメタの指定

   dir=c:¥hirdb¥audit¥file：監査証跡ファイルの HiRDB ファイル名

   file=all：データロード待ちのすべての監査証跡ファイルを対象とするため，all を指定

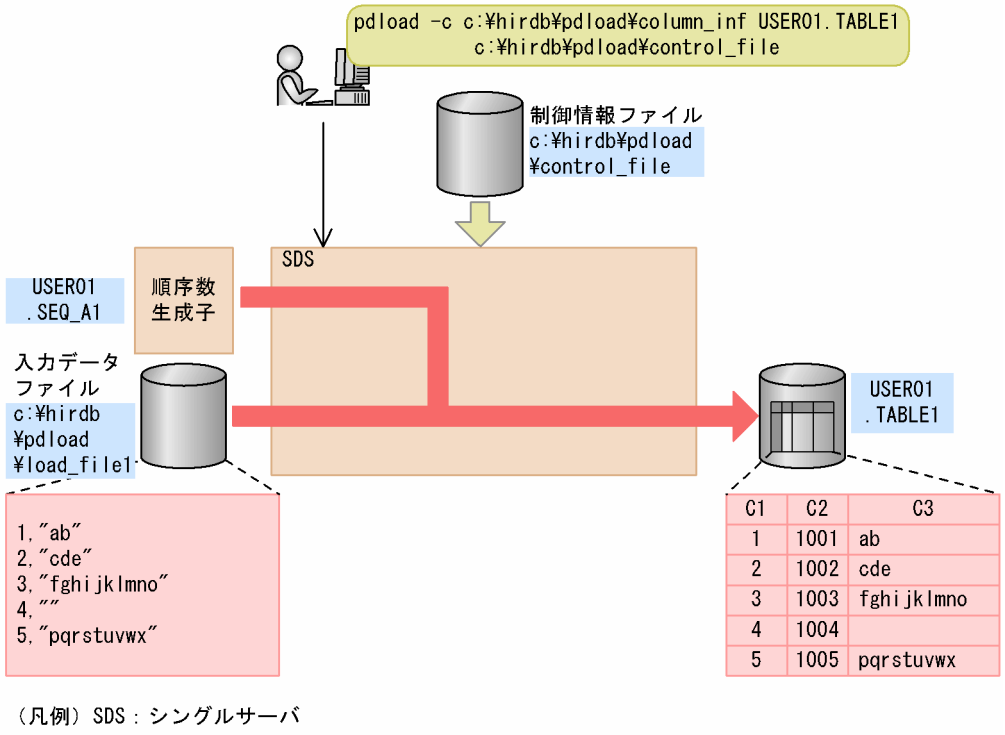
(8)　自動採番機能を使用したデータロード

例 8

表（TABLE1）に，指定単位取得方式でデータロードします。このとき，C2列に列データとして順序番号を追加します。
なお，次の表及び順序数生成子が定義されているものとします。
・表の定義
  CREATE TABLE USER01.TABLE1(C1 INT,C2 INT,C3 VARCHAR(10)) IN PDBUSER01
・順序数生成子の定義

```
CREATE SEQUENCE USER01.SEQ_A1
  AS INTEGER START WITH 1 INCREMENT BY 1 MAXVALUE 9999 CYCLE IN PDBUSER01
```

[入出力ファイルと RD エリアの関連図]



（凡例）SDS：シングルサーバ

[実行するコマンドの説明]

表（TABLE1）へデータロードします。

c:¥hirdb¥pdload¥column_inf：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[列構成情報ファイル（c:¥hirdb¥pdload¥column_inf）の内容]

```
C1                                  1
C3                                  1
*adddata*                           2
C2,sequence=(USER01.SEQ_A1)         3
```

[説明]

1. 入力データを TABLE1 の列（C1，C3）に格納
2. これ以降の指定が入力データの追加であることを示す追加データ文
3. 順序数生成子（USER01.SEQ_A1）が生成した順序番号を列 C2 に格納

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source c:¥hirdb¥pdload¥load_file1     1
option seq_range=1000                    2
```

[説明]

1. 入力データファイルの指定

c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

2. 順序番号の取得方式の指定（指定単位取得方式）

seq_range=1000：データロード時に 1000 個ずつ順序番号を取得するための指定

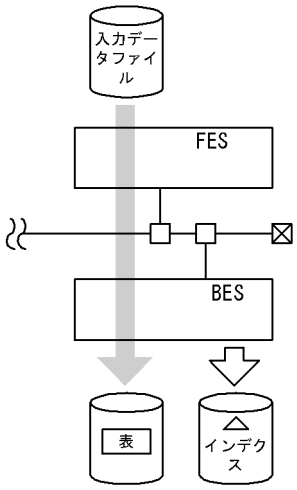
5.14.2　HiRDB/パラレルサーバの場合

(1)　表へのデータロード

例 9

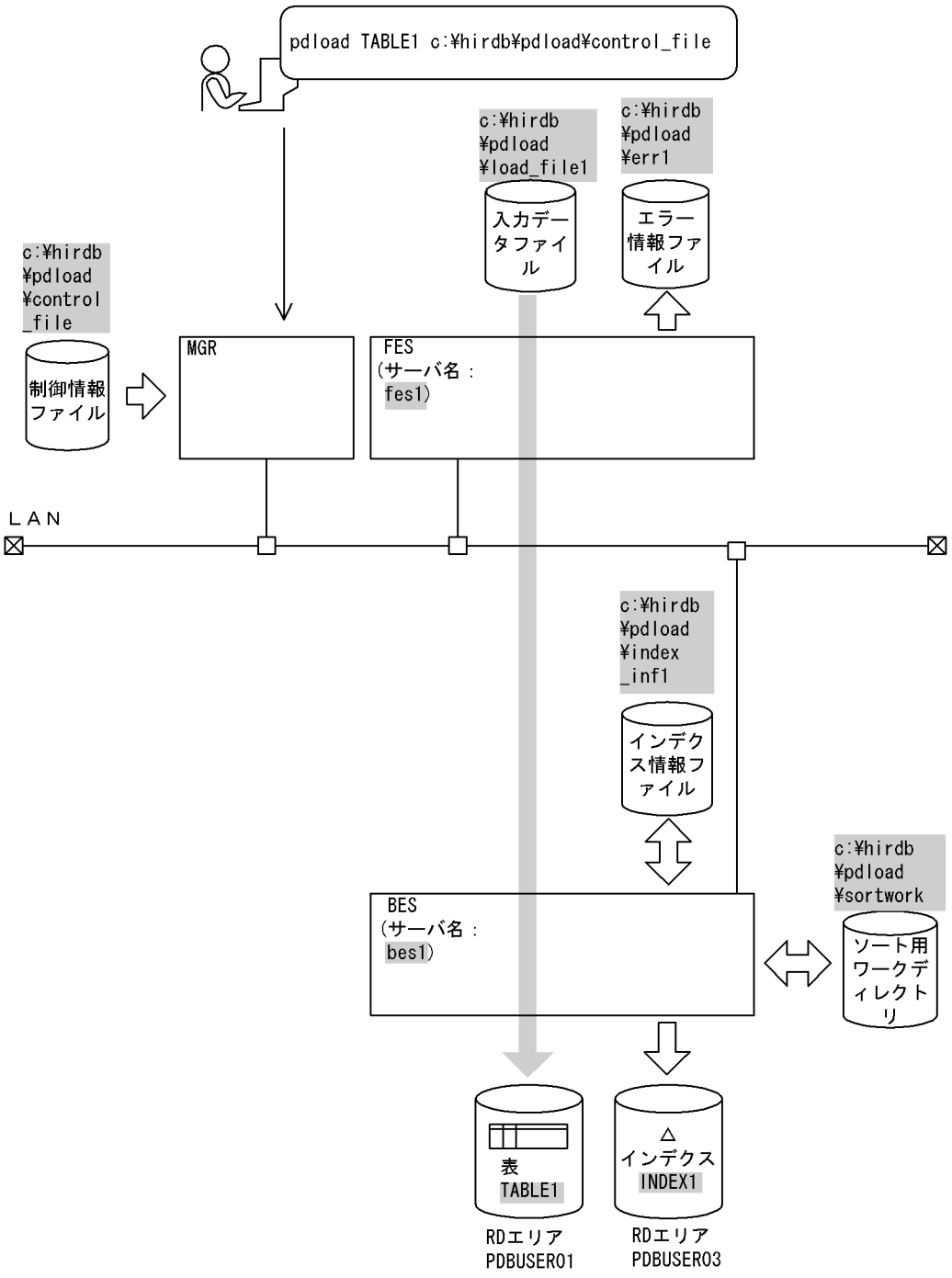
```
表（TABLE1）にデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
  IN PDBUSER01
・インデクスの定義
  CREATE INDEX INDEX1 ON TABLE1(C1) IN PDBUSER03
```

[概要]



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

表（TABLE1）へデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source fes1:c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1   1
index INDEX1 c:¥hirdb¥pdload¥index_inf1                             2
sort bes1 c:¥hirdb¥pdload¥sortwork                                  3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   fes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX1：インデクス識別子

   c:¥hirdb¥pdload¥index_inf1：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   bes1：ソート用ワークファイルを作成するサーバの名称

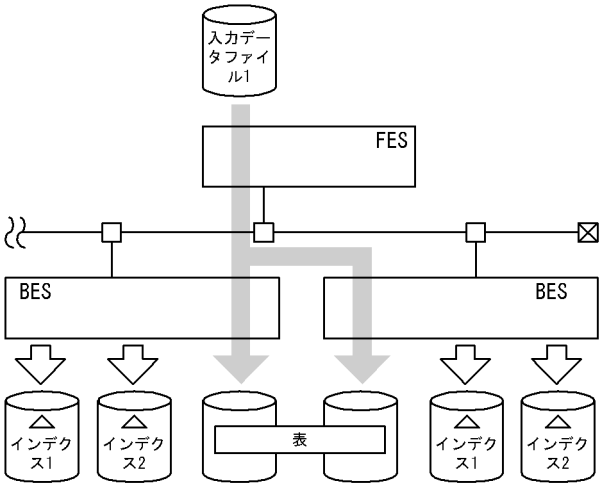
   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(2) 横分割表への表単位のデータロード

例 10

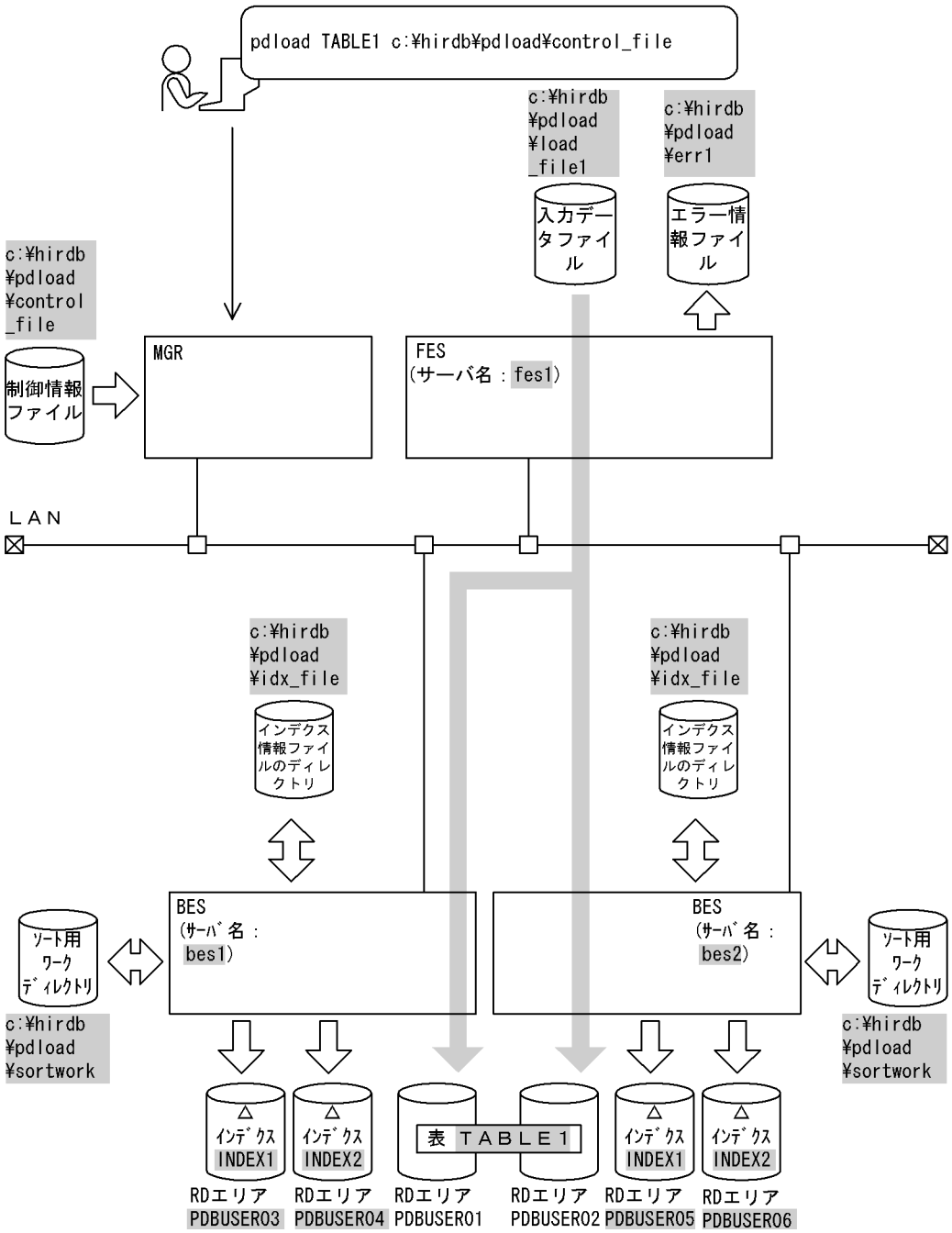
```
横分割表（TABLE1）に，表単位にデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

横分割表（TABLE1）へ表単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source fes1:c:¥hirdb¥pdload¥load_file1 error=c:¥hirdb¥pdload¥err1   1
idxwork bes1 c:¥hirdb¥pdload¥idx_file                               2
```

```
idxwork bes2 c:¥hirdb¥pdload¥idx_file                                    2
sort bes1 c:¥hirdb¥pdload¥sortwork                                       3
sort bes2 c:¥hirdb¥pdload¥sortwork                                       3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   fes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdload¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

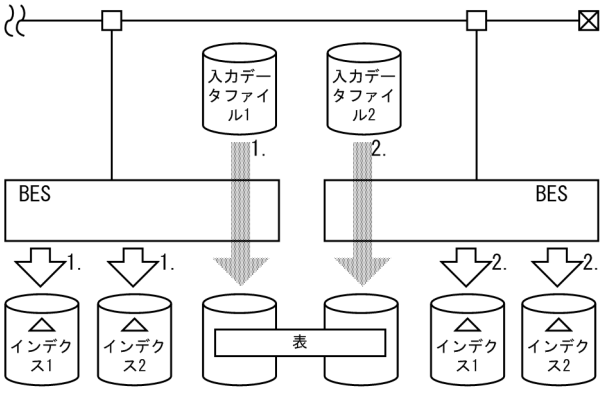
   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(3) 横分割表への RD エリア単位のデータロード

例 11

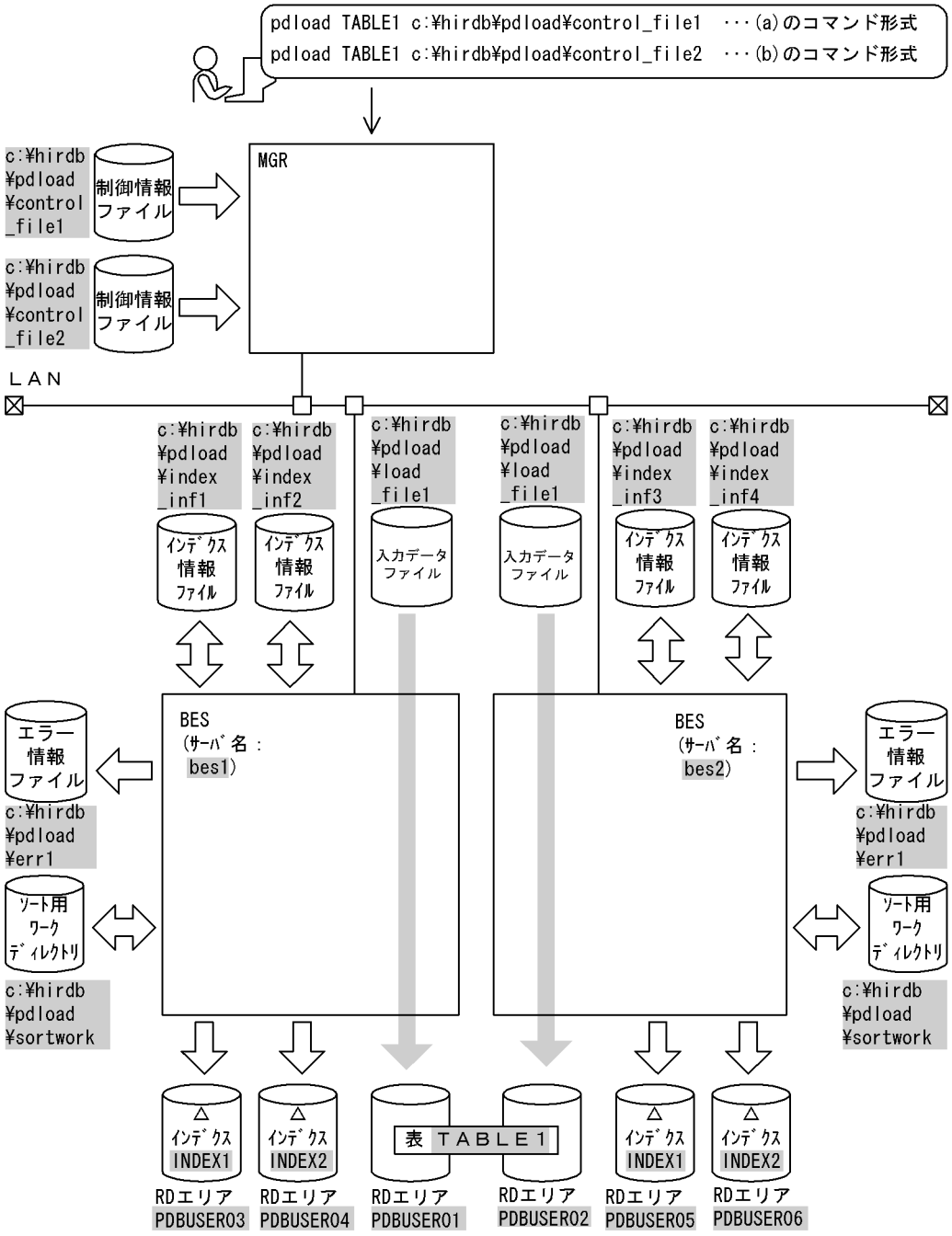
```
横分割表（TABLE1）に，RDエリア単位にデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

(a)　PDBUSER01 へのデータロード

[実行するコマンドの説明]

横分割表（TABLE1）へ RD エリア（PDBUSER01）単位にデータロードします。

c:¥hirdb¥pdload¥control_file1：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file1）の内容]

```
source PDBUSER01 bes1:c:¥hirdb¥pdload¥load_file1
       error=c:¥hirdb¥pdload¥err1                          1
index INDEX1 PDBUSER03 c:¥hirdb¥pdload¥index_inf1          2
index INDEX2 PDBUSER04 c:¥hirdb¥pdload¥index_inf2          2
sort bes1 c:¥hirdb¥pdload¥sortwork                         3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   PDBUSER01：入力データを格納する表がある RD エリアの名称

   bes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX1，INDEX2：インデクス識別子

   PDBUSER03，PDBUSER04：インデクスを格納する RD エリアの名称

   c:¥hirdb¥pdload¥index_inf1，c:¥hirdb¥pdload¥index_inf2：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   bes1：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(b)　PDBUSER02 へのデータロード

[実行するコマンドの説明]

横分割表（TABLE1）へ RD エリア（PDBUSER02）単位にデータロードします。

c:¥hirdb¥pdload¥control_file2：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file2）の内容]

```
source PDBUSER02 bes2:c:¥hirdb¥pdload¥load_file2
       error=c:¥hirdb¥pdload¥err2                          1
index INDEX1 PDBUSER05 c:¥hirdb¥pdload¥index_inf3          2
index INDEX2 PDBUSER06 c:¥hirdb¥pdload¥index_inf4          2
sort bes2 c:¥hirdb¥pdload¥sortwork                         3
```

[説明]

1. 入力データファイル，及び入力データエラーがあった場合にエラーを出力するファイルの指定

   PDBUSER02：入力データを格納する表がある RD エリアの名称

   bes2：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file2：入力データファイルの名称

   c:¥hirdb¥pdload¥err2：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX1，INDEX2：インデクス識別子

   PDBUSER05，PDBUSER06：インデクスを格納する RD エリアの名称

   c:¥hirdb¥pdload¥index_inf3，c:¥hirdb¥pdload¥index_inf4：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   bes2：ソート用ワークファイルを作成するサーバの名称

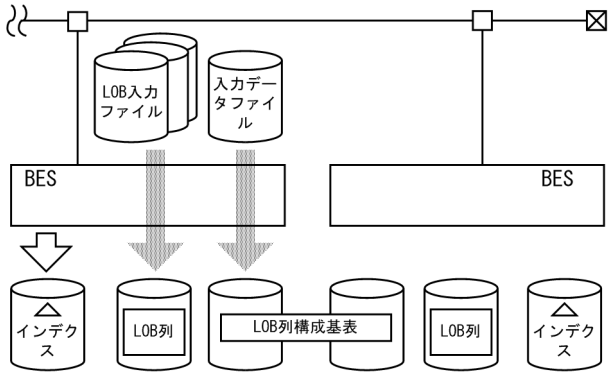
   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(4) LOB 列がある表へのデータロード（LOB 列構成基表と LOB 列を同時にデータロード）

例 12

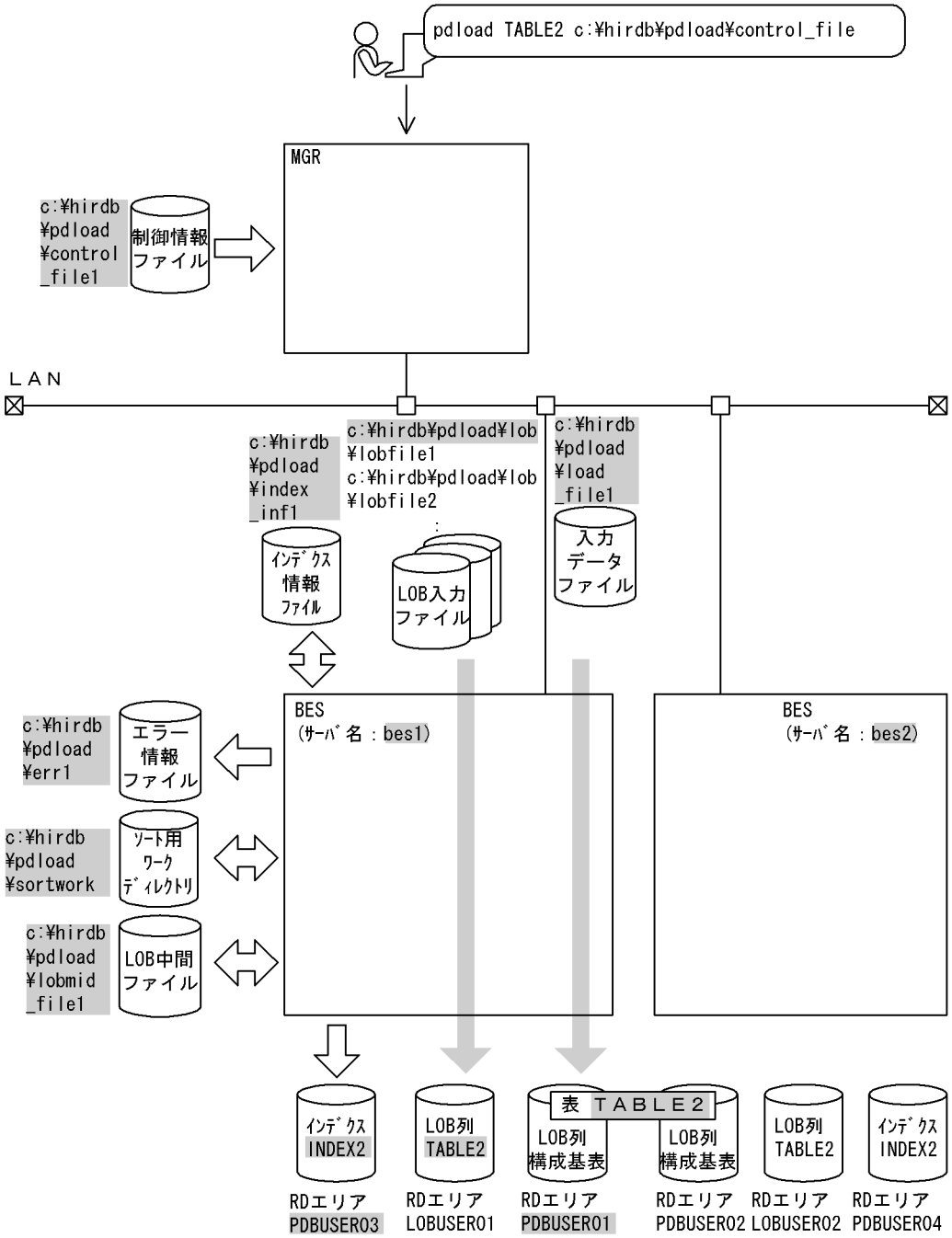
```
LOB列がある表（TABLE2）に対して，RDエリア単位にデータロードします。このとき，LOB列構成基表と
LOB列を同時にデータロードします。
なお，次の表が定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT NOT NULL,C2 BLOB IN ((LOBUSER01),(LOBUSER02)))
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE2(C1) IN((PDBUSER03),(PDBUSER04))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

LOB 列がある表（TABLE2）へ RD エリア（PDBUSER01，LOBUSER01）単位にデータロードします。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source PDBUSER01 bes1:c:¥hirdb¥pdload¥load_file1
       error=c:¥hirdb¥pdload¥err1                          1
```

```
index INDEX2 PDBUSER03 c:¥hirdb¥pdload¥index_inf1              2
sort bes1 c:¥hirdb¥pdload¥sortwork                              3
lobdata c:¥hirdb¥pdload¥lob                                     4
lobmid c:¥hirdb¥pdload¥lobmid_file1                             5
```

[説明]

1. 入力データファイル，及び入力データにエラーがあった場合にエラーを出力するファイルの指定

   PDBUSER01：入力データを格納する表がある RD エリアの名称

   bes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

   入力データファイルの内容（DAT 形式）

   ```
   1,lobfile1
   2,lobfile2
   3,lobfile3
   ```

   c:¥hirdb¥pdload¥err1：エラー情報ファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   PDBUSER03：インデクスを格納する RD エリアの名称

   c:¥hirdb¥pdload¥index_inf1：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   bes1：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. LOB 入力ファイルのディレクトリの指定

   c:¥hirdb¥pdload¥lob：LOB 入力ファイルのパス名

5. LOB 列構成基表データロード時に LOB 列のデータロードに必要な情報を出力するファイルの指定（これを LOB 列のデータロード時にそのまま使用します）

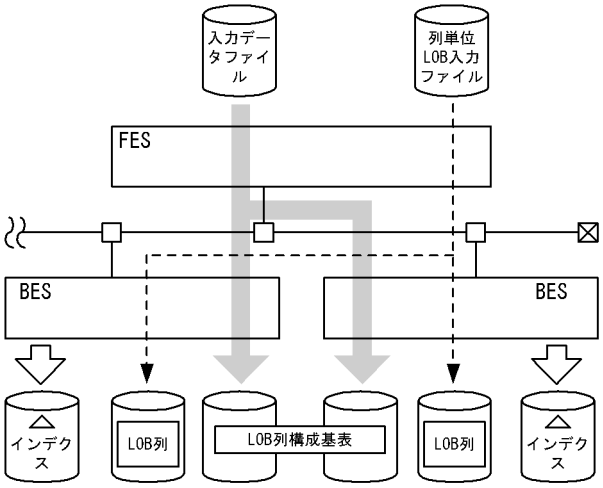
   c:¥hirdb¥pdload¥lobmid_file1：LOB 中間ファイルの名称

(5) LOB 列がある表へのデータロード（列単位 LOB 入力ファイルを使用した場合）

例 13

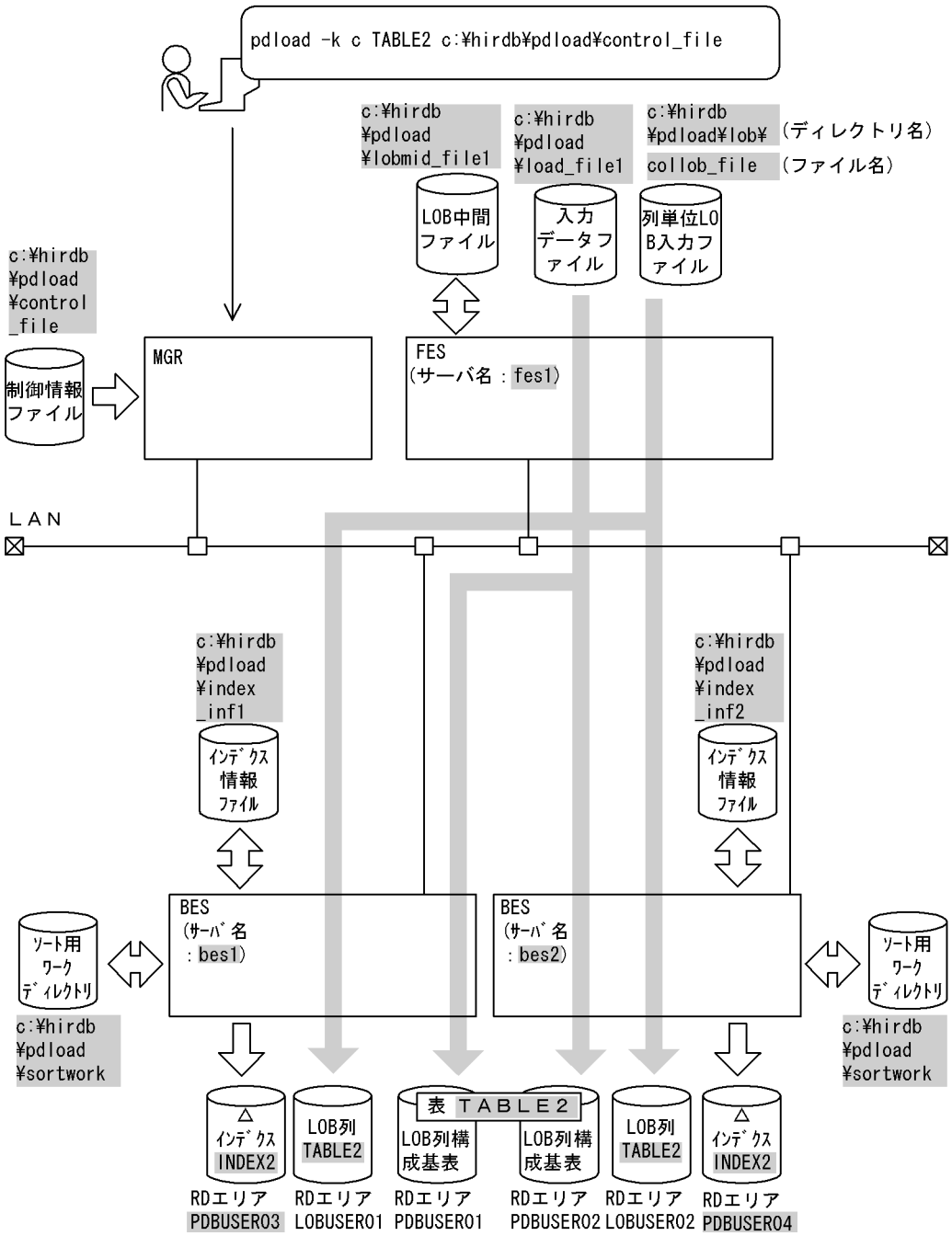
```
LOB列がある表（TABLE2）にデータロードします。このとき，LOB列構成基表とLOB列を同時にデータロードし，LOB列へのデータ格納は列単位LOB入力ファイルから行います。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT NOT NULL,C2 BLOB IN ((LOBUSER01),(LOBUSER02)))
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE2(C1) IN((PDBUSER03),(PDBUSER04))
```

[概要]



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

LOB 列がある表（TABLE2）へデータロードします。

-k c：列単位 LOB 入力ファイルを使用する場合の指定

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source fes1:c:¥hirdb¥pdload¥load_file1               1
index INDEX2 PDUSER03 c:¥hirdb¥pdload¥index_inf1     2
index INDEX2 PDUSER04 c:¥hirdb¥pdload¥index_inf2     2
sort bes1 c:¥hirdb¥pdload¥sortwork                   3
sort bes2 c:¥hirdb¥pdload¥sortwork                   3
lobdata c:¥hirdb¥pdload¥collob                       4
lobcolumn collob_file                                5
lobmid c:¥hirdb¥pdload¥lobmid_file                   6
```

[説明]

1. 入力データファイルの指定

   fes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   PDUSER03，PDUSER04：インデクスが格納されている RD エリアの名称

   c:¥hirdb¥pdload¥index_inf1，c:¥hirdb¥pdload¥index_inf2：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. 列単位 LOB 入力ファイルがあるディレクトリの指定

   c:¥hirdb¥pdload¥collob：列単位 LOB 入力ファイルがあるディレクトリの名称

5. 列単位 LOB 入力ファイルの指定

   collob_file：列単位 LOB 入力ファイルの名称

6. LOB 中間ファイルの指定

   c:¥hirdb¥pdload¥lobmid_file：LOB 中間ファイルの名称

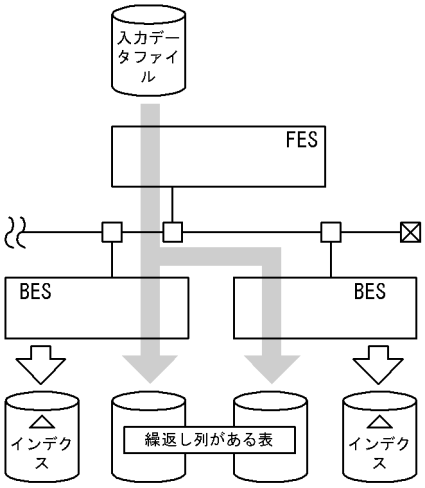
(6) 繰返し列がある表へのデータロード

例 14

繰返し列がある表（成績表）に，表単位にデータロードします。このとき，クラスタキー順のチェックをしないようにします。
なお，次の表が定義されているものとします。
・表の定義

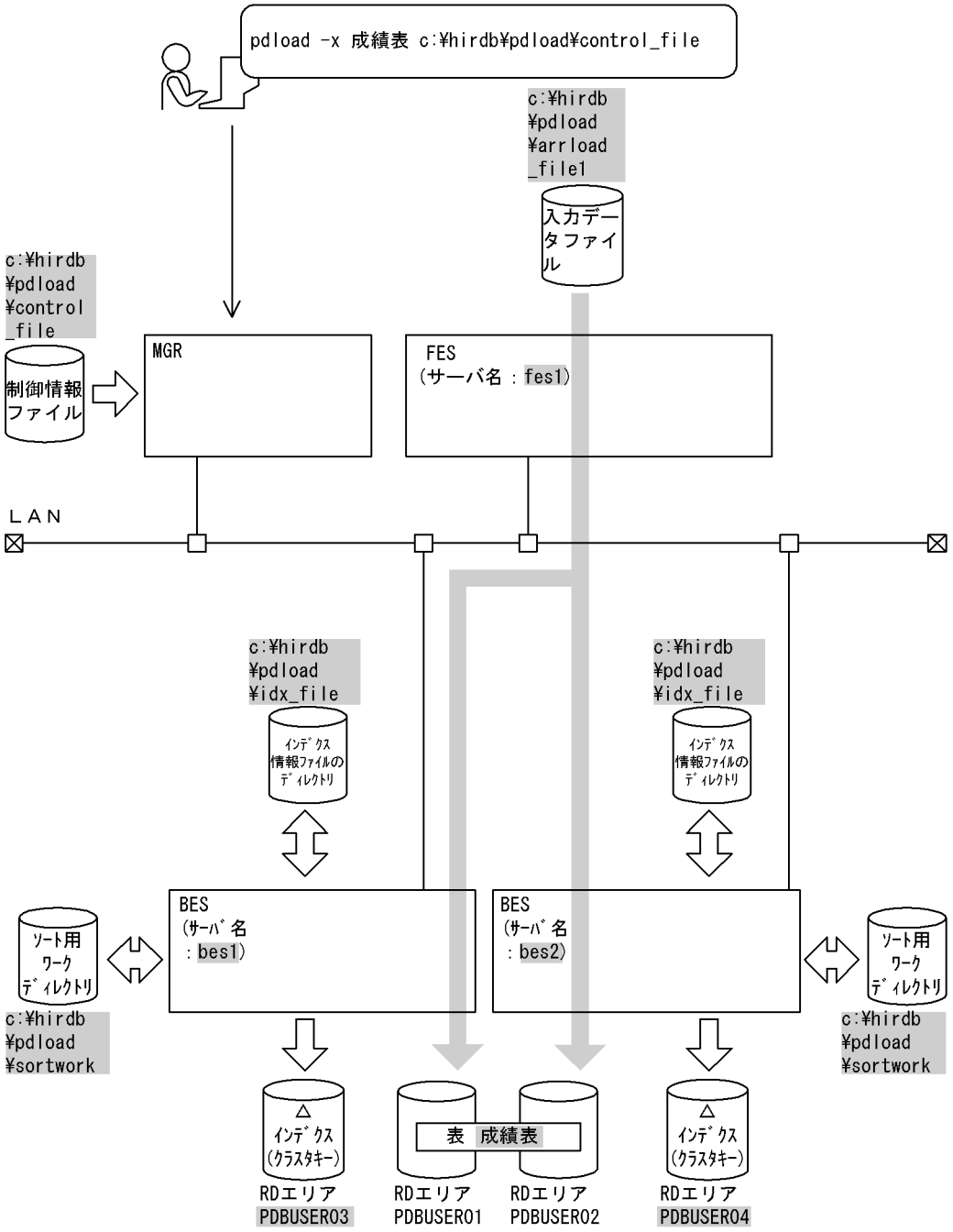
```
CREATE TABLE 成績表
    (出席番号 CHAR(6) UNIQUE CLUSTER KEY IN ((PDUSER03),(PDUSER04)),
     氏名 NCHAR(10),
     科目名 NCHAR(8) ARRAY[5],
     成績 INT ARRAY[5])
  IN ((PDUSER01) 出席番号 <= '1Z9999',(PDUSER02))
```

[概要]



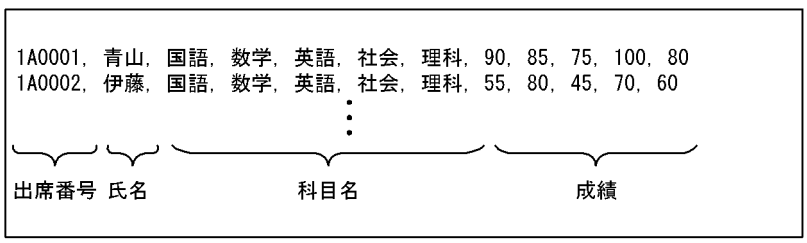
（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

[入力データファイルの内容（DAT 形式）]



[実行するコマンドの説明]

繰返し列がある表（成績表）へ表単位にデータロードします。

-x：クラスタキー順のチェックをしない指定

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source fes1:c:¥hirdb¥pdload¥arrload_file1     1
idxwork bes1 c:¥hirdb¥pdload¥idx_file         2
idxwork bes2 c:¥hirdb¥pdload¥idx_file         2
sort bes1 c:¥hirdb¥pdload¥sortwork            3
sort bes2 c:¥hirdb¥pdload¥sortwork            3
array elmtype=ff nullset=c                    4
```

[説明]

1. 入力データファイルの指定

   fes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥arrload_file1：入力データファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdload¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdload¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. 繰返し列がある表にデータロードする場合の指定

   ff：配列データ形式

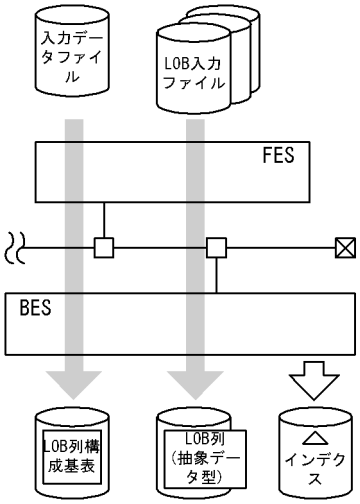
   c：該当する要素に何も格納しない場合の指定

(7) プラグイン使用時の表へのデータロード

例 15

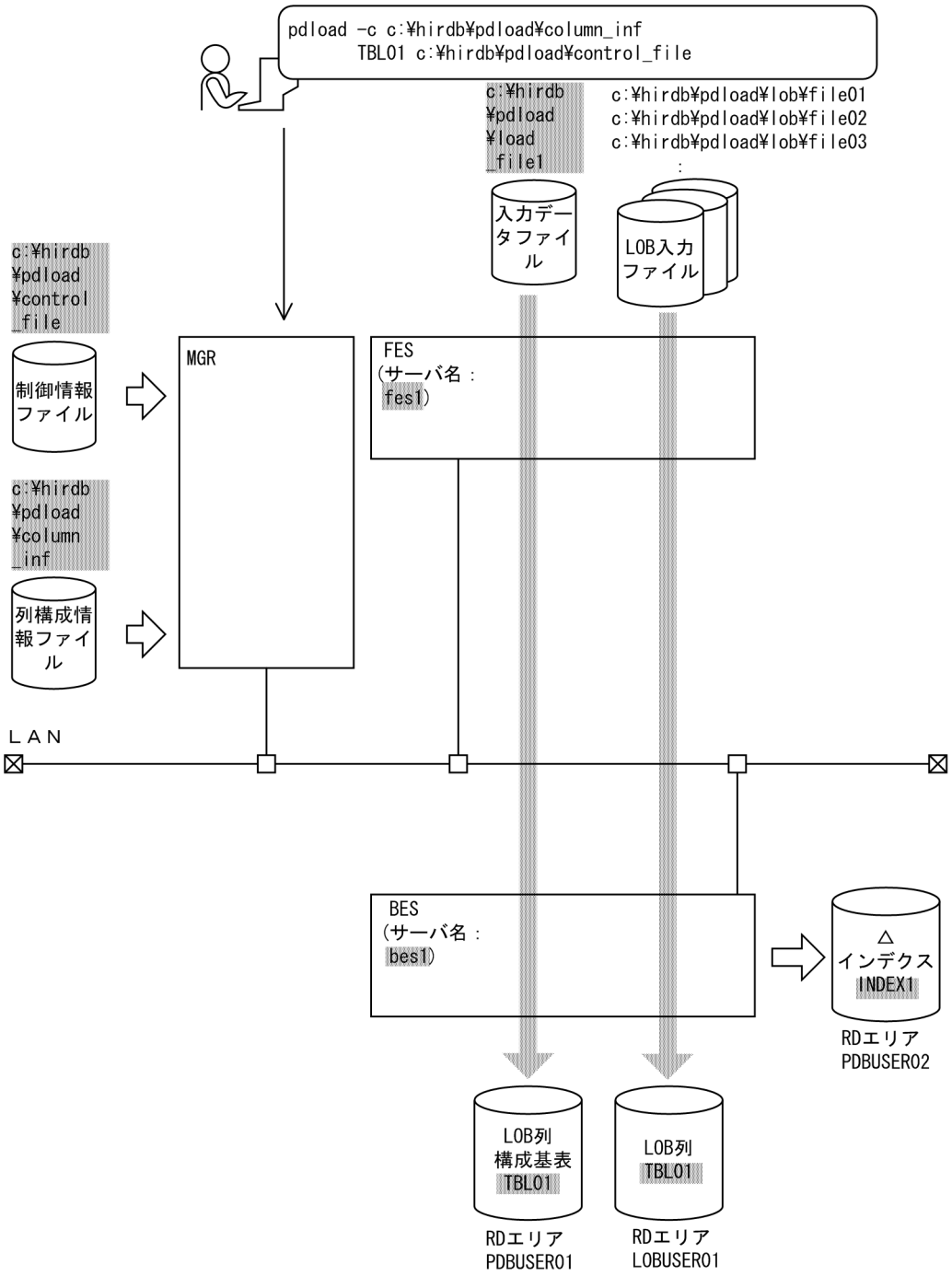
```
抽象データ型（SGMLTEXT）がある表（TBL01）にデータロードします。この例では，HiRDB Text Sea
rch Plug-inを使用した場合について説明します。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TBL01(C1 INTEGER,C2 VARCHAR(10),C3 SGMLTEXT ALLOCATE
                          (SGMLTEXT IN LOBUSER01) PLUGIN …)
                        IN PDBUSER01
・プラグインインデクスの定義
  CREATE INDEX INDEX1 USING TYPE NGRAM ON TBL01(C3) IN PDBUSER02
また，SGMLTEXT型のコンストラクタ関数の内容を次に示します。
・関数名：SGMLTEXT　入力：BLOB　出力：SGMLTEXT型
```

[概要]



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
MGR：システムマネジャ
FES：フロントエンドサーバ
BES：バックエンドサーバ

[入力データファイルの内容（DAT 形式）]

```
1,"HiRDB",c:¥hirdb¥pdload¥lob¥file01
-1,"",c:¥hirdb¥pdload¥lob¥file02
999,"HITACHI",c:¥hirdb¥pdload¥lob¥file03
        :
```

[実行するコマンドの説明]

抽象データ型がある表（TBL01）へデータロードします。

c:¥hirdb¥pdload¥column_inf：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[列構成情報ファイル（c:¥hirdb¥pdload¥column_inf）の内容]

```
C1                                  1
C2                                  1
C3,func=(SGMLTEXT,param=blob)       1,2
```

[説明]

1. 入力データを TBL01 の列（C1，C2，C3）の順に格納
2. コンストラクタ関数を呼び出して，抽象データ型の列（C3）にデータを格納

   SGMLTEXT：コンストラクタ関数名

   blob：コンストラクタ関数に渡す引数の型

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source fes1:c:¥hirdb¥pdload¥load_file1    1
lobdata                                   2
```

[説明]

1. 入力データファイルの指定

   fes1：入力データファイルがあるサーバの名称

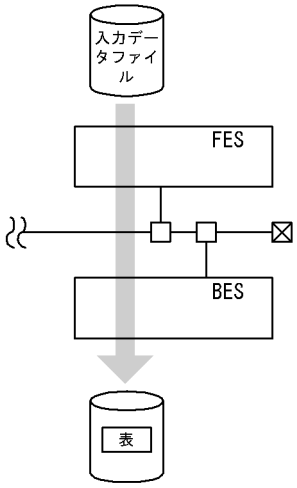
   c:¥hirdb¥pdload¥load_file1：入力データファイルの名称

2. LOB 入力ファイルの指定

例 16

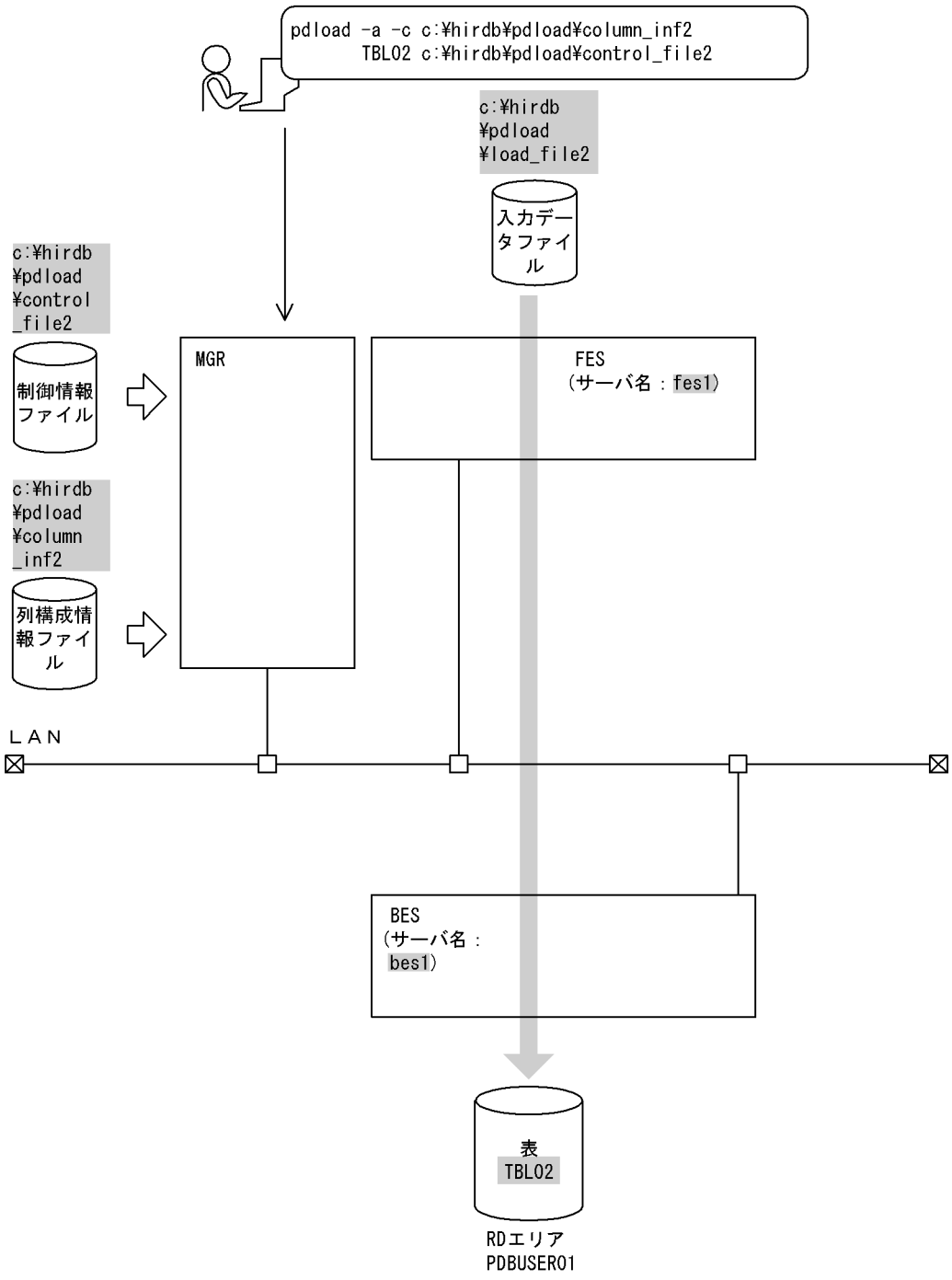
```
抽象データ型（GEOMETRY）がある表（TBL02）にデータロードします。
なお，次の表が定義されているものとします。
・表の定義
  CREATE TABLE TBL02(C1 INTEGER,C2 DECIMAL(7,0),C3 GEOMETRY)
                        IN PDBUSER02
また，GEOMETRY型のコンストラクタ関数の内容を次に示します。
・関数名：GEOMETRY　入力：VARCHAR　出力：GEOMETRY型
```

［概要］



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

[入力データファイルの内容（固定長データ形式）]

```
                                                  [内容]
000000010000001c000affffffffffffffffffff      1，1，10バイトのデータ
ffffffff0000001d0001ff000000000000000000      -1，-1，1バイトのデータ
000003e70000999c0005fffffffff0000000000       999，999，5バイトのデータ

<-int ->＜-dec ->＜-------varchar(10)------>
```

[実行するコマンドの説明]

抽象データ型がある表（TBL02）へデータロードします。

c:¥hirdb¥pdload¥column_inf2：列構成情報ファイルの名称

c:¥hirdb¥pdload¥control_file2：制御情報ファイルの名称

[列構成情報ファイル（c:¥hirdb¥pdload¥column_inf2）の内容]

```
C1,type=integer                                                  1
C2,type=dec(7,0)                                                 2
C3,func=(GEOMETRY,param=varchar,type=varchar(10))                3
```

[説明]

1. 列（C1）に INTEGER のデータを格納
2. 列（C2）に DECIMAL(7,0)のデータを格納
3. 抽象データ型の列（C3）に，入力パラメタが varchar のコンストラクタ関数 GEOMETRY に varchar(10)のデータを渡して生成した値を格納

   GEOMETRY：コンストラクタ関数名

   varchar：コンストラクタ関数に渡す引数の型

[制御情報ファイル（c:¥hirdb¥pdload¥control_file2）の内容]

```
source fes1:c:¥hirdb¥pdload¥load_file2          1
```

[説明]

1. 入力データファイルの指定

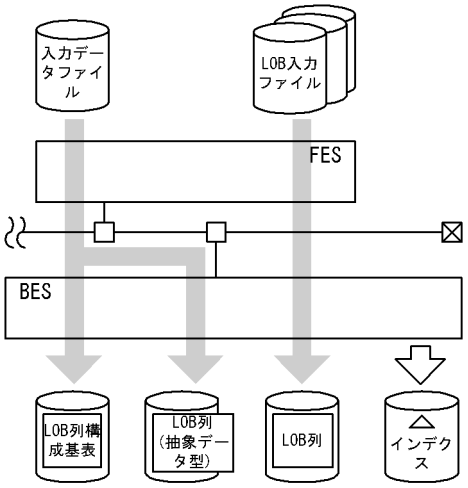
   fes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file2：入力データファイルの名称

例 17

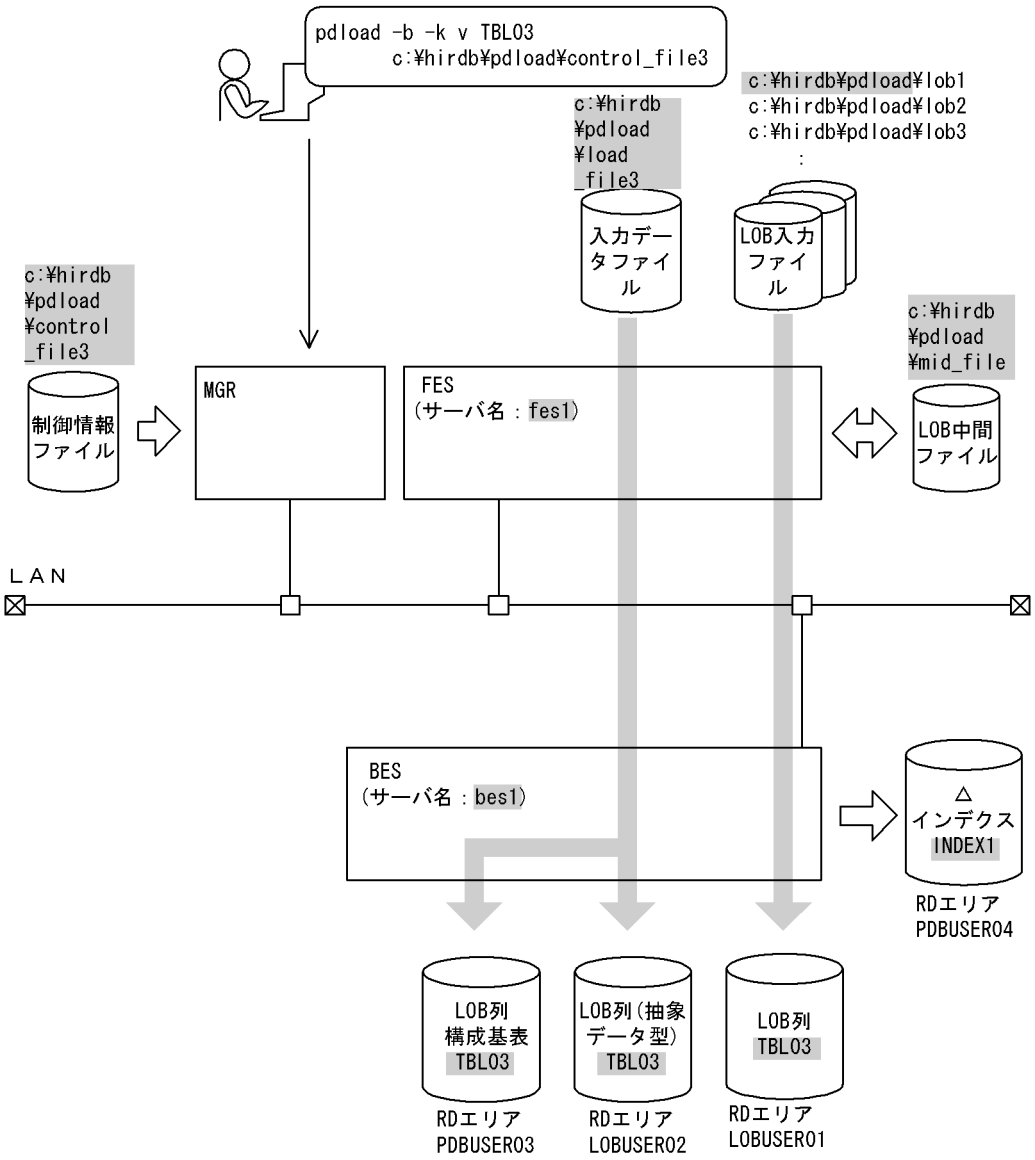
```
抽象データ型（SGMLTEXT）とLOB列がある表（TBL03）にデータロードします。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TBL03(C1 INTEGER,C2 BLOB(4096) IN (LOBUSER01),
                     C3 SGMLTEXT ALLOCATE
                        (SGMLTEXT IN (LOBUSER02) PLUGIN …)
                       IN PDBUSER03
・プラグインインデクスの定義
  CREATE INDEX INDEX1 USING TYPE NGRAM ON TBL03(C3) IN PDBUSER04
また，SGMLTEXT型のコンストラクタ関数の内容を次に示します。
・関数名：SGMLTEXT　入力：BLOB　出力：SGMLTEXT型
```

[概要]



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

[入力データファイルの内容（バイナリ形式）]

	［内容］
00000001000000046c6f62310000000affffffffffffffffffff	1,lob1,10バイトのデータ
ffffffff000000046c6f6232000000001ff	-1,lob2,1バイトのデータ
000003e7000000046c6f623300000005fffffffff	999,lob3,5バイトのデータ

[実行するコマンドの説明]

抽象データ型がある表（TBL03）へデータロードします。

-k v：LOB 列への格納データは LOB 入力ファイル名を入力データファイルに記述し，LOB パラメタは直接入力データファイルに記述する場合の指定

c:¥hirdb¥pdload¥control_file3：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file3）の内容]

```
source fes1:c:¥hirdb¥pdload¥load_file3       1
lobmid c:¥hirdb¥pdload¥mid_file              2
lobdata c:¥hirdb¥pdload                      3
```

[説明]

1. 入力データファイルの指定

   fes1：入力データファイルがあるサーバの名称

   c:¥hirdb¥pdload¥load_file3：入力データファイルの名称

2. LOB 中間ファイルの指定

   c:¥hirdb¥pdload¥mid_file：LOB 中間ファイルの名称

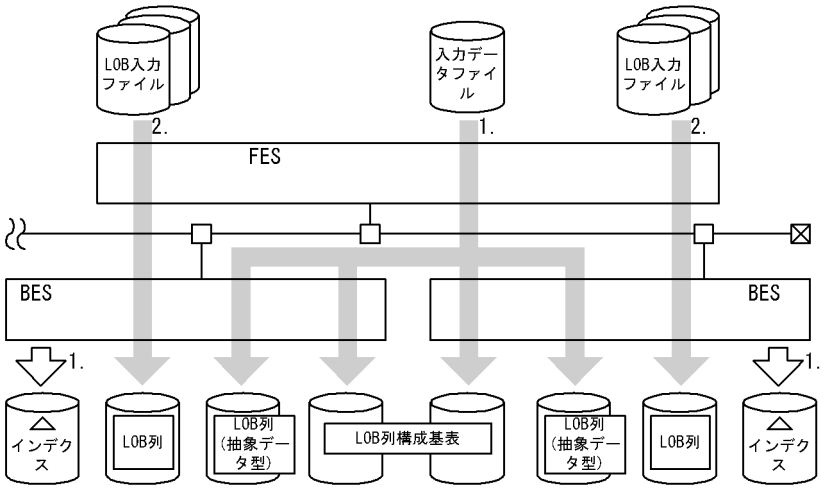
3. LOB 入力ファイルの指定

   c:¥hirdb¥pdload：LOB 入力ファイルがあるディレクトリ名

例 18

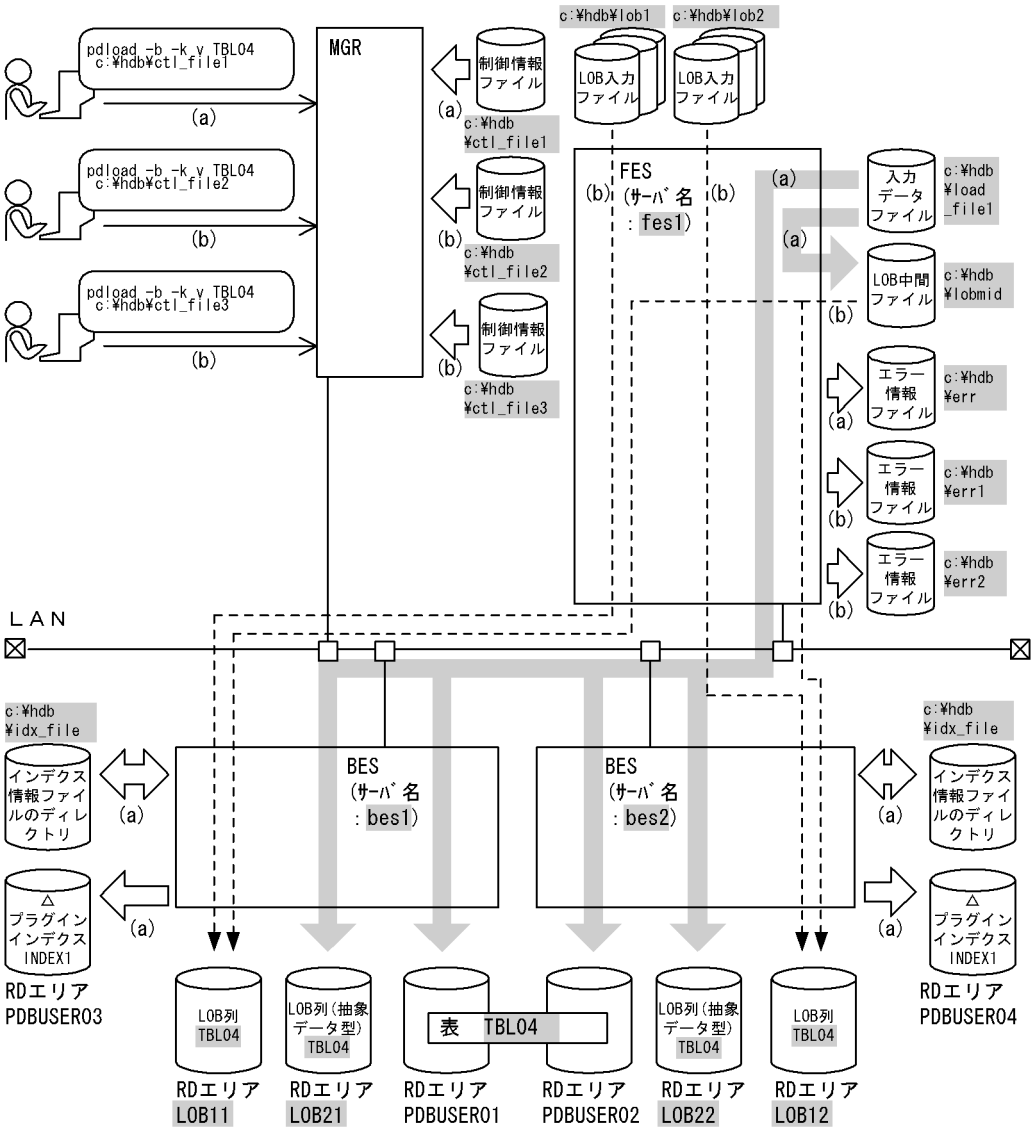
```
抽象データ型（SGMLTEXT）とLOB列がある表（TBL04）にデータロードします。このとき，先にLOB列
構成基表と抽象データ型へデータロードして，その後にLOB列にデータロードをします。LOB列へ
データロードするときは，各バックエンドサーバで並列に実行します。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TBL04(C1 INTEGER,C2 BLOB(100) IN ((LOB11),(LOB12)),
                     C3 SGMLTEXT ALLOCATE
                         (SGMLTEXT IN ((LOB21),(LOB22)) PLUGIN …))
                        IN ((PDBUSER01) C1 > 1000,(PDBUSER02))
・プラグインインデクスの定義
  CREATE INDEX INDEX1 USING TYPE NGRAM ON TBL04(C3)
      IN ((PDBUSER03),(PDBUSER04))
また，SGMLTEXT型のコンストラクタ関数の内容を次に示します。
・関数名：SGMLTEXT　入力：BLOB　出力：SGMLTEXT型
```

[概要]



（凡例）FES：フロントエンドサーバ
BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース作成ユティリティでの指定値
ＭＧＲ：システムマネジャ
ＦＥＳ：フロントエンドサーバ
ＢＥＳ：バックエンドサーバ

注　図中の(a)，(b)は，それぞれ(a)，(b)のコマンドを実行したときのデータの流れを示しています。

(a)　LOB 列構成基表と抽象データ型へのデータロード

[実行するコマンドの説明]

抽象データ型がある表（TBL04）の，LOB 列構成基表と抽象データ型へデータロードします。

-b：

入力データファイルがバイナリ形式の場合の指定

-k v：

LOB 列への格納データは LOB 入力ファイル名を入力データファイルに記述し，LOB パラメタは直接入力データファイルに記述する場合の指定

c:¥hdb¥ctl_file1：

制御情報ファイルの名称

[制御情報ファイル（c:¥hdb¥ctl_file1）の内容]

```
source fes1:c:¥hdb¥load_file1 err=c:¥hdb¥err      1
idxwork bes1 c:¥hdb¥idx_file                      2
idxwork bes2 c:¥hdb¥idx_file                      2
lobmid c:¥hdb¥lobmid                              3
```

[説明]

1. 入力データファイルの指定

   fes1：入力データファイルがあるサーバの名称

   c:¥hdb¥load_file1：入力データファイルの名称

   c:¥hdb¥err：エラー情報ファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hdb¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. LOB 中間ファイルの指定

   c:¥hdb¥lobmid：LOB 中間ファイルの名称

(b)　LOB 列へのデータロード

[実行するコマンドの説明]

抽象データ型がある表（TBL04）の，LOB 列へデータロードします。このとき，各バックエンドサーバで並列にデータロードします。

-b：

入力データファイルがバイナリ形式の場合の指定

-k v：

LOB 列への格納データは LOB データごとに LOB 入力ファイルを用意し，抽象データ型列に格納する値を生成するコンストラクタ関数の入力パラメタとなるデータは入力データファイルに格納する場合の指定

c:¥hdb¥ctl_file2，c:¥hdb¥ctl_file3：

制御情報ファイルの名称

[制御情報ファイル（c:¥hdb¥ctl_file2）の内容]

```
lobdata c:¥hdb¥lob1                                 1
lobmid LOB11 fes1:c:¥hdb¥lobmid error=c:¥hdb¥err1   2
```

[説明]

1. LOB 入力ファイルの指定

   c:¥hdb¥lob1：LOB 入力ファイルがあるディレクトリの名称

2. LOB 中間ファイルの指定

   LOB11：LOB 列があるユーザ LOB 用 RD エリアの名称

   fes1：LOB 中間ファイルがあるサーバの名称

   c:¥hdb¥lobmid：LOB 中間ファイルの名称

   c:¥hdb¥err1：エラー情報ファイルの名称

[制御情報ファイル（c:¥hdb¥ctl_file3）の内容]

```
lobdata c:¥hdb¥lob2                                    1
lobmid LOB12 fes1:c:¥hdb¥lobmid error=c:¥hdb¥err2      2
```

[説明]

1. LOB 入力ファイルの指定

   c:¥hdb¥lob2：LOB 入力ファイルがあるディレクトリの名称

2. LOB 中間ファイルの指定

   LOB12：LOB 列があるユーザ LOB 用 RD エリアの名称

   fes1：LOB 中間ファイルがあるサーバの名称

   c:¥hdb¥lobmid：LOB 中間ファイルの名称

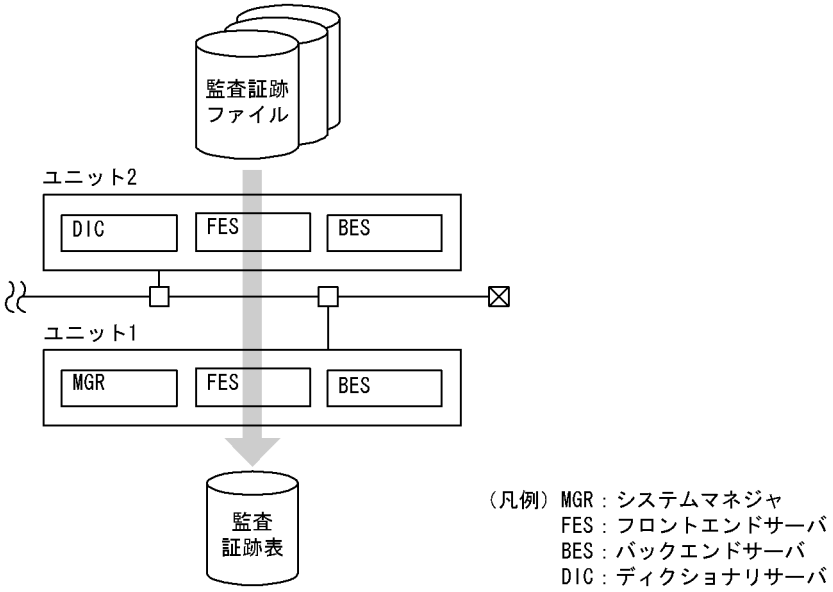
   c:¥hdb¥err2：エラー情報ファイルの名称

(8)　監査証跡表へのデータロード

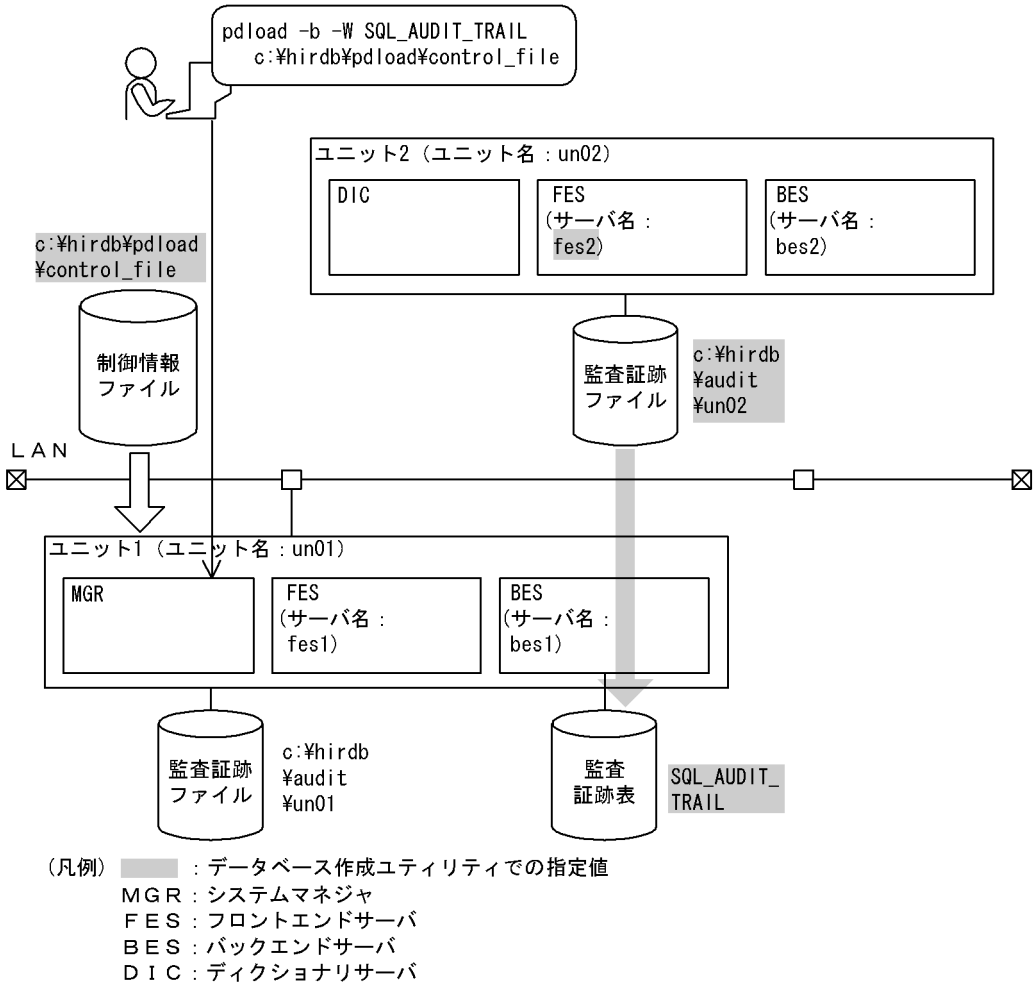
例 19

ユニット1（un01）のバックエンドサーバ（bes1）に定義した監査証跡表（SQL_AUDIT_TRAIL）に対して，ユニット2（un02）にあるデータロード待ち状態の監査証跡ファイルをデータロードします。

[概要]



[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

監査証跡表（SQL_AUDIT_TRAIL）に対して，データロード待ちの監査証跡ファイル（c:¥hirdb¥audit¥un02）をデータロードします。

-b 及び-W：監査証跡表へデータロードする場合に必ず指定するオプションです。

c:¥hirdb¥pdload¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdload¥control_file）の内容]

```
source fes2:(uoc)                                    1
srcuoc param='dir=c:¥hirdb¥audit¥un02,file=all'      2
```

[説明]

1. データロード対象となる監査証跡ファイルがあるユニットの任意のサーバ名称（fes2）と，監査証跡データ登録 UOC の指定
2. 監査証跡データ登録 UOC に渡すパラメタの指定

   dir=c:¥hirdb¥audit¥un02：監査証跡ファイルの HiRDB ファイル名

   file=all：データロード待ちのすべての監査証跡ファイルを対象とするため，all を指定

6 会話型 SQL 実行ユティリティ

この章では，HiRDB のデータベースに対して，会話形式で SQL を実行できる会話型 SQL 実行ユティリティについて説明します。なお，会話型 SQL 実行ユティリティを実行するには，別プログラムプロダクトの HiRDB SQL Executer が必要です。

6.1　概要

実行者

CONNECT 権限を持っているユーザが実行できます。ただし，実行する SQL によって必要な権限が異なります。SQL 実行時の権限については，マニュアル「HiRDB SQL リファレンス」を参照してください。

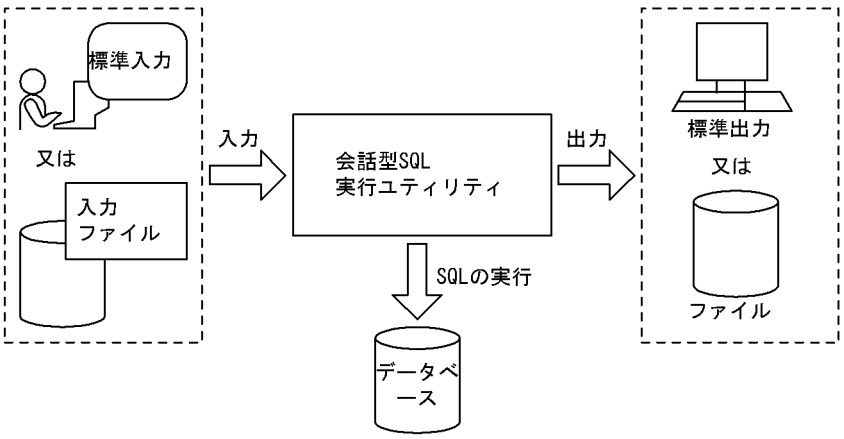
6.1.1　機能

会話型 SQL 実行ユティリティは，HiRDB のデータベースに対して，会話形式で SQL を実行できるユティリティです。また，SQL 文の入力，実行を簡単にしたり，実行結果の形式を指定したりするなどの，会話型 SQL 実行ユティリティ専用のコマンドも指定できます。

このユティリティを実行するためには，別プログラムプロダクトの HiRDB SQL Executer が必要となります。

標準入力，又は入力ファイルから入力した SQL を実行し，その実行結果を標準出力，又はファイルに出力します。会話型 SQL 実行ユティリティの概要を次の図に示します。

図 6–1　会話型 SQL 実行ユティリティの概要



会話型 SQL 実行ユティリティを実行する場合は，［スタート］−［プログラム］−［HiRDB SQL Executer］から選択してください。

会話型 SQL 実行ユティリティの詳細については，HiRDB SQL Executer の「リリースノート」を参照してください。

7 データベース構成変更ユティリティ（pdmod）

この章では，RD エリアの追加，拡張，再初期化など，データベースの物理構成を変更するデータベース構成変更ユティリティ（pdmod）について説明します。

7.1　データベース構成変更ユティリティの機能

実行者　HiRDB 管理者が実行できます

データベース構成変更ユティリティ（pdmod）は，RD エリアの追加，拡張，又は再初期化など，データベースの物理構成を変更するためのユティリティです。

データベースへの表の追加，更新を繰り返すことによって生じる RD エリアの容量不足などを解消します。

pdmod の機能と，RD エリアの種類との関係を次の表に示します。

表 7–1　機能と RD エリアの種類との関係

機能		RD エリアの種類								
		マスタディレクトリ用	データディレクトリ用	データディクショナリ用	データディクショナリ LOB 用	ユーザ用	ユーザ LOB 用	リスト用	レジストリ用	レジストリ LOB 用
RD エリアの追加		×	×	○	○※1	○	○	○	×	×
RD エリアの拡張		○	○	○	○	○	○	○	○	○
RD エリアの再初期化		×	×	×	○※2	○	○	○	○	○
RD エリアの削除		×	×	×※3	×	○	○	○	×	×
RD エリアの移動		×	×	×	×	○※4	○	×	×	×
ディクショナリ表の属性定義変更		–	–	–	–	–	–	–	–	–
RD エリアの属性変更	RD エリア名	×	○	○	○	○	○	○	○	○
	オープン契機	×	×	×	×	○	○	○	×	×
	自動増分	×	×	○	○	○	○	×	○	○
HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更		○	○	○	○	○	○	○	○	○
監査人の登録		–	–	–	–	–	–	–	–	–
監査証跡表の作成		×	×	×	×	○※4	×	×	×	×

（凡例）

○：実行できます。

×：実行できません。

–：該当しません。

注※1

RD エリアを追加する場合は，必ず 2 個追加してください。

注※2

ストアドプロシジャ，又はストアドファンクションのオブジェクトが格納されている RD エリアだけ実行できます。

注※3

解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアの場合，実行できます。

注※4

一時表用 RD エリアの場合，実行できません。

7.2　コマンドの形式

7.2.1　コマンドの形式とオプション

(1)　形式

```
pdmod -a 制御文ファイル名　〔-W 実行監視時間〕
```

(2)　オプション

● **-a 制御文ファイル名**　～＜パス名＞((255 文字以内))

制御文の内容については，以降の節の制御文を参照してください。

【HiRDB/シングルサーバの場合】

pdmod の制御文があるファイルの名称を絶対パス名で指定します。

【HiRDB/パラレルサーバの場合】

pdmod の制御文があるファイルの名称を絶対パス名で指定します。制御文ファイルは，ディクショナリサーバがあるサーバマシンに作成してください。

- 制御文は，複数同時に指定できます。制御文ごとに処理を完結し，エラーになった制御文はスキップして処理を続行します。そのため，正常に終了した制御文は，ほかの制御文でエラーが発生してもロールバックはされません。したがって，pdmod でエラーが発生した場合は，標準エラー出力及びメッセージログに出力されたメッセージのエラー要因を取り除き，エラーがあった制御文だけ再度実行してください。
- 制御文に注釈を付けたい場合は，スラント，アスタリスク（/＊）とアスタリスク，スラント（＊/）で囲んで指定してください。
- pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，制御文ファイル中に BOM が存在しても問題ありません。ただし，制御文ファイル中のコメントには，ASCII コード以外の文字コードを使用できません。ASCII コード以外の文字コードを使用した場合，pdmod が誤動作するおそれがあるため，使用しないでください。
- 制御文に指定する RD エリア名が，制御文で使用しているキーワードと同じ名称である場合は，RD エリア名を引用符（"）で囲んで指定してください。

● **-W 実行監視時間**　～＜符号なし整数＞((0～3600))

データベース構成変更ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

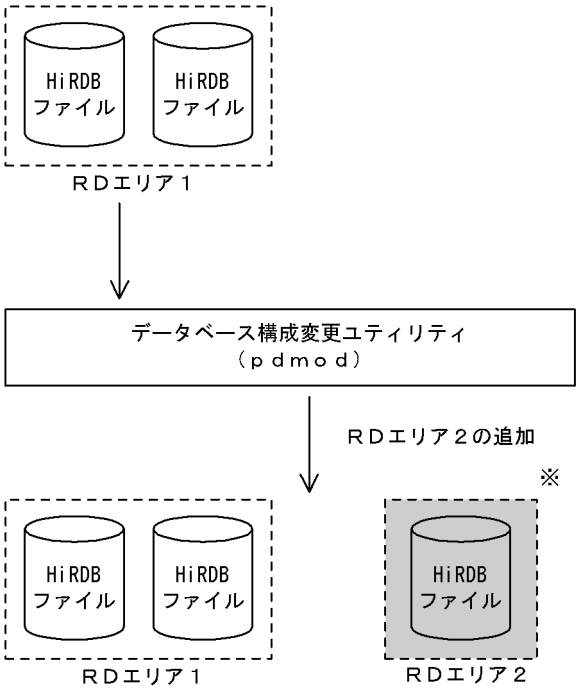
また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

7.3　RD エリアの追加

7.3.1　機能

RD エリアの追加は，表及びインデクスの追加，表定義の変更などで既存の RD エリアだけでは対処できなくなる場合に，新たに RD エリアを追加する機能です。RD エリアの追加の概要を次の図に示します。

図 7–1　RD エリアの追加の概要



（凡例）　：追加したＲＤエリア

注※　追加したＲＤエリアには，次回のＨｉＲＤＢ正常開始時に追加したＲＤエリアをグローバルバッファに割り当てるように，システム共通定義を変更する必要があります。

●前提条件

1. RD エリアを追加する場合は，RD エリアを構成する HiRDB ファイルを作成するための HiRDB ファイルシステム領域を，あらかじめ初期設定しておく（pdfmkfs コマンドを実行しておく）必要があります。
2. RD エリアを追加する場合は，システム共通定義の RD エリアの最大数（pd_max_rdarea_no），及び RD エリアを構成する HiRDB ファイルの最大数（pd_max_file_no）を超えないようにしてください。超える場合は，pdchgconf コマンドでシステム共通定義を変更するか，又は HiRDB システムを正常終了させてから，システム共通定義を変更する必要があります。
3. 共用 RD エリアを追加する場合，その RD エリアはすべてのバックエンドサーバに追加したものとしてカウントされます（共用 RD エリアを追加していないバックエンドサーバも，RD エリアを追加し

たとみなされます)。したがって，どれかのバックエンドサーバが RD エリアの最大数，又は RD エリアを構成する HiRDB ファイルの最大数を超える場合，共用 RD エリアは追加できません。

4. 共用 RD エリアを追加する場合，ユーザ用 RD エリアだけ対象となります。また，使用目的 SDB で作成した HiRDB ファイルシステム領域にだけ追加できます。

5. 0904 互換モードを適用している場合で，一時表用 RD エリアを追加するときは，システム共通定義の一時表用 RD エリアの最大数（pd_max_tmp_table_rdarea_no の値）を超えないようにしてください。一時表用 RD エリアの最大数を超える場合は，HiRDB を終了してからシステム共通定義を変更する必要があります。

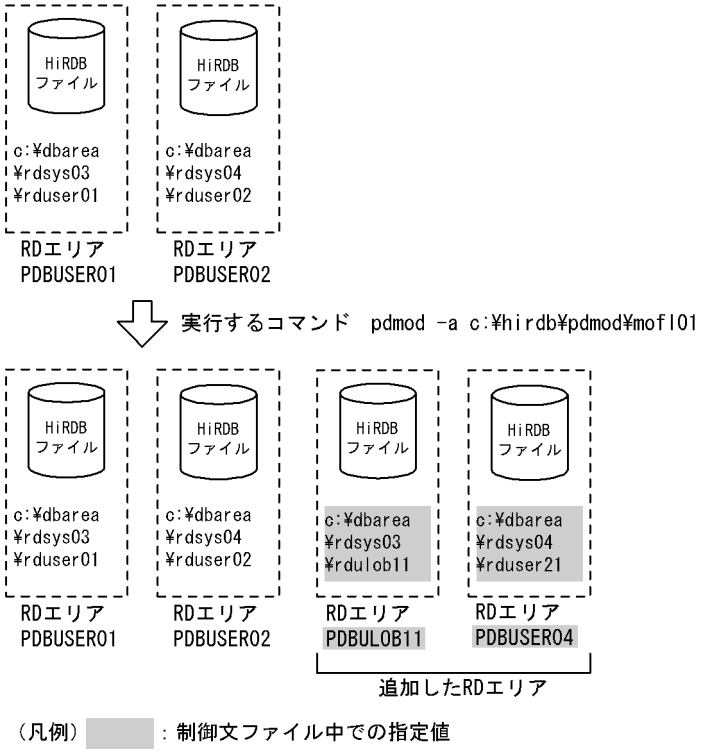
7.3.2 使用例

データベース構成変更ユティリティ（RD エリアの追加）の使用例を次に示します。

(1) RD エリアを追加する例（その 1）

次に示すRDエリアを追加します。
・PDBULOB11（ユーザLOB用RDエリア）
・PDBUSER04（ユーザ用RDエリア）

[概要]



(a) HiRDB/シングルサーバの場合

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
/* ユーザLOB用RDエリアの追加 */
create rdarea PDBULOB11   ..................................................1
         globalbuffer gb01   ...............................................2
         for LOB used by PUBLIC   ..........................................3
         page 8192 characters   ............................................4
         storage control segment 1 pages   .................................5
         file name "c:¥dbarea¥rdsys03¥rdulob11"  ...........................6
           initial 50 segments ;   .........................................7
/* ユーザ用RDエリアの追加 */
create rdarea PDBUSER04   ..................................................1
         globalbuffer gb02   ...............................................2
         for user used by "userid"  ........................................3
         page 4096 characters   ............................................4
         storage control segment 50 pages   ................................5
         file name "c:¥dbarea¥rdsys04¥rduser21"  ...........................6
           initial 100 segments ;   ........................................7
```

[説明]

1. 追加する RD エリアの名称を指定します。
2. RD エリアを割り当てるグローバルバッファの名称を指定します。
3. RD エリアの種類を指定します。
4. ページ長を指定します。
5. セグメントサイズを指定します。
6. RD エリアを構成する HiRDB ファイルの名称を指定します。
7. HiRDB ファイルのセグメント数を指定します。

(b) HiRDB/パラレルサーバの場合

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
/* ユーザLOB用RDエリアの追加 */
create rdarea PDBULOB11   ..................................................1
         globalbuffer gb01   ...............................................2
         for LOB used by PUBLIC   ..........................................3
         server name bes1   ................................................4
         page 8192 characters   ............................................5
         storage control segment 1 pages   .................................6
         file name "c:¥dbarea¥rdsys03¥rdulob11"  ...........................7
           initial 50 segments ;   .........................................8
/* ユーザ用RDエリアの追加 */
create rdarea PDBUSER04   ..................................................1
         globalbuffer gb02   ...............................................2
         for user used by "userid"  ........................................3
         server name bes2   ................................................4
         page 4096 characters   ............................................5
         storage control segment 50 pages   ................................6
```

```
        file name ”c:¥dbarea¥rdsys04¥rduser21”  .........................7
          initial 100 segments ;  ....................................8
```

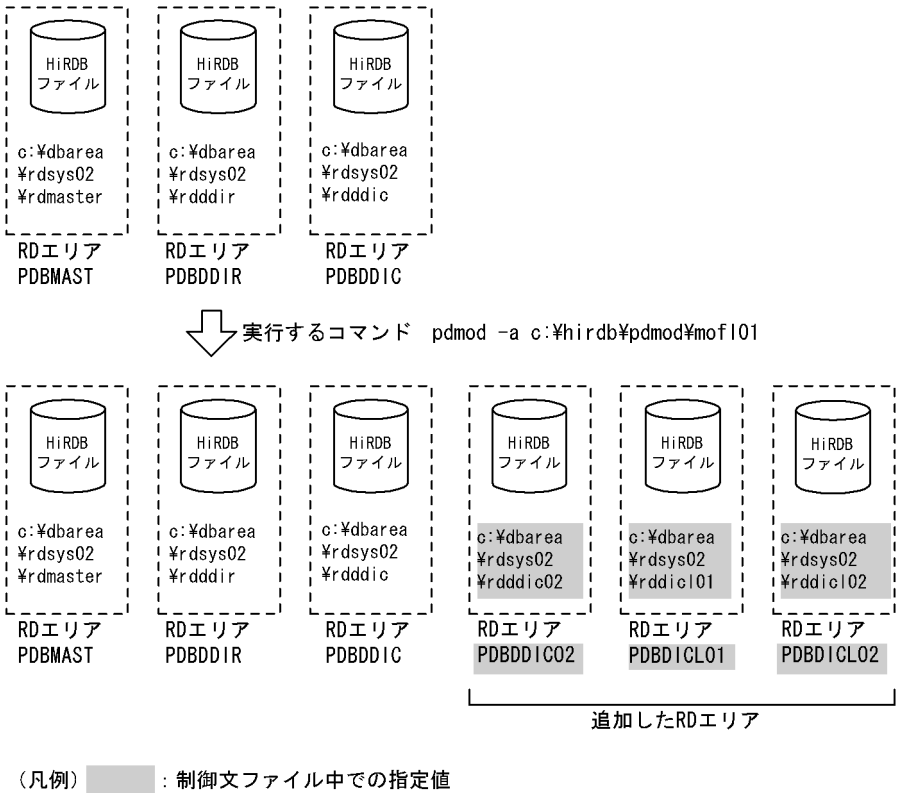
[説明]

1. 追加する RD エリアの名称を指定します。
2. RD エリアを割り当てるグローバルバッファの名称を指定します。
3. RD エリアの種類を指定します。
4. ページ長を指定します。
5. RD エリアを管理するサーバの名称を指定します。
6. セグメントサイズを指定します。
7. RD エリアを構成する HiRDB ファイルの名称を指定します。
8. HiRDB ファイルのセグメント数を指定します。

(2) RD エリアを追加する例（その 2）

次に示すRDエリアを追加します。
・PDBDDIC02（データディクショナリ用RDエリア）
・PDBDICL1（データディクショナリLOB用RDエリア）
・PDBDICL2（データディクショナリLOB用RDエリア）

[概要]



（凡例） ：制御文ファイル中での指定値

(a) HiRDB/シングルサーバの場合

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
/* データディクショナリ用RDエリアの追加 */
create rdarea PDBDDIC02   ..................................................1
         globalbuffer gbp01   ..............................................2
         for datadictionary of routines   ..................................3
         page 4096 characters   ............................................4
         storage control segment 30 pages   ................................5
         file name "c:¥dbarea¥rdsys02¥rdddic02"   ..........................6
           initial 150 segments ;   ........................................7
/* データディクショナリLOB用RDエリアの追加 */
create rdarea PDBDICL01   ..................................................1
         globalbuffer gbp01   ..............................................2
         for LOB used by HiRDB(SQL_ROUTINES)   .............................3
         page 8192 characters   ............................................4
         storage control segment 1 pages   .................................5
         file name "c:¥dbarea¥rdsys02¥rddicl1"   ...........................6
           initial 200 segments ;   ........................................7
/* データディクショナリLOB用RDエリアの追加 */
create rdarea PDBDICL02   ..................................................1
         globalbuffer gbp01   ..............................................2
         for LOB used by HiRDB(SQL_ROUTINES)   .............................3
         page 8192 characters   ............................................4
         storage control segment 1 pages   .................................5
         file name "c:¥dbarea¥rdsys02¥rddicl2"   ...........................6
           initial 2000 segments ;   .......................................7
```

[説明]

1. 追加する RD エリアの名称を指定します。
2. RD エリアを割り当てるグローバルバッファの名称を指定します。
3. RD エリアの種類を指定します。
4. ページ長を指定します。
5. セグメントサイズを指定します。
6. RD エリアを構成する HiRDB ファイルの名称を指定します。
7. HiRDB ファイルのセグメント数を指定します。

(b) HiRDB/パラレルサーバの場合

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
/* データディクショナリ用RDエリアの追加 */
create rdarea PDBDDIC02   ..................................................1
         globalbuffer gbp01   ..............................................2
         for datadictionary of routines   ..................................3
         server name dics   ................................................4
         page 4096 characters   ............................................5
         storage control segment 30 pages   ................................6
         file name "c:¥dbarea¥rdsys02¥rdddic02"   ..........................7
           initial 150 segments ;   ........................................8
```

```
/* データディクショナリLOB用RDエリアの追加 */
create rdarea PDBDICL01   ..................................................1
         globalbuffer gbp01   ..............................................2
         for LOB used by HiRDB(SQL_ROUTINES)   .............................3
         server name dics   ................................................4
         page 8192 characters   ............................................5
         storage control segment 1 pages   .................................6
         file name "c:¥dbarea¥rdsys02¥rddicl1"   ...........................7
           initial 200 segments ;   ........................................8
/* データディクショナリLOB用RDエリアの追加 */
create rdarea PDBDICL02   ..................................................1
         globalbuffer gbp01   ..............................................2
         for LOB used by HiRDB(SQL_ROUTINES)   .............................3
         server name dics   ................................................4
         page 8192 characters   ............................................5
         storage control segment 1 pages   .................................6
         file name "c:¥dbarea¥rdsys02¥rddicl2"   ...........................7
           initial 2000 segments ;   .......................................8
```

[説明]

1. 追加する RD エリアの名称を指定します。
2. RD エリアを割り当てるグローバルバッファの名称を指定します。
3. RD エリアの種類を指定します。
4. RD エリアを管理するサーバの名称を指定します。
5. ページ長を指定します。
6. セグメントサイズを指定します。
7. RD エリアを構成する HiRDB ファイルの名称を指定します。
8. HiRDB ファイルのセグメント数を指定します。

7.3.3 制御文（create rdarea 文）

```
create 〔shared〕
            rdarea RDエリア名
             〔globalbuffer グローバルバッファ名〕
            for {datadictionary of {routines｜dbmanagement} ｜user used by {PUBLIC｜認可
識別子〔, 認可識別子〕…} ｜LOB used by {HiRDB（ディクショナリ表の名称）｜PUBLIC｜認可識別
子〔, 認可識別子〕…} ｜list}
             〔server name サーバ名〕
             〔open attribute {INITIAL｜DEFER｜SCHEDULE} 〕
             〔page ページ長 characters〕
             〔storage control segment セグメントサイズ pages〕
             〔max entries 最大リスト登録数〕
             〔extension {use 増分セグメント数 segments｜nouse} 〕
             〔temporary table {use {shared｜occupied} ｜nouse} 〕
            file name "HiRDBファイルシステム領域名¥HiRDBファイル名"
              initial HiRDBファイル内セグメント数 segments
             〔file name "HiRDBファイルシステム領域名¥HiRDBファイル名"
```

```
                    initial HiRDBファイル内セグメント数 segments〕…
;
```

注

file name オペランド，initial オペランドは，必ず create rdarea 文の最後に指定してください。

(1) shared

このオペランドは，HiRDB/パラレルサーバの場合に指定できます。

RD エリアを共用 RD エリアとして追加する場合に指定します。

＜規則＞

1. shared を指定する場合，システム定義の pd_shared_rdarea_use オペランドに Y を指定する必要があります。指定していない場合はエラーになります。
2. HiRDB/シングルサーバの場合，shared を指定するとエラーになります。
3. 共用 RD エリアとして作成した RD エリアは，server name オペランドで指定したサーバ以外のサーバからも参照できます（すべてのバックエンドサーバから参照できます）。

(2) RD エリア名

～＜識別子＞((1～30))

追加する RD エリアの名称を指定します。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，RD エリア名に空白を含む場合は，引用符で囲んでください。

(3) globalbuffer グローバルバッファ名

～＜識別子＞((1～16))

追加した RD エリアに，一時的にグローバルバッファを割り当てたい場合，HiRDB システムにある（HiRDB/パラレルサーバの場合は，追加するサーバにある）グローバルバッファの名称を指定します。

グローバルバッファについては，pdbufls コマンドで確認できます。

＜規則＞

1. インデクス用，又は LOB 用のグローバルバッファは指定できません。また，インメモリデータバッファも指定できません。
2. データディクショナリ用 RD エリアの場合は，必ずこのオペランドを指定してください。

3. 面数指定のグローバルバッファの場合，指定するグローバルバッファのグローバルバッファ長は，追加する RD エリアのページ長以上でなければなりません。グローバルバッファ長は，pdbufls コマンドで確認できます。
4. グローバルバッファ長は，HiRDB 開始時にそのグローバルバッファに割り当てている RD エリアの最大ページ長となっています。
5. ここで指定したグローバルバッファの割り当ては，次回の HiRDB の正常開始時，又は計画停止後の再開始時には無効となります。そのため，次回の HiRDB の正常開始時，又は計画停止後の再開始時には，システム共通定義のグローバルバッファの割り当てを変更する必要があります。
6. HiRDB がグローバルバッファの割り当てに失敗した場合，RD エリアは追加されません。

(4) for {datadictionary of {routines | dbmanagement} | user used by {PUBLIC | 認可識別子〔，認可識別子〕...} | LOB used by {HiRDB（ディクショナリ表の名称） | PUBLIC | 認可識別子〔，認可識別子〕...} | list}

追加する RD エリアの種類を指定します。

datadictionary of routines

RD エリアがデータディクショナリ用 RD エリアの場合に指定します。

このデータディクショナリ用 RD エリアは，ストアドプロシジャ，ストアドファンクション，及びトリガに関するディクショナリ表（SQL_ROUTINES，SQL_ROUTINE_RESOURCES，及び SQL_ROUTINE_PARAMS）を格納する RD エリアとなります。

データディクショナリ用 RD エリアを追加する場合には，必ずデータディクショナリ LOB 用 RD エリアも同時に指定してください。

datadictionary of dbmanagement

解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアの場合に指定します。

このデータディクショナリ用 RD エリアは，再編成時期予測機能を使用する場合に必要となります。

user used by {PUBLIC | 認可識別子〔，認可識別子〕...}

RD エリアがユーザ用 RD エリアの場合に指定します。

PUBLIC

ユーザ用 RD エリアを公用 RD エリアとする場合に指定します。

認可識別子〔，認可識別子〕... ～＜先頭が英字記号で始まる英数字記号＞((1～30))

ユーザ用 RD エリアを私用 RD エリアにする場合，ユーザ用 RD エリアを利用できるユーザの認可識別子を指定します。

＜規則＞

1. 認可識別子を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として HiRDB システムに登録されます。

2. 認可識別子を指定すると，その私用 RD エリアに対して RD エリア利用権限を持つことができます。

3. 認可識別子は，最大 255 個指定できます。

4. 認可識別子に PUBLIC，MASTER，HiRDB，及び ALL は指定できません。

LOB used by ｛HiRDB（ディクショナリ表の名称）｜ PUBLIC ｜認可識別子〔，認可識別子〕...｝

RD エリアがデータディクショナリ LOB 用 RD エリア，又はユーザ LOB 用 RD エリアの場合に指定します。

HiRDB（ディクショナリ表の名称）

RD エリアがデータディクショナリ LOB 用 RD エリアの場合に指定します。ディクショナリ表の名称には，SQL_ROUTINES を指定してください。

この RD エリアにはストアドプロシジャの定義ソース文とオブジェクトが格納されます。指定する場合は二つ指定する必要があります（この RD エリアを指定した create rdarea 文を 2 個指定します）。一つ目の RD エリアにはストアドプロシジャの定義ソース文が格納され，二つ目の RD エリアにはストアドプロシジャのオブジェクトが格納されます。

＜注意事項＞

1. データディクショナリ LOB 用 RD エリアを追加する場合，ストアドプロシジャに関するディクショナリ表が作成されます。ストアドプロシジャに関するディクショナリ表を，そのほかのディクショナリ表と別のデータディクショナリ用 RD エリアに格納したい場合は，同時にデータディクショナリ用 RD エリアも追加できます。

2. データディクショナリ LOB 用 RD エリアを追加すると，抽象データ型定義，ストアドファンクション，及びストアドプロシジャを使用できるようになります。

3. 既にデータディクショナリ LOB 用 RD エリアがある場合に，このオペランドを指定したときにはエラーとなります。

4. データディクショナリ LOB 用 RD エリアを追加する場合，制御文ファイル中に同時に指定できる RD エリアはデータディクショナリ用 RD エリアだけです。

PUBLIC

ユーザ LOB 用 RD エリアを公用 RD エリアとする場合に指定します。

認可識別子〔，認可識別子〕... ～＜先頭が英字記号で始まる英数字記号＞((1～30))

ユーザ LOB 用 RD エリアを私用 RD エリアにする場合，ユーザ LOB 用 RD エリアを利用できるユーザの認可識別子を指定します。

＜規則＞

1. 認可識別子を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として HiRDB システムに登録されます。

2. 認可識別子を指定すると，その私用 RD エリアに対して RD エリア利用権限を持つことができます。

3. 認可識別子は，最大 255 個指定できます。

4. 認可識別子に PUBLIC，MASTER，HiRDB，及び ALL は指定できません。

list

RD エリアがリスト用 RD エリアの場合に指定します。

(5) server name サーバ名

～＜識別子＞((1～8))

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は server name オペランドを指定しないでください。

【HiRDB/パラレルサーバの場合】

RD エリアを管理するサーバの名称を指定します。

データディクショナリ用 RD エリア及びデータディクショナリ LOB 用 RD エリアの場合，ディクショナリサーバのサーバ名を指定してください。ディクショナリサーバ以外のサーバ名を指定した場合でも，ディクショナリサーバのサーバ名が仮定されます。

ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアの場合，バックエンドサーバのサーバ名を指定してください。バックエンドサーバ以外のサーバ名を指定した場合は，エラーとなります。

共用 RD エリアを追加する場合，このオペランドで指定したサーバが更新可能バックエンドサーバとなります。

(6) open attribute {INITIAL ｜ DEFER ｜ SCHEDULE}

RD エリアのオープン契機を指定します。

＜適用基準＞

特殊な運用形態でない場合は，通常 INITIAL を指定してください。

RD エリア数が多いために HiRDB の開始に時間が掛かるなどの不都合を回避したい場合は，DEFER を指定してください。

＜規則＞

1. このオペランドは，システム共通定義の pd_rdarea_open_attribute_use オペランドに Y を指定しているときだけ有効となります。
2. このオペランドはユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアの場合に指定できます。ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアに対してこのオペランドを指定しても，無視されます（ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアは，INITIAL 固定です）。
3. このオペランドを省略した場合，システム共通定義の pd_rdarea_open_attribute オペランドの指定値が仮定されます。
4. このオペランドの指定値は，HiRDB の開始モードに関係なく次回起動後から有効となります。

5. 該当する RD エリアに割り当てた HiRDB ファイルシステム領域が，ほかの RD エリアに共有される場合，共有する RD エリアのオープン契機はすべて同じにする必要があります。同じでない場合，HiRDB ファイルシステム領域を共有する RD エリアは，期待通りの効果が得られないことがあります。

INITIAL：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンして，RD エリア情報をメモリ上に常駐させるので，最初の SQL から高速な運用ができます。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はオープンで，それ以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. INITIAL の場合，クローズ状態の RD エリアはアクセスできません。

DEFER：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，RD エリアを最初にアクセスするときに HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させるので，2 回目以降のアクセスから高速な運用ができます。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，各 RD エリアに対して最初にアクセスするときに，該当する RD エリアをオープンします。以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. DEFER の場合，クローズ状態の RD エリアでもアクセスできます。

SCHEDULE：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，HiRDB 開始後，各トランザクション内で RD エリアに対して最初にアクセスするときに，HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させます。ただし，トランザクション終了時にそのトランザクション内でオープンした HiRDB ファイルシステム領域をクローズするため，2 回目以降のアクセス時でもオープン以降の処理をして，トランザクションに掛かる負荷は増加します。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，アクセスした RD エリアをトランザクション内でだけオープンし，トランザクション終了時にトランザクション内でオープンしたすべての RD エリアをクローズします。

   また，pdopen コマンドを実行することで，閉塞クローズ状態になるまでの間はオープン状態にできます。そのほかの運用コマンドを使用すれば，RD エリアの状態を任意に遷移させることもできます。なお，障害事象を検知した場合には，障害閉塞となります。
2. SCHEDULE の場合，クローズ状態の RD エリアでもアクセスできます。

(7) page ページ長 characters

～＜符号なし整数＞((4096～30720))《4096 又は 8192》

RD エリアを構成する HiRDB ファイルのページ長を，2,048 の倍数のバイト単位で指定します。HiRDB ファイルシステム領域作成時にセクタ長を指定した場合，2,048 の倍数かつセクタ長の倍数を指定してください。

次の RD エリアは，ページ長が固定となります。

- データディクショナリ LOB 用 RD エリア，及びユーザ LOB 用 RD エリアの場合
  8,192 を指定してください。8,192 以外の値を指定した場合，及び省略した場合でも，8,192 が仮定されます。
- 解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアの場合
  4,096 を指定してください。4,096 以外の値を指定した場合，及び省略した場合でも，4,096 が仮定されます。

そのほかの RD エリアのページ長については，マニュアル「HiRDB システム導入・設計ガイド」を参照して適切な値を指定してください。

ここで指定したページ長は，HiRDB が RD エリアに対して入出力する最小の単位となります。

(8) storage control segment セグメントサイズ pages

～<符号なし整数>((1～16000))《1 又は 50》

1 セグメントの大きさを，ページ数で指定します。

データディクショナリ LOB 用 RD エリア，及びユーザ LOB 用 RD エリアの場合は，1 を指定してください。1 以外の値を指定した場合でも，1 が仮定されます。そのほかの RD エリアのセグメントサイズについては，マニュアル「HiRDB システム導入・設計ガイド」を参照して適切な値を指定してください。

(9) max entries 最大リスト登録数

～<符号なし整数>((500～50000，かつ 500 の倍数))≪500≫

リスト用 RD エリアに作成できるリストの最大数を，500 の倍数で指定します。

500 の倍数でない値を指定した場合，500 の倍数に切り上げた値が仮定されます。

このオペランドをリスト用 RD エリア以外の RD エリアに指定した場合，エラーとなります。

(10) extension {use 増分セグメント数 segments | nouse}

追加する RD エリアに自動増分を適用するかどうかを指定します。

RD エリアの自動増分については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜前提条件＞

RD エリアがある HiRDB ファイルシステム領域に-e オプション又は-a オプションが指定されている必要があります。

use 増分セグメント数 segments：

RD エリアの自動増分を適用します。

RD エリアの自動増分は，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，ユーザ用 RD エリア，及びユーザ LOB 用 RD エリアの場合に有効となります。ほかの RD エリアに use を指定しても，nouse が仮定されます。

自動増分の契機は，RD エリア内の空きセグメント数が自動増分契機に達したときです。
pd_rdarea_extension_timing オペランドで指定できます。

増分セグメント数　～＜符号なし整数＞((1～64000))

増分セグメント数を指定します。

- pdfmkfs -e を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルのエクステント数は最大 24 で，これを超えるとエラーとなります。また，HiRDB ファイルシステム領域の空きが不足したり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

  HiRDB ファイルシステム領域当たりの最大増分回数は，HiRDB ファイルシステム領域作成時の指定値で決まるため，HiRDB ファイルシステム領域内のファイル数，増分頻度を考慮し最大増分回数を定義しておく必要があります。

- pdfmkfs -a を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルを格納しているディスクに，必要な HiRDB ファイルシステム領域を確保する容量が不足していたり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

- 指定した増分セグメント数を適用した場合に，RD エリアを構成する HiRDB ファイルの最終ファイルのサイズが 64 ギガバイトを超える増分セグメント数は指定できません（KFPX24241-E エラーとなります）。なお，このチェックは，最終 file オペランドの initial に指定したセグメント数に，指定した増分セグメント数を 1 回分加算して算出したファイルサイズで判定します。

nouse：

RD エリアの自動増分機能を適用しません。

＜注意事項＞

1. 新たに追加される未使用セグメントは，RD エリアを構成する HiRDB ファイルの最終ファイルに追加されます。
2. HiRDB ファイルシステム領域の容量不足によって，未使用セグメントを確保できない場合，RD エリアの拡張又は再初期化をするか，データベース再編成ユティリティで表の再編成をしてください。
3. エクステントの数が上限値を超えた場合は，RD エリアがある HiRDB ファイルシステム領域のエクステントを統合するか，又は RD エリアに別の HiRDB ファイルシステム領域の HiRDB ファイルを追加してください。

   エクステントを統合するには，pdfbkup でバックアップを取得し，pdfmkfs で HiRDB ファイルシステム領域の初期設定をし，pdfrstr でバックアップから回復します。
4. RD エリア自動増分処理の開始から終了までの間，最終ファイルに対して排他が掛かります。

(11) temporary table {use {shared | occupied} | nouse}

ユーザ用 RD エリアを一時表用 RD エリアとして使用するかどうかを指定します。一時表用 RD エリアについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

＜前提条件＞

追加する RD エリアが公用 RD エリア（user used by PUBLIC 指定）である必要があります。それ以外の場合に temporary table オペランドを指定すると，エラーになります。また，shared オペランドと同時に指定したときも，エラーになります。なお，この前提条件は，nouse を指定したときには該当しません。

use shared：

ユーザ用 RD エリアを，SQL セッション間共有属性の一時表用 RD エリアとして使用します。

use occupied：

ユーザ用 RD エリアを，特定 SQL セッション占有属性の一時表用 RD エリアとして使用します。

nouse：

ユーザ用 RD エリアを一時表用 RD エリアとして使用しません。

＜注意事項＞

RD エリア作成後に一時表用 RD エリアの使用有無，及び一時表用 RD エリアの属性（SQL セッション間共有属性又は特定 SQL セッション占有属性）を変更する場合，データベース構成変更ユティリティの remove rdarea 文で該当する RD エリアを削除し，create rdarea 文で属性を変更して再作成してください。

(12) file name "HiRDB ファイルシステム領域名¥HiRDB ファイル名"

～((167 文字以内))

RD エリアに対して割り当てる，HiRDB ファイルシステム領域の名称と HiRDB ファイルの名称を指定します。

＜規則＞

1. HiRDB ファイルシステム領域名¥HiRDB ファイル名は引用符（"）で囲んで指定してください。
2. 引用符内に改行文字を含まないようにしてください。
3. 一つの RD エリアに対して，最大 16 個の HiRDB ファイルを割り当てられます。
4. 指定した HiRDB ファイルシステム領域に，pdmod が HiRDB ファイルを作成します。
5. HiRDB ファイルシステム領域名¥HiRDB ファイル名は，HiRDB システム内で重複しないようにしてください。
6. リスト用 RD エリアの場合は，使用目的が WORK の HiRDB ファイルシステム領域を使用してください。

7. HiRDB ファイルシステム領域名までは大文字，小文字を区別しませんが，HiRDB ファイル名は大文字，小文字を区別します。

HiRDB ファイルシステム領域名　～＜パス名＞

HiRDB ファイルシステム領域の名称を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞((1～30 文字))

HiRDB ファイルの名称を指定します。pl で始まる名称は指定できません。

(13)　initial HiRDB ファイル内セグメント数 segments

～＜符号なし整数＞

HiRDB ファイルのセグメント数を指定します。

セグメント数は，データページ部分の総ページ数÷セグメントサイズで求めてください。データページ部分の総ページ数の見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。また，HiRDB ファイルの大きさが 64 ギガバイトを超えるようなセグメント数は指定できません。

7.3.4　注意事項

1. 追加した RD エリアは，次の契機で使用できるようになります。
   - 制御文でグローバルバッファの割り当てを指定していない場合

     次回 HiRDB の正常開始後に使用できます。ただし，-o オプション指定のグローバルバッファがある場合で，そのグローバルバッファのバッファ長が追加した RD エリアのページサイズよりも大きいときは，再開始時にも使用できます。なお，再開始後は RD エリアがクローズ状態となっているため，pdopen コマンドで RD エリアをオープン状態にしてから使用してください。
   - 制御文でグローバルバッファの割り当てを指定している場合

     RD エリア追加完了後に使用できます。共用 RD エリアを追加する場合，割り当てるグローバルバッファがすべてのバックエンドサーバになければなりません。

   どちらの場合も，次回 HiRDB の正常開始のときに，追加した RD エリアをグローバルバッファに割り当てるように，システム共通定義を変更する必要があります。システム共通定義については，マニュアル「HiRDB システム定義」を参照してください。システム共通定義を変更しない場合，システム共通定義の pdbuffer オペランドに-o オプションを指定したグローバルバッファがあるときは，次回 HiRDB の正常開始時に自動的に-o オプション指定のグローバルバッファに，追加した RD エリアが割り当てられます。また，-o オプション指定のグローバルバッファがある場合，又は制御文でグローバルバッファを指定している場合は，HiRDB の再開始時にグローバルバッファが割り当てられます。
2. RD エリアの追加後は，データベース複写ユティリティ（pdcopy）で追加した RD エリア，マスタディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアのバックアップを取得しておいてください。データベース複写ユティリティについては，「データベース複写ユティリティ（pdcopy）」を参照してください。

3. データディクショナリ LOB 用 RD エリアを追加すると，ストアドプロシジャ又はストアドファンクションを使用できるようになります。データディクショナリ LOB 用 RD エリアを追加した場合，同時にディクショナリ表が作成されます。そのため，データディクショナリ用 RD エリアに空きがあるかどうか確認し，空きがない場合はあらかじめデータディクショナリ用 RD エリアを拡張しておく必要があります。RD エリアの容量を確認する場合は，データベース状態解析ユティリティを実行してください。

   また，ストアドプロシジャ又はストアドファンクションに関するディクショナリ表をほかのディクショナリ表と別のデータディクショナリ用 RD エリアに格納したい場合は，同時にデータディクショナリ用 RD エリアも追加できます。

   データディクショナリ LOB 用 RD エリアを追加した後，ストアドプロシジャ又はストアドファンクションを使用できるようになる契機は，追加した RD エリアにグローバルバッファが割り当てられたときです。

7.4　RD エリアの拡張

7.4.1　機能

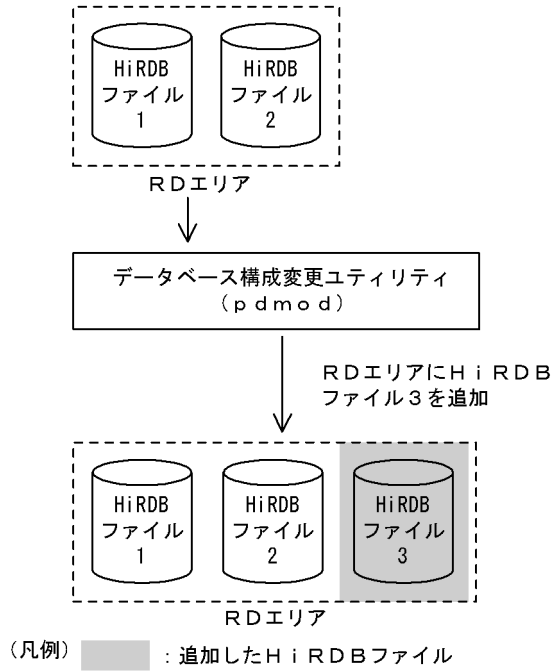
RD エリアの拡張は，データの追加などによって，当初見積もった RD エリアの容量が足りなくなった場合に，RD エリアを構成する HiRDB ファイルを追加する機能です。RD エリアを構成する HiRDB ファイルを追加することで，RD エリアを拡張します。

追加できる HiRDB ファイル数は，RD エリアを構成する HiRDB ファイルの個数（16 個）までです。

ほかのユーザが，RD エリア内の表又はインデクスを参照中であっても，RD エリアは拡張できます。RD エリアを拡張した場合でも，RD エリア内のデータには影響はありません。

RD エリアの拡張の概要を次の図に示します。

図 7–2　RD エリアの拡張の概要



注

RD エリアを拡張した場合，必ずその RD エリアと，同時にバックアップを取得する必要がある RD エリアのバックアップを取得してください。同時にバックアップを取得する必要がある RD エリアについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

●前提条件

1. RD エリアを拡張する場合は，HiRDB ファイルを作成するための HiRDB ファイルシステム領域を，あらかじめ初期設定しておく必要があります。
2. RD エリアを拡張する場合，RD エリアは次の状態でなければなりません。

- **マスタディレクトリ用 RD エリア**
  閉塞解除の状態かつオープン状態
- **データディレクトリ用 RD エリア，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリア**

＜ RD エリアのオープン契機が INITIAL の場合＞

閉塞解除の状態，又はコマンドによる閉塞中でかつオープン状態

＜ RD エリアのオープン契機が DEFER 又は SCHEDULE の場合＞

閉塞解除の状態，コマンドによる閉塞中でかつオープン状態，又は閉塞解除かつクローズ状態

RD エリアのオープン契機は，システム共通定義の pd_rdarea_open_attribute オペランド及び pd_rdarea_open_attribute_use オペランドで指定します。これらの詳細については，マニュアル「HiRDB システム定義」を参照してください。また，RD エリアのオープン契機は，pddbls コマンドで確認できます。

3. RD エリアを拡張する場合は，システム共通定義の最大 RD エリア構成ファイル数（pd_max_file_no）を超えないようにしてください。超える場合は，pdchgconf コマンドでシステム共通定義を変更するか，又は HiRDB システムを正常終了させてから，システム共通定義を変更する必要があります。
4. 共用 RD エリアを拡張する場合，その RD エリアはすべてのバックエンドサーバで拡張したものとしてカウントされます（共用 RD エリアを拡張していないバックエンドサーバも，RD エリアを拡張したとみなされます）。したがって，どれかのバックエンドサーバが RD エリアを構成する HiRDB ファイルの最大数を超える場合，共用 RD エリアは拡張できません。

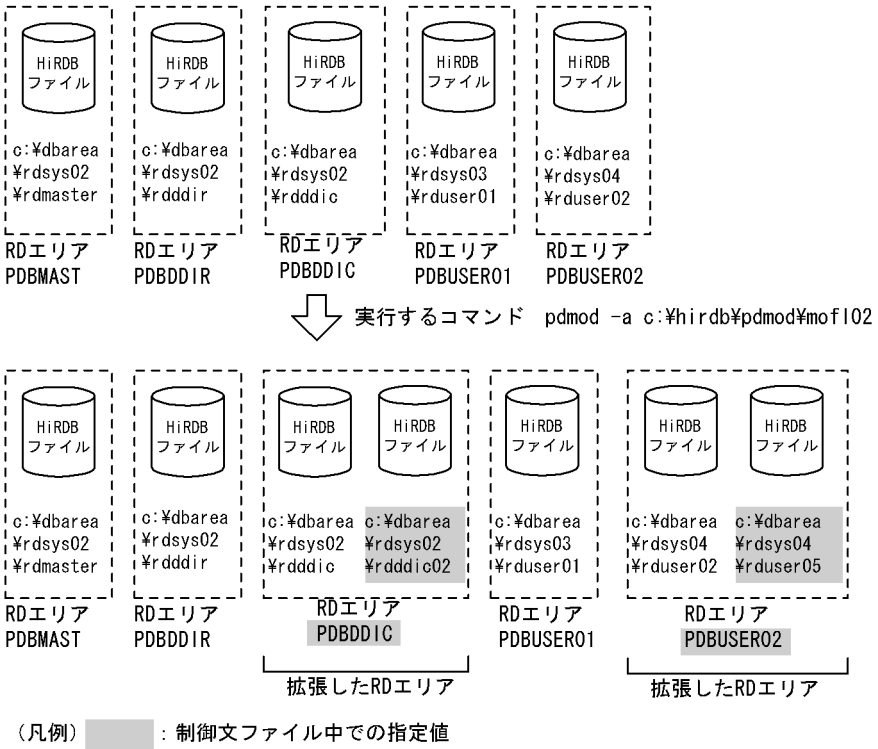
7.4.2　使用例

データベース構成変更ユティリティ（RD エリアの拡張）の使用例を次に示します。

(1)　RD エリアを拡張する例

```
次に示すRDエリアを拡張します。
・PDBDDIC
  新たにHiRDBファイル（c:¥dbarea¥rdsys02¥rdddic02）を割り当てます。
・PDBUSER02
  新たにHiRDBファイル（c:¥dbarea¥rdsys04¥rduser05）を割り当てます。
```

[概要]



[制御文ファイル（c:¥hirdb¥pdmod¥mofl02）の内容]

```
/* データディクショナリ用RDエリアの拡張 */
expand rdarea PDBDDIC   ..................................................1
         file name "c:¥dbarea¥rdsys02¥rdddic02"   ..........................2
           initial 50 segments ;   ..........................................3
/* ユーザ用RDエリアの拡張 */
expand rdarea PDBUSER02   ................................................1
         file name "c:¥dbarea¥rdsys04¥rduser05"   ..........................2
           initial 100 segments ;   .........................................3
```

[説明]

1. 拡張する RD エリアの名称を指定します。
2. RD エリアに追加する HiRDB ファイルの名称を指定します。
3. HiRDB ファイルのセグメント数を指定します。

7.4.3 制御文（expand rdarea 文）

```
expand rdarea RDエリア名
             〔extension  {use 増分セグメント数 segments | nouse} 〕
              file name "HiRDBファイルシステム領域名¥HiRDBファイル名"
                  initial HiRDBファイル内セグメント数 segments
               〔file name "HiRDBファイルシステム領域名¥HiRDBファイル名"
                    initial HiRDBファイル内セグメント数 segments〕…
  ;
```

注

file name オペランド，initial オペランドは，必ず expand rdarea 文の最後に指定してください。

(1) RD エリア名

～＜識別子＞((1～30))

拡張する RD エリアの名称を指定します。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。RD エリア名に空白が含まれる場合は，引用符で囲んでください。

(2) extension ｛use 増分セグメント数 segments ｜ nouse｝

拡張する RD エリアに自動増分を適用するかどうかを指定します。このオペランドを省略した場合，拡張する前の指定を引き継ぎます。

RD エリアの自動増分については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜前提条件＞

RD エリアがある HiRDB ファイルシステム領域に-e オプション又は-a オプションが指定されている必要があります。

use 増分セグメント数 segments：

RD エリアの自動増分を適用します。

RD エリアの自動増分は，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアの場合に有効となります。ほかの RD エリアに use を指定しても，nouse が仮定されます。

自動増分の契機は，RD エリア内の空きセグメント数が自動増分契機に達したときです。
pd_rdarea_extension_timing オペランドで指定できます。

増分セグメント数　～＜符号なし整数＞((1～64000))

増分セグメント数を指定します。

- pdfmkfs -e を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルのエクステント数は最大 24 で，これを超えるとエラーとなります。また，HiRDB ファイルシステム領域の空きが不足したり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

  HiRDB ファイルシステム領域当たりの最大増分回数は，HiRDB ファイルシステム領域作成時の指定値で決まるため，HiRDB ファイルシステム領域内のファイル数，増分頻度を考慮し最大増分回数を定義しておく必要があります。

- pdfmkfs -a を指定した HiRDB ファイルシステム領域の場合

HiRDB ファイルを格納しているディスクに，必要な HiRDB ファイルシステム領域を確保する容量が不足していたり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

- 指定した増分セグメント数を適用した場合に，RD エリアを構成する HiRDB ファイルの最終ファイルのサイズが 64 ギガバイトを超える増分セグメント数は指定できません（KFPX24241-E エラーとなります）。なお，このチェックは，最終 file オペランドの initial に指定したセグメント数に，指定した増分セグメント数を 1 回分加算して算出したファイルサイズで判定します。

nouse：

RD エリアの自動増分機能を適用しません。

＜注意事項＞

1. 新たに追加される未使用セグメントは，RD エリアを構成する HiRDB ファイルの最終ファイルに追加されます。
2. HiRDB ファイルシステム領域の容量不足によって，未使用セグメントを確保できない場合，RD エリアの拡張又は再初期化をするか，データベース再編成ユティリティで表の再編成をしてください。
3. エクステントの数が上限値を超えた場合は，RD エリアがある HiRDB ファイルシステム領域のエクステントを統合するか，又は RD エリアに別の HiRDB ファイルシステム領域の HiRDB ファイルを追加してください。

   エクステントを統合するには，pdfbkup でバックアップを取得し，pdfmkfs で HiRDB ファイルシステム領域の初期設定をし，pdfrstr でバックアップから回復します。
4. RD エリア自動増分処理の開始から終了までの間，最終ファイルに対して排他が掛かります。

(3) file name "HiRDB ファイルシステム領域名¥HiRDB ファイル名"

〜((167 文字以内))

RD エリアに対して追加する，HiRDB ファイルシステム領域の名称と HiRDB ファイルの名称を指定します。

＜規則＞

1. HiRDB ファイルシステム領域名¥HiRDB ファイル名は引用符（"）で囲んで指定してください。
2. 引用符内に改行文字を含まないようにしてください。
3. 一つの RD エリアに対して，最大 16 個の HiRDB ファイルを割り当てられます。
4. 指定した HiRDB ファイルシステム領域に，pdmod が HiRDB ファイルを作成します。
5. HiRDB ファイルシステム領域名¥HiRDB ファイル名は，HiRDB システム内で重複しないようにしてください。
6. リスト用 RD エリアの場合は，使用目的が WORK の HiRDB ファイルシステム領域を使用してください。
7. セクタ長を指定した HiRDB ファイルシステム領域に拡張する場合，RD エリアのページ長は，セクタ長の整数倍でなければなりません。

8. HiRDB ファイルシステム領域名までは大文字，小文字を区別しませんが，HiRDB ファイル名は大文字，小文字を区別します。

HiRDB ファイルシステム領域名　～＜パス名＞

HiRDB ファイルシステム領域の名称を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞((1～30 文字))

HiRDB ファイルの名称を指定します。pl で始まる名称は指定できません。

(4)　initial HiRDB ファイル内セグメント数 segments

～＜符号なし整数＞

HiRDB ファイルのセグメント数を指定します。

セグメント数は，データページ部分の総ページ数÷セグメントサイズで求めてください。データページ部分の総ページ数の見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。また，HiRDB ファイルの大きさが 64 ギガバイトを超えるようなセグメント数は指定できません。

7.4.4　注意事項

1. RD エリアの拡張後は，データベース複写ユティリティ（pdcopy）で拡張した RD エリア，マスタディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアのバックアップを取得しておいてください。データベース複写ユティリティいついては，「データベース複写ユティリティ（pdcopy）」を参照してください。
2. インメモリ RD エリアは拡張できません。いったんインメモリ化を解除してから RD エリアを拡張してください。

7.5　RD エリアの再初期化

7.5.1　機能

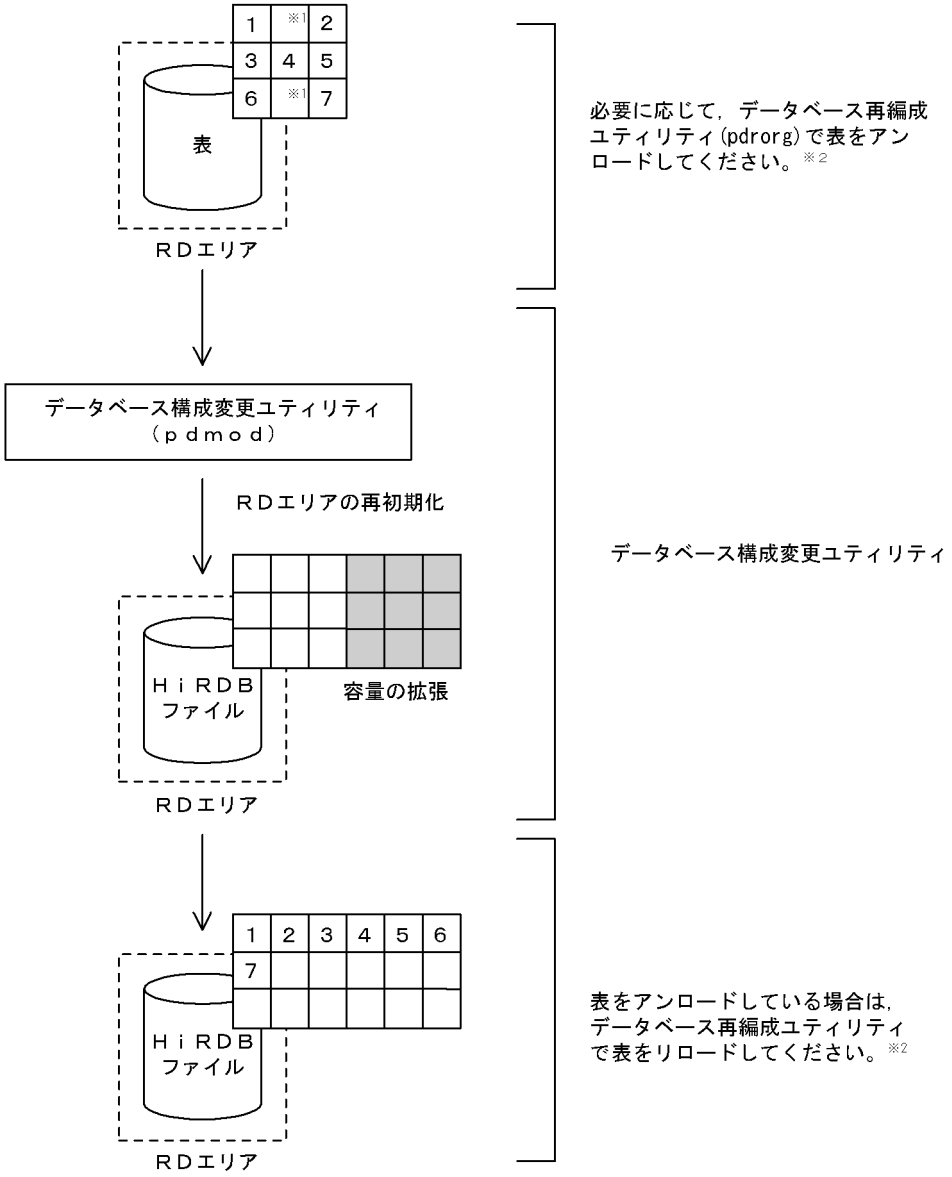
次に示す要因で，再度 RD エリアを初期化したい場合に使用する機能です。

- RD エリアを構成する HiRDB ファイルの容量，個数，及び名称を変更したい場合
- RD エリアのセグメントサイズ及びページ長を変更したい場合（ただし，ユーザ用 RD エリア，及びリスト用 RD エリアの場合）
- RD エリアの障害発生時，バックアップを使用しないで回復したい場合

ただし，RD エリアの再初期化後は，表，及びインデクスの定義だけが引き継がれデータは削除されるため，pdload，又は pdrorg で再作成する必要があります。

RD エリアの再初期化の概要を次の図に示します。

図 7–3　RD エリアの再初期化の概要



注　RD エリアの再初期化後は，必ずその RD エリアのバックアップを取得してください。

注※1　空きページを示します。

注※2　アンロード及びリロードについては，「データベース再編成ユティリティ（pdrorg）」を参照してください。

●前提条件

1. RD エリアを再初期化する場合，RD エリアが閉塞状態かつクローズ状態でなければなりません。共用 RD エリアの場合は，すべてのバックエンドサーバで閉塞状態かつクローズ状態でなければなりません。
2. RD エリアを再初期化する場合は，システム共通定義の最大 RD エリア構成ファイル数（pd_max_file_no）を超えないようにしてください。超える場合は，pdchgconf コマンドでシステ

ム共通定義を変更するか，又は HiRDB システムを正常終了させてから，システム共通定義を変更する必要があります。

3. 共用 RD エリアを再初期化する場合，その RD エリアはすべてのバックエンドサーバで再初期化したものとしてカウントされます（共用 RD エリアを再初期化していないバックエンドサーバも，RD エリアを再初期化したとみなされます）。したがって，どれかのバックエンドサーバが RD エリアを構成する HiRDB ファイルの最大数を超える場合，共用 RD エリアは再初期化できません。
4. データディクショナリ LOB 用 RD エリアを再初期化する場合，ストアドプロシジャ，又はストアドファンクションのオブジェクトを格納している RD エリアだけ実行できます。実行後は，ALTER PROCEDURE（ALL 指定），又は ALTER ROUTINE ですべての SQL オブジェクトを再作成する必要があります。ALTER PROCEDURE，又は ALTER ROUTINE については，マニュアル「HiRDB SQL リファレンス」を参照してください。ストアドプロシジャ，又はストアドファンクションのオブジェクトを格納している RD エリアは，ディクショナリ表を検索すれば分かります。ディクショナリ表の検索方法については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
5. レジストリ LOB 用 RD エリアを再初期化する場合，レジストリ用 RD エリアも再初期化する必要があります（レジストリ LOB 用 RD エリアだけの再初期化はできません）。また，再初期化後にプラグインモジュールが使用するレジストリ情報の再登録が必要です。

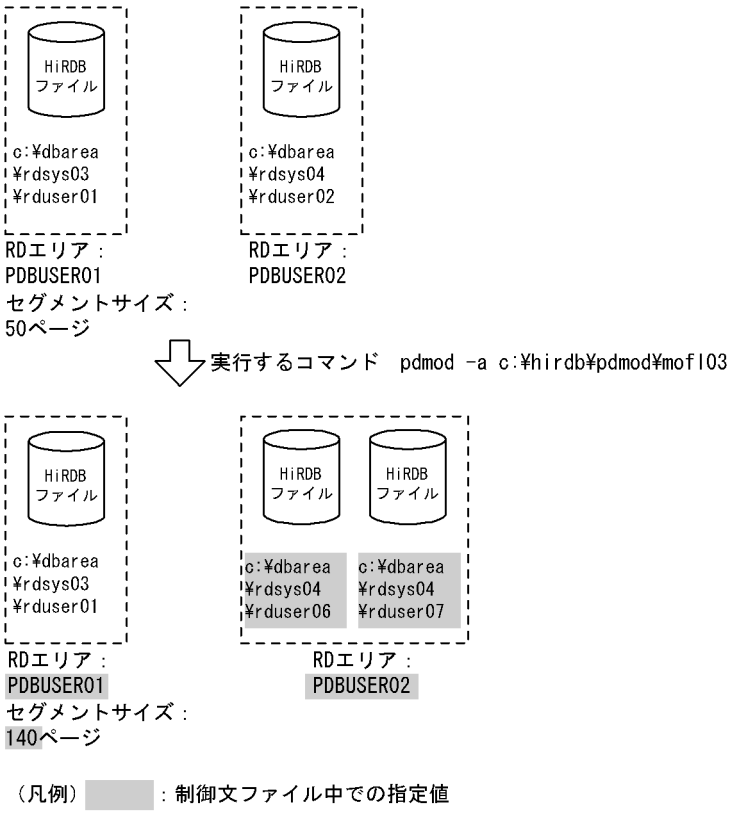
7.5.2 使用例

データベース構成変更ユティリティ（RD エリアの再初期化）の使用例を次に示します。

(1) RD エリアを再初期化する例

```
次に示すRDエリアを再初期化します。
・PDBUSER01
  セグメントサイズを140ページに変更します。
・PDBUSER02
  RDエリアに割り当てているHiRDBファイル（c:¥dbarea¥rdsys04¥rduser02）を，c:¥dbarea¥rdsys04¥rduser06及びc:¥dbarea¥rdsys04¥rduser07で割り当てるように変更します。
```

[概要]



（凡例）　：制御文ファイル中での指定値

[制御文ファイル（c:¥hirdb¥pdmod¥mofl03）の内容]

```
/* ユーザ用RDエリアの再初期化 */
initialize rdarea PDBUSER01   ..................................................1
            storage control segment 140 pages;   ..............................2
/* ユーザ用RDエリアの再初期化 */
initialize rdarea PDBUSER02   ..................................................1
            with reconstruction   ..............................................3
            file name "c:¥dbarea¥rdsys04¥rduser06"   ...........................4
              initial 100 segments   ...........................................5
            file name "c:¥dbarea¥rdsys04¥rduser07"   ...........................4
              initial 60 segments ;   ..........................................5
```

[説明]

1. 再初期化をする RD エリアの名称を指定します。
2. 変更後のセグメントサイズを指定します。
3. RD エリアを構成する，すべての HiRDB ファイルを変更する場合に指定します。
4. RD エリアに対して割り当て直す HiRDB ファイルの名称を指定します。
5. HiRDB ファイルのセグメント数を指定します。

7.5.3　制御文（initialize rdarea 文）

```
initialize rdarea RDエリア名
               〔with reconstruction〕
               〔open attribute　{INITIAL｜DEFER｜SCHEDULE} 〕
               〔page ページ長 characters〕
               〔storage control segment セグメントサイズ pages〕
               〔max entries 最大リスト登録数〕
               〔extension　{use 増分セグメント数 segments｜nouse} 〕
               〔without lock table〕
               〔file name ”HiRDBファイルシステム領域名¥HiRDBファイル名”
                  initial HiRDBファイル内セグメント数 segments
                 〔file name ”HiRDBファイルシステム領域名¥HiRDBファイル名”
                    initial HiRDBファイル内セグメント数 segments〕…〕
  ;
```

注

file name オペランド，initial オペランドを指定する場合は，必ず initialize rdarea 文の最後に指定してください。

(1)　RD エリア名

～＜識別子＞((1～30))

再初期化する RD エリアの名称を指定します。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

(2)　with reconstruction

RD エリアを構成するすべての HiRDB ファイルを変更する場合に指定します。

このオペランドを指定した場合は，file name オペランドに，変更する RD エリアを構成するすべての HiRDB ファイルを指定する必要があります。この場合，指定しなかった既存の HiRDB ファイルの定義（file name オペランドの定義）は無効となるため，使用しない HiRDB ファイルはすべて削除されます。

このオペランドを省略した場合は，RD エリアの再初期化実行前の指定値を引き継ぎます。

このオペランドを省略して HiRDB ファイルの定義を変更したい場合は，file name オペランドで既存の HiRDB ファイル名を指定し，変更したいオペランドに変更後の値を指定します。

(3)　open attribute　{INITIAL ｜ DEFER ｜ SCHEDULE}

RD エリアのオープン契機を指定します。

＜適用基準＞

特殊な運用形態でない場合は，通常 INITIAL を指定してください。

RD エリア数が多いために HiRDB の開始に時間が掛かるなどの不都合を回避したい場合は，DEFER を指定してください。

＜規則＞

1. このオペランドは，システム共通定義の pd_rdarea_open_attribute_use オペランドに Y を指定しているときだけ有効となります。
2. このオペランドはユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアの場合に指定できます。ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアに対してこのオペランドを指定しても，無視されます（ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアは，INITIAL 固定です）。
3. このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。
4. このオペランドの指定値は，HiRDB の開始モードに関係なく次回起動後から有効となります。
5. 該当する RD エリアに割り当てた HiRDB ファイルシステム領域が，ほかの RD エリアに共有される場合，共有する RD エリアのオープン契機はすべて同じにする必要があります。同じでない場合，HiRDB ファイルシステム領域を共有する RD エリアは，期待通りの効果が得られないことがあります。

INITIAL：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンして，RD エリア情報をメモリ上に常駐させるので，最初の SQL から高速な運用ができます。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はオープンで，それ以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. INITIAL の場合，クローズ状態の RD エリアはアクセスできません。

DEFER：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，RD エリアを最初にアクセスするときに HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させるので，2 回目以降のアクセスから高速な運用ができます。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，各 RD エリアに対して最初にアクセスするときに，該当する RD エリアをオープンします。以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. DEFER の場合，クローズ状態の RD エリアでもアクセスできます。

SCHEDULE：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，HiRDB 開始後，各トランザクション内で RD エリアに対して最初にアクセスするときに，HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させます。ただし，トランザクション終了時にそのト

ランザクション内でオープンした HiRDB ファイルシステム領域をクローズするため，2 回目以降のアクセス時でもオープン以降の処理をして，トランザクションに掛かる負荷は増加します。

＜注意事項＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，アクセスした RD エリアをトランザクション内でだけオープンし，トランザクション終了時にトランザクション内でオープンしたすべての RD エリアをクローズします。

   また，pdopen コマンドを実行することで，閉塞クローズ状態になるまでの間はオープン状態にできます。そのほかの運用コマンドを使用すれば，RD エリアの状態を任意に遷移させることもできます。なお，障害事象を検知した場合には，障害閉塞となります。

2. SCHEDULE の場合，クローズ状態の RD エリアでもアクセスできます。

(4) page ページ長 characters

～＜符号なし整数＞((4096～30720))

RD エリアを構成する HiRDB ファイルのページ長を変更する場合に，2,048 の倍数のバイト単位で指定します。HiRDB ファイルシステム領域作成時にセクタ長を指定した場合，セクタ長の倍数を指定してください。

このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。

データディクショナリ LOB 用 RD エリア，及びユーザ LOB 用 RD エリアの場合は，8,192 を指定してください。8,192 以外の値を指定した場合でも，8,192 が仮定されます。そのほかの RD エリアのページ長については，マニュアル「HiRDB システム導入・設計ガイド」を参照して適切な値を指定してください。

ここで指定したページ長は，HiRDB が RD エリアに対して入出力する最小の単位となります。

＜規則＞

1. 再初期化する RD エリアに表がある場合，ページ長を小さくすると行が格納できなくなるときがあります（データ格納時にエラーになります）。
2. FIX 表がある場合，次に示す値が行長より小さくなるときにはエラーとなります。行長は，ディクショナリ表の SQL_TABLES の ROW_LENGTH カラムを検索すれば確認できます。

   （↓変更後ページ長÷1000↓）×1000
3. ページ長を，グローバルバッファに割り当てられている RD エリアのページ長の最大値よりも大きくした場合はエラーとなります。
4. 再初期化対象の RD エリアにインデクスがある場合，インデクスのキー長が次の計算式より大きくなるようなページサイズを指定するとエラーになります。

   MIN（RD エリアのページサイズ÷2−1242，4036）

   （単位：バイト）

(5) storage control segment セグメントサイズ pages

～<符号なし整数>((1～16000))

セグメントサイズを変更する場合に指定します。1 セグメントの大きさを，ページ数で指定してください。

このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。

データディクショナリ LOB 用 RD エリア，及びユーザ LOB 用 RD エリアの場合は，1 を指定してください。1 以外の値を指定した場合でも，1 が仮定されます。そのほかの RD エリアのセグメントサイズについては，マニュアル「HiRDB システム導入・設計ガイド」を参照して適切な値を指定してください。

(6) max entries 最大リスト登録数

～<符号なし整数>((500～50000，かつ 500 の倍数))

リスト用 RD エリアに作成できるリストの最大数を，500 の倍数で指定します。500 の倍数でない値を指定した場合，500 の倍数に切り上げた値が仮定されます。

このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。

このオペランドをリスト用 RD エリア以外の RD エリアに指定した場合，エラーとなります。

(7) extension {use 増分セグメント数 segments | nouse}

再初期化する RD エリアに自動増分を適用するかどうかを指定します。このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。

RD エリアの自動増分については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜前提条件＞

RD エリアがある HiRDB ファイルシステム領域に-e オプション又は-a オプションが指定されている必要があります。

use 増分セグメント数 segments：

RD エリアの自動増分を適用します。

RD エリアの自動増分は，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアの場合に有効となります。ほかの RD エリアに use を指定しても，nouse が仮定されます。

自動増分の契機は，RD エリア内の空きセグメント数が自動増分契機に達したときです。pd_rdarea_extension_timing オペランドで指定できます。

増分セグメント数 ～<符号なし整数>((1～64000))

増分セグメント数を指定します。

- pdfmkfs -e を指定した HiRDB ファイルシステム領域の場合

HiRDB ファイルのエクステント数は最大 24 で，これを超えるとエラーとなります。また，HiRDB ファイルシステム領域の空きが不足したり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

HiRDB ファイルシステム領域当たりの最大増分回数は，HiRDB ファイルシステム領域作成時の指定値で決まるため，HiRDB ファイルシステム領域内のファイル数，増分頻度を考慮し最大増分回数を定義しておく必要があります。

- pdfmkfs -a を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルを格納しているディスクに，必要な HiRDB ファイルシステム領域を確保する容量が不足していたり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

- 指定した増分セグメント数を適用した場合に，RD エリアを構成する HiRDB ファイルの最終ファイルのサイズが 64 ギガバイトを超える増分セグメント数は指定できません（KFPX24241-E エラーとなります）。なお，このチェックは，最終ファイルのセグメント数（file オペランドを指定した場合は最終 file オペランドの initial に指定したセグメント数，省略した場合は RD エリアの最終ファイルの現在のセグメント数）に，それぞれ指定した増分セグメント数を 1 回分加算して算出したファイルサイズで判定します。

nouse：

RD エリアの自動増分機能を適用しません。

＜注意事項＞

1. 新たに追加される未使用セグメントは，RD エリアを構成する HiRDB ファイルの最終ファイルに追加されます。
2. HiRDB ファイルシステム領域の容量不足によって，未使用セグメントを確保できない場合，RD エリアの拡張又は再初期化をするか，データベース再編成ユティリティで表の再編成をしてください。
3. エクステントの数が上限値を超えた場合は，RD エリアがある HiRDB ファイルシステム領域のエクステントを統合するか，又は RD エリアに別の HiRDB ファイルシステム領域の HiRDB ファイルを追加してください。

   エクステントを統合するには，pdfbkup でバックアップを取得し，pdfmkfs で HiRDB ファイルシステム領域の初期設定をし，pdfrstr でバックアップから回復します。
4. RD エリア自動増分処理の開始から終了までの間，最終ファイルに対して排他が掛かります。

(8)　without lock table

再初期化する RD エリア内の表に対して，データベース構成変更ユティリティ（pdmod）が排他を掛けることを抑止したい場合に指定します。

この指定によって，再初期化する RD エリア内に複数の RD エリアに分割格納した表がある場合，再初期化対象外の RD エリアに格納された表に対して SQL が実行できるようになります。

without lock table オペランドの指定有無による再初期化中の SQL 実行可否を次の図，および表に示します。

図 7–4　without lock table オペランドの指定有無による再初期化中の SQL 実行可否

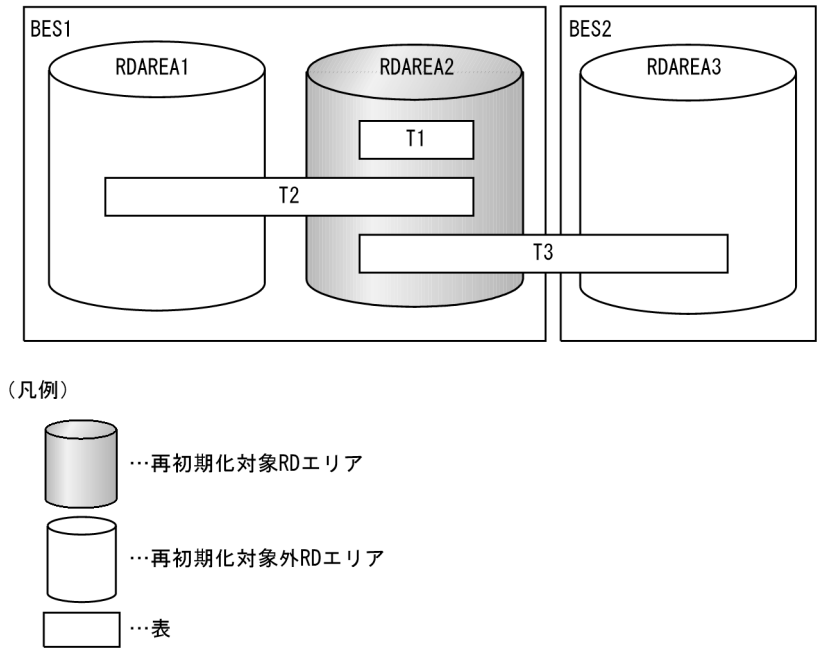


表 7–2　without lock table オペランドの指定有無による再初期化中の SQL 実行可否

操作対象の表	格納先 RD エリア	without lock table 指定による SQL 実行可否	
		指定なし	指定あり
T1	RDAREA2	× （閉塞中エラー（KFPA11920-E）又は排他待ち）	同左
T2	RDAREA2	× （閉塞中エラー（KFPA11920-E）又は排他待ち）	同左
	RDAREA1	× （排他待ち）	○
T3	RDAREA2	× （閉塞中エラー（KFPA11920-E）又は排他待ち）	同左
	RDAREA3	○	同左

（凡例）

○：指定できます。

×：指定できません。

(a)　RD エリア種別による without lock table オペランドの指定可否

再初期化対象 RD エリアの RD エリア種別による without lock table オペランドの指定可否を次の表に示します。

表 7–3　RD エリア種別によるオペランドの指定可否

RD エリア種別		without lock table オペランドの指定可否
ユーザ用	共用 RD エリア	×
	上記以外	○
データディクショナリ LOB 用 ユーザ LOB 用 リスト用 レジストリ用 レジストリ LOB 用		×

（凡例）

○：指定できます。

×：指定できません（KFPX24281-E エラーとなります）。

(b)　インデクス定義の条件による without lock table オペランドの指定可否

再初期化対象 RD エリアに格納している横分割表に対する，インデクス定義の条件による without lock table オペランドの指定可否を次の表に示します。

表 7–4　横分割表に対するインデクス定義の条件によるオペランドの指定可否

インデクス定義の条件					without lock table オペランドの指定可否
インデクスの定義	インデクスの横分割	インデクスの種別	表の分割方法	インデクスの分割方法	
なし	−	−	−	−	○
あり	なし	−	−	−	×
	あり	分割キーインデクス	−	−	○※
		非分割キーインデクス	サーバ内分割	−	○※
			サーバ間分割（サーバ内分割なし）	−	×
			サーバ間分割（サーバ内分割あり）	表の分割数と一致する	○※
				表の分割数と一致しない	×

（凡例）

○：指定できます。

×：指定できません（KFPX24281-E エラーとなります）。

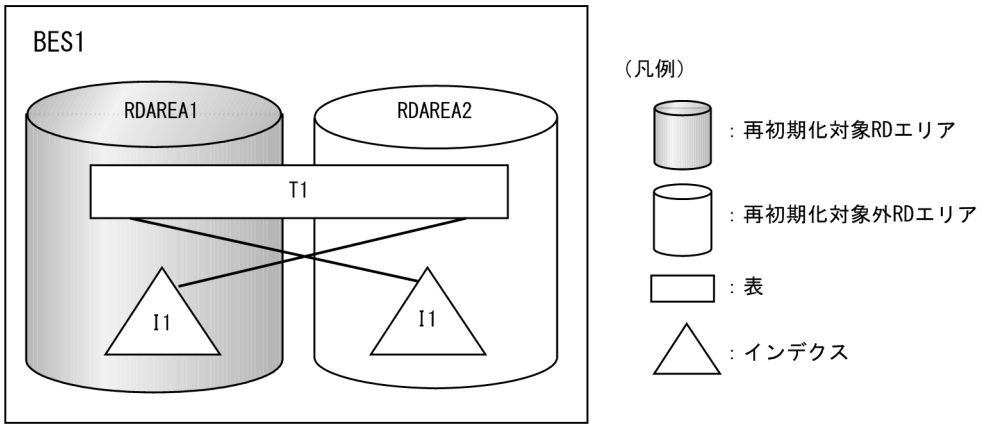
－：該当しません。

注※

横分割表とインデクスをどちらも再初期化対象 RD エリアに格納していて，インデクス定義時に指定したインデクス格納先 RD エリアの指定順序と，表定義時に指定した表格納先 RD エリアの指定順序が異なる場合は，without lock table オペランドは指定できません（KFPX24281-E エラーとなります）。without lock table オペランドを指定できないインデクスの定義例を次の図に示します。

図 7–5　without lock table オペランドを指定できないインデクス定義例

```
CREATE TABLE T1(C1 INT NOT NULL, C2 CHAR(10)) PARTITIONED BY C1
    IN((RDAREA1)100,(RDAREA2));
CREATE INDEX I1 ON T1(C1)
    IN((RDAREA2),(RDAREA1));
```



再初期化対象 RD エリアに複数の表を格納している場合，指定できないインデクスを定義した表が 1 つでもあれば，without lock table オペランドは指定できません。

(9)　file name "HiRDB ファイルシステム領域名¥HiRDB ファイル名"

～((167 文字以内))

再初期化する RD エリアを構成する，HiRDB ファイルシステム領域の名称と HiRDB ファイルの名称を指定します。

HiRDB ファイルシステム領域名¥HiRDB ファイル名は引用符（"）で囲んで指定してください。

＜規則＞

1. このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。
2. 引用符内に改行文字を含まないようにしてください。
3. 一つの RD エリアに対して，最大 16 個の HiRDB ファイルを割り当てられます。
4. 指定した HiRDB ファイルシステム領域に，pdmod が HiRDB ファイルを作成します。既に HiRDB ファイルがある場合は，pdmod が HiRDB ファイルをいったん削除し，新たに作成するため，HiRDB ファイルに対して権限がないときはエラーとなります。

5. HiRDB ファイルシステム領域名¥HiRDB ファイル名は，HiRDB システム内で重複しないようにしてください。
6. リスト用 RD エリアの場合は，使用目的が WORK の HiRDB ファイルシステム領域を使用してください。
7. HiRDB ファイルシステム領域名までは大文字，小文字を区別しませんが，HiRDB ファイル名は大文字，小文字を区別します。

HiRDB ファイルシステム領域名　～＜パス名＞

HiRDB ファイルシステム領域の名称を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞((1～30 文字))

HiRDB ファイルの名称を指定します。pl で始まる名称は指定できません。

(10)　initial HiRDB ファイル内セグメント数 segments

～＜符号なし整数＞

HiRDB ファイルのセグメント数を指定します。

このオペランドを省略した場合，RD エリアの再初期化実行前の指定値を引き継ぎます。

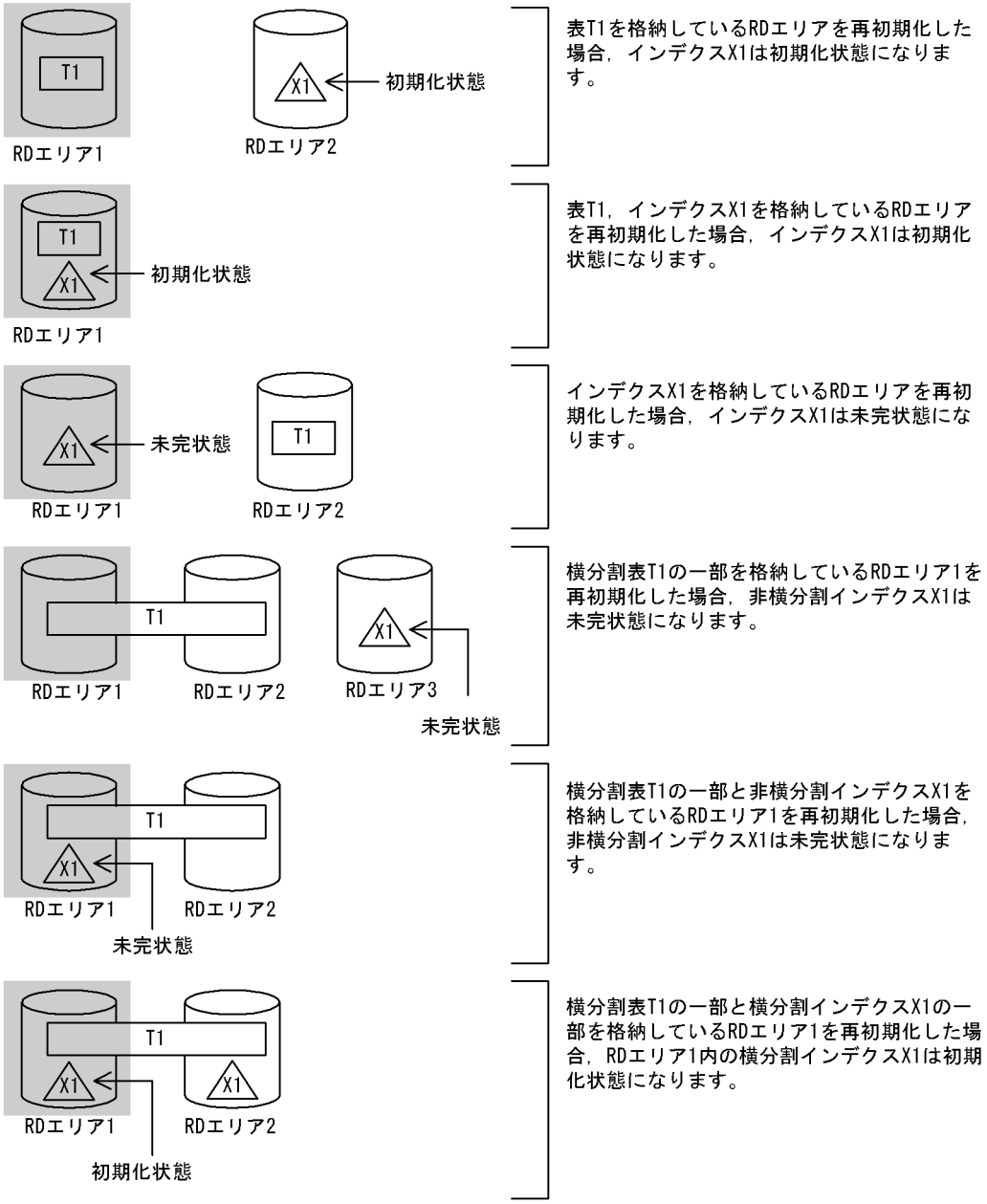
セグメント数は，データページ部分の総ページ数÷セグメントサイズで求めてください。データページ部分の総ページ数については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。また，HiRDB ファイルの大きさが 64 ギガバイトを超えるようなセグメント数は指定できません。

7.5.4　再初期化後の表，及びインデクスの状態

再初期化する RD エリアに表，及びインデクスがある場合の，再初期化後の状態を図「再初期化した後の表，及びインデクスの状態（1/2)」及び図「再初期化した後の表，及びインデクスの状態（2/2)」に示します。また，再初期化する RD エリアに表，LOB 属性の抽象データ型列，プラグイン列，及びプラグインインデクスがある場合の，再初期化後の状態を図「再初期化した後の表，LOB 属性の抽象データ型列，プラグインが提供する抽象データ型列，及びプラグインインデクスの状態」に示します。

なお，RD エリアの再初期化時に KFPX14255-W，KFPX24242-W メッセージが出力され，その原因が RD エリアの状態不正の場合は，マニュアル「HiRDB システム運用ガイド」の「RD エリアの容量又は属性を変更する方法（RD エリアの再初期化)」を参照してください。再初期化によってインデクスが未完状態になったり，LOB 属性の抽象データ型がアクセス禁止状態となったりする場合があります（これらはデータの再ロードで解除されます)。詳細な手順・注意点は，上記のマニュアルを参照してください。

図 7-6　再初期化した後の表，及びインデクスの状態（1/2）

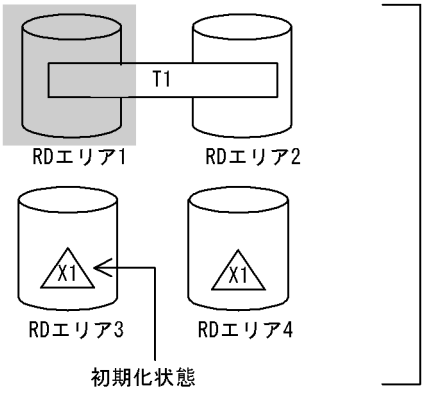


（凡例）　：再初期化対象RDエリアを示しています。

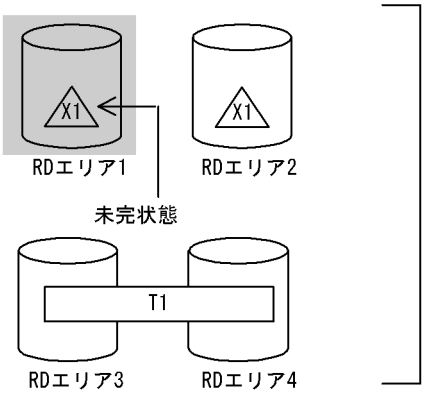
T1：表を示しています。X1：表T1のインデクスを示しています。

注　初期化状態とは，定義はあるがデータはない状態のことをいいます。この状態のインデクスは使用できます。未完状態とは，表とインデクスの整合性がとれていない状態のことをいいます。この状態のインデクスは使用できません。

図 7-7　再初期化した後の表，及びインデクスの状態（2/2）



横分割表T1の一部を格納しているRDエリア1を再初期化した場合，RDエリア3内の横分割インデクスX1は初期化状態になります。
なお，RDエリア1の表データとRDエリア3のインデクスデータが対応しています。



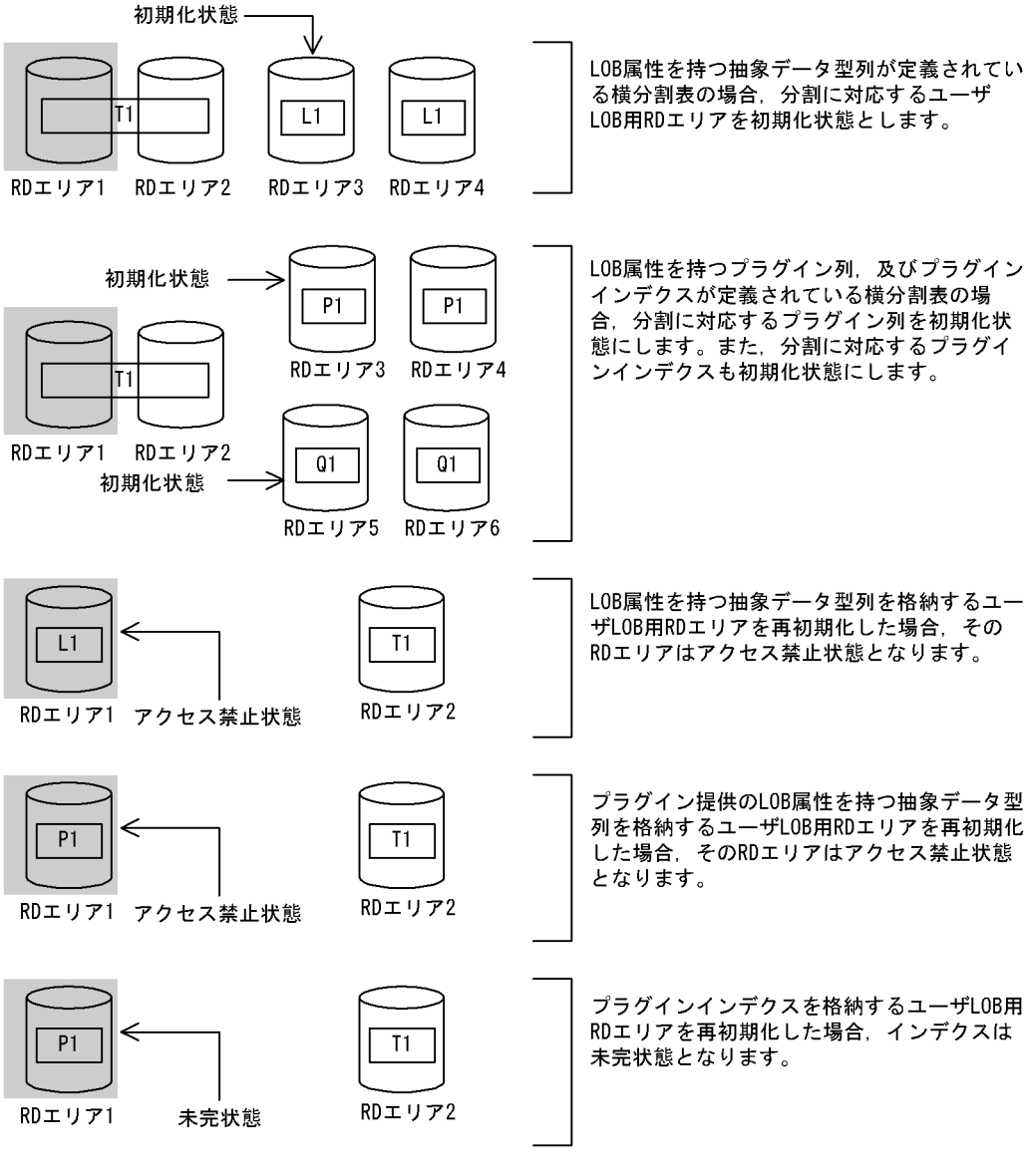
横分割インデクスX1の一部を格納しているRDエリア1を再初期化した場合，RDエリア1内の横分割インデクスX1だけが未完状態になります。

（凡例）　：再初期化対象RDエリアを示しています。

T1　：表を示しています。X1　：表T1のインデクスを示しています。

注　初期化状態とは，定義はあるがデータはない状態のことをいいます。この状態のインデクスは使用できます。未完状態とは，表とインデクスの整合性がとれていない状態のことをいいます。この状態のインデクスは使用できません。

図 7-8　再初期化した後の表，LOB 属性の抽象データ型列，プラグインが提供する抽象データ型列，及びプラグインインデクスの状態

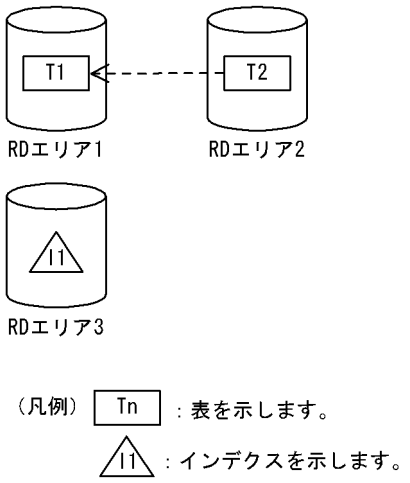


注　初期化状態とは，定義はあるがデータはない状態のことをいいます。この状態のインデクスは使用できます。アクセス禁止状態とは，データが削除されて定義だけの状態のことをいいます。この状態のインデクスは使用できません。未完状態とは，表とインデクスの整合性がとれていない状態のことをいいます。この状態のインデクスは使用できません。

再初期化する RD エリアに格納されている表に参照制約が定義されている場合，検査保留状態は変更されます。検査保留状態は，ディクショナリ表（SQL_TABLES 表の CHECK_PEND 列，及び SQL_REFERENTIAL_CONSTRAINTS 表の CHECK_PEND 列），及び RD エリア中の表情報で管理しています。検査保留状態については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

システム定義の pd_check_pending オペランドに USE を指定している場合，RD エリアを再初期化したときに変更される検査保留状態の例を次に示します。

1. インデクスだけを格納している RD エリアがある場合



T2（参照表）が T1（被参照表）を参照していて，かつ RD エリア 3 が T1 の主キーに付いているインデクスだけを格納している場合，再初期化で変更される検査保留状態を次に示します。

再初期化する RD エリア	T2（参照表）の検査保留状態	
	ディクショナリ表	RD エリア 2 の表情報
RD エリア 1	P	P
RD エリア 2	N	N
RD エリア 3	−	−

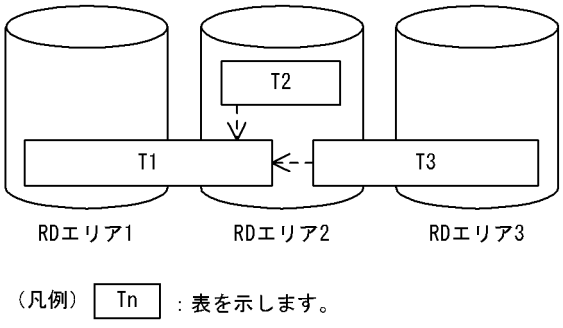
（凡例）

P：検査保留状態を設定します。

N：検査保留状態を解除します。

−：検査保留状態を変更しません。

2. 横分割表の場合



T2（参照表）及び T3（参照表）が T1（被参照表）を参照していて，かつ T1 及び T3 が横分割表の場合，再初期化で変更される検査保留状態を次に示します。

再初期化する RD エリア	T2（参照表）の検査保留状態		T3（参照表）の検査保留状態		
	ディクショナリ表	RD エリア 2 の表情報	ディクショナリ表	RD エリア 2 の表情報	RD エリア 3 の表情報
RD エリア 1	P	P	P	P	P
RD エリア 2	N	N	P	N	P
RD エリア 3	−	−	−	−	N

（凡例）

P：検査保留状態を設定します。

N：検査保留状態を解除します。

−：検査保留状態を変更しません。

7.5.5　実行結果リストの出力

RD エリアの再初期化を実行した場合，実行結果リストが出力されます。出力される実行結果リスト（表，インデクス情報）を次に示します。

```
                                [1]       [2]     [9]
pdmod -information-          yyyy-mm-dd  hh:mm:ss  db1
   initialize rdarea name =   USER_PUBLIC_1 [3]
     initialized table
  [4] USERA.在庫
      USERA.社員
     initialized index                rdarea name
  [5] USERA.従業員番号              [6] USER_PUBLIC_1
      USERA.品名                       USER_PUBLIC_2
    unfinished index                  rdarea name
  [7] USERA.氏名                    [8] USER_PUBLIC_1
    initialize LOB data(table name)    rdarea name
 [10] USERA.在庫                   [11] USER_LOB_1
    initialized sequence
 [12] USERA.商品通番
    pending status table                constraint name
 [13] USERA.給与                [14] *氏名1
      USERA.課長                [15] 氏名2
    release pending status table
 [16] USERA.社員
 [17] *USERA.品名
    access prohibited(LOB rdarea name) table name
 [18] USER_LOB_2                   [19] USERA.住所
```

[説明]

1. pdmod を実行した日付

   yyyy - mm - dd（年(西暦)-月-日）

2. pdmod を実行した時刻

hh : mm : ss（時 : 分 : 秒）

3. 再初期化した RD エリア名称
4. 再初期化した RD エリア内の表識別子
   認可識別子．表識別子
5. 初期化されたインデクス
   認可識別子．インデクス識別子
6. インデクスのデータが削除された RD エリアの名称
7. インデクスのデータが削除された後，未完状態（使用不可の状態）となったインデクス
8. 未完状態（使用不可の状態）となったインデクスの RD エリア名称
9. HiRDB 識別子
10. LOB 列構成基表の表識別子
11. ユーザ LOB 用 RD エリアの名称
12. 初期化された順序数生成子
13. 検査保留状態となった表の表名
14. *が付いている制約名は再初期化実行前から検査保留状態であった制約名
15. 検査保留状態となった制約名
16. 検査保留状態が解除された表の表名
17. *が付いている表は検査保留状態が解除された分割表の表名※
18. アクセス禁止状態となった LOB 属性の抽象データ型列を格納しているユーザ LOB 用 RD エリア名
19. アクセス禁止状態となった LOB 属性の抽象データ型列に対応する表の表名

注※

分割表の検査保留状態が解除された場合，再初期化対象の RD エリアの表情報だけ検査保留状態が解除されるため，分割表を格納するほかの RD エリアが検査保留状態のままとなっていることがあります。また，分割表を格納するほかの RD エリアがすべて非検査保留状態でも，ディクショナリ表の検査保留状態は解除されません。

7.5.6　注意事項

1. RD エリアの再初期化後は，データベース複写ユティリティ（pdcopy）で再初期化した RD エリア，マスタディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアのバックアップを取得しておいてください。データベース複写ユティリティいついては，「データベース複写ユティリティ（pdcopy）」を参照してください。

2. RD エリアの再初期化処理中に異常が発生した場合，再初期化対象の RD エリアの回復処理はしません。この場合，既に HiRDB ファイルが削除されていたり，初期化中でデータベースが破壊されていたりすることがあるため，必ずこの RD エリアに対して再度再初期化を実行してください。
3. 再初期化が正常終了した後の RD エリアは，コマンド閉塞状態かつクローズ状態のままです。したがって，この RD エリアを使用する場合には，運用コマンドで RD エリアを閉塞解除及びオープン状態にする必要があります。
4. RD エリアを構成する HiRDB ファイルの個数，又は名称を変更した場合，使用しなくなった HiRDB ファイルは削除されます。
5. 再初期化後の RD エリア内の格納データ，定義情報，及びユーザの作業は次のようになります。

RD エリア内に表がある場合：

RD エリア内のデータはすべて削除され，表の定義情報だけが引き継がれます。表が分割表の場合は，再初期化対象の RD エリアに格納されているデータだけ削除されます。その表にインデクスがある場合は，削除された表のデータに対応するインデクスのキーデータも同時に削除され，定義情報だけが引き継がれます。ただし，表が分割表でかつインデクスが分割キーに対応していない場合は，インデクスは未完状態となります。

表に LOB 列がある場合は，その LOB データも削除されます。

表に抽象データ型列がある場合は，通常の表と同じように扱われます。ただし，LOB 属性の抽象データ型列がある場合は，対応するユーザ LOB 用 RD エリア内の抽象データ型列のデータも削除されます。

表にプラグインインデクスがある場合は，プラグインインデクスのデータも同時に削除され，定義情報だけが引き継がれます。

再初期化対象 RD エリアに格納されている表の，インデクス及び LOB 列がほかの RD エリアに格納されている場合，インデクス及び LOB 列を格納している RD エリアは閉塞解除状態かつオープン状態でなければなりません。

[ユーザの作業]

再初期化後は，データベース作成ユティリティ，データベース再編成ユティリティ，UAP などを使用してデータを回復します。ただし，UAP を使用する場合，未完状態のインデクスがあるとアクセスできません（KFPA11879-E エラー）。そのため，次のどちらかの作業をする必要があります。

・データベース再編成ユティリティ（pdrorg）でインデクスを定義している表を再編成（インデクス一括作成モード）するか，又はインデクスを再作成します。

・インデクスを定義している表を初期化（SQL の PURGE TABLE 文を実行）し，表及びインデクスのデータを削除します。

RD エリア内にインデクスがある場合：

RD エリア内のインデクスのキーデータはすべて削除されます。インデクスに対応するデータが，ほかの RD エリアにある場合，インデクスは未完状態となります。

[ユーザの作業]

再初期化後にインデクスが初期化状態となっていたら，そのまま使用できます。未完状態になっていたら，次のどれかの作業をする必要があります。

・データベース再編成ユティリティ（pdrorg）でインデクスを定義している表を再編成（インデクス一括作成モード）するか，又はインデクスを再作成します。

・インデクスを定義している表を初期化（SQL の PURGE TABLE 文を実行）し，表及びインデクスのデータを削除します。

・インデクスを定義している表を格納している RD エリアを再初期化します。ただし，インデクスを格納している RD エリアは閉塞解除状態かつオープン状態でなければなりません。

RD エリアがユーザ LOB 用 RD エリアの場合：

RD エリア内の LOB データはすべて削除されます。ただし，その LOB データに対応する LOB 列構成基表のデータはそのまま残ります。LOB 列構成基表のデータが残っている場合は，LOB 列は長さ 0 のデータとして扱われます。

[ユーザの作業]

再初期化後は，データベース作成ユティリティ，データベース再編成ユティリティ，UAP などを使用してデータを回復します。

RD エリアが LOB 属性の抽象データ型列を格納しているユーザ LOB 用 RD エリアの場合：

RD エリア内のデータはすべて削除され，アクセス禁止状態となります。データに対応する LOB 属性の抽象データ型列構成基表（抽象データ型を含む表で，表から抽象データ型データを除いた部分）はそのまま残るため，LOB 属性の抽象データ型列構成基表から検索された場合，エラーメッセージ「KFPA11891-E」が出力されます。

[ユーザの作業]

アクセス禁止状態となった LOB 属性の抽象データ型列の格納ユーザ LOB 用 RD エリアをアクセスするには，次のどれかの作業をする必要があります。

・LOB 属性の抽象データ型列構成基表を初期化（SQL の PURGE TABLE 文を実行）し，データを削除します。

・データベース作成ユティリティ（pdload）を作成モード（-d）で実行します。

・LOB 属性の抽象データ型列構成基表格納 RD エリアを再初期化します。ただし，LOB 属性の抽象データ型列格納 RD エリアは閉塞解除かつオープン状態でなければなりません。

RD エリアがプラグインインデクスを格納しているユーザ LOB 用 RD エリアの場合：

RD エリア内のデータはすべて削除されます。再初期化後のプラグインインデクスは未完状態となります。未完状態となったプラグインインデクスにアクセスした場合，エラーメッセージ「KFPA11879-E」が出力されます。

[ユーザの作業]

未完状態となったプラグインインデクスは，データベース再編成ユティリティのインデクスの再作成で回復します。

6. レジストリ用 RD エリアの再初期化時に KFPX24231-W メッセージが出力され，レジストリ LOB 用 RD エリアを再初期化していない場合には，レジストリ LOB 用 RD エリアも再初期化する必要があります。また，再初期化後にプラグインが使用するレジストリ情報の再登録が必要です。
7. 再初期化対象の RD エリアに格納されている表を基にリストを作成している場合は，そのリストを使用する前に必ずリストを再作成してください。

8. 再初期化する RD エリアに被参照表，参照表，又は検査制約表が格納されている場合，検査保留状態を変更します。検査保留状態は，ディクショナリ表と RD エリア中の表情報で管理されていて，RD エリアの再初期化では次のとおり検査保留状態を変更します。なお，検査保留状態については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

   - 被参照表（主キーを定義した表）を格納している RD エリアを再初期化した場合，ほかの RD エリアで，その表と参照関係にある参照表（外部キーを定義した表）に検査保留状態を設定します。ただし，システム定義の pd_check_pending オペランドに NOUSE が指定されている場合は，検査保留状態は変更しません。また，主キーに付いているインデクスだけを格納している RD エリアを再初期化した場合も，検査保留状態は変更しません。

     参照表に検査保留状態を設定できなかった場合，KFPX24242-W のメッセージを出力します。この場合は次の要因が考えられます。

     ・参照表を格納している RD エリアが閉塞状態です。

     ・検査保留状態となる参照表が，ほかからアクセス中です。

     ・そのほかの I/O エラー，又は RD エリアのアクセスエラーが発生しました。

     このメッセージが出力された場合は，エラーの原因を取り除き，検査保留状態の強制設定をしてください。

   - 参照表を格納している RD エリアを再初期化した場合，参照表が検査保留状態となっているときは，検査保留状態を解除します。ただし，次の場合は RD エリア中の表情報の検査保留状態だけを解除し，ディクショナリ表の検査保留状態は解除しません。

     ・参照表が分割されている場合

     ・システム定義の pd_check_pending オペランドに NOUSE を指定している場合

     ディクショナリ表の検査保留状態が解除されなかった場合は，pdconstck で整合性チェックをして，ディクショナリ表の検査保留状態を解除してください。

   - 検査制約表を格納している RD エリアを再初期化した場合，検査制約を定義した表が検査保留状態となっていれば，検査保留状態を解除します。ただし，次の場合には RD エリア中の表情報の検査保留状態だけを解除し，ディクショナリ表の検査保留状態は解除しません。

     ・検査制約を定義した表が分割されている場合

     ・システム定義の pd_check_pending オペランドに NOUSE を指定している場合

     ディクショナリ表の検査保留状態が解除されなかった場合は，pdconstck で整合性チェックをして，ディクショナリ表の検査保留状態を解除してください。

9. データ連動の抽出の対象となる表が格納されている RD エリアは，再初期化をしないでください。再初期化をすると，以降の HiRDB Datareplicator 連携は不正になります。

10. 改竄防止表が格納されている RD エリアは再初期化できません。

11. インメモリ RD エリアは再初期化できません。いったんインメモリ化を解除してから RD エリアを再初期化してください。

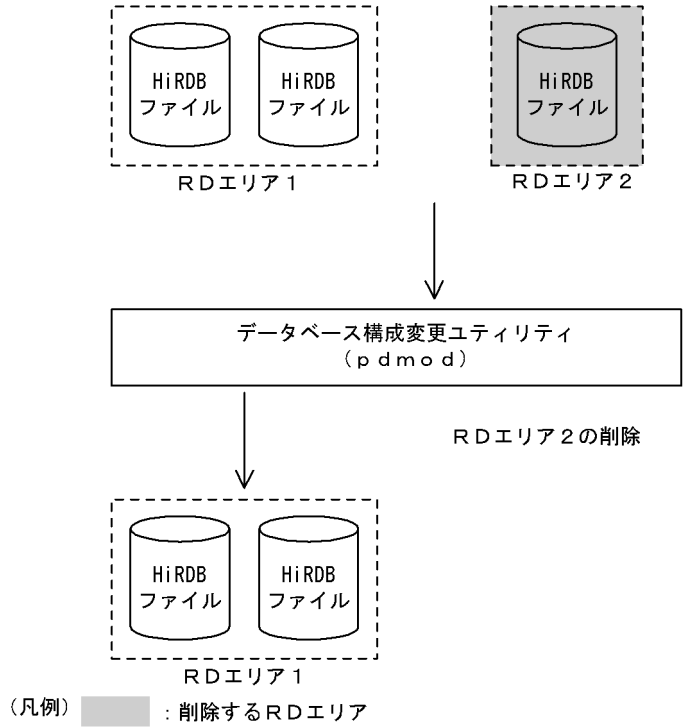
12. RD エリア内に順序数生成子がある場合，順序数生成子の定義情報だけが引き継がれ，現在値は順序数生成子定義時の状態に戻ります。

7.6　RD エリアの削除

7.6.1　機能

不要になった RD エリアを削除する機能です。この機能で RD エリアを削除し，別の RD エリアを追加することで，RD エリアの再定義ができます。RD エリアの削除の概要を次の図に示します。

図 7–9　RD エリアの削除の概要



●前提条件

1. RD エリア内に表，インデクス，LOB 列，及び順序数生成子がある場合，その RD エリアは削除できません。
2. 削除する RD エリアの状態は，閉塞状態かつクローズ状態でなければなりません。共用 RD エリアの場合は，すべてのバックエンドサーバで閉塞状態かつクローズ状態でなければなりません。

7.6.2　使用例

データベース構成変更ユティリティ（RD エリアの削除）の使用例を次に示します。

(1)　RD エリアを削除する例

```
次に示すRDエリアを削除します。
・PDBUSER04
```

[概要]



[制御文ファイル（c:¥hirdb¥pdmod¥mofl04）の内容]

```
/* ユーザ用RDエリアの削除 */
remove rdarea PDBUSER04 ;
```

[説明]

削除する RD エリアの名称を指定します。

7.6.3　制御文（remove rdarea 文）

```
remove rdarea RDエリア名
  ;
```

(1)　RD エリア名

～＜識別子＞((1～30))

削除する RD エリアの名称を指定します。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

表，インデクス，及び LOB 列が格納されている RD エリアの場合，その RD エリアは削除できません。

7.6.4　注意事項

1. RD エリアを削除した場合，その RD エリアで使用していた HiRDB ファイルも削除されます。
2. RD エリアの削除処理中に異常が発生した場合，削除対象の RD エリアの回復処理はされません。この場合，既に HiRDB ファイルが削除されていることがあるため，該当する RD エリアに対して必ず削除処理を再実行してください。
3. RD エリアを削除した後は，HiRDB を正常終了させてから，システム共通定義の pdbuffer オペランドを変更する必要があります。変更しないと必要以上にメモリを使用してしまいます。システム共通定義については，マニュアル「HiRDB システム定義」を参照してください。
4. 解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアを削除した場合，その中に格納されている解析情報表及び運用履歴表も削除されます。
5. インメモリ RD エリアは削除できません。いったんインメモリ化を解除してから RD エリアを削除してください。

7.7　RD エリアの移動

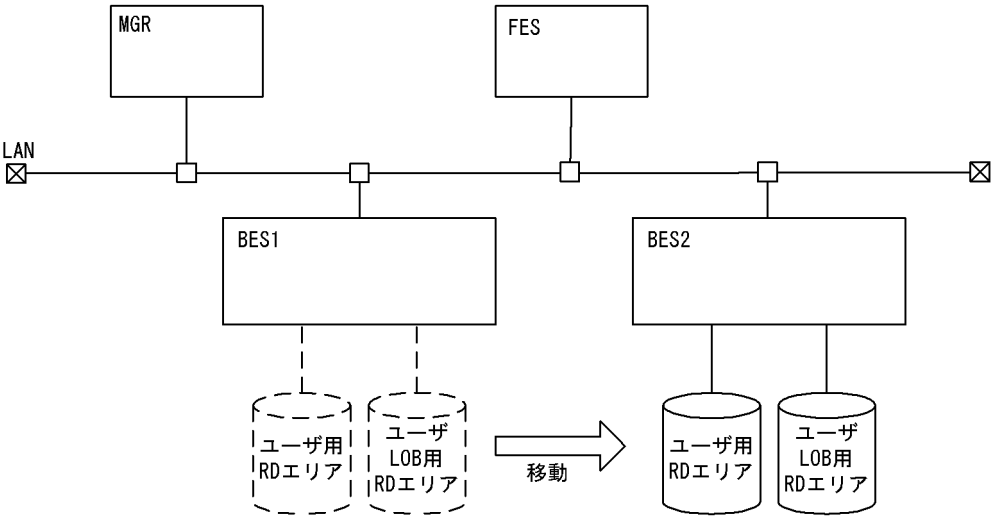
7.7.1　機能

HiRDB/パラレルサーバの場合に，RD エリアを別のサーバに移動する機能です。

既存のバックエンドサーバに RD エリアを移動する場合は，オンライン中の RD エリア移動機能を使用すると，HiRDB を再起動しなくても，移動先バックエンドサーバに移動した RD エリアが使用できるようになります。

RD エリアの移動の概要を次の図に示します。

図 7–10　RD エリアの移動の概要



（凡例）MGR：システムマネジャ
FES：フロントエンドサーバ
BES1，BES2：バックエンドサーバ

●前提条件

1. RD エリアを移動する場合，移動対象の RD エリアは閉塞状態かつクローズ状態でなければなりません。共用 RD エリアの場合は，すべてのバックエンドサーバで閉塞状態かつクローズ状態でなければなりません。
2. RD エリアを移動する場合，事前に pdcopy で移動対象の RD エリア，マスタディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアのバックアップを取得してください。
3. RD エリアを移動する場合，移動先のサーバのリソース所要量を見直しておく必要があります。見直した値で定義を修正してください。
4. RD エリアを移動する場合，関連する RD エリアをすべて移動しないとエラーとなります。関連する RD エリアとは，次のような RD エリアのことをいいます。
   - 非分割表の場合は，表，インデクス，及び LOB データを格納している RD エリア

- 横分割表の場合は，分割条件に対応する表，インデクス，及び LOB データを格納している RD エリア非分割キーインデクス（サーバ内分割をしている非分割キーインデクスも含む）がある場合は，そのインデクスの分割数が表の分割数と一致しているときだけ RD エリアの移動ができます。一致していない場合は，非分割キーインデクスを削除し，RD エリアの移動完了後，非分割キーインデクスを再作成してください。

  インデクスの分割数が表の分割数と一致する例，および非分割キーインデクスでインデクスの分割数と表の分割数が一致しない例を次の図に示します。

図 7–11　非分割キーインデクスでインデクスの分割数と表の分割数が一致する例

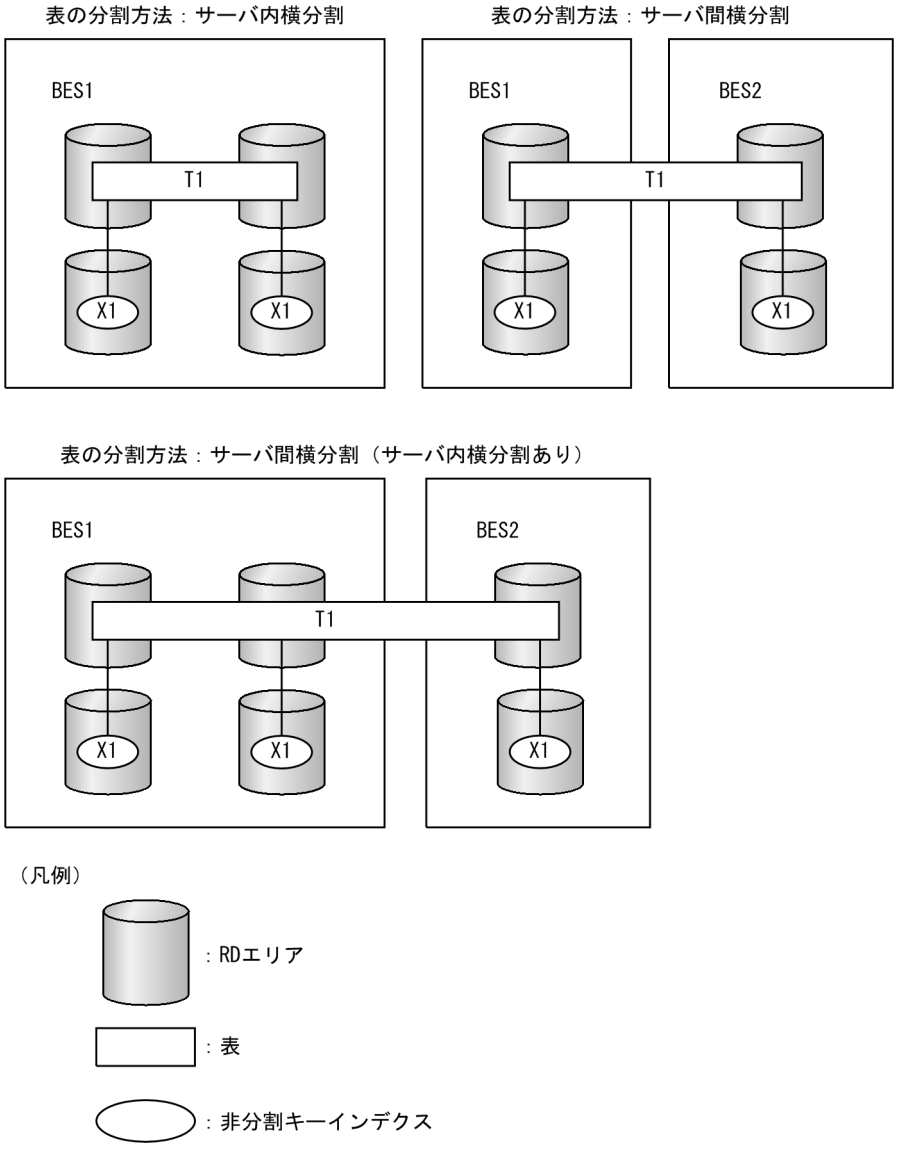
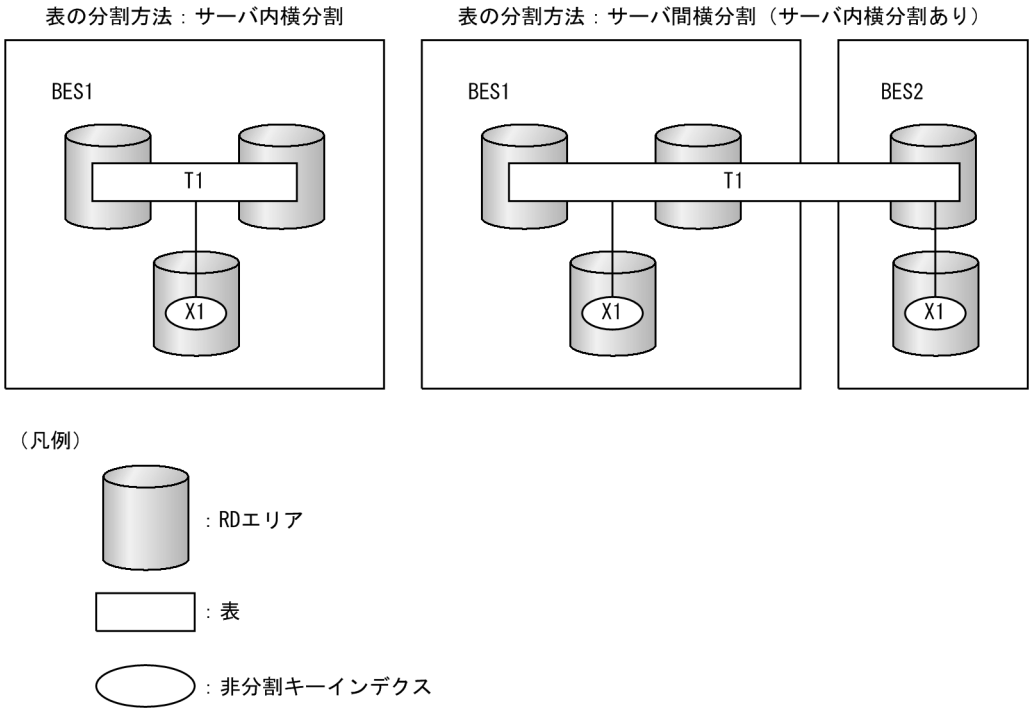


図 7–12　非分割キーインデクスでインデクスの分割数と表の分割数が一致しない例



非分割キーインデクスに UNIQUE を指定している場合は，再作成できないこともあります。詳細については，マニュアル「HiRDB SQL リファレンス」の CREATE INDEX の UNIQUE を参照してください。

非分割キー以外に主キーを定義している場合は，対象となる表のデータを退避した後に表を削除し，RD エリアの移動完了後に表の再定義，及びデータの回復をする必要があります。ただし，分割キー以外に主キーを定義するとユニーク指定のインデクスとなるため，表を再定義できないこともあります。

5. 一時表用 RD エリアに対して，RD エリアの移動はできません。実行するとエラーとなります。一時表用 RD エリアを移動する場合は，remove rdarea 文で該当する RD エリアを削除し，create rdarea 文で移動先のサーバを指定して再作成してください。
6. オンライン中の RD エリア移動機能を使用(norestart オペランドを指定)する場合，次のすべての条件を満たす必要があります。
   - 移動先のバックエンドサーバに，移動対象 RD エリアのページサイズ以上のバッファ長のグローバルバッファが必要です。
   - 移動先のバックエンドサーバの RD エリアの数，及び RD エリアを構成する HiRDB ファイルの数は，次の条件を満たすようにしてください。

     ・移動先のバックエンドサーバの RD エリアの数＋移動する RD エリアの数≦システム共通定義の pd_max_rdarea_no の指定値

     ・移動先のバックエンドサーバの RD エリアを構成する HiRDB ファイルの数＋移動する RD エリアを構成する HiRDB ファイルの数≦pd_max_file_no の指定値

- 移動対象 RD エリアに格納されている表を基に作成したリストがある場合は，移動前にリストを削除しておく必要があります。削除していないと，エラーとなります。削除したリストは移動後に再作成してください。
- ディクショナリサーバが移動対象 RD エリアの格納表に排他を掛けるため，RD エリアの移動に必要な排他制御用プールサイズを確保する必要があります。排他制御用プールサイズは，各サーバのサーバ定義の pd_lck_pool_size オペランドで指定してください。データベース構成変更ユティリティが必要とする排他資源数の見積もりについては，マニュアル「HiRDB システム定義」を参照してください。
- 次の条件をどちらも満たす場合，RD エリアの移動後のファイル転送は不要です。

  ・移動する RD エリアを構成する HiRDB ファイルシステム領域が共有ディスク装置にある

  ・移動元バックエンドサーバと移動先バックエンドサーバの両方から同じパス名で参照できる

  上記の条件を満たさない場合，pdmod コマンドで RD エリアを移動した後，rcp 又は ftp などでファイル転送を行う必要があります。そのため，オンライン中に RD エリアを移動した場合，ファイル転送が完了するまで移動した RD エリアは使用できません。

7. 系切り替え機能を適用した環境で RD エリアの移動を実行する場合，次のすべての条件を満たす必要があります。
   - RD エリア単位のディスク切り替えを行うため，RD エリアごとに共有ディスクを割り当ててください。ディスク定義数の上限はクラスタソフトウェアによって異なるため，使用しているクラスタソフトウェアのマニュアルで確認してください。
   - 次に示す RD エリアの移動を行った場合，移動した RD エリアの共有ディスクが移動先サーバの共有リソースとなるように，クラスタソフトウェアの設定を変更してください。また，OS のコマンドで移動元での共有ディスクの非活性化，及び移動先での共有ディスクの活性化を行ってください。

     ・影響分散スタンバイレス型系切り替えを適用した環境で，RD エリアを移動した場合

     ・影響分散スタンバイレス型系切り替え以外の系切り替え機能を適用した環境で，異なるユニットのバックエンドサーバに RD エリアを移動した場合

     共有リソースの設定変更手順は，次のとおりです。

     ・クラスタソフトウェアに MSFC を使用する場合

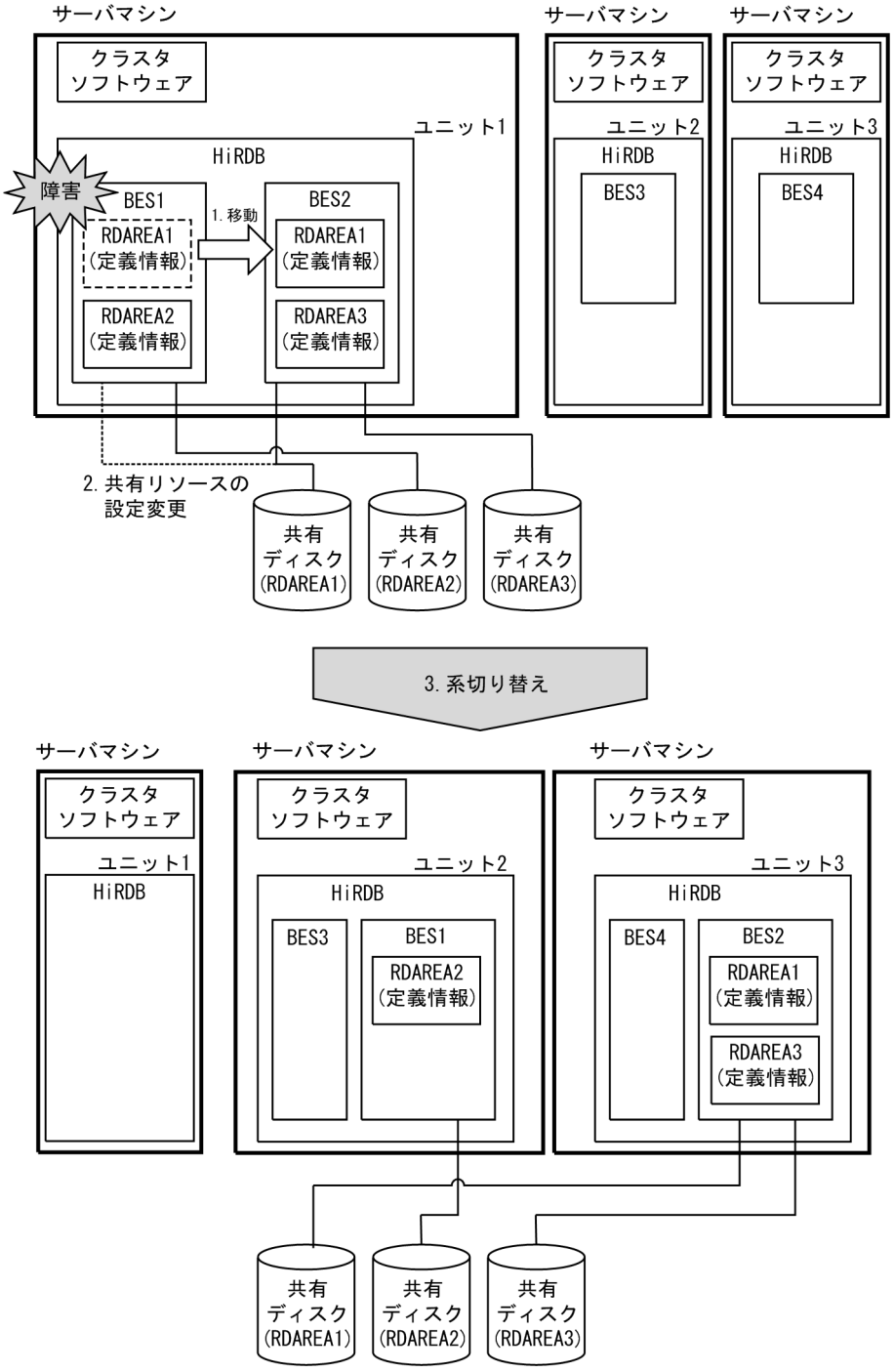
     MSFC のフェールオーバークラスタ管理を使用して，移動元のサービスから移動先のサービスに共有ディスクを移動してください。詳細は MSFC のドキュメントなどを参照してください。

     ・上記以外のクラスタソフトウェアを使用する場合

     クラスタソフトウェアのマニュアルを参照して，共有リソースの設定を変更してください。使用しているクラスタソフトウェアが，HiRDB サーバ稼働中に共有リソースの設定変更をできない場合は，HiRDB サーバを停止して設定を変更してください。

     系切り替え環境での RD エリアの移動手順を次の図に示します。

図 7–13　系切り替え環境での RD エリアの移動手順



[説明]

影響分散スタンバイレス型系切り替え機能を使用し，ユニット 1 に障害が発生した場合は，ユニット 2 で BES1 がゲスト BES として処理を実行し，ユニット 3 で BES2 がゲスト BES として処理を実行します。

1.RD エリア移動機能で BES1 の RDAREA1 を BES2 に移動します。

2.RDAREA1 の共有ディスクが BES2 の共有リソースとなるように，クラスタソフトウェアの設定を変更します。また，OS のコマンドで移動元での共有ディスクの非活性化，及び移動先での共有ディスクの活性化を行います。

3.ユニット 1 に障害が発生し，BES1 と BES2 が系切り替えすると，BES1 の共有ディスクはユニット 2 と接続し，BES2 の共有ディスクはユニット 3 と接続します。そのため，BES2 に移動した RDAREA1 は系切り替え後も使用できます。

上記手順 2.の共有ディスクの設定変更を行わないと，系切り替え後にユニット 3 から RDAREA1 の共有ディスクを参照できなくなり，RDAREA1 が使用できない状態になります。

7.7.2　使用例

データベース構成変更ユティリティ（RD エリアの移動）の使用例を次に示します。

(1)　RD エリアを移動する例

次に示すRDエリアを，別のバックエンドサーバbes2へ移動します。
・PDBUSER01，PDBUSER02，PDBULOB01

[実行するコマンド]

```
pdmod -a c:¥hirdb¥pdmod¥mofl01
```

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
 /* RDエリアの移動 */
  move rdarea PDBUSER01,PDBUSER02,PDBULOB01   .......1
              to bes2 ;   ...........................2
```

[説明]

1. 移動する RD エリアの名称を指定します。
2. 移動先のバックエンドサーバの名称を指定します。

7.7.3　制御文（move rdarea 文）

```
move rdarea 〔norestart〕
            RDエリア名
              〔globalbuffer バッファ名〕
             〔, RDエリア名
                   〔globalbuffer バッファ名〕…〕
            to サーバ名
  ;
```

(1)　norestart

RD エリアの移動完了後，HiRDB を再起動しないで RD エリアを使用する場合に指定します。オンライン中に RD エリアを移動できます。

(2) RD エリア名

～＜識別子＞((1～30))

移動する RD エリアの名称を指定します。

RD エリアを複数指定する場合，同一サーバ内の RD エリアでなければなりません。

RD エリア名には関連する RD エリアをすべて指定しなければなりません。関連する RD エリアについては，「機能」を参照してください。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

(3) globalbuffer バッファ名

～＜識別子＞((1～16))

移動する RD エリアに一時的にグローバルバッファを割り当てたい場合，移動先のバックエンドサーバにあるグローバルバッファの名称を指定します。

グローバルバッファについては，pdbufls コマンドで確認できます。

＜規則＞

1. このオペランドを指定する場合は，norestart オペランドの指定が必要です。norestart オペランドの指定がない場合，このオペランドを指定したときにはエラーとなります。
2. インデクス用，又は LOB 用のグローバルバッファは指定できません。
3. 指定するグローバルバッファのグローバルバッファ長は，移動対象 RD エリアのページ長以上でなければなりません。グローバルバッファ長は，pdbufls コマンドで確認できます。
4. ここで指定したグローバルバッファの割り当ては，次回の HiRDB の正常開始時，又は計画停止後の再開始時には無効となります。そのため，次回の HiRDB の正常開始時，又は計画停止後の再開始時には，システム共通定義のグローバルバッファを割り当てます。
5. このオペランドを省略した場合，移動した RD エリアにはグローバルバッファを割り当てません。そのため，移動した RD エリアは次回 HiRDB の正常開始後に使用できます。
6. このオペランドを指定して，HiRDB がグローバルバッファの割り当てに失敗した場合，RD エリアの移動はエラーとなります。

(4) to サーバ名

～＜識別子＞((1～8))

移動先のサーバ名（バックエンドサーバ）を指定します。

バックエンドサーバ以外のサーバ名，又は HiRDB にないサーバ名を指定した場合，エラーとなります。

7.7.4　実行結果リストの出力

RD エリアの移動を実行した場合，実行結果リストが出力されます。

出力される実行結果リストを次に示します。

```
                                  [1]       [2]        [3]
pdmod -information-           yyyy-mm-dd  hh:mm:ss     db1
                     [4]     [5]
   move rdarea from bes1 to bes2
     rdarea name = RDUSER1 [6]
       table name
     [7] USERA.在庫
         USERA.社員
       index name
     [8] USERA.従業員番号
         USERA.品名
       sequence name
     [9] USERA.商品通番
       LOB data(table name)
    [10] USERA.在庫
```

[説明]

1. pdmod を実行した日付
   yyyy - mm - dd（年(西暦)-月-日）
2. pdmod を実行した時刻
   hh : mm : ss（時 : 分 : 秒）
3. HiRDB 識別子
4. 移動元のサーバ名
5. 移動先のサーバ名
6. 移動した RD エリア名
7. 移動した RD エリアにある表
   認可識別子.表識別子
8. 移動した RD エリアにあるインデクス又はプラグインインデクス
   認可識別子.インデクス識別子
9. 移動した RD エリアにある順序数生成子
   認可識別子.順序数生成子識別子
10. 移動した RD エリアにある LOB 列又は LOB 属性の抽象データ型列
    認可識別子.表識別子

7.7.5　注意事項

1. RD エリアの移動中の UAP，ユティリティなどの実行について次に示します。

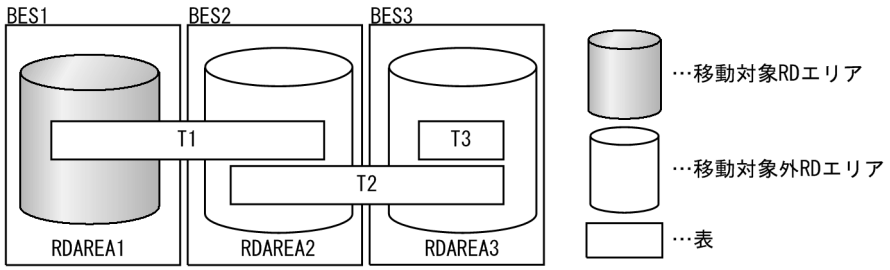
   - オンライン中の RD エリア移動機能を使用しない場合(norestart オペランドを指定しない場合)

     RD エリアの移動開始から HiRDB の再起動完了までの間は，UAP，ユティリティなどを実行しないでください。また，RD エリアの移動完了後に HiRDB を再起動しないで運用した場合は，動作は保証されません。

   - オンライン中の RD エリア移動機能を使用する場合(norestart オペランドを指定する場合)

     RD エリアの移動中に UAP，ユティリティなどを実行した場合の動作を次の図に示します。

     図 7–14　RD エリアの移動中の表アクセス時の動作



項番	操作対象の表	格納先RDエリア	UAP・ユティリティ実行時の動作
1	T1	RDAREA1	排他待ち
2		RDAREA2	排他待ち
3	T2	RDAREA2	実行可
4		RDAREA3	実行可
5	T3	RDAREA3	実行可

2. 移動元の RD エリアを構成する HiRDB ファイルの削除，及び移動先の RD エリアを構成する HiRDB ファイルの作成は，ユーザがする必要があります。また，移動先の RD エリアを構成する HiRDB ファイルのパス名は，移動元と同じでなければなりません。

3. RD エリアを移動した場合，その RD エリアの表を使用するルーチンがあると，ルーチンは無効となります。この場合，SQL の ALTER ROUTINE で再作成する必要があります。

4. RD エリアの移動後は，pdcopy で移動した RD エリア，マスタディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアのバックアップを取得する必要があります。

5. インメモリ RD エリアは移動できません。いったんインメモリ化を解除してから RD エリアを移動してください。

6. 順序数生成子を格納している RD エリアを移動する場合，順序数生成子を使用する表格納用 RD エリアも同時に移動してください。これらの RD エリアを異なるサーバに配置すると，順序数生成子を使用するたびに通信処理が発生し，処理性能が低下します。

7. globalbuffer オペランドで割り当てたグローバルバッファは，次回 HiRDB の正常開始，又は計画停止後の再開始のときに無効となります。開始するときには，システム共通定義で指定したグローバルバッファが割り当てられます。なお，globalbuffer オペランドには LOB 用グローバルバッファを指定できません。移動先のサーバで LOB 用グローバルバッファを割り当てる必要がある場合は，HiRDB の正常

停止及び正常開始による割り当てを行ってください。LOB 用グローバルバッファの割り当てについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

8. RD エリアの移動がエラー終了した場合，制御文に指定した RD エリアはすべて移動前の状態となります。
9. HiRDB Datareplicator の連動対象表が存在する RD エリアを移動する場合は，マニュアル「HiRDB Datareplicator」の「連動対象表の定義変更」を参照してください。

7.8　ディクショナリ表の属性定義変更

7.8.1　機能

ディクショナリ表の属性定義を変更したい場合に使用する機能です。ディクショナリ表の属性定義とは，次に示す二つのことを表します。

- ディクショナリ表の参照権限の変更
- ディクショナリ表の混在文字データの使用

(1)　ディクショナリ表の参照権限の変更

ディクショナリ表をアクセスする場合に，ユーザの持っている権限によって検索できる情報を限定できる機能です。データベース初期設定ユティリティ（define system 文）の dicinf オペランドで指定した内容を変更できます。

(2)　ディクショナリ表の混在文字データの使用

ディクショナリ表に混在文字データを使用できるようにする機能です。データベース初期設定ユティリティ（define system 文）の dictionary datatype mchar オペランドに nouse を指定（ディクショナリ表に混在文字データを使用しない）していた場合に，使用するように変更できます。

7.8.2　使用例

データベース構成変更ユティリティ（ディクショナリ表の属性定義変更）の使用例を次に示します。

(1)　ディクショナリ表の属性変更をする例

ディクショナリ表をアクセスする場合に，ユーザが持っている権限によって検索できる情報を限定するようにします。また，ディクショナリ表のデータに混在文字データを使用できるようにします。なお，この例では，HiRDB/シングルサーバの場合について説明します。

[実行するコマンド]

```
pdmod -a c:¥hirdb¥pdmod¥mofl01
```

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
/* ディクショナリ表の属性定義変更 */
alter system dicinf limited   .....................1
        dictionary datatype mchar use ;   .........2
```

[説明]

1. ディクショナリ表をアクセスする場合に，ユーザが持っている権限によって検索できる情報を限定します。
2. ディクショナリ表のデータに，混在文字データを使用できるようにします。

7.8.3 制御文（alter system 文）

```
alter system  〔dicinf  {unlimited | limited} 〕
              〔dictionary datatype mchar use〕
  ;
```

(1) dicinf {unlimited | limited}

ディクショナリ表をアクセスする場合に，ユーザが持っている権限によって検索できる情報を限定するかどうか（ディクショナリ表の参照権限の設定）を指定します。ディクショナリ表の参照権限の設定については，マニュアル「HiRDB システム運用ガイド」を参照してください。

unlimited

ディクショナリ表をアクセスする場合に，すべての情報を検索できます。

limited

ディクショナリ表をアクセスする場合に，ユーザが持っている権限によって検索できる情報を限定します。

DBA 権限を持っているユーザは，すべての情報を検索できます。DBA 権限を持っていないユーザは，利用できる情報だけ検索できます。

＜規則＞

1. ディクショナリ表を参照するビュー表を定義している場合，又は SQL オブジェクトのルーチンを定義している場合は，ディクショナリ表の参照権限は変更できません。
2. ディクショナリ表の参照権限を変更する場合，あらかじめディクショナリ表を参照するビュー表を削除しておく必要があります。
3. ディクショナリ表を参照する SQL オブジェクトのルーチンを定義している場合は，SQL オブジェクトを無効にする旨の警告メッセージを出力し，正常終了します。そのため，無効となったルーチンの SQL オブジェクトを ALTER PROCEDURE，又は ALTER ROUTINE で再作成する必要があります。無効となったルーチンの SQL オブジェクトはディクショナリ表を検索すれば分かります。ALTER PROCEDURE，又は ALTER ROUTINE についてはマニュアル「HiRDB SQL リファレンス」を，ディクショナリ表の検索方法についてはマニュアル「HiRDB UAP 開発ガイド」を参照してください。
4. limited を指定した場合，ディクショナリ表を検索する UAP のアクセス表数が増大するので，同時アクセス可能実表数（システム共通定義の pd_max_access_tables）の値を大きくする必要があります。一つのディクショナリ表につき，同時アクセス可能実表数は 5 必要となります。

(2) dictionary datatype mchar use

ディクショナリ表のデータに，混在文字データを使用する場合に指定します。

このオペランドを指定すると，ディクショナリ表の一部の列のデータ型が MVARCHAR になります。

＜適用基準＞

表名又は列名にマルチバイト文字を使用している場合，MVARCHAR に変更します。

＜規則＞

1. ディクショナリ表を参照するビュー表を定義している場合は，このオペランドを指定できません。このオペランドを指定するためには，あらかじめディクショナリ表を参照するビュー表を削除する必要があります。
2. ディクショナリ表を参照するルーチンの SQL オブジェクトを定義している場合は，SQL オブジェクトを無効にする旨の警告メッセージを出力し，正常終了します。そのため，無効となったルーチンの SQL オブジェクトを ALTER PROCEDURE，又は ALTER ROUTINE で再作成する必要があります。また，無効となったルーチンの SQL オブジェクトはディクショナリ表を検索すれば分かります。ALTER PROCEDURE，又は ALTER ROUTINE についてはマニュアル「HiRDB SQL リファレンス」を，ディクショナリ表の検索方法についてはマニュアル「HiRDB UAP 開発ガイド」を参照してください。

7.8.4 注意事項

ディクショナリ表をアクセスしている UAP，ユティリティ，又は運用コマンドがある場合に alter system 文を実行すると，エラーとなります。

7.9　RD エリアの属性変更

7.9.1　機能

RD エリアの次の属性を変更する機能です。

- RD エリアの名称
- RD エリアのオープン契機
- RD エリアの自動増分

●前提条件

RD エリアの属性変更をする場合，対象となる RD エリアが次のどれかのとき，RD エリアが閉塞状態かつクローズ状態でなければなりません。

- ユーザ用 RD エリア
- ユーザ LOB 用 RD エリア
- リスト用 RD エリア
- データディクショナリ LOB 用 RD エリア
- レジストリ用 RD エリア
- レジストリ LOB 用 RD エリア

RD エリアの属性変更をする場合，対象となる RD エリアが次のどれかのとき，RD エリアが閉塞状態かつオープン状態でなければなりません。

- データディクショナリ用 RD エリア
- データディレクトリ用 RD エリア

RD エリアの属性変更をした後の RD エリアは，属性変更をする前の状態を引き継ぎます。そのため，RD エリアを使用する前に，運用コマンドで RD エリアをオープン，閉塞解除状態にしてください。

7.9.2　使用例

データベース構成変更ユティリティ（RD エリアの属性変更）の使用例を次に示します。

(1)　RD エリアの属性変更をする例

次に示すRDエリアのオープン契機をSCHEDULEに変更します。なお，この例では，HiRDB/シングルサーバの場合について説明します。
・PDBUSER01（ユーザ用RDエリア）
・PDBULOB11（ユーザLOB用RDエリア）

[実行するコマンド]

```
pdmod -a c:¥hirdb¥pdmod¥mofl01
```

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
/* RDエリアの属性変更 */
alter rdarea PDBUSER01 open attribute SCHEDULE ;
alter rdarea PDBULOB11 open attribute SCHEDULE ;
```

[説明]

RD エリア（PDBUSER01，PDBULOB11）のオープン契機を SCHEDULE に変更します。

7.9.3 制御文（alter rdarea 文）

```
alter rdarea RDエリア名
            〔rename rdarea to 変更後のRDエリア名〕
            〔open attribute  {INITIAL｜DEFER｜SCHEDULE} 〕
            〔extension  {use 増分セグメント数 segments｜nouse} 〕
  ;
```

(1) RD エリア名

～＜識別子＞((1～30))

属性を変更する RD エリアの名称を指定します。

RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

インメモリ RD エリアの属性は変更できません。いったんインメモリ化を解除してから RD エリアの属性を変更してください。

(2) rename rdarea to 変更後の RD エリア名

～＜識別子＞((1～30))

変更後の RD エリアの名称を指定します。

変更後の RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。なお，変更後の RD エリア名に空白が含まれる場合は，引用符で囲んでください。

RD エリアの名称を変更する手順については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜規則＞

このオペランドはユーザ用 RD エリア，ユーザ LOB 用 RD エリア，リスト用 RD エリア，データディクショナリ用 RD エリア，データディレクトリ用 RD エリア，データディクショナリ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアの場合に指定できます。

＜注意事項＞

1. 変更後の RD エリアの名称に，既に存在する RD エリアの名称を指定した場合はエラーとなります。
2. RD エリア名を変更したときに警告メッセージが出力された場合の対処については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(3) open attribute {INITIAL ｜ DEFER ｜ SCHEDULE}

RD エリアのオープン契機を指定します。

＜適用基準＞

特殊な運用形態でない場合は，通常 INITIAL を指定してください。

RD エリア数が多いために HiRDB の開始に時間が掛かるなどの不都合を回避したい場合は，DEFER を指定してください。

＜規則＞

1. このオペランドは，システム共通定義の pd_rdarea_open_attribute_use オペランドに Y を指定しているときだけ有効となります。
2. このオペランドはユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアの場合に指定できます。ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアに対してこのオペランドを指定しても，無視されます（ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリア以外の RD エリアは，INITIAL 固定です）。
3. このオペランドを省略した場合，RD エリアの属性変更実行前の指定値を引き継ぎます。
4. このオペランドの指定値は，HiRDB の開始モードに関係なく次回起動後から有効となります。
5. 該当する RD エリアに割り当てた HiRDB ファイルシステム領域が，ほかの RD エリアに共有される場合，共有する RD エリアのオープン契機はすべて同じにする必要があります。同じでない場合，HiRDB ファイルシステム領域を共有する RD エリアは，期待通りの効果が得られないことがあります。

INITIAL：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンして，RD エリア情報をメモリ上に常駐させるので，最初の SQL から高速な運用ができます。

＜規則＞

1. HiRDB 開始時の RD エリアの初期状態はオープンで，それ以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. INITIAL の場合，クローズ状態の RD エリアはアクセスできません。

DEFER：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，RD エリアを最初にアクセスするときに HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させるので，2 回目以降のアクセスから高速な運用ができます。

＜規則＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，各 RD エリアに対して最初にアクセスするときに，該当する RD エリアをオープンします。以降は障害閉塞への遷移を除いて，運用コマンドを実行しないかぎり RD エリアの状態は遷移しません。
2. DEFER の場合，クローズ状態の RD エリアでもアクセスできます。

SCHEDULE：

HiRDB 開始時に HiRDB ファイルシステム領域をオープンしないで，HiRDB 開始後，各トランザクション内で RD エリアに対して最初にアクセスするときに，HiRDB ファイルシステム領域をオープンします。そのときに RD エリア情報をメモリ上に常駐させます。ただし，トランザクション終了時にそのトランザクション内でオープンした HiRDB ファイルシステム領域をクローズするため，2 回目以降のアクセス時でもオープン以降の処理をして，トランザクションに掛かる負荷は増加します。

＜規則＞

1. HiRDB 開始時の RD エリアの初期状態はクローズで，アクセスした RD エリアをトランザクション内でだけオープンし，トランザクション終了時にトランザクション内でオープンしたすべての RD エリアをクローズします。また，pdopen コマンドを実行することで，閉塞クローズ状態になるまでの間はオープン状態にできます。そのほかの運用コマンドを使用すれば，RD エリアの状態を任意に遷移させることもできます。なお，障害事象を検知した場合には，障害閉塞となります。
2. SCHEDULE の場合，クローズ状態の RD エリアでもアクセスできます。

(4) extension {use 増分セグメント数 segments | nouse}

属性変更をする RD エリアに自動増分を適用するかどうかを指定します。このオペランドを省略した場合，属性変更をする前の指定を引き継ぎます。

RD エリアの自動増分については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜前提条件＞

RD エリアがある HiRDB ファイルシステム領域に-e オプション又は-a オプションが指定されている必要があります。

use 増分セグメント数 segments：

RD エリアの自動増分を適用します。

RD エリアの自動増分は，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアの場合に有効となります。ほかの RD エリアに use を指定しても，nouse が仮定されます。

自動増分の契機は，RD エリア内の空きセグメント数が自動増分契機に達したときです。
pd_rdarea_extension_timing オペランドで指定できます。

増分セグメント数　～＜符号なし整数＞((1～64000))

増分セグメント数を指定します。

- pdfmkfs -e を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルのエクステント数は最大 24 で，これを超えるとエラーとなります。また，HiRDB ファイルシステム領域の空きが不足したり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

  HiRDB ファイルシステム領域当たりの最大増分回数は，HiRDB ファイルシステム領域作成時の指定値で決まるため，HiRDB ファイルシステム領域内のファイル数，増分頻度を考慮し最大増分回数を定義しておく必要があります。

- pdfmkfs -a を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルを格納しているディスクに，必要な HiRDB ファイルシステム領域を確保する容量が不足していたり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

- 指定した増分セグメント数を適用した場合に，RD エリアを構成する HiRDB ファイルの最終ファイルのサイズが 64 ギガバイトを超える増分セグメント数は指定できません（KFPX24241-E エラーとなります）。なお，このチェックは，RD エリアを構成する HiRDB ファイルの最終ファイルの現在のセグメント数に，指定した増分セグメント数を 1 回分加算して算出したファイルサイズで判定します。

nouse：

RD エリアの自動増分機能を適用しません。

＜注意事項＞

1. 新たに追加される未使用セグメントは，RD エリアを構成する HiRDB ファイルの最終ファイルに追加されます。
2. HiRDB ファイルシステム領域の容量不足によって，未使用セグメントを確保できない場合，RD エリアの拡張又は再初期化をするか，データベース再編成ユティリティで表の再編成をしてください。
3. エクステントの数が上限値を超えた場合は，RD エリアがある HiRDB ファイルシステム領域のエクステントを統合するか，又は RD エリアに別の HiRDB ファイルシステム領域の HiRDB ファイルを追加してください。

   エクステントを統合するには，pdfbkup でバックアップを取得し，pdfmkfs で HiRDB ファイルシステム領域の初期設定をし，pdfrstr でバックアップから回復します。
4. RD エリア自動増分処理の開始から終了までの間，最終ファイルに対して排他が掛かります。

7.10 HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更

7.10.1 機能

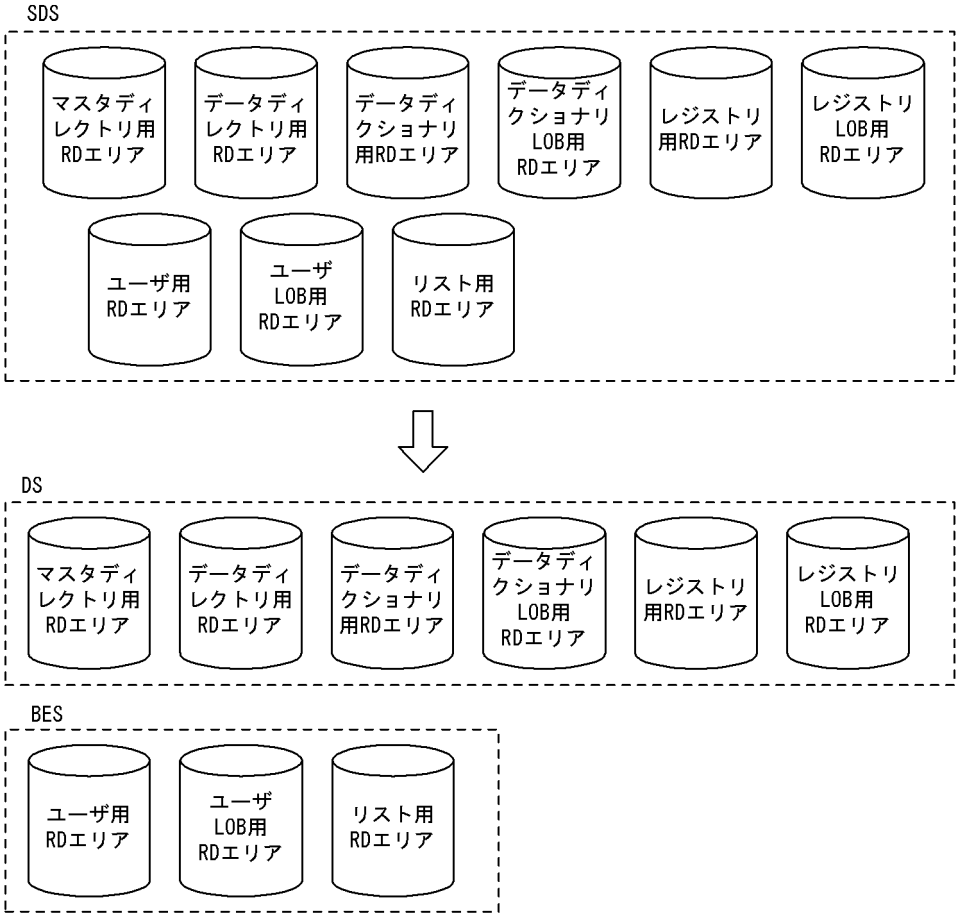
HiRDB システムを HiRDB/シングルサーバから HiRDB/パラレルサーバに移行する場合に，HiRDB/シングルサーバ構成の RD エリアを HiRDB/パラレルサーバ構成の RD エリアに変更する機能です。

すべての RD エリアを，次に示すように変更します。

- マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアを，HiRDB/パラレルサーバのディクショナリサーバ用の RD エリアとして変更します。
- ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアを HiRDB/パラレルサーバのバックエンドサーバ用の RD エリアとして変更します。

HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更の概要を次の図に示します。

図 7–15　HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更の概要



（凡例）SDS：シングルサーバ
　　DS　：ディクショナリサーバ
　　BES：バックエンドサーバ

7.10.2　使用例

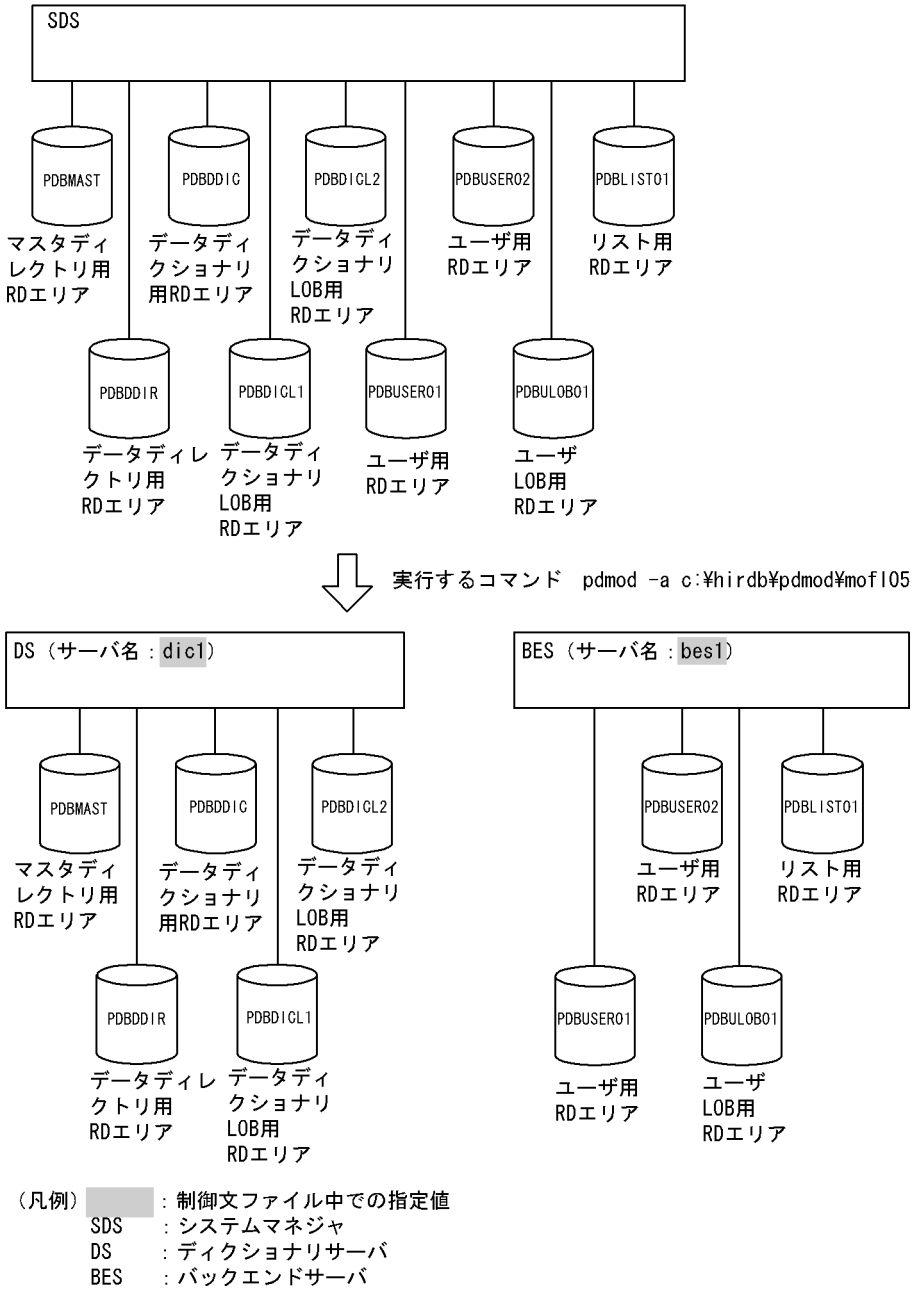
データベース構成変更ユティリティ（HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更）の使用例を次に示します。

(1)　HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成へ RD エリアを変更する例

次に示すRDエリアを，HiRDB/シングルサーバ構成からHiRDB/パラレルサーバ構成に変更します。
・PDBMAST
・PDBDDIR
・PDBDDIC
・PDBDICL1
・PDBDICL2
・PDBUSER01
・PDBUSER02

・PDBULOB01
・PDBLIST01

[概要]



[制御文ファイル（c:¥hirdb¥pdmod¥mofl05）の内容]

```
/* HiRDB/シングルサーバからHiRDB/パラレルサーバへのRDエリアの変更 */
alter HiRDB mode to parallel
        dictionary server name dic1   ..............1
        server name bes1 others ;   ................2
```

[説明]

1. 構成変更後のディクショナリサーバの名称を指定します。

2. 構成変更後のバックエンドサーバの名称を指定します。

7.10.3 制御文（alter HiRDB mode to parallel 文）

```
alter HiRDB mode to parallel
      dictionary server name サーバ名
      server name サーバ名
        {RDエリア名〔, RDエリア名〕…|others}
        〔server name サーバ名
          {RDエリア名〔, RDエリア名〕…|others}〕…
  ;
```

(1) dictionary server name サーバ名

～＜識別子＞((1～8))

HiRDB/シングルサーバから HiRDB/パラレルサーバへの変更後のディクショナリサーバの名称を指定します。

ディクショナリサーバには，マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディクショナリ LOB 用 RD エリアが配置されます。

(2) server name サーバ名

～＜識別子＞((1～8))

HiRDB/シングルサーバから HiRDB/パラレルサーバへの変更後のバックエンドサーバの名称を指定します。

バックエンドサーバには，ユーザ用 RD エリア，ユーザ LOB 用 RD エリア，及びリスト用 RD エリアが配置されます。

(3) {RD エリア名〔，RD エリア名〕...｜others}

RD エリア名〔，RD エリア名〕... ～＜識別子＞((1～30))

指定したサーバ名のバックエンドサーバに割り当てる RD エリアの名称を指定します。

RD エリアに ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，RD エリア名に空白が含まれる場合は，引用符で囲んでください。

others

RD エリア名で指定しなかった RD エリアを，すべて割り当てます。

RD エリア名を指定しないで others だけを指定した場合は，すべての RD エリアが割り当てられます。

7.10.4　注意事項

1. HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成へ RD エリアを変更後は，必ず HiRDB/シングルサーバを正常終了してください。
2. 横分割表を複数のバックエンドサーバに分割するような構成にする場合，非分割キーインデクスは分割されないので，HiRDB/パラレルサーバを起動するときに非分割キーインデクスは不整合となります。そのため，非分割キーインデクスは複数のバックエンドサーバに分割されるように，インデクスを再定義してください。
3. HiRDB/シングルサーバから HiRDB/パラレルサーバに移行後に，ディクショナリサーバを別の（追加する）サーバマシンに配置する場合は，HiRDB/パラレルサーバを起動する前にマスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアをディクショナリサーバのサーバマシンに移動しておいてください。
4. RD エリアを指定してサーバ構成を変更する場合，表と表に付けたインデクスを格納する RD エリアが異なるときには，それぞれの RD エリアを必ず同じサーバマシンに配置してください。ただし，横分割表の分割していないインデクスは，削除しておいた方が望ましいです。
5. ストアドプロシジャ，又はユーザ定義のストアドファンクションを登録している場合，移行後，すべてのストアドプロシジャ，又はユーザ定義のストアドファンクションは無効となります。移行後，無効となったストアドプロシジャ，又はユーザ定義のストアドファンクションは，HiRDB/パラレルサーバを起動して，ALTER PROCEDURE（ALL 指定），又は ALTER ROUTINE ですべての SQL オブジェクトを再作成する必要があります。ALTER PROCEDURE 及び ALTER ROUTINE については，マニュアル「HiRDB SQL リファレンス」を参照してください。
6. レジストリ機能を使用している場合，移行後，レジストリ機能初期設定ユティリティ（pdreginit）の-k renew 指定でレジストリ操作用ストアドプロシジャを再登録してください。また，次に示すプラグインを使用している場合は，pdplgrgst -a を実行して再登録してください。
   - HiRDB Image Search Plug-in
7. 移行するときに，システム共通定義 pd_system_id の HiRDB 識別子は変更しないでください。
8. 共用表及び共用インデクスが定義されている状態で，HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更はしないでください。変更する場合は，HiRDB に定義しているすべての共用表及び共用インデクスを削除してから実行してください。

7.11　監査人の登録（セキュリティ監査機能）

7.11.1　機能

セキュリティ監査機能を使用する場合，監査証跡表を管理・操作する主監査人，及び監査証跡表を操作する副監査人を登録します。セキュリティ監査機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

●前提条件

1. 主監査人は，HiRDB に一人だけ登録できます。既に主監査人を登録している場合に，主監査人の登録をするとエラー（KFPX24269-E）となります。
2. 副監査人は，HiRDB に複数人登録できます。副監査人は，主監査人を登録した後に登録できます。主監査人を登録しないで副監査人の登録をするとエラー（KFPX24272-E）となります。
3. 監査人に指定できるユーザを次の表に示します。指定できないユーザを登録するとエラー（KFPX24271-E）となります。

表 7–5　監査人に指定できるユーザ

指定するユーザ		指定可否	
		主監査人	副監査人
新規ユーザ		○	○
既存ユーザ	DBA 権限を持つユーザ	×	×
	パスワード無効アカウントロック状態のユーザ※1		
	簡易認証ユーザ※2		
	自分のスキーマ操作権限を他人に付与したユーザ※3		
	他人からスキーマ操作権限を付与されたユーザ※3		
	上記以外のユーザ	○	
上記以外	定義済みのロール名	×	×

（凡例）

○：指定できます。

×：指定できません。

注※1

パスワード無効アカウントロック状態については，マニュアル「HiRDB システム運用ガイド」の「パスワードの文字列制限」を参照してください。

注※2

簡易認証ユーザについては，マニュアル「HiRDB システム運用ガイド」の「OS ログインユーザの簡易認証機能」を参照してください。

注※3

スキーマ操作権限については，マニュアル「HiRDB システム運用ガイド」の「ユーザ権限の種類」を参照してください。

4. 主監査人の登録をすると，同時にスキーマも定義されます。したがって，監査人を登録する前に CREATE SCHEMA を実行しておく必要はありません。副監査人にはスキーマは定義されません。

7.11.2　使用例

データベース構成変更ユティリティ（主監査人の登録）の使用例を次に示します。

(1)　主監査人の登録をする例

認可識別子「HIRAUD」を主監査人として登録します。

[実行するコマンド]

```
pdmod -a c:¥hirdb¥pdmod¥mofl01
```

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
 create auditor for HIRAUD ;   ..............................1
```

[説明]

1. 認可識別子「HIRAUD」を主監査人として登録します。

7.11.3　制御文（create auditor 文）

```
create 〔deputy〕 auditor for 認可識別子
  ;
```

(1)　deputy

副監査人を登録する場合に指定します。このオペランドを省略した場合は主監査人を登録します。

(2)　認可識別子

～＜先頭が英字記号で始まる英数字記号＞((1～30))

監査人として登録するユーザの，認可識別子を指定します。

認可識別子に指定できるユーザについては，表「監査人に指定できるユーザ」を参照してください。

認可識別子の大文字小文字を区別したい場合は，引用符（"）で囲んでください。囲まない場合は大文字として扱われます。

7.11.4　注意事項

新規のユーザを主監査人，又は副監査人として登録した場合，登録直後のパスワードはそのユーザの認可識別子となります。また，既存のユーザを主監査人として登録した場合，パスワードを設定していないときは，登録直後のパスワードはそのユーザの認可識別子となります。監査人の登録をした後は，認可識別子となっているパスワードを，GRANT AUDIT 文で変更してください。パスワードの変更は登録された監査人自身が行ってください。

7.12　監査証跡表の作成（セキュリティ監査機能）

7.12.1　機能

セキュリティ監査機能を使用する場合に必要となる，監査証跡表を作成します。

●前提条件

1. 監査証跡表の作成をする前に，主監査人の登録をしておく必要があります。
2. 監査証跡表は HiRDB に一つだけ作成できます。既に監査証跡表が作成されている場合に，監査証跡表の作成をするとエラー（KFPX24269-E）になります。
3. 監査証跡表を格納する RD エリアには，主監査人だけが利用権限を持つユーザ用 RD エリアを指定する必要があります。共用 RD エリア，公用 RD エリア，又は主監査人以外のユーザが利用権限を持つ RD エリアを指定した場合，エラー（KFPX24270-E）となります。

7.12.2　使用例

データベース構成変更ユティリティ（監査証跡表の作成）の使用例を次に示します。

(1)　監査証跡表の作成をする例

RDエリア「RD00AUD」に監査証跡表「SQL_AUDIT_TRAIL」を作成します。

[実行するコマンド]

```
pdmod -a c:¥hirdb¥pdmod¥mofl01
```

[制御文ファイル（c:¥hirdb¥pdmod¥mofl01）の内容]

```
 create audit table SQL_AUDIT_TRAIL   ..............................1
                 in RD00AUD ;   ...................................2
```

[説明]

1. 監査証跡表「SQL_AUDIT_TRAIL」を作成します。
2. RD エリア「RD00AUD」に監査証跡表を作成します。

7.12.3　制御文（create audit table 文）

```
create audit table 〔表識別子〕
        in RDエリア名
  ;
```

(1) 表識別子

～＜識別子＞((1～30))《SQL_AUDIT_TRAIL》

監査証跡表の表識別子を指定します。省略した場合，SQL_AUDIT_TRAIL が仮定されます。

表識別子の大文字小文字を区別したい場合は，引用符（''）で囲んでください。囲まない場合は大文字として扱われます。

表識別子に in は指定できません。

(2) RD エリア名

～＜識別子＞((1～30))

監査証跡表を格納する RD エリア名（ユーザ用 RD エリア）を指定します。主監査人だけ利用権限がある RD エリアを指定してください。主監査人以外のユーザに利用権限がある RD エリアを指定した場合，エラー（KFPX24270-E）となります。

RD エリア名の大文字小文字を区別したい場合は，引用符（''）で囲んでください。囲まない場合は大文字として扱われます。

7.13　規則及び注意事項

7.13.1　規則

- pdmod は，HiRDB が稼働中のときだけ実行できます。
- pdmod は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
- 高速系切り替え機能を適用している場合，pdmod を実行したときは，待機系 HiRDB を再度開始する必要があります。待機系の再開始については，マニュアル「HiRDB システム運用ガイド」の「系の切り替え時間の短縮（ユーザサーバホットスタンバイ，高速系切り替え機能）」を参照してください。
- 1：1 スタンバイレス型系切り替え機能を適用している場合，pdmod を実行したときは，代替部を再度開始する必要があります。代替部の再開始については，マニュアル「HiRDB システム運用ガイド」の「1:1 スタンバイレス型系切り替え機能使用時の注意事項」を参照してください。
- 更新可能なオンライン再編成を実行する場合，準備を開始してから運用を完了するまで，pdmod コマンド（RD エリア再初期化，RD エリア拡張を除く）を実行しないでください。更新可能なオンライン再編成の準備及び運用の詳細については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。
- pdffsck コマンドの実行中には，pdmod コマンドを実行しないでください。実行すると，データベースを破壊するおそれがあります。
- JP1/Performance Management - Agent Option for HiRDB がレコード収集中に，pdmod コマンドを実行すると，不整合なレコードを生成，もしくはレコード自体が欠落することがあります。HiRDB 構成変更中に収集したレコードは無視してください。JP1/Performance Management - Agent Option for HiRDB のレコードについては，マニュアル「JP1 V11 JP1/PFM - Agent Option for HiRDB」を参照してください。

7.13.2　注意事項

(1)　ユティリティ実行後の結果について

pdmod の結果は，次の方法で確認できます。

- pddbls コマンド（-r ALL -a 指定）
- ディクショナリ表の検索
- データベース状態解析ユティリティ

(2) HiRDB ファイルシステム領域が壊れた場合について

pdmod の実行中に，電源断又は pdkill コマンドで実行が中断された場合，HiRDB ファイルシステム領域が壊れるときがあります。pdfls コマンドを実行して矛盾が検知された場合は，次に示すどちらかの手順で回復してください。

- pdfmkfs コマンドで HiRDB ファイルシステム領域を再度作成し，格納していた HiRDB ファイルは RD エリアごとにデータベース構成変更ユティリティを実行します。
- pdfbkup コマンド及び pdfrstr コマンドを実行して，HiRDB ファイルシステム領域のバックアップとリストアをします。

(3) バックアップの取得について

データベース構成変更ユティリティはログを無効化する処理を行っているため，データベース構成変更ユティリティを実行した後には，処理対象 RD エリアのバックアップを必ず取得してください。バックアップを取得しないと，処理対象 RD エリアが障害閉塞した場合，データベース構成変更ユティリティの実行前の状態にしか RD エリアを回復できません（データベース構成変更ユティリティ実行後の更新情報を回復できません）。

また，データベース構成変更ユティリティの操作によっては，システム用 RD エリアや関連する RD エリアと同期を取ってバックアップを取得する必要があります。詳細については，「HiRDB システム運用ガイド」の「同時にバックアップを取得する必要がある RD エリア」を参照してください。

(4) 連続実行について

HiRDB/パラレルサーバで間隔を空けずに pdmod コマンドを連続実行すると，KFPX24219-E メッセージを出力してエラー終了することがあります。複数の制御文を実行する場合は，制御文を 1 つの制御文にまとめて 1 回の pdmod コマンドで実行してください。もし，制御文を 1 つの制御文ファイルにまとめられない場合は，pdmod コマンドの実行間隔を 2 秒程度空けて実行してください。なお，先発の pdmod コマンドの実行が終了していることは，次のコマンドを実行し，DICU(5001)の排他が残っていないことを確認すると判断できます。

```
pdls  -d  lck  -a
```

8 データベース再編成ユティリティ（pdrorg）

この章では，表，及びインデクスのメンテナンスをするデータベース再編成ユティリティ（pdrorg）について説明します。

なお，IVS 対応 UTF-8 文字コード及び IVS 対応 UTF-16 文字集合を使用する場合は，この章の記載を次のように読み替えてください。

UTF-8 -> IVS 対応 UTF-8

utf-8 -> utf-8_ivs

UTF-16 -> IVS 対応 UTF-16

utf-16be -> utf-16ibe

utf-16le -> utf-16ile

8.1.1 pdrorg の機能

データベース再編成ユティリティ（pdrorg）は，表，及びインデクスのメンテナンスをするユティリティです。pdrorg の用途と機能を次の表に，pdrorg の機能の概要を図「pdrorg の機能の概要」に示します。

表 8–1 pdrorg の用途と機能

種別	用途	pdrorg の機能							
		表の再編成	表のアンロード	表へのリロード	ディクショナリ表の再編成	インデクスの一括作成	インデクスの再作成	インデクスの再編成	ディクショナリ表のメンテナンス
表	表の格納効率をよくしたい場合（pddbst で確認）	○			○				
	KFPH00212-I メッセージ，又は KFPH22017-I メッセージが出力された場合	○			○				
	検索性能について期待した効果が得られなかった場合	○			○				
	別表へデータを移行する場合		○	○					
	表の分割条件を変更する場合		○	○					
	アンロードデータを，pdload の入力データファイルに使用したい場合，又は UAP で使用したい場合		○						
	何らかの理由で，表の再編成をアンロードとリロードに分けて実行したい場合		○	○					
インデクス	pdload，又は pdrorg 実行時に-i n を指定してインデクスを作成した場合					○			
	非横分割インデクスがある表に対して，pdload で RD エリア単位にデータロードした場合，又は pdrorg で RD エリア単位に再編成，リロードした場合					○			
	プラグインインデクスの遅延一括作成をする場合					○			
	大量データの追加，削除，更新によって，インデクスの格納ページが使用中空きページとなっている場合，その使用中空きページを解放するとき						○※	○※	
	pdmod でインデクスだけを格納した RD エリアを再初期化した場合，その初期化状態の RD エリアに格納されたインデクスを回復するとき						○		

種別	用途	pdrorg の機能							
		表の再編成	表のアンロード	表へのリロード	ディクショナリ表の再編成	インデクスの一括作成	インデクスの再作成	インデクスの再編成	ディクショナリ表のメンテナンス
	EMPTY 指定の CREATE INDEX を実行した場合，未作成状態となっているインデクスを作成するとき						○		
	プラグインインデクスを再作成する場合						○		
	pdload 実行時に-i x を指定してインデクスを作成した場合						○		
	ディクショナリ表にインデクスを追加する場合								○

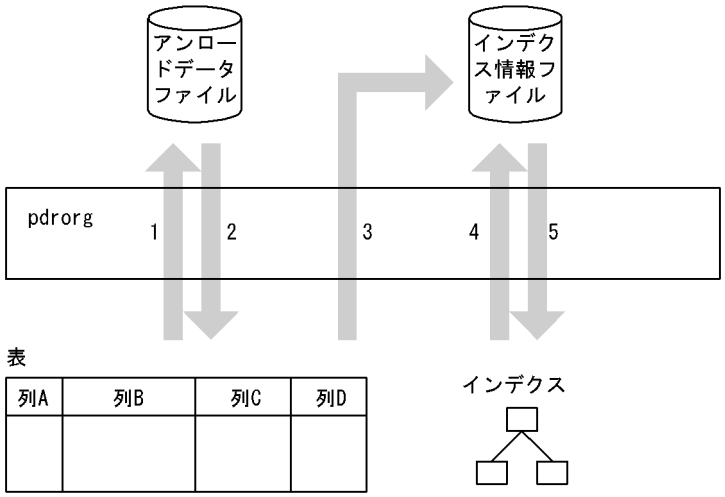
（凡例）

○：用途に該当する pdrorg の機能を示します。

注※

通常時はインデクスの再編成を実行してください。インデクスの再作成では表中のデータを検索しますが，インデクスの再編成では表中のデータを検索しません。したがって，インデクスの再編成の方が処理時間が短くなります。

図 8–1　pdrorg の機能の概要



[説明]

表の再編成：1，及び 2

表のアンロード：1

表へのリロード：2

ディクショナリ表の再編成：1，及び 2

インデクスの一括作成：5
インデクスの再作成：3，及び 5
インデクスの再編成：4，及び 5

8.1.2　表の再編成

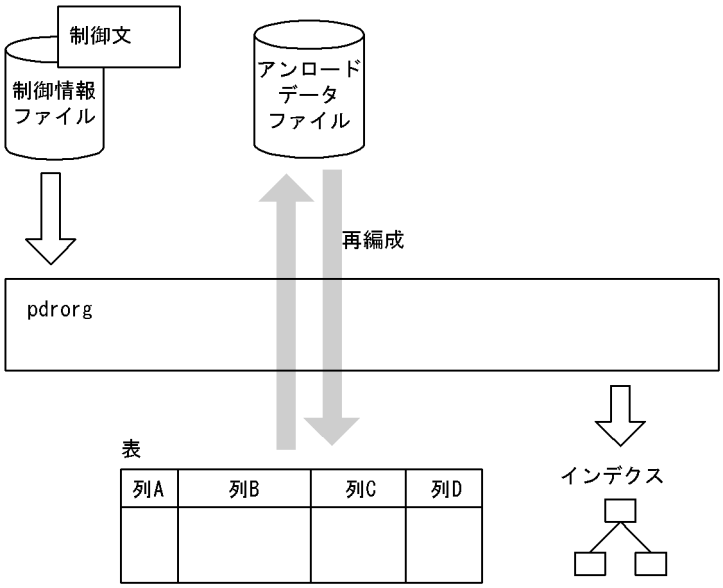
(1)　表の再編成とは

データの追加や削除を繰り返すと，表中の行の配置が乱れて，データのアクセス効率や格納効率が低下します。このような場合，表中のデータをいったんファイルに退避した後，そのファイルのデータを再度表に格納することで，データのアクセス効率，及び格納効率を改善できます。これを**表の再編成**といいます。

表の再編成は，表単位，又は RD エリア単位（横分割表の場合）に実行できます。

表の再編成の概要を次の図に示します。

図 8–2　表の再編成の概要



- **アンロードデータファイル**
  アンロードデータファイルとは，表データをいったん退避しておくファイルのことです。
- **制御情報ファイル**
  制御情報ファイルには，pdrorg の制御文を記述します。
  制御文には，アンロードデータファイルの指定，インデクス情報の指定，及び LOB 列の情報の指定などを記述します。

(2)　LOB 列がある表の場合の再編成

LOB 列がある表の場合，LOB 列構成基表と LOB 列を同時に再編成できます。また，別々に再編成することもできます。

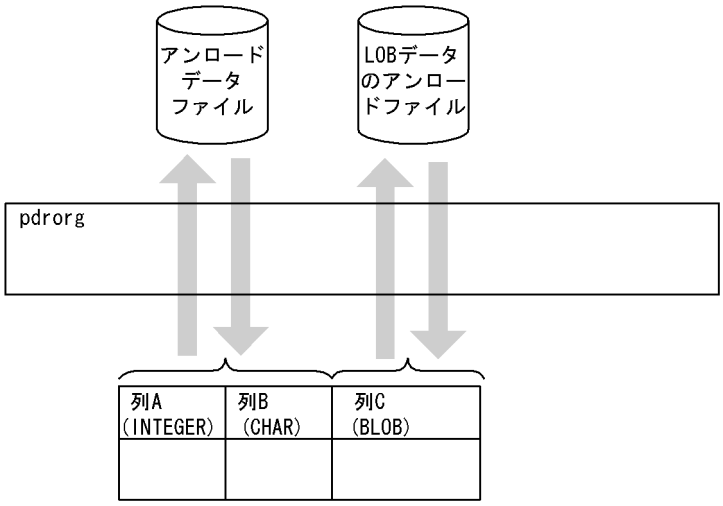
LOB 列構成基表とは，LOB 列がある表の LOB 列を除いた部分のことをいいます。また，**LOB 列**とは，データ型が BLOB の列のことをいいます。

同時に再編成する場合は，-j オプションを指定すると，性能が良くなります。-j オプションを指定しない再編成は，LOB 列構成基表又は LOB 列のどちらかを再編成する場合に採用してください。

プラグインが提供する抽象データ型がユーザ LOB 用 RD エリアに格納されている場合，抽象データ型は LOB 列構成基表と同時に再編成するか，又は再編成対象外にできます。

LOB 列がある表の再編成を次の図に示します。

図 8–3　LOB 列がある表の再編成



[説明]

LOB 列構成基表（列 A，列 B）のデータは，アンロードデータファイルへ退避します。

LOB 列（列 C）のデータは，LOB データのアンロードファイルへ退避します。

-g オプション，及び-j オプションを指定すると，LOB 列構成基表，及び LOB 列のデータをまとめてアンロードデータファイルへ退避することもできます。

(3)　文字集合を定義した表の再編成

表の再編成をする場合に pdrorg が出力するアンロードデータファイルは一時ファイルのため，文字集合の列のデータに対して文字コード変換をしないでアンロードします。リロード時も文字コード変換をしないでデータベースに格納します。ただし，UOC を利用する場合は，文字集合が定義された列データに対して文字コード変換をしてアンロードデータファイルに出力します。

(4)　圧縮表の再編成

圧縮表を再編成する場合，圧縮列のデータは圧縮されたままアンロードし，そのままリロードします。ただし，UOC を利用する場合は，UOC で列データを参照するため，アンロード時に圧縮列の圧縮データを伸張し，リロード時に圧縮してから格納します。

(5)　予備列が定義された FIX 属性の表の再編成

予備列が定義された FIX 属性の表も再編成できます。

(6)　同期点指定の再編成

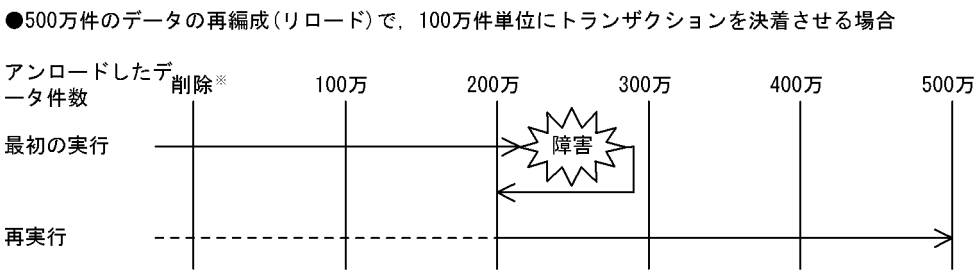
通常，表の再編成をするときは，すべてのデータをリロードした後にトランザクションを決着します。ユティリティが途中で異常終了した場合は，最初の時点までロールバックしてしまいます。このため，再実行時は最初からし直すことになります。

同期点指定の再編成の場合，任意のデータ件数単位にトランザクションを決着できます。これによって，ユティリティが異常終了した場合のロールバック時間や再実行時間を短縮できます。

なお，アンロード処理，インデクスの一括作成，及び LOB 列への再編成処理の場合は，同期点指定の再編成は適用されません。この場合は，すべての処理が完了してからトランザクションが決着します。

同期点指定の再編成をする場合，option 文（job オペランド）を指定します。同期点指定の再編成の概要を次の図に示します。

図 8–4　同期点指定の再編成の概要



注※　既存の表データの削除処理を示します。

[説明]

最初の実行では，障害発生後にロールバックします。200 万件の時点でトランザクションが決着しているため，ロールバックするのは 200 万件の時点までです。

再実行では，既存の表データの削除処理，及び 200 万件までのデータ格納処理をスキップし，それ以降のデータ格納処理をします。

(7)　UOC を利用した再編成

ユーザが作成したプログラム（UOC）を利用して再編成ができます。

UOC を利用した再編成の場合，データベースから検索したデータを直接アンロードデータファイルへ出力するのではなく，データを UOC に渡して編集して出力できます。

例えば，大量のデータを削除する場合など，データベースから検索したデータを UOC でチェックしてアンロードデータファイルに出力するかどうかの判定をして，その結果生成されたアンロードデータファイルからリロードすると，空きページが点在しない（使用中空きページがない）再編成直後の形でデータ削除ができます。また，UOC を使用してアンロードすると，任意のフォーマットでのデータの出力もできます。

8.1.3　表のアンロード

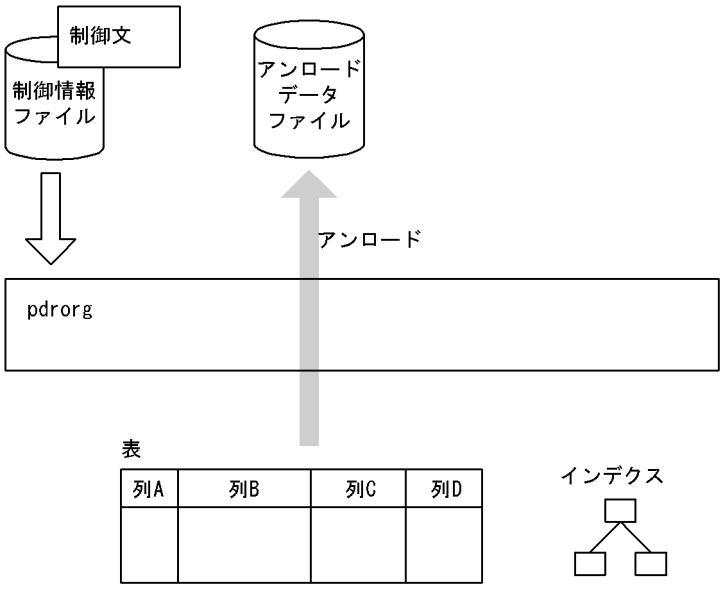
(1)　表のアンロードとは

表の再編成では，表中のデータをいったんファイルに退避した後，そのファイルのデータを再度表に格納します。この表中のデータをファイルに退避する処理のことを**表のアンロード**といいます。

表のアンロードは，表単位，又は RD エリア単位（横分割表の場合）に実行できます。

表のアンロードの概要を次の図に示します。

図 8–5　表のアンロードの概要



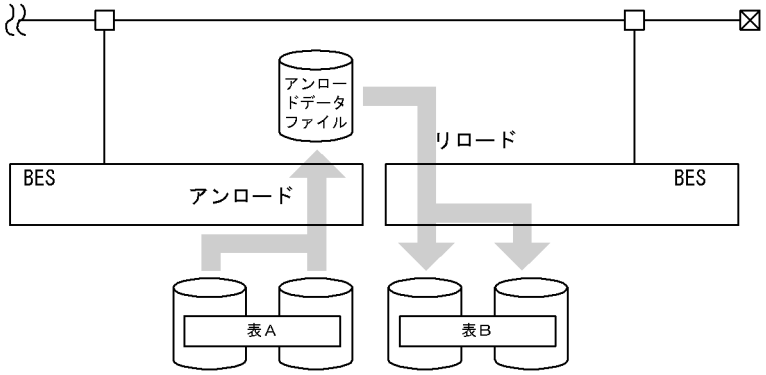
- **アンロードデータファイル**
  アンロードデータファイルとは，表データを退避しておくファイルのことです。
- **制御情報ファイル**
  制御情報ファイルには，pdrorg の制御文を記述します。
  制御文には，アンロードデータファイルの指定，及び LOB 列の情報の指定などを記述します。

(2)　別表へデータを移行する場合

別表へデータを移行する場合，アンロードデータファイルへ表データを退避して，そのアンロードデータファイルから別表へリロードします。このとき，別表の属性（FIX 又は非 FIX）や列定義（列数，データ型など）は同じである必要があります。ただし，定義が異なる場合でも，アンロードデータを pdload の入力データファイル（-W オプション指定）にすると，データ移行ができることがあります。なお，圧縮表を別表にリロードする場合，表の再編成で出力されるアンロードデータファイルは，データが圧縮されているため，使用しないでください。表のアンロードで出力されるファイルを使用してください。

別表へのデータ移行の概要を次の図に示します。

図 8–6　別表へのデータ移行の概要



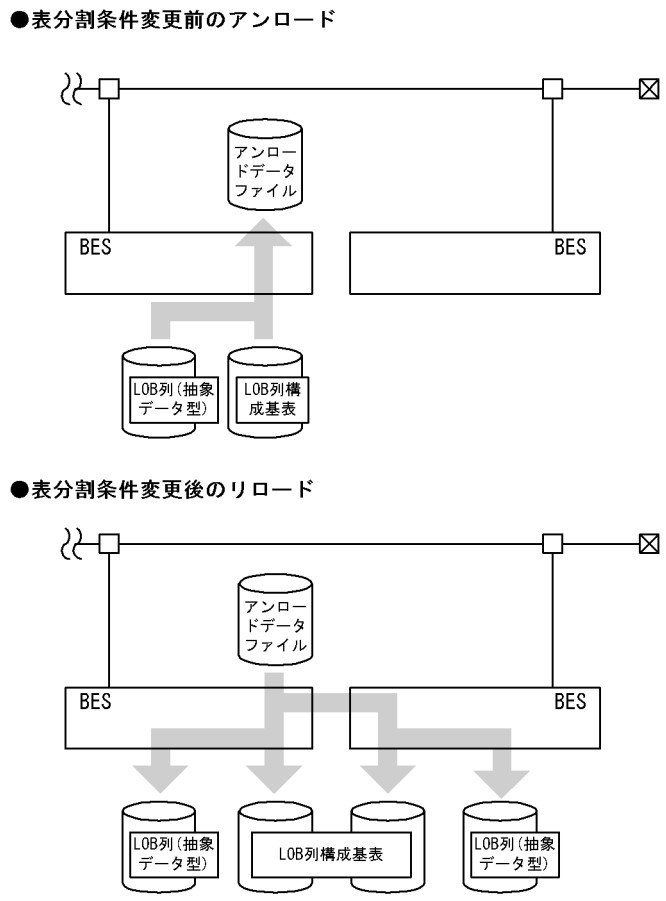
（凡例）BES：バックエンドサーバ

(3)　表の分割条件を変更する場合

表の分割条件を変更する場合，いったんアンロードデータファイルへ表データを退避しておきます。次に，表の分割条件を変更してから，退避しておいたアンロードデータファイルから表へリロードします。

表の分割条件変更の概要を次の図に示します。

図 8–7　表の分割条件変更の概要



（凡例）BES：バックエンドサーバ

(4)　アンロードデータを，pdload の入力データファイルとして使用する場合，又は UAP で使用する場合

アンロードデータを，pdload の入力データファイルとして使用する場合，又は UAP で使用する場合は，-W オプションを指定してアンロードします。

(5)　LOB 列，又はプラグインが提供する抽象データ型列（LOB 属性）がある表の場合のアンロード

LOB 列構成基表と LOB 列は同時にアンロードしてください。LOB 列だけをアンロードすることはできません。

(6)　文字集合を定義した表のアンロード

文字集合を定義した表をアンロードする場合，UOC の利用有無に関係なく，文字コード変換をします。このとき，文字集合の列のデータは，既定文字集合（pdntenv コマンドで指定した文字コード種別）に変換してアンロードデータファイルに出力します。

(7)　圧縮表のアンロード

圧縮表をアンロードする場合，圧縮列のデータを伸張してからアンロードファイルへ出力します。

(8)　予備列が定義された FIX 属性の表のアンロード

予備列が定義された FIX 属性の表もアンロードできます。ただし，pdrorg 実行時に-w オプションを指定して作成した pdload 用アンロードファイルには，予備列は出力されません。

(9)　UOC を利用したアンロード

UOC を利用したアンロードについては，「UOC を利用した再編成」を参照してください。

8.1.4　表へのリロード

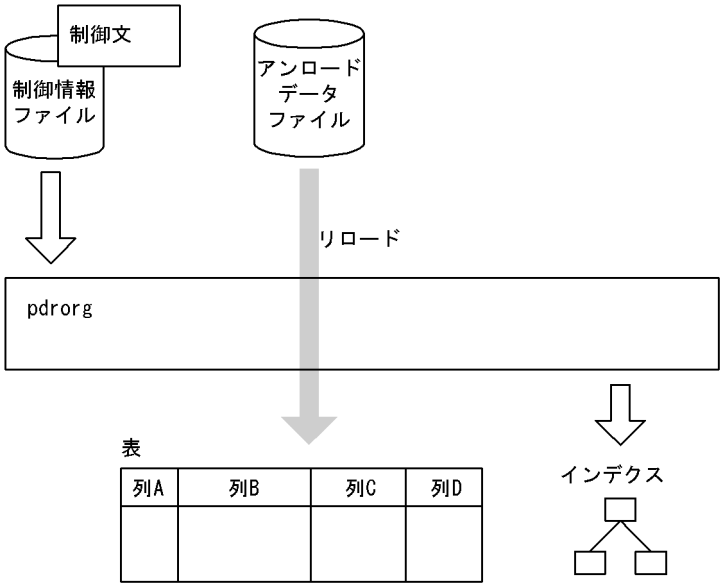
(1)　表へのリロードとは

表の再編成では，表中のデータをいったんファイルに退避した後，そのファイルのデータを再度表に格納します。このファイルのデータを再度表に格納する処理のことを**表へのリロード**といいます。

表へのリロードは，表単位，又は RD エリア単位（横分割表の場合）に実行できます。

表へのリロードの概要を次の図に示します。

図 8–8　表へのリロードの概要



- **アンロードデータファイル**
  アンロードデータファイルとは，表データを退避しておいたファイルのことです。
- **制御情報ファイル**

制御情報ファイルには，pdrorg の制御文を記述します。

制御文には，アンロードデータファイルの指定，インデクス情報の指定，及び LOB 列の情報の指定などを記述します。

(2) 別表へデータを移行する場合

別表へのデータ移行については，「表のアンロード」を参照してください。

(3) 表の分割条件を変更する場合

表の分割条件変更については，「表のアンロード」を参照してください。

(4) LOB 列，又はプラグインが提供する抽象データ型列（LOB 属性）がある表の場合のリロード

表の再編成の場合と同様に，LOB 列構成基表と LOB 列を同時にリロードできます。また，別々にリロードすることもできます。

(5) 文字集合を定義した表のリロード

文字集合を定義した表をリロードする場合，アンロードデータファイル中のデータの文字コードによって，文字コード変換の有無は異なります。アンロードデータファイル中のデータの文字コードが，文字集合の列の文字コードと同じであれば，文字コード変換はしないでそのままデータベースに格納します。異なるときは文字コード変換をしてからデータベースに格納します。

(6) 圧縮表のリロード

圧縮表をリロードする場合，アンロードデータファイルのデータが圧縮されているときは，そのままデータベースに格納します。データが圧縮されていないときは，データを圧縮してからデータベースに格納します。

なお，表の再編成時にエラーが発生し，圧縮されたデータを含むアンロードデータファイル（pdrorg -k rorg でアンロードされたファイル）をリロードする場合，アンロード元とリロード先の表の圧縮指定（圧縮指定の有無，及び圧縮分割サイズの指定値）が同じである必要があります。1 列でも圧縮指定が異なる列がある場合，pdrorg は KFPL15327-E メッセージを出力してエラー終了します。

(7) 予備列が定義された FIX 属性の表のリロード

予備列が定義された FIX 属性の表をリロードする場合，アンロード元とリロード先の表の予備列の有無が異なるときは，pdrorg は KFPL25343-E メッセージを出力して異常終了します。アンロード元とリロード先の表の予備列の有無が同じ，かつ実行可否の条件を満たしているときは，リロードできます。実行可否の条件については，「アンロード元の表とリロード先の表の表定義が異なる場合のリロード実行可否」を参照してください。

(8)　同期点指定のリロード

同期点指定のリロードについては，「表の再編成」の同期点指定の再編成を参照してください。

8.1.5　ディクショナリ表の再編成

ディクショナリ表を再編成する場合，データディクショナリ用 RD エリア（データディクショナリ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアを定義している場合はこれらも含む）をコマンド閉塞状態にすることで，ユーザが定義した表を再編成するのと同様に実行できます。

ディクショナリ表の再編成は，特定のディクショナリ表だけ，又はすべてのディクショナリ表を再編成できます。ディクショナリ表を再編成する場合のアンロード・リロード処理の順序は「ディクショナリ表の場合」の表の項番の順になります。再編成のリロード処理中にエラーとなった場合は，リロード（-k reld）の再実行で再編成を再開できます。ただし，リロード中にデータディクショナリ用 RD エリア又はデータディクショナリ LOB 用 RD エリアに容量不足が発生した場合，RD エリアをバックアップから回復するケースがあります。再編成の手順については，マニュアル「HiRDB システム運用ガイド」の例題（ディクショナリ表の再編成）の「注意事項」を参照してください。

LOB 列があるディクショナリ表の場合，制御情報ファイルの unload 文及び lobunld 文で再編成対象を指定できます。LOB 列構成基表と LOB 列を同時に再編成する場合は，制御情報ファイルに unload 文と lobunld 文を両方とも指定してください。制御情報ファイルの unload 文及び lobunld 文の指定の組み合わせとそれによる再編成対象を次に示します。

制御情報ファイルの指定		再編成対象	
unload 文指定	lobunld 文指定	LOB 列構成基表	LOB 列
あり	あり	○	○
	なし	○	－
なし	あり	－	○
	なし	pdrorg は実行できません。	

（凡例）○：再編成します。　－：再編成しません。

なお，LOB 列があるディクショナリ表は次の表です。

- SQL_ROUTINES
- SQL_REGISTRY_KEY（システムが使用する表）

8.1.6　インデクスの一括作成

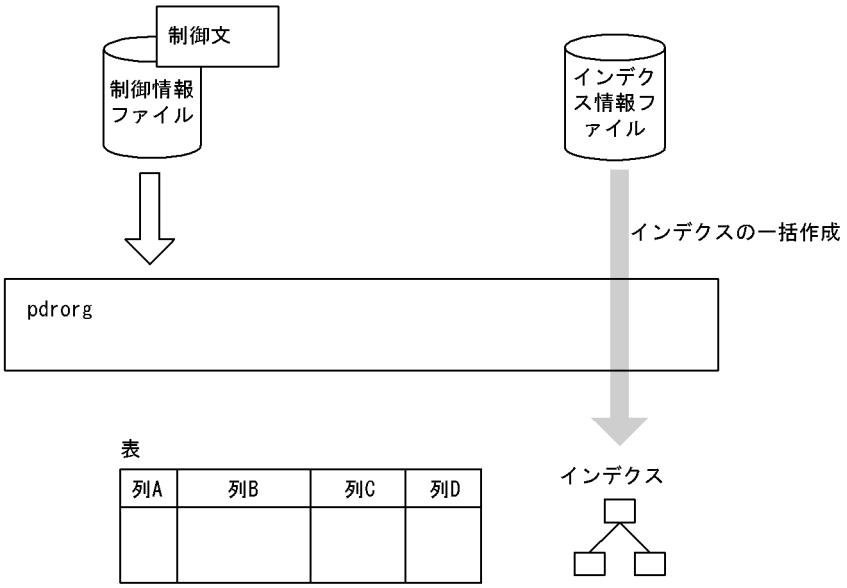
pdload，又は pdrorg が作成したインデクス情報ファイルから，インデクスを作成します。また，プラグインインデクスの遅延一括作成機能で出力されたインデクス情報ファイルからインデクスを作成します。これを**インデクスの一括作成**といいます。

なお，インデクスの再編成でエラーとなった場合（-l a，又は-l p の場合）にも，インデクスの一括作成を実行します。

インデクスの一括作成は，インデクス単位，又はインデクス格納 RD エリア単位に実行できます。

インデクスの一括作成の概要を次の図に示します。

図 8–9　インデクスの一括作成の概要



- **インデクス情報ファイル**
  インデクス情報ファイルとは，pdload，又は pdrorg 実行時にインデクスに関する情報を出力したファイルのことです。また，プラグインインデクスの遅延一括作成機能を使用している場合は，UAP などで更新したインデクス情報が出力されています。
- **制御情報ファイル**
  制御情報ファイルには，pdrorg の制御文を記述します。
  制御文には，インデクス情報の指定などを記述します。

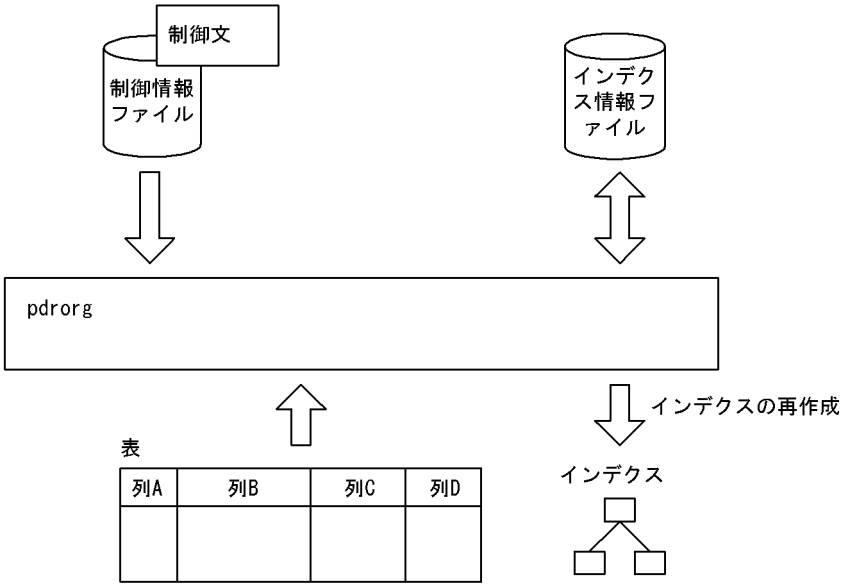
8.1.7　インデクスの再作成

表中のデータを検索して，インデクス情報を生成し，その情報を基にインデクスを作成します。これを**インデクスの再作成**といいます。

インデクスの再作成は，インデクス単位，又はインデクス格納 RD エリア単位に実行できます。

インデクスの再作成の概要を次の図に示します。

図 8–10　インデクスの再作成の概要



- **インデクス情報ファイル**
  インデクス情報ファイルとは，表中のデータを検索して生成したインデクス情報が出力されるファイルのことです。
- **制御情報ファイル**
  制御情報ファイルには，pdrorg の制御文を記述します。
  制御文には，インデクス情報の指定などを記述します。

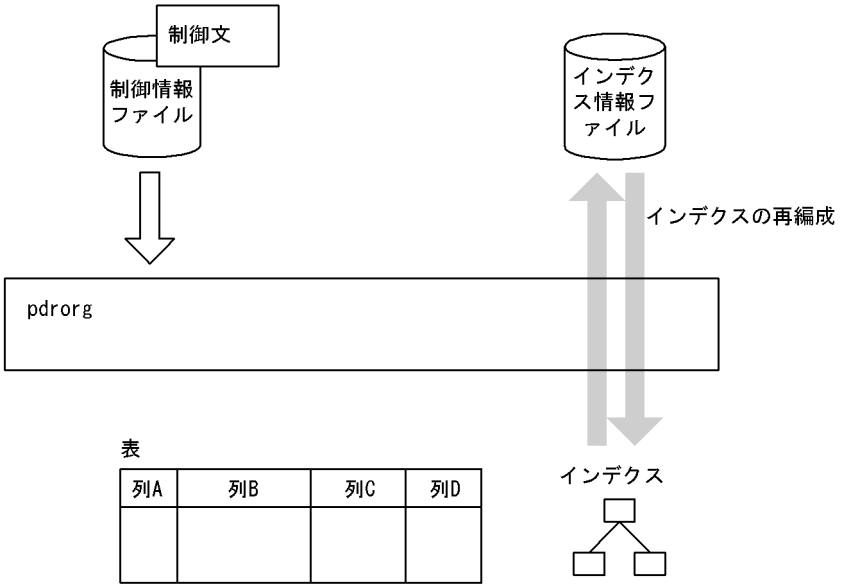
8.1.8　インデクスの再編成

インデクスのキー情報を検索してインデクス情報ファイルを作成し，その情報を基にインデクスを再配置します。これを**インデクスの再編成**といいます。

インデクスの再編成は，インデクス単位，又はインデクス格納 RD エリア単位に実行できます。

インデクスの再編成の概要を次の図に示します。

図 8–11　インデクスの再編成の概要



- **インデクス情報ファイル**

  インデクス情報ファイルとは，インデクスのキー情報を検索して作成したインデクス情報が出力されるファイルのことです。

- **制御情報ファイル**

  制御情報ファイルには，pdrorg の制御文を記述します。

  制御文には，インデクス情報の指定などを記述します。

8.1.9　ディクショナリ表のメンテナンス

ディクショナリ表のメンテナンスを行います。HiRDB をバージョンアップしただけではディクショナリ表はメンテナンスされないため，HiRDB のバージョンアップ実行後に任意で実行します。ディクショナリ表のメンテナンスでできることを次に示します。

ディクショナリ表へのインデクスの追加

すでに登録しているディクショナリ表のデータにインデクスを定義し，インデクスデータを作成します。インデクスを追加できるディクショナリ表及びインデクス種別については，マニュアル「HiRDB システム導入・設計ガイド」の「HiRDB のシステム構築の概要」の「HiRDB のバージョンアップ」の「バージョンアップ時の留意事項」を参照してください。

8.1.10　pdrorg 実行時のログ取得方式

pdrorg を実行する場合，ログ取得方式は更新前ログ取得モード（-l p）にすることをお勧めします。なお，この更新前ログ取得モードは省略時仮定値となっています。

更新前ログ取得モードの場合，更新後のデータベース更新ログを取得しないため，その分の処理時間が短縮されます。したがって，ログ取得モードを適用したときに比べてユティリティの実行時間を短縮できます。

ただし，ディクショナリ表のメンテナンスを実行する場合は，常にログ取得モード（-l a）を適用します。

8.1.11　処理対象の表，インデクスが格納されている RD エリア

処理対象となる表及びインデクスを UAP がアクセスして不当に待ち状態にならないように，処理対象となる表及びインデクスが格納されている RD エリアは，pdhold コマンドで閉塞しておくことをお勧めします。

8.1.12　抽象データ型列がある表に対する pdrorg の実行可否

抽象データ型がある表に対して実行できる，pdrorg の機能について説明します。抽象データ型がある表に対する機能の実行可否を次の表に示します。

表 8–2　抽象データ型がある表に対する機能の実行可否

<table>
<tr><th>機能</th><th colspan="3">条件</th><th>実行可否</th></tr>
<tr><td rowspan="4">表の再編成
(-k rorg，unld，及び reld)</td><td rowspan="4">プラグインが提供する抽象データ型</td><td>LOB 属性なし</td><td></td><td>○</td></tr>
<tr><td rowspan="3">LOB 属性あり</td><td>LOB 列構成基表だけ</td><td>○</td></tr>
<tr><td>同時（-j オプション指定）</td><td>○※1</td></tr>
<tr><td>LOB 列だけ</td><td>×</td></tr>
<tr><td colspan="3">ユーザが定義した抽象データ型</td><td>×※2</td></tr>
<tr><td rowspan="2">インデクスの作成
(-k ixmk，及び ixrc)</td><td colspan="3">プラグインが提供するインデクス型から作成したプラグインインデクス</td><td>○</td></tr>
<tr><td colspan="3">ユーザが定義した抽象データ型のインデクス</td><td>×</td></tr>
<tr><td>インデクスの再編成
(-k ixor)</td><td colspan="3">–</td><td>×</td></tr>
<tr><td rowspan="2">• 別表へのリロード
(tblname 文)
• 表の分割条件変更</td><td colspan="3">プラグインが提供する抽象データ型</td><td>○※3</td></tr>
<tr><td colspan="3">ユーザが定義した抽象データ型</td><td>×</td></tr>
<tr><td rowspan="2">pdload 用の入力データファイル作成
(-W オプション)</td><td colspan="3">プラグインが提供する抽象データ型</td><td>○※4</td></tr>
<tr><td colspan="3">ユーザが定義した抽象データ型</td><td>×</td></tr>
</table>

（凡例）

○：実行できます。

×：実行できません。

–：条件がないことを示します。

注※1

アンロード機能を持つプラグインが提供した抽象データ型の場合だけ実行できます。

注※2

プラグインが実装していない抽象データ型がある表は pdrorg で再編成できません。抽象データ型がある表に対して KFPH00212-I メッセージが出力された場合，次のどちらかで対処してください。

- UAP でデータベースの内容をファイル出力し，PURGE TABLE を実行して表のデータを削除する。その後，ファイルを入力して UAP でデータベースを再作成する。
- pdmod で該当する RD エリアを拡張する。

注※3

コンストラクタパラメタ逆生成関数を指定しないと実行できないことがあります。指定する関数名については，各プラグインのマニュアルを参照してください。

注※4

コンストラクタパラメタ逆生成関数を指定した場合だけ実行できます。指定する関数名については，各プラグインのマニュアルを参照してください。

8.1.13 実行者

pdrorg の各機能を実行する場合に必要となる権限を次に示します。

pdrorg の機能	対象となる表		
	ディクショナリ表	ユーザが定義した表	監査証跡表
表の再編成	DBA 権限	SELECT 権限，INSERT 権限，及び DELETE 権限，又は DBA 権限	SELECT 権限，INSERT 権限，及び DELETE 権限
表のアンロード	DBA 権限	SELECT 権限，又は DBA 権限	SELECT 権限
表へのリロード	－	INSERT 権限，DELETE 権限，又は DBA 権限	INSERT 権限，及び DELETE 権限
スキーマ単位の再編成，アンロード，及びリロード	－	DBA 権限，又はスキーマの所有者	監査権限
インデクスの一括作成	－	権限不要	権限不要
インデクスの再作成	－	権限不要	権限不要
インデクスの再編成	－	権限不要	権限不要
ディクショナリ表のメンテナンス	DBA 権限	－	－

（凡例）

－：実行できません。

8.2　表の再編成

8.2.1　使用例

データベース再編成ユティリティの使用例の一覧（表の再編成）を次に示します。

使用例	内容	区分
例 1	横分割表の表単位の再編成	S
例 2	横分割表の RD エリア単位の再編成	S
例 3	LOB 列がある表の再編成 • LOB 列構成基表と LOB 列を同時に再編成します。	S
例 4	抽象データ型（SGMLTEXT 型）がある表の再編成 • LOB 列構成基表だけ再編成します。	S
例 5	横分割表の表単位の再編成	P
例 6	LOB 列がある表の再編成 • LOB 列構成基表と LOB 列を同時に再編成します。	P
例 7	スキーマ単位の再編成	P

（凡例）

S：HiRDB/シングルサーバ

P：HiRDB/パラレルサーバ

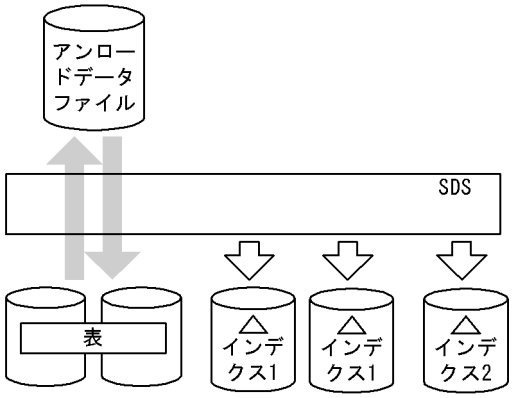
(1)　横分割表の表単位の再編成

例 1

横分割表（TABLE1）を表単位に再編成します。
表（TABLE1），及びインデクスは次のように定義されているものとします。

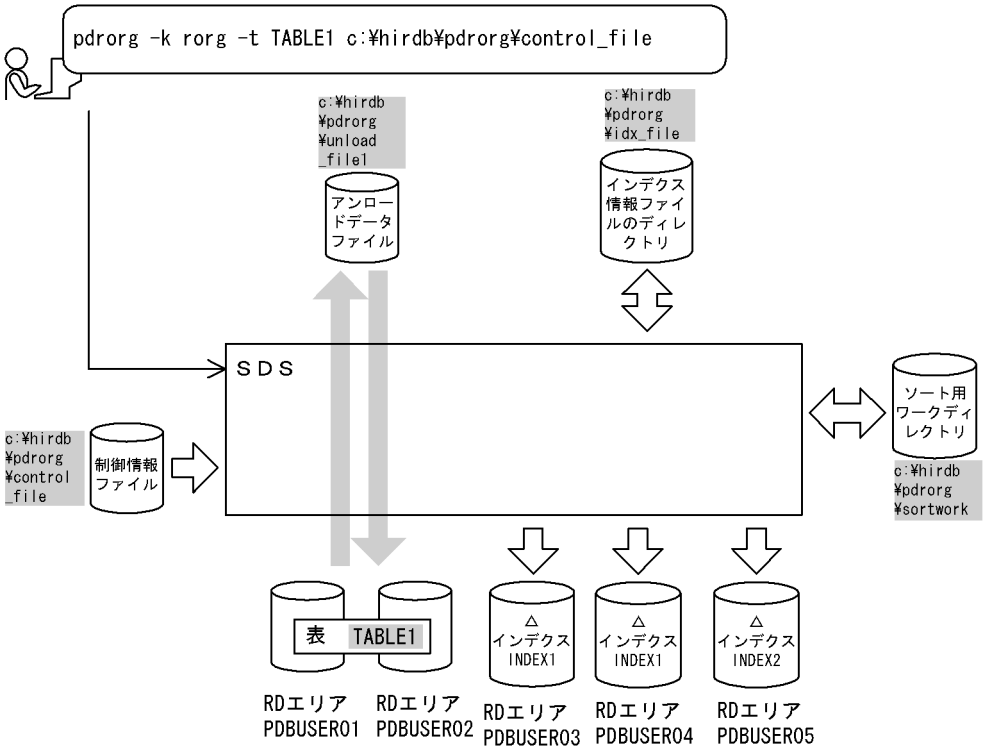
```
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
         IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER04))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN (PDBUSER05)
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース再編成ユティリティでの指定値

ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

横分割表（TABLE1）を表単位に再編成します。

-k rorg：再編成をするときの指定

-t TABLE1：再編成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload c:¥hirdb¥pdrorg¥unload_file1          1
idxwork c:¥hirdb¥pdrorg¥idx_file             2
sort c:¥hirdb¥pdrorg¥sortwork                3
```

[説明]

1. アンロードデータファイルの指定

   c:¥hirdb¥pdrorg¥unload_file1：アンロードデータファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   c:¥hirdb¥pdrorg¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

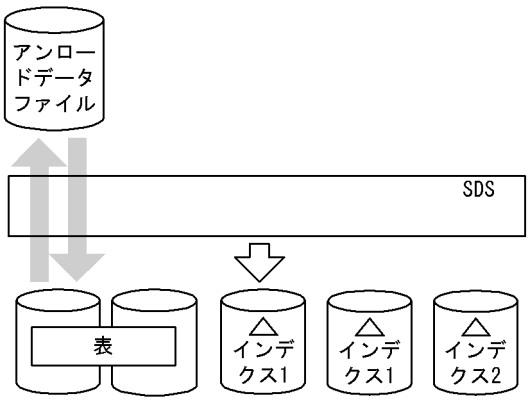
   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(2) 横分割表の RD エリア単位の再編成

例２

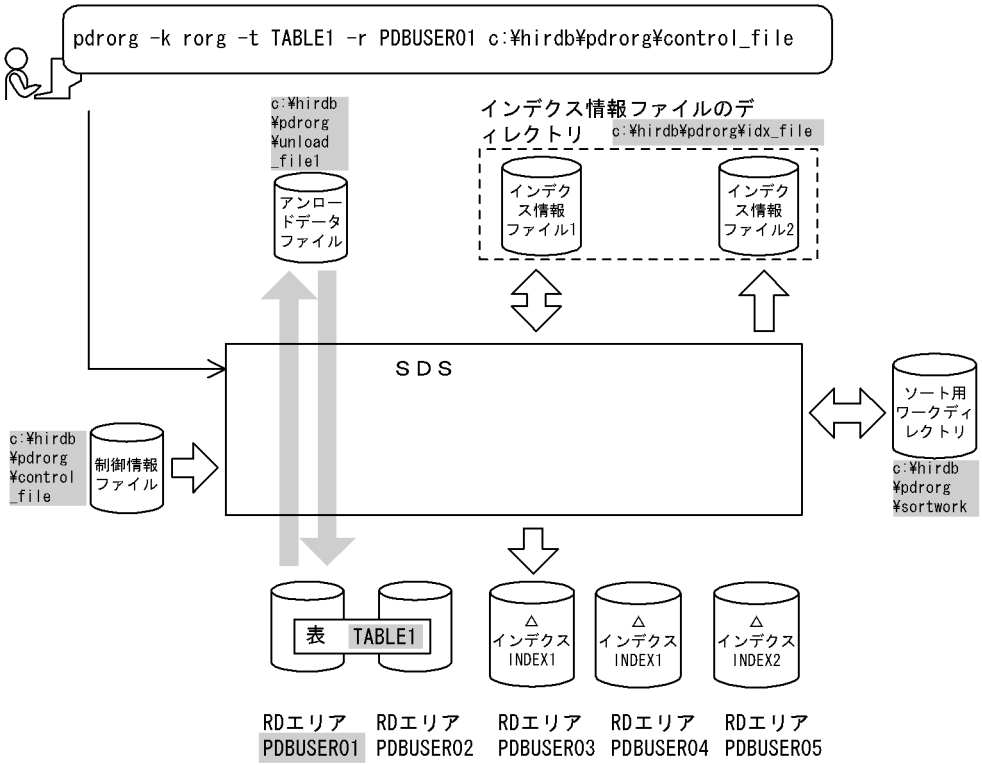
```
横分割表（TABLE1）をRDエリア単位に再編成します。
表（TABLE1），及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
         IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER04))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN (PDBUSER05)
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



(a) TABLE1（RD エリア PDBUSER01）の再編成

[実行するコマンドの説明]

横分割表（TABLE1）を RD エリア単位に再編成します。

-k rorg：再編成をするときの指定

-t TABLE1：再編成対象の表の名称

-r PDBUSER01：再編成対象の RD エリアの名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload c:¥hirdb¥pdrorg¥unload_file1          1
idxwork c:¥hirdb¥pdrorg¥idx_file             2
sort c:¥hirdb¥pdrorg¥sortwork                3
```

[説明]

1. アンロードデータファイルの指定

   c:¥hirdb¥pdrorg¥unload_file1：アンロードデータファイルの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   c:¥hirdb¥pdrorg¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(b)　INDEX2 の作成

INDEX2 はこのままでは使用できません。

INDEX2 を使用する場合は，RD エリア PDBUSER02 の再編成をして，PDBUSER02 のインデクス情報を出力します。このインデクス情報ファイルと，既に出力されているインデクス情報ファイル 2 を入力にして，pdrorg でインデクスの一括作成（-k ixmk）を実行してください。

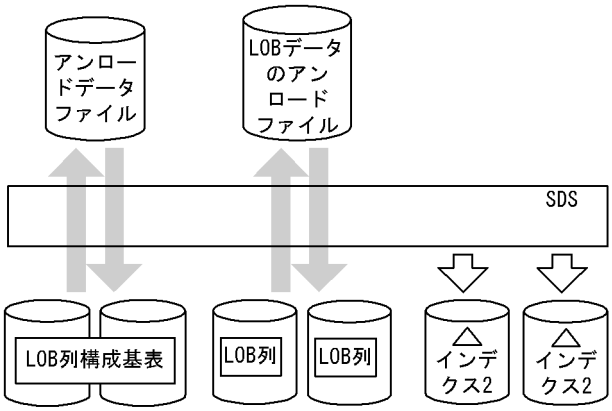
詳細は，「インデクスの一括作成」を参照してください。

(3)　LOB 列がある表の再編成

例 3

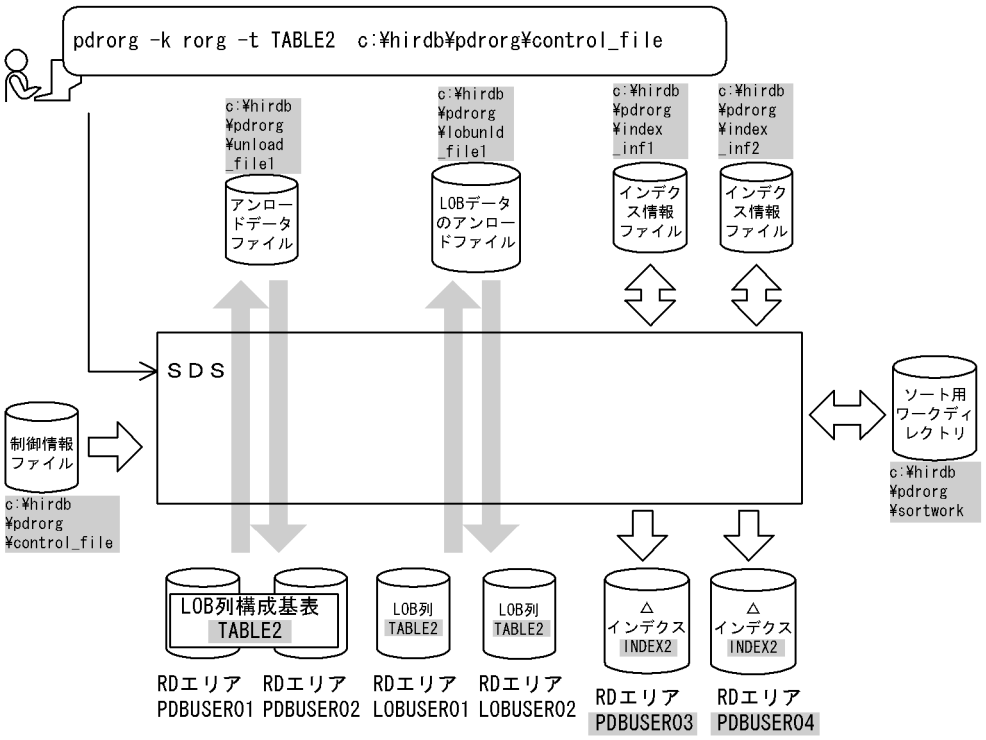
```
LOB列がある表（TABLE2）に対して，LOB列構成基表とLOB列を同時に再編成します。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT NOT NULL,C2 BLOB IN ((LOBUSER01),
                (LOBUSER02))) IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義
  CREATE INDEX INDEX2 ON TABLE2(C1) IN ((PDBUSER03),(PDBUSER04))
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

LOB 列がある表（TABLE2）を，表単位（LOB 列構成基表と LOB 列を同時）に再編成します。

-k rorg：再編成をするときの指定

-t TABLE2：再編成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload c:¥hirdb¥pdrorg¥unload_file1                          1
index INDEX2 PDBUSER03 c:¥hirdb¥pdrorg¥index_inf1            2
index INDEX2 PDBUSER04 c:¥hirdb¥pdrorg¥index_inf2            2
sort c:¥hirdb¥pdrorg¥sortwork                                3
lobunld c:¥hirdb¥pdrorg¥lobunld_file1                        4
```

[説明]

1. アンロードデータファイルの指定

   c:¥hirdb¥pdrorg¥unload_file1：アンロードデータファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   PDBUSER03，PDBUSER04：インデクスを格納する RD エリアの名称

   c:¥hirdb¥pdrorg¥index_inf1，c:¥hirdb¥pdrorg¥index_inf2：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. LOB データのアンロードファイルの指定

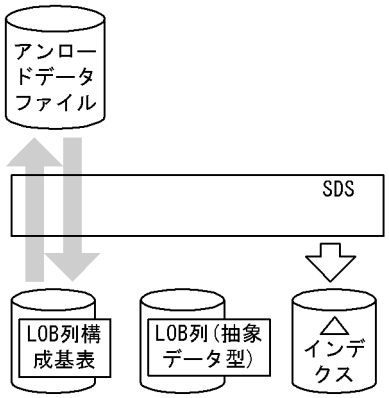
c:¥hirdb¥pdrorg¥lobunld_file1：LOB データのアンロードファイルの名称

(4) 抽象データ型がある表の再編成

例 4

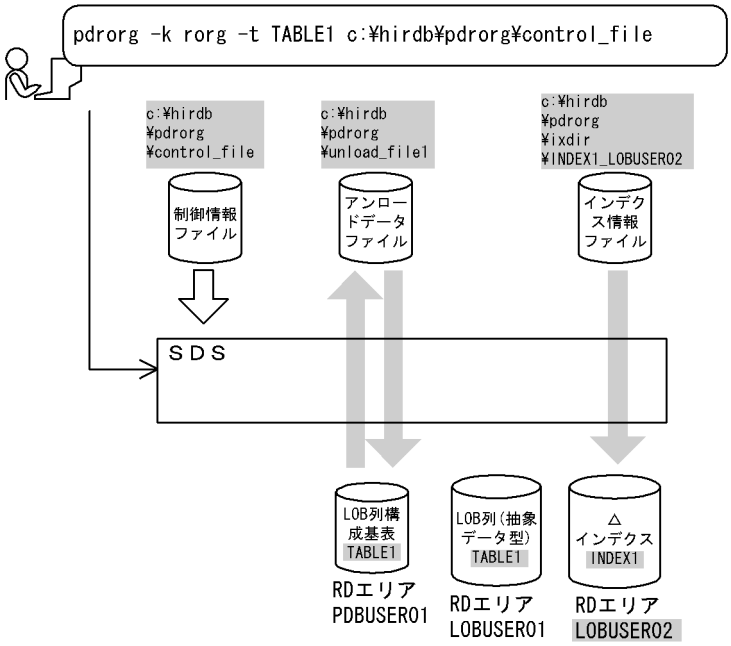
```
抽象データ型（SGMLTEXT）の列を持つ表（TABLE1）の再編成をします。抽象データ型（SGMLTEXT）
はHiRDB Text Search Plug-inで提供されています。
表，及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT,C2 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER01)
            PLUGIN'<DTD>sgml.dtd</DTD>' ) IN PDBUSER01
・プラグインインデクスの定義
  CREATE INDEX INDEX1 USING TYPE NGRAM ON TABLE1(C2) IN LOBUSER02
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

表（TABLE1）を再編成します。

-k rorg：再編成をするときの指定

-t TABLE1：再編成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload c:¥hirdb¥pdrorg¥unload_file1                                    1
index INDEX1 LOBUSER02 c:¥hirdb¥pdrorg¥ixdir¥INDEX1_LOBUSER02         2
```

[説明]

1. アンロードデータファイルの指定

   c:¥hirdb¥pdrorg¥unload_file1：アンロードデータファイルの名称

2. インデクス情報ファイルの指定

   INDEX1：インデクスの一括作成をする，プラグインインデクスのインデクス識別子

   LOBUSER02：インデクスの一括作成をする，プラグインインデクスのインデクス格納 RD エリアの名称

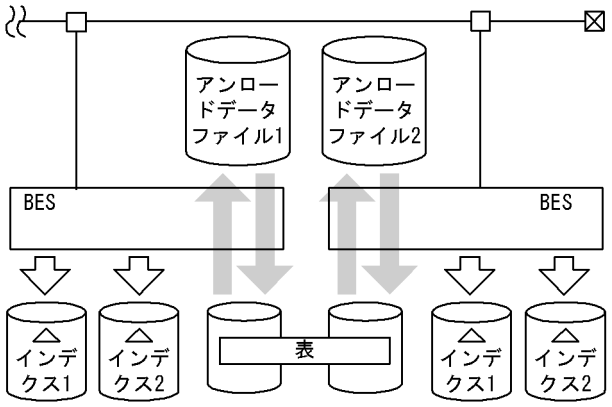
   c:¥hirdb¥pdrorg¥ixdir¥INDEX1_LOBUSER02：インデクス情報ファイルの名称

(5)　横分割表の表単位の再編成

例５

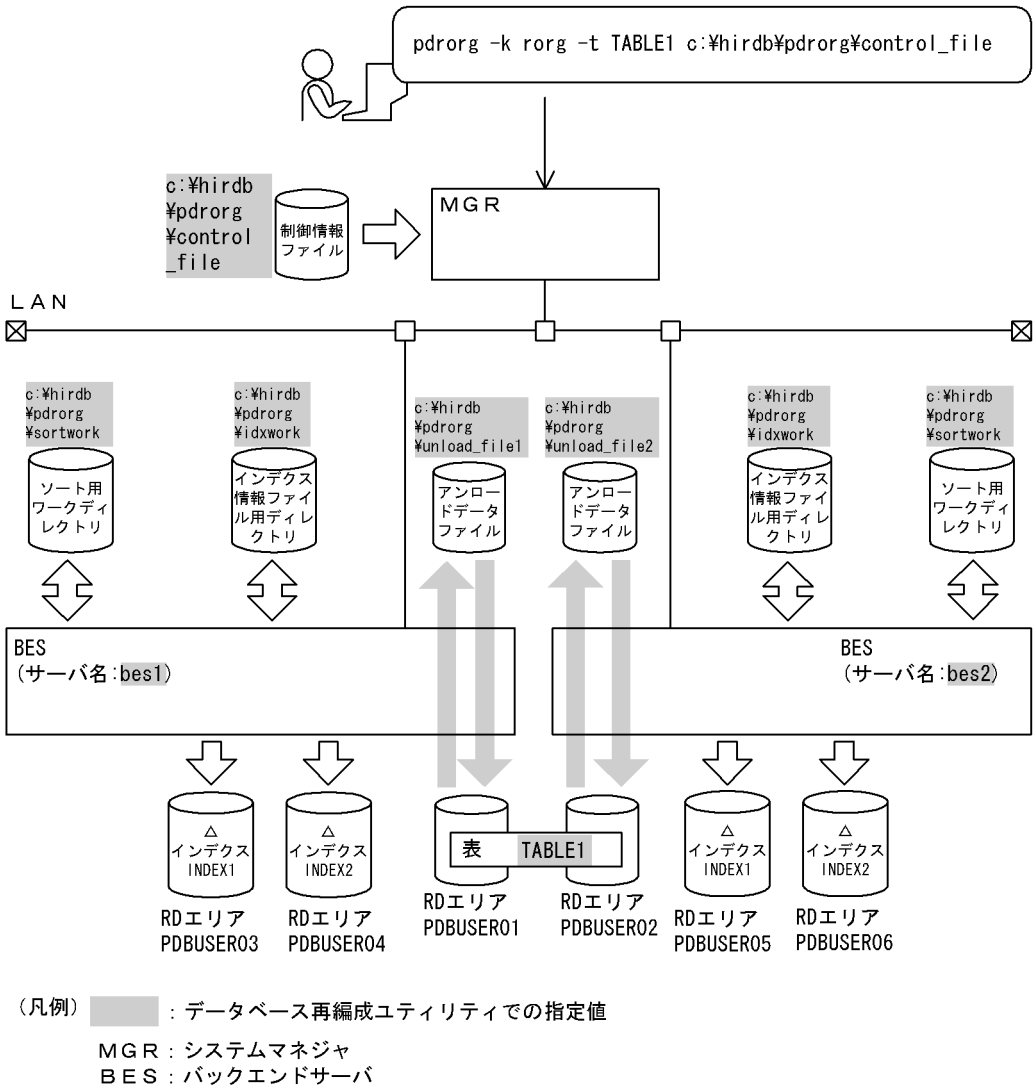
横分割表（TABLE1）を表単位に再編成します。
表，及びインデクスは次のように定義されているものとします。
・表の定義
```
CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
             IN ((PDBUSER01) C1 > 10,(PDBUSER02))
```
・インデクスの定義（分割キーインデクス）
```
CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
```
・インデクスの定義（非分割キーインデクス）
```
CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

横分割表（TABLE1）を表単位に再編成します。

-k rorg：再編成をするときの指定

-t TABLE1：再編成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file1          1
idxwork bes1 c:¥hirdb¥pdrorg¥idxwork              2
sort bes1 c:¥hirdb¥pdrorg¥sortwork                3
unload bes2:c:¥hirdb¥pdrorg¥unload_file2          1
idxwork bes2 c:¥hirdb¥pdrorg¥idxwork              2
sort bes2 c:¥hirdb¥pdrorg¥sortwork                3
```

[説明]

1. アンロードデータファイルの指定

bes1，bes2：アンロードデータファイルがあるサーバの名称

c:¥hirdb¥pdrorg¥unload_file1，c:¥hirdb¥pdrorg¥unload_file2：アンロードデータファイルの名称

2. インデクス情報を出力するインデクス情報ファイル用ディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥idxwork：インデクス情報ファイル用ディレクトリの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

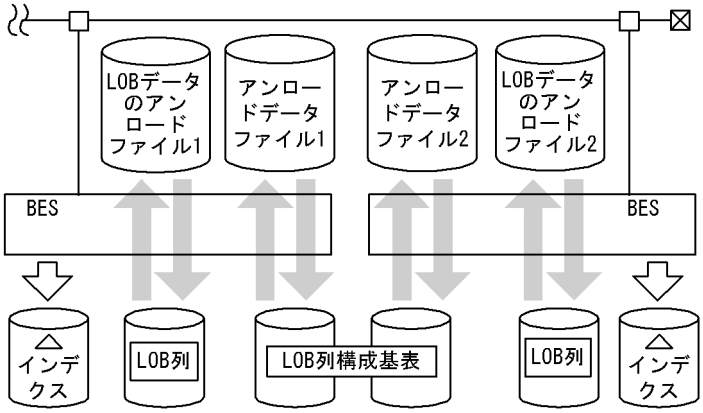
   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(6) LOB 列がある表の再編成

例 6

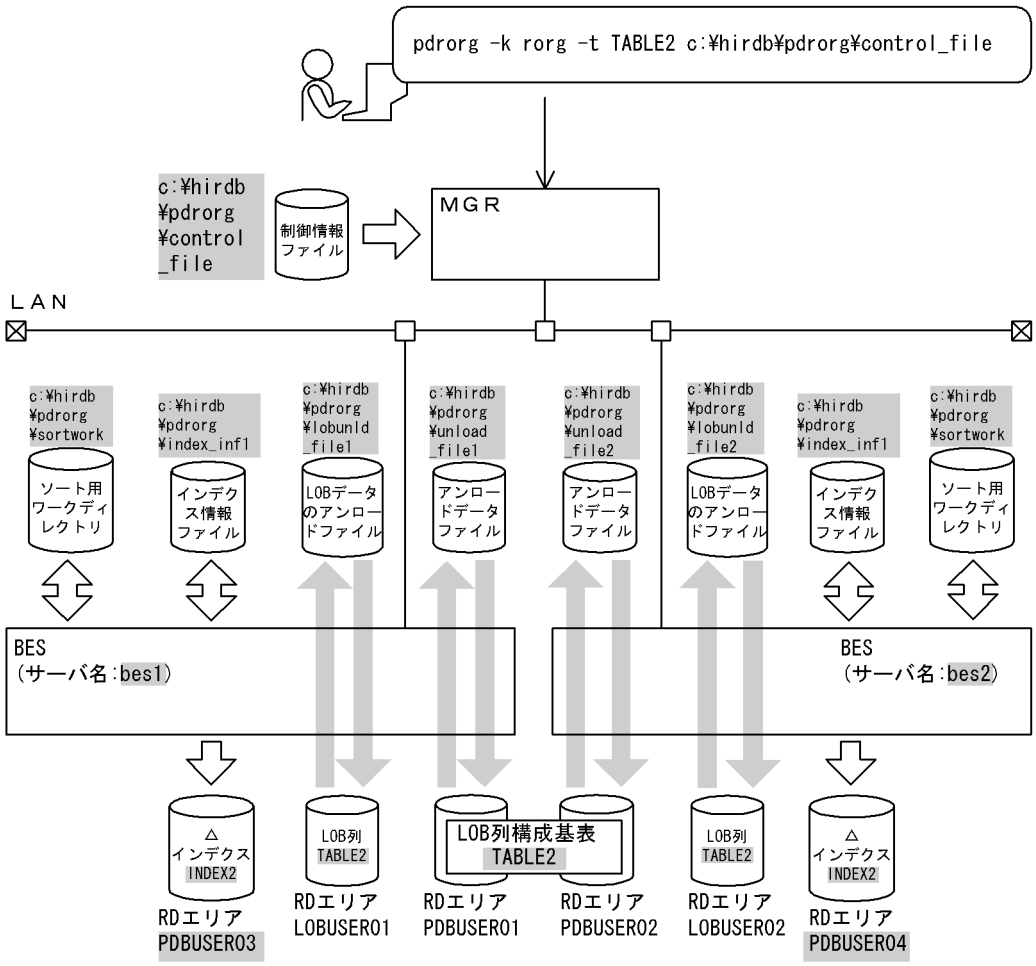
```
LOB列がある表（TABLE2）に対して，LOB列構成基表とLOB列を同時に再編成します。
次の表，及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE2(C1 INT NOT NULL,C2 BLOB IN ((LOBUSER01),
              (LOBUSER02))) IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義
  CREATE INDEX INDEX2 ON TABLE2(C1) IN ((PDBUSER03),(PDBUSER04))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

LOB 列がある表（TABLE2）を，表単位（LOB 列構成基表と LOB 列を同時）に再編成します。

-k rorg：再編成をするときの指定

-t TABLE2：再編成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file1                1
unload bes2:c:¥hirdb¥pdrorg¥unload_file2                1
index INDEX2 PDBUSER03 c:¥hirdb¥pdrorg¥index_inf1       2
index INDEX2 PDBUSER04 c:¥hirdb¥pdrorg¥index_inf2       2
sort bes1 c:¥hirdb¥pdrorg¥sortwork                      3
sort bes2 c:¥hirdb¥pdrorg¥sortwork                      3
lobunld bes1:c:¥hirdb¥pdrorg¥lobunld_file1              4
lobunld bes2:c:¥hirdb¥pdrorg¥lobunld_file2              4
```

［説明］

1. アンロードデータファイルの指定

   bes1，bes2：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file1，c:¥hirdb¥pdrorg¥unload_file2：アンロードデータファイルの名称

2. インデクス情報を出力するインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   PDBUSER03，PDBUSER04：インデクスを格納する RD エリアの名称

   c:¥hirdb¥pdrorg¥index_inf1，c:¥hirdb¥pdrorg¥index_inf2：インデクス情報ファイルの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. LOB データのアンロードファイルの指定

   bes1，bes2：LOB データのアンロードファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥lobunld_file1，c:¥hirdb¥pdrorg¥lobunld_file2：LOB データのアンロードファイルの名称

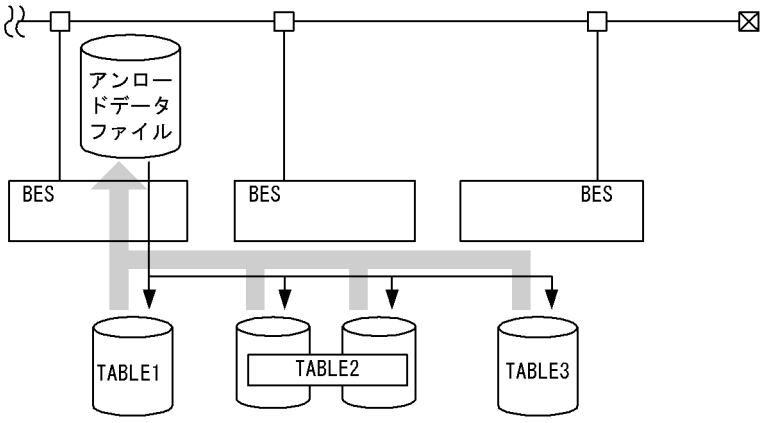
(7)　スキーマ単位の再編成

例 7

```
スキーマ（USER01）が所有するすべての表を再編成します。
USER01は次の表を定義しているものとします。

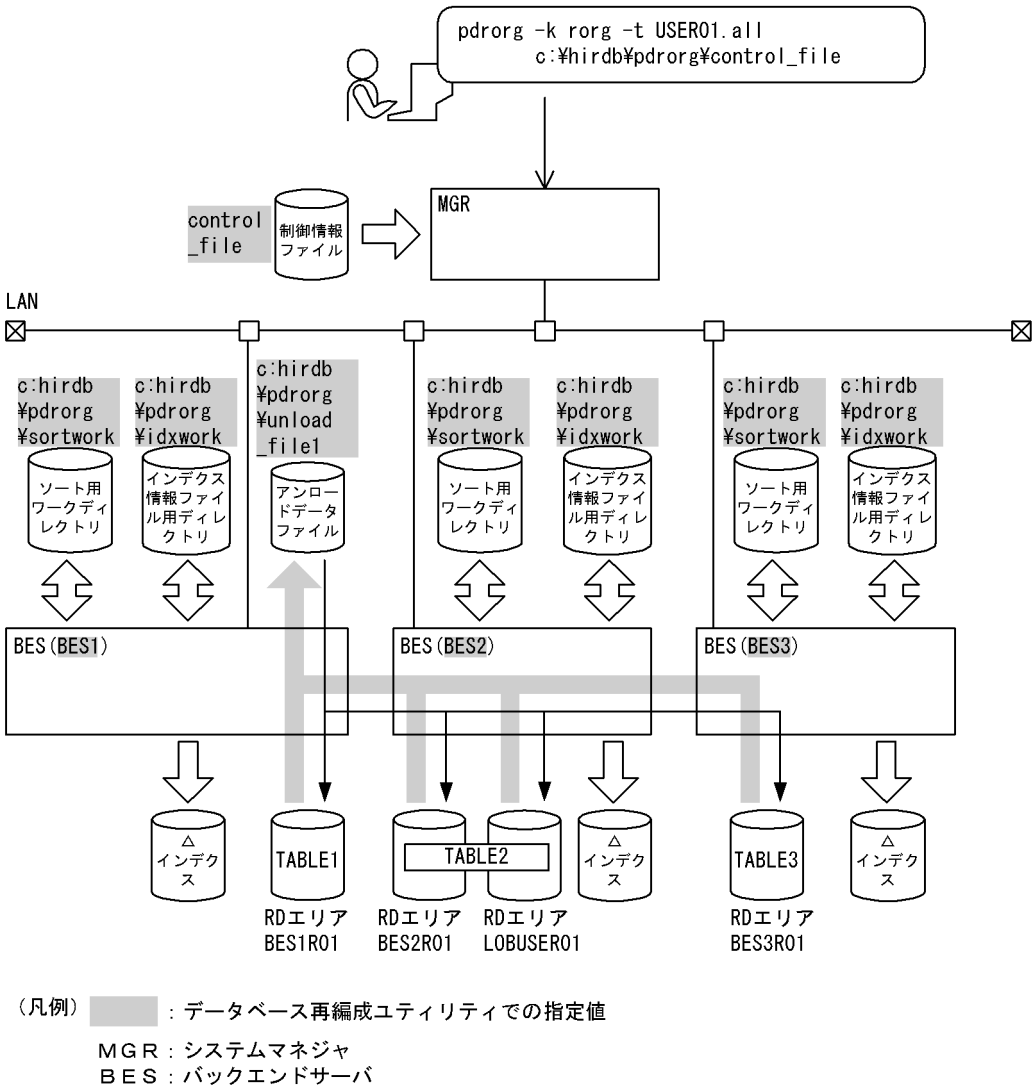
  CREATE TABLE TABLE1(C1 INT) IN (BES1R01)
  CREATE TABLE TABLE2(C1 BLOB IN LOBUSER01) IN(BES2R01)
  CREATE TABLE TABLE3(C1 FREEWORD) IN(BES3R01)
```

［概要］



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

スキーマ（USER01）が所有するすべての表を再編成します。なお，HiRDB/パラレルサーバでスキーマ単位の再編成をする場合，-g 及び-j オプションの指定が仮定されます。

-k rorg：再編成をするときの指定

-t USER01.all：スキーマ（USER01）が所有するすべての表を対象とする場合の指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file1          1
idxwork bes1 c:¥hirdb¥pdrorg¥idxwork              2
sort bes1 c:¥hirdb¥pdrorg¥sortwork                3
idxwork bes2 c:¥hirdb¥pdrorg¥idxwork              2
sort bes2 c:¥hirdb¥pdrorg¥sortwork                3
idxwork bes3 c:¥hirdb¥pdrorg¥idxwork              2
sort bes3 c:¥hirdb¥pdrorg¥sortwork                3
```

[説明]

1. アンロードデータファイルの指定
   bes1：アンロードデータファイルがあるサーバの名称
   c:¥hirdb¥pdrorg¥unload_file1：アンロードデータファイルの名称
2. インデクス情報を出力するインデクス情報ファイル用ディレクトリの指定
   bes1，bes2，bes3：インデクス情報ファイルを作成するサーバの名称
   c:¥hirdb¥pdrorg¥idxwork：インデクス情報ファイル用ディレクトリの名称
3. ソート用ワークディレクトリの指定
   bes1，bes2，bes3：ソート用ワークファイルを作成するサーバの名称
   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

8.2.2　目的別の参照先一覧

ユーザが実行する再編成の形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)～(4)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」を参照してください。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k rorg	－
ユーザが定義した表の再編成（省略時仮定値が-c user なので省略できます）	-c user	－
再編成する表の名称	-t	－
制御文を記述したファイル	制御情報ファイル名	－
アンロードデータファイルの情報	－	unload 文

（凡例）－：該当しません。

(2)　再編成する表の属性によって指定する項目

項目	参照先	
	オプション	制御文
インデクスが定義されている場合	-i	index 文

項目	参照先	
	オプション	制御文
	-o	idxwork 文
		sort 文
LOB 列がある場合	-j	lobunld 文
プラグインが提供する抽象データ型列（LOB 属性）がある場合	-j	unld_func 文
		reld_func 文
サプレスオプションが指定されている表の場合で，列の後方に連続してある空白を取り除きたいとき※	-S	－
制約定義がある場合	－	constraint 文

（凡例）－：該当しません。

注※

このオプションを指定すると，処理時間の短縮，及びアンロードデータファイルの容量を削減できます。

(3) アンロードデータファイル，又は LOB データのアンロードファイルの種別によって指定する項目

項目	参照先	
	オプション	制御文
ファイルを一本化する場合	-g	－
HiRDB ファイルを使用する場合	-f	－
アンロードデータの格納順を指定する場合（クラスタキー順，インデクス順，又はデータ格納順のどれか）	-b	－

（凡例）－：該当しません。

(4) 再編成の方法によって指定する項目

項目	参照先	
	オプション	制御文
横分割表の場合で，特定の RD エリアだけ再編成したい場合	-r	－
データベース更新ログを取得する場合，又は取得しない場合	-l	－

項目	参照先	
	オプション	制御文
大量データの再編成の場合に，同期点を指定したいとき	−	option 文
pdrorg の実行時間を監視したい場合	−	option 文
表のステータスを強制的に通常状態に戻したい場合	−	option 文
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※1	-u	−
グローバルバッファを使用しないで，ローカルバッファを使って一括入出力の再編成をしたい場合※2	-n	−
再編成の実行状態のメッセージ出力間隔を，10 万行単位以外にしたい場合	-m	−
インフォメーションメッセージの出力を抑止したい場合		
サーバ間通信の応答時間を監視したい場合	-X	−
表データ中の空白文字コードを統一したい場合	−	option 文
再編成時に，表，及びインデクスの空き領域比率を変更したい場合※3	−	option 文
再編成時に，不要なデータを削除したい場合	−	unlduoc 文

（凡例）−：該当しません。

注※1

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

注※2

一括入出力ページ数を指定すると，指定したページ数単位で一括入出力をするので，I/O 回数を削減できます。

注※3

pdrorg 実行時に RD エリアの容量不足が発生した場合，暫定的に RD エリアを拡張しないで処理を完了できます（ただし，空き領域比率が 0 の表，及びインデクスを除く）。

8.3　表のアンロード

8.3.1　使用例

データベース再編成ユティリティの使用例の一覧（表のアンロード）を次に示します。

使用例	内　　容	区分
例 1	横分割表の表単位のアンロード • アンロードデータファイルは，pdload の入力データファイルとして利用するものとします。 • アンロードデータファイルを一本化します。	P
例 2	表のアンロードとリロード • アンロードしてからリロードするまでの間に表分割条件を変更します。	
例 3	表のアンロードとリロード • アンロードした表データを，別の表へリロードします。	
例 4	抽象データ型（SGMLTEXT 型）がある表のアンロードとリロード • アンロードしてからリロードするまでの間に表分割条件を変更します。	
例 5	抽象データ型（SGMLTEXT 型）がある表のアンロードとリロード • アンロードした表データを，別の表へリロードします。	
例 6	固定長データ形式の pdload 用入力データファイルの作成 • 表のデータを，固定長データ形式の pdload 用入力データファイルとしてアンロードします。	

（凡例）

P：HiRDB/パラレルサーバ

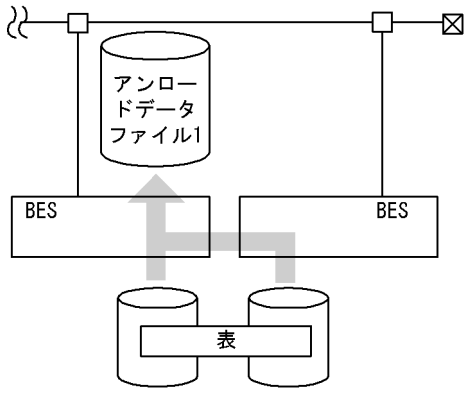
(1)　横分割表の表単位のアンロード

例 1

横分割表（TABLE1）を表単位にアンロードします。
アンロードしたデータは，pdloadの入力データファイル（バイナリ形式）として利用するものとします。また，アンロードデータファイルは一本化します。
表，及びインデクスは次のように定義されているものとします。

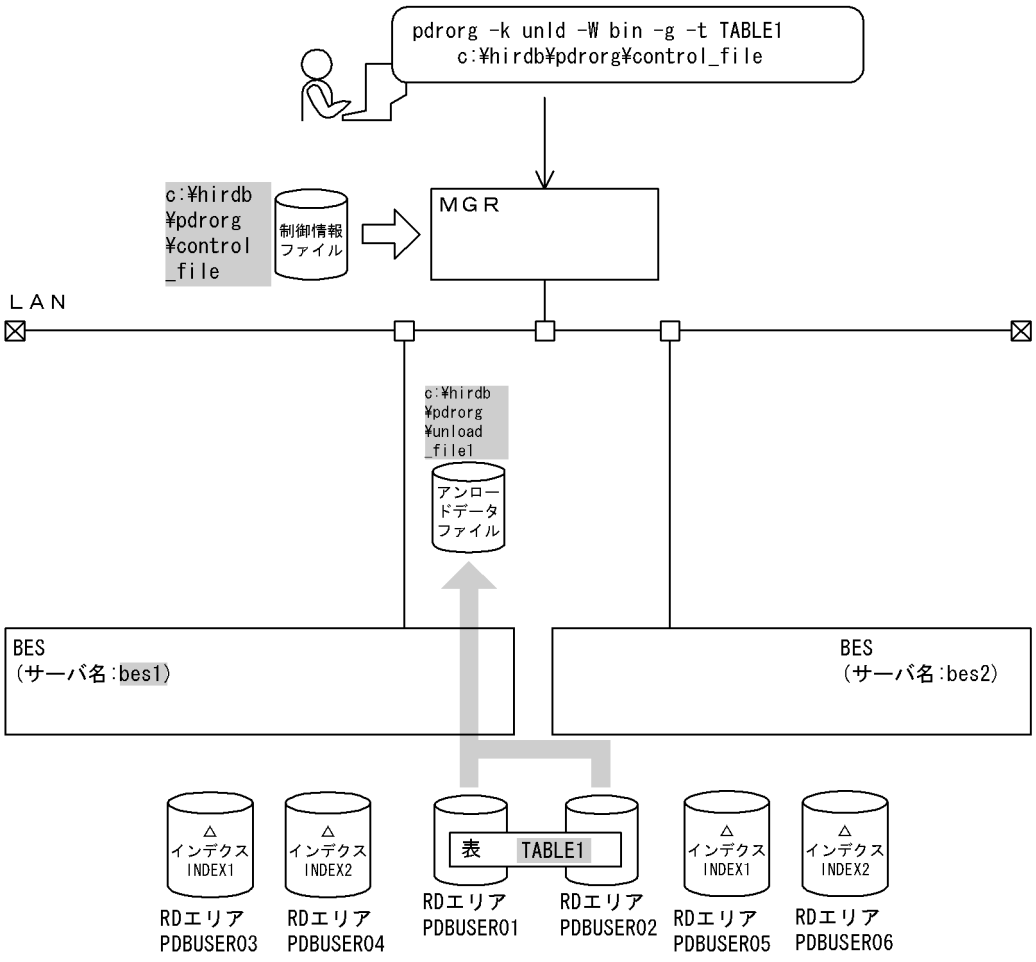
```
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

横分割表（TABLE1）を表単位にアンロードします。

-k unld：アンロードをするときの指定

- W bin：pdload の入力データファイル（バイナリ形式）にする場合の指定

-g：アンロードデータファイルを一本化する場合の指定

-t TABLE1：アンロード対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

［制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容］

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file1          1
```

［説明］

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file1：アンロードデータファイルの名称

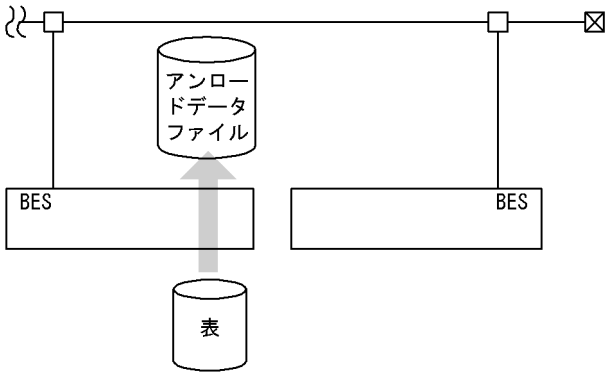
(2)　表の表分割条件を変更する場合のアンロードとリロード

例２

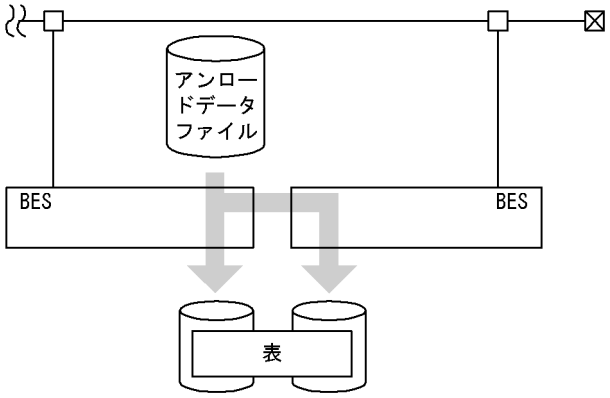
表（TABLE1）の表分割条件を変更するため，まず表をアンロードします。アンロードした後に表の表分割条件を変更して，その表にリロードします。
表，及びインデクスは次のように定義されているものとします。
・表の定義
```
CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT) IN PDBUSER01
```
・インデクスの定義（分割キーインデクス）
```
CREATE INDEX INDEX1 ON TABLE1(C1) IN PDBUSER03
```

[概要]

●表分割条件変更前のアンロード



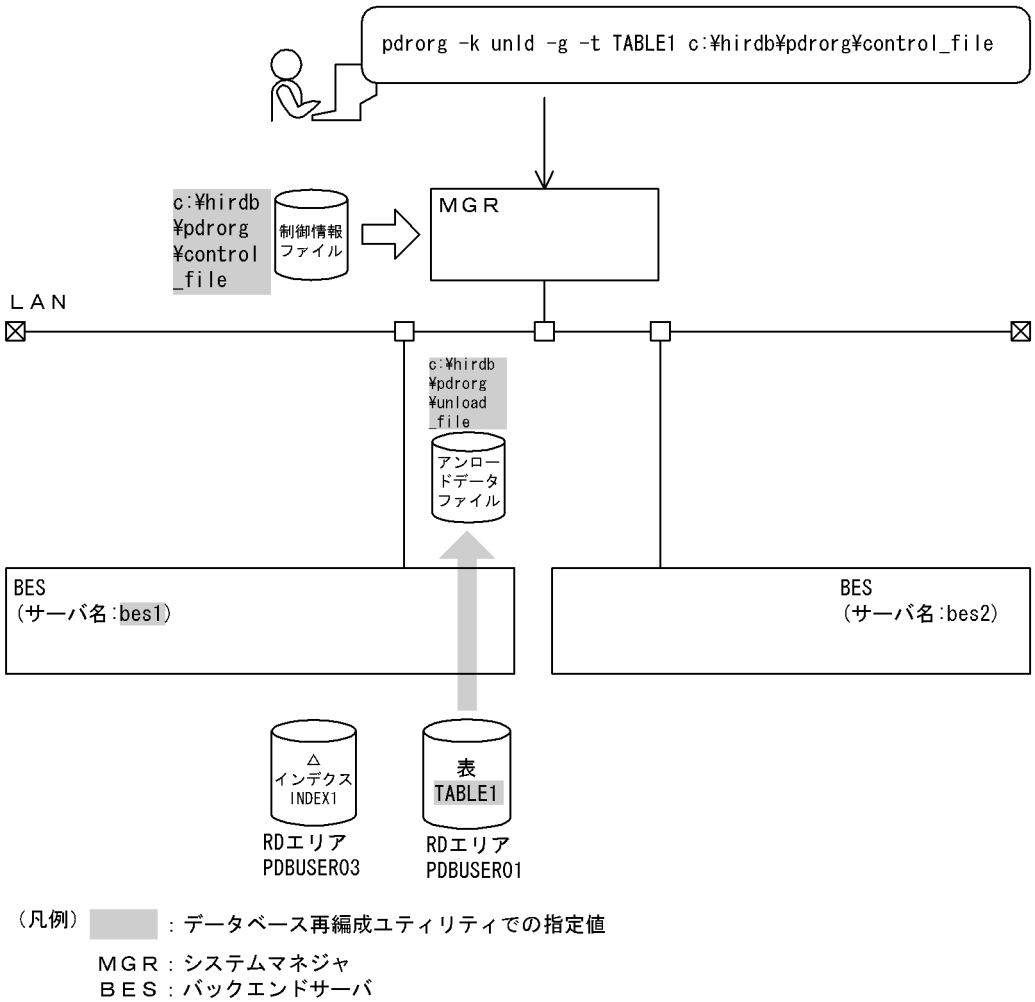
●表分割条件変更後のリロード



（凡例）BES：バックエンドサーバ

１．表分割条件変更前の表のアンロード

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

表（TABLE1）をアンロードします。

-k unld：アンロードをするときの指定

-t TABLE1：アンロード対象の表の名称

-g：表分割条件を変更するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file                    1
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

２．表（TABLE1）の表分割条件の変更

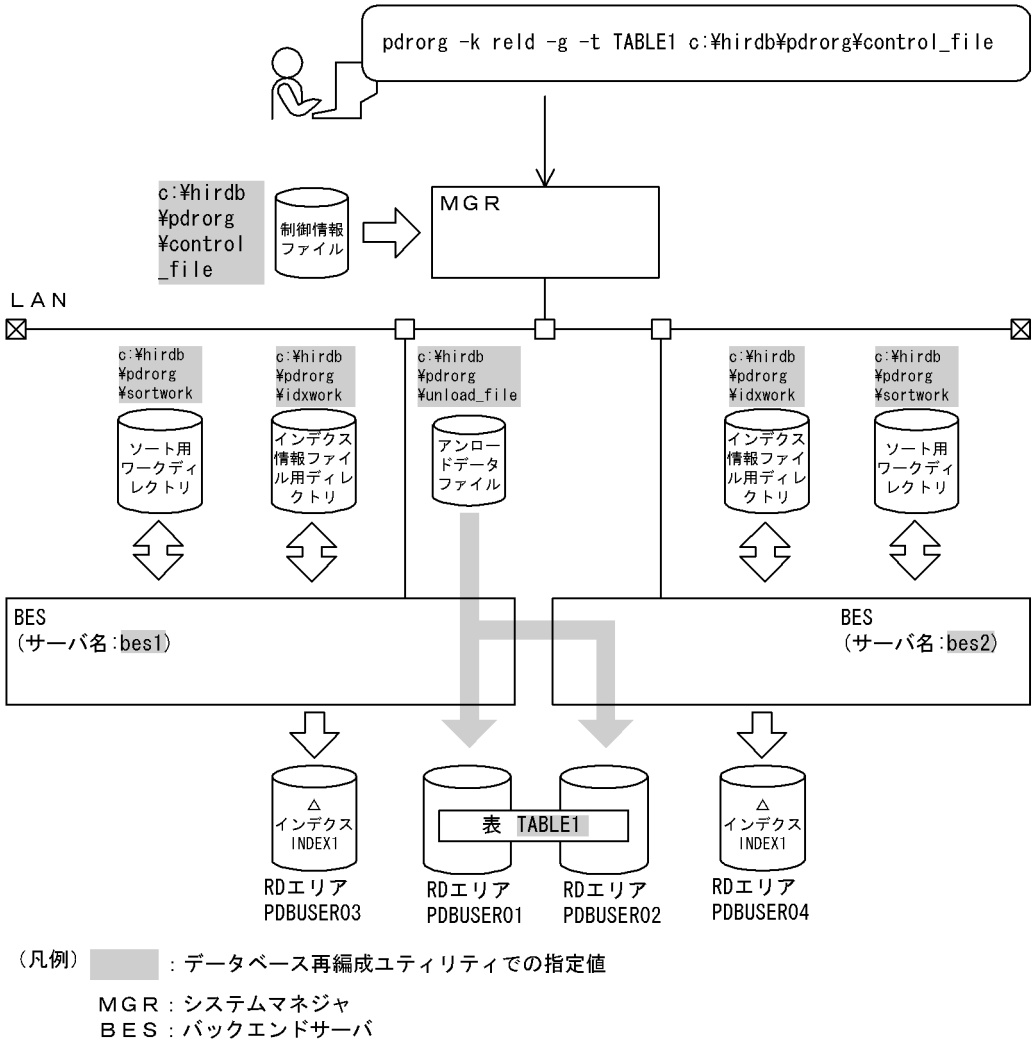
まず，表 TABLE1 を DROP TABLE で削除してください。

そして，表（TABLE1）の表分割条件を次のように定義し直します。また，インデクスも定義変更します。

```
CREATE TABLE TABLE1(C1 INT NOT NULL,
                    C2 CHAR(8),
                    C3 INT)
             IN ((PDBUSER01) C1 > 10,(PDBUSER02))
CREATE INDEX INDEX1 ON TABLE1(C1)
             IN ((PDBUSER03),(PDBUSER04))
```

３．表分割条件を変更した表へのリロード

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

表（TABLE1）へリロードします。

-k reld：リロードをするときの指定

-t TABLE1：リロード対象の表の名称

-g：表分割条件を変更するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file          1
idxwork bes1 c:¥hirdb¥pdrorg¥idxwork             2
idxwork bes2 c:¥hirdb¥pdrorg¥idxwork             2
sort bes1 c:¥hirdb¥pdrorg¥sortwork               3
sort bes2 c:¥hirdb¥pdrorg¥sortwork               3
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. インデクス情報を出力するインデクス情報ファイル用ディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥idxwork：インデクス情報ファイル用ディレクトリの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

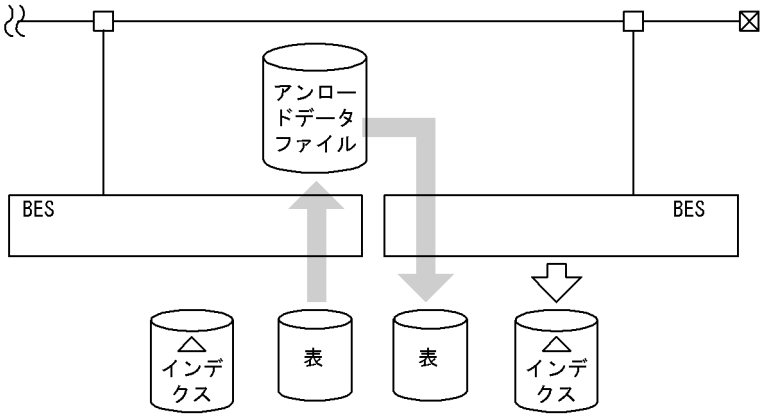
   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(3)　表データを別表に移行する場合のアンロードとリロード

例３

```
表（TABLE1）の表データを別表へ移行します。
表，及びインデクスは次のように定義されているものとします。
・移行元の表，及びインデクスの定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT) IN PDBUSER01
  CREATE INDEX INDEX1 ON TABLE1(C1) IN PDBUSER03
・移行先の表，及びインデクスの定義
  CREATE TABLE TABLE11(C1 INT NOT NULL,C2 CHAR(8),C3 INT) IN PDBUSER11
  CREATE INDEX INDEX11 ON TABLE11(C1) IN PDBUSER13
```

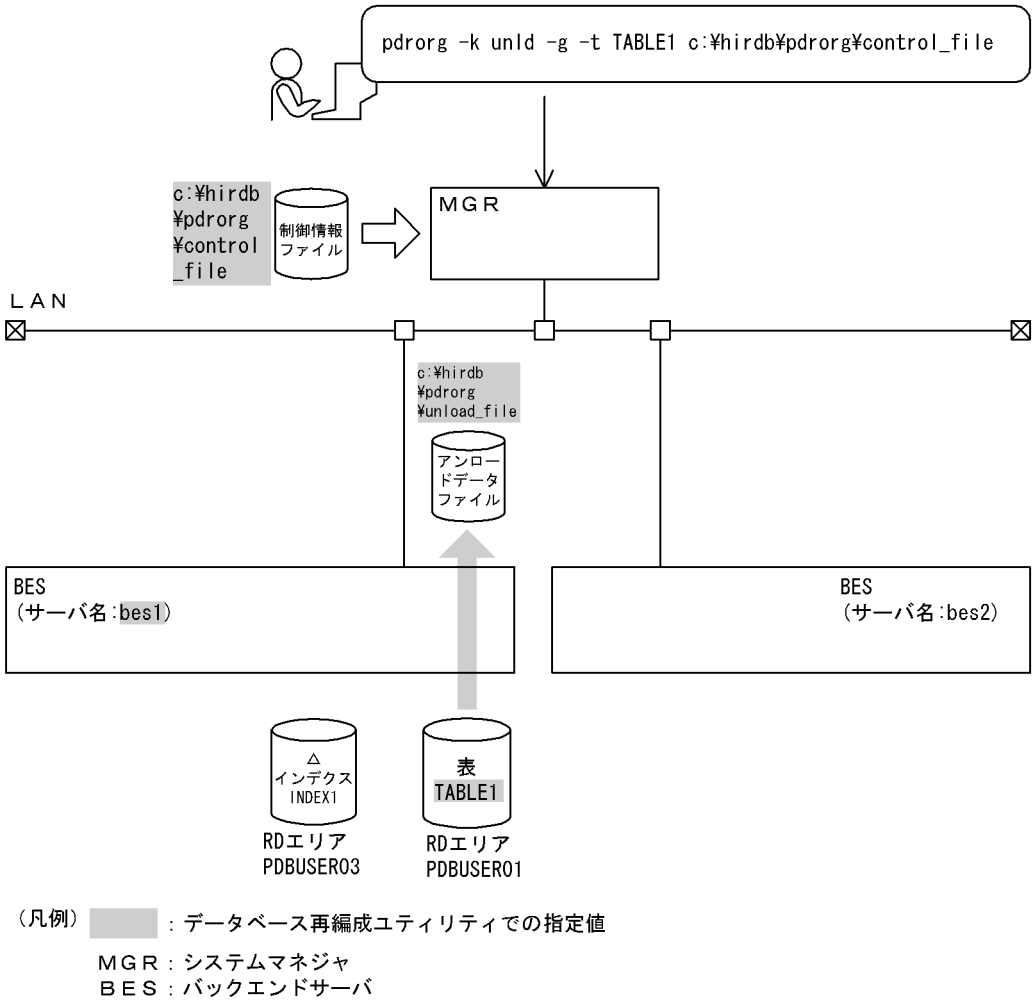
[概要]



（凡例）BES：バックエンドサーバ

１．移行元の表のアンロード

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

表（TABLE1）をアンロードします。

-k unld：アンロードをするときの指定

-t TABLE1：アンロード対象の表の名称

-g：別表へデータを移行するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file                    1
```

[説明]

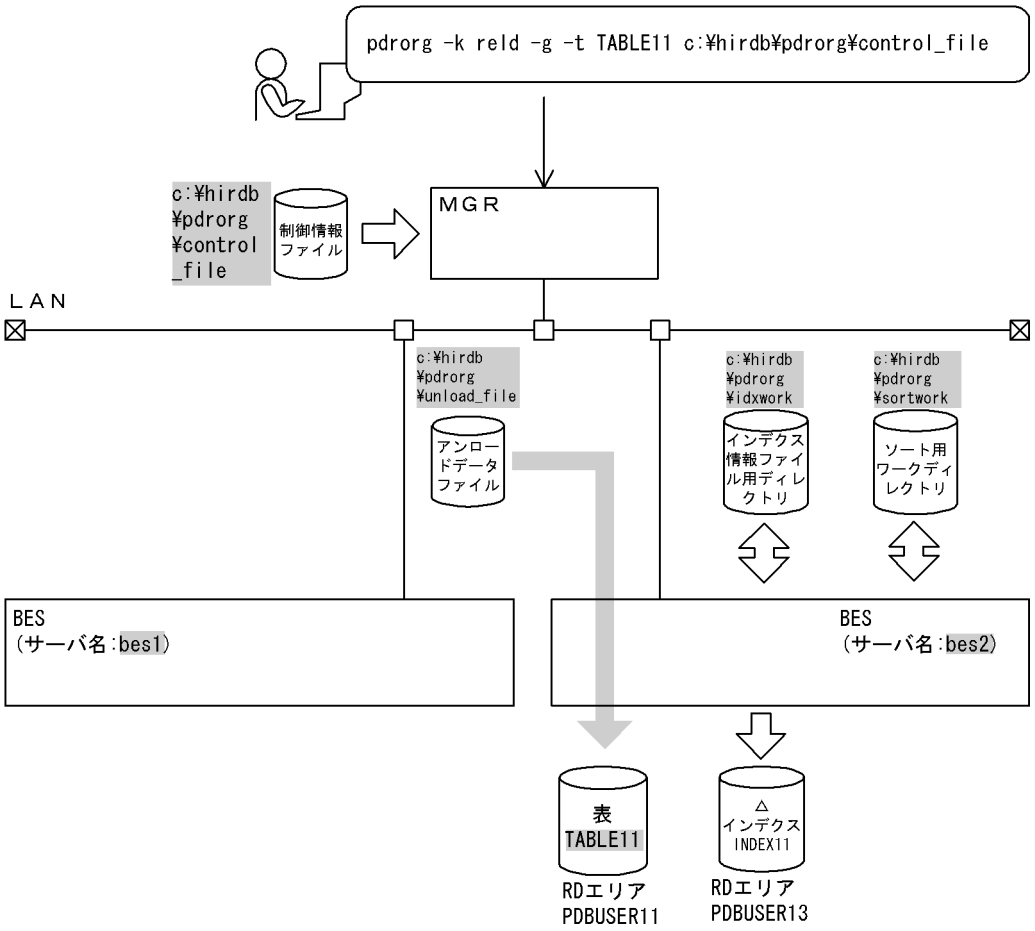
1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

２．移行先の表へのリロード（別表へのリロード）

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

表（TABLE11）へリロードします。

-k reld：リロードをするときの指定

-t TABLE11：リロード対象の表の名称

-g：別表へデータを移行するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file     1
idxwork bes2 c:¥hirdb¥pdrorg¥idxwork        2
sort bes2 c:¥hirdb¥pdrorg¥sortwork          3
tblname TABLE1                              4
```

[説明]

1. アンロードデータファイルの指定

bes1：アンロードデータファイルがあるサーバの名称

c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. インデクス情報を出力するインデクス情報ファイル用ディレクトリの指定

   bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥idxwork：インデクス情報ファイル用ディレクトリの名称

3. ソート用ワークディレクトリの指定

   bes2：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

4. 別表へ表データをリロードする場合の指定

   TABLE1：移行元の表の名称

(4) 抽象データ型がある表の表分割条件を変更する場合のアンロードとリロード

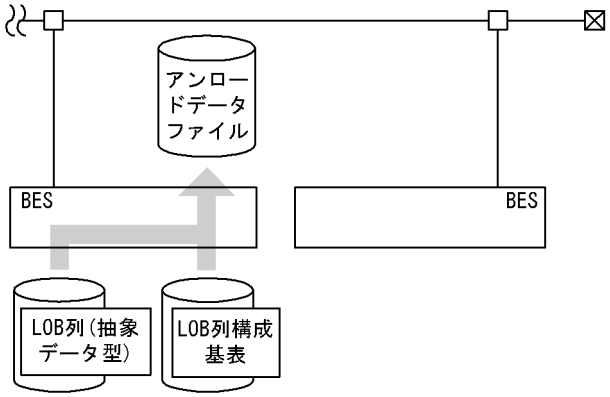
例 4

抽象データ型（SGMLTEXT）がある表（TABLE1）の表分割条件を変更するため，まず表をアンロードします。アンロードした後に表の表分割条件を変更して，その表にリロードします。
なお，抽象データ型（SGMLTEXT）はHiRDB Text Search Plug-inで提供されています。
・表の定義

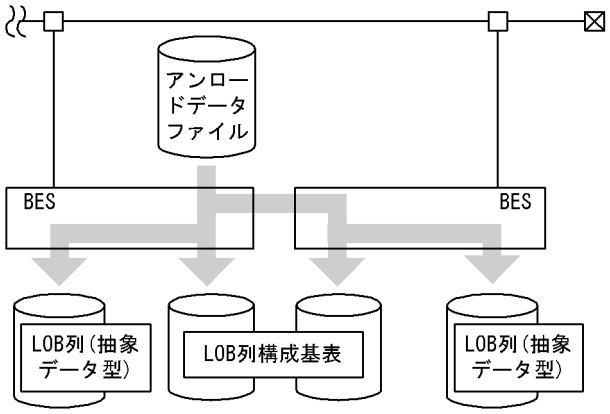
```
CREATE TABLE TABLE1(C1 INT,C2 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER01)
                    PLUGIN'<DTD>sgml.dtd</DTD>' ) IN PDUSER01
```

[概要]

●表分割条件変更前のアンロード



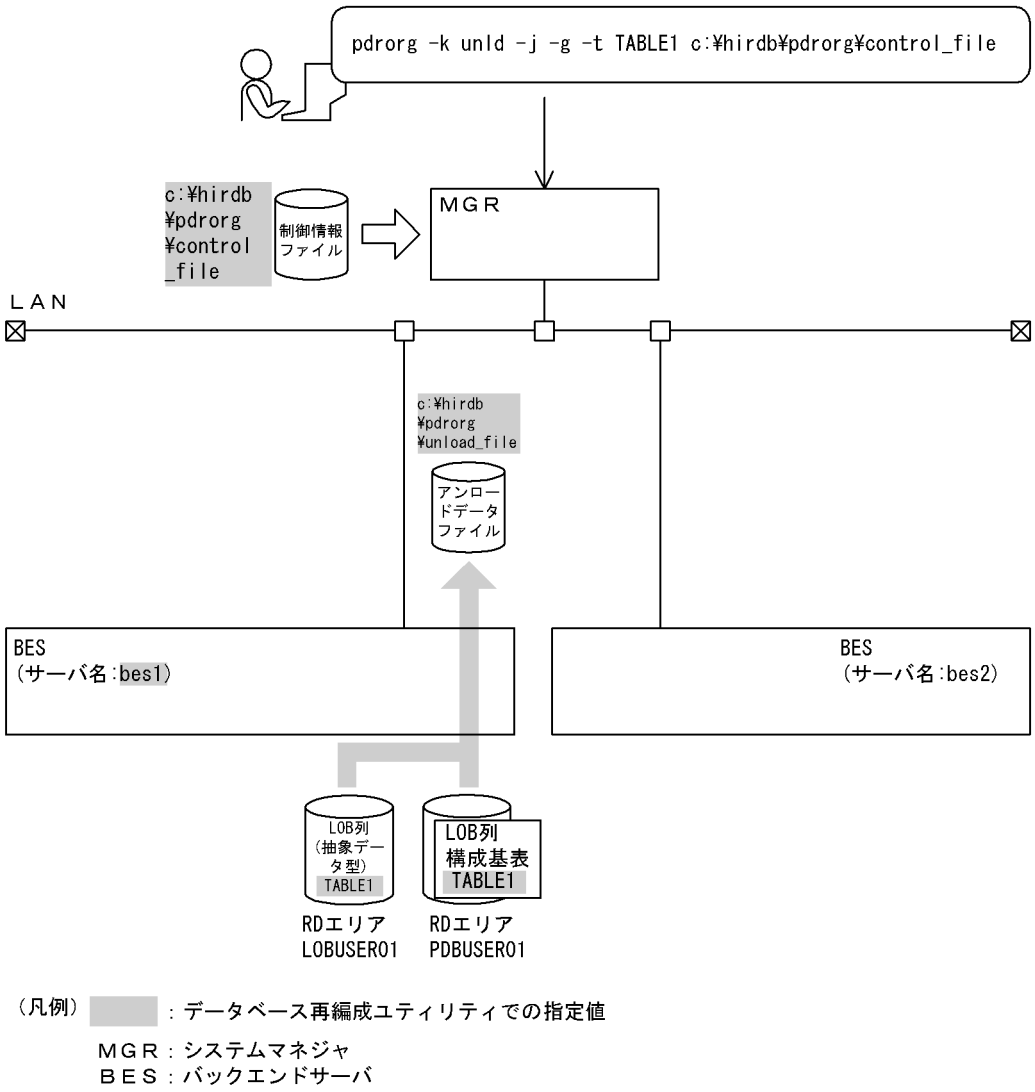
●表分割条件変更後のリロード



（凡例）BES：バックエンドサーバ

１．表分割条件変更前の表のアンロード

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

表（TABLE1）をアンロードします。

-k unld：アンロードをするときの指定

-t TABLE1：アンロード対象の表の名称

-j：LOB 属性の抽象データ型がある表をアンロードするときの指定

-g：表分割条件を変更するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file                    1
unld_func type=SGMLTEXT,func=unsgmltext(sgmltext)          2
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. コンストラクタパラメタ逆生成関数の指定

SGMLTEXT：抽象データ型名（列 C2）

unsgmltext：コンストラクタパラメタ逆生成関数の名称（ここでの名称は例なので，正式な名称については各種プラグインのマニュアルを参照してください）

sgmltext：コンストラクタパラメタ逆生成関数の引数のデータ型

２．表（TABLE1）の表分割条件の変更

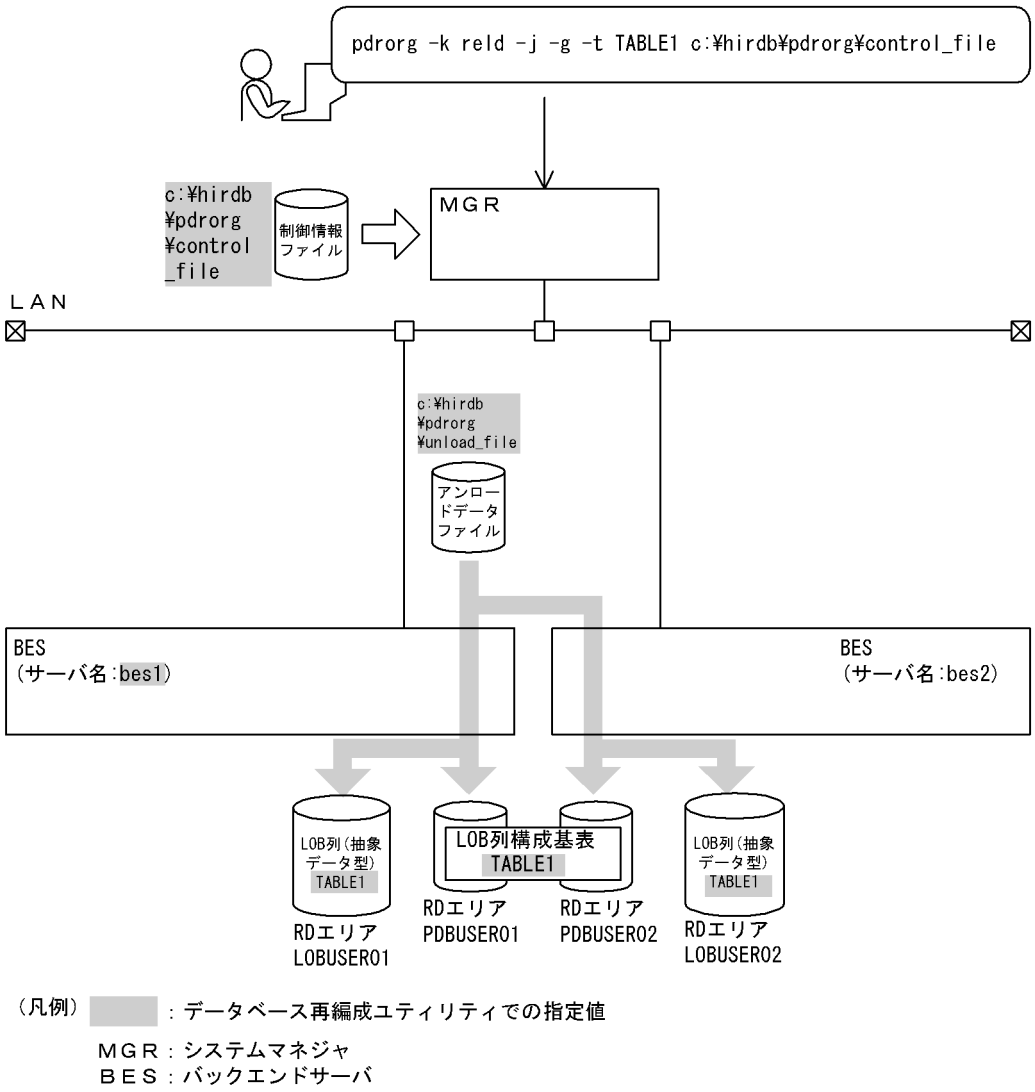
まず，表 TABLE1 を DROP TABLE で削除してください。

そして，表（TABLE1）の表分割条件を次のように定義し直します。

```
CREATE TABLE TABLE1(C1 INT NOT NULL,
                    C2 SGMLTEXT ALLOCATE
                       (SGMLTEXT IN ((LOBUSER01),(LOBUSER02)))
                        PLUGIN'<DTD>sgml.dtd</DTD>' )
   IN ((PDUSER01) C1<2000,(PDBUSER02))
```

３．表分割条件を変更した表へのリロード

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

表（TABLE1）へリロードします。

-k reld：リロードをするときの指定

-t TABLE1：リロード対象の表の名称

-j：LOB 属性の抽象データ型がある表をリロードするときの指定

-g：表分割条件を変更するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file              1
reld_func type=SGMLTEXT,func=sgmltext(blob)          2
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. コンストラクタ関数の指定

SGMLTEXT：抽象データ型名（列 C2）

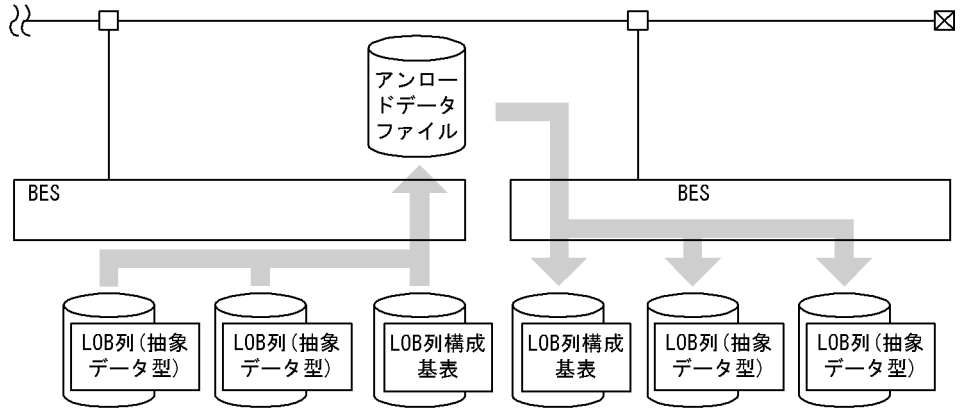
sgmltext(blob)：コンストラクタ関数の名称（括弧内はコンストラクタ関数の引数のデータ型）

(5) 抽象データ型がある表の表データを別表に移行する場合のアンロードとリロード

例５

```
抽象データ型（SGMLTEXT）がある表（TABLE1）の表データを別表に移行します。
なお，抽象データ型（SGMLTEXT）はHiRDB Text Search Plug-inで提供されています。
・移行元の表の定義
  CREATE TABLE TABLE1(C1 INT,
                      C2 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER01)
                        PLUGIN'<DTD>sgml.dtd</DTD>'
                      C3 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER02)
                        PLUGIN'<DTD>sgml.dtd</DTD>'
                      ) IN PDUSER01
・移行先の表の定義
  CREATE TABLE TABLE2(C1 INT,
                      C2 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER03)
                        PLUGIN'<DTD>sgml.dtd</DTD>'
                      C3 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER04)
                        PLUGIN'<DTD>sgml.dtd</DTD>'
                      ) IN PDUSER02
```

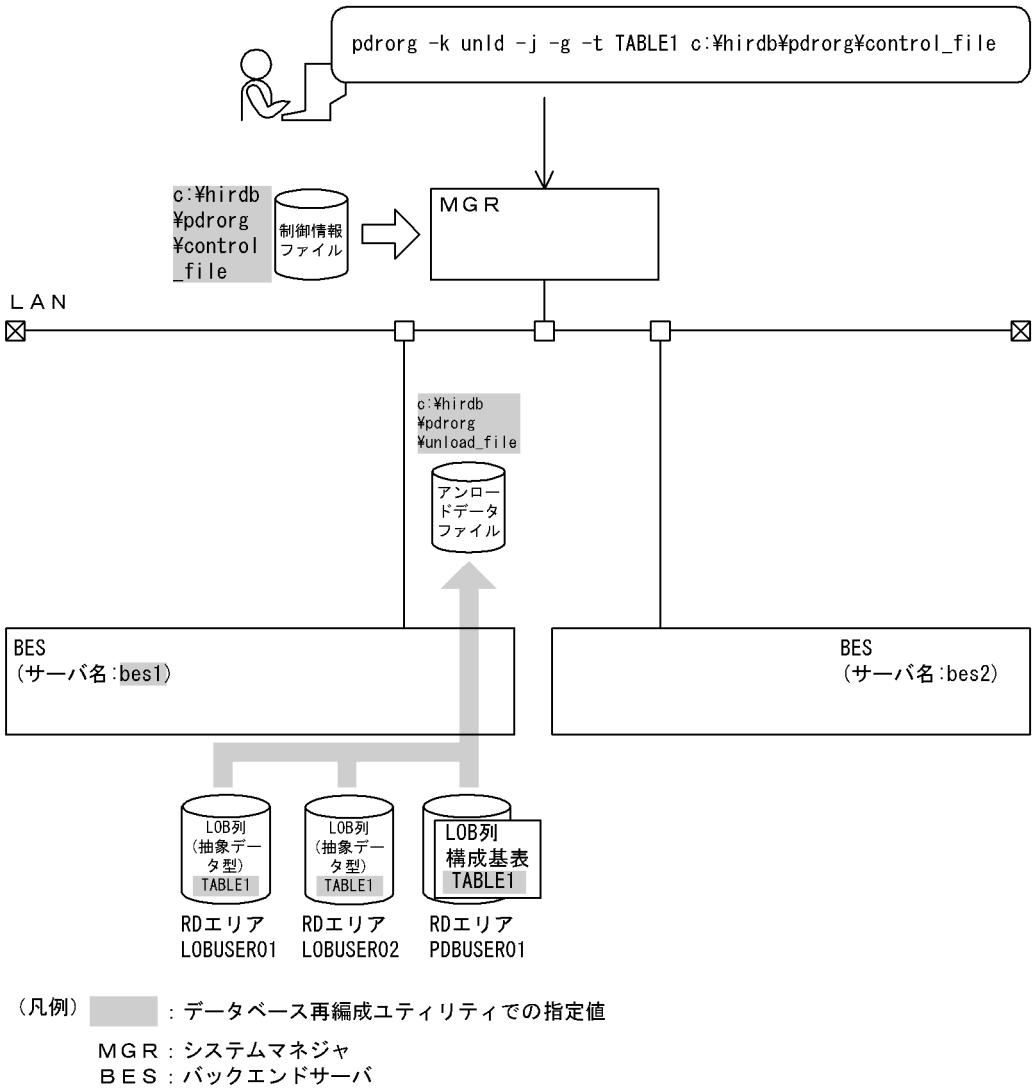
[概要]



（凡例）BES：バックエンドサーバ

１．移行元の表のアンロード

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

表（TABLE1）をアンロードします。

-k unld：アンロードをするときの指定

-t TABLE1：アンロード対象の表の名称

-j：LOB 属性の抽象データ型がある表をアンロードするときの指定

-g：別表へデータを移行するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file                    1
unld_func type=SGMLTEXT,func=unsgmltext(sgmltext)          2
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. コンストラクタパラメタ逆生成関数の指定

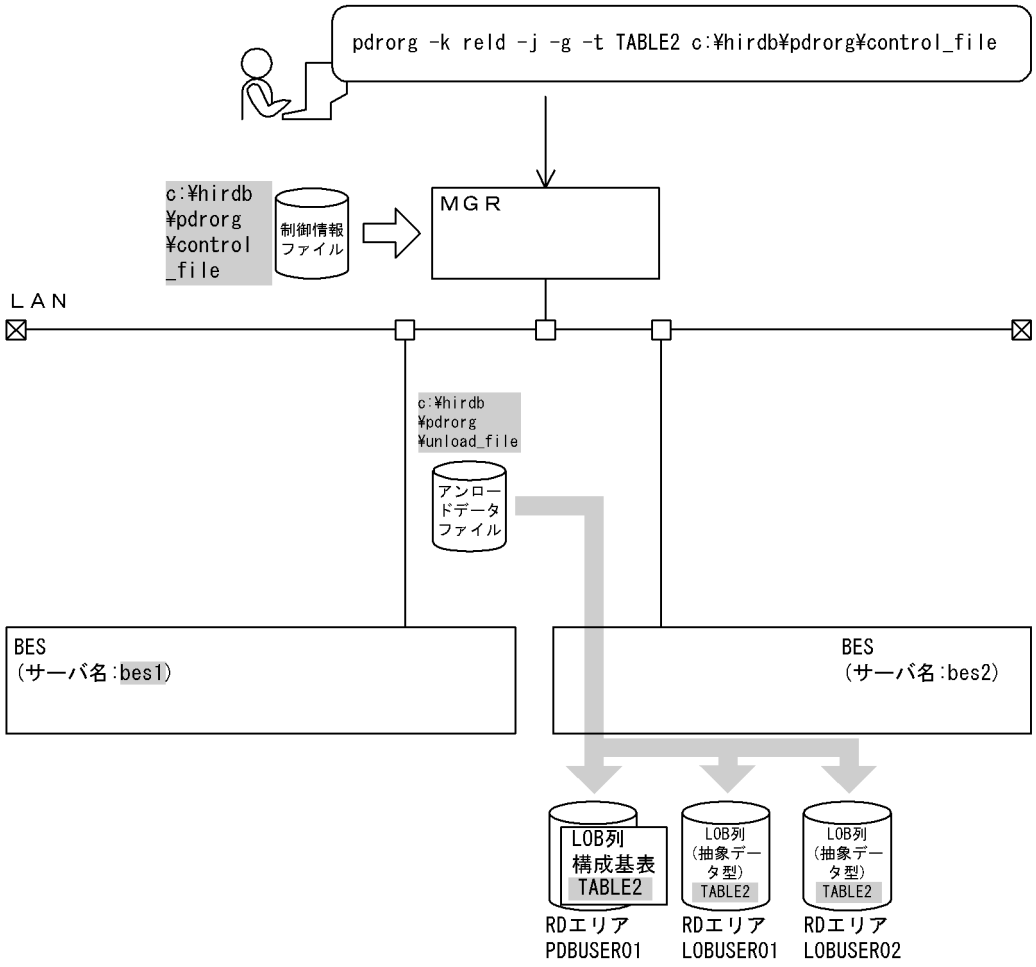
SGMLTEXT：抽象データ型名（列 C2）

unsgmltext：コンストラクタパラメタ逆生成関数の名称（ここでの名称は例なので，正式な名称については各種プラグインのマニュアルを参照してください）

sgmltext：コンストラクタパラメタ逆生成関数の引数のデータ型

２．移行先の表へのリロード（別表へのリロード）

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

表（TABLE2）へリロードします。

-k reld：リロードをするときの指定

-t TABLE2：リロード対象の表の名称

-j：LOB 属性の抽象データ型がある表をリロードするときの指定

-g：別表へデータを移行するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file              1
reld_func type=SGMLTEXT,func=sgmltext(blob)          2
tblname TABLE1                                       3
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. コンストラクタ関数の指定

   SGMLTEXT：抽象データ型名（列 C2）

   sgmltext(blob)：コンストラクタ関数の名称（括弧内はコンストラクタ関数の引数のデータ型）

3. 別表へ表データをリロードする場合の指定

   TABLE1：移行元の表の名称

(6)　固定長データ形式の pdload 用入力データファイルの作成

例 6

表「商品表」のデータを，固定長データ形式のpdload用入力データファイルとしてアンロードします。

```
・表の定義
 CREATE TABLE 商品表
       (商品番号 SMALLINT NOT NULL,商品名 NVARCHAR(12),数量 INTEGER)
       IN ((RDAREA1)商品番号<=10000,(RDAREA2)商品番号<=20000,(RDAREA3));
```

[pdrorg のコマンド]

```
pdrorg -k unld -t 商品表 -W fixtext,@,cr c:¥hirdb¥pdrorg¥control_file
```

[説明]

-k unld：アンロードをするときの指定

-t 商品表：アンロード対象の表の名称

-W fixtext,@,cr：固定長データ形式の pdload 用入力データファイルとしてアンロードする場合の指定（パディング文字は@で，1 バイトの改行コードを付けます）

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload bes1:c:¥hirdb¥pdrorg¥unload_file                              1
fixtext_option enclose=” format=integer,type1 format=smallint,type2  2
```

[説明]

1. アンロードデータファイルの指定

   bes1：アンロードデータファイルがあるサーバの名称

   c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. 出力データを編集する場合の指定

   enclose="：データ型が NVARCHAR の列「商品名」のデータを，囲み文字「"」で囲みます。

   format=integer,type1：データ型が INTEGER の列「数量」のデータを，出力形式 type1 で出力します。

   format=smallint,type2：データ型が SMALLINT の列「商品番号」のデータを，出力形式 type2 で出力します。

[出力データ]

```
商品番号        商品名                  数量
（6バイト）     （14バイト）            （11バイト）          （1バイト）

###100         ”BOOTS”@@@@@@@          @@@@@@@@@@@          改行文字
@@@@@@         ””@@@@@@@@@@@@          #0000000100          改行文字
-12345         @@@@@@@@@@@@@@          -0000000300          改行文字
```

注

＃は空白文字です。また，1 行は 32 バイトです。

8.3.2　目的別の参照先一覧

ユーザが実行するアンロードの形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)～(4)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」が参照先となります。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k unld	－
ユーザが定義した表のアンロード（省略時仮定値が-c user なので省略できます）	-c user	－
アンロードする表の名称	-t	－
制御文を記述したファイル	制御情報ファイル名	－
アンロードデータファイルの情報	－	unload 文

（凡例）－：該当しません。

(2)　アンロードする表の属性によって指定する項目

項目	参照先	
	オプション	制御文
LOB 列がある場合	-j	lobunld 文
プラグインが提供する抽象データ型列（LOB 属性）がある場合	-j	unld_func 文
繰返し列がある場合	-W	array 文
サプレスオプションが指定されている表の場合で，列の後方に連続してある空白を取り除きたいとき※	-S	－

（凡例）－：該当しません。

注※

　このオプションを指定すると，処理時間の短縮，及びアンロードデータファイルの容量を削減できます。

(3)　アンロードデータファイル，又は LOB データのアンロードファイルの種別によって指定する項目

項目	参照先	
	オプション	制御文
アンロードデータを，pdload の入力データファイルに使用する場合，又は UAP で使用する場合	-W	fixtext_option 文
ファイルを一本化する場合	-g	－
HiRDB ファイルを使用する場合	-f	－
アンロードデータの格納順を指定する場合（クラスタキー順，インデクス順，又はデータ格納順のどれか）	-b	－

（凡例）－：該当しません。

(4)　アンロードの方法によって指定する項目

項目	参照先	
	オプション	制御文
横分割表の場合で，特定の RD エリアだけアンロードしたい場合	-r	－

項目	参照先	
	オプション	制御文
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※1	-u	−
グローバルバッファを使用しないで，ローカルバッファを使って一括入出力のアンロードをしたい場合※2	-n	−
アンロードの実行状態のメッセージ出力間隔を，10 万行単位以外にしたい場合	-m	−
インフォメーションメッセージの出力を抑止したい場合		
サーバ間通信の応答時間を監視したい場合	-X	−
pdrorg の実行時間を監視したい場合	−	option 文
表データ中の空白文字コードを統一したい場合	−	option 文
アンロードデータファイルのデータを UTF-8，又は UTF-16 に統一してアンロードしたい場合	−	option 文
BLOB 属性の抽象データ型の列を，VARCHAR 型でアンロードしたい場合	−	blobtovarchar 文
次のどれかを実行したい場合 • 不要なデータを削除してアンロードデータファイルを出力したいとき • データを更新して pdload の入力データファイルにしたいとき • アプリケーション用に任意のフォーマットで出力したいとき	−	unlduoc 文

（凡例）−：該当しません。

注※1

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

注※2

一括入出力ページ数を指定すると，指定したページ数単位で一括入出力をするので，I/O 回数を削減できます。

8.3.3　データベース作成ユティリティの入力用ファイルの形式

-W オプションを指定した場合，作成されたアンロードデータファイルは，データベース作成ユティリティの入力データファイルとして使用できます。

(1)　DAT 形式，及び拡張 DAT 形式の場合

DAT 形式，及び拡張 DAT 形式の場合，データは文字データで出力します。DAT 形式と拡張 DAT 形式の相違点については，「DAT 形式と拡張 DAT 形式の相違点」を参照してください。この相違点以外は，DAT 形式と拡張 DAT 形式は同じになります。

規則を次に示します。

1. 表の 1 行に対して，ファイルの 1 行を出力します（行の終わりには改行文字を出力します）。
2. 列データ間にはセパレータ文字が入ります。各データ型に対する列データの出力形式（DAT 形式）については，「DAT 形式の場合」を参照してください。

   文字データ，各国文字データ，混在文字データ，及び BINARY データの出力形式は，引用符（"）で囲んだ形となります。

   抽象データ型の出力形式は，コンストラクタパラメタ逆生成関数の指定数文のデータを，セパレータ文字で区切った形となります。ただし，コンストラクタパラメタ逆生成関数の戻り値が BLOB 型の場合は，ナル値としてセパレータ文字を連続して出力します。
3. LOB 列のデータは，-j オプションの指定に関係なくナル値として出力します。
4. 抽象データ型の列のデータは，コンストラクタパラメタ逆生成関数の戻り値が BLOB 型の場合はナル値として出力します。
5. 変換後の行長が 512 メガバイトを超える場合はエラーとなります。また，アンロードデータファイルにストリーミングテープ装置を割り当てた場合は，行長が 32 キロバイトを超えるとエラーになります。
6. option 文に spacelvl=3 を指定していて，更に sup オプションを指定している場合，NCHAR 型の半角空白 2 バイトを空白変換してから，半角空白 2 バイト分を一組として出力抑止します。
7. 列のデータ型が文字データ，各国文字データ，混在文字データ，又はバイナリの場合，列値中に囲み文字（"）＋セパレータ文字，又はセパレータ文字＋囲み文字（"）の組み合わせの文字の並びがあるときは，それらを列データの終了位置，開始位置とみなしてデータロードすることがあります。これを防ぐには，-W オプションでセパレータ文字を変更してアンロードするか，又はバイナリ形式で出力するかしてください。
8. 予備列が定義された FIX 属性の表の場合，予備列は出力できません。

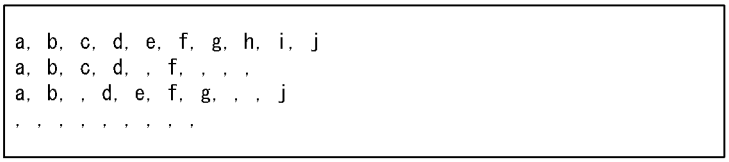
(a)　繰返し列の場合の形式

繰返し列の場合は，各要素のデータをセパレータ文字で区切って出力します。

ただし，制御情報ファイルの array 文の指定によって行データ形式が異なります。array 文に ff を指定した場合の行データ出力形式と vv を指定した場合の行データ出力形式を次の図に示します。

図 8–12　array 文に ff を指定した場合の行データ出力形式

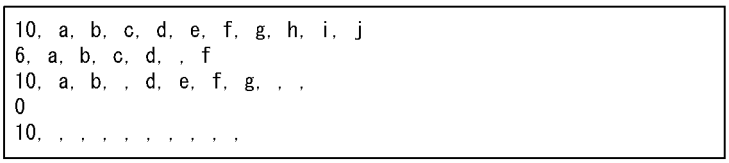
例）CHAR(1) ARRAY[10]の列の場合



[説明]
1. 全要素にデータが格納されています。
2. １～６番目の要素にデータが格納されていて，５番目の要素はナル値が格納されています。７～１０番目の要素はデータが格納されていないか，ナル値が格納されています。
3. 全要素にデータが格納されていて，３，８，９番目の要素はナル値が格納されています。
4. 列値がナル値，全要素がナル値のどちらかです。

図 8–13　array 文に vv を指定した場合の行データ出力形式

例）CHAR(1) ARRAY[10]の列の場合



[説明]
1. 全要素にデータが格納されています。実要素数には１０を出力します。
2. １～６番目の要素にデータが格納されていて，５番目の要素はナル値が格納されています。実要素数には６を出力します。
3. 全要素にデータが格納されていて，３，８～１０番目の要素はナル値が格納されています。実要素数には１０を出力します。
4. 列値にナル値が格納されているため，実要素数に０を出力します。
5. 全要素にナル値が格納されています。実要素数には１０を出力します。

(b)　sup オプションを指定した場合の形式

アンロード時に sup オプションを指定すると，表定義の列長に満たない部分の空白を圧縮した形式でアンロードデータファイルに出力します。sup オプションを指定した場合の形式と，指定しない場合の形式を次に示します。

- **データ型が CHAR，又は MCHAR の場合**

（例１） 表定義　CREATE TABLE T1(C1 INTEGER,C2 CHAR(10),…);

入力したデータ　1,AA,…

sup オプションを指定した場合：

```
1,”AA”,…
```

[説明]

表定義長に満たない部分(８文字分)の空白を圧縮して出力します。

sup オプションを指定しない場合：

```
1,”AA........”,…
```

注　「.」は半角空白を示します。

[説明]

表定義長に満たない部分(８文字分)に半角空白(.)を埋めて出力します。

（例２） 表定義　CREATE TABLE T1(C1 INTEGER,C2 MCHAR(10),…);

入力したデータ　1,.....,…(.は半角空白を示します)

sup オプションを指定した場合：

```
1,".",…
```

[説明]

半角空白を１バイト出力します。

sup オプションを指定しない場合：

```
1,".........",…
```

[説明]

表定義長分半角空白を出力します。

- **データ型が NCHAR の場合**

（例１） 表定義　CREATE TABLE T1(C1 INTEGER,C2 NCHAR(10),…);

入力したデータ　１,ＡＡ,…

sup オプションを指定した場合：

```
１,"ＡＡ",…
```

[説明]

表定義長に満たない部分(８文字分)の空白を圧縮して出力します。

sup オプションを指定しない場合：

```
１,"ＡＡ△△△△△△△△△△",…

注　「△」は２バイトの空白を示します。
```

[説明]

表定義長に満たない部分(８文字分)に２バイトの空白を出力します。

（例２） 表定義　CREATE TABLE T1(C1 INTEGER,C2 NCHAR(10),…);

入力したデータ　１,△△△△△△△△△△,…

(△は２バイトの空白を示します)

sup オプションを指定した場合：

```
１,"△",…
```

[説明]

２バイトの空白を一つ出力します。

sup オプションを指定しない場合：

```
1,”△△△△△△△△△△”,…
```

[説明]

表定義長分に２バイトの空白を出力します。

- **データ型が NCHAR の場合（option 文に spacelvl=3 を指定した場合）**

（例１） 表定義　CREATE TABLE T1(C1 INTEGER,C2 NCHAR(10),…);

入力したデータ　１,ＡＡ....,…(.は半角空白を示します)

sup オプションを指定した場合：

```
1,”ＡＡ”,…
```

[説明]

表定義長に満たない部分(８文字分)の空白を圧縮して出力します。

sup オプションを指定しない場合：

```
1,”ＡＡ..............”,…
```

[説明]

表定義長に満たない部分(８文字分)に半角空白を出力します。

（例２） 表定義　CREATE TABLE T1(C1 INTEGER,C2 NCHAR(10),…);

入力したデータ　１,△△△△△△△△△△,…

(△は２バイトの空白を示します)

sup オプションを指定した場合：

```
1,”..”,…
注　「．」は半角空白を示します。
```

[説明]

半角空白を２バイト分出力します。

sup オプションを指定しない場合：

```
1,”..................”,…
注　「．」は半角空白を示します。
```

[説明]

表定義長分に半角空白を出力します。

(c)　DAT 形式と拡張 DAT 形式の相違点

DAT 形式と拡張 DAT 形式の相違点を次に示します。

項目	DAT 形式	拡張 DAT 形式
データ中に NULL 文字（¥0），又は改行文字（¥n）がある場合	該当する列データ値を含む行は出力しません。	該当する列データ値を含む行を出力します（¥0 及び¥n はそのまま出力します）。
データ中に囲み文字（"）と同じデータがある場合	該当する列データ値としての囲み文字は，そのまま出力します。	該当する列データ値としての囲み文字を，二重にして出力します。※ （例）「"」→「""」

注※

マルチバイト文字も変換対象になるため，AP などでファイルの内容を参照する場合は注意が必要です。なお，pdload では，マルチバイト文字に関係なく「""」を「"」に変換するため，データロードはできます。

(d) 文字コードが UTF-16 の拡張 DAT 形式の pdload 用アンロードファイルを作成する場合

文字コードが UTF-16 の拡張 DAT 形式の pdload 用アンロードファイルを作成する場合，次のデータが表に格納されているときは，アンロード時に対処が必要となります。

列のデータ型	データの内容
BINARY 型	奇数バイト長のデータ
BINARY 型データを返却するコンストラクタパラメタ逆生成関数を使用してアンロードする抽象データ型	コンストラクタパラメタ逆生成関数の返却値が奇数バイト長のデータ

上記のデータが表に格納されているかどうかは，次の SQL 文を実行することで確認できます。BINARY 型，及び BINARY 型データを返却するコンストラクタパラメタ逆生成関数を使用してアンロードする抽象データ型のすべての列に対して実行してください。

```
SELECT COUNT(*) FROM [認可識別子.]表識別子
                          WHERE 制約式 [OR 制約式 ...]

制約式 ::= {  MOD(LENGTH(BINARY型列名),2)=1
            | MOD(LENGTH(関数名(列名[,列名 ...])),2)=1 }
関数名 ::= BINARY型データを返却するコンストラクタパラメタ逆生成関数の名称
```

- SQL 文の実行結果が 0 件の場合

  特に対処は必要ありません。そのままアンロードしてください。

- SQL 文の実行結果が 0 件でない場合

  次のどちらかの方法でアンロードしてください。

  - 文字コードを UTF-8（option 文の charset オペランド指定なし）にして，アンロードしてください。
  - バイナリ形式（-W bin 指定）でアンロードしてください。

(2)　バイナリ形式の場合

バイナリ形式の場合，データはデータベースの列定義に合わせた内部形式で出力します。このため，このファイルをデータベース作成ユティリティの入力とする場合は，アンロードした表とデータロード先の表の列定義は同じでなければなりません。また，非 FIX 表をアンロードした場合は，列定義が同じであっても FIX 表にはデータロードできません。

出力形式を次に示します。



●CMT ヘッダ

CMT に出力する場合は，最初に 512 バイト分の CMT ヘッダを付けて出力します。

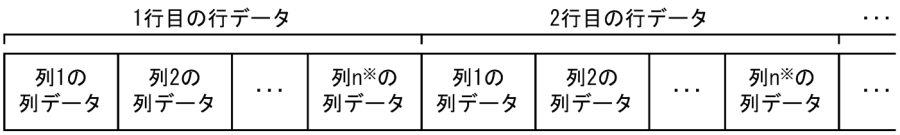
●行データ

FIX 表と非 FIX 表では行データの出力形式が異なります。なお，各データ型に対する列データの出力形式（バイナリ形式）については，「バイナリ形式の場合」を参照してください。

・FIX 表の場合

列の定義順に，列のデータ値をバウンダリしないで詰めて出力します。FIX 表の出力形式を次の図に示します。

図 8–14　FIX 表の出力形式



（凡例）n：表の列定義数
注※：予備列が定義されたFIX属性の表の場合，列nは予備列のため，出力されません。

・非 FIX 表の場合

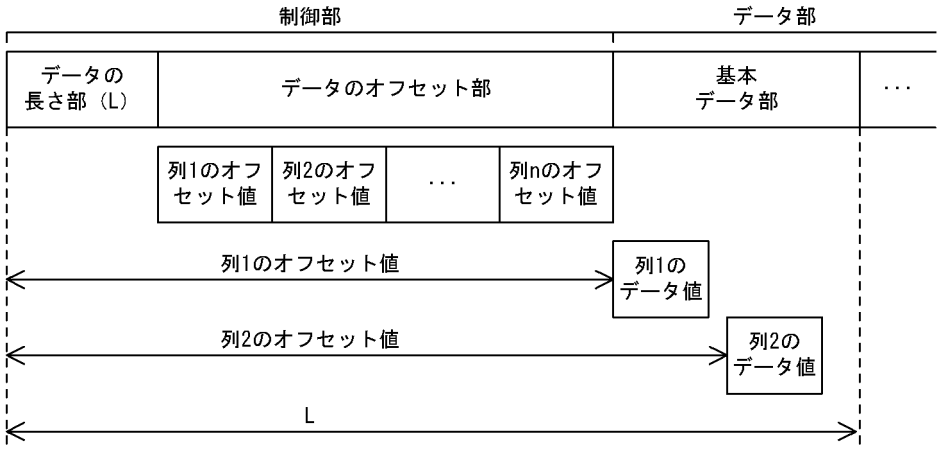
各部を順に，バウンダリしないで詰めて出力します。非 FIX 表の出力形式を次の図に示します。

図 8–15　非 FIX 表の出力形式



制御部の出力形式を次の図に示します。

図 8–16　非 FIX 表の制御部の出力形式



（凡例）n：表の列数

- **データの長さ部**

  データの長さ部の先頭から，データ部の最後までの長さを 2 進数（4 バイト）で出力します。図「非 FIX 表の制御部の出力形式」では，L になります。

- **データのオフセット部**

  データのオフセット部は，列定義順にバウンダリしないで詰めて出力します。対象列が抽象データ型の場合は，コンストラクタ逆生成関数の関数分（unld_func 文に指定した数），関数指定順（unld_func 文に指定した順）にオフセット値をバウンダリしないで詰めて出力します。

- **オフセット値**

  データの長さ部の先頭から，各列（コンストラクタパラメタ逆生成関が生成したデータ値）のデータ値の先頭までの長さを 2 進数（4 バイト）で出力します。データ値がナル値の場合は，0 を出力します。

データ部は，図「非 FIX 表の出力形式」に示すように，基本データ部，BINARY 属性データ部，BINARY 列データ部，BLOB 属性データ部，及び BLOB 列データ部から成ります。

基本データ部

制御部（データのオフセット値）の後からデータの長さ（図「非 FIX 表の制御部の出力形式」の L）分まで，データの長さ列のデータ値，及びコンストラクタパラメタ逆生成関数が返却したデータ値を出力します。なお，該当列のオフセット値が 0（ナル値）の場合は出力しません。また，各列のデータ値及びコンストラクタパラメタ逆生成関数が返却したデータ値の出力順は保障しません（各データ値の先頭位置はオフセット値で参照できます）。基本データ部に出力する，各列及びコンストラクタパラメタ逆生成関数が返却したデータ値を次の表に示します。

表 8–3　基本データ部に出力する値

列の属性	列のデータ型	コンストラクタパラメタ逆生成関数が返却するデータ型	基本データ部に出力する値
繰返し列	−	−	※
上記以外	抽象データ型	BINARY 型	コンストラクタパラメタ逆生成関数が返却したデータ値の長さ

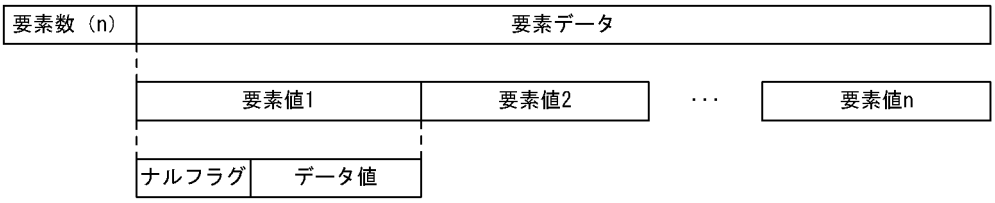
列の属性	列のデータ型	コンストラクタパラメタ逆生成関数が返却するデータ型	基本データ部に出力する値
		BLOB 型	コンストラクタパラメタ逆生成関数が返却したデータ値の長さ
		上記以外	コンストラクタパラメタ逆生成関数が返却したデータ値
	BINARY 型	−	列のデータ値の長さ
	BLOB 型	−	列のデータ値の長さ
	上記以外	−	列のデータ値

（凡例）−：該当しません。

注※

繰返し列の場合，基本データ部に出力する値は要素数，要素データで構成し，バウンダリしないで詰めて出力します。繰返し列の場合の基本データ部の出力形式を次の図に示します。

図 8–17　繰返し列の場合の基本データ部の出力形式



- 要素数

  繰返し列のデータ中の要素の数を 2 進数（2 バイト）で 1〜30,000 の範囲で出力します。pdload は要素数が 0 の場合はエラーデータとして扱うため，要素数を 0 にするのではなく，オフセット値を 0（ナル値）としてください。

- 要素データ

  要素値（ナルフラグとデータ値）を要素ごとに，データベース格納順にバウンダリしないで詰めて出力します。なお，要素数が 0 の場合は出力しません。

- 要素値

  ナルフラグとして，先頭 1 バイトに該当する要素がナル値かどうかを示す値を 2 進数（1 バイト）で出力します。ナル値の場合は「0x01」を，ナル値以外の場合は「0x00」を出力します。

  ナルフラグに続けてデータ値をデータ値の長さ分，バウンダリしないで出力します。

BINARY 属性データ部及び BLOB 属性データ部

抽象データ型の列を含む表の場合，その抽象データ型に対して unld_func 文に指定したコンストラクタパラメタ逆生成関数が返却するデータ型が BINARY 型又は BLOB 型のとき，返却した BINARY データ値又は BLOB データ値（長さを含まない）を出力します。unld_func 文にコンストラクタパラメタ逆生成関数を指定していないときは，BINARY 属性データ部又は BLOB 属性データ部は存在しません。

BINARY 属性データ部又は BLOB 属性データ部は，列定義順及びコンストラクタパラメタ逆生成関数関数の指定順に，バウンダリしないで詰めて，基本データ部に出力した長さ分出力します。コンストラクタパラメタ逆生成関数が返却する値がナル値のときはそのデータ値は出力しません。

次に示す例の場合の，BINARY 属性データ部の出力形式を図「BINARY 属性データ部の出力形式（例）」に示します。

（例）

```
表定義：

CREATE  TABLE T1(c1 ADT1,C2 INTEGER,C3  ADT2)
```

ADT1：抽象データ型（ADT1）

ADT2：抽象データ型（ADT2）

```
unld_func文の指定：

unld_func type=ADT1,func=adt1_func1(ADT1),adt1_func2(ADT1),adt1_func3(ADT1)
unld_func type=ADT2,func=adt2_func1(ADT2),adt2_func2(ADT2),adt2_func3(ADT2)
```

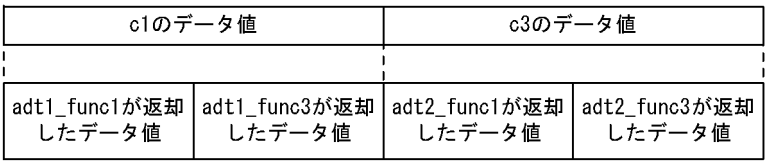
adt1_func1/adt1_func3：BINARY 型のデータを返却するコンストラクタパラメタ逆生成関数

adt1_func2：INTEGER 型のデータを返却するコンストラクタパラメタ逆生成関数

adt2_func1/adt2_func3：BINARY 型のデータを返却するコンストラクタパラメタ逆生成関数

adt2_func2：INTEGER 型のデータを返却するコンストラクタパラメタ逆生成関数

図 8–18　BINARY 属性データ部の出力形式（例）



BINARY 列データ部及び BLOB 列データ部

BINARY 型又は BLOB 型の列が含まれる表の場合，BINARY 型又は BLOB 型の列のデータ値（長さを含まない）を出力します。BINARY 型の列がない表のときは BINARY 列データ部は存在しません。また，BLOB 型の列がない表のとき，及びコマンドラインに-j オプションを指定していないときは BLOB 列データ部は存在しません。

BINARY 型又は BLOB 型の列のデータ値は，列定義順にバウンダリしないで詰めて，基本データ部に出力した長さ分出力します。なお，該当する列の値がナル値のときはそのデータ値は出力しません。

(3)　固定長データ形式の場合

固定長データ形式の場合，データは固定長の文字データで出力します。

(a)　出力形式

- **出力単位**

  データベース中の 1 行に対して，1 行のデータを出力します。

- **1 行の出力形式**
  - 1 行の長さ

    固定長で出力します。
  - 列データの出力順序

    列定義順に出力します。
  - 列間のセパレータ

    列データと列データの間には，セパレータ文字を出力しません。
  - 改行

    オプションを指定すれば，1 行の終端に改行文字を入れられます。
- **各列の出力形式**

  各列のデータの出力形式については，表「各列のデータの出力形式」を参照してください。
- **繰返し列の出力形式**

  各要素のデータを，定義されている要素数分繰り返して出力します。各要素のデータの出力形式については，表「各列のデータの出力形式」を参照してください。
- **特殊な列データの出力形式**
  - 列データがナル値

    列データの出力長分のパディング文字を埋めて出力します。繰返し列の場合，（列データの出力長×定義されている要素数）分のパディング文字を埋めて出力します。
  - 繰返し列の要素がナル値

    列データの出力長分のパディング文字を埋めて出力します。
  - 列データ中の制御文字

    文字データ型，混在文字データ型，及び各国文字データ型の場合，制御文字があったときの動作を fixtext_option 文で指定できます。デフォルトは，制御文字があるかどうかをチェックしません。

表 8–4　各列のデータの出力形式

列のデータ型		出力形式	出力長 (単位：バイト)※1
数データ	INTEGER	• **fixtext_option 文で format=integer,type1 を指定した場合，又は fixtext_option 文を省略した場合** ・数値を右詰めで出力します。 ・先頭 1 バイトを符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 ・空いた部分は"0"で埋めて出力します。 （例） △0000000001 -0000000001 注　△は空白文字です。	11

列のデータ型		出力形式	出力長(単位：バイト)※1
		• **fixtext_option 文で format=integer,type2 を指定した場合** ・数値を右詰めで出力します。 ・数値の左隣を符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 ・空いた部分は空白文字(0x20)で埋めて出力します。 (例) △△△△△△△△△2 △△△△△△△△△-2 注　△は空白文字です。	
	SMALLINT	• **fixtext_option 文で format=smallint,type1 を指定した場合，又は fixtext_option 文を省略した場合** ・数値を右詰めで出力します。 ・先頭 1 バイトを符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 ・空いた部分は"0"で埋めて出力します。 (例) △00003 -00003 注　△は空白文字です。 • **fixtext_option 文で format=smallint,type2 を指定した場合** ・数値を右詰めで出力します。 ・数値の左隣を符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 ・空いた部分は空白文字(0x20)で埋めて出力します。 (例) △△△△△3 △△△△-3 注　△は空白文字です。	6
	DECIMAL	• 整数.小数の形式で，数値を右詰めで出力します。 • 先頭 1 バイトを符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 • 空いた部分は"0"で埋めて出力します。 (例 1) DEC(6,2) △0003.14 -0003.14 (例 2) DEC(6,0) △000314. -000314.	けた数＋ 2

列のデータ型		出力形式	出力長(単位：バイト)※1
		(例 3) DEC(6,6) △.000314 -.000314 注　△は空白文字です。	
	FLOAT	• 仮数部 E 指数部の形式で，左詰めで出力します。 • 先頭 1 バイトを符号部とし，プラスの値の場合は"+"，マイナスの値の場合は"-"を出力します。 • 空いた部分は空白文字(0x20)で埋めて出力します。 (例 1) 20E10 +2.000000000000000E+011 (例 2) -30E222 -3.000000000000000E+223 注　△は空白文字です。	23
	SMALLFLT	• 仮数部 E 指数部の形式で，左詰めで出力します。 • 先頭 1 バイトを符号部とし，プラスの値の場合は"+"，マイナスの値の場合は"-"を出力します。 • 空いた部分は空白文字(0x20)で埋めて出力します。 (例) 20E10 +2.000000000000000E+011 注　△は空白文字です。	23
文字データ	CHARACTER	• 左詰めで定義長分出力します。 • fixtext_option 文の enclose オペランドで囲み文字の付加有無を指定できます。 (例 1) enclose 指定なし，定義長：8 AIKO△△△△ (例 2) enclose="，定義長：8 "AIKO△△△△" 注　△は空白文字です。	条件によって変わります※2
	VARCHAR	• 左詰めで実長分出力します。 • fixtext_option 文の enclose オペランドで囲み文字の付加有無を指定できます。 • -W fixtext オプションを指定すると，定義長より短い可変長文字列の場合は，定義長までパディング文字を埋めて出力します。	条件によって変わります※2

列のデータ型		出力形式	出力長(単位：バイト)※1
		(例 1) enclose 指定なし，定義長：8，実長：4 AIKO△△△△ (例 2) enclose="，定義長：8，実長：4 "AIKO"△△△△ 注　△はパディング文字です。	
混在文字データ	MCHAR	CHARACTER と同じです。	定義長※3
	MVARCHAR	VARCHAR と同じです。	定義長※3
各国文字データ	NCHAR	CHARACTER と同じです。	定義長×2※3
	NVARCHAR	VARCHAR と同じです。	定義長×2※3
日付データ	DATE	yyyy-mm-dd の形式で出力します（yyyy：西暦，mm：月，dd：日)。 (例) 2004-03-12	10
時刻データ	TIME	hh:mm:ss の形式で出力します（hh：時，mm：分，ss：秒)。 (例) 12:12:12	8
日間隔データ	INTERVAL YEAR TO DAY	• [△｜-] yyyymmdd.の形式で出力します（△：空白文字，-：マイナス符号，yyyy：西暦，mm：月，dd：日)。 • 先頭 1 バイトを符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 (例) △00010101. -00010101. 注　△は空白文字です。	10
時間隔データ	INTERVAL HOUR TO SECOND	• [△｜-] hhmmss.の形式で出力します（△：空白文字，-：マイナス符号，hh：時，mm：分，ss：秒)。 • 先頭 1 バイトを符号部とし，プラスの値の場合は空白文字(0x20)，マイナスの値の場合は"-"を出力します。 (例) △010101. -010101. 注　△は空白文字です。	8

列のデータ型		出力形式	出力長 (単位：バイト)※1
時刻印データ	TIMESTAMP	yyyy-mm-dd△hh:mm:ss[.nnnnnn]の形式で出力します（yyyy：西暦，mm：月，dd：日，△：空白文字，hh：時，mm：分，ss：秒，nnnnnn：小数秒部）。なお，小数秒部は，定義によって 0，2，4，又は 6 けたで出力します。0 けたの場合，'.'は出力しません。 （例） 1970-03-12△12:12:12 注　△は空白文字です。	19 小数秒部 0：+0 2：+3 4：+5 6：+7
長大データ	BLOB	データを出力しません。	0
バイナリデータ	BINARY	データを出力しません。	0
抽象データ型	–	データを出力しません。	0

（凡例）

–：該当しません。

注※1

アンロードデータファイルを HiRDB の既定文字コード以外の文字コードで出力する場合，2 倍のバイト長で出力します。

注※2

文字データ型の出力長を次に示します。

項番	アンロードデータファイルの文字コード	列の文字集合	enclose オペランド指定	出力長（バイト）
1	HiRDB の既定文字コード	UTF-16	なし	定義長×2
2			あり	定義長×2 ＋ 2
3		上記以外（指定なしも含む）	なし	定義長
4			あり	定義長＋ 2
5	上記以外	UTF-16	なし	定義長
6			あり	定義長＋ 2×2
7		上記以外（指定なしも含む）	なし	定義長×2
8			あり	定義長×2 ＋ 2×2

注※3

fixtext_option 文に enclose オペランドを指定した場合，出力長に 2 バイトを加算してください。

(b)　規則

1. 1 行（改行含む）は最大 512 メガバイト（536,870,912 バイト）まで出力できます。

2. 抽象データ型列がある表の場合，unld_func 文を指定する必要があります。

3. FLOAT 型又は SMALLFLT 型のデータを文字形式に変換した場合，データ値が切り上げられることがあります。データ値が切り上げられると，データベース作成ユティリティ（pdload）でデータ変換エラーとなる場合があります。その場合は，pdload のオプションに-F を指定することでデータロードができます。

4. 予備列が定義された FIX 表の場合，予備列は出力できません。

8.4　表へのリロード

8.4.1　使用例

リロードの例については，「使用例」の例 2～例 5 を参照してください。

8.4.2　目的別の参照先一覧

ユーザが実行するリロードの形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)～(4)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」を参照してください。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k reld	－
ユーザが定義した表のアンロード（省略時仮定値が-c user なので省略できます）	-c user	－
リロードする表の名称	-t	－
制御文を記述したファイル	制御情報ファイル名	－
アンロードデータファイルの情報	－	unload 文

（凡例）－：該当しません。

(2)　リロードする表の属性によって指定する項目

項目	参照先	
	オプション	制御文
インデクスが定義されている場合	-i	index 文
	-o	idxwork 文
		sort 文
LOB 列がある場合	-j	lobunld 文
プラグインが提供する抽象データ型列（LOB 属性）がある場合	-j	reld_func 文

項目	参照先	
	オプション	制御文
制約定義がある場合	−	constraint 文

（凡例）−：該当しません。

(3) アンロードデータファイル，又は LOB データのアンロードファイルの種別によって指定する項目

項目	参照先	
	オプション	制御文
ファイルを一本化している場合（アンロード時に-g オプションを指定している場合）	-g	−
HiRDB ファイルを使用している場合	-f	−

（凡例）−：該当しません。

(4) リロードの方法によって指定する項目

項目	参照先	
	オプション	制御文
横分割表の場合で，特定の RD エリアにだけリロードしたい場合	-r	−
データベース更新ログを取得する場合，又は取得しない場合	-l	−
大量データのリロードの場合に，同期点を指定したいとき	−	option 文
pdrorg の実行時間を監視したい場合	−	option 文
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※1	-u	−
グローバルバッファを使用しないで，ローカルバッファを使って一括入出力のリロードをしたい場合※2	-n	−
リロードの実行状態のメッセージ出力間隔を，10万行単位以外にしたい場合	-m	−
インフォメーションメッセージの出力を抑止したい場合		
サーバ間通信の応答時間を監視したい場合	-X	−

項目	参照先	
	オプション	制御文
表データを別表に移行したい場合	−	tblname 文
表データ中の空白文字コードを統一したい場合	−	option 文
リロード時に，表，及びインデクスの空き領域比率を変更したい場合※3	−	option 文

（凡例）−：該当しません。

注※1

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

注※2

一括入出力ページ数を指定すると，指定したページ数単位で一括入出力をするので，I/O 回数を削減できます。

注※3

pdrorg 実行時に RD エリアの容量不足が発生した場合，暫定的に RD エリアを拡張しないで処理を完了できます（ただし，空き領域比率が 0 の表，及びインデクスを除く）。

8.4.3 アンロード元の表とリロード先の表の表定義が異なる場合のリロード実行可否

pdrorg では，アンロードした表の表定義情報をアンロードデータファイルに出力しています。リロード時は，アンロードデータファイル内の表定義情報とリロードする表の表定義情報を比較し，リロードできるかどうかを判定しています。

アンロードした表に対するリロードは，表定義が同じ場合はリロードができます。しかし，別表，若しくは別システムへの表データの移行，又は表定義が異なる表へのリロードは，条件によってリロードができない場合があります。

(1) リロード時に指定するオプション

別表，若しくは別システムへ表データを移行する場合，又はアンロードした表とリロードする表の格納条件が異なる表へのリロードをする場合，次のようにオプションを指定してください。

- アンロード時，及びリロード時に-g オプションを指定してください。
- BLOB 列，又は BLOB 属性の抽象データ型列がある表の場合，-g オプションと同時に-j オプションも指定してください。

(2)　アンロードした表とリロードする表の格納条件の比較

アンロードした表とリロードする表の分割方法が同じ場合は，次の表の項目をチェックし，格納条件が異なるかどうかを判定します。格納条件が異なる場合，表「格納条件が異なる場合のリロード可否」に従って，リロード可否が決まります（表単位又はスキーマ単位にリロードする場合の実行可否です。RD エリア単位にリロードする場合はすべての条件で実行できません）。

表 8–5　アンロードした表とリロードする表の分割方法が同じ場合の格納条件チェック項目

分割方法	チェック項目	
非分割	なし	
ハッシュ分割	ハッシュ関数	関数名
	ハッシュ種別	FIX 又はフレキシブル
	分割キーの列構成	定義順
	格納 RD エリア	格納 RD エリア数，RD エリア ID，RD エリア名，及びサーバ名
キーレンジ	分割キーの列構成	定義順，及び列 ID
	分割キーの先頭列	列 ID，データ型，及び定義長
	分割条件数	
	分割条件	格納 RD エリア ID，及び条件値
マトリクス分割	分割キーの列構成	定義順
	分割条件数	
	分割条件値数	
	分割キーの条件	キー内分割数，次元番号，分割列 ID，データ型，定義長，条件値長，及び条件値
	ハッシュ関数	関数名
	ハッシュ種別	FIX 又はフレキシブル
	格納 RD エリア	格納 RD エリア数，RD エリア ID，RD エリア名，及びサーバ名

表 8–6　格納条件が異なる場合のリロード可否

-g 指定※1	HiRDBの構成	アンロードした表の分割条件		リロードする表の分割条件					
				キーレンジ分割	ハッシュ分割		マトリクス分割		非分割
					FIX	フレキシブル	キーレンジとキーレンジ	キーレンジとハッシュ	
あり	HiRDB/シングルサーバ	キーレンジ分割		○	○	○	○	○	○
		ハッシュ分割	FIX	○	○	○	○	○	○
			フレキシブル	○	○	○	○	○	○
		マトリクス分割	キーレンジとキーレンジ	○	○	○	○	○	○
			キーレンジとハッシュ	○	○	○	○	○	○
		非分割		○	○	○	○	○	○
	HiRDB/パラレルサーバ	キーレンジ分割		○	○	○	○	○	○
		ハッシュ分割	FIX	○	○	○	○	○	○
			フレキシブル	○	○	○	○	○	○
		マトリクス分割	キーレンジとキーレンジ	○	○	○	○	○	○
			キーレンジとハッシュ	○	○	○	○	○	○
		非分割		○	○	○	○	○	○
なし	HiRDB/シングルサーバ	キーレンジ分割		○	○	○	○	○	○
		ハッシュ分割	FIX	×	○	○	○	○	○
			フレキシブル	×	○	○	○	○	○
		マトリクス分割	キーレンジとキーレンジ	○	○	○	○	○	○
			キーレンジとハッシュ	○	○	○	○	○	○
		非分割		○	○	○	○	○	○
	HiRDB/パラレルサーバ	キーレンジ分割		○※6	×	○※2※3	×	×	○※5
		ハッシュ分割	FIX	×	×	○※2※4	×	×	○※5
			フレキシブル	×	×	○※7	×	×	○※5

-g 指定※1	HiRDB の構成	アンロードした表の分割条件		リロードする表の分割条件					
				キーレンジ分割	ハッシュ分割		マトリクス分割		非分割
					FIX	フレキシブル	キーレンジとキーレンジ	キーレンジとハッシュ	
		マトリクス分割	キーレンジとキーレンジ	×	×	○※2※3	○※6	×	○※5
			キーレンジとハッシュ	×	×	○※2※3	×	×	○※5
		非分割		×	×	○※2※3	×	×	○※2

（凡例）

○：リロードする表の格納条件に従って格納できます。

×：リロードできません。

注※1

アンロード時，及びリロード時の-g オプションの指定有無を示しています。なお，-g オプションを指定しない場合の分割表から非分割表へのリロードの意味は，サーバ間分割していない分割表から非分割表へのリロードを指します。サーバ間分割している場合は-g オプションを指定してください。

注※2

BLOB 列，又は BLOB 属性の抽象データ型列がある表の場合，アンロード時，及びリロード時に-j オプションを指定しているときだけリロードできます。

注※3

ハッシュ関数を使用しないで，pdrorg がラップアラウンドで格納します。

注※4

アンロード時に格納されていた RD エリアと同じ RD エリアに格納します。アンロード時の RD エリアがリロード時にない場合は，リロードできません。

注※5

BLOB 列，又は BLOB 属性の抽象データ型列がある表の場合はリロードできません。

注※6

表格納条件の変更の内容によって，リロード可否が決まります。リロード可否を次に示します。

表格納条件の変更の内容		リロード可否	備考
分割条件変更あり	分割条件追加	×	エラー終了します。
	分割条件削除		
	分割キー値の変更		

表格納条件の変更の内容			リロード可否	備考
分割条件変更なし	格納 RD エリア変更あり	RD エリア名変更	○	注※2，及び注※5 の条件を満たす場合，アンロード時の格納 RD エリアに対応する RD エリアに格納します。
		RD エリアがあるサーバの変更	×	エラー終了します。
	格納 RD エリア変更なし	分割キーの構成列変更（列名，指定順序など）	×	エラー終了します。

（凡例）

○：リロードできます。

×：リロードできません。

注※7

表格納条件の変更の内容によって，リロード可否が決まります。リロード可否を次に示します。

表格納条件の変更の内容		リロード可否	備考
格納 RD エリア変更あり	RD エリアの追加	○	注※4，及び注※5 の条件を満たす場合，アンロード時の格納 RD エリアに格納します。追加した RD エリアには格納されません。
	RD エリアの削除	×	エラー終了します。
	RD エリアの名称変更		
	RD エリアがあるサーバの変更		
格納 RD エリア変更なし	ハッシュ関数変更	○	注※4，及び注※5 の条件を満たす場合，アンロード時の格納 RD エリアに格納します。
	分割キーの構成列変更（列名，指定順序など）		

（凡例）

○：リロードできます。

×：リロードできません。

(3)　リロードできないときの条件

次のような場合にはリロードできません。

1. LOB 列がある表で-j オプションを指定していない場合
2. FIX 表から非 FIX 表，又は非 FIX 表から FIX 表へのリロード
3. アンロードした表とリロードする表の列数，列定義順序，各列のデータ型，列名，及び列属性（非ナル値制約など）が異なる場合

4. 非ナル値制約の列がある表をアンロードし，リロード先では分割キー構成列とした場合（非ナル値制約でない列を分割キー構成列に変更した場合は，-W オプションを指定してアンロードし，データベース作成ユティリティでデータロードしてください）
5. 値が重複するデータの列に，新たにユニークキーインデクス又は主キーインデクスを定義した場合
6. リロードする表の繰返し列ごとの要素数が，アンロードした表よりも小さい場合
7. 抽象データ型が定義されている列のデータ型を変更した場合

8.4.4　表の再編成時のリロード処理でエラーとなった場合の運用方法

表の再編成時のリロード処理でエラーとなった場合，エラー原因を取り除いてから再度表の再編成（-k rorg）を実行します。

(1)　表の再編成時にエラーとなった処理と，再実行時の再開始位置

表の再編成でエラーとなった場合，表のステータスが「再編成中」となります。表のステータスが再編成中の場合，表の再編成を再実行すると，エラーとなった処理から再開されます。該当する RD エリアが再編成中かどうかは，pddbst の RD エリア単位の論理的解析，又は表単位の解析を実行して，解析結果の Status を見れば分かります。

(a)　LOB 列構成基表の再編成，並びに LOB 列構成基表，及び LOB 列を同時に再編成（-j 指定）する場合

表の再編成時にエラーとなった場合に再実行したときの再開始位置を次の表に示します。なお，LOB 列構成基表，及び LOB 列の同時再編成は，-j オプションを指定して一つのアンロードデータファイルを使用する場合について説明しています。

pdrorg 実行時のオプションの指定：

pdrorg -k rorg 〔-j〕

pdrorg 実行時の制御情報ファイルの指定：

unload 文

表 8–7　表の再編成時にエラーとなった場合に再実行したときの再開始位置（LOB 列構成基表の再編成，並びに LOB 列構成基表，及び LOB 列を同時に再編成（-j 指定）する場合）

項番	表の再編成時の処理		エラー直前の出力メッセージ※		表のステータス			再開始位置
					LOB 列構成基表	LOB 列	LOB 属性	
1	アンロード	アンロード開始	712，732		−	−	−	項番 1
2		アンロード完了	714，734	rc=8	−	−	−	項番 1
				rc=0	未かつ R	未かつ R	未かつ R	改竄防止表の場合：

項番	表の再編成時の処理		エラー直前の出力メッセージ※		表のステータス			再開始位置
					LOB 列構成基表	LOB 列	LOB 属性	
								項番 3
					R	R	R	改竄防止表でない場合： 項番 1（表単位，又は RD エリア単位） 項番 3（スキーマ単位）
3	リロード	データ削除開始	出力されません。		未かつ R	未かつ R	未かつ R	改竄防止表の場合： 項番 3
					R	R	R	改竄防止表でない場合： 項番 1（表単位，又は RD エリア単位） 項番 3（スキーマ単位）
4		データ削除完了	721		未かつ R	未かつ R	未かつ R	項番 5
5		リロード開始	712，732		未かつ R	未かつ R	未かつ R	項番 5
6		リロード完了	714，734	rc=8	未かつ R	未かつ R	未かつ R	項番 5
				rc=0	R	R	R	インデクスが定義されていて，かつ-i c の場合： 項番 7 上記以外の場合： 項番 1
7	インデクスの一括作成	インデクスのアンロード開始（再開始時だけ）	725		R	R	R	項番 7
8		インデクスの一括作成開始	715		R	R	R	項番 7
9		インデクスの一括作成完了	716		R	R	R	作成されていないインデクスがある場合： 項番 7
					−	−	−	上記以外の場合： 項番 1

（凡例）

未：データ未完状態

R：再編成中

rc：リターンコード

−：通常の状態（再編成前の状態，又は再編成後の状態）

注※

メッセージは省略して表記しています。例えば，712 は KFPL00712-I，732 は KFPL00732-I となります。

(b) LOB 列構成基表，及び LOB 列を同時に再編成（lobunld 文指定）する場合

表の再編成時にエラーとなった場合に再実行したときの再開始位置を次の表に示します。なお，LOB 列構成基表，及び LOB 列の同時再編成は，lobunld 文を指定してアンロードデータファイル，及び LOB データのアンロードデータファイルを使用する場合について説明しています。

pdrorg 実行時のオプションの指定：

pdrorg -k rorg

pdrorg 実行時の制御情報ファイルの指定：

unload 文及び lobunld 文

表 8–8 表の再編成時にエラーとなった場合に再実行したときの再開始位置（LOB 列構成基表，及び LOB 列を同時に再編成（lobunld 文指定）する場合）

項番	表の再編成時の処理		エラー直前の出力メッセージ※		表のステータス			再開始位置
					LOB 列構成基表	LOB 列	LOB 属性	
1	LOB 列構成基表のアンロード	アンロード開始	712，732		−	−	−	項番 1
2		アンロード完了	714，734	rc=8	−	−	−	項番 1
				rc=0	未かつ R	未かつ R	未かつ R	改竄防止表の場合： 項番 3
					R	R	R	改竄防止表でない場合： 項番 1
3	LOB 列構成基表へのリロード	データ削除開始	出力されません。		未かつ R	未かつ R	未かつ R	改竄防止表の場合： 項番 3
					R	R	R	改竄防止表でない場合： 項番 1
4		データ削除完了	721		未かつ R	未かつ R	未かつ R	項番 5
5		リロード開始	712，732		未かつ R	未かつ R	未かつ R	項番 5
6		リロード完了	714，734	rc=8	未かつ R	未かつ R	未かつ R	項番 5
				rc=0	R	R	R	インデクスが定義されていて，かつ-i c の場合： 項番 7 上記以外の場合： 項番 1

項番	表の再編成時の処理		エラー直前の出力メッセージ※		表のステータス			再開始位置
					LOB 列構成基表	LOB 列	LOB 属性	
7	インデクスの一括作成	インデクスのアンロード開始（再開始時だけ）	725		R	R	R	項番 7
8		インデクスの一括作成開始	715		R	R	R	項番 7
9		インデクスの一括作成完了	716		R	R	R	作成されていないインデクスがある場合： 項番 7 上記以外の場合： 項番 10
10	LOB 列のアンロード	アンロード開始	712，732		R	R	R	項番 10
11		アンロード完了	714，734	rc=8	R	R	R	項番 10
				rc=0	R	未かつ R	R	改竄防止表の場合： 項番 12
					R	R	R	改竄防止表でない場合： 項番 10
12	LOB 列へのリロード	データ削除開始	出力されません。		R	未かつ R	R	改竄防止表の場合： 項番 12
					R	R	R	改竄防止表でない場合： 項番 10
13		データ削除完了	721		R	未かつ R	R	項番 14
14		リロード開始	712，732		R	未かつ R	R	項番 14
15		リロード完了	714，734	rc=8	R	未かつ R	R	項番 14
				rc=0	−	−	−	項番 1

（凡例）

未：データ未完状態

R：再編成中

rc：リターンコード

−：通常の状態（再編成前の状態，又は再編成後の状態）

注※

メッセージは省略して表記しています。例えば，712 は KFPL00712-I，732 は KFPL00732-I となります。

(c)　LOB 列だけの再編成をする場合

表の再編成時にエラーとなった場合に再実行したときの再開始位置を次の表に示します。

pdrorg 実行時のオプションの指定：

pdrorg -k rorg

pdrorg 実行時の制御情報ファイルの指定：

lobunld 文

表 8–9　表の再編成時にエラーとなった場合に再実行したときの再開始位置（LOB 列だけを再編成する場合）

項番	表の再編成時の処理		エラー直前の出力メッセージ※		表のステータス			再開始位置
					LOB 列構成基表	LOB 列	LOB 属性	
1	LOB 列のアンロード	アンロード開始	712，732		−	−	−	項番 1
2		アンロード完了	714，734	rc=8	−	−	−	項番 1
				rc=0	−	未かつ R	−	改竄防止表の場合： 項番 3
					−	R	−	改竄防止表でない場合： 項番 1
3	LOB 列へのリロード	データ削除開始	出力されません。		−	未かつ R	−	改竄防止表の場合： 項番 3
					−	R	−	改竄防止表でない場合： 項番 1
4		データ削除完了	721		−	未かつ R	−	項番 5
5		リロード開始	712，732		−	未かつ R	−	項番 5
6		リロード完了	714，734	rc=8	−	未かつ R	−	項番 5
				rc=0	−	−	−	項番 1

（凡例）

未：データ未完状態

R：再編成中

rc：リターンコード

−：通常の状態（再編成前の状態，又は再編成後の状態）

注※

メッセージは省略して表記しています。例えば，712 は KFPL00712-I，732 は KFPL00732-I となります。

(d)　スキーマ単位の再編成をする場合

表の再編成時にエラーとなった場合に再実行したときの再開始位置を次の表に示します。

pdrorg 実行時のオプションの指定：

pdrorg -k rorg -t all

表 8–10　表の再編成時にエラーとなった場合に再実行したときの再開始位置（スキーマ単位の再編成をする場合）

項番	表の再編成時の処理	契機	再開始位置
1	アンロード	アンロード中にエラーとなったとき	項番 1
2	リロード	KFPL00722-I メッセージ出力以降にエラーとなったとき	項番 2

(e)　サーバごとに表の再編成（-g なし）をする場合（HiRDB/パラレルサーバ限定）

pdrorg 実行時のオプションの指定：

pdrorg -k rorg

pdrorg 実行時の制御情報ファイルの指定：

unload 文だけ，lobunld 文だけ，又は unload 文及び lobunld 文

HiRDB/パラレルサーバで，サーバ間分割表に対する表の再編成時のリロード処理でエラーとなったときは，制御情報ファイルの unload 文及び lobunld 文で指定したサーバごとに(a)～(c)の再開始位置から再実行します。

(2)　データ未完状態の表に対するコマンド及び SQL の実行可否

表の再編成時のリロード処理でエラーとなった場合，又は改竄防止表の再編成時にエラーなどによって再編成が完了していない場合，処理対象 RD エリアは「データ未完状態」となります。この場合，コマンド及び SQL の実行が一部制限されます。

データ未完状態の表に対するユティリティの実行可否と運用コマンドの実行可否と SQL の実行可否を次の表に示します。

表 8–11　データ未完状態の表に対するユティリティの実行可否

ユティリティ名	機能		表のステータス							
			基:–	基:未	基:–	基:–	基:未	基:未	基:–	基:未
			属:–	属:–	属:未	属:–	属:未	属:–	属:未	属:未
			L:–	L:–	L:–	L:未	L:–	L:未	L:未	L:未
pdload	LOB 列構成基表へのデータロード	-d あり	○	○	○	○	○	○	○	○

ユティリティ名	機能			表のステータス							
				基:−	基:未	基:−	基:−	基:未	基:未	基:−	基:未
				属:−	属:−	属:未	属:−	属:未	属:−	属:未	属:未
				L:−	L:−	L:−	L:未	L:−	L:未	L:未	L:未
			-d なし	○	×	×	×	×	×	×	×
	LOB 列へのデータロード			○	×	×	×	×	×	×	×
	LOB 列構成基表，及び LOB 列への同時データロード		-d あり	○	○	○	○	○	○	○	○
			-d なし	○	×	×	×	×	×	×	×
pdrorg	表の再編成(-k rorg)	-j あり	unload 文	○	△	△	△	△	△	△	△
		-j なし	unload 文	○	△	△	△	△	△	△	△
			lobunld 文	○	△	△	△	△	△	△	△
			unload 文及び lobunld 文	○	△	△	△	△	△	△	△
	表のアンロード(-k unld)			○	○	○	○	○	○	○	○
	表へのリロード(-k reld)	-j あり	unload 文	○	○	○	○	○	○	○	○
		-j なし	unload 文	○	○	○	○	○	○	○	○
			lobunld 文	○	○	○	○	○	○	○	○
			unload 文及び lobunld 文	○	○	○	○	○	○	○	○
	インデクスの一括作成(-k ixmk)		B-tree インデクス	○	×	×	×	×	×	×	×
			プラグインインデクス	○	×	×	×	×	×	×	×
	インデクスの再作成(-k ixrc)		B-tree インデクス	○	×	×	×	×	×	×	×
			プラグインインデクス	○	×	×	×	×	×	×	×
	インデクスの再編成(-k ixor)			○	×	×	×	×	×	×	×
pdrbal				○	×	×	×	×	×	×	×
上記以外のユティリティ				○	○	○	○	○	○	○	○

（凡例）

基：LOB 列構成基表が格納されている RD エリア

属：LOB 属性が格納されている RD エリア

L：LOB 列が格納されている RD エリア

未：データ未完状態

−：通常の状態（再編成前の状態，又は再編成後の状態）

○：実行できます。

△：オプション，及び制御情報ファイルを変更しない再実行はできます。オプション，及び制御情報ファイルを変更して再実行する場合は，実行エラーとなります。

×：実行できません。

表 8–12　データ未完状態の表に対する運用コマンドの実行可否

運用コマンド名	条件		表のステータス							
			基:−	基:未	基:−	基:−	基:未	基:未	基:−	基:未
			属:−	属:−	属:未	属:−	属:未	属:−	属:未	属:未
			L:−	L:−	L:−	L:未	L:−	L:未	L:未	L:未
pdrels	改竄防止表を格納する RD エリア	基	○	×	○	○	×	×	○	×
		属	○	○	×	○	×	○	×	×
		L	○	○	○	×	○	×	×	×
	上記以外の RD エリア		○	○	○	○	○	○	○	○
上記以外の運用コマンド			○	○	○	○	○	○	○	○

（凡例）

基：LOB 列構成基表が格納されている RD エリア

属：LOB 属性が格納されている RD エリア

L：LOB 列が格納されている RD エリア

未：データ未完状態

−：通常の状態（再編成前の状態，又は再編成後の状態）

○：実行できます。

×：実行できません。

表 8–13　データ未完状態の表に対する SQL の実行可否

SQL	表のステータス							
	基:−	基:未	基:−	基:−	基:未	基:未	基:−	基:未
	属:−	属:−	属:未	属:−	属:未	属:−	属:未	属:未
	L:−	L:−	L:−	L:未	L:−	L:未	L:未	L:未
SELECT 文	○	○	△	△	△	△	△	△
UPDATE 文	○	○	△	△	△	△	△	△
INSERT 文	○	○	△	△	△	△	△	△

SQL	表のステータス							
	基:−	基:未	基:−	基:−	基:未	基:未	基:−	基:未
	属:−	属:−	属:未	属:−	属:未	属:−	属:未	属:未
	L:−	L:−	L:−	L:未	L:−	L:未	L:未	L:未
DELETE 文	○	○	△	△	△	△	△	△
上記以外の SQL	○	○	○	○	○	○	○	○

（凡例）

基：LOB 列構成基表が格納されている RD エリア

属：LOB 属性が格納されている RD エリア

L：LOB 列が格納されている RD エリア

未：データ未完状態

−：通常の状態（再編成前の状態，又は再編成後の状態）

○：実行できます。

△：LOB 列又は LOB 属性の抽象データ型列を操作する場合，操作対象となる LOB 列格納 RD エリア又は LOB 属性格納 RD エリアがデータ未完状態のときは実行できません。

(3) データ未完状態の解除契機

データ未完状態は，次の処理をしたときに解除されます。

- PURGE TABLE 文を実行した場合
- pdmod で RD エリアの再初期化をした場合
- 表の再編成（-k rorg）で option 文に tblstatus=clear を指定して再実行し，pdrorg が正常終了した場合
- 表の再編成（-k rorg）を再実行し，pdrorg が正常終了した場合
- 表へのリロード（-k reld）を実行し，pdrorg が正常終了した場合
- pdload（-d 指定）を実行した場合

(4) 注意事項

1. 次の場合は，表の再編成（-k rorg）を再実行しても処理が再開されません。この場合，表へのリロード（-k reld）をするか，又はバックアップからの回復をしてください。
   - ログレスモード（-l n 指定）の場合
   - オプション，及び制御情報ファイルの指定を変更した場合（-b オプション，及び option 文の tblstatus オペランドを除く）
   - 再実行する前に，該当する表に対して次の操作をした場合（この場合，途中の処理からの再開はしないでアンロード処理から行います）

     ・-d 指定の pdload の実行

・表の再編成前に取得したバックアップから RD エリアを回復
・表へのリロード（-k reld）の実行
・pdmod での RD エリアの再初期化
・PURGE TABLE 文の実行

2. 該当する表の再編成が完了するまで，他ユーザからの pdrorg は実行しないでください（表の再編成の再実行と，通常の表の再編成，表へのリロードなどの同時実行はしないでください）。
3. インデクスが定義されている表の再編成時，インデクス更新モード（-i s）以外で再実行すると，インデクス情報ファイルが残ることがあります。この場合，表の再編成が完了した後にインデクス情報ファイルを削除してください。
4. スキーマ単位の再編成を再実行する場合，スキーマ内の表を格納している RD エリアを pdhold コマンドで閉塞するようにしてください（UAP が再編成中の表をアクセスして待ち状態にならないようにするため，及び UAP で表を更新しないようにするため）。また，スキーマ単位の再編成が完了するまでは，該当するスキーマへの表定義（CREATE TABLE），及び表削除（DROP TABLE）はしないでください。
5. -r オプションを指定して，かつ-b オプションに非分割キーインデクスを指定して再実行する場合，再開始位置が LOB 列構成基表のアンロード，又は LOB 列のアンロードとなるときは，再開始できません。この場合，-b オプションを削除して再実行するか，又は表へのリロード（-k reld）を実行してください。
6. インデクスが定義されている表の再編成（-k rorg）でインデクスの一括作成中に異常終了した場合，インデクス一括作成（-k ixmk）を実行して再編成を完了させると，データ未完状態が解除されません。このような運用をした場合，表の再編成（-k rorg）で option 文の tblstatus オペランドに clear を指定してから pdrorg を実行してデータ未完状態を解除してください。

8.5 ディクショナリ表の再編成

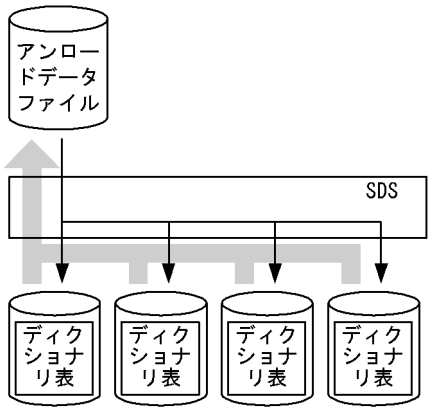
8.5.1 使用例

データベース再編成ユティリティの使用例（ディクショナリ表の再編成）を例 1 に示します。

例 1

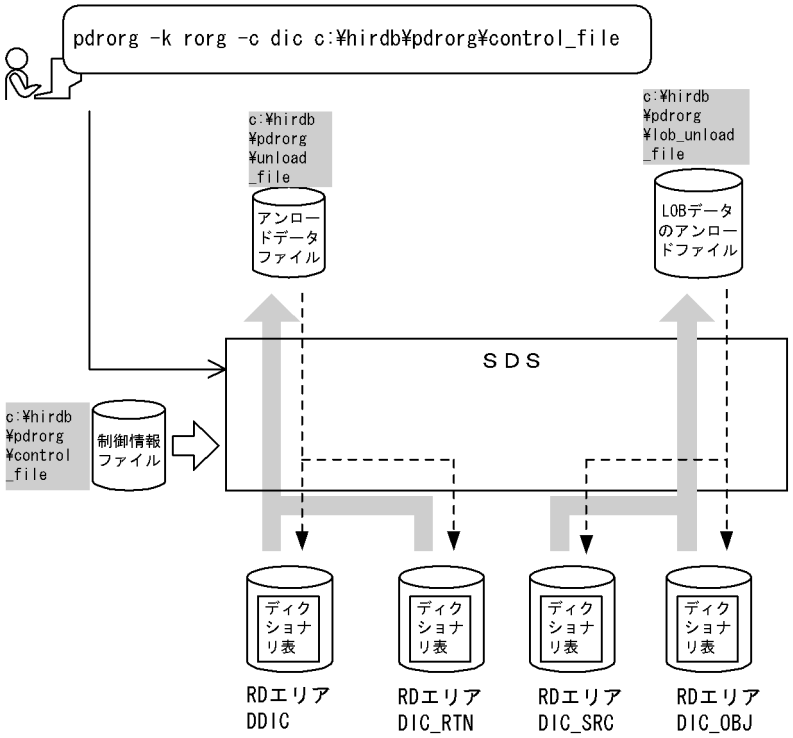
ディクショナリ表を表単位に再編成します。なお，ディクショナリ表は次に示すRDエリアに格納されているものとします。ディクショナリ表を再編成する場合，ディクショナリ表が格納されているデータディクショナリ用RDエリア又はデータディクショナリLOB用RDエリアを閉塞しておいてください。
・データディクショナリ用RDエリア（DDIC，DIC_RTN）
・データディクショナリLOB用RDエリア（RTN_SRC，RTN_OBJ）

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

ディクショナリ表を表単位に再編成します。

-k rorg：再編成をするときの指定

-c dic：ディクショナリ表を再編成するときの指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
unload c:¥hirdb¥pdrorg¥unload_file          1
lobunld c:¥hirdb¥pdrorg¥lob_unload_file     2
```

[説明]

1. アンロードデータファイルの指定

   c:¥hirdb¥pdrorg¥unload_file：アンロードデータファイルの名称

2. LOB データのアンロードファイルの指定

   c:¥hirdb¥pdrorg¥lob_unload_file：LOB データのアンロードファイルの名称

8.5.2 目的別の参照先一覧

ユーザが実行する再編成の形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)，及び(3)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」を参照してください。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k rorg	−
ディクショナリ表の再編成	-c dic	−
制御文を記述したファイル	制御情報ファイル名	−
アンロードデータファイルの情報	−	unload 文

（凡例）−：該当しません。

(2)　アンロードデータファイルの種別によって指定する項目

項目	参照先	
	オプション	制御文
HiRDB ファイルを使用する場合	-f	−

（凡例）−：該当しません。

(3)　再編成の方法によって指定する項目

項目	参照先	
	オプション	制御文
特定のディクショナリ表だけ再編成したい場合	-t	−
データベース更新ログを取得する場合，又は取得しない場合	-l	−
pdrorg の実行時間を監視したい場合	−	option 文
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※	-u	−
再編成の実行状態のメッセージ出力間隔を，10 万行単位以外にしたい場合	-m	−
インフォメーションメッセージの出力を抑止したい場合		
サーバ間通信の応答時間を監視したい場合	-X	−

（凡例）－：該当しません。

注※

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

8.6　インデクスの一括作成

8.6.1　使用例

データベース再編成ユティリティの使用例の一覧（インデクスの一括作成）を次に示します。

使用例	内容	区分
例 1	インデクスの一括作成 • 表の再編成の例 2 で出力されたインデクス情報ファイルを基に，インデクスを一括作成します。	S
例 2	プラグインインデクスの遅延一括作成 • 抽象データ型（SGMLTEXT 型）がある表に定義されている，プラグインインデクスの遅延一括作成をします。	
例 3	インデクスの一括作成 • データベース作成ユティリティ実行時（インデクス情報出力モード）に出力されたインデクス情報ファイルを使用するものとします。	P

（凡例）

S：HiRDB/シングルサーバ

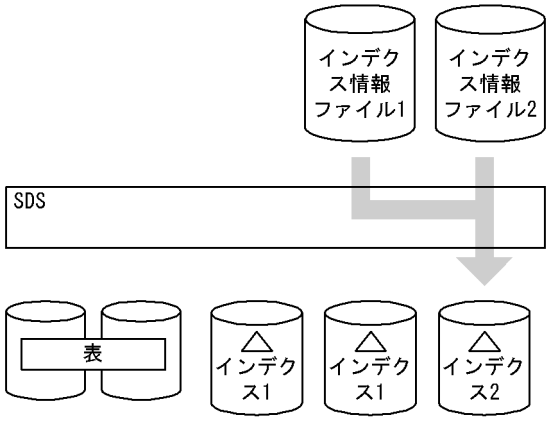
P：HiRDB/パラレルサーバ

(1)　インデクスの一括作成

例 1

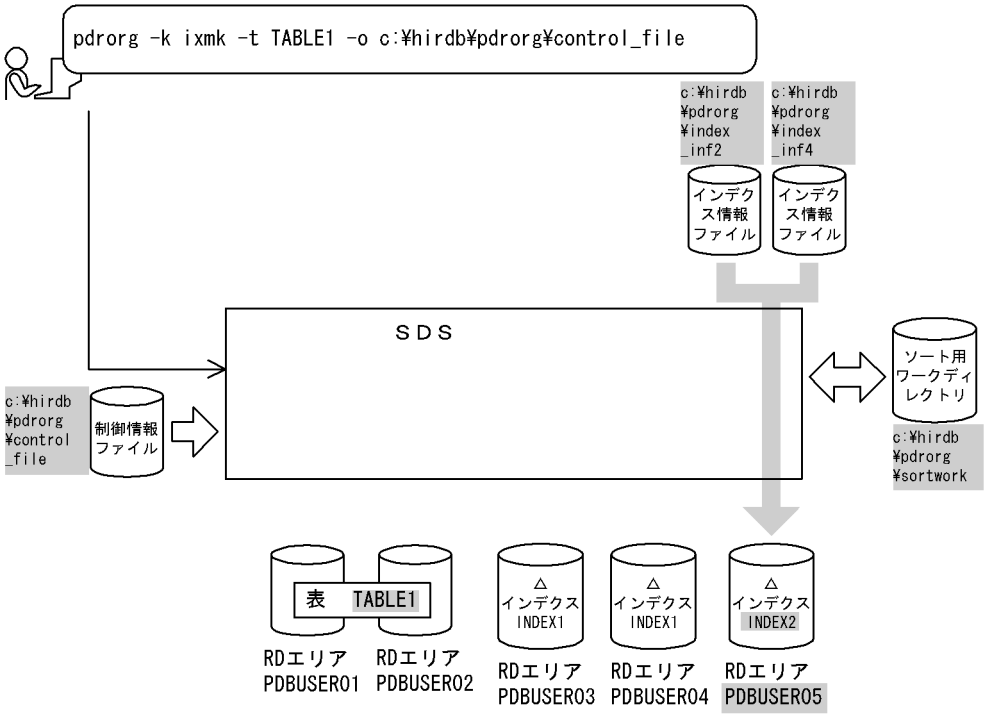
```
表の再編成の例2で出力された，インデクス情報を基にしてインデクス（INDEX2）を一括作成します。
出力されているインデクス情報ファイルの名称は，c:¥hirdb¥pdrorg¥index_inf2とします。
また，PDBUSER02のインデクス情報は既にc:¥hirdb¥pdrorg¥index_inf4に取得したものとします。
表（TABLE1），及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
         IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER04))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN (PDBUSER05)
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース再編成ユティリティでの指定値

ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

インデクス情報を基にして，インデクス（INDEX2）の一括作成をします。

-k ixmk：インデクスの一括作成をするときの指定

-t TABLE1：インデクスの一括作成対象の表の名称

-o：インデクス作成完了後，不要となったインデクス情報ファイルを削除する指定

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
index INDEX2 PDBUSER05 c:¥hirdb¥pdrorg¥index_inf2       1
index INDEX2 PDBUSER05 c:¥hirdb¥pdrorg¥index_inf4       1
sort c:¥hirdb¥pdrorg¥sortwork                           2
```

[説明]

1. インデクス情報があるインデクス情報ファイルの指定

   INDEX2：インデクス識別子

   PDBUSER05：インデクスが格納されている RD エリアの名称

   c:¥hirdb¥pdrorg¥index_inf2，c:¥hirdb¥pdrorg¥index_inf4：インデクス情報ファイルの名称

2. ソート用ワークディレクトリの指定

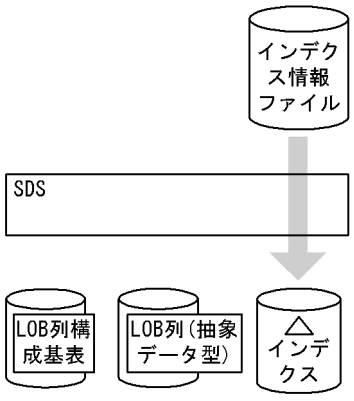
   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(2) プラグインインデクスの遅延一括作成

例２

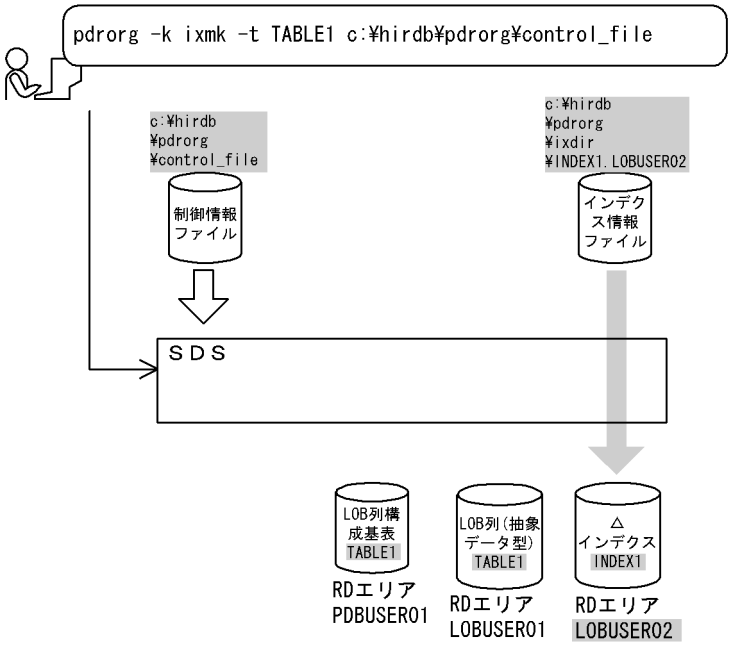
```
抽象データ型（SGMLTEXT）の列を持つ表（TABLE1）に定義されたプラグインインデクス（INDEX1）
を遅延一括作成します。抽象データ型（SGMLTEXT）はHiRDB Text Search Plug-inで提供されています。
なお，表及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT,C2 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER01)
                      PLUGIN'<DTD>sgml.dtd</DTD>' ) IN PDBUSER01
・プラグインインデクスの定義
  CREATE INDEX INDEX1 USING TYPE NGRAM ON TABLE1(C2) IN LOBUSER02
なお，インデクス情報ファイルは，次に示す定義内容のときに更新系（INSERT文やUPDATE文）のUAPを実行して作成されたものです。
・サーバ定義：set pd_plugin_ixmk_dir="c:¥hirdb¥pdrorg¥ixdir"
・クライアント環境変数：PDPLGIXMK=YES
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

表（TABLE1）に定義されたプラグインインデクス（INDEX1）を，遅延一括作成します。

-k ixmk：インデクスの一括作成をするときの指定

-t TABLE1：インデクスの一括作成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
index INDEX1 LOBUSER02 c:¥hirdb¥pdrorg¥ixdir¥INDEX1.LOBUSER02      1
```

[説明]

1. 遅延一括作成するインデクス情報ファイルの指定

   INDEX1：遅延一括作成するプラグインインデクスのインデクス識別子

   LOBUSER02：遅延一括作成するプラグインインデクスのインデクス格納 RD エリアの名称

   c:¥hirdb¥pdrorg¥ixdir¥INDEX1.LOBUSER02：インデクス情報ファイルの名称

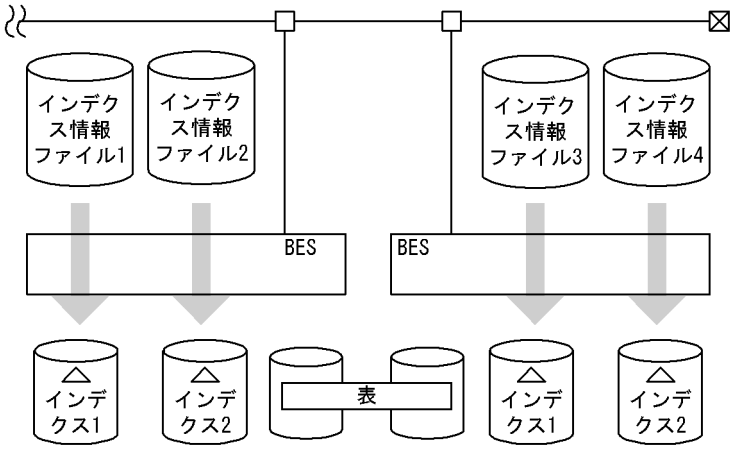
(3)　インデクスの一括作成

例３

```
インデクス情報ファイルを基にして，インデクスを一括作成します。データベース作成ユティリ
ティ実行時（インデクス情報出力モードでデータロード）に出力したインデクス情報ファイル4個を
使用するものとします。
表，及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
```

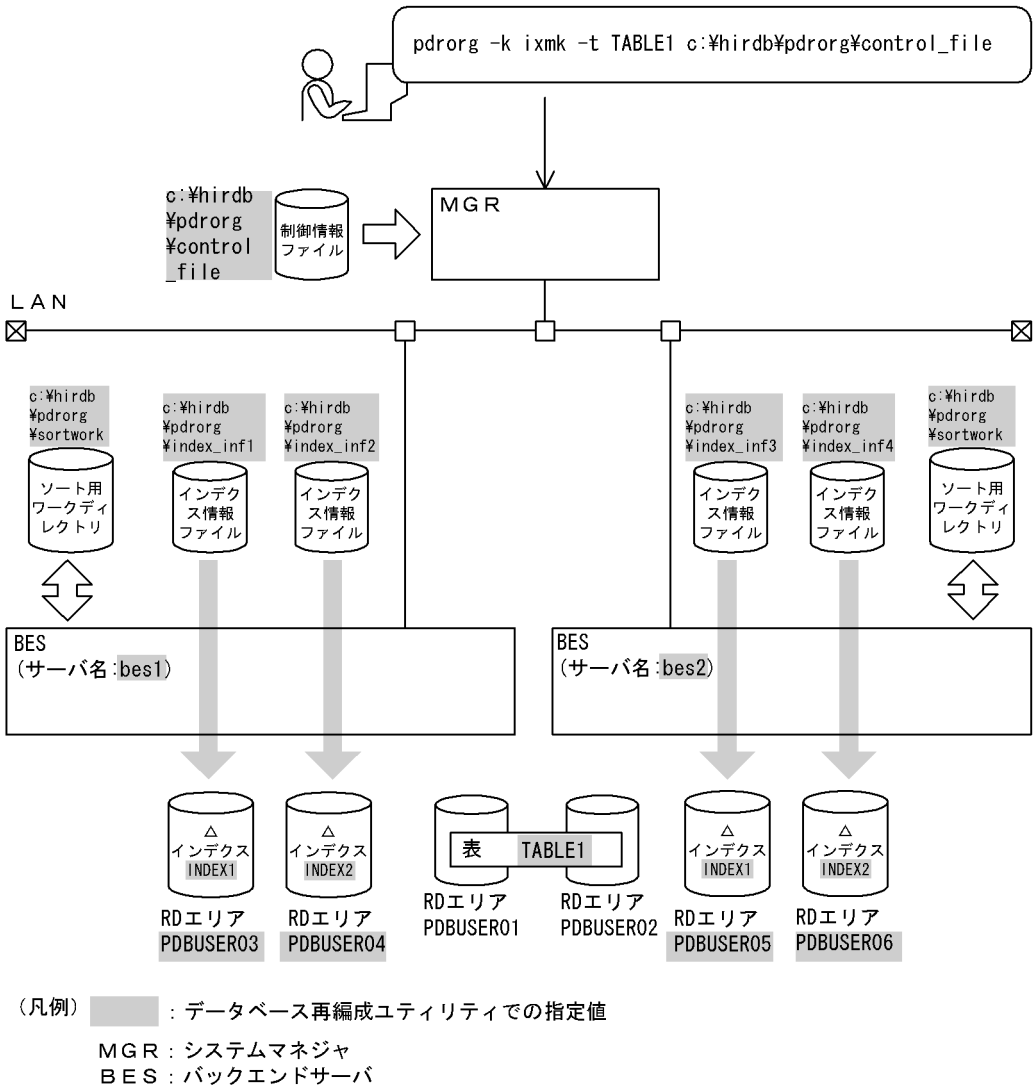
```
              IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

データロード時に出力されたインデクス情報を基にして，インデクスの一括作成をします。

-k ixmk：インデクスの一括作成をするときの指定

-t TABLE1：インデクスの一括作成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
index INDEX1 PDBUSER03 c:¥hirdb¥pdrorg¥index_inf1          1
index INDEX2 PDBUSER04 c:¥hirdb¥pdrorg¥index_inf2          1
index INDEX1 PDBUSER05 c:¥hirdb¥pdrorg¥index_inf3          1
index INDEX2 PDBUSER06 c:¥hirdb¥pdrorg¥index_inf4          1
sort bes1 c:¥hirdb¥pdrorg¥sortwork                         2
sort bes2 c:¥hirdb¥pdrorg¥sortwork                         2
```

[説明]

1. インデクス情報があるインデクス情報ファイルの指定

INDEX1，INDEX2：インデクス識別子

PDBUSER03，PDBUSER04，PDBUSER05，PDBUSER06：インデクスが格納されている RD エリアの名称

c:¥hirdb¥pdrorg¥index_inf1，c:¥hirdb¥pdrorg¥index_inf2，c:¥hirdb¥pdrorg¥index_inf3，c:¥hirdb¥pdrorg¥index_inf4：インデクス情報ファイルの名称

2. ソート用ワークディレクトリの指定

bes1，bes2：ソート用ワークファイルを作成するサーバの名称

c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

8.6.2　目的別の参照先一覧

ユーザが実行するインデクスの一括作成の形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」が参照先となります。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k ixmk	－
インデクスの一括作成をする表の名称	-t	－
制御文を記述したファイル	制御情報ファイル名	－
インデクス情報ファイルの情報	－	index 文

（凡例）－：該当しません。

(2)　インデクスの一括作成の方法によって指定する項目

項目	参照先	
	オプション	制御文
データベース更新ログを取得する場合，又は取得しない場合	-l	－
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※1	-u	－

項目	参照先	
	オプション	制御文
グローバルバッファを使用しないで，ローカルバッファを使って一括入出力のインデクスの一括作成をしたい場合※2	-n	−
インデクス作成後に，自動的にインデクス情報ファイルを削除したい場合	-o	−
インデクスの一括作成の実行状態のメッセージ出力間隔を，10 万行単位以外にしたい場合	-m	−
インフォメーションメッセージの出力を抑止したい場合		
サーバ間通信の応答時間を監視したい場合	-X	−
%PDDIR%¥tmp ディレクトリの空き容量に余裕がない場合	−	sort 文
pdrorg の実行時間を監視したい場合	−	option 文
インデクスの一括作成時に，インデクスの空き領域比率を変更したい場合※3	−	option 文

（凡例）−：該当しません。

注※1

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

注※2

一括入出力ページ数を指定すると，指定したページ数単位で一括入出力をするので，I/O 回数を削減できます。

注※3

pdrorg 実行時に RD エリアの容量不足が発生した場合，暫定的に RD エリアを拡張しないで処理を完了できます（ただし，空き領域比率が 0 のインデクスを除く）。

8.7　インデクスの再作成

8.7.1　使用例

データベース再編成ユティリティの使用例の一覧（インデクスの再作成）を次に示します。

使用例	内　　容	区分
例 1	プラグインインデクスの再作成 • 抽象データ型（SGMLTEXT 型）がある表に定義されている，プラグインインデクスの再作成をします。	S
例 2	インデクス格納 RD エリア単位のインデクスの再作成	P
例 3	インデクス単位のインデクスの再作成	

（凡例）

S：HiRDB/シングルサーバ

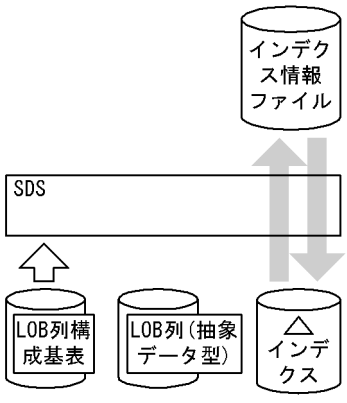
P：HiRDB/パラレルサーバ

(1)　プラグインインデクスの再作成

例１

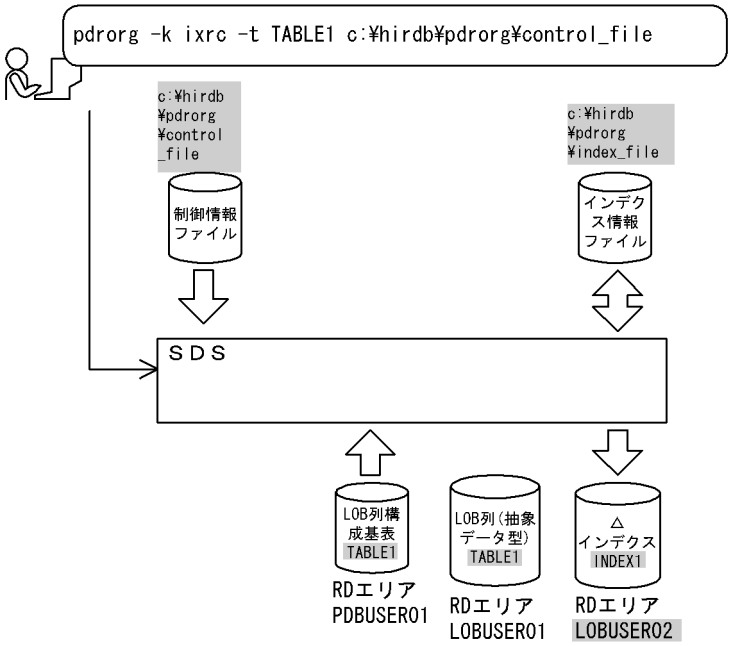
```
抽象データ型（SGMLTEXT）の列を持つ表（TABLE1）に定義されたプラグインインデクス（INDEX1）
を再作成します。抽象データ型（SGMLTEXT）はHiRDB Text Search Plug-inで提供されています。
表，及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT,C2 SGMLTEXT ALLOCATE (SGMLTEXT IN LOBUSER01)
                      PLUGIN'<DTD>sgml.dtd</DTD>' ) IN PDBUSER01
・プラグインインデクスの定義
  CREATE INDEX INDEX1 USING TYPE NGRAM ON TABLE1(C2) IN LOBUSER02
```

[概要]



（凡例）SDS：シングルサーバ

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＳＤＳ：シングルサーバ

[実行するコマンドの説明]

表（TABLE1）に定義されたプラグインインデクス（INDEX1）を再作成します。

-k ixrc：インデクスの再作成をするときの指定

-t TABLE1：インデクスの再作成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
index INDEX1 LOBUSER02 c:¥hirdb¥pdrorg¥index_file        1
```

[説明]

1. プラグインインデクス情報を出力するインデクス情報ファイルの指定

   INDEX1：再作成するプラグインインデクスのインデクス識別子

   LOBUSER02：再作成するプラグインインデクスのインデクス格納 RD エリアの名称

   c:¥hirdb¥pdrorg¥index_file：インデクス情報を出力するインデクス情報ファイルの名称

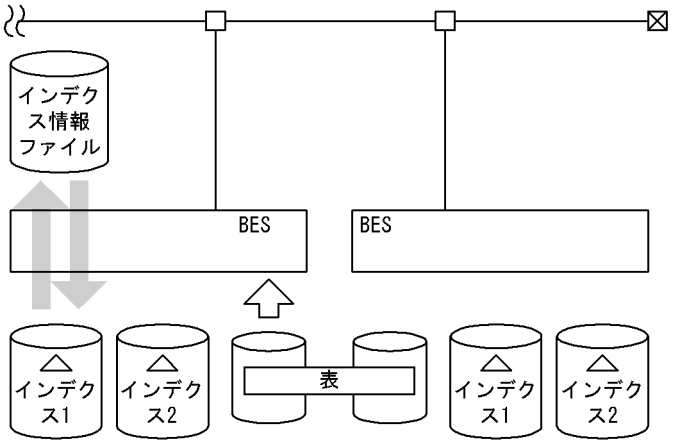
(2)　インデクス格納 RD エリア単位のインデクスの再作成

例２

```
インデクス（INDEX1）をRDエリア単位（PDBUSER03）に再作成します。
表，及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
```

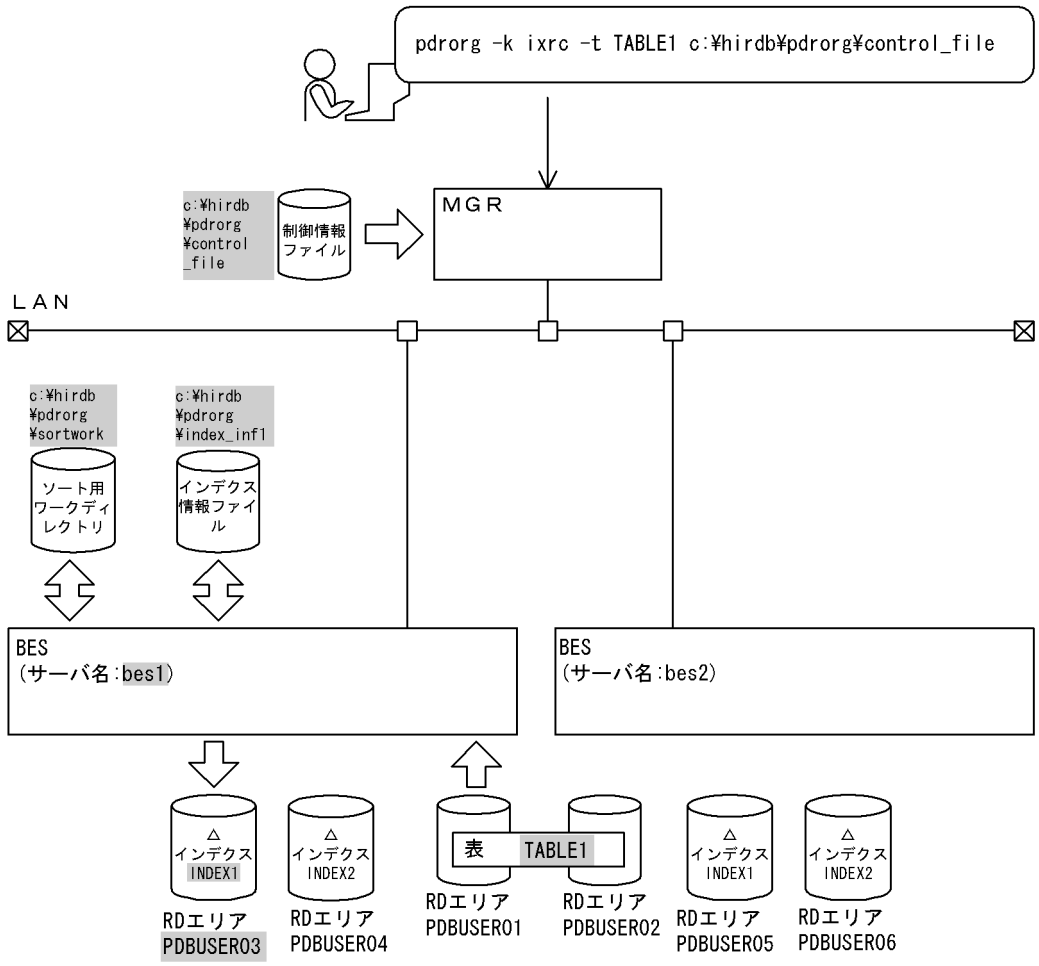
```
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



（凡例） ：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

インデクス（INDEX1，INDEX2）の再作成をします。

-k ixrc：インデクスの再作成をするときの指定

-t TABLE1：インデクスの再作成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
index INDEX1 PDBUSER03 c:¥hirdb¥pdrorg¥index_inf1     1
sort bes1 c:¥hirdb¥pdrorg¥sortwork                    2
```

[説明]

1. インデクス情報ファイルの指定

   INDEX1：インデクス識別子

   PDBUSER03：インデクスが格納されている RD エリアの名称

   c:¥hirdb¥pdrorg¥index_inf1：インデクス情報ファイルの名称

2. ソート用ワークディレクトリの指定

bes1：ソート用ワークファイルを作成するサーバの名称

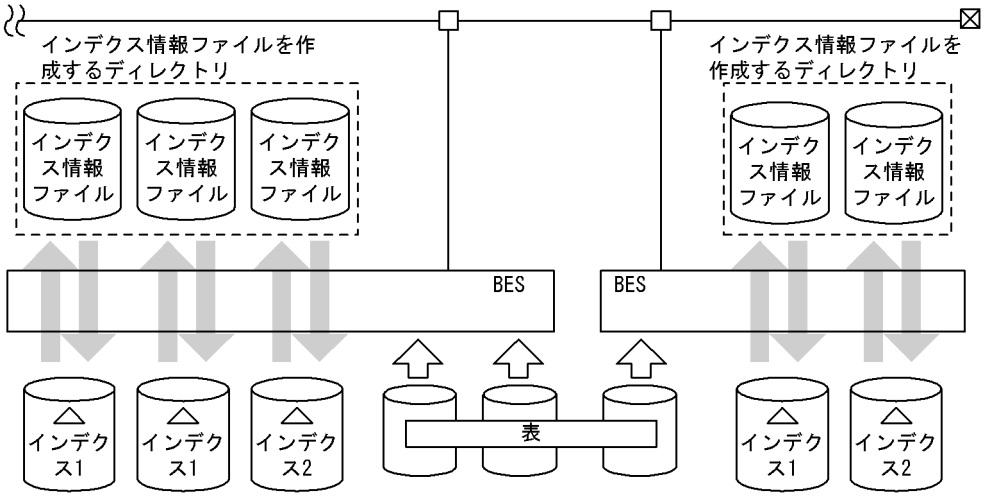
c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(3) インデクス単位のインデクスの再作成

例３

```
横分割表（TABLE1）に定義されたインデクス（INDEX1，INDEX2）を再作成します。
なお，次の表及びインデクスが定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT,C2 INT NOT NULL)
      IN ((PDBUSER01) C2<200,(PDBUSER02) C2<400,(PDBUSER03))
・インデクスの定義
  CREATE INDEX INDEX1 ON TABLE1(C2) IN ((IDX01),(IDX02),(IDX03))
  CREATE INDEX INDEX2 ON TABLE1(C1) IN ((IDX04),(IDX05))
```

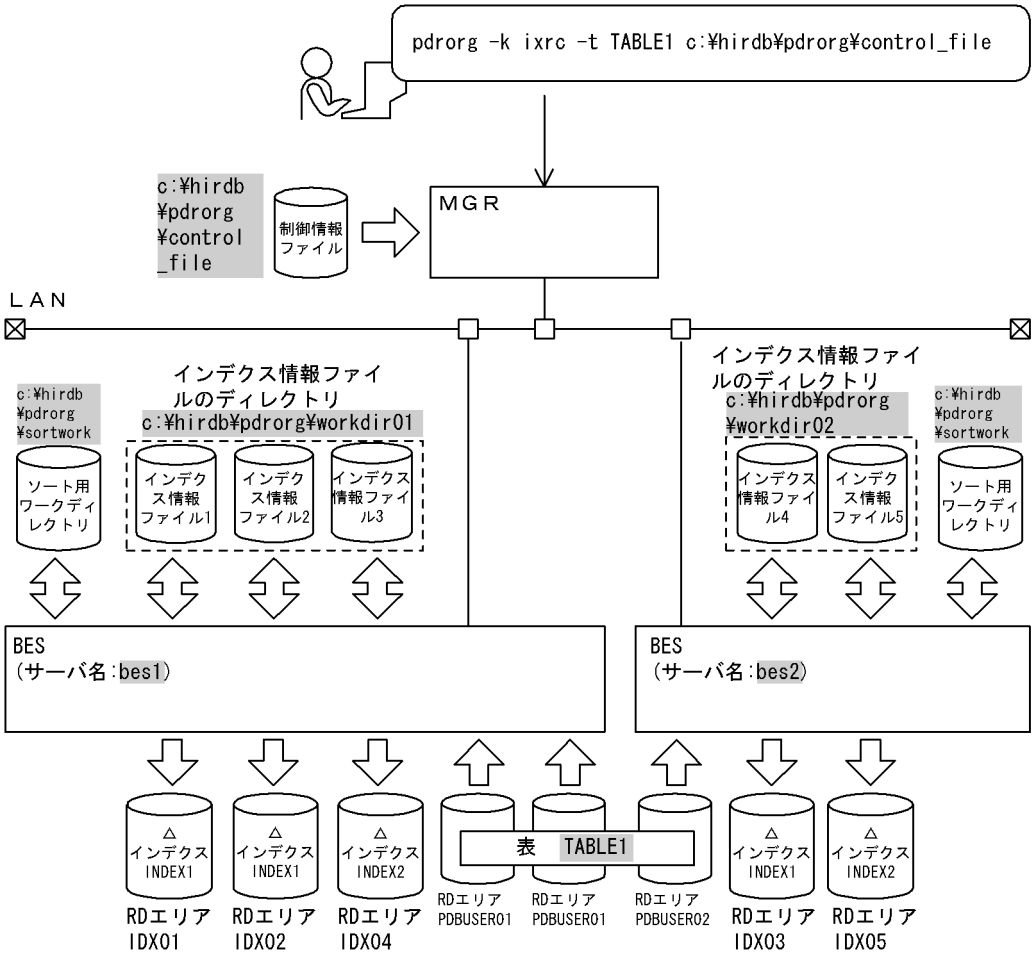
[概要]



（凡例）BES：バックエンドサーバ

(a)　表に定義されているすべてのインデクスを再作成する場合

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

インデクス（INDEX1，INDEX2）の再作成をします。

-k ixrc：インデクスの再作成をするときの指定

-t TABLE1：インデクスの再作成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
idxname name=*                                  1
idxwork bes1 c:¥hirdb¥pdrorg¥workdir01          2
idxwork bes2 c:¥hirdb¥pdrorg¥workdir02          2
sort bes1 c:¥hirdb¥pdrorg¥sortwork              3
sort bes2 c:¥hirdb¥pdrorg¥sortwork              3
```

[説明]

1. インデクス単位にインデクスの再作成をする指定

   *****：表（TABLE1）に定義されているすべてのインデクスを再作成する指定

2. インデクス情報ファイルを作成するディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥workdir01，c:¥hirdb¥pdrorg¥workdir02：インデクス情報ファイルを作成するディレクトリの名称

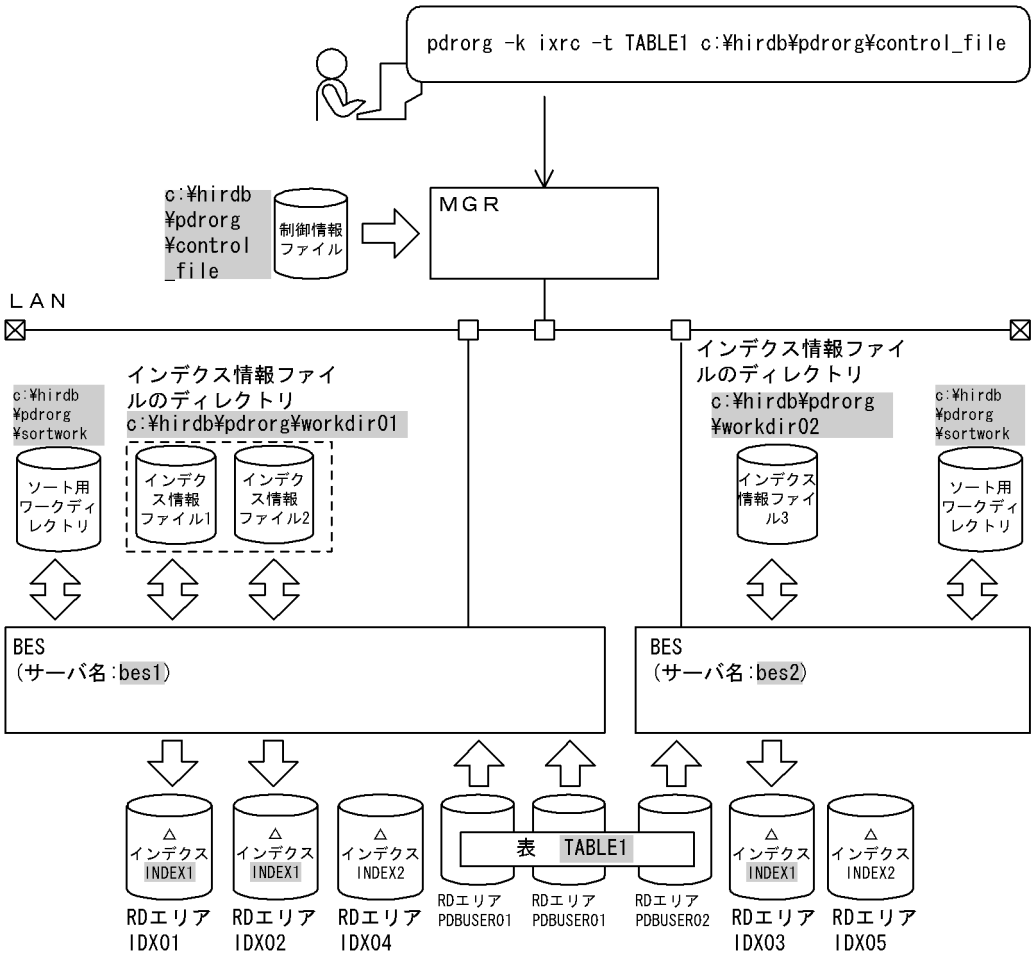
3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

(b)　表に定義されている一つのインデクスを再作成する場合

[入出力ファイルと RD エリアの関連図]



（凡例）　：データベース再編成ユティリティでの指定値

ＭＧＲ：システムマネジャ
ＢＥＳ：バックエンドサーバ

[実行するコマンドの説明]

インデクス（INDEX1）の再作成をします。

-k ixrc：インデクスの再作成をするときの指定

-t TABLE1：インデクスの再作成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
idxname name=INDEX1                          1
idxwork bes1 c:¥hirdb¥pdrorg¥workdir01       2
idxwork bes2 c:¥hirdb¥pdrorg¥workdir02       2
sort bes1 c:¥hirdb¥pdrorg¥sortwork           3
sort bes2 c:¥hirdb¥pdrorg¥sortwork           3
```

[説明]

1. インデクス単位にインデクスの再作成をする指定

   INDEX1：再作成するインデクスの名称

2. インデクス情報ファイルを作成するディレクトリの指定

   bes1，bes2：インデクス情報ファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥workdir01，c:¥hirdb¥pdrorg¥workdir02：インデクス情報ファイルを作成するディレクトリの名称

3. ソート用ワークディレクトリの指定

   bes1，bes2：ソート用ワークファイルを作成するサーバの名称

   c:¥hirdb¥pdrorg¥sortwork：ソート用ワークファイルを作成するディレクトリの名称

8.7.2　目的別の参照先一覧

ユーザが実行するインデクスの再作成の形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」を参照してください。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k ixrc	－
インデクスの再作成をする表の名称	-t	－
制御文を記述したファイル	制御情報ファイル名	－

項目	参照先	
	オプション	制御文
インデクス情報ファイルの情報，又はインデクスの情報	−	index 文，又は idxname 文

（凡例）−：該当しません。

(2) インデクスの再作成の方法によって指定する項目

項目	参照先	
	オプション	制御文
データベース更新ログを取得する場合，又は取得しない場合	-l	−
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※1	-u	−
グローバルバッファを使用しないで，ローカルバッファを使って一括入出力のインデクスの再作成をしたい場合※2	-n	−
インデクス作成後に，自動的にインデクス情報ファイルを削除したい場合	-o	−
インデクスの再作成の実行状態のメッセージ出力間隔を，10 万行単位以外にしたい場合	-m	−
インフォメーションメッセージの出力を抑止したい場合		
サーバ間通信の応答時間を監視したい場合	-X	−
インデクス格納 RD エリア単位にインデクスの再作成をする場合（index 文を指定する場合）で，%PDDIR%¥tmp ディレクトリの空き容量に余裕がないとき	−	sort 文
インデクス単位にインデクスの再作成をする場合（idxname 文を指定する場合）で，%PDDIR%¥tmp ディレクトリの空き容量に余裕がないとき	−	idxwork 文 sort 文
pdrorg の実行時間を監視したい場合	−	option 文
インデクスの再作成時に，インデクスの空き領域比率を変更したい場合※3	−	option 文

（凡例）−：該当しません。

注※1

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

注※2

一括入出力ページ数を指定すると，指定したページ数単位で一括入出力をするので，I/O 回数を削減できます。

注※3

pdrorg 実行時に RD エリアの容量不足が発生した場合，暫定的に RD エリアを拡張しないで処理を完了できます（ただし，空き領域比率が 0 のインデクスを除く）。

8.8 インデクスの再編成

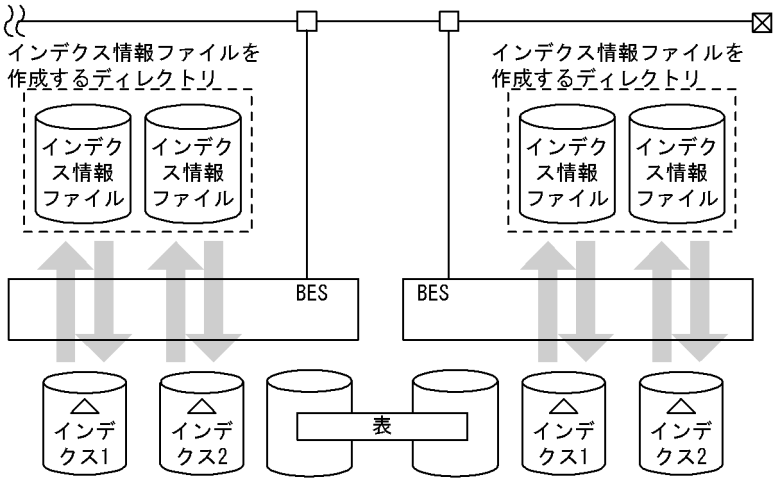
8.8.1 使用例

データベース再編成ユティリティの使用例（インデクスの再編成）を例 1 に示します。

例 1

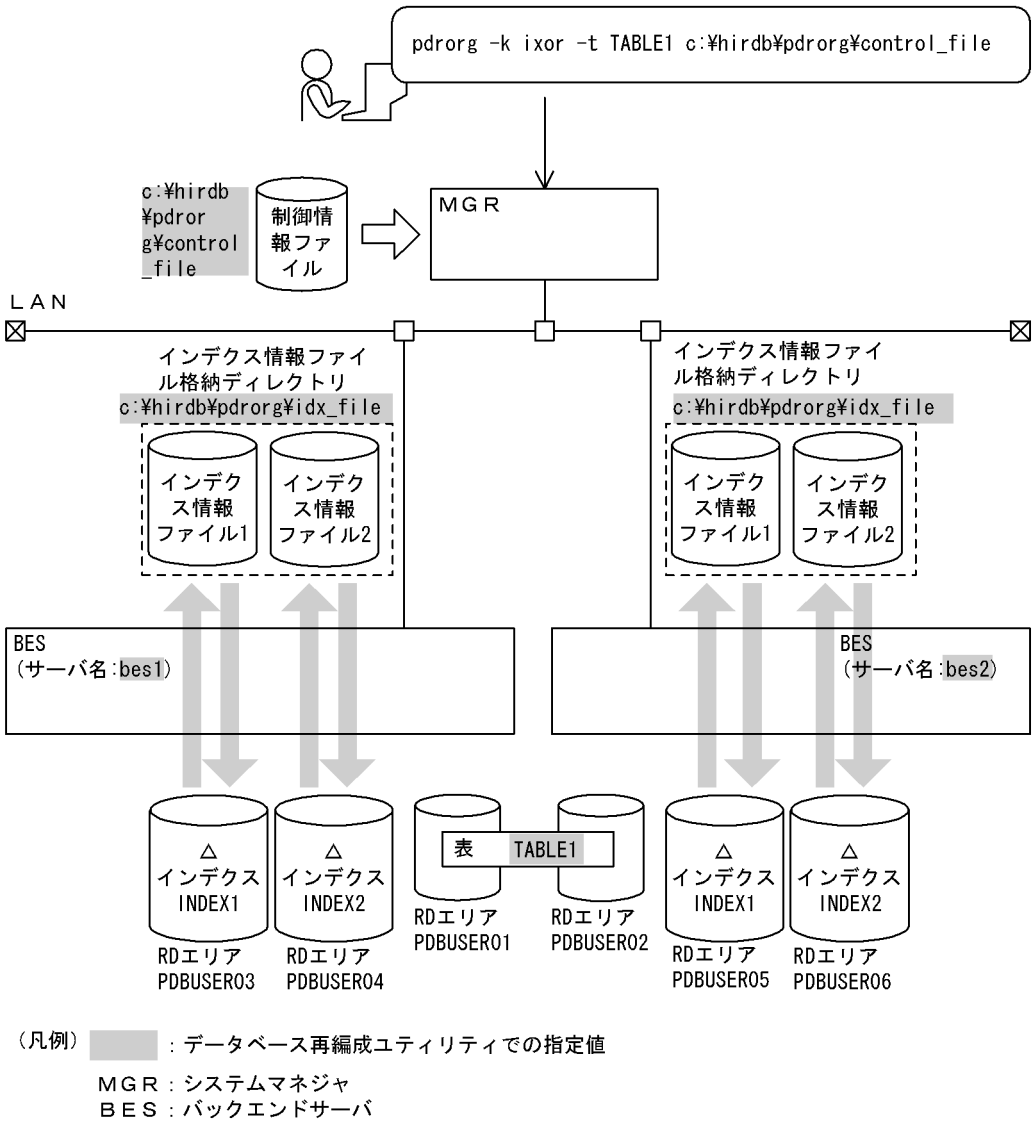
```
表（TABLE1）に定義されているインデクス（INDEX1, INDEX2）を再編成します。
表，及びインデクスは次のように定義されているものとします。
・表の定義
  CREATE TABLE TABLE1(C1 INT NOT NULL,C2 CHAR(8),C3 INT)
                IN ((PDBUSER01) C1 > 10,(PDBUSER02))
・インデクスの定義（分割キーインデクス）
  CREATE INDEX INDEX1 ON TABLE1(C1) IN ((PDBUSER03),(PDBUSER05))
・インデクスの定義（非分割キーインデクス）
  CREATE INDEX INDEX2 ON TABLE1(C2,C1) IN ((PDBUSER04),(PDBUSER06))
```

[概要]



（凡例）BES：バックエンドサーバ

[入出力ファイルと RD エリアの関連図]



[実行するコマンドの説明]

表（TABLE1）に定義されているインデクス（INDEX1，INDEX2）を再編成します。

-k ixor：インデクスの再編成をするときの指定

-t TABLE1：インデクスの再編成対象の表の名称

c:¥hirdb¥pdrorg¥control_file：制御情報ファイルの名称

[制御情報ファイル（c:¥hirdb¥pdrorg¥control_file）の内容]

```
idxname name=*                              1
idxwork bes1 c:¥hirdb¥pdrorg¥idx_file       2
idxwork bes2 c:¥hirdb¥pdrorg¥idx_file       2
```

[説明]

1. インデクス単位にインデクスの再編成をする指定

   ***：**表（TABLE1）に定義されているすべてのインデクスを再編成する指定

2. インデクス情報ファイル用ディレクトリの指定

bes1，bes2：インデクス情報ファイルを作成するサーバの名称

c:¥hirdb¥pdrorg¥idx_file：インデクス情報ファイルを作成するディレクトリの名称

8.8.2　目的別の参照先一覧

ユーザが実行するインデクスの再編成の形態によって，指定するオプション，制御文が異なります。

(1)のオプション，及び制御文は必ず指定してください。(2)で該当する項目があれば，その参照先にあるものが関連するオプション，及び制御文となります。なお，オプションについては，「オプション」が参照先となります。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k ixor	－
インデクスの再編成をする表の名称	-t	－
制御文を記述したファイル	制御情報ファイル名	－
インデクス情報ファイルの情報，又はインデクスの情報	－	index 文，又は idxname 文

（凡例）－：該当しません。

(2)　インデクスの再編成の方法によって指定する項目

項目	参照先	
	オプション	制御文
データベース更新ログを取得する場合，又は取得しない場合	-l	－
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※1	-u	－
インデクス作成後に，自動的にインデクス情報ファイルを削除したい場合	-o	－
サーバ間通信の応答時間を監視したい場合	-X	－
インデクス単位にインデクスの再編成をする場合（idxname 文を指定する場合）で，%PDDIR%¥tmp ディレクトリの空き容量に余裕がない場合	－	idxwork 文
pdrorg の実行時間を監視したい場合	－	option 文

項目	参照先	
	オプション	制御文
インデクスの再編成時に，インデクスの空き領域比率を変更したい場合※2	－	option 文

（凡例）－：該当しません。

注※1

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

注※2

pdrorg 実行時に RD エリアの容量不足が発生した場合，暫定的に RD エリアを拡張しないで処理を完了できます（ただし，空き領域比率が 0 のインデクスを除く）。

8.9　ディクショナリ表のメンテナンス

8.9.1　実行手順

ディクショナリ表のメンテナンスの実行手順を次に示します。

1. HiRDB の全 RD エリアを対象としたバックアップを取得します。

ポイント

ディクショナリ表のメンテナンス実行し，インデクスを追加すると，そのインデクスは削除できません。そのため，ディクショナリ表のメンテナンスを実行する前に，HiRDB の全 RD エリアを対象としたバックアップを取得してください。バックアップ取得後はデータディクショナリ用 RD エリア及びデータディクショナリ LOB 用 RD エリアは閉塞なしのオープン状態にしてください。

バックアップの取得方法については，マニュアル「HiRDB システム運用ガイド」の「バックアップの取得方法」を参照してください。

2. pdrorg コマンドでディクショナリ表のメンテナンスを実行します。

注意事項

実行する際，ほかのコマンドやユティリティ，及び UAP は実行しないでください。これらと同時実行した場合，動作保証できません。pdls -d prc -a コマンドなどによって，ほかのコマンドやユティリティ，及び UAP が実行されていないことを確認した上で実行してください。

コマンドの例を以下に示します。

（例）

```
pdrorg -k maintenance -c dic
```

[実行するコマンドの説明]

-k maintenance：ディクショナリ表のメンテナンスするときの指定

-c dic：ディクショナリ表のメンテナンスするための必須オプション

8.9.2　メンテナンス処理中にエラーが発生した場合の対処方法

ディクショナリ表のメンテナンスの処理中にエラーが発生した場合は，エラーの原因を取り除き，ディクショナリ表のメンテナンス（pdrorg）を再実行してください。

8.9.3　メンテナンスする前の状態に戻す方法

ディクショナリ表のメンテナンスでインデクスを追加した場合，追加したインデクスは削除できません。ディクショナリ表のメンテナンスを実行する前に取得したバックアップを使用して，データベース回復ユティリティで HiRDB の全 RD エリアを回復してください。また，HiRDB を pdstart -r コマンド以外で開始させた状態で回復する場合，HiRDB サーバ上のメモリに情報が残ったままとなるため，HiRDB を再起動してメモリをクリアしてください。

8.9.4　目的別の参照先一覧

ディクショナリの表メンテナンス実行時に指定するオプションについて説明します。

(1)のオプションは必ず指定してください。(2)で該当する項目があれば，その参照先にあるものが関連するオプションとなります。なお，オプションについては，「オプション」を参照してください。

(1)　必ず指定する項目

項目	参照先	
	オプション	制御文
pdrorg の処理内容	-k maintenance	−
処理対象	-c dic	−

（凡例）−：該当しません。

(2)　ディクショナリ表のメンテナンスの方法によって指定する項目

項目	参照先	
	オプション	制御文
pdrorg を実行するユーザの認可識別子を，環境変数 PDUSER 設定値以外に変更したい場合※	-u	−
サーバ間通信の応答時間を監視したい場合	-X	−

（凡例）−：該当しません。

注※

省略した場合は，環境変数 PDUSER の設定値が仮定されます。環境変数 PDUSER を設定していない場合は，ログインウィンドウのユーザ名が仮定されます。

(3)　(1)(2)以外の指定について

項目	参照先	
	オプション	動作
ログ取得方式	-l	常にログ取得モード（-l a）を適用します。
制御文	−	指定の値を無視して処理を続行します。

（凡例）−：該当しません。

8.10　コマンドの形式

8.10.1　形式

```
pdrorg -k 処理内容
       〔-c 処理対象〕
       〔-t {〔認可識別子.〕表識別子|〔認可識別子.〕all|表識別子〔,表識別子〕…}〕
       〔-r RDエリア名〕
       〔-l ログ取得方式〕
       〔-i インデックス作成方法〕
       〔-W {{dat|extdat}〔,〔セパレータ文字〕〔,{cr|crlf}〕〔,sup〕〕|bin〔-w〕|fixtext〔,〔パディング文字〕〔,{cr|crlf}〕〕}〕
       〔-g〕
       〔-j〕
       〔-u 認可識別子〕
       〔-n {一括入出力用ローカルバッファ面数〔,div〕|一括入出力用ローカルバッファ面数,〔div〕,ランダムアクセス用ローカルバッファ面数|,,ランダムアクセス用ローカルバッファ面数}〕
       〔-f アンロードデータファイル・LOBデータのアンロードファイルの種別〕
       〔-b アンロード順指定〕
       〔-S〕
       〔-o〕
       〔-m 〔経過メッセージ出力間隔〕〔,インフォメーションメッセージ出力抑止レベル〕〕
       〔-X サーバ間通信の応答監視時間〕
       制御情報ファイル名
```

注

背景色付き太字表示のオプションは，重要な項目，又は必ず指定する項目を表しています。また，制御情報ファイル名は，必ずオプションの最後に指定してください。

●pdrorg の機能ごとのオプション指定可否

pdrorg の機能ごとのオプション指定可否を次に示します。

オプション	pdrorg の機能							
	表の再編成	表のアンロード	表へのリロード	ディクショナリ表の再編成	インデクスの一括作成	インデクスの再作成	インデクスの再編成	ディクショナリ表のメンテナンス
-k	◎	◎	◎	◎	◎	◎	◎	◎
-c	○	○	○	◎	－	－	－	◎
-t	◎	◎	◎	○	◎	◎	◎	－
-r	○	○	○	－	－	－	－	－
-l	○	－	○	○	○	○	○	－
-i	○	－	○	－	－	－	－	－

オプション	pdrorg の機能							
	表の再編成	表のアンロード	表へのリロード	ディクショナリ表の再編成	インデクスの一括作成	インデクスの再作成	インデクスの再編成	ディクショナリ表のメンテナンス
-W	−	○	−	−	−	−	−	−
-g	○	○	○	−	−	−	−	−
-j	○	○	○	−	−	−	−	−
-u	○	○	○	○	○	○	○	○
-n	○	○	○	−	○	○	−	−
-f	○	○	○	○	−	−	−	−
-b	○	○	−	−	−	−	−	−
-S	○	○	−	−	−	−	−	−
-o	○	−	○	−	○	○	○	−
-m	○	○	○	○	○	○	−	−
-X	○	○	○	○	○	○	○	○
制御情報ファイル名	◎	◎	◎	◎	◎	◎	◎	−

（凡例）

◎：必ず指定してください。

○：任意に指定できます。

−：指定できません。

8.10.2　オプション

(1)　-k 処理内容

pdrorg の処理内容を指定します。

rorg

表の再編成をする場合に指定します。

unld

表のアンロードをする場合に指定します。

reld

表へリロードする場合に指定します。

ixmk

インデクスの一括作成をする場合に指定します。

ixrc

インデクスの再作成をする場合に指定します。

ixor

インデクスの再編成をする場合に指定します。

maintenance

ディクショナリ表のメンテナンスをする場合に指定します。

(a) 文字コード変換の有無

文字集合が定義された列を持つ表に対して pdrorg を実行する場合，-k と-W の指定値によって，文字コード変換をするかどうかが異なります。オプションの指定値と文字コード変換の有無を次の表に示します。

表 8–14 オプションの指定値と文字コード変換の有無

-k の指定		-W の指定			
		あり		なし	
		UOC あり	UOC なし	UOC あり	UOC なし
rorg	アンロード時	–	–	○	×
	リロード時	–	–	○	×
unld		○	○	○	○
reld		–	–	–	△
ixmk		–	–	–	–
ixrc		–	–	–	–
ixor		–	–	–	–
maintenance		–	–	–	–

（凡例）

○：文字コード変換をします。

△：アンロードデータファイル中の文字コードと文字集合が定義された列の文字コードが異なる場合に文字コード変換をします。

×：文字コード変換をしません。

–：該当しません。

(b) 注意事項

-k rorg を指定した場合，アンロード処理，リロード処理，及びインデクス作成処理（-i c 指定の場合）を連続で実行します。処理の途中で異常終了した場合，どの処理中に異常終了したのかを，データベース中

に表のステータスとして記憶しているため，同じ処理内容で再実行したときには異常終了した処理から再開始をします。この場合，必ずしもアンロード処理から開始されるとは限りません。詳細については，「表の再編成時のリロード処理でエラーとなった場合の運用方法」を参照してください。

前回の pdrorg が正常終了したか，異常終了したかに関係なく，常にアンロード処理から再編成を実施したい場合は，-k unld と-k reld を分けて実行する運用を検討してください。

OS の時刻を遅らせた場合に，誤ったインデクス情報ファイルを指定してインデクスの一括作成を実行すると，表とインデクスのデータが不一致になることがあります。詳細については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(2) -c 処理対象

～《user》

再編成する表の種別を指定します。

dic

ディクショナリ表を再編成又はメンテナンスをする場合に指定します。

user

表を再編成する場合に指定します。

(3) -t {〔認可識別子.〕表識別子 | 〔認可識別子.〕all | 表識別子〔,表識別子〕...}

処理対象となる表の名称を指定します。

(a) ユーザが定義した表の場合

ユーザが定義した表の場合，次の形式で指定してください。

```
-t {〔認可識別子.〕表識別子 | 〔認可識別子.〕all}
```

表名を一つだけ指定するか，又はスキーマが所有するすべての表を再編成する場合は all を指定してください。スキーマが所有するすべての表を再編成することを，**スキーマ単位の再編成**といいます。

認可識別子を省略した場合は，HiRDB に接続したときのユーザの名称が仮定されます。

＜規則＞

1. ビュー表は指定できません。
2. 認可識別子，又は表識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

＜スキーマ単位の再編成の規則＞

1. スキーマ単位の再編成は，-k rorg，unld，又は reld の場合で，同期点指定の再編成，リロードをしないときに指定できます。
2. スキーマ単位にアンロードした場合，そのアンロードデータファイルから表単位にリロードできます。逆に，表単位にアンロードした場合，そのアンロードデータファイルからスキーマ単位にはリロードできません。
3. アンロード又はリロードをする表の順序は，表識別子の昇順となります。
4. ある表を処理しているときにエラーが発生した場合は，その時点で処理を終了します。ある表がリロード時にエラーとなった場合は，正常に処理が終了した表については再編成が完了しているので，エラーとなった表，及びまだ処理をしていない表について，表単位にリロードしてください。
5. スキーマ単位の再編成をする場合，UAP が再編成中の表をアクセスして待ち状態にならないように，スキーマ内の表を格納している RD エリアを pdhold コマンドで閉塞するようにしてください。
6. アンロード時とリロード時で，スキーマにある表が変わった場合，次のようになります。

```
アンロードしたときの表：（T1，T3，T5，T7）
      ↓
リロードするときの表：（T1，T2，T3，T5）
```

T1，T3，T5：リロードします。

T2：アンロードしたときにはなかったため，処理をスキップします。

T7：リロードするときはないため，リロードはしません。

(b)　ディクショナリ表の場合

ディクショナリ表の場合，次の形式で指定してください。

```
-t 表識別子〔，表識別子〕…
```

ディクショナリ表の個数分，表名を指定できます。認可識別子は指定できません。すべてのディクショナリ表を再編成したい場合は，-t オプションを省略してください。

ディクショナリ表を再編成する場合，事前にデータディクショナリ用 RD エリアのバックアップを取得してください（リロード処理中にユニットダウンした場合，及び RD エリアの拡張に使用する表のリロード中に RD エリアが容量不足になった場合は，バックアップから再編成前の状態に戻す必要があります）。

指定できるディクショナリ表の表識別子を次に示します。なお，同じディクショナリ表の表識別子は重複指定できません。

項番	表識別子	備考
1	SQL_PHYSICAL_FILES	なし
2	SQL_RDAREAS	
3	SQL_TABLES	
4	SQL_COLUMNS	

項番	表識別子	備考
5	SQL_INDEXES	
6	SQL_USERS	
7	SQL_RDAREA_PRIVILEGES	
8	SQL_TABLE_PRIVILEGES	
9	SQL_DIV_TABLE	
10	SQL_DIV_TABLE_REGULARIZE	システムが使用する表
11	SQL_INDEX_COLINF	なし
12	SQL_TABLE_STATISTICS	
13	SQL_COLUMN_STATISTICS	
14	SQL_INDEX_STATISTICS	
15	SQL_VIEW_TABLE_USAGE	
16	SQL_VIEWS	
17	SQL_VIEW_DEF	システムが使用する表
18	SQL_DIV_INDEX	なし
19	SQL_DIV_COLUMN	
20	SQL_REFERENTIAL_CONSTRAINTS	
21	SQL_ALIASES	
22	SQL_ROUTINES	
23	SQL_ROUTINE_RESOURCES	
24	SQL_ROUTINE_PARAMS	
25	SQL_DATATYPES	
26	SQL_DATATYPE_DESCRIPTORS	
27	SQL_TABLE_RESOURCES	
28	SQL_PLUGINS	
29	SQL_PLUGIN_ROUTINES	
30	SQL_PLUGIN_ROUTINE_PARAMS	
31	SQL_REGISTRY_CONTEXT	システムが使用する表
32	SQL_REGISTRY_KEY	
33	SQL_INDEX_TYPES	なし
34	SQL_INDEX_RESOURCES	
35	SQL_INDEX_DATATYPE	

項番	表識別子	備考
36	SQL_INDEX_FUNCTION	
37	SQL_TYPE_RESOURCES	
38	SQL_INDEX_TYPE_FUNCTION	
39	SQL_EXCEPT	
40	SQL_USAGE_PRIVILEGES	システムが使用する表
41	SQL_IOS_GENERATIONS	システムが使用する表
42	SQL_TRIGGERS	なし
43	SQL_TRIGGER_ACTCOND	システムが使用する表
44	SQL_TRIGGER_COLUMNS	なし
45	SQL_TRIGGER_DEF_SOURCE	
46	SQL_TRIGGER_USAGE	
47	SQL_PARTKEY	
48	SQL_PARTKEY_DIVISION	
49	SQL_AUDITS	
50	SQL_AUDIT_REGULARIZE	システムが使用する表
51	SQL_KEYCOLUMN_USAGE	なし
52	SQL_TABLE_CONSTRAINTS	
53	SQL_CHECKS	
54	SQL_CHECK_COLUMNS	
55	SQL_DIV_TYPE	
56	SQL_SYSPARAMS	
57	SQL_DB_MANAGEMENT	システムが使用する表
58	SQL_DB_STATE_ANALYZED	
59	SQL_PUBLICVIEW_SAME_USERS	
60	SQL_INDEX_XMLINF	なし
61	SQL_SEQUENCES	
62	SQL_SDB_DB	システムが使用する表
63	SQL_SDB_DB_VIEW	
64	SQL_SDB_STORAGE_DB	
65	SQL_SDB_NAME_CODE	
66	SQL_SDB_DIR	

項番	表識別子	備考
67	SQL_ACCESS_SECURITY	なし
68	SQL_ROLES	システムが使用する表
69	SQL_ROLE_PRIVILEGES	

(4)　-r RD エリア名

～＜識別子＞((1～30))

複数の RD エリアに分割格納した横分割表の場合，特定の RD エリアの部分だけ再編成するときに，対象となる RD エリアの名称を指定します。

省略した場合，横分割表のすべてのデータが再編成の対象となります。

(a)　規則

1. RD エリア名を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。また，RD エリア名に空白が含まれる場合は，「¥""」と「"¥"」で囲んでください。
2. フレキシブルハッシュ分割の横分割表に対して，RD エリア単位にアンロードし，ハッシュ関数の変更，又は RD エリアの追加をした場合，リロード時はハッシュ関数を無視して元の RD エリアにそのままデータを格納します。

   キーレンジ分割，又は FIX ハッシュ分割の横分割表に対して，RD エリア単位にアンロードした後，分割条件を変更してリロードできません。

   ハッシュ関数の変更，RD エリアの追加，又は分割条件の変更をして表の再編成をしたい場合は，-g オプション指定の表単位の再編成をしてください。なお，ハッシュ関数の変更，及び RD エリアの追加は，定義系 SQL の ALTER TABLE で実行します。アンロードした表とリロードした表の表定義が異なる場合のリロード実行可否については，「アンロード元の表とリロード先の表の表定義が異なる場合のリロード実行可否」を参照してください。
3. スキーマ単位の再編成をする場合，-r オプションは指定できません。

(5)　-l ログ取得方式

～《p》

pdrorg 実行時の，データベースの更新ログの取得方法を指定します。

なお，pdrorg 実行中に異常終了した場合，更新ログを取得していても，データベースは pdrorg 実行前の状態には戻りません。pdrorg が異常終了した場合の対処方法については，「エラー時のデータベースの状態とその回復方法」を参照してください。

a

ログ取得モードを示します。ロールバック，及びロールフォワードに必要なデータベースの更新ログを取得します。

＜適用基準＞

このモードは，少量データの表に対して処理する場合に適しています。

ログ取得モードで実行した場合，pdrorg の実行前，及び実行後にバックアップを取得する必要はありませんが，処理性能はほかのモードに比べて劣ります。

p

更新前ログ取得モードを示します。ロールバックに必要なデータベースの更新ログは取得しますが，ロールフォワードに必要なデータベースの更新ログは取得しません。

＜適用基準＞

このモードは，大量データの表に対して処理する場合に適しています。

更新前ログ取得モードで実行した場合，ログ取得モードに比べると実行時間は短縮できます。ただし，以後の媒体障害などに備えて，pdrorg の実行後にバックアップを取得する必要があります。

pdrorg 実行中に障害が発生した場合，pdrorg が障害発生直前に同期を取った時点まで回復します。そのため，RD エリアがログレス閉塞になることは防げますが，処理対象表は pdrorg 実行前の状態には回復されません。

n

ログレスモードを示します。データベースの更新ログを取得しません。

＜適用基準＞

ログレスモードで実行した場合，ほかのモードに比べると実行時間は短縮できます。ただし，ユティリティ実行中の障害でバックアップから回復する場合や，以後の媒体障害などの発生によって，データベースのバックアップ及びログから回復する場合に備えて，pdrorg の実行前及び実行後にバックアップを取得する必要があります。このモードは，大量データの表に対して処理する場合に適しています。

＜注意事項＞

1. pdrorg をログレスモードで実行していた場合にエラーが発生したときは，エラーによって閉塞した RD エリアはバックアップから回復するか，又は再初期化をする必要があります。
2. インデクスの再作成（-k ixrc）の場合にログレスモードを指定すると，キー検索処理中はデータベースが障害閉塞となることを防ぐため，ログ取得モードとなります。また，インデクスの再編成（-k ixor）の場合にログレスモードを指定したときも，キー検索処理中はデータベースが障害閉塞となることを防ぐため，更新前ログ取得モードが仮定されます。このため，ログレスモードを指定してもキー検索処理中はセグメント解放ログが出力されます。

(a) 注意事項

1. p，及び n の場合に，以前取得したバックアップとログ，又はアンロードデータ（該当する RD エリアに再編成対象の表だけ格納されている場合）から回復できるときは，pdrorg 実行前のバックアップ取得は不要です。

2. p，及び n の場合の運用方法（データベースの更新ログを取得しないときの運用）については，マニュアル「HiRDB システム運用ガイド」を参照してください。

3. -l オプションの指定に関係なく，pdrorg が生成するトランザクション（T）によってトランザクションログは必ず取得されます。1 サーバ当たり，次のトランザクションログを生成するため，pdrorg 実行時のログ量の計算式は次のようになります。

```
ログ量＝（1328＋176×3）×T＋A　（単位：バイト）

＜-k unldの場合＞
T＝（x×2）＋（y×2）
＜-k reldの場合＞
T＝（x×3）＋（y×3）＋（z×2）
＜-k ixrcの場合＞
T＝（x×2）＋（z×2）
＜-k ixmkの場合＞
T＝z×2
＜-k ixorの場合＞
T＝z×3
＜-k rorgの場合＞
T＝＜-k unldの場合＞＋＜-k reldの場合＞＋（x＋y）
＜-k maintenanceの場合＞
マニュアル「HiRDB システム導入・設計ガイド」の「システムファイル及び監査証跡ファイルの容量の見積もり」の「システムログファイルの容量見積もり」の「ディクショナリ表のメンテナンス実行時に出力されるシステムログ量」を参照してください。

  x：表の数（同期点指定のリロードの場合は同期点の数）※
  y：LOB列（LOB属性）格納用RDエリアの数（-jありの場合は1）※
  z：インデクス数×インデクス格納用RDエリアの数（-i sの場合は不要）※
```

注※

スキーマ単位に処理する場合は，そのスキーマが所有するすべての表ごとに計算をする必要があります。

A ＝データベース操作に応じて出力されるシステムログ量（データベースの更新ログ量）。この値は-l オプションの指定値で変わります。システムログ量の求め方については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。-k maintenance の場合は，-l オプションの指定値に関係なく，マニュアル「HiRDB システム導入・設計ガイド」の「システムファイル及び監査証跡ファイルの容量の見積もり」の「システムログファイルの容量見積もり」の「ディクショナリ表のメンテナンス実行時に出力されるシステムログ量」を参照してください。

また，p 又は n を指定した場合，排他制御によって ENQ ログを取得します。1 サーバ当たり次の ENQ ログを取得します。

```
ENQログ＝（p＋q＋r）×T

  p：表格納用RDエリアの数
  q：LOB列（LOB属性）格納用RDエリアの数
  r：インデクス格納用RDエリアの数
```

したがって，「システムログファイルのレコード長×ENQ ログ」の分だけシステムログが出力されます。

4. インデクスの再作成（-k ixrc）又はインデクスの再編成（-k ixor）の場合にログレスモードを指定したときに出力されるセグメント解放ログのログ量は，PURGE TABLE 実行時に出力するインデクスのログと同じ量になります。ログ量の計算式は，マニュアル「HiRDB システム導入・設計ガイド」の「PURGE TABLE 文実行時に出力されるシステムログ量」の「インデクス」を参照してください。計算式中の確保セグメント数については，pddbst の RD エリア単位の論理的解析，又はインデクス単位の解析を実行して，解析結果の使用中セグメント数で計算してください。
5. -k unld の場合，-l オプションの指定値に関係なく，ログ取得モード（-l a）が仮定されます。

(6) -i インデクス作成方法

インデクスの作成方法を指定します。

このオプションを省略すると，ユーザが定義した表の場合は c，ディクショナリ表の場合は s が仮定されます。

c

インデクス一括作成モードを示します。このモードを指定した場合，表へのリロード処理に引き続き，インデクスを一括作成します。

＜適用基準＞

大量データの表を再編成をする場合に，高速にインデクスを作成できます。

n

インデクス情報出力モードを示します。このモードを指定した場合，インデクス情報だけをインデクス情報ファイルに出力します。

＜適用基準＞

出力されたインデクス情報ファイルを使用して，pdrorg でインデクスの一括作成（-k ixmk）を複数同時に実行することで，高速にインデクスを作成できます。特に，HiRDB/パラレルサーバで複数のサーバに分割格納されている表の場合に効果があります。

＜注意事項＞

1. インデクス情報出力モード（-i n）を指定した場合，表へのリロード後に pdrorg でインデクスの一括作成（-k ixmk）をする必要があります。インデクスの一括作成を実行していない状態で，ALTER TABLE で RD エリアを追加すると，出力されたインデクス情報ファイルは使用できなくなります。ALTER TABLE で RD エリアを追加する場合は，先にインデクスの一括作成を終了させてください。なお，先に ALTER TABLE で RD エリアを追加してしまった場合は，pdrorg のインデクスの再作成（-k ixrc）をする必要があります。
2. 再編成対象の表にプラグインインデクスが定義されている場合，プラグインがプラグインインデクス一括作成部分回復機能をサポートしていないと，-i n は指定できません。
3. インデクス情報出力モード（-i n）を指定した場合，インデクスの一括作成（-k ixmk）が完了するまでは ALTER INDEX でインデクスの名称を変更しないでください。インデクスの名称を変更すると，インデクスの一括作成ができなくなります。なお，インデクスの一括作成が完了する前

にインデクスの名称を変更してしまった場合は，ALTER INDEX で元のインデクス名称に戻してからインデクスの一括作成をし，その後でインデクスの名称を変更してください。

4. インデクス情報出力モード（-i n）を指定した場合，インデクスの一括作成（-k ixmk）が完了するまでは，インデクスを格納した RD エリアの名称を変更しないでください。インデクスを格納した RD エリアの名称を変更すると，インデクスの一括作成ができなくなります。なお，インデクスの一括作成が完了する前にインデクスを格納した RD エリアの名称を変更してしまった場合は，pdmod で元の RD エリア名称に戻してからインデクスの一括作成をし，その後で RD エリアの名称を変更してください。

s

インデクス更新モードを示します。行を格納するたびにインデクスを更新します。

＜適用基準＞

少量データの表を再編成する場合に指定してください。

(a) インデクス作成上の注意事項

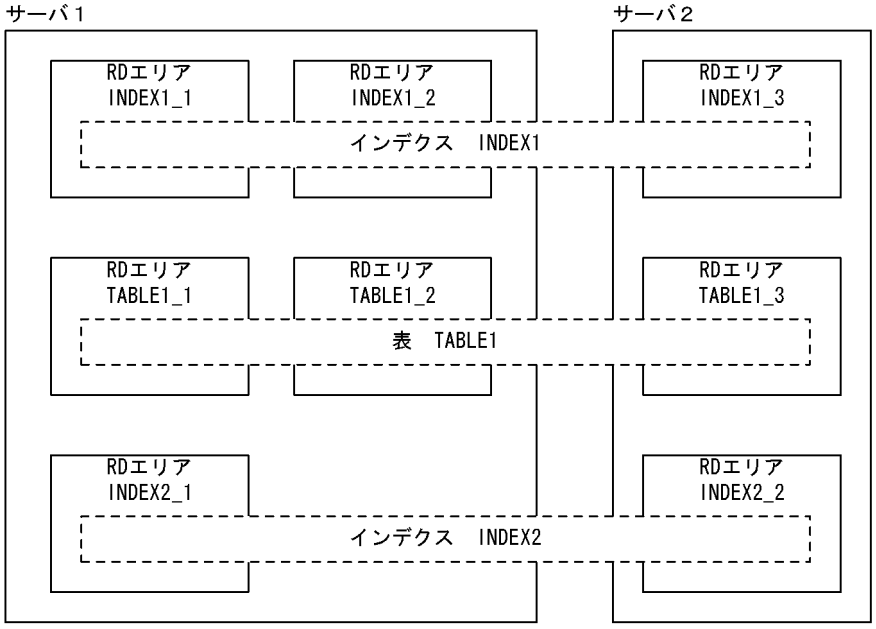
1. 横分割インデクスと非横分割インデクスでのインデクス作成方法を次の図に示します。

1 サーバ内の複数 RD エリアに表を分割格納している場合，表格納用 RD エリアと同じ数のインデクス格納用 RD エリアに格納する横分割インデクスと，表格納用 RD エリアの数とは関係なく一つのインデクス格納用 RD エリアに格納する非横分割インデクスがあります。

横分割インデクスは，サーバ内で分割されている表に対して，同じように分割格納します（図中の RD エリア INDEX1_1，INDEX1_2）。

非横分割インデクスは，表格納用 RD エリアの数に関係なく，サーバ内の 1RD エリアに格納します（図中の RD エリア INDEX1_3，INDEX2_1，INDEX2_2）。

図 8–19　分割キーインデクスと非分割キーインデクスでのインデクス作成方法



［説明］

再編成でのインデクスの作成は，-i オプションの指定に従います。

例えば，RD エリア単位に TABLE1_1 を再編成した場合，横分割インデクス INDEX1_1 が作成されます（INDEX1_2 は影響を受けません）。しかし，非横分割インデクス INDEX2_1 は，TABLE1_1 及び TABLE1_2 の両方の情報がないと作成できないため，インデクスは作成されません（TABLE1_1 のインデクス情報ファイルだけ作成されます）。

非横分割インデクスを作成するためには，TABLE1_2 に対しても再編成をして，TABLE1_2 のインデクス情報ファイルを作成します。そして，作成済みの TABLE1_1 のインデクス情報ファイルと TABLE1_2 のインデクス情報ファイルを使用して，pdrorg のインデクスの一括作成をします。

2. リロードが終了してから，インデクスの一括作成が終了するまでの間は，該当するインデクスは未完状態となり，使用できません。
3. -i c，又は-i n を指定すると，「インデクス数×表格納用 RD エリア数」のインデクス情報ファイルを作成します。このファイルは，リロード時に同時にオープンするため，ファイル数が多いと 1 プロセスで使用できるファイル数の上限を超えることがあります。上限を超えると，性能が劣化するおそれがあります。そのため，RD エリアごとにデータをリロードすることを推奨します。

   なお，1 プロセスで使用できるファイルディスクリプタ数の上限値は，2048 です。

(b)　-i c 又は-i n 指定時の注意事項

-i c 又は-i n 指定時に index 文，及び idxwork 文を省略すると，既定の出力先ディレクトリにインデクス情報ファイルを出力します。pdrorg が異常終了すると，このインデクス情報ファイルが不完全な状態で残ります。このため，pdrorg の異常終了を何度も繰り返すと不完全なインデクス情報ファイルが増加し，ディスク容量など OS の資源を圧迫することになります。したがって，既定の出力先ディレクトリに出力した必要のないインデクス情報ファイルについては削除するようにしてください。

index 文を省略して idxwork 文だけを指定した場合に，idxwork 文に指定したディレクトリ下に出力するインデクス情報ファイルについても同様です。

既定の出力先ディレクトリについては「規則及び注意事項」の「インデクス作成時のファイルの出力先について」を参照してください。

また，インデクス情報ファイルは，次の名称規則で出力します。

```
INDEX-インデクス名※-インデクス格納用RDエリア名※-一意となる文字列
```

注※

インデクス名又はインデクス格納 RD エリア名に「¥」が含まれている場合は，「¥」を「(」に置き換えます。

(7)　-W　{{dat ｜ extdat}〔,〔セパレータ文字〕〔,{cr ｜ crlf}〕〔,sup〕〕 ｜ bin〔 -w〕 ｜ fixtext〔,〔パディング文字〕〔,{cr ｜ crlf}〕〕}

アンロードデータファイルをデータベース作成ユティリティの入力データファイルとして使用する場合，又は UAP などで使用する場合に指定します。また，表やインデクスをエクスポートする場合に指定します。

表のデータを，DAT 形式，拡張 DAT 形式，バイナリ形式，又は固定長データ形式でアンロードデータファイルに出力します。

エクスポートする場合は，-W bin -w を指定します。

-W オプションを指定して出力したアンロードデータファイルの形式については，「データベース作成ユティリティの入力用ファイルの形式」を参照してください。

dat

DAT 形式でデータを出力する場合に指定します。

extdat

拡張 DAT 形式でデータを出力する場合に指定します。

拡張 DAT 形式でデータを出力する場合，アンロードデータファイルは複数個指定できません。

セパレータ文字　～＜文字列＞

DAT 形式，又は拡張 DAT 形式で出力する場合，各データ間のセパレータ文字をコンマ（,）以外に変更するときに指定します。

省略した場合，セパレータ文字はコンマとなります。

＜規則＞

1. 次に示す文字は，データベース作成ユティリティの入力データファイルのセパレータ文字に指定できないため，セパレータ文字として指定できません。

   ＊（アスタリスク），"（引用符），_（下線）

2. 次に示す文字はアンロードするデータと重なる可能性があるため，セパレータ文字として適していません。

   ・英大文字（A～Z），英小文字（a～z）

   ・数字（0～9）

   ・行データを DAT 形式に変換したとき，その文字と重なる文字

   ・日本語入力時に文字コードが重なる文字

   （｜，￥，[,]，(,)，{,}，￣）

   ・数値データがある場合の符号（＋，－）

   ・日付データ入力時のハイフン（-）

   ・時刻データ入力時のコロン（：）

   ・日間隔，時間隔データ入力時のピリオド（.）

3. コンマをセパレータ文字にしたい場合は，セパレータ文字を指定しないでください。

4. セパレータ文字に空白が含まれる場合は，引用符（"）で囲んでください。

{cr ｜ crlf}

DAT 形式，又は拡張 DAT 形式で出力する場合，改行コードの出力方法を指定します。省略した場合は，cr が仮定されます。

cr：0x0a で改行コードを出力します。

crlf：0x0d，0x0a で改行コードを出力します。

セパレータ文字を省略して cr，又は crlf を指定したい場合はコンマで区切ります（例：-W dat, ,cr）。

sup

DAT 形式，又は拡張 DAT 形式で CHAR，NCHAR，MCHAR 型の列データを出力する場合に，列データの後方に連続する空白を出力しないときに指定します。

CHAR，MCHAR 型の場合，表定義の列長に満たない部分の半角空白を圧縮します。列データがすべて空白の場合は，半角空白を 1 バイト出力します。

NCHAR の場合，表定義の列長に満たない部分の 2 バイトの空白を圧縮します。

列データがすべて空白の場合は，2 バイトの空白を 1 文字出力します。2 バイトの空白は使用する文字コードによって異なります。文字コードについては，マニュアル「HiRDB SQL リファレンス」を参照してください。

sup オプションを指定した場合の出力形式については，「データベース作成ユティリティの入力用ファイルの形式」を参照してください。

＜適用基準＞

sup オプションを指定すると，次の効果があります。

- アンロードデータファイルの容量を削減できます。
- 表定義時に CHAR，NCHAR，又は MCHAR 型の列に対してサプレスオプションを指定していると，後方の空白を既に圧縮した形でデータベース作成ユティリティの入力データを生成します。このため，アンロード時も圧縮に必要なオーバヘッドを削減できます。

＜注意事項＞

1. sup オプションを指定してアンロードした場合，アンロードデータファイル中の列データの長さは一定になりません。そのため，出力したデータをソートしたり，UAP で編集したりする場合，注意が必要です。
2. CHAR 型，MCHAR 型の列には，複数バイト文字や，文字コード以外の値も格納できますが，sup オプションを指定した場合，複数バイト文字の何バイト目かの文字が空白と同じ値であったり，文字コード以外の値が空白と同じ値であったとしても，無条件に圧縮されます。この場合，同じ表に再度データロードすれば列の定義長に満たない部分は空白で埋められます。ただし，そのデータをほかのプログラムで参照したり，表示したりする場合，最後の 2 バイトコードの文字が文字化けするため sup オプションを指定しないでください。

bin 〔-w〕

バイナリ形式のデータを出力する場合に指定します。

エクスポートする場合は-w オプションを同時に指定します。エクスポートの詳細については，マニュアル「HiRDB システム運用ガイド」の「表定義情報と表データを同時に移行する方法」を参照してください。

fixtext

固定長データ形式のデータを出力する場合に指定します。

＜指定例＞

1. パディング文字，及び｛cr ｜ crlf｝を省略する場合

   -W fixtext

2. パディング文字が△で，｛cr ｜ crlf｝を省略する場合

   -W fixtext ,△

3. パディング文字が△で，cr を指定する場合

   -W fixtext ,△,cr

4. パディング文字を省略し，crlf を指定する場合

   -W fixtext ,,crlf

パディング文字

列データがナル値，要素データがナル値，又は定義長より短い可変長文字列の場合，定義長まで埋める文字を指定します。

＜規則＞

1. 指定できるパディング文字は 1 バイトの文字です。
2. パディング文字を省略した場合，空白文字（0x20）が仮定されます。パディング文字を空白文字にする場合は，パディング文字を省略してください。
3. アスタリスク（*），引用符（"），及び下線（_）は指定できません。

{cr ｜ crlf}

改行コードを付ける場合に指定します。

cr：

1 バイトの改行コード（0x0a）を付けます。

crlf：

2 バイトの改行コード（0x0d，0x0a）を付けます。

(a) 適用基準

次のような場合に指定します。

- 別の表へ表データを移行する場合
- 表データをアンロードデータファイルに出力し，そのデータを UAP などで使用する場合

(b) 規則

1. -W オプション指定で出力したアンロードデータファイルは，pdrorg のリロードでは使用できません。
2. このオプションを指定する場合，-f オプションに hirdb は指定できません。
3. スキーマ単位の再編成をする場合，-W オプションは指定できません。

4. 表の 1 行の定義長が 512 メガバイトを超える場合，DAT 形式，及び固定長データ形式ではデータを出力できません。この場合，バイナリ形式でデータを出力してください。
5. 文字コード変換が必要な場合は文字コード変換をしてから，移行先へデータロードしてください。
6. 文字集合 UTF-16 を指定した CHAR 型の列を，キーレンジ分割又はマトリクス分割（境界値指定のキーレンジ分割）の分割キーとして指定した表の場合，分割キーの長さが列の定義長を超えるときは，その表はエクスポートできません。
7. LOB 列又は LOB 属性の抽象データ型の列がある表に対して-W オプションを指定する場合，実データ又はナル値のどちらかが出力されます。-W オプションと出力されるデータの関係を次の表に示します。

表 8–15　-W オプションと-j オプションの関係

列のデータ型		-W オプション		
		bin	dat 又は extdat	fixtext
既定義型	BINARY 型	実データ	実データ	出力しません
	BLOB 型	実データ	ナル値	出力しません
	上記以外	実データ	実データ	実データ
抽象データ型	BLOB 属性※	実データ	ナル値	出力しません
	上記以外	実データ	実データ	出力しません

注※

unld_func 文に指定するコンストラクタパラメタ逆生成関数の引数の型です。

(8)　-g

HiRDB/パラレルサーバの場合で，複数のサーバに格納されている表データを，1 サーバの 1 アンロードデータファイルにアンロードする場合に指定します。

LOB 列がある表の場合も，一つの LOB データのアンロードファイルへアンロードできます。

(a)　適用基準

このオプションは次のような場合に指定します。

- バックアップ用にアンロードデータファイルを作成する場合
- 表の分割条件を変更する場合
- HiRDB/パラレルサーバで，該当するホストが容量不足でアンロードできない場合，他ホストに容量が確保できるとき

(b)　注意事項

1. HiRDB/パラレルサーバでスキーマ単位の再編成をする場合，及び表，インデクスを-W bin -w オプションを指定してエクスポートする場合は，-g オプションを省略しても，指定したものと仮定されます。

2. RD エリア単位にアンロードしたデータを，-g オプションを指定して表単位にリロードすると，ほかのサーバの RD エリアも 0 件データとして処理するため，データを削除してしまいます。この場合には，-g オプションを指定しないでください。
3. RD エリア単位にアンロードしたデータを，-g オプションを指定して RD エリア単位でリロードする場合，表の分割条件は変更できません。
4. HiRDB/パラレルサーバで，複数のサーバに分割格納された表をアンロードする場合，-g オプションを指定しているときにはサーバごとに順次アンロードします。この場合，パラレル性能は劣化します。

(9) -j

LOB 列又は LOB 属性を持つ抽象データ型が定義された表に対して基表と LOB データを同時に再編成，アンロード，リロードする場合に指定します。なお，LOB 列及び LOB 属性を持つ抽象データ型が定義されてない表に-j オプションを指定しても指定を無視します。

再編成で，LOB 列が定義された表に対して基表だけ，又は LOB データだけを再編成する場合は，このオプションを指定しないでください。この場合の制御文の指定方法を次に示します。

再編成対象	指定する制御文
基表	unload
LOB データ	lobunld

次の場合は，指定がなくてもこのオプションを仮定します。

- スキーマ単位の再編成，アンロード，リロード
- -W オプションを指定したアンロード

次の場合は，この指定が必須となります。

- LOB 列が定義された表を同期点指定のリロードをする場合
- LOB 列又は LOB 属性を持つ抽象データ型が定義された表をアンロードする場合

同表へのリロード可否，及び別表間でのデータ移行可否を次の表に示します。

表 8–16　同表へのリロード可否

項番	アンロードファイル作成条件				リロード可否	
	対象表	オプション		制御文	-j オプション	
	LOB 列又は LOB 属性	-k	-j		なし	あり
1	なし	rorg	−	unload	○	○
2		unld※1	−	unload	○	○
3	あり	rorg	なし	unload	○	×

項番	アンロードファイル作成条件				リロード可否	
	対象表	オプション		制御文	-j オプション	
	LOB 列又は LOB 属性	-k	-j		なし	あり
4				lobunld	○	×
5				unload， lobunld	×	×
6			あり※2	unload	×	○
7		unld※1	なし※3	unload	非該当	非該当
8			あり	unload	×	○

（凡例）

－：-j オプションの指定によりません。

○：リロードできます。

×：リロードできません（実行時エラー）。

注※1

-W オプションを指定したアンロードはリロード対象外のため除きます。

注※2

-j オプションが仮定される場合を含みます。

注※3

このケースではアンロードできません（エラーとなります）。

表 8–17　別表間でのデータ移行可否

項番	移行対象表	移行方法		移行可否
	LOB 列又は LOB 属性	データ抽出	データ登録	
1	なし	pdrorg -k unld -W ..	pdload ..	○
2		pdrorg -k unld -j ..	pdrorg -k reld -j ..	△
3		pdrorg -k unld ..	pdrorg -k reld ..	△
4	あり	pdrorg -k unld -W ..	pdload ..	○
5		pdrorg -k unld -j ..	pdrorg -k reld -j ..	△
6		pdrorg -k unld ..	pdrorg -k reld ..	×

（凡例）

○：データ移行できます（推奨）。

△：データ移行できます（非推奨）。

×：データ移行できません。

(a)　規則

1. LOB 列がない表に対してこのオプションを指定した場合は，指定が無視されます。

2. スキーマ単位の再編成をする場合，及び表，インデクスを-W bin -w オプションを指定してエクスポートする場合は，-j オプションを省略しても，指定したものと仮定されます。
3. アンロード時に-j オプションを指定した場合には，リロード時にも-j オプションを指定してください。
4. HiRDB/パラレルサーバで，複数のサーバに分割格納された表をアンロードする場合，-g オプションを指定しているときにはサーバごとに順次アンロードします。この場合，パラレル性能は劣化します。

(10)　-u 認可識別子

pdrorg を実行するユーザの認可識別子を指定します。

省略時仮定値については，「省略時仮定値」を参照してください。

このオプションを指定した場合，パスワード入力要求の応答付きメッセージが表示されます。パスワードが不要な場合は，応答付きメッセージに対して空送信してください。

ここで指定した認可識別子で HiRDB に接続し，実行権限のチェックをします。

(a)　適用基準

環境変数 PDUSER に設定されている認可識別子ではない，別の認可識別子で実行したい場合に指定します。

(b)　省略時仮定値

省略した場合，次の方法で認可識別子/パスワードが仮定されます。

1. pdrorg 実行時の環境変数 PDUSER の設定値が仮定されます。パスワードの応答ができない環境で実行するときは，PDUSER を必ず設定してください。環境変数 PDUSER の指定例を次に示します。

   （例）

   パスワードありの場合　PDUSER=¥””認可識別子”¥”/¥””パスワード”¥”

   パスワードなしの場合　PDUSER=¥””認可識別子”¥”
2. 環境変数 PDUSER が設定されていない場合，ログインウィンドウのユーザ名が仮定されます。パスワードについては，入力要求の応答付きメッセージが表示されるので，そこで応答してください。パスワードが不要な場合は，応答付きのメッセージに対して空送信してください。

(c)　規則

1. このオプションは，応答できない環境では指定しないでください。
2. 認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(11) -n {一括入出力用ローカルバッファ面数〔,div〕|一括入出力用ローカルバッファ面数,〔div〕,ランダムアクセス用ローカルバッファ面数|,,ランダムアクセス用ローカルバッファ面数}

アンロード処理でのデータベースからの読み込み，及びリロード処理でのデータベースへの書き込みを行う場合に使用する，ローカルバッファの面数を指定します。-n オプションを指定した際に，一括入出力用ローカルバッファ面数，又はランダムアクセス用ローカルバッファ面数を指定していない場合は，該当ローカルバッファは使用しません。なお，-n の指定自体を省略した場合は，ローカルバッファを使用しません。この場合，グローバルバッファを使用して 1 ページ単位に出力します。

一括入出力用ローカルバッファ面数　～<符号なし整数>((2～4096))

一括入出力用ローカルバッファの面数を指定します。一括入出力用ローカルバッファは，データページに対して使用します。

一括入出力用ローカルバッファ面数には，16～32 面を指定することをお勧めします。184 キロバイト÷ページ長を上限に指定値を決めてください。

div

次の条件をすべて満たす場合に div を指定します。次の条件をすべて満たす場合に div を指定しないと，入出力回数が増加し，性能が劣化することがあります。

- アンロード時の分割条件と，リロード時の分割条件が異なる
- 表格納用 RD エリアが 1 サーバ内に複数ある

なお，div を指定すると，サーバ内表分割数分のバッファを確保するため，メモリ消費量が増えます。

ランダムアクセス用ローカルバッファ面数　～<符号なし整数>((4～125000))

ランダムアクセス用ローカルバッファの面数を指定します。ランダムアクセス用ローカルバッファは，インデクスページに対して使用します。このオペランドは，インデクス更新モードでインデクスを作成する場合に指定します。インデクス更新モードでは，行を格納するたびに B-tree 構造のインデクスページを参照しますが，ランダムアクセス用ローカルバッファを使用すれば，これらのインデクスページを当該バッファに読み込むため，グローバルバッファ競合によるオンライン処理の性能劣化を軽減できます。

推奨する-n オプションの指定を次の表に示します。

表 8–18　推奨する-n オプションの指定（pdrorg）

条件	表種別	列定義	サーバ内での表の分割	
			あり	なし
アンロード時	FIX 表	−	-n x，div	-n x
	非 FIX 表	−	-n の指定を省略	
リロード，再編成時	FIX 表	−	-n x，div	-n x

条件	表種別	列定義	サーバ内での表の分割	
			あり	なし
	非 FIX 表	列長が 256 バイトを超える可変長文字列，又は BINARY 列を定義している場合	-n の指定を省略	
		上記以外	-n x，div	-n x

（凡例）

x：一括入出力用ローカルバッファ面数

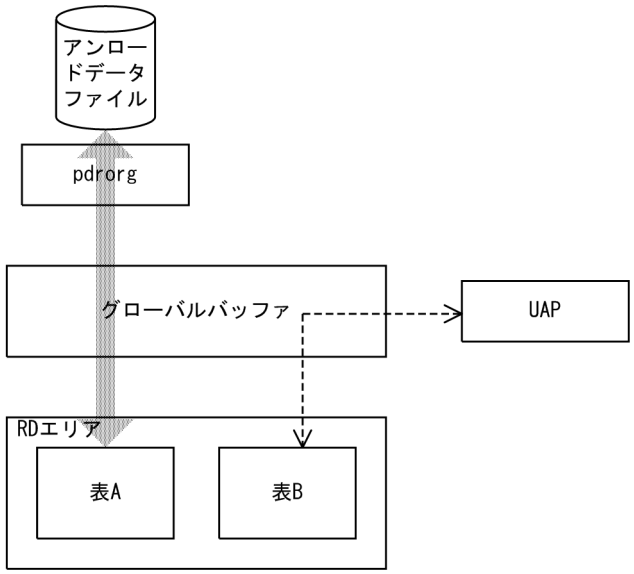
−：該当なし

(a) pdrorg が使用するバッファについて

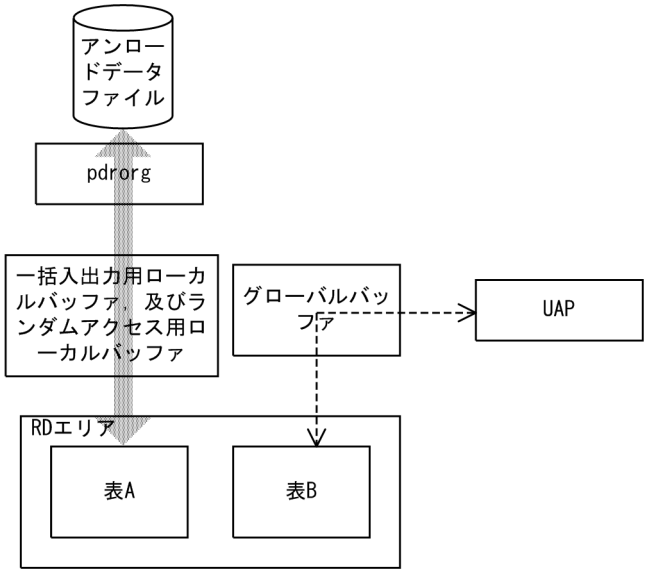
1. -n オプションを省略した場合，グローバルバッファを使用します。この場合，再編成時に大量のグローバルバッファを使用するため，そのグローバルバッファを使用する UAP のトランザクション性能が劣化します。-n オプションを指定すると，このようなバッファ競合はなくなります。pdrorg とバッファの関係を次の図に示します。

図 8–20　pdrorg とバッファの関係

●グローバルバッファだけを使用する場合



●ローカルバッファとグローバルバッファを使用する場合



〔説明〕

グローバルバッファだけを使用する場合（-n オプションを省略した場合），pdrorg と UAP とでバッファの競合が発生します。

ローカルバッファとグローバルバッファを使用する場合（-n オプションを指定した場合），pdrorg と UAP とのバッファの競合は発生しません。ただし，LOB 列の再編成の場合は，-n オプションを指定してもグローバルバッファを使用します。

2. -n オプションに一括入出力用ローカルバッファ面数だけを指定した場合，一つの RD エリアに対して一つの一括入出力用ローカルバッファを使用します。RD エリアが n 個ある場合でも，一つの一括入出力用ローカルバッファを使用するため，バッファ競合が発生します。バッファ競合が発生すると，逆に入出力回数が増加して性能劣化を引き起こします。このような場合には div を指定してください。div を

指定すると，RD エリア数分の一括入出力用ローカルバッファを確保（一つの RD エリアに対して一つの一括入出力用ローカルバッファを確保）するため，バッファの競合はなくなります。

3. 一括入出力用ローカルバッファを指定しても，条件によってはグローバルバッファを使用することがあります。各条件と使用されるバッファとの関係を次の表に示します。

表 8–19　各条件と使用されるバッファとの関係（pdrorg でのアンロード処理時）

条件			ランダムアクセス用ローカルバッファの指定				
			なし		あり		
			グローバルバッファ	一括入出力用ローカルバッファ	グローバルバッファ	一括入出力用ローカルバッファ	ランダムアクセス用ローカルバッファ
データページ	LOB 列構成基表を格納する RD エリア	FIX 表	−	○	−	○	−
		非 FIX 表	○	−	○	−	−
	LOB 列格納用 RD エリア		○	−	○	−	−
	LOB 属性格納用 RD エリア		○	−	○	−	−
インデクスページ	インデクスからキー値検索する場合（-b 指定時）		○	−	−	−	○
ディレクトリページ			○	−	○	−	−

（凡例）

○：使用します。

−：使用しません。

表 8–20　各条件と使用されるバッファとの関係（pdrorg でのリロード処理時）

条件				ランダムアクセス用ローカルバッファの指定				
				なし		あり		
				グローバルバッファ	一括入出力用ローカルバッファ	グローバルバッファ	一括入出力用ローカルバッファ	ランダムアクセス用ローカルバッファ
データページ	LOB 列構成基表を格納する RD エリア	FIX 表		−	○	−	○	−
		非 FIX 表	列長が 256 バイトを超える可変長文字列，又は BINARY 列を定義している場合	○	−	○	−	−
			抽象データ型列又は繰返し列を定義していて，かつ 1 行のデータが 1 ページに入らない場合	○	−	○	−	−

条件				ランダムアクセス用ローカルバッファの指定				
				なし		あり		
				グローバルバッファ	一括入出力用ローカルバッファ	グローバルバッファ	一括入出力用ローカルバッファ	ランダムアクセス用ローカルバッファ
			上記以外	−	○	−	○	−
		表からキー値検索する場合（-i c 又は n 指定時）		−	○	−	○	−
	LOB 列格納用 RD エリア			○	−	○	−	−
	LOB 属性格納用 RD エリア			○	−	○	−	−
インデクスページ	インデクスの同時作成をする場合（-i s 指定時）			○	−	−	−	○
	インデクスの一括作成をする場合（-i c 指定時）			○	−	−	−	○
ディレクトリページ				○	−	○	−	−

（凡例）

○：使用します。

−：使用しません。

(b)　規則

1. リバランス表の場合，グローバルバッファが十分に割り当てられていないと，極端に性能が劣化することがあります。したがって，定義長が 256 バイトを超える列がないときは一括入出力ページ数を指定するようにしてください。

   定義長が 256 バイト以上の列があるなどの理由で，ローカルバッファ指定が有効にならない場合は，次の面数以上のグローバルバッファを割り当ててください。

   1RD エリアに必要なバッファ面数

   ＝ 1024÷(表格納用 RD エリア数)×2 ＋ 3

2. FIX ハッシュ分割のリバランス表の場合，ハッシュグループごとに指定したページ数のバッファを確保するため，横分割していない表やほかの条件で分割している表よりも，メモリを多量に消費します。

(12)　-f アンロードデータファイル・LOB データのアンロードファイルの種別

アンロードデータファイル，又は LOB データのアンロードファイルの種別を指定します。

hirdb

アンロードデータファイル，又は LOB データのアンロードファイルとして HiRDB ファイルを使用する場合に指定します。

(13)　-b アンロード順指定

アンロードするデータの順番を指定します。

省略した場合，表にクラスタキーが定義されていればクラスタキー順に，クラスタキーが定義されていなければ表のデータの格納順にアンロードします。

なお，インデクス定義条件によっては，指定できないことがあります。-b オプションの指定可否を次の表に示します。

表 8–21　-b オプションの指定可否

条件			指定可否
インデクス種別		定義条件	
B-tree インデクス	部分構造インデクス	−	×
	上記以外	除外キー値指定がある	×
		インデクス構成列に繰返し列がある	×
		上記以外	○
プラグインインデクス		−	×

（凡例）

○：指定できます。

×：指定できません。KFPL15323-E を出力します。

−：該当しません。

(a)　適用基準

クラスタキーが定義されている表をアンロードする場合は，このオプションを省略してください。

クラスタキーが定義されていない表をアンロードして，リロードする前に，表にクラスタキーを定義しようとしている場合は，定義しようとしているクラスタキーと同じ順番（ASC，又は DESC）のインデクス順でアンロードしてください。

アンロードするデータ順として，次のどれかを指定します。

cluster

クラスタキー順にアンロードする場合に指定します。

index，インデクス識別子〔，desc〕

インデクス順にアンロードする場合に指定します。

desc を指定すると，インデクスの降順にアンロードできます。なお，desc は，単一列インデクスの場合にだけ指定できます。

primary〔,,desc〕

主キー順にアンロードする場合に指定します。

desc を指定すると，インデクスの降順にアンロードできます。なお，desc は，単一列インデクスの場合にだけ指定できます。

physical

表のデータの格納順にアンロードする場合に指定します。

＜規則＞

1. インデクス識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。また，インデクス識別子に空白が含まれる場合は，「¥""」と「"¥"」で囲んでください。
2. desc は単一列インデクスの場合だけ指定できます。
3. このオプションを指定した場合，表格納用 RD エリア単位ではアンロードデータファイルに出力されたデータが指定したインデクス順に出力されますが，表全体としてはインデクス順にはなりません。したがって，インデクス順に出力したアンロードデータファイルが必要な場合（-W オプションを指定して DAT 形式のデータを出力する場合など）は，横分割表については RD エリア単位にアンロードデータファイルを作成する必要があります。
4. スキーマ単位の再編成の場合，-b オプションは指定できません。
5. LOB 列をインデクス順にアンロード（-b オプションに index を指定）する場合は，インデクスが作成されていなければなりません。次の場合はインデクスが作成されないので，インデクスを作成してから LOB 列をアンロードする必要があります。
   - -i オプションに n を指定した場合
   - サーバ内分割されていない非分割キーインデクスの定義された表に対して，RD エリア単位にリロードする場合

(14)　-S

CHAR，NCHAR，又は MCHAR 型の列にサプレスオプションを指定している表を再編成する場合に，列データの後方に連続する空白があるとき，その空白を取り除いてアンロードするときに指定します。

(a)　オプション指定時の効果

このオプションを指定すると，処理時間の短縮，及びアンロードデータファイルの容量を削減できます。

-S オプションを指定して，処理時間の短縮，及びアンロードデータファイルの容量を削減できるケースを次に示します。

アンロードした表の状態		リロード先の表の状態			
		サプレスオプションなしの列		サプレスオプションありの列	
		条件 A	条件 B	条件 A	条件 B
サプレスオプションありの列	条件 A	○	○	◎	○
	条件 B	×	×	×	×

（凡例）

条件 A：

該当する列にインデクスが定義されていない，かつキーレンジ分割列，ハッシュ分割列でない場合

条件Ｂ：

該当する列にインデクスが定義されている，又はキーレンジ分割列，若しくはハッシュ分割列の場合

◎：

アンロード，及びリロード時の処理時間が短縮され，アンロードデータファイルの容量が削減されます。

○：

アンロード時の処理時間が短縮され，アンロードデータファイルの容量が削減されます。リロード時の処理時間は，◎の場合よりも遅くなります。

×：

処理時間の短縮，及びアンロードデータファイルの容量削減の効果はありません。

(b) 注意事項

1. -S オプションを指定して，かつサプレスオプションを指定している表の列の場合でも，次に示す列は処理時間の短縮，及びアンロードデータファイルの容量の削減はできません。
   - インデクスが定義されている列
   - キーレンジ分割の列
   - ハッシュ分割の列
2. リロード時に，上記の列の定義が変更されている場合は，定義が変更されていない場合に比べて処理時間は遅くなります。
3. -W オプションを指定している場合，このオプションを指定しても無効となります。

(15) -o

インデクス作成処理が正常終了した後に，インデクス情報ファイルを自動的に削除する場合に指定します。

(a) 適用基準

特にプラグインインデクスを作成するときのインデクス情報ファイルは大容量となり，インデクス情報ファイルを削除しないでおくと，ディスク容量を圧迫することになります。このインデクス情報ファイルの削除し忘れに備え，インデクス作成処理終了後に自動的にインデクス情報ファイルを削除する場合に，このオプションを指定します。

(b) 注意事項

インデクス情報ファイルの内容と，-o オプションの関係を次に示します。

-k オプション	インデクス情報ファイルの内容	-o オプション	
		あり	なし
rorg，reld	index 文で指定したインデクス情報ファイル	○※	×
	HiRDB が割り当てたインデクス情報ファイル	○※	○※
ixmk	pdload，又は pdrorg が作成したインデクス情報ファイル	○	×
	プラグインインデクスのインデクス遅延一括作成機能で作成したインデクス情報ファイル	○	○
ixrc，ixor	idxname 文を指定したときに HiRDB が割り当てたインデクス情報ファイル	○	○
	index 文で指定したインデクス情報ファイル	○	×

（凡例）

○：削除されます。

×：削除されません。

注※

サーバ内で分割している表の場合，RD エリア単位の再編成でサーバ内分割されていない非分割キーインデクスのときは削除されません。

(16) -m 〔経過メッセージ出力間隔〕〔,インフォメーションメッセージ出力抑止レベル〕

経過メッセージ出力間隔 ～<符号なし整数>((1～1000))《10》

処理の実行状況を出力するメッセージの出力間隔を，1 万行単位で指定します。なお，このメッセージは，-m オプションに lvl2 を指定すると，トラブルシュート情報を出力するワークファイルにも出力されます。

＜適用基準＞

省略時仮定値の 10 万行間隔出力が長い場合，又は短い場合に，任意に指定してください。

＜注意事項＞

1. 表へリロードする場合のインデクス一括作成中，及びインデクスの一括作成で指定したときは無効となり，インデクスの作成開始時点，及び終了時点でメッセージが出力されます。
2. -k ixrc の場合は，このオプションの指定を無視します。ただし，プラグインインデクスを処理する場合は有効です。

インフォメーションメッセージ出力抑止レベル　～《lvl0》

実行した端末の標準出力，イベントログファイル，及びメッセージログファイルへのインフォメーションメッセージの出力を抑止します。標準出力，イベントログファイル，及びメッセージログファイルのすべての出力を抑止する場合，トラブルシュート情報はワークファイルに出力します。指定できるオプションとメッセージの出力先を次の表に示します。

表 8–22　オプションの指定値とメッセージの出力先

オプションの指定値	意味	出力先		
		標準出力	イベントログ及びメッセージログファイル	ワークファイル
lvl0	出力を抑止しない	○	○	×
lvl1	標準出力への出力を抑止する	×	○	×
lvl2	標準出力，イベントログ及びメッセージログファイルへの出力を抑止する	×	×	○

（凡例）

○：メッセージを出力します。

×：メッセージを出力しません。

ワークファイルにメッセージを出力するのはトラブルシュート目的（ユティリティの動作を確認するため）であり，通常，メッセージを確認する必要がない場合は lvl1 を指定することをお勧めします。

＜ワークファイル＞

ワークファイルは，-m オプションに lvl2 を指定すると，ユティリティを実行するごとに各サーバに作成されます。このため，pdcspool コマンドで HiRDB によって定期的に削除されるように，pd_tmp_directory オペランドにディレクトリを指定しておくことをお勧めします。ワークファイルの出力先を次の表に示します。

表 8–23　ワークファイルの出力先

システム定義の pd_tmp_directory オペランドの指定	環境変数 TMP の指定	ワークファイルの出力先
あり	–	pd_tmp_directory オペランドに指定したディレクトリ
なし	あり	環境変数 TMP に指定したディレクトリ
	なし	%PDDIR%¥tmp ディレクトリ

（凡例）

－：該当しません。

なお，ファイルオープンエラーや I/O エラーなどのためにインフォメーションメッセージがワークファイルへ出力できない場合は，エラーメッセージが出力されますが，処理は続行します。

ワークファイルは，HiRDB が自動で作成します。ファイル名称は，PDRORG-aa-bb の形式で決められます。それぞれの意味について説明します。

- PDRORG：pdrorg が出力したファイルであることを示す固定のプリフィクスです。
- aa：プロセスのサーバ名称です。
- bb：ファイル作成時間，及びプロセス ID です。

(17)　-X サーバ間通信の応答監視時間

～＜符号なし整数＞　((1～65535)) 《300》

コマンドを実行したサーバで通信障害などの障害が発生した場合，コマンドは無応答となり，業務が停止するおそれがあります。pdrorg では障害を検知するために，コマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が，-X オプションに設定した時間を超えた場合は，pdrorg はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。pdrorg の監視時間には，このほかに pdrorg の実行監視時間があります。サーバ間通信の応答監視時間と pdrorg の実行監視時間との関係については，「exectime=pdrorg の実行監視時間」を参照してください。

＜適用基準＞

- 通信障害やユニットダウンによってサーバから応答が返らない状態となった場合に，300 秒より早く障害を検知したいとき，-X オプションに 300 より小さい値を指定してください。
- 系切り替え機能を使用している場合，系切り替えが完了しているにもかかわらず，コマンドが応答を待ち続けることがあります。その場合は，監視時間を短く設定することで，コマンドを即時終了できます。
- 同時実行するアプリケーションやユティリティの多重度が高い場合など，ディクショナリからの応答が遅延し，-X オプションのデフォルトである 300 秒以内にユティリティの前処理が完了しないときは，監視時間でタイムアウトするおそれがあります。このような環境では，-X オプションには 300 より大きい値を指定してください。
- ディクショナリ表のメンテナンスでインデクスを追加する場合，大量のデータが登録されているディクショナリ表に対して実行すると，処理時間が長くなり，デフォルトである 300 秒を超えるおそれがあります。そのため，応答監視時間には 300 より大きい値を指定してください。

(18)　制御情報ファイル名

～＜パス名＞

pdrorg の制御文を記述した制御情報ファイルの名称を指定します。

制御情報ファイルには，次に示す制御文を記述できます。各制御文については，次を参照してください。

- 「unload 文（アンロードデータファイルの情報の記述）」
- 「index 文（インデクス情報ファイルの情報の記述）」
- 「idxname 文（インデクスの情報の記述）」
- 「idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述）」
- 「sort 文（ソート用ワークディレクトリの情報の記述）」
- 「lobunld 文（LOB データのアンロードファイルの情報の記述）」
- 「unlduoc 文（UOC 格納ライブラリの情報の記述）」
- 「tblname 文（別表への表データ再登録の情報の記述）」
- 「array 文（繰返し列の行データ出力形式の記述）」
- 「unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述）」
- 「reld_func 文（コンストラクタ関数の情報の記述）」
- 「constraint 文（検査保留状態に関する記述）」
- 「option 文（データ処理に関する情報の記述）」
- 「blobtovarchar 文（データ変換をする列の記述）」
- 「fixtext_option 文（固定長データ形式の入力データファイル生成時の出力データに関する記述）」

なお，制御情報ファイル中にコメントは記述できません。

制御文	pdrorg の機能						
	表の再編成	表のアンロード	表へのリロード	ディクショナリ表の再編成	インデクスの一括作成	インデクスの再作成	インデクスの再編成
unload 文	◎	◎	◎	◎	－	－	－
index 文	○	－	○	－	◎	◎※6	◎※6
idxname 文	－	－	－	－	－	◎※6	◎※6
idxwork 文	○※1	－	○※1	－	－	○	○
sort 文	○※2	－	○※2	－	○	○	－
lobunld 文	○	○※3	○	○	－	－	－
unlduoc 文	○	○	－	×	×	×	×
tblname 文	－	－	○	－	－	－	－
array 文	－	○※4	－	－	－	－	－

制御文	pdrorg の機能						
	表の再編成	表のアンロード	表へのリロード	ディクショナリ表の再編成	インデクスの一括作成	インデクスの再作成	インデクスの再編成
unld_func 文	○※5	○※8	−	−	−	−	−
reld_func 文	○※5	−	○	−	−	−	−
constraint 文	○	−	○	−	−	−	−
option 文	○	○※4※7	○	○※7	○※7	○※7	○※7
blobtovarchar 文	−	○※4	−	−	−	−	−
fixtext_option 文	−	○※4	−	−	−	−	−

（凡例）

◎：必ず指定してください。

○：任意に指定できます。

−：指定しても無視されます。

×：指定するとエラーになります。

注※1

-i s の場合は指定できません。

注※2

-i n，及び-i s の場合は指定できません。

注※3

-W オプションを指定した場合は指定できません。

注※4

-W オプションを指定した場合に指定できます。

注※5

unld_func 文，及び reld_func 文を同時に指定する必要があります。

注※6

index 文，又は idxname 文のどちらかを指定してください。

注※7

同期点を指定した再編成及びリロードの場合，指定できません。

注※8

抽象データ型の列がある表を，-W bin 指定でアンロードする場合は必ず指定してください。

(a)　制御文とオプションの関係

●LOB 列，又はプラグインが提供する抽象データ型列（LOB 属性）がある表の場合

LOB 列がある表を再編成する場合，unload 文，及び lobunld 文の指定によって再編成の単位が異なります。LOB 列がある表の再編成の単位と，unload 文，lobunld 文，及び-j オプションとの関係を次に示します。

表の再編成の単位	unload 文		lobunld 文	
	-j あり	-j なし	-j あり	-j なし
LOB 列構成基表及び LOB 列	○	○	×	○
LOB 列構成基表	×	○	×	－
LOB 列	×	－	×	○

（凡例）

○：指定できます。

×：指定できません。

－：不要です。

●インデクスの再作成，又はインデクスの再編成をする場合

インデクスの再作成，又はインデクスの再編成をする場合，idxname 文でインデクス単位に処理，又は index 文でインデクス格納 RD エリア単位に処理できます。インデクスの再作成，又はインデクスの再編成単位と，index 文，idxname 文，idxwork 文，及び sort 文の関係を次に示します。

インデクスの再作成， 又は再編成単位	index 文	idxname 文	idxwork 文	sort 文	
				-k ixrc	-k ixor
インデクス単位	×	◎	○	○	－
インデクス格納 RD エリア単位	◎	×	－	○	－

（凡例）

◎：必ず指定してください。

○：任意に指定できます。

×：指定できません。

－：指定しても無視されます。

(b)　制御文で記述するファイル，及びディレクトリについて

制御文で記述するファイル，及びディレクトリの規則を次に示します。

1. HiRDB 管理者に対してアクセス権限を与えておく必要があります。
2. アンロードデータファイル，及びインデクス情報ファイルのファイル名称は，ホストに関係なくすべて一意にしてください。

3. パス名は大文字，小文字を区別しません。このため，大文字，小文字が異なるだけの同一パス名を指定すると，重複エラーとなります。
4. ネットワークドライブに接続するパス名は指定できません。

8.10.3　unload 文（アンロードデータファイルの情報の記述）

アンロードデータファイルの情報を指定します。

＜適用基準＞

-k rorg，-k unld，及び-k reld の場合は，必ず指定してください。

＜規則＞

unload 文は，次の数だけ指定できます。

【HiRDB/シングルサーバの場合】

再編成の単位（表，又は RD エリア）に関係なく，unload 文は一つだけ指定してください。

【HiRDB/パラレルサーバの場合】

横分割表を表単位に再編成する場合，unload 文は表を格納した RD エリアがあるサーバ数分指定してください。

横分割表を RD エリア単位に再編成する場合，又は分割していない表の再編成の場合には，unload 文は一つだけ指定してください。また，-g オプションを指定して一つのアンロードデータファイルにアンロードする場合にも，unload 文は一つだけ指定してください。

(1)　形式

```
unload　{〔サーバ名：〕
                アンロードデータファイル名〔，アンロードデータファイル名〕…
        |（uoc）}
        〔HiRDBファイル情報〕
        〔uoc情報〕
```

(2)　説明

(a)　サーバ名

～＜識別子＞((1～8))

アンロードデータファイルを作成するサーバの名称を指定します。

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。

【HiRDB/パラレルサーバの場合】

- -g オプション指定なし

  横分割表の表単位の再編成の場合，再編成する表を格納した RD エリアがあるサーバごとにサーバ名を指定してください。

  横分割表の RD エリア単位の再編成，分割していない表の再編成，又はディクショナリ表の再編成の場合，サーバ名を指定する必要はありません。このとき，アンロードデータファイルは表を格納した RD エリアがあるサーバから出力されます。

- -g オプション指定あり

  -g オプションを指定して，一つのアンロードデータファイルにアンロードする場合には，アンロードデータファイルを作成するサーバ名を指定してください。

(b)　アンロードデータファイル名

～<パス名>

アンロードデータファイルの名称を絶対パス名で指定します。

-f オプションでアンロードデータファイル種別に HiRDB ファイルを指定した場合は，次のように指定してください。

<アンロードデータファイルが HiRDB ファイルの場合の規則>

- アンロードデータファイル名は，167 文字以内で指定してください。
- HiRDB ファイルを使用するには，あらかじめ pdfmkfs コマンドで，HiRDB ファイルシステム領域を作成しておく必要があります。この場合，-k オプションの使用目的には UTL を指定してください。
- pdfmkfs コマンドで，-k UTL 以外を指定して作成した HiRDB ファイルシステム領域上に作成した HiRDB ファイルを指定した場合は，エラーとなります。
- 指定したアンロードデータファイル名が HiRDB ファイルシステム領域上にない場合，HiRDB が新規にアンロードデータファイルを作成します。既存のアンロードデータファイル名を指定した場合は，上書きされます。

また，アンロードデータファイルに対してテープ装置アクセス機能を使用できます。テープ装置アクセス機能については，「テープ装置アクセス機能」を参照してください。

<アンロードデータファイルに関する注意事項>

1. -W bin -w オプションを指定して出力した表移行用アンロードファイルは，一つだけ指定できます。
2. 表のデータが一つのアンロードデータファイルに入りきらない場合は，複数のファイルを指定できます。

(c) （uoc）

UOC を利用した再編成をする場合，UOC 側で UOC データファイルを作成する場合（pdrorg 側ではアンロードデータファイルを出力しない場合）に指定します。

（uoc）は，-k unld の場合にだけ指定できます。

(d) HiRDB ファイル情報

〔init=｛初期割り当て領域サイズ

｜（初期割り当て領域サイズ〔，初期割り当て領域サイズ〕...)}〕

〔，incr=｛増分割り当て領域サイズ

｜（増分割り当て領域サイズ〔，増分割り当て領域サイズ〕...)}

アンロードデータファイルが HiRDB ファイルの場合に指定します。HiRDB ファイル情報は，必ず１行で指定してください。

init の指定を省略した場合，incr の前のコンマ（,）は指定しないでください。

init=｛初期割り当て領域サイズ｜（初期割り当て領域サイズ〔，初期割り当て領域サイズ〕...)}

～＜符号なし整数＞((1～1048574))《100》

HiRDB ファイルへの書き込み時に割り当てる容量をメガバイト単位で指定します。

＜見積もり式＞

初期割り当て領域サイズの計算方法については，マニュアル「HiRDB システム導入・設計ガイド」のデータベース再編成ユティリティ（pdrorg）実行時のファイルの容量を参照してアンロードデータファイルのバイト数を求め，次の式で計算してください。

```
初期割り当て領域サイズ
＝↑（アンロードデータファイルのバイト数＋1024）÷（1024×1024）↑
  （単位：メガバイト）
```

＜規則＞

1. 初期割り当てサイズは，pdfmkfs コマンドで HiRDB ファイルシステム領域作成時，-n オプションで指定した値より小さくなければなりません（システムが管理用に使用する領域が必要なため）。
2. pdfmkfs コマンドに-e オプションを指定しなかったり，-e オプションに 0 を指定したりすると，増分割り当てができないので，必ず指定してください。
3. どの程度の大きさの初期割り当てサイズが使用できるかは，pdfstatfs コマンドで確認してください。
4. 割り当て領域サイズの指定は，unload 文で指定したファイル名の数分指定します。両者に不一致があった場合，次のように解釈します。

関係	動作
ファイル数＜指定	ファイル数分，指定を有効にして後は無視します。
ファイル数＝指定	ファイルごとに指定を有効にします。
ファイル数＞指定	指定の最後の領域サイズを足りない指定分繰り返します。

incr=｛増分割り当て領域サイズ｜（増分割り当て領域サイズ〔，増分割り当て領域サイズ〕...)｝

～＜符号なし整数＞((1～1048574))《10》

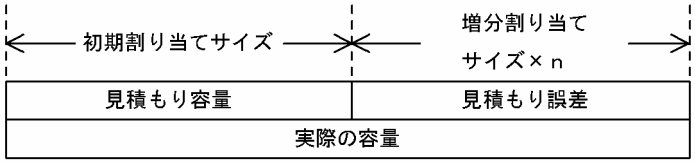
HiRDB ファイル書き込み時に，初期割り当て領域サイズで容量が不足した場合，拡張する割り当て領域の大きさを指定します。

次の場合には HiRDB ファイルの拡張ができません。

- 該当する HiRDB ファイルシステム領域の増分回数指定を超えて拡張しようとしたとき
- 該当する HiRDB ファイルが，個々の HiRDB ファイルの増分回数の上限(23 回固定)を超えて拡張しようとしたとき

したがって，実際の容量が見積もり容量を超える場合は，上記 2 項目の制限に掛からないように，増分割り当てサイズを指定する必要があります。増分割り当てサイズの制限を次の図に示します。図中の n の値は，上記 2 項目の制限を超えないようにしてください。

図 8–21　増分割り当てサイズの制限（unload 文）



HiRDB ファイルシステム領域の増分回数は，新規に HiRDB ファイルシステムを作成する場合，pdfmkfs コマンドの-e オプション指定で設定します。また，既存の HiRDB ファイルシステム領域の状態を見る場合，pdfstatfs コマンドの出力結果の「available expand count」の値によって参照できます。

(e)　UOC 情報

〔uoc_lib=ライブラリ名称〔param='ユーザパラメタ'〕

UOC 情報は，UOC を利用して再編成をする場合に指定してください。unlduoc 文を指定する場合には，必ず UOC 情報も指定してください（指定しないとエラーとなります)。

uoc_lib=ライブラリ名称

～＜パス名＞

UOC を格納しているダイナミックリンクライブラリの名称を，絶対パス名で指定します。

param='ユーザパラメタ'

～＜文字列＞((1～1023))

UOC にパラメタを渡したい場合に，その内容をユーザパラメタに指定します（ユーザは UOC に対して，コマンドラインなどで直接パラメタを渡せないため)。

ここに指定した内容は，UOC インタフェース領域を介して，指定した内容のまま UOC に渡されます。なお，ユーザパラメタに指定する文字列には，空白及びタブは指定できません。

8.10.4　index 文（インデクス情報ファイルの情報の記述）

インデクス情報を出力するインデクス情報ファイルの情報を指定します。

＜適用基準＞

- -k rorg，又は-k reld の場合

  システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，index 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないよう，できるだけ指定するようにしてください。

  インデクスの数，又はインデクスの格納 RD エリア数が多い場合は，idxwork 文を指定すると便利です。

- -k ixmk の場合

  必ず指定してください。

- -k ixrc，又は-k ixor の場合

  インデクス格納 RD エリア単位にインデクスの再作成，又はインデクスの再編成をするときに指定します（インデクス単位に処理する場合は idxname 文を指定します）。

＜規則＞

1. index 文は，インデクス格納用 RD エリアごとに 1 文指定します。
2. インデクスを定義していない表の再編成，及びインデクス更新モードの場合には，index 文を指定する必要はありません。
3. index 文を省略した場合，ユティリティがファイル名称を決定し，インデクス情報ファイルを作成します。作成先ディレクトリは，idxwork 文を省略している場合，インデクスが格納されているサーバの，表「pdrorg がファイルを出力するディレクトリ」に示すディレクトリになります。

   ユティリティが名称を決定してインデクス情報ファイルを作成した場合，インデクス一括作成完了後，ユティリティがファイルを削除しますが，index 文を指定した場合，-o オプションを指定しないとインデクス一括作成完了後にファイルを削除しません。そのため，ユーザがファイルを削除する必要があります。
4. index 文と idxwork 文を同時に指定した場合，index 文の指定が有効となります。

(1)　形式

```
index インデクス識別子 〔RDエリア名〕 インデクス情報ファイル名
```

(2) 説明

(a) インデクス識別子

インデクスのインデクス識別子を指定します。

インデクス識別子を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，インデクス識別子に空白が含まれている場合は，引用符で囲んでください。

(b) RD エリア名

～＜識別子＞((1～30))

横分割表の場合，インデクスを格納している RD エリアの名称を指定します。

RD エリア名を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，RD エリア名に空白が含まれている場合は，引用符で囲んでください。

(c) インデクス情報ファイル名

～＜パス名＞

インデクス情報を出力するインデクス情報ファイルの名称を絶対パス名で指定します。

なお，インデクス遅延一括作成機能で出力したインデクス情報ファイルを使用してインデクスの再作成をする場合，インデクス情報ファイルのファイル名に HiRDB ファイルを使用できます（"HiRDB ファイルシステム領域名/HiRDB ファイル"は 167 文字以内にしてください）。

8.10.5 idxname 文（インデクスの情報の記述）

インデクスの再作成，又はインデクスの再編成をするときに，インデクスの情報を指定します。

＜適用基準＞

インデクス単位にインデクスの再作成，又はインデクスの再編成をするときに指定します（インデクス格納 RD エリア単位に処理する場合は index 文を指定します）。

＜規則＞

1. idxname 文はインデクス数分指定できます。ただし，name=*を指定した場合は，1 文だけ指定できます。複数指定するとエラーになります。
2. idxname 文を使用すると，インデクスを構成するすべてのインデクス格納 RD エリアが処理対象となります。そのため，index 文で複数記述していたものが，idxname 文だと 1 文だけ指定すればよい場合もあります。

3. idxname 文を指定し，かつ idxwork 文を省略した場合，インデクス情報ファイルを出力するディレクトリは，インデクス格納 RD エリアがあるサーバの，表「pdrorg がファイルを出力するディレクトリ」に示すディレクトリになります。

(1) 形式

```
idxname name=インデクス識別子〔server=サーバ名〔，サーバ名〕…〕
```

(2) 説明

(a) name=インデクス識別子

インデクスのインデクス識別子を指定します。

インデクス識別子には，アスタリスク（*）も指定できます。*を指定した場合，表に定義しているすべてのインデクスを再作成します。

インデクス識別子を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，インデクス識別子に空白が含まれている場合は，引用符で囲んでください。

(b) server=サーバ名

～＜識別子＞((1～8))

サーバ名は，HiRDB/パラレルサーバの場合に指定します。

サーバ名を指定した場合，指定したサーバにあるインデクスだけを処理します。省略した場合は，すべてのサーバにあるインデクスを処理対象とします。

(3) オペランドと実行環境との関係

name オペランド，及び server オペランドと，実行環境との関係を次に示します。

name オペランド	server オペランド	実行環境	
		HiRDB/シングルサーバ	HiRDB/パラレルサーバ
インデクス識別子	サーバ名	該当しません。	指定したサーバの，指定したインデクスを構成する，すべてのインデクス格納 RD エリアが対象となります。
	省略	インデクスを構成する，すべてのインデクス格納 RD エリアが対象となります。	すべてのサーバの，指定したインデクスを構成する，すべてのインデクス格納 RD エリアが対象となります。
*	サーバ名	該当しません。	指定したサーバの，表に定義されているすべてのインデクスを構成する，すべてのインデクス格納 RD エリアが対象となります。

name オペランド	server オペランド	実行環境	
		HiRDB/シングルサーバ	HiRDB/パラレルサーバ
	省略	表に定義されているインデクスを構成する，すべてのインデクス格納 RD エリアが対象となります。	すべてのサーバの，表に定義されているすべてのインデクスを構成する，すべてのインデクス格納 RD エリアが対象となります。

8.10.6　idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述）

index 文，又は idxname 文を省略した場合，ユティリティが作成するインデクス情報ファイルの出力ディレクトリの名称を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，idxwork 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないよう，できるだけ指定するようにしてください。

＜規則＞

1. index 文，及び idxwork 文の両方を省略した場合，インデクス格納 RD エリアがあるサーバの，表「pdrorg がファイルを出力するディレクトリ」に示すディレクトリにインデクス情報ファイルが作成されます。
2. idxwork 文は，次の数だけ指定できます。

   【HiRDB/シングルサーバの場合】

   一つだけ idxwork 文を指定します。

   【HiRDB/パラレルサーバの場合】

   分割格納された表のインデクスの場合，分割格納先のサーバの数だけ idxwork 文を指定します。分割格納されていない表や，分割格納された表を RD エリア単位に再編成する場合には一つだけ指定します。指定するサーバ名は，インデクス格納 RD エリアがあるサーバの名称を指定します。
3. idxwork 文と index 文を同時に指定した場合，index 文の指定が有効となり，idxwork 文の指定は無視されます。

(1)　形式

```
idxwork 〔サーバ名〕 ディレクトリ名
```

(2)　説明

(a)　サーバ名

～＜識別子＞((1～8))

インデクス情報ファイルを作成するサーバの名称を指定します。

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。シングルサーバの名称を指定しても無視されます。

【HiRDB/パラレルサーバの場合】

インデクス情報ファイルを作成するサーバの名称を指定します。

(b)　ディレクトリ名

～<パス名>((1～255))

インデクス情報ファイルを作成するディレクトリの名称を絶対パス名で指定します。

(3)　注意事項

自動作成されるインデクス情報ファイルの名称を次に示します。

```
ディレクトリ名¥INDEX-インデクス名-インデクス格納用RDエリア名
--一意となる文字列
```

（例）

idxwork 文のディレクトリ名が c:¥hirdb¥pdload，インデクス名が IDX1，インデクス格納用 RD エリア名が USER01 の場合，作成されるインデクス情報ファイルの名称は次のようになります。

```
c:¥hirdb¥pdload¥INDEX-IDX1-USER01-aaaa00001
```

注意事項

インデクス名又はインデクス格納 RD エリア名に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

8.10.7　sort 文（ソート用ワークディレクトリの情報の記述）

インデクスを作成するときに使用するソート用ワークファイルの情報（ファイルの作成ディレクトリなど）を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，sort 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないよう，できるだけ指定するようにしてください。

＜規則＞

1. sort 文を省略した場合，インデクス格納 RD エリアがあるサーバの，表「pdrorg がファイルを出力するディレクトリ」に示すディレクトリに作成されます。
2. ソート用ワークファイルは，pdrorg の処理終了後に自動的に削除されます。
3. sort 文は，次の数だけ指定できます。

   【HiRDB/シングルサーバの場合】

   一つだけ sort 文を指定します。

   【HiRDB/パラレルサーバの場合】

   分割格納された表のインデクスの場合，分割格納先のサーバの数だけ sort 文を指定します。分割格納された表のインデクスの場合でも，RD エリア単位に再編成する場合には，一つだけ sort 文を指定します。また，分割格納されていない表のインデクスの場合も，一つだけ sort 文を指定します。

(1)　形式

```
sort 〔サーバ名〕　ディレクトリ名
```

(2)　説明

(a)　サーバ名

～＜識別子＞((1～8))

ソート用ワークファイルを作成するサーバの名称を指定します。

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。シングルサーバの名称を指定しても無視されます。

【HiRDB/パラレルサーバの場合】

ソート用ワークファイルを作成するサーバの名称を指定します。

(b)　ディレクトリ名

～＜パス名＞

ソート用ワークファイルを作成するディレクトリの名称を，絶対パス名で指定します。

8.10.8　lobunld 文（LOB データのアンロードファイルの情報の記述）

LOB 列がある表を再編成する場合，LOB データのアンロードファイルの情報を指定します。

＜適用基準＞

LOB 列がある表を再編成する場合に指定します。

ただし，次のどちらかに該当する場合は指定しないでください。

- -j オプションを指定している場合
- LOB 列構成基表だけを再編成する場合

＜規則＞

1. lobunld 文は，次の数だけ指定できます。

   【HiRDB/シングルサーバの場合】

   再編成の単位（表，又は RD エリア）に関係なく，lobunld 文は一つだけ指定してください。

   【HiRDB/パラレルサーバの場合】

   横分割表の場合で LOB 列だけを再編成するとき，lobunld 文は LOB 列を格納した RD エリアがあるサーバ数分指定してください。RD エリア単位の再編成，又は分割していない表の再編成の場合には，lobunld 文は一つだけ指定してください。

   また，-g オプションを指定して一つの LOB データのアンロードファイルにアンロードする場合は，lobunld 文は一つだけ指定してください。

2. 同期点を指定した再編成又はリロードの場合，lobunld 文は指定できません。

(1) 形式

```
lobunld 〔サーバ名：〕
          LOBデータのアンロードファイル名
               〔，LOBデータのアンロードファイル名〕
          〔HiRDBファイル情報〕
```

(2) 説明

(a) サーバ名

～＜識別子＞((1～8))

LOB データのアンロードファイルを作成するサーバの名称を指定します。

【HiRDB/シングルサーバの場合】

HiRDB/シングルサーバの場合は指定しないでください。

【HiRDB/パラレルサーバの場合】

横分割表の場合で複数の LOB 列を再編成するとき，LOB データのアンロードファイルを作成するサーバの名称を指定します。横分割表の RD エリア単位の再編成，又は横分割していない表の場合には，指定する必要はありません。

-g オプションを指定して一つの LOB データのアンロードファイルにアンロードする場合には，LOB データのアンロードファイルを作成するサーバの名称を指定してください。

(b)　LOB データのアンロードファイル名

LOB データのアンロードファイルの名称を絶対パス名で指定します。

-f オプションで LOB データのアンロードファイル種別として HiRDB ファイルを指定した場合は，次のように指定してください。

＜ LOB データのアンロードファイルが HiRDB ファイルの場合の規則＞

- LOB データのアンロードファイル名は，167 文字以内で指定してください。
- HiRDB ファイルを使用するには，あらかじめ pdfmkfs コマンドで，HiRDB ファイルシステム領域を作成しておく必要があります。この場合，-k オプションには UTL を指定してください。
- pdfmkfs コマンドで，-k オプションに UTL 以外を指定して作成した HiRDB ファイルシステム領域上に作成した HiRDB ファイルを指定した場合は，エラーとなります。
- 指定した LOB データのアンロードファイル名が HiRDB ファイルシステム領域上にない場合，HiRDB が新規に LOB データのアンロードファイルを作成します。既存の LOB データのアンロードファイル名を指定した場合は，上書きされます。

また，LOB データのアンロードファイルに対してテープ装置アクセス機能を使用できます。テープ装置アクセス機能については，「テープ装置アクセス機能」を参照してください。

＜ LOB データのアンロードファイルに関する注意事項＞

1. 表のデータが一つの LOB データのアンロードファイルに入りきらない場合は，複数のファイルを指定できます。

(c)　HiRDB ファイル情報

〔init=｛初期割り当て領域サイズ

｜（初期割り当て領域サイズ〔，初期割り当て領域サイズ〕...)}〕

〔，incr=｛増分割り当て領域サイズ

｜（増分割り当て領域サイズ〔，増分割り当て領域サイズ〕...)}〕

LOB データのアンロードファイルが HiRDB ファイルの場合に指定します。HiRDB ファイル情報は，必ず１行で指定してください。

init を省略した場合，incr の前のコンマ（,）は指定しないでください。

init=｛初期割り当て領域サイズ｜（初期割り当て領域サイズ〔，初期割り当て領域サイズ〕...)}

～＜符号なし整数＞((1～1048574))《100》

HiRDB ファイルへの書き込み時に割り当てる容量をメガバイト単位で指定します。

＜見積もり式＞

初期割り当て領域サイズの計算方法については，マニュアル「HiRDB システム導入・設計ガイド」のデータベース再編成ユティリティ（pdrorg）実行時のファイルの容量を参照してください。そこで LOB データのアンロードファイルのバイト数を求め，次の式で計算します。

```
初期割り当て領域サイズ
＝↑（LOBデータのアンロードファイルのバイト数＋1024）
  ÷（1024×1024）↑
  （単位：メガバイト）
```

＜規則＞

1. 初期割り当てサイズは，pdfmkfs コマンドで HiRDB ファイルシステム領域作成時，-n オプションで指定した値より小さくなければなりません（システムが管理用に使用する領域が必要なため）。
2. pdfmkfs コマンドに-e オプションを指定しなかったり，-e オプションに 0 を指定したりすると，増分割り当てができないので，必ず指定してください。
3. どの程度の大きさの初期割り当てサイズが使用できるかは，pdfstatfs コマンドで確認してください。
4. 割り当て領域サイズの指定は，lobunld 文で指定したファイル名の数分指定します。両者に不一致があった場合，次のように解釈します。

関係	動作
ファイル数＜指定	ファイル数分，指定を有効にして後は無視します。
ファイル数＝指定	ファイルごとに指定を有効にします。
ファイル数＞指定	指定の最後の領域サイズを足りない指定分繰り返します。

incr=｛増分割り当て領域サイズ｜（増分割り当て領域サイズ〔，増分割り当て領域サイズ〕...)｝

～＜符号なし整数＞((1～1048574))《10》

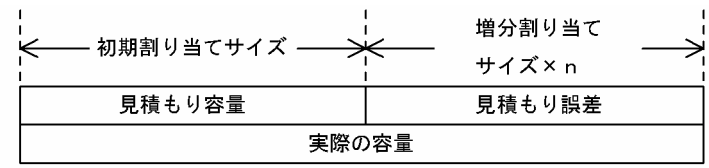
HiRDB ファイル書き込み時に，初期割り当て領域サイズで容量が不足した場合，拡張する割り当て領域の大きさを指定します。

次の場合には HiRDB ファイルの拡張ができません。

- 該当する HiRDB ファイルシステム領域の増分回数指定を超えて拡張しようとしたとき
- 該当する HiRDB ファイルが，個々の HiRDB ファイルの増分回数の上限(23 回固定)を超えて拡張しようとしたとき

したがって，実際の容量が見積もり容量を超える場合は，上記 2 項目の制限に掛からないように，増分割り当てサイズを指定する必要があります。増分割り当てサイズの制限を次の図に示します。図中の n の値は，上記 2 項目の制限を超えないようにしてください。

図 8–22　増分割り当てサイズの制限（lobunld 文）



HiRDB ファイルシステム領域の増分回数は，新規に HiRDB ファイルシステムを作成する場合，pdfmkfs コマンドの-e オプション指定で設定します。また，既存の HiRDB ファイルシステム領域の状態を見る場合，pdfstatfs コマンドの出力結果の「available expand count」の値によって参照できます。

8.10.9　unlduoc 文（UOC 格納ライブラリの情報の記述）

UOC を利用して再編成をする場合，UOC の情報を指定します。

＜適用基準＞

UOC を利用して再編成する場合に指定します。

＜規則＞

unlduoc 文は一つだけ指定できます。

(1)　形式

```
unlduoc entry=関数名
        〔bloblimit=メモリ領域確保サイズ〕
        〔fixrow= {y | n} 〕
```

(2)　説明

(a)　entry=関数名

ライブラリ中の呼び出す関数名を指定します。

(b)　bloblimit=メモリ領域確保サイズ

～＜符号なし整数＞((1～2097152))

UOC に渡すデータを保持するメモリ領域の大きさの上限値を，キロバイト単位で指定します。次の条件に該当する場合，UOC に渡すデータをすべてメモリ領域に保持します。

- 表に抽象データ型列があり，unld_func 文に対応するコンストラクタパラメタ逆生成関数を指定する場合，指定した関数の返却値が長大データ型のとき
- 表に長大データ型の列がある場合

上記の条件の列が複数ある場合，該当する列すべてに対してこのオペランドの指定値が有効となります。

＜適用基準＞

次の条件のどれかを満たす場合に指定してください。

- 表に抽象データ型列があり，unld_func 文に対応するコンストラクタパラメタ逆生成関数を指定する場合，指定した関数の返却値が長大データ型のとき
- 列の定義長でデータのメモリ領域を確保すると，メモリ不足が予測される場合
- 列の定義長よりも実データが明らかに小さい場合

＜規則＞

1. 長大データ型の列の場合，指定したメモリ領域確保サイズよりも列の定義長が小さいとき，その列に対しては列の定義長がメモリ領域確保サイズとなります。
2. 指定した大きさを超えるデータがある場合（指定値よりも列の定義長が小さい場合は，列の定義長を超えるデータがあるとき），エラーとなります。

(c)　fixrow= {y ｜ n}

対象となる表が FIX 表の場合，pdrorg が UOC に渡すデータの格納方法を指定します。非 FIX 表の場合に y を指定するとエラーになります。

y：

列の定義順にデータを詰めて（連続して）渡します。pdrorg が渡したデータを一括して出力する場合は，y を指定するとよいです（アラインメントの保証が必要なデータ型は直接参照できません）。

n：

データ型のバウンダリに従い，データの開始アドレスを補正して渡します。データを直接参照する場合は，n を指定するとよいです（アラインメントを保証する必要があるデータ型の開始アドレスが補正されているため，データを別領域にいったんコピーしなくても直接参照できます）。

8.10.10　tblname 文（別表への表データ再登録の情報の記述）

アンロードした表とは別の表に，表データをリロードする場合に指定します。

＜適用基準＞

同一システムの別表，又は別システムの表へデータを移行する場合に指定します。

＜規則＞

1. tblname 文は，次のような場合には指定できません。
   - -j オプション指定なしで，リロードする表，又はアンロードした表に，LOB 列，又は LOB 属性を含む抽象データ型の列がある場合
   - リロードする表とアンロードした表の表定義が異なっている場合
2. tblname 文を指定する場合は unload 文が一つ以上なければなりません。また，tblname 文は二つ以上指定できません。

(1) 形式

```
tblname {〔認可識別子.〕表識別子 | 認可識別子}
```

(2) 説明

(a) 〔認可識別子.〕表識別子

アンロードした表の〔認可識別子.〕表識別子を指定します。

〔認可識別子.〕表識別子の指定方法については，-t オプションの説明を参照してください。

＜規則＞

1. 認可識別子を省略した場合，-t オプションで指定した認可識別子，又は HiRDB に接続した認可識別子が仮定されます。したがって，アンロードした表の認可識別子とリロードする表の認可識別子が異なる場合は，必ず認可識別子を指定してください。
2. 抽象データ型がある表へリロードする場合，プラグインによってはアンロード時にコンストラクタパラメタ逆生成関数を指定する必要があるものもあります。
3. 〔認可識別子.〕表識別子を指定した場合，-t オプションには〔認可識別子.〕all を指定できません。

(b) 認可識別子

スキーマ単位に別表へリロードする場合に，その認可識別子を指定します。

この場合，-t オプションには〔認可識別子.〕all を指定する必要があります。

8.10.11 array 文（繰返し列の行データ出力形式の記述）

繰返し列がある表をアンロードする場合に，繰返し列の行データ出力形式を指定します。

＜適用基準＞

繰返し列がある表をアンロードする場合で，かつ-W オプションを指定して DAT 形式，又は拡張 DAT 形式で出力するときに指定できます。

＜規則＞

1. DAT 形式，及び拡張 DAT 形式以外で指定しても無視します。
2. 繰返し列がある表をアンロードする場合に，この制御文を省略したときは，行データ出力形式は ff となります。

(1) 形式

```
array 〔elmtype=行データ出力形式〕
```

(2) 説明

(a) 〔elmtype=行データ出力形式〕

～《ff》

ff，又は vv を指定します。

出力するアンロードデータを，データベース作成ユティリティの入力データファイルとする場合は，必ず行データ形式に ff を指定するか，又は省略してください。

ff

表定義時に該当する列に指定した最大要素数分のデータをセパレータ文字で区切って出力します。

データの格納されていない要素，ナル値が格納されている要素はセパレータ文字を続けて出力します。列値がナル値の場合は，最大要素数分セパレータ文字を出力します。

なお，出力されたデータをデータベース作成ユティリティ（pdload）の入力データファイルとする場合は，適切なナル値オプションを指定してナル値の扱いを指定してください。

ff を指定した場合のデータ出力形式については，「データベース作成ユティリティの入力用ファイルの形式」を参照してください。

vv

実際にデータが格納されている要素数分セパレータ文字でデータを区切って出力し，配列データの先頭に出力した要素数を出力します。

列値がナル値の場合は，要素数に 0 を出力します。

vv を指定した場合のデータ出力形式については，「データベース作成ユティリティの入力用ファイルの形式」を参照してください。

8.10.12 unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述）

アンロード時に抽象データ型のデータ値を逆生成する，コンストラクタパラメタ逆生成関数の情報を指定します。

＜適用基準＞

アンロード機能を持つ，プラグインが提供している抽象データ型がある表を再編成する場合に指定します。

＜規則＞

1. unld_func 文は，表の列に定義されている抽象データ型の数分指定できます。
2. -k rorg の場合に unld_func 文を指定するときは，同時に reld_func 文も指定してください。

(1) 形式

```
unld_func type=〔認可識別子.〕抽象データ型名,
          func=関数名（引数の型〔,引数の型…〕）
           〔,func=関数名（引数の型〔,引数の型…〕）…〕
```

(2) 説明

(a) type=〔認可識別子.〕抽象データ型名

抽象データ型の認可識別子，及び抽象データ型名を指定します。

＜規則＞

1. 認可識別子を省略した場合，抽象データ型を定義したユーザの認可識別子（通常は MASTER）が仮定されます。
2. 認可識別子，及び抽象データ型の名称に小文字，空白を含む場合，引用符（”）で囲んでください。

(b) func=関数名（引数の型〔,引数の型...〕）

コンストラクタパラメタ逆生成関数の名称，引数の型を指定します。名称，引数の型については，各プラグインのマニュアルを参照してください。

関数名

コンストラクタパラメタ逆生成関数の名称を指定します。

引数の型

コンストラクタパラメタ逆生成関数の引数のデータ型を指定します。引数のデータ型は，次の形式で指定してください。

記述方法	備　考
integer	INT32
smallint	INT16
char nchar mchar	CHAR8
varchar nvarchar mvarchar	p_pdb_varchar_t
float	DOUBLE64
smallflt	REAL64
blob	BLOB

記述方法	備　考
抽象データ型名	なし

8.10.13　reld_func 文（コンストラクタ関数の情報の記述）

リロード時に抽象データ型の値を生成する，コンストラクタ関数の情報を指定します。

＜適用基準＞

アンロード機能を持つ，プラグインが提供している抽象データ型がある表を再編成する場合に指定します。

＜規則＞

1. reld_func 文は，表の列に定義されている抽象データ型の数分指定できます。
2. -k rorg の場合に reld_func 文を指定するときは，同時に unld_func 文も指定してください。

(1)　形式

```
reld_func type=〔認可識別子.〕抽象データ型名,
          func=関数名（引数の型〔,引数の型…〕）
```

(2)　説明

(a)　type=〔認可識別子.〕抽象データ型名

抽象データ型の認可識別子，及び抽象データ型名を指定します。

＜規則＞

1. 認可識別子を省略した場合，抽象データ型を定義したユーザの認可識別子（通常は MASTER）が仮定されます。
2. 認可識別子，及び抽象データ型の名称に小文字，空白を含む場合，引用符（”）で囲んでください。

(b)　func=関数名（引数の型〔,引数の型...〕）

コンストラクタ関数の名称，引数の型を指定します。名称，引数の型については，各プラグインのマニュアルを参照してください。

関数名

コンストラクタ関数の名称を指定します。

引数の型

コンストラクタ関数の引数のデータ型を指定します。

引数のデータ型の形式については，「unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述）」の引数の型を参照してください。ただし，抽象データ型名は指定できません。

8.10.14　constraint 文（検査保留状態に関する記述）

検査保留状態に関する指定をします。

検査制約又は参照制約を定義した表の再編成では，UOC を使用して被参照表中の参照表が参照している行を削除した場合を除いて，制約の整合性が保証できるため，検査保留状態は変更しません。UOC を使用して被参照表を再編成する場合だけ検査保留状態を設定します。

また，リロードでは，古いアンロードデータファイルを使用してリロードしたり，別表へのリロードをしたりする場合，制約の整合性が保証できなくなる可能性があります。そのため，該当する表や，その表に関係する参照表を検査保留状態に設定します。

なお，ユーザ自身が制御文を指定して，検査保留状態を変更（設定）しないこともできます。検査保留状態については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

＜適用基準＞

整合性を保証できる場合は，constraint 文を指定することをお勧めします（表の検査保留状態を変更（設定）しない運用）。ただし，次に示す条件に該当する場合は，制約の整合性がなくなる可能性があるため，constraint 文を省略することをお勧めします。

- 古いアンロードデータファイルを使用したリロード

  参照表，又は被参照表のアンロード後に，アンロードした表のデータが更新された場合，表更新前のアンロードデータファイルを使用してリロードすると，更新前の表データに置き換わるため，被参照表及び参照表間のデータの整合性を保証できないことがあります。

- 別表又は制約定義を変更した表へのリロード

  アンロードした表とは別の表へリロードする場合，pdrorg はデータの整合性チェックをしないため，制約の整合性が保証できないデータをリロードすることがあります。

- UOC で行を削除する再編成

  被参照表の再編成時に UOC を使用する場合，UOC を使用して行データを削除すると，整合性がなくなる可能性があります。

＜規則＞

1. constraint 文は，再編成（-k rorg）又はリロード（-k reld）で，かつシステム定義の pd_check_pending オペランドの値が USE の場合に指定できます。

   システム定義の pd_check_pending オペランドの値が NOUSE の場合，又はほかの機能（アンロード（-k unld）など）の場合は，この制御文の指定は無視され，表の検査保留状態を変更しません。

(1)　形式

```
constraint 〔pending=no〕
           〔ref_pending=no〕
```

(2)　説明

(a)　pending=no

参照表又は検査制約を定義した表に対して，再編成又はリロードをする場合，その表の検査保留状態を変更（設定）しないときに指定します。このオペランドを省略した場合の仮定値は，「検査保留状態の設定範囲」を参照してください。

(b)　ref_pending=no

被参照表に対して再編成又はリロードをする場合，その表に関係する参照表の検査保留状態を変更（設定）しないときに指定します。このオペランドを省略した場合の仮定値は，「検査保留状態の設定範囲」を参照してください。

(3)　注意事項

1. 被参照表に対して再編成又はリロードをする場合，その表に関係する参照表を検査保留状態に設定するときの注意事項については，「コマンド実行時の RD エリアの状態」を参照してください。
2. 検査保留状態を設定する場合，リロードした件数が 0 件で，かつ制約の整合性が保証できるときでも，表を検査保留状態に設定します。
3. 検査保留状態を設定する再編成実行時にエラーが発生し，再編成を再実行した場合は，再度，表を検査保留状態に設定します。
4. 検査保留状態を設定する場合，関連する資源に対して排他が掛かります。また，検査保留状態の設定が完了したとき，掛かっていた排他が解除されます。排他制御については，「コマンド実行時の排他制御モード」を参照してください。

(4)　検査保留状態の設定範囲

pdrorg 実行条件と検査保留状態の設定有無の関係を次の表に示します。

表 8–24　pdrorg 実行条件と検査保留状態の設定有無

実行条件					検査保留状態設定有無		
システム定義の pd_check_pending オペランドの値	実行機能	constraint 文のオペランド指定		UOC 使用	対象表		
					被参照表	参照表	検査制約が定義された表
USE	再編成	pending=no	あり	あり	×	×	×

実行条件					検査保留状態設定有無		
システム定義の pd_check_pending オペランドの値	実行機能	constraint 文のオペランド指定		UOC 使用	対象表		
					被参照表	参照表	検査制約が定義された表
				なし	×	×	×
			なし	あり	×	×	×
				なし	×	×	×
		ref_pending=no	あり	あり	×	×	×
				なし	×	×	×
			なし	あり	○	×	×
				なし	×	×	×
	リロード	pending=no	あり	−	×	×	×
			なし	−	×	○	○
		ref_pending=no	あり	−	×	×	×
			なし	−	○	×	×
	上記以外	−	−	−	×	×	×
USE 以外	−	−	−	−	×	×	×

(凡例)

○：検査保留状態を設定します。

×：検査保留状態を変更しません（現在の状態を維持します）。

−：該当しません。

再編成又はリロード実行時の，検査制約の検査保留状態の設定範囲と参照制約の検査保留状態の設定範囲を次の表に示します。

表 8–25　検査制約の検査保留状態の設定範囲

実行単位	検査制約の検査保留状態の設定範囲		
	ディクショナリ表		RD エリア中の表情報
	SQL_TABLES 表 CHECK_PEND2 列	SQL_CHECKS 表 CHECK_PEND2 列	検査制約の状態
表	設定	設定	◎
RD エリア	設定	設定	○
サーバ	設定	設定	●

（凡例）

設定：検査保留状態を設定します。

◎：表を格納するすべての RD エリア中の表情報（検査制約の状態）を設定します。

○：-r オプションで指定した RD エリア中の表情報（検査制約の状態）を設定します。

●：指定したすべての unload 文中のバックエンドサーバ内の表を格納する，すべての RD エリア中の表情報（検査制約の状態）を設定します。

表 8–26　参照制約の検査保留状態の設定範囲

実行単位	対象表					
	被参照表			参照表		
	被参照表に関連する参照表の検査保留状態			参照表の検査保留状態		
	ディクショナリ表		RD エリア中の表情報	ディクショナリ表		RD エリア中の表情報
	SQL_TABLES 表 CHECK_PEND 列	SQL_REFERENTIAL_CONSTRAINTS 表 CHECK_PEND 列	参照制約の状態	SQL_TABLES 表 CHECK_PEND 列	SQL_REFERENTIAL_CONSTRAINTS 表 CHECK_PEND 列	参照制約の状態
表	設定	設定	◎	設定	設定	◎
RD エリア	設定	設定	◎	設定	設定	○
スキーマ	設定	設定	◎	設定	設定	◎
サーバ	設定	設定	◎	設定	設定	●

（凡例）

設定：検査保留状態を設定します。

◎：表を格納するすべての RD エリア中の表情報（検査制約の状態）を設定します。

○：-r オプションで指定した RD エリア中の表情報（検査制約の状態）を設定します。

●：指定したすべての unload 文内のバックエンドサーバ内の表を格納する，すべての RD エリア中の表情報（検査制約の状態）を設定します。

注

UOC を使用する場合は検査保留状態を設定します。

(5)　検査保留状態の設定可否の例

次に示す表 T1～T5 について，constraint 文の pending 及び ref_pending オペランドを指定した場合の，表の検査保留状態の設定可否について説明します。

（表の説明）

T1：主キーを持つ表（T2 に対する被参照表）

T2：T1 の主キーを参照する外部キーを持ち，さらに自身が主キーを持つ表（T1 への参照表であり，かつ T3 に対する被参照表である表）

T3：T2 の主キーを参照する外部キーを持つ表（T2 への参照表）

T4：検査制約を定義した表

T5：制約定義がない表

表 T1～T5 に対してリロードを実行したときの表の検査保留状態を次の表に示します。

表 8–27　リロードを実行したときの表の状態

リロード対象表と制約定義有無				constraint 文のオペランドの指定有無		表の検査保留状態				
表名	参照制約		検査制約	pending	ref_pending	T1	T2	T3	T4	T5
	被参照表	参照表								
T1	○	×	×	なし	なし	−	P	−	−	−
					あり	−	N	−	−	−
				あり	なし	−	P	−	−	−
					あり	−	N	−	−	−
T2	○	○	×	なし	なし	−	P	P	−	−
					あり	−	P	N	−	−
				あり	なし	−	N	P	−	−
					あり	−	N	N	−	−
T3	×	○	×	なし	なし	−	−	P	−	−
					あり	−	−	P	−	−
				あり	なし	−	−	N	−	−
					あり	−	−	N	−	−
T4	×	×	○	なし	なし	−	−	−	P	−
					あり	−	−	−	P	−
				あり	なし	−	−	−	N	−
					あり	−	−	−	N	−
T5	×	×	×	なし	なし	−	−	−	−	−
					あり	−	−	−	−	−
				あり	なし	−	−	−	−	−
					あり	−	−	−	−	−

（凡例）

P：検査保留状態を設定します。

N：検査保留状態を変更しません（現在の状態を維持します）。

−：該当しません（pending 及び ref_pending オペランドの指定を無視します）。

○：定義あり。

×：定義なし。

(6)　検査保留状態を設定するタイミング

再編成又はリロードの対象となる表や，該当する表に関係する参照表には，再編成又はリロード処理を実行する前に，検査保留状態を設定します。なお，表の検査保留状態を設定した後，エラーの発生などによって pdrorg がロールバックした場合でも，表の検査保留状態の設定は完了（コミットが完了）しているため，表は検査保留状態となります。

スキーマ単位の場合は，再編成又はリロード処理を実行する前に，対象となるすべての表を一つずつ検査保留状態に設定します。そのため，表の検査保留状態の設定処理がエラーとなった場合，pdrorg はエラー終了するため，エラーになった表以前に処理した表は検査保留状態になりますが，エラーになった表より後の表は検査保留状態に変更されません。

8.10.15　option 文（データ処理に関する情報の記述）

データ処理に関するオプション機能を使用する場合に指定します。

＜適用基準＞

次のどれかのオプション機能を使用したい場合に指定します。

- 空白変換レベルの指定（spacelvl オペランド）
- リロード時の，空き領域比率の変更（tblfree オペランド）
- インデクス作成時の，空き領域比率の変更（idxfree オペランド）
- 同期点指定の再編成，又は同期点指定のリロード（job オペランド）
- pdrorg の実行時間の監視（exectime オペランド）
- 表の再編成の再実行時に，表のステータスを強制的に通常状態に戻す指定（tblstatus オペランド）
- アンロードする表の排他制御モードの変更（unldenq オペランド）
- UTF-8 の列データを，UTF-16 のアンロードデータファイルでアンロードする場合の指定（charset オペランド）

(1)　形式

```
option 〔spacelvl= {0 | 1 | 3} 〕
       〔tblfree= {未使用領域比率 | （〔未使用領域比率〕，セグメント内空きページ比率）} 〕
```

```
〔idxfree=未使用領域比率〕
〔job= {ジョブ名〔, 同期点行数〕〔, CLR〕 |, , CLR} 〕
〔exectime=pdrorgの実行監視時間〕
〔tblstatus=clear〕
〔unldenq= {tblenq | rdenq | nowait} 〕
〔charset= {utf-16be | utf-16le} 〕
```

(2) 説明

(a) spacelvl= {0 | 1 | 3}

再編成するときに，データの空白変換をするかどうかを指定します。0，1，及び 3 のことを空白変換レベルといいます。

0：

データの空白変換をしません。

1：

リロード時にアンロードデータの空白変換をします。

＜適用基準＞

次のような場合に指定します。

- 既存データの空白文字を統一したい場合
- 別システムに表データを移行する場合に，移行先システムで空白文字を統一したいとき

3：

1 と同様の空白変換をして，更に-W オプション指定のアンロード時にアンロードデータの空白変換をします。

＜適用基準＞

-W オプション指定で出力したアンロードデータを UAP などで使用する場合に，そのアンロードデータの空白文字を統一したいとき

＜注意事項＞

- 各指定値による空白変換の有無を次に示します。

pdrorg の処理内容	オプション			spacelvl の指定値		
	-c	-k	-W	0	1	3
再編成	user	rorg	−	×	○※1	○※1
アンロード	user	unld	あり	×	×	○※2
			なし	×	×	×
リロード	user	reld	−	×	○※1	○※1
ディクショナリ表の再編成	dic	rorg	−	×	×	×

（凡例）

○：空白変換をします。

×：空白変換をしません。

－：該当しません。

注※1

処理対象の表の列が各国文字列型，又は混在文字列型の場合，リロード時に次のようにアンロードデータの空白変換をします。

各国文字列型の列の場合：

アンロードデータ中の連続する半角空白 2 バイトを，全角空白 1 文字に変換します。

ただし，先頭から 2 バイト単位で変換します。

混在文字列型の列の場合：

アンロードデータ中の全角空白 1 文字を，半角空白 2 バイトに変換します。

文字コードが utf-8 の場合，全角空白 1 文字（3 バイト）を半角空白 2 バイトに変換します。MCHAR の場合は，データ長が短くなった分，定義長まで後ろから半角空白が埋められます。MVARCHAR の場合は，データ長が短くなったままとなります。

注※2

処理対象の表の列が各国文字列型の場合，アンロード時にアンロードデータ中の全角空白 1 文字を，半角空白 2 バイトに変換します。

- 抽象データ型がある表の場合，抽象データ型のリロード方式によって，次のように空白変換をします。

抽象データ型のリロード方式	コンストラクタ関数の引数のデータ型	option 文の spacelvl オペランドの指定値	
		省略，又は 0	1，又は 3
プラグインのアンロード機能を使用してアンロードしたアンロードデータファイルからリロードする	－	×	×
プラグインのコンストラクタパラメタ逆生成関数を使用したアンロードデータファイルから，コンストラクタ関数を使用してリロードする	NCHAR，NVARCHAR，MCHAR，MVARCHAR	×	○※
	上記以外	×	×
上記の機能を使用しないでリロードする	－	×	×

（凡例）

○：空白変換をします。

×：空白変換をしません。

－：該当しません。

注※

コンストラクタ関数の引数のデータ型（対応する抽象データ型の reld_func 文の func オペランドで指定した引数の型）に対して，空白変換をします。

- クラスタキー，又はインデクスキーの構成列に各国文字列型を含む場合に，キー順アンロードをするときは，spacelvl の指定に関係なくデータベースに格納されているデータの値の順にアンロードします。そのため，-W オプション指定のアンロードで空白変換をした場合，出力後のデータはキー順にならないことがあります。この場合，データロード後に，再度，空白変換をしない再編成をすれば，キー順に格納されます。

  また，サーバ間の横分割表の場合，空白変換の有無に関係なくクラスタキー，又はインデクスキー順のアンロードと同様に，必ずキー順にアンロードされるとは限らないので，キー順にデータを格納したいときは，アンロード後にユーザがソートをする必要があります。
- 空白文字コードが統一されていないデータベースに対して，option 文に spacelvl=1，又は 3 を指定して再編成する場合，次のような現象が発生します。
  - キー順アンロードしたデータが，キー順にならないことがあります。この場合，再度，空白変換をしない再編成をすれば，キー順に格納されます。
  - 空白変換対象の列が，ユニークキーインデクス又は主キーインデクスの構成列の場合，重複キーが発生することがあります。この場合，ユニークキー又は主キーでなければならないか検討し，必要ならば，空白変換をする前にデータを修正（キー値の修正，又は不要な行の削除など）してください。
  - 空白変換対象の列が分割キーの構成列の場合，分割条件の振り分けで格納 RD エリアが変わることがあります。この場合，データの格納位置が変わっても処理ができるように，-g オプションを指定します。また，LOB 列を含む場合は，-g オプション，及び-j オプションを指定します。

(b)　tblfree=｛未使用領域比率｜（〔未使用領域比率〕，セグメント内空きページ比率）｝

リロードのときに，CREATE TABLE で指定した空き領域比率（PCTFREE の指定値）を変更してデータを格納したい場合に指定します。

スキーマ単位の再編成の場合，指定した値がすべての表に対して適用されます。

＜適用基準＞

表定義時に空き領域比率に 0 以外を指定した表の場合，この表の運用中に空き領域がなくなったとします。

この場合，満杯状態になった RD エリアの表に対して pdrorg を実行したときは，定義時の空き領域比率が適用されて，pdrorg 実行時に RD エリアの容量不足が発生することがあります。その場合に，暫定的に RD エリアを拡張しないで，pdrorg の処理を完了できます。

未使用領域比率

0～99 の値を指定できます。

セグメント内空きページ比率

0～50 の値を指定できます。

(c)　idxfree=未使用領域比率

インデクスを作成する場合に，CREATE INDEX で指定した空き領域比率（PCTFREE の指定値）を変更してインデクスを作成したいときに指定します。指定できる値は，0～99 です。

スキーマ単位の再編成の場合，指定した値がすべての表に対して適用されます。

＜適用基準＞

インデクス定義時に空き領域比率に 0 以外を指定したインデクスの場合，この表の運用中に空き領域がなくなったとします。この場合，満杯状態になった RD エリアのインデクスに対して pdrorg を実行したときは，定義時の空き領域比率が適用されて，pdrorg 実行時に RD エリアの容量不足が発生することがあります。その場合に，暫定的に RD エリアを拡張しないで，pdrorg の処理を完了できます。

(d) job={ジョブ名〔,同期点行数〕〔,CLR〕｜,,CLR}

同期点指定の再編成，又は同期点指定のリロードをする場合に指定します。

同期点指定の再編成，及び**同期点指定のリロード**とは，任意の件数のデータを格納するごとに，トランザクションを決着する再編成及びリロードのことをいいます。ログ取得方式がログ取得モード（-l a）又は更新前ログ取得モード（-l p）の場合，データの格納途中で障害が発生したときでも，データを最初から格納し直す必要がなく，短時間でデータベースを回復できます。この場合の対処方法については，マニュアル「HiRDB システム運用ガイド」の「同期点指定の再編成実行中にユティリティが異常終了したときの対処方法」を参照してください。

ジョブ名　～〈英数字〉((1～3))

同期点指定の再編成，又は同期点指定のリロードを実行する場合のジョブ名を指定します。

ここで指定したジョブ名は，再編成又はリロード実行中にユティリティが異常終了した場合，後から再実行するときの名称となります。また，このジョブ名はほかの pdrorg のジョブ名と重複すると，再実行時に誤動作する原因になります。したがって，ジョブ名は HiRDB システムでユニークになる名称を指定してください。

同期点行数　～((1～1000))《100》

何万件格納するごとにトランザクションを決着するか，データ件数を指定します。例えば，100 を指定すると，100 万件ごとに同期点が取得されます。

ここで小さな値を指定した場合，少ない件数でトランザクションが決着するので，異常終了時のロールバック処理は短くなります。しかし，トランザクションの生成，決着や，プロセスの再起動が頻繁に発生するため，オーバヘッドが大きくなります。一方，大きな値を指定した場合，処理性能の劣化は少なくなりますが，障害発生時の回復時間は長くなります。

ジョブ名を指定しないで同期点行数だけを指定した場合，エラーとなります。

CLR

pdrorg の再実行が必要ない場合，又は現在データベース中にある同期点情報をクリアして再度 pdrorg を実行する場合に指定します。

ユティリティが異常終了した場合，同期点情報がデータベースに残ります。この情報が残ったままだと，次に実行したユティリティが動作できない場合があります。

CLR を指定した場合，ほかのユーザやユティリティが設定した同期点情報をクリアしてしまうため，注意して使用してください。

データベース中に同期点情報が設定されているかどうかは，データベース状態解析ユティリティ（pddbst）の表単位の状態解析で確認できます。

＜適用基準＞

大量データの再編成，リロードをする場合など，格納時間が長くなるときに有効となります。ただし，同期点処理が実行される分，処理性能は悪くなります。また，データページについては，同期点のたびに新たなページから格納を開始するため，データ格納に必要なページ数が通常の再編成よりも多くなります。そのため，RD エリアの空き容量が少ない場合は，適用しない方がよいです。

＜ job オペランドの指定例＞

- 100 万件単位に同期点指定の再編成，又は同期点指定のリロードをする場合

  job=JOB,100　又は　job=JOB
- 現在設定されている同期点情報をクリアして，新たに 100 万件単位の同期点指定の再編成，又は同期点指定のリロードをする場合

  job=JOB,100,CLR　又は　job=JOB,,CLR
- 現在設定されている同期点情報をクリアして通常の再編成又はリロードをする場合

  job=,,CLR

＜注意事項＞

1. ディクショナリ表に対しては，同期点指定の再編成はできません。
2. 同期点情報は，ジョブ名で管理されます。例えば，同期点指定の再編成，又は同期点指定のリロードを実行していた pdrorg(A)が異常終了した場合，その表に対して同じジョブ名を指定した pdrorg(B)を実行すると，pdrorg(A)が残した同期点情報を引き継いで pdrorg(B)が動作してしまいます。このため，ジョブ名が重複しないように管理する必要があります。
3. 同期点指定の再編成，又は同期点指定のリロードが異常終了した場合，そのときのジョブ名の同期点指定の再編成，又は同期点指定のリロードを再実行するまでは，その表に対して通常の再編成又はリロードも実行できません。
4. 同期点情報は RD エリア単位に管理されます。このため，横分割した表の場合，次の条件では表全体の同期点情報が不一致となります。

   ・一部の RD エリアだけ再初期化された場合

   ・RD エリア単位の同期点指定の再編成，又は同期点指定のリロードでエラーが発生した場合

   各 RD エリアが管理している同期点情報は，ジョブ名が同じ場合でも行数が不一致の場合は，再実行できません。しかし，同期点情報を保持している RD エリアと保持していない RD エリアが混在する場合は，保持されている同期点情報で再実行できます。同期点情報のジョブ名が同じで行数が異なる場合の動作を次に示します。

保持している同期点情報（同期点行数）			表単位の同期点指定の再編成，又は同期点指定のリロードをした場合の入力開始行
RD エリア 1	RD エリア 2	RD エリア 3	
0	0	0	0 件目
0	n	0	n 件目
n	0	m	実行できません。
n	m	p	実行できません。

（凡例）

0：同期点情報を保持していない状態

n，m，p：保持している同期点行数

5. 次の条件をすべて満たす場合，同期点指定の再編成，又は同期点指定のリロードはお勧めできません。

・HiRDB/パラレルサーバ

・表を複数のサーバに横分割している

・-g オプションを指定していない

この場合，サーバごとに別々の制御で処理がされます。このため，障害発生時には，サーバごとにアンロード中やリロード中であったり，一部のサーバの処理が終了していることがあります。この状態では，異常終了後にサーバごとに異なる再実行方法をユーザが指示する必要があり，運用が複雑になります。

6. 同期点指定の再編成，又は同期点指定のリロードが異常終了した場合，再実行時のインデクス作成方法（-i オプション）は変更できません。

7. 同期点指定の再編成，又は同期点指定のリロードの実行中に，異常終了する前にリマウント操作をした場合，再実行するときは 1 巻目からマウントし直して実行してください。

8. 同期点指定の再編成，又は同期点指定のリロードが異常終了してから再実行するまでに，RD エリアの再初期化，及び PURGE TABLE 文を実行すると，設定されていた同期点情報が削除されます。

9. LOB 列構成基表に対してだけ同期点指定の再編成，又は同期点指定のリロードをする場合，異常終了後の再実行時には CLR を指定してください。CLR を指定しない場合，再編成したのに LOB 列の検索性能が劣化したりすることがあります。

10. 同期点指定の再編成，又は同期点指定のリロードで制御できるデータ件数の上限は 4,294,960,000 件です。これ以上のデータ件数の再編成をする場合は，次のどちらかの運用を検討してください。

・RD エリア単位に再編成する。

・-W 指定で pdload 用入力データファイルを複数個に分けて出力し，複数回 pdload で同期点指定のデータロードをする。

(e)　exectime=pdrorg の実行監視時間

～＜符号なし整数＞((0～35791394))

pdrorg の実行時間を監視する場合に，その監視時間を分単位で指定します。0 を指定した場合は，実行時間の監視はしません。指定した監視時間内に pdrorg の処理が終了しなかった場合，pdrorg 制御プロセスを強制終了し，無応答原因の調査用に障害情報を取得します。

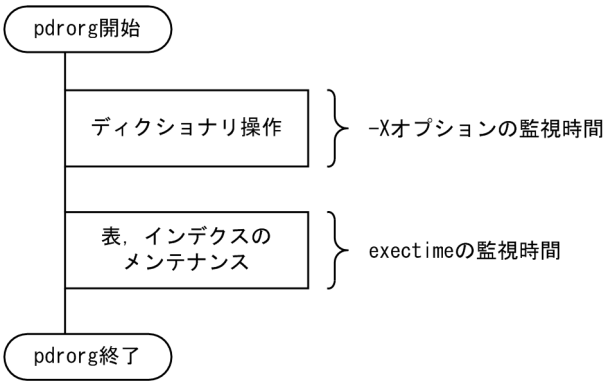
＜ pdrorg の実行時間の監視＞

pdrorg 実行時，次の 2 つの処理に対して実行時間を監視できます。

- 表，インデクスのメンテナンス前準備で，ディクショナリを操作する時間（-X オプション）
- 実際に表，インデクスのメンテナンスをする時間（option 文の exectime オペランド）

-X オプション，及び option 文の exectime で設定する監視時間の関係については，次の図を参照してください。

図 8–23　pdrorg 実行時の監視時間範囲の関係



このオペランドを省略した場合は，次の優先順位でシステム定義の指定値に従います。

1. pd_utl_exec_time
2. pd_cmd_exec_time

＜適用基準＞

夜間バッチなどで監視時間を設定しておけば，pdrorg 実行中に何らかの障害（通信障害（一時的な障害を含む），ディスク障害など）が発生し，pdrorg が無応答状態になった場合は異常終了するため，早期に異変を検知できて，早期に回復処理に着手できます。

＜指定値の目安＞

このオペランドは無応答障害検知を目的とするもので，長大トランザクション実行時間を監視するものではありません。したがって，指定値は，該当する表の処理時間に余裕をもたせた値にする必要があります。例えば，7，8 分程度で終了する pdrorg の実行時間を監視する場合は，exectime=10 と指定しないで，exectime=20 と指定するようにしてください。なお，データ量が単調に増加する表の場合，適時指定値を見直す必要があります。

(f)　tblstatus=clear

表の再編成の再実行時に，強制的に表のステータスモードを通常状態にする場合に指定します。この場合，表の再編成処理は再開始しません。

＜適用基準＞

- データ未完状態となった RD エリア（表）を SQL で参照したい場合など，強制的に表のステータスモードを通常状態に戻すときに指定します。
- 再編成（-k rorg）実行時，インデクスの一括作成中にユティリティが異常終了した場合に，再編成（-k rorg）を再実行しないでインデクス一括作成（-k ixmk）やインデクス再作成（-k ixrc）でインデクス作成だけを行って表とインデクスの状態を回復したとき，表のステータスモードを通常状態に戻すときに指定します。

＜注意事項＞

1. このオペランドは，-k rorg の場合にだけ指定できます。そのほかの場合に指定するとエラーになります。

2. このオペランドを指定して表のステータスを通常状態に戻した後，表の再編成をしても再編成処理は再開始されません。この場合，バックアップから表を回復する必要があります。表を回復しないで表の再編成をすると，アンロードデータファイルの上書き，及びデータ件数不正（データが 0 件）となります。

(g) unldenq= {tblenq ｜ rdenq ｜ nowait}

アンロード時に，他トランザクションに更新を許可するかどうかを指定します（アンロードする表の排他制御モードを変更します）。なお，更新できるのは，アンロード処理でアクセスしていない部分だけです。

tblenq：

他トランザクションに，表に対する参照だけを許可する場合に指定します。この場合，表に共用モード，RD エリアに意図共用モードの排他が掛かります。

rdenq：

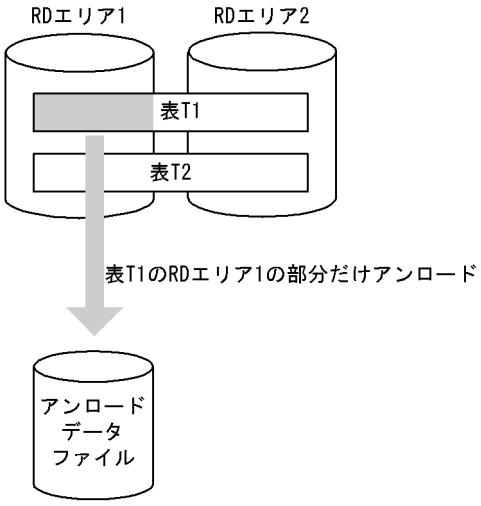
他トランザクションに，RD エリアに対する参照だけを許可する場合に指定します。この場合，表に意図共用モード，RD エリアに共用モードの排他が掛かります。

nowait：

他トランザクションに参照，及びアンロード処理対象外の RD エリアの更新を許可する場合に指定します。この場合，RD エリアにだけ意図共用モードの排他が掛かるため，アンロード中の表が更新できてしまいます。更新処理と同時にアンロードして作成したアンロードデータファイルを使用して，データロードやリロードをすると，データの重複や喪失の原因となるため，アンロード対象表（RD エリア）の更新がないことが保証できる場合にだけ使用してください。

＜適用例＞

次のような表，RD エリアの配置構成で，RD エリア 1 の表 T1 だけ（網掛け部分だけ）アンロードする場合，オペランドの指定によって他トランザクションの更新実行可否が変わります。



〔説明〕

他トランザクションの更新実行可否を次に示します。

unldenq の指定値	アンロード時の順序	排他制御モード				他トランザクションの実行可否							
		RD エリア		表	表（NOWAIT 検索時）	表 T1				表 T2			
		表	インデクス			RD エリア 1		RD エリア 2		RD エリア 1		RD エリア 2	
						参照	更新	参照	更新	参照	更新	参照	更新
tblenq	キー順	SR	SR	PR	−	○	×	○	×	○	○	○	○
	物理順		−										
rdenq	キー順	PR	PR	SR	−	○	×	○	○	○	×	○	○
	物理順		−										
nowait	キー順	SR	SR	−	SR	○	×※	○	○	○	○	○	○
	物理順		−										

（凡例）

SR：意図共用モードの排他が掛かります。

PR：共用モードの排他が掛かります。

−：排他は掛かりません。

○：実行できます。

×：実行できません。

注※

表 T1 の RD エリア 1 のデータを更新しないように，ユーザが制御する必要があります。

(h) charset={utf-16be｜utf-16le}

UTF-8 の列データを，UTF-16 のアンロードデータファイルでアンロードする場合，アンロードデータファイルのエンディアンを指定します。このオペランドは次の条件を満たす場合に指定できます。

- -W オプションを指定している。
- アンロードデータファイルが，拡張 DAT 形式又は固定長データ形式の pdload 用アンロードファイルである。

utf-16be：エンディアンがビッグエンディアンの場合に指定します。

utf-16le：エンディアンがリトルエンディアンの場合に指定します。

このオペランドを省略した場合，アンロードデータファイルの文字コードを UTF-8 にしてアンロードします。

また，このオペランドを指定した場合，UTF-8 の列データを UTF-16 としてアンロードするため，pdrorg が文字コード変換をします。文字コード変換が発生するデータ型について次に示します。

項番	データ型		文字コード変換と charset オペランド指定の関係		
			指定なし	指定あり	
				utf-16be	utf-16le
1	CHAR，VARCHAR	文字集合指定なし	×	○	○
2		文字集合 EBCDIK	◎	×	×
3		文字集合 UTF-16	○	×	◎
4	NCHAR，NVARCHAR		×	×	×
5	MCHAR，MVARCHAR		×	○	○
6	BINARY 型		×	×	×
7	BLOB		×	×	×
8	抽象データ型	CHAR，VARCHAR	×	○	○
9		NCHAR，NVARCHAR	×	×	×
10		MCHAR，MVARCHAR	×	○	○
11		BINARY 型	×	×	×
12		BLOB	×	×	×
13		上記以外	×	×	×
14	上記以外		×	×	×

（凡例）

○：文字コードを変換します（変換後にデータ長が変わります）。

◎：文字コードを変換します（変換後もデータ長は変わりません）。

×：文字コードを変換しません。

変換前と変換後でデータ長が変わる場合，次のように pdrorg がデータ長を調整します。

変換前と変換後でデータ長が変わらない場合

- 拡張 DAT 形式の pdload 用アンロードファイル（固定長文字属性列）のとき
  データを左詰めにし，列定義長まで空白文字を埋めてアンロードします。
- 拡張 DAT 形式の pdload 用アンロードファイル（可変長文字属性列）のとき
  変換後のデータをそのままアンロードします。
- 固定長データ形式の pdload 用アンロードファイルのとき
  データを左詰めにし，列定義長までパディング文字を埋めてアンロードします。

変換前と変換後でデータ長が変わる場合

- 拡張 DAT 形式の pdload 用アンロードファイル（固定長文字属性列）のとき
  データを左詰めにし，列定義長の 2 倍まで空白文字を埋めてアンロードします。

- 拡張 DAT 形式の pdload 用アンロードファイル（可変長文字属性列）のとき
  変換後のデータをそのままアンロードします。
- 固定長データ形式の pdload 用アンロードファイルのとき
  データを左詰めにし，列定義長の 2 倍までパディング文字を埋めてアンロードします。

8.10.16　blobtovarchar 文（データ変換をする列の記述）

BLOB 属性の抽象データ型の列を VARCHAR 型に変換してアンロードする場合，その BLOB 属性の抽象データ型の列名を指定します。

＜適用基準＞

BLOB 属性の抽象データ型の列を VARCHAR 型でアンロードしたい場合に指定します。ただし，コンストラクタパラメタ逆生成関数の戻り値の型が BLOB 型の抽象データ型の場合に指定できます。

＜規則＞

1. blobtovarchar 文は一つだけ指定できます。
2. blobtovarchar 文は，-W bin 指定のアンロードの場合に指定できます。
3. 32 キロバイト（VARCHAR 型で定義できる最大長）以上のデータがある場合は，データ変換エラーとなります。

(1)　形式

```
blobtovarchar 列名〔, 列名〕…
```

(2)　説明

(a)　列名

コンストラクタパラメタ逆生成関数を持つ抽象データ型の列名を指定します。

列名を引用符で囲んだ場合は，英大文字と英小文字が区別されます。囲まない場合は，すべて英大文字として扱われます。

指定できる列名は，コンストラクタパラメタ逆生成関数の戻り値が BLOB 型の抽象データ型の列です。それ以外の列を指定するとエラーになります。

8.10.17　fixtext_option 文（固定長データ形式の入力データファイル生成時の出力データに関する記述）

-W fixtext オプションを指定して，固定長データ形式の入力データファイルを生成する場合，出力データについての情報を指定します。

＜適用基準＞

固定長データ形式の入力データファイルを生成する場合（-W fixtext オプションを指定する場合），出力データを編集するときに指定します。

＜規則＞

1. fixtext_option 文は一つだけ指定できます。
2. オペランドを二つ以上指定する場合，オペランド間を空白文字（0x20）で区切ってください。
3. オペランドの指定がない場合，fixtext_option 文は指定しないでください。

(1)　形式

```
fixtext_option 〔cntlcode={nocheck|check|replace〔,置き換え文字〕}〕
               〔enclose=囲み文字〕
               〔format=データ型，出力形式〕
```

(2)　説明

(a)　cntlcode={nocheck｜check｜replace〔,置き換え文字〕}

データ型が文字データ型，各国文字データ型，及び混在文字データ型の場合，文字データ中に制御文字（ASCII コード 0x00～0x1f）を含む列があるときの，データ出力の制御方法を指定します。

nocheck：

制御文字があるかどうかをチェックしないで，DB 中の列データをそのまま出力します。DB 中の列データをそのまま出力する場合に指定します。

check：

制御文字があるかどうかをチェックします。列データ中に制御文字が含まれている場合，該当行はアンロードデータファイルに出力しません。制御文字を含む行の出力を抑止したい場合に指定します。

replace[，置き換え文字]：

制御文字があるかどうかをチェックします。列データ中に制御文字が含まれている場合，制御文字を置き換え文字に変換し，該当行をアンロードデータファイルに出力します。列データ中に制御文字があっても，テキストビューアなどで参照できる形式で出力する場合に指定します。

置き換え文字は，1 バイトの文字で指定してください。

＜置き換え文字の指定規則＞

1. 置き換え文字を省略した場合，コロンが仮定されます。
2. 次の文字は，置き換え文字に指定できません。
   空白，タブ，アスタリスク，下線
3. 次の文字は，出力データのコードと重なる可能性があるため，置き換え文字には適していません。
   ・出力データ中にある文字
   ・数値データの符号（＋，−）
   ・マルチバイト文字に文字コードが重なる文字（｜，￥，[,]，(,)，{,}，￣）
   ・日間隔，時間隔，及び時刻印データのハイフン
   ・時間，及び時刻印データのコロン
   ・数値，日間隔，時間隔，及び時刻印データのピリオド

＜ cntlcode オペランドの規則＞

1. cntlcode オペランド，又は fixtext_option 文を省略した場合，cntlcode=nocheck が仮定されます。
2. cntlcode オペランドは，fixtext_option 文中に一つだけ指定できます。
3. 列データ中の制御文字有無，及び cntlcode オペランドの指定値と，pdrorg の動作の関係を次に示します。

条件		pdrorg の動作	
列データ中の制御文字有無	cntlcode オペランドの指定値	制御文字のチェック	データ出力方法
あり	check	○※	該当する行を出力しないで処理を続行します（pdrorg はリターンコード 4 で正常終了します）。
	replace		制御文字を置き換えて出力します。
	nocheck	×	DB 中のデータをそのまま出力します。
なし	check	○※	DB 中のデータをそのまま出力します。
	replace		
	nocheck	×	

（凡例）

○：制御文字をチェックします。

×：制御文字をチェックしません。

注※

1 バイト文字をチェックします。マルチバイト文字はチェックしません。

(b)　enclose=囲み文字

列データの前後に囲み文字を付ける場合，データを区別するために 1 バイトの囲み文字を指定します。囲み文字の規則については，cntlcode オペランドの置き換え文字と同じです。

＜適用基準＞

列データの前後に囲み文字を付けたい場合に指定します。また，可変長文字列（文字データ型，各国文字データ型，及び混在文字データ型）の列データ長が定義長より短い場合，又はナル値の場合，定義長までパディング文字（デフォルトでは空白文字）を埋めます。そのため，列データ長が 0 バイトの場合は，0 バイト文字データ，ナル値データ，及び空白文字だけのデータの区別ができません。このような場合に enclose オペランドを指定すると，データに囲み文字が付くため，これらを区別できるようになります。

データ型が varchar(8)，パディング文字が空白文字の場合の，データ出力例を次に示します。

データ	出力されるデータ	
	enclose オペランドなし	enclose="
0 バイト文字	△△△△△△△△	""△△△△△△△△
ナル値	△△△△△△△△	△△△△△△△△△△
すべて空白文字	△△△△△△△△	"△△△△△△△△"

（凡例）

△：空白文字

＜ enclose オペランドの規則＞

1. enclose オペランドは，fixtext_option 文中に一つだけ指定できます。
2. enclose オペランドを指定した場合の囲み文字の有無を次に示します。

列のデータ型	列のデータ値	囲み文字の有無
CHAR，VARCHAR，NCHAR，NVARCHAR，MCHAR，及び MVARCHR	ナル値	囲み文字は付きません（囲み文字ではなく，パディング文字が付きます）。※
	実値	囲み文字が付きます。※
上記以外	ナル値	囲み文字は付きません。
	実値	

注※

出力長が 2 バイト増加します。

(c)　format=データ型，出力形式

列データの出力形式を変更する場合に指定します。変更できるデータ型は INTEGER 及び SMALLINT です。

データ型：

出力形式を変更する列又はパラメタ（コンストラクタパラメタ逆生成関数）のデータ型を指定します。integer 又は smallint を指定してください。

＜規則＞

1. データ型の指定値には，同じデータ型を複数指定できません。

2. 指定したデータ型の列が表にない場合，指定が無視されます。

出力形式：

データ型の出力形式を指定します。type1 又は type2 を指定してください。

省略した場合，type1 が仮定されます。なお，INTEGER 及び SMALLINT 以外のデータ型の場合，type1 となります。

type1，及び type2 については，「出力形式」を参照してください。

8.11　処理性能情報ファイル

%PDDIR%¥spool¥utlrpt ディレクトリ下に内部的な情報を出力します。

8.12　UOC を利用した再編成

UOC を利用して再編成すると，データベースから検索したデータを UOC に渡して，編集してアンロードデータファイルへ出力できます。

8.12.1　再編成時の UOC の利用方法

UOC を利用した再編成は，主に次の場合に適用してください。

- 不要データの削除

  表の再編成とデータ削除を同時にしたい場合，不要なデータを削除したアンロードデータファイルを使用して別表へリロードしたい場合などに，利用します。
- アプリケーション向けデータ更新

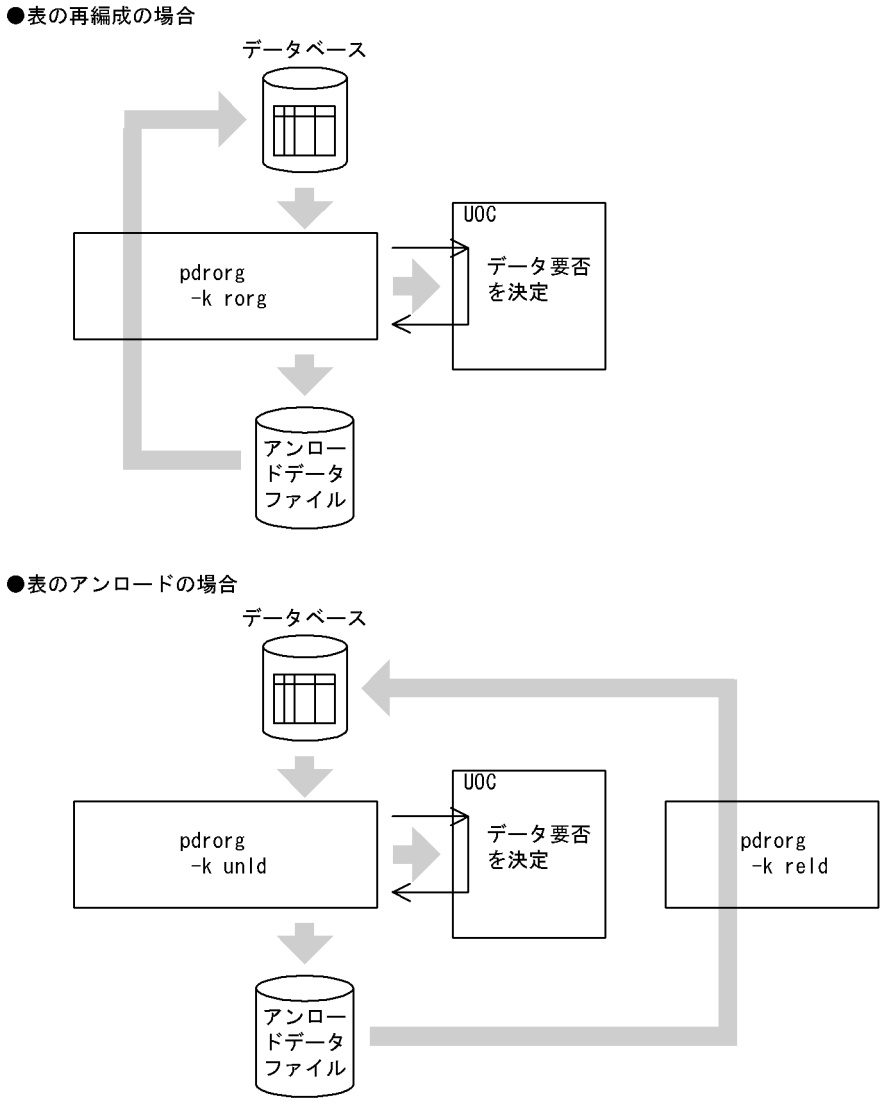
  データを更新して pdload の入力データファイルにしたい場合，アプリケーション用に任意のフォーマットで出力したい場合などに，利用します。

(1)　不要データの削除

表の再編成でのアンロード時，又は表のアンロード時に，UOC でデータの要否を判定して pdrorg に指示することで，必要なデータだけをアンロードデータファイルに出力できます。

UOC を利用した再編成の概要（不要データの削除）を次の図に示します。

図 8–24　UOC を利用した再編成の概要（不要データの削除）



(2)　アプリケーション向けデータ更新

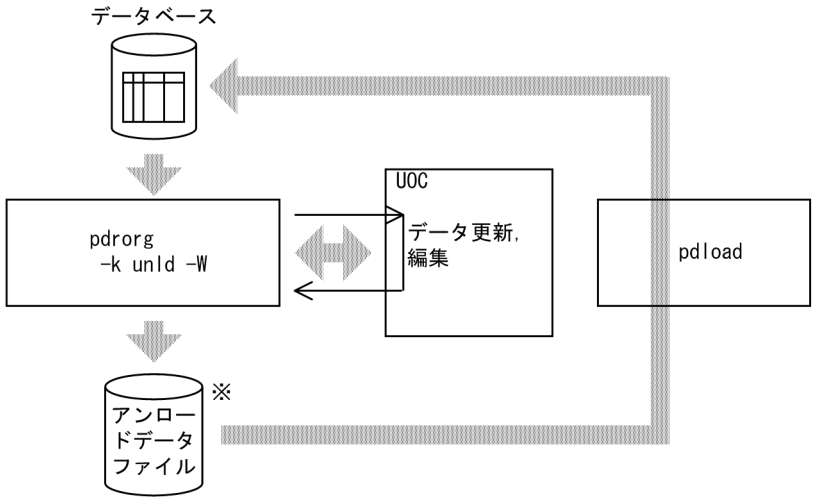
表のアンロード（-k unld 指定）時に，UOC でアンロードするデータを更新又は編集して，次のフォーマットで出力できます。

- pdload の入力データファイル（-W 指定時）
- UOC データファイル（ユーザ任意のフォーマット）

UOC を利用した再編成の概要（アプリケーション向けデータ更新）を次の図に示します。

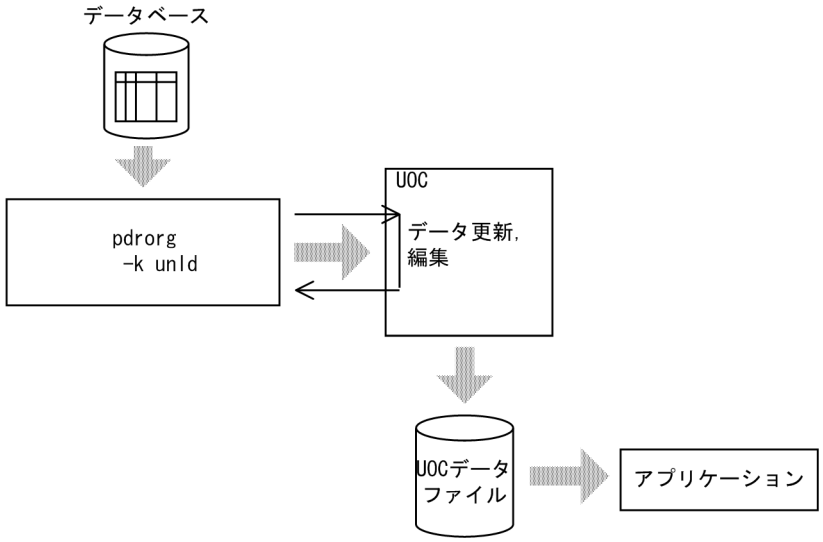
図 8–25　UOC を利用した再編成の概要（アプリケーション向けデータ更新）

●データを更新してアンロードデータファイルに出力する場合



注※　pdloadの入力データファイルのフォーマットとなります。

●アプリケーション用に任意のフォーマットでUOCデータファイルに出力する場合



- **pdrorg での更新後データのチェック**

  pdrorg は，UOC から返された UOC インタフェース領域の更新後データアドレスリストに設定されたデータ値のチェックをします。チェックの結果，不正なデータがあった場合には，pdrorg は処理を中断してリターンコード 8 で終了します。

  更新後データの各データ型のチェック項目を次の表に示します。

表 8–28　更新後データの各データ型のチェック項目

データ型	チェック項目				
	可変長の長さ※1	符号※2	データ範囲※3	繰返し列	
				要素数	ナル値
INTEGER，SMALLINT，FLOAT，SMALLFLT	−	×	×	○	○

データ型	チェック項目				
	可変長の長さ※1	符号※2	データ範囲※3	繰返し列	
				要素数	ナル値
DECIMAL	−	○	○	○	○
CHAR，MCHAR，NCHAR	−	−	×	○	○
DATE，TIME	−	−	○※4	○	○
INTERVAL YEAR TO DAY，INTERVAL HOUR TO SECOND	−	○	○※4	○	○
VARCHAR，MVARCHAR，NVARCHAR	○	−	×	○	○
TIMESTAMP	−	−	○※4	○	○
BINARY	○	−	×	−	−
BLOB（BLOB 属性）	○	−	×	−	−
抽象データ型（パラメタ値）	各パラメタのデータ型に従います。				

（凡例）

○：チェックします。

×：チェックしません。

−：該当しません。

注※1

可変長文字列型，BINARY 型，又は長大データ型の長さ部の内容をチェックします。長さ部の値がマイナスの場合はエラーとなります。

可変長文字列の場合は，長さ部の値が 32,000 バイトを超えるときはエラーとなります。

列のデータ型が長大データ型の場合，又は unld_func 文で指定したコンストラクタパラメタ逆生成関数の戻り値が長大データ型の場合，unlduoc 文で bloblimit を指定したときは，指定したメモリ領域確保サイズを超えるとエラーとなります。unlduoc 文で bloblimit を指定していない場合は，2 ギガバイトを超えるときはエラーとなります。

注※2

パック形式の符号部（4 ビット）をチェックします。符号部が 0xC，0xD，及び 0xF 以外の場合は，エラーとなります。

注※3

パック形式の場合，パックデータ（4 ビット）が 0x0～0x9 以外のときはエラーとなります。そのほかの形式の場合，データ値の内容のチェックはしません。

注※4

日付データ，時刻データ，又は時刻印データとしてあり得ない値（日付データの月が 12 を超えるなど）の場合は，エラーとなります。

8.12.2　オプションと制御文の関係

(1)　UOC を利用した再編成で指定するオプション及び制御文

UOC を利用した再編成で指定するオプション及び制御文を次の表に示します。

表 8–29　UOC を利用した再編成で指定するオプション及び制御文

種別			オプション		制御文		ファイルへの出力	
			-k	-W	unload 文	unlduoc 文	pdrorg	UOC
不要データの削除※1	表の再編成		rorg	×	○※3	○	出力します。	任意です。
	表のアンロード＋リロード	アンロード	unld	−	○※3	○	出力します。	任意です。
		リロード	reld	−	○※3	−	−	−
アプリケーション向けデータ更新※2	pdrorg がファイルを出力		unld	○	○※3	○	出力します。	任意です。
	UOC がファイルを出力		unld	×	○※4	○	該当しません。	必要です。

（凡例）

○：指定してください。

×：指定できません（エラーとなります）。

−：指定しても無視されます。

注※1

表に抽象データ型の列がある場合，unld_func 文にコンストラクタパラメタ逆生成関数を指定した抽象データ型の列だけ，UOC にデータが渡ります。unld_func 文の指定がない抽象データ型の列については，ナル値のデータが渡ります。この場合，UOC にはナル値を渡しますが，アンロードデータファイルには検索したデータを格納します。したがって，UOC でのデータ要否判定はできません。

注※2

表に抽象データ型の列がある場合，すべての抽象データ型の列に対して unld_func 文を指定してください。指定しないとエラーになります。

注※3

unload 文にアンロードデータファイル名を指定します。

注※4

unload 文に(uoc)を指定します。

(2)　UOC を利用した再編成でのオプション指定規則

UOC を利用した再編成でのオプション指定規則を次に示します。

1. UOC を利用した再編成は，表の再編成（-k rorg），又は表のアンロード（-k unld）のときに実行できます。
2. 表に長大データ型の列，又は長大データ属性を持つ抽象データ型の列がある場合，-j オプションを指定する必要があります。-j オプションを指定しないとエラーになります。
3. -c オプションに，dic（処理対象がディクショナリ表）は指定できません。指定するとエラーになります。
4. unload 文に(uoc)を指定した場合，-W オプションは指定できません。指定するとエラーになります。
5. unload 文に(uoc)を指定した場合，-f オプションを指定するとエラーになります。

(3)　UOC を利用した再編成での制御文指定規則

UOC を利用した再編成での制御文指定規則を次に示します。

1. UOC を利用した再編成では，次の二つの制御文を必ず指定する必要があります。
   - unlduoc 文
   - unload 文
2. lobunld 文は指定できません。

(4)　UOC を呼び出すサーバとホストの関係

UOC を呼び出すサーバとホストの関係を次の表に示します。UOC を格納しているダイナミックリンクライブラリを，次の表の該当するホストに用意してください。

表 8–30　UOC を呼び出すサーバとホストの関係

<table>
<tr><th colspan="4">実行環境</th><th>サーバ</th><th>ホスト</th></tr>
<tr><td rowspan="2">HiRDB/シングルサーバ</td><td colspan="3">-g なし</td><td>シングルサーバ</td><td>シングルサーバがあるホスト</td></tr>
<tr><td colspan="3">-g あり</td><td>ユティリティサーバ（pdrorgm）</td><td>unload 文で指定したホスト</td></tr>
<tr><td rowspan="4">HiRDB/パラレルサーバ</td><td rowspan="3">-g なし</td><td rowspan="2">表単位</td><td>非分割表</td><td rowspan="3">バックエンドサーバ</td><td>表が格納されているサーバがあるホスト</td></tr>
<tr><td>分割表</td><td>unload 文で指定したサーバがあるすべてのホスト</td></tr>
<tr><td colspan="2">RD エリア単位</td><td>指定した RD エリアを持つサーバがあるホスト</td></tr>
<tr><td colspan="3">-g あり</td><td>ユティリティサーバ（pdrorgm）</td><td>unload 文で指定したホスト</td></tr>
</table>

(5)　UOC に渡すデータと pdrorg のデータ変換契機

pdrorg には，オプション及び制御文の指定でデータ変換をする機能があります。UOC を利用した再編成でも，この機能を使用できます。なお，データ変換は，UOC に変換しないデータを渡して，アンロードデータファイル出力前にデータの変換をします。UOC に渡すデータと pdrorg のデータ変換契機を次の表に示します。

表 8–31　UOC に渡すデータと pdrorg のデータ変換契機

種別	データ変換ができるオプション又は制御文	UOC に渡すデータ	データ変換契機
不要データの削除（表の再編成）	-S	空白を取り除いていないデータ	アンロードデータファイル出力時
	-W	−	−
	-W dat,sup 又は -W extdat,sup	−	−
	option 文の spacelvl	空白変換していないデータ	表のリロード時
	blobtovarchar 文	−	−
不要データの削除（表のアンロード＋リロード）	-S	空白を取り除いていないデータ	アンロードデータファイル出力時
	-W	−	−
	-W dat,sup 又は -W extdat,sup	−	−
	option 文の spacelvl	空白変換していないデータ	表のリロード時
	blobtovarchar 文	−	−
アプリケーション向けデータ更新（pdrorg がファイルを出力）	-S	−	−
	-W	DAT 形式又はバイナリ形式に変換する前のデータ	アンロードデータファイル出力時※
	-W dat,sup 又は -W extdat,sup	空白を取り除いていないデータ	アンロードデータファイル出力時※
	option 文の spacelvl	空白変換していないデータ	アンロードデータファイル出力時※
	blobtovarchar 文	BLOB 形式のデータ	アンロードデータファイル出力時※
アプリケーション向けデータ更新（UOC がファイルを出力）	-S	空白を取り除いていないデータ	−
	-W	−	−
	-W dat,sup 又は	−	−

種別	データ変換ができるオプション又は制御文	UOC に渡すデータ	データ変換契機
	-W extdat,sup		
	option 文の spacelvl	空白変換していないデータ	−
	blobtovarchar 文	−	−
文字コード変換	option 文の charset	指定した文字コードに変換したデータ	アンロードデータ取得時（UOC コール前）

（凡例）−：該当しません。

注

表定義時に SUPPRESS 指定の DECIMAL 型の列がある場合，UOC には拡張後のデータを渡します。また，データ変換契機はリロード時となります。

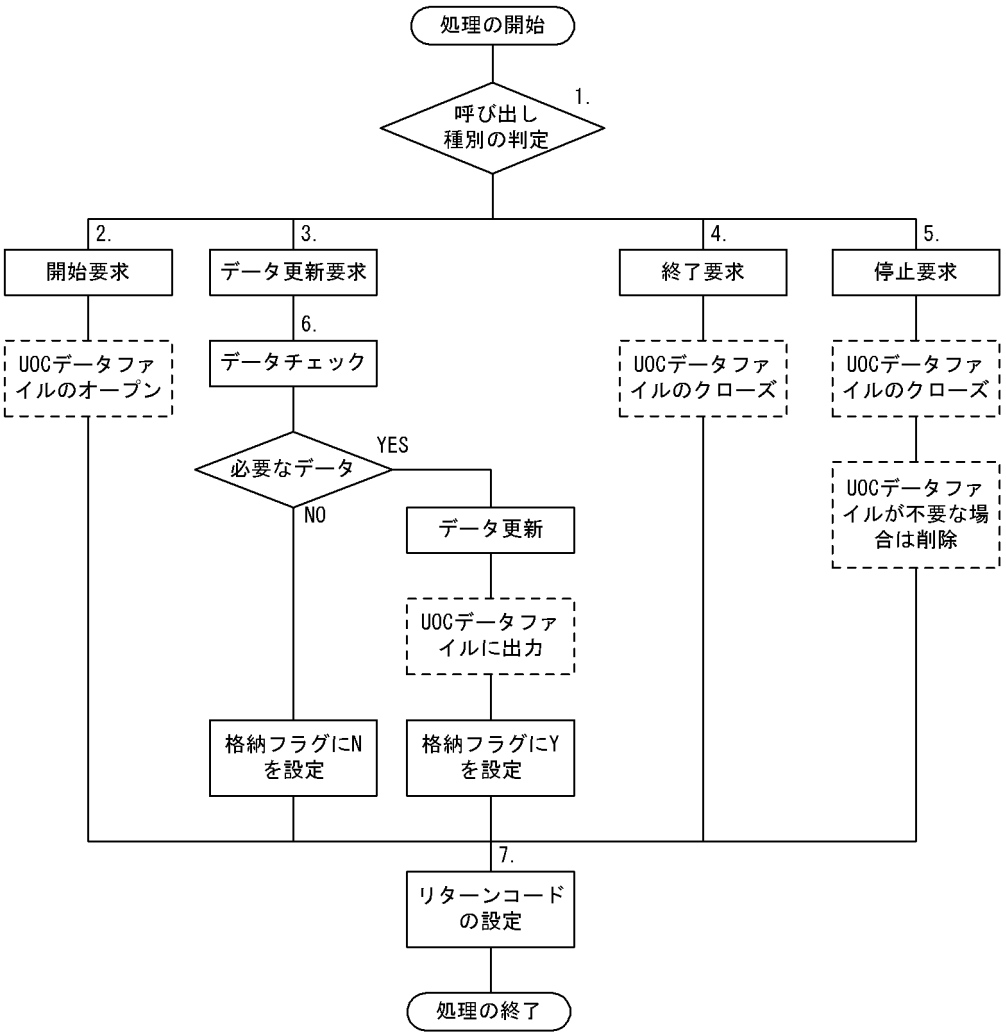
注※

UOC の編集後のデータに対して，データ変換をします。

8.12.3　UOC の処理概要

UOC の処理概要を次の図に示します。

図 8–26　UOC の処理概要



注　破線で囲まれた処理は，UOCでUOCデータファイルを作成する場合に実行してください。

［説明］

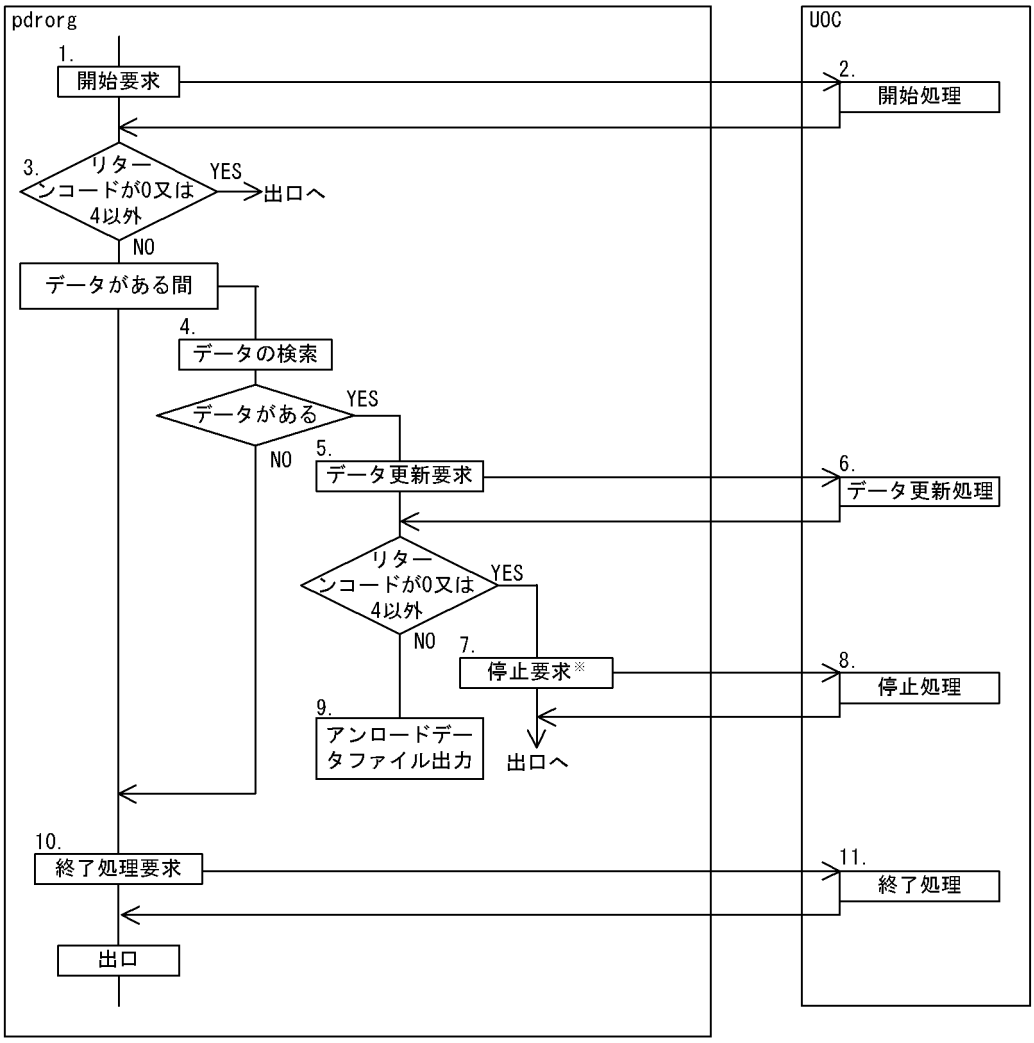
pdrorg から UOC に制御が渡り，処理を開始します。処理が終了したら，pdrorg に制御を返します。

1. UOC インタフェース領域の，pdrorg が設定した呼び出し種別を判定します。
2. 初期化処理，UOC データファイルのオープンなど，開始処理をします。
3. UOC インタフェース領域の，pdrorg が設定したデータを更新又は編集します。
4. データがすべてなくなった場合に，UOC データファイルのクローズなどの終了処理をします。
5. pdrorg の処理でエラーとなった場合，又は UOC 内でエラーとなった場合に，UOC データファイルのクローズ，不要な UOC データファイルの削除など，終了処理をします。
6. 行データをチェックして，必要なデータかどうかを判定します。必要な場合は，データの更新又は編集をして，UOC インタフェース領域の格納フラグに Y を設定します。必要でない場合は，UOC インタフェース領域の格納フラグに N を設定します。また，UOC で UOC データファイルを作成する場合は，データを更新又は編集した行データを UOC データファイルへ出力します。

7. UOC インタフェース領域のリターンコードを設定します。

また，pdrorg，UOC 間の呼び出しシーケンスを次の図に示します。

図 8–27　pdrorg，UOC 間の呼び出しシーケンス



注※　3～9のpdrorgの処理でエラーとなった場合，エラーとなった時点で7の停止処理要求をします。UOCからpdrorgに制御が戻ると，pdrorgは処理を中断して終了します。

[説明]

スキーマ単位の再編成，又はスキーマ単位のアンロードの場合は，スキーマ内の表数分 1～11 を繰り返します。なお，該当する表の処理中にエラーが発生した場合は，直ちに pdrorg の処理を中断します。

1. pdrorg は，呼び出し種別に開始要求を設定して UOC を呼び出します。
2. pdrorg からの呼び出し種別が開始要求の場合，UOC は次の処理をします。
   - UOC で UOC データファイルを出力する場合は，UOC データファイルを作成（オープン）します。
   - 各種領域の確保など，開始準備をします。
   - 開始準備が正常に終了した場合は UOC インタフェース領域のリターンコードに 0 を設定し，正常に終了しなかった場合は 8 を設定してリターンします。

リターンコードに 8 を設定した場合，この呼び出し以降は UOC を呼び出せない（停止要求をしない）ため，ここで UOC データファイルのクローズなどの後処理をする必要があります。

3. pdrorg は，UOC から返却された UOC インタフェース領域のリターンコードを判定し，0 又は 4 以外の場合は直ちに pdrorg の処理を中断します。
4. pdrorg は，データベースからデータを検索します。
5. pdrorg は，表のデータが検索できた場合は，データアドレスリストに検索したデータを設定し，呼び出し種別にデータ更新要求を設定して UOC を呼び出します。検索するデータがなくなった場合は 10 の処理（終了処理）をします。

   なお，データ更新要求は，次の条件のどれかに該当するまで処理を繰り返します。

   - 表のデータがなくなる。
   - UOC が UOC インタフェース領域のリターンコードに 0 又は 4 以外を設定する。
   - pdrorg 内で処理エラーが発生する。
6. pdrorg からの呼び出し種別がデータ更新要求の場合，UOC は次の処理をします。

   各処理が正常終了した場合は UOC インタフェース領域のリターンコードに 0 を設定し，異常終了した場合は 8 を設定してリターンします。

   (a) 表の再編成，表のアンロードでの不要データの削除の場合

   pdrorg が UOC インタフェース領域のデータアドレスリストに渡したデータを参照し，データが必要であれば UOC インタフェース領域の格納フラグに Y を設定します。データが不要な場合は N を設定します。

   (b) アプリケーション向けデータ更新の場合

   データの更新などの処理をします。

   データを更新した場合は，UOC インタフェース領域に更新後データアドレスリストを設定し，格納フラグに Y を設定します。

   不要なデータの場合は，UOC インタフェース領域の格納フラグに N を設定します。

   UOC データファイルを出力する場合は，UOC データファイルに必要なデータを出力します。
7. UOC から返却された UOC インタフェース領域のリターンコードが 8 の場合，pdrorg は呼び出し種別に停止要求を設定して UOC を呼び出します。
8. pdrorg からの呼び出し種別が停止要求の場合，UOC は終了処理をします。なお，この呼び出し以降は UOC を呼び出せません。

   - UOC データファイルを作成した場合は，UOC データファイルをクローズしてください。なお，作成した UOC データファイルが不要な場合は，ここで削除してください。
   - 確保したメモリの解放などをしてください。
9. pdrorg がアンロードデータファイルを出力する場合，UOC インタフェース領域の格納フラグが Y のときは，データをアンロードデータファイルに出力します。N のときは出力しません。なお，データ更新（-W オプション指定）では，UOC インタフェース領域の更新後データアドレスリストを参照して，UOC が更新したデータを-W オプション指定のフォーマットでアンロードデータファイルに出力します。

10. 次の条件をすべて満たす場合，pdrorg は呼び出し種別に終了要求を設定して UOC を呼び出します。
    - 該当する表のデータをすべて検索した場合
    - pdrorg 内でエラーが発生しなかった場合
    - UOC から返却された UOC インタフェース領域のリターンコードがすべて正常（0 又は 4）の場合
11. pdrorg からの呼び出し種別が終了要求の場合，UOC は終了処理をします。
    - UOC データファイルを作成した場合は，UOC データファイルをクローズしてください。
    - 確保したメモリの解放などをしてください。

8.12.4　UOC インタフェース

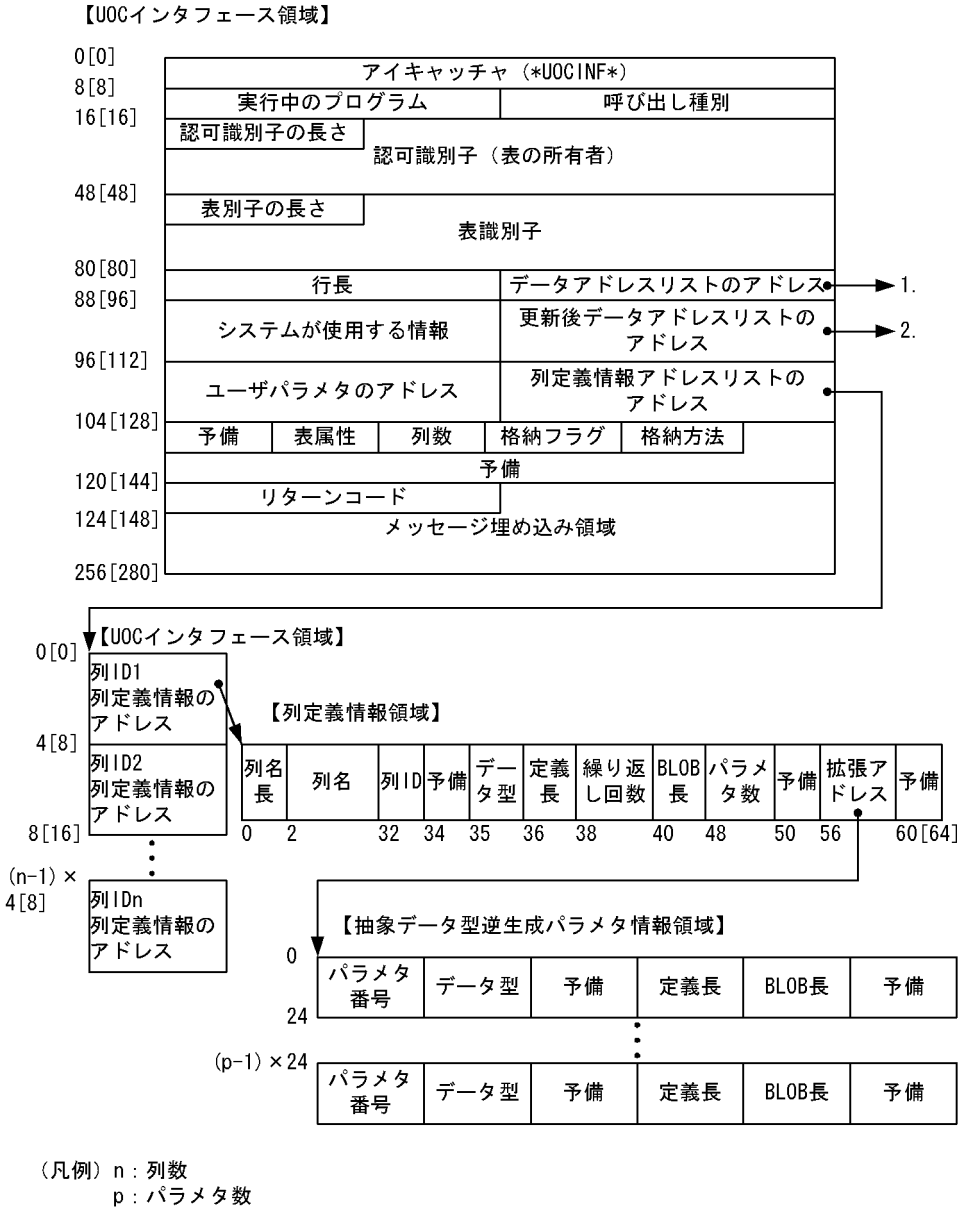
(1)　UOC インタフェース領域の構造と内容

pdrorg と UOC との連絡は，インタフェース領域を使用して情報の入出力をします。これを UOC インタフェース領域といいます。

この領域は，更新バッファを除いて，すべて pdrorg が確保します。UOC は，呼び出す関数の第 1 引数にこの領域のアドレスを受け取って，参照及び更新をします。

UOC インタフェース領域の構造を次の図に示します。

図 8-28　UOC インタフェース領域の構造（1/2）



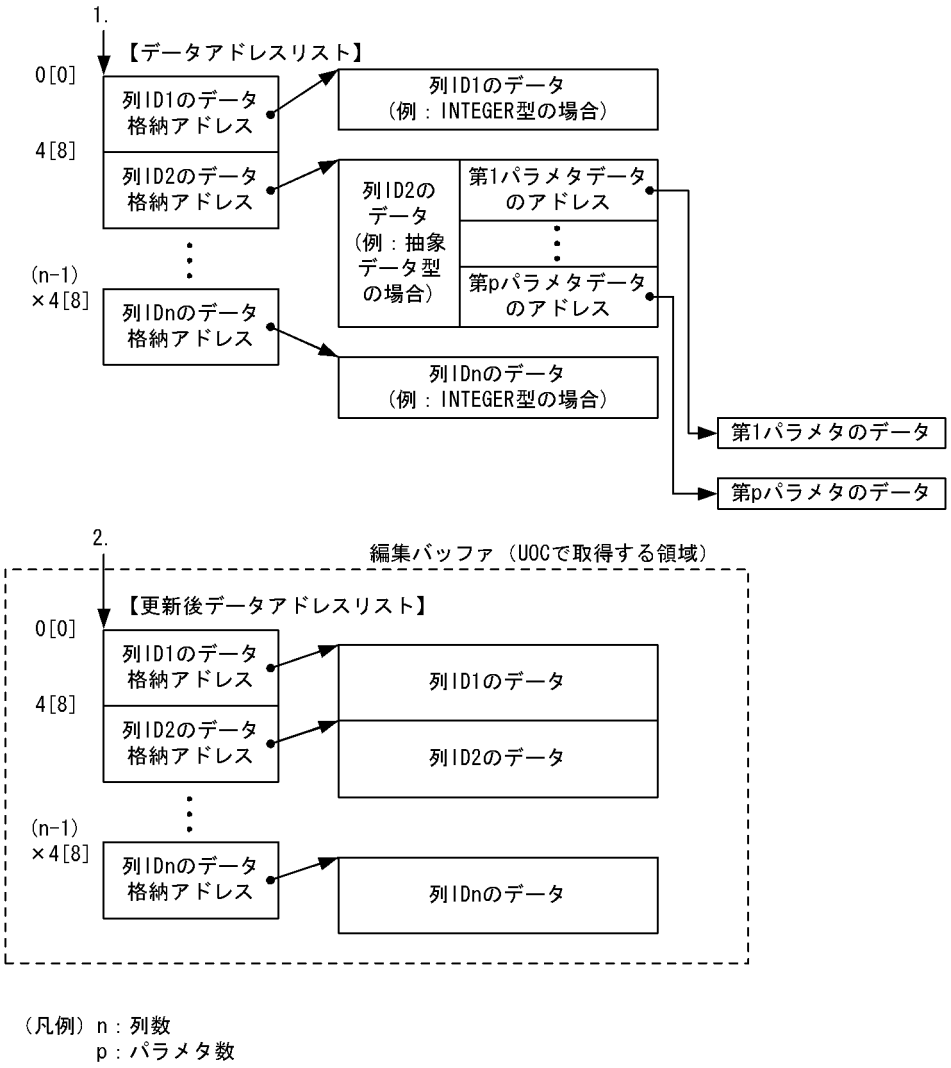
注 1

[ ]内は，64 ビットモードの HiRDB の場合の数値となります。

注 2

UOC インタフェース領域の内容については表「UOC インタフェース領域の内容」を，列定義情報アドレスリストの内容については表「列定義情報アドレスリストの内容」を，列定義情報領域の内容については表「列定義情報領域の内容」を，抽象データ型逆生成パラメタ情報領域の内容については表「抽象データ型逆生成パラメタ情報領域の内容」を参照してください。

図 8–29　UOC インタフェース領域の構造（2/2）



（凡例）n：列数
　　p：パラメタ数

注 1

[ ]内は，64 ビットモードの HiRDB の場合の数値となります。

注 2

データアドレスリストの内容については表「データアドレスリストの内容」を，更新後データアドレスリストの内容については表「更新後データアドレスリストの内容」を参照してください。

表 8–32　UOC インタフェース領域の内容

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
0	0	アイキャッチャ	8	8	char	pdrorg	インタフェース領域('*UOCINF*')
8	8	実行中プログラム	4	4	int	pdrorg	制御が渡っているプログラム 0：pdrorg 1：UOC

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
12	12	呼び出し種別	4	4	int	pdrorg	UOC に対する処理要求種別 o：開始要求 e：データ更新要求 c：終了要求 t：停止要求
16	16	認可識別子の長さ	2	2	short	pdrorg	対象表の所有者名の長さ
18	18	認可識別子	30	30	char	pdrorg	対象表の所有者名
48	48	表識別子の長さ	2	2	short	pdrorg	対象表の名称の長さ
50	50	表識別子	30	30	char	pdrorg	対象表の名称
80	80	行長	4	8	long	pdrorg	データ格納方法が Y の場合，行長を設定します。
84	88	データアドレスリストのアドレス	4	8	void*	pdrorg	pdrorg が検索したデータのアドレスを格納している，データアドレスリストの先頭アドレス
88	96	システムが使用する情報	4	8	long	pdrorg	システムが使用する情報（UOC 側では使用しないでください）
92	104	更新後データアドレスリストのアドレス	4	8	void*	UOC	UOC で更新したデータを pdrorg に返却する場合は，UOC 内で作成したデータのアドレスを格納している更新後データアドレスリストの先頭アドレスを設定します。データを更新しない場合は，0 を設定します。
96	112	ユーザパラメタのアドレス	4	8	void*	pdrorg	unlduoc 文の param オペランドに指定した文字列のアドレスを設定します（文字列の末尾は¥0）。param オペランドを省略した場合は，ナル値を設定します。
100	120	列定義情報アドレスのアドレス	4	8	void*	pdrorg	対象表の列定義情報アドレスリストの先頭アドレス
104	128	予備	1	1	char	pdrorg	予備領域（UOC 側では使用しないでください）
105	129	表の属性	1	1	char	pdrorg	対象表の表属性 F：FIX 表 空白：非 FIX 表
106	130	列数	2	2	short	pdrorg	対象表の列数
108	132	格納フラグ	1	1	char	UOC	格納フラグを設定します。 Y：アンロードデータファイルに格納します。

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
							N：アンロードデータファイルに格納しません。 pdrorg 側でアンロードデータファイルを作成する場合，必ず設定してください（UOC で UOC データファイルを作成する場合は指定しなくてもかまいません）。
109	133	データ格納方法	1	1	char	pdrorg	FIX 表の場合，unlduoc 文の fixrow オペランドに指定した値（fixrow 省略時は N）を設定します。 Y：列定義順にデータを詰めて（連続して）渡します。 N：データ型のバウンダリに従ってデータの開始アドレスを補正して渡します。
110	134	予備	10	10	char	pdrorg	予備領域（UOC 側では使用しないでください）
120	144	リターンコード	4	4	int	UOC	リターンコードを設定します。 pdrorg から制御を受け取った UOC は，次の基準でリターンコードをコールごとに戻してください。 0： 各要求に対する処理が正常に実行された場合に設定します。 4： デバッグなどで，正常時もメッセージを出力したい場合に設定します。内容は 0 と同じです。リターンコードに 4 を設定した場合，メッセージは 3 回まで出力されます。ただし，メッセージが 3 回出力された後に 4 を設定しても無視され，メッセージは出力されません。 8： UOC の処理中にエラーが発生した場合に設定します。 上記以外のリターンコードが設定された場合，pdrorg は処理を中断し終了します。また，リターンコードに 4 又は 8 を設定すると，メッセージ埋め込み領域の内容を標準出力及びメッセージログに出力します。メッセージは，末尾に¥0 を付けて 131 バイト以内の文字列にしてください。先頭データが¥0 の場合はメッセージは出力しません。

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
124	148	メッセージ埋め込み領域	132	132	char	pdrorg UOC	標準出力及びメッセージログに出力するメッセージの格納領域（pdrorg は先頭データを¥0 にして渡します）

（凡例）

32：32 ビットモードの HiRDB の相対位置，及び長さを示します。

64：64 ビットモードの HiRDB の相対位置，及び長さを示します。

注 1

先頭アドレスは，32 ビットモードの HiRDB の場合は 4 バイトのバウンダリ，64 ビットモードの HiRDB の場合は 8 バイトのバウンダリを保証します。

注 2

各フィールドに対する参照，更新可否を次に示します。

フィールド名	呼び出し種別			
	開始要求	データ更新要求	終了要求	停止要求
アイキャッチャ	△	△	△	△
実行中プログラム	△	△	△	△
呼び出し種別	△	△	△	△
認可識別子の長さ	△	△	△	△
認可識別子	△	△	△	△
表識別子の長さ	△	△	△	△
表識別子	△	△	△	△
行長	×	△	×	×
データアドレスリストのアドレス	×	△	×	×
更新後データアドレスリストのアドレス	×	○	×	×
ユーザパラメタのアドレス	△	△	△	△
列定義情報アドレスのアドレス	△	△	△	△
表の属性	△	△	△	△
列数	△	△	△	△
データ格納方法	×	○	×	×
メッセージ埋め込み領域	○	○	○	○
リターンコード	○	○	○	○

フィールド名	呼び出し種別			
	開始要求	データ更新要求	終了要求	停止要求
格納フラグ	×	○	×	×

（凡例）

○：値を設定できます。

△：値を参照できます。

×：値を参照できません（値が保証されません）。

表 8–33　列定義情報アドレスリストの内容

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
0	0	列 ID1 の定義情報アドレス	4	8	void*	pdrorg	列 ID1 の列定義情報のアドレスが設定されます。
4	8	列 ID2 の定義情報アドレス	4	8	void*	pdrorg	列 ID2 の列定義情報のアドレスが設定されます。
(n-1)×4	(n-1)×8	列 IDn の定義情報アドレス	4	8	void*	pdrorg	列 IDn の列定義情報のアドレスが設定されます。

（凡例）

32：32 ビットモードの HiRDB の相対位置，及び長さを示します。

64：64 ビットモードの HiRDB の相対位置，及び長さを示します。

注 1

先頭アドレスは，32 ビットモードの HiRDB の場合は 4 バイトのバウンダリ，64 ビットモードの HiRDB の場合は 8 バイトのバウンダリを保証します。

注 2

列定義情報アドレスリストの内容は，列の定義順に連続して渡します。

表 8–34　列定義情報領域の内容

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
0	0	列名長	2	2	short	pdrorg	列名の長さが設定されます。
2	2	列名	30	30	char	pdrorg	列名が設定されます。
32	32	列 ID	2	2	short	pdrorg	列の ID が設定されます。
34	34	予備 1	1	1	–	–	–

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
35	35	データ型	1	1	unsigned char	pdrorg	列のデータ型が設定されます。各データ型のデータコードとデータ値のバウンダリについては，表「各データ型のデータコードとデータ値のバウンダリ」を参照してください。
36	36	定義長	2	2	short	pdrorg	列の定義長が設定されます。列の定義長領域の長さと内容については，表「列の定義長領域の長さと内容」を参照してください。
38	38	繰り返し回数	2	2	short	pdrorg	列の繰り返し回数が設定されます。
40	40	BLOB 長，又は BINARY 長	8	8	int[2]	pdrorg	列のデータ型が長大データ型，又は BINARY 型の場合に設定されます。上位 4 バイトには 0，下位 4 バイトには長大データ型の定義長がバイト単位で設定されます。
48	48	パラメタ数	2	2	short	pdrorg	抽象データ型の場合，該当する unld_func 文の関数の数（パラメタ数）が設定されます。なお，次の場合は，0 が設定されます。 • 列のデータ型が抽象データ型以外の場合 • unld_func 文を指定していない抽象データ型の場合
50	50	予備 2	6	6	char	pdrorg	予備領域
56	56	拡張アドレス	4	8	void*	pdrorg	抽象データ型逆生成パラメタ情報の先頭アドレスが設定されます。なお，パラメタ数が 0 の場合は，0 が設定されます。
60	64	予備 3	64	64	char	pdrorg	予備領域

（凡例）

32：32 ビットモードの HiRDB の相対位置，及び長さを示します。

64：64 ビットモードの HiRDB の相対位置，及び長さを示します。

−：該当する内容はありません。

注

先頭アドレスは，32 ビットモードの HiRDB の場合は 4 バイトのバウンダリ，64 ビットモードの HiRDB の場合は 8 バイトのバウンダリを保証します。

表 8–35　抽象データ型逆生成パラメタ情報領域の内容

相対位置	フィールド名	長さ（単位：バイト）	属性	値の設定元	内容
0	パラメタ番号	2	short	pdrorg	パラメタ番号（1 が先頭）

相対位置	フィールド名	長さ（単位：バイト）	属性	値の設定元	内容
2	データ型	1	unsigned char	pdrorg	パラメタのデータ型が設定されます。各データ型のデータコードとデータ値のバウンダリについては，表「各データ型のデータコードとデータ値のバウンダリ」を参照してください。
3	予備 1	3	char	pdrorg	予備領域
6	定義長	2	short	pdrorg	パラメタの定義長が設定されます。列の定義長領域の長さと内容については，表「列の定義長領域の長さと内容」を参照してください。
8	BLOB 長，又は BINARY 長	8	int[2]	pdrorg	パラメタのデータ型が長大データ型，又は BINARY 型の場合，BLOB 長が設定されます。上位 4 バイトには 0，下位 4 バイトには長大データ型の定義長がバイト単位で設定されます。
16	予備 2	112	char	pdrorg	予備領域

注 1

先頭アドレスは，32 ビットモードの HiRDB の場合は 4 バイトのバウンダリ，64 ビットモードの HiRDB の場合は 8 バイトのバウンダリを保証します。

注 2

抽象データ型逆生成パラメタ情報領域の内容は，unld_func 文で指定した関数（パラメタ）の順に連続して渡します。

表 8–36　列の定義長領域の長さと内容

データ型	領域長	内容
DECIMAL，LARGE DECIMAL，INTERVAL YEAR TO DAY，及び INTERVAL HOUR TO SECOND	上位 1 バイト	定義で指定した精度です。
	下位 1 バイト	定義で指定した位取りです。
TIMESTAMP(p)	2 バイト	7 ＋↑p/2↑
BLOB，抽象データ型，及び BINARY	2 バイト	値は保証されません。
上記以外	2 バイト	定義長です。

表 8–37　各データ型のデータコードとデータ値のバウンダリ

データ型	データコード(16 進数)		バウンダリ(単位：バイト)
	非ナル値制約指定なし	非ナル値制約指定あり	
MVARCHAR	A1	A0	2
MCHAR	A5	A4	−
NVARCHAR	B1	B0	2

データ型		データコード(16 進数)		バウンダリ(単位：バイト)
		非ナル値制約指定なし	非ナル値制約指定あり	
NCHAR		B5	B4	−
VARCHAR		C1	C0	2
CHAR		C5	C4	−
FLOAT		E1	E0	8
SMALLFLT		E3	E2	4
DECIMAL		E5	E4	−
INTEGER		F1	F0	4
SMALLINT		F5	F4	2
INTERVAL YEAR TO DAY		65	64	−
DATE		71	70	−
TIME		79	78	−
INTERVAL HOUR TO SECOND		6F	6E	−
TIMESTAMP		7D	7C	−
BINARY		91	90	4
BLOB		93	92	4
抽象データ型	抽象データ型パラメタのアドレスリスト	83	82	32 ビットモードの HiRDB の場合は 4，64 ビットモードの HiRDB の場合は 8
	パラメタのデータ	パラメタの各データ型に従います。		

（凡例）−：アドレスのバウンダリを調整しません。

表 8–38　データアドレスリストの内容

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
0	0	列 ID1 のデータ格納アドレス	4	8	void*	pdrorg	列 ID1 のデータのアドレスが設定されます。
4	8	列 ID2 のデータ格納アドレス	4	8	void*	pdrorg	列 ID2 のデータのアドレスが設定されます。
(n-1)×4	(n-1)×8	列 IDn のデータ格納アドレス	4	8	void*	pdrorg	列 IDn のデータのアドレスが設定されます。

（凡例）

32：32 ビットモードの HiRDB の相対位置，及び長さを示します。

64：64 ビットモードの HiRDB の相対位置，及び長さを示します。

注 1

先頭アドレスは，32 ビットモードの HiRDB の場合は 4 バイトのバウンダリ，64 ビットモードの HiRDB の場合は 8 バイトのバウンダリを保証します。

注 2

データアドレスリストの内容は，列の定義順に連続して渡します。

注 3

列値がナル値の場合は，0 が設定されます。

注 4

FIX 表の場合，UOC インタフェース領域のデータ格納方法が Y のときは，データバッファは列 ID1 のデータ格納アドレスが示すアドレスから，UOC 情報の行長分連続した領域で格納します。このとき，列 ID 順に格納します。

注 5

UOC インタフェース領域のデータ格納方法が N の場合，データ型ごとに設定されたバウンダリで，データの開始アドレスを調整して設定します。なお，列ごとにデータは連続していません。

注 6

列定義情報領域のパラメタ数が 0 の場合は，0 が設定されます。

表 8–39　更新後データアドレスリストの内容

相対位置		フィールド名	長さ（単位：バイト）		属性	値の設定元	内容
32	64		32	64			
0	0	列 ID1 のデータ格納アドレス	4	8	void*	UOC	列 ID1 のデータのアドレスを設定します。
4	8	列 ID2 のデータ格納アドレス	4	8	void*	UOC	列 ID2 のデータのアドレスを設定します。
(n-1)×4	(n-1)×8	列 IDn のデータ格納アドレス	4	8	void*	UOC	列 IDn のデータのアドレスを設定します。

（凡例）

32：32 ビットモードの HiRDB の相対位置，及び長さを示します。

64：64 ビットモードの HiRDB の相対位置，及び長さを示します。

注 1

先頭アドレスは，32 ビットモードの HiRDB の場合は 4 バイトのバウンダリ，64 ビットモードの HiRDB の場合は 8 バイトのバウンダリを保証してください。

注 2

データ更新後のデータアドレスリストは，列の定義順に連続した領域で設定してください。

注 3

データのアドレス値として，データアドレスリストのデータ格納アドレスを設定してもかまいません。ただし，データ値は更新しないでください。

注 4

データのアドレスは，バウンダリを保証しなくてもかまいません。

注 5

列定義で非ナル値制約指定なしの列の列値を，ナル値に変更する場合は 0 を設定してください。非ナル値制約指定有無によるデータ変更可否を次に示します。

列の属性	更新前のデータ	更新後のデータ	
		非ナル値	ナル値
非ナル値制約指定なし	ナル値	○	○
	非ナル値	○	○
非ナル値制約指定あり	非ナル値	○	×

（凡例）

○：変更できます。

×：変更できません。

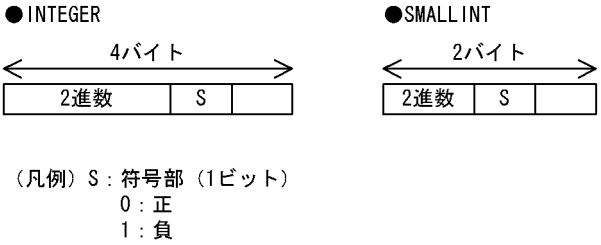
注 6

予備列が定義された FIX 表の場合，予備列の値を更新しても，アンロードファイルには反映されません。予備列にデータを格納してアンロードしたい場合は，UOC を利用してファイルを出力してください。UOC を利用したファイル出力については，「unload 文（アンロードデータファイルの情報の記述）」を参照してください。

(2)　データ値の形式

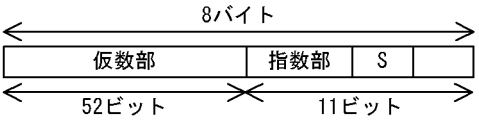
各データ型のデータ値の形式を次に示します。

(a)　INTEGER 又は SMALLINT

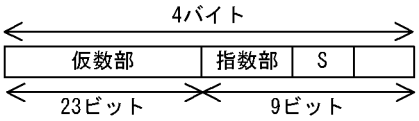


(b)　FLOAT 又は SMALLFLT

●FLOAT

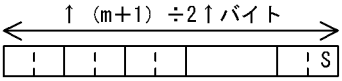


●SMALLFLT（IEEE規格と同一の形式）



（凡例）S：符号部（1ビット）
0：正
1：負

(c)　DECIMAL



（凡例）

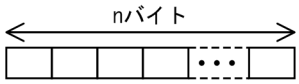


：1バイト
m：精度（全体のけた数）
S：符号部（4ビット）
注　パック形式は，1バイトで2けたの数字を表します。

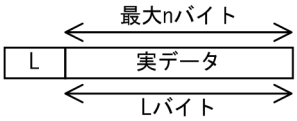
(d)　CHAR，MCHAR，VARCHAR，又は MVARCHAR

文字コード変換なし又は変換後もデータ長が変化しない文字コード変換をした場合の形式を次に示します。

●CHAR(n)又はMCHAR(n)



●VARCHAR(n)又はMVARCHAR(n)

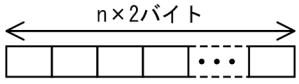


（凡例）n：定義長
L：実データ長（2バイト）

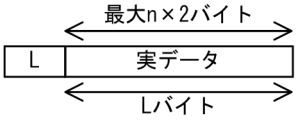
(e)　CHAR 又は VARCHAR

文字コード変換後にデータ長が変化する場合の形式を次に示します。

●CHAR(n)

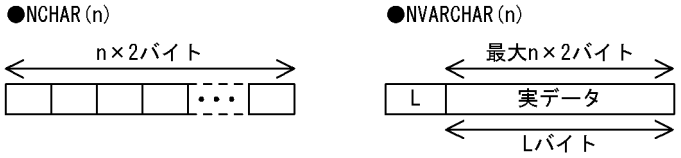


●VARCHAR(n)



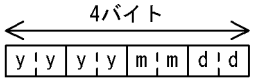
（凡例）n：定義長
L：実データ長（2バイト）

(f)　NCHAR 又は NVARCHAR



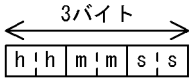
（凡例）n：定義長
　　L：実データ長（2バイト）

(g)　DATE



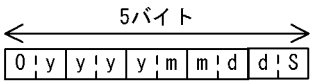
（凡例）
yyyy：年（西暦）
mm：月
dd：日
注１　DATEは，4バイトの符号部のないパック形式10進数となります。
注２　パック形式は，1バイトで2けたの数字を表します。

(h)　TIME



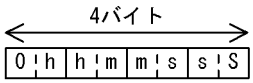
（凡例）
hh：時
mm：分
ss：秒
注１　TIMEは，3バイトの符号部のないパック形式10進数となります。
注２　パック形式は，1バイトで2けたの数字を表します。

(i)　INTERVAL YEAR TO DAY



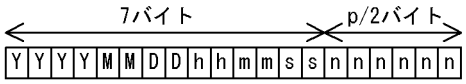
（凡例）
yyyy：年（西暦）
mm：月
dd：日
S：符号部（4ビット）
注１　INTERVAL YEAR TO DAYは，5バイトのパック形式10進数となります。
注２　パック形式は，1バイトで2けたの数字を表します。

(j)　INTERVAL HOUR TO SECOND



（凡例）
hh：時
mm：分
ss：秒
S：符号部（4ビット）
注１　INTERVAL HOUR TO SECONDは，4バイトのパック形式10進数となります。
注２　パック形式は，1バイトで2けたの数字を表します。

(k)　TIMESTAMP

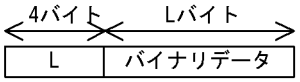


最大10バイトの符号部のない
パック形式10進数

・小数秒は，定義によって0，2，4，又は6となります。
・パック形式は1バイトで2けたの数字を表します。

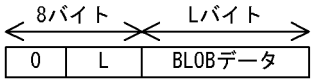
YYYY：年
MM：月
DD：日
hh：時
mm：分
ss：秒
nnnnnn：小数秒

(l)　BINARY



先頭4バイト（L）に2進数でデータの長さを格納します。

(m)　BLOB

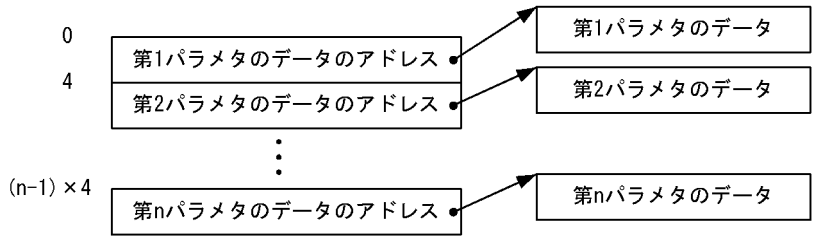


（凡例）
L：データ長（4バイト）
注　先頭4バイトに0（0x00）を格納し，次の4バイト（L）にデータ長を2進数で格納します。

(n)　抽象データ型

抽象データ型の場合は，抽象データ型パラメタのアドレスリストが格納されます。抽象データ型パラメタのアドレスリストの構造を次の図に示します。

図 8–30　抽象データ型パラメタのアドレスリストの構造



（凡例）n：パラメタ数

また，抽象データ型パラメタのアドレスリストの内容を次の表に示します。

表 8–40　抽象データ型パラメタのアドレスリストの内容

相対位置	フィールド名	長さ（単位：バイト）	属性	値の設定元	内容
0	第 1 パラメタのデータのアドレス	4	void*	UOC	第 1 パラメタのデータのアドレスが設定されます。
4	第 2 パラメタのデータのアドレス	4	void*	UOC	第 2 パラメタのデータのアドレスが設定されます。
(n-1)×4	第 n パラメタのデータのアドレス	4	void*	UOC	第 n パラメタのデータのアドレスが設定されます。

注 1

先頭アドレスは，4 バイトのバウンダリを保証します。

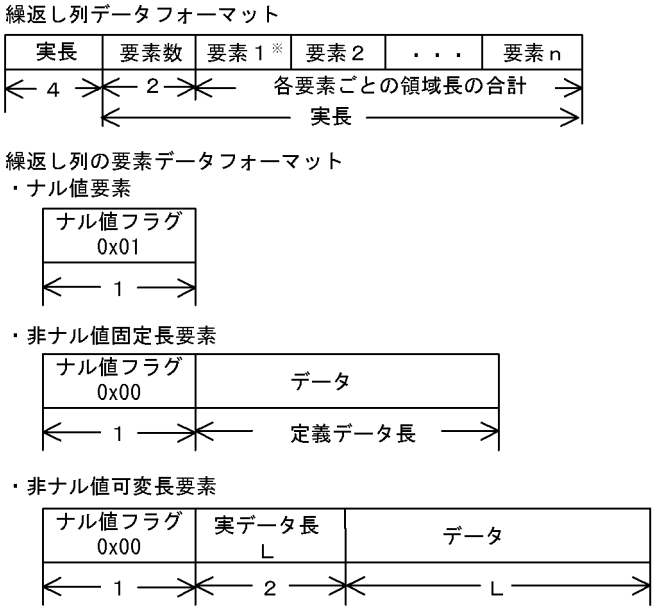
注 2

パラメタのデータの開始アドレスは，データ型ごとに設定したバウンダリで調整したアドレスが設定されます。

注 3

抽象データ型パラメタのアドレスリストの内容は，unld_func 文で指定した関数（パラメタ）の順に連続して渡します。

(o)　繰返し列の場合



注　長さの単位はバイトです。
注※　要素データの開始アドレスは，バウンダリ調整して渡しません。

8.12.5　UOC の使用例

ここでは，UOC を使用した表の再編成の例について説明します。

なお，使用例中の UOC コーディング例は，サンプルデータベースとして提供されています。格納ディレクトリは，%PDDIR%¥sample¥sampleUOC です。

(1)　UOC を使用した表の再編成の例

表の再編成をするときに，データベースに登録された日時が 2001 年以前のデータを削除します。

(a)　データベースの表定義

```
CREATE TABLE 会員名簿（会員番号 INTEGER,
                    会員名 NCHAR(20),
                    会員住所 NVARCHAR(100),
                    入会日時 DATE NOT NULL WITH DEFAULT);
```

(b)　コマンド形式

```
pdrorg -k rorg -t 会員名簿 制御情報ファイル
```

(c)　制御情報ファイルの内容

```
unload アンロードデータファイル名 uoc_lib=ライブラリ名
unlduoc entry=old_data_delete
```

(d)　UOC コーディング例

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <errno.h>
#include <pdutluoc.h>   .....................................................1

void old_data_delete (
    UTL_UOC_INF         *uocinf             /* A(UOC interface area)      */   ...2
) {
    UTL_UOC_DATA_BUF    *addr_list;         /* A(data address list)       */
    unsigned char       *date_adr;          /* A(DATE)                    */
    static unsigned char cyear[2];          /* compare year               */

    switch(uocinf->req_cd){   ...............................................3
    case UTL_UOC_START:
/*-------------------------------------------------------------------------*/
/*  START                                                                  */
/*-------------------------------------------------------------------------*/
        cyear[0] = 0x20;                    /* The 21th century           */
        cyear[1] = 0x01;                    /* (2001)                     */
        uocinf->rtn_code = UTL_UOC_NML;
        break;
    case UTL_UOC_EDIT:
/*-------------------------------------------------------------------------*/
/*  EDIT                                                                   */
/*-------------------------------------------------------------------------*/
        addr_list = uocinf->data_adr;
        date_adr = addr_list->data[3];
        if (memcmp(date_adr,cyear,sizeof(cyear)) < 0){
            uocinf->unload_flg = UTL_UOC_ROWNOPUT;
        }else{
            uocinf->unload_flg = UTL_UOC_ROWPUT;
        }   .....................................................................4
        uocinf->rtn_code = UTL_UOC_NML;
        break;
    case UTL_UOC_END:
    case UTL_UOC_TERM:
/*-------------------------------------------------------------------------*/
/*  END                                                                    */
/*-------------------------------------------------------------------------*/
        uocinf->rtn_code = UTL_UOC_NML;
        break;
    default:
        strcpy(uocinf->err_msg,"Invalid request code");
        uocinf->rtn_code = UTL_UOC_ERR;
    }
    return;
}
```

［説明］

1. HiRDB が提供する，UOC 作成用のヘッダを引き込みます。
2. UOC インタフェース領域のアドレスを引数として受け取ります。
3. 呼び出し種別を判定して，それぞれの処理をします。
4. データ（日付）を判定して，格納フラグを設定します。

8.13　特殊な表に対して pdrorg を実行する場合の注意事項

8.13.1　抽象データ型がある表

(1)　表のアンロード

抽象データ型がある表をアンロードする場合の注意事項を次に示します。

(a)　LOB 属性の抽象データ型が定義されている表のアンロードだけをする場合

- 抽象データ型の列に定義されたプラグインがアンロード機能，又はコンストラクタ逆生成関数を持つ場合だけアンロードできます。
- アンロードだけをする場合は-j オプションを指定してください。

(b)　別表，又は別システムへ表を移行するためのアンロードの場合

- 同一システム内の別表へデータを移行する場合，移行元の表をアンロードしますが，プラグインによってはアンロード機能，又はコンストラクタパラメタ逆生成機能を持つときだけ移行できることがあります。移行可否については，各プラグインマニュアルを参照してください。
- 別システムの表へデータを移行する場合，プラグインがコンストラクタパラメタ逆生成関数を持つときだけ移行できることがあります。移行可否については，各プラグインマニュアルを参照してください。
- プラグインがコンストラクタ逆生成関数を持っている場合にアンロードするときは，unld_func 文にコンストラクタ逆生成関数を指定する必要があります。

(c)　抽象データ型がある表の分割条件変更をする場合

- 表の分割条件を変更する場合，まず変更前の表をアンロードします。プラグインによっては，アンロード機能，又はコンストラクタ逆生成機能を持つ場合にだけ，表の分割条件の変更ができることがあります。表の分割条件の変更可否については，各プラグインのマニュアルを参照してください。
- プラグインがコンストラクタ逆生成関数を持っている場合にアンロードするときは，unld_func 文にコンストラクタ逆生成関数を指定する必要があります。

(2)　表へのリロード

抽象データ型がある表をリロードする場合の注意事項を次に示します。

(a)　LOB 属性の抽象データ型が定義されている表の場合

アンロード時に-j オプションを指定している場合，リロード時も必ず-j オプションを指定してください。

(b)　別表，又は別システムの表の場合

- 別表，又は別システムの表へリロードする場合，tblname 文を指定します。

- アンロード時にコンストラクタパラメタ逆生成関数を指定してアンロードしたアンロードデータファイルを使用する場合，reld_func 文にコンストラクタ関数を指定する必要があります。コンストラクタ関数は，アンロード時に指定したコンストラクタパラメタ逆生成関数に対応していなければなりません。コンストラクタ関数とコンストラクタパラメタ逆生成関数の対応については，各プラグインのマニュアルを参照してください。

(c)　抽象データ型がある表の分割条件変更をする場合

アンロード時にコンストラクタパラメタ逆生成関数を指定してアンロードした場合，表の分割条件の変更後に，reld_func 文にコンストラクタ関数を指定してリロードします。コンストラクタ関数を指定してリロードする場合，コンストラクタパラメタ逆生成関数を指定してアンロードしたアンロードデータファイルだけ使用できます。

なお，指定するコンストラクタ関数は，アンロード時に指定したコンストラクタパラメタ逆生成関数に対応していなければなりません。コンストラクタ関数とコンストラクタパラメタ逆生成関数の対応については，各プラグインのマニュアルを参照してください。

(3)　pdrorg の機能と制御情報ファイルとの関係

pdrorg の機能と制御情報ファイルとの関係を次の表に示します。

表 8–41　pdrorg の機能と制御情報ファイルとの関係

pdrorg の機能	-k オプション	-j オプション	制御情報ファイル				
			unload 文	lobunld 文	tblname 文	unld_func 文	reld_func 文
LOB 列構成基表だけの再編成（1）	rorg	−	◎	×	×	×	×
	unld						
	reld						
LOB 列だけの再編成（3）	rorg	−	×	◎	×	×	×
	reld						
LOB 列構成基表，LOB 列格納用 RD エリア，及び LOB 属性格納用 RD エリアの再編成（1，2，及び 3）	rorg	◎	◎	×	×	○※2	○※2
	unld					○※3	×
	reld					×	○※3
表の分割条件変更※1（1，2，及び 3）	unld	◎	◎	×	×	○	×
	reld					×	○
別表，又は別システムへの移行※1（1，2，及び 3）	unld	◎	◎	×	×	○	×
	reld				◎	×	○

（凡例）

◎：必ず指定します。

○：任意に指定できます。

×：指定できません。

−：該当しません。

注

pdrorg の機能の括弧内の数字は，対象となる RD エリアを示しています。次の表定義の番号と対応しています。

```
CREATE TABLE …
    C1 INT,
    C2 SGMLTEXT ALLOCATE(sgmltext IN lob01)     2
    C3 BLOB(10K) IN LOB02                       3
    IN USER01                                   1
```

注※1

HiRDB/パラレルサーバの場合で，表を複数のバックエンドサーバに分割しているときは，-g オプションを指定する必要があります。

注※2

unld_func 文，及び reld_func 文を同時に指定する必要があります。

注※3

アンロード時に指定した場合は，リロード時にも必ず指定してください。

(4) アンロード元の表とリロード先の表で抽象データ型定義が異なる場合のリロード

別表への移行，又はプラグインのバージョンアップによる抽象データ型変更などで，アンロード元の表とリロード先の表とで抽象データ型定義が異なる場合があります。このような場合にリロードするときの注意事項を次に示します。

1. 次の条件をすべて満たす場合はリロードできます。
   - アンロード時に unld_func 文でコンストラクタパラメタ逆生成関数を指定してアンロードしたアンロードデータファイルを使用する場合
   - アンロード時に指定したコンストラクタパラメタ逆生成関数で逆生成されるパラメタ数とパラメタ属性がすべて一致したコンストラクタ関数を，reld_func 文に指定する場合
2. コンストラクタ関数を指定していない場合，アンロード表とリロード表とで次の項目が一致しているときはリロードできます。
   - 抽象データ型の名称
   - プラグイン ID
   - 属性の数

- 属性のデータ型，及び定義長
- LOB 属性格納用 RD エリア

(5)　表の分割条件を変更した場合のリロード

表の分割条件を変更してリロードする場合，プラグインによってはアンロード時にコンストラクタパラメタ逆生成関数を指定しなければならないものもあります。この場合，unld_func 文にコンストラクタパラメタ逆生成関数を指定してアンロードし，表の分割条件を変更した後，reld_func 文にコンストラクタ関数を指定してリロードする必要があります。

注意事項を次に示します。

- HiRDB/パラレルサーバの場合に複数のバックエンドサーバに分割しているときは，アンロード，リロード共に-g オプションを指定してください。
- RD エリア単位にリロードする場合，表の分割条件が異なるとエラーとなります。
- LOB 列，又は LOB 属性を含む抽象データ型がある表の場合，-j オプションを指定してください。

(6)　コンストラクタパラメタ逆生成関数を使用しないでアンロードした場合

コンストラクタパラメタ逆生成関数を使用しないでアンロードした場合，リロードが完了するまでプラグインの登録，及び削除はしないでください。

8.13.2　改竄防止表

(1)　制限事項

行削除禁止期間が指定されていても，pdrorg はデータの再配置をするため，行データはいったん削除されます。アンロード処理とリロード処理とを連続して行うことで，データの改竄を防止します。したがって，改竄防止表に対しては，表の再編成（-k rorg）だけ実行できます。表のアンロード（-k unld），及び表へのリロード（-k reld）は，ある条件下でだけ実行できます。

(a)　表の再編成（-k rorg）の場合

表の再編成をする場合に，使用できない機能を次に示します。

- UOC を利用した再編成
- 同期点指定の再編成

(b)　表のアンロード（-k unld）の場合

基本的には，改竄防止表に対して表のアンロードはできません。ただし，-W オプションを指定した表のアンロードはできます。

(c)　表のリロード（-k reld）の場合

基本的には，改竄防止表に対して表へのリロードはできません。ただし，表の再編成（-k rorg）のリロード処理が異常終了した場合，それを再実行するためのリロード（-k reld）は実行できます。なお，再実行する場合，-k オプション以外のオプション，及び制御文は変更しないでください（unload 文と lobunld 文を同時に指定してエラーとなった場合は，unload 文又は lobunld 文のどちらか一つを指定して再実行してください）。

(2)　実行条件

表の再編成（-k rorg），及び表へのリロード（-k reld）をする場合，改竄防止表を格納している RD エリアを pdhold コマンドで閉塞しておく必要があります（データの改竄を防止するため）。

(3)　データ未完状態

改竄防止表に対して表の再編成をする場合，エラーなどによってリロード処理が完了していない状態のことを**データ未完状態**といいます。

RD エリアがデータ未完状態かどうかを調べるには，pddbst での RD エリアの論理的解析，又は表単位の解析の実行結果で確認できます。RD エリアがデータ未完状態の場合は，表へのリロード（-k reld）を実行する必要があります。RD エリアがデータ未完状態だと，pdrorg の次の機能を実行できません。

- 表の再編成（-k rorg）
- インデクスの一括作成（-k ixmk）
- インデクスの再作成（-k ixrc）
- インデクスの再編成（-k ixor）

8.14　規則及び注意事項

8.14.1　規則

(1)　ユティリティの実行について

- pdrorg は，HiRDB が稼働中のときだけ実行できます。
- pdrorg は，シングルサーバ，又はシステムマネジャがあるサーバマシンで実行してください。

(2)　ユティリティの実行可否について

再編成する表，インデクス，及び LOB 列が格納されている RD エリアのオープン属性，RD エリアの状態によって，pdrorg の実行可否が変わります。pdrorg の実行可否については，「コマンド実行時の RD エリアの状態」を参照してください。

また，ディクショナリ表を再編成する場合は，ディクショナリ表が格納されている RD エリアを，pdhold コマンドで閉塞状態にしてください。

(3)　ユティリティの最大同時実行数

pdrorg の最大同時実行数は，システム共通定義の pd_utl_exec_mode オペランドの指定値によって異なります。

pd_utl_exec_mode=0 の場合：

最大同時実行数は 32 です。

pd_utl_exec_mode=1 の場合：

最大同時実行数は pd_max_users オペランドの指定値です。

なお，pdreclaim 及び pdpgbfon は内部で pdrorg を呼び出しているため，pdreclaim 及び pdpgbfon 実行中は pdrorg の最大同時実行数は少なくなります。例えば，pd_utl_exec_mode=0 の場合，pdrorg の最大同時実行数は 32 ですが，pdreclaim が 10 個起動中のときは，pdrorg の最大同時実行数は 22 となります。

(4)　ユティリティで処理中の表，インデクスについて

- pdrorg で処理中の表やインデクスは，対象資源に対して排他を掛けるため，ほかの UAP やユティリティからアクセスできません。なお，-k unld の場合は，option 文の unldenq オペランドの指定によって，アクセス可否が変わります。詳細については，「option 文（データ処理に関する情報の記述）」を参照してください。
- 再編成する表が格納されている RD エリアを pdhold コマンドで閉塞しておけば，UAP が閉塞エラーとなるため，UAP の実行によってユティリティの実行が排他待ちになることや，ユティリティの実行に

よって UAP の実行が排他待ちになることを防止できます。pdrorg の排他制御モードについては，「ユティリティの排他制御モード」を参照してください。

- 再編成中の表やインデクスに対して，定義系 SQL は実行しないでください。実行すると，pdrorg が異常終了します。スキーマ単位の再編成をしている場合は，該当スキーマが持つすべての表とインデクスが対象となります。再編成中の定義系 SQL 実行を防止するためには，再編成する表，及びインデクスを格納した RD エリアを，pdhold コマンドで閉塞状態にしておく必要があります。なお，再編成中に同一 RD エリア内の別の表を参照する場合には，pdhold コマンドで参照可能な閉塞状態にしておきます。
- 定義系 SQL を実行中の RD エリアにある表やインデクスに対して，データベース作成ユティリティを実行すると，排他待ちとなります。
- 再編成実行中の表に対して，定義変更や，UAP，他ユティリティを同時に実行しないでください。実行すると，UAP，他ユティリティが排他待ちエラーとなります。また，コマンドなどでアンロードとリロードの処理を分けて実行する場合，表のデータが 0 件になったり，更新結果が無効になったりします。

(5) 分割キーのレンジ変更について

リロード処理の前に分割キーのレンジを変更したとき，再定義したレンジの範囲にアンロードしたデータが納まらない場合，リロード中にそのデータを見付けた時点でエラーとなり，処理を終了します。この場合，アンロードしたデータがすべて納まるように，分割キーのレンジを定義し直し，再度リロード処理をしてください。

(6) データ連動の対象となる抽出側データベースに対するユティリティを実行時のログ取得モードについて

データ連動の対象となる抽出側データベースに対して pdrorg を実行する場合，-l オプションには n 又は p を指定してください（ログレスモード又は更新前ログ取得モードで実行してください）。ログ取得モードで実行した場合，抽出側データベースに対して実行された一部の更新情報だけが反映側データベースに転送されるため，抽出側と反映側のデータベースの内容が不整合になります。

(7) DECIMAL 型の符号正規化機能について

pdrorg では，DECIMAL 型の符号正規化機能の使用有無に関係なく，DECIMAL 型の符号部の正規化はしません。したがって，DECIMAL 型の符号正規化機能を使用しない HiRDB システムで作成したアンロードデータファイルを使用して，DECIMAL 型の符号正規化機能を使用する HiRDB システムへのリロードはできません。

データベース中の正規化されていない DECIMAL 型の符号部を正規化する場合は，-W オプション指定のアンロードデータファイルを作成し，データベース作成ユティリティでデータを格納し直してください。
DECIMAL 型の符号正規化機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(8)　インデクス作成時のファイルの出力先について

インデクス作成時，制御情報ファイルでファイルの出力先を指定しない場合，次の表に示すディレクトリにファイルを出力します。

表 8–42　pdrorg がファイルを出力するディレクトリ

制御文※1の指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※2の指定	
		あり	なし
あり	制御文に指定したディレクトリ又はファイル		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※1

idxwork 文，sort 文

注※2

システム環境変数の指定

(9)　インデクス情報ファイルの出力先について

pdrorg の処理内容と制御文の指定によって，インデクス情報ファイルの出力先が変わります。インデクス情報ファイルの出力先を次の表に示します。

表 8–43　インデクス情報ファイルの出力先

pdrorg の処理内容	制御文			インデクス情報ファイルの出力先
	index 文	idxname 文	idxwork 文	
-k rorg， -k reld， -k ixmk	あり※	×	−	index 文で指定したインデクス情報ファイル
	なし	×	あり	idxwork 文で指定したディレクトリ
	なし	×	なし	表「pdrorg がファイルを出力するディレクトリ」参照
-k ixrc， -k ixor	×	あり	あり	idxwork 文で指定したディレクトリ
	×	あり	なし	表「pdrorg がファイルを出力するディレクトリ」参照
	あり	×	−	index 文で指定したインデクス情報ファイル

（凡例）

×：指定できません。

−：指定の有無は，インデクス情報ファイルの出力先に関係しません。

注※

横分割表の場合は，指定した RD エリアだけが対象であるため，指定していない RD エリアについては index 文なしの項目が該当します。

(10)　リバランス表の再編成

リバランス表に RD エリアを追加した場合，そのリバランス表に対して pdrbal を実行（リターンコード=0）していないと，追加した RD エリアの再編成（RD エリア単位の再編成）はできません。

リバランス表に RD エリアを追加した場合，追加した RD エリアに対する pdrorg の実行可否を次の表に示します。

表 8–44　リバランス表に追加した RD エリアに対する pdrorg の実行可否

-k オプション		表の状態				
		RD エリアの追加～pdrbal 実行前		pdrbal 途中※（リターンコード 4）		pdrbal 完了後（リターンコード 0）
		既存	追加	既存	追加	
unld		○	×	○	○	○
rorg，reld	表単位	○	×	○※	○※	○
	RD エリア単位	○	×	×	×	○
ixmk		○	○	○	○	○
ixrc		○	○	○	○	○
ixor		○	○	○	○	○

（凡例）

既存：表定義（CREATE TABLE）で指定している既存の RD エリア

追加：ALTER TABLE ADD RDAREA で追加した RD エリア

○：pdrorg を実行できます。

×：pdrorg は実行できません。

注※

リバランス途中にリロードする場合は，-g オプションを指定してください。また，LOB 列，及び LOB 属性の抽象データ型列がある場合は，同時に-j オプションも指定してください。-g オプションを指定して再編成をすると，データの再配置はされますが，そのリバランス表はまだリバランス完了となっていないため，リロード後に必ず pdrbal を実行してリターンコード=0 になることを確認してください。

表をアンロードした後に，その表に RD エリアを追加し，リロードする場合の手順を次に示します。

1. pdrorg を実行（-k unld）

2. ALTER TABLE ADD RDAREA で RD エリアを追加
3. PURGE TABLE で全行削除
4. pdrbal を実行
5. pdrorg を実行（-k reld）

上記の手順を実行する場合，pdrorg には必ず-g オプションを指定してください。-g オプションを指定しないとエラーとなります。また，LOB 列又は LOB 属性の抽象データ型列があるリバランス表の場合は，同時に-j オプションも指定してください。

(11) 更新凍結状態の RD エリアに対する処理

更新凍結状態の RD エリアにある表に対して，-k rorg（再編成）又は reld（リロード）を実行する場合，ユーザ LOB 用 RD エリアの管理領域を更新できないため，エラーとなります。また，LOB 列構成基表だけを再編成又はリロードする場合も，同様にエラーとなります。

(12) SEGMENT REUSE 指定の表に対するリロード

SEGMENT REUSE 指定の表にリロードする場合，SEGMENT REUSE の指定は無効となります。

(13) 32bit 版と 64bit 版の HiRDB のアンロードデータファイルの互換性について

32bit 版，又は 64bit 版の HiRDB でアンロードしたアンロードデータファイルは，お互いの bit モードの HiRDB でリロードし，データを移行できます。移行できる条件は，同一の bit モードの HiRDB 間で移行できる条件と同じです。

(14) 一時表に対する pdrorg の実行

一時表に対して，pdrorg は実行できません。-t オプションに一時表を指定して実行すると，pdrorg は KFPL15231-E メッセージを出力して，エラー終了します。スキーマ単位の再編成（-t オプションに all を指定）の場合，pdrorg はスキーマに定義された一時表の処理をスキップします。

(15) 再編成によって RD エリアの格納不足が発生するケースについて

次のケースでは，再編成のリロード時に RD エリアの容量不足が発生するおそれがあります。容量不足が発生した場合は，リロードの再実行前に RD エリアを拡張してください。

- リバランス機能を持つハッシュ関数で横分割された表の場合

  リバランス機能を持つハッシュ関数で FIX ハッシュ分割した表を再編成する場合に，アンロードとリロードの間に分割条件を変更すると，表格納 RD エリアに空きページがあっても，空きセグメントがないために RD エリアの容量が不足することがあります。これは，格納条件の変更によって，アンロードしたときに格納されていたセグメントとは別のセグメントにリロードする必要が生じたが，新たなセグメントを確保できないためです。

- 定義時に空き領域比率（PCTFREE オペランド）に 0 以外を指定した表，インデクスの場合

  表の運用中に空き領域がなくなり，満杯状態になった RD エリアの表又はインデクスに対して再編成をした場合に，定義時の空き領域比率が適用されて RD エリアの容量不足が発生することがあります。

  このケースで，暫定的に RD エリアの拡張をしないでリロード処理を完了させたい場合，option 制御文の tblfree，idxfree オペランドを指定すると，一時的に空き領域比率を変更できます。詳細については，「option 文（データ処理に関する情報の記述）」を参照してください。
- データが格納されている表に ON ROW EXISTS 指定の列追加をした場合

  表に追加した列の既定値が格納されるため，表容量が増加し，RD エリアの容量不足が発生することがあります。これは，列を追加した時点では，表に既定値データが実際に格納されていない状態となり，行の更新や再編成のタイミングで追加列のデータが格納されるためです。

(16)　再編成によって DB 行長が伸長するケースについて

次のケースでは，再編成のリロード時に DB 行長が伸長し，行長が RD エリアのページ長を超えてエラーとなるおそれがあります。エラーが発生した場合は，KFPA11809-I メッセージに従って，RD エリアのページ長を拡張してください。

- データが格納されている表に ON ROW EXISTS 指定の列追加をした場合

8.14.2　注意事項

(1)　リターンコードについて

pdrorg のリターンコードを次に示します。

0：正常終了。

4：正常終了（処理を一部スキップしました）。

8：異常終了。

(2)　シンクポイントダンプの取得について

再編成処理（pdrorg -k rorg）の実行中は，一つの処理が完了するまでトランザクションが決着しないため，シンクポイントを取得できません。一つの処理とは，アンロード処理，リロード処理，及びインデクス作成処理のことをいいます。

このため，pdrorg コマンドとともにほかの UAP などを実行した場合に，システム障害が発生したとき，再開始に要する時間が長くなるため，できるだけ pdrorg コマンドとほかの UAP などは同時に実行しないようにしてください。

同期点指定の再編成，及び同期点指定のリロードの場合，任意の行数単位にシンクポイントを取得できます。このため，通常のシンクポイントを取得しない再編成及びリロードに比べると，異常終了時の再開始に要する時間は短くなります。

なお，アンロード処理だけ（pdrorg -k unld）の場合はデータベースを更新しないため，pdrorg コマンドの実行中でもシンクポイントが取得できます。

(3)　ユティリティ実行中の処理の中断について

pdrorg 実行中に処理を中断したい場合は，pdcancel コマンドを使用します。なお，pdrorg が無応答障害（定例ジョブなど，一定時間で再編成処理が終わるはずなのに終わらない）と判断して強制終了させる場合は，pdls コマンド（-d rpc -a 指定）の表示結果をリダイレクトでファイルに取得してください。

この場合，処理はロールバックされます。このときのデータベースの状態，及び回復方法については，「エラー時のデータベースの状態とその回復方法」を参照してください。

再編成時期予測機能を使用している場合，pdrorg をシグナル割り込みで強制終了すると，運用履歴表が更新できません。再編成時期予測機能を使用している場合に pdrorg を終了させるときは，必ず pdcancel コマンドを使用してください。

(4)　ユティリティ実行中の定義情報の変更について

表をアンロードしてからリロードするまでの間に，次の定義系 SQL は実行しないでください。

- ALTER TABLE
- DROP TABLE，CREATE TABLE による表定義の再作成

ただし，分割キーの分割条件を変更する場合及びハッシュ関数を変更する場合は実行できます。

(5)　ユティリティ実行後の結果について

pdrorg の結果は，データベース状態解析ユティリティで確認できます。

(6)　リロード時の分割キーの分割条件の変更について

リロード時に分割キーの分割条件を変更する場合，次の指定をするとエラーとなります。

- -k オプションでの rorg の指定
- -r オプションの指定

(7)　RD エリア再初期化時のインデクスの状態について

データベース構成変更ユティリティで RD エリアを再初期化した場合，再初期化対象の RD エリアに関連するインデクスが未完状態となることがあります。再初期化によって非分割キーインデクスが未完状態となっ

た場合は，必ずインデクス一括作成モードで表単位に再編成してください。RD エリアの再初期化前に出力されたインデクスファイルでインデクスを一括作成すると，データベースが破壊されることがあります。

(8)　系切り替え機能を利用する場合について

系切り替え機能を利用する場合，-k rorg で再編成するのではなく，-k unld と-k reld に分けて再編成することをお勧めします。この場合，アンロード処理が完了したアンロードデータファイルを使用して，系切り替え先でリロードが続行できます。このとき，アンロードデータファイルは DAT などの磁気テープ装置，又は HiRDB ファイルを使用してください。ブロック入出力をするファイルでアンロードデータファイルを作成すると，系切り替えで OS のバッファが破棄されるため，アンロード処理がリターンコード 0 で終了していても，アンロードデータファイルに出力されている内容は保証されません。

(9)　アンロードデータファイル作成時，pdrorg が異常終了した場合について

アンロードデータファイル作成時，pdrorg が異常終了した場合，不正なファイルが残っていることがあります。このとき，pdrorg を再実行すれば，残っているファイルに上書きされます。そのため，再実行しない場合はそのファイルを削除してください。

表の再編成中，リロード処理で pdrorg が異常終了した場合，ロールバックされてデータベース中のデータが消失した状態になります。これを回復する場合は，アンロード処理で作成されたアンロードデータファイルを使用して，リロード処理だけを実行します。このとき，複数の表を一つのアンロードデータファイルに連続して出力するような実行方法にすると，異常終了した pdrorg の後に続く別の表のアンロード処理でアンロードデータファイルが上書きされ，アンロードデータファイル中にもデータベース中にもデータがない状態になります。そのため，バックアップからの回復ができない場合は，必ず表ごとに別々のアンロードデータファイルを指定してください。

(10)　システム間のアンロード，リロードについて

システム間のアンロード，及びリロードをする場合の注意事項を次に示します。

- 異なるバイトオーダのシステム間でのアンロード及びリロードはできません。
- 異なる文字体系のシステム間でのアンロード及びリロードはできません。
- アンロードデータファイル中には，行データ以外に HiRDB のバージョン固有の制御情報が入っています。このため，あるバージョンの HiRDB でアンロードしたアンロードデータファイルを，異なるバージョンの HiRDB へリロードする場合，上位互換は保証されますが，下位互換は保証されません。下位バージョンの HiRDB へデータをリロードする場合は，-W オプションを指定してアンロードしたアンロードデータファイルを使用してください。
- 表に抽象データ型列が定義されている場合，アンロードするときにコンストラクタパラメタ逆生成関数を使用する必要があります。
- アンロード元の表とリロード先の表の表定義が異なる場合，条件によってリロードができないことがあります。リロードができない条件については，「アンロード元の表とリロード先の表の表定義が異なる場合のリロード実行可否」を参照してください。

(11) 分割キーのレンジ，又はハッシュ関数を変更した場合について

分割キーのレンジを変更した場合，又はハッシュ関数を変更した場合は，次に示すときだけ変更した分割条件に対応して，RD エリアにデータを振り分けて格納します。ただし，分割キーの列を変更した場合，及び LOB 列がある表で-j オプションを指定していない場合を除きます。

- HiRDB/シングルサーバでリロードする場合
- HiRDB/パラレルサーバで，-g オプション指定のアンロードデータファイルからリロードする場合

また，キーレンジ分割，フレキシブルハッシュ分割，又は FIX ハッシュ分割をそれぞれ別の分割方法に変更する手順については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(12) 表定義が同じ場合のアンロード，リロードについて

表定義が同じ場合，HiRDB/シングルサーバの表をアンロードして HiRDB/パラレルサーバの表へリロード，又はその逆に HiRDB/パラレルサーバの表をアンロードして HiRDB/シングルサーバの表へリロードできます。ただし，LOB 列がある表の場合は，-j オプションを指定してアンロード・リロードする必要があります。

HiRDB/パラレルサーバの表をアンロードして HiRDB/シングルサーバにリロードする場合，複数サーバに分割格納された表をアンロードするときには-g オプションを指定して 1 個のアンロードデータファイルを作成する必要があります。

表定義が異なる場合や，サーバ単位に出力されるアンロードデータファイルを使用したい場合は，-W オプション指定のアンロードをして，pdload でデータロードしてください。また，横分割表にクラスタキーインデクスを定義している場合，アンロードデータファイル中のデータは RD エリア単位にはクラスタキー順に並んでいますが，表全体としてはクラスタキー順に並んでいない状態となります。この場合，クラスタリング順序エラーとして扱われます。したがって，pdload で-x オプションを指定するか，又は RD エリア単位に処理をする必要があります。

(13) 抽象データ型の列を持つ表へのリロード処理について

- LOB 列構成基表格納 RD エリア，LOB 列格納 RD エリア，又は LOB 属性格納 RD エリアのどれかを再初期化した場合，-j 指定でアンロードしたアンロードデータファイルだけリロードできます。このとき，リロード時にも-j オプションを指定してください。
- -g オプションを指定しないで表単位に処理をする場合，表があるバックエンドサーバ数分のディクショナリサーバプロセスが必要になります。

(14) プラグインインデクスに対するインデクスの一括作成及びインデクスの再作成の実行可否について

プラグインインデクスに対するインデクスの一括作成及びインデクスの再作成の実行可否を次に示します。

提供機能	インデクスの一括作成	インデクスの再作成
プラグインインデクス遅延一括作成機能	実行できます	実行できます
プラグインインデクス一括作成部分回復機能	実行できます	実行できます
上記機能なし	実行できません	実行できます

プラグインインデクスに対してインデクスの一括作成（-k ixmk 指定）を実行する場合は，プラグインが「インデクス一括作成部分回復機能」を提供している必要があります。提供していない場合は実行できないため，インデクスの再作成（-k ixrc 指定）でインデクスを作り直す運用となります。

「プラグインインデクス遅延一括作成機能」を使用して作成されたインデクス情報ファイルを指定してのインデクス一括作成を実行して正常終了した場合，指定したインデクス情報ファイルを削除します。また，この機能で使用するインデクス情報ファイル出力ディレクトリ（サーバ定義の pd_plugin_ixmk_dir 指定ディレクトリ）は，インデクス一括作成（-k ixmk 指定）処理以外では pdrorg が使用するファイル入出力領域として使用しないでください。

(15) HiRDB Text Search Plug-in の差分インデクス機能を使用している場合のユティリティが更新するインデクスについて

HiRDB Text Search Plug-in の差分インデクス機能を使用している場合，pdrorg が更新するインデクスを次に示します。

インデクス作成方法	環境変数 PDPLUGINNSUB の指定		
	Y	N	設定なし
インデクス再作成の場合（-k ixrc 指定）	M	M	M
全インデクスを作成する場合（-k ixmk 指定）	M	M	M
追加データのインデクスを作成する場合（-k ixmk 指定）	S	M	S

（凡例）

M：MASTER インデクスを更新

S：差分インデクスを更新

(16) 再編成，又はリロード対象の表を基にリストを作成している場合について

再編成，又はリロード対象の表を基にリストを作成している場合，再編成後，又はリロード後にリストを使用して検索したときには，次のような現象が発生します。

- 誤った行を取り出す。
- 検索した行が見付からない。

このような場合，リストを使用して検索する前に，リストを再作成する必要があります。

(17)　DVD-RAM 装置の使用について

pdrorg のアンロードデータファイルとして，DVD-RAM 装置を利用できます。ただし，使用する OS で各装置が使用できる場合に限ります。

HiRDB ファイルとして使用する場合，HiRDB ファイルシステム領域作成時（pdfmkfs コマンド実行時）にセクタ長を指定する必要があります。

(18)　ユティリティ実行中の OS リソース不足を回避する方法

システム全体で大容量のファイル（システム全体で同時に 100 ギガバイト以上）を扱う場合，ファイルキャッシュが増大し，Windows のシステムリソース不足になることがあります。大容量のファイルを扱う場合，pdfmkfs コマンド（-k UTL）で作成した HiRDB ファイルシステム領域にアンロードデータファイルを作成すると，ファイルキャッシュを使用しないでアクセスするため，システムリソース不足を回避できます。ただし，ファイルキャッシュを使用しないでアクセスするため，性能は劣化します。

(19)　ユティリティ実行時に使用できるファイルの媒体について

pdrorg 実行時に使用できるファイルの媒体を次に示します。

ファイル	ファイル	HiRDB ファイル	固定長ブロックテープ	可変長ブロックテープ
アンロードデータファイル	○	○	○	○
LOB データのアンロードファイル	○	○	○	○
インデクス情報ファイル	○	△	×	×
ソート用ワークファイル	○	×	×	×
処理結果ファイル	○	×	×	×

（凡例）

○：使用できます。

△：インデクス遅延一括作成機能を利用するときだけ使用できます。

×：使用できません。

(20)　コマンド実行画面に出力されるメッセージ

pdrorg は，標準出力に経過メッセージを出力しながら処理をします。また，エラー発生時は，標準エラー出力にエラーメッセージを出力します。標準出力や標準エラー出力への出力が抑止されている環境で pdrorg を実行した場合，メッセージ出力待ちで pdrorg が無応答状態となったり，メッセージログファイルに KFPL20003-E メッセージを出力して pdrorg が異常終了したりすることがあります。そのため，標準出力や標準エラー出力への出力ができない環境では pdrorg を実行しないでください。なお，標準出力，及び標準エラー出力に出力するメッセージの順序や数は，メッセージログファイルやイベントログと一致しない

ことがあります。正確にメッセージを把握したい場合は，メッセージログファイル又はイベントログを参照してください。

(21)　共用表の再編成

共用表を再編成する場合，共用表，及び処理対象の表に定義した共用インデクスを格納している RD エリアに対して，EX モードで排他を掛けます。そのため，該当する RD エリアに定義したほかの表やインデクスも参照，及び更新ができません。共用表を再編成する場合の排他制御モードについては，「ユティリティの排他制御モード」を参照してください。

(22)　参照制約又は検査制約を定義した表の再編成

参照制約又は検査制約を定義した表に対して，再編成又はリロードを実行する場合の検査保留状態の設定可否については，「constraint 文（検査保留状態に関する記述）」を参照してください。また，検査保留状態については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

なお，次の条件をすべて満たす場合は，検査保留状態となった表の再編成（-k rorg）は実行できません。

- 表単位に再編成を実行する（-r オプションを指定しない）。
- フレキシブルハッシュ分割，又はフレキシブルハッシュ分割を含むマトリクス分割で分割された表である。
- HiRDB/シングルサーバである，又は HiRDB/パラレルサーバで-g オプションを指定している。

表が検査保留状態かどうかは，ディクショナリ表（SQL_TABLES 表）の CHECK_PEND 列，又は CHECK_PEND2 列の値から判断します（列値が'C'の場合は検査保留状態）。

(23)　再編成時期予測機能との関係

表やインデクスを再編成した運用履歴は，再編成時期予測結果に反映されます。

また，pdrorg が異常終了した状態のまま，pddbst の状態解析結果蓄積機能を実行すると，再編成完了後の状態で再編成時期を予測できないため，予測結果が不正※となります。したがって，pdrorg が異常終了した場合は，pdrorg を再実行して正常終了させた状態で，pddbst の状態解析結果蓄積機能を実行してください。

注※

例えば，リロード中に異常終了してロールバックした状態の DB の情報を蓄積すると，空の DB を基に予測してしまいます。

(24)　BLOB 列を含む表を再編成する場合

BLOB 列を含む表の再編成をする場合，データをアンロードしてからリロードするまでの間に，RD エリアの名称を変更しないでください。RD エリアの名称を変更する前に取得したアンロードデータファイルを，名称を変更した RD エリアの表にリロードするとエラーとなります。

(25)　文字集合 UTF-16 を指定した CHAR 型の列を分割キーとした表のエクスポート

文字集合 UTF-16 を指定した CHAR 型の列を，キーレンジ分割又はマトリクス分割（境界値指定のキーレンジ分割）の分割キーとして指定した表の場合，分割キーの長さが列の定義長を超えるときは，その表をエクスポートできません。

(26)　FLOAT，SMALLFLT 列を含む表のアンロードについて

FLOAT 型・SMALLFLT 型の列を含む表の pdload 用アンロードファイルを作成する場合，バージョン 10-05 以前と比較して，次の変更点があります。

- dat，extdat，fixtext 形式でのアンロード時，FLOAT 型・SMALLFLT 型の出力値の形式が変わります。

詳細はマニュアル「HiRDB システム導入・設計ガイド」の「バージョン 10-05 以前との FLOAT 型・SMALLFLT 型の差異について」を参照してください。

なお，作成した pdload 用アンロードファイルをデータベース作成ユティリティによってロードした際の格納結果については，バージョン 10-05 以前と差異はありません。

8.14.3　BOM を持つファイルの使用

pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdrorg の入力ファイルには，BOM を持つファイルを使用できることがあります。pdrorg での BOM を持つファイルの使用可否を次の表に示します。なお，pdrorg の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdrorg で出力されるファイルには BOM は付きません。

表 8–45　pdrorg での BOM を持つファイルの使用可否（UTF-8 の場合）

制御文	入力ファイル	BOM を持つファイルの使用
–	制御情報ファイル	○
unload	アンロードデータファイル	×
index	インデクス情報ファイル	×
lobunld	LOB データのアンロードファイル	×

（凡例）

○：使用できます。

×：使用できません。

–：該当しません。

8.15　エラー時のデータベースの状態とその回復方法

pdrorg 実行時にエラーが発生した場合は，次の表を参照し，どこまで処理がされているかを調べて，該当する参照先の対処を実施してください。

アンロード処理とリロード処理を連続して実行する再編成（-k rorg）では，リロード中にエラーが発生した表はデータ未完状態となり，pdrorg の再実行はリロード（-k reld）だけ実行できます。また，データ未完状態の表は，pdload での追加モードのデータロード，及び pdrbal でのリバランス処理はできません。なお，アンロードデータファイルを削除してリロードができない状態の場合は，PURGE TABLE 文を実行するか，又は作成モードの pdload を実行することでデータ未完状態は解除できます。

対処の実施は，発生したエラー原因を取り除いた後に行ってください。

表 8–46　pdrorg 実行時に出力されるメッセージと対処方法（1/2）

<table>
<tr><th rowspan="3">pdrorg 実行時のオプション指定値</th><th colspan="8">pdrorg 実行時に出力されるメッセージ</th></tr>
<tr><th>KFPL 00712-I，KFPL 00732-I</th><th>KFPL 00714-I，KFPL 00734-I</th><th>KFPL 00721-I</th><th>KFPL 00712-I，KFPL 00732-I</th><th>KFPL 00714-I，KFPL 00734-I</th><th>KFPL 00725-I</th><th>KFPL 00715-I</th><th>KFPL 00716-I</th></tr>
<tr><th>アンロード開始</th><th>アンロード完了</th><th>データ削除</th><th>リロード開始</th><th>リロード完了</th><th>インデクス情報検索</th><th>インデクス作成開始</th><th>インデクス作成完了</th></tr>
<tr><td>-k rorg -j なし</td><td rowspan="4">「アンロード時」</td><td rowspan="4">「アンロード時」</td><td>「リロード時」</td><td>「リロード時」</td><td>「リロード時」</td><td rowspan="5">–</td><td>「リロード時」</td><td>「リロード時」</td></tr>
<tr><td>-k rorg -j あり</td><td rowspan="2">–</td><td rowspan="2">–</td><td rowspan="2">–</td><td rowspan="2">–</td><td rowspan="2">–</td></tr>
<tr><td>-k unld -j なし</td></tr>
<tr><td>-k unld -j あり</td></tr>
<tr><td>-k reld -j なし</td><td rowspan="5">–</td><td rowspan="5">–</td><td>「リロード時」</td><td>「リロード時」</td><td>「リロード時」</td><td>「リロード時」</td><td>「リロード時」</td></tr>
<tr><td>-k reld -j あり</td><td rowspan="3">–</td><td rowspan="3">–</td><td rowspan="3">–</td><td rowspan="2">「インデクスの一括作成，及びインデクスの再作成実行時」</td><td rowspan="2">「インデクスの一括作成，及びインデクスの再作成実行時」</td></tr>
<tr><td>-k ixmk</td></tr>
<tr><td>-k ixrc</td><td>「インデクスの一括作成，及びインデクスの再作成実行時」</td></tr>
<tr><td>-k ixor</td><td>「インデクスの再編成実行時」</td><td>「インデクスの再編成実行時」</td><td>「インデクスの再編成実行時」</td></tr>
</table>

（凡例）－：該当しません。

表 8–47　pdrorg 実行時に出力されるメッセージと対処方法（2/2）

pdrorg 実行時のオプション指定値	pdrorg 実行時に出力されるメッセージ				
	KFPL00712-I，KFPL00732-I	KFPL00714-I，KFPL00734-I	KFPL00721-I	KFPL00712-I，KFPL00732-I	KFPL00714-I，KFPL00734-I
	LOB データアンロード開始	LOB データアンロード完了	LOB データ削除	LOB データリロード開始	LOB データリロード完了
-k rorg -j なし	「LOB データのアンロード時」	「LOB データのアンロード時」	「LOB データのリロード時」	「LOB データのリロード時」	「LOB データのリロード時」
-k rorg -j あり	－	－	－	－	－
-k unld -j なし	「LOB データのアンロード時」	「LOB データのアンロード時」			
-k unld -j あり	－	－			
-k reld -j なし	「LOB データのアンロード時」	「LOB データのアンロード時」	「LOB データのリロード時」	「LOB データのリロード時」	「LOB データのリロード時」
-k reld -j あり	－	－	－	－	－
-k ixmk					
-k ixrc					
-k ixor					

（凡例）－：該当しません。

8.15.1　アンロード時

アンロード処理中には，エラーが発生してもデータベースの内容はアンロード前と変わりません。したがって，エラーが発生した場合は，エラー要因を取り除いた後に再度アンロードしてください。

8.15.2　リロード時

リロード中にエラーが発生したときの，データベースの状態とその回復方法を次の表に示します。pdrorg 実行時のオプションの指定によって回復方法が異なります。リロード中にエラーが発生したときは，そのときに指定したオプションとエラー発生直前に出力されたメッセージを次の表で調べて，回復してください。

表 8–48　リロード中にエラーが発生したときのデータベースの状態とその回復方法

オプション		区分	エラー発生直前の出力メッセージ					
-l 指定	-i 指定		なし	KFPL00721-I	KFPL00712-I，KFPL00732-I	KFPL00714-I，KFPL00734-I	KFPL00715-I	KFPL00716-I
				行データ削除完了	リロード開始	リロード完了	インデクス作成開始	インデクス作成完了
a	s	状態	実行前状態	空※2	空※2※4	表作成完了	–	–
		回復	再実行	リロード再実行※1	リロード再実行※1	不要		
	c	状態	実行前状態	空※2	空※2※4	表作成完了	インデクス未作成	インデクス作成完了
		回復	再実行	リロード再実行※1	リロード再実行※1	インデクス作成	インデクス作成	不要
p	s	状態	実行前状態	空※2	空※2※4	表作成完了	–	–
		回復	再実行	リロード再実行※1	リロード再実行※1	不要		
	c	状態	実行前状態	空※2	空※2※4	表作成完了	インデクス未作成	インデクス作成完了
		回復	再実行	リロード再実行※1	リロード再実行※1	インデクス作成	インデクス作成	不要
n	s	状態	保証なし	空※2	保証なし	表作成完了	–	–
		回復	DB 回復後再実行	DB 回復後リロード再実行	DB 回復後リロード再実行	不要		
	c	状態	保証なし	空※2	保証なし	表作成完了	インデクス未作成	インデクス作成完了
		回復	DB 回復後再実行	DB 回復後リロード再実行	DB 回復後リロード再実行	DB 回復後リロード再実行※3	DB 回復後リロード再実行※3	不要

（凡例）

–：

　該当しません。

区分：

　状態　エラー発生時のデータベースの状態

　回復　回復方法

注※1

ログ取得モード又は更新前ログ取得モード実行時にエラーが発生した場合，表データが空になっている場合があるので注意してください。この場合，アンロードから再実行すると空のデータをアンロードするため，表データがなくなります。したがって，この場合はリロードだけ再実行してください。

また，-j オプションなしで unload 文及び lobunld 文を指定して再編成をしている場合に，リロード再実行となったときは，次の手順で実行してください。

1. unload 文だけを指定して LOB 列構成基表へリロードをする。
2. lobunld 文だけを指定して表の再編成をする。

注※2

KFPL00721-I メッセージで出力された RD エリアだけデータが空になります。

注※3

表格納用 RD エリアとインデクス格納用 RD エリアが異なる場合，表格納用 RD エリアの閉塞を解除した後，インデクス格納用 RD エリアを再初期化して，インデクスの再作成（-k ixrc）だけ実行する方法もあります。詳細については，マニュアル「HiRDB システム運用ガイド」のインデクスの一括作成処理中にエラーが発生したときの対処方法を参照してください。

注※4

同期点指定の場合，KFPL00800-I メッセージに出力されている行数までデータは格納されます。

8.15.3　インデクスの一括作成，及びインデクスの再作成実行時

インデクスの一括作成，及びインデクスの再作成時にエラーが発生したときの，データベースの状態とその回復方法を次の表に示します。

表 8–49　インデクスの一括作成，及びインデクスの再作成時にエラーが発生したときのデータベースの状態とその回復方法

-l オプション	区分		エラー発生直前の出力メッセージ			
			なし	KFPL00725-I	KFPL00715-I	KFPL00716-I
				インデクスアンロード開始	インデクス一括作成開始	インデクス一括作成完了
a 又は p	状態		実行前状態	実行前状態	インデクス未作成状態	インデクス作成完了
	回復		再作成を再実行	再作成を再実行	一括作成実行※	不要
n	状態	表	実行前状態	実行前状態	実行前状態	実行前状態
		インデクス	実行前状態	実行前状態	保証なし	作成完了
	回復	表	不要	不要	不要	不要

-l オプション	区分		エラー発生直前の出力メッセージ			
			なし	KFPL00725-I	KFPL00715-I	KFPL00716-I
				インデクスアンロード開始	インデクス一括作成開始	インデクス一括作成完了
		インデクス	再作成を再実行	再作成を再実行	DB 回復後に一括作成を再実行※	不要

（凡例）

区分：

状態　エラー発生時のデータベースの状態

回復　回復方法

注※

対象インデクスがプラグインインデクスの場合，「プラグインインデクス一括作成部分回復機能」が提供されている場合だけ実行できます。提供されていない場合は，インデクス再作成を再実行します。

8.15.4　インデクスの再編成実行時

インデクスの再編成時にエラーが発生したときの，データベースの状態とその回復方法を次の表に示します。

表 8–50　インデクスの再編成時にエラーが発生したときのデータベースの状態とその回復方法

-l オプション	区分		エラー発生直前の出力メッセージ			
			なし	KFPL00725-I	KFPL00715-I	KFPL00716-I
				インデクスアンロード開始	インデクス一括作成開始	インデクス一括作成完了
a 又は p	状態		実行前状態	実行前状態	インデクス未作成状態	インデクス作成完了
	回復		再編成を再実行	再編成を再実行	一括作成実行※	不要
n	状態	表	実行前状態	実行前状態	実行前状態	実行前状態
		インデクス	実行前状態	実行前状態	保証なし	作成完了
	回復	表	不要	不要	不要	不要
		インデクス	再編成を再実行	再編成を再実行	DB 回復後に一括作成を再実行※	不要

（凡例）

区分：

状態　エラー発生時のデータベースの状態

回復　回復方法

注※

インデクス格納用 RD エリアをバックアップからではなく，RD エリアの再初期化で回復した場合は，インデクスの再作成を再実行してください。

8.15.5　LOB データのアンロード時

LOB データのアンロード処理中にエラーが発生しても，データベースの内容はアンロード前と変わりません。したがって，エラーが発生した場合は，エラー要因を取り除いた後，再度 LOB 列だけアンロードしてください。LOB 列構成基表と LOB 列を別々に再編成する場合は unload 文と lobunld 文を両方指定しますが，LOB データのアンロード中にエラーとなったときは，LOB データのアンロードから開始するために，unload 文を削除し，lobunld 文だけ指定して再度アンロードしてください。

8.15.6　LOB データのリロード時

LOB データのリロード処理時にエラーが発生したときのデータベースの状態とその回復方法を次の表に示します。

表 8–51　LOB データのリロード処理時にエラーが発生したときのデータベースの状態とその回復方法

-l オプション	区分		エラー発生直前の出力メッセージ			
			なし	KFPL00721-I	KFPL00712-I，KFPL00732-I	KFPL00714-I，KFPL00734-I
				LOB データ削除	LOB データリロード開始	LOB データリロード完了
a 又は p	状態	表，インデクス	実行前状態	LOB 列構成基表再編成完了	LOB 列構成基表再編成完了	表再編成完了
		LOB 列	実行前状態	空	空	LOB 列再編成完了
	回復	表，インデクス	不要	不要	不要	不要
		LOB 列	LOB 列だけリロード再実行	LOB 列だけリロード再実行	LOB 列だけリロード再実行	不要
n	状態	表，インデクス	ログレス閉塞	ログレス閉塞	ログレス閉塞	表再編成完了
		LOB 列	保証なし	保証なし	保証なし	LOB 列再編成完了
	回復	表，インデクス	DB 回復後再実行※	DB 回復後再実行※	DB 回復後再実行※	不要

-l オプション	区分		エラー発生直前の出力メッセージ			
			なし	KFPL00721-I	KFPL00712-I，KFPL00732-I	KFPL00714-I，KFPL00734-I
				LOB データ削除	LOB データリロード開始	LOB データリロード完了
		LOB 列	DB 回復後再実行※	DB 回復後再実行※	DB 回復後再実行※	不要

（凡例）

区分：

状態　エラー発生時のデータベースの状態

回復　回復方法

注

-j オプション指定なしで LOB 列構成基表と LOB 列を同時に再編成する場合，unload 文と lobunld 文を指定します。LOB 列へのアンロード，リロード処理中にエラーが発生して再実行する場合は，LOB 列構成基表の再編成処理は完了しているため，unload 文は削除し，lobunld 文だけにする必要があります。また，アンロード処理中の異常終了での再実行の場合は-k rorg のままでよいですが，リロード処理中の異常終了での再実行の場合は-k reld に変更する必要があります。

注※

ログレスモードの場合，LOB 列へのリロード中に異常終了した場合でも，表（LOB 列構成基表，LOB 列，LOB 属性の抽象データ型列，及びインデクス）をバックアップから回復した後，再度表の再編成をする必要があります。

8.15.7　ディクショナリ表のメンテナンス実行時

ディクショナリ表のメンテナンス処理時にエラーが発生したときのデータベースの状態とその回復方法を次の表に示します。

表 8–52　ディクショナリ表のメンテナンス処理時にエラーが発生したときのデータベースの状態とその回復方法

エラー発生直前の出力メッセージ	状態	回復
なし	実行前状態	ディクショナリ表のメンテナンスを再実行
KFPL00717-I	実行前状態	ディクショナリ表のメンテナンスを再実行
KFPL00743-I	インデクス未作成状態	ディクショナリ表のメンテナンスを再実行
KFPL00719-I	ディクショナリ表のメンテナンス完了	不要

9 ディクショナリ搬出入ユティリティ（pdexp）

この章では，HiRDB システム間で表の定義情報，トリガ定義情報，又はストアドプロシジャの情報を移行するディクショナリ搬出入ユティリティ（pdexp）について説明します。

なお，IVS 対応 UTF-8 文字コード及び IVS 対応 UTF-16 文字集合を使用する場合は，この章の記載を次のように読み替えてください。

UTF-8 -> IVS 対応 UTF-8

UTF-16 -> IVS 対応 UTF-16

9.1 概要

ディクショナリ搬出入ユティリティは，HiRDB システム間で表の定義情報，トリガ定義情報，又はストアドプロシジャの情報を移行するユティリティです。

ディクショナリ搬出入ユティリティは，表の定義情報，トリガ定義情報，又はストアドプロシジャの情報を搬出ファイルに出力し，任意の HiRDB システムに入力することで移行します。表の定義情報，トリガ定義情報，又はストアドプロシジャの情報を搬出ファイルに出力することを**搬出**といい，搬出ファイルの内容を HiRDB システムに入力することを**搬入**といいます。また，搬出ファイルの定義情報を基にして，定義系 SQL の生成もします。

ディクショナリ搬出入ユティリティ（pdexp）の概要（表定義情報，トリガ定義情報，又はストアドプロシジャ情報の搬出入）を図「ディクショナリ搬出入ユティリティ（pdexp）の概要（表定義情報，トリガ定義情報，又はストアドプロシジャ情報の搬出入）」に，（定義系 SQL の生成）を図「ディクショナリ搬出入ユティリティ（pdexp）の概要（定義系 SQL の生成）」に示します。

表定義情報，トリガ定義情報，又はストアドプロシジャ情報の搬出入は次のような場合に使用します。

- 同一の表，トリガ，又はストアドプロシジャを，複数の HiRDB で使用する場合
- テスト用システムから本番用システムへ移行する場合
- システムの再構築などをする場合，いったん既存の表，トリガ，又はストアドプロシジャを搬出し，その後搬入するとき

定義系 SQL の生成は次のような場合に使用します。

- 既に定義されている表，トリガ，又はストアドプロシジャの定義系 SQL を，別の表，トリガ，又はストアドプロシジャを定義するために使用（流用）したいとき

図 9–1　ディクショナリ搬出入ユティリティ（pdexp）の概要（表定義情報，トリガ定義情報，又はストアドプロシジャ情報の搬出入）

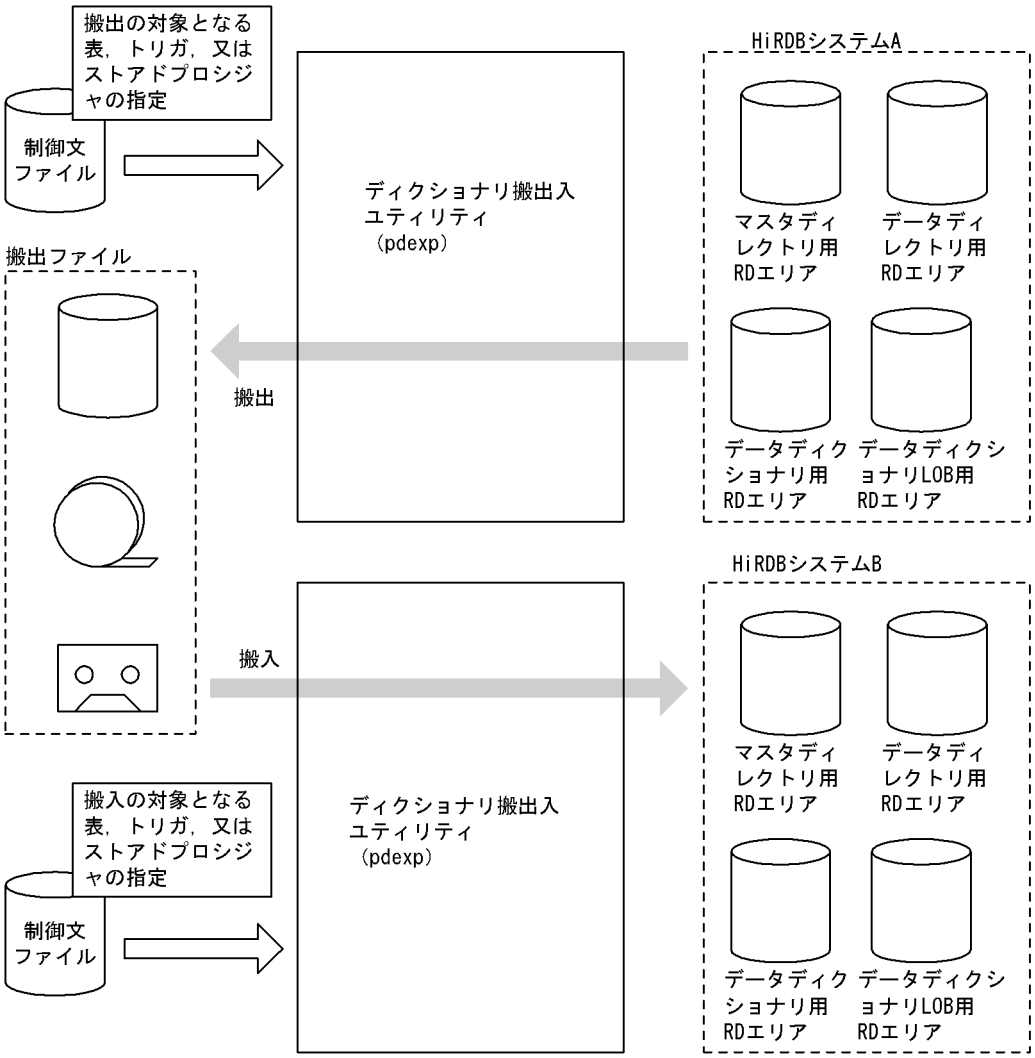
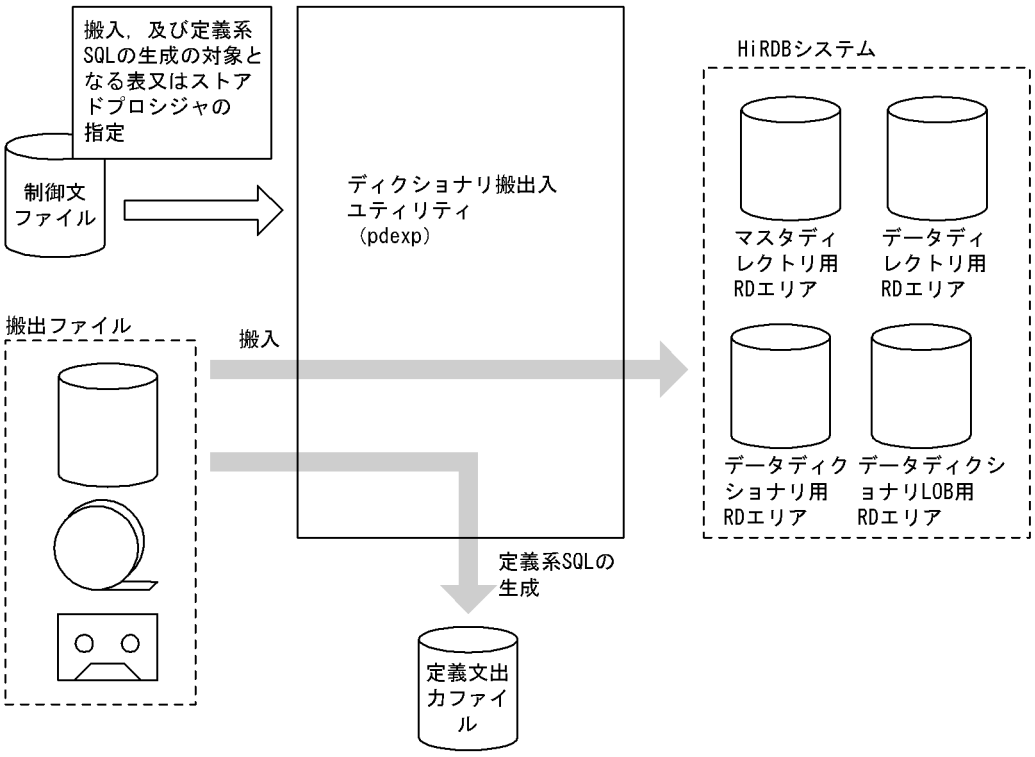


図 9–2　ディクショナリ搬出入ユティリティ（pdexp）の概要（定義系 SQL の生成）



9.1.1　表定義情報の搬出入

表定義情報の搬出入では表定義情報を移行します。移行後も，表定義情報は搬出元のディクショナリ表に残ります。搬出入の対象となる表定義情報を，次に示します。

- 実表の表定義情報
- インデクスの定義情報
- ビュー表の表定義情報
- 実表及びビュー表のコメント情報

表定義情報の搬出入では情報だけを移行するため，表のデータを移行する場合はデータベース再編成ユティリティ（pdrorg）を使用してください。また，搬出ファイルの定義情報を基に，定義系 SQL を生成できます。

参照制約が定義されている表を搬出入する場合，参照関係を考慮してユーザが搬出順序を決めなくても，搬出時に HiRDB が自動的に搬出順序を決定します。ただし，HiRDB が自動的に搬出順序を決定する方式の場合，メモリ使用量が増えます。自動決定にするか，ユーザ指定にするかは，オプションで指定できます。

9.1.2　トリガ定義情報の搬出入

トリガ定義情報の搬出入では，トリガ定義の情報を移行します。移行後も，トリガ定義情報は搬出元のディクショナリ表に残ります。また，搬出ファイルの定義情報を基に，定義系 SQL を生成できます。

9.1.3　ストアドプロシジャ情報の搬出入

ストアドプロシジャ情報の搬出入では，ストアドプロシジャの定義ソース文を移行します。移行後も，ストアドプロシジャ情報は搬出元のディクショナリ表に残ります。

搬出ではストアドプロシジャの定義ソース文を搬出し，搬入ではストアドプロシジャの定義ソース文を基にして HiRDB がオブジェクトを作成します。

また，搬出ファイルの定義情報を基に，定義系 SQL を生成できます。

参考

パブリックプロシジャの定義者情報は，パブリックプロシジャの搬入先でも引き継がれます。

9.1.4　定義系 SQL の生成

搬出ファイルの定義情報を基に定義系 SQL を生成し，ファイルに出力します。生成される定義系 SQL を次に示します。

種別	インデクス定義	コメント定義	生成される定義系 SQL
実表	あり	あり	CREATE TABLE CREATE INDEX COMMENT
		なし	CREATE TABLE CREATE INDEX
	なし	あり	CREATE TABLE COMMENT
		なし	CREATE TABLE
ビュー表	−	あり	CREATE VIEW COMMENT
		なし	CREATE VIEW
トリガ	−	−	CREATE TRIGGER
ストアドプロシジャ	−	−	CREATE PROCEDURE

（凡例）−：該当しません。

9.1.5　実行者

pdexp の機能を実行する場合に必要となる権限を次に示します。

- 搬出，及び定義系 SQL の生成をする場合
  DBA 権限所有者，又は表の所有者が実行できます。
- 搬入をする場合
  DBA 権限所有者が実行できます。

9.2　コマンドの形式

9.2.1　形式

表定義情報，トリガ定義情報，又はストアドプロシジャ情報の搬出入の場合

```
pdexp  {-e 搬出ファイル名 | -i 搬出ファイル名}
       〔-v〕
       〔-f 制御文ファイル名〕  〔-u 認可識別子〕
       〔-W 実行監視時間〕
```

定義系 SQL の生成の場合

```
pdexp -i 搬出ファイル名 〔-f 制御文ファイル名〕

      -o 定義系SQL出力ファイル名 〔-a〕  〔-u 認可識別子〕
        〔-W 実行監視時間〕
```

9.2.2　オプション

- **-e 搬出ファイル名**　～＜〔識別子：〕パス名＞((1023 バイト以内))

表定義情報，トリガ定義情報，又はストアドプロシジャ情報を搬出する場合：

搬出ファイルの名称を指定します。

定義系 SQL を生成する場合：

生成したい表定義情報，トリガ定義情報，又はストアドプロシジャ情報を搬出した搬出ファイルの名称を指定します。

搬出ファイルがあるホストの名称と絶対パス名をコロン（：）で結んで指定します。識別子を省略した場合は，ディクショナリ搬出入ユティリティを実行したサーバマシン（pdexp コマンドを入力したサーバマシン）のホスト名が仮定されます。指定した搬出ファイルがない場合は，HiRDB の起動者のものとして新規に作成します。

- **-i 搬出ファイル名**　～＜〔識別子：〕パス名＞((1023 バイト以内))

表定義情報，トリガ定義情報，ストアドプロシジャ情報を搬入する場合，又は定義系 SQL を生成する場合に，搬出ファイルの名称を指定します。

搬出ファイルがあるホストの名称と絶対パス名をコロン（：）で結んで指定します。識別子を省略した場合は，ディクショナリ搬出入ユティリティを実行したサーバマシン（pdexp コマンドを入力したサーバマシン）のホスト名が仮定されます。

- **-v**

参照制約が定義されている表の搬出入時，HiRDB が自動的に搬出順序を解析することによって増大するメモリ使用量を抑えたい場合に指定します。この場合，制御文ファイルに指定した表の順序で搬出します。

＜注意＞

このオプションを指定して参照制約が定義されている表を搬出する場合，制御文で被参照表を先に指定してください。搬入時には，搬出順に表が搬入されます。なお，被参照表が定義されていない場合に参照表を搬入すると，CREATE TABLE の SQL エラーとなります。この場合，途中で搬入処理が終了します。

- **-f 制御文ファイル名**　～＜パス名＞((1023 バイト以内))

ディクショナリ搬出入ユティリティの制御文があるファイルの名称を指定します。

制御文ファイルは，ディクショナリ搬出入ユティリティを実行するサーバマシン（pdexp コマンドを入力するサーバマシン）に作成してください。搬入時，搬出ファイル中のすべての表定義情報，トリガ定義情報，又はストアドプロシジャ情報を対象とする場合は，指定を省略できます。また，定義系 SQL 生成時，搬出ファイル中のすべての定義情報を対象とする場合も，指定を省略できます。

＜制御文についての規則＞

1. 搬出入で指定できる表，トリガ，又はストアドプロシジャは，最大 10,000 個です。
2. 1 行に複数の指定はできません。
3. 一つの指定は複数行にわたってはなりません。
4. 空行は無視されます。
5. スキーマ名に MASTER，ALL，及び HiRDB は指定できません（トリガ定義情報の搬出の場合は PUBLIC も指定できません）。
6. 表の指定（-t），トリガの指定（-g），及びストアドプロシジャの指定（-p）は混在して指定できません。
7. 一つの制御文ファイル中で実表とビュー表を指定する場合は，実表を先に指定してください。
8. 制御文ファイル中にコメントは記述できません。

＜表定義情報の搬出入，又は定義系 SQL の生成の場合＞

```
 -t スキーマ名.表識別子 〔-w〕
〔-t スキーマ名.表識別子 〔-w〕〕
```

＜トリガ定義情報の搬出入，又は定義系 SQL の生成の場合＞

```
 -g スキーマ名.トリガ識別子 〔-w〕
〔-g スキーマ名.トリガ識別子 〔-w〕〕
```

＜ストアドプロシジャ情報の搬出入，又は定義系 SQL の生成の場合＞

```
 -p スキーマ名.ルーチン識別子
〔-p スキーマ名.ルーチン識別子〕
```

-t スキーマ名.表識別子

搬出入する表，又は定義系 SQL を生成する表を指定します。

同一の表は重複して指定できません。

パブリックビューを搬出入する場合，スキーマ名に PUBLIC を指定してください。

スキーマ名　～＜識別子＞((1～30))

スキーマ名を指定します。スキーマ名を引用符（"）で囲んだ場合は，英大文字と小文字が区別されますが，引用符で囲まない場合はすべて英大文字として扱われます。

表識別子　～＜識別子＞((1～30))

表識別子を指定します。表識別子を引用符（"）で囲んだ場合は，英大文字と小文字が区別されますが，引用符で囲まない場合はすべて英大文字として扱われます。また，表識別子に空白を含む場合も引用符で囲んでください。

-w

WITH PROGRAM 指定なしの参照表の表定義を，WITH PROGRAM 指定ありの参照表の表定義にして搬入する場合に指定します。-w オプションを省略した場合，及び参照表以外の表定義の場合，WITH PROGRAM 指定なしの表定義として搬入します。なお，搬出時には，このオプションの指定は無効となります。

-g スキーマ名.トリガ識別子〔-w〕

搬出入するトリガ，又は定義系 SQL を生成するトリガを指定します。

同一のトリガは重複して指定できません。

スキーマ名　～＜識別子＞((1～30))

スキーマ名を指定します。スキーマ名を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。

トリガ識別子　～＜識別子＞((1～30))

トリガ識別子を指定します。トリガ識別子を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，トリガ識別子に空白を含む場合も引用符で囲んでください。

-w

WITH PROGRAM 指定なしのトリガ定義を，WITH PROGRAM 指定ありのトリガ定義にして搬入する場合に指定します。-w オプションを指定し，WITH PROGRAM 指定ありのトリガ定義にして搬入した場合，そのトリガが使用する表に対する関数及び手続きの有効な SQL オブジェクトがあるときは，その SQL オブジェクトは無効になります。

＜一つの表に複数のトリガを定義している場合の注意事項＞

一つの表に複数のトリガを定義している場合，トリガはトリガ定義順に動作します。搬入時のトリガ定義順が搬出前の順番と異なる場合，搬出前のトリガ定義と異なる順番でトリガが動作するため，搬出前と搬入後とで結果が異なります。搬出前と搬入後とで同じトリガ動作をさせるためには，制御文にトリガ定義順にトリガ名を記述する必要があります。

トリガの定義順に制御文を作成する方法を次に示します。

特定の表に定義しているトリガすべてを搬出入する場合：

次の SQL を実行すると，表に定義されているトリガ所有者，トリガ識別子，及びトリガ定義時刻が，トリガ定義順に求められます。

```
SELECT TRIGGER_SCHEMA,TRIGGER_NAME,CREATE_TIME
  FROM "MASTER".SQL_TRIGGERS
  WHERE TABLE_SCHEMA='所有者' AND TABLE_NAME='表名称'
  ORDER BY CREATE_TIME WITHOUT LOCK NOWAIT;
```

制御文には，トリガ定義時刻の早い順にスキーマ名とトリガ識別子を記述します。

定義しているすべてのトリガを搬出入する場合：

次の SQL を実行すると，表に定義されているトリガ所有者，トリガ識別子，及びトリガ定義時刻が，トリガ定義順に求められます。

```
SELECT TRIGGER_SCHEMA,TRIGGER_NAME,CREATE_TIME
  FROM "MASTER".SQL_TRIGGERS
  ORDER BY CREATE_TIME WITHOUT LOCK NOWAIT;
```

制御文には，トリガ定義時刻の早い順にスキーマ名とトリガ識別子を記述します。

-p スキーマ名.ルーチン識別子

搬出入するストアドプロシジャ，又は定義系 SQL を生成するストアドプロシジャを指定します。

同一のストアドプロシジャは重複して指定できません。

スキーマ名　～＜識別子＞((1～30))

スキーマ名を指定します。スキーマ名を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。

パブリックプロシジャを搬出する場合はスキーマ名に PUBLIC を指定してください。

ルーチン識別子　～＜識別子＞((1～30))

ルーチン識別子を指定します。ルーチン識別子を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。また，ルーチン識別子に空白を含む場合も引用符で囲んでください。

- **-u 認可識別子**　～＜識別子＞((1～30))

ディクショナリ搬出入ユティリティを実行するユーザの認可識別子を指定します。このオプションを指定した場合，パスワード入力要求の応答付きメッセージが表示されます。そこで応答した認可識別子/パスワードで HiRDB システムに接続し，その認可識別子について HiRDB への CONNECT 権限及び DBA 権限があるかどうかがチェックされます。

このオプションは，パスワードを応答できない環境では指定しないでください。

省略した場合，次の方法で認可識別子/パスワードが仮定されます。

- ディクショナリ搬出入ユティリティ実行時の環境変数 PDUSER の設定値が仮定されます。パスワードの応答ができない環境で実行するときは，PDUSER を必ず設定してください。

- 環境変数 PDUSER が設定されていない場合，ログインウィンドウのユーザ名が仮定されます。パスワードについては，入力要求の応答付きメッセージが表示されるので，そこで応答してください。

認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

- **-o 定義系 SQL 出力ファイル名**　～＜パス名＞((1023 バイト以内))

生成した定義系 SQL を出力するファイルの名称を指定します。指定したファイルが既にある場合は上書きし，ない場合は新規に作成します。

- **-a**

定義系 SQL の生成と搬入を同時にする場合に指定します。-o オプションを指定しないで-a オプションを指定した場合は，搬入だけをします。

- **-W 実行監視時間**　～＜符号なし整数＞((0～3600))

ディクショナリ搬出入ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。

このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

オプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

9.3　規則及び注意事項

9.3.1　規則

1. ディクショナリ搬出入ユティリティは，HiRDB が稼働中のときだけ実行できます。ただし，HiRDB/シングルサーバの場合，シングルサーバが起動していなければなりません。また，HiRDB/パラレルサーバの場合は，次に示すサーバが起動していなければなりません。

   ＜表定義情報の搬出入の場合＞

   搬出時：フロントエンドサーバ，ディクショナリサーバ，及びすべてのバックエンドサーバ

   搬入時：フロントエンドサーバ，ディクショナリサーバ，及び対象となるバックエンドサーバ

   搬出時及び搬入時のマスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリア，オープン状態，かつ閉塞解除でなければなりません。

   ＜ストアドプロシジャ情報及びトリガ情報の搬出入の場合＞

   搬出時：フロントエンドサーバ，及びディクショナリサーバ

   搬入時：フロントエンドサーバ，及びディクショナリサーバ

   搬入時には，データディクショナリ LOB 用 RD エリアを定義しておく必要があります。

2. ディクショナリ搬出入ユティリティは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。

3. ディクショナリ搬出入ユティリティ実行後，パスワード入力要求の応答付きメッセージが表示された場合，パスワードを入力してください（入力したパスワードは表示されません）。パスワードに英小文字を含む場合は，「¥""」と「"¥"」で囲んでください。

4. ディクショナリ搬出入ユティリティを実行する場合の RD エリアの状態については，「コマンド実行時の RD エリアの状態」を参照してください。

5. 搬出時，指定した搬出ファイルがない場合，HiRDB の起動者のものとして新規に作成します。そのため，HiRDB の起動者にはファイルの作成権限が必要です。

6. 搬入時は，搬出処理で正常に作成した搬出ファイルを指定してください。また，HiRDB 管理者には，その搬出ファイルの参照権限が必要です。

7. HiRDB/パラレルサーバの場合，サーバと RD エリアの対応を搬出前システムと搬入後システムとで同じにしてください。

8. 異なるバイトオーダのシステム間，及び異なる文字体系のシステム間での搬出入はできません。

9. SQL 予約語削除機能を使用している場合，コマンド実行画面でクライアント環境定義 PDDELRSVWDFILE を設定してから，ディクショナリ搬出入ユティリティを実行してください。なお，PDDELRSVWDFILE の設定値の長さが 9 バイト以上の場合は，PDDELRSVWDFILE の設定が無視されます。また，次の条件を満たしていないと，搬入時に SQL の文法エラーになることがあります。

   - クライアント環境定義 PDDELRSVWDFILE の値が，表などを定義したときの値と同じである。

- 搬出側と搬入側の，システム共通定義 pd_delete_reserved_word_file オペランドの指定値が同じである。
- 搬出側と搬入側の，%PDDIR%¥conf¥pdrsvwd 以下のファイルが同じである。

10. 搬出時，搬出側システムで定義系 SQL を発行しないでください。

9.3.2 表定義情報，トリガ定義情報，及びストアドプロシジャ情報の搬出入についての規則

1. 搬出と搬入は，１回の実行で同時に指定できません。
2. トリガ定義情報，又はストアドプロシジャ情報の搬出順序は，制御文ファイルでの-g，又は-p オプションの指定順になります。搬入時の搬入順序は，-g，又は-p オプションの指定に関係なく，搬出時の搬出順序と同じになります。
3. 搬出入の処理は，一つの表ごと，一つのトリガごと，又は一つのストアドプロシジャごとに処理が完結します。
4. 搬入先に同一の表（認可識別子．表識別子が同じ表），トリガ（認可識別子．トリガ識別子），又はストアドプロシジャ（認可識別子．ルーチン識別子が同じストアドプロシジャ）がある場合，その表，トリガ，又はストアドプロシジャは搬入エラーとなります。エラーになった表以外の搬出処理は続行されます。
5. 搬入前には，搬入先システムを初期設定し，搬入する表，トリガ，又はストアドプロシジャのスキーマを定義しておいてください。
6. ディクショナリ搬出入ユティリティは，搬出元の表定義をそのまま搬入先に移行します。そのため，対象となる表の格納 RD エリアは，搬出元と同じ名称のものを搬入先にも定義しておく必要があります。
7. 搬入する表，トリガ，又はストアドプロシジャの所有者には，表，トリガ，又はストアドプロシジャを格納する RD エリアの利用権限が必要です。
8. ストアドプロシジャの搬入時，ストアドプロシジャが使用する表，手続き，関数，抽象データ型は搬入先に定義しておく必要があります。該当する表，手続き，関数，抽象データ型が定義されていない場合は，ストアドプロシジャの搬入はできません。
9. プラグインが提供する抽象データ型を使用する表，ストアドプロシジャの搬入時，表，ストアドプロシジャが使用するプラグインは搬入先システムに登録しておく必要があります。該当するプラグインが登録されていない場合は，表，ストアドプロシジャの搬入はできません。
10. 制御文ファイルに，-t オプション，-g オプション，及び-p オプションは混在して指定できません。
11. 搬入時に，2,000,000 バイトを超える SQL 文を実行する場合（表，トリガ，又はストアドプロシジャを定義する SQL 文の長さが 2,000,000 バイトを超える場合），搬入はできません。
12. 次に示す表の定義情報，及びストアドプロシジャの情報は，搬出できません。
    - ユーザが定義した抽象データ型を含む表定義（CREATE TABLE）
    - 部分構造インデクス定義（CREATE INDEX）

- 関数定義（CREATE FUNCTION，及びシステム定義スカラ関数）
- 抽象データ型定義（CREATE TYPE）
- 抽象データ型内で宣言されているストアドプロシジャ（CREATE TYPE）
- TABLE に指定した表にユーザが定義した抽象データ型の列を持つ注釈付加（COMMENT）
- COLUMN に指定した列がユーザが定義した抽象データ型である注釈付加（COMMENT）

13. HiRDB/パラレルサーバで共用表を搬入する場合，搬入先の環境の共用表，及び共用表に定義した共用インデクスを格納する RD エリアは，すべて共用 RD エリアである必要があります。共用 RD エリアでない場合，SQL エラーとなります。また，共用表でない表を搬入する場合，及び表の格納 RD エリアが共用 RD エリアの場合，SQL エラーとなります。共用表と搬入先の RD エリアとの関係を次に示します。なお，HiRDB/シングルサーバの共用表の場合，共用 RD エリアを作成しないため，該当しません。

搬入元の表の種別	搬入先の RD エリアの定義		定義の実行可否		エラー後の pdexp の処理※
	表格納用 RD エリア	表に定義したインデクスの格納 RD エリア	表	インデクス	
共用表	共用 RD エリア	共用 RD エリア	○	○	該当しません。
	共用 RD エリア	非共用 RD エリア	○	△	処理を中断し，表定義は無効となります。
	非共用 RD エリア	共用 RD エリア	△	×	処理を中断します。
	非共用 RD エリア	非共用 RD エリア	△	×	
非共用表	共用 RD エリア	共用 RD エリア	△	×	
	共用 RD エリア	非共用 RD エリア	△	×	
	非共用 RD エリア	共用 RD エリア	○	△	処理を中断し，表定義は無効となります。
	非共用 RD エリア	非共用 RD エリア	○	○	該当しません。

（凡例）

○：定義をします。

△：定義をしますが，SQL エラーとなります。

×：定義をしません。

注※

pdexp は，定義系 SQL の内部 COMMIT を抑止し，処理途中でエラーが発生した場合はロールバックします。

14. 参照表定義時に内部的に作成されるトリガは搬出できません。搬出時にそのトリガを指定した場合，KFPX28504-W メッセージが出力されます（該当するトリガの処理はスキップし，次のトリガの搬出をします）。

9.3.3　定義系 SQL の生成についての規則

1. 搬出時（-e オプション）に，定義系 SQL の生成（-o オプション）を指定した場合，定義系 SQL の生成はしません。
2. 定義系 SQL を生成する場合，SQL 文の生成順序は搬出順となります。
3. 次に示す表，手続き，及び関数の定義系 SQL は生成されません。
   - ユーザが定義した抽象データ型を含む表定義（CREATE TABLE）
   - 抽象データ型内で宣言されている手続き及び関数（CREATE TYPE）
   - 部分構造インデクス定義（CREATE INDEX）
   - 関数定義（CREATE FUNCTION）
4. 生成された定義系 SQL には，データベース定義ユティリティの入力ファイルとして利用できるように，行の最後にセミコロン（；）が付きます。ただし，CREATE PROCEDURE，及び CREATE TRIGGER の場合は end_proc;が付きます。
5. CREATE PROCEDURE，及び CREATE TRIGGER を生成する場合，SQL 最適化オプション，及び SQL 拡張最適化オプションの指定値は，数値に変換された形になります。
6. 表の定義系 SQL を生成する場合，その表に外部キー，又は検査制約が定義されているときは，表の定義系 SQL と同時に，外部キー，及び検査制約の定義系 SQL も生成します。この外部キー，及び検査制約の定義系 SQL を生成する場合，外部キー，及び検査制約の制約名の定義位置は，クライアント環境定義の PDCNSTRNTNAME とシステム定義の pd_constraint_name オペランドの指定値によって決まります。PDCNSTRNTNAME についてはマニュアル「HiRDB UAP 開発ガイド」を，pd_constraint_name オペランドについてはマニュアル「HiRDB システム定義」を参照してください。
7. 生成された定義系 SQL の改行条件を次に示します。

生成された定義系 SQL	改行条件
CREATE TABLE	• CREATE から表識別子まで • 列定義ごと • 表格納用 RD エリアごと • 表オプションごと • 外部キーごと • 検査制約定義ごと • WITH PROGRAM • 1 行が 80 バイトを超える場合は，80 バイトごと※
CREATE INDEX	• CREATE からインデクス識別子まで • USING からインデクス識別子まで • ON から表識別子まで • 列定義ごと • インデクス格納用 RD エリアごと • インデクスオプションごと • 1 行が 80 バイトを超える場合は，80 バイトごと※

生成された定義系 SQL	改行条件
COMMENT	1 行が 80 バイトを超える場合は，80 バイトごと※
CREATE VIEW	• CREATE から AS まで • 1 行が 80 バイトを超える場合は，80 バイトごと※
CREATE PROCEDURE	• CREATE からパラメタモード指定まで • 定義ソース文中にある改行記号 • SQL コンパイルオプション • SQL 手続き文 • 1 行が 80 バイトを超える場合は，80 バイトごと※
CREATE TRIGGER	• CREATE からトリガ動作 • 定義ソース文中にある改行記号 • SQL コンパイルオプション • WITH PROGRAM • 1 行が 80 バイトを超える場合は，80 バイトごと※

注※

システム共通定義 pdsys の pd_sql_simple_comment_use オペランドの値による動作を次の表に示します。

項番	システム共通定義 pdsys の pd_sql_simple_comment_use オペランドの値	動作
1	N	改行する
2	Y	改行しない

単純注釈を使用するには，pd_sql_simple_comment_use オペランドを Y にする必要があります。搬出した SQL 文中の単純注釈の有無と，pd_sql_simple_comment_use オペランドに対する指定が合っていない場合，搬入時に SQL 構文エラーとなります。

これらの組み合わせについて次に示します。

項番	搬出した SQL 文中の単純注釈有無	搬入時の pd_sql_simple_comment_use の値	
		Y	N
1	なし	○	○
2	あり	○	×

（凡例）

○：SQL 構文エラーにはなりません。

×：SQL 構文エラー（KFPA11105-E）となります。

8. 文字集合 UTF-16 を指定した CHAR 型の列を，キーレンジ分割又はマトリクス分割（境界値指定のキーレンジ分割）の分割キーとして指定した表の場合，分割キーの長さが列の定義長を超えるときは，その表の定義系 SQL は生成できません。

9. DEFAULT 句の ON ROW EXISTS 指定がある表の定義系 SQL を生成する場合は，ON ROW EXISTS 指定なしの表を生成します。

9.3.4　参照制約を定義した表の搬出入の規則について

(1)　搬出

1. 参照表，及び被参照表を搬出すると，無条件に主キー，及び外部キーの定義も搬出します。
2. 表定義情報は，制御文ファイルに指定した順序に関係なく，HiRDB が搬出順序を決定して被参照表を先に搬出します。-v オプションを指定した場合は，制御文ファイルに指定した順序で搬出します。
3. トリガ定義情報，又はストアドプロシジャ情報は，制御文ファイルに指定した順序で搬出します。

(2)　搬入

1. 参照表，及び被参照表を搬入すると，無条件に主キー，及び外部キーの定義も搬入します。
2. 搬入先に同一名称の制約がある場合，CREATE TABLE の SQL エラーとなります。この場合，次の表の処理をしないで，搬入処理を終了します。
3. 主キー，並びにクラスタキーのインデクス番号，及び表番号は，搬入先で新たに割り当てられます。
4. 主キーのインデクス識別子は，搬入時の表の表番号を基に決まります。したがって，搬出元と同じになるとは限りません。
5. 参照表を搬入する場合の，被参照表を使用する SQL オブジェクトの扱いを次に示します。

表定義時の参照動作	-w オプション	被参照表を使用する SQL オブジェクトの扱い	備考
CASCADE	指定あり	参照表と被参照表の整合性を保つために，内部的にトリガが定義されます。このとき，SQL オブジェクトが作成されるため，被参照表を使用する関数，手続き，及びトリガの SQL オブジェクトが無効になります。	被参照表を使用する関数，手続き，及びトリガの SQL オブジェクトが既にある場合，その SQL オブジェクトは無効な状態となっています。そのため，参照表を搬入した後に，SQL オブジェクトを再作成する必要があります。
	指定なし	被参照表を使用する関数，手続き，及びトリガの SQL オブジェクトが無効になります。そのため，その SQL オブジェクトを再作成しないと，参照表と被参照表の整合性が保たれないため，CREATE TABLE の SQL エラーとなります。	−
RESTRICT	指定あり	被参照表を使用する関数，手続き，及びトリガの SQL オブジェクトが有効な場合，被参照表の行を更新又は削除してしまうため，整合性が保てなくなります。そのため，その SQL オブジェクトは無効となります。	被参照表を使用する関数，手続き，及びトリガの SQL オブジェクトが既にある場合，その SQL オブジェクトは無効な状態となっています。そのため，参照表を搬入した後に，SQL オブジェクトを再作成する必要があります。

表定義時の参照動作	-w オプション	被参照表を使用する SQL オブジェクトの扱い	備考
	指定なし	被参照表を使用する関数，手続き，及びトリガの SQL オブジェクトが無効になります。そのため，その SQL オブジェクトを再作成しないと，参照表と被参照表の整合性が保たれないため，CREATE TABLE の SQL エラーとなります。	−

（凡例）−：特にありません。

9.3.5　注意事項

1. ディクショナリ搬出入ユティリティの結果は，ユティリティ実行時のリターンコード，又はエラーメッセージの有無で確認できます。ユティリティ実行時のリターンコードを次に示します。

   0：正常終了です。

   4：警告レベルのエラーです。

   8：正常に終了したものと，一部処理をスキップしたものとがあります。

   12：正常な処理は１件も完了しないで処理を終了しました。

2. 表のアクセス権限は，搬入後に定義系 SQL の GRANT で設定してください。

3. ビュー表を搬入する場合，ビュー表の搬入前に，基となる実表を搬入又は定義し，アクセス権限を設定しておいてください。

4. ディクショナリ搬出入ユティリティは，同時に複数実行できません。既に処理を実行している場合は，その処理が終了してから次の処理を実行してください。

5. 搬入処理は，搬入先システムで次に示す定義系 SQL を実行するのと同様の機能です。各 SQL の仕様を確認してから搬入を実行してください。

   ＜表定義情報の搬出入の場合＞

   - ALTER TABLE
   - COMMENT
   - CREATE INDEX
   - CREATE TABLE
   - CREATE VIEW

   ＜トリガ定義情報の搬出入の場合＞

   - CREATE TRIGGER

   ＜ストアドプロシジャ情報の搬出入の場合＞

   - CREATE PROCEDURE

   SQL の仕様については，マニュアル「HiRDB SQL リファレンス」を参照してください。

6. ディクショナリ搬出入ユティリティの強制終了，又はサーバの停止などによって，搬出処理が中断された場合，KFPX28402-I メッセージが出力されていても，搬出ファイルに表定義情報，トリガ定義情報，又はストアドプロシジャ情報が搬出されていないことがあります。この場合には，再度ディクショナリ搬出入ユティリティを実行し，搬出処理を正常に終了させてください。
7. pdexp を実行する場合，搬出ファイルと定義系 SQL 出力ファイルのアクセス権限を，HiRDB 管理者に対して与えておく必要があります。
8. pdexp を実行する前に，OS の環境変数 PDUSER と HiRDB.ini の PDUSER の値を一致させてください。一致していない場合，pdexp の実行時にエラーになることがあります。
9. ユーザが定義した抽象データ型を使用する表を搬出しようとした場合，pdexp はリターンコード 12 で異常終了します。この場合，該当する表を制御文ファイルから取り除き，再度ディクショナリ搬出入ユティリティを実行し，搬出処理を正常に終了させてください。制御文ファイルにユーザが定義した抽象データ型を使用する表が複数指定されている場合は，該当する表をすべて取り除いてください。

   ユーザが定義した抽象データ型を使用する表の一覧を取得する SQL の例を次に示します。

```
SELECT X.TABLE_SCHEMA, X.TABLE_NAME
  FROM MASTER.SQL_TABLES X
  WHERE X.TABLE_SCHEMA='スキーマ名'
  AND EXISTS (
    SELECT 1 FROM MASTER.SQL_COLUMNS Y
    WHERE Y.TABLE_SCHEMA = X.TABLE_SCHEMA
      AND Y.TABLE_NAME = X.TABLE_NAME
      AND Y.UDT_OWNER != 'MASTER'
) WITHOUT LOCK NOWAIT;
```

   また，ユーザが定義した抽象データ型を使用しない表の一覧を取得する SQL の例を次に示します。

```
SELECT X.TABLE_SCHEMA, X.TABLE_NAME
  FROM MASTER.SQL_TABLES X
  WHERE X.TABLE_SCHEMA='スキーマ名'
  AND NOT EXISTS (
    SELECT 1 FROM MASTER.SQL_COLUMNS Y
    WHERE Y.TABLE_SCHEMA = X.TABLE_SCHEMA
      AND Y.TABLE_NAME = X.TABLE_NAME
      AND Y.UDT_OWNER != 'MASTER'
) WITHOUT LOCK NOWAIT;
```

9.3.6 搬入時に作成される定義系 SQL が 2,000,000 バイトを超えた場合の修正例

pdexp で搬入をする場合，定義系 SQL が作成され，その定義系 SQL が実行されます。その際，次に示す原因によって，作成される SQL 文が，SQL 文の最大長 2,000,000 バイトを超えることがあります。

- 次に示す構文，及び識別子が省略された表を pdexp で搬入する場合

表 9–1　定義系 SQL 文に記述される構文及び識別子の，デフォルト値及び引用符

項番	構文及び識別子	デフォルト値及び引用符
1	列制約定義	PARTIAL
2	PCTFREE（未使用領域の比率）	30
3	PCTFREE（セグメント内の空きページ比率）	10
4	削除動作	RESTRICT
5	更新動作	RESTRICT
6	制約名定義	制約定義時のタイムスタンプ
7	識別子	大文字，小文字を区別するため引用符が付与されます
8	バージョン 09-50 より前の CREATE TABLE での SEGMENT REUSE	SEGMENT REUSE NO
9	バージョン 09-50 以降の CREATE TABLE での SEGMENT REUSE	SEGMENT REUSE
10	プラグインが提供する抽象データ型，及びインデクス型のスキーマ名	"MASTER"

デフォルト値の構文や識別子の引用符が定義系 SQL 文に記述されるため，2,000,000 バイトを超えるおそれがあります。

定義系 SQL が 2,000,000 バイトを超えた場合の修正例を次に示します。ここでは，CREATE TABLE を例に説明します。

＜ CREATE TABLE が 2,000,000 バイトを超えた場合の例＞

CREATE TABLE が 2,000,000 バイトを超えた場合，2,000,000 バイト以内に収まるように，列 ID の大きいものから列定義部分を削除した状態で CREATE TABLE が作成されます。このとき，削除された列定義部分を追加する必要があるため，削除された列定義部分を追加する ALTER TABLE が作成されます。例を次に示します。

```
CREATE TABLE ”root”.”T1”
(”C1” INTEGER NOT NULL,
”C2” CHAR(100),
      :
      :
”C98” VARCHAR(100)
) IN ((”RDUSER01”) ”C100” < 1500
(”RDUSER02”))
PCTFREE=(30,10)
FOREIGN KEY (”C1”,”C2”) REFERENCES ”root”.”T2”
  ON DELETE RESTRICT ON UPDATE RESTRICT CONSTRAINT CNST01
;   .....................................................1
ALTER TABLE ”root”.”T1”
ADD ”C99” INTEGER
;   .....................................................2
ALTER TABLE ”root”.”T1”
ADD ”C100” INTEGER NOT NULL
;   .....................................................2
```

［説明］

1. 列 C99，及び列 C100 が除かれた状態の CREATE TABLE が作成されます。
2. CREATE TABLE から削除された列 C99，及び列 C100 を追加する ALTER TABLE が作成されます。

［ALTER TABLE の列定義部分に追加されたことによって発生する SQL エラー］

2.で追加される列が，次のどれかに該当する場合，SQL エラーとなります。

- 分割キー
- クラスタキー
- 参照制約で指定した列
- 検査制約の探索条件で指定した列

この例では，分割キーである列 C100 が追加されているため，作成される ALTER TABLE は SQL エラーとなります。

＜ SQL エラーを回避する方法＞

SQL エラーとなる場合，次の手順で回避できます。

1. 搬入時に-o オプションを指定し，定義系 SQL を生成します（定義系 SQL をファイルに出力します）。
2. 1.で出力した定義系 SQL の，先頭の ALTER TABLE から，該当する ALTER TABLE（SQL エラーとなる列定義）までを，CREATE TABLE の列定義に移動します。
3. 1.で出力した定義系 SQL の，CREATE TABLE の省略時仮定値の部分を削除し，2,000,000 バイト以内に収まるようにします。
4. 編集した定義系 SQL を，pddef で実行します。

上記の定義系 SQL の編集例を次に示します。

```
CREATE TABLE "root"."T1"
("C1" INTEGER NOT NULL,
"C2" CHAR(100),
     :
     :
"C98" VARCHAR(100),
"C99" INTEGER,    ..............上記手順2.で追加
"C100" INTEGER NOT NULL   .....上記手順2.で追加
) IN (("RDUSER01") "C100" < 1500
("RDUSER02"))
PCTFREE=(30,10)
FOREIGN KEY ("C1","C2") REFERENCES "root"."T2"
  CONSTRAINT CNST01   .........上記手順の3.で一部分を削除
;
```

9.3.7　BOM を持つファイルの使用

pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdexp の入力ファイルには，BOM を持つファイルを使用できることがあります。pdexp での BOM を持つファイルの使用可否を次

の表に示します。なお，pdexp の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdexp で出力されるファイルには BOM は付きません。

表 9–2　pdexp での BOM を持つファイルの使用可否（UTF-8 の場合）

オプション	入力ファイル	BOM を持つファイルの使用
-l	搬出ファイル	×
-f	制御文ファイル	○

（凡例）

○：使用できます。

×：使用できません。

9.3.8　単純注釈を使用する場合の搬出入

単純注釈を含む定義も搬出入できます。また，定義系 SQL も生成できます。

SQL 文中に単純注釈が含まれるかどうか，及び含まれる箇所を次に示します。

項番	SQL 文	単純注釈	単純注釈が含まれる箇所
1	CREATE TABLE	○	検査制約定義の中
2	CREATE INDEX	×	–
3	COMMENT	×	–
4	CREATE PROCEDURE	○	CREATE 以降のすべての箇所
5	CREATE TRIGGER	○	CREATE 以降のすべての箇所
6	CREATE VIEW	○	CREATE 以降のすべての箇所

（凡例）

○：単純注釈が含まれます。

×：単純注釈が含まれません。

–：該当しません。

なお，単純注釈を含む場合で，次のユティリティ及び PP の実行時，システム定義の pd_sql_simple_comment_use オペランドが Y でない場合は，エラーになります。

- データベース定義ユティリティ（pddef）
- データベース作成ユティリティ（pdload）
- HiRDB SQL Executer

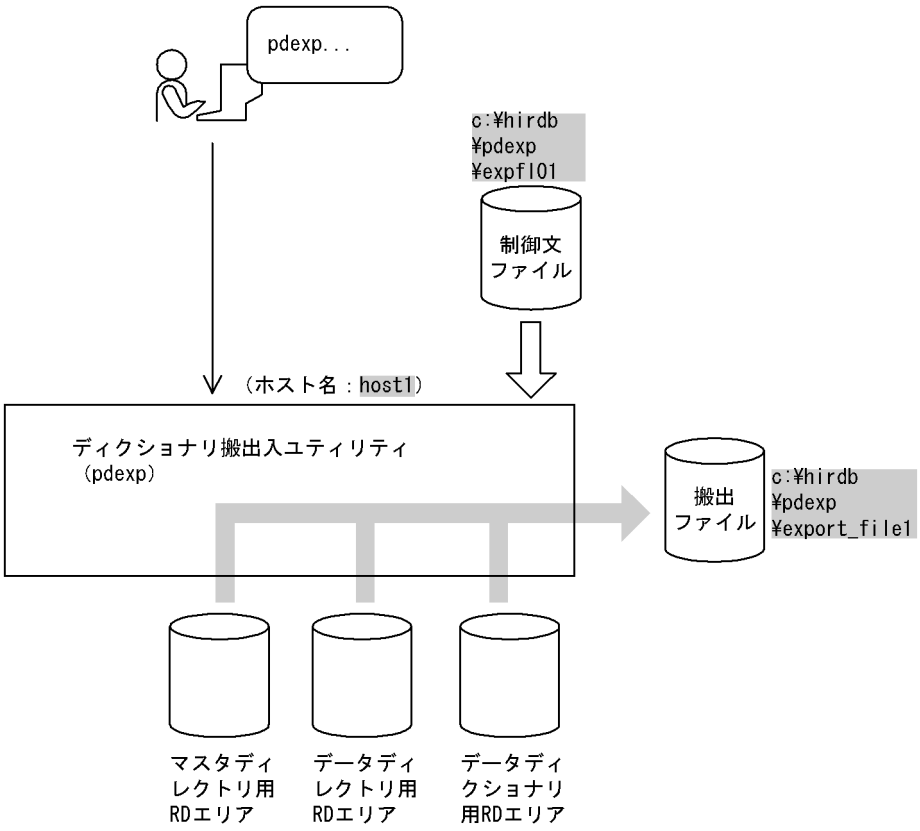
9.4　使用例

ディクショナリ搬出入ユティリティの使用例を，例 1〜例 4 に示します。

例 1

次に示す表の定義情報を搬出します（スキーマ名：USER01）。
・TABLE1
・TABLE2

[概要]



[コマンド実行例]

```
pdexp -e host1:c:¥hirdb¥pdexp¥export_file1
      -f c:¥hirdb¥pdexp¥expfl01
```

[説明]

搬出ファイルの名称　**host1:c:¥hirdb¥pdexp¥export_file1**

制御文ファイルの名称　**c:¥hirdb¥pdexp¥expfl01**

制御文ファイル（c:¥hirdb¥pdexp¥expfl01）の内容

```
-t USER01.TABLE1
-t USER01.TABLE2
```

[説明]

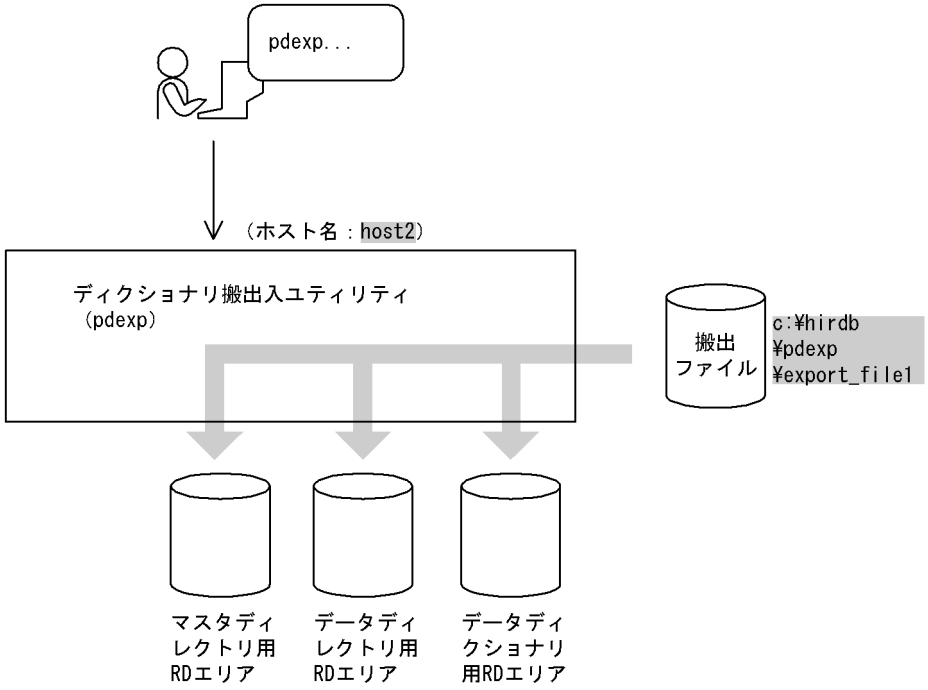
表定義情報を搬出する表の名称　**USER01.TABLE1**

表定義情報を搬出する表の名称　**USER01.TABLE2**

例 2

搬出ファイルのすべての表定義情報を搬入します。

[概要]



[コマンド実行例]

```
pdexp -i host2:c:¥hirdb¥pdexp¥export_file1
```

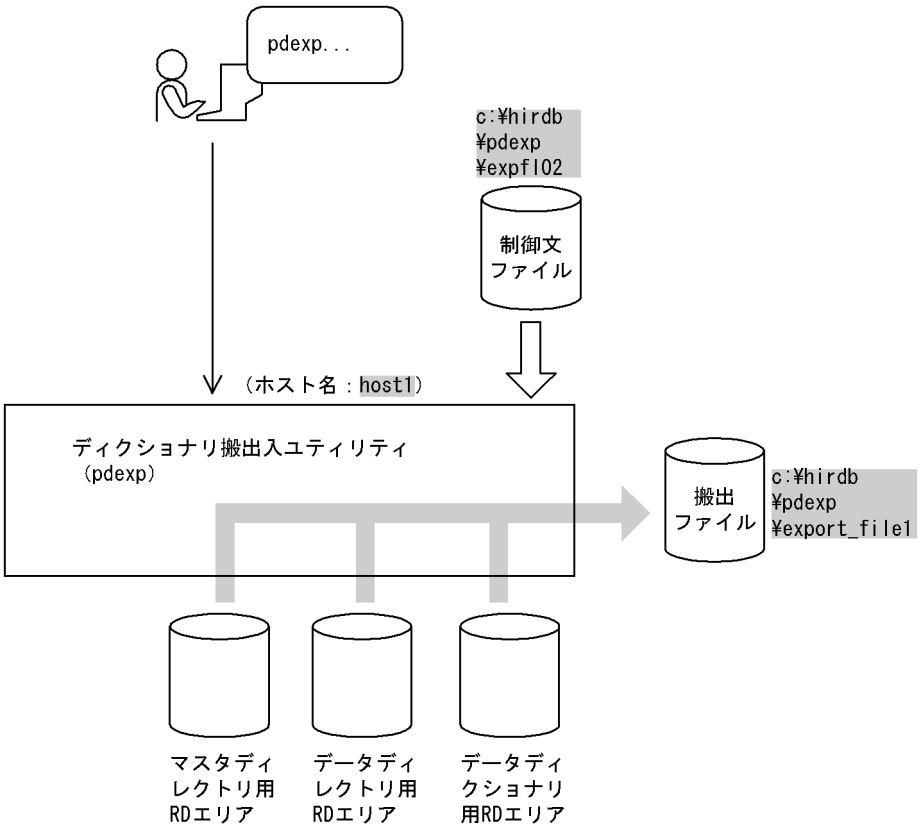
[説明]

搬出ファイルの名称　**c:¥hirdb¥pdexp¥export_file1**

例 3

次に示すストアドプロシジャの定義情報を搬出します（スキーマ名：USER01）。
・PROC1
・PROC2

[概要]



[コマンド実行例]

```
pdexp -e host1:c:¥hirdb¥pdexp¥export_file1
      -f c:¥hirdb¥pdexp¥expfl02
```

[説明]

搬出ファイルの名称　**host1:c:¥hirdb¥pdexp¥export_file1**

制御文ファイルの名称　**c:¥hirdb¥pdexp¥expfl01**

制御文ファイル（c:¥hirdb¥pdexp¥expfl01）の内容

```
-p USER01.PROC1
-p USER01.PROC2
```

[説明]

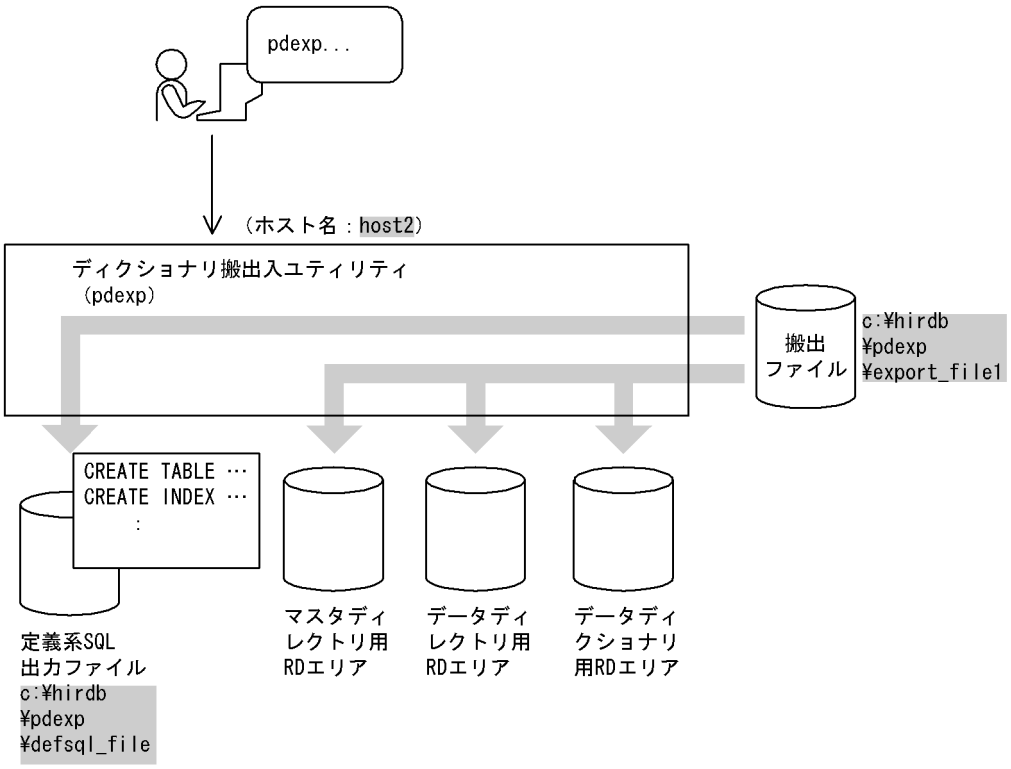
ストアドプロシジャの定義情報を搬出するルーチン名称　**USER01.PROC1**

ストアドプロシジャの定義情報を搬出するルーチン名称　**USER01.PROC2**

例 4

搬出ファイルのすべての表定義情報を搬入し，同時に搬出ファイルの定義情報を基に定義系SQLを生成します。

[概要]



[コマンド実行例]

```
pdexp -i host2:c:¥hirdb¥pdexp¥export_file1   .......1
      -o c:¥hirdb¥pdexp¥defsql_file   ..............2
      -a   ..........................................3
```

[説明]

1. 搬出ファイルの名称　**host2:c:¥hirdb¥pdexp¥export_file1**
2. 定義系 SQL 出力ファイルの名称　**c:¥hirdb¥pdexp¥defsql_file**
3. 搬入と定義系 SQL の生成を同時にする場合の指定

10 リバランスユティリティ（pdrbal）

この章では，ハッシュ分割された表に対してリバランスをするリバランスユティリティ（pdrbal）について説明します。リバランスとは，データ量の増加に備えて新たに RD エリアを追加する場合に，既存の RD エリアと追加した RD エリアとの間でデータの再配置をすることをいいます。

10.1　概要

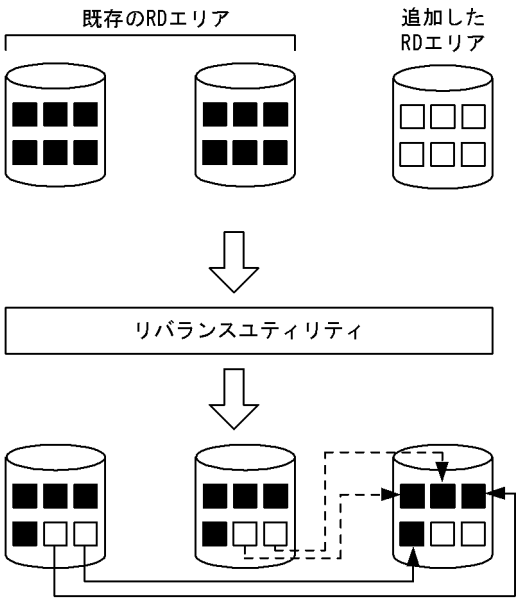
10.1.1　リバランスユティリティの機能

ハッシュ分割表のデータ量が増加した場合に RD エリアを追加すると，既存の RD エリアと追加した RD エリアとの間でデータ量の偏りが生じます。リバランスユティリティ（pdrbal）は，このデータ量の偏りを修正するためのユティリティです。リバランスユティリティを実行してハッシュ分割表のデータ量の偏りを修正することを，**ハッシュ分割表のリバランス機能**といいます。

リバランスユティリティを実行すると，ハッシュグループ単位にデータを再配置します。これを**表のリバランス**といいます。また，ハッシュ関数（HASHA～HASHF）で分割した表のことを**リバランス表**といいます。

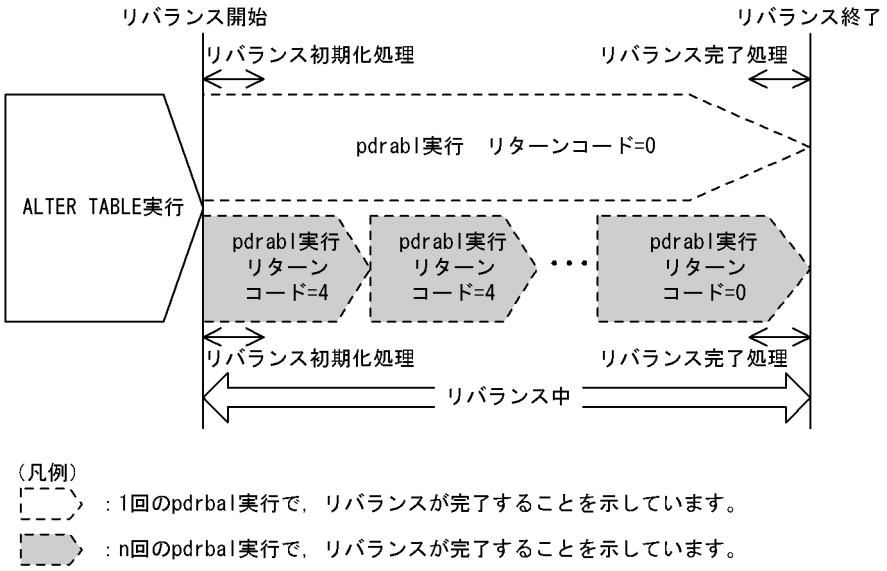
リバランスユティリティの概要とリバランスの概念を次の図に示します。

図 10–1　リバランスユティリティの概要



（凡例）
■：データが格納されていることを示します。
□：データが格納されていないことを示します。

図 10–2　リバランスの概念



[説明]

リバランス開始：

リバランス表に対して，RD エリアの追加（ALTER TABLE ADD RDAREA）をした後，最初に pdrbal を実行した時点をいいます。

リバランス初期化処理：

RD エリア追加後，最初に pdrbal を実行したときの初期化処理をいいます。

リバランス完了処理：

pdrbal がリターンコード 0 で終了したときの完了処理をいいます。

リバランス終了：

すべての表データのリバランスが完了（pdrbal がリターンコード 0 で終了）した時点をいいます。

リバランス中：

リバランス開始からリバランス終了までの間のことをいいます。

10.1.2　リバランスユティリティの動作モード

リバランスユティリティの動作モードを次の表に示します。

表 10–1　リバランスユティリティの動作モード

動作モード	説　　明
共有モード	pdrbal 実行中に表をアクセスできます。 24 時間運転などでオンライン業務を停止できないシステムの場合に適用してください。
占有モード	pdrbal 実行中は表をアクセスできません。

動作モード	説　明
	表に対するアクセスを停止できて，夜間にバッチ処理が実行できるシステムの場合に適用してください。

(1)　共有モード

共有モードの場合，リバランスユティリティの実行中でも，その表に対してほかからアクセスできます。共有モードで実行する場合は，-k オプションに share を指定します。

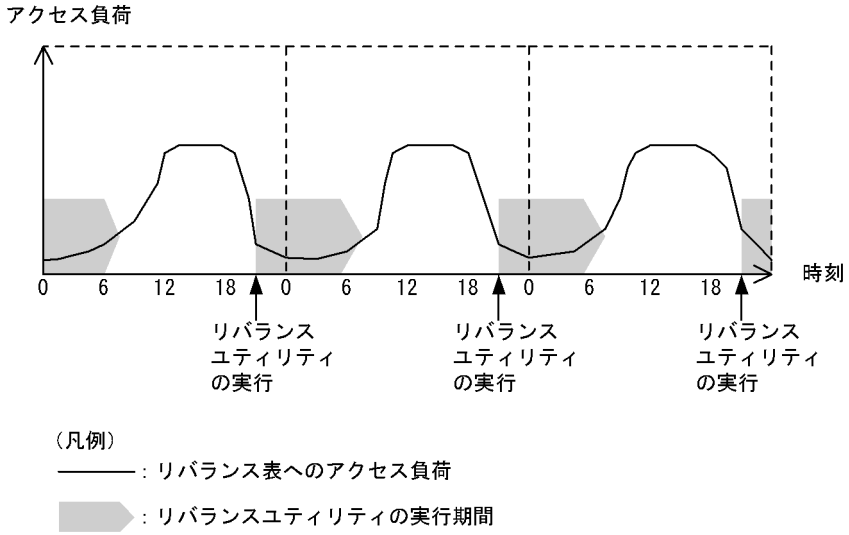
共有モードの場合，ログの取得が必要となります。また，同時実行する SQL などと競合する資源の確保，解放処理が複数回必要となるため，リバランス処理に掛かる時間は占有モードより長くなります。

共有モードで圧縮表のリバランスを実行する場合も，圧縮列がない場合に比べてリバランス処理に掛かる時間が長くなることがあります。これは，圧縮列のデータを移動するときにデータの伸張及び圧縮を行うためです。この場合，KFPL33509-W メッセージが出力されますが，リバランス処理は続行されます。実行時間を短縮したい場合は，pdcancel コマンドで共有モードのリバランス処理を中断し，占有モードで実行し直してください。

＜推奨する運用方法＞

共有モードでリバランスユティリティを実行する場合，リバランス表へのトラフィックの少ない時間帯を選ぶようにしてください。共有モードでのリバランスユティリティの運用例を次の図に示します。

図 10–3　共有モードでのリバランスユティリティの運用例



(2)　占有モード

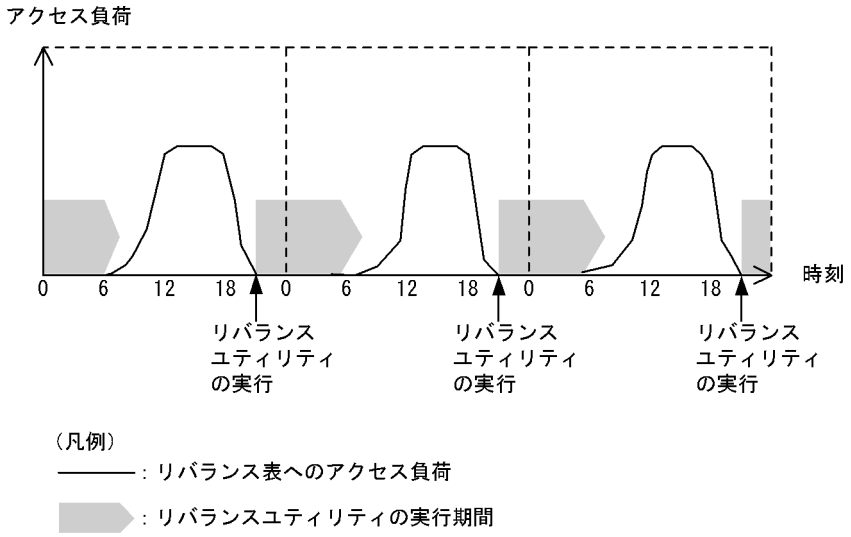
占有モードの場合，リバランスユティリティ実行中はその表に対してアクセスできません。占有モードで実行する場合は，-k オプションに exclusive を指定します。

占有モードの場合，ログの取得は任意となります。また，資源の確保，解放処理は 1 回だけとなり，更にインデクス一括作成もできるため，リバランス処理に掛かる時間は共有モードに比べると短くなります。

＜推奨する運用方法＞

占有モードでリバランスユティリティを実行する場合，リバランス表へのアクセスのない時間帯を選ぶようにしてください。占有モードでのリバランスユティリティの運用例を次の図に示します。

図 10–4　占有モードでのリバランスユティリティの運用例



10.1.3　実行環境

1. リバランスユティリティは，HiRDB が稼働中のときだけ実行できます。
2. リバランスユティリティは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行できます。
3. リバランス表，及びインデクスが格納されている RD エリアの，オープン属性及び RD エリアの状態によってリバランスユティリティの実行可否が変わります。リバランスユティリティの実行可否については，「コマンド実行時の RD エリアの状態」を参照してください。
4. リバランスユティリティ実行時，制御情報ファイルでファイル出力先を指定しない場合，次の表に示すディレクトリにファイルを出力します。

表 10–2　pdrbal がサーバでファイルを出力するディレクトリ

制御文※1の指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※2の指定	
		あり	なし
あり	制御文に指定したディレクトリ又はファイル		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※1　index 文，idxwork 文，又は sort 文

注※2　システム環境変数の指定

表 10–3　コマンド実行環境でファイルを出力するディレクトリ

report 文の指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	制御文に指定したディレクトリ又はファイル		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※　コマンド実行環境の環境変数の指定

5. 回復不要 FES がある場合，回復不要 FES の適用有無とフロントエンドサーバの稼働状態によって pdrbal の実行可否が異なります。pdrbal の実行可否を次に示します。

回復不要 FES		pdrbal の実行可否
適用していないフロントエンドサーバ	適用しているフロントエンドサーバ	
すべて稼働中	なし	○
一部停止中		×（フロントエンドサーバが停止中のため）
すべて停止中		×（表定義が取得できないため）
すべて稼働中	すべて稼働中	○
	一部停止中	○
	すべて停止中	○
なし	すべて稼働中	○
	一部停止中	○
	すべて停止中	×（表定義が取得できないため）

（凡例）

○：実行できます。

×：実行できません。

10.1.4　実行者

リバランスユティリティは，次のユーザが実行できます。

- DBA 権限を持っているユーザ
- SELECT 権限，INSERT 権限，及び DELETE 権限を持っているユーザ

10.2　使用例

リバランスユティリティの使用例を示します。

10.2.1　共有モードの場合

例１：

```
HiRDB/シングルサーバの場合に，表（reb_table）にRDエリア（user03）を追加して，共有モードで
リバランスユティリティを実行します。
・表定義
  CREATE TABLE reb_table(hkeys INT NOT NULL,names CHAR(30))
      FIX HASH HASHA BY hkeys IN (user01,user02)
・インデクス定義
  CREATE INDEX idx1 ON reb_table(hkeys) IN ((idx01),(idx02))
```

●ALTER TABLE での RD エリアの追加

```
 ALTER TABLE reb_table ADD RDAREA user03 FOR INDEX idx1 in idx03
```

[説明]

表 reb_table に RD エリア user03 を追加します。また，インデクス idx1 に対しても RD エリア idx03 を追加します。

●pdrbal のコマンド

```
 pdrbal -k share -t reb_table c:¥hirdb¥pdrbal¥control_file
```

[説明]

-k share：動作モード（共用モード）

-t reb_table：リバランス表の名称

c:¥hirdb¥pdrbal¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdrbal¥control_file）の内容

```
 execstop time,9:00    [1]
 report c:¥hirdb¥pdrbal¥rest_file    [2]
```

[説明]

1. 9 時間後に pdrbal を停止させます。
2. 処理結果ファイル（c:¥hirdb¥pdrbal¥rest_file）に pdrbal の実行結果を出力します。

例２：

```
HiRDB/シングルサーバの場合に，表（sgml_table）にRDエリア（user03）を追加して，共有モード
でリバランスユティリティを実行します。なお，表sgml_tableにはプラグインが提供する抽象デー
```

```
タ型の列があるものとします。
・表定義
 CREATE TABLE sgml_table (
     hkeys INT NOT NULL,
     doctext SGMLTEXT ALLOCATE (sgmltext IN ((LOB1D),(LOB2D)))
     PLUGIN '<DTD>新聞</DTD><EXTRACTparm>extract.prm</EXTRACTparm>'
     )
     FIX HASH HASHA BY hkeys IN (user01,user02)
・インデクス定義
 CREATE INDEX idx1 ON sgml_table(hkeys) IN ((idx01),(idx02))
 CREATE INDEX ngram_index USING TYPE MASTER.NGRAM on sgml_table(doctext)
     in ((LOB1I),(LOB2I))
```

●ALTER TABLE での RD エリアの追加

```
ALTER TABLE sgml_table ADD RDAREA user03
    FOR COLUMN doctext ALLOCATE(sgmltext IN LOB3D)
    FOR INDEX idx1 in idx03,ngram_index in LOB3I
```

[説明]

表 sgml_table に RD エリア user03 を追加します。また，インデクス idx1 に対して RD エリア idx03 を，プラグインインデクス ngram_index に対して RD エリア LOB3I を追加します。

●pdrbal のコマンド

```
pdrbal -k share -t sgml_table c:¥hirdb¥pdrbal¥control_file
```

[説明]

-k share：動作モード（共用モード）

-t sgml_table：リバランス表の名称

c:¥hirdb¥pdrbal¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdrbal¥control_file）の内容

```
execstop time,12:00    [1]
unld_func type=sgmltext,func=unsgmltext(sgmltext)    [2]
reld_func type=sgmltext,func=sgmltext(blob)    [3]
report c:¥hirdb¥pdrbal¥rest_file    [4]
```

[説明]

1. 12 時間後に pdrbal を停止させます。
2. コンストラクタパラメタ逆生成関数を指定します。

   sgmltext：抽象データ型の名称

   unsgmltext：コンストラクタパラメタ逆生成関数の名称

   sgmltext：引数の型
3. コンストラクタ関数を指定します。

   sgmltext：抽象データ型の名称

sgmltext：コンストラクタ関数の名称

blob：引数の型

4. 処理結果ファイル（c:¥hirdb¥pdrbal¥rest_file）に pdrbal の実行結果を出力します。

10.2.2　占有モードの場合

例３：

```
HiRDB/シングルサーバの場合に，表（reb_table）にRDエリア（user03）を追加して，占有モードで
リバランスユティリティを実行します。
・表定義
  CREATE TABLE reb_table(hkeys INT NOT NULL,names CHAR(30))
      FIX HASH HASHA BY hkeys IN (user01,user02)
・インデクス定義
  CREATE INDEX idx1 ON reb_table(hkeys) IN ((idx01),(idx02))
```

●ALTER TABLE での RD エリアの追加

```
 ALTER TABLE reb_table ADD RDAREA user03 FOR INDEX idx1 in idx03
```

[説明]

表 reb_table に RD エリア user03 を追加します。また，インデクス idx1 に対しても RD エリア idx03 を追加します。

●pdrbal のコマンド

```
 pdrbal -k exclusive -t reb_table -l n c:¥hirdb¥pdrbal¥control_file
```

[説明]

-k exclusive：動作モード（占有モード）

-t reb_table：リバランス表の名称

-l n：ログレスモード

c:¥hirdb¥pdrbal¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdrbal¥control_file）の内容

```
 idxwork c:¥hirdb¥pdrbal¥idxwork1    [1]
 sort c:¥hirdb¥pdrbal¥sortwork    [2]
 report c:¥hirdb¥pdrbal¥rest_file    [3]
```

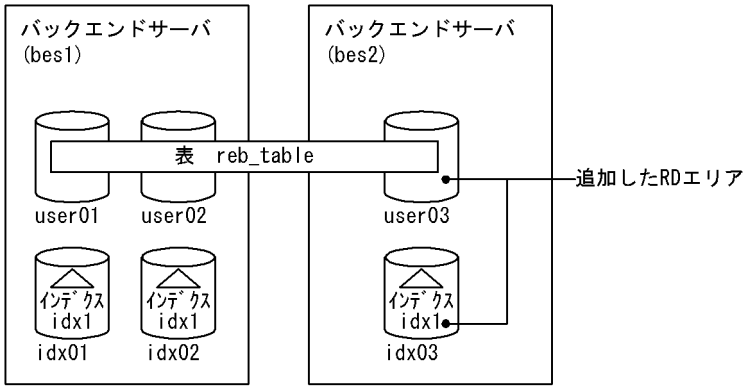
[説明]

1. インデクス情報ファイル用ディレクトリ（c:¥hirdb¥pdrbal¥idxwork1）を指定します。
2. ソート用ワークディレクトリ（c:¥hirdb¥pdrbal¥sortwork）を指定します。
3. 処理結果ファイル（c:¥hirdb¥pdrbal¥rest_file）に pdrbal の実行結果を出力します。

例４：

```
HiRDB/パラレルサーバの場合に，表（reb_table）にRDエリア（user03）を追加して，占有モードでリバランスユティリティを実行します。
・表定義
  CREATE TABLE reb_table(hkeys INT NOT NULL,names CHAR(30))
      FIX HASH HASHA BY hkeys IN (user01,user02)
・インデクス定義
  CREATE INDEX idx1 ON reb_table(hkeys) IN ((idx01),(idx02))
```

●サーバ，RD エリア，表，及びインデクスの関連



●ALTER TABLE での RD エリアの追加

```
ALTER TABLE reb_table ADD RDAREA user03 FOR INDEX idx1 in idx03
```

[説明]

表 reb_table に RD エリア user03 を追加します。また，インデクス idx1 に対しても RD エリア idx03 を追加します。

●pdrbal のコマンド

```
pdrbal -k exclusive -t reb_table -l n c:¥hirdb¥pdrbal¥control_file
```

[説明]

-k exclusive：動作モード（占有モード）

-t reb_table：リバランス表の名称

-l n：ログレスモード

c:¥hirdb¥pdrbal¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdrbal¥control_file）の内容

```
idxwork bes2 c:¥hirdb¥pdrbal¥idxwork_bes2    [1]
sort bes2 c:¥hirdb¥pdrbal¥sortwork_bes2    [2]
report c:¥hirdb¥pdrbal¥rest_file      [3]
```

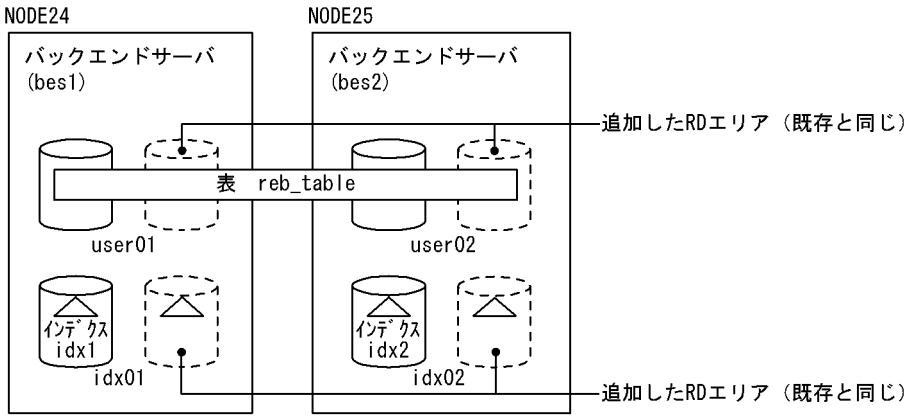
[説明]

1. バックエンドサーバ bes2 のインデクス情報ファイル用ディレクトリ（c:¥hirdb¥pdrbal¥idxwork_bes2）を指定します。
2. バックエンドサーバ bes2 のソート用ワークディレクトリ（c:¥hirdb¥pdrbal¥sortwork_bes2）を指定します。
3. 処理結果ファイル（c:¥hirdb¥pdrbal¥rest_file）に pdrbal の実行結果を出力します。

例５：

```
HiRDB/パラレルサーバの場合に，表（reb_table）に既存と同じRDエリアを追加して，占有モードでリバランスユティリティを実行します。なお，表はホスト（NODE24）上のバックエンドサーバ（bes1）とホスト（NODE25）上のバックエンドサーバ（bes2）に分割格納されています。
・表定義
  CREATE TABLE reb_table (hkeys INT NOT NULL,names CHAR(30))
      FIX HASH HASHF BY hkeys IN (user01,user02)
・インデクス定義
  CREATE INDEX idx1 ON reb_table(hkeys) IN ((idx01),(idx02))
```

●サーバ，RD エリア，表，及びインデクスの関連



●ALTER TABLE での RD エリアの追加

```
ALTER TABLE reb_table ADD RDAREA user01 FOR INDEX idx1 in idx01
ALTER TABLE reb_table ADD RDAREA user02 FOR INDEX idx1 in idx02
```

[説明]

表 reb_table に既存と同じ RD エリア（user01，user02）を追加します。また，インデクス idx1 に対しても既存と同じ RD エリア（idx01，idx02）を追加します。

●pdrbal のコマンド

```
pdrbal -k exclusive -t reb_table -i c c:¥hirdb¥pdrbal¥control_file
```

[説明]

-k exclusive：動作モード（占有モード）

-t reb_table：リバランス表の名称

-i c：インデクス作成モード（一括作成モード）

c:¥hirdb¥pdrbal¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdrbal¥control_file）の内容

```
report c:¥hirdb¥pdrbal¥rest_file    [1]
idxwork bes1 c:¥hirdb¥pdrbal¥idxwork_bes1    [2]
sort bes1 c:¥hirdb¥pdrbal¥sortwork_bes1    [2]
idxwork bes2 c:¥hirdb¥pdrbal¥idxwork_bes2    [3]
sort bes2 c:¥hirdb¥pdrbal¥sortwork_bes2    [3]
```

[説明]

1. 処理結果ファイル（c:¥hirdb¥pdrbal¥rest_file）に pdrbal の実行結果を出力します。
2. 追加した RD エリア（idx01）に対するバックエンドサーバ bes1 の情報を指定します。
   インデクス情報ファイル用ディレクトリ名：c:¥hirdb¥pdrbal¥idxwork_bes1
   ソート用ワークディレクトリ名：c:¥hirdb¥pdrbal¥sortwork_bes1
3. 追加した RD エリア（idx02）に対するバックエンドサーバ bes2 の情報を指定します。
   インデクス情報ファイル用ディレクトリ名：c:¥hirdb¥pdrbal¥idxwork_bes2
   ソート用ワークディレクトリ名：c:¥hirdb¥pdrbal¥sortwork_bes2

10.3　コマンドの形式

10.3.1　形式

```
pdrbal 〔-k 動作モード〕
       -t 〔認可識別子.〕表識別子
        〔-i インデクス作成方法〕※
        〔-l ログ取得方式〕※
        〔-u 認可識別子〕
        〔-c コミット単位〕
        〔-n 〔一括入出力用ローカルバッファ面数〕，〔ランダムアクセス用ローカルバッファ面数〕〕※
        〔-m 〔経過メッセージ出力間隔〕〔,インフォメーションメッセージ出力抑止レベル〕〕
        〔-W 実行監視時間〕
        〔制御情報ファイル名〕
```

注

制御情報ファイル名を指定する場合，必ずオプションの最後に指定してください。

注※

共有モード（-k share）の場合は，指定しても無視されます。

10.3.2　オプション

(1)　-k 動作モード

〜≪share≫

pdrbal の動作モードを指定します。

share

pdrbal 実行中でも，他ユーザが表を参照，更新できるモード（共有モード）です。

共有モードは，表に対するオンライン業務を停止しないで，pdrbal を実行する場合に適用してください。

exclusive

pdrbal 実行中は，他ユーザに表を参照，更新させないモード（占有モード）です。

占有モードは，表に対するオンライン業務を停止して，pdrbal を実行する場合に適用してください。

(2)　-t 〔認可識別子.〕表識別子

リバランスユティリティを実行するリバランス表の名称を指定します。

認可識別子を省略した場合，HiRDB に接続したユーザの認可識別子が仮定されます。

＜規則＞

1. 次の表は指定できません。
   - ビュー表
   - リバランス表（HASH 関数（HASHA～HASHF）で分割した表）以外の表
2. 認可識別子又は表識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。
3. 抽象データ型を含むリバランス表に対する pdrbal の実行可否を次の表に示します。

表 10–4　抽象データ型を含むリバランス表に対する pdrbal の実行可否

抽象データ型の内容		pdrbal 実行可否
ユーザが定義した抽象データ型		×
プラグインが提供する抽象データ型	BLOB 属性あり	○※
	BLOB 属性なし	○

（凡例）

○：実行できます。

×：実行できません。

注※

プラグインによっては，データベース再編成用のコンストラクタ逆生成関数を指定したときだけ実行できます。

(3)　-i インデクス作成方法

～≪c≫

インデクスの作成方法を指定します。

なお，共有モードの場合は指定しても無視されます（s が仮定されます）。

c

インデクス一括作成モードを示します。このモードを指定した場合，行データをリバランスした後，インデクスを一括作成します。

＜適用基準＞

占有モードでリバランスした行データのインデクスを，インデクス情報ファイル，ソート用ワークファイルを使用して高速に作成したいときに指定してください。なお，execstop 文を指定した場合，データ移動後にインデクスの一括作成をするため，移動したデータのインデクスが作成されるまで pdrbal が停止できないことがあります。

s

インデクス更新モードを示します。このモードでは，行データを移動するごとにインデクスを更新します。

＜適用基準＞

占有モードでリバランスした行データが少ない場合，又はインデクス情報ファイル，ソート用ワークファイルが確保できない場合に指定してください。また，execstop 文に指定した pdrbal 実行時間で pdrbal を確実に停止させたい場合にも指定してください。

＜注意事項＞

1. index 文，及び idxwork 文を省略すると，既定の出力先ディレクトリにインデクス情報ファイルを出力します。pdrbal が異常終了すると，このインデクス情報ファイルが不完全な状態で残ります。このため，pdrbal の異常終了を何度も繰り返すと不完全なインデクス情報ファイルが増加し，ディスク容量など OS の資源を圧迫することになります。したがって，既定の出力先ディレクトリに出力した必要のないインデクス情報ファイルについては削除するようにしてください。

   index 文を省略して idxwork 文だけを指定した場合に，idxwork 文に指定したディレクトリ下に出力するインデクス情報ファイルについても同様です。

   既定の出力先ディレクトリについては「注意事項」の表「pdrbal がファイルを出力するディレクトリ」を参照してください。

   また，インデクス情報ファイルは，次の名称規則で出力します。

   ```
   INDEX-インデクス名※-インデクス格納用RDエリア名※-一意となる文字列
   ```

   注※

   インデクス名又はインデクス格納 RD エリア名に「¥」が含まれている場合は，「¥」を「(」に置き換えます。

(4)　-l ログ取得方式

～《a》

pdrbal 実行時の，データベースの更新ログの取得方法を指定します。

なお，占有モードの場合は必ず-l オプションを指定してください。また，共有モードの場合は指定できません（a が仮定されます）。

a

ログ取得モードを示します。ロールバック，及びロールフォワードに必要なデータベースの更新ログを取得します。

＜適用基準＞

リバランスの行データが少ない場合，又は pdrbal の実行前，実行後にバックアップを取得したくない場合に指定します。

ログ取得モードで実行した場合，pdrbal の実行前，実行後のバックアップは取得する必要はありませんが，ログレスモードと比べて処理性能は劣ります。

n

ログレスモードを示します。データベースの更新ログは取得しません。

＜適用基準＞

リバランスの行データが大量の場合に指定します。

ログレスモードで実行した場合，ログ取得モードに比べると実行時間は短縮できます。ただし，pdrbal 実行中の異常終了に備えて，pdrbal の実行前にバックアップを取得する必要があります。また，データベースの更新ログが取得されていないため pdrbal 実行後にもバックアップの取得が必要です。

なお，pdrbal が異常終了した場合，表格納用 RD エリアは障害閉塞となるため，pdrbal 実行前に取得したバックアップで回復する必要があります。

＜規則＞

1. 以前に取得したバックアップから回復できるときは，pdrbal 実行前のバックアップ取得は不要です。
2. ログレスモードでは，データベースの更新ログの代わりに，1 サーバ当たり次の量の ENQ ログを出力します。

   ENQ ログ＝（p ＋ q ＋ r）×T

   p：表格納用 RD エリア数

   q：LOB 列（LOB 属性）格納用 RD エリア数

   r：インデクス格納用 RD エリア数

   T：トランザクション数（-c オプションの説明を参照してください）
3. ログレスモードの運用方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
4. execstop 文を指定して pdrbal を数回に分けて実行する場合，実行するごとにバックアップを取得する必要があります。

(5)　-u 認可識別子

pdrbal を実行するユーザの認可識別子を指定します。

省略した場合の仮定値については，＜省略時仮定値＞を参照してください。

このオプションを指定した場合，パスワード入力要求の応答付きメッセージが表示されます。パスワードが不要な場合は，応答付きメッセージに対して空送信してください。

ここで指定した認可識別子で HiRDB に接続し，実行権限のチェックをします。

＜適用基準＞

環境変数 PDUSER に設定されている認可識別子ではない，別の認可識別子で実行したい場合に指定します。

＜省略時仮定値＞

省略した場合，次の方法で認可識別子/パスワードが仮定されます。

1. pdrbal 実行時の環境変数 PDUSER の設定値が仮定されます。パスワードの応答ができない環境で実行するときは，PDUSER を必ず設定してください。環境変数 PDUSER の指定例を次に示します。

   （例）

   パスワードありの場合　PDUSER=¥””認可識別子”¥”/¥””パスワード”¥”

   パスワードなしの場合　PDUSER=¥””認可識別子”¥”

2. 環境変数 PDUSER が設定されていない場合，ログインウィンドウのユーザ名が仮定されます。パスワードについては，入力要求の応答付きメッセージが表示されるので，そこで応答してください。パスワードが不要な場合は，応答付きのメッセージに対して空送信してください。

＜規則＞

1. このオプションは，応答できない環境では指定しないでください。
2. 認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(6)　-c コミット単位

～＜符号なし整数＞((0～1000000))

≪共有モードの場合は 10，占有モードの場合は 100000≫

リバランス処理でデータを再配置するための行データの移動で，何行移動したときにコミットするかを行数で指定します。

0 を指定した場合は，1 トランザクション当たりの移動行数を制限しません（リバランスが終了するまで，又は execstop 文で指定した時間までを 1 トランザクションとして移動処理をします）。

HiRDB/パラレルサーバの場合，各 RD エリアに格納されている行数によっては，ここで指定したコミット単位より少ない行数でコミットすることがあります。

＜推奨値＞

共用モードの場合，pdrbal では行データ移動のときに移動元と移動先の RD エリアに対して排他を掛けます。そのため，該当する移動元，移動先 RD エリアに対して，移動が終了するまで（コミットするまで）アクセスできなくなります。したがって，共有モードの場合には，コミット単位に小さい値を指定することでほかからの排他待ち時間を短くできます。

UAP などの動作を優先したい場合は小さい値を指定し，リバランス処理を優先したい場合は大きな値を指定してください。

占有モードで，かつログレスモードの場合，処理対象となる表にインデクスが定義しているときは，コミット単位に 0 を指定することで処理性能を向上できます（コミット時のインデクスメンテナンス処理で更新したグローバルバッファのフラッシュ処理が 1 回になるため，性能向上が期待できます。また，

コミット数も削減できます）。ただし，1 トランザクションで処理するため，処理行数の増加に伴い，排他資源数が増加します。また，シンクポイントダンプを取得しない期間も長くなります。

＜注意事項＞

1. オンライン業務でのトランザクション数が比較的多い場合，100 以下を指定することをお勧めします。
2. コミット単位に小さい値を指定した場合，リバランス処理の時間は長くなり，また，コミット回数の増加に伴い多量のトランザクションログが出力されます。トランザクションログは，-l オプションの指定値に関係なく必ず出力されるログで，次の計算式から求められます。

   トランザクションログ量（単位：バイト）

   ＝（1328 ＋ 176×3）×T

   T：(a÷b ＋ 1024÷c)

   a：リバランス対象のデータ件数

   b：-c オプションの指定値

   c：表格納用 RD エリアの数

   トランザクションログは，コミットするごとに上記の計算式のログを，各サーバのシステムログファイルに出力します。各サーバとは，すべてのフロントエンドサーバ，及び表格納用 RD エリアがあるバックエンドサーバのことです。また，抽象データ型がある表の場合は，ディクショナリサーバも含まれます。
3. コミット単位に大きい値を指定した場合，シンクポイントを取得できない期間が長くなり，ほかの UAP と同時実行中に障害が発生したときは，再開始に要する時間が増大します。
4. 占有モードで，かつログレスモードの場合，コミット単位に 0 を指定できます。それ以外の場合は 0 を指定できません。

(7) -n 〔一括入出力用ローカルバッファ面数〕,〔ランダムアクセス用ローカルバッファ面数〕

占有モードでリバランスをする場合，ローカルバッファを使用するときに指定します。このオプションを指定すると，ローカルバッファを使用して DB アクセスができるようになり，一括入出力によって入出力回数を削減できます。

このオプションを省略した場合は，グローバルバッファを使用して 1 ページ単位に入出力をします。

一括入出力用ローカルバッファ面数　～＜符号なし整数＞((2～4096))

一括入出力用ローカルバッファの面数を指定します。一括入出力用ローカルバッファは，データページに対して使用します。

一括入出力用ローカルバッファ面数には，16～32 面を指定することをお勧めします。184 キロバイト÷ページ長を上限に指定値を決めてください。

ランダムアクセス用ローカルバッファ面数　～＜符号なし整数＞((4～125000))

ランダムアクセス用ローカルバッファの面数を指定します。ランダムアクセス用ローカルバッファは，インデクスページに対して使用します。

一括入出力用ローカルバッファ面数，及びランダムアクセス用ローカルバッファ面数は，表定義によって指定の組み合わせを変更することをお勧めします。推奨する-n オプションの指定を次の表に示します。

表 10–5　推奨する-n オプションの指定（pdrbal）

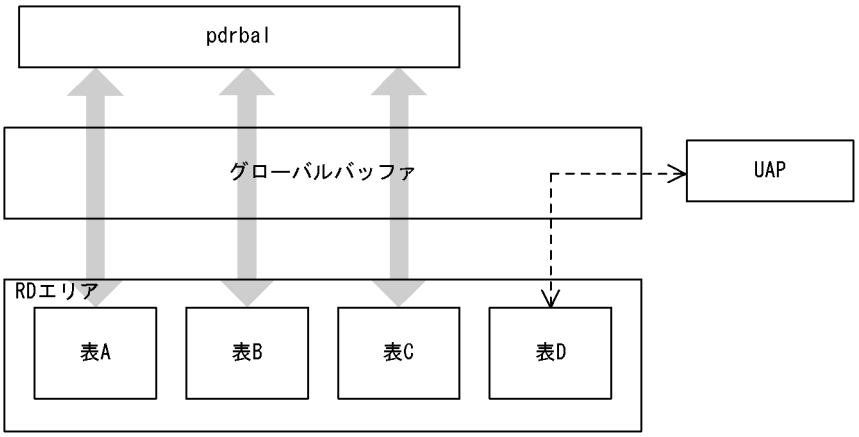
表種別	列定義	-n オプションの指定
FIX 表，又は非 FIX 表	すべての列が NULL である	-n ランダムアクセス用ローカルバッファ面数
非 FIX 表	可変長データ型の列が定義されている	-n ランダムアクセス用ローカルバッファ面数
	抽象データ型列が定義されている	
	BINARY 列が定義されている	
FIX 表，又は非 FIX 表	上記以外	-n 一括入出力用ローカルバッファ面数

＜ pdrbal が使用するバッファについて＞

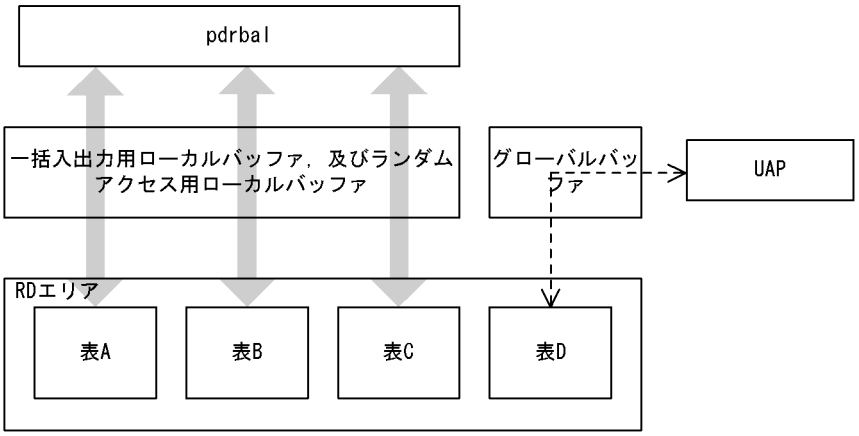
-n オプションを省略した場合，グローバルバッファを使用します。この場合，リバランス時に大量のグローバルバッファを使用するため，そのグローバルバッファを使用する UAP のトランザクション性能が劣化します。-n オプションを指定すると，このようなバッファ競合はなくなります。pdrbal とバッファの関係を次の図に示します。

図 10–5　pdrbal とバッファの関係

●グローバルバッファだけを使用する場合



●ローカルバッファとグローバルバッファを使用する場合



〔説明〕

グローバルバッファだけを使用する場合（-n オプションを省略した場合），pdrbal と UAP とでバッファの競合が発生します。

ローカルバッファとグローバルバッファを使用する場合（-n オプションを指定した場合），pdrbal と UAP とのバッファの競合は発生しません。ただし，LOB 列がある表のリバランスの場合は，-n オプションを指定してもグローバルバッファを使用します。

＜規則＞

1. このオプションを省略した場合は１を仮定して，グローバルバッファを使用します。したがって，一括入出力はしません。
2. FIX ハッシュ分割のリバランス表に対してこのオプションを指定する場合，ハッシュグループごとに指定したページ数のバッファを確保するため，フレキシブルハッシュ分割のリバランス表に比べてメモリ消費量が大きくなります。なお，ハッシュグループとは，分割キーをハッシュした結果を基に HiRDB がデータを 1024 のグループに分けるときの，このグループのことをいいます。リバランス中は，このグループごとに RD エリアのセグメントを割り当ててデータを格納します。

3. このオプションを省略した場合（グローバルバッファを使用する場合），次の計算式以上のバッファ面数がないと十分な性能は得られません。

   1RD エリアに必要なバッファ面数

   ＝ 1024÷リバランス表が格納されている RD エリア数×2 ＋ 3

(8) -m 〔経過メッセージ出力間隔〕〔,インフォメーションメッセージ出力抑止レベル〕

経過メッセージ出力間隔 ～＜符号なし整数＞((1～1000))《10》

処理の実行状況を出力するメッセージの出力間隔を，1 万行単位で指定します。

実行状況のメッセージは RD エリアごとに出力されます。なお，このメッセージは，-m オプションに lvl2 を指定すると，トラブルシュート情報を出力するワークファイルにも出力されます。

インフォメーションメッセージ出力抑止レベル ～《lvl0》

実行した端末の標準出力，イベントログファイル，及びメッセージログファイルへのインフォメーションメッセージの出力を抑止します。標準出力，イベントログファイル，及びメッセージログファイルのすべての出力を抑止する場合，トラブルシュート情報はワークファイルに出力します。指定できるオプションとメッセージの出力先を次の表に示します。

表 10–6 オプションの指定値とメッセージの出力先

オプションの指定値	意味	出力先		
		標準出力	イベントログ及びメッセージログファイル	ワークファイル
lvl0	出力を抑止しない	○	○	×
lvl1	標準出力への出力を抑止する	×	○	×
lvl2	標準出力，イベントログ及びメッセージログファイルへの出力を抑止する	×	×	○

（凡例）

○：メッセージを出力します。

×：メッセージを出力しません。

ワークファイルにメッセージを出力するのはトラブルシュート目的（ユティリティの動作を確認するため）であり，通常，メッセージを確認する必要がない場合は lvl1 を指定することをお勧めします。

＜ワークファイル＞

ワークファイルは，-m オプションに lvl2 を指定すると，ユティリティを実行するごとに各サーバに作成されます。このため，pdcspool コマンドで HiRDB によって定期的に削除されるように，pd_tmp_directory オペランドにディレクトリを指定しておくことをお勧めします。ワークファイルの出力先を次の表に示します。

表 10–7　ワークファイルの出力先

システム定義の pd_tmp_directory オペランドの指定	環境変数 TMP の指定	ワークファイルの出力先
あり	−	pd_tmp_directory オペランドに指定したディレクトリ
なし	あり	環境変数 TMP に指定したディレクトリ
	なし	%PDDIR%¥tmp ディレクトリ

（凡例）

−：該当しません。

なお，ファイルオープンエラーや I/O エラーなどのためにインフォメーションメッセージがワークファイルへ出力できない場合は，エラーメッセージが出力されますが，処理は続行します。

ワークファイルは，HiRDB が自動で作成します。ファイル名称は，PDRBAL-aa-bb の形式で決められます。それぞれの意味について説明します。

- PDRBAL：pdrbal が出力したファイルであることを示す固定のプリフィクスです。
- aa：プロセスのサーバ名称です。
- bb：ファイル作成時間，及びプロセス ID です。

(9)　-W 実行監視時間　～＜符号なし整数＞((0～3600))

リバランスユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。

このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

オプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(10)　制御情報ファイル名

～＜パス名＞

pdrbal の制御文を記述した制御情報ファイルの名称を指定します。

制御情報ファイルに記述できる制御文を次の表に示します。各制御文については，10.3.3～10.3.10 を参照してください。なお，制御情報ファイル中にコメントは記述できません。

表 10–8　制御情報ファイルに記述できる制御文

制御文（記述する内容）	動作モード（-k オプション）	
	共用モード（share）	占有モード（exclusive）
index 文 （インデクス情報ファイルの情報の記述）	×	○
idxwork 文 （インデクス情報ファイル用ディレクトリの情報の記述）	×	○
sort 文 （ソート用ワークディレクトリの情報の記述）	×	○
execstop 文 （pdrbal 実行時間の情報の記述）	○	○
unld_func 文 （コンストラクタパラメタ逆生成関数の情報の記述）	○※	○※
reld_func 文 （コンストラクタ関数の情報の記述）	○※	○※
report 文 （実行結果ファイルの情報の記述）	○	○
option 文 （オプション機能の記述）	○	○

（凡例）

○：任意に指定できます。

×：指定できません。

注※

プラグインによっては，指定しなければならないものもあります。詳細については，各プラグインマニュアルを参照してください。

制御文で記述するファイル，及びディレクトリの規則を次に示します。

1. HiRDB 管理者に対してアクセス権限を与えておく必要があります。
2. パス名は大文字，小文字を区別しません。このため，大文字，小文字が異なるだけの同一パス名を指定すると，重複エラーとなります。
3. ネットワークドライブに接続するパス名は指定できません。

10.3.3　index 文（インデクス情報ファイルの情報の記述）

インデクス一括作成モード（-i c）でインデクスを作成する場合，インデクス情報を出力するインデクス情報ファイルの情報を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，index 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないようにするため，できるだけ指定してください。

インデクスの数，又はインデクスの格納 RD エリア数が多い場合は，idxwork 文を指定すると便利です。

＜規則＞

1. index 文は，ALTER TABLE で追加したインデクス格納用 RD エリアごとに 1 文指定します。
2. インデクスを定義していない表の場合，index 文を指定する必要はありません。
3. index 文を省略し，かつ idxwork 文も省略しているときは，インデクスが格納されているサーバの，表「pdrbal がサーバでファイルを出力するディレクトリ」に示すディレクトリ下にインデクス情報ファイルが作成されます。
4. index 文と idxwork 文を同時に指定した場合，index 文の指定が有効となります。
5. 指定したインデクス情報ファイルが既にある場合，上書きされます。
6. インデクスの作成が正常終了した場合，指定したインデクス情報ファイルは自動的に削除されます。

(1)　形式

```
index インデクス識別子 RDエリア名 インデクス情報ファイル名
```

(2)　説明

(a)　インデクス識別子

～＜識別子＞((1～30))

インデクスのインデクス識別子を指定します。

インデクス識別子を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，インデクス識別子に空白が含まれている場合は，引用符で囲んでください。

(b)　RD エリア名

～＜識別子＞((1～30))

ALTER TABLE で追加したインデクス格納用 RD エリアの名称を指定します。

RD エリア名を引用符（''）で囲んだ場合は，英大文字と英小文字が区別されます。引用符で囲まない場合は，すべて英大文字として扱われます。また，RD エリア名に空白が含まれている場合は，引用符で囲んでください。

(c) インデクス情報ファイル名

～＜パス名＞

インデクス情報を出力するインデクス情報ファイルの名称を絶対パス名で指定します。

10.3.4 idxwork 文（インデクス情報ファイル用ディレクトリの情報の記述）

インデクス一括作成モード（-i c）で index 文を省略した場合，自動的にインデクス情報ファイルが作成されるディレクトリの名称を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，idxwork 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないようにするため，できるだけ指定してください。

＜規則＞

1. index 文，及び idxwork 文の両方を省略した場合，ALTER TABLE で追加したインデクス格納 RD エリアがあるサーバの，表「pdrbal がサーバでファイルを出力するディレクトリ」に示すディレクトリにインデクス情報ファイルが作成されます。
2. HiRDB/シングルサーバの場合，一つだけ idxwork 文を指定します。
   HiRDB/パラレルサーバの場合，ALTER TABLE で追加したインデクス格納用 RD エリアがあるサーバの数だけ idxwork 文を指定します。指定するサーバ名は，インデクス格納 RD エリアがあるサーバの名称を指定します。
3. idxwork 文と index 文を同時に指定した場合，index 文の指定が有効となり，idxwork 文の指定は無視されます。
4. インデクスの作成が正常終了した場合，指定したディレクトリ下のインデクス情報ファイルは自動的に削除されます。

(1) 形式

```
idxwork 〔サーバ名〕 ディレクトリ名
```

(2) 説明

(a) サーバ名

～＜識別子＞((1～8))

インデクス情報ファイル用ディレクトリを作成するサーバの名称を指定します。

なお，HiRDB/シングルサーバの場合はサーバ名を指定しないでください。

(b)　ディレクトリ名

～＜パス名＞((1～255))

インデクス情報ファイルを作成するディレクトリの名称を絶対パス名で指定します。

(3)　作成されるインデクス情報ファイルの名称

作成されるインデクス情報ファイルの名称は次の形式で作成されます。

```
ディレクトリ名¥INDEX-インデクス名-インデクス格納用RDエリア名
  -一意となる文字列
```

例を次に示します。

- **条件**

  インデクス名：IDX1

  インデクス格納用 RD エリア：USER01

  idxwork 文の内容：idxwork c:¥hirdb¥pdrbal

- **作成されるインデクス情報ファイルの名称**

  c:¥hirdb¥pdrbal¥INDEX-IDX1-USER01-aaaa00001

> **注意事項**
>
> インデクス名又はインデクス格納 RD エリア名に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

10.3.5　sort 文（ソート用ワークディレクトリの情報の記述）

インデクス一括作成モード（-i c）の場合，インデクスを作成するために使用するソート用ワークファイルの情報を指定します。

＜適用基準＞

システム定義の pd_tmp_directory オペランド及び環境変数 TMP を指定していない場合，sort 文は，%PDDIR%¥tmp ディレクトリの容量不足を発生させないようにするため，できるだけ指定してください。

＜規則＞

1. sort 文を省略した場合，インデクス格納 RD エリアがあるサーバの，表「pdrbal がサーバでファイルを出力するディレクトリ」に示すディレクトリが仮定されます。

2. ソート用ワークファイルは，pdrbal の処理終了後に自動的に削除されます。
3. HiRDB/シングルサーバの場合，一つだけ sort 文を指定します。
   HiRDB/パラレルサーバの場合，ALTER TABLE で追加したインデクス格納用 RD エリアがあるサーバの数だけ sort 文を指定します。

(1) 形式

```
sort 〔サーバ名〕 ディレクトリ名
```

(2) 説明

(a) サーバ名

～＜識別子＞((1～8))

ソート用ワークファイルを作成するサーバの名称を指定します。

HiRDB/シングルサーバの場合，サーバ名は指定しないでください。

(b) ディレクトリ名

～＜パス名＞

ソート用ワークファイルを作成するディレクトリの名称を，絶対パス名で指定します。

10.3.6 execstop 文（pdrbal 実行時間の情報の記述）

意図的に pdrbal を停止させたい場合，pdrbal の実行時間を指定します。

＜適用基準＞

営業時間中など，オンライン処理が比較的混み合う時間帯に，pdrbal を実行しないでオンライン処理を優先させたい場合に指定してください。

＜規則＞

1. execstop 文を省略した場合，リバランスが完了するまで処理を続行します。
2. 予定終了時間になっても，リバランス中の行データ（インデクスも含まれます）が完全に移動するまで処理を続行します。したがって，予定終了時間で終わらない場合もあります。特に，インデクス一括作成モードの場合はデータ移動後にインデクスの一括作成をするため，これらを考慮した終了時間を指定してください。

(1) 形式

```
execstop time,pdrbal実行時間
```

(2) 説明

(a) pdrbal 実行時間

pdrbal の実行時間（HH:MM）を指定します。

HH：時間（00～168）

MM：分（00 又は 30）

168 時間（1 週間）を超える値は指定できません。また，HH 及び MM に 00 を指定した場合，リバランスはしないですぐに終了します（pdrbal 開始時の検索・削除・更新のアクセス手順，SQL オブジェクトの削除，及び SQL オブジェクトが無効となっている手続き・関数のリコンパイルはします）。

（例）

比較的トラフィックが少ない 21 時から 9 時までの間（12 時間），pdrbal を実行する場合は次のように指定します。

```
execstop time, 12:00
```

10.3.7 unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述）

抽象データ型の列がある表に対して pdrbal を実行する場合，プラグインのアンロード機能を使用してリバランスをします。このとき，アンロードで抽象データ型のデータ値を逆生成する，コンストラクタパラメタ逆生成関数の情報を指定します。

＜適用基準＞

アンロード機能を持つ，プラグインが提供している抽象データ型がある表をリバランスする場合に指定します。

＜規則＞

1. アンロード機能を持つ，プラグインが提供している抽象データ型がある表の場合は，unld_func 文を必ず指定してください。
2. unld_func 文は，表の列に定義されている抽象データ型の数分指定できます。
3. unld_func 文を指定するときは，対で reld_func 文も指定してください。

(1) 形式

```
unld_func type=〔認可識別子.〕抽象データ型名,
          func=関数名（引数の型〔,引数の型…〕）
           〔,func=関数名（引数の型〔,引数の型…〕）…〕
```

(2) 説明

(a) type=〔認可識別子.〕抽象データ型名

抽象データ型の認可識別子，及び抽象データ型名を指定します。

＜規則＞

1. 認可識別子を省略した場合，抽象データ型を定義したユーザの認可識別子（通常は MASTER）が仮定されます。
2. 認可識別子，及び抽象データ型の名称に小文字，空白を含む場合，引用符（”）で囲んでください。

(b) func=関数名（引数の型〔,引数の型...〕）

コンストラクタパラメタ逆生成関数の名称，引数の型を指定します。名称，引数の型については，各プラグインのマニュアルを参照してください。

関数名

コンストラクタパラメタ逆生成関数の名称を指定します。

引数の型

コンストラクタパラメタ逆生成関数の引数のデータ型を指定します。引数のデータ型を次の表に示します。

表 10–9　引数のデータ型（pdrbal）

記述方法	備考
integer	INT32
smallint	INT16
char nchar mchar	CHAR8
varchar nvarchar mvarchar	p_pdb_varchar_t
float	DOUBLE64
smallflt	REAL64
blob	BLOB
抽象データ型名	なし。
binary	p_pdb_binary_t

10.3.8　reld_func 文（コンストラクタ関数の情報の記述）

抽象データ型の列がある表に対して pdrbal を実行する場合，プラグインのアンロード機能を使用してリバランスをします。このとき，リロード時に抽象データ型の値を生成する，コンストラクタ関数の情報を指定します。

＜適用基準＞

アンロード機能を持つ，プラグインが提供している抽象データ型がある表をリバランスする場合に指定します。

＜規則＞

1. アンロード機能を持つ，プラグインが提供している抽象データ型がある表の場合は，reld_func 文を必ず指定してください。
2. reld_func 文は，表の列に定義されている抽象データ型の数分指定できます。
3. reld_func 文を指定するときは，対で unld_func 文も指定してください。

(1)　形式

```
reld_func type=〔認可識別子.〕抽象データ型名,
          func=関数名（引数の型〔,引数の型…〕）
```

(2)　説明

(a)　type=〔認可識別子.〕抽象データ型名

抽象データ型の認可識別子，及び抽象データ型名を指定します。

＜規則＞

1. 認可識別子を省略した場合，抽象データ型を定義したユーザの認可識別子（通常は MASTER）が仮定されます。
2. 認可識別子，及び抽象データ型の名称に小文字，空白を含む場合，引用符（”）で囲んでください。

(b)　func=関数名（引数の型〔,引数の型...〕）

コンストラクタ関数の名称，引数の型を指定します。名称，引数の型については，各プラグインのマニュアルを参照してください。

関数名

コンストラクタ関数の名称を指定します。

引数の型

コンストラクタ関数の引数のデータ型を指定します。

引数のデータ型の形式については，「unld_func 文（コンストラクタパラメタ逆生成関数の情報の記述）」の引数の型を参照してください。ただし，抽象データ型名は指定できません。

10.3.9　report 文（処理結果ファイルの情報の記述）

pdrbal の処理結果を出力するファイルの情報を指定します。

＜適用基準＞

pdrbal 実行ホストの，表「コマンド実行環境でファイルを出力するディレクトリ」に示すディレクトリ以外に処理結果ファイルを作成したい場合に指定してください。

＜規則＞

1. report 文を省略した場合，pdrbal を実行したホストの，表「コマンド実行環境でファイルを出力するディレクトリ」に示すディレクトリ下に処理結果ファイルが作成されます。
2. pdrbal が終了するまで，処理結果ファイルの内容は保証されません。
3. pdrbal がリターンコード=8 で終了した場合，「移動した行データの情報」以降の情報が出力されないことがあります。出力された場合は，その情報はロールバック後（エラーが発生する直前のコミット時）のものとなります。

(1)　形式

```
report 処理結果ファイル名
```

(2)　説明

(a)　処理結果ファイル名

～＜パス名＞

処理結果を出力するファイルの名称を，絶対パス名で指定します。

処理結果ファイルは，pdrbal を実行したホストのファイル名を指定してください。

(3)　システム定義の pd_tmp_directory オペランド，環境変数 TMP，及び report 文省略時に作成される処理結果ファイルの名称

作成される処理結果ファイルの名称は次の形式で作成されます。

```
%PDDIR%¥tmp¥REPORT-表識別子--一意となる文字列
```

表が TABLE1 の場合，作成される処理結果ファイルの名称は次のようになります。

```
%PDDIR%¥tmp¥REPORT-TABLE1-aaaa00001
```

なお，表名称に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

(4)　出力形式

pdrbal の処理結果ファイルの出力形式を次に示します。

```
pdrbal VV-RR(Object Option) *** DB REBALANCE *** yyyy-mm-dd HH:MM:SS [1]
------------------------------------------------------------------------

*** Table rebalance processing list ***

 Schema name  : tanaka  [2]
 Table name   : rev_table  [3]
 Table status : completed  [4]

 ** Moved data information **
 [5]  [6]          [7]                          [8]           [9]
 No. Server   Table rdarea name               Remove row     Insert row
   1 sds      user01                          1,000,000              0
   2 sds      user02                          2,000,000              0
   3 sds      user03                                  0      3,000,000
                                    Total     3,000,000      3,000,000
                                               [10]           [11]
 ** Rebalance information **
 [12] [13]         [14]                       [15]
 No. Server   Table rdarea name               Status
   1 sds      user01                          finished
   2 sds      user02                          finished
   3 sds      user03

 *** Exclusive execute information ***

  Start time : 2000-05-05 11:12:13  [16]
  End   time : 2000-05-05 19:20:21  [17]

 pdrbal terminated, return code=0  [18]
```

[説明]

1. 処理結果のヘッダ情報

   VV-RR：バージョン番号

   yyyy-mm-dd HH:MM:SS：pdrbal を開始した日付と時刻

2. スキーマ名

3. 表識別子

4. 表のリバランス状態

   completed：リバランス完了

processing：リバランス中

5. 移動元 RD エリア又は移動先 RD エリアの通番
6. 移動元 RD エリア又は移動先 RD エリアがあるサーバ名
7. 移動元 RD エリア名又は移動先 RD エリア名
8. 移動元 RD エリアから移動した（削除した）行数
9. 移動先 RD エリアに移動した（追加した）行数
10. 移動元 RD エリアから移動した（削除した）行数の合計
11. 移動先 RD エリアに移動した（追加した）行数の合計
12. 表格納用 RD エリアの通番
13. 表格納用 RD エリアがあるサーバ名
14. 表格納用 RD エリア名
15. RD エリアのリバランス状況
    finished：リバランス完了
    processing：リバランス中
    error：エラー終了
    空白：追加した RD エリア
16. リバランスを開始した日付と時刻
17. リバランスが終了した日付と時刻
18. pdrbal のリターンコード

注

項番 5〜9 及び 12〜15 は，表定義で指定した表格納用 RD エリア，次に ALTER ADD RDAREA で追加した RD エリアの順で出力します。

10.3.10　option 文（オプション機能の記述）

pdrbal のオプション機能を指定します。

＜適用基準＞

共用モードの pdrbal が動作できるかどうかをチェックする間隔を，変更する場合に指定します。

＜規則＞

1. option 文は一つだけ指定できます。

(1)　形式

```
option 〔check_itv=実行可否チェック間隔〕
```

(2) 説明

(a) check_itv=実行可否チェック間隔

～＜符号なし整数＞((10～180000))《10》

共用モード（-k share）の場合，pdrbal は処理対象表をアクセスする UAP があるかどうかをチェックし，アクセスする UAP がないときにデータを移動します。実行可否チェック間隔には，このチェック間隔をミリ秒単位で指定します。

占有モード（-k exclusive）の場合は指定できません。

＜適用基準＞

通常はデフォルト値で運用します。

pdrbal の負荷が大きく，オンライン業務に影響がある場合は，デフォルト値よりも大きい値を指定することで，pdrbal が UAP の動作をチェックする監視間隔が長くなり，オンライン業務を優先できます。

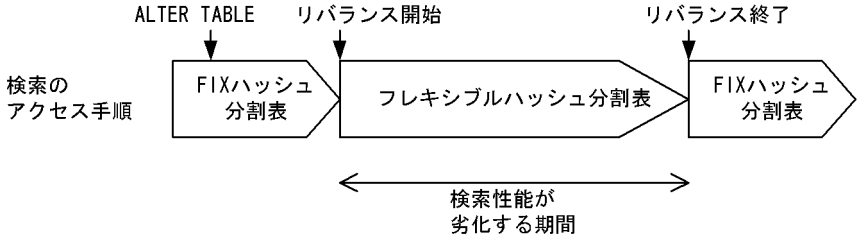
10.4　リバランス中の FIX ハッシュ分割表に対して SQL を実行する場合の注意事項

FIX ハッシュ分割表の場合は，SQL のアクセス手順を決定するとき，ハッシュキー値によってデータを格納又は検索する RD エリアを特定できます。しかし，HASHA～HASHF の FIX ハッシュ分割表に対して RD エリアの追加をした場合，リバランスが完了するまで，検索のアクセス手順がフレキシブルハッシュ分割表とした検索のアクセス手順に変更になります。このようにフレキシブルハッシュ分割表の検索アクセス手順に変更になると，次の SQL の検索性能が劣化します。

- 分割キーに条件を指定した検索
- GROUP BY 句の列が分割キーをすべて含む検索

リバランス処理と検索性能が劣化する期間の関係を次の図に示します。この場合，リバランス中は SQL を実行するためのプロセス数が増加します。

図 10–6　リバランス処理と検索性能が劣化する期間の関係



なお，フレキシブルハッシュ分割表の場合は，リバランス処理によって検索のアクセス手順が変更になることはありません。

10.5　リバランス中の SQL 実行可否

リバランス中の SQL 実行可否を次の表に示します。

表 10–10　リバランス中の SQL 実行可否

SQL 文	リバランス処理中		リバランス途中，及びリバランス停止中
	共用モード	占有モード	
SELECT 文	○	×	○
UPDATE 文	○※1	×	○※1
INSERT 文	○※2	×	○※2
DELETE 文	○	×	○
LOCK 文	○	×	○
PURGE TABLE 文※3	×※4	×	○
DROP TABLE	×※4	×	○
CREATE INDEX	×※4	×	○
DROP INDEX	×※4	×	○
ALTER TABLE	×※4	×	×

（凡例）

○：実行できます。

×：実行できません。

注※1

ユニークキーインデクスを構成する列の更新はできません（エラーとなります）。

注※2

リバランス対象表にユニークキーインデクスが定義されている場合は実行できません（エラーとなります）。

注※3

RD エリア追加後，pdrbal 実行前に PURGE TABLE 文又は RD エリアの再初期化をした場合，リバランス初期化処理がされていないため，追加した RD エリアに対する格納処理は実行されません。

また，pdrbal がリターンコード 0 で終了する前に，PURGE TABLE 文又は RD エリアの再初期化をした場合，リバランス完了処理がされていないため，FIX ハッシュ分割表の SQL 検索性能が劣化したままとなります。このような場合には，PURGE TABLE 文又は RD エリアの再初期化をした後の状態で，pdrbal を実行してください。これによって，リバランス初期化処理，又はリバランス完了処理が実行されます。

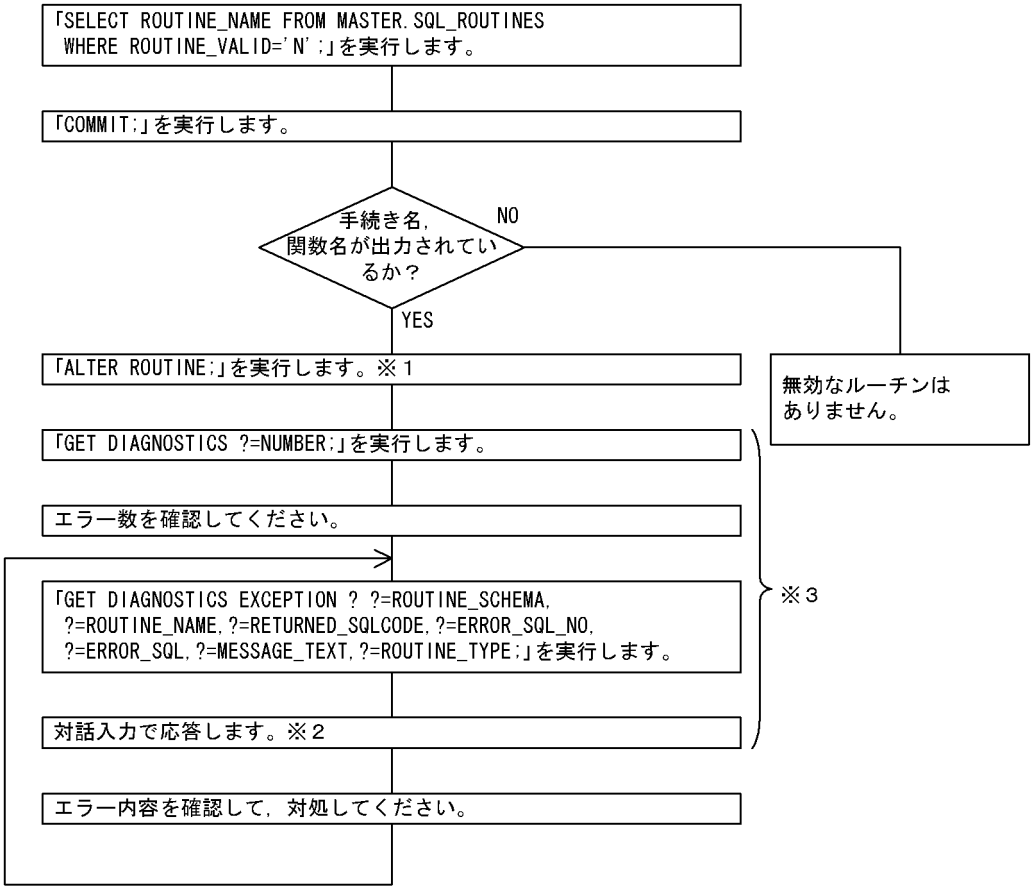
注※4

回復不要 FES を適用している場合，pdrbal 実行中に定義系 SQL や PURGE TABLE 文が実行できてしまうことがあります（回復不要 FES を適用していない場合は，不当に定義系 SQL や PURGE TABLE 文が実行されないように，フロントエンドサーバが排他資源を確保して防いでいます）。このため，pdrbal 実行中は定義系 SQL や PURGE TABLE 文を実行しないでください。なお，占有モードで pdrbal を実行する場合は，バックエンドサーバで表に排他制御が掛かっているため，定義系 SQL は排他待ちとなります。

10.6 注意事項

1. pdrbal のリターンコードを次に示します。
   0：正常終了（リバランス完了）
   4：正常終了（リバランスの途中）
   8：異常終了
2. リバランス中の表に対して定義系 SQL は実行できません。
3. 占有モード（-k exclusive）の場合，リバランス中の表やインデクスは，ほかの UAP やユティリティからアクセスできません。
4. リバランス表を基にリストを作成している場合，リバランス中はリストを使用した検索結果が不正になるときがあります。この場合，リストを使用して検索する前に，リストを再作成してください。
5. 共有モードでのリバランス中に，UAP で行識別子を使用した検索，更新，又は削除をする場合，行識別子の取得と実際のデータ操作との間にコミットを入れると，結果不正になることがあります。
6. リバランス表に表格納用 RD エリアの数と一致しないインデクス格納用 RD エリアがある場合（サーバ内分割していない非分割キーインデクスの場合），pdrbal は実行できません。このような表をリバランスするためには，サーバ内分割していない非分割キーインデクスをいったん削除し，pdrbal を実行してください。そして，リバランス終了後に，再度インデクスを定義してください。なお，このようなインデクスはリバランス中には定義できません。リバランス表に非分割キーインデクスを定義する場合は，同一サーバ内の表格納用 RD エリアと同じ数のインデクス格納用 RD エリアに格納するようにしてください。
7. pdrbal が出力するデータベースの更新ログは，HiRDB Datareplicator でのデータ連動機能の対象とはなりません。
8. リバランスをするデータのインデクス削除，作成では，-n オプションの指定に関係なく，必ずグローバルバッファを使用します。そのため，十分に余裕のあるグローバルバッファを割り当てた環境で pdrbal を実行してください。特に，データの移動元となる RD エリア側では，インデクスごとにグローバルバッファを割り当てると効果的です。グローバルバッファに余裕がない場合，バッファ競合が発生して入出力 BUSY 状態となり，性能が悪くなります。
9. 更新凍結状態のユーザ LOB 用 RD エリアを含む表をリバランスすると，pdrbal がエラー終了します。
10. pdrbal は，リバランス初期化処理時及びリバランス完了処理時に SQL オブジェクトが無効となっている，データベースにあるすべての手続き，関数をリコンパイルします。そのため，SQL オブジェクトが無効で，かつリコンパイルができない手続き，関数がデータベースにある場合，pdrbal がリコンパイルできなくなるため，エラーとなります。エラーとなった場合，エラーメッセージを確認し，対策してから ALTER PROCEDURE 又は ALTER ROUTINE で手続き，関数をリコンパイルしてください（pdrbal を再実行しても，リコンパイルはされません）。
    SQL オブジェクトが無効で，かつリコンパイルできない手続き，関数を pdrbal 実行前に調べたい場合，SQL Executer を使用します。SQL オブジェクトが無効で，かつリコンパイルできない手続き，関数の調査方法を次の図に示します。

図 10–7　SQL オブジェクトが無効で，かつリコンパイルできない手続き，関数の調査方法



注　ALTER ROUTINE以降の手順は，DBA権限を持っているユーザが実行してください。
注※１　ここで実行するALTER ROUTINEはエラーとなります。
注※２　1～エラー数まで，順に応答します。
注※３　GUI版のSQL Executerを使用する場合は「GET DIAGNOSTICS;」を実行してください。

11. pdrbal で使用するファイルの媒体と 1 ファイル 2 ギガバイト以上のファイルの利用可否を次の表に示します。

表 10–11　pdrbal で使用するファイルの媒体と 1 ファイル 2 ギガバイト以上のファイルの利用可否

ファイル	媒体		1 ファイル 2 ギガバイト以上のファイルの利用可否
	ファイル	固定長又は可変長ブロックテープ	
制御情報ファイル	○	×	○
インデクス情報ファイル	○	×	○
ソート用ワークファイル	○	×	○
処理結果ファイル	○	×	○

（凡例）

○：利用できます。

×：利用できません。

12. リバランスする表が改竄防止表で，かつその改竄防止表がデータ未完状態の場合，pdrbal は実行できません。

13. HiRDB/パラレルサーバの場合で，かつ共有モードで pdrbal を実行する場合，回復不要 FES を適用していないフロントエンドサーバはすべて稼働している必要があります。

14. 表が検査保留状態の場合（ディクショナリ表 SQL_TABLES の CHECK_PEND 列又は CHECK_PEND2 列の値が C（検査保留状態）），pdrbal は実行できません。

15. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdrbal の入力ファイルには，BOM を持つファイルを使用できることがあります。pdrbal での BOM を持つファイルの使用可否を次の表に示します。なお，pdrbal の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdrbal で出力されるファイルには BOM は付きません。

表 10–12　pdrbal での BOM を持つファイルの使用可否（UTF-8 の場合）

制御文	入力ファイル	BOM を持つファイルの使用
–	制御情報ファイル	○
index	インデクス情報ファイル	×

（凡例）

○：使用できます。

×：使用できません。

–：該当しません。

16. pdrbal はリバランス対象表の分割数によってハッシングします。そのため，表格納用 RD エリアに重複した RD エリアが存在すると，重複した RD エリアに行データが偏って格納される場合があります。

17. 制御文で指定しない場合のファイル出力先ディレクトリ

pdrbal 実行時，制御情報ファイルでファイル出力先を指定しない場合，次の表に示すディレクトリにファイルを出力します。

表 10–13　pdrbal がファイルを出力するディレクトリ

<table>
<tr><td rowspan="3">制御文※1 の指定</td><td colspan="3">システム定義の pd_tmp_directory オペランドの指定</td></tr>
<tr><td rowspan="2">あり</td><td colspan="2">なし</td></tr>
<tr><td colspan="2">環境変数 TMP※2 の指定</td></tr>
<tr><td>あり</td><td>なし</td></tr>
<tr><td>あり</td><td colspan="3">制御文に指定したディレクトリ又はファイル</td></tr>
<tr><td>なし</td><td>pd_tmp_directory に指定したディレクトリ</td><td>TMP に指定したディレクトリ</td><td>%PDDIR%¥tmp ディレクトリ</td></tr>
</table>

注※1　index 文，idxwork 文，又は sort 文

注※2　システム環境変数の指定

18. ON ROW EXISTS 指定の列を追加した表の場合，pdrbal は実行できません。

10.7　エラー時の対処方法

1. pdrbal のリターンコードが 8 の場合，エラーメッセージを参照して回復した後に，再度 pdrbal を実行してください。
2. pdrbal 実行中に KFPL33003-I メッセージ（STATUS=END）が出力された場合，表のリバランス処理は正常に終了しています。したがって，このメッセージ出力以降にエラーが発生しても pdrbal を再実行する必要はありません。
3. 占有モードでのインデクス一括作成処理でエラー（リターンコード=8）となった場合，エラー要因を取り除いた後，エラーとなったインデクスのインデクス情報ファイルを使用して pdrorg のインデクスの一括作成（-k ixmk）を実行してください。その後，pdrbal を実行してください。
4. 占有モードで圧縮表のリバランスを実行中，KFPL15240-E メッセージ（メモリ不足）が出力された場合，共有モードで必要なメモリ容量を確保できるときは，共有モードで再実行してください。実行に必要なメモリ容量については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
5. pdrbal 実行中にロールバックが発生した場合，ログ取得方式（-l オプション）によって対処方法が異なります。pdrbal 実行中にロールバックが発生した場合の対処方法を次の表に示します。

表 10–14　pdrbal 実行中にロールバックが発生した場合の対処方法

ログ取得方式（-l オプション）		pdrbal の処理内容	
		リバランス	インデクスの一括作成（占有モードの場合）
a	データベースの状態	ロールバックされた行データは移動していない状態です。ロールバック直前までの行データは移動完了しています。	インデクスは未完状態となります。
	対処方法	pdrbal を再実行してください。	pdrorg のインデクスの一括作成（-k ixmk）を実行してください。このとき，エラーとなったインデクス情報ファイルを使用してください。
n	データベースの状態	保証されません。	保証されません。
	対処方法	バックアップから回復した後，pdrbal を再実行してください。	該当するインデクス格納用 RD エリアを再初期化して，pdrorg のインデクスの再作成（-k ixrc）を実行してください。

11 空きページ解放ユティリティ（pdreclaim）

この章では，オンライン中に空きページ（使用中空きページ）の解放ができる空きページ解放ユティリティ（pdreclaim）について説明します。

11.1 概要

11.1.1 空きページとは

pdreclaim について説明する前に，ページ及びセグメントについて説明します。ページ及びセグメントには次の表に示す状態があります。

表 11–1 ページ及びセグメントの状態

ページ及びセグメントの状態	説明
未使用ページ	まだ割り当てられていないページです。
使用中空きページ	データの削除※1によって，データが格納されていないページです。
使用中ページ	データが格納されていて，データを追加できる空き領域があるページです。 空き領域の再利用機能を使用している表の場合は，データの削除※1によってページ内の空き領域が使用できないために，データを追加できなかったページを含みます。
使用中満杯ページ	データが格納されていて，データを追加できる空き領域がないページです。 空き領域の再利用機能を使用していない表及びインデクスの場合は，データの削除※1によってページ内の空き領域が使用できないために，データを追加できなかったページを含みます。
使用中セグメント※2	表又はインデクスのデータを格納しているセグメントです。データが満杯でセグメント内にデータを追加できないセグメントを満杯セグメント，それ以外の使用中セグメントを**空きありセグメント**といいます。 空きありセグメントのうち，データの削除でセグメント内の全ページが空きページ（使用中空きページ又は未使用ページ）のセグメントを使用中空きセグメントといいます。
未使用セグメント	使用されたことがないセグメントです。このセグメントは RD エリア内のすべての表（又はインデクス）が使用できます。
空きセグメント	データを格納していないセグメントです。使用中空きセグメントと未使用セグメントは空きセグメントになります。

注※1

データの削除を実行したトランザクションが COMMIT するまで，データの削除によって発生した空き領域は使用できません。

注※2

使用中セグメントを使用できるのは，このセグメントにデータを格納した表又はインデクスだけです。ほかの表又はインデクスはこのセグメントを使用できません。

11.1.2　pdreclaim の概要

データベースの運用を続けると，データの追加，更新，及び削除を繰り返すことによって，データベース中のデータ，インデクスの配置に乱れが生じ，格納効率の低下や性能劣化を引き起こします。この場合，データベース再編成ユティリティ（pdrorg）を実行して，データ，インデクスの配置の乱れを修復しますが，pdrorg 実行中の表は参照，及び更新ができなくなるため，pdrorg を実行するためにオンラインサービスを停止する時間を確保する必要があります。

これに対して，空きページ解放ユティリティ（pdreclaim）は，オンラインサービス中に，pdrorg での再編成処理の一部である使用中空きページの解放ができます（pdreclaim は，内部的に pdrorg を実行して動作します)。なお，表を排他モード（EX）で排他する AP や，ユティリティを実行しているときは，使用中空きページの解放はできません。また，空きセグメントの解放は，AP やユティリティが該当する RD エリアをアクセスしていない状態のときだけ実行できます。

pdreclaim は，次のような場合に実行してください。

- オンラインサービスを停止できないシステムでの，データベースの再編成をしない運用の場合※
- 格納効率の低下を抑えて，データベースの再編成の実行間隔（運用サイクル）を延ばしたい場合

注※

pdreclaim でデータベースの再編成をしない運用をする場合の制限については，「pdrorg との機能差」を参照してください。

11.1.3　pdreclaim の機能

pdreclaim には，次の二つの機能があります。

- **表の使用中空きページ解放**
- **インデクスの使用中空きページ解放**

-j 又は-a オプションなしで pdreclaim を実行すると，使用中空きページの解放をします。その結果，使用中空きセグメントが発生した場合に，-j オプションありの pdreclaim を実行すると，使用中空きセグメントの解放をします。また，-a オプションありの pdreclaim を実行した場合は，使用中空きページの解放，及び使用中空きセグメントの解放をします。

-j 又は-a オプション有無による pdreclaim の実行可否を次に示します。

pdreclaim のオプション指定	使用中空きページの解放	使用中空きセグメントの解放
-j 又は-a オプションなし	○	×
-j オプションあり	×	○
-a オプションあり	○	○

（凡例）

○：実行します。

×：実行しません。

したがって，使用中空きページの解放をしないで，使用中空きセグメントの解放だけをすることはできません。ただし，次の操作をした場合，-j 又は-a オプションなしの pdreclaim で使用中空きページの解放をしなくても使用中空きセグメントが発生するため，-j オプションありの pdreclaim で使用中空きセグメントの解放ができます。

- 表に排他が掛かった状態で，データを削除した場合
- pdload（option 文の nowait オペランドに yes を指定）が，データロード中に異常終了してロールバックした場合

- **表の使用中空きページ解放の効果**

  表の使用中空きページ解放をすると，次の効果があります。

  - 全件検索性能の改善

    全件検索実行時に参照していた使用率 0%のページが未使用ページとなるため，参照をスキップするようになり，全件検索の性能が向上します。

  - 分岐行の INSERT 文，UPDATE 文実行時のエラー回避

    未使用ページがなくなると，分岐行の INSERT 文，UPDATE 文はエラー（KFPA11756-E）となりますが，使用率 0%のページを未使用ページにすることでエラーを回避できます。

  - INSERT 文，UPDATE 文実行時のページコンパクション処理の回避

    すべての使用中空きページについてページコンパクションを一括して実行するため，INSERT 文，UPDATE 文の延長で実行されるページコンパクションを回避できます。

- **インデクスの使用中空きページ解放の効果**

  インデクスの使用中空きページ解放をすると，次の効果があります。

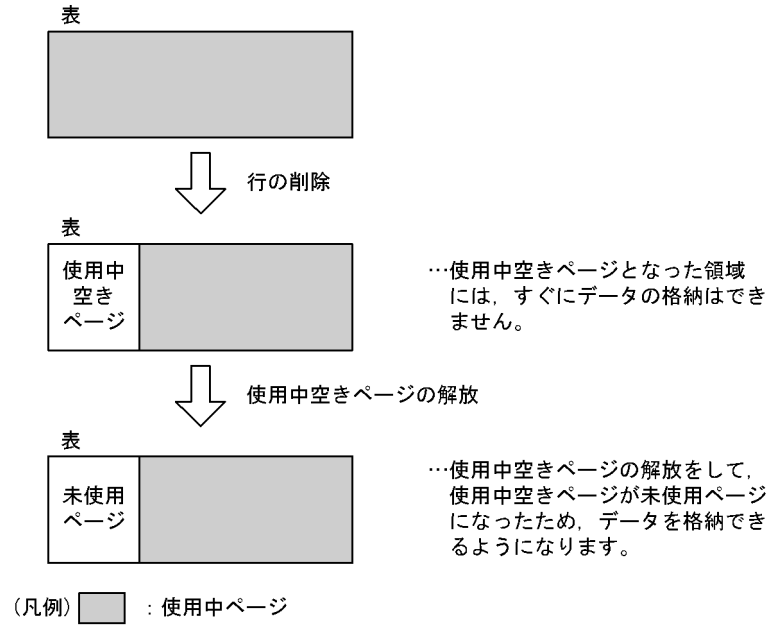
  - 使用中空きページがあるのにエリア不足となる現象を回避できます。
  - インデクス検索実行時に，使用中空きページを参照するために発生する性能劣化を回避できます。

インデクスの使用中空きページ解放実行後の，未使用ページ割り当て処理は，使用中空き領域サーチ処理で解放した未使用ページを優先して割り当てます。使用中セグメント内の未使用ページを優先して割り当てるため，領域が有効活用できて，使用中空き領域サーチ処理時のオーバヘッドも削減できます。

(1)　表の使用中空きページ解放

表の使用中空きページ解放の概要を次の図に示します。

図 11–1　表の使用中空きページ解放の概要



表の使用中空きページ解放では，使用中空きセグメントの解放，使用中空きページの解放，及びページコンパクションの実行有無を選択できます。なお，LOB 列，及び LOB 属性の抽象データ型列に対して，表の使用中空きページ解放は適用できません。

- **使用中空きセグメントの解放，及び使用中空きページの解放**

データを大量に削除した場合，格納データが 1 件もないのに割り当てられたままとなる使用中空きページが発生します。SEGMENT REUSE 表（CREATE TABLE 又は ALTER TABLE で SEGMENT REUSE オプションを指定した表）でない場合，HiRDB は未使用ページ，及び未使用セグメントを使用する格納方式となっています。この新規領域が確保できなくなると使用中空きページを再利用しますが，再利用する場合は使用中ページ内の空き領域をサーチするオーバヘッドが発生し，新規領域確保に比べて性能が劣化します。また，SEGMENT REUSE 表であっても，使用中空きページがある状態では全件検索などでむだなページを検索していることになります。さらに，再利用できるのは該当する使用中セグメントの表，又はインデクスに限ります。

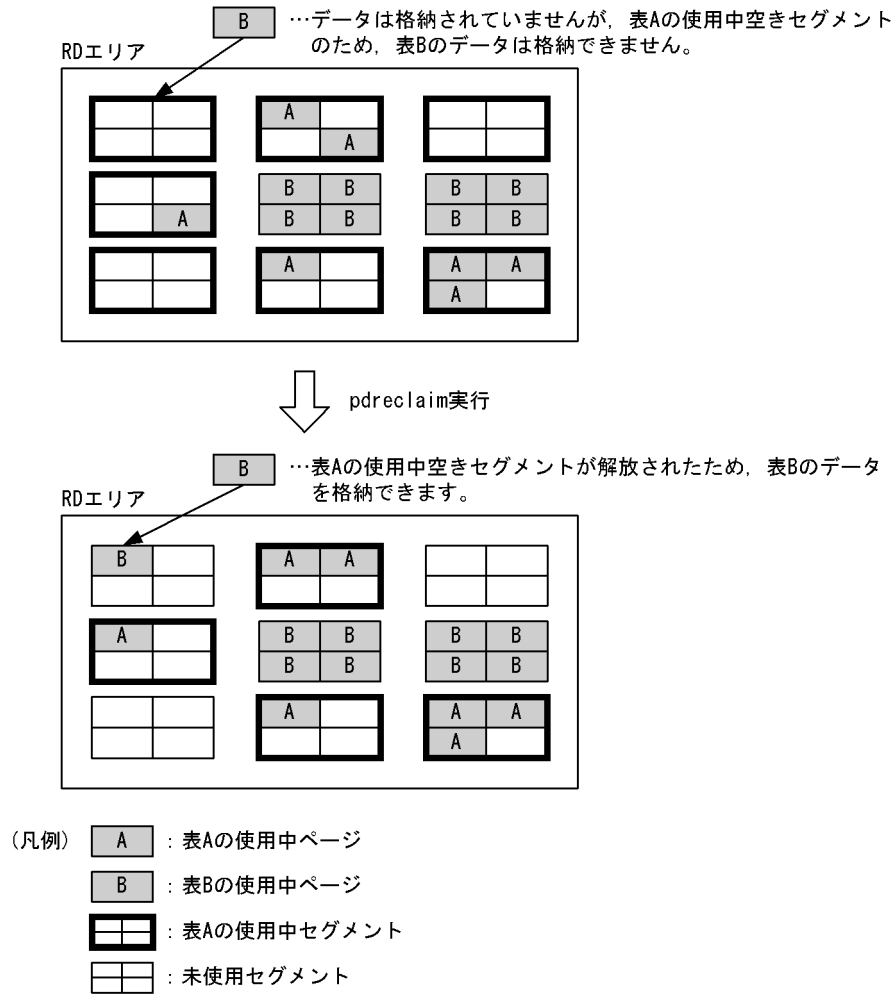
pdreclaim を実行すると，すぐには再利用されない使用中空きページが解放されるため，次の効果があります。

1. すべてのデータが削除されたセグメントは未使用セグメントとなり，すべての表及びインデクスで再利用できるようになります。
2. すべてのデータが削除されたページは未使用ページとなるため，該当する使用中セグメントの表で再利用できます。特に，SEGMENT REUSE 表の場合，再利用するページをサーチするオーバヘッドが削減されるため，効果が大きいです。

1.の効果として，同一 RD エリアに複数の表を格納している場合，表 A がデータを格納していない使用中空きセグメントを保持しているために，表 B が新たな未使用セグメントを確保できないで RD エリアの容

量不足となることを回避できます。同一 RD エリアに複数の表を格納している場合の pdreclaim の効果を次の図に示します。

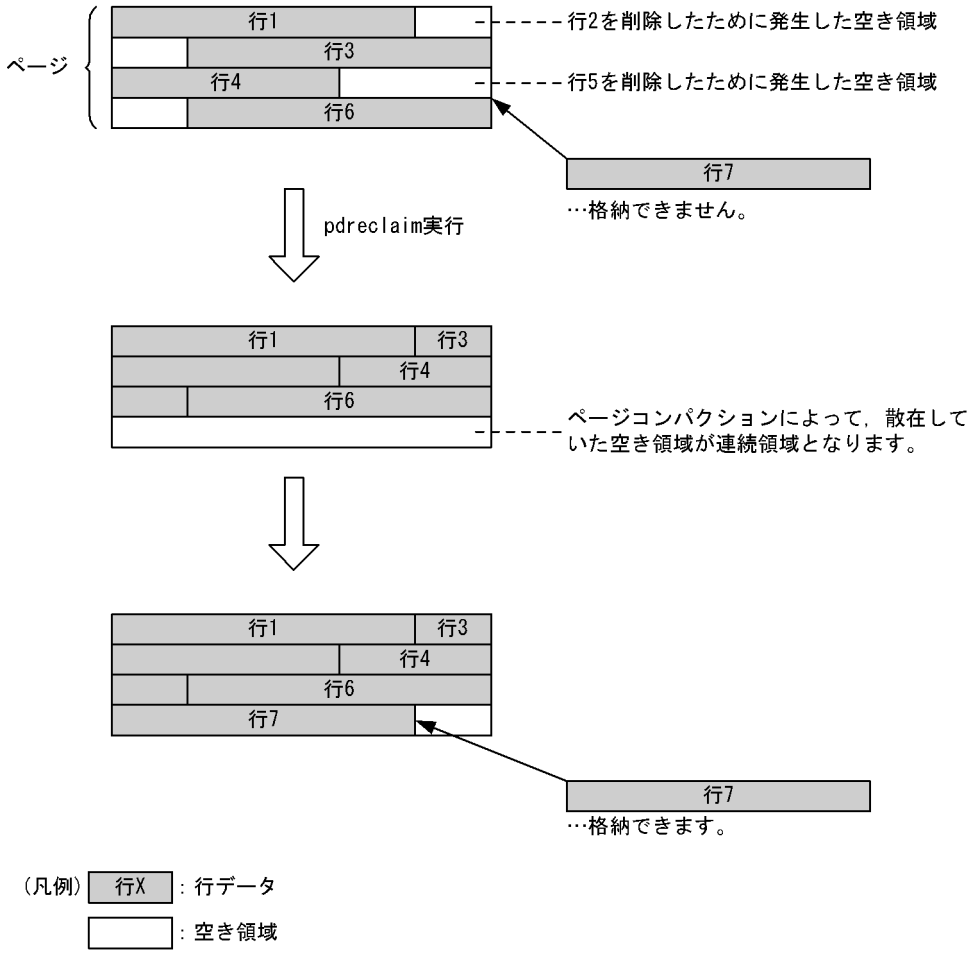
図 11–2　同一 RD エリアに複数の表を格納している場合の pdreclaim の効果



- **表のページコンパクション**

表のページコンパクションの概要を次の図に示します。

図 11–3　表のページコンパクションの概要



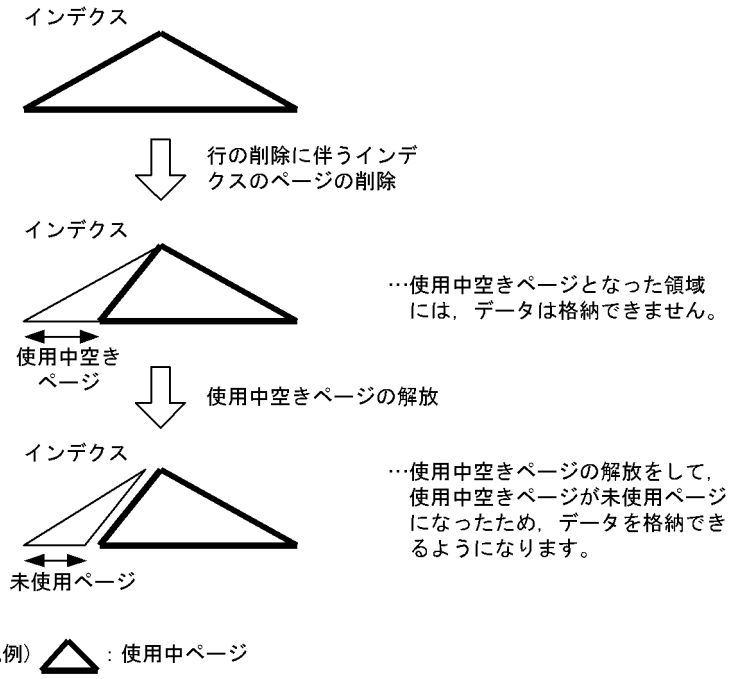
〔説明〕

pdreclaim 実行前の状態では，削除された行 2 及び行 5 より長いデータは，行 2 及び行 5 の削除で発生した空き領域をそのまま使用して格納することはできません。この場合，空き領域を連続領域にすれば格納できます。この散在している空き領域を連続領域に詰め替える動作を，**ページコンパクション**といいます。ページコンパクションは，SQL の延長でも実行されます。ただし，SQL の延長で実行される場合は，オンライン性能が劣化するため，事前に pdreclaim でページコンパクションをしておくことで，オンライン中の性能劣化を回避できます。なお，ページコンパクションが発生すると，入出力処理の負荷が高くなります。

(2)　インデクスの使用中空きページ解放

インデクスの使用中空きページ解放の概要を次の図に示します。

図 11–4　インデクスの使用中空きページ解放の概要



インデクスの使用中空きページ解放では，使用中空きセグメントの解放，又は使用中空きページの解放のどちらを行うか選択できます。なお，プラグインインデクスに対して，インデクスの使用中空きページ解放は適用できません。

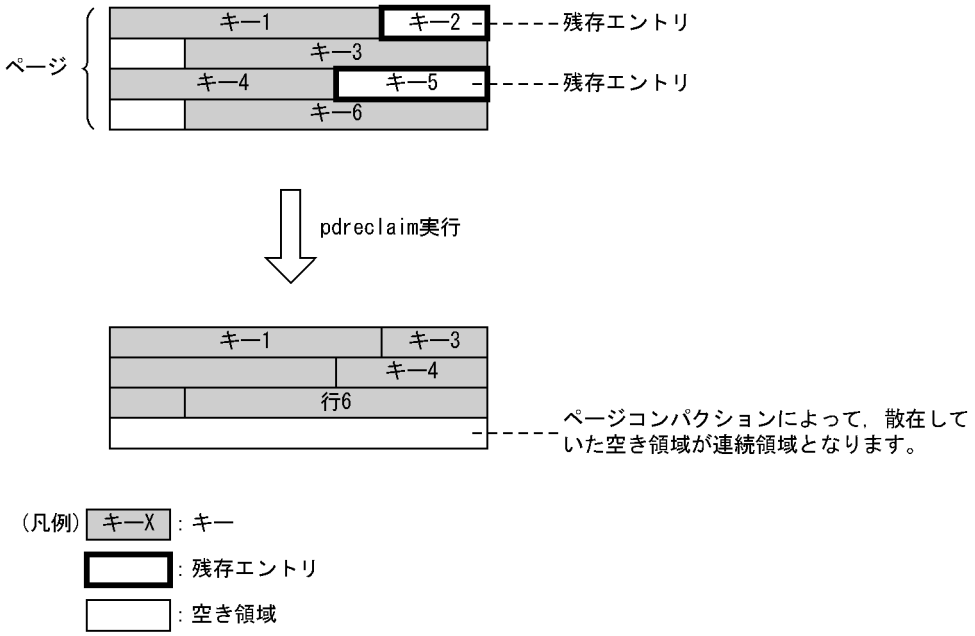
- **使用中空きセグメントの解放，及び使用中空きページの解放**

インデクスの場合も表と同様に，大量にデータを削除すると，再利用できない使用中空きページが発生します。特に，過去のキー値の小さいデータを削除して使用中空きページになった領域は，再度同じキー値が格納された場合に備えて確保状態（ポインタでつながったまま）が続くため，新規のキー値の大きいデータを追加するときに使用中空きページの再利用もされません。このため，単調に格納データのキー値が増加し，過去のデータを削除する運用形態では，再利用できない使用中空きページが大量に発生します。このインデクスに対して pdreclaim を実行すると，使用中空きページを解放して未使用ページとするため，大きいキー値のデータを格納できるようになります。また，下位ページポインタを持たない中間ページも未使用ページにできます。

- **インデクスのページコンパクション**

インデクスのページコンパクションの概要を次の図に示します。

図 11–5　インデクスのページコンパクションの概要



〔説明〕

pdreclaim 実行前の状態では，削除されたキー 2 及びキー 5 は残存エントリとして残っています。この残存エントリを解放して，連続領域に詰め替える動作をページコンパクションといいます。ページコンパクションは SQL の延長でも実行されます。ただし，SQL の延長で実行される場合は，オンライン性能が劣化するため，事前に pdreclaim でページコンパクションを実行しておくことで，オンライン中の性能劣化を回避できます。なお，ページコンパクションが発生すると，入出力処理の負荷が高くなります。

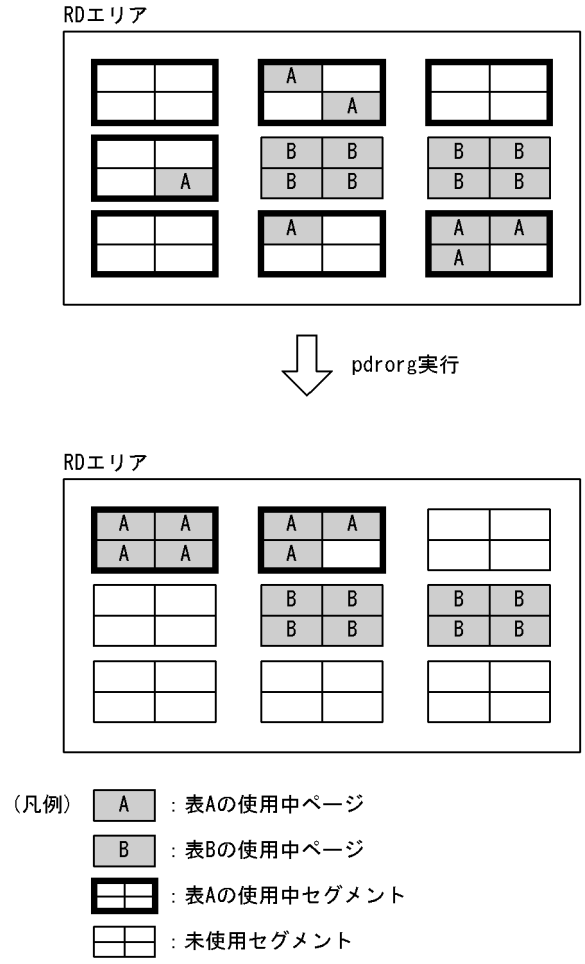
また，残存エントリがあると，排他待ち又はデッドロックが発生することがあります。pdreclaim でページコンパクションを実行しておくことで，排他待ち又はデッドロックも回避できます。

インデクスの残存エントリ，及びユニークインデクスの残存エントリについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

11.1.4　pdrorg との機能差

オンラインを停止する時間を確保できる場合，pdrorg で再編成をする方がよいです。理由は，pdreclaim より pdrorg の再編成の方がデータを再配置する機能が多いためです。pdrorg を実行すると，次の図のように再配置されるため，より多くの使用中空きページ，及び使用中空きセグメントが解放されます。

図 11–6　pdrorg を実行した場合の使用中空きページ，及び使用中空きセグメントの解放



ただし，pdreclaim は，pdrorg と比較して次の利点があります。

- UAP から，ユティリティ実行中の表及びインデクスを参照，更新できます。
- アンロードデータファイル，ワークファイルなどが不要です。

したがって，次の条件をすべて満たす場合は，pdreclaim の実行を推奨します。

1. 可変長文字列の更新がない（分岐行が発生しない）。
2. 繰返し列の要素数の変更がない（分岐行が発生しない）。
3. ナル値を実データ，又は実データをナル値に変更しない（分岐行が発生しない）。
4. クラスタキーインデクスを定義していない（クラスタキーを意識して配置する必要がない）。
5. LOB 列，及び LOB 属性の抽象データ型列がない。
6. データを削除する場合，物理的に格納位置が近いデータを一度に大量に削除する（使用中空きページとなる）。又は，SEGMENT REUSE 表である。

クラスタキーインデクスを定義していない FIX 属性の SEGMENT REUSE 表は，pdreclaim で運用するのに最適な表となります。また，上記の条件を満たさない表でも，pdreclaim を実行することで使用中空きペー

ジの解放はされるため，使用中ページ内の空き領域サーチ処理で発生する極端な性能劣化は回避できます。これによって，毎週末に pdrorg を実行していた運用を，月末にだけ実行する運用に変えることもできます。

インデクスについては，削除したキー値を再度登録することがない場合は，pdreclaim での運用は最適です。また，キー値の更新，削除が多い場合は，pdrorg でのインデクスの再編成をしなくても，インデクス格納用 RD エリアの容量不足が発生しにくくすることができます。

pdrorg，又は pdreclaim のどちらを実行するかについては，データベース状態解析ユティリティの結果を参照して決めてください。ページ使用率が 0%のページが大量にある場合は pdreclaim を実行してください。CREATE TABLE の PCTFREE で指定した値と，大きく異なるページ使用率のページが大量にある場合は，pdrorg を実行してください。

11.1.5　実行環境

1. pdreclaim は，HiRDB が稼働中のときだけ実行できます。
2. pdreclaim は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 処理対象 RD エリアは，オープン状態であれば pdreclaim は実行できます。なお，-j オプションを指定する場合は，RD エリアをオープン状態，かつコマンド閉塞状態にすることをお勧めします。pdreclaim の実行可否については，「コマンド実行時の RD エリアの状態」を参照してください。
4. 同一の表，インデクスに対して，pdreclaim を複数同時には実行できません（エラーとなります）。なお，表，インデクスを格納している RD エリア単位に同時実行することはできます。

11.1.6　実行者

- ディクショナリ表を処理対象とする場合
  DBA 権限が必要となります。
- そのほかの場合
  DBA 権限（監査証跡表が処理対象の場合は監査権限），又は処理対象表に対する INSERT 権限，及び DELETE 権限が必要です。

11.1.7　pdreclaim と他機能との同時実行

pdreclaim と他機能との同時実行性について説明します。

(1)　pdreclaim と運用コマンドの同時実行可否

pdreclaim と運用コマンドの同時実行可否を次の表に示します。

表 11–2　pdreclaim と運用コマンドの同時実行可否

運用コマンド	条件	pdreclaim		
		-j 及び-a オプションなし（空きページ解放）	-j オプションあり（空きセグメント解放）	-a オプションあり（空きページ及び空きセグメント解放）
pdclose	−	×	×	×
pddbfrz	−	×	×	×
pddbls	-a	○	×	○
	-a 以外	○	○	○
pdhold	-bu	○	○	○
	-bu 以外	×	×	×
pdopen	−	×	×	×
pdrels	RD エリアが更新可能バックアップ閉塞	○	○	○
	RD エリアが更新可能バックアップ閉塞以外	×	×	×

（凡例）

○：同時実行できます。

×：同時実行できません（排他待ちとなります）。

−：該当しません。

(2)　pdreclaim とユティリティの同時実行可否

pdreclaim とユティリティの同時実行可否を次の表に示します。

表 11–3　pdreclaim とユティリティの同時実行可否

ユティリティ	条件	pdreclaim					
		-j 及び-a オプションなし（空きページ解放）		-j オプションあり（空きセグメント解放）		-a オプションあり（空きページ及び空きセグメント解放）	
		異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）
pdload	表単位のデータロード	○	×	×	×	○	×
	RD エリア単位のデータロード	×	×	×	×	×	×

ユティリティ	条件	pdreclaim					
		-j 及び-a オプションなし（空きページ解放）		-j オプションあり（空きセグメント解放）		-a オプションあり（空きページ及び空きセグメント解放）	
		異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）
	LOB 列だけのデータロード	○	○	×	×	○	△
pdmod	−	×	×	×	×	×	×
pdrorg	アンロード（unldenq=tblenq 又は unldenq=nowait）	○	○（インデクスは△）	×	×	○	△
	アンロード（unldenq=rdenq）	×	×	×	×	×	×
	表単位のリロード又は表単位の再編成	○	×	×	×	○	×
	RD エリア単位のリロード，又は RD エリア単位の再編成	×	×	×	×	×	×
	インデクスの一括作成	○	×	×	×	○	×
	インデクスの再作成（index 文指定，かつ横分割インデクス）	×	×	×	×	×	×
	インデクスの再作成（index 文指定，かつ非横分割インデクス）	○	×	×	×	○	×
	インデクスの再作成（idxname 文指定）	○	×	×	×	○	×
	インデクスの再編成（インデクス格納 RD エリアをコマンド閉塞している場合）	○	×	×	×	○	×
	インデクスの再編成（インデクス格納 RD エリアをコマンド閉塞していない場合）	×	×	×	×	×	×
	LOB 列だけのリロード	○	○	×	×	○	△
	LOB 列だけの再編成	○	×	×	×	○	×
pdrbal	−	×	×	×	×	×	×

ユティリティ	条件	pdreclaim					
		-j 及び-a オプションなし（空きページ解放）		-j オプションあり（空きセグメント解放）		-a オプションあり（空きページ及び空きセグメント解放）	
		異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）
pdreclaim	空きページ解放	○	×	×	×	○	×
	空きセグメント解放	×	×	×	×	×	×
	空きページ及び空きセグメント解放	○	×	×	×	○	×
pdpgbfon	データページの読み込み	○	○	×	×	○	△
	インデクスページの読み込み	○	×	×	×	○	×
pdconstck	−	○	×	×	×	○	×
pddbst	−	○	○（インデクスは△）	×	×	○	△
pdgetcst	最適化情報の取得	○	○（インデクスは△）	×	×	○	△
	最適化情報の削除	○	○	○	○	○	○
pdcopy	参照・更新不可能モード又は参照可能モード	×	×	×	×	×	×
	更新可能モード	○	○	○	○	○	○
pdrstr	−	×	×	×	×	×	×

（凡例）

○：同時実行できます。

×：同時実行できません（排他待ちとなります）。

△：pdreclaim が，同時実行するユティリティのトランザクションの完了を待ちます。

−：該当しません。

(3) pdreclaim と SQL の同時実行可否

pdreclaim と SQL の同時実行可否を次の表に示します。

表 11–4　pdreclaim と SQL の同時実行可否

SQL	LOCK 文	pdreclaim					
		-j 及び-a オプションなし（空きページ解放）		-j オプションあり（空きセグメント解放）		-a オプションあり（空きページ及び空きセグメント解放）	
		異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）	異なる表（インデクス）	同一表（インデクス）
SELECT 文，UPDATE 文，INSERT 文，DELETE 文	使用しない 又は IN SHARE MODE を使用する	○	○（インデクスは△）	×	×	○	△
	IN EXCLUSIVE MODE を使用する	○	○（インデクスは△）	×	×	○	×
PURGE TABLE 文	−	○	×	×	×	○	×
DROP TABLE	−	○	×※	×	×※	○	×※
CREATE INDEX	−	○	×	×	×	○	×
DROP INDEX	−	○	×※	×	×※	○	×※
ALTER TABLE	−	○	×※	×	×※	○	×※

（凡例）

○：同時実行できます。

×：同時実行できません（排他待ちとなります）。

△：pdreclaim が，同時実行する SQL のトランザクションの完了を待ちます。

−：該当しません。

注※

このユティリティ実行中に，処理中の表又はインデクスに対する表削除，インデクス削除，及び表定義変更は実行できません。しかし，タイミングによっては実行できる場合があり，次のように動作します。

表定義変更で追加した対象は pdreclaim の実行対象になりません。しかし，pdreclaim で対象とする表，インデクス又は RD エリアを削除する表定義変更をしたとき，KFPA11815-E が出力されます。

(4)　pdreclaim とローカルバッファを指定した UAP の同時実行可否

pdreclaim とローカルバッファを指定した UAP の同時実行可否を次の表に示します。

表 11–5　pdreclaim とローカルバッファを指定した UAP の同時実行可否

ローカルバッファの指定方法		同一 RD エリアの操作対象											
		-j 及び-a オプションなし（空きページ解放）				-j オプションあり（空きセグメント解放）				-a オプションあり（空きページ及び空きセグメント解放）			
pd_uap_wait オペランド	pdlbuffer オペランド	異なる表	異なるインデクス	同一表	同一インデクス	異なる表	異なるインデクス	同一表	同一インデクス	異なる表	異なるインデクス	同一表	同一インデクス
Y 指定	-r 指定	×※1	×※1	×※1	×※1	×※1	×※1	×※1	×※1	×※1	×※1	×※1	×※1
	-i 指定	○	○	○	×※1	×※1	×※1	×※1	×※1	○	○	×※1	×※1
N 指定	-r 指定	×※2	×※2	×※2	×※2	×※2	×※2	×※2	×※2	×※2	×※2	×※2	×※2
	-i 指定	○	○	○	×※2	×※2	×※2	×※2	×※2	○	○	×※2	×※2

（凡例）

○：同時実行できます。

×：同時実行できません。

注※1

排他待ちとなります。

注※2

UAP がエラー終了となります。

11.2　使用例

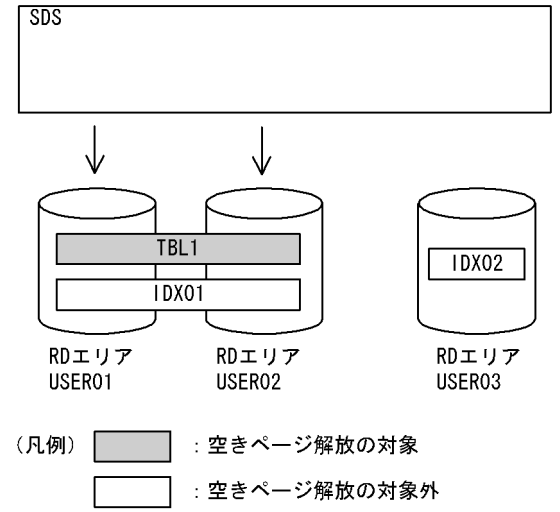
pdreclaim の使用例を次に示します。

11.2.1　HiRDB/シングルサーバの場合

例１：

表（TBL1）の使用中空きページ解放をします。表，及びインデクスは次のように定義されているものとします。
・表定義
```
CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
    IN ((USER01) C1>0,(USER02))
```
・インデクス定義
```
CREATE INDEX IDX01 ON TBL1(C1) IN ((USER01),(USER02))
CREATE INDEX IDX02 ON TBL1(C3) IN USER03
```

[概要]



●pdreclaim のコマンド

```
pdreclaim -k table -t TBL1
```

[説明]

-k table：表の使用中空きページ解放をする指定

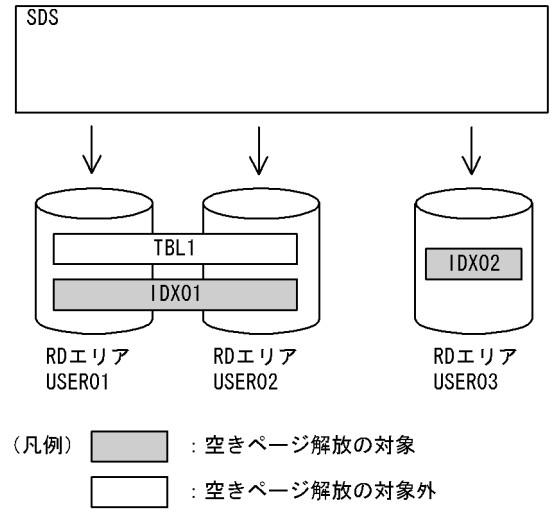
-t TBL1：使用中空きページ解放の対象となる表

例２：

表（TBL1）に定義されている，すべてのインデクスの使用中空きページ解放をします。表，及びインデクスは次のように定義されているものとします。
・表定義
```
CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
    IN ((USER01) C1>0,(USER02))
```
・インデクス定義

```
CREATE INDEX IDX01 ON TBL1(C1) IN ((USER01),(USER02))
CREATE INDEX IDX02 ON TBL1(C3) IN USER03
```

[概要]



●pdreclaim のコマンド

```
pdreclaim -k index -t TBL1
```

[説明]

-k index：インデクスの使用中空きページ解放をする指定

-t TBL1：使用中空きページ解放の対象となる表

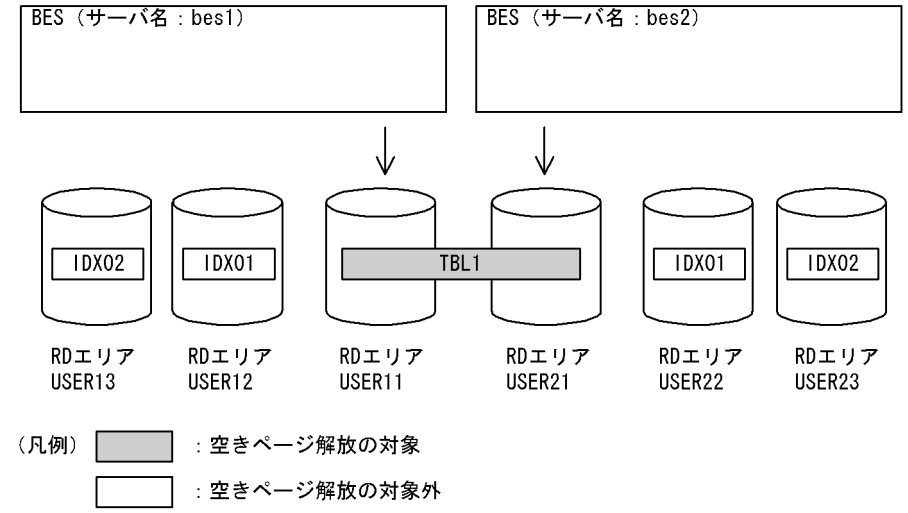
11.2.2　HiRDB/パラレルサーバの場合

例３：

表（TBL1）の使用中空きページ解放をします。表，及びインデクスは次のように定義されているものとします。

```
・表定義
  CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
      IN ((USER11) C1>0,(USER21))
・インデクス定義
  CREATE INDEX IDX01 ON TBL1(C1) IN ((USER12),(USER22))
  CREATE INDEX IDX02 ON TBL1(C3) IN ((USER13),(USER23))
```

[概要]



●pdreclaim のコマンド

```
pdreclaim -k table -t TBL1
```

[説明]

-k table：表の使用中空きページ解放をする指定

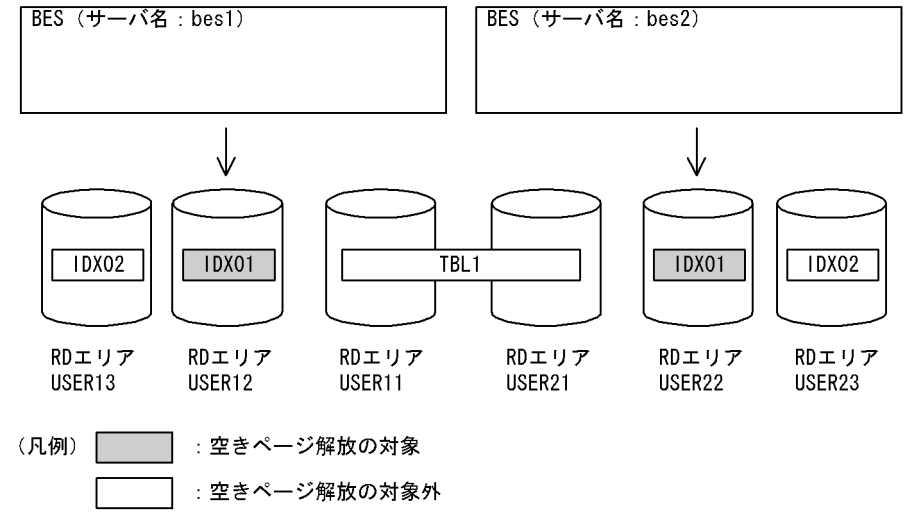
-t TBL1：使用中空きページ解放の対象となる表

例４：

表（TBL1）に定義されているインデクス（IDX01）の使用中空きページ解放をします。表，及びインデクスは次のように定義されているものとします。

```
・表定義
  CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
      IN ((USER11) C1>0,(USER21))
・インデクス定義
  CREATE INDEX IDX01 ON TBL1(C1) IN ((USER12),(USER22))
  CREATE INDEX IDX02 ON TBL1(C3) IN ((USER13),(USER23))
```

[概要]



●pdreclaim のコマンド

```
pdreclaim -k index -t TBL1 c:¥hirdb¥pdreclaim¥control_file
```

[説明]

-k index：インデクスの使用中空きページ解放をする指定

-t TBL1：使用中空きページ解放の対象となる表

c:¥hirdb¥pdreclaim¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdreclaim¥control_file）の内容

```
idxname name=IDX01    [1]
```

[説明]

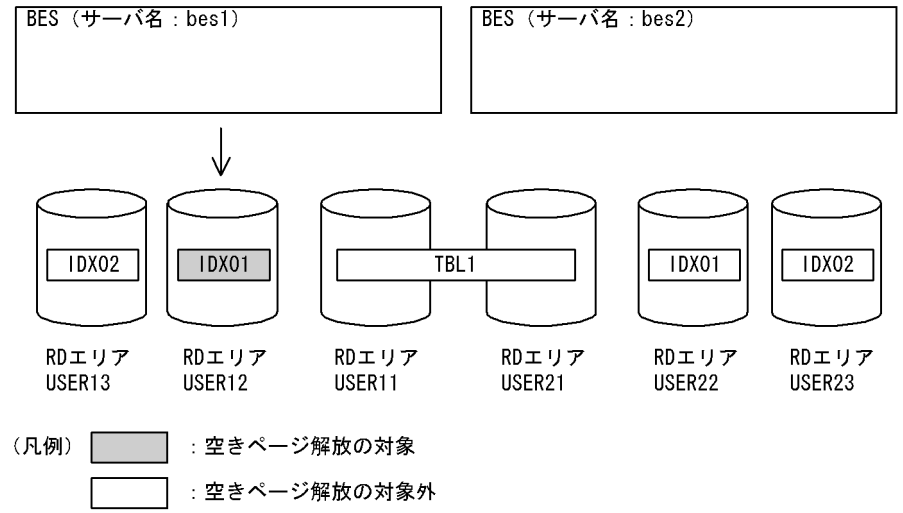
1. 使用中空きページ解放の対象となるインデクスを指定します。

例５：

表（TBL1）に定義されているインデクス（IDX01）の，インデクス格納用RDエリア（USER12）の使用中空きページ解放をします。表，及びインデクスは次のように定義されているものとします。

```
・表定義
  CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
      IN ((USER11) C1>0,(USER21))
・インデクス定義
  CREATE INDEX IDX01 ON TBL1(C1) IN ((USER12),(USER22))
  CREATE INDEX IDX02 ON TBL1(C3) IN ((USER13),(USER23))
```

[概要]



●pdreclaim のコマンド

```
pdreclaim -k index -t TBL1 c:¥hirdb¥pdreclaim¥control_file
```

[説明]

-k index：インデクスの使用中空きページ解放をする指定

-t TBL1：使用中空きページ解放の対象となる表

c:¥hirdb¥pdreclaim¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdreclaim¥control_file）の内容

```
idxname name=IDX01 rdarea=USER12    [1]
```

[説明]

1. 使用中空きページ解放の対象となるインデクスとインデクス格納用 RD エリアを指定します。

11.3　コマンドの形式

11.3.1　形式

```
pdreclaim -k 対象資源
          〔-t　{〔認可識別子.〕表識別子 | 〔認可識別子.〕all} 〕
          〔-r RDエリア名〕
          〔-c 処理対象〕
          〔-m 〔空きページ解放のコミット間隔〕〔, 空きセグメント解放のコミット間隔〕〕
          〔-u 認可識別子〕
          〔-w 同時実行トランザクション決着待ち時間〕
          〔-X サーバ間通信の応答監視時間〕
          〔 {-Z | -j | -a} 〕
          〔-x〕
          〔-p〕
          〔-n 排他制御リトライ回数〕
          〔-s サーバ名〕
          〔制御情報ファイル名〕
```

注

制御情報ファイル名を指定する場合，必ずオプションの最後に指定してください。

- **各オプションの関係**

-c user 指定時の各オプションの関係と-c dic 指定時の各オプションの関係を次の表に示します。

表 11–6　-c user 指定時の各オプションの関係

オプション	-k オプション		-t オプション	
	table	index	表識別子	all
-r RD エリア名	○	×	○	×
-m コミット間隔	○	○	○	○
-u 認可識別子	○	○	○	○
-w 同時実行トランザクション決着待ち時間	○	○	○	○
-X サーバ間通信の応答監視時間	○	○	○	○
{-Z ｜-j ｜-a}	○	○	○	○
-x	×	○	○	○
-p	○	○	○	○
-n 排他制御リトライ回数	○	○	○	○
-s サーバ名	○	○	○	○
制御情報ファイル名	○	○	○	○

（凡例）

○：指定できます。

×：指定できません。

表 11–7　-c dic 指定時の各オプションの関係

オプション	-k オプション	
	table	index
-t {〔認可識別子.〕表識別子｜〔認可識別子.〕all}	×	×
-r RD エリア名	×	×
-m コミット間隔	○	○
-u 認可識別子	○	○
-w 同時実行トランザクション決着待ち時間	○	○
-X サーバ間通信の応答監視時間	○	○
{-Z ｜-j ｜-a}	○	○
-x	×	○
-p	○	○
-n 排他制御リトライ回数	○	○
-s サーバ名	×	×
制御情報ファイル名	○	○

（凡例）

○：指定できます。

×：指定できません。

11.3.2　オプション

(1)　-k 対象資源

使用中空きページ解放の処理対象とする資源（表又はインデクス）を指定します。

table：

処理対象が表の場合に指定します。

index：

処理対象がインデクスの場合に指定します。

(2) -t {〔認可識別子.〕表識別子｜〔認可識別子.〕all}

～<識別子>

使用中空きページ解放の対象となる表名を指定します。

認可識別子を省略した場合，HiRDB に接続したときのユーザの名称が仮定されます。

all を指定した場合，認可識別子をスキーマ名とした，該当スキーマが持つすべての表，又はインデクスを処理対象とします。この場合，idxname 文は指定できません。

(3) -r RD エリア名

～<識別子>

-t オプションに指定した表を格納する RD エリアのうち，ある特定の RD エリアだけを処理対象としたい場合，その RD エリア名を指定します。

このオプションは，-k table の場合に指定できます。-k index の場合は，idxname 文で指定します。

RD エリア名には，ユーザ用 RD エリアだけ指定できます。また，一括指定は指定できません。

(4) -c 処理対象

～《user》

処理対象（ユーザ用 RD エリア，又はデータディクショナリ用 RD エリア）を指定します。

user：

ユーザ用 RD エリアを処理対象とします。

dic：

データディクショナリ用 RD エリアを処理対象とします。

(5) -m 〔空きページ解放のコミット間隔〕〔，空きセグメント解放のコミット間隔〕

空きページ解放，又は空きセグメント解放のコミット間隔を指定します。

空きページ解放のコミット間隔　～<符号なし整数>((0～100000))《1000》

使用中空きページ解放処理のトランザクション決着間隔を，解放したページ数で指定します。表又はインデクスの 1RD エリア単位での解放ページ数が指定値以上になると，トランザクションを決着します。解放したページ数には，ページコンパクションだけ実行され，使用中空きページにできなかったページも含まれます。

0 を指定した場合，該当する RD エリアにあるすべての使用中空きページ解放が完了してから，トランザクションを決着します。

指定値を大きくすると，トランザクションログ量を削減できます。しかし，その反面，pdreclaim が長時間トランザクションとなり，シンクポイント有効化のタイミングがなくなります。また，同時に-p オプションを指定する場合，シンクポイントダンプ取得前にトランザクション決着できるようにコミット間隔を指定する必要があります。-p オプション指定時のコミット間隔の目安は，pdreclaim がアクセスするグローバルバッファのシンクポイント出力ページ数の総和の 50%以下としてください。シンクポイント出力ページ数は「グローバルバッファプールに関する統計情報」の SYNCW で確認できます。

空きセグメント解放のコミット間隔　～<符号なし整数>((0～10000000))《0》

空きセグメント解放のコミット間隔の指定は，同時に-a オプションを指定しているときだけ有効となります。

使用中空きセグメント解放処理のトランザクション決着間隔を，解放したセグメント数で指定します。表又はインデクスの 1RD エリア単位での解放セグメント数が指定値以上になると，トランザクションを決着します。

0 を指定した場合，該当する RD エリアにあるすべての使用中空きセグメント解放が完了してから，トランザクションを決着します。

指定値を大きくすると，トランザクションログ量を削減できます。しかし，その反面，pdreclaim が長時間トランザクションとなり，シンクポイント有効化のタイミングがなくなります。また，同時に-p オプションを指定する場合，シンクポイントダンプ取得前にトランザクション決着できるようにコミット間隔を指定する必要があります。-p オプション指定時のコミット間隔の目安は，「pdreclaim がアクセスするグローバルバッファのシンクポイント出力ページ数の総和÷2×60」以下としてください。シンクポイント出力ページ数は「グローバルバッファプールに関する統計情報」の SYNCW で確認できます。

(6)　-u 認可識別子

pdreclaim を実行するユーザの認可識別子を指定します。

認可識別子については，「-u 認可識別子」を参照してください。

(7)　-w 同時実行トランザクション決着待ち時間

～<符号なし整数>((0～3600))《0》

空きページ解放又は空きセグメント解放をする表やインデクスは，UAP やユティリティから同時にアクセスされると，pdreclaim は排他解除待ち，又はトランザクション決着待ちとなります。-w オプションには，この pdreclaim の待ち時間を秒単位で指定します。

なお，pdreclaim は次の場合に待ち状態になります。

排他解除待ち：

次のどちらかのときに排他解除待ちになります。

- 同じ RD エリアを参照又は更新する UAP やユティリティが実行中のとき
- RD エリア単位のデータロードや，ローカルバッファを使用した UAP が実行中のとき

トランザクション決着待ち：

次のどちらかのときにトランザクション決着待ちになります。

- 空きページ解放するインデクス，及びそのインデクスが定義されている表に対して，次の操作がトランザクション決着待ちのとき

  ・SQL 文（SELECT 文，UPDATE 文，INSERT 文，及び DELETE 文）

  ・pdrorg のアンロード

  ・pddbst（表単位の状態解析，インデクス単位の状態解析，クラスタキー及びクラスタリングデータページの格納状態解析，状態解析結果蓄積機能，及び再編成時期予測機能）
- 空きセグメント解放する表又はインデクスに対して，次の操作がトランザクション決着待ちのとき

  ・SQL 文（SELECT 文，UPDATE 文，INSERT 文，及び DELETE 文）

  ・pdrorg のアンロード

  ・pdpgbfon

  ・pddbst（表単位の状態解析，インデクス単位の状態解析，クラスタキー及びクラスタリングデータページの格納状態解析，状態解析結果蓄積機能，及び再編成時期予測機能）

ホールダブルカーソルを使用した検索の実行待ち：

空きセグメント解放する表又はインデクスに対して，ホールダブルカーソルを使用して検索する UAP がある場合，ホールダブルカーソルを使用した検索の実行待ちとなります（ホールダブルカーソルを使用した検索が，カーソルをクローズしてトランザクションを決着するまで待ちます）。なお，pdreclaim 実行中にホールダブルカーソルをクローズしても，同一 UAP で同じ表又は同じインデクスに対して再度ホールダブルカーソルをオープンすれば，そのカーソルのクローズ，及びトランザクションの決着を待ちます。

-w オプション指定の有無と pdreclaim の待ち時間の関係を次の表に示します。

表 11–8　-w オプション指定の有無と pdreclaim の待ち時間の関係

処理	-w オプションの指定	排他解除待ちの時間	トランザクション決着待ちの時間	ホールダブルカーソルを使用した検索の実行待ちの時間
表の空きページ解放	あり	-w オプション指定値だけ待つ	−	−
	なし	排他待ち状態が解除されるまで待つ	−	−
インデクスの空きページ解放	あり	-w オプション指定値だけ待つ	-w オプション指定値だけ待つ	−
	なし	排他待ち状態が解除されるまで待つ	他トランザクションが決着するまで待つ	−
表の空きセグメント解放（-j）	あり	-w オプション指定値だけ待つ	−	−
	なし	排他待ち状態が解除されるまで待つ	−	−

処理	-w オプションの指定	排他解除待ちの時間	トランザクション決着待ちの時間	ホールダブルカーソルを使用した検索の実行待ちの時間
インデクスの空きセグメント解放（-j）	あり	-w オプション指定値だけ待つ	−	−
	なし	排他待ち状態が解除されるまで待つ	−	−
表の空きページ解放，表の空きセグメント解放（-a）	あり	-w オプション指定値だけ待つ	-w オプション指定値だけ待つ	-w オプション指定値だけ待つ
	なし	排他待ち状態が解除されるまで待つ	他トランザクションが決着するまで待つ※1	ホールダブルカーソルを使用した検索のカーソルがクローズしてトランザクションを決着するまで待つ※2
インデクスの空きページ解放，インデクスの空きセグメント解放（-a）	あり	-w オプション指定値だけ待つ	-w オプション指定値だけ待つ	-w オプション指定値だけ待つ
	なし	排他待ち状態が解除されるまで待つ	他トランザクションが決着するまで待つ※1	ホールダブルカーソルを使用した検索のカーソルがクローズしてトランザクションを決着するまで待つ※2

（凡例）

−：該当しません。

注※1

トランザクションが決着する契機を次に示します。

- COMMIT
- ROLLBACK
- 内部 ROLLBACK

空きページ解放の場合，UNTIL DISCONNECT 指定で検索していて，かつ COMMIT の操作をしたときは，次のどちらかの操作も必要となります。

- カーソルをクローズしてから COMMIT
- 次のデータを FETCH

注※2

ホールダブルカーソルを使用した検索の実行待ちが決着する契機を次に示します。

- ROLLBACK
- 内部 ROLLBACK
- カーソルをクローズしてから COMMIT

＜規則＞

pdreclaim が待ち状態になってから，指定時間以内に UAP 側のトランザクションが決着しない場合，及びホールダブルカーソルを使用した検索がカーソルをクローズしてトランザクションを決着しない場合，pdreclaim はリターンコード 4 で処理を打ち切ります。また，排他解除待ち状態が解除されない場合，pdreclaim はリターンコード 8 で処理を打ち切ります。

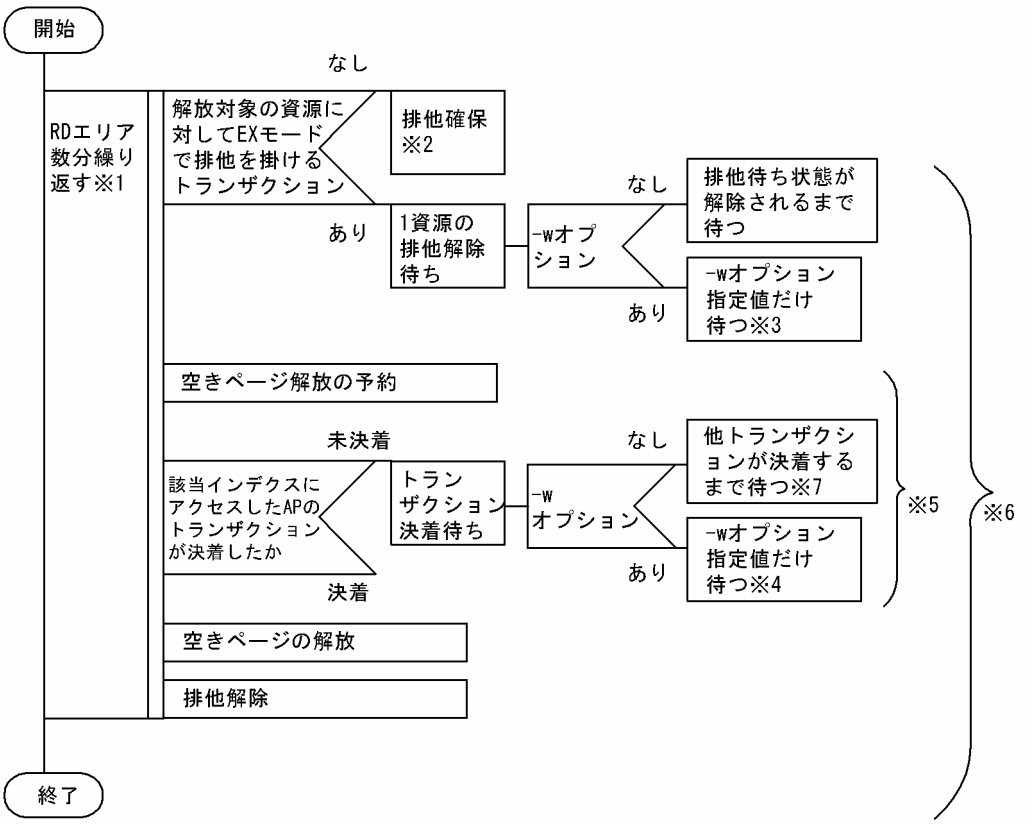
＜ pdreclaim の排他解除待ち＞

pdreclaim の排他解除待ちは，-a オプションの指定有無によって変わります。

- -a オプションなしの場合

  pdreclaim の排他制御の流れ（-a オプションなし）を次の図に示します。

  図 11–7　pdreclaim の排他制御の流れ（-a オプションなし）



注※1

表の空きページ解放の場合は表格納用 RD エリア，インデクスの空きページ解放の場合はインデクス格納用 RD エリアが対象となります。

注※2

空きページ解放の排他対象資源，排他モードについては，「ユティリティの排他制御モード」を参照してください。

注※3

-w オプションで指定した待ち時間を超える場合は，KFPA11770-I メッセージを出力して pdreclaim の処理を打ち切ります。

注※4

-w オプションで指定した待ち時間を超える場合は，KFPL27500-E，及び KFPH25002-E メッセージを出力して RD エリアの処理を打ち切り，次の RD エリアの処理をします。処理を打ち切るときには，KFPH25004-I メッセージを出力してすべての待ち対象のトランザクション情報を表示します。

注※5

インデクスの空きページ解放のときだけ該当します。

注※6

pdreclaim が排他を掛ける区間です。

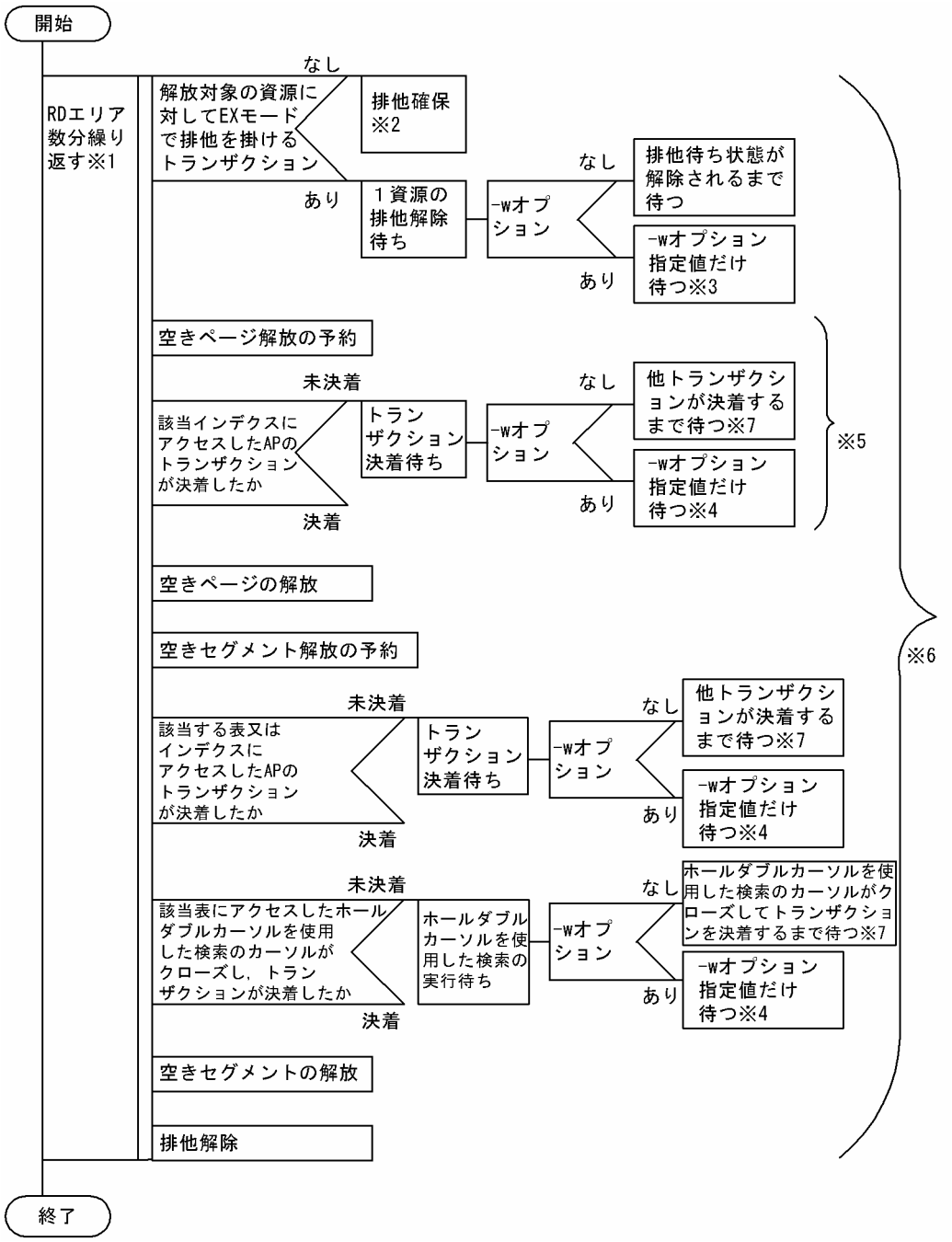
注※7

待ち時間が 10 分を経過した場合，KFPH25004-I メッセージを出力してすべての待ち対象のトランザクション情報を表示します。トランザクションの決着待ち要因ごとにトランザクション情報を出力するため，KFPH25004-I メッセージが複数回出力されることがあります。

- -a オプションありの場合

  pdreclaim の排他制御の流れ（-a オプションあり）を次の図に示します。

図 11–8　pdreclaim の排他制御の流れ（-a オプションあり）



注※1

表の空きページ解放の場合は表格納用 RD エリア，インデクスの空きページ解放の場合はインデクス格納用 RD エリアが対象となります。

注※2

空きページ解放の排他対象資源，排他モードについては，「ユティリティの排他制御モード」を参照してください。

注※3

-w オプションで指定した待ち時間を超える場合は，KFPA11770-I メッセージを出力して pdreclaim の処理を打ち切ります。

注※4

-w オプションで指定した待ち時間を超える場合は，KFPL27500-E，及び KFPH25002-E メッセージを出力して RD エリアの処理を打ち切り，次の RD エリアの処理をします。処理を打ち切るときには，KFPH25004-I メッセージを出力してすべての待ち対象のトランザクション情報を表示します。

注※5

インデクスの空きページ解放のときだけ該当します。

注※6

pdreclaim が排他を掛ける区間です。

注※7

待ち時間が 10 分を経過した場合，KFPH25004-I メッセージを出力してすべての待ち対象のトランザクション情報を表示します。トランザクションの決着待ち要因ごとにトランザクション情報を出力するため，KFPH25004-I メッセージが複数回出力されることがあります。

＜ KFPH25004-I メッセージの対処方法＞

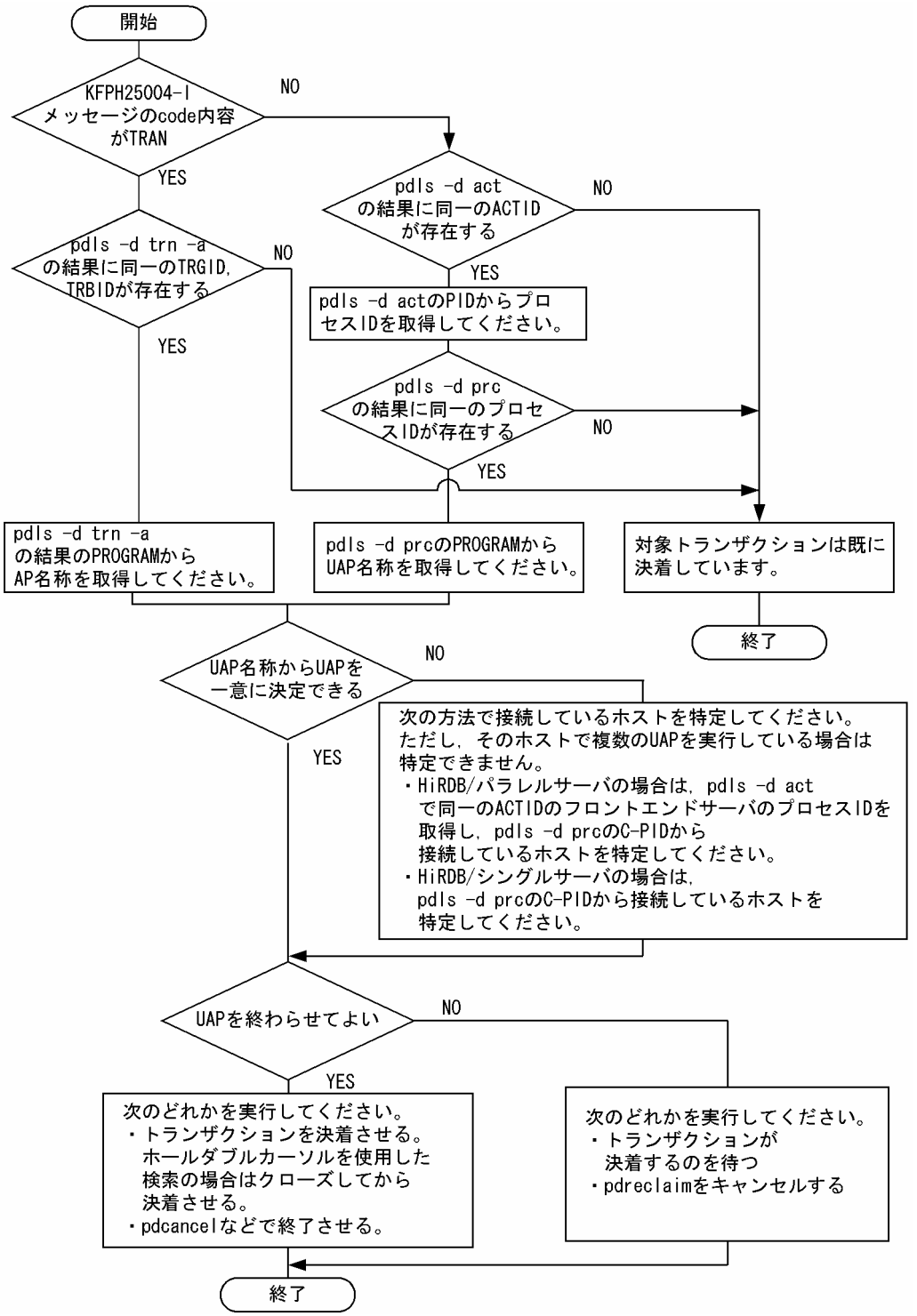
- -w オプションに 1 以上を指定した場合

  タイムアウトが発生したときに待ち対象となっているトランザクションが出力されます。メッセージに出力されたトランザクションがすべて完了した後で pdreclaim を再実行してください。トランザクションが完了しているかどうかは，次の図を参照して確認してください。

- -w オプションに 0 を指定又は省略した場合

  待ち対象になっているトランザクションが出力されます。次の図を参照して待ち対象になっているトランザクションを特定し，不要なトランザクションが残っている場合は終了させてください。そうでない場合は，トランザクションが終了するまで待ってください。なお，トランザクションが待ち状態になる要因は複数あるため，最初のメッセージに出力されたトランザクションがすべて終了した後で，更に別のトランザクションが出力されることがあります。トランザクションが終了する前に pdreclaim を終了させたい場合は，pdreclaim をキャンセルしてください。

図 11–9　KFPH25004-I メッセージの対処方法



(8)　-X サーバ間通信の応答監視時間

～<符号なし整数>　((1～65535))　《300》

コマンドを実行したサーバで通信障害などの障害が発生した場合，コマンドは無応答となり，業務が停止するおそれがあります。pdreclaim では障害を検知するために，コマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が，-X オプションに設定した時間を超えた場合は，pdreclaim はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。

＜適用基準＞

- 通信障害やユニットダウンによってサーバから応答が返らない状態となった場合に，300 秒より早く障害を検知したいとき，-X オプションに 300 より小さい値を指定してください。
- 系切り替え機能を使用している場合，系切り替えが完了しているにもかかわらず，コマンドが応答を待ち続けることがあります。その場合は，監視時間を短く設定することで，コマンドを即時終了できます。
- 同時実行するアプリケーションやユティリティの多重度が高い場合など，ディクショナリからの応答が遅延し，-X オプションのデフォルトである 300 秒以内にユティリティの前処理が完了しないときは，監視時間でタイムアウトするおそれがあります。このような環境では，-X オプションには 300 より大きい値を指定してください。

(9) {-Z ｜-j ｜-a}

空きページ，及び空きセグメントの解放方法を指定します。省略値は推奨モード又は互換モードによって次のように異なります。

省略値の動作モード	省略値
推奨モード	-a
互換モード	-Z

＜適用基準＞

空きセグメント解放が不要な場合は-Z を指定してください。空きセグメント解放をする場合，オンラインサービス停止中に，-j オプション指定の pdreclaim の実行時間を確保できるときは，-j を指定してください。実行時間を確保できない場合は-a を指定してください。

＜注意事項＞

- 空きセグメント解放をする場合で，使用中セグメント内のページがすべて未使用ページであっても，解放されないケースがあります。

  -j オプション指定の場合，1 表に対して，表格納用 RD エリアごとに解放しないセグメントが一つあります。

  -a オプション指定の場合，1 表に対して，表格納用 RD エリアごとに解放しないセグメントが二つ（リバランス表の場合は 1,024 個）あります。
- -j オプションを指定して空きセグメントを解放すると，グローバルバッファ上にある解放対象の表又はインデクスの更新バッファが DB に書き込まれ，更新バッファ及び参照バッファはグローバルバッファ上から削除されます。そのため，pdreclaim 実行直後は該当する表又はインデクスのバッファヒット率が低くなります。バッファヒット率を上げたい場合，pdpgbfon を使用して，再度グローバルバッファ上へデータを読み込んでください。

-Z：

-Z オプションを指定した場合，空きページだけ解放し，空きセグメントは解放しません。

-j：

-j オプションを指定した場合，処理中の表，インデクスはアクセスできなくなります。したがって，次のような使い方をしてください。

1. 昼のオンライン中には-Z オプションを指定（0904 互換モードの場合は-a，-j ともに指定なし）の pdreclaim を実行する。
2. 夜の夜間バッチで，-j 指定の pdreclaim を実行する。

-j オプションを指定すると，解放した未使用セグメントは，以前確保していた表，インデクス以外でも使用できるようになります。ただし，-k table の場合は表格納用 RD エリア，-k index の場合はインデクス格納用 RD エリアに EX モードの排他が掛かるため，pdreclaim と UAP との同時実行ができなくなります。

-a：

-a オプションを指定した場合，pdreclaim が表，インデクスを処理中であっても，UAP や他ユティリティからアクセスできます。ただし，空きセグメント解放の処理時間は長くなり，システムログファイルのログ量も増加します。システムログファイルの見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

また，-a オプションを指定すると，一つの表又はインデクスの格納 RD エリアごとに，最大 2 個の解放しないセグメントが発生します。リバランス表の場合は，格納 RD エリアごとに最大 1,024 個の解放しないセグメントが発生します。

(10) -x

インデクスキー値無排他を使用している場合又はコミットしていない削除データに排他を掛ける場合，インデクスの空きページ解放時にページコンパクションを実行して，インデクスの残存エントリを解放したい場合に指定します。ページコンパクションを実行することで，オンライン性能を維持したり，排他待ち及びデッドロックを回避したりできます。-x オプションは，-k index の場合に指定できます。

ページコンパクションを実行すると，残存エントリを解放するときに，解放されるキーの量に比例してデータベース更新ログ（FJ）が出力されます。

＜適用基準＞

-x オプションを指定すると，ページコンパクションのオーバヘッドが発生します。このため，データベース更新ログ（FJ）の出力量を抑えたい場合，又は性能を優先したい場合は，-x オプションを指定しないでください。

(11) -p

コミット時点で更新バッファをデータベースに反映する場合に指定します。UAP と同時に実行する場合に-p オプションを指定すると，シンクポイント処理の負荷を軽減できるので，ほかのトランザクション性

能に与える影響を小さくできます。ただし，pdreclaim の性能は-p オプションを指定しない場合より悪くなります。

また，シンクポイントダンプ取得前に，コミットが取得されるように，-m オプションを指定してコミット間隔を設定してください。

なお，-p オプション指定の pdreclaim を，サーバ当たり 10 個まで同時実行できます。上限を超えた pdreclaim はエラー終了します。

(12)　-n 排他制御リトライ回数

～＜符号なし整数＞((0～3600))《0》

-n オプションには，空きセグメント解放の排他取得要求を解除する上限の回数を指定します。なお，-n オプションは空きセグメント解放の場合だけ有効となります。このため，-n オプションを指定する場合は，-j オプションも併せて指定してください。

空きセグメント解放の排他取得要求の解除について説明します。

-j オプションを指定して空きセグメント解放をすると，RD エリアに EX モードの排他を掛けます。このため，ほかのトランザクションで処理中の RD エリアに対して空きセグメント解放を実行すると，排他解除待ちになるおそれがあります。

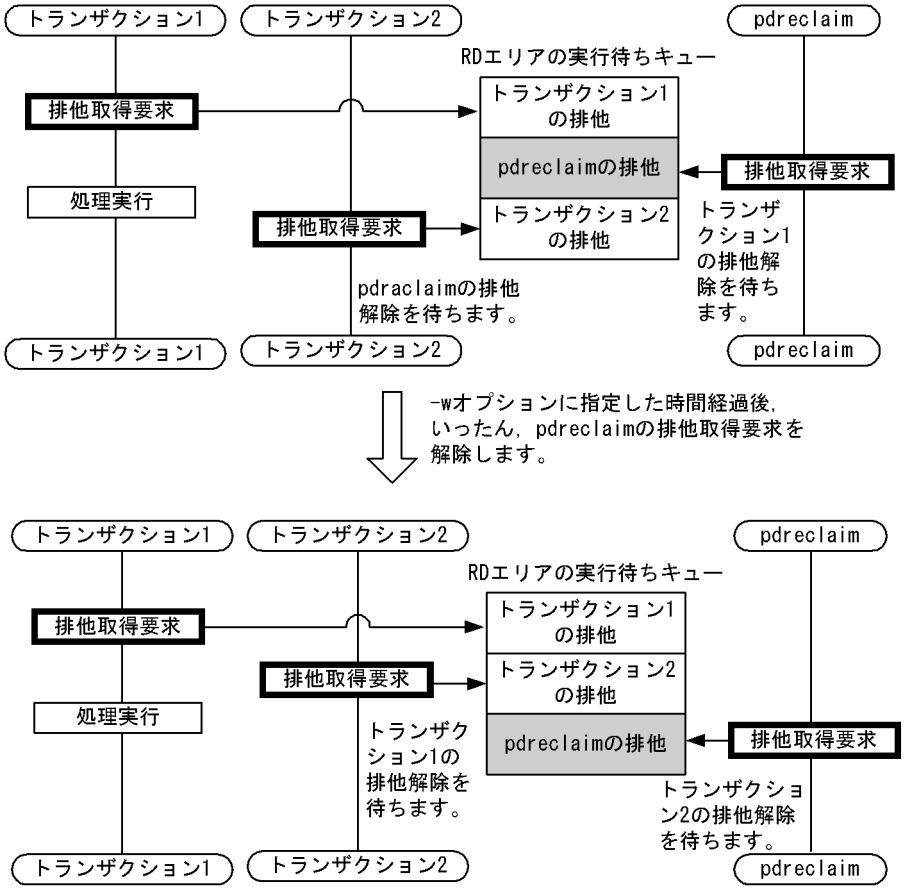
空きセグメント解放が排他解除待ち状態のときに，更にほかのトランザクションが RD エリアに対して排他を取得すると，最初に RD エリアに対して排他取得したトランザクションと空きセグメント解放が，排他解除をするまで待ち状態となります。

この場合に，-n オプションを指定していると，-w オプションに指定した時間が経過すると，空きセグメント解放の排他取得要求を解除し，ほかのトランザクションの排他解除待ちを優先できます。空きセグメント解放は，-w オプションに指定した値までの時間，排他が解除されるのを待ちます。このため，-n オプションを指定する場合，-w オプションの指定も必要になります。

なお，空きセグメントが解除されない場合，いったん，空きセグメント解放の排他取得要求を解除し，再度，空きセグメント解放の排他取得要求をします。

排他取得要求と排他待ちキューの関係を，次の図に示します。

図 11–10　排他取得要求と排他待ちキューの関係



注　-nオプションには，空きセグメント解放の排他取得要求，排他解除待ち，空きセグメント解放の排他取得要求解除の一連の流れについて，リトライの上限回数を指定します。

また，-n オプションを指定することで，最大「-w オプションの指定値×-n オプションの指定値」の時間だけ排他解除待ちをします。この「-w オプションの指定値×-n オプションの指定値」の時間が，option 文の exectime オペランドの pdreclaim 実行監視時間を超える場合，排他解除待ち中に KFPL11111-E メッセージを出力して，pdreclaim は強制終了します。このため，-n オプションには次の内容を考慮して，適切な値を設定してください。

-n オプションと pdreclaim 実行監視時間の関係

A ＋ B ＋ C ＜ D×60

（凡例）

A：トランザクション決着待ち時間（-w オプションの指定値）

B：空きセグメント解放の排他待ち時間（-w オプションの指定値×-n オプションの指定値）

C：空きセグメント解放の実行時間

D：option 文の exectime オペランドの pdreclaim 実行監視時間

(13)　-s サーバ名

～＜識別子＞

このオプションは HiRDB/パラレルサーバの場合に指定できます（HiRDB/シングルサーバの場合に指定しても無視されます）。

pdreclaim の実行制御をするバックエンドサーバの名称を指定します。

＜適用基準＞

省略した場合でも，自動的にどれかのバックエンドサーバが選択されるため，通常時は指定する必要はありません。

pdreclaim を大量に同時実行する場合に，1 サーバ内で pdrorg 用ユティリティサーバプロセス「0mrorg」が 32 個以上となる場合に指定してください。0mrorg の個数は，pdls -d prc -a コマンドの結果のプロセス ID（0mrorg）が何個開始されているかを調べれば分かります。

(14)　制御情報ファイル名

～＜パス名＞

pdreclaim の制御文を記述した制御情報ファイルの名称を指定します。

制御情報ファイルには，次に示す制御文を記述できます。各制御文については，「idxname 文（インデクスの情報の記述）」，「option 文（オプションの情報の記述）」を参照してください。なお，制御情報ファイル中にコメントは記述できません。

- idxname 文
- option 文

11.3.3　idxname 文（インデクスの情報の記述）

インデクスの使用中空きページ解放（-k index）の場合，対象とするインデクスの情報を指定します。

＜適用基準＞

-k index 場合，-t オプションに指定した表に定義されているすべてのインデクスが処理対象になりますが，特定のインデクス，サーバ，又は RD エリアだけを処理対象にするとき，idxname 文を指定します。

＜規則＞

1. name オペランドは，server 及び rdarea オペランドよりも前に指定してください。
2. 識別子に空白及び英小文字を含む場合は，引用符（"）で囲んで指定してください。英小文字が引用符で囲まれていない場合，英大文字として扱われます。
3. rdarea オペランドを指定した idxname 文を指定し，ほかにも idxname 文を指定する場合，name オペランドだけを指定した idxname 文，及び server オペランドを指定した idxname 文は指定できません。
4. 同じインデクス識別子を指定した idxname 文は複数個指定できません。ただし，rdarea オペランドを指定する場合は複数個指定できます。

(1) 形式

```
idxname name={インデクス識別子|*}
        〔{server=サーバ名〔,サーバ名〕…
          |rdarea=RDエリア名〔,RDエリア名〕…}〕
```

(2) 説明

(a) name={インデクス識別子|*}

インデクス識別子 ～＜識別子＞

インデクスの使用中空きページ解放の対象となるインデクス識別子を指定します。

認可識別子は，-t オプションに指定した表の認可識別子を仮定するため，ここでは指定する必要はありません。

*****

表に定義されているすべてのインデクスを対象にする場合，*を指定します。

*指定の idxname 文（server オペランドなし）と，idxname 文を省略した場合とでは，両方とも表のすべてのインデクスが処理対象となります。ただし，*を指定して server オペランドも指定した idxname 文の場合は，表のすべてのインデクスのうち，server オペランドに指定したバックエンドサーバにあるインデクスだけを処理対象にできます。

*を指定した場合，idxname 文は 1 個だけ指定できます。

(b) server=サーバ名〔，サーバ名〕...

～＜識別子＞

このオペランドは，HiRDB/パラレルサーバの場合に指定できます。name オペランドに指定したインデクスに対して，特定サーバのインデクスを対象とする場合に，そのサーバ名（バックエンドサーバ名）を指定します。省略した場合は，インデクスが格納されているすべてのサーバを対象とします。

(c) rdarea=RD エリア名〔，RD エリア名〕...

～＜識別子＞

name オペランドに指定したインデクスに対して，特定のインデクス格納用 RD エリアを対象とする場合に，その RD エリア名を指定します。

省略した場合は，インデクスが格納されているすべての RD エリアを対象とします。また，name オペランドに*を指定している場合，このオペランドは指定できません。

11.3.4　option 文（オプションの情報の記述）

pdreclaim の実行時間監視の情報を指定します。

＜適用基準＞

pdreclaim の実行時間を監視する場合に指定します。

(1)　形式

```
option 〔exectime=pdreclaimの監視時間〕
      〔globalbuffer_lru= {yes | no} 〕
```

(2)　説明

(a)　exectime=pdreclaim の監視時間

～＜符号なし整数＞((0～35791394))

pdreclaim の実行時間を監視する場合に，その監視時間を分単位で指定します。

＜指定値の目安＞

このオペランドは，無応答障害の検知を目的とするもので，長時間トランザクションの実行時間を監視するものではありません。したがって，指定値は，該当する表の処理時間に余裕を持たせた値にする必要があります。例えば，7～8 分程度で終了する pdreclaim の実行時間を監視する場合は，exectime=10 ではなく，exectime=20 と指定してください。

＜規則＞

1. このオペランドに 0 を指定した場合，実行時間の監視はしません。
2. このオペランドを省略した場合は，次の優先順位でシステム定義の指定値に従います。
   1. pd_utl_exec_time
   2. pd_cmd_exec_time
3. 指定時間以内に処理が終了しない場合，pdreclaim は強制終了し，無応答原因の調査用に障害情報を取得します。なお，実行時間の監視対象は，実際に使用中空きページ解放をする pdrorg 用ユティリティサーバプロセス（0mrorg）とシングルサーバ（又はバックエンドサーバ）であり，コマンドのプロセス（pdreclaim 及び pdrorg）に障害が発生した場合は強制終了できません。取得する障害情報については，マニュアル「HiRDB システム定義」の pd_utl_exec_time オペランドを参照してください。

(b)　globalbuffer_lru= {yes ｜ no}

pdreclaim が読み込んだページを LRU 管理するかどうかを指定します。省略値は推奨モード又は互換モードによって次のように異なります。

省略値の動作モード	省略値
推奨モード	no
互換モード	yes

このオペランドが no の場合，pdreclaim の使用するグローバルバッファを LRU 管理せずに，読み込んだページを常に最古のページとして扱います。この結果，グローバルバッファ不足時には pdreclaim が読み込んだページが最初に追い出されるページとなり，アプリケーション側の性能劣化が抑止されます。yes の場合，pdreclaim の使用するグローバルバッファを LRU 管理します。この結果，グローバルバッファ不足時には先にアプリケーションで読み込んでいたページがグローバルバッファから追い出され，アプリケーションの性能が劣化します。pdreclaim と同時実行中又は pdreclaim 実行後のアプリケーションの性能劣化を許容して，pdreclaim の実行速度を優先する場合は yes を指定してください。

＜指定値の目安＞

1. アプリケーションと同時に実行する場合は，アプリケーションの性能が安定するため，no にしてください。
2. アプリケーションと同時に実行しない場合，次に示す条件では，no にしてください。

   ・-j を指定したセグメント解放の場合

   ・使用中空きページ解放の処理対象が表であり，システム共通定義 pd_dbreuse_remaining_entries オペランドが ALL の場合

   ・全データをバッファ上に読み込んでいる場合

   次に示す条件で，pdreclaim の実行時間を短くしたい場合は，yes を指定してください※。

   ・使用中空きページ解放の処理対象が表であり，システム共通定義 pd_dbreuse_remaining_entries オペランドに ALL 以外を指定している場合

   ・使用中空きページ解放の処理対象がインデクスの場合

   注※

   pdreclaim 終了後は，アプリケーションで以前使用したグローバルバッファ上のページを置き換えてしまうため，アプリケーション側のバッファヒット率が低下し，アプリケーション性能が遅延するおそれがあります。

11.4　pdreclaim の処理結果

pdreclaim のリターンコードが 0 又は 4 の場合，標準出力に pdreclaim の処理結果が表示されます。

11.4.1　pdreclaim（-j 及び-a なし）の処理結果（表の使用中空きページ解放）

```
pdreclaim VV-RR         *** DB RECLAIM ***         YYYY-MM-DD hh:mm:ss [1]

TABLE NAME : USERA.TBL01 [2]
[3]     [4]                                  [5]            [6]            [7]
No.   RDAREA NAME                          PAGE COUNT  SEGMENT COUNT   ACCESS COUNT
  1   USER01                                   11,832            250         51,051
  2   USER02                                   10,452            242         51,051

     18244 11:02:08 un01          KFPL00739-I Pdreclaim terminated, return code=0
```

[説明]

1. 処理結果のヘッダ。

   VV-RR：バージョン，リビジョン番号

   YYYY-MM-DD hh:mm:ss：pdreclaim 開始日時

2. 処理対象の表名。

   スキーマ単位に処理した場合は，この項目から繰り返して表示されます。

3. 通番。

4. 表格納用 RD エリア名。

5. 解放したページ数。

6. 解放できるセグメント数。

   解放できるセグメント数は，pdreclaim の使用中空きページ解放で未使用セグメント化ができるようになったセグメント数だけでなく，LOCK 文を実行した表に対する DELETE 文によって未使用セグメント化ができるようになるセグメント数も含みます（ログレスモードで DELETE 文を実行すると，自動的に LOCK 文を実行した状態となります）。

7. グローバルバッファを使用してアクセスしたページ数。

   解放したページ数だけでなく，解放処理の延長で実際にアクセスしたページ数も含まれます。なお，解放対象のページ以外の管理ページをアクセスすることもあります。このため，表示される値は該当資源を格納したページ数より多くなります。

11.4.2 pdreclaim（-j 又は-a あり）の処理結果（表の使用中空きセグメント解放）

```
pdreclaim VV-RR         *** DB RECLAIM ***         YYYY-MM-DD hh:mm:ss [1]

TABLE NAME : USERA.TBL01 [2]
[3]      [4]                                [5]            [6]            [7]
No.   RDAREA NAME                         PAGE COUNT  SEGMENT COUNT   ACCESS COUNT
  1   USER01                                        -            250         17,017
  2   USER02                                        -            242         17,017

     18250 11:03:20 un01          KFPL00739-I Pdreclaim terminated, return code=0
```

[説明]

1. 処理結果のヘッダ。

   VV-RR：バージョン，リビジョン番号

   YYYY-MM-DD hh:mm:ss：pdreclaim 開始日時

2. 処理対象の表名。

   スキーマ単位に処理した場合は，この項目から繰り返して表示されます。

3. 通番。

4. 表格納用 RD エリア名。

5. -j オプション指定時は−（固定）。-a オプション指定時は解放したページ数。

6. 解放したセグメント数。

7. グローバルバッファを使用してアクセスしたページ数。

   解放したページ数だけでなく，解放処理の延長で実際にアクセスしたページ数も含まれます。なお，解放対象のページ以外の管理ページをアクセスすることもあります。このため，表示される値は該当資源を格納したページ数より多くなります。

11.4.3 pdreclaim（-j 及び-a なし）の処理結果（インデクスの使用中空きページ解放）

```
pdreclaim VV-RR         *** DB RECLAIM ***         YYYY-MM-DD hh:mm:ss [1]

INDEX NAME : USERA.(PRIMARY0000131193)  [2]
[3]      [4]                          [5]            [6]            [7]            [8]
No.   RDAREA NAME                   PAGE COUNT  SEGMENT COUNT   ACCESS COUNT     COMPACTION
  1   USER01                                32              0            227             58
  3   USER02                                32              0            227             58

INDEX NAME : USERA.IDX01
No.   RDAREA NAME                   PAGE COUNT  SEGMENT COUNT   ACCESS COUNT     COMPACTION
  1   USER01                               891             16          6,259          1,595
  3   USER02                               891             16          6,259          1,595
```

```
INDEX NAME : USERA.IDX02
No.  RDAREA NAME                  PAGE COUNT  SEGMENT COUNT   ACCESS COUNT     COMPACTION
  1  USER01                               97             14            682            174

     18246 11:04:10 un01          KFPL00739-I Pdreclaim terminated, return code=0
```

[説明]

1. 処理結果のヘッダ。

   VV-RR：バージョン，リビジョン番号

   YYYY-MM-DD hh:mm:ss：pdreclaim 開始日時

2. 処理対象のインデクス名。

   複数のインデクスを処理した場合は，この項目から繰り返して表示されます。

3. 通番。

4. インデクス格納用 RD エリア名。

5. 解放したページ数。

6. 解放できるセグメント数。

   解放できるセグメント数は，pdreclaim の使用中空きページ解放で未使用セグメント化ができるようになるセグメント数だけでなく，LOCK 文を実行した表に対する DELETE 文によって未使用セグメント化ができるようになるセグメント数も含みます（ログレスモードで DELETE 文を実行すると，自動的に LOCK 文を実行した状態となります）。

7. グローバルバッファを使用してアクセスしたページ数。

   解放したページ数だけでなく，解放処理の延長で実際にアクセスしたページ数も含まれます。なお，解放対象のページ以外の管理ページをアクセスしたり，上位ページに戻って解放処理をしたりすることもあります。このため，表示される値は該当資源を格納したページ数より多くなります。

8. 該当資源を格納するページ内でページコンパクションをしたページ数。

   残存エントリがあるページだけがページコンパクション対象となります。

11.4.4 pdreclaim（-j 又は-a あり）の処理結果（インデクスの使用中空きセグメント解放）

```
pdreclaim VV-RR         *** DB RECLAIM ***         YYYY-MM-DD hh:mm:ss [1]

INDEX NAME : USERA.(PRIMARY0000131193)  [2]
[3]      [4]                      [5]          [6]           [7]           [8]
No.  RDAREA NAME                  PAGE COUNT  SEGMENT COUNT   ACCESS COUNT     COMPACTION
  1  USER01                               -              0             44              -
  3  USER02                               -              0             44              -

INDEX NAME : USERA.IDX01
No.  RDAREA NAME                  PAGE COUNT  SEGMENT COUNT   ACCESS COUNT     COMPACTION
```

```
  1  USER01                                -              16          1,204              -
  3  USER02                                -              16          1,204              -

INDEX NAME : USERA.IDX02
No.  RDAREA NAME                 PAGE COUNT  SEGMENT COUNT   ACCESS COUNT     COMPACTION
  1  USER01                                -              14            131              -

     18252 11:05:25 un01          KFPL00739-I Pdreclaim terminated, return code=0
```

[説明]

1. 処理結果のヘッダ。

   VV-RR：バージョン，リビジョン番号

   YYYY-MM-DD hh:mm:ss：pdreclaim 開始日時

2. 処理対象のインデクス名。

   複数のインデクスを処理した場合は，この項目から繰り返して表示されます。

3. 通番。

4. インデクス格納用 RD エリア名。

5. -j オプション指定時は−（固定）。-a オプション指定時は解放したページ数。

6. 解放したセグメント数。

7. グローバルバッファを使用してアクセスしたページ数。

   解放したページ数だけでなく，解放処理の延長で実際にアクセスしたページ数も含まれます。なお，解放対象のページ以外の管理ページをアクセスしたり，上位ページに戻って解放処理をしたりすることもあります。このため，表示される値は該当資源を格納したページ数より多くなります。

8. -j オプション指定時は−（固定）。

11.4.5　処理結果についての注意事項

処理結果の出力順序は，完了した資源の順番となります。

なお，-j 又は-a オプション指定ありの場合，pdreclaim が処理結果に表示する「解放したページ数」と，pddbst での解析結果の「解放できるページ数（Collect Prearranged Page）」は一致しないことがあります。また，解放できるセグメント数（-j 及び-a オプション指定なしの pdreclaim の処理結果）と，解放したセグメント数（-j 又は-a オプション指定ありの pdreclaim の処理結果）も一致しないことがあります。一致しない理由を次に示します。

(1)　解放できると予測しながら pdreclaim で解放できない場合

- pdreclaim は，ほかの UAP やユティリティでの更新処理を同時実行できるようにするため，同時更新の可能性がある使用中空きページ及び使用中空きセグメントは解放しません。また，インデクスの段数を減らす解放はしません。

- 同期点指定の pdload，pdrorg が異常終了した場合，同期点情報を保持するため，該当する表が確保している先頭のセグメントは，データがない状態でも解放はしません。
- pdreclaim と更新処理を同時実行した場合，解放できると判断した使用中空きページ，及び使用中空きセグメントが，更新処理に使用されることがあります。
- インデクスリーフページのインデクスエントリ数が 0 の場合（ページ使用率を 0%と扱う），pddbst は解放できると予測します。ただし，インデクスの中間ページからポイントする，下位のリーフページの中で最も大きいキー値を持つリーフページの場合は，インデクスエントリ数が 0 となっていても，同じ中間ページからポイントするほかの下位のリーフページがすべて解放された状態になるまで解放しません。

(2)　解放できないと予測しながら pdreclaim で解放できる場合

- pddbst 実行時，B-tree インデクスの構造を形成するインデクスの中間ページは，解放できるページ数としてカウントしません（下位にリーフページが存在するため）。しかし，pdreclaim は下位のリーフページをすべて解放した場合，中間ページも同時に解放します。
- キー値が 201 件以上重複したインデクスの場合，キー値を管理する特殊なページが作成されます。pddbst 実行時，この管理ページも解放できるページ数としてカウントしません（下位に重複キーの情報を保持するページが存在するため）。しかし，pdreclaim は下位の重複キーの情報を保持するページをすべて解放した場合，管理ページも同時に解放します。

11.5　注意事項

1. pdreclaim は内部的に pdrorg を実行するため，同時実行可能数は pdrorg の上限と同じになります。つまり，pdreclaim の同時実行可能数は，「pdrorg の同時実行可能数−実行中の pdrorg 数」となります。
2. ビュー表，及び一時表は処理対象外です。-t オプションにビュー表又は一時表を指定して実行すると，pdreclaim は KFPL15231-E メッセージを出力して，エラー終了します。また，-t オプションに all を指定した場合，pdreclaim は該当する表の処理をスキップします。
3. LOB 用 RD エリアは処理対象外です。したがって，抽象データ型，及びプラグインインデクスを格納している LOB 用 RD エリアも処理対象外となります。
4. インデクスの使用中空きページ解放をする場合，解放中のページを UAP がアクセスしているときは，該当する UAP のトランザクションが決着するまで pdreclaim が待ち状態になります。この pdreclaim の待ち時間は，-w オプションで指定できます。
5. 実行中の pdreclaim を強制終了させる場合は，pdcancel コマンドで強制終了させてください。pdkill コマンドなどで強制終了しても，pdreclaim は終了しません。
6. -j オプションを指定した pdreclaim を実行する場合，UAP が不当に排他待ちとならないように，pdhold コマンドで RD エリアを閉塞しておくことをお勧めします。

   pdreclaim に-j オプションを指定すると，表及びインデクスを格納している RD エリアに対して EX モードの排他を掛けるため，表及びインデクスを格納している RD エリア全体がアクセスできなくなります。この排他制御は，-w オプション，及び-n オプションの指定によって，次の表に示すように動作します。

表 11–9　空きセグメント解放時の排他制御の動作

<table>
<tr><th colspan="2">オプションの指定値</th><th rowspan="3">1 回の排他待ち時間</th><th colspan="4">排他エラー発生時</th></tr>
<tr><th rowspan="2">-w</th><th rowspan="2">-n</th><th colspan="3">排他待ちタイムアウトエラー</th><th>タイムアウト以外のエラー</th></tr>
<tr><th>排他のリトライ</th><th>リトライ回数</th><th>対処</th><th>対処</th></tr>
<tr><td>指定なし</td><td>−</td><td>無限</td><td>−</td><td>−</td><td>−</td><td>−</td></tr>
<tr><td rowspan="2">指定あり</td><td>指定なし</td><td rowspan="2">-w オプションに指定した時間</td><td>なし</td><td>0</td><td>エラーメッセージを出力して処理を打ち切ります。</td><td rowspan="2">エラーメッセージを出力して処理を打ち切ります。</td></tr>
<tr><td>指定あり</td><td>あり</td><td>-n オプションに指定した回数</td><td>-n オプションに指定した回数まで排他リトライをします。
-n オプションに指定した回数で，タイムアウトした場合は，メッセージを出力して処理を打ち切ります。</td></tr>
</table>

（凡例）

−：該当しません。

空きセグメント解放で排他する資源を次の表に示します。

表 11–10　空きセグメント解放で排他する資源

資源名称	資源番号	モード	排他の目的	排他リトライの有無
pdreclaim	5006	EX	同一資源への pdreclaim 同時実行抑止	pdreclaim 実行中の場合は，処理を打ち切るため，排他リトライはありません。
RD エリア名	0001	EX	空きセグメント解放中の RD エリアへのアクセス抑止	-n オプションを指定しているときに，排他待ちタイムアウトエラーが発生すると，排他リトライをします。
表名	0002	SR		pdreclaim 実行中の場合は，処理を打ち切るため，排他リトライはありません。
インデクス名	0003	SU		
表名（NOWAIT 検索時）	000B	SR		
RRAMB	0102	SR		RD エリアへの EX モードの排他が成功した後に確保する資源のため，タイムアウトエラーは発生しません。このため，排他リトライはありません。

7. pdreclaim のリターンコードを次の表に示します。

表 11–11　pdreclaim のリターンコード

リターンコード	意味	対処方法
0	使用中空きページ，使用中空きセグメントの解放が完了しました。	なし。
4	UAP のトランザクション決着待ち時間，又はホールダブルカーソルを使用した検索の実行待ち時間のタイムオーバーが発生したため，処理を中断しました。	タイムオーバーで使用中空きページ解放，又は使用中空きセグメント解放を中断した状態のため，表，及びインデクスの状態は保証されます。したがって，このままでも表，及びインデクスは UAP からアクセスできます。 未解放の使用中空きページを解放するには，pdreclaim を再度実行してください。 インデクスの空きページ解放を行った場合，及び-a オプション指定の pdreclaim を実行した場合は，pddbst で解放途中のセグメント及びページがないことを確認してください。インデクスの空きページ解放の場合，解放途中のページがあるときは，pdreclaim を再実行するか，又は pdrorg でインデクスの再編成を実行してください。-a オプション指定の pdreclaim の場合，解放途中セグメントがあるときは，pdreclaim を再実行するか，又は pdrorg で表の再編成を実行してください。
8	異常終了しました。	エラーによって使用中空きページ解放を中断した状態のため，表，及びインデクスの状態は保証されます。したがって，異常終了後でも表，及びインデクスは UAP からアクセスできます。

リターンコード	意味	対処方法
		pdreclaim を再度実行する場合は，出力されたエラーメッセージを参照してエラー原因を取り除いてください。 -a オプションを指定している場合は，解放途中セグメントが残ることがあるため，pdreclaim を再度実行してください。 インデクスの空きページ解放を行った場合，及び-a オプション指定の pdreclaim を実行した場合は，pddbst で解放途中のセグメント及びページがないことを確認してください。インデクスの空きページ解放の場合，解放途中のページがあるときは，pdreclaim を再実行するか，又は pdrorg でインデクスの再編成を実行してください。-a オプション指定の pdreclaim の場合，解放途中セグメントがあるときは，pdreclaim を再実行するか，又は pdrorg で表の再編成を実行してください。

8. 処理対象の表に，ユーザが定義した抽象データ型列がある場合，pdreclaim は実行できません。
9. 共用表，又は共用インデクスに対して pdreclaim を実行する場合，共用表，又は共用インデクスを格納している RD エリアに対して EX モードで排他を掛けます。そのため，該当する RD エリアに定義したほかの表やインデクスも参照，及び更新ができません。共用表に対して pdreclaim を実行する場合の排他制御モードについては，「ユティリティの排他制御モード」を参照してください。
10. HiRDB/パラレルサーバの場合，通信回線の混雑によってバックエンドサーバから pdreclaim に対する送信電文が 1 秒以上遅れたときは，該当するバックエンドサーバの処理結果が標準出力に表示されないことがあります。この場合，メッセージログファイル又はイベントログに出力される KFPL00714-I メッセージで処理結果を確認してください。
11. pdreclaim を実行している表やインデクスに対して，定義系 SQL は実行しないでください。実行すると，pdreclaim が異常終了します。スキーマ単位に pdreclaim を実行している場合は，該当スキーマが持つすべての表とインデクスが対象となります。
12. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdreclaim の制御情報ファイルには，BOM を持つファイルを使用できますが，BOM は読み飛ばされます。

12 グローバルバッファ常駐化ユティリティ（pdpgbfon）

この章では，HiRDB 開始直後，オンライン業務開始前などに，表のデータページ，インデクスページをグローバルバッファに読み込むグローバルバッファ常駐化ユティリティ（pdpgbfon）について説明します。

12.1　概要

12.1.1　pdpgbfon の概要

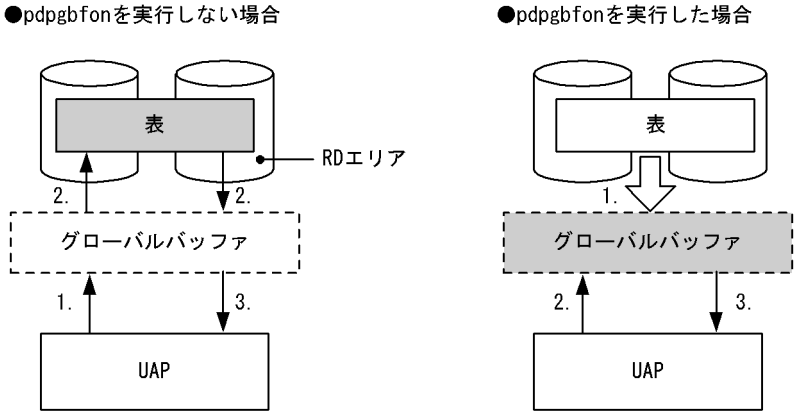
DB にアクセスする場合，物理的な入出力動作なしでグローバルバッファ上から目的の情報が見付かることをバッファヒットといいます。このバッファヒットの確率（バッファヒット率）が高いと，DB アクセスの性能は安定します。しかし，HiRDB 開始直後の場合など，表やインデクスの情報はグローバルバッファ上にはないため，バッファヒット率が低い状態となり，DB アクセス性能は安定しません。

グローバルバッファ常駐化ユティリティ（pdpgbfon）を実行すると，あらかじめ表やインデクスの情報をグローバルバッファに読み込んでおくことができます。したがって，HiRDB 開始直後，オンライン業務開始前などに pdpgbfon を実行しておけば，安定した DB アクセスの性能が期待できます。

例えば，DB のすべての表，インデクスをグローバルバッファに読み込める環境の場合，物理的な入出力動作が発生しない高速な DB として運用できます。

pdpgbfon の効果を次の図に示します。

図 12–1　pdpgbfon の効果



[説明]

pdpgbfon を実行しない場合：

1. 表をアクセスします。
2. グローバルバッファにページ情報がないため，表からページ情報を読み込みます（物理的な入出力が発生します）。
3. 結果が返ります。以降，このページ情報をアクセスする場合は，グローバルバッファに読み込まれているため，表からの読み込みは発生しません。ただし，ほかのページ情報をアクセスする場合は，2.の読み込み処理が発生します。

pdpgbfon を実行する場合：

1. あらかじめ pdpgbfon を実行して，表のページ情報をグローバルバッファに読み込みます。

2. 表をアクセスします。

3. 既にグローバルバッファにページ情報があるため，表からページ情報を読み込まないで結果が返ります。以降，この表をアクセスする場合は，表からの読み込みはありません（物理的な入出力は発生しません）。

12.1.2 pdpgbfon の機能

(1) 機能概要

pdpgbfon は，該当する資源（表又はインデクス）のページをグローバルバッファに読み込むことによって，オンライン業務開始直後でもバッファヒット率を上げる目的で使用します。この場合，既にグローバルバッファに対象となるページが読み込まれていたときは，ページの読み込み処理は発生しません。

pdpgbfon を実行する場合，グローバルバッファの面数は対象資源が格納されているページ数より多く必要です（pdpgbfon 実行前に，対象資源の全データが読み込めるだけのグローバルバッファを，表やインデクスに割り当てていることが前提です）。

pdpgbfon の処理方式は格納ページ順の読み込みとなるため，プリフェッチ機能が有効となります。したがって，グローバルバッファを定義する場合，プリフェッチ数を指定することで実行時間の短縮が図れます。

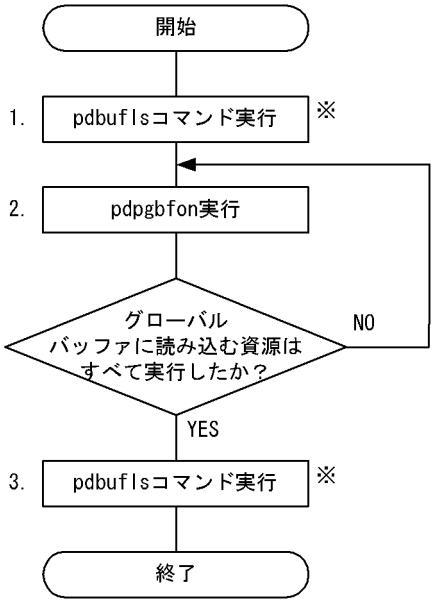
グローバルバッファの面数が十分でない場合，LRU 管理方式によってグローバルバッファから古いページ情報が追い出されます（システム定義の pd_dbbuff_lru_option オペランドの値に従って，アクセスしたグローバルバッファ中の最も古いページが追い出されます）。グローバルバッファの面数が十分でない場合，pdpgbfon を実行しても意味がありません。なお，グローバルバッファ面数は pdbufls コマンドで，対象資源の格納ページ数はデータベース状態解析ユティリティ（pddbst）で確認できます。

pdpgbfon を実行した結果，該当する資源の全ページがグローバルバッファ上に読み込まれたかどうかは，pdpgbfon のリターンコード，又はコマンドを実行した標準出力に処理結果（参照バッファフラッシュ回数）が出力されます。なお，pdpgbfon が出力する処理結果は，pdpgbfon で処理した資源の情報だけです。グローバルバッファ全体の処理結果や，詳細なグローバルバッファの状態については，pdbufls コマンドで確認してください。

(2) 運用方法

pdpgbfon の運用方法の例を次の図に示します。

図 12–2　pdpgbfon の運用方法の例



注※　格納ページ数よりグローバルバッファ面数が大きい場合，実行する必要はありません。

[説明]

1. pdbufls コマンドを実行し，グローバルバッファの統計情報を表示します。これによって，統計情報を初期化しておきます（表示結果は使用しません）。
2. グローバルバッファに読み込む資源（表のデータページ，又はインデクスページ）の数だけ，pdpgbfon を実行します。この場合，実行したすべての pdpgbfon のリターンコードが 0 であれば，3.は実行する必要はありません。
3. 再度 pdbufls コマンドを実行し，グローバルバッファの統計情報を表示します。表示された参照バッファフラッシュ回数（RFFLS），又は更新バッファフラッシュ回数※1（UPFLS）が 0 以外の場合，そのグローバルバッファには追い出されたページがあることを意味するため，該当するグローバルバッファを大きくするか，又は表の再編成※2を検討してください。なお，参照 GET 回数（REFGET）には，管理ページの再読み込みの回数も含まれるため，実際のデータ格納ページ数よりも大きい値が表示されます。

注※1

pdpgbfon がデータを更新することはありません。グローバルバッファは複数のプロセスで使用されるため，pdpgbfon 以外にグローバルバッファをアクセスしたプロセスがある場合，この項目を確認する必要があります。

注※2

DB に格納されているデータ件数の増加が原因の場合，グローバルバッファを大きくしてください。DB の配置乱れによる格納ページ数の増加が原因の場合，空きページ解放ユティリティ（pdreclaim），又はデータベース再編成ユティリティ（pdrorg）を実行して，格納ページ数を適正化してください。

12.1.3　実行環境

1. pdpgbfon は，HiRDB が稼働中のときだけ実行できます。
2. pdpgbfon は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. 処理対象 RD エリアは，オープン状態，又は参照可能閉塞状態であれば pdpgbfon は実行できます。pdpgbfon の実行可否については，「コマンド実行時の RD エリアの状態」を参照してください。
4. pdpgbfon を実行する場合，読み込み対象となる表のデータページ，インデクスページの全データを読み込めるだけのグローバルバッファ面数を割り当てておいてください。グローバルバッファが不足すると，最初に読み込んだページがグローバルバッファから追い出されます。

12.1.4　実行者

- スキーマ単位に読み込む場合
  DBA 権限所有者，又はスキーマの所有者が実行できます。
- そのほかの場合
  DBA 権限所有者，又は表の SELECT 権限所有者が実行できます。

監査証跡表についても，DBA 権限があれば処理できます。

12.1.5　pdpgbfon と他機能との同時実行

pdpgbfon と他機能との同時実行性について説明します。

(1)　pdpgbfon と SQL の同時実行可否

pdpgbfon と SQL の同時実行可否を次の表に示します。

表 12–1　pdpgbfon と SQL の同時実行可否

SQL 文		同一 RD エリアの操作対象	
		異なる表（インデクス）	同一表（インデクス）
検索	NOWAIT 指定	○	○
	WITH SHARE 指定	○	○
	WITH EXCLUSIVE 指定	○	○
	FOR UPDATE 句指定	○	○
	上記以外	○	○
更新	NEXT VALUE 句指定	○	○

SQL 文		同一 RD エリアの操作対象	
		異なる表（インデクス）	同一表（インデクス）
	上記以外	○	○
追加	NEXT VALUE 句指定	○	○
	上記以外	○	○
削除		○	○
LOCK 文	SHARE 指定	○	○
	EXCLUSIVE 指定（非共用表）	○	○
	EXCLUSIVE 指定（共用表）	×	×
表削除		○	×※
インデクス	定義	○	○
	削除	○	×※
全行削除		○	×
表定義変更		○	×※
順序数生成子定義		○（順序数生成子定義は表に対して行う SQL ではないため，同一表かどうかの区別は不要）	
順序数生成子削除		○（順序数生成子定義は表に対して行う SQL ではないため，同一表かどうかの区別は不要）	

（凡例）

○：同時実行できます。

×：同時実行できません（排他待ちとなります）。

注※

このユティリティを実行中に，処理中の表又はインデクスに対する，表削除，インデクス削除及び表定義変更は実行できません。しかし，タイミングによっては実行できる場合があり，次のように動作します。

表定義変更で追加した対象は pdpgbfon の実行対象になりません。しかし，pdpgbfon で対象とする表，インデクス又は RD エリアを削除する表定義変更をしたとき，KFPA11815-E が出力されます。

(2) pdpgbfon とローカルバッファを指定した UAP の同時実行可否

pdpgbfon とローカルバッファを指定した UAP の同時実行可否を次の表に示します。

表 12–2　pdpgbfon とローカルバッファを指定した UAP の同時実行可否

ローカルバッファの指定方法		同一 RD エリアの操作対象			
pd_uap_wait オペランド	pdlbuffer オペランド	異なる表	異なるインデクス	同一表	同一インデクス
Y 指定	-r 指定	×※1	×※1	×※1	×※1
	-i 指定	○	○	○	×※1
N 指定	-r 指定	×※2	×※2	×※2	×※2
	-i 指定	○	○	○	×※2

（凡例）

○：同時実行できます。

×：同時実行できません。

注※1

排他待ちとなります。

注※2

UAP がエラー終了となります。

12.2　使用例

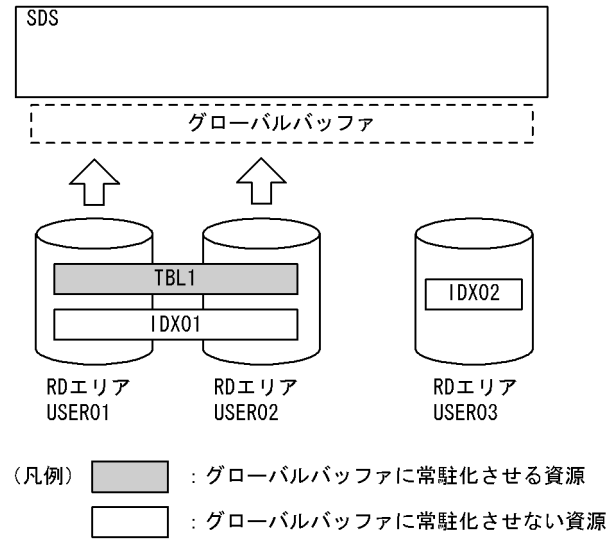
pdpgbfon の使用例を示します。

12.2.1　HiRDB/シングルサーバの場合

例１：

表（TBL1）のデータページをグローバルバッファに読み込みます。表，及びインデクスは次のように定義されているものとします。
・表定義
```
CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
    IN ((USER01) C1>0,(USER02))
```
・インデクス定義
```
CREATE INDEX IDX01 ON TBL1(C1) IN ((USER01),(USER02))
CREATE INDEX IDX02 ON TBL1(C3) IN USER03
```

[概要]



●pdpgbfon のコマンド

```
pdpgbfon -k table -t TBL1
```

[説明]

-k table：表のデータページを読み込む場合に指定します。

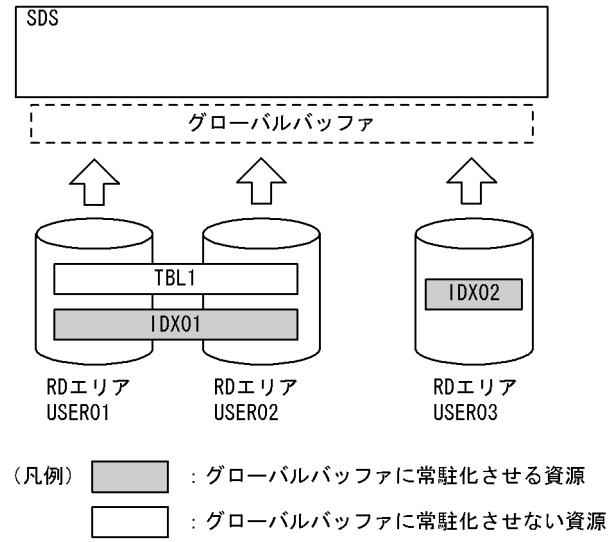
-t TBL1：読み込み対象となる表を指定します。

例２：

表（TBL1）のインデクスページをグローバルバッファに読み込みます。表，及びインデクスは次のように定義されているものとします。
・表定義
```
CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
    IN ((USER01) C1>0,(USER02))
```
・インデクス定義

```
CREATE INDEX IDX01 ON TBL1(C1) IN ((USER01),(USER02))
CREATE INDEX IDX02 ON TBL1(C3) IN USER03
```

[概要]



●pdpgbfon のコマンド

```
pdpgbfon -k index -t TBL1
```

[説明]

-k index：表のインデクスページを読み込む場合に指定します。

-t TBL1：読み込み対象となる表を指定します。

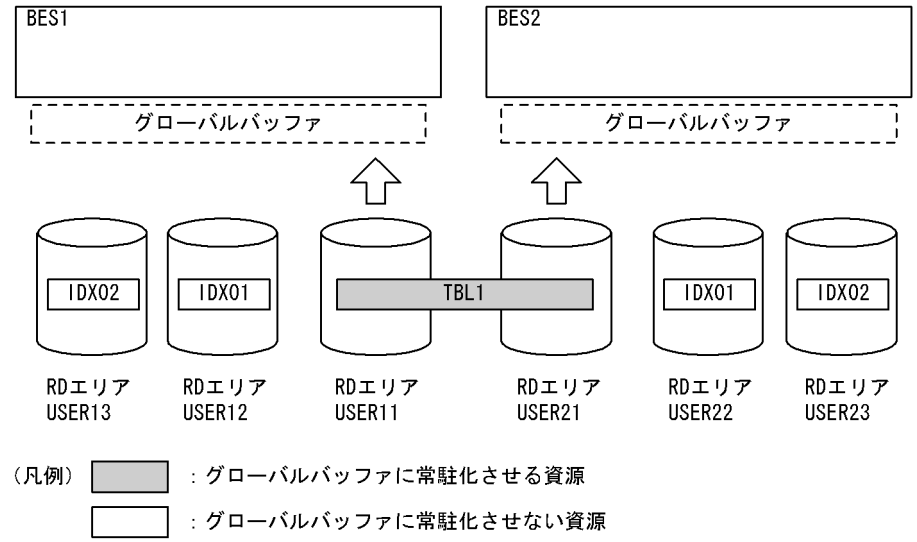
12.2.2 HiRDB/パラレルサーバの場合

例３：

表（TBL1）のデータページをグローバルバッファに読み込みます。表，及びインデクスは次のように定義されているものとします。

```
・表定義
  CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
      IN ((USER11) C1>0,(USER21))
・インデクス定義
  CREATE INDEX IDX01 ON TBL1(C1) IN ((USER12),(USER22))
  CREATE INDEX IDX02 ON TBL1(C3) IN ((USER13),(USER23))
```

[概要]



●pdpgbfon のコマンド

```
pdpgbfon -k table -t TBL1
```

[説明]

-k table：表のデータページを読み込む場合に指定します。

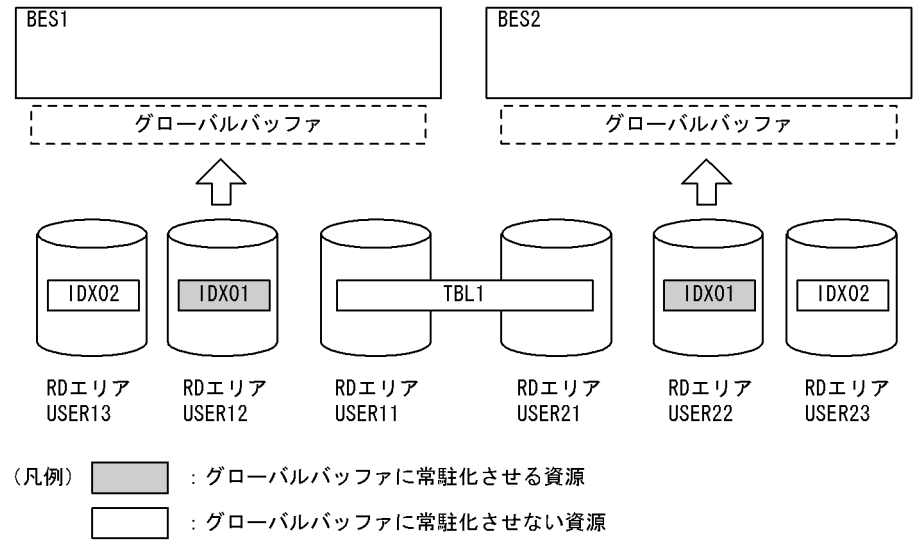
-t TBL1：読み込み対象となる表を指定します。

例４：

表（TBL1）のインデクスページを，インデクス単位でグローバルバッファに読み込みます。表，及びインデクスは次のように定義されているものとします。

```
・表定義
  CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
      IN ((USER11) C1>0,(USER21))
・インデクス定義
  CREATE INDEX IDX01 ON TBL1(C1) IN ((USER12),(USER22))
  CREATE INDEX IDX02 ON TBL1(C3) IN ((USER13),(USER23))
```

[概要]



●pdpgbfon のコマンド

```
pdpgbfon -k index -t TBL1 c:¥hirdb¥pdpgbfon¥control_file
```

[説明]

-k index：表のインデクスページを読み込む場合に指定します。

-t TBL1：読み込み対象となる表を指定します。

c:¥hirdb¥pdpgbfon¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdpgbfon¥control_file）の内容

```
idxname name=IDX01    [1]
```

[説明]

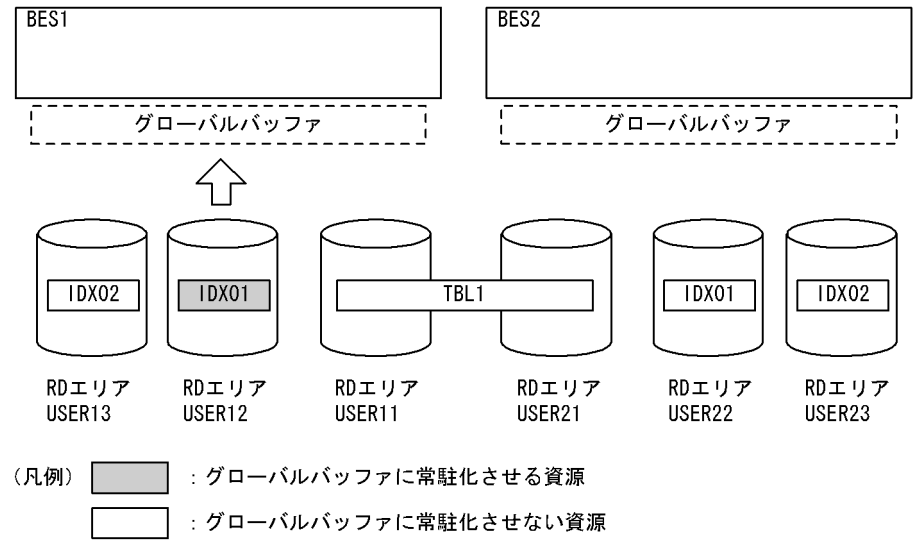
1. グローバルバッファへの読み込み対象となるインデクス名を指定します。

例５：

表（TBL1）のインデクスページを，RDエリア単位でグローバルバッファに読み込みます。表，及びインデクスは次のように定義されているものとします。

```
・表定義
  CREATE FIX TABLE TBL1(C1 INTEGER,C2 CHAR(10),C3 DEC(15))
      IN ((USER11) C1>0,(USER21))
・インデクス定義
  CREATE INDEX IDX01 ON TBL1(C1) IN ((USER12),(USER22))
  CREATE INDEX IDX02 ON TBL1(C3) IN ((USER13),(USER23))
```

[概要]



●pdpgbfon のコマンド

```
pdpgbfon -k index -t TBL1 c:¥hirdb¥pdpgbfon¥control_file
```

[説明]

-k index：表のインデクスページを読み込む場合に指定します。

-t TBL1：読み込み対象となる表を指定します。

c:¥hirdb¥pdpgbfon¥control_file：制御情報ファイル

●制御情報ファイル（c:¥hirdb¥pdpgbfon¥control_file）の内容

```
idxname name=IDX01 rdarea=USER12    [1]
```

[説明]

1. グローバルバッファへの読み込み対象となるインデクス名，及び RD エリア名を指定します。

12.3　コマンドの形式

12.3.1　形式

```
pdpgbfon -k 対象資源
         -t  {〔認可識別子.〕表識別子 | 〔認可識別子.〕all}
          〔-r RDエリア名〕
          〔-u 認可識別子〕
          〔-X サーバ間通信の応答監視時間〕
          〔-s サーバ名〕
          〔-b〕
          〔制御情報ファイル名〕
```

注

制御情報ファイル名を指定する場合，必ずオプションの最後に指定してください。

- **各オプションの関係**

各オプションの関係を次の表に示します。

表 12–3　各オプションの関係（pdpgbfon）

オプション	-k オプション		-t オプション	
	table	index	表識別子	all
-k 対象資源	–	–	◎	◎
-t {〔認可識別子.〕表識別子 | 〔認可識別子.〕all}	◎	◎	–	–
-r RD エリア名	○	×	○	×
-u 認可識別子	○	○	○	○
-X サーバ間通信の応答監視時間	○	○	○	○
-s サーバ名	○	○	○	○
-b	○	×	○	○
制御情報ファイル名	○	○	○	○

（凡例）

◎：必ず指定してください。

○：指定できます。

×：指定できません。

–：該当しません。

12.3.2　オプション

(1)　-k 対象資源

グローバルバッファへ読み込む資源（データページ又はインデクスページ）を指定します。

table：

読み込み対象がデータページの場合に指定します。

index：

読み込み対象がインデクスページの場合に指定します。

(2)　-t {〔認可識別子.〕表識別子｜〔認可識別子.〕all}

～<識別子>

対象となる資源の表名を指定します。

認可識別子を省略した場合，HiRDB に接続したときのユーザの名称が仮定されます。

all を指定した場合，その認可識別子（スキーマ）が持つすべての表，又はすべてのインデクスを読み込みます（ただし，一時表及び一時インデクスを除きます）。この場合，idxname 文は指定できません。

ディクショナリ表，ビュー表，一時表，及びユーザが定義した抽象データ型列がある表は指定できません。

(3)　-r RD エリア名

～<識別子>

-t オプションに指定した表を格納する RD エリアのうち，ある特定の RD エリアだけを読み込む場合，その RD エリア名を指定します。

このオプションは，-k table の場合に指定できます。-k index の場合は，idxname 文で指定します。

LOB 用 RD エリアは指定できません。また，一括指定は指定できません。

＜適用基準＞

HiRDB/パラレルサーバのサーバ間横分割表の場合，このオプションを指定しなくても複数のサーバプロセスで RD エリア単位に並行して処理がされます。しかし，HiRDB/シングルサーバの場合，及び HiRDB/パラレルサーバのサーバ内横分割表の場合，一つのサーバプロセスで 1RD エリアずつ処理がされるため，並行して処理されません。後者の場合に，RD エリア単位に複数のサーバプロセスを使用して並行に処理するときにこのオプションを指定します。

(4)　-u 認可識別子

pdpgbfon を実行するユーザの認可識別子を指定します。

認可識別子については，「-u 認可識別子」を参照してください。

(5)　-X サーバ間通信の応答監視時間

～＜符号なし整数＞　((1～65535)) 《300》

コマンドを実行したサーバで通信障害などの障害が発生した場合，コマンドは無応答となり，業務が停止するおそれがあります。pdpgbfon では障害を検知するために，コマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が，-X オプションに設定した時間を超えた場合は，pdpgbfon はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。

＜適用基準＞

- 通信障害やユニットダウンによってサーバから応答が返らない状態となった場合に，300 秒より早く障害を検知したいとき，-X オプションに 300 より小さい値を指定してください。
- 系切り替え機能を使用している場合，系切り替えが完了しているにもかかわらず，コマンドが応答を待ち続けることがあります。その場合は，監視時間を短く設定することで，コマンドを即時終了できます。
- 同時実行するアプリケーションやユティリティの多重度が高い場合など，ディクショナリからの応答が遅延し，-X オプションのデフォルトである 300 秒以内にユティリティの前処理が完了しないときは，監視時間でタイムアウトするおそれがあります。このような環境では，-X オプションには 300 より大きい値を指定してください。

(6)　-s サーバ名

～＜識別子＞

このオプションは HiRDB/パラレルサーバの場合に指定できます（HiRDB/シングルサーバの場合に指定しても無視されます）。

pdpgbfon の実行制御をするバックエンドサーバの名称を指定します。

＜適用基準＞

省略した場合でも，自動的にどれかのバックエンドサーバが選択されるため，通常時は指定する必要はありません。

pdpgbfon を大量に同時実行して，最大同時起動サーバプロセス数の上限を超えたときに指定します。

(7)　-b

読み込み対象が表のデータページで，その表に定義長 256 バイト以上の BINARY 型の列がある場合，別セグメントに分岐格納された BINARY 型列の格納データも読み込むときに指定します。

定義長 256 バイト以上の BINARY 型の列がある表の場合にこのオプションを省略したときは，BINARY 型列以外のデータ型の列と同じセグメントに格納された BINARY 型列は読み込みますが，別セグメントに分岐格納された BINARY 型列は読み込みません。

(8)　制御情報ファイル名

～＜パス名＞

pdpgbfon の制御文を記述した制御情報ファイルの名称を指定します。

制御情報ファイルには，次に示す制御文を記述できます。各制御文については，「idxname 文（インデクスの情報の記述）」，「option 文（オプションの情報の記述）」を参照してください。なお，制御情報ファイル中にコメントは記述できません。

- idxname 文
- option 文

12.3.3　idxname 文（インデクスの情報の記述）

グローバルバッファへの読み込み対象がインデクスページ（-k index）の場合，対象とするインデクスの情報を指定します。

＜適用基準＞

-k index 場合，-t オプションに指定した表に定義されているすべてのインデクスを読み込みますが，特定のインデクス，サーバ，又は RD エリアだけを読み込み対象にするとき，idxname 文を指定します。

＜規則＞

1. name オペランドは，server 及び rdarea オペランドよりも前に指定してください。
2. 識別子に空白及び英小文字を含む場合は，引用符（"）で囲んで指定してください。英小文字が引用符で囲まれていない場合，英大文字として扱われます。
3. rdarea オペランドを指定した idxname 文を指定し，ほかにも idxname 文を指定する場合，name オペランドだけを指定した idxname 文，及び server オペランドを指定した idxname 文は指定できません。
4. 同じインデクス識別子を指定した idxname 文は複数個指定できません。ただし，rdarea オペランドを指定する場合は複数個指定できます。

(1)　形式

```
idxname name= {インデクス識別子 | *}
        〔 {server=サーバ名〔, サーバ名〕…
           | rdarea=RDエリア名〔, RDエリア名〕…} 〕
```

(2) 説明

(a) name=｛インデクス識別子｜*｝

インデクス識別子 ～＜識別子＞

グローバルバッファに読み込むインデクス識別子を指定します。

認可識別子は，-t オプションに指定した表の認可識別子を仮定するため，ここでは指定する必要はありません。

*****

表に定義されているすべてのインデクスを対象にする場合，*を指定します。

*指定の idxname 文（server オペランドなし）と，idxname 文を省略した場合とでは，両方とも表のすべてのインデクスを読み込みます。ただし，*を指定して server オペランドも指定した idxname 文の場合は，表のすべてのインデクスのうち，server オペランドに指定したバックエンドサーバにあるインデクスだけを読み込み対象にできます。

*を指定した場合，idxname 文は 1 個だけ指定できます。

(b) server=サーバ名〔，サーバ名〕...

～＜識別子＞

このオペランドは，HiRDB/パラレルサーバの場合に指定できます。name オペランドに指定したインデクスに対して，特定サーバのインデクスを対象とする場合に，そのサーバ名（バックエンドサーバ名）を指定します。省略した場合は，インデクスが格納されているすべてのサーバを対象とします。

共用インデクスの場合，このオペランドの指定有無に関係なく，更新可能バックエンドサーバ，及びすべての参照専用バックエンドサーバで処理をします。

(c) rdarea=RD エリア名〔，RD エリア名〕...

～＜識別子＞

name オペランドに指定したインデクスに対して，特定のインデクス格納用 RD エリアを対象とする場合に，その RD エリア名を指定します。省略した場合は，インデクスが格納されているすべての RD エリアを対象とします。

name オペランドに*を指定した場合，このオペランドは指定できません。

12.3.4 option 文（オプションの情報の記述）

pdpgbfon の実行時間監視の情報を指定します。

＜適用基準＞

pdpgbfon の実行時間を監視する場合に指定します。

(1) 形式

```
option exectime=pdpgbfonの監視時間
```

(2) 説明

(a) exectime=pdpgbfon の監視時間 ～＜符号なし整数＞((0～35791394))

pdpgbfon の実行時間を監視する場合に，その監視時間を分単位で指定します。

＜指定値の目安＞

このオペランドは，無応答障害の検知を目的とするもので，長時間トランザクションの実行時間を監視するものではありません。したがって，指定値は，該当する表の処理時間に余裕を持たせた値にする必要があります。例えば，7～8 分程度で終了する pdpgbfon の実行時間を監視する場合は，exectime=10 ではなく，exectime=20 と指定してください。

＜規則＞

1. このオペランドに 0 を指定した場合，実行時間の監視はしません。
2. このオペランドを省略した場合は，次の優先順位でシステム定義の指定値に従います。
   1. pd_utl_exec_time
   2. pd_cmd_exec_time
3. 指定時間以内に pdpgbfon の処理が終了しない場合，ユティリティのプロセスを強制終了し，無応答原因の調査用に障害情報を取得してください。なお，実行時間の監視対象は，実際にグローバルバッファにアクセス要求をするシングルサーバ（又はバックエンドサーバ）と，pdrorg 用ユティリティサーバプロセス（0mrorg）であり，コマンドのプロセス（pdpgbfon 及び pdrorg）に障害が発生した場合は強制終了できません。取得する障害情報については，マニュアル「HiRDB システム定義」の pd_utl_exec_time オペランドを参照してください。

12.4　pdpgbfon の処理結果

12.4.1　pdpgbfon の処理結果の詳細

pdpgbfon が正常終了した場合，コマンドを実行した標準出力に処理結果が表示されます。

(1)　グローバルバッファへの読み込み対象がデータページの場合

```
pdpgbfon VV-RR     *** BUFFER INFORMATION ***     YYYY-MM-DD hh:mm:ss [1]

TABLE NAME : USERA.TBL1 [2]
 [3] [4]              [5]           [6]           [7]
No.  SERVER          RFGET         READ          RFFLS
  1  bes1             1,275          596              0
  2  bes2             1,271          594              0
  3  bes3             1,273          596              0

 8365 20:03:54 un01          KFPL00738-I Pdpgbfon terminated, return code=0
```

[説明]

1. 処理結果のヘッダ

   VV-RR：バージョン，リビジョン番号

   YYYY-MM-DD hh:mm:ss：pdpgbfon 開始日時

2. 処理対象の資源名

3. 通番

4. サーバ名

5. 参照 GET 回数※

   pdpgbfon がグローバルバッファを参照した回数です。

   管理（ディレクトリ）ページの参照回数も含まれるため，該当資源の使用ページ数より多い値が表示されます。なお，参照したページが既にグローバルバッファに読み込まれている場合は，実 READ は発生していません。

6. 実 READ 回数※

   pdpgbfon がグローバルバッファを参照したときに目的のページが見付からなかったために，DB に対して入力要求をした回数です。

7. 参照バッファフラッシュ回数※

   pdpgbfon がグローバルバッファ上にページを読み込むとき，グローバルバッファが不足したために LRU 方式で未更新ページをグローバルバッファから追い出した回数です。

   グローバルバッファ上に別の資源のページ情報がない状態（例えば HiRDB 開始直後など）で実行した場合に，この値が 0 以外のときは，全ページをメモリに読み込むにはグローバルバッファが不足していることを意味します。

注※

表示できる回数は最大 4,294,967,295 です。これを超える場合，オーバフローします。

(2) グローバルバッファへの読み込み対象がインデクスページの場合

```
pdpgbfon VV-RR     *** BUFFER INFORMATION ***     YYYY-MM-DD HH:MM:SS [1]

INDEX NAME : USERA.(PRIMARY0000131193)[2]
 [3] [4]              [5]            [6]           [7]
No.  SERVER           RFGET          READ          RFFLS
  1  bes1               75            69              0
  2  bes2               75            69              0
  3  bes3               75            69              0

INDEX NAME : USERA.IDX01
No.  SERVER           RFGET          READ          RFFLS
  1  bes1              525           478              0
  2  bes2              523           476              0
  3  bes3              524           477              0

 8373 20:04:04 un01          KFPL00738-I Pdpgbfon terminated, return code=0
```

[説明]

「グローバルバッファへの読み込み対象がデータページの場合」の説明を参照してください。なお，インデクス専用のグローバルバッファを割り当てた場合でも，該当するインデクスを格納する RD エリアの管理（ディレクトリ）ページへのアクセスは，RD エリアに割り当てたグローバルバッファ（指定がない場合は-o オプションに指定したグローバルバッファ）を使用します。そのため，インデクス専用のグローバルバッファが十分でも，RD エリアに割り当てたグローバルバッファが不足していると，参照バッファフラッシュ回数が表示されます。

12.5　注意事項

1. pdpgbfon は内部的に pdrorg を実行するため，同時実行可能数は pdrorg の上限と同じになります。つまり，pdpgbfon の同時実行可能数は，「pdrorg の同時実行可能数−実行中の pdrorg 数」となります。
2. ビュー表，及び一時表は処理対象外です。-t オプションにビュー表又は一時表を指定して実行すると，pdpgbfon は KFPL15231-E メッセージを出力して，エラー終了します。また，-t オプションに all を指定した場合，pdpgbfon は該当する表の処理をスキップします。
3. LOB 用 RD エリアは処理対象外です。したがって，抽象データ型，及びプラグインインデクスを格納している LOB 用 RD エリアも処理対象外となります。
4. 実行中の pdpgbfon を強制終了する場合は，pdcancel コマンドで強制終了してください。pdkill コマンドなどで強制終了しても，pdpgbfon は終了しません。
5. 共用表及び共用インデクスを読み込む場合，更新可能バックエンドサーバ，及びすべての参照専用バックエンドサーバが対象となります。
6. オープン状態でない SCHEDULE 属性の RD エリアに対して pdpgbfon を実行しても意味がありません（処理後に RD エリアがクローズされるため）。
7. HiRDB/パラレルサーバの場合，通信回線の混雑によってバックエンドサーバから pdpgbfon に対する送信電文が 1 秒以上遅れたときは，該当するバックエンドサーバの処理結果が標準出力に表示されないことがあります。この場合，メッセージログファイル，又はイベントログに出力される KFPL00714-I メッセージで処理結果を確認できます。
8. 処理対象表にユーザが定義した抽象データ型列がある場合，pdpgbfon は実行できません。
9. pdpgbfon のリターンコードを次の表に示します。

表 12–4　pdpgbfon のリターンコード

リターンコード	意味	対処方法
0	グローバルバッファ上への読み込みを完了しました。	なし。
4	グローバルバッファ上への読み込み時にバッファ面数が不足したため，バッファ落ち（読み込みできなかったページがあること）が発生しました。	必要であれば，グローバルバッファ面数を大きくしてください。 なお，一つのグローバルバッファを複数の資源で共有する場合は，pdpgbfon 実行前や実行中に別のジョブによって別資源のページがグローバルバッファに読み込まれていたりします。この場合，別資源のページを追い出したときでもバッファ落ちと認識されるため，処理対象資源がバッファ落ちしたのか，又は別資源がバッファ落ちしたのか判断できません。したがって，リターンコードだけで全ページがグローバルバッファに読み込まれたかどうかを判断する場合，pdpgbfon が終了するまで別のジョブは実行しないでください。
8	pdpgbfon が異常終了しました（グローバルバッファへの読み込みを中断しました）。	この場合，表やインデクスの状態は保証されます。異常終了後も，UAP から表やインデクスをアクセスできます。

リターンコード	意味	対処方法
		pdpgbfon を再実行する場合は，出力されたエラーメッセージを参照してエラー要因を取り除いてください。
12	pdrorg が異常終了しました（pdpgbfon の延長で動作する pdrorg が異常終了しました）。	この場合，表やインデクスの状態は保証されます。異常終了後も，UAP から表やインデクスをアクセスできます。 出力されたエラーメッセージを参照してエラー要因を取り除いて，再度 pdpgbfon を実行してください。

10. pdpgbfon を実行している表やインデクスに対して，定義系 SQL は実行しないでください。実行すると，pdpgbfon が異常終了します。スキーマ単位に pdpgbfon を実行している場合は，該当スキーマが持つすべての表とインデクスが対象となります。
11. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdpgbfon の制御情報ファイルには，BOM を持つファイルを使用できます。なお，制御情報ファイルに BOM を持つファイルを使用しても，BOM は読み飛ばされます。

13 整合性チェックユティリティ（pdconstck）

この章では，参照制約又は検査制約が定義された表に対して整合性チェック及び検査保留状態の操作をする，整合性チェックユティリティ（pdconstck）について説明します。

13.1　概要

整合性チェックユティリティ（pdconstck）は，参照制約又は検査制約が定義された表に対して，整合性チェック及び検査保留状態の操作（設定又は解除）をするユティリティです。

13.1.1　前提条件

(1)　動作条件

pdconstck 実行時の前提条件を次の表に示します。

表 13–1　pdconstck 実行時の前提条件

項目	前提条件
実行権限	DBA 権限を持つユーザ，又は表の所有者が実行できます。
RD エリアの状態	pdconstck と処理対象 RD エリアの状態の関係については，「コマンド実行時の RD エリアの状態」を参照してください。
同時実行	同一表に対する pdconstck の同時実行はできません。同時実行した場合，該当する表の資源種別名 TABL（資源種別：0002）と DICT（資源種別：3005）間でデッドロックとなることがあります。 対象となる表が異なる場合は，pdconstck を同時実行できます。ただし，検査保留状態を変更する場合，排他待ちとなります。
最大同時実行数	pdconstck の最大同時実行数は，pd_max_users オペランドの指定値÷2 となります。
資源数	pdconstck は，対象表及び外部キーが参照する被参照表に対して排他を掛けます。そのため，次のシステム定義の指定値に注意してください。 • 同時アクセス可能実表数 システム定義の pd_max_access_tables オペランドには，（対象表の外部キー数＋ 1）以上の値を指定してください。 • 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表及び RD エリア数の総和 システム定義の pd_lck_until_disconnect_cnt オペランドには，次の計算式 X 以上の値を指定してください。 X=（2 ＋処理対象表の外部キー数＋処理対象表の格納 RD エリア数 ＋外部キーが参照する主キーの表の格納 RD エリアの総数） • サーバ当たりの排他制御用プールサイズ システム定義の pd_lck_pool_size オペランドに記載している計算式を元に設定してください。
検査保留状態の遷移	システム定義の pd_check_pending オペランドの指定に関係なく，pdconstck の実行によって検査保留状態は遷移します。

(2)　動作環境

pdconstck の動作環境を次の表に示します。

表 13–2　pdconstck の動作環境

項目		動作環境
HiRDB 環境	コマンド実行場所	pdconstck は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
	サーバの状態	pdconstck を実行する場合，次のサーバが稼働していなければなりません。 HiRDB/シングルサーバの場合：シングルサーバ HiRDB/パラレルサーバの場合：ディクショナリサーバ，フロントエンドサーバ，及びバックエンドサーバ（マルチフロントエンドサーバの場合，一つ以上のフロントエンドサーバが稼働中である必要があります）
	環境変数 PDDIR	PDDIR 環境変数に，絶対パス名で HiRDB 運用ディレクトリを設定してください。
	PDCONFPATH	PDCONFPATH 環境変数に，絶対パス名で HiRDB システム定義ファイルを格納するディレクトリを設定してください。
OS 環境	-o オプションで指定するファイル及びディレクトリ	pdconstck 実行ユーザに対してアクセス権限を与えてください。 なお，-o オプションを省略した場合は，表「pdconstck がファイルを出力するディレクトリ」に示すディレクトリにファイルを作成すると解釈されるため，同様にアクセス権限を与えてください。また，パス名は大文字と小文字を区別しません。このため，大文字と小文字が異なるだけの同じパス名を指定した場合は，上書きされます。

表 13–3　pdconstck がファイルを出力するディレクトリ

-o オプションの指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	-o オプションに指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※　コマンド実行環境の環境変数の指定

13.1.2　pdconstck 実行時の認可識別子

pdconstck 実行時には，HiRDB に接続する認可識別子，並びに処理対象表及び制約名称の認可識別子を指定できます。HiRDB に接続する認可識別子，並びに処理対象表及び制約名称の認可識別子を次の表に示します。

表 13–4　HiRDB に接続する認可識別子，並びに処理対象表及び制約名称の認可識別子

<table>
<tr><th colspan="4">指定条件</th><th colspan="2">HiRDB との接続</th><th rowspan="3">処理対象表又は制約名称の認可識別子</th></tr>
<tr><th rowspan="2">HiRDB に接続する認可識別子（-u オプション）</th><th colspan="2">環境変数 PDUSER</th><th rowspan="2">処理対象表又は制約名称の認可識別子（-t 又は-c オプション）</th><th rowspan="2">認可識別子</th><th rowspan="2">パスワード</th></tr>
<tr><th>識別子</th><th>パスワード</th></tr>
<tr><td rowspan="8">指定あり</td><td rowspan="4">指定あり</td><td rowspan="2">指定あり</td><td>指定あり</td><td rowspan="4">-u オプションで指定した認可識別子</td><td rowspan="8">応答付きメッセージに対して入力※</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>-u オプションで指定した認可識別子</td></tr>
<tr><td rowspan="2">省略</td><td>指定あり</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>-u オプションで指定した認可識別子</td></tr>
<tr><td colspan="2" rowspan="2">環境変数 PDUSER 省略</td><td>指定あり</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>-u オプションで指定した認可識別子</td></tr>
<tr><td rowspan="10">指定なし</td><td rowspan="4">指定あり</td><td rowspan="2">指定あり</td><td>指定あり</td><td rowspan="4">環境変数 PDUSER に指定した認可識別子</td><td rowspan="2">環境変数 PDUSER に指定したパスワード</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>環境変数 PDUSER に指定した認可識別子</td></tr>
<tr><td rowspan="2">省略</td><td>指定あり</td><td rowspan="2">省略</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>環境変数 PDUSER に指定した認可識別子</td></tr>
<tr><td colspan="2" rowspan="2">識別子，パスワードの両方に簡易認証キーワードを指定</td><td>指定あり</td><td>簡易認証キーワード</td><td rowspan="2">簡易認証キーワード</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>ログインウィンドウのユーザ名</td></tr>
<tr><td colspan="2" rowspan="2">環境変数 PDUSER 省略</td><td>指定あり</td><td rowspan="2">ログインウィンドウのユーザ名（認可識別子）</td><td rowspan="2">応答付きメッセージに対して入力※</td><td>-t 又は-c オプションで指定した認可識別子</td></tr>
<tr><td>指定なし</td><td>ログインウィンドウのユーザ名（認可識別子）</td></tr>
</table>

注※

パスワード入力要求の応答付きメッセージが表示されます。応答付きメッセージに対して，パスワードを入力してください。なお，応答付きメッセージに対して空送信した場合，パスワードを省略したと解釈されます。

13.2　pdconstck の機能

pdconstck には，次の二つの機能があります。

整合性チェック機能

参照制約又は検査制約が定義された表の，制約の整合性を検査します。また，制約の違反の有無によって，表の検査保留状態を解除又は設定します。

検査保留状態の強制変更機能

表の検査保留状態を強制的に設定又は解除します。

各機能の詳細を次に説明します。

13.2.1　整合性チェック機能

整合性チェック機能は，参照制約又は検査制約の整合性をチェックし，制約に違反する行がない場合は検査保留状態を解除する機能です。

参照制約の整合性チェック

参照制約に基づいて，表の外部キーの値が，その表が参照する表の主キーと同じ値を持つかどうかをチェックします。チェックの結果，制約に違反する行がない場合は，検査保留状態を解除します。制約に違反する行がある場合は，検査保留状態を設定します。ただし，外部キー値にナル値を含む行はチェック対象としません。なお，参照制約の整合性チェックは，データベースの容量によっては時間が掛かることがあります。実行時間については，「参照制約の整合性チェック方法」の SQL から見積もってください。

検査制約の整合性チェック

検査制約の定義に基づいて，行の値が検査制約の範囲内であるかどうかをチェックします。チェックの結果，制約に違反する行がない場合は，検査保留状態を解除します。制約に違反する行がある場合は，検査保留状態を設定します。なお，検査制約の整合性チェックは，データベースの容量によっては時間が掛かることがあります。実行時間については，「検査制約の整合性チェック方法」の SQL から見積もってください。

(1)　整合性チェック機能の実行単位

整合性チェック機能は，次のどちらかの単位で実行します。

- 表単位
  表に定義された参照制約及び検査制約の，すべての制約を対象とします。
- 制約単位
  一つの制約だけを対象とします。

(a)　表単位

表に定義された参照制約及び検査制約のうち，検査保留状態となったすべての制約に対して，制約ごとに整合性チェックをします。表単位で整合性チェックをする場合は，-t オプションを指定します。

表単位の整合性チェックは，次の場合に適用します。

- 表に定義された参照制約及び検査制約のうち，検査保留状態となったすべての制約について，1 回の pdconstck で違反があるかどうかを検査し，検査保留状態を解除する場合。
- pdmod で RD エリアを再初期化して，ディクショナリ表の検査保留状態が解除されなかった場合など，RD エリアの検査保留状態から，表の検査保留状態及び制約の検査保留状態を解除するとき（詳細については，「再初期化後の表，及びインデクスの状態」を参照してください）。

(b)　制約単位

検査保留状態に関係なく，一つの制約だけの整合性を検査します。制約単位で整合性を検査する場合は，-c オプションを指定します。

制約単位の整合性チェックは，次の場合に適用します。

- 個々の制約について，違反があるかどうかをチェックする場合。

(2)　参照制約の整合性チェック

(a)　参照制約の整合性チェック方法

参照制約の整合性チェックでは，制約ごとに次の SQL で整合性をチェックします。

- 参照制約の整合性チェックで実行される SQL

```
SELECT  DISTINCT ”外部キー構成列1”, ”外部キー構成列2”, …, ”外部キー構成列n”
FROM ”参照表の所有者名”.”参照表の表識別子” RIC_REF_TBL
WHERE (”外部キー構成列1” IS NOT NULL AND
       ”外部キー構成列2” IS NOT NULL AND … ”外部キー構成列n” IS NOT NULL)
AND NOT EXISTS (SELECT * FROM ”被参照表の所有者名”. ”被参照表の表識別子” RIC_RFD_TBL
WHERE (RIC_RFD_TBL.”主キー構成列1”, RIC_RFD_TBL.”主キー構成列2”,
        …, RIC_RFD_TBL.”主キー構成列n”) =
      (RIC_REF_TBL.”外部キー構成列1”, RIC_REF_TBL.”外部キー構成列2”,
        …, RIC_REF_TBL.”外部キー構成列n”))
ORDER BY  ”外部キー構成列1” ASC|DESC, ”外部キー構成列2” ASC|DESC,
          …, ”外部キー構成列n” ASC|DESC
WITHOUT LOCK NOWAIT
```

注

この SQL では作業表を作成する場合があるため，シングルサーバ又はバックエンドサーバに，作業表を格納する作業表用ファイルが必要となります。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(b)　整合性有無の判断

(a)の SQL の実行結果（SQLCODE）と整合性の有無の関係を，次の表に示します。

表 13–5　SQL の実行結果と整合性の有無の関係（参照制約）

SQLCODE（検索行数）	整合性の有無
100（0 件）	整合性あり
0（1 件以上）	整合性なし
上記以外	実行エラー

(c)　制約違反となったキー値の出力

(a)の SQL を実行した結果，制約違反となったキー値を処理結果ファイルに出力します。なお，制約違反となったキー値は重複排除して出力します。キー値の出力形式については，「参照制約の整合性チェックの処理結果」を参照してください。

(d)　制約違反キー値の出力数上限値による整合性チェックの中断

制約違反となった行データ数が制約違反キー値の出力数上限値を超えた場合，該当する制約の整合性チェックを終了します。制約違反キー値の出力数上限値は，-w オプションで変更できます。

(3)　検査制約の整合性チェック

(a)　検査制約の整合性チェック方法

検査制約の整合性チェックでは，制約ごとに次の SQL で整合性をチェックします。

- 検査制約の整合性チェックで実行される SQL

```
SELECT DISTINCT※ ”検査制約の対象列1”, ”検査制約の対象列2”, …,  ”検査制約の対象列n”
FROM ”検査制約を定義した表の所有者名”.”検査制約を定義した表の表識別子”
WHERE NOT（検査制約の探索条件） WITHOUT LOCK NOWAIT
```

注

この SQL では作業表を作成する場合があるため，シングルサーバ又はバックエンドサーバに，作業表を格納する作業表用ファイルが必要となります。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

注※

探索条件に指定した列定義が BLOB 型又は BINARY 型の場合，「DISTINCT」は指定しません。また，この場合は作業表を作成しません。

(b)　整合性有無の判断

(a)の SQL の実行結果（SQLCODE）と整合性の有無の関係を，次の表に示します。

表 13–6　SQL の実行結果と整合性の有無の関係（検査制約）

SQLCODE（検索行数）	整合性の有無
100（0 件）	整合性あり
0（1 件以上）	整合性なし
上記以外	実行エラー

(c)　制約違反となったキー値の出力

(a)の SQL を実行した結果，制約違反となったキー値を処理結果ファイルに出力します。なお，探索条件に指定した列のデータ型が BLOB 型又は BINARY 型を含まない場合，制約違反となったキー値は重複排除して出力します。BLOB 型又は BINARY 型を含む場合は，制約違反となったキー値の重複排除をしません。キー値の出力形式については，「検査制約の整合性チェックの処理結果」を参照してください。

(d)　制約違反キー値の出力数上限値による整合性チェックの中断

制約違反となった行データ数が制約違反キー値の出力数上限値を超えた場合，該当する制約の整合性チェックを終了します。制約違反キー値の出力数上限値は，-w オプションで変更できます。

(4)　整合性チェック結果による検査保留状態の遷移

pdconstck は，制約ごとに整合性チェックをした結果に基づいて，整合性の有無で表の検査保留状態，制約の検査保留状態，又は RD エリアの検査保留状態を，検査保留状態又は非検査保留状態に設定します。RD エリアの検査保留状態の遷移は，pdconstck の実行単位によって異なります。

次に，制約の検査保留状態の遷移，表の検査保留状態の遷移，及び RD エリアの検査保留状態の遷移について説明します。なお，検査制約と参照制約はそれぞれ別に，表の検査保留状態，制約の検査保留状態，又は RD エリアの検査保留状態を遷移します。

制約の検査保留状態の遷移

制約の検査保留状態は，制約ごとの整合性チェックの結果によって，制約ごとに遷移します。

制約の検査保留状態の遷移を次の表に示します。

表 13–7　制約の検査保留状態の遷移

制約の整合性チェック結果	制約の検査保留状態（設定する値）
整合性あり	非検査保留状態（NULL 値）
整合性なし	検査保留状態（'C'）

表の検査保留状態の遷移

表の検査保留状態は各制約の検査保留状態によって遷移します。表の検査保留状態の遷移を次の表に示します。

表 13–8　表の検査保留状態の遷移

制約の検査保留状態	表の検査保留状態（設定する値）
すべての制約が非検査保留状態	非検査保留状態（NULL 値）
検査保留状態と非検査保留状態の制約が混在	検査保留状態（'C'）

RD エリアの検査保留状態の遷移

RD エリアの検査保留状態は，各制約で行った整合性チェックの結果によって遷移します。RD エリアの検査保留状態の遷移を次の表に示します。

表 13–9　RD エリアの検査保留状態の遷移

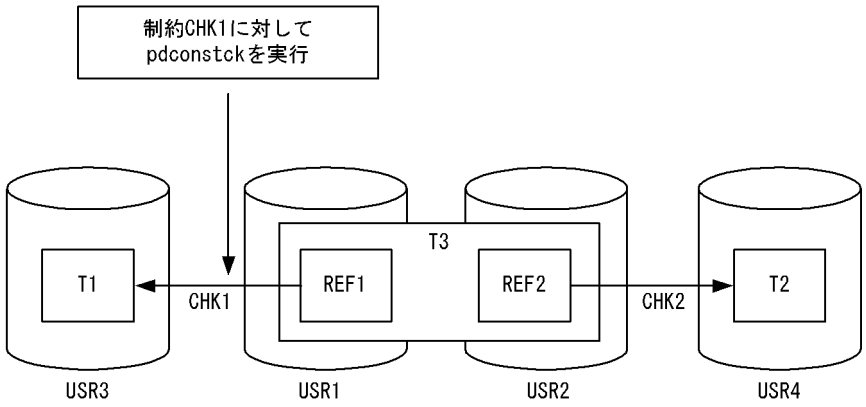
整合性チェックの結果	RD エリアの検査保留状態（設定する値）
すべての制約が整合性あり	非検査保留状態（NULL 値）
整合性がない制約と，整合性がある制約が混在	検査保留状態（'C'）

(a)　表単位で実行した場合の検査保留状態の遷移

表単位実行時は，制約ごとに整合性チェックをした結果によって，表の検査保留状態，制約の検査保留状態，及び RD エリアの検査保留状態を遷移します。

参照制約が定義された表に対して，表単位に整合性チェックを実行する例を次の図に示します。また，この例の検査保留状態の遷移を次の表に示します。

図 13–1　表単位での整合性チェック実行例



（凡例）

T1，T2：主キーが定義された表

T3：外部キー（REF1，REF2）が定義された表

CHK1，CHK2：参照制約

USR1～USR4：表を格納しているユーザ用 RD エリア

REF1，REF2：ほかの表を参照する外部キー

表 13–10　表単位での検査保留状態の遷移（実行例）

整合性チェックの結果		検査保留状態				
REF1 の整合性	REF2 の整合性	表の検査保留状態	制約の検査保留状態		表格納用 RD エリアの検査保留状態	
			REF1	REF2	USR1	USR2
あり	あり	N	N	N	N	N
	なし	P	N	P	P	P
なし	あり	P	P	N	P	P
	なし	P	P	P	P	P

（凡例）

P：検査保留状態

N：非検査保留状態

(b)　制約単位での検査保留状態の遷移

制約単位実行時は，指定した制約の整合性チェックをした結果によって，表の検査保留状態，指定した制約の検査保留状態，及び RD エリアの検査保留状態を遷移します。検査保留状態の遷移を次に説明します。

検査保留状態の遷移

指定した制約の整合性結果によって，制約の検査保留状態を遷移します。

図「表単位での整合性チェック実行例」の実行例について，制約 CHK1 に対して制約単位に整合性チェックをした場合の検査保留状態の遷移を次の表に示します。

表 13–11　制約単位での検査保留状態の遷移（実行例）

整合性チェックの結果	制約の検査保留状態	検査保留状態				
CHK1 の整合性	CHK2	表の検査保留状態	制約の検査保留状態		RD エリアの検査保留状態	
			CHK1	CHK2	U1	U2
あり	非検査保留状態	N	N	−	N	N
あり	検査保留状態	P	N	−	P	P
なし	非検査保留状態	P	P	−	P	P
なし	検査保留状態	P	P	−	P	P

（凡例）

P：検査保留状態

N：非検査保留状態

−：状態を変更しません

(5)　整合性チェックの順番と実行有無

(a)　表単位

整合性チェックの順番

表単位で整合性チェックをする場合，整合性チェックの順番は次のようになります。

参照制約と検査制約の順番

-t オプションで指定した表に参照制約及び検査制約の両方が定義されている場合，参照制約（外部キー），検査制約の順で整合性をチェックします。

同一制約内の順番

-t オプションで指定した表に同一制約が複数定義されている場合は，同一制約内の制約名称順に整合性をチェックします。

表単位の整合性チェックでの順番の例を次の表に示します。

表 13–12　表単位の整合性チェックでの順番の例

表の制約		整合性チェックの順番			
参照制約（外部キー）	検査制約	参照制約（外部キー）		検査制約	
		F1	F2	C1	C2
あり	あり	1.	2.	3.	4.
	なし	1.	2.	–	–
なし	あり	–	–	1.	2.
	なし	–	–	–	–

（凡例）

F1，F2：参照制約（外部キー）の制約名

C1，C2：検査制約の制約名

1.～4.：整合性チェックの検査順番

–：該当しません

整合性チェックの実行有無

表単位の整合性チェックは，検査保留状態となった制約だけ整合性を検査します。整合性をチェックする条件と整合性チェック実行有無を次の表に示します。

表 13–13　整合性をチェックする条件と整合性チェック実行有無

検査保留状態			整合性チェック実行有無	備考
表の検査保留状態	制約の検査保留状態	表格納用 RD エリアの検査保留状態※1		
検査保留状態（'C'）	検査保留状態（'C'）	検査保留状態	あり	pdconstck は正常状態と判断します（検査保留状態）。

検査保留状態			整合性チェック実行有無	備考
表の検査保留状態	制約の検査保留状態	表格納用 RD エリアの検査保留状態※1		
		非検査保留状態	あり	RD エリアが非検査保留状態であるが，ディクショナリ表が検査保留状態のため，整合性をチェックします。
	非検査保留状態（NULL値）※2	検査保留状態	なし	pdconstck は正常状態と判断します（他制約が検査保留状態）。
		非検査保留状態	あり	RD エリアが非検査保留状態であるが，ディクショナリ表が検査保留状態のため，整合性をチェックします。
非検査保留状態（NULL値）	検査保留状態（'C'）	検査保留状態	なし	ディクショナリ表が不正のため，pdconstck をエラー終了します。
		非検査保留状態	なし	
	非検査保留状態（NULL値）※2	検査保留状態	あり	RD エリアが検査保留状態のため，整合性をチェックします。
		非検査保留状態	なし	pdconstck は正常状態と判断します（非検査保留状態）。

注※1

分割表の場合，表を格納する各 RD エリア中の表情報によって，検査保留状態又は非検査保留状態になります。各表格納用 RD エリア中の表情報による検査保留状態を次の表に示します。

表 13–14　各表格納用 RD エリア中の表情報による検査保留状態

条件	表格納 RD エリアの表情報
すべての格納 RD エリアの RD エリアの検査保留状態が検査保留状態	検査保留状態
格納 RD エリアの RD エリアの検査保留状態が検査保留状態と非検査保留状態が混在	検査保留状態
すべての格納 RD エリアが RD エリアの検査保留状態が非検査保留状態	非検査保留状態

注※2

表の検査保留状態と制約の検査保留状態の整合性が取れていない場合は，pdconstck はエラーで終了します。ディクショナリ表の検査保留状態での pdconstck の動作を次の表に示します。

表 13–15　ディクショナリ表の検査保留状態での pdconstck の動作

検査保留状態		pdconstck の動作
表の検査保留状態	制約の検査保留状態	
検査保留状態	検査保留状態となった制約がある	処理続行

検査保留状態		pdconstck の動作
表の検査保留状態	制約の検査保留状態	
	すべての制約が非検査保留状態	エラー終了
非検査保留状態	検査保留状態となった制約がある	エラー終了
	すべての制約が非検査保留状態	処理続行

(b)　制約単位

整合性チェックの順番

制約単位の場合は，制約だけ整合性を検査するため，チェックの順番はありません。

整合性チェックの実行有無

制約単位の整合性チェックでは，表の検査保留状態に関係なく，上記の順に整合性をチェックします。

13.2.2　検査保留状態の強制変更機能

(1)　機能概要

検査保留状態の強制変更機能は，参照制約又は検査制約の整合性を検査しないで，検査保留状態を強制的に変更する機能です。検査保留状態の強制変更機能には，検査保留状態の強制設定，又は強制解除の二つがあります。

検査保留状態の強制設定

検査保留状態の強制設定は，表の検査保留状態を強制的に検査保留状態に設定する機能です。この機能を使用する場合は，-k set を指定します。

検査保留状態の強制設定は，次の場合に適用します。

- システム定義の pd_check_pending オペランドの指定値を NOUSE から USE に変更した場合，参照制約又は検査制約の制約の整合性が不明確なとき。
- システム定義の pd_check_pending オペランドを指定していないシステムで，バージョン 07-03 以降の HiRDB にバージョンアップした場合に，参照制約又は検査制約の整合性が不明確なとき。
- 一時的に，ユーザが表を操作できないように規制する場合。
- 整合性チェックの結果，制約違反があった場合に，検査保留状態の強制解除で検査保留状態を解除して，SQL で制約違反となったデータを修正した後，再度整合性チェックをするとき。

検査保留状態の強制解除

検査保留状態の強制解除は，表の検査保留状態を強制的に非検査保留状態に設定する（検査保留状態を解除する）機能です。この機能を使用する場合は，-k release を指定します。

検査保留状態の強制解除は，次の場合に適用します。

- ユーザ側で制約の整合性を確認できていて，整合性チェックが不要な場合など，強制的に検査保留状態を解除し運用する場合。
- 参照制約で違反した行を，SQL で参照表の外部キーの更新又は削除によって修正する場合。
- 検査制約で違反した行を SQL で修正する場合。

(2) 検査保留状態の強制変更機能の実行単位

検査保留状態の強制変更は，表に定義された参照制約及び検査制約のすべての制約を更新対象とする「表単位」，又は一つの制約だけを更新対象とする「制約単位」に実行できます。

検査保留状態の強制変更を実行すると，ディクショナリ表の検査保留状態の情報が変更されます。ディクショナリ表の検査保留状態の変更箇所を次の表に示します。

表 13–16 ディクショナリ表の検査保留状態の変更箇所

実行単位	-c オプションの指定	表の制約		ディクショナリ表の検査保留状態の変更箇所			
		参照制約	検査制約	SQL_TABLES 表		SQL_REFERENTIAL_CONSTRAINTS 表	SQL_CHEKS 表
				CHECK_PEND 列	CHECK_PEND2 列	CHECK_PEND 列	CHECK_PEND2 列
表単位	該当しません	なし	あり	−	○	−	○※1
		あり	なし	○	−	○※1	−
		あり	あり	○	○	○※1	○※1
制約単位	参照制約	あり	なし	○	−	○※2	−
		あり	あり	○	−	○※2	−
	検査制約	なし	あり	−	○	−	○※2
		あり	あり	−	○	−	○※2

（凡例）

○：検査保留状態を変更します。

−：検査保留状態を変更しません（現在の状態を維持します）。

注※1

表に定義されたすべての参照制約又は検査制約の検査保留状態を変更します。

注※2

-c オプションに指定した制約だけ検査保留状態を変更します。

RD エリア中の表情報の検査保留状態の変更箇所を次の表に示します。

表 13–17　RD エリア中の表情報の検査保留状態の変更箇所

実行単位	-c オプションの指定	表の制約		RD エリア中の表情報の検査保留状態の変更箇所	
		参照制約	検査制約	参照制約の状態	検査制約の状態
表単位	該当しません	なし	あり	−	○
		あり	なし	○	−
		あり	あり	○	○
制約単位	参照制約	あり	なし	○	−
		あり	あり	○	−
	検査制約	なし	あり	−	○
		あり	あり	−	○

（凡例）

○：検査保留状態を変更します。

−：検査保留状態を変更しません（現在の状態を維持します）。

(a)　表単位

表単位の場合，定義された参照制約又は検査制約のすべてを変更対象とします。表単位で実行する場合は，-t オプションを指定します。

表単位は次の場合に適用します。

- 表に定義された参照制約及び検査制約のすべての制約について，1 回の pdconstck で検査保留状態を変更する場合。

(b)　制約単位

表に定義された参照制約又は検査制約のうち，一つの制約だけ変更対象とします。制約単位に実行する場合，-c オプションを指定します。

制約単位は，個々の制約について検査保留状態を変更する場合に適用します。

(3)　検査保留状態の強制設定

表の検査保留状態，制約の検査保留状態，及び RD エリアの検査保留状態を検査保留状態に設定します。

(4)　検査保留状態の強制解除

表の検査保留状態，制約の検査保留状態，及び RD エリアの検査保留状態を非検査保留状態に設定します。なお，pdconstck の実行単位，表の検査保留状態，制約の検査保留状態，及び RD エリアの検査保留状態によって，設定内容（解除可否）が変わります。実行単位ごとの説明を次に示します。

表単位での検査保留状態の強制解除

表単位の場合，無条件に表の検査保留状態，制約の検査保留状態，及び RD エリアの検査保留状態を非検査保留状態に設定します。

制約単位での検査保留状態の強制解除

制約単位の場合，-c オプションで指定した制約の検査保留状態を非検査保留状態に設定します。また，ほかの制約の検査保留状態によって，表及び RD エリアの検査保留状態は変わらないこともあります。検査保留状態の遷移を次の表に示します。

表 13–18　制約単位での検査保留状態の遷移（検査保留状態の強制解除）

条件		検査保留状態（表及び RD エリア）
-c オプションに指定した制約以外の制約が定義されている場合	検査保留状態の制約があるとき	遷移なし
	すべての制約が非検査保留状態のとき	非検査保留状態
-c オプションに指定した制約だけ表に定義されている場合		非検査保留状態

13.3　使用例

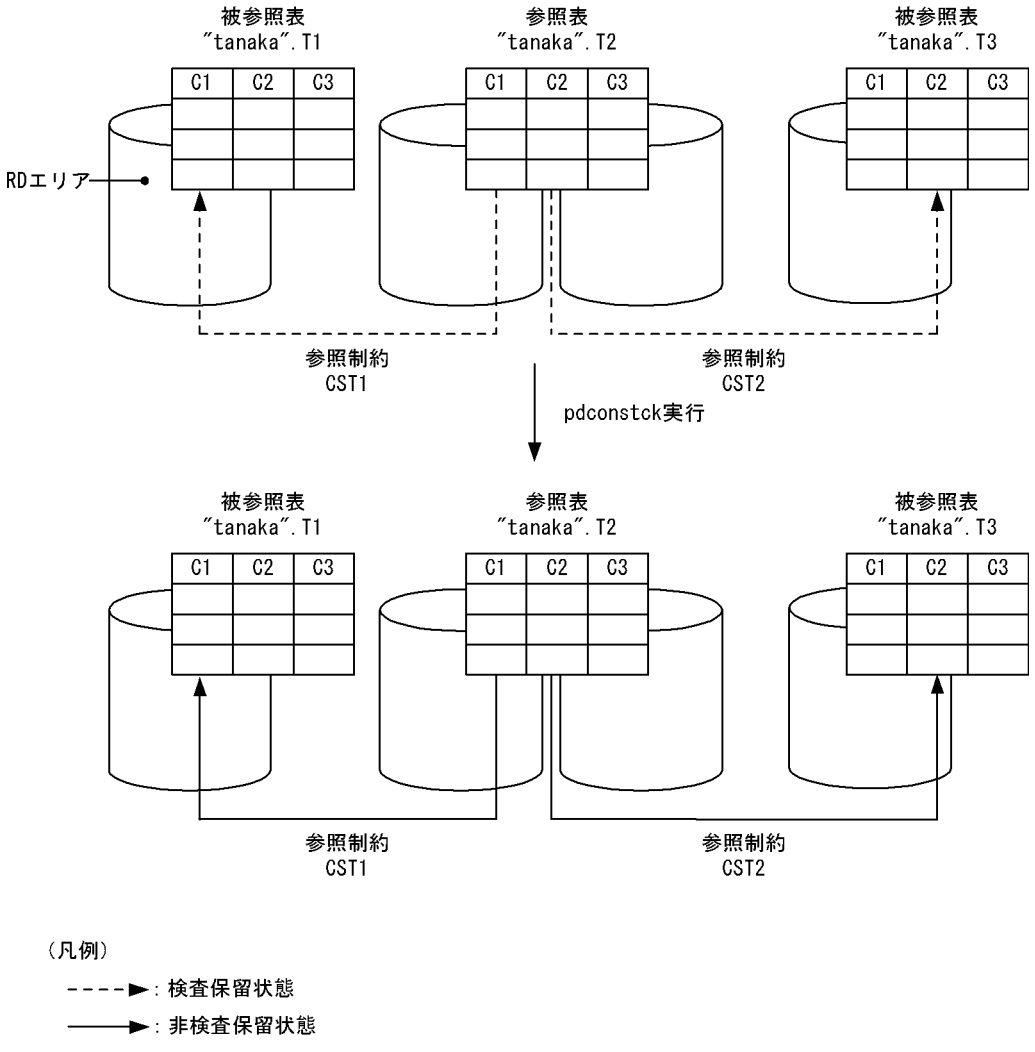
pdconstck の使用例を次に示します。

13.3.1　表単位での整合性チェック

表T2に定義されたすべての制約（CST1及びCST2）に対して，整合性チェックをします。なお，表T2は検査保留状態になっていると仮定します。
・表定義

```
T1:  CREATE TABLE T1(C1 INT PRIMARY KEY)
T3:  CREATE TABLE T3(C2 INT PRIMARY KEY)
T2:  CREATE TABLE T2(C1 INT, C2 INT,
                     CONSTRAINT CST1 FOREIGN KEY (C1) REFERENCES T1,
                     CONSTRAINT CST2 FOREIGN KEY (C2) REFERENCES T3)
```

[概要]



（凡例）

----►：検査保留状態

——►：非検査保留状態

●実行するコマンド

```
pdconstck -k check -t T2
```

[説明]

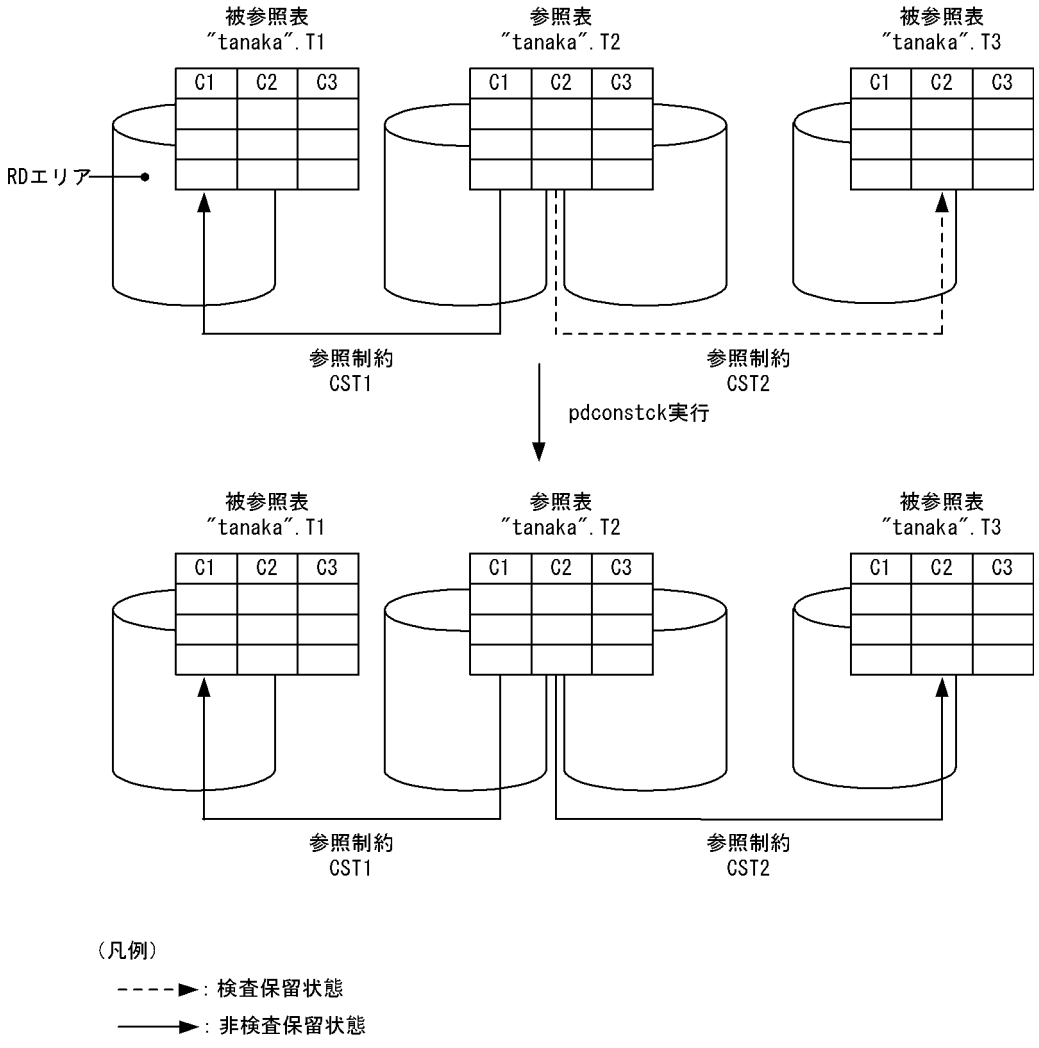
-k check：整合性チェック機能の実行

-t T2：整合性チェック機能の対象となる表名の指定

13.3.2　制約単位での整合性チェック

表T2に定義された制約（CST2）の整合性チェックをします。なお，表定義は前述の例と同じで，表T2は検査保留状態となっていると仮定します。

[概要]



●実行するコマンド

```
pdconstck -k check -c CST2
```

[説明]

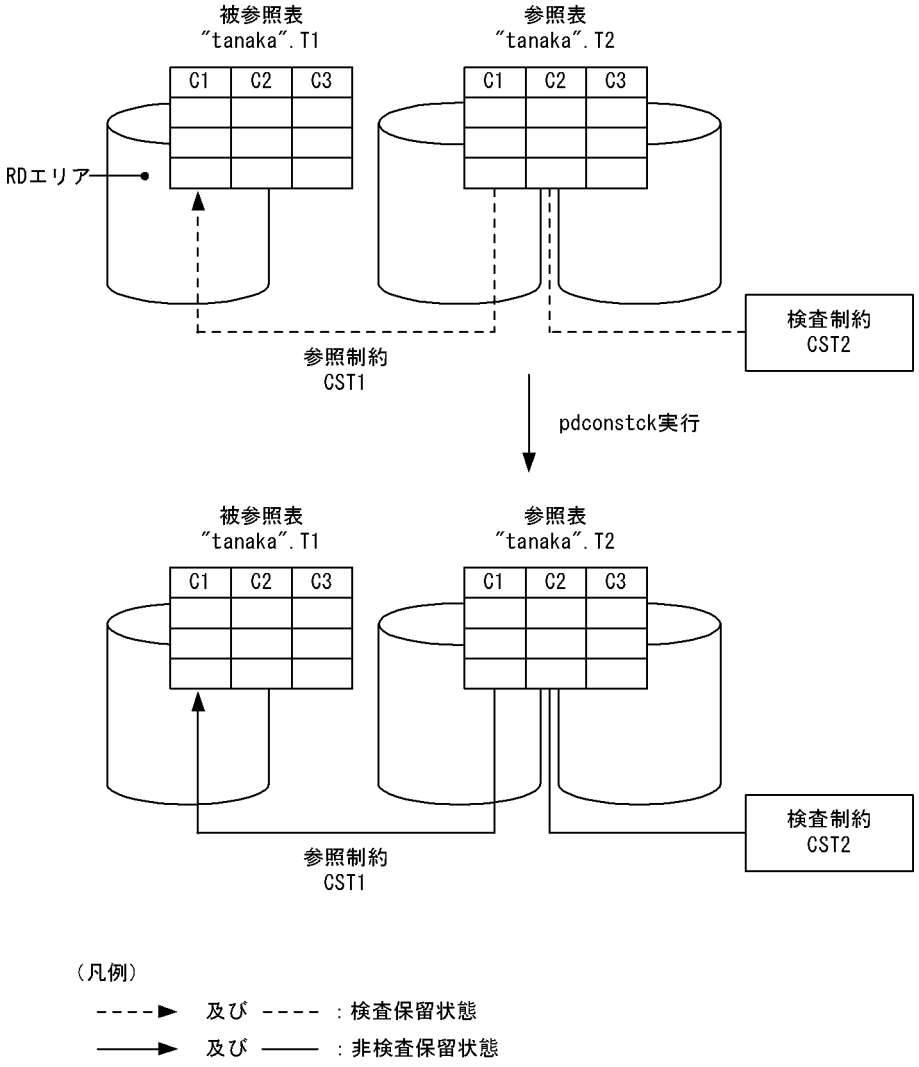
-k check：整合性チェック機能の実行

-c CST2：整合性チェック機能の対象となる制約名の指定

13.3.3　表単位での検査保留状態の強制設定

表T2に定義されたすべての制約（CST1，CST2）を検査保留状態に設定します。なお，表T2は非検査保留状態になっていると仮定します。
・表定義
```
T1:  CREATE TABLE T1(C1 INT PRIMARY KEY)
T2:  CREATE TABLE T2(C1 INT, C2 INT,
                        CONSTRAINT CST1 FOREIGN KEY (C1) REFERENCES T1,
                        CONSTRAINT CST2 CHECK (c2 > 10))
```

[概要]



（凡例）

----► 及び ---- ：検査保留状態

——► 及び —— ：非検査保留状態

●実行するコマンド

```
pdconstck -k set -t T2
```

[説明]

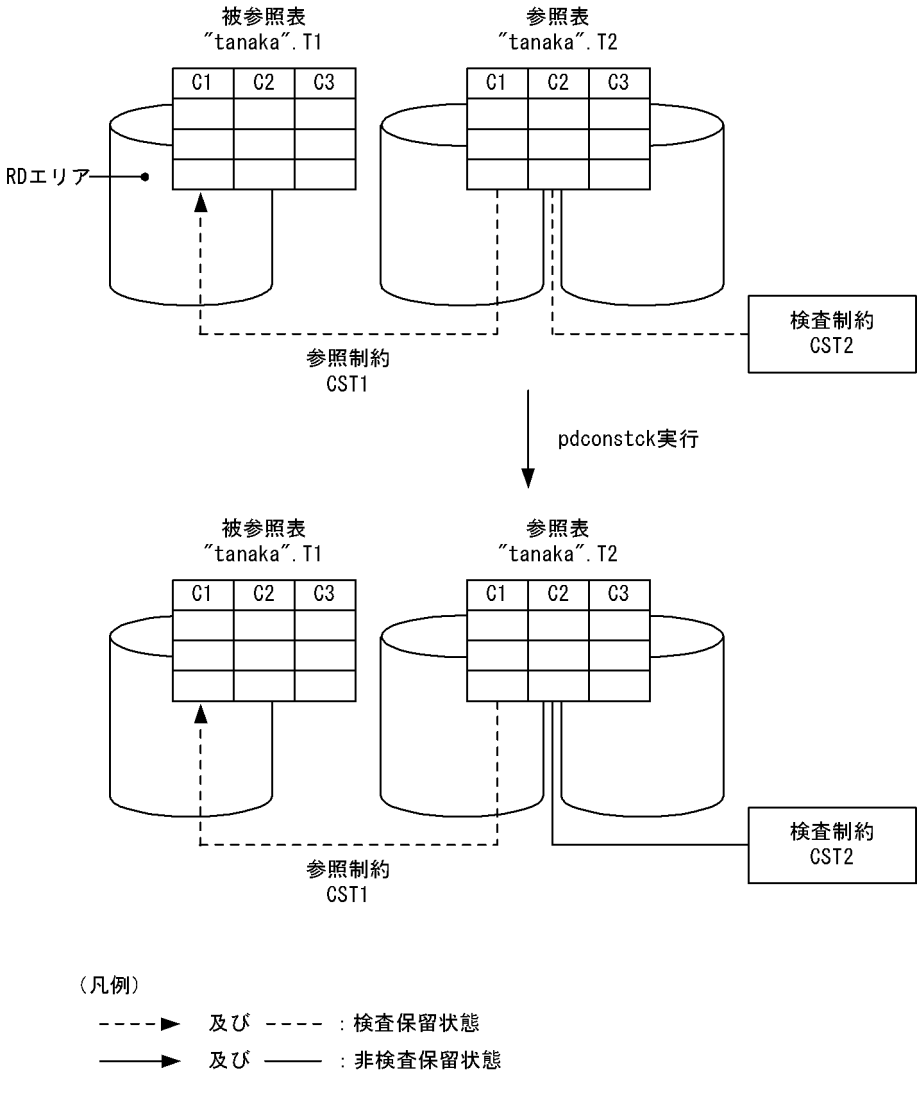
-k set：検査保留状態の強制設定機能の実行

-t T2：検査保留状態の強制設定機能の対象となる表名の指定

13.3.4　制約単位での検査保留状態の強制解除

表T2の制約（CST2）の検査保留状態を強制解除します。なお，表定義は前述の例と同じで，制約（CST2）は検査保留状態になっていると仮定します。

[概要]



（凡例）

----► 及び ---- ：検査保留状態

——► 及び —— ：非検査保留状態

●実行するコマンド

```
pdconstck -k release -c CST2
```

[説明]

-k release：検査保留状態の強制解除機能

-c CST2：検査保留状態の強制解除機能の対象となる制約名の指定

13.4　コマンドの形式

13.4.1　形式

```
pdconstck -k　{check｜set｜release}
          {-t　〔認可識別子.〕表識別子
            ｜-c　〔認可識別子.〕制約名}
          〔-u 認可識別子〕
          〔-o 処理結果ファイル名〕
          〔-w 制約違反キー値の出力数上限値〕
          〔-x 整合性チェックの実行監視時間〕
```

13.4.2　オプション

(1)　-k　{check ｜ set ｜ release}

pdconstck の機能を指定します。

check

整合性チェック機能を使用する場合に指定します。

set

検査保留状態の強制変更機能（検査保留状態の強制設定）を使用する場合に指定します。

release

検査保留状態の強制変更機能（検査保留状態の強制解除）を使用する場合に指定します。

(2)　-t 〔認可識別子.〕表識別子

～＜識別子＞

処理対象とする参照表又は検査制約表の表識別子を指定します。

認可識別子を省略した場合は，HiRDB に接続したときのユーザの認可識別子が仮定されます。HiRDB に接続したときのユーザの認可識別子ついては，「pdconstck 実行時の認可識別子」を参照してください。

＜適用基準＞

整合性チェック機能，又は検査保留状態の強制変更機能を，表単位に実行する場合に指定します。

＜規則＞

1. 参照制約（外部キー），又は検査制約が定義されている表だけ指定できます。
2. ビュー表は指定できません。

3. 認可識別子，又は表識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(3) -c〔認可識別子.〕制約名

～＜識別子＞

処理対象とする制約の制約名を指定します。

認可識別子を省略した場合は，HiRDB に接続したときのユーザの認可識別子が仮定されます。HiRDB に接続したときのユーザの認可識別子ついては，「pdconstck 実行時の認可識別子」を参照してください。

＜適用基準＞

整合性チェック機能，又は検査保留状態の強制変更機能を，制約単位に実行する場合に指定します。

＜規則＞

1. 認可識別子，又は制約名称を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(4) -u 認可識別子

～＜識別子＞

HiRDB に接続する認可識別子を指定します。

このオプションを指定した場合，パスワード入力要求の応答付きメッセージが表示されます。パスワードが不要な場合には，応答付きメッセージに対して空送信してください。このオプションを省略した場合の認可識別子，及びパスワードの仮定値については，「pdconstck 実行時の認可識別子」を参照してください。

＜適用基準＞

環境変数 PDUSER に設定されている認可識別子ではなく，別の認可識別子で HiRDB に接続する場合に指定します。

＜規則＞

1. 認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

＜注意事項＞

パスワードの応答ができない環境で実行するときは，必ず PDUSER を設定してください。

(5) -o 処理結果ファイル名

～＜パス名＞

処理結果を出力する処理結果ファイルのファイル名を指定します。

＜適用基準＞

意図したディレクトリに処理結果ファイルを作成したい場合に指定します。

＜規則＞

1. pdconstck を実行するホスト上のパス名を指定します。
2. パス名に指定したファイル（上位ディレクトリ）には，pdconstck 実行者から書き込み権限が与えられている必要があります。

＜省略時仮定値＞

このオプションを省略した場合，次のファイル名で処理結果ファイルが作成されます。

- 処理結果ファイルが作成されるディレクトリ

  %PDDIR%¥tmp ディレクトリ
- 処理結果ファイルのファイル名

  CONSTCK-REPORT-表名称-xxxxxxxxx

  表名称：

  -t オプションに指定した表識別子，又は-c オプションに指定した制約が定義されている表識別子（1～30 文字）

  表名称に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

  xxxxxxxxx：

  OS 関数（tmpnam 関数）で得られたユニークな文字（9 文字）

(6)　-w 制約違反キー値の出力数上限値

～＜符号なし整数＞((1～30000))≪60≫

整合性チェック機能では，制約違反となったキー値を処理結果ファイルに出力します。このオプションでは，制約違反となったキー値の数について，制約ごとに出力する件数の上限値を指定します。

＜適用基準＞

制約違反となったキー値の出力を抑止し，ファイル容量を削減したい場合に指定します。

＜規則＞

1. 制約違反となったキー値は，重複排除して出力されます。ただし，検査制約の場合に，検査制約に関連する列のデータ型が BLOB 又は BINARY の列を含む場合は，重複排除しません。
2. 制約違反となったキー値の数が指定値，又は仮定値を超えた場合，以降の制約違反となったキー値は，処理結果ファイルに出力されません。また，該当する制約の整合性チェックを中止して，次の制約の整合性チェックをします。

(7)　-x 整合性チェックの実行監視時間

～＜符号なし整数＞((0～65535))≪0≫

整合性チェックの前処理，及び 1 制約に対する整合性チェックの実行監視時間を秒単位で指定します。整合性チェックの前処理，又は 1 制約に対する整合性チェックの実行時間が，このオプションで指定した実行監視時間を超えた場合，次のどちらかのメッセージを出力して pdconstck はリターンコード 8 で終了します。

- KFPL24512-E（整合性チェックの前処理）
- KFPA11732-E（1 制約に対する整合性チェック）

＜適用基準＞

次のような障害が発生すると，pdconstck が無応答状態になることがあるため，運用上，実行時間の監視が必要な場合にこのオプションを指定してください。

- 通信障害（pdconstck と HiRDB サーバ，及び HiRDB サーバ間の通信障害）
- ディスク障害などでのプロセスの沈み込み

＜規則＞

システム定義の pd_lck_wait_timeout オペランドと-x オプションとの組み合わせによって，pdconstck の動作が変わります。組み合わせと pdconstck の動作を次に示します。

pd_lck_wait_timeout オペランドの指定値	-x オプションの指定値	pdconstck の動作
0	0	HiRDB サーバから応答が返るまで待ち続けます。
	上記以外	-x オプションの指定値を超えた場合，リターンコード 8 で終了します。
上記以外	0	pd_lck_wait_timeout オペランドの指定値を超えた場合，リターンコード 8 で終了します。
	pd_lck_wait_timeout オペランドの指定値より小さい	-x オプションの指定値を超えた場合，リターンコード 8 で終了します。
	pd_lck_wait_timeout オペランドの指定値と同じ	pd_lck_wait_timeout オペランドの指定値を超えた場合，リターンコード 8 で終了します。
	pd_lck_wait_timeout オペランドの指定値より大きい	

13.5　処理結果

pdconstck の処理結果を次に示します。

13.5.1　pdconstck の処理結果

pdconstck の処理結果には，整合性チェックの処理結果，及び検査保留状態の情報が出力されます。
pdconstck の処理結果を次に示します。

```
          [1]                                    [2]      [3]
pdconstck VV-RR *** DB VALIDATION CHECK *** yyyy-mm-dd hh:mm:ss
整合性チェックの処理結果 [4]
検査保留状態の情報 [5]
pdconstck TERMINATED, RETURN CODE=0[6]
```

[説明]

1. HiRDB のバージョン番号
2. pdconstck 開始日付
3. pdconstck 開始時刻
4. 整合性チェックの処理結果
5. 検査保留状態の情報
6. pdconstck のリターンコード

13.5.2　整合性チェックの処理結果

整合性チェックの処理結果には，整合性チェック機能の実行結果が出力されます。この情報は，-k check を指定した場合に出力されます。

整合性チェックの処理結果の出力例を次に示します。

```
** DB VALIDATION CHECK INFORMATION **
参照制約の整合性チェックの処理結果 [1]
検査制約の整合性チェックの処理結果 [2]
```

[説明]

1. 参照制約の整合性チェックの処理結果
2. 検査制約の整合性チェックの処理結果

   検査制約の整合性チェックを実行した処理結果が出力されます。この情報は，-t オプションを指定した場合，表に検査制約が定義されている場合，又は-c オプションに検査制約の制約名を指定した場合に出力されます。

(1)　参照制約の整合性チェックの処理結果

参照制約の整合性チェックの処理結果は，-t オプションを指定して表に参照制約が定義されている場合，又は-c オプションに参照制約の制約名を指定した場合に，制約名の順に出力されます。

参照制約の整合性チェックの処理結果の出力例を次に示します。

```
      ** REFERENTIAL CONSTRAINT INFORMATION **
      CONSTRAINT    = SCHEMA1.CONST1 [1]
      REFERENCED  TABLE = SCHEMA1.P1 [2]
      REFERENCING TABLE = SCHEMA1.F1 [3]
      [5]              [6]
[4] ┌ NO.  FOREIGN KEY COLUMN NAME
    │ ---- ------------------------------
    │  1   COL1
    └  2   COL2

      実行結果　[7]
```

[説明]

1. 制約名（認可識別子.制約名）
2. 被参照表の表名称（認可識別子.表識別子）
3. 参照表の表名称（認可識別子.表識別子）
4. 外部キーの列情報
5. 定義順番
6. 列名（外部キー構成列の定義順）
7. 実行結果

   整合性チェックをした制約の実行結果です。整合性チェックをしなかった制約については出力されません。

   参照制約の整合性チェックの結果，違反するキーがなかった場合の実行結果，及び違反するキーがあった場合の実行結果の出力例を次に示します。

違反するキーがなかった場合の実行結果

```
 NO ERROR FOREIGN KEY [1]
```

[説明]

1. 違反する外部キーがないことを示す情報

違反するキーがあった場合の実行結果

```
      [2]          [3]
[1] ┌ NO.   ERROR FOREIGN KEY
    │ ----- -----------------
    │    1   1234567890
    │        tanaka
    │    2   777
```

```
    └       yamada
      PROCESSING DISCONTINUED [4]
      EXIST ERROR FOREIGN KEY [5]
```

[説明]

1. 違反キー値の情報（外部キーが参照する主キーのキー順）
2. 違反キーの通番（1 から昇順）

   外部キーの列名順に違反キーが出力されます。外部キーの構成列が複数ある場合は，最初のキー値だけ出力されます。
3. 違反キーの値

   違反キー値は既定文字集合の文字形式で出力されます（文字集合が指定された文字データ型の場合は，既定文字集合に変換して出力されます）。

   なお，文字データ型の場合は，違反キー値の先頭から最大 70 バイト分だけ出力されます。各国文字が 71 バイト目になる場合は，その各国文字は出力されません。
4. 整合性チェックの処理の中止を示す情報
5. 違反する外部キーがあることを示す情報

(2) 検査制約の整合性チェックの処理結果

検査制約の整合性チェックの処理結果は，-t オプションを指定して表に検査制約が定義されている場合，又は-c オプションに検査制約の制約名を指定した場合に出力されます。

検査制約の整合性チェック処理結果の出力例を次に示します。

```
      ** CHECK CONSTRAINT INFORMATION **
      CONSTRAINT = SCHEMA1.CONST2 [1]
      TABLE      = SCHEMA1.TABLE1 [2]
      [4]            [5]
[3] ┌ NO.    CHECK KEY COLUMN NAME
    │ ---- ------------------------------
    │  1   COL1
    └  2   COL2

      実行結果 [6]
```

[説明]

1. 制約名（認可識別子.制約名）
2. 表名（認可識別子.表識別子）
3. 探索条件中の列情報（ディクショナリ表 SQL_CHECK_COLUMNS の格納順）
4. 列情報の通番
5. 列名
6. 実行結果

整合性チェックをした制約の実行結果です。整合性チェックをしなかった制約については出力されません。

検査制約の整合性チェックの結果，違反するキーがなかった場合の実行結果，及び違反するキーがあった場合の実行結果の出力例を次に示します。

違反するキーがなかった場合の実行結果

```
 NO ERROR CHECK KEY [1]
```

[説明]

1. 違反する外部キーがないことを示す情報

違反するキーがあった場合の実行結果

```
      [2]          [3]
[1] ┌ NO.   ERROR CHECK KEY
    │----- -----------------
    │    1   1234567890
    │        tanaka
    │    2   777
    └        yamada
      PROCESSING DISCONTINUED [4]
      EXIST ERROR CHECK KEY [5]
```

[説明]

1. 違反キーの値（検査制約の整合性チェック順）
2. 違反キーの通番（1 から昇順）
3. 違反キーの値

   違反キー値は既定文字集合の文字形式で出力されます（文字集合が指定された文字データ型の場合は，既定文字集合に変換して出力されます）。

   検査制約を構成する列のデータ型が BLOB 及び BINARY の場合，違反キー値は 16 進数文字形式で，違反キー値の先頭から最大 35 バイト分だけ出力されます。検査制約を構成する列のデータ型が BLOB 型及び BINARY 型以外の場合は，違反キー値は文字形式で出力されます。

   なお，文字データ型の場合は，違反キー値の先頭から最大 70 バイト分だけ出力されます。各国文字が 71 バイト目になる場合は，その各国文字は出力されません。
4. 整合性チェックの処理の中止を示す情報
5. 違反する外部キーがあることを示す情報

13.5.3 検査保留状態情報

-k オプションの指定に関係なく，表の検査保留状態，及び制約の検査保留状態が出力されます。検査保留状態情報の出力例を次に示します。なお，各 RD エリアの検査保留状態は出力されません。各 RD エリアの検査保留状態については，pddbst の RD エリア単位の状態解析（論理的解析），又は表単位の状態解析で確認してください。

```
** CHECK PENDING STATUS INFORMATION **
                                              [2]            [3]
TABLE                                    CONSTRAINT TYPE PENDING STATUS
SCHEMA1.T5 [1]
                                         REFERENCE       PENDING
                                         CHECK           RELEASE
                                               [2]            [3]
CONSTRAINT                               CONSTRAINT TYPE PENDING STATUS
SCHEMA1.CONST1 [4]                       REFERENCE       PENDING
SCHEMA2.CONST2 [4]                       CHECK           RELEASE
```

[説明]

1. 表名
2. 制約の種別

   REFERENCE：参照制約（参照制約が定義されていない場合は出力されません）

   CHECK：検査制約（検査制約が定義されていない場合は出力されません）
3. 検査保留状態

   PENDING：検査保留状態

   RELEASE：非検査保留状態
4. 制約名

   該当表に定義している参照制約，検査制約の順で，かつ制約名の昇順に出力されます。

13.6　注意事項

1. pdconstck のリターンコードの意味と対処方法を次の表に示します。

表 13–19　pdconstck のリターンコードの意味と対処方法

リターンコード	意味	対処方法
0	正常終了しました。	なし。
4	正常終了しましたが，整合性チェックの結果，整合性を満たさない制約がありました（制約違反となったキー値の数が上限値を超えて，制約の整合性チェックを中止した場合も含みます）。	検査保留状態となっている制約の整合性を修正して，再実行してください。
8	異常終了しました。	必要に応じて，エラーの原因を取り除いて，再実行してください。

注

KFPL50001-I メッセージでリターンコードが 0 又は 4 を出力後，pdconstck の終了処理で異常が発生した場合，リターンコードは 8 ではなく，0 又は 4 が出力されます。

2. セキュリティ監査機能使用時に pdconstck を実行した場合，実行後のリターンコードによって取得される監査証跡が異なります。

0 又は 4 の場合：

CREATE AUDIT の WHENEVER に SUCCESSFUL 又は ANY を指定している場合，監査証跡が取得されます。

8 の場合：

CREATE AUDIT の WHENEVER に UNSUCCESSFUL 又は ANY を指定している場合，監査証跡が取得されます。

3. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdconstck で出力されるファイルには BOM は付きません。

14 統計解析ユティリティ（pdstedit）

この章では，HiRDB システムの稼働情報などの統計情報を編集する統計解析ユティリティ（pdstedit）について説明します。

14.1　概要

14.1.1　統計解析ユティリティ（pdstedit）の概要

実行者　HiRDB 管理者が実行できます

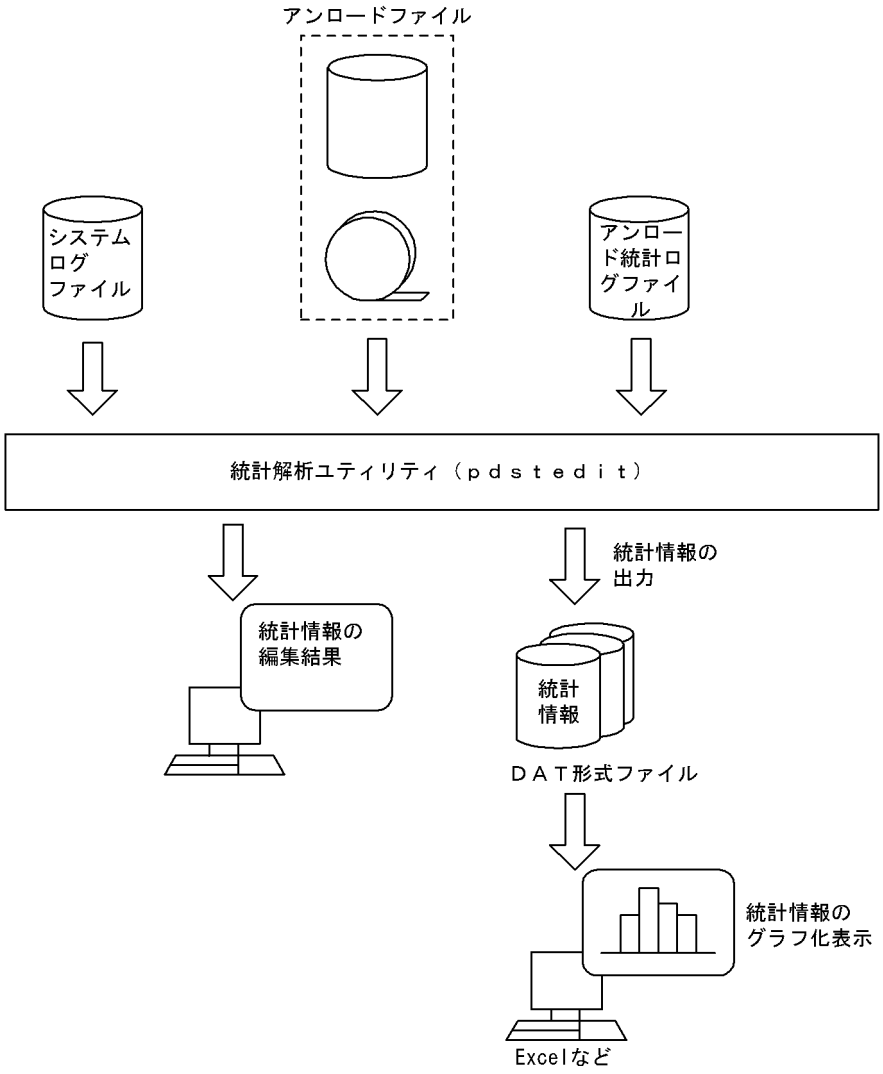
統計解析ユティリティは，統計入力アンロードファイル，及びシステムログファイルから情報を入力し，統計情報を編集するユティリティです。

HiRDB システムの処理効率を向上させるには，データベースを最適に設計する必要があり，その際には HiRDB システムの稼働状況が容易に分かると便利です。

統計解析ユティリティを使用すれば，編集開始位置と終了位置を指定することで必要な統計情報だけを編集，出力できます。ユーザは，この統計情報を解析することで HiRDB システムの稼働状況を知ることができます。また，標準出力には統計情報の編集結果を出力し，DAT 形式ファイルには編集前の統計情報を出力します。したがって，DAT 形式ファイルの情報を利用し，ユーザ任意の統計レポートも作成できます。

統計解析ユティリティ（pdstedit）の概要を次の図に示します。

図 14–1　統計解析ユティリティ（pdstedit）の概要



(1)　統計情報の基になるファイル

統計解析ユティリティが入力するファイルには，次に示す三つのファイルがあります。

・アンロード統計ログファイル

統計ログファイルをアンロードしたファイルのことをいいます。

・アンロードログファイル

システムログファイルをアンロードしたファイルのことをいいます。

・システムログのファイルグループ

システムログファイルの論理的な単位のことをいいます。統計解析ユティリティを起動するホストから参照できる，物理ファイルを持つファイルグループが入力対象となります。

(2) 統計情報の種類

統計解析ユティリティが編集する統計情報の種類を次の表に示します。なお，統計情報（CONNECT/DISCONNECT に関する統計情報を除く）は，システム定義の pdstbegin オペランドを指定するか，又は pdstbegin コマンドを実行すると取得されます。CONNECT/DISCONNECT に関する統計情報は，システム定義の pdhibegin -k cnc オペランドを指定して取得します。

表 14–1　統計解析ユティリティが編集する統計情報の種類

統計情報の種別	概要
システムの稼働に関する統計情報	HiRDB システムでのプロセス，RPC，ログなど，システムの稼働に関する稼働情報を，HiRDB システム単位及びサーバ単位に編集し，出力します。
UAP に関する統計情報※1	検索行数，各 SQL 実行回数など，UAP に関する情報を，編集時間単位に編集し，UAP 又はサービスごとに出力します。
SQL に関する統計情報※1	検索行数，作業表作成回数，SQL 内分割コマンドを発行したバックエンドサーバ数など，SQL の発行に関する情報を UAP 又はサービスごとに編集し，出力します。
グローバルバッファに関する統計情報※2	バッファヒット率，実入出力回数など，グローバルバッファのアクセスに関する情報を，編集時間単位に編集し，サーバ単位及びグローバルバッファ単位に出力します。
データベース操作に関する HiRDB ファイルの統計情報※2	同期入出力回数，入出力エラー回数など，HiRDB ファイルへのアクセスに関する情報を編集時間単位に編集し，サーバ単位，HiRDB ファイル単位，及び RD エリア単位に出力します。
デファードライト処理に関する統計情報	デファードライト処理の回数，動作要因，入出力並列度など，デファードライト処理に関する情報を編集時間単位に編集し，サーバ単位に出力します。
インデクスに関する統計情報※3	統計ログ及びシステムログを基にして，インデクスキー排他情報，インデクススプリット情報など，インデクスに関する情報を編集時間単位に編集し，サーバ単位及びインデクス単位に出力します。なお，この情報は，DAT 形式ファイルには出力されません。
SQL 静的最適化に関する統計情報	SQL の静的最適化情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。
SQL 動的最適化に関する統計情報	SQL の動的最適化情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。
SQL オブジェクト実行に関する統計情報	SQL オブジェクトの実行情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。
SQL 文の履歴に関する統計情報	発行された操作系 SQL，定義系 SQL，及び LOCK 文の情報，並びに SQL に関する情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。
CONNECT/DISCONNECT に関する統計情報※4	CONNECT/DISCONNECT に関する情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。

統計情報の種別	概要
SQL オブジェクト転送に関する統計情報	SQL オブジェクトの転送情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。
データベースの入出力に関する統計情報	データベースの入出力に関する統計情報を出力します。なお，この情報は，DAT 形式ファイルにだけ出力します。

注※1

UAP に関する統計情報と SQL に関する統計情報は，pdstbegin，pdstend コマンドを入力するタイミングによっては情報が一致しない場合があります。また，UAP から実際に発行している SQL の数と表示する数とが一致しない場合があります。

注※2

-t オプションの時間範囲内にシンクポイントが発生していない場合又はシンクポイントが 1 回だけ発生している場合，その時間帯に対しての情報は出力されません。このような場合には，-t オプションの時間範囲を大きく設定する必要があります。

注※3

インデクスに関する統計情報の，インデクススプリットの情報は，アンロードログファイル，又はシステムログのファイルグループから情報を取得します。したがって，インデクススプリットの情報を編集したい場合は，アンロードログファイル，又はシステムログのファイルグループを入力としてください。そのほかの情報はアンロード統計ログファイルから情報を取得します。

注※4

CONNECT/DISCONNECT に関する統計情報は，アンロードログファイル，又はシステムログのファイルグループから情報を取得します。このため，アンロードログファイル，又はシステムログのファイルグループを入力としてください。なお，この統計情報は，システム定義の pdhibegin -k cnc オペランドを指定した場合，シングルサーバ，又はフロントエンドサーバがあるサーバに出力されます。

(3) 統計解析ユティリティ実行時の条件

1. 統計解析ユティリティは，HiRDB の稼働に関係なく実行できます。
2. 統計解析ユティリティは，任意のサーバマシンで実行してください。

14.2　コマンドの形式

14.2.1　形式

```
pdstedit 〔-k 編集項目〔，編集項目〕…〕
         〔-m 時間間隔〕
         〔-t 〔開始時刻〕〔，終了時刻〕〕
         〔-u UAP名〔，UAP名〕…〕
         〔-x ホスト名〔，ホスト名〕…〕
         〔-s サーバ名〔，サーバ名〕…〕
         〔-i {統計入力アンロードファイル名|HiRDBファイルシステム領域名|統計入力アンロードファイル格納ディレクトリ名}〕
         〔-o DAT形式ファイル出力先ディレクトリ名〕
         〔-w ワーク用一時ファイル作成ディレクトリ名〕
         〔-d 制御文ファイル名〕
         〔-b〕
         〔-e 拡張フォーマット指定値〔，拡張フォーマット指定値〕〕
         〔-y〕
```

14.2.2　オプション

(1)　-k 編集項目〔，編集項目〕...

～《sys》

編集する項目を指定します。

sop，dop，pcd，obj，sqh，cnc，及び dio を指定した場合，必ず-o オプションも指定してください（指定しなかった場合はエラーとなります）。

sop，dop，pcd，obj，sqh，cnc，及び dio を指定した場合，DAT 形式ファイルには統計情報を出力しますが，統計情報の編集出力はしません。また，all を指定して-o オプションを指定しなかった場合，エラーにはならないで処理は続行します。

sys：

システムの稼働に関する統計情報

svr：

サーバごとのシステムの稼働に関する統計情報

uap：

UAP に関する統計情報

sql：

SQL に関する統計情報

buf：

グローバルバッファプールに関する統計情報

fil：

データベース操作に関する HiRDB ファイルの統計情報

dfw：

デファードライト処理に関する統計情報

idx：

インデクスに関する統計情報

sop：

SQL 静的最適化に関する統計情報

dop：

SQL 動的最適化に関する統計情報

pcd：

SQL オブジェクト実行に関する統計情報

obj：

SQL オブジェクト転送に関する統計情報

sqh：

SQL 文の履歴に関する統計情報

cnc：

CONNECT/DISCONNECT に関する統計情報

dio：

データベースの入出力に関する統計情報

all：

上記のすべての統計情報（ただし，sop，dop，pcd，obj，sqh，cnc，及び dio については，DAT 形式ファイルへの出力だけとなります）

(2)　-m 時間間隔

～<符号なし整数>((1～1440))《60》

統計情報を編集するときに，各統計情報の合計値を出す時間間隔を，分単位で指定します。ここで指定した時間間隔で各統計情報の合計値を出力します。

なお，標準出力に統計情報の編集結果を出力する場合に，このオプションの指定値が有効になります。DAT 形式ファイルに出力する場合は，このオプションの指定は無視されます。

(a) 注意事項

このオプションには，統計情報を統計ログファイルに出力する間隔と同じ値か又は倍数での指定を推奨します。統計ログファイルに出力する間隔より小さい値を指定すると，該当する編集時間範囲内に解析対象のログレコードがない場合，NO DATA が出力されます。統計ログファイルに出力する間隔については，システム定義の pd_statistics，pdstbegin オペランド，及び pdstbegin コマンドを参照してください。

(3) -t 〔開始時刻〕〔，終了時刻〕

統計情報の出力範囲を，ログレコードの出力時刻の開始時刻と終了時刻で指定します。開始時刻及び終了時刻の指定範囲は，1970 年 1 月 1 日 0 時 0 分 0 秒から当年当月当日の現在時刻までです。

このオプションを省略した場合，統計入力アンロードファイル，及びシステムログファイル内のすべての統計情報が編集されます。

開始時刻を省略した場合，統計入力アンロードファイル，及びシステムログファイルの先頭から指定した終了時刻までが編集対象となります。終了時刻を省略した場合，指定した開始時刻から統計入力アンロードファイル，及びシステムログファイルの最後までが編集対象となります。

オプションフラグを指定した場合は，開始時刻か終了時刻のどちらかは必ず指定してください。

開始時刻及び終了時刻は，hhmmss〔MMDD〔YYYY〕〕の形式で指定してください。

hh：

時((00～23))

mm：

分((00～59))

ss：

秒((00～59))

MM：

月((01～12))

省略した場合，当月とみなされます。

DD：

日((01～31))

省略した場合，当日とみなされます。また，その月にあり得ない日付を指定した場合（30 日までの月に 31 日を指定するなど），その超過分の日数を繰り上げて，次の月の日付が指定されたものと判断します。

例えば，00000006312000 と指定した場合，2000 年 7 月 1 日 0 時 0 分 0 秒と判断します。

YYYY：

年((1970～9999))

省略した場合，当年とみなされます。

(4) -u UAP 名〔，UAP 名〕...

～＜文字列＞((1～30))

UAP に関する統計情報，又は SQL に関する統計情報を編集する場合，編集対象とする UAP 名（クライアント環境定義の PDCLTAPNAME で指定した UAP の識別名称）を指定します。

UAP 名は最大 16 個指定できます。省略した場合，-t オプションで指定した範囲の UAP に関する統計情報又は SQL に関する統計情報が編集されます。

(5) -x ホスト名〔，ホスト名〕...

～＜識別子＞((1～32))

編集対象のホストの名称を指定します。ホスト名は最大 32 個指定できます。省略した場合，標準ホスト名を指定して，すべてのホストが編集対象となります。なお，pdunit オペランドの-x オプションで指定したホスト名を変更した場合は，変更前と変更後の両方のホスト名が編集対象になります。統計情報は，変更前のホスト名と変更後のホスト名に分けて出力されます。

(6) -s サーバ名〔，サーバ名〕...

～＜識別子＞((1～8))

編集対象のサーバの名称を指定します。サーバ名は最大 32 個指定できます。省略した場合，すべてのサーバが編集対象となります。

(7) -i {統計入力アンロードファイル名｜HiRDB ファイルシステム領域名｜統計入力アンロードファイル格納ディレクトリ名}

統計解析の入力情報となる統計入力アンロードファイルを指定します。

統計入力アンロードファイルは，pdstedit を実行するホストから参照できるファイルでなければなりません。なお，統計入力アンロードファイルとは，アンロード統計ログファイル又はアンロードログファイルのことをいいます。

このオプションを省略した場合，標準入力からの入力とみなされます。

統計入力アンロードファイル名　～＜パス名＞

統計入力アンロードファイルの名称を指定します。

HiRDB ファイルシステム領域名　～＜パス名＞((165 文字以内))

統計入力アンロードファイルが HiRDB ファイルシステム領域にある場合，その HiRDB ファイルシステム領域の名称を指定します。

指定した HiRDB ファイルシステム領域にあるすべての統計入力アンロードファイルが解析対象となります。

統計入力アンロードファイル格納ディレクトリ名　～＜パス名＞

統計入力アンロードファイルがあるディレクトリの名称を指定します。

ディレクトリ名を指定した場合，そのディレクトリの子又は孫の位置にあるファイル（HiRDB ファイルシステム領域があるときは，その領域にあるファイル）が編集対象となります。

(a)　注意事項

1. ユティリティ実行中の OS リソース不足を回避する方法

   システム全体で大容量のファイル（システム全体で同時に 100 ギガバイト以上）を扱う場合，ファイルキャッシュが増大し，Windows のシステムリソース不足になることがあります。大容量のファイルを扱う場合，pdfmkfs コマンド（-k UTL）で作成した HiRDB ファイルシステム領域にシステムログファイルをアンロードすると，ファイルキャッシュを使用しないでアクセスするため，システムリソース不足を回避できます。ただし，ファイルキャッシュを使用しないでアクセスするため，性能は劣化します。

2. ファイル種別

   -i オプションに指定したファイル又はディレクトリ下のファイルの種別によっては，解析処理が待ち状態になったり，該当するファイルの解析をスキップしたりすることがあります。ファイル種別による解析処理の可否を次に示します。

-i オプション指定値	ファイル種別		解析処理
ファイル	RAW ファイル		処理を続行します。
	パイプファイル（FIFO）※		別プロセスがライトモードで該当するファイルをオープンするまで待ち状態となります。
	上記以外		該当するファイルの解析処理をスキップします。
ディレクトリ	ディレクトリ下のファイル	RAW ファイル	処理を続行します。
		パイプファイル（FIFO）	ワーニングメッセージが出力されます。
		上記以外	該当するファイルの解析処理をスキップします。

注※　統計ログファイルが一つだけの場合，パイプファイル（FIFO）で指定できます。

(b)　アンロードログファイルを入力する場合の注意事項

システムログファイルを pdlogunld コマンドでアンロードする場合は，-n オプションを指定してください（システムログのステータスが変更されないようにするためです）。システムログのステータスを変更してしまうと，次のようにシステムログファイルのアンロードができなくなることがあります。

- システムログファイルの自動ログアンロード機能を使用している場合，アンロード済みと判断されるため，自動アンロードがされません。
- システムログのステータスがアンロード済みになると，HiRDB は上書きしてもよいと判断するため，アンロードする前に上書きされてしまう可能性があります。

(8)　-o DAT 形式ファイル出力先ディレクトリ名

～＜パス名＞((1～128))

統計入力アンロードファイルから統計情報を収集し，DAT 形式ファイルを作成したい場合，その DAT 形式ファイルを作成するディレクトリの名称を指定します。

(9)　-w ワーク用一時ファイル作成ディレクトリ名

～＜パス名＞((1～128))

統計解析ユティリティで使用する，ワーク用一時ファイルを作成するディレクトリの名称を指定します。省略した場合，次の表に示すディレクトリが仮定されます。

表 14–2　pdstedit がワーク用一時ファイルを出力するディレクトリ

-w オプションの指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	-w オプションに指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ 上記が使用できない場合は/var/tmp

注※　コマンド実行環境の環境変数の指定

何らかの要因で表「pdstedit がワーク用一時ファイルを出力するディレクトリ」に示すディレクトリにワーク用一時ファイルが作成できない場合は，エラーとなります。

統計解析ユティリティの実行中にプロセスを強制終了したり，異常終了したりするなど，統計解析ユティリティの実行が途中で中断したとき，ワーク用一時ファイルのディレクトリ下にファイルが残ることがあります。残った場合は，ユーザがそのファイルを削除する必要があります。ワーク用一時ファイルのディレクトリ下に作成されるファイル名を次に示します。

-k オプションの指定値	作成されるファイルの名称
sys	PID.syi，PID.syo
uap	PID.uai，PID.uao
sql	PID.sqi，PID.sqo
buf	PID.bui，PID.buo
fil	PID.fii，PID.fio

-k オプションの指定値	作成されるファイルの名称
dfw	PID.dfi，PID.dfo
idx	PID.ixi，PID.ixo，PID.isi，PID.iso
sop	PID.soi，PID.soo
dop	PID.doi，PID.doo
pcd	PID.pci，PID.pco
obj	PID.obi，PID.obo
sqh	PID.qhi，PID.qho
cnc	PID.cni，PID.cno
dio	PID.dii，PID.dio

（凡例）

PID：プロセス ID

注

-k all の場合，すべてのファイルが作成されます。

(10) -d 制御文ファイル名

～＜パス名＞

統計解析ユティリティの入力情報のシステムログが格納されたファイルグループ名を記述した制御文ファイルの名称を指定します。

-d オプションが指定されていても，-i オプションが指定されていない場合，標準入力待ち状態になります。

制御文(file_group 文）のファイルの形式を次に示します。なお，制御文は，最大 34 個指定できます。

```
file_group サーバ名：ファイルグループ名〔，ファイルグループ名〕…
```

サーバ名 ～＜識別子＞((1～8))

ファイルグループに対応するサーバ名を指定します。-s オプションを指定した場合，-s オプションで指定したサーバ名と異なるサーバ名を指定したときには，解析対象外として無視されます。また，制御文ファイル中のサーバ名が重複している場合は，エラーとなります。

ファイルグループ名 ～＜識別子＞((1～8))

統計解析ユティリティを起動するホストから参照できる物理ファイルのファイルグループを指定します。

ファイルグループ名は最大 200 個指定できます。

別のホストにあるファイルグループを指定した場合には，エラーとなります。ただし，ファイルグループが共用ディスクにあるファイルの場合には，解析対象となります。

＜規則＞

1. 制御文 1 文は 1 行で記述してください。
2. 指定値の間にスペース及びタブを指定しないでください。
3. "/＊"から"＊/"まではコメントとみなします。
4. 現用ファイルグループを指定した場合，又は不整合な状態のファイルグループを指定した場合，警告メッセージを出力し，エラー検出直前までの統計情報を編集します（二重化指定の場合には，警告メッセージを二つ出力することがあります）。

   警告エラーの原因としては，次のことが考えられます。

   - ファイルの最後を検出
   - ファイルの内容が不正
5. 初期化後，未使用のファイルグループを指定した場合，統計情報の編集をしないで，警告メッセージを出力します（二重化指定の場合には，警告メッセージを二つ出力することがあります）。
6. 指定したファイルがシステムログファイルでない場合，そのファイルグループは解析しないで警告メッセージを出力します。
7. -i オプションと-d オプションを同時に指定して，-d オプションで指定した制御文ファイル中のファイルグループがアンロード済みのファイルで，そのアンロードログファイルを-i オプションに指定している場合，指定したファイルグループの情報が二重に編集されます。

(11)　-b

DAT 形式ファイルにタイトルバーを出力する場合に指定します。このオプションを指定する場合，同時に-o オプションも指定する必要があります。-o オプションを指定していない場合，このオプションの指定は無視されます。

(12)　-e 拡張フォーマット指定値〔，拡張フォーマット指定値〕

DAT 形式ファイル出力時に，出力フォーマットを変更する場合に指定します。このオプションを指定する場合，同時に-o オプションも指定する必要があります。

拡張フォーマット指定値には，「sec」及び「er1」を指定できます。

sec：

sec を指定した場合，次の統計情報の統計ログ取得時刻に秒値が出力されます。

- グローバルバッファプールに関する統計情報
- データベース操作に関する HiRDB ファイルの統計情報
- デファードライト処理に関する統計情報

このオプションを指定しない場合，統計ログ取得時刻の出力形式は「MM/DD/hh:mm」となります。このオプションを指定した場合，統計ログ取得時刻の出力形式は「MM/DD/hh:mm:ss」となります。

なお，このオプションは-y オプションを指定した場合は無効となります。

er1：

er1 は，オーバフローデータ※1及び編集できないデータ※2を，有効けた数以上出力しない場合に指定します。er1 を指定すると，オーバフローデータ及び編集できないデータの格納値が，1 個のアスタリスク（*）になります。

注※1

マイクロ秒単位の時間を出力する場合，時間が 4,200 秒を超えているときのエラーデータを示します。

注※2

pdstedit を実行する HiRDB のバージョンより，入力する統計ログを取得した HiRDB のバージョンが古い場合，出力対象項目の情報が統計ログ内にないことがあります。統計ログ内にない項目の情報を，「編集できないデータ」といいます。

(13)　-y

DAT 形式ファイル出力時，日付時刻の出力形式を「YYYY/MM/DD hh:mm:ss」とする場合に指定します。このオプションを指定していても，-o オプションが指定されていない場合は無視します。このオプションを指定した場合は，-e sec の指定は無効となります。

14.2.3　各オプション指定時の留意事項

1. 統計情報は，-k オプションとそのほかのオプションの指定によって情報を絞り込めます。-k オプションとそのほかのオプションとの関係を次の表に示します。

表 14–3　-k オプションとそのほかのオプションとの関係

-k オプション（内容）	そのほかのオプション			
	-t 開始時刻，終了時刻	-u UAP 名	-x ホスト名	-s サーバ名
sys（システムの稼働に関する統計情報）	○	×	○	×
svr（サーバごとのシステムの稼働に関する統計情報）	○	×	○	○
uap（UAP に関する統計情報）	○	○	○	×
sql（SQL に関する統計情報）	○	○	○	×
buf（グローバルバッファプールに関する統計情報）	○	×	○	○
fil（データベース操作に関する HiRDB ファイルの統計情報）	○	×	○	○
dfw（デファードライト処理に関する統計情報）	○	×	○	○
idx（インデクスに関する統計情報）	○	×	×	○
sop（SQL 静的最適化に関する統計情報）	○	○	○	×
dop（SQL 動的最適化に関する統計情報）	○	○	○	×

-k オプション（内容）	そのほかのオプション			
	-t 開始時刻，終了時刻	-u UAP 名	-x ホスト名	-s サーバ名
pcd（SQL オブジェクト実行に関する統計情報）	○	○	○	×
obj（SQL オブジェクト転送に関する統計情報）	○	○	○	×
sqh（SQL 文の履歴に関する統計情報）	○	○	○	○
cnc（CONNECT/DISCONNECT に関する統計情報）	○	○	○	○
dio（データベースの入出力に関する統計情報）	○	×	○	○
all（上記のすべての統計情報）	○	○	○	○

（凡例）

○：指定できます。指定した場合，その情報は絞り込めます。

×：指定しても無視されます。その情報は絞り込めません。

2. -k idx の場合，-o オプションを指定しても無視されます。

3. -k idx，-k all の場合，-d オプションを指定すると，その情報が絞り込めます。

4. -k オプションに sop，dop，又は pcd を指定した場合，同時に-o オプションも指定してください。

5. -i，又は-d オプション指定時に入力できるファイルと，解析できる統計情報を次に示します。

オプション	入力できるファイル	解析できる統計情報
-i	アンロードログファイル	• インデクスに関する統計情報のスプリットに関する情報 • CONNECT/DISCONNECT に関する統計情報
	アンロード統計ログファイル	上記以外のすべての統計情報
-d	システムログファイル	• インデクスに関する統計情報のスプリットに関する情報 • CONNECT/DISCONNECT に関する統計情報

6. -k uap の場合，UAP の実行環境によって，統計情報の取得単位が変わります。UAP 実行時のクライアント環境定義（PDSTJTRNOUT）と，統計情報の取得単位の関係を次に示します。

PDSTJTRNOUT の内容	UAP の種類	
	OLTP 環境の場合	OLTP 環境以外の場合
YES	トランザクション単位	トランザクション単位
NO	UAP 実行単位	UAP 実行単位
省略	トランザクション単位	トランザクション単位

7. k svr の場合，次のフィールドはサーバ単位の情報を出力しません（若しくはフィールド値に 0 を表示します）。そのため，-k sys を指定して編集したユニット単位の情報で確認してください。

表 14–4　サーバ単位の情報を出力しないフィールド（標準出力）

統計情報の種別	フィールド名（項目）	
sys（システムの稼働に関する統計情報）	<PROCESS>（プロセス情報）	# OF USER SERVER ABORT（サーバ異常終了回数）
		# OF SYSTEM SERVER ABORT（HiRDB が使用する内部的なサーバが異常終了した回数）
		# OF PROCESS（サーバ，及び HiRDB が使用する内部的なサーバのプロセス個数）
	<RPC>（RPC 情報）	# OF REGISTERED PORTS（HiRDB 予約ポート使用数）
		# OF ASSIGNED PORTS（HiRDB 予約ポートオーバー時の OS 自動割り当てポート使用数）
	<SHARED MEMORY>（共用メモリ情報）	STATIC GET SIZE（システムが使用する情報（ユーザ未使用））
		STATIC POOL SIZE（システムが使用する情報（ユーザ未使用））
		DYNAMIC GET SIZE（システムが使用する情報（ユーザ未使用））
		DYNAMIC POOL SIZE（システムが使用する情報（ユーザ未使用））
		SIZE EXCEPT GLOBAL BUFFER（サーバ用，及び HiRDB が使用する内部的なサーバ用に確保した共用メモリのサイズ）
		STATIC SIZE（静的共用メモリ確保サイズ）
		DYNAMIC SIZE（動的共用メモリ確保サイズ）
		SIZE FOR GLOBAL BUFFER（グローバルバッファプール用共用メモリ確保サイズ）

表 14–5　サーバ単位の情報を出力しないフィールド（DAT 形式ファイル）

統計情報の種別	フィールド名（項目）	
sys（システムの稼働に関する統計情報）	PROCESS（プロセス情報）	SERVER ABORT（サーバの異常終了回数）
		SYSTEM SERVER ABORT（HiRDB が使用する内部的なサーバの異常終了回数）
		PROCESS COUNT（プロセス個数）
	RPC（RPC 情報）	REGISTERED PORTS（HiRDB 予約ポートの使用数）
		ASSIGNED PORTS（HiRDB 予約ポートオーバー時の OS 自動割り当てポート使用数）
	SHARED MEMORY（共用メモリ情報）	STATIC GET SIZE（システム固有情報）
		STATIC POOL SIZE（システム固有情報）
		DYNAMIC GET SIZE（システム固有情報）
		DYNAMIC POOL SIZE（システム固有情報）

統計情報の種別	フィールド名（項目）	
		SEGMENT SIZE（サーバ用，及び HiRDB が使用する内部的なサーバ用に確保した共用メモリのサイズ）
		STATIC SIZE（静的共用メモリ確保サイズ）
		DYNAMIC SIZE（動的共用メモリ確保サイズ）
		SIZE FOR BUFFER（グローバルバッファ用共用メモリ確保サイズ）

14.3　統計情報の詳細

14.3.1　統計情報の出力形式

(1)　編集項目ごとのログレコードサマリ情報

各統計情報出力後，編集項目ごとにログレコードのサマリ情報を出力します。

ただし，SQL 静的最適化に関する統計情報，SQL 動的最適化に関する統計情報，SQL オブジェクト実行に関する統計情報，SQL オブジェクト転送に関する統計情報，SQL 文の履歴に関する統計情報，CONNECT/DISCONNECT に関する統計情報，及びデータベースの入出力に関する統計情報の場合，編集項目ごとのログサマリ情報は出力されません。

```
          :
     統計情報
          :
FILE KIND LOG KIND FIRST               LAST                       NUM
XXX       XXX      XXXX/XX/XX XX:XX:XX XXXX/XX/XX XX:XX:XX XXXXXXXXXX
 :         :                :                  :             :
```

[説明]

FILE KIND：

解析対象のログファイル種別を示します。

アンロード統計ログファイルの場合は"STJ"，システムログファイル（アンロードログファイル又はシステムログのファイルグループ）の場合は"FJ"を出力します。

LOG KIND：

解析対象のログレコードの種別を示します。

sys：システムの稼働に関する統計情報

uap：UAP に関する統計情報

sql：SQL に関する統計情報

buf：グローバルバッファプールに関する統計情報

fil：データベース操作に関する HiRDB ファイルの統計情報

dfw：デファードライト処理に関する統計情報

idx：インデクスに関する統計情報

FIRST：

解析対象のログレコードの最も古いログ取得時刻を示します。

解析対象のログファイル内に解析対象のログレコードが 1 件もない場合は，"****/**/** **:**:**"を出力します。

LAST：

解析対象のログレコードの最も新しいログ取得時刻を示します。

解析対象のログファイル内に解析対象のログレコードが 1 件もない場合は，"****/**/** **:**:**"を出力します。

NUM：

解析対象のログレコードの件数を示します。

(2) 入力ログファイルのサマリ情報

すべての統計情報出力後，入力したすべてのログファイルのサマリ情報を，すべてのログレコード種別ごとに出力します。

システムログのファイルグループを入力とした場合（-d オプション指定時），入力ログファイルのサマリ情報は出力されません。また，アンロードログファイルを入力とした場合，インデクスに関する統計情報の編集項目指定があるときだけ，入力ログファイルのサマリ情報が出力されます。それ以外の統計情報の編集項目を指定した場合は，レコード件数 0 が表示されます。

```
                    :
   編集項目ごとのログレコードサマリ情報
                    :
NO FILE KIND:LOG FILE NAME
 LOG KIND  FIRST                LAST                          NUM
XX XXX      :XX...X
 XXX       XXXX/XX/XX XX:XX:XX  XXXX/XX/XX XX:XX:XX  XXXXXXXXXX
  :                  :                     :              :
```

[説明]

NO：

ログファイルの通番を示します。

FILE KIND：

解析対象のログファイル種別を示します。

STJ：

32 ビットモードの HiRDB で取得したアンロード統計ログファイル

STJ64：

64 ビットモードの HiRDB で取得したアンロード統計ログファイル

FJ：

アンロードログファイル（通常のファイル）

FJIOS：

アンロードログファイル（HiRDB ファイル）

LOG FILE NAME：

入力したアンロード統計ログファイル又はアンロードログファイル（インデクスに関する統計情報の場合）の名称を示します。

名称は，最大 256 バイトの絶対パス名で表示されます。256 バイトを超えた場合，最後尾から 256 バイトまでを表示します。

LOG KIND：

解析対象のログレコードの種別を示します。

sys：システムの稼働に関する統計情報

uap：UAP に関する統計情報

sql：SQL に関する統計情報

buf：グローバルバッファプールに関する統計情報

fil：データベース操作に関する HiRDB ファイルの統計情報

dfw：デファードライト処理に関する統計情報

idx：インデクスに関する統計情報

sop：SQL 静的最適化に関する統計情報

dop：SQL 動的最適化に関する統計情報

pcd：SQL オブジェクト実行に関する統計情報

obj：SQL オブジェクト転送に関する統計情報

sqh：SQL 文の履歴に関する統計情報

cnc：CONNECT/DISCONNECT に関する統計情報

dio：データベースの入出力に関する統計情報

FIRST：

ログファイルの最も古いログ取得時刻を示します。

解析対象のログファイル内にログレコードが 1 件もない場合は，"****/**/** **:**:**"を出力します。

LAST：

ログファイルの最も新しいログ取得時刻を示します。

解析対象のログファイル内にログレコードが 1 件もない場合は，"****/**/** **:**:**"を出力します。

NUM：

ログレコード種別ごとのレコード件数を示します。

(3) 編集データがないときの出力内容

表示内容	説明	対処方法
”***** NO DATA NOOPT *****”	解析対象の統計入力アンロードファイル内に，該当する編集項目レコードがありません。	オプションの指定誤り，又はログレコード取得時の出力項目の誤りが考えられます。次のどちらかの方法で対処してください。

表示内容	説明	対処方法
”***** NO DATA HOST *****”	-x オプションで指定したホストから出力された，ログレコードがありません。	• -x，-s，-t，-u，及び-f オプションの指定値を見直し，誤っていた場合は修正して，再度実行してください。 • 統計入力アンロードファイルのサマリ情報から編集対象のログレコードが出力されているか，又は統計入力アンロードファイルへの出力時間が正しいか確認してください。また，誤った統計入力アンロードファイルを入力していないか，又は解析対象のログレコードが出力されているか確認してください。
”***** NO DATA SERVER *****”	-s オプションで指定したサーバから出力された，ログレコードがありません。	
”***** NO DATA TIME *****”	-t オプションで指定した編集時間範囲内に出力された，ログレコードがありません。	
”***** NO DATA UAP *****”	-u オプションで指定した UAP から出力された，ログレコードがありません。	
”***** NO DATA 1 REC *****”	解析対象の統計入力アンロードファイル内に，差分情報を取得するためのログレコードがなかったため，差分情報が編集できません。編集項目ごとのログサマリ情報，又は統計入力アンロードファイルのサマリ情報にログレコードが 2 件以上あっても，1 回のシンクポイントで出力されたレコード（サマリ情報の最も古いログ取得時刻（FIRST）と最も新しいログ取得時刻（LAST）が同じ時間帯）ならば，差分情報取得のためのログレコードがないことを示します。ただし，グローバルバッファプールに関する統計情報，及びデータベース操作に関する HiRDB ファイルの統計情報の場合だけ該当します。	2 レコード以上になるように，統計入力アンロードファイルを設定してください。
”***** NO DATA *****”	該当する編集時間範囲内に，解析対象のログレコードがありません。	解析したい時間間隔で，NO DATA が多く出力される場合，編集時間範囲を長くしてください。
	該当する編集項目に対して，編集できるログレコードがありません。	編集できるログレコードがあるか，統計入力アンロードファイルサマリ情報を確認してください。

注

SQL 静的最適化情報，SQL 動的最適化情報，及び SQL オブジェクト実行情報の編集項目の解析対象レコードがなかった場合は，インフォメーションメッセージが出力されます。

(4) 統計情報の数値の形式

統計情報の出力情報に数値がある場合，表示形式は次のようになります。

数値の範囲	表示形式
0～999	0～999
1,000～9,994	1.00k～9.99k※
9,995～99,949	10.0k～99.9k※
99,950～999,499	100k～999k※
999,500～9,994,999	1.00M～9.99M※
9,995,000～99,949,999	10.0M～99.9M※
99,950,000～999,499,999	100M～999M※
999,500,000～9,994,999,999	1.00G～9.99G※
9,995,000,000～99,949,999,999	10.0G～99.9G※
99,950,000,000～999,499,999,999	100G～999G※
999,500,000,000～4,398,046,511,103	1.00T～4.40T※

（凡例）

k：キロ

M：メガ

G：ギガ

T：テラ

注※

表示けた数の次の位が，四捨五入されます。

(5)　出力される統計情報に関する注意事項

出力される統計情報についての注意事項を次に示します。

1. 数字は右詰め，そのほかの文字は左詰めで出力されます。
2. 出力指定日時は，編集範囲時刻を指定した場合に出力されます。指定しなかった場合には，’＊’が出力されます。
3. 編集時間の範囲は，編集開始時刻を指定した場合に開始時刻を含む時間帯から出力されます。指定しなかった場合には，ファイル中の最も早い時刻を持つ，レコードを含む時間帯から出力されます。
4. 編集時間の範囲は，編集終了時刻を指定した場合に，ファイル内の最も早い時刻を持つレコードを含む時間帯から，指定終了時刻を含む時間帯まで出力されます。
5. 集計中にオーバフローした場合，又は累計を出力しない項目には，’＊＊＊’，’＊＊＊＊’，又は空白が出力されます。
6. 平均値がオーバフローしたデータには，’＊’が出力されます。ただし，このときの最大値及び最小値は保証されます。

7. 取得編集時間帯にデータがない場合は，その時間帯に’＊＊＊＊＊　NO DATA aa...a　＊＊＊＊＊’が出力されます。

8. 項目の単位がバイトの場合，1.00K=2^{10}バイト，1.00M=2^{20}バイト，1.00G=2^{30}バイト，1.00T=2^{40}バイトが基準になります。

14.3.2　システムの稼働に関する統計情報

システムの稼働に関する統計情報を次に示します。

```
pdstedit VV-RR(Object Option) ***** SYSTEM INFORMATION *****
INPUT          :c:¥tmp¥stjdata [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
--------------------------------------------------------------------------------
HOST = test [3]
--------------------------------------------------------------------------------
EDIT TIME 2012/02/03 09:00:00 - 2012/02/03 10:00:00 [4]
SERVER : ******** [5]                              [6]    [7]    [8]    [9]
                                                  FREQ    MAX    MIN    AVG
                                                 -----  -----  -----  -----
<SCHEDULE>[10]     QUEUE LENGTH[11]                77      1      1      1
                   MESSAGE LENGTH[12]              77    660    432    615
<PROCESS>[13]      # OF USER SERVER ABORT[14]       0
                   # OF SYSTEM SERVER ABORT[15]     0
                   # OF PROCESS[16]                        48     44     44
                   # OF PROCESS ON SERVICE[17]             8      0      0
                   # OF REQ PROCESS OVER MAX[18]    0
<TRANSACTION>[19]  # OF COMMIT[20]                 242
                   # OF ROLLBACK[21]               12
<NAME>[22]         # OF CACHE HIT[23]               0
                   # OF LOCAL HIT[24]              743
                   # OF LOOK-UP[25]                742
<RPC>[26]          # OF TIME OUT[27]                0
                   # OF ERROR[28]                   0
                   RESPONSE ON OWN UNIT[29]       192  63.6k    192  4.97k
                   RESPONSE TO OTHER UNIT[30]       0      0      0      0
                   EXEC TIME ON OWN UNIT[31]     1.48k  91.2k      0  1.41k
                   EXEC TIME FROM OTHER UNIT[32]    0      0      0      0
                   # OF SEND TO OWN PRCS[33]       0
                   # OF SEND TO OTHER PRCS[34]    200
                   # OF SEND TO OTHER UNIT[35]      0
                   # OF RECV FROM OWN PRCS[36]      0
                   # OF RECV FROM OTHER PRCS[37]   12
                   # OF RECV FROM OTHER UNIT[38]    0
                   # OF REGISTERED PORTS[39]        0       0      0      0
                   # OF ASSIGNED PORTS[40]          0       0      0      0
<LOCK>[41]         WAIT TIME[42]                   30      97      1     27
                   QUEUE LENGTH[43]                30       4      1      2
                   # OF DEADLOCK[44]                0
                   % OF USE LOCK TABLE[45]          5       0      0      0
                   # OF REQUEST WAIT QUEUE[124]  9.56k       1      0      0
                   # OF REQUEST[128]             9.60k
                   # OF REQUEST WAIT[129]          27
                   REQUEST WAIT TIME[130]                  363      2     58
```

```
<SHARED MEMORY>[46]STATIC GET SIZE[47]               0       0     0      0
                   STATIC POOL SIZE[48]              0       0     0      0
                   DYNAMIC GET SIZE[49]              0       0     0      0
                   DYNAMIC POOL SIZE[50]             0       0     0      0
                   SIZE EXCEPT GLOBAL BUFFER[51] 63.4M
                   STATIC SIZE[52]               9.08M
                   DYNAMIC SIZE[53]              3.90M
                   SIZE FOR GLOBAL BUFFER[54]    1.77M
<SYNC POINT>[55]   SYNC POINT GET INTERVAL[56]      10    169k 21.0k  75.3k
                   SYNC POINT GET TIME[57]          10   3.00k 1.00k  1.40k
<LOG>[58]          # OF BUFFER FULL[59]              0
                   # OF WAIT THREAD[60]              0
                   OUTPUT BLOCK LENGTH[61]         656    121k   340  3.17k
                   NOT BUS LENGTH[62]              656    121k   340  3.04k
                   # OF BUFFER FOR WAIT I/O[63]    139     100   100    100
                   # OF WRITE TO FILE[64]          671
                   # OF WRITE ERROR[65]              0
                   LOG FILE SWAP TIME[66]           5     159    57    116
                   LOG INPUT DATA LENGTH[67]       10    392k 20.0k   258k
                   # OF READ FROM FILE[68]          15
                   # OF READ ERROR[69]               0
                   FILE WRITE TIME[125]           671    115k 2.46k  19.6k
                   FILE READ TIME[126]             15    144k     9  16.5k
                   # OF BUFFERING[127]           10.4k
                   # OF BUF FLUSH WAIT QUEUE[131]   71       1     1      0
                   # OF BUF FLUSH[132]              71
                   # OF BUF FLUSH WAIT[133]         71
                   BUF FLUSH WAIT TIME[134]              169k 22.8k  31.0k
<DICTIONARY>[70]   # OF TBL-DEF GET REQ[71]        295
                   # OF TBL-CACHE HIT[72]          272
                   # OF CACHED TBL-DEF[73]          27      12     1      6
                   USED TBL-DEF SIZE[74]            23   26.0k 5.60k  9.95k
                   TBL-CACHE SIZE[75]               27    118k 26.0k  79.1k
                   # OF ACCESS PRIV CHECK[76]        0
                   # OF CACHE HIT (AP CHECK)[77]     0
                   # OF CON/DBA DEF GET REQ[78]     45
                   # OF CON/DBA CACHE HIT[79]       43
                   # OF CON/DBA CACHED USER[80]      6       1     1      1
                   DICT SERV TRANS DATA SIZE[81]     0       0     0      0
                   # OF TRANS[82]                    0
                   # OF VIEW DEF GET REQ[83]         0
                   # OF VIEW CACHE HIT[84]           0
                   # OF VIEW CACHED DEF[85]          0
                   USED VIEW SIZE[86]                0       0     0      0
                   VIEW CACHE SIZE[87]               0       0     0      0
                   CACHE-MISS VIEW SIZE[88]          0       0     0      0
                   # OF TYPE-DEF GET REQ[89]         0
                   # OF TYPE-DEF CACHE HIT[90]       0
                   # OF CACHED TYPE-DEF[91]          0       0     0      0
                   TYPE-DEF CACHE SIZE[92]           0       0     0      0
                   TYPE-DEF CACHE TOTAL SIZE[93]     0       0     0      0
                   TYPE-DEF CACHE ALLOC SIZE[94]     4       0     0      0
                   # OF RTN-DEF GET REQ[95]          0
                   # OF RTN-DEF CACHE HIT[96]        0
                   # OF CACHED RTN-DEF[97]           0       0     0      0
                   RTN-DEF CACHE SIZE[98]            0       0     0      0
                   RTN-DEF CACHE TOTAL SIZE[99]      0       0     0      0
                   RTN-DEF CACHE ALLOC SIZE[100]     4    100k   100k   100k
```

```
                  # OF PLG-RTN GET REQ[101]         0
                  # OF PLG-RTN CACHE HIT[102]       0
                  # OF REGISTRY GET REQ[103]        0
                  # OF REGISTRY CACHE HIT[104]      0
                  # OF CACHED REGISTRY-DEF[105]     0       0     0       0
                  REGISTRY CACHE SIZE[106]          0       0     0       0
                  REGISTRY CACHE TOTAL SIZE[107]    0       0     0       0
                  DIRECTORY USER CHECK TIME[108]    0       0     0       0
                  GROUP CHECK TIME[109]             0       0     0       0
<FES-BES-DIC(SDS)  # OF SQLOBJ INFO GET[111]       240
 INFORMATION>     # OF CACHE HIT (SQLOBJ)[112]     41
 [110]            # OF CACHED SQLOBJ[113]         203      65     1      27
                  CACHED SQLOBJ TOTAL SIZE[114]   203    277k 5.00k    137k
                  # OF SWAP OUT SQLOBJ[115]        36
                  REQUEST SQLOBJ SIZE[116]        163   9.44k 2.02k  4.62k
                  # OF STRT INFO GET[117]          0
                  # OF CACHED HIT (STRT)[118]      0
                  # OF CACHED STRT[119]            0       0     0       0
                  CACHED STRT TOTAL SIZE[120]     0       0     0       0
                  # OF SWAP OUT STRT[121]          0
                  REQUEST STRT SIZE[122]          0       0     0       0
                  # OF STRT RECOMPILE[123]        0
--------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                  LAST                         NUM
STJ         sys        2012/02/03 09:30:49   2012/02/03 09:37:00           5
--------------------------------------------------------------------------------

NO FILE KIND:LOG FILE NAME
  LOG KIND             FIRST                  LAST                         NUM
 1 STJ      :c:¥tmp¥stjdata¥pdstj01
  sys                  2012/02/03 09:30:49   2012/02/03 09:31:50           4
  uap                  2012/02/03 09:29:26   2012/02/03 09:31:04          26
  sql                  2012/02/03 09:29:25   2012/02/03 09:31:04         440
  sop                  2012/02/03 09:29:27   2012/02/03 09:31:04          98
  dop                  2012/02/03 09:29:27   2012/02/03 09:31:04          92
  pcd                  ****/**/** **:**:**   ****/**/** **:**:**           0
  obj                  ****/**/** **:**:**   ****/**/** **:**:**           0
  sqh                  ****/**/** **:**:**   ****/**/** **:**:**           0
  buf                  2012/02/03 09:30:00   2012/02/03 09:32:24          21
  fil                  2012/02/03 09:30:00   2012/02/03 09:32:24         224
  dfw                  2012/02/03 09:29:30   2012/02/03 09:32:24          71
  idx                  2012/02/03 09:30:00   2012/02/03 09:32:24           7
  fsv                  ****/**/** **:**:**   ****/**/** **:**:**           0
  hba                  ****/**/** **:**:**   ****/**/** **:**:**           0
  dio                  2012/02/03 09:30:00   2012/02/03 09:32:24          77
 2 STJ      :c:¥tmp¥stjdata¥pdstj02
  sys                  2012/02/03 09:34:58   2012/02/03 09:37:00           6
  uap                  2012/02/03 09:34:00   2012/02/03 09:36:17          19
  sql                  2012/02/03 09:33:59   2012/02/03 09:36:17         456
  sop                  2012/02/03 09:34:02   2012/02/03 09:36:16          65
  dop                  2012/02/03 09:34:02   2012/02/03 09:36:16          96
  pcd                  ****/**/** **:**:**   ****/**/** **:**:**           0
  obj                  ****/**/** **:**:**   ****/**/** **:**:**           0
  sqh                  ****/**/** **:**:**   ****/**/** **:**:**           0
  buf                  2012/02/03 09:34:10   2012/02/03 09:36:19          27
  fil                  2012/02/03 09:34:10   2012/02/03 09:36:19         288
  dfw                  2012/02/03 09:34:04   2012/02/03 09:36:19          41
  idx                  2012/02/03 09:34:10   2012/02/03 09:36:19           9
```

```
fsv                 ****/**/** **:**:**   ****/**/** **:**:**          0
hba                 ****/**/** **:**:**   ****/**/** **:**:**          0
dio                 2012/02/03 09:30:00   2012/02/03 09:32:24         77
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時−出力終了日時）
3. システムの稼働に関する統計情報の出力要求元ホスト名
4. 編集時間の範囲（取得開始時刻−取得終了時刻）
5. サーバ名。-k オプションに sys を指定した場合は，"＊＊＊＊＊＊＊＊＊"を出力します。
6. 各項目の発生回数
7. 各項目の最大値
8. 各項目の最小値
9. 各項目の平均値
10. スケジュール情報
11. スケジュール待ち行列数

    サーバに対する要求がスケジュールキューに登録された件数と，スケジュールキューへの登録が同時に発生したときの最大，最小，平均値です。
12. スケジュールメッセージ長（単位：バイト）

    サーバに対する処理要求を格納したメッセージの個数と，メッセージ長の最大，最小，平均値です。
13. プロセス情報
14. サーバ異常終了回数

    シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，及びバックエンドサーバのプロセスが異常終了した回数です。
15. HiRDB が使用する内部的なサーバが異常終了した回数

    HiRDB が使用する内部的なサーバのプロセスが異常終了した回数です。
16. サーバ，及び HiRDB が使用する内部的なサーバのプロセス個数

    シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，バックエンドサーバ，及び HiRDB が使用する内部的なサーバのプロセスの個数です。
17. サービス実行中のサーバプロセス数

    サービス実行中のサーバのプロセス数です。
18. 最大起動プロセス数を超えたサービス要求数

    最大起動プロセス数を超えたサービス要求の数です。 最大起動プロセス数は，シングルサーバ及びフロントエンドサーバの場合は pd_max_users オペランド指定値，バックエンドサーバの場合は pd_max_bes_process オペランド指定値，ディクショナリサーバの場合は pd_max_dic_process オペ

ランド指定値となります。サービス要求数は，サービスを受けられずにリトライした要求も含めた値のため，実際に超えた数より大きくなることがあります。

19. トランザクション情報

20. コミット回数

    シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，及びバックエンドサーバで処理したトランザクションのうち，コミットしたトランザクションの件数です。

21. ロールバック回数

    シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，及びバックエンドサーバで処理したトランザクションのうち，ロールバックしたトランザクションの件数です。

22. ネームサーバ情報

23. システムが使用する情報（ユーザ未使用）

24. システムが使用する情報（ユーザ未使用）

25. システムが使用する情報（ユーザ未使用）

26. RPC 情報

27. システムが使用する情報（ユーザ未使用）

28. システムが使用する情報（ユーザ未使用）

29. 自ユニットのサーバへのサービスレスポンス時間（単位：100 マイクロ秒）※2

    自ユニットのサーバへのサービスレスポンス回数と，レスポンス時間の最大，最小，平均値です。

30. 他ユニットのサーバへのサービスレスポンス時間（単位：100 マイクロ秒）※2

    他ユニットのサーバへのサービスのレスポンス回数と，レスポンス時間の最大，最小，平均値です。

31. 自ユニットのサーバからの 1 サービス当たりの実行時間（単位：100 マイクロ秒）※2

    自ユニットのサーバからの 1 サービス当たりの実行回数と，実行時間の最大，最小，平均値です。

32. 他ユニットのサーバからの 1 サービス当たりの実行時間（単位：100 マイクロ秒）※2

    他ユニットのサーバからの 1 サービス当たりの実行回数と，実行時間の最大，最小，平均値です。

33. 自プロセスへの SEND 回数

    自プロセスへ SEND（メッセージ送信）をした回数です。

34. 自ユニットの他プロセスへの SEND 回数

    自ユニットの他プロセスへ SEND（メッセージ送信）をした回数です。

35. 他ユニットへの SEND 回数

    他ユニットへ SEND（メッセージ送信）をした回数です。

36. 自プロセスからの RECEIVE 回数

    自プロセスから RECEIVE（メッセージ受信）をした回数です。

37. 自ユニットの他プロセスからの RECEIVE 回数

    自ユニットの他プロセスから RECEIVE（メッセージ受信）をした回数です。

38. 他ユニットからの RECEIVE 回数

    他ユニットから RECEIVE（メッセージ受信）をした回数です。

39. HiRDB 予約ポート使用数

    HiRDB が予約したポート番号の範囲のうち，実際に使用しているポート番号の数と，ポート番号の最大，最小，平均値です。

40. HiRDB 予約ポートオーバー時の OS 自動割り当てポート使用数

    HiRDB が予約した範囲のポート番号がすべて使用済みの場合に，OS が自動的に割り当てたポート番号の数と，ポート番号の最大，最小，平均値です。

41. 排他制御情報

42. 排他待ち時間（単位：ミリ秒）※2

    サーバ内で発生した排他取得要求に対して，ほかのユーザが排他を取得しているために排他取得待ちとなった件数と，待ち時間の最大，最小，平均値です。

43. 排他待ち数（ユーザ数）

    サーバ内で発生した排他取得要求に対して，ほかのユーザが排他を取得しているために排他取得待ちとなった件数と，ユーザ数の最大，最小，平均値です。

44. デッドロック件数

    サーバ内で発生した排他取得要求に対して，デッドロックとなった件数です。

45. 排他制御用プールパーティションの使用率（%）

    排他制御用プールパーティションの使用率を取得した回数と，その事象が発生したときの排他制御用プールパーティションの使用率の最大，最小，平均値です。排他制御用プールパーティションの使用率は，次に示すタイミングで取得されます。

    - システムの稼働に関する統計情報を統計ログファイルに出力したときから，一定時間経過したとき（条件によって経過時間が異なります）

      ・pdstbegin オペランドを指定している，又は pdstbegin コマンドで統計情報の取得を開始した場合は，pdstbegin オペランド又は pdstbegin コマンドの-m オプションで指定した時間（単位：分）が経過すると，排他制御用プールパーティションの使用率が取得されます。

      ・pd_statistics オペランドに Y を指定し，pdstbegin オペランドを指定していない場合は，10 分経過すると排他制御用プールパーティションの使用率が取得されます。

    - 前回取得した排他制御用プールパーティションの使用率から，5%使用率が増加したとき
    - HiRDB を正常終了したとき
    - pdstjsync コマンドを実行したとき

    なお，排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition 又は pd_fes_lck_pool_partition オペランドに 2 以上を指定した場合）は，排他制御用プールパーティションごとに計算した使用率を出力します。

46. 共用メモリ情報

47. システムが使用する情報（ユーザ未使用）

48. システムが使用する情報（ユーザ未使用）

49. システムが使用する情報（ユーザ未使用）

50. システムが使用する情報（ユーザ未使用）

51. サーバ用，及び HiRDB が使用する内部的なサーバ用に確保した共用メモリのサイズ（単位：バイト）

    ユニットで確保したサーバ用，及び HiRDB が使用する内部的なサーバ用の共用メモリサイズです。

52. 静的共用メモリ確保サイズ（単位：バイト）

    ユニットで確保したサーバ用，及び HiRDB が使用する内部的なサーバ用の共用メモリサイズのうち，静的共用メモリとして確保したサイズです。

53. 動的共用メモリ確保サイズ（単位：バイト）

    ユニットで確保したサーバ用，及び HiRDB が使用する内部的なサーバ用の共用メモリサイズのうち，動的共用メモリとして確保したサイズです。

54. グローバルバッファプール用共用メモリ確保サイズ（単位：バイト）

    サーバ内で確保した，グローバルバッファプールのサイズ（グローバルバッファプールが複数の共用メモリセグメントを使用している場合は，その合計値）です。

55. シンクポイント情報

56. シンクポイントダンプ取得間隔時間（単位：ミリ秒）※2

    シンクポイントダンプの取得件数と，シンクポイントダンプを取得してから次のシンクポイントダンプを取得するまでの経過時間の最大，最小，平均値です。

57. シンクポイントダンプ取得時間（単位：ミリ秒）※2

    シンクポイントダンプの取得件数と，処理に要した時間の最大，最小，平均値です。

    この時間が長い場合，長いトランザクションが並行して実行されているためにシンクポイントダンプが有効になるまで時間が掛かったという要因が考えられます。

58. ログ情報

59. バッファ満杯回数

    HiRDB システム又はサーバで，ログ出力バッファが満杯となった回数です。

60. カレントバッファなしによる待ち回数

    HiRDB システム又はサーバで，ログ入出力バッファへの出力時に，該当するログ入出力バッファがシステムログファイルへの出力中のため，待ち状態となった回数です。

61. 出力したブロック長（単位：バイト）

    HiRDB システム又はサーバで，ログ入出力バッファからシステムログファイルに出力した回数と，ブロック長の最大，最小，平均値です。

    バス出力の場合は，その合計の長さとなります。HiRDB のシステムログ出力制御では，現在ファイルに出力中のログバッファが解放されるまでの間，ほかのログバッファに対してファイル出力要求が発生した場合でも，ファイル出力要求が発生したバッファ中に後続のシステムログを格納します。そして，現在出力中のログバッファが解放された後で一緒に出力します。これをバス出力といいます。

62. 非バス部分のデータ長（単位：バイト）

HiRDB システム又はサーバで，ログ入出力バッファからシステムログファイルにバス出力以外で出力した回数と，ログブロック長の最大，最小，平均値です。

63. 出力待ちバッファ面数

HiRDB システム又はサーバで，システムログファイルへの出力完了時に，システムログファイルへの出力完了待ちをしているログ出力バッファがあった件数と，バッファの面数の最大×100，最小×100，平均値×100 です。

表示されている最大，最小，平均値をそれぞれ 100 で除算した値が，バッファ面数の最大，最小，平均値（単位：面数）となります。

64. ファイルへの書き込み回数

HiRDB システム又はサーバで，システムログファイルへ書き込みをした回数です。

システムログのファイル出力以外にも，システムログファイルの状態を変更するときの書き込み回数が加算されます。システムログファイルを二重化している場合には，系ごとの書き込み回数が加算されます。また，システムログブロックの書き込み回数だけでなく，システムログファイルのスワップなどでファイルの状態が変化する場合に発生する，ファイルの書き込み回数も含まれます。

65. 書き込みエラー回数

HiRDB システム又はサーバで，システムログファイルへの書き込み時にエラーが発生した回数です。

66. ログファイルスワップ時間（単位：ミリ秒）※2

HiRDB システム又はサーバで，システムログファイルがスワップした回数と，スワップするまでに要した経過時間の最大，最小，平均値です。

67. ログ入力データ長（単位：バイト）

HiRDB システム又はサーバで，ロールバックのときにシステムログファイルから入力したログの件数と，ログデータ長の最大，最小，平均値です。

68. ファイルからの読み出し回数

HiRDB システム又はサーバで，ロールバックのときにシステムログファイルからログを入力した回数です。

69. 読み出しエラー回数

HiRDB システム又はサーバで，システムログファイルからのログの読み込み時にエラーが発生した回数です。

70. ディクショナリ情報

71. 表定義情報取得要求回数

シングルサーバ及びフロントエンドサーバで，表操作の処理をした回数です。自動採番機能使用時は順序数生成子も含まれます。

72. 表定義情報用バッファヒット回数

シングルサーバ及びフロントエンドサーバで，取得要求した表定義情報が表定義情報用バッファ内で見付かった回数です。自動採番機能使用時は順序数生成子も含まれます。

表定義情報用バッファヒット回数が表定義情報取得要求回数より少ない場合は，表定義情報用バッファ長を見直す必要があります。

73. 表定義情報用バッファ中の表定義情報数

シングルサーバ及びフロントエンドサーバで，表定義情報用バッファ内にバッファリングされた表定義情報の数と，表定義情報数の最大，最小，平均値です。自動採番機能使用時は順序数生成子も含まれます。

74. 表定義情報用バッファに取得した 1 表定義情報当たりの表定義情報用バッファ使用領域長（単位：バイト）

シングルサーバ及びフロントエンドサーバで，表定義情報用バッファに取得された表定義情報の件数と，1 表定義情報当たりの表定義情報用バッファ使用領域長の最大，最小，平均値です。自動採番機能使用時は順序数生成子も含まれます。

75. 表定義情報用バッファ使用領域長（単位：バイト）

シングルサーバ及びフロントエンドサーバで，表定義情報用バッファ上に取得された表定義情報の件数と，表定義情報全体の長さの最大，最小，平均値です。自動採番機能使用時は順序数生成子も含まれます。

この値のうち，最大値が表定義情報用バッファ長より極端に少ない場合は，表定義情報用バッファ長が大き過ぎて無駄な領域を確保していると考えられるため，表定義情報用バッファ長の値を見直す必要があります。

76. 表アクセス権限情報取得回数

シングルサーバ及びフロントエンドサーバで，表操作のために表のアクセス権限を取得した回数です。

77. 表アクセス権限情報用バッファヒット回数

シングルサーバ及びフロントエンドサーバで，取得要求した表のアクセス権限が表アクセス権限情報用バッファ内で見付かった回数です。

この回数が取得回数より極端に少ない場合，一つの表に対して 100 を超えるユーザに権限を与えていると考えられます。その場合には，該当する表に対して PUBLIC で権限を与えられないか検討する必要があります。

78. ユーザ権限情報取得要求回数

UAP から，シングルサーバ及びフロントエンドサーバに対して CONNECT 要求をした回数です。

79. ユーザ権限情報用バッファヒット回数

UAP からシングルサーバ及びフロントエンドサーバに対して CONNECT 要求をした回数に伴って，取得要求したユーザ権限情報がユーザ権限情報用バッファ内で見付かった回数です。

この回数がユーザ権限情報取得要求回数より少ない場合，ユーザ権限情報用バッファ長を見直す必要があります。

80. ユーザ権限情報用バッファ使用ユーザ数

UAP から，シングルサーバ及びフロントエンドサーバに対して CONNECT 要求をした回数に伴って，取得要求したユーザ権限情報がユーザ権限情報用バッファ上に取得された件数と，ユーザの数の最大，最小，平均値です。

この数値うち，最大値×34（バイト）がユーザ権限情報用バッファ長より極端に小さい場合は，ユーザ権限情報用バッファ長を見直す必要があります。

81. ディクショナリサーバとの通信長（単位：バイト）

フロントエンドサーバで，取得要求した表定義情報が表定義情報用バッファ内で見付からなかった場合の表定義情報取得の要求件数と，ディクショナリサーバに対して通信をした通信長の最大，最小，平均値です。

82. ディクショナリサーバとの通信回数

フロントエンドサーバで取得要求した表定義情報が表定義情報用バッファ内で見付からなかった場合，ディクショナリサーバに対して表定義情報取得の通信をしたときの通信回数です。

83. ビュー解析情報取得要求回数

シングルサーバ及びフロントエンドサーバで，ビュー操作のためにビュー解析情報を取得した回数です。

84. ビュー解析情報用バッファヒット回数

シングルサーバ及びフロントエンドサーバで取得要求したビュー解析情報がビュー解析情報用バッファ内で見付かった回数です。

ビュー解析情報用バッファヒット回数がビュー解析情報取得要求回数より少ない場合は，ビュー解析情報用バッファ長を見直す必要があります。

85. ビュー解析情報用バッファ中の解析情報数

ビュー解析情報用バッファ上に取得したビュー解析情報の数です。

86. ビュー解析情報用バッファに取得した 1 ビュー定義情報当たりのビュー解析情報用バッファ使用領域長（単位：バイト）

ビュー解析情報をビュー解析情報用バッファに取得した回数と，1 ビュー解析情報当たりのビュー解析情報用バッファ使用領域長の最大，最小，平均値です。ビュー解析情報をビュー解析情報用バッファに取得した回数とバッファミスの発生回数（「VIEW DEF GET REQ−VIEW CACHE HIT」）が一致しない場合は，ビュー解析情報用バッファ長が不足しています。マニュアル「HiRDB システム運用ガイド」の「ビュー解析情報用バッファ長のチューニング」を参考にしてチューニングしてください。

87. 使用したビュー解析情報用バッファ長（単位：バイト）

ビュー解析情報用バッファ上にあったビュー解析情報の件数と，長さの最大，最小，平均値です。

この数値のうち，最大値がビュー解析情報用バッファ長より極端に少ない場合は，ビュー解析情報用バッファ長が大き過ぎて無駄な領域を確保していると考えられます。この場合，ビュー解析情報用バッファ長の値が妥当か検討する必要があります。

88. バッファミスとなったビュー解析情報長（単位：バイト）

発生回数，最大，最小，平均値には 0 を出力します。バッファミスしたビュー解析情報長は，チューニング要素として使用することはないため，考慮する必要はありません。なお，バッファミスの発生回数を確認する場合は，「VIEW DEF GET REQ−VIEW CACHE HIT」で求めてください。

89. 型定義情報取得要求回数

シングルサーバ及びフロントエンドサーバで，型定義を参照した回数です。

90. ユーザ定義型情報用バッファヒット回数

シングルサーバ及びフロントエンドサーバで，取得要求した型定義情報がユーザ定義型情報用バッファ内で見付かった回数です。

91. ユーザ定義型情報用バッファ中の型定義情報数

シングルサーバ及びフロントエンドサーバで，ユーザ定義型情報用バッファ内にバッファリングされた型定義情報の数と，型定義情報数の最大，最小，平均値です。

92. ユーザ定義型情報用バッファに取得した１型定義情報当たりのユーザ定義型情報用バッファ使用領域長（単位：バイト）

    シングルサーバ及びフロントエンドサーバで，ユーザ定義型情報用バッファに型定義情報が取得された件数と，１型定義情報当たりのユーザ定義型情報用バッファ使用領域長の最大，最小，平均値です。

93. ユーザ定義型情報用バッファ総使用領域長（単位：バイト）

    シングルサーバ及びフロントエンドサーバで，型定義情報を取得した件数と，ユーザ定義型情報用バッファ長の最大，最小，平均値です。

    この値の最大値が確保したユーザ定義型情報用バッファ長より小さい場合は，無駄な領域を確保していることが考えられるため，ユーザ定義型情報用バッファサイズの値を見直す必要があります。

94. 確保したユーザ定義型情報用バッファ長（単位：バイト）

    シングルサーバ及びフロントエンドサーバで，確保したユーザ定義型情報用バッファの個数と，ユーザ定義型情報用バッファ長の最大，最小，平均値です。この値の最大，最小，平均値はすべて同じ値になります。

95. ルーチン定義情報取得要求回数

    シングルサーバ及びフロントエンドサーバで，ルーチン定義を参照した回数です。

96. ルーチン定義情報用バッファヒット回数

    シングルサーバ及びフロントエンドサーバで，取得要求したルーチン定義情報がルーチン定義情報用バッファ内で見付かった回数です。

    ルーチン定義情報用バッファヒット回数がルーチン定義情報取得回数より少ない場合，ルーチン定義情報用バッファサイズの値を見直す必要があります。

97. ルーチン定義情報用バッファ中のルーチン定義情報数

    シングルサーバ及びフロントエンドサーバで，ルーチン定義情報用バッファ内にバッファリングされたルーチン定義情報の数と，ルーチン定義情報数の最大，最小，平均値です。

98. ルーチン定義情報用バッファに取得した１ルーチン定義情報当たりのルーチン定義情報用バッファ使用領域長（単位：バイト）

    シングルサーバ及びフロントエンドサーバで，ルーチン定義情報用バッファにルーチン定義情報が取得された件数と，１ルーチン定義情報当たりのルーチン定義情報用バッファ使用領域長の最大，最小，平均値です。

99. ルーチン定義情報用バッファ総使用領域長（単位：バイト）

    シングルサーバ及びフロントエンドサーバで，ルーチン定義情報用バッファに取得された件数と，使用したルーチン定義情報用バッファ長の最大，最小，平均値です。

100. 確保したルーチン定義情報用バッファ長（単位：バイト）

    シングルサーバ及びフロントエンドサーバで，確保したルーチン定義情報用バッファの個数と，ルーチン定義情報用バッファ長の最大，最小，平均値です。この値の最大，最小，平均値はすべて同じ値になります。

101. プラグイン提供関数のルーチン定義取得要求回数

シングルサーバ及びフロントエンドサーバで，プラグイン提供関数のルーチンを参照した回数です。

102. プラグイン提供関数のルーチン定義情報用バッファヒット回数

シングルサーバ及びフロントエンドサーバで，取得要求したプラグイン提供関数のルーチン情報がルーチン定義情報用バッファ内で見付かった回数です。

プラグイン提供関数のルーチン定義情報用バッファヒット回数がプラグイン提供関数のルーチン定義情報取得要求回数より少ない場合，ルーチン定義情報用バッファサイズを見直す必要があります。

103. レジストリ情報取得要求回数

シングルサーバ及びフロントエンドサーバで，レジストリ情報を参照した回数です。

104. レジストリ情報用バッファヒット回数

シングルサーバ及びフロントエンドサーバで，取得要求したレジストリ情報がレジストリ情報用バッファ内で見付かった回数です。

レジストリ情報用バッファヒット回数がレジストリ情報取得要求回数より少ない場合，レジストリ情報用バッファサイズの値を見直す必要があります。

105. レジストリ情報用バッファ中のレジストリ情報数

シングルサーバ及びフロントエンドサーバで，レジストリ情報用バッファ内にバッファリングされたレジストリ情報の数と，レジストリ情報数の最大，最小，平均値です。

106. 1 レジストリ情報当たりのレジストリ情報用バッファ長（単位：バイト）

シングルサーバ及びフロントエンドサーバで，レジストリ情報用バッファにレジストリ情報が取得された件数と，１レジストリ情報当たりのレジストリ情報用バッファ長の最大，最小，平均値です。

107. レジストリ情報用バッファ総使用領域長（単位：バイト）

シングルサーバ及びフロントエンドサーバで，レジストリ情報用バッファにレジストリ情報が取得された件数と，使用したレジストリ情報用バッファ長の最大，最小，平均値です。

108. ディレクトリ登録のユーザ認証時間（単位：マイクロ秒）※2

システムで使用する内部情報です。

109. グループ判定実行時間（単位：マイクロ秒）※2

システムで使用する内部情報です。

110. シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバの SQL オブジェクト情報

111. SQL オブジェクト取得要求回数

シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバで実行した SQL オブジェクトの取得回数です。

112. SQL オブジェクト用バッファヒット回数

シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバで，取得要求した SQL オブジェクトが SQL オブジェクト用バッファ内で見付かった回数です。

SQL オブジェクト用バッファヒット回数が SQL オブジェクト取得要求回数より少ない場合は，SQL オブジェクト用バッファ長を見直す必要があります。

113. SQL オブジェクト用バッファ中の SQL オブジェクト数

シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバの，SQL オブジェクト用バッファにある SQL オブジェクトの個数，SQL オブジェクト数の情報更新回数，最大，最小，平均値です。情報更新回数とは，この SQL オブジェクト数の最大，最小，平均値のどれかが更新された回数のことです。

114. SQL オブジェクト用バッファ中の SQL オブジェクトの合計長（単位：バイト）

シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバで，SQL オブジェクト用バッファにある SQL オブジェクト合計長の情報更新回数，最大，最小，平均値です。情報更新回数とは，この SQL オブジェクト合計長の最大，最小，平均値のどれかが更新された回数のことです。

注

HiRDB では，SQL オブジェクト用バッファの管理を，1 キロバイト単位で行います。このため，各 SQL オブジェクトは，キロバイト単位に切り上げられた SQL オブジェクト長以上のサイズの領域を，SQL オブジェクト用バッファで占有します。ここで出力される SQL オブジェクト合計長は，SQL オブジェクトが占有した領域のサイズの合計ではありません。SQL オブジェクト合計長が，SQL オブジェクト用バッファ長の指定値より小さい場合でも，SQL オブジェクト用バッファに空きがないことがあります。

115. SQL オブジェクト用バッファから出された SQL オブジェクトの数

シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバで，新たに要求された SQL オブジェクトを作成及び入力する空きバッファ作成のために，参照した後に使用されていない SQL オブジェクトをメモリ上で無効にした数です。

116. SQL オブジェクト長（単位：バイト）

シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバで，SQL オブジェクト用バッファ内に取得した SQL オブジェクトの件数と，SQL オブジェクトの長さの最大，最小，平均値です。

117. ストアドプロシジャのオブジェクトの取得要求回数

シングルサーバ及びフロントエンドサーバで，SQL オブジェクト用バッファに対してストアドプロシジャのオブジェクトを取得要求した回数です。

バックエンドサーバ及びディクショナリサーバの場合は，SQL オブジェクト取得要求回数に値が含まれます。

118. ストアドプロシジャのオブジェクトの，SQL オブジェクト用バッファヒット回数

シングルサーバ及びフロントエンドサーバで，取得要求したストアドプロシジャのオブジェクトが SQL オブジェクト用バッファ内で見付かった回数です。

シングルサーバ及びフロントエンドサーバで，ストアドプロシジャのオブジェクトの SQL オブジェクト用バッファヒット回数がストアドプロシジャのオブジェクトの取得要求回数より少ない場合は，SQL オブジェクト用バッファ長を見直す必要があります。

バックエンドサーバ及びディクショナリサーバの場合は，SQL オブジェクト用バッファヒット回数に値が含まれます。

119. SQL オブジェクト用バッファ中の，ストアドプロシジャのオブジェクト数

シングルサーバ及びフロントエンドサーバで，SQL オブジェクト用バッファ内に取得したストアドプロシジャのオブジェクトの件数と，オブジェクト数の最大，最小，平均値です。

バックエンドサーバ及びディクショナリサーバの場合は，SQL オブジェクト用バッファ中の SQL オブジェクト数に値が含まれます。

120. SQL オブジェクト用バッファ中の，ストアドプロシジャのオブジェクトの合計長（単位：バイト）

シングルサーバ及びフロントエンドサーバで，SQL オブジェクト用バッファにあるストアドプロシジャのオブジェクトを取得した回数と，オブジェクトの長さの最大，最小，平均値です。

バックエンドサーバ及びディクショナリサーバの場合は，SQL オブジェクト用バッファ中の SQL オブジェクトの合計長に値が含まれます。

121. SQL オブジェクト用バッファから出された，ストアドプロシジャのオブジェクトの数

シングルサーバ及びフロントエンドサーバで，新たに要求されたストアドプロシジャのオブジェクトを作成及び入力する空きバッファ作成のために，参照した後に使用されていないストアドプロシジャのオブジェクトをメモリ上で無効にした数です。

バックエンドサーバ及びディクショナリサーバの場合は，SQL オブジェクト用バッファから出された SQL オブジェクトの数に値が含まれます。

122. ストアドプロシジャのオブジェクト長（単位：バイト）

シングルサーバ及びフロントエンドサーバで，SQL オブジェクト用バッファ内に取得したストアドプロシジャのオブジェクトの件数と，オブジェクトの長さの最大，最小，平均値です。

バックエンドサーバ及びディクショナリサーバの場合は，SQL オブジェクト長に値が含まれます。

123. ストアドプロシジャのオブジェクトのリコンパイル回数

シングルサーバ及びフロントエンドサーバで，インデクスが変更されたときにストアドプロシジャのオブジェクトをリコンパイルした回数です。

バックエンドサーバ及びディクショナリサーバの場合は該当しません。

124. 排他制御用プールパーティションのスレッド間ロック待ち数

排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition オペランド又は pd_fes_lck_pool_partition オペランドに 2 以上を指定した場合）は，排他制御用プールパーティションごとに求めたスレッド間ロック待ち数を集計して，最大，最小，平均値を出力します。

125. ファイルへの書き込み時間※2

HiRDB システム又はサーバで，システムログファイルへ書き込みをした時間の最大，最小，及び平均値です。

システムログのファイル出力以外にも，システムログファイルの状態を変更するときの書き込み時間が加算されます。システムログファイルを二重化している場合には，系ごとの書き込み時間が加算されます。また，システムログブロックの書き込み時間だけでなく，システムログファイルのスワップなどでファイルの状態が変化する場合に発生する，ファイルの書き込み時間も含まれます。HiRDB のバージョ

ンが 08-04 より前の場合に取得した統計入力アンロードファイルを，08-04 以降の統計解析ユティリティで実行すると，平均値及び最大値に 0 を，最小値に 4.29G を表示します。

126. ファイルからの読み出し時間※2

HiRDB システム又はサーバで，ロールバックのときにシステムログファイルから読み出しをした時間の最大，最小，及び平均値です。HiRDB のバージョンが 08-04 より前の場合に取得した統計入力アンロードファイルを，08-04 以降の統計解析ユティリティで実行すると，平均値及び最大値に 0 を，最小値に 4.29G を表示します。

127. ログ出力バッファへの書き込み回数

HiRDB システム又はサーバで，ログ出力バッファへ書き込みをした回数です。HiRDB のバージョンが 08-04 より前の場合に取得した統計入力アンロードファイルを，08-04 以降の統計解析ユティリティで実行すると，発生回数に 0 を表示します。

128. 排他制御用プールパーティションのスレッド間ロック要求回数

排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition オペランド又は pd_fes_lck_pool_partition オペランドに 2 以上を指定した場合）は，排他制御用プールパーティションごとに求めたスレッド間ロック待ち数を集計して，最大，最小，平均値を出力します。

129. 排他制御用プールパーティションのスレッド間ロック待ち回数

排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition オペランド又は pd_fes_lck_pool_partition オペランドに 2 以上を指定した場合）は，排他制御用プールパーティションごとに求めたスレッド間ロック待ち数を集計して，最大，最小，平均値を出力します。

130. 排他制御用プールパーティションのスレッド間ロック待ち時間（単位：マイクロ秒）※1※2

pdstbegin -w コマンドを実行して，スレッド間ロック待ち時間を取得している場合に出力します。スレッド間ロック待ち時間を取得していない場合は，0 を出力します。

排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition オペランド又は pd_fes_lck_pool_partition オペランドに 2 以上を指定した場合）は，排他制御用プールパーティションごとに求めたスレッド間ロック待ち数を集計して，最大，最小，平均値を出力します。

131. ログフラッシュ待ち行列数

132. ログフラッシュ要求回数

133. ログフラッシュ待ち回数

134. ログフラッシュ要求時間※1※2

pdstbegin -w コマンドを実行して，スレッド間ロック待ち時間を取得している場合に出力します。スレッド間ロック待ち時間を取得していない場合は，0 を出力します。

注※1

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

注※2

求めた時間が OS の時刻戻しなどによって負の値となる場合は，値が 0 の統計情報として出力します。そのため，表示する値は実際の値より小さいことがあります。

14.3.3　UAP に関する統計情報

UAP に関する統計情報を次に示します。

```
pdstedit VV-RR(Object Option) ***** UAP INFORMATION *****
INPUT          :c:¥tmp¥stjdata [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
--------------------------------------------------------------------------------
HOST = test [3]
--------------------------------------------------------------------------------
EDIT TIME 2003/04/03 22:00:00 - 2003/04/03 23:00:00 [4]
UAP NAME [5]
  SERVICE NAME [6]               [7]   [8]   [9]   [10]  [11]  [12]  [13]
                                 EXEC  NORM  ERROR TTIME TAVG  CACHE LOCK
------------------------------   ----- ----- ----- ----- ----- ----- -----
                                 [14]  [15]  [16]  [17]  [18]  [19]
                                 LMAX  LAVG  CTIME CTAVG SVEXT SVEXA
                                 ----- ----- ----- ----- -----
                                 [20]  [21]  [22]  [23]  [24]  [25]  [26]
   *SQL OBJECT*                  REQ   RAVG  HITS  HAVG  CRT   CAVG  SMAX
                                 ----- ----- ----- ----- ----- ----- -----
         [27]  [28]  [29]  [30]  [31]  [32]  [33]  [34]  [35]  [36]  [37]
   *LN1* COMT  ROLB  FROW  DROW  IROW  UROW  SET   OPEN  FETC  CLOS  DESC
         ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
         [38]  [39]  [40]  [41]  [42]  [43]  [44]  [45]  [46]  [47]  [48]
   *LN2* SEL   INS   UPD   DEL   LOCK  CRTT  DRPT  ALTT  CRTI  DRPI  CMTT
         ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
         [49]  [50]  [51]  [52]  [53]  [54]  [55]  [56]  [57]  [58]  [59]
   *LN3* CMTC  CRTS  DRPS  GRTR  GRTS  GRTA  GRTC  GRTD  RVKR  RVKS  RVKA
         ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
         [60]  [61]  [62]  [63]  [64]  [65]  [66]  [67]  [68]  [69]  [70]
   *LN4* RVKC  RVKD  CRTV  DRPV  PRGT  CRTP  DRPP  ALTP  CALL  DESI  MISC
         ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
                                 [71]  [72]  [73]  [74]
   *STORED ROUTINE OBJECT*       REQ   RAVG  HITS  HAVG
                                 ----- ----- ----- -----
                                 [75]  [76]  [77]  [78]  [79]
   *DB ACCESS*                   MAXIO MINIO MWFN  MWFEC MWFVL
                                 ----- ----- ----- ----- -----
                                 [80]  [81]  [82]  [83]
                                 MWHTS MBSL1 MBSL2 MBSL3
                                 ----- ----- ----- -----
         [84]  [85]  [86]  [87]  [88]  [89]  [90]  [91]  [92]  [93]  [94]
   *LN1* IOTIM DIDRC DIDUC DIDHC DIDHR DIDRD DIDWT LBRFC LBUPC LBRHC LBUHC
         ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
         [95]  [96]  [97]  [98]  [99]  [100] [101] [102] [103] [104] [105]
   *LN2* LRFHR LUPHR LBRDC LBWTC BFSHC BRDWC BWTWC BLKWC WFRDC WFWTC WBFOC
         ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
         [106] [107] [108] [109] [110] [111] [112] [113]  [114] [115] [116]
   *LN3* SCHSK SCHCH LDIRC LDIUC LDIHC LDIRD LDIWT LBFSHC ARREQ ARWC  ARWT
         ----- ----- ----- ----- ----- ----- ----- -----  ----- ----- -----
         [117] [118] [119] [120] [121] [122] [123] [124] [125] [126]
   *LN4* ARSTA HJMAX HJAVG TUSEG COMPC COMPT CMPMT EXTEC EXTET EXTMT
         ----- ----- ----- ----- ----- ----- ----- -----  ----- -----
   *THREAD LOCK*
         [130] [131] [132] [133] [134] [135] [136] [137] [138] [139] [140]
   *LN1* LCKTC LCKWC LCKWT LCKTM LCKWQ LCKQM JBFTC JBFWC JBFWT JBFTM JBFWQ
```

```
          ----- ----- ----- ----- ----- ----- ----- ----- ----- ----- -----
          [141] [142] [143] [144] [145] [146] [147]
   *LN2* JBFQM BUFTC BUFWC BUFWT BUFTM BUFWQ BUFQM
          ----- ----- ----- ----- ----- ----- -----
uap01
  service01
                                        12    12     0 1.70k   142 26.0k    111
                                        37     9   260    22 1.58k   132
   *SQL OBJECT*                        108     9     0     0   108     9  9.19k
   *LN1*     96      0   168     0     0     0   108   108   204   108      0 [127]
      **      8      0    14     0     0     0     9     9    17     9      0 [128]
   *LN2*      0      0     0     0     0     0     0     0     0     0      0
      **      0      0     0     0     0     0     0     0     0     0      0
   *LN3*      0      0     0     0     0     0     0     0     0     0      0
      **      0      0     0     0     0     0     0     0     0     0      0
   *LN4*      0      0     0     0     0     0     0     0     0     0      0
      **      0      0     0     0     0     0     0     0     0     0      0
   *STORED ROUTINE OBJECT*             0     0     0     0
   *DB ACCESS*                         8     0     0     0     0
                                       0     0     0     0
   *LN1*     41  2.32k     0 2.26k    97    64     0     0     0     0      0
      **      3    194     0   188           5     0     0     0     0      0
   *LN2*      0      0     0     0    64     0     0     0     0     0      0
      **                   0     0     5     0     0     0     0     0      0
   *LN3*      0      0     0     0     0     0     0     0     0     0      0
      **                   0     0     0     0     0     0     0     0      0
   *LN4*      0      0     0
      **      0      0     0
   *THREAD LOCK*
   *LN1*  1.31k      8   542   140     0     0    84    38  466k 26.5k      7
      **                 68           0                    12.3k            0
   *LN2*      1  4.71k     1    74    74     0     0
      **                        74           0
uap02
  service02
                                        8     8     0 3.43k   428 5.93k    676
                                      150    85   140    18 3.36k   420
   *SQL OBJECT*                        56     7     5     1    51     6  3.02k
   *LN1*     96      0   204     0     12     0   124   124   256   124     16
      **      5      0    10     0      1     0     6     6    13     6      1
   *LN2*      0      0    36     0     12     0    16    16    52    16     16
      **      0      2     0     0      0     0     0     0     0     0      0
   *LN3*      0      0     0     0      0     0     0     0     0     0      0
      **      0      0     0     0      0     0     0     0     0     0      0
   *LN4*      0      0     0     0      0     0     0     0     0     0      0
      **      0      0     0     0      0     0     0     0     0     0      0
   *STORED ROUTINE OBJECT*             0     0     0     0
   *DB ACCESS*                        83     0     0     0     0
                                       0     0     0     0
   *LN1*    906 2.44k   156 1.60k     66   833    66     0     0     0      0
      **      3   259     0   224           35     0     0     0     0      0
   *LN2*      0     0     0     0    900     0     1    10     0     0      0
      **                   0     0    45     0     0     1     0     0      0
   *LN3*      0     0     0     0     0     0     0     0     0     0      0
      **                   0     0     0     0     0     0     0     0      0
   *LN4*      0     0     0    14     0     0     0    24     0     0
      **      0     0     0
   *THREAD LOCK*
```

```
   *LN1* 2.34k     8   310    62       0     0    35     1  121k  121k      0
      **                 39             0                      121k            0
   *LN2*     0 5.83k     2 7.54k   7.48k     0     0
      **                   3.77k            0
*TOTAL* [129]                             33    33     0 11.6k   351  4.36k     0
                                           0     0 1.88k    57 10.9k    330
   *SQL OBJECT*                          242     7    66     2   176      5 4.13k
   *LN1*   233     0   993     0       0     0   242   242 1.02k   225     0
      **   113   306    20   200           104     8     0     0     0     0
   *LN2*     0     0     0     0     836     0     1    10     0     0     0
      **                 0     0     105     0     0     1     0     0     0
   *LN3*     0     0     0     0       0     0     0     0     0     0     0
      **     0     0     0     0       0     0     0     0     0     0     0
   *LN4*     0     0     0    14       0     0     0    24     0     0     0
      **     0     0     0     0       0     0     0     0     0     0     0
   *STORED ROUTINE OBJECT*              0     0     0     0
   *DB ACCESS*                         83     0     0     0     0
                                        0     0     0     0
   *LN1*   948 4.77k   156 3.86k      81   897    66     0     0     0     0
      **    47   238     8   193           45     3     0     0     0     0
   *LN2*     0     0     0     0     900     0     1    10     0     0     0
      **                 0     0      45     0     0     1     0     0     0
   *LN3*     0     0     0     0       0     0     0     0     0     0     0
      **                 0     0       0     0     0     0     0     0     0
   *LN4*     0     0     0    14       0     0     0    24     0     0
      **     0     0     0
   *THREAD LOCK*
   *LN1* 3.65k    16   852   140       0     0   119    39  586k   121k     7
      **                 53             0                   15.0k            0
   *LN2*     1 10.5k     3 7.62k   7.48k     0     0
      **                   2.54k            0
-------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                 LAST                          NUM
STJ         uap        2012/02/03 09:29:28   2012/02/03 09:35:13            20
-------------------------------------------------------------------------------

NO FILE KIND:LOG FILE NAME
  LOG KIND             FIRST                 LAST                          NUM
 1 STJ      :c:¥tmp¥stjdata¥pdstj1
  sys                  2003/04/03 22:46:30   2003/04/03 22:46:30            2
  uap                  2003/04/03 22:45:56   2003/04/03 22:47:53           32
  sql                  2003/04/03 22:45:56   2003/04/03 22:47:53          898
  sop                  2012/02/03 09:29:27   2012/02/03 09:31:04           98
  dop                  2012/02/03 09:29:27   2012/02/03 09:31:04           92
  pcd                  ****/**/** **:**:**   ****/**/** **:**:**            0
  obj                  ****/**/** **:**:**   ****/**/** **:**:**            0
  sqh                  ****/**/** **:**:**   ****/**/** **:**:**            0
  buf                  2012/02/03 09:30:00   2012/02/03 09:32:24           21
  fil                  2012/02/03 09:30:00   2012/02/03 09:32:24          224
  dfw                  2012/02/03 09:29:30   2012/02/03 09:32:24           71
  idx                  2012/02/03 09:30:00   2012/02/03 09:32:24            7
  fsv                  ****/**/** **:**:**   ****/**/** **:**:**            0
  hba                  ****/**/** **:**:**   ****/**/** **:**:**            0
 2 STJ      :c:¥tmp¥stjdata¥pdstj2
  sys                  2012/02/03 09:34:58   2012/02/03 09:37:00            6
  uap                  2012/02/03 09:34:00   2012/02/03 09:36:17           19
  sql                  2012/02/03 09:33:59   2012/02/03 09:36:17          456
  sop                  2012/02/03 09:34:02   2012/02/03 09:36:16           65
```

```
dop                 2012/02/03 09:34:02   2012/02/03 09:36:16          96
pcd                 ****/**/** **:**:**   ****/**/** **:**:**           0
obj                 ****/**/** **:**:**   ****/**/** **:**:**           0
sqh                 ****/**/** **:**:**   ****/**/** **:**:**           0
buf                 2012/02/03 09:34:10   2012/02/03 09:36:19          27
fil                 2012/02/03 09:34:10   2012/02/03 09:36:19         288
dfw                 2012/02/03 09:34:04   2012/02/03 09:36:19          41
idx                 2012/02/03 09:34:10   2012/02/03 09:36:19           9
fsv                 ****/**/** **:**:**   ****/**/** **:**:**           0
hba                 ****/**/** **:**:**   ****/**/** **:**:**           0
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時–出力終了日時）
3. UAP 又はサービスに関する統計情報の出力要求元ホスト名
4. 編集時間の範囲（取得開始時刻–取得終了時刻）
5. UAP 名称

   統計情報を編集した UAP の名称です。
6. サービス名称

   統計情報を編集したサービスの名称です。HiRDB にアクセスした UAP の種類によって，出力されるサービス名称は異なります。※1
7. UAP 又はトランザクションの実行回数

   統計ログから集計した同一 UAP 又はトランザクションの実行回数です。

   UAP 実行単位に UAP 統計情報を取得する場合，UAP が DISCONNECT を発行した回数を実行回数としてカウントしています。
8. UAP 又はトランザクションの正常終了回数

   統計ログから集計した同一 UAP 又はトランザクションの正常終了回数です。
9. UAP 又はトランザクションの異常終了回数

   統計ログから集計した同一 UAP 又はトランザクション内で最後に実行したトランザクションがロールバックした状態で終了した回数です。
10. UAP 又はトランザクションの実行時間（単位：ミリ秒）

    統計ログから集計した同一 UAP 又はトランザクションの実行時間の合計値です。

    UAP 実行単位に UAP 統計情報を取得する場合，UAP が CONNECT を発行してから DISCONNECT を発行するまでの経過時間を実行時間としています。
11. UAP 又はトランザクションの平均実行時間（単位：ミリ秒）

    統計ログから集計した同一 UAP 又はトランザクションの実行時間の平均値です。

    UAP 実行単位に UAP 統計情報を取得する場合，UAP が CONNECT を発行してから DISCONNECT を発行するまでの経過時間を実行時間としています。
12. 表定義情報用バッファ使用サイズ（単位：バイト）

統計ログから集計した同一 UAP 又はトランザクションで取得した表定義情報の，表定義情報用バッファの使用サイズの平均値です。

13. UAP，又はトランザクションの排他待ち時間の合計値（単位：ミリ秒）※2

    UAP，又はトランザクションでの排他待ち時間の合計値です。

14. UAP，又はトランザクションの排他待ち時間の最大値（単位：ミリ秒）※2

    UAP，又はトランザクションでの排他待ち時間の最大値です。

15. 1UAP，又は 1 トランザクション当たりの排他待ち時間の平均値（単位：ミリ秒）※2

    1UAP，又は 1 トランザクション当たりの排他待ち時間の平均値です。

16. トランザクション内のすべてのサーバの CPU 時間の合計値（単位：ミリ秒）

    CPU 時間の精度は，使用している OS，及びハードウェアに依存します。したがって，CPU 時間の精度が 1 ミリ秒より大きい環境の場合，実際の値より小さい値が表示されることがあります。

17. 1UAP 又は 1 トランザクション当たりの CPU 時間の平均値（CTIME÷EXEC）（単位：ミリ秒）

    CPU 時間の精度は，使用している OS，及びハードウェアに依存します。したがって，CPU 時間の精度が 1 ミリ秒より大きい環境の場合，実際の値より小さい値が表示されることがあります。

18. UAP 又はトランザクションの，シングルサーバ又はフロントエンドサーバの処理時間の合計（単位：ミリ秒）

19. 1UAP 又は 1 トランザクション当たりの，シングルサーバ又はフロントエンドサーバの処理時間の平均値（SVEXT÷EXEC）（単位：ミリ秒）

20. SQL オブジェクト取得要求回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して，SQL オブジェクトの取得要求をした回数です。

21. UAP 又はトランザクション単位の SQL オブジェクト取得要求回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して，SQL オブジェクトの取得要求をした 1 回の UAP 又はトランザクション実行単位の平均取得要求回数です。

22. SQL オブジェクト用バッファヒット回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して，取得要求した SQL オブジェクトが SQL オブジェクト用バッファ内で見付かった回数です。

23. UAP 又はトランザクション単位の SQL オブジェクト用バッファヒット回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して，取得要求した SQL オブジェクトが SQL オブジェクト用バッファで見付かった，1 回の UAP 又はトランザクション実行単位の平均ヒット回数です。

24. SQL オブジェクト作成回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して SQL オブジェクトを作成した回数です。

25. UAP 又はトランザクション単位の SQL オブジェクト作成回数

集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して，SQL オブジェクトを作成した 1 回の UAP 又はトランザクション実行単位の平均作成回数です。

この値が大きい場合は，SQL オブジェクト用バッファ長を大きくする必要があります。

26. 作成した SQL オブジェクトの最大長（単位：バイト）

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して作成した SQL オブジェクトの最大長です。

27. COMMIT 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した COMMIT 文の実行回数です。

28. ROLLBACK 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した ROLLBACK 文の実行回数です。

29. 実検索行数

    集計した時間内の，同一 UAP 又はトランザクションで発行した FETCH 文及び SELECT 文の実行によって，UAP 又はトランザクションに返した検索行数です。

30. 実削除行数

    集計した時間内の，同一 UAP 又はトランザクションで発行した DELETE 文の実行によって削除した行数です。

31. 実挿入行数

    集計した時間内の，同一 UAP 又はトランザクションで発行した INSERT 文の実行によって挿入した行数です。

32. 実更新行数

    集計した時間内の，同一 UAP 又はトランザクションで発行した UPDATE 文の実行によって更新した行数です。

33. SETUP 実行回数

    集計した時間内の，同一 UAP 又はトランザクションでの前処理実行回数です。

34. OPEN 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションでの OPEN 文の実行回数です。

35. FETCH 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションでの FETCH 文の実行回数です。

36. CLOSE 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションでの CLOSE 文の実行回数です。

37. DESCRIBE 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションでの DESCRIBE 文の実行回数です。

38. SELECT 文実行回数

    集計した時間内の，同一 UAP 又はトランザクションでの SELECT 文の実行回数です。

39. INSERT 文実行回数

集計した時間内の，同一 UAP 又はトランザクションでの INSERT 文の実行回数です。

40. UPDATE 文実行回数

集計した時間内の，同一 UAP 又はトランザクションでの UPDATE 文の実行回数です。

41. DELETE 文実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DELETE 文の実行回数です。

42. LOCK 文実行回数

集計した時間内の，同一 UAP 又はトランザクションでの LOCK 文の実行回数です。

43. CREATE TABLE 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの CREATE TABLE の実行回数です。

44. DROP TABLE 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DROP TABLE の実行回数です。

45. ALTER TABLE 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの ALTER TABLE の実行回数です。

46. CREATE INDEX 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの CREATE INDEX の実行回数です。

47. UAP 又はトランザクションで発行した DROP INDEX の実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DROP INDEX の実行回数です。

48. COMMENT（TABLE）実行回数

集計した時間内の，同一 UAP 又はトランザクションでの COMMENT（TABLE）の実行回数です。

49. COMMENT（COLUMN）実行回数

集計した時間内の，同一 UAP 又はトランザクションでの COMMENT（COLUMN）の実行回数です。

50. CREATE SCHEMA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの CREATE SCHEMA の実行回数です。

51. DROP SCHEMA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DROP SCHEMA の実行回数です。

52. GRANT RDAREA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの GRANT RDAREA の実行回数です。

53. GRANT SCHEMA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの GRANT SCHEMA 及び GRANT SCHEMA OPERATION の実行回数です。

54. GRANT アクセス権限実行回数

集計した時間内の，同一 UAP 又はトランザクションでの GRANT アクセス権限の実行回数です。

55. GRANT CONNECT 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの GRANT CONNECT の実行回数です。

56. GRANT DBA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの GRANT DBA の実行回数です。

57. REVOKE RDAREA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの REVOKE RDAREA の実行回数です。

58. REVOKE SCHEMA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの REVOKE SCHEMA 及び REVOKE SCHEMA OPERATION の実行回数です。

59. REVOKE アクセス権限実行回数

集計した時間内の，同一 UAP 又はトランザクションでの REVOKE アクセス権限の実行回数です。

60. REVOKE CONNECT 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの REVOKE CONNECT の実行回数です。

61. REVOKE DBA 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの REVOKE DBA の実行回数です。

62. CREATE VIEW 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの CREATE VIEW の実行回数です。

63. DROP VIEW 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DROP VIEW の実行回数です。

64. PURGE TABLE 文実行回数

集計した時間内の，同一 UAP 又はトランザクションでの PURGE TABLE 文の実行回数です。

65. CREATE PROCEDURE 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの CREATE PROCEDURE の実行回数です。

66. DROP PROCEDURE 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DROP PROCEDURE の実行回数です。

67. ALTER PROCEDURE 実行回数

集計した時間内の，同一 UAP 又はトランザクションでの ALTER PROCEDURE の実行回数です。

68. CALL 文実行回数

集計した時間内の，同一 UAP 又はトランザクションでの CALL 文の実行回数です。

69. DESCRIBE 文（INPUT）実行回数

集計した時間内の，同一 UAP 又はトランザクションでの DESCRIBE 文（INPUT）の実行回数です。

70. そのほかの実行回数

集計した時間内の，同一 UAP 又はトランザクションでのそのほかの実行回数です。

71. ストアドプロシジャのオブジェクトの取得要求回数

集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対してストアドプロシジャのオブジェクトの取得要求をした回数です。

72. UAP 又はトランザクション単位の，ストアドプロシジャのオブジェクトの取得要求回数

    集計した時間内で，同一 UAP 又はトランザクションで発行した SQL 文に対してストアドプロシジャのオブジェクトの取得要求をした場合の，1 回の UAP 又はトランザクション実行単位の平均取得要求回数です。

73. SQL オブジェクト用バッファヒット回数

    集計した時間内の，同一 UAP 又はトランザクションで発行した SQL 文に対して取得要求したストアドプロシジャのオブジェクトが，SQL オブジェクト用バッファ内で見付かった回数です。

74. UAP 又はトランザクション単位の SQL オブジェクト用バッファヒット回数

    集計した時間内で，同一 UAP 又はトランザクションで発行した SQL 文に対して取得要求したストアドプロシジャのオブジェクトが，SQL オブジェクト用バッファ内で見付かった場合の，1 回の UAP 又はトランザクション実行単位の平均ヒット回数です。

75. 1 回のデータベースの入出力に掛かった最大時間（単位：ミリ秒）

76. 1 回のデータベースの入出力に掛かった最小時間（単位：ミリ秒）

77. UAP での作業表用ファイルの最大使用数

78. UAP での作業表用ファイルの最大増分数

79. UAP での作業表用ファイルの最大容量（単位：メガバイト）

80. ハッシュジョイン，副問合せのハッシュ実行で，処理したハッシュデータを一度に展開するために必要なハッシュサイズの推定値（単位：キロバイト）

81. ハッシュジョイン，副問合せのハッシュ実行で，1 レベルバケット分割の後の最大バケットサイズ（単位：キロバイト）

82. ハッシュジョイン，副問合せのハッシュ実行で，2 レベルバケット分割の後の最大バケットサイズ（単位：キロバイト）

83. ハッシュジョイン，副問合せのハッシュ実行で，3 レベルバケット分割の後の最大バケットサイズ（単位：キロバイト）

84. データベースに対する入出力時間の合計（単位：ミリ秒）※3

85. データページ，インデクスページ，及びディレクトリページを参照した回数※3

86. データページ，インデクスページ，及びディレクトリページを更新した回数※3

87. データページ，インデクスページ，及びディレクトリページがバッファヒットした回数※3

88. データページ，インデクスページ，及びディレクトリページのバッファヒット率（DIDHC÷DIDRC）（単位：%）

89. データページ，インデクスページ，及びディレクトリページの実 READ 数※3

90. データページ，インデクスページ，及びディレクトリページを実 WRITE 数※3

91. LOB 列のデータページを参照した回数※3

92. LOB 列のデータページを更新した回数※3

93. LOB 列のデータページの参照でバッファヒットした回数※3
94. LOB 列のデータページの更新でバッファヒットした回数※3
95. LOB 列のデータページの参照でのバッファヒット率（LBRHC÷LBRFC）（単位：%）
96. LOB 列のデータページの更新でのバッファヒット率（LBUHC÷LBUPC）（単位：%）
97. LOB 列のデータページの実 READ 回数※3
98. LOB 列のデータページの実 WRITE 回数※3
99. グローバルバッファフラッシュ回数※3
100. グローバルバッファ READ 待ち発生回数※3
101. グローバルバッファ WRITE 待ち発生回数※3
102. グローバルバッファ排他待ち発生回数※3
103. 作業表のデータを，作業表用ファイルからグローバルバッファへ読み込んだ回数※3
104. 作業表のデータを，グローバルバッファから作業表用ファイルへ書き込んだ回数※3
105. 作業表用バッファが不足したため，使用中の作業表用バッファを強制的に作業表用ファイルへ出力した回数※3
106. SEGMENT REUSE 指定の表で，新規ページ追加モードから空きページ再利用モードに切り替わった直後のページサーチで，再利用できる空き領域がなく，再度新規ページ追加モードへ戻った回数※3
107. SEGMENT REUSE 指定の表で，新規ページ追加モードから空きページ再利用モードに切り替わった回数※3
108. 該当する UAP から，ローカルバッファを使用してデータページ，及びインデクスページを参照した回数※3
109. 該当する UAP から，ローカルバッファを使用してデータページ，及びインデクスページを更新した回数※3
110. ローカルバッファでのデータページ，及びインデクスページのバッファヒット回数※3
ランダムアクセスする UAP で，バッファヒット率（LDIHC÷LDIRC×100）が低い場合は，そのローカルバッファのチューニングをする必要があります。
111. 該当する UAP からの，ローカルバッファを使用したデータページ，及びインデクスページの実 READ 回数※3
プリフェッチ機能を使用している場合，プリフェッチで先読みした READ 回数も含まれます。バッファヒット率（LDIHC÷LDIRC×100）が低い場合，実 READ 回数も多くなります。
112. 該当する UAP からの，ローカルバッファを使用したデータページ，及びインデクスページの実 WRITE 回数※3
113. ローカルバッファフラッシュ回数（新たなページを入力する空きバッファを作成するために，バッファを無効にした回数）※3

114. 非同期 READ 機能使用時，非同期 READ プロセスに一括先読み処理を要求した回数※3

115. 非同期 READ 機能使用時，非同期 READ プロセスの一括先読み処理での同期待ち発生回数※3

116. 非同期 READ 機能使用時，非同期 READ プロセスの一括先読み処理での同期待ち時間（単位：ミリ秒）※3

117. 非同期 READ 機能使用時，最初の先頭ページ一括読み込みでの同期 READ 時間（単位：ミリ秒）※3

118. ハッシュジョイン，副問合せのハッシュ実行でのハッシュ分割表検索時の最大比較回数

119. ハッシュジョイン，副問合せのハッシュ実行でのハッシュ分割表検索時の平均比較回数※3

120. 一時表を使用する SQL 実行時の使用セグメント数の合計

121. データ圧縮処理の総実行回数※4

圧縮列にデータを挿入，及び圧縮列のデータを更新するときに発生したデータ圧縮処理の総実行回数です。

122. データ圧縮処理の総実行時間（単位：ミリ秒）※4

圧縮列にデータを挿入，及び圧縮列のデータを更新するときに発生した，すべてのデータ圧縮処理に掛かった時間です。

123. データ圧縮処理の最大実行時間（単位：ミリ秒）※4

圧縮列にデータを挿入，又は圧縮列のデータを更新するときに発生した，データ圧縮処理 1 件ごとに掛かった時間の最大値です。

124. データ伸張処理の総実行回数※4

圧縮列のデータを検索，又は更新するときに発生したデータ伸張処理の総実行回数です。

125. データ伸張処理の総実行時間（単位：ミリ秒）※4

圧縮列のデータを検索，又は更新するときに発生した，すべてのデータ伸張処理に掛かった時間（概算値）です。

126. データ伸張処理の最大実行時間（単位：ミリ秒）※4

圧縮列のデータを検索，又は更新するときに発生した，データ伸張処理 1 件ごとに掛かった時間の最大値です。

127. 1 回の UAP 又はトランザクション実行ごとの，各項目の合計値

128. 1 回の UAP 又はトランザクション実行ごとの，各項目の平均値

129. 全体の各項目の合計値

130. 排他制御用プールパーティションのスレッド間ロック要求回数

131. 排他制御用プールパーティションのスレッド間ロック待ち回数

132. 排他制御用プールパーティションのスレッド間ロック待ち時間の合計値（単位：マイクロ秒）※5※6

この UAP の排他制御用プールパーティションのスレッド間ロック待ち時間の合計値を表示します。平均値は排他制御用プールパーティションのスレッド間ロック待ち 1 件の待ち時間です。

133. 排他制御用プールパーティションのスレッド間ロック待ち時間の最大値※5※6

この UAP の排他制御用プールパーティションのスレッド間ロック待ち時間の最大値を表示します。

134. 排他制御用プールパーティションのスレッド間ロック待ち行列数の合計値

この UAP の排他制御用プールパーティションのスレッド間ロック待ち行列数を表示します。平均値は，排他制御用プールパーティションのスレッド間ロック 1 件の待ち時間です。

135. 排他制御用プールパーティションの待ち行列数の最大値

この UAP の排他制御用プールパーティションのスレッド間ロック待ち行列数の最大値を表示します。

136. ログフラッシュ要求回数

137. ログフラッシュ待ち回数

138. ログフラッシュ待ち時間の合計値（単位：マイクロ秒）※5※6

この UAP のログフラッシュのスレッド間ロック待ち時間の合計値を表示します。平均値は，ログフラッシュのスレッド間ロック 1 件の待ち時間です。

139. ログフラッシュ待ち時間の最大値（単位：マイクロ秒）※5※6

この UAP のログフラッシュのスレッド間ロック待ち時間の最大値を表示します。

140. ログフラッシュ待ち行列数の合計値

この UAP のログフラッシュのスレッド間ロック待ち行列数の合計値を表示します。平均値は，ログフラッシュのスレッド間ロック 1 件の待ち行列数です。

141. ログフラッシュ待ち行列数の最大値

この UAP のログフラッシュ待ち行列数の最大値を表示します。

142. グローバルバッファプールのスレッド間ロック要求回数

143. グローバルバッファプールのスレッド間ロック待ち回数

144. グローバルバッファプールのスレッド間ロック待ち時間の合計値（単位：マイクロ秒）※5※6

この UAP のグローバルバッファの排他獲得処理によるスレッド間ロック待ち時間の合計値を表示します。平均値は，グローバルバッファプールのスレッド間ロック待ち 1 件の待ち時間です。

145. グローバルバッファプールのスレッド間ロック待ち時間の最大値（単位：マイクロ秒）※5※6

この UAP のグローバルバッファの排他獲得処理によるスレッド間ロック待ち時間の最大値を表示します。

146. グローバルバッファプールのスレッド間ロック待ち行列数

この UAP のグローバルバッファの排他獲得処理によるスレッド間ロック待ち行列数の合計値を表示します。平均値は，グローバルバッファプールのスレッド間ロック待ち 1 件の待ち行列数です。

147. グローバルバッファプールのスレッド間ロック待ち行列数の最大値

この UAP のグローバルバッファの排他獲得処理によるスレッド間ロック待ち行列数の最大値を表示します。

注※1

UAP の種類と出力されるサービス名称を次に示します。

UAP の種類		出力されるサービス名称
OpenTP1 の UAP	OpenTP1 の SUP（サービス利用プログラム）が SPP（サービス提供プログラム）に要求したサービス	該当するサービス名称
	TP1/Message Control が MHP（メッセージ処理プログラム）に要求したサービス	
	上記以外	連続した 31 個の＊（アスタリスク）
OpenTP1 以外の UAP		

注※2

HiRDB/パラレルサーバの場合，トランザクション内で発生したすべての排他待ち時間（並列処理も含む）の合計値となります。

注※3

トランザクション単位に情報を出力する場合は，CONNECT からの累積値となります。

注※4

この項目は BINARY 型のデータに対する情報です。

注※5

pdstbegin コマンドの-w オプションを指定し，スレッド間ロック待ち時間を取得している場合に出力します。pdstbegin コマンドの-w オプションについては，「pdstbegin（統計情報の出力開始）のオプション」を参照してください。

注※6

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

14.3.4　SQL に関する統計情報

SQL に関する統計情報を次に示します。

```
pdstedit VV-RR          ***** SQL INFORMATION *****
INPUT          :c:¥tmp¥pdstj1 [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
--------------------------------------------------------------------------------
HOST = test [3]
--------------------------------------------------------------------------------
FES = fes01 [4]
--------------------------------------------------------------------------------
EDIT TIME 2001/09/30 00:00:00 - 2001/09/31 00:00:00 [5]
UAP NAME [6]
  SERVICE NAME [7]                [8]   [9]   [10]  [11]
                                  PDATA PDAVG BES#  B#AVG
------------------------------    ----- ----- ----- -----
         [12] [13]     [14]  [15]  [16]  [17]  [18]  [19]  [20]  [21]
         KIND SERV     EXEC  TTIME FTCR  INSR  UPDR  DELR  CRTL  DRPL
         ---- -------- ----- ----- ----- ----- ----- ----- ----- -----
```

```
uap01
  service01
                                    745k   270      0      0
   *CNT* SQL            2.82k  272M     0   532      0      0      0      0 [22]
      **                       96.5k     0     0      0      0      0      0 [23]
   *CNT* FSQL              0      0      0     0      0      0 [22]
      **                            0     0     0      0      0 [23]
uap02
  service02
                                    467k    60      0      0
   *CNT* SQL            7.90k 79.0M 7.55k     0      0      0      0      0
      **                       10.0k     1     0      0      0      0      0
   *CNT* FSQL              0      0      0     0      0      0
      **                            0     0     0      0      0
uap03
  service03
                                   11.2M   461      0      0
   *CNT* SQL            25.5k  310M 5.39k     0      0      0      0      0
      **                       12.2k     0     0      0      0      0      0
   *CNT* FSQL              0      0      0     0      0      0
      **                            0     0     0      0      0
uap04
  service04
                                   7.50M   534      0      0
   *CNT* SQL            14.7k  192M 5.99k   716    728    716      0      0
      **                       13.1k     0     0      0      0      0      0
   *CNT* FSQL              0      0      0     0      0      0
      **                            0     0     0      0      0
*TOTAL*[24]                        19.9M   410      0      0
   *CNT* SQL            51.0k  854M 18.9k 1.25k   728    716      0      0
      **                       16.8k     0     0      0      0      0      0
   *CNT* FSQL              0      0      0     0      0      0
      **                            0     0     0      0      0
--------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                 LAST                          NUM
STJ         sql        2001/09/30 10:36:37   2001/09/30 12:34:03         50971
--------------------------------------------------------------------------------
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時−出力終了日時）
3. SQL に関する統計情報の出力要求元ホスト名
4. SQL を実行したフロントエンドサーバの名称
5. 編集時間の範囲（取得開始時刻−取得終了時刻）
6. UAP 名称

   統計情報を編集した UAP の名称です。
7. サービス名称

統計情報を編集したサービスの名称です。HiRDB にアクセスした UAP の種類によって，出力されるサービス名称は異なります。※

8. 作成した SQL オブジェクトの長さ（単位：バイト）
   UAP 又はサービス内で発行した SQL 文の実行によって作成した，SQL オブジェクトの長さです。
9. 作成した SQL オブジェクトの長さの平均値（単位：バイト）
   UAP 又はサービス内で発行した SQL 文に対して作成した，SQL オブジェクトの長さの平均値です。
   この値は，作成した SQL オブジェクトの長さ÷SQL 実行回数で算出した値です。
10. 使用したバックエンドサーバの数
    UAP 又はサービス内で発行した SQL 文に対して使用した，バックエンドサーバの数です。
11. 使用したバックエンドサーバの数の平均値
    UAP 又はサービス内で発行した SQL 文に対して使用した，バックエンドサーバの数の平均値です。
12. 出力情報の種別
    SQL の欄には，SQL の情報が表示されます。
    FSQL の欄は，確認する必要はありません。この欄には，0 又は****が表示されます。
13. システムが使用する情報（ユーザ未使用）
14. SQL 実行回数
    UAP 又はサービス内で発行した SQL 文の実行回数です。
15. SQL 実行時間（単位：マイクロ秒）
    UAP 又はサービス内で発行した SQL 文の処理時間の総和です。
16. 検索行数
    UAP 又はサービス内で発行した，FETCH 文及び SELECT 文での検索行数です。
17. 挿入行数
    UAP 又はサービス内で発行した，INSERT 文での挿入行数です。
18. 更新行数
    UAP 又はサービス内で発行した，UPDATE 文での更新行数です。
19. 削除行数
    UAP 又はサービス内で発行した，DELETE 文での削除行数です。
20. 作業表作成回数
    UAP 又はサービス内で発行した SQL 文によって，内部処理で作成した作業表の作成回数です。
21. 作業表削除回数
    UAP 又はサービス内で発行した SQL 文によって，内部処理で作成した作業表の削除回数。ただし，エラーが発生した場合は，作業表の削除回数が正しく表示されない場合があります。
22. 各項目の合計値
    合計値が表示されます。

23. 平均値（各項目の合計値÷実行回数）が表示されます。

24. 全体の各項目の合計値

注※

UAP の種類と出力されるサービス名称を次に示します。

UAP の種類		出力されるサービス名称
OpenTP1 の UAP	OpenTP1 の SUP（サービス利用プログラム）が SPP（サービス提供プログラム）に要求したサービス	該当するサービス名称
	TP1/Message Control が MHP（メッセージ処理プログラム）に要求したサービス	
	上記以外	連続した 31 個の＊（アスタリスク）
OpenTP1 以外の UAP		

14.3.5　グローバルバッファプールに関する統計情報

グローバルバッファプールに関する統計情報を次に示します。

参考

グローバルバッファプールに関する統計情報には，インメモリデータバッファの情報も出力されますが，各項目の出力値は概算値となります。

```
pdstedit VV-RR(Object Option) ***** GLOBAL BUFFER INFORMATION *****
INPUT          :c:¥tmp¥stjdata [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
--------------------------------------------------------------------------
HOST = test [3]
--------------------------------------------------------------------------
EDIT TIME 2003/04/03 22:00:00 - 2003/04/03 23:00:00 [4]
               [5]   [6]   [7]   [8]  [9]    [10]  [11]  [12] [13]  [14]
         *LN1* SYNCW MAXB  UPGET UPHIT(HIT)  UPFLS RFGET RFHIT(HIT) RFFLS
               ----- ----- ----- ----- ---   ----- ----- ----- ---  -----
               [15]  [16]  [17]  [18]  [19]  [20]  [21]  [22] [23]  [24]
         *LN2* READ  WRITE WAITR WAITW WAITL BFINS PRRED PRHIT(HIT) PRINS
               ----- ----- ----- ----- ----- ----- ----- ----- ---  -----
               [25]  [26]  [27]  [28]  [29]  [30]  [31]  [32]       [33]
         *LN3* GBHIT CURRF CURUP TRGUP SYNCC PRRDR LRDRC LWTRC      LBBKR
               ----- ----- ----- ----- ----- ----- ----- -----      -----
               [34]  [35]  [36]  [37]  [38]     [39]  [40]     [41]  [42]
         *LN4* LBBKW CINSM CFMAX CFAVG SLEPC    SLEPR SLEPA    SPINR SPINA
               ----- ----- ----- ----- -------- ----- -------- ----- -----
               [43]  [44]  [45]  [46]  [51]  [52]
         *LN5* SYNCL SYNCB ALTRW ALTUW BUFWT BUFWQ
               ----- ----- ----- ----- ----- -----
SERVER :  sds [47]
  *BUFFER NAME:bp01 [48]            BUFFER:   100 [49]
```

```
        *LN1*     0     3 2.19k 1.55k( 37)    264 2.04k 1.42k( 33)   702
        *LN2*   663   692     0     6     0     0     0     0(  0)     0
        *LN3*    70 1.60k     0   800     4     0     0     0          0
        *LN4*     0     0     0  ****  0.0e+00   0.0  *******   0.8  277
        *LN5*  178k     0     0     0    57     0
  *BUFFER NAME:bp02                 BUFFER:   100
        *LN1*     0     4 2.19k 1.56k( 37)    302 2.04k 1.44k( 34)   640
        *LN2*   638   692     0     5     0     0     0     0(  0)     0
        *LN3*    71 1.60k     0   800     4     0     0     0          0
        *LN4*     0     0     0  ****  2.0e+00   0.0  1.5e+00   1.0  356
        *LN5*  178k     0     0     0 3.07k     0
  *BUFFER NAME:bp03                 BUFFER:   100
        *LN1*     0     3 2.61k 2.38k( 24)      0 7.35k 7.16k( 72)   184
        *LN2*   372 1.15k     0     1    10     0     0     0(  0)     0
        *LN3*    96 1.60k     0   800     8     0     0     0          0
        *LN4*     0     0     0  ****  5.0e+00   0.0  1.2e+00   0.5  481
        *LN5*  259k     0     0     0 8.70k     1
  *TOTAL* [50]
        *LN1*     0     4 7.00k 5.49k( 30)    566 11.4k 10.0k( 54) 1.53k
        *LN2* 1.67k 2.54k     0    12    10     0     0     0(  0)     0
        *LN3*    84 4.79k     0 2.40k    16     0     0     0          0
        *LN4*     0     0     0  ****  7.0e+00   0.0  1.3e+00   0.7  380
        *LN5*  205k           0     0 11.8k     1
--------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                 LAST                          NUM
STJ         buf        2012/02/03 09:30:00   2012/02/03 09:36:19            48
--------------------------------------------------------------------------------

NO FILE KIND:LOG FILE NAME
  LOG KIND             FIRST                 LAST                          NUM
 1 STJ      :c:¥tmp¥stjdata¥pdstj01
  sys                  2012/02/03 09:30:49   2012/02/03 09:31:50            4
  uap                  2012/02/03 09:29:26   2012/02/03 09:31:04           26
  sql                  2012/02/03 09:29:25   2012/02/03 09:31:04          440
  sop                  2012/02/03 09:29:27   2012/02/03 09:31:04           98
  dop                  2012/02/03 09:29:27   2012/02/03 09:31:04           92
  pcd                  ****/**/** **:**:**   ****/**/** **:**:**            0
  obj                  ****/**/** **:**:**   ****/**/** **:**:**            0
  sqh                  ****/**/** **:**:**   ****/**/** **:**:**            0
  buf                  2012/02/03 09:30:00   2012/02/03 09:32:24           21
  fil                  2012/02/03 09:30:00   2012/02/03 09:32:24          224
  dfw                  2012/02/03 09:29:30   2012/02/03 09:32:24           71
  idx                  2012/02/03 09:30:00   2012/02/03 09:32:24            7
  fsv                  ****/**/** **:**:**   ****/**/** **:**:**            0
  hba                  ****/**/** **:**:**   ****/**/** **:**:**            0
 2 STJ      :c:¥tmp¥stjdata¥pdstj02
  sys                  2012/02/03 09:34:58   2012/02/03 09:37:00            6
  uap                  2012/02/03 09:34:00   2012/02/03 09:36:17           19
  sql                  2012/02/03 09:33:59   2012/02/03 09:36:17          456
  sop                  2012/02/03 09:34:02   2012/02/03 09:36:16           65
  dop                  2012/02/03 09:34:02   2012/02/03 09:36:16           96
  pcd                  ****/**/** **:**:**   ****/**/** **:**:**            0
  obj                  ****/**/** **:**:**   ****/**/** **:**:**            0
  sqh                  ****/**/** **:**:**   ****/**/** **:**:**            0
  buf                  2012/02/03 09:34:10   2012/02/03 09:36:19           27
  fil                  2012/02/03 09:34:10   2012/02/03 09:36:19          288
  dfw                  2012/02/03 09:34:04   2012/02/03 09:36:19           41
  idx                  2012/02/03 09:34:10   2012/02/03 09:36:19            9
```

```
fsv                 ****/**/** **:**:**   ****/**/** **:**:**           0
hba                 ****/**/** **:**:**   ****/**/** **:**:**           0
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時−出力終了日時）
3. グローバルバッファプールに関する統計情報の出力要求元ホスト名
4. 編集時間の範囲（取得開始時刻−取得終了時刻）
5. シンクポイント出力ページ数

   シンクポイント取得時にグローバルバッファプールから HiRDB ファイルに出力したページ数の最大値です。

   この値が大きい場合，該当するグローバルバッファプールに対しての更新ページの発生が多く，かつデファードライトトリガ時の更新ページ出力比率が低いために，出力されていないページがグローバルバッファプール上に多く残っていると考えられます。デファードライトトリガ時の更新ページ出力比率を高くできないか検討してください。
6. 最大同時要求バッファ面数

   グローバルバッファプールに対して，複数のトランザクションから同時に要求したグローバルバッファプールの取得要求回数です。

   この値がグローバルバッファ面数とほぼ同じで，かつバッファ不足発生回数が 0 でない場合，バッファが不足していると考えられます。
7. 更新 GET 回数

   グローバルバッファプールに対して，更新のためのバッファ取得を要求した回数です。
8. 更新バッファヒット回数

   グローバルバッファプールに対するバッファ取得の要求ページが，更新されたバッファとして見付かった回数です。
9. 更新バッファヒット率（単位：%）

   更新 GET 回数と参照 GET 回数の合計に対する更新バッファヒット回数の割合です。更新 GET 回数と参照 GET 回数に対して，目的のページが既に更新されたバッファとして物理的な入出力なしにグローバルバッファプール上で見付かった確率を示します。

   この値は，次の計算式から求めています。

   ```
   更新バッファヒット率＝
     （更新バッファヒット回数÷（参照GET回数＋更新GET回数））
     ×100
   ```
10. 更新バッファフラッシュ回数

    参照終了したバッファが更新されている場合，その内容を HiRDB ファイルへ出力してから新たなページを入力する空きバッファを作成するために，そのバッファを無効にした回数です。

この回数が多い場合，デファードライト処理の効果がないと考えられます。この場合，グローバルバッファプールの面数を大きくするか，又はデファードライトトリガ時の更新ページ出力比率を小さくする必要があります。

11. 参照 GET 回数

    グローバルバッファプールに対して，参照のためのバッファ取得を要求した回数です。

12. 参照バッファヒット回数

    グローバルバッファプールにするバッファ取得要求のページが，参照された（更新されていない）バッファとして見付かった回数です。

13. 参照バッファヒット率（単位：%）

    参照 GET 回数と更新 GET 回数の合計に対する参照バッファヒット回数の割合です。参照 GET 回数と更新 GET 回数に対して，目的のページが物理的な入出力なしにグローバルバッファプール上で見付かった確率を示します。

    この値は，次の計算式から求めています。

    ```
    参照バッファヒット率＝
      （参照バッファヒット回数÷（参照GET回数＋更新GET回数））
      ×100
    ```

14. 参照バッファフラッシュ回数

    参照終了したバッファがその後使用されていないので，新たなページを入力する空きバッファを作成するために，そのバッファを無効にした回数です。

15. 実 READ 回数

    HiRDB ファイルからページを入力した回数です。

16. 実 WRITE 回数

    HiRDB ファイルへページを出力した回数です。

17. 入力待ち発生回数

    バッファ上のページが，ほかのユーザによって HiRDB ファイルからの入力中であったため，待ち状態になった回数です。

18. 出力待ち発生回数

    バッファ上のページが，ほかのユーザによって HiRDB ファイルへの出力中であったため，待ち状態になった回数です。

19. バッファ排他待ち発生回数

    バッファ上のページが，ほかのユーザによって使用中であったため，待ち状態になった回数です。

20. バッファ不足発生回数

    バッファ（定義したバッファ面数）が，すべて参照中又は更新中で使用されているため，空きバッファのない状態が発生した回数です。

    この回数は，0 にする必要があります。

21. プリフェッチ入力ページ数

プリフェッチ対象としてメモリに入力したページ数です。

22. プリフェッチヒット回数
    プリフェッチ対象として入力したページがメモリ上でヒットした回数です。
23. プリフェッチヒット率（単位：%）
    プリフェッチ入力ページ数に対するプリフェッチヒット回数の割合です。
24. プリフェッチ実行資源不足発生回数
    プリフェッチ実行のための資源が不足した回数です。
25. グローバルバッファのヒット率（単位：%）
    グローバルバッファのヒット率は 80%を目安とします。80%以下の場合は，バッファ面数を増やす必要があります。
26. カレントの参照バッファ数
27. カレントの更新バッファ数
28. デファードライトトリガ時の出力契機となる更新バッファ数
29. シンクポイントの発生回数
30. プリフェッチの READ 要求回数
31. LOB 用グローバルバッファの READ 要求回数
32. LOB 用グローバルバッファの WRITE 要求回数
33. LOB 用グローバルバッファの一括入力ページ数
34. LOB 用グローバルバッファの一括出力ページ数
35. システムが使用する情報（ユーザ未使用）
36. システムが使用する情報（ユーザ未使用）
37. システムが使用する情報（ユーザ未使用）
38. グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他処理での，排他競合待ちとなった回数
39. グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他処理での，排他競合待ち発生率（単位：%）
40. スリープ処理実行での，グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他処理のスリープ処理実行回数の平均値
41. スピン処理実行での，グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他処理発生率（単位：%）
42. グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他処理中の，スピンループの実行回数の平均値
43. シンクポイント時のバッファプール排他占有時間（単位：マイクロ秒）
44. シンクポイント時のバッファプール排他占有時間内の処理バッファ数

45. シンクポイント時の参照要求ヒットによるデータベース書き込み処理代行回数
46. シンクポイント時の更新要求ヒットによるデータベース書き込み処理代行回数
47. サーバ名
    グローバルバッファプールに関する統計情報の出力要求元サーバです。
48. グローバルバッファプール名
    統計情報を編集したグローバルバッファプールの名称です。
49. グローバルバッファプールのバッファ面数
    該当するグローバルバッファプールのバッファ面数です。
50. 各項目の合計値
51. グローバルバッファの排他獲得処理での排他競合待ち時間（単位：マイクロ秒）※1※2
52. グローバルバッファの排他獲得処理での排他競合待ち行列数

注※1

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

注※2

pdstbegin コマンドに-w オプションを指定し，スレッド間ロック待ち時間を取得している場合に出力します。pdstbegin コマンドの-w オプションについては，「pdstbegin（統計情報の出力開始）のオプション」を参照してください。

14.3.6　データベース操作に関する HiRDB ファイルの統計情報

データベース操作に関する HiRDB ファイルの統計情報を次に示します。

```
pdstedit VV-RR          ***** HiRDB FILE INFORMATION *****
INPUT          :c:¥tmp¥pdstj1 [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
--------------------------------------------------------------------------------
HOST = test [3]
--------------------------------------------------------------------------------
EDIT TIME 1996/10/30 00:00:00 - 1996/10/31 00:00:00 [4]
              [5]    [6]    [7]    [8]    [9]    [10]   [11]   [12]
         *LN1* SYNC-R SYNC-W AIO-R  AIO-W  LISTIO OPEN   CLOSE  IOERR
               -----  -----  -----  -----  -----  -----  -----  -----
SERVER : sds [13]
 *FILE NAME : C:¥hirdb¥iospt0¥h001 [14]
   *RDAREA NAME :RDMASTER [15]
         *LN1*    67    142      0     85     85      8      7      0
 *FILE NAME : C:¥hirdb¥iospt0¥h002
   *RDAREA NAME :RDDIR01
         *LN1*    67    116      0     81     81      8      7      0
 *FILE NAME : C:¥hirdb¥iospt0¥h003
   *RDAREA NAME :RDDIC01
```

```
        *LN1*   824  2.15k      0  2.00k  2.00k     19     19      0
 *FILE NAME : C:¥hirdb¥iospt1¥u001
   *RDAREA NAME :RU01
        *LN1*   256    483      0    401    401     23     22      0
 *FILE NAME : C:¥hirdb¥iospt2¥u002
   *RDAREA NAME :RU02
        *LN1*   261    501      0    416    416     26     25      0
 *FILE NAME : C:¥hirdb¥iospt3¥u003
   *RDAREA NAME :RU03
        *LN1*   254    467      0    384    384     23     22      0
 *FILE NAME : C:¥hirdb¥iospt4¥u001
   *RDAREA NAME :RULOB01
        *LN1*    16     43      0     33     33      4      4      0
 *FILE NAME : C:¥hirdb¥iospt4¥u002
   *RDAREA NAME :RULOB02
        *LN1*    16     43      0     33     33      4      4      0
 *FILE NAME : C:¥hirdb¥iospt4¥u011
   *RDAREA NAME :RUBLOB01
        *LN1*    30     59      0      0      0      7      7      0
 *FILE NAME : C:¥hirdb¥iospt4¥u012
   *RDAREA NAME :RUBLOB02
        *LN1*     0      0      0      0      0      0      0      0
 *TOTAL* [16]
        *LN1* 1.79k  4.00k      0  3.43k  3.43k    122    117      0
--------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                 LAST                          NUM
STJ         fil        1996/10/30 10:38:19   1996/10/30 12:31:45           440
--------------------------------------------------------------------------------
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時−出力終了日時）
3. データベース操作に関する HiRDB ファイルの統計情報の出力要求元ホスト名
4. 編集時間の範囲（取得開始時刻−取得終了時刻）
5. 同期 READ 回数

   HiRDB ファイルから同期モードでページを入力した回数です。
6. 同期 WRITE 回数

   HiRDB ファイルへ同期モードでページを出力した回数です。
7. 非同期 READ 回数

   非同期 READ 機能を使用した場合に，HiRDB ファイルから非同期モードでページを入力した回数です。
8. 非同期 WRITE 回数

   HiRDB ファイルへ非同期モードでページを出力した回数です。

   この回数は，次の回数を合計した値です。
   - デファードライトトリガ時に HiRDB ファイルへ出力した回数
   - シンクポイント時に HiRDB ファイルへ出力した回数

9. システムが使用する情報（ユーザ未使用）

10. オープン発行回数

    HiRDB ファイルのオープン処理をした回数です。

11. クローズ発行回数

    HiRDB ファイルのクローズ処理をした回数です。

12. 入出力エラー発生回数

    HiRDB ファイルの入出力時に発生したエラーの回数です。

13. サーバ名

    データベース操作に関する HiRDB ファイルの統計情報の出力要求元サーバです。

14. HiRDB ファイル名※

    統計情報を編集した HiRDB ファイルの名称です。

15. RD エリア名

    該当する HiRDB ファイルを使用する RD エリアの名称です。

16. 各項目の合計値

注※

15 に示す RD エリアのオープン属性が INITIAL 以外で，一度もこの RD エリアにアクセスしていない場合は空白になるときがあります。空白になっているレコードは無視してください。なお，RD エリアを構成する HiRDB ファイルはデータベース状態解析ユティリティで-k phys -f を指定すると確認できます。

14.3.7　デファードライト処理に関する統計情報

デファードライト処理に関する統計情報を次に示します。

```
pdstedit VV-RR(Object Option) ***** DEFERRED WRITE INFORMATION *****
INPUT          :c:¥tmp¥pdstj1 [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
--------------------------------------------------------------------------------
HOST = test [3]
--------------------------------------------------------------------------------
EDIT TIME 2004/03/11 10:00:00 - 2004/03/11 12:00:00 [4]
 [5]       [6]         [7]   [8]   [9]   [10]  [11]        [12]  [13]
 SERVER    EXEC        TRG   PRSYC SYNC  DBSYC RDSYC       END   CANCL
           -----       ----- ----- ----- ----- -----       ----- -----
                       [14]  [15]  [16]        [17]  [18]
                       PMAX  PMIN  PAVG        TOTAL AVG
                       ----- ----- -----       ----- -----
 sds         103          15    44    44     0     0          32    12
                           1     1     1       3.56k    35
--------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                 LAST                          NUM
STJ         dfw        2004/03/11 10:15:08   2004/03/11 12:33:31           103
--------------------------------------------------------------------------------
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時−出力終了日時）
3. デファードライト処理に関する統計情報の出力要求元ホスト名
4. 編集時間の範囲（取得開始時刻−取得終了時刻）
5. デファードライト処理に関する統計情報の出力要求元サーバ名
6. デファードライト要求回数

   集計した時間内にデファードライト（遅延書き込み処理）をした回数です。
7. トリガ回数

   集計した時間内に，グローバルバッファ内の更新ページ比率（総バッファ面数に占める更新バッファ数の比率）が，ある一定値に達した場合にデファードライト処理を要求した回数です。
8. プレシンク回数

   集計した時間内に，シンクポイントダンプ出力間隔に達する前の，ある一定値に達した場合（次のシンクポイントまでにシステムログファイルへの出力回数が，ある一定値に達した場合）にデファードライト処理を要求した回数です。
9. シンクポイント発生回数

   集計した時間内に発生した DB シンクポイント回数（データベース整合性保証のため同期点を取った回数）です。
10. データベースのシンクポイント発生回数

    シンクポイント処理を有効化できないシンクポイント契機で，更新バッファをデータベースに反映する処理の回数です。
11. RD エリアのシンクポイント発生回数

    RD エリアに対応するすべての更新ページを，ディスクへ出力する処理の回数です。
12. プレシンク完了状態（出力回数）

    集計した時間内に，シンクポイントダンプ出力間隔に達する前の，ある一定値に達した場合（次のシンクポイントまでにシステムログファイルへの出力回数が，ある一定値に達した場合）にデファードライト処理を要求したとき，シンクポイントに達するまでにバッファ上のすべての更新バッファを HiRDB ファイルに出力できた回数です。
13. プレシンク完了状態（キャンセル回数）

    集計した時間内に，シンクポイントダンプ出力間隔に達する前の，ある一定値に達した場合（次のシンクポイントまでにシステムログファイルへの出力回数が，ある一定値に達した場合）にデファードライト処理を要求したとき，すべての更新バッファを HiRDB ファイルに出力できないままシンクポイント処理が発生したため，途中でキャンセルした回数です。
14. ディスクボリューム単位の並列度（最大値）

集計した時間内に発生したデファードライト処理によって更新があったディスクボリューム（HiRDB ファイルシステム領域）数の最大値です。ただし，NTFS に HiRDB ファイルシステム領域を作成している場合は，HiRDB ファイルシステム領域が複数でもまとめて 1 としてカウントされます。

最大値及び最小値が 1 の場合は，特定のディスクボリューム（HiRDB ファイルシステム領域）にだけ出力が集中していることを意味します。そのため，更新が発生する RD エリアを作成するディスクボリューム（HiRDB ファイルシステム領域）を分けることをお勧めします。また，デファードライト処理の並列 WRITE 機能使用時は最大値が pd_dfw_awt_process 指定値以上がよいです。

15. ディスクボリューム単位の並列度（最小値）

    集計した時間内に発生したデファードライト処理によって更新があったディスクボリューム（HiRDB ファイルシステム領域）数の最小値です。

    最大値及び最小値が 1 の場合は，特定のディスクボリューム（HiRDB ファイルシステム領域）にだけ入出力が集中しているので，更新が発生する一つ又は複数の RD エリアを作成するディスクボリューム（HiRDB ファイルシステム領域）を分けることをお勧めします。

16. ディスクボリューム単位の並列度（平均値）

    集計した時間内に発生したデファードライト処理によって更新があったディスクボリューム（HiRDB ファイルシステム領域）数の平均値です。

17. 出力ページ数（合計）

    集計した時間内にバックグラウンドで動作するデファードライト（遅延書き込み処理）によって，HiRDB ファイルへ出力したページ数の合計値です。

18. 出力ページ数（平均）

    集計した時間内にバックグラウンドで動作するデファードライト（遅延書き込み処理）によって，HiRDB ファイルへ出力したページ数の平均値です。

14.3.8　インデクスに関する統計情報

インデクスに関する統計情報を次に示します。

```
pdstedit VV-RR          ***** INDEX INFORMATION *****
INPUT          :c:¥tmp¥pdstj1 [1]
OUTPUT RANGE   :**/**/** **:**:** - **/**/** **:**:** [2]
CONTROL FILE     :c:¥hirdb¥pdstedit¥CONTROL1 [18]
                                    [19]
FILE GROUP     :SERV1  : FILE1 FILE2 FILE3 FILE4 FILE5
                         FILE6 FILE7 FILE8 FILE9 FILE10
--------------------------------------------------------------------------------
EDIT TIME 1996/10/30 00:00:00 - 1996/10/31 00:00:00 [3]
             [4]   [5]   [6]   [7]   [8]   [9]   [10]  [11]
    *LN1*    C_SLK W_SLK XLOCK DEADL UNQCK UNQER REPOS REPG
             ----- ----- ----- ----- ----- ----- ----- -----
             [12]  [13] [14]
    *LN2*    SP_NM LVL ( NUM )
             ----- ---  -----
SERVER : sds [15]
 *INDEX_ID :     196611 [16]
```

```
  *RDID    :           3 [17]
    *LN1*    4.32k     0     0     0     0     0     0     0
    *LN2*        0
 *INDEX_ID :     196612
  *RDID    :           3
    *LN1*    5.31k     0     0     0     0     0     0     0
    *LN2*        0
 *INDEX_ID :     196613
  *RDID    :           3
    *LN1*    4.64k     0 1.35k     0   684     0     0     0
    *LN2*        0
 *INDEX_ID :     196614
  *RDID    :           3
    *LN1*       12     0 2.03k     0   684     0     0     0
    *LN2*        0
 *INDEX_ID :     196615
  *RDID    :           3
    *LN1*    4.86k     0 1.94k     0     0     0     0     0
    *LN2*        0
--------------------------------------------------------------------------------
FILE KIND   LOG KIND   FIRST                 LAST                          NUM
STJ         idx        1996/10/30 10:38:19   1996/10/30 12:31:45            44
FJ          idx        ****/**/** **:**:**   ****/**/** **:**:**             0
--------------------------------------------------------------------------------
NO FILE KIND:LOG FILE NAME
  LOG KIND             FIRST                 LAST                          NUM
 1 STJ      :c:¥tmp¥pdstj1
  sys                  1996/10/30 10:37:33   1996/10/30 12:33:07           112
  uap                  1996/10/30 10:36:38   1996/10/30 12:34:04          1854
  sql                  1996/10/30 10:36:37   1996/10/30 12:34:03         50971
  sop                  ****/**/** **:**:**   ****/**/** **:**:**             0
  dop                  ****/**/** **:**:**   ****/**/** **:**:**             0
  pcd                  ****/**/** **:**:**   ****/**/** **:**:**             0
  buf                  1996/10/30 10:38:19   1996/10/30 12:31:45            44
  fil                  1996/10/30 10:38:19   1996/10/30 12:31:45           440
  dfw                  1996/10/30 10:38:12   1996/10/30 12:33:31           103
  idx                  1996/10/30 10:38:19   1996/10/30 12:31:45            44
```

[説明]

1. 統計入力アンロードファイル名，又は統計入力アンロードファイル格納ディレクトリ名（最大 58 バイト）
2. 出力指定日時（出力開始日時−出力終了日時）
3. 編集時間の範囲（取得開始時刻−取得終了時刻）
4. CHECK モード・キー値参照排他発行回数

   CHECK モード（ほかのユーザが排他を取得している場合は待たないモード）で，キー値に対して参照の排他（PR）を取得した回数です。
5. WAIT モード・キー値参照排他発行回数

   WAIT モード（ほかのユーザが排他を取得している場合は待つモード）で，キー値に対して参照の排他（PR）を取得した回数です。
6. キー値更新排他発行回数

キー値に対して更新の排他（EX）を取得した回数です。

7. キー値排他のデッドロック回数

   キー値に対して更新の排他（EX）を取得したことによって発生した，デッドロックの回数です。

8. ユニークチェック回数

   INSERT 文や UPDATE 文によって，値の一意性をチェックする処理をした回数です。なお，この値はユニークインデクスの場合にだけ統計情報が取得されます。

9. ユニークエラー発生回数

   INSERT 文や UPDATE 文によって，値の一意性をチェックする処理をした結果，既に同じ値を持ったキー値があるため，エラーとなった回数です。なお，この値はユニークインデクスの場合にだけ統計情報が取得されます。

   この回数が多い場合は，ユニークインデクスが必要かどうか再検討する必要があります。

10. リポジショニング（カレントポジション）回数

    インデクスページ内のキー値を参照している間に，ほかのユーザによって同じページ内にほかの異なるキー値が追加されたり，削除されたりした場合に，参照しているキー値のページ内の位置が変わってしまったときのリポジショニング（補正処理）をした回数です。

11. リポジショニング（カレントページ）回数

    インデクス内でキー値を探している途中に，目的のキー値があるページでほかのユーザによってスプリットが発生した場合，探しているキー値があるページが変わってしまったときにリポジショニング（補正処理）をした回数です。

12. スプリット（インデクス分割処理）回数※2

    スプリット（インデクス分割処理）※1 した回数です。

    この回数が多い場合は，インデクスの PCTFREE の比率を高くする必要があります。また，大量にデータを追加する場合は，データベース作成ユティリティ（pdload）を使用した方がよいです。

13. スプリットが波及した段数※2

14. スプリットが波及した段数別スプリット回数

    スプリット発生によって，スプリット処理が波及した上位ページへの段数ごとのスプリット回数の合計です。

    波及した段数が大きいところでスプリット回数が多い場合は，インデクスページの大きさに比べてキー値が大きいか，又はインデクスの PCTFREE の比率が高過ぎると考えられます。

15. サーバ名

    インデクスに関する統計情報の出力要求元サーバです。

16. インデクス ID

    統計情報を編集したインデクスのインデクス番号です。

    出力されたインデクス番号を基にディクショナリ表（SQL_INDEXES 表）を検索すれば，インデクス名称を知ることができます。

17. RD エリア ID

統計情報を編集したインデクスを格納している RD エリアの RD エリア番号です。

出力された RD エリア番号を基にディクショナリ表（SQL_RDAREAS 表）を検索すれば，RD エリア名称を知ることができます。

18. 制御文ファイル名（最大 58 バイト）

-d オプションで指定した制御文ファイルの名称です。

19. サーバ名及びファイルグループ名

-d オプションで指定した制御文ファイル中のサーバ名（解析対象※3 のサーバ名を表示），及び対応するファイルグループ名です。

注 1

13，14 は，スプリットが発生した最大インデクス段数分繰り返し表示されます。

注 2

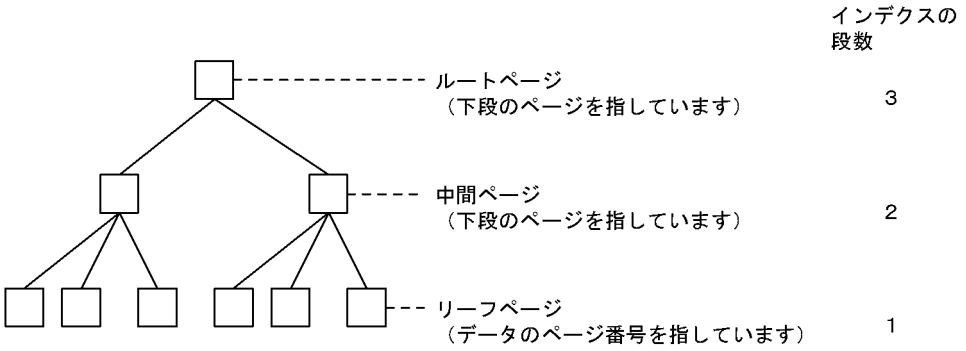
12～14 は，アンロードログファイル，又は-d オプション指定でシステムログファイルを入力したときだけ表示されます。

注 3

4～11 は，シンクポイント間で回数をカウントし，各シンクポイント間の合計値を表示します。シンクポイント間でカウントされる回数の最大値は 65535 のため，実際は表示されている数よりも多く発生している可能性があります。

注 4

スプリットが波及した段数について説明します。



［説明］

最初にスプリットが発生するのは，リーフページです。リーフページでスプリットが発生した場合，スプリットによって新しいページが作成されるので，上位ページ（中間ページ）にその新ページを登録する処理がされ，スプリット処理が完了するとします。そうすると，この場合のスプリットが波及した段数は 2（LVL）となり，スプリットが波及した段数を 2 として回数（NUM）をカウントします。

スプリットの発生によって，インデクスの段数が 2 の上位ページに対してスプリットが波及した場合，更に新ページを上位ページに登録しようとしたときに上位ページに空きがなければ，その上位ページに対してスプリットが発生することになります。そうすると，この場合のスプリットが波及した段数は 3（LVL）となり，スプリットが波及した段数を 3 として回数（NUM）をカウントします。

注※1

スプリットとは，キー値を追加しようとしたときにそのインデクスを格納する RD エリア内の使用中ページに空きがなかったため，空き領域を確保するためにその使用中ページの内容を分割し，後ろ半分を未使用ページへ追加する処理のことをいいます。

注※2

システムログを入力したときだけ出力されます。

注※3

次に示すサーバは解析対象になります。

- -s オプションで指定したサーバ名と制御文ファイル中のサーバ名が同じ場合のサーバ
- -s オプションを指定していない場合の制御文ファイルの全サーバ

14.4　DAT 形式ファイルへの統計情報出力

DAT 形式ファイルへ出力した統計情報は，ほかの表計算ソフトの入力ファイルとして利用できます。

14.4.1　DAT 形式ファイルのデータ格納形式

DAT 形式ファイルへ出力した統計情報は，次の表のように編集し，格納されます。なお，DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。

表 14–6　DAT 形式ファイルのデータ編集形式

データ形式	格納方法
オーバフローデータ	連続した 4 個のアスタリスク（*）※を格納します。
編集できないデータ	連続した 10 個のアスタリスク（*）※を格納します。

注※

-e オプションに er1 を指定した場合，1 個のアスタリスク（*）を格納します。

14.4.2　DAT 形式ファイルの単位

DAT 形式ファイルへ統計情報を出力する場合，統計情報ごとにファイルを作成します。ファイルはユーザが指定したディレクトリ下に作成します。DAT 形式ファイルのファイル名称一覧を次の表に示します。

表 14–7　DAT 形式ファイルのファイル名称一覧

編集項目	出力単位	ファイル名称
システムの稼働に関する統計情報	ログ内 1 システム稼働情報	sys_DAT
UAP に関する統計情報	1UAP 又は 1 サービス実行単位	uap_DAT
SQL に関する統計情報	1SQL 実行単位	sql_DAT
グローバルバッファに関する統計情報	シンクポイントごと	buf_DAT
データベース操作に関する HiRDB ファイルの統計情報	シンクポイントごと	fil_DAT
デファードライト処理に関する統計情報	デファードライト処理用並列 WRITE プロセス起動ごと	dfw_DAT
静的最適化に関する統計情報	1SQL 前処理単位	sop_DAT
動的最適化に関する統計情報	1SQL 内の 1 割り当てサーバ単位	dop_DAT
SQL オブジェクト実行に関する統計情報	1SQL 並列単位	pcd_DAT

編集項目	出力単位	ファイル名称
SQL オブジェクト転送に関する統計情報	1SQL 並列単位	obj_DAT
SQL 文の履歴に関する統計情報	1SQL 実行単位	sqh_DAT
CONNECT/DISCONNECT に関する統計情報	CONNECT/DISCONNECT 単位	cnc_DAT
データベースの入出力に関する統計情報	情報出力時間間隔ごとに入出力があった RD エリアを構成する HiRDB ファイル単位	dio_DAT

注 1

各ファイルはファイル名が固定のため，ユーザ指定のディレクトリ下に既に同一名称のファイルがある場合は，上書きされる可能性があります。上書きを防ぐためには，既にあるファイル名称を変更するか，又は既にファイルを出力したディレクトリを指定しないように注意する必要があります。

注 2

DAT 形式ファイルには，2 ギガバイトを超えるデータは格納できません。大量のシステムログファイルを使用する場合には，オプションの指定などで出力するデータを絞り込み，出力データが 2 ギガバイトを超えないようにしてください。

注 3

システムの稼働に関する統計情報は列が多いため，そのままでは Excel への入力ができません。そのため，Excel のテキストファイルウィザードで不要な列を削除してから入力するようにしてください。

14.4.3 DAT 形式ファイルのレコード形式

DAT 形式ファイルのレコード形式を，次の表に示します。

- 表「DAT 形式ファイルのレコード形式（システムの稼働に関する統計情報（その 1））」
- 表「DAT 形式ファイルのレコード形式（システムの稼働に関する統計情報（その 2））」
- 表「DAT 形式ファイルのレコード形式（システムの稼働に関する統計情報（その 3））」
- 表「DAT 形式ファイルのレコード形式（UAP に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（SQL に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（グローバルバッファプールに関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（データベース操作に関する HiRDB ファイルの統計情報）」
- 表「DAT 形式ファイルのレコード形式（デファードライト処理に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（SQL 静的最適化に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（SQL 動的最適化に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（SQL オブジェクト実行に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（SQL オブジェクト転送に関する統計情報）」

- 表「DAT 形式ファイルのレコード形式（SQL 文の履歴に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（CONNECT/DISCONNECT に関する統計情報）」
- 表「DAT 形式ファイルのレコード形式（データベースの入出力に関する統計情報）」

表の見方についての注意事項を次に示します。

- 表中のタイトルバーは-b オプションを指定した場合に出力され，システムの稼働に関する統計情報のときは，ファイルの先頭 3 行に出力されます。それ以外の統計情報の場合は，ファイルの先頭 1 行に出力されます。
- 各フィールドのタイトルバーの長さは，表中に示す最大長の値には入っていません。そのため，タイトルバーを含めて DAT 形式ファイルのデータを処理する場合は注意してください。
- 備考の−は，特に注意する点はないことを示しています。
- 最大長は，属性が文字列の場合，文字列データを囲む""は最大長の値に入っていません。また，属性が数値の項目で，編集できないデータを検知した場合は，**********を出力します。この場合，表中の最大長（3，5，又は 6 バイト）の値を超えて，10 個の「*」を出力します。

表 14–8　DAT 形式ファイルのレコード形式（システムの稼働に関する統計情報（その 1））

項番	フィールド名（タイトルバー）			属性	最大長	備考
1	ホスト名(HOST)			文字	32	−
2	サーバ名(SERVER)				8	全体の統計情報の場合は,’＊＊＊＊＊＊＊＊’
3	編集開始時刻(START)				19	"MM/DD/hh:mm"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
4	スケジュール情報(SCHEDULE)	スケジュール待ち行列数(QUEUE LEN(SCH))	発生件数(FREQ)	数値	10	−
5			最大値(MAX)			
6			最小値(MIN)			
7			平均値(AVG)			
8		スケジュールメッセージ長(MESSAGE LEN)	発生件数(FREQ)			
9			最大値(MAX)			単位：バイト
10			最小値(MIN)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
11			平均値(AVG)			
12	プロセス情報(PROCESS)	サーバの異常終了回数(SERVER ABORT)	発生件数(FREQ)	数値	10	項番 174～177 にプロセス情報の続きがあります。
13		HiRDB が使用する内部的なサーバの異常終了回数(SYSTEM SERVER ABORT)	発生件数(FREQ)			－
14		プロセス個数(PROCESS COUNT)	最大値(MAX)			
15			最小値(MIN)			
16			平均値(AVG)			
17	トランザクション情報(TRANSACTION)	コミット回数(COMMIT)	発生件数(FREQ)	数値	10	－
18		ロールバック回数(ROLLBACK)	発生件数(FREQ)			
19	ネームサーバ情報(NAME)	システム固有情報(CACHE HIT)	発生件数(FREQ)	数値	10	－
20		システム固有情報(LOCAL HIT)	発生件数(FREQ)			
21		システム固有情報(LOOK UP)	発生件数(FREQ)			
22	RPC 情報(RPC)	システム固有情報(RESPONSE TIME)	発生回数(FREQ)	数値	10	ユーザが使用する RPC 情報は項番 178～199，252～259 にあります。
23			最大値(MAX)			－
24			最小値(MIN)			
25			平均値(AVG)			
26		システム固有情報(USER SERVICE TIME)	発生回数(FREQ)			
27			最大値(MAX)			
28			最小値(MIN)			
29			平均値(AVG)			
30		システム固有情報(TIME OUT)	発生回数(FREQ)			
31		システム固有情報(ERROR)	発生回数(FREQ)			
32		システム固有情報(SEND)	発生件数(FREQ)			
33		システム固有情報(RECEIVE)	発生件数(FREQ)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
34	排他制御情報(LOCK)	排他待ち時間(WAIT TIME)※3	発生件数(FREQ)	数値	10	排他制御情報は項番268～271，281～285に続きがあります。
35			最大値(MAX)			単位：ミリ秒
36			最小値(MIN)			
37			平均値(AVG)			
38		排他待ち数(ユーザ数) (QUEUE LEN)	発生件数(FREQ)			－
39			最大値(MAX)			
40			最小値(MIN)			
41			平均値(AVG)			
42		デッドロック件数(DEADLOCK)	発生件数(FREQ)			
43		排他制御用プールパーティションの使用率 (USE LOCK TABLE)※1	発生件数(FREQ)			
44			最大値(MAX)		3	単位：%
45			最小値(MIN)			
46			平均値(AVG)			
47	共用メモリ情報 (SHARED MEMORY)	システム固有情報(STATIC GET SIZE)	発生件数(FREQ)	数値	10	－
48			最大値(MAX)			単位：バイト
49			最小値(MIN)			
50			平均値(AVG)			
51		システム固有情報(STATIC POOL SIZE)	発生件数(FREQ)			－
52			最大値(MAX)			単位：バイト
53			最小値(MIN)			
54			平均値(AVG)			
55		システム固有情報(DYNAMIC GET SIZE)	発生件数(FREQ)			－
56			最大値(MAX)			単位：バイト
57			最小値(MIN)			
58			平均値(AVG)			
59		システム固有情報(DYNAMIC POOL SIZE)	発生件数(FREQ)			－

項番	フィールド名（タイトルバー）			属性	最大長	備考
60			最大値(MAX)			単位：バイト
61			最小値(MIN)			
62			平均値(AVG)			
63		サーバ用，及び HiRDB が使用する内部的なサーバ用に確保した共用メモリのサイズ(SEGMENT SIZE)	平均値(AVG)			単位：キロバイト
64		静的共用メモリ確保サイズ(STATIC SIZE)	平均値(AVG)			
65		動的共用メモリ確保サイズ(DYNAMIC SIZE)	平均値(AVG)			
66		グローバルバッファ用共用メモリ確保サイズ(SIZE FOR BUFFER)	平均値(AVG)			
67	シンクポイント情報(SYNC POINT)	シンクポイントダンプ取得間隔時間(GET INTERVAL TIME)※3	発生件数(FREQ)	数値	10	－
68			最大値(MAX)			単位：ミリ秒
69			最小値(MIN)			
70			平均値(AVG)			
71		シンクポイントダンプ取得時間(GET TIME)※3	発生件数(FREQ)			－
72			最大値(MAX)			単位：ミリ秒
73			最小値(MIN)			
74			平均値(AVG)			
75	ログ情報(LOG)	バッファ満杯回数(BUFFER FULL)	発生件数(FREQ)	数値	10	項番 272～280，286～294 にログ情報の続きがあります。
76		カレントバッファなしによる待ち回数(WAIT THREAD)	発生件数(FREQ)			
77		出力ブロック長(OUTPUT BLOCK LEN)	発生件数(FREQ)			－
78			最大値(MAX)			単位：バイト
79			最小値(MIN)			
80			平均値(AVG)			
81		非バス部分のデータ長(NOT BUS LEN)	発生件数(FREQ)			－
82			最大値(MAX)			単位：バイト
83			最小値(MIN)			
84			平均値(AVG)			
85		入出力待ちバッファ面数(WAIT BUFFER FOR IO)	発生件数(FREQ)			－
86			最大値(MAX)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
87			最小値(MIN)			
88			平均値(AVG)			
89		ファイルへの書き込み回数(WRITE TO FILE)※2	発生件数(FREQ)			－
90		書き込みエラー回数(WRITE ERROR)	発生件数(FREQ)			－
91		ログファイルスワップ時間(LOG FILE SWAP TIME)※3	発生件数(FREQ)			－
92			最大値(MAX)			単位：ミリ秒
93			最小値(MIN)			
94			平均値(AVG)			
95		ログ入力データ長(LOG INPUT DATA)	発生件数(FREQ)			－
96			最大値(MAX)			単位：バイト
97			最小値(MIN)			
98			平均値(AVG)			
99		ファイルからの読み出し回数(READ FROM FILE)	発生件数(FREQ)			－
100		読み出しエラー回数(READ ERROR)	発生件数(FREQ)			－

注※1

排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition オペランド又は pd_fes_lck_pool_partition オペランドに 2 以上を指定した場合）は，排他制御用プールパーティションごとに求めた使用率を集計して，最大，最小，平均値を出力します。

注※2

システムログのファイル出力以外にも，システムログファイルの状態を変更するときの書き込み回数が加算されます。システムログファイルを二重化している場合には，系ごとの書き込み回数が加算されます。また，システムログブロックの出力回数だけでなく，システムログファイルのスワップなどでファイルの状態が変化する場合に発生する，ファイルの書き込み回数も含まれます。

注※3

求めた時間が OS の時刻戻しなどによって負の値となる場合は，値が 0 の統計情報として出力します。そのため，表示する値は実際の値より小さいことがあります。

表 14–9　DAT 形式ファイルのレコード形式（システムの稼働に関する統計情報（その 2））

項番	フィールド名（タイトルバー）			属性	最大長	備考
101	ディクショナリ情報(DICTIONARY)	表定義情報取得要求回数(TBL-DEF GET REQ)※1	発生件数(FREQ)	数値	10	項番 200～251，260～

項番	フィールド名（タイトルバー）			属性	最大長	備考
						267 にディクショナリ情報の続きがあります。
102		表定義情報用バッファヒット回数(TABLE CACHE HIT)※1	発生件数(FREQ)			－
103		表定義情報用バッファ中の定義情報数(CASHED TBL-DEF)※1	発生件数(FREQ)			－
104			最大値(MAX)			
105			最小値(MIN)			
106			平均値(AVG)			
107		表定義情報用バッファに取得した 1 表定義情報当たりの表定義情報用バッファ使用領域長(USE TBL-DEF SIZE)※1	発生件数(FREQ)			－
108			最大値(MAX)			単位：バイト
109			最小値(MIN)			
110			平均値(AVG)			
111		表定義情報用バッファ使用領域長(CACHED TBL-DEF SIZE)※1	発生件数(FREQ)			－
112			最大値(MAX)			単位：バイト
113			最小値(MIN)			
114			平均値(AVG)			
115		表アクセス権限情報取得回数(ACCESS PRIV CHECK)	発生件数(FREQ)			－
116		表アクセス権限情報用バッファヒット回数(CACHE HIT(ACCESS))	発生件数(FREQ)			－
117		ユーザ権限情報取得要求(CON/DBA DEF GET REQ)	発生件数(FREQ)			－
118		ユーザ権限情報用バッファヒット回数(CON/DBA CACHE HIT)	発生件数(FREQ)			－
119		ユーザ権限情報用バッファ使用ユーザ数(CON/DBA CACHED USER)	発生件数(FREQ)			－
120			最大値(MAX)			
121			最小値(MIN)			
122			平均値(AVG)			
123		ディクショナリサーバとの通信長(DIC TRANS DATA LEN)	発生件数(FREQ)			－
124			最大値(MAX)			単位：バイト
125			最小値(MIN)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
126			平均値(AVG)			
127		ディクショナリサーバとの通信回数(TRANS)	発生件数(FREQ)			－
128		ビュー解析情報取得要求回数(VIEW DEF GET REQ)	発生件数(FREQ)			－
129		ビュー解析情報用バッファヒット回数(VIEW DEF CACHE HIT)	発生件数(FREQ)			－
130		ビュー解析情報用バッファ中の解析情報数(VIEW CACHED DEF)	発生件数(FREQ)			－
131		ビュー解析情報用バッファに取得した 1 ビュー解析情報当たりのビュー解析情報用バッファ使用領域長(USED VIEW SIZE)	発生件数(FREQ)			－
132			最大値(MAX)			単位：バイト
133			最小値(MIN)			
134			平均値(AVG)			
135		使用したビュー解析情報用バッファ長(VIEW CACHE SIZE)	発生件数(FREQ)			－
136			最大値(MAX)			単位：バイト
137			最小値(MIN)			
138			平均値(AVG)			
139		バッファミスとなったビュー解析情報長(CACHE MISS VIEW SIZE)	発生件数(FREQ)			－
140			最大値(MAX)			単位：バイト
141			最小値(MIN)			
142			平均値(AVG)			
143	シングルサーバ，フロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバの SQL オブジェクト情報(FES-BES-DIC(SDS) INFORMATION)	SQL オブジェクト取得要求回数(SQLOBJ GET REQ)	発生件数(FREQ)	数値	10	－
144		SQL オブジェクト用バッファヒット回数(SQLOBJ CACHE HIT)	発生件数(FREQ)			－
145		SQL オブジェクト用バッファ中の SQL オブジェクト数(CACHED SQLOBJ)	発生件数(FREQ)			－
146			最大値(MAX)			
147			最小値(MIN)			
148			平均値(AVG)			
149		SQL オブジェクト用バッファ中の SQL オブジェクトの合計長(CACHED SQLOBJ SIZE)	発生件数(FREQ)			－
150			最大値(MAX)			単位：キロバイト
151			最小値(MIN)			
152			平均値(AVG)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
153		SQL オブジェクト用バッファから出された SQL オブジェクトの数(SWAP OUT SQLOBJ)	発生件数(FREQ)			−
154		SQL オブジェクト長(SQLOBJ LEN)	発生件数(FREQ)			−
155			最大値(MAX)			単位：バイト
156			最小値(MIN)			
157			平均値(AVG)			
158		ストアドプロシジャのオブジェクトの取得要求回数(STROBJ GET REQ)	発生件数(FREQ)			−
159		ストアドプロシジャのオブジェクトの SQL オブジェクト用バッファヒット回数(STROBJ CACHE HIT)	発生件数(FREQ)			−
160		SQL オブジェクト用バッファ中の，ストアドプロシジャのオブジェクト数(CACHED STROBJ)	発生件数(FREQ)			−
161			最大値(MAX)			
162			最小値(MIN)			
163			平均値(AVG)			
164		SQL オブジェクト用バッファ中の，ストアドプロシジャのオブジェクトの合計長(CACHED STROBJ SIZE)	発生件数(FREQ)			−
165			最大値(MAX)			単位：キロバイト
166			最小値(MIN)			
167			平均値(AVG)			
168		SQL オブジェクト用バッファから出されたストアドプロシジャのオブジェクトの数(SWAP OUT STROBJ)	発生件数(FREQ)			−
169		ストアドプロシジャのオブジェクト長(STROBJ LEN)	発生件数(FREQ)			−
170			最大値(MAX)			単位：バイト
171			最小値(MIN)			
172			平均値(AVG)			
173		ストアドプロシジャのオブジェクトのリコンパイル回数(STROBJ RECOMPILE)	発生件数(FREQ)			−
174	プロセス情報(PROCESS)	サービス実行中のサーバプロセス数(SERVICE COUNT)	最大値(MAX)	数値	10	−
175			最小値(MIN)			
176			平均値(AVG)			
177		最大起動プロセス数を超えたサービス要求数(REQUEST OVER)	要求回数(FREQ)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
178	RPC 情報(RPC)	自ユニットサーバへのサービスレスポンス時間(RESPONSE ON OWN UNIT)※2	発生件数(FREQ)	数値	10	－
179			最大値(MAX)			単位：100マイクロ秒
180			最小値(MIN)			
181			平均値(AVG)			
182		他ユニットのサーバへのサービスレスポンス時間(RESPONSE TO OTHER UNIT)※2	発生件数(FREQ)			－
183			最大値(MAX)			単位：100マイクロ秒
184			最小値(MIN)			
185			平均値(AVG)			
186		自ユニットのサーバからの 1 サービス当たりの実行時間(EXEC TIME ON OWN UNIT)※2	発生件数(FREQ)			－
187			最大値(MAX)			単位：100マイクロ秒
188			最小値(MIN)			
189			平均値(AVG)			
190		他ユニットのサーバからの 1 サービス当たりの実行時間(EXEC TIME FROM OTHER UNIT)※2	発生件数(FREQ)			－
191			最大値(MAX)			単位：100マイクロ秒
192			最小値(MIN)			
193			平均値(AVG)			
194		自プロセスへの SEND 回数(SEND TO OWN PRCS)	発生件数(FREQ)			－
195		自ユニットの他プロセスへの SEND 回数(SEND TO OTHER PRCS)	発生件数(FREQ)			－
196		他ユニットへの SEND 回数(SEND TO OTHER UNIT)	発生件数(FREQ)			－
197		自プロセスからの RECEIVE 回数(RECEIVE FROM OWN PRCS)	発生件数(FREQ)			－
198		自ユニットの他プロセスからの RECEIVE 回数(RECEIVE FROM OTHER PRCS)	発生件数(FREQ)			－
199		他ユニットからの RECEIVE 回数(RECEIVE FROM OTHER UNIT)	発生件数(FREQ)			－

注※1

自動採番機能使用時は順序数生成子も含まれます。

注※2

求めた時間が OS の時刻戻しなどによって負の値となる場合は，値が 0 の統計情報として出力します。そのため，表示する値は実際の値より小さいことがあります。

表 14–10　DAT 形式ファイルのレコード形式（システムの稼働に関する統計情報（その 3））

項番	フィールド名（タイトルバー）			属性	最大長	備考
200	ディクショナリ情報(DICTIONARY)	型定義情報取得要求回数(TYPE-DEF GET REQ)	発生件数(FREQ)	数値	10	−
201		ユーザ定義型情報用バッファヒット回数(TYPE-DEF CACHE HIT)	発生件数(FREQ)			−
202		ユーザ定義型情報用バッファ中の型定義情報数(CACHED TYPE-DEF)	発生件数(FREQ)			−
203			最大値(MAX)			
204			最小値(MIN)			
205			平均値(AVG)			
206		ユーザ定義型情報用バッファに取得した１型定義情報当たりのユーザ定義型情報用バッファ使用領域長(TYPE-DEF CACHE SIZE)	発生件数(FREQ)			−
207			最大値(MAX)			単位：バイト
208			最小値(MIN)			
209			平均値(AVG)			
210		ユーザ定義型情報用バッファ総使用領域長(TYPE-DEF CACHE TOTAL SIZE)	発生件数(FREQ)			−
211			最大値(MAX)			単位：バイト
212			最小値(MIN)			
213			平均値(AVG)			
214		確保したユーザ定義型情報用バッファ長(TYPE-DEF CACHE ALLOC SIZE)	発生件数(FREQ)			−
215			最大値(MAX)			単位：バイト
216			最小値(MIN)			
217			平均値(AVG)			
218		ルーチン定義情報取得要求回数(RTN-DEF GET REQ)	発生件数(FREQ)			−
219		ルーチン定義情報用バッファヒット回数(RTN-DEF CACHE HIT)	発生件数(FREQ)			−
220		ルーチン定義情報用バッファ中のルーチン定義情報数(CACHED RTN-DEF)	発生件数(FREQ)			−
221			最大値(MAX)			
222			最小値(MIN)			
223			平均値(AVG)			
224		ルーチン定義情報用バッファに取得した１ルーチン定義情報当たりのルーチン定義情報用バッファ使用領域長(RTN-DEF CACHE SIZE)	発生件数(FREQ)			−
225			最大値(MAX)			単位：バイト
226			最小値(MIN)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
227			平均値(AVG)			
228		ルーチン定義情報用バッファ総使用領域長(RTN-DEF CACHE TOTAL SIZE)	発生件数(FREQ)			－
229			最大値(MAX)			単位：バイト
230			最小値(MIN)			
231			平均値(AVG)			
232		確保したルーチン定義情報用バッファ長(RTN-DEF CACHE ALLOC SIZE)	発生件数(FREQ)			－
233			最大値(MAX)			単位：バイト
234			最小値(MIN)			
235			平均値(AVG)			
236		プラグイン提供関数のルーチン定義取得要求回数(PLG-RTN GET REQ)	発生件数(FREQ)			－
237		プラグイン提供関数のルーチン定義情報用バッファヒット回数(PLG-RTN CACHE HIT)	発生件数(FREQ)			－
238		レジストリ情報取得要求回数(REGISTRY GET REQ)	発生件数(FREQ)			－
239		レジストリ情報用バッファヒット回数(REGISTRY CACHE HIT)	発生件数(FREQ)			－
240		レジストリ情報用バッファ中のレジストリ情報数(CACHED REGISTRY-DEF)	発生件数(FREQ)			－
241			最大値(MAX)			
242			最小値(MIN)			
243			平均値(AVG)			
244		1 レジストリ情報当たりのレジストリ情報用バッファ長(REGISTRY CACHE SIZE)	発生件数(FREQ)			－
245			最大値(MAX)			単位：バイト
246			最小値(MIN)			
247			平均値(AVG)			
248		レジストリ情報用バッファ総使用領域長(REGISTRY CACHE TOTAL SIZE)	発生件数(FREQ)			－
249			最大値(MAX)			単位：バイト
250			最小値(MIN)			
251			平均値(AVG)			
252	RPC 情報(RPC)	HiRDB 予約ポートの使用数(REGISTERED PORTS)	発生件数(FREQ)	数値	10	－
253			最大値(MAX)			
254			最小値(MIN)			

項番	フィールド名（タイトルバー）			属性	最大長	備考
255			平均値(AVG)			
256		HiRDB 予約ポートオーバー時の OS 自動割り当てポート使用数(ASSIGNED PORTS)	発生件数(FREQ)			
257			最大値(MAX)			
258			最小値(MIN)			
259			平均値(AVG)			
260	ディクショナリ情報(DICTIONARY)	ディレクトリ登録のユーザ認証時間(DIRECTORY USER CHECK TIME)※7	発生件数(FREQ)	数値	10	－
261			最大値(MAX)			単位：マイクロ秒
262			最小値(MIN)			
263			平均値(AVG)			
264		グループ判定実行時間(GROUP CHECK TIME)※7	発生件数(FREQ)			－
265			最大値(MAX)			単位：マイクロ秒
266			最小値(MIN)			
267			平均値(AVG)			
268	排他制御情報(LOCK)	排他制御用プールパーティションのスレッド間ロック待ち数(REQUEST WAIT QUEUE)※1	発生件数(FREQ)	数値	10	－
269			最大値(MAX)			
270			最小値(MIN)			
271			平均値(AVG)			
272	ログ情報(LOG)	ファイルへの書き込み時間(FILE WRITE TIME)※2※7	発生件数(FREQ) ※2	数値	10	－
273			最大値(MAX) ※2※3			単位：マイクロ秒
274			最小値(MIN) ※2※3			
275			平均値(AVG) ※2※3			
276		ファイルからの読み出し時間(FILE READ TIME)※7	発生件数(FREQ)			－
277			最大値(MAX) ※3			単位：マイクロ秒
278			最小値(MIN) ※3			
279			平均値(AVG) ※3			
280		バッファリング回数(BUFFERING)	発生件数(FREQ) ※4			－
281	排他制御情報(LOCK)	スレッド間ロック要求回数(REQUEST)※1	発生件数(FREQ)			－
282		スレッド間ロック待ち回数(REQUEST WAIT)※1	発生件数(FREQ)			－

項番	フィールド名（タイトルバー）			属性	最大長	備考
283		スレッド間ロック待ち時間(REQUEST WAIT TIME)※1※5※6※7	最大値(MAX)			単位：マイクロ秒
284			最小値(MIN)			
285			平均値(AVG)			
286	ログ情報(LOG)	ログフラッシュ待ち行列数(BUF FLUSH WAIT QUEUE)	発生件数(FREQ)			－
287			最大値(MAX)			－
288			最小値(MIN)			－
289			平均値(AVG)			－
290		ログフラッシュ要求回数(BUF FLUSH)	発生件数(FREQ)			－
291		ログフラッシュ待ち回数(BUF FLUSH WAIT)	発生件数(FREQ)			－
292		ログフラッシュ待ち時間(BUF FLUSH WAIT TIME)※5※6※7	最大値(MAX)			単位：マイクロ秒
293			最小値(MIN)			
294			平均値(AVG)			

注※1

排他制御用プールパーティションが複数ある場合（pd_lck_pool_partition オペランド又は pd_fes_lck_pool_partition オペランドの値に 2 以上を指定した場合）は，排他制御用プールパーティションごとに求めたスレッド間ロック待ち数を集計して，最大，最小，平均値を出力します。

注※2

システムログのファイル出力以外にも，システムログファイルの状態を変更するときの書き込み回数及び書き込み時間が加算されます。システムログファイルを二重化している場合には，系ごとの書き込み回数及び書き込み時間が加算されます。また，システムログブロックの書き込み回数及び書き込み時間だけでなく，システムログファイルのスワップなどでファイルの状態が変化する場合に発生する，ファイルの書き込み回数及び書き込み時間も含まれます。

注※3

HiRDB のバージョンが 08-04 より前の場合に取得した統計入力アンロードファイルを，08-04 以降の統計解析ユティリティで実行すると，平均値及び最大値に 0，最小値に 4,294,967,295 が表示されます。

注※4

HiRDB のバージョンが 08-04 より前の場合に取得した統計入力アンロードファイルを，08-04 以降の統計解析ユティリティで実行すると，発生件数に 0 を表示します。

注※5

pdstbegin コマンドの-w オプションを指定し，スレッド間ロック待ち時間を取得している場合に出力します。待ち時間を取得しない場合は，0 を表示します。pdstbegin コマンドの-w オプションについては，「pdstbegin（統計情報の出力開始）のオプション」を参照してください。

注※6

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

注※7

求めた時間が OS の時刻戻しなどによって負の値となる場合は，値が 0 の統計情報として出力します。そのため，表示する値は実際の値より小さいことがあります。

表 14–11　DAT 形式ファイルのレコード形式（UAP に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	−
2	UAP 名称(UAP NAME)			
3	UAP 又はトランザクション実行開始時刻(START)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
4	項番 3 のマイクロ秒(START(MICRO))	数値	6	秒値は含みません。
5	UAP 又はトランザクション実行終了時刻(END)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
6	項番 5 のマイクロ秒(END(MICRO))	数値	6	秒値は含みません。
7	終了状態(STATUS)	文字	1	"N"正常終了 "E"ロールバックをして終了
8	UAP 又はトランザクションの実行時間(UAP EXEC TIME(S))	数値	10	単位：秒（秒値未満は四捨五入）
9	表定義情報用バッファ使用サイズ(CACHE SIZE)			単位：バイト
10	SQL オブジェクト取得要求回数(REQ(SQL))			−
11	SQL オブジェクト用バッファヒット回数(HITS(SQL))			−
12	SQL オブジェクト作成回数(CREATE)			−
13	作成した最大 SQL オブジェクトサイズ(SQL MAX)			単位：バイト
14	COMMIT 文実行回数(COMMIT)			−

項番	フィールド名（タイトルバー）	属性	最大長	備考
15	ROLLBACK 文実行回数(ROLLBACK)			−
16	FETCH/SELECT 文で実際に検索した行数(FETCH ROW)			−
17	DELETE 文で実際に削除した行数(DELETE ROW)			−
18	INSERT 文で実際に挿入した行数(INSERT ROW)			−
19	UPDATE 文で実際に更新した行数(UPDATE ROW)			−
20	SQL 文の前処理回数(SET)			−
21	OPEN 文実行回数(OPEN)			−
22	FETCH 文実行回数(FETCH)			−
23	CLOSE 文実行回数(CLOSE)			−
24	DESCRIBE 文実行回数(DESCRIBE)			−
25	SELECT 文実行回数(SELECT)			−
26	INSERT 文実行回数(INSERT)			−
27	UPDATE 文実行回数(UPDATE)			−
28	DELETE 文実行回数(DELETE)			−
29	LOCK 文実行回数(LOCK)			−
30	CREATE TABLE 実行回数(CREATE TABLE)			−
31	ALTER TABLE 実行回数(ALTER TABLE)			−
32	DROP TABLE 実行回数(DROP TABLE)			−
33	CREATE INDEX 実行回数(CREATE INDEX)			−
34	DROP INDEX 実行回数(DROP INDEX)			−
35	COMMENT TABLE 実行回数(COMMENT TABLE)			−
36	COMMENT COLUMN 実行回数(COMMENT COLUMN)			−
37	CREATE SCHEMA 実行回数(CREATE SCHEMA)			−

項番	フィールド名（タイトルバー）	属性	最大長	備考
38	DROP SCHEMA 実行回数(DROP SCHEMA)			−
39	GRANT RDAREA 実行回数(GRANT RDAREA)			−
40	GRANT SCHEMA 及び GRANT SCHEMA OPERATION の実行回数(GRANT SCHEMA)			−
41	GRANT アクセス権限実行回数(GRANT ACCESS)			−
42	GRANT CONNECT 実行回数(GRANT CONNECT)			−
43	GRANT DBA 実行回数(GRANT DBA)			−
44	REVOKE RDAREA 実行回数(REVOKE RDAREA)			−
45	REVOKE SCHEMA 及び REVOKE SCHEMA OPERATION の実行回数(REVOKE SCHEMA)			−
46	REVOKE アクセス権限実行回数(REVOKE ACCESS)			−
47	REVOKE CONNECT 実行回数(REVOKE CONNECT)			−
48	REVOKE DBA 実行回数(REVOKE DBA)			−
49	CREATE VIEW 実行回数(CREATE VIEW)			−
50	DROP VIEW 実行回数(DROP VIEW)			−
51	システム固有情報(UNUSED)			−
52	システム固有情報(UNUSED)			−
53	PURGE TABLE 文実行回数(PURGE TABLE)			−
54	そのほかの実行回数(MISC)			−
55	ストアドプロシジャのオブジェクトの取得要求回数(SQLOBJ GET REQ)			−
56	ストアドプロシジャのオブジェクトの SQL オブジェクト用バッファヒット回数(SQLOBJ CACHE HIT)			−

項番	フィールド名（タイトルバー）	属性	最大長	備考
57	CREATE PROCEDURE 実行回数(CREATE PROCEDURE)			−
58	DROP PROCEDURE 実行回数(DROP PROCEDURE)			−
59	CALL 文実行回数(CALL)			−
60	DESCRIBE 文(INPUT)実行回数(DESCRIBE INPUT)			−
61	ALTER PROCEDURE 実行回数(ALTER PROCEDURE)			−
62	UAP，又はトランザクション実行時間(UAP EXEC TIME(MS))			単位：ミリ秒（秒値を含みます。ミリ秒未満は四捨五入）
63	サービス名称(SERVICE NAME)	文字	31	−
64	排他待ち時間(LOCK TIME)※1	数値	10	単位：秒
65	項番 64 のマイクロ秒(LOCK TIME(MICRO))※1		6	秒値は含みません。
66	該当する行の出力タイミング(OUTPUT TIMMING)	文字	1	C：UAP 実行単位 T：トランザクション単位
67	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
68	コネクト通番(CONNECT NO)	数値	10	1 フロントエンドサーバ内の 1 接続ごとの通番
69	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID
70	CPU 時間(CPU TIME)※4			単位：ミリ秒
71	シングルサーバ又はフロントエンドサーバの処理時間(SERV EXEC TIME)※2			単位：秒
72	項番 71 のマイクロ秒(SERV EXEC TIME(MICRO))		6	秒値は含みません。
73	入出力時間の合計値(IO TIME)※5		10	単位：秒
74	項番 73 のマイクロ秒(IO TIME(MICRO))※5		6	秒値は含みません。
75	入出力時間の最大値(MAX IO TIME)※3		10	単位：秒

項番	フィールド名（タイトルバー）	属性	最大長	備考
76	項番 75 のマイクロ秒(MAX IO TIME(MICRO))		6	秒値は含みません。
77	入出力時間の最小値(MIN IO TIME)※3		10	単位：秒
78	項番 77 のマイクロ秒(MIN IO TIME(MICRO))		6	秒値は含みません。
79	データページ，インデクスページ，及びディレクトリページの参照回数(DID RF CNT)※5		10	－
80	データページ，インデクスページ，及びディレクトリページの更新回数(DID UPD CNT)※5			－
81	データページ，インデクスページ，及びディレクトリページのバッファヒット回数(DID HIT CNT)※5			－
82	データページ，インデクスページ，及びディレクトリページのバッファヒット率(DID BUF HIT RATIO)		3	単位：%
83	データページ，インデクスページ，及びディレクトリページの実 READ 回数(DID READ CNT)※5		10	－
84	データページ，インデクスページ，及びディレクトリページの実 WRITE 回数(DID WRITE CNT)※5			－
85	LOB 列のデータページの参照回数(LOB RF CNT)※5			－
86	LOB 列のデータページの更新回数(LOB UPD CNT)※5			－
87	LOB 列のデータページの参照バッファヒット回数(LOB RF HIT CNT)※5			－
88	LOB 列のデータページの更新バッファヒット回数(LOB UPD HIT CNT)※5			－
89	LOB 列のデータページの参照バッファヒット率(LOB RF BUF HIT RATIO)		3	単位：%

項番	フィールド名（タイトルバー）	属性	最大長	備考
90	LOB 列のデータページの更新バッファヒット率(LOB UPD BUF HIT RATIO)			
91	LOB 列のデータページの実 READ 回数(LOB READ CNT)※5		10	−
92	LOB 列のデータページの実 WRITE 回数(LOB WRITE CNT)※5			−
93	グローバルバッファフラッシュ回数(BUF FLASH CNT)※5			−
94	グローバルバッファの READ 待ち回数(BUF READ WAIT CNT)※5			−
95	グローバルバッファの WRITE 待ち回数(BUF WRITE WAIT CNT)※5			−
96	グローバルバッファの排他待ち発生回数(BUF LOCK WAIT CNT)※5			−
97	作業表用ファイル数の最大値(MAX WKFILE NUM)			−
98	作業表用ファイル増分数の最大値(MAX WKFILE EXP CNT)			−
99	作業表用ファイル容量の最大値(MAX WKFILE VOLUME)			単位：メガバイト
100	作業表用ファイルの READ 回数(WKFILE READ CNT)※5			−
101	作業表用ファイルの WRITE 回数(WKFILE WRITE CNT)※5			−
102	作業表用バッファの強制出力回数(WKFILE BUF FORCED OUT CNT)※5			−
103	一括で処理する場合のハッシュ表サイズの最大値(MAX WHOLE HASH TABLE SIZE)			単位：キロバイト
104	1 レベルバケットサイズの最大値(MAX BUCKET SIZE LEVEL 1)			
105	2 レベルバケットサイズの最大値(MAX BUCKET SIZE LEVEL 2)			
106	3 レベルバケットサイズの最大値(MAX BUCKET SIZE LEVEL 3)			

項番	フィールド名（タイトルバー）	属性	最大長	備考
107	空き領域の再利用のページサーチ空回り回数(REUSE FAILURE CNT)※5			−
108	SEGMENT REUSE 指定の表で，新規ページ追加モードから空きページ再利用モードに切り替わった回数(REUSE CHANGE CNT)※5			−
109	ローカルバッファを使用してデータページ，及びインデクスページを参照した回数(LOCAL BUF RF CNT)※5			−
110	ローカルバッファを使用してデータページ，及びインデクスページを更新した回数(LOCAL BUF UPD CNT)※5			−
111	ローカルバッファでのデータページ，及びインデクスページのバッファヒット回数(LOCAL BUF HIT CNT)※5			−
112	ローカルバッファ使用時のデータページ，及びインデクスページの実 READ 回数(LOCAL BUF READ CNT)※5			−
113	ローカルバッファ使用時のデータページ，及びインデクスページの実 WRITE 回数(LOCAL BUF WRITE CNT)※5			−
114	ローカルバッファフラッシュ回数(LOCAL BUF FLUSH CNT)※5			−
115	非同期 READ 要求回数(AR REQ CNT)※5			−
116	非同期 READ 時の同期待ち回数(AR SYNC WAIT CNT)※5			−
117	非同期 READ 時の同期待ち時間の合計値(AR SYNC WAIT TIME TOTAL)※5			単位：秒
118	項番 117 のマイクロ秒(AR SYNC WAIT TIME TOTAL(MICRO))※5		6	秒値は含みません。
119	非同期 READ 時の同期待ち時間の平均値(AR SYNC WAIT TIME AVERAGE)		10	単位：秒

項番	フィールド名（タイトルバー）	属性	最大長	備考
120	項番 119 のマイクロ秒(AR SYNC WAIT TIME AVERAGE（MICRO))		6	秒値は含みません。
121	非同期 READ 時の同期入出力時間の合計値(AR SYNC IO TIME TOTAL)※5		10	単位：秒
122	項番 121 のマイクロ秒(AR SYNC IO TIME TOTAL（MICRO))※5		6	秒値は含みません。
123	非同期 READ 時の同期入出力時間の平均値(AR SYNC IO TIME AVERAGE)		10	単位：秒
124	項番 123 のマイクロ秒(AR SYNC IO TIME AVERAGE(MICRO))		6	秒値は含みません。
125	ハッシュジョイン，副問合せのハッシュ実行でのハッシュ分割表検索時の最大比較回数（MAX HASH COMP COUNT）※3		10	－
126	ハッシュジョイン，副問合せのハッシュ実行でのハッシュ分割表検索時の総比較回数（HASH COMP COUNT）※5		10	－
127	ハッシュジョイン，副問合せのハッシュ実行での総ハッシュ分割表検索回数（HASH HIT COUNT）※5		10	－
128	一時表を使用する SQL 実行時の使用セグメント数の合計（TOTAL SEGMENTS USED FOR TEMP TABLE）		10	－
129	データ圧縮処理の総実行回数（COMPRESSION CNT）※6		10	－
130	データ圧縮処理の総実行時間（COMPRESSION TIME）※6		10	単位：秒
131	項番 130 のマイクロ秒（COMPRESSION TIME(MICRO)）※6		6	秒値は含みません。
132	データ圧縮処理の最大実行時間（MAX COMPRESSION TIME）※6		10	単位：秒
133	項番 132 のマイクロ秒（MAX COMPRESSION TIME(MICRO)）※6		6	秒値は含みません。
134	データ伸張処理の総実行回数（EXTENSION CNT）※6		10	－

項番	フィールド名（タイトルバー）	属性	最大長	備考
135	データ伸張処理の総実行時間（EXTENSION TIME）※6		10	単位：秒
136	項番 135 のマイクロ秒（EXTENSION TIME(MICRO)）※6		6	秒値は含みません。
137	データ伸張処理の最大実行時間（MAX EXTENSION TIME）※6		10	単位：秒
138	項番 137 のマイクロ秒（MAX EXTENSION TIME(MICRO)）※6		6	秒値は含みません。
139	排他制御用プールパーティションのスレッド間ロック総要求回数（LOCK POOL REQ CNT）		10	－
140	排他制御用プールパーティションのスレッド間ロック総待ち回数（LOCK POOL WAIT CNT）			－
141	排他制御用プールパーティションのスレッド間ロック待ち時間の累計（LOCK POOL WAIT TIME）※7※8			単位：秒
142	項番 141 のマイクロ秒（LOCK POOL WAIT TIME(MICRO)）※7※8		6	秒値は含みません。
143	排他制御用プールパーティションのスレッド間ロック待ち時間の最大値（LOCK POOL MAX WAIT TIME）※7※8		10	単位：秒
144	項番 143 のマイクロ秒（LOCK POOL MAX WAIT TIME(MICRO)）※7※8		6	秒値は含みません。
145	排他制御用プールパーティションのスレッド間ロック待ち行列数の累計（LOCK POOL WAIT QUEUE）		10	－
146	排他制御用プールパーティションのスレッド間ロック待ち行列数の最大値（LOCK POOL MAX WAIT QUEUE）			－
147	ログフラッシュ総要求回数（BUF FLUSH REQ CNT）			－
148	ログフラッシュ総待ち回数（BUF FLUSH WAIT CNT）			－
149	ログフラッシュ待ち時間の累計（BUF FLUSH WAIT TIME）※7※8			単位：秒

項番	フィールド名（タイトルバー）	属性	最大長	備考
150	項番 149 のマイクロ秒（BUF FLUSH WAIT TIME(MICRO)）※7※8		6	秒値は含みません。
151	ログフラッシュ待ち時間の最大値（BUF FLUSH MAX WAIT TIME）※7※8		10	単位：秒
152	項番 151 のマイクロ秒（BUF FLUSH MAX WAIT TIME(MICRO)）※7※8		6	秒値は含みません。
153	ログフラッシュ待ち行列数の累計（BUF FLUSH WAIT QUEUE）		10	－
154	ログフラッシュ待ち行列数の最大値（BUF FLUSH MAX WAIT QUEUE）			－
155	グローバルバッファプールロック総要求回数（BUF POOL LOCK REQ CNT）			－
156	グローバルバッファプールのスレッド間ロック総待ち回数（BUF POOL LOCK WAIT CNT）			－
157	グローバルバッファプールのスレッド間ロック待ち時間の累計（BUF POOL LOCK WAIT TIME）※7※8			単位：秒
158	項番 157 のマイクロ秒（BUF POOL LOCK WAIT TIME(MICRO)）※7※8		6	秒値は含みません。
159	グローバルバッファプールのスレッド間ロック待ち時間の最大値（BUF POOL LOCK MAX WAIT TIME）※7※8		10	単位：秒
160	項番 159 のマイクロ秒（BUF POOL LOCK MAX WAIT TIME(MICRO)）※7※8		6	秒値は含みません。
161	グローバルバッファプールのスレッド間ロック待ち行列数の累計（BUF POOL LOCK WAIT QUEUE）		10	－
162	グローバルバッファプールのスレッド間ロック待ち行列数の最大値（BUF POOL LOCK MAX WAIT QUEUE）			－

注

HiRDB/パラレルサーバの場合，項番 73，74，79～102，126，127 はすべてのバックエンドサーバの合計値となります。項番 139～162 は集計中にオーバフローした場合，‘****’が出力されます。

注※1

HiRDB/パラレルサーバの場合，トランザクション内で発生したすべての排他待ち時間（並列処理も含む）の合計値となります。

注※2

コネクション中又はトランザクション中に pdstbegin コマンドを実行した場合，そのコネクション又はトランザクションの情報は取得されないで，次のコネクション又はトランザクションから情報が取得されます。

注※3

HiRDB/パラレルサーバの場合，すべてのバックエンドサーバを対象とした値となります。

注※4

トランザクション内のすべてのサーバの CPU 時間の合計値となります。

注※5

トランザクション単位に情報を出力する場合は，CONNECT からの累積値となります。

注※6

この項目は BINARY 型のデータに対する情報です。

注※7

pdstbegin コマンドの-w オプションを指定し，スレッド間ロック待ち時間を取得している場合に取得します。pdstbegin コマンドの-w オプションについては，「pdstbegin（統計情報の出力開始）のオプション」を参照してください。

注※8

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

表 14–12　DAT 形式ファイルのレコード形式（SQL に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	－
2	UAP 名称(UAP NAME)			
3	SQL 実行開始時刻(START)		19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
4	項番 3 のマイクロ秒(START(MICRO))	数値	6	秒値は含みません。

項番	フィールド名（タイトルバー）	属性	最大長	備考
5	SQL 実行終了時刻(END)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
6	項番 5 のマイクロ秒(END(MICRO))	数値	6	秒値は含みません。
7	作成した SQL オブジェクトの長さ(PDATA)		10	単位：バイト
8	SQL オブジェクトを送信したバックエンドサーバの数(BES COUNT)			－
9	SQL 実行時間(SQL EXEC TIME(MICRO))			単位：マイクロ秒（秒値を含みます）
10	検索行数(FETCH ROW)			－
11	挿入行数(INSERT ROW)			
12	更新行数(UPDATE ROW)			
13	削除行数(DELETE ROW)			
14	作業表作成回数(CREATE LIST)			
15	作業表削除回数(DROP LIST)			
16	SQL 実行時間(SQL EXEC TIME(SEC))			単位：秒（秒値未満は切り捨て）
17	サービス名称(SERVICE NAME)	文字	31	－
18	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
19	コネクト通番(CONNECT NO)	数値	10	1 フロントエンドサーバ内の，1 コネクトごとの通し番号
20	SQL カウンタ(SQL COUNTER)			－
21	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID
22	オペレーションコード(OPERATION CODE)※1	文字	4	－
23	セクション番号(SECTION NO)	数値	5	
24	SQL タイプコード(SQL TYPE CODE)※2	文字	4	
25	SQL コード(SQLCODE)	数値	10	
26	サーバ名(SERVER NAME)	文字	8	
27	システム固有情報(FOREIGN REQUEST COUNT)	数値	10	－
28	システム固有情報(FOREIGN EXEC TIME)			
29	システム固有情報(FOREIGN EXEC TIME(MICRO))			
30	システム固有情報(FOREIGN FETCH ROW)			

項番	フィールド名（タイトルバー）	属性	最大長	備考
31	ハッシュジョイン，副問合せのハッシュ実行でのハッシュ分割表検索時の総比較回数(HASH COMP COUNT)	数値	10	−
32	ハッシュジョイン，副問合せのハッシュ実行での総ハッシュ分割表検索回数(HASH HIT COUNT)	数値	10	−
33	システム固有情報(FOREIGN INSERT ROW)	数値	10	−
34	システム固有情報(FOREIGN UPDATE ROW)	数値	10	
35	システム固有情報(FOREIGN DELETE ROW)	数値	10	

注※1

オペレーションコードを表「オペレーションコード」に示します。

注※2

SQL タイプコードを表「SQL タイプコード」に示します。

表 14–13　オペレーションコード

オペレーションコード	対応する SQL 文
ALCR	ALLOCATE CURSOR 文
AUI2	DELETE 文（静的 SQL），INSERT 文（静的 SQL），UPDATE 文（静的 SQL），LOCK 文（静的 SQL），PURGE TABLE 文（静的 SQL），1 行 SELECT 文（静的 SQL），FREE LOCATOR 文（静的 SQL）
AUI3	代入文（静的 SQL）
AUX	EXECUTE 文
AUXI	EXECUTE IMMEDIATE 文，すべての定義系 SQL
AUXO	EXECUTE 文（INTO 指定）
CALL	CALL 文
CLIN	INSTALL CLIB
CLRP	REPLACE CLIB
CLRM	REMOVE CLIB
CLOS	CLOSE 文
CMIT	COMMIT 文
CNCT	CONNECT 文
CPRP	コミットプリペア※
DESC	DESCRIBE 文（OUTPUT 指定）
DEST	DESCRIBE TYPE 文

オペレーションコード	対応する SQL 文
DISC	DISCONNECT 文，COMMIT 文（RELEASE 指定）
DISR	ROLLBACK 文（RELEASE 指定）
DIST	Disconnect ＋ Tran Check※
DSCM	システムが使用
DSET	DEALLOCATE PREPARE 文
DSPR	システムが使用
DSRL	システムが使用
FETC	FETCH 文
GETD	GET DIAGNOSTICS
HVAR	DESCRIBE 文（INPUT 指定）
JARI	INSTALL JAR
JARR	REPLACE JAR
JARU	REMOVE JAR
OPEN	OPEN 文（動的 SQL）
OPN2	OPEN 文（静的 SQL）
OPNR	OPEN 文（動的 SQL(複数カーソル)）
RENV	システムが使用
RNCN	CONNECT 文（TO 指定）
RNDS	DISCONNECT 文（TO 指定）
ROLL	ROLLBACK 文
RSDC	DESCRIBE 文（OUTPUT，RESULT SET 指定）
RSFT	FETCH 文（RESULT SET 指定）
RSCL	CLOSE 文（RESULT SET 指定）
SAUH	SET SESSION AUTHORIZATION 文
SET	PREPARE 文
SINF	システムが使用
SOPT	システムが使用
SVLS	システムが使用
THRE	システムが使用
THSU	システムが使用
TRCK	システムが使用

オペレーションコード	対応する SQL 文
TRC2	システムが使用
TRST	システムが使用
TSCM	システムが使用
TSRL	Transfer Rollback※
TSPR	Transfer Prepare※

注※　XA インタフェースを使用した場合だけ出力されます。

表 14–14　SQL タイプコード

項番	SQL 文	SQL タイプコード（16 進数）
1	CREATE TABLE	1038
2	DROP TABLE	103C
3	ALTER TABLE	1040
4	CREATE INDEX	1044
5	DROP INDEX	1048
6	COMMENT ON TABLE	104C
7	COMMENT ON COLUMN	1050
8	CREATE SCHEMA	1054
9	DROP SCHEMA	1058
10	GRANT RDAREA	105C
11	GRANT SCHEMA 及び GRANT SCHEMA OPERATION	1060
12	GRANT ACCESS	1064
13	REVOKE RDAREA	1068
14	REVOKE SCHEMA 及び REVOKE SCHEMA OPERATION	106C
15	REVOKE ACCESS	1070
16	CREATE VIEW	1074
17	DROP VIEW	1078
18	ALTER INDEX	1079
19	GRANT CONNECT	1098
20	GRANT DBA	109C
21	REVOKE CONNECT	10A0
22	REVOKE DBA	10A4

項番	SQL 文	SQL タイプコード（16 進数）
23	CREATE PROCEDURE	10B0
24	DROP PROCEDURE	10B4
25	ALTER PROCEDURE	10B8
26	CREATE TYPE	10C0
27	DROP DATA TYPE	10C4
28	CREATE FUNCTION	10C8
29	DROP FUNCTION	10CC
30	ALTER ROUTINE	10E8
31	LOCK 文	2134
32	SELECT 文	4000，4900，4100，4A04，4204
33	INSERT 文	4108
34	UPDATE 文	4114
35	DELETE 文	411C
36	ASSIGN LIST 文	4224，4324
37	DROP LIST 文	422C
38	PURGE TABLE 文	4184
39	CALL PROCEDURE	40BC
40	GET DIAGNOSTICS	40C0
41	代入文	40C4
42	FREE LOCATOR 文	40C8

表 14–15　DAT 形式ファイルのレコード形式（グローバルバッファプールに関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	−
2	サーバ名(SERVER)		8	
3	グローバルバッファ名称(BUFFER NAME)		16	
4	統計ログ取得時刻(LOG GET TIME)		19	-y 省略時 "MM/DD/hh:mm"形式（-e sec 省略時） "MM/DD/hh:mm:ss"形式（-e sec 指定時）

項番	フィールド名（タイトルバー）	属性	最大長	備考
				-y 指定時 "YYYY/MM/DD hh:mm:ss"形式
5	グローバルバッファプールのバッファ面数(BUFFER COUNT)	数値	10	−
6	シンクポイント出力ページ数(SYNC POINT)			
7	最大同時要求バッファ面数(BUFFER MAX)			
8	更新 GET 回数(UPDATE GET)			
9	更新バッファヒット回数(UPDATE HIT)			
10	更新バッファヒット率(UPDATE HIT RATIO)		3	単位：%
11	更新バッファフラッシュ回数(UPDATE FLUSH)		10	−
12	参照 GET 回数(REF GET)			
13	参照バッファヒット回数(REF HIT)			
14	参照バッファヒット率(REF HIT RATIO)		3	単位：%
15	参照バッファフラッシュ回数(REF FLUSH)		10	−
16	実 READ 回数(READ)			
17	入力待ち発生回数(WAIT READ)			
18	実 WRITE 回数(WRITE)			
19	出力待ち発生回数(WAIT WRITE)			
20	バッファ排他待ち発生回数(WAIT EXCLUSIVE) （バッファ上のページが，ほかのユーザによって使用中であったため，待ち状態になった回数）			
21	バッファ不足発生回数(BUFFER INSUFFICIENCY)			
22	プリフェッチ入力ページ数(PRRED)			
23	プリフェッチヒットページ数(PRHIT)			
24	プリフェッチヒット率(PRHIT RATIO)		3	単位：%
25	プリフェッチ実行資源不足発生回数 (PR INSUFFICIENCY)		10	−
26	グローバルバッファのヒット率(BUFHIT RATIO)		3	単位：%
27	カレントの参照バッファ数(CUR REF BUFNUM)		10	−
28	カレントの更新バッファ数(CUR UPDATE BUFNUM)			
29	デファードライトトリガ時の出力契機となる更新バッファ数(TRG UPDATE BUFNUM)			

項番	フィールド名（タイトルバー）	属性	最大長	備考
30	シンクポイントの発生回数(SYNC COUNT)			
31	プリフェッチの READ 要求回数(PR READ REQ CNT)			
32	LOB 用グローバルバッファの READ 要求回数(LOB READ REQ CNT)			
33	LOB 用グローバルバッファの WRITE 要求回数(LOB WRITE REQ CNT)			
34	LOB 用グローバルバッファの一括入力ページ数(LOB BLK READ PGNUM)			
35	LOB 用グローバルバッファの一括出力ページ数(LOB BLK WRITE PGNUM)			
36	システム固有情報(CACHE BUFFER INSUFFICIENCY)			
37	システム固有情報(CACHE BUFFER FLUSH MAX)			
38	システム固有情報(CACHE BUFFER FLUSH AVERAGE)			
39	グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他競合待ち回数(LOCK WAIT CNT)			指数表示
40	グローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他競合待ち発生率(LOCK WAIT RATIO)		5	単位：%（小数点 1 けたまで）
41	スリープ処理実行回数の平均値(SLEEP EXEC AVERAGE)		10	指数表示
42	スピンループ発生率(SPIN EXEC RATIO)		5	単位：%（小数点 1 けたまで）
43	スピンループ回数の平均値(SPIN EXEC AVERAGE)		10	指数表示
44	シンクポイント時のグローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他占有時間(SYNC LOCK)		10	−
45	項番 44 のマイクロ秒(SYNC LOCK(MICRO))		6	秒値を含みません。
46	シンクポイント時のグローバルバッファプール（一つの pdbuffer で定義したバッファ全体）の排他占有時間内の処理バッファ数(SYNC BUFF)		10	−
47	シンクポイント時の参照要求ヒットによるデータベース書き込み処理代行回数(REF ALTWRITE)			
48	シンクポイント時の更新要求ヒットによるデータベース書き込み処理代行回数(UPD ALTWRITE)			
49	排他競合待ち時間の累計(LOCK WAIT TIME)※1※2			単位：秒
50	項番 49 のマイクロ秒(LOCK WAIT TIME(MICRO))※1※2		6	秒値は含みません。
51	排他競合待ち行列数の累計(LOCK WAIT QUEUE)		10	−

注※1

pdstbegin コマンドの-w オプションを指定し，スレッド間ロック待ち時間を取得している場合に取得します。pdstbegin コマンドの-w オプションについては，「pdstbegin（統計情報の出力開始）のオプション」を参照してください。

注※2

表示する値の精度は，使用する OS 及びハードウェアに依存します。そのため，表示される値は実際の値より小さい場合があります。

参照 GET 回数と更新 GET 回数の合計が 4,294,967,295 回を超えるグローバルバッファが検出されると，該当するグローバルバッファの統計情報はリセットされます。詳細は，「pdbufls（グローバルバッファ情報の表示）の注意事項」を参照してください。

グローバルバッファプールに関する統計情報には，インメモリデータバッファの情報も出力されますが，各項目の出力値は概算値となります。

表 14–16　DAT 形式ファイルのレコード形式（データベース操作に関する HiRDB ファイルの統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	－
2	サーバ名(SERVER)		8	
3	HiRDB ファイル名称(FILE NAME)※		167	
4	RD エリア名称(RDAREA NAME)		30	
5	統計ログ取得時刻(LOG GET TIME)		19	-y 省略時 "MM/DD/hh:mm"形式（-e sec 省略時） "MM/DD/hh:mm:ss"形式（-e sec 指定時） -y 指定時 "YYYY/MM/DD hh:mm:ss"形式
6	同期 READ 発生回数(SYNC READ)	数値	10	－
7	同期 WRITE 発生回数(SYNC WRITE)			
8	システム固有情報(AIO READ)			
9	非同期 WRITE 発生回数(AIO WRITE)			
10	システム固有情報(LIST IO)			
11	オープン発行回数(OPEN)			
12	クローズ発行回数(CLOSE)			

項番	フィールド名（タイトルバー）	属性	最大長	備考
13	入出力エラー発生回数(IO ERROR)			

注※

項番 4 に示す RD エリアのオープン属性が INITIAL 以外で，一度もこの RD エリアにアクセスしていない場合は空白になるときがあります。空白になっているレコードは無視してください。なお，RD エリアを構成する HiRDB ファイルはデータベース状態解析ユティリティで-k phys -f を指定すると確認できます。

表 14–17　DAT 形式ファイルのレコード形式（デファードライト処理に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名称(HOST)	文字	32	－
2	サーバ名(SERVER)		8	
3	ログ取得時刻(LOG GET TIME)		19	-y 省略時 "MM/DD/hh:mm"形式（-e sec 省略時） "MM/DD/hh:mm:ss"形式（-e sec 指定時） -y 指定時 "YYYY/MM/DD hh:mm:ss"形式
4	動作要因(CAUSE)		1	トリガ："T" プレシンク："P" シンクポイント："S" データベースのシンクポイント："D" RD エリアのシンクポイント："R"
5	プレシンク完了状態(STATUS)※			プレシンク完了："P" 完全終了　　："E"
6	入出力並列度(PARALLEL)	数値	10	－
7	出力ページ総数(OUT PAGE)			動作要因ごとにデファードライトで出力したページの総数
8	WRITE 回数(DWEXEC)			－
9	WRITE 単価の最小値(DWMIN)			単位：秒（秒値未満は切り捨て）
10	項番 9 のマイクロ秒(DWMINM)		6	秒値を含みません。
11	WRITE 単価の最大値(DWMAX)		10	単位：秒（秒値未満は切り捨て）
12	項番 11 のマイクロ秒(DWMAXM)		6	秒値を含みません。

項番	フィールド名（タイトルバー）	属性	最大長	備考
13	WRITE 単価の平均値(DWAVG)		10	単位：秒（秒値未満は切り捨て）
14	項番 13 のマイクロ秒(DWAVGM)		6	秒値を含みません。
15	合計 WRITE 時間(DWSUM)		10	単位：秒（秒値未満は切り捨て）
16	項番 15 のマイクロ秒(DWSUMM)		6	秒値を含みません。
17	並列 WRITE 時間(DWPARA)		10	単位：秒（秒値未満は切り捨て）
18	項番 17 のマイクロ秒(DWPARAM)		6	秒値を含みません。
19	実行時間(DWTOTAL)		10	単位：秒（秒値未満は切り捨て）
20	項番 19 のマイクロ秒(DWTOTALM)		6	秒値を含みません。
21	システム固有情報(DWLSNGET)		10	単位：秒（秒値未満は切り捨て）
22	システム固有情報(DWLSNGETM)		6	秒値を含みません。
23	システム固有情報(DWSYNCLIST)		10	単位：秒（秒値未満は切り捨て）
24	システム固有情報(DWSYNCLISTM)		6	秒値を含みません。
25	システム固有情報(DWSEMOP)		10	単位：秒（秒値未満は切り捨て）
26	システム固有情報(DWSEMOPM)		6	秒値を含みません。
27	更新要求操作完了待ちページ数		10	シンクポイント取得処理中のバッファ更新要求時の代行処理抑止機能使用時の出力対象ページが，更新要求中のため，完了を待ったページ数
28	更新要求操作完了待ちループ回数		10	シンクポイント取得処理中のバッファ更新要求時の代行処理抑止機能使用時の出力対象ページが，更新要求中のため，完了待ち処理をループした回数

注

出力ページ総数(OUT PAGE)より WRITE 回数が少ない場合，情報が一括出力されます。一括出力された場合，一括出力しないときと比べて，項番 9～14 の WRITE 単価が増加することがあります。

注※

動作要因がプレシンク"P"の場合に出力されます。そのほかの場合はナル値が出力されます。

表 14–18　DAT 形式ファイルのレコード形式（SQL 静的最適化に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	−

項番	フィールド名（タイトルバー）	属性	最大長	備考
2	UAP 名称(UAP NAME)			
3	サービス名称(SERVICE NAME)		31	
4	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
5	コネクト通番(CONNECT NO)	数値	10	1 フロントエンドサーバ内の，コネクトごとの通し番号
6	SQL カウンタ(SQL COUNTER)			－
7	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID
8	オペレーションコード(OPERATION CODE)	文字	4	オペレーションコードについては，表「オペレーションコード」を参照してください。
9	セクション番号(SECTION NO)	数値	5	－
10	SQL タイプコード(SQL TYPE CODE)	文字	4	SQL タイプコードについては，表「SQL タイプコード」を参照してください。
11	サーバ名(SERVER NAME)		8	－
12	構文解析実行開始時刻(PARSE START(S))		19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
13	項番 12 のマイクロ秒(PARSE START(MICRO))	数値	6	秒値は含みません。
14	構文解析実行終了時刻(PARSE END(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
15	項番 14 のマイクロ秒(PARSE END(MICRO))	数値	6	秒値は含みません。
16	構文解析実行時間(PARSE EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
17	項番 16 のマイクロ秒(PARSE EXEC TIME(MICRO))			秒値を含みます。
18	意味解析実行開始時刻(SEMAN START(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
19	項番 18 のマイクロ秒(SEMAN START(MICRO))	数値	6	秒値は含みません。
20	意味解析実行終了時刻(SEMAN END(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時）

項番	フィールド名（タイトルバー）	属性	最大長	備考
				"YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
21	項番 20 のマイクロ秒(SEMAN END(MICRO))	数値	6	秒値は含みません。
22	意味解析実行時間(SEMAN EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
23	項番 22 のマイクロ秒(SEMAN EXEC TIME(MICRO))			秒値を含みます。
24	最適化実行開始時刻(OPT START(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
25	項番 24 のマイクロ秒(OPT START(MICRO))	数値	6	秒値は含みません。
26	最適化実行終了時刻(OPT END(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
27	項番 26 のマイクロ秒(OPT END(MICRO))	数値	6	秒値は含みません。
28	最適化実行時間(OPT EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
29	項番 28 のマイクロ秒(OPT EXEC TIME(MICRO))			秒値を含みます。
30	SQL オブジェクト生成実行開始時刻(SQLOBJ GEN START(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
31	項番 30 のマイクロ秒(SQLOBJ GEN START(MICRO))	数値	6	秒値は含みません。
32	SQL オブジェクト生成実行終了時刻(SQLOBJ GEN END(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
33	項番 32 のマイクロ秒(SQLOBJ GEN END(MICRO))	数値	6	秒値は含みません。
34	SQL オブジェクト生成実行時間(SQLOBJ GEN EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
35	項番 34 のマイクロ秒(SQLOBJ GEN EXEC TIME(MICRO))			秒値を含みます。
36	SQL オブジェクト登録実行開始時刻(SQLOBJ ENT START(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
37	項番 36 のマイクロ秒(SQLOBJ ENT START(MICRO))	数値	6	秒値は含みません。
38	SQL オブジェクト登録実行終了時刻(SQLOBJ ENT END(S))	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時）

項番	フィールド名（タイトルバー）	属性	最大長	備考
				"YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
39	項番 38 のマイクロ秒(SQLOBJ ENT END(MICRO))	数値	6	秒値は含みません。
40	SQL オブジェクト登録実行時間(SQLOBJ ENT EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
41	項番 40 のマイクロ秒(SQLOBJ ENT EXEC TIME(MICRO))			秒値を含みます。

表 14–19　DAT 形式ファイルのレコード形式（SQL 動的最適化に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	－
2	UAP 名(UAP NAME)			
3	サービス名称(SERVICE NAME)		31	
4	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
5	コネクト通番(CONNECT NO)	数値	10	1 フロントエンドサーバ内の，1 コネクトごとの通し番号
6	SQL カウンタ(SQL COUNTER)			－
7	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID
8	オペレーションコード(OPERATION CODE)	文字	4	オペレーションコードについては，表「オペレーションコード」を参照してください。
9	セクション番号(SECTION NO)	数値	5	－
10	SQL タイプコード(SQL TYPE CODE)	文字	4	SQL タイプコードについては，表「SQL タイプコード」を参照してください。
11	サーバ名(SERVER NAME)		8	－
12	動的最適化実行開始時刻(DYNAMIC START)		19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
13	項番 12 のマイクロ秒(START(MICRO))	数値	6	秒値は含みません。
14	動的最適化実行終了時刻(DYNAMIC END)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
15	項番 14 のマイクロ秒(END(MICRO))	数値	6	秒値は含みません。

項番	フィールド名（タイトルバー）	属性	最大長	備考
16	動的最適化実行時間(DYNAMIC EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
17	項番 16 のマイクロ秒(DYNAMIC EXEC TIME(MICRO))			秒値を含みます。
18	システム固有情報(SUB SECTION COUNT)		5	－
19	システム固有情報(SUB SECTION NO)			
20	SQL オブジェクト種別(SQLOBJ KIND)	文字	1	'F'：フロントエンド 'J'：ソート/ジョイン 'T'：実表スキャン
21	要求サーバ数(REQ SERVER)	数値	10	－
22	割り当てサーバ数(ALC SERVER)			
23	割り当てサーバ名(ALC SERVER NAME)	文字	8	
24	RD エリア ID(RDAREA ID)	数値	10	

表 14–20　DAT 形式ファイルのレコード形式（SQL オブジェクト実行に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	－
2	UAP 名(UAP NAME)			
3	サービス名称(SERVICE NAME)		31	
4	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
5	コネクト通番(CONNECT NO)	数値	10	1 フロントエンドサーバ内の，1 コネクトごとの通し番号
6	SQL カウンタ(SQL COUNTER)			－
7	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID
8	サーバ名(SERVER NAME)	文字	8	－
9	スレッド番号(THREAD NO)	数値	5	
10	オペレーションコード(OPERATION CODE)	文字	4	オペレーションコードについては，表「オペレーションコード」を参照してください。
11	セクション番号(SECTION NO)	数値	5	－
12	システム固有情報(SUB SECTION NO)			
13	SQL コード(SQLCODE)		10	
14	SQL オブジェクト実行開始時刻(SQLOBJ START)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時）

項番	フィールド名（タイトルバー）	属性	最大長	備考
				"YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
15	項番 14 のマイクロ秒(START(MICRO))	数値	6	秒値は含みません。
16	SQL オブジェクト実行終了時刻(SQLOBJ END)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
17	項番 16 のマイクロ秒(END(MICRO))	数値	6	秒値は含みません。
18	SQL オブジェクト実行時間(SQLOBJ EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
19	項番 18 のマイクロ秒(SQLOBJ EXEC TIME(MICRO))			秒値を含みます。
20	SQL オブジェクト検索行数(FETCH ROW)			−
21	SQL オブジェクト挿入行数(INSERT ROW)			
22	SQL オブジェクト更新行数(UPDATE ROW)			
23	SQL オブジェクト削除行数(DELETE ROW)			
24	作業表作成行数(CREATE LIST)			
25	作業表削除行数(DROP LIST)			
26	送信回数(SEND)			
27	送信データ長(SEND DATA SIZE)			
28	受信回数(RECEIVE)			
29	受信データ長(RECEIVE DATA SIZE)			
30	作業表作成回数(CREATE TMP LIST)			
31	作業表削除回数(DROP TMP LIST)			
32	HiRDB ファイル出力回数(IOS OUTPUT)			
33	バッファ入力回数(BUFFER INPUT)			
34	SELECT APSL の番号(SEL APSL NO)			
35	SQL オブジェクトの種別(SQLOBJ KIND)	文字	1	'F'：フロントエンド 'J'：ソート/ジョイン 'T'：実表スキャン
36	システム固有情報(FOREIGN REQUEST COUNT)	数値	10	−
37	システム固有情報(FOREIGN EXEC TIME)			
38	システム固有情報(FOREIGN EXEC TIME(MICRO))			
39	システム固有情報(FOREIGN FETCH ROW)			
40	システム固有情報(FOREIGN INSERT ROW)			

項番	フィールド名（タイトルバー）	属性	最大長	備考
41	システム固有情報(FOREIGN UPDATE ROW)			
42	システム固有情報(FOREIGN DELETE ROW)			

表 14–21　DAT 形式ファイルのレコード形式（SQL オブジェクト転送に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	−
2	UAP 名(UAP NAME)			
3	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
4	コネクト通番(CONNECT NO)	数値	10	−
5	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID
6	サーバ名(SERVER)	文字	8	−
7	システム固有情報(SERVICE KIND)		1	
8	オペレーションコード(OPERATION CODE)		4	オペレーションコードについては，表「オペレーションコード」を参照してください。
9	セクション番号(SECTION NO)	数値	5	−
10	システム固有情報(SUB SECTION NO)			
11	SQL タイプコード(SQL TYPE CODE)	文字	4	SQL タイプコードについては，表「SQL タイプコード」を参照してください。
12	SQL コード(SQLCODE)	数値	10	−
13	サービス開始時刻(SERVICE START)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
14	項番 13 のマイクロ秒(SERVICE START(MICRO))	数値	6	秒値は含みません。
15	サービス終了時刻(SERVICE END)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
16	項番 15 のマイクロ秒(SERVICE END(MICRO))	数値	6	秒値は含みません。
17	サービス実行時間(SERVICE EXEC TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
18	項番 17 のマイクロ秒(SERVICE EXEC TIME(MICRO))			秒値を含みます。

項番	フィールド名（タイトルバー）	属性	最大長	備考
19	SQL オブジェクトの転送開始時刻(TRANSFER START)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
20	項番 19 のマイクロ秒(TRANSFER START(MICRO))	数値	6	秒値は含みません。
21	SQL オブジェクトの転送終了時刻(TRANSFER END)	文字	19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
22	項番 21 のマイクロ秒(TRANSFER END(MICRO))	数値	6	秒値は含みません。
23	SQL オブジェクトの転送時間(TRANSFER TIME(SEC))		10	単位：秒（秒値未満は切り捨て）
24	項番 23 のマイクロ秒(TRANSFER TIME(MICRO))			秒値を含みます。
25	SQL オブジェクトの転送量(TRANSFER SIZE)		10	単位：バイト

注 1

次の場合，項番 8～11 には NULL（0，又は" "）を設定します。

- SERVICE KIND が'N'，'P'，又は'R'の場合

注 2

次の場合，項番 19～25 には NULL（0，又は" "）を設定します。

- SQL オブジェクトが SQL オブジェクト用バッファ上で見付かった場合
- SQL オブジェクトの転送前にエラーが発生した場合（SQL コードには，0 以外の値が設定されています）

表 14–22　DAT 形式ファイルのレコード形式（SQL 文の履歴に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	－
2	UAP 名(UAP NAME)			
3	モジュール名(MODULE NAME)			
4	サービス名(SERVICE NAME)		31	
5	フロントエンドサーバの名称(FES NAME)		8	シングルサーバの場合は，シングルサーバの名称
6	コネクト通番(CONNECT NO)	数値	10	１フロントエンドサーバ内の１コネクトごとの通し番号
7	SQL カウント(SQL COUNTER)			－
8	フロントエンドサーバのプロセス ID(FES PROCESS ID)			シングルサーバの場合は，シングルサーバのプロセス ID

項番	フィールド名（タイトルバー）	属性	最大長	備考
9	オペレーションコード(OPERATION CODE)	文字	4	オペレーションコードについては，表「オペレーションコード」を参照してください。
10	セクション番号(SECTION NO)	数値	5	−
11	SQL 長(SQL LENGTH)		10	
12	SQL 文(SQL STRING)	文字	2M	SQL 文は先頭から最大 30K バイトまでしか出力しません。

注

SQL 文の履歴に関する統計情報は，オペレーションコードが，"AUXI"，"OPN2"，"CALL"，"AUI2"，"SET"のどれかの場合だけ出力します。

表 14–23　DAT 形式ファイルのレコード形式（CONNECT/DISCONNECT に関する統計情報）

項番	フィールド名（タイトルバー）	属性	最大長	備考
1	ホスト名(HOST)	文字	32	−
2	UAP 名(UAP NAME)			
3	コネクト・ディスコネクト時間(CONNECT/DISCONNECT TIME)		19	"MM/DD/hh:mm:ss"形式（-y 省略時） "YYYY/MM/DD hh:mm:ss"形式（-y 指定時）
4	項番３のマイクロ秒(CONNECT/DISCONNECT TIME(MICRO))	数値	6	秒値は含みません。
5	フロントエンドサーバのプロセス ID(FES PROCESS ID)		10	シングルサーバの場合は，シングルサーバのプロセス ID
6	SQL コード(SQLCODE)			−
7	コネクト・ディスコネクト状態(CONNECT/DISCONNECT STATUS)	文字	1	'C'：CONNECT 'D'：DISCONNECT
8	クライアントグループ ID(CLT GROUP)			'1'：WS クライアント '2'：PC クライアント '8'：X/Open XA インタフェース '0'：ユティリティ '#'：その他 ' '：なし(スペース：0x20)
9	権限情報(PRIVILEGE)			'C'：CONNECT 権限 'D'：DBA 権限 ' '：なし(スペース：0x20)
10	コネクト・ディスコネクト種別(CONNECT/DISCONNECT KIND)			'N'：Normal(CONNECT/DISCONNECT) 'S'：set session authorization

項番	フィールド名（タイトルバー）	属性	最大長	備考
				'I'：内部ディスコネクト 'T'：トランザクション解決(OLTPだけ)
11	認可識別子(USER NAME)		30	－
12	パスワード(PASSWORD)			KFPA11560-E エラー(パスワード不正)時だけ
13	CLT IP アドレス(CLT IP ADDRESS)		15	XXX.XXX.XXX.XXX 形式

表 14–24　DAT 形式ファイルのレコード形式（データベースの入出力に関する統計情報）

項番	フィールド名（タイトルバー）		属性	最大長	備考
1	ホスト名(HOST)		文字	32	－
2	サーバ名(SERVER)			8	－
3	入出力が発生した RD エリアの名称(RDAREA NAME)			30	－
4	項番 3 に示す RD エリアを構成する HiRDB ファイルの通番(FILE NUMBER)		数値	2	－
5	統計ログを取得した時刻(LOG GET TIME)		文字	19	"YYYY/MM/DD hh:mm:ss"形式
6	HiRDB ファイルのページ長(PAGE SIZE(K))		数値	6	単位：キロバイト
7	取得時間間隔内に発生した read 回数(READ CNT)			10	－
8	取得時間間隔内に発生した read のうち，時間を計測した回数(READ MEASURE CNT)				－
9	時間を計測した read の時間のマイクロ秒(READ TIME(MICRO))	合計値(SUM)		12	秒値を含みます。
10		最大値(MAX)			
11	取得時間間隔内に発生した write 回数(WRITE CNT)			10	－
12	取得時間間隔内に発生した write のうち，時間を計測した回数(WRITE MEASURE CNT)				－
13	時間を計測した write の時間のマイクロ秒(WRITE TIME(MICRO))	合計値(SUM)		12	秒値を含みます。
14		最大値(MAX)			－
15	ランク種別(RANK KIND)		文字	2	上位 10 件を選定した項目 read 回数 ：RC read 合計時間：RS read 最大時間：RM write 回数 ：WC write 合計時間：WS write 最大時間：WM 選定項目なし※：*

注※

pdstbegin -D everyfile 指定時

14.5　注意事項

1. pdstedit 実行時のリターンコードは，0 の場合は正常終了又は警告終了，1 の場合は異常終了となります。
2. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdstedit の入力ファイルには，BOM を持つファイルを使用できることがあります。pdstedit での BOM を持つファイルの使用可否を次の表に示します。なお，pdstedit の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。

表 14–25　pdstedit での BOM を持つファイルの使用可否（UTF-8 の場合）

オプション	入力ファイル	BOM を持つファイルの使用
-l	統計入力アンロードファイル	×
-d	制御文ファイル	○

（凡例）

○：使用できます。

×：使用できません。

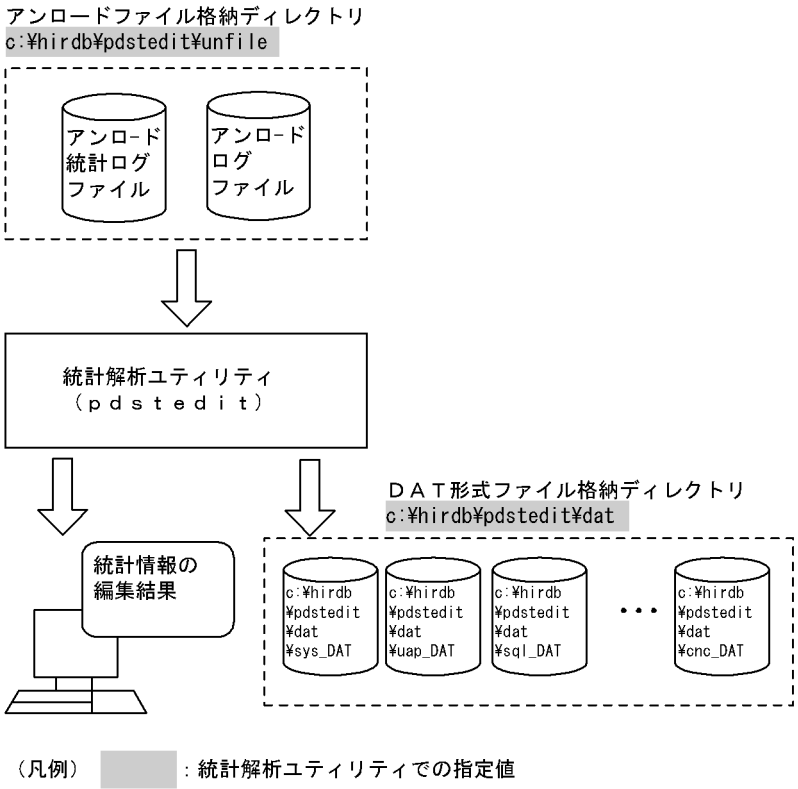
14.6　使用例

統計解析ユティリティの使用例を次に示します。

例 1

統計入力アンロードファイルから情報を入力し，すべての統計情報を編集します。また，DAT形式ファイルにも情報を出力します。

[概要]



[コマンド実行例]

```
pdstedit -k all   ......................1
         -i c:¥hirdb¥pdstedit   ........2
         -o c:¥hirdb¥pdstedit¥dat   ....3
```

[説明]

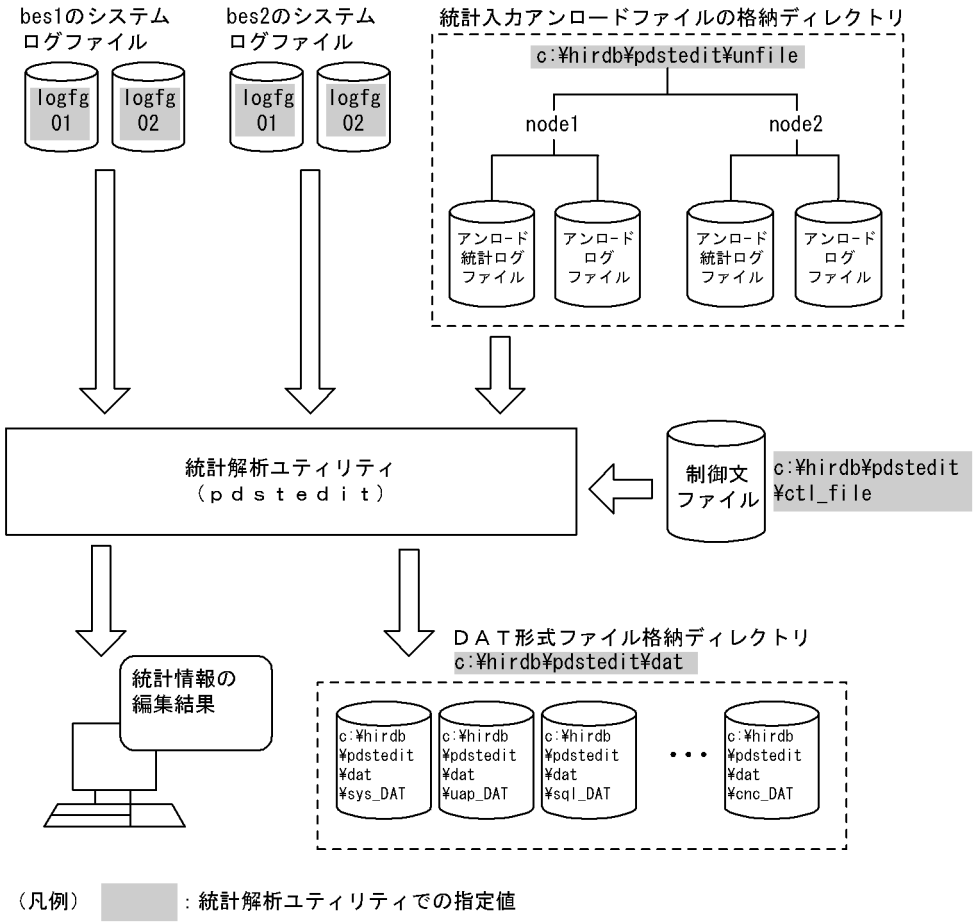
1. すべての統計情報を編集する場合の指定
2. 統計入力アンロードファイルがある統計入力アンロードファイル格納ディレクトリの名称
   c:¥hirdb¥pdstedit
3. 統計情報を格納する DAT 形式ファイル格納ディレクトリの名称　**c:¥hirdb¥pdstedit¥dat**

例 2

pdstedit実行ホストから参照できる次の解析対象ファイルを入力して，統計情報を編集します。また，DAT形式ファイルにも情報を出力します。

・一つのディレクトリ下に集めた複数のアンロード統計ログファイル及びアンロードログファイル
・アンロード前のシステムログファイル

[概要]



（凡例）　：統計解析ユティリティでの指定値

[コマンド実行例]

```
pdstedit -k all   ..........................1
         -i c:¥hirdb¥pdstedit   ............2
         -d c:¥hirdb¥pdstedit¥ctl_file   ...3
         -o c:¥hirdb¥pdstedit¥dat   ........4
```

[説明]

1. すべての統計情報を編集する場合の指定
2. 統計入力アンロードファイルがある統計入力アンロードファイル格納ディレクトリの名称
   c:¥hirdb¥pdstedit¥unfile
3. 制御文ファイルの名称　　**c:¥hirdb¥pdstedit¥ctl_file**
4. 統計情報を格納する DAT 形式ファイル格納ディレクトリの名称　**c:¥hirdb¥pdstedit¥datfile**

[制御文ファイル（c:¥hirdb¥pdstedit¥ctl_file）の内容]

```
file_group bes1:logfg01,logfg02
file_group bes2:logfg01,logfg02
```

[説明]

bes1，bes2：

指定したファイルグループに対応するサーバ名

logfg01，logfg02：

解析対象のシステムログファイルがあるファイルグループ名

15 データベース状態解析ユティリティ（pddbst）

この章では，データベースの状態解析，状態解析結果の蓄積，及び再編成時期を予測するデータベース状態解析ユティリティ（pddbst）について説明します。

15.1　概要

実行者　DBA 権限を持っているユーザが実行できます

pddbst には，次の三つの機能があります。

- DB 状態解析機能
- 状態解析結果蓄積機能
- 再編成時期予測機能

15.1.1　DB 状態解析機能

DB 状態解析機能とは，HiRDB のディレクトリを参照して，ディクショナリ表，又はユーザ用 RD エリアの表とインデクスの格納状態を解析し，これらの解析結果を集計して表示する機能です。

DB 状態解析機能は，次の単位で実行できます。

- **RD エリア単位の状態解析**
- **表又はインデクス単位の状態解析**
- **RD エリア単位でのクラスタキー及びクラスタリングデータページの格納状態解析**

(1)　RD エリア単位の状態解析

RD エリア単位の状態解析には，論理的解析と物理的解析があります。

RD エリア単位の状態解析（論理的解析）と（物理的解析）を次の図に示します。

図 15–1　RD エリア単位の状態解析（論理的解析）

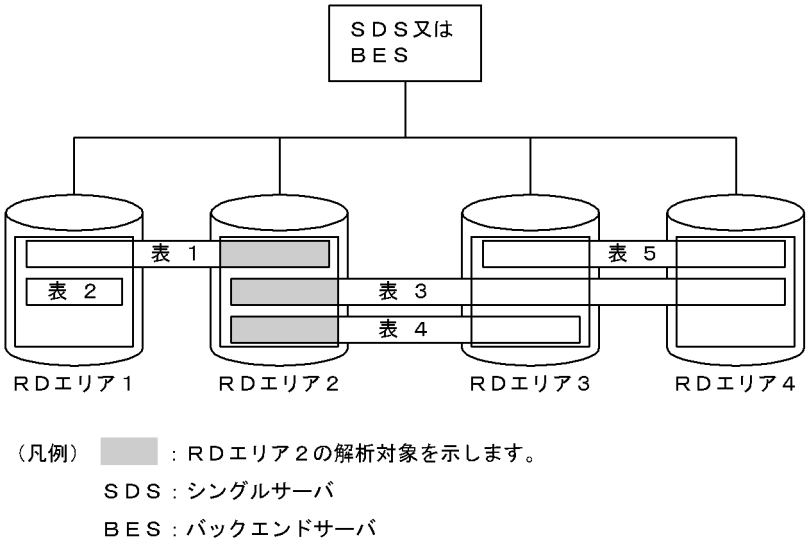
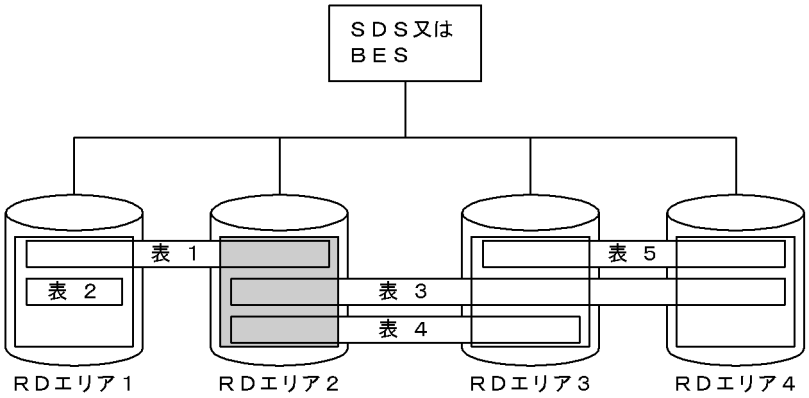


図 15–2　RD エリア単位の状態解析（物理的解析）

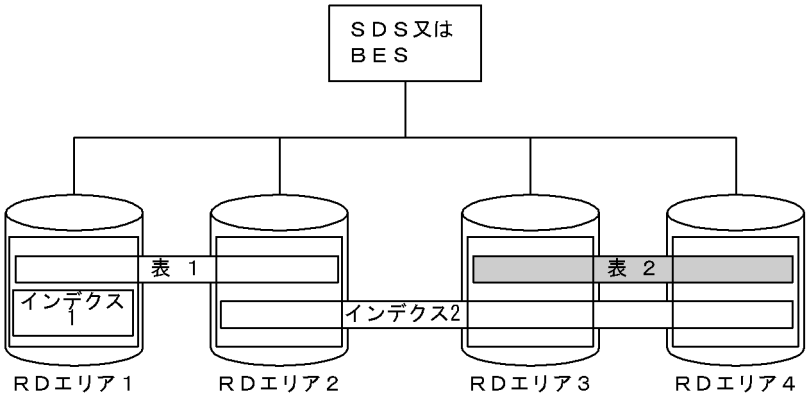


（凡例）　：ＲＤエリア２の解析対象を示します。
ＳＤＳ：シングルサーバ
ＢＥＳ：バックエンドサーバ

(2)　表単位又はインデクス単位の状態解析

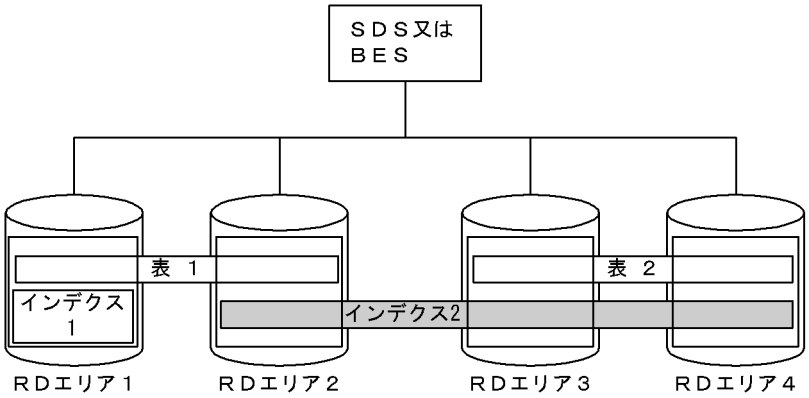
表単位の状態解析とインデクス単位の状態解析を次の図に示します。

図 15–3　表単位の状態解析



（凡例）　：表２の解析対象を示します。
ＳＤＳ：シングルサーバ
ＢＥＳ：バックエンドサーバ

図 15–4　インデクス単位の状態解析



（凡例）　：インデクス２の解析対象を示します。
ＳＤＳ：シングルサーバ
ＢＥＳ：バックエンドサーバ

(3)　RD エリア単位でのクラスタキー及びクラスタリングデータページの格納状態解析

クラスタキーを解析し，インデクス及び表の乱れ具合を示す格納不正率を表示します。

解析結果としては，クラスタキーの並びの解析結果とクラスタリングデータページの並びの解析結果を，格納 RD エリアごとに表示します。

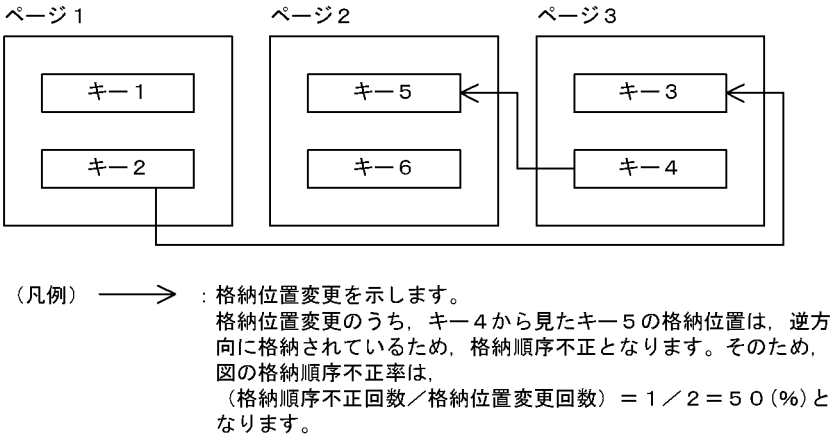
(a)　クラスタキーの並び解析

クラスタキーをキー順に検索し，次の解析結果を表示します。

- 格納位置がページ，又はセグメントにわたって格納されている格納位置変更回数を表示します。
- 格納位置変更回数のうち，格納順序がページ（セグメント）の昇順に対し，逆方向に格納されている格納順序不正回数（率）を，ページ単位/セグメント単位に表示します。ページ分割の発生によって，格納順序不正率は上昇します。

格納位置変更回数と格納順序不正回数（率）を次の図に示します。

図 15–5　格納位置変更回数と格納順序不正回数（率）



(b)　クラスタリングデータページの並び解析

クラスタリングデータページとは，クラスタキーの付けられた表のデータページのことをいいます。クラスタキー中の行データの格納位置情報を基に，クラスタキー順に検索した場合の行データの格納位置変更回数と格納順序不正回数（率）を，ページ単位/セグメント単位に表示します。

行の追加及び更新によって格納状態が乱れると，格納順序不正率か格納位置変更回数のどちらか，又は両方が増加します。

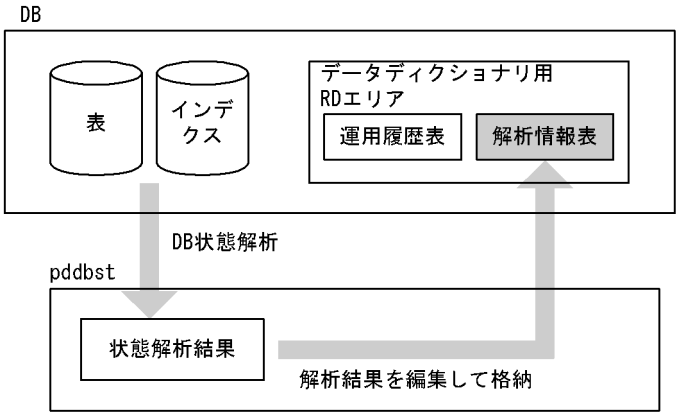
15.1.2　状態解析結果蓄積機能

状態解析結果蓄積機能とは，再編成時期予測機能を使用する場合に必要となる，DB の状態解析結果を蓄積するための機能です。

状態解析結果蓄積機能では，DB 状態解析機能の RD エリア単位の論理的解析を実行し，解析結果を編集して解析情報表に格納します。状態解析結果蓄積機能は，毎日実行することをお勧めします。

状態解析結果蓄積機能の概要を次の図に示します。

図 15–6　状態解析結果蓄積機能の概要



状態解析結果蓄積機能には，**予測レベル 1**，**予測レベル 2** の二つのレベルがあります。これらのレベルは，再編成時期予測機能の予測レベル 1，予測レベル 2 にそれぞれ対応しています。なお，予測レベル 2 の場合，より詳細な予測をするため，予測レベル 1 に比べて実行時間が長くなります。予測レベル 2 の状態解析結果蓄積機能を実行する場合は，「予測レベル 2 での状態解析結果蓄積機能の実行方法」を参照してください。

なお，この機能で蓄積されるデータ量の見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」の「解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

状態解析結果蓄積機能を使用する場合の前提条件：

状態解析結果蓄積機能を使用する場合，次の準備をしておく必要があります。

1. システム定義に pd_rorg_predict=Y を設定する。
2. pdmod で解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアを作成する。

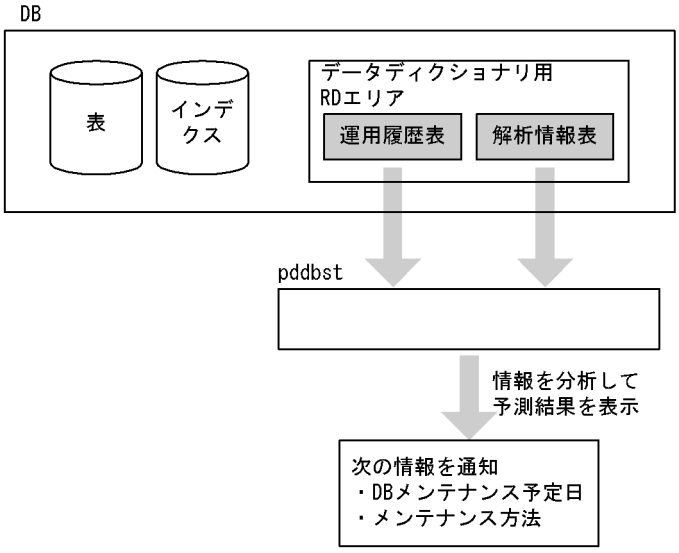
15.1.3　再編成時期予測機能

(1)　再編成時期予測機能とは

再編成時期予測機能とは，状態解析結果蓄積機能で蓄積した情報（解析情報表）と運用履歴表の情報を分析して，RD エリアのメンテナンスが必要となる時期（**DB メンテナンス予定日**）を予測し，メンテナンスの予定日と方法を通知する機能です。再編成時期予測機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

再編成時期予測機能の概要を次の図に示します。

図 15–7　再編成時期予測機能の概要



注

RD エリア内にリソースが一つだけある場合（例えば，1RD エリアに 1 表だけ格納されているなど），解析情報表及び運用履歴表の情報から予測します。RD エリア内にリソースが複数ある場合，解析情報表の情報から予測します。

再編成時期予測機能には，次の二つのレベルがあります。

- **予測レベル 1**
  RD エリアの容量不足を予測する方法です。
- **予測レベル 2**
  RD エリアの容量不足のほかに，表及びインデクスの格納効率の悪化を予測する方法です。

(2)　再編成時期予測機能を使用する場合の前提条件

再編成時期予測機能を使用する場合，次の準備をしておく必要があります。

1. システム定義に pd_rorg_predict=Y を設定する。
2. pdmod で解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアを作成する。
3. 状態解析結果蓄積機能で 4 回分以上の解析情報を蓄積する。

(3)　再編成時期の予測の仕方

再編成時期の予測の仕方を次に示します。

(a)　予測レベル 1 の場合

予測レベル 1 では，RD エリアの使用中セグメント数の比率が，基準値を超える時期（DB メンテナンス予定日）を予測します。指定した日数以内（監視期間※1）に，その基準値に到達すると予測された RD エリアについて，メンテナンスが必要であると判断します。メンテナンスが必要な場合，そのメンテナンス方法も通知します。メンテナンス方法は，メンテナンス延長期間※1 の指定値によって，再編成した方がよいのか，又は RD エリアの拡張をした方がよいのかを判断します。

なお，基準値については，基準値定義ファイル※2 を指定することで変更できます。

注※1

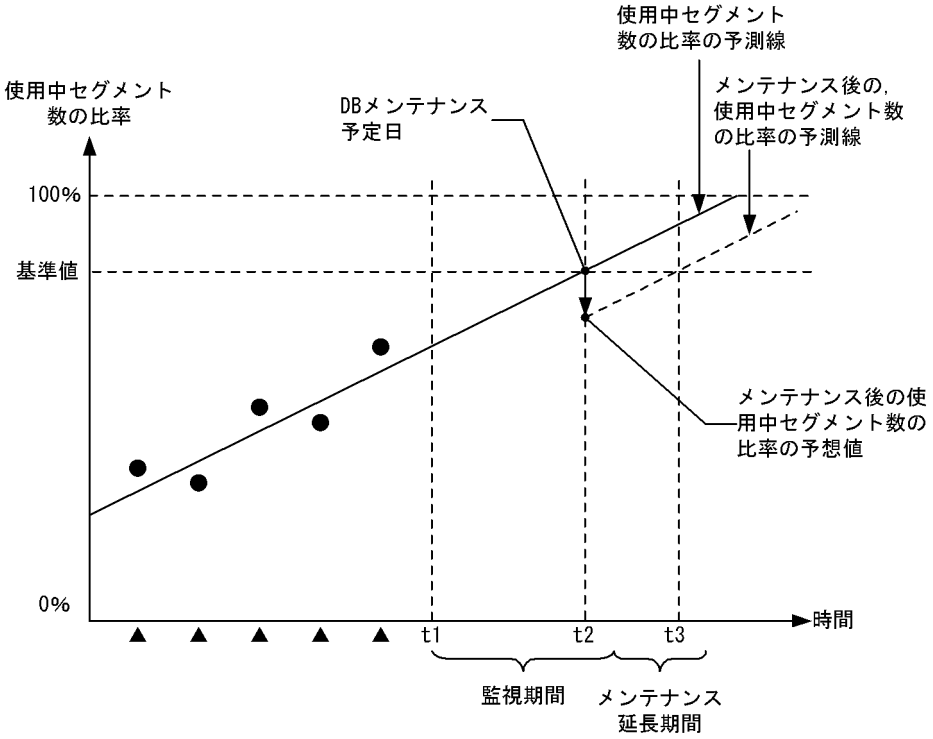
監視期間，及びメンテナンス延長期間は，-R オプションで指定します。

注※2

基準値定義ファイルは，-c オプションで指定します。

RD エリアの使用中セグメント数の比率と，メンテナンス予定日，監視期間，及びメンテナンス延長期間の関係を次の図に示します。

図 15–8　RD エリアの使用中セグメント数の比率と，メンテナンス予定日，監視期間，及びメンテナンス延長期間の関係



（凡例）

▲：状態解析結果蓄積機能の実行

●：状態解析結果蓄積機能の実行時に取得した使用中セグメント数の比率

t1：再編成時期予測機能の実行

t2：使用中セグメント数の比率の予測線が，基準値に到達する予測日

t3：メンテナンス後に，使用中セグメント数の比率の予測線が，基準値に到達する予測日

(b)　予測レベル 2 の場合

予測レベル 2 では，予測レベル 1 の RD エリアの容量不足の予測のほかに，表及びインデクスの格納効率の悪化についても予測をします。

(4)　障害発生時にデータベースを最新の状態に回復しない場合の再編成時期予測

障害発生時に，データベースを最新の状態に回復しない場合（バックアップ取得時点への回復，又は範囲指定の回復），そのデータベースの状態で再編成時期予測機能を実行すると，正しい予測ができなくなります。このような運用を行っているシステムで再編成時期予測機能を使用する場合は，**状態解析結果蓄積情報のリセット**をすれば，正しい予測ができます。

例えば，状態解析結果蓄積機能を実行した後に障害が発生し，バックアップ取得時点（状態解析結果蓄積機能を実行する前の時点）までデータベースを回復したとします。この状態で再編成時期予測機能を実行

すると，状態解析結果蓄積情報よりデータベースの内容が古い状態であるため，正しい予測ができなくなります。このような場合に，バックアップ取得時点に回復したときに状態解析結果蓄積情報のリセットを行うと，状態解析結果蓄積情報とデータベースの内容が同じになるため，正しい予測ができるようになります。

状態解析結果蓄積機能のリセットを行う場合，-I オプションを指定します。

15.2　コマンドの形式

15.2.1　形式

(1)　DB 状態解析機能の場合

RD エリア単位の状態解析

```
pddbst -r　{RDエリア名〔, RDエリア名〕…|ALL}
        〔-k　{logi　〔-d〕　〔-z〕|phys　〔-f〕}〕
        〔-a　〔-h〕〕　〔-u 認可識別子　〔-p パスワード〕〕
        〔-b〕
        〔-X サーバ間通信の応答監視時間〕
        〔-v 制御文ファイル名〕
        〔-W 実行監視時間〕
```

表単位の状態解析

```
pddbst -t　〔認可識別子.〕表識別子
        〔-s〕　〔-u 認可識別子　〔-p パスワード〕〕　〔-d〕〔-z〕
        〔-b〕
        〔-X サーバ間通信の応答監視時間〕
        〔-v 制御文ファイル名〕
        〔-W 実行監視時間〕
```

インデクス単位の状態解析

```
pddbst -i　〔認可識別子.〕インデクス識別子
        〔-u 認可識別子　〔-p パスワード〕〕　〔-d〕　〔-z〕
        〔-b〕
        〔-X サーバ間通信の応答監視時間〕
        〔-v 制御文ファイル名〕
        〔-W 実行監視時間〕
```

クラスタキー及びクラスタリングデータページの格納状態解析

```
pddbst -t 〔認可識別子.〕表識別子
       〔-u 認可識別子 〔-p パスワード〕〕 -k clus
       〔-b〕
       〔-X サーバ間通信の応答監視時間〕
       〔-v 制御文ファイル名〕
       〔-W 実行監視時間〕
```

(2) 状態解析結果蓄積機能の場合

```
pddbst -r {RDエリア名〔, RDエリア名〕…|ALL} 〔-k logi〕 -e 予測レベル
       〔-w 休止時間, セグメント数〕 〔-n 解析分割数〕 〔-I〕
       〔-u 認可識別子 〔-p パスワード〕〕
       〔-X サーバ間通信の応答監視時間〕
       〔-v 制御文ファイル名〕
       〔-W 実行監視時間〕
```

(3) 再編成時期予測機能の場合

```
pddbst 〔-r ALL〕 -k pred -e 予測レベル
       〔-m〕 〔-R 監視期間〔, メンテナンス延長期間〕〕
       〔-c 基準値定義ファイル名〕 〔-u 認可識別子 〔-p パスワード〕〕
       〔-X サーバ間通信の応答監視時間〕
       〔-v 制御文ファイル名〕
       〔-W 実行監視時間〕
```

15.2.2 オプション

(1) -r {RD エリア名〔, RD エリア名〕...| ALL}

～<識別子>((1～30))

解析する RD エリアの名称を指定します。

指定できる RD エリアを次に示します。

- データディクショナリ用 RD エリア
- データディクショナリ LOB 用 RD エリア
- ユーザ用 RD エリア（ただし，RD エリアの論理的解析の場合は一時表用 RD エリアを除きます）
- ユーザ LOB 用 RD エリア
- レジストリ用 RD エリア
- レジストリ LOB 用 RD エリア

RD エリアの指定方法については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。

ALL を指定すると，上記のすべての RD エリアが処理対象になります。上記以外の RD エリアの処理はスキップされます。

＜規則＞

1. 状態解析結果蓄積機能，及び再編成時期予測機能では，1RD エリアに格納されているすべての表を，1 回のトランザクションで解析します。そのため，システム定義の pd_max_access_tables オペランドの指定値は，解析対象 RD エリア内で最も表を多く格納している RD エリアの，表数以上必要となります。

(2) -k {logi | phys | clus | pred}

DB 状態解析機能，状態解析結果蓄積機能，又は再編成時期予測機能のどれを実行するのかを指定します。

logi：

RD エリアの状態解析（論理的解析），又は状態解析結果蓄積機能を実行する場合に指定します。

phys：

RD エリアの状態解析（物理的解析）を実行する場合に指定します。

clus：

クラスタキー及びクラスタリングデータページの格納状態解析を実行する場合に指定します。

pred：

再編成時期予測機能を実行する場合に指定します。

＜-k オプションとほかのオプションとの組み合わせ＞

-k オプションの指定値よって，-r，-t，及び-i オプションの指定可否が変わります。

-k オプションの指定	オプションの指定可否			備考
	-r	-t	-i	
logi	○	×	×	RD エリア単位の状態解析（論理的解析）

-k オプションの指定	オプションの指定可否			備考
	-r	-t	-i	
phys	○	×	×	RD エリア単位の状態解析（物理的解析）
clus	×	○	×	クラスタキー及びクラスタリングデータページの格納状態解析
省略	×	○	×	表単位の状態解析
	×	×	○	インデクス単位の状態解析
	○	×	×	RD エリア単位の状態解析（論理的解析）
pred	△	×	×	再編成時期予測機能

（凡例）

○：必ず指定してください。

△：指定は任意です。

×：指定できません。

(3) -d

ページの詳細情報を表示する場合に指定します。このオプションは，論理的解析（-k logi），表単位の状態解析，及びインデクス単位の状態解析の場合に指定できます。ページの詳細情報は，pdrorg，及び pdreclaim の実行要否の基になる情報です。

(4) -z

ページに格納されている表やインデクスの行の情報を表示する場合に指定します。このオプションで表示した内容を基に，空きページ解放ユティリティ（pdreclaim）を実行して削除状態の管理行を解放するかどうかを判断します。解析対象ごとに表示できる情報を次に示します。

項番	表示情報		RD エリア論理解析	RD エリア物理解析	表単位解析	インデクス解析	クラスタキー解析
1	表情報	削除状態の管理行数※	○	−	○	−	−
2	インデクス情報	残存エントリが管理する行数	○	−	−	○	−
3		インデクスが管理する総行数	○	−	−	○	−

（凡例）

○：表示できます。

−：表示できません。

注※

次の行数の合計値を表示します。

- 削除中（トランザクション決着待ち）の行数
- 削除済み（トランザクション決着済み）の行数

＜規則＞

このオプションを指定する場合，同時に-d オプションを指定する必要があります。

(5) -f

HiRDB ファイル情報を表示する場合に指定します。このオプションは，物理的解析（-k phys）の場合に指定できます。

(6) -a

解析結果を DAT 形式で表示する場合に指定します。

なお，RD エリアの状態解析（物理的解析）でこのオプションを指定した場合，-f オプションの指定有無に関係なく，-f オプション指定が仮定されます。

(7) -h

-a オプション指定時に，ヘッダを付ける場合に指定します。

(8) -u 認可識別子

pddbst を実行するユーザの，認可識別子を指定します。

＜規則＞

1. このオプションを省略した場合，クライアント環境定義の PDUSER で定義した認可識別子とパスワードが仮定されます。PDUSER が設定されていない場合，ユティリティを実行するユーザの OS 上のユーザ ID に対応するユーザ名が仮定されます。
2. 認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(9) -p パスワード

-u オプションで指定した認可識別子に対応するパスワードを指定します。

＜規則＞

1. このオプションの指定を省略し，クライアント環境定義の PDUSER にも値が設定されていない場合，ユティリティの実行画面に入力された値が仮定されます。
2. パスワードを「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

(10)　-t〔認可識別子.〕表識別子

解析する表の名称を指定します。

＜規則＞

1. ビュー表，及び一時表は指定できません。このオプションにビュー表又は一時表を指定して実行すると，pddbst は KFPK10106-W メッセージを出力して，処理を終了します。
2. 認可識別子を省略した場合は，-u オプションで指定した認可識別子が仮定されます。
3. 認可識別子又は表識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。
4. 表識別子に空白が含まれる場合は，「¥""」と「"¥"」で囲んでください。

(11)　-s

表単位の状態解析で，格納行数を表示する場合に指定します。表の総格納行数及び RD エリアごとの格納行数を表示します。

オンライン業務と同時に実行する場合は-b も指定してください。-b を指定しない場合は，pddbst によるデータベースの読み込み及び参照によって，先にアプリケーションなどで読み込んでいたページがグローバルバッファから追い出されるため，オンライン業務の性能が遅くなる場合があります。

(12)　-i〔認可識別子.〕インデクス識別子

解析するインデクスの名称を指定します。

＜規則＞

1. 一時インデクスは指定できません。このオプションに一時インデクスを指定して実行すると，pddbst は KFPK10106-W メッセージを出力して，処理を終了します。
2. 認可識別子を省略した場合は，-u オプションで指定した認可識別子が仮定されます。
3. 認可識別子又はインデクス識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。
4. インデクス識別子に空白が含まれる場合は，「¥""」と「"¥"」で囲んでください。

(13)　-b

pddbst 実行によるアプリケーションへのオンライン性能劣化を防ぐ場合に指定します。

-s オプションの指定がない場合は，このオプションが指定されたものとして動作します。

-s オプションと同時に指定すると，解析時間が長くなることがあります。また，ディクショナリ表からデータベースの状態解析対象となる表，インデクスの情報を検索する処理や，再編成時期予測機能での運用管理表の検索処理では，-b オプションの指定は無効となります。

＜適用基準＞

pddbst を実行すると，解析対象の資源のページをグローバルバッファに読み込んでから処理をします。このとき，グローバルバッファ不足が発生すると，先にアプリケーションなどで読み込んでいたページがグローバルバッファから追い出されるため，オンライン性能が劣化する可能性があります。この性能劣化を回避する場合，-b オプションを指定します。

-b オプションを指定した場合，pddbst だけが読み込んだページは常に最古のページとして扱い，グローバルバッファ不足時には最初に追い出されるページとなります。-b オプションを指定しない場合は LRU 管理方式となります。

＜利点＞

グローバルバッファが不足した場合でも，アプリケーションのオンライン性能への影響がありません。ただし，pddbst 実行時に，行数を取得するためのディレクトリページの再読み込みが発生します。

(14)　-e 予測レベル

状態解析結果蓄積機能，又は再編成時期予測機能を実行する場合の予測レベルを指定します。

1：

状態解析結果蓄積機能，又は再編成時期予測機能を予測レベル 1 で実行する場合に指定します。

2：

状態解析結果蓄積機能，又は再編成時期予測機能を予測レベル 2 で実行する場合に指定します。

＜規則＞

1. 状態解析結果蓄積機能と再編成時期予測機能の予測レベルは同じにしてください。
2. -e オプションを指定した場合，-b オプション指定が仮定されます。

(15)　-w 休止時間，セグメント数

予測レベル 2 の状態解析結果蓄積機能を，インターバル解析で実行する場合に指定します。インターバル解析については，「インターバル解析」を参照してください。指定したセグメント数分の解析をした後，休止時間に指定した時間だけ処理を休止します。

休止時間　　～＜符号なし整数＞((10～60000))

休止時間を 10 ミリ秒単位に指定します。10 ミリ秒単位の指定でない場合，1 けた目を切り上げます（例：15→20）。

セグメント数　　～＜符号なし整数＞((1～2147483647))

解析をするセグメント数を指定します。

＜規則＞

1. -w オプションは，予測レベル 2（-e 2）の場合に有効となります。そのほかの場合に指定すると，制御文エラーになります。
2. HiRDB/パラレルサーバの場合，指定値はサーバごとに有効となります。

3. グローバルバッファに解析対象のページがない場合，休止時間が来るまで（指定したセグメント数×セグメントサイズ）回の入出力が発生します。

(16)　-n 解析分割数

～<符号なし整数>((2～10))

予測レベル 2 の状態解析結果蓄積機能を，マージ解析で実行する場合に指定します。マージ解析については，「マージ解析」を参照してください。

1 回の状態解析結果蓄積機能の実行を，指定した解析分割数に分けて行います。ここで指定した解析分割数（n）を分母にして，1/n のデータが 1 回の解析対象となります。最終的には，n 回実行すると，対象となる全データが解析されます。

＜規則＞

1. -n オプションは，予測レベル 2（-e 2）の場合に有効となります。そのほかの場合に指定すると，制御文エラーになります。
2. -n オプションを指定する場合，指定した回数分 pddbst を実行する必要があります。途中で-n オプションの指定値を変更した場合は，前回までに蓄積した解析情報は無効となり，再度最初から解析を始めます。
3. 全体の解析が終わる前の状態（例えば，全体の 2/3 が完了している状態）で，表やインデクスをメンテナンスした場合は，再度最初から解析を始めます（1/3 から開始します）。

(17)　-l

状態解析結果蓄積情報をリセットする場合に指定します。状態解析結果蓄積情報をリセットについては，「障害発生時にデータベースを最新の状態に回復しない場合の再編成時期予測」を参照してください。

＜規則＞

1. -l オプションを指定する場合，-r オプションに ALL は指定できません。
2. -l オプションは，データベースを回復した後の 1 回目の状態解析結果蓄積機能実行時にだけ指定してください。それ以外のタイミングで-l オプションを指定すると，状態解析結果蓄積情報がリセットされます。

(18)　-m

再編成時期予測機能で，DB メンテナンス予定日の情報，及びメンテナンス方法の情報を出力する場合に指定します。省略した場合は，DB メンテナンス予定日の情報だけ出力されます。

＜注意事項＞

予測の結果，メンテナンスが必要となる RD エリアが存在した場合，RD エリア及び RD エリア内のリソースの状態を解析します。このため，処理対象となる RD エリア数，RD エリア内の表及びインデクスの総数に応じて，コマンド実行時間が長くなることがあります。

(19)　-R 監視期間〔，メンテナンス延長期間〕

再編成時期予測機能を実行する場合，対象とする期間とメンテナンスの延長期間を指定します。

監視期間　～＜符号なし整数＞((1～400))《14》

監視対象期間を日数で指定します。

例えば，この先 2 週間以内に，DB メンテナンス予定日となる RD エリアを監視したい場合，14 と指定します。

メンテナンス延長期間　～＜符号なし整数＞((0～400))《監視期間÷2》

DB メンテナンス予定日にメンテナンスをした場合，次の DB メンテナンス予定日を迎えるまでの余裕期間を，日数で指定します。

例えば，DB メンテナンスを実行した後，次の DB メンテナンスまで 1 週間の余裕を持たせたい場合は，7 と指定します。

(20)　-c 基準値定義ファイル名

～＜パス名＞((1023 バイト以内))

再編成時期予測機能を実行する場合，あらかじめ決められた各項目の基準値を基に予測をしています。この各項目の基準値を変更する場合に，基準値定義ファイルを絶対パス名又は相対パス名で指定します。

基準値定義ファイルには，変更後の各項目の基準値を記述します。基準値定義ファイルについては，「基準値定義ファイル（再編成時期予測機能）」を参照してください。

(21)　-X サーバ間通信の応答監視時間

～＜符号なし整数＞　((1～65535))　《300》

コマンドを実行したサーバで通信障害などの障害が発生した場合，コマンドは無応答となり，業務が停止するおそれがあります。pddbst では障害を検知するために，コマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が，-X オプションに設定した時間を超えた場合は，pddbst はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。

＜適用基準＞

- 通信障害やユニットダウンによってサーバから応答が返らない状態となった場合に，300 秒より早く障害を検知したいとき，-X オプションに 300 より小さい値を指定してください。
- 系切り替え機能を使用している場合，系切り替えが完了しているにもかかわらず，コマンドが応答を待ち続けることがあります。その場合は，監視時間を短く設定することで，コマンドを即時終了できます。

- 同時実行するアプリケーションやユティリティの多重度が高い場合など，ディクショナリからの応答が遅延し，-X オプションのデフォルトである 300 秒以内にユティリティの前処理が完了しないときは，監視時間でタイムアウトするおそれがあります。このような環境では，-X オプションには 300 より大きい値を指定してください。

(22)　-v 制御文ファイル名

～＜パス名＞((1023 バイト以内))

pddbst の制御文を指定したファイルを，絶対パス名又は相対パス名で指定します。

このファイルは，pddbst を実行するホストになければなりません。指定できる制御文を次に示します。なお，制御文ファイル中にコメントは記述できません。

- predict 文
- workdir 文
- option 文

(23)　-W 実行監視時間

～＜符号なし整数＞((0～3600))

データベース状態解析ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。

このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

オプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

15.2.3　predict 文

再編成時期予測機能の実行結果を CSV 形式でファイル出力する場合に指定します。

＜規則＞

1. predict 文は，再編成時期予測機能を実行するときだけ指定できます。
2. predict 文は，制御文ファイル中に一つだけ指定できます。

(1)　形式

```
predict file=CSV出力ファイル名
```

(2)　説明

(a)　CSV 出力ファイル名

～＜パス名＞((1023 バイト以内))

再編成時期予測機能の結果を出力するファイル名を絶対パス名で指定します。

出力されるファイルの形式については，「CSV 形式での出力形式」を参照してください。

＜規則＞

1. HiRDB/パラレルサーバの場合，システムマネジャがあるホストに作成してください。
2. 指定するファイルには，HiRDB 管理者のアクセス権限がなければなりません。
3. ファイル名に空白を含む場合は，引用符（"）で囲んでください。
4. 指定したファイルがない場合はファイルを作成し，既にある場合はファイルが上書きされます。

15.2.4　workdir 文

pddbst 実行時に使用する作業用ワークファイルの出力先ディレクトリを指定します。作業用ワークファイルは，DB 状態解析の実行やソート処理のために，pddbst が使用するファイルです。

＜規則＞

1. workdir 文は，制御文ファイル中に一つだけ指定できます。
2. workdir 文を省略した場合，シングルサーバ又はシステムマネジャのホストの，次の表に示すディレクトリに作業用ワークファイルを作成します。このファイルは pddbst 終了後に削除されますが，異常終了した場合などは削除されないことがあります。

表 15–1　pddbst が作業用ワークファイルを出力するディレクトリ

workdir 文の指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	workdir 文に指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※　コマンド実行環境の環境変数の指定

(1) 形式

```
workdir dir=作業用ワークファイル用ディレクトリ名
```

(2) 説明

(a) 作業用ワークファイル用ディレクトリ名

～<パス名>((1000 バイト以内))

作業用ワークファイルの出力先ディレクトリを絶対パス名で指定します。

<規則>

1. HiRDB/パラレルサーバの場合，システムマネジャがあるホストのディレクトリを指定してください。
2. 指定するディレクトリには，HiRDB 管理者のアクセス権限がなければなりません。
3. ディレクトリ名に空白を含む場合は，引用符（"）で囲んでください。
4. 指定したディレクトリに同一ファイルがない場合はファイルを作成し，既にある場合はファイルが上書きされます。

15.2.5 option 文

pddbst 実行時のオプションを指定します。

<規則>

option 文は，制御文ファイル中に一つだけ指定できます。

(1) 形式

```
sortbufsize=ソート用バッファサイズ
```

(2) 説明

(a) ソート用バッファサイズ

～<符号なし整数>((256～2097151))《256》

解析結果のソート処理で使用するバッファのサイズを指定します。単位はキロバイトです。

<規則>

ソート用バッファサイズは，次の計算式を目安に指定してください。

バッファサイズ　≧ ↑√n ×0.16↑
（単位：キロバイト）

n は処理対象数です。解析種別によって，次の表のように処理対象数が変わります。

表 15–2　解析種別ごとの処理対象数

解析種別	処理対象数
RD エリアの状態解析（論理的解析）	解析対象となる RD エリアに定義された表，及びインデクスの合計値
RD エリアの状態解析（物理的解析）	解析対象となる RD エリア数
表単位の解析	解析対象となる表が格納された RD エリア数
インデクス単位の解析	解析対象となるインデクスが格納された RD エリア数
クラスタキー及びクラスタリングデータページの格納状態解析	解析対象となるクラスタキーインデクスが格納された RD エリア数

15.3　DB 状態解析機能の出力形式

15.3.1　RD エリア単位の状態解析（論理的解析）

RD エリア内の表及びインデクスに対する，全セグメント及び全ページの格納状態を解析します。ただし，複数の RD エリアに分割格納された表及びインデクスは，指定した RD エリアに格納された部分だけ解析します。

(1)　使用目的

一つの RD エリア内の表及びインデクスの乱れの検知，容量の状態が分かります。

(2)　解析結果

RD エリア単位の状態解析（論理的解析）で表示される解析結果を次に示します。

(a)　LOB 用 RD エリア以外の場合

```
 pddbst VV-RR(Object Option) ** RD Area Logical Analysis **  2008/12/25 15:54:10 [1]
 RD Area Name   : user_rdarea_1 [2]
  Server        : bes01 [3]
  Total Segment :        150[4] Segment Size :          5 Pages [6]
  Unused Segment:          9[5] Page Size    :       4096 Bytes [7]
  Original RD Area Name : user_rdarea_1 [21]
  Generation Number :  0[22]  Replica RD Area Count :  0 [23]
  [24]     [25]               [26]               [27]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
-------------------------------------------------------------------------------

 Table Name : TBL01 [8]
 Auth Id    : user1 [9]
 Status     :   [20]
 Reference Pending Status :   [33]
 Check     Pending Status :   [34]
 Segment Reuse :          0 segments [29]
 Segment Reuse Option :          1
  <Base row segment>
  Search Mode : INS [28]
  Branch Row Search Mode :       - [58]
  Reuse Search Failure :          0/         0 [30]
              Used(Full)       Used(      Full)        Sum
              [10]  [11]        [12]       [13]        [14]
  Segment [39]100%(  0%)        107(         0)        107
              [15]  [16]        [17]       [18]        [19]
  Page [40]   100%(  0%)        535(         0)        535
  Collect On Segment :          0 [53]
  Collect Prearranged Page :          0 [31]
  Collect Prearranged Full Page :          0 [59]
  Used Page Ratio       Page(Ratio) [32]
             0% :          0(   0%)
```

```
         1- 10% :           1(   1%)
        11- 20% :           0(   0%)
        21- 30% :           0(   0%)
        31- 40% :           0(   0%)
        41- 50% :           0(   0%)
        51- 60% :           0(   0%)
        61- 70% :         118(  22%)
        71- 80% :         416(  78%)
        81- 90% :           0(   0%)
       91-100% :            0(   0%)
 Total                    535
 Del RowID   :                      0 [60]

Table Name : TBL02
Auth Id    : user1
Status     :
Reference Pending Status :
Check     Pending Status :
Segment Reuse :          - segments
Segment Reuse Option :           - [61]
 <Base row segment>
 Search Mode : INS
 Branch Row Search Mode : -
 Reuse Search Failure :           0/         0
         Used(Full)       Used(      Full)        Sum
 Segment 100%(  0%)           1(         0)          1
 Page     60%(  0%)           3(         0)          5
 Collect On Segment :           0
 Collect Prearranged Page :           0
 Collect Prearranged Full Page :           0
 Used Page Ratio       Page(Ratio)
            0% :           2(  40%)
         1- 10% :           1(  20%)
        11- 20% :           1(  20%)
        21- 30% :           0(   0%)
        31- 40% :           1(  20%)
        41- 50% :           0(   0%)
        51- 60% :           0(   0%)
        61- 70% :           0(   0%)
        71- 80% :           0(   0%)
        81- 90% :           0(   0%)
       91-100% :            0(   0%)
 Total                     5
 Del RowID   :                      0
 <BINARY segment>
 Search Mode : INS
 Branch Row Search Mode : -
 Reuse Search Failure :           0/         0
             Used(Full)       Used(      Full)        Sum
             [41]  [42]        [43]       [44]        [45]
 Segment [46]100%( 93%)          14(        13)         14
             [47]  [48]        [49]       [50]        [51]
 Page  [52]  96%( 95%)          67(        66)         70
 Collect On Segment :           0
 Collect Prearranged Page :           0
 Collect Prearranged Full Page :           0
 Used Page Ratio       Page(Ratio)
            0% :           3(   5%)
```

```
         1- 10% :          0(   0%)
        11- 20% :          0(   0%)
        21- 30% :          0(   0%)
        31- 40% :          0(   0%)
        41- 50% :          0(   0%)
        51- 60% :          0(   0%)
        61- 70% :          0(   0%)
        71- 80% :          0(   0%)
        81- 90% :          0(   0%)
       91-100% :         67(  96%)
 Total                   70

 Index Name : IDX_TBL01_C1
 Auth Id    : user1
 Status     :
          Used(Full)       Used(      Full)        Sum
 Segment 100%(  0%)          19(         0)         19
 Page    100%(  0%)          95(         0)         95
 Collect On Segment :           0
 Collect Prearranged Page :          0
 Collect Prearranged Full Page :          0
 Collect On Page :          0 [54]
 Used Page Ratio       Page(Ratio) Del RowID [35]
            0% :          0(   0%)          0
         1- 10% :          0(   0%)          0
        11- 20% :          0(   0%)          0
        21- 30% :          1(   1%)          0
        31- 40% :          1(   1%)          0
        41- 50% :          0(   0%)          0
        51- 60% :          0(   0%)          0
        61- 70% :         93(  98%)          0
        71- 80% :          0(   0%)          0
        81- 90% :          0(   0%)          0
       91-100% :          0(   0%)          0
 Total                   95                 0 [36]
 Del RowID   :                    0     (  0%) [37]
 RowID Count :               10,000 [38]

 Sequence Name : SEQ1 [55]
 Auth Id       : user1 [56]
  Current Value :                              10000001 [57]
```

[説明]

処理対象 RD エリアが複数ある場合，次の順序で表示されます。

- RD エリア名（昇順）
- 表名（昇順）
- インデクス名（昇順）
- 順序数生成子識別子（昇順）

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒
2. 解析対象の RD エリアの名称を表示します。

3. RD エリアを管理するサーバの名称を表示します。
4. RD エリアの総セグメント数（使用中セグメント数＋未使用セグメント数）を表示します。
5. RD エリアの未使用セグメント数の合計を表示します。
6. 1 セグメントのサイズをセグメント内のページ数で表示します。
7. 1 ページのサイズをバイト単位で表示します。
8. RD エリアにある表又はインデクスの名称を表示します。
9. 表又はインデクスの認可識別子を表示します。データディクショナリ用 RD エリアの場合は，(Data dictionary)と表示されます。
10. 使用中セグメント数の比率を表示します。
    ↑使用中セグメント数÷14 の値×100↑（%）
11. 満杯セグメント数の比率を表示します。
    ↑満杯セグメント数÷14 の値×100↑（%）
12. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
13. 満杯セグメント数を表示します。
14. 表又はインデクスに割り当てられているセグメント数の合計（使用中セグメント数＋未使用セグメント数）を表示します。
15. 使用中ページ数の比率を表示します。
    ↑使用中ページ数÷19 の値×100↑（%）
16. 満杯ページ数の比率を表示します。
    ↑満杯ページ数÷19 の値×100↑（%）
17. 使用中ページ数を表示します。
18. 満杯ページ数を表示します。
19. 表又はインデクスに割り当てられているセグメント中のページ数の合計（使用中ページ数＋未使用ページ数）を表示します。
20. 表の場合は，データ未完状態であるか，又は再編成実行中であるかを表示します。次のどちらでもない場合は，空白を表示します。

    DATA_UNFINISH：

    データ未完状態です。表の再編成を実行し，途中でエラーなどによってリロード処理が完了していない状態のことをいいます。データ未完状態を解除するためには，表の再編成を再実行するか，又は表へのリロードを実行する必要があります。

    ON_RORG：

    再編成実行中です。

    インデクスの場合は，インデクスの未完状態を表示します。未完状態でない場合は空白を表示します。

UNFINISH_0：

図「表単位のインデクス作成未完状態の種類（1/2）」，図「表単位のインデクス作成未完状態の種類（2/2）」を参照してください。

UNFINISH_1：

図「RD エリア単位のインデクス作成未完状態の種類（1/2）」，図「RD エリア単位のインデクス作成未完状態の種類（2/2）」を参照してください。

UNFINISH_2：

プラグインインデクス遅延更新中。プラグインインデクス遅延一括作成機能を使用した UAP を実行中，又は UAP 終了後に pdrorg のインデクス一括作成を実行していない状態です。プラグインインデクスの遅延一括作成については，マニュアル「HiRDB システム運用ガイド」を参照してください。

21. 解析対象の RD エリアの名称を表示します。
22. 常に 0 を表示します。
23. 常に 0 を表示します。
24. 該当する RD エリアの閉塞履歴情報です。

    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）

    History2：History1 の一つ前の閉塞情報
25. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

    CMD：HiRDB の障害検知によるコマンド閉塞

    FLT：障害閉塞
26. 該当する RD エリアの閉塞要因コードを表示します。履歴がない場合は 0 を表示します。

    0：閉塞解除

    10：入出力エラー

    20：ページ破壊

    40：オープン，確保エラー

    70：ロールバックエラー

    80：DROP TABLE，DROP INDEX 実行時のエラーによる閉塞

    90：ログレス閉塞（pdload，pdrorg，及び pdrbal）

    92：ログレス閉塞（UAP）

    93：LOB 用 RD エリアの閉塞

    96：タイムスタンプ不正

    97：オブジェクト ID 不正

    98：リスト用 RD エリアの異常
27. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。
28. ページサーチモードを表示します。

    INS：新規ページ追加モード

    REU：空きページ再利用モード

29. CREATE TABLE 実行時，又は ALTER TABLE 実行時の SEGMENT REUSE で指定したセグメント数です。SEGMENT REUSE でセグメント数を指定していない場合は"0"を表示します。また，SEGMENT REUSE に NO を指定している場合，及び SEGMENT REUSE を省略している場合は"-"を表示します。
30. 新規ページ追加モードから空きページ再利用モードに切り替えたときに，再利用できる空き領域がなく，新規ページ追加モードに戻した回数（n/m）を表示します。

    n：空回り回数

    m：新規ページ追加モードから空きページ再利用モードに切り替えた回数

    n 及び m は，次の契機でリセットされ，0 が設定されます。

    - RD エリアのクローズ時
    - HiRDB の再開始時
    - PURGE TABLE 文実行時
31. pdreclaim で解放できるページ数（使用率 0%の使用中空きページの数）を表示します。
32. 使用中ページの比率別ページ数を 10%ごとに表示します。括弧内の値は，すべての使用中ページに対する比率を表します。比率は小数点以下を切り上げて表示するため，比率を合計すると 100 を超えることがあります。注意事項を次に示します。

    - 201 件以上重複したキー値を格納するページは，表定義時の PCTFREE の比率が無効となります。そのため，再編成する場合でも PCTFREE の比率どおりに再配置されません。
    - インデクスキー値無排他を適用したインデクスの場合，削除キー値は空き領域として扱い，使用中ページのすべてが削除キー値のときは，使用率 0%となります。
    - 満杯ページであっても，使用率が 91−100%であるとは限りません。
    - 行長が短くなる更新をして空いた領域は，空き領域として扱いません。
    - （100−PCTFREE で指定した空き領域比率）よりも，使用中ページ比率が小さいページが多い場合，格納効率が低下している可能性があります。この場合，pdrorg の実行を検討してください。ただし，分岐行（256 バイト以上の可変長文字列データ，繰返し列のデータ，及び抽象データ型列のデータ）を格納している領域は，pdrorg を実行しても格納効率は変わりません。
33. RD エリア中の表情報の参照制約の検査保留状態を表示します。表に参照制約の定義がない場合は空白を表示します。

    P：検査保留状態

    空白：非検査保留状態
34. RD エリア中の表情報の検査制約の検査保留状態を表示します。表に検査制約の定義がない場合は空白を表示します。

    P：検査保留状態

    空白：非検査保留状態
35. インデクスキー値無排他を使用している場合又はコミットしていない削除データに排他を掛ける場合，次の数値を表示します。

    - 使用中インデクスページの比率別残存エントリが管理する行数

（残存エントリとは，インデクスキー値の更新や削除によって，キーの削除後も排他制御のためにインデクス上に残っているインデクスエントリ）

表示けた数は 3 けたで，小数点と単位を含めた最大表示けた数は 5 けたです。表示する行数が 3 けたを超える場合に小数点と単位を表示します。その場合，先頭から 4 けた目を四捨五入します。残存エントリが管理する行数の表示形式を次に示します。

数値の範囲	表示形式
0～999	0～999
1,000～999,499	1.00K～999K
999,500～999,499,999	1.00M～999M
999,500,000～999,499,999,999	1.00G～999G
999,500,000,000～999,499,999,999,999	1.00T～999T

（凡例）

K：キロ　M：メガ　G：ギガ　T：テラ

36. 使用中ページの比率別ページ数の合計と，残存エントリが管理する行数の合計を表示します。
37. インデクスの削除状態の，管理行数の合計と，全管理行数に対する割合を表示します。
38. インデクスが管理する総行数を表示します。15 けたまでは整数，16 けた以上になると指数で表示します。例えば，1000000000000000 は 1E+15 と表示します。表示形式を次に示します。

数値の範囲	表示形式
0～999	0～999
1000～9999	1,000～9,999
100000000000000～999999999999999	100,000,000,000,000～999,999,999,999,999
1000000000000000～	z,zzz,zzz,zzz,zzz,zz9E+zz9 （有効けた数 16 けたの指数形式）

39. セグメント単位のサマリ情報です。バイナリ専用セグメントの情報は含みません。
40. ページ単位のサマリ情報です。
41. 使用中バイナリ専用セグメント数の比率を表示します。

    ↑使用中バイナリ専用セグメント数÷表に割り当てられているバイナリ専用セグメント数の合計×100↑（%）
42. 満杯バイナリ専用セグメント数の比率を表示します。

    ↑満杯バイナリ専用セグメント数÷表に割り当てられているバイナリ専用セグメント数の合計×100↑(%)
43. 使用中バイナリ専用セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中バイナリ専用セグメントとして表示します。
44. 満杯バイナリ専用セグメント数を表示します。

45. 表に割り当てられているバイナリ専用セグメント数の合計を表示します。

46. バイナリ専用セグメント単位のサマリ情報です。

47. バイナリ専用セグメントの使用中ページ数の比率を表示します。

    ↑バイナリ専用セグメントの使用中ページ数÷表に割り当てられているバイナリ専用セグメントのページ数の合計×100↑（%）

48. バイナリ専用セグメントの満杯ページ数の比率を表示します。

    ↑使用中バイナリ専用セグメントの満杯ページ数÷表に割り当てられているバイナリ専用セグメントのページ数の合計×100↑（%）

49. バイナリ専用セグメントの使用中ページ数を表示します。

50. バイナリ専用セグメントの満杯ページ数を表示します。

51. 表に割り当てられているバイナリ専用セグメントのページ数の合計を表示します。

52. バイナリ専用セグメントのページ単位のサマリ情報です。

53. -a オプションを指定した pdreclaim が同時実行中，又はエラー終了した場合に，解放途中のセグメント数を表示します。-a オプションを指定した pdreclaim を実行していない場合は 0 を表示します。

54. pdreclaim の実行途中，又は pdreclaim がエラーで終了しているために解放処理が途中になっている状態のページ数を表示します。エラーで終了している場合は，pdreclaim を再実行してください（このページ数は 0 になります）。なお，表示するページ数は，使用中ページの比率別（0%）のページ数にカウントされます。

55. 順序数生成子識別子です。

56. 順序数生成子の認可識別子です。

57. 順序数生成子の現在値です。現在値が設定されていない場合，値を表示しません。順序数生成子にログ出力間隔を指定している場合，サーバの強制停止などによって，表示した現在値と次に使用する順序番号の値が異なることがあります。この場合，現在値の後に"*"を表示しますが，UAP などで現在値を使用するときに正しい値に補正されるため，その後の論理的解析実行時には正しい値が表示されます。

58. 分岐行に対する空き領域サーチモードを表示します。

    INS：新規ページ追加モード

    REU：空きページ再利用モード

    SEGMENT REUSE の再利用オプション値で「分岐行が多発する表の格納効率向上」を指定していない場合は，"-"が表示されます。

59. pdreclaim 又は pdrorg で解放できる満杯ページ数を表示します。

60. 表の削除状態の，管理行数の合計を表示します。

61. 表定義の SEGMENT REUSE で指定した再利用オプション値を表示します。

注 1

31，及び 32 は，-d オプションを指定した場合に表示される情報です。

また，35～38 は-d オプションと-z オプションを同時に指定した場合に表示される情報です。

注 2

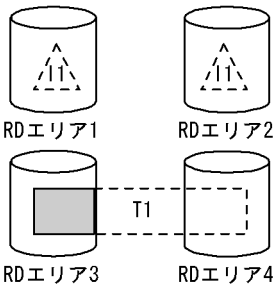
39～52 は，表に，定義長 256 バイト以上の BINARY 型の列，又は抽象データ型の列が定義されている場合に出力されます。

図 15–9　表単位のインデクス作成未完状態の種類（1/2）

B-treeインデクスの場合

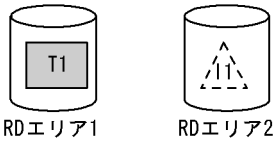
● 表単位インデクス作成未完状態
表単位のインデクス作成未完状態は，pdload又はpdrorgでインデクス一括作成，
又はインデクス再作成をすれば，インデクスを使用できる状態に変更できます。

・横分割表のRDエリア単位のデータロード又は再編成の，分割キーインデクス，又は
サーバ内分割した非分割キーインデクスの一括作成前



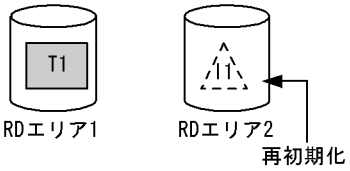
表が格納されているRDエリア3に対応する，RDエリア1のインデクスが未完状態となります。RDエリア2のインデクスの状態は変わりません。

・EMPTYオプションを指定してCREATE INDEXを実行した場合



RDエリア2のインデクスが未完状態となります。

・pdmodでインデクス格納用RDエリアを再初期化した後

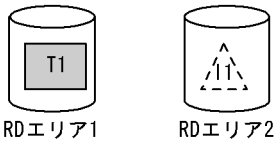


RDエリア2のインデクスが未完状態となります。

図 15–10　表単位のインデクス作成未完状態の種類（2/2）

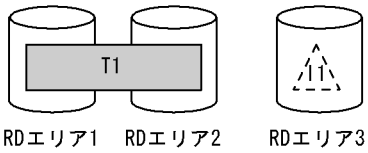
・表単位のデータロード又は再編成実行後の，分割キーインデクス又は非分割キーインデクスの一括作成前

＜非分割表，非分割キーインデクスの場合＞



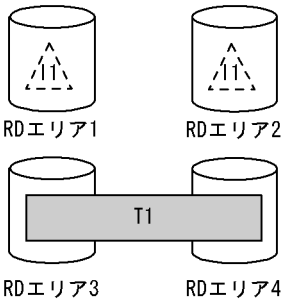
RDエリア2のインデクスが未完状態となります。

＜横分割表，非分割キーインデクスの場合＞



RDエリア3のインデクスが未完状態となります。

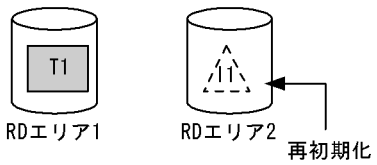
＜横分割表で，分割キーインデクス，又はサーバ内分割した非分割キーインデクスの場合＞



表が格納されているRDエリア3，RDエリア4に対応する，RDエリア1，RDエリア2のインデクスが未完状態となります。

プラグインインデクスの場合

- pdmodによるインデクス格納RDエリア再初期化状態
  pdmodによるインデクス格納RDエリア再初期化状態は，pdload又はpdrorgでインデクスを再作成すれば，インデクスを使用できる状態に変更できます。



RDエリア2のインデクスが未完状態となります。

（凡例）

T1　：表

：表に定義してあるインデクス

又は　：データがあることを示しています。

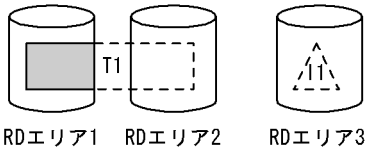
又は　：データがないことを示しています。

図 15–11　RD エリア単位のインデクス作成未完状態の種類（1/2）

B-treeインデクスの場合

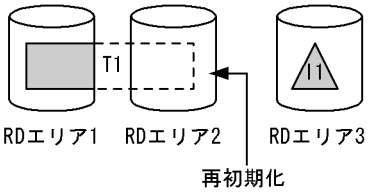
- RDエリア単位のインデクス作成未完状態
  RDエリア単位のインデクス作成未完状態は，pdload又はpdrorgでインデクス一括作成，又はインデクス再作成をすれば，インデクスを使用できる状態に変更できます。

・横分割表のRDエリア単位のデータロード又は再編成の，サーバ内分割していない非分割キーインデクスの一括作成前



RDエリア3のインデクスが未完状態となります。

・pdmodで，サーバ内分割していない非分割キーインデクスを定義している横分割表の一部を再初期化した後

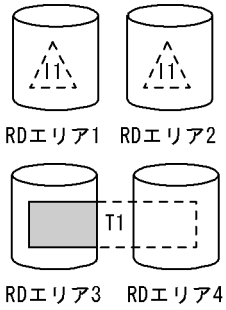


RDエリア3のインデクスが未完状態となります。

プラグインインデクスの場合

- 表単位のインデクス作成未完状態
  表単位のインデクス作成未完状態は，pdload又はpdrorgでインデクス一括作成，又はインデクス再作成をすれば，インデクスを使用できる状態に変更できます。

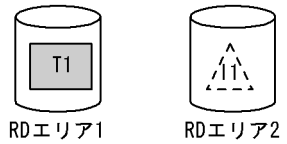
・横分割表のRDエリア単位のデータロード又は再編成の，分割キーインデクス，又はサーバ内分割した非分割キーインデクスの一括作成前



表が格納されているRDエリア3に対応する，RDエリア1のインデクスが未完状態となります。RDエリア2のインデクスの状態は変わりません。

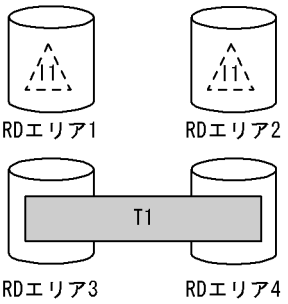
図 15–12　RD エリア単位のインデクス作成未完状態の種類（2/2）

・表単位のデータロード又は再編成実行後の，分割キーインデクス又は非分割キーインデクスの一括作成前
〈非分割表，非分割インデクスの場合〉



RDエリア2のインデクスが未完状態となります。

〈横分割表で，分割キーインデクス，又はサーバ内分割した非分割キーインデクスの場合〉



表が格納されているRDエリア3，RDエリア4に対応する，RDエリア1，RDエリア2のインデクスが未完状態となります。

（凡例）

T1　：表

：表に定義してあるインデクス

又は　：データがあることを示しています。

又は　：データがないことを示しています。

(b)　LOB 用 RD エリアの場合

```
 pddbst VV-RR(Object Option) ** RD Area Logical Analysis **  2003/04/03 12:36:43 [1]
 RD Area Name   : user_rdlob_2 [2]
  Server        : bes1 [3]
  Total Segment :      11000 [4] Segment Size :          1 Pages [6]
  Unused Segment:       9178(      9178) [5]
  Page Size     :       8192 Bytes [7]
  Original RD Area Name : user_rdlob_2 [16]
  Generation Number :  0[17]  Replica RD Area Count :  0 [18]
  [19]     [20]               [21]               [22]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
-------------------------------------------------------------------------------
 Table Name : TBL03 [8]
 Auth Id    : user1 [9]
 Status     :   [23]
          Used       Used        Sum
  Segment 100% [10]  1822 [11]  1822 [12]
  Segment Over  : N [13] Last Segment :       1822/     11000 [14]
  Lobmap  Over  : N [15]
```

[説明]

処理対象 RD エリアが複数ある場合，次の順序で表示されます。

- RD エリア名（昇順）

- 表名（昇順）
- インデクス名（昇順）

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒
2. 解析対象の RD エリアの名称を表示します。
3. RD エリアを管理するサーバの名称を表示します。
4. RD エリアの総セグメント数（使用中ページ数＋未使用セグメント数）を表示します。
5. RD エリアの未使用セグメント数の合計を表示します。括弧内は該当する RD エリア内の，更新凍結状態でない HiRDB ファイルの未使用セグメント数の合計となります。
6. 1 セグメントのサイズをセグメント内のページ数で表示します。
7. 1 ページのサイズをバイト単位で表示します。
8. RD エリアにある表又はインデクスの名称を表示します。
9. 表の認可識別子を表示します。データディクショナリ LOB 用 RD エリアの場合は，(Data dictionary)と表示されます。
10. 使用中セグメント比率を表示します。
    ↑11 の値÷12 の値×100↑（%）
11. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
12. 表に割り当てられているセグメント数の合計（使用中セグメント数）を表示します。
13. LOB 用 RD エリアが乱れているかどうかを表示します。
    Y：乱れています。
    N：乱れていません。
14. 使用されているセグメントの最後を示す位置情報を表示します。最終位置/総セグメント数の形式で表示します。13 が Y のときは，常に最後のセグメントを示します。
15. LOB 管理エントリがすべて使用されているかどうかを表示します。
    Y：すべて使用されています。
    N：未使用のエントリが残っています。
16. 解析対象の RD エリアの名称を表示します。
17. 常に 0 を表示します。
18. 常に 0 を表示します。
19. 該当する RD エリアの閉塞履歴情報です。
    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）
    History2：History1 の一つ前の閉塞情報
20. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

CMD：HiRDB の障害検知によるコマンド閉塞

FLT：障害閉塞

21. 該当する RD エリアの閉塞要因コードを表示します。閉塞要因コードについては，「LOB 用 RD エリア以外の場合」を参照してください。
22. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。
23. 表の場合は，データ未完状態であるか，又は再編成実行中であるかを表示します。次のどちらでもない場合は，空白を表示します。

    DATA_UNFINISH：

    データ未完状態です。表の再編成を実行し，途中でエラーなどによってリロード処理が完了していない状態のことをいいます。データ未完状態を解除するためには，表の再編成を再実行するか，又は表へのリロードを実行する必要があります。

    ON_RORG：

    再編成実行中です。

    インデクスの場合は，インデクスの未完状態を表示します。未完状態でない場合は空白を表示します。

    UNFINISH_0：

    図「表単位のインデクス作成未完状態の種類（1/2）」，図「表単位のインデクス作成未完状態の種類（2/2）」を参照してください。

    UNFINISH_1：

    図「RD エリア単位のインデクス作成未完状態の種類（1/2）」，図「RD エリア単位のインデクス作成未完状態の種類（2/2）」を参照してください。

    UNFINISH_2：

    プラグインインデクス遅延更新中。プラグインインデクス遅延一括作成機能を使用した UAP を実行中，又は UAP 終了後に pdrorg のインデクス一括作成を実行していない状態です。プラグインインデクスの遅延一括作成については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(c)　DAT 形式の場合（-a オプションを指定した場合）

DAT 形式での出力規則を次に示します。

- DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。
- RD エリアごとに，その RD エリアにある表，インデクス，及び順序数生成子の情報をそれぞれ 1 行ずつ表示します。
- RD エリアの情報は，RD エリア名の昇順に表示します。また，表，インデクス，及び順序数生成子の情報は，それぞれ名称の昇順に表示します。

  RD エリア単位の状態解析（論理的解析）の DAT 形式出力内容を次の表に示します。

表 15–3　RD エリア単位の状態解析（論理的解析）の DAT 形式出力内容

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB 用 RD エリア以外			LOB 用 RD エリア	
				表	インデクス	順序数生成子	表	インデクス
VERSION	HiRDB のバージョン	文字列(vv-rr)	5	◎	◎	◎	◎	◎
INF_GET_TIME	情報取得が完了した年月日時刻	文字列(yyyy/mm/dd hh:mm:ss)	19	◎	◎	◎	◎	◎
RDAREA_NAME	対象となる RD エリア名	文字列	30	◎	◎	◎	◎	◎
SERVER	RD エリアがあるサーバ名	文字列	8	◎	◎	◎	◎	◎
TOTAL_SEGMENT	RD エリア内の総セグメント数	数値	10	◎	◎	◎	◎	◎
UNUSED_SEGMENT	RD エリア内の未使用セグメント数	数値	10	◎	◎	◎	◎	◎
UNUSED_SEGMENT_NOT_FREEZE	RD エリア内の更新凍結状態でない未使用セグメント数	数値	10	−	−	−	◎	◎
SEGMENT_SIZE	セグメントサイズ	数値	10	◎	◎	◎	◎	◎
PAGE_SIZE	ページサイズ	数値	10	◎	◎	◎	◎	◎
ORIGINAL_RDAREA_NAME	常に空要素	文字列	30	◎	◎	◎	◎	◎
GENERATION_NUMBER	常に 0	数値	2	◎	◎	◎	◎	◎
REPLICA_RDAREA_COUNT	常に 0	数値	2	◎	◎	◎	◎	◎
HOLD_STATUS(HISTORY1)	RD エリアの閉塞種別（閉塞解除する一つ前の閉塞種別（現在閉塞中であれば現在の閉塞種別）） "CMD"：HiRDB の障害検知によるコマンド閉塞 "FLT"：障害閉塞	文字列	3	◎	◎	◎	◎	◎
HOLD_CODE(HISTORY1)	RD エリアの閉塞要因コード（閉塞解除する一つ前の閉塞要因コード（現在閉塞中であれば現在の閉塞要因コード））	数値	5	◎	◎	◎	◎	◎
HOLD_TIME(HISTORY1)	RD エリアの閉塞時刻（閉塞解除する一つ前の閉塞時刻（現在閉塞中であれば現在の閉塞時刻））	文字列(yyyy/mm/	19	◎	◎	◎	◎	◎

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB用RDエリア以外			LOB用RDエリア	
				表	インデクス	順序数生成子	表	インデクス
		dd hh:mm:ss)						
HOLD_STATUS(HISTORY2)	RDエリアの閉塞種別（History1の一つ前の閉塞種別） "CMD"：HiRDBの障害検知によるコマンド閉塞 "FLT"：障害閉塞	文字列	3	◎	◎	◎	◎	◎
HOLD_CODE(HISTORY2)	RDエリアの閉塞要因コード（History1の一つ前の閉塞要因コード）	数値	5	◎	◎	◎	◎	◎
HOLD_TIME(HISTORY2)	RDエリアの閉塞時刻（History1の一つ前の閉塞時刻）	文字列(yyyy/mm/dd hh:mm:ss)	19	◎	◎	◎	◎	◎
INF_KIND	表示情報の種別 "TBL"：RDエリア内の表情報 "IDX"：RDエリア内のインデクス情報 "SEQ"：RDエリア内の順序数生成子情報 "LBT"：LOB用RDエリア内の表情報 "LBX"：LOB用RDエリア内のインデクス情報	文字列	3	◎	◎	◎	◎	◎
TABLE_NAME	表識別子	文字列	30	◎	－	－	◎	－
INDEX_NAME	インデクス識別子	文字列	30	－	◎	－	－	◎
SEQUENCE_NAME	順序数生成子識別子	文字列	30	－	－	◎	－	－
AUTH_ID	表，インデクス，又は順序数生成子の認可識別子（データディクショナリ用RDエリアの場合は(Data dictionary)）	文字列	30	◎	◎	◎	◎	◎
STATUS	**表の場合：** 次のどれかのステータスとなります。 TA_UNFINISH：データ未完状態	文字列	13	◎	◎	－	◎	◎

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB 用 RD エリア以外			LOB 用 RD エリア	
				表	インデクス	順序数生成子	表	インデクス
	N_RORG：再編成中 空要素：上記以外 **インデクスの場合：** 次のどれかのステータスとなります。 UNFINISH_0，UNFINISH_1，又は UNFINISH_2：詳細は「LOB 用 RD エリア以外の場合」の status を参照 空要素：上記以外							
REFERENCE_PENDING_STATUS	RD エリアの表情報で管理している，参照制約の検査保留状態 P：検査保留状態 空白：非検査保留状態（表に参照制約の定義がない場合も含まれます）	文字列	1	◎	−	−	−	−
CHECK_PENDING_STATUS	RD エリアの表情報で管理している，検査制約の検査保留状態 P：検査保留状態 空白：非検査保留状態（表に検査制約の定義がない場合も含まれます）	文字列	1	◎	−	−	−	−
SEGMENT_REUSE	CREATE TABLE 実行時，又は ALTER TABLE 実行時の SEGMENT REUSE で指定したセグメント数 "0"：SEGMENT REUSE でセグメント数を指定していない場合 空要素：SEGMENT REUSE に NO を指定している場合，及び SEGMENT REUSE を省略している場合	数値	10	◎	−	−	−	−
SEARCH_MODE	ページサーチモード INS：新規ページ追加モード REU：空きページ再利用モード	文字列	3	◎	−	−	−	−
REUSE_SEARCH_FAILURE	新規ページ追加モードから空きページ再利用モードに切り替えたときに，再利用できる空き領域がなく，新規ページ追加モードに戻した回数（次の契機で 0 にリセットされます） ・RD エリアのクローズ時	数値	10	◎	−	−	−	−

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB 用 RD エリア以外			LOB 用 RD エリア	
				表	インデクス	順序数生成子	表	インデクス
	・HiRDB の再開始時 ・PURGE TABLE 文実行時							
REUSE_SEARCH	新規ページ追加モードから空きページ再利用モードに切り替えた回数（次の契機で 0 にリセットされます） ・RD エリアのクローズ時 ・HiRDB の再開始時 ・PURGE TABLE 文実行時	数値	10	◎	－	－	－	－
USED_SEGMENT	使用中セグメント数	数値	10	◎	◎	－	◎	◎
USED_SEGMENT_RATIO	使用中セグメント比率（単位：%） ↑(使用中セグメント数÷RD エリア内の総セグメント数)×100↑	数値	3	◎	◎	－	◎	◎
FULL_SEGMENT	満杯セグメント数	数値	10	◎	◎	－	－	－
FULL_SEGMENT_RATIO	満杯セグメント比率（単位：%） ↑(満杯セグメント数÷RD エリア内の総セグメント数)×100↑	数値	3	◎	◎	－	－	－
SUM_SEGMENT	RD エリア内の総セグメント数	数値	10	◎	◎	－	◎	◎
SEGMENT_OVER	全セグメントが使用中，又は使用されたかどうかを示す情報 "Y"：使用されています "N"：使用されていません	文字列	1	－	－	－	◎	◎
LAST_SEGMENT_IN_USE	使用中セグメントの最終位置（SEGMENT_OVER が Y の場合は最後のセグメント）	数値	10	－	－	－	◎	◎
TOTAL_SEGMENT_NUMBER	総セグメント数	数値	10	－	－	－	◎	◎
LOBMAP_OVER	LOB 管理エントリがすべて使用されたかどうかを示す情報 "Y"：使用されています "N"：使用されていません	文字列	1	－	－	－	◎	◎
USED_PAGE	使用中ページ数	数値	10	◎	◎	－	－	－
USED_PAGE_RATIO	使用中ページ比率（単位：%）	数値	3	◎	◎	－	－	－

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB用RDエリア以外			LOB用RDエリア	
				表	インデクス	順序数生成子	表	インデクス
	↑(使用中ページ数÷RDエリア内の総セグメント中のページ数の合計)×100↑							
FULL_PAGE	満杯ページ数	数値	10	◎	◎	−	−	−
FULL_PAGE_RATIO	満杯ページ比率（単位：%） ↑(満杯ページ数÷RDエリア内の総セグメント中のページ数の合計)×100↑	数値	3	◎	◎	−	−	−
SUM_PAGE	RDエリア内の総セグメント中のページ数の合計	数値	10	◎	◎	−	−	−
COLLECT_ON_SEGMENT	pdreclaimで解放途中のセグメント数	数値	10	◎	◎	−	−	−
COLLECT_PREARRANGED_PAGE	pdreclaimで解放できるページ数	数値	10	○	○	−	−	−
COLLECT_ON_PAGE	pdreclaimで解放途中のページ数	数値	10	−	○	−	−	−
RATIO0_PAGE	使用率0%のページ数	数値	10	○	○	−	−	−
RATIO0_PAGE_RATIO	使用率0%のページ比率（単位：%）	数値	3	○	○	−	−	−
RATIO0_DEL_ROWID	使用率0%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO1_PAGE	使用率1-10%のページ数	数値	10	○	○	−	−	−
RATIO1_PAGE_RATIO	使用率1-10%のページ比率（単位：%）	数値	3	○	○	−	−	−
RATIO1_DEL_ROWID	使用率1-10%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO2_PAGE	使用率11-20%のページ数	数値	10	○	○	−	−	−
RATIO2_PAGE_RATIO	使用率11-20%のページ比率（単位：%）	数値	3	○	○	−	−	−

タイトルバー	内容	出力形式	最大長(単位：バイト) ※1	LOB用RDエリア以外			LOB用RDエリア	
				表	インデクス	順序数生成子	表	インデクス
RATIO2_DEL_ROWID	使用率11-20%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO3_PAGE	使用率21-30%のページ数	数値	10	○	○	−	−	−
RATIO3_PAGE_RATIO	使用率21-30%のページ比率（単位：%）	数値	3	○	○	−	−	−
RATIO3_DEL_ROWID	使用率21-30%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO4_PAGE	使用率31-40%のページ数	数値	10	○	○	−	−	−
RATIO4_PAGE_RATIO	使用率31-40%のページ比率（単位：%）	数値	3	○	○	−	−	−
RATIO4_DEL_ROWID	使用率31-40%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO5_PAGE	使用率41-50%のページ数	数値	10	○	○	−	−	−
RATIO5_PAGE_RATIO	使用率41-50%のページ比率（単位：%）	数値	3	○	○	−	−	−
RATIO5_DEL_ROWID	使用率41-50%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO6_PAGE	使用率51-60%のページ数	数値	10	○	○	−	−	−
RATIO6_PAGE_RATIO	使用率51-60%のページ比率（単位：%）	数値	3	○	○	−	−	−
RATIO6_DEL_ROWID	使用率51-60%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	−	△	−	−	−
RATIO7_PAGE	使用率61-70%のページ数	数値	10	○	○	−	−	−
RATIO7_PAGE_RATIO	使用率61-70%のページ比率（単位：%）	数値	3	○	○	−	−	−

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB用RDエリア以外			LOB用RDエリア	
				表	インデクス	順序数生成子	表	インデクス
RATIO7_DEL_ROWID	使用率61-70%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	－	△	－	－	－
RATIO8_PAGE	使用率71-80%のページ数	数値	10	○	○	－	－	－
RATIO8_PAGE_RATIO	使用率71-80%のページ比率（単位：%）	数値	3	○	○	－	－	－
RATIO8_DEL_ROWID	使用率71-80%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	－	△	－	－	－
RATIO9_PAGE	使用率81-90%のページ数	数値	10	○	○	－	－	－
RATIO9_PAGE_RATIO	使用率81-90%のページ比率（単位：%）	数値	3	○	○	－	－	－
RATIO9_DEL_ROWID	使用率81-90%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	－	△	－	－	－
RATIO10_PAGE	使用率91-100%のページ数	数値	10	○	○	－	－	－
RATIO10_PAGE_RATIO	使用率91-100%のページ比率（単位：%）	数値	3	○	○	－	－	－
RATIO10_DEL_ROWID	使用率91-100%のページの，残存エントリが管理する行数	数値	仮数部5，指数部5	－	△	－	－	－
TOTAL_PAGE	使用中ページの比率別ページ数の合計	数値	10	○	○	－	－	－
TOTAL_DEL_ROWID	残存エントリが管理する行数の合計	数値	仮数部5，指数部5	－	△	－	－	－
BINARY_SEARCH_MODE	バイナリ専用セグメントでのページサーチモード INS：新規ページ追加モード REU：空きページ再利用モード	文字列	3	◎	－	－	－	－
BINARY_REUSE_SEARCH_FAILURE	バイナリ専用セグメントで新規ページ追加モードから空きページ再利用	数値	10	◎	－	－	－	－

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB 用 RD エリア以外			LOB 用 RD エリア	
				表	インデクス	順序数生成子	表	インデクス
	モードに切り替えたときに，再利用できる空き領域がなく，新規ページ追加モードに戻した回数（次の契機で 0 にリセットされます） ・RD エリアのクローズ時 ・HiRDB の再開始時 ・PURGE TABLE 文実行時							
BINARY_REUSE_SEARCH	バイナリ専用セグメントで新規ページ追加モードから空きページ再利用モードに切り替えた回数（次の契機で 0 にリセットされます） ・RD エリアのクローズ時 ・HiRDB の再開始時 ・PURGE TABLE 文実行時	数値	10	◎	－	－	－	－
BINARY_USED_SEGMENT	使用中バイナリ専用セグメント数	数値	10	◎	－	－	－	－
BINARY_USED_SEGMENT_RATIO	使用中バイナリ専用セグメント数の比率（単位：%） ↑(使用中バイナリ専用セグメント数÷表に割り当てられているバイナリ専用セグメント数の合計)×100↑	数値	3	◎	－	－	－	－
BINARY_FULL_SEGMENT	満杯バイナリ専用セグメント数	数値	10	◎	－	－	－	－
BINARY_FULL_SEGMENT_RATIO	満杯バイナリ専用セグメント数の比率（単位：%） ↑(満杯バイナリ専用セグメント数÷表に割り当てられているバイナリ専用セグメント数の合計)×100↑	数値	3	◎	－	－	－	－
BINARY_SUM_SEGMENT	表に割り当てられているバイナリ専用セグメント数の合計	数値	10	◎	－	－	－	－
BINARY_USED_PAGE	バイナリ専用セグメントの使用中ページ数	数値	10	◎	－	－	－	－
BINARY_USED_PAGE_RATIO	バイナリ専用セグメントの使用中ページ数の比率（単位：%） ↑(バイナリ専用セグメントの使用中ページ数÷表に割り当てられている	数値	3	◎	－	－	－	－

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB用RDエリア以外			LOB用RDエリア	
				表	インデクス	順序数生成子	表	インデクス
	バイナリ専用セグメントのページ数の合計)×100↑							
BINARY_FULL_PAGE	バイナリ専用セグメントの満杯ページ数	数値	10	◎	−	−	−	−
BINARY_FULL_PAGE_RATIO	バイナリ専用セグメントの満杯ページ数の比率（単位：%） ↑(使用中バイナリ専用セグメントの満杯ページ数÷表に割り当てられているバイナリ専用セグメントのページ数の合計)×100↑	数値	3	◎	−	−	−	−
BINARY_SUM_PAGE	表に割り当てられているバイナリ専用セグメントのページ数の合計	数値	10	◎	−	−	−	−
BINARY_COLLECT_ON_SEGMENT	pdreclaim で解放途中のバイナリ専用セグメント数	数値	10	◎	−	−	−	−
BINARY_COLLECT_PREARRANGED_PAGE	pdreclaim で解放できるバイナリ専用セグメントのページ数	数値	10	○	−	−	−	−
BINARY_RATIO0_PAGE	バイナリ専用セグメントの使用率 0%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO0_PAGE_RATIO	バイナリ専用セグメントの使用率 0%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO1_PAGE	バイナリ専用セグメントの使用率 1-10%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO1_PAGE_RATIO	バイナリ専用セグメントの使用率 1-10%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO2_PAGE	バイナリ専用セグメントの使用率 11-20%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO2_PAGE_RATIO	バイナリ専用セグメントの使用率 11-20%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO3_PAGE	バイナリ専用セグメントの使用率 21-30%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO3_PAGE_RATIO	バイナリ専用セグメントの使用率 21-30%のページ比率（単位：%）	数値	3	○	−	−	−	−

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB用RDエリア以外			LOB用RDエリア	
				表	インデクス	順序数生成子	表	インデクス
BINARY_RATIO4_PAGE	バイナリ専用セグメントの使用率31-40%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO4_PAGE_RATIO	バイナリ専用セグメントの使用率31-40%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO5_PAGE	バイナリ専用セグメントの使用率41-50%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO5_PAGE_RATIO	バイナリ専用セグメントの使用率41-50%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO6_PAGE	バイナリ専用セグメントの使用率51-60%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO6_PAGE_RATIO	バイナリ専用セグメントの使用率51-60%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO7_PAGE	バイナリ専用セグメントの使用率61-70%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO7_PAGE_RATIO	バイナリ専用セグメントの使用率61-70%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO8_PAGE	バイナリ専用セグメントの使用率71-80%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO8_PAGE_RATIO	バイナリ専用セグメントの使用率71-80%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO9_PAGE	バイナリ専用セグメントの使用率81-90%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO9_PAGE_RATIO	バイナリ専用セグメントの使用率81-90%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_RATIO10_PAGE	バイナリ専用セグメントの使用率91-100%のページ数	数値	10	○	−	−	−	−
BINARY_RATIO10_PAGE_RATIO	バイナリ専用セグメントの使用率91-100%のページ比率（単位：%）	数値	3	○	−	−	−	−
BINARY_TOTAL_PAGE	バイナリ専用セグメントの使用中ページ数の比率別ページ数合計	数値	10	○	−	−	−	−

タイトルバー	内容	出力形式	最大長(単位：バイト)※1	LOB 用 RD エリア以外			LOB 用 RD エリア	
				表	インデクス	順序数生成子	表	インデクス
DEL_ROWID	表又はインデクスの削除状態の管理行数の合計	数値	仮数部 16，指数部 5	△	△	−	−	−
DEL_ROWID_RATIO	残存エントリが管理する行数の比率（単位：%）	数値	3	−	△	−	−	−
ROWID_COUNT	インデクスが管理する総行数	数値	仮数部 16，指数部 5	−	△	−	−	−
SEQUENCE_CURRENT_VALUE	順序数生成子の現在値	数値	39	−	−	◎	−	−
SEQUENCE_CURRENT_VALUE_REVICE	順序数生成子の現在値の補正要否	数値	1	−	−	◎	−	−
SEGMENT_REUSE_OPTION※2	表定義の SEGMENT REUSE で指定した再利用オプション値	数値	10	◎	−	−	−	−
BRANCH_ROW_SEARCH_MODE※3	分岐行に対する空き領域サーチモード	文字列	3	◎	−	−	−	−
COLLECT_PREARRANGED_FULL_PAGE	pdreclaim で解放できる満杯ページ数	数値	10	○	−	−	−	−
BINARY_BRANCH_ROW_SEARCH_MODE※3	バイナリ専用セグメントの分岐行に対する空き領域サーチモード	文字列	3	◎	−	−	−	−
BINARY_COLLECT_PREARRANGED_FULL_PAGE	バイナリ専用セグメントの pdreclaim で解放できる満杯ページ数	数値	10	○	−	−	−	−

（凡例）

◎：-d オプションの指定に関係なく出力されます（出力データがない場合は空要素となります）。

○：-d オプションを指定した場合に出力されます（出力データがない場合は空要素となります）。

△：-d 及び-z オプションを指定した場合に出力されます（出力データがない場合は空要素となります）。

−：空要素が出力されます。

注※1

出力形式が文字列の場合，" "で囲んで表示します。最大長の値には，" "を含みません。

注※2

表定義の SEGMENT REUSE の指定によって再利用オプション値の出力データが異なります。SEGMENT REUSE の指定値ごとの出力データを次に示します。

項番	SEGMENT REUSE の指定及び指定値		SEGMENT REUSE OPTION の指定	出力するデータ
1	あり	NO 以外	あり	指定した値
2			なし	0
3		NO	−	空データ
4	なし	−	−	空データ

（凡例）

−：該当しません。

注※3

表定義の SEGMENT REUSE の指定によって空きページサーチモードの出力データが異なります。SEGMENT REUSE の指定値ごとの出力データを次に示します。

項番	SEGMENT REUSE の指定及び指定値		SEGMENT REUSE OPTION の指定値	出力するデータ
1	あり	NO 以外	2,3	• INS：新規ページ追加モード • REU：空ページ再利用モード
2			2,3 以外	空データ
3		NO	−	空データ
4	なし	−	−	空データ

（凡例）

−：該当しません。

(3)　解析結果の分析

RD エリア単位の状態解析（論理的解析）の解析結果は，次の表のように分析できます。

表 15–4　RD エリア単位（論理的解析）の解析結果の分析

RD エリア内の状態	内容
表の総未使用ページ比率が，定義系 SQL の CREATE TABLE で指定したセグメント内空きページ比率よりも小さい	表のデータが，次のどちらかの状態であることが考えられます。 • データを繰り返して追加したため，未使用ページが少なくなっています。 • データの追加，削除，更新を繰り返したため，使用中空きページが散在しています。 このような場合には，必要に応じて，表が格納されている RD エリアを拡張し，その後表を再編成することで，表の乱れを修正できます。※1※2

RD エリア内の状態			内容
表の格納行数から算出したインデクスの格納ページ数より，解析結果として表示されたインデクスの総使用中ページ数が多い			インデクスが次の状態であることが考えられます。 • 特定範囲の行を大量に削除したため，それに対応したインデクスも大量に削除され，使用中空きページとしてそのまま残っています。 このような場合には，インデクスの再編成をすることでインデクスの乱れを修正できます。また，インデクスの空きページ解放をすることで使用ページ数を適正化できます。インデクスの格納ページ数の計算方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。※1
使用中ページの比率別ページ数	使用率が 0%のページが多い	使用中空きページが総ページ数の30%以上	使用中空きページが多く残っていると，性能劣化の要因になります。使用中空きページの解放が必要です。
		同一 RD エリアにほかの表又はインデクスを定義している	未使用ページは，そのページを含むセグメントを確保した表又はインデクス以外は使用できないため，同一 RD エリアにほかの表又はインデクスを定義している場合，それらの資源に割り当て可能なページ数が減っている状態です。使用中空きセグメントの解放，又は再編成が必要です。なお，未使用ページ数は，使用率が 0%のページ数と使用中空きページ数の差です。
	使用率が低いページが多い		データの削除や更新で一度割り当てられた領域が未解放のまま空き領域として残っており，再使用できない状態になっています。再編成が必要です。
	使用率が高いページが多い		データの追加や更新時に割り当てるページ内の空き領域が少ない状態になっています。次のような処理をする場合，再編成が必要です。 • データ長を長くする更新を頻繁に行う場合 • クラスタキーを定義している表に挿入処理を頻繁に行う場合
インデクスが管理する総行数に対する，残存エントリが管理する行数の比率		高い	残存エントリが多くなっています。不当な排他待ちや，排他待ちによるデッドロック，さらに検索性能が劣化するため，使用中空きページの解放，又は再編成が必要です。 残存エントリが増えることによる影響については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
		低い	残存エントリが少ない場合でも，残存エントリが管理する行数が検索範囲に多いと，検索性能が劣化することがあります。検索実行時のページアクセス量が，検索の HIT 件数に比べて多い（性能が劣化）と判断した場合，空きページ解放，又は再編成を実行してください。ページアクセス量は，pdbufls コマンド，UAP 統計レポートなどで判断してください。
クラスタリングありの表	使用中空きページ	多い	CREATE TABLE のセグメント内空きページ比率が大き過ぎます。データの追加，削除，更新によって使用中空きページができているため，必要に応じて表の再編成をしてください。※1

RD エリア内の状態			内容
		少ない	データの追加がある場合は，表の再編成をした方がよいです。※1
クラスタリングなしの表	使用中空きページ	多い	CREATE TABLE のセグメント内空きページ比率が大き過ぎます。データの追加，削除，更新によって使用中空きページができているため，必要に応じて表の再編成をしてください。※1
		少ない	RD エリア内の未使用セグメントが少ない場合，必要に応じて RD エリアの拡張をしてください。※2
セグメントオーバーが Y の場合			LOB 列に対する検索性能が劣化することがあります。LOB 用 RD エリアを再編成してください。※1
Lobmap オーバーが Y の場合			LOB 列に対する検索性能が劣化することがあります。LOB 列構成基表を再編成してください。※1

注

プラグインが提供する抽象データ型の LOB 属性を格納する RD エリア，又はプラグインインデクスを格納する RD エリアの状態を解析した場合，RD エリア内の使用中セグメントの割合は，プラグインのセグメントの割り当て方に依存します。例えば，インデクス定義時にセグメントの割り当てを終えてしまうようなプラグインの場合は，定義直後（データをまだ格納していない状態）でもほとんどのセグメントが使用中セグメントになることがあります。

注※1

表の再編成，及びインデクスの再編成については，「データベース再編成ユティリティ（pdrorg）」を参照してください。インデクスの空きページ解放については，「空きページ解放ユティリティ（pdreclaim）」を参照してください。

注※2

RD エリアの拡張については，「データベース構成変更ユティリティ（pdmod）」を参照してください。

15.3.2　RD エリア単位の状態解析（物理的解析）

表やインデクスを意識しないで，RD エリア内の全セグメント，及び全ページの格納状態を解析します。

(1)　使用目的

一つの RD エリア内の全セグメント，及び全ページの格納状態が表示されます。表示された解析結果の情報から，RD エリアの使用効率，容量の状態が分かります。

(2)　解析結果の情報

RD エリア単位の状態解析（物理的解析）で表示される解析結果を次に示します。

- **テキスト形式の場合（-a オプションを省略した場合）**

```
 pddbst VV-RR(Object Option) ** RD Area Physical Analysis ** 2002/03/29 16:40:10 [1]
  Max Sum Segment :        150 [2]
  Max Sum Page    :        750 [3]
  RD Area Count   :          1/         1 [4]
--------------------------------------------------------------------------------
 RD Area Name   : USER_RDAREA_1 [5]
  Server        : BES01 [6]
  Original RD Area Name : USER_RDAREA_1 [30]
  Generation Number :  0[31] Replica RD Area Count :  0[32]
    [33]               [34]               [35]           [36]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
  Unused Segment:         98 [7]
  Segment Size  :          5 Pages[8] Page Size    :       4096 Bytes[9]
          Used(Full)       Used(      Full)        Sum
           [10] [11]        [12]       [13]       [14]
  Segment  35%(  0%)         52(         0)        150
           [15] [16]        [17]       [18]       [19]
  Page     35%(  0%)        256(         0)        750
  Auto Extend Use    : USE[20]   Auto Extend Segment :     4 [21]
  Auto Extend Status : NOSUP[22] Error Code          :     0 [23]
   HiRDB File Name : c:¥dbarea¥rdsys04¥u_rd20_1 [24]
    File Size      :        5 segments[25] Extent Count :   1/ 24 [26]  Expand :NOUSE [40
]
    Auto Extend Status : SUP     Error Code : -1535
   HiRDB File Name : c:¥dbarea¥rdsys04¥u_rd20_2
    File Size      :        6 segments     Extent Count :   1/ 24       Expand :NOUSE
    Auto Extend Status : NOSUP   Error Code :     0
   HiRDB File Name : c:¥dbarea¥rdsys04¥u_rd20_3
    File Size      :       18 segments     Extent Count :   1/ 24       Expand :NOUSE
    Auto Extend Status : NOSUP   Error Code :     0
 RD Area Name   : USER_RDLOB_1
  Server        : BES01
  Original RD Area Name : USER_RDLOB_1
  Generation Number :  0   Replica RD Area Count :  0
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
  Unused Segment:        770(       770)
  Segment Size  :          1 Pages   Page Size    :       8192 Bytes
          Used       Used        Sum
  Segment  93%      10230      11000
               [27]                          [28]
  Segment Over  : N  Last Segment :      10230/     11000
               [29]
  Lobmap  Over  : N
  Freeze Specified : Y [37]
  Auto Extend Use    : USE     Auto Extend Segment :    10
  Auto Extend Status : NOSUP   Error Code          :     0
   HiRDB File Name : c:¥dbarea¥rdsys04¥u101
    File Size      :       4000 segments   Extent Count :   1/ 24  Expand :NOUSE
    Segment        :       4000/      4000
    Freeze Status  : U[38] ****/**/**    **:**:** [39]
   HiRDB File Name : c:¥dbarea¥rdsys04¥u201
    File Size      :       5000 segments   Extent Count :   1/ 24  Expand :NOUSE
    Segment        :       5000/      5000
    Freeze Status  : F    2002/03/29    16:38:19
   HiRDB File Name : c:¥dbarea¥rdsys04¥u301
```

```
File Size      :       2000 segments   Extent Count :   1/ 24  Expand :NOUSE
Segment        :       1230/      2000
Freeze Status  : U    ****/**/**    **:**:**
```

[説明]

RD エリアの情報は，RD エリア名の昇順に表示します。また，HiRDB ファイルの情報は，物理ファイル ID 順に表示します。

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒
2. 解析した RD エリアの中で，最大の総セグメント数を表示します。
3. 解析した RD エリアの中で，最大の総ページ数を表示します。
4. 指定した RD エリア数／その中で正常に解析できた RD エリア数を表示します。
5. 解析した RD エリアの名称を表示します。
6. RD エリアを管理するサーバの名称を表示します。
7. RD エリア内の未使用セグメント数の合計を表示します。括弧内は該当する RD エリア内の，更新凍結状態でない HiRDB ファイルの未使用セグメント数の合計となります。
8. 1 セグメントのサイズをセグメント内のページ数で表示します。
9. 1 セグメントのサイズをページ単位で表示します。
10. 使用中セグメント数の比率を表示します。
    ↑使用中セグメント数÷14 の値×100↑（%）
11. 満杯セグメント数の比率を表示します。
    ↑満杯セグメント数÷14 の値×100↑（%）
12. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
13. 満杯セグメント数を表示します。
14. RD エリア内のセグメント数の合計（使用中セグメント数＋未使用セグメント数）を表示します。
15. 使用中ページ数の比率を表示します。
    ↑使用中ページ数÷19 の値×100↑（%）
16. 満杯ページ数の比率を表示します。
    ↑満杯ページ数÷19 の値×100↑（%）
17. 使用中ページ数を表示します。
18. 満杯ページ数を表示します。
19. RD エリア内のセグメント中のページ数の合計（使用中ページ数＋未使用ページ数）を表示します。
20. RD エリアの自動増分機能の適用有無を表示します。
    USE：RD エリアの自動増分機能を適用しています。

NOUSE：RD エリアの自動増分機能を適用していません。

最後の HiRDB File Name に表示されている HiRDB ファイルが，自動増分の対象となります。

21. 増分セグメント数を表示します。
22. 自動増分可否を表示します。

    SUP：抑止されている状態です。

    NOSUP：抑止されていない状態です。
23. 自動増分ができなかった場合のエラーコードを表示します。

    -1556 の場合：

    次のどちらかの対策をしてください。

    ・該当する HiRDB ファイルシステムがある HiRDB ファイルシステム領域のエクステントを，pdfbkup，pdfmkfs，及び pdfrstr コマンドで統合する。

    ・pdmod で RD エリアを拡張する。

    上記以外の場合：

    マニュアル「HiRDB メッセージ」の HiRDB ファイルシステムのエラーコード一覧を参照してください。
24. 該当する RD エリアの HiRDB ファイル名称を表示します。
25. HiRDB ファイルの総セグメント数を表示します。
26. HiRDB ファイルの分割格納数／分割格納上限数を表示します。
27. LOB 用 RD エリアが乱れているかどうかを表示します。

    Y ：乱れています。

    N ：乱れていません。
28. 使用されているセグメントの最後を示す位置情報を表示します。最終位置/総セグメント数の形式で表示します。27 が Y のときは，常に最後のセグメントを示します。
29. LOB 管理エントリがすべて使用されているかどうかを表示します。

    Y ：すべて使用されています。

    N ：未使用のエントリが残っています。
30. 解析した RD エリアの名称を表示します。
31. 常に 0 を表示します。
32. 常に 0 を表示します。
33. 該当する RD エリアの閉塞履歴情報です。

    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）

    History2：History1 の一つ前の閉塞情報
34. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

    CMD：HiRDB の障害検知によるコマンド閉塞

    FLT：障害閉塞

35. 該当する RD エリアの閉塞要因コードを表示します。閉塞要因コードについては，「LOB 用 RD エリア以外の場合」を参照してください。
36. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。
37. RD エリアが更新凍結されているかどうかを表示します。

    Y：pddbfrz コマンドで更新凍結されています。

    N：pddbfrz コマンドで更新凍結されていません。
38. RD エリアを構成する HiRDB ファイルが更新凍結されているかどうかを表示します。

    U：

    更新凍結されていません。バックアップ取得時には，バックアップ対象にする必要があります。

    F：

    更新凍結されています。表示された日時以降にバックアップを取得している場合は，新たにバックアップを取得する必要はありません。
39. 更新凍結状態の変更日時です。

    ******/**/** **:**:**の場合：**

    該当する RD エリアを作成又は再初期化してから，一度も更新凍結していない場合に表示されます。

    上記以外の場合：

    更新凍結状態を設定又は解除した日時が表示されます。
40. RD エリアの自動増分時に HiRDB ファイルシステム領域内の空き領域がなくなった場合，HiRDB ファイルシステム領域を自動的に拡張するかどうかを表示します。

    USE：自動増分機能を使用しています。

    NOUSE：自動増分機能を使用していません。

注 1

10，11，及び 15～19 は，LOB 用 RD エリアの場合は表示しません。

注 2

27～29，及び 37 は，LOB 用 RD エリアの場合に表示します。

- **DAT 形式の場合（-a オプションを指定した場合）**

  RD エリア単位の状態解析（物理的解析）の DAT 形式出力内容を次の表に示します。

  DAT 形式での出力規則を次に示します。

  - DAT 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。
  - 1 行には，1RD エリアの情報を表示します。ただし，1RD エリアが複数の HiRDB ファイルで構成される場合は，その HiRDB ファイル数分の行で表示します。
  - RD エリアの情報は，RD エリア名の昇順に表示します。また，HiRDB ファイルの情報は，物理ファイル ID 順に表示します。

表 15–5　RD エリア単位の状態解析（物理的解析）の DAT 形式出力内容

タイトルバー※1	内容	出力形式	最大長(単位：バイト)※2
VERSION	HiRDB のバージョン	文字列(vv-rr)	5
INF_GET_TIME	情報取得が完了した年月日時刻	文字列(yyyy/mm/dd hh:mm:ss)	19
RDAREA_NAME	対象となる RD エリア名	文字列	30
SERVER	RD エリアがあるサーバ名	文字列	8
MAX_SUM_SEGMENT	RD エリアの最大総セグメント数	数値	10
MAX_SUM_PAGE	RD エリアの最大総ページ数	数値	10
SPECIFIED_RDAREA_COUNT	解析対象ＲＤエリア数	数値	10
ANALYZED_RDAREA_COUNT	正常に解析できた RD エリアの数	数値	10
SEGMENT_SIZE	セグメント内ページ数	数値	10
PAGE_SIZE	ページサイズ	数値	10
HOLD_STATUS(HISTORY1)※5	RD エリアの閉塞種別（閉塞解除する一つ前の閉塞種別，又は現在閉塞中の場合は現在の閉塞種別） "CMD"：HiRDB の障害検知によるコマンド閉塞 "FLT"：障害閉塞	文字列	3
HOLD_CODE(HISTORY1)	RD エリアの閉塞要因コード（閉塞解除する一つ前の閉塞要因コード，又は現在閉塞中の場合は現在の閉塞要因コード）。 閉塞要因コードについては，テキスト形式の場合の説明を参照してください。	数値	5
HOLD_TIME(HISTORY1)※5	RD エリアの閉塞時刻（閉塞解除する一つ前の閉塞時刻，又は現在閉塞中の場合は現在の閉塞時刻）	文字列(yyyy/mm/dd hh:mm:ss)	19
HOLD_STATUS(HISTORY2)※5	RD エリアの閉塞種別（History1 の一つ前の閉塞種別） "CMD"：コマンド閉塞 "FLT"：障害閉塞	文字列	3
HOLD_CODE(HISTORY2)	RD エリアの閉塞要因コード（History1 の一つ前の閉塞要因コード）。 閉塞要因コードについては，テキスト形式の場合の説明を参照してください。	数値	5
HOLD_TIME(HISTORY2)※5	RD エリアの閉塞時刻（History1 の一つ前の閉塞時刻）	文字列(yyyy/mm/dd hh:mm:ss)	19
FREEZE_SPECIFIED※3	RD エリアの更新凍結指定の有無 "Y"：pddbfrz コマンドでの更新凍結指定あり	文字列	1

タイトルバー※1	内容	出力形式	最大長(単位：バイト)※2
	"N"：pddbfrz コマンドでの更新凍結指定なし		
UNUSED_SEGMENT	RD エリア内の未割り当てセグメント数の合計	数値	10
UNUSED_SEGMENT_NOT_FREEZE※3	RD エリア内の更新凍結状態でない HiRDB ファイルの未割り当てセグメント数の合計	数値	10
USED_SEGMENT	使用中セグメント数	数値	10
USED_SEGMENT_RATIO	使用中セグメント比率（単位：%） ↑(使用中セグメント数÷RD エリア内の総セグメント数)×100↑	数値	3
FULL_SEGMENT	満杯セグメント数	数値	10
FULL_SEGMENT_RATIO	満杯セグメント比率（単位：%） ↑(満杯セグメント数÷RD エリア内の総セグメント数)×100↑	数値	3
SUM_SEGMENT	RD エリア内の総セグメント数	数値	10
SEGMENT_OVER※3	全セグメントが使用中，又は使用されたかどうかを示す情報 "Y"：使用されています。 "N"：使用されていません。	文字列	1
LAST_SEGMENT_IN_USE※3	使用中セグメントの最終位置（セグメントオーバーが Y の場合は最後のセグメント）	数値	10
TOTAL_SEGMENT_ NUMBER※3	総セグメント数	数値	10
LOBMAP_OVER※3	LOB 管理エントリがすべて使用されたかどうかを示す情報 "Y"：使用されています。 "N"：使用されていません。	文字列	1
USED_LOB_ MANAGEMENT_ENTRY_NUMBER	システムが使用する情報	数値	10
TOTAL_LOB_ MANAGEMENT_ENTRY_NUMBER	システムが使用する情報	数値	10
USED_PAGE	使用中ページ数	数値	10
USED_PAGE_RATIO	使用中ページ比率（単位：%） ↑(使用中ページ数÷RD エリア内の総セグメント中のページ数の合計)×100↑	数値	3
FULL_PAGE	満杯ページ数	数値	10

タイトルバー※1	内容	出力形式	最大長(単位：バイト)※2
FULL_PAGE_RATIO	満杯ページ比率（単位：%） ↑(満杯ページ数÷RD エリア内の総セグメント中のページ数の合計)×100↑	数値	3
SUM_PAGE	RD エリア内の総セグメント中のページ数の合計	数値	10
AUTO_EXTEND_USE	自動増分機能の使用状況 "USE"：使用しています。 "NOUSE"：使用していません。 HiRDB ファイルの情報の最後に表示した HiRDB ファイルが，自動増分機能を使用するファイルとなります。	文字列	5
AUTO_EXTEND_SEGMENT	増分セグメント数	数値	5
AUTO_EXTEND_STATUS	自動増分機能の抑止状態 • 自動増分対象ファイルの指示方法（システム定義の pd_rdarea_extension_file オペランドの値）が"last"の場合 "SUP"：抑止されています。 "NOSUP"：抑止されていません。 • 自動増分対象ファイルの指示方法（システム定義の pd_rdarea_extension_file オペランドの値）が"all"の場合※6	文字列	5
ERROR_CODE	自動増分ができなかった場合のエラーコード※7	数値	5
HiRDB_FILE_NAME	RD エリアの HiRDB ファイル名（絶対パス名）	文字列	167
FILE_SIZE	HiRDB ファイルのサイズ（総セグメント数）	数値	10
EXTENT_COUNT	HiRDB ファイルの分割格納数	数値	3
MAX_EXTENT_COUNT	HiRDB ファイルの分割格納上限数	数値	3
USED_SEGMENT_IN_FILE※3	HiRDB ファイルの使用中セグメント数	数値	10
TOTAL_SEGMENT_IN_FILE※3	HiRDB ファイルの総セグメント数	数値	10
FREEZE_STATUS※3	HiRDB ファイルの更新凍結状態 "U"：更新凍結されていません。バックアップ取得時には，バックアップ対象にする必要があります。 "F"：更新凍結されています。表示された日時以降にバックアップを取得している場合は，新たに取得する必要はありません。	文字列	1
FREEZE_TIME※3	更新凍結状態の変更日時	文字列(yyyy/mm/dd hh:mm:ss)	19

タイトルバー※1	内容	出力形式	最大長(単位：バイト)※2
	"****/**/** **:**:**"の場合：RD エリアを作成，又は再初期化して以降，一度も更新凍結状態にしていません。 上記以外の場合：更新凍結状態を設定，又は解除した日時です。		
ORIGINAL_RDAREA_NAME※4	対象となる RD エリア名	文字列	30
GENERATION_NUMBER※4	常に 0 を表示	数値	2
REPLICA_RDAREA_COUNT※4	常に 0 を表示	数値	2
AREA_AUTO_EXPAND	RD エリアの自動増分時に HiRDB ファイルシステム領域内の空き領域がなくなった場合，HiRDB ファイルシステム領域を自動的に拡張するかどうかを示す情報 "USE"：使用しています。 "NOUSE"：使用していません。	文字列	5

注※1

タイトルバーは，-h オプションを指定した場合に表示します。

注※2

出力形式が文字列の場合，""で囲んで表示します。最大長の値には，""を含みません。出力形式が文字列以外の場合は，""で囲まないで表示します。

注※3

LOB 用 RD エリアの場合だけ表示します。LOB 用 RD エリアでない場合，データを出力しません（例：...,,...）。

注※4

データを出力しません（例：...,,...）。

注※5

一度も閉塞されていない場合は，データを出力しません（例：...,,...）。

注※6

自動増分抑止状態の出力値を次に示します。

システム定義の pd_rdarea_extension_file オペランドの指定値	RD エリアの状態	ファイルの状態	出力値
"all"	抑止されている	抑止されている	"RSFS"
		抑止されていない	"RSFN"
	抑止されていない	抑止されている	"RNFS"

システム定義の pd_rdarea_extension_file オペランドの指定値	RD エリアの状態	ファイルの状態	出力値
		抑止されていない	"RNFN"
"last"	抑止されている	−	"SUP"
	抑止されていない	−	"NOSUP"

（凡例）

−：該当しません。

注※7

自動増分エラーコードの出力値を次に示します。

システム定義の pd_rdarea_extension_file オペランドの指定値	出力値
"all"	HiRDB ファイルごとの値を出力します。
"last"	最後に割り当てた，HiRDB ファイルの情報を出力します。

(3)　解析結果の分析

RD エリア単位の状態解析（物理的解析）の解析結果は，次の表のように分析できます。

表 15–6　RD エリア単位（物理的解析）の解析結果の分析

RD エリア内の状態					内容
RD エリア全体	未使用セグメント	多い			容量に余裕があります。
		少ない	使用中空きページ	多い	RD エリアが効率的に使用されていません。RD エリア単位の状態解析（論理的解析）を実行し，使用中セグメント数が多く，未使用ページ数が多い表又はインデクスがないかを調査し，必要に応じて表若しくはインデクスの再編成，又は空きページの解放をしてください。※1
				少ない	容量が不足しています。RD エリアを拡張してください。※2

注※1

表及びインデクスの再編成については，「データベース再編成ユティリティ（pdrorg）」を参照してください。表及びインデクスの空きページ解放については，「空きページ解放ユティリティ（pdreclaim）」を参照してください。

注※2

RD エリアの拡張については，「データベース構成変更ユティリティ（pdmod）」を参照してください。

ユーザ LOB 用 RD エリアの場合は，次のように分析できます。

- RD エリア内のセグメントの情報

使用中セグメント比率	セグメントオーバー	内容
大きい（未使用セグメントが少ない）	Y	容量が不足していて，DB へのアクセス性能が劣化しています。ユーザ LOB 用 RD エリアを拡張して，表又はインデクスの再編成をしてください。※1※2
	N	容量が不足しています。ユーザ LOB 用 RD エリアを拡張する必要があります。※1
少ない（未使用セグメントが多い）	Y	DB へのアクセス性能が劣化しています。表又はインデクスの再編成をしてください。※2
	N	対策する必要はありません。

注

プラグインが提供する抽象データ型の LOB 属性を格納する RD エリア，又はプラグインインデクスを格納する RD エリアの状態を解析した場合，RD エリア内の使用中セグメントの割合は，プラグインのセグメントの割り当て方に依存します。例えば，インデクス定義時にセグメントの割り当てを終えてしまうようなプラグインの場合は，定義直後（データをまだ格納していない状態）でもほとんどのセグメントが使用中セグメントになることがあります。

注※1

RD エリアの拡張については，「データベース構成変更ユティリティ（pdmod）」を参照してください。

注※2

表及びインデクスの再編成については，「データベース再編成ユティリティ（pdrorg）」を参照してください。

- Lobmap オーバー

  Y の場合，LOB 列に対する検索性能が劣化することがあります。LOB 列構成基表を再編成してください。

15.3.3　表単位又はインデクス単位の状態解析

表又はインデクスの，格納ページのセグメントの状態及び格納ページ全体の状態を，RD エリアごとに解析します。

(1)　使用目的

表又はインデクスの乱れを検知できます。また，複数の RD エリアに分割格納された表又はインデクスは，RD エリアごとに均等に格納されているかなどを確認できます。

(2)　解析結果の情報

表又はインデクス単位の状態解析で表示される解析結果を次に示します。

(a)　表単位の状態解析

- **LOB 用 RD エリアを含まない場合**

```
pddbst
 pddbst VV-RR(Object Option) ** Table Analysis **            2008/12/25 15:54:13 [1]
 Table Name     : TBL02 [2]
 Auth Id        : user1 [3]
 Hash           :   [4]
  Total Segment :         15 [5] Max Sum Segment :         15 [7]
  Total Page    :         75 [6] Max Sum Page    :         75 [8]
  Total Row     :                    16 [9]
  RD Area Count :          1/         1 [10]
--------------------------------------------------------------------------------
 RD Area Name   : user_rdarea_1 [11]
  Server        : bes01 [12]
  Status        :   [38]
  Reference Pending Status : [41]
  Check     Pending Status : [42]
  Original RD Area Name : user_rdarea_1 [28]
  Generation Number :  0[29] Replica RD Area Count :  0[30]
  [31]     [32]              [33]              [34]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :

  Job Name      :   [25]  Line Count :   [26]         Index Method :   [27]
  Unit :     [43] Generation :    [44]
  Unused Segment:          9 [13]
  Segment Reuse :          - segments [36]
  Segment Reuse Option :          - [60]
  Row Count     :                   16 [24]
  <Base row segment>
  Search Mode : INS [35]
  Branch Row Search Mode : - [61]
  Reuse Search Failure :          0/         0 [37]
               Used(Full)       Used(      Full)        Sum
               [14]  [15]        [16]       [17]       [18]
  Segment [45] 100%(  0%)          1(         0)          1
               [19]  [20]        [21]       [22]       [23]
  Page    [46] 60%(  0%)           3(         0)          5
  Collect On Segment :          0 [59]
  Collect Prearranged Page :          0 [39]
  Collect Prearranged Full Page :          0[62]
  Used Page Ratio       Page(Ratio) [40]
              0% :          2(  40%)
           1- 10% :          1(  20%)
          11- 20% :          1(  20%)
          21- 30% :          0(   0%)
          31- 40% :          1(  20%)
          41- 50% :          0(   0%)
          51- 60% :          0(   0%)
          61- 70% :          0(   0%)
          71- 80% :          0(   0%)
          81- 90% :          0(   0%)
          91-100% :          0(   0%)
  Total                      5
  Del RowID   :                    0 [63]
  <BINARY segment>
```

```
Search Mode : INS
Branch Row Search Mode : -
Reuse Search Failure :          0/         0
             Used(Full)       Used(      Full)        Sum
             [47]  [48]        [49]       [50]       [51]
Segment [52] 100%( 93%)         14(        13)         14
             [53]  [54]        [55]       [56]       [57]
Page    [58]  96%( 95%)         67(        66)         70
Collect On Segment :          0
Collect Prearranged Page :          0
Collect Prearranged Full Page :          0
Used Page Ratio       Page(Ratio)
           0% :          3(   5%)
        1- 10% :          0(   0%)
       11- 20% :          0(   0%)
       21- 30% :          0(   0%)
       31- 40% :          0(   0%)
       41- 50% :          0(   0%)
       51- 60% :          0(   0%)
       61- 70% :          0(   0%)
       71- 80% :          0(   0%)
       81- 90% :          0(   0%)
       91-100% :         67(  96%)
Total                    70
```

[説明]

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月　DD：日　hh：時　mm：分　ss：秒
2. 解析対象の表の名称を表示します。
3. 解析対象の表の認可識別子を表示します。
4. フレキシブルハッシュ分割及び FIX ハッシュ分割の場合に，使用しているハッシュ関数の名称を表示します。フレキシブルハッシュ分割の場合は見出しが"Hash"，FIX ハッシュ分割の場合は見出しが"Fix Hash"となります。フレキシブルハッシュ分割及び FIX ハッシュ分割をしていない場合には，空白を表示します。
5. 表に割り当てられているセグメント数の合計（使用中セグメント数）を表示します。
6. 表に割り当てられているセグメント（使用中セグメント＋未使用セグメント）の中の，ページ数の合計を表示します。
7. 1RD エリア当たりの表に割り当てられている，セグメント数の合計の最大値を表示します。
8. 1RD エリア当たりの表に割り当てられている，セグメント中のページ数の合計の最大値を表示します。
9. 表の総格納行数（RD エリアごとの格納行数の合計）を表示します。この情報は，-s オプションを指定したときだけ表示します。
10. 正常に解析できた RD エリア数／表の格納ページを含む RD エリア数を表示します。
11. 解析対象の RD エリアの名称を表示します。
12. RD エリアを管理するサーバの名称を表示します。

13. RD エリア内の未使用セグメント数の合計を表示します。括弧内は該当する RD エリア内の，更新凍結状態でない HiRDB ファイルの未使用セグメント数の合計となります。
14. 使用中セグメント数の比率を表示します。

    ↑使用中セグメント数÷18 の値×100↑（%）
15. 満杯セグメント数の比率を表示します。

    ↑満杯セグメント数÷18 の値×100↑（%）
16. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
17. 満杯セグメント数を表示します。
18. RD エリア内の該当する表に割り当てられているセグメント数の合計（使用中セグメント数）を表示します。
19. 使用中ページ数の比率を表示します。

    ↑使用中ページ数÷23 の値×100↑（%）
20. 満杯ページ数の比率を表示します。

    ↑満杯ページ数÷23 の値×100↑（%）
21. 使用中ページ数を表示します。
22. 満杯ページ数を表示します。
23. RD エリアの該当する表に割り当てられている，セグメント中のページ数の合計（使用中ページ数＋未使用ページ数）を表示します。
24. 該当する RD エリアに格納されている行数を表示します。この情報は，-s オプションを指定したときだけ表示します。また，格納行数のカウントは NOWAIT 検索をするため，誤差が生じる場合があります。
25. 同期点指定のデータロード，再編成，及びリロードをしたときに設定されたジョブ名を表示します。括弧内はジョブ名を使用するユティリティ種別です。

    L：pdload

    R：pdrorg

    空白：設定されていません。
26. 同期点指定のデータロード，再編成，及びリロードをしたときに，データベースに格納された行数を表示します。
27. 同期点指定のデータロード，再編成，及びリロードをしたときのインデクス作成方法（-i オプションの指定値）を表示します。
28. 解析対象の RD エリアの名称を表示します。
29. 常に 0 を表示します。
30. 常に 0 を表示します。
31. 該当する RD エリアの閉塞履歴情報です。

    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）

History2：History1 の一つ前の閉塞情報

32. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

    CMD：HiRDB の障害検知によるコマンド閉塞

    FLT：障害閉塞

33. 該当する RD エリアの閉塞要因コードを表示します。閉塞要因コードについては，「LOB 用 RD エリア以外の場合」を参照してください。

34. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。

35. ページサーチモードを表示します。

    INS：新規ページ追加モード

    REU：空きページ再利用モード

36. CREATE TABLE 実行時，又は ALTER TABLE 実行時の SEGMENT REUSE で指定したセグメント数です。SEGMENT REUSE でセグメント数を指定していない場合は"0"を表示します。また，SEGMENT REUSE に NO を指定している場合，及び SEGMENT REUSE を省略している場合は"-"を表示します。

37. 新規ページ追加モードから空きページ再利用モードに切り替えたときに，再利用できる空き領域がなく，新規ページ追加モードに戻した回数（n/m）を表示します。

    n：空回り回数

    m：新規ページ追加モードから空きページ再利用モードに切り替えた回数

    n 及び m は，次の契機でリセットされ，0 が設定されます。

    - RD エリアのクローズ時
    - HiRDB の再開始時
    - PURGE TABLE 文実行時

38. データ未完状態であるか，又は再編成実行中であるかを表示します。次のどちらでもない場合は，空白を表示します。

    DATA_UNFINISH：

    データ未完状態です。表の再編成を実行し，途中でエラーなどによってリロード処理が完了していない状態のことをいいます。データ未完状態を解除するためには，表の再編成を再実行するか，又は表へのリロードを実行する必要があります。

    ON_RORG：

    再編成実行中です。

39. pdreclaim で解放できるページ数（使用率 0%の使用中空きページの数）を表示します。

40. 使用中ページの比率別ページ数を 10%ごとに表示します。括弧内の値は，すべての使用中ページに対する比率を表します。比率は小数点以下を切り上げて表示するため，比率を合計すると 100 を超えることがあります。注意事項を次に示します。

    - 満杯ページであっても，使用率が 91−100%であるとは限りません。
    - 行長が短くなる更新をして空いた領域は，空き領域として扱いません。

- （100−PCTFREE で指定した空き領域比率）よりも，使用中ページ比率が小さいページが多い場合，格納効率が低下している可能性があります。この場合，pdrorg の実行を検討してください。ただし，分岐行（256 バイト以上の可変長文字列データ，繰返し列のデータ，及び抽象データ型列のデータ）を格納している領域は，pdrorg を実行しても格納効率は変わりません。

41. RD エリア中の表情報の参照制約の検査保留状態を表示します。表に参照制約の定義がない場合は空白を表示します。

    P：検査保留状態

    空白：非検査保留状態

42. RD エリア中の表情報の検査制約の検査保留状態を表示します。表に検査制約の定義がない場合は空白を表示します。

    P：検査保留状態

    空白：非検査保留状態

43. 監査証跡表へのデータロード中に pdload が異常終了したため，監査証跡ファイルのデータロードが完了しているのに，監査証跡ファイルの状態をデータロード済み状態に変更できなかったおそれがあるユニットの識別子を表示します。異常終了していない場合や，異常終了したタイミングが監査証跡ファイルの状態変更と無関係な場合は空白を表示します。

44. 43 と同様に，監査証跡ファイルの状態を変更できなかったおそれがある監査証跡ファイルの世代番号（監査証跡ファイル名の後ろから 3 文字）を表示します。異常終了していない場合や，異常終了したタイミングが監査証跡ファイルの状態変更と無関係な場合は空白を表示します。

45. セグメント単位のサマリ情報です。バイナリ専用セグメントの情報は含みません。

46. ページ単位のサマリ情報です。

47. 使用中バイナリ専用セグメント数の比率を表示します。

    ↑使用中バイナリ専用セグメント数÷表又はインデクスに割り当てられているバイナリ専用セグメント数の合計×100↑（%）

48. 満杯バイナリ専用セグメント数の比率を表示します。

    ↑満杯バイナリ専用セグメント数÷表又はインデクスに割り当てられているバイナリ専用セグメント数の合計×100↑（%）

49. 使用中バイナリ専用セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中バイナリ専用セグメントとして表示します。

50. 満杯バイナリ専用セグメント数を表示します。

51. 表又はインデクスに割り当てられているバイナリ専用セグメント数の合計を表示します。

52. バイナリ専用セグメント単位のサマリ情報です。

53. バイナリ専用セグメントの使用中ページ数の比率を表示します。

    ↑バイナリ専用セグメントの使用中ページ数÷表又はインデクスに割り当てられているバイナリ専用セグメントのページ数の合計×100↑（%）

54. バイナリ専用セグメントの満杯ページ数の比率を表示します。

↑使用中バイナリ専用セグメントの満杯ページ数÷表又はインデクスに割り当てられているバイナリ専用セグメントのページ数の合計×100↑（%）

55. バイナリ専用セグメントの使用中ページ数を表示します。

56. バイナリ専用セグメントの満杯ページ数を表示します。

57. 表又はインデクスに割り当てられているバイナリ専用セグメントのページ数の合計を表示します。

58. バイナリ専用セグメントのページ単位のサマリ情報です。

59. -a オプションを指定した pdreclaim が同時実行中，又はエラー終了した場合に，解放途中のセグメント数を表示します。-a オプションを指定した pdreclaim を実行していない場合は 0 を表示します。

60. 表定義の SEGMENT REUSE で指定した再利用オプション値を表示します。

61. 分岐行に対する空き領域サーチモードを表示します。

    INS：新規ページ追加モード

    REU：空きページ再利用モード

    SEGMENT REUSE の再利用オプション値で「分岐行が多発する表の格納効率向上」を指定していない場合は，"-"が表示されます。

62. pdreclaim 又は pdrorg で解放できる満杯ページ数を表示します。

63. 表の削除状態の，管理行数の合計を表示します。

注 1

39 及び 40 は，-d オプションを指定した場合に表示される情報です。

注 2

43 及び 44 は，監査証跡表の場合に表示される情報です。

注 3

45～58 は，表に，定義長 256 バイト以上の BINARY 型の列，又は抽象データ型の列が定義されている場合に出力されます。

- **LOB 用 RD エリアを含む場合**

```
 pddbst VV-RR(Object Option) ** Table Analysis **           2008/12/25 15:54:14 [1]
 Table Name     : TBL03 [2]
 Auth Id        : user1 [3]
 Hash           :    [4]
  Total Segment :       1247 [5] Max Sum Segment :       1234 [7]
  Total Page    :       1299 [6] Max Sum Page    :       1234 [8]
  Total Row     :                  1,234 [9]
  RD Area Count :          2/         2 [10]
-------------------------------------------------------------------------------
 RD Area Name   : user_rdarea_3 [11]
  Server        : bes01 [12]
  Status        :   [42]
  Reference Pending Status :
  Check     Pending Status :
  Original RD Area Name : user_rdarea_3 [32]
  Generation Number :  0 [33]  Replica RD Area Count :  0[34]
```

```
  [35]    [36]              [37]              [38]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
  Job Name      :   [29]  Line Count :   [30]         Index Method :   [31]
  Unused Segment:        137 [13]
  Segment Reuse :          - segments [40]
  Segment Reuse Option :          - [46]
  Row Count     :                 1,234 [24]
  <Base row segment>
  Search Mode : INS [39]
  Branch Row Search Mode : - [47]
  Reuse Search Failure :          0/         0 [41]
              Used(Full)       Used(      Full)        Sum
              [14]  [15]        [16]       [17]        [18]
  Segment [43]100%( 85%)         13(        11)         13
              [19]  [20]        [21]       [22]        [23]
  Page    [44]100%( 97%)         65(        63)         65
  Collect On Segment :          0 [45]
  Collect Prearranged Page :          0
  Collect Prearranged Full Page :          0 [48]
  Used Page Ratio       Page(Ratio)
             0% :          0(   0%)
          1- 10% :          0(   0%)
         11- 20% :          0(   0%)
         21- 30% :          1(   2%)
         31- 40% :          0(   0%)
         41- 50% :          0(   0%)
         51- 60% :          0(   0%)
         61- 70% :          0(   0%)
         71- 80% :          0(   0%)
         81- 90% :          0(   0%)
        91-100% :         64(  99%)
  Total                    65
  Del RowID   :                    0 [49]
 RD Area Name   : user_rdlob_2
  Server        : bes01
  Status        :
  Column Name   : C4 [25]
  Original RD Area Name : user_rdlob_2
  Generation Number :  0   Replica RD Area Count :  0
  History1 Hold Status :      Hold Code :     0  Hold Time :

  History2 Hold Status :      Hold Code :     0  Hold Time :
  Unused Segment:        266(       266)
          Used       Used        Sum
  Segment 100%       1234       1234
  Segment Over  : N  [26] Last Segment :       1234/      1500 [27]
  Lobmap  Over  : N [28]
```

[説明]

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒
2. 解析対象の表の名称を表示します。
3. 解析対象の表の認可識別子を表示します。

4. フレキシブルハッシュ分割及び FIX ハッシュ分割の場合に，使用しているハッシュ関数の名称を表示します。フレキシブルハッシュ分割の場合は見出しが"Hash"，FIX ハッシュ分割の場合は見出しが"Fix Hash"となります。フレキシブルハッシュ分割及び FIX ハッシュ分割をしていない場合には，空白を表示します。
5. 表に割り当てられているセグメント数の合計（使用中セグメント数＋未使用セグメント数）を表示します。
6. 表に割り当てられているセグメント（使用中セグメント＋未使用セグメント）の中の，ページ数の合計を表示します。
7. 1RD エリア当たりの表に割り当てられている，セグメント数の合計の最大値を表示します。
8. 1RD エリア当たりの表に割り当てられている，セグメント中のページ数の合計の最大値を表示します。
9. 表の総格納行数（RD エリアごとの格納行数の合計）を表示します。この情報は，-s オプションを指定したときだけ表示します。
10. 正常に解析できた RD エリア数／表の格納ページを含む RD エリア数を表示します。
11. 解析対象の RD エリアの名称を表示します。ユーザ LOB 用 RD エリアがある場合，先に LOB 列構成基表のユーザ用 RD エリア名を表示し，次に LOB データのユーザ LOB 用 RD エリア名を表示します。
12. RD エリアを管理するサーバの名称を表示します。
13. RD エリア内の未使用セグメント数の合計を表示します。括弧内は該当する RD エリア内の，更新凍結状態でない HiRDB ファイルの未使用セグメント数の合計となります。
14. 使用中セグメント数の比率を表示します。
    ↑使用中セグメント数÷18 の値×100↑（%）
15. 満杯セグメント数の比率を表示します。
    ↑満杯セグメント数÷18 の値×100↑（%）
16. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
17. 満杯セグメント数を表示します。
18. RD エリア内の該当する表に割り当てられているセグメント数の合計（使用中セグメント数）を表示します。
19. 使用中ページ数の比率を表示します。
    ↑使用中ページ数÷23 の値×100↑（%）
20. 満杯ページ数の比率を表示します。
    ↑満杯ページ数÷23 の値×100↑（%）
21. 使用中ページ数を表示します。
22. 満杯ページ数を表示します。
23. RD エリアの該当する表に割り当てられている，セグメント中のページ数の合計（使用中ページ数＋未使用ページ数）を表示します。

24. 該当する RD エリアに格納されている行数を表示します。この情報は，-s オプションを指定したときだけ表示します。また，格納行数のカウントは NOWAIT 検索をするため，誤差が生じる場合があります。
25. LOB 列の名称を表示します。
26. LOB 用 RD エリアが乱れているかどうかを表示します。

    Y：乱れています。

    N：乱れていません。
27. 使用されているセグメントの最後を示す位置情報を表示します。最終位置/総セグメント数の形式で表示します。26 が Y のときは，常に最後のセグメントを示します。
28. LOB 管理エントリがすべて使用されているかどうかを表示します。

    Y：すべて使用されています。

    N：未使用のエントリが残っています。
29. 同期点指定のデータロード，再編成，及びリロードをしたときに設定されたジョブ名を表示します。括弧内はジョブ名を使用するユティリティ種別です。

    L：pdload

    R：pdrorg

    空白：設定されていません。
30. 同期点指定のデータロード，再編成，及びリロードをしたときに，データベースに格納された行数を表示します。
31. 同期点指定のデータロード，再編成，及びリロードをしたときのインデクス作成方法（-i オプションの指定値）を表示します。
32. 解析対象の RD エリアの名称を表示します。
33. 常に 0 を表示します。
34. 常に 0 を表示します。
35. 該当する RD エリアの閉塞履歴情報です。

    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）

    History2：History1 の一つ前の閉塞情報
36. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

    CMD：HiRDB の障害検知によるコマンド閉塞

    FLT：障害閉塞
37. 該当する RD エリアの閉塞要因コードを表示します。閉塞要因コードについては，「LOB 用 RD エリア以外の場合」を参照してください。
38. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。
39. ページサーチモードを表示します。

    INS：新規ページ追加モード

    REU：空きページ再利用モード

40. CREATE TABLE 実行時，又は ALTER TABLE 実行時の SEGMENT REUSE で指定したセグメント数です。SEGMENT REUSE でセグメント数を指定していない場合は"0"を表示します。また，SEGMENT REUSE に NO を指定している場合，及び SEGMENT REUSE を省略している場合は"-"を表示します。
41. 新規ページ追加モードから空きページ再利用モードに切り替えたときに，再利用できる空き領域がなく，新規ページ追加モードに戻した回数（n/m）を表示します。

    n：空回り回数

    m：新規ページ追加モードから空きページ再利用モードに切り替えた回数

    n 及び m は，次の契機でリセットされ，0 が設定されます。

    - RD エリアのクローズ時
    - HiRDB の再開始時
    - PURGE TABLE 文実行時
42. データ未完状態であるか，又は再編成実行中であるかを表示します。次のどちらでもない場合は，空白を表示します。

    DATA_UNFINISH：

    データ未完状態です。表の再編成を実行し，途中でエラーなどによってリロード処理が完了していない状態のことをいいます。データ未完状態を解除するためには，表の再編成を再実行するか，又は表へのリロードを実行する必要があります。

    ON_RORG：

    再編成実行中です。
43. セグメント単位のサマリ情報です。バイナリ専用セグメントの情報は含みません。
44. ページ単位のサマリ情報です。
45. -a オプションを指定した pdreclaim が同時実行中，又はエラー終了した場合に，解放途中のセグメント数を表示します。-a オプションを指定した pdreclaim を実行していない場合は 0 を表示します。
46. 表定義の SEGMENT REUSE で指定した再利用オプション値を表示します。
47. 分岐行に対する空き領域サーチモードを表示します。

    INS：新規ページ追加モード

    REU：空きページ再利用モード

    SEGMENT REUSE の再利用オプション値で「分岐行が多発する表の格納効率向上」を指定していない場合は，"-"が表示されます。
48. pdreclaim 又は pdrorg で解放できる満杯ページ数を表示します。
49. 表の削除状態の，管理行数の合計を表示します。

(b) インデクス単位の状態解析

- **B-tree インデクスの場合**

```
pddbst VV-RR(Object Option) ** Index Analysis **           2008/12/25 18:56:00 [1]
Index Name     : IDX_TBL01_C1 [2]
```

```
 Auth Id        : user1 [3]
  Total Segment :         70 [4] Max Sum Segment :         70 [6]
  Total Page    :        700 [5] Max Sum Page    :        700 [7]
  RD Area Count :          1/         1 [8]
--------------------------------------------------------------------------------
 RD Area Name   : user_rdarea_1 [9]
  Server        : bes1 [10]
  Status        :   [22]
  Original RD Area Name : user_rdarea_1 [23]
  Generation Number :  0[24] Replica RD Area Count :  0[25]
  [26]     [27]               [28]               [29]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
  Unused Segment:        815 [11]
          Used(Full)       Used(      Full)        Sum
          [12]  [13]        [14]       [15]        [16]
  Segment 100%(  0%)         70(         0)         70
          [17]  [18]        [19]       [20]        [21]
  Page     99%(  0%)        693(         0)        700
  Collect On Segment :           0 [37]
  Collect Prearranged Page :          26 [30]
  Collect On Page :          0 [31]
  Used Page Ratio       Page(Ratio) [32] Del RowID [33]
             0% :         33(   5%)              85
           1- 10% :          1(   1%)              17
          11- 20% :          2(   1%)              52
          21- 30% :          1(   1%)              45
          31- 40% :          0(   0%)              0
          41- 50% :          0(   0%)              0
          51- 60% :          0(   0%)              0
          61- 70% :        663(  95%)              633
          71- 80% :          0(   0%)              0
          81- 90% :          0(   0%)              0
          91-100% :          0(   0%)              0
  Total                    700                     832 [34]
  Del RowID   :                  832          (  1%) [36]
  RowID Count :                95,998 [35]
```

[説明]

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月　DD：日　hh：時　mm：分　ss：秒
2. 解析対象のインデクスの名称を表示します。
3. 解析対象のインデクスの認可識別子を表示します。
4. インデクスに割り当てられているセグメント数の合計（使用中セグメント数＋未使用セグメント数）を表示します。
5. インデクスに割り当てられているセグメント（使用中セグメント＋未使用セグメント）の中の，ページ数の合計を表示します。
6. 1RD エリア当たりのインデクスに割り当てられている，セグメント数の合計の最大値を表示します。

7. 1RD エリア当たりのインデクスに割り当てられている，セグメント中のページ数の合計の最大値を表示します。
8. 正常に解析できた RD エリア数／インデクスの格納ページを含む RD エリア数を表示します。
9. 解析対象の RD エリアの名称を表示します。
10. RD エリアを管理するサーバの名称を表示します。
11. RD エリア中の未使用セグメント数の合計を表示します。
12. 使用中セグメント数の比率を表示します。
    ↑使用中セグメント数÷16 の値×100↑（%）
13. 満杯セグメント数の比率を表示します。
    ↑満杯セグメント数÷16 の値×100↑（%）
14. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
15. 満杯セグメント数を表示します。
16. RD エリア内の該当するインデクスに割り当てられているセグメント数の合計（使用中セグメント数）を表示します。
17. 使用中ページ数の比率を表示します。
    ↑使用中ページ数÷21 の値×100↑（%）
18. 満杯ページ数の比率を表示します。
    ↑満杯ページ数÷21 の値×100↑（%）
19. 使用中ページ数を表示します。
20. 満杯ページ数を表示します。
21. RD エリアの該当するインデクスに割り当てられている，セグメント中のページ数の合計（使用中ページ数＋未使用ページ数）を表示します。
22. インデクス作成時の，インデクスの未完状態を表示します。未完状態でない場合は空白を表示します。
    UNFINISH_0：図「表単位のインデクス作成未完状態の種類（1/2）」，図「表単位のインデクス作成未完状態の種類（2/2）」を参照してください。
    UNFINISH_1：図「RD エリア単位のインデクス作成未完状態の種類（1/2）」，図「RD エリア単位のインデクス作成未完状態の種類（2/2）」を参照してください。
23. 解析対象の RD エリアの名称を表示します。
24. 常に 0 を表示します。
25. 常に 0 を表示します。
26. 該当する RD エリアの閉塞履歴情報です。
    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）
    History2：History1 の一つ前の閉塞情報

27. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

    CMD：HiRDB の障害検知によるコマンド閉塞

    FLT：障害閉塞

28. 該当する RD エリアの閉塞要因コードを表示します。閉塞要因コードについては，「LOB 用 RD エリア以外の場合」を参照してください。

29. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。

30. pdreclaim で解放できるページ数（使用率 0%の使用中空きページの数）を表示します。

31. pdreclaim の実行途中，又は pdreclaim がエラーで終了しているために解放処理が途中になっている状態のページ数を表示します。エラーで終了している場合は，pdreclaim を再実行してください（このページ数は 0 になります）。なお，表示するページ数は，使用中ページの比率別（0%）のページ数にカウントされます。

32. 使用中ページの比率別ページ数を 10%ごとに表示します。括弧内の値は，すべての使用中ページに対する比率を表します。比率は小数点以下を切り上げて表示するため，比率を合計すると 100 を超えることがあります。注意事項を次に示します。

    - 201 件以上重複したキー値を格納するページは，表定義時の PCTFREE の比率が無効となります。そのため，再編成する場合でも PCTFREE の比率どおりに再配置されません。
    - インデクスキー値無排他を適用したインデクスの場合，削除キー値は空き領域として扱い，使用中ページのすべてが削除キー値のときは，使用率 0%となります。
    - 満杯ページであっても，使用率が 91−100%であるとは限りません。
    - 行長が短くなる更新をして空いた領域は，空き領域として扱いません。
    - （100−PCTFREE で指定した空き領域比率）よりも，使用中ページ比率が小さいページが多い場合，格納効率が低下している可能性があります。この場合，pdrorg の実行を検討してください。ただし，分岐行（256 バイト以上の可変長文字列データ，繰返し列のデータ，及び抽象データ型列のデータ）を格納している領域は，pdrorg を実行しても格納効率は変わりません。

33. インデクスキー値無排他を使用している場合又はコミットしていない削除データに排他を掛ける場合，次の数値を表示します。

    - 使用中インデクスページの比率別残存エントリが管理する行数

      （残存エントリとは，インデクスキー値の更新や削除によって，キーの削除後も排他制御のためにインデクス上に残っているインデクスエントリ）

    表示けた数は 3 けたで，小数点と単位を含めた最大表示けた数は 5 けたです。表示する行数が 3 けたを超える場合に小数点と単位を表示します。その場合，先頭から 4 けた目を四捨五入します。残存エントリが管理する行数の表示形式については，「LOB 用 RD エリア以外の場合」を参照してください。

34. 使用中ページの比率別ページ数の合計と，残存エントリが管理する行数の合計を表示します。

35. インデクスが管理する総行数を表示します。15 けたまでは整数，16 けた以上になると指数で表示します。例えば，1000000000000000 は 1E+15 と表示します。表示形式については，「LOB 用 RD エリア以外の場合」を参照してください。

36. インデクスの，削除状態である管理行数の合計を表示します。又は，インデクスの削除状態である管理行数の，すべての管理行数に対する割合を表示します。

37. -a オプションを指定した pdreclaim が同時実行中，又はエラー終了した場合に，解放途中のセグメント数を表示します。-a オプションを指定した pdreclaim を実行していない場合は 0 を表示します。

注 1

30〜32 は，-d オプションを指定した場合に表示される情報です。

また，33，34，及び 35 は-d オプションと-z オプションを同時に指定した場合に表示される情報です。

- **プラグインインデクスの場合**

```
 pddbst VV-RR(Object Option) ** Index Analysis **           2002/03/29 18:05:06 [1]
 Index Name     : IDX05 [2]
 Auth Id        : MANUAL [3]
  Total Segment :       9501[4] Max Sum Segment :       9501 [6]
  Total Page    :       9501[5] Max Sum Page    :       9501 [7]
  RD Area Count :          1/         1 [8]
-------------------------------------------------------------------------------
 RD Area Name   : USER_RDLOB_10 [9]
  Server        : BES02 [10]
  Status        :     [15]
  Original RD Area Name : USER_RDLOB_10 [19]
  Generation Number :  0[20] Replica RD Area Count :  0 [21]
    [22]                [23]                [24]           [25]
  History1 Hold Status :      Hold Code :     0  Hold Time :
  History2 Hold Status :      Hold Code :     0  Hold Time :
  Unused Segment:        499(       499) [11]
          [12]       [13]       [14]
          Used       Used        Sum
  Segment 100%       9501       9501
                 [16]                        [17]
  Segment Over  : N  Last Segment :       9501/     10000
                 [18]
  Lobmap  Over  : N
```

[説明]

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。

   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒

2. 解析対象のインデクスの名称を表示します。

3. 解析対象のインデクスの認可識別子を表示します。

4. インデクスに割り当てられているセグメント数（使用中セグメント数＋未使用セグメント数）の合計を表示します。

5. インデクスに割り当てられているセグメント（使用中セグメント＋未使用セグメント）の中の，ページ数の合計を表示します。

6. 1RD エリア当たりのインデクスに割り当てられている，セグメント数の合計の最大値を表示します。

7. 1RD エリア当たりのインデクスに割り当てられている，セグメント中のページ数の合計の最大値を表示します。
8. 正常に解析できた RD エリア数／インデクスの格納ページを含む RD エリア数を表示します。
9. 解析対象の RD エリアの名称を表示します。
10. RD エリアを管理するサーバの名称を表示します。
11. RD エリア内の未使用セグメント数の合計を表示します。括弧内は該当する RD エリア内の，更新凍結状態でない HiRDB ファイルの未使用セグメント数の合計となります。
12. 使用中セグメント数の比率を表示します。
    ↑使用中セグメント数÷16 の値×100↑（%）
13. 使用中セグメント数を表示します。pdreclaim での解放途中セグメントは，使用中セグメントとして表示します。
14. RD エリア内の該当するインデクスに割り当てられているセグメント数の合計（使用中セグメント数）を表示します。
15. インデクス作成時の，インデクスの未完状態を表示します。未完状態でない場合は空白を表示します。

    UNFINISH_0：

    図「表単位のインデクス作成未完状態の種類（1/2）」，図「表単位のインデクス作成未完状態の種類（2/2）」を参照してください。

    UNFINISH_1：

    図「RD エリア単位のインデクス作成未完状態の種類（1/2）」，図「RD エリア単位のインデクス作成未完状態の種類（2/2）」を参照してください。

    UNFINISH_2：

    プラグインインデクス遅延更新中。プラグインインデクス遅延一括作成機能を使用した UAP を実行中，又は UAP 終了後に pdrorg のインデクス一括作成を実行していない状態です。プラグインインデクスの遅延一括作成については，マニュアル「HiRDB システム運用ガイド」を参照してください。
16. LOB 用 RD エリアが乱れているかどうかを表示します。

    Y：乱れています。

    N：乱れていません。
17. 使用されているセグメントの最後を示す位置情報を表示します。最終位置/総セグメント数の形式で表示します。Segment Over の表示が Y の場合，常に最後のセグメントを示します。
18. LOB 管理エントリがすべて使用されているかどうかを表示します。

    Y ：すべて使用されています。

    N ：未使用のエントリが残っています。
19. 解析対象の RD エリアの名称を表示します。
20. 常に 0 を表示します。
21. 常に 0 を表示します。

22. 該当する RD エリアの閉塞履歴情報です。

    History1：閉塞解除する一つ前の閉塞情報（現時点で閉塞中の場合は現在の閉塞情報）

    History2：History1 の一つ前の閉塞情報

23. 該当する RD エリアの閉塞種別を表示します。履歴がない場合は空白を表示します。

    CMD：HiRDB の障害検知によるコマンド閉塞

    FLT：障害閉塞

24. 該当する RD エリアの閉塞要因コードを表示します。閉塞要因コードについては，「LOB 用 RD エリア以外の場合」を参照してください。

25. 該当する RD エリアの閉塞時刻を表示します。履歴がない場合は空白を表示します。

(3)　解析結果の分析

表又はインデクス単位の状態解析の解析結果は，RD エリア単位の状態解析（論理的解析）と同じように分析できます。

格納行数が RD エリアごとに均等でない場合，HiRDB/パラレルサーバの性能を最大限に引き出すことができません。この場合，キーレンジ分割の見直しや，フレキシブルハッシュ分割又は FIX ハッシュ分割を適用するなど（又はハッシュ関数を変更するなど）してください。これによって，格納行数が RD エリアごとに均等分割されるようになります。

なお，プラグインが提供する抽象データ型の LOB 属性を格納する RD エリア，又はプラグインインデクスを格納する RD エリアの状態を解析した場合，RD エリア内の使用中セグメントの割合は，プラグインのセグメントの割り当て方に依存します。

15.3.4　クラスタキー及びクラスタリングデータページの格納状態解析

(1)　使用目的

クラスタキー及びクラスタリングデータページの格納状態の乱れ具合が分かるため，再編成をするかどうかの判断ができます。

(2)　解析結果の情報

クラスタキー及びクラスタリングデータページの格納状態解析で表示される解析結果を次に示します。

```
pddbst VV-RR(Object Option) ** Index Status Analysis **     2001/08/22 16:44:01 [1]
Index Name : (CLUSTER0000131198) [2]
Auth Id    : MANUAL [3]
Table Name : TBL06 [4]


***** Cluster Key Analysis *****
 RD Area Count :          1/         1 [5]
```

```
--------------------------------------------------------------------------------
 RD Area Name : USER_RDAREA_4                      Server : BES01 [6]
  Original RD Area Name : USER_RDAREA_4 [24]
  Generation Number :  0[25] Replica RD Area Count :  0[26]
  Segment Size:          5 Pages   Page Size :       4096 [7]
  Index Level :          1 [8]
                      Total            Repetitional
  Key Count             100 [9]                   0 [10]
  Row Count             100 [11]                  0 [12]
               Total   Disordered
  Segment          0 [13]       0(  0.0%) [14]
  Page             0 [15]       0(  0.0%) [16]

 ***** Clustering Table Analysis *****
  RD Area Count :          1/         1 [17]
--------------------------------------------------------------------------------
 RD Area Name : USER_RDAREA_4                      Server : BES01 [18]
  Original RD Area Name : USER_RDAREA_4
  Generation Number :  0   Replica RD Area Count :  0
  Segment Size:          5 Pages   Page Size :       4096 [19]
               Total   Disordered
  Segment          0 [20]       0(  0.0%) [21]
  Page             0 [22]       0(  0.0%) [23]
```

[説明]

1. 状態解析の基になる情報の取得完了年月日時刻を，YYYY/MM/DD hh:mm:ss の形式で表示します。
   YYYY：年（西暦） MM：月　DD：日　hh：時　mm：分　ss：秒
2. 解析対象となるクラスタキーのインデクス名称を表示します。
3. 解析対象となるクラスタキーのインデクス所有者の認可識別子を表示します。
4. 解析対象となるクラスタキーを定義してある表の名称を表示します。
5. 正常に解析できた RD エリアの数／インデクスを格納している RD エリアの数を表示します。
6. 解析したインデクスを格納している RD エリアの名称と，その RD エリアを管理するサーバ名を表示します。
7. 6 の RD エリアのセグメントサイズ（1セグメント内のページ数）とページ長（バイト）を表示します。
8. 6 の RD エリアのインデクスキーのインデクスの段数を表示します。
9. 6 の RD エリアのインデクス中の格納キー数を表示します。値の表示については，15 けたまでは整数で表示し，15 けたを超えると指数で表示します。
10. 6 の RD エリアのインデクス中に重複キー構造※で格納されているキー数（値の表示については 9 と同様）を表示します。
    注※　キー値の重複数が 201 以上になったとき，インデクスページ中の該当するキー管理情報を複数ページに格納する場合の構造を意味します。
11. 10 の RD エリアのインデクス中の格納行数（値の表示については 9 と同様）を表示します。
12. 6 に対応する行数（値の表示については 9 と同様）を表示します。

13. 6 の RD エリアのインデクス中セグメントに対して，格納位置が変更した回数を表示します。
14. 13 のうち，格納順序が不正（逆方向）になった回数を表示します。括弧内は，セグメントの格納順序不正率（(14 の値÷13 の値）×100%）を表します。
15. 6 の RD エリアのインデクス中ページに対して，格納位置が変更した回数を表示します。
16. 15 のうち，格納順序が不正（逆方向）になった回数を表示します。括弧内は，ページの格納順序不正率（(16 の値÷15 の値）×100%）を表します。
17. 正常に解析できた RD エリアの数／表を格納している RD エリアの数を表示します。
18. 解析した表を格納している RD エリアの名称と，その RD エリアを管理するサーバ名を表示します。
19. 18 の RD エリアのセグメントサイズ（1セグメント内のページ数）とページ長（バイト）を表示します。
20. 18 の RD エリアのインデクス中セグメントに対して，格納位置が変更した回数を表示します。
21. 20 のうち，格納順序が不正（逆方向）になった回数を表示します。括弧内は，セグメントの格納順序不正率（(21 の値÷20 の値）×100%）を表します。
22. 18 の RD エリアの表格納ページに対して，格納位置が変更した回数を表示します。
23. 22 のうち，格納順序が不正（逆方向）になった回数を表示します。括弧内は，ページの格納順序不正率（(23 の値÷22 の値）×100%）を表します。
24. 解析対象の RD エリアの名称を表示します。
25. 常に 0 を表示します。
26. 常に 0 を表示します。

注 1

6〜17 は，クラスタキーが複数の RD エリアに分割格納されている場合，分割数分の RD エリアの情報が表示されます。表示順序は RD エリア名の昇順に出力されます。

注 2

18〜23 は，該当する表が複数の RD エリアに分割格納されている場合，分割数分の RD エリアの情報が表示されます。表示順序は RD エリア名の昇順に出力されます。

(3) 解析結果の分析

インデクスキーの状態解析及びクラスタリング状態解析の結果と対策を，次の表に示します。

なお，行の削除はクラスタリングデータ状態解析結果の不正率に反映されないので，表に対する操作が削除中心の場合は，次に示す処理を実行して，再編成するかどうか判断する必要があります。

1. 表の状態解析による使用中空きページの検知
2. 格納行数と使用中ページ数による空きエリアの算出

2 の場合の不当な空きエリア長を次に示します。

（表の現在の使用中ページ数※1−表の格納ページ数※2）×ページサイズ
（単位：バイト）

注※1

表単位の状態解析の結果から算出してください。

注※2

マニュアル「HiRDB システム導入・設計ガイド」を参照して，算出してください。

表 15–7　インデクスキーの状態解析及びクラスタリング状態解析の結果と対策

対象	解析結果	意味・対策
クラスタキー	インデクス中の格納キー数に比べて行数が多いです。	キーの重複率が高くなっているので，インデクスを使用した検索性能に影響があります。 インデクス定義の列構成を見直してください。
	重複キー構造※1で格納されているキーがあります。	重複度の高いキーがある（又は過去にあった）ので，インデクスを使用した検索性能に影響があります。 インデクス定義の列構成を見直してください（過去にあった場合は，pdrorg でインデクスを再編成すると，重複キー構造はなくなります）。※2
	格納順序不正率が高いです。	キーの格納順序が乱れているため，検索性能に影響があります。 pdrorg でインデクスを再編成してください。なお，データページの格納順序も乱れている場合は，表を再編成するとインデクスも再編成されます。
クラスタリングデータページ	格納順序不正率が高いです。	データページの格納順序が乱れているため，クラスタキー順アクセス性能に影響があります。 pdrorg で表を再編成してください。
	格納位置変更回数＞表単位の状態解析結果の使用セグメント数，又は格納位置変更回数＞ページ数−1 となっています。	

注※1

キー値の重複数が 201 以上になったとき，インデクスページ中の該当するキーの管理情報を，複数ページに格納する場合の構造をいいます。一度重複キー構造になると，重複数が 201 未満になっても情報管理ページは削除されません。

注※2

インデクス構成列中のデータの重複度の高い列を除いて，クラスタ付きの表を再定義してください。

15.4　再編成時期予測機能の実行結果

15.4.1　実行結果の出力形式

再編成時期予測機能の実行結果は，-e オプション及び-m オプションの指定値によって変わります。オプションと出力情報の関係を次の表に示します。

表 15–8　オプションと出力情報の関係（再編成時期予測機能）

-e オプション	-m オプション	DB メンテナンス予定日の情報	メンテナンス方法の情報	各解析項目別情報
1（予測レベル 1）	あり	○	○	–
	なし	○	–	–
2（予測レベル 2）	×	○	○	○

（凡例）

○：該当する情報を出力します。

–：該当する情報は出力されません。

×：指定できません。

(1)　DB メンテナンス予定日の情報

監視期間内にセグメント使用率が基準値を超える RD エリアを，DB メンテナンス予定日が早い順に出力します。

```
 pddbst vv-rr       ***** Rdarea resource forecast *****     yyyy/mm/dd hh:mm:ss
No         Date        RdareaName                      ResKind
---------- ----------- ------------------------------- ----------
         1 2005/05/31  "RDUSER01"                      Segment
         2 2005/06/01  "RDUSER12"                      Segment
         3 2005/06/03  "rd0002"                        Segment
         4 2005/06/04  "lobrdarea701"                  Segment
         5 2005/06/11  "RDUSER02"                      Segment
---------- ----------- ------------------------------- ----------
```

注

HiRDB/パラレルサーバの場合，DB メンテナンス予定日は，ディクショナリサーバの日時を基に予測しています。

[説明]

No：

解析結果番号です（最大 10 けた）。

Date：

セグメント使用率が基準値を超えると予測した日（DB メンテナンス予定日）です（YYYY/MM/DD の形式）。

RdareaName：

メンテナンスが必要と予測した RD エリアの名称です（最大 32 文字）。

ResKind：

基準値を超えると予測した項目です（最大 10 文字）。

Segment：セグメント使用率

(2) メンテナンス方法の情報

監視期間内にセグメント使用率が基準値を超える RD エリアを，DB メンテナンス予定日が早い順に出力します。このとき，メンテナンス方法も出力します。また，予測レベル 2 の場合は，解析項目別情報でメンテナンス要と判断された表，インデクスのうち，RD エリアのメンテナンスに影響のあるものも出力します。

```
 pddbst vv-rr       ***** Maintenance Information *****       yyyy/mm/dd hh:mm:ss
No          : 1
Rdarea Name : "RDUSER01"
Method      :
Segment     : 42560
--------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                      Date
------------- ------------ ---- ----------------------------------------- ----------
      28089 *       9363    T   "k1234567"."table01"                      2005/05/31
          0        12342 *  T   "k1234567"."table02"                      2005/05/31
        851         7235 *  I   "k1234567"."index01"                      2005/05/31
          3          -10    T   "k1234567"."table10"                      2005/05/31
================================================================================
No          : 2
Rdarea Name : "RDUSER12"
Method      :
Segment     : 32985
--------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                      Date
------------- ------------ ---- ----------------------------------------- ----------
       3628        23749 *  T   "k1234567"."table01"                      2005/06/01
       1319         5937 *  T   "k1234567"."table03"                      2005/06/01
       1649 *       3628    I   "k1234567"."index01"                      2005/06/01
       2638 *        989    T   "k1234567"."table11"                      2005/06/01
================================================================================
No          : 3
Rdarea Name : "rd0002"
Method      : Expand
Segment     : 1523
--------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                      Date
------------- ------------ ---- ----------------------------------------- ----------
        228         1035    T   "k1234567"."table01"                      2005/06/03
          0          121    T   "k1234567"."table03"                      2005/06/03
         30           60    I   "k1234567"."index07"                      2005/06/03
================================================================================
```

```
No          : 4
Rdarea Name : "lobrdarea701"
Method      :
Segment     : 2498
--------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                      Date
------------ ------------ ---- ----------------------------------------- ----------
           0         2764 *  L   "k1234567"."table701"                   2005/06/04
================================================================================
No          : 5
Rdarea Name : "RDUSER02"
Method      : Expand
Segment     : 31854
--------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                      Date
------------ ------------ ---- ----------------------------------------- ----------
        3503        23934    T   "k1234567"."table21"                    2005/06/11
        1592         3503    I   "k1234567"."index22"                    2005/06/11
        2548          955    T   "k1234567"."table23"                    2005/06/11
================================================================================
```

注

HiRDB/パラレルサーバの場合，DB メンテナンス予定日は，ディクショナリサーバの日時を基に予測しています。

[説明]

No：

解析結果番号です（最大 10 けた）。DB メンテナンス予定日の情報の No と対応しています。

Rdarea Name：

メンテナンスが必要と予測した RD エリアの名称です（最大 32 文字）。

Method：

RD エリアに対するメンテナンス方法です（最大 20 文字）。

Expand：

RD エリアの拡張をしてください。

なお，該当 RD エリアが次の条件をすべて満たす場合，実際には自動増分できるときでも，メンテナンス方法に Expand と表示されることがあります。

- HiRDB ファイルシステム領域を自動的に拡張する方式によって，RD エリアの自動増分機能を適用している。
- 上記 HiRDB ファイルシステム領域を，ダイレクトディスクアクセス上に作成している。

このため，上記に該当する RD エリアについて Expand が表示された場合は，該当する HiRDB ファイルシステム領域の容量，及び該当する領域を作成したダイレクトディスクアクセスの容量を確認してください。HiRDB ファイルシステムの自動増分に必要な容量がダイレクトディスクアクセスに残っている場合は，RD エリアを拡張する必要はありません。

HiRDB ファイルシステム領域の容量については，pdfstatfs で確認してください。

ダイレクトディスクアクセスの容量については，OS の DiskPart コマンドで確認してください。DiskPart の詳細については OS のマニュアルを参照してください。

Extend：

メンテナンスは不要です（HiRDB が RD エリアの自動増分をします）。

Reinit：

RD エリアの再初期化をしてください。

空白：

メンテナンスは不要です。

Segment：

解放セグメント数です（最大 10 けた）。

- 予測レベル 1 の場合

  メンテナンスで最低限確保する必要があるセグメント数です。

  Method に Expand が表示されている場合，ここに表示されているセグメント数に更に余裕を持たせた値になるように拡張してください。

- 予測レベル 2 の場合

  メンテナンスで最低限確保する必要があるセグメント数です。ただし，表又はインデクスのメンテナンスが必要と予測した場合は，必要なメンテナンスの実行結果を考慮した値が表示されます。例えば，Method に Expand が表示されている場合，RD エリアを拡張する前に，必要なメンテナンスを実行していることを前提にして，最低限確保する必要があるセグメント数が表示されます。

Reclaim：

使用中空きページの解放をした後，使用中空きセグメントの解放を実行したときにできる，解放セグメント数の予測値です（最大 13 けた）。

表又はインデクスのメンテナンス方法として，使用中空きセグメントの解放が必要と予測した場合，この項目の右側に*を出力します。

Reorganize：

再編成を実行した場合にできる解放セグメント数の予測値です（最大 13 けた）。

表又はインデクスのメンテナンス方法として，再編成が必要と予測した場合，この項目の右側に*を出力します。

なお，再編成をするとデータの格納ページ数が多くなると予測した場合は，負の値を出力します。

Type：

メンテナンス対象の種別です（1 文字）。

T：表

I：インデクス

L：LOB 用 RD エリア

Name：

メンテナンス対象の表又はインデクス（認可識別子を含む）の名称です（最大 65 文字）。Type が L の場合，LOB 列を定義している表の名称を出力します。

出力形式は，"認可識別子"."表識別子"又は"認可識別子"."インデクス識別子"となります。なお，ディクショナリ表の場合，認可識別子は"(Data dictionary)"となります。

Date：

DB メンテナンス予定日です（YYYY/MM/DD の形式）。

予測レベル 2 の場合の補足

予測レベル 2（-e 2）の場合，出力情報が次の状態になることがあります。これらは，解析項目別情報の表又はインデクスがメンテナンス要と判断された場合に表示されます。

1. 再編成すると，逆に使用セグメント数が増える資源に対して，再編成要「*」と指示される。
2. 同一 RD エリアで，メンテナンス時期が異なる。
3. RD エリアの拡張と再編成の両方を指示される。

上記の出力例を次に示します。上記の番号は，出力例の[1]～[3]と対応しています。

```
 pddbst VV-RR       ***** Maintenance Information *****       yyyy/mm/dd hh:mm:ss
No          : 1
Rdarea Name : ”RDUSER01”
Method      :
Segment     : 42560
------------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                        Date
------------- ------------ ---- ------------------------------------------ ----------
          3           10 *  T   ”k1234567”.”table10”                  [2] 2005/05/16
      28089 *       9363    T   ”k1234567”.”table01”                      2005/05/31
          0        12342 *  T   ”k1234567”.”table02”                      2005/05/31
        851         7535 *  I   ”k1234567”.”index01”                      2005/05/31
          3      [1] -10 *  T   ”k1234567”.”table11”                      2005/05/31
          3     [1] -300 *  T   ”k1234567”.”table12”                      2005/05/31
====================================================================================
No          : 2
Rdarea Name : ”rd0002”
Method      : Expand [3]
Segment     : 1523
------------------------------------------------------------------------------------
 Reclaim      Reorganize   Type Name                                        Date
------------- ------------ ---- ------------------------------------------ ----------
        228     [3] 1035 *  T   ”k1234567”.”table01”                      2005/06/03
          0      [3] 121 *  T   ”k1234567”.”table03”                      2005/06/03
         30       [3] 60 *  I   ”k1234567”.”index07”                      2005/06/03
          3     [3] -300 *  T   ”k1234567”.”table22”                      2005/06/03
====================================================================================
```

(3) 解析項目別情報

監視期間内にセグメント使用率が基準値を超える RD エリア，表，及びインデクスを，DB メンテナンス予定日が早い順に出力します。このとき，メンテナンス方法も出力します。また，メンテナンス方法の情報に出力されたリソースについても，詳細情報を出力します。

```
 pddbst VV-RR       ***** Analysis Item Data *****       yyyy/mm/dd hh:mm:ss
No        : 1
Date      : 2005/05/25
Method    : Reclaim
Type      : T
Name      : "k1234567"."table01" ("RDUSER01")
StateDate : 2005/05/19 00:10:32
-------------------------------------------------------------------------------
Information                           Value      Method               Date
------------------------------------ ---------- -------------------- ----------
Empty Page Ratio                      40 ( 50) * Reclaim pages        2005/05/25
Unused Page Ratio (30)                20 ( 30)
Used Segment for Cluster              40 ( 50)
Unused Page Differ From PCTFREE (10,10)  50 ( 75)
===============================================================================
No        : 2
Date      : 2005/05/31
Method    :
Type      : R
Name      : "RDUSER01"
StateDate : 2005/05/19 00:10:32
-------------------------------------------------------------------------------
Information                           Value      Method               Date
------------------------------------ ---------- -------------------- ----------
Used Segment Ratio (0,10)             40 ( 80)                        2005/05/31
===============================================================================
No        : 3
Date      : 2005/06/01
Method    : Reorganize
Type      : T*
Name      : "k1234567"."table10" ("RDUSER01")
StateDate : 2005/05/19 00:10:32
-------------------------------------------------------------------------------
Information                            Value     Method               Date
------------------------------------ ---------- -------------------- ----------
Empty Page Ratio                      25 ( 50)   Reclaim pages        2005/08/10
Unused Page Ratio (30)                25 ( 30) * Reorganize           2005/06/01
Used Segment for Cluster              40 ( 50)
Unused Page Differ From PCTFREE (10,10)  60 ( 75)
===============================================================================
No        : 4
Date      : 2005/06/03
Method    : Expand
Type      : R
Name      : "rd0002"
StateDate : 2005/05/19 00:10:32
-------------------------------------------------------------------------------
Information                           Value      Method               Date
------------------------------------ ---------- -------------------- ----------
Used Segment Ratio (0,10)             78 ( 95)   Expand               2005/06/03
```

```
================================================================================
No        : 5
Date      : 2005/06/04
Method    : Reclaim
Type      : I
Name      : "k1234567"."index07" ("rd0002")
StateDate : 2005/05/19 00:10:32
--------------------------------------------------------------------------------
Information                              Value     Method              Date
---------------------------------------- --------- ------------------- ----------
Empty Page Ratio                         38 ( 50)  Reclaim             2005/06/04
Unused Page Ratio (30)                   20 ( 30)
Unused Page Differ From PCTFREE (10,10)  70 ( 75)
Delete RowID Ratio                       23 ( 30)  ReclaimPx           2005/06/07
================================================================================
No        : 6
Date      : 2005/06/04
Method    : Reorganize
Type      : L
Name      : "k77a777"."table707" ("lobrdarea701")
StateDate : 2005/05/19 00:10:32
--------------------------------------------------------------------------------
Information                              Value     Method              Date
---------------------------------------- --------- ------------------- ----------
Used Segment for LOB Columns (0,10)      70 ( 80) * Reorganize         2005/06/04
================================================================================
No        : 7
Date      : 2005/06/06
Method    : Expand
Type      : R
Name      : "RDUSER12"
StateDate : 2005/05/19 00:10:32
--------------------------------------------------------------------------------
Information                              Value     Method              Date
---------------------------------------- --------- ------------------- ----------
Used Segment Ratio (0,10)                70 ( 80) * Expand              2005/07/01
================================================================================
      :
```

注

HiRDB/パラレルサーバの場合，DB メンテナンス予定日は，ディクショナリサーバの日時を基に予測しています。

[説明]

No：

解析結果番号です（最大 10 けた）。DB メンテナンス予定日の情報の No と対応しています。

Method：

対象種別（Type）に対するメンテナンス方法です（最大 14 文字）。

ReclaimS：

pdreclaim で使用中空きセグメントの解放をしてください。

ReclaimP：

pdreclaim で使用中空きページの解放をしてください。

Reorganize：

pdrorg で再編成をしてください。再編成を実行した場合，PCTFREE の指定値に従って空き領域を確保するため，使用する領域が増えることがあります。領域が不足する場合は，事前に RD エリアの拡張をしてください。

Expand：

pdmod で RD エリアの拡張をしてください。

なお，該当 RD エリアが次の条件をすべて満たす場合，実際には自動増分できるときでも，メンテナンス方法に Expand と表示される場合があります。

- HiRDB ファイルシステム領域を自動的に拡張する方式によって，RD エリアの自動増分機能を適用している
- 上記 HiRDB ファイルシステム領域を，ダイレクトディスクアクセス上に作成している。

このため，上記に該当する RD エリアについて Expand が表示された場合は，該当する HiRDB ファイルシステム領域の容量，及び該当する領域を作成したダイレクトディクスアクセスの容量を確認してください。HiRDB ファイルシステムの自動増分に必要な容量がダイレクトディスクアクセスに残っている場合は，RD エリアを拡張する必要はありません。

HiRDB ファイルシステム領域の容量については，pdfstatfs で確認してください。

ダイレクトディスクアクセスの容量については，OS の DiskPart コマンドで確認してください。DiskPart の詳細については OS のマニュアルを参照してください。

Extend：

メンテナンスは不要です（HiRDB が RD エリアの自動増分をします）。ただし，自動増分に必要な領域を確保できない場合は，ユーザが pdmod で RD エリアを拡張する必要があります。

Reinit：

RD エリアの再初期化をしてください。RD エリアに既に 16 個の HiRDB ファイルが割り当てられている場合，RD エリアの拡張はできないため，RD エリアの再初期化をして，HiRDB ファイルの容量や個数を変更するなどの対処が必要です。

ReclaimPx：

-x オプションを指定した pdreclaim でインデクスの使用中空きページの解放をしてください。

空白：

メンテナンスは不要です。

Type：

メンテナンス要と判断された対象種別です（2 文字）。

T：表

I：インデクス

L：LOB 用 RD エリア

R：RD エリア

解析項目別情報にだけ出力されていて，メンテナンス方法の情報に出力されていない対象種別には*が付きます（例：T*）。したがって，*が付いている対象種別については，提示されたメンテナンスを実施する必要があります。また，*が付いていない対象種別については，メンテナンス方法の情報に出力されているメンテナンスだけ実施してください。

Name：

メンテナンス対象の RD エリア，表，又はインデクス（認可識別子を含む）の名称です（最大 78 文字）。

RD エリアの場合：

RD エリアの拡張を指示するとき："RD エリア名"

LOB 用 RD エリアのとき："認可識別子"."表識別子"("RD エリア名")

表の場合：

非横分割表のとき："認可識別子"."表識別子"

横分割表のとき："認可識別子"."表識別子"("RD エリア名")

インデクスの場合：

非横分割インデクスのとき："認可識別子"."インデクス識別子"

横分割インデクスのとき："認可識別子"."インデクス識別子"("RD エリア名")

StateDate：

分析に使用した解析情報のうち，最新の情報の取得日時です（YYYY/MM/DD hh:mm:ss の形式）。

Information：

解析した項目の項目名と，解析に使用した基準値を出力します。項目名と基準値については，「基準値定義ファイル（再編成時期予測機能）」を参照してください。

Value：

解析値です。解析値は，解析値（基準値）の形式で出力します。基準値については，「基準値定義ファイル（再編成時期予測機能）」を参照してください。

予測の結果，メンテナンスが必要となる解析項目のうち，予測日が最も早い項目には*付きの解析値が表示されます。

Date：

DB メンテナンス予定日です（YYYY/MM/DD の形式）。

15.4.2　CSV 形式での出力形式

再編成時期予測機能実行時に predict 文を指定すると，予測結果を CSV 形式のファイルに出力できます。

(1)　出力するデータの規則

1. CSV 形式で出力する場合の出力規則については，「コマンド実行結果を DAT 形式で出力する場合の規則」を参照してください。
2. 1 行分のデータの最後は改行します（改行文字は 0x0d0a）。

3. 先頭の行には，タイトルの行を出力します。

(2)　予測結果の出力項目

CSV 形式で出力した場合の再編成時期予測機能の実行結果を次の表に示します。

表 15–9　再編成時期予測機能の実行結果（CSV 形式で出力した場合）

タイトル	出力項目	出力形式	最大長(単位：バイト)	出力情報		
				DB メンテナンス予定日の情報	メンテナンス方法の情報	解析項目別情報
kind	出力種別 **p**：DB メンテナンス予定日の情報 **m**：メンテナンス方法の情報 **d**：解析項目別情報	文字列	8	○	○	○
No	解析結果番号	数値	10	○	○	○
Date	DB メンテナンス予定日	YYYY/MM/DD	10	○	○	○
Type	対象種別 **T**：表 **I**：インデクス **L**：LOB 用 RD エリア **R**：データディクショナリ用 RD エリア，ユーザ用 RD エリア，又はレジストリ用 RD エリア	文字列	1	○	○	○
*	メンテナンスが必要と判断したリソースに対して*を出力します。 ユーザは，*が出力されたリソースに対して，メンテナンスをする必要があります。	文字	1	空白	○	空白
AuthID	認可識別子 ディクショナリ表の場合，(Data dictionary)となります。	文字列	30	×	○※	○※
Name	表又はインデクスの名称	文字列	30	×	○※	○※
Rdarea	RD エリア名	文字列	30	○	○	○
Method	メンテナンス方法の番号 **0**：メンテナンス不要 **1**：ReclaimS（使用中空きセグメントの解放） **2**：ReclaimP（使用中空きページの解放）	数値	5	×	○	○

タイトル	出力項目	出力形式	最大長(単位：バイト)	出力情報		
				DB メンテナンス予定日の情報	メンテナンス方法の情報	解析項目別情報
	3：Reorganize（再編成） **4**：Expand（RD エリアの拡張） **5**：Extend（RD エリアの自動増分（メンテナンス不要）） **6**：Reinit（RD エリアの再初期化） **7**：ReclaimPx（-x オプションを指定したインデクスの使用中空きページの解放） メンテナンス方法については，「実行結果の出力形式」を参照してください。					
NextExec	次回の状態解析結果蓄積機能の実行推奨時期	YYYY/MM/DD	10	×	×	×
StateDate	解析情報取得日時。 分析に使用した解析情報のうち，最新の情報の取得日時となります。	YYYY/MM/DD hh:mm:ss	19	○	×	○
InfoNo	解析項目種別の番号 **1**：Empty Page Ratio **2**：Unused Page Ratio **3**：Number of Branch Row **8**：Used Segment for LOB Columns **10**：Used Segment for Cluster **11**：Unused Page Differ From PCTFREE **12**：Delete RowID Ratio **13**：Used Segment Ratio	数値	5	○	×	○
Value	解析項目種別に対する解析値	数値	5	○	×	○
PredictBase	解析に使用した解析項目種別の基準値	数値	5	○	×	○
Count	蓄積データ数	数値	5	×	×	×
Segment	解放セグメント数	数値	10	×	○	×
Reclaim	使用中空きセグメントの解放での解放セグメント予測数	数値	10	×	○※	×
Reorganize	再編成での解放セグメント予測数	数値	10	×	○※	×
ItemMethod	解析項目ごとのメンテナンス方法の番号	数値	5	×	○	○
CheckNo	チェック用基準値数	数値	5	○	×	○

タイトル	出力項目	出力形式	最大長(単位：バイト)	出力情報		
				DB メンテナンス予定日の情報	メンテナンス方法の情報	解析項目別情報
Check1	解析に使用した解析項目種別のチェック用基準値	数値	5	○	×	○
Check2	解析に使用した解析項目種別のチェック用基準値	数値	5	○	×	○

（凡例）

○：出力します。

×：出力しません。

空白：空白を 1 文字出力します。

注

出力形式が文字列の場合，引用符（"）で囲んで出力します。なお，最大長に引用符は含まれません。

注※

対象種別が R の場合，出力しません。

(3) 出力例

CSV 形式の出力例を次に示します。

(a) -e 1 指定，-m オプション省略時

```
Kind,No,Date,Type,*,AuthID,Name,Rdarea,Method,NextExec,StateDate,InfoNo,Value,PredictBase,
Count,Segment,Reclaim,Reorganize,ItemMethod,CheckNo,Check1,Check2
"p",1,2005/01/26,"R"," ",,,"RDUSER12",,,2004/12/27 15:58:08,13,50,80,,,,,2,0,0
"p",1,2005/01/30,"R"," ",,,"rd0002",,,2004/12/27 15:58:08,13,60,80,,,,,2,0,0
"p",1,2005/01/11,"R"," ",,,"lobrdarea701",,,2004/12/27 15:58:08,13,70,80,,,,,2,0,0
"p",1,2005/01/15,"R"," ",,,"RDUSER02",,,2004/12/27 15:58:08,13,65,80,,,,,2,0,0
```

(b) -e 1，-m オプション指定時

```
Kind,No,Date,Type,*,AuthID,Name,Rdarea,Method,NextExec,StateDate,InfoNo,Value,PredictBase,
Count,Segment,Reclaim,Reorganize,ItemMethod,CheckNo,Check1,Check2
"p",1,2005/01/26,"R"," ",,,"RDUSER01",,,2004/12/27 15:58:08,13,50,80,,,,,2,0,0
"m",1,2005/01/26,"R"," ",,,"RDUSER01",0,,,,,,,2,0,0,0,,,
"m",1,2005/01/26,"I"," ","k1234567","index01","RDUSER01",0,,,,,,,2,0,1,0,,,
"m",1,2005/01/26,"T","*","k1234567","table01","RDUSER01",3,,,,,,,2,0,4,3,,,
            :
"p",1,2005/01/30,"R"," ",,,"RDUSER02",,,2004/12/27 15:58:08,13,50,80,,,,,2,0,0
"m",1,2005/01/30,"R"," ",,,"RDUSER02",0,,,,,,,0,0,0,0,,,
"m",1,2005/01/30,"T"," ","k1234567","table03","RDUSER02",0,,,,,,,0,0,0,0,,,
            :
"p",1,2005/01/21,"R"," ",,,"RDUSER02",,,2004/12/27 15:58:08,13,50,80,,,,,2,0,0
```

```
"m",1,2005/01/21,"R","*",,,"RDUSER02",4,,,,,,6,0,0,4,,,
"m",1,2005/01/21,"I"," ","k1234567","index03","RDUSER02",0,,,,,,6,0,1,0,,,
"m",1,2005/01/21,"T"," ","k1234567","table03","RDUSER02",0,,,,,,6,0,4,0,,,
            :
```

(c)　-e 2 指定時

```
Kind,No,Date,Type,*,AuthID,Name,Rdarea,Method,NextExec,StateDate,InfoNo,Value,PredictBase,Ch
eckNo,Check1,Check2
"p",1,2005/05/31,"R",,,,"RDUSER01",,2005/05/28,2005/04/12 17:30:12:42,13,78,80,2,0,10
"m",1,2005/05/31,"T","*","k1234567","table01","RDUSER01",1,,,,,,0,,
"m",1,2005/05/31,"T",,"k1234567","table02","RDUSER01",2,,,,,,0,,
"m",1,2005/05/31,"I","*","k1234567","index01","RDUSER01",2,,,,,,0,,
"m",1,2005/05/31,"T","*","k1234567","table10","RDUSER01",2,,,,,,0,,
"d",1,2005/05/31,"T",,"k1234567","table01","RDUSER01",1,2005/05/28,2005/04/12 17:30:12:42,1,
40,50,0,,
"d",1,2005/08/10,"T","*","k1234567","table10","RDUSER01",1,2005/07/31,2005/04/12 17:30:12:42
,1,25,50,0,,
"d",1,2005/06/01,"T","*","k1234567","table10","RDUSER01",2,2005/05/29,2005/04/12 17:30:12:42
,2,25,30,1,50,
"p",2,2005/06/01,"R",,,,"RDUSER12",,2005/05/29,2005/04/12 17:30:12:42,13,78,80,2,0,10
"m",2,2005/06/01,"T","*","k1234567","table01","RDUSER12",2,,,,,,0,,
"m",2,2005/06/01,"T","*","k1234567","table03","RDUSER12",2,,,,,,0,,
"m",2,2005/06/01,"I","*","k1234567","index01","RDUSER12",1,,,,,,0,,
"m",2,2005/06/01,"T","*","k1234567","table11","RDUSER12",1,,,,,,0,,
"p",3,2005/06/03,"R",,,,"rd0002",,2005/05/30,2005/04/12 17:30:12:42,13,78,80,2,0,10
"m",3,2005/06/03,"R",,,,"rd0002",3,,,,,,0,,
"d",3,2005/06/03,"R",,,,"rd0002",3,2005/05/30,2005/04/12 17:30:12:42,13,78,80,2,0,10
            :
```

15.5　基準値定義ファイル（再編成時期予測機能）

15.5.1　形式および規則

再編成時期予測機能では，あらかじめ決まっている各項目の基準値を基に予測していますが，この基準値はユーザの環境に合わせて変更できます。基準値定義ファイルには，変更後の各項目の基準値を指定します。

(1)　形式

```
<threshold>
項目名=基準値
      :
[rdarea=RDエリア名]
項目名=基準値
      :
[table=認可識別子.表識別子]
項目名=基準値
      :
[index=認可識別子.インデクス識別子]
項目名=基準値
      :
```

(2)　説明

<threshold>

共通項目であることを意味しています。ファイルの先頭行には必ず<threshold>を指定してください。

[rdarea=RD エリア名]

RD エリアごとに基準値を変える場合に指定します。

[table=認可識別子.表識別子]

表ごとに基準値を変える場合に指定します。

指定した認可識別子のすべての表を対象としたい場合は，all を指定してください。ディクショナリ表を指定する場合は，認可識別子を省略します。

[index=認可識別子.インデクス識別子]

インデクスごとに基準値を変える場合に指定します。

指定した認可識別子のすべてのインデクスを対象としたい場合は，all を指定してください。

項目名=基準値

基準値を変更する項目名と基準値を指定します。指定できる項目名については，「(4)基準値を変更できる項目名」を参照してください。

＜規則＞

1. 同じ項目名を複数指定した場合，最後の指定が有効となります。

2. 指定値に*を指定した場合，その項目を再編成時期予測機能の判定項目から除外します。定期的に，PURGE TABLE や初期データロード（既存データを削除して pdload）を行う表の場合，再編成をする必要がありません。そのため，これらの資源を再編成時期予測機能の対象外とするときに，*を指定してください。再編成時期予測機能は，PURGE TABLE などが定期的に実行されなかった場合を想定して，予測を行っているためです。

(3)　基準値定義ファイル中の指定規則

1. 最初の[rdarea=RD エリア名]，[table=認可識別子.表識別子]，又は[index=認可識別子.インデクス識別子]の前に指定した項目は，共通の指定として扱われます。
2. <threshold>，[rdarea=RD エリア名]，[table=認可識別子.表識別子]，及び[index=認可識別子.インデクス識別子]に同一項目がある場合，最後の指定が有効となります。
3. 同じ[rdarea=RD エリア名]，[table=認可識別子.表識別子]，又は[index=認可識別子.インデクス識別子]が複数あり，その中に同じ項目名を指定している場合，最後の指定が有効となります。
4. <threshold>，[rdarea=RD エリア名]，[table=認可識別子.表識別子]，[index=認可識別子.インデクス識別子]，及び項目名=基準値は，それぞれ 1 行で指定してください。
5. 「項目名」，「=」，「指定値」の間，及び「項目名」の前には，空白及びタブは指定できません。
6. /*と*/で囲まれた部分は無視されます。また，空行も無視されます。
7. 基準値定義ファイルの内容に誤りがある場合，エラー終了します。
8. RD エリア名，認可識別子，表識別子，又はインデクス識別子に，英小文字又は空白が含まれる場合は，「¥""」と「"¥"」で囲んでください。英小文字を「¥""」と「"¥"」で囲まないと，すべて大文字として扱われます。
9. 全表共通，及び全インデクス共通（表識別子，インデクス識別子に all を指定）は，指定表共通，指定インデクス共通，及び指定 RD エリア共通の前に指定してください。

(4)　基準値を変更できる項目名

基準値を変更できる項目と基準値を変更する目安を次の表に示します。

表 15–10　基準値を変更できる項目

項番	項目名	指定範囲	基準値	内容	変更可否		
					R	T	I
1	EMPTY_PAGE_RATIO	*， 10～100	30	総割り当てページ数に対して，pdreclaim で解放できるページ数の比率（単位：%）を指定します。	○	○	○
2	UNDER_PAGE_RATIO	10～100	10， 又は 80-PCTFREE	項番 3 の判定に使用するページ使用率（単位：%）を指定します。	○	○	○

項番	項目名	指定範囲	基準値	内容	変更可否		
					R	T	I
3	UNUSED_PAGE_RATIO	*， 10～100	50	ページ使用率が項番 2 の指定値以下のページ数と総割り当てページ数の比率（単位：%）を指定します。	○	○	○
4	BRANCH_ROW	*， 10～100	50	複数ページにわたって格納された行数と総行数の比率（単位：%）を指定します。	○	○	×
5	USED_SEGMENT_LOB	*， 10～100	80	LOB 用 RD エリアの最終セグメント番号と総セグメント数の比率（単位：%）を指定します。	○	×	×
6	USED_SEGMENT_RATIO_CLUS	*， 10～100	50	クラスタリングデータページに割り当てられた全セグメント数に対する，使用率 100%のセグメント数の比率（単位：%）を指定します。	○	○	×
7	DIFF_PCTFREE_M	*， 10～100	10	ページ使用率の下限値を決める値（単位：%）を指定します。 指定する値は，PCTFREE に指定した未使用領域比率を基に算出したページ使用率から，ページ使用率の下限値を引いた値となります。 指定値が（100−PCTFREE に指定した未使用領域比率）より大きい場合，（100−PCTFREE に指定した未使用領域比率）が指定値として仮定されます。 *を指定した場合は，項番 8 の指定値だけ有効になります。	○	○	○
8	DIFF_PCTFREE_P	*， 10～100	10	ページ使用率の上限値を決める値（単位：%）を指定します。 指定する値は，ページ使用率の上限値から，PCTFREE に指定した未使用領域比率を基に算出したページ使用率を引いた値となります。 指定値が PCTFREE に指定した未使用領域比率より大きい場合，PCTFREE に指定した未使用領域比率が指定値として仮定されます。 *を指定した場合は，項番 7 の指定値だけ有効になります。	○	○	○

項番	項目名	指定範囲	基準値	内容	変更可否		
					R	T	I
9	DIFF_PCTFREE_RATIO	*， 10～100	*	ページ使用率が項番 7 の下限値以下のページと，項番 8 の上限値以上のページの数と総割り当てページ数の比率（単位：%）を指定します。省略した場合は，この項目についての解析はしません。	○	○	○
10	DELETE_ROWID_RATIO	*， 1～100	30	インデクスが管理する行の総数の中で，残存エントリが管理する削除行数の比率（単位：%）を指定します。	○	×	○
11	USED_SEGMENT	*， 10～100	80	RD エリア全体の使用セグメント数と総セグメント数の比率（単位：%）を指定します。	○	×	×
12	EXTEND_COUNT_MIN	0～24	0	RD エリアの自動増分で増分できる最低限の増加回数を指定します。EXTEND_COUNT_MIN≦EXTEND_COUNT になるようにしてください。	○	×	×
13	EXTEND_COUNT	0～24	0	RD エリアの自動増分で増分できる増分回数を指定します。EXTEND_COUNT_MIN≦EXTEND_COUNT になるようにしてください。自動増分した結果，HiRDB ファイルシステム領域が満杯になるかどうかは，pddbst ではチェックできません。そのため，EXTEND_COUNT の指定値は，HiRDB ファイルシステム領域中の RD エリア数と最大増分回数を考慮し，決める必要があります。	○	×	×

（凡例）

R：RD エリア

T：表

I：インデクス

○：該当する項目名の基準値を変更できます。

×：該当する項目名の基準値を変更できません。

表 15–11　基準値を変更する目安

項番	項目名	基準値	メンテナンス要否の判定方法	基準値を変更する目安
1	EMPTY_PAGE_RATIO	(p)=30%	ページ使用率 0%の使用中ページの割合が(p)%以上になると，空きページの解放が必要と判断します。	RD エリアの空き容量が少ない場合で，無効なページを少しでも早く解放したいときは，(p)を小さくしてください。
2	UNDER_PAGE_RATIO	(a)=X-20%	ページ使用率が(a)%以下の使用ページ数の割合が，全使用中ページの(p)%以上になると，再編成が必要と判断します。	• ページサイズに対して格納行長が極端に短い場合で，再編成した直後でもページ使用率が(a)%に満たない表のときは，(a)の値を小さくしてください。 • 無効領域があるページの割合をできるだけ小さくしたい場合，(p)の値を小さくしてください。 • NO SPLIT 指定のない定義長 256 バイト以上の可変長文字列がある表の場合で，再編成した直後でもページ使用率が(a)%以下のページが 50%以上あるとき（分岐した可変長文字列データの方が，分岐しないデータよりも多い状態），(p)を大きくしてください。 • RD エリアの空き容量が少ない場合で，無効領域があるページを少しでも早く解放したいときは，(p)を小さくしてください。
3	UNUSED_PAGE_RATIO	(p)=50%		
4	BRANCH_ROW	(p)=50%	総行数に対する，分岐行の行数の割合が(p)%以上になると，再編成が必要と判断します。	データ長が長くなる UPDATE 処理が多い場合で，検索性能の維持が必要なときは，(p)の値を小さくしてください。
5	USED_SEGMENT_LOB	(p)=80%	LOB 用 RD エリアの最終セグメント番号が，総セグメント数の(p)%以上となった場合，再編成して(p)%以上になるときは RD エリアの拡張が必要と判断します。また，再編成して(p)%未満となる場合は，再編成が必要と判断します。	急激に格納データ量が増加する可能性がある場合で，早めに対処する必要があるときは，(p)を小さくしてください。
6	USED_SEGMENT_RATIO_CLUS	(p)=50%	セグメント内の使用ページ率が 100%のセグメント（空きページがないセグメント）数が，総使用セグメント数の割合(p)%以上になると，未使用領域比率による空きではなく，未使用ページ比率による空きの消費が始まっているので，再編成が必要と判断します。	同一ページ内にデータが格納できる間はクラスタリング効果が保たれているが，同一セグメントに格納する状態が既にクラスタリングの効果がなくなっていると判断できる場合，(p)を小さくしてください。

項番	項目名	基準値	メンテナンス要否の判定方法	基準値を変更する目安
7	DIFF_PCTFREE_M	(a)=X-10%	ページ使用率が(a)%～(b)%の範囲外にある使用ページ数の割合が，総使用ページ数の(p)%以上になると，再編成が必要と判断します。	ページ使用率が高い状態であっても再編成を必要と判断したい場合（行長が長くなるUPDATE が頻発する表や，クラスタキーを定義した表への中間キーを持つデータのINSERT が頻発する表）には，(p)を指定してください。なお，重複キーを多量に持つインデクスの場合，重複キーを格納するページは PCTFREE を無視して格納されるため，(p)を指定する場合は大きめの値にしてください。
8	DIFF_PCTFREE_P	(b)=X+10%		
9	DIFF_PCTFREE_RATIO	(p)=*		
10	DELETE_ROWID_RATIO	(p)=30%	インデクスが管理する行の総数の中で，残存エントリが管理する削除行数の比率が(p)%以上となった場合，再編成が必要と判断します。	• インデクスの残存エントリによるオーバヘッドを削減したい場合，及び残存エントリによって発生した空き領域を少しでも早く解放したい場合は，(p)の値を小さくしてください。 • 削除したキー値と同一のキー値をすぐに格納する場合は，インデクスの残存エントリが再利用されますので，(p)の値を大きくしてください。
11	USED_SEGMENT	(p)=80%	RD エリアの使用セグメント数が総セグメント数の(p)%以上となった場合，再編成が必要と判断します。 再編成後でも(p)%以上となる場合は，拡張が必要と判断しますが，自動増分指定がある RD エリアでは(a)回まで自動増分ができると判断し，拡張の指示はしません。	• RD エリアの拡張が指示された場合に，早めに容量不足を検知できるようにしたいときは，(p)の値を小さくしてください（例えば，予備のディスクがない場合に容量不足が発生したとき，ディスクの手配から開始しないといけないケースなど）。 • 急激に格納データ量が増加する可能性がある場合に，早めに対処したいときは，(p)を小さくしてください。
12	EXTEND_COUNT_MIN	(a)=23 回		
13	EXTEND_COUNT			

（凡例）

X：100−PCTFREE の第 1 引数の指定値

(5) 基準値定義ファイルの指定例

基準値定義ファイルの指定例を次に示します。

```
<threshold> ..................................共通指定
USED_SEGMENT=70
USED_SEGMENT_LOB=65
  :
[table=all] ..................................全表共通指定
EMPTY_PAGE_RATIO=30
USED_SEGMENT=20
  :
[index=all] ..................................全インデクス共通指定
EMPTY_PAGE_RATIO=20
```

```
USED_SEGMENT=20
  :
[table=認可識別子.表名称] .....................指定表共通指定
EMPTY_PAGE_RATIO=35
USED_SEGMENT=30
  :
[index=認可識別子.インデクス名称] .............指定インデクス共通指定
EMPTY_PAGE_RATIO=80
USED_PAGE_RATIO=60
  :
[rdarea=ＲＤエリア名称] .......................RDエリア個別指定
EXTEND_COUNT=10
  :
[table=認可識別子.表名称] .....................RDエリア内の表個別指定
EMPTY_PAGE_RATIO=35
USED_SEGMENT=30
  :
[index=認可識別子.インデクス名称] .............RDエリア内のインデクス個別指定
EMPTY_PAGE_RATIO=80
USED_PAGE_RATIO=60
  :
```

15.6　予測レベル 2 での状態解析結果蓄積機能の実行方法

予測レベル 1 の状態解析結果蓄積機能では，RD エリア単位の情報だけ取得するため，ディレクトリページだけ解析します。そのため，実行時間は比較的短時間で済みます。

予測レベル 2 の状態解析結果蓄積機能では，更に表単位，及びインデクス単位の情報も取得するため，データページ，及びインデクスページの解析も必要となります。そのため，予測レベル 1 に比べて実行時間が長くなります。

オンライン業務中に，予測レベル 2 の状態解析結果蓄積機能を実行する場合，業務 AP へ影響を与える可能性もあります。オンライン中の業務 AP への影響を最小限にしたい場合は，次のどちらかの実行方法を選択できます（二つ同時にも指定できます）。

- **インターバル解析**
- **マージ解析**

これらの実行方法と選択基準を次に説明します。

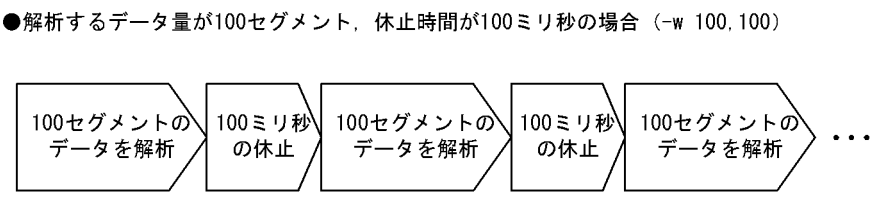
15.6.1　インターバル解析

インターバル解析は，1 回の状態解析結果蓄積機能の実行中に，休止時間を設ける実行方法です。指定したデータ量の解析を行った後，指定した時間だけ解析を休止します。指定した休止時間が過ぎたら，再度データ量の解析を行い，これを繰り返します。一定間隔で解析処理を休止させるため，オンライン業務への影響を少なくできます。

インターバル解析を実行する場合は，-w オプションで「解析するデータ量（単位：セグメント数）」と「休止時間（単位：ミリ秒）」を指定します。

インターバル解析の実行の概要を次の図に示します。

図 15–13　インターバル解析の実行の概要



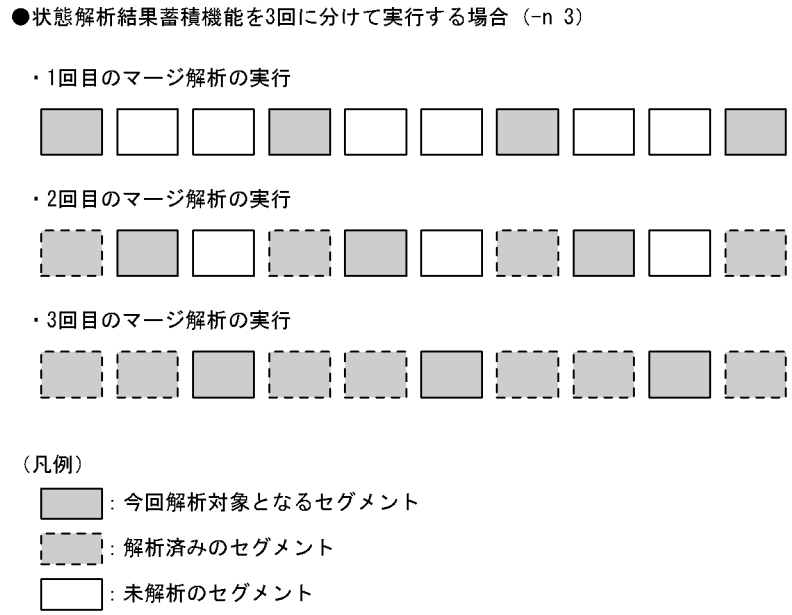
15.6.2　マージ解析

マージ解析は，複数回（数日）に分けて状態解析結果蓄積機能を実行する方法です（状態解析結果は複数回分マージして蓄積します）。例えば，オンライン業務が比較的空いている時間帯がある場合，数日に分けて実行すると効果的です。

マージ解析を実行する場合は，-n オプションに実行する回数を指定します。

マージ解析の実行の概要を次の図に示します。なお，この図では，1RD エリア（10 セグメント）を解析対象としていることを想定しています。

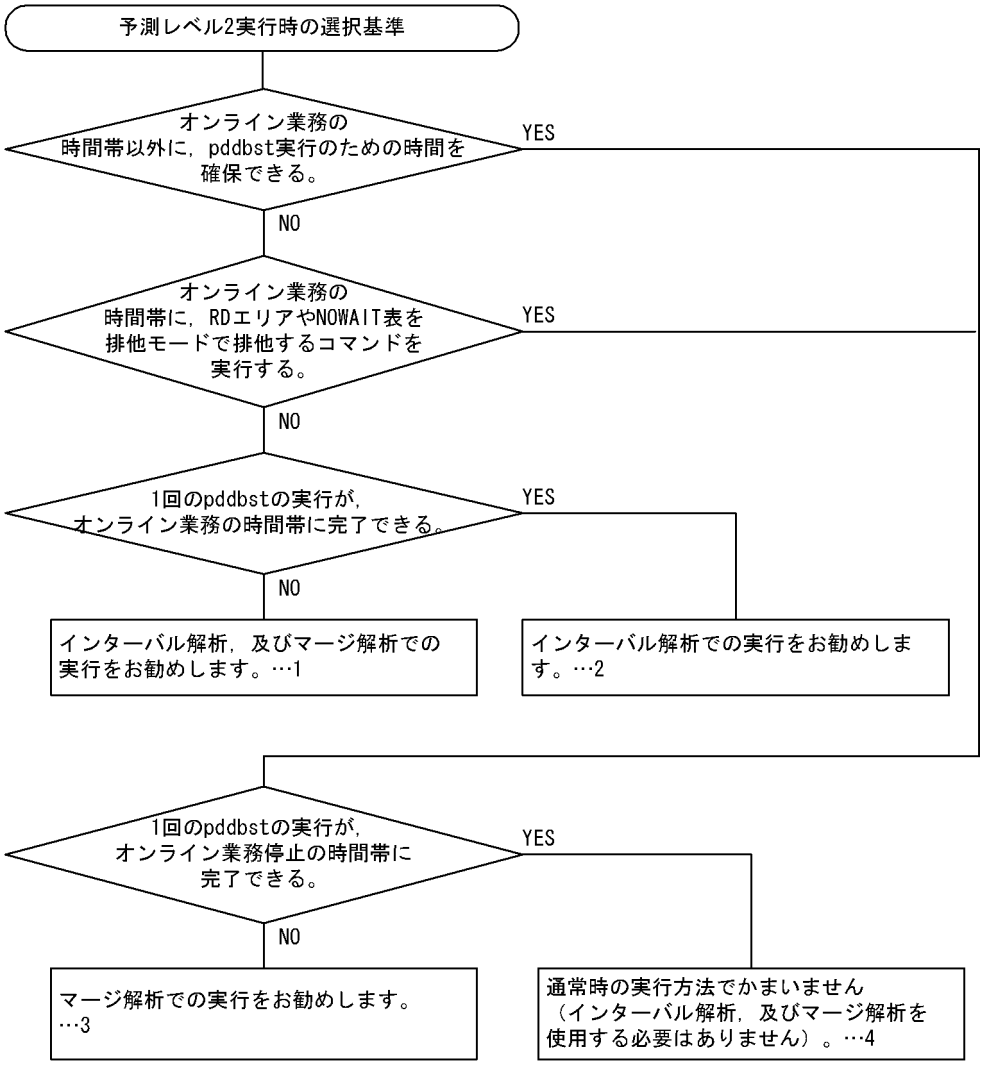
図 15–14　マージ解析の実行の概要



15.6.3　選択基準

どのように実行するかは，業務特性によって変わります。実行方法の選択基準を次の図に示します。

図 15–15　実行方法の選択基準



〔説明〕

1. インターバル解析，及びマージ解析を使用する場合のコマンド実行例を次に示します。

```
pddbst -r ALL -e 2 -w 100,100 -n 5
```

〔説明〕

オンライン業務への影響を少なくするために，100 セグメントのデータを解析したら，処理を 100 ミリ秒休止します。これを繰り返しながら実行します。さらに，5 日間に分けて実行することで，1 日当たりの実行時間を短縮する運用です。1 日に 1 回，上記コマンドを実行し，それを 5 日間行います。

2. インターバル解析を使用する場合のコマンド実行例を次に示します。

```
pddbst -r ALL -e 2 -w 100,100
```

〔説明〕

オンライン業務への影響を少なくするために，100 セグメントのデータを解析したら，処理を 100 ミリ秒休止します。これを繰り返しながら実行します。

3. マージ解析を使用する場合のコマンド実行例を次に示します。

```
pddbst -r ALL -e 2 -n 5
```

〔説明〕

pddbst の実行を 5 日間に分けることで，1 日当たりの実行時間を短縮する運用です。1 日に 1 回，上記コマンドを実行し，それを 5 日間行います。

4. 通常時のコマンド実行例（インターバル解析，及びマージ解析を使用しないコマンド実行例）を次に示します。

```
pddbst -r ALL -e 2
```

〔説明〕

通常時の実行方法で，1 回の pddbst の実行で処理が完了します。

15.7　規則及び注意事項

15.7.1　規則

1. pddbst は，HiRDB が稼働中のときだけ実行できます。
2. pddbst は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pddbst の最大同時実行数は，システム定義の pd_utl_exec_mode オペランドの指定値によって異なります。

   pd_utl_exec_mode=0 の場合：

   　最大同時実行数は 16 です。

   pd_utl_exec_mode=1 の場合：

   　最大同時実行数は pd_max_users オペランドの指定値です。
4. 解析対象となる RD エリア（データディクショナリ用 RD エリア，データディクショナリ LOB 用 RD エリア，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリア）は，次のどれかの状態でなければなりません。
   - 閉塞なしオープン状態
   - 参照可能閉塞オープン状態

   ユーザ用 RD エリア（RD エリアの論理的解析の場合は一時表用 RD エリアを除きます），及びユーザ LOB 用 RD エリアについては，「コマンド実行時の RD エリアの状態」を参照してください。

15.7.2　注意事項

1. pddbst のリターンコードを次に示します。

   0：正常終了

   4：警告終了（指定不正又は解析処理スキップ）

   8：異常終了
2. pddbst を実行する場合，排他制御の最小単位は表（意図共用モード：SR）であり，インデクスや行に対して排他は掛かりません。そのため，pddbst の実行中に表やインデクスの更新があった場合，更新トランザクションが決着しない状態で pddbst を実行するので，更新トランザクションがロールバックしたとき，実際のデータベースの状態と解析結果が異なることがあります。
3. pdrorg の再編成処理が異常終了した表を，論理的解析（-k logi 指定）又は表単位解析（-t 指定）で解析すると，Status には DATA_UNFINISH が表示されます。
4. 再編成時期予測機能を実行する場合，シングルサーバ定義，又はディクショナリサーバ定義に指定する pd_lck_pool_size オペランドには，デフォルト値未満の値は指定しないでください。次の計算式から求まる値以上を指定してください。

pd_lck_pool_size オペランドの値=↑y÷x↑（単位：キロバイト）

x：32 ビットモードの場合は 6，64 ビットモードの場合は 4

y：pddbst が消費する排他資源数

y=MAX（a，b，c）

a：定義した RD エリア数×30

b：定義した表数＋定義した抽象データ型数＋抽象データ型列の総数※＋ 61

c：定義したインデクス数＋ 124

注※

同じ抽象データ型でも，表や列が異なる場合はカウントします。

15.7.3　BOM を持つファイルの使用

pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pddbst の入力ファイルには，BOM を持つファイルを使用できることがあります。pddbst での BOM を持つファイルの使用可否を次の表に示します。なお，pddbst の入力ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pddbst で出力されるファイルには，BOM は付きません。

表 15–12　pddbst での BOM を持つファイルの使用可否（UTF-8 の場合）

オプション	入力ファイル	BOM を持つファイルの使用
-c	基準値定義ファイル	○
-v	制御文ファイル	○

（凡例）

○：使用できます。

15.7.4　コミットしていない削除データに排他を掛ける場合の注意事項

残存エントリが残ったままで pd_lock_uncommited_delete_data オペランドの指定を wait から nowait に変更して pddbst を実行すると，正しい状態解析ができません。pd_lock_uncommited_delete_data オペランドの指定を nowait に変更する場合，次のどちらかの方法で残存エントリがない状態にしてから変更する必要があります。

- ページコンパクション（-x オプション）指定ありの空きページ解放（pdreclaim）
- インデクスの再編成（-k ixor オプション指定ありの pdrorg）

15.8　使用例

pddbst の使用例を次に示します。

15.8.1　DB 状態解析機能

例

表単位の状態解析をします。解析するのは表ZAIKOです。
なお，表ZAIKOはpddefの使用例で定義してあるものとします。

[コマンド実行例]

```
pddbst -t ZAIKO
```

15.8.2　状態解析結果蓄積機能

例

RDエリア名「RD001」の状態解析結果蓄積機能を次の条件で実行します。
・予測レベル2
・マージ解析（3回）

[コマンド実行例]

```
pddbst -r RD001 -e 2 -n 3
```

注

上記のコマンドを 3 回に分けて実行します。

15.8.3　再編成時期予測機能

例

(2)で蓄積した解析情報を基に，再編成時期予測機能を予測レベル2で実行します。

[コマンド実行例]

```
pddbst -k pred -e 2
```

16 最適化情報収集ユティリティ（pdgetcst）

この章では，コストベースの最適化のための最適化情報を取得してディクショナリ表へ登録する最適化情報収集ユティリティ（pdgetcst）について説明します。

なお，IVS 対応 UTF-8 文字コードを使用する場合は，この章の記載を次のように読み替えてください。

utf-8 -> utf-8_ivs

16.1　概要

実行者　対象となる表に対して SELECT 権限を持っているユーザが実行できます

最適化情報収集ユティリティは，コストベースの最適化のための最適化情報を取得し，ディクショナリ表へ登録するユティリティです。また，ディクショナリ表へ登録した最適化情報は削除できます。最適化情報収集ユティリティの機能を次に示します。

- **検索による最適化情報の収集**
- **最適化情報パラメタファイルによる最適化情報の登録**
- **最適化情報パラメタファイルへの最適化情報の退避**

最適化情報収集ユティリティを実行する場合の条件：

最適化情報収集ユティリティは，次の条件をすべて満たす場合に実行してください。

- コストベース最適化モードが 2 である※1
- 更新系業務を行わない※2
- 結合検索などの複雑な検索が多い
- 本番環境と同一データのテスト環境がある※3

注※1

コストベース最適化モードが 2 でなくても，最適化情報収集レベルが lvl2 であれば適用できます。

注※2

更新系業務を行っていても，最適化情報収集レベルが lvl1 で，かつ表の行数が変化しない更新であれば適用できます。

注※3

本番環境と同一データのテスト環境がなくても，最適化情報パラメタファイルによる最適化情報の登録を行えば適用できます。

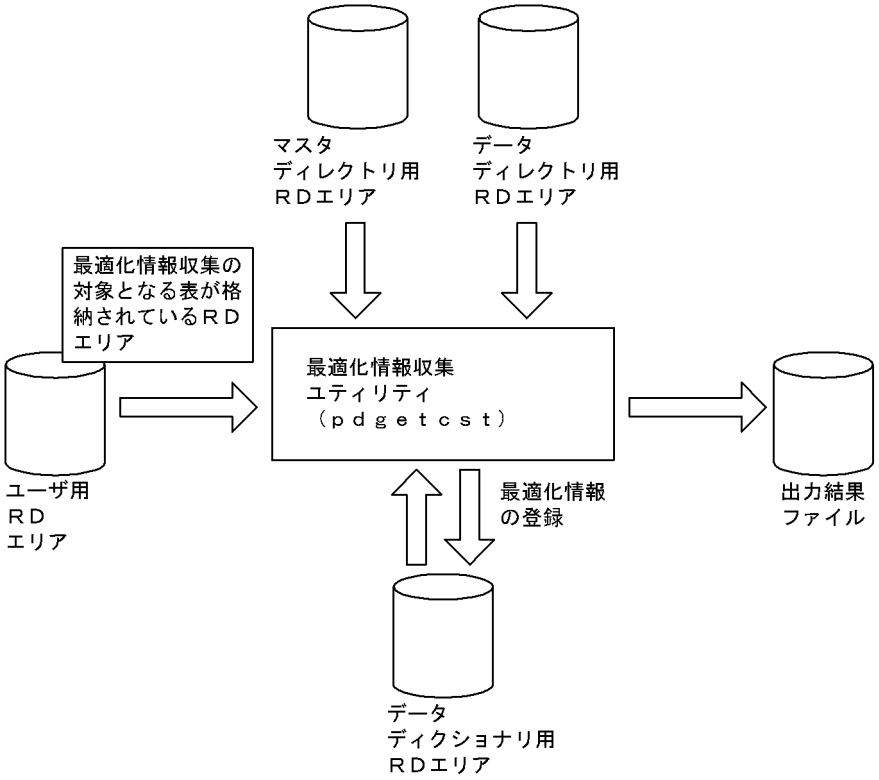
16.1.1　検索による最適化情報の収集

現在のデータベースの状態を考慮してアクセスパスを決定する最適化のことをコストベースの最適化といいます。このコストベースの最適化によって，HiRDB は最も効率的なアクセスパスを決定するので，ユーザは SQL の最適化を意識しないで UAP を作成できます。最適化情報については，マニュアル「HiRDB システム運用ガイド」の「最適化情報の運用」を参照してください。

最適化情報収集ユティリティは，実行条件を満たす場合に実行してください。実行条件については，「概要」を参照してください。

最適化情報収集ユティリティ（pdgetcst）の概要（検索による最適化情報の収集）を次の図に示します。

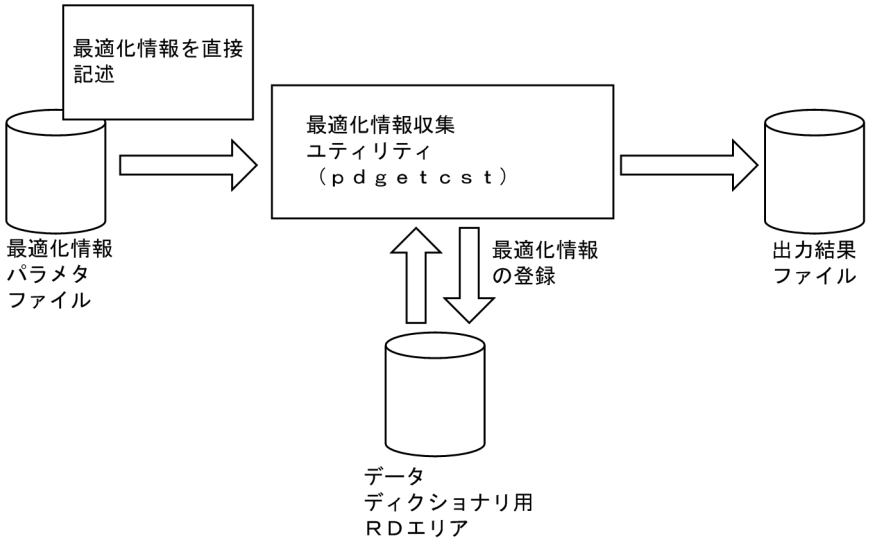
図 16–1　最適化情報収集ユティリティ（pdgetcst）の概要（検索による最適化情報の収集）



16.1.2　最適化情報パラメタファイルによる最適化情報の登録

現在のデータベースの状態から最適化情報を収集しないで，最適化情報パラメタファイルに任意に設定した最適化情報を，ディクショナリ表へ登録します。また，最適化情報の退避で保存しておいた最適化情報パラメタファイルを登録することもできます。最適化情報収集ユティリティ（pdgetcst）の概要（最適化情報パラメタファイルによる最適化情報の登録）を次の図に示します。

図 16–2　最適化情報収集ユティリティ（pdgetcst）の概要（最適化情報パラメタファイルによる最適化情報の登録）

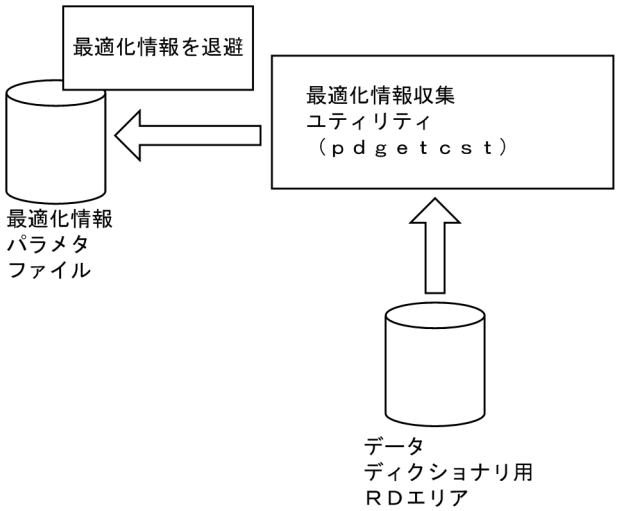


16.1.3　最適化情報パラメタファイルへの最適化情報の退避

ディクショナリ表へ登録された最適化情報を，最適化情報パラメタファイルへ退避します。ディクショナリ表に最適化情報が登録されていない場合は，退避できません。

最適化情報収集ユティリティ（pdgetcst）の概要（最適化情報パラメタファイルへの最適化情報の退避）を次の図に示します。

図 16–3　最適化情報収集ユティリティ（pdgetcst）の概要（最適化情報パラメタファイルへの最適化情報の退避）



最適化情報を出力したファイルは，最適化情報パラメタファイルとしてディクショナリ表へ登録できます。

16.1.4　ユティリティの最大同時実行数

最適化情報収集ユティリティ（pdgetcst）の最大同時実行数は，システム共通定義の pd_utl_exec_mode オペランドの指定値によって異なります。

pd_utl_exec_mode=0 の場合

　最大同時実行数は 16 です。

pd_utl_exec_mode=1 の場合

　最大同時実行数は pd_max_users オペランドの指定値÷ 2 です。

16.2　コマンドの形式

16.2.1　形式

検索による最適化情報の収集

```
pdgetcst 〔-u 認可識別子〕〔-a 認可識別子〕〔-p パスワード〕
            -t {表識別子 | ALL}
             〔-c {lvl1 | lvl2}〕
             〔-l 出力結果ファイル名〕
             〔-W 実行監視時間〕
```

最適化情報の削除

```
pdgetcst -d -t {表識別子 | ALL}
             〔-u 認可識別子〕〔-a 認可識別子〕〔-p パスワード〕
             〔-l 出力結果ファイル名〕
             〔-W 実行監視時間〕
```

最適化情報パラメタファイルによる最適化情報の登録

```
pdgetcst 〔-u 認可識別子〕〔-a 認可識別子〕〔-p パスワード〕
            -t 表識別子 -s 最適化情報パラメタファイル名
             〔-l 出力結果ファイル名〕
             〔-v 空白変換レベル〕
             〔-W 実行監視時間〕
```

最適化情報パラメタファイルへの最適化情報の退避

```
pdgetcst -e 最適化情報パラメタファイル名 -t 表識別子
             〔-u 認可識別子〕〔-a 認可識別子〕〔-p パスワード〕
             〔-W 実行監視時間〕
```

-c，-d，-s，又は-e オプションを指定したときの他オプションの指定可否を次の表に示します。なお，次の場合は KFPN00100-E メッセージを出力して，エラーとなります。

- 同じオプションを重複して指定した場合
- 同時に指定できないオプションの指定値を指定した場合

表 16–1　-c，-d，-s，又は-e オプションを指定したときの他オプションの指定可否

オプション		-c オプション		-d オプション	-s オプション	-e オプション
		lvl1	lvl2			
-u 認可識別子		○	○	○	○	○
-a 認可識別子		○	○	○	○	○
-p パスワード		○	○	○	○	○
-t	表識別子	◎	◎	◎	◎	◎
	ALL		×		×	×
-d		×	×	×	×	×
-c	lvl1	×	×	×	×	×
	lvl2	×	×	×	×	×
-s 最適化パラメタファイル名		×	×	×	×	×
-l 出力結果ファイル名		○	○	○	○	–
-v 空白変換レベル		×	×	×	○	×
-e 最適化情報パラメタファイル名		×	×	×	×	×
-W 実行監視時間		○	○	○	○	○

（凡例）

◎：必ず指定してください。

○：任意に指定できます。

×：指定できません。

–：指定できますが，無視されます。

-c，-d，-s，及び-e オプションをすべて省略した場合は，-c オプションの lvl2 を指定したとみなして，コマンドを実行します。

16.2.2　オプション

- **-u 認可識別子**

HiRDB に接続する際に使用する認可識別子を指定します。省略した場合は，環境変数 PDUSER に設定された値が仮定されます。PDUSER に値が設定されていない場合，ユティリティを実行するユーザの OS 上のユーザ ID に対応するユーザ名が仮定されます。

認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

- **-a 認可識別子**

最適化情報の取得対象となる表の所有者の認可識別子を指定します。

省略した場合，-u オプションで与えられた認可識別子が仮定されます。

認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。

- **-p パスワード**

-u オプションで指定した認可識別子に対応するパスワードを指定します。

-u オプションが指定されていて，-p オプションを省略した場合，ユーザに対して入力を要求し，入力された値を用います。パスワードが不要な場合は，空送信を押してください。

-u オプションが省略されていて，-p オプションも省略した場合，環境変数 PDUSER に設定された値が仮定されます。PDUSER に値が設定されていない場合，ユーザに対し入力を要求し，入力された値を用います。パスワードが不要な場合は，空送信を押してください。

パスワードを応答できない環境で実行するとき，パスワードの入力が要求される条件では実行しないでください。パスワードの入力が要求される条件とは次のどちらかです。

- -u オプション指定あり，かつ-p オプション指定なし
- -u オプション指定なし，-p オプション指定なし，かつ環境変数 PDUSER にパスワードの設定なし

パスワードを「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」'で囲まない場合は，すべて英大文字として扱われます。

- **-t {表識別子｜ALL}**

表識別子

最適化情報の取得対象となる表の識別子を指定します。

ディクショナリ表，ビュー表，及び一時表は指定できません。指定して実行すると，pdgetcst は KFPN00202-E メッセージを出力してエラー終了します。

表識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。「¥""」と「"¥"」で囲まない場合は，すべて英大文字として扱われます。また，表識別子に空白が含まれる場合は，「¥""」と「"¥"」で囲んでください。

ALL

スキーマ内のすべての表を対象とする場合に指定します。ただし，ディクショナリ表，ビュー表，及び一時表の場合，pdgetcst は処理をスキップします。

- **-d**

-t オプションで指定した表に関する最適化情報，及び表中に定義したインデクスに関する最適化情報を削除する場合に指定します。

- **-c {lvl1 | lvl2}**

最適化情報収集レベルを指定します。

最適化情報収集レベルについては，「最適化情報収集レベル」を参照してください。

lvl1：表の行数，及び統計情報キャッシュサイズを取得します。

lvl2：表のすべての最適化情報を取得します。

- **-s 最適化情報パラメタファイル名** ～＜パス名＞

最適化情報を設定してある最適化情報パラメタファイルの名称を，絶対パス名で指定します。このオプションを指定した場合，現在のデータベース状態からの最適化情報は収集しないで，最適化情報パラメタファイルの最適化情報をディクショナリ表へ登録します。このオプションを指定した場合，-d オプションは指定できません。

- **-l 出力結果ファイル名** ～＜パス名＞

最適化情報収集ユティリティの実行結果を出力するファイルの名称を絶対パス名で指定します。省略した場合，実行結果は出力されません。

-t ALL 指定の pdgetcst がエラーとなった場合，このオプションを指定しているときは，エラーとなった表を特定できます。

- **-v 空白変換レベル**

列の最小値，列の最大値，及び区間内の列の最大値に指定した値の中の空白について，空白変換するかどうかを指定します。

このオプションを省略した場合，システム共通定義の pd_space_level オペランドの指定値に従って空白変換します。

空白変換レベルには，次の 0，1，又は 3 のどれかの値を指定します。

0：

空白変換をしません。

1，又は 3：

空白変換をします。1 と 3 のどちらを指定しても意味は同じです。

空白変換をする場合，登録する表の列が各国文字列型，又は混在文字列型のとき，対応する最適化パラメタファイルの値の中の空白を，次のように変換します。

- 各国文字列型の列の場合

連続する半角空白 2 バイトを，全角空白 1 文字に変換します。ただし，先頭から 2 バイト単位で変換します。

- 混在文字列型の列の場合

  全角空白 1 文字を，半角空白 2 バイトに変換します。

  文字コードが utf-8 の場合，全角空白 1 文字（3 バイト）を半角空白 2 バイトに変換します。MCHAR の場合は，データ長が短くなった分，定義長まで後ろから半角空白が埋められます。MVARCHAR の場合は，データ長が短くなったままとなります。

- **-e 最適化情報パラメタファイル名**　～＜パス名＞

最適化情報を退避する場合，最適化情報を出力するファイル名を絶対パスで指定します。

＜規則＞

1. ディクショナリ表に最適化情報が登録されていない場合は，最適化情報を退避できません。
2. -e オプションで指定した名称のファイルが既に存在する場合，KFPN00015-I メッセージを出力して，既存のファイルに最適化情報を上書きします。
3. ディクショナリ表に最適化情報が登録されていない状態で，かつ-e オプションで指定した名称のファイルが既に存在する場合は，既存のファイルを削除し，-e オプションで指定した名称のファイルも作成しません。この場合にも，KFPN00015-I メッセージを出力します。
4. -e オプションで退避した最適化情報パラメタファイルは，表定義が同じ場合に限り，ディクショナリ表へ再登録できます。表定義が異なる場合は，再登録できないことがあります。詳細については，マニュアル「HiRDB システム運用ガイド」の「表及びインデクスの定義変更をした場合の最適化情報パラメタファイルの使用可否」を参照してください。

- **-W 実行監視時間**　～＜符号なし整数＞((0～3600))

最適化情報収集ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。

このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

オプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

16.3　最適化情報

16.3.1　取得の有無

インデクス構成列及びインデクスに関する最適化情報の場合，条件によっては最適化情報を取得しないことがあります。最適化情報を取得しないと，最適化情報パラメタファイルで最適化情報を登録できません。最適化情報を取得しない条件を次に示します。

インデクス構成列に関する最適化情報を取得しない条件

- 列構成がインデクス構成列※で，データ型が抽象データ型
- 列構成がインデクス構成列※で，データ型が抽象データ型以外で，繰返し列がある
- 列構成がインデクス構成列ではない

注※

複数列インデクスのときは第 1 構成列になります。

インデクスに関する最適化情報を取得しない条件

- B-tree インデクスで，インデクスの第 1 構成列が繰返し列
- 部分構造インデクス※
- プラグインインデクス※

注※

最適化情報パラメタファイルで最適化情報を登録しようとすると，KFPN00201-E メッセージを出力してエラー終了します。

16.3.2　取得する最適化情報

最適化情報収集ユティリティが取得する最適化情報，又は最適化情報パラメタファイルによって登録する最適化情報を次の表に示します。

表 16–2　最適化情報収集ユティリティが取得する最適化情報，又は最適化情報パラメタファイルによって登録する最適化情報

項番	取得項目	分類	内容	最適化の精度が向上する SQL	登録した最適化情報がデータベースの状態と一致しなくなる処理	項目名※1
1	行数	T	対象表のすべての行数	結合を含む SQL	対象表の行数が変わる更新系 SQL	NROWS
2	データページ数	T	対象表の行データ格納ページ数の合計	検索条件にインデクス構成列を含む SQL	更新系 SQL	NPAGES

項番	取得項目	分類	内容	最適化の精度が向上する SQL	登録した最適化情報がデータベースの状態と一致しなくなる処理	項目名※1
3	インデクスキーのユニーク値の数	I	ナル値。ただし，最適化情報パラメタファイルを指定して登録する場合は，指定値を登録します。			NENTRY
4	インデクスページ数	I	インデクス格納使用ページ数の合計			NIPAGES
5	インデクス段数	I	各シングルサーバ内又は各バックエンドサーバ内のインデクス段数の最大値			NLEVEL
6	シーケンシャル度	I	インデクスキー値の順番で行データが格納されている割合			SEQ_RATIO
7	ナル値数	C	列中のナル値の数	検索条件に単一列インデクスの構成列，又は複数列インデクスの第 1 構成列を含む SQL		NNULLS
8	最大重複列値数	C	一つの列値に対する重複数の最大値			N_MAX_DUP_KEY
9	最小重複列値数	C	一つの列値に対する重複数の最小値			N_MIN_DUP_KEY
10	列値の分布情報	C	列値の分布情報※2			MAX_VALUE，MIN_VALUE，SECTION_COUNT，SEC_TOTAL_COUNT，SEC_UNIQUE，SEC_MAX_VALUE
11	列ユニーク値数	C	列のユニーク値数			NUNIQUE
12	統計情報キャッシュサイズ	T	HiRDB システムが利用する作業用の領域	−	−	−

（凡例）

T：

表に関する情報であることを示します。

I：

インデクスに関する情報であることを示します。

C：

列に関する情報であることを示します。

- 検索による最適化情報の収集の場合

  インデクス構成列に関する情報です。インデクスが定義されている場合に収集されます。ただし，複数列インデクスの場合は，第 1 構成列の情報だけを収集します。

- 最適化情報パラメタファイルによる最適化情報の登録の場合

  最適化情報パラメタファイルで指定した列に関する情報です。

–：

該当しません。

注※1

最適化情報パラメタファイルに記述する項目名のことです。最適化情報パラメタファイルに記述する項目名については，「最適化情報パラメタファイルの項目」を参照してください。

注※2

列が文字データ，各国文字データ，及び混在文字データの場合は，引用符（''）で囲んで表示します。ただし，データに制御文字が含まれている場合，及び列に文字集合が指定されている場合は，引用符（''）を表示しません。また，バイナリデータの場合は 16 進表記で表示します。

列のデータ型が CHAR，NCHAR，及び MCHAR の場合は先頭 16 バイトを，VARCHAR，NVARCHAR，及び MVARCHAR の場合は先頭 14 バイトを，BINARY の場合は先頭 12 バイトを表示します。ただし，マルチバイト文字が MCHAR で 16 バイト以上，MVARCHAR で 14 バイト以上の場合は，そのマルチバイト文字を表示します。

なお，16 進表記は，x'****'，又は X'****'の形式とします。****の部分は，16 進文字列（0〜9，a〜f，及び A〜F）を 2 文字単位に記述します。

インデクス構成列のキー値の分布情報として取得又は登録する情報を次の表に示します。

表 16–3　インデクス構成列のキー値の分布情報として取得又は登録する情報

項番	取得項目	内容
1	ナル値数	列中のナル値の数
2	表の行数	対象表のすべての行数
3	列のデータ長	列のデータ長
4	列のデータ型	列のデータ型
5	区間数	• （表の行数–列のデータがナル値の行数）の数を，最大 100 区間に分割した場合の区間数 • 行数が 100 未満の場合は行数を設定 • 項番 8〜10 の情報を区間数分取得
6	列値の最大値	列の値の最大値※
7	列値の最小値	列の値の最小値※
8	区間の累積度数	第 1 区間から該当する区間までの要素数の合計

項番	取得項目	内容
9	区間内のユニーク値の数	該当する区間内の列値のユニーク数
10	区間内の列値の最大値	該当する区間内の列値の最大値※

注※

- 表の格納データが 0 件，又はすべてナル値の場合
  次の表の初期値が取得，又は登録されます。
- 表の格納データにナル値以外のデータがある場合
  次の表の格納内容が取得，又は登録されます。

項番	列のデータ型	初期値	格納内容
1	INT〔EGER〕	0	表データ※，又は初期値
2	SMALLINT	0	
3	〔LARGE〕 DEC〔IMAL〕	+0	
4	FLOAT，又は DOUBLE PRECISION	0	
5	SMALLFLT，又は REAL	0	
6	CHAR〔ACTER〕	• 定義長が 16 バイト以下の場合 0 が定義長バイト分取得，又は登録されます。 • 定義長が 17 バイト以上の場合 0 が 16 バイト分取得，又は登録されます。	• 定義長が 16 バイト以下の場合 表データ※，又は初期値 • 定義長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値
7	VARCHAR	長さ 0 の文字列	• 文字列長，又はデータ長が 16 バイト以下の場合 表データ※，又は初期値 • 文字列長，又はデータ長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値
8	NCHAR	• 定義長が 8 バイト以下の場合 0 が定義長×2 バイト分取得，又は登録されます。 • 定義長が 9 バイト以上の場合 0 が 16 バイト分取得，又は登録されます。	• 定義長が 16 バイト以下の場合 表データ※，又は初期値 • 定義長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値
9	NVARCHAR	長さ 0 の文字列	• 文字列長，又はデータ長が 16 バイト以下の場合 表データ※，又は初期値

項番	列のデータ型	初期値	格納内容
			• 文字列長，又はデータ長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値
10	MCHAR	• 定義長が 16 バイト以下の場合 0 が定義長バイト分取得，又は登録されます。 • 定義長が 17 バイト以上の場合 0 が 16 バイト分取得，又は登録されます。	• 定義長が 16 バイト以下の場合 表データ※，又は初期値 • 定義長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値
11	MVARCHAR	長さ 0 の文字列	• 文字列長，又はデータ長が 16 バイト以下の場合 表データ※，又は初期値 • 文字列長，又はデータ長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値
12	DATE	0000-00-00	表データ※，又は初期値
13	TIME	00:00:00	
14	INTERVAL YEAR TO DAY	+00000000.	
15	INTERVAL HOUR TO SECOND	+000000.	
16	TIMESTAMP	0000-00-00 00:00:00.0....0	
17	BINARY	長さ 0 のデータ	• 文字列長，又はデータ長が 16 バイト以下の場合 表データ※，又は初期値 • 文字列長，又はデータ長が 17 バイト以上の場合 表データ※の先頭 16 バイト，又は初期値

注※

文字集合が指定された CHAR 型，又は VARCHAR 型の列の場合，指定された文字集合のデータで格納されます。

16.3.3 最適化情報収集レベル

検索によって最適化情報を収集する場合，収集する情報のレベルを指定できます。これを最適化情報収集レベルといいます。

各最適化情報レベルの長所，短所を次の表に示します。

表 16–4　各最適化情報レベルの長所，短所

最適化情報収集レベル	長所	短所
lvl1	• 表の行数を最適化に反映できるため，結合順序などでデータ特性を反映でき，最適化の精度は比較的高いです。 • pdgetcst の実行時間が比較的短くなります。	• データの更新を意識した運用となります。 • アクセスパスが，期待した結果にならなかった場合，表の行数のコスト情報を考慮する必要があるため，アクセスパスの予想が難しくなります。
lvl2	表の行数，最大値，最小値，分布情報などを最適化に反映できるため，結合順序，インデクスの選択などでデータ特性を反映できて，最適化の精度は高いです。	• データの更新を意識した運用となります。 • pdgetcst の実行時間が長くなります。 • アクセスパスが，期待した結果にならなかった場合，取得したコスト情報をすべて考慮する必要があるため，アクセスパスの予想が難しくなります。 • 列情報は，インデクスの第 1 構成列の情報だけを取得するので，複数列インデクスを定義している場合，最適化の精度が悪くなります。

アクセスする表の特性を調べて，最適化情報収集レベルを決めてください。ただし，最適化情報を取得しない（pdgetcst を実行しない）方がよい場合もあります。

最適化情報を取得しない場合，更新系 SQL でのデータの変更を意識しなくても良くなりますが，最適化にデータ特性を反映できないので最適化の精度は良くないです。

最適化情報収集レベルの推奨値を次の表に示します。

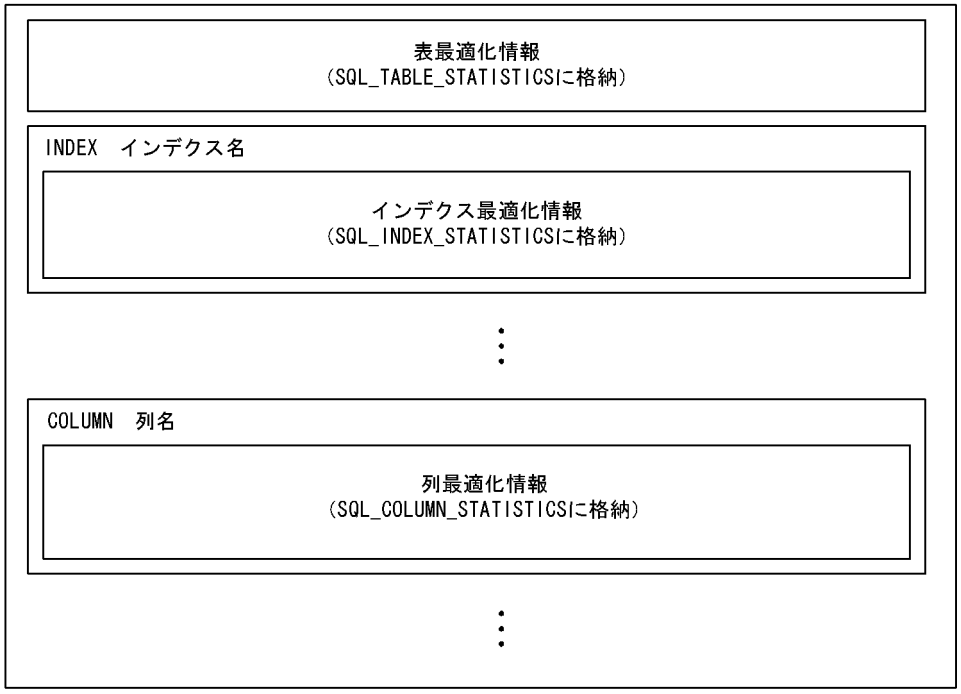
表 16–5　最適化情報収集レベルの推奨値

項番	アクセスする表の特性	推奨する最適化情報収集レベル
1	次の条件をすべて満たす場合 • 更新系業務を行わない • 検索条件にインデクス構成列を含む SQL を実行する	lvl2
2	次の条件をすべて満たす場合 • 項番 1 の条件に該当しない • コストベース最適化モードが 2 である • 格納行数が変わる更新系業務を行わない • 次のどちらかの条件に該当する ・ハッシュジョインを行う ・副問合せのハッシュ実行を行う ・3 表以上の結合検索を行う	lvl1
3	上記以外	最適化情報を収集しない（pdgetcst を実行しない）ことをお勧めします。

16.4　最適化情報パラメタファイル

最適化情報を退避するときに出力される最適化情報パラメタファイルの項目について説明します。なお，最適化情報を退避していない場合に最適化情報の登録をするときには，あらかじめ最適化情報を記述した最適化情報パラメタファイルを用意しておく必要があります。最適化情報パラメタファイルの形式を次の図に示します。

図 16–4　最適化情報パラメタファイルの形式



16.4.1　最適化情報パラメタファイルの項目

最適化情報パラメタファイルに記述する項目の一覧と最適化情報パラメタファイルに記述する項目の内容を次の表に示します。

表 16–6　最適化情報パラメタファイルに記述する項目の一覧

分類	項目名	内容
表最適化情報	NROWS	表の行数
	NPAGES	表のデータページ数
インデクス最適化情報	INDEX	インデクス名
	NIPAGES	インデクスページ数
	NLEVEL	インデクスの段数
	SEQ_RATIO	インデクスのシーケンシャル度

分類	項目名	内容
	NENTRY	インデクスのキー値数
列最適化情報	COLUMN	列名
	NUNIQUE	列のユニーク値数
	NNULLS	列のナル値数
	N_MAX_DUP_KEY	列の最大重複列値の数
	N_MIN_DUP_KEY	列の最小重複列値の数
	MAX_VALUE	列値の最大値
	MIN_VALUE	列値の最小値
区間分布情報	SECTION_COUNT	区間分布情報の区間数
	SECTION_NO	区間番号
	SEC_TOTAL_COUNT	区間要素数の累積度数
	SEC_UNIQUE	区間内ユニーク値数
	SEC_MAX_VALUE	区間内最大値

表 16–7　最適化情報パラメタファイルに記述する項目の内容

項目名	内容
NROWS	表の行数を指定します。 表の行数は，次の SQL 文で求められます。 SELECT COUNT(*) FROM 表名
NPAGES	表の使用中データページ数を指定します。 データベース状態解析ユティリティを実行するか，又はマニュアル「HiRDB システム導入・設計ガイド」を参照してください。
INDEX	インデクス名を指定します。 指定するインデクスは，-t オプションで指定した表に定義されている必要があります。 なお，指定できるインデクスは B-tree インデクスだけであり，プラグインインデクスは指定できません。
NIPAGES	インデクスの使用中インデクスページ数を指定します。 データベース状態解析ユティリティを実行するか，又はマニュアル「HiRDB システム導入・設計ガイド」を参照してください。
NLEVEL	インデクスの段数を指定します。 マニュアル「HiRDB システム導入・設計ガイド」を参照してください。また，クラスタキーの場合は，データベース状態解析ユティリティで求められます。
SEQ_RATIO	インデクスのシーケンシャル度を指定します。 0〜100 の値を指定します。インデクスのキー順にデータが格納されている場合は 100，ランダムにデータが格納されている場合は 0 を指定してください。この値を大きくすると，よりインデクスを使用することになります。

項目名	内容
NENTRY	インデクスのキー値数を指定します。 • ユニークインデクスの場合は，表の行数を指定してください。 • ユニークインデクスでない場合は，重複排除したインデクスのキー値数を指定してください。 インデクスのキー値数は，次の SQL 文で求められます。 ＜単一列インデクスの場合＞ SELECT COUNT(DISTINCT インデクス構成列名※1) FROM 表名 ＜複数列インデクスの場合＞ WITH W1(WC1,WC2,…) AS (SELECT DISTINCT インデクス構成列名 1※1,インデクス構成列名 2※1,… FROM 表名) SELECT COUNT(*) FROM W1
COLUMN	列名を指定します。 指定する列は，-t オプションで指定した表になければなりません。ただし，LOB 列，抽象データ型列は指定できません。
NUNIQUE	列のユニーク値数を指定します。 列のユニーク値数は，次の SQL 文で求められます。 SELECT COUNT(DISTINCT 列名※2) FROM 表名
NNULLS	列のナル値数を指定します。 列のナル値数は，次の SQL 文で求められます。 • 列の定義長※3が 32,000 バイト以下の場合 SELECT COUNT(*) FROM 表名 WHERE 列名 IS NULL • 列の定義長※3が 32,001 バイト以上の場合 WITH W1(WSUBCLM) AS (SELECT SUBSTR(列名,1,255※4)) FROM 表名) SELECT COUNT(*) FROM W1 WHERE WSUBCLM IS NULL
N_MAX_DUP_KEY	最も重複が多い列値の重複数を指定します。 最も重複が多い列値の重複数は，次の SQL 文で求められます。 • 列名で指定する列の定義長※3が 255 バイト以下の場合 WITH W1(WCOUNT) AS (SELECT COUNT(列名) FROM 表名 WHERE 列名 IS NOT NULL GROUP BY 列名) SELECT MAX(WCOUNT) FROM W1 • 列名で指定する列の定義長※3が 256 バイト以上，32,000 バイト以下の場合 SELECT COUNT(SUBSTR(列名,1,255※4)) FROM 表名 WHERE 列名 IS NOT NULL GROUP BY SUBSTR(列名,1,255※4) ORDER BY 1 DESC LIMIT 1 • 列名で指定する列の定義長※3が 32,001 バイト以上の場合 WITH W1(WSUBCLM) AS (SELECT SUBSTR(列名,1,255※4)) FROM 表名) SELECT COUNT(WSUBCLM) FROM W1 WHERE WSUBCLM IS NOT NULL GROUP BY WSUBCLM ORDER BY 1 DESC LIMIT 1
N_MIN_DUP_KEY	最も重複が少ない列値の重複数を指定します。 最も重複が少ない列値の重複数は，次の SQL 文で求められます。

項目名	内容
	• 列名で指定する列の定義長※2が 255 バイト以下の場合 WITH W1(WCOUNT) AS (SELECT COUNT(列名) FROM 表名 WHERE 列名 IS NOT NULL GROUP BY 列名) SELECT MIN(WCOUNT) FROM W1 • 列名で指定する列の定義長※3が 256 バイト以上，32,000 バイト以下の場合 SELECT COUNT(SUBSTR(列名,1,255※4)) FROM 表名 WHERE 列名 IS NOT NULL GROUP BY SUBSTR(列名,1,255※4) ORDER BY 1 ASC LIMIT 1 • 列名で指定する列の定義長※3が 32,001 バイト以上の場合 WITH W1(WSUBCLM) AS (SELECT SUBSTR(列名,1,255※4)) FROM 表名) SELECT COUNT(WSUBCLM) FROM W1 WHERE WSUBCLM IS NOT NULL GROUP BY WSUBCLM ORDER BY 1 ASC LIMIT 1
MAX_VALUE※5	列値の最大値を指定します。 列値の最大値は，次の SQL 文で求められます。 SELECT MAX(列名※6) FROM 表名
MIN_VALUE※5	列値の最小値を指定します。 列値の最小値は，次の SQL 文で求められます。 SELECT MIN(列名※6) FROM 表名
SECTION_COUNT	区間分布情報の区間数を指定します。 列値の種類数が 100 以下の場合は，列値の種類数を設定します。列値の種類数が 100 より多い場合は，100 を設定します。 なお，この指定は区間分布情報の指定より前に記述しなければなりません。また，この指定値は，区間分布情報の指定数と同じ値にしてください。
SECTION_NO	該当する区間の区間番号を 1〜区関数の範囲内で指定します。
SEC_TOTAL_COUNT	区間要素数の累積度数を指定します。 区間要素数の累積度数は，次の計算式で求められます。 （列値の種類数÷100）×該当する区間番号 なお，列値の種類数÷100 が割り切れない場合は，最終区間の区間分布情報の定義時に，ヒストグラム累積度数に列値の種類数を設定します。
SEC_UNIQUE	該当する区間内の列値のユニーク値数を指定します。
SEC_MAX_VALUE	該当する区間内の列値の最大値を指定します。

注※1

インデクス構成列名に指定する列の定義長が 256 バイト以上の場合は，「インデクス構成列名」を「SUBSTR（インデクス構成列名,1,255※4）」に置き換えます。

注※2

列名で指定する列の定義長が 256 バイト以上の場合は，「列名」を「SUBSTR（列名,1,255※4）」に置き換えます。

注※3

定義長の単位はバイトです。各国文字列データ型の場合は，定義長×2 となります。

注※4

列のデータ型が文字データ型，及び BINARY 型の場合は 255 となります。各国文字列データ型，及び混在文字列データ型の場合は 127 となります。

注※5

MAX_VALUE は，ディクショナリ表 SQL_COLUMN_STATISTICS の RANGE_VALUES 列の 32 バイト目に格納されます。

MIN_VALUE は，ディクショナリ表 SQL_COLUMN_STATISTICS の RANGE_VALUES 列の 48 バイト目に格納されます。ただし，列名で指定する列のデータ型が DECIMAL 型，又は NUMERIC 型で，精度が 32 けた以上の場合は，列の定義長※3 に 32 バイトを足した位置に格納されます。

注※6

列の定義長が 256 バイト以上の場合は，次のように置き換えてください。

- 列のデータ型が CHAR，及び BINARY のときは，「列名」を「SUBSTR（列名,1,255）」に置き換えます。
- 列のデータ型が NCHAR，及び MCHAR のときは，「列名」を「SUBSTR（列名,1,127）」に置き換えます。
- 列のデータ型が VARCHAR，NVARCHAR，及び MVARCHAR のときは，「列名」を次の指定に置き換えます。

```
「CASE LENGTH(列名) WHEN  0 THEN  ‘’
  WHEN  1 THEN SUBSTR(列名,1,1)  WHEN  2 THEN SUBSTR(列名,1,2)
  WHEN  3 THEN SUBSTR(列名,1,3)  WHEN  4 THEN SUBSTR(列名,1,4)
  WHEN  5 THEN SUBSTR(列名,1,5)  WHEN  6 THEN SUBSTR(列名,1,6)
  WHEN  7 THEN SUBSTR(列名,1,7)  WHEN  8 THEN SUBSTR(列名,1,8)
  WHEN  9 THEN SUBSTR(列名,1,9)  WHEN 10 THEN SUBSTR(列名,1,10)
  WHEN 11 THEN SUBSTR(列名,1,11)  WHEN 12 THEN SUBSTR(列名,1,12)
  WHEN 13 THEN SUBSTR(列名,1,13) ELSE SUBSTR(列名,1,14) END 」
```

16.4.2　最適化情報パラメタファイルの項目の指定方法

最適化情報パラメタファイルの項目の指定方法を次の表に示します。

- 表「最適化情報パラメタファイルの項目の指定方法」
- 表「MAX_VALUE 及び MIN_VALUE の指定方法」
- 表「最適化パラメタファイルの項目省略時，及び指定値省略時の設定値」

表 16–8　最適化情報パラメタファイルの項目の指定方法

項目名	指定方法	指定最小値	指定最大値	単位
NROWS	正の整数を指定します。また，指数も指定できます。	0	HiRDB での FLOAT 型の最大値	行
NPAGES		0		ページ
INDEX	インデクス定義時の規則に従います。			
NIPAGES	正の整数を指定します。また，指数も指定できます。	1	HiRDB での FLOAT 型の最大値	ページ
NLEVEL	正の整数を指定します。	1	HiRDB での SMALLINT 型の最大値	段
SEQ_RATIO		0	100	%
NENTRY	正の整数を指定します。また，指数も指定できます。	0	HiRDB での FLOAT 型の最大値	個
COLUMN	表定義時の規則に従います。			
NUNIQUE	正の整数を指定します。また，指数も指定できます。	0	HiRDB での FLOAT 型の最大値	個
NNULLS		0		個
N_MAX_DUP_KEY		0		個
N_MIN_DUP_KEY		0		個
MAX_VALUE	−			
MIN_VALUE				
SECTION_COUNT	正の整数を指定します。また，指数も指定できます。	0	100	区間
SECTION_NO		1	区間数	−
SEC_TOTAL_COUNT		1	HiRDB での SMALLFLT 型の最大値	行
SEC_UNIQUE		1	HiRDB での SMALLFLT 型の最大値	個
SEC_MAX_VALUE	−			

（凡例）−：該当しません。

表 16–9　MAX_VALUE 及び MIN_VALUE の指定方法

COLUMN で指定した列のデータ型	指定方法	最小値，最大値
INT〔EGER〕	数値を指定します。	HiRDB での各データ型の規則に従います。指定できるけた数の最大値は，HiRDB での数定数の使用上の制限に従います。詳細は，マニュアル「HiRDB SQL リファレンス」の「データ型」を参照してください。
SMALLINT		
[LARGE]DEC[IMAL]		
FLOAT 又は DOUBLE PRECISION	数値を指定します。また，指数も指定できます。	

COLUMN で指定した列のデータ型	指定方法	最小値，最大値
SMALLFLT 又は REAL		
CHAR VARCHAR NCHAR NVARCHAR MCHAR MVARCHAR	文字又は文字列を指定します。 CHAR，VARCHAR，MCHAR，及び MVARCHAR で，¥0 や改行などの制御文字が含まれる場合や，列に文字集合が指定されている場合は，文字列を 16 進表記※2 で指定します。 次の指定をする場合は，引用符（'）で囲んでください。 • 空白を含む場合 • 長さ 0 の文字列の場合 • 「x'」，又は「X'」で始まり「'」で終わる文字列の場合	項目の名称を含めて，最適化情報パラメタファイルに 1 行以内で記述できます。ただし，最適化情報としては先頭 16 バイトを登録するため，17 バイト目以降の文字列は切り捨てられます（引用符は含まれません）。 また，データ型が MCHAR 又は MVARCHAR の場合，指定した文字列の 16，17 バイト目が 2 バイトコードとなっているときには，17 バイト以上の文字列を指定してください。
DATE	YYYY-MM-DD で指定します。	HiRDB での各データ型の規則に従います。詳細は，マニュアル「HiRDB SQL リファレンス」の「データ型」を参照してください。
TIME	hh:mm:ss で指定します。	
INTERVAL YEAR TO DAY	(±)YYYYMMDD.で指定します。	
INTERVAL HOUR TO SECOND	(±)hhmmss.で指定します。	
TIMESTAMP	"YYYY-MM-DD△hh:mm:ss〔.nnnnnn〕" △：半角空白 小数部〔.nnnnnn〕は，定義に合わせて 0，2，4，又は 6 けたの値を指定します。※1 また，指定値は引用符で囲んでください。	
BINARY	バイナリデータを 16 進表記※2 で指定します。長さ 0 のデータを指定する場合は，X' 'と指定します。	最適化情報としては先頭の 12 バイト分のデータを登録します。13 バイト以降は指定しても切り捨てられます。

注※1

TIMESTAMP 型の小数秒部のけた数を指定した場合の格納方法を次に示します。

定義けた数	入力データの小数秒部のけた数				
	0	2	4	6	0，2，4，及び 6 以外
0	そのまま格納します。	定義けた数以降を切り捨てます。			エラー（KFPL31002-E）
2	定義けた数まで 0 を埋めます。	そのまま格納します。	定義けた数以降を切り捨てます。		

定義けた数	入力データの小数秒部のけた数				
	0	2	4	6	0，2，4，及び 6 以外
4	定義けた数まで 0 を埋めます。		そのまま格納します。	定義けた数以降を切り捨てます。	
6	定義けた数まで 0 を埋めます。			そのまま格納します。	

注※2

16 進表記は，x'****'，又は X'****'の形式とします。****の部分は，16 進文字列（0〜9，a〜f，及び A〜F）を 2 文字単位に記述します。

表 16–10　最適化パラメタファイルの項目省略時，及び指定値省略時の設定値

項目名		指定値	格納先のディクショナリ表名	格納先の列名	設定値	格納先のディクショナリ表の設定対象行
NROWS※1	あり	あり	SQL_TABLE_STATISTICS	N_ROW	NROWS の指定値	-t オプションで指定した表
		なし			ナル値	
	なし	–			更新前の設定値	
NPAGES	あり	あり		N_PAGE	NPAGES の指定値	
		なし			ナル値	
	なし	–			更新前の設定値	
NIPAGES	あり	あり	SQL_INDEX_STATISTICS	N_IXPG	NIPAGES の指定値	INDEX で指定したインデクス
		なし			ナル値	
	なし	–			更新前の設定値	
NLEVEL	あり	あり		N_LEVEL	NLEVEL の指定値	
		なし			ナル値	
	なし	–			更新前の設定値	
SEQ_RATIO	あり	あり		SEQ_RATIO	SEQ_RATIO の指定値	
		なし			ナル値	
	なし	–			更新前の設定値	
NENTRY	あり	あり		N_ENTRY	N_ENTRY の指定値	
		なし			ナル値	

項目名		指定値	格納先のディクショナリ表名	格納先の列名	設定値	格納先のディクショナリ表の設定対象行
	なし	–			更新前の設定値	
NUNIQUE	あり	あり	SQL_COLUMN_STATISTICS	N_UNIQUE	NUNIQUE の指定値	COLUMN で指定した列
		なし			ナル値	
	なし	–			更新前の設定値	
NNULLS※1	あり	あり		N_NULL	NNULLS の指定値	
		なし			ナル値	
	なし	–			更新前の設定値	
N_MAX_DUP_KEY	あり	あり		N_MAX_DUP_KEY	N_MAX_DUP_KEY の指定値	
		なし			ナル値	
	なし	–			更新前の設定値	
N_MIN_DUP_KEY	あり	あり		N_MIN_DUP_KEY	N_MIN_DUP_KEY の指定値	
		なし			ナル値	
	なし	–			更新前の設定値	
MAX_VALUE※1	あり	あり		RANGE_VALUES	MAX_VALUE の指定値	
		なし			ナル値	
	なし	–			ナル値	
MIN_VALUE※1	あり	あり			MIN_VALUE の指定値	
		なし			ナル値	
	なし	–			ナル値	
SECTION_COUNT	あり	あり			SECTION_COUNT の指定値※2	
		なし			0	
	なし	–			0	
SECTION_NO	あり	あり			SECTION_NO の指定値	
		なし			エラー	

項目名		指定値	格納先のディクショナリ表名	格納先の列名	設定値	格納先のディクショナリ表の設定対象行
	なし	−			エラー	
SEC_TOTAL_COUNT	あり	あり			SEC_TOTAL_COUNT の指定値	
		なし			エラー	
	なし	−			エラー	
SEC_UNIQUE	あり	あり			SEC_UNIQUE の指定値	
		なし			エラー	
	なし	−			エラー	
SEC_MAX_VALUE	あり	あり			SEC_MAX_VALUE の指定値	
		なし			エラー	
	なし	−			エラー	

（凡例）

−：該当しません。

注※1

NROWS，NNULLS，MAX_VALUE，又は MIN_VALUE のどれかを省略した場合，SQL_COLUMN_STATISTICS の RANGE_VALUES 列にナル値を設定します。

注※2

区関数（SECTION_COUNT），及び区間分布情報の項目をすべて省略した場合，0 が設定されます。区関数分の区間分布情報の指定項目のどれかを省略した場合はエラーとなります。

16.4.3　最適化情報パラメタファイルの項目の出力内容

最適化情報を退避するとき，最適化情報パラメタファイルの各項目で参照するディクショナリ表と列名，及び出力内容を次の表に示します。

表 16–11　最適化情報パラメタファイルの項目の出力内容

分類	項目名	参照するディクショナリ表名と列名		出力内容		
		表名	列名	該当列なし	参照列がナル値	参照列がナル値以外
表最適化情報	NROWS	SQL_TABLE_STATISTICS	N_ROW	×	△	○

分類	項目名	参照するディクショナリ表名と列名		出力内容		
		表名	列名	該当列なし	参照列がナル値	参照列がナル値以外
	NPAGES		N_PAGE	×	△	○
インデクス最適化情報	INDEX	SQL_INDEX_STATISTICS	INDEX_NAME	×	−	○※1
	NIPAGES		N_IXPG	×	△	○
	NLEVEL		N_LEVEL	×	△	○
	SEQ_RATIO		SEQ_RATIO	×	△	○
	NENTRY		N_ENTRY	×	△	○
列最適化情報	COLUMN	SQL_COLUMN_STATISTICS	COLUMN_NAME	×	−	○※1
	NUNIQUE		N_UNIQUE	×	△	○
	NNULLS		N_NULL	×	△	○
	N_MAX_DUP_KEY		N_MAX_DUP_KEY	×	△	○
	N_MIN_DUP_KEY		N_MIN_DUP_KEY	×	△	○
	MAX_VALUE		RANGE_VALUES	×	×	○※2
	MIN_VALUE			×	×	○※2
区間分布情報	SECTION_COUNT			×	×	○
	SECTION_NO			×	×	○
	SEC_TOTAL_COUNT			×	×	○
	SEC_UNIQUE			×	×	○
	SEC_MAX_VALUE			×	×	○※2

（凡例）

○：項目名と列値が出力されます。

△：項目名だけ出力されます。

×：出力されません。

−：該当しません。

注※1

引用符（"）で囲んで出力します。

注※2

出力値の詳細については，表「インデクス構成列のキー値の分布情報として取得又は登録する情報」を参照してください。なお，データ型が CHAR，NCHAR，MCHAR，VARCHAR，NVARCHAR，MVARCHAR

の場合は，英字の大文字を使用した 16 進表記で出力します。16 進表記は，X'****'の形式となります。****の部分は，16 進文字列（0～9，及び A～F）が 2 文字単位で出力されます。

16.4.4　最適化情報パラメタファイルの例

(1)　記述例

最適化情報パラメタファイルの記述例を次に示します。

```
# 表最適化情報
NROWS            100          # 表の全行数
NPAGES           5            # 表のデータページ数
# インデクス最適化情報
INDEX            IDX_I01      # インデクス名
NIPAGES          2            # インデクスページ数
NLEVEL           2            # インデクスの段数
SEQ_RATIO        80           # インデクスのシーケンシャル度
NENTRY           50           # インデクスのキー値数
# 列最適化情報
COLUMN           COL_01       # 列名
NUNIQUE          50           # 列のユニーク値数
NNULLS           2            # 列のナル値数
N_MAX_DUP_KEY    5            # 列の最大重複列値の数
N_MIN_DUP_KEY    1            # 列の最小重複列値の数
MAX_VALUE        ZZZ          # 列値の最大値
MIN_VALUE        ABC          # 列値の最小値
# 区間分布情報
SECTION_COUNT    2            # 区間数
SECTION_NO       1            # 区間番号
SEC_TOTAL_COUNT  1            # 区間要素数の累積度数
SEC_UNIQUE       1            # 区間内のユニーク値数
SEC_MAX_VALUE    ABC          # 区間内の最大値
SECTION_NO       2
SEC_TOTAL_COUNT  2
SEC_UNIQUE       1
SEC_MAX_VALUE    DEF
```

(2)　出力例

最適化情報パラメタファイルの出力例を次に示します。

```
# STATISTICS INFORMATION FOR SQL OPTIMIZATION
# TABLE "USER1"."TBL1"
# TABLE INFORMATION
# "USER1"."TBL1"
NROWS            100          # 表の全行数
NPAGES           5            # 表のデータページ数
# INDEX INFORMATION
INDEX            "IDX_I01"    # インデクス名
NIPAGES          2            # インデクスページ数
NLEVEL           2            # インデクスの段数
```

```
SEQ_RATIO        80          # インデクスのシーケンシャル度
NENTRY           50          # インデクスのキー値数
# COLUMN INFORMATION
COLUMN           "COL_01"    # 列名
NUNIQUE          50          # 列のユニーク値数
NNULLS           2           # 列のナル値数
N_MAX_DUP_KEY    5           # 列の最大重複列値の数
N_MIN_DUP_KEY    1           # 列の最小重複列値の数
MAX_VALUE        X'5A5A5A'   # 列値の最大値
MIN_VALUE        X'414243'   # 列値の最小値
## SECTION INFORMATION
SECTION_COUNT    2           # 区間数
SECTION_NO       1           # 区間番号
SEC_TOTAL_COUNT  1           # 区間要素数の累積度数
SEC_UNIQUE       1           # 区間内のユニーク値数
SEC_MAX_VALUE    X'414243'   # 区間内の最大値
SECTION_NO       2
SEC_TOTAL_COUNT  2
SEC_UNIQUE       1
SEC_MAX_VALUE    X'444546'
```

16.4.5　最適化情報パラメタファイルの記述規則

- 各項目の名称は，1 バイト目から記述してください。また，各項目は 1 行以内で記述してください。
- 項目名と値の間は，空白（全角）又はタブにしてください。
- 1 行に複数の項目及び値は記述できません。また，1 行は 80 バイトで，改行記号は含まれません。
- ＃から改行記号までの間は，コメントとなります。
- 区間分布情報を除き，表最適化情報，インデクス最適化情報，及び列最適化情報内で，同一の項目は複数記述できません。
- 次を指定する場合は，引用符（"）で囲んでください。

  ・指定値に空白を含む場合

  ・長さ 0 の文字列を指定する場合

  ・「x'」，又は「X'」で始まり「'」で終わる文字列を指定する場合

  ・半角英小文字と半角英大文字を区別して，インデクス名称，又は列名称を指定する場合

  インデクス名称，又は列名称を引用符（"）で囲まないで指定した場合は，半角英小文字を半角英大文字として扱います。

  また，指定値に￥がある場合は￥を無視し，￥の直後のデータを有効とします。データとして￥を記述する場合は，￥￥と記述してください。
- 各項目の指定順序は，インデクス最適化情報指定内及び列最適化情報指定内で順不同です。インデクス最適化情報指定内，列最適化情報指定内とは，インデクス名又は列名を記述してから，次のインデクス名又は列名を記述するまでの間を意味します。

  ただし，列最適化情報の定義のうち，区間数は区間分布情報より先に記述する必要があります。また，区間番号は区間分布情報の各区間定義の最初に記述してください。

- 同一のインデクス名又は列名を 2 回以上指定した場合，後に指定したものが有効となります。

16.5　規則及び注意事項

16.5.1　規則

1. pdgetcst は，HiRDB が稼働中のときだけ実行できます。
2. pdgetcst は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. pdgetcst は，ディクショナリ表，ビュー表，及び一時表に対して実行できません。なお，次に示す場合はエラーメッセージが出力されます。

   表単位で最適化情報の退避を実行したとき

   　最適化情報を収集及び登録できないため，KFPN00014-I メッセージが出力され，正常終了します。

   スキーマ単位で最適化情報の登録又は最適化情報の退避を実行したとき

   　スキーマ単位の実行をサポートしていないため，KFPN00100-E メッセージが出力され，エラー終了します。
4. 最適化情報パラメタファイルに抽象データ型の列は指定できません。
5. pdgetcst で収集した最適化情報は，pdgetcst 実行後から SQL の最適化処理で使用できるようになります。
6. pdgetcst を実行して登録した最適化情報と実際のデータベースの状態が一致していないと，適切なアクセスパスを選択できないため，性能が劣化することがあります。データベースの状態が変わる SQL を実行した場合は，pdgetcst を実行して最適化情報と一致させてください。
7. 本番環境とテスト環境がある場合，表の行数，データ値の分布が同じ最適化情報になるようにしてください。ただし，最適化情報収集レベルが lvl1 の場合は，表の行数を同じにするだけでよいです。
8. pdgetcst を実行する場合，データディクショナリ用 RD エリアは閉塞解除状態かつオープン状態でなければなりません。
9. pdgetcst の処理対象となる表，インデクスが格納されている RD エリアの状態によって，その RD エリアに対する pdgetcst の実行可否が決まります。pdgetcst の実行可否については，「コマンド実行時の RD エリアの状態」を参照してください。
10. 出力結果ファイルと最適化情報パラメタファイルについては，HiRDB 管理者に対してアクセス権限を与えておく必要があります。

16.5.2　注意事項

1. pdgetcst のリターンコードを次に示します。

   0：正常終了

   4：警告終了（ディクショナリ表操作で警告発生）

   8：異常終了

2. pdgetcst の実行中に，処理対象の表及びインデクスの定義変更はしないでください。定義を変更した場合は，HiRDB の動作が保証されません。
3. pdgetcst の実行中に，処理対象の表に対して行の挿入，削除，又はデータの更新をした場合，最適化情報に誤差が生じることがあります。
4. pdgetcst は，表にデータがある状態で実行してください。表定義直後など，データが１件もない状態で実行すると，正しいコスト情報を取得できません。
5. 最適化情報パラメタファイルに記述されている，文字列の文字コードが HiRDB システムで使用されている文字コードと異なる場合，pdgetcst の動作は保証しません。
6. 同一の表に対して，pdgetcst を同時に実行しないでください。同時に実行した場合，HiRDB の動作は保証されません。ディクショナリ表に最適化情報を登録する pdgetcst を同時に実行した場合，処理の終了を示す KFPN00013-I メッセージが出力されますが，最適なアクセスパスにならないおそれがあります。
7. pdgetcst がエラー終了した場合，HiRDB の動作は保証されません。出力されているエラーメッセージを参照してエラー要因を取り除き，再度 pdgetcst を実行してください。pdgetcst を再実行してもエラーが続く場合は，マニュアル「HiRDB システム運用ガイド」の「最適化情報の登録，更新，又は削除時にエラーが発生した場合」を参照して対処してください。
8. pdgetcst を実行するとアクセス手順が変更になるので，実際のデータ特性が最適化情報と一致していないときは，性能が劣化することもあります。したがって，pdgetcst を実行する場合には，十分にテストをするようにしてください。
9. 本番環境とテスト環境がある場合で，本番環境で最適化情報を取得するときは，テスト環境と同等の最適化情報を取得するため，テスト環境及び本番環境でデータ件数，データ値の分布は同じにしてください。

16.5.3　BOM を持つファイルの使用

pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdgetcst の最適化情報パラメタファイルには，BOM を持つファイルを使用できます。なお，pdgetcst の最適化情報パラメタファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdgetcst で出力されるファイルには BOM は付きません。

16.6　出力形式

16.6.1　検索による最適化情報の収集

(1)　最適化情報収集レベルが lvl2 の場合

最適化情報収集ユティリティの実行結果（-c lvl2 の検索による最適化情報の収集）を次に示します。

```
**      pdgetcst                                1995-03-31 12:00
                                                        [1]
  **  TABLE INFORMATION  **
  AUTHORIZATION          : user-id [2]
  TABLE NAME             : TABLE01 [3]
  ROW COUNT              : 120     [4]
  DATA PAGES             : 5       [5]
  CACHE SIZE             : 0       [6]

  **  INDEX-COLUMN INFORMATION  **
  INDEX NAME             : Idx001  [7]
  INDEX PAGES            : 5       [8]
  INDEX LEVELS           : 3       [9]
  COLUMN NAME            : Clm1    [10]
  COLUMN TYPE            : INTEGER [11]
  SEQUENTIAL RATIO       : 100     [12]
  MAX VALUE              : 2500    [13]
  MIN VALUE              : 0       [14]
  UNIQUE KEYS            : 23      [15]
  NULL VALUE COUNT       : 20      [16]
  MAX DUPLICATE KEY      : 10      [17]
  MIN DUPLICATE KEY      : 1       [18]
  SECTION COUNT          : 39      [19]

    **  SECTION INFORMATION  **
    SECTION NO           :1  [20]
    TOTAL COUNT          :1  [21]
    UNIQUE KEYS          :1  [22]
    MAX VALUE            :0  [23]

    **  SECTION INFORMATION  **
    SECTION NO           :2
    TOTAL COUNT          :2
    UNIQUE KEYS          :1
    MAX VALUE            :10

    **  SECTION INFORMATION  **
    SECTION NO           :3
    TOTAL COUNT          :6
    UNIQUE KEYS          :1
    MAX VALUE            :10

             :

    **  SECTION INFORMATION  **
```

```
    SECTION NO          :39
    TOTAL COUNT         :100
    UNIQUE KEYS         :1
    MAX VALUE           :2500

**  pdgetcst ENDED    RETURN CODE = 0 [24]
```

[説明]

** SECTION INFORMATION **は，区間数分繰り返し表示します。** INDEX-COLUMN INFORMATION **（** SECTION INFORMATION **も含む）は，インデクス数分繰り返し表示します。

1. 最適化情報収集ユティリティを実行した（pdgetcst コマンドを入力した）日付と時刻を，YYYY-MM-DD hh:mm の形式で表示します。
   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分
2. 最適化情報を取得する表の認可識別子を表示します。
3. 最適化情報を取得する表の表識別子を表示します。
4. 表の行数を表示します。
5. 表の使用中データページ数を表示します。
6. HiRDB が利用する作業用の領域を表示します。
7. 最適化情報を取得するインデクスの名称を表示します。
8. インデクスの使用中インデクスページ数を表示します。
9. インデクスの段数を表示します。
10. インデクスの第 1 構成列の名称を表示します。
11. インデクスの第 1 構成列のデータ型を表示します。
12. シーケンシャル度を表示します。
13. インデクスの第 1 構成列の列値の最大値を表示します。
14. インデクスの第 1 構成列の列値の最小値を表示します。
15. インデクスの第 1 構成列のユニーク値の数を表示します。
16. インデクスの第 1 構成列のナル値の数を表示します。
17. インデクスの第 1 構成列の最大重複値の数を表示します。
18. インデクスの第 1 構成列の最小重複値の数を表示します。
19. 区間分布情報の区間数を表示します。
20. 区間番号を表示します。
21. 区間データ数の累計を表示します。
22. 区間内のユニーク値の数を表示します。
23. 区間内の最大値を表示します。
24. リターンコードを表示します。

注

-d オプション指定時は，8，9，及び 12～19 は出力されません。また，行数が 0 の場合には，13，14 に空白，それ以外の項目には 0 が出力されます。

(2)　最適化情報収集レベルが lvl1 の場合

最適化情報収集ユティリティの実行結果（-c lvl1 の検索による最適化情報の収集）を次に示します。

```
**      pdgetcst                                2000-03-31 12:00
                                                        [1]

  **  TABLE INFORMATION  **
  AUTHORIZATION         : user-id [2]
  TABLE NAME            : TABLE01 [3]
  ROW COUNT             : 120     [4]    [6]
  CACHE SIZE            : 0       [5]

**  pdgetcst ENDED    RETURN CODE = 0 [7]
```

[説明]

1. 最適化情報収集ユティリティを実行した（pdgetcst コマンドを入力した）日付と時刻を，YYYY-MM-DD hh:mm の形式で表示します。

   YYYY：年（西暦）　MM：月　DD：日　hh：時　mm：分

2. 最適化情報を取得する表の認可識別子を表示します。
3. 最適化情報を取得する表の表識別子を表示します。
4. 表の行数を表示します。
5. HiRDB が利用する作業用の領域を表示します。
6. -t ALL の場合は，表数分繰り返して表示されます。
7. リターンコードを表示します。

16.6.2　最適化情報パラメタファイルによる最適化情報の登録

```
  **     pdgetcst                                 1997-10-30 15:06
                                                          [1]
    **  TABLE INFORMATION    **
    AUTHORIZATION       : MANUAL [2]
    TABLE NAME          : TBL01  [3]
    ROW COUNT           : 1      [4]
    DATA PAGES          : 1      [5]
    CACHE SIZE          : 2560   [6]

    **  INDEX INFORMATION  **
    INDEX NAME          : IDX_TBL01_C1 [7]
    INDEX PAGES         : 1            [8]
    INDEX LEVELS        : 1            [9]
```

```
  SEQUENTIAL RATIO    : 1            [10]
  INDEX KEY           : 1            [11]

  **  COLUMN INFORMATION  **
  COLUMN NAME         : C1   [12]
  COLUMN TYPE         : CHAR [13]
  UNIQUE KEYS         : 1    [14]
  NULL VALUE COUNT    : 1    [15]
  MAX DUPLICATE KEY   : 1    [16]
  MIN DUPLICATE KEY   : 1    [17]
  MAX VALUE           : "1"  [18]
  MIN VALUE           : "1"  [19]
  SECTION COUNT       : 100  [20]


**  pdgetcst ENDED    RETURN CODE = 0 [21]   **
```

[説明]

** COLUMN INFORMATION **は，列数分繰り返し表示します。** INDEX INFORMATION **及び** COLUMN INFORMATION **は，インデクス数分繰り返し表示します。

1. 最適化情報収集ユティリティを実行した（pdgetcst コマンドを入力した）日付と時刻を，YYYY-MM-DD hh:mm の形式で表示します。

   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分
2. 最適化情報を登録する表の認可識別子を表示します。
3. 最適化情報を登録する表の表識別子を表示します。
4. 表の行数を表示します。
5. 表の使用中データページ数を表示します。
6. HiRDB システムが利用する作業用の領域を表示します。
7. 最適化情報を登録するインデクスの名称を表示します。
8. インデクスの使用中インデクスページ数を表示します。
9. インデクスの段数を表示します。
10. シーケンシャル度を表示します。
11. インデクスのキー値数を表示します。
12. インデクスを定義している列の名称を表示します。
13. インデクスを定義している列のデータ型を表示します。
14. インデクスを定義している列の，列値のユニーク値の数を表示します。
15. インデクスを定義している列の，列値のナル値の数を表示します。
16. インデクスを定義している列の，列値の最大重複値の数を表示します。
17. インデクスを定義している列の，列値の最小重複値の数を表示します。
18. インデクスを定義している列の，列値の最大値を表示します。

19. インデクスを定義している列の，列値の最小値を表示します。

20. 区間分布情報の区間数を表示します。

21. リターンコードを表示します。

16.7　使用例

最適化情報収集ユティリティ（pdgetcst）の使用例を次に示します。

なお，使用例では，HiRDB に接続する認可識別子を USERID，表の所有者の認可識別子を AUTHID，パスワードを PASSWD とします。

16.7.1　最適化情報を収集する場合

表 T1 の最適化情報を収集する場合は，次のように指定します。

pdgetcst -u USERID -a AUTHID -p PASSWD -t T1

16.7.2　最適化情報を収集し，処理結果をファイルに出力する場合

表 T2 の最適化情報を収集し，処理結果をファイル（/tmp/LOG）に出力する場合は，次のように指定します。

pdgetcst -u USERID -a AUTHID -p PASSWD -t T2 -l /tmp/LOG

16.7.3　最適化情報をディクショナリ表から削除する場合

表 T3 の最適化情報をディクショナリ表から削除する場合は，次のように指定します。

pdgetcst -u USERID -a AUTHID -p PASSWD -t T3 -d

16.7.4　最適化情報パラメタファイルの最適化情報を登録する場合

最適化情報パラメタファイル（/tmp/prmfile）の内容を登録する場合は，次のように指定します。なお，表 T4 に関する最適化情報だけ登録するものとします。

pdgetcst -u USERID -a AUTHID -p PASSWD -t T4 -s /tmp/prmfile

16.7.5　最適化情報を最適化情報パラメタファイルに退避する場合

表 T5 の最適化情報を最適化情報パラメタファイル（/tmp/prmfile）に退避する場合は，次のように指定します。

pdgetcst -u USERID -a AUTHID -p PASSWD -t T5 -e /tmp/prmfile

17 アクセスパス表示ユティリティ（pdvwopt）

この章では，SQL 最適化処理によって決定したアクセスパスの情報を表示するアクセスパス表示ユティリティ（pdvwopt）について説明します。

17.1　概要

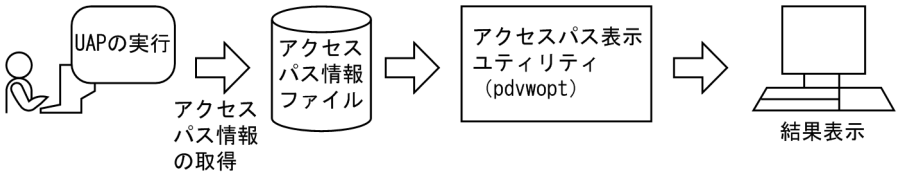
実行者　権限がないユーザでも実行できます

17.1.1　機能

HiRDB は，ユーザが入力した SQL に対し，SQL ごとに SQL 最適化処理をして，データベースへの最適なアクセス方法（アクセスパス）を決定します。アクセスパス表示ユティリティ（pdvwopt）は，この SQL 最適化処理によって決定したアクセスパスの情報を表示するユティリティです。ユーザは，この表示されたアクセスパスを分析することで，HiRDB のチューニングに役立てることができます。

アクセスパス表示ユティリティ（pdvwopt）の概要を次の図に示します。

図 17–1　アクセスパス表示ユティリティ（pdvwopt）の概要



17.2　コマンドの形式

17.2.1　形式

```
pdvwopt 〔-d〕 アクセスパス情報ファイル
```

17.2.2　オプション

●-d

結合処理情報，実表検索処理情報に，表 ID 及びインデクス ID を表示する場合に指定します。

17.2.3　引数

●アクセスパス情報ファイル　～＜パス名＞

クライアント環境定義 PDVWOPTMODE に 1 以上を指定して UAP を実行すると，アクセスパス情報ファイルにアクセスパスの情報が格納されます。

アクセスパス表示ユティリティを実行するときに，対象となるアクセスパス情報ファイルをここで指定します。クライアント環境定義 PDVWOPTMODE については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

アクセスパス情報ファイルは，UAP の接続したシングルサーバ又はフロントエンドサーバがあるユニットの SQL 情報ディレクトリ（%PDDIR%¥spool¥pdsqldump）下に作成します。CONNECT 文から DISCONNECT 文までの単位で情報が取得されます。アクセスパス情報ファイルの名称は，HiRDB が次のように付けます。

```
認可識別子_UAPソースファイル名_識別番号_サーバ種別
```

認可識別子

CONNECT 文で指定した認可識別子です。

UAP ソースファイル名

UAP ソースファイルの名称です。

名称が 30 バイトを超える場合，先頭から 30 バイトまでとなります。

会話型 SQL 実行ユティリティから SQL を実行した場合は，pdsql-プロセス ID，pdsql.exe-プロセス ID，又は pdsqlw-プロセス ID となります。

識別番号

CONNECT 文から DISCONNECT 文までを一つの単位とし，それを識別するための番号です。

サーバ種別

HiRDB/シングルサーバの場合は s，HiRDB/パラレルサーバの場合は p です。

注意事項

認可識別子又は UAP ソースファイル名称に¥が含まれている場合，¥をファイルパス名に使用するとパスセパレータとして扱われるため，「¥」を「(」に置き換えます。

＜アクセスパス情報ファイルの名称の例＞

```
USER1_sample.ec_1-1-01_s
USER2_pdsql-00001_1-1-01_p
```

17.3　実行手順

アクセスパス表示ユティリティの実行手順を次に示します。

＜クライアント側＞

1.

クライアント環境変数を設定します。クライアント環境変数については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

2.

アクセスパス情報を取得したい SQL がある UAP を実行します。UAP を終了しなくても，CONNECT 文～DISCONNECT 文が実行されていれば，アクセスパス情報ファイルは作成されます。

＜サーバ側＞

3.

調査対象となる UAP が接続した，シングルサーバ又はフロントエンドサーバがあるサーバマシンでアクセスパス表示ユティリティを実行します。

＜実行例＞

```
pdvwopt c:¥hirdb¥spool¥pdsqldump¥USER1_pdsql-00001_1-1-02_p
```

17.4　規則及び注意事項

17.4.1　規則

1. アクセスパス表示ユティリティは，HiRDB の稼働に関係なく実行できます。
2. アクセスパス表示ユティリティは，調査対象となる UAP の接続したシングルサーバ又はフロントエンドサーバがあるサーバマシンで実行してください。

17.4.2　注意事項

(1)　リターンコードについて

pdvwopt 実行時のリターンコードは，0 の場合は正常終了，8 の場合は異常終了となります。

(2)　アクセスパス情報ファイル取得時の注意事項

1. クライアント環境定義の PDVWOPTMODE に 1 以上を設定していると，アクセスパス情報ファイルを作成するためのサーバ負荷が増加します。そのため，アクセスパス情報を取得する必要がある場合だけ設定するようにしてください。ただし，UAP レポート機能でアクセスパス情報の出力を指定していると，サーバ負可が増加することがあります。UAP レポート機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
2. クライアント環境定義の PDVWOPTMODE に 1 を設定した場合，調査対象となる SQL の SQL オブジェクトがキャッシュヒットすると，SQL 最適化処理をしないため，以前に作成した SQL オブジェクトを使用して検索します。このため，アクセスパス情報は取得されません。
3. クライアント環境定義の PDVWOPTMODE に 2 を設定していると，調査対象となる SQL の SQL オブジェクトがキャッシュヒットする場合でも，アクセスパス情報を取得します。したがって，キャッシュヒットに関係なく SQL 最適化処理をして SQL オブジェクトを作成するため，PDVWOPTMODE に 1 を設定した場合に比べてサーバ負荷が増えることがあります。
4. ストアドプロシジャに記述した SQL については，CALL 文を実行してもアクセスパス情報は取得されません。この場合，次のどれかの方法でアクセスパス情報を取得してください（ただし，現在のコスト情報を基に SQL オブジェクトを再作成するため，従来のアクセスパスと変わることがあります）。
   - ALTER PROCEDURE を実行して SQL オブジェクトを再作成してください。
   - DROP PROCEDURE でストアドプロシジャを削除し，CREATE PROCEDURE で SQL オブジェクトを再作成してください。
5. トリガに記述したトリガ SQL については，トリガが実行されてもアクセスパス情報は取得されません。トリガ SQL のアクセスパス情報は，次の方法で取得してください（ただし，現在のコスト情報を基に SQL オブジェクトを再作成するため，従来のアクセスパスと変わることがあります）。

- ALTER TRIGGER を実行して SQL オブジェクトを再作成してください。

6. HiRDB はアクセスパス情報ファイルを削除しません。したがって，不要になったアクセスパス情報ファイルは削除してください。ただし，SQL 情報ディレクトリ（%PDDIR%¥spool¥pdsqldump）は削除しないように注意してください。削除した場合は，アクセスパス情報ファイルを作成できなくなります。
7. CREATE TABLE の参照制約動作に CASCADE を指定した参照表を定義すると，内部的にトリガが生成されるため，参照表定義時にアクセスパス情報を取得します。
8. 参照表定義時に生成したトリガについては，トリガが実行されてもアクセスパス情報は取得されません。この場合，ALTER ROUTINE を実行して SQL オブジェクトを再作成し，アクセスパス情報を取得してください。なお，現在のコスト情報を基に SQL オブジェクトを再作成するため，既存のアクセスパスと変わることがあります。
9. VALUES 指定の INSERT 文を実行した場合，次の機能を実行したときだけアクセスパス情報を取得します。
   - トリガ
   - 参照制約
   - スカラ副問合せ

(3)　アクセスパス表示ユティリティ実行時の注意事項

アクセスパス表示ユティリティとアクセスパス情報ファイルのバージョンが異なるとエラーとなります。バージョンの一致したアクセスパス表示ユティリティとアクセスパス情報ファイルを使用してください。

(4)　出力形式の出力情報についての注意事項

アクセスパス表示ユティリティの実行結果には｛｝で囲まれた情報が表示されますが，これはシステム内部情報です。

17.5　出力形式

```
Version         :aa...a bb...bBES
UAP Name        :cc...c
Authorization   :dd...d
-----------------------------------------------------
Section No      :ee...e
UAP Source      :ff...f
Optimize Time   :gggg-gg-gg gg:gg:gg.gggggg
Optimize Mode   :hh...h
SQL Opt Level   :iiiiiiiiii(ii...i) = jj...j(kk...k),...
Add Opt Level   :llllllllll(ll...l) = mm...m(nn...n),...
Routine Name    :ss...s.ss...s
SQL             :oo...o
Work Table      :pp...p
Total Cost      :qq...q
Trigger Name    :tt...t
                     Procedure Name :uu...u
                     Action Time    :vv...v(ww...w)
                     Action Type    :xx...xx
----- QUERY EXPRESSION BODY ID : ... -----  ...................1
 :
----- QUERY ID : ... ----- ....................................2
 :
JOIN  ..................................................... 3
 :
SCAN  ..................................................... 4
 :
```

[説明]

1. 集合演算情報です。集合演算情報については，「集合演算処理情報」を参照してください。
2. 問合せ処理情報です。問合せ処理情報については，「問合せ処理情報」を参照してください。
3. 結合処理情報です。結合処理情報については，「結合処理情報」を参照してください。
4. 実表検索処理情報です。実表検索処理情報については，「実表検索処理情報」を参照してください。

Section No 以降の情報は，SQL の個数分繰り返して表示します。

aa...a：

HiRDB のバージョンを表示します。修正版 HiRDB の場合は，バージョンの後ろに記号が付きます。

bb...b：

HiRDB/パラレルサーバでのバックエンドサーバ数を表示します。HiRDB/シングルサーバの場合は表示しません。

cc...c：

クライアント環境定義の PDCLTAPNAME で指定した UAP 名称を表示します。

dd...d：

認可識別子を表示します。

ee...e：

セクション番号（SQL の対応を確認するための番号）を表示します。

ff...f：

UAP ソースファイルの名称を表示します。

名称が 30 バイトを超える場合，先頭から 30 バイトまでとなります。

会話型 SQL 実行ユティリティから SQL を実行した場合は，pdsql-プロセス ID，pdsql.exe-プロセス ID，又は pdsqlw-プロセス ID となります。

gggg-gg-gg gg:gg:gg.gggggg：

SQL 最適化処理をした年−月−日　時：分：秒．マイクロ秒を表示します。

hh...h：

SQL 最適化モードを表示します。SQL 最適化モードについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

COST_BASE_1：

コストベース最適化モード 1 が適用されています。

COST_BASE_2：

コストベース最適化モード 2 が適用されています。

iiiiiiiiii(ii...i) = jj...j(kk...k),...：

SQL コンパイル時の SQL 最適化オプションの値を表示します。

iiiiiiiiii(ii...i)：

SQL 最適化オプションの値を 16 進数（10 進数）で表示します。

jj...j(kk...k),...：

SQL 最適化オプションに指定している機能名を，識別子（符号なし整数）で表示します。複数の機能を指定している場合は，コンマで区切って表示します。

lllllllll(ll...l) = mm...m(nn...n),...：

SQL コンパイル時の SQL 拡張最適化オプションの値を表示します。

lllllllll(ll...l)：

SQL 拡張最適化オプションの値を 16 進数（10 進数）で表示します。

mm...m(nn...n),...：

SQL 拡張最適化オプションに指定している機能名を，識別子（符号なし整数）で表示します。複数の機能を指定している場合は，コンマで区切って表示します。

oo...o：

コンパイルした SQL 文を表示します。

pp...p：

SQL 文を実行するための，作業表作成の契機の数を表示します。作業表を作成しないで実行する場合には 0 を表示します。

ソートをしない契機では，作業表を一つ作成します。ソートをする契機では，作業表を二つ作成します。

作業表作成の契機は，必ずしも同時とは限らないため，作業表用ファイルの容量見積もりで見積もった値より，少ない値で SQL 文を実行できる場合があります。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

qq...q：

HiRDB がコストベース最適化を用いて計算した，問合せの総コストを表示します。なお，この行は SQL 最適化モードがコストベース最適化モード 2 の場合に表示します。

ss...s.ss...s：

ルーチン中の SQL の場合，ルーチン名を「認可識別子.ルーチン識別子」の形式で表示します。ルーチン中の SQL でない場合は表示しません。

tt...t：

HiRDB が実行するトリガの名称を表示します。実行するトリガがない場合は，tt...t～xx...x を表示しません。

実行するトリガが複数ある場合は，複数行に分けて表示します。複数行に分けて表示する場合は，実行順となります。

uu...u：

トリガ動作手続きの名称を表示します。

vv...v：

トリガ動作時期の種別を表示します。

BEFORE：

表に対する操作の前に実行するトリガ

AFTER：

表に対する操作の後に実行するトリガ

ww...w：

参照表定義時に定義した生成したトリガの場合，参照制約動作の種別を表示します。

CASCADE：

参照制約動作が CASCADE

xx...x：

トリガ動作の単位の種別を表示します。

FOR EACH ROW：

行単位トリガ

FOR EACH STATEMENT：

文単位トリガ

トリガ動作時期が BEFORE の行単位トリガ（TRIGGER1），トリガ動作時期が AFTER の文単位トリガ（TRIGGER2），参照動作が CASCADE で更新動作のときに生成したトリガ（URA2004050521222324）を実行する場合を例にすると，出力結果は次のようになります。

```
Trigger Name   : TRIGGER1
                     Procedure Name : (TRIG2003030405060708)
                     Action Time    : BEFORE
                     Action Type    : FOR EACH ROW
                 (URA2004050521222324)
                     Procedure Name : (TRIG2004050521222324)
                     Action Time    : AFTER(CASCADE)
                     Action Type    : FOR EACH ROW
                 TRIGGER2
                     Procedure Name : (TRIG2003091011121314)
                     Action Time    : AFTER
                     Action Type    : FOR EACH STATEMENT
```

17.5.1　集合演算処理情報

```
----- QUERY EXPRESSION BODY ID : aa...a -----
Query Exp Type : bb...b cc...c(cc...c)  dd...d{AA...A}
Sub Query Type : ee...e{BB...B} [ff...f]
Transfer Type  : gg...g-CLM hh...h
Order by Mode  : ii...i
FLT Server     : jj...j
SetOpe Process : kk...k = [ll...l mm...m{CC...C}] nn...n [ll...l mm...m{CC...C}]
                    :
```

[説明]

aa...a：

問合せ式本体 ID を表示します。

集合演算を含む問合せ式本体ごとに番号を付けます。コンパイルした SQL が複数の問合せ式本体で構成される場合，この行で情報を区切ります。

bb...b：

問合せ式の種類を表示します。

QUERY：

問合せ

DERIVED TABLE：

FROM 句に問合せ式本体を指定した場合の導出表の問合せ

SUBQUERY：

副問合せ

WITH：

WITH 句問合せ

VIEW：

ビュー定義の問合せ

cc...c(cc...c)：

問合せ式本体の種類によって，次の情報を表示します。

- WITH 句の問合せ式本体の場合，WITH 句問合せ名（相関名）
- ビュー定義の問合せ式本体の場合，ビュー表名（相関名）
- FROM 句に指定した導出表中の問合せ式本体の場合，"(NO NAME)"，又は"(NO NAME)(相関名)"

dd...d{AA...A}：

集合演算結果に対して作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。

ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

ee...e：

副問合せの実行方式を表示します。副問合せの実行方式については，マニュアル「HiRDB UAP 開発ガイド」の「外への参照のない副問合せの実行方式」及び「外への参照のある副問合せの実行方式」を参照してください。

WORK TABLE ATS SUBQ：

副問合せの処理方式が作業表 ATS 実行であることを示します。

WORK TABLE SUBQ：

副問合せの処理方式が作業表実行であることを示します。

ROW VALUE SUBQ：

副問合せの処理方式が行値実行であることを示します。

HASH SUBQ：

副問合せの処理方式がハッシュ実行であることを示します。

NESTED LOOPS WORK TABLE SUBQ：

副問合せの処理方式がネストループ作業表実行であることを示します。

NESTED LOOPS ROW VALUE SUBQ：

副問合せの処理方式がネストループ行値実行であることを示します。

ff...f：

副問合せの実行方式で SQL 最適化指定が指定されている場合，その指定が有効であったかどうかを表示します。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。

AS SPECIFIED：

指定が有効であったため，SQL 最適化指定を適用しました。

SPECIFICATION IGNORED：

指定が無効であったため，SQL 最適化指定を無視しました。

gg...g-CLM hh...h：

副問合せ，WITH 句問合せ式本体，ビュー定義，及び HiRDB が内部的に問合せ式本体を作成する場合，上位の問合せに転送をするときの転送方法を表示します。

gg...g は転送に使用する列数，hh...h は転送方法の種類となります。転送方法の種類については，「転送方法の種類」を参照してください。

ii...i：

集合演算処理の ORDER BY の処理方式を表示します。ORDER BY を指定していない場合，この行を表示しません。ORDER BY の処理方式については，「ORDER BY 処理方式の種類」を参照してください。

HiRDB/シングルサーバの場合は，問合せ処理情報にも問合せ処理の ORDER BY の処理方式を表示します。

jj...j：

集合演算処理のための，フロータブルサーバの要求数を表示します。フロータブルサーバが不要な場合は 0 を表示します。HiRDB/シングルサーバの場合は表示しません。

kk...k = [ll...l mm...m{CC...C}] nn...n [ll...l mm...m{CC...C}]：

集合演算の演算順序を表示します。集合演算を複数指定した場合は，複数行に分けて表示します。

kk...k：

集合演算結果の集合演算番号を，"LID(集合演算番号)"で表示します。

ll...l：

結合する問合せ式本体が問合せ指定の場合，"QID(問合せ ID)"を表示します。結合する問合せ式本体が複数の問合せ指定の結合結果の場合，"LID(集合演算番号)"を表示します。

mm...m{CC...C}：

集合演算をする前に，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

LIMIT 句指定時に，作業表を作成しないで，ソート結果の先頭から n 件だけをメモリ上で求める場合には，"MEM(SUBSORT)"を表示します。

nn...n：

集合演算の種別（"UNION"，"UNION ALL"，"EXCEPT"，又は"EXCEPT ALL"）を表示します。前後にある[**ll...l mm...m{CC...C}**]は，結合する問合せ式本体となります。

集合演算の演算順序の表示例を次に示します。

```
((SELECT C1 FROM T1   ..................問合せID：1
    UNION SELECT C1 FROM T2)   .........問合せID：2
      UNION ALL SELECT C1 FROM T3)   ....問合せID：3
```

```
SetOpe Process : LID(1) = [QID(1) LIST(SORT) {...}] UNION [QID(2) LIST(SORT) {...}]
                 LID(2) = [LID(1) LIST{...}] UNION ALL [QID(3) LIST{...}]
```

17.5.2　問合せ処理情報

```
----- QUERY ID : aa...a -----
Query Type     : bb...b cc...c(cc...c)  dd...d{AA...A},...
Sub Query Type : ee...e{BB...B}  [ff...f]
IfThenCnd      :{CC...C}
Transfer Type  : gg...g-CLM hh...h
Order by Mode  : ii...i
Having         :{DD...D}
Group by Mode  : jj...j
FLT Server     : kk...k(ll...l)  mm...m-CLM nn...n
```

[説明]

aa...a：

問合せ ID を表示します。

問合せ指定単位に番号を付けます。コンパイルした SQL が複数の問合せ指定で構成される場合，この行で区切って情報を出力します。

bb...b：

問合せの種類を表示します。

QUERY：

問合せ

DERIVED TABLE：

FROM 句に問合せ式本体を指定した場合の導出表の問合せ

SUBQUERY：

副問合せ

WITH：

WITH 句問合せ指定

VIEW：

ビュー定義の問合せ指定

DUMMY：

HiRDB が内部的に作成する問合せ

CONSTRAINT：

参照制約の制約条件内の副問合せ

cc...c(cc...c)：

問合せの種類によって，次のように表示します。

- WITH 句問合せ指定の場合は WITH 句問合せ名（相関名）となります。
- ビュー定義の問合せ指定の場合はビュー表名（相関名）となります。

- FROM 句に指定した導出表中の問合せ指定の場合は，"(NO NAME)"，又は"(NO NAME)(相関名)"を表示します。
- HiRDB が内部的に問合せを作成する場合は，その問合せ結果を一時的に格納するために，内部的に作成する作業表名"(DUMMY 作業表番号)"となります。なお，作業表番号は 3 けたの整数です。
- 参照制約の場合，制約条件内の副問合せのときには制約名となります。

dd...d{AA...A},...：

データを検索し，必要に応じて結合した結果に対して，GROUP BY，ORDER BY，DISTINCT，集合演算，排他処理などのために，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

作業表を作成する処理の組み合わせによって，複数の作業表を作成する場合はコンマで区切られます。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

LIMIT 句指定時に，作業表を作成しないで，ソート結果の先頭から n 件だけをメモリ上で求める場合には，"MEM(SUBSORT)"を表示します。

ee...e{BB...B}：

副問合せの実行方式を表示します。副問合せの実行方式については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

WORK TABLE ATS SUBQ：

副問合せの処理方式が作業表 ATS 実行の場合

WORK TABLE SUBQ：

副問合せの処理方式が作業表実行の場合

ROW VALUE SUBQ：

副問合せの処理方式が行値実行の場合

HASH SUBQ：

副問合せの処理方式がハッシュ実行の場合

NESTED LOOPS WORK TABLE SUBQ：

副問合せの処理方式がネストループ作業表実行の場合

NESTED LOOPS ROW VALUE SUBQ：

副問合せの処理方式がネストループ行値実行の場合

ff...f：

副問合せ実行方式の SQL 最適化指定を指定した場合，その指定が有効か無効かを表示します。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。

AS SPECIFIED：

有効です。

SPECIFICATION IGNORED：

無効です。

gg...g-CLM **hh...h**：

副問合せ，WITH 句問合せ指定，ビュー定義，集合演算，及び HiRDB が内部的に問合せを作成する場合，上位の問合せに結果を転送するときの転送方法を表示します。gg...g には転送に使用する列数，hh...h には転送方法の種類を表示します。転送方法の種類については，「転送方法の種類」を参照してください。

ii...i：

ORDER BY の処理方式を表示します。ORDER BY 句を指定していない場合でも，暗黙的に ORDER BY の処理をすることがあります。ORDER BY の処理方式については，「ORDER BY 処理方式の種類」を参照してください。

ORDER BY の処理をしない場合，この行を表示しません。

jj...j：

グループ分け処理（暗黙的グループ分け処理を含む）の処理方式を表示します。グループ分け処理方式については，「グループ分け処理方式の種類」を参照してください。

グループ分け処理をしない場合，この行を表示しません。

kk...k：

フロータブルサーバの要求数を表示します。

ll...l：

フロータブルサーバの使用目的を表示します。

検索実行時は，要求数を超えない範囲で，そのときに利用できるすべてのフロータブルサーバを使用します。HiRDB/シングルサーバの場合，この行を表示しません。

IMPLICIT GROUP BY：

明示的に GROUP BY を指定しない場合でも，集合関数処理のための内部的なグループ分け処理に使用します。

GROUP BY：

GROUP BY で指定したグループ分け処理のために使用します。

DISTINCT：

DISTINCT 処理のために使用します。

ORDER BY：

ORDER BY 処理をするために，フロータブルサーバを割り当てます。

FOR READ ONLY：

FOR READ ONLY 句の処理のために，フロータブルサーバを割り当てます。

FOR UPDATE：

FOR UPDATE 句の処理のために，フロータブルサーバを割り当てます。

FOR UPDATE OF：

FOR UPDATE OF 列名〔，列名〕...の処理のために，フロータブルサーバを割り当てます。

PRE-UPDATE DATA：

更新前データの参照処理のためにフロータブルサーバを割り当てます。

mm...m-CLM **nn...n**：

フロータブルサーバへの転送方法を表示します。mm...m には転送に使用する列数を，nn...n には転送方法の種類を表示します。転送方法の種類については，「転送方法の種類」を参照してください。

17.5.3　結合処理情報

(1)　HiRDB/シングルサーバで結合検索をする場合，又は HiRDB/パラレルサーバで結合検索（SELECT-APSL なし）をする場合

```
JOIN
 # Join ID     :aa...a
   L Join ID   : bb...b ff...f{BB...B}
   L Table     : cc...c(cc...c) ddddddddddd(dd...d) ff...f{BB...B} {AA...A}
   L SubQ ID   : ee...e ff...f{BB...B}
   R Join ID   : gg...g kk...k{DD...D}
   R Table     : hh...h(hh...h) iiiiiiiiii(ii...i) kk...k{DD...D} {CC...C}
   R SubQ ID   : jj...j kk...k{DD...D}
   R SubQEX ID : tt...t kk...k{DD...D}
   JoinCnd     :{EE...E}
   Join Type   :ll...l-CLM mm...m(nn...n) oo...o-CLM pp...p(qq...q)  [rr...r]
   FLT Server  :ss...s
   RowCnd      :{FF...F}
   RowCndAfter :{GG...G}
   IfThenCndOn :{HH...H}
   IfThenCnd   :{II...I}
```

[説明]

aa...a：

結合処理 ID を表示します。

結合処理単位で番号を付け，結合処理が複数ある場合にはこの行で区切られます。

bb...b：

左側の結合相手がほかの結合結果の場合，結合処理 ID を表示します。

結合相手がほかの結合結果でない場合は，この行を表示しません。

cc...c(cc...c)：

左側の結合相手が表の場合に，表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。また，表の種類に応じて，次の情報を表示します。

- WITH 句の導出表の場合は問合せ名
- ビュー表の場合はビュー表名（相関名）

- FROM 句に指定した導出表の場合は"(NO NAME)"，又は"(NO NAME)(相関名)"

結合相手が表でない場合は，この行を表示しません。

ddddddddddd(dd...d)：

左側の結合相手が表の場合に，表 ID を 16 進数（10 進数）で表示します。

結合相手が表でない場合は，この行を表示しません。

ee...e：

左側の結合相手が副問合せの場合に，副問合せの問合せ ID を表示します。

結合相手が副問合せでない場合は，この行を表示しません。

ff...f{BB...B}：

結合をする前に作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

gg...g：

右側の結合相手がほかの結合結果の場合に，結合処理 ID を表示します。

結合相手がほかの結合結果でない場合は，この行を表示しません。

hh...h(hh...h)：

右側の結合相手が表の場合に，表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。また，表の種類に応じて，次の情報を表示します。

- WITH 句の導出表の場合は問合せ名
- ビュー表の場合はビュー表名（相関名）
- FROM 句に指定した導出表の場合は"(NO NAME)"，又は"(NO NAME)(相関名)"

結合相手が表でない場合は，この行を表示しません。

iiiiiiiiii(ii...i)：

右側の結合相手が表の場合に，表 ID を 16 進数（10 進数）で表示します。

結合相手が表でない場合は，この行を表示しません。

jj...j：

右側の結合相手が集合演算を含まない副問合せの場合に，副問合せの問合せ ID を表示します。結合相手が集合演算を含まない副問合せでない場合，この行を表示しません。

kk...k{DD...D}：

結合をする前に，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

ll...l-CLM **mm...m(nn...n)：**

結合方法を表示します。

ll...l：

結合に使用する結合条件の数（列数）を表示します。

mm...m：

結合方法の種類を表示します。結合方法の種類については，「結合方法の種類」を参照してください。

nn...n：

結合処理の実行種別を表示します。

INNER：

内結合をします。探索条件を評価し，真となる場合に結合結果を返します。

LEFT OUTER[(R)]：

左外結合をします。SQL に RIGHT [OUTER] JOIN を指定した場合，(R)を表示します（LEFT [OUTER] JOIN を指定した場合は表示しません）。このとき，SQL 最適化処理が右外結合を等価な左外結合に変換して，左外結合をします。

ON 探索条件に指定した条件を評価し，真となる場合に結合結果を返します。外表の行に対して，内表が 1 行も真にならない場合は，内表の列に NULL 値を埋めて結合結果を返します。

EXIST：

探索条件を評価し，外表の行に対して内表のうち少なくとも 1 行が真となる場合に，外表の行を返します。

NOT EXIST：

探索条件を評価し，外表の行に対して内表が 1 行も真とならない場合に，外表の行を返します。

ALL：

探索条件を評価し，外表の行に対して内表のすべての行が真となる場合に，外表の行を返します。

VALUE：

探索条件を評価し，外表の行に対して内表のうち 1 行だけが真となる場合に，外表の行を返します。2 行以上が真となる場合にはエラーとなります。

oo...o-CLM **pp...p(qq...q)：**

結合方法をするサーバへの転送方法を表示します。

oo...o：

転送に使用する列数を表示します。

pp...p：

転送方法の種類を表示します。転送方法の種類については，「転送方法の種類」を参照してください。

qq...q：

転送方向を表示します。

FROM L TO R：

左側を右側に転送して結合します。

FROM R TO L：

右側を左側に転送して結合します。

TO FLT：

左側，右側の両方を，フロータブルサーバに転送して結合します。

rr...r：

結合方式の SQL 最適化指定を指定した場合，その指定が有効か無効かを表示します。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。

AS SPECIFIED：

有効です。

SPECIFICATION IGNORED：

無効です。

ss...s：

HiRDB/パラレルサーバでソート処理に使うフロータブルサーバの要求数を表示します。

検索実行時は，要求数を超えない範囲で，そのときに利用できるすべてのフロータブルサーバを使用します。結合方法が"NESTED LOOPS JOIN"の場合は 0 となります。HiRDB/シングルサーバの場合は，この行を表示しません。

tt...t：

右側の結合相手が集合演算を含む副問合せの場合に，副問合せの問合せ ID を表示します。

結合相手が集合演算を含む副問合せでない場合は，この行を表示しません。

(2) HiRDB/パラレルサーバで結合検索（SELECT-APSL あり）をする場合

```
JOIN
 # Join ID     :aa...a
   L Join ID   : bb...b ee...e{AA...A}
   L Table     : cc...c(cc...c) ddddddddddd(dd...d)  ee...e{AA...A}
   R Join ID   : ff...f  ii...i{BB...B}
   R Table     : gg...g(gg...g) hhhhhhhhhh(hh...h)  ii...i{BB...B}
   JoinCnd     :{CC...C}
   Join Type   :SELECT-APSL
         Table Name   :jj...j(jj...j) kkkkkkkkkk(kk...k)
         Column ID    :llllllllll(ll...l)
         Predicate    :mm...m
         Threshold    :nn...n
      [1] oo...o-CLM pp...p(qq...q)  rr...r-CLM ss...s(tt...t) ........1
          FLT Server  :uu...u
          IfThenCnd   :{DD...D}
      [2] oo...o-CLM pp...p(qq...q)  rr...r-CLM ss...s(tt...t) ........2
          FLT Server  :uu...u
          IfThenCnd   :{DD...D}
```

［説明］

1 は結合方法の第一候補，2 は結合方法の第二候補を示しています。

aa...a：

結合処理 ID を表示します。

結合処理単位で番号を付け，結合処理が複数ある場合にはこの行で区切って情報を表示します。

bb...b：

左側の結合相手がほかの結合結果の場合に，結合処理 ID を表示します。

結合相手がほかの結合結果でない場合は，この行を表示しません。

cc...c(cc...c)：

左側の結合相手が表の場合に，表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。また，表の種類に応じて，次の情報を表示します。

- WITH 句の導出表の場合は問合せ名
- ビュー表の場合はビュー表名（相関名）
- FROM 句に指定した導出表の場合は"(NO NAME)"，又は"(NO NAME)(相関名)"

結合相手が表でない場合は，この行を表示しません。

ddddddddddd(dd...d)：

左側の結合相手が表の場合に，表 ID を 16 進数（10 進数）で表示します。

結合相手が表でない場合は，この行を表示しません。

ee...e{AA...A}：

結合をする前に，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

ff...f：

右側の結合相手がほかの結合結果の場合に，結合処理 ID を表示します。

結合相手がほかの結合結果でない場合は，この行を表示しません。

gg...g(gg...g)：

右側の結合相手が表の場合に，表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。また，表の種類に応じて，次の情報を表示します。

- WITH 句の導出表の場合は問合せ名
- ビュー表の場合はビュー表名（相関名）
- FROM 句に指定した導出表の場合は"(NO NAME)"，又は"(NO NAME)(相関名)"

結合相手が表でない場合は，この行を表示しません。

hhhhhhhhhh(hh...h)：

右側の結合相手が表の場合に，表 ID を 16 進数（10 進数）で表示します。

結合相手が表でない場合は，この行を表示しません。

ii...i{BB...B}：

結合をする前に，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

jj...j(jj...j)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表名（相関名）を表示します。

kkkkkkkkkk(kk...k)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表 ID を 16 進数（10 進数）で表示します。

llllllllll(ll...l)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列 ID を 16 進数（10 進数）で表示します。

mm...m：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる条件の述語を表示します。

nn...n：

SQL オブジェクトを選択するための基準値となるヒット率を表示します。

実行時に算出したヒット率がこの値より小さい場合は，第一候補の SQL オブジェクトを使用します。
実行時に算出したヒット率がこの値と等しい又は大きい場合は，第二候補の SQL オブジェクトを使用します。

oo...o-CLM **pp...p(qq...q)：**

結合方法を表示します。

oo...o：

結合に使用する結合条件の数（列数）を表示します。

pp...p：

結合方法の種類を表示します。結合方法の種類については，「結合方法の種類」を参照してください。

qq...q：

結合処理の実行種別を表示します。

INNER：

内結合をします。探索条件を評価し，真となる場合に結合結果を返します。

rr...r-CLM **ss...s(tt...t)：**

結合処理をするサーバへの転送方法を表示します。

rr...r：

転送に使用する列数を表示します。

ss...s：

転送方法の種類を表示します。転送方法の種類については，「転送方法の種類」を参照してください。

tt...t：

転送方向を表示します。

FROM L TO R：

左側を右側に転送して結合します。

FROM R TO L：

右側を左側に転送して結合します。

TO FLT：

左側，右側の両方を，フロータブルサーバに転送して結合します。

uu...u：

HiRDB/パラレルサーバで，ソート処理に使用するフロータブルサーバ要求数を表示します。

検索実行時は，要求数を超えない範囲で，そのときに利用できるすべてのフロータブルサーバを使用します。結合方法が"NESTED LOOPS JOIN"の場合は 0 となります。HiRDB/シングルサーバの場合，この行を表示しません。

17.5.4　実表検索処理情報

(1)　HiRDB/シングルサーバで実表検索をする場合，又は HiRDB/パラレルサーバの実表検索（SELECT-APSL なし）で，インデクスを使用しないか，一つだけ使用した検索をする場合

```
SCAN
 # Table Name  :aa...a(aa...a) bbbbbbbbbb(bb...b) (xx...x) cc...c{AA...A}
   Cost        :d(ee...eROW) {T-BB...B,I-CC...C,P-DD...D,AND-EE...E,OR-FF...F}
   RDAREA      :ff...f-CLM gg...g(hh...h) (ii...iRD/jj...jBES) [kkkk(kk...k),...] ll...l
   Rebalance   :mm...m
   Scan Type   :nn...n(oo...o)  [pp...p]
   Index Name  :qq...q rrrrrrrrrr(rr...r) (ss...s)tt...t
                     SearchCnd :uu...u[vv...v],...
                     KeyCnd    :ww...w
   RowCnd      :{HH...H}
   IfThenCnd   :{II...I}
```

[説明]

一時表に対する検索処理情報の場合，"RDAREA"の行を表示しません。

VALUES を指定した INSERT 文を実行した場合，"Scan Type"の行を表示しません。

aa...a(aa...a)：

検索対象となる表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。検索処理が複数ある場合は，この行で区切って情報を表示します。

bbbbbbbbbb(bb...b)：

検索対象となる表 ID を 16 進数（10 進数）で表示します。

cc...c{AA...A}：

実表からデータを検索した後に，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

d：

最適化情報収集ユティリティ（pdgetcst）による最適化情報の取得の有無を表示します。

Y：最適化情報を取得しています。

N：最適化情報を取得していません。

ee...e：

HiRDB がアクセスパスを決定するために使用する表の行数を表示します。

デフォルトの行数は，SQL 最適化モードによって異なります。

- コストベース最適化モード 1，又は HiRDB/シングルサーバでコストベース最適化モード 2 の場合
  1000 万行
- HiRDB/パラレルサーバでコストベース最適化モード 2 の場合
  1000 万行×表が格納された RD エリアがあるバックエンドサーバ数

最適化情報収集ユティリティ（pdgetcst）を実行すると，HiRDB がアクセスパスを決定するために使用する表の行数を変更できます。

ff...f：

表の分割列数を表示します。

gg...g：

表分割の種別を表示します。

NON DIVISION：非分割

KEY RANGE：キーレンジ分割（格納条件指定）

PARTITION：キーレンジ分割（境界値指定）

MULTIDIM PARTITION：マトリクス分割（境界値指定）

MULTIDIM：ハッシュ分割を含むマトリクス分割

FLEXIBLE HASH：リバランス機能を使用しないフレキシブルハッシュ分割

FIX HASH：リバランス機能を使用しない FIX ハッシュ分割

RB FLEXIBLE HASH：リバランス機能を使用するフレキシブルハッシュ分割

RB FIX HASH：リバランス機能を使用する FIX ハッシュ分割

hh...h：

ハッシュ分割時に，ハッシュ関数名を表示します。ハッシュ分割時以外は表示しません。また，ハッシュ分割を含むマトリクス分割の場合も表示しません。

ii...iRD/**jj...j**BES：

ii...i には表の分割数（RD エリア数）が，jj...j には表が格納された RD エリアがあるバックエンドサーバ数を表示します。

非横分割表の場合，ii...i は 1 となります。また，HiRDB/シングルサーバの場合，"/jj...jBES"を表示しません。

kkkk(kk...k),...：

表格納用 RD エリアの RD エリア ID を 16 進数（10 進数）で表示します。

表分割をしている場合は分割数分の RD エリア ID を表示します。RD エリア ID に対応する RD エリア名は，pddbls コマンドで調べることができます。

ll...l：

ALL：

表を定義しているすべての RD エリアを検索する場合に表示します。

RESTRICTED：

表を定義している RD エリアの内，探索条件によって該当データがある可能性のある RD エリアだけを検索する場合に表示します。

mm...m：

リバランス機能を使用する表の場合，リバランス状態を表示します。

リバランス機能を使用しない表の場合，この行を表示しません。リバランス機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

NORMAL：

通常の状態，又は ALTER TABLE で RD エリアを追加しただけでリバランスユティリティ（pdrbal）を実行していない状態です。

ON REBALANCE：

表がリバランス中（リバランス開始からリバランス終了まで）の状態です。

nn...n：

検索方法を表示します。

検索方法については，「検索方法の種類」を参照してください。

oo...o：

Group by Mode が IMPLICIT MIN-MAX INDEX の場合に，インデクスの検索方法が示されます。

MIN：

インデクスの検索する範囲を昇順にサーチして，探索条件を満たす行があった時点でインデクスの検索を打ち切ります。

MAX：

インデクスの検索する範囲を降順にサーチして，探索条件を満たす行があった時点でインデクスの検索を打ち切ります。

MIN,MAX：

まず，MIN の値を求めるためにインデクスを昇順にサーチして，次に MAX の値を求めるためにインデクスを降順にサーチします。

集合関数に MIN だけ，又は MAX だけを指定した場合に比べて，検索時間は約 2 倍になります。

MIN と MAX では，インデクスのサーチする順序が逆になるため，サーチ条件のインデクスのサーチ範囲も逆になりますが，MIN と MAX を混在している場合には MIN 用のサーチ範囲だけ表示します。

pp...p：

使用インデクスの SQL 最適化指定を指定した場合，その指定が有効か無効かを表示します。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。

AS SPECIFIED：

有効です。

SPECIFICATION IGNORED：

無効です。

PARTIALLY IGNORED：

一部の指定は有効ですが，それ以外の指定は無効です。

qq...q：

検索で使用するインデクス名を表示します。

インデクスを使用しないで検索する場合には，この行を表示しません。

rrrrrrrrrr(rr...r)：

検索で使用するインデクス ID を 16 進数（10 進数）で表示します。

ss...s：

インデクス属性を組み合わせた情報を表示します。

数字：インデクス構成列数

G：プラグインインデクス

U：ユニークキーインデクス

C：クラスタキーインデクス

D：分割キーインデクス又はプラグインインデクス

d：サーバ内分割された非分割キーインデクス

E：除外キー値を指定したインデクス

M：繰返し列を含むインデクス

P：主キーインデクス

tt...t：

[プラグインインデクス及び部分構造インデクス以外のインデクスの場合]

インデクス構成列情報を列ごとにコンマ(,)で区切って，全体を（ ）で囲んで表示します。

インデクス構成列情報は，構成列のサーチ方向とインデクス構成列名で表示します。

構成列のサーチ方向は，構成列を昇順にサーチする場合には＋，降順にサーチする場合には−を表示します。

```
例）　第1構成列(C1)は昇順，第2構成列(C2)は降順にサーチする場合
      (+C1,-C2)
```

[部分構造インデクスの場合]

インデクスを構成するインデクス構成情報を，全体を（）で囲んで表示します。インデクス構成情報全体には次の情報を表示します。

1. XML 型が定義された列の列名
2. 部分構造パスのサーチ方向

   部分構造パスのサーチ方向は，部分構造パスを昇順にサーチする場合には＋，降順にサーチする場合には−を表示します。
3. 部分構造インデクスを構成する部分構造パス
4. 部分構造インデクスを構成する部分構造パスの数
5. CREATE INDEX への USING UNIQUE TAG の指定有無

   USING UNIQUE TAG の指定がある場合は U を表示します。

これらの情報は，2.〜5.の全体を［］で囲んで表示します。その中でも 4.と 5.は（）で囲んで表示します。

```
例1)　XML型の列(C1)のUSING UNIQUE TAGを指定した部分構造パス(/A/B/C)を昇順にサーチした場合(XML名前空間宣言の指定なし)
      (C1[(1U)+/A/B/C])

例2)　XML型列(C1)のUSING UNIQUE TAGを指定した部分構造パス(/A/B/C)を昇順にサーチした場合(XML名前空間宣言に指定した接頭辞を使用)
      (C1[(1U)+declare namespace aaa
        =" http://www.w3.org/XML/1998/namespace" ;/A/B/aaa:C])

例3)　XML型列(C1)のUSING UNIQUE TAGを指定した部分構造パス(/A/B/C)を昇順にサーチした場合(HiRDBで定義されている接頭辞(xml)を使用)
      (C1[(1U)+/A/B/xml:C])
```

[プラグインインデクスの場合]

プラグイン提供関数の種別によって，詳細な情報を表示します。プラグイン提供関数の種別については，「プラグイン提供関数の種別」を参照してください。関数呼出しの第 2 引数以降は表示しません。

```
例1)　種別がSCAN TYPEとなるプラグイン提供関数(WITHIN)で第1構成列がC1の場合
      WITHIN(C1,...)[SCAN TYPE]

例2)　種別がINDEX SCAN TYPEとなるプラグイン提供関数(CONTAINS)で第1構成列がC1の場合
      CONTAINS(C1,...)[INDEX SCAN TYPE]

例3)　種別がFULL SCAN TYPEとなるプラグイン提供関数(SEARCHFEATUREDATA)で第1構成列がC1の場合
      SEARCHFEATUREDATA(C1,...)[FULL SCAN TYPE]
```

uu...u[vv...v],...：

サーチ条件の種類，サーチ条件の絞り込み範囲を表示します。サーチ条件については，「サーチ条件」を参照してください。

インデクスを使用した検索でサーチ条件がない場合，インデクスの全範囲検索をするときは"NONE(FULL SCAN)"を表示します。Group by mode が IMPLICIT MIN-MAX INDEX の場合，"NONE"を表示します（探索条件が真になった時点で検索を打ち切り，インデクスの全範囲検索になるとは限らないため，"(FULL SCAN)"は表示しません）。

サーチ条件によって，絞り込み範囲の数が 0 個となる場合，又は 1 個以上となる場合があります。

インデクスの第 2 構成列以降にサーチ条件があるが，第 1 構成列にはサーチ条件がない場合には，サーチ条件の絞り込み範囲の後ろに"(FULL SCAN)"を表示します。

ww...w：

キー条件を表示します。キー条件については，「キー条件」を参照してください。

キー条件がない場合は，この行を表示しません。

xx...x：

検索対象が共用表の場合は"SHARED"を，一時表の場合は"TEMPORARY"を表示します。共用表，又は一時表でない場合，表示しません。

(2) HiRDB/シングルサーバで実表検索，又は HiRDB/パラレルサーバで実表検索（SELECT-APSL なし）で，インデクスを二つ以上使用した検索をする場合

```
SCAN
 # Table Name   :aa...a(aa...a) bbbbbbbbbb(bb...b) (CC...C) cc...c{DD...DD}
   Cost         :d (ee...eROW)  {T-EE...E,I-FF...F,P-GG...G,AND-HH...H,OR-II...I}
   RDAREA       :ff...f-CLM gg...g(hh...h) (ii...iRD/jj...jBES) [kkkk(kk...k),...] LL...L
   Rebalance    :mm...m
   Scan Type    :nn...n  [oo...o]
   Index Name   :pp...p = qq...q rrrrrrrrrr(rr...r) (ss...s) tt...t BB...B{KK...K}
                         Scan Type :uu...u
                         SearchCnd :vv...v[ww...w],...
                         KeyCnd    :xx...x
                         RowCnd    :{HH...H}
                         IfThenCnd :{NN...N}
                  pp...p = qq...q rrrrrrrrrr(rr...r) (ss...s) tt...t BB...B{KK...K}
                         Scan Type :uu...u
                         SearchCnd :vv...v[ww...w],...
                         KeyCnd    :xx...x
                         RowCnd    :{HH...H}
                         IfThenCnd :{NN...N}
                  pp...p = [yy...y zz...z{JJ...J}] AA...A[yy...y zz...z{JJ...J}]
   RowCnd       :{MM...M}
   IfThenCnd    :{OO...O}
```

[説明]

一時表に対する検索処理情報の場合，"RDAREA"の行を表示しません。

aa...a(aa...a)：

検索対象となる表名（相関名）を表示します。

相関名を使用していない場合，（相関名）を表示しません。検索処理が複数ある場合は，この行で区切って情報を表示します。

bbbbbbbbbb(bb...b)：

検索対象となる表 ID を 16 進数（10 進数）で表示します。

cc...c{DD...D}：

実表からデータを検索した後に，作業表を作成する場合に情報を表示します。作業表を作成しない場合は表示しません。

ソートをしないで作業表を作成する場合は，"LIST"を表示します。ソートをして作業表を作成する場合は，"LIST(SORT)"を表示します。

d：

最適化情報収集ユティリティ（pdgetcst）による最適化情報の取得の有無を表示します。

Y：最適化情報を取得しています。

N：最適化情報を取得していません。

ee...e：

HiRDB がアクセスパスを決定するために使用する表の行数を表示します。

デフォルトの行数は，SQL 最適化モードによって異なります。

- コストベース最適化モード 1，又は HiRDB/シングルサーバでコストベース最適化モード 2 の場合
  1000 万行
- HiRDB/パラレルサーバでコストベース最適化モード 2 の場合
  1000 万行×表が格納された RD エリアがあるバックエンドサーバ数

最適化情報収集ユティリティ（pdgetcst）を実行すると，HiRDB がアクセスパスを決定するために使用する表の行数を変更できます。

ff...f：

表の分割列数を表示します。

gg...g：

表分割の種別を表示します。

NON DIVISION：非分割

KEY RANGE：キーレンジ分割（格納条件指定）

PARTITION：キーレンジ分割（境界値指定）

MULTIDIM PARTITION：マトリクス分割（境界値指定）

MULTIDIM：ハッシュ分割を含むマトリクス分割

FLEXIBLE HASH：リバランス機能を使用しないフレキシブルハッシュ分割

FIX HASH：リバランス機能を使用しない FIX ハッシュ分割

RB FLEXIBLE HASH：リバランス機能を使用するフレキシブルハッシュ分割

RB FIX HASH：リバランス機能を使用する FIX ハッシュ分割

hh...h：

ハッシュ分割時にハッシュ関数名を表示します。ハッシュ分割時以外は表示しません。また，ハッシュ分割を含むマトリクス分割の場合も表示しません。

ii...iRD/**jj...j**BES：

ii...i には表の分割数（RD エリア数）が，jj...j には表が格納された RD エリアがあるバックエンドサーバ数を表示します。

非横分割表の場合，ii...i は 1 となります。また，HiRDB/シングルサーバの場合，"/jj...jBES"を表示しません。

kkkk(kk...k),...：

表格納用 RD エリアの RD エリア ID を 16 進数（10 進数）で表示します。

表分割をしている場合は分割数分の RD エリア ID を表示します。RD エリア ID に対応する RD エリア名は，pddbls コマンドで調べられます。

ll...l：

ALL：

表を定義しているすべての RD エリアを検索する場合に表示します。

RESTRICTED：

表を定義している RD エリアのうち，探索条件によって該当するデータがある可能性のある RD エリアだけ検索する場合に表示します。

mm...m：

リバランス機能を使用する表の場合，リバランス状態を表示します。

リバランス機能を使用しない表の場合，この行を表示しません。リバランス機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

NORMAL：

通常の状態，又は ALTER TABLE で RD エリアを追加しただけでリバランスユティリティ（pdrbal）を実行していない状態です。

ON REBALANCE：

表がリバランス中（リバランス開始からリバランス終了まで）の状態です。

nn...n：

検索方法を表示します。

検索方法については，「検索方法の種類」を参照してください。

oo...o：

使用インデクスの SQL 最適化指定を指定した場合，その指定が有効か無効かを表示します。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。

AS SPECIFIED：

有効です。

SPECIFICATION IGNORED：

無効です。

PARTIALLY IGNORED：

一部の指定は有効ですが，それ以外の指定は無効です。

pp...p：

AND PLURAL INDEXES SCAN のときに作成する作業表番号を，"LID(作業表番号)"で表示します。

qq...q：

AND PLURAL INDEXES SCAN，又は OR PLURAL INDEXES SCAN の場合に，作業表を作成するために使用するインデクス名を複数行表示します。ただし，インデクスを使用しないで作成する作業表については，インデクス名に"(NO USE)"を表示します。

rrrrrrrrrr(rr...r)：

検索で使用するインデクス ID を 16 進数（10 進数）で表示します。

ss...s：

次のインデクス属性を組み合わせた情報を表示します。

数字：インデクス構成列数

G：プラグインインデクス

U：ユニークキーインデクス

C：クラスタキーインデクス

D：分割キーインデクス又はプラグインインデクス

d：サーバ内分割された非分割キーインデクス

E：除外キー値を指定したインデクス

M：繰返し列を含むインデクス

P：主キーインデクス

tt...t：

[プラグインインデクス及び部分構造インデクス以外のインデクスの場合]

インデクス構成列情報を列ごとにコンマ(,)で区切って，全体を（）で囲んで表示します。

インデクス構成列情報は，構成列のサーチ方向とインデクス構成列名で表示します。

構成列のサーチ方向は，構成列を昇順にサーチする場合には＋，降順にサーチする場合には−を表示します。

```
例）　第1構成列(C1)は昇順，第2構成列(C2)は降順にサーチする場合
      (+C1,-C2)
```

[部分構造インデクスの場合]

インデクスを構成するインデクス構成情報を，全体を（）で囲んで表示します。インデクス構成情報全体には次の情報を表示します。

1. XML 型が定義された列の列名
2. 部分構造パスのサーチ方向

部分構造パスのサーチ方向は，部分構造パスを昇順にサーチする場合には+，降順にサーチする場合には-を表示します。

3. 部分構造インデクスを構成する部分構造パス
4. 部分構造インデクスを構成する部分構造パスの数
5. CREATE INDEX への USING UNIQUE TAG の指定有無

   USING UNIQUE TAG の指定がある場合は U を表示します。

これらの情報は，2.～5.の全体を［］で囲んで表示します。その中でも 4.と 5.は（）で囲んで表示します。

```
例1)　XML型の列(C1)のUSING UNIQUE TAGを指定した部分構造パス(/A/B/C)を昇順にサーチした場合(XML名前空間宣言の指定なし)
      (C1[(1U)+/A/B/C])

例2)　XML型列(C1)のUSING UNIQUE TAGを指定した部分構造パス(/A/B/C)を昇順にサーチした場合(XML名前空間宣言に指定した接頭辞を使用)
      (C1[(1U)+declare namespace aaa
        =" http://www.w3.org/XML/1998/namespace" ;/A/B/aaa:C])

例3)　XML型列(C1)のUSING UNIQUE TAGを指定した部分構造パス(/A/B/C)を昇順にサーチした場合(HiRDBで定義されている接頭辞(xml)を使用)
      (C1[(1U)+/A/B/xml:C])
```

[プラグインインデクスの場合]

プラグイン提供関数の種別によって，詳細な情報を表示します。プラグイン提供関数の種別については，「プラグイン提供関数の種別」を参照してください。関数呼出しの第 2 引数以降は表示しません。

```
例1)　種別がSCAN TYPEとなるプラグイン提供関数(WITHIN)で第1構成列がC1の場合
      WITHIN(C1,...)[SCAN TYPE]

例2)　種別がINDEX SCAN TYPEとなるプラグイン提供関数(CONTAINS)で第1構成列がC1の場合
      CONTAINS(C1,...)[INDEX SCAN TYPE]

例3)　種別がFULL SCAN TYPEとなるプラグイン提供関数(SEARCHFEATUREDATA)で第1構成列がC1の場合
      SEARCHFEATUREDATA(C1,...)[FULL SCAN TYPE]
```

uu...u：

それぞれの作業表を作成するための検索方法を表示します。

検索方法については，「検索方法の種類」を参照してください。

vv...v [ww...w],...：

サーチ条件の種類，及びサーチ条件の絞り込み範囲を表示します。サーチ条件については，「サーチ条件」を参照してください。

インデクスを使用した検索でサーチ条件がない場合，サーチ条件の種類には"NONE(FULL SCAN)"を表示します。

サーチ条件によって，絞り込み範囲の数が 0 個となる場合，又は 1 個以上となる場合があります。

インデクスの第 2 構成列以降にサーチ条件があるが，第 1 構成列にはサーチ条件がない場合には，サーチ条件の絞り込み範囲の後ろに"(FULL SCAN)"を表示します。

xx...x：

キー条件を表示します。キー条件については，「キー条件」を参照してください。

キー条件がない場合は，この行を表示しません。

pp...p = [yy...y zz...z{JJ...J}] AA...A [yy...y zz...z{JJ...J}] ：

AND PLURAL INDEXES SCAN 時の作業表の作成順序を表示します。インデクスを三つ以上使用して検索する場合には，複数行に分けて表示します。

pp...p には，作業表番号が"LID(作業表番号)"を表示します。

AA...A には，作業表間の演算の種別｛"AND"，"OR"，又は"ANDNOT"｝を表示します。

yy...y には，演算をするための入力となる作業表を"LID(作業表番号)"で表示します。

zz...z には，ソートをしないで作業表を作成する場合は"LIST"が，ソートを使用して作業表を作成する場合は"LIST(SORT)"を表示します。

例：インデクス IX1(C1)及び IX2(C2)が定義されている場合に，次の条件の SQL を実行したものとします。

```
where C1='A'                    ←作業表番号：1
  or C2=between 'a' and 'z'     ←作業表番号：2
```

ScanType が AND PLURAL INDEXES SCAN の場合を例にすると，出力結果は次のようになります。

```
      :
    Scan Type     :AND PLURAL INDEXES SCAN
    Index Name    :LID(1)=IX1 (1) (+C1)
                           Scan Type :INDEX SCAN
                           SearchCnd :AT['A']
                   LID(2)=IX2 (1) (+C2)
                           Scan Type :INDEX SCAN
                           SearchCnd :RANGE(CS-CE) ['a','z']
                   LID(3)=[LID(1) LIST(SORT){...}] OR [LID(2) LIST(SORT){...}]
```

BB...B{KK...K}：

AND PLURAL INDEXES SCAN のための作業表とは別に，表の検索後に作業表を作成する場合に情報を表示します。

ソートを伴わないで作業表を作成する場合，II...I に"LIST"を表示します。作業表を作成する場合にソートをするときは，II...I に"LIST(SORT)"を表示します。作業表を作成しない場合，情報を表示しません。

CC...C：

検索対象が共用表の場合は"SHARED"を，一時表の場合は"TEMPORARY"を表示します。共用表，又は一時表でない場合，表示しません。

(3) HiRDB/パラレルサーバで実表検索（SELECT-APSL あり）をする場合

```
SCAN
 # Table Name   :aa...a(aa...a) bbbbbbbbbb(bb...b) (zz...z)
   Cost         :c (dd...dROW) {T-AA...A,I-BB...B,P-CC...C,AND-DD...D,OR-EE...E}
   RDAREA       :ee...e-CLM ff...f(gg...g) (hh...hRD/ii...iBES) [jjjj(jj...j),...] kk...k
```

```
    Rebalance    :ll...l
    Scan Type    :SELECT-APSL
          Table Name   :mm...m(mm...m) nnnnnnnnnn(nn...n)
          Column ID    :oooooo(oo...o)
          Predicate    :pp...p
          Threshold    :qq...q
       [1] rr...r  ..............................................1
            Index Name :ss...s ttttttttttt(tt...t) (uu...u) vv...v
                           SearchCnd :ww...w[xx...x],...
                           KeyCnd    :yy...y
                           RowCnd    :{FF...F}
       [2] rr...r  ..............................................2
            Index Name :ss...s ttttttttttt(tt...t) (uu...u) vv...v
                           SearchCnd :ww...w[xx...x],...
                           KeyCnd    :yy...y
                           RowCnd    :{FF...F}
    IfThenCnd    :{GG...G}
```

[説明]

1 は実表の検索方法の第一候補，2 は実表の検索方法の第二候補を示しています。検索方法については，「検索方法の種類」を参照してください。

aa...a(aa...a)：

検索対象となる表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。検索処理が複数ある場合は，この行で区切って情報を表示します。

bbbbbbbbbb(bb...b)：

検索対象となる表 ID を 16 進数（10 進数）で表示します。

c　：

最適化情報収集ユティリティ（pdgetcst）による最適化情報の取得の有無を表示します。

Y：最適化情報を取得しています。

N：最適化情報を取得していません。

dd...d：

HiRDB がアクセスパスを決定するために使用する表の行数を表示します。

デフォルトの行数は，SQL 最適化モードによって異なります。

- コストベース最適化モード 1，又は HiRDB/シングルサーバでコストベース最適化モード 2 の場合
  1000 万行
- HiRDB/パラレルサーバでコストベース最適化モード 2 の場合
  1000 万行×表が格納された RD エリアがあるバックエンドサーバ数

最適化情報収集ユティリティ（pdgetcst）を実行すると，HiRDB がアクセスパスを決定するために使用する表の行数を変更できます。

ee...e：

表の分割列数を表示します。

ff...f：

表分割の種別を表示します。

NON DIVISION：非分割

KEY RANGE：キーレンジ分割（格納条件指定）

PARTITION：キーレンジ分割（境界値指定）

MULTIDIM PARTITION：マトリクス分割（境界値指定）

MULTIDIM：ハッシュ分割を含むマトリクス分割

FLEXIBLE HASH：リバランス機能を使用しないフレキシブルハッシュ分割

FIX HASH：リバランス機能を使用しない FIX ハッシュ分割

RB FLEXIBLE HASH：リバランス機能を使用するフレキシブルハッシュ分割

RB FIX HASH：リバランス機能を使用する FIX ハッシュ分割

gg...g：

ハッシュ分割時に，ハッシュ関数名を表示します。ハッシュ分割時以外は表示しません。また，ハッシュ分割を含むマトリクス分割の場合も表示しません。

hh...hRD/**ii...i**BES：

hh...h には表の分割数（RD エリア数）が，ii...i には表が格納された RD エリアがあるバックエンドサーバ数を表示します。

非横分割表の場合，hh...h は 1 となります。また，HiRDB/シングルサーバの場合，"/ii...iBES"を表示しません。

jjjj(jj...j),...：

表格納用 RD エリアの RD エリア ID を 16 進数（10 進数）で表示します。

表分割をしている場合は分割数分の RD エリア ID を表示します。RD エリア ID に対応する RD エリア名は，pddbls コマンドで調べることができます。

kk...k：

ALL：

表を定義しているすべての RD エリアを検索する場合に表示します。

RESTRICTED：

表を定義している RD エリアの内，探索条件によって該当データがある可能性のある RD エリアだけを検索する場合に表示します。

ll...l：

リバランス機能を使用する表の場合，リバランス状態を表示します。

リバランス機能を使用しない表の場合，この行を表示しません。リバランス機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

NORMAL：

通常の状態，又は ALTER TABLE で RD エリアを追加しただけでリバランスユティリティ（pdrbal）を実行していない状態です。

ON REBALANCE：

表がリバランス中（リバランス開始からリバランス終了まで）の状態です。

mm...m(mm...m)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表名（相関名）を表示します。

nnnnnnnnnn(nn...n)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表 ID を，16 進数（10 進数）で表示します。

oooooo(oo...o)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列 ID を，16 進数（10 進数）で表示します。

pp...p：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる条件の述語を表示します。

qq...q：

SQL オブジェクトを選択するための基準値となるヒット率を表示します。

実行時に算出したヒット率がこの値より小さい場合は，第一候補の SQL オブジェクトを使用します。
実行時に算出したヒット率がこの値と等しい又は大きい場合は，第二候補の SQL オブジェクトを使用します。

rr...r：

検索方法を表示します。

検索方法については，「検索方法の種類」を参照してください。

ss...s：

検索で使用するインデクス名を表示します。

インデクスを使用しない検索の場合，この行を表示しません。

tttttttttt(tt...t)：

検索で使用するインデクス ID を 16 進数（10 進数）で表示します。

uu...u：

インデクス属性を組み合わせた情報を表示します。

数字：インデクス構成列数

G：プラグインインデクス

U：ユニークキーインデクス

C：クラスタキーインデクス

D：分割キーインデクス又はプラグインインデクス

d：サーバ内分割された非分割キーインデクス

E：除外キー値を指定したインデクス

M：繰返し列を含むインデクス

P：主キーインデクス

vv...v：

インデクス構成列情報を列ごとにコンマ(,)で区切って，全体を（）で囲んで表示します。

インデクス構成列情報は，構成列のサーチ方向とインデクス構成列名で表示します。

構成列のサーチ方向は，構成列を昇順にサーチする場合には＋，降順にサーチする場合には−を表示します。

例）　第1構成列(C1)は昇順，第2構成列(C2)は降順にサーチする場合
(+C1,-C2)

ww...w [xx...x],...：

サーチ条件の種類，及びサーチ条件の絞り込み範囲を表示します。サーチ条件については，「サーチ条件」を参照してください。

インデクスを使用した検索でサーチ条件がない場合，サーチ条件の種類には"NONE(FULL SCAN)"を表示します。

サーチ条件によって，絞り込み範囲の数が 0 個となる場合，1 個以上となる場合があります。

インデクスの第 2 構成列以降にサーチ条件があるが，第 1 構成列にはサーチ条件がない場合には，サーチ条件の絞り込み範囲の後ろに"(FULL SCAN)"を表示します。

yy...y：

キー条件を表示します。キー条件については，「キー条件」を参照してください。

キー条件がない場合は，この行を表示しません。

zz...z：

共用表が検索対象の場合，"SHARED"を表示します。共用表でない場合，表示しません。

(4)　HiRDB/パラレルサーバで表検索（SELECT-APSL あり）と結合処理（SELECT-APSL あり）を組み合わせた場合

```
SCAN
 # Table Name   :aa...a(aa...a) bbbbbbbbbb(bb...b) (AA...A)
   Cost         :c (dd...dROW) {T-BB...B,I-CC...C,P-DD...D,AND-EE...E,OR-FF...F}
   RDAREA       :ee...e-CLM ff...f(gg...g) (hh...hRD/ii...iBES) [jjjj(jj...j),...] kk...k
   Rebalance    :ll...l
   Scan Type    :SELECT-APSL
         Table Name   :mm...m(mm...m) nnnnnnnnnn(nn...n)
         Column ID    :oooooo(oo...o)
         Predicate    :pp...p
         Threshold    :qq...q
      [1] rr...r (ss...s)  .....................................1
           Index Name :tt...t uuuuuuuuuu(uu...u) (vv...v) ww...w
                           SearchCnd :xx...x[yy...y],...
                           KeyCnd    :zz...z
                           RowCnd    :{GG...G}
                           IfThenCnd :{HH...H}
      [2] SELECT-APSL (ss...s)  .................................2
            Table Name   :mm...m(mm...m) nnnnnnnnnn(nn...n)
            Column ID    :oooooo(oo...o)
```

```
    Predicate    :pp...p
    Threshold    :qq...q
[1] rr...r  ..........................................3
  Index Name :tt...t uuuuuuuuuu(uu...u) (vv...v) ww...w
                SearchCnd :xx...x[yy...y],...
                KeyCnd    :zz...z
  RowCnd     :{GG...G}
  IfThenCnd  :{HH...H}
[2] rr...r  ..........................................4
  RowCnd     :{GG...G}
  IfThenCnd  :{HH...H}
```

[説明]

1 は，結合方法で第一候補が選ばれたときの表の検索方法です。

2 は，結合方法で第二候補が選ばれたときの表の検索方法（この場合 SELECT-APSL となります）です。

3 は，結合方法で第二候補が選ばれたときの，表の検索方法の第一候補です。

4 は，結合方法で第二候補が選ばれたときの，表の検索方法の第二候補です。

aa...a(aa...a)：

検索対象となる表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。検索処理が複数ある場合は，この行で区切って情報を表示します。

bbbbbbbbbb(bb...b)：

検索対象となる表 ID を 16 進数（10 進数）で表示します。

c　：

最適化情報収集ユティリティ（pdgetcst）による最適化情報の取得の有無を表示します。

Y：最適化情報を取得しています。

N：最適化情報を取得していません。

dd...d：

HiRDB がアクセスパスを決定するために使用する表の行数を表示します。

デフォルトの行数は，SQL 最適化モードによって異なります。

- コストベース最適化モード 1，又は HiRDB/シングルサーバでコストベース最適化モード 2 の場合
  1000 万行
- HiRDB/パラレルサーバでコストベース最適化モード 2 の場合
  1000 万行×表が格納された RD エリアがあるバックエンドサーバ数

最適化情報収集ユティリティ（pdgetcst）を実行すると，HiRDB がアクセスパスを決定するために使用する表の行数を変更できます。

ee...e：

表の分割列数を表示します。

ff...f：

表分割の種別を表示します。

NON DIVISION：非分割

KEY RANGE：キーレンジ分割（格納条件指定）

PARTITION：キーレンジ分割（境界値指定）

MULTIDIM PARTITION：マトリクス分割（境界値指定）

MULTIDIM：ハッシュ分割を含むマトリクス分割

FLEXIBLE HASH：リバランス機能を使用しないフレキシブルハッシュ分割

FIX HASH：リバランス機能を使用しない FIX ハッシュ分割

RB FLEXIBLE HASH：リバランス機能を使用するフレキシブルハッシュ分割

RB FIX HASH：リバランス機能を使用する FIX ハッシュ分割

gg...g：

ハッシュ分割時に，ハッシュ関数名を表示します。ハッシュ分割時以外は表示しません。また，ハッシュ分割を含むマトリクス分割の場合も表示しません。

hh...hRD/**ii...i**BES：

hh...h には表の分割数（RD エリア数）が，ii...i には表が格納された RD エリアがあるバックエンドサーバ数を表示します。

非横分割表の場合，hh...h は 1 となります。また，HiRDB/シングルサーバの場合，"/ii...iBES"を表示しません。

jjjj(jj...j),...：

表格納用 RD エリアの RD エリア ID を 16 進数（10 進数）で表示します。

表分割をしている場合は分割数分の RD エリア ID を表示します。RD エリア ID に対応する RD エリア名は，pddbls コマンドで調べることができます。

kk...k：

ALL：

表を定義しているすべての RD エリアを検索する場合に表示します。

RESTRICTED：

表を定義している RD エリアの内，探索条件によって該当データがある可能性のある RD エリアだけを検索する場合に表示します。

ll...l：

リバランス機能を使用する表の場合，リバランス状態を表示します。

リバランス機能を使用しない表の場合，この行を表示しません。リバランス機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

NORMAL：

通常の状態，又は ALTER TABLE で RD エリアを追加しただけでリバランスユティリティ（pdrbal）を実行していない状態です。

ON REBALANCE：

表がリバランス中（リバランス開始からリバランス終了まで）の状態です。

mm...m(mm...m)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表名（相関名）を表示します。

nnnnnnnnnn(nn...n)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表 ID を 16 進数（10 進数）で表示します。

oooooo(oo...o)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列 ID を 16 進数（10 進数）で表示します。

pp...p：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる条件の述語を表示します。

qq...q：

SQL オブジェクトを選択するための基準値となるヒット率を表示します。

実行時に算出したヒット率がこの値より小さい場合は，第一候補の SQL オブジェクトを使用します。
実行時に算出したヒット率がこの値と等しい又は大きい場合は，第二候補の SQL オブジェクトを使用します。

rr...r：

検索方法を表示します。

検索方法については，「検索方法の種類」を参照してください。

ss...s：

検索処理に対応する結合方法を表示します。

結合方法については，「結合方法の種類」を参照してください。

tt...t：

検索で使用するインデクス名を表示します。

インデクスを使用しない検索の場合，この行を表示しません。

uuuuuuuuuu(uu...u)：

検索で使用するインデクス ID を 16 進数（10 進数）で表示します。

vv...v：

インデクス属性を組み合わせた情報を表示します。

数字：インデクス構成列数

G：プラグインインデクス

U：ユニークキーインデクス

C：クラスタキーインデクス

D：分割キーインデクス

E：除外キー値を指定したインデクス

M：繰返し列を含むインデクス

P：主キーインデクス

ww...w：

インデクス構成列情報を列ごとにコンマ(,)で区切って，全体を（）で囲んで表示します。

インデクス構成列情報は，構成列のサーチ方向とインデクス構成列名で表示します。

構成列のサーチ方向は，構成列を昇順にサーチする場合には＋，降順にサーチする場合には−を表示します。

例）　第1構成列(C1)は昇順，第2構成列(C2)は降順にサーチする場合
(+C1,-C2)

xx...x[yy...y],...：

サーチ条件の種類，及びサーチ条件の絞り込み範囲を表示します。サーチ条件については，「サーチ条件」を参照してください。

インデクスを使用した検索でサーチ条件がない場合，サーチ条件の種類には"NONE(FULL SCAN)"を表示します。

サーチ条件によって，絞り込み範囲の数が 0 個となる場合，1 個以上となる場合があります。

インデクスの第 2 構成列以降にサーチ条件があるが，第 1 構成列にはサーチ条件がない場合には，サーチ条件の絞り込み範囲の後ろに"(FULL SCAN)"を表示します。

zz...z：

キー条件を表示します。キー条件については，「キー条件」を参照してください。

キー条件がない場合は，この行を表示しません。

AA...A：

共用表が検索対象の場合，"SHARED"を表示します。共用表でない場合，表示しません。

(5)　ビュー表検索のために作業表を作成する場合

```
SCAN
 # Table Name :aa...a(aa...a) bbbbbbbbbb(bb...b)  {AA...A}
   Cost       :(cc...cROW)  {T-BB...B}
   Scan Type  :dd...d
   RowCnd     :{CC...C}
   IfThenCnd  :{DD...D}
```

[説明]

aa...a(aa...a)：

ビュー表名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。

bbbbbbbbbb(bb...b)：

ビュー表 ID を 16 進数（10 進数）で表示します。

cc...c：

HiRDB がアクセスパスを決定するために使用する，表の行数を表示します。

dd...d：

"LIST SCAN"を表示します。

"LIST SCAN"については，「検索方法の種類」を参照してください。

(6)　WITH 句のために作業表を作成する場合

```
SCAN
 # Table Name :aa...a(aa...a)  {AA...A}
   Cost       :(bb...bROW)  {T-BB...B}
   Scan Type  :cc...c
   RowCnd     :{CC...C}
   IfThenCnd  :{DD...D}
```

[説明]

aa...a(aa...a)：

WITH 句問合せ名（相関名）を表示します。

相関名を使用していない場合は，（相関名）を表示しません。

bb...b：

HiRDB がアクセスパスを決定するために使用する，表の行数を表示します。

cc...c：

"LIST SCAN"を表示します。

"LIST SCAN"については，「検索方法の種類」を参照してください。

(7)　FROM 句に指定した導出表のために作業表を作成する場合

```
SCAN
 # Table Name :aa...a(aa...a) {AA...A}
   Cost       :(bb...bROW)  {T-BB...B}
   Scan Type  :cc...c
   RowCnd     :{CC...C}
   IfThenCnd  :{DD...D}
```

[説明]

aa...a(aa...a)：

"(NO NAME)"又は"(NO NAME)(相関名)"を表示します。

bb...b：

HiRDB がアクセスパスを決定するために使用する，表の行数を表示します。

cc...c：

"LIST SCAN"を表示します。

"LIST SCAN"については，「検索方法の種類」を参照してください。

(8)　HiRDB が内部的に作業表を作成する場合

```
SCAN
 # Table Name :aa...a  {AA...A}
   Cost       :(bb...bROW)  {T-BB...B}
   Scan Type  :cc...c
   RowCnd     :{CC...C}
   IfThenCnd  :{DD...D}
```

[説明]

aa...a：

HiRDB が内部的に作成する作業表名を表示します。

HiRDB が内部的に作成する作業表名は，"(DUMMY 作業表番号)"となります。作業表番号は 3 けたの整数です。

bb...b：

HiRDB がアクセスパスを決定するために使用する，表の行数を表示します。

cc...c：

"LIST SCAN"を表示します。

"LIST SCAN"については，「検索方法の種類」を参照してください。

(9)　HiRDB/シングルサーバで SELECT-APSL をする場合

```
--- SELECT-APSL ---
  Table Name    :aa...a(aa...a) bbbbbbbbbb(bb...b)
  Column ID     :cccccc(cc...c)
  Predicate     :dd...d
  Threshold     :ee...e
[1] ..................................................1
Section No     :ff...f
   :
[2] ..................................................2
Section No     :ff...f
   :
```

[説明]

1 はアクセスパスの第一候補，2 はアクセスパスの第二候補を示しています。アクセスパスについては，「アクセスパスの種類」を参照してください。

HiRDB/シングルサーバの場合，シングルサーバは結合処理及び検索処理ごとに SELECT-APSL をしないで，SQL 文全体の SQL オブジェクトを二つ作成します。

aa...a(aa...a)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表名（相関名）を表示します。

bbbbbbbbbb(bb...b)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列の表 ID を 16 進数（10 進数）で表示します。

cccccc(cc...c)：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる述語で使用する列 ID を 16 進数（10 進数）で表示します。

dd...d：

SQL オブジェクトを選択するための，SQL 実行時ヒット率の算出対象となる条件の述語を表示します。

ee...e：

SQL オブジェクトを選択するための基準値となるヒット率を表示します。

実行時に算出したヒット率がこの値より小さい場合は，第一候補の SQL オブジェクトを使用します。

実行時に算出したヒット率がこの値と等しい又は大きい場合は，第二候補の SQL オブジェクトを使用します。

ff...f：

セクション番号を表示します（SQL ごとに番号が付きます）。

動的 SQL の場合は，すべて 1 となります。

17.5.5　結合方法の種類

編集結果の L は外表，R は内表を表しています。結合方式については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

(1)　マージジョイン

マージジョインは，外表を余り絞り込めないときに有効です。

SORT MERGE JOIN

外表と内表から行を取り出して，それぞれ作業表を作成しソートします。その後，結合条件を満たした行同士を結合します。

KEY SCAN MERGE JOIN

外表と内表から，KEY SCAN で行を取り出します。その後，結合条件を満たした行同士を結合します。

LIST SCAN MERGE JOIN

外表と内表から作業表を作成し，ソートはしないで結合列の昇順に行を取り出します。その後，結合条件を満たした行同士を結合します。

L-KEY R-LIST MERGE JOIN

外表は KEY SCAN で行を取り出し，内表は作業表を作成し，ソートはしないで行を取り出します。その後，結合条件を満たした行同士を結合します。

L-KEY R-SORT MERGE JOIN

外表は KEY SCAN で行を取り出し，内表は作業表を作成し，ソートをして行を取り出します。その後，結合条件を満たした行同士を結合します。

L-LIST R-KEY MERGE JOIN

外表は作業表を作成し，ソートはしないで行を取り出し，内表は KEY SCAN で行を取り出します。その後，結合条件を満たした行同士を結合します。

L-LIST R-SORT MERGE JOIN

外表は作業表を作成し，ソートはしないで行を取り出し，内表は作業表を作成し，ソートをして行を取り出します。その後，結合条件を満たした行同士を結合します。

L-SORT R-KEY MERGE JOIN

外表は作業表を作成し，ソートをして行を取り出し，内表は KEY SCAN で行を取り出します。その後，結合条件を満たした行同士を結合します。

L-SORT R-LIST MERGE JOIN

外表は作業表を作成し，ソートをして行を取り出し，内表は作業表を作成し，ソートをしないで行を取り出します。その後，結合条件を満たした行同士を結合します。

(2) ネストループジョイン

ネストループジョインは，内表にインデクスが定義されていて，外表をかなり絞り込めるときに有効です。

NESTED LOOPS JOIN

外表から 1 行ずつ行を取り出し，内表のそれぞれの行に突き合わせて，結合条件を満たす行を取り出す入れ子型のループ処理をする結合方法です。

R-LIST NESTED LOOPS JOIN

内表から行を取り出して作業表を作成します。その後，外表から 1 行ずつ行を取り出して，それぞれの行に対して内表から作成した作業表を突き合わせ，結合条件を満たす行を取り出す入れ子型のループ処理をする結合方法です。

(3) ハッシュジョイン

HASH JOIN{FOR EACH}

あらかじめ内表の結合列の値でハッシングしてハッシュ表を作成しておいて，外表を 1 行取り出すごとに外表の結合列の値でハッシングして，内表から作成しておいたハッシュ表と突き合わせて結合します。

(4) SELECT-APSL

SELECT-APSL は，SQL 実行時に結合方法を動的に決定する方式です。

SELECT-APSL（HiRDB/パラレルサーバの場合だけ）

条件に？パラメタを含んでいる場合，？パラメタの値によって最適な結合方法が変わることがあります。また，SQL 最適化処理時に？パラメタの値が決まらないため，最適な結合方法が決定できません。そのため，SQL 実行時にヒット率を計算して結合方法を決定します。

(5) 直積

CROSS JOIN

外表のすべての行と，内表のすべての行をそれぞれ組み合わせて結合します。二つ以上の表にわたった条件があれば，結合した後に判定します。

17.5.6 検索方法の種類

検索方式については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

(1) インデクスを使用しない検索

TABLE SCAN

インデクスを使用しないで，表のデータページを検索します。

(2) 一つのインデクスを使用した検索

INDEX SCAN

単一列インデクスのインデクスページを検索して絞り込んだ後，表のデータページを検索します。

KEY SCAN

単一列インデクスのインデクスページだけを検索します。データページは検索しません。

MULTI COLUMNS INDEX SCAN

複数列インデクスのインデクスページを検索して絞り込んだ後，表のデータページを検索します。

MULTI COLUMNS KEY SCAN

複数列インデクスのインデクスページだけを検索します。データページは検索しません。

PLUGIN INDEX SCAN

プラグインインデクスを使用して絞り込んだ後，表のデータページを検索します。

PLUGIN KEY SCAN

プラグインインデクスを使用してインデクスページだけを検索します。データページは検索しません。

(3) SELECT-APSL

SELECT-APSL（HiRDB/パラレルサーバの場合だけ）

条件に？パラメタを含んでいる場合，？パラメタの値によって最適な検索方法が変わることがあります。また，前処理時に？パラメタの値が決まらないため，最適な検索方法が決定できません。そのため，SQL 実行時にヒット率を計算して検索方法を決定します。

(4) 複数のインデクスを使用した検索

AND PLURAL INDEXES SCAN

AND 演算子又は OR 演算子で結ばれた探索条件に対して，それぞれのインデクスを使用して検索し，行識別子（ROWID）をそれぞれの作業表に格納します。これらの作業表を，AND 演算子の場合は積集合，OR 演算子の場合は和集合，ANDNOT 演算子（ASSIGN LIST 文でだけ使用できます）の場合は差集合をとって，一つの作業表にまとめます。その後，この作業表の行識別子を基に行を取り出します。

各条件から行識別子の作業表を作成するときに，条件列にインデクスがなくても，TABLE SCAN によって作業表を作成する場合があります。

OR PLURAL INDEXES SCAN

OR 演算子で結ばれた探索条件に対して，それぞれのインデクスを使用して検索し，行識別子（ROWID）を一つの作業表に格納します。この作業表の重複する行を重複排除した後，行識別子を基に行を取り出します。

各条件から行識別子の作業表を作成するときに，条件列にインデクスがなくても，TABLE SCAN によって作業表を作成する場合があります。

(5) 作業表の検索

LIST SCAN

内部的に作成した作業表を検索します。

(6) 行識別子を使用した検索

ROWID FETCH

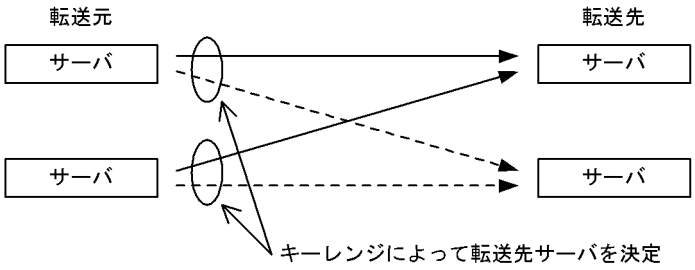
行識別子（ROWID）をキーにして表を検索します。行を取り出す必要がない場合は検索しません。

17.5.7 転送方法の種類

(1) キーレンジ転送

KEY RANGE

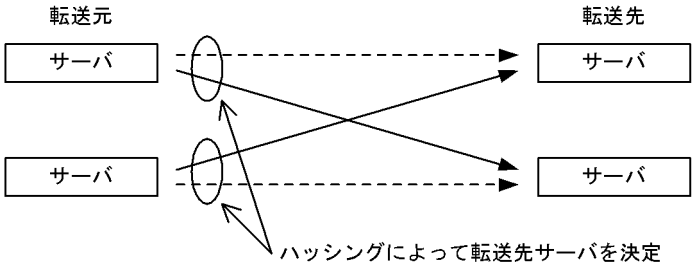
キーレンジによって転送先のサーバを決定して転送する方式です。



(2) ハッシュ転送

HASH

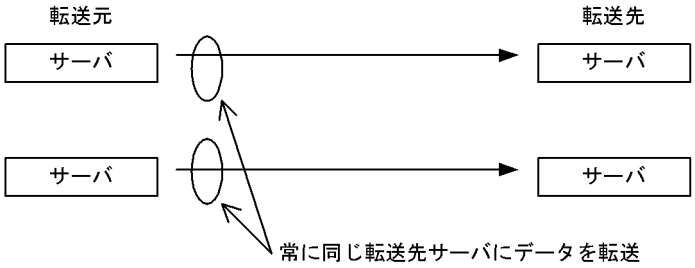
ハッシングによって転送先のサーバを決定して転送する方式です。



(3) 1 対 1 転送

1 TO 1

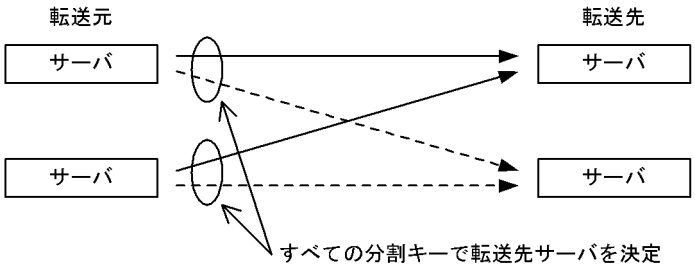
転送元のサーバと転送先のサーバが 1 対 1 に決定するように転送する方式です。



(4) マトリクス分割転送

MULTIDIM

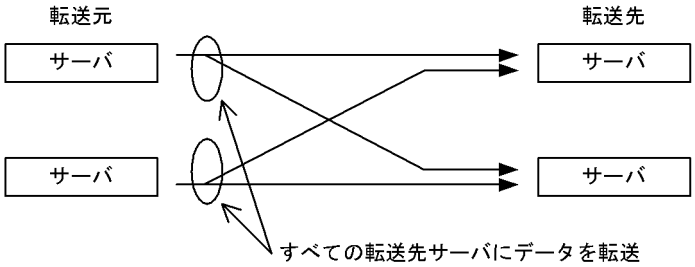
ハッシュ分割を含むマトリクス分割表のすべての分割キーで転送先のサーバを決定して転送する方式です。



(5) ブロードキャスト転送

BROADCAST

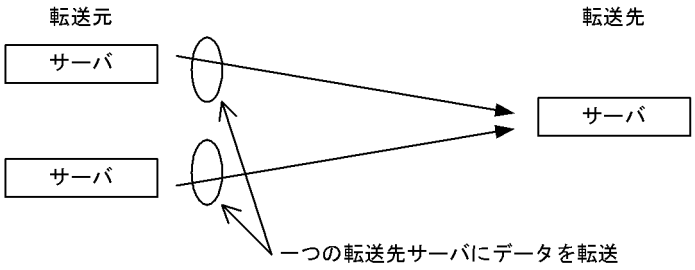
転送元のデータをコピーしてすべての転送先サーバに転送する方式です。



(6) ユニキャスト転送

UNICAST

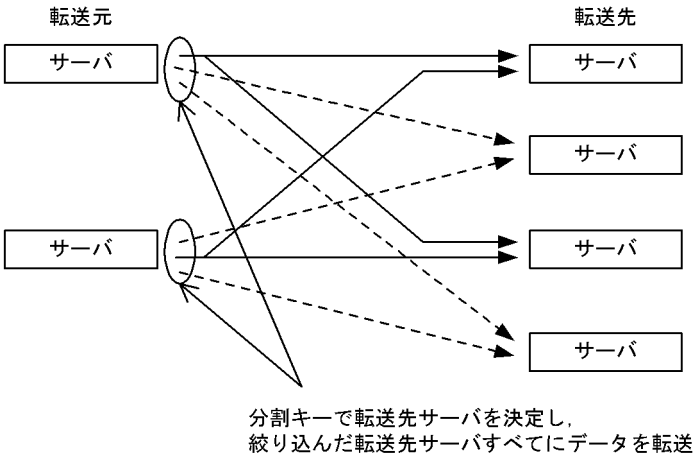
転送元のデータを一つの転送先サーバに転送する方式です。ただし，結合処理情報の Join Type 以外の箇所は"BROADCAST"と表示します。



(7) キーレンジ部分ブロードキャスト転送

KEY RANGE PARTIAL BROADCAST

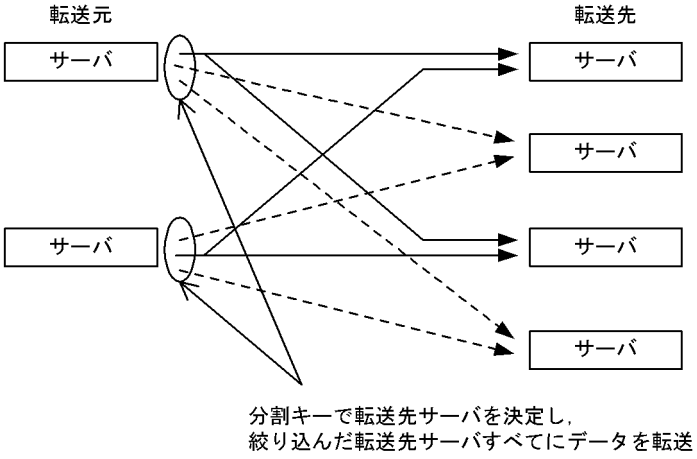
マトリクス分割（境界値）表の一部のキーレンジで転送先サーバを絞り込み，絞り込んだ転送先サーバすべてに転送元のデータをコピーして転送する方式です。



(8)　部分ブロードキャスト転送

PARTIAL BROADCAST

ハッシュ分割を含むマトリクス分割表の一部の分割キーで転送先サーバを絞り込み，転送元のデータをコピーしてから，絞り込んだ転送先サーバすべてに転送する方式です。



17.5.8　アクセスパスの種類

SELECT-APSL（HiRDB/シングルサーバの場合だけ）

条件に？パラメタを含んでいる場合，？パラメタの値によって最適なアクセスパスが変わることがあります。また，SQL 最適化処理時に？パラメタの値が決まらないため，最適なアクセスパスが決定できません。そのため，SQL 実行時にヒット率を計算してアクセスパスを決定します。

17.5.9　グループ分け処理方式の種類

暗黙的グループ分け処理：

問合せ式に GROUP BY 句を指定しないで，選択式又は HAVING 句中に集合関数を指定した場合の処理です。

明示的グループ分け処理：

問合せ式に GROUP BY 句を指定した場合の処理です。

(1)　HiRDB/シングルサーバ

(a)　暗黙的グループ分け処理をする場合

IMPLICIT MIN-MAX INDEX：

集合関数 MIN，MAX を求めるときに，インデクスの最小値，最大値だけを検索する処理方式です。

IMPLICIT SET FUNCTION SCAN{+}：

行を検索するときに，集合関数の結果を同時に求める処理方式です。

IMPLICIT NORMAL：

行を検索してから，集合関数の結果を求める処理方式です。

IMPLICIT LIST SORT{SET FUNCTION SCAN}：

集合関数中に DISTINCT が含まれる場合に，作業表を作成し重複排除のためにソートしてから集合関数の結果を求める処理方式です。

IMPLICIT SURROGATE(COUNT)：

サロゲート機能を使用して集合関数の結果（行数）を求める処理方式です。サロゲート機能とは，プラグインインデクスを呼び出して探索条件を評価すると同時に，プラグイン提供関数の内部で集合関数の結果を求め，集合関数の結果を高速に返却する専用処理のことです。

IMPLICIT SORT CANCEL BY INDEX{SET FUNCTION SCAN}：

重複排除のためのソート処理をしなくてもインデクスを検索することで，DISTINCT 集合関数に指定した列をソートできる場合に，重複排除のためのソート処理を省略し，集合関数の結果を求める処理方式です。

(b)　明示的グループ分け処理をする場合

SORT CANCEL BY INDEX{SET SCAN}：

グループ化のためのソート処理をしなくても，インデクスを検索することでグループ化列をソートできる場合に，グループ化のためのソート処理を省略する方式です。

SORT CANCEL BY JOIN：

グループ化のためのソート処理が，マージジョインのためのソート処理によってキャンセルされる場合に，グループ化のためのソート処理を省略する方式です。

HASH：

グループ分け高速化機能を使用する方式です。グループ分け高速化機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

LIST SORT：

作業表を作成してソートすることでグループ化処理をし，グループ化の結果を求める処理方式です。

PRE-SORT JOIN：

すべてのグループ化列が，結合する前の外表又は外側の結合表に含まれる場合に，結合する前の外表又は外側の結合結果に対してソートをして，ソート順を保ったまま結合してグループ化処理をすることで，グループ化の結果を求める処理方式です。

(2) HiRDB/パラレルサーバ

(a) 暗黙的グループ分け処理をする場合

IMPLICIT MIN-MAX INDEX：

集合関数 MIN，MAX を求めるときに，各バックエンドサーバで，インデクスの最小値，最大値だけを検索する処理方式です。

IMPLICIT SET FUNCTION SCAN{+}：

各バックエンドサーバで，行を検索するときに，集合関数の結果を同時に求める処理方式です。

IMPLICIT NORMAL：

各バックエンドサーバで，行を検索してから，バックエンドサーバごとの集合関数の結果を求める処理方式です。

IMPLICIT FLOATABLE：

フロータブルサーバに集めて全体の集合関数の結果を求める処理方式です。

IMPLICIT FLOATABLE SORT{SET FUNCTION SCAN}：

集合関数中に DISTINCT が含まれる場合に，フロータブルサーバに集めて作業表を作成し，重複排除のためにソートしてから集合関数の結果を求める処理方式です。

IMPLICIT SURROGATE(COUNT)：

各バックエンドサーバで，サロゲート機能を使用してバックエンドサーバごとの集合関数の結果（行数）を求める処理方式です。サロゲート機能とは，プラグインインデクスを呼び出して探索条件を評価すると同時に，プラグイン提供関数の内部で集合関数の結果を求め，集合関数の結果を高速に返却する専用処理のことです。

IMPLICIT SORT CANCEL BY INDEX{SET FUNCTION SCAN}：

重複排除のためのソート処理をしなくてもインデクスを検索することで，DISTINCT 集合関数に指定した列をソートできる場合に，重複排除のためのソート処理を省略し，集合関数の結果を求める処理方式です。

IMPLICIT LIST SORT{SET FUNCTION SCAN}：

集合関数に DISTINCT が含まれる場合に，各バックエンドサーバで作業表を作成し，重複排除のためにソートしてから集合関数の結果を求める処理方式です。

(b)　明示的グループ分け処理をする場合

SORT CANCEL BY INDEX{SET SCAN}：

グループ化のためのソート処理をしなくても，インデクスを検索することでグループ化列をソートできる場合に，グループ化のためのソート処理を省略する方式です。

SORT CANCEL BY JOIN：

グループ化のためのソート処理が，マージジョインのためのソート処理によってキャンセルされる場合に，グループ化のためのソート処理を省略する方式です。

HASH：

グループ分け高速化機能を使用してグループ化の結果を求める処理方式です。グループ分け高速化機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

LIST SORT：

各バックエンドサーバで作業表を作成してソートし，グループ化の結果を求める処理方式です。

PRE-SORT JOIN：

すべてのグループ化列が，結合する前の外表又は外側の結合表に含まれる場合に，結合する前にソートをし，ソート順を保ったまま結合してグループ化処理をすることで，グループ化の結果を求める処理方式です。

FLOATABLE SORT：

複数のフロータブルサーバに転送し，作業表を作成してソートすることでグループ化の結果を求める処理方式です。

17.5.10　プラグイン提供関数の種別

インデクス型プラグインを使用して検索をするプラグイン提供関数は，検索時のプラグインインデクス使用条件によって，次の種別に分かれます。

(1)　SCAN TYPE プラグイン提供関数

探索条件に論理述語を使用して指定するプラグイン提供関数です。

通常はプラグインインデクスを使用して検索しますが，プラグインインデクスを定義していないか，又はプラグインインデクスを使用できないような検索の場合でも動作できるプラグイン提供関数のことです。

(2)　INDEX SCAN TYPE プラグイン提供関数

探索条件に論理述語（ただし，IS TRUE だけ指定できます）を使用して指定するプラグイン提供関数です。必ずプラグインインデクスを使用して検索します。これをインデクス型プラグイン専用関数といいます。

プラグインインデクスを定義していないか，又はプラグインインデクスを使用できないような検索の場合にはエラーとなります。

(3)　FULL SCAN TYPE プラグイン提供関数

探索条件に論理述語を使用して指定するプラグイン提供関数です。

プラグインインデクスを定義すると，インデクスを使用した検索はできますが，インデクスの全範囲をサーチします。そのため，探索条件に FULL SCAN TYPE プラグイン提供関数の条件だけある場合には，プラグインインデクスを使用して検索します。しかし，ほかに探索条件がある場合には，ほかの条件を判定した後にプラグインインデクスを使用しないで判定します。

17.5.11　サーチ条件

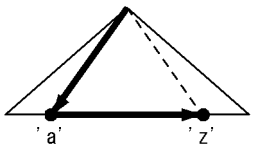
(1)　サーチ条件の概要

SearchCnd：サーチ条件の種別 [絞り込み範囲],...

サーチ条件とは，インデクスのサーチ範囲を決定する条件です。

(a)　サーチ条件を持つ場合

例）　where C1 between 'a' and 'z'



SearchCnd：RANGE(CS-CE)['a','z']

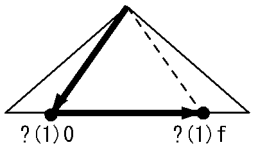
注　使用するインデクスの構成列が C1 の場合です。

(b)　実行時にサーチ条件が削除される場合

パターン文字列に，埋込み変数，？パラメタ，SQL 変数，SQL パラメタを持つ LIKE 述語，又は SIMILAR 述語を指定すると，実行時に与えられるパターン文字列が'abc%'のように前方一致となる場合にはインデクスのサーチ範囲を絞り込めます。しかし，'%abc'のように前方一致以外の場合には，インデクスのサーチ範囲を絞り込めなくなります。

このように，実行時に与えられるパターン文字列の値によって，インデクスのサーチ範囲が絞り込めなくなる場合に，インデクスの絞り込み開始値，終了値を＜＞で囲んで表示します。

例） where C1 LIKE ? （C1 のデータ型：CHAR(5)）



SearchCnd : RANGE(CS-CE)[<?(1)0>,<?(1)f>]

注　使用するインデクスの構成列が C1 の場合です。

［説明］

- ？が’abc%’の場合，X’6162630000’～X’616263ffff’の範囲を検索します。
  文字コードは，a を X’61’，b を X’62’，c を X’63’とします。
- ？が’%abc’の場合，インデクスの全範囲をサーチします。
- ？(1)の番号は？番号を表します。？番号については「絞り込み範囲に使用する値」を参照してください。

(2)　探索条件のサーチ条件種別

各サーチ条件種別に該当する探索条件を次の表に示します。

表 17–1　各サーチ条件種別に該当する探索条件

サーチ条件種別	特長	サーチ条件種別を作成するための条件	
		単一列インデクス	複数列インデクス
IS NULL	NULL 値のキーが少ない場合に有効です。	すべてのインデクス構成列に NULL 述語（IS NULL）があります。	
IS NOT NULL	NULL 値以外のキーが少ない場合に有効です。	インデクス構成列に NULL 述語（IS NOT NULL）があります。	作成しません。
IS TRUE	プラグイン提供関数によって異なります。	プラグイン提供関数の実行結果に対する，論理述語（IS TRUE）を使用した判定があります。	作成しません。
IS TRUE (AND)	プラグイン提供関数によって異なります。	プラグイン提供関数の実行結果に対する，論理述語（IS TRUE）を使用した判定があります。	作成しません。
AT	絞り込み値の重複度が少ない場合に有効です。	インデクス構成列の 1 列以上に比較述語（=）があり，これ以外の全インデクス構成列に NULL 述語（IS NULL）があります。	
RANGE	絞り込む範囲の件数が少ない場合に有効です。	インデクス構成列に比較述語（>，>=，<，<=），BETWEEN 述語，前方一致となる LIKE 述語，又は SIMILAR 述語のどれかがあります。 又は，複数列インデクスの場合，インデクス構成列の一部に比較述語（=），又は NULL 述語（IS NULL）のどちらかがあります。	

サーチ条件種別	特長	サーチ条件種別を作成するための条件	
		単一列インデクス	複数列インデクス
		ただし，RANGES となる場合を除きます。	
ATS	絞り込み値の重複度が低く，絞り込み値の個数が少ない場合に有効です。	インデクス構成列の 1 列以上に IN(行値構成子※3)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)のどれかがあり※1，これら以外の全インデクス構成列に比較述語(=)，又は IS NULL のどちらかがあります。 ただし，IN(行値構成子※3)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)のどれかに指定した絞り込み値の個数の積が pd_apply_search_ats_num オペランドの指定値を超えると，RANGE 又は RANGES となります※2。	
RANGES	絞り込む範囲に該当する件数が少なく，絞り込む範囲の個数も少ない場合に有効です。	作成しません。	インデクス構成列の一部に，IN(行値構成子※3)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)のどれかがあり※1，これらがインデクス構成列の先頭の場合，又は先頭ではなく，IN(行値構成子※3)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)より前の構成列のすべてに比較述語(=)，又は IS NULL のどちらかがあります。 IN(行値構成子※3)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)のどれかに指定した絞り込み値の個数の積が pd_apply_search_ats_num オペランドの指定値を超えると，RANGE 又は RANGES となります※2。 ただし，ATS となる場合を除きます。

注※1

複数インデクスを利用する場合でも，SQL 最適化モードがコストベース最適化モード 2 のときは，IN（表副問合せ），=ANY（表副問合せ），及び=SOME（表副問合せ）は，1 列だけをサーチ範囲の絞り込みに使用します。また，複数列インデクスを利用する場合，SQL 最適化モードがコストベース最適化モード 1 のときは，IN（表副問合せ），=ANY（表副問合せ），及び=SOME（表副問合せ）は，サーチ範囲の絞り込みに使用しません。

注※2

絞り込み値の個数は，IN（行値構成子）では行値構成子の個数となります。また，IN（表副問合せ），=ANY（表副問合せ），及び=SOME（表副問合せ）では 44 となります。絞り込み値の個数の組み合わせ数によるサーチ条件の分類については，表「絞り込み値の個数の組み合わせ数によるサーチ条件の分類」を参照してください。

注※3

すべての行値構成子の行値構成子要素が値指定，スカラ副問合せ，又は行副問合せの場合に限ります。

絞り込み値の個数の組み合わせ数によるサーチ条件の分類について，次の表に示します。

表 17–2　絞り込み値の個数の組み合わせ数によるサーチ条件の分類

指定条件	絞り込み値の個数の組み合わせ		サーチ条件種別
インデクス構成列の 1 列以上に IN(行値構成子)，IN（表副問合せ），=ANY(表副問合せ)，及び=SOME(表副問合せ)のどれかがあり，これら以外の全インデクス構成列に比較述語(=)，又は IS NULL のどちらかがある	p>n	q>n	RANGE
		q<=n	RANGE
	p<=n	q>n	RANGES※1
		q<=n	ATS※2
インデクス構成列の一部に，IN(行値構成子)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)のどれかがあり，これらがインデクス構成列の先頭の場合，又は先頭ではなく，IN(行値構成子)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)より前の構成列のすべてに比較述語(=)，又は IS NULL のどちらかがある	p>n	q>n	RANGE
		q<=n	RANGE
	p<=n	q>n	RANGES※1
		q<=n	RANGES※3

（凡例）

n：HiRDB のバージョンが 08-03 以前の場合は 255，HiRDB のバージョンが 08-04 以降の場合は pd_apply_search_ats_num オペランドに指定した値

p：インデクスの第 x 構成列に指定した述語の絞り込み値の個数

q：インデクスの第 x 構成列目以降に指定した述語の絞り込み値の組み合わせ個数

x：初めて IN(行値構成子)，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)が指定されたインデクス構成列

注※1

指定された述語のうち，インデクスの第 x 構成列目以降に指定された条件を使用しない場合に該当します。

注※2

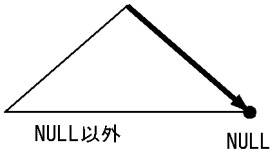
IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)を複数指定した場合，表副問合せを二つ以上評価できないため，絞り込み値の組み合わせ個数に関係なく RANGES となります。

注※3

指定された条件をすべて使用する場合に該当します。ただし，IN(表副問合せ)，=ANY(表副問合せ)，又は=SOME(表副問合せ)を複数指定した場合，表副問合せを二つ以上評価できないため，インデクスの第 x 構成列目以降に指定された条件を使用しない RANGES となります。

(a)　IS NULL

IS NULL は，インデクスの NULL の範囲を検索します。



注　使用するインデクスの構成列が C1 の場合です。

```
例）　where C1 IS NULL
　　→　SearchCnd：IS NULL
```

(b)　IS NOT NULL

IS NOT NULL は，インデクスの NULL 以外の範囲を検索します。



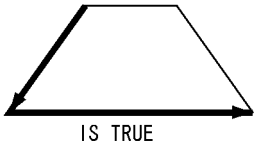
注　使用するインデクスの構成列が C1 の場合です。

```
例）　where C1 IS NOT NULL
　　→　SearchCnd：IS NOT NULL
```

(c)　IS TRUE

IS TRUE は，プラグインインデクスを使用した検索の場合に表示します。

- **FULL SCAN TYPE プラグイン提供関数の場合**

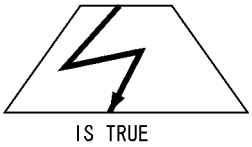


注　プラグインインデクスの構造，及び絞り込む方法は，プラグインによって異なります。

注　使用するインデクスの構成列が feature の場合です。

```
例）　where SearchImageData (feature,? as blob(1M),? as varchar(1024)) IS TRUE
　　→　SearchCnd：IS TRUE
```

- **FULL SCAN TYPE プラグイン提供関数以外の場合**



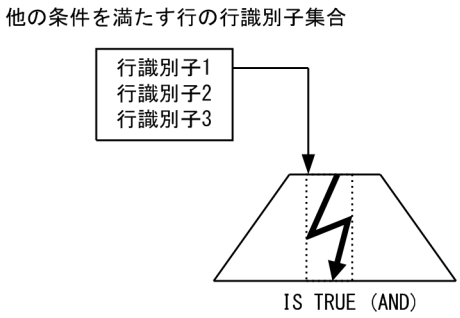
注　プラグインインデクスの構造，及び絞り込む方法は，プラグインによって異なります。

注　使用するインデクスの構成列が SENTENCES の場合です。

```
例）　where Contains (SENTENCES,'添付文書データ[効能{"食あたり"}]') IS TRUE
　　→　SearchCnd：IS TRUE
```

(d)　IS TRUE (AND)

IS TRUE (AND)は，プラグインインデクスを使用した条件の絞り込み時に，ほかのインデクスを使用した条件による絞り込み結果の行識別子集合を利用する場合に表示します。



注　プラグインインデクスの構造，及び絞り込む方法は，プラグインによって異なります。

注　使用するインデクスの構成列が SENTENCES の場合です。

```
例）　where C1 = 'a' AND
      Contains (SENTENCES,'添付文書データ[効能{"食あたり"}]') IS TRUE
      →　下線部分がSearchCnd：IS TRUE (AND)
```

詳細については，マニュアル「HiRDB UAP 開発ガイド」の「プラグインインデクスでの他インデクス絞り込み結果利用機能」を参照してください。

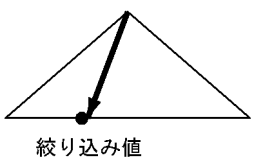
(e)　AT

- **サーチ条件種別**

AT [絞り込み値]：

AT は，インデクスからある値をサーチする場合に使用します。

探索条件は，C1='a',C1=(Select C1 from T2)などです。



注　使用するインデクスの構成列が C1 の場合です。

```
例）　where C1='a'
　　→　SearchCnd：AT ['a']
```

- **絞り込み値**

単一列インデクスの AT

AT [値]：

単一列インデクスの場合に値を表示します。

```
例）　where C1='a'
　　→　SearchCnd：AT ['a']
```

注　使用するインデクスの構成列が C1 の場合です。

複数列インデクスの AT

AT [(値,...,値)]：

複数列インデクスの場合に，値をインデクス構成列の先頭から順にコンマ(,)で区切り，全体を（）で囲んで表示します。

```
例1）　where C1='a' and C2='A'
　　→　SearchCnd：AT [('a','A')]
```

注　使用するインデクスの構成列が C1，C2 の場合です。

```
例2）　where (C1,C2) in (('a','A'))
　　→　SearchCnd：AT [('a','A')]
```

注　使用するインデクスの構成列が C1，C2 の場合です。

```
例3）　where (C1,C2) = ((select C1 from T2),(select C2 from T3))
　　→　SearchCnd：AT [((SUBQ(2)),(SUBQ(3)))]
```

注　使用するインデクスの構成列が C1，C2 の場合です。

```
例4）　where array(C1,C2) [any] ((C1,C2)=('a','A'))
　　→　SearchCnd：AT [('a','A')]
```

注　使用するインデクスの構成列が C1，C2 の場合です。

(f)　RANGE

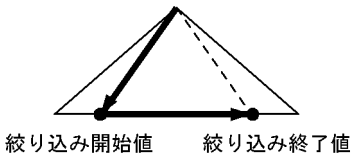
- **サーチ条件種別**

RANGE は，インデクスの絞り込み開始値から終了値までの範囲をサーチする場合に使用します。絞り込み範囲に開始値が含まれるかどうか，終了値が含まれるかどうかによって，RANGE には次の 1〜4 の分類があります。これらは，RANGE(...)の"..."部分に表示します。

1. **絞り込み開始値，絞り込み終了値共に含む**

RANGE(CS-CE) [絞り込み開始値,絞り込み終了値]：

探索条件は，C1>='a' and C1<='z',C1 between 'a' and 'z',C1 like 'a%'などです。




```
例1)   where C1 >= 'a' and C1 <= 'z'
         →  SearchCnd：RANGE(CS-CE) ['a','z']

注　使用するインデクスの構成列がC1の場合です。

例2)   where C1 >= 'a'
         →  SearchCnd：RANGE(CS-CE) ['a',MAX]

注　使用するインデクスの構成列がC1の場合です。

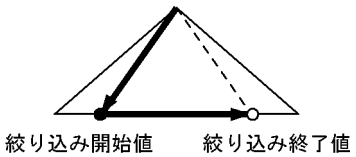
例3)   where C1 <= 'a'
         →  SearchCnd：RANGE(CS-CE) [MIN,'a']

注　使用するインデクスの構成列がC1の場合です。
```

2. **絞り込み開始値は含むが，絞り込み終了値は含まない**

RANGE(CS-OE) [絞り込み開始値,絞り込み終了値]：

探索条件は，C1>='a' and C1 < 'z'などです。




```
例1)   where C1 >= 'a' and C1 < 'z'
         →  SearchCnd：RANGE(CS-OE) ['a','z']

注　使用するインデクスの構成列がC1の場合です。

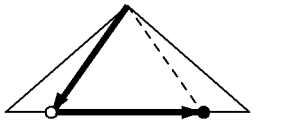
例2)   where C1 < 'a'
         →  SearchCnd：RANGE(CS-OE) [MIN,'a']

注　使用するインデクスの構成列がC1の場合です。
```

3. **絞り込み開始値は含まないが，絞り込み終了値は含む**

RANGE(OS-CE) [絞り込み開始値,絞り込み終了値]：

探索条件は，C1>'a' and C1<='z'などです。




```
例1)   where C1 > 'a' and C1 <= 'z'
         →  SearchCnd：RANGE(OS-CE) ['a','z']

注　使用するインデクスの構成列がC1の場合です。

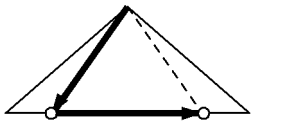
例2)   where C1 > 'a'
         →  SearchCnd：RANGE(OS-CE) ['a',MAX]

注　使用するインデクスの構成列がC1の場合です。
```

4. **絞り込み開始値，絞り込み終了値共に含まない**

   RANGE(OS-OE) [絞り込み開始値,絞り込み終了値]：

   探索条件は，C1>'a' and C1<'z'などです。




```
例)   where C1 > 'a' and C1 < 'z'
        →  SearchCnd：RANGE(OS-OE) ['a','z']

注　使用するインデクスの構成列がC1の場合です。
```

- **絞り込み範囲（絞り込み開始値，絞り込み終了値）**

単一列インデクスの RANGE

RANGE(...) [値,値]：

単一列インデクスの場合，絞り込む範囲の開始位置，終了位置をコンマ(,)で区切って表示します。RANGE(...)の"..."の箇所はサーチ条件種別によって異なります。

```
例)   where C1 between 'a' and 'z'
        →  SearchCnd：RANGE(CS-CE) ['a','z']

注　使用するインデクスの構成列がC1の場合です。
```

複数列インデクスの RANGE

RANGE(...) [(値,...,値),(値,...,値)]：

複数列インデクスの場合，開始位置，終了位置それぞれに対して，インデクス構成列の先頭から順にコンマ(,)で区切って，全体を（）で囲みます。さらに，開始位置，終了位置をコンマで区切って表示します。RANGE(...)の"..."の箇所はサーチ条件種別によって異なります。

```
例1)  where C1 = 'a' and C2 between 'A' and 'Z'
      →  SearchCnd : RANGE(CS-CE) [('a','A'),('a','Z')]

注  使用するインデクスの構成列がC1，C2の場合です。

例2)  where (C1,C2) between ('A', 'Z') and (select C1, C2 from T2)
      →  SearchCnd : RANGE(CS-CE)[( 'A', 'Z'),((SUBQ(2)),(SUBQ(2)))]

注  使用するインデクスの構成列がC1，C2の場合です。

例3)  where (C1,C2) > ('a', 'z') and (C1,C2) < ('A', 'Z')
      →  SearchCnd : RANGE(OS-OE)[( 'a', 'z'),( 'A', 'Z')]

注  使用するインデクスの構成列がC1，C2の場合です。

例4)  where array(C1,C2) [any] ((C1,C2)<( 'A', 'Z'))
      →  SearchCnd : RANGE(CS-OE)[(MIN,MIN),( 'A', 'Z')]

注  使用するインデクスの構成列がC1，C2の場合です。
```

(g)　ATS

- **サーチ条件種別**

複数の絞り込み値による ATS

ATS [絞り込み値], ... ,[絞り込み値]：

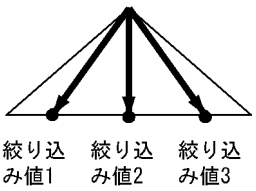
ATS [絞り込み値], ... ,[絞り込み値],...(Number of All Row Values : n)...,[絞り込み値]：

絞り込み値を使用して，インデクスを複数回サーチする場合に使用します。

探索条件は，C1 in ('a','b','c')などです。

絞り込み値の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み値の総数です。

なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。




```
例)  where C1 in ('a','b','c')
     →  SearchCnd : ATS ['a'],['b'],['c']

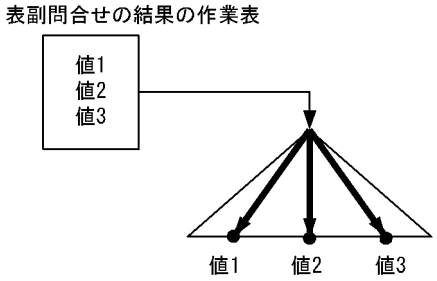
注  使用するインデクスの構成列がC1の場合です。
```

絞り込み値の集合による ATS

ATS [絞り込み値の集合]：

絞り込み値の集合（表副問合せから作成される作業表）を使用して，インデクスを複数回サーチする場合に使用します。

探索条件は，C1=any(Select C1 from T2),C1=some(Select C1 from T2),C1 in (Select C1 from T2)などです。




```
例)   where C1 =any(select C1 from T2)
      →  SearchCnd : ATS [(SUBQ(2))]

注   使用するインデクスの構成列がC1の場合です。
```

複数の絞り込み値の集合による ATS

ATS [絞り込み値の集合], ... ,[絞り込み値の集合]：

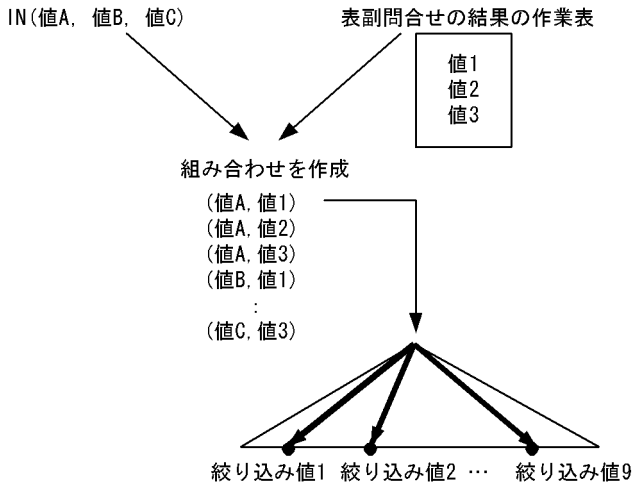
ATS [絞り込み値の集合], ... ,[絞り込み値の集合],...(Number of All Row Values : n)...,[絞り込み値の集合]：

絞り込み値の集合（表副問合せから作成される作業表）と IN 述語を組み合わせた結果を使用して，インデクスを複数回サーチする場合に使用します。

探索条件は，C1 in ('a','b','c') and C2=any(Select C1 from T2)などです。

絞り込み値の集合の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み値の集合の総数です。

なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。




```
例)   where C1 in ('a','b','c') and C2=any(Select C1 from T2)
      →  SearchCnd : ATS [( 'a' ,(SUBQ(2)))],[( 'b' ,(SUBQ(2)))],
                     [( 'c' ,(SUBQ(2)))]

注   使用するインデクスの構成列がC1，C2の場合です。
```

- **絞り込み値，又は絞り込み値の集合**

複数の絞り込み値による単一列インデクスの ATS

ATS [値], ... ,[値]：

ATS [値], ... ,[値],...(Number of All Row Values : n)...,[値]：

値を","で区切って表示します。

絞り込み値の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み値の総数です。

なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。

```
例）  where C1 in ('a','b','c')
      →  SearchCnd : ATS ['a'],['b'],['c']

注  使用するインデクスの構成列がC1の場合です。
```

絞り込み値の集合による単一列インデクスの ATS

ATS [(SUBQ(問合せ ID))又は(SUBQEX(問合せ式本体 ID))]：

表副問合せの問合せ ID，又は問合せ式本体 ID を表示します。

```
例）  where C1 =any(select C1 from T2)
    →  SearchCnd : ATS [(SUBQ(2))]

注  使用するインデクスの構成列がC1の場合です。
```

複数の絞り込み値による複数列インデクスの ATS

ATS [(値,...,値)], ... ,[(値,...,値)]：

ATS [(値,...,値)], ... ,[(値,...,値)],...(Number of All Row Values : n)...,[(値,...,値)：

複数列インデクスの場合，値をインデクス構成列の先頭から順に","で区切り，全体を"(",")"で囲んだものが一つの絞り込み値となります。この絞り込み値を","で区切って，複数の絞り込み値を表示します。

絞り込み値の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み値の総数です。

なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。

```
例1）  where C1 in ('a','b') and C2 in ('A','B') and C3=1
      →  SearchCnd : ATS [('a','A',1)],[('a','B',1)],[('b','A',1)],[('b','B',1)]

注  使用するインデクスの構成列がC1，C2，C3の場合です。

例2）  where (C1,C2) in (('a','b'), ('A','B'))
      →  SearchCnd : ATS [('a','b'),( 'A','B')]

注  使用するインデクスの構成列がC1，C2の場合です。
```

絞り込み値の集合による複数列インデクスの ATS

ATS [(値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)),...,値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)))]：

複数列インデクスの場合，値又は表副問合せの問合せ ID をインデクス構成列の先頭から順に","で区切り，全体を"(",")"で囲んだものが絞り込み値の集合となります。

```
例1)  where C1=any(select C1 from T2) and C2='a' and C3='A'
      →  SearchCnd : ATS [((SUBQ(2)),'a','A')]

注  使用するインデクスの構成列がC1，C2，C3の場合です。

例2)  where C1= 'a' and (C2,C3) in (select C2, C3 from T2)
      →  SearchCnd : ATS [('a',(SUBQ(2)),(SUBQ(2)))]

注  使用するインデクスの構成列がC1，C2，C3の場合です。
```

複数の絞り込み値の集合による複数列インデクスの ATS

ATS [(値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)),...,値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)))]

, ... ,[(値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)),...,値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)))]：

ATS [(値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)),...,値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)))]

, ... ,[(値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)),...,値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)))]

,...(Number of All Row Values : n)...

, [(値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)),...,値，(SUBQ(問合せ ID))，又は(SUBQEX(問合せ式本体 ID)))]：

複数列インデクスの場合，値又は表副問合せの問合せ ID をインデクス構成列の先頭から順に","で区切り，全体を"(",")"で囲んだものが一つの絞り込み値の集合となります。この絞り込み値の集合を","で区切って，複数の絞り込み値の集合を表示します。

絞り込み値の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み値の集合の総数です。

なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。

```
例)  where C1=any(select C1 from T2) and C2 in ('a','b')
           and C3 in ('A','B')
     →  SearchCnd : ATS [((SUBQ(2)),'a','A')],[((SUBQ(2)),'a','B')],
                        [((SUBQ(2)),'b','A')],[((SUBQ(2)),'b','B')]

注  使用するインデクスの構成列がC1，C2，C3の場合です。
```

(h) RANGES

- **サーチ条件種別**

RANGES は，複数列インデクスにだけ適用されます。インデクスの絞り込み開始値から終了値までの範囲を，複数回サーチする場合に使用します。

絞り込み範囲に開始値が含まれるかどうか，終了値が含まれるかどうかによって，RANGES には次の 1〜3 の分類があります。これらは，RANGES(...)の"..."部分に表示します。

1. **複数の絞り込み範囲による RANGES**

   RANGES(...) [絞り込み開始値,絞り込み終了値], ... ,[絞り込み開始値,絞り込み終了値]：

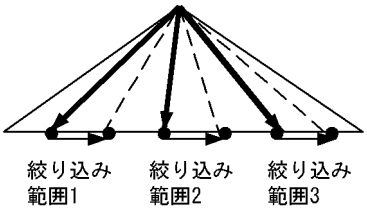
   RANGES(...) [絞り込み開始値,絞り込み終了値], ... ,[絞り込み開始値,絞り込み終了値],...(Number of All Row Values : n)...,[絞り込み開始値,絞り込み終了値]：

   絞り込み範囲を使用して，インデクスを複数回サーチする場合に使用します。

   探索条件は，C1 in ('a','b','c') and C2 between 'A' and 'Z'，C1 in ('a','b','c')などです。

   絞り込み値の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み範囲の総数です。

   なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。




```
例1)   where C1 in (’a’,’b’,’c’) and C2 between ’A’ and ’Z’
        →  SearchCnd : RANGES(CS-CE)
                       [(’a’,’ A’ ),(’a’,’ Z’ )],[(’b’,’ A’ ),(’b’,’ Z’ )],[(’c’,’ A’ ),(
’c’,’ Z’ )]

注　使用するインデクスの構成列がC1，C2の場合です。

例2)   where C1 in (’a’,’b’,’c’)
        →  SearchCnd : RANGES(CS-CE)
                       [(’a’,MIN),(’a’,MAX)],[(’b’,MIN),(’b’,MAX)],
                       [(’c’,MIN),(’c’,MAX)]

注　使用するインデクスの構成列がC1，C2の場合です。

例3)   where (C1,C2) in ((’a’, ’a’), (’b’, ’b’))
        →  SearchCnd : RANGES(CS-CE)
                       [(’a’, ’a’,MIN),( ’a’, ’a’,MAX)] ,
                       [(’b’, ’b’,MIN),( ’b’, ’b’,MAX)]

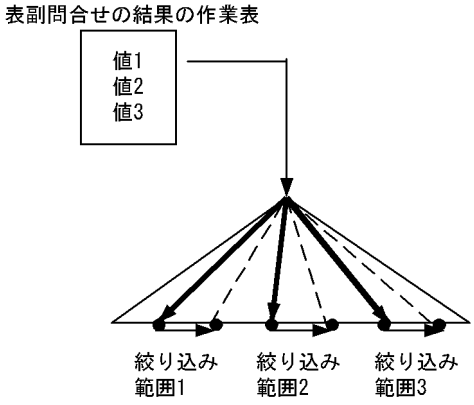
注　使用するインデクスの構成列がC1，C2，C3の場合です。
```

2. **絞り込み範囲の集合による RANGES**

   RANGES(...) [絞り込み開始値の集合,絞り込み終了値の集合]：

   絞り込み範囲の集合（表副問合せから作成される作業表）を使用して，インデクスを複数回サーチする場合に使用します。

探索条件は，C1 =any(select C1 from T2)などです。




```
例1)   where C1 =any(select C1 from T2)
       →  SearchCnd : RANGES(CS-CE)
                      [((SUBQ(2)),MIN),((SUBQ(2)),MAX)]

注　使用するインデクスの構成列がC1，C2の場合です。

例2)   where (C1,C2) = any(select C1, C2 from T2)
       →  SearchCnd : RANGES(CS-CE)
                      [((SUBQ(2)),(SUBQ(2)),MIN),((SUBQ(2)),
                      (SUBQ(2)),MAX)]

注　使用するインデクスの構成列がC1，C2，C3の場合です。
```

3. **複数の絞り込み範囲の集合による RANGES**

RANGES(...) [絞り込み開始値又は絞り込み開始値の集合,絞り込み終了値又は絞り込み終了値の集合],...,[絞り込み開始値又は絞り込み開始値の集合,絞り込み終了値又は絞り込み終了値の集合]：

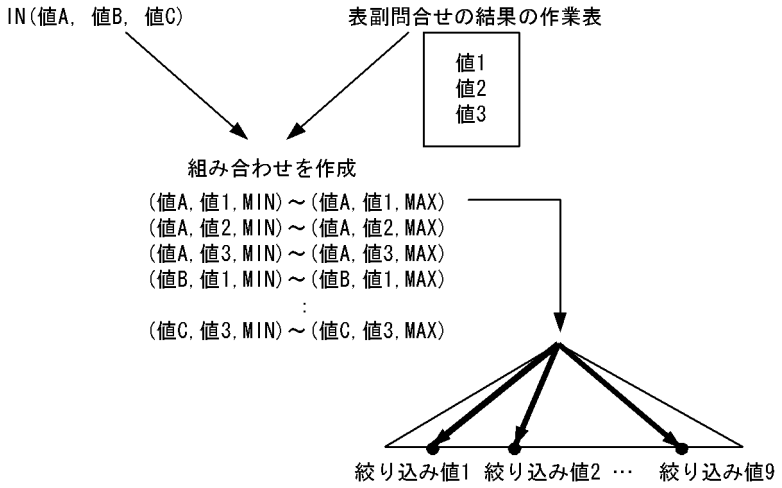
RANGES(...) [絞り込み開始値又は絞り込み開始値の集合,絞り込み終了値又は絞り込み終了値の集合],...,[絞り込み開始値又は絞り込み開始値の集合,絞り込み終了値又は絞り込み終了値の集合],...(Number of All Row Values : n)...,[絞り込み開始値又は絞り込み開始値の集合,絞り込み終了値又は絞り込み終了値の集合]：

絞り込み範囲の集合（表副問合せから作成される作業表）と IN 述語を組み合わせた結果を使用して，インデクスを複数回サーチする場合に使用します。

探索条件は，C1 in ('a','b','c') and C2=any(Select C1 from T2)などです。

絞り込み値の総数が 256 個以上の場合，「...(Number of All Row Values : n)...」を表示します。n は絞り込み範囲の集合の総数です。

なお，「...(Number of All Row Values : n)...」の...は，省略を示すものではありません。




```
例1)   where C1 in ('a','b','c') and C2=any(Select C1 from T2)
      →  SearchCnd : RANGES(CS-CE)
                     [(' a' ,(SUBQ(2)),MIN),(' a' ,(SUBQ(2)),MAX)],
                     [(' b' ,(SUBQ(2)),MIN),(' b' ,(SUBQ(2)),MAX)],
                     [(' c' ,(SUBQ(2)),MIN),(' c' ,(SUBQ(2)),MAX)]

注  使用するインデクスの構成列がC1，C2，C3の場合です。

例2)   where C1 in ('a','b') and (C2,C3)=any(Select C1,C2 from T2)
           and C4>'A'
      →  SearchCnd : RANGES(CS-CE)
                     [(' a' ,(SUBQ(2)),(SUBQ(2)),'A'),
                     (' a' ,(SUBQ(2)),(SUBQ(2)),MAX)],
                     [(' b' ,(SUBQ(2)),(SUBQ(2)),'A'),
                     (' b' ,(SUBQ(2)),(SUBQ(2)),MAX)]

注  使用するインデクスの構成列がC1，C2，C3，C4の場合です。
```

(3)　絞り込み範囲に使用する値

(a)　定数

定数を使用して絞り込む場合に値を表示します。定数の表示形式については，マニュアル「HiRDB SQL リファレンス」を参照してください。

パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合，定数は次の表のように表示します。

表 17–3　パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合の表示例，サーチ範囲

列のデータ型	パターン文字列	サーチ条件の絞り込み範囲の表示例	インデクスのサーチ範囲※
固定長の場合	定数	RANGE(CS-CE) ['abc'00,'abc'ff] (定数を'abc%'とする)	X'6162630000' ～X'616263ffff'

列のデータ型	パターン文字列	サーチ条件の絞り込み範囲の表示例	インデクスのサーチ範囲※
(例： char(5))		最初に出現する'%'，又は'_'以降を取り除いた文字列に対して，開始値は定義長まで 0 を，終了値は定義長まで f を埋めます。	
	埋込み変数，又は？パラメタ	RANGE(CS-CE)［<?(1)0>,<?(1)f>］ 開始値は 0 を一つ，終了値は f を一つ埋めます。	X'6162630000' ～X'616263ffff' （？パラメタを'abc%'とします）
	SQL 変数，又は SQL パラメタ（SQL 変数名を SQLVAR と仮定します）	RANGE(CS-CE)［<SQLVAR0>,<SQLVARf>］ 開始値は 0 を一つ，終了値は f を一つ埋めます。	X'6162630000' ～X'616263ffff' （SQL 変数を'abc%'とします）
可変長の場合 (例： varchar(5))	定数	RANGE(CS-CE)［'abc','abc'ff］ (定数を'abc%'とする) 開始値は最初に出現する'%'又は'_'以降の文字列を取り除いて，終了値は開始値に定義長まで f を埋めます。	X'616263' ～X'616263ffff'
	埋込み変数，又は？パラメタ	RANGE(CS-CE)［<?(1)>,<?(1)f>］ 終了値だけ f を一つ埋めます。	X'616263' ～X'616263ffff' （？パラメタを'abc%'とします）
	SQL 変数，又は SQL パラメタ（SQL 変数名を SQLVAR と仮定します）	RANGE(CS-CE)［<SQLVAR>,<SQLVARf>］ 終了値だけ f を一つ埋めます。	X'616263' ～X'616263ffff' （SQL 変数を'abc%'とします）

注※　文字コードは a を X'61'，b を X'62'，c を X'63'とします。

(b)　？（？番号）

埋込み変数，又は？パラメタの場合に表示します。？番号とは，SQL 文中にある埋込み変数，又は？パラメタに対して，1 を開始値として左から出現順に番号を付けたものです。

パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合，？は表「パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合の表示例，サーチ範囲」のように表示します。

(c)　SQL 変数名

SQL 変数の場合に表示します。

パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合，SQL 変数名は表「パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合の表示例，サーチ範囲」のように表示します。

(d)　SQL パラメタ名

SQL パラメタの場合に表示します。

トリガ SQL 文中で新値相関名の列，及び旧値相関名の列を使用している場合，新値相関名の列は「(NEWROW).列名」，旧値相関名の列は「(OLDROW).列名」と表示します。

パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合，SQL パラメタ名は表「パターン文字列が前方一致となる LIKE 述語，又は SIMILAR 述語を指定した場合の表示例，サーチ範囲」のように表示します。

(e)　USER

USER 値関数を使用して絞り込む場合に表示します。

(f)　CURRENT_DATE

CURRENT_DATE 値関数を使用して絞り込む場合に表示します。

(g)　CURRENT_TIME

CURRENT_TIME 値関数を使用して絞り込む場合に表示します。

(h)　CURRENT_TIMESTAMP(p)

CURRENT_TIMESTAMP 値関数を使用して絞り込む場合に表示します（P=0，2，4，又は 6）。

(i)　SUBQ(問合せ ID)

集合演算を含まない副問合せ結果を使用して絞り込む場合に表示します。（）の中に問合せ ID を表示します。

(j)　SUBQEX(問合せ式本体 ID)

集合演算を含む副問合せ結果を使用して絞り込む場合に表示します。（）の中に問合せ ID を表示します。

(k)　集合関数

集合関数を使用して絞り込む場合に表示します。

(l)　NULL

NULL 値を使用して絞り込む場合に表示します。

(m)　表名.列名

列を使用して絞り込む場合に表示します。

表名が相関名の場合には，表名の代わりに相関名を（）で囲んで表示します。

(n)　(NEW ROW).列名

制約判定のために HiRDB が内部的に生成する副問合せ中で，外部キーへの挿入値，又は更新値を使用して絞り込む場合に表示します。

(o)　表名.列名[添字]

繰返し列（添字は整数）を使用して絞り込む場合に表示します。

表名が相関名の場合には，表名の代わりに相関名を（ ）で囲んで表示します。

(p)　MIN

インデクスキー値の中で一番小さい値（文字列の場合には文字コードが一番小さい値）を検索する場合に表示します。

(q)　MAX

インデクスキー値の中で一番大きい値（文字列の場合には文字コードが一番大きい値）を検索する場合に表示します。

17.5.12　キー条件

(1)　キー条件の概要

KeyCnd：キー条件

キー条件とは，インデクスの構成列だけで評価できる条件のことをいいます。

データページを参照しないでインデクスページだけで探索条件を評価できるため，高速に検索できます。サーチ条件でインデクスのサーチ範囲を絞り込んだ後に，キー条件で評価すると効果的です。サーチ条件，及びキー条件で評価できない述語については，データページを検索する際に評価します。

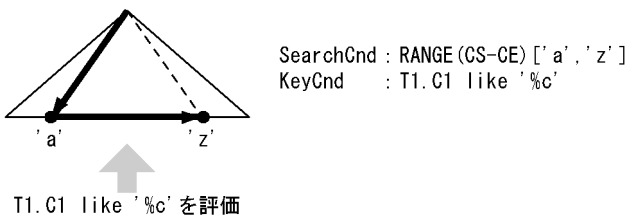
キー条件が複数ある場合には，条件を AND，又は OR で結合した形になります。

AND，及び OR の優先順位によって，（ ）で囲んで表示します。

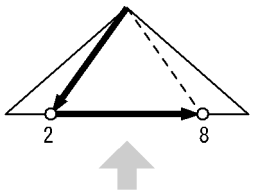
(a)　サーチ条件とキー条件の両方を持つ場合

サーチ条件でインデクスを絞り込んで，キー条件で評価します。

例）　where T1.C1 between 'a' and 'z' and T1.C1 like '%c'



例） where XMLEXISTS('/A/B/C/D/E[./F<8 and ./F>2 and ./F<>5] '
passing by value T1.C1)



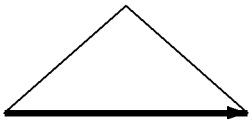
SearchCnd: RANGE(OS-OE) [2,8]
KeyCnd : /A/B/C/D/E/F<>5

T1.C1について（'/A/B/C/D/E[./F<>5] '）を評価
ただし，名前空間は表示しません。

(b) キー条件だけ持つ場合

インデクスの端から端までをキー条件で評価します。インデクスの全範囲をサーチする必要があるため，サーチ条件とキー条件の両方がある場合に比べて，性能的に遅くなります。

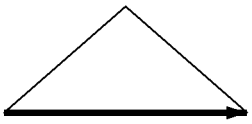
例） where T1.C1 like '%c'



SearchCnd:NONE(FULL SCAN)
KeyCnd : T1.C1 like '%c'

T1.C1 like '%c'を評価

例） where XMLEXISTS('/A/B/C/D/E[./F<>0] 'passing by value T1.C1)



SearchCnd: NONE(FULL SCAN)
KeyCnd : /A/B/C/D/E/F<>0

T1.C1について（'/A/B/C/D/E[./F<>0] '）を評価
ただし，名前空間は表示しません。

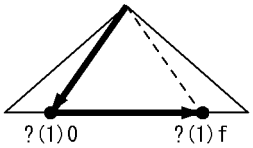
(c) 実行時にキー条件が削除される場合

パターン文字列に埋込み変数，？パラメタ，SQL 変数，SQL パラメタを持つ LIKE 述語，又は SIMILAR 述語を指定すると，実行時に与えられるパターン文字列が'abc%d'のような場合には，サーチ条件で'abc'から始まる値をサーチし，キー条件で最後が'd'で終わるかどうかを判定します。しかし，実行時に与えられるパターン文字列が'abc%'のような前方一致となる値を指定すると，'abc'から始まる値をサーチするだけで条件を評価できると判断し，実行時にキー条件を評価しなくなります。

このようにパターン文字列の値によって，削除できるようになるキー条件を＜＞で囲んで表示します。

例） where T1.C1 like ？

？パラメタの値が'abc%'（C1 のデータ型：CHAR(5)）




```
SearchCnd : RANGE(CS-CE) [<?(1)0>,<?(1)f>]
KeyCnd    : <T1.C1 like ?(1)>
```

［説明］

- ？が’abc%’の場合：
  X’6162630000’～X’616263ffff’の範囲を検索します（キー条件は削除されます）。
- ？が’abc%d’の場合：
  X’6162630000’～X’616263ffff’の範囲を検索し，キー条件で T1.C1 like ’abc%d’を評価します。
- ？が’%abc’の場合：
  インデクスの全範囲をサーチし，キー条件で T1.C1 like ’%abc’を評価します（サーチ条件だけ削除されます。サーチ条件については，「サーチ条件」を参照してください）。
- ？(1)の番号は，？番号を表します。？番号については「キー条件に使用する値」を参照してください。

(2)　キー条件に含まれる述語

キー条件に含まれる述語の形式を示します。限定述語，EXISTS 述語はキー条件にはなりません。

ここで説明するキー条件の例は，インデクスの定義方法によっては，より高速に検索をするためサーチ条件にする場合があります。

(a)　NULL 述語

値 is null：

インデクスのキー値が NULL かどうかを評価します。

```
例）  where T1.C1 is null
      →  KeyCnd : T1.C1 is null
```

値 is not null：

インデクスのキー値が NULL 以外かどうかを評価します。

```
例）  where T1.C1 is not null
      →  KeyCnd : T1.C1 is not null
```

(b)　IN 述語

値 in (値,値,...,値)：

インデクスのキー値が値に含まれるかどうかを評価します。

```
例）  where T1.C1 in ('a','b','c')
      →  KeyCnd : T1.C1 in ('a','b','c')
```

値 not in (値,値,...,値)：

インデクスのキー値が値に含まれないかを評価します。

```
例）  where T1.C1 not in ('a','b','c')
      →  KeyCnd：T1.C1 not in ('a','b','c')
```

(c) LIKE 述語

値 like パターン文字列 [escape エスケープ文字]：

インデクスのキー値がパターンと一致するかどうかを評価します。

```
例）  where T1.C1 like '%a'
      →  KeyCnd：T1.C1 like '%a'
```

値 not like パターン文字列 [escape エスケープ文字]：

インデクスのキー値がパターンと一致しないかを評価します。

```
例）  where T1.C1 not like '%a¥_' escape '¥'
      →  KeyCnd：T1.C1 not like '%a¥_' escape '¥'
```

(d) XLIKE 述語

値 xlike パターン文字列 [escape エスケープ文字]：

インデクスのキー値が大文字，小文字を区別しないでパターンと一致するかどうかを評価します。

```
例）  where T1.C1 xlike '%a'
      →  KeyCnd：T1.C1 xlike '%a'
```

値 not xlike パターン文字列 [escape エスケープ文字]：

インデクスのキー値が大文字，小文字を区別しないでパターンと一致しないかを評価します。

```
例）  where T1.C1 not xlike '%a¥_' escape '¥'
      →  KeyCnd：T1.C1 not xlike '%a¥_' escape '¥'
```

(e) BETWEEN 述語

値 between 値 1 and 値 2：

インデクスのキー値が値 1 以上で，値 2 以下の範囲に含まれるかどうかを評価します。なお，not between は比較述語（<，>）と OR の組み合わせに変換して評価します。

```
例1）  where T1.C1 >= 'a' and T1.C1 <= 'z'
        →  KeyCnd：T1.C1 between 'a' and 'z'

例2）  where T1.C1 not between 'a' and 'z'
        →  KeyCnd：T1.C1<'a' OR T1.C1>'z'
```

(f)　比較述語

値{=，<，<=，>，>=，<>}値：

インデクスのキー値が比較述語（=，<，<=，>，>=，<>）を満たすかどうかを評価します。

```
例1）  where T1.C1=T1.C2
      →  KeyCnd：T1.C1=T1.C2

例2）  where T1.C1=(select C1 from T2)
      →  KeyCnd：T1.C1=SUBQ(2)
```

(g)　構造化繰返し述語

array(...)[any](キー条件)：

構造化繰返し述語の探索条件の中で，サーチ条件で評価できる条件を取り除いてからキー条件で評価する条件を表示します。

```
例）  where array(T1.C1,T1.C2)[any]
        (T1.C1<'a' or (T1.C1>'z' and T2.C2='a'))
      →  KeyCnd：array(...)[any]
                (T1.C1<'a' or (T1.C1>'z' and T2.C2='a'))
```

(h)　SIMILAR 述語

値 similar to パターン文字列 [escape エスケープ文字]：

インデクスのキー値がパターンと一致するかどうかを評価します。

```
例）  where T1.C1 similar to '%a'
      →  KeyCnd：T1.C1 similar to '%a'
```

値 not similar to パターン文字列 [escape エスケープ文字]：

インデクスのキー値がパターンと一致しないかを評価します。

```
例）  where T1.C1 not similar to '%a¥_' escape '¥'
      →  KeyCnd：T1.C1 not similar to '%a¥_' escape '¥'
```

(3)　キー条件に使用する値

(a)　表名.列名

キー条件に列名を指定した場合に表示します。

表名が相関名の場合には，表名の代わりに相関名を（）で囲んで表示します。

(b)　(NEW ROW).列名

制約判定のために HiRDB が内部的に生成する副問合せ中で，外部キーへの挿入値，又は更新値を使用して絞り込む場合に表示します。

(c)　表名.列名[添字]

キー条件に繰返し列（添字は整数，又は ANY）を指定した場合に表示します。

表名が相関名の場合には，表名の代わりに相関名を（ ）で囲んで表示します。

(d)　定数

キー条件に定数を指定した場合に値を表示します。定数の表示形式については，マニュアル「HiRDB SQL リファレンス」を参照してください。

(e)　？（？番号）

キー条件に埋込み変数，又は？パラメタを指定した場合に表示します。？番号とは，SQL 文中にある埋込み変数，又は？パラメタに対して，１を開始値として左から出現順に番号を付けたものです。

(f)　SQL 変数名

キー条件に SQL 変数を指定した場合に表示します。

(g)　SQL パラメタ名

キー条件に SQL パラメタを指定した場合に表示します。

トリガ SQL 文中で新値相関名の列，及び旧値相関名の列を使用している場合，新値相関名の列は「(NEWROW).列名」，旧値相関名の列は「(OLDROW).列名」と表示します。

(h)　USER

キー条件に USER 値関数を指定した場合に表示します。

(i)　CURRENT_DATE

キー条件に CURRENT_DATE 値関数を指定した場合に表示します。

(j)　CURRENT_TIME

キー条件に CURRENT_TIME 値関数を指定した場合に表示します。

(k)　CURRENT_TIMESTAMP(p)

キー条件に CURRENT_TIMESTAMP 値関数を指定した場合に表示します（p=0，2，4，又は 6）。

(l)　(SUBQ(問合せ ID))

キー条件に集合演算を含まないスカラ副問合せ，又は行副問合せを指定した場合に表示します。（ ）の中には問合せ ID を表示します。

(m)　SUBQEX(問合せ式本体 ID)

キー条件に集合演算を含むスカラ副問合せ，又は行副問合せを指定した場合に表示します。（）の中には問合せ ID を表示します。

(n)　スカラ演算

キー条件にスカラ演算を指定した場合に表示します。

(o)　行値構成子

キー条件に行値構成子を指定した場合に表示します。

17.5.13　ORDER BY 処理方式の種類

(1)　HiRDB/シングルサーバの場合

HiRDB/シングルサーバの場合，次の処理方式があります。

SORT CANCEL BY INDEX[(LIMIT SCAN)]

ORDER BY のためのソート処理をしなくても，インデクスを検索することで ORDER BY に含まれる列をソートできる場合に，ORDER BY のソート処理を省略します。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，インデクスの検索は先頭 n 行だけになり，検索結果のうち先頭 n 行を求めます。

SORT CANCEL BY JOIN

ORDER BY に含まれる列をすべて結合条件に指定している場合，ORDER BY のためのソート処理をしなくてもマージジョインによるソート処理がされるため，ORDER BY のソート処理を省略します。

SORT CANCEL BY DISTINCT[(LIMIT SCAN)]

ORDER BY の対象の値式がすべて DISTINCT 処理によってソートされる場合，ORDER BY のソート処理を省略します。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，DISTINCT 処理結果のうち先頭 n 行を求めます。

SORT CANCEL BY GROUPING[(LIMIT SCAN)]

ORDER BY に含まれる列がすべてグループ化処理によってソートされる場合，ORDER BY のソート処理を省略します。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，グループ化処理結果のうち先頭 n 行を求めます。

LIST SORT

作業表を作成してソートすることで，ORDER BY の結果を求めます。

LIMIT SCAN

先頭から n 行の検索結果を取得する機能を使用した場合に表示することがあります。探索条件を満たしたすべての行を対象とするソート処理はしないで，限られた行だけをソートすることで ORDER BY の結果のうち先頭 n 行を求めます。

PRE-SORT JOIN

ORDER BY に含まれるすべての列又は値式が，結合する前の外表又は外側の結合表に含まれる場合，結合する前にソートをして，そのソート順のまま結合し ORDER BY の結果を求めます。

PRE-SORT SET OPERATION[(LIMIT SCAN)]

集合演算の処理の前に，問合せごとにソートをして，そのソート順のまま演算します。そして，ORDER BY 及び集合演算の結果を求める処理方式です。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，先頭 n 行を取得するまで集合演算をして，ORDER BY 及び集合演算のうち先頭 n 行を求めます。

SORT CANCEL FOR SAME VALUES

ORDER BY の対象の値式がすべてウィンドウ関数の結果であり，かつすべての行で同じ値であることが保証される場合，ORDER BY のソート処理を省略する処理方式です。

(2) HiRDB/パラレルサーバの場合

HiRDB/パラレルサーバの場合，次の処理方式があります。

SORT CANCEL BY INDEX[(LIMIT SCAN)]

ORDER BY のためのソート処理をしなくても，インデクスを検索することで ORDER BY に含まれる列をソートできる場合に，ORDER BY のソート処理を省略します。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，インデクスの検索は先頭 n 行だけになり，検索結果のうち先頭 n 行を求めます。

SORT CANCEL BY DISTINCT[(LIMIT SCAN)]

ORDER BY の対象の値式がすべて DISTINCT 処理によってソートされる場合，ORDER BY のソート処理を省略します。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，DISTINCT 処理結果のうち先頭 n 行を求めます。

SORT CANCEL BY GROUPING[(LIMIT SCAN)]

ORDER BY に含まれる列がすべてグループ化処理によってソートされる場合，ORDER BY のソート処理を省略します。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，グループ化処理結果のうち先頭 n 行を求めます。

LIST SORT

各バックエンドサーバで作業表を作成してソートすることで，ORDER BY の結果を求めます。

LIMIT SCAN

先頭から n 行の検索結果を取得する機能を使用した場合に表示することがあります。探索条件を満たしたすべての行を対象とするソート処理はしないで，限られた行だけをソートすることで ORDER BY の結果のうち先頭 n 行を求めます。

PRE-SORT JOIN

ORDER BY に含まれるすべての列又は値式が，結合する前の外表又は外側の結合表に含まれる場合，結合する前にソートをして，そのソート順のまま結合し ORDER BY の結果を求めます。

FLOATABLE SORT

複数のフロータブルサーバに転送し，作業表を作成してソートすることで ORDER BY の結果を求めます。

LIST SORT WITH SET OPERATION[(LIMIT SCAN)]

問合せごとの ORDER BY のためのソートはしないで，集合演算の処理のときに ORDER BY のソートをします。その後，そのソート順のまま演算して，ORDER BY 及び集合演算の結果を求める処理方式です。

先頭から n 行の検索結果を取得する機能を使用した場合，(LIMIT SCAN)を表示することがあります。この場合，先頭 n 行を取得するまで集合演算をして，ORDER BY 及び集合演算の結果のうち先頭 n 行を求めます。

SORT CANCEL FOR SAME VALUES

ORDER BY の対象の値式がすべてウィンドウ関数の結果であり，かつすべての行で同じ値であることが保証される場合，ORDER BY のソート処理を省略する処理方式です。

17.6　チューニングの考え方

pdvwopt の出力結果を基に，SQL 文，インデクス定義の変更をして，検索性能の向上を図ります。

チューニングをしてアクセスパスを変更する場合，データの特性によっては必ずしも性能が改善されるとは限りません。アクセスパスを変更する場合には必ず性能を評価するようにしてください。

17.6.1　チューニング手順の例

pdvwopt の出力結果に基づいてチューニングをする方法の例を次に示します。17.6.2～17.6.4 の観点を基にチューニングをしてください。

```
*****     Result of SQL Optimizer     *****
Version         : HiRDB/Parallel Server VV-RR 4BES
UAP Name        : pdsql   ----------------------> UAP名は正しいか？
UAP Source      : pdsql-21775
Authorization   : user1   ----------------------> 認可識別子名は正しいか？
--------------------------------------------------------------------------------
Section No      : 1
Optimize Time   : 2001-01-07 10:59:16.591727  ---> SQL文を実行した時間は正しいか？
Optimize Mode   : COST_BASE_2
SQL Opt Level   : 0x00000420(1056) = "PRIOR_NEST_JOIN"(32),"RAPID_GROUPING"(1024)
Add Opt Level   : 0x00000003(3) = "COST_BASE_2"(1),"APPLY_HASH_JOIN"(2)
  ------------> SQL最適化オプション，SQL拡張最適化オプションは最適か？※１
SQL             : select T1.C1,T1.C2 from T1,T2,T3
                       where T1.C1='a' and T1.C2 like '%A'
                         and T1.C1=T2.C1 and T1.C2=T2.C2 and T2.C3>=1995
                         and T1.C1=T3.C1 and T1.C2=T3.C2 order by T1.C2
                                              --> SQL文は正しいか？
Work Table      : 3
Total Cost      : 3314.567244
----- QUERY ID : 1 -----
Query Type      : QUERY LIST(SORT) {LLID1(2)}
Order by Mode   : FLOATABLE SORT
FLT Server      : 2 (ORDER BY)  2-CLM 1TO1
JOIN
 # Join  ID     : 1   ---------------------------> 表の結合順序は意図した通りか？
   L Table      :  T1
   R Table      :  T2
   Join Type    : 2-CLM NESTED LOOPS JOIN(INNER) BROADCAST(FROM L TO R)
 -------------> 内表の分割キーを変更して，BROADCAST以外にできないか？※２
   FLT Server   : 0
 # Join  ID     : 2
   L Table      :  T3
   R Join ID    :  1 LIST{JRLST(1)}
   Join Type    : 2-CLM HASH JOIN(INNER) 1-CLM HASH(FROM R TO L)
 -------------> 結合方式は意図した通りか？※３
   FLT Server   : 0
SCAN
 # Table Name   : T1
   Cost         : N (10000000ROW) {T-2962.358541,I-763.682244,AND-766.202330}
   RDAREA       : NON DIVISION (1RD/1BES) [0x07(7)] ALL
```

```
   Scan Type   : MULTI COLUMNS KEY SCAN
   Index Name  : X1 (2) (+C2,+C1)   --------------> 意図したインデクスを使用しているか？
                     SearchCnd : RANGE(CS-CE) [(MIN,'a    '),(MAX,'a    ')] (FULL SCAN)
                     ----------------------------> サーチ範囲が絞り込まれているか？※４
                     KeyCnd    : T1.C1='a    ' AND T1.C2 like '%A'
 # Table Name  : T2
   Cost        : N (20000000ROW) {T-1388.652368,I-408.252884,AND-2647.479838}
   RDAREA      : 1-CLM FIX HASH(HASH6) (2RD/2BES) [0x07(7),0x0d(13)] ALL
   Scan Type   : MULTI COLUMNS KEY SCAN
   Index Name  : X2 (3) (+C1,+C2,+C3)
                     SearchCnd : RANGE(CS-CE) [(T1.C1,T1.C2,1995),(T1.C1,T1.C2,MAX)]
 # Table Name  : T3
   Cost        : N (20000000ROW) {T-1621.172368}
   RDAREA      : 1-CLM FIX HASH(HASH6)(4RD/2BES)[0x07(7),0x08(8),0x0d(13),0x0e(14)]ALL
   Scan Type   : TABLE SCAN   ------------------> 意図したスキャンタイプになっているか？
```

注　この出力結果は一部編集しているため，実際の表示例とは異なります。

注※1

データベースの状態に最適な SQL 最適化オプション，SQL 拡張最適化オプションを指定しているか確認してください。

SQL 最適化オプション，SQL 拡張最適化オプションについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

注※2

ネストループジョイン時に転送方法が BROADCAST 転送，KEY RANGE PARTIAL BROAD CAST 転送，又は PARTIAL BROADCAST 転送となる場合は，ネストループジョインの内表の分割キーを変更して，BROADCAST，KEY RANGE PARTIAL BROAD CAST，及び PARTIAL BROADCAST 以外にすることで，性能が向上することがあります。

詳細については，「分割キーを使用したネストループジョイン」を参照してください。

＜変更前＞

```
 # Join  ID    : 1
   L Table     :  T1
   R Table     :  T2
   Join Type   : 2-CLM NESTED LOOPS JOIN(INNER) BROADCAST(FROM L TO R)
   FLT Server  : 0
```

↓　表 T2 の分割キーを C1（結合キー）にします。

＜変更後＞

```
 # Join  ID    : 1
   L Table     :  T1
   R Table     :  T2
   Join Type   : 2-CLM NESTED LOOPS JOIN(INNER) 1-CLM HASH(FROM L TO R)
   FLT Server  : 0
```

最もデータの偏りの少ない列を分割キーにしてください。データの偏りの少ない列がなく，かつ複数の列で結合している場合には，結合列に使用している列から，それらを組み合わせることでデータの偏りが少なくなる列を複数選択して，分割キーに指定してください。

注※3

T1 と T2 の結合結果の件数が絞り込まれている場合，T1，T2 の結合結果と T3 をネストループジョインにした方が高速に検索できます。そこで，T3 の列を含んでいる結合条件（T1.C1=T3.C1 and T1.C2=T3.C2）から，T3 の列をすべて含む複数列インデクス T3（C1，C2）を定義します。

詳細については，「結合検索に使用する表のインデクス定義」を参照してください。

＜変更前＞

```
# Join  ID    : 2
  L Table     :  T3
  R Join ID   :  1 LIST{JRLST(1)}
  Join Type   : 2-CLM HASH JOIN(INNER) 1-CLM HASH(FROM R TO L)
  FLT Server  : 0
                .
                .
                .
# Table Name  : T3
  Cost        : N (2000000000ROW)  {T-1621.172368}
  RDAREA      : 1-CLM FIX HASH (HASH6)  (4RD/2BES) [...] ALL
  Scan Type   : TABLE SCAN
```

↓　T3（C1，C2）にインデクスを定義します。

＜変更後＞

```
# Join  ID    : 2
  L Join ID   :  1
  R Table     :  T3
  Join Type   : 2-CLM NESTED LOOPS JOIN(INNER) 1-CLM HASH(FROM L TO R)
  FLT Server  : 0
                .
                .
                .
# Table Name  : T3
  Cost        : N (2000000000ROW)  {T-1621.172368,T-1581.852884}
  RDAREA      : 1-CLM FIX HASH (HASH6)  (4RD/2BES) [...] ALL
  Scan Type   : MULTI COLUMNS KEY SCAN
  Index Name  : X3 (2D) (+C1,+C2)
                   SearchCnd : AT [(T1.C1,T1,C2)]
```

注※4

インデクスの第 1 構成列に対する条件が範囲を形成できないため，インデクスの全範囲をサーチしてしまいます。インデクスの第 2 構成列に＝があるため，インデクスの第 1 構成列と第 2 構成列を逆にします。

詳細については，「インデクスの定義」を参照してください。

＜変更前＞

```
Index Name  : X1 (2) (+C2,+C1)
                  SearchCnd : RANGE(CS-CE) [(MIN,'a    '),(MAX,'a    ')] (FULL SCAN)
                  KeyCnd    : T1.C1='a    ' AND T1.C2 like '%A'
```

↓

T1（C2，C1）のインデクスを削除し，T1（C1，C2）のインデクスを定義します。

＜変更後＞

```
Index Name  : X1 (2) (+C1,+C2)
                  SearchCnd : RANGE(CS-CE) [('a    ',MIN),('a    '),MAX]
                  KeyCnd    : T1.C2 like '%A'
```

17.6.2　インデクス定義について

まず，会話型 SQL 実行ユティリティ（HiRDB SQL Executer）でインデクスの定義情報を調べます。

- **HiRDB SQL Executer がある場合**

  インデクスの一覧を取得したい場合には，HiRDB SQL Executer の INDEXES コマンドを使用します。

  インデクス名からインデクス構成列名を取得したい場合には，INDEXCLM コマンドを使用します。

  なお，GUI 版 HiRDB SQL Executer の場合は，［ディクショナリビュー］画面でも確認できます。［ディクショナリビュー］画面の参照方法については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

  詳細については，HiRDB SQL Executer のヘルプを参照してください。

- **HiRDB SQL Executer がない場合**

  HiRDB SQL Executer がない場合は，ディクショナリ表を検索して情報を取得します。ディクショナリ表の検索例を次に示します。なお，ディクショナリ表については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

1. 定義されているインデクスの一覧を取得したい場合は，ディクショナリ表の SQL_INDEXES を検索します。

```
SELECT INDEX_ID , INDEX_NAME , TABLE_NAME ,
       UNIQUE_TYPE , VALUE(ARRAY_TYPE,' ') , COLUMN_COUNT
  FROM MASTER.SQL_INDEXES
  WHERE TABLE_SCHEMA LIKE USER
  ORDER BY 1
```

2. インデクス名からインデクス構成列名などを取得したい場合は，ディクショナリ表の SQL_INDEX_COLINF と SQL_COLUMNS を結合検索します。

```
SELECT Y.INDEX_ORDER , X.COLUMN_ID , X.COLUMN_NAME ,
        VALUE(X.MAX_ELM,1) , Y.ASC_DESC
  FROM MASTER.SQL_COLUMNS X , MASTER.SQL_INDEX_COLINF Y
  WHERE X.TABLE_SCHEMA = Y.TABLE_SCHEMA
    AND X.TABLE_NAME = Y.TABLE_NAME
    AND X.COLUMN_NAME = Y.COLUMN_NAME
    AND Y.INDEX_NAME LIKE 'IDX1'
    AND Y.TABLE_SCHEMA LIKE USER
  ORDER BY 1
```

(1) インデクスの定義

(a) 表示の確認

1. Scan Type の情報を確認してください。TABLE SCAN，又は AND PLURAL INDEXES SCAN となる場合で，性能が余り良くないときにはインデクスを定義している列をすべて含む，複数列インデクスを定義することを検討してください。
2. Index Name の情報を確認して，意図したインデクスが使用されているかどうかを確認してください。
3. サーチ条件，キー条件から，インデクスが有効に絞り込まれているかどうか確認してください。インデクスの検索する範囲が有効に絞り込まれていない例を次に示します。
   - サーチ条件がない（サーチ条件に SearchCnd：NONE(FULL SCAN)が表示される）
   - サーチ条件の第 1 構成列の絞り込み範囲が，MIN～MAX までである（サーチ条件の絞り込み範囲の後ろに(FULL SCAN)が表示される）
   - インデクスのサーチする範囲が広い

(b) ベターユース

AND を使用して，複数の述語を指定した検索をすることが多い表には，それぞれの述語の列を構成列とする複数列インデクスを定義します。この場合，＝述語で組み合わせて指定することが多い列を，値の重複の少ないものから順にインデクス構成列として指定します。

(c) 理由

インデクスのサーチ範囲が狭くなるため，サーチ時間が短縮されます。

(d) 例

```
SELECT * FROM T1 WHERE C1=10 AND C2=30 AND C3 BETWEEN 10 AND 20
SELECT * FROM T1 WHERE C1=10 AND C2=30 AND C3 <= 15
SELECT * FROM T1 WHERE C1=20 AND C2=40 AND C4 = 60
             ↓
```

T1(C1,C2,C3)，T1(C1,C2,C4)のインデクスを定義します。

(e) 注意事項

1. インデクスの構成列数を増やし過ぎると，インデクス自体を検索する時間が長くなります。
2. 列を更新すると，インデクスのメンテナンスが発生します。したがって，更新が多い列は，できるだけインデクス構成列に含めないようにしてください。
3. インデクスを定義する数はできるだけ少なくしてください。

(2)　結合検索に使用する表のインデクス定義

(a)　表示の確認

Join Type がソートマージジョインで，大量データのソートがされていて性能が良くない場合，外表の探索条件が十分に絞り込めるのであれば，内表の結合列にインデクスを作成してネストループジョインにすると，効率的に検索できる可能性があります。

(b)　ベターユース

制限条件で絞り込んだ表との結合検索に使用する表には，結合列を第 1 構成列とするインデクスを定義します。

結合条件を複数（n 個）指定することが多い場合には，結合列が第 1 構成列から第 n 構成列となるインデクスを定義します。この場合，値の重複の少ない列を先に指定してください。さらに，結合条件以外の探索条件のある場合には，結合列の後ろ（第ｎ＋１構成列以降）に指定してインデクスを定義します。

(c)　理由

制限条件の指定された表を外表，結合列にインデクスの定義された表を内表としてネストループジョインをするため，結合にインデクスを使用でき，入出力回数を削減できます。

(d)　例

```
SELECT * FROM T1,T2
WHERE T1.C3=20 AND T1.C1=T2.C1 AND T1.C2=T2.C2 AND T2.C3>10
                    内表の結合列はT2(C1,C2)
             ↓
```

T2(C1,C2,C3)のインデクスを定義します。

(e)　注意事項

1. 結合条件で外表に制限条件がない場合，通常はネストループジョインになりません。
2. 内表，外表の両側に制限条件を指定している場合，ネストループジョインにならないことがあります。
3. ネストループジョインを優先的，又は強制的にしたい場合には，SQL 最適化オプションの「ネストループジョイン強制」，又は「ネストループジョイン優先」を指定してください。詳細については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

(3)　外結合時に使用するインデクス

(a)　表示の確認

1. 外結合（LEFT OUTER JOIN）時に，意図したインデクスが使用されているかどうか確認してください。

(b) ベターユース

次のようなインデクスは，外結合時には使用されません。

- FROM 句中の ON 探索条件に指定した，外表の列に対するインデクス
- WHERE 句に指定した，内表の列に対するインデクス

インデクスを使用して検索するためには，外表の条件は WHERE 句に，内表の条件は ON 探索条件に指定できるかどうか検討してください。

注　外結合時の外表，内表の区分を次に示します。

外表 LEFT OUTER JOIN 内表 ON ...

WHERE ...

(c) 理由

外表の列は，FROM 句中の ON 探索条件の真偽に関係なく，すべての行が取り出されます。そのため，FROM 句中の ON 探索条件に指定した外表の列に対する制限条件は，インデクスを使用した絞り込みをしません。

内表の列は，FROM 句中の ON 探索条件を評価した後に，NULL 値を補います。NULL 値を補った後に WHERE 句の条件を評価するため，WHERE 句に指定した内表の列に対する制限条件は，インデクスを使用した絞り込みをしません。

(d) 例

```
SELECT * FROM T1 LEFT JOIN T2 ON T1.C1=T2.C1 and T1.C2='a' and T2.C2='b'
  WHERE T1.C3='c' and T2.C4='d'
        ↓
```

下線のある列は，インデクスを使用して評価します。

(4) ORDER BY，GROUP BY を指定した検索に使用する表のインデクス定義

(a) 表示の確認

1 表の検索で ORDER BY 処理のためにソートをしている場合，インデクスが昇順，又は降順に作成されていることを利用して，ORDER BY 列がソートされて出てくるようにインデクス定義を変更することで，ソート処理をなくすことができます。

(b) ベターユース

＝述語（「列指定＝値指定」だけ），又は IS NULL 述語(「列指定 IS NULL」)の探索条件で絞り込んだ後，ORDER BY，又は GROUP BY を指定した 1 表の検索をする場合，次の二つの条件を共に満たすインデクスを定義してください。

- ＝述語，又は IS NULL 述語を指定した列指定が，第１構成列から第ｎ構成列まで連続して指定されている。
- GROUP BY，又は ORDER BY で指定する列が，その順番通りに第ｎ＋１構成列以降連続して指定されている。

(c) 理由

インデクスを利用してアクセスするデータを絞り込み，更にソート処理を省略できるため，CPU 時間，入出力時間を削減できます。

(d) 例

```
SELECT C2, C3 FROM T1
WHERE C1=10 AND C2= ? AND C3>10
ORDER BY C4 DESC,C5 ASC
        ↓
```

T1(C1 ASC,C2 ASC,C4 DESC,C5 ASC)，又は T1(C1 ASC,C2 ASC,C4 DESC,C5 ASC,C3 ASC)のインデクスを定義します。

(e) 注意事項

次の条件のどれかを満たす場合，ソート処理をなくすことはできなくなります。

- **HiRDB/シングルサーバ，HiRDB/パラレルサーバ共通の制限**
  - DISTINCT を指定している
  - 探索条件によって，インデクスを使用しない検索を選択している
  - T1 が分割されていて，サーバ内分割された非分割キーインデクス（Index Name のインデクス属性が"d"のインデクス）は，ソート処理を省略できません。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。
- **HiRDB/シングルサーバの制限**

  T1 が分割していて，CREATE TABLE で指定した分割キーがインデクス構成列の先頭から同じ順で含まれている場合（Index Name のインデクス属性が"d"のインデクス），ソート処理を省略できません。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。
- **HiRDB/パラレルサーバの制限**
  - INSERT 文−SELECT 文を実行する SQL である
  - 副問合せ中である
  - 集合演算を指定している
  - HAVING 句を指定している
  - FOR UPDATE，FOR READ ONLY を指定している

- T1 が分割されていて，CREATE TABLE で指定した分割キーがインデクス構成列の先頭から同じ順で含まれていて（Index Name のインデクス属性が"d"のインデクス），かつ同一バックエンドサーバの RD エリアを複数割り当てている場合には，ソート処理を省略できません。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。

(5) 集合関数 MIN，MAX を指定した検索をする表のインデクス定義

(a) 表示の確認

GROUP BY 指定がなく，選択式に集合関数の MIN(列)，MAX(列)を指定した検索の性能が悪い場合には，集合関数中の列指定にインデクスを定義することで，インデクスの最小値，又は最大値だけを参照するだけで集合関数の値を求められるようになります。この場合，Group by Mode は IMPLICIT MIN-MAX INDEX となります。

(b) ベターユース

次の二つのベターユースがあります。

- **探索条件を指定しない場合**

  集合関数の MIN，又は MAX のどちらか少なくとも一方を指定した検索に使用する表には，MIN，又は MAX を指定する列を第１構成列とするインデクスを定義してください。

- **＝述語（又は IS NULL 述語）の探索条件を指定する場合**

  ＝述語（又は IS NULL 述語）の探索条件（ｎ個）で絞り込んだ後に，集合関数の MIN，MAX を指定した検索に使用する表には，次の条件をすべて満たすインデクスを定義してください。

  - ＝述語（又は IS NULL 述語）で指定した列指定が，第１構成列から第ｎ構成列まで連続して指定する。
  - MIN，MAX に指定する列を第ｎ＋１構成列に指定する。
  - ほかの探索条件の列は，第ｎ＋２構成列以降に指定する。

(c) 理由

集合関数 MIN，又は MAX に対しては，インデクスの最小値，又は最大値を参照して結果を得るので，入出力回数，CPU 時間を削減できます。

(d) 例

- **探索条件を指定しない場合**

```
SELECT MIN(C1), MAX(C1) FROM T1
            ↓
```

  T1(C1)のインデクスを定義します。

- **＝述語（又は IS NULL 述語）の探索条件を指定する場合**

```
SELECT MIN(C1), MAX(C1) FROM T1 WHERE C2=10 AND C3=20 AND C4<30
            ↓
```

T1(C2,C3,C1,C4)のインデクスを定義します。

(e) 注意事項

次の条件のどれかを満たす場合，IMPLICIT MIN-MAX INDEX にはなりません。

- **HiRDB/シングルサーバ，HiRDB/パラレルサーバ共通の制限**
  - MIN，MAX 以外の集合関数を指定している。
  - MIN，MAX が複数ある場合に，集合関数の引数の列が異なる。
  - GROUP BY 句がある。
  - 探索条件にシステム定義スカラ関数，関数呼出し，又は IS_USER_CONTAINED_IN_HDS_GROUP のどれかを含む。
  - 探索条件にインデクス構成列以外の列を含む。
  - FLAT 指定を，集合関数 MAX 又は MIN の引数に指定している場合，探索条件に繰返し列を含む。
  - (b)のベターユースに該当するインデクスがない。
  - T1 が分割されていて，サーバ内分割された非分割キーインデクス（Index Name のインデクス属性が"d"のインデクス）を使用して検索しています。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。
- **HiRDB/シングルサーバの制限**
  - T1 が分割されていて，CREATE TABLE で指定した分割キーがインデクス構成列の先頭から同じ順で含まれている場合（Index Name のインデクス属性が"d"のインデクス）はソート処理を省略できません。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。
  - T1 が分割されていて，CREATE TABLE で指定した分割キーがインデクス構成列の先頭から同じ順で含まれるインデクス（Index Name のインデクス属性が"d"のインデクス）を使用して検索しています。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。
- **HiRDB/パラレルサーバの制限**
  - 集合演算を指定している。
  - INSERT 文−SELECT 文を実行する SQL である。
  - HAVING 句がある。
  - FOR READ ONLY を指定している。
  - T1 が分割されていて，CREATE TABLE で指定した分割キーがインデクス構成列の先頭から同じ順で含まれるインデクス（Index Name のインデクス属性が"d"のインデクス）を使用して検索していて，かつ同一バックエンドサーバの RD エリアを複数割り当てています。ただし，すべての分割キーに「列＝値指定」を指定している場合を除きます。

(6) NULL 値を除外キーとするインデクスの定義

(a) 表示の確認

次の項目が該当しないか確認してください。

- NULL 値の重複度が高いインデクス構成列に対して，サーチ条件種別が IS NULL となる検索をしていて性能が悪い。
- NULL 値の挿入，NULL 値への更新，NULL 値から別の値への更新，NULL 値の行の削除をしていて性能が悪い。

(b) ベターユース

NULL 値を除外キーとするインデクスを定義します。

(c) 理由

NULL 値を除外キーとするインデクスを定義すると，NULL 値を除いてインデクスを作成します。そのため，探索条件に IS NULL を指定した場合には，インデクスを使用しないで TABLE SCAN となります。ただし，IS NULL 以外の述語（=，between など）を指定した場合にはインデクスを使用します。

NULL 値の件数が多く，インデクスを使用してデータページをランダムにアクセスしている場合には，NULL 値を除外キーとするインデクスを定義して TABLE SCAN にした方が高速に検索できるようになります。

また，NULL 値をインデクスキー値から除外することで，インデクスメンテナンスのオーバヘッドとログ量を削減できます。

(d) 例

```
SELECT * FROM T1 WHERE C1 IS NULL
            ↓
```

NULL 値を除外キーとする T1(C1)のインデクスを定義します。

```
SELECT * FROM T1 WHERE C1 IS NULL（→ インデクスを使用しません）
SELECT * FROM T1 WHERE C1 ='a'（→ インデクスを使用します）
```

(7) その他

- 検索方法が TABLE SCAN となっている表で，排他待ちが多発している場合は，探索条件を指定する列にインデクスを定義すると，アクセスするデータを絞り込めるので排他待ちが少なくなる可能性があります。
- インデクスを使用した検索時に排他待ちが多発している場合，探索条件を満たした行，又はキー値に対してだけ排他を掛ける無排他条件判定を指定できるかどうか検討してください。無排他条件判定については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

- 検索方法が AND PLURAL INDEXES SCAN となっている場合に，複数のユーザが同時に同一表を更新すると，デッドロックになることがあります。この場合，システム定義の pd_work_table_option オペランドを変更することを検討してください。pd_work_table_option オペランドについては，マニュアル「HiRDB システム定義」を参照してください。
- アクセスパス表示ユティリティの結果から，使用されていないインデクスがある場合は，削除することをお勧めします。不要なインデクスがあると，更新処理で余分な負荷が掛かったり，余分にデータベース容量が必要になったりします。

17.6.3　インデクス検索について

(1)　探索条件に OR を指定した検索（その 1）

(a)　表示の確認

1 表検索の場合に，TABLE SCAN となって性能が悪くないか確認してください。

(b)　ベターユース

OR の両辺に同じ条件を指定している場合には，同じ条件を OR の外へ抜き出してください。

(c)　理由

- **1 表検索の場合**
  抜き出した述語の列にインデクスが定義してあれば，インデクス内のサーチ範囲を小さくし，インデクス内のサーチ時間を短縮できます。
- **結合検索の場合**
  結合条件を OR 演算子で結んでいる場合には，結合する表の直積（CROSS JOIN）をしてから，直積結果に対して結合条件を評価します。抜き出した述語が結合条件であれば，ネストループジョイン，又はマージジョインをするので，結合時のデータの突き合わせ回数を削減できます。さらに，結合列にインデクスが定義してあれば，結合にインデクスを使用できるため，入出力回数を削減できます。

(d)　例

OR は更に IN 述語に変換した方がよい場合があります。詳細については，「探索条件に OR を指定した検索（その 2）」を参照してください。

- **1 表検索の場合**

```
C1にインデクス定義
SELECT * FROM T1 WHERE (C1=10 AND C2=20) OR (C1=10 AND C2=30)
                  ↓
SELECT * FROM T1 WHERE C1=10 AND (C2=20 OR C2=30)
```

ただし，C1=10 が十分に絞り込めないで，(C1,C2)に複数列インデクスを定義する場合には，C1=10 を抜き出さないほうが性能が良い場合があります。

- **結合検索の場合**

```
SELECT * FROM T1, T2
    WHERE (T1.C1=T2.C1 AND T1.C2=10) OR (T1.C1=T2.C1 AND T2.C2=20)
                ↓
SELECT * FROM T1, T2 WHERE T1.C1=T2.C1 AND (T1.C2=10 OR T2.C2=20)
```

(2) 探索条件に OR を指定した検索（その 2）

(a) 表示の確認

探索条件に対して同一列に対する条件を OR で指定している場合に，サーチ条件がない，又はサーチ条件が有効でなくインデクスのサーチ範囲が絞れずに性能が悪くないか確認してください。

(b) ベターユース

HiRDB のバージョン 06-02 以降の場合，HiRDB のバージョンが 06-01 よりも前で単一列インデクスが定義されている場合，及び HiRDB のバージョンが 06-01 よりも前で複数列インデクスが定義されている場合とでベターユースが異なります。

- **HiRDB のバージョンが 06-02 以降の場合**

  OR 演算の両辺に，同じ列に対する＝述語を指定し，かつインデクスが定義されている場合には，IN 述語を使用してください。

- **HiRDB のバージョンが 06-01 よりも前で単一列インデクスが定義されている場合**

  OR 演算の両辺に，同じ列に対する＝述語を指定し，かつ列に単一列インデクスが定義されている場合には，IN 述語を使用してください。

- **HiRDB のバージョンが 06-01 よりも前で複数列インデクスが定義されている場合**

  複数列インデクスの第１構成列に定義した列に，IN 述語を指定している場合には，OR 演算を使用してください。

(c) 理由

- **HiRDB のバージョンが 06-02 以降の場合**

  単一列インデクス，及び複数列インデクスは，IN 述語内の値指定ごとにインデクスをサーチするため（サーチ条件種別の ATS，RANGES），インデクスのサーチ範囲を絞り込めます。そのため，CPU 時間，及び入出力回数を削減できます。

- **HiRDB のバージョンが 06-01 よりも前で単一列インデクスが定義されている場合**

  単一列インデクスの場合には，IN 述語内の値指定ごとにインデクスをサーチするため（サーチ条件種別の ATS），インデクスのサーチ範囲を絞り込めます。そのため，CPU 時間，及び入出力回数を削減できます。

- **HiRDB のバージョンが 06-01 よりも前で複数列インデクスが定義されている場合**

複数列インデクスの場合には，IN 述語内の値指定の最小値から最大値までサーチするため（ただし，値指定に定数以外が含まれる場合にはすべてのリーフページを参照します），インデクスのサーチ範囲が広くなります。

OR 演算子に変換すると，Scan Type が OR PLURAL INDEXES SCAN となる場合があります。この場合，OR 演算子の左辺と右辺で別々にインデクスをサーチするので，インデクスのサーチ範囲を絞り込めるため，入出力時間を削減できます。

(d) 例

- **T1(C1)に単一列インデクス，又は T1(C1,C2)に複数列インデクスが定義されている場合（HiRDB のバージョンが 06-02 以降のとき）**

```
SELECT * FROM T1 WHERE C1=10 OR C1=20
          ↓
SELECT * FROM T1 WHERE C1 IN (10,20)
```

- **T1(C1)に単一列インデクスが定義されている場合（HiRDB のバージョンは 06-01 よりも前のとき）**

```
SELECT * FROM T1 WHERE C1=10 OR C1=20
          ↓
SELECT * FROM T1 WHERE C1 IN (10,20)
```

- **T1(C1,C2)に複数列インデクス（C1 が第１構成列であること）が定義されている場合（HiRDB のバージョンは 06-01 よりも前のとき）**

```
SELECT * FROM T1 WHERE C1 IN (10,20)
          ↓
SELECT * FROM T1 WHERE C1=10 OR C1=20
```

(e) 注意事項

HiRDB のバージョンが 06-01 よりも前の複数列インデクスの場合でも，次の二つの条件を共に満たすときには，IN 述語を使用してください。

- IN 述語内の値指定が定数だけである。
- IN 述語内の値指定の最小値から最大値までのキー値数が少ない。

(3) 結合条件に OR を指定した結合検索

(a) 表示の確認

Join Type が CROSS JOIN となっている場合，内部的に直積処理をしています。結合条件に OR 演算子を指定している場合，OR の両辺を別々に指定した集合演算（UNION，又は UNION ALL）を用いた等価な SQL にすると，効率的に検索できる可能性があります。

(b) ベターユース

結合条件が OR で結ばれている場合は，OR の両辺を別々に指定した集合演算を使用します。

- **選択式に DISTINCT を指定している場合**

  UNION を使用します。
- **選択式に DISTINCT を指定していない場合**

  UNION ALL を使用します。

(c)　理由

結合条件を OR 演算子で結んでいる場合，結合する表の直積をしてから，直積結果に対して結合条件を評価します。

OR 演算子を分離すると，ネストループジョイン，又はマージジョインをするので，結合時のデータの突き合わせ回数を削減できます。

さらに，ネストループジョインをすると，結合にインデクスを使用できるので，入出力回数を削減できます。

(d)　例

- **DISTINCT 指定ありの場合**

```
SELECT DISTINCT * FROM T1, T2
    WHERE (T1.C1=T2.C1 OR T1.C2=T2.C2) AND T1.C3=10
            ↓
SELECT * FROM T1, T2 WHERE T1.C1=T2.C1 AND T1.C3=10
  UNION
SELECT * FROM T1, T2 WHERE T1.C2=T2.C2 AND T1.C3=10
```

- **DISTINCT 指定なしの場合**

```
SELECT * FROM T1, T2 WHERE (T1.C1=T2.C1 OR T1.C2=T2.C2) AND T1.C3=10
            ↓
SELECT * FROM T1, T2 WHERE T1.C1=T2.C1 AND T1.C3=10
  UNION ALL
SELECT * FROM T1, T2 WHERE T1.C2=T2.C2
    AND (T1.C1<>T2.C1 OR T1.C1 IS NULL OR T2.C1 IS NULL) AND T1.C3=10
```

(e)　注意事項

1. UNION を指定すると，内部的に各問合せ指定の結果の作業表を作成します。この場合，重複排除のためにソート処理をします。

   また，UNION ALL を指定した場合も，内部的に各問合せ指定の結果の作業表を作成します。ただし，この場合はソート処理をしません。
2. 結合列にインデクスが定義されていても，外側表が制限条件で絞り込まれていなければ，結合にインデクスを使用しません。結合列にインデクスが定義されている表が内側表になり，もう一方の表が外側表になります。
3. 「(d)　例」の DISTINCT 指定なしの場合の例は，T1.C1，T2.C1 が NOT NULL 列，又は NULL 値を挿入しない運用であれば，IS NULL 述語は必要ありません。また，選択式に DISTINCT を指定しても意図した結果を得られる場合は，集合演算を UNION ALL ではなく UNION にするようにしてください。

(4)　範囲述語の検索

(a)　ベターユース

AND 演算で同じ列の＞＝述語と＜＝述語を結んでいる場合，SQL 最適化処理をする前に BETWEEN 述語に変換します。この場合，WHERE 句の先に指定した述語から順に，BETWEEN 述語に変換していきます。

AND 演算で同じ列の＞＝述語と＜＝述語を結んだ条件と BETWEEN 述語は，HiRDB の内部では等価に扱われるため，どちらで指定しても同じになります。

＞＝述語と＜＝述語が範囲を形成しているかどうか確認してください。

(b)　例

- **次の二つの SQL 文は同じアクセスパスを選択します**

```
SELECT * FROM T1 WHERE C1 BETWEEN 10 AND 20
SELECT * FROM T1 WHERE C1>=10 AND C1 <= 20
```

- **HiRDB は内部的に次のような変換をします**

```
SELECT * FROM T1 WHERE C1<=50 AND C1>=20 AND C1<=30
                       ↓
SELECT * FROM T1 WHERE C1 BETWEEN 20 AND 50 AND C1<=30
```

(c)　注意事項

冗長な述語がある場合，データベースアクセス時に不要な条件判定がされます。冗長な条件はできるだけ削除するようにしてください。

(5)　GROUP BY 指定した検索

(a)　ベターユース

HiRDB/パラレルサーバで複数の列に対する GROUP BY 検索をする場合は，重複の少ない GROUP 化列を GROUP BY 句の先に指定してください。ただし，SQL 最適化オプションに 1024 を指定していない場合だけです。

(b)　理由

HiRDB/パラレルサーバで複数のフロータブルサーバを使用して GROUP BY 処理をする場合に，フロータブルサーバへデータを振り分けるために GROUP BY 句の先頭の列でハッシングしています。GROUP BY 句の先頭の列に重複が少ない場合，各フロータブルサーバへ均等にハッシングできるので，並列処理の効果が上がります。

(c)　例

```
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C1,C2
```

（C1 の重複数）＞（C2 の重複数）の場合は次のように変更します。

```
          ↓
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C2,C1
```

(d)　注意事項

SQL 最適化オプションに 1024 を指定している場合には，すべてのグループ化列を使用してハッシングするので，GROUP BY の順番を入れ替える必要はありません。ただし，「ORDER BY，GROUP BY を指定した検索に使用する表のインデクス定義」に該当する場合には，グループ化列を入れ替えないでください。

(6)　GROUP BY，ORDER BY の両方を指定した検索

(a)　ベターユース

- **すべての GROUP BY 列を構成列に含むインデクスがある場合（HiRDB/シングルサーバの場合）**
  GROUP BY 列をインデクス構成列順に指定します。
- **すべての GROUP BY 列を構成列に含むインデクスがない場合**
  ORDER BY，GROUP BY を両方指定した検索で，ORDER BY 列が GROUP BY 列にすべて含まれるときは，ORDER BY 列の指定順に GROUP BY 列を指定してください。

(b)　理由

- **すべての GROUP BY 列を構成列に含むインデクスがある場合（HiRDB/シングルサーバの場合）**
  GROUP BY 処理のためのソート処理を省略できます。この場合，ORDER BY 処理のためのソート処理はされるが，グループ化処理後のデータ（データ件数が少ない）に対してソート処理をするため，検索性能が良くなります。
- **すべての GROUP BY 列を構成列に含むインデクスがない場合**
  GROUP BY 句と ORDER BY 句の列指定の順が同じ場合，ORDER BY のためのソート処理を省略できます。

(c)　例

- **すべての GROUP BY 列を構成列に含むインデクスがある場合（HiRDB/シングルサーバ）**

```
インデクス　T1(C2, C1)
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C1,C2 ORDER BY C1,C2
          ↓
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C2,C1 ORDER BY C1,C2
```

- **すべての GROUP BY 列を構成列に含むインデクスがない場合**

```
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C1,C2 ORDER BY C2,C1
          ↓
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C2,C1 ORDER BY C2,C1

SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C1,C2 ORDER BY C2 DESC
```

```
                ↓
SELECT C1,C2,SUM(C3) FROM T1 GROUP BY C2,C1 ORDER BY C2 DESC
```

(d)　注意事項

GROUP BY 句に対して，「ORDER BY，GROUP BY を指定した検索に使用する表のインデクス定義」に該当するインデクスを定義できる場合，そちらを優先してください。

17.6.4　結合検索について

(1)　分割キーを使用したネストループジョイン

(a)　表示の確認

HiRDB/パラレルサーバで，Join Type が NESTED LOOPS JOIN となり，転送情報が BROADCAST，KEY RANGE PARTIAL BROAD CAST，又は PARTIAL BROADCAST と表示されている場合，ネストループジョインの内表の分割キーを変更することで，性能が向上することがあります。

表のデータ件数が少ない場合には影響は少ないですが，分割数が多い場合に効果があります。

(b)　ベターユース

結合検索時にネストループジョインの内表を分割する場合には，結合列が分割キーになるように表を定義すると，並列に処理ができます。ただし，フレキシブルハッシュ分割の場合は並列に処理できません。転送情報が数字-CLM HASH，数字-CLM KEY RANGE，1TO1，又は数字-CLM MULTIDIM になるようにしてください。

(c)　理由

ネストループジョインの内表がキーレンジ分割，又は FIX ハッシュ分割していて，内表の分割キーが結合列にすべて含まれている場合には，表の分割条件から内表のサーチ範囲を絞り込めます。そのため，入出力回数，CPU 時間を削減できます。

(d)　例

```
SELECT * FROM T1, T2 WHERE T1.C1=T2.C1 and T1.C1='a'
                ↓
CREATE TABLE T1 (C1 INT NOT NULL, C2 INT)
    FIX HASH HASH6 BY C1 IN (USER1,USER2,USER3,USER4)
CREATE TABLE T2 (C1 INT NOT NULL, C2 INT)
    FIX HASH HASH6 BY C1 IN (USER1,USER2,USER3,USER4)
```

(e)　注意事項

フレキシブルハッシュ分割を使用すると，データの格納位置が特定できないため，分割キーを変更しても常に BROADCAST となり，並列には実行できません。フレキシブルハッシュ分割を使用する必要がないの

であれば，FIX ハッシュ分割にすることをお勧めします。フレキシブルハッシュ分割と FIX ハッシュ分割については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

18 データベース複写ユティリティ（pdcopy）

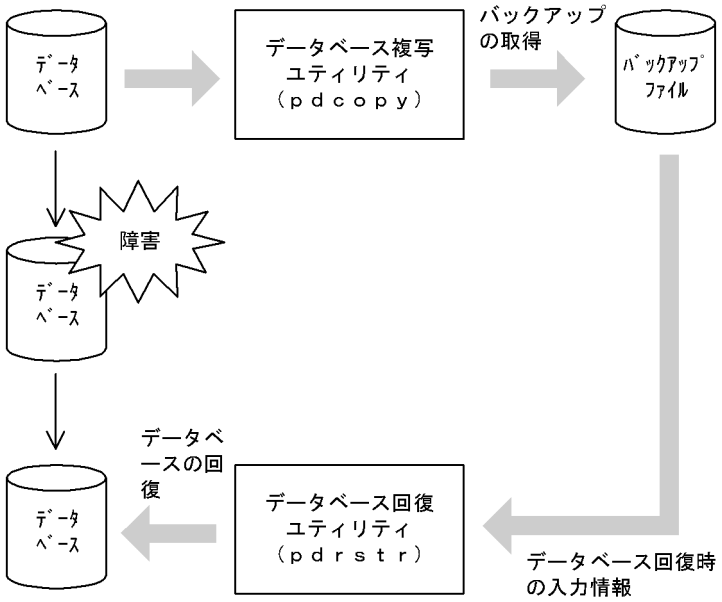
この章では，データベースのバックアップを取得するデータベース複写ユティリティ（pdcopy）の使い方について説明します。

18.1　データベース複写ユティリティの機能

実行者　HiRDB 管理者が実行できます

HiRDB 管理者は，データベースの障害に備えて定期的にデータベースのバックアップを取得する必要があります。データベースのバックアップはデータベース複写ユティリティで取得します。ディスク障害などでデータベースを回復する必要が生じた場合，取得したバックアップを入力情報にしてデータベース回復ユティリティ（pdrstr コマンド）でデータベースを回復します。データベース回復の概要を次の図に示します。

図 18–1　データベース回復の概要



18.2　バックアップを取得するために必要な知識

18.2.1　バックアップの取得単位

バックアップは次に示す単位で取得できます。バックアップの取得単位はデータベース複写ユティリティのオプションで指定します。バックアップの取得単位を次の表に示します。

表 18–1　バックアップの取得単位

バックアップの取得単位	説明	オプション
システム単位	システム内の全 RD エリアのバックアップを取得します。ユーザ用 RD エリア以外（例えば，マスタディレクトリ用 RD エリアなど）のバックアップも取得されます。ただし，リスト用 RD エリア及び一時表用 RD エリアはバックアップの対象となりません。	-a
ユニット単位※	ユニット下の全 RD エリアのバックアップを取得します。	-u
サーバ単位※	サーバ下の全 RD エリアのバックアップを取得します。	-s
RD エリア単位	RD エリアごとにバックアップを取得します。 RD エリア名を正規表現することで，グループ化した RD エリアのバックアップを取得できます。	-r

注※

HiRDB/パラレルサーバの場合に該当します。HiRDB/シングルサーバの場合は該当しません。

18.2.2　バックアップ取得モード

データベース複写ユティリティの **-M**オプションでバックアップ取得モードを選択します。バックアップ取得モードを次の表に示します。

表 18–2　バックアップ取得モード

バックアップ取得モード （-Mオプション の指定値）	モードの説明	データベースの 回復方法
x	バックアップ取得中，バックアップ対象 RD エリアの参照及び更新ができません。バックアップ取得対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。	バックアップだけでデータベースをバックアップ取得時点に回復できます。
r	バックアップ取得中，バックアップ取得対象の RD エリアの参照はできますが，更新はできません。	

バックアップ取得モード（-Mオプションの指定値）	モードの説明	データベースの回復方法
s	バックアップ取得中，バックアップ取得対象 RD エリアの参照及び更新ができます。バックアップ取得対象 RD エリアをバックアップ閉塞にする必要があります。	データベースを回復するときに，バックアップ及びバックアップ取得中のシステムログ※が必要になります。

注※

データベース複写ユティリティの処理結果リストに，RD エリアを回復するときに必要なシステムログファイルの名称及び世代番号が出力されます。データベース複写ユティリティの処理結果リストについては，「データベース複写ユティリティの処理結果リスト」を参照してください。

●注意事項

HiRDB の稼働中に全 RD エリアのバックアップを取得する場合（システム単位でバックアップを取得する場合）は，バックアップ取得モード（-M オプション）に r 又は s を指定してください。 x は指定できません。

理由：

x を指定する場合は，バックアップ取得対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。しかし，マスタディレクトリ用 RD エリアは閉塞かつクローズ状態にできません。このため，-M オプションに x を指定して，HiRDB の稼働中に全 RD エリアのバックアップを取得できません。

18.2.3　バックアップを取得するタイミング

バックアップは定期的に取得してください。ただし，次に示す作業をする場合は必ずバックアップを取得してください。このタイミングで取得していないと，障害が発生したときに最新の同期点までデータベースを回復できなくなります。

- ログレスモードの UAP 又はユティリティを実行する場合，その実行前後
- 更新前ログ取得モードの UAP 又はユティリティを実行した場合，その実行後
- RD エリアに対してデータベース構成変更ユティリティを実行した場合，その実行後
  なお，RD エリアの移動を実行する場合，その実行前にも取得してください。
- システムログを使用した回復を実行した後
- アンロードログファイルだけを使用して回復を実行する前
- HiRDB のバージョンアップをする場合，その実行前後
- プラグインの登録後（pdplgrgst コマンドの実行後）
- プラグインの使用を中止した後（pdplugin オペランドを削除した後）

- RD エリアをインメモリ化した場合，又はインメモリデータバッファとインメモリ RD エリアの同期を取った場合，その実行後

また，次に示す作業をする場合にバックアップを取得しておくと，障害が発生したときに回復に掛かる時間を短縮できます。

- システム用 RD エリアのバックアップを取得する必要がある処理の実行後

  システム用 RD エリアのバックアップを取得する必要がある処理とは，マニュアル「HiRDB システム運用ガイド」に記載されている「同時にバックアップを取得する必要がある RD エリア」で，マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，又はデータディクショナリ用 RD エリアのバックアップ取得が必要と記載されている処理です。
- 表のデータを大量更新した後

18.2.4　同時にバックアップを取得する必要がある RD エリア

RD エリア単位にバックアップを取得する場合は注意が必要です。バックアップを取得する RD エリアは，処理対象の RD エリアだけではなく，その処理によって更新される RD エリア（例えば，データディクショナリ用 RD エリアなど）についてもバックアップを取得する必要があります。同時にバックアップを取得する必要がある RD エリアについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

18.2.5　バックアップファイルを格納するサーバマシン

バックアップファイルは，HiRDB が稼働するサーバマシンであればどこにでも作成できます。バックアップを取得した RD エリアと同じサーバマシンに作成する必要はありません。CMT などのデバイスが，ほかのサーバマシンにある場合に適用してください。

バックアップファイルを格納するサーバマシンは，データベース複写ユティリティのオプションで指定できます。

18.2.6　アンロードレスシステムログ運用をしている場合

アンロードレスシステムログ運用をしている場合，バックアップを取得するときに，**ログポイント情報ファイル**も同時に取得する必要があります。ログポイント情報ファイルを取得するには，データベース複写ユティリティの**-z** オプションを指定します。

アンロードレスシステムログ運用については，マニュアル「HiRDB システム運用ガイド」を参照してください。

18.2.7　差分バックアップ機能を使用してバックアップを取得する場合

差分バックアップ機能は，前回のバックアップ取得時点からの差分情報だけをバックアップとして取得する機能です。このため，バックアップの取得処理時間を短縮できます。データベースの容量が大きくてデータ更新量が少ない場合に，差分バックアップ機能の使用を検討してください。

差分バックアップ機能を使用する場合，-g，-d，-K，-o，及び-L オプションを指定します。

差分バックアップ機能の運用方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

なお，インメモリ RD エリアに対しては，差分バックアップ機能を使用できません。

18.2.8　インメモリデータ処理を使用している場合

インメモリ RD エリアのバックアップを取得した場合，RD エリアの回復方法が通常の方法と異なることがあります。インメモリ RD エリアのバックアップ及び回復方法については，マニュアル「HiRDB バッチ高速化機能」を参照してください。

18.2.9　データベース複写ユティリティを実行するときの HiRDB の状態について

1. データベース複写ユティリティは，HiRDB が稼働中のときだけ実行できます。
2. データベース複写ユティリティは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. データベース複写ユティリティを実行する場合，バックアップ対象の RD エリアがあるユニット，及びバックアップファイルがあるユニットが開始されていなければなりません。また，バックアップ対象の RD エリアがあるサーバは，-M オプションに x 又は r を指定した場合は開始されていなくてもよいが，-M オプションに s を指定した場合は開始されていなければなりません。

18.3　使用例

ここでは，バックアップの取得例について説明します。説明する項目は次のとおりです。

- システム単位にバックアップを取得する場合
- RD エリア単位にバックアップを取得する場合
- サーバ単位にバックアップを取得する場合
- DAT 上のファイルへバックアップを取得する場合
- 差分バックアップ機能を使用してバックアップを取得する場合
- NetBackup を使用してバックアップを取得する場合

なお，各使用例中にでてくる pdcopy コマンドのオプションの詳細については，「コマンドの形式」を参照してください。

18.3.1　システム単位にバックアップを取得する場合

全 RD エリアのバックアップを一括して取得します。

```
pdcopy -m c:¥sysarea¥rdsys011¥mast01 -M r -p c:¥hirdb¥pdcopy¥list01
-z c:¥hirdb¥pdcopy¥logpoint01 -b c:¥hirdb¥pdcopy¥backup01 -a
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-M：バックアップ取得モードを指定します。

-p：データベース複写ユティリティの処理結果リストの出力ファイル名を指定します。

-z：ログポイント情報ファイル名を指定します。アンロードレスシステムログ運用をしていない場合は，指定する必要はありません。また，HiRDB/シングルサーバの場合は指定できますが，HiRDB/パラレルサーバの場合は指定できません。

-b：バックアップファイル名を指定します。

-a：全 RD エリアのバックアップを取得することを指定します。

18.3.2　RD エリア単位にバックアップを取得する場合

ユーザ用 RD エリア（RDAREA01，RDAREA02）のバックアップを取得します。

```
pdcopy -m c:¥sysarea¥rdsys011¥mast01 -M r
-p c:¥hirdb¥pdcopy¥list01 -b c:¥hirdb¥pdcopy¥backup01 -r RDAREA01,RDAREA02
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-M：バックアップ取得モードを指定します。

-p：データベース複写ユティリティの処理結果リストの出力ファイル名を指定します。

-b：バックアップファイル名を指定します。

-r：バックアップを取得する RD エリア（RDAREA01，RDAREA02）の名称を指定します。

18.3.3　サーバ単位にバックアップを取得する場合

バックエンドサーバ（bes1）下の全 RD エリアのバックアップを一括して取得します。

```
pdcopy -m c:¥sysarea¥rdsys011¥mast01 -M r -p c:¥hirdb¥pdcopy¥list01
-z c:¥hirdb¥pdcopy¥logpoint01 -b c:¥hirdb¥pdcopy¥backup01 -s bes1
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-M：バックアップ取得モードを指定します。

-p：データベース複写ユティリティの処理結果リストの出力ファイル名を指定します。

-z：ログポイント情報ファイル名を指定します。アンロードレスシステムログ運用をしていない場合は，指定する必要はありません。

-b：バックアップファイル名を指定します。

-s：バックエンドサーバ（bes1）下の全 RD エリアのバックアップを取得することを指定します。

18.3.4　DAT 上のファイルへバックアップを取得する場合

DAT 上のファイルへ全 RD エリアのバックアップを取得します。

```
pdcopy -m c:¥sysarea¥rdsys011¥mast01 -M r -p c:¥hirdb¥pdcopy¥list01
-z c:¥hirdb¥pdcopy¥logpoint01 -b ¥¥.¥tape -a
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-M：バックアップ取得モードを指定します。

-p：データベース複写ユティリティの処理結果リストの出力ファイル名を指定します。

-z：ログポイント情報ファイル名を指定します。アンロードレスシステムログ運用をしていない場合は，指定する必要はありません。また，HiRDB/シングルサーバの場合は指定できますが，HiRDB/パラレルサーバの場合は指定できません。

-b：バックアップファイル名を指定します。

-a：全 RD エリアのバックアップを取得することを指定します。

18.3.5　差分バックアップ機能を使用してバックアップを取得する場合

差分バックアップ機能を使用して，ユーザ用 RD エリア（RDAREA01，RDAREA02）のバックアップを取得します。なお，既に-d a 指定のフルバックアップを取得しているものとします。

```
pdcopy -m C:¥rdarea¥mast¥mast01 -M r -g backupg1 -b C:¥pdcopy¥backup02
-d d -K C:¥pdcopy¥admfile -o C:¥pdcopy¥rfile
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-M：バックアップ取得モードに参照可能モードを指定します。

-g：差分バックアップグループ名を指定します。(S)の指定は必要ありません。

-b：バックアップファイル名（差分バックアップファイル名）を指定します。

-d：バックアップ種別を指定します（差分バックアップを取得するため，d を指定します）。

-K：差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名を指定します。

-o：差分バックアップの履歴情報ファイル名を指定します。

18.3.6　NetBackup を使用してバックアップを取得する場合

NetBackup を使用して全 RD エリアのバックアップを一括して取得します。

```
pdcopy -m c:¥sysarea¥rdsys011¥mast01 -M r -p c:¥hirdb¥pdcopy¥list01
-z c:¥hirdb¥pdcopy¥logpoint01 -b host01:POLICY01 -a -k n
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-M：バックアップ取得モードを指定します。

-p：データベース複写ユティリティの処理結果リストの出力ファイル名を指定します。

-z：ログポイント情報ファイル名を指定します。アンロードレスシステムログ運用をしていない場合は，指定する必要はありません。また，HiRDB/シングルサーバの場合は指定できますが，HiRDB/パラレルサーバの場合は指定できません。

-b：バックアップファイル名として NetBackup のポリシー名を指定します。”ホスト名：ポリシー名”の形式で指定します。

-a：全 RD エリアのバックアップを取得することを指定します。

-k：バックアップファイルの種別を指定します。NetBackup を使用してバックアップを取得するので n を指定します。

18.4　コマンドの形式

18.4.1　オプションの形式

```
pdcopy -m 〔ホスト名：〕マスタディレクトリ用RDエリアの先頭のHiRDBファイル名
       〔-M {x｜r｜s}〕
       〔-p 処理結果出力ファイル名〕
        〔-i〕
        〔-j 排他待ち限界経過時間〕
        〔-z ログポイント情報ファイル名〕
        〔-J〕
        〔-w 休止時間，読み込み回数〕
        〔-W 実行監視時間〕
        〔-I メッセージ出力レベル〔，メッセージ出力間隔〕〕
        〔-f 制御文ファイル名※〕
```

注

必ず指定するオプション，又は指定することが多いオプションを背景色付き太字表示しています。なお，オプションのフラグ引数をコンマで区切る場合，コンマの前後に空白を入れないでください。空白を入れると，空白以降が無視されます。

注※

制御文ファイルに指定するオプションを次に示します。制御文ファイルに指定するオプションは，pdcopy コマンド中に直接指定できます。ただし，制御文ファイルと pdcopy コマンド中の両方に指定することはできません。どちらか片方だけに指定できます。

●制御文ファイルの内容

```
       -b {〔ホスト名：〕バックアップファイル名〔，バックアップファイル名〕… ｜〔ホスト名：〕ポリシー名}
        {-a
        ｜-u ユニット識別子〔，ユニット識別子〕…
        ｜-s サーバ名〔，サーバ名〕…
        ｜-r RDエリア名〔，RDエリア名〕…}
       〔-k {u｜i｜n}〕
        〔-g 差分バックアップグループ名〕
        〔-d バックアップ種別〕
        〔-K 差分バックアップ管理ファイルを格納するHiRDBファイルシステム領域名〕
        〔-o 差分バックアップの履歴情報ファイル名〕
        〔-L 差分バックアップ管理ファイルの容量〕
        〔-S {バックアップファイル初期容量，増分値｜（バックアップファイル初期容量，増分値）〔，（バックアップファイル初期容量，増分値）〕…}〕
```

●NetBackup 連携時のオプション指定可否

オプション	NetBackup
-m	◎
-M {x ｜ r ｜ s}	○

オプション		NetBackup
-p		○
-i		○
-j		○
-z		○
-J		○
-w		○
-W		○
-I		○
-f		○
-b		◎
{-a ｜-u ｜-s ｜-r}		◎※
-k	u	×
	i	×
	n	◎
-g		○
-d {a ｜ b ｜ c ｜ d}		○
-K		○
-o		○
-L		○
-S		×

（凡例）

◎：必ず指定します。

○：任意に指定できます。

×：指定できません。

注※

-a，-u，-s，又は-r の，どれか一つを指定する必要があります。

18.4.2　オプションの説明

(1)　-m〔ホスト名：〕マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名

～＜識別子：パス名＞((167 文字以内))

マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名を指定します。マスタディレクトリ用 RD エリアがあるホストの名称とパス名をコロン（：）で結んで指定してください。システム共通定義の pd_master_file_name オペランドに指定した名称を指定します。

```
（例）-m host01:C:¥sysarea¥rdsys02¥rdmast
```

＜規則＞

1. データベース複写ユティリティを実行したサーバマシン（pdcopy コマンドを入力したサーバマシン）にマスタディレクトリ用 RD エリアがある場合は，ホスト名を省略できます。この場合は，パス名だけを指定します。
2. ホスト名を指定する場合は，システム定義の pdunit オペランドの-x オプションで指定したホスト名を指定してください。
3. 系切り替え機能を使用している場合は，必ず現用系のホスト名を指定してください。
4. HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。

   C:¥hirdb¥rdarea¥master¥master01 の場合，C:¥hirdb¥rdarea¥master については大文字と小文字の区別をしません。master01 については大文字と小文字の区別をします。

(2)　-M {x｜r｜s}

バックアップ取得モードを指定します。バックアップ取得モードを次に示します。

-Mオプション の指定値	モードの説明	データベースの 回復方法
x （参照・更新不可能モード）	バックアップ取得中，バックアップ対象 RD エリアの参照及び更新ができません。バックアップ取得対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。	バックアップだけでデータベースをバックアップ取得時点に回復できます。
r （参照可能モード）	バックアップ取得中，バックアップ取得対象の RD エリアの参照はできますが，更新はできません。 インメモリ RD エリアの場合は，参照及び更新できます。	

-Mオプション の指定値	モードの説明	データベースの 回復方法
s （更新可能モード）	バックアップ取得中，バックアップ取得対象RD エリアの参照及び更新ができます。	データベースを回復するときに，バックアップ及びバックアップ取得直前のシンクポイントからのシステムログ※が必要になります。

注 1

-M x 指定でマスタディレクトリ用 RD エリアのバックアップを取得する場合は，pdstart -r コマンドで HiRDB を開始する必要があります。その後，データベース複写ユティリティで，マスタディレクトリ用 RD エリアのバックアップを取得します。

-M r 指定でマスタディレクトリ用 RD エリアのバックアップを取得する場合は，通常の pdstart コマンドで HiRDB を開始すると，pdcopy でバックアップが取得できます。

注 2

CREATE TABLE の LOB 列の定義で RECOVERY PARTIAL 又は RECOVERY NO を指定している場合（RECOVERY オペランド省略時も含まれます），その LOB 列が格納されているユーザ LOB 用 RD エリアについては-M s 指定のバックアップ取得はできません。また，オブジェクト格納用のデータディクショナリ LOB 用 RD エリアも，-M s 指定のバックアップ取得はできません。

注 3

-M s 指定をした場合，他トランザクションが更新中のページについては，更新をしたトランザクションが COMMIT を発行するまで排他待ちします。したがって，場合によっては pdcopy がタイムアウトすることがあります。

注 4

pdcopy を更新可能モードで実行している場合，対象 RD エリアをバックアップ閉塞にする必要があります。

注※

データベース複写ユティリティの処理結果リストに，RD エリアを回復するときに必要なシステムログファイルの名称及び世代番号が出力されます。データベース複写ユティリティの処理結果リストについては，「データベース複写ユティリティの処理結果リスト」を参照してください。

＜注意事項＞

1. HiRDB の稼働中に全 RD エリアのバックアップを取得する場合（システム単位でバックアップを取得する場合）は，バックアップ取得モード（-M オプション）に r 又は s を指定してください。 x は指定できません。

   理由：

   x を指定する場合は，バックアップ取得対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。しかし，マスタディレクトリ用 RD エリアは閉塞かつクローズ状態にできません。このため，-M オプションに x を指定して，HiRDB の稼働中に全 RD エリアのバックアップを取得できません。

2. ログレスモード又は更新前ログ取得モードの UAP，ユティリティ実行中は，-M s のバックアップは取得できません。また，ログレスモード又は更新前ログ取得モードの UAP，ユティリティ実行後のバックアップは，-M ｒ又は-M ｘで実行する必要があります。
3. 次のどちらかの場合に-M s でバックアップを取得するときは，事前に pdlogswap -w コマンドでシステムログをスワップさせてください。
   - pdmod を実行した場合（システムログファイルに構成変更処理が含まれないようにするため）
   - ログレスモードでの運用後（システムログファイルにログレスの処理が含まれないようにするため）
4. バックアップ対象 RD エリアにインメモリ RD エリアが含まれている場合は，-M s 指定でバックアップを取得できません。
5. 更新可能モードで差分バックアップが取得できるのは，差分バックアップを一切取得しないで累積差分バックアップだけ取得する運用の場合だけです。

(3)　-p 処理結果出力ファイル名

～<パス名>

データベース複写ユティリティの処理結果を出力するファイル名を指定します。

```
（例）-p C:¥hirdb¥pdcopy¥list01
```

＜規則＞

1. データベース複写ユティリティを実行したサーバマシン（pdcopy コマンドを入力したサーバマシン）のパス名を指定します。
2. このオプションを省略した場合，データベース複写ユティリティを実行したサーバマシンの次に示すディレクトリに処理結果出力ファイルを作成します。

-p オプションの指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	-p オプションに指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※　コマンド実行環境の環境変数の指定

出力先は KFPR26022-I メッセージに表示されます。

3. このオプションの指定に関係なくエラーメッセージはシステムログファイル及び標準出力に，最終処理結果は標準出力に出力されます。ただし，システムログファイルと標準出力に出力されるメッセージの順序は必ずしも同じではありません。

(4) -i

RD エリアの使用状況を処理結果出力ファイルに出力する場合に指定します。対象となる RD エリアは次のとおりです。

- データディクショナリ用 RD エリア
- データディクショナリ LOB 用 RD エリア
- ユーザ用 RD エリア
- ユーザ LOB 用 RD エリア

出力される情報を次に示します。

種別	情報	内容
ページ	ページ数	• 使用可能ページ数 • 未使用ページ数 • 使用中ページ数 ・インデクスの格納ページ数※1 ・表の格納ページ数※1
	比率	• 使用可能ページ数に対する未使用ページ数の比率 • 使用可能ページ数に対する使用中ページ数の比率 • 使用中ページ数に対するインデクスの格納ページ数の比率※1 • 使用中ページ数に対する表の格納ページ数の比率※1
領域長※1	長さ	• 使用中ページ内の空き領域の合計 • インデクスの格納ページ内の空き領域の合計 • 表の格納ページ内の空き領域の合計 • インデクスの格納ページ全体の領域長 • 表の格納ページ全体の領域長
	比率	• インデクスの格納ページ全体の領域長に対する空き領域の合計の比率 • 表の格納ページ全体の領域に対する空き領域の合計の比率
システム情報の乱れ率※2	比率	データディクショナリ LOB 用 RD エリア及びユーザ LOB 用 RD エリアのシステム情報の乱れ具合※3

注※1

データディクショナリ LOB 用 RD エリア及びユーザ LOB 用 RD エリアの場合は出力されません。

注※2

データディクショナリ LOB 用 RD エリア及びユーザ LOB 用 RD エリアの場合だけ出力されます。

注※3

1%以上の場合，アクセス性能が劣化します。この場合，データベース再編成ユティリティで再編成する必要があります。データディクショナリ LOB 用 RD エリアの場合は，ストアドプロシジャに関する

ディクショナリ表（SQL_ROUTINES，SQL_ROUTINE_RESOURCES，SQL_ROUTINE_PARAMS）を再編成します。

(5)　-j 排他待ち限界経過時間

～＜符号なし整数＞((0～200000))

排他待ち時間を監視する最大待ち時間を秒単位で指定します。

排他待ち時間とは，排他要求が待ち状態になってから解除されるまでの時間のことです。指定した時間内に待ち状態が解除されない場合，pdcopy はエラー終了します。

0 を指定した場合，排他待ち時間を監視しないで，待ち状態が解除されるまで待ち続けます。このオペランドを省略した場合は，システム定義の pd_lck_wait_timeout オペランドの値が仮定されます。

＜注意事項＞

1. -M r と同時に指定した場合，-j オプションの指定は無視され，排他待ちはしません。ほかのユーザが既に RD エリアに排他を掛けていた場合，pdcopy は即時エラー終了します。
2. -M s を指定した場合，ほかのユーザがページ解放をしているときは，そのユーザのトランザクションが終了するまで排他待ちになります。このような場合，排他待ち限界経過時間には，ページ解放をするトランザクションの実行時間より長い値を指定してください。なお，ページ解放は，主に次のような場合に発生します。
   - DROP TABLE
   - DROP INDEX
   - PURGE TABLE 文
   - インデクスがある列に対する INSERT 文又は UPDATE 文
   - インデクスがある列（重複キーあり）に対する DELETE 文
   - LOCK 文を実行した後の，DELETE 文又は行長が変わる UPDATE 文
   - LOB 列に対する INSERT 文，DELETE 文，及び UPDATE 文

(6)　-z ログポイント情報ファイル名

～＜パス名＞

このオプションはアンロードレスシステムログ運用をしている場合に指定するオプションです。

ログポイント情報を格納するファイル名を指定します。

```
（例）-z C:¥hirdb¥pdcopy¥logp01
```

＜規則＞

1. データベース複写ユティリティを実行したサーバマシン（pdcopy コマンドを入力したサーバマシン）のパス名を指定します。
2. このログポイント情報ファイルを pdlogchg コマンドで指定すると，ログポイント以前の不要となったシステムログファイルをアンロード済み状態にできます。また，このオプションの指定でバックアップファイルにもログポイント情報が格納されます。

＜注意事項＞

1. HiRDB/パラレルサーバの場合，バックアップを取得する RD エリア名には，同一サーバ下の RD エリアだけを指定してください。
2. バックアップ対象の RD エリアが配置されたサーバを起動しておく必要があります。
3. pdstart -r コマンドで HiRDB を開始してバックアップを取得する場合，-z オプションは指定できません。-z オプションを指定して全体のバックアップを取るには，pdstart で HiRDB を開始し，pdcopy -M r を指定してください。ただし，ログポイント情報ファイルはサーバごとに取得するため，HiRDB/パラレルサーバの場合は一度に全体のバックアップの取得はできません。また，マスタディレクトリ用 RD エリアは，閉塞，クローズ状態にはできません。そのため，pdstart で開始した場合，pdcopy -M x -a は指定できません。
4. インメモリ RD エリアがある場合はこのオプションを指定できません。

(7) -J

RD エリアの複写処理時にスキップ対象エラーを検知した場合に，pdcopy を終了させないで次の RD エリアの複写処理を続行するときに指定します。

-J オプションは，-g，-K，及び-d オプションとは同時に指定できません。同時に指定した場合は-J オプションの指定が無視されます。

スキップ対象エラーを次に示します。なお，エラーメッセージの前後に KFPR26061-W メッセージが出力されていない場合は，スキップされないことがあります。

スキップ対象エラー	スキップ対象エラーで出力されるメッセージ
• RD エリアとして定義していない名称を-r オプションに指定した場合 • リスト用 RD エリア又は一時表用 RD エリアを指定した場合	KFPR26006-E
-M x 指定時に，複写対象にマスタディレクトリ用 RD エリアが含まれている場合	KFPR26111-E
複写処理の前のファイルのオープン，クローズチェックでエラーを検知した場合	KFPR26012-E
-M s 指定時に，複写対象となる RD エリアがあるサーバが停止している場合	KFPR26015-E
表定義時に recovery partial 又は recovery no を指定した場合に，-M s を指定して複写対象にそのユーザ LOB 用 RD エリアが含まれているとき	KFPR26030-E
• -M x 指定時，複写対象の RD エリアが閉塞かつクローズ状態でない場合 • -M r 指定時，複写対象の RD エリアがインメモリ RD エリアの場合	KFPR26110-E

スキップ対象エラー	スキップ対象エラーで出力されるメッセージ
• -z オプション指定時，複写対象の RD エリアがインメモリ RD エリアの場合	

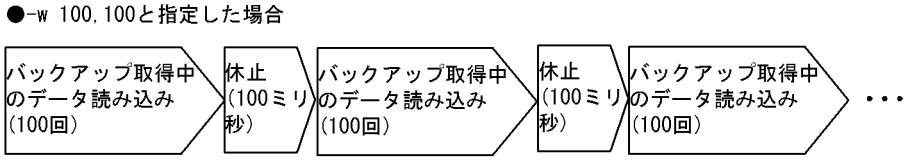
(8) -w 休止時間，読み込み回数

オンライン業務中に pdcopy を実行すると，大量のデータ転送を伴うため，CPU 使用率が高くなり，ほかのオンライン業務に影響を与えることがあります。-w オプションを指定すると，一定量のデータ読み込み処理を実行した後，一定間隔処理を休止させるため，ほかのオンライン業務への影響を少なくできます。

なお，このオプションを指定すると，CPU 使用率は低くなりますが，pdcopy の処理時間は長くなります。

-w オプション指定時の処理の概要を次の図に示します。

図 18–2 -w オプション指定時の処理の概要



休止時間 ～<符号なし整数>((10～60000))

データ読み込み処理中の休止時間を 10 ミリ秒単位に指定します（例：10，20，30...）。10 ミリ秒単位に指定していない場合，1 けた目が切り上げられます。

読み込み回数に指定した回数分データを読み込んだ後に，休止時間に指定した時間だけ処理を休止します。

読み込み回数 ～<符号なし整数>((1～10000))

処理を休止するまでのデータの読み込み回数を指定します。

ここで指定した回数分データを読み込んだ後，休止時間に指定した時間だけ処理を休止します。

<規則>

1. 休止時間，及び読み込み回数は，pdcopy が読み込むデータの総容量と，データ読み込み処理の総回数を考慮して決めてください。
2. HiRDB/パラレルサーバの場合，-w オプションの処理はサーバごとに実行されます。読み込み回数はサーバごとにカウントされ，休止処理はサーバごとに行われます。処理対象の RD エリアが，どのサーバにあるのかを確認してから，休止時間と読み込み回数を指定してください。

<-w オプションを指定した場合の処理時間の求め方>

1. 各 RD エリアのデータ読み込み回数を求めます。
   ・データディレクトリ用 RD エリアの場合
   ↑（4096×HiRDB ファイルシステムのレコード数）÷65536↑
   ＋（RD エリアを構成する HiRDB ファイル数−1）
   ・そのほかの RD エリアの場合

↑ページ長×（HiRDB ファイルシステムのレコード数
−（未使用セグメント数×セグメントサイズ））÷65536↑
＋（RD エリアを構成する HiRDB ファイル数−1）

2. データ読み込み処理の総回数を求めます。
   ・HiRDB/シングルサーバの場合
   処理対象の各 RD エリアのデータ読み込み回数の和
   ・HiRDB/パラレルサーバの場合
   処理対象 RD エリアのデータ読み込み回数の和が，各サーバの中で最も大きい値
3. 処理時間を求めます。
   ↓（2.で求めたデータ読み込み処理の総回数÷読み込み回数）↓
   ×休止時間（ミリ秒））
   ＋ pdcopy の処理時間

(9)　-W 実行監視時間

実行監視時間　～＜符号なし整数＞((0～3600))

データベース複写ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(10)　-I メッセージ出力レベル〔，メッセージ出力間隔〕

データベース複写ユティリティの稼働状況を確認するために指定します。次の二つのメッセージを出力します。

- KFPR26094-I：RD エリアごとのバックアップ処理を開始する際に表示します。
- KFPR26095-I：各 RD エリアに対し，KFPR26094-I メッセージ出力時点からメッセージ出力間隔に指定した時間が経過するごとに表示します。

上記メッセージを確認することで，無応答となっていないことを確認できます。

メッセージ出力レベル　～＜符号なし整数＞ {0 | 1}

稼働状況を確認するためのメッセージを出力するかどうかを指定します。

0：稼働状況を確認するためのメッセージを出力しません。

1：稼働状況を確認するためのメッセージを出力します。

メッセージ出力間隔　～＜符号なし整数＞((1～300))≪60≫

メッセージ出力間隔を，分単位で指定します。この指定によって，KFPR26095-I メッセージの出力間隔を変更できます。出力間隔を省略値から変更したい場合に指定してください。

(11)　-f 制御文ファイル名

～＜パス名＞

制御文ファイル名を指定します。

```
（例）-f C:¥hirdb¥pdcopy¥cont01
```

制御文ファイル中には，(11)以降のオプションを指定できます。なお，これらのオプションは，pdcopy コマンド中に直接指定することもできます。

＜注意事項＞

1. 制御文ファイルは，データベース複写ユティリティを実行するサーバマシン（pdcopy コマンドを入力するサーバマシン）に作成してください。
2. 制御文ファイル中には，バックアップファイル名，バックアップ対象 RD エリアのオプション，ファイル種別，及びボリューム名の組を，一組以上記述してください。ただし，これらのオプションの組は１行で記述する必要があります（1 行は 32,768 バイト以内）。これらの情報は最大 16 行記述できます。
3. 制御文ファイル中にはコメントを記述できます。ただし，コメントだけの行は記述できません。
4. -a オプションを指定して，差分バックアップ機能を使用する場合，制御文は 1 行だけ記述できます。

＜制御文ファイルの形式＞

```
 -b フラグ引数  {-a｜-u フラグ引数｜-s フラグ引数｜-r フラグ引数}
                〔-k フラグ引数〕〔-S フラグ引数〕
〔-b フラグ引数  〔 {-u フラグ引数｜-s フラグ引数｜-r フラグ引数} 〕
                 〔-k フラグ引数〕〔-S フラグ引数〕〕
〔-b フラグ引数  〔 {-u フラグ引数｜-s フラグ引数｜-r フラグ引数} 〕
                 〔-k フラグ引数〕〔-S フラグ引数〕〕
                       .
                       .
                       .
```

注 1

各オプションフラグについては，-b オプション指定時の各オプションの説明を参照してください。

注 2

複数行指定する場合は，1 行目に指定したオプションフラグだけ 2 行目以降も指定できます（ただし，-a オプションは除きます）。例えば，1 行目に-u オプションを指定した場合，2 行目以降は-u オプションだけ指定できます。

(12) -b {〔ホスト名：〕バックアップファイル名〔，バックアップファイル名〕...|〔ホスト名：〕ポリシー名}

～<識別子：パス名>((-k i 指定時は 167 文字以内))

バックアップファイル名，又は NetBackup で設定したポリシー名を指定します。

- **バックアップファイル名を指定する場合**

  バックアップファイルがあるホストの名称と，バックアップファイル名，装置記号名，又は装置グループ名をコロン（：）で結んで指定します。バックアップファイル名はパス名で，装置記号名及び装置グループ名は識別子で指定してください。

  ```
  （例）-b host01:C:¥hirdb¥pdcopy¥backup01
  ```

  また，バックアップファイルに対してテープ装置アクセス機能を使用できます。テープ装置アクセス機能については，「テープ装置アクセス機能」を参照してください。

- **NetBackup のポリシー名を指定する場合**

  NetBackup を使用する場合は，NetBackup クライアントがあるホストの名称とポリシー名をコロンで結んで指定します。なお，ハイフンを含んだポリシー名は指定できません。

  ```
  （例）-b host01:POLICY01
  ```

＜規則＞

1. データベース複写ユティリティを実行したサーバマシン（pdcopy コマンドを入力したサーバマシン）にバックアップファイルを作成する場合は，ホスト名を省略できます。この場合，パス名だけを指定します。
2. ホスト名を指定する場合は，システム定義の pdunit オペランドの-x オプションで指定したホスト名を指定してください。
3. データベース複写ユティリティを実行したサーバマシン（pdcopy コマンドを入力したサーバマシン）に NetBackup クライアントがある場合は，ホスト名を省略できます。
4. 制御文ファイルに指定する場合，各行に指定するバックアップファイル名を重複しないようにしてください。
5. 既にあるバックアップファイル名を指定してバックアップを取得した場合，既にあるバックアップファイルが上書きされます。
6. 系切り替え機能を使用しているホスト下のファイルにバックアップを取得する場合は，必ず現用系のホスト名を指定してください。バックアップファイルは，実行系のホスト下に作成されます。
7. 系切り替え機能を使用し，システムマネジャのユニット以外にある NetBackup クライアントを使用する場合，必ず現用系のホスト名を指定してください。使用する NetBackup クライアントは，実行系となります。

8. NetBackup を使用する場合に同一のポリシー名を使用してバックアップを取得するときは，取得対象の RD エリアを変えないでください。ポリシー名が同一なのに RD エリアが異なるバックアップがあると，回復できないことがあります。
9. -b オプションは-k オプションと次に示す関係があります。

-k オプションの指定	-b オプションの指定
-k u	ファイル名を指定できます。ファイル名は複数指定できます。
-k i	"HiRDB ファイルシステム領域名¥HiRDB ファイル名"を複数指定できます。HiRDB ファイルシステム領域は，あらかじめユティリティ専用領域として，pdfmkfs コマンドで作成しておかなければなりません。なお，HiRDB ファイルシステム名まで大文字，小文字の区別はしません。ただし，HiRDB ファイル名の大文字，小文字の区別はします。
-k n	NetBackup で設定したポリシー名を指定できます。ポリシー名は，最大 128 バイトの英数字で指定してください。 ホスト名は，NetBackup クライアントがあり，HiRDB があるホスト名を指定してください。 ポリシー名は，NetBackup の bpimagelist コマンドで参照できます。

＜注意事項＞

1. システム全体で大容量のファイル（システム全体で同時に 100 ギガバイト以上）を扱う場合，ファイルキャッシュが増大し，Windows のシステムリソース不足になることがあります。大容量のファイルを扱う場合，pdfmkfs コマンド（-k UTL）で作成した HiRDB ファイルシステム領域にバックアップファイルを作成すると，ファイルキャッシュを使用しないでアクセスするため，システムリソース不足を回避できます。ただし，ファイルキャッシュを使用しないでアクセスするため，性能は劣化します。
2. 同一のポリシー名を指定して pdcopy を複数実行する場合，回復時（pdrstr 実行時）には-U オプションの時刻指定でバックアップファイルを特定できます。指定するバックアップファイルの時刻は，pdcopy 実行時に出力される KFPR26071-I メッセージの日付，時刻を指定してください。

(13)　-a

システム単位でバックアップを取得する場合（全 RD エリアのバックアップを取得する場合）に指定します。ただし，リスト用 RD エリア及び一時表用 RD エリアはバックアップの対象となりません。

＜規則＞

1. 制御文ファイルに指定する場合，-a オプションを指定したときは，制御文は 1 行しか指定できません。

(14)　-u ユニット識別子〔，ユニット識別子〕...

～＜識別子＞((4 文字))

ユニット単位でバックアップを取得する場合（ユニット下の全 RD エリアのバックアップを取得する場合）に，バックアップ取得対象のユニット名を指定します。ただし，リスト用 RD エリア及び一時表用 RD エリアはバックアップの対象となりません。

＜規則＞

1. ユニット識別子を複数指定する場合は，重複しないようにしてください。
2. 制御文ファイルに指定する場合，ユニット識別子は制御文全体で一意にしてください。

(15)　-s サーバ名〔，サーバ名〕...

～＜識別子＞((最大 8 文字))

サーバ単位でバックアップを取得する場合（サーバ下の全 RD エリアのバックアップを取得する場合）に，バックアップ取得対象のサーバ名を指定します。ただし，リスト用 RD エリア及び一時表用 RD エリアはバックアップの対象となりません。

＜規則＞

1. サーバ名を複数指定する場合は，重複しないようにしてください。
2. 制御文ファイルに指定する場合，サーバ名は制御文全体で一意にしてください。

(16)　-r RD エリア名〔，RD エリア名〕...

～＜識別子＞((最大 30 文字))

RD エリア単位でバックアップを取得する場合に，バックアップを取得する RD エリアの名称を指定します。

＜規則＞

1. RD エリア名の指定方法については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。
   ただし，制御文ファイルに指定した RD エリア名が重複していた場合は次のとおり処理されます。
   - 直接指定した RD エリア名が重複した場合
     次のように指定すると，KFPR26007-E エラーになります（下線部分が重複部分です）。

```
-r LOB1,USER1 -b c:¥HiRDB¥pdcopy¥bkup1
-r DDIC,USER1 -b c:¥HiRDB¥pdcopy¥bkup2
```

   - 一括指定した RD エリア名が重複した場合
     次のように指定すると，最初に指定した値が有効になります（下線部分が重複部分です）。

```
-r LOB1,USER* -b c:¥HiRDB¥pdcopy¥bkup1…有効
-r DDIC,USER* -b c:¥HiRDB¥pdcopy¥bkup2
```

   - 直接指定した RD エリア名と一括指定した RD エリア名が重複した場合
     次のように指定すると，USER1 に対しては直接指定での指定値が有効になります。USER1 は c:¥HiRDB¥pdcopy¥bkup1 に取得されます（下線部分が重複部分です）。

```
-r LOB1,USER1 -b c:¥HiRDB¥pdcopy¥bkup1…有効
-r DDIC,USER* -b c:¥HiRDB¥pdcopy¥bkup2
```

2. リスト用 RD エリア，及び一時表用 RD エリアは指定できません。

(17) -k {u | i | n}

バックアップファイルの種別を指定します。

u：

通常のファイルにバックアップを取得する場合に指定します。

i：

HiRDB ファイルシステム領域にバックアップを取得する場合に指定します。実行系システムと待機系システムとでバックアップファイルを共用し，実行系システムで取得したバックアップを使用して待機系システムの回復をする場合に有効です。

n：

NetBackup を使用してバックアップを取得する場合に指定します。この場合，-b オプションには NetBackup で設定したポリシー名を指定してください。

(18) -g 差分バックアップグループ名

～＜英数字＞((1～30))

差分バックアップ機能を使用する場合，その単位となるグループの名称を指定します。

差分バックアップグループ名の最後に(S)を付けた場合，その時点から差分バックアップ管理を始めます。このときに，-k オプションの差分バックアップ管理ファイルが作成されます。同一名称の差分バックアップ管理ファイルがある場合は削除されるので，前回の差分バックアップグループのバックアップファイルは削除しておく必要があります。

差分バックアップ機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜規則＞

1. ユーザ LOB 用 RD エリア，データディクショナリ LOB 用 RD エリア，及びレジストリ LOB 用 RD エリアは，必ずフルバックアップが取得されます。
2. 一つの差分バックアップグループでは，バックアップを取得する RD エリアは変更できません。
3. 同じ差分バックアップグループのバックアップファイルは，同じホストに格納してください（-b オプションで指定するホスト名は一つにしてください）。
4. 差分バックアップファイルと差分バックアップ管理ファイルは，別々の HiRDB ファイルシステムに作成してください。
5. 差分バックアップ機能を使用する場合，-k オプションの指定によって，-b オプションに指定する差分バックアップファイル名の指定方法が変わります。

- -k u，及び-k i の場合，差分バックアップファイルごとに異なる名称を指定する必要があります。
- -k n の場合，差分バックアップファイルごとに異なるポリシー名を指定する必要はありません（異なる名称でも運用できますが，ポリシーを差分バックアップファイルごとに作成する必要があるため，運用が煩雑になります）。

(19) -d バックアップ種別

差分バックアップ機能を使用する場合，バックアップの種別を指定します。

a：フルバックアップを取得します。

b：最新のフルバックアップからの累積差分バックアップを取得します。

c：次のどちらかの最新のものからの累積差分バックアップを取得します。

- 前回取得した累積差分バックアップ
- 前回取得したフルバックアップ

d：前回取得したバックアップからの差分を取得します。

(20) -K 差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名

～<パス名>

差分バックアップ機能を使用する場合，差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域の名称を指定します。差分バックアップ管理ファイルの名称は，-g オプションで指定する差分バックアップグループ名と同じになります。

pdcopy 実行後には，pdfbkup コマンドで差分バックアップ管理ファイルのバックアップを取得する必要があります。差分バックアップ管理ファイルに障害が発生した場合，このバックアップを使用して pdfrstr コマンドで回復します。

差分バックアップ機能使用時に pdmod を実行した場合，フルバックアップを取得してください。このとき，-K オプションを指定して差分バックアップ管理ファイルも作成してください。

"差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名¥差分バックアップグループ名"が，167 文字以内になるようにしてください。

(21) -o 差分バックアップの履歴情報ファイル名

～<パス名>

差分バックアップ機能を使用する場合，差分バックアップの履歴情報を出力するファイルの名称を指定します。

履歴情報の内容については，「データベース複写ユティリティの処理結果リスト」を参照してください。

(22)　-L 差分バックアップ管理ファイルの容量

～<符号なし整数>((1～2046))≪1≫

差分バックアップ機能を使用する場合，差分バックアップファイルの容量をメガバイト単位で指定します。

容量は，HiRDB ファイルシステム領域作成時の容量（pdfmkfs コマンドの-n オプション指定値−5）より小さくしてください。

このファイルの増分は，最大 23 回できます。増分するときの容量は，ここで指定した値となります。

(23)　-S {バックアップファイル初期容量，増分値｜（バックアップファイル初期容量，増分値）〔,（バックアップファイル初期容量，増分値）〕...}

～<符号なし整数>((1～1048574))≪100,10≫

このオプションは，-k オプションで"i"が指定されたバックアップファイルに対して有効となります。

バックアップファイルの初期容量と，増分値（バックアップデータが初期容量を超えた場合に確保する容量）を，メガバイト単位で指定します。

＜規則＞

1. 初期容量は，HiRDB ファイルシステム領域作成時に，pdfmkfs コマンドの-n オプションで指定した値より小さくなければなりません。これは，システムが管理用に使用する領域が必要となるためです。
2. このオプションを省略した場合は，バックアップファイル初期容量に 100 メガバイト，増分値に 10 メガバイトが仮定されます。
3. バックアップファイル容量の増分は，最大 23 回できます。ただし，HiRDB ファイルシステム領域作成時に指定した最大増分回数を超えて増分はできません。
4. ファイルを複数指定した場合，それぞれのファイルに対しても，初期容量と増分値を設定できます。初期容量と増分値を括弧で囲んで指定したファイルの数分指定してください。
5. 初期容量と増分値の組が指定したファイルの数よりも少ない場合は，最後に指定された値が指定されなかったファイルに適用されます。初期容量と増分値の組が指定したファイルの数よりも多い場合は，多い分が無視されます。

（例）

```
-b file1,file2,file3,file4
-k i
-S (10,1),(1,1)
```

ファイル	初期容量	増分値	備考
file1	10	1	（10，1）の指定で設定される
file2	1	1	（1，1）の指定で設定される
file3	1	1	指定がないため最後の指定値 （1，1）が適用される
file4	1	1	

18.5 注意事項

1. 障害が発生している RD エリアのバックアップは取得しないでください。障害が発生した RD エリアのバックアップは不正であり，そのバックアップファイルはデータベースの回復に使用できません。
2. データベース複写ユティリティの最大同時実行数は，システム共通定義の pd_utl_exec_mode オペランドの指定値によって異なります。pd_utl_exec_mode=0 の場合は 32，pd_utl_exec_mode=1 の場合は pd_max_users の値となります。データベース複写ユティリティの同時実行数は指定するバックアップファイルの数に依存します。同時実行数の考え方は次のとおりです。

   考え方：

   - 一つのバックアップファイルを使うデータベース複写ユティリティを六つ同時に実行すると，同時実行数は，1×6 ＝ 6 になります。
   - 二つのバックアップファイルを使うデータベース複写ユティリティを四つ同時に実行すると，同時実行数は，2×4 ＝ 8 になります。
   - 上記のデータベース複写ユティリティが，すべて同時に動作している場合，同時実行数は 6 ＋ 8 ＝ 14 になります。

     なお，HiRDB/パラレルサーバで pd_utl_exec_mode=0 が設定された環境で，1 サーバマシンに一つのバックアップファイルを作成する場合は，ユティリティの同時実行数が pdcopy の実行数よりも多くなるため，pd_utl_exec_mode=0 のときの同時実行最大数を超えてエラーとなるおそれがあります。エラーとなった場合は，次のどちらかの方法で対処してください。

     ・pdcopy の実行数を少なくする

     ・pd_utl_exec_mode=1 を指定する
3. 制御文ファイルで複数のバックアップファイルを指定し，-r オプションで正規表現をした場合，バックアップ対象の RD エリアが複数のバックアップファイルに出力されないようにしてください。この場合，エラーとなります。ただし，バックアップファイルが一つの場合には，エラーにはなりません。例を次に示します。

   （例 1）

   ```
   -b バックアップファイル1 -r RDUSER*,RDMASTER
   -b バックアップファイル2 -r ..USER,RDDICTIONALY
   ```

   〔説明〕

   RDUSER で始まる RD エリアがバックアップファイル 1 及びバックアップファイル 2 に出力されるため，エラーとなります。

   （例 2）

   ```
   -b バックアップファイル1 -r RD*,RDMASTER
   ```

   〔説明〕

   RDMASTER は，RD*でも該当するが，バックアップファイルが一つなのでエラーにはなりません。
4. pdcopy のリターンコードを次に示します。

   0：正常終了

8：異常終了（一部の複写に失敗又はスキップしました）

次のどちらかの条件に該当する場合は，取得したバックアップを使用できます。

・-J オプションを指定している

・-b オプションを 1 つだけ指定している

上記に該当しない場合は，エラー要因を取り除き，再度コマンドを実行してください。

12：異常終了（すべての複写に失敗しました）

エラー要因を取り除き，再度コマンドを実行してください。

5. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdcopy の制御文ファイルには，BOM を持つファイルを使用できます。なお，pdcopy の制御文ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdcopy で出力されるファイルには BOM は付きません。
6. OS の時刻を遅らせる場合のバックアップ取得については，マニュアル「HiRDB システム運用ガイド」を参照してください。
7. 次の運用後にバックアップを取得する場合，回復時に必要となるアンロードログファイルを明確にするため，バックアップする前に pdlogswap コマンドを実行して HiRDB が使用するシステムログファイルを切り替えてください。
   - ログレスモードの運用
   - データベース構成変更（pdmod 実行）

18.6　データベース複写ユティリティの処理結果リスト

18.6.1　処理結果出力ファイルの内容

データベース複写ユティリティの処理結果出力ファイルの内容を次に示します。

```
pdcopy (VV-RR)  ***** DB COPY *****   2007-04-24 14:08:20 [1] utl3 [2]
----------------------------------------------------------------
*** DB BACKUP INFORMATION LIST ***       BACKUPMODE : SHARED [3]
<<LOG FILE INFORMATION>>
  SYSTEM ID      : utl13 [4]
  UNIT ID        : sds1  [5]
  TYPE           : sys   [6]
  SERVER NAME    : sds   [7]
  SERVER RUN ID  : 455d67c7 [8]
  LOG SERVER RUN ID : 455d67c7 [9]
  FILE NAME      : logfg01 [10]
  GENERATION NO  : 1 [11]
  BLOCK NO       : d2 [12]
  HEADER WRITE COUNTER : 3 [13]
  HEADER WRITE TIME : 455d68d2 [14]
<<BACKUP FILE INFORMATION>>
  VOL NAME       : vol1,vol2 [15]
  FILE NAME      : host1:c:¥hirdb¥pdcopy¥bkup01 [16]
                        c:¥hirdb¥pdcopy¥bkup02
  FILE KIND      : u [17]
  STARTED AT     : 2007-04-24 14:08:20 [18]
  ENDED AT       : 2007-04-24 14:08:21 [19]
<<RDAREA INFORMATION>>
  UNIT NAME      :un16 [20] SERVER NAME : sds [21]

  RDAREA NAME    :USER1[22][23]           [24]                       [25]
  RDAREA ID      :          4 ATTRIBUTE : USER          PAGE SIZE : 4096
  STARTED AT     : 2007-04-24 14:08:20 [26]
  ENDED AT       : 2007-04-24 14:08:21 [27]
  IN-MEMORY DB-SYNC TIME   :2007-04-24 14:04:26 [42]
 <RDAREA STATUS>      [28]
 TOTAL PAGE:           200 USED PAGE(RATE)   :        3   ( 1.5%) [29]
                           FREE SPACE        :       10.8         [30]
                           UNUSED PAGE(RATE) :      197   (98.5%) [31]
                               [32]
  INDEX PAGE(RATE) :            1( 33.3%) TOTAL SPACE :  4.0k      [33]
                           FREE SPACE(RATE)  :        3.8k( 94.7%) [34]
                               [35]
  DATA PAGE(RATE)  :            2( 66.7%) TOTAL SPACE :  8.0k      [36]
                           FREE SPACE(RATE)  :        7.0k( 87.6%) [37]
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt06 [38]
   EXTENT COUNT :   1 [40]
 2 c:¥hirdb¥area1¥rdmt07
   EXTENT COUNT :   1
<<RDAREA INFORMATION>>
  UNIT NAME   : un16        SERVER NAME : sds
```

```
  RDAREA NAME : USER2
  RDAREA ID   :            5 ATTRIBUTE  : USER             PAGE SIZE : 4096
  STARTED AT  : 2007-04-24 14:08:21
  ENDED AT    : 2007-04-24 14:08:21
  IN-MEMORY DB-SYNC TIME   :2007-04-24 14:04:26
 <RDAREA STATUS>
 TOTAL PAGE:           200 USED PAGE(RATE)   :         3   ( 1.5%)
                           FREE SPACE        :       10.8
                           UNUSED PAGE(RATE) :      197   (98.5%)
  INDEX PAGE(RATE) :            1( 33.3%) TOTAL SPACE :  4.0k
                           FREE SPACE(RATE)  :        3.8k( 94.7%)
  DATE PAGE(RATE)  :            2( 66.7%) TOTAL SPACE :  8.0k
                           FREE SPACE(RATE)  :        7.0k( 87.6%)
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt08
   EXTENT COUNT :   2
 2 c:¥hirdb¥area1¥rdmt09
   EXTENT COUNT :   2

<<RDAREA INFORMATION>>
  UNIT NAME   : un16        SERVER NAME : sds

  RDAREA NAME : USER3
  RDAREA ID   :            6 ATTRIBUTE  : USER            PAGE SIZE : 4096
  STARTED AT  : 2007-04-24 14:08:21
  ENDED AT    : 2007-04-24 14:08:21
  IN-MEMORY DB-SYNC TIME   :2007-04-24 14:04:12
 <RDAREA STATUS>
  TOTAL PAGE:        200 USED PAGE(RATE)   :          3   (  1.5%)
                         FREE SPACE        :         10.8
                         UNUSED PAGE(RATE) :        197   ( 98.5%)
  INDEX PAGE(RATE) :          1( 33.3%) TOTAL SPACE :    4.0k
                         FREE SPACE(RATE)  :          3.8k( 94.7%)
  DATA PAGE(RATE)  :          2( 66.7%) TOTAL SPACE :    8.0k
                         FREE SPACE(RATE)  :          7.0k( 87.6%)
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt10
   EXTENT COUNT :   2
 2 c:¥hirdb¥area1¥rdmt11
   EXTENT COUNT :   2
<<RDAREA INFORMATION>>
  UNIT NAME : un16     SERVER NAME : sds

  RDAREA NAME : USER_LOB1
  RDAREA ID   :         31  ATTRIBUTE : USER_LOB       PAGE SIZE : 8192
  STARTED AT  : 2007-04-24 14:08:21
  ENDED AT    : 2007-04-24 14:08:21
  IN-MEMORY DB-SYNC TIME   :2007-04-24 14:04:55
 <RDAREA STATUS>
  TOTAL PAGE:     75000  USED PAGE(RATE)   :      30000   ( 40.0%)
                         UNUSED PAGE(RATE) :      45000   ( 60.0%)

  DISORDERED LOB DIRECTORY RATIO :   0% [39]
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt12
   EXTENT COUNT :   2
 2 c:¥hirdb¥area1¥rdmt13
   EXTENT COUNT :   2
```

```
<<ERROR SKIPPED INFORMATION>>
  RDAREA NAME : USER04,USER05,USER06,USER07,USER08,
                USER09 [41]
-----------------------------------------------------------------
```

〔説明〕

1. データベース複写ユティリティを実行した日時（YYYY-MM-DD hh:mm:ss の形式）
   YYYY：年（西暦） MM：月 DD：日 hh：時 mm：分 ss：秒
2. HiRDB 識別子
3. バックアップ取得モード
   EXCLUSIVE：-M x 又は-M ｒ
   SHARED：-M s
4. HiRDB 識別子
5. ユニット識別子
6. ログ種別
   sys：システムログファイル
7. サーバ名
8. サーバラン ID（16 進 8 けた）
9. ログサーバラン ID（16 進 8 けた）
10. 該当するバックアップファイルを使用してデータベースを回復する場合に必要な，システムログファイルの名称
11. 10 のシステムログファイルの世代番号
12. 先頭ブロック番号（16 進 8 けた以内）
13. ヘッダ更新回数（16 進 8 けた以内）
14. 使用開始時刻（16 進 8 けた）
15. ボリューム名
    ＊＊＊＊＊＊が出力されます。
16. バックアップファイル名（ホスト名：パス名の形式で表示※）
    ファイル名を複数指定した場合は，使用したファイル名だけが表示されます。
    -k n を指定した場合，見出しの FILE NAME が POLICY NAME になります。この場合，NetBackup で設定したポリシー名が出力されます。
    注※ 系切り替え機能を使用している場合は，現用系のホスト名が表示されます。
17. バックアップファイルの種別（-k オプションに指定する値）
18. バックアップ取得開始時刻（形式は 1 と同じ）
19. バックアップ取得終了時刻（形式は 1 と同じ）

20. RD エリアがあるユニットの名称
21. RD エリアがあるサーバの名称
22. 複写した RD エリアの名称
23. 複写した RD エリアの番号
24. RD エリアの種類
    MASTERDIRECTORY：マスタディレクトリ用 RD エリア
    DATADIRECTORY：データディレクトリ用 RD エリア
    DATADICTIONARY：データディクショナリ用 RD エリア
    SYSTEM_LOB：データディクショナリ LOB 用 RD エリア
    USER：ユーザ用 RD エリア
    USER_LOB：ユーザ LOB 用 RD エリア
    REG：レジストリ用 RD エリア
    REG_LOB：レジストリ LOB 用 RD エリア
25. RD エリアのページ長（単位：バイト）
26. 該当する RD エリアのバックアップ取得開始時刻（形式は 1 と同じ）
27. 該当する RD エリアのバックアップ取得終了時刻（形式は 1 と同じ）
28. 該当する RD エリアの使用可能ページ数（表又はインデクスを格納できる RD エリア内のページ数）
29. 使用中ページ数（既に表又はインデクスを格納している RD エリア内のページ数）と，使用可能ページ数に対する使用中ページ数の比率
30. 使用中ページ内の空き領域長（単位：バイト）
    次に示す形式で出力されます。
    ZZZ9（0 以上，1 キロバイト未満の場合）
    ZZZ9.9k（1 キロバイト以上，1 メガバイト未満の場合）
    ZZZ9.9M（1 メガバイト以上，1 ギガバイト未満の場合）
    ZZZ9.9G（1 ギガバイト以上の場合）
    注　Z は，値が 0 の場合省略されます。9 は，値が 0 の場合でも省略されません。
31. 未使用ページ数（使用可能ページのうち，まだ使用されていない RD エリア内のページ数）と，使用可能ページ数に対する未使用ページ数の比率
32. インデクス格納ページ数（使用中ページのうち，インデクスを格納するために使用しているページ数）と，使用中ページ数に対するインデクス格納ページ数の比率
33. インデクス格納ページ全体の領域長（単位：バイト）
34. インデクス格納ページ内の空き領域を合計した長さ（単位：バイト）と，インデクス格納ページ内の空き領域の比率
35. 表格納ページ数（使用中ページのうち，表を格納するために使用しているページ数）と，使用中ページ数に対する表格納ページ数の比率

36. 表格納ページ全体の領域長（単位：バイト）
37. 表格納ページ内の空き領域を合計した長さ（単位：バイト）と，表格納ページ内の空き領域の比率
38. RD エリアを構成する HiRDB ファイルの名称
39. データディクショナリ LOB 用 RD エリア又はユーザ LOB 用 RD エリアのシステム情報の乱れ率 1%以上の場合，アクセス性能が劣化します。この場合，データベース再編成ユティリティで再編成をしてください。データディクショナリ LOB 用 RD エリアの場合は，ストアドプロシジャに関するディクショナリ表（SQL_ROUTINES，SQL_ROUTINE_RESOURCES，SQL_ROUTINE_PARAMS）を再編成します。
40. HiRDB ファイルのエクステント数
41. 処理をスキップした RD エリア名

    スキップした RD エリアが複数ある場合はコンマで区切り，5 個単位で改行します。スキップした RD エリアがない場合は＊＊＊＊＊が出力されます。エラー要因については，イベントログにメッセージが出力されているので参照してください。イベントログに出力されるメッセージの詳細は，「-J」を参照してください。
42. インメモリデータバッファとインメモリ RD エリアの同期を取った時刻

    最後に同期を取った時刻が表示されます。インメモリ RD エリアのバックアップを取得した場合にこの情報が表示されます。

注

<<LOG FILE INFORMATION>>は，-M オプションに s を指定した場合，又は-z オプションを指定した場合に出力されます。

4〜6，8，9，及び 12〜14 は，-z オプションを指定した場合に出力されます。

41 は-J オプションを指定した場合に出力されます。

18.6.2　差分バックアップの履歴情報ファイルの内容

データベース複写ユティリティの差分バックアップの履歴情報ファイルの内容を次に示します。

```
 pdcopy (VV-RR) ***** DB COPY *****  2000-10-26 21:30:20[1] utl3[2]

 c:¥hirdb¥pdcopy¥admfile[3]
 backupg1[4]
 2000-10-19 21:00:20[5]
 2000-10-26 21:18:20[6]
 RDAREA NAME :rdarea01,rdarea02[7]
[8]          [9]                  [10]    [11][12]           [13]
 a 2000-10-19 21:08:20,2000-10-19 21:09:25 r u host01: c:¥hirdb¥pdcopy¥backup¥backup01
 d 2000-10-20 21:01:20,2000-10-20 21:01:38 r u host01: c:¥hirdb¥pdcopy¥backup¥backup02
 c 2000-10-21 21:05:20,2000-10-21 21:06:10 r u host01: c:¥hirdb¥pdcopy¥backup¥backup03
 b 2000-10-22 21:01:20,2000-10-22 21:01:45 r u host01: c:¥hirdb¥pdcopy¥backup¥backup04
```

［説明］

1. pdcopy コマンドの実行日時
2. HiRDB 識別子
3. 差分バックアップ管理ファイルを格納している HiRDB ファイルシステム領域名
4. バックアップグループ名
5. バックアップグループの作成日時
6. バックアップグループの最終更新日時
7. バックアップグループに属する RD エリア名
8. バックアップ種別

   a：フルバックアップ

   b：最新のフルバックアップからの累積差分バックアップ

   c：前回取得した累積差分バックアップ又は前回取得したフルバックアップのどちらか最新のものからの累積差分バックアップ

   d：差分バックアップ
9. バックアップ取得開始日時
10. バックアップ取得終了日時
11. バックアップ取得モード

    x：参照・更新不可能モード

    r：参照可能モード

    s：更新可能モード
12. バックアップファイルの種別

    u：ファイル

    i：バックアップファイル用の HiRDB ファイルシステム領域
13. バックアップファイル名

    バックアップファイル名は，ホスト名：バックアップファイル名の形式で表示されます。

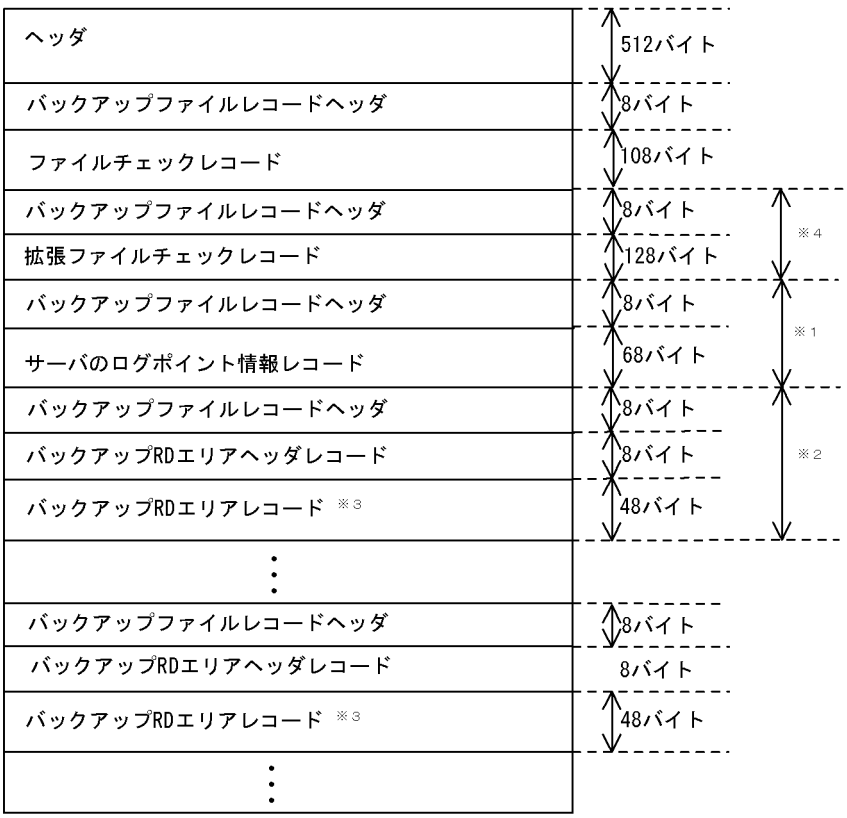
18.7　バックアップファイルの形式

データベース複写ユティリティで取得するデータベースのバックアップファイルの形式を次の図に示します。

- 図「データベースのバックアップファイルの構造」
- 図「バックアップファイルレコードヘッダのデータ形式」
- 図「ファイルチェックレコードのデータ形式」
- 図「バックアップ RD エリアヘッダレコードのデータ形式」
- 図「バックアップ RD エリアレコードのデータ形式」

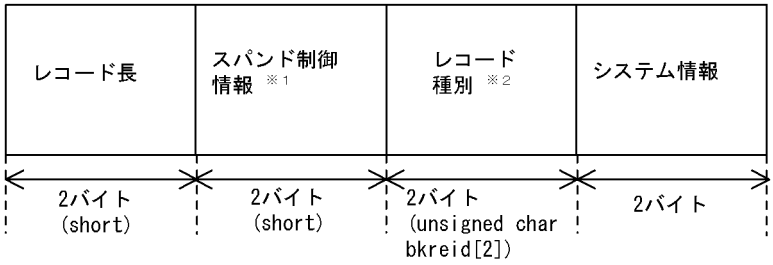
これらの情報から，バックアップ管理ツールなどを作成するときに，バックアップの世代管理ができるようになります。ただし，このバックアップファイルの形式は，HiRDB/シングルサーバのときだけ該当します。

図 18–3　データベースのバックアップファイルの構造



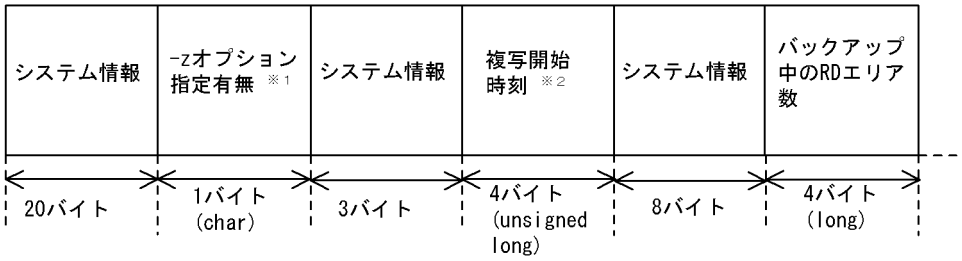
注※1　データベース複写ユティリティで-zオプションを指定したときだけ出力されます。-zオプションを指定していない場合には，下のレコードが上に詰められます。
注※2　長さが32,768バイトを超えた場合，超えたレコードからスパンドレコードとなります。このとき，32,768バイトのバウンダリからバックアップファイルレコードヘッダ，バックアップRDエリアヘッダレコード，バックアップRDエリアレコードが続きます。
注※3　バックアップRDエリアヘッダレコードに入っているRDエリア数分連続します。
注※4　差分バックアップ機能を使用している場合に有効となります。

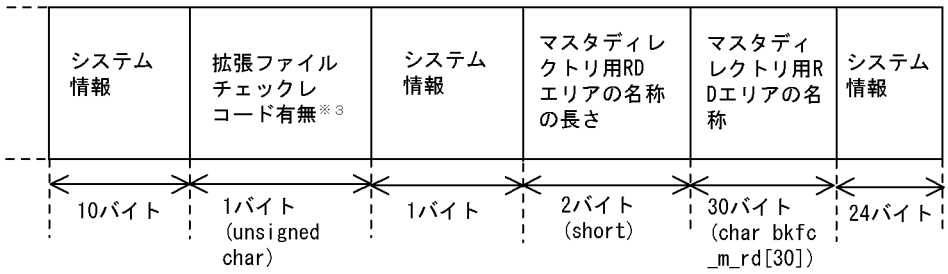
図 18–4　バックアップファイルレコードヘッダのデータ形式



注※1　スパンド制御情報を次に示します。
　　1 ：スパンド開始
　　2 ：スパンド中間
　　3 ：スパンド終了
　　0 ：スパンドなし
注※2　レコード種別を次に示します。
　　fc：ファイルチェックレコード
　　rd：バックアップRDエリアヘッダレコードとバックアップRDエリアレコード
　　sv：サーバのログポイント情報レコード
　　f2：拡張ファイルチェックレコード

図 18–5　ファイルチェックレコードのデータ形式





注※1　-zオプションを指定している場合は’z’が入ります。それ以外の場合には’0x00’が入ります。
注※2　1970年1月1日からの通算秒が，16進数で入ります（例：3372bfb3）。
注※3　拡張ファイルチェックレコードがある場合，0x01が入ります。

図 18–6　バックアップ RD エリアヘッダレコードのデータ形式

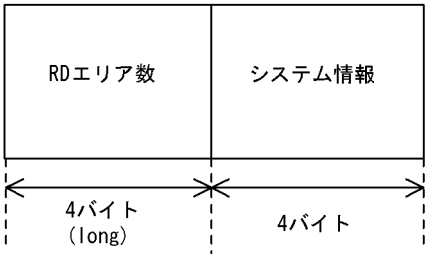
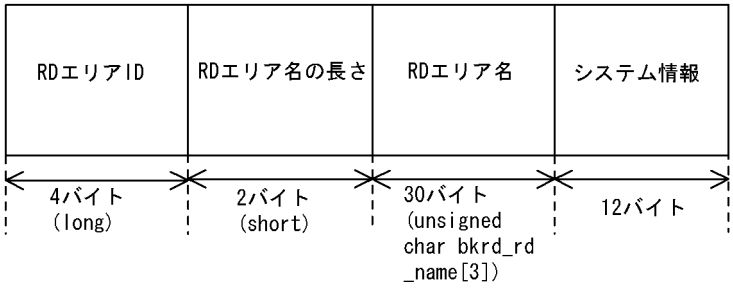


図 18–7　バックアップ RD エリアレコードのデータ形式



19 データベース回復ユティリティ（pdrstr）

この章では，データベース回復ユティリティ（pdrstr）の使い方について説明します。

19.1　データベース回復ユティリティの機能

実行者　HiRDB 管理者が実行できます

データベース回復ユティリティには次に示す機能があります。

- データベースの回復
- ログポイント情報ファイルの再作成

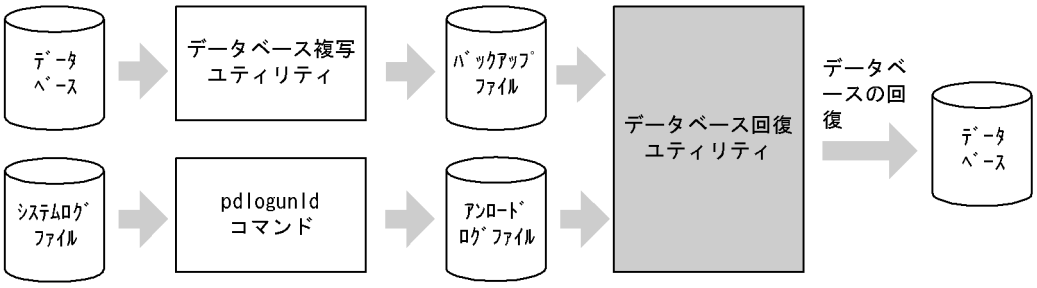
19.1.1　データベースの回復

ディスク障害など，データベースを回復する必要が生じた場合，データベース回復ユティリティでデータベースを回復します。このとき，次に示す入力情報が必要になります。

- バックアップファイル
- アンロードログファイル（アンロードレスシステムログ運用の場合はシステムログファイル）

データベース回復の概要（システムログをアンロードする運用の場合）とデータベース回復の概要（アンロードレスシステムログ運用の場合）を次の図に示します。

図 19–1　データベース回復の概要（システムログをアンロードする運用の場合）



〔説明〕

データベースは次に示す時点（状態）に回復できます。

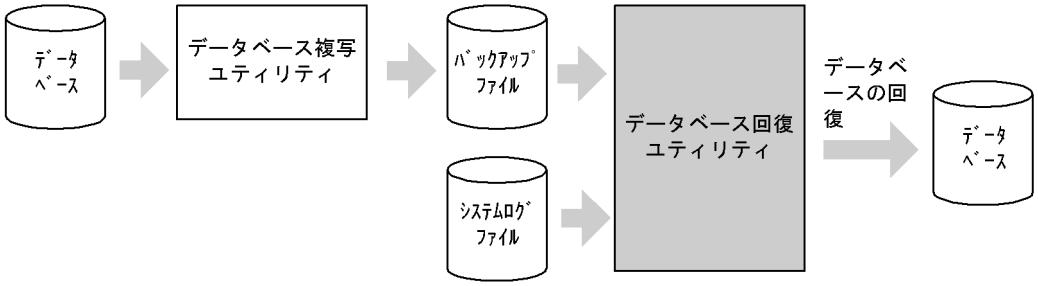
- **バックアップ取得時点**

  バックアップ取得時点に回復する場合は，アンロードログファイルは必要ありません。バックアップファイルだけが必要です。

- **最新の状態（最新の同期点）**

  最新の状態に回復する場合は，バックアップファイルのほかに，バックアップ取得以降に出力されたシステムログをアンロードしたアンロードログファイルが必要になります。

図 19–2　データベース回復の概要（アンロードレスシステムログ運用の場合）



〔説明〕

データベースは次に示す時点（状態）に回復できます。

- **バックアップ取得時点**

  バックアップ取得時点に回復する場合は，システムログファイルは必要ありません。バックアップファイルだけが必要です。

- **最新の状態（最新の同期点）**

  最新の状態に回復する場合は，バックアップファイルのほかに，バックアップ取得以降に出力されたシステムログを格納したシステムログファイルが必要になります。

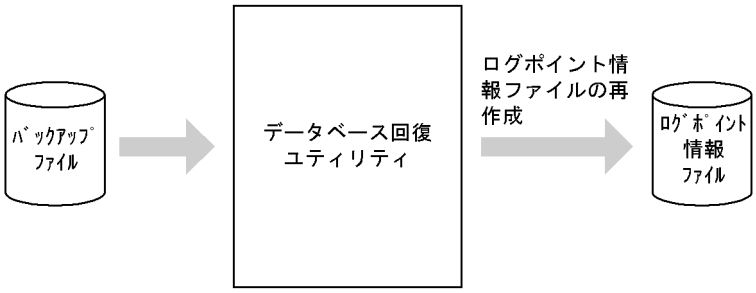
19.1.2　ログポイント情報ファイルの再作成

アンロードレスシステムログ運用をしている場合，ログポイント情報ファイルを保管しておく必要があります。ディスク障害などでログポイント情報ファイルが壊れた場合，データベース回復ユティリティでログポイント情報ファイルを再作成してください。このとき，次に示す入力情報が必要になります。

- バックアップファイル

ログポイント情報ファイルの再作成の概要を次の図に示します。

図 19–3　ログポイント情報ファイルの再作成の概要



19.2　データベースを回復するために必要な知識

19.2.1　データベースの回復単位

データベースは次に示す単位で回復できます。回復単位はデータベース回復ユティリティのオプションで指定します。データベースの回復単位を次の表に示します。なお，マスタディレクトリ用 RD エリアを含む RD エリアの回復を複数回の pdrstr コマンドで実行する場合は，最初に実行する pdrstr コマンドにマスタディレクトリ用 RD エリアを含めて回復してください。

表 19–1　データベースの回復単位

データベースの回復単位	説明	オプション
システム単位	全 RD エリアを回復の対象とします。ユーザ用 RD エリア以外（例えば，マスタディレクトリ用 RD エリアなど）も回復対象になります。	-a
バックアップファイル単位	バックアップファイル中の RD エリアを回復の対象とします。	-c
ユニット単位※	ユニット下の全 RD エリアを回復の対象とします。	-u
サーバ単位※	サーバ下の全 RD エリアを回復の対象とします。	-s
RD エリア単位	特定の RD エリアを回復します。RD エリア名を正規表現することで，グループ化した RD エリアを同時に回復できます。	-r

注※

HiRDB/パラレルサーバの場合に該当します。HiRDB/シングルサーバの場合は該当しません。

19.2.2　回復対象の RD エリアの状態

回復対象の RD エリア（マスタディレクトリ用 RD エリアは除きます）は，閉塞かつクローズ状態（pddbls コマンドの実行結果が CLOSE HOLD(CMD)又は CLOSE HOLD の状態）でなければなりません。

インメモリ RD エリアを回復する場合は，インメモリ RD エリアとインメモリデータバッファが次に示す状態である必要があります。

- インメモリ RD エリアの状態が閉塞かつクローズ状態である（pddbls コマンドの実行結果が CLOSE HOLD(CMD)又は CLOSE HOLD の状態）
- インメモリデータバッファがバッファ障害状態である

前記以外の状態の場合は，pdmemdb -k rels コマンドでいったんインメモリ化を解除した後に，pdrstr コマンドを実行してください。

19.2.3　バックアップ取得モードとの関係

データベース複写ユティリティの-M オプションに s を指定して取得したバックアップファイルの場合は，バックアップファイルだけでは回復できません。この場合，同時にアンロードログファイル又はシステムログファイルも使用する必要があります。

19.2.4　HiRDB の開始モードとの関係

HiRDB の開始モードと，回復できる RD エリアの関係を次に示します。

RD エリア種別	HiRDB の開始モード			
	-i	-r〔-t〕	-l	dbdestroy
マスタディレクトリ用 RD エリア	×	○	×	×
データディレクトリ用 RD エリア	○	△	×	○
データディクショナリ用 RD エリア	○	△	×	○
データディクショナリ LOB 用 RD エリア	○	△	×	○
ユーザ用 RD エリア	○	△	×	○
ユーザ LOB 用 RD エリア	○	△	×	○
リスト用 RD エリア	○	△	×	○
レジストリ用 RD エリア	○	△	×	○
レジストリ LOB 用 RD エリア	○	△	×	○

(凡例)

○：バックアップ取得時点，及び最新の同期点に回復できます。

△：バックアップ取得時点に回復できますが，最新の同期点には回復できません。

×：回復できません。

19.2.5　マスタディレクトリ用 RD エリアを回復する場合

RD エリアを回復するとき，回復対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。しかし，マスタディレクトリ用 RD エリアは閉塞かつクローズ状態にできません。このため，マスタディレクトリ用 RD エリアを回復する場合は，いったん HiRDB を終了して，その後 **pdstart -r** コマンドで HiRDB を開始する必要があります。

したがって，全 RD エリアを回復する場合はマスタディレクトリ用 RD エリアの回復も含まれるため，pdstart -r コマンドで HiRDB を開始する必要があります。

19.2.6　構成変更前に取得したバックアップファイルを使用して，バックアップだけの回復をする場合

構成変更（HiRDB ファイルの追加）前に取得したバックアップファイルを使用して，バックアップだけの回復をする場合は，RD エリアを構成する HiRDB ファイルをすべて削除してから回復してください。

19.2.7　最新の状態への回復

RD エリアを最新の状態に回復するには，バックアップ及びバックアップ取得以降に出力されたシステムログが必要です。したがって，現用のシステムログファイルに格納されているシステムログが必要になります。現用のシステムログファイルはアンロードできないため，**pdlogswap** コマンドで現用ファイルをいったんスワップしてください。その後，元現用ファイルの内容をアンロードしてください。

19.2.8　最新の状態以外に回復する場合

障害が発生した RD エリアを最新の状態に回復する場合は，その RD エリアだけが回復の対象になります。しかし，最新の状態以外（バックアップ取得時点又は任意の同期点）に回復する場合は，障害が発生した RD エリアだけではなく，関連する RD エリアも一緒に回復する必要があります。

例えば，表格納 RD エリアに障害が発生し，インデクス格納 RD エリアは正常の場合，表格納 RD エリアだけをバックアップ取得時点に戻すと，表データはバックアップ取得時点に戻り，インデクスデータは最新の状態となり，両者が不一致な状態になります。そのため，インデクス格納 RD エリアも表格納 RD エリアと一緒にバックアップ取得時点に回復する必要があります。

ポイント

関連する RD エリアは，まとめてバックアップ及び回復を行うことで，運用を容易にするとともに，運用ミスによる回復漏れを防止できます。ここでいう「関連する RD エリア」とは，データの整合性を維持するために，同時にバックアップする必要がある RD エリアを指します。

- 表を格納する RD エリアと，その表に定義されたインデクスを格納する RD エリア
- 表を横分割している場合，その表のデータを格納するすべての RD エリア
- 表を格納する RD エリアと，該当表に BLOB 列がある場合のユーザ LOB 用 RD エリア

上記に加え，参照制約やアプリケーション上のデータ依存関係など，ユーザデータ間の関連も合わせて考慮することを推奨します。

次のように，データ整合性に影響がない場合は，意図的に一部の RD エリアだけを回復しても問題ありません。

- 表だけを回復し，インデクスを再作成する場合

- 表がキーレンジで横分割されており，一部のデータだけを回復しても業務上問題がない場合

データの整合性を守る必要がある RD エリア（関連する RD エリア）については，マニュアル「HiRDB システム運用ガイド」の「同時にバックアップを取得する必要がある RD エリア」を参照してください。

19.2.9　アンロードログファイル又はシステムログファイルを使用してデータベースを回復した場合

アンロードログファイル又はシステムログファイルを使用してデータベースを回復した場合，回復完了後に必ず回復した RD エリアのバックアップを取得してください。ここでバックアップを取得しないと，その後障害が発生したときに，データベース回復ユティリティでデータベースを障害発生直前の同期点まで回復できなくなります。

アンロードログファイルを使用して回復する場合，必要なアンロードログファイルをすべて指定する必要があります。必要なアンロードログファイルが足りない場合，メッセージ（KFPR16203-E，KFPR16301-E）が出力され，エラー終了します。

19.2.10　アンロードレスシステムログ運用の場合の回復

アンロードレスシステムログ運用の回復では，システムログをアンロードしなくても，データベースの障害発生時に，直接システムログをデータベース回復ユティリティの入力情報にしてデータベースを回復します。この運用方法をアンロードレスシステムログ運用といいます。

アンロードレスシステムログ運用には，ログポイントという概念があります。データベースに障害が発生してデータベースを回復する場合，バックアップ取得時点よりも前のシステムログは必要ありません。このデータベースの回復に必要なシステムログと必要のないシステムログを区別する位置をログポイントといいます。ログポイントは，データベース複写ユティリティでバックアップを取得すると設定されます。

アンロードレスシステムログ運用の回復をする場合，-L 及び-z オプションを指定します。

アンロードレスシステムログ運用については，マニュアル「HiRDB システム運用ガイド」を参照してください。

19.2.11　差分バックアップ機能を使用してバックアップからの回復

差分バックアップ機能を使用して取得したバックアップから回復する場合，-g 及び-K オプションを指定します。

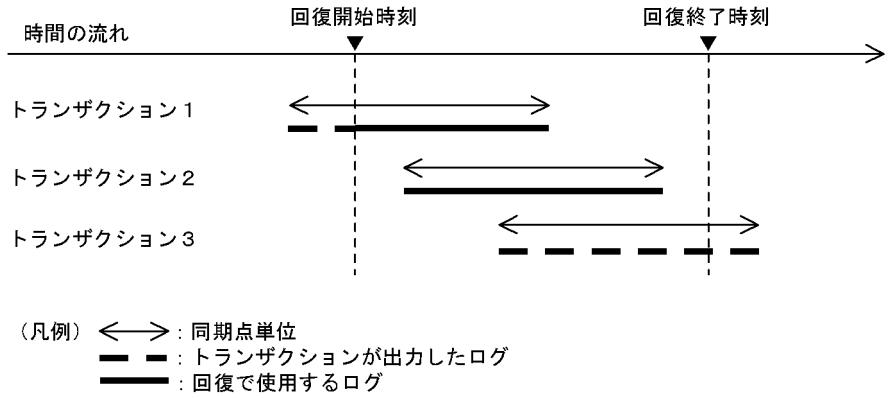
差分バックアップ機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

19.2.12　範囲指定の回復

アンロードログファイル又はシステムログファイルを指定して回復する場合，回復対象とする時間の範囲を指定できます。この範囲指定は，回復対象の RD エリアごとにも指定できます。範囲指定をした場合，指定した範囲内で同期点を迎えたトランザクションが出力するログだけが回復対象となります。範囲指定の回復をする場合に対象となるログの例を次に示す。範囲指定の回復については，マニュアル「HiRDB システム運用ガイド」を参照してください。

範囲指定の回復の概要を次の図に示します。

図 19–4　範囲指定の回復の概要



〔説明〕

トランザクション 1 のように，開始時点で実行中のトランザクションがある場合は，開始時点以降のログを使用して回復します。

トランザクション 3 のように，回復終了時刻までに同期点のないトランザクションについては回復しません。指定した範囲内で同期点を迎えたトランザクションについては，同期点までのログを使用して回復します。なお，トランザクション 3 のように，同期点のないトランザクションがある場合にはその旨ワーニングメッセージを出力します。

19.2.13　データベース回復ユティリティを実行するときの HiRDB の状態について

1. データベース回復ユティリティは，HiRDB が稼働中のときだけ実行できます。
2. データベース回復ユティリティは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. データベース回復ユティリティを実行する場合，回復対象の RD エリアがあるユニット，及びアンロードログファイル，システムログファイル，又はバックアップファイルを入力するユニットが開始されていなければなりません。また，回復対象の RD エリアがあるサーバは，バックアップファイルだけを入力とする場合は開始されていなくてもよいが，アンロードログファイル又はシステムログファイルを入力とする場合は開始されていなければなりません。

19.3　使用例

ここでは，データベースの回復例，及びログポイント情報ファイルの再作成例について説明します。説明する項目は次のとおりです。

- システム単位の回復をする場合
- RD エリア単位の回復をする場合
- サーバ単位の回復をする場合
- 差分バックアップ機能を使用したバックアップから回復をする場合
- NetBackup を使用したバックアップから RD エリアを回復する場合
- ログポイント情報ファイルを再作成する場合

なお，各使用例中にでてくる pdrstr コマンドのオプションの詳細については，「コマンドの形式」を参照してください。

19.3.1　システム単位の回復をする場合

(1)　全 RD エリアの回復

全 RD エリアをバックアップ取得時点に回復します。

```
pdrstr -m c:¥sysarea¥rdsys011¥mast01 -b c:¥hirdb¥pdrstr¥backup01
-p c:¥hirdb¥pdrstr¥list01 -a
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-b：バックアップファイル名を指定します。

-p：データベース回復ユティリティの処理結果リストの出力ファイル名を指定します。

-a：全 RD エリアを回復対象とする指定です。

(2)　DAT 上のバックアップからの全 RD エリアの回復

全 RD エリアをバックアップ取得時点に回復します。DAT 上のファイルへ取得されたバックアップを使用します。

```
pdrstr -m c:¥sysarea¥rdsys011¥mast01 -b ¥¥.¥tape
-p c:¥hirdb¥pdrstr¥list01 -a
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-b：バックアップファイル名を指定します。

-p：データベース回復ユティリティの処理結果リストの出力ファイル名を指定します。

-a：全 RD エリアを回復対象とする指定です。

19.3.2　RD エリア単位の回復をする場合

(1)　RD エリア単位の回復

ユーザ用 RD エリア（RDAREA01，RDAREA02）を最新の状態に回復します。

```
pdrstr -m c:¥sysarea¥rdsys011¥mast01 -b c:¥hirdb¥pdrstr¥backup01
-l c:¥hirdb¥pdrstr¥unld01 -p c:¥hirdb¥pdrstr¥list01 -w c:¥hirdb¥pdrstr¥sortwork
-r RDAREA01,RDAREA02
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-b：バックアップファイル名を指定します。

-l：アンロードログファイル名を指定します。バックアップの取得後にアンロードしたアンロードログファイル名を指定します。

-p：データベース回復ユティリティの処理結果リストの出力ファイル名を指定します。

-w：ソート用ワークディレクトリを指定します。

-r：回復対象の RD エリアを指定します。

(2)　アンロードレスシステムログ運用での RD エリアの回復

ユーザ用 RD エリア（RDAREA01，RDAREA02）を最新の状態に回復します。アンロードレスシステムログ運用をしています。

```
pdrstr -m c:¥sysarea¥rdsys011¥mast01 -b c:¥hirdb¥pdrstr¥backup01
-L -p c:¥hirdb¥pdrstr¥list01 -w c:¥hirdb¥pdrstr¥sortwork -r RDAREA01,RDAREA02
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-b：バックアップファイル名を指定します。

-L：システムログファイルを入力情報にする指定です。

-p：データベース回復ユティリティの処理結果リストの出力ファイル名を指定します。

-w：ソート用ワークディレクトリを指定します。

-r：回復対象の RD エリアを指定します。

19.3.3　サーバ単位の回復をする場合

バックエンドサーバ（bes1）下の全 RD エリアを最新の状態に回復します。HiRDB/パラレルサーバでアンロードレスシステムログ運用をしています。

```
pdrstr -m c:¥sysarea¥rdsys011¥mast01 -b c:¥hirdb¥pdrstr¥backup01
-L -p c:¥hirdb¥pdrstr¥list01 -w c:¥hirdb¥pdrstr¥sortwork -s bes1
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-b：バックアップファイル名を指定します。

-L：システムログファイルを入力情報にする指定です。

-p：データベース回復ユティリティの処理結果リストの出力ファイル名を指定します。

-w：ソート用ワークディレクトリを指定します。

-s：バックエンドサーバ（bes1）下の全 RD エリアを回復対象とする指定です。

19.3.4　差分バックアップ機能を使用したバックアップから回復をする場合

ユーザ用 RD エリア（rdarea01，rdarea02）を差分バックアップ取得時点に回復します。

```
pdrstr -m C:¥rdarea¥mast¥mast01 -g backupg1 -K C:¥pdcopy¥admfile -r rdarea01,rdarea02
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-g：差分バックアップグループ名を指定します。

-K：差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名を指定します。

-r：回復する RD エリア（rdarea01，rdarea02）の名称を指定します。

19.3.5　NetBackup を使用したバックアップから RD エリアを回復する場合

ユーザ用 RD エリア（RDAREA01，RDAREA02）をバックアップ取得時点に回復します。

```
pdrstr -m c:¥sysarea¥rdsys011¥mast01 -b host01:POLICY01 -k n -r RDAREA01,RDAREA02
```

〔説明〕

-m：マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。

-b：バックアップファイル名（NetBackup のポリシー名）を指定します。

-k：バックアップファイルの種別を指定します。NetBackup のポリシーを使用するため n を指定します。

-r：回復対象の RD エリアを指定します。

19.3.6　ログポイント情報ファイルを再作成する場合

ログポイント情報ファイルを再作成します。

```
pdrstr -b c:¥hirdb¥pdrstr¥backup01 -z c:¥hirdb¥pdrstr¥logpoint01
```

〔説明〕

-b：入力情報となるバックアップファイル名を指定します。

-z：再作成するログポイント情報ファイル名を指定します。

19.4　コマンドの形式

19.4.1　オプションの形式

必ず指定するオプション，又は指定することが多いオプションを背景色付き太字表示しています。

なお，オプションのフラグ引数をコンマで区切る場合，コンマの前後に空白を入れないでください。空白を入れると，空白以降が無視されます。

(1)　データベースの回復

```
pdrstr -m〔ホスト名：〕マスタディレクトリ用RDエリアの先頭のHiRDBファイル名
      {-b　{〔ホスト名：〕バックアップファイル名〔，バックアップファイル名〕…|〔ホスト
名：〕ポリシー名} }
      {-p 処理結果出力ファイル名}
      {-w ソート用ワークディレクトリ名}
      〔-y ソート用ワークバッファサイズ〕
      {-k　{u|i|n} }
      〔-U　{バックアップ検索条件開始時刻，バックアップ検索条件終了時刻|，バックアップ検索条
件終了時刻} 〕
      〔-g 差分バックアップグループ名〕
      〔-K 差分バックアップ管理ファイルを格納するHiRDBファイルシステム領域名〕
      {-Y 書き込みバッファサイズ}
      {-W 実行監視時間}
      {-I メッセージ出力レベル〔，メッセージ出力間隔〕}
      {-f 制御文ファイル名※}
      〔-J〕
```

注※

制御文ファイルに指定するオプションを次に示します。また，制御文ファイルに指定するオプションは，pdrstr コマンド中に直接指定できます。

●制御文ファイルの内容

```
    { {-l 〔ホスト名：〕アンロードログファイル名〔，アンロードログファイル名〕…
        |-L
        |-d 〔ホスト名：〕アンロードログファイル格納ディレクトリ名〔，アンロードログ
ファイル格納ディレクトリ名〕…}
      | {-a
        |-c
        |-u ユニット識別子〔，ユニット識別子〕…
        |-s サーバ名〔，サーバ名〕…
        |-r RDエリア名〔，RDエリア名〕…}
           〔-T　{回復開始時刻，回復終了時刻|回復開始時刻|，回復終了時刻} 〕 }
```

(2)　ログポイント情報ファイルの再作成

```
pdrstr -b　{〔ホスト名：〕バックアップファイル名|〔ホスト名：〕ポリシー名}
      {-k　{u|i|n} }
```

〔-U 〔バックアップ検索条件開始時刻，バックアップ検索条件終了時刻｜，バックアップ検索条件終了時刻〕〕
〔-W 実行監視時間〕
-z ログポイント情報ファイル名

(3) NetBackup 連携時のオプション指定可否

オプション		NetBackup
-m		◎
-b		◎
-l		○
-L		○
-d		○
-p		○
-w		○
-y		○
-k	u	×
	i	×
	n	◎
-U		○
-g		○
-K		○
-Y		○
-W		○
-I		○
-f		○
{-a ｜-c ｜-u ｜-s ｜-r}		◎※
-T		○
-z		○
-J		○

（凡例）

◎：必ず指定します。

○：任意に指定できます。

×：指定できません。

注※

-a，-c，-u，-s，又は-r の，どれか一つを指定する必要があります。

19.4.2　オプションの説明

(1)　-m〔ホスト名：〕マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名

～＜識別子：パス名＞

マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名を指定します。マスタディレクトリ用 RD エリアがあるホストの名称とパス名をコロン（：）で結んで指定してください。システム共通定義の pd_master_file_name オペランドに指定した名称を指定します。

```
（例）-m host01:C:¥sysarea¥rdsys02¥rdmast
```

＜規則＞

- データベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）にマスタディレクトリ用 RD エリアがある場合は，ホスト名を省略できます。この場合は，パス名だけを指定します。
- ホスト名を指定する場合は，システム定義の pdunit オペランドの-x オプションで指定したホスト名を指定してください。
- 系切り替え機能を使用している場合は，必ず現用系のホスト名を指定してください。
- HiRDB ファイルシステム領域名は大文字と小文字の区別をしませんが，HiRDB ファイル名は大文字と小文字の区別をします。

  C:¥hirdb¥rdarea¥master¥master01 の場合，C:¥hirdb¥rdarea¥master については大文字と小文字の区別をしません。master01 については大文字と小文字の区別をします。

(2)　-b {〔ホスト名：〕バックアップファイル名〔，バックアップファイル名〕...｜〔ホスト名：〕ポリシー名}

～＜識別子：パス名＞((-k i 指定時は 167 文字以内))

バックアップファイル名，又は NetBackup のポリシー名を指定します。

- **バックアップファイル名を指定する場合**

  バックアップファイルがあるホストの名称と，バックアップファイル名，装置記号名，又は装置グループ名をコロン（：）で結んで指定します。バックアップファイル名はパス名で，装置記号名及び装置グループ名は識別子で指定してください。

```
（例）-b host01:C:¥hirdb¥pdcopy¥backup01
```

また，バックアップファイルに対してはテープ装置アクセス機能を使用できます。テープ装置アクセス機能については，「テープ装置アクセス機能」を参照してください。

- **NetBackup のポリシー名を指定する場合**

  NetBackup を使用する場合は，NetBackup クライアントがあるホストの名称とポリシー名をコロンで結んで指定します。

```
（例）-b host01:POLICY01
```

＜規則＞

- データベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）にバックアップファイルがある場合は，ホスト名を省略できます。この場合，パス名だけを指定します。
- ホスト名を指定する場合は，システム定義の pdunit オペランドの-x オプションで指定したホスト名を指定してください。
- データベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）に NetBackup クライアントがある場合は，ホスト名を省略できます。
- 系切り替え機能を使用している場合は，必ず現用系のホスト名を指定してください。ただし，実際のバックアップファイル又は NetBackup クライアントは，実行系になければなりません。
- ログポイント情報ファイルの再作成の場合は，バックアップファイルが複数ファイル又は複数ボリュームの場合でも，先頭ファイル又は先頭ボリュームだけを指定します（複数ファイル又は複数ボリュームが指定されても先頭のものだけ参照します）。
- RD エリア単位に個々に取得したバックアップファイルを，複数個同時に指定すると，最初のバックアップファイルだけ認識されます。2 個目以降のバックアップファイルは認識されません。
- -b オプションに，-M s 指定で取得したバックアップファイルを指定した場合は，必ず-l，-L 又は-d オプションを指定してください。
- -b オプションは-k オプションと次に示す関係があります。

-k オプションの指定	指定できるバックアップファイル名
-k u	ファイル名を指定できます。バックアップの取得時，一つのバックアップファイルが複数のファイルにわたって取得された場合，バックアップを構成するすべてのファイル名を作成順に指定してください。
-k i	"HiRDB ファイルシステム領域名¥HiRDB ファイル名"で指定します。バックアップの取得時，一つのバックアップファイルが複数のファイルにわたって取得された場合，バックアップを構成するすべてのファイル名を作成順に指定してください。また，HiRDB ファイルシステム名までは大文字，小文字の区別はしませんが，HiRDB ファイル名の大文字，小文字の区別はします。
-k n	pdcopy で取得した NetBackup のポリシー名を指定します。ホスト名には，NetBackup クライアントがあり，HiRDB があるホスト名を指定してください。HiRDB が取り出す NetBackup のバックアップは最新のものだけです。

- バックアップは，pdcopy コマンドを実行してコマンドの戻り値が 0 又は 4 で取得したバックアップを指定してください。

(3) -l〔ホスト名：〕アンロードログファイル名〔，アンロードログファイル名〕...

～＜識別子：パス名＞

バックアップファイルの取得時点以降のアンロードログファイルの名称を指定します。アンロードログファイルがあるホストの名称とパス名をコロン（：）で結んで指定してください。

```
（例）-l host01:C:¥hirdb¥pdlogunld¥unld01
```

＜規則＞

- -l オプションを指定する場合は，-L，-c，及び-d オプションは指定できません。
- データベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）にアンロードログファイルがある場合は，ホスト名を省略できます。この場合，パス名だけを指定します。
- 系切り替え機能を使用している場合は，必ず現用系のホスト名を指定してください。ただし，実際のアンロードログファイルは実行系になければなりません。
- アンロードログファイルを HiRDB ファイルで作成した場合は，アンロードログファイル名には"HiRDB ファイルシステム領域名¥HiRDB ファイル名"を指定してください。このとき，"HiRDB ファイルシステム領域名¥HiRDB ファイル名"は 167 文字以内にしてください。

＜注意事項＞

- アンロードログファイルが複数ある場合，世代の古いものから順にアンロードログファイルの数だけ指定する必要があります。ただし，ホスト名は先頭に 1 回だけ指定してください。すなわち，すべてのアンロードログファイルは同一ホストになければなりません。
- アンロードログファイルを使用して回復する場合，必要なアンロードログファイルをすべて指定する必要があります。必要なアンロードログファイルが足りない場合，メッセージ（KFPR16203-E，KFPR16301-E）が出力され，エラー終了します。

(4) -L

アンロードレスシステムログ運用の場合に指定します。この指定をすると，システムログファイルが入力情報になり，データベースを回復します。

＜規則＞

- このオプションを指定した場合，-b オプションでバックアップファイル名を指定する必要があります。このバックアップファイルは，pdcopy コマンドで-z オプションを指定したファイルでなければなりません。

- このオプションを指定する場合は，-l，-c，及び-d オプションは指定できません。

(5) -d 〔ホスト名：〕アンロードログファイル格納ディレクトリ名〔，アンロードログファイル格納ディレクトリ名〕...

～＜識別子：パス名＞

回復に必要なすべてのアンロードログファイルが格納されているディレクトリ名を指定します。アンロードログファイルがあるホストの名称とパス名をコロン（：）で結んで指定してください。

＜適用基準＞

回復に必要なアンロードログファイルの個数が多い場合に使用します（例えば，1 回の pdrstr 実行のコマンドラインに，すべてのアンロードログファイルを指定できないときなど）。

＜規則＞

- アンロードログファイル格納ディレクトリ名とディレクトリ下のファイル名を組み合わせたパス名の長さが，1,023 バイトを超えないようにしてください。また，指定できるディレクトリ数は，最大 128 個です。
- ディレクトリの直下には，アンロードログファイル以外のファイルは格納してないでください。アンロードログファイル以外のファイルがある場合，そのファイルは無視されます。
- このオプションは，-l 及び-L オプションと組み合わせて指定できません。
- このオプションを指定した場合，アンロードログファイルを作成順（システムログ割り当て時刻順）に入力するため，マシンの時刻を変更する前後に作成したアンロードログファイルを同一ディレクトリ下に混在させないでください。
- このオプションに HiRDB ファイルシステム領域は指定できません。
- 指定したディレクトリ下のアンロードログファイルの入力開始位置（入力対象になるアンロードログファイル）は，バックアップ取得時のバックアップ取得モード，及び回復時のオプションの指定によって変わります。アンロードログファイルの入力開始位置を次に示します。

条件	アンロードログファイルの入力開始位置
更新可能モード（pdcopy -M s）で取得したバックアップファイルを使用する場合	ディレクトリ下のすべてのアンロードログファイルが入力対象となります。
参照可能モード（pdcopy -M r）のバックアップファイル，又は参照・更新不可能モード（pdcopy -M x）のバックアップファイルを使用する場合	バックアップ取得時刻以降にアンロードしたファイルをすべて入力対象とします。 上記入力対象のうち，バックアップ取得時に既に現用でないシステムログファイルを，バックアップ取得後にアンロードする運用の場合は，回復時にそのファイルをディレクトリから除外してください。 なお，時刻がマシン間で異なっていることによって，バックアップ取得開始時刻よりも前に入力が必要なアンロードログファイルがある場合は，-T オプションで回復開始時刻を指定してください。指定する時刻は，バックアップ出力先ホストと回復対象ホストとの時間差で求めてください。

条件	アンロードログファイルの入力開始位置
バックアップファイルを使用しないで，アンロードログファイルだけを使用して RD エリアを回復する場合	ディレクトリ下のすべてのアンロードログファイルが入力対象となります。
回復時に-T オプションで回復開始時刻を指定している場合	回復開始時刻以降のアンロードログファイルが入力対象となります。
インメモリ RD エリアを回復する場合	• **インメモリ RD エリアのバックアップとアンロードログファイルを使用して回復する場合** インメモリデータバッファとインメモリ RD エリアの同期を取った時刻以降にアンロードしたファイルをすべて入力対象とします。 インメモリ RD エリアが複数ある場合，その中で同期を取った時刻が一番古い時刻以降にアンロードログしたファイルが入力対象となります。 例えば，バックアップファイル中に RD エリア RD01，RD02 のバックアップがあり，RD01 の同期時刻が 2007 年 12 月 15 日 13 時 45 分 23 秒で，RD02 の同期時刻が同じ日の 14 時 5 分 55 秒の場合，2007 年 12 月 15 日 13 時 45 分 23 秒以降にアンロードしたファイルが入力対象となります。 上記入力対象のうち，上記の同期化時に既に現用でないシステムログファイルを同期後にアンロードする運用の場合は，回復時にそのファイルをディレクトリから除外してください。 • **バックアップファイルを使用しないで，アンロードログファイルだけを使用して回復する場合** 前記と同じです。 • **インメモリ RD エリアとそうでない RD エリアを一緒に回復する場合** ディレクトリ下のすべてのアンロードログファイルが入力対象となります。

- 指定したディレクトリに，ログレスモード又は更新前ログ取得モードで UAP やユティリティを実行したときのアンロードログファイルがあるとエラーになります。この場合，ログレスモード又は更新前ログ取得モードでの実行以降に取得したバックアップ開始時刻を-T オプションに指定して，ログレスモード又は更新前ログ取得モードのシステムログを除外する必要があります。
- 指定ディレクトリがデータベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）にあり，回復に必要なすべてのアンロードログファイルを指定ディレクトリに格納している場合は，ホスト名を省略できます。

(6) -p 処理結果出力ファイル名

～<パス名>

データベース回復ユティリティの処理結果を出力するファイル名を指定します。

```
（例）-p C:¥hirdb¥pdrstr¥list01
```

＜規則＞

- データベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）のパス名を指定します。

- このオプションを省略した場合，データベース回復ユティリティを実行したサーバマシンの次に示すディレクトリに処理結果出力ファイルを作成します。

-p オプションの指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	-p オプションに指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※　コマンド実行環境の環境変数の指定

出力先は KFPR26222-I メッセージに表示されます。

- このオプションの指定に関係なくエラーメッセージはシステムログファイル及び標準出力に，最終処理結果は標準出力に出力されます。ただし，システムログファイルと標準出力に出力されるメッセージの順序は必ずしも同じではありません。

(7)　-w ソート用ワークディレクトリ名

～＜パス名＞

-l オプション，- L オプション，及び-d オプションを指定した場合に，一時ファイルを作成するディレクトリの名称を指定します。回復対象の RD エリアが複数のサーバマシンにある場合，指定したディレクトリが各サーバマシンになければなりません。省略した場合，次のディレクトリが仮定されます。

-w オプションの指定	システム定義の pd_tmp_directory オペランドの指定		
	あり	なし	
		環境変数 TMP※の指定	
		あり	なし
あり	-w オプションに指定したディレクトリ		
なし	pd_tmp_directory に指定したディレクトリ	TMP に指定したディレクトリ	%PDDIR%¥tmp ディレクトリ

注※

システム環境変数の指定

ソート用ワークディレクトリ下の空き領域は，次の値以上にする必要があります。

a＋（a÷130×36）

a：max｛min（入力ログ総容量，1536000000），シンクポイントの最大間隔出力ログ量｝（単位はバイト）

入力ログ総容量：

pdrstr 実行時に指定するアンロードログファイル，又はシステムログファイルの総容量です。

シンクポイントの最大間隔出力ログ量：

前回のシンクポイントから新たに取得されるシンクポイントまでに出力されるログ量です。次の計算式で求められます。

シンクポイントダンプ有効化処理のスキップ回数の最大値

×（pd_log_sdinterval で指定したシステムログ出力量の値

×↑pd_log_max_data_size の値÷pd_log_rec_leng の値↑

×pd_log_rec_leng の値）

(8) -y ソート用ワークバッファサイズ

～<符号なし整数>((256～2097151))《1024》

-l オプション，- L オプション，及び-d オプションを指定したときに使用するワークバッファサイズを，キロバイト単位で指定します。この指定値が少ないと，KBLS300-E メッセージが出力され，データベース回復プロセスが異常終了します。指定値の目安を次に示します。

ソート用ワークバッファサイズ ＞ b（単位はキロバイト）

$b = \{ \sqrt{a + (a \div 130 \times 36)} \times 2 \}$

a：max｛min（入力ログ総容量，1536000000），シンクポイントの最大間隔出力ログ量｝（単位はバイト）

入力ログ総容量：

pdrstr 実行時に指定するアンロードログファイル，又はシステムログファイルの総容量です。

シンクポイントの最大間隔出力ログ量：

前回のシンクポイントから新たに取得されるシンクポイントまでに出力されるログ量です。次の計算式で求められます。

シンクポイントダンプ有効化処理のスキップ回数の最大値

×（pd_log_sdinterval で指定したシステムログ出力量の値

×↑pd_log_max_data_size の値÷pd_log_rec_leng の値↑

×pd_log_rec_leng の値）

(9) -k {u｜i｜n}

バックアップファイルの種別を指定します。

u：

ファイルのバックアップを入力する場合に指定します。

i：

HiRDB ファイルシステム領域に取得したバックアップを入力する場合に指定します。実行系システムと待機系システムとでバックアップファイルを共用し，実行系システムで取得したバックアップを使用して待機系システムを回復する場合に有効です。

n：

NetBackup を使用してデータベースを回復する場合に指定します。この場合，-b オプションには NetBackup のポリシー名を指定します。

(10) -U {バックアップ検索条件開始時刻，バックアップ検索条件終了時刻|，バックアップ検索条件終了時刻}

同一ポリシー名で取得したバックアップ中の，任意の時点で取得したバックアップから回復したい場合に指定します。このオプションは，NetBackup 連携機能使用時（-k n 指定時）に指定します。それ以外の場合に指定しても無効となります。

バックアップ検索条件開始時刻，及びバックアップ検索条件終了時刻の両方を指定した場合，指定した期間内での最新のバックアップを使用して回復します。バックアップ検索条件終了時刻だけを指定した場合，終了時刻時点でのバックアップを使用して回復します。

＜規則＞

- このオプションを省略した場合，同一ポリシー名で取得したバックアップ中の最新のバックアップを使用します。
- バックアップの取得日時は，bpimagelist コマンド（-policy オプション指定）で参照できます。
- バックアップ検索条件開始時刻及びバックアップ検索条件終了時刻は，日付と時刻を_（アンダースコア）で結んで指定します。形式を次に示します。時刻を省略した場合，開始時刻は 000000 が，終了時刻は 235959 が仮定されます。

```
-U YYYYMMDD〔_hhmmss〕, YYYYMMDD〔_hhmmss〕
```

YYYY：年　～＜符号なし整数＞((1990～2037))

西暦で指定します。

MM ：月　～＜符号なし整数＞((01～12))

DD ：日　～＜符号なし整数＞((01～31))

hh ：時　～＜符号なし整数＞((00～23))

mm ：分　～＜符号なし整数＞((00～59))

ss ：秒　～＜符号なし整数＞((00～59))

終了時刻だけを指定する場合は，コンマ（,）に続けて終了時刻を指定します。

(11) -g 差分バックアップグループ名

～＜英数字＞((1～30))

差分バックアップ機能を使用する場合，pdcopy で指定した差分バックアップグループ名を指定します。

(12) -K 差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名

～＜パス名＞

差分バックアップ機能を使用する場合，pdcopy で指定した差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名を指定します。

"差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名¥差分バックアップグループ名"が，167 文字以内になるようにしてください。

(13) -Y 書き込みバッファサイズ

～＜符号なし整数＞((4～131072))

データベースの回復時に，回復データを書き込むときのバッファサイズをキロバイト単位で指定します。このオプションを指定することで，一度に書き込むデータ量が増えるため，書き込み回数を削減できます。

＜適用基準＞

次の適用条件に該当する場合，性能改善が見込めるため，指定することをお勧めします。

回復に使用するデータ	適用条件
差分バックアップファイル	前回の差分バックアップ取得後のデータ更新が，追加（INSERT 文）だけ，又は全データ削除（DELETE 文，PURGE TABLE 文）を行っている場合。
アンロードログファイル又はシステムログファイル	データベースへの更新業務で，主なデータ更新が追加（INSERT 文）の場合。

＜指定値の目安＞

指定値として，1024～2048 を指定することをお勧めします。

指定値の目安は，データベースの回復対象となる全 RD エリアのページ長の公倍数となります。

＜規則＞

1. このオプションを省略した場合，次の値が仮定されます。

   ・差分バックアップファイル，又はアンロードログファイル若しくはシステムログファイルを使用した回復の場合は，各回復対象 RD エリアのページ長

   ・上記以外の場合は 60

2. 回復対象 RD エリア中の最大ページ長より小さい値を指定した場合，その指定は無視され，オプション省略時の仮定値が適用されます。

(14) -W 実行監視時間

実行監視時間　～＜符号なし整数＞((0～3600))

データベース回復ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，このオプションを省略した場合は，システム共通定義の pd_cmd_exec_time オペランドの指定値に従います。

(15) -l メッセージ出力レベル〔，メッセージ出力間隔〕

データベース回復ユティリティの稼働状況を確認するために指定します。次の四つのメッセージを出力します。

- KFPR26294-I：回復を行う RD エリア一つに対して一つ表示します※1。
- KFPR26295-I：RD エリアがメッセージ出力間隔を経過しても回復実行中の場合に，メッセージ出力間隔ごとに 1 回表示します※1。
- KFPR26296-I：ログレコードを読み込みソートに入力する際に 1 回表示します※1※2。
- KFPR26297-I：ログレコードをソートするごとに 1 回表示します※1※2。

注※1

このメッセージは，-l，-d，-L オプションを指定している場合，サーバ単位にログを 1.5 ギガバイト使用して，回復を行うごとに表示します。

注※2

このメッセージは，-l，-d，-L オプションを指定している場合，回復に使用するログが存在する場合だけ表示します。

これらのメッセージを確認することで，無応答となっていないことを確認できます。

メッセージ出力レベル　～<符号なし整数>｛0 | 1｝

稼働状況を確認するためのメッセージを出力するかどうかを指定します。

0：稼働状況を確認するためのメッセージを出力しません。

1：稼働状況を確認するためのメッセージを出力します。

メッセージ出力間隔　～<符号なし整数>((1～300)) ≪60≫

メッセージ出力間隔を，分単位で指定します。この指定によって，KFPR26295-I メッセージの出力間隔を変更できます。出力間隔を省略値から変更したい場合に指定してください。

(16) -f 制御文ファイル名

～<パス名>

制御文ファイル名を指定します。

```
（例）-f C:¥hirdb¥pdrstr¥cont01
```

制御文ファイルには「オプションの形式」の制御文ファイルに記載のオプションが指定できます。なお，これらのオプションは，pdrstr コマンド中に直接指定することもできます。

＜制御文ファイルの形式＞

```
   { {-l フラグ引数 | -L | -d フラグ引数} | {-a | -c | -u フラグ引数 | -s フラグ引数 | -r フラグ引数} 〔-T フラグ引数〕}
   〔 { {-l フラグ引数 | -L | -d フラグ引数} | {-u フラグ引数 | -s フラグ引数 | -r フラグ引数} 〔-T フラグ引数〕} 〕
   〔 { {-l フラグ引数 | -L | -d フラグ引数} | {-u フラグ引数 | -s フラグ引数 | -r フラグ引数} 〔-T フラグ引数〕} 〕
                                    .
                                    .
                                    .
```

＜注意事項＞

- 制御文ファイルは，データベース回復ユティリティを実行したサーバマシン（pdrstr コマンドを入力したサーバマシン）に作成してください。
- 制御文ファイル中には，回復対象の RD エリアを指定するオプションと-T オプションの組み合わせを 1 行で指定してください。
- 制御文ファイルの 1 行は，32,768 バイト以内で指定してください。
- 制御文ファイル中にはコメントを記述できます。ただし，コメントだけの行は記述できません。
- 各オプションフラグについては，それぞれ対応するオプションの説明を参照してください。
- -u，-s，又は-r オプションを複数行指定する場合は，1 行目に指定したオプションフラグだけ 2 行目以降も指定できます（-a 及び-c オプションは除きます）。例えば，1 行目に-u オプションを指定した場合，2 行目以降は-u オプションだけ指定できます。
- -l オプションを複数行指定した場合，最後の指定が有効となります。また，-d オプションも同様です。
- コマンドライン中に指定したオプションは，制御文ファイル中には指定できません。ただし，-T オプションはコマンドライン及び制御文ファイルの両方に指定できます。この場合，制御文ファイル中で-T オプションを指定していない項目に対しては，コマンドラインの-T オプションの指定が有効となります。
- コマンドライン中に指定したオプションの組み合わせによっては，制御文ファイルを指定できないことがあります。オプションの組み合わせと制御文ファイルの指定可否を次に示します。

-l，-L，又は-d オプションの指定	-a，-c，-u，-s，又は-r オプションの指定	
	なし	あり
なし	○※1	○※2
あり	○※3	×

（凡例）

○：指定できます。

×：指定できません。

注※1

制御文ファイルに指定できるオプションを，すべて指定できます。

注※2

-l，-L，又は-d オプションを指定できます。

注※3

-a，-c，-u，-s，又は-r オプションと，-T オプションを指定できます。

＜正しい制御文の例＞

```
-r pdbuser1 -T 20040809,20040810
-r pdbuser2 -T 20040809
-l C:¥hirdb¥unldlog¥unld01,C:¥hirdb¥unldlog¥unld02,
C:¥hirdb¥unldlog¥unld03,C:¥hirdb¥unldlog¥unld04
```

アンロードログファイル unld01，unld02，unld03，及び unld04 に格納されているデータベース更新履歴を使用してデータベースを回復します。このとき，RD エリア pdbuser1 は，2004 年 8 月 9 日 0 時 0 分 0 秒から 2004 年 8 月 10 日 24 時 59 分 59 秒までのデータベース更新履歴を使用して回復します。また，RD エリア pdbuser2 は，2004 年 8 月 9 日 0 時 0 分 0 秒以降のデータベース更新履歴を使用して回復します。

＜誤った制御文の例 1 ＞

```
-r pdbuser1 -T 20040809,20040810
-r pdbuser2 -T 20040809 -l C:¥hirdb¥unldlog¥unld01,C:¥hirdb¥unldlog¥unld02   ←エラー
```

-l，-L，又は-d オプションを指定した行に，ほかのオプションは指定できません。

＜誤った制御文の例 2 ＞

```
-r pdbuser1 -T 20040809,20040810
-r pdbuser2 -T 20040809
-l C:¥hirdb¥unldlog¥unld01
,C:¥hirdb¥unldlog¥unld02   ←エラー
```

1 行に指定できる文字列長には上限（改行コードを含んで 32,768 バイト）があるため，-l，又は-d オプションの引数が長くなる場合には，オプションの途中で改行ができます。ただし，改行ができる箇所は，アンロードログファイルのパス名を区切っているコンマ（,）の後だけとなります。-l，又は-d オプションを指定した行がコンマ以外で終わっている場合，そこを 1 行の終わりと判断します。

(17) -a

全 RD エリア（ただし，リスト用 RD エリア及び一時表用 RD エリアを除きます）を回復する場合に指定します。

＜規則＞

- 制御文ファイルに指定する場合，-a オプションを指定したときは，制御文は 1 行しか指定できません。

(18) -c

-b オプションで指定したバックアップファイル中の全 RD エリア（ただし，リスト用 RD エリアを除きます）を回復する場合に指定します。

＜規則＞

- 制御文ファイルに指定する場合，-c オプションを指定したときは，制御文は 1 行しか指定できません。
- -c オプションを指定する場合は，必ず-b オプションも指定してください。ただし，差分バックアップ機能を使用する場合は，-b オプションを指定する必要はありません（-b オプションを指定すれば手動回復となり，-b オプションを省略すれば自動回復となります）。
- -c オプションを指定する場合は，-l オプションは指定しないでください。
- pdcopy で-J オプションを指定してバックアップを取得した場合に KFPR26063-I メッセージが出力されているときは，そのバックアップを使用して回復しても正常終了しません（処理がスキップされた RD エリアも回復対象となるため）。

(19) -u ユニット識別子〔，ユニット識別子〕...

～＜識別子＞((4 文字))

ユニット下の全 RD エリア（ただし，リスト用 RD エリア及び一時表用 RD エリアを除きます）を回復する場合に，回復対象のユニット識別子を指定します。

＜規則＞

- ユニット識別子を複数指定する場合は，重複しないようにしてください。
- 制御文ファイルに指定する場合，ユニット識別子は制御文全体で一意にしてください。

(20) -s サーバ名〔，サーバ名〕...

～＜識別子＞((最大 8 文字))

サーバ下の全 RD エリア（ただし，リスト用 RD エリア及び一時表用 RD エリアを除きます）を回復する場合に，回復対象のサーバ名を指定します。

＜規則＞

- サーバ名を複数指定する場合は，重複しないようにしてください。
- 制御文ファイルに指定する場合，サーバ名は制御文全体で一意にしてください。

(21)　-r RD エリア名〔，RD エリア名〕...

～＜識別子＞((最大 30 文字))

回復対象の RD エリア名を指定します。

＜規則＞

- RD エリア名の指定方法については，「運用コマンド，ユティリティでの RD エリアの指定」を参照してください。

  ただし，制御文ファイルに指定した RD エリア名が重複していた場合は次のとおり処理されます。

・直接指定した RD エリア名が重複した場合

次のように指定すると，KFPR26007-E エラーになります（下線部分が重複部分です）。

```
-r LOB1,USER1 -T 20081125_000000,20081125_240000
-r DDIC,USER1 -T 20081126_000000,20081126_240000
```

・一括指定した RD エリア名が重複した場合

次のように指定すると，最初に指定した値が有効になります（下線部分が重複部分です）。

```
-r LOB1,USER* -T 20081125_000000,20081125_240000…有効
-r DDIC,USER* -T 20081126_000000,20081126_240000
```

・直接指定した RD エリア名と一括指定した RD エリア名が重複した場合

次のように指定すると，USER1 に対しては直接指定での指定値が有効になります（下線部分が重複部分です）。

```
-r LOB1,USER1 -T 20081125_000000,20081125_240000…有効
-r DDIC,USER* -T 20081126_000000,20081126_240000
```

- リスト用 RD エリア，及び一時表用 RD エリアは指定できません。

(22)　-T　{回復開始時刻，回復終了時刻｜回復開始時刻｜，回復終了時刻}

範囲指定の回復をする場合に指定します。

-l，-d，又は-L オプションを指定した場合に，データベース回復時の回復対象となる先頭のログの出力時刻と最終のログの出力時刻を指定します。

＜規則＞

- -l，-d，又は-L オプションを指定した場合にこのオプションを省略したときは，システムログファイル又はアンロードログファイルの最初のログの出力時刻と最終のログの出力時刻が仮定されます。
- -l，-d，又は-L オプションを指定していない場合，このオプションを指定しても無視されます。ただし，差分バックアップ機能を使用する場合は指定すると有効となります。
- 回復開始時刻は，回復終了時刻よりも前の時刻を指定してください。回復終了時刻よりも後の時刻を指定した場合はエラーとなり，処理を終了します。また，このオプションを指定した場合は回復開始時刻又は回復終了時刻のどちらかは必ず指定してください。

- 回復開始時刻及び回復終了時刻は，日付と時刻を_（アンダースコア）で結んで指定します。形式を次に示します。時刻を省略した場合，回復開始時刻は 000000 が，回復終了時刻は 235959 が仮定されます。

```
-T YYYYMMDD〔_hhmmss〕, YYYYMMDD〔_hhmmss〕
```

YYYY：年　～<符号なし整数>((1990～2105))

西暦で指定します。

MM ：月　～<符号なし整数>((01～12))

DD ：日　～<符号なし整数>((01～31))

hh ：時　～<符号なし整数>((00～23))

mm ：分　～<符号なし整数>((00～59))

ss ：秒　～<符号なし整数>((00～59))

回復終了時刻だけを指定する場合は，コンマ（,）に続けて回復終了時刻を指定します。また，回復開始時刻だけを指定する場合は，回復開始時刻の後にコンマを指定しないでください。

＜注意事項＞

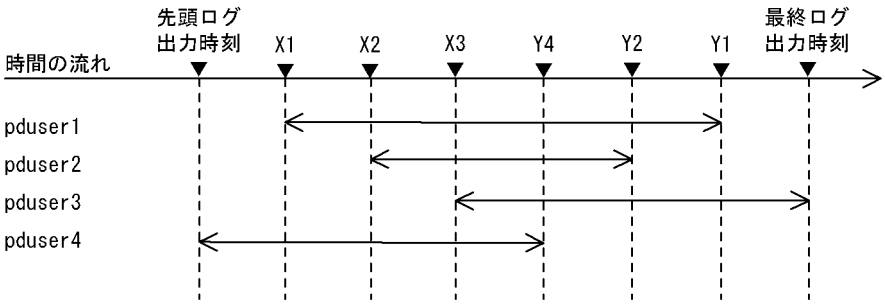
1. 範囲指定の回復をする場合，サーバマシンによって時間が異なるので，範囲指定をする RD エリアがあるサーバマシンの時刻を指定してください。
2. バックアップファイルを使用して範囲指定の回復をする場合，回復開始時刻は指定する必要はありません。指定した場合は無視され，バックアップ開始時刻が仮定されます。また，回復終了時刻にはバックアップ取得終了時刻以降の値を指定してください。
3. 一つのトランザクションが更新した複数の RD エリアを，RD エリア単位の範囲指定の回復をする場合，回復終了時刻にはトランザクション終了時刻（同期点）以降の値を指定してください。トランザクション終了時刻より前の値を指定した場合はエラーとなります。この場合，再度正しい時刻（トランザクション終了時刻以降）を指定する必要があります。
4. -T オプションと，制御文ファイル中の-r オプションに-T オプションを指定した例を次に示します。

   ＜コマンドの指定＞

```
pdrstr …
       -T X1, Y1
       …
       -f 制御文ファイル1
```

   ＜制御文ファイル１の内容＞

```
-r pduser1
-r pduser2 -T X2,Y2
-r pduser3 -T X3
-r pduser4 -T ,Y4
```



（凡例）←→：各RDエリアの回復範囲

注　制御文ファイル中で，回復開始時刻又は回復終了時刻のどちらかを省略していても，コマンドで指定した回復開始時刻又は回復終了時刻は引き継がれません。指定値が引き継がれるのはpduser1のように，回復開始時刻及び回復終了時刻の両方を省略したときだけです。

5. データベース回復ユティリティ（pdrstr）を複数回に分けて実行する場合は，関連するすべてのRD エリアの回復開始時刻及び回復終了時間を合わせてください。関連するすべての RD エリアの詳細については，マニュアル「HiRDB システム運用ガイド」の「同時にバックアップを取得する必要がある RD エリア」を参照してください。

(23)　-z ログポイント情報ファイル名

～＜パス名＞

ログポイント情報ファイルを再作成する場合，そのファイル名を指定します。このファイルは pdrstr コマンドを入力したホスト上に作成されます。

＜規則＞

- このオプションを指定した場合，-b オプションでバックアップファイル名を指定する必要があります。この時に指定するバックアップファイルは pdcopy コマンドで-z オプションを指定して取得したものでなければなりません。
- 既にファイルがある場合，ユティリティはこのファイルを削除してから再作成します。

(24)　-J

このオプションは，マスタディレクトリ用 RD エリアの回復漏れ確認機能による，警告メッセージ（KFPR26298-W）の出力を抑止する場合に指定してください。

なお，このオプションを指定する場合は，回復対象としてマスタディレクトリ用 RD エリアを回復することをすでに計画しているときや，マスタディレクトリ用 RD エリアに対する更新が行われた操作で更新した RD エリアとは関係のない RD エリアを回復するときに指定してください。

19.5 注意事項

1. データベース回復中は，回復対象の RD エリアに対してほかのユーザはアクセスできません。
2. データベース初期設定ユティリティを実行する前のアンロードログは，データベース回復ユティリティを実行するときに指定しないでください。指定した場合，動作は保証できません。
3. データ連動の対象となる抽出側データベースを回復する場合，データベースのバックアップとバックアップ取得以降のログを基にして回復してください。それ以外の方法でデータベースを回復すると，抽出側と反映側のデータベースの内容が不整合になります。
4. データベース回復ユティリティの実行中に，電源断又は pdkill コマンドで実行が中断された場合，HiRDB ファイルシステム領域が壊れる場合があります。pdfls コマンドを実行して矛盾が検知された場合は，次に示すどちらかの手順で回復してください。
   - pdfmkfs コマンドで HiRDB ファイルシステム領域を再度作成し，格納していた HiRDB ファイルは RD エリアごとに回復ユティリティを実行します。
   - pdfbkup コマンド及び pdfrstr コマンドを実行して，HiRDB ファイルシステム領域のバックアップとリストアをします。
5. データベース回復ユティリティの最大同時実行数は，システム共通定義の pd_utl_exec_mode オペランドの指定値によって異なります。
   - **pd_utl_exec_mode=0 の場合**

     最大同時実行数は 32 です。
   - **pd_utl_exec_mode=1 の場合**

     最大同時実行数は pd_max_users オペランドの指定値です。
6. pdcopy で-J オプションを指定してバックアップを取得した場合に，処理がスキップされた RD エリアがあるときは，そのバックアップを使用して回復しても処理をスキップした RD エリアは回復されません。
7. バックアップファイル（-b オプション）を指定しないでアンロードログファイルを使用（-l 又は-d オプション指定）して回復する場合，事前に RD エリアをバックアップ取得時点の状態に回復しておく必要があります。この場合，pdcopy 又はほかの製品のバックアップ機能を使用して取得したバックアップファイルを使用します。

   なお，一度回復に使用したアンロードログファイルは，連続で同じ RD エリアの回復には使用できません。再度やり直す場合は，バックアップファイルから回復してください。
8. pdrstr のリターンコードを次に示します。

   0：正常終了（すべての回復に成功した，又は KFPR26298-W メッセージを出力しているがすべての回復に成功しました）

   4：警告終了（回復処理は正常終了しましたが，回復処理に関係のない警告レベルのエラーが発生しました）

   8：異常終了（回復処理中にエラーが発生しました。エラー要因を取り除き，再度回復を実行してください）

12：異常終了（すべての RD エリアの回復に失敗しました。エラー要因を取り除き，再度回復を実行してください）

9. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，pdrstr の制御文ファイルには，BOM を持つファイルを使用できます。なお，pdrstr の制御文ファイルとして BOM を持つファイルを使用しても，BOM は読み飛ばされます。また，pdrstr で出力されるファイルには BOM は付きません。
10. 時刻を遅らせた後で障害が発生した場合のデータベースの回復については，マニュアル「HiRDB システム運用ガイド」を参照してください。
11. マスタディレクトリ用 RD エリア以外の RD エリアを回復する場合，マスタディレクトリ用 RD エリアの更新時刻より前に取得したバックアップから回復するときに，KFPR26298-W メッセージを出力します。RD エリアの整合性に問題がないか確認してください。OS の時刻を変更したり，サーバマシン間で OS の時刻がずれていたりする場合は，警告メッセージが出力されることがあります。この場合は，この警告メッセージを無視してください。

19.6　データベース回復ユティリティの処理結果リスト

データベース回復ユティリティの処理結果リストの出力例を次に示します。

```
pdrstr(VV-RR)     ***** DB RECOVERY *****   1995-05-16 11:35:35 [1] utl3 [2]
*** DB RECOVERY INFORMATION LIST ***
-------------------------------------------------------
<<RDAREA INFORMATION>>
  UNIT NAME  :      un16 [3] SERVER NAME:sds [4]
  RDAREA NAME: pdbuser01 [5]
  RDAREA ID :          4 [6] ATTRIBUTE : USER [7]      PAGE SIZE :  4096 [8]
  RECOVERY STARTED AT : 1995-05-16 11:34:19 [9]
  RECOVERY ENDED AT   : 1995-05-16 11:34:46 [10]
<<BACKUP INFORMATION>>
  BACKUP STARTED AT : 1995-05-16 11:11:42 [11]
  BACKUP ENDED AT   : 1995-05-16 11:11:42 [12]
<<LOG INFORMATION>>
  1ST LOG    : 1995-05-16 10:57:48 [13]
  LAST LOG   : 1995-05-16 10:58:26 [14]
  LOG COUNT  :         42 [15]
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt06 [16]
   EXTENT COUNT :   2 [18]
   PAGE NUMBER  :    124 [17]
 2 c:¥hirdb¥area1¥rdmt07 [16]
   EXTENT COUNT :   2 [18]
   PAGE NUMBER  :    108 [17]
-------------------------------------------------------
<<RDAREA INFORMATION>>
  UNIT NAME  :      un16 SERVER NAME:sds
  RDAREA NAME: pdbuser03
  RDAREA ID :           5 ATTRIBUTE : USER            PAGE SIZE :  4096
  RECOVERY STARTED AT : 1995-05-16 11:34:46
  RECOVERY ENDED AT   : 1995-05-16 11:35:10
<<BACKUP INFORMATION>>
  BACKUP STARTED AT : 1995-05-16 11:11:42
  BACKUP ENDED AT   : 1995-05-16 11:11:43
<<LOG INFORMATION>>
  1ST LOG    : 1995-05-16 10:59:50
  LAST LOG   : 1995-05-16 11:02:35
  LOG COUNT  :         42
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt08
   EXTENT COUNT :   2
   PAGE NUMBER  :    124
 2 c:¥hirdb¥area1¥rdmt09
   EXTENT COUNT :   2
   PAGE NUMBER  :    108
-------------------------------------------------------
<<RDAREA INFORMATION>>
  UNIT NAME  :      un16 SERVER NAME:sds
  RDAREA NAME: pdbuser05
  RDAREA ID :           6 ATTRIBUTE : USER            PAGE SIZE :  4096
  RECOVERY STARTED AT : 1995-05-16 11:35:10
  RECOVERY ENDED AT   : 1995-05-16 11:35:34
<<BACKUP INFORMATION>>
```

```
  BACKUP STARTED AT : 1995-05-16 11:11:43
  BACKUP ENDED AT   : 1995-05-16 11:11:43
<<LOG INFORMATION>>
  1ST LOG    : 1995-05-16 11:03:21
  LAST LOG   : 1995-05-16 11:08:47
  LOG COUNT  :         42
 <FILE INFORMATION>
 1 c:¥hirdb¥area1¥rdmt10
   EXTENT COUNT :   2
   PAGE NUMBER  :    124
 2 c:¥hirdb¥area1¥rdmt11
   EXTENT COUNT :   2
   PAGE NUMBER  :    108
-------------------------------------------------------
```

〔説明〕

1. データベース回復ユティリティを実行した日時（YYYY-MM-DD hh:mm:ss の形式）
   YYYY：年（西暦） MM：月　DD：日　hh：時　mm：分　ss：秒
2. HiRDB 識別子
3. RD エリアがあるユニットの名称
4. RD エリアがあるサーバの名称
5. 回復した RD エリアの名称
6. 回復した RD エリアの番号
7. RD エリアの種類
   MASTERDIRECTORY：マスタディレクトリ用 RD エリア
   DATADIRECTORY：データディレクトリ用 RD エリア
   DATADICTIONARY：データディクショナリ用 RD エリア
   SYSTEM_LOB：データディクショナリ LOB 用 RD エリア
   USER：ユーザ用 RD エリア
   USER_LOB：ユーザ LOB 用 RD エリア
   REG：レジストリ用 RD エリア
   REG_LOB：レジストリ LOB 用 RD エリア
8. RD エリアのページ長（単位：バイト）
9. 該当する RD エリアの回復開始時刻（形式は 1 と同じ）
10. 該当する RD エリアの回復終了時刻（形式は 1 と同じ）
11. 該当する RD エリアのバックアップ取得開始時刻（形式は 1 と同じ）
12. 該当する RD エリアのバックアップ取得終了時刻（形式は 1 と同じ）
13. 回復処理のときに，最初に使用したログレコードの出力時刻（形式は 1 と同じ）
14. 回復処理のときに，最後に使用したログレコードの出力時刻（形式は 1 と同じ）
15. 回復処理に使用した更新ログレコード件数

16. RD エリアを構成する HiRDB ファイルの名称

17. 該当する HiRDB ファイルに割り当てられているページ数

18. HiRDB ファイルのエクステント数

注 1

回復するデータがない場合は 9 に’No data for recovery’を出力し，8〜15 は出力されません。

注 2

バックアップがない場合は，11，12 は出力されません。

注 3

該当する RD エリアに対するログレコードがない場合は，13，14 は出力されません。

20 レジストリ機能初期設定ユティリティ（pdreginit）

この章では，レジストリ機能を管理，操作するためのレジストリ用 RD エリア，レジストリ LOB 用 RD エリアを登録し，レジストリ管理表を作成するレジストリ機能初期設定ユティリティ（pdreginit）について説明します。

20.1　概要

実行者　HiRDB 管理者が実行できます

20.1.1　機能

レジストリ機能初期設定ユティリティは，プラグインモジュールを使用するためにレジストリ機能を HiRDB に組み込むためのユティリティです。

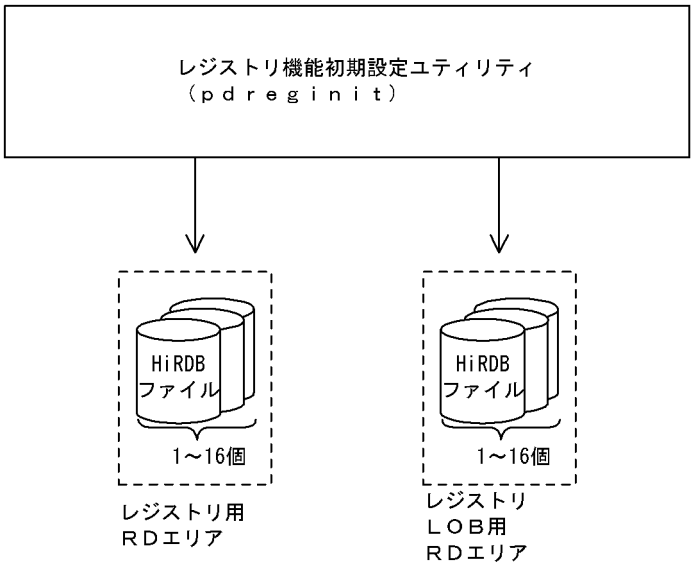
レジストリ機能初期設定ユティリティでレジストリ用 RD エリア及びレジストリ LOB 用 RD エリアを追加すると，プラグインが使用するレジストリの情報を管理するレジストリ管理表が作成されます。

- **前提条件**

レジストリ機能初期設定ユティリティを実行する場合は，あらかじめ HiRDB がストアドプロシジャ機能を使用できるようにしておく必要があります。ストアドプロシジャ機能が使用できるかどうかは，pddbls コマンドで判断できます。pddbls コマンドでデータディクショナリ LOB 用 RD エリアがあるかを調べます。データディクショナリ LOB 用 RD エリアがあれば，ストアドプロシジャ機能が使用できます。データディクショナリ LOB 用 RD エリアがない場合は，データベース構成変更ユティリティ（pdmod）でデータディクショナリ LOB 用 RD エリアを追加して，ストアドプロシジャ機能を使用できる状態にしてください。

レジストリ機能初期設定ユティリティ（pdreginit）の概要を次の図に示します。

図 20–1　レジストリ機能初期設定ユティリティ（pdreginit）の概要



20.2　コマンドの形式

20.2.1　形式

```
pdreginit 〔-k 処理種別〕〔-a 制御文ファイル名〕〔-W 実行監視時間〕
```

20.2.2　オプション

● **-k 処理種別**

レジストリ用 RD エリアやレジストリ LOB 用 RD エリアを追加するか，又はレジストリ操作用ストアドプロシジャの登録だけをするかなどの処理種別を指定します。省略した場合，all が仮定されます。

all：

次の処理をする場合に指定します。なお，-a オプションで制御文ファイルを指定する必要があります。

- レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアの追加
- レジストリ機能を使用するためのレジストリ管理表の作成
- レジストリ操作用ストアドプロシジャの登録

proc：

レジストリ操作用ストアドプロシジャの登録だけをする場合に指定します。

レジストリ用 RD エリアの追加処理が正常終了（データベース構成変更ユティリティで KFPX24200-I メッセージのリターンコードが 0 又は 4）した後，レジストリ機能初期設定ユティリティがエラーとなった場合に，レジストリ操作用ストアドプロシジャの登録処理だけをするときなどに指定します。なお，-a オプションは同時に指定できません。

renew：

レジストリ操作用ストアドプロシジャを再登録する場合に指定します。

HiRDB/シングルサーバから HiRDB/パラレルサーバへ移行した場合に，レジストリ操作用ストアドプロシジャの再登録をするときなどに指定します。なお，-a オプションは同時に指定できません。

● **-a 制御文ファイル名**　～＜パス名＞((255 文字以内))

レジストリ機能初期設定ユティリティの制御文があるファイルの名称を絶対パス名で指定します。

HiRDB/パラレルサーバの場合，制御文ファイルは，ディクショナリサーバがあるサーバマシンから参照できるようにする必要があります。

この制御文では，レジストリ用 RD エリア及びレジストリ LOB 用 RD エリアの create rdarea 文を同時に一つずつ指定する必要があります。既にレジストリ用 RD エリア，レジストリ LOB 用 RD エリアがある場合には，エラーとなります。

● **-W 実行監視時間**　～<符号なし整数>((0～3600)) ≪0≫

レジストリ機能初期設定ユティリティの実行時間を監視する場合に，その監視時間を分単位で指定します。このオプションに指定する値の目安や動作については，マニュアル「HiRDB システム定義」のシステム共通定義 pd_cmd_exec_time オペランドの説明を参照してください。

なお，このオプションに 0 を指定した場合は，監視を行いません。

また，システム共通定義の pd_cmd_exec_time オペランドに指定した実行監視時間は適用されません。オプションを省略した場合は，0 を仮定します。

20.3　制御文

20.3.1　create rdarea 文

```
create rdarea RDエリア名
          〔globalbuffer グローバルバッファ名〕
         for  {registry | LOB used by HiRDB (SQL_REGISTRY) }
          〔page ページ長 characters〕
          〔storage control segment セグメントサイズ pages〕
          〔extension  {use 増分セグメント数 segments | nouse} 〕
         file name "HiRDBファイルシステム領域名¥HiRDBファイル名"
             initial HiRDBファイル内セグメント数 segments
           〔file name "HiRDBファイルシステム領域名¥HiRDBファイル名"
                initial HiRDBファイル内セグメント数 segments〕…
  ;
```

(1)　RD エリア名

～<識別子>((1～30))

HiRDB で使用する RD エリアの名称を指定します。RD エリア名に ALL は指定できません。RD エリア名を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。

（例）

- **create rdarea "pdbuser01" for ...**
  この場合，RD エリア名は pdbuser01 で扱われます。
- **create rdarea pdbuser01 for ...**
  この場合，RD エリア名は PDBUSER01 で扱われます。

(2)　globalbuffer グローバルバッファ名

～<識別子>((1～16))

追加した RD エリアに一時的にグローバルバッファを割り当てたい場合に，HiRDB システムにある（HiRDB/パラレルサーバの場合は，ディクショナリサーバにある）グローバルバッファ名を指定します。ただし，インデクス用のグローバルバッファは指定できません。グローバルバッファについては，pdbufls コマンドで確認できます。

＜規則＞

1. レジストリ用 RD エリアを追加するときは，このオペランドは必ず指定します。
2. 指定するグローバルバッファのグローバルバッファ長は，追加する RD エリアのページ長以上とします。グローバルバッファ長については，pdbufls コマンドで確認できます。

3. グローバルバッファ長は，システム共通定義の pdbuffer 文の-l オペランドの指定値，及び HiRDB 起動時のグローバルバッファに割り当たっている RD エリアの最大ページ長の大きい方の値です。
4. このオペランドの指定は，次回の HiRDB の正常開始時には無効となります。したがって，次回の正常開始時に，システム共通定義のグローバルバッファの指定を変更する必要があります。
5. このオペランドの指定時にグローバルバッファを割り当てられない場合は，RD エリアは追加できません。

(3) for {registry | LOB used by HiRDB（SQL_REGISTRY）}

RD エリアの種類を指定します

registry

RD エリアがレジストリ用 RD エリアの場合に指定します。

LOB used by HiRDB（SQL_REGISTRY）

RD エリアがレジストリ LOB 用 RD エリアの場合に指定します。

(4) page ページ長 characters

～<符号なし整数>((4096～30720))《4096 又は 8192》

RD エリアを構成する HiRDB ファイルのページ長を，2048 の倍数のバイト単位で指定します。

＜レジストリ LOB 用 RD エリアの場合＞

8192 を指定してください。8192 以外の値を指定した場合でも，8192 が仮定されます。

ここで指定したページ長は，HiRDB が RD エリアに対して入出力する最小の単位となります。

(5) storage control segment セグメントサイズ pages

～<符号なし整数>((1～16000))《1 又は 50》

1 セグメントの大きさを，ページ数で指定します。

＜レジストリ LOB 用 RD エリアの場合＞

1 を指定してください。1 以外の値を指定した場合でも，1 が仮定されます。

(6) extension {use 増分セグメント数 segments | nouse}

RD エリアに自動増分を適用するかどうかを指定します。

RD エリアの自動増分については，マニュアル「HiRDB システム運用ガイド」を参照してください。

＜前提条件＞

RD エリアがある HiRDB ファイルシステム領域に-e オプション又は-a オプションが指定されている必要があります。

use 増分セグメント数 segments：

RD エリアの自動増分を適用します。

RD エリアの自動増分は，レジストリ用 RD エリア，及びレジストリ LOB 用 RD エリアの場合に有効となります。ほかの RD エリアに use を指定しても，nouse が仮定されます。

自動増分の契機は，RD エリア内の空きセグメント数が自動増分契機に達したときです。
pd_rdarea_extension_timing オペランドで指定できます。

増分セグメント数　～＜符号なし整数＞((1～64000))

増分セグメント数を指定します。

- pdfmkfs -e を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルのエクステント数は最大 24 で，これを超えるとエラーとなります。また，HiRDB ファイルシステム領域の空きが不足したり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

  HiRDB ファイルシステム領域当たりの最大増分回数は，HiRDB ファイルシステム領域作成時の指定値で決まるため，HiRDB ファイルシステム領域内のファイル数，増分頻度を考慮し最大増分回数を定義しておく必要があります。

- pdfmkfs -a を指定した HiRDB ファイルシステム領域の場合

  HiRDB ファイルを格納しているディスクに，必要な HiRDB ファイルシステム領域を確保する容量が不足していたり，使用する領域が HiRDB ファイルサイズの上限を超えたりすると，エラーとなります。

nouse：

RD エリアの自動増分機能を適用しません。

＜注意事項＞

1. 新たに追加される未使用セグメントは，RD エリアを構成する HiRDB ファイルの最終ファイルに追加されます。
2. HiRDB ファイルシステム領域の容量不足によって，未使用セグメントを確保できない場合，RD エリアの拡張又は再初期化をするか，データベース再編成ユティリティでディクショナリ表の再編成をしてください。
3. エクステントの数が上限値を超えた場合は，RD エリアがある HiRDB ファイルシステム領域のエクステントを統合するか，又は RD エリアに別の HiRDB ファイルシステム領域の HiRDB ファイルを追加してください。

   エクステントを統合するには，pdfbkup でバックアップを取得し，pdfmkfs で HiRDB ファイルシステム領域の初期設定をし，pdfrstr でバックアップから回復します。
4. RD エリア自動増分処理の開始から終了までの間，最終ファイルに対して排他が掛かります。

(7)　file name "HiRDB ファイルシステム領域名¥HiRDB ファイル名"

～((167 文字以内))

RD エリアに対して割り当てる，HiRDB ファイルシステム領域の名称と HiRDB ファイルの名称を指定します。

＜規則＞

1. HiRDB ファイルシステム領域名¥HiRDB ファイル名は引用符（"）で囲んで指定してください。
2. 引用符内に改行文字を含まないようにしてください。
3. 一つの RD エリアに対して，最大 16 個の HiRDB ファイルを割り当てられます。
4. HiRDB ファイルシステム領域名¥HiRDB ファイル名は，HiRDB システム内で重複しないようにしてください。

HiRDB ファイルシステム領域名　～＜パス名＞

HiRDB ファイルシステム領域の名称を指定します。

HiRDB ファイル名　～＜ HiRDB ファイル名＞((1～30 文字))

HiRDB ファイルの名称を指定します。pl で始まる名称は指定できません。

(8)　initial HiRDB ファイル内セグメント数 segments

～＜符号なし整数＞

HiRDB ファイルのセグメント数を指定します。ただし，HiRDB ファイルの大きさが 2 ギガバイトを超えるようなセグメント数は指定できません。

20.3.2　注意事項

1. RD エリアの数は，システム共通定義の RD エリアの最大数（pd_max_rdarea_no）を超えないようにしてください。また，RD エリアを構成する HiRDB ファイルの数も，RD エリアを構成する HiRDB ファイルの最大数（pd_max_file_no）を超えないようにしてください。超える場合は，pdchgconf コマンドでシステム共通定義を変更するか，又は HiRDB システムを正常終了させてから，システム共通定義を変更する必要があります。システム共通定義については，マニュアル「HiRDB システム定義」を参照してください。
2. 制御文に注釈を付けたい場合は，スラント，アスタリスク（/＊）とアスタリスク，スラント（＊/）で囲んで指定してください。

20.4　規則及び注意事項

20.4.1　規則

1. レジストリ機能初期設定ユティリティは，HiRDB が稼働中のときだけ実行できます。
2. レジストリ機能初期設定ユティリティは，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
3. レジストリ機能が使用できるようになるのは，レジストリ機能初期設定ユティリティの終了後，追加したレジストリ LOB 用 RD エリアにグローバルバッファが割り当てられたときです。レジストリ機能が使用可能となった後，各プラグインのレジストリ情報を登録できます。
4. レジストリ機能初期設定ユティリティの実行前に，あらかじめ HiRDB ファイルシステム領域をシングルサーバ，又はディクショナリサーバに定義しておく必要があります。HiRDB ファイルシステム領域は，pdfmkfs コマンドで初期設定します。
5. レジストリ機能初期設定ユティリティを実行する場合，クライアント環境変数（PDNAMEPORT，PDHOST，PDUSER（CONNECT 権限のある認可識別子））を設定しておく必要があります。クライアント環境変数の詳細については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
6. レジストリ機能初期設定ユティリティの終了後は，データベース複写ユティリティ（pdcopy）を使用して，必ず追加したレジストリ用 RD エリア，レジストリ LOB 用 RD エリア，マスタディレクトリ，及びデータディクショナリ用 RD エリアのバックアップを取得してください。

20.4.2　注意事項

1. pdreginit のリターンコードを次に示します。

   0：正常終了。

   4：正常終了（警告レベルのエラーがありましたが，処理は正常に終了しました）。

   8：異常終了。
2. レジストリ用 RD エリアとレジストリ LOB 用 RD エリアの追加，及びレジストリ管理表の作成は，データベース構成変更ユティリティ（pdmod）に制御が渡され処理されます。
3. レジストリ操作用ストアドプロシジャの登録は，データベース構成変更ユティリティが正常終了（KFPX24200-I メッセージのリターンコードが 0 又は 4）した後に実行されます。データベース構成変更ユティリティが異常終了（KFPX24200-I メッセージのリターンコードが 12）した場合は，RD エリアの追加処理，及びレジストリ管理表の作成処理は，すべてロールバックされます。
4. レジストリ操作用ストアドプロシジャの登録処理でエラーが発生した場合は，次回のレジストリ機能初期設定ユティリティの実行時に，-k オプションでレジストリ操作用ストアドプロシジャの登録だけの指定をする必要があります。レジストリ操作用ストアドプロシジャを登録しないで，レジストリ用 RD エリア及びレジストリ LOB 用 RD エリアの追加を指定するとエラーになります。

5. pdntenv コマンドで文字コードとして utf-8，又は utf-8_ivs を選択した場合，制御文ファイル中に BOM が存在しても問題ありません。ただし，制御文ファイル中のコメントには，ASCII コード以外の文字コードを使用できません。ASCII コード以外の文字コードを使用した場合，pdreginit が誤動作するおそれがあるため，使用しないでください。

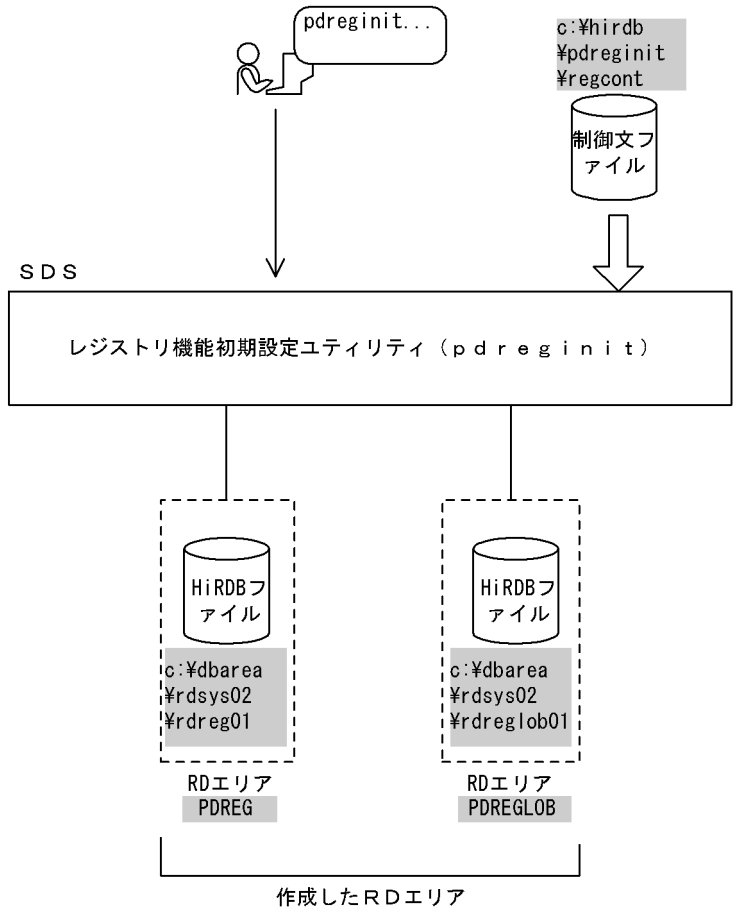
20.5　使用例

レジストリ機能初期設定ユティリティの使用例を，例 1 に示します。

例 1

次に示すRDエリアを作成します。
　・PDREG（レジストリ用RDエリア）
　・PDREGLOB（レジストリLOB用RDエリア）
なお，既に次のHiRDBファイルシステム領域が作成されているものとします。
　・c:¥dbarea¥rdsys02
　・c:¥dbarea¥rdsys03

[概要]



（凡例）　：レジストリ機能初期設定ユティリティでの指定値
　　ＳＤＳ：シングルサーバ

[コマンド実行例]

```
pdreginit -a c:¥hirdb¥pdreginit¥regcont
```

制御文ファイル（c:¥hirdb¥pdreginit¥regcont）の内容

```
/* レジストリ用RDエリアの定義 */
 create rdarea PDREG globalbuffer gb1   ...................1
               for registry   ..............................2
               page 4096 characters   ......................3
```

```
                storage control segment 50 pages   ........4
                file name ”c:¥dbarea¥rdsys02¥rdreg01”
                    initial 50 segments ;   ..............5
/* レジストリLOB用RDエリアの定義 */
 create rdarea PDREGLOB globalbuffer gb2   ..............6
                for LOB used by HiRDB(SQL_REGISTRY)   .....7
                page 8192 characters   .....................8
                storage control segment 1 pages   .........9
                file name ”c:¥dbarea¥rdsys03¥rdreglob01”
                    initial 1000 segments ;   ............10
```

[説明]

1. RD エリアの名称　**PDREG**
2. RD エリアの種類　**レジストリ用 RD エリア**（レジストリ管理情報を格納）
3. ページ長　**4096 バイト**
4. セグメントサイズ　**50 ページ**
5. RD エリアを構成する HiRDB ファイルの指定
   名称　**c:¥dbarea¥rdsys02¥rdreg01**
   セグメント数　**50**
6. RD エリアの名称　**PDREGLOB**
7. RD エリアの種類　**レジストリ LOB 用 RD エリア**
8. ページ長　**8192 バイト**
9. セグメントサイズ　**1 ページ**
10. RD エリアを構成する HiRDB ファイルの指定
    名称　**c:¥dbarea¥rdsys03¥rdreglob01**
    セグメント数　**1000**

21 JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput）

この章では，JP1/NETM/Audit 用監査ログ出力ファイルに，監査証跡表のデータを出力するためのユティリティ（pdaudput）について説明します。

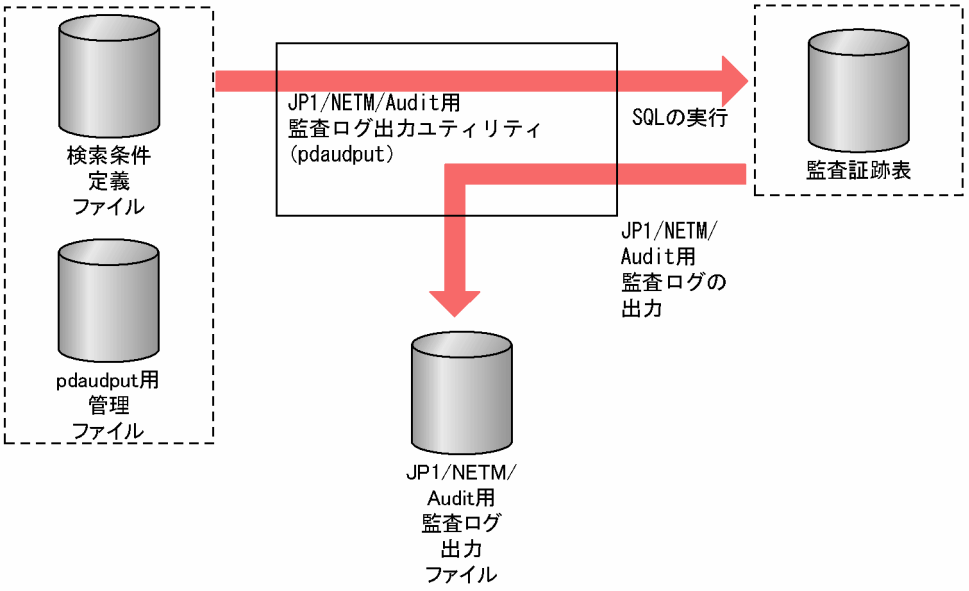
21.1　概要

JP1/NETM/Audit 用監査ログ出力ユティリティは，HiRDB の監査証跡表のデータを，JP1/NETM/Audit で読み込める形式でファイルに出力するユティリティです。JP1/NETM/Audit 用監査ログ出力ユティリティが出力する監査証跡表のデータを JP1/NETM/Audit 用監査ログといい，出力されたファイルを，JP1/NETM/Audit 用監査ログ出力ファイルといいます。

なお，HiRDB のバージョンが 08-04 以前の時点から存在する監査証跡表のデータは，pdaudput で出力されません。

JP1/NETM/Audit 用監査ログ出力ユティリティの概要を次に示します。

図 21–1　JP1/NETM/Audit 用監査ログ出力ユティリティの概要



図中のファイルについては，「JP1/NETM/Audit 用監査ログ出力ユティリティで使用するファイル」を参照してください。

21.1.1　前提条件及び動作環境

JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput）を実行するための前提条件，及び動作環境について説明します。

表 21–1　pdaudput 実行時の前提条件

項目	前提条件
実行権限	HiRDB 管理者が，-u オプションで監査人の認可識別子を指定してコマンドを実行してください。実行すると，パスワード入力要求のメッセージが表示されます。このとき，監査人がパスワードを入力してください。
同時実行	pdaudput は同時実行できません。

表 21–2　pdaudput 実行時の動作環境

項目		動作環境
HiRDB 環境	ユティリティ実行場所	pdaudput は，シングルサーバ又はシステムマネジャがあるサーバマシンで実行してください。
	HiRDB の状態	pdaudput は，HiRDB が稼働中のときだけ実行できます。
	監査証跡表へのインデクスの作成	監査証跡表の PDLOAD_TIMESTAMP，PDLOAD_SEQNUM の列の値を昇順に取得するため，PDLOAD_TIMESTAMP，PDLOAD_SEQNUM の順に複数列インデクスを作成しておくことをお勧めします。 インデクスを作成しないと，pdaudput の実行性能が低下するおそれがあります。
	RD エリアの状態	pdaudput の対象であるディクショナリ用 RD エリア，及び監査証跡表格納 RD エリアが，閉塞解除かつオープン状態である必要があります。 pdaudput と処理対象 RD エリアの状態の関係については，「RD エリアの状態によるユティリティ及び UAP の実行可否」を参照してください。
OS 環境	-d オプションで指定するファイル及びディレクトリ	次のように権限を付与してください。 • JP1/NETM/Audit 用監査ログ出力ファイル HiRDB 管理者に読み込み及び書き込み権限，JP1/NETM/Audit で収集を行う OS ユーザに読み取り権限 • JP1/NETM/Audit 用監査ログ出力ファイル出力先ディレクトリ HiRDB 管理者及び JP1/NETM/Audit で収集を行う OS ユーザに読み込み権限 なお，-d オプションを省略した場合は，$PDDIR/auditlog%PDDIR%¥auditlog にファイルを作成すると解釈されるため，同様にアクセス権限を与えてください。また，パス名は大文字と小文字を区別しません。このため，大文字と小文字が異なるだけの同じパス名を指定した場合は，上書きされます。
	-w オプションで指定するファイル及びディレクトリ	HiRDB 管理者に対して読み込み権限を与えてください。
	クライアント環境定義	pdaudput を実行する前に，クライアント環境定義を設定してください。クライアント環境定義については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

21.1.2　JP1/NETM/Audit 用監査ログ出力ユティリティで使用するファイル

pdaudput では，次のファイルを使用します。

- JP1/NETM/Audit 用監査ログ出力ファイル
- pdaudput 用管理ファイル
- 検索条件定義ファイル

それぞれのファイルについて説明します。

(1)　JP1/NETM/Audit 用監査ログ出力ファイル

監査証跡のデータを，JP1/NETM/Audit で読み込める形式で出力したファイルです。

- 出力先のサーバマシン
  シングルサーバ又はシステムマネジャがあるサーバマシンに出力されます。
- ファイルの出力先及び名称
  -d オプションで指定したディレクトリに auditmsg1.log 及び auditmsg2.log という名称のファイルが作成されます。-d オプションの指定を省略した場合は，$PDDIR/auditlog%PDDIR%¥auditlog が仮定されます。
- ファイルの所有者及びアクセス権限
  ファイルの所有者及びアクセス権限については，「-d 出力ディレクトリ」を参照してください。
- 使用できないファイル
  ・BOM を持つファイル
  ・ダイレクトディスクアクセス（raw I/O）
  ・キャラクタ型スペシャルファイル
  ・ラージファイル

(2)　pdaudput 用管理ファイル

pdaudput 用管理ファイルは，ユティリティが使用するファイルです。JP1/NETM/Audit 用監査ログ出力ファイルの面数や，監査証跡データの出力履歴を管理します。pdaudput を再実行する場合，このファイルの履歴を基にして，監査証跡表から出力するデータを選択します。

- 出力先のサーバマシン
  シングルサーバ又はシステムマネジャがあるサーバマシンに出力されます。
- ファイルの出力先及び名称
  -d オプションで指定したディレクトリに pdaudputctrl という名称のファイルが作成されます。-d オプションの指定を省略した場合は，$PDDIR/auditlog%PDDIR%¥auditlog が仮定されます。
- ファイルの所有者及びアクセス権限
  ファイルの出力先の上位ディレクトリの設定が引き継がれます。
  ファイルの所有者は，HiRDB 管理者とそのプライマリグループです。アクセス権限は「600」を付与します。
- 使用できないファイル
  ・BOM を持つファイル
  ・ダイレクトディスクアクセス（raw I/O）
  ・キャラクタ型スペシャルファイル
  ・ラージファイル

(3)　検索条件定義ファイル

JP1/NETM/Audit 用監査ログ出力ファイルに出力する情報を定義するためのファイルです。監査人が作成します。詳細は，「-w 検索条件定義ファイル」を参照してください。

21.1.3　JP1/NETM/Audit 用監査ログ出力ファイルの内容

(1)　JP1/NETM/Audit 用監査ログ出力ファイルの形式

JP1/NETM/Audit 用監査ログ出力ファイルの形式を次に示します。

```
ファイルのヘッダ情報
JP1/NETM/Audit用監査ログ
JP1/NETM/Audit用監査ログ
・・・
```

ファイルの先頭に，システムが使用するファイルのヘッダ情報として 512 バイト分のバイナリデータを出力します。ファイルのヘッダ情報の下に JP1/NETM/Audit 用監査ログを出力します。JP1/NETM/Audit 用監査ログは，HiRDB が出力する 1 レコードの監査証跡に対し，1 メッセージ出力します。

(2)　JP1/NETM/Audit 用監査ログの形式

```
CALFHM 1.0,出力項目1=値1,出力項目2=値2,出力項目3=値3,・・・出力項目n=値n
```

CALFHM 1.0 はヘッダ情報です。すべての JP1/NETM/Audit 用監査ログの先頭に，共通で出力されます。出力項目の説明については，「JP1/NETM/Audit 用監査ログの出力項目」を参照してください。

（例）

監査証跡が「USERA がコネクトに失敗した場合」の JP1/NETM/Audit 用監査ログの出力例を示します。

```
CALFHM 1.0,seqnum=1,msgid=-561,date=2007-01-22T16:09:59.884+09:00,
progid=HiRDB,compid=CCC,pid=1234,ocp:host=host01,
ctgry=Authentication,result=Failure,subj:uid=USERA,
op=CNT,from:ipv4=111.222.333.444,to:host=host01,outp:host=SRV1,
subjp:ipv4=111.222.333.444,loc=HRD1,msg=”2007-01-25 16:09:59.00,1”
```

実際に出力されるメッセージは，改行されません。

(3)　JP1/NETM/Audit 用監査ログの出力項目

JP1/NETM/Audit 用監査ログの出力項目と HiRDB の監査証跡表の列との対応を次に示します。

表 21–3　JP1/NETM/Audit 用監査ログの出力項目と HiRDB が管理する監査証跡表で該当する列との対応

JP1/NETM/Audit 用監査ログの出力項目		出力項目の概要	HiRDB が管理する監査証跡表で該当する列
ヘッダ情報	共通仕様識別子	監査ログ共通フォーマット識別子	なし
	共通仕様リビジョン番号	監査ログ共通フォーマット管理リビジョン	なし
共通情報	通番	JP1/NETM/Audit 用監査ログの通番	なし
	メッセージ ID	WS/PC 製品共通メッセージ	SQL コード／終了コード
	日付・時刻	JP1/NETM/Audit 用監査ログを出力した日時，タイムゾーン	イベント実行日
			イベント実行時刻
			イベント実行時刻（マイクロ秒）
	プログラム名	監査対象イベントが発生したプログラムの名称	なし
	コンポーネント名	監査対象イベントが発生したコンポーネントの名称	ユニット識別子
	プロセス ID	監査対象イベントが発生したプロセスのプロセス ID	プロセス ID
	発生場所	監査対象イベントが発生した場所の情報	ホスト名
	監査対象イベントの種別	監査対象イベントのカテゴリ名	イベントタイプ
	監査対象イベントの結果	監査対象イベントの結果（成功，失敗，又は発生）	イベント成否
	サブジェクト識別情報	監査対象イベントを発生させた利用者情報，及び利用者を 1：1 で対応させた識別情報，又は利用者を代行するプロセス情報（利用者に関連しない場合はプロセス情報）	イベント実行者
固有情報	オブジェクト情報	該当する事象	オブジェクトの種別
	動作情報	サブジェクトが指示したオブジェクトに対する行為	イベントサブタイプ
	オブジェクトロケーション情報	オブジェクト情報を特定するための位置情報	オブジェクトの所有者名
			オブジェクトの名称
	変更前情報	変更前の情報	セキュリティ関連変更種別
			セキュリティ関連変更前定義値
	変更後情報	変更後の情報	セキュリティ関連変更種別
			セキュリティ関連変更後定義値

JP1/NETM/Audit 用監査ログの出力項目		出力項目の概要	HiRDB が管理する監査証跡表で該当する列
	権限情報	監査対象イベントを発生させたサブジェクトがアクションを取ったときの権限	使用した権限
	サービスインスタンス名	日立ミドル製品が提供するサービスに対するサービス利用者の識別子	関連製品付加情報
	リクエスト送信元ホスト	監査対象イベントが，複数のプログラム間での連係動作に関連する場合の，リクエストの送信元の場所の情報，及び，リクエストの送信先の場所の情報	IP アドレス
	リクエスト送信先ホスト		ホスト名
	出力元の場所	出力元が動作している場所の情報	サーバ名
	指示元の場所	サブジェクトが指示を出した場所の情報	IP アドレス
	ロケーション識別情報	顧客が設定したロケーション識別情報	なし
	自由記述	監査対象イベントの内容を示すメッセージなどの記述	pdload 実行時刻印
			pdload 内通番

JP1/NETM/Audit 用監査ログの出力項目の詳細について説明します。なお，共通情報とは，監査ログを出力する製品で共通して出力される項目です。固有情報とは，監査ログを出力する製品ごとに出力される項目です。

(a) ヘッダ情報

ヘッダ情報に出力される項目について説明します。

- 共通仕様識別子
  CALFHM が表示されます。
- 共通仕様リビジョン番号
  1.0 が表示されます。

（例）

1 レコードにつき，次のヘッダ情報が出力されます。

```
CALFHM 1.0
```

(b) 共通情報

共通情報に出力される項目について説明します。

- 通番

pdaudput を実行するたびに，1〜10 けたの 10 進数値（1〜（2^{31}–1））を出力行ごとにカウントした値が SequenceNum に出力されます。最大値 2^{31}–1 に達した場合は，1 に戻ります。出力形式を次に示します。

```
seqnum=SequenceNum
```

- メッセージ ID

  SQL コード及びリターンコードを n に出力します。SQL コード及びリターンコードが NULL 値の場合，ハイフン（–）を出力します。出力形式を次に示します。

```
msgid=n
```

- 日付・時刻

  次の値が出力されます。

値	内容
YYYY-MM-DD	イベント実行日が出力されます。
T	区切り記号です。
hh:mm:ss.sss	hh:mm:ss にイベント実行時刻が出力されます。また，sss にはイベント実行時刻（マイクロ秒）がミリ秒に切り捨てて設定されます。
TZD	UTC（Coordinated Universal Time）からのオフセットを示したタイムゾーン指定です。 “+hh:mm”は UTC から hh:mm だけ進んでいることを示します。“-hh:mm”は UTC から hh:mm だけ遅れていることを示します。“Z”は UTC と同じであることを示します。

  出力形式を次に示します。

```
date=YYYY-MM-DDThh:mm:ss.sssTZD
```

- プログラム名

  HiRDB が表示されます。出力形式を次に示します。

```
progid=HiRDB
```

- コンポーネント名

  ユニット識別子を ComponentID に出力します。ユニット識別子が NULL 値の場合，ハイフン（–）を出力します。出力形式を次に示します。

```
compid=ComponentID
```

- プロセス ID

  イベント発行元のプロセス ID を，INTEGER 型から文字列に変換して ProcID に出力します。イベント発行元のプロセス ID が NULL 値の場合，「0」を出力します。

```
pid=ProcID
```

- 発生場所

イベント発行元のホスト名を Hostname に出力します。イベント発行元のホスト名が NULL 値の場合，「0」を出力します。

```
ocp:host=Hostname
```

- 監査対象イベントの種別

  HiRDB が出力するイベントタイプ及びイベントサブタイプに応じて，監査対象イベントを EventCategory に出力します。次の表に，HiRDB が出力するイベントタイプ及びイベントサブタイプと，pdaudput が出力する監査対象イベントの種別についての対応を示します。なお，次の表が該当するのはイベント終了証跡を監査対象とする場合だけです。権限チェック時の監査証跡の場合は，AccessControl を出力します。

表 21–4　HiRDB が出力するイベントタイプ及びイベントサブタイプと，pdaudput が出力する監査対象イベントの種別についての対応

HiRDB が出力するイベントタイプ，及びイベントサブタイプ				pdaudput が出力する値，及び説明	
監査イベント	イベントタイプの値	イベントサブタイプの値	イベント	EventCategory に出力する値	説明
システム管理者セキュリティイベント	SYS	STR	HiRDB の開始	StartStop	起動・停止
		STP	HiRDB の終了	StartStop	起動・停止
		MOD	主監査人の登録，又は監査証跡表の作成（pdmod コマンド）	ConfigurationAccess	構成定義
		MDA	副監査人の登録（pdmod コマンド）	ConfigurationAccess	構成定義
		ARM	監査証跡ファイルの削除（pdaudrm コマンド）	ConfigurationAccess	構成定義
		ABG	監査証跡の取得開始（pdaudbegin コマンド，HiRDB 起動）	StartStop	起動・停止
		AEN	監査証跡の取得終了（pdaudend コマンド，HiRDB 停止）	StartStop	起動・停止
		OVW	監査証跡ファイルの上書き開始	ConfigurationAccess	構成定義
		CLK	連続認証失敗アカウントロック状態への遷移	Authentication	識別・認証
		CUL	連続認証失敗アカウントロック状態の解除	ConfigurationAccess	構成定義
		PLK	パスワード無効アカウントロック状態への遷移	Authentication	識別・認証

HiRDB が出力するイベントタイプ，及びイベントサブタイプ				pdaudput が出力する値，及び説明	
監査イベント	イベントタイプの値	イベントサブタイプの値	イベント	EventCategory に出力する値	説明
		PUL	パスワード無効アカウントロック状態の解除	ConfigurationAccess	構成定義
		SPR	CONNECT 関連セキュリティ機能に関するオペランドの指定値変更	ConfigurationAccess	構成定義
		ULK	pdacunlck コマンドの実行	ConfigurationAccess	構成定義
監査人セキュリティイベント	AUD	ALD	監査人による監査証跡表へのデータロード（pdload コマンド）	ConfigurationAccess	構成定義
		ATL	HiRDB による監査証跡表へのデータロード（pdload コマンド）	ConfigurationAccess	構成定義
		ASW	監査証跡ファイルのスワップ（pdaudswap コマンド）	ConfigurationAccess	構成定義
		CRT	監査対象イベントの定義（CREATE AUDIT）	ConfigurationAccess	構成定義
		DRP	監査対象イベントの削除（DROP AUDIT）	ConfigurationAccess	構成定義
		GRT	監査人のパスワード変更（GRANT AUDIT）	ConfigurationAccess	構成定義
		ATB	監査証跡表の自動データロード機能再開始（pdaudatld -b コマンド）	ConfigurationAccess	構成定義
		ATT	監査証跡表の自動データロード機能停止（pdaudatld -t コマンド，データロード失敗）	ConfigurationAccess	構成定義
		APT	JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput ユティリティ）	ConfigurationAccess	構成定義
セッションセキュリティイベント	SES	CNT	HiRDB への接続（CONNECT 文）	Authentication	識別・認証
		ATH	ユーザの変更（SET SESSION AUTHORIZATION 文）	Authentication	識別・認証
		ROL	ロールの設定（SET ROLE 文）	Authentication	識別・認証

HiRDB が出力するイベントタイプ，及びイベントサブタイプ				pdaudput が出力する値，及び説明	
監査イベント	イベントタイプの値	イベントサブタイプの値	イベント	EventCategory に出力する値	説明
		DIS	HiRDB との切り離し（DISCONNECT 文）	Authentication	識別・認証
権限管理イベント	PRV	GRT	ユーザ権限の付与（GRANT 文）	AccessControl	アクセス制御
			パスワードの変更（CONNECT 関連セキュリティ機能の設定変更種別にパスワードの変更であることが分かる情報を付与する）		
		RVK	ユーザ権限の削除（REVOKE 文）	AccessControl	アクセス制御
オブジェクト定義イベント	DEF	CRT	オブジェクトの定義（CREATE 系 SQL）	ContentAccess	重要情報アクセス
		DRP	オブジェクトの削除（DROP 系 SQL）	ContentAccess	重要情報アクセス
		ALT	オブジェクトの変更（ALTER 系 SQL）	ContentAccess	重要情報アクセス
オブジェクト操作イベント	ACS	SEL	表の検索（SELECT 文）	ContentAccess	重要情報アクセス
		INS	表への行挿入（INSERT 文）	ContentAccess	重要情報アクセス
		UPD	表の行更新（UPDATE 文）	ContentAccess	重要情報アクセス
		DEL	表からの行削除（DELETE 文）	ContentAccess	重要情報アクセス
		PRG	表の全行削除（PURGE TABLE 文）	ContentAccess	重要情報アクセス
		CAL	ストアドプロシジャの実行（CALL 文）	ContentAccess	重要情報アクセス
		LCK	表の排他制御（LOCK TABLE 文）	ContentAccess	重要情報アクセス
		ASN	リストの作成（ASSIGN LIST 文）	ContentAccess	重要情報アクセス
ユティリティ操作イベント	UTL	LOD	データベース作成ユティリティ（pdload コマンド）	ContentAccess	重要情報アクセス
		ORG	データベース再編成ユティリティ（pdrorg コマンド）	ContentAccess	重要情報アクセス

HiRDB が出力するイベントタイプ，及びイベントサブタイプ				pdaudput が出力する値，及び説明	
監査イベント	イベントタイプの値	イベントサブタイプの値	イベント	EventCategory に出力する値	説明
		EXP	ディクショナリ搬出入ユティリティ（pdexp コマンド）	ContentAccess	重要情報アクセス
			pddefrev コマンド		
		CST	整合性チェックユティリティ（pdconstck コマンド）	ContentAccess	重要情報アクセス

出力形式を次に示します。

```
ctgry=EventCategory
```

- 監査対象イベントの結果

  イベント成否の結果を EventResult に出力します。イベント成否と出力される値を次の表に示します。

  表 21–5　イベント成否と出力される値

イベント成否	出力される値
成功	Success
失敗	Failure
一部失敗	Failure

出力形式を次に示します。

```
result=EventResult
```

- サブジェクト識別情報

  ユーザ識別子を出力します。実行したイベントが SQL イベントの場合は，AccountID に HiRDB が管理しているユーザ ID が出力されます。実行したイベントがコマンド及びユティリティイベントの場合は，Effective UserID に OS ユーザ名が出力されます。なお，OS ユーザ名は 30 バイトまで出力し，31 バイト以降は切り捨てます。

  なお，ユーザ識別子のデータ長が 0 の場合は「SYSTEM」と出力されます。

  次の表に，イベント及びイベントタイプとサブジェクト識別情報の対応を示します。

  表 21–6　イベント及びイベントタイプとサブジェクト識別情報の対応

イベントの種類			サブジェクト識別情報
監査イベント	イベントタイプの値	イベントサブタイプの値	
システム管理者セキュリティイベント	SYS	STR	Effective UserID
		STP	Effective UserID
		MOD	Effective UserID

イベントの種類			サブジェクト識別情報
監査 イベント	イベントタイプの値	イベントサブタイプの値	
		MDA	Effective UserID
		ARM	Effective UserID
		ABG	Effective UserID
		AEN	Effective UserID
		OVW	Effective UserID
		CLK	AccountID
		CUL	AccountID
		PLK	AccountID
		PUL	AccountID
		SPR	AccountID
		ULK	Effective UserID
監査人セキュリティイベント	AUD	ALD	Effective UserID
		ATL	Effective UserID
		ASW	Effective UserID
		CRT	AccountID
		DRP	AccountID
		GRT	AccountID
		ATB	Effective UserID
		ATT	Effective UserID
		APT	Effective UserID
セッションセキュリティイベント	SES	CNT	AccountID
		ATH	AccountID
		ROL	AccountID
		DIS	AccountID
		CMT	AccountID
		RLB	AccountID
権限管理イベント	PRV	GRT	AccountID
		RVK	AccountID
オブジェクト定義イベント	DEF	CRT	AccountID

イベントの種類			サブジェクト識別情報
監査イベント	イベントタイプの値	イベントサブタイプの値	
		DRP	AccountID
		ALT	AccountID
オブジェクト操作イベント	ACS	SEL	AccountID
		INS	AccountID
		UPD	AccountID
		DEL	AccountID
		PRG	AccountID
		CAL	AccountID
		LCK	AccountID
		ASN	AccountID
ユティリティ操作イベント	UTL	LOD	Effective UserID
		ORG	Effective UserID
		EXP	Effective UserID
		CST	Effective UserID

SQL イベントを実行した場合の出力形式を次に示します。

```
subj:uid=AccountID
```

コマンド及びユティリティイベントを実行した場合の出力形式を次に示します。

```
subj:euid=Effective UserID
```

(c) 固有情報

- オブジェクト情報

  オブジェクトの種別を Object に出力します。オブジェクト種別が NULL 値の場合は出力しません。出力形式を次に示します。

  ```
  obj=Object
  ```

- 動作情報

  表「HiRDB が出力するイベントタイプ及びイベントサブタイプと，pdaudput が出力する監査対象イベントの種別についての対応」に示した，イベントサブタイプの値が Operation に出力されます。出力形式を次に示します。

  ```
  op=Operation
  ```

- オブジェクトロケーション情報

オブジェクト情報を特定するためのオブジェクト所有者，及びオブジェクト名称を ObjectLocation に出力します。NULL 値の場合はどちらも出力しません。

オブジェクト所有者の出力形式を次に示します。

```
objloc:user=ObjectLocation
```

オブジェクト名称の出力形式を次に示します。

```
objloc:name=ObjectLocation
```

- 変更前情報

  セキュリティ関連変更種別及びセキュリティ関連変更前定義値を Before に出力します。「セキュリティ関連変更種別_セキュリティ関連変更前定義値」の形式で出力します。

  ただし，セキュリティ関連変更前定義値が NULL 値の場合は，セキュリティ関連変更前定義値の部分にはハイフン（-）を出力します。出力形式を次に示します。

  ```
  before=Before
  ```

- 変更後情報

  セキュリティ関連変更種別及びセキュリティ関連変更後定義値を After に出力します。「セキュリティ関連変更種別_セキュリティ関連変更後定義値」の形式で出力します。

  ただし，セキュリティ関連変更後定義値が NULL 値の場合は，セキュリティ関連変更後定義値の部分にはハイフン（-）を出力します。出力形式を次に示します。

  ```
  after=After
  ```

- 権限情報

  使用した権限を Authority に出力します。使用した権限が‘△△△’（空白 3 文字）の場合は出力しません。出力形式を次に示します。

  ```
  auth=Authority
  ```

- サービスインスタンス名

  関連製品付加情報を ServiceInstance に出力します。関連製品付加情報が NULL 値の場合は出力しません。出力形式を次に示します。

  ```
  sins=ServiceInstance
  ```

- リクエスト送信元ホスト

  リクエスト送信元ホストを IP アドレス（IPv4）で aaa.bbb.ccc.ddd に出力します。イベント発行元の IP アドレスが NULL 値の場合は出力しません。出力形式を次に示します。

  ```
  from:ipv4=aaa.bbb.ccc.ddd
  ```

- リクエスト送信先ホスト

  リクエスト送信先ホストをホスト名で Hostname に出力します。イベント受け付け元のホスト名が NULL 値の場合は出力しません。出力形式を次に示します。

  ```
  to:host=Hostname
  ```

- 出力元の場所

  出力元の場所のサーバ名を，ホスト名形式で Hostname に出力します。サーバ名称が NULL 値の場合は出力しません。出力形式を次に示します。

  ```
  outp:host=Hostname
  ```

- 指示元の場所

  指示元の場所を IP アドレス（IPv4）で出力します。イベント発行元の IP アドレスが NULL 値の場合は出力しません。出力形式を次に示します。

  ```
  subjp:ipv4=aaa.bbb.ccc.ddd
  ```

- ロケーション識別情報

  HiRDB 識別子を Location に出力します。出力形式を次に示します。

  ```
  loc=Location
  ```

- 自由記述

  PDLOAD_TIMESTAMP 値を YYYY-MM-DD hh:mm:ss.nn に，PDLOAD_SEQNUM 値を pdloadseqnum に出力します。どの監査証跡が JP1/NETM/Audit 用監査ログ出力ファイルに出力されたかについて，この項目の値，監査証跡表の PDLOAD_TIMESTAMP 値，及び PDLOAD_SEQNUM 値を用いて確認できます。出力形式を次に示します。

  ```
  msg=”YYYY-MM-DD hh:mm:ss.nn,pdloadseqnum”
  ```

21.1.4 監査証跡の出力に関する留意事項

- pdaudput を連続して実行する場合は，pdaudput 用管理ファイルに記載された，最後に JP1/NETM/Audit 用監査ログ出力ファイルを取得した日時以降の監査証跡を対象とします。
- 監査証跡のデータを 1 レコード分出力するごとに，1 メッセージが出力されます。

21.2　コマンドの形式

21.2.1　形式

```
pdaudput 〔-u 認可識別子〕
         〔-k 〔r｜i｜m〕〕
         〔-d 出力ディレクトリ〕
         〔-s ファイルサイズ〕
         〔-w 検索条件定義ファイル〕
```

21.2.2　オプション

(1)　-u 認可識別子

監査人の認可識別子を指定します。

このオプションを指定した場合，パスワード入力要求の応答付きメッセージが表示されます。そのときに応答した認可識別子とパスワードで HiRDB システムに接続し，その認可識別子について監査証跡表へのアクセス権限があるかどうかがチェックされます。

このオプションは，パスワードを応答できない環境では指定しないでください。

このオプションを省略した場合，次の方法で認可識別子とパスワードが仮定されます。

- JP1/NETM/Audit 用監査ログ出力ユティリティ実行時の環境変数 PDUSER の設定値が仮定されます。シェルで&を付けてバックグラウンドで実行する場合や，リモートシェル環境で実行する場合など，パスワードの応答ができない環境で実行するときは，PDUSER を必ず設定してください。環境変数 PDUSER の指定例を次に示します。

  （例）C シェルの場合

  setenv PDUSER '"認可識別子"/"パスワード"'

  PDUSER=¥""認可識別子"¥"/¥""パスワード"¥"

  環境変数 PDUSER が設定されていない場合，このユティリティは実行できません。

  認可識別子を引用符（"）で囲んだ場合は，英大文字と英小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。

  sh（Bourne シェル），csh（C シェル），及び ksh（K シェル）の場合は，更にアポストロフィ（’）で囲む必要があります。

  認可識別子を「¥""」と「"¥"」で囲んだ場合は，英大文字と英小文字が区別されます。¥''で囲まない場合は，すべて英大文字として扱われます。

(2) -k〔r｜i｜m〕

実行モードを指定します。

r：

JP1/NETM/Audit 用監査ログを JP1/NETM/Audit 用監査ログ出力ファイルに出力する場合に指定します。通常指定する実行モードです。

-d オプションで指定するディレクトリ下に pdaudput 用管理ファイル，及び JP1/NETM/Audit 用監査ログ出力ファイルがない場合は，自動的にファイルが作成されます。

i：

pdaudput 用管理ファイル，及び JP1/NETM/Audit 用監査ログ出力ファイルを新規作成又は再作成します。

ファイルが既に存在する場合はメッセージが出力されます。メッセージに対して y を指定するとファイルを再作成します。y 以外を指定した場合は，リターンコード 0 で，一切処理を行わないで終了します。

＜適用基準＞

JP1/NETM/Audit 用監査ログ出力ファイルの出力先を変更したい場合などに指定します。-d オプションで指定した出力先のファイルが処理の対象となります。

＜注意事項＞

- このオプションを指定しただけでは JP1/NETM/Audit 用監査ログは出力されません。JP1/NETM/Audit 用監査ログを出力したい場合は続けて-k r を指定してください。
- 既存の pdaudput 用管理ファイル，及び JP1/NETM/Audit 用監査ログ出力ファイルの情報は削除されます。
- 再作成した pdaudput 用管理ファイル，及び JP1/NETM/Audit 用監査ログ出力ファイルの所有者やアクセス権限は，削除したファイルのものを引き継ぎます。

m：

JP1/NETM/Audit 用監査ログ出力ファイルだけを再作成します。

＜適用基準＞

JP1/NETM/Audit 用監査ログ出力ファイルの容量などを変更したい場合に指定します。

＜注意事項＞

- このオプションを指定しただけでは JP1/NETM/Audit 用監査ログは出力されません。JP1/NETM/Audit 用監査ログを出力したい場合は続けて-k r を指定してください。
- 既存の pdaudput 用管理ファイル，及び JP1/NETM/Audit 用監査ログ出力ファイルの情報は削除されます。
- -d オプションで指定したディレクトリ下に JP1/NETM/Audit 用監査ログ出力ファイルが格納されていない場合は，エラーとなります。

(3)　-d 出力ディレクトリ

～＜パス名＞((254 文字以内))《$PDDIR/auditlog%PDDIR%¥auditlog》

JP1/NETM/Audit 用監査ログ出力ファイルを出力するファイルを，指定したディレクトリに作成します。ディレクトリは，絶対パス名で指定します。

＜規則＞

- このオプションを省略した場合は，$PDDIR/auditlog%PDDIR%¥auditlog が仮定されます。その場合，ディレクトリの所有者やアクセス権限は，上位ディレクトリと同様です。ディレクトリの所有者は，HiRDB の管理者とそのプライマリグループです。アクセス権限は任意に変更できますが，変更する場合は表「pdaudput 実行時の動作環境」で示すユーザがアクセスできるようにアクセス権限を付与してください。
- このオプションで指定したディレクトリが存在しない場合は，エラーとなります。
- このオプションで指定したディレクトリには，auditmsg1.log，及び auditmsg2.log という名称のファイルが作成されます。
- auditmsg1.log，又は auditmsg2.log のどちらかの容量がファイルサイズの上限に達すると，古い方のファイルの情報を削除して，スワップします。そのとき，ユティリティが停止します。続けて JP1/NETM/Audit 用監査ログ出力ファイルを出力する場合は，再度ユティリティを実行してください。なお，スワップ先のファイルの内容は削除されます。

(4)　-s ファイルサイズ

～＜符号なし整数＞((4～2000000))《256000》

JP1/NETM/Audit 用監査ログ出力ファイルの 1 ファイルの最大サイズを指定します。単位はキロバイトです。

(5)　-w 検索条件定義ファイル

～＜パス名＞((254 文字以内))《インストールディレクトリ/sample/sampleAUDIT/sampleaud1 インストールディレクトリ¥sample¥ sampleAUDIT¥sampleaud1》

監査証跡表を検索する条件を指定したファイル（検索条件定義ファイル）を，絶対パス名で指定します。このオプションの指定を省略した場合は，インストールディレクトリ/sample/sampleAUDIT/sampleaud1 インストールディレクトリ¥sample¥ sampleAUDIT¥sampleaud1 に示す検索条件が仮定されます。インストールディレクトリ/sample/sampleAUDIT/sampleaud1 インストールディレクトリ¥sample¥ sampleAUDIT¥sampleaud1 の内容については，＜規則＞で説明します。

＜規則＞

1. 検索条件として指定できる列については，マニュアル「HiRDB システム運用ガイド」の「監査証跡表の列構成」を参照してください。
2. 検索条件定義ファイルには，次のファイルは使用できません。
   ・BOM を持つファイル

・ダイレクトディスクアクセス（raw I/O）
・キャラクタ型スペシャルファイル
・ラージファイル

3. 検索条件定義ファイルは，シングルサーバ又はシステムマネジャがあるサーバマシンに作成してください。

4. 検索条件定義ファイルには，SQL で監査証跡の探索条件を記入します。記入例を次に示します。

（例 1）

次の探索条件を指定すると，認可識別子 USERA が所有する表 T1 にアクセスしたイベントに関する監査証跡を出力できます。

```
OBJECT_SCHEMA ='USERA' AND OBJECT_NAME ='T1' AND OBJECT_TYPE ='TBL'
```

（例 2）

検索条件定義ファイルのサンプルファイル（sampleaud1）を指定すると，システム管理セキュリティイベント，及び監査人セキュリティイベントに関する監査証跡を出力できます。

```
EVENT_TYPE IN('SYS','AUD')
```

（例 3）

検索条件定義ファイルのサンプルファイル（sampleaud2）を指定すると，セッションセキュリティイベントに関して失敗した監査証跡を出力できます。

```
EVENT_TYPE='SES' AND EVENT_RESULT IN('F','U')
```

21.3　指定できるオプションの組み合わせ

-k オプションの指定値によって，指定が無効になる場合があります。次の表に-k オプションとほかのオプションの組み合わせについて示します。

-k オプションに指定する実行モード	-u	-d	-s	-w
r	○	○	△	○
i	○	○	○	×
m	○	○	○	×

（凡例）

○：指定が有効です。

△：pdaudput 用管理ファイルが存在しない場合は，pdaudput 用管理ファイルを作成するため，指定が有効になります。pdaudput 用管理ファイルが存在する場合は，pdaudput 用管理ファイルを作成しないため，指定を無視します。

×：指定が無効です。

21.4　リターンコード一覧

pdaudput が出力するリターンコードを次の表に示します。

表 21–7　pdaudput が出力するリターンコード

リターンコード	意味
0	正常終了です。
4	正常終了です（JP1/NETM/Audit 用監査ログ出力ファイルのスワップが発生したことによる終了）。
8	異常終了です。

付録

付録 A　コマンド一覧

HiRDB で使用できるコマンドには，運用コマンドとユティリティとがあります。HiRDB で使用できるコマンドの一覧を次の表に示します。

表 A–1　コマンド一覧

種別	コマンド	内容	参照箇所
システムの運用	pdadmvr	HiRDB バージョン情報の取得	pdadmvr
	pdcat	ファイルの内容表示	pdcat
	pdchgconf	システム構成変更コマンド	pdchgconf
	pdclibsync	C ライブラリファイルの操作	pdclibsync
	pdconfchk	システム定義のチェック	pdconfchk
	pdcspool	トラブルシュート情報及び作業用一時ファイルの削除	pdcspool
	pdinfoget	障害情報の取得及び容量見積もり	pdinfoget
	pdjarsync	JAR ファイルの操作	pdjarsync
	pdlistls	リスト定義情報の表示	pdlistls
	pdls	HiRDB システムの状態表示	pdls
	pdntenv	HiRDB の動作環境の設定	pdntenv
	pdobjconv	64 ビットモードの HiRDB への SQL オブジェクトの移行	pdobjconv
	pdsvhostname	サーバのホスト名表示	pdsvhostname
	pdvrup	HiRDB バージョンアップ	pdvrup
HiRDB ファイルシステム	pdfbkup	HiRDB ファイルシステムのバックアップ	pdfbkup
	pdfchfs	HiRDB ファイルシステム領域の管理情報の変更	pdfchfs
	pdffsck	HiRDB ファイルシステム領域の整合性の検証及び修復	pdffsck
	pdfls	HiRDB ファイルシステムの内容表示	pdfls
	pdfmkfs	HiRDB ファイルシステム領域の初期設定	pdfmkfs
	pdfrm	HiRDB ファイルの削除	pdfrm
	pdfrstr	HiRDB ファイルシステムのリストア	pdfrstr
	pdfstatfs	HiRDB ファイルシステム領域の状態表示	pdfstatfs
	pdfzeroinit	HiRDB ファイルシステム領域内の使用済み領域の初期化	pdfzeroinit

種別	コマンド	内容	参照箇所
ログ関係のファイル	pdlogadpf	ログ関係のファイルの割り当て	pdlogadpf
	pdlogatul	自動ログアンロード機能の制御	pdlogatul
	pdlogchg	ログ関係のファイルのステータス変更	pdlogchg
	pdlogcls	ログ関係のファイルのクローズ	pdlogcls
	pdloginit	ログ関係のファイルの初期設定	pdloginit
	pdlogls	ログ関係のファイルの情報表示	pdlogls
	pdlogopen	ログ関係のファイルのオープン	pdlogopen
	pdlogrm	ログ関係のファイルの削除	pdlogrm
	pdlogswap	ログ関係のファイルのスワップ	pdlogswap
	pdlogsync	シンクポイントダンプの取得	pdlogsync
	pdlogucat	アンロードログファイルの情報表示	pdlogucat
	pdlogunld	ログ関係のファイルのアンロード	pdlogunld
ステータスファイル	pdstscls	ステータスファイルのクローズ	pdstscls
	pdstsinit	ステータスファイルの初期設定	pdstsinit
	pdstsopen	ステータスファイルのオープン	pdstsopen
	pdstsrm	ステータスファイルの削除	pdstsrm
	pdstsswap	ステータスファイルのスワップ	pdstsswap
HiRDB の開始・終了	pdstart	HiRDB システム・ユニット・サーバの開始	pdstart
	pdstop	HiRDB システム・ユニット・サーバの終了	pdstop
統計ログ	pdstbegin	統計情報の出力開始	pdstbegin
	pdstend	統計情報の出力停止	pdstend
	pdstjswap	統計ログファイルの切り替え	pdstjswap
	pdstjsync	統計ログファイルへの統計ログバッファの反映	pdstjsync
RD エリア	pdclose	RD エリアのクローズ	pdclose
	pddbls	RD エリアの状態表示	pddbls
	pdhold	RD エリアの閉塞	pdhold
	pdopen	RD エリアのオープン	pdopen
	pdrels	RD エリアの閉塞解除	pdrels
	pddbfrz	ユーザ LOB 用 RD エリアの満杯 HiRDB ファイルの更新凍結	pddbfrz
	pdrdrefls	関連する RD エリアの情報の表示	pdrdrefls

種別	コマンド	内容	参照箇所
グローバルバッファ	pdbufls	グローバルバッファ情報の表示	pdbufls
	pdbufmod	グローバルバッファの動的変更	pdbufmod
インメモリデータ処理	pdmemdb	インメモリデータ処理に関する操作	pdmemdb
トランザクションの制御	pdcmt	トランザクションのコミット	pdcmt
	pdfgt	トランザクションの強制終了	pdfgt
	pdrbk	トランザクションのロールバック	pdrbk
	pdtrndec	未決着トランザクションの強制自動決着	pdtrndec
プロセスの制御	pdcancel	UAP，ユティリティ処理の強制終了	pdcancel
	pdchprc	サーバプロセスの起動本数変更	pdchprc
	pdkill	プロセスの強制停止	pdkill
	pdpfresh	サーバプロセスのリフレッシュ	pdpfresh
修正版 HiRDB の入れ替え	pdprgcopy	修正版 HiRDB のコピー	pdprgcopy
	pdprgrenew	修正版 HiRDB の入れ替え	pdprgrenew
HiRDB Datareplicator 連携	pdrplstart	HiRDB Datareplicator 連携の開始	pdrplstart
	pdrplstop	HiRDB Datareplicator 連携の終了	pdrplstop
セキュリティ監査	pdaudbegin	監査証跡の取得開始	pdaudbegin
	pdaudend	監査証跡の取得停止	pdaudend
	pdaudatld	監査証跡表の自動データロード	pdaudatld
	pdaudrm	閉塞中の監査証跡ファイルの削除	pdaudrm
	pdaudswap	現用の監査証跡ファイルのスワップ	pdaudswap
	pdaudput	監査証跡表のデータを JP1/NETM/Audit 用監査ログ出力ファイルに出力	「JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput）」
CONNECT 関連セキュリティ機能	pdacunlck	連続認証失敗アカウントロック状態の解除	pdacunlck
トランザクションキューイング機能	pdtrnqing	トランザクションキューイング機能の開始，解除	pdtrnqing
SQL トレースの取得	pdclttrc	SQL トレースの動的取得	pdclttrc
PRF トレース情報	pdprfed	PRF トレース情報の編集出力	pdprfed
	pdprflevel	PRF トレース取得レベルの表示，変更	pdprflevel
SQL オブジェクトの情報表示	pdobils	SQL オブジェクト用バッファの統計情報表示	pdobils
SQL の仕様関連	pdextfunc	拡張システム定義スカラ関数の定義及び削除	pdextfunc

種別	コマンド	内容	参照箇所
Windows ファイアウォールの例外登録	pddelfw.bat	Windows ファイアウォールの例外リスト削除コマンド	pddelfw.bat
	pdsetfw.bat	Windows ファイアウォールの例外リスト登録コマンド	pdsetfw.bat
ファイルセキュリティ強化機能	pdsetacl	ファイルセキュリティ強化機能の設定	pdsetacl
SQL の翻訳	pdcbl	COBOL プリプロセサ	–※1
	pdcpp	C プリプロセサ	–※1
	pdocb	OOCOBOL プリプロセサ	–※1
	pdocc	C++プリプロセサ	–※1
データベースの生成	pdinit	データベース初期設定ユティリティ	「データベース初期設定ユティリティ（pdinit）」
	pddef	データベース定義ユティリティ	「データベース定義ユティリティ（pddef）」
	pdload	データベース作成ユティリティ	「データベース作成ユティリティ（pdload）」
	pdparaload	パラレルローディングの実行	pdparaload
	–	会話型 SQL 実行ユティリティ※2	「会話型 SQL 実行ユティリティ（pdsql）」
	pddefrev	定義系 SQL の生成	pddefrev
データベースの運用	pdmod	データベース構成変更ユティリティ	「データベース構成変更ユティリティ（pdmod）」
	pdrorg	データベース再編成ユティリティ	「データベース再編成ユティリティ（pdrorg）」
	pdexp	ディクショナリ搬出入ユティリティ	「ディクショナリ搬出入ユティリティ（pdexp）」

種別	コマンド	内容	参照箇所
	pdrbal	リバランスユティリティ	「リバランスユティリティ（pdrbal）」
	pdreclaim	空きページ解放ユティリティ	「空きページ解放ユティリティ（pdreclaim）」
	pdpgbfon	グローバルバッファ常駐化ユティリティ	「グローバルバッファ常駐化ユティリティ（pdpgbfon）」
	pdconstck	整合性チェックユティリティ	「整合性チェックユティリティ（pdconstck）」
	pdchpathf	ファイルパス変更オフラインコマンド	pdchpathf
	pdchpathn	ファイルパス変更オンラインコマンド	pdchpathn
チューニング	pdstedit	統計解析ユティリティ	「統計解析ユティリティ（pdstedit）」
	pddbst	データベース状態解析ユティリティ	「データベース状態解析ユティリティ（pddbst）」
	pdgetcst	最適化情報収集ユティリティ	「最適化情報収集ユティリティ（pdgetcst）」
	pdvwopt	アクセスパス表示ユティリティ	「アクセスパス表示ユティリティ（pdvwopt）」
データベースの障害対策	pdcopy	データベース複写ユティリティ	「データベース複写ユティリティ（pdcopy）」
	pdbkupls	バックアップファイルの情報表示	pdbkupls

種別	コマンド	内容	参照箇所
	pdrstr	データベース回復ユティリティ	「データベース回復ユティリティ（pdrstr）」
	pdmstchk	マスタディレクトリ用 RD エリアの整合性チェック	「マスタディレクトリ用 RD エリアの整合性チェック（pdmstchk）」
プラグイン関連	pdplgrgst	プラグインの登録・削除	pdplgrgst
	pdreginit	レジストリ機能初期設定ユティリティ	「レジストリ機能初期設定ユティリティ（pdreginit）」

（凡例）

－：該当しません。

注※1

pdcbl，pdcpp，pdocb，及び pdocc については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

注※2

会話型 SQL 実行ユティリティを実行する場合は，HiRDB SQL Executer が必要です。

付録 B　コマンド実行時の排他制御モード

HiRDB のコマンドの中には，各排他資源に対して排他制御モードを使い分け，自動的に排他を掛けるものがあります。このため，UAP を含めプログラムを同時実行する場合は，該当するコマンドの排他制御モードの表と UAP（SQL）の排他制御の表を参考にして，排他制御モードの組み合わせを確認する必要があります。

SQL 文の種類及び実行環境の違いによる排他制御モードの組み合わせ例については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

表中で使用している排他制御モードを次に示します。

PR：共用モードの排他が掛かります。

EX：排他モードの排他が掛かります。

SR：意図共用モードの排他が掛かります。

SU：意図排他モードの排他が掛かります。

PU：共用意図排他モードの排他が掛かります。

↓：検査保留状態取得時の排他制御モードが引き継がれます。

－：排他は掛かりません。

付録 B.1　運用コマンドの排他制御モード

運用コマンドの排他制御モードを次の表に示します。

表 B–1　運用コマンドの排他制御モード

実行環境		資源					
コマンド	オプション	ディクショナリ表	RD エリア状態	RD エリア	バックアップ閉塞	pdhold コマンド受付状態	インメモリ同期
pdbufls	すべて	－	－	－	－	－	－
pdclose	なし	－	EX	EX	－	－	EX※3
pddbfrz	すべて	－	EX	EX	－	－	－
pddbls	-a	－	－	SR	－	－	－
	なし，-l，又は-b	－	－	－	－	－	－
pdhold	なし	－	EX	EX	－	EX	－

実行環境		資源					
コマンド	オプション	ディクショナリ表	RD エリア状態	RD エリア	バックアップ閉塞	pdhold コマンド受付状態	インメモリ同期
	-c	−	EX	EX	−	EX	EX
	-i	−	EX	PR	−	EX	−
	-b	−	EX	PR	−	EX	EX
	-bw	−	EX	PR	EX※2	−	EX
	-buw	−	EX	PR	EX※2	−	−
	-bu	−	EX	−	EX※2	−	−
pdopen	なし	−	EX	EX	−	−	−
pdrels	なし，又は-o	−	EX	PR/−※1	−	−	−
pdchpathn	すべて	EX※4	−	SU※5	−	−	−
pdrdrefls	すべて	−	−	SR※6	−	−	−

注※1

更新可能バックアップ閉塞時には，排他を掛けません。

注※2

pdrels コマンドで閉塞解除するか，又は障害閉塞になるまで，排他は解除されません。

注※3

コマンド実行前の RD エリアの状態が障害閉塞状態又はクローズ状態以外のときに排他を取得します。

注※4

さらに，ディクショナリ表（SQL_PHYSICAL_FILES，SQL_IOS_GENERATIONS）に表排他を取得します。

注※5

マスタディレクトリ用 RD エリア，データディクショナリ表用 RD エリアが対象となります。

注※6

データディクショナリ用 RD エリアが対象となります。

付録 B.2　ユティリティの排他制御モード

各ユティリティの排他制御モードを次の表に示します。

- 表「データベース作成ユティリティの排他制御モード」

- 表「データベース作成ユティリティの排他制御モード（共用表の場合）」
- 表「データベース構成変更ユティリティの排他制御モード」
- 表「データベース再編成ユティリティの排他制御モード」
- 表「データベース再編成ユティリティの排他制御モード（共用表の場合）」
- 表「リバランスユティリティの排他制御モード」
- 表「空きページ解放ユティリティの排他制御モード」
- 表「空きページ解放ユティリティの排他制御モード（共用表の場合）」
- 表「グローバルバッファ常駐化ユティリティの排他制御モード」
- 表「グローバルバッファ常駐化ユティリティの排他制御モード（共用表の場合）」
- 表「データベース状態解析ユティリティの排他制御モード」
- 表「最適化情報収集ユティリティの排他制御モード」
- 表「データベース複写ユティリティの排他制御モード」
- 表「データベース回復ユティリティの排他制御モード」
- 表「レジストリ機能初期設定ユティリティの排他制御モード」
- 表「整合性チェックユティリティの排他制御モード」
- 表「JP1/NETM/Audit 用監査ログ出力ユティリティ」

なお，次のユティリティについてはユティリティ自身で排他制御をしないため，記述を省略します。また，データベース定義ユティリティについては，定義系 SQL の場合と同じであるため，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

- データベース初期設定ユティリティ（pdinit）
- ディクショナリ搬出入ユティリティ（pdexp）
- 統計解析ユティリティ（pdstedit）
- アクセスパス表示ユティリティ（pdvwopt）

表 B–2　データベース作成ユティリティの排他制御モード

実行環境	資源									
	表		インデクス		表（NOW AIT 検索中）	資源管理テーブル	ユーザ LOB 用 RD エリア	順序数生成子		
	RD エリア	表 ID	RD エリア	インデクス ID				順序数生成子 ID		RD エリア
								全数一括取得方式	全数一括取得方式以外	
表単位のデータロード	SU	EX	SU	EX	EX※2	SR	SU	EX	SU	SU
					PU※2					

実行環境	資源									
	表		インデクス		表（NOWAIT 検索中）	資源管理テーブル	ユーザ LOB 用 RD エリア	順序数生成子		
	RD エリア	表 ID	RD エリア	インデクス ID				順序数生成子 ID		RD エリア
								全数一括取得方式	全数一括取得方式以外	
RD エリア単位のデータロード※3	EX	SR	SU	EX※1	−	SR	EX	EX	SU	SU
LOB 列だけのデータロード	SR	PR	−	−	−	SR	EX	−	−	−

注

LOB 列構成基表だけのデータロードの場合，表単位にデータロードするときは表単位のデータロード，RD エリア単位にデータロードするときは RD エリア単位のデータロードと同じになります。

注※1

RD エリア単位の場合，非横分割インデクスに対して排他は掛かりません。横分割インデクスに対して排他は掛かりますが，RD エリアごとに排他が掛かるため，RD エリア単位のデータロードは同時に実行できます。

注※2

データを削除するときは EX モードで排他が掛かります。そのほかの場合，option 文が nowait=no のときは EX モード，nowait=yes のときは PU モードで排他が掛かります。

注※3

分割表の場合，データロード対象でない RD エリアに対して排他は掛かりません。

表 B–3　データベース作成ユティリティの排他制御モード（共用表の場合）

実行環境	資源											
	表				インデクス			表（NOWAIT 検索中）	資源管理テーブル	順序数生成子		
	更新可能バックエンドサーバ		参照専用バックエンドサーバ		更新可能バックエンドサーバの共用 RD エリア	参照専用バックエンドサーバの共用 RD エリア	インデクス ID			順序数生成子 ID		RD エリア
	共用 RD エリア	表 ID	共用 RD エリア	表 ID						全数一括取得方式	全数一括取得方式以外	
データロード	EX	EX	EX	EX	EX	EX	EX	EX	SR	EX	SU	SU

表 B–4　データベース構成変更ユティリティの排他制御モード

実行環境	資源									
	データベース構成変更ユティリティ	ディクショナリ表	RD エリア状態	RD エリア	表	行（ディクショナリ）※6	資源管理テーブル	一時表実体化管理	一時表用 RD エリア管理情報	一時表用 RD エリア情報
RD エリアの追加	EX	EX	EX※3	EX※3	–	EX	–	EX※4	EX※4	EX※4
RD エリアの拡張	EX	EX	EX※3	PU(EX)※3	–	EX	EX※3	–	–	–
RD エリアの再初期化	EX	EX	EX※3	EX※3	EX※5	EX	–	EX※4	EX※4	EX※4
RD エリアの削除	EX	EX	EX※3	EX※3	–	EX	–	EX※4	EX※4	EX※4
ディクショナリ表の属性定義変更	EX	EX	–	EX※2	–	EX	–	–	–	–
RD エリアの属性変更	EX	EX	EX	EX	–	EX	–	–	–	–
HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更	EX	EX	EX	EX	–	EX	–	–	–	–
バックエンドサーバへの構成変更※1	EX	EX	EX	EX	–	EX	–	–	–	–
RD エリアの移動	EX	EX	EX※3	EX※3	–	EX	–	–	–	–
監査人の登録	EX	EX	–	–	–	EX	–	–	–	–
監査証跡表の作成	EX	EX	–	EX	–	EX	EX	–	–	–

注※1

HiRDB/シングルサーバの場合は該当しません。HiRDB/パラレルサーバの場合だけ該当します。

注※2

データディクショナリ用 RD エリアだけ該当します。

注※3

共用 RD エリアの場合，更新可能バックエンドサーバ及び参照専用バックエンドサーバの両方が該当します。

注※4

一時表用 RD エリアの場合だけ該当します。

注※5

without lock table オペランド指定時は該当しません。

注※6

行排他を取得するディクショナリ表を次の表に示します。

表 B–5　データベース構成変更ユティリティで行排他を取得するディクショナリ表（1/2）

実行環境		ディクショナリ表													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
RD エリアの追加		–	–	–	–	–	–	–	–	–	–	–	○	○	○
RD エリアの拡張		–	–	–	–	–	–	–	–	–	–	–	○	–	○
RD エリアの再初期化	HiRDB ファイル構成変更あり	–	–	–	–	–	–	–	–	–	–	–	○	–	○
	被参照表を格納し，他 RD エリアに参照関係にある参照表がある	○	–	–	–	–	–	–	–	–	–	–	–	–	–
	再編成時期予測機能適用あり	–	–	○	–	–	–	–	–	–	–	–	–	–	–
RD エリアの削除	削除対象 RD エリアが解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリア	–	○	–	–	–	–	–	–	○	○	○	○	○	–
	再編成時期予測機能適用あり	–	–	○	○	–	–	–	–	–	–	○	○	○	–
	上記以外	–	–	–	–	–	–	–	–	–	–	○	○	○	–
ディクショ	ディクショナリ表の表参照	–	○	–	–	–	–	–	–	–	–	–	–	–	–

実行環境			ディクショナリ表												
			1	2	3	4	5	6	7	8	9	10	11	12	13	14
ナリ表の属性定義変更	ナリ表の参照権限の変更	ルーチン定義あり														
		ディクショナリ表の表参照トリガ定義あり	−	○	−	−	−	−	−	−	−	−	−	−	−	−
		上記以外	−	○	−	−	−	−	−	−	−	−	−	−	−	−
	ディクショナリ表の混在文字データの使用		−	○	−	−	−	−	−	−	−	−	−	−	−	−
RD エリアの属性変更		extension オペランド指定あり	−	−	−	−	−	−	−	−	−	−	−	○	−	−
		rename オペランド指定あり	−	−	−	−	○	○	○	○	−	○	−	−	○	○
HiRDB/シングルサーバ構成からHiRDB/パラレルサーバ構成へのRD エリアの変更		分割表あり	−	−	−	−	−	−	○	○	−	−	−	○	−	○
		ルーチン定義あり	−	−	−	−	−	−	−	−	−	−	−	○	−	○
		上記以外	−	−	−	−	−	−	−	−	−	−	−	○	−	○
バックエンドサーバへの構成変更※1		分割表あり	−	−	−	−	−	−	○	○	−	−	−	○	−	○
		ルーチン定義あり	−	−	−	−	−	−	−	−	−	−	−	○	−	○
		上記以外	−	−	−	−	−	−	−	−	−	−	−	○	−	○
RD エリアの移動		分割表あり	−	−	−	−	−	−	○	○	−	−	−	○	−	○
		分割表にインデクス定義あり	−	−	−	−	−	−	−	−	−	○	−	○	−	○
		格納表参照ルーチン定義あり	−	−	−	−	−	−	−	−	−	−	−	○	−	○

実行環境		ディクショナリ表													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
	格納表表参照トリガ定義あり	−	−	−	−	−	−	−	−	−	−	−	○	−	○
	norestart オペランド指定あり	−	−	−	−	−	−	−	−	−	−	−	○	−	○
	上記以外	−	−	−	−	−	−	−	−	−	−	−	○	−	○
監査人の登録		−	−	−	−	−	−	−	−	−	−	−	−	−	−
監査証跡表の作成		−	○	−	−	−	−	−	−	−	−	−	−	−	−

（凡例）

○：取得します。

−：取得しません。

1：SQL_CHECKS

2：SQL_COLUMNS

3：SQL_DB_MANAGEMENT

4：SQL_DB_STATE_ANALYZED

5：SQL_DIV_COLUMN

6：SQL_DIV_INDEX

7：SQL_DIV_TABLE

8：SQL_DIV_TABLE_REGULARIZE

9：SQL_INDEX_COLINF

10：SQL_INDEXES

11：SQL_IOS_GENERATIONS

12：SQL_PHYSICAL_FILES

13：SQL_RDAREA_PRIVILEGES

14：SQL_RDAREAS

注※1

HiRDB/シングルサーバの場合は該当しません。HiRDB/パラレルサーバの場合だけ該当します。

表 B–6　データベース構成変更ユティリティで行排他を取得するディクショナリ表（2/2）

実行環境	ディクショナリ表													
	15	16	17	18	19	20	21	22	23	24	25	26	27	28
RD エリアの追加	−	−	−	−	−	−	−	−	−	−	−	−	−	−
RD エリアの拡張	−	−	−	−	−	−	−	−	−	−	−	−	−	−

実行環境			ディクショナリ表												
			15	16	17	18	19	20	21	22	23	24	25	26	27	28
RD エリアの再初期化		HiRDB ファイル構成変更あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
		被参照表を格納し，他 RD エリアに参照関係にある参照表がある	○	−	−	−	−	−	−	−	−	−	−	−	−	−
		再編成時期予測機能適用あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
RD エリアの削除		削除対象 RD エリアが解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリア	−	−	−	−	−	○	○	−	−	−	−	−	−	−
		再編成時期予測機能適用あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
		上記以外	−	−	−	−	−	−	−	−	−	−	−	−	−	−
ディクショナリ表の属性定義変更	ディクショナリ表の参照権限の変更	ディクショナリ表の表参照ルーチン定義あり	−	○	○	○	−	−	○	−	−	−	−	○	○	○
		ディクショナリ表の表参照トリガ定義あり	−	−	−	−	−	−	○	○	○	○	−	○	○	○
		上記以外	−	−	−	−	−	−	○	−	−	−	−	○	○	○
	ディクショナリ表の混在文字データの使用		−	−	−	−	−	−	○	−	−	−	−	○	○	○
RD エリアの属性変更		extension オペランド指定あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−

実行環境		ディクショナリ表													
		15	16	17	18	19	20	21	22	23	24	25	26	27	28
	rename オペランド指定あり	−	−	−	−	○	−	○	−	−	−	−	−	−	−
HiRDB/シングルサーバ構成からHiRDB/パラレルサーバ構成へのRD エリアの変更	分割表あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
	ルーチン定義あり	−	−	−	○	−	−	−	−	−	−	−	−	−	−
	上記以外	−	−	−	−	−	−	−	−	−	−	−	−	−	−
バックエンドサーバへの構成変更※1	分割表あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
	ルーチン定義あり	−	−	−	○	−	−	−	−	−	−	−	−	−	−
	上記以外	−	−	−	−	−	−	−	−	−	−	−	−	−	−
RD エリアの移動	分割表あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
	分割表にインデクス定義あり	−	−	−	−	−	−	−	−	−	−	−	−	−	−
	格納表参照ルーチン定義あり	−	○	○	○	−	−	−	−	−	−	−	−	−	−
	格納表表参照トリガ定義あり	−	−	−	−	−	−	−	○	○	○	−	−	−	−
	norestart オペランド指定あり	−	−	−	−	−	−	○	−	−	−	−	−	−	−
	上記以外	−	−	−	−	−	−	−	−	−	−	−	−	−	−
監査人の登録		−	−	−	−	−	○	−	−	−	−	○	−	−	−
監査証跡表の作成		−	−	−	−	−	○	○	−	−	−	−	−	−	−

（凡例）

○：取得します。

−：取得しません。

15：SQL_REFERENTIAL_CONSTRAINTS

16：SQL_ROUTINE_PARAMS

17：SQL_ROUTINE_RESOURCES

18：SQL_ROUTINES

19：SQL_SEQUENCES

20：SQL_TABLE_PRIVILEGES

21：SQL_TABLES

22：SQL_TRIGGER_COLUMNS

23：SQL_TRIGGER_USAGE

24：SQL_TRIGGERS

25：SQL_USERS

26：SQL_VIEW_DEF

27：SQL_VIEW_TABLE_USAGE

28：SQL_VIEWS

注※1

HiRDB/シングルサーバの場合は該当しません。HiRDB/パラレルサーバの場合だけ該当します。

表 B–7　データベース再編成ユティリティの排他制御モード

実行環境		資源							
		表		インデクス		表（NOWAIT 検索中）	資源管理テーブル	インデクス情報ファイル	ユーザ LOB 用 RD エリア※1
		RD エリア※1	表 ID	RD エリア※1	インデクス ID				
表単位のアンロード（-k unld）	option 文の指定が unldenq=tblenq	SR	PR	SR※2	−	−	SR	−	SR
	option 文の指定が unldenq=rdenq	PR	SR	PR※2	−	−	SR	−	PR
	option 文の指定が unldenq=nowait	SR	−	SR※2	−	SR	SR	−	SR
RD エリア単位のアンロード（-k unld）	option 文の指定が unldenq=tblenq	SR	PR	SR※2	−	−	SR	−	SR
	option 文の指定が unldenq=rdenq	PR	SR	PR※2	−	−	SR	−	PR
	option 文の指定が unldenq=nowait	SR	−	SR※2	−	SR	SR	−	SR
表単位のリロード（-k reld）		SU	EX	SU	EX	EX	SR	−	SU
RD エリア単位のリロード（-k reld）		EX	SR	SU	EX※3	−	SR	−	EX
表単位の再編成（-k rorg）		SU	EX	SU	EX	EX	SR	−	SU
RD エリア単位の再編成		EX	SR※7	SU	EX※3	−	SR	−	EX

実行環境		資源							
		表		インデクス		表（NOAIT 検索中）	資源管理テーブル	インデクス情報ファイル	ユーザ LOB 用 RD エリア※1
		RD エリア※1	表 ID	RD エリア※1	インデクス ID				
（-k rorg）									
インデクスの一括作成（-k ixmk）		SR	SR	SU	EX	−	SR	−	−
インデクスの再作成（-k ixrc）	index 文指定時※4	EX→SR※5	−→SR※5	SU	EX	−	SR	−	EX→−※5
		SU→SR※5	EX→SR※5			EX→−※5			SU→−※5
	idxname 文指定時	SU	EX	SU	EX	EX	SR	−	−
インデクスの再編成（-k ixor）※6		SR	PR	SU	EX	−	SR	−	−
				EX	−				
プラグインインデクスの一括作成（-k ixmk）		SR	SR	SU	EX	−	SR	EX	−
プラグインインデクスの再作成（-k ixrc）	index 文指定時※4	EX→SR※5	−→SR※5	SU	EX	−	SR	EX	EX→−※5
		SU→SR※5	EX→SR※5			EX→−※5			SU→−※5
	idxname 文指定時	SU	EX	SU	EX	EX	SR	−	SU
LOB データだけのリロード（-k reld）		SR	PR	−	−	−	SR	−	EX
LOB データだけの再編成（-k rorg）		SU	EX	SR※2	−	EX	SR	−	SU
ディクショナリ表のメンテナンス（-k maintenance）		SU	EX	SU	−	−	−	−	−

注

LOB 列構成基表だけのリロード又は再編成の場合，表単位にリロード又は再編成するときは表単位のリロード又は再編成，RD エリア単位にリロード又は再編成するときは RD エリア単位のリロード又は再編成と同じになります。

注※1

処理対象でない RD エリアには，排他は掛かりません。

注※2

物理順のアンロードの場合，排他は掛かりません。

注※3

非横分割インデクスに対して，リロード中は排他が掛かりません。横分割インデクスに対して排他は掛かりますが，RD エリアごとに排他が掛かるため，RD エリア単位のリロード又は再編成は同時に実行できます。

注※4

上段が横分割インデクスの場合，下段が非横分割インデクスの場合を示しています。

注※5

インデクス情報のアンロード時（KFPL00725-I から KFPL00726-I までの間）の排他モードを，インデクス作成時（KFPL00715-I から KFPL00716-I までの間）には変更します。

注※6

上段がインデクス格納 RD エリアをコマンド閉塞している場合，下段がインデクス格納 RD エリアをコマンド閉塞していない場合を示しています。

注※7

リロード中は排他が掛かりません。

表 B–8　データベース再編成ユティリティの排他制御モード（共用表の場合）

実行環境	資源									
	表				インデクス			表（NOW AIT 検索中）	資源管理テーブル	インデクス情報ファイル
	更新可能バックエンドサーバ		参照専用バックエンドサーバ		更新可能バックエンドサーバの共用 RD エリア	参照専用バックエンドサーバの共用 RD エリア	インデクス ID			
	共用 RD エリア	表 ID	共用 RD エリア	表 ID						
表のアンロード（-k unld）	SR	PR	−	−	SR※1	−	−	−	SR	−
表のリロード（-k reld）	EX	EX	EX	EX	EX	EX	EX	EX	SR	−
表の再編成（-k rorg）	EX	EX	EX	EX	EX	EX	EX	EX	SR	−
インデクスの一括作成（-k ixmk）	SR	SR	−	−	EX	EX	EX	−	SR	−
インデクスの再作成（-k ixrc，及び index 文指定時）※2	EX→SR	EX→SR	EX	EX	EX	EX	EX	EX→−	SR	−
インデクスの再作成（-k ixrc，及び idxname 文指定時）	EX	EX	EX	EX	EX	EX	EX	EX	SR	−

実行環境	資源									
	表				インデクス			表（NOW AIT 検索中）	資源管理テーブル	インデクス情報ファイル
	更新可能バックエンドサーバ		参照専用バックエンドサーバ		更新可能バックエンドサーバの共用 RD エリア	参照専用バックエンドサーバの共用 RD エリア	インデクス ID			
	共用 RD エリア	表 ID	共用 RD エリア	表 ID						
インデクスの再編成（-k ixor）	SR	PR	−	−	EX	EX	EX	−	SR	−

注※1

物理順のアンロードの場合，排他は掛かりません。

注※2

インデクス情報のアンロード時（KFPL00725-I から KFPL00726-I までの間）の排他モードを，インデクス作成時（KFPL00715-I から KFPL00716-I までの間）には変更します。

表 B–9　リバランスユティリティの排他制御モード

実行環境	資源					
	RD エリア	表	表（NOWAIT 検索中）	インデクス	リバランス※	前処理表
共有モード	SU	EX	EX	EX	EX	SR
占有モード	SU	EX	EX	EX	−	SR

注※

リバランスの資源は，シングルサーバ又はフロントエンドサーバで排他が掛かります。

表 B–10　空きページ解放ユティリティの排他制御モード

実行環境	資源										
	表					表（NOW AIT 検索中）	インデクス				
	RD エリア	資源管理テーブル	表 ID	pdreclaim	トランザクション決着待ち		RD エリア	資源管理テーブル	インデクス ID	pdreclaim	トランザクション決着待ち
表の空きページ解放	SU	SR	−	EX	−	SR	−	−	−	−	−
表の空きセグメント解放（-j）	EX	SR	−	EX	−	−	−	−	−	−	−
表の空きページ解放，表の	SU	SR	SR	EX	PR	SR	−	−	−	−	−

実行環境	資源										
	表					表（NOAIT 検索中）	インデクス				
	RD エリア	資源管理テーブル	表 ID	pdreclaim	トランザクション決着待ち		RD エリア	資源管理テーブル	インデクス ID	pdreclaim	トランザクション決着待ち
空きセグメント解放（-a）											
インデクスの空きページ解放	−	−	−	−	−	SR	SU	SR	EX	EX	PR
インデクスの空きセグメント解放（-j）	−	−	SR	−	−	−	EX	SR	−	EX	−
インデクスの空きページ解放，インデクスの空きセグメント解放（-a）	−	−	SR	−	−	SR	SU	SR	EX	EX	PR

表 B–11　空きページ解放ユティリティの排他制御モード（共用表の場合）

実行環境	資源												
	表							インデクス					
	更新可能バックエンドサーバ		参照専用バックエンドサーバ		資源管理テーブル	pdreclaim	トランザクション決着待ち	更新可能バックエンドサーバの共用 RD エリア	参照専用バックエンドサーバの共用 RD エリア	資源管理テーブル	インデクス ID	pdreclaim	トランザクション決着待ち
	共用 RD エリア	表 ID	共用 RD エリア	表 ID									
表の空きページ解放	EX	EX	EX	EX	SR	EX	−	−	−	−	−	−	−
表の空きセグメント解放（-j）	EX	EX	EX	EX	SR	EX	−	−	−	−	−	−	−
表の空きページ解放，表の空きセグメント解放（-a）	EX	EX	EX	EX	SR	EX	PR	−	−	−	−	−	−
インデクスの空きページ解放	−	SR	−	−	−	−	−	EX	EX	SR	EX	EX	PR

実行環境	資源												
	表							インデクス					
	更新可能バックエンドサーバ		参照専用バックエンドサーバ		資源管理テーブル	pdreclaim	トランザクション決着待ち	更新可能バックエンドサーバの共用RDエリア	参照専用バックエンドサーバの共用RDエリア	資源管理テーブル	インデクスID	pdreclaim	トランザクション決着待ち
	共用RDエリア	表ID	共用RDエリア	表ID									
インデクスの空きセグメント解放（-j）	−	SR	−	−	−	−	−	EX	EX	SR	−	EX	−
インデクスの空きページ解放，インデクスの空きセグメント解放（-a）	−	SR	−	−	−	−	−	EX	EX	SR	EX	EX	PR

表 B–12　グローバルバッファ常駐化ユティリティの排他制御モード

実行環境	資源					
	表			インデクス		
	RDエリア	資源管理テーブル	表（NOWAIT検索中）	RDエリア	資源管理テーブル	インデクスID
データページの読み込み (-k table)	SR	SR	SR	−	−	−
インデクスページの読み込み (-k index)	−	−	SR	SR	SR	SR

表 B–13　グローバルバッファ常駐化ユティリティの排他制御モード（共用表の場合）

実行環境	資源											
	表						インデクス					
	更新可能バックエンドサーバ			参照専用バックエンドサーバ			更新可能バックエンドサーバ			参照専用バックエンドサーバ		
	RD エリア	資源管理テーブル	表（NO WAIT 検索中）	RD エリア	資源管理テーブル	表（NO WAIT 検索中）	RD エリア	資源管理テーブル	インデクス ID	RD エリア	資源管理テーブル	インデクス ID
データページの読み込み (-k table)	SR	SR	SR	SR	SR	SR	–	–	–	–	–	–
インデクスページの読み込み (-k index)	–	–	SR	–	–	SR	SR	SR	SR	SR	SR	SR

表 B–14　データベース状態解析ユティリティの排他制御モード

実行環境		資源						
		RD エリア			資源管理テーブル	表	表（NOWAIT 検索中）※1	HiRDB ファイル
		表用	インデクス用	RD エリア指定				
RD エリア単位の解析	論理的解析	–	–	SR	SR	–	SR	–
	物理的解析	–	–	SR	SR	–	–	SR
表単位の解析		SR	–	–	SR	SR	SR	–
インデクス単位の解析		–	SR	–	SR	SR	SR	–
状態解析結果蓄積機能		SR	SR	SR	SR	SR	–	–
再編成時期予測機能※2		SR	SR	SR	SR	SR	SR	–

注※1

-d オプションを指定した場合にだけ表（NOWAIT 検索中）に排他が掛かります。

注※2

-m オプションを指定した場合の排他です。

表 B–15　最適化情報収集ユティリティの排他制御モード

実行環境	資源							
	RD エリア		資源管理テーブル		表	表（NOW AIT 検索中）	ディクショナリ表	
	表用	インデクス用	表用	インデクス用			行	キー
最適化情報の取得	SR	SR	SR	SR	SR/SU※1	SR	PR※1/EX※2	PR※1/EX※2
最適化情報の削除	–	–	–	–	SU※1	–	PR※1/EX※2	PR※1/EX※2
最適化情報の登録	–	–	–	–	SU※1	–	PR※1/EX※2	PR※1/EX※2
最適化情報の退避	–	–	–	–	SU※1	–	PR※1	PR※1

注※1

ディクショナリ表検索時に，一時的に排他が掛かります。

注※2

ディクショナリ表更新時に排他が掛かります。

表 B–16　データベース複写ユティリティの排他制御モード

実行環境			資源				
			ディクショナリ表	RD エリア	ディレクトリブロック 6※1	資源管理テーブル	インメモリ同期
-M オプション	x		PR	PR	–	–	–
	r	下記以外の場合	PR	PR	–	–	–
		インメモリ RD エリアの場合	PR	–	–	–	PR
	s		–	–	PR	SR※2	–

注※1

ディクショナリセグメント情報又はユーザディレクトリセグメント情報のことです。

注※2

RD エリアの自動増分指定を行っている場合に排他を取得します。

表 B–17　データベース回復ユティリティの排他制御モード

実行環境	資源	
	ディクショナリ表	RD エリア
RD エリアの回復	EX※	EX

注※

マスタディレクトリ用 RD エリアが回復対象の場合に排他が掛かります。

表 B–18　レジストリ機能初期設定ユティリティの排他制御モード

実行環境	資源			
	データベース構成変更ユティリティ	ディクショナリ表	RD エリア状態	RD エリア
レジストリ用 RD エリア， レジストリ LOB 用 RD エリア， 及びレジストリ管理表の追加 (-k all)	EX	EX	EX	EX

表 B–19　整合性チェックユティリティの排他制御モード

実行環境		資源								
		ユーザ表				ディクショナリ表				
		RD エリア		表	表（NOWAIT 検索中）	行	前処理表	表	RD エリア	
		表用	インデクス用						表用	インデクス用
整合性チェック（-k check）	ディクショナリ表検索時	–	–	–	–	PR	PR	SR	SR	SR
	RD エリアの検査保留状態取得時※	SR	SR	PR	–	–	–	–	–	–
	整合性チェック時	↓	↓	↓	–	–	–	–	–	–
	検査保留状態設定時	SU	SU	EX	EX	EX	PR	SU	SU	SU
検査保留状態の強制設定(-k set)，又は検査保留状態の強制解除（-k release）	ディクショナリ表検索時	–	–	–	–	PR	PR	SR	SR	SR
	RD エリアの検査保留状態取得時※	SR	SR	PR	–	–	–	–	–	–

実行環境		資源								
		ユーザ表				ディクショナリ表				
		RD エリア		表	表（NOWAIT 検索中）	行	前処理表	表	RD エリア	
		表用	インデクス用						表用	インデクス用
	検査保留状態の設定時又は解除時	SU	↓	EX	EX	EX	PR	SU	SU	SU

注※

UNTIL DISCONNECT で排他を掛けます（検査保留状態設定が完了し，DISCONNECT するまで排他を保持します）。

表 B–20　JP1/NETM/Audit 用監査ログ出力ユティリティ

実行環境		資源								
		監査証跡表(ユーザ表)					ディクショナリ表			
		RD エリア			表	行	行	前処理表	表	RD エリア	
		表用	インデクス用	BLOB						表用	インデクス用
-k i	ディクショナリ表検索時	–	–	–	–	–	PR※1	PR※1	SR※1	SR※1	SR※1
-k m	ディクショナリ表検索時	–	–	–	–	–	PR※1	PR※1	SR※1	SR※1	SR※1
-k r	ディクショナリ表検索時	–	–	–	–	–	PR※1	PR※1	SR※1	SR※1	SR※1
	監査証跡表検索時	SR	SR※2	–	SR	–	–	–	–	–	–

注※1

ディクショナリ表（SQL_USERS,SQL_TABLES）検索時に一時的に排他を掛けます。

注※2

監査証跡表に対してインデクスを定義している場合，排他を掛けます。

付録 C　コマンド実行時の RD エリアの状態

付録 C.1　RD エリアの状態遷移

運用コマンド実行時の RD エリアの状態遷移を次の表に示します。

表 C–1　RD エリアの状態遷移（1/2）

RD エリアの状態（遷移前）	RD エリアの状態（遷移後）					
	オープン	閉塞	クローズ	閉塞，かつクローズ	参照可能閉塞	参照可能閉塞，かつクローズ
オープン※1	−	pdhold	トランザクション終了※3	pdhold -c	pdhold -i	×
閉塞	pdrels 又は pdrels -o	−	×	pdclose	pdhold -i	×
クローズ	pdopen，pdrels -o，又はトランザクション開始※2	×	−	pdhold 又は pdhold -c	×	pdhold -i
閉塞，かつクローズ	pdrels -o	pdopen	pdrels	−	×	pdhold -i
参照可能閉塞	pdrels 又は pdrels -o	×	×	×	−	pdclose 又はトランザクション終了※3
参照可能閉塞，かつクローズ	×	×	pdrels	pdhold 又は pdhold -c	pdopen 又はトランザクション開始※2	−
参照可能バックアップ閉塞	pdrels 又は pdrels -o	×	×	×	×	×
参照可能バックアップ閉塞，かつクローズ	pdrels -o	×	pdrels	×	×	×
更新可能バックアップ閉塞	pdrels 又は pdrels -o	×	×	×	×	×
更新可能バックアップ閉塞，かつクローズ	pdrels -o	×	pdrels	×	×	×

（凡例）

－：該当しません。

×：遷移後の RD エリアの状態に遷移できません。

注※1

pdclose コマンドを実行すると，閉塞されていないためエラーとなります。

注※2

RD エリアのオープン契機が DEFER 又は SCHEDULE の場合に状態が遷移します。

注※3

RD エリアのオープン契機が SCHEDULE の場合に状態が遷移します。

表 C–2　RD エリアの状態遷移（2/2）

RD エリアの状態（遷移前）	RD エリアの状態（遷移後）			
	参照可能バックアップ閉塞	参照可能バックアップ閉塞，かつクローズ	更新可能バックアップ閉塞	更新可能バックアップ閉塞，かつクローズ
オープン※1	pdhold -b	×	pdhold -b -u	×
閉塞	×	×	×	×
クローズ	pdhold -b※5	×	pdhold -b -u※5	×
閉塞，かつクローズ	×	pdhold -b※2	×	pdhold -b -u※2
参照可能閉塞	×	×	×	×
参照可能閉塞，かつクローズ	×	×	×	×
参照可能バックアップ閉塞	－	トランザクション終了※3又は pdclose※4	×	×
参照可能バックアップ閉塞，かつクローズ	トランザクション開始※2又は pdopen※4	－	×	×
更新可能バックアップ閉塞	×	×	－	トランザクション終了※3
更新可能バックアップ閉塞，かつクローズ	×	×	トランザクション開始※2	－

（凡例）

－：該当しません。

×：遷移後の RD エリアの状態に遷移できません。

注※1

pdclose コマンドを実行すると，閉塞されていないためエラーとなります。

注※2

RD エリアのオープン契機が DEFER 又は SCHEDULE の場合に状態が遷移します。

注※3

RD エリアのオープン契機が SCHEDULE の場合に状態が遷移します。

注※4

参照可能バックアップ閉塞（更新 WAIT モード）でない場合，状態が遷移します。

注※5

RD エリアのオープン契機が DEFER の場合に状態が遷移します。

付録 C.2　RD エリアの状態によるユティリティ及び UAP の実行可否

ユティリティ及び UAP を実行する場合，RD エリアのオープン契機，及び RD エリアの状態によって実行可否が決まります。RD エリアの状態によるユティリティ及び UAP の実行可否を次の表に示します。

- 表「RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が INITIAL の場合）(1/3)」
- 表「RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が INITIAL の場合）(2/3)」
- 表「RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が INITIAL の場合）(3/3)」
- 表「RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が DEFER 又は SCHEDULE の場合）(1/3)」
- 表「RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が DEFER 又は SCHEDULE の場合）(2/3)」
- 表「RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が DEFER 又は SCHEDULE の場合）(3/3)」

表 C–3　RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が INITIAL の場合）(1/3)

ユティリティ又は UAP			閉塞なし		コマンド閉塞		参照可能閉塞	
			オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース構成変更ユティリティ	RD エリアの追加		－	－	－	－	－	－
	RD エリアの削除		×	×	×	○	×	○
	RD エリアの再初期化		×	×	×	○	×	○
	RD エリアの拡張		○	×	○	×	○	×
	RD エリアの属性変更		×	×	×	○	×	○
	RD エリアの移動	オンライン中の RD エリアの	×	×	×	○	×	○

ユティリティ又は UAP			閉塞なし		コマンド閉塞		参照可能閉塞	
			オープン	クローズ	オープン	クローズ	オープン	クローズ
		移動機能適用なし						
		オンライン中の RD エリアの移動機能適用あり	×	×	×	○	×	×
データベース作成ユティリティ			○※3	×	○※3※4	×	○※3※4	×
データベース再編成ユティリティ	再編成（-k rorg）		○※2※3	×	○	×	○※2※3	×
	アンロード（-k unld）		○	×	○	×	○	×
	リロード（-k reld）		○※2※3	×	○	×	○※2※3	×
	インデクスの一括作成（-k ixmk）		○	×	○	×	○	×
	インデクスの再作成（-k ixrc）		○	×	○	×	○	×
	インデクスの再編成（-k ixor）		○	×	○	×	○	×
	ディクショナリ表のメンテナンス（-k maintenance）		○	×	○	×	○	×
空きページ解放ユティリティ			○	×	○	×	○	×
グローバルバッファ常駐化ユティリティ			○	×	×	×	○	×
リバランスユティリティ			○	×	○	×	○	×
データベース複写ユティリティ	-M x		×	○	×	○	×	○
	-M r		○	○	○	○	○	○
	-M s		○	○	○	○	○	○
データベース回復ユティリティ			×	×	×	○	×	×
データベース状態解析ユティリティ			○	×	○	×	○	×
最適化情報収集ユティリティ			○	×	×	×	○	×
ディクショナリ搬出入ユティリティ			○	×	×	×	×	×
データベース定義ユティリティ			○	×	×	×	×	×
整合性チェックユティリティ			○	×	×	×	×	×

ユティリティ又は UAP	閉塞なし		コマンド閉塞		参照可能閉塞	
	オープン	クローズ	オープン	クローズ	オープン	クローズ
JP1/NETM/Audit 用監査ログ出力ユティリティ	○	×	×	×	○	×
ファイルパス変更オンラインコマンド（pdchpathn コマンド）	○	×	×	×	×	×
UAP	○	×	×	×	○※1	×

（凡例）

○：実行できます。

×：実行できません。

－：該当しません。

注※1

参照系の UAP だけ実行できます。

注※2

改竄防止表に対しては実行できません。

注※3

被参照表に対してユティリティを実行し，参照表を検査保留状態にできるかどうかについては，「検査保留状態の設定可否」を参照してください。

注※4

インポートで表を定義する場合（pdload -b -W -w all 指定時）は実行できません。

表 C–4　RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が INITIAL の場合）（2/3）

ユティリティ又は UAP			参照可能バックアップ閉塞		更新可能バックアップ閉塞	障害閉塞	
			オープン	クローズ	オープン	オープン	クローズ
データベース構成変更ユティリティ	RD エリアの追加		－	－	－	－	－
	RD エリアの削除		×	×	×	×	○
	RD エリアの再初期化		×	×	×	×	○
	RD エリアの拡張		×	×	×	×	×
	RD エリアの変更		×	×	×	×	×
	RD エリアの移動	オンライン中の RD エリアの移動機能適用なし	×	×	×	×	×
		オンライン中の RD エリア	×	×	×	×	×

ユティリティ又は UAP			参照可能バックアップ閉塞		更新可能バックアップ閉塞	障害閉塞	
			オープン	クローズ	オープン	オープン	クローズ
		の移動機能適用あり					
データベース作成ユティリティ			×	×	×	×	×
データベース再編成ユティリティ	再編成（-k rorg）		×	×	×	×	×
	アンロード（-k unld）		○	×	○	×	×
	リロード（-k reld）		×	×	×	×	×
	インデクスの一括作成（-k ixmk）		×	×	×	×	×
	インデクスの再作成（-k ixrc）		×	×	×	×	×
	インデクスの再編成（-k ixor）		×	×	×	×	×
	ディクショナリ表のメンテナンス（-k maintenance）		×	×	○	−	−
空きページ解放ユティリティ			×	×	×	×	×
グローバルバッファ常駐化ユティリティ			○	×	○	×	×
リバランスユティリティ			×	×	×	×	×
データベース複写ユティリティ	-M x		×	○	×	×	○
	-M r		○	○	○※1	×	×
	-M s		○	○	○	○	○
データベース回復ユティリティ			×	×	×	×	○
データベース状態解析ユティリティ			○	×	○	×	×
最適化情報収集ユティリティ			○※2	×	○※2	×	×
ディクショナリ搬出入ユティリティ			×	×	○	×	×
データベース定義ユティリティ			×	×	○	×	×
整合性チェックユティリティ			○	×	○	×	×
JP1/NETM/Audit 用監査ログ出力ユティリティ			○	×	○	×	×
ファイルパス変更オンラインコマンド（pdchpathn コマンド）			×	×	×	×	×
UAP			○※3	×	○	×	×

（凡例）

○：実行できます。

×：実行できません。

－：該当しません。

注※1

更新のトランザクション実行中の場合は実行できません。

注※2

ディクショナリ用 RD エリアが参照可能バックアップ閉塞の場合は実行できません。

注※3

参照系の UAP だけ実行できます。

表 C–5　RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が INITIAL の場合）（3/3）

ユティリティ又は UAP			ログレス閉塞	
			オープン	クローズ
データベース構成変更ユティリティ	RD エリアの追加		－	－
	RD エリアの削除		×	○
	RD エリアの再初期化		×	○
	RD エリアの拡張		×	×
	RD エリアの変更		×	×
	RD エリアの移動	オンライン中の RD エリアの移動機能適用なし	×	×
		オンライン中の RD エリアの移動機能適用あり	×	×
データベース作成ユティリティ			×	×
データベース再編成ユティリティ	再編成（-k rorg）		×	×
	アンロード（-k unld）		×	×
	リロード（-k reld）		×	×
	インデクスの一括作成（-k ixmk）		×	×
	インデクスの再作成（-k ixrc）		×	×
	インデクスの再編成（-k ixor）		×	×
	ディクショナリ表のメンテナンス		－	－

ユティリティ又は UAP		ログレス閉塞	
		オープン	クローズ
	（-k maintenance）		
空きページ解放ユティリティ		×	×
グローバルバッファ常駐化ユティリティ		×	×
リバランスユティリティ		×	×
データベース複写ユティリティ	-M x	×	○
	-M r	×	×
	-M s	○	○
データベース回復ユティリティ		×	○
データベース状態解析ユティリティ		×	×
最適化情報収集ユティリティ		×	×
ディクショナリ搬出入ユティリティ		×	×
データベース定義ユティリティ		×	×
整合性チェックユティリティ		×	×
JP1/NETM/Audit 用監査ログ出力ユティリティ		×	×
ファイルパス変更オンラインコマンド（pdchpathn コマンド）		×	×
UAP		○※	×

（凡例）

○：実行できます。

×：実行できません。

－：該当しません。

注※

PURGE TABLE 文だけ実行できます。

表 C–6　RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が DEFER 又は SCHEDULE の場合）（1/3）

ユティリティ又は UAP		閉塞なし		コマンド閉塞		参照可能閉塞	
		オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース構成変更ユティリティ	RD エリアの追加	－	－	－	－	－	－
	RD エリアの削除	×	×	×	○	×	○
	RD エリアの再初期化	×	×	×	○	×	○
	RD エリアの拡張	○	○	○	×	○	×

ユティリティ又は UAP			閉塞なし		コマンド閉塞		参照可能閉塞	
			オープン	クローズ	オープン	クローズ	オープン	クローズ
	RD エリアの属性変更		×	×	×	○	×	○
	RD エリアの移動	オンライン中の RD エリアの移動機能適用なし	×	×	×	○	×	○
		オンライン中の RD エリアの移動機能適用あり	×	×	×	○	×	×
データベース作成ユティリティ			○※4	○※4※6	○※4※6	×	○※4※6	○※4※6
データベース再編成ユティリティ	再編成（-k rorg）		○※3※4	○※3※4	○	×	○※3※4	○※3※4
	アンロード（-k unld）		○	○	○	×	○	○
	リロード（-k reld）		○※3※4	○※3※4	○	×	○※3※4	○※3※4
	インデクスの一括作成（-k ixmk）		○	○	○	×	○	○
	インデクスの再作成（-k ixrc）		○	○	○	×	○	○
	インデクスの再編成（-k ixor）		○	○	○	×	○	○
	ディクショナリ表のメンテナンス（-k maintenance）		−	−	−	−	−	−
空きページ解放ユティリティ			○	○	○	×	○	○
グローバルバッファ常駐化ユティリティ			○	○	×	×	○	○
リバランスユティリティ			○	○	○	×	○	○
データベース複写ユティリティ	-M x		×	×※5	×	○	×	○
	-M r		○	○	○	○	○	○
	-M s		○	○	○	○	○	○
データベース回復ユティリティ			×	×	×	○	×	×
データベース状態解析ユティリティ			○	○	○	×	○	○
最適化情報収集ユティリティ			○	○	×	×	○	○
ディクショナリ搬出入ユティリティ※1			−	−	−	−	−	−

ユティリティ又は UAP	閉塞なし		コマンド閉塞		参照可能閉塞	
	オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース定義ユティリティ	○	○	×	×	×	×
整合性チェックユティリティ	○	○	×	×	×	×
JP1/NETM/Audit 用監査ログ出力ユティリティ	○	○	×	×	○	○
ファイルパス変更オンラインコマンド（pdchpathn コマンド）	−	−	−	−	−	−
UAP	○	○	×	×	○※2	○※2

（凡例）

○：実行できます。

×：実行できません。

−：該当しません。

注※1

対象となる RD エリアのオープン契機が，DEFER 又は SCHEDULE になることはないため，該当しません。

注※2

参照系の UAP だけ実行できます。

注※3

改竄防止表に対しては実行できません。

注※4

被参照表に対してユティリティを実行し，参照表を検査保留状態にできるかどうかについては，「検査保留状態の設定可否」を参照してください。

注※5

-M x オプション指定の pdcopy コマンドは実行できますが，実行中に更新 SQL，又はデータベースを更新するユティリティ（コマンドも含む）を実行した場合，バックアップ中のデータの整合性は保障されません。

注※6

インポートで表を定義する場合（pdload -b -W -w all 指定時）は実行できません。

表 C–7　RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が DEFER 又は SCHEDULE の場合）（2/3）

ユティリティ又は UAP			参照可能バックアップ閉塞		更新可能バックアップ閉塞		障害閉塞	
			オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース構成変更ユティリティ	RD エリアの追加		－	－	－	－	－	－
	RD エリアの削除		×	×	×	×	×	○
	RD エリアの再初期化		×	×	×	×	×	○
	RD エリアの拡張		×	×	×	×	×	×
	RD エリアの属性変更		×	×	×	×	×	×
	RD エリアの移動	オンライン中の RD エリアの移動機能適用なし	×	×	×	×	×	×
		オンライン中の RD エリアの移動機能適用あり	×	×	×	×	×	×
データベース作成ユティリティ			×	×	×	×	×	×
データベース再編成ユティリティ	再編成（-k rorg）		×	×	×	×	×	×
	アンロード（-k unld）		○	○	○	○	×	×
	リロード（-k reld）		×	×	×	×	×	×
	インデクスの一括作成（-k ixmk）		×	×	×	×	×	×
	インデクスの再作成（-k ixrc）		×	×	×	×	×	×
	インデクスの再編成（-k ixor）		×	×	×	×	×	×
	ディクショナリ表のメンテナンス（-k maintenance）		－	－	－	－	－	－
空きページ解放ユティリティ			×	×	×	×	×	×
グローバルバッファ常駐化ユティリティ			○	○	○	○	×	×
リバランスユティリティ			×	×	×	×	×	×
データベース複写ユティリティ	-M x		×	○	×	○	×	○
	-M r		○	○	○※1	○※1	×	×

ユティリティ又は UAP		参照可能バックアップ閉塞		更新可能バックアップ閉塞		障害閉塞	
		オープン	クローズ	オープン	クローズ	オープン	クローズ
	-M s	○	○	○	○	○	○
データベース回復ユティリティ		×	×	×	×	×	○
データベース状態解析ユティリティ		○	○	○	○	×	×
最適化情報収集ユティリティ		○※3	○※3	○※3	○※3	×	×
ディクショナリ搬出入ユティリティ※2		−	−	−	−	−	−
データベース定義ユティリティ		×	×	○	○	×	×
整合性チェックユティリティ		×	×	○	○	×	×
JP1/NETM/Audit 用監査ログ出力ユティリティ		○	○	○	○	×	×
ファイルパス変更オンラインコマンド（pdchpathn コマンド）		−	−	−	−	−	−
UAP		○※4	○※4	○	○	×	×

（凡例）

○：実行できます。

×：実行できません。

−：該当しません。

注※1

更新のトランザクション実行中の場合は実行できません。

注※2

対象となる RD エリアのオープン契機が，DEFER 又は SCHEDULE になることはないため，該当しません。

注※3

ディクショナリ用 RD エリアが参照可能バックアップ閉塞の場合は実行できません。

注※4

参照系の UAP だけ実行できます。

表 C–8　RD エリアの状態によるユティリティ及び UAP の実行可否（オープン契機が DEFER 又は SCHEDULE の場合）（3/3）

ユティリティ又は UAP		ログレス閉塞	
		オープン	クローズ
データベース構成変更ユティリティ	RD エリアの追加	−	−
	RD エリアの削除	×	○

ユティリティ又は UAP			ログレス閉塞	
			オープン	クローズ
	RD エリアの再初期化		×	○
	RD エリアの拡張		×	×
	RD エリアの変更		×	×
	RD エリアの移動	オンライン中の RD エリアの移動機能適用なし	×	×
		オンライン中の RD エリアの移動機能適用あり	×	×
データベース作成ユティリティ			×	×
データベース再編成ユティリティ	再編成（-k rorg）		×	×
	アンロード（-k unld）		×	×
	リロード（-k reld）		×	×
	インデクスの一括作成（-k ixmk）		×	×
	インデクスの再作成（-k ixrc）		×	×
	インデクスの再編成（-k ixor）		×	×
	ディクショナリ表のメンテナンス（-k maintenance）		−	−
空きページ解放ユティリティ			×	×
グローバルバッファ常駐化ユティリティ			×	×
リバランスユティリティ			×	×
データベース複写ユティリティ	-M x		×	○
	-M r		×	×
	-M s		○	○
データベース回復ユティリティ			×	○
データベース状態解析ユティリティ			×	×
最適化情報収集ユティリティ			×	×
ディクショナリ搬出入ユティリティ※1			−	−
データベース定義ユティリティ			×	×
整合性チェックユティリティ			×	×

ユティリティ又は UAP	ログレス閉塞	
	オープン	クローズ
JP1/NETM/Audit 用監査ログ出力ユティリティ	×	×
ファイルパス変更オンラインコマンド（pdchpathn コマンド）	−	−
UAP	○※	○※

（凡例）

○：実行できます。

×：実行できません。

−：該当しません。

注※

PURGE TABLE 文だけ実行できます。

付録 C.3　検査保留状態の設定可否

被参照表に対してユティリティを実行して，参照表を検査保留状態にする場合，RD エリアのオープン契機，及び RD エリアの状態によって実行可否が変わります。RD エリアの状態によるユティリティでの検査保留状態の設定可否を次の表に示します。

- 表「RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が INITIAL の場合）（1/3）」
- 表「RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が INITIAL の場合）（2/3）」
- 表「RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が INITIAL の場合）（3/3）」
- 表「RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が DEFER 又は SCHEDULE の場合）（1/3）」
- 表「RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が DEFER 又は SCHEDULE の場合）（2/3）」
- 表「RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が DEFER 又は SCHEDULE の場合）（3/3）」

表 C–9　RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が INITIAL の場合）（1/3）

ユティリティ	閉塞なし		コマンド閉塞		参照可能閉塞	
	オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース構成変更ユティリティ	○	△	△	△	△	△

ユティリティ	閉塞なし		コマンド閉塞		参照可能閉塞	
	オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース作成ユティリティ	○	×	○	×	○	×
データベース再編成ユティリティ	○	×	○	×	○	×

（凡例）

○：検査保留状態に設定できます。

△：検査保留状態に設定できません。なお，処理をスキップして，次の処理を続行します。

×：検査保留状態に設定できません。

表 C–10　RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が INITIAL の場合）（2/3）

ユティリティ	参照可能バックアップ閉塞		更新可能バックアップ閉塞	障害閉塞	
	オープン	クローズ	オープン	オープン	クローズ
データベース構成変更ユティリティ	△	△	○	△	△
データベース作成ユティリティ	×	×	○	×	×
データベース再編成ユティリティ	×	×	○	×	×

（凡例）

○：検査保留状態に設定できます。

△：検査保留状態に設定できません。なお，処理をスキップして，次の処理を続行します。

×：検査保留状態に設定できません。

表 C–11　RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が INITIAL の場合）（3/3）

ユティリティ	ログレス閉塞	
	オープン	クローズ
データベース構成変更ユティリティ	△	△
データベース作成ユティリティ	×	×
データベース再編成ユティリティ	×	×

（凡例）

△：検査保留状態に設定できません。なお，処理をスキップして，次の処理を続行します。

×：検査保留状態に設定できません。

表 C–12　RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が DEFER 又は SCHEDULE の場合）（1/3）

ユティリティ	閉塞なし		コマンド閉塞		参照可能閉塞	
	オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース構成変更ユティリティ	○	○	△	△	△	○
データベース作成ユティリティ	○	○	○	×	○	○
データベース再編成ユティリティ	○	○	○	×	○	○

（凡例）

○：検査保留状態に設定できます。

△：検査保留状態に設定できません。なお，処理をスキップして，次の処理を続行します。

×：検査保留状態に設定できません。

表 C–13　RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が DEFER 又は SCHEDULE の場合）（2/3）

ユティリティ	参照可能バックアップ閉塞		更新可能バックアップ閉塞		障害閉塞	
	オープン	クローズ	オープン	クローズ	オープン	クローズ
データベース構成変更ユティリティ	△	△	○	○	△	△
データベース作成ユティリティ	×	×	○	○	×	×
データベース再編成ユティリティ	×	×	○	○	×	×

（凡例）

○：検査保留状態に設定できます。

△：検査保留状態に設定できません。なお，処理をスキップして，次の処理を続行します。

×：検査保留状態に設定できません。

表 C–14　RD エリアの状態によるユティリティでの検査保留状態の設定可否（オープン契機が DEFER 又は SCHEDULE の場合）（3/3）

ユティリティ	ログレス閉塞	
	オープン	クローズ
データベース構成変更ユティリティ	△	△
データベース作成ユティリティ	×	×
データベース再編成ユティリティ	×	×

（凡例）

△：検査保留状態に設定できません。なお，処理をスキップして，次の処理を続行します。

×：検査保留状態に設定できません。

付録 D　ユティリティの最大同時実行数

付録 D.1　各ユティリティの最大同時実行数

ユティリティの最大同時実行数を次の表に示します。この表に示す最大同時実行数は，各ユティリティを単独で同時実行したときの値です。pd_max_users オペランドの値に依存するユティリティの最大同時実行数は，ほかのユティリティ実行数と UAP の実行数に影響されます。

最大同時接続数を超えると，ユティリティは次に示すどれかの動作をします。

- 異常終了します。
- 実行中のユティリティが終了するまで無応答になります。
- プロセス割り当て時の要求電文を格納するための共用メモリ中のバッファが不足し，ユティリティが異常終了します。

表 D–1　ユティリティの最大同時実行数

ユティリティ	最大同時実行数	
	pd_utl_exec_mode=0	pd_utl_exec_mode=1
データベース初期設定ユティリティ（pdinit）	1	
データベース定義ユティリティ（pddef）※1	pd_max_users オペランドの値	
データベース作成ユティリティ（pdload）	• -b -W -w all を指定した場合 MIN（32，pd_max_users オペランドの値）÷2 • -b -W -w all 以外を指定した場合 MIN（32，pd_max_users オペランドの値）	• -b -W -w all を指定した場合 pd_max_users オペランドの値÷2 • -b -W -w all 以外を指定した場合 pd_max_users オペランドの値
会話型 SQL 実行ユティリティ	pd_max_users オペランドの値	
データベース構成変更ユティリティ（pdmod）	1	
ディクショナリ搬出入ユティリティ（pdexp）	1	
リバランスユティリティ（pdrbal）	MIN（32，pd_max_users オペランドの値）	pd_max_users オペランドの値
データベース再編成ユティリティ（pdrorg）	MIN（32，pd_max_users オペランドの値）	pd_max_users オペランドの値
空きページ解放ユティリティ（pdreclaim）	MIN（32，pd_max_users オペランドの値）	pd_max_users オペランドの値

ユティリティ	最大同時実行数	
	pd_utl_exec_mode=0	pd_utl_exec_mode=1
グローバルバッファ常駐化ユティリティ（pdpgbfon）	MIN（32，pd_max_users オペランドの値）	pd_max_users オペランドの値
整合性チェックユティリティ（pdconstck）	pd_max_users オペランドの値÷2	
統計解析ユティリティ（pdstedit）	上限はありません。	
データベース状態解析ユティリティ（pddbst）	MIN（16，pd_max_users オペランドの値）	pd_max_users オペランドの値
最適化情報収集ユティリティ（pdgetcst）	MIN（16，pd_max_users オペランドの値÷2）	pd_max_users オペランドの値÷2
アクセスパス表示ユティリティ（pdvwopt）	上限はありません。	
データベース複写ユティリティ（pdcopy）	32（-b オプションの指定が 1 の場合）※2	pd_max_users オペランドの値
データベース回復ユティリティ（pdrstr）	32	pd_max_users オペランドの値
レジストリ機能初期設定ユティリティ（pdreginit）	1	
JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput）	1	

注※1

定義系 SQL は同時実行できないため，排他待ちが発生します。

注※2

最大同時実行数は，pdcopy に指定する-b オプション（バックアップファイル）の数に依存します。最大同時実行数の考え方は次のとおりです。

- 一つの-b オプション（バックアップファイル）を使うデータベース複写ユティリティを六つ同時に実行すると，最大同時実行数は，1×6 ＝ 6 になります。
- 制御文ファイルを使用して，二つの-b オプション（バックアップファイル）を使うデータベース複写ユティリティを四つ同時に実行すると，最大同時実行数は，2×4 ＝ 8 になります。
- 上記のデータベース複写ユティリティが，すべて同時に動作している場合，データベース複写ユティリティの実行数は 6 ＋ 4 ＝ 10 ですが，最大同時実行数は 6 ＋ 8 ＝ 14 になります。

付録 D.2　複数のユティリティの同時実行数

複数のユティリティを同時実行して，次に示す式を満たす場合，pd_utl_exec_mode オペランドに 1 を指定してください。

HiRDB/パラレルサーバの場合

システム内のバックエンドサーバ数×2×すべてのユティリティの同時実行数≧824

HiRDB/シングルサーバの場合

2×すべてのユティリティの同時実行数≧824

付録 D.3　注意事項

HiRDB/パラレルサーバで，システムマネジャがあるユニットの，フロントエンドサーバの接続ユーザ数がpd_max_users オペランド指定値に達した場合，表「ユティリティの最大同時実行数」に示すユティリティは，同時接続ユーザ数オーバーを検知してエラー終了します。

付録 E　pdload 及び pdrorg で使用する UOC の作成方法と注意事項

付録 E.1　ダイナミックリンクライブラリの作成方法

UOC のコーディングが完了した後，ダイナミックリンクライブラリを作成します。作成したダイナミックリンクライブラリは，呼び出す関数と一緒に，pdload 又は pdrorg の制御文に指定します。

「Microsoft Visual C++ 4.2」を使用して作成する例を，次に示します。

(1)　モジュール定義ファイル（.def）の作成

1. pdload 又は pdrorg からコールする，DLL 内の関数をエクスポートするためのファイルを作成します。DLL 名が sample1.dll，関数名が date_change_func の場合のファイルの作成例を次に示します。

```
sample1.def
  EXPORTS
  date_change_func
```

(2)　DLL の作成

1. Microsoft Visual C++グループ内の Microsoft Developer Studio を起動し，［ファイル］－［新規作成］で種類「プロジェクトワークスペース」，タイプ「Dynamic- Link Library」を選択します。
2. ［挿入］－［プロジェクトへファイルの追加］を選択し，UOC のソースファイル（.c）と 1.で作成したモジュール定義ファイル（.def）を追加します。
3. UOC インタフェース領域や記号定数を宣言した UOC 作成用ヘッダファイル（pdutluoc.h）を引き込む場合は，%PDDIR%¥include をインクルードファイルのパスに設定します。設定方法は，［ビルド］－［設定］－［C/C++］－［プリプロセッサ］－［インクルードファイルのパス］に記述します。
4. 「ビルド」を選択し，DLL のビルドを実行します。この場合，インポートライブラリ（.lib）とエクスポートライブラリファイル（.exp）も作成されていることを確認してください。

(3)　注意事項

- 作成する DLL のベースアドレス（デフォルト・ロード・アドレス）は指定しないでください。指定した場合，HiRDB やシステムの DLL と，アドレスが競合して，DLL のローディング処理に負荷が掛かることがあります。
- Microsoft Visual C++のバージョンは 4.0 以上を使用してください。
- 作成する DLL の呼び出し規約は_cdecl を使用してください（_cdecl は，C と C++プログラミングではデフォルトの呼び出し規約です）。
- 使用する Microsoft Visual C++ランタイムライブラリは，マルチスレッド DLL 版（/MD）でなければなりません。それ以外のライブラリを使用すると，領域管理の処理が不正となり，サーバプロセスが異常終了するおそれがあります。

付録 E.2　UOC を作成する場合の注意事項

UOC を利用する場合，ユーザが作成したプログラムが HiRDB に組み込まれて処理をすることになります。つまり，データベースマネジメントシステムの一部のプログラムという位置づけであるため，UOC の不具合がそのまま HiRDB の不具合に発展するおそれがあります。そのため，UOC を作成する場合は，十分な注意とテストが必要となります。また，UOC 作成時には HiRDB のプログラム規則に従う必要があります。

(1)　pdload 及び pdrorg の共通規則

1. 作成言語は C 言語だけです。
2. UOC は，ダイナミックリンクライブラリの形式で作成してください。
3. 作成したダイナミックリンクライブラリには，読み込み，実行の権限を与えてください。書き込み権限は与えないでください。誤って書き込み権限を与えた場合，UOC の実行時間が長くなり処理性能が悪くなります。
4. データベースにアクセスした関数は，グローバルバッファを占有します。また，バッファの占有データベースにアクセスした関数もグローバルバッファを占有します。バッファの占有時間が長いと，HiRDB はシステムの不具合と認識するため，UOC 側での処理は必要最小限にして，早めに pdload 側又は pdrorg 側に制御を戻すようにしてください。
5. UOC 内で異常終了した場合，pdload 又は pdrorg も異常終了します。異常終了時のデータベースの状態とその回復方法については，「エラー時のデータベースの状態とその回復方法」又は「エラー時のデータベースの状態とその回復方法」を参照してください。
6. UOC インタフェース領域や記号定数を宣言した，UOC 作成用のヘッダファイル（pdutluoc.h）は%PDDIR%¥include 下にあります。インタフェース領域を参照又は更新する場合は，UOC のコンパイル時に引き込んで使用することをお勧めします。なお，変数名称の設定などで，UOC 作成用のヘッダファイルを使用できないで独自に定義する場合，テーブルの構造に違いがあるので注意してください。
7. グローバル変数及び関数名には，次の文字で始まる名称を使用しないでください。
   - 大文字の「SQL」，「Y」，又は「Z」
   - 小文字の「p_」，「pd」，「yy」，又は「z」

   プラグイン及び Java™ ストアドルーチンを使用する場合は，更に次の文字で始まる名称も使用しないでください。

   プラグインを使用する場合：

   小文字の「_p」

   Java ストアドルーチンを使用する場合：

   小文字の「da」

   小文字の「dbr」

   小文字の「dp」
8. UOC のリターン値は，void 型にしてください。
9. UOC で環境変数の設定，変更処理はしないでください。

10. UOC 内では，SQL は発行できません。
11. シグナル操作はしないでください。
12. main 関数は UOC として使用できません。
13. リカーシブルな関数としないでください。
14. スタック不足の場合は，UOC 内で異常終了することがあります。
15. スレッドは生成しないでください。
16. 次の関数以外は使用しないでください。
    - ファイル操作関数
    - 文字処理関数
    - メモリ取得・解放関数
17. 次の関数は使用しないでください。
    - fork()，exit()，abort()，exec()などのプロセス操作関数
    - sleep()，select()，wait()
    - スタック操作関数（setjmp()，longjmp()など）
    - 共用メモリ操作関数
    - セマフォ操作関数
    - ソケット操作関数
    - システム資源操作関数（setrlimit など）
    - mmap()，munmap()
    - gethostent()，sethostent()，endhostent()，gethostbyname()，gethostbyaddr()，herror()
    - tempnam()，tmpnam()
    - pstat()
    - system()
18. UOC が動作しているプロセスのプロセス ID は，UOC 内では使用しないでください。
19. ファイルの操作について，次の点に注意してください。
    - HiRDB インストールディレクトリ，及び HiRDB 運用ディレクトリ下のファイルは操作しないでください。
    - OS のファイルは操作しないでください。
    - ファイルを作成する場合は，絶対パス名で指定してください。絶対パス名で指定しないと，カレントディレクトリ下にファイルが作成されます。HiRDB のプロセスのカレントディレクトリは，%PDDIR%¥tmp¥home 下のディレクトリとなっているため，ここにはファイルを作成しないでください。
    - UOC 内でファイルを作成する場合は，ファイルのパーミッションを明示的に設定してください。

- UOC が一時的に作成したファイルは，必ず削除（unlink）してください。
- PIPE などのスペシャルファイルを使用しないでください。
- 標準入力，標準出力，及び標準エラー出力を使用しないでください。
- ユティリティを複数実行すると，UOC も同時に複数実行されます。このため，UOC がファイル資源に対して排他を掛ける場合は，同時に実行されるほかの UOC が排他待ちにならないように注意してください。
- UOC が出力したファイルは，ユーザが管理してください。

(2)　pdload 固有の規則

1. HiRDB/パラレルサーバの場合，UOC を格納したダイナミックリンクライブラリは source 文に指定したサーバがあるホストに用意してください。
2. pdload がバイナリ形式の入力デ−タファイルを入力して，UOC へ１行分のデータを渡す場合に，可変長文字列の長さが不正なために１行に編集できないときは，処理を打ち切ります。
3. pdload の応答がなくなった場合，pdload と UOC のどちらに制御があるかを調べるには，pdls -d rpc コマンドを実行します。この結果の USR_EVENT が 0x00052601 の場合は，UOC に制御があることになります。
4. pdrorg 用アンロードファイルの場合，UOC を利用したデータロードはできません。
5. エンディアンが異なる pdrorg 出力バイナリ形式ファイルを入力データファイルとする場合，次の点に注意してください。
   - UOC でデータを入力するとき

     データを入力後，エンディアン変換はしないため，入力データは pdload を実行するプラットフォームのエンディアンで渡してください。
   - UOC にデータを渡すとき

     option 文の srcendian オペランドを指定して，エンディアン変換後のデータを UOC に渡します。

(3)　pdrorg 固有の規則

1. UOC を格納したダイナミックリンクライブラリは UOC を呼び出すサーバがあるホストに用意してください。UOC を呼び出すサーバとホストについては，「オプションと制御文の関係」を参照してください。
2. UOC で操作できるファイルは，UOC データファイルだけです。
3. UOC 内で，malloc などの OS 関数でメモリ領域を取得した場合，次の呼び出し方法で必ず解放してください。
   - 開始処理でリターンコードに 0 以外を返す
   - 終了処理
   - 中止処理

4. pdrorg の応答がなくなった場合，pdrorg と UOC のどちらに制御があるかを調べるには，pdls -d rpc コマンドを実行します。この結果の USR_EVENT が 0x00051601 の場合は，UOC に制御があることになります。

付録 F　コマンドの同時接続数

コマンド（運用コマンド及びユティリティ）の中には，内部的に HiRDB に対して接続処理をするものがあります。したがって，コマンドを実行する場合，及びシステム定義の pd_max_users オペランドの指定値を決定する場合，次の点に注意する必要があります。

コマンドの接続時には，pd_max_users オペランドで指定した接続枠を使用するため，ユーザから接続できる数が一時的に減少します。

コマンド実行時に，接続枠の空きよりコマンドの接続の数が多くなる場合，接続エラーとなり，コマンドがエラー終了することがあります。

各コマンドの同時接続数を次の表に示します。なお，同時接続数は 1 回のコマンド実行の値です。

表 F–1　各コマンドの同時接続数

コマンド名	同時接続数
pdacunlck	0
pdadmvr	0
pdaudatld	1
pdaudbegin	0
pdaudend	0
pdaudput	1
pdaudrm	0
pdaudswap	1
pdbkupls	0
pdbufls	0
pdbufmod	0
pdcancel	0
pdcat	0
pdchgconf	0
pdchpathf	0
pdchpathn	0
pdchprc	0
pdclibsync	0
pdclose	0
pdclttrc	0

コマンド名	同時接続数
pdcmt	0
pdconfchk	0
pdconstck	2
pdcopy	0
pdcspool	0
pddbfrz	0
pddbls	0
pddbst	1
pddef	1
pddefrev	1
pdexp	1
pdextfunc	1
pdfbkup	0
pdfchfs	0
pdffsck	0
pdfgt	0
pdfls	0
pdfmkfs	0
pdfrm	0
pdfrstr	0
pdfstatfs	0
pdfzeroinit	0
pdgetcst	2
pdhold	0
pdinfoget	0
pdinit	1
pdjarsync	0
pdkill	0
pdlistls	0
pdload	-b -W -w all を指定した場合は 2，それ以外の場合は 1
pdlogadpf	0

コマンド名	同時接続数
pdlogatul	0
pdlogchg	0
pdlogcls	0
pdloginit	0
pdlogls	0
pdlogopen	0
pdlogrm	0
pdlogswap	0
pdlogsync	0
pdlogucat	0
pdlogunld	0
pdls -d aud	HiRDB 稼働中の場合は 1，停止中の場合は 0
pdls（-d aud 以外）	0
pdmemdb	0
pdmod	1
pdmstchk	1
pdntenv	0
pdobils	0
pdobjconv	1
pdopen	0
pdpfresh	0
pdpgbfon	1
pdplgrgst	1
pdprfed	0
pdprflevel	0
pdprgcopy	0
pdprgrenew	0
pdrbal	1
pdrbk	0
pdrdrefls	1
pdreclaim	1

コマンド名	同時接続数
pdreginit	1
pdrels	0
pdrorg	1
pdparaload	表を構成する RD エリアの数
pdrplstart	0
pdrplstop	0
pdrstr	0
pdsetacl	0
pdsetfw.bat	0
SQL Executer	1
pdstart	0
pdstbegin	0
pdstedit	0
pdstend	0
pdstjswap	0
pdstjsync	0
pdstop	0
pdstscls	0
pdstsinit	0
pdstsopen	0
pdstsrm	0
pdstsswap	0
pdsvhostname	0
pdtrndec	0
pdtrnqing	0
pdvrup	1
pdvwopt	0

付録 G　コマンドのリターンコード一覧

HiRDB のコマンドのリターンコード一覧を次の表に示します。

表 G–1　コマンドのリターンコード一覧

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
pdacunlck	KFPD01113-I	0	0	正常終了。
		8	8	異常終了。
pdadmvr	-	-	0	正常終了。
			8	異常終了（オプション指定不正）。
pdaudatld	-	-	0	正常終了。
			4	警告終了（pdaudatld -t 実行時に動作していた監査証跡表の自動データロードがエラー終了）。
			8	異常終了。
pdaudbegin	-	-	0	正常終了。
			4	正常終了（監査証跡取得開始済みのユニットがあり，処理をスキップしました）。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdaudend	-	-	0	正常終了。
			4	正常終了（監査証跡取得停止済みのユニットがあり，処理をスキップしました）。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdaudput	KFPN10501-I	0	0	正常終了。
		4	4	正常終了（JP1/NETM/Audit 用監査ログ出力ファイルのスワップが発生したことによる終了）。
		8	8	異常終了。
pdaudrm	-	-	0	正常終了。
			8	異常終了。
pdaudswap	-	-	0	正常終了。
			8	異常終了。
pdbkupls	KFPR26276-I	0	0	正常終了。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
		12	12	異常終了。
pdbufls	-	-	0	正常終了。
			4	警告終了（一部のサーバの処理がエラー終了）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pdbufmod	-	-	0	正常終了。
			4	警告終了（一部のサーバの処理がエラー終了）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pdcancel	-	-	0	正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdcat	-	-	0	正常終了。
			8	異常終了。
pdcbl	-	-	0	正常終了。
			4	エラー発生（プロセスを最後まで続行しました）。
			8	エラー発生（プロセスを最後まで続行しました）。
			12	エラー発生（プロセスを途中で終了しました）。
			16	エラー発生（プロセスを途中で終了しました）。
pdchgconf	KFPS04661-I	0	0	正常終了。
		8	8	異常終了。
pdchpathf	KFPH27072-I	0	0	正常終了。
		4	4	異常終了（回復不要）。
		8	8	異常終了（回復要）。
pdchpathn	KFPD01031-I	0	0	正常終了。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
		8	8	異常終了。
pdchprc	-	-	0	正常終了。
			4	一部のユニットが正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdclibsync	-	-	0	正常終了。
			4	異常終了（C ライブラリファイルが登録済み，又は未登録）。
			8	異常終了。
pdclose	-	-	0	正常終了。
			4	警告終了（一部の RD エリアの処理がエラー終了）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pdclttrc	-	-	0	正常終了。
			8	異常終了。
pdcmt	-	-	0	正常終了。
			1	異常終了（オプション指定不正，rsh 失敗など）。
pdconfchk	KFPS05007-I	0	0	正常終了。
		4	4	警告終了（定義に対する警告あり）。
		8	8	異常終了（引数不正，又は pdconfchk コマンドの実行エラー）。
pdconstck	KFPL50001-I	0	0	正常終了。
		4	4	正常終了（制約違反あり）。 制約違反となったキー数が上限値を超え，処理を打ち切った場合も含みます。
		8	8	異常終了。
pdcpp	-	-	0	正常終了。
			4	エラー発生（プロセスを最後まで続行しました）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
			8	エラー発生（プロセスを最後まで続行しました）。
			12	エラー発生（プロセスを途中で終了しました）。
			16	エラー発生（プロセスを途中で終了しました）。
pdcopy	KFPR00756-I	0	0	正常終了。
		8	8	異常終了（一部の複写に失敗，又はスキップしました）。
		12	12	異常終了（すべての複写に失敗しました）。
pdcspool	-	-	0	正常終了。
			4	正常終了。
			12	異常終了。
pddbfrz	-	-	0	正常終了。
			4	警告終了（一部の RD エリアの処理がエラー終了）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pddbls	-	-	0	正常終了。
			4	警告終了。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pddbst	KFPK10301-I	0	0	正常終了。
		4	4	警告終了（指定不正又は解析処理スキップ）。
		8	8	異常終了。
pddef	-	-	0	正常終了。
			8	異常終了。
pddelfw.bat	KFPC00405-I	0	0	正常終了。
	KFPC00406-E	1	1	異常終了。
		2	2	

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
		3	3	
		8	8	
		9	9	
		10	10	
		11	11	
		12	12	
		13	13	
pddefrev	KFPX18400-I	0	0	正常終了。
		8	8	異常終了。
		12	12	異常終了。
pdexp	KFPX18400-I	0	0	正常終了。
		4	4	正常終了（無効なプロシジャ，又はトリガの搬出）。
		8	8	異常終了。
		12	12	異常終了。
pdextfunc	-	-	0	正常終了。
			4	警告終了。
			8	異常終了。
pdfbkup	-	-	0	正常終了。
			1	警告終了。
			255※	異常終了。
pdfchfs	-	-	0	正常終了。
			255※	異常終了。
pdffsck	-	-	0	正常終了。
			1	警告終了（検証で不整合を検知）。
			255※	異常終了。
pdfgt	-	-	0	正常終了。
			1	異常終了（オプション指定不正，rsh 失敗など）。
pdfls	-	-	0	正常終了。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
			255※	異常終了。
pdfmkfs	-	-	0	正常終了。
			-1	異常終了。
			12	セクタ長不正。
pdfrm	-	-	0	正常終了。
			255※	異常終了。
pdfrstr	-	-	0	正常終了。
			1	警告終了。
			255※	異常終了。
pdfstatfs	-	-	0	正常終了。
			255※	異常終了。
pdfzeroinit	-	0	0	正常終了（初期化完了）。
		4	1	警告終了（HiRDB ファイルシステム領域内にファイルが存在するため，初期化に失敗）。
		8	255※	警告終了（引数不正など）。
pdgetcst	KFPN00011-I	0	0	正常終了。
		4	4	正常終了（ディクショナリ表操作で警告発生）。
		8	8	異常終了。
pdhold	-	-	0	正常終了。
			4	警告終了（一部の RD エリアの処理がエラー終了）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pdinfoget	KFPN10401-I	0	0	正常終了。
		4	4	警告終了（一部の情報取得処理をスキップ）。
		8	8	異常終了。
pdinit	KFPX24013-I	0	0	正常終了。
		4	4	正常終了（警告レベルのエラーがありましたが，処理は正常に終了しました）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
		8	8	正常終了（初期化は正常に終了しましたが，初期化終了の通信がエラーになりました）。
		12	12	異常終了。
pdjarsync	-	-	0	正常終了。
			4	異常終了（JAR ファイルが登録済み，又は未登録）。
			8	異常終了。
pdkill	-	-	0	正常終了。
			-1	異常終了。
pdlistls	-	-	0	正常終了。
			8	異常終了。
pdload	KFPL00704-I	0	0	正常終了。
		4	4	正常終了（入力データエラーの検知）。
		8	8	異常終了。
pdlogadpf	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdlogatul	-	-	0	正常終了。
			4	中断，又は時間超過。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdlogchg	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdlogcls	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
pdloginit	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdlogls	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdlogopen	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdlogrm	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdlogswap	-	-	0	正常終了。
			4	スワップしましたが，シンクポイントダンプが有効化されないでコマンドを強制停止しました。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdlogsync	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdlogucat	-	-	0	正常終了。
			8	異常終了。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
pdlogunld	-	-	0	正常終了。
			4	異常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdls（-d act）	-	-	0	正常終了。
			8	異常終了。
pdls（-d aud）	-	-	0	正常終了。
			8	異常終了。
pdls（-d ha）	-	-	0	正常終了。
			8	異常終了。
pdls（-d lck）	-	-	0	正常終了。
			8	異常終了。
pdls（-d mem）	-	-	0	正常終了。
			8	異常終了。
pdls（-d prc）	-	-	0	正常終了。
			8	異常終了。
pdls（-d rpc）	-	-	0	正常終了。
			8	異常終了。
pdls（-d rpl）	-	-	0	正常終了。
			8	異常終了。
pdls（-d scd）	-	-	0	正常終了。
			8	異常終了。
pdls（-d stj）	-	-	0	正常終了。
			8	異常終了。
pdls（-d sts）	-	-	0	正常終了。
			8	異常終了。
pdls（-d svr）	-	-	0	全ユニットが ACTIVE で，かつ HiRDB 開始処理完了（KFPS05210-I メッセージ出力後）。なお，HiRDB 開始後に pdstop -s コマンドで

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
				すべてのフロントエンドサーバを終了した状態の場合も 0 となります。
			4	一部のユニットが非 ACTIVE。
				HiRDB 開始処理の完了前（フロントエンドサーバの状態が SUSPEND）。
			8	異常終了。
pdls（-d ust）	-	-	0	ユニット稼働中（コマンド実行ユニットにフロントエンドサーバがある場合は，フロントエンドサーバに接続できます）。
			1	異常終了。
			4	ユニット開始途中（コマンド実行ユニットにフロントエンドサーバがあるが，フロントエンドサーバに接続できません），又はユニット停止途中。
			8	プロセスサーバプロセスの再起動中断状態。
			12	ユニットの停止状態。
			16	サービス停止状態。
pdls（-d trn）	-	-	0	正常終了。
			8	異常終了。
pdmemdb	-	-	0	正常終了。
			4	警告終了（一部の RD エリアの処理がエラー終了しました）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生しました）。
pdmod	KFPX24213-I	0	0	正常終了。
		4	4	正常終了（警告レベルのエラーがありましたが，処理は正常に終了しました）。
		8	8	正常終了（一部の構成変更処理が正常終了）。
		12	12	異常終了。
		16	16	異常終了（データベースの不整合）。
pdmstchk	KFPH27072-I	0	0	正常終了（整合性に問題なし）。
		4	4	正常終了（整合性に問題あり）。
		8	8	異常終了（コマンド実行中にエラーが発生）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
pdntenv	-	-	0	正常終了。
			8	異常終了。
pdobils	-	-	0	正常終了。
			4	異常終了。
pdobjconv	KFPX21002-I	0	0	正常終了。
		4	4	警告終了。
		8	8	正常終了しましたが，一部再登録に失敗したものがあります。又は，コマンド実行上のエラーが発生しました。
		12	12	異常終了。
pdocb	-	-	0	正常終了。
			4	エラー発生（プロセスを最後まで続行しました）。
			8	エラー発生（プロセスを最後まで続行しました）。
			12	エラー発生（プロセスを途中で終了しました）。
			16	エラー発生（プロセスを途中で終了しました）。
pdocc	-	-	0	正常終了。
			4	エラー発生（プロセスを最後まで続行しました）。
			8	エラー発生（プロセスを最後まで続行しました）。
			12	エラー発生（プロセスを途中で終了しました）。
			16	エラー発生（プロセスを途中で終了しました）。
pdopen	-	-	0	正常終了。
			4	警告終了（一部の RD エリアの処理がエラー終了）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
pdparaload	KFPL00794-I	0	0	正常終了。
		4	4	正常終了（一部の入力データに論理エラーが発生したため，データベースへの格納処理をスキップしました）。
		8	8	異常終了。
pdpfresh	KFPS00730-I	0	0	正常終了。
		1	1	正常終了（リフレッシュ中プロセスあり）。
		4	4	警告終了（一つ以上のサーバでエラー発生）。
		6	6	警告終了（運用コマンド又はユティリティ実行中のため，リフレッシュ処理をしないで終了）。
		8	8	異常終了（オプション指定不正，rsh 失敗など）。
pdpgbfon	KFPL00738-I	0	0	正常終了。
		4	4	正常終了（グローバルバッファ上への読み込み時にバッファ面数が不足したため，バッファ落ちが発生しました）。
		8	8	異常終了。
		12	12	異常終了（pdrorg が異常終了）。
pdplgrgst	KFPY02016-I	0	0	登録・削除成功。
		8	8	登録・削除失敗。
pdprfed	-	-	0	正常終了。
			8	異常終了。
			12	異常終了（内部矛盾）。
pdprflevel	-	-	0	正常終了。
			8	異常終了。
pdprgcopy	KFPS04645-I	0	0	正常終了。
		8	8	異常終了。
pdprgrenew	KFPS04646-I	0	0	正常終了。
		8	8	異常終了（入れ替え前の HiRDB の状態）。
		12	12	異常終了（HiRDB が終了）。
	KFPS05033-E	8	8	異常終了（system 関数異常終了）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
pdrbal	KFPL33001-I	0	0	正常終了。
		4	4	正常終了（リバランスの一時停止）。
		8	8	異常終了。
pdrbk	-	-	0	正常終了。
			1	異常終了（オプション指定不正，rsh 失敗など）。
pdrdrefls	-	-	0	正常終了。
			4	警告終了（解析対象の資源がありません）。
			8	異常終了。
pdreclaim	KFPL00739-I	0	0	正常終了。
		4	4	正常終了（UAP のトランザクション決着待ち時間のタイムオーバーが発生したため，処理を中断しました）。
		8	8	異常終了。
pdreginit	KFPX24601-I	0	0	正常終了。
		4	4	正常終了（警告レベルのエラーがありましたが，処理は正常に終了しました）。
		8	8	異常終了。
pdrels	-	-	0	正常終了。
			1	警告終了。 更新可能バックアップ閉塞（WAIT モード）時に，更新バッファの内容が RD エリアに反映されました。閉塞解除処理は正常終了しています。
			4	警告終了（複数指定した RD エリアのうち，一つ以上にエラーがありました）。
			8	異常終了。
			12	異常終了（エラーメッセージを表示できない事象が発生）。
pdrorg	KFPL00719-I	0	0	正常終了。
		4	4	正常終了（処理を一部スキップしました）。
		8	8	異常終了。
pdrplstart	-	-	0	正常終了。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
			8	異常終了。
pdrplstop	-	-	0	正常終了。
			4	異常終了（一部のユニットが停止中（縮退起動によって開始されていないユニットは除く），又は HiRDB Datareplicator との連携で未抽出ログがあります）。
			8	異常終了。
pdrstr	KFPR00765-I	0	0	正常終了（すべての回復に成功した，又は KFPR26298-W メッセージを出力しているがすべての回復に成功した）。
		4	4	警告終了（回復処理は正常終了しましたが，回復処理に関係のない警告レベルのエラーが発生しました）。
		8	8	異常終了（回復処理中にエラーが発生しましたが，一部回復された RD エリアもあります）。
		12	12	異常終了。
pdsetacl	-	-	0	正常終了。
			4	警告終了（一部のディレクトリ・ファイルのアクセス権限設定に失敗しました）。
			8	異常終了。
pdsetfw.bat	KFPC00405-I	0	0	正常終了。
	KFPC00406-E	1	1	異常終了。
		2	2	
		3	3	
		8	8	
		9	9	
		10	10	
		11	11	
		12	12	
		13	13	
pdstart	-	-	0	正常終了。
			4	異常終了（タイムアウト）。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
			8	異常終了（オプション指定不正，又は HiRDB 起動処理失敗）。
pdstbegin	-	-	0	正常終了。
			4	異常終了（一部ユニットの統計情報取得開始に失敗している）。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdstedit	KFPK00300-I	0	0	正常終了。
		4	0	警告終了。
		8	1	異常終了。
pdstend	-	-	0	正常終了。
			4	異常終了（一部ユニットの統計情報の出力停止に失敗している）。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdstjswap	-	-	0	正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdstjsync	KFPS05840-I	-	0	正常終了。
	KFPS01841-E	-	8	異常終了（オプション指定不正，又は統計情報未取得）。
pdstop	-	-	0	正常終了。
			4	異常終了（計画停止の通信タイムアウト）。
			8	異常終了（オプション指定不正，又は HiRDB 停止処理失敗）。
pdstscls	-	-	0	正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdstsinit	-	-	0	正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdstsopen	-	-	0	正常終了。

コマンド名	出力メッセージ		リターンコード	内容
	メッセージ ID	コード		
			8	異常終了（オプション指定不正，rsh 失敗など）。
pdstsrm	-	-	0	正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
			12	異常終了（IP アドレス引き継ぎなし構成の待機系でリトライしたとき）。
pdstsswap	-	-	0	正常終了。
			8	異常終了（オプション指定不正，rsh 失敗など）。
	KFPS05210-I	-	0	正常終了。
pdsvhostname	-	-	0	正常終了。
			1	正常終了（サーバ停止中）。
			4	異常終了（HiRDB 停止中）。
			8	異常終了（システムマネジャ以外のサーバマシンで実行，指定したサーバがないなど）。
pdtrndec	-	-	0	正常終了。
			4	警告終了（一つ以上の未決着状態のトランザクションがあります）。
			8	異常終了（オプション指定不正）。
pdtrnqing	-	-	0	正常終了。
			4	警告終了（システム定義の pd_ha_transaction=queing なし）。
			8	警告終了（実行環境不正，操作誤りなど）。
			12	異常終了（上記以外）。
pdvrup	KFPX24404-I	0	0	正常終了。
		4	4	正常終了（pdvrup 実行時にエラーが発生しましたが，バージョンアップは正常に終了しました）。
		12	12	異常終了。
pdvwopt	-	-	0	正常終了。
			8	異常終了。

（凡例）

-：メッセージ，又はコードは出力されません。

注※

使用しているシェルによっては-1 になります。

付録 H　コマンドが出力するファイル一覧

HiRDB のコマンドが出力するファイル一覧を次の表に示します。

なお，「ファイルの出力先」が複数ある場合は，優先順位に従いファイルの出力先が決まります。1 の出力先が一番優先順位が高くなります。

表 H–1　HiRDB のコマンドが出力するファイル一覧

コマンド名	出力ファイル， 又は出力ディレクトリ	ファイルの出力先	自動削除の対象可否
pdaudput	pdaudput 用管理ファイル	1. -d オプションの指定先 2. $PDDIR/auditlog 下	×
	JP1/NETM/Audit 用監査ログ出力ファイル	1. -d オプションの指定先 2. $PDDIR/auditlog 下	×
pdcancel	core ファイル	1. $PDDIR/spool/save 下	×
	deb ファイル	1. $PDDIR/spool/save 下	×
pdchpathf	パス変更情報ファイル	1. コマンドの指定先	×
	コマンド作業用一時ファイル	1. pd_tmp_directory オペランドの指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	○
pdconstck	処理結果ファイル	1. -o オプションの指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	×
pdcopy	バックアップファイル	1. -b オプションの指定先	×
	処理結果出力ファイル	1. -p オプションの指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	×
	ログポイント情報ファイル	1. -z オプションの指定先	×
	差分バックアップの履歴情報ファイル	1. -o オプションの指定先	×
pddbst	再編成時期予測機能の CSV 出力ファイル	1. predict 文の predict file オペランドの指定先	×
	作業用ワークファイル用ディレクトリ	1. workdir 文の指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	×

コマンド名	出力ファイル， 又は出力ディレクトリ	ファイルの出力先	自動削除の対象可否
pddefrev	定義系 SQL 出力ファイル	1. -o オプションの指定先	×
	ディクショナリ搬出用のワークディレクトリ	1. -w オプションの指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. %PDDIR%¥tmp 下	○
pdexp	搬出ファイル	1. -e 又は-i オプションの指定先	×
	定義系 SQL 出力ファイル	1. -o オプションの指定先	×
pdgetcst	出力結果ファイル	1. -l オプションの指定先	×
	最適化情報パラメタファイル	1. -e オプションの指定先	×
pdinfoget	初期情報ファイル出力先ディレクトリ	1. -e オプションの指定先	○
	詳細情報ファイル出力先ディレクトリ	1. -d オプションの指定先	○
	ワークディレクトリ	1. -w オプション指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	○
pdjarsync	JAR ファイル	1. pd_java_archive_directory オペランドの指定先 2. $PDDIR/java 下	×
pdload	LOB 中間ファイル	1. lobmid 文の指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	×
	インデクス情報ファイル	1. index 文の指定先 2. idxwork 文の指定先 3. pd_tmp_directory オペランドの指定先 4. 環境変数 TMPDIR の指定先 5. /tmp 下	○
	ソート用ワークファイル	1. sort 文の指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	○
	エラー情報ファイル	1. source 文の指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	×
	エラー情報ファイル作成用一時ファイル	1. pd_tmp_directory オペランドの指定先	○

コマンド名	出力ファイル， 又は出力ディレクトリ	ファイルの出力先	自動削除の対象 可否
		2. 環境変数 TMPDIR の指定先 3. /tmp 下	
	エラーデータファイル	1. source 文の errdata オペランドの指定先	×
	エラーデータファイル作成用一時ファイル	1. source 文の errdata オペランドの指定先	○
	処理性能情報ファイル	1. $PDDIR/spool 下	×
	ユーザ用トラブルシュート情報ファイル※	1. /tmp 下及び$PDDIR/spool/save 下	×
	ワークファイル	1. pd_tmp_directory オペランドの指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	×
pdlogunld	アンロードログファイル	1. -o オプションの指定先 2. 標準出力	×
pdparaload	pdload 制御文ファイル	1. pd_tmp_directory オペランドの指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	○
	pdload が出力するファイル	pdload コマンドが出力するファイルと同じです。詳細は，pdload の項目を参照してください。	−
pdpgbfon	ユーザ用トラブルシュート情報ファイル※	1. /tmp 下及び$PDDIR/spool/save 下	×
pdrbal	処理結果ファイル	1. report 文の report オペランドの指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	×
	インデクス情報ファイル	1. index 文の指定先 2. idxwork 文の指定先 3. pd_tmp_directory オペランドの指定先 4. 環境変数 TMPDIR の指定先 5. /tmp 下	○
	ソート用ワークファイル	1. sort 文の指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	○
	ワークファイル	1. pd_tmp_directory オペランドの指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	×

コマンド名	出力ファイル， 又は出力ディレクトリ	ファイルの出力先	自動削除の対象可否
pdreclaim	ユーザ用トラブルシュート情報ファイル※	1. /tmp 下及び$PDDIR/spool/save 下	×
pdrorg	アンロードファイル	1. unload 文の指定先	×
	LOB データのアンロードファイル	1. lobunld 文の指定先	×
	処理性能情報ファイル	1. $PDDIR/spool 下	×
	インデクス情報ファイル	1. index 文の指定先 2. idxwork 文の指定先 3. pd_tmp_directory オペランドの指定先 4. 環境変数 TMPDIR の指定先 5. /tmp 下	○
	ソート用ワークファイル	1. sort 文の指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	○
	ユーザ用トラブルシュート情報ファイル※	1. /tmp 下及び$PDDIR/spool/save 下	×
	ワークファイル	1. pd_tmp_directory オペランドの指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	×
pdrstr	処理結果出力ファイル	1. -p オプションの指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	×
	ソート用ワークディレクトリ	1. -w オプションの指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	○
pdstbegin	統計ログファイル	1. $PDDIR/spool 下	×
pdstedit	DAT 形式ファイル	1. -o オプション指定先	×
	ワーク用一時ファイル	1. -w オプション指定先 2. pd_tmp_directory オペランドの指定先 3. 環境変数 TMPDIR の指定先 4. /tmp 下	○
	ソート用ワークファイル	1. 環境変数 TMPDIR の指定先 2. /tmp 下	○
pdstop	共用メモリダンプファイル	1. $PDDIR/spool/pdshmdump/shmdump 下	×

コマンド名	出力ファイル， 又は出力ディレクトリ	ファイルの出力先	自動削除の対象可否
	接続ユーザ情報ファイル	1. $PDDIR/spool/cnctusrinf 下	×
	接続ユーザ詳細ファイル	1. $PDDIR/spool/cnctusrdtl 下	×
pdtrndec	シェルスクリプトの出力先ディレクトリ	1. -o オプションの指定先 2. コマンドを実行したカレントディレクトリ下	×
	実行履歴の出力先ディレクトリ	1. -o オプションの指定先 2. コマンドを実行したカレントディレクトリ下	×
pdchpathf	パス変更情報ファイル	1. コマンドの指定先	×
	コマンド作業用一時ファイル	1. pd_tmp_directory オペランドの指定先 2. 環境変数 TMPDIR の指定先 3. /tmp 下	○

（凡例）

○：コマンドの終了時に，自動的にファイルが削除されます。

×：コマンドの終了時に，自動的にファイルが削除されません。ファイルは残ります。

−：該当しません。

注※

pd_utl_exec_time オペランドに指定したユティリティの実行監視時間を超えてもユティリティが終了しない場合に出力されるファイルのことです。詳細については，マニュアル「HiRDB システム定義」の pd_utl_exec_time オペランドの説明を参照してください。

付録 I　バージョンアップによって省略値が変更，又は指定不要になったオプション及び制御文

付録 I.1　省略値が変更，又は指定不要になったオプション，制御文

バージョン 09-50 以降で変更したユティリティ及び運用コマンドの，オプション及び制御文の一覧を次の表に示します。

表 I–1　バージョン 09-50 以降で変更したユティリティ及び運用コマンドのオプション及び制御文の一覧

分類	コマンド名	オプション，制御ファイル，制御文種別		説明	省略値変更の有無	指定値の範囲の変更の有無	公開又は非公開	省略値変更のメリット	省略値変更のデメリット
ユティリティ	pdreclaim	オプション	-j 又は-a	空きセグメント解放オプション -j：オンライン業務と同時実行不可 -a：オンライン業務と同時実行可能	○	−	公開	オンライン中にセグメント解放する場合のオプション指定不要化	pdreclaim の処理時間増
			-o	ページコンパクションを実行しない	○	−	非公開	コンパクション要否自動判定による指定不要化	なし
		制御文	option globalbuffer_lru	グローバルバッファへの読み込みページを最古ページにする場合に指定する	○	−	公開	オンライン性能安定化	pdreclaim の処理時間増
			option io_interruption_interval	空きページ解放時の休止間隔	−	−	非公開	ユティリティ実行時の制御文指定不要化	なし
	pddbst	オプション	-b	pddbst 実行によるアプリケーションへのオンライン性能劣化を防ぐ場合に指定する	○	−	公開	オンライン性能安定化	なし

分類	コマンド名	オプション，制御ファイル，制御文種別		説明	省略値変更の有無	指定値の範囲の変更の有無	公開又は非公開	省略値変更のメリット	省略値変更のデメリット
	pdrorg	オプション	-j	BLOB 列のアンロード，リロード	○	−	公開	データ移行を行う際のトラブル防止及びオプション指定不要化	なし
	pdload	制御文	option file_buff_size	seq_range=buff の場合に使用する入力バッファ用サイズ	○	−	公開	自動採番機能を使用してデータロードする場合のオペランド指定不要化	シングルサーバ，バックエンドサーバ，及びユティリティサーバのプロセス固有メモリ増加
	pdload，及び pdrorg	制御文	report	処理性能情報ファイル名	○	−	非公開	性能トラブル要因の特定容易化	HiRDB 運用ディレクトリ下のファイル容量増加
	pdload，pdrorg，及び pdrbal	制御文	sort	ソート用バッファサイズ	−	−	非公開	ユティリティ実行時のオペランド指定不要化	なし
運用コマンド	pdntenv	オプション	-k	OS シャットダウン時の HiRDB の終了方法（none 又は force）	−	−	非公開	シャットダウンに起因する障害防止	なし
	pdfmkfs	オプション	-k UTL 又は NUTL	UTL を NUTL と同じ動作にして UTL に一本化	−	○	公開	UTL 又は NUTL の選択不要化	なし

（凡例）

○：変更があります。

−：変更はありません。

(1)　省略値が変更になったオプション及び制御文

省略値が変更になったオプション及び制御文があります。

バージョンアップを行うユーザは，次に示す表を参照して，指定内容を見直してください。省略値が変更されたことで問題がある場合は，オプション及び制御文を明示的に指定してください。なお，新規に HiRDB を構築するユーザは指定内容の見直しは不要です。

表 I–2　省略値が変更になったオプション及び制御文

コマンド名	種別	オプション又は制御文	バージョン 09-50 より前	バージョン 09-50 以降	
				互換モード	推奨モード
pdrorg	オプション	-j	次の場合に-j を仮定します。 • スキーマ単位（-t all）での再編成，アンロード，又はリロード • エクスポート（-W -w）	次の場合に-j を仮定します。 • スキーマ単位（-t all）の再編成，アンロード，又はリロード • エクスポート（-W -w） • アンロードデータファイルをデータベース作成ユティリティの入力データファイルとして使用する場合，又は UAP などで使用する場合（-W）	同左
pddbst	オプション	-b	-b オプションを仮定しません。	-s オプションを指定していない場合，-b オプションを仮定します。	同左
pdreclaim	オプション	-j，又は-a	-j 又は-a オプションを仮定しません。	-j 又は-a オプションを仮定しません（-Z を仮定します）。	-a オプションを仮定します。
	制御文	option globalbuffer_lru	yes	同左	no
pdload	制御文	option file_buff_size	システム定義 pd_utl_file_buff_size オペランドの値を仮定します。	1024	同左
pdload，及び pdrorg	制御文	report	処理性能情報ファイルを出力しません。	$PDDIR/spool/utlrpt 下に処理性能情報ファイルを出力します。	同左

(2)　指定値の範囲が変更になったオプション

指定値の範囲が変更になったオプションがあります。

バージョンアップを行うユーザは，次に示す表を参照して，指定内容を見直してください。バージョンアップ時にバージョン 09-50 以降指定できなくなったオプションの指定値を指定していてもエラーにはなりま

せんが，指定値を変更することを検討してください。なお，新規に HiRDB を構築するユーザは指定内容の見直しは不要です。

表 I–3　指定値の範囲が変更になったオプション

コマンド名	種別	オプション	バージョン 09-50 より前	バージョン 09-50 以降	
				互換モード	推奨モード
pdfmkfs	オプション	-k	[-k {DB|SDB|SYS|WORK|UTL|NUTL|SVR}]	[-k {DB|SDB|SYS|WORK|UTL|SVR}] • UTL 指定時は，Windows のキャッシュを使用しない領域として初期化します。 • バージョン 09-50 より前の HiRDB で UTL 指定で作成した HiRDB ファイルシステム領域は，バージョン 09-50 以降は Windows のキャッシュを使用しない領域として扱います。 • バージョン 09-50 より前の HiRDB で NUTL 指定で作成した HiRDB ファイルシステム領域は，バージョン 09-50 以降も動作に変更はありません（Windows のキャッシュを使用しません）。 • NUTL は互換のために指定可能ですが，NUTL を指定すると UTL を指定したものとみなします。	

(3)　指定不要となったオプション及び制御文で，省略値に変更があるもの

指定不要となったオプション及び制御文で，省略値に変更があるものがあります。

バージョンアップを行うユーザは，以下を参照し指定内容の見直しを行ってください。省略値が変更されたことで問題がある場合は，オプション及び制御文を明示的に指定してください。なお，新規に HiRDB を構築するユーザは指定内容の見直しは不要です。

表 I–4　指定不要となったオプション及び制御文で，省略値に変更があるもの

コマンド名	種別	オプション又は制御文	バージョン 09-50 より前	バージョン 09-50 以降	
				互換モード	推奨モード
pdload pdrorg	制御文	report	チューニング情報を出力しません。	チューニング情報を出力します。	
pdreclaim	オプション	-o	表のページコンパクションションを実行します。	表「pdreclaim -o オプションのバージョン 09-50 以降の省略値」を参照してください。	

表 I–5　pdreclaim -o オプションのバージョン 09-50 以降の省略値

システム共通定義（pd_dbreuse_remaining_entries 指定）	バージョン 09-50 以降	
	互換モード	推奨モード
ALL	表のページコンパクションを実行しません（-o オプション指定ありとして動作します）。	
ONLY_DIC，または，NONE	• ユーザ表の場合，表のページコンパクションを実行します（-o オプション指定なしとして動作します）。 • データディクショナリ表の場合，表のページコンパクションを実行しません（-o オプション指定ありとして動作します）。	
ONLY_USER	• ユーザ表の場合，表のページコンパクションを実行しません（-o オプション指定ありとして動作します）。 • データディクショナリ表の場合，表のページコンパクションを実行します（-o オプション指定なしとして動作します）。	
NOTHING	表のページコンパクションを実行します（-o オプション指定なしとして動作します）。	

(4)　指定不要となったオプション及び制御文で，省略値に変更がないもの

指定不要となったオプション及び制御文で，省略値に変更がないものがあります。

バージョンアップを行うユーザは，次に示す表を参照して，指定内容を見直してください。なお，新規に HiRDB を構築するユーザは指定内容の見直しは不要です。

表 I–6　指定不要となったオプション及び制御文で，省略値に変更がないもの

コマンド名	種別	オプション又は制御文	バージョン 09-50 より前	バージョン 09-50 以降	
				互換モード	推奨モード
pdload pdrorg pdrbal	制御文	sort [,ソート用バッファサイズ]	1024	同左	
pdreclaim	制御文	option io_interruption_interval	空きページ解放時の休止はしません。	同左	
pdntenv	オプション	-k	force	同左	

索引

記号

-a〔pdload〕 636
-a〔pdreclaim〕 1317
-b〔pdload〕 636
-d〔pdload〕 635
-g〔pdrorg〕 1104
-j〔pdreclaim〕 1317
-j〔pdrorg〕 1105
-U〔pdload〕 636
-w〔pdload〕 636
-W〔pdload〕 636
-W〔pdrorg〕 1100
-Z〔pdreclaim〕 1317

数字

1 TO 1〔pdvwopt〕 1693
1 対 1 転送 1693
64 ビットモードの HiRDB への SQL オブジェクトの移行 437

A

ADEC 型 739
ALL〔pdgetcst〕 1615
all〔pdload〕 636
all〔pdpgbfon〕 1346
all〔pdreclaim〕 1308
all〔pdrorg〕 1091
allspace〔pdload の option 文〕 712
alter HiRDB mode to parallel 文〔pdmod〕 963
alter rdarea 文〔pdmod〕 956
alter system 文〔pdmod〕 953
AND PLURAL INDEXES SCAN〔pdvwopt〕 1692
array 文〔pdload〕 684
array 文〔pdrorg〕 1137

B

BDEC 型 741
bloblimit〔pdload の option 文〕 706
bloblimit〔pdrorg の unlduoc 文〕 1135
blobtovarchar 文〔pdrorg〕 1157
BROADCAST〔pdvwopt〕 1694

C

charset〔pdload の option 文〕 714
charset〔pdrorg の option 文〕 1155
cntlcode〔pdrorg の fixtext_option 文〕 1158
CONNECT/DISCONNECT に関する統計情報 1495
constraint 文〔pdload〕 694
constraint 文〔pdrorg〕 1141
create auditor 文〔pdmod〕 966
create audit table 文〔pdmod〕 968
create rdarea 文〔pdinit〕 569
create rdarea 文〔pdmod〕 900
create rdarea 文〔pdreginit〕 1822
create rdarea 文の指定順序 580
CROSS JOIN〔pdvwopt〕 1691
csetnck〔pdload〕 636
cutdtmsg〔pdload の option 文〕 705
C ライブラリファイルの操作 136

D

dataerr〔pdload の option 文〕 708
data〔pdload〕 636
DAT 形式 608, 717
DAT 形式ファイル出力先ディレクトリ名〔pdstedit〕 1396
DAT 形式ファイルへの統計情報出力 1452
DB 状態解析機能 1503
DB メンテナンス予定日の情報 1581
DECIMAL 型の符号正規化機能 817, 1199
decin〔pdload の extdat 文〕 688
define system 文〔pdinit〕 568
deputy〔pdmod の create auditor 文〕 966
dicinf〔pdinit の define system 文〕 568

dictionary datatype mchar〔pdinit の define system 文〕 568
divermsg〔pdload の option 文〕 711
div〔pdload〕 647
div〔pdrorg〕 1108

E

elmtype〔pdload の array 文〕 685
elmtype〔pdrorg の array 文〕 1138
enclose_del〔pdload の列名文〕 777
enclose〔pdload の extdat 文〕 688
enclose〔pdrorg の fixtext_option 文〕 1159
entry〔pdload の srcuoc 文〕 682
entry〔pdrorg の unlduoc 文〕 1135
errdata〔pdload の source 文〕 668
error〔pdload の lobmid 文〕 681
error〔pdload の source 文〕 667
errwork〔pdload の source 文〕 669
execstop 文〔pdrbal〕 1270
exectime〔pdload の option 文〕 707
exectime〔pdrorg の option 文〕 1152
expand rdarea 文〔pdmod〕 913
extdat 文 687

F

FF 形式 685
FF 形式でのナル値の扱い 748
file_buff_size〔pdload の option 文〕 714
filedir〔pdload の列名文〕 779
fixrow〔pdrorg の unlduoc 文〕 1136
fixtext_option 文〔pdrorg〕 1158
format〔pdrorg の fixtext_option 文〕 1160
FULL SCAN TYPE プラグイン提供関数 1699
func〔pdload の列名文〕 779
func〔pdrorg の reld_func 文〕 1140
func〔pdrorg の unld_func 文〕 1139
FV 形式 685

H

HASH〔pdvwopt〕 1693
HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更 960
HiRDB Datareplicator 連携に関する情報 352, 354
HiRDB Datareplicator 連携の開始 483
HiRDB Datareplicator 連携の終了 485
HiRDB Datareplicator 連携の状態表示 351
HiRDB システムの開始 495
HiRDB システムの終了 526
HiRDB システムの状態表示 313
HiRDB の系の状態表示 321
HiRDB の製品情報の取得 65
HiRDB の動作環境の設定 394
HiRDB バージョンアップ 563
HiRDB ファイル 49
HiRDB ファイルシステムの内容表示 196
HiRDB ファイルシステムのバックアップ 181
HiRDB ファイルシステムのリストア 214
HiRDB ファイルシステム領域内の使用済み領域の初期化 223
HiRDB ファイルシステム領域の管理情報変更 184
HiRDB ファイルシステム領域の管理領域サイズ 210
HiRDB ファイルシステム領域の状態表示 217
HiRDB ファイルシステム領域の使用目的 202
HiRDB ファイルシステム領域の初期設定 201
HiRDB ファイルシステム領域の整合性の検証及び修復 188
HiRDB ファイルシステム領域名〔pdinit の create rdarea 文〕 576
HiRDB ファイルシステム領域名〔pdmod の create rdarea 文〕 908
HiRDB ファイルシステム領域名〔pdmod の expand rdarea 文〕 915
HiRDB ファイルシステム領域名〔pdmod の initialize rdarea 文〕 928
HiRDB ファイルシステム領域名〔pdreginit の create rdarea 文〕 1825
HiRDB ファイルシステム領域名〔pdstedit〕 1394
HiRDB ファイル情報〔pddbst〕 1515

HiRDB ファイル情報〔pdrorg の lobunld 文〕 1133
HiRDB ファイル情報〔pdrorg の unload 文〕 1124
HiRDB ファイル内セグメント数〔pdinit の create rdarea 文〕 577
HiRDB ファイル内セグメント数〔pdmod の create rdarea 文〕 909
HiRDB ファイル内セグメント数〔pdmod の expand rdarea 文〕 916
HiRDB ファイル内セグメント数〔pdmod の initialize rdarea 文〕 929
HiRDB ファイル内セグメント数〔pdreginit の create rdarea 文〕 1825
HiRDB ファイルの削除 212
HiRDB ファイル名〔pdinit の create rdarea 文〕 576
HiRDB ファイル名〔pdmod の create rdarea 文〕 908
HiRDB ファイル名〔pdmod の expand rdarea 文〕 915
HiRDB ファイル名〔pdmod の initialize rdarea 文〕 928
HiRDB ファイル名〔pdreginit の create rdarea 文〕 1825

I

idxfree〔pdload の option 文〕 702
idxfree〔pdrorg の option 文〕 1149
idxname 文〔pdpgbfon〕 1348
idxname 文〔pdreclaim〕 1321
idxname 文〔pdrorg〕 1127
idxwork 文〔pdload〕 673
idxwork 文〔pdrbal〕 1268
idxwork 文〔pdrorg〕 1129
INDEX SCAN〔pdvwopt〕 1691
INDEX SCAN TYPE プラグイン提供関数 1699
index 文〔pdload〕 672
index 文〔pdrbal〕 1267
index 文〔pdrorg〕 1126
initialize rdarea 文〔pdmod〕 921

J

JAR ファイルの操作 246
job〔pdload の option 文〕 702
job〔pdrorg の option 文〕 1150
JP1/NETM/Audit 用監査ログ出力ファイル 1833
JP1/NETM/Audit 用監査ログ出力ファイルの形式 1834
JP1/NETM/Audit 用監査ログ出力ユティリティ 1830
JP1/NETM/Audit 用監査ログ出力ユティリティ〔ユティリティー覧〕 45
JP1/NETM/Audit 用監査ログの出力項目 1834

K

KEY RANGE PARTIAL BROADCAST〔pdvwopt〕 1694
KEY RANGE〔pdvwopt〕 1692
KEY SCAN MERGE JOIN〔pdvwopt〕 1689
KEY SCAN〔pdvwopt〕 1691

L

lengover〔pdload の option 文〕 710
LIST SCAN MERGE JOIN〔pdvwopt〕 1689
LIST SCAN〔pdvwopt〕 1692
lobcolumn 文〔pdload〕 677
lobdata 文〔pdload〕 676
lobmid 文〔pdload〕 679
lobunld 文〔pdrorg〕 1131
LOB 作成種別〔pdload〕 644
LOB 中間ファイル〔pdload〕 679
LOB 中間ファイル名〔pdload の lobmid 文〕 681
LOB データ，及び LOB パラメタの記述方法〔-k v 又は-k d の場合〕 730
LOB データのアンロードファイルの種別〔pdrorg〕 1112
LOB データのアンロードファイル名〔pdrorg〕 1133
LOB 入力ファイル〔pdload〕 611, 676
LOB パラメタ 612
LOB 列 610
LOB 列がある表へのデータロード 610, 625
LOB 列構成基表 610
LOB 列の列名〔pdload の lobcolumn 文〕 678
L-KEY R-LIST MERGE JOIN〔pdvwopt〕 1690
L-KEY R-SORT MERGE JOIN〔pdvwopt〕 1690
L-LIST R-KEY MERGE JOIN〔pdvwopt〕 1690

L-LIST R-SORT MERGE JOIN〔pdvwopt〕 1690
L-SORT R-KEY MERGE JOIN〔pdvwopt〕 1690
L-SORT R-LIST MERGE JOIN〔pdvwopt〕 1690

M

maxreclen〔pdload の source 文〕 670
mode〔pdload の列名文〕 776
move rdarea 文〔pdmod〕 947
MULTI COLUMNS INDEX SCAN〔pdvwopt〕 1691
MULTI COLUMNS KEY SCAN〔pdvwopt〕 1691
MULTIDIM〔pdvwopt〕 1693

N

name〔pdrorg の index 文〕 1128
NESTED LOOPS JOIN〔pdvwopt〕 1690
NetBackup のポリシー名〔pdcopy〕 1765
NetBackup のポリシー名〔pdrstr〕 1798
norestart 947
nowait〔pdload の option 文〕 705
null_string〔pdload の option 文〕 708
null〔pdload の列名文〕 771
nullset〔pdload の array 文〕 686

O

option 文〔pddbst〕 1522
option 文〔pdload〕 699
option 文〔pdpgbfon〕 1349
option 文〔pdrbal〕 1276
option 文〔pdreclaim〕 1323
option 文〔pdrorg〕 1146
ORDER BY 処理方式の種類 1723
OR PLURAL INDEXES SCAN〔pdvwopt〕 1692

P

param〔pdload の srcuoc 文〕 682, 683
param〔pdload の列名文〕 779
PARTIAL BROADCAST〔pdvwopt〕 1695
pdacunlck 63
pdacunlck〔運用コマンド一覧〕 37
pdadmvr 65
pdadmvr〔運用コマンド一覧〕 37
pdaudatld 68
pdaudatld〔運用コマンド一覧〕 37
pdaudbegin 72
pdaudbegin〔運用コマンド一覧〕 37
pdaudend 74
pdaudend〔運用コマンド一覧〕 37
pdaudput 1830
pdaudput 用管理ファイル 1833
pdaudput〔ユティリティー一覧〕 45
pdaudrm 76
pdaudrm〔運用コマンド一覧〕 37
pdaudswap 78
pdaudswap〔運用コマンド一覧〕 37
pdbkupls 80
pdbkupls〔運用コマンド一覧〕 37
pdbufls 87
pdbufls〔運用コマンド一覧〕 37
pdbufmod 99
pdbufmod〔運用コマンド一覧〕 37
pdcancel 109
pdcancel〔運用コマンド一覧〕 37
pdcat 113
pdcat〔運用コマンド一覧〕 37
pdcat【-d sts】 114
pdcat【-d 省略】 117
pdcbl〔運用コマンド一覧〕 37
pdchgconf 122
pdchgconf〔運用コマンド一覧〕 37
pdchpathf 127
pdchpathf〔運用コマンド一覧〕 37
pdchpathn 130
pdchpathn〔運用コマンド一覧〕 38
pdchprc 132
pdchprc〔運用コマンド一覧〕 38
pdclibsync 136
pdclibsync〔運用コマンド一覧〕 38
pdclose 141

pdclose〔運用コマンド一覧〕 38
pdclttrc 144
pdclttrc〔運用コマンド一覧〕 38
pdcmt 148
pdcmt〔運用コマンド一覧〕 38
pdconfchk 150
pdconfchk〔運用コマンド一覧〕 38
pdconstck 1355
pdconstck〔ユティリティー一覧〕 45
pdcopy 1745
pdcopy〔ユティリティー一覧〕 45
pdcpp〔運用コマンド一覧〕 38
pdcspool 158
pdcspool〔運用コマンド一覧〕 38
pddbfrz 162
pddbfrz〔運用コマンド一覧〕 38
pddbls 165
pddbls〔運用コマンド一覧〕 38
pddbst 1502
pddbst〔ユティリティー一覧〕 45
pddef 594
pddefrev 175
pddefrev〔運用コマンド一覧〕 38
pddef〔ユティリティー一覧〕 44
pddelfw.bat 174
pddelfw.bat〔運用コマンド一覧〕 38
pdexp 1218
pdexp〔ユティリティー一覧〕 44
pdextfunc 179
pdextfunc〔運用コマンド一覧〕 38
pdfbkup 181
pdfbkup〔運用コマンド一覧〕 38
pdfchfs 184
pdfchfs〔運用コマンド一覧〕 38
pdffsck 188
pdffsck〔運用コマンド一覧〕 39
pdfgt 193
pdfgt〔運用コマンド一覧〕 39
pdfls 196
pdfls〔運用コマンド一覧〕 39
pdfmkfs 201
pdfmkfs〔運用コマンド一覧〕 39
pdfrm 212
pdfrm〔運用コマンド一覧〕 39
pdfrstr 214
pdfrstr〔運用コマンド一覧〕 39
pdfstatfs 217
pdfstatfs〔運用コマンド一覧〕 39
pdfzeroinit 223
pdfzeroinit〔運用コマンド一覧〕 39
pdgetcst 1608
pdgetcst〔ユティリティー一覧〕 45
pdhold 225
pdhold〔運用コマンド一覧〕 39
pdinfoget 235
pdinfoget〔運用コマンド一覧〕 39
pdinit 564
pdinit〔ユティリティー一覧〕 44
pdjarsync 246
pdjarsync〔運用コマンド一覧〕 39
pdkill 251
pdkill〔運用コマンド一覧〕 39
pdlistls 252
pdlistls〔運用コマンド一覧〕 39
pdload 607
pdload が使用するバッファ 648
pdload の実行監視時間〔pdload の option 文〕 707
pdload〔ユティリティー一覧〕 44
pdload 用アンロードファイル 757
pdlogadpf 256
pdlogadpf〔運用コマンド一覧〕 40
pdlogatul 259
pdlogatul〔運用コマンド一覧〕 40
pdlogchg 264
pdlogchg〔運用コマンド一覧〕 40
pdlogcls 268
pdlogcls〔運用コマンド一覧〕 40
pdloginit 270

pdloginit〔運用コマンド一覧〕 40
pdlogls 276
pdlogls〔運用コマンド一覧〕 40
pdlogls【-d spd】 277
pdlogls【-d sys】 280
pdlogopen 294
pdlogopen〔運用コマンド一覧〕 40
pdlogrm 296
pdlogrm〔運用コマンド一覧〕 40
pdlogswap 299
pdlogswap〔運用コマンド一覧〕 40
pdlogsync 302
pdlogsync〔運用コマンド一覧〕 40
pdlogucat 304
pdlogucat〔運用コマンド一覧〕 40
pdlogunld 308
pdlogunld〔運用コマンド一覧〕 40
pdls 313
pdls〔運用コマンド一覧〕 40
pdls【-d act】 316
pdls【-d aud】 318
pdls【-d ha】 321
pdls【-d lck】 329
pdls【-d mem】 338
pdls【-d prc】 341
pdls【-d rpc】 346
pdls【-d rpl】 351
pdls【-d scd】 356
pdls【-d stj】 359
pdls【-d sts】 363
pdls【-d svr】 367
pdls【-d trn】 371
pdls【-d ust】 379
pdmemdb 383
pdmemdb〔運用コマンド一覧〕 41
pdmod 890
pdmod の機能 891
pdmod〔ユティリティー覧〕 44
pdmstchk 390
pdmstchk〔運用コマンド一覧〕 41
pdntenv 394
pdntenv〔運用コマンド一覧〕 41
pdobils 406
pdobils〔運用コマンド一覧〕 41
pdobjconv 437
pdobjconv〔運用コマンド一覧〕 41
pdocb〔運用コマンド一覧〕 41
pdocc〔運用コマンド一覧〕 41
pdopen 441
pdopen〔運用コマンド一覧〕 41
pdosinfoget〔運用コマンド一覧〕 41
pdparaload 443
pdparaload〔運用コマンド一覧〕 41
pdpfresh 452
pdpfresh〔運用コマンド一覧〕 41
pdpgbfon 1333
pdpgbfon の監視時間〔pdpgbfon の option 文〕 1350
pdpgbfon の処理結果 1351
pdpgbfon〔ユティリティー覧〕 45
pdplgrgst 456
pdplgrgst〔運用コマンド一覧〕 41
pdprfed 460
pdprfed〔運用コマンド一覧〕 41
pdprflevel 463
pdprflevel〔運用コマンド一覧〕 41
pdprgcopy 465
pdprgcopy〔運用コマンド一覧〕 41
pdprgrenew 467
pdprgrenew〔運用コマンド一覧〕 41
pdrbal 1244
pdrbal が使用するバッファ 1262
pdrbal 実行時間〔pdrbal の execstop 文〕 1271
pdrbal〔ユティリティー覧〕 44
pdrbk 470
pdrbk〔運用コマンド一覧〕 41
pdrdrefls 472
pdrdrefls〔運用コマンド一覧〕 42

pdreclaim 1285
pdreclaim の監視時間〔pdreclaim の option 文〕 1323
pdreclaim の処理結果 1325
pdreclaim〔ユティリティー一覧〕 44
pdreginit 1818
pdreginit〔ユティリティー一覧〕 45
pdrels 480
pdrels〔運用コマンド一覧〕 42
pdrorg 972
pdrorg が UOC に渡すデータの格納方法〔pdrorg の unlduoc 文〕 1136
pdrorg が使用するバッファ 1109
pdrorg 出力バイナリ形式 609
pdrorg 出力バイナリ形式ファイル 757
pdrorg の機能 973
pdrorg の機能ごとのオプション指定可否 1088
pdrorg の実行監視時間〔pdrorg の option 文〕 1152
pdrorg〔ユティリティー一覧〕 44
pdrorg 用アンロードファイル 757
pdrplstart 483
pdrplstart〔運用コマンド一覧〕 42
pdrplstop 485
pdrplstop〔運用コマンド一覧〕 42
pdrstr 1783
pdrstr〔ユティリティー一覧〕 45
pdsetacl 488
pdsetacl〔運用コマンド一覧〕 42
pdsetfw.bat 493
pdsetfw.bat〔運用コマンド一覧〕 42
pdstart 495
pdstart〔運用コマンド一覧〕 42
pdstbegin 502
pdstbegin〔運用コマンド一覧〕 42
pdstedit 1386
pdstedit〔ユティリティー一覧〕 45
pdstend 515
pdstend〔運用コマンド一覧〕 42
pdstjswap 520
pdstjswap〔運用コマンド一覧〕 42
pdstjsync 522
pdstjsync〔運用コマンド一覧〕 42
pdstop 526
pdstop〔運用コマンド一覧〕 42
pdstscls 532
pdstscls〔運用コマンド一覧〕 42
pdstsinit 535
pdstsinit〔運用コマンド一覧〕 42
pdstsopen 541
pdstsopen〔運用コマンド一覧〕 42
pdstsrm 544
pdstsrm〔運用コマンド一覧〕 42
pdstsswap 547
pdstsswap〔運用コマンド一覧〕 42
pdsvhostname 549
pdsvhostname〔運用コマンド一覧〕 42
pdtrndec 551
pdtrndec〔運用コマンド一覧〕 42
pdtrnqing 555
pdtrnqing〔運用コマンド一覧〕 42
pdvrup 563
pdvrup〔運用コマンド一覧〕 43
pdvwopt 1646
pdvwopt〔ユティリティー一覧〕 45
pending〔pdload の constraint 文〕 695
pending〔pdrorg の constraint 文〕 1142
PLUGIN INDEX SCAN〔pdvwopt〕 1691
PLUGIN KEY SCAN〔pdvwopt〕 1691
predict 文〔pddbst〕 1520
PRF トレース取得レベルの表示及び変更 463
PRF トレース情報の編集出力 460

R

rdaomit〔pdload〕 636
RD エリア単位の状態解析 1503, 1524, 1551
RD エリアの移動 942
RD エリアのオープン 441

RD エリアのオープン契機〔pdinit の create rdarea 文〕 572
RD エリアのオープン契機〔pdmod の alter rdarea 文〕 957
RD エリアのオープン契機〔pdmod の create rdarea 文〕 904
RD エリアのオープン契機〔pdmod の initialize rdarea 文〕 921
RD エリアの拡張 911
RD エリアのクローズ 141
RD エリアの再初期化 917
RD エリアの削除 939
RD エリアの種類〔pdinit の create rdarea 文〕 570
RD エリアの種類〔pdmod の create rdarea 文〕 902
RD エリアの種類〔pdreginit の create rdarea 文〕 1823
RD エリアの状態遷移 1879
RD エリアの状態による UAP の実行可否 1881
RD エリアの状態によるユティリティの実行可否 1881
RD エリアの状態表示 165
RD エリアの属性変更 955
RD エリアの追加 895
RD エリアの閉塞 225
RD エリアの閉塞解除 480
RD エリア名〔pdcopy〕 1767
RD エリア名〔pddbst〕 1512
RD エリア名〔pdinit の create rdarea 文〕 569
RD エリア名〔pdload の index 文〕 672
RD エリア名〔pdload の lobmid 文〕 680
RD エリア名〔pdload の source 文〕 665
RD エリア名〔pdmod の alter HiRDB mode to parallel 文〕 963
RD エリア名〔pdmod の alter rdarea 文〕 956
RD エリア名〔pdmod の create audit table 文〕 969
RD エリア名〔pdmod の create rdarea 文〕 901
RD エリア名〔pdmod の expand rdarea 文〕 914
RD エリア名〔pdmod の initialize rdarea 文〕 921
RD エリア名〔pdmod の move rdarea 文〕 948
RD エリア名〔pdmod の remove rdarea 文〕 940
RD エリア名〔pdpgbfon〕 1346
RD エリア名〔pdpgbfon の idxname 文〕 1349
RD エリア名〔pdrbal の index 文〕 1267
RD エリア名〔pdreclaim〕 1308
RD エリア名〔pdreclaim の idxname 文〕 1322
RD エリア名〔pdreginit の create rdarea 文〕 1822
RD エリア名〔pdrorg〕 1095
RD エリア名〔pdrorg の index 文〕 1127
RD エリア名〔pdrstr〕 1810
ref_pending〔pdload の constraint 文〕 695
ref_pending〔pdrorg の constraint 文〕 1142
reld_func 文〔pdrbal〕 1273
reld_func 文〔pdrorg〕 1140
remove rdarea 文〔pdmod〕 940
report 文〔pdrbal〕 1274
ROWID FETCH〔pdvwopt〕 1692
R-LIST NESTED LOOPS JOIN〔pdvwopt〕 1690

S

SCAN TYPE プラグイン提供関数 1698
SELECT-APSL〔pdvwopt〕 1690, 1692, 1695
seq_range〔pdload の option 文〕 713
sequence〔pdload の列名文〕 778
server〔pdrorg の index 文〕 1128
SORT MERGE JOIN〔pdvwopt〕 1689
sort 文 1269
sort 文〔pdload〕 674
sort 文〔pdrorg〕 1130
source 文〔pdload〕 665
spacelvl〔pdload の option 文〕 700
spacelvl〔pdrorg の option 文〕 1147
SQL オブジェクト移行情報 438
SQL オブジェクト実行に関する統計情報 1491
SQL オブジェクト転送に関する統計情報 1493
SQL オブジェクト用バッファの統計情報表示 406
SQL 静的最適化に関する統計情報 1487
SQL 動的最適化に関する統計情報 1490
SQL トレースの動的取得 144
SQL に関する統計情報 1435, 1477
SQL 文の終了指定〔pddef〕 596

SQL 文の履歴に関する統計情報　1494
src_work 文〔pdload〕　689
srcendian〔pdload の option 文〕　711
srcuoc 文〔pdload〕　682

T

TABLE SCAN〔pdvwopt〕　1691
tblfree〔pdload の option 文〕　702
tblfree〔pdrorg の option 文〕　1149
tblname 文〔pdrorg〕　1136
tblstatus〔pdrorg の option 文〕　1153
type〔pdrorg の reld_func 文〕　1140
type〔pdrorg の unld_func 文〕　1139

U

UAP 処理の強制終了　109
UAP に関する統計情報　1423, 1467
UAP の識別情報　343, 375
UAP 名〔pdstedit〕　1394
UNICAST〔pdvwopt〕　1694
unld_func 文〔pdrbal〕　1271
unld_func 文〔pdrorg〕　1138
unldenq〔pdrorg の option 文〕　1154
unlduoc 文〔pdrorg〕　1135
unload 文〔pdrorg〕　1122
UNTIL DISCONNECT 指定の排他保持状態　317, 331
UOC　667, 682, 804
uoc〔pdload の source 文〕　667
uoc〔pdrorg の unload 文〕　1124
UOC インタフェース領域の内容　807
UOC 情報〔pdrorg の unload 文〕　1125
UOC の使用例〔pdload〕　813
UOC の使用例〔pdrorg〕　1190
UOC を利用した再編成　977, 1163
UOC を利用したデータロード　804

V

VV 形式　686

W

whitespace〔pdload の option 文〕　713
Windows ファイアウォールの例外リスト削除　174
Windows ファイアウォールの例外リスト登録　493
without lock table〔pdmod の initialize rdarea 文〕　925
with reconstruction〔pdmod の initialize rdarea 文〕　921
workdir 文〔pddbst〕　1521

あ

空きありセグメント　1286
空きセグメント　1286
空きセグメント解放のコミット間隔〔pdreclaim〕　1308
空きページ解放のコミット間隔〔pdreclaim〕　1308
空きページ解放ユティリティ　1285
空きページ解放ユティリティの排他制御モード〔共用表の場合〕　1873
空きページ解放ユティリティ〔ユティリティー覧〕　44
アクセスパス情報ファイル〔pdvwopt〕　1648
アクセスパス情報ファイル取得時の注意事項　1651
アクセスパスの種類　1695
アクセスパス表示ユティリティ　1646
アクセスパス表示ユティリティの実行手順　1650
アクセスパス表示ユティリティ〔ユティリティー覧〕45
アプリケーション向けデータ更新〔UOC を利用した再編成〕　1164
暗黙的グループ分け処理　1696
アンロード順指定〔pdrorg〕　1113
アンロードデータファイルの種別〔pdrorg〕　1112
アンロードデータファイル名〔pdrorg〕　1123
アンロード統計ログファイル　1388
アンロードレスシステムログ運用　1749, 1760, 1785, 1789, 1799
アンロードログファイル　1388
アンロードログファイルの情報表示　304
アンロードログファイル名〔pdrstr〕　1799

い

一括出力用ローカルバッファ面数〔pdload〕 647
一括入出力用ローカルバッファ面数〔pdrbal〕 1261
一括入出力用ローカルバッファ面数〔pdrorg〕 1108
一括入力最大ページ数 92
インターバル解析〔状態解析結果蓄積機能（pddbst)〕 1601
インデクス一括作成モード〔pdload〕 638
インデクス一括作成モード〔pdrbal〕 1257
インデクス一括作成モード〔pdrorg〕 1098
インデクスが定義されている表へのデータロード 623
インデクス検索〔pdvwopt でのチューニング〕 1737
インデクス更新モード〔pdload〕 638
インデクス更新モード〔pdrbal〕 1258
インデクス更新モード〔pdrorg〕 1099
インデクス作成上の注意事項〔pdload〕 639
インデクス作成上の注意事項〔pdrorg〕 1099
インデクス作成方法〔pdload〕 637
インデクス作成方法〔pdrbal〕 1257
インデクス作成方法〔pdrorg〕 1098
インデクス識別子〔pddbst〕 1516
インデクス識別子〔pdload の index 文〕 672
インデクス識別子〔pdpgbfon の idxname 文〕 1349
インデクス識別子〔pdrbal の index 文〕 1267
インデクス識別子〔pdreclaim の idxname 文〕 1322
インデクス識別子〔pdrorg の index 文〕 1127, 1128
インデクス情報出力モード〔pdload〕 638
インデクス情報出力モード〔pdrorg〕 1098
インデクス情報出力抑止モード〔pdload〕 639
インデクス情報ファイル〔pdload〕 672
インデクス情報ファイル〔pdrbal〕 1267
インデクス情報ファイル〔pdrorg〕 1126
インデクス情報ファイル名〔pdload の index 文〕 673
インデクス情報ファイル名〔pdrbal の index 文〕 1268
インデクス情報ファイル名〔pdrorg の index 文〕 1127
インデクス単位の状態解析 1504, 1561
インデクス定義〔pdvwopt でのチューニング〕 1729
インデクスに関する統計情報 1447
インデクスの一括作成 984, 1063
インデクスの再作成 984, 1071
インデクスの再編成 985, 1081
インデクスを使用しない検索 1691
インフォメーションメッセージ出力抑止レベル〔pdload〕 656
インフォメーションメッセージ出力抑止レベル〔pdrbal〕 1264
インフォメーションメッセージ出力抑止レベル〔pdrorg〕 1116
インメモリデータ処理 383
インメモリデータ処理を使用している場合〔バックアップの取得〕 1750

う

埋め字データ〔pdload の列名文〕 777
運用コマンド 62
運用コマンド，ユティリティでの RD エリアの指定 51
運用コマンド一覧 37
運用コマンドの実行者 37–43
運用コマンドの排他制御モード 1859
運用方法〔pdpgbfon〕 1335

え

エラー情報ファイル〔pdload〕 758
エラー情報ファイル名〔pdload の lobmid 文〕 681
エラー情報ファイル名〔pdload の source 文〕 667
エラーデータファイル作成用バッファサイズ〔pdload の source 文〕 669
エラーデータファイル名〔pdload の source 文〕 668

お

オプション 34
オプションを指定するときの注意事項 35
オペレーションコード 1479

か

改竄防止表〔pdrorg を実行する場合の注意事項〕 1196
開始時刻〔pdstedit〕 1393

解析項目別情報　1586
回復開始時刻〔pdrstr〕　1810
回復終了時刻〔pdrstr〕　1810
会話型 SQL 実行ユティリティ　888
会話型 SQL 実行ユティリティ〔ユティリティ一覧〕　44
書き込みバッファサイズ〔pdrstr〕　1805
拡張 DAT 形式　609, 687
拡張機能の内容〔pdload の extdat 文〕　689
拡張システム定義スカラ関数の定義，削除　179
格納行数の表示〔pddbst〕　1516
囲み文字　687
囲み文字〔pdload の extdat 文〕　688
可変長データ型の場合の配列データ　746
監査証跡データ登録 UOC　682
監査証跡データ登録 UOC 用パラメタ〔pdload の srcuoc 文〕　683
監査証跡の取得開始　72
監査証跡表の作成　968
監査証跡表の自動データロード機能の制御　68
監査証跡ファイルの状態表示　318
監査人の登録　965
関数名〔pdload の srcuoc 文〕　682
関数名〔pdrbal の reld_func 文〕　1273
関数名〔pdrbal の unld_func 文〕　1272
関数名〔pdrorg の reld_func 文〕　1140
関数名〔pdrorg の unld_func 文〕　1139
関数名〔pdrorg の unlduoc 文〕　1135
関連する RD エリア　1788
関連する RD エリアの情報の表示　472

き

キー条件　1717
キーレンジ転送　1692
キーレンジ部分ブロードキャスト転送　1694
基準値定義ファイル〔pddbst〕　1594
行識別子を使用した検索　1692
行データ出力形式〔pdrorg の array 文〕　1138
共有モード〔pdrbal〕　1247

く

空白変換レベル〔pdgetcst〕　1616
空白変換レベル〔pdload の option 文〕　700
空白変換レベル〔pdrorg の option 文〕　1147
クライアント環境定義の設定〔pddef を実行する場合〕　595
クラスタキーの格納状態解析　1505, 1577
クラスタキーの並び解析　1505
クラスタリングデータページの格納状態解析　1505, 1577
クラスタリングデータページの並び解析　1506
繰返し列　684, 1137
繰返し列を含む表の配列データ形式　742
グループ分け処理方式の種類　1696
グローバルバッファ常駐化ユティリティ　1333
グローバルバッファ常駐化ユティリティの排他制御モード　1874
グローバルバッファ常駐化ユティリティの排他制御モード〔共用表の場合〕　1875
グローバルバッファ常駐化ユティリティ〔ユティリティ一覧〕　45
グローバルバッファ情報の表示　87
グローバルバッファの動的変更　99
グローバルバッファのヒット率　93
グローバルバッファプールに関する統計情報　1438, 1482
グローバルバッファ名〔pdmod の create rdarea 文〕　901
グローバルバッファ名〔pdreginit の create rdarea 文〕　1822

け

経過メッセージ出力間隔〔pdload〕　656
経過メッセージ出力間隔〔pdrbal〕　1264
経過メッセージ出力間隔〔pdrorg〕　1116
継続行の扱い〔pddef〕　598
結合検索〔pdvwopt でのチューニング〕　1743
結合処理情報　1662
結合方法の種類　1689
検索条件定義ファイル　1834

検索条件定義ファイル〔pdaudput〕 1848
検索による最適化情報の収集 1609, 1640
検索方法の種類 1691
検査制約の整合性チェック 1359, 1361
検査制約の整合性チェックの処理結果 1382
検査制約の整合性チェック方法 1361
検査保留状態情報 1383
検査保留状態の強制解除〔検査保留状態の強制変更機能〕 1367, 1369
検査保留状態の強制設定〔検査保留状態の強制変更機能〕 1367, 1369
検査保留状態の強制変更機能 1367
検査保留状態の強制変更機能（検査保留状態の強制解除） 1376
検査保留状態の強制変更機能（検査保留状態の強制設定） 1376
検査保留状態の強制変更機能の実行単位 1368
検査保留状態の設定可否 1892
検査保留状態の遷移 1364
検査保留状態の遷移〔RD エリア〕 1363
検査保留状態の遷移〔制約〕 1362
検査保留状態の遷移〔制約単位で実行した場合〕 1364
検査保留状態の遷移〔表〕 1362
検査保留状態の遷移〔表単位で実行した場合〕 1363
現用の監査証跡ファイルのスワップ 78

こ

更新可能バックアップ閉塞 228
更新可能バックアップ閉塞（WAIT モード） 228
更新可能モード〔pdcopy〕 1757
更新凍結状態 162
更新前ログ取得モード〔pdload〕 642
更新前ログ取得モード〔pdrorg〕 1096
コストベースの最適化 1609
固定長データ型の場合の配列データ 745
固定長データ形式 609, 733
コマンド一覧 1853
コマンド実行時の RD エリアの状態 1879
コマンド実行中の割り込みによる強制終了 51
コマンドの記述形式 33
コマンドの実行形態 32
コマンドの同時接続数 1903
コマンドの入力方法 32
コマンドの別名での利用 35
コマンド引数 35
コミット単位〔pdrbal〕 1260
コンストラクタ関数〔pdrbal〕 1273
コンストラクタ関数〔pdrorg〕 1140
コンストラクタパラメタ逆生成関数〔pdrbal〕 1271
コンストラクタパラメタ逆生成関数〔pdrorg〕 1138

さ

サーチ条件 1699
サーチ条件種別 1700
サーバごとの統計情報出力種別の指定有無の表示 359
サーバの開始 495
サーバの共用メモリの状態表示 338
サーバの終了 526
サーバの状態表示 367
サーバのスケジュールの状態表示 356
サーバのステータス情報 369
サーバの通信制御情報の表示 346
サーバのトランザクションの状態表示 371
サーバの排他制御の状態表示 329
サーバのプロセスの状態表示 341
サーバのホスト名表示 549
サーバプロセスの起動本数変更 132
サーバプロセスのリフレッシュ 452
サーバ名〔pdcopy〕 1767
サーバ名〔pdinit の create rdarea 文〕 571
サーバ名〔pdload の idxwork 文〕 674
サーバ名〔pdload の lobmid 文〕 680
サーバ名〔pdload の sort 文〕 675
サーバ名〔pdload の source 文〕 666
サーバ名〔pdmod の alter HiRDB mode to parallel 文〕 963
サーバ名〔pdmod の create rdarea 文〕 904
サーバ名〔pdmod の move rdarea 文〕 948
サーバ名〔pdpgbfon〕 1347

サーバ名〔pdpgbfon の idxname 文〕 1349
サーバ名〔pdrbal の idxwork 文〕 1268
サーバ名〔pdrbal の sort 文〕 1270
サーバ名〔pdreclaim〕 1320
サーバ名〔pdreclaim の idxname 文〕 1322
サーバ名〔pdrorg の idxwork 文〕 1129
サーバ名〔pdrorg の index 文〕 1128
サーバ名〔pdrorg の lobunld 文〕 1132
サーバ名〔pdrorg の sort 文〕 1131
サーバ名〔pdrorg の unload 文〕 1122
サーバ名〔pdrstr〕 1809
サーバ名〔pdstedit〕 1394
再初期化後のインデクスの状態 929
再初期化後の表の状態 929
最大同時実行数 816
最大要素数 684
最大リスト登録数〔pdinit の create rdarea 文〕 574
最大リスト登録数〔pdmod の create rdarea 文〕 906
最大リスト登録数〔pdmod の initialize rdarea 文〕 924
最適化情報 1618
最適化情報収集ユティリティ 1608
最適化情報収集ユティリティが取得する最適化情報 1618
最適化情報収集ユティリティの排他制御モード 1876
最適化情報収集ユティリティ〔ユティリティー覧〕 45
最適化情報収集レベル 1622
最適化情報収集レベル〔pdgetcst〕 1616
最適化情報パラメタファイル 1624
最適化情報パラメタファイルによって登録する最適化情報 1618
最適化情報パラメタファイルによる最適化情報の登録 1610, 1642
最適化情報パラメタファイルへの最適化情報の退避 1611
最適化情報パラメタファイル名〔pdgetcst〕 1616
再編成時期予測機能 1507
再編成時期予測機能の実行結果 1581
作業表の検索 1692
作業用一時ファイルの削除 158
作成されるインデクス情報ファイルの名称〔pdrbal の idxwork 文〕 1269
作成モード〔pdload〕 635
差分バックアップ管理ファイルの容量〔pdcopy〕1770
差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名〔pdcopy〕 1769
差分バックアップ管理ファイルを格納する HiRDB ファイルシステム領域名〔pdrstr〕 1805
差分バックアップ機能 1750
差分バックアップ機能使用時のバックアップ種別〔pdcopy〕 1769
差分バックアップグループ名〔pdcopy〕 1768
差分バックアップグループ名〔pdrstr〕 1804
差分バックアップの履歴情報ファイル名〔pdcopy〕 1769
参照可能バックアップ閉塞 227
参照可能バックアップ閉塞（更新 WAIT モード） 227
参照可能モード〔pdcopy〕 1756
参照制約の整合性チェック 1359, 1360
参照制約の整合性チェックの処理結果 1381
参照制約の整合性チェック方法 1360
参照制約を定義した表の搬出入の規則について〔pdexp〕 1234
参照・更新不可能モード〔pdcopy〕 1756

し

時間間隔〔pdstedit〕 1392
システム構成変更コマンド 122
システム単位でのバックアップ〔pdcopy〕 1766
システム定義のチェック 150
システムの稼働に関する統計情報 1408, 1454
システムログファイルの情報表示 280
システムログファイルのファイルグループの状態 283
システムログファイルの物理ファイルの状態 287
システムログファイルの要素ファイルの状態 285
実行可否チェック間隔〔pdrbal の option 文〕 1277
実行監視時間〔pdexp〕 1228
実行監視時間〔pdgetcst〕 1617
実行監視時間〔pdinit〕 567
実行監視時間〔pdmod〕 893

実行監視時間〔pdrbal〕 1265
実行監視時間〔pdreginit〕 1821
実行監視時間〔pdrstr〕 1805
実行監視時間〔pdcopy〕 1763
実行結果リスト〔pdmod【RD エリアの移動】〕 949
実行結果リスト〔pdmod【RD エリアの再初期化】〕 934
実表検索処理情報 1668
実要素数 684
指定単位取得方式〔順序番号の取得方式〕 615
指定できる制御文の個数〔pdload〕 663
指定要素数 684
自動採番機能を使用したデータロード 613
自動ログアンロード機能 259
自動ログアンロード機能の制御 259
集合演算処理情報 1656
修正版 HiRDB の入れ替え 467
修正版 HiRDB のコピー 465
終了時刻〔pdstedit〕 1393
出力行数〔pdload の source 文〕 668
出力形式 411
出力形式〔pdadmvr〕 66
出力形式〔pdaudatld〕 70
出力形式〔pdbufls〕 91
出力形式〔pdcat【-d sts】〕 115
出力形式〔pdcat【-d 省略】〕 120
出力形式〔pdchprc〕 135
出力形式〔pdclibsync〕 139
出力形式〔pdconfchk〕 153
出力形式〔pddbls〕 168
出力形式〔pdfls〕 197
出力形式〔pdfstatfs〕 219
出力形式〔pdjarsync〕 249
出力形式〔pdlistls〕 253
出力形式〔pdlogatul〕 262
出力形式〔pdlogls【-d spd】〕 278
出力形式〔pdlogls【-d sys】〕 282
出力形式〔pdls【-d act】〕 317
出力形式〔pdls【-d aud】〕 319
出力形式〔pdls【-d ha】〕 322
出力形式〔pdls【-d lck】〕 330
出力形式〔pdls【-d mem】〕 338
出力形式〔pdls【-d rpc】〕 347
出力形式〔pdls【-d rpl】〕 352
出力形式〔pdls【-d scd】〕 357
出力形式〔pdls【-d stj】〕 360
出力形式〔pdls【-d sts】〕 364
出力形式〔pdls【-d svr】〕 368
出力形式〔pdls【-d trn】〕 372
出力形式〔pdls【-d ust】〕 380
出力形式〔pdls【-d prc】〕 342
出力形式〔pdntenv〕 401
出力形式〔pdobjconv〕 438
出力形式〔pdrbal〕 1275
出力形式〔pdrdrefls〕 475
出力形式〔pdsvhostname〕 549
出力形式〔pdtrndec〕 553
出力結果ファイル名〔pdgetcst〕 1616
出力ディレクトリ〔pdaudput〕 1848
順序番号の格納方式 615
順序番号の取得方式 615
障害情報の取得と容量見積もり 235
障害発生時にデータベースを最新の状態に回復しない場合の運用〔再編成時期予測機能〕 1509
状態解析結果蓄積機能 1506
状態解析結果蓄積情報のリセット 1509
使用中空きページ 1286
使用中セグメント 1286
使用中ページ 1286
使用中満杯ページ 1286
使用例〔pdbufmod〕 108
使用例〔pdconfchk〕 156
使用例〔pdconstck〕 1371
使用例〔pdcopy〕 1751
使用例〔pddbst〕 1607
使用例〔pddef〕 604
使用例〔pdexp〕 1240
使用例〔pdfmkfs〕 211

使用例〔pdgetcst〕 1645
使用例〔pdinit〕 584
使用例〔pdload〕 837
使用例〔pdlogadpf〕 257
使用例〔pdloginit〕 273
使用例〔pdmod【HiRDB/シングルサーバ構成からHiRDB/パラレルサーバ構成への RD エリアの変更】〕 961
使用例〔pdmod【RD エリアの移動】〕 947
使用例〔pdmod【RD エリアの拡張】〕 912
使用例〔pdmod【RD エリアの再初期化】〕 919
使用例〔pdmod【RD エリアの削除】〕 939
使用例〔pdmod【RD エリアの属性変更】〕 955
使用例〔pdmod【RD エリアの追加】〕 896
使用例〔pdmod【監査証跡表の作成】〕 968
使用例〔pdmod【監査人の登録】〕 966
使用例〔pdmod【ディクショナリ表の属性定義変更】〕 952
使用例〔pdpgbfon〕 1340
使用例〔pdrbal〕 1250
使用例〔pdrdrefls〕 476
使用例〔pdreclaim〕 1301
使用例〔pdreginit〕 1828
使用例〔pdrorg【インデクスの一括作成】〕 1063
使用例〔pdrorg【インデクスの再作成】〕 1071
使用例〔pdrorg【インデクスの再編成】〕 1081
使用例〔pdrorg【ディクショナリ表の再編成】〕 1059
使用例〔pdrorg【表のアンロード】〕 1006
使用例〔pdrorg【表の再編成】〕 989
使用例〔pdrorg【表へのリロード】〕 1042
使用例〔pdrstr〕 1791
使用例〔pdstedit〕 1499
使用例〔pdstsinit〕 538
処理結果 1380
処理結果出力ファイル名〔pdcopy〕 1758
処理結果出力ファイル名〔pdrstr〕 1801
処理結果ファイル〔pdrbal〕 1274
処理結果ファイル名〔pdconstck〕 1377
処理結果ファイル名〔pdrbal の report 文〕 1274
処理種別〔pdreginit〕 1820
処理性能情報ファイル〔pdload〕 798
処理性能情報ファイル〔pdrorg〕 1162
処理対象〔pdreclaim〕 1308
処理対象〔pdrorg〕 1091
処理内容〔pdrorg〕 1089
シンクポイントダンプの取得 302
シンクポイントダンプファイルの情報表示 277

す

スキーマ単位の再編成 1091
スキーマ名〔pdexp〕 1226, 1227
スキップデータ文〔pdload〕 787
ステータスファイルのオープン 541
ステータスファイルのクローズ 532
ステータスファイルの削除 544
ステータスファイルの状態表示 363
ステータスファイルの初期設定 535
ステータスファイルのスワップ 547
ステータスファイルの内容表示 114
ステータスファイルの物理ファイルの状態 365
ステータスファイルの論理ファイルの状態 365
ストアドプロシジャ情報の搬出入 1222
ストアドプロシジャ情報の搬出入についての規則〔pdexp〕 1230
スレッド間ロック待ち時間 507

せ

制御情報ファイル〔pdload〕 609
制御情報ファイル名〔pdload〕 662
制御情報ファイル名〔pdpgbfon〕 1348
制御情報ファイル名〔pdrbal〕 1265
制御情報ファイル名〔pdreclaim〕 1321
制御情報ファイル名〔pdrorg〕 1118
制御文〔pdinit〕 568
制御文で指定しない場合のファイル出力先ディレクトリ 829
制御文とオプションの関係〔pdrorg〕 1121
制御文ファイル名〔pdcopy〕 1764

制御文ファイル名〔pdexp〕 1225
制御文ファイル名〔pdinit〕 566
制御文ファイル名〔pdmod〕 893
制御文ファイル名〔pdreginit〕 1820
制御文ファイル名〔pdrstr〕 1806
制御文ファイル名〔pdstedit〕 1397
制御文を指定するときの注意事項 35
整合性チェック機能 1359, 1376
整合性チェック機能の実行単位 1359
整合性チェック結果による検査保留状態の遷移 1362
整合性チェックの実行有無 1365
整合性チェックの実行有無〔制約単位で実行する場合〕 1367
整合性チェックの実行有無〔表単位で実行した場合〕 1365
整合性チェックの実行監視時間〔pdconstck〕 1378
整合性チェックの順番 1365
整合性チェックの順番〔制約単位で実行する場合〕 1367
整合性チェックの順番〔表単位で実行した場合〕 1365
整合性チェックの処理結果 1380
整合性チェックユティリティ 1355
整合性チェックユティリティの排他制御モード 1877
整合性チェックユティリティ〔ユティリティー覧〕 45
制約違反キー値の出力数上限値〔pdconstck〕 1378
制約単位〔検査保留状態の強制変更機能の実行単位〕 1369
制約単位〔整合性チェック機能の実行単位〕 1359, 1360
制約名〔pdconstck〕 1377
セグメントサイズ〔pdinit の create rdarea 文〕 574
セグメントサイズ〔pdmod の create rdarea 文〕 906
セグメントサイズ〔pdmod の initialize rdarea 文〕 924
セグメントサイズ〔pdreginit の create rdarea 文〕 1823
セパレータ文字〔pdload〕 652
セパレータ文字〔pdrorg〕 1101
全 RD エリアの回復〔pdrstr〕 1808
全数一括取得方式〔順序番号の取得方式〕 615
占有モード〔pdrbal〕 1247

そ

増分セグメント数〔pdinit の create rdarea 文〕 574
増分セグメント数〔pdmod の alter rdarea 文〕 958
増分セグメント数〔pdmod の create rdarea 文〕 906
増分セグメント数〔pdmod の expand rdarea 文〕 914
増分セグメント数〔pdmod の initialize rdarea 文〕 924
増分セグメント数〔pdreginit の create rdarea 文〕 1823
増分値〔pdcopy〕 1770
ソート用ワークディレクトリ〔pdload〕 674
ソート用ワークディレクトリ〔pdrbal〕 1269
ソート用ワークディレクトリ〔pdrorg〕 1130
ソート用ワークディレクトリ名〔pdrstr〕 1802
ソート用ワークバッファサイズ〔pdrstr〕 1803
ソート用ワークファイル〔pdload〕 674
ソート用ワークファイル〔pdrorg〕 1130

た

対象資源〔pdpgbfon〕 1346
対象資源〔pdreclaim〕 1307
ダイナミックリンクライブラリの作成 815
代表的なデータロードの例 623

ち

抽象データ型がある表〔pdrorg を実行する場合の注意事項〕 1193
抽象データ型名〔pdrbal の reld_func 文〕 1273
抽象データ型名〔pdrbal の unld_func 文〕 1272
抽象データ型名〔pdrorg の reld_func 文〕 1140
抽象データ型名〔pdrorg の unld_func 文〕 1139
抽象データ型列がある表に対する pdrorg の実行可否 987
チューニングの考え方〔pdvwopt〕 1726
直積 1691

つ

追加データ文〔pdload〕 790
追加モード〔pdload〕 635

て

定義系 SQL 出力ファイル名〔pdexp〕 1228
定義系 SQL の記述〔pddef〕 597
定義系 SQL の生成 175, 1222
定義系 SQL の生成についての規則〔pdexp〕 1232
ディクショナリ搬出入ユティリティ 1218
ディクショナリ搬出入ユティリティ〔ユティリティー一覧〕 44
ディクショナリ表の混在文字データの使用〔pdmod の alter system 文〕 954
ディクショナリ表の再編成 983, 1059
ディクショナリ表の参照権限の設定〔pdmod の alter system 文〕 953
ディクショナリ表の属性定義変更 952
ディレクトリ名〔pdload の idxwork 文〕 674
ディレクトリ名〔pdload の sort 文〕 675
ディレクトリ名〔pdrbal の idxwork 文〕 1269
ディレクトリ名〔pdrbal の sort 文〕 1270
ディレクトリ名〔pdrorg の idxwork 文〕 1130
ディレクトリ名〔pdrorg の sort 文〕 1131
データ型〔pdload のスキップデータ文〕 788
データ型〔pdload の列名文〕 768
データベース回復ユティリティ 1783
データベース回復ユティリティの処理結果リスト 1815
データベース回復ユティリティの排他制御モード 1877
データベース回復ユティリティ〔ユティリティー一覧〕 45
データベース構成変更ユティリティ 890
データベース構成変更ユティリティの排他制御モード 1863
データベース構成変更ユティリティ〔ユティリティー一覧〕 44
データベース再編成ユティリティ 972
データベース再編成ユティリティの排他制御モード 1869
データベース再編成ユティリティの排他制御モード〔共用表の場合〕 1871
データベース再編成ユティリティ〔ユティリティー一覧〕 44
データベース作成ユティリティ 607
データベース作成ユティリティの入力用ファイルの形式 1026
データベース作成ユティリティの排他制御モード 1861
データベース作成ユティリティの排他制御モード〔共用表の場合〕 1862
データベース作成ユティリティ〔ユティリティー一覧〕 44
データベース状態解析ユティリティ 1502
データベース状態解析ユティリティの排他制御モード 1875
データベース状態解析ユティリティ〔ユティリティー一覧〕 45
データベース初期設定ユティリティ 564
データベース初期設定ユティリティ〔ユティリティー一覧〕 44
データベース操作に関する HiRDB ファイルの統計情報 1443, 1485
データベース定義ユティリティ 594
データベース定義ユティリティ〔ユティリティー一覧〕 44
データベースの回復 1784
データベースの回復単位 1786
データベースの静止化 226
データベースの入出力に関する統計情報 1496
データベース複写ユティリティ 1745
データベース複写ユティリティの処理結果リスト 1774
データベース複写ユティリティの排他制御モード 1876
データベース複写ユティリティ〔ユティリティー一覧〕 45
データ未完状態 1197
データロード 608
テープ装置 49
テープ装置アクセス機能 50
デッドロックプライオリティ値 332
デファードライト処理に関する統計情報 1445, 1486
デファードライトトリガ時の更新ページ出力比率 92
デファードライトトリガ時の出力契機となる更新バッファ数 94
転送方法の種類 1692

と

問合せ処理情報　1659
同期点指定の再編成　977, 1150
同期点指定のデータロード　617, 702
同期点指定のリロード　983, 1150
統計解析ユティリティ　1386
統計解析ユティリティが編集する統計情報　1389
統計解析ユティリティ〔ユティリティ一覧〕　45
統計情報出力種別の指定有無　359
統計情報の出力開始　502
統計情報の出力契機　512
統計情報の出力形式　1403
統計情報の出力停止　515
統計情報の種別　503, 515
統計情報の詳細　1403
統計入力アンロードファイル格納ディレクトリ名〔pdstedit〕　1395
統計入力アンロードファイル名〔pdstedit〕　1394
統計ログファイルの切り替え　520
統計ログファイルへの統計ログバッファの反映　522
動作環境〔pdconstck〕　1356
動作条件〔pdconstck〕　1356
動作モード〔pdrbal〕　1256
同時実行最大プリフェッチ数　92
同時実行トランザクション決着待ち時間〔pdreclaim〕　1309
同時にバックアップを取得する必要がある RD エリア　1749
トラブルシュート情報の削除　158
トランザクションキューイング機能の開始　555
トランザクションキューイング機能の解除　555
トランザクションの強制終了　193
トランザクションのコミット　148
トランザクションのロールバック　470
トリガ識別子〔pdexp〕　1226
トリガ定義情報の搬出入　1222
トリガ定義情報の搬出入についての規則〔pdexp〕　1230

な

ナル値オプション〔pdload の array 文〕　686
ナル値オプション〔pdload の列名文〕　776
ナル値・関数情報ファイル〔pdload〕　796
ナル値・関数情報ファイル名〔pdload〕　646

に

入力開始行〔pdload〕　654
入力データ長〔pdload の source 文〕　670
入力データファイル　608, 665, 717
入力データファイル名〔pdload の source 文〕　666
入力待ちのプロンプトの表示〔pddef〕　596
入力ログファイルのサマリ情報　1404
認可識別子〔pdaudput〕　1846
認可識別子〔pdconstck〕　1377
認可識別子〔pddbst〕　1515, 1516
認可識別子〔pdexp〕　1227
認可識別子〔pdgetcst〕　1614, 1615
認可識別子〔pdinit〕　566
認可識別子〔pdload〕　651, 662
認可識別子〔pdmod の create auditor 文〕　966
認可識別子〔pdpgbfon〕　1346
認可識別子〔pdrbal〕　1256, 1259
認可識別子〔pdrbal の reld_func 文〕　1273
認可識別子〔pdrbal の unld_func 文〕　1272
認可識別子〔pdreclaim〕　1308, 1309
認可識別子〔pdrorg〕　1091, 1107
認可識別子〔pdrorg の reld_func 文〕　1140
認可識別子〔pdrorg の tblname 文〕　1137
認可識別子〔pdrorg の unld_func 文〕　1139

ね

ネストループジョイン　1690

は

ハードウェア圧縮機能〔テープ装置アクセス機能〕　50
排他制御モード　1859
排他制御リトライ回数　1319
排他待ち限界経過時間〔pdcopy〕　1760

バイト数〔pdload のスキップデータ文〕 787
バイナリ形式 609, 725
配列データ形式〔pdload の array 文〕 685
配列データ形式〔pdload の列名文〕 775
パスワード〔pddbst〕 1515
パスワード〔pdgetcst〕 1615
パスワード〔pdinit〕 566
バックアップ検索条件開始時刻〔pdrstr〕 1804
バックアップ検索条件終了時刻〔pdrstr〕 1804
バックアップ取得モード 1747, 1756, 1787
バックアップの取得単位 1747
バックアップファイル初期容量〔pdcopy〕 1770
バックアップファイルの形式 1780
バックアップファイルの種別〔pdcopy〕 1768
バックアップファイルの種別〔pdrstr〕 1803
バックアップファイルの情報表示 80
バックアップファイル名〔pdcopy〕 1765
バックアップファイル名〔pdrstr〕 1797
バックアップファイルを格納するサーバマシン 1749
バックアップ閉塞 226
バックアップを取得するために必要な知識 1747
ハッシュジョイン〔pdvwopt〕 1690
ハッシュ転送 1693
ハッシュ分割表のリバランス機能 1245
バッファ単位取得方式〔順序番号の取得方式〕 615
範囲指定の回復 1790, 1810
搬出ファイル名〔pdexp〕 1224
搬入時に作成される定義系 SQL が 2,000,000 バイトを超えた場合の修正例 1236

ひ

引数の型〔pdrbal の reld_func 文〕 1273
引数の型〔pdrbal の unld_func 文〕 1272
引数の型〔pdrorg の reld_func 文〕 1140
引数の型〔pdrorg の unld_func 文〕 1139
一つのインデクスを使用した検索 1691
表移行用アンロードファイル 757
表識別子〔pdconstck〕 1376
表識別子〔pddbst〕 1516
表識別子〔pdexp〕 1226
表識別子〔pdgetcst〕 1615
表識別子〔pdload〕 662
表識別子〔pdmod の create audit table 文〕 969
表識別子〔pdpgbfon〕 1346
表識別子〔pdrbal〕 1256
表識別子〔pdreclaim〕 1308
表識別子〔pdrorg〕 1091
表識別子〔pdrorg の tblname 文〕 1137
表単位〔検査保留状態の強制変更機能の実行単位〕 1369
表単位〔整合性チェック機能の実行単位〕 1359, 1360
表単位の状態解析 1504, 1561
表定義情報の搬出入 1221
表定義情報の搬出入についての規則〔pdexp〕 1230
表のアンロード 978, 1006
表の再編成 975, 989
表のリバランス 1245
表へのリロード 981, 1042

ふ

ファイル 47
ファイルサイズ〔pdaudput〕 1848
ファイルセキュリティ強化機能の設定 488
ファイルの内容表示 113
ファイルパス変更オフラインコマンド 127
ファイルパス変更オンラインコマンド 130
付加 PP 61
複数のインデクスを使用した検索 1692
物理的解析 1551
部分ブロードキャスト転送 1695
不要データの削除〔UOC を利用した再編成〕 1163
プラグインインデクス作成上の注意事項〔pdload〕 641
プラグインが提供する抽象データ型列（LOB 属性）がある表へのデータロード 611
プラグインが提供する抽象データ型列がある表へのデータロード 626
プラグイン提供関数の種別 1698
プラグインの登録・削除 456

ブロードキャスト転送　1694
プロセスの強制停止　251
分割入力データファイル　621
分割入力データファイルの出力先ディレクトリ名〔pdload の src_work 文〕　691

へ

閉塞中の監査証跡ファイルの削除　76
ページコンパクション　1291
ページ長〔pdinit の create rdarea 文〕　573
ページ長〔pdmod の create rdarea 文〕　905
ページ長〔pdmod の initialize rdarea 文〕　923
ページ長〔pdreginit の create rdarea 文〕　1823
ページの詳細情報〔pddbst〕　1514
別表への表データ再登録　1136
変更後の RD エリア名〔pdmod の alter rdarea 文〕　956
編集項目〔pdstedit〕　1391

ほ

ホスト名〔pdload の lobmid 文〕　680
ホスト名〔pdload の source 文〕　666
ホスト名〔pdstedit〕　1394
ボリューム　47

ま

マージ解析〔状態解析結果蓄積機能（pddbst）〕　1601
マージジョイン　1689
マスタディレクトリ用 RD エリアの整合性チェック　390
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名〔pdcopy〕　1756
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名〔pdrstr〕　1797
マスタディレクトリ用 RD エリアを回復する場合　1787
マトリクス分割転送　1693

み

未決着トランザクションの強制自動決着　551
未使用セグメント　1286
未使用ページ　1286

め

明示的グループ分け処理　1696
メッセージログファイルの内容表示　117
メモリ領域確保サイズ〔pdrorg の unlduoc 文〕　1135
メンテナンス方法の情報　1582

も

文字集合を定義した表へのデータロード　612
文字列形式の数値データ　733
文字列形式の文字データ　733

ゆ

ユーザ LOB 用 RD エリアの満杯 HiRDB ファイルの更新凍結　162
ユーザの状態表示　316
ユーザパラメタ〔pdload の srcuoc 文〕　682
ユティリティ一覧　44
ユティリティ処理の強制終了　109
ユティリティの最大同時実行数　1895
ユティリティの実行者　44, 45
ユティリティの入出力ファイル　47
ユティリティの排他制御モード　1860
ユニキャスト転送　1694
ユニットごとの統計情報出力種別の指定有無の表示　359
ユニット識別子〔pdcopy〕　1766
ユニット識別子〔pdrstr〕　1809
ユニットの開始　495
ユニットの稼働状態の表示　379
ユニットの終了　526
ユニットの状態表示　367
ユニットのステータス情報　369

よ

要素数〔pdload のスキップデータ文〕　789
要素数〔pdload の列名文〕　775
要素としてのナル値　747
予測レベル 1　1507, 1508
予測レベル 2　1507–1509

予測レベル 2 での状態解析結果蓄積機能の実行方法 1601

ら

ライブラリ名称〔pdload の srcuoc 文〕 682
ランダムアクセス用ローカルバッファ面数〔pdload〕 647
ランダムアクセス用ローカルバッファ面数〔pdrbal〕 1261
ランダムアクセス用ローカルバッファ面数〔pdrorg〕 1108

り

リスト定義情報の表示 252
リバランス中の SQL 実行可否 1279
リバランス表 1245
リバランス表の再編成 1201
リバランスユティリティ 1244
リバランスユティリティの排他制御モード 1872
リバランスユティリティ〔ユティリティー覧〕 44
領域確保サイズ〔pdload の option 文〕 706

る

ルーチン識別子〔pdexp〕 1227

れ

レジストリ機能初期設定ユティリティ 1818
レジストリ機能初期設定ユティリティの排他制御モード 1877
レジストリ機能初期設定ユティリティ〔ユティリティー覧〕 45
列構成情報ファイル〔pdload〕 765
列構成情報ファイル名〔pdload〕 646
列単位 LOB 入力ファイル〔pdload〕 611, 677, 753
列単位 LOB 入力ファイル名〔pdload の lobcolumn 文〕 678
列データ一部置換〔順序番号の格納方式〕 615
列データ全置換〔順序番号の格納方式〕 615
列データ追加〔順序番号の格納方式〕 615
列データ・パラメタの記述形式（DAT 形式） 718
列データ・パラメタの記述形式（バイナリ形式） 725
列としてのナル値 747
列番号〔pdload の列番号文〕 796
列番号文〔pdload〕 796
列名〔pdload の列名文〕 768
列名〔pdrorg の blobtovarchar 文〕 1157
列名文〔pdload〕 766
連続認証失敗アカウントロック状態の解除 63

ろ

ログ関係のファイルのアンロード 308
ログ関係のファイルのオープン 294
ログ関係のファイルのクローズ 268
ログ関係のファイルの削除 296
ログ関係のファイルの情報表示 276
ログ関係のファイルの初期設定 270
ログ関係のファイルのステータス変更 264
ログ関係のファイルのスワップ 299
ログ関係のファイルの割り当て 256
ログ取得方式〔pdload〕 642
ログ取得方式〔pdrbal〕 1258
ログ取得方式〔pdrorg〕 1095
ログ取得モード〔pdload〕 642
ログ取得モード〔pdrbal〕 1258
ログ取得モード〔pdrorg〕 1096
ログポイント情報ファイル 1749, 1812
ログポイント情報ファイルの再作成 1785
ログポイント情報ファイル名〔pdcopy〕 1760
ログポイント情報ファイル名〔pdrstr〕 1812
ログレコードサマリ情報 1403
ログレスモード〔pdload〕 643
ログレスモード〔pdrbal〕 1259
ログレスモード〔pdrorg〕 1096
論理エラー番号 758
論理的解析 1524

わ

ワークファイル〔pdload〕 657
ワークファイル〔pdrbal〕 1264

ワークファイル〔pdrorg〕　1117
ワーク用一時ファイル作成ディレクトリ名〔pdstedit〕
1396