HITACHI

ノンストップデータベース

HiRDB Version 10 システム定義（UNIX(R)用）

文法書

3020-6-554-C0

前書き

■ 対象製品

●適用 OS：AIX V7.2，AIX V7.3

P-1M62-35A1 HiRDB Server Version 10 10-12

P-1M62-1BA1 HiRDB/Run Time Version 10 10-12

P-1M62-1CA1 HiRDB/Developer's Kit Version 10 10-12

P-1M62-1DA1 HiRDB/Run Time Version 10(64) 10-12

P-1M62-1EA1 HiRDB/Developer's Kit Version 10(64) 10-12

P-F1M62-11A13 HiRDB Staticizer Option Version 10 10-00

P-F1M62-11A15 HiRDB Non Recover Front End Server Version 10 10-00

P-F1M62-11A16 HiRDB Advanced High Availability Version 10 10-00

P-F1M62-11A18 HiRDB Disaster Recovery Light Edition Version 10 10-00

P-F1M62-11A1A HiRDB Accelerator Version 10 10-00

●適用 OS：Red Hat Enterprise Linux 8 (64-bit x86_64)，Red Hat Enterprise Linux 9 (64-bit x86_64)，Red Hat Enterprise Linux 10 (64-bit x86_64)

P-8462-35A1 HiRDB Server Version 10 10-12

P-8462-1DA1 HiRDB/Run Time Version 10(64) 10-12

P-8462-1EA1 HiRDB/Developer's Kit Version 10(64) 10-12

P-F8462-11A13 HiRDB Staticizer Option Version 10 10-00

P-F8462-11A15 HiRDB Non Recover Front End Server Version 10 10-00

P-F8462-11A16 HiRDB Advanced High Availability Version 10 10-00

P-F8462-11A18 HiRDB Disaster Recovery Light Edition Version 10 10-00

P-F8462-11A1A HiRDB Accelerator Version 10 10-00

●適用 OS：Red Hat Enterprise Linux 8 (64-bit x86_64)，Red Hat Enterprise Linux 9 (64-bit x86_64)，Red Hat Enterprise Linux 10 (64-bit x86_64)

P-8362-1BA1 HiRDB/Run Time Version 10 10-12

P-8362-1CA1 HiRDB/Developer's Kit Version 10 10-12

P-8362-3CA1 HiRDB Developer's Suite Version 10 10-12

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

■ 輸出時の注意

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

なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ 発行

2026 年 9 月　3020-6-554-C0

■ 著作権

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

変更内容

変更内容(3020-6-554-C0)　HiRDB Version 10 10-12

追加・変更内容	関連するオペランド	変更個所
サーバのユニット間通信で，定期的に KeepAlive パケットを送信する機能をサポートしました。 これによって，ルータやファイアウォールなどの無通信時間監視機能で，HiRDB のユニット間接続が意図しないで切断されることを防止できます。	pd_ipc_so_keepalive pd_ipc_tcp_keepidle pd_ipc_tcp_keepintvl pd_ipc_tcp_keepcnt	2.1.1(40)，9.1.1(50)，9.2.50，表 E-1，表 F-1，表 G-1
デファードライト処理の並列 WRITE 機能で，連続ページの一括出力を促進し，書き込み発行回数を削減する機能をサポートしました。 これによって，デファードライト処理の書き込み性能（スループット）の向上が期待できます。	pd_dfw_bulk_write	4.1，5.1，7.1，8.1，9.1.1(22)，9.2.22，表 E-1，表 F-1，表 G-1

単なる誤字・脱字などはお断りなく訂正しました。

変更内容(3020-6-554-B0)　HiRDB Version 10 10-11

追加・変更内容	関連するオペランド
システムログが複製又は共有される環境で，更新 DB の更新情報を自動的に参照 DB に反映する機能をサポートしました。 また，プラグインインデクスやユティリティによるインデクス一括作成時の更新情報の反映もサポートしました。 これらによって，HiRDB のすべての表形式に対して，ユーザ操作なしで自動反映できるようになります。	pdlogadfg -d rrp pdlogadpf -d rrp pd_read_replica_max_trn_count pd_read_replica_mode
HiRDB の適用 OS に Red Hat Enterprise Linux 10 を追加しました。	－

変更内容(3020-6-554-A0)　HiRDB Version 10 10-10

追加・変更内容	関連するオペランド
リードレプリカ機能は FIX 表の表データの反映だけをサポートしていましたが，非 FIX 表の表データ及び FIX 表と非 FIX 表のインデクスデータも反映できるようになります。	pd_clt_waittime_over_dump_level pd_rpl_reflect_mode
表のアクセス権限を一括で管理できるロール機能をサポートしました。これによって，権限の管理や移行が容易になるだけでなく，アクセス制御の一貫性が向上し，セキュリティの強化にもつながります。	－

変更内容(3020-6-554-90)　HiRDB Version 10 10-09

追加・変更内容	関連するオペランド
データベースの内容を複製した参照専用のデータベースを構築できるようにしました。 これによって，アプリケーション側で更新用 DB と参照用 DB（リードレプリカ）を使い分け，負荷分散及びシステム全体のスループット向上につながります。	pd_read_replica_use pd_read_replica_workdir

はじめに

このマニュアルは，プログラムプロダクト ノンストップデータベース　HiRDB Version 10 のシステム定義について説明したものです。なお，ここに記載されていない前提情報については，マニュアル「HiRDB Version 10 解説」を参照してください。

■ 対象読者

HiRDB Version 10（以降，**HiRDB** と表記します）を使ってリレーショナルデータベースシステムを構築または運用する方々を対象にしています。

このマニュアルは次に示す知識があることを前提に説明しています。

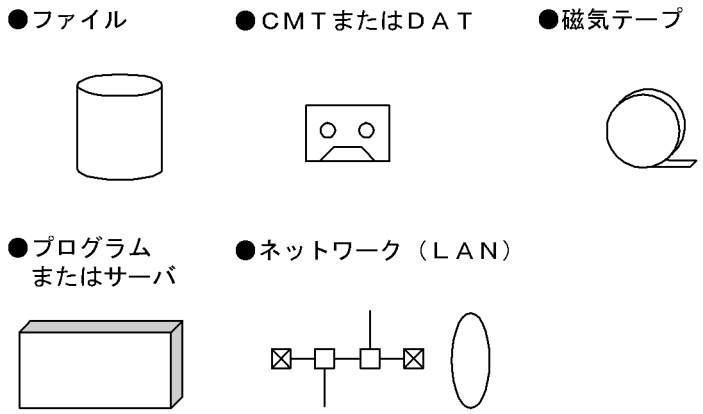
- UNIX または Linux のシステム管理の基礎的な知識
- SQL の基礎的な知識

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

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

■ 図中で使用している記号

このマニュアルの図中で使用している記号を，次のように定義します。



■ このマニュアルで使用している記号

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

記号	意味
〔 〕	この記号で囲まれている項目は省略できます。 （例）pdbuffer 〔-p〕 　これは，pdbuffer と指定するか，または pdbuffer -p と指定することを示します。
…	この記号の直前の項目を繰り返して指定できます。 （例）pdbuffer -r RD エリア名 〔，RD エリア名〕… 　これは，-r オプションの RD エリア名を繰り返し指定できることを示します。
{{ }}	この記号で囲まれた複数の項目を一つの単位として，繰り返して指定できます。 （例）{{pdbuffer -a バッファ名}} 　これは，pdbuffer -a バッファ名 　　pdbuffer -a バッファ名 と指定できることを示します。
｜	この記号で区切られた項目は選択できます。 （例）pdlogadfg -d sys ｜ spd 　これは，-d オプションに sys か spd のどちらかを指定できることを示します。
{ }	この記号で囲まれている複数の項目のうちから，一つを選択できます。 （例）pdbuffer 〔{-r RD エリア名｜-i 認可識別子．インデクス識別子｜-o}〕 　これは，-r RD エリア名，-i 認可識別子．インデクス識別子，-o の三つのオプションのうち，どれか一つを指定することを示します。
__ （下線）	この記号で示す項目は，省略時の解釈値です。 （例）〔 pd_rpc_trace = Y ｜ N 〕 　これは，オペランドの指定を省略した場合，N が仮定されることを示します。
～	この記号のあとにユーザ指定値の属性を示します。
《 》	ユーザが指定しなかった場合に仮定される値を示します。
＜ ＞	ユーザ指定値の構文要素を示します。
(( ))	ユーザ指定値の指定範囲を示します。

■ このマニュアルで使用している構文要素記号

このマニュアルで使用している構文要素記号を次に示します。

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

構文要素記号	意味
＜符号なし整数＞	数字
＜符号なし 10 進数＞※1	数字（0～9 の並び）ピリオド（.）数字（0～9 の並び）
＜ 16 進数字＞	数字と（A～F，a～f）
＜識別子＞※2	先頭がアルファベットの英数字列
＜記号名称＞	先頭が英字記号の英数字記号列
＜文字列＞	任意の文字の配列
＜パス名＞※3	/，英数字，ピリオド（.），#，@
＜ホスト名＞※4	アルファベット（A～Z，a～z），数字，ピリオド（.），ハイフン（-），下線（_），および@で構成される文字列

注

すべて半角文字を使用してください。また，英字の大文字と小文字は区別されます。

注※1

ピリオドの前の数字がすべて 0 の場合，ピリオドより前の 0 を省略できます。また，ピリオドの後ろの数字がすべて 0 の場合，ピリオド以降を省略できます。

例１：0.008 → .008

例２：15.000 → 15

注※2

RD エリア名を指定する場合：

・英数字記号，__（下線），ハイフン（-），または空白を使用してください。ただし，先頭は英字記号を使用してください。

・空白を使用する場合は RD エリア名を引用符（"）で囲んでください。

注※3

パス名は使用している OS に依存します。

注※4

ホスト名は次に示すどれかの形式で指定できます。

・ホスト名

・IP アドレス（XXX.XXX.XXX.XXX 形式）

・FQDN

ただし，同じホスト名を異なる形式で指定しないでください。異なる形式で指定した場合，HiRDB は異なるホスト名として認識します。

また，ループバックアドレスを指定することもできます。

システム定義に指定するホスト名については，ホスト名の登録が必要になります。詳細については，マニュアル「HiRDB Version 10 システム導入・設計ガイド」の「ホスト名の登録」を参照してください。

目次

前書き　2
変更内容　4
はじめに　6

1　概要　15
1.1　HiRDB/シングルサーバを利用される方へ　16
1.1.1　HiRDB システム定義の種類　16
1.1.2　HiRDB システム定義の作成方法　16
1.1.3　HiRDB システム定義ファイルの構成　17
1.1.4　各定義の関係　18
1.1.5　HiRDB システム定義（UAP 環境定義を除く）の変更手順　19
1.1.6　UAP 環境定義の追加又は変更手順　21
1.2　HiRDB/パラレルサーバを利用される方へ　22
1.2.1　HiRDB システム定義の種類　22
1.2.2　HiRDB システム定義の作成方法　23
1.2.3　HiRDB システム定義ファイルの構成　24
1.2.4　各定義の関係　25
1.2.5　HiRDB システム定義（UAP 環境定義を除く）の変更手順　27
1.2.6　UAP 環境定義の追加又は変更手順　29
1.2.7　影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成　29
1.2.8　HiRDB システム定義ファイルの共用化　30
1.3　HiRDB システム定義を記述するときの文法規則　33
1.3.1　オペランドの指定形式　33
1.3.2　コメントの記述　33
1.3.3　行の継続　34
1.3.4　オペランドの重複指定について　34
1.3.5　注意事項　34
1.4　バージョンによって省略値が異なるオペランド，及び指定不要になったオペランド　35
1.4.1　バージョン，リビジョンによる HiRDB システム定義の変更点　35
1.4.2　バージョン 09-50 より前の HiRDB システム定義の変更点　48
1.4.3　32 ビットモードと 64 ビットモードで省略値が異なるオペランド　51
1.5　省略値で定義できるデータベースの規模　54

2　システム共通定義　55
2.1　オペランドの形式　56

2.1.1 オペランドの形式の説明 56

3 ユニット制御情報定義 74
3.1 オペランドの形式 75
3.1.1 オペランドの形式の説明 75

4 サーバ共通定義 83
4.1 オペランドの形式 84
4.1.1 オペランドの形式の説明 84

5 シングルサーバ定義 89
5.1 オペランドの形式 90
5.1.1 オペランドの形式の説明 90

6 フロントエンドサーバ定義 97
6.1 オペランドの形式 98
6.1.1 オペランドの形式の説明 98

7 ディクショナリサーバ定義 104
7.1 オペランドの形式 105
7.1.1 オペランドの形式の説明 105

8 バックエンドサーバ定義 111
8.1 オペランドの形式 112
8.1.1 オペランドの形式の説明 112

9 オペランドの詳細 119
9.1 オペランドの一覧 120
9.1.1 オペランド名および定義内容の一覧 120
9.2 オペランドの説明 161
9.2.1 システム構成に関するオペランド 161
9.2.2 同時実行最大数に関するオペランド 181
9.2.3 プロセスに関するオペランド 187
9.2.4 作業表に関するオペランド 196
9.2.5 HiRDB の開始方法に関するオペランド 202
9.2.6 縮退起動に関するオペランド 206
9.2.7 HiRDB の処理方式に関するオペランド 207
9.2.8 全面回復処理に関するオペランド 218
9.2.9 トランザクション決着処理に関するオペランド 220
9.2.10 SQL の仕様に関するオペランド 226
9.2.11 SQL の最適化に関するオペランド 232

9.2.12 拡張 SQL エラー情報出力機能に関するオペランド 239
9.2.13 SQL 予約語削除機能に関するオペランド 241
9.2.14 SQL からのコマンド実行に関するオペランド 242
9.2.15 SQLSTATE の細分化に関するオペランド 242
9.2.16 絞込み検索に関するオペランド 243
9.2.17 システム監視に関するオペランド 245
9.2.18 排他制御に関するオペランド 264
9.2.19 バッファに関するオペランド 286
9.2.20 共用メモリに関するオペランド 293
9.2.21 RD エリアに関するオペランド 301
9.2.22 グローバルバッファに関するオペランド 319
9.2.23 インメモリデータ処理に関するオペランド 339
9.2.24 表又はインデクスの予約数に関するオペランド 341
9.2.25 参照制約及び検査制約に関するオペランド 342
9.2.26 一時表に関するオペランド 343
9.2.27 システムログファイルに関するオペランド 345
9.2.28 シンクポイントダンプファイルに関するオペランド 356
9.2.29 ステータスファイルに関するオペランド 361
9.2.30 メッセージログファイルに関するオペランド 376
9.2.31 統計情報に関するオペランド 377
9.2.32 RPC トレース情報に関するオペランド 387
9.2.33 PRF トレース情報に関するオペランド 388
9.2.34 トラブルシュート情報に関するオペランド 393
9.2.35 バックエンドサーバ接続保持機能に関するオペランド 406
9.2.36 再編成時期予測機能に関するオペランド 407
9.2.37 セキュリティに関するオペランド 408
9.2.38 系切り替え機能に関するオペランド 420
9.2.39 HiRDB Datareplicator に関するオペランド 439
9.2.40 リアルタイム SAN レプリケーションに関するオペランド 442
9.2.41 インナレプリカ機能に関するオペランド 444
9.2.42 OLTP に関するオペランド 450
9.2.43 日付・時刻に関するオペランド 451
9.2.44 ユーザ認証に関するオペランド 451
9.2.45 メッセージの出力抑止機能に関するオペランド 452
9.2.46 HiRDB システム定義ファイルの共用化に関するオペランド 453
9.2.47 クライアントグループに関するオペランド 454
9.2.48 プラグインに関するオペランド 455
9.2.49 バージョンアップに関するオペランド 457
9.2.50 通信処理に関するオペランド 457

9.2.51 Java に関するオペランド 474
9.2.52 外部 C ストアドルーチンに関するオペランド 477
9.2.53 SQL 実行時間警告出力機能に関するオペランド 478
9.2.54 文字コードに関するオペランド 480
9.2.55 ワークファイル出力先ディレクトリの変更 482
9.2.56 HiRDB ファイルシステム領域に関するオペランド 483
9.2.57 リードレプリカ機能に関するオペランド 485

10 UAP 環境定義 488
10.1 オペランドの形式 489
10.2 オペランドの説明 490

11 互換モード適用時に検討が必要なオペランド 494
11.1 オペランドの一覧 495
11.2 オペランドの形式 498
11.2.1 オペランドの形式の説明 498
11.3 オペランドの説明 500
11.3.1 同時実行最大数に関するオペランド 500
11.3.2 HiRDB の処理方式に関するオペランド 501
11.3.3 トランザクション決着処理に関するオペランド 502
11.3.4 SQL の仕様に関するオペランド 503
11.3.5 SQL の最適化に関するオペランド 503
11.3.6 一時表に関するオペランド 505
11.3.7 通信処理に関するオペランド 506
11.3.8 日付・時刻に関するオペランド 520
11.3.9 ユニット用ステータスファイル（障害発生時）に関するオペランド 521
11.3.10 サーバ用ステータスファイル（障害発生時）に関するオペランド 526
11.3.11 システムログファイルに関するオペランド 531
11.4 pdconfchk コマンドでチェックできるオペランド 534
11.5 適用 OS ごとのオペランド指定可否 536

付録 537
付録 A 影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義） 538
付録 B 定義例 540
付録 B.1 HiRDB/シングルサーバの場合 540
付録 B.2 HiRDB/シングルサーバの場合：ユティリティ専用ユニットあり 542
付録 B.3 HiRDB/シングルサーバの場合：系切り替え機能使用時 544
付録 B.4 HiRDB/パラレルサーバの場合 548
付録 B.5 HiRDB/パラレルサーバの場合：スタンバイ型系切り替え機能使用時 556

付録 B.6　HiRDB/パラレルサーバの場合：1：1 スタンバイレス型系切り替え機能使用時　558
付録 C　オペランド指定値の見積もり式　562
付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式　562
付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式　568
付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式　578
付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式　584
付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式　588
付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式　590
付録 D　排他資源数の見積もり　593
付録 D.1　定義系 SQL　593
付録 D.2　操作系 SQL　650
付録 D.3　制御系 SQL　682
付録 D.4　ユティリティ及びコマンド　683
付録 D.5　HiRDB 起動時に必要となる排他資源数　690
付録 E　pdconfchk コマンドでチェックできるオペランド　692
付録 F　適用 OS ごとのオペランド指定可否　706
付録 G　リードレプリカ機能適用時の参照 DB の定義変更可否　719

索引　732

1 概要

この章では，HiRDB システム定義の体系，及び文法規則について説明します。

1.1　HiRDB/シングルサーバを利用される方へ

HiRDB 管理者は，HiRDB の実行環境を設定するために HiRDB システム定義を作成します。ここで説明する項目は次のとおりです。

- HiRDB システム定義の種類
- HiRDB システム定義の作成方法
- HiRDB システム定義ファイルの構成
- 各定義の関係
- HiRDB システム定義（UAP 環境定義を除く）の変更手順
- UAP 環境定義の追加又は変更手順

1.1.1　HiRDB システム定義の種類

HiRDB システム定義は次の表に示す定義で構成されています。HiRDB 管理者は各定義のオペランドの指定値を決めてください。

表 1–1　HiRDB システム定義の種類（HiRDB/シングルサーバの場合）

定義の種類	内容
システム共通定義	HiRDB の構成及び共通情報を定義します。一つの HiRDB/シングルサーバに一つ必要です。システム共通定義のオペランドについては，「システム共通定義」を参照してください。
ユニット制御情報定義	ユニットの制御情報を定義します。一つのユニットに一つ必要です。ユニット制御情報定義のオペランドについては，「ユニット制御情報定義」を参照してください。
サーバ共通定義	シングルサーバ定義のオペランドの仮定値を定義します。必要に応じて定義してください。サーバ共通定義のオペランドについては，「サーバ共通定義」を参照してください。
シングルサーバ定義	シングルサーバの実行環境を定義します。一つのシングルサーバに一つ必要です。ユティリティ専用ユニットには定義する必要はありません。シングルサーバ定義のオペランドについては，「シングルサーバ定義」を参照してください。
UAP 環境定義	UAP の実行環境を定義します。必要に応じて定義してください。UAP 環境定義のオペランドについては，「UAP 環境定義」を参照してください。UAP 環境定義は最大 4096 個作成できます。
SQL 予約語定義	SQL の予約語を定義します。SQL 予約語削除機能使用時に必要です。SQL 予約語削除機能については，pd_delete_reserved_word_file オペランドを参照してください。

1.1.2　HiRDB システム定義の作成方法

HiRDB 管理者は次に示す方法で HiRDB システム定義を作成してください。

- OS のテキストエディタで作成

作成した HiRDB システム定義は次の表に示すファイルに格納してください。なお，次の表に示すファイルを総称して **HiRDB システム定義ファイル**といいます。

表 1–2　HiRDB システム定義を格納するファイル（HiRDB/シングルサーバの場合）

定義の種類	格納ファイル名
システム共通定義	$PDDIR/conf/pdsys
ユニット制御情報定義	$PDDIR/conf/pdutsys
サーバ共通定義	$PDDIR/conf/pdsvrc
シングルサーバ定義	$PDDIR/conf/サーバ名※1
UAP 環境定義	$PDDIR/conf/pduapenv/※2 任意の名称
SQL 予約語定義	$PDDIR/conf/pdrsvwd/※2 任意の名称

注※1

システム共通定義の pdstart オペランドの-s オプションで指定するサーバ名と同じにしてください。

注※2

- ファイル名称は先頭がアルファベットの英数字列（最大 8 文字）にしてください。
- UAP 環境定義又は SQL 予約語定義を使用するユーザに対して，それぞれのファイルが格納されているディレクトリの読み込み権限（r）と実行権限（x）を与えてください。また，ファイルには読み込み権限（r）を与えてください。

> **注意事項**
>
> HiRDB システム定義ファイルのパーミッションは，ファイルの所有者（HiRDB 管理者）にだけ，読み込み権限及び書き込み権限を持たせるように設定，維持するようにしてください。

(1)　OS のテキストエディタで作成する場合

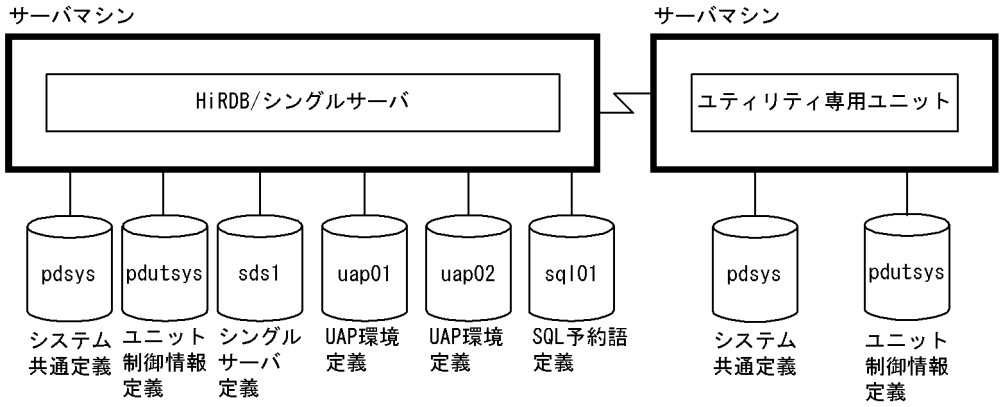
コマンドで HiRDB の環境設定をする場合は，OS のテキストエディタで HiRDB システム定義を作成します。HiRDB の実行環境に合わせて，HiRDB システム定義の各オペランドを指定してください。

なお，HiRDB システム定義を作成した後に，**pdconfchk** コマンドで HiRDB システム定義のオペランドの整合性をチェックしてください。チェック対象となるオペランドについては，「pdconfchk コマンドでチェックできるオペランド」を参照してください。

1.1.3　HiRDB システム定義ファイルの構成

OS のテキストエディタで HiRDB システム定義を作成する場合は，次の図を参考にして作成してください。

図 1–1　HiRDB システム定義ファイルの構成（HiRDB/シングルサーバの場合）



注 ユティリティ専用ユニットにシングルサーバ定義，UAP環境定義，及びSQL予約語定義は必要ありません。

1.1.4　各定義の関係

異なる定義で同じオペランドを指定した場合，HiRDB は次に示す優先順位に従って指定値を決定します。

1. シングルサーバ定義
2. サーバ共通定義
3. ユニット制御情報定義
4. システム共通定義

優先順位の低い定義で標準となる値を指定して，優先順位の高い定義で値を変更できます。

(1)　クライアント環境定義との関係

次の表に示す HiRDB システム定義のオペランドの値は，クライアント環境定義でクライアントごとに変更できます。

表 1–3　クライアント環境定義で指定値を変更できるオペランド（HiRDB/シングルサーバの場合）

HiRDB システム定義のオペランド	クライアント環境定義のオペランド	オペランドの内容
pd_additional_optimize_level	PDADDITIONALOPTLVL	SQL 拡張最適化オプションを指定します。
pd_cwaittime_wrn_pnt	PDCWAITTIMEWRNPNT	SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。
pd_delete_reserved_word_file	PDDELRSVWDFILE	SQL 予約語削除機能を使用する場合，SQL 予約語削除ファイルの名称を指定します。

HiRDB システム定義のオペランド	クライアント環境定義のオペランド	オペランドの内容
pd_hash_table_size	PDHASHTBLSIZE	SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシュ表サイズを指定します。
pd_optimize_level	PDSQLOPTLVL	SQL 最適化オプションを指定します。
pd_space_level	PDSPACELVL	空白変換レベルを指定します。
pd_uap_exerror_log_param_size	PDUAPEXERLOGPRMSZ	クライアントエラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。
pd_uap_exerror_log_use	PDUAPEXERLOGUSE	拡張 SQL エラー情報出力機能を使用するかどうかを指定します。
pd_watch_pc_client_time	PDSWATCHTIME	Windows 対応の HiRDB クライアントからの要求待ち時間を指定します。
pd_isolation_level	PDISLLVL	データ保証レベルを指定します。

クライアントごとに値を変更する場合は，クライアント環境定義の該当するオペランドを指定してください。クライアント環境定義については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

1.1.5　HiRDB システム定義（UAP 環境定義を除く）の変更手順

(1)　HiRDB システム定義の変更手順

HiRDB システム定義の変更手順を次に示します。なお，$PDDIR/conf はユニット制御情報定義ファイルを格納しているディレクトリを意味しています。$PDCONFPATH はそれ以外の HiRDB システム定義ファイルを格納しているディレクトリを意味しています。

〈手順〉

1. $PDDIR/conf 及び$PDCONFPATH 下にサブディレクトリを作成します。この例ではサブディレクトリとして work を作成します。
2. ユニット制御情報定義ファイルを$PDDIR/conf/work 下にコピーします。そのほかの HiRDB システム定義ファイルを$PDCONFPATH/work 下にコピーします。
3. $PDDIR/conf/work 及び$PDCONFPATH/work 下にコピーした HiRDB システム定義を変更します。
4. **pdconfchk -d work** コマンドで，$PDDIR/conf/work 及び$PDCONFPATH/work 下の HiRDB システム定義の内容をチェックします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdstop** コマンドで HiRDB を正常終了します。
6. **pdlogunld** コマンドで，アンロード待ち状態のシステムログファイルをアンロードします。

7. 3 で変更した HiRDB システム定義ファイルを$PDDIR/conf 又は$PDCONFPATH 下にコピーして，HiRDB システム定義ファイルを置き換えます。
8. 次に示すオペランドの指定値を変更した場合は，**pdloginit** コマンドでシステムログファイルを初期化します。
   ・pd_log_dual
   ・pdstart
9. **pdstart** コマンドで HiRDB を正常開始します。

(2) システム構成変更コマンドを使用した HiRDB システム定義の変更手順

システム構成変更コマンド（pdchgconf コマンド）を使用すると，HiRDB の稼働中に HiRDB システム定義を変更できるため，HiRDB を正常終了する必要はありません。ただし，このコマンドを使用する場合は HiRDB Advanced High Availability が必要になります。システム構成変更コマンドを使用した HiRDB システム定義の変更手順を次に示します。

〈手順〉

1. $PDDIR/conf/chgconf ディレクトリを作成します。
2. 使用中の HiRDB システム定義ファイルを 1 で作成したディレクトリ下にコピーします。
3. $PDDIR/conf/chgconf 下の HiRDB システム定義を変更します。
4. **pdconfchk -d chgconf** コマンドで，$PDDIR/conf/chgconf 下の HiRDB システム定義のチェックをします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdchgconf** コマンドで，HiRDB システム定義を変更後の HiRDB システム定義に置き換えます。
   pdchgconf コマンドを実行すると，使用中（変更前）の HiRDB システム定義ファイルが$PDDIR/conf/backconf 下に退避されます。そして，$PDDIR/conf/chgconf 下の変更後の HiRDB システム定義ファイルが$PDDIR/conf 下にコピーされます。

注意事項

- pdchgconf コマンドの入力後，15 分以上トランザクション又はユティリティが動き続けた場合，pdchgconf コマンドが異常終了します。
- システム構成変更コマンドを使用した HiRDB システム定義の変更には制限事項があります。制限事項については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(3) 注意事項

- 稼働中の HiRDB が使用している HiRDB システム定義は変更又は削除しないでください。変更又は削除した場合はその HiRDB の動作を保証できません。
- HiRDB が計画停止，強制終了，又は異常終了した場合，HiRDB システム定義のオペランドで変更できるものと変更できないものがあります。詳細については，「オペランドの一覧」を参照するか，又は各定義文のオペランド一覧を参照してください。

- HiRDB システム定義を変更した後に，$PDDIR/conf 下のファイルのバックアップを取得してください。HiRDB 運用ディレクトリがあるディスクの障害などに備えて，HiRDB 運用ディレクトリ下のファイル（$PDDIR/conf 下のファイル）のバックアップを取得します。HiRDB 運用ディレクトリを回復するには，$PDDIR/conf 下のファイルのバックアップが必要になります。また，$PDCONFPATH が HiRDB 運用ディレクトリ下にある場合は，同様にバックアップを取得してください。

(4) HiRDB Datareplicator と連携している場合

次に示すオペランドを追加，変更，又は削除する場合は HiRDB Datareplicator を終了してください。そして，オペランドの追加，変更，又は削除後に HiRDB Datareplicator を開始してください。

- pd_log_dual
- pd_log_max_data_size
- pdlogadfg -d sys
- pdlogadpf -d sys

HiRDB Datareplicator を稼働したまま，これらのオペランドを追加，変更，又は削除した場合，HiRDB Datareplicator による抽出が失敗することがあります。

1.1.6 UAP 環境定義の追加又は変更手順

UAP 環境定義の追加又は変更手順を次に示します。

〈手順〉

1. 変更対象の UAP 環境定義を使用する UAP が実行中でないか確認し，実行中の場合は UAP が終了するまで待ちます。UAP の実行中に UAP 環境定義を追加又は変更すると，実行中の UAP は変更前の UAP 環境定義が適用されます。ただし，タイミングによっては変更後の UAP 環境定義が適用されることがあります。
2. UAP 環境定義を追加又は変更します。
3. 追加又は変更した UAP 環境定義を使用して UAP を実行します。

1.2　HiRDB/パラレルサーバを利用される方へ

HiRDB 管理者は HiRDB の実行環境を設定するために HiRDB システム定義を作成します。ここで説明する項目は次のとおりです。

- HiRDB システム定義の種類
- HiRDB システム定義の作成方法
- HiRDB システム定義ファイルの構成
- 各定義の関係
- HiRDB システム定義（UAP 環境定義を除く）の変更手順
- UAP 環境定義の追加又は変更手順
- 影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成
- HiRDB システム定義ファイルの共用化

1.2.1　HiRDB システム定義の種類

HiRDB システム定義は次の表に示す定義で構成されています。HiRDB 管理者は各定義のオペランドの指定値を決めてください。

表 1–4　HiRDB システム定義の種類（HiRDB/パラレルサーバの場合）

定義の種類		内容
システム共通定義		HiRDB の構成及び共通情報を定義します。一つのユニットに一つ必要です。なお，**各ユニットのシステム共通定義の内容は同一にしてください。** システム共通定義のオペランドについては，「システム共通定義」を参照してください。
ユニット制御情報定義		ユニットの制御情報を定義します。一つのユニットに一つ必要です。ユニット制御情報定義のオペランドについては，「ユニット制御情報定義」を参照してください。
サーバ定義	サーバ共通定義	各サーバ定義のオペランドの仮定値を定義します。必要に応じて定義してください。サーバ共通定義のオペランドについては，「サーバ共通定義」を参照してください。
	フロントエンドサーバ定義	フロントエンドサーバの実行環境を定義します。一つのフロントエンドサーバに一つ必要です。フロントエンドサーバ定義のオペランドについては，「フロントエンドサーバ定義」を参照してください。
	ディクショナリサーバ定義	ディクショナリサーバの実行環境を定義します。一つのディクショナリサーバに一つ必要です。ディクショナリサーバ定義のオペランドについては，「ディクショナリサーバ定義」を参照してください。
	バックエンドサーバ定義	バックエンドサーバの実行環境を定義します。一つのバックエンドサーバに一つ必要です。バックエンドサーバ定義のオペランドについては，「バックエンドサーバ定義」を参照してください。

定義の種類	内容
UAP 環境定義	UAP の実行環境を定義します。必要に応じて定義してください。UAP 環境定義はフロントエンドサーバがあるユニットに作成します。マルチフロントエンドサーバの場合は，UAP 環境定義を適用したいフロントエンドサーバに定義してください。UAP 環境定義のオペランドについては，「UAP 環境定義」を参照してください。UAP 環境定義は最大 4096 個作成できます。
SQL 予約語定義	SQL 予約語削除機能使用時に必要です。SQL の予約語を定義します。フロントエンドサーバがあるユニットに定義します。マルチフロントエンドサーバの場合は，フロントエンドサーバがある各ユニットに同一内容の SQL 予約語定義が必要です。SQL 予約語削除機能については，pd_delete_reserved_word_file オペランドを参照してください。

1.2.2 HiRDB システム定義の作成方法

HiRDB 管理者は，次に示す方法で HiRDB システム定義を作成してください。

- OS のテキストエディタで作成

作成した HiRDB システム定義は，次の表に示すファイルに格納してください。なお，次の表に示すファイルを総称して **HiRDB システム定義ファイル**といいます。

表 1–5　HiRDB システム定義を格納するファイル（HiRDB/パラレルサーバの場合）

定義の種類	格納ファイル名
システム共通定義	$PDDIR/conf/pdsys
ユニット制御情報定義	$PDDIR/conf/pdutsys
サーバ共通定義	$PDDIR/conf/pdsvrc
フロントエンドサーバ定義	$PDDIR/conf/サーバ名※1
ディクショナリサーバ定義	$PDDIR/conf/サーバ名※1
バックエンドサーバ定義	$PDDIR/conf/サーバ名※1
UAP 環境定義	$PDDIR/conf/pduapenv/任意の名称※2
SQL 予約語定義	$PDDIR/conf/pdrsvwd/任意の名称※2

注※1

システム共通定義の pdstart オペランドの-s オプションで指定するサーバ名と同じにしてください。

注※2

- ファイル名称は先頭がアルファベットの英数字列（最大 8 文字）にしてください。
- UAP 環境定義又は SQL 予約語定義を使用するユーザに対して，それぞれのファイルが格納されているディレクトリの読み込み権限（r）と実行権限（x）を与えてください。また，ファイルには読み込み権限（r）を与えてください。

注意事項

HiRDB システム定義ファイルのパーミッションは，ファイルの所有者（HiRDB 管理者）にだけ，読み込み権限及び書き込み権限を持たせるように設定，維持するようにしてください。

(1)　OS のテキストエディタで作成する場合

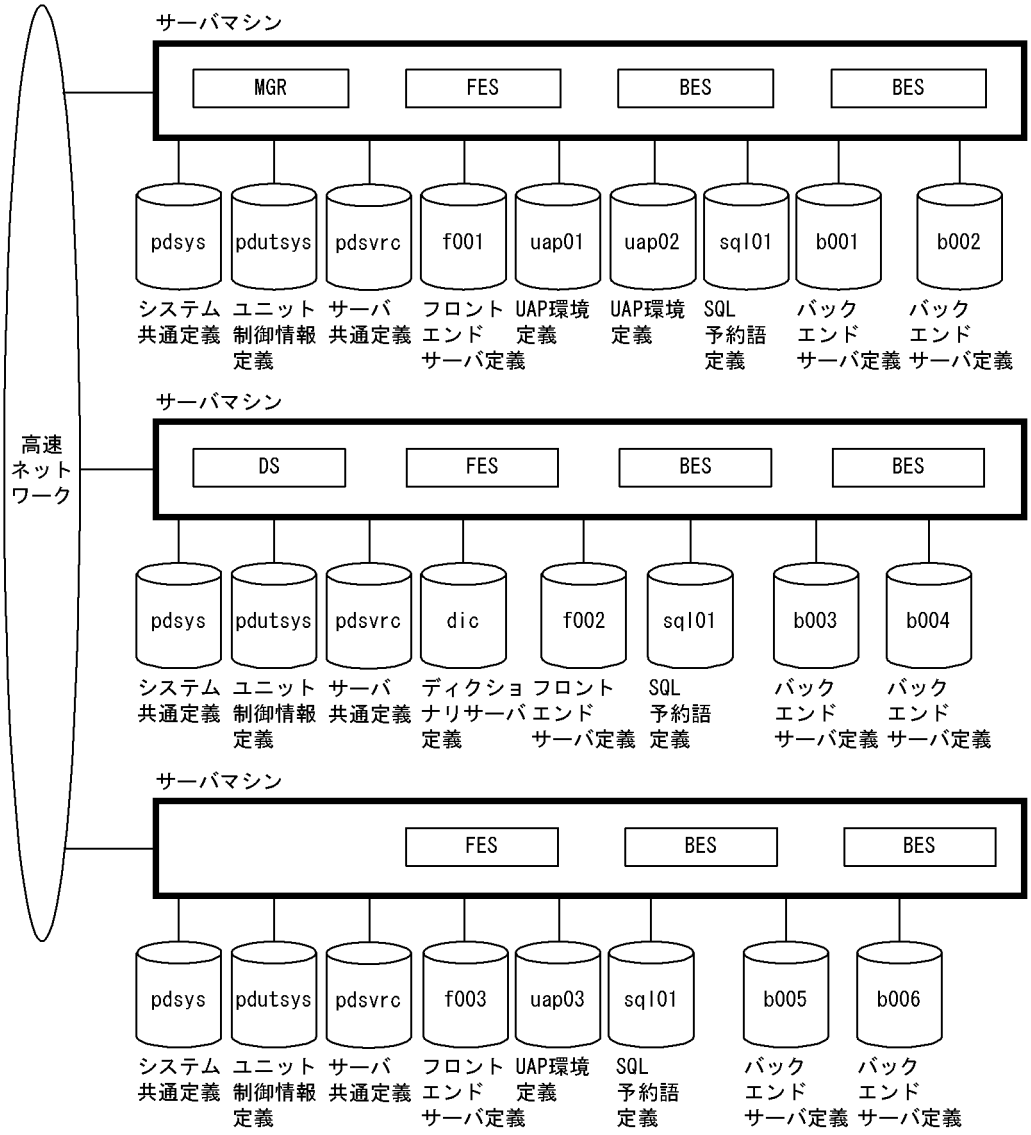
コマンドで HiRDB の環境設定をする場合は，OS のテキストエディタで HiRDB システム定義を作成します。HiRDB の実行環境に合わせて，HiRDB システム定義の各オペランドを指定してください。

なお，HiRDB システム定義を作成した後に，**pdconfchk** コマンドで HiRDB システム定義のオペランドの整合性をチェックしてください。チェック対象となるオペランドについては，「pdconfchk コマンドでチェックできるオペランド」を参照してください。

1.2.3　HiRDB システム定義ファイルの構成

OS のテキストエディタで HiRDB システム定義を作成する場合は，次の図を参考にして作成してください。

図 1–2　HiRDB システム定義ファイルの構成（HiRDB/パラレルサーバの場合）



注１　各サーバマシンのシステム共通定義は同一内容にしてください。
注２　各サーバマシンのSQL予約語定義は同一内容にしてください。

1.2.4　各定義の関係

異なる定義で同じオペランドを指定した場合，HiRDB は次に示す優先順位に従って指定値を決定します。

1. フロントエンドサーバ定義，ディクショナリサーバ定義，バックエンドサーバ定義
2. サーバ共通定義
3. ユニット制御情報定義
4. システム共通定義

優先順位の低い定義で標準となる値を指定して，優先順位の高い定義で値を変更できます。例えば，システム共通定義で標準となる値を指定し，各サーバ定義でサーバごとにその値を変更できます。

(1)　クライアント環境定義との関係

次の表に示す HiRDB システム定義のオペランドの値は，クライアント環境定義でクライアントごとに変更できます。

表 1–6　クライアント環境定義で指定値を変更できるオペランド（HiRDB/パラレルサーバの場合）

HiRDB システム定義のオペランド	クライアント環境定義のオペランド	オペランドの内容
pd_additional_optimize_level	PDADDITIONALOPTLVL	SQL 拡張最適化オプションを指定します。
pd_cwaittime_wrn_pnt	PDCWAITTIMEWRNPNT	SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。
pd_delete_reserved_word_file	PDDELRSVWDFILE	SQL 予約語削除機能を使用する場合，SQL 予約語削除ファイルの名称を指定します。
pd_ha_transaction	PDHATRNQUEUING	トランザクションキューイング機能を使用するかどうかを指定します。
pd_hash_table_size	PDHASHTBLSIZE	SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシュ表サイズを指定します。
pd_optimize_level	PDSQLOPTLVL	SQL 最適化オプションを指定します。
pd_space_level	PDSPACELVL	空白変換レベルを指定します。
pd_uap_exerror_log_param_size	PDUAPEXERLOGPRMSZ	クライアントエラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。
pd_uap_exerror_log_use	PDUAPEXERLOGUSE	拡張 SQL エラー情報出力機能を使用するかどうかを指定します。
pd_watch_pc_client_time	PDSWATCHTIME	Windows 対応の HiRDB クライアントからの要求待ち時間を指定します。
pd_isolation_level	PDISLLVL	データ保証レベルを指定します。

クライアントごとに値を変更する場合は，クライアント環境定義の該当するオペランドを指定してください。クライアント環境定義については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

1.2.5　HiRDB システム定義（UAP 環境定義を除く）の変更手順

(1)　HiRDB システム定義の変更手順

HiRDB システム定義の変更手順を次に示します。なお，$PDDIR/conf はユニット制御情報定義ファイルを格納しているディレクトリを意味しています。$PDCONFPATH はそれ以外の HiRDB システム定義ファイルを格納しているディレクトリを意味しています。

〈手順〉

1. $PDDIR/conf 及び$PDCONFPATH 下にサブディレクトリを作成します。この例ではサブディレクトリとして work を作成します。
2. ユニット制御情報定義ファイルを$PDDIR/conf/work 下にコピーします。そのほかの HiRDB システム定義ファイルを$PDCONFPATH/work 下にコピーします。
3. $PDDIR/conf/work 及び$PDCONFPATH/work 下にコピーした HiRDB システム定義を変更します。
4. **pdconfchk -d work** コマンドで，$PDDIR/conf/work 及び$PDCONFPATH/work 下の HiRDB システム定義の内容をチェックします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdstop** コマンドで HiRDB を正常終了します。
6. **pdlogunld** コマンドで，アンロード待ち状態のシステムログファイルをアンロードします。
7. 3 で変更した HiRDB システム定義ファイルを$PDDIR/conf 又は$PDCONFPATH 下にコピーして，HiRDB システム定義ファイルを置き換えます。
8. 次に示すオペランドの指定値を変更した場合は，**pdloginit** コマンドでシステムログファイルを初期化します。
   - pd_log_dual
   - pdstart
9. **pdstart** コマンドで HiRDB を正常開始します。

(2)　システム構成変更コマンドを使用した HiRDB システム定義の変更手順

システム構成変更コマンド（pdchgconf コマンド）を使用すると，HiRDB の稼働中に HiRDB システム定義を変更できるため，HiRDB を正常終了する必要はありません。ただし，このコマンドを使用する場合は HiRDB Advanced High Availability が必要になります。システム構成変更コマンドを使用した HiRDB システム定義の変更手順を次に示します。

〈手順〉

1. $PDDIR/conf/chgconf ディレクトリを作成します。
2. 使用中の HiRDB システム定義ファイルを 1 で作成したディレクトリ下にコピーします。
3. $PDDIR/conf/chgconf 下の HiRDB システム定義を変更します。

4. **pdconfchk -d chgconf** コマンドで，$PDDIR/conf/chgconf 下の HiRDB システム定義のチェックをします。エラーがある場合は HiRDB システム定義を修正して，再度 pdconfchk コマンドを実行してください。
5. **pdchgconf** コマンドで，HiRDB システム定義を変更後の HiRDB システム定義に置き換えます。
   pdchgconf コマンドを実行すると，使用中（変更前）の HiRDB システム定義ファイルが$PDDIR/conf/backconf 下に退避されます。そして，$PDDIR/conf/chgconf 下の変更後の HiRDB システム定義ファイルが$PDDIR/conf 下にコピーされます。

注意事項

- pdchgconf コマンドの入力後，15 分以上トランザクション又はユティリティが動き続けた場合，pdchgconf コマンドが異常終了します。
- システム構成変更コマンドを使用した HiRDB システム定義の変更には制限事項があります。制限事項については，マニュアル「HiRDB システム運用ガイド」を参照してください。

(3) 注意事項

- システム共通定義を修正する場合はすべてのサーバマシンのシステム共通定義を同じように修正してください。
- 正常終了又は計画停止した場合に一部のユニットが異常終了したときは，次回開始前に HiRDB システム定義を変更しないでください。変更すると，HiRDB の開始に失敗します。失敗しなくても，開始後に正常に稼働できなくなります。
- HiRDB/パラレルサーバの場合はユニットごとに$PDDIR/conf 及び$PDCONFPATH 下にサブディレクトリを作成して，HiRDB システム定義の内容をチェックしてください。
- 稼働中の HiRDB が使用している HiRDB システム定義は変更又は削除しないでください。変更又は削除した場合はその HiRDB の動作を保証できません。
- HiRDB が計画停止，強制終了，又は異常終了した場合，HiRDB システム定義のオペランドで変更できるものと変更できないものがあります。詳細については，「オペランドの一覧」を参照するか，又は各定義文のオペランド一覧を参照してください。
- HiRDB システム定義を変更した後に，$PDDIR/conf 下のファイルのバックアップを取得してください。HiRDB 運用ディレクトリがあるディスクの障害などに備えて，HiRDB 運用ディレクトリ下のファイル（$PDDIR/conf 下のファイル）のバックアップを取得します。HiRDB 運用ディレクトリを回復するには，$PDDIR/conf 下のファイルのバックアップが必要になります。また，$PDCONFPATH が HiRDB 運用ディレクトリ下にある場合は，同様にバックアップを取得してください。
- 1：1 スタンバイレス型系切り替え機能使用時の注意事項です。正規 BES ユニットの HiRDB システム定義を変更する場合は，正規 BES ユニット及び代替 BES ユニットを pdstop -u コマンドで正常終了してから変更してください。HiRDB システム定義の変更後，正規 BES ユニットのユニット制御情報定義ファイル及びバックエンドサーバ定義ファイルを代替 BES ユニットにコピーしてください。詳細については，マニュアル「HiRDB システム運用ガイド」の「HiRDB システム定義の作成〔系切り替え機能〕」を参照してください。

(4) HiRDB Datareplicator と連携している場合

次に示すオペランドを追加，変更，又は削除する場合は HiRDB Datareplicator を終了してください。そして，オペランドの追加，変更，又は削除後に HiRDB Datareplicator を開始してください。

- pd_log_dual
- pd_log_max_data_size
- pdlogadfg -d sys
- pdlogadpf -d sys

HiRDB Datareplicator を稼働したまま，これらのオペランドを追加，変更，又は削除した場合，HiRDB Datareplicator による抽出が失敗することがあります。

1.2.6 UAP 環境定義の追加又は変更手順

UAP 環境定義の追加又は変更手順を次に示します。

〈手順〉

1. 変更対象の UAP 環境定義を使用する UAP が実行中でないか確認し，実行中の場合は UAP が終了するまで待ちます。UAP の実行中に UAP 環境定義を追加又は変更すると，実行中の UAP は変更前の UAP 環境定義が適用されます。ただし，タイミングによっては変更後の UAP 環境定義が適用されることがあります。
2. UAP 環境定義を追加又は変更します。
3. 追加又は変更した UAP 環境定義を使用して UAP を実行します。

1.2.7 影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成

影響分散スタンバイレス型系切り替え機能使用時の HiRDB システム定義の作成方法と注意事項を説明します。

ポイント

ホスト BES が稼働するユニットとゲスト BES が稼働するユニットのバックエンドサーバの稼働環境を合わせる必要があります。

(1) システム共通定義

全ユニットで同じシステム共通定義を使用します。作成したシステム共通定義をコピーしてください。

注意事項

バックエンドサーバ定義の省略値として指定するオペランドについては，システム共通定義に指定してください。例えば，pd_sql_object_cache_size オペランドが該当します。このオペランドをサーバ共通定義だけに指定することはできません。

(2)　ユニット制御情報定義

影響分散スタンバイレス型系切り替え機能を適用するユニットのユニット制御情報定義に指定できるオペランドは限定されています。ユニット制御情報定義に指定できるオペランドについては，「影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義）」を参照してください。

ユニット制御情報定義に指定できないオペランドを指定する必要がある場合は，システム共通定義又はバックエンドサーバ定義に指定してください。

注意事項

次に示すオペランドについては，HA グループを構成するユニットのユニット制御情報定義で同じ指定をする必要があります。

- ユニット制御情報定義とバックエンドサーバ定義の両方に指定できるオペランド
- ユニット制御情報定義のオペランドでバックエンドサーバの動作に影響を与えるオペランド

(3)　サーバ共通定義

HA グループを構成するユニットは同じサーバ共通定義を使用します。作成したサーバ共通定義をコピーしてください。

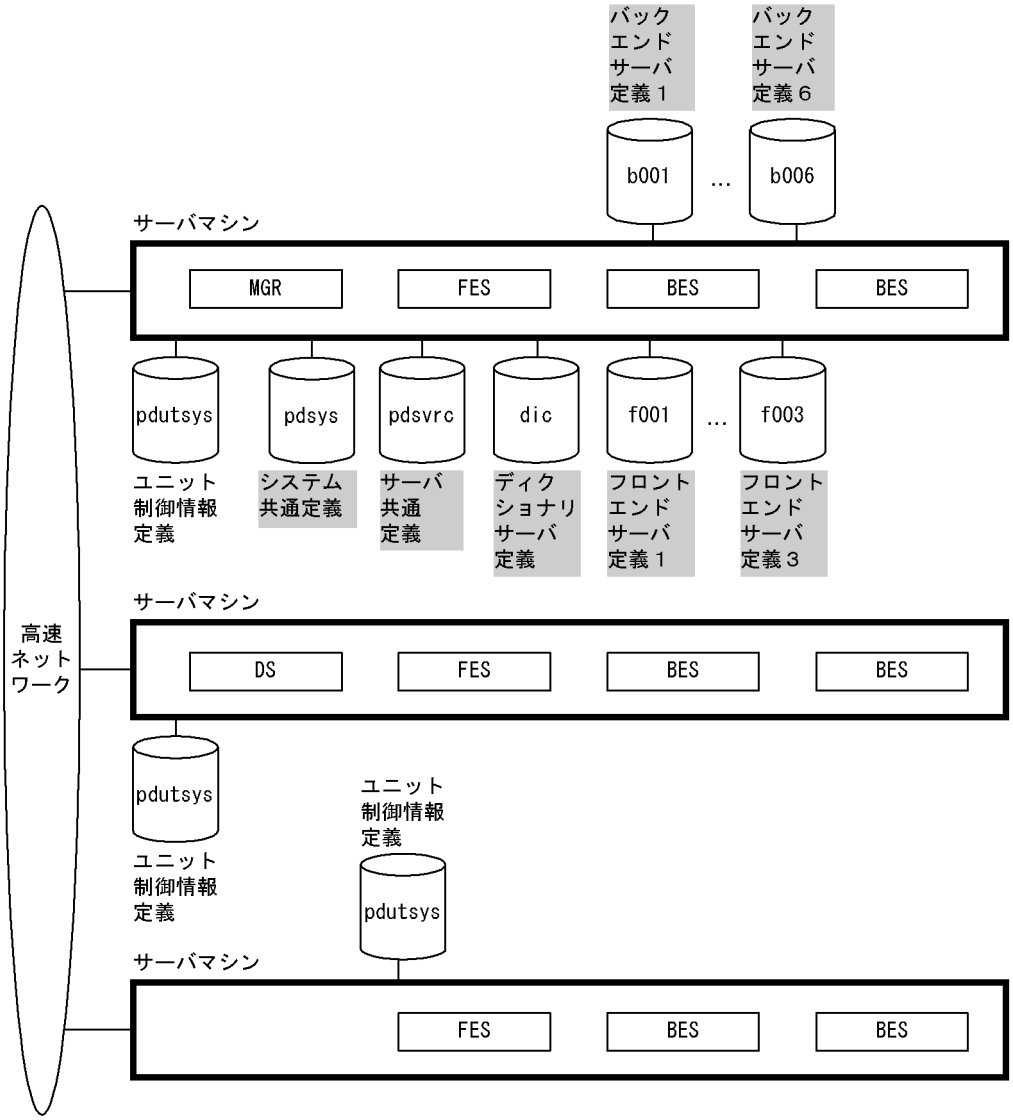
(4)　バックエンドサーバ定義

HA グループを構成する各ユニットにホスト BES のバックエンドサーバ定義を定義します。作成したバックエンドサーバ定義をコピーしてください。

1.2.8　HiRDB システム定義ファイルの共用化

OS のファイル共用機能（NFS）を利用すると，ユニット制御情報定義ファイルを除いた HiRDB システム定義ファイルを一つのサーバマシンで管理できます。これを **HiRDB システム定義ファイルの共用化**といいます。HiRDB システム定義ファイルの共用化を次の図に示します。

図 1–3　HiRDB システム定義ファイルの共用化



〔説明〕

網掛け部分が共用化したファイルです。

(1)　共用化すると便利なこと

共用化していない場合	共用化している場合
HiRDB システム定義をサーバマシンごとに管理する必要があります。	HiRDB システム定義を一つのサーバマシンで管理できます。ただし，ユニット制御情報定義を除きます。
システム共通定義を変更する場合，サーバマシンの数だけ修正が必要となります。例えば，四つのサーバマシンで構成されている場合，四つのシステム共通定義があります。システム共通定義の内容は同じである必要があるため，四つとも修正が必要となります。	• システム共通定義が一つになるため，修正が１回で済みます。 • 修正回数が少なくなるため，修正誤りの可能性が少なくなります。 • HiRDB が共用ディレクトリを参照できなくなると，HiRDB が異常終了することがあります。また，ネットワークを介したファイルアクセスであるため，性能遅延，データ欠損

共用化していない場合	共用化している場合
	などが発生する場合があり，信頼性の面から使用を推奨しません。

(2) 共用化の手順

〈手順〉

1. 共用化する HiRDB システム定義ファイルを格納する任意のディレクトリを作成します。このディレクトリは，HiRDB システム定義ファイルを共通管理するサーバマシンに作成してください。このディレクトリを**共用ディレクトリ**といいます。共通管理するサーバマシンは任意ですが，システムマネジャを定義するサーバマシンで管理することをお勧めします。
2. 共用ディレクトリ（$PDCONFPATH）下にユニット制御情報定義ファイルを除いた HiRDB システム定義ファイルを作成します。UAP 環境定義ファイルは$PDCONFPATH/pduapenv 下に作成してください。
3. ユニット制御情報定義ファイルを各サーバマシンの$PDDIR/conf 下に作成します。このとき，**PDCONFPATH** オペランドに共用ディレクトリ名称を指定します。

1.3　HiRDB システム定義を記述するときの文法規則

ここでは，HiRDB システム定義のオペランドを記述するときの文法規則について説明します。

1.3.1　オペランドの指定形式

HiRDB システム定義のオペランドは次に示す三つの形式で記述します。

set 形式

オペランドに値を設定します。指定例を次に示します。

set pd_max_users=15

コマンド形式

オペランドにオプション及びコマンド引数を設定します。指定例を次に示します。

pdlogadfg -d sys -g loggrp01 ONL

pdlogadfg：コマンド名

-d sys -g loggrp01：オプション

マイナス記号で始まる文字列です。フラグ引数を指定しない形式 1 と，フラグ引数を指定する形式 2 の二つがあります。

形式 1　−オプションフラグ

形式 2　−オプションフラグ　フラグ引数

オプションフラグ：マイナス記号に続く 1 文字の英字です。英大文字と英小文字は区別されます。

フラグ引数：オプションフラグに対する操作対象です。

ONL：コマンド引数

マイナス記号以外で始まる引数です。

putenv 形式

オペランドに環境変数と環境変数値を設定します。指定例を次に示します。

putenv SHMMAX 16

1.3.2　コメントの記述

各オペランドにコメントを記述できます。コメントを記述する場合は，コメントの先頭に"#"を記述してください。"#"を記述するとその行の終わりまでがコメントとしてみなされます。行の先頭に"#"を記述した場合は，1 行全体がコメント扱いになります。

入力例

```
set pd_max_users=15 # 最大同時接続数
pdlogadfg -d sys -g loggrp01 ONL
```

システムログファイルのファイルグループの定義

1.3.3　行の継続

定義の 1 行の長さは最大 80 文字です。80 文字を超える指定の場合は，複数行に分けて記述してください。このとき，行の末尾には，継続記号である"¥"を記述します。

入力例

```
pdbuffer -a buffer ABC -n 160 -r rdareaA,rdareaB,……¥
rdareaZ
```

なお，コメントを記述した行には，継続行を続けられません。"#"を記述すると，その行の終わりまでがコメントとみなされるため，"¥"を記述しても行の継続はできません。

1.3.4　オペランドの重複指定について

1 定義文（1 ファイル）にオペランドを重複指定した場合，HiRDB は次に示す規則に従って処理をします。

- set 形式又は putenv 形式の場合は，重複指定したオペランドのうち一番最後に指定したオペランドの値が有効になります。
- コマンド形式の場合はオペランドによって異なります。各オペランドの説明を参照してください。

1.3.5　注意事項

- HiRDB システム定義に指定している putenv 形式のオペランドの指定値をデフォルト値に変更する場合は，オペランドを削除しないで，デフォルト値を設定してください。

1.4　バージョンによって省略値が異なるオペランド，及び指定不要になったオペランド

1.4.1　バージョン，リビジョンによる HiRDB システム定義の変更点

バージョン 09-50 では，HiRDB システム定義の省略値を見直し，省略値の変更及び指定が必要なオペランド数の大幅な削減を実施しました。これによって，より安全なシステムを構築できます。ここでは，バージョン 09-50 以降で省略値が異なるオペランド，指定できる範囲が変更となるオペランド，及び指定不要となったオペランドの一覧を示します。

バージョン 09-50 より前のバージョンからバージョンアップする場合は，省略値変更によるメリット及びデメリットを確認してください。確認の結果，旧バージョンとの互換性を重視する場合は，旧バージョンと同等の省略値になる互換モードを適用してください。ただし，この場合は推奨値を指定できるオペランドがあれば，個別に指定することを検討してください。

バージョンアップに関する手順及び注意事項は，マニュアル「HiRDB システム導入・設計ガイド」の「HiRDB のバージョンアップ」を参照してください。

(1)　バージョン，リビジョンによって省略値が異なるオペランド

バージョン，リビジョンによって省略値が異なるオペランドを次の表に示します。

注意事項

太字のオペランドを省略している場合，バージョンアップ時に推奨モードを適用し，デメリットに記載されている HiRDB ファイルの再作成などを行わないと，HiRDB が開始できなくなるおそれがあります。省略値変更によるメリット及びデメリットを確認の上，オペランド省略時動作のモードを決定してください。

表 1–7　バージョン 09-60 で省略値が変更となるオペランド

オペランド名	バージョン 09-60 以降の省略値 (推奨モード適用時)	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-60 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
pd_clt_waittime_over_dump_level	shm_fesonly	shm_fesonly	all	共用メモリダンプ出力による I/O 負荷を軽減	なし

表 1–8　バージョン 09-50 で省略値が変更となるオペランド

オペランド名	バージョン 09-50 以降の省略値 （推奨モード適用時）	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
pd_log_rec_leng	1,024	→	4,096	システムログファイルの格納効率向上	システムログファイルのレコード長変更によるシステムログファイルの再作成
pd_lck_until_disconnect_cnt	【ディクショナリサーバ】 256 【上記以外のサーバ】 10,000	→	256	• ユティリティの同時実行数の省略値拡大によるオペランド指定不要化※7 • ユティリティが使用する RD エリア数の省略値拡大によるオペランド指定不要化※7	• シンクポイントダンプファイルの容量増加によるシンクポイントダンプファイルの再作成 • 共用メモリサイズ増加による共用メモリの再見積もり※12
pd_max_file_no	2,000	→	400	HiRDB ファイル数の省略値拡大によるオペランド指定不要化※5	• ステータスファイルの容量増加によるステータスファイルの再作成 • プロセス固有メモリ増加によるプロセス固有メモリの再見積もり • 共用メモリサイズ増加による共用メモリの再見積もり※12
pd_max_rdarea_no	1,000	→	200	RD エリア数の省略値拡大によるオペランド指定不要化※5	• ステータスファイルの容量増加によるステータスファイルの再作成 • プロセス固有メモリ増加によるプロセス

オペランド名	バージョン 09-50 以降の省略値 （推奨モード適用時）	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
					固有メモリの再見積もり • 共用メモリサイズ増加による共用メモリの再見積もり※12
pd_max_add_dbbuff_shm_no	【32 ビットモード】 500 ＋（512−a） 【64 ビットモード】 1000 ＋（512−a）	【32 ビットモード】 500 ＋（512−a） 【64 ビットモード】 1000 ＋（512−a）	【32 ビットモード】 500 ＋（16−a） 【64 ビットモード】 1000 ＋（16−a）	pdbufmod コマンドを運用で使用している場合，動的追加用共用メモリセグメント数の上限拡大によるオペランド指定不要化	pdbufmod コマンドを運用で使用している場合 • ステータスファイルの容量増加によるステータスファイルの再作成 • 共用メモリサイズ増加による共用メモリの再見積もり※12
pd_utl_exec_mode	1	→	0	ユティリティの最大同時実行数の拡張指定に伴うオペランド指定不要化	• ユティリティ最大同時実行数の見直し※10 • 共用メモリサイズ増加による共用メモリの再見積もり※12
pd_additional_optimize_level	"COST_BASE_2", "APPLY_JOIN_COND_FOR_VALUE_EXP", "APPLY_SRCH_COND_FOR_VALUE_EXP", "MERGE_FROM_DERIVED_TABLE", "CONVERT_OUTER_INNER_JOIN"	→	"COST_BASE_2"	アクセスパス変更による SQL の実行性能の向上	アクセスパス変更によるインデクス及び SQL の見直し※6
pd_optimize_level	【HiRDB/シングルサーバ】 "PRIOR_NEST_JOIN", "PRIOR_OR_INDEXES",	→	【HiRDB/シングルサーバ】 "PRIOR_NEST_JOIN", "PRIOR_OR_INDEXES",	アクセスパス変更による SQL の実行性能の向上	アクセスパス変更によるインデクス

オペランド名	バージョン 09-50 以降の省略値 (推奨モード適用時)	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
	"DETER_AND_INDEXES", "RAPID_GROUPING", "DETER_WORK_TABLE_FOR_UPDATE", "APPLY_ENHANCED_KEY_COND", "MOVE_UP_DERIVED_COND" 【HiRDB/パラレルサーバ】 "PRIOR_NEST_JOIN", "PRIOR_OR_INDEXES", "SORT_DATA_BES", "DETER_AND_INDEXES", "RAPID_GROUPING", "DETER_WORK_TABLE_FOR_UPDATE", "APPLY_ENHANCED_KEY_COND", "MOVE_UP_DERIVED_COND", "FLTS_ONLY_DATA_BES"		"DETER_AND_INDEXES", "RAPID_GROUPING", "DETER_WORK_TABLE_FOR_UPDATE", "APPLY_ENHANCED_KEY_COND" 【HiRDB/パラレルサーバ】 "PRIOR_NEST_JOIN", "PRIOR_OR_INDEXES", "SORT_DATA_BES", "DETER_AND_INDEXES", "RAPID_GROUPING", "DETER_WORK_TABLE_FOR_UPDATE", "APPLY_ENHANCED_KEY_COND"		及び SQL の見直し※6
pd_apply_search_ats_num	30,000	→	255	アクセスパス変更による SQL の実行性能の向上	SQL オブジェクトキャッシュサイズの再見積もり
pd_hashjoin_hashing_mode	TYPE2	→	TYPE1	アクセスパス変更による SQL の実行性能の向上	ハッシュ表サイズの再見積もり
pd_sql_dec_op_maxprec	38	→	29	演算結果の精度向上	DECIMAL 型の演算を含む場合 • ビュー表，トリガ，ルーチン，検査制約を含む表定義の再作成 • アプリケーションの改修

オペランド名	バージョン 09-50 以降の省略値 （推奨モード適用時）	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
pd_leap_second	Y	→	N	うるう秒使用時のオペランド指定不要化	時刻の値の範囲の変更に伴うアプリケーションの改修
pd_standard_sqlstate	Y	→	N	SQLSTATE の細分化による障害要因の特定容易化	SQLSTATE の値の変更に伴うアプリケーションの改修
pd_trn_rcvmsg_store_buffle n	8,192	→	4,096	トランザクション回復時の性能安定化	• プロセス数の増加によるオペレーティングシステムパラメタの再見積もり • 共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_assurance_index_no	2500	→	500	インデクス数の省略値拡大によるオペランド指定不要化※5	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_auth_cache_size	10	→	1	ユーザ権限情報用バッファサイズの省略値拡大によるオペランド指定不要化※5	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_lck_pool_size	【32 ビットモード】 16,000 【64 ビットモード】 128,000	→	【32 ビットモード】 16,000 【64 ビットモード】 32,000	• トランザクション実行時の排他資源要求数の省略値拡大によるオペランド指定不要化※8 • トランザクションの同時実行数の省略値拡大によるオペランド指定不要化※8	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_max_temporary_object_no	600	→	0	一時表で使用できるインデクスの省	共用メモリサイズ増加による共用メ

オペランド名	バージョン 09-50 以降の省略値 (推奨モード適用時)	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
				略値拡大による指定不要化※9	モリサイズの再見積もり※12
pd_routine_def_cache_size	300	→	100	ルーチン定義情報用バッファサイズの省略値拡大による性能安定化※5	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_sql_object_cache_size	【32 ビットモード】 (pd_max_users の値＋ 3※11) ×22 キロバイト 【64 ビットモード】 (pd_max_users の値＋ 3※11) ×40 キロバイト	→	(pd_max_users の値＋ 3※11) ×22 キロバイト	SQL オブジェクトを格納するバッファサイズの省略値拡大による性能安定化	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_table_def_cache_size	【32 ビットモード】 ↑√pd_max_users＋3※11↑ ×100 【64 ビットモード】 16,000	→	↑√pd_max_users＋3※11↑ ×100	表定義情報を格納するバッファサイズの省略値拡大による性能安定化※5	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_view_def_cache_size	【32 ビットモード】 ↑√(pd_max_users＋3※11)↑ ×8 【64 ビットモード】 30000	→	↑√(pd_max_users＋3※11)↑ ×8	表定義情報を格納するバッファサイズの省略値拡大による性能安定化※5	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_log_max_data_size※1	523,000	→	400,000	システムログの入出力に使用するバッファサイズの省略値拡大による性能安定化	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_log_rollback_buff_count	0	→	省略	ロールバック処理で使用するバッファ面数の省略値拡大によるロールバック時間の短縮	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_max_tmp_table_rdarea_no	min(131088,pd_max_rdarea_no-3)	→	0	一時表使用時のオペランド指定不要化	共用メモリサイズ増加による共用メモリサイズの再見積もり※12
pd_uap_exerror_log_use	AUTO	AUTO	NO	障害箇所の特定容易化	• 共用メモリサイズ増加によ

オペランド名	バージョン 09-50 以降の省略値 （推奨モード適用時）	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
pd_uap_exerror_log_dir	$PDDIR/spool/pduaperr	$PDDIR/spool/pduaperr	なし		る共用メモリサイズの再見積もり※12 • 運用ディレクトリのディスク所要量増加
pd_uap_exerror_log_size	10,000,000	10,000,000	1,000,000		
pd_cwaittime_wrn_pnt	1800sec,auto	1800sec,auto	0	SQL 実行時間の警告による障害の早期発見	• 警告メッセージ（KFPA20009-W）追加によるメッセージ監視の検討 • 運用ディレクトリのディスク所要量の増加
pd_cwaittime_report_size	10,000,000	10,000,000	100,000		
pd_cwaittime_report_dir	$PDDIR/spool/pdcwwrn	$PDDIR/spool/pdcwwrn	なし		
pd_mlg_file_size	10,240	10,240	1,024	メッセージログファイルサイズの拡大によるトラブルシュート情報の拡充	運用ディレクトリのディスク所要量の増加
pd_statistics	A	→	N	統計情報取得によるトラブルシュート情報の拡充	運用ディレクトリのディスク所要量の増加
pd_stj_file_size	102,400	→	1,024		
pd_watch_resource	DEFAULT	→	MANUAL	システム稼働状況の取得	警告メッセージ（KFPS05123-W 及び KFPH22023-W）追加によるメッセージ監視の検討
pd_sts_initial_error※2	excontinue	→	stop	障害時の影響局所化	片系障害時の運用変更
pd_syssts_initial_error※3	excontinue	→	stop	障害時の影響局所化	片系障害時の運用変更
pd_rdarea_extension_timing	use	→	nouse	RD エリア自動増分の契機変更による自動増分エラー時の業務継続化	RD エリア自動増分の契機変更による HiRDB ファイルシステム領域の再見積もり

オペランド名	バージョン 09-50 以降の省略値 （推奨モード適用時）	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
pd_registered_port_level	1	→	0	HiRDB 予約ポート機能使用時の自動割り当てポート使用数削減	HiRDB 予約ポート機能使用時の予約ポートの範囲の変更
pd_dbsync_altwrite_skip	Y	→	N	参照トランザクションの性能安定化	なし
pd_deter_restart_on_stop_fail	Y	Y	N	強制停止実行時の異常終了発生による系切り替えの抑止	なし
pd_ipc_tcp_nodelayack	Y	Y	N	即時 ACK の適用による通信性能の安定化	なし
pd_max_dbbuff_shm_no（Windows 版 32bit 限定オペランド）	512	512	16	グローバルバッファ用共用メモリセグメント数の設定不要化	なし
pd_inet_unix_bufmode※4	os	→	【Linux の場合】 os 【AIX の場合】 conf	通信バッファサイズの拡大による性能安定化	なし
pd_sql_simple_comment_use	Y	→	N	単純注釈使用時のオペランド指定不要化	SQL 文中の「--」の扱いの変更と，改行コードを含む制御文字の扱いの変更に伴う，SQL 文の見直し
pd_trn_rollback_msg_interval	5	→	60	メッセージ出力間隔の短縮	なし
pd_module_trace_max	256	256	126	モジュールトレース拡大によるトラブルシュート情報の拡充	プロセス固有メモリ増加によるプロセス固有メモリの再見積もり
pd_utl_file_buff_size	1024	1024	32	入出力バッファの拡大によるユティリティの実行性能の向上	なし

オペランド名	バージョン 09-50 以降の省略値 （推奨モード適用時）	0904 互換モード適用時の省略値	バージョン 08-00 からバージョン 09-50 より前の省略値	省略値変更によるメリット	省略値変更によるデメリット
SHMMAX	【32 ビットモード】 100 【64 ビットモード】 1,024	【32 ビットモード】 100 【64 ビットモード】 1,024	【32 ビットモード】 200 【64 ビットモード】 1,024	メモリの効率利用によるシステム安定稼働	なし

（凡例）

→：

バージョン 08-00 からバージョン 09-50 より前の省略値と同じ省略値であることを示します。

a：

HiRDB 開始時に各サーバに割り当てられている共用メモリセグメント数です。共用メモリセグメントの情報は pdls -d mem コマンド，又は OS のコマンドで確認できます。

b：

16 になります。

注※1

推奨モードを適用し，pd_log_max_data_size オペランドの値が 523000 未満の場合，pdconfchk コマンドで警告（KFPS05632-W メッセージ）を出力します。

注※2

pd_sts_initial_error オペランドは，pd_sts_singleoperation オペランドと関連があります。pd_sts_initial_error オペランドの指定値を検討する場合は pd_sts_singleoperation オペランドも併せて検討してください。

注※3

pd_syssts_initial_error オペランドは pd_syssts_singleoperation オペランドと関連があります。pd_syssts_initial_error オペランドの指定値を検討する場合は pd_syssts_singleoperation オペランドも併せて検討してください。

注※4

pd_inet_unix_bufmode オペランドは次に示すオペランドと関連があります。pd_inet_unix_bufmode オペランドの指定値を検討する場合は併せて検討してください。

- pd_inet_unix_utl_bufmode
- pd_ipc_inet_bufsize
- pd_ipc_unix_bufsize
- pd_listen_socket_bufset
- pd_tcp_inet_bufsize

- pd_tcp_unix_bufsize
- pd_utl_buff_size

注※5

「省略値で定義できるデータベースの規模」に記載されているデータベースの規模を超える場合は，オペランドの指定が必要な場合があります。

注※6

詳細は，マニュアル「HiRDB コマンドリファレンス」の「アクセスパス表示ユティリティ（pdvwopt）」の「チューニングの考え方」を参照してください。

注※7

ユティリティ数やユティリティが使用する RD エリア数が pd_lck_until_disconnect_cnt オペランドの省略値を超える場合は，pd_lck_until_disconnect_cnt オペランドの指定が必要です。詳細は，「1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式」を参照してください。

注※8

使用する排他制御用プールサイズが pd_lck_pool_size オペランドの省略値を超える場合は，pd_lck_pool_size オペランドの指定が必要です。詳細は，「排他資源数の見積もり」を参照してください。

注※9

一時表で使用するインデクスの数が 600 を超える場合は，このオペランドを指定してください。

注※10

pd_max_users オペランドの指定値が 32 未満でデータベース回復ユティリティ（pdrstr）又はデータベース複写ユティリティ（pdcopy）を多重で実行している場合，ユティリティの最大同時実行数を減らす必要がある場合があります。詳細は，マニュアル「HiRDB コマンドリファレンス」の「ユティリティの最大同時実行数」のデータベース回復ユティリティ（pdrstr）又はデータベース複写ユティリティ（pdcopy）の最大同時実行数を参照してください。

注※11

システムに複数のフロントエンドサーバがある場合は 4 となります。

注※12

共用メモリサイズが増加すると，ページフォルトの発生回数が増加し，トランザクション性能に影響を与えるおそれがあります。各オペランドの指定値の目安を参照し，システムに合わせて適切な値を指定することを検討してください。

(2) バージョン，リビジョンによってユーザ指定値の指定範囲が異なるオペランド

バージョン，リビジョンによってユーザ指定値の指定範囲が異なるオペランドを次の表に示します。バージョン 09-50 以降でバージョン 09-50 より前の指定範囲で指定した場合，その指定値で動作しますが，バージョン 09-50 以降の指定範囲で指定することを検討してください。

表 1–9　バージョン 09-50 でユーザ指定値の範囲が異なるオペランド

オペランド名	バージョン 09-50 以降の指定範囲	バージョン 09-50 より前の指定範囲	指定範囲変更によるメリット	指定範囲変更によるデメリット
pd_log_write_buff_count※	10～65000《10》	3～65000《10》	省略値で使用できるログバッファ面数の拡大による性能安定化	共用メモリサイズ増加によるメモリ計算式の再見積もり

注※

推奨モードを適用し，pd_log_write_buff_count オペランドに 10 未満の値を指定している場合，pdconfchk コマンドを実行すると警告メッセージ（KFPS05632-W）が出力されます。

(3)　指定不要となったオペランド

省略値以外を指定する必要がないために指定不要となったオペランド，及び省略値以外を指定している場合に特に指定値の変更をお勧めするオペランドの一覧を次の表に示します。これらのオペランドを指定したままでもエラーにはなりませんが，オペランドを省略，又は指定値を変更することを検討してください。

推奨モード適用時は，ここで示すオペランドについて，すべて省略することを検討してください。

互換モード適用時は，バージョン 09-50 の省略値と 0904 互換モード適用時の省略値が異なるオペランドについて，バージョン 09-50 の省略値になるように明示的に指定することを検討してください。オペランドの説明は，「互換モード適用時に検討が必要なオペランド」を参照してください。また，バージョン 09-50 の省略値と 0904 互換モード適用時の省略値が同じオペランドは，省略することを検討してください。

指定値変更によるメリット及びデメリットについては，「バージョン，リビジョンによって省略値が異なるオペランド」を参照してください。

注意事項

太字のオペランドを省略している場合，バージョンアップ時に推奨モードを適用し，デメリットに記載されている HiRDB ファイルの再作成などを行わないと，HiRDB が開始できなくなるおそれがあります。

表 1–10　バージョン 09-50 で指定不要となったオペランド

オペランド名	バージョン 09-50 の省略値（推奨モード適用時）	0904 互換モード適用時の省略値	オペランドの説明有無
pd_log_rec_leng	1,024	4,096	○
pd_sysdef_default_option※1	→	recommendable	－
pd_large_file_use※2，※3	→	Y	－
pd_apply_search_ats_num	30,000	255	○

オペランド名	バージョン 09-50 の省略値（推奨モード適用時）	0904 互換モード適用時の省略値	オペランドの説明有無
pd_hashjoin_hashing_mode	TYPE2	TYPE1	○
pd_leap_second	Y	N	○
pd_dbsync_altwrite_skip	Y	N	○
pd_sts_initial_error	excontinue	stop	○
pd_sts_singleoperation	stop	stop	○
pd_syssts_initial_error	excontinue	stop	○
pd_syssts_singleoperation	stop	stop	○
pd_registered_port_level	1	0	○
pd_log_max_data_size	523,000	400,000	○
pd_log_rollback_buff_count	0	省略	○
pd_max_tmp_table_rdarea_no	min（131088，pd_max_rdarea_no-3）	0	○
pd_inet_unix_bufmode※4	os	【Linux の場合】 os 【AIX の場合】 conf	○
pd_sql_simple_comment_use	Y	N	○
pd_trn_rollback_msg_interval	5	60	○
pd_utl_exec_mode	1	0	○
pd_watch_time※5	→	0	－
pd_dbbuff_lock_interval	→	1	－
pd_dbbuff_lock_release_detect	→	pipe	－
pd_dbbuff_lock_spn_count	→	100	－
pd_dbbuff_wait_interval	→	【Linux の場合】 5 【AIX の場合】 なし	－
pd_dbbuff_wait_spn_count	→	0	－
pd_dbsync_lck_release_count	→	10,000	－
pd_deter_restart_on_stop_fail	→	Y	－
pd_ipc_tcp_nodelayack	→	Y	－
pd_lck_release_detect	→	pipe	－

オペランド名	バージョン 09-50 の省略値（推奨モード適用時）	0904 互換モード適用時の省略値	オペランドの説明有無
pd_lck_release_detect_interval	→	10	−
pd_max_commit_write_reclaim_no	→	10	−
pd_sql_send_buff_size	→	4	−
pd_server_entry_queue	→	spnfifo	−
pd_thdlock_pipe_retry_interval	→	1,000,000	−
pd_thdlock_retry_time	→	1,000	−
pd_thdlock_sleep_func	→	0	−
pd_thdlock_wakeup_lock※6	→	N	−
pd_thdspnlk_spn_count	→	512	−
pd_utl_file_buff_size	1024	1024	−
pd_server_process_start_mode	→	3	−
pd_server_process_stop_method	→	なし	−

（凡例）

→：

0904 互換モード適用時の省略値です。

○：

「互換モード適用時に検討が必要なオペランド」に定義の説明があります。

−：

オペランドの説明はありません。

注※1

pd_sysdef_default_option オペランドは指定できません。指定するとエラーになるおそれがあります。pd_sysdef_default_option オペランドを指定している場合は，pd_sysdef_default_option オペランドを省略し，「バージョン 09-50 より前の HiRDB システム定義の変更点」を参照して各オペランドの指定値に推奨値を指定することを検討してください。

注※2

pd_large_file_use オペランドを N から Y に変更すると，共用メモリサイズが増加します。メモリ計算式の再見積もりをしてください。

注※3

pd_large_file_use オペランドの値が N の場合，pdconfchk コマンドを実行すると警告メッセージ（KFPS05632-W）が出力されます。

注※4

pd_inet_unix_bufmode オペランドは次に示すオペランドと関連があります。pd_inet_unix_bufmode オペランドの指定値を検討する場合は，次のオペランドも併せて検討してください。

- pd_inet_unix_utl_bufmode
- pd_ipc_inet_bufsize
- pd_ipc_unix_bufsize
- pd_listen_socket_bufset
- pd_tcp_inet_bufsize
- pd_tcp_unix_bufsize
- pd_utl_buff_size

注※5

pd_watch_time オペランドに 0 以外を指定していた場合は，次のすべての対策を実施してください。

1. pd_watch_time オペランドを省略してください。
2. 次のオペランドを指定して SQL，コマンド，及びユティリティの実行時間を監視してください。

   ・クライアント環境定義 PDCWAITTIME

   ・システム定義 pd_cmd_exec_time

   指定値の詳細は，PDCWAITTIME オペランドについてはマニュアル「HiRDB UAP 開発ガイド」，pd_cmd_exec_time オペランドについては，「pd_cmd_exec_time」を参照してください。
3. pd_watch_time オペランドに 180 より大きい値を指定して，pd_lck_wait_timeout オペランドを省略している場合は，pd_watch_time オペランドの指定値を pd_lck_wait_timeout オペランドに指定してください。

注※6

Linux の場合だけ指定不要です。AIX の場合は指定できます。

1.4.2　バージョン 09-50 より前の HiRDB システム定義の変更点

(1)　バージョンによって省略値が異なるオペランド

バージョン 09-50 より前のバージョンによって省略値が異なるオペランドを次の表に示します。

表 1–11　バージョンによって省略値が異なるオペランド

オペランド名	HiRDB V8 以降の省略値	HiRDB V7 の省略値	HiRDB V6 以前の省略値
pd_additional_optimize_level	"COST_BASE_2"	"COST_BASE_2"	"NONE"
pd_assurance_index_no	500	• HiRDB/シングルサーバの場合	• HiRDB/シングルサーバの場合

オペランド名	HiRDB V8 以降の省略値	HiRDB V7 の省略値	HiRDB V6 以前の省略値
		データディクショナリ表のインデクス数（240）＋ 50 • HiRDB/パラレルサーバの場合 50	データディクショナリ表のインデクス数（240）＋ 50 • HiRDB/パラレルサーバの場合 50
pd_assurance_table_no	500	100	0
pd_aud_async_buff_count	Max（1，ユニット内 HiRDB サーバ数×10）	3	3
pd_aud_async_buff_size	401408	4096	4096
pd_bes_shmpool_size	HiRDB による自動計算	HiRDB による自動計算	1024
pd_cancel_down_msgchange	Y	N	N
pd_check_pending※1	USE	• 07-03 以降のバージョン：USE • 07-02 以前のバージョン：NOUSE（USE※1）	NOUSE（USE※1）
pd_dbbuff_attribute	free	fixed（64 ビットモードの場合は free）	fixed（64 ビットモードの場合は free）
pd_dbbuff_lru_option	MIX	SEPARATE	SEPARATE
pd_dic_shmpool_size	HiRDB による自動計算	HiRDB による自動計算	1024
pd_large_file_use	• 08-05 以降のバージョン：Y • 08-04 以前のバージョン：N	N	N
pd_lck_hash_entry	0（HiRDB による自動計算）	0（HiRDB による自動計算）	11261
pd_lck_pool_size	16000（64 ビットモードの場合は 32000）	16000（64 ビットモードの場合は 32000）	1024
pd_lck_release_detect	pipe	interval	interval
pd_lck_wait_timeout	Max（180，pd_watch_time の値）	Max（180，pd_watch_time の値）	180
pd_log_max_data_size	400000	32000	32000
pd_log_sdinterval	• システムログ出力量：5000 • 経過時間：60	• システムログ出力量：1000 • 経過時間：60	• システムログ出力量：1000 • 経過時間：60
pd_log_write_buff_count	10	3	3
pd_max_access_tables	64	64	16

オペランド名	HiRDB V8 以降の省略値	HiRDB V7 の省略値	HiRDB V6 以前の省略値
pd_max_commit_write_reclaim_no	10	0	0
pd_max_server_process	HiRDB による自動計算	HiRDB による自動計算	100
pd_optimize_level	pd_optimize_level オペランドを参照してください。	pd_optimize_level オペランドを参照してください。	"SELECT_APSL"
pd_pageaccess_mode	SNAPSHOT	SNAPSHOT	NORMAL
pd_process_terminator	fixed	fixed	nonresident
pd_process_terminator_max	Max（3，↑（pd_max_users の値＋pd_max_reflect_process_count の値）÷100↑）	↑（pd_max_users の値＋pd_max_reflect_process_count の値）÷100↑	↑（pd_max_users の値＋pd_max_reflect_process_count の値）÷100↑
pd_sds_shmpool_size	HiRDB による自動計算	HiRDB による自動計算	1024
pd_spool_cleanup_interval_level	• 日数：7 • 削除種別：all	• 日数：7 • 削除種別：dump	• 日数：7 • 削除種別：dump
pd_spool_cleanup_level	• 日数：7 • 削除種別：all	• 日数：7 • 削除種別：dump	• 日数：7 • 削除種別：dump
pd_thdlock_pipe_retry_interval	1000000	1000000	4000000（1000000※2）
pd_thdlock_retry_time	1000	10000	10000
pd_trn_commit_optimize	ONEPHASE	ONEPHASE	NOUSE
pd_trn_send_decision_interval	pd_trn_send_decision_intval_sec オペランドの値	5	5
pd_work_buff_mode	pool	pool	each
pd_work_table_option	1	1	0
SHMMAX	200（32 ビットモードの場合だけ省略値が変更になります）	200（32 ビットモードの場合だけ省略値が変更になります）	6

注※1

pd_check_pending オペランドの省略値は，バージョン 07-03 で NOUSE から USE に変更しました。

注※2

pd_thdlock_pipe_retry_interval オペランドの省略値は，バージョン 07-00 で 4000000 から 1000000 に変更しました。

(2)　指定不要になったオペランド

バージョンアップによって，次に示すオペランドを指定する必要がなくなりました。バージョンアップした場合に，これらのオペランドを指定したままでもエラーにはなりません。

バージョン 08-00 から指定する必要がなくなったオペランド

- pd_dynamic_sql_object_cache

バージョン 07-00 から指定する必要がなくなったオペランド

- pd_multi_fes
- pd_redo_skip_inf

1.4.3　32 ビットモードと 64 ビットモードで省略値が異なるオペランド

32 ビットモードと 64 ビットモードでは，次の表に示すシステム定義のオペランドの省略値が変わります。

なお，バージョン 09-50 の省略値は，推奨モード適用時の省略値を記載しています。互換モード適用時の省略値については，「バージョン，リビジョンによって省略値が異なるオペランド」を参照してください。

表 1–12　32 ビットモードと 64 ビットモードで省略値が異なるオペランド

オペランド名	アドレッシングモード	バージョン 09-50 以降の省略値	バージョン 09-00 から 09-50 より前の省略値	HiRDB V8 の省略値	HiRDB V7 の省略値	HiRDB V6 以前の省略値
pd_dbbuff_attribute	32 ビット	→	→	free	→	fixed
	64 ビット				→	free
pd_fes_lck_pool_size	32 ビット	→	→	→	→	※1
	64 ビット	→	→	→	→	※2
pd_lck_pool_size	32 ビット	→	→	→	→	16000
	64 ビット	128000	→	→	→	32000
pd_max_add_dbbuff_no	32 ビット	→	→	→	• a≧500 の場合：256 • a ＜ 500 の場合：500−a	−
	64 ビット	→	→	→	• a≧1000 の場合：256 • a ＜ 1000 の場合：1000−a	−
pd_max_add_dbbuff_shm_no※3	32 ビット	500 ＋（512−b）	→	→	500 ＋（16−b）	−

オペランド名	アドレッシングモード	バージョン 09-50 以降の省略値	バージョン 09-00 から 09-50 より前の省略値	HiRDB V8 の省略値	HiRDB V7 の省略値	HiRDB V6 以前の省略値
	64 ビット	1000 ＋（512−b）	→	→	1000 ＋（16−b）	−
pd_sql_object_cache_size	32 ビット	→	→	→	→	※4
	64 ビット	※5				
pd_table_def_cache_size	32 ビット	→	→	→	→	※6
	64 ビット	16000				
pd_view_def_cache_size	32 ビット	→	→	→	→	※7
	64 ビット	30000				
pd_work_buff_size※8	32 ビット	→	→	→	→	• HiRDB/シングルサーバの場合：384 • HiRDB/パラレルサーバの場合：1024
	64 ビット	→	→	→	→	5120
SHMMAX	32 ビット	100	→	→	200	6
	64 ビット	→	→	→	→	1024

（凡例）

→：

右の列と同じ値であることを示します。

−：

未サポートです。

a：

HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数です。

b：

HiRDB 開始時に各サーバに割り当てられている共用メモリセグメント数です。

共用メモリセグメントの情報は pdls -d mem コマンド，又は OS のコマンドで確認できます。

c：

16 になります。

n：

HiRDB/パラレルサーバでマルチフロントエンドサーバ構成の場合は 4，それ以外は 3 になります。

注※1

{(pd_max_users の値＋ pd_max_reflect_process_count の値＋ 3）×（pd_max_access_tables の値＋ 4)} ÷6

なお，pd_max_reflect_process_count は，インナレプリカ機能を使用している場合だけ加算します。

注※2

{(pd_max_users の値＋ pd_max_reflect_process_count の値＋ 3）×（pd_max_access_tables の値＋ 4)} ÷4

なお，pd_max_reflect_process_count は，インナレプリカ機能を使用している場合だけ加算します。

注※3

pd_max_add_dbbuff_no オペランドを省略した場合の省略値です。pd_max_add_dbbuff_no オペランドに値を設定した場合の省略値については，オペランドの説明を参照してください。

注※4

（pd_max_users の値＋ n）×22

注※5

（pd_max_users の値＋ n）×40

注※6

$\uparrow\sqrt{(\text{pd\_max\_usersの値}+n)}\uparrow \times 100$

注※7

$\uparrow\sqrt{(\text{pd\_max\_usersの値}+n)}\uparrow \times 8$

注※8

pd_work_buff_mode オペランドの値が pool の場合の省略値です。pd_work_buff_mode オペランドの値が each の場合の省略値については，オペランドの説明を参照してください。

1.5　省略値で定義できるデータベースの規模

システム定義の省略値を前提としたデータベースの規模について次の表に示します。この規模を超えるリソース数を使用する場合，省略値ではリソース不足になるおそれがあるため，関連するオペランドを含めて見積もりをして設定してください。

表 1–13　省略値で定義できるデータベースの規模

代表的なリソース	数	関連するオペランド
認可識別子	100	pd_auth_cache_size
実表	500	pd_table_def_cache_size pd_assurance_table_no
インデクス	2500	pd_assurance_index_no
ビュー表	500	pd_table_def_cache_size pd_view_def_cache_size
ルーチン	100	pd_routine_def_cache_size
RD エリア	1000 （1 サーバ当たり）	pd_max_rdarea_no
HiRDB ファイル	2000 （1 サーバ当たり）	pd_max_file_no

●表の分割数による影響

HiRDB/シングルサーバは非分割（分割数 1）を想定しています。また，HiRDB/パラレルサーバは，次の条件で分割数 6 を想定しています。

- システム内のユニット数：3
- ユニット内のバックエンドサーバ数：2
- サーバ内分割は適用しない

表の分割数が上記の値を超える場合は，次のオペランドについて検証が必要です。

- pd_table_def_cache_size

●表の分割方法による影響

次のオペランドについては，サーバ内分割をしている表やインデクスがある場合，分割数を加算することになります。したがって，サーバ内分割をしている場合は，リソース数が想定規模以内であっても，設定値を検証してください。

- pd_assurance_table_no
- pd_assurance_index_no

設定値の目安については，各オペランドの説明を参照してください。

2 システム共通定義

この章では，システム共通定義の各オペランドの形式について説明します。

2.1　オペランドの形式

2.1.1　オペランドの形式の説明

システム共通定義では HiRDB 全体の構成や，HiRDB のユニットに共通な情報を定義します。ここでは，システム共通定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　システム構成

(a)　set 形式

set pd_system_id = HiRDB 識別子

〔set pd_name_port = HiRDB のポート番号〕

set pd_master_file_name = "マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称"

〔set pd_system_expand_unit = ユニット識別子〕

(b)　コマンド形式

{{pdunit -x ホスト名

-u ユニット識別子

〔-d "HiRDB 運用ディレクトリ名"〕

〔-c ホスト名〕

〔-p HiRDB のポート番号〕

〔-s スケジューラプロセスのポート番号〕

〔-t トランザクションサーバプロセスのポート番号〕

〔-m メッセージログサーバプロセスのポート番号〕

〔-a ユニット監視プロセスのポート番号〕}}

{{pdstart -t サーバ種別

〔-s サーバ名〕

{-x ホスト名｜-u ユニット識別子}

〔-m ホスト名〔，ホスト名〕…

〔-n ホスト名〔，ホスト名〕…〕〕

〔-c サーバ名｜-g HA グループ識別子〕

〔-k stls〕}}

(2)　同時実行最大数

(a)　set 形式

〔set pd_max_users = 最大同時接続数〕

〔set pd_max_server_process = 最大同時起動サーバプロセス数〕

〔set pd_max_access_tables = 同時アクセス可能実表数〕

(3)　HiRDB の開始方法

(a)　set 形式

〔set pd_mode_conf = AUTO ｜ MANUAL1 ｜ MANUAL2〕

〔set pd_system_complete_wait_time = pdstart コマンド完了待ち時間〕

〔set pd_start_time_out = HiRDB 開始準備処理の最大待ち時間〕

〔set pd_term_watch_count = 連続異常終了回数の上限〕

(4)　縮退起動

(a)　set 形式

〔set pd_start_level = 0 ｜ 1〕

〔set pd_reduced_check_time = 縮退起動開始の連絡待ち時間〕

〔set pd_start_skip_unit = 開始しないユニットの名称〔，開始しないユニットの名称〕…〕

(5)　HiRDB の処理方式

(a)　set 形式

〔set pd_dbsync_point = sync ｜ commit〕

〔set pd_system_dbsync_point = sync ｜ commit〕

〔set pd_process_terminator = resident ｜ fixed ｜ nonresident〕

〔set pd_process_terminator_max = 後処理プロセスの最大常駐数〕

〔set pd_thdlock_wakeup_lock = Y ｜ N〕

〔set pd_pageaccess_mode = SNAPSHOT ｜ NORMAL〕

〔set pd_cmdhold_precheck = Y ｜ N〕

〔set pd_db_io_error_action = dbhold ｜ unitdown〕

〔set pd_connect_errmsg_hide = Y ｜ N〕

〔set pd_cancel_down_msgchange = Y ｜ N〕

〔set pd_server_process_adjust_limit = 1 回の要求で起動・停止するサーバプロセス数の上限値〕

〔set pd_server_process_adjust_intvl = サーバプロセスの分割起動・停止要求間隔〕

〔set pd_cmd_rmode = rsh ｜ ssh〕

〔set pd_cmd_rconfig = SSH クライアント設定ファイル〕

〔set pd_cmd_rlog = Y ｜ N〕

(6)　全面回復処理

(a)　set 形式

〔set pd_max_recover_process = 全面回復処理の並列実行プロセス数〕

〔set pd_redo_allpage_put = Y ｜ N〕

(7)　トランザクション決着処理

(a)　set 形式

〔set pd_trn_rerun_branch_auto_decide = Y ｜ N〕

〔set pd_trn_send_decision_intval_sec = 秒単位でのトランザクション自動決着の送信リトライ時間間隔〕

〔set pd_trn_send_decision_interval = 分単位でのトランザクション自動決着の送信リトライ時間間隔〕

〔set pd_trn_send_decision_retry_time = トランザクション自動決着の最大待ち時間〕

〔set pd_trn_watch_time = トランザクション同期点処理時の最大通信待ち時間〕

〔set pd_trn_rcvmsg_store_buflen = トランザクション回復メッセージキューサイズ〕

〔set pd_trn_commit_optimize = ONEPHASE ｜ NOUSE〕

〔set pd_trn_rollback_watch_time = ロールバック完了応答の最大待ち時間〔，ロールバック指示再送限界時間〕〕

(8)　SQL の仕様

(a)　set 形式

〔set pd_overflow_suppress = Y ｜ N 〕

〔set pd_space_level = 0 ｜ 1 ｜ 3 〕

〔set pd_dec_sign_normalize = Y ｜ N 〕

〔set pd_sql_dec_op_maxprec = 演算結果が 29 けた以下の DECIMAL 型となる精度の最大値〕

〔set pd_sql_mode = 0 ｜ 1〕

〔set pd_ddl_tbl_pctfree = 表の未使用領域比率,表のセグメント内空きページ比率〕

〔set pd_ddl_tbl_fix_pctfree = 表の未使用領域比率,表のセグメント内空きページ比率〕

〔set pd_ddl_idx_pctfree = インデクスの未使用領域比率〕

〔set pd_ddl_tbl_recovery = {ALL ｜ PARTIAL ｜ NO}〕

(9)　SQL の最適化

(a)　set 形式

〔set pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕…〕

〔set pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕…〕

〔set pd_hash_table_size = ハッシュ表サイズ〕

〔set pd_work_table_option = 作業表処理オプション〕

(10)　絞込み検索

(a)　set 形式

〔set pd_max_list_users = 同時リスト所有可能ユーザ数〕

〔set pd_max_list_count = 1 ユーザ当たりのリスト作成数〕

〔set pd_list_initialize_timing = INITIAL ｜ DEFER ｜ STANDBY〕

〔set pd_list_rowcount_in_rename = Y ｜ N〕

(11)　システム監視

(a)　set 形式

〔set pd_utl_exec_time = ユティリティの実行監視時間〕

〔set pd_cmd_exec_time = 運用コマンド及びユティリティの実行監視時間〕

〔set pd_queue_watch_time = メッセージキュー監視時間〕

〔set pd_queue_watch_timeover_action = continue ｜ stop〕

〔set pd_cmd_process_conwaittime = 運用コマンド・ユティリティ実行におけるプロセス割り当て監視時間〕

〔set pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕〕

〔set pd_host_watch_interval = ホスト間監視時間間隔〕

〔set pd_watch_resource = MANUAL ｜ AUTO ｜ DEFAULT〕

〔set pd_max_users_wrn_pnt = HiRDB サーバへの接続数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕〕

〔set pd_max_access_tables_wrn_pnt = 同時アクセス可能実表数に関する警告メッセージの出力契機〕

〔set pd_max_rdarea_no_wrn_pnt = RD エリア数に関する警告メッセージの出力契機〕

〔set pd_max_file_no_wrn_pnt = HiRDB ファイル数に関する警告メッセージの出力契機〕

〔set pdwork_wrn_pnt = 作業表用ファイルに関する警告メッセージの出力契機〕

〔set pd_max_list_users_wrn_pnt = リスト作成ユーザ数に関する警告メッセージの出力契機〕

〔set pd_max_list_count_wrn_pnt = 1 ユーザ当たりのリスト作成数に関する警告メッセージの出力契機〕

〔set pd_rdarea_list_no_wrn_pnt = サーバ内のリスト作成数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕〕

(12)　SQL 実行時間警告出力機能

(a)　set 形式

〔set pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜ SQL 実行時間警告情報の出力条件（時間指定）〕

〔set pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ〕

〔set pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量〕

(13)　拡張 SQL エラー情報出力機能

(a)　set 形式

〔set pd_uap_exerror_log_use = AUTO ｜ YES ｜ NO〕

〔set pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ〕

〔set pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ〕

〔set pd_uap_exerror_log_param_size = クライアントエラーログファイル及び SQL エラーレポートファイルへ出力するパラメタ情報の最大データ長〕

(14)　SQL 予約語削除機能

(a)　set 形式

〔set pd_delete_reserved_word_file = SQL 予約語削除ファイル名称 1〔，SQL 予約語削除ファイル名称 2〕…〕

(15)　SQL からのコマンド実行

(a)　set 形式

〔set pd_sql_command_exec_users = 認可識別子〔，認可識別子〕…〕

(16)　SQLSTATE の細分化

(a)　set 形式

〔set pd_standard_sqlstate = Y ｜ N〕

(17)　排他制御

(a)　set 形式

〔set pd_lck_deadlock_info = Y ｜ N〕

〔set pd_lck_wait_timeout = 排他待ち限界経過時間〕

〔set pd_nowait_scan_option = LOCK ｜ NOLOCK〕

〔set pd_lck_queue_limit = 排他待ちユーザ数警告メッセージ出力契機〕

〔set pd_deadlock_priority_use = Y ｜ N〕

〔set pd_command_deadlock_priority = 32 ｜ 64 ｜ 96 ｜ 120〕

〔set pd_key_resource_type = TYPE1 ｜ TYPE2〕

〔set pd_indexlock_mode = KEY ｜ NONE〕

〔set pd_lock_uncommited_delete_data = WAIT ｜ NOWAIT〕

〔set pd_dbreuse_remaining_entries = ALL ｜ ONLY_DIC ｜ NONE ｜ ONLY_USER ｜ NOTHING〕

〔set pd_lck_deadlock_check = Y ｜ N〕

〔set pd_lck_deadlock_check_interval = デッドロック監視時間間隔〕

〔set pd_idx_without_rollback =Y ｜ N〕

〔set pd_isolation_level = データ保証レベル〕

〔set pd_unique_indexlock_info=Y ｜ N〕

〔set pd_unique_check_mode=0 ｜ 1〕

〔set pd_lck_deadlock_sql_info = Y ｜ N〕

〔set pd_lck_deadlock_check_timeout = デッドロック監視処理の最大処理時間〕

〔set pd_lck_deadlock_check_pause = デッドロック監視処理中断時の監視処理休止時間〕

(18)　バッファ

(a)　set 形式

〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕

〔set pd_def_buf_control_area_assign = INITIAL ｜ TRAN〕

〔set pd_thread_max_stack_size = 1 スレッドが使用する最大スタックサイズ〕

〔set pd_thread_stack_expand_size = 1 スレッド当たりのスタック拡張サイズ〕

(19)　共用メモリ

(a)　set 形式

〔set pd_shmpool_attribute = free ｜ fixed ｜ hugepage〕

〔set pd_shmpool_control = unit ｜ server〕

〔set pd_dbbuff_attribute = free ｜ fixed ｜ hugepage〕

(b)　putenv 形式

〔putenv SHMMAX 共用メモリセグメントサイズの上限値〕

(20)　メッセージログファイル

(a)　set 形式

〔set pd_mlg_msg_log_unit = manager ｜ local〕

〔set pd_mlg_file_size = メッセージログファイルの最大容量〕

(21)　統計情報

(a)　set 形式

〔set pd_statistics = A ｜ Y ｜ N〕

〔set pd_stj_file_size = 統計ログファイルの最大容量〕

〔set pd_stj_buff_size = 統計ログバッファ長〕

〔set pd_sqlobject_stat_timing = deallocate ｜ tran〕

(b)　コマンド形式

〔pdstbegin〔-k 統計情報種別〔，統計情報種別〕…〕

〔-m システム稼働統計情報出力時間間隔〕

〔-I データベースの入出力統計情報出力時間間隔〕

〔-D データベースの入出力統計情報オプション

〔，データベースの入出力統計情報オプション〕〕

〔{-x ホスト名｜-u ユニット識別子}〕

〔{-a ｜-s サーバ名〔，サーバ名〕…}〕〕

〔-w〕

〔pdhibegin -k 履歴情報種別〕

(22)　RPC トレース情報

(a)　set 形式

〔set pd_rpc_trace = Y ｜ N〕

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕

(23)　PRF トレース情報

(a)　set 形式

〔set pd_prf_trace = Y ｜ N〕

〔set pd_prf_level = 00000007 ｜ 0000001f ｜ 0000007f ｜ 000001ff ｜ 00000000〕

〔set pd_prf_file_count = PRF トレース情報ファイルの世代数〕

〔set pd_prf_file_size = PRF トレース情報ファイルサイズ〕

(24)　トラブルシュート情報

(a)　set 形式

〔set pd_cancel_dump = put ｜ noput〕

〔set pd_client_waittime_over_abort = Y ｜ N〕

〔set pd_clt_waittime_over_dump_level = all ｜ shm_fesonly〕

〔set pd_dump_suppress_watch_time =トラブルシュート情報の出力抑止時間〕

〔set pd_debug_info_netstat = Y ｜ N〕

〔set pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔〕

〔set pd_spool_cleanup_interval_level = 日数〔，削除種別〕〕

〔set pd_spool_cleanup = normal ｜ force ｜ no〕

〔set pd_spool_cleanup_level = 日数〔，削除種別〕〕

〔set pd_module_trace_max = モジュールトレース格納最大数〕

〔set pd_module_trace_timer_level = 0 ｜ 10 ｜ 20〕

〔set pd_pth_trace_max = 通信トレース格納最大数〕

(25) RD エリア

(a) set 形式

〔set pd_max_rdarea_no = RD エリアの最大数〕

〔set pd_max_file_no = RD エリアを構成する HiRDB ファイルの最大数〕

〔set pd_rdarea_warning_point = セグメント使用比率 1〔，セグメント使用比率 2〔，セグメント使用比率 3〕〕〕

〔set pd_rdarea_warning_point_msgout = Y ｜ N〕

〔set pd_stored_lob_data_wrn_pnt = セグメント使用比率 1〔，セグメント使用比率 2〔，セグメント使用比率 3〕〕〕

〔set pd_rdarea_extension_timing = use ｜ nouse〕

〔set pd_rdarea_expand_format = Y ｜ N〕

〔set pd_rdarea_extension_file = all ｜ last〕

〔set pd_rdarea_open_attribute_use = Y ｜ N〕

〔set pd_rdarea_open_attribute = INITIAL ｜ DEFER ｜ SCHEDULE〕

〔set pd_shared_rdarea_use = Y ｜ N〕

〔set pd_db_access_error_action = dbhold ｜ unitdown〕

〔set pd_db_hold_action = dbhold ｜ unitdown〕

〔set pd_dbbuff_read_retry_count = RD エリアのページ読み込み時に不正を検知した際の IO リトライ回数〕

〔set pd_dbbuff_read_retry_intvl = RD エリアのページ読み込み時に不正を検知した際の IO リトライ間隔〕

〔set pd_db_timestamp_invalid_action = dbhold | unitdown〕

(26)　グローバルバッファ

(a)　set 形式

〔set pd_dbbuff_lru_option = SEPARATE ｜ MIX〕

〔set pd_dbbuff_binary_data_lru = Y ｜ N〕

〔set pd_dbbuff_dev_sector_size = 512 ｜ 4096〕

〔set pd_dbbuff_modify = Y ｜ N〕

〔set pd_dbbuff_rate_updpage = デファードライトトリガの要求比率〕

〔set pd_dbbuff_trace_level = グローバルバッファ制御情報トレース取得レベル〕

(b)　コマンド形式

〔{{pdbuffer -a グローバルバッファ名

{-r RD エリア名〔，RD エリア名〕...｜

-b RD エリア名〔，RD エリア名〕...｜

-o ｜

-i 認可識別子．インデクス識別子}

-n バッファ面数〔- l バッファサイズ〕

〔-m 同時実行最大プリフェッチ数〕

〔-p 一括入力最大ページ数〕

〔-w デファードライトトリガ時の更新ページ出力比率〕

〔-c〕

〔-y デファードライトトリガ契機の更新バッファ面数〕

〔-D 分割数〕}}〕

(27)　インメモリデータ処理

(a)　set 形式

〔set pd_max_resident_rdarea_no = インメモリ RD エリアの最大数〕

〔set pd_max_resident_rdarea_shm_no = インメモリデータバッファが使用する共用メモリセグメントの最大数〕

(28)　表又はインデクスの予約数

(a)　set 形式

〔set pd_assurance_table_no = 表予約数〕

〔set pd_assurance_index_no = インデクス予約数〕

(29)　参照制約及び検査制約

(a)　set 形式

〔set pd_constraint_name = LEADING ｜ TRAILING〕

〔set pd_check_pending = USE ｜ NOUSE〕

(30)　一時表

(a)　set 形式

〔set pd_max_temporary_object_no = ある一時点で使用する一時表と一時インデクスの最大数〕

〔set pd_tmp_table_initialize_timing = STARTING ｜ ACCESS〕

(31)　再編成時期予測機能

(a)　set 形式

〔set pd_rorg_predict = Y ｜ N〕

(32)　セキュリティ監査機能

(a)　set 形式

〔set pd_audit = Y ｜ N〕

〔set pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名〕

〔set pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量〕

〔set pd_aud_max_generation_num = 監査証跡ファイルの最大数〕

〔set pd_aud_no_standby_file_opr = down ｜ forcewrite〕

〔set pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長〕

〔set pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数〕

〔set pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔〕

〔set pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ〕

〔set pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ〕

〔set pd_aud_file_wrn_pnt = 警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕〕

〔set pd_aud_auto_loading = Y ｜ N〕

(b)　コマンド形式

〔pdaudload〔-i インデクス作成方法〕

〔-l ログ取得方式〕

〔-n 〔一括出力用ローカルバッファ面数〕,,

〔ランダムアクセス用ローカルバッファ面数〕〕

〔-y〕

〔-X サーバ間通信の応答監視時間〕〕

(33)　セキュリティの強化

(a)　set 形式

〔set pd_security_host_group = ホスト名〔，ホスト名〕…〕

〔set pd_security_local_constraint = Y ｜ N〕

(34)　系切り替え機能

(a)　set 形式

〔set pd_ha = use ｜ nouse〕

〔set pd_ha_ipaddr_inherit = Y ｜ N〕

〔set pd_ha_switch_timeout = Y ｜ N〕

〔set pd_ha_prc_cleanup_check = Y ｜ N〕

〔set pd_ha_mgr_rerun = wait ｜ notwait〕

〔set pd_ha_transaction = error ｜ queuing〕

〔set pd_ha_trn_queuing_wait_time = トランザクションのキューイング待ち時間〕

〔set pd_ha_trn_restart_retry_time = トランザクション開始要求エラー時のリトライ時間の上限〕

〔set pd_ha_trn_queuing_initial = Y ｜ N〕

〔set pd_ha_resource_act_wait_time = リソース活性化の最大待ち時間〕

〔set pd_ha_switch_error = ignore｜retry〕

〔set pd_ha_observe_timeout = 系切り替え処理の監視時間〕

(b)　コマンド形式

〔pdhagroup -g HA グループ識別子 -u ユニット識別子〔，ユニット識別子〕…〕

(35)　HiRDB Datareplicator

(a)　set 形式

〔set pd_rpl_init_start = Y ｜ N〕

〔set pd_rpl_reflect_mode = server ｜ uap〕

〔set pd_log_rpl_no_standby_file_opr = stop ｜ continue〕

〔set pd_rpl_func_control = BACKWARD_CUTOFF_UPDATE ｜ NONE〕

(36)　リアルタイム SAN レプリケーション

(a)　set 形式

〔set pd_rise_use = Y ｜ N〕

〔set pd_rise_pairvolume_combination = sync ｜ async ｜ hybrid ｜ syssync〕

〔set pd_rise_disaster_mode = normal ｜ alone〕

〔set pd_rise_fence_level = data ｜ never〕

(b)　putenv 形式

〔putenv HORCMINST RAID Manager のインスタンス番号〕

(37) インナレプリカ機能

(a) set 形式

〔set pd_inner_replica_control = インナレプリカ最大グループ数〕

〔set pd_inner_replica_lock_shift = Y | N〕

〔set pd_lv_mirror_use = Y | N〕

〔set pd_inner_replica_hold_status = NORMAL | HOLD〕

〔set pd_max_reflect_process_count = 追い付き反映処理時に確保するプロセス数〕

〔set pd_log_org_reflected_logpoint = keep | release〕

〔set pd_log_org_no_standby_file_opr = stop | continue〕

(38) OLTP

(a) set 形式

〔set pd_oltp_holdcr = use | nouse〕

(39) バージョンアップ

(a) set 形式

〔set pd_auto_vrup = Y | N〕

(40) 通信処理

(a) set 形式

〔set pd_service_port = スケジューラプロセスのポート番号〕

〔set pd_name_fixed_port_lookup = Y | N〕

〔set pd_scd_port = スケジューラプロセスのポート番号〕

〔set pd_trn_port = トランザクションサーバプロセスのポート番号〕

〔set pd_mlg_port = メッセージログサーバプロセスのポート番号〕

〔set pd_alv_port = ユニット監視プロセスのポート番号〕

〔set pd_change_clt_ipaddr = 0 | 1 〕

〔set pd_registered_port = "ポート番号の予約範囲"〔，"ポート番号の予約範囲"〕…〕

〔set pd_registered_port_check = Y ｜ N ｜ C ｜ W〕

〔set pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数〕

〔set pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間〕

〔set pd_ipc_send_count = サーバ間の送信処理のリトライ回数〕

〔set pd_ipc_recv_count = サーバ間の受信処理のリトライ回数〕

〔set pd_ipc_conn_nblock = Y ｜ N〕

〔set pd_ipc_conn_nblock_time = ノンブロックモードでのコネクション確立監視時間〕

〔set pd_ipc_clt_conn_nblock = Y ｜ N〕

〔set pd_ipc_clt_conn_nblock_time = ノンブロックモードでのコネクション確立監視時間（HiRDB クライアントとの通信）〕

〔set pd_ipc_conn_interval = コネクション確立処理のリトライ間隔〕

〔set pd_ipc_conn_count =コネクション確立処理のリトライ回数〕

〔set pd_ipc_file_dir = 通信情報ファイルディレクトリ〕

〔set pd_crypto_service_port = スケジューラプロセスの暗号化通信接続用のポート番号〕

〔set pd_crypto_private_key = 通信暗号化機能に使用する秘密鍵〕

〔set pd_crypto_certificate_file = 通信暗号化機能に使用するサーバ証明書〕

〔set pd_ipc_so_keepalive= Y ｜ N〕

〔set pd_ipc_tcp_keepidle= KeepAlive パケットの送信間隔〕

〔set pd_ipc_tcp_keepintvl= ネットワーク障害検知後の KeepAlive パケットの送信間隔〕

〔set pd_ipc_tcp_keepcnt= ネットワーク障害検知後の KeepAlive パケット再送回数の上限値〕

(41)　Java

(a)　set 形式

〔set pd_java_option = "Java オプション"〔，"Java オプション"〕…〕

〔set pd_java_routine_stack_size = 外部 Java ルーチンが使用するスタック領域長〕

〔set pd_java_archive_directory = "JAR ファイル格納ディレクトリ"〕

〔set pd_java_classpath = "Java クラスパス"〕

〔set pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"〕

〔set pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"〕

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕

(42) 外部 C ストアドルーチン

(a) set 形式

〔set pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"〕

(43) 文字コード

(a) set 形式

〔set pd_substr_length = 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10〕

(44) ユーザ認証

(a) set 形式

〔set pd_os_authenticate = Y | N〕

〔set pd_connect_auth_type = ALL |方式〔，方式〕…〕

(45) クライアントグループ

(a) コマンド形式

{{〔pdcltgrp -g クライアントグループ名称

-u グループごとの接続保証ユーザ数〕}}

(46) プラグイン

(a) コマンド形式

{{〔pdplugin -n プラグイン名称〕}}

(47)　日付・時刻

(a)　putenv 形式

〔putenv TZ タイムゾーン〕

(48)　メッセージの出力抑止機能

(a)　コマンド形式

〔{{pdmlgput -s 出力有無

{-c ALL ｜〔-l メッセージの重要度〕 -m メッセージ ID〔，メッセージ ID〕…}}}〕

(49)　HiRDB ファイルシステム領域

(a)　set 形式

〔set pd_duplicated_fs_suffix = "複製ディスク機能使用時のセカンダリパスのサフィックス"〕

〔set pd_max_utl_ios_file_no = pdfmkfs -k UTL -m コマンドで作成した HiRDB ファイルシステム領域数〕

(50)　リードレプリカ機能

(a)　set 形式

〔set pd_read_replica_use = Y ｜ N〕

〔set pd_read_replica_mode = unld ｜ sys〕

〔set pd_read_replica_workdir = 作業用ディレクトリ〕

〔set pd_read_replica_max_trn_count = トランザクション数〕

3 ユニット制御情報定義

この章では，ユニット制御情報定義の各オペランドの形式について説明します。

3.1　オペランドの形式

3.1.1　オペランドの形式の説明

ユニット制御情報定義ではユニットの情報を定義します。ここでは，ユニット制御情報定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

●影響分散スタンバイレス型系切り替え機能を使用される方へ

影響分散スタンバイレス型系切り替え機能を使用する場合，指定できるオペランドが決まっています。詳細については，「影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義）」を参照してください。

(1)　システム構成

(a)　set 形式

set pd_unit_id = ユニット識別子

〔set pd_hostname = ホスト名〕

(2)　同時実行最大数

(a)　set 形式

〔set pd_max_server_process = 最大同時起動サーバプロセス数〕※

(3)　HiRDB の開始方法

(a)　set 形式

〔set pd_term_watch_count = 連続異常終了回数の上限〕※

(4)　HiRDB の処理方式

(a)　set 形式

〔set pd_thdlock_wakeup_lock = Y ｜ N〕※

〔set pd_db_io_error_action = dbhold ｜ unitdown〕※

〔set pd_cmd_rconfig = SSH クライアント設定ファイル〕※

(5)　全面回復処理

(a)　set 形式

〔set pd_max_recover_process = 全面回復処理の並列実行プロセス数〕※

(6)　トランザクション決着処理

(a)　set 形式

〔set pd_trn_rcvmsg_store_buflen = トランザクション回復メッセージキューサイズ〕※

(7)　システム監視

(a)　set 形式

〔set pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕〕※

(8)　SQL 実行時間警告出力機能

(a)　set 形式

〔set pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ〕※

〔set pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量〕※

(9)　拡張 SQL エラー情報出力機能

(a)　set 形式

〔set pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ〕※

〔set pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ〕※

〔set pd_uap_exerror_log_param_size = クライアントエラーログファイル及び SQL エラーレポートファイルへ出力するパラメタ情報の最大データ長〕※

(10)　排他制御

(a)　set 形式

〔set pd_lck_wait_timeout = 排他待ち限界経過時間〕※

〔set pd_lck_deadlock_info = Y ｜ N〕※

〔set pd_lck_deadlock_check = Y ｜ N〕※

〔set pd_lck_deadlock_check_interval = デッドロック監視時間間隔〕※

〔set pd_lck_deadlock_sql_info = Y ｜ N〕※

〔set pd_lck_deadlock_check_timeout = デッドロック監視処理の最大処理時間〕※

〔set pd_lck_deadlock_check_pause = デッドロック監視処理中断時の監視処理休止時間〕※

(11) バッファ

(a) set 形式

〔set pd_thread_stack_expand_size = 1 スレッド当たりのスタック拡張サイズ〕

(12) 共用メモリ

(a) set 形式

〔set pd_shmpool_attribute = free ｜ fixed ｜ hugepage〕※

〔set pd_dbbuff_attribute = free ｜ fixed ｜ hugepage〕※

(b) putenv 形式

〔putenv SHMMAX 共用メモリセグメントサイズの上限値〕※

(13) 統計情報

(a) set 形式

〔set pd_stj_file_size = 統計ログファイルの最大容量〕※

〔set pd_stj_buff_size = 統計ログバッファ長〕※

(14) RPC トレース情報

(a) set 形式

〔set pd_rpc_trace = Y ｜ N〕※

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※

(15)　PRF トレース情報

(a)　set 形式

〔set pd_prf_level = 00000007 ｜ 0000001f ｜ 0000007f ｜ 000001ff ｜ 00000000〕※

〔set pd_prf_file_count = PRF トレース情報ファイルの世代数〕※

〔set pd_prf_file_size = PRF トレース情報ファイルサイズ〕※

〔set pd_prf_output_directory = PRF トレースを出力するディレクトリ〕

(16)　トラブルシュート情報

(a)　set 形式

〔set pd_cancel_dump = put ｜ noput〕※

〔set pd_dump_suppress_watch_time = トラブルシュート情報の出力抑止時間〕※

〔set pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔〕※

〔set pd_spool_cleanup_interval_level = 日数〔，削除種別〕〕※

〔set pd_spool_cleanup = normal ｜ force ｜ no〕※

〔set pd_spool_cleanup_level = 日数〔，削除種別〕〕※

〔set pd_module_trace_max = モジュールトレース格納最大数〕※

〔set pd_module_trace_timer_level = 0 ｜ 10 ｜ 20〕※

〔set pd_pth_trace_max = 通信トレース格納最大数〕※

(17)　ユニット用ステータスファイル

(a)　set 形式

set pd_syssts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

〔set pd_syssts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_syssts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_syssts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_syssts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_syssts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_syssts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_syssts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_syssts_subfile_name_2 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_syssts_subfile_name_3 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_syssts_subfile_name_4 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_syssts_subfile_name_5 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_syssts_subfile_name_6 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_syssts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

(18)　ユニット用ステータスファイル（障害発生時）

(a)　set 形式

〔set pd_syssts_last_active_file = "論理ファイル名"〕

〔set pd_syssts_last_active_side = A ｜ B〕

〔set pd_syssts_last_active_subfile = "論理ファイル名"〕

〔set pd_syssts_last_active_side_sub = A ｜ B〕

(19)　セキュリティ

(a)　set 形式

〔set pd_audit = Y ｜ N〕※

〔set pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名〕※

〔set pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量〕※

〔set pd_aud_max_generation_num = 監査証跡ファイルの最大数〕※

〔set pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長〕※

〔set pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数〕※

〔set pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔〕※

〔set pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ〕

〔set pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ〕

(20)　系切り替え機能

(a)　set 形式

〔set pd_ha_acttype = monitor ｜ server〕

〔set pd_ha_unit = nouse〕

〔set pd_ha_restart_failure = コマンド名〕

〔set pd_ha_switch_timeout = Y ｜ N〕※

〔set pd_ha_server_process_standby = Y ｜ N〕

〔set pd_ha_agent = standbyunit ｜ server ｜ activeunits〕

〔set pd_ha_max_act_guest_servers = 受け入れ可能なゲスト BES の最大数〕

〔set pd_ha_max_server_process = 受け入れユニット内の最大ユーザサーバプロセス数〕

〔set pd_ha_process_count = ゲスト BES 受け入れ後のユニット内常駐プロセス数〕

〔set pd_ha_resource_act_wait_time = リソース活性化の最大待ち時間〕※

〔set pd_ha_ipaddr_inherit = Y ｜ N〕※

〔set pd_ha_switch_error = ignore | retry〕

〔set pd_ha_observe_timeout = 系切り替え処理の監視時間〕

(21) HiRDB Datareplicator

(a) set 形式

〔set pd_rpl_hdepath = 抽出側 HiRDB Datareplicator 運用ディレクトリ名〕

(22) 通信処理

(a) set 形式

〔set pd_service_port = スケジューラプロセスのポート番号〕※

〔set pd_change_clt_ipaddr = 0 | 1〕※

〔set pd_registered_port = "ポート番号の予約範囲"〔,"ポート番号の予約範囲"〕…〕※

〔set pd_registered_port_check = Y | N | C | W〕※

〔set pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数〕※

〔set pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間〕※

〔set pd_ipc_send_count = サーバ間の送信処理のリトライ回数〕※

〔set pd_ipc_recv_count = サーバ間の受信処理のリトライ回数〕※

〔set pd_crypto_service_port = スケジューラプロセスの暗号化通信接続用のポート番号〕※

〔set pd_crypto_private_key = 通信暗号化機能に使用する秘密鍵〕※

〔set pd_crypto_certificate_file = 通信暗号化機能に使用するサーバ証明書〕※

(23) Java

(a) set 形式

〔set pd_java_archive_directory = "JAR ファイル格納ディレクトリ"〕※

〔set pd_java_classpath = "Java クラスパス"〕※

〔set pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"〕※

〔set pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"〕※

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※

(24)　外部 C ストアドルーチン

(a)　set 形式

〔set pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"〕

(25)　ワークファイル出力先ディレクトリの変更

(a)　set 形式

〔set pd_tmp_directory = ワークファイル出力先ディレクトリ名〕

(26)　HiRDB システム定義ファイルの共用化

(a)　putenv 形式

〔putenv PDCONFPATH 共用ディレクトリ名〕

(27)　HiRDB ファイルシステム領域

(a)　set 形式

〔set pd_max_utl_ios_file_no = pdfmkfs -k UTL -m コマンドで作成した HiRDB ファイルシステム領域数〕

注※

このオペランドを省略すると，システム共通定義の同じオペランドの指定値が仮定されます。

4 サーバ共通定義

この章では，サーバ共通定義の各オペランドの形式について説明します。

4.1　オペランドの形式

4.1.1　オペランドの形式の説明

サーバ共通定義では，各サーバ定義（フロントエンドサーバ定義，ディクショナリサーバ定義，及びバックエンドサーバ定義）で共通な情報を定義します。ここで定義した内容は，各サーバ定義で指定しなかったオペランドの仮定値となります。ここでは，サーバ共通定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

(a)　set 形式

〔set pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数〕

〔set pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数〕

〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕

〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕

〔set pd_svr_castoff_size = 1 サーバプロセスが使用するメモリサイズの上限値〕

〔set pd_max_ard_process = 非同期 READ プロセス数〕

〔set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数〕

(2)　作業表

(a)　set 形式

〔set pd_work_buff_mode = each ｜ pool ｜ pool2〕

〔set pd_work_buff_size = 作業表用バッファ長〕

〔set pd_work_buff_expand_limit = 作業表用バッファの増分上限値〕

(3)　システム監視

(a)　set 形式

〔set pd_watch_pc_client_time = クライアントからの要求間隔監視時間〕

〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕

〔set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値〕

(4)　排他制御

(a)　set 形式

〔set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ〕

〔set pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ〕

〔set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数〕

〔set pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数〕

〔set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和〕

〔set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数〕

〔set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数〕

(5)　バッファ

(a)　set 形式

〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕※

〔set pd_table_def_cache_size = 表定義情報用バッファ長〕

〔set pd_auth_cache_size = ユーザ権限情報用バッファ長〕

〔set pd_view_def_cache_size = ビュー解析情報用バッファ長〕

〔set pd_type_def_cache_size = ユーザ定義型情報用バッファ長〕

〔set pd_routine_def_cache_size = ルーチン定義情報用バッファ長〕

〔set pd_registry_cache_size = レジストリ情報用バッファ長〕

(6)　共用メモリ

(a)　set 形式

〔set pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ〕

〔set pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ〕

〔set pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ〕

(7)　RPC トレース情報

(a)　set 形式

〔set pd_rpc_trace = Y ｜ N〕※

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※

(8)　トラブルシュート情報

(a)　set 形式

〔set pd_module_trace_max = モジュールトレース格納最大数〕※

〔set pd_module_trace_timer_level = 0 ｜ 10 ｜ 20〕※

〔set pd_pth_trace_max = 通信トレース格納最大数〕※

(9)　グローバルバッファ

(a)　set 形式

〔set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値〕

〔set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値〕

〔set pd_dfw_bulk_write = awt | none〕

(10)　一時表

(a)　set 形式

〔set pd_max_temporary_object_no = ある一時点で使用する一時表と一時インデクスの最大数〕※

(11)　セキュリティ

(a)　set 形式

〔set pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長〕

(12)　Java

(a)　set 形式

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※

〔set pd_java_castoff = Y ｜ N〕

(13)　システムログファイル

(a)　set 形式

〔set pd_log_dual = Y ｜ N〕

〔set pd_log_dual_write_method = serial ｜ parallel〕

〔set pd_log_remain_space_check = warn ｜ safe〕

〔set pd_log_auto_unload_restart = Y ｜ N〕

〔set pd_log_singleoperation = Y ｜ N〕

〔set pd_log_rerun_reserved_file_open = Y ｜ N〕

〔set pd_log_rerun_swap = Y ｜ N〕

〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕

〔set pd_log_unload_check = Y ｜ N〕

〔set pd_log_write_buff_count = ログ出力バッファ面数〕

〔set pd_log_full_write_async = Y ｜ N〕

〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕

〔set pd_log_fg_warning_point = 空きシステムログファイルの数 1〔，空きシステムログファイルの数 2〔，空きシステムログファイルの数 3〕〕〕

〔set pd_log_block_check = Y ｜ N〕

(14)　シンクポイントダンプファイル

(a)　set 形式

〔**set pd_spd_dual = Y** ｜ **N**〕

〔set pd_spd_assurance_msg = Y ｜ N〕

〔set pd_spd_assurance_count = 有効保証世代数〕

〔set pd_spd_reduced_mode = 縮退運転オプション〕

〔set pd_spd_reserved_file_auto_open = Y ｜ N〕

〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕

〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕

(15)　バックエンドサーバ接続保持機能

(a)　set 形式

〔set pd_bes_connection_hold = Y ｜ N〕

〔set pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間〕

注※

このオペランドを省略すると，システム共通定義又はユニット制御情報定義の同じオペランドの指定値が仮定されます。

5 シングルサーバ定義

この章では，シングルサーバ定義の各オペランドの形式について説明します。

5.1　オペランドの形式

5.1.1　オペランドの形式の説明

シングルサーバ定義ではシングルサーバの実行環境を定義します。ここでは，シングルサーバ定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

(a)　set 形式

〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕※1

〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕※1

〔set pd_svr_castoff_size = 1 サーバプロセスが使用するメモリサイズの上限値〕※1

〔set pd_max_open_fds = 1 プロセスがアクセスするファイル及びパイプの最大数〕

〔set pd_max_ard_process = 非同期 READ プロセス数〕※1

〔set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数〕

(2)　作業表

(a)　set 形式

〔set pd_work_buff_mode = each ｜ pool ｜ pool2〕※1

〔set pd_work_buff_size = 作業表用バッファ長〕※1

〔set pd_work_buff_expand_limit = 作業表用バッファの増分上限値〕※1

(b)　コマンド形式

pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕...

(3)　システム監視

(a)　set 形式

〔set pd_watch_pc_client_time = クライアントからの要求間隔監視時間〕※1

〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕※1

〔set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値〕※1

(4)　SQL 実行時間警告出力機能

(a)　set 形式

〔set pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜ SQL 実行時間警告情報の出力条件（時間指定）〕※1

(5)　拡張 SQL エラー情報出力機能

(a)　set 形式

〔set pd_uap_exerror_log_use = AUTO ｜ YES ｜ NO〕※1

(6)　排他制御

(a)　set 形式

〔set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ〕※1

〔set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数〕※1

〔set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和〕※1

〔set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数〕※1

〔set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数〕※1

(7)　バッファ

(a)　set 形式

〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕※1

〔set pd_table_def_cache_size = 表定義情報用バッファ長〕※1

〔set pd_auth_cache_size = ユーザ権限情報用バッファ長〕※1

〔set pd_view_def_cache_size = ビュー解析情報用バッファ長〕※1

〔set pd_type_def_cache_size = ユーザ定義型情報用バッファ長〕※1

〔set pd_routine_def_cache_size = ルーチン定義情報用バッファ長〕※1

〔set pd_registry_cache_size = レジストリ情報用バッファ長〕※1

(8) 共用メモリ

(a) set 形式

〔set pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ〕※1

(9) RPC トレース情報

(a) set 形式

〔set pd_rpc_trace = Y | N〕※1

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※1

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※1

(10) トラブルシュート情報

(a) set 形式

〔set pd_module_trace_max = モジュールトレース格納最大数〕※2

〔set pd_module_trace_timer_level = 0 | 10 | 20〕※2

〔set pd_pth_trace_max = 通信トレース格納最大数〕※1

(11) グローバルバッファ

(a) set 形式

〔set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値〕※1

〔set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値〕※1

〔set pd_dfw_bulk_write = awt | none〕※1

(12) 一時表

(a) set 形式

〔set pd_max_temporary_object_no = ある一時点で使用する一時表と一時インデクスの最大数〕※1

(13) セキュリティ

(a) set 形式

〔set pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長〕※1

(14) プラグイン

(a) set 形式

〔set pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"〕

(b) コマンド形式

{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

(15) Java

(a) set 形式

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※1

〔set pd_java_castoff = Y | N〕※1

(16) システムログファイル

(a) set 形式

〔set pd_log_dual = Y | N〕※1

〔set pd_log_dual_write_method = serial | parallel〕※1

〔set pd_log_remain_space_check = warn | safe〕※1

〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔, "アンロードログファイルの出力先ディレクトリ"〕…〕

〔set pd_log_auto_unload_restart = Y ｜ N〕※1

〔set pd_log_singleoperation = Y ｜ N〕※1

〔set pd_log_rerun_reserved_file_open = Y ｜ N〕※1

〔set pd_log_rerun_swap = Y ｜ N〕※1

〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※1

〔set pd_log_unload_check = Y ｜ N〕※1

〔set pd_log_write_buff_count = ログ出力バッファ面数〕※1

〔set pd_log_full_write_async = Y ｜ N〕※1

〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※1

〔set pd_log_fg_warning_point = 空きシステムログファイルの数 1〔，,空きシステムログファイルの数 2〔，,空きシステムログファイルの数 3〕〕〕※1

〔set pd_log_block_check = Y ｜ N〕※1

(b)　コマンド形式

{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕}}

{{pdlogadfg -d rrp -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d rrp -g ファイルグループ名 -a "反映対象システムログファイル名"〔-b "反映対象システムログファイル名"〕}}

(17)　シンクポイントダンプファイル

(a)　set 形式

〔set pd_spd_dual = Y ｜ N〕※1

〔set pd_spd_assurance_msg = Y ｜ N〕※1

〔set pd_spd_assurance_count = 有効保証世代数〕※1

〔set pd_spd_reduced_mode = 縮退運転オプション〕※1

〔set pd_spd_reserved_file_auto_open = Y ｜ N〕※1

〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※1

〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※1

(b)　コマンド形式

{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名"〔-b "シンクポイントダンプファイル名"〕}}

{{pdlogadfg -d ssp -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d ssp -g ファイルグループ名 -a "副シンクポイントダンプファイル名"〔-b "副シンクポイントダンプファイル名"〕}}

(18)　サーバ用ステータスファイル

(a)　set 形式

set pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

〔set pd_sts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_2 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_3 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_4 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_5 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_6 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

(19)　サーバ用ステータスファイル（障害発生時）

(a)　set 形式

〔set pd_sts_last_active_file = "論理ファイル名"〕

〔set pd_sts_last_active_side = A ｜ B〕

〔set pd_sts_last_active_subfile = "論理ファイル名"〕

〔set pd_sts_last_active_side_sub = A ｜ B〕

注※1

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

ただし，次に示すオペランドについては，サーバ共通定義ではなく，システム共通定義の同じオペランドの指定値が有効となります。

- pd_cwaittime_wrn_pnt
- pd_uap_exerror_log_use

注※2

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，又はシステム共通定義の同じオペランドの指定値が有効となります。

6 フロントエンドサーバ定義

この章では，フロントエンドサーバ定義の各オペランドの形式について説明します。

6.1.1　オペランドの形式の説明

フロントエンドサーバ定義ではフロントエンドサーバの実行環境を定義します。ここでは，フロントエンドサーバ定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

(a)　set 形式

〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕※1

〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕※1

〔set pd_svr_castoff_size = 1 サーバプロセスが使用するメモリサイズの上限値〕※1

(2)　SQL の最適化

(a)　set 形式

〔set pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕…〕※1

〔set pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕…〕※1

〔set pd_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕…〕

〔set pd_non_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕…〕

(3)　システム監視

(a)　set 形式

〔set pd_watch_pc_client_time = クライアントからの要求間隔監視時間〕※1

〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕※1

(4)　SQL 実行時間警告出力機能

(a)　set 形式

〔set pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜ SQL 実行時間警告情報の出力条件（時間指定）〕※1

(5)　拡張 SQL エラー情報出力機能

(a)　set 形式

〔set pd_uap_exerror_log_use = AUTO ｜ YES ｜ NO〕※1

(6)　排他制御

(a)　set 形式

〔set pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ〕※1

〔set pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数〕※1

〔set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数〕※1

〔set pd_isolation_level = データ保証レベル〕※1

(7)　バッファ

(a)　set 形式

〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕※1

〔set pd_table_def_cache_size = 表定義情報用バッファ長〕※1

〔set pd_auth_cache_size = ユーザ権限情報用バッファ長〕※1

〔set pd_view_def_cache_size = ビュー解析情報用バッファ長〕※1

〔set pd_type_def_cache_size = ユーザ定義型情報用バッファ長〕※1

〔set pd_routine_def_cache_size = ルーチン定義情報用バッファ長〕※1

〔set pd_registry_cache_size = レジストリ情報用バッファ長〕※1

(8)　RPC トレース情報

(a)　set 形式

〔set pd_rpc_trace = Y ｜ N〕※1

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※1

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※1

(9)　トラブルシュート情報

(a)　set 形式

〔set pd_module_trace_max = モジュールトレース格納最大数〕※2

〔set pd_module_trace_timer_level = 0 ｜ 10 ｜ 20〕※2

〔set pd_pth_trace_max = 通信トレース格納最大数〕※1

(10)　セキュリティ

(a)　set 形式

〔set pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長〕※1

(11)　Java

(a)　set 形式

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※1

〔set pd_java_castoff = Y ｜ N〕※1

(12)　システムログファイル

(a)　set 形式

〔set pd_log_dual = Y ｜ N〕※1

〔set pd_log_dual_write_method = serial ｜ parallel〕※1

〔set pd_log_remain_space_check = warn ｜ safe〕※1

〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…〕

〔set pd_log_auto_unload_restart = Y ｜ N〕※1

〔set pd_log_singleoperation = Y ｜ N〕※1

〔set pd_log_rerun_reserved_file_open = Y ｜ N〕※1

〔set pd_log_rerun_swap = Y ｜ N〕※1

〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※1

〔set pd_log_unload_check = Y ｜ N〕※1

〔set pd_log_write_buff_count = ログ出力バッファ面数〕※1

〔set pd_log_full_write_async = Y ｜ N〕※1

〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※1

〔set pd_log_fg_warning_point = 空きシステムログファイルの数 1〔，,空きシステムログファイルの数 2〔，,空きシステムログファイルの数 3〕〕〕※1

(b)　コマンド形式

{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕}}

(13)　シンクポイントダンプファイル

(a)　set 形式

〔set pd_spd_dual = Y ｜ N〕※1

〔set pd_spd_assurance_msg = Y ｜ N〕※1

〔set pd_spd_assurance_count = 有効保証世代数〕※1

〔set pd_spd_reduced_mode = 縮退運転オプション〕※1

〔set pd_spd_reserved_file_auto_open = Y ｜ N〕※1

〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※1

〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※1

(b)　コマンド形式

{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名"〔-b "シンクポイントダンプファイル名"〕}}

{{pdlogadfg -d ssp -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d ssp -g ファイルグループ名 -a "副シンクポイントダンプファイル名"〔-b "副シンクポイントダンプファイル名"〕}}

(14)　サーバ用ステータスファイル

(a)　set 形式

set pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

〔set pd_sts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_2 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_3 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_4 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_5 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_6 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

(15)　サーバ用ステータスファイル（障害発生時）

(a)　set 形式

〔set pd_sts_last_active_file = "論理ファイル名"〕

〔set pd_sts_last_active_side = A ｜ B〕

〔set pd_sts_last_active_subfile = "論理ファイル名"〕

〔set pd_sts_last_active_side_sub = A ｜ B〕

(16)　プラグイン

(a)　コマンド形式

{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

注※1

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

ただし，次に示すオペランドについては，サーバ共通定義ではなく，システム共通定義の同じオペランドの指定値が有効となります。

- pd_optimize_level
- pd_additional_optimize_level
- pd_cwaittime_wrn_pnt
- pd_uap_exerror_log_use
- pd_isolation_level

注※2

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，又はシステム共通定義の同じオペランドの指定値が有効となります。

7 ディクショナリサーバ定義

この章では，ディクショナリサーバ定義の各オペランドの形式について説明します。

7.1　オペランドの形式

7.1.1　オペランドの形式の説明

ディクショナリサーバ定義ではディクショナリサーバの実行環境を定義します。ここでは，ディクショナリサーバ定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

(a)　set 形式

〔set pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数〕※1

〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕※1

〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕※1

〔set pd_svr_castoff_size = 1 サーバプロセスが使用するメモリサイズの上限値〕※1

〔set pd_max_ard_process = 非同期 READ プロセス数〕※1

〔set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数〕

(2)　作業表

(a)　set 形式

〔set pd_work_buff_mode = each ｜ pool ｜ pool2〕※1

〔set pd_work_buff_size = 作業表用バッファ長〕※1

〔set pd_work_buff_expand_limit = 作業表用バッファの増分上限値〕※1

(b)　コマンド形式

pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕...

(3)　システム監視

(a)　set 形式

〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕※1

〔set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値〕※1

(4)　排他制御

(a)　set 形式

〔set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ〕※1

〔set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数〕※1

〔set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和〕※1

〔set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数〕※1

〔set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数〕※1

(5)　バッファ

(a)　set 形式

〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕※1

(6)　共用メモリ

(a)　set 形式

〔set pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ〕※1

(7)　RPC トレース情報

(a)　set 形式

〔set pd_rpc_trace = Y ｜ N〕※1

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※1

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※1

(8) トラブルシュート情報

(a) set 形式

〔set pd_module_trace_max = モジュールトレース格納最大数〕※2

〔set pd_module_trace_timer_level = 0 | 10 | 20〕※2

〔set pd_pth_trace_max = 通信トレース格納最大数〕※1

(9) グローバルバッファ

(a) set 形式

〔set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値〕※1

〔set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値〕※1

〔set pd_dfw_bulk_write = awt | none〕※1

(10) Java

(a) set 形式

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※1

〔set pd_java_castoff = Y | N〕※1

(11) システムログファイル

(a) set 形式

〔set pd_log_dual = Y | N〕※1

〔set pd_log_dual_write_method = serial | parallel〕※1

〔set pd_log_remain_space_check = warn | safe〕※1

〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔,"アンロードログファイルの出力先ディレクトリ"〕…〕

〔set pd_log_auto_unload_restart = Y | N〕※1

〔set pd_log_singleoperation = Y | N〕※1

〔set pd_log_rerun_reserved_file_open = Y ｜ N〕※1

〔set pd_log_rerun_swap = Y ｜ N〕※1

〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※1

〔set pd_log_unload_check = Y ｜ N〕※1

〔set pd_log_write_buff_count = ログ出力バッファ面数〕※1

〔set pd_log_full_write_async = Y ｜ N〕※1

〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※1

〔set pd_log_fg_warning_point = 空きシステムログファイルの数 1〔，,空きシステムログファイルの数 2〔，,空きシステムログファイルの数 3〕〕〕※1

(b)　コマンド形式

{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名" 〔-b "システムログファイル名"〕}}

(12)　シンクポイントダンプファイル

(a)　set 形式

〔set pd_spd_dual = Y ｜ N〕※1

〔set pd_spd_assurance_msg = Y ｜ N〕※1

〔set pd_spd_assurance_count = 有効保証世代数〕※1

〔set pd_spd_reduced_mode = 縮退運転オプション〕※1

〔set pd_spd_reserved_file_auto_open = Y ｜ N〕※1

〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※1

〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※1

(b)　コマンド形式

{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕}}

{{set pdlogadfg -d ssp -g ファイルグループ名〔ONL〕}}

{{set pdlogadpf -d ssp -g ファイルグループ名 -a "副シンクポイントダンプファイル名" 〔-b "副シンクポイントダンプファイル名"〕}}

(13)　サーバ用ステータスファイル

(a)　set 形式

set pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

〔set pd_sts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_2 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_3 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_4 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_5 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_6 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

(14)　サーバ用ステータスファイル（障害発生時）

(a)　set 形式

〔set pd_sts_last_active_file = "論理ファイル名"〕

〔set pd_sts_last_active_side = A ｜ B〕

〔set pd_sts_last_active_subfile = "論理ファイル名"〕

〔set pd_sts_last_active_side_sub = A ｜ B〕

(15)　プラグイン

(a)　コマンド形式

{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

注※1

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

注※2

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，又はシステム共通定義の同じオペランドの指定値が有効となります。

8 バックエンドサーバ定義

この章では，バックエンドサーバ定義の各オペランドの形式について説明します。

8.1　オペランドの形式

8.1.1　オペランドの形式の説明

バックエンドサーバ定義ではバックエンドサーバの実行環境を定義します。ここでは，バックエンドサーバ定義の各オペランドの形式について説明します。オペランドの詳細については，「オペランドの詳細」を参照してください。

●HiRDB システム定義を初めて作成される方へ

太字表示されているオペランドの指定値をまず決めてください。基本的には，太字表示されているオペランドを指定すれば，HiRDB を開始できます。

(1)　プロセス

(a)　set 形式

〔set pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数〕※1

〔set pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕〕※1

〔set pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔〕※1

〔set pd_svr_castoff_size = 1 サーバプロセスが使用するメモリサイズの上限値〕※1

〔set pd_max_open_fds = 1 プロセスがアクセスするファイル及びパイプの最大数〕

〔set pd_max_ard_process = 非同期 READ プロセス数〕※1

〔set pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数〕

(2)　作業表

(a)　set 形式

〔set pd_work_buff_mode = each ｜ pool ｜ pool2〕※1

〔set pd_work_buff_size = 作業表用バッファ長〕※1

〔set pd_work_buff_expand_limit = 作業表用バッファの増分上限値〕※1

(b)　コマンド形式

pdwork -v "HiRDB ファイルシステム領域名称"〔，"HiRDB ファイルシステム領域名称"〕...

(3)　システム監視

(a)　set 形式

〔set pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〕※1

〔set pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値〕※1

(4)　排他制御

(a)　set 形式

〔set pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ〕※1

〔set pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数〕※1

〔set pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和〕※1

〔set pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数〕※1

〔set pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数〕※1

(5)　バッファ

(a)　set 形式

〔set pd_sql_object_cache_size = SQL オブジェクト用バッファ長〕※1

(6)　共用メモリ

(a)　set 形式

〔set pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ〕※1

(7)　RPC トレース情報

(a)　set 形式

〔set pd_rpc_trace = Y ｜ N〕※1

〔set pd_rpc_trace_name = "RPC トレースを取得するファイル名称"〕※1

〔set pd_rpc_trace_size = RPC トレースを取得するファイルの容量〕※1

(8) トラブルシュート情報

(a) set 形式

〔set pd_module_trace_max = モジュールトレース格納最大数〕※2

〔set pd_module_trace_timer_level = 0 | 10 | 20〕※2

〔set pd_pth_trace_max = 通信トレース格納最大数〕※1

(9) グローバルバッファ

(a) set 形式

〔set pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値〕※1

〔set pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値〕※1

〔set pd_dfw_bulk_write = awt | none〕※1

(10) 一時表

(a) set 形式

〔set pd_max_temporary_object_no = ある一時点で使用する一時表と一時インデクスの最大数〕※1

(11) プラグイン

(a) set 形式

〔set pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"〕

(b) コマンド形式

{{〔 pdplgprm -n プラグイン名称 〔 -s 共用メモリサイズ〕〕}}

(12) Java

(a) set 形式

〔set pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"〕※1

〔set pd_java_castoff = Y ｜ N〕※1

(13)　システムログファイル

(a)　set 形式

〔set pd_log_dual = Y ｜ N〕※1

〔set pd_log_dual_write_method = serial ｜ parallel〕※1

〔set pd_log_remain_space_check = warn ｜ safe〕※1

〔set pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕…〕

〔set pd_log_auto_unload_restart = Y ｜ N〕※1

〔set pd_log_singleoperation = Y ｜ N〕※1

〔set pd_log_rerun_reserved_file_open = Y ｜ N〕※1

〔set pd_log_rerun_swap = Y ｜ N〕※1

〔set pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間〕※1

〔set pd_log_unload_check = Y ｜ N〕※1

〔set pd_log_write_buff_count = ログ出力バッファ面数〕※1

〔set pd_log_full_write_async = Y ｜ N〕※1

〔set pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕〕※1

〔set pd_log_fg_warning_point = 空きシステムログファイルの数 1〔，空きシステムログファイルの数 2〔，空きシステムログファイルの数 3〕〕〕※1

(b)　コマンド形式

{{pdlogadfg -d sys -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名"〔-b "システムログファイル名"〕}}

(14)　シンクポイントダンプファイル

(a)　set 形式

〔set pd_spd_dual = Y ｜ N〕※1

〔set pd_spd_assurance_msg = Y ｜ N〕※1

〔set pd_spd_assurance_count = 有効保証世代数〕※1

〔set pd_spd_reduced_mode = 縮退運転オプション〕※1

〔set pd_spd_reserved_file_auto_open = Y ｜ N〕※1

〔set pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長〕※1

〔set pd_log_sdinterval = システムログ出力量〔，経過時間〕〕※1

(b)　コマンド形式

{{pdlogadfg -d spd -g ファイルグループ名〔ONL〕}}

{{pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名"〔-b "シンクポイントダンプファイル名"〕}}

{{set pdlogadfg -d ssp -g ファイルグループ名〔ONL〕}}

{{set pdlogadpf -d ssp -g ファイルグループ名 -a "副シンクポイントダンプファイル名"〔-b "副シンクポイントダンプファイル名"〕}}

(15)　サーバ用ステータスファイル

(a)　set 形式

set pd_sts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

〔set pd_sts_file_name_2 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_3 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_4 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_5 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_6 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"〕

〔set pd_sts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_2 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_3 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_4 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_5 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_6 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

〔set pd_sts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"〕

(16)　サーバ用ステータスファイル（障害発生時）

(a)　set 形式

〔set pd_sts_last_active_file = "論理ファイル名"〕

〔set pd_sts_last_active_side = A ｜ B〕

〔set pd_sts_last_active_subfile = "論理ファイル名"〕

〔set pd_sts_last_active_side_sub = A ｜ B〕

(17)　バックエンドサーバ接続保持機能

(a)　set 形式

〔set pd_bes_connection_hold = Y ｜ N〕※1

〔set pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間〕※1

注※1

このオペランドを省略すると，サーバ共通定義の同じオペランドの指定値が有効となります。

注※2

このオペランドを省略すると，サーバ共通定義，ユニット制御情報定義，又はシステム共通定義の同じオペランドの指定値が有効となります。

9 オペランドの詳細

この章では，オペランドの一覧，及び各オペランドの内容について説明します。

9.1　オペランドの一覧

9.1.1　オペランド名および定義内容の一覧

HiRDB システム定義で定義する内容，及びそれに対応するオペランド名を示します。また，各オペランドが指定できる定義名と再開始時の変更可否を次の表に示します。異なる定義で同じオペランドを指定した場合の優先順位については，次の章を参照してください。

- HiRDB/シングルサーバの場合：「各定義の関係」
- HiRDB/パラレルサーバの場合：「各定義の関係」

なお，次に示す定義のオペランドはここには記載していません。これらの定義のオペランドは HiRDB の再開始前（計画停止，強制終了，又は異常終了後）に変更できます。

- UAP 環境定義

各表の凡例と注について，次に示します。

（凡例）

○：指定値を変更できます。

×：指定値を変更できません。

◎：整合性を保って変更してください。

空白：該当しません。

SYS：システム共通定義

UNT：ユニット制御情報定義

SVR：サーバ共通定義

SDS：シングルサーバ定義

FES：フロントエンドサーバ定義

DS ：ディクショナリサーバ定義

BES：バックエンドサーバ定義

FORCE：強制終了，異常終了後の変更可否（系切り替えや HiRDB 稼働中に OS を再起動して HiRDB を停止した場合を含む）

PLAN：計画停止後の変更可否

注※1

オペランドの追加だけができます。オペランドの削除，変更はできません。

注※2

オペランドの指定値を小さくすると，HiRDB を再開始できない場合があります。その場合は，指定値を変更前の値に戻して再開始してください。

注※3

オペランドの指定値を大きくすると，HiRDB を再開始できない場合があります。その場合は，指定値を変更前の値に戻して再開始してください。

注※4

オペランドの値を変更するときに制限事項があります。詳細はオペランドの説明を参照してください。

注※5

HiRDB Datareplicator（抽出側）と連携している場合に注意が必要なオペランドです。このオペランドの指定を追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度停止してから再開始してください。HiRDB Datareplicator を稼働したまま，この定義を変更して HiRDB を再度開始すると，HiRDB Datareplicator による抽出が失敗することがあります。

注※6

次に示す HiRDB システム定義のオペランドの値は，クライアント環境定義でクライアントごとに変更できます。クライアントごとに値を変更する場合は，クライアント環境定義の該当するオペランドを指定してください。クライアント環境定義については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

HiRDB システム定義のオペランド	クライアント環境定義のオペランド
pd_additional_optimize_level	PDADDITIONALOPTLVL
pd_cwaittime_wrn_pnt	PDCWAITTIMEWRNPNT
pd_delete_reserved_word_file	PDDELRSVWDFILE
pd_ha_transaction	PDHATRNQUEUING
pd_hash_table_size	PDHASHTBLSIZE
pd_lck_wait_timeout	PDLCKWAITTIME
pd_optimize_level	PDSQLOPTLVL
pd_space_level	PDSPACELVL
pd_uap_exerror_log_param_size	PDUAPEXERLOGPRMSZ
pd_uap_exerror_log_use	PDUAPEXERLOGUSE
pd_watch_pc_client_time	PDSWATCHTIME
pd_isolation_level	PDISLLVL

注※7

システム共通定義（pdsys）に pd_dbbuff_modify=Y を指定している場合，×となります。

注※8

オプションによって変更可否が異なります。各オペランドの説明を参照してください。

(1)　システム構成に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB 識別子を指定します。	pd_system_id	○							×	×
HiRDB のポート番号を指定します。	pd_name_port	○							◎	◎
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名を指定します。	pd_master_file_name	○							×	×
拡張ユニット名称を指定します。	pd_system_expand_unit	○							×	×
ユニットをホストに割り当てる指定をします。	pdunit	○							×	○※8
サーバをホストに割り当てる指定をします。	pdstart	○							×	○※8
ユニット識別子を指定します。	pd_unit_id		○						×	×
標準ホスト名を指定します。	pd_hostname		○						×	○

(2)　同時実行最大数に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
最大同時接続数を指定します。	pd_max_users	○							×	○
1 ユニット内で同時に起動するサーバプロセス数の最大数を指定します。	pd_max_server_process	○	○						○	○
1 トランザクションで同時にアクセスできる表数の最大数を指定します。	pd_max_access_tables	○							○※3	○

(3)　プロセスに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
バックエンドサーバ当たりの最大起動プロセス数を指定します。	pd_max_bes_process			○				○	×	○※2

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
ディクショナリサーバ当たりの最大起動プロセス数を指定します。	pd_max_dic_process			○			○		×	○※2
あらかじめサーバ開始時から起動しておく常駐プロセス数を指定します。	pd_process_count			○	○	○	○	○	○※2	○※2
HiRDB の非常駐サーバプロセスの停止処理を行う間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。	pd_server_cleanup_interval			○	○	○	○	○	○	○
サーバプロセスが使用するメモリサイズの上限値を指定します。使用メモリサイズがここで指定した値を超えた場合，そのサーバプロセスを終了させます。	pd_svr_castoff_size			○	○	○	○	○	○	○
HiRDB のプロセスがアクセスするファイル及びパイプの最大数を指定します。	pd_max_open_fds				○			○	○	○
非同期 READ 機能を使用する場合に，非同期 READ 処理に必要なプロセス数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（バックエンドサーバ又はディクショナリサーバ）当たりのプロセス数を指定します。	pd_max_ard_process			○	○		○	○	○	○
デファードライト処理の並列 WRITE 機能を使用する場合，並列処理するプロセス数を指定します。	pd_dfw_awt_process			○	○		○	○	○	○

(4)　作業表に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL 実行時に，一時的に情報を格納する作業表を作成する場合のバッファの確保方式を指定します。	pd_work_buff_mode			○	○		○	○	○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL 実行時に使用する作業表用バッファ長を指定します。	pd_work_buff_size			○	○		○	○	○	○
作業表用バッファが不足したとき，増分する作業表用バッファの上限値を指定します。	pd_work_buff_expand_limit			○	○		○	○	○	○
作業表用ファイル用の HiRDB ファイルシステム領域の名称を指定します。	pdwork				○		○	○	×	×

(5)　HiRDB の開始方法に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB の開始方法を指定します。	pd_mode_conf	○							○	○
pdstart コマンドの完了待ち時間を指定します。	pd_system_complete_wait_time	○							○	○
HiRDB の開始準備処理の最大待ち時間を指定します。	pd_start_time_out	○							○	○
HiRDB の再開始処理に失敗したときの異常終了回数の上限を指定します。	pd_term_watch_count	○	○						○	○

(6)　縮退起動に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
縮退起動をするかどうかを指定します。	pd_start_level	○							○	○
縮退起動開始の連絡待ち時間を設定します。	pd_reduced_check_time	○							○	○
HiRDB の開始時，障害などで開始しないユニットを指定します。	pd_start_skip_unit	○							○	○

(7)　HiRDB の処理方式に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
データベースの更新内容をファイルに反映するタイミングを指定します。	pd_dbsync_point	○							×	○
次に示す RD エリアの更新内容をファイルに反映するタイミングを指定します。 • マスタディレクトリ用 RD エリア • データディレクトリ用 RD エリア • データディクショナリ用 RD エリア • データディクショナリ LOB 用 RD エリア • レジストリ用 RD エリア • レジストリ LOB 用 RD エリア	pd_system_dbsync_point	○							×	○
HiRDB のプロセスが異常終了したときに後処理をする後処理プロセスを HiRDB の開始時に起動しておくかどうかを指定します。	pd_process_terminator	○							○	○
HiRDB の開始時に起動する後処理プロセスの数を指定します。	pd_process_terminator_max	○							○	○
スレッド間ロックの解放通知方式を指定します。	pd_thdlock_wakeup_lock	○	○						○	○
データベース検索時のページアクセス方式を指定します。	pd_pageaccess_mode	○							○	○
RD エリアの閉塞状態のチェック方式を指定します。	pd_cmdhold_precheck	○							○	○
RD エリア（マスタディレクトリ用 RD エリアを除く）の入出力エラーが発生したときの HiRDB の処理を指定します。	pd_db_io_error_action	○	○						○	○
CONNECT 失敗時にエラーの要因を隠すかどうかを指定します。	pd_connect_errmsg_hide	○							○	○
サーバプロセスの強制終了が発生した場合に，出力されるエラーメッセージを変更するかどうかを指定します。	pd_cancel_down_msgchange	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
サーバプロセスの起動，停止処理で，1 回の要求で起動，停止するサーバプロセス数を指定します。	pd_server_process_adjust_limit	○							○	○
サーバプロセスを複数回に分けて起動，又は停止する間隔を指定します。	pd_server_process_adjust_intvl	○							○	○
HiRDB コマンドのリモート実行やファイル転送時に使用するコマンドの種類を指定します。	pd_cmd_rmode	○							○	○
HiRDB コマンドのリモート実行やファイル転送時に使用する ssh，scp コマンドの設定ファイルを指定します。	pd_cmd_rconfig	○	○						○	○
リモート操作ログを取得するかどうかを指定します。	pd_cmd_rlog	○							○	○

(8)　全面回復処理に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
全面回復処理をするときの回復処理プロセス（REDO プロセス）数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（ディクショナリサーバ又はバックエンドサーバ）当たりの回復処理プロセス（REDO プロセス）数になります。	pd_max_recover_process	○	○						○	○
全面回復処理で，シンクポイント以降に更新されたページをすべてデータベースに書き出すかを指定します。	pd_redo_allpage_put	○							○	○

(9)　トランザクション決着処理に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
トランザクションが COMMIT 一相目未完了状態でユニットが異常終了した場合，ユニットの再開始時にこのトランザクションからブランチした未決着状態のトランザクションを自動決着するかどうかを指定します。	pd_trn_rerun_branch_auto_decide	○							○	○
ブランチしたトランザクションに自動決着指示を送信した場合，送信が何らかの原因で失敗したときに決着指示を再送するまでの時間を秒単位で指定します。	pd_trn_send_decision_intval_sec	○							○	○
ブランチしたトランザクションに自動決着指示を送信した場合，送信が何らかの原因で失敗したときに決着指示を再送するまでの時間を分単位で指定します。	pd_trn_send_decision_interval	○							○	○
ブランチしたトランザクションに自動決着指示を送信してから，決着完了の連絡が戻ってくるまでの待ち時間を指定します。	pd_trn_send_decision_retry_time	○							○	○
HiRDB サーバプロセス上で実行されるトランザクションの同期点処理で，トランザクションブランチ間で行う通信（プリペア，コミット指示，又は応答など）の受信待ち時間の最大値を指定します。	pd_trn_watch_time	○							○	○
トランザクション回復メッセージキューサイズを指定します。	pd_trn_rcvmsg_store_buflen	○	○						○	○
HiRDB/パラレルサーバのコミットメント制御で一相コミットを使用するかどうかを指定します。	pd_trn_commit_optimize	○							○	○
フロントエンドサーバがトランザクションブランチからのロールバック完了応答を受信するまでの，待ち時間の最大値を指定します。また，通信障害発生時に，ロールバック指示の送信をリトライし続ける時間を指定します。	pd_trn_rollback_watch_time	○							○	○

(10)　SQL の仕様に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
演算中のエラー抑止をするかどうかを指定します。	pd_overflow_suppress	○							○	○
空白変換機能を使用する場合の空白変換レベルを指定します。	pd_space_level ※6	○							○	○
DECIMAL 型の符号正規化機能を使用するかどうかを指定します。	pd_dec_sign_normalize	○							×	○
演算結果が 29 けた以下の DECIMAL 型となるとき，演算結果の精度の最大値を指定します。	pd_sql_dec_op_maxprec	○							○	○
SQL 文中に集合演算子 MINUS を使用するかどうかを指定します。	pd_sql_mode	○							○	○
PCTFREE(表オプション)の省略値を指定します。	pd_ddl_tbl_pctfree	○							○	○
クラスターキーのない FIX 表における，PCTFREE(表オプション)の省略値を指定します。	pd_ddl_tbl_fix_pctfree	○							○	○
PCTFREE(インデクスオプション)の省略値を指定します。	pd_ddl_idx_pctfree	○							○	○
列回復制約の省略値を指定します。	pd_ddl_tbl_recovery	○							○	○

(11)　SQL の最適化に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL 最適化オプションを指定します。	pd_optimize_level※6	○				○			○	○
SQL 拡張最適化オプションを指定します。	pd_additional_optimize_level※6	○				○			○	○
SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を適用した場合のハッシュ表サイズを指定します。	pd_hash_table_size※6	○							×	×

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
作業表を使用する SQL を実行する場合，次に示す HiRDB の処理方式を指定します。 • AND の複数インデクス利用時の排他取得方式 • 作業表用バッファ長の自動増分時のメッセージ出力	pd_work_table_option	○							○	○
バックエンドサーバの中からフロータブルサーバとして使用するサーバを指定します。	pd_floatable_bes					○			○	○
バックエンドサーバをフロータブルサーバとして使用しないことを指定します。	pd_non_floatable_bes					○			○	○

(12)　拡張 SQL エラー情報出力機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
拡張 SQL エラー情報出力機能を使用するかどうかを指定します。	pd_uap_exerror_log_use※6	○			○	○			○	○
SQL エラーレポートファイルの格納ディレクトリを指定します。	pd_uap_exerror_log_dir	○	○						○	○
SQL エラーレポートファイルの最大サイズを指定します。	pd_uap_exerror_log_size	○	○						○	○
クライアントエラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。	pd_uap_exerror_log_param_size※6	○	○						○	○

(13)　SQL 予約語削除機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL 予約語削除機能を使用する場合，SQL 予約語削除ファイルの名称を指定します。	pd_delete_reserved_word_file※6	○							○	○

(14)　SQL からのコマンド実行に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL の CALL COMMAND 文でコマンド及びユティリティを実行できる認可識別子を指定します。	pd_sql_command_exec_users	○							○	○

(15)　SQLSTATE の細分化に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQLSTATE の値を詳細に出力するかどうかを指定します。	pd_standard_sqlstate	○							○	○

(16)　絞込み検索に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
同時にリストを所有できるユーザの最大数を指定します。	pd_max_list_users	○							○※3	○
１ユーザ当たりのリスト作成数を指定します。	pd_max_list_count	○							×	○
リストの初期化（削除）時期を指定します。通常，HiRDB を開始（再開始を含む）するときにリストが初期化されます。このオペランドでリストの初期化時期を変更できます。	pd_list_initialize_timing	○							○	○
リストの名前を変更したときに，変更したリストの行数を SQL 連絡領域の SQLERRD[2]に返却するかどうかを指定します。	pd_list_rowcount_in_rename	○							○	○

(17)　システム監視に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
次に示すユティリティの実行時間を監視する場合，その監視時間を分単位で指定します。	pd_utl_exec_time	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
• データベース作成ユティリティ（pdload コマンド） • データベース再編成ユティリティ（pdrorg コマンド） • 空きページ解放ユティリティ（pdreclaim コマンド） • グローバルバッファ常駐化ユティリティ（pdpgbfon コマンド） このオペランドで指定した監視時間を超えてもユティリティが終了しない場合，実行中のユティリティを強制終了して，無応答原因を調査するための障害情報を出力します。										
運用コマンド及びユティリティの実行時間を監視する場合，その監視時間を分単位で指定します。pd_utl_exec_time が対象とするユティリティよりも多くのユティリティ及びコマンドが対象になります。	pd_cmd_exec_time	○							○	○
メッセージキュー中のメッセージが，このオペランドに指定した時間を過ぎても取り出されない場合，警告メッセージ（KFPS00888-W 又は KFPS00889-E）を出力します（メッセージキュー監視機能）。	pd_queue_watch_time	○							○	○
pd_queue_watch_time オペランドの値以上の時間，メッセージキューからメッセージが取り出されない場合の HiRDB の処理を指定します。	pd_queue_watch_timeover_action	○							○	○
運用コマンド・ユティリティ実行時に，プロセスを割り当てるまでの監視時間を指定します。	pd_cmd_process_conwaittime	○							○	○
サーバプロセスの異常終了回数がこのオペランドに指定した値を超えた場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を異常終了させます（プロセスの異常終了回数監視機能）。	pd_down_watch_proc	○	○						○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
ほかのホストの稼働状況を監視するために問い合わせをする時間間隔を指定します。	pd_host_watch_interval	○							○	○
Windows 対応の HiRDB クライアントからの要求に対する応答をサーバが返してから，次に HiRDB クライアントからの要求があるまでの，サーバ側の最大待ち時間を指定します。	pd_watch_pc_client_time※6			○	○	○			○	○
シンクポイントダンプの有効化処理が連続してスキップされるとき，そのスキップ回数（1 トランザクション中のスキップ回数）の上限値を指定します。	pd_spd_syncpoint_skip_limit			○	○	○	○	○	○	○
デファードライト処理によってシンクポイントダンプの取得が遅れ，シンクポイントダンプの有効化処理がスキップされるとき，そのスキップ回数（1 トランザクション中のスキップ回数）の上限値を指定します。	pd_dfw_syncpoint_skip_limit			○	○		○	○	○	○
リソースの使用率に関する警告メッセージを出力するかどうかを指定します。	pd_watch_resource	○							○	○
pd_max_users オペランドで指定した最大同時接続数に対する警告メッセージを出力するかどうかを指定します。	pd_max_users_wrn_pnt	○							○	○
pd_max_access_tables オペランドで指定した同時アクセス可能実表数に対する警告メッセージを出力するかどうかを指定します。	pd_max_access_tables_wrn_pnt	○							○	○
pd_max_rdarea_no オペランドで指定した RD エリアの最大数に対する警告メッセージを出力するかどうかを指定します。	pd_max_rdarea_no_wrn_pnt	○							○	○
pd_max_file_no オペランドで指定した RD エリアを構成する HiRDB ファイルの最大数に対する警告メッセージを出力するかどうかを指定します。	pd_max_file_no_wrn_pnt	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
pdwork オペランドで指定した作業表用ファイル用の HiRDB ファイルシステム領域の使用率に対する警告メッセージを出力するかどうかを指定します。	pdwork_wrn_pnt	○							○	○
pd_max_list_users オペランドで指定したリスト作成ユーザ数に対する警告メッセージを出力するかどうかを指定します。	pd_max_list_users_wrn_pnt	○							○	○
pd_max_list_count オペランドで指定した１ユーザ当たりのリスト作成数に対する警告メッセージを出力するかどうかを指定します。	pd_max_list_count_wrn_pnt	○							○	○
サーバ内のリスト作成数に対する警告メッセージを出力するかどうかを指定します。	pd_rdarea_list_no_wrn_pnt	○							○	○

(18)　排他制御に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
デッドロック情報を出力するかどうかを指定します。	pd_lck_deadlock_info	○	○						○	○
排他要求が待ち状態になってから解除されるまでの待ち時間を指定します。	pd_lck_wait_timeout	○	○						○	○
WITHOUT LOCK NOWAIT 指定の検索を実行したときの処理方式を指定します。	pd_nowait_scan_option	○							○	○
排他待ちユーザ数警告メッセージの出力契機を指定します。	pd_lck_queue_limit	○							○	○
デッドロックの優先順位を制御するかどうかを指定します。	pd_deadlock_priority_use	○							○	○
コマンドのデッドロックプライオリティ値を指定します。	pd_command_deadlock_priority	○							○	○
インデクスキー値の排他資源の作成方法を指定します。	pd_key_resource_type	○							×	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
サーバ当たりの排他制御用プールの大きさを指定します。	pd_lck_pool_size			○	○		○	○	○※2	○※2
フロントエンドサーバの排他制御用プールの大きさを指定します。	pd_fes_lck_pool_size			○		○			○	○
排他制御処理を分散する場合，サーバ当たりの排他制御用プールの分割数を指定します。	pd_lck_pool_partition			○	○		○	○	×	○
排他制御処理を分散する場合，フロントエンドサーバの排他制御用プールの分割数を指定します。	pd_fes_lck_pool_partition			○		○			×	○
トランザクションを越えて保持する表，RD エリア，インナレプリカ構成管理，及びレプリカグループ構成管理の排他資源数を指定します。	pd_lck_until_disconnect_cnt			○	○		○	○	×	○
UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。	pd_max_open_holdable_cursors			○	○		○	○	×	×
排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。	pd_lck_hash_entry			○	○	○	○	○	○	○
B-tree インデクスの排他制御方式を指定します（インデクスキー値無排他を実施するかどうかを指定します）。	pd_indexlock_mode	○							×	○
コミットしていない削除データの排他待ち制御方式を指定します。	pd_lock_uncommited_delete_data	○							×	◎
行データ挿入の際に，行識別子を再利用するかどうかを指定します。	pd_dbreuse_remaining_entries	○							○	○
デッドロックの発生を監視するかどうかを指定します。	pd_lck_deadlock_check	○	○						○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
デッドロックの発生を監視する間隔を指定します。	pd_lck_deadlock_check_interval	○	○						○	○
WITHOUT ROLLBACK オプション指定表のインデクス構成列の更新可否を指定します。	pd_idx_without_rollback	○							○	○
SQL 文のデータ保証レベルを指定します。	pd_isolation_level※6	○				○			○	○
残存エントリによる排他待ち情報のメッセージ出力の有無を指定します。	pd_unique_indexlock_info	○							○	○
ユニークインデクスの一意性制約保証処理での排他制御方式を指定します。	pd_unique_check_mode	○							×	○
デッドロック SQL 情報を出力するかどうかを指定します。	pd_lck_deadlock_sql_info	○	○						○	○
デッドロック監視処理の最大処理時間を指定します。	pd_lck_deadlock_check_timeout	○	○						○	○
デッドロック監視処理中断時の監視処理休止時間を指定します。	pd_lck_deadlock_check_pause	○	○						○	○

(19)　バッファに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL オブジェクトを置くバッファの大きさを指定します。	pd_sql_object_cache_size	○		○	○	○	○	○	×	○
表定義情報用バッファ，ビュー解析情報用バッファ，ユーザ定義型情報用バッファ，及びルーチン定義情報用バッファの管理領域を HiRDB 開始時に一括して確保するか，又はトランザクション開始時に確保するかを指定します。	pd_def_buf_control_area_assign	○							○	○
1 スレッドが使用するスタックサイズの最大値を指定します。	pd_thread_max_stack_size	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
1 スレッド当たりのスタック領域の拡張サイズを指定します。	pd_thread_stack_expand_size	○	○						○	○
表定義情報を置くバッファの大きさを指定します。	pd_table_def_cache_size			○	○	○			○※2，※3	○※2
ユーザ権限情報を置くバッファの大きさを指定します。	pd_auth_cache_size			○	○	○			○※2，※3	○※2
ビュー解析情報を置くバッファの大きさを指定します。	pd_view_def_cache_size			○	○	○			○※2，※3	○※2
ユーザ定義型情報を置くバッファの大きさを指定します。	pd_type_def_cache_size			○	○	○			○	○
ルーチン定義情報を置くバッファの大きさを指定します。	pd_routine_def_cache_size			○	○	○			○	○
レジストリ情報を置くバッファの大きさを指定します。	pd_registry_cache_size			○	○	○			○	○

(20)　共用メモリに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB が使用する共用メモリを固定するかどうかを指定します。	pd_shmpool_attribute	○	○						○	○
HiRDB が使用する共用メモリの制御方法を指定します。	pd_shmpool_control	○							○	○
グローバルバッファが使用する共用メモリを実メモリ上に固定するかどうかを指定します。	pd_dbbuff_attribute	○	○						○	○
シングルサーバが使用する共用メモリの大きさを指定します。	pd_sds_shmpool_size			○	○				○※2	○※2
ディクショナリサーバが使用する共用メモリの大きさを指定します。	pd_dic_shmpool_size			○			○		○※2	○※2
バックエンドサーバが使用する共用メモリの大きさを指定します。	pd_bes_shmpool_size			○				○	○※2	○※2
グローバルバッファ用の共用メモリセグメントの大きさの上限値を指定します。	SHMMAX	○	○						○※7	○※7

(21)　RD エリアに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
RD エリアの最大数を指定します。	pd_max_rdarea_no	○							×	○
RD エリアを構成する HiRDB ファイルの最大数を指定します。	pd_max_file_no	○							○※2，※3	○
RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I），又は増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）の出力契機を指定します。	pd_rdarea_warning_point	○							○	○
RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I）を出力するかどうかを指定します。	pd_rdarea_warning_point_msgout	○							○	○
LOB 用 RD エリアのセグメント使用率通知メッセージ（KFPH22040-W）の出力契機を指定します。	pd_stored_lob_data_wrn_pnt	○							○	○
RD エリアの自動増分機能を適用している場合，自動的に RD エリアの容量を拡張する自動増分契機を指定します。	pd_rdarea_extension_timing	○							○	○
RD エリアの自動増分が発生した場合に増分領域を初期化するかどうかを指定します。	pd_rdarea_expand_format	○							○	○
RD エリアの自動増分機能を使用している場合，自動増分の対象 HiRDB ファイルを指定します。	pd_rdarea_extension_file	○							○	○
RD エリアのオープン属性の DEFER 属性，又は SCHEDULE 属性を使用するかどうかを指定します。	pd_rdarea_open_attribute_use	○							○	○
RD エリアのオープン契機の標準値を指定します。	pd_rdarea_open_attribute	○							○	○
共用 RD エリアを使用するかどうかを指定します。	pd_shared_rdarea_use	○							×	○
RD エリアへのアクセス時にファイルアクセスエラーが発生した場合の動作を指定します。	pd_db_access_error_action	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
RD エリアへのアクセス時に物理エラー（入出力エラー及びファイルオープンエラー）が発生した場合の動作を指定します。	pd_db_hold_action	○							○	○
RD エリアアクセス時にページの読み込み内容に不整合を検出した場合（KFPH23202-W 出力時）の読み込みリトライ回数を指定します。	pd_dbbuff_read_retry_count	○							○	○
RD エリアアクセス時にページの読み込み内容に不整合を検出した場合（KFPH23202-W 出力時）の読み込み間隔を指定します。	pd_dbbuff_read_retry_intvl	○							○	○
RD エリアアクセス時にページの読み込み内容に不整合を検出し（KFPH23202-W 出力時），読み込みリトライしてもページの読み込み内容の不整合が解消しなかった場合の動作を指定します。	pd_db_timestamp_invalid_action	○							○	○

(22) グローバルバッファに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
グローバルバッファの LRU 管理方式を指定します。	pd_dbbuff_lru_option	○							○	○
BINARY 型を指定した列にアクセスする UAP を実行する場合，グローバルバッファの LRU 管理をするかどうかを指定します。	pd_dbbuff_binary_data_lru	○							○	○
グローバルバッファを，論理セクタ長が 4096 バイトのディスク対応にするかどうかを指定します。	pd_dbbuff_dev_sector_size	○								○
グローバルバッファを動的変更するかどうかを指定します。	pd_dbbuff_modify	○							×	×
デファードライトトリガのトリガ契機を更新バッファ比率で指定します。	pd_dbbuff_rate_updpage	○							○	○
DB バッファ制御情報トレースの取得レベルを指定します	pd_dbbuff_trace_level	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
グローバルバッファに割り当てる RD エリアを定義します。	pdbuffer	○							×	○
動的追加用グローバルバッファ数の上限値を指定します。	pd_max_add_dbbuff_no			○	○		○	○	×	○
動的追加用共用メモリセグメント数の上限値を指定します。	pd_max_add_dbbuff_shm_no			○	○		○	○	×	○
デファードライト処理の一括出力機能を適用するかどうかを指定します。	pd_dfw_bulk_write			○	○		○	○	○	○

(23)　インメモリデータ処理に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
インメモリ RD エリアの最大数を指定します。	pd_max_resident_rdarea_no	○							○	○
インメモリデータバッファが使用する共用メモリセグメントの最大数を指定します。	pd_max_resident_rdarea_shm_no	○							○	○

(24)　表又はインデクスの予約数に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
空き領域の再利用機能を使用する表数を指定します。	pd_assurance_table_no	○							○	○
使用するインデクスの数を指定します。	pd_assurance_index_no	○							○	○

(25)　参照制約及び検査制約に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
参照制約及び検査制約で，制約名の定義を制約定義の前後どちらにするかを指定します。	pd_constraint_name	○							○	○
参照制約及び検査制約使用時に指定します。	pd_check_pending	○							○	○

(26)　一時表に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
サーバごとに，ある一時点で使用する一時表と一時インデクスの最大数を指定します。	pd_max_temporary_object_no	○		○	○			○	○	○
一時表用 RD エリアの初期化を HiRDB 開始時ではない時に変更する場合に指定します。	pd_tmp_table_initialize_timing	○							○	○

(27)　システムログファイルに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
システムログファイルを二重化するかどうかを指定します。	pd_log_dual※5			○	○	○	○	○	×	×
システムログの並列出力機能を使用するかどうかを指定します。	pd_log_dual_write_method			○	○	○	○	○	○	○
システムログファイルの空き率が警告値未満になったときの HiRDB の処理方式を指定します（システムログファイルの空き容量監視機能）。	pd_log_remain_space_check			○	○	○	○	○	○	○
システムログの自動ログアンロード機能を使用する場合に，アンロードログファイルの出力先ディレクトリ名又は出力先 HiRDB ファイルシステム領域名を絶対パス名で指定します。	pd_log_auto_unload_path				○	○	○	○	×	×
障害で自動ログアンロード機能が停止した旨のメッセージ（KFPS01150-E メッセージ）出力後，次回のシステムログファイルのスワップ時に自動ログアンロード機能を再開始するかどうかを指定します。	pd_log_auto_unload_restart			○	○	○	○	○	○	○
システムログファイルの片系運転をするかどうかを指定します。	pd_log_singleoperation			○	○	○	○	○	×	×
オープンされたファイルがすべてスワップできない状態の場合，予約ファイルを使用するかどうかを指定します。	pd_log_rerun_reserved_file_open			○	○	○	○	○	○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
再開始時にシステムログファイルをスワップするかどうかを指定します。	pd_log_rerun_swap			○	○	○	○	○	○	○
システムログファイルのスワップが完了するまでの待ち時間を指定します。	pd_log_swap_timeout			○	○	○	○	○	○	○
システムログファイルをスワップ先にできる状態にする条件から，アンロード済み状態であることを除くかどうかを指定します。アンロード済み状態であることを除くと，データベースの回復にシステムログを使用しないときの運用となります。	pd_log_unload_check			○	○	○	○	○	○	○
システムログの出力に使用するバッファの面数を指定します。	pd_log_write_buff_count			○	○	○	○	○	○※3	○
ログ出力バッファが満杯になった場合のシステムログファイルへの出力方式を指定します。	pd_log_full_write_async			○	○	○	○	○	○	○
一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限を指定します。	pd_log_auto_expand_size			○	○	○	○	○	○	○
空きシステムログファイルの数が少なくなったことを警告するメッセージ（KFPS01165-W）の出力契機を指定します。	pd_log_fg_warning_point			○	○	○	○	○	○	○
システムログファイル出力時にブロックチェック用の情報を付加してシステムログを出力するかを指定します。	pd_log_block_check			○	○				○	○
システムログファイルを構成するファイルグループの名称を指定します。	pdlogadfg -d sys ※5				○	○	○	○	○※1	○※1
ファイルグループに割り当てるシステムログファイルの名称を指定します。	pdlogadpf -d sys ※5				○	○	○	○	○※1	○※1
リードレプリカの自動反映方式を使用する場合に，参照 DB で反映対象とする更新 DB のシステムログファイル（反映対象システ	pdlogadfg -d rrp				○				○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
ムログファイル）のファイルグループの名称を指定します。										
リードレプリカの自動反映方式を使用する場合に，参照 DB で反映対象とする更新 DB のファイルグループに割り当てるシステムログファイル（反映対象システムログファイル）の名称を指定します。	pdlogadpf -d rrp				○				○	○

(28)　シンクポイントダンプファイルに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
シンクポイントダンプファイルを二重化するかどうかを指定します。	pd_spd_dual			○	○	○	○	○	×	×
シンクポイントダンプが有効化されたときにメッセージを出力するかどうかを指定します。	pd_spd_assurance_msg			○	○	○	○	○	○	○
有効保証世代数を指定します。	pd_spd_assurance_count			○	○	○	○	○	×	×
縮退運転機能を使用するかどうかを指定します。	pd_spd_reduced_mode			○	○	○	○	○	×	○
予約ファイルを自動的にオープンするかどうかを指定します。	pd_spd_reserved_file_auto_open			○	○	○	○	○	×	○
シンクポイントダンプの入出力に使用するバッファの大きさを指定します。	pd_spd_max_data_size			○	○	○	○	○	×	×
シンクポイントダンプを取得する間隔を指定します。	pd_log_sdinterval			○	○	○	○	○	○	○
シンクポイントダンプファイルを構成するグループ名を指定します。	pdlogadfg -d spd				○	○	○	○	○※1	○※1
ファイルグループに割り当てるシンクポイントダンプファイルを指定します。	pdlogadpf -d spd				○	○	○	○	○※1	○※1

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
副シンクポイントダンプファイルを構成するグループ名を指定します。	pdlogadfg -d ssp				○	○	○	○	○※1	○※1
ファイルグループに割り当てる副シンクポイントダンプファイルを指定します。	pdlogadpf -d ssp				○	○	○	○	○※1	○※1

(29)　ステータスファイルに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
ユニット用ステータスファイル名を指定します。	pd_syssts_file_name_1～7		○						×	×
ユニット用副ステータスファイル名を指定します。	pd_syssts_subfile_name_1～7		○						×	×
現用として使用するユニット用ステータスファイル名を指定します。	pd_syssts_last_active_file		○						○	○
再開始時に，前回の稼働時に正常だったユニット用ステータスファイルの系を指定します。	pd_syssts_last_active_side		○						○	○
現用として使用するユニット用副ステータスファイル名を指定します。	pd_syssts_last_active_subfile		○						○	○
再開始時に，前回の稼働時に正常だったユニット用副ステータスファイルの系を指定します。	pd_syssts_last_active_side_sub		○						○	○
サーバ用ステータスファイル名を指定します。	pd_sts_file_name_1～7				○	○	○	○	×	×
サーバ用副ステータスファイル名を指定します。	pd_sts_subfile_name_1～7				○	○	○	○	×	×
現用として使用するサーバ用ステータスファイル名を指定します。	pd_sts_last_active_file				○	○	○	○	○	○
再開始時に，前回の稼働時の正常だったサーバ用ステータスファイルの系を指定します。	pd_sts_last_active_side				○	○	○	○	○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
現用として使用するサーバ用副ステータスファイル名を指定します。	pd_sts_last_active_subfile				○	○	○	○	○	○
再開始時に，前回の稼働時の正常だったサーバ用副ステータスファイルの系を指定します。	pd_sts_last_active_side_sub				○	○	○	○	○	○

(30)　メッセージログファイルに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
メッセージログの出力先ユニットを指定します。	pd_mlg_msg_log_unit	○							×	×
メッセージログファイルの最大容量を指定します。	pd_mlg_file_size	○							○	○

(31)　統計情報に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB の開始時から統計情報を取得するかどうかを指定します。	pd_statistics	○							○	○
統計ログファイルの最大容量を指定します。	pd_stj_file_size	○	○						○※3	○
統計ログバッファの大きさを指定します。	pd_stj_buff_size	○	○						○※3	○
SQL オブジェクト用バッファの統計情報を取得するタイミングを指定します。	pd_sqlobject_stat_timing	○							○	○
HiRDB の開始時から統計情報を取得する場合に指定します。	pdstbegin	○							○	○
HiRDB の開始時から CONNECT 及び DISCONNECT に関する履歴情報を取得する場合に指定します。	pdhibegin	○							○	○

(32)　RPC トレース情報に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
RPC トレース情報を取得するかどうかを指定します。	pd_rpc_trace	○	○	○	○	○	○	○	○	○
RPC トレース情報を取得するファイル名を指定します。	pd_rpc_trace_name	○	○	○	○	○	○	○	○	○
RPC トレース情報を取得するファイルの容量を指定します。	pd_rpc_trace_size	○	○	○	○	○	○	○	○	○

(33)　PRF トレース情報に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
PRF トレース情報を取得するかどうかを指定します。	pd_prf_trace	○							○	○
PRF トレース情報の取得レベルを指定します。	pd_prf_level	○	○						○	○
PRF トレース情報ファイルの世代数を指定します。	pd_prf_file_count	○	○						○	○
一つの PRF トレース情報ファイルのサイズを指定します。	pd_prf_file_size	○	○						○	○
PRF トレースを出力するディレクトリを指定します。	pd_prf_output_directory		○						○	○

(34)　トラブルシュート情報に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
トラブルシュート情報を出力するかどうかを指定します。	pd_cancel_dump	○	○						○	○
トランザクションの実行中にクライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）を超えた場合にトラブルシュート情報を出力するかどうかを指定します。	pd_client_waittime_over_abort	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
トランザクションの実行中にクライアントの最大待ち時間（クライアント環境定義のPDCWAITTIMEオペランドの値）を超えた場合に出力される共用メモリダンプを取得するユニットの範囲を指定します。	pd_clt_waittime_over_dump_level	○							○	○
トラブルシュート情報の再出力を抑止する時間を指定します。	pd_dump_suppress_watch_time	○	○						○	○
HiRDBプロセスの異常終了，又はHiRDB（ユニット）が異常終了したときに取得するトラブルシュート情報中に，ネットワーク情報を取得するかどうかを指定します。	pd_debug_info_netstat	○							○	○
HiRDBが出力したトラブルシュート情報ファイル（$PDDIR/spool下のファイル），及び作業用一時ファイル（$PDDIR/tmp下のファイル）の削除間隔を指定します。	pd_spool_cleanup_interval	○	○						○	○
pd_spool_cleanup_intervalオペランドの削除対象となるファイルを指定します。	pd_spool_cleanup_interval_level	○	○						○	○
HiRDBが出力したトラブルシュート情報ファイル（$PDDIR/spool下のファイル）をHiRDB開始時に削除するかどうかを指定します。	pd_spool_cleanup	○	○						○	○
pd_spool_cleanupオペランドの削除対象となるトラブルシュート情報ファイルを指定します。	pd_spool_cleanup_level	○	○						○	○
モジュールトレース格納最大数を指定します。	pd_module_trace_max	○	○	○	○	○	○	○	○	○
モジュールトレース出力時刻取得方法を指定します。	pd_module_trace_timer_level	○	○	○	○	○	○	○	○	○
通信トレース格納最大数を指定します。	pd_pth_trace_max	○	○	○	○	○	○	○	○	○

(35)　バックエンドサーバ接続保持機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
バックエンドサーバ接続保持機能を使用するかどうかを指定します。	pd_bes_connection_hold			○				○	○	○
バックエンドサーバ接続保持時間を指定します。	pd_bes_conn_hold_trn_interval			○				○	○	○

(36)　再編成時期予測機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
再編成時期予測機能を使用するかどうかを指定します。	pd_rorg_predict	○							○	○

(37)　セキュリティに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時から監査証跡を取得するかどうかを指定します。	pd_audit	○	○						×	×
監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。	pd_aud_file_name	○	○						○	○
1 監査証跡ファイルの最大容量をメガバイト単位で指定します。	pd_aud_max_generation_size	○	○						○	○
監査証跡ファイル用の HiRDB ファイルシステム領域内に作成する監査証跡ファイルの最大数（世代数）を指定します。	pd_aud_max_generation_num	○	○						○	○
スワップ先にできる監査証跡ファイルがない場合の HiRDB の処理方式を指定します。	pd_aud_no_standby_file_opr	○							○	○
監査証跡を非同期に出力する場合に使用するバッファ長をバイト単位で指定します。	pd_aud_async_buff_size	○	○						○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
監査証跡を非同期に出力する場合に使用するバッファの面数を指定します。	pd_aud_async_buff_count	○	○						○	○
監査証跡を非同期に出力する場合に使用するバッファがすべて使用されているとき，未使用のバッファが確保できるまでバッファを監視する処理のリトライ間隔を指定します。	pd_aud_async_buff_retry_intvl	○	○						○	○
セキュリティ監査機能を使用する場合に監査証跡に出力するSQL 文のサイズをバイト単位で指定します。	pd_aud_sql_source_size	○	○						○	○
セキュリティ監査機能を使用する場合に監査証跡に出力するSQL データのサイズをバイト単位で指定します。	pd_aud_sql_data_size	○	○						○	○
スワップ先にできない監査証跡ファイル数が警告値以上になったとき，警告メッセージを出力します。このオペランドには警告値を，pd_aud_max_generation_num オペランドに指定した監査証跡ファイルの最大数に対する比率で指定します。	pd_aud_file_wrn_pnt	○							○	○
監査証跡表の自動データロード機能を使用するかどうかを指定します。	pd_aud_auto_loading	○							×	×
監査証跡表の自動データロード機能で動作させるデータベース作成ユティリティ（pdload）の環境情報を定義します。	pdaudload	○							○	○
セキュリティ監査機能の情報を置くバッファの大きさを指定します。	pd_audit_def_buffer_size			○	○	○			○	○
HiRDB サーバを構成するネットワークで使用している全ホスト名を指定します。	pd_security_host_group	○							○	○
IP アドレスによる接続制限機能で，FES 又は SDS のあるユニットのマシンからの自マシンに対	pd_security_local_constraint	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
するローカル接続についても接続制限の対象とするかどうかを指定します。										

(38)　系切り替え機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
系切り替え機能を使用するかどうかの指定をします。	pd_ha	○							×	×
系切り替え機能を使用する場合に IP アドレスを引き継ぐかどうかを指定します。	pd_ha_ipaddr_inherit	○	○						×	×
特定のユニットだけ系切り替えしない場合に指定します。	pd_ha_unit		○						×	×
HiRDB の再開始に失敗したときに実行するコマンドを絶対パス名で指定します。	pd_ha_restart_failure		○						○	○
系切り替え機能の運用をモニタモードで行うか，又はサーバモードで行うかを指定します。	pd_ha_acttype		○						×	×
ユーザサーバホットスタンバイを使用するかどうかを指定します。	pd_ha_server_process_standby		○						×	×
高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用するかどうかを指定します。	pd_ha_agent		○						×	×
影響分散スタンバイレス型系切り替え機能を使用する場合，ユニット内で同時に実行系として稼働できるゲスト BES 数の最大値を指定します。	pd_ha_max_act_guest_servers		○						×	×
影響分散スタンバイレス型系切り替え機能を使用する場合，ユニット内の最大起動ユーザサーバプロセス数を指定します。	pd_ha_max_server_process		○						×	○※2
影響分散スタンバイレス型系切り替え機能を使用する場合，ゲ	pd_ha_process_count		○						○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
スト BES 受け入れ後のユニット内のホスト BES とゲスト BES の常駐プロセス数の合計値を指定します。										
影響分散スタンバイレス型系切り替え機能を使用する場合，ユニットを開始するときに，実行系サーバのリソースが活性化されるまでの最大待ち時間を指定します。	pd_ha_resource_act_wait_time	○	○						○	○
トランザクションキューイング機能を使用するかどうかを指定します。また，系の切り替え中に最大同時接続数（pd_max_users オペランドの値）を超えた場合の処理を指定します。	pd_ha_transaction※6	○							○	○
トランザクションキューイング機能を使用する場合のトランザクションのキューイング待ち時間を指定します。	pd_ha_trn_queuing_wait_time	○							○	○
系切り替えが発生してから待機系ユニットが再開始するまでの間で，トランザクションの開始要求のリトライ処理時間の上限を指定します。	pd_ha_trn_restart_retry_time	○							○	○
トランザクションキューイング機能を使用する場合に，フロントエンドサーバを配置したユニットの起動直後からトランザクションキューイング機能を有効にするかどうかを指定します。	pd_ha_trn_queuing_initial	○							○	○
系切り替え時の HiRDB（HiRDB/パラレルサーバの場合はユニット）の内部停止処理がサーバ障害監視時間を超えた場合に，HiRDB の内部停止処理を待たないで系を切り替えるかどうかを指定します。	pd_ha_switch_timeout	○	○						○	○
サーバモードの系切り替え機能を使用する場合に，切り替え対象のすべてのサーバプロセスが終了するまで，系切り替え処理	pd_ha_prc_cleanup_check	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
を待ち合わせるかどうかを指定します。										
システムマネジャユニットの系が切り替わる時に，ほかのユニットの開始を待ち合わせるかどうか指定します。	pd_ha_mgr_rerun	○							○	○
系切り替え失敗時の自動再起動機能を使用するかどうかを指定します。	pd_ha_switch_error	○	○						○	○
系切り替えの実行時間監視機能における，系切り替え処理の監視時間を指定します。	pd_ha_observe_timeout	○	○						○	○
HA グループを定義します。	pdhagroup	○							×	×

(39)　HiRDB Datareplicator に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB Datareplicator 連携機能を HiRDB 開始時から使用するかどうかを指定します。	pd_rpl_init_start	○							×	×
HiRDB Datareplicator 連携機能を使用する場合に，トランザクションを反映する単位を指定します。	pd_rpl_reflect_mode	○							×	○
HiRDB Datareplicator 連携機能を使用しているときに，HiRDB Datareplicator でのシステムログの抽出が完了していないため，すべてのシステムログファイルがスワップ先にできない状態でスワップ要求が発生した場合の処理方法を指定します。	pd_log_rpl_no_standby_file_opr	○							○	○
HiRDB Datareplicator 連携機能を使用する場合，抽出側 HiRDB Datareplicator 運用ディレクトリ名を指定します。	pd_rpl_hdepath		○						×	×
HiRDB Datareplicator 連携機能を使用する場合の，抽出側 HiRDB で実行する機能を指定します。	pd_rpl_func_control	○							○	○

(40)　リアルタイム SAN レプリケーションに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
リアルタイム SAN レプリケーションを使用するかどうかを指定します。	pd_rise_use	○							×	×
リアルタイム SAN レプリケーションの処理方式を選択します。	pd_rise_pairvolume_combination	○							×	×
リアルタイム SAN レプリケーションの処理方式にハイブリッド方式を選択している場合，メインサイトとリモートサイト間で同期をとり，データの整合性を維持するかどうか指定します。	pd_rise_disaster_mode	○							○	○
リアルタイム SAN レプリケーションの処理方式に全同期方式又はハイブリッド方式を選択している場合に，障害が発生して，リモートサイトにあるボリュームへデータの同期コピーに失敗した場合の動作を指定します。	pd_rise_fence_level	○							○	○
HiRDB が使用するペア論理ボリュームを定義した RAID Manager のインスタンス番号を設定します。	HORCMINST	○							○	○

(41)　インナレプリカ機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
インナレプリカグループの最大数を指定します。	pd_inner_replica_control	○							×	○
インナレプリカ機能を使用するとき，UAP とコマンドを同時実行させるかどうかを指定します。	pd_inner_replica_lock_shift	○							○	○
レプリカ RD エリアのオープン属性を SCHEDULE にするかどうかを指定します。	pd_lv_mirror_use	○							○	○
DEFER 又は SCHEDULE 属性のレプリカ RD エリアを，HiRDB	pd_inner_replica_hold_status	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
の正常開始時にコマンド閉塞にするかどうかを指定します。										
追い付き反映処理をするときに使用するプロセス数を指定します。	pd_max_reflect_process_count	○							×	×
追い付き反映処理が完了したシステムログファイルの状態を変更するかどうかを指定します。	pd_log_org_reflected_logpoint	○							○※4	○※4
全システムログファイルがオンライン再編成上書き禁止状態の場合に，システムログファイルのスワップが発生したときのHiRDB の処理を指定します。	pd_log_org_no_standby_file_opr	○							○	○

(42)　OLTP に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
OLTP 環境下の UAP で，ホールダブルカーソル機能を使用するかどうかを指定します。	pd_oltp_holdcr	○							○	○

(43)　日付・時刻に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
タイムゾーンを指定します。	TZ	○							○	○

(44)　ユーザ認証に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
OS ログインユーザの簡易認証機能を有効にするかどうかを指定します。	pd_os_authenticate	○							○	○
HiRDB サーバが許可する認証方式を指定します。	pd_connect_auth_type	○							○	○

(45)　メッセージの出力抑止機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB が syslogfile に出力するメッセージに対して，出力，出力抑止の制御又は優先度の変更をします。	pdmlgput	○							○	○

(46)　HiRDB システム定義ファイルの共用化に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB システム定義ファイルの共用化をするとき，共用ディレクトリ名を指定します。	PDCONFPATH		○						◎	◎

(47)　クライアントグループに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
クライアントグループの接続枠保証機能を使用する場合に，接続保証ユーザ数を指定します。	pdcltgrp	○							○	○

(48)　プラグインに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
使用するプラグインの名称を指定します。	pdplugin	○							×	×
プラグインが使用する共用メモリの大きさを指定します。	pdplgprm				○	○	○	○	○	○
プラグインインデクスの遅延一括作成をするときに，インデクス情報ファイルを作成するディレクトリを指定します。	pd_plugin_ixmk_dir				○			○	×	○

(49)　バージョンアップに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB をバージョンアップするときに実行する pdvrup コマンドを自動起動するかどうかを指定します。	pd_auto_vrup	○							×	×

(50)　通信処理に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
スケジューラプロセスのポート番号を指定します。	pd_service_port	○	○						◎	◎
ユニット間のシステムサーバと通信する場合のアドレス解決方法を指定します。	pd_name_fixed_port_lookup	○							○	○
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，スケジューラプロセスのポート番号を指定します。	pd_scd_port	○							○	○
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，トランザクションサーバプロセスのポート番号を指定します。	pd_trn_port	○							○	○
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，メッセージログサーバプロセスのポート番号を指定します。	pd_mlg_port	○							○	○
pd_name_fixed_port_lookup オペランドに Y を指定した場合に，ユニット監視プロセスのポート番号を指定します。	pd_alv_port	○							○	○
HiRDB サーバから HiRDB クライアントへの通信時に使用するネットワークを指定します。	pd_change_clt_ipaddr	○	○						○	○
HiRDB が通信処理で使用するポート番号の範囲を指定します。	pd_registered_port	○	○						×	◎
pd_registered_port オペランドで範囲指定したポート番号と/etc/services に登録された	pd_registered_port_check	○	○						×	◎

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
ポート番号との重複チェックをするかどうかを指定します。										
プロセス間通信の送信リトライ処理の連続繰り返し回数を指定します。	pd_ipc_send_retrycount	○	○						○	○
プロセス間通信の送信リトライ処理のスリープ時間を指定します。	pd_ipc_send_retrysleeptime	○	○						○	○
サーバ間の送信処理が終了するまでの送信処理のリトライ回数を指定します。	pd_ipc_send_count	○	○						○	○
サーバ間の受信処理が終了するまでの受信処理のリトライ回数を指定します。	pd_ipc_recv_count	○	○						○	○
HiRDB のサーバ間通信（ユニット間の通信）で，コネクション確立をノンブロックモードで行うかどうかを指定します。	pd_ipc_conn_nblock	○							○	○
HiRDB のサーバ間通信（ユニット間の通信）におけるノンブロックモードでのコネクション確立処理のリトライ時間と，コネクションの確立監視時間を指定します。	pd_ipc_conn_nblock_time	○							○	○
HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ間隔を指定します。	pd_ipc_conn_interval	○							○	○
HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ回数を指定します。	pd_ipc_conn_count	○							○	○
HiRDB クライアントとの通信で，コネクション確立をノンブロックモードで行うかどうかを指定します。	pd_ipc_clt_conn_nblock	○							○	○
HiRDB クライアントとの通信におけるノンブロックモードでのコネクションの確立監視時間を指定します。	pd_ipc_clt_conn_nblock_time	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
通信情報ファイルディレクトリのパスを指定します。	pd_ipc_file_dir	○							○	○
スケジューラプロセスがクライアントからの暗号化通信（SSL）の接続要求を受け付けるポート番号を指定します。	pd_crypto_service_port	○	○						○	○
通信暗号化機能に使用する秘密鍵のファイルパスを指定します。	pd_crypto_private_key	○	○						○	○
通信暗号化機能に使用するサーバ証明書のファイルパスを指定します。	pd_crypto_certificate_file	○	○						○	○
サーバのユニット間通信で，定期的に KeepAlive パケットを送信する機能を使用するかを指定します。	pd_ipc_so_keepalive	○							○	○
KeepAlive パケットを送信する間隔を指定します。	pd_ipc_tcp_keepidle	○							○	○
pd_ipc_tcp_keepidle オペランドの指定値を経過しても送信した KeepAlive パケットに対する応答がなかった場合の，KeepAlive パケット送信間隔を指定します。	pd_ipc_tcp_keepintvl	○							○	○
pd_ipc_tcp_keepidle オペランドの指定値を経過しても送信した KeepAlive パケットに対する応答がなかった場合の，KeepAlive パケット再送回数の上限値を指定します。	pd_ipc_tcp_keepcnt	○							○	○

(51)　Java に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
Java 仮想マシンの起動オプションを指定します。	pd_java_option	○							○	○
外部 Java ルーチンが使用するスタック領域長を指定します。	pd_java_routine_stack_size	○							○	○
Java ストアドプロシジャ又は Java ストアドファンクションで	pd_java_archive_directory	○	○						○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
使用する JAR ファイルを格納するディレクトリ名を指定します。										
Java 仮想マシンが使用するクラスパスを指定します。	pd_java_classpath	○	○						○	○
Java Runtime Environment のルートディレクトリを絶対パス名で指定します。	pd_java_runtimepath	○	○						○	○
Java 仮想マシンのライブラリが格納されているディレクトリを，Java Runtime Environment のルートディレクトリへの相対パス名で指定します。	pd_java_libpath	○	○						○	○
Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合，そのファイル名を指定します。	pd_java_stdout_file	○	○	○	○	○	○	○	○	○
Java 仮想マシンの起動プロセスを，特定の契機で終了させるかどうかを指定します。	pd_java_castoff			○	○	○	○	○	○	○

(52)　外部 C ストアドルーチンに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
外部 C ストアドルーチンで使用する C ライブラリファイルを格納するディレクトリを指定します。	pd_c_library_directory	○	○						○	○

(53)　SQL 実行時間警告出力機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL 実行時間警告情報を出力する条件を指定します。	pd_cwaittime_wrn_pnt※6	○			○	○			○	○
SQL 実行時間警告情報ファイルの出力先ディレクトリを指定します。	pd_cwaittime_report_dir	○	○						○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
SQL 実行時間警告情報ファイルの最大容量を指定します。	pd_cwaittime_report_size	○	○						○	○

(54)　文字コードに関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
1 文字を表現する最大バイト数を指定します。	pd_substr_length	○							○	○

(55)　ワークファイル出力先ディレクトリの変更に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB が出力するワークファイルの出力先ディレクトリ名を指定します。	pd_tmp_directory		○						○	○

(56)　HiRDB ファイルシステム領域に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
複製ディスク機能に使用する HiRDB ファイルシステム領域のサフィックス名称を指定します。	pd_duplicated_fs_suffix	○							○	○
pdfmkfs -k UTL -m コマンドで作成した HiRDB ファイルシステム領域数	pd_max_utl_ios_file_no	○	○						○	○

(57)　リードレプリカ機能に関するオペランド

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
HiRDB システムで参照 DB を使用するかどうかを指定します。	pd_read_replica_use	○							○	○

オペランドの内容	オペランド名	定義名							FORCE	PLAN
		SYS	UNT	SVR	SDS	FES	DS	BES		
参照 DB 側のシステム共通定義に，更新ログを反映する方式を指定します。	pd_read_replica_mode	○							×	×
pd_read_replica_use オペランドに Y を指定した場合に，pdrrpref コマンドで使用する一時ファイル作成ディレクトリを指定します。	pd_read_replica_workdir	○							○	○
更新 DB の更新トランザクションを参照 DB に反映時に指定されたトランザクション数を反映するごとに参照 DB で COMMIT を発行する回数を指定します。	**pd_read_replica_max_trn_count**	○							○	○

9.2　オペランドの説明

9.2.1　システム構成に関するオペランド

◆ **pd_system_id = HiRDB 識別子**

～＜識別子＞((4 文字))

HiRDB サーバの識別子を指定します。このオペランドは省略できません。必ず指定してください。

次に示す場合は，システム間で HiRDB 識別子が一意になるようにしてください。

- 複数の HiRDB/シングルサーバを使用している場合
- ユティリティ専用ユニットを使用している場合

《注意事項》

一度設定した HiRDB 識別子を変更するには，データベース初期設定ユティリティで再度システムを構築し直す必要があります。したがって，後で変更が発生するような名称は避けてください。

◆ **pd_name_port = HiRDB のポート番号**

～＜符号なし整数＞((5001～65535))《20000》

HiRDB のポート番号を指定します。

《注意事項》

注意事項については，マニュアル「HiRDB システム導入・設計ガイド」の「HiRDB で指定するポート番号」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pdunit オペランドの-p オプションと関連があります。

◆ **pd_master_file_name = "マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称"**

～＜パス名＞((167 文字以内))

マスタディレクトリ用 RD エリアを構成する先頭の HiRDB ファイル名称を，絶対パス名で指定します。このオペランドは省略できません。必ず指定してください。

《注意事項》

- ユティリティ専用ユニットの場合は，このオペランドを指定しないでください。
- 一度設定したマスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を変更するには，データベース初期設定ユティリティで再度システムを構築し直す必要があります。したがって，後で変更が発生するような名称は避けてください。

◆ **pd_system_expand_unit ＝ユニット識別子〔，ユニット識別子〕...**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

拡張ユニットのユニット識別子を指定します。拡張ユニットは，バッチジョブ分散実行システムと HiRDB を組み合わせたシステムで使用します。バッチジョブ分散実行システム及び拡張ユニットについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

このオペランドに指定できるユニット識別子は 63 個までです。

また，このオペランドを指定する場合は，基本ユニット及び拡張ユニットの合計数が 64 個以下になるようにしてください。

《注意事項》

- このオペランドの値は HiRDB の再開始時に変更できません。
- システムマネジャ又はディクショナリサーバがあるユニットは，拡張ユニットにできません。したがって，システムマネジャ又はディクショナリサーバがあるユニットをこのオペランドに指定しないでください。
- 基本ユニットには，一つ以上のフロントエンドサーバ及びバックエンドサーバを設定する必要があります。それを考慮してこのオペランドに指定するユニットを決定してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_max_file_no
- pd_max_rdarea_no
- pd_max_users
- pd_start_level
- pd_start_skip_unit
- pdbuffer
- pdstart
- pdunit -u オプション

《各見積もり式への影響》

pd_system_expand_unit オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「ステータスファイルの容量の見積もり」の「S の求め方」

◆ **pdunit -x ホスト名**
-u ユニット識別子
〔-d "HiRDB 運用ディレクトリ名"〕
〔-c ホスト名〕
〔-p HiRDB のポート番号〕
〔-s スケジューラプロセスのポート番号〕
〔-t トランザクションサーバプロセスのポート番号〕
〔-m メッセージログサーバプロセスのポート番号〕
〔-a ユニット監視プロセスのポート番号〕

《備考》

-x オプション，及び-c オプションは，計画停止後の再開始時でも指定値の変更ができます。それ以外のオプションの場合は，正常終了後に指定値を変更してください。

●HiRDB/シングルサーバの場合

HiRDB/シングルサーバ又はユティリティ専用ユニットのユニット構成を定義します。

《指定値の目安》

- HiRDB/シングルサーバだけでシステムが構成されている場合（ユティリティ専用ユニットを使用しない場合）はこのオペランドを省略できます。

  ただし，IP アドレスを引き継がない系切り替え機能を使用する場合はこのオペランドを省略できません。
- 系切り替え機能を使用する場合は，現用系と予備系のシステム定義の内容を同じにします。
- 相互系切り替え構成の場合，マルチ HiRDB の構成（1 サーバマシンに二つの HiRDB/シングルサーバが共存する構成）となりますが，二つの HiRDB/シングルサーバでシステム定義の内容を同じにはできません。
- 指定するすべての pdunit オペランドの-x 及び-c オプションに指定するホスト名には，それぞれ異なる名称を指定してください。
- ユティリティ専用ユニットを使用する場合，HiRDB/シングルサーバのシステム共通定義には，HiRDB/シングルサーバ及びユティリティ専用ユニットのユニット情報を指定します。HiRDB/シングルサーバの分の pdunit オペランドと，ユティリティ専用ユニットの分の pdunit オペランドの二つを指定します。
- ユティリティ専用ユニットのシステム共通定義には，HiRDB/シングルサーバ及びユティリティ専用ユニットのユニット情報を指定します。ユティリティ専用ユニットを使用する HiRDB/シングルサーバの分の pdunit オペランドと，ユティリティ専用ユニットの分の pdunit オペランドを指定します。

《注意事項》

ユティリティ専用ユニットを使用する場合，HiRDB はこのオペランドで指定したホスト名を使用して，HiRDB/シングルサーバとユティリティ専用ユニットとの通信をします。このため，ホスト名の指定を誤ると，通信障害やタイムオーバとなることがあります。

-x ホスト名

～<ホスト名>((1～32 文字))

HiRDB/シングルサーバを定義するサーバマシンのホスト名を指定します。

IP アドレス又は FQDN 形式で指定することもできます。また，ループバックアドレスを指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

指定例については，マニュアル「HiRDB システム導入・設計ガイド」の「DNS サーバで IP アドレスを管理する場合の設定」を参照してください。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

《ほかのオペランドとの関連》

pdunit オペランドの-x オプション，-c オプション，及び pd_hostname オペランドに指定するホスト名の関係を次に示します。

<table>
<tr><th colspan="2" rowspan="2">システム構成の条件</th><th colspan="2">pdunit オペランド</th><th rowspan="2">pd_hostname オペランドの指定</th></tr>
<tr><th>-x オプションの指定</th><th>-c オプションの指定</th></tr>
<tr><td colspan="2" rowspan="2">系切り替え機能を使用しない場合</td><td>• ホスト名※1 を指定
• pdunit オペランドを省略</td><td>指定不要</td><td>• pd_hostname オペランドを省略
• 標準ホスト名※1 を指定
• -x オプションと同じホスト名※1 を指定</td></tr>
<tr><td>ループバックアドレスを指定
※2，※4</td><td>指定不要</td><td>• pd_hostname オペランドを省略
• ループバックアドレスを指定</td></tr>
<tr><td rowspan="3">系切り替え機能を使用する場合</td><td rowspan="2">モニタモードで IP アドレスを引き継ぐ場合</td><td>再配置可能な IP アドレス，再配置可能な IP アドレスのホスト名，又は FQDN を指定</td><td>指定不要</td><td>• pd_hostname オペランドを省略
• 現用系の標準ホスト名※1 を指定
• -x オプションと同じホスト名※1 を指定</td></tr>
<tr><td>ループバックアドレスを指定
※2，※3，※4，※5</td><td>指定不要</td><td>• pd_hostname オペランドを省略
• ループバックアドレスを指定</td></tr>
<tr><td>サーバモードで IP アドレスを引き継ぐ場合</td><td>再配置可能な IP アドレス，再配置可能な IP アドレスのホスト名，又は FQDN を指定</td><td>指定不要</td><td>現用系の標準ホスト名※1 を指定</td></tr>
<tr><td colspan="2">IP アドレスを引き継がない場合</td><td>現用系のホスト名※1 を指定</td><td>予備系のホスト名※1 を指定</td><td>現用系の標準ホスト名※1 を指定</td></tr>
</table>

注

- ここで指定したホスト名は hosts ファイル又は DNS などに登録し，名前解決しておく必要があります。
- ホスト名を FQDN 形式で指定する場合は，ホスト名を FQDN 形式で定義しておく必要があります。
- HiRDB では，localhost を通常のホスト名として扱うため，ループバックアドレスとして使用する場合は，名前解決しておく必要があります。
- pdunit オペランドを省略した場合，-x オプションに標準ホスト名が指定されたと仮定します。この場合，「はじめに」の「■このマニュアルで使用している構文要素記号」で説明している規則に従って標準ホスト名が付けられている必要があります。

  また，pdunit オペランドを省略した場合は標準ホスト名が仮定されるため，標準ホスト名を hosts ファイル又は DNS に登録して名前解決してください。

注※1

指定形式は，ホスト名，IP アドレス，又は FQDN 形式のどれかになります。

注※2

ループバックアドレスを指定する場合，hosts ファイル又は DNS などへのホスト名の登録が不要になります。

注※3

この場合，クラスタソフトウェアの IP アドレス引き継ぎの設定が不要になります。

注※4

この場合，ほかのマシンの HiRDB クライアントから HiRDB サーバに接続できません。HiRDB サーバに接続するには，マルチコネクションアドレス機能又は高速接続機能を使用してください。

注※5

マルチコネクションアドレス機能を使用する場合は，pdstart オペランドの-m オプション（HiRDB クライアントが接続するホスト名を指定するオプション）に，系の切り替え時に引き継ぐ再配置可能な IP アドレスのホスト名を指定してください。

-u ユニット識別子

～＜識別子＞((4 文字))

ユニット識別子を指定します。ユニット制御情報定義の pd_unit_id オペランドにも，ここで指定したユニット識別子を指定します。

-d "HiRDB 運用ディレクトリ名"

- Linux の場合：**～＜ 118 文字以内のパス名＞《/opt/HiRDB_S》**
- Linux 以外の場合：**～＜ 128 文字以内のパス名＞《/opt/HiRDB_S》**

HiRDB 運用ディレクトリ名を指定します。

pdunit オペランドを省略すると，-d オプションに pdsetup コマンドで指定した HiRDB 運用ディレクトリが指定されたと仮定します。

-c ホスト名

～＜ホスト名＞((1～32 文字))

IP アドレスを引き継がない系切り替えをする場合にこのオプションを指定します。

このオプションには予備系のホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

なお，ユティリティ専用ユニットの場合は系切り替えができないため，このオプションを指定しないでください。指定すると定義エラーとなり，HiRDB を開始できません。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

-p HiRDB のポート番号

～＜符号なし整数＞((5001～65535))

ユティリティ専用ユニットがある構成で，すべての HiRDB/シングルサーバ又はユティリティ専用ユニットでそれぞれ異なるポート番号を使用するときに指定します。

次の場合には，自システムの HiRDB のポート番号（pd_name_port オペランドで指定した値）が仮定されます。

- ユティリティ専用ユニットがない構成
- ユティリティ専用ユニットがある構成で，このオプションを省略した場合

《指定値の目安》

- HiRDB/シングルサーバで指定する場合は，HiRDB/シングルサーバが利用するユティリティ専用ユニットの pdunit オペランドに，ユティリティ専用ユニットの HiRDB のポート番号（pd_name_port オペランドで指定した値）を指定してください。
- ユティリティ専用ユニットで指定する場合は，ユティリティ専用ユニットを利用する HiRDB/シングルサーバの pdunit オペランドに，HiRDB/シングルサーバの HiRDB のポート番号（pd_name_port オペランドで指定した値）を指定してください。
- マルチ HiRDB で一つのユティリティ専用ユニットを共用する場合には，必ず指定してください。

《注意事項》

- OS が自動的に割り当てるポート番号の範囲とは異なる番号を指定してください。OS が自動的に割り当てるポート番号の範囲は OS によって異なります。
- OS が自動的に割り当てるポート番号の範囲内の番号を指定した場合，その番号はほかのプログラムに使用されているおそれがあります。使用されていた場合，HiRDB を開始できません。
- HiRDB のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_name_port オペランドと関連があります。

-s スケジューラプロセスのポート番号

～＜符号なし整数＞((5001～65535))

スケジューラプロセスのポート番号を指定します。

《注意事項》

- 通常，このオプションを指定する必要はありません。
- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_service_port

- pd_scd_port

-t トランザクションサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

トランザクションサーバプロセスのポート番号を指定します。

《注意事項》

- 通常，このオプションを指定する必要はありません。
- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- トランザクションサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_trn_port オペランドと関連があります。

-m メッセージログサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

メッセージログサーバプロセスのポート番号を指定します。

《注意事項》

- 通常，このオプションを指定する必要はありません。
- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- メッセージログサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_mlg_port オペランドと関連があります。

●HiRDB/パラレルサーバの場合

HiRDB/パラレルサーバのユニット構成を定義します。

《指定値の目安》

- このオペランドには，HiRDB/パラレルサーバを構成するすべてのユニットの情報を定義します。例えば，ユニットが三つある場合は，pdunit オペランドを三つ指定します。
- 系切り替え機能を使用する場合はこのオペランドを省略できません。
- 系切り替え機能を使用する場合は，現用系と予備系のシステム定義の内容を同じにします。
- 相互系切り替え構成の場合，マルチ HiRDB の構成（1 ホストに異なる二つのユニットが共存する構成）となりますが，異なる二つのユニットでシステム定義の内容を同じにはできません。
- 指定するすべての pdunit オペランドの-x 及び-c オプションに指定するホスト名には，それぞれ異なる名称を指定してください。

- HiRDB/パラレルサーバを 1 ユニットだけで構成する場合は，このオペランドを省略できます。省略すると，pdsetup コマンドで指定した HiRDB 運用ディレクトリが仮定されます。

-x ホスト名

～＜ホスト名＞((1～32 文字))

HiRDB/パラレルサーバの各サーバを定義するサーバマシンのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

指定例については，マニュアル「HiRDB システム導入・設計ガイド」の「DNS サーバで IP アドレスを管理する場合の設定」を参照してください。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

《ほかのオペランドとの関連》

pdunit オペランドの-x オプション，-c オプション，及び pd_hostname オペランドに指定するホスト名の関係を次に示します。

システム構成の条件		pdunit オペランド		pd_hostname オペランドの指定
		-x オプションの指定	-c オプションの指定	
系切り替え機能を使用しない場合		ホスト名※1 を指定	指定不要	• pd_hostname オペランドを省略 • 標準ホスト名※1 を指定
系切り替え機能を使用する場合	IP アドレスを引き継ぐ場合	再配置可能な IP アドレス，再配置可能な IP アドレスのホスト名，又は FQDN を指定	指定不要	現用系の標準ホスト名※1 を指定
	IP アドレスを引き継がない場合	現用系のホスト名※1 を指定	予備系のホスト名※1 を指定※2	現用系の標準ホスト名※1 を指定

注

- ここで指定したホスト名は hosts ファイル又は DNS などに登録し，名前解決しておく必要があります。
- ホスト名を FQDN 形式で指定する場合は，ホスト名を FQDN 形式で定義しておく必要があります。

注※1

指定形式は，ホスト名，IP アドレス，又は FQDN 形式のどれかになります。

注※2

pdunit オペランドの-x オプション及び-c オプションに指定するホスト名は重複できません。正しい例と間違った例を次に示します。

（正しい例）

```
pdunit -x hostA ... -c hostAA
pdunit -x hostB ... -c hostBB
```

（間違った例）

```
pdunit -x hostA ... -c hostB
pdunit -x hostB ... -c hostA
```

IP アドレスを引き継がない相互系切り替え構成の HiRDB システム定義の例については，「HiRDB/パラレルサーバの場合：スタンバイ型系切り替え機能使用時」を参照してください。

-u ユニット識別子

～＜識別子＞((4 文字))

ユニット識別子を指定します。各ユニット制御情報定義の pd_unit_id オペランドにも，ここで指定したユニット識別子を指定します。

-d "HiRDB 運用ディレクトリ名"

- Linux の場合：**～＜ 118 文字以内のパス名＞《/opt/HiRDB_P》**
- Linux 以外の場合：**～＜ 128 文字以内のパス名＞《/opt/HiRDB_P》**

このユニットの HiRDB 運用ディレクトリ名を指定します。

《注意事項》

- 1：1 スタンバイレス型系切り替え機能を使用する場合

  正規バックエンドサーバユニットと代替バックエンドサーバユニットで同じ HiRDB 運用ディレクトリ名を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合

  同じ HA グループに属する全ユニットで，同じ HiRDB 運用ディレクトリ名を指定してください。

-c ホスト名

～＜ホスト名＞((1～32 文字))

IP アドレスを引き継がない系切り替えをする場合にこのオプションを指定します。

このオプションには予備系のホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。ただし，同じサーバマシンに対する指定形式は統一してください。例えば，同じサーバマシンのホスト名を，ホスト名形式で指定したり，IP アドレス形式で指定したりすることはできません。

なお，次に示すユニットにはこのオプションを指定できません。指定した場合は HiRDB を開始できません（KFPS01896-E メッセージが出力されます）。

- 1：1 スタンバイレス型系切り替え機能を適用するユニット（IP アドレスの引き継ぎが発生しない）
- 影響分散スタンバイレス型系切り替え機能を適用するユニット（IP アドレスの引き継ぎが発生しない）
- ユティリティ専用ユニット（ユティリティ専用ユニットは系切り替えができない）

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

-p HiRDB のポート番号

～<符号なし整数>((5001～65535))

各ユニットでの HiRDB のポート番号を指定します。このオプションを省略した場合は，pd_name_port オペランドで指定した値が仮定されます。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- HiRDB のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，pd_name_port オペランドと関連があります。

-s スケジューラプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのスケジューラプロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，次に示すオペランドと関連があります。
  - ・pd_service_port
  - ・pd_scd_port

-t トランザクションサーバプロセスのポート番号

～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのトランザクションサーバプロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- トランザクションサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pd_trn_port オペランドと関連があります。

-m メッセージログサーバプロセスのポート番号

～＜符号なし整数＞((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのメッセージログサーバプロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットに同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- メッセージログサーバプロセスが起動しないユニットではポート番号を指定しても使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。メッセージログサーバプロセスが起動しないユニットは，二つ以上のユニットで構成される HiRDB/パラレルサーバで pd_mlg_msg_log_unit の指定値が manager の場合，システムマネジャがないユニットです。
- メッセージログサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pd_mlg_port オペランドと関連があります。

-a ユニット監視プロセスのポート番号

～＜符号なし整数＞((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，各ユニットでのユニット監視プロセスのポート番号を指定します。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニットごとに異なるポート番号を指定してください。
- 1：1 スタンバイレス型系切り替え機能を使用する場合は，正規 BES ユニットと代替 BES ユニットとで同じポート番号を指定してください。
- 影響分散スタンバイレス型系切り替え機能を使用する場合，同じ HA グループに属するユニットには，すべて同じポート番号を指定してください。
- ユニット監視プロセスが起動しないユニットではポート番号を指定しても使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。ユニット監視プロセスは，二つ以上のユニットで構成される HiRDB/パラレルサーバでシステムマネジャがあるユニットを除いて，すべてのユニットで起動しません。
- ユニット監視プロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pd_alv_port オペランドと関連があります。

◆ **pdstart -t サーバ種別〔-s サーバ名〕**
{-x ホスト名｜-u ユニット識別子}
〔-m ホスト名〔，ホスト名〕...
〔-n ホスト名〔，ホスト名〕...〕〕
〔-c サーバ名｜-g HA グループ識別子〕
〔-k stls〕

このオペランドは省略できません。必ず指定してください。

《注意事項》

- 一度設定したサーバ名（-s オプションの指定）を変更する場合は，データベース初期設定ユティリティで再度システムを構築し直す必要があります。したがって，後で変更が発生するようなサーバ名は避けてください。
- -t，-u，-s，又は-c オプションの指定を変更する場合は，次に示す操作をする必要があります。したがって，後で変更が発生するような指定値は避けてください。

1. システムログファイルの初期化（オプションを変更したサーバのシステムログファイルが初期化の対象になります）

2. シンクポイントダンプファイルの初期化（オプションを変更したサーバのシンクポイントダンプファイルが初期化の対象になります）
3. ユニット用ステータスファイル及びサーバ用ステータスファイルの初期化（オプションを変更したユニット及びサーバのステータスファイルが初期化の対象になります）

《備考》

-x オプション，-m オプション，及び-n オプションは，計画停止後の再開始時でも指定値の変更ができます。それ以外のオプションの場合は，正常終了後に指定値を変更してください。

●HiRDB/シングルサーバの場合

HiRDB/シングルサーバのサーバ構成を定義します。指定する項目は次のとおりです。

- サーバ種別
- サーバ名
- ホスト名又はユニット識別子

《指定値の目安》

- HiRDB/シングルサーバを複数個接続して使用する場合，サーバ名は各 HiRDB/シングルサーバ間で一意になるように指定してください。
- ユティリティ専用ユニットを使用する場合も，ほかのサーバ名と重複しないようにしてください。
- ユティリティ専用ユニットの場合，ユティリティ専用ユニットに対する pdstart オペランドの指定は必要ありません。ただし，ユティリティ専用ユニットを使用するすべての HiRDB/シングルサーバに対する pdstart オペランドを指定する必要があります。

-t サーバ種別

サーバの種別を指定します。

SDS：シングルサーバ

-s サーバ名

～＜識別子＞((1～8 文字))

シングルサーバのサーバ名を指定します。

-x ホスト名

～＜ホスト名＞((1～32 文字))

pdunit オペランドの-x オプションに指定したホスト名を指定します。

pdunit オペランドを省略した場合，pdunit オペランドの-x オプションに標準ホスト名が指定されたと仮定されるため，このオプションに標準ホスト名を指定してください。

-u ユニット識別子

～＜識別子＞((4 文字))

サーバを実行するユニットの識別子を指定します。pdunit オペランドの-u オプションで指定したユニット識別子を指定します。

系切り替え機能を使用する場合は，「-x ホスト名」ではなく，「-u ユニット識別子」を指定することをお勧めします。

-m ホスト名〔，ホスト名〕...

～＜ホスト名＞((1～32 文字))

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続する HiRDB/シングルサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

マルチコネクションアドレス機能については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。また，pd_hostname オペランドにループバックアドレスを指定している場合に，ほかのホストの HiRDB クライアントから HiRDB サーバにアクセスするときは，このオプションを指定する必要があります。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

《注意事項》

- このオプションは 4 つまで指定できます。
- IP アドレスのネットワーク部に同じアドレス値を持つホスト名を複数指定しないでください。複数指定した場合は 1 つ目のホスト名が使用されます。特に，サブネット化を使用している場合はご注意ください。
- HiRDB/シングルサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- -m オプションと-n オプションの指定方法を次に示します。

HiRDB/シングルサーバの構成		pdstart オペランドの指定方法	
系切り替え構成	系切り替え時に HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレス（クライアントの接続先となる IP アドレス）を引き継ぐ構成	-m オプションの指定	-n オプションの指定
適用	適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスに対応するホスト名を指定してください。	オプションの指定を省略してください。
	非適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの現用系のホスト名を指定してください。	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの予備系のホスト名を指定してください。
非適用	−	HiRDB クライアントと HiRDB サーバ間で使用するネットワークのホスト名を指定してください。	オプションの指定を省略してください。

（凡例）

−：IP アドレスを引き継ぐ構成の適用対象外

- 系切り替え構成を適用する際は，HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない場合でも，HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスは引き継ぐ構成としてください。HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない構成とした場合，予備系のホストに系が切り替わると UAP の接続処理時間が遅くなります。
- IP アドレスを引き継ぐ系切り替え機能を使用する場合に，pdunit オペランドの-x オプションにループバックアドレスを指定しているときは，再配置可能な IP アドレスのホスト名を-mオプションに指定してください。

-n ホスト名〔，ホスト名〕...

～<ホスト名>((1～32 文字))

このオプションは-mオプションと一緒に指定してください。

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続する HiRDB/シングルサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。IP アドレスを引き継がない系切り替え機能を使用する場合に，予備系のホスト名を指定してください。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

《注意事項》

- このオプションは 4 つまで指定できます。
- IP アドレスのネットワーク部に同じアドレス値を持つホスト名を複数指定しないでください。複数指定した場合は 1 つ目のホスト名が使用されます。特に，サブネット化を使用している場合はご注意ください。
- HiRDB/シングルサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- -m オプションと-n オプションの指定方法は，-m オプションの項を参照してください。
- このオプションを指定する場合は，pd_hostname オペランドに標準ホスト名を指定してください。

-c サーバ名

～<識別子>((1～8 文字))

このオプションは HiRDB/パラレルサーバ限定のオプションのため，省略してください。

-g HA グループ識別子

～<識別子>((1～8 文字))

このオプションは HiRDB/パラレルサーバ限定のオプションのため，省略してください。

-k stls

～<識別子>((4 文字))

このオプションは HiRDB/パラレルサーバ限定のオプションのため，省略してください。

●HiRDB/パラレルサーバの場合

HiRDB/パラレルサーバのサーバ構成を定義します。指定する項目は次のとおりです。

- サーバ種別
- サーバ名
- ホスト名又はユニット識別子

《指定値の目安》

サーバ構成の目安を次に示します。

- 各サーバを別サーバマシンに定義すると，SQL 処理の並列度を向上できます。
- TCP/IP プロトコルで接続されているサーバマシンにシステムマネジャ及びフロントエンドサーバを定義すると，SQL 処理の並列度を向上できます。
- バックエンドサーバの CPU 負荷によっては，1 サーバマシンに複数のバックエンドサーバを割り当てた方がよい場合もあります。

《注意事項》

システムマネジャ及びディクショナリサーバは複数個指定できません。

-t サーバ種別

サーバの種別を指定します。

MGR：システムマネジャ

FES：フロントエンドサーバ

BES：バックエンドサーバ

DIC：ディクショナリサーバ

-s サーバ名

～＜識別子＞((1～8 文字))

サーバ名を指定します。サーバ種別が MGR（システムマネジャ）の場合は指定する必要はありません。

-x ホスト名

～＜ホスト名＞((1～32 文字))

pdunit オペランドの-x オプションに指定したホスト名を指定します。

-u ユニット識別子

～＜識別子＞((4 文字))

サーバを実行するユニットの識別子を指定します。pdunit オペランドの-u オプションで指定したユニット識別子を指定します。

系切り替え機能を使用する場合は，「-x ホスト名」ではなく，「-u ユニット識別子」を指定することをお勧めします。

-m ホスト名〔，ホスト名〕...

～＜ホスト名＞((1～32 文字))

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続するフロントエンドサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

マルチコネクションアドレス機能については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

《注意事項》

- このオプションは 4 つまで指定できます。
- IP アドレスのネットワーク部に同じアドレス値を持つホスト名を複数指定しないでください。複数指定した場合は 1 つ目のホスト名が使用されます。特に，サブネット化を使用している場合はご注意ください。
- フロントエンドサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- -m オプションと-n オプションの指定方法を次に示します。

フロントエンドサーバを配置するユニットの構成		pdstart オペランドの指定方法	
系切り替え構成	系切り替え時に HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレス（クライアントの接続先となる IP アドレス）を引き継ぐ構成	-m オプションの指定	-n オプションの指定
適用	適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスに対応するホスト名を指定してください。	オプションの指定を省略してください。
	非適用	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの現用系のホスト名を指定してください。	HiRDB クライアントと HiRDB サーバ間で使用するネットワークの予備系のホスト名を指定してください。
非適用	−	HiRDB クライアントと HiRDB サーバ間で使用するネットワークのホスト名を指定してください。	オプションの指定を省略してください。

（凡例）

−：IP アドレスを引き継ぐ構成の適用対象外

- 系切り替え構成を適用する際は，HiRDB サーバ間で使用するネットワークの IP アドレスを引き継がない場合でも，HiRDB クライアントと HiRDB サーバ間で使用するネットワークの IP アドレスは引き継ぐ構成としてください。HiRDB クライアントと HiRDB サーバ間で使用するネット

ワークの IP アドレスを引き継がない構成とした場合，予備系のホストに系が切り替わると UAP の接続処理時間が遅くなります。

- IP アドレスを引き継ぐ系切り替え機能を使用する場合に，pdunit オペランドの-x オプションにループバックアドレスを指定しているときは，再配置可能な IP アドレスのホスト名を-mオプションに指定してください。

-n ホスト名〔，ホスト名〕...

～＜ホスト名＞((1～32 文字))

このオプションは-mオプションと一緒に指定してください。

マルチコネクションアドレス機能を使用する場合に，HiRDB クライアントが接続するフロントエンドサーバのホスト名を指定します。IP アドレス又は FQDN 形式で指定することもできます。

HiRDB クライアントと HiRDB サーバ間で使用しているネットワークと，HiRDB サーバのサーバマシン間で使用しているネットワークが異なる場合にこのオプションを指定します。IP アドレスを引き継がない系切り替え機能を使用する場合に，予備系のホスト名を指定してください。

ホスト名形式で指定する場合の注意事項

- ホスト名には別名を指定できません。

《注意事項》

- このオプションは 4 つまで指定できます。
- IP アドレスのネットワーク部に同じアドレス値を持つホスト名を複数指定しないでください。複数指定した場合は 1 つ目のホスト名が使用されます。特に，サブネット化を使用している場合はご注意ください。
- フロントエンドサーバがないサーバマシンのホスト名を指定した場合，その指定は無視されます。
- -m オプションと-n オプションの指定方法は，-m オプションの項を参照してください。
- このオプションを指定する場合は，pd_hostname オペランドに標準ホスト名を指定してください。

-c サーバ名

～＜識別子＞((1～8 文字))

このオプションは 1：1 スタンバイレス型系切り替え機能に関するオプションです。-s オプションで指定したバックエンドサーバの代替 BES 名を指定します。

-c オプションを指定するには，次に示す条件をすべて満たす必要があります。満たさない場合は HiRDB を開始できません。なお，ここでは pdstart オペランドで-u オプションを指定したとして説明しています。

- -t オプションに BES が指定されている必要があります。
- 正規 BES ユニット及び代替 BES ユニット内にバックエンドサーバ以外のサーバがあってはなりません。
- 正規 BES ユニット（-u オプションに指定したユニット）内の全バックエンドサーバに対して代替 BES を指定してください。また，その代替 BES は同じユニット（代替 BES ユニット）に定義する必要があります。

- 正規 BES ユニット内で-c オプションに指定するサーバ名を重複できません。
- 正規 BES ユニット内のバックエンドサーバ数と代替 BES ユニット内のバックエンドサーバ数は同じ必要があります。
- 代替 BES の pdstart オペランドに-c オプションを指定する場合，-c オプションには正規 BES ユニット内のバックエンドサーバを指定する必要があります。

-g HA グループ識別子

～＜識別子＞((1～8 文字))

影響分散スタンバイレス型系切り替え機能を使用する場合に **HA グループ**識別子を指定します。HA グループ識別子は-s オプションで指定したサーバの移動先となるユニットの集合で，pdhagroup オペランドに指定しておく必要があります。HA グループについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

なお，このオプションを指定する場合は次に示す条件をすべて満たす必要があります。すべて満たさない場合は HiRDB を開始できません。

●-s オプションに指定するサーバの条件

- -t オプションに BES を指定している（バックエンドサーバを指定する必要があります）
- -c オプションは指定していない（このサーバに対する代替 BES は指定できません）

●-u オプションに指定するユニットの条件

- 正規ユニットは-g オプションで指定した HA グループに属している
- 正規ユニットを構成するサーバの種類はバックエンドサーバだけである
- 正規ユニットを構成するサーバの移動先に HA グループを指定している

●-g オプションに指定する HA グループの条件

- HA グループ内の全ユニットが同じネットワークセグメントに属している
- 影響分散スタンバイレス型系切り替え機能を適用している（pd_ha_agent = activeunits を指定している）
- ユニット内に一つ以上のホスト BES がある
- ユニット内のサーバの種類はバックエンドサーバだけである
- ユニット内のサーバはすべて HA グループに属している

-k stls

～＜識別子＞((4 文字))

回復不要 FES を使用する場合に指定します。回復不要 FES を使用する場合は HiRDB Non Recover FES が必要になります。

なお，次に示すどれかの条件を満たす場合は HiRDB を開始できません。

- このオプションの指定に誤りがある
- -t オプションに指定したサーバがフロントエンドサーバでない
- フロントエンドサーバ以外のサーバがあるユニットに対してこのオプションを指定している

- HiRDB Non Recover FES をセットアップしていない

《注意事項》

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは回復不要 FES を使用できません。-k stls オプションを指定した場合，KFPS04689-W メッセージが出力されます。

《ほかのオペランドとの関連》

- このオプションを指定する場合は，pd_log_max_data_size オペランドの指定値を見積もり直してください。
- このオプションを指定したフロントエンドサーバのユニットを pd_start_skip_unit オペランドに指定しても，指定は無視されます。
- 回復不要 FES ユニットでは，系切り替え機能を適用できません。系切り替え機能を適用するシステムの場合，回復不要 FES ユニットの pd_ha_unit オペランドに必ず nouse を指定してください。

◆ pd_unit_id = ユニット識別子　～<識別子>((4 文字))

このオペランドは省略できません。必ず指定してください。

ユニット識別子を指定します。システム共通定義の pdunit -u オペランドに指定したユニット識別子を指定してください。

複数の HiRDB/シングルサーバを使用している場合は，システム間でユニット識別子が一意になるようにしてください。

ユティリティ専用ユニットの場合も，ほかのユニット識別子と重複しないようにしてください。同じユニット識別子を指定した場合，ユティリティ専用ユニットを開始するときに，エラーとなります。

HiRDB/パラレルサーバの場合は，システム内でユニット識別子が一意になるように指定してください。

《注意事項》

ユニット識別子を変更する場合は，次に示すファイルを再初期化する必要があります。

- ユニット用ステータスファイル
- システムログファイル

◆ pd_hostname = ホスト名　～<ホスト名>((1～32 文字))

ユニットを定義した現用系のサーバマシンの標準ホスト名を指定します。ホスト名は，ホスト名，IP アドレス，又は FQDN のどれかの指定形式で指定してください。

また，システム共通定義の pdunit オペランドの-x オプションに指定したホスト名を指定できます。ホスト名は，ホスト名，IP アドレス，又は FQDN のどれかの指定形式で指定してください。ただし，pdunit オペランドの-x オプションの指定形式と合わせてください。例えば，pdunit オペランドの-x オプションで IP アドレスを指定した場合は，このオペランドにも IP アドレスを指定してください。

《指定値の目安》

- このオペランドは，システム共通定義の pdunit 及び pdstart オペランドと関係が深いため，pdunit 及び pdstart オペランドの説明を読んでから，このオペランドを指定してください。

- ここに指定するホスト名は，/etc/brc.ipc に定義してある名称で，hostname コマンド又は gethostname 関数を実行すると分かります。
- リアルタイム SAN レプリケーションを使用する場合はこのオペランドを省略できません。メインサイトの pd_hostname オペランドにはメインサイトの標準ホスト名を，リモートサイトの pd_hostname オペランドにはリモートサイトの標準ホスト名を指定してください。

《注意事項》

マルチコネクションアドレス機能を使用するために，pdstart オペランドの-n オプションを指定した場合は，このオペランドを省略できません。この場合，このオペランドには標準ホスト名を指定してください。

9.2.2 同時実行最大数に関するオペランド

◆ **pd_max_users = 最大同時接続数**

～<符号なし整数>

- HiRDB/シングルサーバの場合：**((1～3000))《10》**
- HiRDB/パラレルサーバの場合：**((1～2000))《10》**

HiRDB サーバに対する最大同時接続数を指定します。HiRDB/パラレルサーバの場合は 1 フロントエンドサーバに対する最大同時接続数を指定します。HiRDB サーバへの接続要求数がこのオペランドの値を超えると，HiRDB サーバはそれ以上の接続要求を受け付けません（接続要求はエラーになります）。なお，ここでいう接続とは，CONNECT 文による HiRDB サーバへの接続を意味しています。

《指定値の目安》

- OpenTP1 を使用する場合は，HiRDB サーバに接続する（XA 接続を含む）OpenTP1 のサーバプロセス数が接続数になります。
- DBPARTNER/Server を使用する場合は，DBPARTNER/Server の接続クライアント数が接続数になります。
- HiRDB SQL Executer を使用する場合は，HiRDB SQL Executer の接続クライアント数が接続数になります。
- 複数接続機能を使用する場合は，各 UAP の同時接続数の総和が接続数になります。
- HiRDB のコマンド及びユティリティの中には，内部的に HiRDB に接続するものがあるため，これらのコマンド及びユティリティの実行中は接続できるユーザ数が一時的に減少します。このことを考慮してオペランドの値を決めてください。コマンド及びユティリティの接続数については，マニュアル「HiRDB コマンドリファレンス」の「コマンドの同時接続数」を参照してください。

《注意事項》

- pd_max_users オペランドの値を大きくした場合は，pd_max_server_process オペランドの値も大きくする必要があります。また，これに伴い HiRDB が使用する共用メモリ，及びポート数

が増加します。pd_max_users オペランドの値を大きくした場合（ユーザ数が増えた場合）の注意事項については，マニュアル「HiRDB システム運用ガイド」を参照してください。なお，pd_max_server_process オペランドを省略すると，このオペランドの値を HiRDB が自動計算します。

- マルチフロントエンドサーバの場合は，最大で pd_max_users オペランドの値×フロントエンドサーバ数の接続要求を処理できますが，pd_max_users オペランドの値×フロントエンドサーバ数の値が pd_max_users オペランドの指定可能上限値（2000）を超えないようにしてください。
- pd_max_reflect_process_count オペランドの値と追い付き反映処理プロセスの総数によっては，pd_max_users オペランドの値まで接続できないことがあります。詳細については，pd_max_reflect_process_count オペランドの説明を参照してください。
- HiRDB サーバへの接続要求数がこのオペランドの値を超えると，接続要求がエラーとなり，プロセスが残ったままになることがあります。pdls -d prc コマンドでプロセスが残っていないか確認し，残っていた場合は pdcancel コマンドでそのプロセスを強制終了させてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_hash_entry
- pd_lck_pool_size
- pd_max_bes_process
- pd_max_dic_process
- pd_max_reflect_process_count
- pd_max_server_process
- pd_process_count
- pd_process_terminator_max
- pdcltgrp

《各見積もり式への影響》

pd_max_users オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「64 ビットモードに移行する際の考慮点」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバで起動するプロセス」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

- 「Linux のオペレーティングシステムパラメタの見積もり」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「SQL 文が使用する作業表用ファイルの容量」
- 「最大ファイル数の見積もり（pdfmkfs -l コマンド）」
- 「システム構成に関する最大値と最小値」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 2」，「計算式 3」，「計算式 4」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 2」，「計算式 3」，「計算式 4」
- 「フロントエンドサーバが使用する共用メモリの計算式」
- 「メッセージキュー及びセマフォ所要量の見積もり」
- 「ユニットが使用する通信ポート数の見積もり」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

◆ pd_max_server_process = 最大同時起動サーバプロセス数 ～＜符号なし整数＞

- HiRDB/シングルサーバの場合：((50～10000))
- HiRDB/パラレルサーバの場合：((50～15000))

ユニット内で同時に起動するサーバのプロセス数の最大値を指定します。サーバのプロセス数には，システムサーバ，各サーバ，ユティリティなどのプロセス数を含めます。システムサーバとは HiRDB が内部的に使用するサーバのことです。

《指定値の目安》

- 通常はこのオペランドを省略してください。このオペランドを省略すると，次に示す計算式の値が仮定されます。なお，変数の説明中に出てくるオペランドの値を変更した場合は HiRDB が自動的に再計算します。また，最大同時起動サーバプロセス数が pd_max_server_process の最大値（HiRDB/シングルサーバの場合は 10000，HiRDB/パラレルサーバの場合は 15000）を超える場合は，各種定義値を小さくし，定義省略時の計算式が最大値内に収まるようにしてください。

  省略値＝ａ＋ｂ×（ｃ＋31）＋70＋ｉ＋ｊ＋ｋ＋m＋ｎ

- このオペランドを指定する場合は次に示す計算式を参考にして求めてください。HiRDB/パラレルサーバの場合はユニットごとに求めてください。その結果のうち一番大きな値を目安としてください。

  推奨値の目安＝ａ＋ｂ×（ｃ＋ｄ×ｅ＋ｆ＋3）＋ｄ×ｇ＋ｈ＋ｉ＋ｊ＋ｋ＋m＋ｎ＋Ａ

変数	変数の説明
a	**HiRDB/シングルサーバの場合**

変数	変数の説明
	pd_max_users の値 **HiRDB/パラレルサーバの場合** ユニット内のサーバごとに次の計算式を実行して求めた値を合計してください。 バックエンドサーバの場合：pd_max_bes_process の値 ディクショナリサーバの場合：pd_max_dic_process の値 フロントエンドサーバの場合：pd_max_users の値 • pd_max_bes_process 又は pd_max_dic_process オペランドを省略している場合は pd_max_users の値で計算してください。 • ユニット内に複数のバックエンドサーバがある場合は，バックエンドサーバごとに計算してください。 • マルチフロントエンドサーバの場合，フロントエンドサーバごとに 1 を加算してください。 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_server_process オペランドの値を代入します。
b	**HiRDB/シングルサーバの場合** 1 **HiRDB/パラレルサーバの場合** ユニット内サーバ数（システム共通定義の pdstart オペランドで該当ユニットに割り当てたサーバ数） • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_servers オペランドの値（省略している場合はデフォルト値）を加算します。
c	全面回復処理の並列実行プロセス数（pd_max_recover_process オペランドの値）
d	同時実行するユティリティの最大数（このオペランドを省略した場合 2 が仮定されます）
e	ユティリティがサーバごとに起動するプロセス数（10 としてください）
f	HiRDB がサーバ制御のために起動するプロセス数（8 としてください）
g	ユティリティがユニット単位に起動するプロセス数（10 としてください）
h	HiRDB がユニット制御のために起動するプロセス数（50 としてください）
i	**HiRDB/シングルサーバの場合** 1 **HiRDB/パラレルサーバの場合** ユニット内バックエンドサーバ数 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，pd_ha_max_act_guest_servers オペランドの値を加算します。
j	**HiRDB/シングルサーバの場合** pd_max_ard_process オペランドの値 **HiRDB/パラレルサーバの場合** ユニット内バックエンドサーバ数×pd_max_ard_process オペランドの値＋ユニット内ディクショナリサーバ数×pd_max_ard_process オペランドの値 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，次に示す値を加算してください。 ゲスト BES の pd_max_ard_process オペランドの最大値×pd_ha_max_act_guest_servers オペランドの値
k	pd_process_terminator オペランドの指定値によって値が変わります。 • resident の場合：1

変数	変数の説明
	• fixed の場合（省略値）：pd_process_terminator_max オペランドの値 • nonresident の場合：0
m	更新可能なオンライン再編成をする場合は次に示す値を代入します。更新可能なオンライン再編成を行わない場合は 0 を代入します。 **HiRDB/シングルサーバの場合** x **HiRDB/パラレルサーバの場合** x ×（y ＋ z） x ：pd_max_reflect_process_count オペランドの値 y ：ユニット内のサーバ数 z ：pd_ha_max_act_guest_servers オペランドの値 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合に z を加算します。影響分散スタンバイレス型系切り替え機能を適用しないユニットの場合は 0 になります。
n	**HiRDB/シングルサーバの場合** pd_dfw_awt_process オペランドの値 **HiRDB/パラレルサーバの場合** バックエンドサーバ数×pd_dfw_awt_process オペランドの値＋ディクショナリサーバ数×pd_dfw_awt_process オペランドの値 • 影響分散スタンバイレス型系切り替え機能を適用するユニットの場合は，次に示す値を加算してください。 ゲスト BES の pd_dfw_awt_process オペランドの最大値×pd_ha_max_act_guest_servers オペランドの値
A	トランザクション回復プロセスの最大起動数 ↓pd_trn_rcvmsg_store_buflen オペランドの値÷72↓

《ほかのオペランドとの関連》

- このオペランドの指定値で，pd_process_count オペランドの上限値が制限されます。
- 各ユニットでサーバの構成が大きく異なるため，ユニットごとに値を調整する場合は，ユニット制御情報定義の pd_max_server_process オペランドを指定してください。

《注意事項》

- この指定値にはユニット内のサーバ，ユティリティなどのプロセス数が含まれます。したがって，この値が小さ過ぎると，次に示す現象が発生することがあります。
  ・ユニット又はサーバの開始処理がエラーになります。
  ・トランザクションの回復処理ができなくなります。
  ・HiRDB の計画停止ができなくなります。
  ・常駐プロセス数より最大同時接続数が大きい場合は，接続できなくなります。
  ・コマンド実行の延長で動作する，内部プロセス起動時にコマンドエラーになります。
- 実際に起動できるプロセス数は，システムのリソースなどに影響されるため，リソースの調整やサーバの配置変更などが必要となる場合があります。

《各見積もり式への影響》

pd_max_server_process オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「AIX のオペレーティングシステムパラメタの見積もり」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「Linux のオペレーティングシステムパラメタの見積もり」

◆ pd_max_access_tables = 同時アクセス可能実表数

～＜符号なし整数＞((4～32000))《64》

1 トランザクションで同時にアクセスできる表数と順序数生成子数の合計の最大値を指定します。同時にアクセスできる表数と順序数生成子数とは，1 トランザクション中の SQL 文に記述した表と順序数生成子の延べ数のことです。

《指定値の目安》

- 異なる SQL 文で同じ表，又は順序数生成子を指定している場合，それぞれに表，又は順序数生成子の指定回数を加えてください。
- 一つの SQL 文で同じ表，又は順序数生成子を複数回指定している場合，それぞれに表，又は順序数生成子の指定回数を加えてください。
- ディクショナリ表の参照権限を設定する場合は，同時アクセス可能実表数にアクセスするディクショナリ表の 5 倍の数を加算してください。ディクショナリ表の参照権限については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- HiRDB Datareplicator の反映表定義で load 文を指定した回数以上の値を指定してください。load 文の指定回数よりも小さい値を指定すると，KFPA11931-E エラーとなります。
- HiRDB では，このオペランドの指定値を基に 1 トランザクションで排他できる表のエントリを，HiRDB サーバプロセスごとに管理しています。不足した場合は SQL エラーとなります。

《各見積もり式への影響》

pd_max_access_tables オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「64 ビットモードに移行する際の考慮点」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」,「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」,「計算式 2」,「計算式 3」
- 「フロントエンドサーバが使用する共用メモリの計算式」

9.2.3 プロセスに関するオペランド

◆ **pd_max_bes_process = 1 バックエンドサーバ当たりの最大起動プロセス数**
～＜符号なし整数＞((1～12000))《pd_max_users の値》

1 バックエンドサーバ当たりの**最大起動プロセス数**を指定します。マルチフロントエンドサーバの場合，1 バックエンドサーバに pd_max_users の値を超える処理が集中することがあります。このオペランドには，1 バックエンドサーバのプロセスを pd_max_users の値を超えて起動する場合の最大起動プロセス数を指定します。

《前提条件》

このオペランドはマルチフロントエンドサーバのときに指定します。

《指定値の目安》

- 1 バックエンドサーバに集中する可能性がある処理の最大値を次に示します。

  pd_max_users オペランドの値×フロントエンドサーバ数

  ここで求めた値を最大値として，バックエンドサーバへの処理の集中度を考慮してこのオペランドの値を指定してください。必要以上に大きな値を指定すると，メモリを圧迫する原因となります。
- pd_max_users の値よりも小さな値を指定した場合は pd_max_users の値を仮定します。このとき，警告メッセージ（KFPS01888-W）を出力します。
- 起動できるバックエンドサーバプロセス数以上の処理要求が発生した場合，フロントエンドサーバからバックエンドサーバへの接続処理に時間が掛かります。
- 更新可能なオンライン再編成をする場合，次に示す計算式を満たすようにオペランドの値を決めてください。満たさない場合，HiRDB を開始できません。

  pd_max_bes_process ＋ pd_max_reflect_process_count の値≦2048
- ユニットに複数のバックエンドサーバを配置する場合は，各バックエンドサーバの pd_max_bes_process オペランドの指定値の合計が 12000 以下になるように設計してください。

《指定値のチューニング方法》

最大起動プロセス数のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

バックエンドサーバプロセス数が増えた場合，HiRDB が使用するポート数が増加します。ポート数が不足すると，処理が中断したり，ほかのプログラムの通信処理に影響を与えたりします。OS が自動的に割り当てるポート数は OS によって異なるため，各サーバマシンのネットワークに関する設定を確認してください。HiRDB が使用するポート数の目安はマニュアル「HiRDB システム導入・設計ガイド」の「HiRDB が使用するポート数」を参照してください。

OS が自動的に割り当てるポート数で足りない場合は，pd_registered_port オペランドで HiRDB が使用するポート番号を範囲指定してください。詳細についてはマニュアル「HiRDB システム導入・設計ガイド」の「HiRDB 予約ポート機能」を参照してください。

◆ **pd_max_dic_process = 1 ディクショナリサーバ当たりの最大起動プロセス数**

～＜符号なし整数＞((1～2048))《pd_max_users の値》

1 ディクショナリサーバ当たりの**最大起動プロセス数**を指定します。マルチフロントエンドサーバの場合，ディクショナリサーバに pd_max_users の値を超える処理が集中することがあります。また，マルチフロントエンドサーバでない場合も，RD エリアやグローバルバッファに関連する運用コマンド（pdbufls，pddbls，pdopen，pdclose，pdhold，pdrels）を pd_max_users の値以上同時に実行すると，ディクショナリサーバに pd_max_users の値を超える処理が集中することがあります。

このオペランドには，1 ディクショナリサーバのプロセスを pd_max_users の値を超えて起動する場合の最大起動プロセス数を指定します。

《指定値の目安》

- 1 ディクショナリサーバに集中する可能性がある処理の最大値を次に示します。

  pd_max_users オペランドの値×フロントエンドサーバ数

  ここで求めた値を最大値として，ディクショナリサーバへの処理の集中度を考慮してこのオペランドの値を指定してください。必要以上に大きな値を指定すると，メモリを圧迫する原因となります。

- RD エリアやグローバルバッファに関する運用コマンドの実行で，ディクショナリサーバに処理が集中する場合は，同時に実行する運用コマンドの数が最大値になります。
- pd_max_users の値よりも小さな値を指定した場合は pd_max_users の値を仮定します。このとき，警告メッセージ（KFPS01888-W）を出力します。
- 起動できるディクショナリサーバプロセス数以上の処理要求が発生した場合，処理に時間が掛かります。
- 更新可能なオンライン再編成をする場合，次に示す計算式を満たすようにオペランドの値を決めてください。満たさない場合，HiRDB を開始できません。

  pd_max_dic_process ＋ pd_max_reflect_process_count の値≦2048

《指定値のチューニング方法》

最大起動プロセス数のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_process_count = 常駐プロセス数〔，サーバ開始時の常駐プロセス数〕**

～＜符号なし整数＞

- シングルサーバの場合：**((0〜3000))《最大常駐プロセス数》**
- バックエンドサーバの場合：**((0〜12000))《最大常駐プロセス数》**
- ディクショナリサーバの場合：**((0〜2048))《↑最大常駐プロセス数÷2↑》**
- フロントエンドサーバの場合：**((0〜2000))《↑最大常駐プロセス数÷2↑》**

常駐プロセス数：

各サーバの常駐プロセス数を指定します。常駐プロセスとは，サーバ開始時からあらかじめ起動しておくプロセスのことです。

《利点》

各サーバで並行処理できるトランザクションが使用するプロセスを，あらかじめシステム開始時に起動して常駐しておくことで，新たにトランザクションが入力されてもプロセス起動時間を削減できます。ただし，HiRDB の開始に時間が掛かります。

《指定値の目安》

- 各サーバプロセスのプロセス固有領域とプロセサの実メモリ量から値を求めてください。サーバプロセスのプロセス固有領域については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- マルチフロントエンドサーバ構成で，更に pd_max_bes_process 又は pd_max_dic_process オペランドを指定している場合は，次に示す条件を満たすようにこのオペランドを指定してください。

  pd_process_count の値≦pd_max_bes_process 又は pd_max_dic_process の値
- このオペランドには，次の表に示す各サーバの**最大常駐プロセス数**の範囲内で値を指定します。

サーバの種類		最大起動プロセス数	最大常駐プロセス数
シングルサーバ		pd_max_users ＋ pd_max_reflect_process_count の値	
フロントエンドサーバ	マルチフロントエンドサーバの場合	pd_max_users ＋ pd_max_reflect_process_count の値＋ 1	pd_max_users ＋ pd_max_reflect_process_count の値
	マルチフロントエンドサーバでない場合	pd_max_users ＋ pd_max_reflect_process_count の値	
ディクショナリサーバ		pd_max_dic_process※＋ pd_max_reflect_process_count の値	
バックエンドサーバ		pd_max_bes_process※＋ pd_max_reflect_process_count の値	

注※

pd_max_dic_process 又は pd_max_bes_process オペランドを省略している場合は，pd_max_users の値が仮定されます。

《指定値のチューニング方法》

常駐プロセス数のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

- 常駐プロセス数はメモリや CPU と密接に関連しているため，必要以上に大きい値を指定すると，HiRDB を開始できなくなったり，サーバマシンの処理性能が低下したりすることがあります。
- 常駐プロセス数を超えたプロセスが必要になると，最大起動プロセス数までプロセスを動的に起動します。ただし，pd_max_server_process オペランドの指定値によっては，最大起動プロセス数分のプロセスを起動できないことがあります。
- 最大常駐プロセス数を超える値を指定した場合，エラーにはしないで，最大常駐プロセス数に補正します。

《オペランドの省略値》

このオペランドを省略（又は 0 を指定）し，更に各サーバ定義でもオペランドを省略（又は 0 を指定）すると，次に示す値が仮定されます。

- シングルサーバ又はバックエンドサーバの場合：最大起動プロセス数
- フロントエンドサーバ又はディクショナリサーバの場合：↑最大起動プロセス数÷2↑

サーバ開始時の常駐プロセス数：

HiRDB の開始処理時に常駐化する常駐プロセスの数を指定します。

通常，常駐プロセスの起動処理は HiRDB の開始処理中に行われます。このため，常駐プロセス数が多くなると，それに比例して HiRDB の開始処理時間が長くなります。目安として，100MIPS 程度のサーバマシンでプロセスを一つ起動するのに約 1 秒掛かります。

サーバ開始時の常駐プロセス数の指定有無による処理の違いを次に示します。

●サーバ開始時の常駐プロセス数を指定しない場合（pd_process_count = 500 と指定した場合）

すべての常駐プロセスの起動処理を HiRDB の開始処理中に実行します。常駐プロセス（この場合は 500 個）がすべて起動されないと，HiRDB を開始できません。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 500 秒掛かります。

●サーバ開始時の常駐プロセス数を指定する場合（pd_process_count = 500,50 と指定した場合）

一部の常駐プロセスの起動処理を HiRDB の開始処理中に実行し，残りの常駐プロセスの起動処理を HiRDB の開始処理終了後に実行します。一部の常駐プロセス（この場合は 50 個）が起動されれば，HiRDB を開始できます。100MIPS 程度のサーバマシンで，HiRDB 開始処理時の常駐プロセスの起動に約 50 秒掛かります。残りの常駐プロセス（この場合は 450 個）は，HiRDB の開始処理終了後に起動されます。

《利点》

HiRDB の開始処理時間を短縮できます。系切り替え機能を使用している場合など，HiRDB の開始処理時間をなるべく短くしたいときに適用します。

《指定値の目安》

HiRDB の開始処理終了直後に必要なプロセス数を指定します。

《注意事項》

サーバ開始時の常駐プロセス数を指定する場合は，クライアント環境定義の PDCWAITTIME オペランドの値を見直してください。

HiRDB の開始直後に，サーバ開始時の常駐プロセス数の値を超える UAP が実行されると，残りの常駐プロセスを起動した後にトランザクション処理が実行されます。したがって，クライアント環境定義の PDCWAITTIME オペランドの値が小さいと，一部の UAP がタイムアウトで処理できない場合があります。

PDCWAITTIME オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ **pd_server_cleanup_interval = 非常駐サーバプロセス停止処理の実行間隔**

～＜符号なし整数＞((0～1440))《0》（単位：分）

HiRDB の非常駐サーバプロセスの停止処理をする間隔を分単位で指定します。この機能は，稼働中のサーバプロセス数が常駐プロセス数（pd_process_count オペランドの値）より多くなった場合に適用されます。この機能によって停止するサーバプロセス数は HiRDB が自動的に算出して決定します。

《利点》

業務（稼働中のサーバプロセス数）がピークのときに再利用できる非常駐サーバプロセスが増えるため，メモリなどのプロセス資源の利用効率が向上します。

《指定値の目安》

- 例えば，一日のうち 1 時間だけ業務のピーク時間があり，その時間内でピークとなる間隔が 2 分ぐらいの場合，このオペランドに 2 を指定します。
- ピーク時に同時実行されるサーバプロセス数が常駐プロセス数以下の場合，この機能を使用しても効果は期待できないため，このオペランドを省略します。

《指定値のチューニング方法》

システムの稼働に関する統計情報をサーバ単位に 1 週間取得してください。「サービス実行中のサーバプロセス数（# OF PROCESS ON SERVICE）」の値から業務のピークを求めてください。そのピークが現在設定されている常駐プロセス数（pd_process_count オペランドの値）を超えている場合は，ピークごとの間隔を求めてその時間を設定します。

ただし，サーバマシンのメモリ，CPU などの資源に余裕がある場合は，不足している分のプロセス数を常駐プロセス数に加算する方（pd_process_count オペランドの値を大きくする方）がこのオペランドを指定するより性能が向上します。

《注意事項》

このオペランドを省略するか，又は 0 を指定した場合は 10 秒間隔でサービス待ち状態の非常駐サーバプロセスを調査し，サービス待ち状態の非常駐サーバプロセスがあれば，その非常駐サーバプロセスを停止します。

◆ **pd_svr_castoff_size = 1 サーバプロセスが使用するメモリサイズの上限値**

～＜符号なし整数＞((0～2048))《0》（単位：メガバイト）

1 サーバプロセスの使用メモリサイズの上限値を指定します。1 サーバプロセスの使用メモリサイズがここで指定した値を超えた場合，そのサーバプロセスを次に示す契機で終了させます。サーバプロセスが終了した場合，KFPS00350-W メッセージを出力します。これを**サーバプロセスのメモリサイズ監視機能**といいます。サーバプロセスのメモリサイズ監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

サーバの種類	プロセス名称	プロセス終了契機
シングルサーバ	pdsds	UAP の切り離し時 ユティリティの終了時
フロントエンドサーバ	pdfes	UAP の切り離し時
ディクショナリサーバ	pddic	トランザクション決着時 ユティリティの終了時
バックエンドサーバ	pdbes	トランザクション決着時 ユティリティの終了時

《利点》

サーバプロセスのメモリサイズ監視機能を使用すると，次に示すような問題を解決できます。

- **特定の SQL 処理でサーバの常駐プロセスのメモリサイズが大きくなり，システムの空きメモリを圧迫している**
- **ユティリティ実行時，ローカルバッファサイズやソート用ワークバッファサイズに大きな値を指定して，サーバの常駐プロセスのメモリサイズが大きくなり，システムの空きメモリを圧迫している**

HiRDB は不要となったメモリを解放しますが，OS はプログラムがメモリを解放しても，領域自体は該当するプロセス内のメモリ管理機構で保持しています。このため，一度でも大量の領域を使用して大きくなったプロセスサイズは小さくなることはなく，特に常駐プロセスの場合はシステムを圧迫し続けます。この機能を使用すると，常駐プロセスであってもプロセスを終了させるため，メモリを圧迫する問題を回避できます。

《適用基準》

HiRDB のサーバプロセスが使用するメモリサイズが大きくなり，メモリを圧迫する場合に適用します。

《指定値の目安》

- このオペランドの上限値の考え方

  通常は HiRDB の最大処理能力を考慮して指定値を決めてください。最大同時接続数分の SQL が同時に実行されることを想定すると，各ユニットで次の式を満たすようにこのオペランドの指定値を決めてください。

  a × （b ＋ c） ＜ d

  a：ユニット内のサーバプロセス数（最大同時接続数×ユニット内のサーバ数）

  b：HiRDB 開始直後の 1 サーバプロセス分の仮想メモリサイズ

  c：pd_svr_castoff_size オペランドの値

  d：ユニットで確保できるメモリサイズ（他プログラム使用分を除いたメモリサイズ）

- このオペランドの下限値の考え方

  このオペランドの指定値が通常の SQL 処理に必要なメモリサイズより小さいと，常駐プロセスの効果が低下して，プロセスの終了及び開始が頻繁に起こります。また，そのたびに syslogfile

又はメッセージログファイルにメッセージが出力されるため，性能が低下します。これを防ぐために，各サーバで次の式を満たすようにこのオペランドの指定値を決めてください。

a－b＜c

a：SQL 処理終了後，又はユティリティ終了後のサーバプロセス仮想メモリサイズ

b：HiRDB 開始直後のサーバプロセス仮想メモリサイズ

c：pd_svr_castoff_size オペランドの値

仮想メモリサイズは OS のコマンドなどで調査してください。

《オペランドの規則》

このオペランドに 0 を指定すると，サーバプロセスのメモリサイズ監視機能を使用しません。

◆ **pd_max_open_fds = 1 プロセスがアクセスするファイル及びパイプの最大数**

～＜符号なし整数＞((1～8192))《320》

HiRDB のプロセスがアクセスするファイル及びパイプの最大数を指定します。

《指定値の目安》

シングルサーバ定義に指定する場合

- 1 トランザクションで更新したプラグインインデクス格納 RD エリアの最大数が 100 未満の場合は，このオペランドを省略してください。
- 1 トランザクションで更新したプラグインインデクス格納 RD エリアの最大数が 100 以上の場合は，最大数＋ 320 をこのオペランドに指定してください。

バックエンドサーバ定義に指定する場合

- ａ＋ｂ＜ 100 の場合は，このオペランドを省略してください。
- ａ＋ｂ≧100 の場合は，ａ＋ｂ＋ 320 をこのオペランドに指定してください。

ａ：1 トランザクションで更新した 1 バックエンドサーバプロセス当たりのプラグインインデクス格納 RD エリアの最大数

ｂ：（全文検索プラグイン（HiRDB Text Search Plug-in）又は XML データ検索プラグイン（HiRDB XML Extension）の検索用関数実行時に使用する最大ファイル数※）×（一つの SQL 文で実行する HiRDB Text Search Plug-in 又は HiRDB XML Extension 提供関数の最大個数）

注※　各検索用関数で，最大 10 ファイルを使用します。

HiRDB Text Search Plug-in の検索用関数については，マニュアル「HiRDB 全文検索プラグイン HiRDB Text Search Plug-in」を参照してください。

HiRDB XML Extension の検索用関数については，マニュアル「HiRDB XML 拡張機能　HiRDB XML Extension」を参照してください。

《注意事項》

- 指定値が小さいと SQL エラーになることがあります。
- 次に示す操作を行う場合，このオペランドの値は無効になります。

  ・データベース作成ユティリティ，データベース再編成ユティリティ，リバランスユティリティでインデクス情報ファイルを作成する場合

・データベース作成ユティリティで分割入力データファイルを作成する場合

これらの場合，ソフトリミットが次に示す計算式の値に変更されます。

MIN（A，B）

A：一つ前の注意事項で説明した OS の種類ごとの pd_max_open_fds オペランドの最大値

B：OS のオペレーティングシステムパラメタの単一プロセスがオープン又はロックできるファイル数の物理限界値（次に示すパラメタの値です）

・AIX の場合：nofiles_hard の値

・Linux の場合：hard nofile の値

- バックエンドサーバの 1 プロセスが使用するファイル又はパイプの数が多い場合，使用できるファイル又はパイプの数がこのオペランドの指定値よりも少なくなることがあります。マニュアル「HiRDB システム導入・設計ガイド」の「ユニット数又はサーバ数が多いシステムを構築する場合の考慮点」を参照して対策するか，次に示す値を小さくしてプロセスが使用するファイル及びパイプの数を減らしてください。

  ・1 トランザクションで更新したプラグインインデクス格納 RD エリアの最大数

  ・HiRDB Text Search Plug-in の 1 契機で使用する最大ファイル数

  ・一つの SQL 文で実行する HiRDB Text Search Plug-in 提供関数の最大個数

◆ pd_max_ard_process = 非同期 READ プロセス数

～＜符号なし整数＞((0～256))《0》

非同期 READ 機能を使用する場合にこのオペランドを指定します。このオペランドには非同期 READ 処理に必要なプロセス数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（バックエンドサーバ又はディクショナリサーバ）当たりのプロセス数を指定します。非同期 READ 機能については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《前提条件》

pdbuffer オペランドの-m オプションで 1 以上を指定している必要があります。

《利点》

非同期 READ 機能は入出力時間が掛かるキャラクタ型スペシャルファイルを使用している場合に特に効果があります（性能が向上します）。逆に，入出力時間が掛からない通常ファイル又は日立ディスクアレイシステムのディスクなどを使用している場合は，次に示す理由によって効果が得られないことがあります。

- 入出力時間と CPU 時間がオーバラップする割合が少ない
- 通信処理にオーバヘッドが掛かる

《指定値の目安》

- 0 又は 1 を指定してください。ただし，pdbuffer オペランドの-m オプションの値が 2～256 の場合は，-m オプションと同じ値を指定してください。-m オプションの値が 257 以上の場合は，RD エリアやシステムファイルを格納するディスク装置の数（HiRDB/パラレルサーバの場合は 1 サーバ当たりの数）と同じ値を指定するか，又は 256 を指定してください。

- このオペランドの値を大きくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に処理時間を短縮できます。このオペランドの値を小さくすると，非同期 READ 機能が適用される SQL の多重度が高い場合に，非同期 READ プロセスの処理完了待ちが発生する分，処理時間が余計に掛かることがあります。
- このオペランドの値×サーバ数分のプロセスが起動されるため，リソース（共用メモリ及びメッセージキュー）を考慮してこのオペランドの値を決定してください。共用メモリ及びメッセージキューの見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《指定値のチューニング方法》

指定値（非同期 READ プロセス数）のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《オペランドの規則》

このオペランドに 0 を指定すると，非同期 READ 機能を使用しません。

1：1 スタンバイレス型系切り替え機能を使用している場合，正規 BES とそれに対応する代替 BES に同じ値を指定してください。

《ほかのオペランドとの関連》

このオペランドの値を変更した場合は pd_max_server_process オペランドの値を見直してください。

《各見積もり式への影響》

pd_max_ard_process オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバで起動するプロセス」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバで起動するプロセス」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 2」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

◆ pd_dfw_awt_process = デファードライト処理用並列 WRITE プロセス数 ～<符号なし整数>((2～255))

すべてのバッファプールに対して，**デファードライト処理の並列 WRITE 機能**を使用する場合にこのオペランドを指定します。このオペランドには並列処理するプロセス数を指定します。プロセス数を増加させることで書き込み処理時間を短縮できます。デファードライト処理の並列 WRITE 機能については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《指定値の目安》

デファードライト処理の並列 WRITE 機能が有効になる最小値の 2 を指定してください。また，マニュアル「HiRDB システム運用ガイド」の「デファードライト処理のチューニング」を参照してオペランドの値を決定してください。

《注意事項》

デファードライト処理の並列 WRITE 機能を指定すると，プロセス数が増加するため，CPU 利用率が上がります。

《各見積もり式への影響》

pd_dfw_awt_process オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバで起動するプロセス」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバで起動するプロセス」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 5」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 4」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 5」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

9.2.4 作業表に関するオペランド

◆ **pd_work_buff_mode = each ｜ pool ｜ pool2**

HiRDB が作業表を作成するときのバッファの確保方式を指定します。

each：個々の作業表ごとにバッファを確保します。

pool：サーバプロセス単位にバッファプールとして一括して確保します。確保するタイミングは，プロセス起動時に行います。

pool2：サーバプロセス単位にバッファプールとして一括して確保します。確保するタイミングは，サーバプロセス内で 1 つ目の作業表を作成する時です。サーバプロセスで使用する作業表がなくなった時に解放します。

《指定値の目安》

- 通常は pool（省略値）を指定してください。特に大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合には pool を指定してください。
- ユーザサーバの常駐プロセス数よりも作業表を使用する操作（SQL 又はユティリティ）を行うトランザクションの同時実行数の方が少ない場合は，pool2 の指定を推奨します。
- あらかじめ，作業表用バッファに使用できるプロセス固有領域サイズが決まっている場合，pool 又は pool2 を指定してください。pool 又は pool2 を指定すると，HiRDB が効率良く作業表用バッファを各作業表に配分します。

  この場合，pd_work_buff_size に指定した値でプロセス固有領域を占有し，そのプール内で作業表の入出力をバッファリングします。したがって，pd_work_buff_size に指定した値以上にプロセス固有メモリを占有することはありません。

  pool2 は，作業表を使用するトランザクションが動作する際にプロセス固有領域を確保し，そのトランザクションが終了する際にプロセス固有領域を解放します。したがって，作業表を使用している最中のプロセスでだけ pd_work_buff_size 分のメモリを消費します。

《注意事項》

このオペランドに each を指定すると，メモリ使用量が増加します。

《各見積もり式への影響》

pd_work_buff_mode オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「64 ビットモードに移行する際の考慮点」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

◆ pd_work_buff_size = 作業表用バッファ長 ～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：**((128～1000000))**
- 64 ビットモードの場合：**((128～4000000000))**

HiRDB が作成する作業表のバッファの大きさをキロバイト単位で指定します。

項目	pd_work_buff_mode ＝ each の場合	pd_work_buff_mode に pool 又は pool2 を指定した場合
利点	作業表バッファ長が大きいと，データ操作時に発生する作業表用ファイルへの入出力回数を削減できるため，作業表を使用する SQL の実行時間を削減できます。ただし，各サーバのプロセス固有メモリを使用するので，システム全体のメモリ量（実メモリ，仮想メモリ）を考慮して指定してください。 特に pd_work_buff_mode=each の場合，メモリは「pd_work_buff_size の値×必要作業表数」分確保されます。一度確保されたプロセス固有メモリは，その後 HiRDB が解放要求しても仮想メモリ上はそのプロセスに占有されてしまいます（そのプロセスがなくなるまで解放されません）。したがって，必要以上に大きな値を指定すると，ほかのプロセスで仮想メモリ不足が生じることがあります。	

項目	pd_work_buff_mode ＝ each の場合	pd_work_buff_mode に pool 又は pool2 を指定した場合
適用基準	大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合に指定します。このとき，pd_work_buff_mode = pool 又は pool2 を指定してください。	
指定値の目安	• 個々の作業表ごとに確保するバッファの大きさを指定します。 • 作業表用バッファ長に，作業表メモリ容量より大きな値を指定すると，作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。作業表メモリ容量の見積もり式は次のとおりです。 **作業表メモリ容量＝対象となる作業表容量※÷2**	• サーバプロセス単位に一括して確保する作業表用バッファプールの大きさを指定します。 • 大量データを検索して，結合，ORDER BY，GROUP BY などの操作をする場合は，4352～5120 を指定してください。指定すると，ソートでの入出力単位が大きくなるため，ソート処理時間が短くなります。 • 作業表用バッファ長に，SQL 文単位の作業表メモリ総容量を上限として，大きな値を指定するほど作業表作成時の作業表用ファイルへの入出力がなくなるので，作業表を作成する時間が短くなります。SQL 文単位の作業表メモリ総容量の見積もり式は次のとおりです。 **SQL 文単位の作業表メモリ総容量＝ a×b ＋ c ×d**
注意事項	• 複数のユーザが同時に実行する場合，及び複数個の作業表を使用する SQL 文などを実行する場合には，指定したサイズのバッファが，個々の作業表ごとに確保されます。そのため，大きな値を指定するとメモリを圧迫する原因となります。 • システムで確保できない大きなバッファサイズを指定した場合，サーバは起動できますが，作業表作成時にプロセス固有メモリの確保に失敗して，SQL エラーとなります。	• SQL 文単位に使用する作業表の数に比べて作業表用バッファ長に指定した値が小さいと，each を指定した場合に比べて処理時間が増加することがあります。具体的には，「SQL 文単位の作業表最大数×128」以上の値を指定してください。 SQL 文単位の作業表最大数の見積もり式は次のとおりです。 **SQL 文単位の作業表最大数＝ b ＋ d** • pool を指定した場合，システムで確保できない大きなバッファサイズを指定した場合，プロセス固有メモリの確保に失敗して，サーバを起動できません。
オペランドの規則	指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。	• 指定値は，128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。 • 384 以上の値を指定してください。384 未満の値が指定されると，自動的に 384 に切り上げられます。
省略値	• このオペランドを省略し，更にシングルサーバ定義の pd_work_buff_size オペランドを省略した場合，128 が仮定されます。 • このオペランドを省略して，更にバックエンドサーバ定義及びディクショナリサーバ定義の pd_work_buff_size オペランドを省略した場合，512 が仮定されます。	• このオペランドを省略し，更にシングルサーバ定義の pd_work_buff_size オペランドを省略した場合，384（32 ビットモードの場合）又は 5120（64 ビットモードの場合）が仮定されます。 • このオペランドを省略して，更にバックエンドサーバ定義及びディクショナリサーバ定義の pd_work_buff_size オペランドを省略した場合，1024（32 ビットモードの場合）又は 5120（64 ビットモードの場合）が仮定されます。

a：↑｛作業表（列情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

b：作業表（列情報格納用）の最大数※

c：↑｛作業表（位置情報格納用）の容量※（単位：キロバイト）÷2｝÷128↑×128

d：作業表（位置情報格納用）の最大数※

注※

これらの項目の見積もり方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《各見積もり式への影響》

pd_work_buff_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

◆ pd_work_buff_expand_limit = 作業表用バッファの増分上限値

～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：**((128～1000000))**
- 64 ビットモードの場合：**((128～4000000000))**

HiRDB が作成する作業表用バッファ長は pd_work_buff_size オペランドで指定します。この作業表用バッファの不足時，作業表用バッファを自動増分する場合にこのオペランドを指定します。このオペランドで指定した値になるまで作業表用バッファを増分します。

例えば，オペランドに次の指定をした場合，通常 1,024 キロバイトの作業表用バッファを確保します。作業表用バッファが 1,024 キロバイトで不足すると，2,048 キロバイトまで作業表用バッファを増分します。

- pd_work_buff_size = 1024
- pd_work_buff_expand_limit = 2048

次に示すときに HiRDB は作業表用バッファを増分します。

- 結合方式がハッシュジョイン，又は副問合せの実行方式にハッシュ実行が適用されている場合に，必要な作業表用バッファが確保できないとき
- 複数の作業表を同時に使用する場合，各作業表に作業表用バッファを 128 キロバイト割り当てて作業表用バッファが不足するとき

《前提条件》

pd_work_buff_mode オペランドを省略するか，又は pool 又は pool2 を指定している必要があります。

《利点》

作業表用バッファ不足（pd_work_buff_size オペランドの指定値不足）で UAP がエラーになるのを防げます。

《注意事項》

- 次に示すどちらかの条件を満たす場合は作業表用バッファを増分しません。
  ・pd_work_buff_expand_limit オペランドを指定しない
  ・pd_work_buff_expand_limit オペランドの値≦pd_work_buff_size オペランドの値
- システムで確保できない大きなバッファサイズを指定した場合，サーバは起動できますが，作業表作成時にプロセス固有メモリの確保に失敗して，SQL エラーとなります。
- 作業表用バッファを増分した場合はメモリを解放するため，常駐サーバプロセスについても次の契機にサーバプロセスを終了します（Linux 版を除く）。

  HiRDB/シングルサーバの場合：

  UAP を HiRDB から切り離したとき（DISCONNECT したとき）にシングルサーバプロセスを終了します。

  HiRDB/パラレルサーバの場合：

  ホールダブルカーソルを使用していない場合はトランザクションを正常終了するか，又は取り消したときに，バックエンドサーバプロセス又はディクショナリサーバプロセスを終了します。

  ホールダブルカーソルを使用している場合は，UAP を HiRDB から切り離したとき（DISCONNECT したとき）に，バックエンドサーバプロセス又はディクショナリサーバプロセスを終了します。

《オペランドの規則》

指定値は 128 の倍数を指定してください。128 の倍数でない場合，自動的に 128 の倍数に切り上げられます。

《ほかのオペランドとの関連》

各サーバプロセスでの作業表用バッファの初回増分時に KFPH29008-I メッセージを出力します。なお，pd_work_table_option オペランドで KFPH29008-I メッセージの出力を抑止できます。

《留意事項》

作業表用バッファを増分した場合，該当するサーバプロセスで使用中の作業表数が 0 になると，増分した作業表用バッファを解放します。作業表の使用数が 0 になるのは次に示すときです。

- 使用中のカーソルをすべてクローズしたとき（この場合，作業表の使用数が 0 にならないことがあります）
- ホールダブルカーソルを使用していない場合は，トランザクションを正常終了するか，又は取り消したとき
- ホールダブルカーソルを使用している場合は，UAP を HiRDB から切り離したとき（DISCONNECT したとき）

《備考》

「ハッシュジョイン，副問合せのハッシュ実行」は，次に示す場合に適用されます。

- pd_additional_optimize_level オペランド，クライアント環境定義の PDADDITIONALOPTLVL オペランド，又は SQL コンパイルオプションの ADD OPTIMIZE LEVEL オペランドで，「コストベース最適化モード 2 の適用」と一緒に「ハッシュジョイン，副問合せのハッシュ実行」を指定する

- SQL 文中の結合方式の SQL 最適化指定に HASH を指定する
- SQL 文中の副問合せ実行方式の SQL 最適化指定に HASH を指定する

《各見積もり式への影響》

pd_work_buff_expand_limit オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

◆ pdwork -v "HiRDB ファイルシステム領域名称"〔,"HiRDB ファイルシステム領域名称"〕... ～＜141 文字以内のパス名＞

作業表用ファイル用の HiRDB ファイルシステム領域の名称を指定します。作業表用ファイルとは，SQL 文を実行するときに一時的に情報を格納するファイルのことで HiRDB が自動的に作成します。作業表用ファイルを必要とする SQL 文については，マニュアル「HiRDB システム導入・設計ガイド」の「作業表用ファイルの概要」を参照してください。

なお，このオペランドは省略しないでください。省略した場合，作業表用ファイルを必要とする SQL 文を実行できないことがあります。

《注意事項》

- このオペランドに指定する HiRDB ファイルシステム領域は pdfmkfs コマンドで初期設定した HiRDB ファイルシステム領域を指定してください。
- 作業表用ファイルの容量が大きい場合は，領域長が大きい HiRDB ファイルシステム領域を指定してください。作業表用ファイルの容量見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

  ただし，HiRDB ファイルシステム領域を pdfmkfs -a コマンドで初期設定した場合，-n オプションで指定した値を使い切ると，HiRDB は領域サイズを自動で拡張します。pdfmkfs コマンドについては，マニュアル「HiRDB コマンドリファレンス」を参照してください。
- 作業表用ファイル用の HiRDB ファイルシステム領域は，RD エリア及びシステムファイル用の HiRDB ファイルシステム領域と同じものを指定できません。
- このオペランドに HiRDB ファイルシステム領域を二つ以上指定している場合，作業表用ファイルの作成時にエラーが発生した HiRDB ファイルシステム領域は，それ以降，原則使用しません。ほかの HiRDB ファイルシステム領域だけを使用します。

  例外として，容量不足やファイル数オーバーなどによって，ほかのすべての HiRDB ファイルシステム領域で作業表用ファイルの作成が失敗した場合に，使用されなくなった HiRDB ファイルシステム領域で作業表用ファイルを作成しようとします。その結果，作業表用ファイルが正常に作成された場合，その HiRDB ファイルシステム領域を再使用するようになります。

  なお，該当のサーバ（HiRDB/シングルサーバの場合は HiRDB）を一度終了して開始すると，作業表用ファイルの作成時にエラーが発生して使用されなくなった HiRDB ファイルシステム領域が使用できるようになります。

《オペランドの規則》

- HiRDB ファイルシステム領域名称は必ず 1 個以上指定してください。
- HiRDB ファイルシステム領域名称は最大 16 個指定できます。
- このオペランドはバックエンドサーバ定義内に一つだけ指定できます。複数個指定した場合は最初に指定したものが有効となります。
- HiRDB/パラレルサーバの場合は，ほかのサーバと同じ HiRDB ファイルシステム領域を指定しないでください。
- HiRDB ファイルシステム領域を複数個指定する場合，キャラクタ型スペシャルファイル上に作成した HiRDB ファイルシステム領域と通常ファイル上に作成した HiRDB ファイルシステム領域を混在して指定できます。

9.2.5 HiRDB の開始方法に関するオペランド

◆ **pd_mode_conf = AUTO ｜ MANUAL1 ｜ MANUAL2**

HiRDB の開始方法を指定します。

AUTO：

HiRDB を自動開始します。自動開始とは，OS を起動すると自動的に HiRDB も開始する開始方法です。

ただし，次に示す場合は手動開始になります。手動開始とは pdstart コマンドを実行して HiRDB を開始する開始方法です。

- 正常終了後の正常開始（OS を再起動しない場合）
- 計画停止後の再開始（OS を再起動しない場合）
- 強制終了後の再開始（OS を再起動しない場合）

MANUAL1：

手動開始します。ただし，異常終了後の再開始は自動開始になります。

MANUAL2：

手動開始します。

《指定値の目安》

- HiRDB（ユニット）が異常終了した場合に HiRDB（ユニット）を自動的に再開始したい場合は，AUTO 又は MANUAL1 を指定します。
- 系切り替え機能を使用する場合は MANUAL1 又は MANUAL2 を指定します。MANUAL1 を指定するか，又は MANUAL2 を指定するかは，系切り替え機能の運用方法によって異なります。系切り替え機能の運用方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- このオペランドの指定値と前回の終了モード（正常終了，強制終了，計画停止，又は異常終了）の組み合わせによって，次のように開始モードが決まります。

pd_mode_conf の値	前回の終了モード	開始モード	開始方法
AUTO	正常終了	正常開始	自動開始又は手動開始※1
	計画停止	再開始※2	
	強制終了	再開始※2	
	異常終了	再開始	自動開始
MANUAL1	正常終了	正常開始	手動開始
	計画停止	再開始※2	
	強制終了	再開始※2	
	異常終了	再開始	OS 起動時は手動開始 それ以外は自動開始
MANUAL2	正常終了	正常開始	手動開始
	計画停止	再開始※2	
	強制終了	再開始※2	
	異常終了	再開始※2	

注※1

自動開始になるのは OS を起動するときだけです。正常終了後の正常開始又は計画停止後の再開始（OS を再起動しない場合）のときは手動開始になります。

注※2

pdstart コマンドの dbdestroy オプションで強制開始できます。ただし，強制開始すると，HiRDB はデータベースの内容を回復しません。したがって，HiRDB 管理者がデータベースを回復する必要があります。

- ログ同期方式のリアルタイム SAN レプリケーションを使用する場合は，このオペランドの指定値と前回の終了モードに関係なく，開始モードは常に正常開始になります。ただし，ログ適用サイトの場合，変更できるシステム定義のオペランドは，再開始時に変更できるオペランドだけとなります。

《注意事項》

- HiRDB/パラレルサーバで自動開始（pd_mode_conf=AUTO）をする場合は，最初のユニットを開始してから 20 分以内に全ユニットを開始するようにしてください。20 分以内に全ユニットを開始しないと，HiRDB の開始処理を中止します。この 20 分という制限時間は pd_reduced_check_time オペランドで変更できます。

  なお，ユニットの異常終了，又は OS の異常終了後の再起動ではこのような制限時間はありません。
- リアルタイム SAN レプリケーションを使用する場合は，AUTO を指定できません。AUTO を指定すると，HiRDB の開始処理がエラーになります。

- ログ同期方式のリアルタイム SAN レプリケーションを使用する場合，ログ適用サイトでこのオペランドに MANUAL1 を指定しても，MANUAL2 を指定したものと仮定されます。
- pd_mode_conf オペランドに AUTO や MANUAL1 を指定した場合でも，HiRDB の開始処理中又は終了処理中に HiRDB が異常終了したときは，次回の開始は必ず手動開始となります。異常終了時のメッセージを確認して対処したあと，手動で HiRDB を開始してください。

◆ pd_system_complete_wait_time = pdstart コマンド完了待ち時間

～<符号なし整数>((610～3600))《610》（単位：秒）

このオペランドは，pdstart コマンドの完了待ち時間を長くする場合に指定します。HiRDB は，pdstart コマンドが入力されてから 610 秒（10 分 10 秒）を過ぎても開始処理が終了しないと，KFPS05078-I メッセージを出力して pdstart コマンドをエラーリターンしています。この 610 秒を長くする場合に指定します。

このオペランドの指定を省略した場合に HiRDB の常駐プロセス数が 610 個を超えていると，pdstart コマンド完了待ち時間がオペランドの省略値（610 秒）を超えることがあります。

なお，pdstart コマンドがエラーリターンしても，開始処理は続行され HiRDB は開始します。したがって，次に示す場合にこのオペランドを指定します。

- KFPS05078-I メッセージを監視対象としている場合
- pdstart コマンドの正常終了後にほかの操作を自動的に行っている場合

《適用基準》

通常，このオペランドを指定する必要はありません。

HiRDB の開始処理時に KFPS05078-I メッセージが出力されて，pdstart コマンドがエラーリターンする場合に指定します。

◆ pd_start_time_out = HiRDB 開始準備処理の最大待ち時間

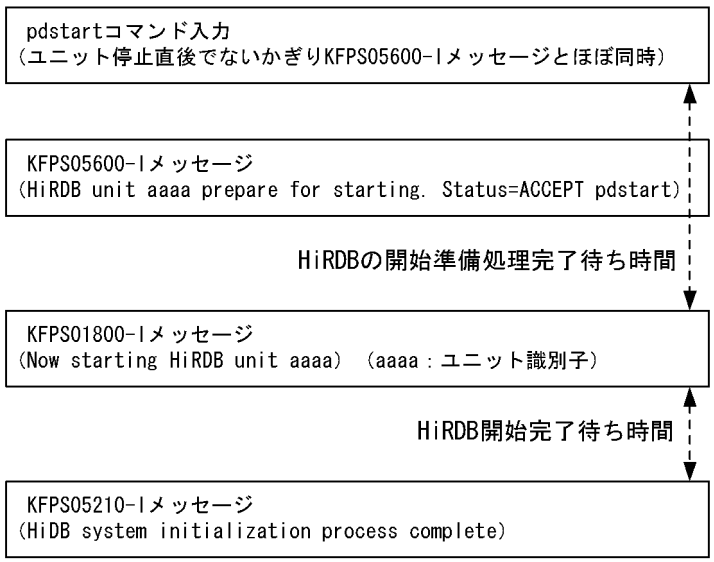
～<符号なし整数>((1～1440))《15》（単位：分）

pdstart コマンドの入力から HiRDB の開始準備処理の完了までの最大待ち時間を指定します。pdstart コマンドの入力からこのオペランドに指定した時間が経過しても HiRDB の開始準備処理が終了しない場合，pdstart コマンド入力画面に KFPS01861-E メッセージ（reason code = TIMEOUT）が出力されます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。一般的な環境では HiRDB の開始準備処理時間は長くても 3～6 分のため，最大待ち時間はこのオペランドの省略値の 15 分で十分です。サーバマシンの性能，ディスク容量，又はメモリ負荷などの要因によっては，HiRDB の開始準備処理に 15 分以上掛かることがまれにあります。この場合，pdstart コマンド入力画面に KFPS01861-E メッセージ（reason code = TIMEOUT）が出力されます。このとき，このオペランドに 15 より大きい値を指定してください。

> **参考**
>
> HiRDB の開始準備処理の完了待ち時間と HiRDB 開始完了待ち時間は，次に示すメッセージの出力時刻から算出できます。



◆ pd_term_watch_count = 連続異常終了回数の上限

～＜符号なし整数＞((1～3))《3》

HiRDB（HiRDB/パラレルサーバの場合はユニット）は 30 分以内にこのオペランドに指定した回数の異常終了が発生すると，KFPS00715-E メッセージを出力して PAUSE 状態になります。

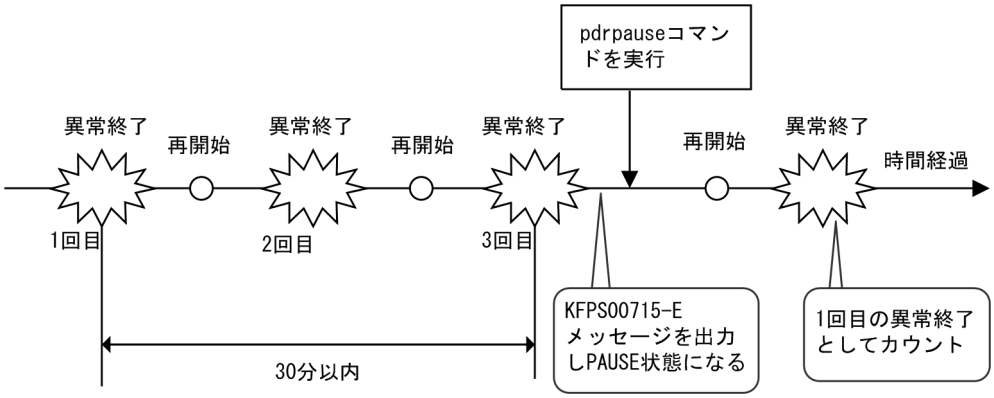
PAUSE 状態になると，pdstart コマンドが実行できなくなり，HiRDB を再開始できなくなります。

PAUSE 状態になっているかどうかは，pdls -d ust コマンドで確認できます。

PAUSE 状態を解除するには，KFPS00715-E メッセージの付加情報に従って異常終了の原因を取り除いた後に，pdrpause コマンドを実行してください。なお，HiRDB/パラレルサーバの場合は，PAUSE 状態のユニットで pdrpause コマンドを実行してください。

《指定値の目安》

- このオペランドに 1 を指定すると，異常終了が 1 回発生すると PAUSE 状態になります（異常終了した後に自動的に再開始しません）。再開始したくない場合は，1 を指定してください。再開始したい場合は，2 又は 3 を指定してください。
- このオペランドに 3 を指定すると，1 回目の異常終了については再開始します，2 回目の異常終了についても再開始しますが，3 回目の異常終了が発生すると PAUSE 状態になります。このオペランドに 3 を指定したとき時の動作を次の図に示します。



《注意事項》

系切り替え機能を適用している場合，実行系と待機系の間で異常終了回数のカウントを引き継ぎません。

9.2.6　縮退起動に関するオペランド

◆ **pd_start_level = 0 ｜ 1**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB の開始時に開始できないユニットがある場合，縮退起動をするかどうかを指定します。

縮退起動については，マニュアル「HiRDB システム運用ガイド」を参照してください。

0：

縮退起動をしません。障害などで開始できないユニットが一つでもあると HiRDB を開始できません。

1：

縮退起動をします。障害などで開始できないユニットがあっても，HiRDB（それ以外のユニット）を開始できます。

◆ **pd_reduced_check_time = 縮退起動開始の連絡待ち時間**

～＜符号なし整数＞((300～1200))《1200》（単位：秒）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

各ユニットからの開始処理完了の連絡待ち時間を指定します。ここで指定した時間を過ぎても，ユニット開始処理完了の連絡がない場合は，そのユニットを除いて縮退起動をします。

《前提条件》

pd_start_level オペランドに 1 を指定する必要があります。

《指定値の目安》

pdstart コマンドを入力してから，KFPS01826-I メッセージ（システムマネジャを定義しているユニットに出力される KFPS01826-I メッセージ）が出力されるまでの時間を調べてください。このオペランドには，それ以上の値を指定してください。

《注意事項》

このオペランドで指定した連絡待ち時間が過ぎても，縮退起動ができる条件をすべて満たしていないと，縮退起動できません。このとき，HiRDB は異常終了します。縮退起動できる条件については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_start_skip_unit = 開始しないユニットの名称〔，開始しないユニットの名称〕...**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

障害などで開始できないユニットがあるため，縮退起動をする場合，そのユニット名称を指定します。ここで指定したユニットは起動されません。HiRDB を縮退起動するとき，KFPS05048-Q メッセージを表示して縮退起動してよいかどうかの応答を待ちます。"y"を応答すると，縮退起動します。"y"以外を応答すると，HiRDB の開始処理を中止します。

《前提条件》

pd_start_level オペランドに 1 を指定する必要があります。

《利点》

HiRDB を開始するとき，HiRDB は各ユニットからの起動開始の連絡を最大 20 分待ちます。したがって，縮退起動をするとき，最大 20 分間の起動開始の連絡待ち時間が発生します。このオペランドに起動できないユニットを指定すると，このユニットの起動開始の連絡待ちをしません。したがって，最大 20 分間の起動開始の連絡待ち時間が解消されるため，縮退起動の時間を短縮できます。

《注意事項》

- 影響分散スタンバイレス型系切り替え機能を使用している場合，HA グループ内の全ユニットが起動していなくてもよいため，このオペランドの指定は無効になります。ただし，HA グループ内の全バックエンドサーバがどこかのユニットで起動していないと，HiRDB を開始できません。
- 回復不要 FES ユニットをこのオペランドに指定しても，回復不要 FES ユニットは縮退起動の対象になりません。

9.2.7 HiRDB の処理方式に関するオペランド

◆ **pd_dbsync_point = sync | commit**

データベースの更新内容をファイルに反映するタイミングを指定します。

sync：

シンクポイント時点でデータベースの更新内容をファイルに反映します。シンクポイント間で同一ページを更新するようなトランザクションが多く起動される場合に性能が向上します。COMMIT 文が発行されてもファイルに更新情報を反映しないため，入出力の負荷が軽減されます。ただし，commit を指定したときに比べて全面回復処理が遅くなります。

commit：

COMMIT 文発行時点でデータベースの更新内容をファイルに反映します。トランザクションの完了時点でデータベースの内容が保証されるため，シンクポイント時点からデータベースを回復する必要がなく，全面回復処理に要する時間が短縮できます。ただし，シンクポイント間で同一ページを更新するようなトランザクションが多く起動される場合は，sync を指定したときに比べて性能が低下します。

《注意事項》

次の場合はこのオペランドに sync を指定するか，このオペランドを省略してください。

- ハイブリッド方式のリアルタイム SAN レプリケーションを使用している場合
- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合

《備考》

LOB 用 RD エリアはこのオペランドの影響を受けません。ディレクトリ部は COMMIT 文発行時点で反映されます。データ部は LOB 用グローバルバッファを割り当てているかどうかによって処理が異なります。LOB 用グローバルバッファを割り当てていない場合は更新要求時にすぐに反映されま

す。LOB 用グローバルバッファを割り当てている場合は COMMIT 文発行時点で反映されます。ただし，グローバルバッファが満杯になったときはその時点で反映されます。

《ほかのオペランドとの関連》

このオペランドは pd_system_dbsync_point オペランドと関連があります。

《各見積もり式への影響》

pd_dbsync_point オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「シングルサーバが使用する共用メモリの計算式」
- 「ディクショナリサーバが使用する共用メモリの計算式」
- 「バックエンドサーバが使用する共用メモリの計算式」

◆ **pd_system_dbsync_point = sync ｜ commit**

次に示す RD エリアの更新内容をファイルに反映するタイミングを指定します。

- マスタディレクトリ用 RD エリア
- データディレクトリ用 RD エリア
- データディクショナリ用 RD エリア
- データディクショナリ LOB 用 RD エリア
- レジストリ用 RD エリア
- レジストリ LOB 用 RD エリア

sync：

シンクポイント時点で前記の RD エリアの更新内容をファイルに反映します。COMMIT 文が発行されてもファイルに更新情報を反映しないため，定義系 SQL の処理性能が commit を指定したときに比べて若干向上します。ただし，全面回復処理が commit を指定したときに比べて遅くなります。

commit：

COMMIT 文発行時点で前記の RD エリアの更新内容をファイルに反映します。トランザクションの完了時点で前記の RD エリアの更新内容が保証されるため，前記の RD エリアをシンクポイント時点から回復する必要がなく，全面回復処理に要する時間が短縮できます。ただし，sync を指定したときに比べて，定義系 SQL の処理性能が若干低下します。

《ほかのオペランドとの関連》

このオペランドは pd_dbsync_point オペランドと関連があります。pd_dbsync_point オペランドとの関連を次に示します。

pd_dbsync_point オペランドの値	pd_system_dbsync_point オペランドの値	
	sync	commit（省略値）
sync（省略値）	全 RD エリアの更新内容をシンクポイント時点で反映します。	前記の RD エリアの更新内容は COMMIT 文発行時点で反映します。そのほかの RD エリアの更新内容はシンクポイント時点で反映します。
commit	全 RD エリアの更新内容を COMMIT 文発行時点で反映します。	

《各見積もり式への影響》

pd_system_dbsync_point オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」
- 「ディクショナリサーバが使用する共用メモリの計算式」

◆ **pd_process_terminator = resident ｜ fixed ｜ nonresident**

HiRDB のプロセスが異常終了した場合，HiRDB は後処理を実行するプロセス（これを**後処理プロセス**といいます）を起動して後処理をします。このオペランドでは，後処理プロセスを HiRDB の開始時に起動しておくかどうかを指定します。

resident：

HiRDB の開始時に後処理プロセスを一つ起動します。HiRDB/パラレルサーバの場合は，一つのユニットに一つの後処理プロセスを起動します。

複数のプロセスが同時に異常終了した場合，HiRDB が規定した数まで後処理プロセスを起動して後処理を並列実行します。メモリ不足などで後処理プロセスを新たに起動できない場合は，既に起動している後処理プロセスで後処理を順次実行します。

なお，メモリ不足などによって，HiRDB が規定した数まで後処理プロセスを起動できない場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）が異常終了することがあります。

fixed：

HiRDB の開始時に，pd_process_terminator_max オペランドで指定した数の後処理プロセスを起動します。HiRDB/パラレルサーバの場合は，ユニットごとに pd_process_terminator_max オペランドで指定した数の後処理プロセスを起動します。メモリ不足などで後処理プロセスを起動できない場合は HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を開始しません。

pd_process_terminator_max オペランドの値を超えるプロセスが同時に異常終了した場合，後処理プロセスを追加起動しません。この場合，既に起動している後処理プロセスで後処理を順次実行します。

nonresident：

HiRDB の開始時に後処理プロセスを起動しません。プロセスが異常終了するたびに後処理プロセスを起動します。

複数のプロセスが同時に異常終了した場合，複数の後処理プロセスを同時に起動して後処理を並列実行します。メモリ不足などで後処理プロセスを起動できない場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）が異常終了することがあります。

《指定値の目安》

- 信頼性を向上する場合は resident 又は fixed を指定します。後処理についての性能面は resident より fixed の方が優れていますが，fixed の方がより多くのメモリを必要とします。
- nonresident を指定すると，オンデマンドに後処理プロセスが起動されます。このため，メモリ不足が発生すると，後処理プロセスが起動できなくなります。また，プロセスの異常終了が重なった場合，複数の後処理プロセスが起動されるため，性能面でも好ましくありません。
- 後処理プロセスを起動できない場合に，HiRDB が異常終了することを防止したいときは，fixed の指定を推奨します。

《注意事項》

指定値を fixed に変更する場合は注意が必要です。HiRDB の開始時に後処理プロセスを起動するため，より多くのメモリが必要になります。メモリ不足などで後処理プロセスを起動できない場合は HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を開始しません。

◆ **pd_process_terminator_max = 後処理プロセスの最大常駐数**

～<符号なし整数>((1～100))《Max(3，↑(pd_max_users の値＋ pd_max_reflect_process_count の値)÷100↑)》

pd_process_terminator オペランドを省略するか，又は fixed を指定した場合にこのオペランドを指定します。このオペランドには，HiRDB の開始時に起動する後処理プロセスの数を指定します。メモリ不足などで，ここで指定した数の後処理プロセスを起動できない場合は，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を開始しません。

《指定値の目安》

- 必要になる後処理プロセス数は，pd_max_users の値＋ pd_max_reflect_process_count の値に比例します。値が小さいと回復処理が遅延するおそれがあり，値が大きいと余分にメモリを消費します。

◆ **pd_thdlock_wakeup_lock = Y ｜ N**

このオペランドは，AIX 版限定のオペランドです。

スレッド間ロックの解放通知方式を指定します。このオペランドに Y を指定することで，解放通知が確実に送信されることを保証します。

Y：

スレッド間ロックの解放を通知するときに，これとは別の一時的なロックを新たに取得します。

N：

スレッド間ロックの解放を通知するときに，これとは別の一時的なロックを新たに取得しません。

《指定値の目安》

UAP の多重実行時に，SQL 実行時間の平均値が数十ミリ秒であるが，最大値が 1 秒以上になる場合には Y を指定してください。

《注意事項》

- N を指定した場合，又はこのオペランドを省略した場合は，UAP の多重実行時に，SQL 実行時間が 1 秒程度遅延することがあります。

- Y を指定した場合は，UAP の多重実行時に，SQL 実行時間が数ミリ秒程度遅延することがあります。

◆ **pd_pageaccess_mode = SNAPSHOT | NORMAL**

データベース検索時のページアクセス方式を指定します。

SNAPSHOT：

ページアクセス方式を**スナップショット方式**にします。グローバルバッファへの初回アクセス時，探索条件に一致する行をプロセス固有メモリにコピーします。2 回目の検索要求時にはプロセス固有メモリを参照して検索結果を返します。スナップショット方式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

NORMAL：

ページアクセス方式を通常方式にします。検索要求ごとにグローバルバッファをアクセスします。

《指定値の目安》

グループ分け高速化機能など，性能向上を目的とした機能を使用できない場合に，スナップショット方式の適用を検討してください。通常の検索 SQL では，指定された探索条件に一致する行の数とほぼ同じ回数分だけグローバルバッファへのアクセスを行っています。このため，検索 SQL の実行が重なるとグローバルバッファへのアクセスが集中して，期待した性能が得られないことがあります。このような場合にスナップショット方式を適用すると，検索 SQL のグローバルバッファへのアクセス回数を削減できるため，検索性能が向上することがあります。ただし，スナップショット方式を適用した場合，HiRDB が使用するプロセス固有メモリが増加します。スナップショット方式を適用した場合のプロセス固有メモリの計算式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《各見積もり式への影響》

pd_pageaccess_mode オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「スナップショット方式指定時に必要なメモリ所要量の求め方」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「スナップショット方式指定時に必要なメモリ所要量の求め方」

◆ **pd_cmdhold_precheck = Y | N**

RD エリアの閉塞状態のチェックを RD エリアの排他処理の前に行うかどうかを指定します。

Y：

RD エリアの閉塞状態のチェックを RD エリアの排他処理の前に行います。

N：

RD エリアの閉塞状態のチェックを RD エリアの排他処理の前に行いません。排他取得後に行います。

チェック対象になる閉塞の種類を次に示します。

- コマンド閉塞
- 参照可能閉塞

- 参照可能バックアップ閉塞

《指定値の目安》

通常はこのオペランドを省略するか，又は Y を指定してください。Y を指定したときと，N を指定したときの違いを次に示します。

項目	Y を指定した場合	N の場合
HiRDB の処理	UAP 又はコマンド※の実行時，RD エリアの排他を取得する前に，アクセスする可能性があるすべての RD エリアの閉塞状態をチェックします。例えば，RD エリア 1〜4 に横分割されている表にアクセスする場合，RD エリア 1〜4 の閉塞状態をチェックします。ただし，キーレンジ分割，又は FIX ハッシュ分割で条件を指定して，UAP 側でアクセスする可能性がある RD エリアを絞り込んでいる場合，アクセスする可能性がない RD エリアが閉塞していてもエラーにはなりません。	UAP 又はコマンド※の実行時，RD エリアの排他を取得した後に，アクセスする可能性があるすべての RD エリアの閉塞状態をチェックします。例えば，RD エリア 1〜4 に横分割されている表にアクセスするとします。インデクスを使用してアクセスする RD エリアを絞り込んだ結果，アクセス対象 RD エリアが RD エリア 1 の場合，RD エリア 1 に対してだけ閉塞状態をチェックします。これは HiRDB Version 5.0 以前の処理方式です。
閉塞中の RD エリアに UAP がアクセスした場合	RD エリアの排他を取得する前に閉塞チェックをするため，RD エリアが閉塞していることを，N を指定したときよりも早く検知できます。	RD エリアの排他を取得した後に閉塞チェックをするため，閉塞中の RD エリアに UAP がアクセスした場合，RD エリアの排他によるタイムアウトエラー（KFPA11770-E）が発生することがあります。 また，アクセス対象 RD エリアがデータロード又は再編成で閉塞中の場合，データロード又は再編成処理の終了後に UAP が閉塞エラー（KFPA11920-E）になることがあります。
非横分割インデクスを使用してアクセスする RD エリアを絞り込む場合	表が横分割されていてインデクスが横分割されていない場合に注意が必要です。非横分割インデクスを使用してアクセス対象 RD エリアを絞り込む場合，アクセス対象外の RD エリアが閉塞していても，閉塞エラー（KFPA11920-E）になります。 HiRDB の処理で説明した例の場合，RD エリア 1〜4 のどれかが閉塞していると，UAP が閉塞エラー（KFPA11920-E）になります。	非横分割インデクスを使用してアクセス対象 RD エリアを絞り込む場合，アクセス対象外の RD エリアが閉塞していても，UAP 又はコマンドを実行できます。HiRDB の処理で説明した例の場合，RD エリア 2〜4 が閉塞していても，UAP を実行できます。

注※　RD エリアが閉塞状態では実行できない UAP 及びコマンドが対象になります。

◆ pd_db_io_error_action = dbhold ｜ unitdown

RD エリア（マスタディレクトリ用 RD エリアを除く）の入出力エラーが発生したときの HiRDB の処理を指定します。なお，マスタディレクトリ用 RD エリアに入出力エラーが発生した場合，このオペランドの指定に関係なく常に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。RD エリアの入出力エラーが発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

ここでいう入出力エラーとは，HiRDB がファイルを操作したときに HiRDB が判別できない理由でファイル操作に失敗したエラーを意味し，HiRDB ファイルシステムに対するアクセス要求から返されるエラーコードに-1544 が出力されるエラーのことです。

dbhold：

RD エリアの入出力エラーが発生した場合，その RD エリアを障害閉塞します。

unitdown：

RD エリアの入出力エラーが発生した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。ただし，異常終了後に再度入出力エラーが発生した場合，その RD エリアを障害閉塞します。再び，unitdown の指定を有効にするには，次に示すどちらかの方法を実行してください。

- HiRDB を正常開始する
- システム構成変更コマンド（pdchgconf コマンド）を実行する

《指定値の目安》

マニュアル「HiRDB システム運用ガイド」の「RD エリアの入出力エラーが発生したときの対処方法」を参照して，このオペランドの指定値を決めてください。

《注意事項》

- unitdown を指定したときに入出力エラーが発生すると HiRDB が異常終了するため，次の場合に，処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している場合

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している場合

  物理エラー検知時ユニットダウン機能を使用する場合は，できるだけこれらの運用は避けてください。もし，これらの運用が必要な場合は，RD エリアが閉塞しても最新の状態に回復できるよう，UAP 又はユティリティの実行前にバックアップを取得してください。バックアップの取得については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- 開始処理中又は終了処理中の入出力エラーについては，unitdown を指定しても HiRDB を異常終了しません。
- ログ同期方式のリアルタイム SAN レプリケーション機能を適用している場合，ログ適用サイトの HiRDB の開始中は，unitdown を指定しても無効になります。
- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していても HiRDB を異常終了しません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_mode_conf オペランド
- pd_db_access_error_action オペランド
- pd_db_hold_action オペランド

なお，pd_db_io_error_action オペランド，pd_db_access_error_action オペランド，及び pd_db_hold_action オペランドのうち，複数のオペランドで unitdown を指定した場合は，次の順序でオペランドの指定値が有効になります。

1. pd_db_io_error_action オペランド
2. pd_db_access_error_action オペランド
3. pd_db_hold_action オペランド

そのため，RD エリアの入出力エラー，ファイルアクセスエラー，及び物理エラーのうち複数のエラーが同時に起こった場合，どのエラー要因によってユニットダウンしたのかは，この優先順位を参考に判断してください。また，出力されたメッセージを参照してください。

◆ **pd_connect_errmsg_hide = Y ｜ N**

CONNECT 失敗時に出力されるメッセージで，エラーの要因を隠すかどうかを指定します。

Y：CONNECT 失敗時にエラーの要因を隠します。

N：CONNECT 失敗時にエラーの要因を隠しません。

このオペランドの指定値によって CONNECT 失敗時に出力されるメッセージが変わることがあります。詳細を次に示します。

エラー要因	出力されるメッセージ	
	Y を指定した場合	N（省略値）を指定した場合
認可識別子の不正（指定されたユーザは存在しない）	KFPA19632-E	KFPA11561-E
パスワードの不正（指定されたパスワードが一致しない）	KFPA19632-E	KFPA11560-E

◆ **pd_cancel_down_msgchange = Y ｜ N**

サーバプロセスの強制終了が発生した場合に，出力されるエラーメッセージを変更するかどうかを指定します。

Y：

エラーメッセージを警告メッセージに変更します。エラーメッセージを変更する機能を**トランザクションキャンセル時のプロセスダウンメッセージ変更機能**といいます。

N：

エラーメッセージを変更しません。

このオペランドの指定値と出力されるエラーメッセージの関係を次に示します。

条件	出力されるメッセージ	
	Y（省略値）を指定した場合	N を指定した場合
次に示す原因によってサーバプロセスの強制終了が発生した場合※ • ユーザ操作による意図的な強制終了 • タイムアウトによる強制終了 • クライアント側の障害による強制終了	• KFPS01852-W • KFPO00115-W	• KFPS01820-E • KFPO00105-E
上記以外の原因によってサーバプロセスの強制終了が発生した場合	• KFPS01820-E • KFPO00105-E	

条件	出力されるメッセージ	
	Y（省略値）を指定した場合	N を指定した場合
サーバプロセスの強制終了の原因を HiRDB が判断できない場合		

注※

メッセージの変更が発生する原因はほかにもあります。詳細については，マニュアル「HiRDB システム運用ガイド」の「トランザクションキャンセル時のプロセスダウンメッセージ変更機能」を参照してください。

《利点》

このオペランドに Y を指定すると，サーバプロセスの強制終了が発生した原因で出力されるメッセージを切り分けられます。

《備考》

KFPS01820-E 及び KFPO00105-E メッセージが出力された場合，その原因が HiRDB の不正検知によるものか，又はトランザクションキャンセルなどのユーザ操作による意図的なものかをメッセージ ID から判断できません。判断するには，各メッセージに出力されるプロセス ID を比較する必要があります。

JP1 を使用してメッセージを監視している場合，複数のメッセージ情報を比較できないため，KFPS01820-E 及び KFPO00105-E メッセージが出力されたときに対処が難しくなることがあります。このオペランドに Y を指定すると，出力されるメッセージがエラー原因によって切り分けられるため，対処が特定しやすくなります。したがって，JP1 を使用してメッセージを監視している場合，このオペランドに Y を指定することをお勧めします。

◆ pd_server_process_adjust_limit = 1 回の要求で起動，停止するサーバプロセス数の上限値 ～<符号なし整数>

- HiRDB/シングルサーバの場合：((1～3000))
- HiRDB/パラレルサーバの場合：((1～12000))

サーバプロセスの起動，停止処理で，1 回の要求で起動，停止するサーバプロセス数の上限を指定します。

このオペランドを指定すると，サーバプロセスの起動，停止処理を 1 回の要求でまとめて実行しないで，複数回の契機に分割して起動，停止処理を行います。

なお，オペランドを指定しない場合は，サーバプロセスの起動，停止処理を 1 回の要求でまとめて実行します。

このオペランドが有効になる契機を次に示します。

- HiRDB システムを開始する場合
- HiRDB ユニットを開始する場合
- HiRDB サーバを開始する場合
- 系切り替え構成で待機系を起動する場合
- pdpfresh コマンドを実行する場合

- pdchprc コマンドを実行し，常駐プロセス数を減らす場合

《利点》

常駐するユーザサーバプロセス数が多いシステムの場合，サーバプロセスの起動，又は停止時に CPU の負荷を軽減できます。

《指定値の目安》

サーバプロセスの起動，又は停止時に CPU 利用率が高騰する場合，このオペランドを指定してください。

なお，pd_server_process_adjust_limit オペランドを指定すると，サーバプロセスの起動，停止処理に要する時間が次のとおりとなります。

運用	時間
HiRDB システム開始	↓サーバの常駐プロセス数／pd_server_process_adjust_limit オペランド指定値↓×pd_server_process_adjust_intvl オペランド指定値（ミリ秒）
HiRDB ユニット開始	
HiRDB サーバ開始	
待機系起動	
pdpfresh コマンド実行	↓リフレッシュ対象のサーバプロセス数／pd_server_process_adjust_limit オペランド指定値↓×2×pd_server_process_adjust_intvl オペランド指定値（ミリ秒）
pdchprc コマンド実行（サーバプロセス数減少）	↓（現在の常駐プロセス数−pdchprc コマンドに指定した値）／pd_server_process_adjust_limit オペランド指定値↓×10 秒

《注意事項》

- HiRDB/パラレルサーバでユニット内に複数のサーバを配置している場合は，オペランドに指定した値がそれぞれのサーバに対して有効となります。
- 起動しているサーバプロセスの数よりも大きい値を指定した場合，サーバプロセスの起動，停止要求は 1 回になります。
- pdpfresh コマンドを複数同時に実行しないでください。

  同時に実行した場合，サーバプロセスのリフレッシュに失敗し，サーバプロセスの起動ができなくなることがあります。

  この場合は，pdchprc コマンドの-p オプションに pdpfresh コマンド実行前の常駐プロセス数を指定して，pdchprc コマンドを実行してください。
- pdpfresh コマンドと-p オプションを指定した pdchprc コマンドを同時に実行しないでください。

  同時に実行した場合，サーバプロセスのリフレッシュに失敗することがあります。

  この場合は再度 pdpfresh コマンドを実行して，リフレッシュを完了してください。

  リフレッシュが完了した時点では，同時実行した pdchprc コマンドによる常駐プロセス数の変更内容が反映されていないことがあるため，pdchprc コマンドを再度実行して常駐プロセス数を変更してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_server_process_adjust_intvl オペランド

◆ **pd_server_process_adjust_intvl = サーバプロセスの分割起動，停止要求間隔〜<符号なし整数>((1〜3600000))<<50>>(単位：ミリ秒)**

サーバプロセスを複数回に分けて起動，又は停止する間隔を指定します。

このオペランドは，pd_server_process_adjust_limit オペランドを指定している場合に有効となります。

《指定値の目安》

pd_server_process_adjust_limit オペランドを指定しても CPU 利用率が長時間高騰する現象が発生する場合，このオペランドを指定して間隔を長くしてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_server_process_adjust_limit オペランド

◆ **pd_cmd_rmode = rsh | ssh**

このオペランドは UNIX 版限定のオペランドです。

HiRDB がコマンドをリモート実行する場合や，ファイル転送を行う場合に使用する OS のコマンドを選択します。

rsh：

rsh，rcp コマンドを用いてリモート実行やファイル転送を行います。

ssh：

ssh，scp コマンドを用いてリモート実行やファイル転送を行います。

《前提条件》

このオペランドに ssh を指定するには，リモート実行やファイル転送の対象となるすべてのサーバマシンで SSH プロトコルを利用可能にする必要があります。対象となるサーバマシンについては，マニュアル「HiRDB システム導入・設計ガイド」の「リモートシェル実行環境の設定」を参照してください。

《指定値の目安》

HiRDB を構築する環境に合わせて，適切な値を指定してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_cmd_rconfig オペランド
- pd_cmd_rlog オペランド

◆ **pd_cmd_rconfig = SSH クライアント設定ファイル〜<パス名>((255 文字以内))**

このオペランドはシステム定義 pd_cmd_rmode に ssh を指定した場合に有効なオペランドです。

HiRDB がリモート実行やファイル転送で ssh，scp コマンドを使用する場合に，コマンドに指定する SSH クライアント設定ファイルを絶対パス名で指定します。指定したファイルは，ssh，scp コマンドの-F オプションに指定されます。

HiRDB が ssh，scp コマンドを実行する場合に，ユーザ設定ファイル（~/.ssh/config）やシステム全体の設定ファイル（/etc/ssh/ssh_config）とは異なる設定を適用したいときに指定してください。設定ファイルについては，OS のマニュアルを参照してください。

《前提条件》

pd_cmd_rcmd オペランドに ssh を指定している必要があります。

◆ **pd_cmd_rlog = Y ｜ N**

リモート操作ログを取得するかどうかを指定します。

Y：

リモート操作ログを取得します。リモート操作ログを取得することで，リモートシェル実行環境に関するトラブルシュート情報を取得できます。

N：

リモート操作ログを取得しません。

《適用基準》

リモートシェル実行環境に関するトラブルシュート情報を取得するために，このオペランドに Y を指定することを検討してください。

《注意事項》

リモート操作ログは$PDDIR/spool/cmdlog 下に取得されます。

9.2.8 全面回復処理に関するオペランド

◆ **pd_max_recover_process = 全面回復処理の並列実行プロセス数**
～＜符号なし整数＞((1～10))《3》

全面回復処理をするときの回復処理プロセス（REDO プロセス）数を指定します。HiRDB/パラレルサーバの場合は，1 サーバ（ディクショナリサーバ又はバックエンドサーバ）当たりの回復処理プロセス（REDO プロセス）数になります。

《前提条件》

通常ファイルについては，入出力並列処理をしません。

《指定値の目安》

- 見積もり方法はファイルの種類によって異なります。

条件		指定値の目安
キャラクタ型スペシャルファイル	RD エリアを定義したキャラクタ型スペシャルファイル数が 10 以上のとき（1 サーバ当たり）	10

条件		指定値の目安
	RD エリアを定義したキャラクタ型スペシャルファイル数が 10 未満のとき（1 サーバ当たり）	キャラクタ型スペシャルファイル数
通常ファイル		1

- このオペランドの値を大きくすると，全面回復処理時の入出力並列度が増すため，回復処理時間を短縮できます。ただし，「このオペランドの値×サーバ数」分のプロセスが起動されるため，前記の指定値の目安と HiRDB のリソースを考慮して指定値を決めてください。

◆ **pd_redo_allpage_put = Y ｜ N**

全面回復処理時に，シンクポイント以降に更新されたページをデータベースに書き出すかどうかを指定します。

Y：

全面回復処理時に，シンクポイント以降に更新されたすべてのページを書き出します。

N：

全面回復処理時に，障害発生時にデータベースに書き出していなかったページだけを書き出します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。

LVM（論理ボリューム・マネジャ）のミラーリング機能の使用によってデータベースをミラー化する場合，OS 又はマシンの障害発生時，及びユニット障害による系切り替え発生時に，ミラー正副不一致状態になることがあります。その場合は，次の方法で正副ボリュームを一致させてください。

1. LVM の機能を使用して正副ボリュームを一致させてから HiRDB を再開始してください。
2. 1 の方法で正副ボリュームを一致させる運用ができない場合，又は正副ボリュームを一致させる処理に時間が掛かり，系切り替え時間などのシステム要件を満たせない場合は，このオペランドに Y を指定してください。障害発生後の HiRDB 再開始での全面回復処理で，シンクポイント以降に更新されたすべてのページをデータベースに書き出します。これによって，HiRDB が正副ボリュームを一致させてミラー正副不一致状態を解消します。

ミラー正副不一致状態が発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

- このオペランドに Y を指定した場合，全面回復処理時にすべてのページをデータベースに書き出すためのオーバヘッドが掛かります。
- このオペランドを指定した場合，全面回復処理終了時に KFPH24004-I メッセージを出力します。このオペランドを省略した場合は出力しません。

9.2.9　トランザクション決着処理に関するオペランド

◆ **pd_trn_rerun_branch_auto_decide = Y ｜ N**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

トランザクションのコミットメント制御で一相目のプリペア処理が完了する前にユニットが異常終了した場合，ユニットの再開始時にこのトランザクションからブランチした未決着状態のトランザクションを自動決着するかどうかを指定します。

Y：

未決着状態のトランザクションを自動決着します。

N：

未決着状態のトランザクションを自動決着しません。この場合，HiRDB 管理者が未決着状態のトランザクションを決着させる必要があります。未決着状態のトランザクションを決着させる方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

Y を指定した場合，又はこのオペランドを省略した場合の注意事項を次に示します。

- 縮退起動機能を使用している場合，縮退起動時に開始していなかったユニットのトランザクションは自動決着の対象とはなりません。したがって，開始していなかったユニットを障害復旧後などに再開始した場合は，未決着状態のトランザクションがないかどうかを確認してください。未決着状態のトランザクションがある場合は，未決着状態のトランザクションを決着させてください。未決着状態のトランザクションの確認方法，及び未決着状態のトランザクションを決着させる方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- ブランチ元のトランザクションがあるサーバのシステムログ量が増加します。
- 未決着状態のトランザクションの決着処理を実行している間，ブランチ元のトランザクションがあるサーバのシステムログが上書き禁止状態になります。

◆ **pd_trn_send_decision_intval_sec = 秒単位でのトランザクション自動決着の送信リトライ時間間隔～＜符号なし整数＞～((0～65535))《15》（単位：秒）**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ブランチしたトランザクションに自動決着指示を送信した場合，通信障害などによって送信が失敗したときに決着指示を再送するまでの時間を秒単位で指定します。

《前提条件》

pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している必要があります。

《注意事項》

0 を指定した場合は決着指示を連続して再送するため，通信負荷が高くなります。

《ほかのオペランドとの関連》

このオペランドは，pd_trn_send_decision_interval オペランドと次に示す関係があります。

- 再送するまでの時間を秒単位で指定する場合は pd_trn_send_decision_intval_sec オペランドを，分単位で指定する場合は pd_trn_send_decision_interval オペランドを指定します。
- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定すると，pd_trn_send_decision_intval_sec オペランドの指定値が優先されます。

◆ **pd_trn_send_decision_interval = 分単位でのトランザクション自動決着の送信リトライ時間間隔～<符号なし整数>((0～65535))（単位：分）**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ブランチしたトランザクションに自動決着指示を送信した場合，通信障害などによって送信が失敗したときに決着指示を再送するまでの時間を分単位で指定します。通常は pd_trn_send_decision_intval_sec オペランドを指定し，このオペランドは省略してください。

《前提条件》

pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している必要があります。

《注意事項》

0 を指定した場合は決着指示を連続して再送するため，通信負荷が高くなります。

《ほかのオペランドとの関連》

このオペランドは pd_trn_send_decision_intval_sec オペランドと次に示す関係があります。

- 再送するまでの時間を秒単位で指定する場合は pd_trn_send_decision_intval_sec オペランドを，分単位で指定する場合は pd_trn_send_decision_interval オペランドを指定します。
- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定すると，pd_trn_send_decision_intval_sec オペランドの指定値が優先されます。
- pd_trn_send_decision_intval_sec オペランドと pd_trn_send_decision_interval オペランドを両方とも指定しないと，pd_trn_send_decision_intval_sec オペランドの省略値が仮定されます。

◆ **pd_trn_send_decision_retry_time = トランザクション自動決着の最大待ち時間～<符号なし整数>((0～65535))《360》（単位：分）**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ブランチしたトランザクションに自動決着指示を送信してから決着完了の応答があるまでの最大待ち時間を指定します。ここで指定した時間を過ぎても決着完了の応答がない場合，通信障害などの回復不能な障害が発生したとみなし，ブランチしたトランザクションへの決着指示を中止して，ブランチ元のトランザクションを決着させます。

なお，このオペランドに 0 を指定した場合は時間監視をしません。

《前提条件》

pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している必要があります。

◆ **pd_trn_watch_time = トランザクション同期点処理時の最大通信待ち時間～<符号なし整数>((0，300～65535))《3600》（単位：秒）**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB サーバプロセス上で実行されるトランザクションの同期点処理で，トランザクションブランチ間で行う通信（プリペア及びコミットの要求又は完了通信）の受信待ち時間の最大値を指定します。指定時間を過ぎても要求又は完了通信がない場合，該当するトランザクションブランチが 2 相コミットの 1 相目完了前であればロールバックします。1 相目完了後であれば，指示された処理を実行してトランザクションを決着します。トランザクション間で行う通信については，マニュアル「HiRDB 解説」の「コミットとロールバック」を参照してください。

《利点》

HiRDB クライアントがトランザクション決着指示を中断（クライアントプロセスの強制終了など）しても，HiRDB サーバがこのトランザクションの実行を中止しないと，トランザクションを実行し続けます。このため，データベースの排他資源などを長時間占有することがあります。このオペランドを指定すると，排他資源などの占有時間を短縮できます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。次に示す場合にこのオペランドを指定してください。

- コミット時に KFPA11989-E 又は KFPA11722-E メッセージが出力される場合
- 該当トランザクションでデータベースの更新件数が少ないのに COMMIT 文の処理時間が掛かる場合

《オペランドの規則》

- 0 を指定すると，プリペア及びコミットの要求又は完了を受信するまで要求又は完了通信を待ち続けます。
- 1～299 を指定した場合は 300 に切り上げられます。

《注意事項》

- 2 相コミットの 2 相目で行うコミットの要求又は完了通信に対しては，pd_dbsync_point=commit を指定したときだけ，このオペランドの指定値が有効になります。

◆ pd_trn_rcvmsg_store_buflen = トランザクション回復メッセージキューサイズ

～＜符号なし整数＞((4096～25600000))《8192》（単位：バイト）

- 0904 互換モードの場合：**((4096～25600000))《4096》**

HiRDB は，トランザクションの回復処理を行うとき，回復対象となるトランザクションをトランザクション回復メッセージキューに登録して管理します。このオペランドでは，トランザクション回復メッセージキューサイズを指定します。

《指定値の目安》

- 通常は，このオペランドを指定する必要はありません。
- HiRDB 稼働中に KFPS00854-W メッセージ（server=_trnrcv）が出力された場合に，このオペランドの指定を検討してください。指定値の見積もり式を次に示します。なお，見積もりの結果が省略値以下だった場合は，省略値を指定してください。

  ・HiRDB/シングルサーバの場合

  ↑72×A÷1024↑×1024〔バイト〕

・HiRDB/パラレルサーバの場合

↑72×（B ＋ C ＋ D ＋ E）÷1024↑×1024〔バイト〕

A：（pd_max_users オペランドの値＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7

B：

ユニット内に FES がある場合：b×2 ＋ 7

b：

マルチ FES の場合：pd_max_users オペランドの値＋ pd_max_reflect_process_count オペランドの値＋ 1

マルチ FES でない場合：pd_max_users オペランドの値＋ pd_max_reflect_process_count オペランドの値

ユニット内に FES がない場合：0

C：

ユニット内に BES がある場合（BES が複数ある場合は，それぞれの BES で求めた値を加算します）：

（MAX（pd_max_bes_process オペランドの値，pd_max_users オペランドの値）＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7

ユニット内に BES がない場合：0

D：

ユニット内に DS がある場合：

（MAX（pd_max_dic_process オペランドの値，pd_max_users オペランドの値）＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7

ユニット内に DS がない場合：0

E：

影響分散スタンバイレス型系切り替えを適用している場合：

（（MAX（ゲスト BES の中で最大の pd_max_bes_process オペランドの値，pd_max_users オペランドの値）＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7）× pd_ha_max_act_guest_servers オペランドの値

影響分散スタンバイレス型系切り替えを適用していない場合：0

《注意事項》

- このオペランドに省略値より大きい値を指定した場合，ユニットコントローラ用共用メモリが増加します。詳細は，マニュアル「HiRDB システム導入・設計ガイド」の「ユニットコントローラが使用する共用メモリの計算式」を参照してください。
- このオペランドに省略値より大きい値を指定した場合，トランザクション回復プロセス（pdtrnrvd）の最大プロセス数が増加します。トランザクション回復プロセス（pdtrnrvd）の最大プロセス数の見積もり式を次に示します。

  ・HiRDB/シングルサーバの場合

MIN（トランザクション回復メッセージキューサイズ÷72，（pd_max_users オペランドの値＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7）

・HiRDB/パラレルサーバの場合

MIN（トランザクション回復メッセージキューサイズ÷72，A ＋ B ＋ C ＋ D）

A：

ユニット内に FES がある場合：a×2 ＋ 7

a：

マルチ FES の場合：pd_max_users オペランドの値＋ pd_max_reflect_process_count オペランドの値＋ 1

マルチ FES でない場合：pd_max_users オペランドの値＋ pd_max_reflect_process_count オペランドの値

ユニット内に FES がない場合：0

B：

ユニット内に BES がある場合：（MAX（pd_max_bes_process オペランドの値，pd_max_users オペランドの値）＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7

ユニット内に BES がない場合：0

C：

ユニット内に DS がある場合：（MAX（pd_max_dic_process オペランドの値，pd_max_users オペランドの値）＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7

ユニット内に DS がない場合：0

D：

影響分散スタンバイレス型系切り替えを適用している場合：

（(MAX（ゲスト BES の中で最大の pd_max_bes_process オペランドの値，pd_max_users オペランドの値）＋ pd_max_reflect_process_count オペランドの値）×2 ＋ 7）× pd_ha_max_act_guest_servers オペランドの値

影響分散スタンバイレス型系切り替えを適用していない場合：0

《ほかのオペランドとの関連》

このオペランドは pd_max_server_process オペランドと関連があります。

◆ **pd_trn_commit_optimize = ONEPHASE ｜ NOUSE**

このオペランドは，HiRDB/パラレルサーバ限定のオペランドです。

HiRDB/パラレルサーバのコミットメント制御で一相コミットを使用するかどうかを指定します。一相コミットについては，マニュアル「HiRDB 解説」を参照してください。

ONEPHASE：

トランザクション内の更新ブランチ数が一つの場合（一つのトランザクションが更新するサーバ数が一つの場合），コミットメント制御に**一相コミット**を使用します。なお，コミットメント制御で一相コミットを行うことを**一相最適化**といいます。

NOUSE：

コミットメント制御に**二相コミット**を使用します。一相コミットは使用しません。

《指定値の目安》

通常は，ONEPHASE を指定するか，又はこのオペランドを省略してください。

《注意事項》

- OLTP システムから二相コミットの指示がある場合は，このオペランドの指定は無効になります。
- 回復不要 FES を使用している場合，このオペランドの指定よりも回復不要 FES の制限事項が優先されます。制限事項を次に示します。

  ・フロントエンドサーバ上に出力されるログを抑止します。

  ・回復不要 FES を適用したフロントエンドサーバでは，X/Open XA インタフェースを使用して接続する UAP を実行できません。

◆ pd_trn_rollback_watch_time = ロールバック完了応答の最大待ち時間〔，ロールバック指示再送限界時間〕

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

このオペランドは，pd_trn_rerun_branch_auto_decide オペランドに Y を指定，又は指定を省略している場合に有効になります。

また，クライアント環境定義の PDCWAITTIME に 0 を指定，又は指定を省略している UAP を監視対象とします。

ロールバック完了応答の最大待ち時間：～＜符号なし整数＞((0～65535))《3600》（単位：秒）

ロールバック処理で，フロントエンドサーバがバックエンドサーバ又はディクショナリサーバからロールバック完了応答を受信するまでの，待ち時間の最大値を指定します。指定した時間を過ぎても応答が受信できない場合は，該当するバックエンドサーバ又はディクショナリサーバに対してロールバック指示を再送します。

《指定値の目安》

通常，ロールバック完了応答の最大待ち時間を指定する必要はありません。トランザクションが決着できない事態が発生する可能性を考慮し，システム要件として 3,600 秒よりも早くトランザクションを決着させる必要がある場合は，トランザクション決着まで待てる秒数を指定してください。

《注意事項》

- ロールバック完了応答の最大待ち時間に 0 を指定した場合，ロールバック指示は再送されません。
- ロールバック完了応答の最大待ち時間に 1～9 を指定した場合，自動的に 10 秒に切り上げられます。

ロールバック指示再送限界時間：～＜符号なし整数＞((0～65535))《600》（単位：秒）

ロールバック指示の再送が失敗した場合に，ロールバック指示の再送を繰り返す時間を指定します。

ロールバック完了応答の最大待ち時間を経過しても，バックエンドサーバ又はディクショナリサーバからのロールバック完了応答を受信できなかった場合，フロントエンドサーバは応答がないサーバに対してロールバック指示を再送します。再送に失敗すると，ロールバック指示再送限界時間まで，再送を繰り返します。

なお，ロールバック指示再送限界時間まで再送が成功しなかった場合，KFPS00936-E メッセージを出力し，フロントエンドサーバプロセスを異常終了させ，トランザクションをキャンセルします。

《指定値の目安》

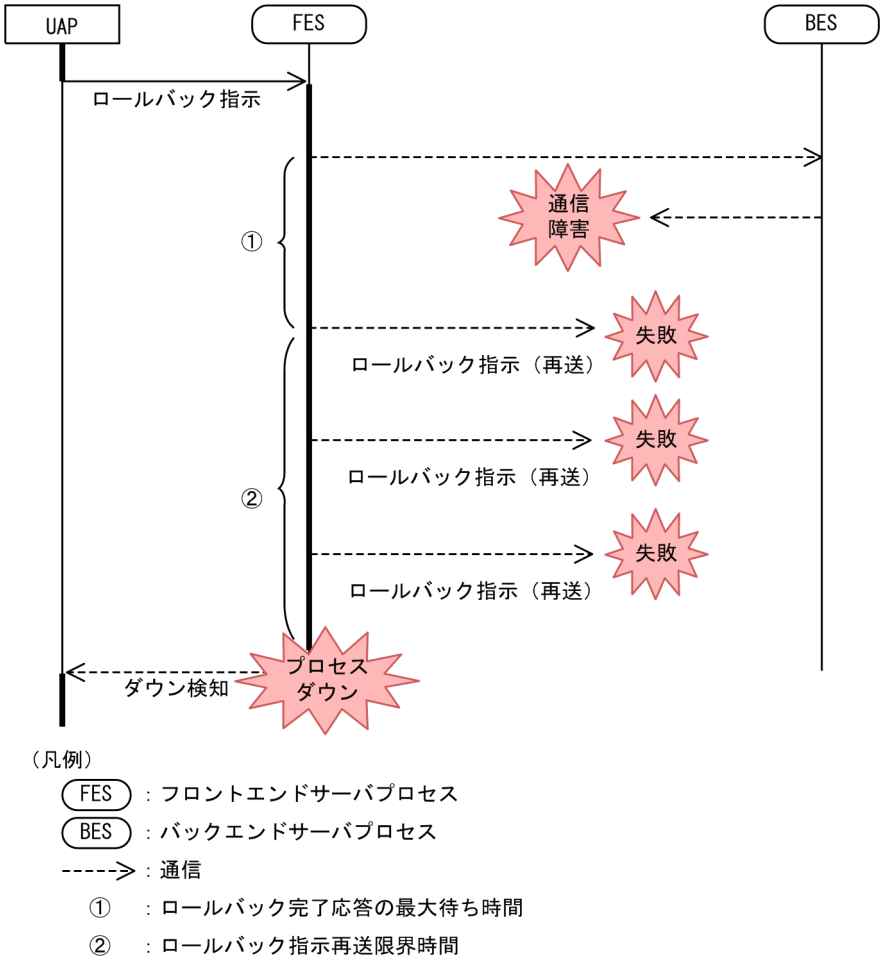
通常，ロールバック指示再送限界時間を指定する必要はありません。ロールバック指示の再送が失敗し続ける可能性を考慮し，システム要件として 600 秒よりも早くトランザクションを決着させる必要がある場合は，トランザクション決着まで待てる秒数を指定してください。

なお，通信障害時に UAP に応答するまでには，「ロールバック完了応答の最大待ち時間＋ロールバック指示再送限界時間」の時間が必要です。この点に留意して，値を設定してください。

《注意事項》

- ロールバック指示再送限界時間に 1～239 を指定した場合，自動的に 240 秒に切り上げられます。

このオペランドの適用範囲を次の図に示します。



9.2.10　SQL の仕様に関するオペランド

◆ pd_overflow_suppress = Y ｜ N

演算中のエラー抑止について指定します。エラー抑止の対象を次に示します。

- 演算途中でのオーバフロー

- 0 除算エラー

Y：

SQL 実行時の演算中に対象となるエラーが発生した場合，演算結果をナル値にして処理を続行します。

N：

SQL 実行時の演算中に対象となるエラーが発生した場合，エラーで処理を取り消します。

◆ **pd_space_level = 0 ｜ 1 ｜ 3**

空白変換機能を使用する場合の空白変換レベルを指定します。空白変換機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

0：空白変換レベル 0 を指定します。

1：空白変換レベル 1 を指定します。

3：空白変換レベル 3 を指定します。

空白変換レベルの処理方式を次に示します。

レベル	説明
レベル 0	空白変換をしません。
レベル 1	操作系 SQL での定数，埋込み変数，又は？パラメタのデータの空白を次のように変換します。 • 文字列定数を各国文字列定数とみなした場合，半角空白 2 文字を全角空白 1 文字に変換します。なお，半角空白が 1 文字単独で現れる場合は変換しません。 • 文字列定数を混在文字列定数とみなした場合，全角空白 1 文字を半角空白 2 文字に変換します。 • 各国文字列型の列へのデータの格納時，及び各国文字列型の値式との比較時は，埋込み変数，又は？パラメタの半角空白 2 文字を全角空白 1 文字に変換します。なお，半角空白が 1 文字単独で現れる場合は変換しません。 • 混在文字列型の列へのデータの格納時，及び混在文字列型の値式との比較時は，埋込み変数，又は？パラメタの全角空白 1 文字を半角空白 2 文字に変換します。
レベル 3	空白変換レベル 1 の処理に加えて次の処理が加わります。 • 各国文字列型の値式のデータを検索するときに，全角空白 1 文字を半角空白 2 文字に変換します。

《注意事項》

- 空白変換レベルを変更した場合，変更の前後で UAP の結果が異なる場合があります。UAP の結果を同じにしたい場合には，空白変換レベルを変更しないでください。
- 空白変換レベル 3 を指定してソートをした場合，HiRDB はソートの結果に対して空白変換をするため，期待した結果を得られないことがあります。
- クラスタキーの列へデータを格納する場合，空白変換によってユニークエラーとなることがあります。この場合，空白変換をしないでデータを格納するか，又はデータベース再編成ユティリティで既存のデータベースの空白を統一してください。
- 各国文字列の空白変換は，先頭から 2 文字単位で変換します。
- 空白変換レベルに 1 又は 3 を指定したときの注意事項を次に示します。

ハッシュ分割した表に対して，表分割ハッシュ関数を使用した UAP で格納先 RD エリアを求める場合は，表分割ハッシュ関数の引数に空白変換レベルを指定してください。指定しないと表分割ハッシュ関数の結果が不正になることがあります。

キーレンジ分割（分割キーが各国文字データ又は混在文字データの場合）した表に対して，UAP でキーレンジ分割処理をしている場合は，分割キー値を空白変換関数で変換してください。変換しないとキーレンジ分割の結果が不正になることがあります。

表分割ハッシュ関数，及び空白変換関数については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSPACELVL オペランドを指定してください。PDSPACELVL オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ **pd_dec_sign_normalize = Y ｜ N**

UAP から渡された符号付きパック形式データの符号部を正規化するかどうかを指定します（**DECIMAL 型の符号正規化機能**を使用するかどうかを指定します）。符号部を正規化するとは，符号付きパック形式データの符号部 A～F を C 又は D に変換することを意味しています。

DECIMAL 型の符号正規化機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

Y：

DECIMAL 型の符号正規化機能を使用します。符号付きパック形式データの符号部を正規化します。A～F を DECIMAL データの符号部とみなします。

N：

DECIMAL 型の符号正規化機能を使用しません。符号付きパック形式データの符号部を正規化しません。C，D，F を DECIMAL データの符号部とみなします。

《注意事項》

Y を指定した場合（DECIMAL 型の符号正規化機能を使用する場合）の注意事項を次に示します。

- ハッシュ分割した表に対して，表分割ハッシュ関数を使用した UAP で格納先 RD エリアを求める場合は，表分割ハッシュ関数の引数に「DECIMAL 型の符号正規化機能あり」を指定してください。指定しないと表分割ハッシュ関数の結果が不正になることがあります。
- キーレンジ分割（分割キーが DECIMAL 型のデータの場合）した表に対して，UAP でキーレンジ分割処理をしている場合は，分割キー値を DECIMAL 型符号正規化関数で変換してください。変換しないとキーレンジ分割の結果が不正になることがあります。

表分割ハッシュ関数，及び DECIMAL 型符号正規化関数については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ **pd_sql_dec_op_maxprec = 演算結果が 29 けた以下の DECIMAL 型となる精度の最大値～<符号なし整数>((29 ｜ 38))**

- 0904 互換モードの場合：**《29》**

次に示す最大値を指定します。

- 演算結果のデータ型が DECIMAL 型となる精度の最大値
  ・29 けた以下のデータだけに対する四則演算
  ・29 けた以下のデータだけに対する集合関数 AVG/SUM
  ・29 けた以下のデータだけに対するスカラ関数 DECIMAL/VALUE/GREATEST/LEAST
  ・引数が FLOAT 型でかつ，精度を省略したスカラ関数 DECIMAL
  ・対応する THEN 句及び ELSE 句の値が 29 けた以下のデータだけの CASE 式
  ・抽象データ型のパラメタを含み，戻り値のデータ型が 29 けた以下であるユーザ定義関数だけが「候補となる関数」となる関数の呼び出し
- 集合演算の結果列のデータ型が DECIMAL 型となる場合，集合演算の導出表に対応する列が 29 けた以下のデータだけのときの精度の最大値

《指定値の目安》

- バージョン 09-04 以前の HiRDB をバージョンアップする場合
  バージョンアップ前にこのオペランドに 38 を指定していないときは，このオペランドに 29 を指定するか，0904 互換モードを適用してください。
- 上記以外の場合
  このオペランドはアプリケーションの作りに依存します。
  アプリケーションで使用する 29 けた以下のデータに対する演算結果の DECIMAL 型の精度を 29 けたで扱いたい場合は 29 にしてください。
  そうでない場合は，38 を指定するか，このオペランドを省略してください。

《注意事項》

このオペランドを変更した場合，DECIMAL 型を使用するアプリケーションに影響が出ます。

次に示す値式の結果のデータや問合せ指定の結果を取得するアプリケーションは修正が必要になります。また，ビュー表，トリガ，ルーチン，又は検査制約を持つ表に，次の値式又は問合せ指定を含む場合，実行時に DECIMAL 型の演算がオーバフローしたり，ビュー表の検索がエラーとなったりするおそれがあるため，ビュー表，トリガ，ルーチン，又は検査制約を持つ表を再作成してください。

なお，トリガの再作成は，ALTER TRIGGER，及び ALTER ROUTINE でするのではなく，トリガを一度削除した後，再度作成してください。

- 29 けた以下のデータだけに対する四則演算
- 29 けた以下のデータだけに対する集合関数 AVG，SUM
- 29 けた以下のデータだけに対するスカラ関数 DECIMAL，VALUE，GREATEST，LEAST
- 引数が FLOAT 型でかつ，精度を省略したスカラ関数 DECIMAL
- 対応する THEN 句及び ELSE 句の値が 29 けた以下のデータだけの CASE 式
- 抽象データ型のパラメタを含み，戻り値のデータ型が 29 けた以下であるユーザ定義関数だけが「候補となる関数」となる関数の呼び出し

- 集合演算の結果列のデータ型が DECIMAL となる場合で，集合演算の演算対象導出表の対応する列が 29 けた以下のデータだけの問合せ指定

《各見積もり式への影響》

pd_sql_dec_op_maxprec オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「グループ分け高速化機能実行時に必要なメモリ所要量の求め方」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「グループ分け高速化機能実行時に必要なメモリ所要量の求め方」

◆ pd_sql_mode = 0 | 1

SQL 文中に集合演算子 MINUS を使用するかどうかを指定します。

0：集合演算子 MINUS を使用しません。

1：集合演算子 MINUS を使用します。

《注意事項》

- このオペランドに 1 を指定した場合，MINUS が SQL の予約語になります。そのため，表名，列名などの名前に MINUS を使用している場合は，引用符（"）で囲んだ"MINUS"に変更してください。名前の指定方法については，マニュアル「HiRDB SQL リファレンス」の「名前の指定」を参照してください。
- ルーチンの定義，及びルーチンのリコンパイル時に，集合演算子 MINUS を使用するかどうかは，このオペランドの値によって決定されます。
- ルーチンの定義と，そのルーチンのリコンパイル時はこのオペランドの値を同じにしてください。このオペランドを変更し，ルーチンのリコンパイルを行うと構文エラーになる場合があります。

◆ pd_ddl_tbl_pctfree = 表の未使用領域比率,表のセグメント内空きページ比率

定義系 SQL での PCTFREE(表オプション)の省略値を変更する場合，未使用領域の比率とセグメント内の空きページ比率を指定します。表の属性によって，指定するオペランドが異なります。表の属性と指定するオペランドを次の表に示します。

表 9–1　表の属性と指定するオペランド

表の属性		指定するオペランド
FIX 表	クラスタキーなし	pd_ddl_tbl_fix_pctfree
	クラスタキーあり	pd_ddl_tbl_pctfree
非 FIX 表		pd_ddl_tbl_pctfree

PCTFREE(表オプション)については，マニュアル「HiRDB SQL リファレンス」の「CREATE TABLE（表定義）」を参照してください。

《オペランドの指定値》

- 表の未使用領域の比率 ～<符号なし整数> ((0～99))（単位：%）
- 表のセグメント内の空きページ比率 ～<符号なし整数> ((0～50))（単位：%）

《オペランドの省略値》

このオペランドを省略した場合に仮定する値を次の表に示します。

表 9–2　オペランドを省略した場合に仮定する値

オペランド	仮定値
pd_ddl_tbl_pctfree	30,10
pd_ddl_tbl_fix_pctfree	0,0

《利点》

定義するすべての表に対して，定義系 SQL の省略値とは異なる PCTFREE(表オプション)の値に統一したい場合，定義系 SQL ごとに PCTFREE を指定する必要がなくなります。

《注意事項》

- 定義系 SQL の実行時動作を変更する指定のため，定義済みの表には影響しません。
- クラスタキーのない FIX 表については，空き領域を作るメリットはないため，通常は pd_ddl_tbl_fix_pctfree オペランドを指定しないでください。

◆ **pd_ddl_tbl_fix_pctfree = 表の未使用領域比率,表のセグメント内空きページ比率**

このオペランドの説明については，pd_ddl_tbl_pctfree オペランドを参照してください。

◆ **pd_ddl_idx_pctfree = インデクスの未使用領域比率　～<符号なし整数> ((0～99))《30》（単位：%）**

定義系 SQL での PCTFREE(インデクスオプション)の省略値を変更する場合，未使用領域の比率を指定します。PCTFREE(インデクスオプション)については，マニュアル「HiRDB SQL リファレンス」の「CREATE INDEX　形式 1（インデクス定義）」を参照してください。

《利点》

定義するすべてのインデクスに対して定義系 SQL の省略値とは異なる PCTFREE(インデクスオプション)の値に統一したい場合，定義系 SQL ごとに PCTFREE を指定する必要がなくなります。

《注意事項》

定義系 SQL の実行時動作を変更する指定のため，定義済みの表やインデクスには影響しません。

◆ **pd_ddl_tbl_recovery = {ALL ｜ PARTIAL ｜ NO}**

定義系 SQL での列回復制約の省略値を変更する場合，列回復制約を指定します。列回復制約については，マニュアル「HiRDB SQL リファレンス」の「CREATE TABLE（表定義）」を参照してください。

《利点》

定義するすべての表に対して定義系 SQL の省略値とは異なる回復制約の値に統一したい場合，定義系 SQL ごとに RECOVERY を指定する必要がなくなります。

《注意事項》

定義系 SQL の実行時動作を変更する指定のため，定義済みの表には影響しません。

9.2.11　SQL の最適化に関するオペランド

◆ **pd_optimize_level = SQL 最適化オプション〔，SQL 最適化オプション〕...**

～＜識別子又は符号なし整数＞

SQL 最適化オプションを指定します。

SQL 最適化オプションの指定値及びその内容については，マニュアル「HiRDB UAP 開発ガイド」の「PDSQLOPTLVL」を参照してください。

《オペランドの省略値》

このオペランドを省略すると次に示す値が仮定されます。

- **HiRDB/シングルサーバの場合**

  "PRIOR_NEST_JOIN","PRIOR_OR_INDEXES","DETER_AND_INDEXES","RAPID_GROUPING", "DETER_WORK_TABLE_FOR_UPDATE","APPLY_ENHANCED_KEY_COND" ,"MOVE_UP_DERIVED_COND"

- **HiRDB/パラレルサーバの場合**

  "PRIOR_NEST_JOIN","PRIOR_OR_INDEXES","SORT_DATA_BES","DETER_AND_INDEXES", "RAPID_GROUPING","DETER_WORK_TABLE_FOR_UPDATE","APPLY_ENHANCED_KEY_COND" ,"MOVE_UP_DERIVED_COND","FLTS_ONLY_DATA_BES"

0904 互換モードの場合は，次に示す値が仮定されます。

- **HiRDB/シングルサーバの場合**

  "PRIOR_NEST_JOIN","PRIOR_OR_INDEXES","DETER_AND_INDEXES","RAPID_GROUPING", "DETER_WORK_TABLE_FOR_UPDATE","APPLY_ENHANCED_KEY_COND"

- **HiRDB/パラレルサーバの場合**

  "PRIOR_NEST_JOIN","PRIOR_OR_INDEXES","SORT_DATA_BES","DETER_AND_INDEXES", "RAPID_GROUPING","DETER_WORK_TABLE_FOR_UPDATE","APPLY_ENHANCED_KEY_COND"

《オペランドの指定方法》

適用する SQL 最適化オプションを選択して，その識別子又は符号なし整数を指定します。識別子で指定する方法と，符号なし整数（計算値）で指定する方法の二つの方法がありますが，通常は識別子で指定してください。

- **識別子で指定する場合**

  「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用する場合は，次のように指定します。

  pd_optimize_level = "FORCE_NEST_JOIN","SELECT_APSL"

- **符号なし整数で指定する場合**

  「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用する場合は，次のように指定します。

  pd_optimize_level = 4,10

●HiRDB Version 5.0 以前からバージョンアップした場合

HiRDB Version 5.0 以前の合計値指定も有効です。最適化オプションを変更する必要がない場合は，HiRDB Version 6 以降にバージョンアップしたときにこのオペランドの指定値を変更する必要はありません。また，最適化オプションを追加する場合は次のように指定できます。

（例）

HiRDB Version 5.0 で「ネストループジョイン強制」及び「複数の SQL オブジェクト作成」を適用していて，今回「OR の複数インデクス利用の優先」を追加します。

pd_optimize_level = 14,128

ただし，この指定方法はどの機能を適用しているか分かりにくいため，識別子指定に変更することをお勧めします。

《オペランドの規則》

- 識別子と符号なし整数の混在指定はできません。

●識別子指定の場合

- SQL 最適化オプションを引用符（"）で囲んでください。
- ここで説明した SQL 最適化オプションを使用しない場合は"NONE"を指定してください。ただし，NONE と NONE 以外の識別子を指定した場合，NONE の指定は無効になります。
- 識別子は大文字でも小文字でも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回指定した場合と同じになります。

●符号なし整数指定の場合

- ここで説明した SQL 最適化オプションを使用しない場合は 0 を指定してください。ただし，0 と 0 以外の符号なし整数を指定した場合，0 の指定は無効になります。
- 同じ符号なし整数を 2 回以上指定しても，1 回指定した場合と同じになります。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB UAP 開発ガイド」の「PDSQLOPTLVL」を参照してください。

《注意事項》

- SQL 文中に SQL 最適化指定を指定している場合は，このオペランドの指定よりも SQL 最適化指定が優先されます。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。
- ストアドルーチン及びトリガ中の SQL 文（CREATE PROCEDURE，CREATE TYPE，ALTER PROCEDURE，CREATE TRIGGER，ALTER ROUTINE，又は ALTER TRIGGER）に SQL 最適化オプ

ションを指定している場合は，このオペランドの指定よりも SQL 文中の SQL 最適化オプションが優先されます。

《クライアント環境定義との関連》

- このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSQLOPTLVL オペランドを指定してください。PDSQLOPTLVL オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

- フロントエンドサーバ定義に pd_floatable_bes，又は pd_non_floatable_bes オペランドを指定している場合には，「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」及び「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」の指定は無効になります。
- pd_indexlock_mode オペランドに KEY を指定している場合，「更新 SQL の作業表作成抑止」の指定は無効になります。

◆ pd_additional_optimize_level = SQL 拡張最適化オプション〔，SQL 拡張最適化オプション〕... ～＜識別子又は符号なし整数＞

SQL 拡張最適化オプションを指定します。

SQL 拡張最適化オプションの指定値及びその内容については，マニュアル「HiRDB UAP 開発ガイド」の「PDADDITIONALOPTLVL」を参照してください。

《オペランドの省略値》

このオペランドを省略すると次に示す値が仮定されます。

"COST_BASE_2","APPLY_JOIN_COND_FOR_VALUE_EXP","APPLY_SRCH_COND_FOR_VALUE_EXP","MERGE_FROM_DERIVED_TABLE","CONVERT_OUTER_INNER_JOIN"

0904 互換モードの場合は，次に示す値が仮定されます。

"COST_BASE_2"

《オペランドの指定方法》

適用する SQL 拡張最適化オプションを選択して，その識別子又は符号なし整数を指定します。通常は識別子で指定してください。

- **識別子で指定する場合**

  「コストベース最適化モード 2 の適用」及び「ハッシュジョイン，副問合せのハッシュ実行」を適用する場合は，次のように指定します。

  pd_additional_optimize_level = "COST_BASE_2","APPLY_HASH_JOIN"

- **符号なし整数で指定する場合**

  「コストベース最適化モード 2 の適用」及び「ハッシュジョイン，副問合せのハッシュ実行」を適用する場合は，次のように指定します。

  pd_additional_optimize_level = 1,2

《オペランドの規則》

- 識別子と符号なし整数の混在指定はできません。

●識別子指定の場合

- SQL 拡張最適化オプションを引用符（"）で囲んでください。
- ここで説明した SQL 拡張最適化オプションを使用しない場合は"NONE"を指定してください。ただし，NONE と NONE 以外の識別子を指定した場合，NONE の指定は無効になります。
- 識別子は大文字でも小文字でも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回指定した場合と同じになります。

●符号なし整数指定の場合

- ここで説明した SQL 拡張最適化オプションを使用しない場合は 0 を指定してください。ただし，0 と 0 以外の符号なし整数を指定した場合，0 の指定は無効になります。
- 同じ符号なし整数を 2 回以上指定しても，1 回指定した場合と同じになります。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB UAP 開発ガイド」の「PDADDITIONALOPTLVL」を参照してください。

《注意事項》

- SQL 文中に SQL 最適化指定を指定している場合は，このオペランドの指定よりも SQL 最適化指定が優先されます。SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。
- ストアドルーチン及びトリガ中の SQL 文（CREATE PROCEDURE，CREATE TYPE，CREATE TRIGGER，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER）に SQL 拡張最適化オプションを指定している場合は，このオペランドの指定よりも SQL 文中の SQL 拡張最適化オプションが優先されます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDADDITIONALOPTLVL オペランドを指定してください。
PDADDITIONALOPTLVL オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ pd_hash_table_size = ハッシュ表サイズ

～＜符号なし整数＞《256》（単位：キロバイト）

- 32 ビットモードの場合：**((128～524288))**
- 64 ビットモードの場合：**((128～2097152))**

SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を指定した場合のハッシュ表サイズを指定します。

《指定値の目安》

このオペランドに指定するハッシュ表サイズについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《オペランドの規則》

このオペランドには 128 の倍数を指定してください。128 の倍数を指定しなかった場合は自動的に 128 の倍数に切り上げられます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDHASHTBLSIZE オペランドを指定してください。PDHASHTBLSIZE オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《各見積もり式への影響》

pd_hash_table_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ハッシュジョイン及び副問合せのハッシュ実行時に必要なメモリ所要量の求め方」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ハッシュジョイン及び副問合せのハッシュ実行時に必要なメモリ所要量の求め方」

◆ pd_work_table_option = 作業表処理オプション

～＜符号なし整数＞《1》

作業表を使用する SQL を実行するときの HiRDB の処理方式を指定します。このオペランドの指定値は次に示す計算式で求めてください。

作業表処理オプション＝ａ＋ｂ

作業表の処理区分	変数名	値	HiRDB の処理方式
AND の複数インデクス利用時の排他取得方式※1	a	0	アクセスパスとして AND の複数インデクス利用が選択されると，探索条件中で探索に使用するインデクスが定義された列を含む述語のうち一つでも真となるデータがある場合，そのデータに対して共用モード（PR）で排他が掛かります。 このため，排他オプション※2 に WITH EXCLUSIVE LOCK を指定しても，排他モード（EX）ではなく，共用モード（PR）で排他が掛かります。
		1	アクセスパスとして AND の複数インデクス利用が選択されると，探索処理中には，探索条件の中で探索に使用するインデクスが定義された列を含む述語のうち一つでも真となるデータがある場合，そのデータに対して指定したモードで排他が掛かります。ただし，１件目のデータの探索完了までは，探索に使用したインデクスが定義された列に対する述語を論理演算し，偽となったデータについては排他が解除されます。
作業表用バッファの自動増分時のメッセージ出力抑止※3	b	0	作業表用バッファの自動増分時，KFPH29008-I メッセージを出力します。このメッセージは各サーバプロセスの初回の作業表用バッファの自動増分時に出力されます。
		8	作業表用バッファの自動増分時，KFPH29008-I メッセージを出力しません。

注※1

AND の複数インデクス利用をするときの HiRDB の処理方式を指定します。

AND の複数インデクス利用とは，探索条件に AND で結ばれた条件が複数あり，それぞれの列に異なるインデクスが定義してある場合（例えば，SELECT ROW FROM T1 WHERE C1 = 100 AND C2 = 200），それぞれのインデクスを使って条件を満たす行の作業表を作成し，これらの積集合を求める方式です。

注※2

SQL に排他オプションを指定していない場合は，通常 WITH SHARE LOCK が仮定されます。ただし，次に示す指定によっては仮定される排他オプションが変わるため，注意してください。

- カーソルへの FOR UPDATE 句の指定
- システム定義の pd_isolation_level オペランドの指定
- クライアント環境定義の PDISLLVL オペランドの指定
- 手続き，関数，又はトリガの CREATE PROCEDURE，CREATE TRIGGER，CREATE TYPE，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER のデータ保証レベルの指定

注※3

pd_work_buff_expand_limit オペランドを指定していることが前提条件となります。作業表用バッファの自動増分については，pd_work_buff_expand_limit オペランドを参照してください。

《指定値の目安》

作業表の処理区分	値	指定値の目安
AND の複数インデクス利用時の排他取得方式	0	0 を指定すると排他を解除する処理が実行されない分，SQL の処理時間が短縮されます。ただし，複数のユーザが同時に同一表を，AND の複数インデクス利用で更新をするとデッドロックになることがあります。デッドロックを避けるには 1 を指定してください。
	1	1 を指定すると排他を解除する処理が実行される分，SQL の処理時間が増加します。しかし，最終的な排他範囲は狭くなるので，更新 UAP の同時実行性が向上します。
作業表用バッファの増分確保メッセージの出力抑止	0	0 を指定すると，作業表用バッファの増分確保がされたかどうかを監視できます。
	8	8 を指定すると，メッセージの出力量を削減できます。

《注意事項》

作業表の処理区分に AND の複数インデクス利用時の排他取得方式を選択したときの注意事項を次に示します。

- 手続き，関数，及びトリガに対しては，ストアドルーチン及びトリガ生成時（CREATE PROCEDURE，CREATE TRIGGER，CREATE TYPE，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER の実行時）の AND の複数インデクス利用時の排他取得方式が適用されます。ストアドルーチン実行時（CALL 文の実行時）及びトリガ実行時の AND の複数インデクス利用時の排他取得方式は適用されません。

- 次に示すオペランド又はオプションで AND の複数インデクス利用が抑止される場合，このオペランドの指定は無効になります。

  ・pd_optimize_level オペランド

  ・クライアント環境定義の PDSQLOPTLVL オペランド

  ・CREATE PROCEDURE，CREATE TRIGGER，CREATE TYPE，ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER の SQL 最適化オプション

◆ pd_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕...

～＜識別子＞((1～8 文字))

フロータブルサーバとして使用するバックエンドサーバを指定します。

ただし，指定したバックエンドサーバをすべてフロータブルサーバとして使用するとは限りません。

通常はデータ取り出しに使用するサーバは，フロータブルサーバとして使用しませんが，このオペランドを指定した場合はデータ取り出しに使用するサーバもフロータブルサーバの候補とします。

《オペランドの規則》

- 定義されていないバックエンドサーバ名を指定した場合は無視します。
- すべてが定義されていないバックエンドサーバ名の場合は，このオペランドを無視します。

《ほかのオペランドとの関連》

- pd_non_floatable_bes オペランドを同時に指定している場合，pd_non_floatable_bes オペランドを無効とします。
- pd_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」は適用されません。
- pd_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」は適用されません。

◆ pd_non_floatable_bes = "バックエンドサーバ名"〔，"バックエンドサーバ名"〕...

～＜識別子＞((1～8 文字))

フロータブルサーバとして使用しないバックエンドサーバを指定します。

すべてのバックエンドサーバを指定している場合は，このオペランドを無効とします。

《オペランドの規則》

定義されていないバックエンドサーバ名を指定した場合は無視します。

《ほかのオペランドとの関連》

- pd_floatable_bes オペランドを同時に指定している場合，このオペランドを無効とします。
- pd_non_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象拡大（データ取り出しバックエンドサーバ）」は適用されません。
- pd_non_floatable_bes オペランドを指定すると，pd_optimize_level オペランドの「フロータブルサーバ対象限定（データ取り出しバックエンドサーバ）」は適用されません。

9.2.12　拡張 SQL エラー情報出力機能に関するオペランド

◆ **pd_uap_exerror_log_use = AUTO ｜ YES ｜ NO**

拡張 SQL エラー情報出力機能の適用を AUTO，YES，又は NO で指定します。拡張 SQL エラー情報出力機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

AUTO：

拡張 SQL エラー情報出力機能を適用します。クライアントエラーログファイル及び SQL エラーレポートファイルに SQL エラー情報を出力します。

SQL エラーレポートファイルのエラー情報を出力した日時，SQL の開始，終了，実行時間の単位を秒単位で表示します。

YES：

拡張 SQL エラー情報出力機能を適用します。クライアントエラーログファイル及び SQL エラーレポートファイルに SQL エラー情報を出力します。

SQL エラーレポートファイルのエラー情報を出力した日時，SQL の開始，終了，実行時間の単位をマイクロ秒単位で表示します。

NO：

拡張 SQL エラー情報出力機能を適用しません。

《指定値の目安》

SQL エラー情報を一元管理する場合，又はエラーとなった SQL 文，パラメタ情報を出力する場合，YES を指定することを推奨します。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGUSE オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGUSE の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGUSE が優先されます。

PDUAPEXERLOGUSE オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《各見積もり式への影響》

pd_uap_exerror_log_use オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「拡張 SQL エラー情報出力機能使用時に必要なメモリ所要量の求め方」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「拡張 SQL エラー情報出力機能使用時に必要なメモリ所要量の求め方」

◆ **pd_uap_exerror_log_dir = SQL エラーレポートファイルの格納ディレクトリ**

～＜255 文字以内のパス名＞《$PDDIR/spool/pduaperr》

SQL エラーレポートファイルの格納ディレクトリを絶対パス名で指定します。

指定したディレクトリに，二つの SQL エラーレポートファイルが作成されます。ファイル名は pduaperrlog1 と pduaperrlog2 です。

拡張 SQL エラー情報出力機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ **pd_uap_exerror_log_size = SQL エラーレポートファイルの最大サイズ**

～＜符号なし整数＞((2048～2147483647))《10000000》（単位：バイト）

SQL エラーレポートファイルの最大サイズを指定します。このオペランドで指定した値は，二つ作成される SQL エラーレポートファイルのうち一つ分の容量になります。SQL エラーレポートファイルへの出力容量がこのオペランドの値を超えると，出力先をもう片方のファイルに切り替えます。これを繰り返して二つのファイルを交互に使用します。1 回の処理で出力する SQL エラー情報のサイズがこのオペランドの値を超える場合，先頭からこの指定値−1 バイトまで（このオペランドに 1000000 を指定した場合 999999）の SQL エラー情報に#（シャープ 1 文字）を付けて出力します。

《指定値の目安》

このオペランドの指定値は，残しておきたい SQL エラー情報件数を考慮して決定してください。計算式を次に示します。

（A+B）×保持したい件数

- **A ＝ 1100+SQL 文のサイズ（単位：バイト）**

  パラメタ情報の出力サイズを除いた SQL エラー情報の 1 件当たりのサイズです。SQL 文に注釈（コメント）や SQL 最適化指定を記述している場合は，注釈（コメント）及び SQL 最適化指定分のサイズも，SQL 文のサイズに含めます。注釈及び SQL 最適化指定については，マニュアル「HiRDB SQL リファレンス」を参照してください。

- **B ＝（↑pd_uap_exerror_log_param_size オペランドの値÷16↑+1）×89×パラメタの個数**

  パラメタ情報の出力サイズです。

◆ **pd_uap_exerror_log_param_size = クライアントエラーログファイル及び SQL エラーレポートファイルへ出力するパラメタ情報の最大データ長**

～＜符号なし整数＞((0～32008))《0》（単位：バイト）

クライアントエラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の最大データ長を指定します。

- **1 以上を指定した場合**

  クライアントエラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力します。

- **0 を指定した場合**

  クライアントエラーログファイル及び SQL エラーレポートファイルにパラメタ情報を出力しません。

可変長文字列型，BLOB 型，及び BINARY 型の場合，データ長の領域もこの指定値に含まれます。

クライアントエラーログファイル及び SQL エラーレポートファイルに出力するパラメタ情報の大きさがこのオペランドの指定値を超えた場合，指定値の長さまでパラメタ情報を出力します。この場合，超過分は切り捨てられます。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDUAPEXERLOGPRMSZ オペランドを指定してください。このオペランドと，クライアント環境定義 PDUAPEXERLOGPRMSZ の両方を指定した場合は，クライアント環境定義 PDUAPEXERLOGPRMSZ が優先されます。

PDUAPEXERLOGPRMSZ オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

9.2.13　SQL 予約語削除機能に関するオペランド

◆ pd_delete_reserved_word_file = SQL 予約語削除ファイル名称 1〔，SQL 予約語削除ファイル名称 2〕...

～＜ 8 文字以下の識別子＞

SQL 予約語削除機能を使用する場合に，このオペランドを指定します。このオペランドには SQL 予約語削除ファイルの名称を指定します。SQL 予約語削除ファイルは，$PDDIR/conf/pdrsvwd 下に作成してください。SQL 予約語削除機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《オペランドの規則》

- SQL 予約語削除ファイルの名称は，先頭がアルファベットの英数字（最大 8 文字）の任意の名称にしてください。
- SQL 予約語削除ファイル名称は最大 16 個指定できます。

《SQL 予約語削除ファイルの形式》

予約語〔,予約語...〕

- 予約語を複数指定する場合は，コンマ（,）で区切って指定してください。また，コンマの前後に空白やタブを挿入しないでください。
- 定義の 1 行の長さは最大 80 文字です。80 文字を超える指定の場合は，改行してください。ただし，予約語（キーワード）の途中では改行しないでください。
- 予約語は，大文字でも小文字でも指定できます。
- SQL の予約語については，マニュアル「HiRDB SQL リファレンス」を参照してください。

《注意事項》

予約語を削除した場合，その予約語を使う SQL の機能が使用できなくなります。予約語削除機能で削除できる予約語については，マニュアル「HiRDB SQL リファレンス」を参照してください。

《クライアント環境定義との関連》

SQL 予約語削除ファイルは，クライアント環境定義 PDDELRSVWDFILE で指定してください。PDDELRSVWDFILE オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

9.2.14　SQL からのコマンド実行に関するオペランド

◆ **pd_sql_command_exec_users = 認可識別子〔，認可識別子〕...**

～＜ 30 文字以下の記号名称＞

SQL の CALL COMMAND 文でコマンド及びユティリティを実行できる認可識別子を指定します。認可識別子を引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されます。引用符で囲まない場合は，すべて大文字として扱われます。

《注意事項》

認可識別子に引用符を含める場合は，指定する認可識別子中の引用符の前に「¥」を記述してください。

9.2.15　SQLSTATE の細分化に関するオペランド

◆ **pd_standard_sqlstate = Y ｜ N**

- 0904 互換モードの場合：**《N》**

SQLSTATE の値を詳細に出力するかどうかを指定します。

Y：

SQLSTATE の値を詳細に出力します。

N：

SQLSTATE の値を詳細に出力しません。

出力される SQLSTATE の値については，マニュアル「HiRDB メッセージ」を参照してください。

《注意事項》

次の条件をすべて満たす場合は，N を指定してください。

- 推奨モードを適用している。
- バージョン 09-50 より前の HiRDB をバージョンアップする。
- バージョンアップ前の環境でこのオペランドを省略，又は N を指定している。
- クライアント環境変数 PDSTANDARDSQLSTATE を省略している。
- UAP 中，又はプロシジャ中で SQLSTATE を使用している。

これらの条件をすべて満たす場合で，このオペランドを省略，又は Y を指定する場合，SQLSTATE を参照する UAP，及びプロシジャを，詳細な SQLSTATE の値を参照できるように変更してください。

9.2.16　絞込み検索に関するオペランド

◆ pd_max_list_users = リスト操作可能ユーザ数

～＜符号なし整数＞((0～32767))《0》

リストを操作するすべてのユーザ数を指定します。

《ほかのオペランドとの関連》

このオペランドを指定する場合，同時に pd_max_list_count オペランドも指定してください。

《注意事項》

- このオペランドに必要以上に大きい値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。サーバが使用する共用メモリについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- このオペランド及び pd_max_list_count オペランドの両方に最大値（32767）を指定すると，OS が確保できる共用メモリの上限値を超えるおそれがあります。

《各見積もり式への影響》

pd_max_list_users オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」

◆ pd_max_list_count = 1 ユーザ当たりのリスト作成数

～＜符号なし整数＞((0～32767))《0》

1 ユーザが作成できるリストの最大数を指定します。

《ほかのオペランドとの関連》

このオペランドを指定する場合，同時に pd_max_list_users オペランドも指定してください。

《注意事項》

- このオペランドに必要以上に大きい値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。サーバが使用する共用メモリについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- このオペランド及び pd_max_list_users オペランドの両方に最大値（32767）を指定すると，OS が確保できる共用メモリの上限値を超えるおそれがあります。

《各見積もり式への影響》

pd_max_list_count オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」

- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」

◆ **pd_list_initialize_timing = INITIAL ｜ DEFER ｜ STANDBY**

リストの初期化（削除）時期を指定します。通常，HiRDB を開始（再開始を含む）するときにリストが初期化されます。このオペランドでリストの初期化時期を変更できます。リストの作成数が多い場合に，このオペランドの指定値の変更を検討してください。

リストの初期化時期の変更については，マニュアル「HiRDB システム運用ガイド」を参照してください。

INITIAL：

HiRDB を開始するときにリストを初期化します。全リストの初期化が終了しないと HiRDB を開始しません。

DEFER：

HiRDB を開始するときにリストを初期化しません。ASSIGN LIST 文を実行するときにリストを初期化します。そのため，ASSIGN LIST 文を実行するときに初期化する分のオーバヘッドが掛かります。次に示す場合に DEFER 指定を検討してください。

- HiRDB の開始時間を短縮したい
- 系の切り替え時間を短縮したい（ユーザサーバホットスタンバイ又は高速系切り替え機能を使用している）

STANDBY：

高速系切り替え機能を使用するときに有効になる指定です。高速系切り替え機能を使用しないときに STANDBY を指定しても INITIAL が仮定されます。

待機系 HiRDB を開始するときにリストを初期化します。系の切り替え時にはリストを初期化しません。また，系切り替え後の ASSIGN LIST 文の実行時にもリストを初期化しません。ただし，実行系と待機系の両方に同じリストを作成しておく必要があります。

◆ **pd_list_rowcount_in_rename = Y ｜ N**

リスト名変更時に，変更したリストの行数を SQL 連絡領域の SQLERRD[2]に返却するかどうかを指定します。リストの名前を変更する機能を利用し，かつリスト名変更時にリストの行数を取得する必要がある場合に Y を指定し，そうでなければ N を指定してください。

Y：

リストの行数を返却します。

N：

リストの行数を返却しません。

《注意事項》

このオペランドに Y を指定した場合，HiRDB/パラレルサーバのときは，ASSIGN LIST 文でのリスト作成時に，作成したリストの行数をディクショナリサーバに保存します。そのため，ディクショナリサーバとの通信オーバヘッドが発生し，リスト作成時の処理が遅くなります。HiRDB/シングルサーバの場合は，シングルサーバにリストの行数を保存するため，リスト作成時の処理速度に影響はありません。

9.2.17　システム監視に関するオペランド

◆ pd_utl_exec_time = ユティリティの実行監視時間 ～＜符号なし整数＞((0～35791394))（単位：分）

次に示すユティリティの実行時間を監視する場合，その監視時間を分単位で指定します。

- データベース作成ユティリティ（pdload コマンド）
- データベース再編成ユティリティ（pdrorg コマンド）
- 空きページ解放ユティリティ（pdreclaim コマンド）
- グローバルバッファ常駐化ユティリティ（pdpgbfon コマンド）

このオペランドで指定した監視時間を超えてもユティリティが終了しない場合，実行中のユティリティを強制終了して，無応答原因の調査用に次に示す障害情報を出力します。

取得する障害情報	出力先
• コアファイル • .deb ファイル • .ext.deb ファイル	ユティリティが動作したサーバマシンの$PDDIR/spool/save 下に出力されます。
• pdls -d rpc -a コマンドの実行結果 • pdls -d lck コマンドの実行結果	ユティリティのコマンドを入力したサーバマシンの，pd_tmp_directory オペランドで指定したディレクトリ下に出力されます。pd_tmp_directory オペランドの指定を省略した場合は，環境変数 TMPDIR※に指定したディレクトリ下に出力されます。環境変数 TMPDIR※の指定も省略した場合は，/tmp 下に出力されます。 ただし，HiRDB 管理者がコマンドを実行した場合は，$PDDIR/spool/save 下にも出力されます。ファイル名称は次のとおりです。 **コマンド名 YYYYMMDDHHMMSSpid.txt** • pid はプロセス ID です。 • pdreclaim コマンド又は pdpgbfon コマンドを実行した場合，内部的に HiRDB が pdrorg コマンドを実行するため，ファイル名に付けられるコマンド名が pdrorg になります。

注※

init（OS のプロセス）から起動されたプロセスサーバプロセス（pdprcd）に設定される環境変数

《利点》

夜間バッチジョブなどでユティリティ実行中に何らかの障害（通信障害又はディスク障害など）が発生し，ユティリティが無応答状態になっても，後続のジョブの停滞を防止できます。

《指定値の目安》

- このオペランドは無応答障害の対処を目的とし，長大トランザクションの監視を目的としていません。したがって，このオペランドの指定値にはユティリティ実行時間の実績値の最大値に余裕を持たせた値を指定してください。例えば，データベース作成ユティリティの最大実行時間が 60 分程度で，データベース再編成ユティリティの最大実行時間が 90 分程度の場合，pd_utl_exec_time=120 と余裕を持った値を指定してください。これは，通常 90 分で終了する処理が 30 分過ぎても応答が返ってこないのは無応答障害のおそれがあると判断しています。

- ユティリティの実行時間を監視しない場合，次に示すような障害が発生するとユティリティが無応答状態になることがあります。したがって，0 以外の値を指定してください。
  ・サーバ間の通信障害（一時的な障害を含む）
  ・ディスク障害などによるプロセスの沈み込み

《オペランドの規則》

- このオペランドを省略すると，システム共通定義の pd_cmd_exec_time オペランドの指定値が有効になります。pd_cmd_exec_time オペランドも省略すると 0 が仮定されます。
- このオペランドに 0 を指定した場合は，ユティリティの実行時間を監視しません。
- 各ユティリティの option 制御文の exectime オペランドに監視時間を指定した場合は，exectime オペランドの指定が優先されます。

《ほかのオペランドとの関連》

- このオペランドは pd_cmd_exec_time オペランドと関連があります。

◆ pd_cmd_exec_time = 運用コマンド及びユティリティの実行監視時間 ～＜符号なし整数＞((0～3600))《0》（単位：分）

通信障害又は CPU 利用率の高騰や，ディスク入出力の無反応となったときに，運用コマンドやユティリティが無応答状態になることがあります。無制限に無応答となってしまうことを避けるために，このオペランドを指定します。

このオペランドによって実行時間を監視できる運用コマンド及びユティリティを次の表に示します。このオペランドは，システム共通定義の pd_utl_exec_time オペランドよりも多くの運用コマンド及びユティリティを対象としています。実行時間を監視する場合，監視の開始時と終了時に，次の表に示すメッセージを出力します。また，監視開始メッセージ出力後から監視時間が経過しても運用コマンド又はユティリティが終了しない場合は，タイムアウト検知メッセージを出力後，実行中の運用コマンド又はユティリティを強制終了します。

表 9–3　pd_cmd_exec_time オペランドの対象となる運用コマンド及びユティリティ

項番	運用コマンド，ユティリティ名	監視開始メッセージ	監視終了メッセージ	タイムアウト検知メッセージ	タイムアウト検知アボートコード
1	pdacunlck	KFPD00041-I	KFPD00042-I	KFPD00043-E	Pd00009
2	pdbufls	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
3	pdbufmod	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
4	pdchpathn	KFPD00041-I	KFPD00042-I	KFPD00043-E	Pd00009
5	pdclose	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
6	pdcopy	KFPR26092-I	KFPR26093-I	KFPR26091-E	Pra0001
7	pddbchg	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
8	pddbfrz	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
9	pddbls	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009

項番	運用コマンド，ユティリティ名	監視開始メッセージ	監視終了メッセージ	タイムアウト検知メッセージ	タイムアウト検知アボートコード
10	pddbst	KFPK10452-I	KFPK10453-I	KFPK10021-E	Pu02004
11	pdexp	KFPX28414-I	KFPX28415-I	KFPX28416-E	Pu02004
12	pdfssync	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
13	pdgetcst	KFPN00500-I	KFPN00501-I	KFPN00502-E	Pu02004
14	pdhold	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
15	pdlistls	KFPD00041-I	KFPD00042-I	KFPD00043-E	Pd00009
16	pdload	KFPL00901-I	KFPL00902-I	KFPL24511-E	Pu02004
17	pdmemdb	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
18	pdmod	KFPS05629-I	KFPS05630-I	KFPS05631-E	Pscm271
19	pdopen	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
20	pdorbegin	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
21	pdorchg	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
22	pdparaload	KFPL00901-I	KFPL00902-I	KFPL24511-E	Pu02004
23	pdpgbfon	KFPL00901-I	KFPL00902-I	KFPL24511-E	Pu02004
24	pdrbal	KFPL00901-I	KFPL00902-I	KFPL24512-E	Pu02004
25	pdreclaim	KFPL00901-I	KFPL00902-I	KFPL24511-E	Pu02004
26	pdrels	KFPH27067-I	KFPH27068-I	KFPH27069-E	Phu0009
27	pdrorg	KFPL00901-I	KFPL00902-I	KFPL24511-E	Pu02004
28	pdrstr	KFPR26292-I	KFPR26293-I	KFPR26291-E	Pra0001

このオペランドに指定した時間以内に運用コマンド又はユティリティの処理が終了しなかった場合は，タイムアウト発生を示すメッセージを出力し，次の障害情報を取得後に強制終了します。

項番	取得情報	出力先
1	core ファイル	ユティリティが使用したプロセスがあるサーバの$PDDIR/spool/save下に出力されます。
2	.deb ファイル	
3	.ext.deb ファイル	

《オペランドの規則》

- このオペランドを省略，又は 0 を指定した場合は，運用コマンド及びユティリティの実行時間を監視しません。
- 次に示すユティリティの option 制御文の exectime オペランドに監視時間を指定した場合は，exectime オペランドの指定が優先されます。また，exectime オペランドを省略して，かつシス

テム共通定義の pd_utl_exec_time オペランドに監視時間を指定した場合は，pd_utl_exec_time オペランドの指定が優先されます。

・データベース作成ユティリティ（pdload コマンド）

・データベース再編成ユティリティ（pdrorg コマンド）

・空きページ解放ユティリティ（pdreclaim コマンド）

・グローバルバッファ常駐化ユティリティ（pdpgbfon コマンド）

運用コマンド及び上記以外のユティリティの-W オプションに監視時間を指定した場合は，-W オプションの指定が優先されます。

《指定値の目安》

このオペランドは，運用コマンド及びユティリティが長時間無応答となることを防ぐために指定するものです。したがって，運用上許容できる運用コマンド又はユティリティの実行時間を指定してください。

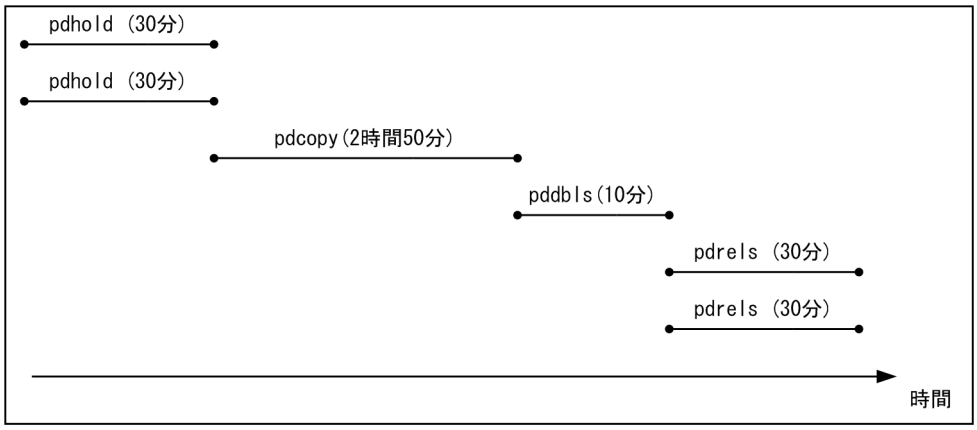
（例 1）

このオペランドで監視できる運用コマンド及びユティリティの中で最大実行時間が 90 分程度であれば，「通常 90 分で終了する処理が更に 30 分過ぎても応答が返ってこないのは無応答障害である。」と判断して，pd_cmd_exec_time に 120 と余裕を持った値を指定してください。

（例 2）

複数の運用コマンド及びユティリティで構成する夜間バッチジョブ全体に許容できる実行時間が 5 時間である場合を例にして説明します。全体方針で，各コマンドを実行するのに許容できる時間が 30 分である場合，pd_cmd_exec_time に 30 を指定します。個別のコマンドで，許容できる実行時間が全体方針と異なる時間で監視したい場合は，コマンドごとの実行監視時間を指定するオプションを使用し，個別に指定する必要があります。例を次の図に示します。

図 9–1　複数の運用コマンド及びユティリティを実行する場合の例



注　括弧内の時間は，各コマンドの実行許容時間です。

この場合は，pdhold と pdrels の 4 コマンドの実行許容時間が 30 分のため，pd_cmd_exec_time に 30 を指定します。pdcopy は，2 時間 50 分まで許容できるため，pdcopy の実行監視時間を指定するオプションに 170 を指定します。また，pddbls は，10 分までしか許容できないため，pddbls コマンドの実行監視時間を指定するオプションに 10 を指定します。このように監視値（30 ＋ 170 ＋ 10 ＋ 30 ＝ 240 ＝ 4 時間）を設定することで，全体で許容できる実行時間の 5 時間に対し，1 時間余裕を持って実行時間を監視します。

《注意事項》

コマンドの監視開始メッセージ出力前，又はコマンドの監視終了メッセージ出力後に，障害によって無応答となった場合，コマンドは自動で終了しないため，コマンドプロセスを手動でキャンセルしてください。

《ほかのオペランドとの関連》

このオペランドは pd_utl_exec_time オペランドと関連があります。

◆ **pd_queue_watch_time = メッセージキュー監視時間**
～<符号なし整数>((0～3600))《600》（単位：秒）

このオペランドは，HiRDB のサーバプロセスの沈み込みを防止することを目的としています。サーバプロセスの沈み込みについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

参考

HiRDB サーバプロセス数は次に示すオペランドで制限されています。

- pd_max_server_process
  ユニット内で稼働するサーバ数が多い場合は，このオペランドの指定値を詳細に見積もってください。また，スタンバイレス型系切り替え機能を使用する場合は，系切り替えが発生したときを想定した見積もりが必要になります。
- pd_max_bes_process
  マルチフロントエンドサーバ，又は 1：1 スタンバイレス型系切り替え機能を使用する場合は，このオペランドの指定値を詳細に見積もってください。
- pd_max_dic_process
  マルチフロントエンドサーバを使用する場合は，このオペランドの指定値を詳細に見積もってください。
- pd_ha_max_server_process
  影響分散スタンバイレス型系切り替え機能を使用する場合は，このオペランドの指定値を詳細に見積もってください。
- pd_max_users
  同時接続数が多い場合は，適切な値を指定する必要があります。

HiRDB では，サーバプロセスの割り当て処理でメッセージキューを使用しています。サーバプロセスの沈み込みが発生すると，メッセージキューからメッセージを取り出せなくなります。メッセージキュー中のメッセージが，このオペランドに指定した時間（**メッセージキュー監視時間**）を過ぎても取り出されない場合，警告メッセージ又はエラーメッセージ（KFPS00888-W 又は KFPS00889-E）を出力します。この機能を**メッセージキュー監視機能**といいます。このメッセージが出力されると，サーバプロセスが沈み込んでいるおそれがあります。

なお，このオペランドに 0 を指定した場合はメッセージキューを監視しません。

メッセージキュー監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《オペランドの規則》

メッセージキューの監視は 10 秒単位で行われるため，このオペランドには 10 秒単位の時間（100 秒，110 秒など）を指定してください。10 秒単位で指定しない場合，下一けたを切り上げます。例えば，105 秒と指定した場合，110 秒が仮定されます。

《ほかのオペランドとの関連》

このオペランドは pd_queue_watch_timeover_action オペランドと関連があります。

◆ **pd_queue_watch_timeover_action = continue ｜ stop**

メッセージキュー監視時間を超えても，メッセージキュー中のメッセージが取り出されない場合の HiRDB の処理を指定します。

continue：

警告メッセージ（KFPS00888-W）を出力します。

stop：

警告メッセージ及びエラーメッセージ（KFPS00888-W 及び KFPS00889-E）を出力して，該当するメッセージキューがあるユニットは異常終了します。

《指定値の目安》

- 通常は stop を指定します（又はこのオペランドを指定しません）。メッセージキュー監視時間を超えても，メッセージキュー中のメッセージが取り出されない場合，HiRDB のサーバプロセスが沈み込んでいるおそれがあります。サーバプロセスの沈み込みが発生したユニットを再開始すると，サーバプロセスの沈み込みを解決できることがあります。
- HiRDB を停止したくない場合は continue を指定します。この場合，サーバプロセスの沈み込みが発生したサーバではトランザクションを実行できません。それ以外のサーバではトランザクションを実行できます。サーバプロセスの沈み込みを解決するには，サーバプロセスの沈み込みが発生したサーバで実行中のトランザクションを pdcancel コマンドなどで終了させてください。その後，サーバプロセスの沈み込み要因を特定して対策してください。また，トランザクションが発生していない場合には，サーバプロセスの沈み込みが発生したサーバを OS の kill コマンドなどで終了させた後，沈み込み要因を特定して対策してください。サーバプロセスの沈み込み要因及び対策については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_cmd_process_conwaittime = 運用コマンド・ユティリティ実行におけるプロセス割り当て監視時間**

～＜符号なし整数＞((0,16～4000))《60》（単位：秒）

運用コマンド及びユティリティ実行時に，プロセスを割り当てるまでの監視時間を指定します。

このオペランドに 0 を指定した場合，プロセスを割り当てるまでの時間を監視しません。

1～15 を指定した場合，プロセスを割り当てるまでの監視時間を 16 秒とします。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

このオペランドは，サーバプロセスの割り当て処理でメッセージキューからメッセージを取り出すサーバプロセス数が不足したことによって KFPS00358-E メッセージが出力され，運用コマンド又はユティリティが異常終了した場合，KFPS00358-E メッセージの対策に従って指定してください。

《注意事項》

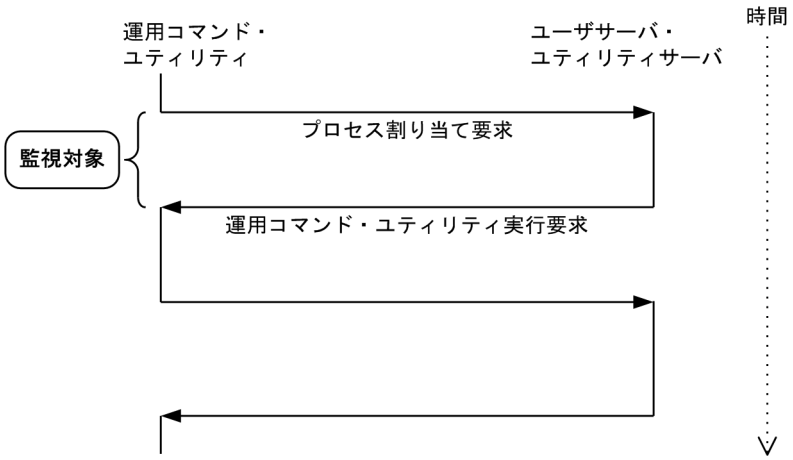
監視時間内にプロセスの割り当てが完了しない場合，KFPS00358-E メッセージを出力し，運用コマンド又はユティリティが異常終了します。

《備考》

運用コマンド及びユティリティは，ユーザサーバプロセス，ユティリティサーバプロセスを割り当ててから，処理を実行します。このオペランドは，ユーザサーバプロセス，ユティリティサーバプロセスを割り当てるまでの時間を監視します。

このオペランドの監視対象を次の図に示します。

図 9–2　このオペランドの監視対象



また，適用する運用コマンド及びユティリティを次の表に示します。

表 9–4　この機能を適用する運用コマンド及びユティリティ

項番	運用コマンド及びユティリティ
1	pdacunlck（連続認証失敗アカウントロック状態の解除）
2	pdclose（RD エリアのクローズ）
3	pdconstck（整合性チェックユティリティ）
4	pdcopy（データベース複写ユティリティ）
5	pddbchg（レプリカ RD エリアのレプリカステータスの切り替え）
6	pddbfrz（ユーザ LOB 用 RD エリアの満杯 HiRDB ファイルの更新凍結）
7	pddbls（RD エリアの状態表示）
8	pddbst（データベース状態解析ユティリティ）
9	pddefrev（定義系 SQL の生成）
10	pdexp（ディクショナリ搬出入ユティリティ）
11	pdbufls（グローバルバッファ情報の表示）
12	pdbufmod（グローバルバッファの動的変更）

項番	運用コマンド及びユティリティ
13	pdgetcst（最適化情報収集ユティリティ）
14	pdhold（RD エリアの閉塞）
15	pdinit（データベース初期設定ユティリティ）
16	pdload（データベース作成ユティリティ）
17	pdmod（データベース構成変更ユティリティ）
18	pdopen（RD エリアのオープン）
19	pdorbegin（オンライン再編成のデータベース静止化）
20	pdorchg（オンライン再編成のカレント RD エリアの切り替え）
21	pdorend（オンライン再編成の追い付き反映）
22	pdparaload（パラレルローディング）
23	pdrbal（リバランスユティリティ）
24	pdrels（RD エリアの閉塞解除）
25	pdrorg（データベース再編成ユティリティ）
26	pdreclaim（空きページ解放ユティリティ）
27	pdpgbfon（グローバルバッファ常駐化ユティリティ）
28	pdrstr（データベース回復ユティリティ）
29	pdvrup（HiRDB バージョンアップ）
30	pdlistls（リスト定義情報の表示）
31	pdmemdb（インメモリデータ処理に関するコマンド）
32	pdreginit（レジストリ機能初期設定ユティリティ）
33	pdchpathn（ファイルパス変更オンラインコマンド）
34	pdrrpref（参照 DB の更新）

◆ pd_down_watch_proc = サーバプロセスの異常終了回数の上限値〔，監視間隔〕

このオペランドは，HiRDB のサーバプロセスの異常終了回数を監視するオペランドです。PDCWAITTIME オーバ，又はアボートによって異常終了するプロセスを監視対象とします。

サーバプロセスの異常終了が多発すると，新たなサービスを受け付けられないことがあります。しかし，サーバプロセスの異常終了では HiRDB は異常終了しないため，実質オンライン停止状態になります。このオペランドを指定すると，HiRDB の再開始によってこの状態を抜け出せます。

サーバプロセスの異常終了回数の上限値：～＜符号なし整数＞((0～65535))《0》

サーバプロセスの異常終了回数がこのオペランドに指定した値を超えた場合，HiRDB（HiRDB/パラレルサーバの場合は該当するユニット）を異常終了させます。これを**プロセスの異常終了回数監視機能**といいます。プロセスの異常終了回数監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

HiRDB/シングルサーバの場合は，シングルサーバプロセスの異常終了回数がカウントされます。
HiRDB/パラレルサーバの場合は，ユニット内のフロントエンドサーバ，バックエンドサーバ，及びディクショナリサーバプロセスの異常終了回数の合計値がカウントされます。

なお，0 を指定すると，サーバプロセスの異常終了回数を監視しません。

監視間隔：～＜符号なし整数＞((10～3600))《600》（単位：秒）

サーバプロセスの異常終了回数を監視する時間を秒単位で指定します。

例えば，100 を指定すると，100 秒間隔でサーバプロセスの異常終了回数を監視します。

《利点》

- HiRDB の再開始によってメモリ及びリソース状態がリフレッシュされて処理効率が向上します。
- サーバプロセスの異常終了が多発した場合，HiRDB が異常終了するため，すぐに系を切り替えられます。

《注意事項》

- 異常終了が許されないシステムの場合は，このオペランドを適用しないでください。
- サーバプロセスが異常終了すると，KFPS01820-E メッセージが出力されます。pdcancel コマンドでサーバプロセスが異常終了したときにも KFPS01820-E メッセージが出力されますが，これは異常終了回数のカウントに含まれません。
- 相互系切り替え構成の場合，系が切り替わると同一サーバマシンで複数の HiRDB を稼働するため，逆にトラフィックが上昇して効果が得られないことがあります。このオペランドを指定する場合は，異常終了した系で HiRDB を再開始する運用をお勧めします。
- HiRDB サーバプロセスが異常終了を繰り返すと，トラブルシュート情報が大量に出力されるため，HiRDB 運用ディレクトリへの入出力が多くなったり，ディスク容量が満杯になったりすることがあります。

《備考》

- プロセスの異常終了回数監視機能によって HiRDB が異常終了する場合，KFPS01821-E 及び KFPS00729-E メッセージを出力します。
- サーバプロセスが異常終了する要因と異常終了回数のカウント対象を次に示します。

サーバプロセスが異常終了する要因	異常終了回数のカウント対象			
	シングルサーバプロセス	フロントエンドサーバプロセス	ディクショナリサーバプロセス	バックエンドサーバプロセス
クライアント環境定義の PDCWAITTIME オペランドの値を超えた	○	○	×※1	×※1
pdcancel コマンド	×	×※2	×	×
内部強制終了（HiRDB が内部的に SIGKILL を発行してプロセスを終了させる場合）	○※3	○※3	×※1	×※1
アボート	○	○	○	○

サーバプロセスが異常終了する要因	異常終了回数のカウント対象			
	シングルサーバプロセス	フロントエンドサーバプロセス	ディクショナリサーバプロセス	バックエンドサーバプロセス
OLTP システムのトランザクション回復処理によるサーバプロセスの異常終了	○	○	×	×
上記以外のプロセス異常終了	○	○	○	○

（凡例）

○：プロセスの異常終了回数にカウントされます。

×：プロセスの異常終了回数にカウントされません。

注※1

トランザクションブランチが発生した状態で障害を検知した場合，同一トランザクションブランチから発生したフロントエンドサーバプロセスの異常終了がカウントの対象になります。

注※2

pdcancel コマンドでバックエンドサーバプロセス又はディクショナリサーバプロセスを強制終了すると，フロントエンドサーバプロセスを内部的に強制終了します。この場合，フロントエンドサーバプロセスの異常終了がカウントされることがあります。

注※3

OLTP システムによるグローバルトランザクションが発生した状態で障害を検知した場合，同一グローバルトランザクションから発生したシングルサーバプロセス又はフロントエンドサーバプロセスの異常終了がカウントの対象になります。

◆ pd_host_watch_interval = ホスト間監視時間間隔

～<符号なし整数>((1～180))《10》（単位：秒）

ほかのホスト（サーバマシン）の稼働状態を監視するために，問い合わせをする時間間隔を秒単位で指定します。

《指定値の目安》

通常，このオペランドは省略してください。

《注意事項》

- このオペランドは，HiRDB/シングルサーバの場合，及び HiRDB/パラレルサーバでも 1 ユニットの場合は，無効になります。
- この値が大き過ぎると，ほかのホストの異常を検出しにくくなります。
- この値が小さ過ぎると，ユニット数の多いシステムの場合，稼働状態を監視するための通信が多量に発生するため，ネットワークに負荷を与えます。

《ほかのオペランドとの関連》

このオペランドは pd_ipc_conn_nblock_time オペランドと関連があります。

◆ pd_watch_pc_client_time = クライアントからの要求間隔監視時間

～<符号なし整数>((0～65535))《3600》（単位：秒）

Windows 版の HiRDB クライアントからの要求に対する応答を HiRDB サーバが返してから，次に HiRDB クライアントからの要求があるまでの最大待ち時間（サーバ側の最大待ち時間）を秒単位で指定します。

指定した時間内に HiRDB クライアントから全く要求がない場合には，クライアントに異常が発生したものとして，強制的にサーバとクライアントとの接続を切り離します。この時に HiRDB クライアントに対しては，接続の切り離しについての通知はしません。

時間監視の対象となるのは，CONNECT から DISCONNECT までの間で，非トランザクション状態（SQL の実行を開始してから COMMIT 又は ROLLBACK までの間以外）の間です。

《注意事項》

- 0 を指定した場合には，HiRDB クライアントからの次の要求を無限に待ち続けます。
- このオペランドの指定値を小さい値（1〜600 程度）にした場合には，HiRDB クライアントが，SQL 実行時にサーバダウンを検知するなど，不当に異常終了することがあります。
- UNIX 版の HiRDB クライアント（Linux 版の HiRDB クライアントも含む）の場合は，このオペランドの指定に関係なく時間を監視しません。UNIX 版の HiRDB クライアントに対して時間監視をする場合は，HiRDB クライアントのクライアント環境定義 PDSWATCHTIME を指定してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDSWATCHTIME オペランドを指定してください。

PDSWATCHTIME オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ pd_spd_syncpoint_skip_limit = シンクポイントダンプ有効化処理のスキップ回数上限値〜＜符号なし整数＞((0，2〜100000))

UAP の無限ループなどが発生すると，シンクポイントダンプの有効化処理ができずにスキップされることがあります。シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

また，上書きできない状態のシステムログファイルが，全システムログファイルの 1/3 以上になったときに HiRDB が異常終了，又は強制終了すると，HiRDB を再開始するときのロールバック処理でシステムログファイルが不足する場合があります。この場合，システムログファイルを新規追加しないと，HiRDB を再開始できません。そして，この再開始処理に要する時間も長くなります。

このオペランドでは，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。シンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，対象トランザクションを強制的に中断してロールバックをします。これを**シンクポイントダンプ有効化のスキップ回数監視機能**といいます。

なお，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，このオペランドの指定値に関係なくシンクポイントダンプ有効化のスキップ回数監視機能は使用できません。

シンクポイントダンプ有効化のスキップ回数監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《利点》

このオペランドを指定すると，UAP の無限ループなどを防止できます。

《指定値の目安》

- シンクポイントダンプ有効化のスキップ回数監視機能を使用する場合は，0 を指定してください。この機能を使用しない場合は，pd_spd_syncpoint_skip_limit オペランドを省略してください。
- 0 を指定すると，スキップ回数の上限値を HiRDB が計算します。0 を指定して不都合が発生した場合，又は KFPS02101-I メッセージが出力された場合に，このオペランドの値を変更してください。そのときの指定値の目安については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- 指定値が大き過ぎると，すべてのシステムログファイルが上書きできない状態になる可能性があります。この場合，HiRDB が異常終了します。
- 指定値が小さ過ぎると，強制的にロールバックされるトランザクションが増える可能性があります。
- pd_log_sdinterval の指定値と，実行時間又はログ出力量が最大のトランザクション実行時のシンクポイント取得回数を考慮して指定してください。

《オペランドの省略値》

このオペランドを省略すると，シンクポイントダンプ有効化のスキップ回数監視機能を使用しません。

《注意事項》

- このオペランドを指定すると，ログレスモードで実行している UAP も監視対象になります。ログレスモードで実行している UAP の処理が中断すると，データベースを自動回復できないため RD エリアを障害閉塞します。このため，上限値の設定に際しては，ログレスモードで実行する UAP のトランザクション処理中に，該当するサーバ内のほかのトランザクションから出力されるシステムログ量も考慮に入れてください。
- 高い負荷が掛かってトランザクションが遅延した場合，遅延したトランザクションが完了するまでシンクポイントダンプを有効化できないため，スキップ回数が増えます。
- スキップ回数が上限値を超えると，トランザクション実行中のプロセスは強制終了されます。このとき，強制終了後にも，そのトランザクションで出力されたログと同等のロールバックログが出力されます。
- pdload，pdmod，pdrorg，pdexp，pddbst，pdgetcst，pdrbal，pdvrup，pdmemdb，pdextfunc，pdconstck，pdreclaim，pdpgbfon，pdparaload，pddefrev，pdobjconv，pdchpathn，pdrrpref，及びプラグインが提供するコマンドはこの機能の監視対象外になります。

◆ pd_dfw_syncpoint_skip_limit = デファードライト処理によるシンクポイントダンプ取得遅延に伴うシンクポイントダンプ有効化処理のスキップ回数上限値

～＜符号なし整数＞((0～100000))《0》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになると，シンクポイントダンプの有効化処理ができずにスキップされます。これは，デファードライト処理に

よってシンクポイントダンプの取得が遅れることで，シンクポイントで出力される更新バッファ数がシンクポイントダンプの取得間隔内で出力できる更新バッファ数を超えるためです。

シンクポイントダンプの有効化処理が連続してスキップされると，上書きできない状態のシステムログファイルが増えます。このため，システムログファイルの容量不足などが発生し，ユニットが異常終了する場合があります。

このオペランドでは，デファードライト処理によってシンクポイントダンプの取得が遅れた場合に，シンクポイントダンプの有効化処理がスキップされる回数（1 トランザクション中のスキップ回数）の上限値を指定します。

デファードライト処理によってシンクポイントダンプの有効化処理がスキップされた回数がこのオペランドで指定した値に達すると，シンクポイントダンプの取得間隔内でシンクポイントダンプの取得が完了するように，HiRDB が更新バッファ数の上限値を決定します。そして，更新バッファ数が上限値を超えた時点で最も古い更新バッファを出力して，シンクポイント時の更新バッファ総量を制限します。これを**更新バッファ量抑制機能**といいます。

《利点》

シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生する，ユニット異常終了を回避できます。

《指定値の目安》

通常はこのオペランドを省略してください。更新処理のスループットを低下させてでも，シンクポイントダンプの取得間隔内でデファードライト処理が終了する前にシンクポイントになることで発生するユニット異常終了を回避したい場合に，1 を指定します。

許容できるシンクポイントダンプの有効化処理のスキップ回数がログ容量などから分かる場合は，その回数を指定します。

《オペランドの規則》

このオペランドに 0 を指定すると，更新バッファ量抑制機能を使用しません。

《注意事項》

更新バッファ量抑制機能の有効期間について，注意事項を次に示します。なお，更新バッファ量抑制機能の有効期間とは，KFPH23035-I メッセージが出力されてから KFPH23036-I メッセージが出力されるまでの間です。

- 更新バッファ数が，HiRDB が決定した更新バッファの上限値を超えている場合，更新処理の実行後に更新バッファを出力するため，更新処理のスループットが低下します。HiRDB が決定する更新バッファの上限値は，次の計算式で求められます。

  （シンクポイントインターバル時間÷WRITE単価※）
  ×（1-（前回のシンクポイントからプレシンクまでのログ出力量÷シンクポイント間のログ出力量）
  ×（バッファプールのバッファ面数÷シンクポイントで更新があったバッファプールの総バッファ面数）

  注※　WRITE 単価についての詳細は，マニュアル「HiRDB システム運用ガイド」を参照してください。

- pd_dbbuff_rate_updpage オペランド，又は pdbuffer -y オペランドによってデファードライトトリガのトリガ契機を指定している場合，それぞれの指定値が，HiRDB が決定した更新バッファの上限値より大きくなると，HiRDB が決定した更新バッファの上限値をデファードライトトリガのトリガ契機となる更新バッファ数に変更します。

  また，pdbuffer -w オペランドの値は，各バッファの更新バッファ上限値まで出力されるように自動調整されます。
- シンクポイントダンプの有効化処理がスキップした場合，シンクポイント時の更新バッファ出力処理中に検知されます。そのため，更新バッファ量抑制機能が有効となる時期は，シンクポイントダンプの有効化処理がスキップされ，エラーメッセージが出力されるときよりも後になる場合があります。
- 通常，並列 write 機能使用時のシンクポイント処理では各デファードライト処理用並列 WRITE プロセスに対する出力要求はシンクポイントごとに 1 回ですが，更新バッファ量抑制機能使用時はシンクポイントスキップの検出を早めるため複数回に分割して行われます。

◆ **pd_watch_resource = MANUAL ｜ AUTO ｜ DEFAULT**

- 0904 互換モードの場合：**《MANUAL》**

リソースの使用率が 80%以上になった場合に，警告メッセージを出力するかどうかを指定します。

MANUAL：警告メッセージを出力しません。

AUTO 又は DEFAULT：警告メッセージを出力します。

監視対象のリソースを次に示します。

監視対象のリソース	出力されるメッセージ
pd_max_users オペランドで指定した最大同時接続数※	KFPS05123-W
pd_max_access_tables オペランドで指定した同時アクセス可能実表数	
pd_max_rdarea_no オペランドで指定した RD エリアの最大数	
pd_max_file_no オペランドで指定した RD エリアを構成する HiRDB ファイルの最大数	
pdwork オペランドで指定した作業表用ファイル用の HiRDB ファイルシステム領域	
pd_max_list_users で指定したリスト作成ユーザ数	
pd_max_list_count で指定した 1 ユーザ当たりのリスト作成数	
スワップ先にできない監査証跡ファイル数	
サーバ内のリスト作成数	KFPH22023-W

注※　DEFAULT の場合は警告メッセージを出力しません。

《ほかのオペランドとの関連》

このオペランドに AUTO 又は DEFAULT を指定すると，次に示すオペランドの省略値に 80 が設定されます。したがって，AUTO 又は DEFAULT を指定すると，次に示すオペランドを各々指定する必要がなくなります。ただし，各オペランドの値を 80 から変更したい場合は，次に示す各オペランドで変更してください。

- pd_max_users_wrn_pnt※1，※2
- pd_max_access_tables_wrn_pnt
- pd_max_rdarea_no_wrn_pnt
- pd_max_file_no_wrn_pnt
- pdwork_wrn_pnt
- pd_max_list_users_wrn_pnt
- pd_max_list_count_wrn_pnt
- pd_aud_file_wrn_pnt※1
- pd_rdarea_list_no_wrn_pnt※1

注※1　警告メッセージの出力済み状態のリセット契機には 50 が設定されます。

注※2　DEFAULT の場合は除きます。

《留意点》

バージョンアップ時に，監視できるリソースの対象が増えることがあります（前記以外のオペランドが増えることがあります）。したがって，常に現状と同じ監視体制にしておきたい場合は，このオペランドを指定しないで，前記のオペランドで項目ごとにリソース監視を実施してください。

◆　pd_max_users_wrn_pnt = HiRDB サーバへの接続数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕

HiRDB サーバへの接続数に関する警告メッセージの出力契機：～＜符号なし整数＞((0～100))《0 又は 80》（単位：%）

HiRDB サーバへの接続数がある一定の比率（pd_max_users オペランドで指定した最大同時接続数に対する比率）以上になったとき，警告メッセージを出力する比率を%単位で指定します。例えば，pd_max_users オペランドに 200 を指定し，このオペランドに 90 を指定すると，HiRDB サーバへの接続数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

- このオペランドに 0 を指定すると，警告メッセージを出力しません。
- pd_max_users オペランドの指定値が 9 以下の場合，このオペランドの指定は無効になります。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

警告メッセージの出力済み状態のリセット契機：～＜符号なし整数＞((0～99))（単位：%）

警告メッセージの出力済み状態のリセット契機を指定します。警告メッセージ（KFPS05123-W）が出力されると，HiRDB は警告メッセージ出力済み状態となります。警告メッセージ出力済み状態となると，HiRDB サーバへの接続数がもう一度警告メッセージの出力契機の値以上になっても警告メッセージが出力されません。しかし，HiRDB サーバへの接続数がここで指定する警告メッセージの出力済み状態のリセット契機以下になると，警告メッセージの出力済み状態が解除されます。

例えば，pd_max_users_wrn_pnt=90,70 と指定すると，HiRDB サーバへの接続数が最大同時接続数の 90%以上になると警告メッセージが出力されます。この後，HiRDB サーバへの接続数が最大同時接続数の 70%以下になるまで警告メッセージは出力されません。一度 70%以下になり，再度 90%以上になったときに警告メッセージが出力されます。

《注意事項》

- この指定を省略すると，「警告メッセージの出力契機−30」の値が仮定されます。その値がマイナスになる場合は，0 が仮定されます。
- 警告メッセージの出力契機より大きい値を指定すると，その指定は無効になり，警告メッセージの出力契機の指定値と同じ値が仮定されます。

◆ **pd_max_access_tables_wrn_pnt = 同時アクセス可能実表数に関する警告メッセージの出力契機～＜符号なし整数＞((0～100))《0 又は 80》（単位：%）**

同時にアクセスする表数と順序数生成子数の合計が，ある一定の比率（pd_max_access_tables オペランドで指定した同時アクセス可能実表数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_access_tables オペランドに 200 を指定し，このオペランドに 90 を指定すると，同時にアクセスする表数と順序数生成子数の合計が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

◆ **pd_max_rdarea_no_wrn_pnt = RD エリア数に関する警告メッセージの出力契機～＜符号なし整数＞((0～100))《0 又は 80》（単位：%）**

RD エリア数が，ある一定の比率（pd_max_rdarea_no オペランドで指定した RD エリアの最大数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_rdarea_no オペランドに 200 を指定し，このオペランドに 90 を指定すると，RD エリア数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

◆ **pd_max_file_no_wrn_pnt = HiRDB ファイル数に関する警告メッセージの出力契機**

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

RD エリアを構成する HiRDB ファイル数が，ある一定の比率（pd_max_file_no オペランドで指定した RD エリアを構成する HiRDB ファイルの最大数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_file_no オペランドに 200 を指定し，このオペランドに 90 を指定すると，RD エリアを構成する HiRDB ファイル数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

◆ **pdwork_wrn_pnt = 作業表用ファイルに関する警告メッセージの出力契機**

～<符号なし整数>((0～100))《0 又は 80》（単位：%）

作業表用ファイル用の HiRDB ファイルシステム領域（pdwork オペランドで指定した作業表用ファイル用の HiRDB ファイルシステム領域）の使用率※が，ある一定の比率以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，このオペランドに 90 を指定すると，作業表用ファイル用の HiRDB ファイルシステム領域の使用率が 90%以上になると警告メッセージ KFPS05123-W が出力されます。

注※

pdfmkfs コマンドで HiRDB ファイルシステム領域を作成するときに指定する次の項目の使用率を示しています。次に示す項目ごとにそれぞれ警告メッセージが出力されます。

- 容量に対する使用率
- 最大ファイル数に対する使用率
- 最大増分回数に対する使用率

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

◆ **pd_max_list_users_wrn_pnt = リスト作成ユーザ数に関する警告メッセージの出力契機～<符号なし整数>((0～100))《0 又は 80》（単位：%）**

リストを使用するユーザ数がある一定の比率（pd_max_list_users オペランドで指定したリスト作成ユーザ数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_list_users オペランドに 200 を指定し，このオペランドに 90 を指定すると，リストを使用するユーザ数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

- このオペランドに 0 を指定すると，警告メッセージを出力しません。
- 警告メッセージは 1 回しか出力されません。ただし，警告メッセージ出力後に対応するサーバを開始し直すと，再度使用率のチェックが実行され，警告メッセージが出力されます。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

◆ **pd_max_list_count_wrn_pnt = 1 ユーザ当たりのリスト作成数に関する警告メッセージの出力契機～<符号なし整数>((0～100))《0 又は 80》（単位：%）**

1 ユーザ当たりのリスト作成数がある一定の比率（pd_max_list_count オペランドで指定した 1 ユーザ当たりのリスト作成数に対する比率）以上になった場合，警告メッセージを出力するときにその比率を%単位で指定します。例えば，pd_max_list_count オペランドに 200 を指定し，このオペランドに 90 を指定すると，あるユーザのリスト作成数が 180 以上になると警告メッセージ KFPS05123-W が出力されます。

《留意点》

- このオペランドに 0 を指定すると，警告メッセージを出力しません。
- 警告メッセージは 1 回しか出力されません。ただし，警告メッセージ出力後に対応するサーバを開始し直すと，再度使用率のチェックが実行され，警告メッセージが出力されます。
- 複数のユーザが同時にリストを作成している場合，タイミングによっては一つのサーバで警告メッセージが 2 回以上出力されることがあります。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

◆ pd_rdarea_list_no_wrn_pnt = サーバ内のリスト作成数に関する警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕

サーバ内のリスト作成数に関する警告メッセージの出力契機：～＜符号なし整数＞((0～100))《0 又は 80》（単位：%）

サーバ内のリスト作成数がリスト作成可能数に対してある一定の比率以上になったとき，警告メッセージを出力する比率を%単位で指定します。例えば，サーバ内のリスト作成可能数が 1000 の場合，このオペランドに 90 を指定すると，リスト作成数が 900 以上になると警告メッセージ KFPH22023-W が出力されます。

《留意点》

このオペランドに 0 を指定すると，警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。
- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

警告メッセージの出力済み状態のリセット契機：～＜符号なし整数＞((0～99))（単位：%）

警告メッセージの出力済み状態のリセット契機を指定します。

警告メッセージ（KFPH22023-W）が出力されると，HiRDB は警告メッセージ出力済み状態となります。警告メッセージ出力済み状態となると，リスト作成数がもう一度警告メッセージの出力契機の値以上になっても警告メッセージが出力されません。しかし，ここで指定する警告メッセージの出力済み状態のリセット契機をリスト作成数が下回ると，警告メッセージの出力済み状態が解除されます。

例えば，pd_rdarea_list_no_wrn_pnt=90,70 と指定すると，リスト作成数がリスト作成可能数の 90%以上になると警告メッセージが出力されます。この後，リスト作成数がリスト作成可能数の 70%を下回るまで警告メッセージは出力されません。一度 70%を下回り，再度 90%以上になったときに警告メッセージが出力されます。

《注意事項》

- この指定を省略すると，「警告メッセージの出力契機−30」の値が仮定されます。その値がマイナスになる場合は，0 が仮定されます。
- 警告メッセージの出力契機より大きい値を指定すると，その指定は無効になり，警告メッセージの出力契機の指定値と同じ値が仮定されます。

9.2.18　排他制御に関するオペランド

◆ **pd_lck_deadlock_info = Y ｜ N**

デッドロック情報及びタイムアウト情報を出力するかどうかを指定します。デッドロック情報及びタイムアウト情報は，$PDDIR/spool/pdlckinf 下に出力されます。デッドロック情報及びタイムアウト情報については，マニュアル「HiRDB システム運用ガイド」を参照してください。

Y：デッドロック情報及びタイムアウト情報を出力します。

N：デッドロック情報及びタイムアウト情報を出力しません。

《ほかのオペランドとの関連》

- pd_lck_deadlock_check オペランドに N を指定した場合は，このオペランドに Y を指定しても，デッドロック情報を出力しません。
- このオペランドは，pd_lck_deadlock_sql_info オペランドと関連があります。

◆ **pd_lck_wait_timeout = 排他待ち限界経過時間**

～＜符号なし整数＞((0～65535))《180》（単位：秒）

排他待ち時間を監視する最大時間を秒単位で指定します。排他待ち時間とは，排他要求が待ち状態になってから解除されるまでの時間のことです。

指定した時間内に待ち状態が解除されない場合，SQL はエラーリターンします。0 を指定した場合，排他待ち時間を監視しないで，待ち状態が解除されるまで待ち続けます。

《クライアント環境定義との関連》

- PDLCKWAITTIME

  このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDLCKWAITTIME を指定してください。PDLCKWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

- PDCWAITTIME

  次に示す関係を満たすように各オペランドを設定することで，排他待ち限界経過時間を迎えても待ち状態が解除されない場合に，タイムアウト情報を出力するようになります。

  クライアント環境定義の PDCWAITTIME の指定値＞ pd_lck_wait_timeout の指定値（クライアントごとに設定している場合は，PDLCKWAITTIME の指定値）

  PDCWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」，タイムアウト情報については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《ほかのオペランドとの関連》

- このオペランドは，次に示すオペランドと関連があります。
  - ・pd_lck_deadlock_check
  - ・pd_lck_deadlock_check_interval
  - ・pd_lck_deadlock_info

◆ **pd_nowait_scan_option = LOCK ｜ NOLOCK**

WITHOUT LOCK NOWAIT 指定の検索を実行したときの処理方式を指定します。

LOCK：

WITHOUT LOCK NOWAIT 指定の検索を実行した場合，行単位で排他を確保します。このとき，ほかのトランザクションが更新前か更新後の行だけを参照できます。ほかのトランザクションが更新中の場合は，その行の更新処理が終了するまで待機します。

NOLOCK：

WITHOUT LOCK NOWAIT 指定の検索を実行した場合，排他を確保しません。このとき，ほかのトランザクションが更新中の行も参照できます。ただし，更新中の行を参照した場合，検索対象の行を受け取れないことがあります。

《注意事項》

このオペランドに LOCK を指定して，非 FIX 表の UPDATE 及び WITHOUT LOCK NOWAIT 指定の検索を実行すると，排他資源数が 1 トランザクション当たり 1 個増加します。また，非 FIX 表に対する UPDATE 文による更新で出力されるデータ 1 件当たりの基本行ログ量が 628 バイト増加します。

《各見積もり式への影響》

pd_nowait_scan_option オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「表データ更新時に出力されるシステムログ量」の「基本行ログ量の見積もり」

◆ **pd_lck_queue_limit = 排他待ちユーザ数警告メッセージ出力契機**
～<符号なし整数>((0～500))《10》

同一リソースへの排他処理で，排他待ちユーザ数が多数発生したとき，警告メッセージを出力する境界となる排他待ちユーザ数を指定します。

同一リソースでの排他待ちユーザ数が指定値に達すると，警告メッセージ（KFPS00446-W）を出力します。メッセージの出力先は，メッセージログファイル及び syslogfile です。

指定値が 0 の時は，警告メッセージを出力しません。

◆ **pd_deadlock_priority_use = Y | N**

デッドロックの優先順位を制御するかどうかを指定します。

Y：

デッドロック発生時，どのプログラムを優先的に実行させるかを制御します。デッドロックが発生した場合，優先順位の一番低いプログラムの SQL がエラーとなります。優先順位が等しい場合は，トランザクションを実行した時刻が一番遅いプログラムの SQL がエラーとなります。デッドロックの優先順位は，クライアント環境定義の PDDLKPRIO オペランドで指定してください。クライアント環境定義の PDDLKPRIO オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

N：

実行したトランザクションの時刻が一番遅いプログラムの SQL がエラーとなります。

◆ **pd_command_deadlock_priority = 32 | 64 | 96 | 120**

コマンドのデッドロックプライオリティ値を指定します。

このオペランドの指定値が有効になる運用コマンドを次に示します。

- pdhold -b（バックアップ閉塞）
- pdhold -s（同期化閉塞）
- pddbchg（レプリカ RD エリアのレプリカステータスの切り替え）
- pdorbegin（オンライン再編成のデータベース静止化）
- pdorend（オンライン再編成の追い付き反映）

コマンドのデッドロックプライオリティ値を変更する方法についてはマニュアル「HiRDB システム運用ガイド」を参照してください。

《前提条件》

pd_deadlock_priority_use オペランドに Y が指定されている必要があります。

《指定値の目安》

デッドロック発生時，業務プログラム側でリトライ可能なシステムの場合，運用コマンドの優先度を高くして業務プログラム側をエラーにできます。指定する値が小さいほど排他時の優先度は高くなります。

◆ **pd_key_resource_type = TYPE1 | TYPE2**

インデクスキー値の排他資源の作成方法を指定します。インデクスキー値の排他資源の作成方法については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

TYPE1：インデクスキー値の排他資源をビットシフトしながら排他的論理和で作成します。

TYPE2：インデクスキー値の排他資源をバイトオーダの排他的論理和で作成します。

《指定値の目安》

- 通常は TYPE1 を指定します。キー長が 10 バイトを超える場合，インデクスキー値の排他資源でシノニムによる排他競合が発生することがあります。TYPE1 を指定すると，これを防止できます。
- TYPE2 は，HiRDB Version 5.0 05-02 又は HiRDB Version 4.0 04-05 より前の処理方式です。バージョンアップしたユーザで，以前の処理方式でインデクスキー値の排他資源でシノニムによる排他競合が発生していなかった場合は，TYPE2 を指定してください。

◆ **pd_lck_pool_size = サーバ当たりの排他制御用プールサイズ**
～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：**((1～2000000))《16000》**
- 64 ビットモードかつ推奨モードの場合：**((1～2000000000))《128000》**
- 64 ビットモードかつ 0904 互換モードの場合：**((1～2000000000))《32000》**

HiRDB/シングルサーバの場合は，シングルサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。

HiRDB/パラレルサーバの場合は，バックエンドサーバ及びディクショナリサーバの排他制御で使用する共用メモリ領域の大きさをキロバイト単位で指定します。なお，フロントエンドサーバの排他制御で使用する共用メモリ領域の大きさは，pd_fes_lck_pool_size オペランドで指定してください。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式でこのオペランドの指定値を見積もってください。

HiRDB の種類	計算式（単位：キロバイト）
HiRDB/シングルサーバの場合（32 ビットモード）	↑↑a÷pd_lck_pool_partition の値↑÷6↑×pd_lck_pool_partition の値
HiRDB/パラレルサーバの場合（32 ビットモード）	↑↑b÷pd_lck_pool_partition の値↑÷6↑×pd_lck_pool_partition の値
HiRDB/シングルサーバの場合（64 ビットモード）	↑↑a÷pd_lck_pool_partition の値↑÷4↑×pd_lck_pool_partition の値
HiRDB/パラレルサーバの場合（64 ビットモード）	↑↑b÷pd_lck_pool_partition の値↑÷4↑×pd_lck_pool_partition の値

a：シングルサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「排他資源数の見積もり」を参照してください。

b：各サーバ（ディクショナリサーバ又はバックエンドサーバ）で同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「排他資源数の見積もり」を参照してください。

注

特に，定義系 SQL の DROP TABLE 又は DROP SCHEMA を実行する場合は，このオペランドの指定値を正確に見積もって指定してください。

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報※）の「排他資源管理テーブル使用率（%OF USE LOCK TABLE）」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

注※

HiRDB/パラレルサーバの場合は「サーバごとのシステムの稼働に関する統計情報」で，ディクショナリサーバ及びバックエンドサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE）」を参照してください。

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- 行数が多い表に対して，行単位に排他を取得しながら全件検索すると，件数の増加に応じてこの指定値も拡張しなければならなくなり，メモリ所要量が増加してしまいます。このような場合は，UAP で次のような工夫をしてください。

  ・表単位に排他を取得する

  ・無排他検索機能が適用できれば，無排他で検索する

  ・検索条件を絞り，複数回のトランザクションに分けて処理する

《ほかのオペランドとの関連》

- このオペランドは，pd_lck_pool_partition オペランドと関連があります。

◆ pd_fes_lck_pool_size = フロントエンドサーバの排他制御用プールサイズ ～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：**((1～2000000))**
- 64 ビットモードの場合：**((1～2000000000))**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

フロントエンドサーバの排他制御で使用する共用メモリ領域（排他制御用プール）の大きさをキロバイト単位で指定します。なお，シングルサーバ，ディクショナリサーバ，又はバックエンドサーバの排他制御で使用する共用メモリ領域の大きさは，pd_lck_pool_size オペランドで指定してください。

このオペランドの指定対象の領域は，排他資源管理テーブルとして使用されます。排他資源管理テーブル数，排他要求数，及び排他制御用プールサイズの間には，次の関係があります。

「排他資源管理テーブル数」＝「排他要求数」＝「排他制御用プールサイズ」×係数

《指定値の目安》

- 32 ビットモードの場合，6 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 64 ビットモードの場合，4 件の排他要求で 1 キロバイトの排他制御用プールサイズが必要です。
- 次に示す計算式を基に，このオペランドの指定値を見積もってください。

  ↑↑（ａ＋ｂ）÷pd_fes_lck_pool_partition の値↑÷ｃ↑×pd_fes_lck_pool_partition の値　（キロバイト）

  ａ：フロントエンドサーバで同時実行するトランザクションの排他要求数の総和です。排他要求数は SQL によって異なります。排他要求数の総和の求め方については，「排他資源数の見積もり」を参照してください。

  ｂ：（pd_max_users ＋ pd_max_reflect_process_count の値＋ 3）×（pd_max_access_tables の値＋ 4）

  ｃ：32 ビットモードの場合は 6，64 ビットモードの場合は 4 を代入してください。

《指定値のチューニング方法》

統計解析ユティリティ（サーバごとのシステムの稼働に関する統計情報）で，フロントエンドサーバの「排他資源管理テーブル使用率（%OF USE LOCK TABLE)」を参照してください。使用率の最大値が 80%以上の場合は，これからのデータベース拡張に備えて，このオペランドの値を大きくすることをお勧めします。使用率の最大値が 10%以下の場合は，共用メモリを節約するために，このオペランドの値を小さくすることをお勧めします。

《オペランドの省略値》

このオペランドの省略値を次に示します。

・32 ビットモードの場合

{(pd_max_users ＋ pd_max_reflect_process_count の値＋ 3）×（pd_max_access_tables の値＋ 4)} ÷ 6

・64 ビットモードの場合

{(pd_max_users ＋ pd_max_reflect_process_count の値＋ 3）×（pd_max_access_tables の値＋ 4)} ÷ 4

《注意事項》

- このオペランドの指定値が小さ過ぎると，SQL がエラーリターンすることがあります。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは，pd_fes_lck_pool_partition オペランドと関連があります。

◆ pd_lck_pool_partition = サーバ当たりの排他制御用プール分割数

～＜符号なし整数＞((1～5000))《1》

排他制御処理を分散する場合にこのオペランドを指定してください。

HiRDB/シングルサーバの場合は，排他制御処理を分散するときにシングルサーバの排他制御で使用する排他制御用プールパーティション数を指定します。

HiRDB/パラレルサーバの場合は，排他制御処理を分散するときにバックエンドサーバ及びディクショナリサーバの排他制御で使用する排他制御用プールパーティション数を指定します。なお，フロントエンドサーバの排他制御で使用する排他制御用プールパーティション数は，pd_fes_lck_pool_partition オペランドで指定してください。

排他制御処理の分散については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり

SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。

- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。
- このオペランドに 2 以上を指定した場合は，デッドロック監視の処理方式がインターバル監視方式となり，デッドロック発生から検出までに時間差が発生します。デッドロック監視の処理方式については，マニュアル「HiRDB UAP 開発ガイド」の「デッドロックと回避策」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_pool_size
- pd_lck_deadlock_check_interval

《各見積もり式への影響》

pd_lck_pool_partition オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバで起動するプロセス」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバで起動するプロセス」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ pd_fes_lck_pool_partition = フロントエンドサーバの排他制御用プール分割数

～＜符号なし整数＞((1～5000))《1》

排他制御処理を分散するときにフロントエンドサーバの排他制御で使用する排他制御用プールパーティション数を指定します。なお，シングルサーバ，ディクショナリサーバ，又はバックエンドサーバの排他制御で使用する排他制御用プールパーティション数は，pd_lck_pool_partition オペランドで指定してください。

排他制御処理の分散については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《指定値のチューニング方法》

排他制御用プールパーティション数のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

- このオペランドの指定値を大きくし過ぎると，各排他制御用プールパーティションに割り当てられる共用メモリのサイズが小さくなるため，排他制御用プールパーティションが容量不足となり SQL がエラーリターンすることがあります。また，排他制御の処理に時間が掛かるため，システムのパフォーマンスが下がります。この場合，このオペランドの指定値を小さくしてください。

- 排他制御用プールサイズは 1 キロバイト以上必要であるため，pd_lck_pool_size の値よりも大きな値を指定した場合，このオペランドには pd_lck_pool_size の値が仮定されます。また，このとき，KFPS00421-W メッセージが出力されます。
- このオペランドに 2 以上を指定した場合は，デッドロック監視の処理方式がインターバル監視方式となり，デッドロック発生から検出までに時間差が発生します。デッドロック監視の処理方式については，マニュアル「HiRDB UAP 開発ガイド」の「デッドロックと回避策」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_fes_lck_pool_size
- pd_lck_deadlock_check_interval

《各見積もり式への影響》

pd_fes_lck_pool_partition オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ pd_lck_until_disconnect_cnt = 1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和

～＜符号なし整数＞((0～140000))

- ディクショナリサーバ以外かつ推奨モードの場合：**《10000》**
- ディクショナリサーバかつ推奨モードの場合：**《256》**
- 0904 互換モードの場合：**《256》**

トランザクションを越えて保持する表，RD エリア，インナレプリカ構成管理，及びレプリカグループ構成管理の排他資源数を指定します。このオペランドの値に従って，表，RD エリア，インナレプリカ構成管理，レプリカグループ構成管理の UNTIL DISCONNECT 指定の排他を管理するブロックを共用メモリに確保します。

《注意事項》

HiRDB/パラレルサーバの場合，バックエンドサーバとディクショナリサーバでこのオペランドの指定値の見積もり方法が異なります。サーバ共通定義でこのオペランドを指定した場合，バックエンドサーバとディクショナリサーバの両方に指定値が適用されます。ディクショナリサーバだけ異なる値を有効にしたい場合は，ディクショナリサーバ定義にこのオペランドを指定してください。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。次に示す場合に指定値を変更するかどうかを検討してください。

- 同時に実行するユティリティの数を大量に増やす場合
- ホールダブルカーソルを使用する場合

- pdlbuffer オペランドで指定したローカルバッファを使用する場合
- 共用 RD エリアを使用する場合
- SQL セッション固有一時表を使用する場合

このオペランドの指定値の見積もり方法については，「1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式」を参照してください。

《指定値のチューニング方法》

このオペランドの指定値が小さいと，トランザクションがロールバックしたり，リターンコード 8 でユティリティが異常終了したりします。このとき，KFPA11914-E，又は KFPH28001-E メッセージが出力されます。この現象が発生したら，このオペランドの値を大きくしてください。

なお，このオペランドの値を大きくすると，それに比例して必要なメモリと，シンクポイントダンプファイルの容量が増加します。《各見積もり式への影響》に従って見積もる必要があります。

《各見積もり式への影響》

pd_lck_until_disconnect_cnt オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「シンクポイントダンプファイルのレコード数の求め方」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 2」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」

◆ pd_max_open_holdable_cursors = UNTIL DISCONNECT 指定 LOCK 文非実行時のホールダブルカーソルの最大同時オープン数

～<符号なし整数>((16～1024))《16》

UNTIL DISCONNECT 指定の LOCK 文を実行しない表に対してホールダブルカーソルを使用する場合に，そのホールダブルカーソルのトランザクション当たりの最大同時オープン数を指定します。

《注意事項》

このオペランドに省略時解釈値以外の値を指定すると，共用メモリ使用量が増加します。

《ほかのオペランドとの関連》

このオペランドと次に示すオペランドの指定値は，ロックサーバの共用メモリサイズ計算に使用されます。32 ビットモードの HiRDB の場合，各オペランドの指定値が大き過ぎると，ロックサーバの共用メモリサイズが 2 ギガバイトを超えるため，HiRDB を開始できなくなることがあります。ロックサーバの共用メモリサイズが 2 ギガバイトを超えないように，このオペランドと次に示すオペランドの指定値を調整してください。

- pd_max_access_tables
- pd_max_users
- pd_max_bes_process
- pd_max_dic_process
- pd_lck_hash_entry
- pd_lck_pool_size

共用メモリについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

◆ **pd_lck_hash_entry = 排他制御用プールで使用するハッシュエントリ数**
～＜符号なし整数＞((0～2147483647))《0》

排他制御用プールで使用するハッシュテーブルのエントリ数を指定します。HiRDB はここで指定された値に従ってサーバ（シングルサーバ，フロントエンドサーバ，バックエンドサーバ，ディクショナリサーバ）ごとにユニットコントローラ用の共用メモリに排他制御用プールを確保します。

《指定値の目安》

通常はこのオペランドを省略してください。

次に示す条件に該当する場合は指定値を検討してください。

- バージョン 06-02 以降にバージョンアップする場合に共用メモリサイズをできるだけ変更したくないときは 11261 を指定してください。この場合，バージョンアップ前と同数のハッシュエントリ数を確保するため，排他制御用プール中のハッシュテーブルサイズがバージョンアップ前と同じになります。
- このオペランドに推奨値より大きい値を指定すると，性能が向上することがあります。ただし，変数 a より大きい値を指定しても，a を指定したときよりも性能が向上することはありません。

  推奨値を次に示します。

  推奨値＝ MAX（↑a÷10↑，11261）を超えない最大の素数

変数	変数の算出式		
a	シングルサーバの場合	（pd_max_users ＋ pd_max_reflect_process_count の値＋ 3）×（pd_max_access_tables の値＋ 14）＋ pd_lck_pool_size×c	
	フロントエンドサーバの場合	pd_fes_lck_pool_size が省略されている場合	（b ＋ 3）×（pd_max_access_tables の値＋ 4）
		pd_fes_lck_pool_size が指定されている場合	pd_fes_lck_pool_size 値×c
	バックエンドサーバ，ディクショナリサーバの場合	（b ＋ 3）×10 ＋ pd_lck_pool_size の値×c	
b	フロントエンドサーバの場合	マルチフロントエンドサーバの場合	pd_max_users ＋ pd_max_reflect_process_count の値＋ 1
		マルチフロントエンドサーバでない場合	pd_max_users ＋ pd_max_reflect_process_count の値

変数	変数の算出式		
	バックエンドサーバの場合	pd_max_users の値＞ pd_max_bes_process の値の場合	pd_max_users ＋ pd_max_reflect_process_count の値
		pd_max_users の値≦ pd_max_bes_process の値の場合	pd_max_bes_process ＋ pd_max_reflect_process_count の値
	ディクショナリサーバの場合	pd_max_users の値＞ pd_max_dic_process の値の場合	pd_max_users ＋ pd_max_reflect_process_count の値
		pd_max_users の値≦ pd_max_dic_process の値の場合	pd_max_dic_process ＋ pd_max_reflect_process_count の値
c	32 ビットモードの場合は 6，64 ビットモードの場合は 4		

《オペランドの規則》

- このオペランドが 0 の場合，HiRDB はサーバごとに推奨値を計算します。
- このオペランドに 0 でも素数でもない値を指定した場合，HiRDB はその指定値を超えない最大の素数が指定されたと仮定します。

《注意事項》

このオペランドの指定値が小さ過ぎると，ハッシュエントリ不足が発生して性能が低下することがあります。このオペランドを省略すれば，ハッシュエントリ不足及びハッシュエントリ不足による性能低下は発生しません。

◆ **pd_indexlock_mode = KEY ｜ NONE**

B-tree インデクスの排他制御方式を指定します。

KEY：インデクスキー値で排他をします。

NONE：インデクスキー値で排他をしません（**インデクスキー値無排他**を実施します）。

インデクスキー値無排他については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《指定値の目安》

通常は NONE を推奨します。ただし，ユニークインデクスの更新（削除したキー値の追加）が発生する場合は，KEY にするか，又は残存エントリを残さない運用を検討してください。

《ほかのオペランドとの関連》

pd_inner_replica_control オペランドを指定すると，このオペランドの指定値は NONE が仮定されます。

このオペランドは pd_lock_uncommited_delete_data オペランドと関連があります。詳細については，pd_lock_uncommited_delete_data オペランドの説明を参照してください。

◆ **pd_lock_uncommited_delete_data = WAIT ｜ NOWAIT**

コミットしていない削除データの排他待ち制御方式を指定します。

WAIT：

DELETE 文で削除した行，又は UPDATE 文で更新した更新前のインデクスキーに対し，検索時に削除又は更新したトランザクションが決着するまで排他待ちし，トランザクション決着後のデータを

基に検索結果を返す方式を指定します。また，**インデクスキー値排他**と同じ方式で一意性制約が保証されます。インデクスキー値排他については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

NOWAIT：

DELETE 文で削除した行，又は UPDATE 文で更新した更新前のインデクスキーに対し，検索時に削除又は更新したトランザクションの決着を待たないで検索結果を返す方式を指定します。

《注意事項》

次の場合には，このオペランドに WAIT を指定しても無効になります。

- ディクショナリ表を検索又は更新する場合
- WITHOUT ROLLBACK オプションを指定した表にインデクスを定義しないで検索又は更新する場合

このオペランドに WAIT を指定すると，インデクスが定義された表の表格納 RD エリアを構成する HiRDB ファイルの使用可能サイズが次のように制限されます。

表格納 RD エリアのページ長（バイト）	pd_lock_uncommited_delete_data の指定値	
	WAIT	NOWAIT
4096	32GB	64GB
6144	48GB	
上記以外	64GB	

HiRDB ファイルのサイズが既に使用可能サイズを超えている場合，又は使用可能サイズ以上のサイズのファイルを使用したい場合には，表格納 RD エリアのページサイズを 8192 バイト以上に変更してください。ページサイズを変更しないでそのまま運用すると，次の場合にエラーとなり，警告メッセージ KFPA19176-E が出力されます。

- 行の追加で使用可能サイズを超える領域を割り当てたとき
- 使用可能サイズを超える領域のページに格納されている行に対して更新又は削除の操作をしたとき

《クライアント環境定義との関連》

クライアント環境定義の PDLOCKSKIP オペランドに YES を指定している場合は，このオペランドに WAIT を指定しても無効となり，無排他条件判定の処理が優先されます。そのため，削除中の行では排他待ちをしません。ただし，削除した行のインデクスは残存エントリとして残ります。

《ほかのオペランドとの関連》

このオペランドの指定値と pd_indexlock_mode オペランドの指定値には，指定できない組み合わせがあります。指定できない組み合わせを次に示します。

pd_lock_uncommited_delete_data の指定値	pd_indexlock_mode の指定値	
	NONE	KEY
WAIT	○	×※1
NOWAIT	○	○

pd_lock_uncommited_delete_data の指定値	pd_indexlock_mode の指定値	
	NONE	KEY
NOWAIT（過去に WAIT 指定履歴あり）	○	×※2

（凡例）

○：指定できる組み合わせです。

×：指定できない組み合わせです。

注※1

指定した場合，HiRDB 起動時にエラー（KFPS01857-E）となります。

注※2

このオペランドに WAIT を指定して HiRDB を開始したことがあると，このオペランドに NOWAIT を指定して pd_indexlock_mode オペランドに KEY を指定した場合，DB アクセス時に不正な検索結果が出力されます。データベース初期設定ユティリティ（pdinit）を実行して HiRDB を初期化してから再度オペランドを指定してください。

◆ **pd_dbreuse_remaining_entries = ALL ｜ ONLY_DIC ｜ NONE ｜ ONLY_USER ｜ NOTHING**

インデクスキー値無排他を適用している場合，インデクスのキー値を削除した際に，インデクス格納ページに残存エントリが発生します。この場合，複数のトランザクションで次の操作を実行すると，異なるインデクスキー値の操作であっても排他待ちが発生することがあります。

- ユーザ表の場合

  INSERT 文，UPDATE 文，又は，DELETE 文を繰り返し実行する。

- データディクショナリ表の場合

  定義系 SQL 文，pdmod を繰り返し実行する。

このオペランドは，行データ挿入の際に，行識別子（行を一意に識別するためにシステムが割り当てる値）の再利用を抑止して，排他待ちの発生頻度を低くするためのオペランドです。

ALL：

行識別子を再利用します。

複数のトランザクションで表を更新したときに，操作対象，及び挿入対象とした行識別子と残存エントリの行識別子が重複した場合，異なるインデクスやキー値の操作であっても排他待ちが発生することがあります。

ONLY_DIC：

データディクショナリ表だけ，行識別子を再利用します。

ユーザ表は，ページごとに決められた行識別子の上限（255）を超過するまで行識別子の再利用を抑止します。

これによって，複数トランザクションで異なるインデクスキー値を操作したときの排他待ちの発生頻度を低くできます。

NONE：

指定した場合，ONLY_DIC とみなして動作します。通常は ONLY_DIC を指定してください。

ONLY_USER：

ユーザ表だけ，行識別子を再利用します。

データディクショナリ表は，ページごとに決められた行識別子の上限（255）を超過するまで行識別子の再利用を抑止します。

これによって，複数トランザクションで異なるインデクスキー値を操作したときの排他待ちの発生頻度を低くできます。

NOTHING：

ユーザ表とデータディクショナリ表について，ページごとに決められた行識別子の上限（255）を超過するまで行識別子の再利用を抑止します。

これによって，複数トランザクションで異なるインデクスキー値を操作したときの排他待ちの発生頻度を低くできます。

指定値ごとの，排他待ちの軽減効果を次に示します。

表 9–5　pd_dbreuse_remaining_entries の指定値ごとの，排他待ちの軽減効果

pd_dbreuse_remaining_entries の指定値	排他待ちの軽減効果	
	ユーザ表	データディクショナリ表
ALL	×	×
ONLY_DIC	○	×
NONE	○	×
ONLY_USER	×	○
NOTHING	○	○

（凡例）

○：排他待ちの軽減効果があります（行識別子を再利用しません）。

×：排他待ちの軽減効果はありません（行識別子を再利用します）。

《適用基準》

インデクスに対する削除，又は挿入処理を行う場合，インデクスの削除によって発生する残存エントリを，新たに追加するキー値の挿入処理が再利用しようとしたときに，排他待ちが発生することがあります。このオペランドは，表の更新が発生する UAP を実行する場合，格納効率を下げてでも排他待ちが発生する可能性を下げるときに指定します。詳細は，マニュアル「HiRDB UAP 開発ガイド」の「残存エントリによる排他待ちの回避（行識別子の再利用抑止）」を参照してください。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

次のすべての条件を満たす場合に，このオペランドの指定を検討してください。

（a）ユーザ表の場合

1. インデクスキー値無排他機能を適用（pd_indexlock_mode オペランドに NONE を指定）している場合

2. 次のどちらかの条件に該当する場合

・UNIQUE 又は PRIMARY を指定したインデクスを定義した表があるとき

・コミットしていない削除データの排他制御を（pd_lock_uncommited_delete_data オペランドに WAIT を指定）していて，システムにインデクスを定義した表が存在するとき

3. 2.に該当するインデクスに対して，複数のトランザクションで INSERT 文，UPDATE 文，又は DELETE 文を実行する場合

（b）データディクショナリ表の場合

1. インデクスキー値無排他機能を適用（pd_indexlock_mode オペランドに NONE を指定）している場合
2. 複数のトランザクションで，残存エントリによる排他待ちが発生する可能性がある操作（RD エリアの構成変更，又は，表，インデクスなどの定義変更）を実行する場合

なお，行識別子の再利用が抑止されるのは，上記（a）の 1.と 2.を満たしたユーザ表，及び，上記（b）の 1.を満たしたデータディクショナリ表に対して，使用中ページに行を追加する操作が発生した場合だけです。それ以外の表や操作の場合，ALL 以外を指定していても行識別子は再利用されます。

このオペランドの指定値による行識別子の割り当て方の変化，及び残存エントリによる排他待ち発生の流れについては，マニュアル「HiRDB UAP 開発ガイド」の「残存エントリによる排他待ちの回避（行識別子の再利用抑止）」を参照してください。

《注意事項》

- インデクスの残存エントリは，インデクスキー値無排他を適用している場合に，ユニークインデクスキーを更新，又は削除することで発生します。また，コミットしていない削除データの排他制御をしている場合は，通常のインデクスキーに対する更新，及び削除を行った場合も残存エントリが発生します。
- このオペランドに ALL 以外を指定した場合，ページ内の管理領域が増えるため，格納効率が低下することがあります。そのため，このオペランドの指定値を変更した場合，表格納用 RD エリア，及びインデクス格納用 RD エリアの容量を再度見積もってください。見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」の「表の格納ページ数の計算方法」及び「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

  また，格納効率の低下を防ぐため，空きページ解放ユティリティ（pdreclaim）で，定期的に空きページの解放を行ってください。詳細は，マニュアル「HiRDB システム運用ガイド」の「使用中空きページの再利用」を参照してください。

  なお，格納効率が低下するケースについては，マニュアル「HiRDB UAP 開発ガイド」の「残存エントリによる排他待ちの回避（行識別子の再利用抑止）」を参照してください。
- このオペランドに ALL 以外を指定した場合でも，255 個すべての行識別子が割り当て済みになると，ページ内に存在する残存エントリの中で行識別子の値が一番小さいものから順に再利用されます。このため，新たに追加するインデクスキー値の挿入処理が残存エントリの行識別子を再利用した際に，この残存エントリの排他が解除されていない場合（削除，又は更新を行ったトランザクションが未決着の場合），排他待ちが発生します。

排他待ちを回避するには，空きページ解放ユティリティ（pdreclaim）で，定期的に空きページの解放を行ってください。詳細は，マニュアル「HiRDB システム運用ガイド」の「使用中空きページの再利用」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_indexlock_mode
- pd_lock_uncommited_delete_data

◆ **pd_lck_deadlock_check = Y | N**

デッドロックの発生を監視するかどうかを指定します。

Y：デッドロックの発生を監視します。

N：デッドロックの発生を監視しません。

《指定値の目安》

デッドロックが発生しない業務システムでは，このオペランドに N を指定すると SQL の実行性能が改善することがあります。特に，デッドロックの検出方法がインターバル監視方式の場合，排他制御用プールパーティション数が増加するとデッドロックを検出するたびに排他制御の性能が劣化することがあるため，デッドロックが発生しない業務システムを構築した上でこのオペランドに N を指定することをお勧めします。

デッドロックが発生する業務システムでは，このオペランドに Y を指定してください。N を指定すると，デッドロックが発生した場合，pd_lck_wait_timeout オペランドに指定した時間が経過するまで SQL が終了しません。また，HiRDB がデッドロック情報を出力しないため，デッドロックが発生した要因が分からなくなるおそれがあります。

《注意事項》

このオペランドに N を指定すると，デッドロックが発生したトランザクションはエラーとなりません。この場合，トランザクションは次の要因でキャンセルされます。

- 排他待ち時間を監視する最大時間が経過し，SQL がエラーリターンする
- HiRDB クライアントの最大待ち時間が経過し，要求が UAP にエラーリターンする

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME
- PDLCKWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_info
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_deadlock_check_interval

- pd_lck_wait_timeout
- pd_lck_deadlock_sql_info

◆ **pd_lck_deadlock_check_interval＝デッドロック監視時間間隔**
～＜符号なし整数＞((1～2000000000))《1000》（単位：ミリ秒）

インターバル監視方式でデッドロックの発生を監視する場合の，監視を実施する間隔を指定します。

《前提条件》

このオペランドの前提条件は，サーバ種別によって異なります。

シングルサーバ，バックエンドサーバ，又はディクショナリサーバの場合は，次に示す条件をすべて満たしている必要があります。

- pd_lck_deadlock_check オペランドに Y を指定している
- pd_lck_pool_partition オペランドに 2 以上の値を指定している

フロントエンドサーバの場合は，次に示す条件をすべて満たしている必要があります。

- pd_lck_deadlock_check オペランドに Y を指定している
- pd_fes_lck_pool_partition オペランドに 2 以上の値を指定している

《指定値の目安》

デッドロックが発生してから検出するまでの時間を短くしたい場合は，このオペランドの指定値を小さくしてください。

《注意事項》

- このオペランドの指定値を小さくし過ぎると，システムのパフォーマンスが低下する場合があります。
- このオペランドの指定値を大きくし過ぎると，次の要因でトランザクションがキャンセルされることがあります。
  ・排他待ち時間を監視する最大時間が経過し，SQL がエラーリターンする
  ・HiRDB クライアントの最大待ち時間が経過し，要求が UAP にエラーリターンする

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_check
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_wait_timeout

◆ **pd_idx_without_rollback = Y | N**

WITHOUT ROLLBACK オプション指定表のインデクス構成列の更新可否を指定します。

Y：WITHOUT ROLLBACK オプション指定表のインデクス構成列を更新できるようになります。

N：WITHOUT ROLLBACK オプション指定表のインデクス構成列を更新すると，次のようになります。

行の追加，削除：WITHOUT ROLLBACK オプションの機能が無効になります。行更新による行排他が残り，ロールバックでデータが戻ります。

行の更新：SQL エラーとなります。

《指定値の目安》

09-50 以前の動作を前提にアプリケーションを作成している場合，このオペランドに N を指定してください。

《注意事項》

- pdsetup コマンドに 0904 互換モードを指定した場合，このオペランドの省略値は N になります。
- このオペランドに Y を指定し，WITHOUT ROLLBACK オプション指定表のインデクス構成列を更新すると，1 行の更新ごとにシステムログ量が 312 バイト増加します。システムログファイルの容量見積もり時に加算してください。

《ほかのオペランドとの関連》

このオペランドに Y を指定する場合，pd_indexlock_mode オペランドには NONE を指定してください。pd_indexlock_mode オペランドに KEY を指定した場合，pd_idx_without_rollback オペランドに Y を指定しても N を仮定します。

◆ **pd_isolation_level = データ保証レベル**

～＜符号なし整数＞((0～2))《2》

SQL 文のデータ保証レベルを指定します。データ保証レベルとは，トランザクションのどの時点までデータの内容を保証するかのレベルのことをいいます。このオペランドは，SQL 文に指定する排他オプションに相当します。

データ保証レベルの指定値については，マニュアル「HiRDB UAP 開発ガイド」の「クライアント環境定義の設定内容」の PDISLLVL オペランドを参照してください。

《前提条件》

このオペランドのデータ保証レベルを有効にするためには，HiRDB クライアントのバージョン 09-65 以降を使用して HiRDB と接続してください。このオペランドは，HiRDB クライアントのバージョン 09-65 以降を使用している場合だけ有効です。HiRDB クライアントのバージョン 09-60 以前を使用した場合，このオペランドの値は無効となり，データ保証レベルは 2 となります。

◆ **pd_unique_indexlock_info = Y | N**

ユニークインデクスの残存エントリによる排他待ち情報(KFPH25005-I メッセージ)の出力有無を設定します。

Y：KFPH25005-I メッセージを出力します。

N：KFPH25005-I メッセージを出力しません。

《指定値の目安》

通常は Y を指定してください。KFPH25005-I メッセージを抑止したい場合，N を指定してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_indexlock_mode

◆ pd_unique_check_mode = 0 | 1

インデクスキー値無排他方式の，ユニークインデクスの一意性制約保証処理での排他制御方式を指定します。

このオペランドの指定値に 1 を指定すると，残存エントリによって待たされたり，デッドロックが発生したりすることがなくなります。

ユニークインデクスの一意性制約の保証処理及び残存エントリについては，マニュアル「HiRDB UAP 開発ガイド」の「インデクスキー値無排他」を参照してください。

0：行排他だけでユニークインデクスの一意性制約保証を制御します。

1：行排他，トランザクション待ち用排他，ユニークチェック排他でユニークインデクスの一意性制約保証を制御します。

《指定値の目安》

複数のトランザクションで以下のどれかの運用を行う場合にこのオペランドに 1 を指定することを検討してください。

- ユニークインデクスを定義している表で行の削除した後，削除したキー値を使って再挿入する。
- ユニークインデクスのインデクス構成列を更新する。
- RD エリアの構成変更（pdmod）や表やインデクスなどの定義変更を頻繁に行う。

《注意事項》

このオペランドの指定値に 1 を指定した場合，ユニークインデクスのキー値を削除するときにトランザクション待ち用排他とユニークチェック排他を取得します。そのため，必要な排他資源数を見積もった上で排他制御用プールサイズ（pd_lck_pool_size）を決定してください。

《ほかのオペランドとの関連》

このオペランドは pd_indexlock_mode オペランドに NONE を指定した場合に有効です。

《各見積もり式への影響》

pd_unique_check_mode オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「システムファイル及び監査証跡ファイルの容量の見積もり」

マニュアル「HiRDB システム定義」の「付録 D　排他資源数の見積もり」

◆ **pd_lck_deadlock_sql_info = Y | N**

デッドロック SQL 情報を出力するかどうかを指定します。デッドロック SQL 情報は，$PDDIR/spool/pdlckinf 下に出力されます。

Y：デッドロック SQL 情報を出力します。

N：デッドロック SQL 情報を出力しません。

デッドロック SQL 情報については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。

《ほかのオペランドとの関連》

- pd_lck_deadlock_check オペランドに N を指定した場合は，このオペランドに Y を指定しても，デッドロック情報を出力しません。
- pd_lck_deadlock_info オペランドに N を指定した場合は，このオペランドに Y を指定しても，デッドロック情報を出力しません。

◆ **pd_lck_deadlock_check_timeout = デッドロック監視処理の最大処理時間**

～＜符号なし整数＞ ((0～2000000000))《0》（単位：ミリ秒）

インターバル監視方式でのデッドロック監視処理の最大処理時間を指定します。指定した時間内にデッドロック監視処理が完了しない場合は，監視処理を中断します。これによって，デッドロック監視処理で取得する排他制御用プールのスレッド間ロックを解放し，トランザクションの排他取得又は排他解除の制御をできるようにします。

このオペランドに 0 を指定した場合は，デッドロック監視処理の時間監視及び中断を行いません。

詳細は，マニュアル「HiRDB システム運用ガイド」の「デッドロック監視処理時間を変更する方法」を参照してください。

《前提条件》

このオペランドの前提条件は，サーバ種別によって異なります。

シングルサーバ，バックエンドサーバ，ディクショナリサーバの場合：

以下のすべての条件を満たす場合，有効になります。

- pd_lck_deadlock_check オペランドに Y を指定している
- pd_lck_pool_partition オペランドに 2 以上の値を指定している

フロントエンドサーバの場合：

以下のすべての条件を満たす場合，有効になります。

- pd_lck_deadlock_check オペランドに Y を指定している
- pd_fes_lck_pool_partition オペランドに 2 以上の値を指定している

《指定値の目安》

多くのトランザクションが，同じ排他資源の排他モードを遷移する UAP[※1] を実行する場合（タイミングによってデッドロックが発生する），次に示す計算式を満たす値をこのオペランドに指定してください。

システム内で最小の排他待ち時間限界経過時間※2≧
pd_lck_deadlock_check_interval ＋ pd_lck_deadlock_check_timeout

注※1

例えば，以下のようなトランザクションを実行する UAP が該当します。

- for update 指定なしで検索した行を更新するトランザクション
- ページ排他指定の表を検索・更新するトランザクション

注※2

この値は以下のどれかの指定値を指します。

- PDLCKWAITTIME
- pd_lck_wait_timeout
- ユティリティ，運用コマンドのオプション
  （空きページ解放ユティリティ(pdreclaim)の-w オプションなど）

《注意事項》

このオペランドに 0 以外の値を指定し，デッドロック監視処理を中断した場合，デッドロックが発生しても検出しない場合があります。

その場合，次の要因でトランザクションがキャンセルされるまで，デッドロックが解消されません。

- 排他待ち限界経過時間が経過して，SQL がエラーリターンする場合
- HiRDB クライアントの最大待ち時間が経過して， UAP にエラーリターンする場合

また，このとき HiRDB がデッドロック情報を出力しないため，デッドロックが発生した要因が分からなくなるおそれがあります。

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME
- PDLCKWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_check
- pd_lck_deadlock_check_interval
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_wait_timeout
- pd_lck_deadlock_check_pause

◆ pd_lck_deadlock_check_pause = デッドロック監視処理中断時の監視処理休止時間～<符号なし整数> ((1～65535))《180》（単位：秒）

デッドロック監視処理が pd_lck_deadlock_check_timeout オペランド指定値以上掛かったことによって中断した場合，次回デッドロック監視処理を再開するまでの時間を指定します。

デッドロックの多発によってデッドロック監視処理が中断している場合，デッドロックしているトランザクションが終了してデッドロックの発生数が減少するまで，繰り返しデッドロック監視処理が中断することがあります。このオペランドに pd_lck_deadlock_check_interval オペランド指定値の時間よりも大きな時間を指定することで，デッドロック監視処理よりもトランザクションの排他制御を優先します。これによって，デッドロックしているトランザクション数が減り，デッドロック監視時間の短縮が期待できます。

このオペランドは pd_lck_deadlock_check_timeout オペランドに 0 以外の値を指定した場合に有効になります。

詳細は，マニュアル「HiRDB システム運用ガイド」の「デッドロック監視処理時間を変更する方法」を参照してください。

《指定値の目安》

デッドロック監視処理が中断してからデッドロック発生数が減少するまでに時間が掛かる場合に，このオペランドの値を設定してください。

システム内で最小の排他待ち時間限界経過時間より大きな値を指定してください。

《注意事項》

監視処理休止時間の間は，デッドロックが発生していても検出しないため，次の要因でトランザクションがキャンセルされるまで，デッドロックが解消されません。

- 排他待ち限界経過時間が経過して，SQL がエラーリターンする場合
- HiRDB クライアントの最大待ち時間が経過して，UAP にエラーリターンする場合

また，このとき HiRDB がデッドロック情報を出力しないため，デッドロックが発生した要因が分からなくなるおそれがあります。

《クライアント環境定義との関連》

このオペランドは次に示すクライアント環境定義と関連があります。

- PDCWAITTIME
- PDLCKWAITTIME

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_deadlock_check
- pd_lck_deadlock_check_interval
- pd_lck_pool_partition
- pd_fes_lck_pool_partition
- pd_lck_wait_timeout

- pd_lck_deadlock_check_timeout

9.2.19　バッファに関するオペランド

◆ **pd_sql_object_cache_size = SQL オブジェクト用バッファ長**
～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：**((22～256000))《(pd_max_users の値＋ n）×22》**
- 64 ビットモードかつ推奨モードの場合：**((22～2000000))《(pd_max_users の値＋ n）×40》**
- 64 ビットモードかつ 0904 互換モードの場合：**((22～2000000))《(pd_max_users の値＋ n）×22》**

n：HiRDB/パラレルサーバでマルチフロントエンドサーバ構成を適用する場合は 4，それ以外は 3

SQL オブジェクトを格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- SQL オブジェクトは，ユーザごとにトランザクションが終了するまでバッファに保存されます。したがって，同時に実行するトランザクションの SQL オブジェクトがバッファに格納できる大きさが必要です。
- SQL オブジェクト用バッファのヒット率が低い場合，SQL 文解析処理のオーバヘッドによって性能が低下することがあります。
- 静的 SQL の SQL オブジェクトはトランザクション終了後もバッファに空きがなくなるまで保存して，同一 UAP を実行する複数のユーザで共用することで，SQL 解析処理の削減を図ります。バッファを有効に利用するためには，使用頻度の高い UAP の SQL オブジェクトがバッファに常駐するようにバッファを確保してください。
- バッファ長の見積もり時は，まず UAP で発行する SQL 文の SQL オブジェクト長から UAP 実行に必要なバッファ長を見積もります。次に，同時に実行される UAP と同時実行ユーザ数を考慮して必要なバッファ長を算出してください。
- 1SQL 文の SQL オブジェクト長の見積もり方法については，「SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式」を参照してください。
- Type4 JDBC ドライバを使用している場合，SELECT 文，INSERT 文，DELETE 文，UPDATE 文，PURGE TABLE 文，及び CALL 文の SQL オブジェクトは，トランザクションが終了しても解放せずコネクションが終了するまでバッファに保存されます。Java EE アプリケーションサーバと連携し，コネクションプーリング及びステートメントプーリング機能を使用しているとき，プーリング中のステートメントに対応する SQL オブジェクトはバッファに保存されているため，コネクションプーリング数及びステートメントプーリング数を考慮して必要なバッファ長を算出してください。

《指定値のチューニング方法》

SQL オブジェクト用バッファ長のチューニングについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

《各見積もり式への影響》

pd_sql_object_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ **pd_def_buf_control_area_assign = INITIAL ｜ TRAN**

表定義情報用バッファ，ビュー解析情報用バッファ，ユーザ定義型情報用バッファ，及びルーチン定義情報用バッファの管理領域を HiRDB 開始時に一括して確保するか，又はトランザクション開始時に確保するかを指定します。

INITIAL：

各種バッファの管理領域を HiRDB 開始時に一括して確保します。この管理領域は HiRDB を終了するまで解放されません。

TRAN：

各種バッファの管理領域をトランザクション開始時に確保します。この管理領域はトランザクション終了時に解放されます。

《指定値の目安》

性能を重視する場合は INITIAL を指定してください。トランザクション性能が向上します。ただし，TRAN 指定時よりも共用メモリを多く必要とするため，メモリ所要量を見直してください。共用メモリの見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《各見積もり式への影響》

pd_def_buf_control_area_assign オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ **pd_thread_max_stack_size = 1 スレッドが使用する最大スタックサイズ**
～＜符号なし整数＞((0～OS が許容する最大スタックサイズ（最大 2047))《0》（単位：メガバイト）

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

1 スレッドが使用するスタックサイズの最大値を指定します。0 を指定するか，又はこのオペランドを省略した場合，1 スレッドが使用するスタックサイズを制限しません。HiRDB は擬似スレッド制御をしているため，独自に確保したメモリをスタックとして割り当てています。このため，擬似スレッドに対しては OS のオペレーティングシステムパラメタの maxssiz（Linux の場合は stacksize）の値が有効となりません。maxssiz の代わりにこのオペランドを指定します。

このオペランドには maxssiz 以下の値を指定してください。

《利点》

UAP の不良，又は UAP からの複雑な SQL の要求などによって，1 スレッドが多量のシステムリソースを使用することがあります。これが原因で OS がハングアップすることがあります。このオペランドを指定すると，1 スレッドが使用するスタックサイズを制限できるため，OS のハングアップが防止できます。

《注意事項》

このオペランドの指定値が，OS が許容する最大スタックサイズ以下であることを確認してください。OS が許容する最大スタックサイズは maxssiz で確認できます。

◆ **pd_thread_stack_expand_size = 1 スレッド当たりのスタック拡張サイズ**
～＜符号なし整数＞((0～2146435072))《0》（単位：バイト）

HiRDB が制御するスレッドの，1 スレッド当たりのスタックサイズを拡張するオペランドです。

通常，このオペランドは省略します。保守員から依頼があった場合に変更してください。

◆ **pd_table_def_cache_size = 表定義情報用バッファ長**
～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：**((100～65535))** 《↑$\sqrt{\text{pd\_max\_users の値}+n}$↑×100》
- 64 ビットモードかつ推奨モードの場合：**((100～2000000))《16000》**
- 64 ビットモードかつ 0904 互換モードの場合：**((100～2000000))** 《↑$\sqrt{\text{pd\_max\_users の値}+n}$↑×100》

n：HiRDB/パラレルサーバでマルチフロントエンドサーバ構成を適用する場合は 4。それ以外は 3。

一度使用した表と順序数生成子の定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。表と順序数生成子の定義情報は，SQL 文の前処理時に使用します。また，このバッファ内の定義情報は，LRU で管理されます。

《利点》

- 一度使用した表と順序数生成子の定義情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。
- 動的 SQL を多く使用する場合，性能が向上します。
- ディクショナリサーバとの通信回数を削減できます（HiRDB/パラレルサーバの場合）。

《指定値の目安》

- 使用頻度が高い表と順序数生成子の定義情報長の合計値を指定してください。
- 1 順序数生成子当たりの表定義情報バッファサイズは，8 キロバイトです。
- 1 表当たりの表定義情報用バッファ長の求め方については，「表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式」を参照してください。なお，一時表の場合は，非分割表と同様として計算してください。

《指定値のチューニング方法》

表定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《各見積もり式への影響》

pd_table_def_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ pd_auth_cache_size = ユーザ権限情報用バッファ長 ～＜符号なし整数＞((1～100))《10》（単位：キロバイト）

- 0904 互換モードの場合：**《1》**

ユーザ権限情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

- ユーザ権限情報用バッファには CONNECT 権限，DBA 権限，及び監査権限の情報を格納します。このバッファに情報がない場合は HiRDB 接続時にディクショナリ表から情報を取得するため，応答時間が遅くなります。したがって，常時接続しているユーザ数分の情報を格納できるようにバッファ長を指定してください。
- 1 ユーザ当たりのユーザ権限情報を格納するのに 80 バイト必要になります。それを基に計算してください。

《指定値のチューニング方法》

ユーザ権限情報用バッファ長のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《各見積もり式への影響》

pd_auth_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ pd_view_def_cache_size = ビュー解析情報用バッファ長 ～＜符号なし整数＞（単位：キロバイト）

- 32 ビットモードの場合：**((0～32000))** 《↑$\sqrt{(\text{pd\_max\_users の値}+n)}$↑×8》
- 64 ビットモードかつ推奨モードの場合：**((0～2000000))《30000》**
- 64 ビットモードかつ 0904 互換モードの場合：**((0～2000000))** 《↑$\sqrt{(\text{pd\_max\_users の値}+n)}$↑×8》

n：HiRDB/パラレルサーバでマルチフロントエンドサーバ構成を適用する場合は 4。それ以外は 3。

ビュー解析情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《利点》

一度使用したビュー解析情報を共用メモリ内に格納して，次回使用するときは入出力なしで使用できます。

《指定値の目安》

使用頻度が高いビュー表のビュー解析情報用バッファ長の合計値を指定してください。ビュー解析情報用バッファ長の見積もりについては，「ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ビュー解析情報用バッファ長のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《各見積もり式への影響》

pd_view_def_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ pd_type_def_cache_size = ユーザ定義型情報用バッファ長 ～<符号なし整数>《0》（単位：キロバイト）

- 32 ビットモードの場合：**((100～65536))**
- 64 ビットモードの場合：**((100～2000000))**

このオペランドは，ユーザ定義型に関するオペランドです。ユーザ定義型を使用する場合は，このオペランドを指定することをお勧めします。

ユーザ定義型の情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。ユーザ定義型の情報を使用すると，その情報がこのバッファ内に格納されます。この情報は SQL 文の前処理時に使用します。

《利点》

- このバッファにユーザ定義型の情報を格納すると，このユーザ定義型の情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。
- 動的 SQL を多く使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

使用頻度が高い表に定義してあるユーザ定義型の定義情報長を合計します。一つ当たりのユーザ定義型の定義情報長は，次に示す計算式から求めてください。

↑｛((0.3 ＋ 0.2× a ＋ 0.1× b ）＋ 3 ）÷ 4｝↑× 4

（単位：キロバイト）

a：ユーザ定義型の属性数

b：スーパタイプを継承したサブタイプ数

《指定値のチューニング方法》

ユーザ定義型情報用バッファ長のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《各見積もり式への影響》

pd_type_def_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ **pd_routine_def_cache_size = ルーチン定義情報用バッファ長**

～＜符号なし整数＞《300》（単位：キロバイト）

- 0904 互換モードの場合：**《100》**
- 32 ビットモードの場合：**((0，20～65536))**
- 64 ビットモードの場合：**((0，20～2000000))**

次に示す定義情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。これらの定義情報は SQL 文の前処理時に使用します。

- プラグイン関数の定義情報
- システム定義スカラ関数の定義情報
- ルーチンの定義情報

《利点》

- このバッファに前記の定義情報を格納すると，これらの情報を次回使用するときに，ディクショナリ表をアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。

《適用基準》

次に示す SQL を多く使用する場合に指定します。

- プラグインを使用する SQL
- システム定義スカラ関数を使用する SQL
- ルーチンを使用する SQL

《指定値の目安》

このオペランドの指定値の求め方については，「ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式」を参照してください。

《指定値のチューニング方法》

ルーチン定義情報用バッファ長のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

このオペランドの指定値が全プラグイン提供関数の定義情報の合計値より小さい場合，プラグイン提供関数の定義情報はバッファ中に確保されません。

《各見積もり式への影響》

pd_routine_def_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ pd_registry_cache_size = レジストリ情報用バッファ長

～＜符号なし整数＞((0～65536))《10》（単位：キロバイト）

このオペランドは，プラグインに関するオペランドです。レジストリ情報を使用するプラグインを使用する場合は，このオペランドを指定することをお勧めします。HiRDB Text Search Plug-in を使用する場合は，指定することをお勧めします。

レジストリ情報を格納するバッファ（共用メモリ）の大きさを，キロバイト単位で指定します。レジストリ情報を使用すると，レジストリ情報がバッファ内に格納されます。レジストリ情報は SQL 文の実行時に使用します。

《利点》

- このバッファにレジストリ情報を格納すると，レジストリ情報を次回使用するときに，レジストリをアクセスする必要がなくなります。そのため，入出力回数の削減や，CPU 使用時間を短縮できます。
- ディクショナリサーバとの通信回数が削減できます。
- レジストリ情報を多く使用するプラグインを使用する場合，このオペランドを指定すると性能が向上します。

《指定値の目安》

レジストリ情報用バッファ長は，次に示す計算式から求めてください。

↑（0.3＋a）↑×b　　（単位：キロバイト）

a：レジストリキー長の平均長（単位：キロバイト）

レジストリキー値の平均長は次に示す SQL で求められます。

この SQL の結果の単位はバイトのため，キロバイトに変換してください。

```
SELECT AVG(KEY_LENGTH) FROM MASTER.SQL_REGISTRY
```

b：レジストリキーの登録数

レジストリキーの登録数は次に示す SQL で求められます。

```
SELECT COUNT(*) FROM MASTER.SQL_REGISTRY
```

《指定値のチューニング方法》

レジストリ情報用バッファ長のチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《各見積もり式への影響》

pd_registry_cache_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」

9.2.20 共用メモリに関するオペランド

◆ **pd_shmpool_attribute = free | fixed | hugepage**

HiRDB のユニットコントローラが使用する共用メモリをメモリ上に固定するかどうかを指定します。なお，AIX 版の HiRDB（32 ビットモード）では共用メモリの固定化をサポートしていないため，このオペランドに fixed を指定しても無効になります。

free：

メモリ上に固定しません。実メモリ量によっては，別の共用メモリのページングが発生し，性能に影響する場合があります。

fixed：

メモリ上に固定します。

hugepage：

この指定値は Linux 版限定の指定値です。

Linux の Hugepage 機能を用いてメモリ上に固定します。

《利点》

HiRDB が使用する共用メモリをメモリ上に固定（fixed 又は hugepage 指定）すると，共用メモリのページングを防止できます。このため，共用メモリに対するアクセス性能が向上します。

Linux の Hugepage 機能を用いてメモリ上に固定（hugepage 指定）すると，共用メモリのページサイズが通常のページサイズ（4KB）から hugepages のページサイズ（2MB）に拡大されます。ページサイズが拡大すると，仮想アドレスを実アドレスに変換するための管理領域（PTE：Page Table Entry）を通常よりも小さくできるため，メモリ不足を防止できます。また，仮想アドレスと物理アドレスの変換回数が少なくなり，ページフォルトの発生を抑えることができます。これによって，トランザクション性能が向上することがあります。

《指定値の目安》

算出した共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。HiRDB がメモリを固定できるサイズに上限があるため，メモリを固定するとほかのメモリのページングが多発することがあります。したがって，実メモリや全使用量に対する共用メモリプールの占める割合をよく検討する必要があります。

ユニットコントローラが使用する共用メモリの計算式，及び各サーバが使用する共用メモリについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかの機能との関連》

AIX 版で fixed を指定する場合，次に示す制限があるため，64 ビットモードの HiRDB サーバと 32 ビットモードのクライアントプロセスとの間ではプロセス間メモリ通信機能を使用できません。

- OS のオペレーティングシステムパラメタの設定が必要
- システム共通定義に環境変数 putenv EXTSHM ON を設定できない

OS のオペレーティングシステムパラメタ，環境変数 EXTSHM の詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「AIX 固有のパラメタ指定」を参照してください。

《注意事項》

AIX 版で fixed を指定する場合の注意事項を次に示します。

- OS のオペレーティングシステムパラメタの設定をしてください。OS のオペレーティングシステムパラメタの設定をしないと，このオペランドの指定は無効になります。設定する OS のオペレーティングシステムパラメタの詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「AIX 固有のパラメタ指定」を参照してください。
- 共用メモリがメモリ上に固定されているかどうかは，AIX のコマンドで確認できます。詳細については，AIX のマニュアルを参照してください。

Linux 版で hugepage を指定する場合，Linux の Hugepage 機能を有効にする必要があります。Hugepage 機能については，OS のマニュアルを参照してください。Hugepage 機能を有効にするときに設定する OS のオペレーティングシステムパラメタの詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「Linux の Hugepage 機能の指定」を参照してください。

◆ **pd_shmpool_control = unit ｜ server**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB が使用する共用メモリの制御方法を指定します。

unit：

HiRDB/パラレルサーバをユニットごとに開始，終了する場合に適した制御方法です。

server：

HiRDB/パラレルサーバをユーザサーバごとに開始，終了する場合（pdstart 又は pdstop コマンドに-s オプションを指定する場合）に適した制御方法です。

なお，ここでいうユーザサーバとは，フロントエンドサーバ，ディクショナリサーバ，及びバックエンドサーバを指しています。

《指定値の目安》

- 通常は，このオペランドを省略するか，又は unit を指定してください。
- 影響分散スタンバイレス型系切り替え機能を適用しないシステムで，ユーザサーバの開始及び終了操作をする場合は，次に説明する「■共用メモリの確保及び解放の仕組み」を参照し，このオペランドに server を指定するかどうかを検討してください。

■共用メモリの確保及び解放の仕組み

HiRDB が使用する共用メモリは，ユニットの開始時に OS から一括確保し，HiRDB が共用メモリプール上で管理します。ユーザサーバの開始時，そのユーザサーバで使用する分の共用メモリは，共用メモリプールから複数回に分けて確保されます。

また，ユーザサーバの終了時には，ユーザサーバの開始時に確保した共用メモリを解放します。

このため，HiRDB/パラレルサーバの開始及び終了操作をユーザサーバごとに行っている場合，共用メモリプール内で領域の断片化が発生することがあります。領域の断片化が進むと，ユーザサーバの開始時に共用メモリが確保できないためにユニットが異常終了することがあります。このオペランドに server を指定すると，断片化が発生しても，ユーザサーバの開始時に必要な共用メモリが確保できるようになります。

なお，影響分散スタンバイレス型系切り替え機能を適用しているユニットの場合，ユーザサーバの開始及び終了操作を行っても，共用メモリプール内で領域の断片化が発生しないように HiRDB が制御しています。そのため，影響分散スタンバイレス型系切り替え機能を適用したユニット内のバックエンドサーバに対してだけ，開始及び終了操作を行う場合は，このオペランドに server を指定する必要はありません。

◆ **pd_dbbuff_attribute = free ｜ fixed ｜ hugepage**

グローバルバッファが使用する共用メモリを実メモリ上に固定するかどうかを指定します。なお，AIX 版の HiRDB（32 ビットモード）では共用メモリの固定化をサポートしていないため，このオペランドに fixed を指定しても無効になります。

free：

グローバルバッファが使用する共用メモリを実メモリにページ固定しません。

fixed：

グローバルバッファが使用する共用メモリを実メモリにページ固定します。

hugepage：

この指定値は Linux 版限定の指定値です。

Linux の Hugepage 機能を用いて，グローバルバッファが使用する共用メモリを実メモリにページ固定します。

《利点》

グローバルバッファが使用する共用メモリを実メモリにページ固定（fixed 又は hugepage 指定）すると，共用メモリのページングを防止できます。このため，共用メモリに対するアクセス性能が向上します。

Linux の Hugepage 機能を用いてメモリ上に固定（hugepage 指定）すると，共用メモリのページサイズが通常のページサイズ（4KB）から hugepages のページサイズ（2MB）に拡大されます。ページサイズが拡大すると，仮想アドレスを実アドレスに変換するための管理領域（PTE：Page Table Entry）を通常よりも小さくできるため，メモリ不足を防止できます。また，仮想アドレスと物理アドレスの変換回数が少なくなり，ページフォルトの発生を抑えることができます。これによって，トランザクション性能が向上することがあります。

《指定値の目安》

- 実メモリに余裕があり，性能を重視する場合は fixed 又は hugepage を指定してください。
- 算出した共用メモリサイズとサーバマシンの実メモリから，メモリを固定するかどうかを決定してください。実メモリに比べて多大なメモリをページ固定すると，ページングの多発や仮想メモリ不足の原因になります。したがって，実メモリや全仮想メモリに対する共用メモリプールの占める割合を検討する必要があります。グローバルバッファが使用する共用メモリの計算式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- ページ固定は，実メモリサイズからページ固定する共用メモリサイズを減算したサイズが，スワップ領域サイズからページ固定する共用メモリサイズを減算したサイズの 1/2 以上にならないことを目安にしてください。

《ほかの機能との関連》

- fixed 又は hugepage を指定した場合，動的変更したグローバルバッファが使用する共用メモリも実メモリ上に固定されます。そのため，実メモリサイズを十分考慮してグローバルバッファを追加又は変更してください。
- AIX 版で fixed を指定する場合，次に示す制限があるため，64 ビットモードの HiRDB サーバと 32 ビットモードのクライアントプロセスとの間ではプロセス間メモリ通信機能を使用できません。

  ・OS のオペレーティングシステムパラメタの設定が必要

  ・システム共通定義に環境変数 putenv EXTSHM ON を設定できない

  OS のオペレーティングシステムパラメタ，環境変数 EXTSHM の詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「AIX 固有のパラメタ指定」を参照してください。

《注意事項》

AIX 版で fixed を指定する場合の注意事項を次に示します。

- OS のオペレーティングシステムパラメタの設定をしてください。OS のオペレーティングシステムパラメタの設定をしないと，このオペランドの指定は無効になります。設定する OS のオペレーティングシステムパラメタの詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「AIX 固有のパラメタ指定」を参照してください。
- 共用メモリがメモリ上に固定されているかどうかは，AIX のコマンドで確認できます。詳細については，AIX のマニュアルを参照してください。

Linux 版で hugepage を指定する場合，Linux の Hugepage 機能を有効にする必要があります。Hugepage 機能については，OS のマニュアルを参照してください。Hugepage 機能を有効にするときに設定する OS のオペレーティングシステムパラメタの詳細については，マニュアル「HiRDB システム導入・設計ガイド」の「Linux の Hugepage 機能の指定」を参照してください。

◆ pd_sds_shmpool_size = シングルサーバ用共用メモリサイズ ～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：**((1～200000))**
- 64 ビットモードの場合：**((1～4000000000))**

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。ユニットコントローラ用の共用メモリの中で，シングルサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。省略した場合，HiRDB がこのオペランドの値を自動で計算します。その際，関連する定義の指定値を基に値を計算するため，関連する定義には適切な値を指定してください。関連する定義については，マニュアル「HiRDB システム導入・設計ガイド」に記載されている「シングルサーバが使用する共用メモリの計算式」を参照してください。

計算式中の変数の説明に現れるオペランドの値を途中で変更した場合は，このオペランドの値を HiRDB が自動的に再計算します。

また，計算式中の変数「インデクス用のグローバルバッファプール数」，及び「グローバルバッファ総数（pdbuffer オペランドの指定数）」は，32 ビットモードの場合は 500，64 ビットモードの場合は 1000 が仮定されます。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

このオペランドの省略時，関連する定義に適切な値が指定されていないと，HiRDB が自動計算した値が必要以上に大きくなって共用メモリをむだに確保してしまったり，逆に小さ過ぎて次に示す問題が発生したりします。

- HiRDB を開始できない
- UAP 又はユティリティが実行できない

また，このオペランドに直接指定した値が，必要以上に大き過ぎたり，小さ過ぎたりした場合も同様の問題が発生します。

《各見積もり式への影響》

pd_sds_shmpool_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」

◆ pd_dic_shmpool_size = ディクショナリサーバ用共用メモリサイズ ～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：**((1～200000))**
- 64 ビットモードの場合：**((1～4000000000))**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。ユニットコントローラ用の共用メモリの中で，ディクショナリサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。省略した場合，HiRDB がこのオペランドの値を自動で計算します。その際，関連する定義の指定値を基に値を計算するため，関連する定義には適切な値を指定してください。関連する定義については，マニュアル「HiRDB システム導入・設計ガイド」に記載されている「ディクショナリサーバが使用する共用メモリの計算式」を参照してください。

計算式中の変数の説明に現れるオペランドの値を途中で変更した場合は，このオペランドの値を HiRDB が自動的に再計算します。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

このオペランドの省略時，関連する定義に適切な値が指定されていないと，HiRDB が自動計算した値が必要以上に大きくなって共用メモリをむだに確保してしまったり，逆に小さ過ぎて次に示す問題が発生したりします。

- ユニットを開始できない
- UAP 又はユティリティが実行できない

また，このオペランドに直接指定した値が，必要以上に大き過ぎたり，小さ過ぎたりした場合も同様の問題が発生します。

◆ pd_bes_shmpool_size = バックエンドサーバ用共用メモリサイズ ～<符号なし整数>（単位：キロバイト）

- 32 ビットモードの場合：**((1～200000))**
- 64 ビットモードの場合：**((1～4000000000))**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。ユニットコントローラ用の共用メモリの中で，バックエンドサーバが使用する領域の大きさをキロバイト単位で指定します。

《指定値の目安》

通常はこのオペランドを省略してください。省略した場合，HiRDB がこのオペランドの値を自動で計算します。その際，関連する定義の指定値を基に値を計算するため，関連する定義には適切な値を指定してください。関連する定義については，マニュアル「HiRDB システム導入・設計ガイド」に記載されている「バックエンドサーバが使用する共用メモリの計算式」を参照してください。

計算式中の変数の説明に現れるオペランドの値を途中で変更した場合は，このオペランドの値をHiRDB が自動的に再計算します。

《指定値のチューニング方法》

次に示すメッセージが出力された場合は，このオペランドの指定値を大きくしてください。

- KFPA20003-E
- KFPD00005-E
- KFPD00012-E
- KFPD00021-E
- KFPH20003-E

《注意事項》

このオペランドの省略時，関連する定義に適切な値が指定されていないと，HiRDB が自動計算した値が必要以上に大きくなって共用メモリをむだに確保してしまったり，逆に小さ過ぎて次に示す問題が発生したりします。

- ユニットを開始できない
- UAP 又はユティリティが実行できない

また，このオペランドに直接指定した値が，必要以上に大き過ぎたり，小さ過ぎたりした場合も同様の問題が発生します。

◆ SHMMAX 共用メモリセグメントサイズの上限値

～＜符号なし整数＞（単位：メガバイト）((6～4194304))《1024》

グローバルバッファ用の共用メモリセグメントサイズの上限値をメガバイト単位で指定します。

HiRDB はこのオペランドの指定値を上限として，グローバルバッファ用の共用メモリセグメントを確保します。サーバマシン内の RD エリアに割り当てるグローバルバッファサイズの合計がこのオペランドの値を超える場合は，複数の共用メモリセグメントを割り当てます。

HiRDB は，HiRDB 開始時に 1 サーバ当たり最大 512 セグメントのグローバルバッファプール用の共用メモリを確保できます。グローバルバッファプールが使用している共用メモリセグメントの情報は pdls -d mem コマンドで確認できます。pdls -d mem コマンドについては，マニュアル「HiRDB コマンドリファレンス」を参照してください。

《指定値の目安》

- 32 ビットモードの場合

  通常，このオペランドを指定する必要はありません。ただし，HiRDB 開始時に KFPH23005-E メッセージ（エラーコード 24）を出力して，HiRDB が開始に失敗する場合だけ指定するようにしてください。詳細は，KFPH23051-W メッセージの対策を参照してください。

  KFPH23005-E メッセージのエラーコード 20 が出力される場合，マニュアル「HiRDB メッセージ」を参照して対策してください。

- 64 ビットモードの場合

  (1)次の条件をどちらも満たす場合，このオペランドを指定する必要はありません。

(a)すべてのグローバルバッファの面数（pdbuffer -n）の指定値が次の条件を満たしている。

・影響分散スタンバイレス型系切り替え機能を適用しない場合：1620000 以下

・影響分散スタンバイレス型系切り替え機能を適用する場合※：1420000 以下

注※　ホスト BES 数と受入可能なゲスト BES の最大数の合計が 6 以下である必要があります。

(b)グローバルバッファプール用共用メモリのサイズが 512GB 以下

(2)グローバルバッファプール用共用メモリの管理領域長が 1GB より大きいグローバルバッファがある場合，すべてのグローバルバッファで一番大きい管理領域長以上の値を指定してください。

グローバルバッファプール用共用メモリの管理領域長の計算式については，マニュアル「HiRDB システム導入・設計ガイド」の「グローバルバッファが使用する共用メモリの計算式」を参照してください。

(3)グローバルバッファプール用共用メモリのサイズが 512GB より大きい場合，次の式を満たす値を指定してください。

（SHMMAX オペランドの指定値）×512≧（グローバルバッファプール用共用メモリのサイズ）

《オペランドの規則》

指定範囲外の値を指定した場合は省略値が適用されます。

《注意事項》

- SHMMAX の指定値には，グローバルバッファプール用共用メモリの管理領域長より大きい値を指定してください。大きい値を指定しないと，HiRDB を開始できません。グローバルバッファプール用共用メモリの管理領域長の計算式については，マニュアル「HiRDB システム導入・設計ガイド」の「グローバルバッファが使用する共用メモリの計算式」を参照してください。
- グローバルバッファの動的変更を適用している場合，SHMMAX の指定値は HiRDB の再開始時に変更できません。
- Linux の場合，共用メモリセグメント数は次のオペレーティングシステムパラメタの制限を受けます。

  shmmax，shmmni，及び shmall

  このため，グローバルバッファ用の共用メモリセグメントを複数確保する場合，この制限を超えて共用メモリを確保できないことがあります。この場合，次に示す優先順位で対処をしてください。

  1.shmmni 及び shmseg（Linux の場合は shmall）の値を大きくする

  2.shmmax の値を SHMMAX オペランドの値より大きくする

  3.グローバルバッファプール用の共用メモリのサイズが小さくなるように，グローバルバッファ数を見積もり直す

  OS のオペレーティングシステムパラメタについては，OS のマニュアルを参照してください。また，指定値の見積もりについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_dbbuff_modify
- pdbuffer
- pd_max_add_dbbuff_no
- pd_max_add_dbbuff_shm_no

《各見積もり式への影響》

SHMMAX オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「インメモリデータ処理で必要なメモリ所要量」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「インメモリデータ処理で必要なメモリ所要量」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「Linux のオペレーティングシステムパラメタの見積もり」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

9.2.21 RD エリアに関するオペランド

◆ **pd_max_rdarea_no = RD エリアの最大数**

～＜符号なし整数＞((5～8388592))《1000》

- 0904 互換モードの場合：**《200》**

RD エリアの最大数を指定します。RD エリアの合計数がこのオペランドの値を超えている場合，HiRDB を正常開始できません。なお，ここでいう RD エリアには，マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアも含まれます。

《指定値の目安》

- RD エリアの合計数以上の値を指定してください（ある程度余裕をもたせてください）。RD エリアの合計数は pddbls コマンドで確認できます。なお，RD エリアを追加する予定がある場合は，その数を加算した値を指定してください。
- HiRDB/パラレルサーバの場合，このオペランドの値はバックエンドサーバごとに適用されます。例えば，このオペランドに 100 を指定した場合，各バックエンドサーバに最大 100 個の RD エリアを作成できます。したがって，RD エリア数が最も多いバックエンドサーバを目安にしてこのオペランドを指定してください。

- インナレプリカ機能を使用している場合，レプリカ RD エリアの数も加算してください。
- 共用 RD エリアを使用している場合，参照専用バックエンドサーバで使用する共用 RD エリアの数も加算してください。

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《各見積もり式への影響》

pd_max_rdarea_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 2」，「計算式 3」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 2」
- 「データベース構成変更ユティリティ（pdmod）実行時のメモリ所要量」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 2」，「計算式 3」
- 「フロントエンドサーバが使用する共用メモリの計算式」

◆ pd_max_file_no = RD エリアを構成する HiRDB ファイルの最大数 ～＜符号なし整数＞((5～134217728))《2000》

- 0904 互換モードの場合：**《400》**

RD エリアを構成する HiRDB ファイルの最大数を指定します。HiRDB ファイルの合計数がこのオペランドの値を超えている場合，HiRDB を正常開始できません。なお，ここでいう RD エリアを構成する HiRDB ファイルには，マスタディレクトリ用 RD エリア，データディレクトリ用 RD エリア，及びデータディクショナリ用 RD エリアの HiRDB ファイルも含まれます。

《指定値の目安》

- RD エリアを構成する HiRDB ファイルの合計数以上の値を指定してください（ある程度余裕をもたせてください）。各 HiRDB ファイルシステム領域内の HiRDB ファイルの数は pdfstatfs コマンドで確認できます。なお，HiRDB ファイルを追加する予定がある場合は，その数を加算した値を指定してください。RD エリアの追加，RD エリアの再初期化，及び RD エリアの拡張時に HiRDB ファイルを追加します。

- HiRDB/パラレルサーバの場合，このオペランドの値はバックエンドサーバごとに適用されます。例えば，このオペランドに 100 を指定した場合，各バックエンドサーバに最大 100 個の HiRDB ファイルを作成できます。したがって，HiRDB ファイル数が最も多いバックエンドサーバを目安にしてこのオペランドを指定してください。
- インナレプリカ機能を使用している場合，レプリカ RD エリアを構成する HiRDB ファイルの数も加算してください。
- 共用 RD エリアを使用している場合，参照専用バックエンドサーバで使用する共用 RD エリアを構成する HiRDB ファイルの数も加算してください。

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《各見積もり式への影響》

pd_max_file_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 2」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」

◆ pd_rdarea_warning_point = セグメント使用比率 1〔，セグメント使用比率 2〔，セグメント使用比率 3〕〕

～＜符号なし整数＞((0～100))（単位：%）

RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I），又は増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）の出力契機を指定します。

なお，LOB 用 RD エリアの使用中の容量を監視したい場合は pd_stored_lob_data_wrn_pnt オペランドの説明も参照してください。

LOB 用 RD エリアのセグメント使用率通知メッセージ（KFPH22040-W）の出力契機は pd_stored_lob_data_wrn_pnt オペランドで指定できます。

RD エリアの自動増分を適用しない場合

セグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I）が出力されます。

このオペランドの指定の有無，及び対象となる RD エリア種別ごとのセグメント使用率通知メッセージの内容を次に示します。

対象となる RD エリア	pd_rdarea_warning_point 省略時	pd_rdarea_warning_point 指定時
• マスタディレクトリ用 RD エリア • データディレクトリ用 RD エリア	メッセージを出力しません。	対象となる RD エリアの全体のセグメントのうち，指定された使用率のセグメントの使用を開始しました。
• データディクショナリ用 RD エリア • レジストリ用 RD エリア • ユーザ用 RD エリア • リスト用 RD エリア	対象となる RD エリアの最終ファイルのセグメントのうち，相対位置で 80，90，100%のどれかのセグメントの使用を開始しました。	対象となる RD エリアの全体のセグメントのうち，指定された使用率のセグメントの使用を開始しました。
• LOB 用 RD エリア	対象となる RD エリアの最終ファイルのセグメントのうち，相対位置で 80，90，100%のどれかのセグメントの使用を開始しました。	指定された相対位置に当たるセグメントの使用を開始しました。

RD エリアの自動増分を適用する場合

自動増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）が出力されます。

自動増分する HiRDB ファイルの領域使用率通知メッセージは，自動増分を実行した場合に，このオペランドのセグメント使用比率以上に領域を割り当てたときに出力するメッセージです。

RD エリアの自動増分を適用する場合，このオペランドの指定の有無と出力するメッセージの内容を次に示します。

オペランドの指定値	pd_rdarea_warning_point 省略時	pd_rdarea_warning_point 指定時
メッセージの出力条件	増分した HiRDB ファイルの領域は，使用率 80，90，100%のどれかの比率以上に領域を割り当てました。	増分した HiRDB ファイルの領域は，指定された使用率以上に領域を割り当てました。

例えば，このオペランドの省略時で使用率 80%のメッセージが出力された場合，増分前の領域使用率は 80%未満で，増分後の領域使用率が 80%以上 90%未満になったことを表します。

自動増分する HiRDB ファイルの領域使用率は，増分できる最大容量に対する使用率です。次の計算式で求めます。

```
自動増分するHiRDBファイルの領域使用率（単位：%）＝A÷B×100
```

1. pdfmkfs -a コマンド指定の通常ファイルの HiRDB ファイルシステム領域の場合

   A：自動増分する HiRDB ファイルの容量

   B：自動増分できる最大容量＝ min｛(A ＋ C)，64GB｝

   C：自動増分する HiRDB ファイルがある HiRDB ファイルシステム領域の空き容量＋自動増分する HiRDB ファイルがある HiRDB ファイルシステム領域が存在するディスクの空き容量

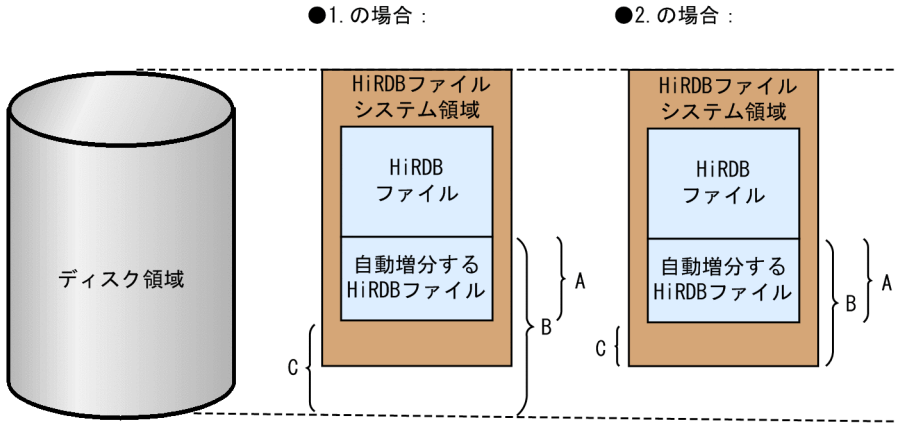
2. pdfmkfs -e コマンド指定の HiRDB ファイルシステム領域，又は pdfmkfs -a 指定のキャラクタ型スペシャルファイル上の HiRDB ファイルシステム領域の場合

   A：自動増分する HiRDB ファイルの容量

   B：自動増分できる最大容量＝ min｛(A ＋ C)，64GB｝

   C：自動増分する HiRDB ファイルがある HiRDB ファイルシステム領域の空き容量

上記の 1 と 2 で，自動増分できる範囲が異なります。自動増分できる範囲を次の図に示します。



《自動増分する HiRDB ファイルの領域使用率通知メッセージに関する注意事項》

- 一回の自動増分で複数のセグメント使用比率を超えた場合は，超えたセグメント使用比率の中で最も大きいセグメント使用比率に対するメッセージを出力します。
- 次の要因により空き容量が減少し，増分前の領域使用率がセグメント使用比率を超えた場合は，そのセグメント使用比率に対するメッセージを出力しません。
  - ・別の RD エリアで自動増分が発生したとき
  - ・データベース構成変更ユティリティ(pdmod コマンド)でデータベースの物理構成を変更したとき
  - ・HiRDB 以外の製品によるファイル追加

《指定例》

メッセージの出力条件	オペランドの指定値
対象となる RD エリアの RD エリア全体のセグメントのうち，80，90，100%分の使用を開始した，それぞれの時点でメッセージを出力する場合	pd_rdarea_warning_point = 80,90,100
対象となる RD エリアの RD エリア全体のセグメントのうち，50，90%分の使用を開始した，それぞれの時点でメッセージを出力する場合	pd_rdarea_warning_point = 50,90
セグメント使用率通知メッセージを出力しない場合	pd_rdarea_warning_point = 0

《オペランドの規則》

- この指定値は，三つまで指定できます。
- 同じ値を複数指定した場合には，その値に対して 1 回だけメッセージが出力されます。
- セグメント使用比率 1～3 に 0 だけを指定した場合には，セグメント使用率通知メッセージが出力されません。
- 0 と 0 以外の数値が混在している場合には，指定された 0 以外の数値をメッセージの出力契機とします。
- セグメント使用率通知メッセージ（KFPA12300-I）は，クライアント環境定義 PDEXWARN に YES を指定した場合にだけ出力されます。

《注意事項》

次のユティリティでは，このオペランドの指定を無視し，オペランド指定省略時の動作となります。

- データベース初期設定ユティリティ（pdinit コマンド）
- データベース回復ユティリティ（pdrstr コマンド）

  なお，データベース回復ユティリティでは，次の場合はコマンド実行中に再度のメッセージ出力はしません。

  1.バックアップだけを使用した回復で，バックアップ取得時点で既にメッセージを出力済み

  2.ログを使用した回復で，指定したログの最終時点までで既にメッセージを出力済み

◆ pd_rdarea_warning_point_msgout = Y | N

RD エリアのセグメント使用率通知メッセージ（KFPH00211-I，又は KFPA12300-I）を出力するかどうかを指定します。

Y：pd_rdarea_warning_point の指定値に従って，RD エリアのセグメント使用率通知メッセージを出力します。

N：RD エリアのセグメント使用率通知メッセージを出力しません。

《指定値の目安》

RD エリアのセグメント使用率通知メッセージの指定値の目安については，マニュアル「HiRDB システム導入・設計ガイド」の「空き領域の再利用機能」の「環境設定」を参照してください。

《注意事項》

次の場合には，このオペランドにNを指定してもセグメント使用率通知メッセージを出力します。

- データベース作成ユティリティ（pdload），又はデータベース再編成ユティリティ（pdrorg）でデータを格納する場合
- 空き領域の再利用機能を適用していない表にデータを格納する場合
- インデクスにデータを格納する場合
- 空き領域の再利用機能を適用している表の構成列に可変長の列を含み，データ長を長くする更新を頻繁に行う場合
- 表の定義数が pd_assurance_table_no オペランドの指定値を超えたため，空き領域の再利用機能が使用できなくなっている場合

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_rdarea_warning_point
- pd_assurance_table_no

◆ pd_stored_lob_data_wrn_pnt = セグメント使用比率 1〔，セグメント使用比率 2〔，セグメント使用比率 3〕〕

～＜符号なし整数＞((0～100))《80，90，100》（単位：%）

LOB 用 RD エリアのセグメント使用率通知メッセージ（KFPH22040-W）の出力契機を指定します。

なお，LOB 用 RD エリアの空き容量がなくなると LOB データが追加できなくなるため，LOB 用 RD エリアのセグメント使用率通知メッセージ（KFPH22040-W）で使用中の容量を監視してください。

このオペランドの指定例を次に示します。

表 9–6　pd_stored_lob_data_wrn_pnt の指定例

メッセージの出力条件	オペランドの指定値
LOB 用 RD エリアが 50，80，95%のセグメントを使用した，それぞれの時点でメッセージを出力する場合	pd_stored_lob_data_wrn_pnt = 50,80,95
LOB 用 RD エリアが 50，90%のセグメントを使用した，それぞれの時点でメッセージを出力する場合	pd_stored_lob_data_wrn_pnt = 50,90
LOB 用 RD エリアのセグメント使用率通知メッセージを出力しない場合	pd_stored_lob_data_wrn_pnt = 0

《オペランドの規則》

- この指定値は，三つまで指定できます。
- 0 を除く同じ値を複数指定した場合には，その値に対して 1 回だけメッセージが出力されます。
- セグメント使用率 1〜3 に 0 だけを指定した場合には，セグメント使用率通知メッセージが出力されません。
- 0 と 0 以外の数値が混在している場合には，指定された 0 以外の数値をメッセージの出力契機とします。

《注意事項》

- このオペランドの指定は，pdinit コマンド，pdrstr コマンドに対しては LOB 用 RD エリアのセグメント使用率通知メッセージを出力しません。
- プラグインインデクスを使用している場合，プラグインインデクスの作成・再初期化時に LOB 用 RD エリアのセグメント使用率通知メッセージが出力されることがあります。

  ・LOB 用 RD エリアのセグメント使用率通知メッセージを出力したくない場合は，このオペランドに 0 を指定してください。

  ・LOB 用 RD エリアのセグメント使用率通知メッセージの監視レベルを下げたい場合は，syslogfile に出力するメッセージの重要度を変更してください。syslogfile に出力するメッセージの重要度の変更についての詳細は，「メッセージの出力抑止機能に関するオペランド」の説明，及び，マニュアル「HiRDB システム運用ガイド」の「syslogfile に出力するメッセージの重要度の変更」を参照してください。
- LOB 用 RD エリアの最終セグメントを使用した後に LOB データを追加，又は更新すると KFPH00213-W メッセージを出力して先頭セグメントから空き領域サーチする（セグメントオーバ状態となる）ため，領域内の空き状況によっては LOB データの更新性能が低下することがあります。

  ・LOB データの追加，又は更新での性能低下契機を監視する場合は，セグメント使用率通知メッセージ（KFPH00211-I）及び自動増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）を監視してください。セグメント使用率通知メッセージ（KFPH00211-I）及び

自動増分する HiRDB ファイルの領域使用率通知メッセージ（KFPH22037-W）の出力契機については，pd_rdarea_warning_point の説明を参照してください。

・セグメントオーバ状態かどうかについては，データベース状態解析ユティリティ（pddbst）で確認してください。データベース状態解析ユティリティ（pddbst）については，マニュアル「HiRDB コマンドリファレンス」を参照してください。

◆ pd_rdarea_extension_timing = use ｜ nouse

- 0904 互換モードの場合：《nouse》

RD エリアの自動増分機能を適用している場合，どのタイミング（契機）で RD エリアを自動増分するかを指定します。

use：RD エリア内の空きセグメント数が 1 回の自動増分で増分するセグメント数未満になったときに自動増分します。例えば，1 回の自動増分セグメント数を 50 セグメントとしている場合，空きセグメント数が 50 セグメント未満になったときに自動増分します。

nouse：RD エリア内に空きセグメントがなく，新しいセグメントを確保できないときに自動増分します。

《指定値の目安》

それぞれの指定値のメリットとデメリットを次の表に示します。

項目	オペランドの指定値	
	use	nouse
メリット	自動増分ができない場合※でも，データ追加によるページ不足エラーになるまで，空きセグメント（増分セグメント数分）を利用できます。この間に，RD エリアの状態を解析し，使用状況に応じて再編成や RD エリアの拡張などの対処ができるため，業務への影響を最小限にできます。	空きセグメントがなくなってから，自動増分を行うため，HiRDB ファイルシステム領域内に複数 HiRDB ファイルを作成する場合，格納効率が良くなります。
デメリット	HiRDB ファイルシステム領域内に複数 HiRDB ファイルを作成する場合，格納効率が悪くなることがあります。	自動増分ができない場合※，空きセグメントがない状態のため，データ追加によるページ不足エラーで業務が停止するおそれがあります。

注※　次のような場合，自動増分できません。

- HiRDB ファイルシステム領域内で HiRDB ファイルを拡張する方式（pdfmkfs -e コマンド）の場合，HiRDB ファイルシステム領域に空きがないとき。
- HiRDB ファイルシステム領域を自動的に拡張する方式（pdfmkfs -a コマンド）の場合，ディスクに空きがない，又は HiRDB ファイルの最大サイズ（64GB）に達しているとき。
- RD エリアがバックアップ閉塞状態のため，自動増分が抑止されるとき。

このオペランドの指定値は，業務への影響を最小限にできる use をお勧めします。

◆ pd_rdarea_expand_format ＝ Y ｜ N

RD エリアの自動増分が発生した場合に増分領域を初期化するかどうかを指定します。通常ファイルで，pdfmkfs コマンドに-a オプションを指定して増分した HiRDB ファイルシステム領域に対して指定できます。

Y：増分領域を初期化します。

N：増分領域を初期化しません。

《指定値の目安》

このオペランドにYを指定すると，増分領域を初期化するためディスクへの入出力が発生します。ディスクへの入出力処理に時間が掛かるため，次のトランザクションが遅延します。

- 初期化しているセグメントを更新するトランザクション
- 自動増分で新規セグメントを割り当てるトランザクション

そのため，通常はこのオペランドの指定値をNにすることをお勧めします。

ただし，RDエリアなどの容量を監視できる運用を行えるかどうかによってお勧めする指定値が変わります。詳細については，マニュアル「HiRDBシステム運用ガイド」の「RDエリアの自動増分を適用する場合の運用」を参照してください。

《注意事項》

このオペランドの指定値をYにした場合，自動増分が発生しているRDエリアをアクセスする更新トランザクションの実行時間が延びるため，次のどちらかの要因でトランザクションがキャンセルされることがあります。

- 排他待ち限界経過時間を超過した
- HiRDBクライアントの最大待ち時間やユティリティの最大実行時間を超過した

トランザクションがキャンセルされることを避けるため，《ほかのオペランドとの関連》に示すオペランドについて，次に示すRDエリアの構成情報，及びHiRDBファイルシステム領域を配置しているディスクの入出力性能を考慮して，増分領域の初期化に掛かる処理時間の目安を秒単位で見積もってください。

- ページサイズ
- セグメントサイズ
- 増分セグメント数

```
増分領域の初期化に掛かる処理時間の目安（単位：秒）＝
ページサイズ（バイト）×（セグメントサイズ×増分セグメント数＋α）÷HiRDBファイルシステム領域を配置しているディスクのI/O性能（バイト/秒）
```

α：増分時に作成するディレクトリページ数

```
・データディクショナリ用RDエリア，ユーザ用RDエリア，又はレジストリ用RDエリアの場合
  ↑d÷b↑＋↑d÷f↑
・LOB用RDエリアの場合
  ↑S÷64000↑×96
```

d：増分セグメント数

b：↓(P−20)÷(↑S÷32↑×8 ＋ 56)↓

f：↓(125×P)÷(16×b)↓×b

P：ページサイズ

S：セグメントサイズ

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- PDCWAITTIME
- pd_lck_wait_timeout
- pd_cmd_exec_time
- pd_utl_exec_time

◆ **pd_rdarea_extension_file = all ｜ last**

RD エリアの自動増分機能を使用している場合に，自動増分対象 HiRDB ファイルを指定します。

all：RD エリアを構成する HiRDB ファイルのすべてを自動増分の対象とします。

last：RD エリアを構成する HiRDB ファイルの内，最終ファイルを自動増分の対象とします。

この機能を適用した場合，pddbst コマンド（-f 指定）を実行すると，HiRDB ファイル単位に「自動増分抑止状態」,「自動増分エラーコード」を追加で表示します。確認方法については，マニュアル「HiRDB コマンドリファレンス」の「pddbst」を参照してください。

《指定値の目安》

推奨値は all です。

all を指定すると，RD エリア作成時に最大構成ファイル数の 16 ファイルを最小ファイルサイズで設置することで，RD エリア拡張なしで最小サイズから最大サイズ（64GB×16 ファイル≒1TB）まで増分できます。

詳細については，マニュアル「HiRDB システム運用ガイド」の「RD エリアの自動増分」を参照してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_rdarea_extension_timing

《注意事項》

この機能は，ユーザ用 RD エリアにだけ適用されます（一時表用 RD エリアを除く）。

◆ **pd_rdarea_open_attribute_use = Y ｜ N**

RD エリアのオープン契機に DEFER 属性又は SCHEDULE 属性を使用するかどうかを指定します。

Y：DEFER 属性又は SCHEDULE 属性を使用します。

N：DEFER 属性又は SCHEDULE 属性を使用しません。

このオペランドを省略するか，又は N を指定した場合，RD エリアのオープン契機は常に INITIAL 属性になります。したがって，次に示すオペランド又はユティリティで RD エリアのオープン契機を DEFER 属性又は SCHEDULE 属性にしても無効になります。

- pd_rdarea_open_attribute オペランド
- データベース初期設定ユティリティ
- データベース構成変更ユティリティ

《注意事項》

- Y を指定すると，HiRDB が必要とする共用メモリ量が増加するため，共用メモリが不足して HiRDB を開始できないことがあります。

- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドに Y が仮定されます。このため，サーバが使用する共用メモリが増加するので共用メモリを見積もり直してください。サーバが使用する共用メモリの見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《各見積もり式への影響》

pd_rdarea_open_attribute_use オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバで起動するプロセス」
- 「HiRDB/パラレルサーバで起動するプロセス」
- 「シングルサーバが使用する共用メモリの計算式」
- 「バックエンドサーバが使用する共用メモリの計算式」

◆ pd_rdarea_open_attribute = INITIAL ｜ DEFER ｜ SCHEDULE

RD エリアのオープン契機の標準値を指定します。

データベース初期設定ユティリティ又はデータベース構成変更ユティリティで open attribute の指定がない RD エリアは，このオペランドに指定した属性が仮定されます。ただし，システム用 RD エリアは，常に INITIAL が仮定されます。

《指定値の目安》

- 同じ HiRDB ファイルシステム領域の RD エリアには，同じ属性を指定してください。異なる属性を指定すると，期待どおりの効果が得られない場合があります。
- 各属性の RD エリアのオープン／クローズ契機，及び長所／短所を次に示します。

属性	初期状態	オープン契機	クローズ契機	長所	短所
INITIAL	オープン状態	• HiRDB 開始時 • pdopen コマンド実行時	pdclose コマンド実行時	初回 SQL から高速実行	システムの開始に時間が掛かる
DEFER	クローズ状態	• RD エリアの初回アクセス時 • pdopen コマンド実行時	pdclose コマンド実行時	• システムの開始が高速 • 初回アクセス以後は通常 SQL も高速	各 RD エリアの初回アクセスに時間が掛かる
SCHEDULE	クローズ状態	• トランザクション内での RD エリア初回アクセス時 • pdopen コマンド実行時	• トランザクション終了時 • pdclose コマンド実行時	• システムの開始が高速 • ファイルオープンの集中を回避	• トランザクションごとに RD エリアの初回アクセスが高負荷となる • リアルタイム SAN レプリケーションの処理方

属性	初期状態	オープン契機	クローズ契機	長所	短所
					式がハイブリッド方式の場合，トランザクション終了時に 1 トランザクション当たり 2 秒以上のオーバヘッドが掛かる

- 各属性に適する運用形態を次に示します。

属性	適する運用形態
INITIAL	システム開始時に HiRDB ファイルシステム領域をオープンして，RD エリア情報をメモリ上に常駐します。RD エリアの初回アクセス時にも，そのプロセス上でオープンします。ただし，この場合には RD エリア情報の再作成をしないため，初回 SQL から高速な運用ができます。 システム開始時の RD エリア初期状態はオープン状態であり，以降は障害閉塞への遷移を除いて，運用コマンドの入力がないかぎり RD エリアの状態は遷移しません。 **特殊な運用形態を用いない場合は，この属性を推奨します。** この属性のとき，クローズ状態の RD エリアに対してはアクセスできません。
DEFER	システム開始時に HiRDB ファイルシステム領域のオープンをしないで，RD エリアに対する初回アクセス時にオープンし，RD エリア情報をメモリ上に常駐します。2 回目以降のアクセスでは，HiRDB ファイルシステム領域のオープン以降の処理をしないため，高速な運用ができます。 システム開始時の RD エリア初期状態はクローズ状態であり，各 RD エリアに対する初回アクセス時に該当する RD エリアをオープン状態にします。以降は障害閉塞への遷移を除いて，運用コマンドの入力がないかぎり RD エリアの状態は遷移しません。 **多数の HiRDB ファイルシステム領域に対するオープンが重なるケースを回避したい場合や，HiRDB の開始時間を短縮したい場合に，この属性を指定します。** HiRDB を再開始する場合は，回復処理時に回復対象 RD エリアをオープンします。 この属性を指定した場合，クローズ状態の RD エリアに対してもアクセスできます。
SCHEDULE	システム開始時に HiRDB ファイルシステム領域のオープンをしないで，システム開始後，各トランザクション内での RD エリアに対する初回アクセス時にオープンし，RD エリア情報をメモリ上に常駐します。トランザクションの終了時に，そのトランザクション内でオープンした HiRDB ファイルシステム領域をクローズします。以降もトランザクションが変わると RD エリアに対する初回アクセス時にオープン以降の処理をするため，トランザクションに掛かる負荷は増加します。 システム開始時の RD エリアの初期状態はクローズ状態であり，アクセスのあった RD エリアのトランザクション処理中だけオープン状態とします。トランザクション終了時に，トランザクション内でオープン状態としたすべての RD エリアをクローズ状態にします。 pdopen コマンドを入力すると，次回閉塞クローズ状態になるまでの間オープン状態を継続できます。そのほかの運用コマンドを用いて，RD エリアのステータスを任意に遷移させることもできます。障害を検知したときは障害閉塞となります。 **多数の HiRDB ファイルシステム領域のオープンが重なることを回避したい場合や，システムの開始時間を短縮したい場合に，この属性を指定します。** HiRDB を再開始する場合は，回復処理時に回復対象 RD エリアをオープンして，回復処理の終了後にクローズします。 この属性を指定した場合，クローズ状態の RD エリアに対してもアクセスできます。

属性	適する運用形態
	リアルタイム SAN レプリケーションの処理方式がハイブリッド方式の場合，リモートサイトへのデータベースの同期待ち合わせを行います。トランザクション終了時に，アクセスした RD エリア数×2 秒以上のオーバヘッドが掛かることがあります。

《注意事項》

- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合の注意事項を次に示します。

使用する機能	注意事項
高速系切り替え機能	高速系切り替え機能の対象になる待機系ユニットは，待機状態のときに RD エリアをオープンしていません。また，系の切り替え時間を最小限に抑えるため，系切り替えの発生時に全面回復で必要な RD エリアだけをオープンして，そのほかの RD エリアはオープンしません。したがって，待機系の RD エリアのオープン契機は INITIAL 属性になりません。INITIAL 属性の RD エリアは DEFER 属性になります。
1：1 スタンバイレス型系切り替え機能	系の切り替え時間を最小限に抑えるため，系切り替えの発生時に全面回復で必要な RD エリアだけをオープンして，そのほかの RD エリアはオープンしません。したがって，正規 BES 又は代替部の RD エリアのオープン契機は次に示すようになります。 • 系切り替えが発生した場合，代替部の RD エリアのオープン契機は SCHEDULE 属性になります。 • 障害が回復して正規 BES に系を切り戻した場合，正規 BES 下の INITIAL 又は DEFER 属性の RD エリアのオープン契機は DEFER 属性になります。SCHEDULE 属性の RD エリアは SCHEDULE 属性のままです。
影響分散スタンバイレス型系切り替え機能	系の切り替え時間を最小限に抑えるため，系切り替えによる再開始時には全面回復で必要な RD エリアだけをオープンして，そのほかの RD エリアはオープンしません。したがって，系切り替えが発生した場合，ゲスト BES 下の INITIAL 属性の RD エリアのオープン契機は DEFER 属性になります。

- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能と RD エリアのオープン契機の関係を次に示します。

<table>
<tr><th colspan="3" rowspan="3">条件</th><th colspan="4">pd_rdarea_open_attribute_use の指定値</th></tr>
<tr><th rowspan="2">N</th><th colspan="3">Y</th></tr>
<tr><th colspan="3">RD エリアのオープン契機</th></tr>
<tr><th>INITIAL</th><th>DEFER</th><th>SCHEDULE</th></tr>
<tr><td colspan="3">系切り替え機能を使用しない※1</td><td rowspan="3">INITIAL</td><td rowspan="3">INITIAL</td><td rowspan="3">DEFER</td><td rowspan="3">SCHEDULE</td></tr>
<tr><td rowspan="2">スタンバイ型系切り替え機能※1</td><td colspan="2">高速系切り替え機能を使用しない</td></tr>
<tr><td rowspan="2">高速系切り替え機能</td><td>実行系</td></tr>
<tr><td>待機系</td><td>DEFER※2</td><td>DEFER</td><td>DEFER</td><td>SCHEDULE</td></tr>
<tr><td rowspan="2">1：1 スタンバイレス型系切り替え機能※1</td><td rowspan="2">受け入れ部（代替 BES ユニットの代替部以外）</td><td>実行系</td><td>INITIAL
※2，※3</td><td>INITIAL
※3</td><td>DEFER</td><td>SCHEDULE</td></tr>
<tr><td>待機系</td><td>DEFER※2</td><td>DEFER</td><td>DEFER</td><td>SCHEDULE</td></tr>
</table>

条件			pd_rdarea_open_attribute_use の指定値			
			N	Y		
				RD エリアのオープン契機		
				INITIAL	DEFER	SCHEDULE
	代替部	実行系	SCHEDULE ※2	SCHEDULE	SCHEDULE	SCHEDULE
		待機系				
影響分散スタンバイレス型系切り替え機能※1	正常開始又は再開始		INITIAL	INITIAL	DEFER	SCHEDULE
	系切り替えによる再開始		DEFER※2	DEFER	DEFER	SCHEDULE

注※1　システム用 RD エリアのオープン契機は INITIAL になります。

注※2　pd_rdarea_open_attribute_use オペランドに Y が指定されているとみなします。

注※3　再開始時は DEFER になります。

◆ **pd_shared_rdarea_use = Y ｜ N**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

共用 RD エリアを使用するかどうかを指定します。

Y：共用 RD エリアを使用します。

N：共用 RD エリアを使用しません。

《注意事項》

- 共用 RD エリアを定義しているのに，このオペランドを省略した場合（又はこのオペランドに N を指定した場合），HiRDB を正常開始できません。
- このオペランドに Y を指定すると，pd_dbsync_point オペランドに commit を指定したときと同じ共用メモリブロックを確保します。

《各見積もり式への影響》

pd_shared_rdarea_use オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「バックエンドサーバが使用する共用メモリの計算式」

◆ **pd_db_access_error_action = dbhold ｜ unitdown**

RD エリアへのアクセス時にファイルアクセスエラーが発生した場合の動作を指定します。

RD エリアにアクセスする際，アクセス対象の RD エリアを構成する HiRDB ファイルシステム領域にあるファイルを HiRDB 以外のプロセスが占有排他でオープンしていると，ファイルアクセスエラーとなります（エラーコード-1540※）。

マスタディレクトリ用 RD エリアの場合はファイルアクセスエラー検知時ユニットダウン機能を使用していなくても，障害閉塞しないでユニットダウンします。ただし，ユニットダウンする際には KFPH23040-I メッセージが出力されます。

注※　HiRDB ファイルシステム領域のアクセス権限の設定を誤った場合にも返却されます。

dbhold：

ファイルアクセスエラーが発生した場合，アクセス対象の RD エリアを閉塞します。

unitdown：

ファイルアクセスエラーが発生した場合，KFPH23040-I メッセージを出力して，ファイルアクセスエラー検知時ユニットダウン機能を適用します。ファイルアクセスエラー検知時ユニットダウン機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《適用基準》

通常，このオペランドを指定する必要はありません。

HiRDB の稼働中に HiRDB 以外のプロセスが RD エリアを構成する HiRDB ファイルシステム領域にアクセスする運用を行っていて，そのプロセスが HiRDB ファイルシステム領域を占有排他する可能性がある場合に，このオペランドの指定を検討してください。

《注意事項》

- unitdown を指定した場合にファイルアクセスエラーが発生すると，次のケースでは処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している

  物理エラー検知時ユニットダウン機能を使用する場合は，できるだけこれらの運用は避けてください。もし，これらの運用が必要な場合は，RD エリアが閉塞しても最新の状態に回復できるよう，UAP 又はユティリティの実行前にバックアップを取得してください。バックアップの取得については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- ログ同期方式のリアルタイム SAN レプリケーション機能を適用している場合，ログ適用サイトの HiRDB の開始中は，unitdown を指定しても無効になります。
- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していてもユニットダウンしません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

《ほかのオペランドとの関連》

- pd_mode_conf オペランド
- pd_db_io_error_action オペランド
- pd_db_hold_action オペランド

なお，pd_db_io_error_action オペランド，pd_db_access_error_action オペランド，及び pd_db_hold_action オペランドのうち，複数のオペランドで unitdown を指定した場合は，次の順序でオペランドの指定値が有効になります。

1. pd_db_io_error_action オペランド
2. pd_db_access_error_action オペランド
3. pd_db_hold_action オペランド

そのため，RD エリアの入出力エラー，ファイルアクセスエラー，及び物理エラーのうち複数のエラーが同時に起こった場合，どのエラー要因によってユニットダウンしたのかは，この優先順位を参考に判断してください。また，出力されたメッセージを参照してください。

◆ **pd_db_hold_action = dbhold ｜ unitdown**

RD エリアへのアクセス時に物理エラーが発生した場合（KFPH00306-E メッセージの閉塞理由に i/o error occurred，又は open error occurred を出力して閉塞するエラーの場合）の動作を指定します。マスタディレクトリ用 RD エリアの場合は物理エラー検知時ユニットダウン機能を使用していなくても，障害閉塞しないでユニットダウンします。物理エラー検知時ユニットダウン機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

dbhold：

物理エラーが発生した場合，アクセス対象の RD エリアを閉塞します。

unitdown：

物理エラーが発生した場合，KFPH23047-I メッセージを出力して，物理エラー検知時ユニットダウン機能を適用します。

《適用基準》

通常，このオペランドを指定する必要はありません。

RD エリアの数が少なく，RD エリアが一つでも障害閉塞したら業務全体が停止するようなシステムの場合に，このオペランドの指定を検討してください。

unitdown を指定する場合は，pd_mode_conf オペランドに MANUAL2 を指定してください。

《注意事項》

- RD エリアに対するアクセスで物理エラーが発生し，KFPH00307-E メッセージを出力して RD エリアがコマンド閉塞した場合は，pd_db_hold_action オペランドに unitdown を指定していてもユニットダウンしません。
- unitdown を指定した場合に物理エラーが発生すると，次のケースでは処理対象の RD エリアが障害閉塞することがあります。

  ・更新前ログ取得モード又はログレスモードで UAP やユティリティを実行している

  ・CREATE TABLE の RECOVERY オペランドで NO を指定してログレスモードにしたユーザ LOB 用 RD エリアに対して，UAP 又はユティリティを実行している

  物理エラー検知時ユニットダウン機能を使用する場合は，できるだけこれらの運用は避けてください。もし，これらの運用が必要な場合は，RD エリアが閉塞しても最新の状態に回復できるよう，UAP 又はユティリティの実行前にバックアップを取得してください。バックアップの取得については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- ログ同期方式のリアルタイム SAN レプリケーション機能を適用している場合，ログ適用サイトの HiRDB の開始中は，unitdown を指定しても無効になります。
- データベース回復ユティリティ（pdrstr）での回復処理中は，unitdown を指定していてもユニットダウンしません。この場合は，データベース回復ユティリティ（pdrstr）を再度実行して回復してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_mode_conf オペランド
- pd_db_io_error_action オペランド
- pd_db_access_error_action オペランド

なお，pd_db_io_error_action オペランド，pd_db_access_error_action オペランド，及び pd_db_hold_action オペランドのうち，複数のオペランドで unitdown を指定した場合は，次の順序でオペランドの指定値が有効になります。

1. pd_db_io_error_action オペランド
2. pd_db_access_error_action オペランド
3. pd_db_hold_action オペランド

そのため，RD エリアの入出力エラー，ファイルアクセスエラー，及び物理エラーのうち複数のエラーが同時に起こった場合，どのエラー要因によってユニットダウンしたのかは，この優先順位を参考に判断してください。また，出力されたメッセージを参照してください。

◆ **pd_dbbuff_read_retry_count＝RD エリアのページ読み込み時に不正を検知した際の IO リトライ回数～＜符号なし整数＞((0～65535)) <<3>>（単位：回）**

RD エリアへのアクセス時に，ページの読み込み内容に不整合を検出した場合（KFPH23202-W メッセージ出力時）の読み込みリトライ回数を指定します。

このオペランドに 0 を指定した場合，リトライは行いません。

1～3 を指定した場合は，即時リトライします（間隔 0 秒）。

4 以上を指定した場合に，3 回目までは即時リトライし，4 回目以降は pd_dbbuff_read_retry_intvl オペランドの指定値の間隔でリトライします。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《注意事項》

RD エリアへのアクセス時にページの読み込み内容に不整合を検出した場合，保守情報としてメッセージ（KFPH00313-E）を出力します。メッセージは，最大で pd_dbbuff_read_retry_count 指定値+1 回出力します。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_dbbuff_read_retry_intvl
- pd_db_timestamp_invalid_action

◆ **pd_dbbuff_read_retry_intvl＝RD エリアのページ読み込み時に不正を検知した際の IO リトライ間隔～＜符号なし整数＞((0～65535)) <<0>>（単位：秒）**

RD エリアへのアクセス時に，ページの読み込み内容に不整合を検出した場合（KFPH23202-W メッセージ出力時）の読み込みリトライ間隔を指定します。

pd_dbbuff_read_retry_count オペランドに 4 以上を指定した場合だけ，4 回目以降のリトライ処理で有効となり，3 回目までは pd_dbbuff_read_retry_intvl オペランドの指定に関係なく即時リトライします。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《注意事項》

- 読み込みリトライに要する最大時間は，次の式で求めることができます。

  [(MAX(3, pd_dbbuff_read_retry_count 指定値 - 3)) × pd_dbbuff_read_retry_intvl 指定値]（単位：秒）

  この値は次の指定値を超えないようにしてください。

  クライアント環境定義の PDCWAITTIME

  クライアント環境定義の PDLCKWAITTIME（省略時はシステム定義の pd_lck_wait_timeout オペランド）

- PDCWAITTIME によるタイムアウトが発生した場合，トランザクションはリトライ処理が完了するまでキャンセルされません。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_dbbuff_read_retry_count
- pd_db_timestamp_invalid_action

◆ pd_db_timestamp_invalid_action = dbhold ｜ unitdown

RD エリアへのアクセス時にページの読み込み内容に不整合を検出し（KFPH23202-W メッセージ出力時），読み込みリトライが終了してもページの読み込み内容の不整合が解消しなかった場合の動作を指定します。

dbhold：

アクセス対象の RD エリアを閉塞します。

unitdown：

KFPH23054-I メッセージを出力して，閉塞しないでユニットダウンさせます。

ページ不正が解消しなかった場合の HiRDB の動作を次に示します。

RD エリアの種類	HiRDB の動作（pd_db_timestamp_invalid_action の指定値）	
	dbhold（省略値）	unitdown
マスタディレクトリ用 RD エリア	RD エリアを閉塞しないでユニットダウンします。	
その他の RD エリア	RD エリアを閉塞します。	RD エリアを閉塞しないでユニットダウンします。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《注意事項》

- pd_db_timestamp_invalid_action オペランドに unitdown を指定して，HiRDB を自動的に再開始する設定を行っている場合，ページ読み込みリトライの結果，ページの読み込み内容の不整合が解消しないでユニットダウンした後，自動的に再開始します。しかし，再開始処理の失敗による異常終了が 30 分以内に 3 回※1※2 発生すると，KFPS00715-E メッセージを出力して再開始処理が中断されます。再開始処理が中断した場合は，メッセージの対処に従って HiRDB の再開始処理を行ってください。

  注※1　系切り替え機能を適用している場合，実行系と待機系の間で異常終了回数のカウントを引き継ぎません。

  注※2　UNIX の場合は pd_term_watch_count 指定値（1～3）の回数になります。
- データベース回復ユティリティ（pdrstr）及びデータベース複写ユティリティ（pdcopy）は，pd_db_timestamp_invalid_action オペランドの指定が無効になります。ページの読み込み内容に不整合を検出した場合，各ユティリティが異常終了するため，出力されたメッセージの対処に従って，再度ユティリティを実行してください。
- ログ同期方式のリアルタイム SAN レプリケーション機能を適用している場合，ログ適用サイトでは pd_db_timestamp_invalid_action オペランドに unitdown を指定しても無効になります。ページの読み込み内容に不整合を検出した場合，該当ボリュームに配置した RD エリアを障害閉塞し，障害閉塞していない RD エリアに対してだけログ適用を継続します。システムログ適用化の実施を検討してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_dbbuff_read_retry_count
- pd_dbbuff_read_retry_intvl

9.2.22　グローバルバッファに関するオペランド

◆ pd_dbbuff_lru_option = SEPARATE ｜ MIX

グローバルバッファの LRU 管理方式を指定します。ただし，次のどちらかに該当する場合は無条件に MIX を仮定します。

- pd_dbsync_point に commit を指定している場合
- pd_dbbuff_binary_data_lru に N を指定している場合

SEPARATE：

参照バッファ，及び更新バッファをそれぞれ独立した LRU で管理します。グローバルバッファの不足時には，グローバルバッファ内のアクセスした参照バッファの中で，最も古いバッファがメモリから追い出されます。オンライン業務のように 1 トランザクション当たりの参照，更新件数が比較的少ない場合に指定します。

MIX：

グローバルバッファを一括した LRU で管理します。グローバルバッファの不足時には，グローバルバッファ内のアクセスしたバッファで，最も古いバッファがメモリから追い出されます。オンライン業務とバッチ業務など，大量検索，大量更新が共存する場合に指定します。

グローバルバッファの LRU 管理方式については，マニュアル「HiRDB 解説」を参照してください。

◆ **pd_dbbuff_binary_data_lru = Y | N**

BINARY 型を指定した列にアクセスする UAP を実行する場合，グローバルバッファの LRU 管理を行うかどうかを指定します。UAP がアクセスするバイナリデータの LRU 管理抑止設定については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

Y：

バイナリデータが格納された分岐行ページに LRU 管理を適用して，グローバルバッファ上にキャッシュします。

N：

バイナリデータが格納された分岐行ページに LRU 管理を適用しないで，アクセス頻度に関係なく最も古くアクセスしたページとしてグローバルバッファ上にキャッシュします。

ただし，基本行ページは LRU 管理を適用してグローバルバッファ上にキャッシュします。

グローバルバッファを使用してサイズが大きいバイナリデータに大量にアクセスする UAP を実行する場合，バイナリデータをグローバルバッファにキャッシュするために，グローバルバッファにキャッシュされた直近の内容がメモリから追い出され，一時的に性能を低下させてしまうことがあります。その場合，サイズが大きいバイナリデータのアクセス頻度が低いのであれば，LRU 管理を抑止することによって性能低下を回避できます。

それぞれの指定値による LRU 管理，メリット及びデメリットを次の表に示します。

項目	オペランドの指定値	
	Y	N
LRU 管理	すべてのグローバルバッファを LRU で管理し，グローバルバッファが不足したときは，グローバルバッファプール内で最も古くアクセスしたグローバルバッファをメモリから追い出します。	バイナリデータが格納された分岐行ページのグローバルバッファの LRU 管理を抑止し，最も古くアクセスしたページとしてキャッシュします。グローバルバッファが不足したときは，バイナリデータが格納された分岐行ページのグローバルバッファをメモリから追い出します。
メリット	アクセスデータの種類に関係なく，グローバルバッファを均等に使用するため，UAP の性能が偏りません。	サイズが大きいバイナリデータに連続してアクセスしても，グローバルバッファからバイナリデータを優先的に追い出すため，バイナリデータ以外の検索性能を維持できます。
デメリット	サイズが大きいバイナリデータに連続してアクセスするとグローバルバッファから直近にアクセスしたデータが追い出されるため，バイナリデータ以外の検索性能が低下します。	• サイズが大きいバイナリデータにアクセスする UAP は，バッファヒット率が低くなります。それによって，入出力回数が増加し，レスポンス性能が低下する可能性があります。

項目	オペランドの指定値	
	Y	N
		• バイナリデータの更新時に，バイナリデータの分岐行の追い出しが多発することでシステムログ量が増加します。

《指定値の目安》

- 通常はこのオペランドを指定しないでください。
- バイナリデータにアクセスしないで，蓄積するだけのシステムの場合は，N を指定してください。バイナリデータに頻繁にアクセスするシステムの場合，このオペランドに N を指定するとデメリットの影響が大きいので適用するときは十分検討してください。
- BINARY 型や BINARY 型の属性を含む抽象データ型，及び XML 型などのサイズの大きなバイナリデータを含む表が存在するシステムの場合，サイズの大きなバイナリデータにほとんどアクセスしないときは，このオペランドに N を指定してください。バイナリデータが格納されたページをアクセス頻度に関係なく優先的にグローバルバッファから追い出し対象とするため，バイナリデータ以外のデータ格納ページがグローバルバッファから追い出される頻度を少なくできます。

《クライアント環境定義との関連》

クライアント環境定義の PDDBBUFLRU に NO を指定した場合は，このオペランドの指定に関係なく，指定した UAP でアクセスするすべてのデータの LRU 管理を抑止します。

《ほかのオペランドとの関連》

pd_dbbuff_lru_option オペランドに SEPARATE を指定しても，このオペランドで N を指定すると，pd_dbbuff_lru_option オペランドに MIX を仮定します。

◆ **pd_dbbuff_dev_sector_size = 512 | 4096**

グローバルバッファを，論理セクタ長が 4096 バイトのディスク対応にするかどうかを指定します。

512：グローバルバッファを論理セクタ長が 4096 バイトのディスクに対応しません。

4096：グローバルバッファを論理セクタ長が 4096 バイトのディスクに対応します。

《指定値の目安》

RD エリアを格納する HiRDB ファイルシステム領域を，論理セクタ長が 4096 バイトのディスクに作成する場合に，4096 を指定してください。

AIX の場合は指定しても無視されます。

《注意事項》

このオペランドに 4096 を指定すると，グローバルバッファの共用メモリサイズが最大で pdbuffer オペランド数×4KB 増加します。詳細はマニュアル「HiRDB システム導入・設計ガイド」の「グローバルバッファが使用する共用メモリの計算式」を参照してください。

◆ **pd_dbbuff_modify = Y | N**

HiRDB の稼働中に pdbufmod コマンドでグローバルバッファを動的変更するかどうかを指定します。グローバルバッファの動的変更については，マニュアル「HiRDB システム運用ガイド」を参照してください。

Y：グローバルバッファを動的変更します。

N：グローバルバッファを動的変更しません。

《前提条件》

このオペランドに Y を指定する場合（グローバルバッファを動的変更する場合）は HiRDB Advanced High Availability が必要になります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no
- pd_max_add_dbbuff_shm_no

《各見積もり式への影響》

pd_dbbuff_modify オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 4」，「計算式 5」，「計算式 6」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 3」，「計算式 4」，「計算式 5」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 4」，「計算式 5」，「計算式 6」

◆ pd_dbbuff_rate_updpage = デファードライトトリガの要求比率 ～<符号なし整数>((1～100))（単位：%）

デファードライトトリガのトリガ契機を更新バッファ比率で指定します。

《指定値の目安》

- 通常はこのオペランドを省略してください。デファードライトの書き込み処理がシンクポイントダンプ取得間隔時間内で終了しない場合など，更新バッファ数を減らし，更新バッファヒット率

を多少抑えてでも書き込み時間を短縮したいときにこのオペランドを指定します。指定する場合は，50%（HiRDB が設定する初期値）を目安にし，マニュアル「HiRDB システム運用ガイド」の「デファードライト処理のチューニング」及び「デファードライト処理適用時のシンクポイント処理時間のチューニング」を参照してオペランドの値を決定してください。

- このオペランドを指定すると，トリガの出力契機が一定となるため，更新量が増加すると更新バッファヒット率が低下し，レスポンスが遅延することがあります。

《ほかのオペランドとの関連》

このオペランドは，pdbuffer オペランドの-y オプションと次に示す関係があります。

- pd_dbbuff_rate_updpage オペランドの指定値は全グローバルバッファに適用されます。
- pdbuffer オペランドの-y オプションの指定値はグローバルバッファごとに適用されます。
- pdbuffer オペランドの-y オプションの指定値が優先されます。
- pdbuffer オペランドの-y オプションを指定しない場合，デファードライトトリガ契機の更新バッファ面数は，このオペランドの指定によって次のように異なります。

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

◆ pd_dbbuff_trace_level = グローバルバッファ制御情報トレース取得レベル ～＜符号なし整数＞((0～2147483647))《0》

グローバルバッファ制御情報トレースの取得レベルを符号なし整数で指定します。取得レベルは，グローバルバッファ制御情報トレース取得機能の各機能に対応したレベル値を合計して算出します。0 を指定した場合，グローバルバッファ制御情報トレースを取得しません。グローバルバッファ制御情報トレース取得機能とレベル値の対応を次に示します。グローバルバッファ制御情報トレース取得機能の詳細については，マニュアル「HiRDB システム運用ガイド」を参照してください。

グローバルバッファ制御情報トレース取得機能	レベル値
シンクポイント出力同期制御情報取得機能	1

《指定値の目安》

通常は，このオペランドを指定する必要はありません。障害調査などで保守員にこのオペランドの指定を依頼された場合は，保守員の指示に従って指定してください。

《各見積もり式への影響》

pd_dbbuff_trace_level オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 5」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 4」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 5」

◆ **pdbuffer -a グローバルバッファ名**

{-r RD エリア名〔，RD エリア名〕...|
-b RD エリア名〔，RD エリア名〕...|
-o |
-i 認可識別子．インデクス識別子 }
-n バッファ面数〔- l バッファサイズ〕
〔-m 同時実行最大プリフェッチ数〕
〔-p 一括入力最大ページ数〕
〔-w デファードライトトリガ時の更新ページ出力比率〕
〔-c〕
〔-y デファードライトトリガ契機の更新バッファ面数〕
〔-D 分割数〕

グローバルバッファをどの RD エリアに割り当てるかを定義します。グローバルバッファとは，RD エリアへ表のデータを入出力するときにデータを確保する領域のことであり，共用メモリ上に確保されます。

すべての RD エリアにグローバルバッファを割り当ててください。グローバルバッファを割り当てない RD エリアに対しては，SQL 文及び pdopen コマンドを実行できません。グローバルバッファの設計方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《オペランドの規則》

pdbuffer オペランドは 2,000,000 個まで指定できます。HiRDB/パラレルサーバの場合は 1 サーバ当たり 2,000,000 個までになります。ただし，全体で 2,147,483,647 個が上限になります。

-D オプションを指定している場合，分割数分 pdbuffer 文を指定したとみなします。-D オプションを指定しない場合は 1 と数えます。

《指定値のチューニング方法》

グローバルバッファのチューニング方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《オペランドの省略値》

pdbuffer オペランドを省略すると，次に示す条件でグローバルバッファを割り当てます。

- グローバルバッファ名：gbuf00
- グローバルバッファの種別：-o オプション指定相当
- グローバルバッファサイズ：6 メガバイト
- バッファ面数：6 メガバイト÷RD エリアのページ長の最大値

《注意事項》

pdbuffer オペランドに指定する以下の値について，名称を重複指定して HiRDB を起動した場合，KFPH23002-E メッセージを出力して HiRDB の起動処理が失敗します。

- グローバルバッファ名
- RD エリア名
- インデクス識別子

なお，pdconfchk コマンドではオペランドの文法チェックはしますが，名称の重複指定に関してはチェックしません。

-a グローバルバッファ名

～＜識別子＞((1～16 文字))

グローバルバッファの名称を指定します。同じ名称は指定できません。

-D オプションを指定した場合，グローバルバッファ名長は 12 文字までになります。

-r RD エリア名〔，RD エリア名〕...

～＜識別子＞((1～30 文字))

グローバルバッファを割り当てる RD エリアの名称を指定します。次に示す RD エリアの名称を指定してください。

- マスタディレクトリ用 RD エリア
- データディクショナリ用 RD エリア
- データディレクトリ用 RD エリア
- ユーザ用 RD エリア
- データディクショナリ LOB 用 RD エリア※
- ユーザ LOB 用 RD エリア※
- レジストリ用 RD エリア
- レジストリ LOB 用 RD エリア※
- リスト用 RD エリア

注※

これらの LOB 用 RD エリアは，-b オプションの LOB 用グローバルバッファも割り当てることをお勧めします（-r オプションと-b オプションの両方を指定します）。LOB 用グローバルバッファを割り当てると性能が向上します。

《指定値の目安》

ページ長が同じか，又はページ長が近い RD エリアを同じグローバルバッファに割り当てると，入出力回数を削減できます。ただし，同じページ長の RD エリアでも，次に示すような RD エリアは別々のグローバルバッファに割り当てた方が入出力回数を削減できます。

- 用途の異なる表を格納する RD エリア
- ランダムアクセスが多い RD エリアと順次アクセスが多い RD エリア

《オペランドの規則》

- RD エリア名に半角英数字以外の文字を含む場合は引用符（"）で囲んでください。
- RD エリア名を引用符（"）で囲んだ場合は英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。
- 一つのグローバルバッファには最大 4096 個の RD エリアを定義できます。

-b RD エリア名〔，RD エリア名〕...

～＜識別子＞((1～30 文字))

LOB 用グローバルバッファを割り当てる RD エリアの名称を指定します。次に示す RD エリアの名称を指定してください。

- データディクショナリ LOB 用 RD エリア
- ユーザ LOB 用 RD エリア
- レジストリ LOB 用 RD エリア

ここで指定した LOB 用 RD エリアは，-r オプションでも指定しておく必要があります。-b オプションの指定だけでは LOB 用 RD エリアをアクセスできません。指定例を次に示します。

（例）

LOB 用 RD エリア（RDLOB01）に-r 及び-b オプションのグローバルバッファを割り当てます。

```
pdbuffer -a gbuf01 -r RDLOB01 -n 1000
pdbuffer -a gbuf02 -b RDLOB01 -n 1000
```

備考

- LOB 用 RD エリアにはディレクトリ部とデータ部があります。ディレクトリ部とデータ部はそれぞれ異なるグローバルバッファで管理しています。-r 指定のグローバルバッファにはディレクトリ部がキャッシュされ，-b 指定のグローバルバッファにはデータ部がキャッシュされます。このため，-r（又は-o）及び-b オプションを同時に指定する必要があります。
- -r 指定のグローバルバッファと-b 指定のグローバルバッファは異なる用途のグローバルバッファのため，別々に見積もる必要があります。-r 指定のグローバルバッファはディレクトリ部分だけをキャッシュするため，比較的小さくてもよいです。-b 指定のグローバルバッファは LOB のデータ部をキャッシュするため，使用可能なメモリ容量及びヒット率を考慮して見積もってください。グローバルバッファの見積もり方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《指定値の目安》

次に示すような LOB 用 RD エリアを指定します。

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

なお，一つの LOB 用グローバルバッファには一つの LOB 用 RD エリアを割り当てることをお勧めします。

《オペランドの規則》

- RD エリア名称が半角英数字以外の文字を含む場合は，引用符（"）で囲んでください。
- RD エリア名を引用符（"）で囲んだ場合は英字の大文字と小文字が区別されますが，引用符で囲まない場合はすべて大文字として扱われます。
- 一つのグローバルバッファには，最大 4096 個の LOB 用 RD エリアを定義できます。

《注意事項》

LOB 用グローバルバッファは，プリフェッチ機能及びデファードライトトリガ機能を使用しません。したがって，-m，-p，及び-w オプションを指定する必要はありません。

-o

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

-i 認可識別子．インデクス識別子

認可識別子：～＜識別子＞((1～30 文字))

インデクス識別子：～＜識別子＞((1～30 文字))

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

《指定値の目安》

使用頻度の高いインデクスを指定してください。使用頻度の高いインデクスにグローバルバッファを割り当てると，インデクスページのメモリ常駐度が上がり，入出力回数の削減効果が出ます。

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

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

《オペランドの規則》

- 「認可識別子．インデクス識別子」は，同じ名称で複数指定できません。
- 認可識別子又はインデクス識別子に半角英数字以外の文字を含む場合は，認可識別子及びインデクス識別子をそれぞれ引用符（"）で囲んでください。
- 認可識別子及びインデクス識別子をそれぞれ引用符（"）で囲んだ場合は，英字の大文字と小文字が区別されますが，引用符で囲まない場合は，すべて大文字として扱われます。

《注意事項》

- -i オプションに指定したインデクスを格納している RD エリアにもグローバルバッファを割り当てる必要があります。-r オプションで RD エリアを指定するか，又は-o オプションでグローバルバッファを割り当ててください。

RD エリアには，ディレクトリ部（ディレクトリページ），データ部（表データのページ），及びインデクス部（インデクスページ）が格納されています。インデクス用グローバルバッファにはインデクス部だけがキャッシュされるため，ディレクトリ部とデータ部をキャッシュするための-r 又は-o 指定のグローバルバッファが必要になります。

- DROP TABLE 又は DROP INDEX を実行した場合，削除したインデクスに割り当てていたインデクス用グローバルバッファは使用されなくなります。また，削除したインデクスと同じ名称のインデクスを定義しても，そのインデクス用グローバルバッファは使用できません。そのインデクス用グローバルバッファを使用するには，HiRDB を一度正常終了して再度開始する必要があります。
- HiRDB の開始時に存在しないインデクスを-i オプションに指定した場合，そのインデクス用グローバルバッファは割り当てられません。このとき，KFPH23014-W メッセージが出力されます。

  また，HiRDB の開始後，-i オプションに指定したインデクスを定義しても，インデクス用グローバルバッファは使用できません。インデクス用グローバルバッファを使用するには，HiRDB を一度正常終了して再度開始する必要があります。
- 一時インデクスに割り当てたインデクス用グローバルバッファは，HiRDB 開始時に無効になります。このとき，KFPH23014-W メッセージが出力されます。
- インデクス用グローバルバッファのバッファサイズより大きなページ長のインデクス格納 RD エリアを ALTER TABLE で追加した場合，追加した RD エリアのインデクスページはインデクス用グローバルバッファにキャッシュされません。インデクス用グローバルバッファにキャッシュするには，HiRDB を一度正常終了して再度開始する必要があります。
- データベース構成変更ユティリティの RD エリアの再初期化で，RD エリアのページ長をインデクス用グローバルバッファのバッファサイズより大きくした場合，再初期化した RD エリアのインデクスページはインデクス用グローバルバッファにキャッシュされません。インデクス用グローバルバッファにキャッシュするには，HiRDB を一度正常終了して再度開始する必要があります。

-n バッファ面数

～＜符号なし整数＞

- 32 ビットモードの場合：**((4～460000))**
- 64 ビットモードの場合：**((4～1073741824))**

グローバルバッファの面数を指定します。このオプションは必ず指定してください。

《指定値の目安》

- シンクポイント取得のタイミング，及びグローバルバッファ内の更新ページの比率によっては，データベースへの出力が集中します。そのため，入出力回数のバランスも考慮する必要があります。
- グローバルバッファの面数は，次の表から求めてください。

条件		グローバルバッファ面数の求め方
-r 又は-o オプション指定のグローバルバッファの場合※1	HiRDB/シングルサーバの場合	同時に発生する SQL 処理要求の数×1SQL で使用するページ数（3～6 程度）
	HiRDB/パラレルサーバの場合	同時実行ユーザ数×1 トランザクション内の平均同時アクセス表数×3×n※2
-b オプション指定のグローバルバッファの場合		LOB 用 RD エリアの総ページ数×常駐度※3(%)
-i オプション指定のグローバルバッファの場合※1		インデクスの格納ページ数×常駐度※4(%)

注※1

最低限必要なバッファ面数は次のとおりです。バッファ面数にはこの値以上を指定するようにしてください。この値より小さい値を指定した場合はバッファ不足による SQL エラーが発生することがあります。

- HiRDB/シングルサーバの場合

  同時に発生する SQL 処理要求の数×4

- HiRDB/パラレルサーバの場合

  同時実行ユーザ数×1 トランザクション内の平均同時アクセス表数×4

注※2

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

注※3

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

注※4

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

《指定値のチューニング方法》

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

- 統計解析ユティリティ（グローバルバッファに関する統計情報）の「更新バッファヒット率」及び「参照バッファヒット率」で確認できます。
- pdbufls コマンドのヘッダ名 HIT で確認できます。

《注意事項》

- グローバルバッファは，正しく見積った値を指定してください。必要以上に大きな値を指定すると，次の現象が発生するおそれがあります。

a. 共用メモリ確保失敗に伴う起動失敗，又はユニットダウン

b. メモリ圧迫によるページング多発に伴う性能低下

c. バッファサーチのためのオーバヘッド増加に伴う性能低下

- 同一のグローバルバッファにアクセスするユーザ数が増加し，バッファ面数が不足した場合は，SQL エラーが発生することがあります。

-l バッファサイズ

～＜符号なし整数＞((4～30 の偶数))（単位：キロバイト）

グローバルバッファのバッファ 1 面のサイズをキロバイト単位で指定します。このオプションは-n オプションと一緒に指定します。

《指定値の目安》

通常，このオプションは省略してください。このオプションを省略すると，このグローバルバッファに割り当てた RD エリアの最大ページ長がバッファサイズになります。なお，HiRDB/パラレルサーバの場合はサーバ内の RD エリアの最大ページ長がバッファサイズになるため，サーバごとにバッファサイズが異なることがあります。

また，次に示す場合は指定値の変更を考えてください。

- 将来，このオプションで指定した値を超えるページ長の RD エリアを追加したり，再初期化して RD エリアのページ長を大きくしたりすることがある場合，このオプションの指定値を大きめに設定しておくことをお勧めします。ただし，HiRDB を一度終了できる場合，HiRDB の次回開始時にはバッファサイズに最大ページ長が仮定されるため，このオプションの指定値を大きめに設定しておく必要はありません。

《オペランドの規則》

- このオプションの指定値が RD エリアの最大ページ長より小さい場合は，RD エリアの最大ページ長がバッファサイズになります。
- このオプションに奇数を指定した場合，バッファサイズは指定した値に 1 を加算した値になります。

《注意事項》

このオプションの値を変更しても，HiRDB を正常開始しないと，その値が反映されません。HiRDB を再開始した場合は，前回稼働時（変更前）のバッファサイズとなります。

-m 同時実行最大プリフェッチ数

～＜符号なし整数＞((0～95000))《0》

同時にプリフェッチ機能を使用する最大数を指定します。プリフェッチ機能は，キャラクタ型スペシャルファイルを使用して大量検索をする場合に入出力時間を削減する効果があります。

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

《指定値の目安》

指定値の設定は，このグローバルバッファに割り当てられた RD エリアの表の中で，プリフェッチ機能が適用される SQL 文又はユティリティを同時に実行する回数を指定します。プリフェッチ機能は次に示す SQL 文又はユティリティの場合にページを一括して入力します。

1. インデクスを使用しない SELECT，UPDATE，DELETE 文の場合にデータページを一括して入力します。
2. インデクスを使用した昇順検索※をする SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）の場合にインデクスリーフページを一括して入力します。
3. クラスタキーを使用した昇順検索※をする SELECT，UPDATE，DELETE 文（=条件，IN 条件を除く）の場合にインデクスリーフページ及びデータページを一括して入力します。
4. ローカルバッファを使用しないデータベース再編成ユティリティ（pdrorg）のアンロードの場合にインデクスリーフページ及びデータページを一括して入力します。

注※　複数列インデクスの場合はインデクス定義で指定した順序

《注意事項》

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

また，グローバルバッファに割り当てられた RD エリアの表の中で，プリフェッチ機能を適用した SQL 文及びユティリティを同時に実行する場合，実行する回数が指定値を超えた SQL 文及びユティリティについてはプリフェッチ機能が適用されません。

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

-p 一括入力最大ページ数

～＜符号なし整数＞((2～256))《32》

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

《指定値の目安》

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

a×b ＝ 64～128　（単位：キロバイト）

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

b：一括入力最大ページ数

-w デファードライトトリガ時の更新ページ出力比率

～＜符号なし整数＞((0～100))《20》（単位：%）

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

このオプションに 0 を指定した場合は，デファードライトトリガでの更新ページ出力をしません。

《指定値の目安》

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

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

-c

このオプションはスタンバイレス型系切り替え機能に関するオプションです。この機能を使用しない場合にはこのオプションを指定しないでください。

- 1：1 スタンバイレス型系切り替え機能の場合

  正規 BES の RD エリア又はインデクスが使用するグローバルバッファを，切り替え先の代替 BES 側にも作成する場合にこのオプションを指定します。このオプションを省略した場合，代替 BES 側にはグローバルバッファは作成されないため，代替中は代替 BES に定義している-o オプション指定のグローバルバッファを使用します。

  なお，-c 及び-o オプションの両方を省略した場合は代替 BES ユニットを開始できません。

  このオプションの指定によって代替中のグローバルバッファの割り当て方式が決まります。代替中のグローバルバッファの割り当て方式については，マニュアル「HiRDB システム運用ガイド」の「グローバルバッファの定義（1：1 スタンバイレス型系切り替え機能限定）」を参照してください。

- 影響分散スタンバイレス型系切り替え機能の場合

  影響分散スタンバイレス型系切り替えの対象となるバックエンドサーバに配置されている RD エリア又はインデクスのグローバルバッファを定義する場合はこのオプションを指定します。ここで定義したグローバルバッファは同じ HA グループ内の全ユニットに作成されます。

  なお，-c オプションを省略した場合，このグローバルバッファが割り当てられないため，このグローバルバッファを使用する RD エリアにアクセスできません。

  このオプションの指定によって影響分散スタンバイレス型系切り替え機能の対象となるバックエンドサーバの RD エリア，又はインデクスに対するグローバルバッファの割り当て方式が決まります。影響分散スタンバイレス型系切り替え機能使用時のグローバルバッファの割り当て方式については，マニュアル「HiRDB システム運用ガイド」を参照してください。

-y デファードライトトリガ契機の更新バッファ面数

～＜符号なし整数＞((2～2147483647))

デファードライトトリガのトリガ契機を更新バッファ面数で指定します。更新バッファ面数が指定値に達すると，更新ページをディスクに書き込みます。グローバルバッファごとにデファードライトトリガのトリガ契機を設定したい場合に，このオプションを指定してください。

《指定値の目安》

通常はこのオプションを省略してください。デファードライトの書き込み処理がシンクポイントダンプ取得間隔時間内で終了しない場合など，更新バッファ数を減らし，更新バッファヒット率を多

少抑えてでも書き込み時間を短縮したいときにこのオプションを指定します。指定する場合は，50%（HiRDB が設定する初期値）を目安にし，マニュアル「HiRDB システム運用ガイド」の「デファードライト処理適用時のシンクポイント処理時間のチューニング」を参照してオペランドの値を決定してください。

《オペランドの規則》

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

《注意事項》

このオプションの指定値が小さ過ぎると，デファードライト処理の回数が増え，デファードライト処理による負荷を高めるため，指定値の目安に従った設定をお勧めします。

《ほかのオペランドとの関連》

このオペランドは，pd_dbbuff_rate_updpage オペランドと次に示す関係があります。

- pd_dbbuff_rate_updpage オペランドの指定値は全グローバルバッファに適用されます。
- pdbuffer オペランドの-y オプションの指定値はグローバルバッファごとに適用されます。
- pdbuffer オペランドの-y オプションの指定値が優先されます。
- pdbuffer オペランドの-y オプションを指定しない場合，デファードライトトリガ契機の更新バッファ面数は，pd_dbbuff_rate_updpage オペランドの指定によって次のように異なります。

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

-D 分割数

～<符号なし整数>((0,2～1000))

-n 指定したバッファ面数を分割して管理します。分割したグローバルバッファごとに処理をシリアライズするため，グローバルバッファの競合を低減できます。

0 を指定すると，バッファ面数が 10 万以下になるように分割します（分割数=↑バッファ面数/100000↑）。分割数が 1000 以上になる場合は，1000 に分割します。-n に指定したバッファ面数が 10 万以下の場合，グローバルバッファは分割しません。

2～1000 を指定した場合，指定した分割数でグローバルバッファを分割します。-n に指定したバッファ面数が分割数×4 未満の場合，-n に分割数×4 を指定したとみなします。

アクセスするページの RD エリア ID，ファイル番号，及びページ番号から求めたハッシュ値で，分割したグローバルバッファのうちどれを使用するかが決まります。

《指定値の目安》

通常はこのオプションを省略してください。分割する場合は，10 から 20 程度を目安に指定してください。

《注意事項》

- LOB 用グローバルバッファ(-b)には-D オプションを指定できません。指定した場合，-D オプションは無効になります。
- -D オプションを指定する場合，-a オプションに指定するグローバルバッファ名は 12 文字以下にしてください。13 文字以上の場合は HiRDB 起動時にエラーになり，HiRDB は起動できません。
- -D オプションを指定すると，pdbufls，グローバルバッファの統計情報のグローバルバッファ名称に「:xxx」（xxx は 000～999 の数値をつけたグローバルバッファ名称）で分割ごとに出力します。
- グローバルバッファが使用する共用メモリ計算式は，↑バッファ面数÷分割数↑のバッファ面数を指定した分割数分の pdbuffer を指定したとして計算してください。
- -D オプションを指定した場合，-p，-m は指定できません。-p 及び-m を指定しても無効になります。
- -n オプションに指定したバッファ面数が分割数×4 未満の場合，-n の指定値は分割数×4 を指定したとみなします。
- -D オプションに 1 を指定した場合，指定は無効になります。

《各見積もり式への影響》

pdbuffer オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「グローバルバッファが使用する共用メモリの計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」，「計算式 4」，「計算式 5」，「計算式 6」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 3」，「計算式 4」，「計算式 5」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 4」，「計算式 5」，「計算式 6」

《各見積もり式への影響》

pdbuffer オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「グローバルバッファが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「グローバルバッファが使用する共用メモリの計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 1」,「計算式 4」,「計算式 5」,「計算式 6」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」,「計算式 3」,「計算式 4」,「計算式 5」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」,「計算式 4」,「計算式 5」,「計算式 6」
- 「メッセージキュー及びセマフォ所要量の見積もり」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

◆ **pd_max_add_dbbuff_no = 動的追加用グローバルバッファ数の上限値**
～<符号なし整数>((1～32752))

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加するグローバルバッファ数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

- pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，その数に余裕を持たせた値を指定してください。
- 次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_max_add_dbbuff_no の値≦2000000−HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《オペランドの省略値》

このオペランドの省略値を次に示します。

条件		省略値
32 ビットモードの場合	a ≧500 の場合	256
	a ＜ 500 の場合	500− a
64 ビットモードの場合	a ≧1000 の場合	256
	a ＜ 1000 の場合	1000− a

a：HiRDB 開始時に各サーバに割り当てられているグローバルバッファ数

《注意事項》

このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《ほかのオペランドとの関連》

このオペランドは次のオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_shm_no

《各見積もり式への影響》

pd_max_add_dbbuff_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「シングルサーバが使用する共用メモリの計算式」の，「計算式 2」，「計算式 4」，「計算式 5」，「計算式 6」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 3」，「計算式 4」，「計算式 5」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」，「計算式 4」，「計算式 5」，「計算式 6」

◆ **pd_max_add_dbbuff_shm_no = 動的追加用共用メモリセグメント数の上限値 ～<符号なし整数>((1～32752))**

グローバルバッファの動的変更をする場合にこのオペランドを指定します。

このオペランドには，pdbufmod コマンドで動的追加したときに割り当てられる，共用メモリセグメント数（1 サーバ当たり）の上限値を指定します。

《前提条件》

pd_dbbuff_modify オペランドに Y を指定している必要があります。

《指定値の目安》

pdbufmod コマンドで動的追加するグローバルバッファ数を予測し，値を指定してください。

《オペランドの省略値》

このオペランドの省略値を次に示します。

条件		省略値
pd_max_add_dbbuff_no オペランドを省略した場合	32 ビットモードの場合	500 ＋ A
	64 ビットモードの場合	1000 ＋ A

条件		省略値
pd_max_add_dbbuff_no オペランドを指定した場合	32 ビットモードの場合	↓pd_max_add_dbbuff_no の値×1.5 ＋ A↓ （計算値が 32752 以上だった場合は自動的に 32752 が設定されます）
	64 ビットモードの場合	

A：HiRDB 開始時に確保できる共用メモリセグメント数の余剰分です。次の計算式で値を求められます。

```
A=512-HiRDB開始時に各サーバに割り当てた共用メモリセグメント数
```

共用メモリセグメントの情報は pdls -d mem コマンド，又は OS のコマンドで確認できます。

《注意事項》

- 次に示す条件式を満たす場合は，このオペランドに pd_max_add_dbbuff_no オペランドの値が仮定されます。

  pd_max_add_dbbuff_shm_no の値＜ pd_max_add_dbbuff_no の値

  省略値が上記の条件を満たす場合も，pd_max_add_dbbuff_no オペランドの値が仮定されます。
- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- 追加する共用メモリセグメントサイズが SHMMAX オペランドの値を超える場合，SHMMAX オペランドの値を上限値とした複数の共用メモリセグメントに分割されます。追加する共用メモリセグメントサイズを想定して SHMMAX オペランドの値を増やすか，又は分割されたときに不足しないように pd_max_add_dbbuff_shm_no オペランドの値を増やしてください。
- グローバルバッファの動的変更機能を適用している場合，グローバルバッファ用共用メモリセグメントとして割り当てられる共用メモリセグメントの総数は次の計算式で求められます。

  512 ＋ pd_max_add_dbbuff_shm_no 指定値

  そのため，HiRDB 開始時に確保した共用メモリセグメント数が 512 を超えている場合，動的変更時には pd_max_add_dbbuff_shm_no オペランドの指定値からその超過分を引いた数の共用メモリセグメントが割り当てられることになります。
- このオペランドの値を変更した場合は，OS パラメタの共用メモリセグメントサイズの上限値，システム上の共用メモリセグメント数の上限値，及び 1 プロセス当たりの共用メモリセグメント数の上限値を見直してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- SHMMAX
- pdbuffer
- pd_max_add_dbbuff_no

《各見積もり式への影響》

pd_max_add_dbbuff_shm_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「Linux のオペレーティングシステムパラメタの見積もり」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

◆ **pd_dfw_bulk_write = awt | none**

デファードライト処理の一括出力機能を適用するかどうかを指定します。

awt：デファードライト処理の一括出力機能を適用します。

none：デファードライト処理の一括出力機能を適用しません。

デファードライト処理の並列 WRITE 機能は，出力対象ページ一覧上で同一 RD エリア内の連続ページが順番に並んでいる場合だけ，連続ページを一括出力します。

デファードライト処理の一括出力機能を適用すると，出力対象ページ一覧上で，出力対象ページを RD エリアごとのページ順に並び替えます。これによって，一括出力の機会を増やすことができます。そのため，デファードライト処理での並列 WRITE 機能の WRITE 回数を削減し，スループット向上が期待できます。特に，pdload による大量データのロードのようなケースで期待できます。

デファードライト処理の並列 WRITE 機能は，デファードライト処理の性能向上を目的とした機能です。デファードライト処理の並列 WRITE 機能を適用していて，さらなる性能向上を図る場合はこの機能の適用を検討してください。

《注意事項》

pdbuffer -D オペランドでグローバルバッファの分割数を指定している場合，そのグローバルバッファはデファードライト処理の一括出力機能の対象外になります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_dfw_awt_process

《各見積もり式への影響》

pd_dfw_bulk_write オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「64 ビットモードに移行する際の考慮点」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

9.2.23　インメモリデータ処理に関するオペランド

インメモリデータ処理については，マニュアル「HiRDB バッチ高速化機能」を参照してください。

◆ **pd_max_resident_rdarea_no = インメモリ RD エリアの最大数**

～＜符号なし整数＞((0～8388592))《0》

インメモリ RD エリアの最大数を指定します。このオペランドの指定値を超えた数の RD エリアをインメモリ化できません。

《前提条件》

このオペランドを指定する場合は HiRDB Accelerator が必要になります。

《指定値の目安》

- インメモリデータ処理を行う場合は，このオペランドに 1 以上を指定してください。0 を指定すると，インメモリデータ処理を実行できません。
- HiRDB/パラレルサーバの場合，このオペランドの指定値は 1 バックエンドサーバ当たりの最大数になります。

《注意事項》

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。

《各見積もり式への影響》

pd_max_resident_rdarea_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 7」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 7」

◆ **pd_max_resident_rdarea_shm_no = インメモリデータバッファが使用する共用メモリセグメントの最大数**

～＜符号なし整数＞((1～2147450879))《pd_max_resident_rdarea_no の値×1.5》

インメモリデータバッファが使用する共用メモリセグメントの最大数を指定します。このオペランドの指定値を超えた数の共用メモリセグメントを使用することはできません。

《前提条件》

- このオペランドを指定する場合は HiRDB Accelerator が必要になります。
- pd_max_resident_rdarea_no オペランドが指定されている必要があります。

《指定値の目安》

- インメモリデータ処理を行う場合はこのオペランドの指定を検討してください。マニュアル「HiRDB システム導入・設計ガイド」の「インメモリデータバッファが使用する共用メモリセグメント数」を参照してから，このオペランドの指定値を決定してください。
- HiRDB/パラレルサーバの場合，このオペランドの指定値は 1 バックエンドサーバ当たりの最大数になります。

《注意事項》

- このオペランドには，必要以上に大きな値を指定しないでください。このオペランドの指定値を大きくすると，HiRDB が使用する共用メモリが増加します。これによって，共用メモリが不足し HiRDB を開始できないことがあります。
- このオペランドの指定値を変更した場合，OS のオペレーティングシステムパラメタ（共用メモリセグメントの最大サイズ，システム上の共用メモリセグメントの最大数，1 プロセス当たりの共用メモリセグメントの最大数）の指定値を見直す必要があります。
- このオペランドに指定するのはあくまで HiRDB が管理する上限値です。OS が管理する上限値とは異なります。例えば，このオペランドに 200 を指定しても，OS の上限値が 100 の場合，確保できる共用メモリセグメントの最大数は 100 になります。

《各見積もり式への影響》

pd_max_resident_rdarea_shm_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「Linux のオペレーティングシステムパラメタの見積もり」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 7」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 7」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

9.2.24　表又はインデクスの予約数に関するオペランド

◆ **pd_assurance_table_no = 表予約数**

～＜符号なし整数＞((0～4194294500))《500》

空き領域の再利用機能を使用する表数を指定します。空き領域の再利用機能については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。このオペランドには，必ず《指定値の目安》に記載されている値を指定してください。

空き領域の再利用機能を使用する表数が，このオペランドの指定値以下の場合は空き領域の再利用機能を使用できます。ただし，空き領域の再利用機能を使用する表数がこのオペランドの指定値を超えると，空き領域の再利用機能を使用できなくなります。

《指定値の目安》

このオペランドの指定値の計算方法を次に示します。

- 1 表を 1 と数えてください。
- 横分割表の場合は 1 分割当たり 1 として数えてください。
- インナレプリカ機能を使用している場合，レプリカ RD エリアに格納した表は別表として数えてください。
- HiRDB/パラレルサーバの場合は，指定値がバックエンドサーバごとに適用されます。バックエンドサーバごとに計算して最大値をこのオペランドに指定してください。
- 次の値を指定してください。

  現在の空き領域の再利用機能を使用する表数＋次回の HiRDB 開始から終了までの間に定義予定の空き領域の再利用機能を使用する表数

《注意事項》

指定した値が不要に大きいと共用メモリ不足で HiRDB（HiRDB/パラレルサーバの場合はバックエンドサーバ）を開始できないことがあります。共用メモリの算出式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《各見積もり式への影響》

pd_assurance_table_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」

◆ **pd_assurance_index_no = インデクス予約数**

～＜符号なし整数＞((50～4194294500))《2500》

- 0904 互換モードの場合：**《500》**

使用するインデクスの数を指定します。このオペランドには，必ず《指定値の目安》に記載されている値を指定してください。インデクス数が，このオペランドの指定値以下の場合は，インデクス管理情報をメモリ上に常駐できるため，次に示すメリットがあります。

- インデクス情報をメモリ上に常駐し，性能を向上できる
- インデクスに関する統計情報を取得できる
- メッセージの重複出力を抑止できる
- インデクス空きページ解放後に，使用中セグメント内の未使用ページ優先割り当てができる

インデクス数がこのオペランドの指定値を超える場合は，これらのメリットがなくなります。

《指定値の目安》

このオペランドの指定値の計算方法を次に示します。

- 1 インデクスを 1 と数えてください。
- インデクスを横分割している場合は，1 分割当たり 1 として数えてください。例えば，3 分割している場合は 3 となります。
- インナレプリカ機能を使用している場合，レプリカ RD エリアに格納したインデクスは別インデクスとして数えてください。
- HiRDB/パラレルサーバの場合は，指定値がバックエンドサーバごとに適用されます。バックエンドサーバごとに計算し，最大値をこのオペランドに指定してください。
- 次の値を指定してください。

  現在のインデクス数＋次回の HiRDB 開始から終了までの間に追加予定のインデクス数

《注意事項》

指定した値が不要に大きいと共用メモリ不足で HiRDB（HiRDB/パラレルサーバの場合はバックエンドサーバ）を開始できないことがあります。

共用メモリの算出式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

9.2.25　参照制約及び検査制約に関するオペランド

◆ **pd_constraint_name = LEADING ｜ TRAILING**

参照制約又は検査制約で，制約名の定義を制約定義の前後どちらにするかを指定します。

LEADING：制約名定義を制約定義の前に指定します（標準 SQL 仕様）。

TRAILING：制約名定義を制約定義の後に指定します（XDM/RD 互換仕様）。

◆ **pd_check_pending = USE ｜ NOUSE**

参照制約又は検査制約で，検査保留状態（整合性が保証できなくなったデータの操作を制限している状態）を使用するかどうかを指定します。

USE：検査保留状態を使用します。

NOUSE：検査保留状態を使用しません。

《指定値の目安》

- 参照制約，及び検査制約を使用する場合

このオペランドに NOUSE を指定すると，データの整合性が保証されないことがあるため，通常は USE を指定してください。データの整合性よりも処理性能を優先する場合は，NOUSE を指定してください。

- 参照制約，及び検査制約を使用しない場合

  NOUSE を指定してください。

《注意事項》

このオペランドに USE を指定するか，又は指定を省略した場合，PURGE TABLE 文を実行時に HiRDB は次の動作をします。

- ディクショナリ表（資源種別：3005，種別名：DICT）に対して一時的に EX モードで排他を掛ける
- データディクショナリ用 RD エリア（資源種別：0001，種別名：RDAR）に対してトランザクションが終了するまで SU モードで排他を掛ける

そのため，これらに排他を掛けるコマンド，又はユティリティは同時に実行できないことがあります。コマンド，又はユティリティが次に示す条件の場合は，コマンド，又はユティリティを同時実行しないようにしてください。

コマンド，又はユティリティ	条件
pddbchg（レプリカ RD エリアのレプリカステータスの切り替え）	-w オプションを指定していない場合
pdmod（データベース構成変更ユティリティ）	ディクショナリ表の属性定義変更の場合
pdcopy（データベース複写ユティリティ）	次の二つの条件を満たす場合 • -M オプションに x 又は r を指定した場合 • 複写対象にマスタディレクトリ用 RD エリア，又はデータディクショナリ用 RD エリアを含む場合
pdreginit（レジストリ機能初期設定ユティリティ）	-k オプションに all を指定した場合

9.2.26　一時表に関するオペランド

◆ pd_max_temporary_object_no = ある一時点で使用する一時表と一時インデクスの最大数～<符号なし整数>((0～131072))《600》

- 0904 互換モードの場合：**《0》**

サーバごとに，ある一時点で使用する一時表と一時インデクスの最大数を指定します。

《指定値の目安》

このオペランドの指定値は，次の計算式で求めてください。HiRDB/パラレルサーバの場合は，バックエンドサーバごとに求めてください。

（1トランザクションで使用するトランザクション固有一時表
＋その一時表の一時インデクスの総数）の最大数×アクティビティ数※

＋（SQLセッションで使用するSQLセッション固有一時表
＋その一時表の一時インデクスの総数）×接続ユーザ数

注※

アクティビティ数：

（(pd_max_users の指定値＋ 3）×2 ＋ 1）＋ α

α：pd_max_users の指定値が 60 以下の場合は 5，pd_max_users の指定値が 61 以上の場合は 0

HiRDB/パラレルサーバの場合，このオペランドの値はバックエンドサーバごとに適用されます。そのため，指定値が最も多いバックエンドサーバを目安にしてこのオペランドを指定してください。

《注意事項》

0904 互換モードの場合，このオペランドの指定値を 1 以上にするときは，
pd_max_tmp_table_rdarea_no オペランドの指定値も 1 以上にしてください。

《ほかのオペランドとの関連》

このオペランドは，pd_max_tmp_table_rdarea_no オペランドと関連があります。

《各見積もり式への影響》

pd_max_temporary_object_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 8」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 8」

◆ **pd_tmp_table_initialize_timing = STARTING ｜ ACCESS**

一時表用 RD エリアは，通常，HiRDB 開始時に初期化されます。このオペランドは，一時表用 RD エリアの初期化を HiRDB 開始時ではない時に変更する場合に指定します。一時表用 RD エリアの初期化については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

STARTING：

HiRDB 開始時に，前回稼働中に使用したすべての一時表用 RD エリアを初期化します。

ACCESS：

一時表に最初に INSERT 文が実行された時に，格納先候補となった一時表用 RD エリアを初期化します。格納先候補となる一時表用 RD エリアについては，マニュアル「HiRDB システム導入・設計ガイド」の「格納先 RD エリアの決定規則」を参照してください。

《注意事項》

ACCESS を指定すると，HiRDB を開始してから一時表に最初に INSERT 文が実行された時に一時表用 RD エリアを初期化するオーバヘッドが発生します。そのため，次の要因でトランザクションがキャンセルされるおそれがあります。

- 排他待ち限界経過時間を超過した
- HiRDB クライアントの最大待ち時間を超過した

一時表用 RD エリアの初期化によるオーバヘッド量（ディレクトリページの初期化ページ数）は次の計算式で求められます。

```
↑1000÷（50×（ページサイズ÷2048））↑
＋Σ↑各HiRDBファイルのセグメント数
÷（16000×（ページサイズ÷2048））↑
（単位：ページ）
```

《ほかのオペランドとの関連》のオペランドの指定値は，上記のオーバヘッドの時間を考慮して見積もってください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_lck_wait_timeout
- クライアント環境定義の PDCWAITTIME

9.2.27　システムログファイルに関するオペランド

◆ **pd_log_dual = Y ｜ N**

システムログファイルを二重化するかどうかを指定します。

Y：システムログファイルを二重化します。

N：システムログファイルを二重化しません。

《利点》

システムログファイルを二重化すると，HiRDB はＡ系及びＢ系の両方に同じシステムログを取得します。取得したシステムログを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシステムログを読み込めるため，システムの信頼性を向上できます。

《ほかのオペランドとの関連》

システムログファイルを二重化する場合は，pdlogadpf オペランドで B 系のシステムログファイル名を指定してください。

◆ **pd_log_dual_write_method = serial ｜ parallel**

このオペランドは AIX 版及び Linux 版限定のオペランドです。

システムログの並列出力機能を使用するかどうかを指定します。システムログの並列出力機能については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

serial：システムログの並列出力機能を使用しません。

parallel：システムログの並列出力機能を使用します。

このオペランドに parallel を指定した場合，HiRDB は aio ライブラリ（AIX の場合は Asynchronous I/O Subsystem，Linux の場合は libaio）を使用して，システムログファイルへの出力を並列に実行します。

《前提条件》

parallel を有効にする場合は，次に示す条件をすべて満たす必要があります。

- pd_duplicated_fs_suffix オペランドを指定していない
- aio ライブラリを導入して必要な設定が行われている
- pd_log_dual = Y
- システムログファイルがキャラクタ型スペシャルファイルに配置されている

この条件を満たしていない場合は，このオペランドの指定値に関係なくシステムログの並列出力機能を使用できません。

◆ **pd_log_remain_space_check = warn ｜ safe**

システムログファイルの空き率が**警告値**未満になったときの HiRDB の処理方式を指定します。この機能を**システムログファイルの空き容量監視機能**といいます。システムログファイルの空き容量監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

warn：

システムログファイルの空き率が警告値未満になった場合，警告メッセージ KFPS01162-W を出力します。

safe：

システムログファイルの空き率が警告値未満になった場合，新規トランザクションのスケジューリングを抑止して，サーバ内の全トランザクションを強制終了します。このとき，KFPS01160-E メッセージを出力します。

《指定値の目安》

システムログファイルの空き容量不足によるユニットの異常終了の可能性を低くできるため，safe の指定を推奨します。ただし，safe を指定すると，システムログファイルの空き容量が不足したときに，サーバ内の全トランザクションが強制終了されます。このため，システムログファイルの設計をより正確に行う必要があります。システムログファイルの設計については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

◆ **pd_log_auto_unload_path = "アンロードログファイルの出力先ディレクトリ"〔，"アンロードログファイルの出力先ディレクトリ"〕...**

～＜パス名＞((1～136 文字))

システムログの**自動ログアンロード機能**を使用する場合に，アンロードログファイルの出力先ディレクトリを絶対パス名で指定します。アンロードログファイルを HiRDB ファイルシステム領域に作成する場合は，その HiRDB ファイルシステム領域名を指定してください。なお，**このオペランドに指定するディレクトリ又は HiRDB ファイルシステム領域は，HiRDB の開始前に作成しておいてください。**

また，このオペランドには，サーバごとに異なるディレクトリ又は HiRDB ファイルシステム領域を指定してください。

自動ログアンロード機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。ログをアンロードする運用については，マニュアル「HiRDB システム運用ガイド」の「システムログをアンロードする運用」を参照して，自動ログアンロード機能の適用を検討してください。

《指定値の目安》

空きディスク容量を確認してからディレクトリを指定してください。作成されるアンロードログファイルによって，ディスク容量が圧迫される原因になります。

また，ディスク容量の不足によって，指定したディレクトリにアンロードログファイルが作成できない場合，自動ログアンロード機能が停止します。このようなことが発生する可能性がある場合は，ディレクトリを複数指定することをお勧めします。

ただし，ディレクトリは一つだけの方がデータベース回復時の運用（回復に必要なアンロードログファイルの選別）が若干簡単になります。

ディレクトリを複数指定する場合の補足事項を次に示します。

- ディスク障害などに備えて，異なるパーティションのディレクトリを指定することをお勧めします。
- ディスク容量満杯又はディスク障害などで一つのディレクトリ下にアンロードログファイルが作成できない場合，ほかのディレクトリ下にアンロードログファイルを作成します。このとき，HiRDB はこのオペランドに指定した順番にディレクトリを使用します。

《オペランドの規則》

- ディレクトリは最大 128 個指定できます。
- 複数のディレクトリを指定した場合，同じパス名を指定できません。

《注意事項》

- 次の場合は，自動ログアンロード機能を使用できません。

  ・pd_log_unload_check オペランドに N を指定している場合

  ・ログ同期方式のリアルタイム SAN レプリケーションを使用している，ログ適用サイトの場合（KFPS04689-W メッセージが出力されます）
- アンロードログファイル作成ディレクトリを二つ以上作成している場合，HiRDB を正常開始したときに空のディレクトリがないと自動ログアンロード機能は停止します。
- マルチ HiRDB を利用している場合は，HiRDB ごとに異なるディレクトリを作成してください。同じディレクトリを指定すると，どのアンロードログファイルがどの HiRDB に対応しているか分からなくなる可能性があります。

◆ **pd_log_auto_unload_restart = Y | N**

障害で自動ログアンロード機能が停止した旨のメッセージ（KFPS01150-E メッセージ）出力後，次回のシステムログファイルのスワップを契機に自動ログアンロード機能を再開始するかどうかを指定します。

Y：自動ログアンロード機能を再開始します。

N：自動ログアンロード機能を再開始しません。

《前提条件》

次に示す二つの条件を満たす必要があります。

- pd_log_unload_check オペランドに Y を指定しているか，又は指定を省略している。
- pd_log_auto_unload_path オペランドを指定している。

《利点》

- このオペランドにYを指定すると，ワークファイルの増加でディスク容量が不足した場合やプロセス生成失敗などの一時的な障害でアンロードが失敗した場合でも，次回のシステムログファイルのスワップ発生時に自動的にシステムログのアンロードを再開始します。
- このオペランドにNを指定すると，自動ログアンロード機能が障害停止中にアンロードログファイルを退避する運用を行っている場合，pdlogatul -t コマンドを実行しないで退避してしまうなどのミスが抑止できます。

《指定値の目安》

通常はYを指定するか，指定を省略してください。

なお，既に自動ログアンロード機能の停止メッセージ（KFPS01150-E メッセージ）を監視して，pdlogatul -b コマンドで再開始する運用を行っている場合は，Nを指定してください。

◆ **pd_log_singleoperation = Y | N**

このオペランドは，システムログファイルを二重化している場合に指定してください。二重化していない場合は指定する必要はありません。

システムログファイルの片系運転をするかどうかを指定します。システムログファイルに障害が発生して，両系ともに使用できるシステムログファイルがない場合でも，HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了しないで正常な片方の系だけで処理を続行します。これを**システムログファイルの片系運転**といいます。

また，両方のシステムログファイルで処理を続行すること(通常の処理形態)を**システムログファイルの両系運転**といいます。

Y：システムログファイルの片系運転をします。

N：システムログファイルの片系運転をしません。常に両系運転とします。

《前提条件》

pd_log_dual オペランドにYを指定している必要があります。

◆ **pd_log_rerun_reserved_file_open = Y | N**

システムログファイルの自動オープンをするかどうかを指定します。

HiRDB（又はユニット）を再開始するときに，「上書きできる状態」のシステムログファイルがない場合，予約のファイルがあればHiRDBが予約のファイルをオープンして「上書きできる状態」にし，処理を続行します。これを**システムログファイルの自動オープン**といいます。

予約ファイルを使用するのは，次の場合です。

- 再開始後，最初のシンクポイントダンプを取得するまでの間
- オープンされたファイルグループすべてが「上書きできない状態」の場合

Y：システムログファイルの自動オープンをします（予約ファイルをオープンして使用します）。

N：システムログファイルの自動オープンをしません（予約ファイルを使用しません）。

《利点》

このオペランドにYを指定すると，ユニットの再開始時に「スワップ先にできる状態」のファイルがなくても，予約のファイルがあればユニットを再開始できます。

ただし，「アンロード待ち状態」のファイルがあるときは，ユニットを停止します。「アンロード待ち状態」のファイルのアンロードを実行した後，再度ユニットを開始してください。

◆ **pd_log_rerun_swap = Y ｜ N**

HiRDB（又はユニット）の再開始時に，システムログファイルをスワップするかどうかを指定します。

Y：システムログファイルをスワップします。

N：システムログファイルをスワップしません。

《利点》

このオペランドに Y を指定すると，再開始前後でシステムログファイルを物理的に分けられます。このため，再開始前のシステムログファイルをサーバ稼働中に使用できます。

◆ **pd_log_swap_timeout = システムログファイルのスワップ完了待ち時間**
～<符号なし整数>((1～32580))《180》（単位：秒）

システムログファイルのスワップが完了するまでの待ち時間を秒単位で指定します。指定した時間を過ぎてもシステムログファイルのスワップが完了しない場合は，ユニットが異常終了します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。マシンの性能が低いなどの理由によってシステムログファイルのスワップに時間が掛かる場合は，このオペランドの値を大きくしてください。また，ディスク障害などによってシステムログファイルのスワップに遅延が発生した場合に，より短い時間で異常を検知し，ユニットを異常終了させたいときは，このオペランドの値を小さくしてください。

◆ **pd_log_unload_check = Y ｜ N**

システムログファイルのアンロード状態を，HiRDB がチェックするかどうかを指定します。

Y：

アンロード状態をチェックします（通常の運用となります）。

N：

アンロード状態をチェックしません。アンロードの状態に関係なく，次に示す条件がすべて満たされれば，システムログファイルを「スワップ先にできる状態」にします。

- 上書きできる状態
- 抽出完了状態（HiRDB Datareplicator）
- オンライン再編成上書き可能状態（HiRDB Staticizer Option）

このときシステムログファイルの運用方法が，アンロード状態のチェックを解除する運用になります。アンロード状態のチェックを解除する運用については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《利点》

N を指定したときの利点を次に示します。

- システムログファイルのアンロード操作がなくなるため，運用方法が簡単になります。
- アンロードログファイルを保管するためのファイル容量が不要になります。

《指定値の目安》

データベースを回復するときにシステムログを使用しない場合（バックアップ取得時点に回復できればよい場合）に N を指定します。

《注意事項》

N を指定したときの注意事項を次に示します。

- バックアップ取得時点にしかデータベースを回復できません。
- データベースの回復にシステムログが必要なのにこの運用をした場合はデータベースの回復手段がなくなります。

◆ **pd_log_write_buff_count = ログ出力バッファ面数**

～＜符号なし整数＞((10～65000))《10》

システムログの出力に使用するバッファの面数を指定します。

《指定値のチューニング方法》

最初は 10（省略値）を指定してください。その後，統計解析ユティリティ（システムの稼働に関する統計情報）でカレントバッファなしによる待ち回数（# OF WAIT THREAD）を調査してください。この値が多ければ，スループットが向上するように指定値を大きくしてください。

《注意事項》

このオペランドの指定値が小さく，トランザクションの量が多い場合は，複数のトランザクションでシステムログの出力待ちが発生するため，システム性能が低下します。

《ほかのオペランドとの関連》

- pd_log_max_data_size オペランドとともに，ログ出力バッファ面数を決定してください。

◆ **pd_log_full_write_async = Y ｜ N**

ログ出力バッファが満杯になった場合のシステムログファイルへの出力方式を指定します。

Y：

トランザクションとは非同期にシステムログファイルへの出力処理を実施します。

N：

トランザクションと同期してシステムログファイルへの出力処理を実施します。

《指定値の目安》

通常はこのオペランドを指定する必要はありません。

システムログ出力量が多いトランザクションを実行する場合は，このオペランドに Y を指定するとトランザクションの実行時間が改善されることがあります。

《注意事項》

- 1:1 スタンバイレス型系切り替え機能を適用しているユニットでは，このオペランドは指定できません。
- 短期間に大量の更新が発生するトランザクションを実行する場合にこのオペランドに Y を指定すると，N を指定した場合に比べてシステムの稼働に関する統計情報のカレントバッファなしによる待ち回数（# OF WAIT THREAD）の値が増加する場合があります。その場合，

pd_log_write_buff_count オペランドの《指定値のチューニング方法》を参考に，pd_log_write_buff_count の指定値を設定することで，カレントバッファなしによる待ち回数（# OF WAIT THREAD）の値を減らせる場合があります。

◆ pd_log_auto_expand_size = システムログファイルの拡張契機 1 回当たりに拡張するサイズ〔，拡張上限サイズ〕

～＜符号なし整数＞((0～104857600))《0，0》（単位：レコード）

このオペランドは，システムログファイルの自動拡張機能を使用する場合に指定します。

一つのシステムログファイルが拡張契機 1 回当たりに拡張するサイズ，及び拡張できるファイルサイズの上限をレコード数で指定します。

1 回当たりに拡張するサイズを省略するか，又は 0 を指定した場合，システムログファイルの自動拡張を行いません。拡張上限サイズを省略するか，又は 0 を指定した場合，ファイルシステム領域のあるディスクが満杯になるか，システムログファイルの容量が上限に達するまでシステムログファイルの自動拡張を行います。また，1 回当たりに拡張するサイズに拡張上限サイズより大きい値を指定した場合，拡張上限サイズに指定した値まで自動拡張を行います。

システムログファイルの自動拡張機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《指定値の目安》

1.サーバ共通定義に指定する場合

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。サーバごとにシステムログファイルのレコード数の平均値の 1/10 を計算し，最大値を持つサーバの値を指定してください。

  ・1:1 スタンバイレス型系切り替え機能を適用している BES の場合，代替 BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

  ・影響分散スタンバイレス型系切り替え機能を適用している BES の場合，ゲスト BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

- 通常，拡張上限サイズは省略してください。

2.シングルサーバ定義，フロントエンドサーバ定義及びディクショナリサーバ定義に指定する場合

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。全システムログファイルのレコード数の平均値の 1/10 を計算し，指定してください。
- 通常，拡張上限サイズは省略してください。

3.バックエンドサーバ定義に指定する場合

- 1 回当たりに拡張するサイズには，pdloginit コマンドの-n オプションでシステムログファイルを作成した際に指定したレコード数を基に指定します。全システムログファイルのレコード数の平均値の 1/10 を計算し，指定してください。

  ・1:1 スタンバイレス型系切り替え機能を適用している BES の場合，代替 BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

・影響分散スタンバイレス型系切り替え機能を適用している BES の場合，ゲスト BES のシステムログファイルを含めた平均値の 1/10 を計算してください。

- 通常，拡張上限サイズは省略してください。

《指定値のチューニング方法》

システムログの出力量が自動拡張で拡張するサイズを超えると，システムログファイルが満杯になってユニットダウンする場合があります。その場合は，1 回当たりに拡張するサイズを大きくしてください。また，拡張処理に時間が掛かり，トランザクション性能に影響を及ぼす場合は，1 回当たりに拡張するサイズを小さくしてください。

◆ pd_log_fg_warning_point = 空きシステムログファイルの数 1〔，空きシステムログファイルの数 2〔，空きシステムログファイルの数 3〕〕

～<符号なし整数>((0～197))《0》

空きシステムログファイルの数が少なくなったことを警告するメッセージ（KFPS01165-W）の出力契機を指定します。この機能をシステムログファイルの残量警告機能といいます。システムログファイルの残量警告機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《利点》

このオペランドに，運用上，最小限空きとして確保しておきたいシステムログファイル数を指定することで，運用変化によって想定よりシステムログの使用量が多くなったことを検知できます。また，併せて出力する KFPS01166-I メッセージの情報から，システムログファイルの容量やオンライン再編成運用の見直しに利用できます。

《指定値の目安》

運用上，最小限空きとして確保しておきたいシステムログファイル数を指定します。

空きシステムログファイルは，運用中に pdlogls -d sys コマンドを実行することで確認できます。

《オペランドの規則》

このオペランドに 0 を 1 つだけ指定すると，警告メッセージを出力しません。

また，複数個の値を指定し，指定した値に 0 を含んでいる場合，0 以外の値が有効になります。

同じ値を複数個指定した場合は，1 つだけ有効になります。

《注意事項》

運用しているシステムログファイルグループ数−3 以下の値を指定してください。

運用しているシステムログファイルグループ数−2 以上の値を指定した場合，一度警告メッセージを出力すると二度と警告メッセージを出力しなくなります。また，定義しているシステムログファイルグループ数以上の値を指定した場合は，その指定値は無効となります。

◆ pd_log_block_check = Y ｜ N

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。

システムログファイル出力時にブロックチェック用の情報を付加してシステムログを出力するかを指定します。

Y：

システムログファイル出力時にブロックチェック用の情報を付加します。

N：

システムログファイル出力時にブロックチェック用の情報を付加しません。

《利点》

このオペランドに Y を指定すると，リードレプリカの自動反映処理で反映対象システムログの読み込み時にデータ破損をチェックできます。

《指定値の目安》

このオペランドは通常指定する必要はありません。

次の条件を満たす場合はこのオペランドに Y を指定してください。

- 参照 DB でリードレプリカの自動反映処理を適用している
  （システム共通定義に pd_read_replica_use = Y を指定している，かつシステム共通定義に pd_read_replica_mode = sys を指定している，又は省略している）
- このオペランドを指定する HiRDB が更新 DB である

◆ **pdlogadfg -d sys -g ファイルグループ名〔ONL〕**

システムログファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシステムログファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～200 個指定できます。

《オペランドの規則》

このオペランドは 2～200 個指定できます。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

◆ **pdlogadpf -d sys -g ファイルグループ名 -a "システムログファイル名" 〔-b "システムログファイル名"〕**

ファイルグループを構成するシステムログファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "システムログファイル名"：～＜パス名＞((167 文字以内))

ファイルグループを構成するシステムログファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

-b "システムログファイル名"：～＜パス名＞((167 文字以内))

システムログファイルを二重化（pd_log_dual オペランドに Y を指定）する場合に，B 系のシステムログファイル名を絶対パス名で指定します。pd_log_dua オペランドに Y を指定しない場合は，システムログファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシステムログファイルの名称を指定してください。なお，システムログファイル名はユニット内で一意になるようにしてください。

《注意事項》

HiRDB Datareplicator（抽出側）と連携しているときにこのオペランドを追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度終了してください。HiRDB Datareplicator を稼働した状態でオペランドを変更すると，HiRDB Datareplicator の抽出処理が失敗することがあります。

◆ pdlogadfg -d rrp -g ファイルグループ名〔ONL〕

参照 DB で反映対象とする更新 DB のシステムログファイル（反映対象システムログファイル）のファイルグループを指定します。リードレプリカ機能の自動反映方式を使用している場合に参照 DB 側のサーバ定義に指定してください。ここで指定したファイルグループに対して，pdlogadpf -d rrp オペランドで反映対象システムログファイルを割り当てます。

必ず pdlogadfg -d rrp オペランド，pdlogadpf -d rrp オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

ファイルグループ名を指定します。対応する更新 DB のシステムログファイルのファイルグループ名と同じ名称にしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。リードレプリカ機能の自動反映方式を使用している場合，ONL の指定の有無に関係なく，参照可能なファイルについて参照します。

《前提条件》

pd_read_replica_use オペランドに Y を指定し，かつ pd_read_replica_mode オペランドに sys を指定している必要があります。

《オペランドの規則》

このオペランドは 2～200 個指定できます。

《注意事項》

- オペランドの前提条件を満たさない場合は，このオペランドを指定しても無視します。
- 更新 DB のサーバ定義に指定した pdlogadfg -d sys オペランドのすべてのファイルグループに対して，pdlogadfg -d rrp オペランドを指定してください。
- 参照 DB 側のサーバ定義では，pdlogadfg -d sys オペランドと，pdlogadfg -d rrp オペランドをそれぞれ指定してください。

なお，pdlogadfg -d sys と，pdlogadfg -d rrp のファイルグループ名は重複しても問題ありません。

◆ pdlogadpf -d rrp -g ファイルグループ名 -a "反映対象システムログファイル名" 〔-b "反映対象システムログファイル名"〕

参照 DB で反映対象とする更新 DB のファイルグループを構成するシステムログファイル（反映対象システムログファイル）を指定します。1 つのファイルグループに対して，このオペランドを 1 つ指定します。

必ず pdlogadfg -d rrp オペランド，pdlogadpf -d rrp オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。対応する更新 DB のシステムログファイルのファイルグループ名と同じ名称にしてください。

-a "反映対象システムログファイル名"：～＜パス名＞((167 文字以内))

ファイルグループを構成する反映対象システムログファイルの名称を絶対パス名で指定します。更新 DB 側で作成したシステムログファイルの名称を指定してください。なお，反映対象システムログファイル名はユニット内で一意になるようにしてください。

-b "反映対象システムログファイル名"：～＜パス名＞((167 文字以内))

システムログファイルを二重化（pd_log_dual オペランドに Y を指定）する場合に，B 系の反映対象システムログファイル名を絶対パス名で指定します。pd_log_dua オペランドに Y を指定しない場合は，反映対象システムログファイル名を指定しても無効になります。

更新 DB 側で作成したシステムログファイルの名称を指定してください。なお，反映対象システムログファイル名はユニット内で一意になるようにしてください。

《前提条件》

pd_read_replica_use オペランドに Y を指定し，かつ pd_read_replica_mode オペランドに sys を指定している必要があります。

《注意事項》

- オペランドの前提条件を満たさない場合は，このオペランドを指定しても無視します。
- pdlogadfg -d rrp オペランドのすべてのファイルグループに対して，pdlogadpf -d rrp オペランドを指定してください。
- 参照 DB 側のサーバ定義では，pdlogadpf -d sys オペランドと，pdlogadpf -d rrp オペランドをそれぞれ指定してください。

なお，pdlogadpf -d sys と，pdlogadpf -d rrp のファイルグループ名は重複しても問題ありません。

9.2.28　シンクポイントダンプファイルに関するオペランド

◆ **pd_spd_dual = Y ｜ N**

シンクポイントダンプファイルを二重化するかどうかを指定します。

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，サーバ用副シンクポイントダンプファイルも対象となります。

Y：シンクポイントダンプファイルを二重化します。

N：シンクポイントダンプファイルを二重化しません。

《利点》

シンクポイントダンプファイルを二重化すると，HiRDB はＡ系及びＢ系の両方に同じシンクポイントダンプを取得します。取得したシンクポイントダンプを読み込むとき，片方のファイルに異常が発生しても，もう一方のファイルからシンクポイントダンプを読み込めるため，システムの信頼性を向上できます。

《ほかのオペランドとの関連》

シンクポイントダンプファイルを二重化する場合は，pdlogadpf オペランドで B 系のシンクポイントダンプファイル名を指定してください。

◆ **pd_spd_assurance_msg = Y ｜ N**

シンクポイントダンプが有効化されたとき，メッセージ KFPS02183-I を出力するかどうかを指定します。

Y：出力します。

N：出力しません。

◆ **pd_spd_assurance_count = 有効保証世代数**

～＜符号なし整数＞((1～2))《1》

システム回復時のシンクポイントダンプファイルの入力障害などに備え，HiRDB 稼働中に保存しておくシステムログファイルの範囲を，シンクポイントダンプファイルの世代数によって指定します。この世代数のことを**有効保証世代数**といいます。ここで指定した世代数分のシンクポイントダンプファイルは，上書きできない状態になります。

なお，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，このオペランドを指定しても無視されます。また，ログ適用サイトのシステムログファイルは，ログ適用していないシステムログが含まれていても，業務サイトによって上書きされます。

《利点》

有効保証世代数を 2 にすると，最新世代のシンクポイントダンプファイルに障害が発生しても，1 世代前のシンクポイントからシステムを回復できるため，信頼性が向上します。

《指定値の目安》

- シンクポイントダンプファイルを二重化(pd_spd_dual = Y)して信頼性を向上している場合は，このオペランドを省略するか，又は 1 を指定してください。
- 信頼性を向上させる場合は有効保証世代数を 2 にしてください。ただし，有効保証世代数を 2 にすると，上書きできない状態のシンクポイントダンプファイルが増えます（二つになります）。

《注意事項》

- シンクポイントダンプファイルの必要最低数は**有効保証世代数＋ 1** となります。
- 有効保証世代数を 2 にすると，システムログファイル数が不足して HiRDB が異常終了する可能性が上がります。HiRDB が異常終了するとシステムログファイルを追加して復旧するまでの間サービスが停止するため，システムログファイル不足による HiRDB 異常終了が発生しないようにあらかじめシステムログファイルの容量を確保しておく必要があります。詳細は，マニュアル「HiRDB システム導入・設計ガイド」の「システムログファイルの設計」を参照してください。有効保証世代数を 2 にするとシステムログファイル数が不足する理由は，2 世代前のシンクポイント取得以降のシステムログファイルを用いて回復処理することに伴い上書きできない状態のシステムログファイルが増えるためです。

◆ **pd_spd_reduced_mode = 縮退運転オプション**

～＜符号なし整数＞((0～2))《0》

シンクポイントダンプファイルの縮退運転をするかどうかを指定します。

縮退運転とは，HiRDB 稼働中又は再開始処理中のファイル障害などによって，シンクポイントダンプファイルの数が運用に必要なファイル数（**有効保証世代数**※**＋ 1**）以下になった場合でも，最低二つのファイルが使用できれば処理を続行する機能です。

注※　pd_spd_assurance_count オペランドの指定値です。

0：縮退運転を使用しません。

1：縮退運転を使用します。

2：縮退運転を使用し，縮退運転でシンクポイントダンプ取得の契機ごとに警告メッセージを出力します。

◆ **pd_spd_reserved_file_auto_open = Y ｜ N**

シンクポイントダンプファイルの自動オープンをするかどうかを指定します。

シンクポイントダンプファイルに障害が発生して，使用できるシンクポイントダンプファイル数が運用に必要なファイル数（**有効保証世代数**※**＋ 1**）未満になった場合，予約のファイルがあれば HiRDB が予約のファイルをオープンして上書きできる状態にし，処理を続行します。これを**シンクポイントダンプファイルの自動オープン**といいます。

なお，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，サーバ用副シンクポイントダンプファイルも対象となります。

注※　pd_spd_assurance_count オペランドの指定値です。

Y：

シンクポイントダンプファイルの自動オープンをします。使用できるシンクポイントダンプファイル数が運用に必要なファイル数（有効保証世代数＋ 1）未満になった場合，予約ファイルを自動的にオープンします。

N：

シンクポイントダンプファイルの自動オープンをしません。使用できるシンクポイントダンプファイル数が運用に必要なファイル数（有効保証世代数＋ 1）未満になっても，予約ファイルを自動的にオープンしません。

《ほかのオペランドとの関連》

このオペランドの指定は pd_spd_reduced_mode オペランドより優先されます。

◆ **pd_spd_max_data_size = シンクポイントダンプファイルのバッファ長**
～＜符号なし整数＞((32000～4000000))《32768》（単位：バイト）

シンクポイントダンプをシンクポイントダンプファイルに入出力するときに使用するバッファ（共用メモリ）の大きさをバイト単位で指定します。

ここで指定した値によってシンクポイントダンプファイルの入出力回数を制御します。

なお，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，サーバ用副シンクポイントダンプファイルも対象となります。

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。
- 指定値を大きくするほど，シンクポイントダンプファイルに対する入出力回数が少なくなります。

◆ **pd_log_sdinterval = システムログ出力量〔，経過時間〕**

シンクポイントダンプの取得間隔を指定します。次に示す条件で指定できます。

- システムログの出力量
- 前回シンクポイントダンプを取得してからの経過時間

システムログ出力量：～＜符号なし整数＞((100～100000))《5000》（単位：ログブロック数）

シンクポイントダンプの取得間隔を指定します。ここで指定したログブロック数分のシステムログが出力されるたびに，シンクポイントダンプを取得します。

経過時間：～＜符号なし整数＞((0,10～1440))《60》（単位：分）

シンクポイントダンプの取得間隔を指定します。前回シンクポイントダンプを取得してから，ここで指定した経過時間が過ぎると，シンクポイントダンプを取得します。

- 経過時間に 0 を指定すると，HiRDB は経過時間でのシンクポイントダンプを取得しません。
- 前回シンクポイントダンプを取得してからトランザクションが一度も発生しない場合，ここで指定した経過時間が過ぎてもシンクポイントダンプを取得しません。

《指定値の目安》

- HiRDB を再開始するときの所要時間に特に規定のない場合は，このオペランドを指定する必要はありません。
- このオペランドの指定値によって，HiRDB を再開始するときの所要時間が決まります。

  このオペランドの指定値を小さくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が短くなります。ただし，シンクポイントダンプの取得回数が増えるため，オンライン中の性能が低下する場合があります。

  逆に，このオペランドの指定値を大きくすれば，HiRDB を再開始するときのデータベース回復に掛かる所要時間が長くなります。ただし，シンクポイントダンプの取得回数が減るため，オンライン中の性能が向上する場合があります。

《指定値のチューニング方法》

シンクポイントダンプがどの程度の間隔で取得されているかは，統計解析ユティリティのシステムの稼働に関する統計情報の「シンクポイントダンプ取得間隔時間（SYNC POINT GET INTERVAL）」で確認できます。SYNC POINT GET INTERVAL の平均値を見て，シンクポイントダンプを取得する間隔が長いと判断したら，ここでの指定値を小さくしてください。また，反対に短いと判断したら，ここでの指定値を大きくしてください。

《注意事項》

- シンクポイントダンプの取得間隔はシステムログの出力量で決まります。このため，更新系のトランザクションがほとんどない時間帯は，メモリからデータベースに反映するまでの時間が長くなります。このような状態で障害が発生すると，その間に発生したトランザクションの回復時間が長くなります。このようなケースが考えられる場合は，「経過時間」でもシンクポイントダンプの取得間隔を設定してください。
- シンクポイントダンプを取得すると，グローバルバッファからの更新ページ出力に伴い，CPU，入出力処理，及び排他に掛かる負荷が増大します。そのため，シンクポイントダンプの取得間隔を短くすると，処理が遅延することがあります。

◆ pdlogadfg -d spd -g ファイルグループ名〔ONL〕

シンクポイントダンプファイルのファイルグループを指定します。このオペランドは省略できません。必ず指定してください。ここで指定したファイルグループに対して，pdlogadpf オペランドでシンクポイントダンプファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

ファイルグループ名を指定します。ファイルグループ名はサーバ内で一意になるようにしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～30 個指定できます。

《オペランドの規則》

このオペランドは 2～60 個指定できます。

◆ pdlogadpf -d spd -g ファイルグループ名 -a "シンクポイントダンプファイル名" 〔-b "シンクポイントダンプファイル名"〕

ファイルグループを構成するシンクポイントダンプファイルを指定します。このオペランドは省略できません。必ず指定してください。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。ファイルグループ名はユニット内で一意になるようにしてください。

-a "シンクポイントダンプファイル名"：～＜パス名＞((167 文字以内))

ファイルグループを構成するシンクポイントダンプファイルの名称を絶対パス名で指定します。pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

-b "シンクポイントダンプファイル名"：～＜パス名＞((167 文字以内))

シンクポイントダンプファイルを二重化（pd_spd_dual = Y を指定）する場合に，B 系のシンクポイントダンプファイル名を絶対パス名で指定します。pd_spd_dual = Y の指定がない場合は，シンクポイントダンプファイル名を指定しても無効になります。

pdloginit コマンドで初期設定したシンクポイントダンプファイルの名称を指定してください。なお，シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

◆ **pdlogadfg -d ssp -g ファイルグループ名〔ONL〕**

このオペランドは，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に指定します。ログ適用サイトで使用する副シンクポイントダンプファイルのファイルグループを指定します。ここで指定したファイルグループに対して，pdlogadpf オペランドで副シンクポイントダンプファイルを割り当てます。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

ファイルグループ名を指定します。対応する正シンクポイントダンプファイルのファイルグループ名と同じ名称にしてください。

ONL：

このファイルグループを HiRDB の稼働時に使用できる状態（オープン状態）にする場合に指定します。ONL を指定したファイルグループは 2～30 個指定できます。

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，システムログ適用化後に初めてログ適用サイトとして HiRDB を開始すると，業務サイトから引き継ぐ必要のあるファイルグループは ONL の指定の有無に関係なく，シングルサーバの開始と同時に使用できるようになります。

《前提条件》

- 前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。
- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している必要があります。

《オペランドの規則》

このオペランドは 2～60 個指定できます。

《注意事項》

pdlogadfg -d spd オペランドで指定したすべてのファイルグループに対して，pdlogadfg -d ssp オペランドを指定してください。

◆ pdlogadpf -d ssp -g ファイルグループ名 -a "副シンクポイントダンプファイル名" 〔-b "副シンクポイントダンプファイル名"〕

ファイルグループを構成する副シンクポイントダンプファイルを指定します。一つのファイルグループに対してこのオペランドを一つ指定します。

必ず pdlogadfg オペランド，pdlogadpf オペランドの順で指定してください。この順で指定しないとエラーになります。

-g ファイルグループ名：～＜識別子＞((1～8 文字))

pdlogadfg オペランドで指定したファイルグループ名を指定します。対応する正シンクポイントダンプファイルのファイルグループ名と同じ名称にしてください。

-a "副シンクポイントダンプファイル名"：～＜パス名＞((167 文字以内))

ファイルグループを構成する副シンクポイントダンプファイルの名称を絶対パス名で指定します。システムログ適用化で作成した副シンクポイントダンプファイルの名称を指定してください。なお，副シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

-b "副シンクポイントダンプファイル名"：～＜パス名＞((167 文字以内))

副シンクポイントダンプファイルを二重化（pd_spd_dual オペランドに Y を指定）する場合に，B 系の副シンクポイントダンプファイル名を絶対パス名で指定します。pd_spd_dual オペランドに Y を指定しない場合は，-b オプションに副シンクポイントダンプファイル名を指定しても無効になります。システムログ適用化で作成した副シンクポイントダンプファイルの名称を指定してください。なお，副シンクポイントダンプファイル名はユニット内で一意になるようにしてください。

《前提条件》

- 前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。
- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している必要があります。

《注意事項》

pdlogadfg -d ssp オペランドで指定したすべてのファイルグループに対して，pdlogadpf -d ssp オペランドを指定してください。

9.2.29　ステータスファイルに関するオペランド

(1)　ユニット用ステータスファイルに関するオペランド

◆ pd_syssts_file_name_1 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

：

pd_syssts_file_name_7 = "論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

ユニット用ステータスファイルを定義します。pd_syssts_file_name_2～7 オペランドは省略できますが，pd_syssts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～＜識別子＞((1～8 文字))

ユニット用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～＜パス名＞((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～＜パス名＞((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**になります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要になります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を指定できません。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_syssts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。
- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のないステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドでステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- **メリット及びデメリット**

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

  実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- **注意事項**

  実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_syssts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_syssts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_syssts_last_active_file オペランドに指定する必要があります。

《ほかのオペランドとの関連》

このオペランドは pd_syssts_subfile_name_1～7 オペランドと関連があります。

◆ pd_syssts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"

：

pd_syssts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に，ログ適用サイトで使用する，ユニット用副ステータスファイルを定義します。

"論理ファイル名"：～＜識別子＞((1～8 文字))

ユニット用副ステータスファイルの論理ファイル名を指定します。pd_syssts_file_name_1～7 と同じ論理ファイル名を指定してください。副ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系副ステータスファイル名"：～＜パス名＞((167 文字以内))

A 系の副ステータスファイル名を絶対パス名で指定します。

"B 系副ステータスファイル名"：～＜パス名＞((167 文字以内))

B 系の副ステータスファイル名を絶対パス名で指定します。

《前提条件》

- 前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。
- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している必要があります。

《指定値の目安》

- A 系及び B 系の副ステータスファイル名には，システムログ適用化で作成した副ステータスファイルを指定してください。システムログ適用化で作成していない副ステータスファイルを指定すると，その副ステータスファイルは**実体のない副ステータスファイル**になります。
- 副ステータスファイルに障害が発生すると，HiRDB は副ステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要になります。
- A 系と B 系には同じレコード長，及び同じ容量の副ステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- 業務サイトでは，このオペランドの指定は無視されます。
- HiRDB 開始時にシステム適用化を実施する場合，pd_syssts_file_name_1～7 に指定したオープン可能なすべての正ステータスファイルに対応する副ステータスファイルを作成してください。
- 副ステータスファイルは A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系副ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系副ステータスファイル名，及び B 系副ステータスファイル名には同じ名称を重複して指定できません。
- A 系及び B 系副ステータスファイル名には，pd_syssts_file_name_1～7 オペランドの，A 系及び B 系ステータスファイル名と同じ名称を指定できません。

《留意事項》

HiRDB を正常開始，又は再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，すべての副ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときはログ適用サイトを開始できなくなります。この場合は，システムログ適用化を実施してください。

●実体のない副ステータスファイルの使い方

pd_syssts_file_name_1～7 オペランドの説明を参照してください。

《ほかのオペランドとの関連》

このオペランドは pd_syssts_file_name_1～7 オペランドと関連があります。

(2) ユニット用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ pd_syssts_last_active_file = "論理ファイル名"

～＜識別子＞((1～8 文字))

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に現用にするステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_syssts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_syssts_file_name_1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_syssts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. **実体のないステータスファイルを指定している場合**

   現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I

- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，業務サイトでは，ユニット用正ステータスファイルが対象となります。ログ適用サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

◆ **pd_syssts_last_active_side = A ｜ B**

現用ファイルの片系に障害が発生している状態で HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_syssts_initial_error オペランドに continue，又は excontinue を指定している
- pd_syssts_last_active_file オペランドを指定している

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，業務サイトでは，正ステータスファイルが対象になります。ログ適用サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

◆ **pd_syssts_last_active_subfile = "論理ファイル名"**

～＜識別子＞((1～8 文字))

このオペランドは，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に指定します。HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に現用にする副ステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定

した現用ファイルを比較します。ファイルが一致する場合は HiRDB を開始しますが，一致しない場合は開始しません。

《前提条件》

前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。

次に示す条件を満たす必要があります。

- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している
- pd_syssts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **すべての副ステータスファイルを初期化した直後に HiRDB を開始する場合**

   システムログ適用化を実施してください。システムログ適用化が正常に完了したら，このオペランドを指定しないで，HiRDB を開始してください。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，すべての副ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_syssts_last_active_side_sub オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   システムログ適用化を実施してください。システムログ適用化が正常に完了したら，このオペランドを指定しないで，HiRDB を開始してください。

5. **実体のない副ステータスファイルを指定している場合**

   現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されている副ステータスファイルが現用ファイルになります。

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，ユニット用副ステータスファイルが対象になります。業務サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

◆ **pd_syssts_last_active_side_sub = A ｜ B**

このオペランドは，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に指定します。現用ファイルの片系に障害が発生している状態で HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB を開始しません。

《前提条件》

前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。

次に示す条件を満たす必要があります。

- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している
- pd_syssts_initial_error オペランドに continue，又は excontinue を指定している
- pd_syssts_last_active_subfile オペランドを指定している

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，ユニット用副ステータスファイルが対象になります。業務サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

(3) サーバ用ステータスファイルに関するオペランド

◆ pd_sts_file_name_1＝"論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

：

pd_sts_file_name_7＝"論理ファイル名"，"A 系ステータスファイル名"，"B 系ステータスファイル名"

サーバ用ステータスファイルを定義します。pd_sts_file_name_2～7 オペランドは省略できますが，pd_sts_file_name_1 オペランドは省略できません。

"論理ファイル名"：～＜識別子＞((1～8 文字))

サーバ用ステータスファイルの論理ファイル名を指定します。ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系ステータスファイル名"：～＜パス名＞((167 文字以内))

A 系のステータスファイル名を絶対パス名で指定します。

"B 系ステータスファイル名"：～＜パス名＞((167 文字以内))

B 系のステータスファイル名を絶対パス名で指定します。

《指定値の目安》

- A 系及び B 系のステータスファイル名には，pdstsinit コマンドで作成したステータスファイルを指定してください。pdstsinit コマンドで作成していないステータスファイルを指定すると，そのステータスファイルは**実体のないステータスファイル**となります。
- ステータスファイルに障害が発生すると，HiRDB はステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要となります。
- A 系と B 系には同じレコード長，及び同じ容量のステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- ステータスファイルは，A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系ステータスファイル名，及び B 系ステータスファイル名には同じ名称を重複して指定できません。

《留意事項》

- HiRDB を正常開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，全ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときは pd_sts_file_name_1～7 の中で最初に指定したステータスファイルが現用ファイルになります。残りのファイルのうちオープンできたファイルが予備ファイルになり，オープンできないファイルは予約ファイルになります。

- HiRDB を再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。

●実体のないステータスファイルの使い方

実体のないステータスファイルを指定すると，HiRDB の稼働中にステータスファイルを新規追加できます。例えば，ステータスファイルの障害などで予備ファイルが少なくなったときに，実体のないステータスファイルを予備ファイルにします。実体のないステータスファイルを予備ファイルにする手順を次に示します。

〈手順〉

1. pdstsinit コマンドで，システムファイル用の HiRDB ファイルシステム領域にステータスファイルを作成します。
2. pdstsopen コマンドで，ステータスファイルをオープンします。

この操作は HiRDB 稼働中に実行できます。HiRDB を一度停止する必要はありません。

- **メリット及びデメリット**

  実体のないステータスファイルを定義すると，HiRDB ファイルシステム領域の占有量は小さくなります。しかし，実体のないステータスファイルを予備ファイルとして追加するときに，システムファイル用の HiRDB ファイルシステム領域に十分な空き（ファイルを追加するだけの空き領域）がないと追加できないため，システムの信頼性は低くなります。

  実体のないステータスファイルを定義しないと，HiRDB ファイルシステム領域の占有量は大きくなるが，ファイル障害によるスワップ先が保証されるため，信頼性は高くなります。

- **注意事項**

  実体のないステータスファイルを定義した場合，HiRDB は HiRDB の開始時にステータスファイルに異常があると認識します。このため，pd_sts_initial_error オペランドに stop（省略値）を指定していると，HiRDB を開始できないので注意してください。実体のないステータスファイルを定義する場合は，pd_sts_initial_error オペランドに continue 又は excontinue を指定してください。また，HiRDB を開始する前に現用ファイルを pd_sts_last_active_file オペランドに指定する必要があります。

《ほかのオペランドとの関連》

このオペランドは pd_sts_subfile_name_1～7 オペランドと関連があります。

◆ pd_sts_subfile_name_1 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"

：

pd_sts_subfile_name_7 = "論理ファイル名"，"A 系副ステータスファイル名"，"B 系副ステータスファイル名"

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に，ログ適用サイトで使用する，サーバ用副ステータスファイルを定義します。

"論理ファイル名"：～＜識別子＞((1～8 文字))

サーバ用副ステータスファイルの論理ファイル名を指定します。pd_sts_file_name_1～7 と同じ論理ファイル名を指定してください。副ステータスファイルを操作するコマンドを実行するときに，ここで定義した論理ファイル名を指定します。

"A 系副ステータスファイル名"：～＜パス名＞((167 文字以内))

A 系の副ステータスファイル名を絶対パス名で指定します。

"B 系副ステータスファイル名"：～＜パス名＞((167 文字以内))

B 系の副ステータスファイル名を絶対パス名で指定します。

《前提条件》

- 前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。
- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している必要があります。

《指定値の目安》

- A 系及び B 系の副ステータスファイル名には，システムログ適用化で作成した副ステータスファイルを指定してください。システムログ適用化で作成していない副ステータスファイルを指定すると，その副ステータスファイルは**実体のない副ステータスファイル**になります。
- 副ステータスファイルに障害が発生すると，HiRDB は副ステータスファイルをスワップします。スワップ先となる予備ファイルがないと，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了します。したがって，システムファイルを多く定義するほど，システムの信頼性が向上します。ただし，その分のディスク容量が必要になります。
- A 系と B 系には同じレコード長，及び同じ容量の副ステータスファイルを指定してください。

《オペランドの規則》

- このオペランドは 7 個まで指定できます。
- 業務サイトでは，このオペランドの指定は無視されます。
- HiRDB 開始時にシステム適用化を実施する場合，pd_sts_file_name_1～7 に指定したオープン可能なすべての正ステータスファイルに対応する副ステータスファイルを作成してください。
- 副ステータスファイルは A 系と B 系に二重化されています。両方必ず指定してください。
- A 系及び B 系副ステータスファイル名の絶対パス名に環境変数を使用できません。
- 論理ファイル名，A 系副ステータスファイル名，及び B 系副ステータスファイル名には同じ名称を重複して指定できません。
- A 系及び B 系副ステータスファイル名には，pd_sts_file_name_1～7 オペランドの，A 系及び B 系ステータスファイル名と同じ名称を指定できません。

《留意事項》

HiRDB を正常開始，又は再開始する場合は現用ファイル（終了時点での現用ファイル）を引き継ぎます。ただし，すべての副ステータスファイルを初期化した場合など，引き継ぐ現用ファイルがないときはログ適用サイトを開始できなくなります。この場合は，システムログ適用化を実施してください。

●実体のない副ステータスファイルの使い方

pd_sts_file_name_1～7 オペランドの説明を参照してください。

《ほかのオペランドとの関連》

このオペランドは pd_sts_file_name_1～7 オペランドと関連があります。

(4) サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ pd_sts_last_active_file = "論理ファイル名" ～＜識別子＞((1～8 文字))

HiRDB（HiRDB/パラレルサーバの場合は該当するサーバ）の開始時に現用にするステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しますが，一致しない場合は開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **全ステータスファイルを初期化した直後に HiRDB を開始する場合**

   pd_sts_file_name1～7 オペランドに指定した最も小さい番号の正常な論理ファイル名を指定してください。この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，全ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_sts_last_active_side オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   全ステータスファイルを初期化してください。その後，1 の方法を実施してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

5. **実体のないステータスファイルを指定している場合**

   現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I
- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されているステータスファイルが現用ファイルになります。

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，業務サイトでは，サーバ用正ステータスファイルが対象になります。ログ適用サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

◆ **pd_sts_last_active_side = A ｜ B**

現用ファイルの片系に障害が発生している状態で HiRDB（HiRDB/パラレルサーバの場合は該当するサーバ）を開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。

《前提条件》

次に示す条件を満たす必要があります。

- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- pd_sts_last_active_file オペランドを指定している

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，業務サイトでは，サーバ用正ステータスファイルが対象になります。ログ適用サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

◆ **pd_sts_last_active_subfile = "論理ファイル名"**
～＜識別子＞((1～8 文字))

このオペランドは，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に指定します。HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に現用にする副ステータスファイルの論理ファイル名を指定します。HiRDB は，このオペランドに指定されたファイルと HiRDB が特定した現用ファイルを比較します。ファイルが一致する場合は HiRDB を開始しますが，一致しない場合は開始しません。ただし，pdrisedbto コマンドによってデータベースを引き継いだ直後に HiRDB を開始した場合は，このオペランドの指定は無視されます。

《前提条件》

前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。

次に示す条件を満たす必要があります。

- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している
- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- HiRDB システムが選択した現用ファイルが，前回までの実行で最新の現用ファイルかどうか確定できない

《指定値の目安》

1. **すべての副ステータスファイルを初期化した直後に HiRDB を開始する場合**

   システムログ適用化を実施してください。システムログ適用化が正常に完了したら，このオペランドを指定しないで，HiRDB を開始してください。

2. **現用ファイルの両系とも正常な場合**

   現用ファイル名※を指定してください。現用ファイル名を指定したのに HiRDB が開始できない場合は，現用ファイルを初期化した可能性があります。この場合，すべての副ステータスファイルを初期化した後に，1 の方法で HiRDB を開始してください。なお，この場合，前回の終了モードに関係なく強制開始となるので注意が必要です。

3. **現用ファイルの片系に異常がある場合**

   2 の方法のほかに次に示すオペランドを指定してください。

   ・pd_syssts_singleoperation オペランドに continue を指定

   ・pd_sts_last_active_side_sub オペランドを指定

4. **現用ファイルの両系に異常がある場合**

   システムログ適用化を実施してください。システムログ適用化が正常に完了したら，このオペランドを指定しないで，HiRDB を開始してください。

5. **実体のない副ステータスファイルを指定している場合**

   現用ファイル名※を指定してください。

注※

現用ファイル名（終了時点での現用ファイル）は次に示すメッセージから分かります。

- KFPS01001-I

- KFPS01010-E
- KFPS01011-I
- KFPS01063-I

これらのメッセージのうちで，最も直前に出力されたメッセージに表示されている副ステータスファイルが現用ファイルになります。

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，サーバ用副ステータスファイルが対象になります。業務サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

◆ **pd_sts_last_active_side_sub = A ｜ B**

このオペランドは，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合に指定します。現用ファイルの片系に障害が発生している状態で HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始する場合にこのオペランドを指定します。このオペランドには正常な系を指定します。HiRDB は，このオペランドに指定された系と HiRDB が特定した系を比較します。系が一致する場合は正常な系の内容を予備の A 系，及び B 系ファイルに複写します。その後，予備を現用に切り替えて HiRDB を開始します。系が一致しない場合は HiRDB を開始しません。また，pdrisedbto コマンドによってデータベースを引き継いだ直後に HiRDB を開始した場合は，このオペランドの指定は無視されます。

《前提条件》

前提プラットフォームについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。

次に示す条件を満たす必要があります。

- pd_rise_use オペランドに Y を指定し，かつ pd_rise_pairvolume_combination オペランドに syssync を指定している
- pd_sts_initial_error オペランドに continue，又は excontinue を指定している
- pd_sts_last_active_subfile オペランドを指定している

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，サーバ用副ステータスファイルが対象になります。業務サイトでは，このオペランドを指定しても無視されます。
- 障害発生時にこのオペランドを指定した場合は，障害の復旧が完了し障害の要因を取り除いた後，HiRDB を正常停止してこのオペランドの指定を省略してください。障害から復旧した後もこのオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

9.2.30　メッセージログファイルに関するオペランド

◆ **pd_mlg_msg_log_unit = manager ｜ local**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

メッセージログの出力先ユニットを指定します。

manager：

システムマネジャユニットのメッセージログファイル及び syslogfile に，メッセージログを出力します。メッセージログファイルはシステムマネジャのサーバマシンの$PDDIR/spool 下に作成されます。

local：

メッセージ出力元のユニットのメッセージログファイル及び syslogfile に，メッセージログを出力します。メッセージログファイルは，各サーバマシンの$PDDIR/spool 下に作成されます。

《指定値の目安》

- システムマネジャユニットでメッセージログを一元管理したい場合は，このオペランドに manager を指定してください。
- システムマネジャユニットが障害などで停止している場合，メッセージログはメッセージ発行元のユニットの syslogfile に一定時間遅延して出力されたり，又はメッセージ自体が出力されなかったりします。このオペランドに local を指定すると，メッセージを遅延することなくメッセージ出力元のユニットに出力できます。メッセージログの出力分散化については，マニュアル「HiRDB システム運用ガイド」を参照してください。

このオペランドの値とメッセージログの出力先の関係を次に示します。

pd_mlg_msg_log_unit の値	条件	メッセージログの出力先
manager	通常時	システムマネジャがあるサーバマシンのメッセージログファイル及び syslogfile に出力されます。
	システムマネジャユニットの障害時又は通信障害時	各サーバマシンの syslogfile に出力されます。出力されないメッセージログがあります。また，メッセージログが遅れて出力されることがあります。
local	通常時	各サーバマシンのメッセージログファイル及び syslogfile に出力されます。
	システムマネジャユニットの障害時又は通信障害時	

《ほかのオペランドとの関連》

このオペランドに local を指定した場合，メッセージログファイルの容量を見積もり直し，その結果を pd_mlg_file_size オペランドに指定してください。

《各見積もり式への影響》

pd_mlg_msg_log_unit オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「ステータスファイルの容量の見積もり」の「S の求め方」

◆ **pd_mlg_file_size = メッセージログファイルの最大容量**
～＜符号なし整数＞((64～1000000))《10240》（単位：キロバイト）

メッセージログファイルの最大容量をキロバイト単位で指定します。

メッセージログファイルは二つあり，メッセージの量がここで指定した最大容量に達した場合，メッセージログファイルをスワップします。

pd_mlg_msg_log_unit オペランドに local を指定した場合，各サーバマシンにここで指定した容量のメッセージログファイルが作成されます。

9.2.31 統計情報に関するオペランド

◆ **pd_statistics = A** ｜ **Y** ｜ **N**

- 0904 互換モードの場合：**《N》**

HiRDB の開始時から統計ログを取得するかどうかを指定します。

A：HiRDB の開始時から「システムの稼働に関する統計情報（出力間隔 1 分）」，「グローバルバッファに関する統計情報」，「デファードライト処理に関する統計情報」「データベースの入出力に関する統計情報（出力間隔 1 分）」を取得します。取得される統計情報は，pdstbegin コマンド -k sys,buf,dfw,dio -m 1 -I 60 相当となります。

Y：HiRDB の開始時から「システムの稼働に関する統計情報（出力間隔 10 分）」を取得します。取得される統計情報は，pdstbegin コマンド -k sys -m 10 相当となります。

N：HiRDB の開始時から統計ログを取得しません。

Y を指定した場合，取得する統計ログはユニット全体のシステムの稼働に関する統計情報だけです。サーバ別の情報，又はシステムの稼働に関する統計情報以外の情報が必要なときは，HiRDB の開始完了後に一度 pdstend コマンドで統計情報取得を停止し，改めて pdstbegin コマンドで統計情報取得を開始する必要があります。

なお，N を指定した場合でも，pdstbegin コマンドを入力すれば，HiRDB の稼働中に統計ログを取得できます。

また，次の条件を満たす場合，統計情報を取得する設定をしても統計情報は取得されません。

- 影響分散スタンバイレス型系切り替え機能を使用したユニットで pd_statistics に Y を指定した場合（pd_statistic が A の場合は取得されます）
- 影響分散スタンバイレス型系切り替え機能を使用したユニットでユニット内に起動しているサーバがない場合
- ユニット内のサーバをすべて正常停止した場合
- HiRDB/パラレルサーバでシステムマネジャだけのユニットの場合（同じユニットにフロントエンドサーバ，バックエンドサーバ，又はディクショナリサーバがない）

- ログ同期方式のリアルタイム SAN レプリケーションを使用している，ログ適用サイトの場合

《指定値の目安》

性能に関するチューニングをする際に，速やかに原因を切り分けられるようにするため，A の指定を推奨します。

《注意事項》

pdstbegin と pd_statistics を両方指定した場合，pd_statistics の設定が無効になります。
pd_statistics の設定に従って統計情報を取得する場合，pdstbegin は設定しないでください。

◆ pd_stj_file_size = 統計ログファイルの最大容量

～＜符号なし整数＞((64～1000000))《102400》（単位：キロバイト）

- 0904 互換モードの場合：**《1024》**

統計ログファイルの最大容量をキロバイト単位で指定します。統計ログファイルは二つあり，統計ログの出力量がここで指定した最大容量に達したときに統計ログファイルをスワップします。

《指定値の目安》

- このオペランドの指定値の見積もり方法については，「統計ログファイル（pd_stj_file_size）の見積もり式」を参照してください。
- pd_stj_buff_size オペランドとの間に次に示す関係が保たれるように指定してください。

  pd_stj_file_size≧pd_stj_buff_size×2

◆ pd_stj_buff_size = 統計ログバッファ長

～＜符号なし整数＞((32～512))《32》（単位：キロバイト）

統計ログバッファの大きさを指定します。

《指定値の目安》

次に示す条件によって指定値の目安が異なります。なお，複数の条件に該当する場合は，各計算式で算出した値のうち大きい方の値を指定してください。

1. pdstbegin コマンドの-k オプションに pcd，又は obj を指定する場合

   pcd，obj それぞれの統計ログ出力量を次に示す計算式で算出し，算出値の合計に 32 を加えた値を指定してください。ただし，この計算式で算出した値が 512 を超える場合は 512 を指定してください。

   （a÷1024）×（0.03÷b）

   a：統計ログ出力量（バイト）

   「統計ログファイル（pd_stj_file_size）の見積もり式」を参照してください。

   b：pdstend コマンド入力時刻−pdstbegin 入力時刻（秒）

2. pdstbegin コマンドの-k オプションに dio，-D オプションに everyfile を指定する場合

   次に示す計算式で算出した値に 32 を加えた値を指定してください。

   {(a×104）＋（↑a÷16↑×204)}÷1024÷b×60

   a：ユニット内の出力対象サーバが管理する RD エリア構成ファイル数の合計

   b：pdstbegin コマンドの-I オプションに指定した出力時間間隔

ただし，この計算式で算出した値が 512 を超える場合は，pdstbegin コマンドの-D オプションに everyfile の指定を行わないでください。

3. 1.と 2.のどちらにも該当しない場合

   省略値の 32 を指定してください。

◆ pd_sqlobject_stat_timing = deallocate | tran

SQL オブジェクト用バッファの統計情報を取得するタイミングを指定します。統計情報を取得するタイミングと取得対象となる統計情報を次の表に示します。

項番	統計情報を取得するタイミング	取得対象となる統計情報	
		deallocate	tran
1	次の SQL 文を実行することで，前処理結果が無効になるとき • DEALLOCATE PREPARE 文 • PREPARE 文 • EXECUTE IMMEDIATE 文	無効になる前処理結果	無効になる前処理結果
2	コミットしたとき	コミットによって無効になる前処理結果	すべての前処理結果
3	ロールバックしたとき	取得しない	すべての前処理結果
4	DISCONNECT 文，RELEASE 指定ありの COMMIT 文，定義系 SQL を実行したとき	すべての前処理結果	すべての前処理結果
5	RELEASE 指定ありの ROLLBACK 文を実行したとき	RELEASE 指定ありの ROLLBACK 文を実行した時点で，ロールバックによって無効にならない前処理結果	すべての前処理結果

《指定値の目安》

次の場合は，tran を指定します。

- ホールダブルカーソルや，JDBC ドライバで STATEMENT_COMMIT_BEHAVIOR に TRUE を指定した場合など，前処理結果がトランザクションを超えて有効になる場合
- ロールバック時に SQL オブジェクト用バッファの統計情報を取得したい場合

上記以外の場合は，deallocate を指定します。

《注意事項》

- サーバプロセスのキャンセルに伴うトランザクション決着時は，統計情報を取得できません。
- ホールダブルカーソルや，JDBC ドライバで STATEMENT_COMMIT_BEHAVIOR に TRUE を指定した場合は，前処理結果はトランザクションを超えても有効です。そのため，このオペランドに deallocate を指定した場合は，トランザクションが決着してもこれらの SQL オブジェクト用バッファの統計情報を取得できません。
- ホールダブルカーソルや，JDBC ドライバで STATEMENT_COMMIT_BEHAVIOR に TRUE を指定した場合は，そのカーソルが閉じている場合だけ SQL オブジェクト用バッファの統計情報を取得できます。

ホールダブルカーソル，及び STATEMENT_COMMIT_BEHAVIOR については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ pdstbegin

●HiRDB/シングルサーバの場合のオペランド指定形式

```
pdstbegin 〔-k 統計情報種別〔, 統計情報種別〕…〕
        〔-m システム稼働統計情報出力時間間隔〕
        〔-I データベースの入出力統計情報出力時間間隔〕
        〔-D データベースの入出力統計情報オプション
         〔, データベースの入出力統計情報オプション〕〕
        〔-w〕
```

●HiRDB/パラレルサーバの場合のオペランド指定形式

```
pdstbegin 〔-k 統計情報種別〔, 統計情報種別〕…〕
        〔-m システム稼働統計情報出力時間間隔〕
        〔-I データベースの入出力統計情報出力時間間隔〕
        〔-D データベースの入出力統計情報オプション
         〔, データベースの入出力統計情報オプション〕〕
        〔 {-x ホスト名 | -u ユニット識別子} 〕
        〔 {-a | -s サーバ名〔, サーバ名〕…} 〕
        〔-w〕
```

HiRDB の開始時から統計情報を取得する場合に指定します。統計情報の出力を中止するときは pdstend コマンドを実行してください。出力先ファイルは統計ログファイルになります。

-k 統計情報種別〔，統計情報種別〕...

～≪sys≫

出力する統計情報の種別を指定します。なお，統計情報の出力種別の指定状況は，pdls -d stj コマンドで確認できます。

統計情報の種類（-k オプションの指定）	出力する統計情報の種類	統計情報の出力契機
sys	システムの稼働に関する統計情報	-m オプションに指定した時間間隔ごとに統計情報を出力します。
uap	UAP に関する統計情報	HiRDB への接続時及び HiRDB からの切り離し時に統計情報を出力します。
sql	SQL に関する統計情報	SQL の実行開始時及び SQL の実行終了時に統計情報を出力します。
buf	グローバルバッファに関する統計情報※1	シンクポイント時に統計情報を出力します。
fil	データベース操作に関する HiRDB ファイルの統計情報	
dfw	デファードライト処理に関する統計情報	
idx	インデクスに関する統計情報	

統計情報の種類（-k オプションの指定）	出力する統計情報の種類	統計情報の出力契機
sop	SQL 静的最適化に関する統計情報	動的 SQL の前処理時，及び静的 SQL の前処理時に，SQL オブジェクトバッファにヒットしなかった場合に統計情報を出力します。
dop	SQL 動的最適化に関する統計情報	FETCH 文及び CLOSE 文以外の SQL 文を実行したときに統計情報を出力します。
pcd	SQL オブジェクト実行に関する統計情報	SQL オブジェクトの実行時に統計情報を出力します。
sqh	SQL 文の履歴に関する統計情報※2	PREPARE 時，及び埋込み型 SQL 文の実行時に統計情報を出力します。
obj	SQL オブジェクト転送に関する統計情報※3	SQL オブジェクトの実行時に統計情報を出力します。
dio	データベースの入出力に関する統計情報	-I オプションに指定した時間間隔（秒）ごとにデータベースへの入出力の情報を出力します。

注※1

これらの統計情報はシンクポイント時に取得され，シンクポイント間の情報が編集されます。このため，シンクポイントが２回以上発生しないと統計情報が取得されません。これらの統計情報を確実に取得するためには，pdstend コマンド実行直前に，pdlogswap 又は pdlogsync コマンドを実行してシンクポイントを発生させてください。

注※2

SQL 文の履歴に関する統計情報は，SQL に関する統計情報（sql 指定）を出力している場合に出力されます。

注※3

HiRDB/パラレルサーバ限定の統計情報です。HiRDB/シングルサーバの場合は出力しません。

-m システム稼働統計情報出力時間間隔

～＜符号なし整数＞ ((1～1440)) ≪10≫（単位：分）

このオプションは「システムの稼働に関する統計情報」を取得する場合（-k オプションで sys を指定した場合）に指定します。

「システムの稼働に関する統計情報」を統計ログファイルに出力する間隔を分単位で指定します。

-I データベースの入出力統計情報出力時間間隔

～＜符号なし整数＞ ((10～3600)) ≪60≫（単位：秒）

このオプションは「データベースの入出力に関する統計情報」を取得する場合（-k オプションで dio を指定した場合）に指定します。

「データベースの入出力に関する統計情報」を統計ログファイルに出力する時間間隔を秒単位で指定します。

指定時間間隔内にデータベースへの入出力が 1 回も発生しなかった場合は，統計ログファイルを出力しません。

-k オプションに dio の指定がない場合は，このオプションを無視します。

-D データベースの入出力統計情報オプション

〔，データベースの入出力統計情報オプション〕

データベースの入出力に関する統計情報の計測動作を制御するオプションです。-k オプションに dio の指定がない場合は，このオプションを無視します。

このオプションでは，次の項目を制御できます。

- 入出力時間の計測頻度
- 統計情報の出力量

各項目に対応するオプション値指定有無による動作差異を次の表に示します。

表 9–7　入出力時間の計測頻度

指定値	動作	メリット	デメリット
指定なし	RD エリアを構成する各 HiRDB ファイルに発生した入力と出力に対して，それぞれ発生順に秒間 100 回の入出力時間を計測します。	性能影響を最小限に抑えられます。	RD エリアを構成する各 HiRDB ファイルに発生した入力と出力に対して，それぞれ秒間 101 回目以降については入出力時間を計測しません。
everyio	RD エリアを構成する HiRDB ファイルに発生した入出力に対して，すべての入出力時間を計測します。	入出力情報を漏れなく取得できます。	入出力時間計測のオーバヘッドによって，トランザクション性能が劣化します。

everyio 指定時は，RD エリアを構成する HiRDB ファイルに発生した入出力に対して，すべての入出力時間を計測するため，オーバヘッドが増加します。事前にテストをして，本番業務に性能上問題がないと判断した場合だけ指定してください。

表 9–8　統計情報の出力量

指定値	動作	メリット	デメリット
指定なし	• 出力時間間隔内にアクセスのあった RD エリアを構成する HiRDB ファイルの入出力情報のうち，統計情報として取得する次の項目ごとに，それぞれのユニットで上位 10 ファイルの情報を統計ログに出力します。 ・read 最大回数 ・read 最大時間 ・read 合計時間 ・write 最大回数 ・write 最大時間 ・write 合計時間 • 同じファイルが複数の項目で出力対象になった場合，重複排除しないで出力します。	1 回の出力契機の統計ログサイズが固定であるため，ディスク消費量が少なく，統計ログバッファのスワップ処理多発による性能への影響が少なくなります。	それぞれのユニットで上位 10 ファイルの情報しか出力しないため，RD エリアを構成する HiRDB ファイルすべての入出力の情報は取得できません。

指定値	動作	メリット	デメリット
	• 同順となるレコードが複数あり，10 件を超える場合は，次に示す順序が早いものを優先します。 1. RD エリア定義順 2. HiRDB ファイル指定順		
everyfile	出力時間間隔内にアクセスのあった RD エリアを構成する HiRDB ファイルの入出力情報をすべて出力します。	RD エリアを構成する HiRDB ファイルすべての入出力を取得できます。	1 回の出力契機の統計ログサイズが固定でないため，ディスク消費量が多く，統計ログバッファのスワップ処理多発による性能への影響が大きくなります。

everyfile 指定時は，1 回の出力契機での統計ログの出力量が固定ではありません。「システム共通定義の pd_stj_file_size オペランド」には，「統計ログファイル（pd_stj_file_size）の見積もり式」で見積もった値を指定してください。

「システム共通定義の pd_stj_buff_size オペランド」には，このオペランドで示す見積もり式で見積もった値を指定してください。

-x ホスト名

～＜ホスト名＞ ((1～32 文字))

HiRDB/パラレルサーバの場合に，統計情報の出力対象となるユニットを限定するとき，pdunit オペランドの-x オプションに指定したホスト名を指定します。統計情報の出力対象となるユニットのホスト名を指定してください。

このユニットに影響分散スタンバイレス型系切り替えを適用している場合は受け入れ中のゲスト BES の統計情報も出力します。

-u ユニット識別子

～＜識別子＞ ((4 文字))

HiRDB/パラレルサーバのときに，統計情報の出力対象となるユニットを限定する場合に指定します。統計情報の出力対象となるユニットのユニット名を指定します。

このホスト上のユニットに影響分散スタンバイレス型系切り替えを適用している場合は受け入れ中のゲスト BES の統計情報も出力します。

-a

HiRDB/パラレルサーバのときに，統計情報の出力対象となるサーバを限定しない場合に指定します。このオプションを指定すると，全サーバが統計情報の出力対象となります。

-s サーバ名〔，サーバ名〕...

～＜識別子＞ ((1～8 文字))

HiRDB/パラレルサーバのときに，統計情報の出力対象となるサーバを限定する場合に指定します。統計情報の出力対象となるサーバのサーバ名を指定します。

サーバの種類によっては統計情報の出力対象にならない場合があります。統計情報を出力するサーバの種類を次に示します。

統計情報の種類 （-k オプションの指定）	サーバの種類		
	FES	DS	BES
sys	○	○	○
uap	○	×	×
sql	○	×	×
buf	×	○	○
fil	×	○	○
dfw	×	○	○
idx	×	○	○
sop	○	×	×
dop	○	×	×
pcd	○	○	○
sqh	○	×	×
obj	×	○	○
dio	×	○	○

（凡例）

○：統計情報を出力します。

×：統計情報を出力しません。

-w

スレッド間ロック待ち時間の統計情報を出力する場合に指定します。ただし，このオプションはシステムの稼働に関する統計情報を出力する場合に有効となります。そのため，スレッド間ロック待ち時間の統計情報を取得できるのは，-k オプションの指定を省略，又は-k オプションに sys を含む指定をする場合です。

なお，このオプションを指定すると，システム全体の性能に影響を与えることがあるため，通常は指定しないでください。

《指定値の目安》

- pd_statistics オペランドに A を指定，又は省略値として A が仮定されている場合（推奨モード）は，設定不要です。pd_statistics オペランドが A の場合に取得する統計情報を次に示します。

  ・システムの稼働に関する統計情報（1 分間隔）

  ・グローバルバッファプールに関する統計情報

  ・デファードライト処理に関する統計情報

  ・データベースの入出力に関する統計情報

  なお，上記以外の統計情報も取得する場合は，上記の統計情報と追加取得する統計情報の指定を推奨します。

（指定例）

UAP に関する統計情報を追加取得する場合の指定例を次に示します。

```
pdstbegin -k sys,buf,dfw,dio,uap -m 1
```

- pd_statistics オペランドに Y を指定，又は N 指定若しくは省略値として N が仮定されている場合（互換モード）は，次に示す統計情報を取得する指定を推奨します。

  ・システムの稼働に関する統計情報（1 分間隔）

  ・グローバルバッファプールに関する統計情報

  ・デファードライト処理に関する統計情報

  ・データベースの入出力に関する統計情報

  （指定例）

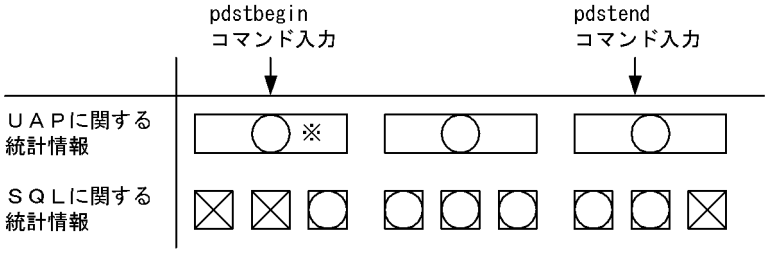
  ```
  pdstbegin -k sys,buf,dfw,dio -m 1
  ```

《ほかのオペランドとの関連》

- このオペランドは，pd_stj_file_size と関連があります。
- このオペランドを指定すると，pd_statistics オペランドの指定は無視されます。

《注意事項》

- pdstbegin オペランドは一つだけ指定できます。二つ以上指定した場合は一つ目の指定が有効になり，二つ目以降は無視されます。
- pdstbegin オペランドを指定した場合は，pdstend コマンドを入力しないと，HiRDB を終了するまで統計情報が出力されます。
- 1：1 スタンバイレス型系切り替え機能を使用している場合で，代替 BES ユニットに系を切り替えるときはこのオペランドの指定が無効になります。
- 影響分散スタンバイレス型系切り替え機能を使用している場合の注意事項です。HiRDB の正常開始時，各サーバはこのオペランドの指定値に従います。HiRDB の再開始時（系切り替えを含む）はこのオペランドの指定値が無効になります。再開始時は前回の統計情報の取得状態を引き継ぎます。系の切り替え時は切り替え前の統計情報の取得状態を引き継ぎます。
- ユニット内に開始しているサーバが存在しない場合は，ユニットの統計情報は取得されません。
- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは統計情報を取得できません。このオペランドを指定した場合，KFPS04689-W メッセージが出力されます。
- pdstbegin 及び pdstend コマンドの入力タイミングによっては，UAP に関する統計情報と SQL に関する統計情報の取得情報が一致しないことがあります。pdstbegin 及び pdstend コマンドの入力タイミングと出力される統計情報の関係を次に示します。



（凡例）○：統計情報を出力します。
×：統計情報を出力しません。
注※　ＯＬＴＰ環境下では出力されません。

- HiRDB，ユニット，又はサーバを終了（異常終了を含む）して，その後，開始した場合，開始後に統計情報の取得状態が引き継がれないことがあります。pd_statistics オペランド又はこのオペランドを指定することで，再開始又は系切り替え時に統計情報を取得し続けられます。HiRDB，ユニット，又はサーバ開始時の，統計情報の取得状態の引き継ぎ有無を次に示します。

開始モード	統計情報の取得環境	開始条件			
		HiRDBの開始	ユニット単位の開始（ユニット単独開始）	サーバ単位の開始（サーバ単独開始）	
				影響分散スタンバイレス型系切り替え機能を使用していない場合	影響分散スタンバイレス型系切り替え機能を使用している場合
正常開始	pdstbegin 又は pd_statistics オペランドを指定して統計情報を取得している場合	○	○	×※	○
	pdstbegin コマンドを実行して統計情報を取得している場合	×※	×※	×※	×※
再開始	pdstbegin 又は pd_statistics オペランドを指定して統計情報を取得している場合	○	○	－	△
	pdstbegin コマンドを実行して統計情報を取得している場合	×※	×※	－	○

（凡例）

○：統計情報の取得状態を引き継ぎます。

△：統計情報の取得状態を引き継ぎます。pdstbegin コマンド実行時はコマンド指定の統計情報の取得状態を引き継ぎます。

×：統計情報の取得状態を引き継ぎません（統計情報を取得しない状態になります）。

－：該当しません。

注※

統計情報を取得する場合，HiRDB の開始後に pdstbegin コマンドを実行する必要があります。

◆ **pdhibegin -k 履歴情報種別**

HiRDB の開始時から履歴情報を取得する場合に指定します。このオペランドを指定すると，HiRDB が終了するまで履歴情報を出力し続けます。出力先ファイルはシステムログファイルになります。

-k 履歴情報種別：

出力する履歴情報種別を指定します。

cnc：CONNECT/DISCONNECT に関する統計情報

HiRDB/パラレルサーバの場合，出力先ファイルはフロントエンドサーバのシステムログファイルになります。

9.2.32 RPC トレース情報に関するオペランド

◆ **pd_rpc_trace = Y ｜ N**

RPC トレース情報を取得するかどうかを指定します。RPC トレース情報には，HiRDB の保守情報が出力されます。

通常，このオペランドは省略してください。

Y：RPC トレースを取得します。

N：RPC トレースを取得しません。

《注意事項》

このオペランドに Y を指定すると，通信性能が低下します。

◆ **pd_rpc_trace_name = "RPC トレースを取得するファイル名称"**

～＜ 254 文字以内のパス名＞《$PDDIR/spool/rpctr》

RPC トレースファイルの名称を絶対パス名で指定します。RPC トレースファイルとして，指定したファイル名の後に 1，2，l（エル）を付けた三つのファイルが作成されます。

《注意事項》

- このオペランドで指定したディレクトリ下に，最大「pd_rpc_trace_size の値×２」の大きさのファイルが作成されます。ファイル容量に注意してください。
- システム共通定義ファイルを共用化していて，サーバマシンごとに異なるパスに RPC トレースを出力する場合，このオペランドは各ユニット制御情報定義で指定してください。

◆ **pd_rpc_trace_size = RPC トレースを取得するファイルの容量**

～＜符号なし整数＞((1024～2147483648))《4096》（単位：バイト）

RPC トレースファイルの容量をバイト単位で指定します。

このオペランド指定値は，"RPC トレースファイル"1 及び"RPC トレースファイル"2 のそれぞれの容量となります。

《指定値の目安》

RPC トレースは，すべてのプロセス間の通信電文が一つのファイルに時系列に蓄積されます。その量がこの指定値を超えると，もう一方のファイルにトレースデータの出力を切り替えます。古いト

レースは上書きされて消去されます。このため，指定値が小さいと十分なトレースを取得できなくなり，トラブルシュートが困難になることがあります。したがって，このオペランドには 1,000,000 以上の値を指定してください。

《注意事項》

このオペランドで指定した値は RPC トレースファイル l（エル）の容量に影響を与えません。RPC トレースファイル l（エル）の容量は 0 バイト固定です。

9.2.33　PRF トレース情報に関するオペランド

◆ **pd_prf_trace = Y | N**

PRF トレース情報を取得するかどうかを指定します。PRF トレース機能については，マニュアル「HiRDB システム運用ガイド」の「PRF トレース機能」を参照してください。

Y：

PRF トレース情報を取得します。

N：

PRF トレース情報を取得しません。PRF トレース情報を取得しない場合，pdprfd プロセスを開始しません。

《指定値の目安》

通常，このオペランドを指定する必要はありません。障害発生時の原因調査が困難になるため，N は指定しないでください。

《注意事項》

- このオペランドに N を指定した場合，HiRDB 稼働中に PRF トレース情報を取得するように変更することはできません。PRF トレースを取得するには，HiRDB をいったん終了させて，PRF トレースの取得を有効にしてから，再度 HiRDB を開始する必要があります。
- Linux でこのオペランドに Y を指定または省略した場合，HiRDB は開始時にトラブルシュート情報取得用共用メモリを/dev/shm 下に配置するため，/dev/shm の容量が不足すると HiRDB の開始に失敗します。使用している OS のマニュアルを参照して，/dev/shm の最大容量が次の計算式を満たしているか確認してください。/dev/shm の最大容量が小さい場合は，計算式を満たすように容量を変更してください。

  /dev/shm の最大容量　≧　稼働させる HiRDB のユニット数 × 10 メガバイト

◆ **pd_prf_level = 00000007 | 0000001f | 0000007f | 000001ff | 00000000**

PRF トレース情報の取得レベルを 16 進数で指定します。PRF トレース取得レベルによって取得できる情報については，マニュアル「HiRDB システム運用ガイド」の「PRF トレース情報の詳細と取得ポイント」を参照してください。

pd_prf_level オペランドの指定値に対する取得レベルと取得内容を次に示します。

表 9–9　pd_prf_level オペランドの指定値に対する取得レベルと取得内容

pd_prf_level オペランドの指定値	取得レベル	位置づけ	取得内容
00000007 （最小レベル）	00000004	取得内容を最小限に絞ったレベル	主に CONNECT 時，ユティリティ実行時，特定のイベントのトレースを取得します。
0000001f （標準レベル）	00000004， 00000010	HiRDB との境界（入口と出口）を識別できるレベル	最小レベルの取得内容に加えて，主に HiRDB との通信の前後でトレースを取得します。
0000007f （詳細レベル）	00000004， 00000010， 00000040	SQL 処理のトレースに必要なレベル	標準レベルの取得内容に加えて，HiRDB の内部処理の流れが確認できるトレースを取得します。
000001ff （保守レベル）	00000004， 00000010， 00000040， 00000100	障害発生時などの保守情報を取得するためのレベル	詳細レベルの取得内容に加えて，内部排他などの繰り返し処理も含めたトレースを取得します。
00000000 （抑止レベル）	取得しない	トレースの取得を抑止するレベル	取得しません。

《指定値の目安》

- 性能に関する障害調査をする際に，速やかに原因を切り分けられるようにするため，0000001f（標準レベル）の指定を推奨します。また，環境構築時などオンライン性能の低下が許容できる場合は，0000007f（詳細レベル）の指定を検討してください。
- 000001ff（保守レベル）は，通常は指定する必要はありません。サポートサービスから依頼があった場合に指定してください。
- 障害発生時の原因調査が困難になるため，00000000（抑止レベル）は指定しないでください。

《注意事項》

- PRF トレースの取得箇所を増やし，より詳細な PRF トレースが取得できるレベルに変更した場合，オンライン性能が低下する場合があります。
- 性能検証や障害調査が困難になるため，00000007，0000001f，0000007f，000001ff，00000000 以外の値を指定しないように注意してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_prf_file_count
- pd_prf_file_size

《備考》

pdprflevel コマンドを使用することで，HiRDB 稼働中でも取得レベルを変更できます。

◆ pd_prf_file_count＝PRF トレース情報ファイルの世代数

～＜符号なし整数＞((3～256))《15》

PRF トレース情報ファイルの世代数を指定します。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- PRF トレース情報ファイルの情報を長時間保持したい場合に，このオペランドの指定を検討してください。次に示す PRF トレース情報ファイルの総容量から，PRF トレース情報ファイルの世代数及びファイルサイズを見積もってください。

  PRF トレースファイルの総容量＝ pd_prf_file_count オペランドの指定値×pd_prf_file_size オペランドの指定値

  （例）

  PRF トレース情報を 1 時間保存するための，PRF トレース情報ファイルの総容量の設定例を次に示します。

pd_prf_level オペランドの指定値	PRF トレース情報ファイルの総容量	
	1 秒当たりの SQL 実行数が 200 件程度	1 秒当たりの SQL 実行数が 500 件程度
00000007	750MB（省略値※）	750MB（省略値※）
0000001f	750MB（省略値※）	2GB
0000007f	2GB	5GB

注※　pd_prf_file_count オペランドの省略値×pd_prf_file_size オペランドの省略値です。

- 1 ユニット当たりに必要な PRF トレース情報ファイルの総容量を概算で見積もるには，次の考え方を目安に計算してください。

  PRF トレース情報ファイルの総容量＝ MAX（768000，ａ×ｂ×ｃ×ｄ）（単位：キロバイト）

  ａ：1 ユニットの FES 及び BES のサーバ数

  HiRDB/シングルサーバの場合，及び MGR，DS だけの場合は 1。

  ｂ：サーバごとの 1 秒当たり SQL 実行数

  SQL に関する統計情報の「SQL 実行回数」の合計値「TOTAL」を，編集時間「EDIT TIME」（秒）で割ることで，編集時間における平均値を計算できます。

  ｃ：1SQL 当たりの平均 PRF トレースの取得量（単位：キロバイト）

  pd_prf_level オペランドの指定値に応じて，次の値を仮定してください。

  ・00000007：0.5 キロバイト

  ・0000001f：0.5 キロバイト

  ・0000007f：1 キロバイト

  ただし，上記の値は，SQL の種類，データ量，システムの構成，機能の適用有無，チューニングの状態などの条件によって変動します。

  ｄ：PRF トレース情報を保持する時間（単位：秒）

- トランザクションの種類や HiRDB の構成などによって，1 ファイル当たりの PRF トレース情報を保持できる時間が大きく変わることがあります。正確に判定するには，想定されるトランザクション及びユティリティを実行したときに，1 ファイル当たりに PRF トレースが保持されている時間を測定してください。時間はファイルのタイムスタンプ又は pdprfed コマンドでの編集結果から判断できます。測定した時間を基に，必要な世代数を計算してください。

《注意事項》

- このオペランドの指定値は，HiRDB 運用ディレクトリ下のディスク所要量に影響します。PRF トレース情報ファイルのために，次のディスク容量が必要になります。

  pd_prf_file_count オペランドの指定値×pd_prf_file_size オペランドの指定値
- このオペランドは，pdinfoget コマンドや pdgeter コマンドなどで取得する，障害情報ファイルの容量に影響があります。
- このオペランドの指定値を小さい値に変更した場合，減らした分の PRF トレース情報ファイルは使用されなくなりますが，自動的に削除されません。不要な PRF トレース情報ファイルを削除する必要がある場合は，HiRDB を終了状態にして削除を行ってください。

《ほかのオペランドとの関連》

このオペランドは pd_prf_file_size オペランドと関連があります。

◆ **pd_prf_file_size = PRF トレース情報ファイルサイズ**
～<符号なし整数>((1024～1048576))《51200》（単位：キロバイト）

一つの PRF トレース情報ファイルのサイズを指定します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。PRF トレース情報ファイルに情報を長時間保持したい場合に，このオペランドの指定を検討してください。

《注意事項》

- このオペランドの指定値は，HiRDB 運用ディレクトリ下のディスク所要量に影響します。PRF トレース情報ファイルのために，次のディスク容量が必要になります。

  pd_prf_file_count オペランドの指定値×pd_prf_file_size オペランドの指定値
- このオペランドは，pdinfoget コマンドや pdgeter コマンドなどで取得する，障害情報ファイルの容量に影響があります。
- このオペランドの指定値を大きくすると，PRF トレースの編集出力をする pdprfed コマンドの処理時間が長くなります。
- このオペランドの指定値が大き過ぎると，性能問題の調査などでサポートサービスに資料を送付する場合に，ファイルの圧縮を行っても送付が困難になるおそれがあります。資料の送付を考慮したファイルサイズを指定してください。
- PRF トレース情報は，10 秒ごとに PRF トレース情報ファイルへ出力されますが，1 回当たりの出力件数が多い場合，PRF トレース情報ファイルが最大 10 メガバイト小さくなることがあります。

《ほかのオペランドとの関連》

このオペランドは pd_prf_file_count オペランドと関連があります。

◆ pd_prf_output_directory = PRF トレース情報ファイルを出力するディレクトリ～<200 文字以下の絶対パス名>

PRF トレース情報ファイルを出力するディレクトリを指定します。このオペランドに指定するディレクトリは，ユニットごとに異なるディレクトリ名にしてください。指定したディレクトリには，PRF トレース情報ファイル（pdprf_世代番号），及び PRF トレース制御ファイル（.pdprf.conf）を出力します。PRF トレース情報ファイルは pdprf_001 から出力されます。

《前提条件》

このオペランドに指定するディレクトリについては次に示す前提条件を満たしてください。

- PRF トレース出力専用のディレクトリとし，ほかのファイルやディレクトリを配置しないでください。
- ネットワークドライブを指定しないでください。
- 系切り替え構成の場合，現用系及び待機系がそれぞれ起動開始するときにアクセスできるディスク装置内のディレクトリを指定してください。

《注意事項》

- 次の空きディスク容量があるディレクトリを指定してください。

  pd_prf_file_count オペランドの指定値×pd_prf_file_size オペランドの指定値＋ 1KB
- HiRDB を起動する前にあらかじめディレクトリを作成してください。なお，指定したパスの最下位のディレクトリだけがない場合は，HiRDB 開始時に指定したディレクトリを作成し処理を続行します。
- このオペランドに指定するディレクトリの管理について，次に示す項目に注意してください。

  （a）障害発生時には，pdinfoget コマンドで障害情報を取得してください。個別に障害情報を取得する場合は，このオペランドに指定したディレクトリ下を取得してください。

  （b）このオペランドの指定を変更したときや HiRDB システムの OS からの削除（pdsetup -d）をしたときに，以前の PRF トレース情報ファイルが不要な場合は，ユーザが OS のコマンドで削除してください。

  （c）（b）で削除するとき以外では，このオペランドに指定したディレクトリに出力される，PRF トレース制御ファイルを削除しないでください。

《オペランドの省略値》

このオペランドを省略すると，出力先ディレクトリに$PDDIR/spool/pdprf を仮定します。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_prf_file_size
- pd_prf_file_count

9.2.34　トラブルシュート情報に関するオペランド

◆ **pd_cancel_dump = put ｜ noput**

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

次に示す場合にトラブルシュート情報を取得するかどうかを指定します。

- クライアント環境定義の PDCWAITTIME オペランドで指定した監視時間内に SQL が終了しなかった場合
- pdcancel コマンドで実行中の UAP を中断させた場合

取得されるトラブルシュート情報については，マニュアル「HiRDB システム運用ガイド」，及び pd_dump_suppress_watch_time オペランドの「異常終了時に出力される障害情報」を参照してください。

put：

トラブルシュート情報を取得します。トラブルシュート情報が$PDDIR/spool 下のファイルに出力されるため，ファイルシステムが圧迫されるおそれがあります。

なお，出力されたトラブルシュート情報は次に示すタイミングで HiRDB が自動的に削除します。

- HiRDB が稼働中の場合は 24 時間ごと（削除間隔を pd_spool_cleanup_interval オペランドで変更できます）
- HiRDB を開始したとき（削除するかどうかを pd_spool_cleanup オペランドで変更できます）

HiRDB 管理者がトラブルシュート情報を削除する場合は，pdcspool コマンドを実行してください。

noput：

トラブルシュート情報を取得しません。トラブルシュート情報が出力されないため，ファイルシステムへの負荷を軽減できます。通常の運用で UAP の中断が頻繁に発生し，かつ，その原因を調査する必要がない場合に指定してください。

ただし，このオペランドに noput を指定しても，トラブルシュート情報を取得する場合があります。詳細は，pd_dump_suppress_watch_time オペランドの「異常終了時に出力される障害情報」を参照してください。

《注意事項》

pdcancel コマンドに-d オプションを指定して中断させた場合は，このオペランドの指定に関係なくトラブルシュート情報は取得されます。

◆ **pd_client_waittime_over_abort = Y ｜ N**

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

トランザクション実行中にクライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）を超えた場合，次に示すトラブルシュート情報を取得するかどうかを指定します。

- 共用メモリダンプ※
- 簡易ダンプ
- ユニット内の詳細情報

注※　クライアント環境定義 PDHOST，PDNAMEPORT に指定した接続先ユニットの稼働中に，pd_clt_waittime_over_dump_level オペランドに指定したユニットで，1 回だけ取得します。

トラブルシュート情報は$PDDIR/spool/save 下に取得されます。

Y：

トラブルシュート情報を取得します。HiRDB 開始後の最初のクライアントの最大待ち時間オーバが起きた場合だけ，共用メモリダンプを取得します。HiRDB/パラレルサーバの場合，接続したフロントエンドサーバプロセスと関連するバックエンドサーバプロセス，ディクショナリサーバプロセスのトラブルシュート情報を取得します。HiRDB/シングルサーバの場合，シングルサーバプロセスのトラブルシュート情報を取得します。

N：

トラブルシュート情報を取得しません。HiRDB/パラレルサーバの場合，接続したフロントエンドサーバプロセスのトラブルシュート情報を取得しません。関連するバックエンドサーバプロセス，ディクショナリサーバプロセスはロールバックし，トラブルシュート情報は取得しません。HiRDB/シングルサーバの場合，シングルサーバプロセスのトラブルシュート情報を取得しません。

《注意事項》

- トラブルシュート情報を放置しておくと，ファイル容量を圧迫する原因になります。出力されたトラブルシュート情報は次に示すタイミングで HiRDB が自動的に削除します。

  ・HiRDB が稼働中の場合は 24 時間ごと（削除間隔を pd_spool_cleanup_interval オペランドで変更できます）

  ・HiRDB を開始したとき（削除するかどうかを pd_spool_cleanup オペランドで変更できます）

  HiRDB 管理者がトラブルシュート情報を削除する場合は，pdcspool コマンドを実行してください。
- クライアントの最大待ち時間を超えた場合でも，クライアントが接続しているシングルサーバプロセス又はフロントエンドサーバプロセス がクリティカル状態のときは，トラブルシュート情報が取得できないことがあります。プロセスがクリティカル状態かどうかは pdls -d rpc コマンドで確認できます。

《ほかのオペランドとの関連》

HiRDB/パラレルサーバの場合，pd_clt_waittime_over_dump_level オペランドに指定したユニットで共用メモリダンプを取得します。

◆ pd_clt_waittime_over_dump_level = all ｜ shm_fesonly

このオペランドは，HiRDB/パラレルサーバ限定のオペランドです。

pd_client_waittime_over_abort オペランドに Y（省略値）を指定している場合に，共用メモリダンプを取得するユニットの範囲を指定します。

all：

すべてのユニットで，共用メモリダンプを取得します。

shm_fesonly：

クライアントが接続した FES ユニットで，共用メモリダンプを取得します。

このオペランドの指定値によるトラブルシュート情報の出力有無を次に示します。

トラブルシュート情報の種類		クライアントが接続した FES ユニット	トランザクションがアクセスしたユニット	左記以外のユニット
共用メモリダンプ	このオペランドに all を指定した場合	○	○	○
	このオペランドに shm_fesonly を指定した場合	○	×	×
簡易ダンプ		○	○	×
ユニット内の詳細情報		○	○	○

（凡例）

○：トラブルシュート情報を取得します。

×：トラブルシュート情報を取得しません。

注　簡易ダンプ及びユニット内の詳細情報は，このオペランドの指定に関係なく取得されます。

《適用基準》

通常，このオペランドを指定する必要はありません。

障害調査などで保守員にこのオペランドの指定を依頼された場合は，保守員の指示に従って指定してください。

◆ **pd_dump_suppress_watch_time =トラブルシュート情報の出力抑止時間〜<符号なし整数>((0〜3600))《0》（単位：秒）**

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

次に示す現象が発生した場合に出力されるトラブルシュート情報（$PDDIR/spool の下のファイル）の再出力を抑止する時間を秒単位で指定します。

- PDCWAITTIME タイムオーバが発生した場合
- pdcancel コマンドで実行中の UAP を中断させた場合（-d オプション指定時を除く）
- プロセスが異常終了した場合

トラブルシュート情報を出力してから，このオペランドで指定した時間を経過するまでトラブルシュート情報を出力しません。例えば，このオペランドに 60 を指定した場合，トラブルシュート情報を出力した後，60 秒間はトラブルシュート情報を出力しません。

なお，このオペランドに 0 を指定した場合は，トラブルシュート情報の出力を抑止しません。

《利点》

HiRDB サーバのプロセス数が複数あると，タイムアウトなどでサーバプロセスの異常終了が連続して発生することがあります。サーバプロセスの異常終了が連続して発生すると，core 及び簡易ダンプなどのトラブルシュート情報を繰り返し出力するため，HiRDB 運用ディレクトリがあるディスクの容量を圧迫します。HiRDB 運用ディレクトリがあるディスクの容量が不足すると HiRDB が異常終了することがあるため，このオペランドを指定してディスク容量の圧迫を防いでください。

《注意事項》

- pdcancel コマンドで-d オプションを指定した場合や，内部矛盾による異常終了，外部からのシグナル受信処理などは，このオペランドの指定値に関係なくトラブルシュート情報を出力します。
- 異常終了時に出力される障害情報を次の表に示します。

表 9–10　異常終了時に出力される障害情報

異常終了の要因		障害情報	pd_dump_suppress_watch_time の値			
			0		0 以外	
			pd_cancel_dump の値			
			put	noput	put	noput
PDSWAITTIME オーバ		退避コアファイル	×	×	×	×
		障害時のスナップ	×	×	×	×
		簡易ダンプファイル	×	×	×	×
		KFPA20009-W メッセージ	×	×	×	×
		SQL 実行時間警告情報ファイル	×	×	×	×
PDSWATCHTIME オーバ		退避コアファイル	×	×	×	×
		障害時のスナップ	×	×	×	×
		簡易ダンプファイル	×	×	×	×
		KFPA20009-W メッセージ	×	×	×	×
		SQL 実行時間警告情報ファイル	×	×	×	×
PDCWAITTIME オーバ	pd_client_waittime_over_abort=Y	退避コアファイル	○	○	●	●
		障害時のスナップ	○	○	●	●
		障害時のスナップ 2	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●
		SQL 実行時間警告情報ファイル	○	○	●	●
		共用メモリダンプファイル	▲	▲	▲	▲
	pd_client_waittime_over_abort=N	退避コアファイル	×	×	×	×
		障害時のスナップ	○	×	●	×
		障害時のスナップ 2	×	×	×	×
		簡易ダンプファイル	○	×	●	×

異常終了の要因		障害情報	pd_dump_suppress_watch_time の値			
			0		0 以外	
			pd_cancel_dump の値			
			put	noput	put	noput
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
		共用メモリダンプファイル	×	×	×	×
pdcancel コマンド（-d オプション指定あり）		退避コアファイル	○	○	○	○
		障害時のスナップ	○	○	○	○
		簡易ダンプファイル	○	○	○	○
		KFPA20009-W メッセージ	○	○	○	○
		SQL 実行時間警告情報ファイル	○	○	○	○
pdcancel コマンド（-d オプション指定なし）		退避コアファイル	×	×	×	×
		障害時のスナップ	○	×	●	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
内部 kill9※1		退避コアファイル	×	×	×	×
		障害時のスナップ	○	×	●	×
		簡易ダンプファイル	○	×	●	×
		KFPA20009-W メッセージ	○	×	●	×
		SQL 実行時間警告情報ファイル	○	×	●	×
内部 kill3※2		退避コアファイル	○	○	●	●
		障害時のスナップ	○	○	●	●
		簡易ダンプファイル	○	○	●	●
		KFPA20009-W メッセージ	○	○	●	●
		SQL 実行時間警告情報ファイル	○	○	●	●
Abort※3		退避コアファイル	○	○	●	●
		障害時のスナップ	○	○	●	●

異常終了の要因	障害情報	pd_dump_suppress_watch_time の値			
		0		0 以外	
		pd_cancel_dump の値			
		put	noput	put	noput
	簡易ダンプファイル	○	○	●	●
	KFPA20009-W メッセージ	○	○	●	●
	SQL 実行時間警告情報ファイル	○	○	●	●
	アボート情報ファイル	○	○	●	●
上記以外※4	退避コアファイル	△	△	△	△
	障害時のスナップ	○	○	○	○
	簡易ダンプファイル	△	△	△	△
	KFPA20009-W メッセージ	△	△	△	△
	SQL 実行時間警告情報ファイル	△	△	△	△

（凡例）

○：障害情報を出力します。pd_dump_suppress_watch_time オペランドの指定は無効になります。

×：障害情報を出力しません。

●：障害情報を出力します。pd_dump_suppress_watch_time オペランドの指定が有効になります。

△：プロセスの終了形態によっては障害情報が出力されないことがあります。

▲：最大待ち時間を超えたクライアントの，クライアント環境定義 PDHOST，PDNAMEPORT に指定した接続先ユニットの稼働中に，1 回だけ取得します。HiRDB/パラレルサーバの場合，pd_clt_waittime_over_dump_level オペランドに指定したユニットで共用メモリダンプを取得します。

注※1

OpenTP1 の UAP 異常終了処理など，内部的に SIGKILL を発行したときのことです。PDCWAITTIME オーバ，pdcancel コマンドによる異常終了を含みません。

注※2

異常検知時など，内部的に SIGQUIT を発行したときのことです。PDCWAITTIME オーバ，pdcancel コマンドによる異常終了を含みません。

注※3

HiRDB が矛盾を検知して abort()したときのことです。

注※4

SIGSEGV，SIGBUS，外部からのシグナル受信，exit，そのほかの予期しない障害などのことです。

◆ **pd_debug_info_netstat = Y | N**

このオペランドは，トラブルシュート情報の出力量を削減するためのオペランドです。

HiRDB プロセスの異常終了，又は HiRDB（ユニット）が異常終了したときに取得するトラブルシュート情報中に，ネットワーク情報を取得するかどうかを指定します。

Y：取得します。

N：取得しません。

ネットワーク情報に関するトラブルシュート情報は，$PDDIR/spool/save 下の**サーバ名 n .deb** ファイルに出力されます（サーバ名が不明な場合，サーバ名が数字となる場合があります）。n は 1〜3 の通番です。ファイル名の例を次に示します。

（例）

sds1.deb

fes2.deb

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- N を指定すると，HiRDB プロセス又は HiRDB（ユニット）の異常終了時のネットワーク情報取得によるネットワーク負荷を軽減できますが，異常終了の原因が判明できないことがあります。

◆ pd_spool_cleanup_interval = トラブルシュート情報の削除処理間隔

〜＜符号なし整数＞((0〜744))《24》（単位：時間）

このオペランドは，出力したトラブルシュート情報及び作業用一時ファイルを削除するためのオペランドです。トラブルシュート情報及び作業用一時ファイルを残しておくと HiRDB 運用ディレクトリがあるディスクの容量を圧迫する原因になります。HiRDB 運用ディレクトリがあるディスクの容量が不足すると HiRDB が異常終了することがあるため，HiRDB は次に示すファイルを定期的に削除します。

- トラブルシュート情報ファイル（$PDDIR/spool 下のファイル）
- 作業用一時ファイル（$PDDIR/tmp 下のファイル）
- pd_tmp_directory オペランドに指定したディレクトリ下のファイル

このオペランドでは，その削除間隔を時間単位で指定します。例えば，このオペランドに 48 を指定すると，48 時間ごとにこれらのファイルを削除します。通常は（このオペランドを省略すると），24 時間ごとにファイルを削除します。

なお，HiRDB の正常開始時点が時間のカウント開始時点になります。HiRDB を正常終了すると，時間のカウントも止まります。そして，次の正常開始時に時間のカウントが 0 に戻ります。

削除対象のファイルは，次で説明している pd_spool_cleanup_interval_level オペランドで指定します。

《オペランドの規則》

0 を指定すると，ファイルの削除処理を実行しません。

《指定値の目安》

このオペランドに 24，48，72，...を指定すると，決まった時間にファイルの削除処理が実行されます。削除処理はシステムに負荷が掛かる時間帯を外して実施するようにしてください。

《注意事項》

HiRDB が計画停止，強制終了，及び異常終了して HiRDB が停止している間は，時間がカウントされます。ただし，HiRDB の停止中に削除時間になった場合は，ファイルを削除しません。次の削除時間になるまでファイルを削除しません。ファイルを削除してから HiRDB を再開始する場合は，pdcspool コマンドを実行してください。

《備考》

pd_spool_cleanup_interval オペランドと pd_spool_cleanup オペランドの違いについて説明します。

- pd_spool_cleanup_interval オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup オペランドの指定を検討してください。

作業用一時ファイル（$PDDIR/tmp 下のファイル）については HiRDB 開始時は pd_spool_cleanup オペランドの指定に関係なくすべてのファイルを削除します。HiRDB の稼働中は pd_spool_cleanup_interval オペランドの指定に従って定期削除します。

◆ pd_spool_cleanup_interval_level = 日数〔，削除種別〕

このオペランドは，出力したトラブルシュート情報及び作業用一時ファイルを削除するためのオペランドです。トラブルシュート情報ファイル及び作業用一時ファイルの定期削除の削除条件を指定します。

日数：～＜符号なし整数＞((1～24855))《7》（単位：日）

ここで指定した日よりも前に作成したトラブルシュート情報ファイルを削除します。例えば，3 と指定した場合，3 日以内（3 日×24 時間＝ 72 時間以内）に作成したトラブルシュート情報ファイルだけを残し，残りをすべて削除します。

削除種別：＜文字列＞《all》

削除するトラブルシュート情報ファイルの種類を指定します。

all：すべてのファイルを削除対象にします。

dump：HiRDB が内部取得するファイルだけを削除対象にします。

削除対象のファイルを次に示します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
デッドロック・タイムアウト情報，排他資源管理テーブル情報，及びデッドロック SQL 情報	pdlckinf	○	×	排他制御でエラーが発生したとき，及び排他資源管理テーブル不足が発生したときに出力します。
アクセスパス情報	pdsqldump	○	×	アクセスパス表示ユティリティ使用時に出力します。
退避コアファイルなど	save	○	○	プロセスが異常終了したときに出力します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
共用メモリダンプファイル	pdshmdump	○	○	プロセス又はユニットが異常終了したときに出力します。
簡易ダンプファイル	pdsysdump	○	○	なし。
	pdsdsdump	○	○	HiRDB/パラレルサーバには存在しません。
	pdfesdump pddicdump pdbesdump	○	○	HiRDB/シングルサーバには存在しません。
システムログファイルの状態情報ファイル	pdjnlinf	○	×	/pdjnlinf/errinf 下のファイルは削除しません。
トランザクション情報ファイル	pdtrninf	○	×	リアルタイム SAN レプリケーション使用時に出力します。
作業用一時ファイル	tmp	○	○	なし。

（凡例）

○：削除対象のファイルです。

×：削除対象外のファイルです。

注　ディレクトリ名は$PDDIR/spool 下のディレクトリ名です。

作業用一時ファイルについては，削除種別の指定に関係なく次に示すファイル以外を削除します。（　）内は$PDDIR/tmp 下のディレクトリ名です。

- HiRDB が起動するプロセスのカレントワーキングディレクトリ（home）
- 共用メモリ情報ファイル（pdommenv）
- pdbufls コマンドの差分情報ファイル（ファイル名称が"CMb"で始まるファイル）

pd_tmp_directory オペランドに指定したディレクトリ下のファイルについては，削除種別の指定に関係なく次に示すファイル以外を削除します。

- home
- pdommenv
- ファイル名称が"CMb"で始まるファイル

《前提条件》

pd_spool_cleanup_interval オペランドに 0 以外を指定している必要があります。

《指定値の目安》

- コマンド（ユティリティを含む）の実行時間より多い日数を指定してください。例えば，バックアップを取得する pdcopy コマンドの実行に 24 時間（1 日）必要な場合は，日数に 2 以上の値を指定してください。コマンドの実行時間より多い日数を指定しないと，コマンドが使用中の作業用一時ファイルも削除されるため，コマンドが正常に動作しないことがあります。

- 指定値が大き過ぎるとディスクを圧迫し，小さ過ぎると調査に必要なトラブルシュート情報ファイルが消失します。

《オペランドの規則》

日数を指定しないで削除種別だけの指定はできません。

《注意事項》

pd_tmp_directory オペランドを指定しないで環境変数 TMPDIR だけを指定している場合，コマンド又はユティリティが使用する作業用一時ファイルの出力先は環境変数 TMPDIR に指定したディレクトリになります。環境変数 TMPDIR に指定したディレクトリに出力された作業用一時ファイルは定期削除の対象外になるため，OS の rm コマンドなどで削除してください。

《備考》

pd_spool_cleanup_interval_level オペランドと pd_spool_cleanup_level オペランドの違いについて説明します。

- pd_spool_cleanup_interval_level オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup_level オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval_level オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup_level オペランドの指定を検討してください。

作業用一時ファイル（$PDDIR/tmp 下のファイル）については HiRDB 開始時は pd_spool_cleanup_level オペランドの指定に関係なくすべてのファイルを削除します。HiRDB の稼働中は pd_spool_cleanup_interval_level オペランドの指定に従って定期削除します。

◆ **pd_spool_cleanup = normal ｜ force ｜ no**

このオペランドは，出力したトラブルシュート情報を削除するためのオペランドです。次に示すファイルを HiRDB 開始時に削除します。

- トラブルシュート情報ファイル（$PDDIR/spool 下のファイル）
- pd_tmp_directory オペランドに指定したディレクトリ下のファイル

削除対象のファイルは，次で説明している pd_spool_cleanup_level オペランドで指定します。

normal：HiRDB の正常開始及び計画停止後の再開始時に削除します。

force：HiRDB の開始モードに関係なく HiRDB の開始時に削除します。

no：削除しません。

《指定値の目安》

トラブルシュート情報ファイルがディスク容量を圧迫する場合に normal 又は force を指定してください。

《注意事項》

HiRDB 管理者以外のユーザが実行したコマンド又はユティリティによって出力されたトラブルシュート情報ファイルは削除されないことがあります。この場合，トラブルシュート情報ファイルの削除権限を持つユーザが OS の rm コマンドなどでファイルを削除してください。

《備考》

pd_spool_cleanup_interval オペランドと pd_spool_cleanup オペランドの違いについて説明します。

- pd_spool_cleanup_interval オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup オペランドの指定を検討してください。

作業用一時ファイル（$PDDIR/tmp 下のファイル）については HiRDB 開始時は pd_spool_cleanup オペランドの指定に関係なくすべてのファイルを削除します。HiRDB の稼働中は pd_spool_cleanup_interval オペランドの指定に従って定期削除します。

◆ pd_spool_cleanup_level = 日数〔，削除種別〕

このオペランドは，出力したトラブルシュート情報を削除するためのオペランドです。HiRDB 開始時のトラブルシュート情報ファイルの削除条件を指定します。

日数：～＜符号なし整数＞((0～24855))《7》（単位：日）

ここで指定した日よりも前に作成したトラブルシュート情報ファイルを削除します。例えば，3 と指定した場合，3 日以内（3 日×24 時間＝ 72 時間以内）に作成したトラブルシュート情報ファイルだけを残し，残りをすべて削除します。

0 を指定した場合はすべてのファイルを削除します。

削除種別：＜文字列＞《all》

削除するトラブルシュート情報ファイルの種類を指定します。

all：すべてのファイルを削除対象にします。

dump：HiRDB が内部取得するファイルだけを削除対象にします。

削除対象のファイルを次に示します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
デッドロック・タイムアウト情報，排他資源管理テーブル情報，及びデッドロック SQL 情報	pdlckinf	○	×	排他制御でエラーが発生したとき，及び排他資源管理テーブル不足が発生したときに出力します。
アクセスパス情報	pdsqldump	○	×	アクセスパス表示ユティリティ使用時に出力します。

トラブルシュート情報ファイルの種類	ディレクトリ名	all	dump	備考
退避コアファイルなど	save	○	○	プロセスが異常終了したときに出力します。
共用メモリダンプファイル	pdshmdump	○	○	プロセス又はユニットが異常終了したときに出力します。
簡易ダンプファイル	pdsysdump	○	○	なし。
	pdsdsdump	○	○	HiRDB/パラレルサーバには存在しません。
	pdfesdump pddicdump pdbesdump	○	○	HiRDB/シングルサーバには存在しません。
システムログファイルの状態情報ファイル	pdjnlinf	○	×	/pdjnlinf/errinf 下のファイルは削除しません。
トランザクション情報ファイル	pdtrninf	○	×	リアルタイム SAN レプリケーション使用時に出力します。
作業用一時ファイル	tmp	○	○	なし。

（凡例）

○：削除対象のファイルです。

×：削除対象外のファイルです。

注　ディレクトリ名は$PDDIR/spool 下のディレクトリ名です。

pd_tmp_directory オペランドに指定したディレクトリ下のファイルについては，削除種別の指定に関係なく次に示すファイル以外を削除します。

- home
- pdommenv
- ファイル名称が"CMb"で始まるファイル

《前提条件》

pd_spool_cleanup オペランドで normal 又は force（省略値）を指定している必要があります。

《オペランドの規則》

日数を指定しないで削除種別だけを指定できません。

《注意事項》

pd_tmp_directory オペランドを指定しないで環境変数 TMPDIR だけを指定している場合，コマンド又はユティリティが使用する作業用一時ファイルの出力先は環境変数 TMPDIR に指定したディレクトリになります。環境変数 TMPDIR に指定したディレクトリに出力された作業用一時ファイルは定期削除の対象外になるため，OS の rm コマンドなどで削除してください。

《備考》

pd_spool_cleanup_interval_level オペランドと pd_spool_cleanup_level オペランドの違いについて説明します。

- pd_spool_cleanup_interval_level オペランドは，トラブルシュート情報の定期削除に関するオペランドです。
- pd_spool_cleanup_level オペランドは，HiRDB 開始時のトラブルシュート情報削除に関するオペランドです。

したがって，HiRDB を 24 時間連続運転する場合は pd_spool_cleanup_interval_level オペランドの指定を検討してください。HiRDB を毎日終了する場合は pd_spool_cleanup_level オペランドの指定を検討してください。

作業用一時ファイル（$PDDIR/tmp 下のファイル）については HiRDB 開始時は pd_spool_cleanup_level オペランドの指定に関係なくすべてのファイルを削除します。HiRDB の稼働中は pd_spool_cleanup_interval_level オペランドの指定に従って定期削除します。

◆ pd_module_trace_max = モジュールトレース格納最大数

～<符号なし整数>((126～16383))《256》

HiRDB のプロセスは，実行した関数やマクロの履歴をプロセス固有メモリ中に記録しています。この履歴をモジュールトレースといい，このオペランドではモジュールトレースのレコード数を指定します。この履歴の内容は core ファイルに取り込まれ，プロセス異常終了時に出力されます。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《注意事項》

各プロセスでは，以下のサイズのプロセス固有メモリが確保されます。

32 ビットモードの場合：64 ＋ 48×pd_module_trace_max オペランドの値（バイト）

64 ビットモードの場合：64 ＋ 64×pd_module_trace_max オペランドの値（バイト）

◆ pd_module_trace_timer_level = 0 ｜ 10 ｜ 20

モジュールトレースに出力する時刻の取得方法を指定します。このオペランドの指定値の意味を次に示します。

指定値	時刻取得方法
0	すべてのモジュールトレース出力箇所で，秒単位で取得します。
10	すべてのモジュールトレース出力箇所のうち，入出力処理前後など性能調査で問題となりやすい箇所だけマイクロ秒単位で取得し，そのほかの箇所では秒単位で取得します。
20	すべてのモジュールトレース出力箇所で，マイクロ秒単位で取得します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《注意事項》

このオペランドに 0 以外の値を指定すると，時刻をマイクロ秒で取得する関数を発行するため，性能が低下することがあります。

◆ **pd_pth_trace_max = 通信トレース格納最大数**

～＜符号なし整数＞((1024～8388608))《1024》

トラブルシュート情報として使用する通信トレースのレコード数を指定します。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。性能調査などで，保守員がこのオペランドの指定を依頼した場合は，保守員の指示に従ってください。

《注意事項》

このオペランドの指定値を大きくすると，HiRDB のプロセスが確保するプロセス固有メモリが増加します。

また，通信トレースのプロセス固有メモリは，オペランドの指定値を 2 のべき乗に切り上げた値を基に計算します。メモリ所要量については，マニュアル「HiRDB システム導入・設計ガイド」の「メモリ所要量の計算式」を参照してください。

《各見積もり式への影響》

pd_pth_trace_max オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

9.2.35 バックエンドサーバ接続保持機能に関するオペランド

バックエンドサーバ接続保持機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_bes_connection_hold = Y | N**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

バックエンドサーバ接続保持機能を使用するかどうかを指定します。

Y：バックエンドサーバ接続保持機能を使用します。フロントエンドサーバ，バックエンドサーバ間の接続を，同期点を超えて保持するため，接続処理のオーバヘッドを軽減できます。ただし，接続保持する数だけ，メモリ，ポート，ソケットも保持されます。

N：バックエンドサーバ接続保持機能を使用しません。初回 SQL 実行時，フロントエンドサーバ，バックエンドサーバ間の接続処理を行い，同期点時に切断します。

《指定値の目安》

指定値の目安については，マニュアル「HiRDB システム運用ガイド」の「バックエンドサーバ接続保持機能（HiRDB/パラレルサーバ限定）」の「適用基準」を参照してください。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDBESCONHOLD オペランドを指定します。PDBESCONHOLD オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《注意事項》

バックエンドサーバ接続保持機能を使用する場合は，次に示す計算式を満たすようにしてください。

各バックエンドサーバプロセス数（pd_max_bes_process オペランドの値）

≧全フロントエンドサーバプロセス数（pd_max_users オペランドの値×フロントエンドサーバ数）

この計算式を満たさない場合，バックエンドサーバプロセス数が不足して SQL エラーになることがあります。また，HiRDB の稼働中にユティリティなどを実行する場合は，バックエンドサーバプロセス数にユティリティの分の余裕値を確保してください。

◆ **pd_bes_conn_hold_trn_interval = バックエンドサーバ接続保持時間**

～＜符号なし整数＞((0～3600))《1》（単位：秒）

バックエンドサーバ接続保持時間を秒単位で指定します。

バックエンドサーバ接続保持機能を使用すると，HiRDB はトランザクションの終了後から次のトランザクションが実行されるまでの時間を監視します。次のトランザクションが実行されるまでの時間が，指定値の範囲内の場合はフロントエンドサーバとバックエンドサーバの接続を保持します。指定値を超えた場合は，トランザクション終了後にフロントエンドサーバとバックエンドサーバの接続を切断します。

なお，このオペランドに 0 を指定した場合は時間を監視しません。SQL DISCONNECT（XA ライブラリを使用時は xa_close），PDCWAITTIME オーバなどでフロントエンドサーバとクライアント間の接続が切断されたときだけ，フロントエンドサーバとバックエンドサーバの接続を切断します。

9.2.36 再編成時期予測機能に関するオペランド

再編成時期予測機能の運用方法については，マニュアル「HiRDB システム運用ガイド」及び「HiRDB コマンドリファレンス」を参照してください。

◆ **pd_rorg_predict = Y ｜ N**

再編成時期予測機能を使用するかどうかを指定します。

Y：再編成時期予測機能を使用します。

N：再編成時期予測機能を使用しません。

《注意事項》

- 再編成時期予測機能を実行するには，解析情報表及び運用履歴表を格納するデータディクショナリ用 RD エリアを作成する必要があります。
- このオペランドに Y を指定した場合，共用メモリを使用します。使用するメモリ容量については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

9.2.37　セキュリティに関するオペランド

(1)　セキュリティ監査機能に関するオペランド

セキュリティ監査機能の運用方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_audit = Y ｜ N**

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時から監査証跡を取得するかどうかを指定します。

Y：HiRDB の開始時から監査証跡を取得します。

N：HiRDB の開始時から監査証跡を取得しません。

このオペランドに N を指定しても，pdaudbegin コマンドを実行すると監査証跡を取得できます。

《前提条件》

次に示すすべての条件を満たす必要があります。満たしていない場合に Y を指定すると，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始できません。

- 監査証跡ファイル用の HiRDB ファイルシステム領域を作成している
- pd_aud_file_name オペランドに監査証跡ファイル用の HiRDB ファイルシステム領域名を指定している

◆ **pd_aud_file_name = 監査証跡ファイル用の HiRDB ファイルシステム領域名**
～＜パス名＞((150 文字以内))

セキュリティ監査機能を使用する場合はこのオペランドを必ず指定してください。**指定しないとセキュリティ監査機能を使用できません。**

監査証跡ファイル用の HiRDB ファイルシステム領域名を絶対パス名で指定します。

HiRDB/パラレルサーバでセキュリティ監査機能を使用する場合，システム全体で監査証跡を取得することを推奨します。システム全体で監査証跡を取得するには，次のどちらかの指定をしてください。

- システム共通定義で pd_aud_file_name オペランドを指定する
- すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する

ただし，同一サーバマシンで複数のユニットを稼働させるシステム構成の場合は，すべてのユニット制御情報定義で pd_aud_file_name オペランドを指定する必要があります。

《注意事項》

- このオペランドを指定した場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に監査証跡ファイル用の HiRDB ファイルシステム領域に対するアクセスエラーが発生すると，HiRDB を開始できません。
- システム共通定義の pd_aud_file_name オペランドの指定によって同一サーバマシン上の複数のユニットが同じ監査証跡ファイルを指定した場合，監査証跡を正しく取得できません。

◆ **pd_aud_max_generation_size = 1 監査証跡ファイルの最大容量**

～＜符号なし整数＞((1～5240))《100》（単位：メガバイト）

1 監査証跡ファイルの最大容量をメガバイト単位で指定します。

《指定値の目安》

- HiRDB が管理用に使用する領域が 20 メガバイト必要なため，次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_aud_max_generation_size の値×pd_aud_max_generation_num の値＜監査証跡ファイル用の HiRDB ファイルシステム領域長（pdfmkfs コマンドの-n オプションの値）−20（メガバイト）

- 監査証跡の 1 レコードの容量よりも小さい値を指定した場合，又は，このオペランドの指定を省略した場合で，かつ監査証跡の 1 レコードのサイズがこのオペランドの省略時仮定値よりも大きいときには，HiRDB ユニットを起動できません。

  HiRDB ユニットを起動するには，次に示す条件式を満たすようにこのオペランドの値を決定してください。

  pd_aud_max_generation_size の指定値≧↑監査証跡のレコードサイズの最大長÷1024↑×1024 ＋ 2048（バイト）

  監査証跡のレコードサイズの最大長は，次の計算式で算出します。

  監査証跡のレコードサイズの最大長＝ 1067 ＋↑pd_aud_sql_source_size の指定値÷4↑×4＋↑pd_aud_sql_data_size の指定値÷4↑×4（バイト）

- 監査証跡を非同期に出力する場合は，マニュアル「HiRDB システム運用ガイド」の「監査証跡ファイルへの出力（非同期出力）」も併せて参照してください。

- このオペランドの指定値を変更する場合は，マニュアル「HiRDB システム運用ガイド」の「監査証跡ファイルの容量変更」を参照してください。

◆ **pd_aud_max_generation_num = 監査証跡ファイルの最大数**

～＜符号なし整数＞((2～200))《50》

監査証跡ファイル用の HiRDB ファイルシステム領域内に作成する監査証跡ファイルの最大数（世代数）を指定します。

《指定値の目安》

- 全監査証跡ファイルに障害が発生したときのことを想定して，最大値である 200 を指定しないことをお勧めします。監査証跡ファイルに障害が発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

- HiRDB が管理用に使用する領域が 20 メガバイト必要なため，次に示す条件式を満たすようにこのオペランドの値を決定してください。

pd_aud_max_generation_size の値×pd_aud_max_generation_num の値＜監査証跡ファイル用の HiRDB ファイルシステム領域長（pdfmkfs コマンドの-n オプションの値）−20（メガバイト）

《注意事項》

HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，このオペランドの値よりも大きい世代番号のファイルがある場合，このオペランドの値は無効になります。この場合，HiRDB ファイルシステム領域内に作成される監査証跡ファイルの最大数には，最大世代番号が仮定されます。

◆ **pd_aud_no_standby_file_opr = down ｜ forcewrite**

スワップ先にできる監査証跡ファイルがない場合，及び非同期出力用バッファのすべての面がフラッシュ待ちになった場合の HiRDB の処理方式を指定します。

down：

スワップ先にできる監査証跡ファイルが残り一つ以下になった場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。down の指定が有効になるスワップの契機は，監査証跡ファイルの容量が一杯になった場合，及び現用ファイルに障害が発生した場合です。なお，このオペランドの指定によって HiRDB が強制終了した場合の対処については，マニュアル「HiRDB システム運用ガイド」の「スワップ先にできる監査証跡ファイルがないため HiRDB が強制終了した場合」を参照してください。

非同期出力用バッファのすべての面がフラッシュ待ちになった場合，HiRDB（HiRDB/パラレルサーバの場合はユニット）を強制終了します。その後の対処については，マニュアル「HiRDB システム運用ガイド」の「非同期出力用バッファのすべての面がフラッシュ待ちになった場合」を参照してください。

forcewrite：

スワップ先にできる監査証跡ファイルがない場合，データロード待ち（閉塞状態のファイルを除く）の監査証跡ファイルを強制的にスワップ先にして監査証跡の出力を続行します。このとき，最終更新日時が一番古いデータロード待ちの監査証跡ファイルをスワップ先にします。

ただし，全ファイルが閉塞状態の場合は監査証跡の出力を中止します。

全ファイルが閉塞状態の場合に pdaudswap コマンドを実行したときは，pdaudswap コマンドは KFPS05752-E（NO_STANDBY_FILE）メッセージを出力して異常終了します。

非同期出力用バッファのすべての面がフラッシュ待ちになった場合，最初にフラッシュ待ちになった非同期出力用バッファを強制的に上書きして再利用することで，監査証跡の出力を続行します。上書きされた非同期出力用バッファ上の監査証跡は破棄されます。

◆ **pd_aud_async_buff_size = 監査証跡の非同期出力時に使用するバッファ長**

～＜符号なし整数＞((0，4096～6553600))《401408》（単位：バイト）

監査証跡を非同期に出力する場合に使用するバッファ長をバイト単位で指定します。0 を指定した場合は監査証跡を同期出力します。監査証跡のレコードサイズの最大長よりも小さい値を指定した場合，又は，このオペランドの指定を省略した場合で，かつ監査証跡のレコードサイズの最大長がこのオペランドの省略時仮定値よりも大きいときには，HiRDB ユニットを起動できません。

監査証跡のレコードサイズの最大長については，pd_aud_max_generation_size オペランドの説明を参照してください。

各出力方式の長所及び短所を次に示します。

pd_aud_async_buff_size の値	監査証跡の出力方式	長所	短所
0	同期出力	監査証跡を確実に監査証跡ファイルに出力できます。	SQL 処理の延長上でファイル入出力が発生するため，性能に与える影響が大きくなります。
4096～6553600	非同期出力	SQL 処理の性能に与える影響を小さくできます。	バッファへの出力後から監査証跡ファイルに出力するまでの間に，HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了すると，監査証跡が失われることがあります。

《指定値の目安》

監査証跡を非同期に出力する場合，このバッファ長は大きめに設定することを推奨します。このバッファは，ユニットに一つしかないリソースであるため，負荷が高いトランザクションが多い環境では，競合の発生によって性能が劣化する可能性があります。

《オペランドの規則》

このオペランドには 4096 の整数倍を指定してください。4096 の整数倍以外を指定した場合は，指定した値を 4096 の整数倍に切り上げてその値をこのオペランドに設定します。例えば，5000 を指定すると 8192 が設定されます。

《注意事項》

- HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，pd_aud_async_buff_size×pd_aud_async_buff_count バイト分のユニットコントローラ用共用メモリが必要です。この計算式の値がユニットコントローラ全体の共用メモリサイズの上限を超えないように設定してください。ユニットコントローラが使用する共用メモリの計算式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- pd_aud_async_buff_size オペランド，及び pd_aud_async_buff_count オペランドに指定した値が小さい場合，非同期出力用バッファのすべての面がフラッシュ待ちになり，トランザクションの実行時間が長くなったり，pd_aud_no_standby_file_opr オペランドの指定によっては HiRDB（HiRDB/パラレルサーバの場合はユニット）が強制終了したりするおそれがあります。

  単位時間当たりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランド，及び pd_aud_async_buff_count オペランドの指定値を決めてください。詳細は，マニュアル「HiRDB システム運用ガイド」の「監査証跡ファイルへの出力（非同期出力）」を参照してください。

《各見積もり式への影響》

pd_aud_async_buff_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ **pd_aud_async_buff_count = 監査証跡の非同期出力時に使用するバッファ面数**
～＜符号なし整数＞((1～6500))

- ユニット内 HiRDB サーバが存在する場合：**《ユニット内 HiRDB サーバ数×10》**
- ユニット内 HiRDB サーバが存在しない場合：**《1》**

監査証跡を非同期に出力する場合に使用するバッファの面数を指定します。

《指定値の目安》

このバッファ面数は，大きめに設定することを推奨します。値が小さいと，監査証跡ファイルへの書き込みによって，バッファへの書き込みに時間が掛かります。その結果，性能が劣化する可能性があります。

《注意事項》

- HiRDB（HiRDB/パラレルサーバの場合はユニット）の開始時に，pd_aud_async_buff_size×pd_aud_async_buff_count バイト分のユニットコントローラ用共用メモリが必要です。この計算式の値がユニットコントローラ全体の共用メモリサイズの上限を超えないように設定してください。ユニットコントローラが使用する共用メモリの計算式については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- pd_aud_async_buff_size オペランド，及び pd_aud_async_buff_count オペランドに指定した値が小さい場合，非同期出力用バッファのすべての面がフラッシュ待ちになり，トランザクションの実行時間が長くなったり，pd_aud_no_standby_file_opr オペランドの指定によっては HiRDB（HiRDB/パラレルサーバの場合はユニット）が強制終了したりするおそれがあります。

  単位時間当たりの監査証跡の出力件数を考慮して，pd_aud_async_buff_size オペランド，及び pd_aud_async_buff_count オペランドの指定値を決めてください。詳細は，マニュアル「HiRDB システム運用ガイド」の「監査証跡ファイルへの出力（非同期出力）」を参照してください。

《各見積もり式への影響》

pd_aud_async_buff_count オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ **pd_aud_async_buff_retry_intvl = 監査証跡の非同期出力時に使用するバッファの確保リトライ間隔**
～＜符号なし整数＞((1～1000))《50》（単位：ミリ秒）

監査証跡を非同期に出力する場合に使用するバッファがすべて使用中のとき，未使用のバッファが確保できるまでバッファを監視する処理のリトライ間隔を指定します。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

セキュリティ監査機能使用時，UAP の実行に時間が掛かる場合にこのオペランドの値を小さくすると，UAP の実行時間が短くなることがあります。

◆ pd_aud_sql_source_size = 監査証跡に出力する SQL 文のサイズ

～<符号なし整数>((0～2000000))《0》（単位：バイト）

セキュリティ監査機能を使用する場合に監査証跡に出力する SQL 文のサイズをバイト単位で指定します。0 を指定した場合は，監査証跡に SQL 文を出力しません。指定した値よりもサイズが大きい SQL 文については，指定した値を超えた部分は監査証跡に出力されません。

《ほかのオペランドとの関連》

このオペランドを指定した場合，オペランド pd_aud_max_generation_size 及び pd_aud_async_buff_size の指定値を見積もり直してください。

《各見積もり式への影響》

pd_aud_sql_source_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「監査証跡ファイルの容量の見積もり」

◆ pd_aud_sql_data_size = 監査証跡に出力する SQL データのサイズ

～<符号なし整数>((0～1000000))《0》（単位：バイト）

セキュリティ監査機能を使用する場合に監査証跡に出力する SQL データのサイズをバイト単位で指定します。0 を指定した場合は，監査証跡に SQL データを出力しません。指定した値よりもサイズが大きい SQL データについては，指定した値を超えた部分は監査証跡に出力されません。

《ほかのオペランドとの関連》

このオペランドを指定した場合，オペランド pd_aud_max_generation_size 及び pd_aud_async_buff_size の指定値を見積もり直してください。

《各見積もり式への影響》

pd_aud_sql_data_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「監査証跡ファイルの容量の見積もり」

◆ pd_aud_file_wrn_pnt = 警告メッセージの出力契機〔，警告メッセージの出力済み状態のリセット契機〕

警告メッセージの出力契機：～<符号なし整数>((0～100))《0 又は 80》（単位：%）

スワップ先にできない監査証跡ファイル数が警告値以上になったとき，警告メッセージを出力します。このオペランドには警告値を，pd_aud_max_generation_num オペランドに指定した監査証跡ファイルの最大数に対する比率で指定します。例えば，pd_aud_max_generation_num オペランドに 100 を指定し，このオペランドに 90 を指定すると，スワップ先にできない監査証跡ファイル数が 90 以上になると警告メッセージ KFPS05123-W が出力されます。

なお，HiRDB/パラレルサーバの場合はユニット単位でチェックします。

また，このオペランドに 0 を指定すると警告メッセージを出力しません。

《ほかのオペランドとの関連》

- pd_watch_resource オペランドに MANUAL を指定して，かつこのオペランドを省略すると，このオペランドの値に 0 が仮定されます。すなわち，警告メッセージが出力されません。

- pd_watch_resource オペランドに AUTO 又は DEFAULT を指定して，かつこのオペランドを省略すると，このオペランドの値に 80 が仮定されます。すなわち，80%以上になると警告メッセージが出力されます。

警告メッセージの出力済み状態のリセット契機：～＜符号なし整数＞((0～99))（単位：%）

警告メッセージ出力済み状態のリセット契機を指定します。警告メッセージ（KFPS05123-W）が出力されると，警告メッセージ出力済み状態になります。警告メッセージ出力済み状態になると，スワップ先にできない監査証跡ファイル数がもう一度警告値以上になっても警告メッセージが出力されません。しかし，スワップ先にできない監査証跡ファイル数がここで指定する警告メッセージの出力済み状態のリセット契機以下になると，警告メッセージ出力済み状態が解除されます。

例えば，pd_aud_file_wrn_pnt=90,70 と指定すると，スワップ先にできない監査証跡ファイル数が監査証跡ファイルの最大数の 90%以上になると警告メッセージが出力されます。この後，スワップ先にできない監査証跡ファイル数が監査証跡ファイルの最大数の 70%以下になるまで警告メッセージは出力されません。一度 70%以下になり，再度 90%以上になったときに警告メッセージが出力されます。

《注意事項》

- この指定を省略すると，「警告メッセージの出力契機−30」の値が仮定されます。その値がマイナスになる場合は，0 が仮定されます。
- 警告メッセージの出力契機より大きい値を指定すると，その指定は無効になり，警告メッセージの出力契機の指定値と同じ値が仮定されます。

◆ **pd_aud_auto_loading = Y ｜ N**

監査証跡表の自動データロード機能を使用するかどうかを指定します。監査証跡表の自動データロード機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

Y：

監査証跡表の自動データロード機能を使用します。この場合，監査証跡ファイルの世代スワップ（障害によるスワップは除く）を契機として，監査証跡表へのデータロード処理が自動実行されます。

N：

監査証跡表の自動データロード機能を使用しません。この場合，監査証跡表へのデータロード処理は自動実行されません。監査人が手動で実行してください。

《指定値の目安》

監査証跡表の自動データロード機能を使用すると監査人の負荷を低減できます。しかし，オンライン業務中にデータロードが始まるため，CPU やディスクへの入出力回数が増加し，システムの負荷が増大します。これらのメリットとデメリットを考慮した上で，監査証跡表の自動データロード機能を使用するかどうか判断してください。

《注意事項》

HiRDB/パラレルサーバでこのオペランドに Y を指定する場合は，システムマネジャがあるユニットに，監査証跡ファイル用の HiRDB ファイルシステム領域を作成し，pd_aud_file_name オペランドを指定する必要があります。指定しないと，監査証跡表の自動データロード機能を使用できません。

《各見積もり式への影響》

pd_aud_auto_loading オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバで起動するプロセス」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバで起動するプロセス」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ **pdaudload**

〔-i インデクス作成方法〕
〔-l ログ取得方式〕
〔-n 〔一括出力用ローカルバッファ面数〕,,
〔ランダムアクセス用ローカルバッファ面数〕〕
〔-y〕
〔-X サーバ間通信の応答監視時間〕

監査証跡表の自動データロード機能を使用するとき，監査証跡表の自動データロード機能で動作させるデータベース作成ユティリティ（pdload）の環境情報を定義します。

《前提条件》

pd_aud_auto_loading オペランドに Y を指定している必要があります。

《指定値の目安》

マニュアル「HiRDB システム運用ガイド」の「監査証跡表の自動データロード機能」にある「適用条件」を参照してオペランドの値を決定してください。

《注意事項》

このオペランドを 2 回以上指定した場合，1 回目に指定した値が適用されて，2 回目以降の指定値は無効となります。

-i インデクス作成方法

～《c》

インデクスの作成方法を指定します。インデクスの作成方法には次の二つがあります。

c：

インデクス一括作成モードを指定します。行データの格納中は，インデクスを作成しないでインデクス作成情報をインデクス情報ファイルに出力します。行データの格納処理が完了した後，引き続きインデクスを作成します。

s：

インデクス更新モードを指定します。行データを格納するたびにインデクスを更新します。

-l ログ取得方式

～《p》

pdload を実行するとき，データベースの更新ログを取得する方法を指定します。

a：

ログ取得モードを指定します。ロールバック及びロールフォワードに必要なデータベースの更新ログを取得します。

p：

更新前ログ取得モードを指定します。ロールバックに必要なデータベースの更新ログは取得しますが，ロールフォワードに必要なデータベースの更新ログは取得しません。

《注意事項》

リアルタイム SAN レプリケーションを適用する環境でログ同期方式を使用する場合は，必ず a を指定してください。p を指定すると，リモートサイトの監査証跡表がある RD エリアが閉塞します。

-n〔一括出力用ローカルバッファ面数〕,,〔ランダムアクセス用ローカルバッファ面数〕

ローカルバッファを使用して表にデータロードする場合に指定します。このオプションを指定すると，ローカルバッファを使用して DB アクセスできるようになり，一括出力することで入出力回数を削減できます。

このオプションを省略した場合は，グローバルバッファを使用して 1 ページ単位で出力します。

一括出力用ローカルバッファ面数を省略してランダムアクセス用ローカルバッファ面数だけを指定する場合は，コンマ（,）を含めた内容を引用符（"）で囲んで指定してください。例えば，一括出力用ローカルバッファ面数を省略してランダムアクセス用ローカルバッファ面数だけに 1000 を指定する場合は，次のように指定してください。

pdaudload -n ",,1000"

ただし，一括出力用ローカルバッファ面数とランダムアクセス用ローカルバッファ面数の両方を同時に省略することはできません。両方を省略して-n オプションを指定した場合，定義エラーとなり KFPS01895-E メッセージが出力されます。

一括出力用ローカルバッファ面数：～＜符号なし整数＞((2～4096))

一括出力用ローカルバッファの面数を指定します。一括出力用ローカルバッファは，データページに対して使用します。

ランダムアクセス用ローカルバッファ面数：～＜符号なし整数＞((4～125000))

ランダムアクセス用ローカルバッファの面数を指定します。ランダムアクセス用ローカルバッファは，インデクスページに対して使用します。

-y

データロード時に未使用ページを使い切った場合に，データを使用中ページの未使用領域に格納したいときに指定します。このオプションを指定した場合，未使用領域にデータを格納する前に KFPH26010-I メッセージを出力します。

このオプションを指定する場合は-l オプションに a を指定してください。-l オプションに p を指定した場合，又は-l オプションの指定を省略した場合，定義エラーとなり KFPS01895-E メッセージが出力されます。

-X サーバ間通信の応答監視時間

～＜符号なし整数＞((1～65535))《300》（単位：秒）

コマンドを実行したサーバで通信などに障害が発生した場合，コマンドが無応答となり，業務が停止するおそれがあります。pdload は，障害を検知するためにコマンドで実施されるディクショナリ操作についての通信の応答時間を監視できます。

-X オプションには，ディクショナリ操作での応答監視時間を秒数で設定します。ディクショナリ操作時の実行時間が-X オプションに設定した時間を超えた場合，pdload はディクショナリアクセスで障害が発生したと判断し，リターンコード 8 で処理を打ち切ります。処理を打ち切った場合，監査証跡表の自動データロード処理も停止します。

◆ **pd_audit_def_buffer_size = セキュリティ監査情報用バッファ長**
～<符号なし整数>((1～2000000))（単位：キロバイト）

セキュリティ監査機能の情報を格納するバッファ（共用メモリ）の大きさをキロバイト単位で指定します。

《指定値の目安》

セキュリティ監査情報用バッファ長は，次に示す計算式から求めてください。

↑0.3 ＋ a ×0.25↑　（単位：キロバイト）

a：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《オペランドの省略値》

このオペランドを省略した場合，次に示すサイズのバッファ長が HiRDB の開始時に取得されます。このメモリが確保できない場合，HiRDB は開始しますが，セキュリティ監査情報用バッファは作成されません。このとき，KFPD00032-W メッセージが出力されます。

↑0.3 ＋ MAX｛(a ＋ 100)，(a×1.2)｝×0.25↑

（単位：キロバイト）

a：セキュリティ監査機能の監査証跡に指定した絞り込み条件のオブジェクト数

《注意事項》

このオペランドで指定した大きさのメモリが確保できない場合，HiRDB は開始できません。

《各見積もり式への影響》

pd_audit_def_buffer_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

(2) セキュリティの強化に関するオペランド

◆ **pd_security_host_group = ホスト名〔，ホスト名〕...**

このオペランドは，セキュリティレベルの高いシステムを構成する場合，HiRDB サーバ構成で使用するホストを明確に定義し，セキュリティに影響する HiRDB の操作（ユティリティや関連プログラムプロダクトが定義したホスト以外からの操作）を制限するものです。

このオペランドには，HiRDB サーバを構成するネットワークで使用している全ホスト名を指定してください。このオペランドを指定することで，セキュリティ上のリスクを軽減できます。ホスト名は，IP アドレス，FQDN 形式でも指定でき，ループバックアドレスも指定できます。

また，オペランドの指定値は，強制終了，異常終了，及び計画停止後でも変更できます。

なお，ホスト名は 256 文字以内で指定してください。

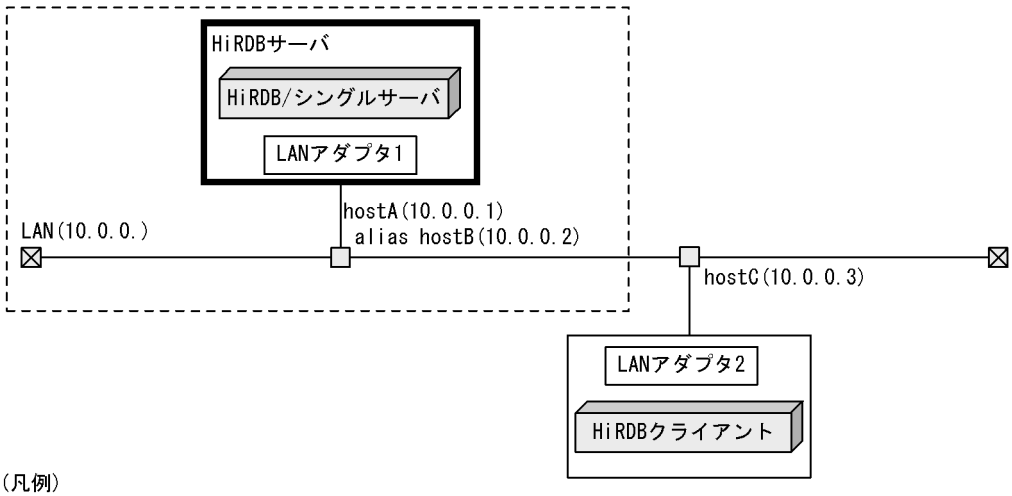
《注意事項》

- DNS サーバを使用する場合は，ここで指定するホスト名を DNS サーバに登録してください。また，DNS サーバを使用しない場合は，ここで指定するホスト名を hosts ファイルに登録してください。
- 名称解決ができない場合，KFPS04693-E メッセージが出力され，HiRDB の開始処理が終了します。
- ホスト名，IP アドレス，又は FQDN の文字列を重複して指定した場合，KFPS04693-E メッセージが出力され，HiRDB の開始処理が終了します。また，IP アドレスが同じでホスト名が異なる場合，エラーにはなりません。
- pdunit オペランドの-x オプションにループバックアドレスを指定した場合は，このオペランドにもループバックアドレスを指定してください。
- JDBC4.0 以降の Type4 JDBC ドライバを使用する場合，DatabaseMetaData インタフェースの getSchemas メソッドについても，セキュリティに影響する HiRDB の操作として扱います。

《備考》

pd_security_host_group オペランドの指定例を次に示します。

pd_security_host_group オペランドの指定例（HiRDB/シングルサーバの場合）

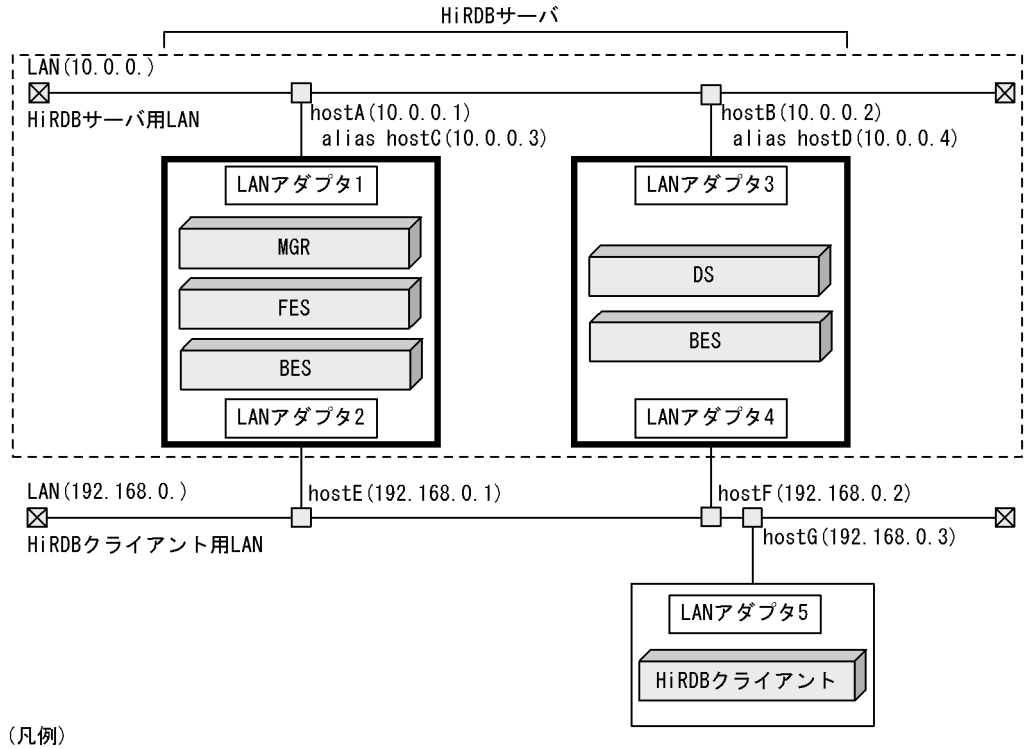


（凡例）
hostA～hostC：ホスト名
xx.x.x.x　　：IPアドレス
：HiRDBサーバを構成するネットワーク

●システム共通定義の内容

```
                    :
set pd_security_host_group=hostA,hostB
pdstart -t SDS -s sds -x hostA
                    :
```

pd_security_host_group オペランドの指定例（HiRDB/パラレルサーバの場合）



（凡例）
hostA～hostG：ホスト名
xxx.xxx.x.x：IPアドレス
：HiRDBサーバを構成するネットワーク

●システム共通定義の内容

```
                    :
set pd_security_host_group=hostA,hostB,hostC,hostD,hostE,hostF
pdstart -t FES -s fes1 -x hostA
                    :
```

◆ pd_security_local_constraint = Y ｜ N

IP アドレスによる接続制限機能を，FES 又は SDS のあるユニットのマシンからの自マシンに対するローカル接続であっても適用するかどうかを指定します。

Y：

IP アドレスによる接続制限機能をローカル接続の場合にも適用します。

N：

IP アドレスによる接続制限機能をローカル接続の場合には適用しません。

《指定値の目安》

FES 又は SDS のあるユニットのマシンからのローカル接続を拒否したい場合，このオペランドに Y を指定してください。

《注意事項》

- このオペランドに Y を指定した場合，誤った接続制限を登録して，すべてのマシンから接続できなくなるおそれがあります。登録を誤って接続できなくなった場合は，いったん HiRDB を停止し，このオペランドを削除した上で再起動し，FES 又は SDS のあるユニットのマシンからローカル接続を行うことで，誤った接続制限を修正してください。

- このオペランドに Y を指定し FES 又は SDS のあるユニットのマシンからローカル接続を拒否する際，pd_security_host_group オペランドを指定している場合は，接続を拒否したい FES 又は SDS のあるユニットのマシンの IP アドレスについて，明示的に接続拒否制限を登録してください。pd_security_host_group オペランドに指定したホストからの接続は，明示的に接続拒否制限が登録されていないかぎり許可されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_security_host_group

9.2.38　系切り替え機能に関するオペランド

◆ **pd_ha = use ｜ nouse**

系切り替え機能を使用するかどうかを指定します。

use：系切り替え機能を使用します。

nouse：系切り替え機能を使用しません。

《注意事項》

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは系切り替え機能を使用できません。このオペランドに use を指定した場合，KFPS04689-W メッセージが出力されます。

◆ **pd_ha_ipaddr_inherit = Y ｜ N**

系切り替え機能を使用する場合に IP アドレスを引き継ぐかどうかを指定します。IP アドレスを引き継ぐかどうかによって，系切り替え機能の運用方法が異なります。詳細については，マニュアル「HiRDB システム運用ガイド」を参照してください。

Y：IP アドレスを引き継ぎます。

N：IP アドレスを引き継ぎません。

1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドを省略してください。指定しても無効になります。

《指定値の目安》

高速系切り替え機能を使用するときの指定値の目安を次に示します。

- HiRDB/シングルサーバの場合

  高速系切り替え機能を使用する場合は IP アドレスを引き継げないため，N を指定してください。

- HiRDB/パラレルサーバの場合

  IP アドレスを引き継ぐ場合は Y を，引き継がない場合は N を指定してください。ただし，高速系切り替え機能の対象にするユニットは IP アドレスを引き継げません。したがって，ここで Y を指定した場合，高速系切り替え機能の対象にするユニットのユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定する必要があります。

なお，このオペランドに N を指定した場合は，ユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに Y を指定できません。

◆ **pd_ha_unit = nouse**

このオペランドはユニットに系切り替え機能を適用しない場合に指定します。系切り替え機能を適用するユニットと，適用しないユニットを分ける場合に指定してください。このオペランドを指定すると，この項で説明しているオペランドの指定がすべて無効になります。

《ほかのオペランドとの関連》

フロントエンドサーバの pdstart オペランドに -k stls オプションを指定すると回復不要 FES ユニットになります。回復不要 FES ユニットでは，系切り替え機能を適用できません。系切り替え機能を適用するシステムの場合，回復不要 FES ユニットのユニット制御情報定義には，必ずこのオペランドを指定してください。

◆ **pd_ha_restart_failure = コマンド名**

～＜パス名＞

このオペランドはモニタモードの場合に指定できます。サーバモードの場合にこのオペランドを指定しても無効になります。

HiRDB の再開始に失敗したときに実行するコマンドを絶対パス名で指定します。このオペランドにクラスタソフトウェアの系を切り替えるコマンドを指定したり，ユーザコマンド（シェル）の中で系を切り替えるコマンドを発行したりすることで，再開始に失敗したときの系切り替えの運用を自動化できます。

◆ **pd_ha_acttype = monitor ｜ server**

系切り替え機能の運用をモニタモードで行うか，サーバモードで行うかを指定します。モニタモード及びサーバモードについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

monitor：系切り替え機能の運用をモニタモードで行います。

server：系切り替え機能の運用をサーバモードで行います。

次に示す機能を使用する場合はサーバモードでの運用が前提になります。

- ユーザサーバホットスタンバイ
- 高速系切り替え機能
- 1：1 スタンバイレス型系切り替え機能
- 影響分散スタンバイレス型系切り替え機能

《前提条件》

このオペランドを指定するには pd_ha オペランドに use を指定する必要があります。また，server を指定する場合は次に示す条件をすべて満たす必要があります。条件を満たしていないと，HiRDB を開始できません。

- クラスタソフトウェアが HA モニタである
- Hitachi HA Toolkit Extension を導入している

《注意事項》

- クラスタソフトウェアが HA モニタの場合は，HA モニタの acttype オペランドの指定とこのオペランドの指定を合わせてください。指定値が異なると，HiRDB を開始できない，系が切り替わらないなどの現象が発生することがあります。acttype オペランドについては，マニュアル「高信頼化システム監視機能 HA モニタ」を参照してください。
- クラスタソフトウェアが PowerHA，LifeKeeper，又は CLUSTERPRO の場合は，このオペランドに server を指定しないでください。

◆ **pd_ha_server_process_standby = Y | N**

ユーザサーバホットスタンバイを使用するかどうかを指定します。ユーザサーバホットスタンバイについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

Y：ユーザサーバホットスタンバイを使用します。

N：ユーザサーバホットスタンバイを使用しません。

なお，1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用する場合はこのオペランドを省略してください。

《前提条件》

Y（省略値）を指定する場合はサーバモードである必要があります。モニタモードの場合は常に N が仮定されます。

《ほかのオペランドとの関連》

高速系切り替え機能はユーザサーバホットスタンバイを包括しています。このため，pd_ha_agent オペランドに standbyunit を指定した場合（高速系切り替え機能を使用する指定をした場合），このオペランドに N を指定しても Y が仮定されます。

◆ **pd_ha_agent = standbyunit | server | activeunits**

次に示す機能を使用する場合にこのオペランドを指定します。

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

これらの機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

standbyunit：高速系切り替え機能を使用する場合に指定します。

server：1：1 スタンバイレス型系切り替え機能を使用する場合に指定します。

activeunits：影響分散スタンバイレス型系切り替え機能を使用する場合に指定します。

なお，同一システム内で，1：1 スタンバイレス型系切り替え機能と影響分散スタンバイレス型系切り替え機能を混在して指定できません。

《前提条件》

このオペランドを指定する場合はサーバモードである必要があります。モニタモードの場合にこのオペランドを指定すると KFPS01896-E エラーになります。

《注意事項》

- 高速系切り替え機能の対象ユニットは IP アドレスを引き継げません。したがって，HiRDB/シングルサーバの場合はシステム共通定義又はユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定してください。HiRDB/パラレルサーバで IP アドレスを引き継ぐ構成の場合は，高速系切り替え機能対象ユニットのユニット制御情報定義の pd_ha_ipaddr_inherit オペランドに N を指定してください。
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合は，グローバルバッファを動的変更できません。グローバルバッファの動的変更については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《ほかのオペランドとの関連》

- 高速系切り替え機能はユーザサーバホットスタンバイを包括しています。このため，このオペランドに standbyunit を指定すると pd_ha_server_process_standby オペランドに Y が仮定されます。
- 1：1 スタンバイレス型系切り替え機能及び影響分散スタンバイレス型系切り替え機能は，系を切り替えた場合，切り替え先のユニットのサーバプロセスを利用します。そのため，このオペランドに server 又は activeunits を指定すると pd_ha_server_process_standby オペランドの指定が無効になります。
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用する場合は，pd_rdarea_open_attribute_use オペランド（RD エリアのオープン契機の指定）に Y が仮定されます。
- 影響分散スタンバイレス型系切り替え機能を使用している場合，HA グループ内の全ユニットが起動していなくてもよいため，pd_start_skip_unit オペランドの指定が無効になります。ただし，HA グループ内の全バックエンドサーバがどこかのユニットで起動していないと，HiRDB を開始できません。
- このオペランドを指定した場合，次に示すオペランドの指定値を確認してください。指定値に誤りがある場合，KFPS01896-E エラーになります。

オペランド名	pd_ha_agent オペランドの値		
	standbyunit	server	activeunits
pd_ha	use	use	use
pd_ha_unit	省略	省略	省略
pd_ha_acttype	server	server	server
pd_ha_ipaddr_inherit（ユニット制御情報定義）	N	－	－
pd_ha_agent	－	ほかのユニットに activeunits を指定してはいけない	ほかのユニットに server を指定してはいけない
pdstart オペランドの-c オプション	－	○	－

オペランド名	pd_ha_agent オペランドの値		
	standbyunit	server	activeunits
pdstart オペランドの-g オプション	−	−	○
pdhagroup	−	−	○

（凡例）

○：オペランドの指定が必要です。

−：該当しません。

《各見積もり式への影響》

pd_ha_agent オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバで起動するプロセス」
- 「メッセージキュー及びセマフォ所要量の見積もり」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/シングルサーバの場合」
- 「リソース数に関連する環境変数の見積もり」の「HiRDB/パラレルサーバの場合」

◆ pd_ha_max_act_guest_servers = 受け入れ可能なゲスト BES の最大数～＜符号なし整数＞((0～33))

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。ユニット内で実行系として稼働するゲスト BES の最大数を指定します。ユニット内で実行系として稼働するゲスト BES 数がこのオペランドの指定値に達すると，ゲスト BES の受け入れ可能状態を解除します。

このオペランドの値によって，受け入れユニットで確保するゲスト用領域の数が決定します。このオペランドの値を変更してゲスト BES 用のリソース所要量を調整してください。

なお，このオペランドの値が小さ過ぎると，系の切り替えが発生したとき，一部のバックエンドサーバの処理が実行されないことがあります。HA グループ内の障害バックエンドサーバ数が HA グループ内の受け入れ可能ゲスト BES 数の合計を超えると，超えた分のバックエンドサーバの処理がどのユニットにも受け入れられないため，そのバックエンドサーバの処理が実行されません。

《前提条件》

このオペランドを指定する場合は次に示すオペランドを指定する必要があります。指定しないと，システム定義の不正によるエラーになります（KFPS01896-E メッセージが出力されます）。

- pdhagroup
- pd_ha_agent = activeunits

《指定値の目安》

このオペランドを省略すると，このオペランドの値を HiRDB が計算します。また，指定値の上限値は次に示す値の小さい方になります。

- 34−ユニット内のホスト BES 数
- HA グループ内のサーバ数−ユニット内のホスト BES 数

上限値以上の値を指定しても，この上限値に補正されます（KFPS05613-W メッセージが出力されます）。

なお，このオペランドを省略したときに HiRDB が算出する受け入れ可能なゲスト BES の最大数は，次に示す条件をすべて満たしていることを想定しています。

- HA グループ内の各ユニットのバックエンドサーバ数が同じである
- 2 点障害が発生した

したがって，HA グループ内の各ユニットのバックエンドサーバ数が異なる場合，又は 3 点障害以上が発生した場合，このオペランドの省略値では一部のバックエンドサーバの処理が実行されないことがあります。

《オペランドの省略値》

このオペランドの省略値の計算式を次に示します。

- HA グループ内のユニット数が 2 の場合：a
- HA グループ内のユニット数が 3 以上の場合：↑（a×2）÷（b−2）↑

ａ：ユニット内のホスト BES 数

ｂ：HA グループ内のユニット数

このオペランドを指定する場合，次に示す計算式（推奨値）からこのオペランドの指定値を求めてください。この計算式は，多点障害時に自ユニットを除くユニットで，ホスト BES 数が多いユニットから順に障害が発生した場合，残りのユニットが障害ユニットのホスト BES を均等に受け入れることを想定しています。

$$推奨値＝ \uparrow \sum_{i=1}^{c} a_i \div （b－c） \uparrow$$

ａ：自ユニット以外で，i 番目にホスト BES 数が多いユニットのホスト BES 数

ｂ：HA グループ内のユニット数

ｃ：想定している多点障害数（＜ｂ）

《各見積もり式への影響》

pd_ha_max_act_guest_servers オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ pd_ha_max_server_process = 受け入れユニット内の最大ユーザサーバプロセス数 ～＜符号なし整数＞((1～15000))《ユニット内ホスト BES の pd_max_bes_process の合計値》

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。ユニット内で起動するユーザサーバプロセスの最大数を指定します。受け入れユニットでは，ホスト BES 及びゲスト BES の各 pd_max_bes_process オペランドの値（最大起動プロセス数）によってサーバプロセス数が制限されます。このオペランドを指定すると，それに加えてユニット内のサーバプロセス数の合計値が制限できます。

《前提条件》

このオペランドを指定する場合は次に示すオペランドを指定する必要があります。指定しないと，このオペランドの指定は無効になります。

- pdhagroup
- pd_ha_agent = activeunits

《指定値の目安》

系の切り替え後，ユニット内のプロセス数を増加させたくない場合はこのオペランドを省略してください。ただし，同時に処理できるサービス要求数の上限が制限されることがあります。リソースが十分にある場合は，ユニット内のホスト BES と，受け入れ可能なゲスト BES の最大起動プロセス数（pd_max_bes_process オペランドの値）の合計値を指定してください。そうすると，系の切り替え後も同時に処理できるサービス要求数の上限が系の切り替え前と同じになります。

このオペランドの指定によって，受け入れユニットに過剰な負荷が掛かることを抑止できます。系切り替え発生後のユニットの負荷上昇，及び同時に処理できるサービス要求数の両者を考慮してこのオペランドの指定値を決定してください。

《注意事項》

ユニット内ホスト BES の pd_max_bes_process オペランドの合計値より小さい値をこのオペランドに指定した場合，その指定値は無効になります。この場合，ユニット内ホスト BES の pd_max_bes_process オペランドの合計値が仮定されます（KFPS05614-W メッセージが出力されます）。

◆ **pd_ha_process_count = ゲスト BES 受け入れ後のユニット内常駐プロセス数**
～＜符号なし整数＞((0～15000))

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。ゲスト BES 受け入れ後のユニット内のホスト BES とゲスト BES の常駐プロセス数の合計値を指定します。

ゲスト BES 受け入れ後の各サーバの常駐プロセス数をこのオペランドの指定で調整できます。

《前提条件》

このオペランドを指定する場合は次に示すオペランドを指定する必要があります。指定しないと，このオペランドの指定は無効になります。

- pdhagroup
- pd_ha_agent = activeunits

《指定値の目安》

- このオペランドを省略すると，このオペランドの値を HiRDB が計算します。
- 非サービス中の常駐プロセスがプロセスを占有していて，一部のバックエンドサーバがプロセスを起動できないため，サービスの要求を処理できない場合，このオペランドの値を小さくしてください。ただし，系の切り替え後，トランザクションのスループットが悪くなることがあります。系の切り替え後，トランザクションのスループットを良くしたい場合は，このオペランドの値を大きくしてください。ただし，このオペランドの値が大き過ぎると，一部のバックエンドサーバでサービスの要求を処理できないことがあります。

《注意事項》

ユニット内ホスト BES の pd_process_count オペランドの合計値より小さい値をこのオペランドに指定した場合，その指定値は無効になります。この場合，ユニット内ホスト BES の pd_process_count オペランドの合計値が仮定されます（KFPS05615-W メッセージが出力されます）。

《備考》

ゲスト BES を受け入れた後，各バックエンドサーバに常駐するプロセス数は，次に示す値のうち小さい方になります。

- このオペランドの値を各バックエンドサーバの pd_process_count オペランドの値で比例配分した値
- 各バックエンドサーバの pd_process_count オペランドの値

$$\mathrm{MIN}\left\{\begin{array}{l}\mathrm{pd\_process\_count}\\ \downarrow \mathrm{pd\_ha\_process\_count} \times \dfrac{\mathrm{pd\_process\_count}}{\sum_{\text{実行中BES}} \mathrm{pd\_process\_count}} \downarrow\end{array}\right\}$$

このオペランドの省略値は，ゲスト BES を受け入れた後の最大起動ユーザサーバプロセス数に，ゲスト BES を受け入れる前の最大起動プロセス数に対する常駐プロセス数の比率を掛けた値です。計算式を次に示します。

$$\uparrow \mathrm{pd\_ha\_max\_server\_process} \times \frac{\sum_{\text{ホストBES}} \mathrm{pd\_process\_count}}{\sum_{\text{ホストBES}} \mathrm{pd\_max\_bes\_process}} \uparrow$$

◆ pd_ha_resource_act_wait_time = リソース活性化の最大待ち時間

～＜符号なし整数＞((2～3600))《10》（単位：秒）

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドを指定します。このオペランドには，ユニットを開始するときに，実行系サーバのリソースが活性化されるまでの最大待ち時間を秒単位で指定します。このオペランドに指定した待ち時間の間は，ユニットの開始処理を待ち合わせます。ただし，ここで指定した時間内にリソースが活性化された場合は，すぐにユニットの開始処理を再開します。

《利点》

ユニットの開始処理が完了したときに，ユニット内の実行系サーバの起動処理が完了していないと，業務を開始できません※。このオペランドに適切な値を指定すると，ユニットの開始処理を待ち合わせるため，ユニットの開始処理が完了した直後から業務を開始できます。

注※　HA モニタの sysdef ファイルの usrcommand オペランドに指定したコマンドの処理に時間が掛かる場合，この現象が発生することがあります。

《指定値の目安》

通常は，このオペランドを指定する必要はありません。次に示す条件をすべて満たす場合にこのオペランドを指定してください。

- 影響分散スタンバイレス型系切り替え機能を使用している場合
- KFPS05623-I メッセージが出力された場合

- メッセージ出力対象ユニットに実行系サーバがある場合

指定値の目安を次に示します。

10 ＋リソース活性化処理に掛かる時間（秒）

リソース活性化処理に掛かる時間：HA モニタの sysdef ファイルの usrcommand オペランドに指定したコマンドの最大処理時間

《備考》

ユニット内に実行系サーバが存在しない場合，このオペランドに指定した時間だけ実行系サーバの開始を待ち合わせます。ただし，ユニット内の全サーバが待機系サーバとして開始した場合は，このオペランドで指定した時間を待たないでユニットの開始処理を再開します。

◆ **pd_ha_transaction = error ｜ queuing**

トランザクションキューイング機能を使用するかどうかを指定します。また，系の切り替え中に HiRDB サーバへの接続数が最大同時接続数（pd_max_users オペランドの値）を超えた場合の処理を指定します。トランザクションキューイング機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

error：

- トランザクションキューイング機能を使用しません。系切り替え中のバックエンドサーバ又はディクショナリサーバで処理するトランザクションをエラーにします。
- 系の切り替え中に HiRDB サーバへの接続数が最大同時接続数を超えた場合，HiRDB サーバへの接続をエラーにします。

queuing：

- トランザクションキューイング機能を使用します。系切り替え中のバックエンドサーバ又はディクショナリサーバで処理するトランザクションをエラーにしないで，系切り替えが完了するまでフロントエンドサーバでトランザクションをキューイングします。このとき，バックエンドサーバ又はディクショナリサーバのユニットの起動が完了するまで待ち合わせを行うため，SQL のレスポンスが通常より遅くなります。
- 系の切り替え中に HiRDB サーバへの接続数が最大同時接続数を超えた場合，HiRDB サーバへの接続処理を pd_ha_trn_queuing_wait_time ＋ pd_ha_trn_restart_retry_time の時間だけ HiRDB クライアントでリトライします。ただし，HiRDB クライアントのバージョンが 07-00 以上の必要があります。

《前提条件》

トランザクションキューイング機能を使用する場合は，次に示す条件をすべて満たす必要があります。

- HiRDB/パラレルサーバである
- 高速系切り替え機能，1：1 スタンバイレス型系切り替え機能，又は影響分散スタンバイレス型系切り替え機能を使用している

なお，HiRDB サーバへの接続処理を HiRDB クライアントでリトライする処理については，前記の前提条件を満たす必要はありません。

《クライアント環境定義との関連》

このオペランドに queuing を指定しても，クライアントごとにトランザクションキューイング機能を使用しないように変更できます。クライアントごとにトランザクションキューイング機能をキャンセルする場合は，クライアント環境定義の PDHATRNQUEUING オペランドを指定してください。PDHATRNQUEUING オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_ha_trn_queuing_wait_time
- pd_ha_trn_restart_retry_time
- pd_ha_trn_queuing_initial

◆ **pd_ha_trn_queuing_wait_time = トランザクションのキューイング待ち時間～<符号なし整数>((1～3600))《180》（単位：秒）**

トランザクションキューイング機能を使用する場合のトランザクションのキューイング待ち時間を指定します。このオペランドに指定した待ち時間を超えても待機系ユニット又はサーバが開始しない場合，キューイング中のトランザクションをエラーにします。これ以降発生するトランザクションはキューイングされないでエラーになります。なお，この待ち時間より先に開始処理中のユニット又はサーバが開始した場合は，この指定時間を待たないでトランザクションが再開されます。

また，1：1 スタンバイレス型系切り替え機能の場合は，正常時は代替部が待機系になり，代替中は正規 BES ユニットが待機系になります。

《前提条件》

pd_ha_transaction オペランドに queuing を指定する必要があります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありませんが，系切り替え時のロールフォワード時間が 180 秒以上掛かる場合にこのオペランドの指定値を大きくしてください。
- 系切り替えに掛かる時間を考慮して，値を指定してください。トランザクションの処理時間が長い UAP を実行している場合は，系切り替え時に行われるロールフォワードの時間も考慮する必要があります。
- このオペランドを指定する場合は，系切り替え時のディスク切り替え時間を考慮して，適正な値を指定してください。指定値が大きいと，系切り替えに失敗した場合，トランザクションがエラーを検知する時間が遅延します。また，指定値が小さいと，系切り替えに成功しても，トランザクションがエラーになることがあります。

《クライアント環境定義との関連》

- PDCWAITTIME

  PDCWAITTIME を指定する場合は，pd_ha_trn_restart_retry_time オペランドの《クライアント環境定義との関連》の PDCWAITTIME を参照してください。

◆ pd_ha_trn_restart_retry_time = トランザクション開始要求エラー時のリトライ時間の上限～<符号なし整数>((1～3600))《60》（単位：秒）

トランザクションキューイング機能使用時に系切り替えが発生すると，トランザクションをフロントエンドサーバでキューイングします。しかし，系切り替えが発生してから待機系ユニット又はサーバが再開始するまでの間，フロントエンドサーバは系切り替えを検知できません。この間（系切り替えが発生してから待機系ユニット又はサーバが再開始するまでの間），フロントエンドサーバは実行系ユニット又はサーバに対してトランザクションの開始要求をしますが，実行系ユニットは既に異常終了しているため，このトランザクション開始要求はエラーになります。エラーになったトランザクションはフロントエンドサーバから開始要求が再度行われます（トランザクションの開始要求をリトライします）。

このオペランドには，このリトライ時間の上限を指定します。このオペランドの値を超えても待機系ユニット又はサーバが再開始されない場合は，リトライ中のトランザクションをエラーにします。また，これ以降発生するトランザクションはリトライされないでエラーになります。なお，このリトライ時間より前に待機系のユニット又はサーバが再開始した場合は，この指定時間リトライは行わないでキューイングされます。

《前提条件》

pd_ha_transaction オペランドに queuing を指定する必要があります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありませんが，クラスタソフトウェアによるディスクの切り替え時間が 60 秒以上掛かる場合にこのオペランドの指定値を大きくしてください。
- このオペランドを指定する場合は，系切り替え時のディスク切り替え時間を考慮して，適正な値を指定してください。指定値が大きいと，系切り替えに失敗した場合，トランザクションがエラーを検知する時間が遅延します。また，指定値が小さいと，系切り替えに成功しても，トランザクションがエラーになることがあります。

《クライアント環境定義との関連》

- PDCWAITTIME

  PDCWAITTIME を指定する場合は，次に示す関係になるように値を設定してください。

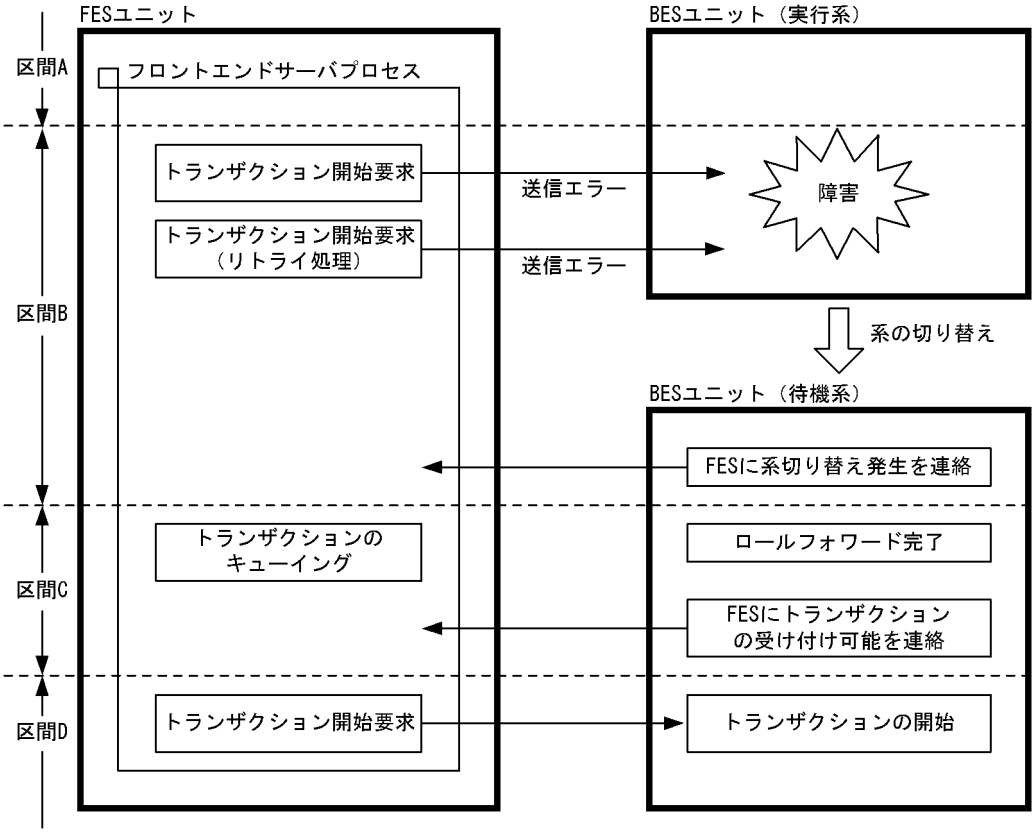
  PDCWAITTIME の指定値＞ pd_ha_trn_queuing_wait_time の指定値＋ pd_ha_trn_restart_retry_time の指定値

  この関係を満たさない場合は，トランザクションキューイング機能によってトランザクションをリトライ，又はキューイングしている間に，PDCWAITTIME に指定した時間を超過してトランザクションがエラーになるときがあります。

  PDCWAITTIME については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《備考》

pd_ha_trn_queuing_wait_time オペランドと pd_ha_trn_restart_retry_time オペランドの関連を次に示します。



〔説明〕

区間 A，D：

トランザクションを開始できる状態です（正常な状態です）。

区間 B：

バックエンドサーバがあるユニットが系切り替え中で，フロントエンドサーバがそれを検知できていない状態です。pd_ha_trn_restart_retry_time オペランドに指定した時間までトランザクション開始要求をリトライします。フロントエンドサーバが系の切り替えを検知すると，トランザクションがキューイングされます。指定した時間を過ぎても検知できない場合は，そのトランザクションをエラーにします。

区間 C：

バックエンドサーバがあるユニットが系切り替え中で，フロントエンドサーバがそれを検知している状態です。pd_ha_trn_queuing_wait_time オペランドに指定した時間までトランザクションをキューイングします。指定した時間を過ぎてもトランザクションを開始できない場合は，そのトランザクションをエラーにします。

◆ pd_ha_trn_queuing_initial = Y ｜ N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

トランザクションキューイング機能を使用する場合に，フロントエンドサーバを配置したユニットの起動直後からトランザクションキューイング機能を有効にするかどうかを指定します。

Y：

フロントエンドサーバを配置したユニットの起動直後からトランザクションキューイング機能を有効にします。

N：

フロントエンドサーバを配置したユニットの起動後に，バックエンドサーバやディクショナリサーバを配置したユニットが稼働中又は起動し始めてからトランザクションキューイング機能を有効にします。

《前提条件》

pd_ha_transaction オペランドに queuing を指定する必要があります。

《指定値の目安》

Y を推奨する系切り替え構成を次の表に示します。

表 9–11　Y を推奨する系切り替え構成

バックエンドサーバやディクショナリサーバを配置したユニットの系切り替え構成			Y を推奨する構成
系切り替え機能を使用しない			–※1
系切り替え機能を使用する	通常の系切り替え	IP アドレスを引き継ぐ	–※1
		IP アドレスを引き継がない	×
	ユーザサーバホットスタンバイ	IP アドレスを引き継ぐ	–※1
		IP アドレスを引き継がない	×
	高速系切り替え機能		○
	1：1 スタンバイレス型系切り替え機能		○
	影響分散スタンバイレス型系切り替え機能		–※2

（凡例）

○：Y を推奨する構成です。

×：Y を推奨しない構成です。

–：トランザクションキューイング機能の適用有無が決まっている構成です。

注※1　トランザクションキューイング機能を使用できない構成です。

注※2　このオペランドの指定値に関係なく，フロントエンドサーバを配置したユニットの起動直後からトランザクションキューイング機能を有効にする構成です。

上記の表で，「Y を推奨する構成」が「×」の系切り替え構成ユニットを含まないシステム構成の場合，指定値を Y にすることを推奨します。

ただし，表の「Y を推奨する構成」が「○」のユニットに次のどちらかの機能を適用する場合は，Y を推奨しません。

- 縮退起動機能（pd_start_level オペランドに 1 を指定し，pd_start_skip_unit オペランドを指定する）

- 拡張ユニット（pd_system_expand_unit オペランドを指定する）

Y を推奨しないシステム構成で Y を適用した際の影響については，《注意事項》を参照してください。

《利点》

Y を適用すると，フロントエンドサーバを配置したユニットの起動直後からトランザクションキューイング機能を有効にします。これによって，フロントエンドサーバを含む複数ユニットの同時障害時にバックエンドサーバやディクショナリサーバを配置したユニットの起動を待たないで，トランザクションキューイング機能が有効になります。

《注意事項》

Y を推奨しないシステム構成で Y を適用した場合，バックエンドサーバやディクショナリサーバを配置したユニットの系切り替え時に，該当ユニットに対しての SQL で，通信障害が発生した旨のエラーになるまでの時間が長くなります（KFPA11981-E メッセージが出力されます）。

また，エラーになるまでの時間が長くなることで，次のどちらかのメッセージを出力することがあります。

- KFPS00993-I（REQUEST=PDCWAITTIME_over）
- KFPS00612-E（reason=pd_ha_trn_restart_retry_time）

◆ **pd_ha_switch_timeout = Y | N**

このオペランドはサーバモードの場合に指定できます。モニタモードの場合にこのオペランドを指定しても無効になります。

系切り替え時の HiRDB（HiRDB/パラレルサーバの場合はユニット）の内部停止処理が**サーバ障害監視時間**を超えた場合に，HiRDB の内部停止処理を待たないで系を切り替えるかどうかを指定します。ここでいうサーバ障害監視時間とは，HA モニタ又は Hitachi HA Toolkit Extension の patrol オペランドに指定した時間のことです。

HA モニタの patrol オペランドについては，マニュアル「高信頼化システム監視機能 HA モニタ」を参照してください。Hitachi HA Toolkit Extension の patrol オペランドについては，マニュアル「Hitachi HA Toolkit」を参照してください。

Y：

系切り替え時の HiRDB の内部停止処理がサーバ障害監視時間を超えた場合，HiRDB の内部停止処理を待たないで系を切り替えます。このとき，HiRDB のスローダウンとして系を切り替えます。

1：1 スタンバイレス型系切り替え機能又は影響分散スタンバイレス型系切り替え機能を使用している場合，計画系切り替え時はこのオペランドの指定が無効になります。

N：

系切り替え時の HiRDB の内部停止処理が終了するまで系を切り替えません。

《利点》

系の切り替え時にディスク障害などで HiRDB の内部停止処理に時間が掛かると，それが系の切り替え時間の遅延要因になります。このオペランドに Y（省略値）を指定すると，HiRDB の内部停止処理に時間が掛かる場合でも，HiRDB の内部停止処理を待たないで系を切り替えられます。

《注意事項》

- patrol オペランドの値が小さいときにこのオペランドに Y を指定すると，計画系切り替えがスローダウンによる系切り替えになることがあります。これは，計画系切り替え時の HiRDB の内部停止処理が patrol オペランドで指定した時間を超えてしまうためです。
- HA モニタ又は Hitachi HA Toolkit Extension の switchtype オペランドに restart を指定した場合は注意が必要です。pd_ha_switch_timeout = Y（省略値）を指定した場合，HiRDB の内部停止処理がサーバ障害監視時間を超えると，障害が発生した系で HiRDB を再開始しません。この場合，すぐに系を切り替えます。

  HA モニタの switchtype オペランドについては，マニュアル「高信頼化システム監視機能 HA モニタ」を参照してください。Hitachi HA Toolkit Extension の switchtype オペランドについては，マニュアル「Hitachi HA Toolkit」を参照してください。

◆ **pd_ha_prc_cleanup_check = Y ｜ N**

サーバモードの系切り替え機能を使用する場合に，切り替え対象のユニット又はユーザサーバのすべてのサーバプロセスが終了するまで，系切り替え処理を待ち合わせるかどうかを指定します。このオペランドに Y を指定すると，サーバ障害による系切り替えのときに，系切り替え前に共有ディスクにアクセスしていたサーバプロセスがすべて終了していることを保証できます。

なお，切り替え対象はスタンバイ型系切り替え機能，及び 1：1 スタンバイレス型系切り替え機能の場合はユニット，影響分散スタンバイレス型系切り替え機能の場合はバックエンドサーバになります。

Y：すべてのプロセスが終了した後に系を切り替えます。

N：切り替え対象のプロセスの終了を待ち合わせません。

このオペランドを指定する場合は，マニュアル「HiRDB システム運用ガイド」の「共有ディスクのアクセス制御」を参照してください。

《前提条件》

次に示す条件をすべて満たしている必要があります。

- pd_ha オペランドに use を指定している
- pd_ha_acttype オペランドに server を指定している
- pd_ha_unit オペランドを指定していない

また，このオペランドに Y を指定する場合は，バージョン 01-08 以降の HA モニタを使用してください。

◆ **pd_ha_mgr_rerun = wait ｜ notwait**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

システムマネジャユニットの系が切り替わるときに，ほかのユニットの開始処理完了を待ち合わせるかどうかを指定します。

wait：

システムマネジャユニットの系を切り替えるときに，ほかのユニットの開始処理完了を待ち合わせます。この場合，システムマネジャユニットを開始するため，以下の確認が行われます。

- 各ユニットのバージョンチェック

- ディクショナリサーバが稼働中であるか
- フロントエンドサーバが一つ以上稼働中であるか
- バックエンドサーバが一つ以上稼働中であるか

システムマネジャユニットの系を切り替える場合に，ほかのユニットが停止しているとき，切り替えに時間が掛かる，又は失敗することがあります。一部のユニットが停止中に，システムマネジャユニットが系切り替えした場合の動作を次に示します。

縮退起動の指定（pd_start_level の値）	開始しないユニット名称の指定	動作
なし（0）	−	開始できない
あり（1）	pd_start_skip_unit オペランドに指定なし	pd_reduced_check_time オペランドで指定した時間だけ開始を待ち合わせた後に系が切り替わる※
	pd_start_skip_unit オペランドに停止中のユニットを指定	すぐに系の切り替えが完了する※

（凡例）

−：該当しません。

注※

次の条件をすべて満たしていないと，系切り替えが完了しません。

- ディクショナリサーバが稼働している
- フロントエンドサーバが一つ以上稼働している
- バックエンドサーバが一つ以上稼働している

notwait：

システムマネジャユニットの系切り替え時に，ほかのユニットの開始処理完了を待ち合わせません。このオペランドを指定すると，システムマネジャユニットの系切り替え時にほかのユニットが停止中でも，そのユニットの開始を待ち合わせることなく，系の切り替えが高速で行われます。

システムマネジャユニットの切り替えが完了すると，HiRDB の動作環境が整っていない場合でも，KFPS05210-I メッセージ（システム初期化完了メッセージ）が出力されます。システムマネジャ切り替え完了後も，UAP がエラーになるなど，業務が続行できない場合は，pdls コマンドでシステムの稼働状況を確認してください。

《前提条件》

このオペランドに notwait を指定する場合は，システム構成がある条件を満たしている必要があります。前提となるシステム構成については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《指定値の目安》

システムマネジャユニットの系切り替えの失敗を防ぐ場合は，notwait を指定してください。システムマネジャユニットの系切り替えに失敗するときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_ha_switch_error = ignore ｜ retry**

このオペランドは AIX 版及び Linux 版限定のオペランドです。それ以外のプラットフォームでは指定できません。

系切り替え失敗時の自動再起動機能を使用するかどうかを指定します。系切り替え失敗時の自動再起動機能については，マニュアル「HiRDB 解説」の「系切り替え失敗時の自動再起動機能」を参照してください。

ignore：

系切り替え失敗時の自動再起動機能を使用しません。

retry：

系切り替え失敗時の自動再起動機能を使用します。

系の切り替え中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了した場合に，系切り替え先の HiRDB を再起動します。

《前提条件》

次に示す条件をすべて満たしている必要があります。

- クラスタソフトウェアに HA モニタを使用
- HA モニタのバージョンが，系切り替え失敗時の自動再開始機能に対応したバージョン以降
  AIX の場合：01-25 以降
  Linux の場合：01-55-01 以降
- HA モニタの server 定義文の switch_error オペランドに retry を指定
- ユーザサーバホットスタンバイ又は高速系切り替え機能を使用

《指定値の目安》

システムの可用性を高めるため，このオペランドに retry の指定を検討してください。

《注意事項》

- 前提条件を満たさないシステム構成でこのオペランドを指定しても，HiRDB は起動時にエラーになりません。必ず前提条件を満たした上で系切り替え失敗時の自動再起動機能を使用してください。
- ユーザサーバホットスタンバイ及び高速系切り替え機能を使用していない場合，このオペランドの指定値に関わらず系切り替え失敗時の自動再起動機能は無効になります。

◆ **pd_ha_observe_timeout = 系切り替え処理の監視時間**

～＜符号なし整数＞((0，30～3600))《0 又は 900》（単位：秒）

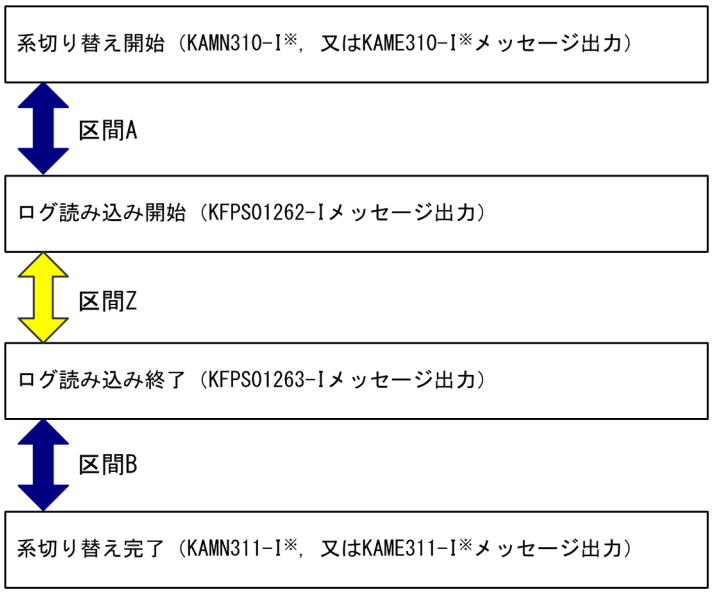
系切り替えの実行時間監視機能の，系切り替え処理の監視時間を指定します。

系切り替えの実行時間監視機能については，マニュアル「HiRDB 解説」の「系切り替えの実行時間監視機能」を参照してください。

このオペランドに 0 を指定した場合は，系切り替えの実行時間監視機能を使用しません。

系切り替えの実行時間監視機能では，次に示す区間 A と B の処理時間の合計が指定値を越えた場合に，HiRDB（HiRDB/パラレルサーバの場合はユニット）をアボートコード Psp5015 で異常終了させます。

図 9–3　系切り替えの実行時間監視機能による監視区間



注※

HA モニタ及び Hitachi HA Toolkit Extension が出力するメッセージです。

〔説明〕

区間 A：

次の処理を行う区間です。監視対象に含まれます。

- システムサーバの起動処理（ユーザサーバホットスタンバイの場合）
- システムファイルの引き継ぎ処理

区間 B：

次の処理を行う区間です。監視対象に含まれます。

- HiRDB 起動完了処理

区間 Z：

次の処理を行う区間です。監視対象に含まれません。

- ロールフォワード処理

《オペランドの規則》

- このオペランドの省略値は，系切り替え失敗時の自動再起動機能を使用（pd_ha_switch_error オペランドに retry を指定）しているかどうかで異なります。

  系切り替え失敗時の自動再起動機能を使用している場合：900（秒）

  系切り替え失敗時の自動再起動機能を使用していない場合：0
- このオペランドに 1～29 を指定した場合は，30 に切り上げます。

《前提条件》

次に示す条件をすべて満たしている必要があります。

- ユーザサーバホットスタンバイ又は高速系切り替え機能を使用している

- マルチスタンバイ構成又は系切り替え失敗時の自動再起動機能を使用している

《指定値の目安》

HiRDB の可用性を高めるため，系切り替え時間の実測値又は想定される系切り替え時間に対し，300 秒以上の余裕値を加えて指定することをお勧めします。

《注意事項》

- 前提条件を満たさないシステム構成でこのオペランドを指定しても，HiRDB は起動時にエラーになりません。必ず前提条件を満たした上で系切り替えの実行時間監視機能を使用してください。
- ユーザサーバホットスタンバイ及び高速系切り替え機能を使用していない場合，このオペランドの指定値に関わらず系切り替えの実行時間監視機能は無効になります。

◆ **pdhagroup -g HA グループ識別子 -u ユニット識別子〔，ユニット識別子〕...**

影響分散スタンバイレス型系切り替え機能を使用する場合にこのオペランドで **HA グループ**を定義します。

一つのシステムに定義できる HA グループ数の上限はありません。

HA グループについては，マニュアル「HiRDB システム運用ガイド」を参照してください。

-g HA グループ識別子

～＜識別子＞((1～8 文字))

HA グループ識別子を指定します。システム内で HA グループを一意に識別できるように指定してください。複数の pdhagroup オペランドに同一の HA グループ識別子を指定すると，エラーになります。

-u ユニット識別子〔，ユニット識別子〕...

～＜識別子＞((4 文字))

HA グループを構成するユニットのユニット識別子を指定します。指定できるユニット数は 2～32 です。ユニットを重複して指定すると，エラーになります。なお，このオプションに指定するユニットは次に示す条件をすべて満たす必要があります。

- HA グループ内の全ユニットが同じネットワークセグメントに属している
- ほかの HA グループに属していない
- 影響分散スタンバイレス型系切り替え機能を適用している（pd_ha_agent = activeunits を指定している）
- ユニット内に一つ以上のホスト BES がある（受け入れ専用ユニットは定義できない）
- ユニット内のホスト BES 数と受け入れ可能なゲスト BES の最大数（pd_max_act_guest_servers オペランドの値）の合計は 34 以内である
- ユニット内のサーバはすべてバックエンドサーバである
- ユニット内のサーバはすべて HA グループに属している

《各見積もり式への影響》

pdhagroup オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

9.2.39 HiRDB Datareplicator に関するオペランド

◆ **pd_rpl_init_start = Y ｜ N**

HiRDB Datareplicator 連携機能を HiRDB 開始時から使用するかどうかを指定します。

Y：HiRDB Datareplicator 連携機能を HiRDB 開始時から使用します。

N：HiRDB Datareplicator 連携機能を HiRDB 開始時から使用しません。

《運用方法》

- このオペランドに N を指定した場合でも，pdrplstart コマンドを入力すれば，その時点から HiRDB Datareplicator 連携を開始できます。
- HiRDB 稼働中に HiRDB Datareplicator 連携を中止する場合，pdrplstop コマンドを入力します。

  なお，pdrplstop コマンドの入力によってデータ連動を中止した場合でも，pdrplstart コマンドを入力すると HiRDB Datareplicator 連携を再開始できます。ただし，pdrplstart コマンドを入力する前に，必ず抽出側データベースを基に反映側データベースを再作成してください。

《注意事項》

- ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは HiRDB Datareplicator 連携機能を使用できません。このオペランドに Y を指定した場合，KFPS04689-W メッセージが出力されます。
- このオペランドに Y を指定する場合，HiRDB を開始（再開始を含む）する前に HiRDB が HiRDB Datareplicator のデータ連動用連絡ファイルにアクセスできるようにしてください。系切り替え機能を使用する場合は，データ連動用連絡ファイルを配置したディスクを活性化した後に HiRDB を開始するようにクラスタソフトウェアを設定してください。データ連動用連絡ファイルを配置したディスクを活性化する前に HiRDB を開始するようにクラスタソフトウェアを設定してしまうと，HiRDB の開始に失敗します。

◆ **pd_rpl_reflect_mode = server ｜ uap**

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB Datareplicator の反映トランザクション同期機能を使用する場合，このオペランドに uap を指定してください。反映トランザクション同期機能を使用しない場合は，このオペランドを指定する必要はありません。反映トランザクション同期機能については，マニュアル「HiRDB データ連動機能 HiRDB Datareplicator」を参照してください。

server：HiRDB サーバ単位でトランザクションを反映します。

uap：UAP 単位でトランザクションを反映します。

《前提条件》

このオペランドを指定する場合は，必ず pd_rpl_hdepath オペランドを指定してください。
pd_rpl_hdepath オペランドを指定していない場合，このオペランドの指定は無視されます。

《注意事項》

uap を指定した場合は更新ログ量が増加します。マニュアル「HiRDB システム導入・設計ガイド」を参照して，システムログファイルの総容量を見積もり直してください。

《ほかのオペランドとの関連》

このオペランドに uap を指定する場合は，pd_log_max_data_size オペランドの指定値を見積もり直してください。

《各見積もり式への影響》

pd_rpl_reflect_mode オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「システムログ量の求め方」

◆ pd_log_rpl_no_standby_file_opr = stop ｜ continue

HiRDB Datareplicator 連携機能を使用しているときに，HiRDB Datareplicator でのシステムログの抽出が完了していないため，すべてのシステムログファイルがスワップ先にできない状態でスワップ要求が発生した場合の運用方法を指定します。

stop：HiRDB のユニットを強制終了します。

continue：HiRDB Datareplicator 連携を中止し，HiRDB だけで運用を続行します。

《前提条件》

この指定値は，pd_rpl_init_start オペランドに Y を指定するか，又は pdrplstart コマンドを入力して，かつ HiRDB Datareplicator 連携機能を使用している場合に有効になります。ただし，HiRDB/パラレルサーバの場合，フロントエンドサーバ，及びディクショナリサーバに対しては無効です。

《注意事項》

- このオペランドに stop（省略時の仮定値）を指定し，ユニットを強制終了した場合，HiRDB Datareplicator 側の抽出中のシステムログファイルが別のファイルに移った（抽出完了のファイルが一つ以上できた）ことを確認してからユニットを再開始してください。強制終了した直後に HiRDB のユニットを再開始しても，スワップ先にできる状態のファイルがないと，HiRDB は異常状態と認識して再度強制終了します。
- このオペランドに continue を指定し，HiRDB Datareplicator 連携を中止した場合，HiRDB Datareplicator 側ではデータ連動の対象となっている抽出側データベースと反映側データベースとの間で不整合が生じるため，反映側データベースの再作成が必要になります。

◆ pd_rpl_hdepath = 抽出側 HiRDB Datareplicator 運用ディレクトリ名

HiRDB Datareplicator 連携機能を使用する場合，抽出側 HiRDB Datareplicator 運用ディレクトリ名を指定します。抽出側 HiRDB Datareplicator の環境変数 HDEPATH で指定した名称をこのオペランドに指定します。

《前提条件》

pd_rpl_init_start オペランドに Y を指定，又は pdrplstart コマンドを入力して HiRDB Datareplicator 連携機能を使用している場合に，このオペランドは有効になります。

《注意事項》

- 抽出側 HiRDB Datareplicator 運用ディレクトリ名を変更する場合は，HiRDB Datareplicator の環境変数 HDEPATH などを変更する必要があります。
- ログ同期方式のリアルタイム SAN レプリケーション機能を使用している場合，ログ適用サイトでは HiRDB Datareplicator 連携機能を使用できません。このオペランドを指定しても，指定がないものとみなされます。

《ほかのオペランドとの関連》

pd_rpl_reflect_mode オペランドを指定する場合は，このオペランドを必ず指定してください。

◆ **pd_rpl_func_control = BACKWARD_CUTOFF_UPDATE ｜ NONE**

～＜識別子＞

HiRDB Datareplicator 連携機能を使用する場合の，抽出側 HiRDB で実行する機能を指定します。

BACKWARD_CUTOFF_UPDATE：

BLOB データ，BINARY データの後方削除更新を行います。BLOB データ，BINARY データの後方削除更新については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

NONE：

BLOB データ，BINARY データの後方削除更新を行いません。

《指定値の目安》

- BACKWARD_CUTOFF_UPDATE の指定可否を次に示します。

ユニット制御情報定義の pd_rpl_hdepath オペランドの指定	抽出側 HiRDB Datareplicator のバージョン	BACKWARD_CUTOFF_UPDATE の指定
あり	08-01 以降	※1
	08-00 以前	不可※2
なし	－	可能

（凡例）

－：該当しません。

注※1　マニュアル「HiRDB データ連動機能 HiRDB Datareplicator」の「BLOB 型と BINARY 型を使った表のデータ連動」を参照してください。

注※2　このオペランドに NONE を指定してください。

上記の条件を満たしていない場合は，BACKWARD_CUTOFF_UPDATE を指定しないでください。指定しても，抽出側 DB での抽出対象表に BLOB データ，BINARY データの後方削除更新の結果は反映側 DB に反映されません。

- pd_rpl_hdepath オペランドを指定していて，かつ，抽出側 Datareplicator のバージョンが 08-00 以前の場合は，このオペランドに NONE を指定してください。

《注意事項》

- 識別子は大文字と小文字のどちらでも指定できます。
- 同じ識別子を 2 回以上指定しても，1 回だけ指定した場合と同じ結果になります。
- BACKWARD_CUTOFF_UPDATE を指定した後に NONE を指定する場合は，変更する前にすべての更新ログが反映し終わっているかどうかを確認してください。反映されていない更新ログがある状態でオペランドの指定値を変更すると，抽出側 DB と反映側 DB で SQL の実行数が一致しないなどの問題が発生します。
- BACKWARD_CUTOFF_UPDATE を指定した状態でストアドプロシジャ，ストアドファンクション，又はトリガを定義した後に NONE を指定する場合は，それぞれ ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER を再作成してください。再作成しないと，ストアドプロシジャ，ストアドファンクション，又はトリガの実行時にエラーとなります。ただし，BLOB データ，BINARY データの後方削除更新を行っていない場合は再作成する必要はありません。ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER については，マニュアル「HiRDB SQL リファレンス」を参照してください。
- NONE を指定した状態でストアドプロシジャ，ストアドファンクション，又はトリガを定義した後に BACKWARD_CUTOFF_UPDATE を指定する場合は，それぞれ ALTER PROCEDURE，ALTER ROUTINE，又は ALTER TRIGGER を再作成してください。再作成しなくてもエラーにはなりませんが，BLOB データ，BINARY データの後方削除更新は行いません。

9.2.40　リアルタイム SAN レプリケーションに関するオペランド

リアルタイム SAN レプリケーションについては，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。

◆ **pd_rise_use = Y ｜ N**

リアルタイム SAN レプリケーションを使用するかどうかを指定します。

Y：リアルタイム SAN レプリケーションを使用します。

N：リアルタイム SAN レプリケーションを使用しません。

このオペランドに Y を指定した場合，HiRDB は日立ディスクアレイシステムの TrueCopy，Universal Replicator，及び RAID Manager を使用してデータベースファイルやシステムファイルの管理及び制御をします。

《ほかのオペランドとの関連》

このオペランドに Y を指定する場合は，必ず pd_rise_pairvolume_combination オペランドを指定してください。pd_rise_pairvolume_combination オペランドを指定しないと，HiRDB を開始できません。

◆ **pd_rise_pairvolume_combination = sync ｜ async ｜ hybrid ｜ syssync**

リアルタイム SAN レプリケーションの処理方式を選択します。

sync：全同期方式

async：全非同期方式

hybrid：ハイブリッド方式

syssync：ログ同期方式

《前提条件》

- pd_rise_use オペランドに Y を指定している必要があります。
- syssync を指定する場合は，HiRDB Disaster Recovery Light Edition が必要です。HiRDB Disaster Recovery Light Edition を導入しないで syssync を指定すると，HiRDB を開始できません。

《注意事項》

インナレプリカ機能使用時に syssync を指定すると，HiRDB を開始できません。

《ほかのオペランドとの関連》

- pd_rise_use オペランドに Y を指定する場合は，必ずこのオペランドを指定してください。このオペランドを指定しないと，HiRDB を開始できません。
- sync，hybrid，又は syssync を指定する場合は，必ず pd_rise_fence_level オペランドを指定してください。pd_rise_fence_level オペランドを指定しないと，HiRDB を開始できません。

◆ **pd_rise_disaster_mode = normal ｜ alone**

リアルタイム SAN レプリケーションの処理方式にハイブリッド方式を選択している場合このオペランドを指定します。このオペランドは，メインサイトとリモートサイト間で同期を取り，データの整合性を維持するかどうか指定します。

normal：

メインサイトとリモートサイト間で同期を取り，データの整合性を維持します。

alone：

メインサイトとリモートサイト間で同期を取らないので，データの整合性を維持しません。災害発生などによって，リモートサイトへデータを更新コピーできない場合に指定します。

《前提条件》

このオペランドを指定する場合は，次に示す条件をすべて満たす必要があります。

- pd_rise_use オペランドに Y を指定します。
- pd_rise_pairvolume_combinaion オペランドに hybrid を指定します。

《指定値の目安》

このオペランドは，災害発生などによってリモートサイトにあるボリュームへデータを更新コピーできない場合に使用するため，通常は，このオペランドを省略するか，又は normal を指定します。

◆ **pd_rise_fence_level = data ｜ never**

リアルタイム SAN レプリケーションの処理方式に，全同期方式，ハイブリッド方式，又はログ同期方式のどれかを選択している場合に，障害が発生して，リモートサイトにあるボリュームへのデータの同期コピー（HiRDB ファイルの全部又は一部を転送すること）に失敗したときの動作を指定します。同期コピーが失敗した場合の動作については，マニュアル「HiRDB ディザスタリカバリシステム　構築・運用ガイド」を参照してください。

data：メインサイトにあるボリュームの更新を中止します。

never：メインサイトにあるボリュームだけ更新をして，続行します。

《ほかのオペランドとの関連》

pd_rise_pairvolume_combinaion オペランドに sync，hybrid，又は syssync を指定する場合は，必ずこのオペランドを指定してください。このオペランドを指定しないと，HiRDB を開始できません。

◆ **HORCMINST RAID Manager のインスタンス番号**

～＜符号なし整数＞

リアルタイム SAN レプリケーションを使用する場合，HiRDB が使用するペア論理ボリュームを定義した RAID Manager のインスタンス番号を設定します。指定値の詳細については，RAID Manager のマニュアルを参照してください。

9.2.41　インナレプリカ機能に関するオペランド

インナレプリカ機能を使用するには，HiRDB Staticizer Option が必要です。インナレプリカ機能については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。

◆ **pd_inner_replica_control = インナレプリカ最大グループ数**

～＜符号なし整数＞((1～4194296))《0》

インナレプリカグループの最大数を指定します。このオペランドには，HiRDB 正常開始時点のインナレプリカグループ数よりも大きい値を指定してください。

HiRDB/パラレルサーバの場合は，各バックエンドサーバが管理するインナレプリカグループのうち，最大グループ数を指定してください。例えば，次に示すようなバックエンドサーバ構成の場合は，このオペランドに 31 以上を指定します。

- バックエンドサーバ 1：インナレプリカグループ数が 10
- バックエンドサーバ 2：インナレプリカグループ数が 20
- バックエンドサーバ 3：インナレプリカグループ数が 30

《注意事項》

- このオペランドの指定値より，実際のインナレプリカグループ数の方が大きい場合は，HiRDB を開始できません。
- このオペランドで指定した値を超えるインナレプリカグループは定義できません。

- このオペランドの指定値を大きくすると，シングルサーバ又はバックエンドサーバが使用する共用メモリが増えるため，不要に大きな値を指定しないでください。共用メモリが不足すると，HiRDB を開始できなくなることがあります。シングルサーバが使用する共用メモリ，及びバックエンドサーバが使用する共用メモリについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- ログ同期方式のリアルタイム SAN レプリケーションとインナレプリカ機能は同時に使用できません。同時に使用しようとした場合は，HiRDB を開始できません。

《ほかのオペランドとの関連》

このオペランドを指定すると，pd_indexlock_mode オペランドに NONE が仮定されます。

《各見積もり式への影響》

pd_inner_replica_control オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 5」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 5」

◆ pd_inner_replica_lock_shift = Y ｜ N

インナレプリカ機能を使用するとき，UAP とコマンドを同時実行させるかどうかを指定します。UAP と同時実行できるコマンドを次に示します。

- pddbchg
- pdorbegin
- pdorchg
- pdorend

Y：

サーバ内で UAP が実行中の場合，UAP が処理対象とする RD エリアのオリジナル RD エリアとコマンドが処理対象とする RD エリアのオリジナル RD エリアが異なっていれば，UAP とコマンドを同時実行できます。オリジナル RD エリアが同じときには同時実行できません。

N：

サーバ内で UAP が実行中の場合，UAP とコマンドは同時実行できません。N を指定すると，Y を指定したときよりも UAP 実行中の排他資源数を少なくできます。

このオペランドの指定値による，コマンドと UAP の同時実行可否を次に示します。

実行順序	pd_inner_replica_lock_shift オペランドの指定値	処理対象 RD エリアのオリジナル RD エリア	
		異なる	同じ
UAP 実行中にコマンド開始	N	×	×

実行順序	pd_inner_replica_lock_shift オペランドの指定値	処理対象 RD エリアのオリジナル RD エリア	
		異なる	同じ
	Y	○	×
コマンド実行中に UAP 開始	−	○	×

（凡例）

○：コマンドと UAP を同時実行できます。

×：コマンドと UAP を同時実行できません。

−：該当しません。

このオペランドの指定値による，排他制御の違いを次に示します。

実行順序	pd_inner_replica_lock_shift オペランドの指定値	排他制御		
		排他資源	排他が掛かる範囲	インナレプリカを使用しない場合と比較した排他資源数
UAP 実行中にコマンド開始	N	インナレプリカ構成管理	シングルサーバ，又はバックエンドサーバ	一つ増加
	Y	レプリカグループ構成管理	インナレプリカグループ	処理対象 RD エリア数分増加
コマンド実行中に UAP 開始	−	レプリカグループ構成管理	インナレプリカグループ	処理対象 RD エリア数分増加

（凡例）

−：該当しません。

◆ pd_lv_mirror_use = Y | N

レプリカ RD エリアのオープン属性を SCHEDULE にするかどうかを指定します。

Y：

レプリカ RD エリアのオープン属性を SCHEDULE にします。データベース構成変更ユティリティ（pdmod）でレプリカ RD エリアのオープン属性を変更していても，オープン属性は SCHEDULE になります。

N：

レプリカ RD エリアのオープン属性をオリジナル RD エリアと同じ属性にします。ただし，データベース構成変更ユティリティ（pdmod）でレプリカ RD エリアのオープン属性を変更していると，変更後のオープン属性になります。

《指定値の目安》

すべてのレプリカ RD エリアのオープン属性を SCHEDULE にする場合に Y を指定してください。

《注意事項》

- レプリカ RD エリアのオープン属性に関する注意事項や，RD エリアのオープン属性の変更方法については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。
- このオペランドに Y を指定すると，シングルサーバ又はバックエンドサーバが使用する共用メモリが増えます。シングルサーバが使用する共用メモリ，及びバックエンドサーバが使用する共用メモリについては，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

◆ **pd_inner_replica_hold_status = NORMAL ｜ HOLD**

DEFER 又は SCHEDULE 属性のレプリカ RD エリアを，HiRDB の正常開始時にコマンド閉塞にするかどうかを指定します。ただし，障害閉塞している RD エリアの状態は変更しません。

NORMAL：コマンド閉塞状態にしません。

HOLD：コマンド閉塞状態にします。

表 9–12　レプリカ RD エリアの状態と，pd_inner_replica_hold_status オペランドの指定値

項番	レプリカ RD エリアの状態		pd_inner_replica_hold_status オペランドの指定値	
	RD エリアのオープン属性	HiRDB 正常終了時の RD エリアの状態	NORMAL	HOLD
1	DEFER	障害閉塞	障害閉塞かつクローズ状態	障害閉塞かつクローズ状態
2		障害閉塞以外	クローズ状態	コマンド閉塞かつクローズ状態
3	SCHEDULE	障害閉塞	障害閉塞かつクローズ状態	障害閉塞かつクローズ状態
4		障害閉塞以外	クローズ状態	コマンド閉塞かつクローズ状態

《前提条件》

このオペランドは，次の条件を満たす場合に有効になります。

- HiRDB Staticizer Option を組み込んでインナレプリカ機能を使用している場合
- レプリカ RD エリアのオープン属性が DEFER 又は SCHEDULE の場合

《利点》

通常，HiRDB を正常開始すると，DEFER 又は SCHEDULE 属性のレプリカ RD エリアはクローズ状態ですが，ディスクは更新できます（ディスクアクセス時に自動的にオープン状態になるため）。このとき，オリジナル RD エリアとレプリカ RD エリアの，ペアボリュームがペア状態のまま，定義系 SQL などでレプリカ RD エリアに更新を行うと，レプリカ RD エリアを更新しようとして入出力エラーで障害閉塞となります。

このオペランドの指定値が HOLD の場合，ユーザが意図的に RD エリアの閉塞解除を行わないとレプリカ RD エリアにアクセスできません。そのため運用ミスによる RD エリアの障害閉塞を防止できます。

《指定値の目安》

HiRDB の正常開始時に，オリジナル RD エリアとレプリカ RD エリアの，ペアボリュームがペア状態となる運用を行う場合は，このオペランドに HOLD を指定することをお勧めします。

《注意事項》

- このオペランドに HOLD を指定した場合，正常開始直後のレプリカ RD エリアはコマンド閉塞かつクローズ状態になります。レプリカ RD エリアにアクセスする準備が整ったら，pdrels コマンドで RD エリアの閉塞解除を行ってください。
- このオペランドに HOLD を指定して，レプリカ RD エリアがコマンド閉塞かつクローズ状態のままで定義系 SQL を実行すると KFPH22032-W メッセージが出力されます。マニュアル「インナレプリカ機能 HiRDB Staticizer Option」の「KFPH22032-W メッセージの出力と対処」を参照して，対処してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_rdarea_open_attribute_use
- pd_rdarea_open_attribute
- pd_lv_mirror_use

◆ **pd_max_reflect_process_count = 追い付き反映処理時に確保するプロセス数～<符号なし整数>((1～256))**

更新可能なオンライン再編成をする場合にこのオペランドを指定します。**このオペランドを省略すると更新可能なオンライン再編成ができません。**

このオペランドには，追い付き反映処理をするときに使用するプロセス数を指定します。HiRDB はこのオペランドの値の数だけプロセスを確保します。なお，HiRDB/パラレルサーバの場合は 1 フロントエンドサーバ当たりのプロセス数になります。

《指定値の目安》

- このオペランドの値は次に示す計算式を満たすようにしてください。この条件を満たさないと HiRDB を開始できません。

  HiRDB/シングルサーバの場合は **pd_max_reflect_process_count の値＋ pd_max_users の値≦3000（pd_max_users の最大値）**

  HiRDB/パラレルサーバの場合は **pd_max_reflect_process_count の値＋ pd_max_users の値≦2000（pd_max_users の最大値）**
- 追い付き反映処理時に確保するプロセス数（pd_max_reflect_process_count オペランドの見積もり式）については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。

《各見積もり式への影響》

pd_max_reflect_process_count オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「64 ビットモードに移行する際の考慮点」
- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB のプロセス一覧」
- 「Linux のオペレーティングシステムパラメタの見積もり」
- 「SQL 文が使用する作業表用ファイルの容量」
- 「ステータスファイルの容量の見積もり」の「S の求め方」
- 「最大ファイル数の見積もり（pdfmkfs -l コマンド）」
- 「シングルサーバが使用する共用メモリの計算式」の「計算式 2」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 2」
- 「メッセージキュー及びセマフォ所要量の見積もり」

◆ **pd_log_org_reflected_logpoint = keep ｜ release**

更新可能なオンライン再編成をする場合にこのオペランドを指定します。

追い付き反映処理が終了したシステムログファイルの状態を変更するかどうかを指定します。

keep：

追い付き反映処理が終了したシステムログファイルの状態を変更しません。ファイル中の全更新ログの追い付き反映処理が終了してもシステムログファイルの状態をオンライン再編成上書き禁止状態のままとします。

release：

追い付き反映処理が終了したシステムログファイルの状態を変更します。ファイル中の全更新ログの追い付き反映処理が終了した場合，システムログファイルの状態をオンライン再編成上書き禁止状態からオンライン再編成上書き可能状態に変更します。

なお，次に示す場合はこのオペランドの値に関係なく，システムログファイルの状態をオンライン再編成上書き禁止状態からオンライン再編成上書き可能状態に変更します。

- 追い付き反映処理がすべて終了した場合
- pdorbegin -u コマンドを実行した場合
- pdorend -u コマンドを実行した場合

《指定値の目安》

このオペランド及び pd_log_org_no_standby_file_opr オペランドの値によって，システムログファイルのオンライン再編成上書き禁止状態の状態遷移と更新可能なオンライン再編成の運用方法が決まります。このオペランドの指定値の目安については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。

《注意事項》

強制終了，異常終了，又は計画停止後にこのオペランドの指定値を変更する場合は，keep（省略値）から release への変更だけができます。

◆ **pd_log_org_no_standby_file_opr = stop ｜ continue**

更新可能なオンライン再編成をする場合にこのオペランドを指定します。

全システムログファイルがオンライン再編成上書き禁止状態の場合に，システムログファイルのスワップが発生したときの HiRDB の処理を指定します。

stop：

HiRDB を強制終了します。HiRDB/パラレルサーバの場合はユニットを強制終了します。

continue：

オンライン再編成上書き禁止状態をオンライン再編成上書き可能状態に変更して処理を続行します。HiRDB/パラレルサーバの場合はサーバ内のシステムログファイルが対象になります。

《指定値の目安》

このオペランド及び pd_log_org_reflected_logpoint オペランドの値によって，システムログファイルのオンライン再編成上書き禁止状態の状態遷移と更新可能なオンライン再編成の運用方法が決まります。このオペランドの指定値の目安については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。

《注意事項》

このオペランドに continue を指定してオンライン再編成上書き禁止状態を強制的に解除した場合，その時点より後に使用したシステムログファイルはオンライン再編成上書き禁止状態になりません。このため，正系 RD エリアにまだ反映していない更新ログを持つシステムログファイルが上書きされると，追い付き反映処理ができないことがあります。この場合，HiRDB はカレントの副系 RD エリアだけを使用して業務を続行します。正系 RD エリアにカレントを戻す場合，又は更新可能なオンライン再編成を再度実行する場合は，事前に副系 RD エリアのバックアップから正系 RD エリアを回復する必要があります。正系 RD エリア回復手順については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。

9.2.42 OLTP に関するオペランド

◆ **pd_oltp_holdcr = use ｜ nouse**

OLTP 環境下の UAP で，ホールダブルカーソル機能を使用するかどうかを指定します。

use：OLTP 環境下の UAP でホールダブルカーソル機能を使用します。

nouse：OLTP 環境下の UAP でホールダブルカーソル機能を使用しません。

《指定値の目安》

OLTP 環境下の UAP でホールダブルカーソル機能を使用する場合だけ，このオペランドに use を指定してください。

《注意事項》

OLTP 環境下の UAP でホールダブルカーソル機能を使用する場合，特定の条件を満たす UAP を使用しなければなりません。ホールダブルカーソルについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

9.2.43　日付・時刻に関するオペランド

◆ **TZ タイムゾーン**

～＜文字列＞《JST−9》

出力するログの日付・時刻情報のタイムゾーンを指定します。タイムゾーンとは，時刻などを表示する場合の環境を設定する環境変数のことをいいます。

このオペランドには，OS のタイムゾーンと同じ値を指定してください。

9.2.44　ユーザ認証に関するオペランド

◆ **pd_os_authenticate = Y ｜ N**

OS ログインユーザの簡易認証機能を有効にするかどうかを指定します。

OS ログインユーザの簡易認証機能についてはマニュアル「HiRDB システム運用ガイド」の「OS ログインユーザの簡易認証機能とは」を参照してください。

Y：

OS ログインユーザの簡易認証機能を有効にします。

N：

OS ログインユーザの簡易認証機能を無効にします。

《指定値の目安》

通常は省略又は N を指定してください。

◆ **pd_connect_auth_type = ALL ｜方式〔，方式〕...**

ALL：

すべての認証方式による接続を許可します。

方式：

HiRDB サーバが許可する認証方式を指定します。

方式に指定できる値を次に示します。

- **PA：**
  PA 方式による接続を許可します。
- **CHA：**
  CHA 方式による接続を許可します。

次のオプションを指定することで，CHA 方式で使用するハッシュ関数を選択できます。

/SHA-256：

CHA 方式のハッシュ関数として SHA-256 を使用します。

/SHA-512：

CHA 方式のハッシュ関数として SHA-512 を使用します。

/SHA-256 と/SHA-512 の両方指定した場合，及び両方を省略した場合は，クライアント環境定義 PDAUTHTYPE の指定値との組み合わせによって，使用するハッシュ関数を決定します。詳細については，マニュアル「HiRDB システム運用ガイド」の「HiRDB 接続時のパスワード秘匿化の設定方法」を参照してください。

《指定値の目安》

- セキュリティを強化したい場合は，CHA 方式を指定してください。
- CHA 方式を指定した場合は，CHA 方式に対応した HiRDB クライアントだけ接続を許可します。

9.2.45　メッセージの出力抑止機能に関するオペランド

◆ **pdmlgput -s 出力有無**

{-c ALL ｜〔-l メッセージの重要度〕 -m メッセージ ID〔，メッセージ ID〕...}

HiRDB が syslogfile に出力するメッセージに対して，次に示す指定ができます。

- 全メッセージを出力抑止する
- 出力対象とするメッセージを指定する
- 指定したメッセージの重要度表示を変更して出力する

また，指定を省略すると，syslogfile に出力するメッセージの出力を抑止しません。

-s 出力有無

メッセージの出力，又は出力抑止を指定します。

Y：メッセージを出力します。

N：メッセージの出力を抑止します。

syslogfile へのメッセージ出力抑止については，マニュアル「HiRDB システム運用ガイド」を参照してください。

-c ALL

このオプションは，HiRDB が syslogfile に出力するすべてのメッセージを出力抑止する場合に指定します。

このオプションを指定する場合は，-s オプションに N を指定します。

《注意事項》

障害の切り分けができなくなるため，通常はこのオプションを指定しないでください。

-l メッセージの重要度

出力されるメッセージの変更後の重要度を指定します。

このオプションを指定する場合は，-s オプションに Y を指定します。

E：エラーメッセージ

W：警告メッセージ

Q：応答待ちメッセージ

I：インフォメーションメッセージ

syslogfile に出力するメッセージの重要度の変更については，マニュアル「HiRDB システム運用ガイド」を参照してください。

-m メッセージ ID〔，メッセージ ID〕...

出力，又は出力抑止をするメッセージ ID を指定します。ただし，メッセージ ID のハイフン（-）以降の重要度は記述しません（KFPS01820-E の場合には KFPS01820 を指定します）。

このオプションに，元から syslogfile に出力されないメッセージ ID を指定しても，指定しない場合と同じです。

《オペランドの規則》

- このオペランドは，複数行指定できます。
- ALL 指定，及びメッセージ ID 指定によって，同一メッセージに対する制御行が複数存在する場合は，後から指定した行が有効となります。

《ほかのオペランドとの関連》

- pd_mlg_file_size：メッセージの出力抑止をする場合，メッセージログファイルの最大容量を指定します。
- pd_mlg_msg_log_unit：メッセージを出力する場合，メッセージログの出力先（システムマネジャユニット，又はメッセージ出力元のユニット）を指定します。

9.2.46　HiRDB システム定義ファイルの共用化に関するオペランド

◆　PDCONFPATH 共用ディレクトリ名

～<パス名>

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

HiRDB システム定義ファイルの共用化をする場合，その共用ディレクトリ名を指定します。共用化したファイルは，このディレクトリ下に作成されます。

HiRDB システム定義ファイルの共用化については，「HiRDB システム定義ファイルの共用化」を参照してください。

《前提条件》

この機能を使用するには，ファイル共用機能（NFS）が必要となります。ファイル共用機能（NFS）については，OS のマニュアルを参照してください。なお，ネットワークを介したファイルアクセスであるため，性能遅延，データ欠損などが発生する場合があり，信頼性の面から使用を推奨しません。

9.2.47　クライアントグループに関するオペランド

◆ pdcltgrp -g クライアントグループ名称 -u グループごとの接続保証ユーザ数

クライアントグループの接続枠保証機能を使用する場合に指定します。クライアントグループの接続枠保証機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

-g クライアントグループ名称

～＜英字＞((1～2 文字))

クライアントグループの名称を指定します。

クライアントの種別	クライアント グループ名称	備考
X/Open XA インタフェース	XA※1	X/Open XA インタフェースで HiRDB にアクセスするクライアントのことです。クライアントが PC 又は WS であっても，X/Open XA インタフェースで HiRDB にアクセスすると，クライアントグループは XA になります。
PC クライアント	PC※2	Windows 系及び Linux 系クライアントのことです。
WS クライアント	WS※2	UNIX 系クライアントのことです。
ユーザが任意に決めたクライアントグループ※3	英大文字 1 文字※1，※4	－

（凡例）－：該当しません。

注※1

このクライアントグループ名称を指定するには，HiRDB クライアントのバージョンが次に示すどちらかの条件を満たす必要があります。

- 04-05

  05-00 及び 05-01 では指定できません。
- 05-02 以降

注※2

このクライアントグループ名称を指定するには，HiRDB クライアントのバージョンが次に示す条件を満たす必要があります。

- 04-00 以降

注※3

ユーザが任意に決めたクライアントグループは，10 グループまで指定できます。

注※4

ここで指定したクライアントグループ名称を，クライアント環境定義の PDCLTGRP オペランドに指定してください。pdcltgrp オペランドと PDCLTGRP オペランドの指定値が合っていない場合，ここでの指定を無視します。

-u　グループごとの接続保証ユーザ数

～＜符号なし整数＞

- HiRDB/シングルサーバの場合：**((1～2999))**
- HiRDB/パラレルサーバの場合：**((1～1999))**

各クライアントグループの接続保証数を指定します。

ほかのクライアントグループ又はユティリティからの HiRDB へのアクセスが集中した場合でも，ここで指定したユーザ数までは，該当するクライアントグループから HiRDB への接続が保証されます。

《注意事項》

- -u オプションで指定したユーザ数の合計が，pd_max_users オペランドの指定値以上にならないようにしてください。指定値以上になると，HiRDB を開始しません。
- データベース定義ユティリティ以外のユティリティは，この機能の対象外です。

《各見積もり式への影響》

pdcltgrp オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

9.2.48　プラグインに関するオペランド

◆ **pdplugin -n プラグイン名称**

～＜識別子＞((1～30 文字))

HiRDB で使用するプラグインの名称を指定します。使用しない場合は，省略してください。

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

◆ **pdplgprm -n プラグイン名称〔 -s 共用メモリサイズ〕**

プラグインの名称及びプラグインが使用する共用メモリの大きさを指定します。プラグインを使用しない場合，又はプラグインが各サーバで稼働しない場合は，このオペランドを省略してください。

《前提条件》

ここで指定するプラグインは，あらかじめ pdplgrgst コマンドで HiRDB に登録しておいてください。

-n プラグイン名称：～＜識別子＞((1～30 文字))

ここで指定するプラグイン名称については，プラグインに付いているマニュアルを参照してください。

-s 共用メモリサイズ：～＜符号なし整数＞((1～2000000))《0》（単位：キロバイト）

プラグインが使用する共用メモリの大きさをキロバイト単位で指定します。プラグインが使用する共用メモリの大きさについては，プラグインに付いているマニュアルを参照してください。

《各見積もり式への影響》

pdplgprm オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリ」の「計算式 1」
- 「フロントエンドサーバが使用する共用メモリの計算式」
- 「ディクショナリサーバが使用する共用メモリの計算式」の「計算式 1」
- 「バックエンドサーバが使用する共用メモリの計算式」の「計算式 1」

◆ pd_plugin_ixmk_dir = "インデクス情報ファイルを作成するディレクトリ名"又は "インデクス情報ファイルを作成する HiRDB ファイルシステム領域名"

～＜パス名＞

プラグインインデクスの遅延一括作成をする場合に，インデクス情報ファイルを作成するディレクトリ名を指定します。インデクス情報ファイルを HiRDB ファイルシステム領域に作成する場合は，HiRDB ファイルシステム領域名を指定します。ディレクトリ名及び HiRDB ファイルシステム領域名は絶対パス名で指定してください。

プラグインインデクスの遅延一括作成については，マニュアル「HiRDB システム運用ガイド」を参照してください。

《注意事項》

- ここで指定するディレクトリ（又は HiRDB ファイルシステム領域）を必ず作成しておいてください。存在しないディレクトリ名（又は HiRDB ファイルシステム領域名）を指定すると，プラグインインデクスの遅延一括作成を指定した UAP（クライアント環境定義に PDPLGIXMK ＝ YES を指定した環境で実行した UAP）の実行時にエラーとなります。
- UAP 実行後，データベース再編成ユティリティでプラグインインデクスの遅延一括作成をする前にこのオペランドの指定値を変更しないでください。変更すると，プラグインインデクスを遅延一括作成できなくなります。

9.2.49　バージョンアップに関するオペランド

◆ **pd_auto_vrup = Y ｜ N**

HiRDB をバージョンアップするときに実行する pdvrup コマンドを自動起動するかどうかを指定します。HiRDB のバージョンアップ方法については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

Y：自動起動します。

N：自動起動しません。pdvrup コマンドの入力が必要な場合は KFPS05203-Q メッセージが表示されます。このときに HiRDB 管理者が手動で pdvrup コマンドを入力します。

9.2.50　通信処理に関するオペランド

◆ **pd_service_port = スケジューラプロセスのポート番号**
～<符号なし整数>((5001～65535))

次の場合に，スケジューラプロセスのポート番号を指定します。

- 高速接続機能を使用する場合
  高速接続機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
- HiRDB サーバ側にファイアウォールや NAT が設置されている場合
  この場合の HiRDB の環境設定については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《指定値の目安》

- HiRDB/シングルサーバの場合，又は HiRDB クライアントと HiRDB サーバの間にファイアウォールがある場合は，このオペランドを適用してください。
- サーバと接続するときの通信回数を削減できるため，広域 LAN を介して接続するシステムではこのオペランドを適用してください。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合，ユニット制御情報定義では必ずこのオペランドを指定してください。また，各ユニットで異なるポート番号を指定してください。同じポート番号を指定すると，系切り替え時発生時にどちらかのユニットで系切り替えが失敗します。
- ユニット制御情報定義とシステム共通定義の両方でこのオペランドを指定した場合，ユニット制御情報定義の指定値が有効になります。
- HiRDB のポート番号（pd_name_port オペランド又は pdunit オペランドの-p オプションに指定したポート番号）と同じポート番号を指定した場合，このオペランドの指定値が無効となり，高速接続機能を使用する場合及び HiRDB サーバ側にファイアウォールや NAT が設定されている場合の運用ができなくなります。このとき，KFPS00860-W メッセージが出力されます。

- 高速接続機能を使用して pd_max_users オペランドの値以上の接続要求を同時に発行した場合，接続要求をメッセージキューから取り出すフロントエンドサーバやシングルサーバの稼働数が不足します。そのため，接続要求をメッセージキューから取り出せなくなり，メッセージキュー監視機能によってユニットが停止することがあります。pd_max_users オペランドの値以上の同時接続要求が発生しないようにしてください。メッセージキュー監視機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- スケジューラプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- 複数のフロントエンドサーバを持つパラレルサーバで適用すると，接続するフロントエンドサーバが固定になり，負荷分散ができなくなります。このため，接続するフロントエンドサーバのバランスを検討してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_scd_port
- pdunit -s

◆ pd_name_fixed_port_lookup = Y ｜ N

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

ユニット間のシステムサーバと通信する場合のアドレス解決方法を指定します。ここでいうアドレス解決とは，システムサーバの IP アドレスとポート番号を取得する処理をいいます。

Y：自ユニットの共用メモリでアドレス解決をします。

N：全ユニットに通信をして（ブロードキャスト）アドレス解決をします。

このオペランドに Y を指定する場合，次に示すオペランドを指定して，ユニット間通信をするシステムサーバのポート番号を指定する必要があります。ポート番号を指定しない場合，HiRDB を起動できません。ポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

ポート種別	オペランド
HiRDB のポート	pd_name_port オペランド又は pdunit オペランドの-p オプション
トランザクションサーバプロセスのポート	pd_trn_port オペランド又は pdunit オペランドの-t オプション
スケジューラプロセスのポート	pd_scd_port オペランド又は pdunit オペランドの-s オプション
ユニット監視プロセスのポート※1	pd_alv_port オペランド又は pdunit オペランドの-a オプション
メッセージログサーバプロセスのポート※2	pd_mlg_port オペランド又は pdunit オペランドの-m オプション

注※1　シングルサーバ及びシステムマネジャがないユニットでは，ユニット監視プロセスが起動しないため，ポート番号を指定する必要はありません。

注※2　pd_mlg_msg_log_unit オペランドに local を指定した場合，ブロードキャストが発生しないため，ポート番号を指定する必要はありません。また，pd_mlg_msg_log_unit オペランドに manager

を指定した場合，システムマネジャがないユニットでは，メッセージログサーバプロセスが起動しないため，ポート番号を指定する必要はありません。

《利点》

このオペランドに Y を指定し，ユニット間通信をするシステムサーバのポート番号を指定すると，アドレス解決時にブロードキャストが発生しません。そのため，ユニット数が多い場合に発生する次のような問題を避けられます。

- 通信資源の圧迫（ネットワーク負荷，ポートの消費）
- 性能の劣化

《注意事項》

- このオペランドに N を指定すると，アドレス解決時に全ユニットに対するブロードキャストが発生することがあります。ブロードキャストが発生する場合を次に示します。

  ・最初のアドレス解決時

  ・アドレス解決したシステムサーバがダウンした後のアドレス解決時

  ・アドレス解決したシステムサーバのユニットがダウンした後のアドレス解決時
- HiRDB/シングルサーバ又は一つのユニットだけで構成される HiRDB/パラレルサーバでは，ユニット間通信が発生しないため，このオペランドの指定有無に関係なく N を指定したものとして動作します。
- クライアントの接続形態は高速接続を使用してください。通常接続を使用した場合，クライアント接続時に一時的にブロードキャストが発生します。クライアントの接続形態については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

◆ pd_scd_port = スケジューラプロセスのポート番号 ～<符号なし整数>((5001～65535))

次の場合に，スケジューラプロセスのポート番号を指定します。

- pd_name_fixed_port_lookup オペランドに Y を指定した場合
- 高速接続機能を使用する場合

  高速接続機能については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
- HiRDB サーバ側にファイアウォールや NAT が設置されている場合

  この場合の HiRDB の環境設定については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《注意事項》

- ユニットごとに異なるポート番号を指定する場合は，pdunit オペランドの-s オプションを指定してください。
- HiRDB のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。

- このオペランドは次に示すオペランドと関連があります。
  ・pd_service_port
  ・pdunit -s

◆ pd_trn_port = トランザクションサーバプロセスのポート番号 ～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，トランザクションサーバプロセスのポート番号を指定します。

《注意事項》

トランザクションサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pdunit オペランドの-t オプションと関連があります。

◆ pd_mlg_port = メッセージログサーバプロセスのポート番号 ～<符号なし整数>((5001～65535))

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，メッセージログサーバプロセスのポート番号を指定します。

《注意事項》

- メッセージログサーバプロセスが起動しないユニットではポート番号は使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。メッセージログサーバプロセスが起動しないユニットは，二つ以上のユニットで構成される HiRDB/パラレルサーバで pd_mlg_msg_log_unit の指定値が manager の場合，システムマネジャがないユニットです。
- メッセージログサーバプロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pdunit オペランドの-m オプションと関連があります。

◆ pd_alv_port = ユニット監視プロセスのポート番号 ～<符号なし整数>((5001～65535))

このオペランドは HiRDB/パラレルサーバ限定のオペランドです。

pd_name_fixed_port_lookup オペランドに Y を指定した場合に，ユニット監視プロセスのポート番号を指定します。

《注意事項》

- ユニット監視プロセスが起動しないユニットではポート番号は使用されませんが，定義したポート番号がほかのポート番号と異なるかどうかはチェックされます。ユニット監視プロセスは，二

つ以上のユニットで構成される HiRDB/パラレルサーバでシステムマネジャがあるユニットを除いて，すべてのユニットで起動しません。

- ユニット監視プロセスのポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

《ほかのオペランドとの関連》

- pd_name_fixed_port_lookup オペランドに Y を指定する必要があります。
- このオペランドは，pdunit オペランドの-a オプションと関連があります。

◆ **pd_change_clt_ipaddr = 0 ｜ 1**

HiRDB サーバから HiRDB クライアントへの通信時に使用するネットワークを指定します。通常，このオペランドを指定する必要はありません。

0：

HiRDB サーバから HiRDB クライアントへの通信時，クライアント環境定義の PDCLTRCVADDR オペランドに指定した IP アドレスがあるネットワークを使用します。PDCLTRCVADDR オペランドを省略した場合は標準ホストの IP アドレスが仮定されます。

1：

HiRDB サーバから HiRDB クライアントへの通信時，HiRDB クライアントから HiRDB サーバへの通信に使用したネットワークを使用します。

《クライアント環境定義との関連》

クライアント環境定義 PDCONTYPE に PASSIVE を指定，又は指定を省略しているクライアントからの接続は，このオペランドの指定値に従った動作になります。

クライアント環境定義 PDCONTYPE に ACTIVE を指定しているクライアントからの接続は，このオペランドの指定値にかかわらず，クライアントがサーバに対して接続した時のネットワークを使用して，通信をします。

◆ **pd_registered_port = "ポート番号の予約範囲"〔，"ポート番号の予約範囲"〕...**

～〈文字列〉

HiRDB 予約ポート機能を使用する場合に，HiRDB が通信処理で使用するポート番号の範囲を指定します。

HiRDB 予約ポート機能は，サーバ間通信及び対クライアント通信で有効になります。詳細は，pd_registered_port_level オペランドの説明を参照してください。なお，ポート番号の使用量が少ない場合には，HiRDB 予約ポート機能を使用する必要はありません。

《オペランドの指定方法》

6000～8999，12500～29999，30500～44999 の計 35,000 個の範囲でポート番号を使用する場合の例を次に示します。

set pd_registered_port = "6000：8999"，"12500：29999"，"30500：44999"

《利点》

HiRDB サーバと HiRDB クライアント間，又は HiRDB サーバ間の通信処理で使用するポート番号は OS が自動的に割り当てています。通信処理が大量に発生すると，ポート番号の不足によって処理が中断したり，又はほかのプログラムの通信処理に影響を与えたりします。

このオペランドで HiRDB 専用のポート番号の範囲を指定すると，これらの現象を防げます。

《指定値の目安》

- 指定できるポート番号の範囲は 5001～49151 です。
- HiRDB が使用するポート数については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- /etc/services（NIS 又は DNS 環境の場合はそれぞれに定義した場所）には，このオペランドで指定したポート番号は登録しないでください。pd_registered_port_check オペランドに Y，C，又は W を指定すると，/etc/services との重複チェックをします。

《指定値のチューニング方法》

次に示す計算式で求めた値が，必要となるポート番号の総数の目安です。

a＋b＋100

a：HiRDB 予約ポートの使用数（#OF REGISTERED PORTS）

b：HiRDB 予約ポートオーバ時の OS 自動割り当てポートの使用数（#OF ASSIGNED PORTS）

これらの情報は，統計解析ユティリティのシステムの稼働に関する統計情報で取得できます。

《オペランドの規則》

- 予約範囲は最大 10 個指定できます。
- 予約範囲を二つ以上指定した場合，範囲に含まれるポート番号が重複しないようにしてください。
- 終了ポート番号は開始ポート番号以上の値でなければいけません。

《注意事項》

- 相互系切り替え構成などで一つのサーバマシンに複数のユニットがある場合は，ユニット制御情報定義で必ずこのオペランドを指定してください。また，各ユニットで異なるポート番号を指定してください。同じポート番号を指定すると，系切り替え時発生時にどちらかのユニットで系切り替えが失敗します。
- HiRDB 予約ポート機能を使用できないポート番号では，このオペランドで指定したポート番号は使用されません。HiRDB 予約ポート機能の対象範囲については，pd_registered_port_level オペランドを参照してください。
- HiRDB 予約ポート機能を使用する場合のポート番号の指定方法，及びほかのポート番号との重複に関する注意事項については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。
- ユニット制御情報定義とシステム共通定義の両方でこのオペランドを指定した場合，ユニット制御情報定義の指定値が有効になります。

《各見積もり式への影響》

pd_registered_port オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

◆ **pd_registered_port_check = Y ｜ N ｜ C ｜ W**

pd_registered_port オペランドで範囲指定したポート番号と/etc/services（NIS 又は DNS 環境の場合はそれぞれに定義した場所）に登録されたポート番号との重複チェックをするかどうかを指定します。

Y：

重複チェックをします。重複している場合，KFPS00348-E メッセージを出力して HiRDB の開始処理を中断します。

N：

重複チェックをしません。

C：

重複チェックをします。重複している場合，重複しているポート番号は HiRDB 予約ポート機能の対象外になります。

W：

重複チェックをします。重複している場合，KFPS00354-W メッセージを出力します。重複しているポート番号は HiRDB 予約ポート機能の対象外になります。

《前提条件》

pd_registered_port オペランドを指定している必要があります。

《指定値の目安》

- ポート番号の重複があると，不正電文の受信や送信電文の未達など HiRDB の通信処理に影響を及ぼすことがあります。
- Y，C，又は W を指定すると，NIS 又は DNS 環境のときにプロセスサーバプロセスの起動が遅くなることがあります。

◆ **pd_ipc_send_retrycount = プロセス間通信の送信リトライ処理の連続繰り返し回数～＜符号なし整数＞((1～32767))《200》**

プロセス間通信の送信リトライ処理の連続繰り返し回数を指定します。

このオペランドは pd_ipc_send_retrysleeptime オペランドと関連しています。

（例）

- pd_ipc_send_retrycount = 500
- pd_ipc_send_retrysleeptime = 2

上記のようにオペランドを指定すると，送信リトライ処理 1 回の繰り返し単位が 500 回連続実行で 2 秒スリープとなります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドの指定値を大きくすると CPU 占有率が高くなります。

◆ **pd_ipc_send_retrysleeptime = プロセス間通信の送信リトライ処理のスリープ時間～＜符号なし整数＞((0～60))《0》（単位：秒）**

プロセス間通信の送信リトライ処理のスリープ時間を指定します。

このオペランドは pd_ipc_send_retrycount オペランドと関連しています。

（例）

- pd_ipc_send_retrycount = 500
- pd_ipc_send_retrysleeptime = 2

上記のようにオペランドを指定すると，送信リトライ処理 1 回の繰り返し単位が 500 回連続実行で 2 秒スリープとなります。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドの指定値を大きくすると送信完了時間が長くなります。

◆ **pd_ipc_send_count = サーバ間の送信処理のリトライ回数～＜符号なし整数＞((1～32767))《60》**

サーバ間の送信処理が終了するまでの送信処理のリトライ回数を指定します。なお，1 回で最大 5 秒間のリトライを続けます。省略値では 5 秒×60 回となり 300 秒間リトライを続けます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。送信タイムアウトが頻発する場合に，このオペランドの指定値を大きくしてください。

◆ **pd_ipc_recv_count = サーバ間の受信処理のリトライ回数～＜符号なし整数＞((1～32767))《120》**

サーバ間の受信処理が終了するまでの受信処理のリトライ回数を指定します。なお，1 回で最大 5 秒間のリトライを続けます。省略値では 5 秒×120 回となり 10 分間リトライを続けます。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

◆ **pd_ipc_conn_nblock = Y ｜ N**

HiRDB のサーバ間通信（ユニット間の通信）で，コネクション確立をノンブロックモードで行うかどうか指定します。

Y：コネクション確立をノンブロックモードで行います。

N：コネクション確立をブロックモードで行います。

《利点》

このオペランドに Y を指定した場合，HiRDB のサーバ間通信（ユニット間の通信）で，connect()システムコールをノンブロックモードで発行できます。それによって，LAN 障害時に connect()システムコールが数十秒間（OS 依存）プロセスをブロックしていたのが，pd_ipc_conn_nblock_time オペランドで指定した時間まで短くなります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_ipc_conn_nblock_time
- pd_ipc_conn_interval
- pd_ipc_conn_count
- pd_host_watch_interval

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《備考》

クライアントとサーバ間の通信でのコネクション確立については，クライアント環境定義 PDNBLOCKWAITTIME，又は pd_ipc_clt_conn_nblock オペランドで指定してください。

◆ pd_ipc_conn_nblock_time = ノンブロックモードでのコネクション確立監視時間～<符号なし整数>((1～120))《8》（単位：秒）

HiRDB のサーバ間通信（ユニット間の通信）におけるノンブロックモードでのコネクション確立処理のリトライ時間と，コネクションの確立監視時間を指定します。

《前提条件》

このオペランドは，pd_ipc_conn_nblock オペランドに Y を指定した場合に有効になります。

《利点》

不当な通信エラーが多発している場合に，このオペランドの指定値を大きくすると，通信エラーを減らせることがあります。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

不当な通信エラーが多発している場合は，OS の ping コマンドなどを使用して，HiRDB のサーバ間の到達時間を計ってください。このオペランドの値より時間が掛かる場合は，到達時間より大きい値をこのオペランドに指定してください。

《注意事項》

- このオペランドの指定値を大きくすると，通信できなくなった場合のタイムアウト時間が延びます。このため，ほかの処理に影響を与えることがあります。
- このオペランドには，コネクション確立をブロックモードで行ったときの OS 待ち時間よりも大きい値を指定しないでください。指定した場合は，OS の待ち時間で接続タイムアウトになります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

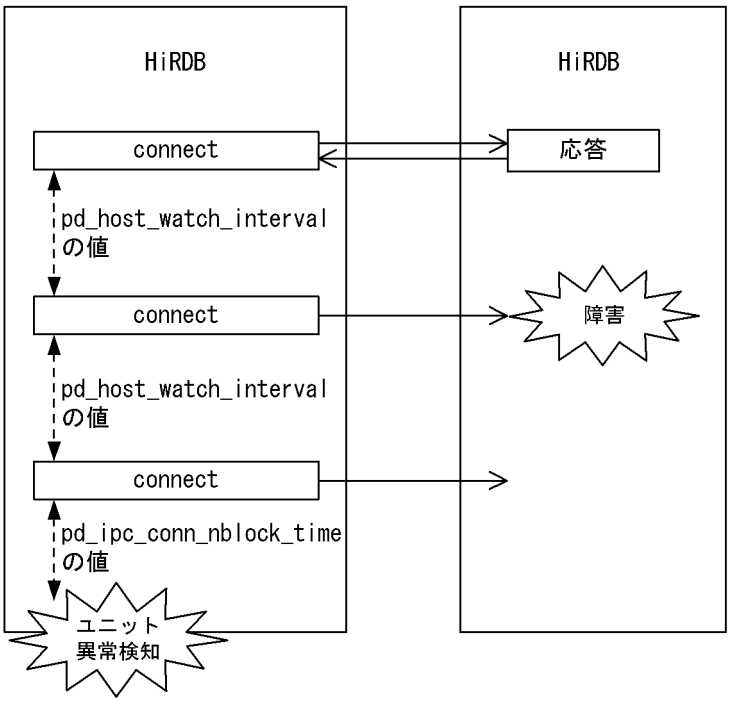
- pd_ipc_conn_nblock
- pd_ipc_conn_interval
- pd_ipc_conn_count

《備考》

HiRDB は一定間隔（pd_host_watch_interval オペランドの値：省略値は 10 秒）でほかのユニットに対して通信を行い，ほかのユニットの稼働状況を確認しています。

監視対象ユニットが応答を返した直後に LAN 障害が発生すると，一定時間（pd_host_watch_interval オペランドの値）待った後に次の通信処理をします。その後，通信のリトライ処理が pd_ipc_conn_nblock_time オペランドの値（デフォルト 8 秒）の間実行されます。そのため，監視対象ユニットの異常終了が検知されるまでの時間は，両オペランドの値を加算した時間になります。

HiRDB が監視対象ユニットの異常終了を検知するまでの時間の関係を次に示します。



◆ **pd_ipc_conn_interval = コネクション確立処理のリトライ間隔**

～＜符号なし整数＞((0～50))《1》（単位：秒）

HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ間隔を指定します。次に示す場合にコネクション確立処理をリトライします。

- 接続タイムアウトになった場合
- Listen キューが不足した場合

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。
- このオペランドを指定する場合は，次の条件を満たすようにしてください。

  このオペランドの指定値＜ pd_ipc_conn_nblock_time の値

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

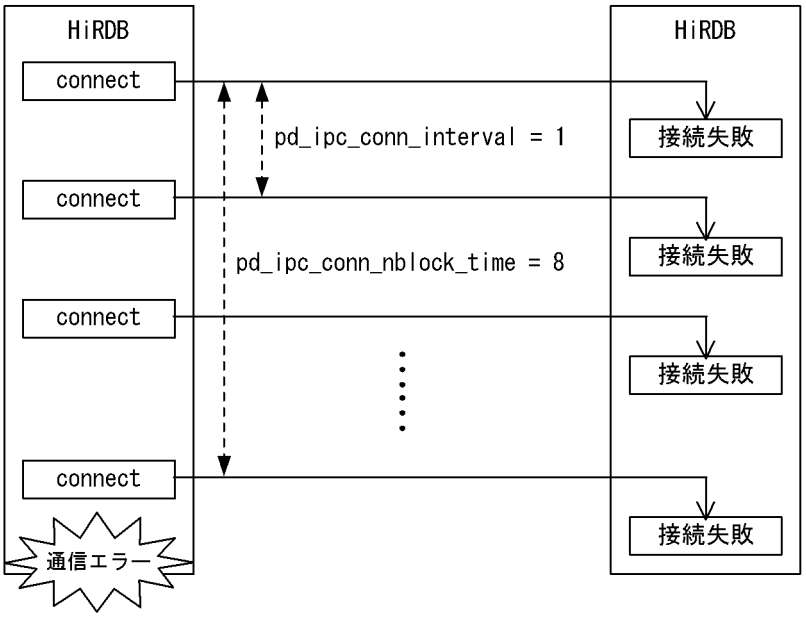
- pd_ipc_conn_count
- pd_ipc_conn_nblock

- pd_ipc_conn_nblock_time

《備考》

ノンブロックモード及びブロックモード通信時のリトライ処理の例を次に示します。

・ノンブロックモード通信時のリトライ処理（pd_ipc_conn_nblock = Yの場合）



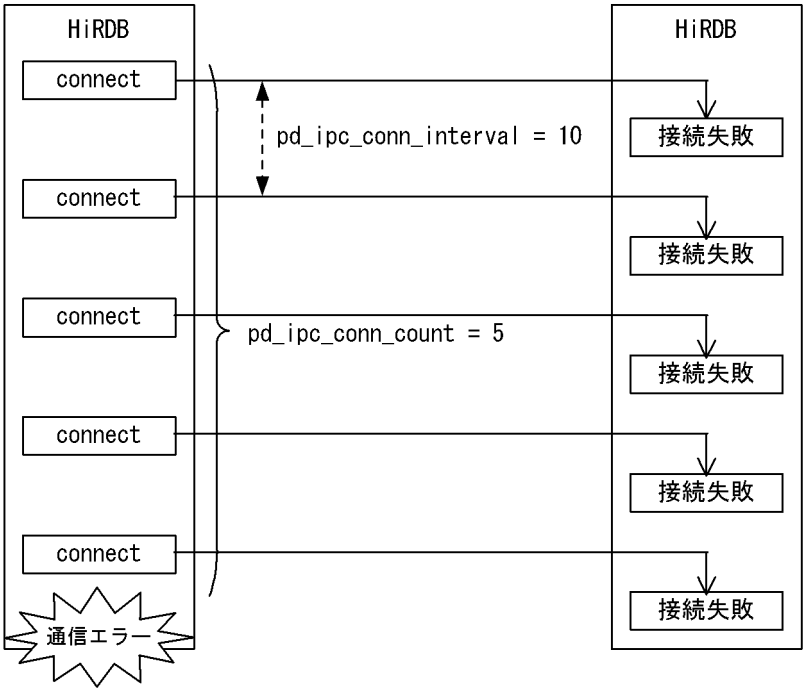
〔説明〕

次に示すように指定すると，ノンブロックモードのコネクション確立処理を，1 秒間隔で 8 秒間リトライします。

・pd_ipc_conn_nblock_time＝8

・pd_ipc_conn_interval＝1

・ブロックモード通信時のリトライ処理（pd_ipc_conn_nblock = Nの場合）



〔説明〕

次に示すように指定すると，ブロックモードでのコネクション確立処理を 10 秒間隔で 5 回リトライします。

・pd_ipc_conn_interval = 10

・pd_ipc_conn_count = 5

◆ **pd_ipc_conn_count =コネクション確立処理のリトライ回数**

～＜符号なし整数＞((0～600))《50》

HiRDB のサーバ間通信で，データ送信時に行うコネクション確立処理のリトライ回数を指定します。次に示す場合に，コネクション確立処理をリトライします。

- 接続タイムアウトになった場合
- Listen キューが不足した場合

このオペランドは pd_ipc_conn_interval オペランドと関連しています。

（例）

- pd_ipc_conn_interval = 10
- pd_ipc_conn_count = 5

このようにオペランドに指定すると，コネクション確立処理を 10 秒ごとに最大 5 回行います。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《注意事項》

ノンブロックモードのコネクション確立処理が実行される場合，このオペランドの指定は無効になります。

pd_ipc_conn_nblock オペランドに Y を指定すると，ノンブロックモードのコネクション確立処理が実行されます。

◆ **pd_ipc_clt_conn_nblock = Y ｜ N**

HiRDB クライアントとの通信で，コネクション確立をノンブロックモードで行うかどうかを指定します。

Y：コネクション確立をノンブロックモードで行います。pd_ipc_clt_conn_nblock_time オペランドに指定した値の間隔で監視します。

N：コネクション確立をブロックモードで行います。バージョン 09-02 までと同様の動作です。

《利点》

このオペランドに Y を指定した場合，HiRDB クライアントとの通信で，connect()システムコールをノンブロックモードで発行できます。それによって，LAN 障害時に connect()システムコールが数十秒間（OS 依存）プロセスをブロックしていたのが，pd_ipc_clt_conn_nblock_time オペランドで指定した時間まで短くなります。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

《クライアント環境定義との関連》

クライアント環境定義の PDNBLOCKWAITTIME の指定によって，このオペランドの指定値が無効になることがあります。クライアント環境定義の PDNBLOCKWAITTIME との組み合わせで，コネクション確立をどちらのモードで行うかを次の表に示します。

クライアント環境定義 PDNBLOCKWAITTIME の指定値	pd_ipc_clt_conn_nblock オペランドの指定値	
	Y	N
0	• **サーバがクライアントにコネクション確立する場合** ノンブロックモード。 このとき，pd_ipc_clt_conn_nblock_time オペランドに指定した監視時間で監視します。 • **クライアントがサーバにコネクション確立する場合** ブロックモード。	ブロックモード。
1 以上	ノンブロックモード。 このとき，クライアント環境定義の PDNBLOCKWAITTIME に指定した監視時間で監視します。	

◆ pd_ipc_clt_conn_nblock_time = ノンブロックモードでのコネクション確立監視時間（HiRDB クライアントとの通信）

～<符号なし整数>((1～120))《8》（単位：秒）

HiRDB クライアントとの通信におけるノンブロックモードでのコネクションの確立監視時間を指定します。

《前提条件》

このオペランドは pd_ipc_clt_conn_nblock オペランドに Y を指定した場合に有効になります。

《指定値の目安》

通常，このオペランドを指定する必要はありません。

不当な通信エラーが多発している場合は，OS の ping コマンドなどを使用して，到達時間を計ってください。このオペランドの値より時間が掛かる場合は，到達時間より大きい値をこのオペランドに指定してください。

《注意事項》

このオペランドには，コネクション確立をブロックモードで行ったときの OS 待ち時間よりも大きい値を指定しないでください。指定した場合は，OS の待ち時間で接続タイムアウトになります。

◆ pd_ipc_file_dir = 通信情報ファイルディレクトリ

～<パス名>((80 バイト以内))《/dev/HiRDB/pth》

通信情報ファイルディレクトリのパスを指定します。指定を省略した場合には通信情報ファイルディレクトリ変更機能は適用されないため，通信情報ファイルディレクトリは/dev/HiRDB/pth となります。

《前提条件》

このオペランドに，存在するディレクトリを絶対パスで指定してください。指定するディレクトリの詳細は，マニュアル「HiRDB システム導入・設計ガイド」の「通信情報ファイルディレクトリの作成」を参照してください。

《利点》

マルチ HiRDB 環境でそれぞれ専用の通信情報ファイルディレクトリを指定できます。詳細は，マニュアル「HiRDB システム運用ガイド」の「通信情報ファイルディレクトリの変更方法」を参照してください。

《クライアント環境定義との関連》

- PDIPCFILEDIR

  クライアント環境定義 PDIPCFILEDIR にはこのオペランドの指定と同一のパスを指定してください。詳細は，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《注意事項》

- 通信情報ファイルディレクトリ変更機能を使用しない場合は，このオペランドを指定しないで省略してください。通信情報ファイルディレクトリとしてデフォルトのパス名を使用する場合はこのオペランドを指定しないでください。
- ほかの用途で使用しているディレクトリ（pd_tmp_directory オペランドで指定したワークファイル出力先ディレクトリなど）をこのオペランドに指定しないでください。
- 系切り替え構成の場合，現用系と予備系でこのオペランドに同一のパスを指定してください。

◆ pd_crypto_service_port = スケジューラプロセスの暗号化通信（SSL）接続用のポート番号～<符号なし整数>((5001～65535))

スケジューラプロセスがクライアントからの暗号化通信（SSL）の接続要求を受け付けるポート番号を指定します。

暗号化通信プロトコルは TLS1.3 を使用します。

《利点》

クライアントーサーバ間の通信を暗号化できます。

《前提条件》

以下のオペランドを指定してください。

- pd_crypto_private_key
- pd_crypto_certificate_file

《クライアント環境定義との関連》

クライアント−サーバ間の通信を暗号化する場合，クライアント環境定義 PDSOCKETCRYPTO の指定が必要です。詳細は，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《注意事項》

- 他オペランドで指定していない値や OS の自動割り当てポートの範囲外の値をこのオペランドに指定してください。HiRDB 起動時に前提条件に記載したオペランドの指定がない場合やポートが確保できない場合は，KFPS00862-E メッセージを出力して，該当ユニットは異常終了します。
- シングルサーバ及びフロントエンドサーバを配置していないユニットでは，このオペランドの指定は無視します。

◆ pd_crypto_private_key = 通信暗号化機能に使用する秘密鍵～<パス名>((255 バイト以内))

通信暗号化機能に使用する秘密鍵のファイルパスを指定します。

《利点》

クライアント−サーバ間の通信を暗号化できます。

《前提条件》

以下のオペランドを指定してください。

- pd_crypto_service_port
- pd_crypto_certificate_file

《注意事項》

- 指定する秘密鍵のファイルは HiRDB 管理者ユーザに対して読み込み権限を与えてください。
- 通信暗号化機能を使用する場合，このオペランドを指定してください。
- ほかの用途で使用しているディレクトリ（pd_tmp_directory オペランドで指定したワークファイル出力先ディレクトリなど）をこのオペランドに指定しないでください。
- 系切り替え構成の場合，現用系と予備系でこのオペランドに同一のパスを指定してください。

◆ pd_crypto_certificate_file = 通信暗号化機能に使用するサーバ証明書～<パス名>((255 バイト以内))

通信暗号化機能に使用するサーバ証明書のファイルパスを指定します。

《利点》

クライアント−サーバ間の通信を暗号化できます。

《前提条件》

以下のオペランドを指定してください。

- pd_crypto_service_port
- pd_crypto_private_key

《注意事項》

- 指定するサーバ証明書は HiRDB 管理者ユーザに対して読み込み権限を与えてください。
- 通信暗号化機能を使用する場合，このオペランドを指定してください。
- 前提条件のオペランドを指定しないで，pd_crypto_certificate_file オペランドだけを指定している場合，HiRDB ユニット起動時に通信暗号化機能のチェックは行いません。通信暗号化機能は無効化の状態で起動します。
- ほかの用途で使用しているディレクトリ（pd_tmp_directory オペランドで指定したワークファイル出力先ディレクトリなど）をこのオペランドに指定しないでください。
- 系切り替え構成の場合，現用系と予備系でこのオペランドに同一のパスを指定してください。

◆ pd_ipc_so_keepalive = Y | N

サーバのユニット間通信で，定期的に KeepAlive パケットを送信する機能を使用するかを指定します。

Y：サーバのユニット間通信で定期的に KeepAlive パケットを送信します。

N：サーバのユニット間通信で定期的に KeepAlive パケットを送信しません。

《指定値の目安》

サーバのユニット間通信の経路上に，無通信時間監視機能を備えているルータやファイアウォールなどが存在する場合にこのオペランドに Y を指定してください。

- メリット及びデメリット

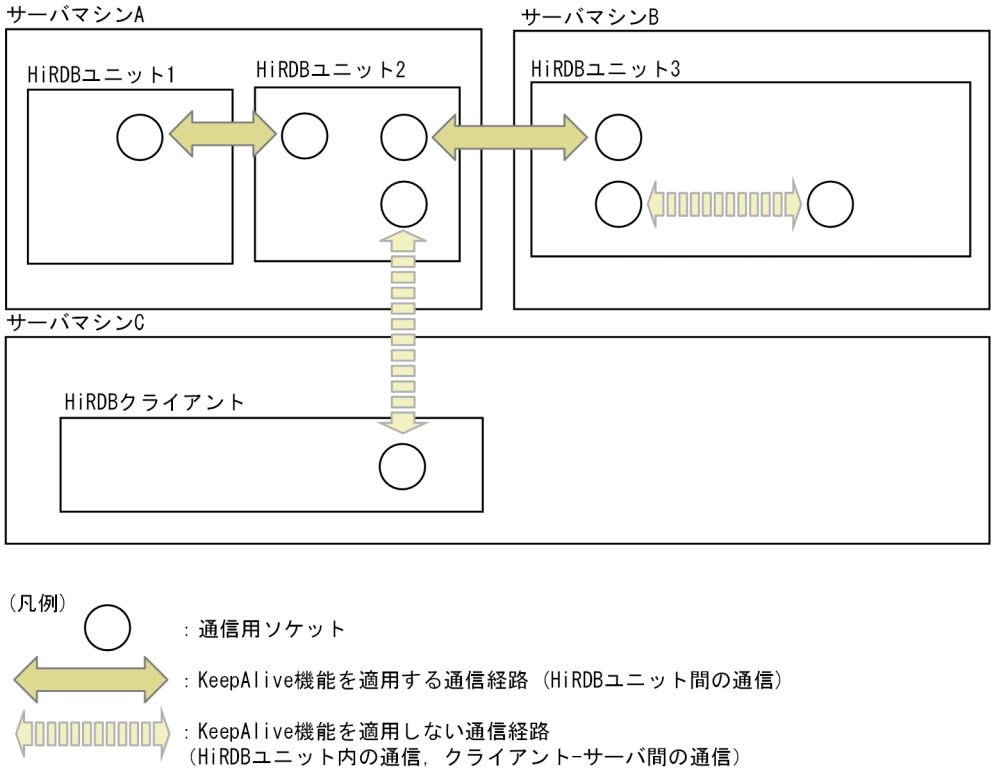
  このオペランドに Y を指定すると，サーバのユニット間通信の間隔が空いた場合に定期的に KeepAlive パケットを送受信するようになります。

  これによって，ルータやファイアウォールなどの無通信時間監視機能が有効になります。また，サーバのユニット間の接続が不当に切断されることを防ぎますが，パケットの送信が定期的に発生するためネットワーク帯域の負荷が増加します。

  （240 バイト×pd_max_users×HiRDB システム内ユニット数）／MIN（pd_ipc_tcp_keepidle, pd_ipc_tcp_keepintvl）の値が 1 メガバイトを超える場合は pd_ipc_tcp_keepidle オペランド，pd_ipc_tcp_keepintvl オペランドの値を大きくしてください。

《備考》

このオペランドに Y を指定した場合の，HiRDB ユニット間の KeepAlive 機能が適用される通信経路を次に示します。



◆ pd_ipc_tcp_keepidle = KeepAlive パケットの送信間隔 ～＜符号なし整数＞（単位：秒）

- Linux の場合：((1～32767))
- AIX の場合：((1～15783))

KeepAlive パケットを送信する間隔を指定します。

《前提条件》

このオペランドは，pd_ipc_so_keepalive オペランドに Y を指定した場合に有効となります。

《指定値の目安》

- このオペランドを指定する場合は，ルータやファイアウォールなどの無通信時間監視機能の監視時間より短い時間を設定してください。また，pd_ipc_tcp_keepidle ≧ pd_ipc_tcp_keepintvl の関係になるように値を設定してください。この関係を満たさない場合は，KeepAlive パケットが意図した間隔で送信されないおそれがあります。
- このオペランドを省略した場合は，Linux のオペレーティングシステムパラメタ tcp_keepalive_time 又は AIX のネットワークチューニングパラメタ tcp_keepidle の設定値が適用されます。

  詳細は OS のマニュアルを参照してください。

◆ **pd_ipc_tcp_keepintvl = ネットワーク障害検知後の KeepAlive パケットの送信間隔 ～＜符号なし整数＞（単位：秒）**

- Linux の場合：((1～32767))
- AIX の場合：((1～1147))

pd_ipc_tcp_keepidle オペランドの指定値を経過しても送信した KeepAlive パケットに対する応答がなかった場合の，KeepAlive パケット送信間隔を指定します。

《前提条件》

このオペランドは，pd_ipc_so_keepalive オペランドに Y を指定した場合に有効となります。

《指定値の目安》

- このオペランドを指定する場合は，ルータやファイアウォールなどの無通信時間監視機能の監視時間より短い時間を設定してください。また，pd_ipc_tcp_keepidle ≧ pd_ipc_tcp_keepintvl の関係になるように値を設定してください。この関係を満たさない場合は，KeepAlive パケットが意図した間隔で送信されないおそれがあります。
- このオペランドを省略した場合は，Linux のオペレーティングシステムパラメタ tcp_keepalive_intvl 又は AIX のネットワークチューニングパラメタ tcp_keepintvl の設定値が適用されます。

  詳細は OS のマニュアルを参照してください。

◆ **pd_ipc_tcp_keepcnt = ネットワーク障害検知後の KeepAlive パケット再送回数の上限値 ～＜符号なし整数＞**

- Linux の場合：((1～127))
- AIX の場合：((1～122))

ネットワーク障害検知後の KeepAlive パケット再送回数の上限値を指定します。

《前提条件》

このオペランドは，pd_ipc_so_keepalive オペランドに Y を指定した場合に有効となります。

《指定値の目安》

このオペランドを省略した場合は，Linux のオペレーティングシステムパラメタ tcp_keepalive_probes 又は AIX のネットワークチューニングパラメタ tcp_keepcnt の設定値が適用されます。

詳細は OS のマニュアルを参照してください。

9.2.51 Java に関するオペランド

Java ストアドプロシジャ又は Java ストアドファンクションを使用する場合に，Java に関するオペランドを指定します。Java ストアドプロシジャ及び Java ストアドファンクションについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

なお，HiRDB の稼働プラットフォームによっては Java ストアドプロシジャ又は Java ストアドファンクションを使用できません。詳細については，マニュアル「HiRDB システム運用ガイド」の「Java ストアドプロシジャ，Java ストアドファンクションを使用できる環境」を参照してください。

◆ **pd_java_option = "Java オプション"〔，"Java オプション"〕...**

～＜文字列＞

Java 仮想マシンの起動オプションを指定します。起動オプションについては，Java 仮想マシンのドキュメントを参照してください。

《オペランドの規則》

- 一つの Java オプションの文字列長は 255 文字以内とします。
- 全 Java オプションの文字列長の合計は 1024 文字以内とします。
- Java オプションに " を指定できません。
- Java オプションは最大 20 個指定できます。

《オペランドの指定方法》

Java 仮想マシンが使用するヒープサイズの初期値を 32 メガバイト，ヒープサイズの最大値を 64 メガバイトに設定する場合の例を次に示します。

pd_java_option = "-Xms32m","-Xmx64m"

◆ **pd_java_routine_stack_size = 外部 Java ルーチンが使用するスタック領域長**

～＜符号なし整数＞((1024～2146435072))《131072》（単位：バイト）

外部 Java ルーチンが使用するスタック領域長をバイト単位で指定します。

《指定値の目安》

- Java オプションで指定したスタック領域長以上の値を指定してください。

- Java 仮想マシンの起動オプションに，Java 仮想マシンのスタック領域長とネイティブメソッドスタック領域長の両方を指定するオプションがある場合，このオペランドにはネイティブメソッドスタック領域長以上の値を指定してください。

《注意事項》

- Java オプションで指定したスタック領域長より小さい値を指定すると，HiRDB が正しく動作しないことがあります。
- 使用スタック領域長がこのオペランドの値を超えると，HiRDB が正しく動作しないことがあります。ただし，Java 仮想マシンがスタックオーバフローを検知した場合は正しく動作します。

◆ **pd_java_archive_directory = "JAR ファイル格納ディレクトリ"**

～＜パス名＞《$PDDIR/java》

Java ストアドプロシジャ又は Java ストアドファンクションで使用する JAR ファイルを格納するディレクトリ名を絶対パス名で指定します。

《注意事項》

- このオペランドを指定する場合，ここで指定するディレクトリを JAR ファイルのインストール前に作成しておく必要があります。
- JAR ファイル格納ディレクトリは，JAR ファイルを格納するための専用のディレクトリです。
- JAR ファイル格納ディレクトリには，インストールした JAR ファイル以外を格納しないでください。

《オペランドの規則》

- パス名は 255 文字以内とします。

◆ **pd_java_classpath = "Java クラスパス"**

～＜パス名＞

Java 仮想マシンで使用するクラスパスを絶対パス名で指定します。

このオペランドに指定したパスに含まれるクラスは，Java ストアドプロシジャ又は Java ストアドファンクションの処理手続きとして実行される Java メソッドから参照できます。

Java ストアドプロシジャ又は Java ストアドファンクションの外部ルーチン名として指定した JAR ファイルとこのオペランドに指定したパスに同じ名称のクラスがある場合，このオペランドに指定したパスが優先されます。

《オペランドの規則》

- パス名は 1024 文字以内とします。

◆ **pd_java_runtimepath = "Java Runtime Environment のルートディレクトリ"**

～＜パス名＞《$PDDIR/jre》

Java Runtime Environment のルートディレクトリを絶対パス名で指定します。

《オペランドの規則》

- パス名は 255 文字以内とします。

《注意事項》

JRE（Java Runtime Environment）の同梱を廃止したため，HiRDB を 07-03 より前のバージョンから 07-03 以降にバージョンアップする場合，このオペランドを追加するか，又はこのオペランドの指定値を変更してください。バージョンアップの注意事項については，マニュアル「HiRDB システム導入・設計ガイド」の「Java ストアドプロシジャ及び Java ストアドファンクションを使用する場合」を参照してください。

◆ pd_java_libpath = "Java 仮想マシンのライブラリディレクトリ"

～＜パス名＞

Java 仮想マシンのライブラリ（libjvm.so）が格納されているディレクトリを，Java Runtime Environment のルートディレクトリ（pd_java_runtimepath オペランドの値）への相対パス名で指定します。

《オペランドの省略値》

このオペランドの省略値を次に示します。

Java Runtime Environment のバージョンによって Java 仮想マシンのライブラリが格納されているディレクトリが省略値と異なる場合があります。省略値と異なる場合は，使用する Java Runtime Environment の Java 仮想マシンのライブラリが格納されているディレクトリを指定してください。

OS の種類	省略値
Linux	lib/i386/server
Linux (EM64T)	lib/amd64/server
AIX (64 ビットモード)	bin/classic

《オペランドの規則》

- パス名は 255 文字以内とします。

◆ pd_java_stdout_file = "Java 仮想マシンの標準・標準エラー出力の設定先ファイル"

～＜パス名＞

Java 仮想マシンでの標準出力又は標準エラー出力の出力先をファイルにする場合，そのファイル名を絶対パス名で指定します。このオペランドを省略すると，Java 仮想マシンでの標準出力又は標準エラー出力が無視されます。

《指定値の目安》

このオペランドで指定するファイルの容量が非常に大きくなるため，通常はこのオペランドを指定しないでください。Java ストアドプロシジャ又は Java ストアドファンクションのデバッグ時に指定することをお勧めします。なお，このオペランドで指定するファイルの容量制限はありません。

《注意事項》

複数プロセスからの同時書き込みがあった場合は，その出力内容を保証しません。

《オペランドの規則》

- パス名は 255 文字以内とします。

◆ **pd_java_castoff = Y ｜ N**

Java 仮想マシンを起動したサーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，及びバックエンドサーバ）プロセスを，次の契機で終了させるかどうかを指定します。

項番	サーバの種類	プロセスの名称	プロセスが終了する契機
1	シングルサーバ	pdsds	UAP を切り離したとき
2	フロントエンドサーバ	pdfes	UAP を切り離したとき
3	ディクショナリサーバ	pddic	トランザクションが決着したとき，又は UAP を切り離したとき
4	バックエンドサーバ	pdbes	トランザクションが決着したとき，又は UAP を切り離したとき

Y：表に示した契機でサーバプロセスを終了させます。

N：表に示した契機でサーバプロセスを終了させません。

《指定値の目安》

通常はこのオペランドを指定しないでください。ただし，次に示す問題が発生した場合は，このオペランドに Y を指定することをお勧めします。

- Java 仮想マシンの使用によってメモリサイズが大きくなり，システムの空きメモリが圧迫されている
- 別のコネクションで Java 仮想マシンが設定したスタックサイズの上限が原因で，Java 仮想マシンを使用しないコネクションでも探索条件が多い SQL を実行するとスタックの拡張ができなくなり，サーバプロセスがセグメンテーション障害でアボートする

なお，Java 仮想マシンの機能については Java 仮想マシンのドキュメントを参照してください。

《注意事項》

Java ストアドルーチンを頻繁に実行するシステムで，このオペランドに Y を指定すると，サーバプロセスの再起動，及び Java 仮想マシン起動のオーバヘッドが発生します。

《ほかのオペランドとの関連》

このオペランドは，pd_process_count オペランドと関連があります。

9.2.52　外部 C ストアドルーチンに関するオペランド

◆ **pd_c_library_directory = "C ライブラリファイル格納ディレクトリ"**
～＜パス名＞《$PDDIR/clib》

C ライブラリファイルを格納するディレクトリを絶対パス名で指定します。

《オペランドの規則》

- パス名は 255 文字以内とします。

《注意事項》

このオペランドを指定する場合，ここで指定するディレクトリを C ライブラリファイルの登録前に作成しておく必要があります。

9.2.53　SQL 実行時間警告出力機能に関するオペランド

◆　pd_cwaittime_wrn_pnt = SQL 実行時間警告情報の出力条件（比率指定）｜SQL 実行時間警告情報の出力条件（時間指定）《1800sec,auto》

SQL 実行時間警告出力機能を使用する場合にこのオペランドを指定します。このオペランドの指定方法は次に示す二つの方法があります。

- 比率で指定する方法
- 時間で指定する方法

SQL 実行時間警告出力機能については，マニュアル「HiRDB システム運用ガイド」を参照してください。

SQL 実行時間警告情報の出力条件（比率指定）：～＜符号なし整数＞((0～99))又は＜符号なし 10 進数＞((0～99.999999))（単位：%）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を，クライアントの最大待ち時間（クライアント環境定義の PDCWAITTIME オペランドの値）に対する比率で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル
- 警告メッセージ（KFPA20009-W）

《オペランドの指定方法》

PDCWAITTIME オペランドの値に対する比率（%単位）で指定します。例えば，PDCWAITTIME オペランドに 100（秒）を指定し，このオペランドに 90（%）を指定すると，SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間が 90 秒以上 100 秒未満の場合に警告情報が出力されます。

（例）

PDCWAITTIME = 100

pd_cwaittime_wrn_pnt = 90

SQL 実行時間警告情報の出力条件（時間指定）：～＜符号なし 10 進数＞((0～65534.999999))sec 又は＜符号なし整数＞((0～65534))sec,auto（単位：秒）

SQL 実行時間警告出力機能の SQL 実行時間警告情報を出力する条件を時間で指定します。SQL の実行後に HiRDB が SQL の実行時間を調べます。その結果，SQL の実行時間がこのオペランドの指定値によって設定される時間以上になった場合，次に示す警告情報を出力します。これを SQL 実行時間警告出力機能といいます。

- SQL 実行時間警告情報ファイル

- 警告メッセージ（KFPA20009-W）

auto を指定した場合，PDCWAITTIME を省略又は 0 を指定した場合も，その SQL に対して警告情報を出力します。

《オペランドの指定方法》

出力契機とする時間を秒単位で指定します（小数点以下 6 けたまで指定できます）。指定値に「**sec**」を付けてください。

（例）

```
pd_cwaittime_wrn_pnt = 0.001sec
```

これ以降の説明は，比率指定及び時間指定の両方共通の説明です。

《オペランドの規則》

このオペランドに 0，0sec，又は 0sec,auto を指定すると，警告情報を出力しません（SQL 実行時間警告出力機能を使用しません）。

《クライアント環境定義との関連》

このオペランドの値はクライアントごとに変更できます。クライアントごとに変更する場合は，クライアント環境定義の PDCWAITTIMEWRNPNT オペランドを指定してください。
PDCWAITTIMEWRNPNT オペランドについては，マニュアル「HiRDB UAP 開発ガイド」を参照してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_cwaittime_report_dir
- pd_cwaittime_report_size

◆ pd_cwaittime_report_dir = SQL 実行時間警告情報ファイルの出力先ディレクトリ ～＜パス名＞《$PDDIR/spool/pdcwwrn》

SQL 実行時間警告情報ファイルを出力する場合に，出力先のディレクトリを絶対パス名で指定します。ここで指定したディレクトリ下に SQL 実行時間警告情報ファイルが二つ（pdcwwrn1 及び pdcwwrn2）作成されます。

HiRDB/パラレルサーバの場合，警告対象となった SQL を発行する UAP が接続するフロントエンドサーバがあるサーバマシンに SQL 実行時間警告情報ファイルを出力します。

《オペランドの規則》

- パス名は 255 文字以内とします。

◆ pd_cwaittime_report_size = SQL 実行時間警告情報ファイルの最大容量 ～＜符号なし整数＞((2048～2147473627))《10000000》（単位：バイト）

SQL 実行時間警告情報ファイルの最大容量を指定します。このオペランドに指定した値は SQL 実行時間警告情報ファイル一つ分の容量です。SQL 実行時間警告情報ファイルは二つ作成されるため，指定値に注意してください。例えば，10,000 を指定すると，ディレクトリ下に最大 10,000 バイトのファイルが二つ作成されます。

《指定値の目安》

次に示す計算式を目安にこのオペランドの指定値を決めてください。

{1280 ＋ SQL 文のサイズ（バイト）} ×ファイルに格納する警告情報数

SQL 文に注釈（コメント）や SQL の最適化を指定している場合は，注釈及び SQL の最適化の指定サイズ（バイト）も，SQL 文のサイズに含めてください。

《備考》

- SQL 実行時間警告情報ファイルへの出力容量がこのオペランドの値を超えると，出力先をもう片方のファイルに切り替えます。これを繰り返して二つのファイルを交互に使用します。このとき，切り替え先の古い情報は削除されます。
- 1 回で出力する SQL 実行時間警告情報の大きさがファイル容量を超えた場合，SQL 実行時間警告情報をすべて出力しません。ファイル容量までの SQL 実行時間警告情報を出力します。この場合，SQL 実行時間警告情報の最後に#を付けます。

9.2.54　文字コードに関するオペランド

◆ **pd_substr_length = 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10**

1 文字を表現する最大バイト数を指定します。

セットアップした文字コード種別によって，推奨値と指定できる値の範囲が異なります。セットアップした文字コード種別と，対応する推奨値，指定できる値を次に示します。

セットアップした文字コード種別	推奨値	指定できる値
utf-8	4	3〜6
utf-8_ivs	8	3〜10
上記以外（このオペランドは無効）	−	−

（凡例）

−：このオペランドは無効となるため値はありません。

このオペランドは，文字コード種別を utf-8，又は utf-8_ivs でセットアップした場合に有効で，スカラ関数 SUBSTR の結果の長さに影響します。

スカラ関数 SUBSTR については，マニュアル「HiRDB SQL リファレンス」を参照してください。

《指定値の目安》

UCS-2 の範囲の文字（1〜3 バイト）だけを使用する場合は，このオペランドを省略できます。

UCS-4 の範囲の文字を使用する場合は，1 文字を表現する最大バイト数を指定します。ISO/IEC 10646 の規格では文字が割り当てられているのは，UTF-8 の場合 1 文字 1〜4 バイト，IVS 対応 UTF-8 の場合 1 文字 1〜4，7，又は 8 バイトです。指定値はそれぞれの最大バイト数を推奨します。

ただし，実際に使用している 1 文字を表現するバイト数以上の値を指定すると，スカラ関数 SUBSTR の結果が長くなることがあるため，使用している文字コードに基づいて指定値を検討してください。

《注意事項》

- ISO/IEC 10646 の規格は，1 文字当たり 1〜4 バイト（IVS 対応 UTF-8 では 1〜4，7，又は 8 バイト）の範囲に文字が割り当てられ，5，6 バイト（IVS 対応 UTF-8 では 5，6，9，又は 10 バイト）の範囲は将来の規格のために予約されています。HiRDB では 1 文字当たり 1〜6 バイト（IVS 対応 UTF-8 では 1〜10 バイト）の範囲まで使用できますが，文字が割り当てられていない 5，6 バイト（IVS 対応 UTF-8 では 5，6，9，又は 10 バイト）の範囲を使用する場合，将来発生するおそれのある問題については保証できません。
- バージョン 06-02 でサポートした Unicode（UTF-8）は，1 文字を表現する最大バイト数は 3 バイトの範囲まで使用できます。バージョン 08-00 からは，最大 6 バイトの範囲まで使用できます。バージョン 09-50 でサポートした文字コード種別 utf-8_ivs は，1 文字を表現する最大バイト数が 10 バイトの範囲まで使用できます。そのため，データによっては 1 文字を表現するバイト数が大きくなることがあります。
- 次の条件をすべて満たしている場合は，ルーチンの SQL オブジェクトの再作成やビュー表の再定義をする必要があります。

  ・文字コード種別を utf-8，又は utf-8_ivs でセットアップしている

  ・新規で pd_substr_length オペランドの指定値に 3 以外を指定する，又は既に pd_substr_length オペランドを指定している環境で，ルーチンやビュー表を定義し，かつ pd_substr_length オペランドの指定値を変更する

  ・混在文字列型（MCHAR，MVARCHAR）を引数に指定したスカラ関数 SUBSTR をルーチンやビュー定義に指定する

  ただし，クライアント環境定義の PDSUBSTRLEN オペランドを指定してコネクション単位に制御したり，SQL コンパイルオプションの SUBSTR LENGTH の指定によってルーチン単位で制御したりもできます。
- 文字コード種別を utf-8，又は utf-8_ivs でセットアップしている場合の pd_substr_length オペランドと，そのほかのオペランドの指定値との優先順位を次に示します。

  ルーチン単位で制御する場合

制御のタイミング	pd_substr_length オペランド	クライアント環境定義 PDSUBSTRLEN	SQL コンパイルオプション SUBSTR LENGTH	備考
定義時	有効（2）	無効	有効（1）	省略値 pd_substr_length
コール時又は関数呼出し時	無効（定義時の指定に従う）	無効（定義時の指定に従う）	−	特になし

（凡例）

（）：優先順位

−：指定できません。

ビュー表単位で制御する場合

制御のタイミング	pd_substr_length オペランド	クライアント環境定義 PDSUBSTRLEN	SQL コンパイルオプション SUBSTR LENGTH	備考
定義時	有効（2）	有効（1）	−	省略値 pd_substr_length
操作時	無効 （定義時の指定に従う）	無効 （定義時の指定に従う）	−	特になし

（凡例）

（）：優先順位

−：指定できません。

9.2.55　ワークファイル出力先ディレクトリの変更

◆ **pd_tmp_directory = ワークファイル出力先ディレクトリ名**

～＜絶対パス名＞((1～512))

HiRDB が出力するワークファイルの出力先ディレクトリ名を指定します。指定するディレクトリは HiRDB を起動する前に作成し，全ユーザに対してアクセス権限を付与しておいてください。

HiRDB は，各コマンドで決められた優先順位に従ってワークファイル出力先ディレクトリを決定します。出力先を決定する上でこのオペランドの指定値が有効となる出力ファイルについては，マニュアル「HiRDB コマンドリファレンス」を参照してください。

指定したディレクトリに格納されたワークファイルは，pdcspool コマンドで定期的に削除できます。pdcspool コマンドについては，マニュアル「HiRDB コマンドリファレンス」を参照してください。

《前提条件》

このオペランドに指定するディレクトリは，HiRDB を起動する前に作成してください。また，ディレクトリのアクセス権限に「777」を指定してください。

HiRDB 停止中に誤ってディレクトリを削除するなどして，HiRDB の起動時にこのオペランドに指定した名称のディレクトリが存在しなくなった場合は，HiRDB が指定した名称のディレクトリを作成し，アクセス権限に「777」を付与します。

《利点》

- HiRDB の運用ディレクトリ以外に HiRDB が出力するワークファイルの出力先を 1 か所に統一できます。
- pdcspool コマンドと併用することで，コマンド実行後に残るワークファイルを定期的に削除できます。

これらの効果によって，ディスク管理が容易になります。

《注意事項》

- このオペランドには，ルートディレクトリ（/）は指定できません。

- HiRDB 運用ディレクトリ下のディレクトリは指定しないでください。
- HiRDB 以外のプログラムで使用しているディレクトリ名を指定しないでください。HiRDB 以外のプログラムで使用するディレクトリ名を指定すると，pdcspool コマンドを実行した際に，HiRDB 以外のプログラムが出力したファイルが削除されることがあります。

  ただし，次の名称を指定した場合には，pdcspool コマンドを実行してもファイルが削除されないため，ユーザが手動でファイルを削除する必要があります。

  ・/tmp

  ・/usr/tmp

  ・/var/tmp

  ・/
- 指定したパス名が正しくない，又は，指定したディレクトリにユーザのアクセス権限がない場合，HiRDB はこのオペランドで指定したディレクトリにアクセスできません。このとき，警告メッセージが出力され，このオペランドの指定値に/tmp が仮定されます。
- このオペランドを指定しなかった場合，環境変数 TMPDIR に値が正しく設定されていれば，このオペランドの指定値に環境変数 TMPDIR の設定値が仮定されます。このとき，環境変数 TMPDIR に設定するパス名長は 512 バイト以内に設定してください。
- 環境変数 TMPDIR に値が設定されていないとき，又は環境変数の設定値が不正のときは，このオペランドの指定値に/tmp が仮定されます。

9.2.56　HiRDB ファイルシステム領域に関するオペランド

◆ pd_duplicated_fs_suffix = "複製ディスク機能使用時のセカンダリパスのサフィックス" ～< 1～3 文字のパス名(スラッシュ"/"を除く)>

複製ディスク機能使用時の，セカンダリパスに付加するサフィックスを指定します。

pdfmkfs コマンドに-m オプション及びプライマリパスを指定して実行することで，このオペランドに指定したサフィックスを付加したセカンダリパスにも，複製ディスク機能用の HiRDB ファイルシステム領域を同時に作成します。

（例）

このオペランドに"_R"を指定します。

```
set pd_duplicated_fs_suffix = ”_R”
```

pdfmkfs コマンドに-m を指定して実行します。

```
pdfmkfs -n 4096 -l 10 -e 230 -k DB -m “/hirdb/RDAREA”
```

次の 2 つのパスに同一の HiRDB ファイルシステム領域が作成されます。

- プライマリパス：“/hirdb/RDAREA”
- セカンダリパス：“/hirdb/RDAREA_R”

このオペランドを指定し，かつ複製ディスク機能用の HiRDB ファイルシステム領域を作成した場合に，複製ディスク機能が適用されます。複製ディスク機能については，マニュアル「HiRDB システム運用ガイド」の「複製ディスク機能」を参照してください。

《注意事項》

- このオペランドは Linux 限定です。Linux 以外で指定した場合，定義エラーになります。
- このオペランドを指定した場合，システム定義の pd_log_dual_write_method オペランドに parallel を指定しても，システムログファイルの並列出力機能は無効になります。
- このオペランドを指定する場合，システム定義に次のオペランドを指定してください。

  pd_ha_prc_cleanup_check=Y

  pd_redo_allpage_put=Y
- 指定値を変更する場合，セカンダリパスのパス名も変更してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_log_dual_write_method
- pd_ha_prc_cleanup_check
- pd_redo_allpage_put

◆ **pd_max_utl_ios_file_no＝pdfmkfs -k UTL -m コマンドで作成した HiRDB ファイルシステム領域数～＜符号なし整数＞((0～32768))《0》**

pdfmkfs -k UTL -m コマンド（複製ディスクの指定，かつ使用目的に UTL を指定）で作成した HiRDB ファイルシステム領域の数を指定します。

pd_duplicated_fs_suffix オペランドを省略した場合，このオペランドを指定しても，0 を指定したとみなします。

《指定値の目安》

- 複製ディスクの指定，かつ使用目的に UTL を指定した HiRDB ファイルシステム領域を使用する場合，その HiRDB ファイルシステム領域数以上の値を指定してください。

  HiRDB/パラレルサーバの場合，このオペランドの値はユニットごとに適用されます。ユニットごとに使用する数を指定してください。システム共通定義に記述する場合は，ユニットごとに使用する数の最大値を指定します。

《注意事項》

- このオペランドは Linux 限定です。Linux 以外で指定した場合，定義エラーになります。

《各見積もり式への影響》

pd_max_utl_ios_file_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_duplicated_fs_suffix

9.2.57　リードレプリカ機能に関するオペランド

◆ **pd_read_replica_use = Y ｜ N**

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。

HiRDB システムで参照 DB を使用するかどうかを指定します。

Y：

参照 DB を使用します。

N：

参照 DB を使用しません。

《指定値の目安》

リードレプリカを構成したシステムの内,参照 DB として動作する場合は Y を指定します。なお，このオペランドを省略した場合は，通常のユニット（リードレプリカ構成では更新 DB）として動作します。

《注意事項》

このオペランドに Y を指定した場合，一部の他機能では使用に注意が必要な機能，又は使用できない機能があります。詳細は，マニュアル「HiRDB システム運用ガイド」の「他機能との連携」を参照してください。

このオペランドに Y を指定する場合，pd_indexlock_mode オペランドに NONE を指定するか省略してください。pd_indexlock_mode オペランドに KEY を指定した場合，表にインデクスを定義していると pdrrpref コマンドによる反映がエラーになります。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_read_replica_workdir
- pd_indexlock_mode

◆ pd_read_replica_mode = unld ｜ sys

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。

参照 DB 側のシステム共通定義に，更新ログを反映する方式を指定します。

unld：

アンロードログファイルを使用して手動で反映する方式です（手動反映方式）。

sys：

システムログファイルを使用して自動で反映する方式です（自動反映方式）。

《前提条件》

このオペランドは pd_read_replica_use オペランドに Y を指定した場合に有効になります。

《注意事項》

バージョン 10-09，10-10 の HiRDB でリードレプリカ機能を使用していて，バージョン 10-11 以降でリードレプリカ機能を引き続き使用する場合は，pd_read_replica_mode オペランドのデフォルト値が変わります。

手動反映方式を引き続き使用する場合は，pd_read_replica_mode オペランドに unld を指定してください。

このオペランドに sys を指定する，又は指定を省略してリードレプリカの自動反映方式を採用する場合，pd_log_block_check オペランドに Y を指定して，ブロックチェック処理を有効にしてください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_read_replica_use
- pd_log_block_check

◆ pd_read_replica_workdir = 作業用ディレクトリ ～<パス名>((1～255 文字))

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。

リードレプリカ機能で使用する一時ファイルを格納するディレクトリを指定します。

このオペランドを省略した場合に仮定する値を次の表に示します。なお，ディレクトリ容量が下記の仮定値では小さい場合，このオペランドを指定します。

システム定義の pd_tmp_directory オペランドの指定	環境変数 TMPDIR の指定※	仮定値（一時ファイルの出力先）
あり	−	pd_tmp_directory オペランドに指定したディレクトリ
なし	あり	環境変数 TMPDIR に指定したディレクトリ
	なし	/tmp ディレクトリ

注※

システム共通定義（pdsys）に putenv 形式で指定します。

《指定値の目安》

リードレプリカ機能で使用する一時ファイルの総容量の最大は，リードレプリカ機能で指定するアンロードログファイルの合計サイズと同等の容量が必要になります。このオペランドの省略時に仮定されるディレクトリに十分な空き容量がない場合，このオペランドに空き容量のあるディレクトリを指定します。

《注意事項》

このオペランドを指定した場合，HiRDB のトラブルシュート情報の削除機能の対象外になります。参照 DB が強制終了や異常終了などで反映処理が中断した場合，一時ファイルが残ることがあるため，手動で削除してください。

《ほかのオペランドとの関連》

このオペランドは pd_read_replica_use オペランドと関連があります。

◆ pd_read_replica_max_trn_count = トランザクション数 ～<符号なし整数>((1～32768))《100》

このオペランドは，HiRDB/シングルサーバ限定のオペランドです。

pd_read_replica_use オペランドに Y 及び pd_read_replica_mode オペランドに sys を指定した場合，更新 DB の更新トランザクションを参照 DB へ反映する際に，指定されたトランザクション数を反映するごとに参照 DB で COMMIT を発行します。

《指定値の目安》

このオペランドの指定値を大きくすると，HiRDB の資源を多く使用します。排他資源不足やログ満杯にならない程度に指定してください。

《注意事項》

- このオペランドに 1 以外を指定する場合，参照 DB の一つの反映トランザクションが確保する排他資源数及び出力するシステムログ量が増加します。排他資源数の見積もりは更新 DB で一つのトランザクションが確保する排他資源数×このオペランド指定値で見積もってください。システムログファイルの見積もりは更新 DB の 1 トランザクションが出力するシステムログ量×このオペランド指定値で見積もってください。
- トランザクション数がこのオペランドに達する前に commit を発行することがあります。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_read_replica_use
- pd_read_replica_mode
- pd_lck_pool_size
- pd_max_access_tables

10 UAP 環境定義

この章では，UAP 環境定義の各オペランドの内容について説明します。

10.1　オペランドの形式

UAP 環境定義では UAP の実行環境を定義します。ここでは，UAP 環境定義の各オペランドの形式について説明します。

形式	強制，異常終了後の変更可否	計画停止後の変更可否
〔set pd_uap_wait = Y ｜ N〕	○	○
〔{{pdlbuffer -a ローカルバッファ名 {-r RD エリア名〔，RD エリア名〕… ｜-i 認可識別子．インデクス識別子} -n バッファ面数 〔-p 一括入力最大ページ数〕}}〕	○	○
〔putenv PDDBACCS インナレプリカ RD エリア世代番号〕	○	○

（凡例）

○：指定値を変更できます。

10.2　オペランドの説明

◆ **pd_uap_wait = Y ｜ N**

ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。

Y：使用中のユーザの終了を待ちます。UAP は WAIT 状態になります。

N：ほかのユーザが使用中の場合は UAP をエラーリターンします。

二つ以上の pdlbuffer オペランドと一緒にこのオペランドを指定した場合，そのすべてのローカルバッファに対してこのオペランドの指定が有効になります。

《前提条件》

pdlbuffer オペランドを指定する必要があります。

◆ **pdlbuffer -a ローカルバッファ名**

{-r RD エリア名〔，RD エリア名〕...｜-i 認可識別子．インデクス識別子}

-n バッファ面数

〔-p 一括入力最大ページ数〕

UAP が使用するローカルバッファを定義します。ローカルバッファとはディスク上の表又はインデクスなどのデータを入出力するときに使用する領域のことで，シングルサーバ又はバックエンドサーバのプロセス固有メモリ上に確保されます。なお，ローカルバッファは 100 個まで定義できます。HiRDB/パラレルサーバの場合は 1 バックエンドサーバ当たり 100 個まで定義できます。

また，ローカルバッファを指定するときは，pdlbuffer オペランドのオプションの指定によって次のように排他が掛かります。

pdlbuffer オペランドの指定	排他資源	排他モード
pdlbuffer オペランドの-r オプションを指定した場合	指定された RD エリア	EX
pdlbuffer オペランドの-i オプションを指定した場合	インデクス格納 RD エリア	SU
	インデクス格納表	EX
	指定されたインデクス	EX

《前提条件》

ローカルバッファに指定する RD エリア及びインデクスに対して，pdbuffer オペランドでグローバルバッファを割り当てておく必要があります。

《適用基準》

次に示す条件をすべて満たす場合にローカルバッファを定義します。

- 大量のデータを検索又は更新する
- アクセス対象の RD エリアがほかの UAP からアクセスされない

また，HiRDB に常時接続する UAP はシステムへの影響（メモリの圧迫，プロセスの占有など）が大きいため，ローカルバッファを定義しないでください。

《注意事項》

HiRDB/パラレルサーバの場合の注意事項を次に示します。

- ローカルバッファの使用時にサーバプロセスが異常終了すると，アボートコード Phb3008 を出力してユニットが異常終了することがあります。
- サーバプロセスの異常終了時に更新ページがあると，ロールバックプロセスで回復処理ができないことがあります。その場合，ユニットの再開始時に回復処理を実行します。
- ローカルバッファを使用している場合に障害が発生したときの HiRDB の処理と対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

-a ローカルバッファ名

～＜識別子＞((1～16 文字))

ローカルバッファの名称を指定します。ほかのローカルバッファと同じ名称を指定できません。ただし，UAP 環境定義ファイルが異なる場合は同じ名称を指定できます。

-r RD エリア名〔，RD エリア名〕...

～＜識別子＞((1～30 文字))

データ用のローカルバッファを割り当てる RD エリアの名称を指定します。ユーザ用 RD エリアの名称だけ指定できます。

同じ名称は指定できません。ただし，UAP 環境定義ファイルが異なる場合は同じ名称を指定できます。

《指定値の目安》

一つのローカルバッファに複数の RD エリアを割り当てる場合は，RD エリアの最大ページ長がバッファ長になります。したがって，ページ長が同じか，又はページ長が近い RD エリアを同じローカルバッファに割り当てると，入出力回数を削減できます。ただし，同じページ長の RD エリアでも，次に示すような RD エリアは別々のローカルバッファに割り当てた方が入出力回数を削減できます。

- 用途の異なる表を格納する RD エリア
- ランダムアクセスが多い RD エリアと順次アクセスが多い RD エリア

《オペランドの規則》

- RD エリア名に半角英数字以外の文字を含む場合は引用符（"）で囲んでください。
- 一つのローカルバッファには最大 4096 個の RD エリアを定義できます。

-i 認可識別子．インデクス識別子

認可識別子：～＜識別子＞((1～30 文字))

インデクス識別子：～＜識別子＞((1～30 文字))

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

同じ名称は指定できません。ただし，UAP 環境定義ファイルが異なる場合は同じ名称を指定できます。

《オペランドの規則》

認可識別子又はインデクス識別子に半角英数字以外の文字を含む場合は，認可識別子及びインデクス識別子をそれぞれ引用符（"）で囲んでください。

《指定値の目安》

使用頻度の高いインデクスを指定してください。使用頻度の高いインデクスにローカルバッファを割り当てると，インデクスページのメモリ常駐度が上がり，入出力回数の削減効果が出ます。

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

《注意事項》

一時インデクス名を指定したインデクス用ローカルバッファは無効となります。

-n バッファ面数

～＜符号なし整数＞((4～125000))

ローカルバッファの面数を指定します。

《指定値の目安》

- ローカルバッファはプロセス固有メモリ上に確保されます。必要以上に大きくするとほかのメモリの使用時にページングが多発し，性能が低下します。
- バッファ面数が大き過ぎるとプロセス固有メモリを確保できないことがあるので注意してください。
- シーケンシャルアクセスをする UAP の場合は，バッファリングの効果が望めないため，バッファ面数に最小値 4 を指定してください。

-p 一括入力最大ページ数

～＜符号なし整数＞((2～256))

プリフェッチ機能の一括入力最大ページ数を指定します。このオペランドを省略するとプリフェッチ機能は動作しません。

《指定値の目安》

メモリ量と入出力時間の削減効果のコストパフォーマンスから，次に示す計算式を満足する値を指定してください。

a × b ＝ 64～128　（単位：キロバイト）

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

b：一括入力最大ページ数

なお，RD エリアのセグメントサイズが１の場合は，このオプションを指定しても効果が得られないため指定しないでください。

◆ PDDBACCS インナレプリカ RD エリア世代番号

～＜符号なし整数＞((0～10))

クライアント環境変数 PDDBACCS が省略された場合の仮定値として設定します。

インナレプリカ機能を使用している場合，インナレプリカグループ内でカレント RD エリアでない RD エリアをアクセスしたいときに，その RD エリアの世代番号を指定します。省略した場合は，カレント RD エリアをアクセスします。

インナレプリカ機能については，マニュアル「インナレプリカ機能 HiRDB Staticizer Option」を参照してください。

《注意事項》

このオペランドの指定は，システム内に定義されているすべてのインナレプリカグループに対して適用されます。このオペランドで指定した世代のレプリカ RD エリアが定義されていない場合，該当するインナレプリカグループ内のカレント RD エリアが処理対象となります。このため，レプリカ RD エリアを使用するテスト環境を設定する場合，誤って本番用の RD エリアをアクセスしないように，アクセスするすべての RD エリアについて，指定する世代のレプリカ RD エリアが定義されているか確認してください。

11 互換モード適用時に検討が必要なオペランド

この章では，互換モードを適用する場合に，検討が必要なオペランドの内容について説明します。
推奨モードを適用している場合は，この章に説明されているオペランドは省略してください。

11.1　オペランドの一覧

互換モードでは，旧バージョンとの互換性を保つために，推奨値と省略値が異なるオペランドがあります。互換モードを適用する場合は，これらのオペランドについて，推奨モードでの省略値を確認の上，推奨値を明示的に指定することを検討してください。なお，推奨モードを適用している場合，この章で説明するオペランドは省略してください。

互換モードを適用する場合に，検討が必要なオペランドの名称，及び各オペランドの変更可否の一覧を次の表に示します。オペランド名はアルファベット順に記載しています。

表 11–1　互換モードを適用する場合に，検討が必要なオペランドの一覧

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否	検討要否
	SYS	UNT	SVR	SDS	FES	DS	BES			
pd_apply_search_ats_num	○							○	○	0904
pd_dbsync_altwrite_skip	○							○	○	0904
pd_hashjoin_hashing_mode	○							○	○	0904
pd_inet_unix_bufmode	○	○						○	○	0904
pd_inet_unix_utl_bufmode	○	○						○	○	0904
pd_ipc_inet_bufsize	○	○						○	○	0904
pd_ipc_unix_bufsize	○	○						○	○	0904
pd_leap_second	○							○	○	0904
pd_listen_socket_bufset	○	○						○	○	0904
pd_log_max_data_size※3			○	○	○	○	○	○※1，※2	○※1	0904
pd_log_rec_leng			○	○	○	○	○	○	○	0904

オペランド名	定義名							強制終了，異常終了後の変更可否	計画停止後の変更可否	検討要否
	SYS	UNT	SVR	SDS	FES	DS	BES			
pd_log_rollback_buff_count			○	○	○	○	○	○※2	○	0904
pd_max_tmp_table_rdarea_no	○							○	○	0904
pd_registered_port_level	○	○						×	○	0904
pd_sql_simple_comment_use	○							○	○	0904
pd_sts_initial_error			○	○	○	○	○	○	○	0904
pd_sts_singleoperation			○	○	○	○	○	○	○	0904
pd_syssts_initial_error		○						○	○	0904
pd_syssts_singleoperation		○						○	○	0904
pd_tcp_inet_bufsize	○	○						○	○	0904
pd_tcp_unix_bufsize	○	○						○	○	0904
pd_trn_rollback_msg_interval	○							○	○	0904
pd_utl_buff_size	○							○※1	○※1	0904
pd_utl_exec_mode	○							×	×	0904

（凡例）

○：指定値を変更できます。

×：指定値を変更できません。

空白：該当しません。

SYS：システム共通定義

UNT：ユニット制御情報定義

SVR：サーバ共通定義

SDS：シングルサーバ定義

FES：フロントエンドサーバ定義

DS：ディクショナリサーバ定義

BES：バックエンドサーバ定義

0904：0904 互換モード適用時に検討が必要なオペランド

注※1

オペランドの指定値を小さくすると，HiRDB を再開始できない場合があります。その場合は，指定値を変更前の値に戻して再開始してください。

注※2

オペランドの指定値を大きくすると，HiRDB を再開始できない場合があります。その場合は，指定値を変更前の値に戻して再開始してください。

注※3

HiRDB Datareplicator（抽出側）と連携している場合に注意が必要なオペランドです。このオペランドの指定を追加，変更，又は削除する場合は，対応する HiRDB Datareplicator を一度停止してから再開始してください。HiRDB Datareplicator を稼働したまま，この定義を変更して HiRDB を再度開始すると，HiRDB Datareplicator による抽出が失敗することがあります。

11.2　オペランドの形式

11.2.1　オペランドの形式の説明

互換モードを適用する場合に，検討が必要な各オペランドの形式について説明します。

(1)　同時実行最大数

〔set pd_utl_exec_mode = 0 ｜ 1〕

(2)　HiRDB の処理方式

〔set pd_dbsync_altwrite_skip = Y ｜ N ｜ REF ｜ ALL〕

(3)　トランザクション決着処理

〔set pd_trn_rollback_msg_interval = ロールバックが進行中であることを示すメッセージの出力間隔〕

(4)　SQL の仕様

〔set pd_sql_simple_comment_use = Y ｜ N〕

(5)　SQL の最適化

〔set pd_hashjoin_hashing_mode = TYPE1 ｜ TYPE2〕

〔set pd_apply_search_ats_num = サーチ条件 ATS を適用する絞り込み値の組み合わせ個数の上限〕

(6)　一時表

〔set pd_max_tmp_table_rdarea_no = 一時表用 RD エリアの最大数〕

(7)　通信処理

〔set pd_registered_port_level = 0 ｜ 1〕

〔set pd_inet_unix_bufmode = os ｜ conf〕

〔set pd_ipc_inet_bufsize = サーバのユニット間通信で使用する送受信バッファ長〕

〔set pd_ipc_unix_bufsize = サーバのユニット内通信で使用する送受信バッファ長〕

〔set pd_tcp_inet_bufsize = HiRDB サーバが存在するホスト外の HiRDB クライアントとの通信で使用する送受信バッファ長〕

〔set pd_tcp_unix_bufsize = HiRDB サーバが存在するホスト内の HiRDB クライアントとの通信で使用する送受信バッファ長〕

〔set pd_listen_socket_bufset = 0 | 1〕

〔set pd_utl_buff_size = ユティリティの通信用バッファ長〕

〔set pd_inet_unix_utl_bufmode = auto | conf〕

(8) 日付・時刻

〔set pd_leap_second = Y | N〕

(9) ユニット用ステータスファイル（障害発生時）

〔set pd_syssts_initial_error = stop | continue | excontinue〕

〔set pd_syssts_singleoperation = stop | continue〕

(10) サーバ用ステータスファイル（障害発生時）

〔set pd_sts_initial_error = stop | continue | excontinue〕

〔set pd_sts_singleoperation = stop | continue〕

(11) システムログファイル

〔set pd_log_max_data_size = ログ入出力バッファ長〕

〔set pd_log_rec_leng = システムログファイルのレコード長〕

〔set pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数〕

11.3　オペランドの説明

11.3.1　同時実行最大数に関するオペランド

◆ **pd_utl_exec_mode = 0 ｜ 1**

《オペランドの省略値》

- 推奨モード適用時の省略値：1
- 0904 互換モード適用時の省略値：0

ユティリティの最大同時実行数を拡張するかどうかを指定します。

0：

ユティリティの最大同時実行数を拡張しません。この場合，ユティリティの最大同時実行数は，HiRDB が決定します。

1：

ユティリティの最大同時実行数を拡張します。この場合，pd_max_users オペランドの値に応じてユティリティの最大同時実行数が拡張されます。

このオペランドの指定値によって決まるユティリティの最大同時実行数については，マニュアル「HiRDB コマンドリファレンス」の「ユティリティの最大同時実行数」を参照してください。

《指定値の目安》

次に示す場合は 1 を指定してください。

- 0 を指定したときの最大同時実行数では足りない場合
- 次に示す計算式を満たす場合

  全バックエンドサーバ数（HiRDB/シングルサーバの場合は 1）×2×全ユティリティの同時実行数≧824

また，次に示す条件式をすべて満たす場合は，このオペランドに 1 を指定すると共用メモリの使用量を少なくできます。

- pd_max_users の値＜ 32
- pd_max_users の値＞ユティリティの最大同時実行数

1 を指定した場合，ユティリティ実行時に必要な共用メモリはユティリティ実行時に動的に確保されます。

《各見積もり式への影響》

pd_utl_exec_mode オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「メモリ所要量の計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

11.3.2 HiRDB の処理方式に関するオペランド

◆ **pd_dbsync_altwrite_skip = Y ｜ N ｜ REF ｜ ALL**

《オペランドの省略値》

- 推奨モード適用時の省略値：Y
- 0904 互換モード適用時の省略値：N

シンクポイント取得処理中に更新バッファに対して参照要求及び更新要求が発生した場合，その更新バッファの内容をデータベースに書き込む処理は，参照要求及び更新要求をしたトランザクションを実行するサーバプロセスが代行します。この代行処理をスキップするかどうかをこのオペランドで指定します。

Y 又は REF：

シンクポイント取得処理中のグローバルバッファ参照要求ヒット時のデータベース書き込み処理をスキップします。シンクポイント出力対象バッファに参照要求がヒットした場合，トランザクションプロセスでの代行出力を行いません。

ALL：

シンクポイント取得処理中のグローバルバッファ参照要求ヒット時，及びグローバルバッファ更新要求ヒット時のデータベース書き込み処理をスキップします。シンクポイント出力対象バッファに参照要求，及び更新要求がヒットした場合，トランザクションプロセスでの代行出力を行いません。そのため，Y 又は REF を指定したときよりシンクポイント処理中の更新トランザクションの性能は安定します。

N：

シンクポイント取得処理中のグローバルバッファ参照要求ヒット時，及びグローバルバッファ更新要求ヒット時のデータベース書き込み処理代行をスキップしません。シンクポイント出力対象バッファに参照要求及び更新要求がヒットした場合，トランザクションプロセスでの代行出力を行います。

このオペランドの指定値の比較を次に示します。

pd_dbsync_altwrite_skip オペランドの値	更新バッファに対する要求	
	参照要求	更新要求
Y 又は REF	○	−
ALL	○	○

pd_dbsync_altwrite_skip オペランドの値	更新バッファに対する要求	
	参照要求	更新要求
N	−	−

（凡例）

◯：サーバプロセスは要求の代行処理をスキップします。

−：サーバプロセスは要求の代行処理をスキップしません。

《指定値の目安》

シンクポイント取得処理中にトランザクション性能が安定しない場合に指定します。シンクポイント時のデータベース書き込み処理代行が発生しているかどうかは，統計解析ユティリティのグローバルバッファプールに関する統計情報を実行し，シンクポイント時の参照要求ヒットによるデータベース書き込み処理代行回数（ALTRW），及び更新要求ヒットによるデータベース書き込み処理代行回数（ALTUW）を確認してください。この値が大きいと，シンクポイント取得処理中の参照トランザクション，及び更新トランザクションの性能が安定しません。

pd_dbsync_altwrite_skip オペランドの値	指定値の目安
Y 又は REF	参照トランザクションの性能とシンクポイントの取得時間を安定させる場合に指定します。
ALL	参照トランザクションと更新トランザクションの性能を安定させる場合に指定します。
N	シンクポイントの取得時間を短くしたい場合に指定します。

《注意事項》

このオペランドに ALL を指定すると，更新トランザクションの性能が安定し，更新処理量が増加することがあります。この場合，シンクポイント間の出力ログ量が増えるおそれがあるため，システムログファイル容量の再見積もりをしてください。

《ほかのオペランドとの関連》

このオペランドは，pd_dfw_awt_process オペランドと関連があります。

デファードライト処理の並列 WRITE 機能を使用し，シンクポイントの書き込み処理に掛かる負荷を分散させてください。

11.3.3　トランザクション決着処理に関するオペランド

◆ pd_trn_rollback_msg_interval = ロールバックが進行中であることを示すメッセージの出力間隔～<符号なし整数>((0～1440))（単位：分）

《オペランドの省略値》

- 推奨モード適用時の省略値：5
- 0904 互換モード適用時の省略値：60

トランザクションのロールバック処理が進行中であることを示す KFPS02235-I メッセージの出力間隔を指定します。0 を指定した場合，KFPS02235-I メッセージを出力しません。

ロールバック処理が進行中であることを確認する方法は，マニュアル「HiRDB システム運用ガイド」を参照してください。

11.3.4　SQL の仕様に関するオペランド

◆ **pd_sql_simple_comment_use = Y ｜ N**

《オペランドの省略値》

- 推奨モード適用時の省略値：Y
- 0904 互換モード適用時の省略値：N

SQL 文中に単純注釈を使用するかどうかを指定します。

Y：単純注釈を使用します。

N：単純注釈を使用しません。

《注意事項》

- このオペランドに Y を指定した場合，減算のマイナスと単項演算子のマイナスを括弧なしで連続して使用した演算が指定できなくなります。
- ルーチンの定義，及びルーチンのリコンパイル時に，単純注釈を使用するかどうかは，このオペランドの値によって決定されます。
- ルーチンの定義と，そのルーチンのリコンパイル時はこのオペランドの値を同じにしてください。このオペランドを変更し，ルーチンのリコンパイルを行うと，次に示す現象が起こる場合があります。

  ・リコンパイル時に構文エラーとなる

  ・リコンパイル後のルーチンを実行した場合，リコンパイル前と実行結果が変わる
- このオペランドの値によって，データベース定義ユティリティ(pddef)に入力する SQL 中の制御文字の扱いや，ディクショナリ搬出入ユティリティ(pdexp)，定義系 SQL の生成コマンド(pddefrev)が出力する SQL 中の改行コードの有無が変わります。詳細は，マニュアル「HiRDB コマンドリファレンス」を参照してください。

11.3.5　SQL の最適化に関するオペランド

◆ **pd_hashjoin_hashing_mode = TYPE1 ｜ TYPE2**

《オペランドの省略値》

- 推奨モード適用時の省略値：TYPE2
- 0904 互換モード適用時の省略値：TYPE1

SQL 拡張最適化オプションで「ハッシュジョイン，副問合せのハッシュ実行」を指定した場合のハッシング方式を指定します。

TYPE1：バージョン 07-02 より前のハッシング方式です。

TYPE2：TYPE1 に比べて，均等にハッシングされます。

《指定値の目安》

- 通常は TYPE2 を指定してください。ただし，結合条件に指定した列のデータによっては均等にならないこともあるので，その場合は TYPE1 を指定してください。
- 07-02 以降にバージョンアップしたユーザで，TYPE1 を指定して期待した性能が得られなかった場合は，TYPE2 を指定してください。

◆ pd_apply_search_ats_num = サーチ条件 ATS を適用する絞り込み値の組み合わせ個数の上限 ～＜符号なし整数＞((255～30000))

《オペランドの省略値》

- 推奨モード適用時の省略値：30000
- 0904 互換モード適用時の省略値：255

インデクスを用いた検索で，IN 述語及び限定述語（＝ ANY（表副問合せ），＝ SOME（表副問合せ））を指定する場合，サーチ条件 ATS 又は RANGES を適用する絞り込み値の組み合わせ個数の上限を指定します。

《指定値の目安》

バージョン 09-50 より前の HiRDB をバージョンアップする場合，バージョンアップする前の HiRDB のバージョンによって指定値の目安が異なります。

- バージョン 08-04 以降の HiRDB をバージョンアップする場合

  指定値を，バージョンアップ前と同じにすることをお勧めします。
- バージョン 08-04 より前の HiRDB をバージョンアップする場合

  指定値を，255 にすることをお勧めします。

  IN 述語及び限定述語（＝ ANY（表副問合せ），＝ SOME（表副問合せ））を指定し，サーチ条件の絞り込み値の組み合わせ個数が 255 を超えて RANGE，又は RANGES になっている SQL 文の中には，このオプションの指定値を変更すれば性能向上することがあります。ただし，IN 述語を使用したケースで 1 件もヒットしない場合は性能劣化することがあります。サーチ条件種別については，マニュアル「HiRDB コマンドリファレンス」を参照してください。

《注意事項》

指定値を超えた場合は，内部的に異なるサーチ条件に変更して SQL を実行します。そのため，意図したアクセスパスにならないおそれがあります。

11.3.6　一時表に関するオペランド

◆ **pd_max_tmp_table_rdarea_no = 一時表用 RD エリアの最大数**
～＜符号なし整数＞((0～131088))

《オペランドの省略値》

- 推奨モード適用時の省略値：次に示す計算式の値を仮定します。
  省略値＝ min（131088，pd_max_rdarea_no−3）
- 0904 互換モード適用時の省略値：0

一時表用 RD エリアの最大数を指定します。一時表用 RD エリアの合計数がこのオペランドの値を超えている場合，HiRDB を開始できません。

《指定値の目安》

- 今後追加する一時表用 RD エリアの数を含めた一時表用 RD エリアの合計数を指定してください。
- HiRDB/パラレルサーバの場合，このオペランドの値はバックエンドサーバごとに適用されます。そのため，一時表用 RD エリア数が最も多いバックエンドサーバを目安にしてこのオペランドを指定してください。

《注意事項》

- このオペランドの指定値は HiRDB が使用する共用メモリサイズに影響するため，《指定値の目安》の値より大きい値を指定しないでください。《指定値の目安》の値より大きい値を指定すると，共用メモリ不足で HiRDB を開始できないおそれがあります。
- このオペランドの指定値は，pd_max_rdarea_no オペランドの指定値より小さい値にしてください。
- pd_max_temporary_object_no オペランドの指定値が 1 以上の場合は，このオペランドの指定値も 1 以上にしてください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_max_rdarea_no オペランド
- pd_max_temporary_object_no オペランド

《各見積もり式への影響》

pd_max_tmp_table_rdarea_no オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「シングルサーバが使用する共用メモリの計算式」の「計算式 9」
- 「フロントエンドサーバが使用する共用メモリの計算式」

11.3.7　通信処理に関するオペランド

◆ pd_registered_port_level = 0 ｜ 1

《オペランドの省略値》

- 推奨モード適用時の省略値：1
- 0904 互換モード適用時の省略値：0

このオペランドでは，HiRDB 予約ポート機能の対象範囲を指定します。

《前提条件》

pd_registered_port オペランドを指定している必要があります。

《指定値の目安》

0 を指定した場合と 1 を指定した場合の対象範囲を次に示します。

HiRDB が使用するポート番号		pd_registered_port_level の値	
		0	1
HiRDB サーバが使用するポート番号	サーバ間通信用送信ポート番号	○	○
	サーバ間通信用受信ポート番号	○	○
	pd_name_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-p オプションに指定したポート番号	×	×
	pd_service_port オペランドに指定したポート番号	×	×
	pd_scd_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-s オプションに指定したポート番号	×	×
	pd_trn_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-t オプションに指定したポート番号	×	×
	pd_mlg_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-m オプションに指定したポート番号	×	×
	pd_alv_port オペランドに指定したポート番号	×	×
	pdunit オペランドの-a オプションに指定したポート番号	×	×
	pd_crypto_service_port オペランドに指定したポート番号	×	×
	コマンド用送信ポート番号	×	×
	コマンド用受信ポート番号	×	×
	対クライアント用送信ポート番号	×	○
	対クライアント用受信ポート番号	○	○

HiRDB が使用するポート番号		pd_registered_port_level の値	
		0	1
HiRDB クライアントが使用するポート番号	送信ポート番号	×	×
	受信ポート番号	×	×

（凡例）

○：HiRDB 予約ポート機能の対象範囲となります。pd_registered_port オペランドで指定したポート番号が使用されます。

×：HiRDB 予約ポート機能の対象範囲となりません。

◆ pd_inet_unix_bufmode = os ｜ conf

《オペランドの省略値》

- 推奨モード適用時の省略値：os
- 0904 互換モード適用時の省略値：次の値になります。
  Linux の場合：os
  AIX の場合：conf

HiRDB の通信処理で使用する送受信バッファ長を指定します。

os：OS の設定値を使用します。

次のオペランドに送受信バッファ長を指定していても，指定値を無視して OS の設定値を使用します。

- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド

conf：次のオペランドで指定した値を使用します。

- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド

《指定値の目安》

- 通常は，このオペランドに os を指定して OS の設定値を使用してください。
- このオペランドに conf を指定する場合は，次のメリットとデメリットがあります。conf を指定する場合は，マシン検証を行うなど，十分に検討してください。

  ・BLOB データなどの大きいデータを送受信する際に，送受信バッファ長が小さいとデータの再送が発生し，通信性能及び CPU 利用率に影響を及ぼすことがあります。このような場合，送受

信バッファ長を大きくすると，通信性能が向上したり，CPU 利用率が低下したりすることがあります。

・数値データなどの小さいデータを送受信する際に，送受信バッファ長が大きいと，TCP/IP の制御によってデータを受信しても，すぐに ACK を返さない（遅延 ACK）ことがあります。この影響で性能が劣化することがあります。

《注意事項》

送受信バッファ長の OS 設定値，変更方法，変更できるサイズの範囲は，OS によって異なります。詳細は，各 OS のマニュアルを参照してください。

《ほかのオペランドとの関連》

このオペランドは，次に示すオペランドと関連があります。

- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_inet_unix_utl_bufmode オペランド
- pd_utl_buff_size オペランド

このオペランドの指定値と関連するオペランドの指定値との組み合わせで設定される送受信バッファ長を次の表に示します。TCP INET ドメイン通信でのクライアントとの通信，及びサーバとの通信を表 11-2，表 11-3，表 11-4 に，TCP UNIX ドメイン通信でのクライアントとの通信，及びサーバとの通信を表 11-5，表 11-6，表 11-7 に示します。

■TCP INET ドメインでの通信

表 11–2　TCP INET ドメインでのクライアントとの通信

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				設定される送受信バッファ長
		pd_tcp_inet_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_inet_bufsize	
1	os	–	–	–	–	• connect ソケット，accept ソケット，listen ソケット OS 設定値
2	conf	0		0	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
3					0 以外	• connect ソケット，listen ソケット OS 設定値 • accept ソケット

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				設定される送受信バッファ長
		pd_tcp_inet_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_inet_bufsize	
						pd_ipc_inet_bufsize オペランドの指定値
4				1	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
5					0 以外	• connect ソケット OS 設定値 • accept ソケット，listen ソケット pd_ipc_inet_bufsize オペランドの指定値
6		0 以外		0	0	• connect ソケット pd_tcp_inet_bufsize オペランドの指定値 • accept ソケット，listen ソケット OS 設定値
7					0 以外	• connect ソケット pd_tcp_inet_bufsize オペランドの指定値 • accept ソケット pd_ipc_inet_bufsize オペランドの指定値 • listen ソケット OS 設定値
8				1	0	• connect ソケット pd_tcp_inet_bufsize オペランドの指定値 • accept ソケット，listen ソケット OS 設定値
9					0 以外	• connect ソケット pd_tcp_inet_bufsize オペランドの指定値 • accept ソケット，listen ソケット pd_ipc_inet_bufsize オペランドの指定値

（凡例）

－：指定しても無視されます。

表 11–3　TCP INET ドメインでのサーバとの通信（pd_utl_buff_size オペランドを使用しない通信の場合）

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_inet_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_inet_bufsize	
1	os	–	–	–	–	• connect ソケット，accept ソケット，listen ソケット OS 設定値
2	conf			0	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
3					0 以外	• connect ソケット，accept ソケット pd_ipc_inet_bufsize オペランドの指定値 • listen ソケット OS 設定値
4				1	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
5					0 以外	• connect ソケット，accept ソケット，listen ソケット pd_ipc_inet_bufsize オペランドの指定値

（凡例）

–：指定しても無視されます。

表 11–4　TCP INET ドメインでのサーバとの通信（pd_utl_buff_size オペランドを使用した通信の場合）

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_inet_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_inet_bufsize	
1	os	–	–	–	–	• connect ソケット，accept ソケット，listen ソケット OS 設定値
2	conf		conf	0	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
3					0 以外	• connect ソケット，accept ソケット pd_ipc_inet_bufsize オペランドの指定値

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_inet_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_inet_bufsize	
						• listen ソケット OS 設定値
4				1	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
5					0 以外	• connect ソケット，accept ソケット，listen ソケット pd_ipc_inet_bufsize オペランドの指定値
6			auto	0	0	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値
7					0 以外	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値
8				1	0	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値
9					0 以外	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット pd_ipc_inet_bufsize オペランドの指定値

（凡例）

－：指定しても無視されます。

■TCP UNIX ドメインでの通信

表 11–5　TCP UNIX ドメインでのクライアントとの通信

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_unix_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_unix_bufsize	
1	os	–	–	–	–	• connect ソケット，accept ソケット，listen ソケット OS 設定値
2	conf	0		0	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
3					0 以外	• connect ソケット，listen ソケット OS 設定値 • accept ソケット pd_ipc_unix_bufsize オペランドの指定値
4				1	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
5					0 以外	• connect ソケット，listen ソケット OS 設定値 • accept ソケット pd_ipc_unix_bufsize オペランドの指定値
6		0 以外		0	0	• connect ソケット pd_tcp_unix_bufsize 指定値 • accept ソケット，listen ソケット OS 設定値
7					0 以外	• connect ソケット pd_tcp_unix_bufsize 指定値 • accept ソケット pd_ipc_unix_bufsize オペランドの指定値 • listen ソケット OS 設定値
8				1	0	• connect ソケット pd_tcp_unix_bufsize 指定値 • accept ソケット，listen ソケット OS 設定値
9					0 以外	• connect ソケット pd_tcp_unix_bufsize 指定値 • accept ソケット

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_unix_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_unix_bufsize	
						pd_ipc_unix_bufsize オペランドの指定値 • listen ソケット OS 設定値

（凡例）

−：指定しても無視されます。

表 11–6　TCP UNIX ドメインでのサーバとの通信（pd_utl_buff_size オペランドを使用しない通信の場合）

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_unix_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_unix_bufsize	
1	os	−	−	−	−	• connect ソケット，accept ソケット，listen ソケット OS 設定値
2	conf			0	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
3					0 以外	• connect ソケット，accept ソケット pd_ipc_unix_bufsize オペランドの指定値 • listen ソケット OS 設定値
4				1	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
5					0 以外	• connect ソケット，accept ソケット，listen ソケット pd_ipc_unix_bufsize オペランドの指定値

（凡例）

−：指定しても無視されます。

表 11–7　TCP UNIX ドメインでのサーバとの通信（pd_utl_buff_size オペランドを使用した通信の場合）

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_unix_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_unix_bufsize	
1	os	−	−	−	−	• connect ソケット，accept ソケット，listen ソケット OS 設定値
2	conf		conf	0	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
3					0 以外	• connect ソケット，accept ソケット pd_ipc_unix_bufsize オペランドの指定値 • listen ソケット OS 設定値
4				1	0	• connect ソケット，accept ソケット，listen ソケット OS 設定値
5					0 以外	• connect ソケット，accept ソケット pd_ipc_unix_bufsize オペランドの指定値 • listen ソケット OS 設定値
6			auto	0	0	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値
7					0 以外	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値
8				1	0	• connect ソケット，accept ソケット pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値
9					0 以外	• connect ソケット，accept ソケット

項番	pd_inet_unix_bufmode オペランドの指定値	関連するオペランドの指定値				使用する送受信バッファ長
		pd_tcp_unix_bufsize	pd_inet_unix_utl_bufmode	pd_listen_socket_bufset	pd_ipc_unix_bufsize	
						pd_utl_buff_size オペランドの指定値から自動計算した値 • listen ソケット OS 設定値

（凡例）

－：指定しても無視されます。

◆ **pd_ipc_inet_bufsize = サーバのユニット間通信で使用する送受信バッファ長～＜符号なし整数＞((0～262144))《16384》（単位：バイト）**

サーバのユニット間通信（TCP INET ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。

《指定値の目安》

- 通常は，このオペランドを指定する必要はありません。pd_inet_unix_bufmode オペランドに os を指定し，OS の設定値を使用してください。
- pd_inet_unix_bufmode オペランドに conf を指定し，HiRDB の通信処理で使用する送受信バッファ長をオペランドで指定する場合は，このオペランドを指定してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_inet_unix_bufmode オペランド
- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_inet_unix_utl_bufmode オペランド
- pd_utl_buff_size オペランド

◆ **pd_ipc_unix_bufsize = サーバのユニット内通信で使用する送受信バッファ長～＜符号なし整数＞((0～262144))《16384》（単位：バイト）**

サーバのユニット内通信（TCP UNIX ドメイン）で使用する送受信バッファの最大値を 4096 の倍数で指定します。

《指定値の目安》

- 通常は，このオペランドを指定する必要はありません。pd_inet_unix_bufmode オペランドに os を指定し，OS の設定値を使用してください。
- pd_inet_unix_bufmode オペランドに conf を指定し，HiRDB の通信処理で使用する送受信バッファ長をオペランドで指定する場合は，このオペランドを指定してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_inet_unix_bufmode オペランド
- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_inet_unix_utl_bufmode オペランド
- pd_utl_buff_size オペランド

◆ pd_tcp_inet_bufsize = HiRDB サーバが存在するホスト外の HiRDB クライアントとの通信で使用する送受信バッファ長

～＜符号なし整数＞((0～262144))《0》（単位：バイト）

HiRDB サーバがあるホスト以外のホスト上から接続した HiRDB クライアントとの通信（TCP INET ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。pd_inet_unix_bufmode オペランドに os を指定し，OS の設定値を使用してください。
- pd_inet_unix_bufmode オペランドに conf を指定し，HiRDB の通信処理で使用する送受信バッファ長をオペランドで指定する場合は，このオペランドを指定してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_inet_unix_bufmode オペランド
- pd_tcp_unix_bufsize オペランド

- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_inet_unix_utl_bufmode オペランド
- pd_utl_buff_size オペランド

◆ pd_tcp_unix_bufsize = HiRDB サーバが存在するホスト内の HiRDB クライアントとの通信で使用する送受信バッファ長

～＜符号なし整数＞((0～262144))《16384》（単位：バイト）

HiRDB サーバがあるホスト上から接続した次の HiRDB クライアントとの通信（TCP UNIX ドメイン）で使用する送受信バッファの最大値を，4096 の倍数で指定します。

- HiRDB Version4.0 の 04-02 以前の HiRDB クライアント
- マルチスレッド対応の HiRDB XA ライブラリを使用した HiRDB クライアント

《指定値の目安》

- 通常はこのオペランドを指定する必要はありません。pd_inet_unix_bufmode オペランドに os を指定し，OS の設定値を使用してください。
- pd_inet_unix_bufmode オペランドに conf を指定し，HiRDB の通信処理で使用する送受信バッファ長をオペランドで指定する場合は，このオペランドを指定してください。

《注意事項》

- 0 を指定した場合は OS の設定値を使用します。
- TCP のバッファサイズの上限値は各 OS で異なるので注意してください。
- OS が Linux の場合，TCP UNIX ドメインプロトコルによる通信をしないため，指定したバッファ長が無効になります。この場合，pd_tcp_inet_bufsize 指定値が適用されます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_inet_unix_bufmode オペランド
- pd_tcp_inet_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_inet_unix_utl_bufmode オペランド
- pd_utl_buff_size オペランド

◆ pd_listen_socket_bufset = 0 | 1

HiRDB の通信処理で使用する TCP INET ドメイン用の listen ソケット生成時に送受信バッファを listen ソケットに設定するかどうかを指定します。

0：listen ソケットに送受信バッファサイズを設定しません。

1：listen ソケットに送受信バッファサイズを設定します。

送受信バッファサイズは，pd_ipc_inet_bufsize オペランドに指定した値を設定します。ただし，次の条件を満たす必要があります。

- pd_inet_unix_bufmode オペランドに conf を指定する。
- pd_ipc_inet_bufsize オペランドに 0 以外の値を指定する。

詳細は，pd_inet_unix_bufmode オペランドの説明を参照してください。

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。pd_inet_unix_bufmode オペランドに os を指定し，OS の設定値を使用してください。
- pd_inet_unix_bufmode オペランドに conf を指定し，このオペランドで指定した送受信バッファサイズを使用する場合，pd_ipc_inet_bufsize オペランドで指定した値が OS 設定値よりも小さくなると，通信時間の遅延が発生することがあります。このようなとき，このオペランドに 1 を指定することで通信性能が向上することがあります。

《注意事項》

このオペランドに 1 を指定した場合，pd_utl_inet_unix_bufmode オペランドに conf を指定し，pd_utl_buff_size オペランドを使用した通信処理をするときでも，このオペランドで指定した送受信バッファサイズを使用してください。また，listen ソケット，accept ソケットで使用する送受信バッファサイズが同じサイズになるよう設定してください。listen ソケット，accept ソケットで使用する送受信バッファサイズが異なっていると，通信性能が改善されないことがあります。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_inet_unix_bufmode オペランド
- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_inet_unix_utl_bufmode オペランド
- pd_utl_buff_size オペランド

◆ pd_utl_buff_size = ユティリティの通信用バッファ長 ～＜符号なし整数＞((8～64))《32》（単位：キロバイト）

次に示すユティリティのプロセスが使用する通信用バッファ 1 面のサイズをキロバイト単位で指定します。

- データベース作成ユティリティ（pdload）
- データベース再編成ユティリティ（pdrorg の-g オプション指定時）

- リバランスユティリティ（pdrbal）
- データベース回復ユティリティ（pdrstr）
- データベース複写ユティリティ（pdcopy）

《利点》

ユティリティを処理するサーバプロセスと HiRDB サーバ間の通信処理でデータの再送がある場合，このオペランドの指定値を大きくすると通信回数を減らすことができます。その結果，性能を向上できることがあります。

《指定値の目安》

通信用バッファ長＝↑(ページ長※×２)÷1024↑（単位：キロバイト）

注※　ユティリティの処理対象となる RD エリアの最大ページ長

《オペランドの規則》

- 指定値は，4 の倍数単位に指定してください。4 の倍数でない場合は，自動的に 4 の倍数に切り上げられます。
- 指定値が 32 より小さい場合でも，データベース回復ユティリティ及びデータベース複写ユティリティ実行時には，自動的に 32 に切り上げられます。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。詳細は，pd_inet_unix_bufmode オペランドの説明を参照してください。

- pd_inet_unix_bufmode オペランド
- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_inet_unix_utl_bufmode オペランド

《各見積もり式への影響》

pd_utl_buff_size オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「データベース回復ユティリティ（pdrstr）実行時のメモリ所要量」
- 「データベース作成ユティリティ（pdload）実行時のメモリ所要量」
- 「データベース複写ユティリティ（pdcopy）実行時のメモリ所要量」

◆ **pd_inet_unix_utl_bufmode = auto ｜ conf**

pd_utl_buff_size オペランドを使用する通信処理で使用する送受信バッファ長を指定します。

auto：

pd_utl_buff_size オペランドの指定値を基に自動計算した値を使用します。

conf：

次のオペランドで指定した値を使用します。

- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド

《指定値の目安》

- 通常，このオペランドを指定する必要はありません。pd_inet_unix_bufmode オペランドに os を指定し，OS の設定値を使用してください。
- pd_inet_unix_bufmode オペランドに conf を指定し，pd_listen_socket_bufset オペランドに 1 を指定する場合は，このオペランドに conf を指定してください。

《ほかのオペランドとの関連》

このオペランドは次に示すオペランドと関連があります。

- pd_inet_unix_bufmode オペランド
- pd_tcp_inet_bufsize オペランド
- pd_tcp_unix_bufsize オペランド
- pd_ipc_inet_bufsize オペランド
- pd_ipc_unix_bufsize オペランド
- pd_listen_socket_bufset オペランド
- pd_utl_buff_size オペランド

11.3.8　日付・時刻に関するオペランド

◆ **pd_leap_second = Y ｜ N**

《オペランドの省略値》

- 推奨モード適用時の省略値：Y
- 0904 互換モード適用時の省略値：N

時刻データ，時刻印データ，及び日時書式でうるう秒を使用するかどうかを指定します。

Y：

- 時刻データ又は時刻印データの場合
  埋込み変数及び?パラメタの値，並びにそれらを文字列表現した値でうるう秒を使用できます。
- 日付書式の場合
  秒に対する値でうるう秒を使用できます。

N：

うるう秒を使用しません。

《注意事項》

うるう秒が含まれる時刻データ又は時刻印データがデータベースに格納されている場合，このオペランドで N を指定すると，うるう秒を含む時刻データ又は時刻印データを使用した時刻演算を行ったときにエラーとなることがあります。このとき，スカラ関数 SECOND ではうるう秒を抽出できます。

11.3.9　ユニット用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_sysst_initial_error = stop ｜ continue ｜ excontinue**

《オペランドの省略値》

- 推奨モード適用時の省略値：excontinue
- 0904 互換モード適用時の省略値：stop

HiRDB（又はユニット）の開始時，HiRDB はユニット用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- ユニット用ステータスファイルの実体がない
- ユニット用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_syssts_file_name_1～7 オペランドに指定したユニット用ステータスファイルを対象とします。

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，pd_syssts_subfile_name1～7 オペランドに指定したユニット用副ステータスファイルも対象とします。ログ適用サイトの場合は，説明中の pd_syssts_last_active_file オペランド，及び pd_syssts_last_active_side オペランドを，それぞれ pd_syssts_last_active_subfile オペランド，及び pd_syssts_last_active_side_sub オペランドに置き換えてください。

stop：

現用ファイルの特定処理時にユニット用ステータスファイルの異常を検知した場合，HiRDB（又はユニット）の開始処理を中止します。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にユニット用ステータスファイルの異常を検知しても，現用ファイルが正常な場合は HiRDB（又はユニット）の開始処理を続行します。ただし，pd_syssts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_syssts_singleoperation オペランドとの関係を次に示します。

●pd_syssts_singleoperation オペランドとの関係

pd_syssts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	ユニット用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，HiRDB（又はユニット）の開始処理を中止します。	HiRDB 管理者が現用ファイルを特定して pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドを指定します。その後，HiRDB（又はユニット）を開始してください。 ステータスファイルの異常の要因を取り除き障害から復旧した後は，HiRDB を正常停止して pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドの指定を省略してください。障害から復旧した後もオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB（又はユニット）の開始処理に失敗するおそれがあります。
stop（省略値）	ユニット用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定して HiRDB（又はユニット）の開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，HiRDB の開始処理を中止します。	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドを指定します。その後，HiRDB（又はユニット）を開始してください。 ステータスファイルの異常の要因を取り除き障害から復旧した後は，HiRDB を正常停止して pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドの指定を省略してください。障害から復旧した後もオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB（又はユニット）の開始処理に失敗するおそれがあります。

●HiRDB が現用ファイルを特定できないケース

pd_syssts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞

pd_syssts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_syssts_initial_error オペランドに excontinue を指定
- pd_syssts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_syssts_initial_error オペランドの値	
	stop	continue 又は excontinue
HiRDB 又はユニット開始時の HiRDB の処理	ユニット用ステータスファイルの異常を検知した場合，HiRDB（又はユニット）の開始処理を中止します。	一部のユニット用ステータスファイルに異常があっても，現用ファイルが正常な場合は HiRDB（又はユニット）の開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	HiRDB の開始時に全ユニット用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害が発生した場合，予備ファイルにスワップできます。	HiRDB の開始時に一部のユニット用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	ユニット用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはユニット用ステータスファイルをスワップできないことがあります。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なく HiRDB（又はユニット）の開始処理を中止します。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

このオペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●このオペランドの指定値と HiRDB の処理

pd_syssts_initial_errorオペランドの値	ステータスファイル障害	pd_sysstss_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_syssts_last_active_fileの指定	pd_syssts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_syssts_last_active_sideオペランドの指定	pd_syssts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しない
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しない
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[3]	pd_syssts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_syssts_last_active_file 及び pd_syssts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_syssts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_syssts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_syssts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

◆ pd_syssts_singleoperation = stop ｜ continue

ユニット用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することをステータスファイルの片系運転といいます。ステータスファイルの片系運転については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

また，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，ユニット用副ステータスファイルも対象となります。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_syssts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。このため，ステータスファイルの障害によって HiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《ほかのオペランドとの関連》

pd_syssts_singleoperation 及び pd_syssts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

11.3.10　サーバ用ステータスファイル（障害発生時）に関するオペランド

ステータスファイルに障害が発生したときの対処方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。

◆ **pd_sts_initial_error = stop ｜ continue ｜ excontinue**

《オペランドの省略値》

- 推奨モード適用時の省略値：excontinue
- 0904 互換モード適用時の省略値：stop

サーバ（シングルサーバ，フロントエンドサーバ，ディクショナリサーバ，又はバックエンドサーバ）の開始時，HiRDB はサーバ用ステータスファイルの現用ファイルを特定する処理を実行します。この特定処理で，次に示す異常を検知したときの HiRDB の処置を指定します。

- サーバ用ステータスファイルの実体がない
- サーバ用ステータスファイルの障害を検知した

現用ファイルの特定処理は pd_sts_file_name_1～7 オペランドに指定したサーバ用ステータスファイルを対象とします。

ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，pd_sts_subfile_name_1～7 オペランドに指定したサーバ用副ステータスファイルも対象とします。ログ適用サイトの場合は，説明中の pd_sts_last_active_file オペランド，及び pd_sts_last_active_side オペランドを，それぞれ pd_sts_last_active_subfile オペランド，及び pd_sts_last_active_side_sub オペランドに置き換えてください。

stop：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。この場合，異常を検知したステータスファイルの対策を実施した後に HiRDB を開始してください。

continue 又は excontinue：

現用ファイルの特定処理時にサーバ用ステータスファイルの異常を検知しても，現用ファイルが正常な場合はサーバの開始処理を続行します。ただし，pd_sts_singleoperation オペランドの値（ステータスファイルの片系運転をするかどうか）によっては開始処理を中止することがあります。pd_sts_singleoperation オペランドとの関係を次に示します。

●pd_sts_singleoperation オペランドとの関係

pd_sts_singleoperation オペランドの値	HiRDB の処理	HiRDB 管理者の処置
continue	サーバ用ステータスファイルの異常を検知した場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。 ステータスファイルの異常の要因を取り除き障害から復旧した後は，HiRDB を正常停止して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドの指定を省略してください。障害から復旧した後もオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。
stop（省略値）	サーバ用ステータスファイルの異常を検知した場合，HiRDB が現用ファイルを特定してサーバの開始処理を続行します。ただし，A 系と B 系のファイルが次に示す表（**HiRDB が現用ファイルを特定できないケース**）の条件のどれかに当てはまる場合，HiRDB は現用ファイルを特定できないため，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	HiRDB が現用ファイルを特定できない場合は，HiRDB 管理者が現用ファイルを特定して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定します。その後，HiRDB を開始してください。 ステータスファイルの異常の要因を取り除き障害から復旧した後は，HiRDB を正常停止して pd_sts_last_active_file 及び pd_sts_last_active_side オペランドの指定を省略してください。障害から復旧した後もオペランドを指定したままだと現用のステータスファイルを認識できず HiRDB の開始処理に失敗するおそれがあります。

●HiRDB が現用ファイルを特定できないケース

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
continue	障害閉塞	障害閉塞
	障害閉塞	オープン（初期状態）
	障害閉塞	ファイル実体なし

pd_sts_initial_error オペランドの値	A 系ファイルの状態	B 系ファイルの状態
	オープン（初期状態）	障害閉塞
	オープン（初期状態）	ファイル実体なし
	ファイル実体なし	オープン（初期状態）
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし
excontinue	障害閉塞	障害閉塞
	障害閉塞	ファイル実体なし
	ファイル実体なし	障害閉塞
	ファイル実体なし	ファイル実体なし

したがって，HiRDB 管理者の操作をなるべく少なくしたい場合（pd_sts_last_active_file 及び pd_sts_last_active_side オペランドを指定するケースを減らしたい場合）は，次に示すようにオペランドを指定してください。

- pd_sts_initial_error オペランドに excontinue を指定
- pd_sts_singleoperation オペランドに stop を指定

《指定値の目安》

各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_initial_error オペランドの値	
	stop	continue 又は excontinue
サーバ開始時の HiRDB の処理	サーバ用ステータスファイルの異常を検知した場合，サーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。	一部のサーバ用ステータスファイルに異常があっても，現用ファイルが正常な場合はサーバの開始処理を続行します。
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB 開始時の障害対策を簡易化する場合は continue 又は excontinue を指定します。
メリット	サーバの開始時に該当するサーバの全サーバ用ステータスファイルの正常が保証されます。このため，HiRDB の開始後に現用ファイルの障害が発生した場合，予備ファイルにスワップできます。	サーバの開始時に一部のサーバ用ステータスファイルの障害を検知しても，残りの正常なファイルだけで HiRDB を開始できます。このため，HiRDB の停止時間を短くできます。この場合，予備ファイルが少なくなっているため，障害状態のステータスファイルをすぐに修復する必要があります。
デメリット	サーバ用ステータスファイルの障害によって HiRDB の開始処理が中止される可能性が高くなります。	予備ファイルが少ない状態で HiRDB を稼働することがあるため，システムの信頼性が低下します。予備ファイルの数によってはサーバ用ステータスファイルをスワップできないことがあります。

《注意事項》

- 現用ファイルが両系とも異常の場合は，このオペランドの指定値に関係なくサーバの開始処理を中止し，HiRDB（HiRDB/パラレルサーバの場合はユニット）を開始しません。
- HiRDB を開始する前に現用ファイルを pdstsinit コマンドで初期化しないでください。初期化すると HiRDB を再開始できません。
- このオペランドに excontinue を指定し，すべての世代のステータスファイルが片系運転でかつオープン（初期状態）の場合，HiRDB は起動しません。

《備考》

オペランドの指定値と HiRDB の処理及び HiRDB 管理者の処置を次に示します。

●オペランドの指定値と HiRDB の処理

pd_sts_initial_errorオペランドの値	ステータスファイル障害	pd_sts_singleoperationオペランドの値	HiRDBが現用ファイルを特定できるか	pd_sts_last_active_fileの指定	pd_sts_last_active_fileオペランドの値と，オープンできたステータスのうち，最新のファイルが一致するか	現用ファイル障害	pd_sts_last_active_sideオペランドの指定	pd_sts_last_active_sideオペランドに指定されたファイルが使用できるか	表「HiRDBの処理及びHiRDB管理者の処置」の対応番号
stop	なし	－	－	－	－	－	－	－	[1]
	あり	－	－	－	－	－	－	－	[5]
continue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しない
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
excontinue	なし	－	－	－	－	－	－	－	[1]
	あり	stop（省略値）	特定できる	－	－	－	－	－	[2]
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]
		continue	特定できる	－	－	－	－	－	発生しない
			特定できない	なし	－	－	－	－	[6]
				あり	一致する	なし	－	－	[3]
						あり	なし	－	[9]
							あり	使用できる	[4]
								使用できない	[7]
					一致しない	－	－	－	[8]

（凡例）－：該当しません（条件がHiRDBの処理に影響を与えません）。

●HiRDB の処理及び HiRDB 管理者の処置

対応番号	HiRDB の処理	HiRDB 管理者の処置
[1]	HiRDB の開始処理を続行します。	なし
[2]	HiRDB が最新の現用ファイルを特定して，開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。

対応番号	HiRDB の処理	HiRDB 管理者の処置
[3]	pd_sts_last_active_file オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[4]	pd_sts_last_active_file 及び pd_sts_last_active_side オペランドに指定されたファイルを現用ステータスファイルとして HiRDB の開始処理を続行します。	障害によって閉塞しているファイルを予備ファイルにしてください。
[5]	pd_sts_initial_error オペランドに stop が指定されているため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000010 が示す対策をしてください。
[6]	前回稼働時の現用ファイルが特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000016 が示す対策をしてください。
[7]	HiRDB が特定した現用ファイルの正常な系と，pd_sts_last_active_side オペランドに指定された系が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000017 が示す対策をしてください。
[8]	HiRDB が特定した現用ファイル名と，pd_sts_last_active_file オペランドに指定したファイル名が一致しないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000015 が示す対策をしてください。
[9]	前回稼働時の現用ファイルの正常な系が特定できないため，HiRDB の開始処理を中止します。	マニュアル「HiRDB メッセージ」を参照して，KFPS01005-E メッセージの理由コード 0000000018 が示す対策をしてください。

◆ pd_sts_singleoperation = stop | continue

サーバ用ステータスファイルの片系運転をするかどうかを指定します。予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，正常な系（片方の系）だけで処理を続行することをステータスファイルの片系運転といいます。ステータスファイルの片系運転については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

なお，予備ファイルがある状況で現用ファイルの片系に障害が発生した場合は，このオペランドの指定値に関係なく，ステータスファイルをスワップして処理を続行します（片系運転にはなりません）。

また，ログ同期方式のリアルタイム SAN レプリケーションを使用している場合，ログ適用サイトでは，サーバ用副ステータスファイルも対象となります。

stop：

片系運転をしません。片系運転になる場合は HiRDB（HiRDB/パラレルサーバの場合はユニット）を異常終了します。HiRDB が異常終了した場合，予備ファイルを準備した後に HiRDB を開始してください。

continue：

片系運転をします。片系運転になると KFPS01044-I メッセージが出力されます。片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB（HiRDB/パラレルサーバの場合はユニット）が異常終了したりすると，HiRDB を再開始できなくなるため，片系運転になった場合はすぐに予備ファイルを準備してください。

《指定値の目安》

- stop を指定してシステムの信頼性を上げることを推奨します。また，現用ファイルの障害に対しては，予備ファイルの数を増やして備えることを推奨します。
- 各指定値の目安，メリット，及びデメリットを次に示します。

項目	pd_sts_singleoperation オペランドの値	
	stop	continue
指定値の目安	システムの信頼性を上げる場合は stop を指定します。	HiRDB を停止させないことを重視する場合は continue を指定します。
メリット	予備ファイルがない状況で現用ファイルの片系に障害が発生した場合，片系運転をしないで HiRDB を異常終了します。このため，現用ファイルの内容が失われる可能性が低くなります。	予備ファイルがない状況で現用ファイルの片系に障害が発生しても，処理を続行できます。このため，ステータスファイルの障害によって HiRDB が停止する可能性が低くなります。
デメリット	ステータスファイルの障害によって HiRDB が停止する可能性が高くなります。ただし，予備ファイルの数を増やすことで可能性を低くできます。	片系運転中に正常な系に障害が発生したり，又はステータスファイルの更新中に HiRDB が異常終了したりすると，現用ファイルの内容が失われるため，HiRDB を再開始できなくなります。

《ほかのオペランドとの関連》

pd_sts_singleoperation 及び pd_sts_initial_error オペランドの指定値の組み合わせによって，ステータスファイル障害時の HiRDB の処理が決定します。したがって，これら二つのオペランドの指定値は一緒に考えるようにしてください。

11.3.11　システムログファイルに関するオペランド

◆ pd_log_max_data_size = ログ入出力バッファ長 ～<符号なし整数>((32000～523000))（単位：バイト）

《オペランドの省略値》

- 推奨モード適用時の省略値：523000
- 0904 互換モード適用時の省略値：400000

システムログの入出力に使用するバッファの大きさをバイト単位で指定します。

《指定値の目安》

- **HiRDB/シングルサーバの場合**

  指定値のチューニング方法に従って指定値を変更してください。
- **HiRDB/パラレルサーバの場合**

  条件式 1 を満たす値を指定してください。pd_rpl_reflect_mode オペランドに uap を指定している場合，又は回復不要 FES を使用している場合は，条件式 1 と条件式 2 の両方を満たす値を

指定してください。指定値を最適化する場合は，指定値のチューニング方法に従って指定値を変更してください。

条件式 1：ログ入出力バッファ長≧a

a：72×（一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数）＋ 1344

条件式 2：ログ入出力バッファ長≧b

b：(一つのトランザクションによって参照処理又は更新処理の対象となる，バックエンドサーバ及びディクショナリサーバの最大数＋ 1）×128 ＋ 64

《指定値のチューニング方法》

統計解析ユティリティ（システムの稼働に関する統計情報）の次に示す情報を調べて，このオペランドの指定値を変更してください。

- 入出力待ちバッファ面数（# OF BUFFER FOR WAIT I/O）

  入出力待ちバッファ面数の平均値が 100 を大きく超える場合は，このオペランドの値を大きくして，平均値が 100 に近付くようにしてください。

- カレントバッファなしによる待ち回数（# OF WAIT THREAD）

  カレントバッファなしによる待ち回数が 0 以外の値になる場合は，このオペランドの値を大きくしてください。

《注意事項》

このオペランドの指定値は，トランザクションで実行する SQL レスポンスやスループットに影響します。小さな値を指定すると，システムログファイルへの書き込みが多発し，性能が低下するおそれがあります。

《ほかのオペランドとの関連》

- pd_log_write_buff_count オペランドとともに，ログ入出力バッファ長を決定してください。

◆ pd_log_rec_leng = システムログファイルのレコード長 ～<符号なし整数>((1024，2048，4096))（単位：バイト）

《オペランドの省略値》

- 推奨モード適用時の省略値：1024
- 0904 互換モード適用時の省略値：4096

システムログファイルのレコード長を指定します。1024，2048，4096 のどれかを指定します。

《指定値の目安》

システムログファイルのレコード長の設計方針に従って，レコード長を指定してください。システムログファイルのレコード長の設計方針については，マニュアル「HiRDB システム導入・設計ガイド」の「システムログファイルのレコード長」を参照してください。

《注意事項》

- pdloginit コマンドの-l オプションで設定するレコード長と異なるレコード長をこのオペランドに指定すると，そのシステムログファイルはオープンできません。

- システムログファイルのレコード長を変更する方法については，マニュアル「HiRDB システム運用ガイド」を参照してください。
- システムログファイルを配置した HiRDB ファイルシステム領域のセクタ長（pdfmkfs コマンドの-s オプション値）以上の値を指定してください。HiRDB ファイルシステム領域のセクタ長は，pdfstatfs -b コマンドで確認できます。詳細はマニュアル「HiRDB コマンドリファレンス」を参照してください。

◆ pd_log_rollback_buff_count = ロールバック用ログ入力バッファ面数 ～<符号なし整数>((0～256))

《オペランドの省略値》

- 推奨モード適用時の省略値：0
- 0904 互換モード適用時の省略値：次の値になります。
  - **ユニットに 1：1 スタンバイレス型系切り替え機能を適用している場合**
    代替 BES 数×2
  - **ユニットに影響分散スタンバイレス型系切り替えを適用している場合**
    （ホスト BES 数＋ pd_ha_max_act_guest_servers の値）×2
  - **上記以外の場合**
    ユニット内サーバ数×2

ロールバック処理でシステムログの入力に使用するバッファの面数を指定します。このオペランドに 0 を指定した場合，HiRDB がロールバック用ログ入力バッファ面数を決定します。

《指定値の目安》

- 0 を指定することを推奨します。0 を指定すると，HiRDB が指定値を自動計算します。
- 指定値が小さい場合は，ロールバックの同時実行が遅くなります。また，指定値が大きい場合は，ユニットコントローラ用共用メモリを必要以上に消費します。

《指定値のチューニング方法》

このオペランドに 0 を指定してください。0 を指定してメモリ不足が発生する場合は，このオペランドの指定を省略してください。

《各見積もり式への影響》

pd_log_rollback_buff_count オペランドの指定値を変更すると，次の見積もり式に影響があります。

マニュアル「HiRDB システム導入・設計ガイド」：

- 「HiRDB/シングルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」
- 「HiRDB/パラレルサーバのメモリ所要量の見積もり」の「ユニットコントローラが使用する共用メモリの計算式」

11.4　pdconfchk コマンドでチェックできるオペランド

互換モードを適用する場合に，検討が必要な各オペランドのうち，pdconfchk コマンドでチェックできるオペランドを次の表に示します。オペランド名はアルファベット順に記載しています。

表 11–8　pdconfchk コマンドでチェックできるオペランド

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_apply_search_ats_num	○	×	×	×	×	○
pd_dbsync_altwrite_skip	○	×	×	×	×	○
pd_hashjoin_hashing_mode	○	×	×	×	×	○
pd_inet_unix_bufmode	○	×	×	×	×	○
pd_inet_unix_utl_bufmode	○	×	×	×	×	○
pd_ipc_inet_bufsize	○	×	×	×	×	○
pd_ipc_unix_bufsize	○	×	×	×	×	○
pd_leap_second	○	×	×	×	×	○
pd_listen_socket_bufset	○	×	×	×	×	○
pd_log_max_data_size	○	×	×	×	×	○
pd_log_rec_leng	○	×	×	×	×	○
pd_log_rollback_buff_count	○	×	×	×	×	○
pd_max_tmp_table_rdarea_no	○	×	×	×	×	○
pd_registered_port_level	○	×	×	×	×	○
pd_sql_simple_comment_use	○	×	×	×	×	○
pd_sts_initial_error	○	×	×	×	×	○
pd_sts_singleoperation	○	×	×	×	×	○
pd_syssts_initial_error	○	×	×	×	×	○
pd_syssts_singleoperation	○	×	×	×	×	○
pd_tcp_inet_bufsize	○	×	×	×	×	○
pd_tcp_unix_bufsize	○	×	×	×	×	○
pd_trn_rollback_msg_interval	○	×	×	×	×	○
pd_utl_buff_size	○	×	×	×	×	○
pd_utl_exec_mode	○	×	×	×	×	○

（凡例）

○：チェック対象です。

×：チェック対象外です。

文法のチェック：

オペランドの文法が正しいかチェックします。

ファイルのチェック：

システムログファイル，シンクポイントダンプファイル，及びステータスファイルの有無をチェックします。pdconfchk コマンドで-n オプションを指定した場合はファイルのチェックをしません。

アクセス権限のチェック：

HiRDB 管理者にファイルのアクセス権があるかを確認します。

HiRDB 管理者に hosts ファイルのアクセス権があるかを確認します。

pdconfchk コマンドで-n オプションを指定した場合はアクセス権限のチェックをしません。

重複指定のチェック：

システムログファイル，シンクポイントダンプファイル，及びステータスファイルが重複していないかどうかをチェックします。

ホスト名のチェック：

ホスト名が hosts ファイルに記述されているかをチェックします。

サーバマシン間のチェック（HiRDB/パラレルサーバ限定）：

システムマネジャのサーバマシンを基準に，サーバマシン間の妥当性をチェックします。

11.5　適用 OS ごとのオペランド指定可否

互換モードを適用する場合に，検討が必要な各オペランドのうち，適用 OS ごとのオペランド指定可否を次の表に示します。オペランド名はアルファベット順に記載しています。

表 11–9　適用 OS ごとのオペランド指定可否

オペランド名	AIX	Linux
pd_apply_search_ats_num	○	○
pd_dbsync_altwrite_skip	○	○
pd_hashjoin_hashing_mode	○	○
pd_inet_unix_bufmode	○	○
pd_inet_unix_utl_bufmode	○	○
pd_ipc_inet_bufsize	○	○
pd_ipc_unix_bufsize	○	○
pd_leap_second	○	○
pd_listen_socket_bufset	○	○
pd_log_max_data_size	○	○
pd_log_rec_leng	○	○
pd_log_rollback_buff_count	○	○
pd_max_tmp_table_rdarea_no	○	○
pd_registered_port_level	○	○
pd_sql_simple_comment_use	○	○
pd_sts_initial_error	○	○
pd_sts_singleoperation	○	○
pd_syssts_initial_error	○	○
pd_syssts_singleoperation	○	○
pd_tcp_inet_bufsize	○	○
pd_tcp_unix_bufsize	○	○
pd_trn_rollback_msg_interval	○	○
pd_utl_buff_size	○	○
pd_utl_exec_mode	○	○

（凡例）

○：指定できるオペランドです。

付録

付録 A　影響分散スタンバイレス型系切り替え機能使用時に指定できるオペランド一覧（ユニット制御情報定義）

影響分散スタンバイレス型系切り替え機能使用時，ユニット制御情報定義に指定できるオペランドを次に示します。ユニット制御情報定義にはこれら以外のオペランドを指定できません。指定した場合は HiRDB を開始できません（KFPS05618-E メッセージが出力されます）。

- pd_aud_file_name※
- pd_down_watch_proc
- pd_ha_acttype
- pd_ha_agent
- pd_ha_max_act_guest_servers
- pd_ha_max_server_process
- pd_ha_process_count
- pd_ha_resource_act_wait_time
- pd_ha_unit
- pd_hostname
- pd_lck_deadlock_check
- pd_lck_deadlock_check_interval
- pd_prf_file_count
- pd_prf_file_size
- pd_prf_level
- pd_prf_output_directory
- pd_registered_port※
- pd_rpc_trace_name※
- pd_rpl_hdepath
- pd_service_port
- pd_syssts_file_name_1～7
- pd_syssts_initial_error
- pd_syssts_last_active_file
- pd_syssts_last_active_side
- pd_syssts_last_active_side_sub
- pd_syssts_last_active_subfile

- pd_syssts_singleoperation
- pd_syssts_subfile_name_1〜7
- pd_tmp_directory
- pd_trn_rcvmsg_store_buflen
- pd_unit_id
- pd_cmd_rconfig
- pd_lck_deadlock_check_timeout
- pd_lck_deadlock_check_pause

注※

同じサーバマシンに複数の HiRDB ユニットを配置した場合は，ユニットごとに指定値を変更してください。

付録 B.1　HiRDB/シングルサーバの場合

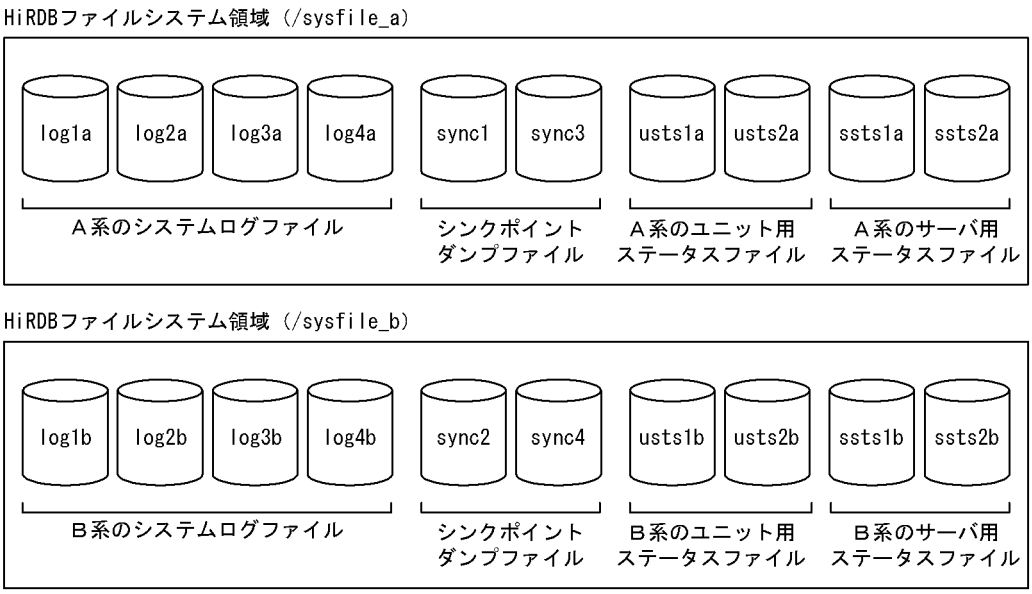
HiRDB/シングルサーバのシステム定義を作成します。システム構成は次のとおりです。

●ユニット構成



注　（　）内はサーバ名を表しています。

●システムファイルの構成



(1)　システム共通定義

```
set pd_system_id = PDB1                                          1
set pd_name_port = 20001                                         2
set pd_master_file_name = "/rdarea/mast/mast01"                  3
set pd_max_users = 100                                           4
pdunit -x host1 -u UNT1 -d "/HiRDB"                              5
pdstart -t SDS -s sds1 -u UNT1                                   6
pdbuffer -a gbuf01 -r RDMAST,RDDIC,RDDIR -n 1000                 7
pdbuffer -a gbuf02 -r RDAREA1,RDAREA2,RDAREA3 -n 1000            8
```

```
pdbuffer -a gbuf03 -r RDAREA4,RDAREA5,RDAREA6 -n 1000          9
pdbuffer -a gbuf04 -o -n 1000                                  10
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。
4. 最大同時接続数を指定します。
5. ユニットの構成を指定します。

   -x：ホスト名を指定します。

   -u：ユニット識別子を指定します。

   -d：HiRDB 運用ディレクトリ名を指定します。
6. サーバの構成を指定します。

   -t：サーバの種別（シングルサーバ：SDS）を指定します。

   -s：サーバ名を指定します。

   -u：ユニット識別子を指定します。
7. マスタディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディレクトリ用 RD エリアにグローバルバッファを割り当てます。
8. ユーザ用 RD エリア（RDAREA1～RDAREA3）にグローバルバッファを割り当てます。
9. ユーザ用 RD エリア（RDAREA4～RDAREA6）にグローバルバッファを割り当てます。
10. そのほかの RD エリアにグローバルバッファを割り当てます。

(2) ユニット制御情報定義

```
set pd_unit_id = UNT1                                                    1
set pd_syssts_file_name_1 = ”untsts1”,”/sysfile_a/usts1a”,¥              2
                                    ”/sysfile_b/usts1b”
set pd_syssts_file_name_2 = ”untsts2”,”/sysfile_a/usts2a”,¥
                                    ”/sysfile_b/usts2b”
```

〔説明〕

1. ユニット識別子を指定します。
2. ユニット用ステータスファイルの構成を指定します。

(3) シングルサーバ定義

```
set pd_log_dual = Y                                                      1
set pd_sts_file_name_1 = ”svrsts1”,”/sysfile_a/ssts1a”,¥                 2
                                  ”/sysfile_b/ssts1b”
set pd_sts_file_name_2 = ”svrsts2”,”/sysfile_a/ssts2a”,¥
```

```
                                    "/sysfile_b/ssts2b"
pdwork -v "/work01","/work02"                                          3
pdlogadfg -d sys -g log1 ONL                                           4
pdlogadfg -d sys -g log2 ONL
pdlogadfg -d sys -g log3 ONL
pdlogadfg -d sys -g log4 ONL
pdlogadpf -d sys -g log1 -a "/sysfile_a/log1a" -b "/sysfile_b/log1b"
pdlogadpf -d sys -g log2 -a "/sysfile_a/log2a" -b "/sysfile_b/log2b"
pdlogadpf -d sys -g log3 -a "/sysfile_a/log3a" -b "/sysfile_b/log3b"
pdlogadpf -d sys -g log4 -a "/sysfile_a/log4a" -b "/sysfile_b/log4b"
pdlogadfg -d spd -g sync1 ONL                                          5
pdlogadfg -d spd -g sync2 ONL
pdlogadfg -d spd -g sync3 ONL
pdlogadfg -d spd -g sync4 ONL
pdlogadpf -d spd -g sync1 -a "/sysfile_a/sync1"
pdlogadpf -d spd -g sync2 -a "/sysfile_b/sync2"
pdlogadpf -d spd -g sync3 -a "/sysfile_a/sync3"
pdlogadpf -d spd -g sync4 -a "/sysfile_b/sync4"
```

〔説明〕

1. システムログファイルの二重化を指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
4. システムログファイルの構成を指定します。
5. シンクポイントダンプファイルの構成を指定します。

(4) UAP 環境定義

```
set pd_uap_wait = Y                                    1
pdlbuffer -a localbuf1 -r RDAREA10 -n 1000 -p 16       2
pdlbuffer -a localbuf2 -r RDAREA11,RDAREA12 -n 1000    3
pdlbuffer -a localbuf3 -i USER01.INDX01 -n 1000        4
```

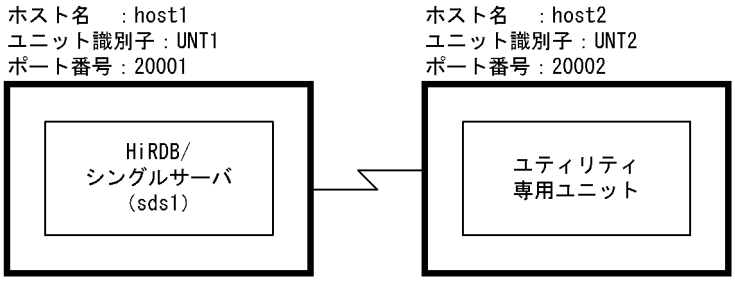
〔説明〕

1. ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。
2. ユーザ用 RD エリア（RDAREA10）にローカルバッファを割り当てます。
3. ユーザ用 RD エリア（RDAREA11，RDAREA12）にローカルバッファを割り当てます。
4. インデクス（INDX01）にローカルバッファを割り当てます。

付録 B.2 HiRDB/シングルサーバの場合：ユティリティ専用ユニットあり

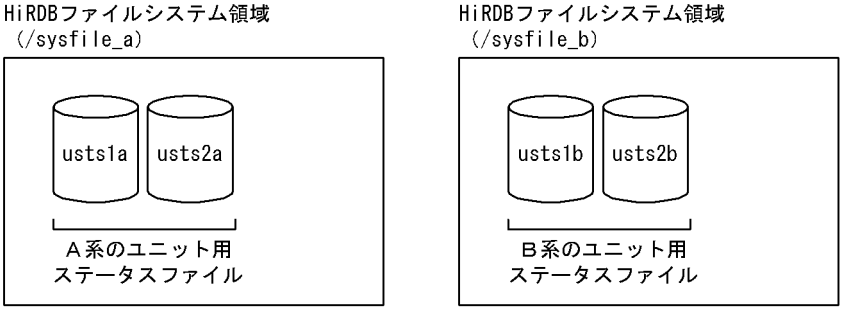
例題１で定義した HiRDB/シングルサーバにユティリティ専用ユニットを設置します。HiRDB/シングルサーバ及びユティリティ専用ユニットのシステム定義を作成します。システム構成は次のとおりです。

●ユニット構成



注　（　）内はサーバ名を表しています。

●ユティリティ専用ユニットのシステムファイルの構成



(1)　HiRDB/シングルサーバのシステム共通定義

```
          :
pdunit -x host2 -u UNT2 -d "/HiRDB" -p 20002
          :
```

〔説明〕

例題１で作成したシステム共通定義を修正します（pdunit オペランドを追加します）。

ユティリティ専用ユニットのユニット構成を定義した pdunit オペランドを追加します。

(2)　ユティリティ専用ユニットのシステム共通定義

```
set pd_system_id = PDB2                                    1
set pd_name_port = 20002                                   2
pdunit -x host1 -u UNT1 -d "/HiRDB" -p 20001               3
pdunit -x host2 -u UNT2 -d "/HiRDB"                        4
pdstart -t SDS -s sds1 -u UNT1                             5
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. HiRDB/シングルサーバのユニットの構成を指定します。

4. ユティリティ専用ユニットのユニットの構成を指定します。

5. HiRDB/シングルサーバのサーバの構成を指定します。

(3)　ユティリティ専用ユニットのユニット制御情報定義

```
set pd_unit_id = UNT2                                                    1
set pd_syssts_file_name_1 = ”untsts1”,”/sysfile_a/usts1a”,¥              2
                                    ”/sysfile_b/usts1b”
set pd_syssts_file_name_2 = ”untsts2”,”/sysfile_a/usts2a”,¥
                                    ”/sysfile_b/usts2b”
```

〔説明〕

1. ユニット識別子を指定します。

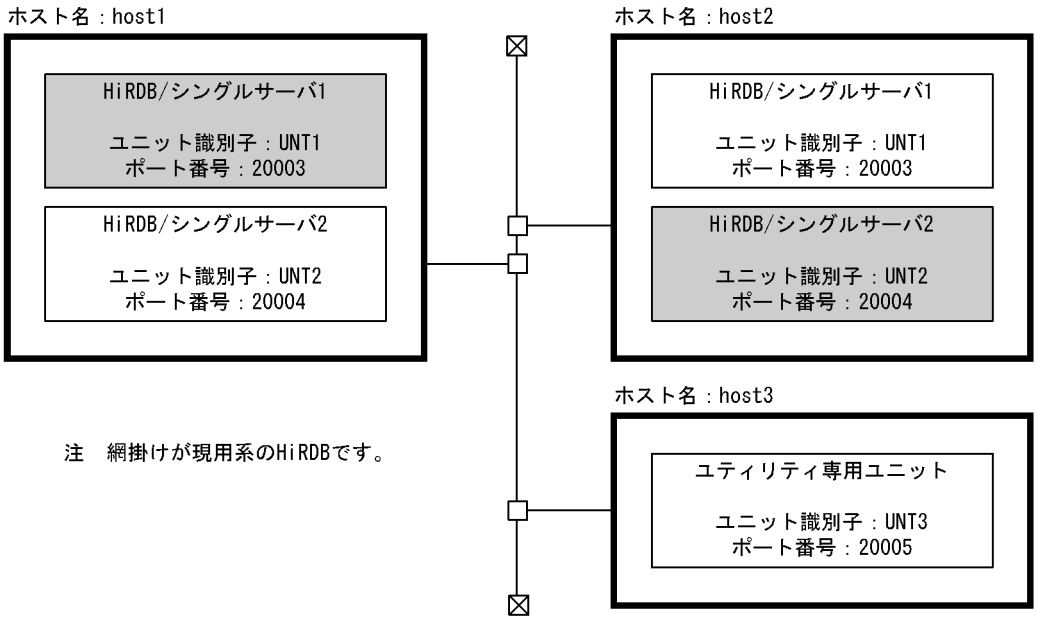
2. ユニット用ステータスファイルの構成を指定します。

付録 B.3　HiRDB/シングルサーバの場合：系切り替え機能使用時

HiRDB/シングルサーバのシステム定義を作成します。システム構成は次のとおりです。

- HiRDB/シングルサーバ 1 と HiRDB/シングルサーバ 2 の相互系切り替え構成です。
- HiRDB/シングルサーバ 1 と HiRDB/シングルサーバ 2 が使用するユティリティ専用ユニットを設置します。
- IP アドレスを引き継がない系切り替え機能とします。

なお，定義例はシステム共通定義及びユニット制御情報定義の関連のあるオペランドについてだけ説明しています。



(1)　HiRDB/シングルサーバ 1 のシステム共通定義

```
set pd_system_id = SDS1                                          1
set pd_name_port = 20003                                         2
          :
          :
set pd_ha = use                                                  3
set pd_ha_ipaddr_inherit = N                                     4
          :
          :
pdunit -x host1 -u UNT1 -d "/HiRDB1" -c host2                    5
pdunit -x host3 -u UNT3 -d "/HiRDB3" -p 20005                    6
pdstart -t SDS -s sds01 -u UNT1                                  7
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. 系切り替え機能を使用することを指定します。
4. IP アドレスを引き継がないことを指定します。
5. HiRDB/シングルサーバ 1 のユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
6. ユティリティ専用ユニットの構成を指定します。
   -p：ポート番号を指定します。
7. HiRDB/シングルサーバ 1 のサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。

(2)　HiRDB/シングルサーバ 1 のユニット制御情報定義

```
set pd_unit_id = UNT1                                            1
          :
          :
set pd_hostname = host1                                          2
```

〔説明〕

1. ユニット識別子を指定します。
2. 現用系のホスト名を指定します。

(3)　HiRDB/シングルサーバ 2 のシステム共通定義

```
set pd_system_id = SDS2                                          1
set pd_name_port = 20004                                         2
          :
          :
set pd_ha = use                                                  3
set pd_ha_ipaddr_inherit = N                                     4
          :
          :
pdunit -x host2 -u UNT2 -d "/HiRDB2" -c host1                    5
pdunit -x host3 -u UNT3 -d "/HiRDB3" -p 20005                    6
pdstart -t SDS -s sds02 -u UNT2                                  7
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. 系切り替え機能を使用することを指定します。
4. IP アドレスを引き継がないことを指定します。
5. HiRDB/シングルサーバ 2 のユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
6. ユティリティ専用ユニットの構成を指定します。
   -p：ポート番号を指定します。
7. HiRDB/シングルサーバ 2 のサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。

(4)　HiRDB/シングルサーバ 2 のユニット制御情報定義

```
set pd_unit_id = UNT2                                            1
          :
          :
set pd_hostname = host2                                          2
```

〔説明〕

1. ユニット識別子を指定します。
2. 現用系のホスト名を指定します。

(5)　ユティリティ専用ユニットのシステム共通定義

```
set pd_system_id = UTL1                                          1
set pd_name_port = 20005                                         2
         :
         :
set pd_ha = use                                                  3
set pd_ha_ipaddr_inherit = N                                     4
         :
         :
pdunit -x host1 -u UNT1 -d "/HiRDB1" -c host2 -p 20003           5
pdunit -x host2 -u UNT2 -d "/HiRDB2" -c host1 -p 20004           6
pdunit -x host3 -u UNT3 -d "/HiRDB3"                             7
pdstart -t SDS -s sds01 -u UNT1                                  8
pdstart -t SDS -s sds02 -u UNT2                                  9
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. 系切り替え機能を使用することを指定します。
4. IP アドレスを引き継がないことを指定します。
5. HiRDB/シングルサーバ 1 のユニット構成を指定します。
   -x：ホスト名を指定します。
   -u：ユニット識別子を指定します。
   -d：HiRDB 運用ディレクトリ名を指定します。
   -c：予備系のホスト名を指定します。
   -p：ポート番号を指定します。
6. HiRDB/シングルサーバ 2 のユニット構成を指定します。
7. ユティリティ専用ユニットの構成を指定します。
8. HiRDB/シングルサーバ 1 のサーバ構成を指定します。
   -t：サーバの種別を指定します。
   -s：サーバ名を指定します。
   -u：ユニット識別子を指定します。
9. HiRDB/シングルサーバ 2 のサーバ構成を指定します。

(6)　ユティリティ専用ユニットのユニット制御情報定義

```
set pd_unit_id = UNT3                                            1
         :
         :
set pd_hostname = host3                                          2
```

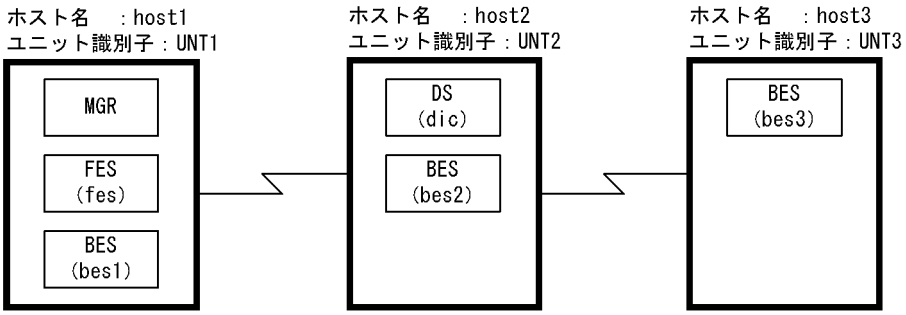
〔説明〕

1. ユニット識別子を指定します。
2. ホスト名を指定します。

付録 B.4　HiRDB/パラレルサーバの場合

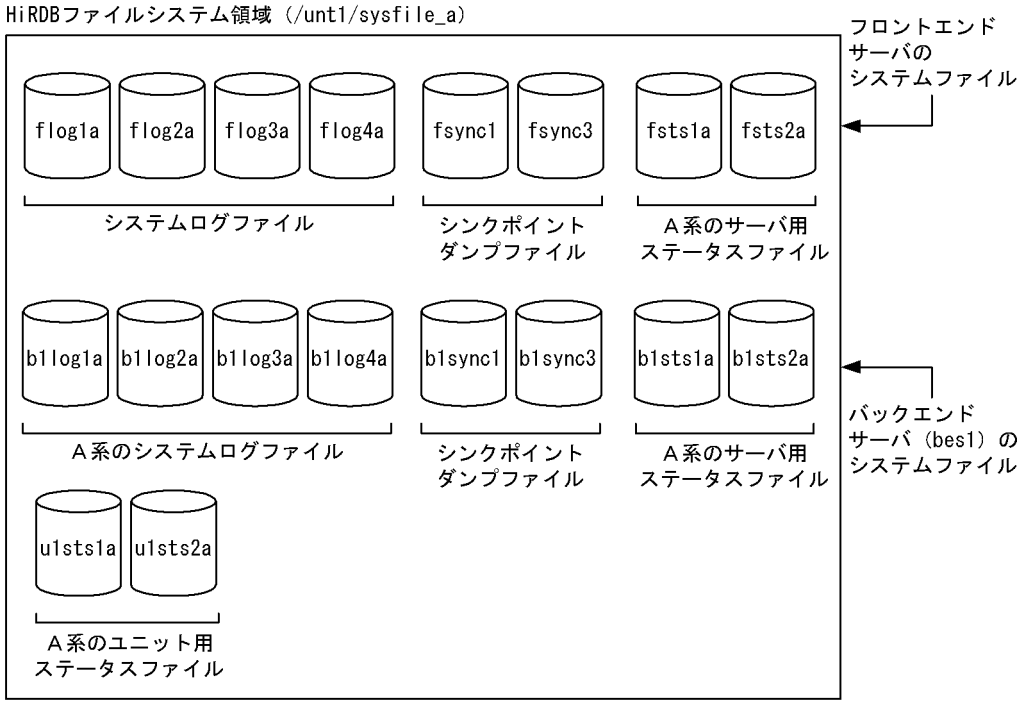
HiRDB/パラレルサーバのシステム定義を作成します。システム構成は次のとおりです。

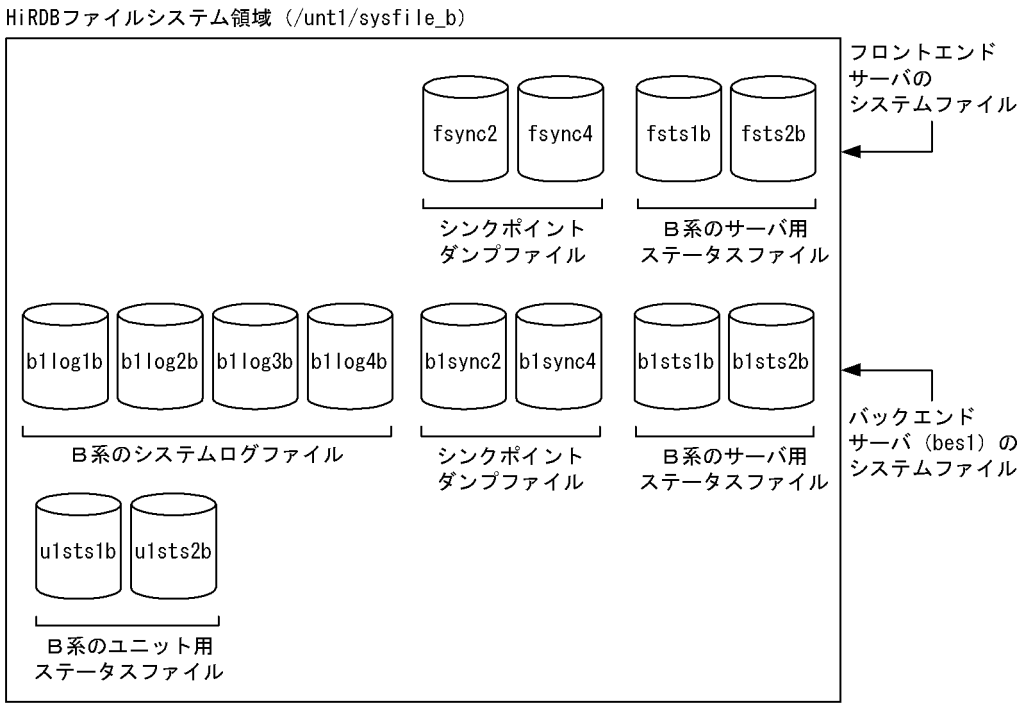
●ユニット構成



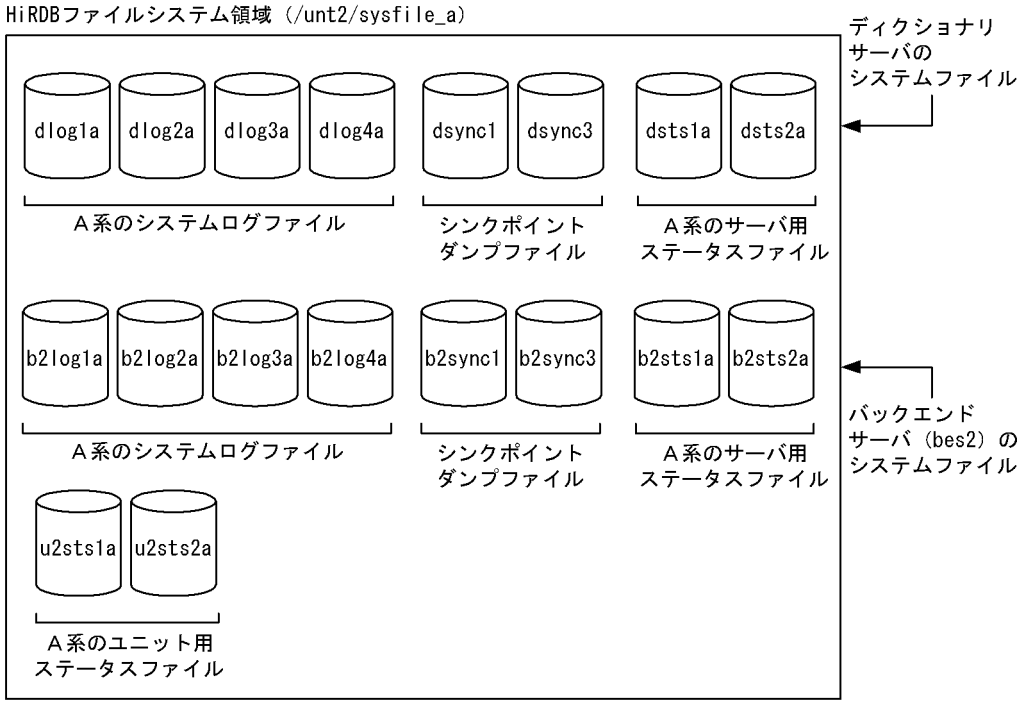
注　（　）内はサーバ名を表しています。

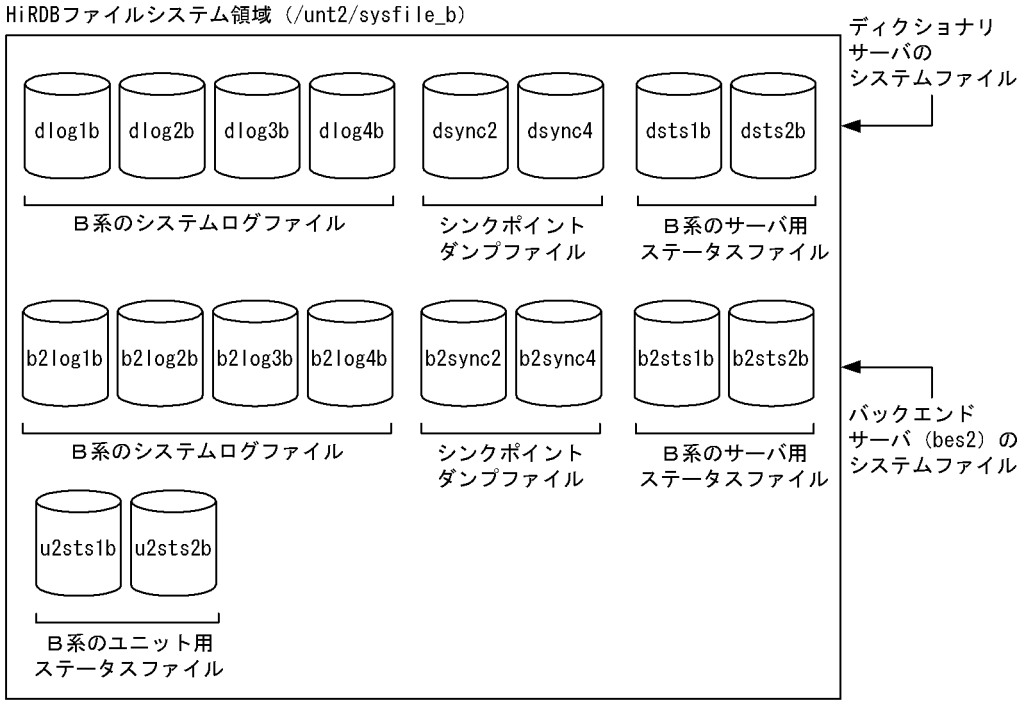
●UNT1 のシステムファイルの構成



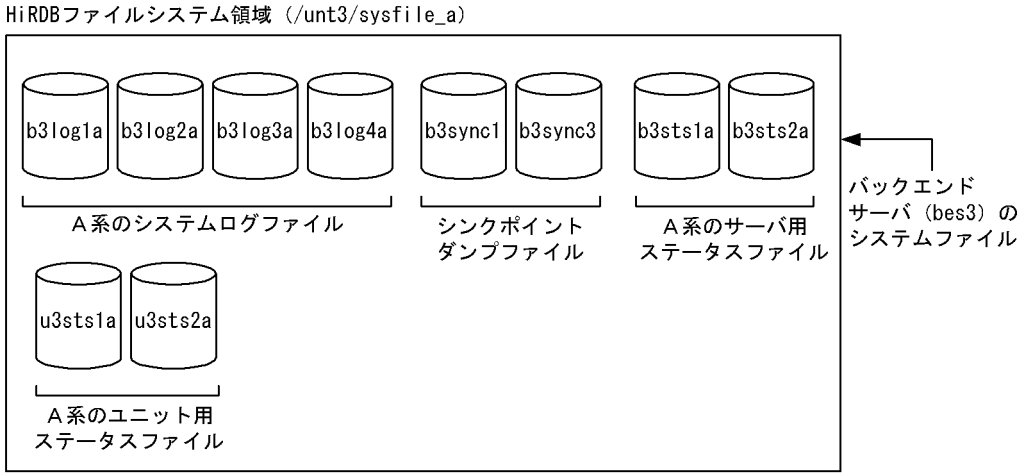


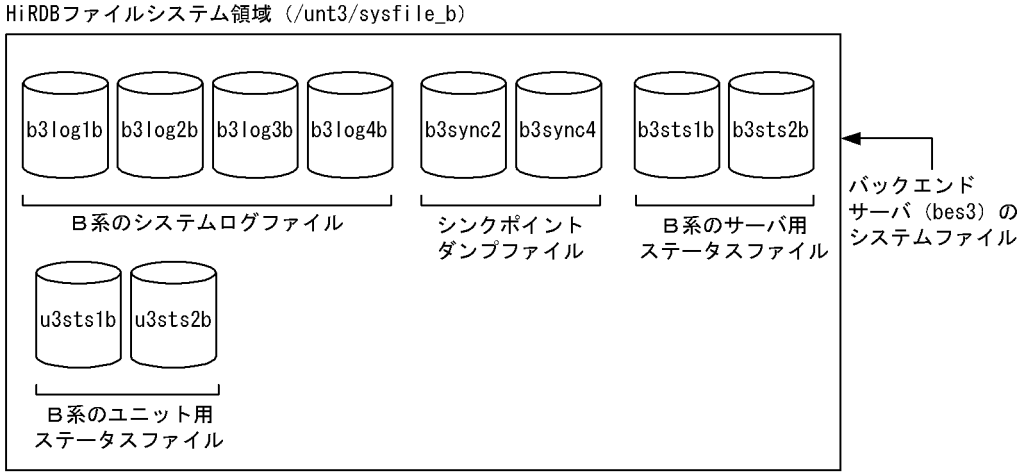
●UNT2 のシステムファイルの構成





●UNT3 のシステムファイルの構成





(1)　システム共通定義

```
set pd_system_id = PDB1                                                    1
set pd_name_port = 20001                                                   2
set pd_master_file_name = ”/rdarea/mast/mast01”                            3
set pd_max_users = 100                                                     4
pdunit -x host1 -u UNT1 -d ”/HiRDB”                                        5
pdunit -x host2 -u UNT2 -d ”/HiRDB”
pdunit -x host3 -u UNT3 -d ”/HiRDB”
pdstart -t MGR -u UNT1                                                     6
pdstart -t FES -s fes -u UNT1
pdstart -t DIC -s dic -u UNT2
pdstart -t BES -s bes1 -u UNT1
pdstart -t BES -s bes2 -u UNT2
pdstart -t BES -s bes3 -u UNT3
pdbuffer -a gbuf01 -r RDMAST,RDDIC,RDDIR -n 1000                           7
pdbuffer -a gbuf02 -r RDAREA1,RDAREA2,RDAREA3 -n 1000                      8
pdbuffer -a gbuf03 -r RDAREA4,RDAREA5,RDAREA6 -n 1000                      9
pdbuffer -a gbuf04 -o -n 1000                                              10
```

〔説明〕

1. HiRDB 識別子を指定します。
2. HiRDB のポート番号を指定します。
3. マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。
4. 最大同時接続数を指定します。
5. HiRDB/パラレルサーバの全ユニットの構成を指定します。

   -x：ホスト名を指定します。

   -u：ユニット識別子を指定します。

   -d：HiRDB 運用ディレクトリ名を指定します。
6. HiRDB/パラレルサーバの全サーバの構成を指定します。

   -t：サーバの種別を指定します。

   -s：サーバ名を指定します。

   -u：ユニット識別子を指定します。
7. マスタディレクトリ用 RD エリア，データディクショナリ用 RD エリア，及びデータディレクトリ用 RD エリアにグローバルバッファを割り当てます。
8. ユーザ用 RD エリア（RDAREA1～RDAREA3）にグローバルバッファを割り当てます。
9. ユーザ用 RD エリア（RDAREA4～RDAREA6）にグローバルバッファを割り当てます。
10. そのほかの RD エリアにグローバルバッファを割り当てます。

(2)　UNT1 のユニット制御情報定義

```
set pd_unit_id = UNT1                                                      1
set pd_syssts_file_name_1 = ”u1sts1”,”/unt1/sysfile_a/u1sts1a”,¥           2
```

```
                                    "/unt1/sysfile_b/u1sts1b"
set pd_syssts_file_name_2 = "u1sts2","/unt1/sysfile_a/u1sts2a",¥
                                    "/unt1/sysfile_b/u1sts2b"
```

〔説明〕

1. ユニット識別子を指定します。
2. ユニット用ステータスファイルの構成を指定します。

(3) UNT2 のユニット制御情報定義

```
set pd_unit_id = UNT2                                                    1
set pd_syssts_file_name_1 = "u2sts1","/unt2/sysfile_a/u2sts1a",¥         2
                                    "/unt2/sysfile_b/u2sts1b"
set pd_syssts_file_name_2 = "u2sts2","/unt2/sysfile_a/u2sts2a",¥
                                    "/unt2/sysfile_b/u2sts2b"
```

〔説明〕

1. ユニット識別子を指定します。
2. ユニット用ステータスファイルの構成を指定します。

(4) UNT3 のユニット制御情報定義

```
set pd_unit_id = UNT3                                                    1
set pd_syssts_file_name_1 = "u3sts1","/unt3/sysfile_a/u3sts1a",¥         2
                                    "/unt3/sysfile_b/u3sts1b"
set pd_syssts_file_name_2 = "u3sts2","/unt3/sysfile_a/u3sts2a",¥
                                    "/unt3/sysfile_b/u3sts2b"
```

〔説明〕

1. ユニット識別子を指定します。
2. ユニット用ステータスファイルの構成を指定します。

(5) フロントエンドサーバ定義

```
set pd_log_dual = N                                                      1
set pd_sts_file_name_1 = "fsts1","/unt1/sysfile_a/fsts1a",¥              2
                                  "/unt1/sysfile_b/fsts1b"
set pd_sts_file_name_2 = "fsts2","/unt1/sysfile_a/fsts2a",¥
                                  "/unt1/sysfile_b/fsts2b"
pdlogadfg -d sys -g flog1 ONL                                            3
pdlogadfg -d sys -g flog2 ONL
pdlogadfg -d sys -g flog3 ONL
pdlogadfg -d sys -g flog4 ONL
pdlogadpf -d sys -g flog1 -a "/unt1/sysfile_a/flog1a"
pdlogadpf -d sys -g flog2 -a "/unt1/sysfile_a/flog2a"
pdlogadpf -d sys -g flog3 -a "/unt1/sysfile_a/flog3a"
pdlogadpf -d sys -g flog4 -a "/unt1/sysfile_a/flog4a"
```

```
pdlogadfg -d spd -g fsync1 ONL                                          4
pdlogadfg -d spd -g fsync2 ONL
pdlogadfg -d spd -g fsync3 ONL
pdlogadfg -d spd -g fsync4 ONL
pdlogadpf -d spd -g fsync1 -a "/unt1/sysfile_a/fsync1"
pdlogadpf -d spd -g fsync2 -a "/unt1/sysfile_b/fsync2"
pdlogadpf -d spd -g fsync3 -a "/unt1/sysfile_a/fsync3"
pdlogadpf -d spd -g fsync4 -a "/unt1/sysfile_b/fsync4"
```

〔説明〕

1. システムログファイルを二重化しないことを指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. システムログファイルの構成を指定します。
4. シンクポイントダンプファイルの構成を指定します。

(6) ディクショナリサーバ定義

```
set pd_log_dual = Y                                                     1
set pd_sts_file_name_1 = "dsts1","/unt2/sysfile_a/dsts1a",¥             2
                         "/unt2/sysfile_b/dsts1b"
set pd_sts_file_name_2 = "dsts2","/unt2/sysfile_a/dsts2a",¥
                         "/unt2/sysfile_b/dsts2b"
pdwork -v "/unt2/work01","/unt2/work02"                                 3
pdlogadfg -d sys -g dlog1 ONL                                           4
pdlogadfg -d sys -g dlog2 ONL
pdlogadfg -d sys -g dlog3 ONL
pdlogadfg -d sys -g dlog4 ONL
pdlogadpf -d sys -g dlog1 -a "/unt2/sysfile_a/dlog1a"¥
                          -b "/unt2/sysfile_b/dlog1b"
pdlogadpf -d sys -g dlog2 -a "/unt2/sysfile_a/dlog2a"¥
                          -b "/unt2/sysfile_b/dlog2b"
pdlogadpf -d sys -g dlog3 -a "/unt2/sysfile_a/dlog3a"¥
                          -b "/unt2/sysfile_b/dlog3b"
pdlogadpf -d sys -g dlog4 -a "/unt2/sysfile_a/dlog4a"¥
                          -b "/unt2/sysfile_b/dlog4b"
pdlogadfg -d spd -g dsync1 ONL                                          5
pdlogadfg -d spd -g dsync2 ONL
pdlogadfg -d spd -g dsync3 ONL
pdlogadfg -d spd -g dsync4 ONL
pdlogadpf -d spd -g dsync1 -a "/unt2/sysfile_a/dsync1"
pdlogadpf -d spd -g dsync2 -a "/unt2/sysfile_b/dsync2"
pdlogadpf -d spd -g dsync3 -a "/unt2/sysfile_a/dsync3"
pdlogadpf -d spd -g dsync4 -a "/unt2/sysfile_b/dsync4"
```

〔説明〕

1. システムログファイルの二重化を指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
4. システムログファイルの構成を指定します。

5. シンクポイントダンプファイルの構成を指定します。

(7) bes1 のバックエンドサーバ定義

```
set pd_log_dual = Y                                                          1
set pd_sts_file_name_1 = ”b1sts1”,”/unt1/sysfile_a/b1sts1a”,¥                2
                                  ”/unt1/sysfile_b/b1sts1b”
set pd_sts_file_name_2 = ”b1sts2”,”/unt1/sysfile_a/b1sts2a”,¥
                                  ”/unt1/sysfile_b/b1sts2b”
pdwork -v ”/unt1/work01”,”/unt1/work02”                                      3
pdlogadfg -d sys -g b1log1 ONL                                               4
pdlogadfg -d sys -g b1log2 ONL
pdlogadfg -d sys -g b1log3 ONL
pdlogadfg -d sys -g b1log4 ONL
pdlogadpf -d sys -g b1log1 -a ”/unt1/sysfile_a/b1log1a”¥
                           -b ”/unt1/sysfile_b/b1log1b”
pdlogadpf -d sys -g b1log2 -a ”/unt1/sysfile_a/b1log2a”¥
                           -b ”/unt1/sysfile_b/b1log2b”
pdlogadpf -d sys -g b1log3 -a ”/unt1/sysfile_a/b1log3a”¥
                           -b ”/unt1/sysfile_b/b1log3b”
pdlogadpf -d sys -g b1log4 -a ”/unt1/sysfile_a/b1log4a”¥
                           -b ”/unt1/sysfile_b/b1log4b”
pdlogadfg -d spd -g b1sync1 ONL                                              5
pdlogadfg -d spd -g b1sync2 ONL
pdlogadfg -d spd -g b1sync3 ONL
pdlogadfg -d spd -g b1sync4 ONL
pdlogadpf -d spd -g b1sync1 -a ”/unt1/sysfile_a/b1sync1”
pdlogadpf -d spd -g b1sync2 -a ”/unt1/sysfile_b/b1sync2”
pdlogadpf -d spd -g b1sync3 -a ”/unt1/sysfile_a/b1sync3”
pdlogadpf -d spd -g b1sync4 -a ”/unt1/sysfile_b/b1sync4”
```

〔説明〕

1. システムログファイルの二重化を指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
4. システムログファイルの構成を指定します。
5. シンクポイントダンプファイルの構成を指定します。

(8) bes2 のバックエンドサーバ定義

```
set pd_log_dual = Y                                                          1
set pd_sts_file_name_1 = ”b2sts1”,”/unt2/sysfile_a/b2sts1a”,¥                2
                                  ”/unt2/sysfile_b/b2sts1b”
set pd_sts_file_name_2 = ”b2sts2”,”/unt2/sysfile_a/b2sts2a”,¥
                                  ”/unt2/sysfile_b/b2sts2b”
pdwork -v ”/unt2/work03”,”/unt2/work04”                                      3
pdlogadfg -d sys -g b2log1 ONL                                               4
pdlogadfg -d sys -g b2log2 ONL
pdlogadfg -d sys -g b2log3 ONL
pdlogadfg -d sys -g b2log4 ONL
```

```
pdlogadpf -d sys -g b2log1 -a "/unt2/sysfile_a/b2log1a"¥
                           -b "/unt2/sysfile_b/b2log1b"
pdlogadpf -d sys -g b2log2 -a "/unt2/sysfile_a/b2log2a"¥
                           -b "/unt2/sysfile_b/b2log2b"
pdlogadpf -d sys -g b2log3 -a "/unt2/sysfile_a/b2log3a"¥
                           -b "/unt2/sysfile_b/b2log3b"
pdlogadpf -d sys -g b2log4 -a "/unt2/sysfile_a/b2log4a"¥
                           -b "/unt2/sysfile_b/b2log4b"
pdlogadfg -d spd -g b2sync1 ONL                                              5
pdlogadfg -d spd -g b2sync2 ONL
pdlogadfg -d spd -g b2sync3 ONL
pdlogadfg -d spd -g b2sync4 ONL
pdlogadpf -d spd -g b2sync1 -a "/unt2/sysfile_a/b2sync1"
pdlogadpf -d spd -g b2sync2 -a "/unt2/sysfile_b/b2sync2"
pdlogadpf -d spd -g b2sync3 -a "/unt2/sysfile_a/b2sync3"
pdlogadpf -d spd -g b2sync4 -a "/unt2/sysfile_b/b2sync4"
```

〔説明〕

1. システムログファイルの二重化を指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
4. システムログファイルの構成を指定します。
5. シンクポイントダンプファイルの構成を指定します。

(9) bes3 のバックエンドサーバ定義

```
set pd_log_dual = Y                                                          1
set pd_sts_file_name_1 = "b3sts1","/unt3/sysfile_a/b3sts1a",¥                2
                                  "/unt3/sysfile_b/b3sts1b"
set pd_sts_file_name_2 = "b3sts2","/unt3/sysfile_a/b3sts2a",¥
                                  "/unt3/sysfile_b/b3sts2b"
pdwork -v "/unt3/work01","/unt3/work02"                                      3
pdlogadfg -d sys -g b3log1 ONL                                               4
pdlogadfg -d sys -g b3log2 ONL
pdlogadfg -d sys -g b3log3 ONL
pdlogadfg -d sys -g b3log4 ONL
pdlogadpf -d sys -g b3log1 -a "/unt3/sysfile_a/b3log1a"¥
                           -b "/unt3/sysfile_b/b3log1b"
pdlogadpf -d sys -g b3log2 -a "/unt3/sysfile_a/b3log2a"¥
                           -b "/unt3/sysfile_b/b3log2b"
pdlogadpf -d sys -g b3log3 -a "/unt3/sysfile_a/b3log3a"¥
                           -b "/unt3/sysfile_b/b3log3b"
pdlogadpf -d sys -g b3log4 -a "/unt3/sysfile_a/b3log4a"¥
                           -b "/unt3/sysfile_b/b3log4b"
pdlogadfg -d spd -g b3sync1 ONL                                              5
pdlogadfg -d spd -g b3sync2 ONL
pdlogadfg -d spd -g b3sync3 ONL
pdlogadfg -d spd -g b3sync4 ONL
pdlogadpf -d spd -g b3sync1 -a "/unt3/sysfile_a/b3sync1"
pdlogadpf -d spd -g b3sync2 -a "/unt3/sysfile_b/b3sync2"
```

```
pdlogadpf -d spd -g b3sync3 -a "/unt3/sysfile_a/b3sync3"
pdlogadpf -d spd -g b3sync4 -a "/unt3/sysfile_b/b3sync4"
```

〔説明〕

1. システムログファイルの二重化を指定します。
2. サーバ用ステータスファイルの構成を指定します。
3. 作業表用ファイル用の HiRDB ファイルシステム領域を指定します。
4. システムログファイルの構成を指定します。
5. シンクポイントダンプファイルの構成を指定します。

(10)　UAP 環境定義

```
set pd_uap_wait = Y                                          1
pdlbuffer -a localbuf1 -r RDAREA10 -n 1000 -p 16             2
pdlbuffer -a localbuf2 -r RDAREA11,RDAREA12 -n 1000          3
pdlbuffer -a localbuf3 -i USER01.INDX01 -n 1000              4
```

〔説明〕

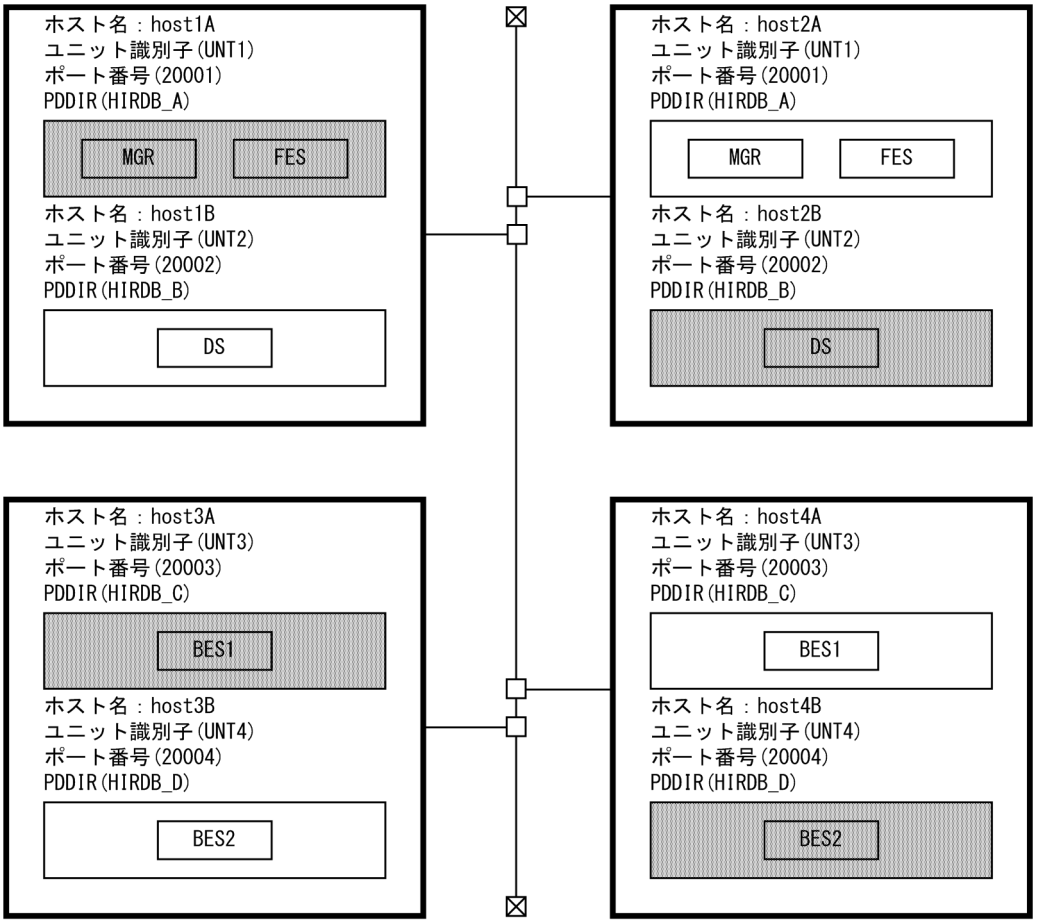
1. ローカルバッファを使用してアクセスする RD エリア又はインデクスがほかのユーザに使用されている場合の UAP の動作を指定します。
2. ユーザ用 RD エリア（RDAREA10）にローカルバッファを割り当てます。
3. ユーザ用 RD エリア（RDAREA11，RDAREA12）にローカルバッファを割り当てます。
4. インデクス（INDX01）にローカルバッファを割り当てます。

付録 B.5　HiRDB/パラレルサーバの場合：スタンバイ型系切り替え機能使用時

HiRDB/パラレルサーバのシステム定義を作成します。システム構成は次のとおりです。

- 相互系切り替え構成です。
- IP アドレスを引き継がない構成です。
- 全ユニットに高速系切り替え機能を適用します。

なお，定義例はシステム共通定義及びユニット制御情報定義の関連のあるオペランドについてだけ説明しています。



注　網掛けが現用系のユニットです。

ポイント

- IP アドレスを引き継がない場合はサーバマシンからの独立性を図るために，pdunit オペランドの-x 及び-c オプションに標準ホスト名ではなく，エイリアス IP アドレスのホスト名を指定してください。
- IP アドレスを引き継がない相互系切り替え構成の場合は，pdunit オペランドの-x 及び-c オプションに指定するホスト名は重複できません。(1)の pdunit オペランドの指定例を参考にしてください。

(1)　システム共通定義

```
        :
set pd_name_port = 20001                                          1
        :
set pd_ha = use                                                   2
set pd_ha_ipaddr_inherit = N
set pd_ha_switch_timeout = Y
set pd_ha_transaction = queuing
set pd_ha_trn_queuing_wait_time = 240
set pd_ha_trn_restart_retry_time = 90
        :
pdunit -x host1A -u UNT1 -d "/HiRDB_A" -c host2A -p 20001         3
pdunit -x host2B -u UNT2 -d "/HiRDB_B" -c host1B -p 20002
pdunit -x host3A -u UNT3 -d "/HiRDB_C" -c host4A -p 20003
pdunit -x host4B -u UNT4 -d "/HiRDB_D" -c host3B -p 20004
pdstart -t MGR -u UNT1                                            4
```

```
pdstart -t FES -s fes1 -u UNT1
pdstart -t DIC -s dic  -u UNT2
pdstart -t BES -s bes1 -u UNT3
pdstart -t BES -s bes2 -u UNT4
         :
```

〔説明〕

1. HiRDB のポート番号を指定します。
2. 系切り替え機能に関するオペランドの指定です。
3. HiRDB/パラレルサーバのユニット構成を指定します。

   -x：ホスト名を指定します。

   -u：ユニット識別子を指定します。

   -d：HiRDB 運用ディレクトリ名を指定します。

   -c：予備系のホスト名を指定します。

   -p：ユニットのポート番号を指定します。
4. HiRDB/パラレルサーバのサーバ構成を指定します。

   -t：サーバの種別を指定します。

   -s：サーバ名を指定します。

   -u：ユニット識別子を指定します。

(2)　UNT1～UNT4 のユニット制御情報定義

```
         :
set pd_ha_acttype = server                          1
set pd_ha_agent = standbyunit                      2
         :
```

〔説明〕

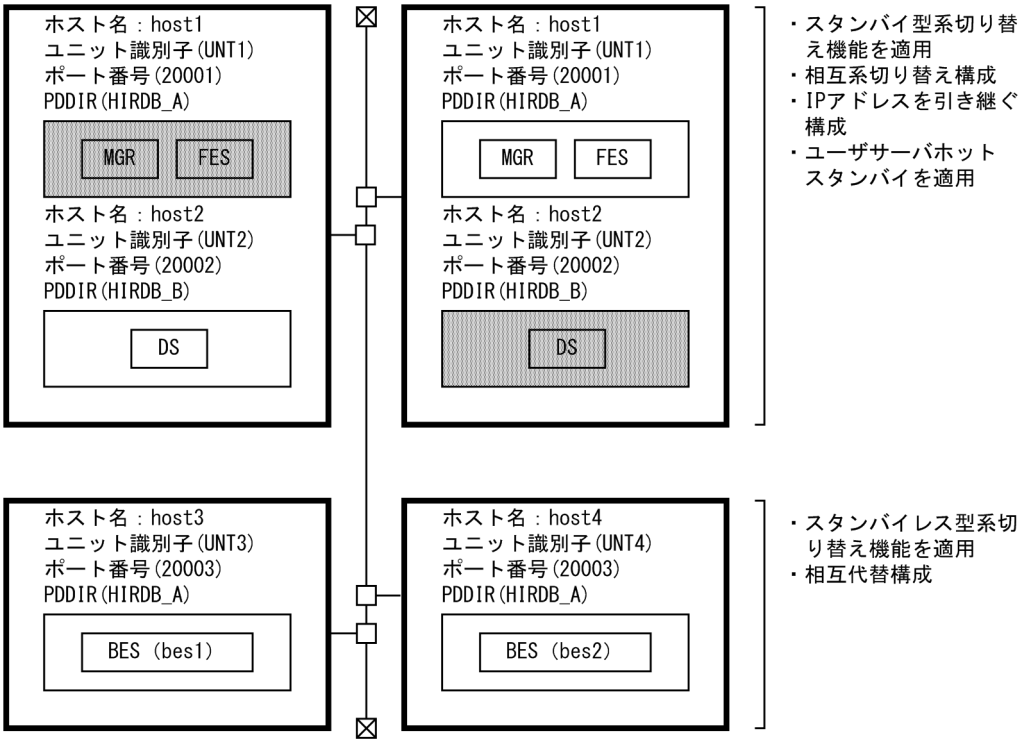
1. 系切り替え機能をサーバモードで運用することを指定します。
2. 高速系切り替え機能を使用することを指定します。

付録 B.6　HiRDB/パラレルサーバの場合：1：1 スタンバイレス型系切り替え機能使用時

HiRDB/パラレルサーバのシステム定義を作成します。システム構成は次のとおりです。

- スタンバイ型とスタンバイレス型を混合した構成です。
- スタンバイ型系切り替え機能を適用するユニットは，相互系切り替え構成で IP アドレスを引き継ぐ構成です。また，ユーザサーバホットスタンバイを適用します。
- 1：1 スタンバイレス型系切り替え機能を適用するユニットは相互代替構成です。

なお，定義例はシステム共通定義及びユニット制御情報定義の関連のあるオペランドについてだけ説明しています。



注

- 網掛けが現用系のユニットです。
- （　）内にはサーバ名を表記しています。
- バックエンドサーバ（bes1）は正規 BES であると同時に，バックエンドサーバ（bes2）の代替 BES です。
- バックエンドサーバ（bes2）は正規 BES であると同時に，バックエンドサーバ（bes1）の代替 BES です。

(1)　システム共通定義

```
        :
set pd_name_port = 20001                                    1
        :
set pd_ha = use                                             2
set pd_ha_ipaddr_inherit = Y
set pd_ha_switch_timeout = Y
set pd_ha_transaction = queuing
set pd_ha_trn_queuing_wait_time = 240
set pd_ha_trn_restart_retry_time = 90
        :
pdunit -x host1 -u UNT1 -d "/HiRDB_A" -p 20001              3
pdunit -x host2 -u UNT2 -d "/HiRDB_B" -p 20002
pdunit -x host3 -u UNT3 -d "/HiRDB_A" -p 20003
pdunit -x host4 -u UNT4 -d "/HiRDB_A" -p 20003
```

```
pdstart -t MGR -u UNT1                                          4
pdstart -t FES -s fes1 -u UNT1
pdstart -t DIC -s dic  -u UNT2
pdstart -t BES -s bes1 -u UNT3 -c bes2                          5
pdstart -t BES -s bes2 -u UNT4 -c bes1
         :
```

〔説明〕

1. HiRDB のポート番号を指定します。
2. 系切り替え機能に関するオペランドの指定です。
3. HiRDB/パラレルサーバのユニット構成を指定します。

   -x：ホスト名を指定します。

   -u：ユニット識別子を指定します。

   -d：HiRDB 運用ディレクトリ名を指定します。

   -p：ユニットのポート番号を指定します。
4. HiRDB/パラレルサーバのサーバ構成を指定します。

   -t：サーバの種別を指定します。

   -s：サーバ名を指定します。

   -u：ユニット識別子を指定します。
5. HiRDB/パラレルサーバのサーバ構成を指定します。

   -t：サーバの種別を指定します。

   -s：正規 BES 名を指定します。

   -u：ユニット識別子を指定します。

   -c：代替 BES 名を指定します。

(2)　UNT1～UNT2 のユニット制御情報定義

```
         :
set pd_ha_acttype = server                                       1
set pd_ha_server_process_standby = Y                            2
         :
```

〔説明〕

1. 系切り替え機能をサーバモードで運用することを指定します。
2. ユーザサーバホットスタンバイを使用することを指定します。

(3)　UNT3～UNT4 のユニット制御情報定義

```
         :
set pd_ha_acttype = server                                       1
```

```
set pd_ha_agent = server                                        2
        :
```

〔説明〕

1. 系切り替え機能をサーバモードで運用することを指定します。
2. 1：1 スタンバイレス型系切り替え機能を使用することを指定します。

付録 C　オペランド指定値の見積もり式

ここでは，システム定義の各オペランドの指定値の見積もり式について説明します。説明する項目は次のとおりです。

1. 統計ログファイル（pd_stj_file_size）の見積もり式
2. SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式
3. ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式
4. 表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式
5. 1 サーバ当たりの UNTIL DISCONNECT 指定排他の表と RD エリアの総和（pd_lck_until_disconnect_cnt）の見積もり式
6. ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式

付録 C.1　統計ログファイル（pd_stj_file_size）の見積もり式

出力される統計ログ量は統計情報ごとに異なります。統計情報ごとの統計ログ量を(1)以降に示します。pd_stj_file_size オペランドには，(1)以降で求めた統計ログ量の合計値以上の値を指定してください。

注意事項

計算式の単位はバイトのため，合計後，キロバイトに変換してください。

(1)　システムの稼働に関する統計情報（sys）

pdstbegin オペランド又は pdstbegin コマンドに指定したオプションによって異なります。

pd_statistics オペランドに A，又は Y を指定した場合は，次の「-a 及び-s オプションを指定しなかった場合」の見積もり式を用いて計算してください。

-a 及び-s オプションを指定しなかった場合

統計ログ量＝ 2412×（↓ａ÷ｂ↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

-a オプションを指定した場合

統計ログ量＝（1836×ｃ＋ 2412）×（↓ａ÷ｂ↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

-s オプションを指定した場合

統計ログ量＝（1836×ｄ＋ 2412）×（↓ａ÷ｂ↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

ａ：統計情報出力時間　（単位：分）

b：統計情報出力間隔 （単位：分）

pdstbegin オペランド又は pdstbegin コマンドの-m オプションで指定した時間間隔です。

pd_statistics オペランドに A を指定した場合は 1，Y を指定した場合は 10 となります。

c：ユニット内のサーバ数

システムマネジャは数に入れません。

d：pdstbegin オペランド又は pdstbegin コマンドの-s オプションで指定したサーバ数

(2) UAP に関する統計情報（uap）

〈計算式〉

統計ログ量＝ 1252×統計情報の取得中に実行する UAP の数 （単位：バイト）

実際に実行する UAP の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する UAP 数が不明確な場合，UAP 数は次の式から求められます。

- **UAP 数＝単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間**

例えば，30 分に 10 個の UAP を実行し，統計情報を 60 分ほど収集した場合は，10×60÷30=20 となります。

(3) SQL に関する統計情報（sql）

〈計算式〉

統計ログ量＝ 728×統計情報の取得中に実行する SQL の数※ （単位：バイト）

実際に実行する SQL の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL 数が不明確な場合，SQL 数は次の式から求められます。

SQL 数※＝ UAP が実行する SQL 文の平均的な数×

（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

注※

ストアドプロシジャ又はストアドファンクションを実行する場合，手続き又は関数中に記述する SQL 文数も数えてください。

(4) グローバルバッファに関する統計情報（buf）

〈計算式〉

統計ログ量＝ 412× a ×b （単位：バイト）

a：pdbuffer オペランドの指定数

b：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量
÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)
×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑
システムログ量については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(5) データベース操作に関する HiRDB ファイルの統計情報（fil）

〈計算式〉

統計ログ量＝ 428×ａ×ｂ （単位：バイト）

ａ：データベース初期設定ユティリティの制御文で指定した HiRDB ファイル数

ｂ：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量
÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)
×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑
システムログ量については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(6) デファードライト処理に関する統計情報（dfw）

〈計算式〉

統計ログ量＝ 384×ａ×2 ＋（384×ｂ×ｃ）÷（ｄ×ｅ×ｆ）（単位：バイト）

ａ：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量
÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)
×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑
システムログ量については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

ｂ：1 トランザクション当たりの平均更新ページ数

ｃ：統計情報時間内に実行するトランザクション数

ｄ：pdbuffer オペランドの−ｎオプションで指定したバッファ面数の総数

ｅ：全トランザクションに占めるデータベースを更新するトランザクションの比率

ｆ：pdbuffer オペランドで指定したデファードライトトリガ時の更新ページ出力比率

(7) インデクスに関する統計情報（idx）

〈計算式〉

統計ログ量＝ 3768×↑ａ÷128↑×ｂ （単位：バイト）

a：分割されたインデクスの総数

分割されたインデクスの総数は，定義系 SQL の各 CREATE INDEX の IN RD エリア名で指定する RD エリア名の数の合計値です。

b：統計情報取得中のシンクポイントダンプ取得回数

統計情報収集時間内に取得するシンクポイントダンプ取得回数は，次の計算式から算出してください。

↑統計情報収集時間内に出力するシステムログ量

÷(各サーバの定義で指定した pd_log_max_data_size オペランドの指定値の合計値÷3)

×各サーバの定義で指定した pd_log_sdinterval オペランドの指定値の合計値↑

システムログ量については，マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

(8) SQL 静的最適化に関する統計情報（sop）

〈計算式〉

統計ログ量＝ 92× a （単位：バイト）

a：統計情報の取得中に実行する SQL 文で SQL オブジェクト生成処理を行う数

ここでいう SQL 文とは，PREPARE 文，EXECUTE IMMEDIATE 文，及び静的 SQL 文のことです。

実際に SQL オブジェクト生成処理数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL オブジェクト生成処理数が不明確な場合，SQL オブジェクト生成処理数は次の式から求められます。

SQL オブジェクト生成処理数＝

UAP が実行する SQL 文の平均的な数×SQL オブジェクトキャッシュのミスヒット率×（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

SQL オブジェクトキャッシュのミスヒット率はシステムに関する統計情報で調べてください。

(9) SQL 動的最適化に関する統計情報（dop）

〈計算式〉

統計ログ量＝

$$\sum_{i=1}^{a}\left\{\begin{array}{l}28+16\times\text{SQL中の表数}\times 2\\ +12\times\left(1+\sum_{j=1}^{b}\text{表cを定義しているRDエリア数}\times 2\right)\end{array}\right\}$$

（単位：バイト）

a：実行する SQL 文数

b：c 番目の SQL 文中の表数

(10) SQL オブジェクト実行に関する統計情報（pcd）

●HiRDB/シングルサーバの場合

〈計算式〉

統計ログ量＝ 388×｛(a ＋ b ＋ c ＋ d ＋ e ＋ f) ＋ g ＋ h｝（単位：バイト）

●HiRDB/パラレルサーバの場合

〈計算式〉

統計ログ量＝ 388×｛(a ＋ b ＋ c ＋ d ＋ e ＋ f) ×i ＋ g ＋ h｝（単位：バイト）

a：OPEN 文の実行数

b：CLOSE 文の実行数

c：INSERT 文の実行数

d：DELETE 文の実行数

e：ASSIGN LIST 文の実行数

f：UPDATE 文の実行数

g：FETCH 文の実行数

h：DESCRIBE 文の実行数

i：次に示す計算式の値

$$\sum_{k=1}^{n}\left\{1+\sum_{m=1}^{Tm}\text{表mを定義しているRDエリア数}\times 2\right\}$$

n：実行する SQL 文数

Tm：m 番目の SQL 文中の表数

(11) SQL 文の履歴に関する統計情報（sqh）

〈計算式〉

統計ログ量＝（728 ＋ a）× b （単位：バイト）

a：SQL 文の平均 SQL 長

b：統計情報の取得中に実行する SQL の数

実際に実行する SQL の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL 数が不明確な場合，SQL 数は次の式から求められます。

SQL 数※＝ UAP が実行する SQL 文の平均的な数×

（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

注※

ストアドプロシジャ又はストアドファンクションを実行する場合，手続き又は関数中に記述するSQL 文数も数えてください。

(12) SQL オブジェクト転送に関する統計情報（obj）

〈計算式〉

統計ログ量＝ 300×統計情報の取得中に実行する SQL の数 （単位：バイト）

実際に実行する SQL の延べ数が分かる場合は，前記の計算式で求めてください。統計情報の収集時間内に実行する SQL 数が不明確な場合，SQL 数は次の式から求められます。

SQL 数※＝ UAP が実行する SQL 文の平均的な数×

（単位時間当たりの UAP 実行数×統計情報収集時間÷単位時間）

注※

ストアドプロシジャ又はストアドファンクションを実行する場合，手続き又は関数中に記述するSQL 文数も数えてください。

(13) データベースの入出力に関する統計情報（dio）

pdstbegin オペランド又は pdstbegin コマンドに指定したオプションによって異なります。

pd_statistics オペランドに A を指定した場合は，次の「-D everyfile を指定しない場合」の見積もり式を用いて計算してください。

●-D everyfile を指定しない場合

〈計算式〉

統計ログ量＝ 7464×（↓a÷b↓＋統計情報出力時間中の pdstjsync コマンド実行回数）（単位：バイト）

a：統計ログ出力時間（単位：秒）

統計情報出力終了時刻−統計情報出力開始時刻

b：統計情報出力間隔（単位：秒）

pdstbegin オペランド又は pdstbegin コマンドの-I オプションで指定した時間間隔です。

pd_statistics オペランドに A を指定した場合は 60 となります。

●-D everyfile を指定する場合

〈計算式〉

統計ログ量＝

$$\sum_{k=1}^{c} \{ (d \times 104) + (\uparrow d \div 16 \uparrow \times 204) \}$$

（単位：バイト）

a：統計ログ出力時間（単位：秒）

統計情報出力終了時刻−統計情報出力開始時刻

b：統計情報出力間隔（単位：秒）

pdstbegin オペランド又は pdstbegin コマンドの-I オプションで指定した時間間隔です。

c：↓a÷b↓＋統計情報出力時間中の pdstjsync コマンド実行回数（統計ログ取得回数）

d：各統計情報出力契機間※1 に入出力が発生する RD エリア※2 を構成する HiRDB ファイル数

注※1

- 統計情報出力間隔経過時
- 統計情報出力時間中の pdstjsync コマンド実行時

注※2

pdstbegin オペランド又は pdstbegin コマンドに-s オプションを指定した場合は，統計情報出力対象サーバが管理する RD エリアです。

付録 C.2　SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式

計算式で使用する変数については(3)を参照してください。

(1)　1SQL 文の SQL オブジェクト長の計算式

1SQL 文の SQL オブジェクト長は次に示す計算式から求めます。

〈計算式〉

```
1SQL文のSQLオブジェクト長（単位：キロバイト）＝
    ↑ ｛
    1600＋46×RCN＋394×Si＋24×Pi＋2058×Ti＋76×Ti×Di＋80×Ci＋40×Ii＋586×Wi
    ＋24×Ki＋Li＋8×TCi＋656×Di＋116×Ti×QX＋28×QX＋200×Ai＋48×nFF
    ＋100×nFP＋148×nFC＋712×nPFF＋32×（nAT＋nPAT）＋20×nCAT
    ＋28×（nAF＋nCAF）＋20×（nAA＋nPAA＋nCAA）＋1057×nSPA＋120×nSPP
    ＋287×nSFF＋8×nSFP＋813×nJFC＋20×nJFP
    トリガを使用する場合に加算します。
    ＋1057×nTR＋120×（nTSN＋nTSO）＋20×（nTCN＋nTCO）
    参照制約を使用する場合に加算します。
    ＋760＋376×RCC＋1880×RCT
    集合演算を使用する場合に加算します。
    ＋64×Ui
    マトリクス分割をする場合に加算します。
    ＋16×Tmi＋16×Tmi×Di
    行値構成子を指定する場合に加算します。
    ＋384×Wri＋32×QXs
    検査制約を使用する場合に加算します。
    ＋72×CDi＋88
```

```
SQL実行時の中間結果情報を取得する場合に加算します。
＋24×PIX＋192×Ti＋68×QX＋96
LIMIT句を指定する場合に加算します。
＋160
改竄防止表を定義する場合に加算します。
＋200
SET SESSION AUTHORIZATION文を使用する場合に加算します。
＋32
リストから受け渡し値を取得を使用する場合に加算します。
＋32×ALP
XML型を使用する場合に加算します。
＋36×Ti＋4240×XQX＋8×XQX×nURI
文字集合を使用する場合に加算します。
＋80×TCCi＋88×sRi
順序数生成子を使用する場合に加算します。
＋168×nSQ
データ圧縮機能を使用する場合に加算します。
＋8×Ci＋32×nUOC＋28
一時表を使用する場合に加算します。
＋200×nTT
列名記述領域長の計算式です。動的SQLの場合に加算します。
＋32×Si＋16
型名記述領域長の計算式です。動的SQLの場合に加算します。
＋↑（42×SiT）＋｛52＋152×（SiTA＋SiSA＋SiNA）×（SiT＋SiS＋SiN）｝↑
｝÷1024↑
```

注意事項

- ストアドプロシジャ又はストアドファンクションを使用する場合は，手続き又は関数中に記述する SQL 文も計算対象にしてください。
- トリガを使用する場合はトリガ中に記述するトリガ SQL 文も計算対象にしてください。
- ストアドプロシジャ，ストアドファンクション，又はトリガを使用する場合は，ストアドプロシジャ，ストアドファンクション，又はトリガごとにルーチン制御用オブジェクト長を計算して SQL オブジェクト用バッファ長に加算してください。1 ルーチンのルーチン制御用オブジェクト長の計算式を(2)に示します。

(2)　1 ルーチンのルーチン制御用オブジェクト長の計算式

(a)　ユーザが定義する場合

ストアドプロシジャ，ストアドファンクション，又はトリガを定義した場合の，1 ルーチンのルーチン制御用オブジェクト長は次に示す計算式から求めます。

〈計算式〉

```
1ルーチンのルーチン制御用オブジェクト長（単位：キロバイト）＝
    ↑｛
    600＋28×sRi＋32×（sRUi＋sDi）＋56×sSXi＋sCUi＋sSi＋sPi＋sLA＋sKi＋sL＋80×sWi
    ＋24×sCM＋32×sCCR＋2×sDCR＋60×sCHD＋72×sDHD＋64×sHCN＋8×sCHD×sHCN
    ＋48×nRFF＋100×nRFP＋148×nRFC＋200×nPRFF＋8×nPRFP＋196×nPA＋64×nPP
    ＋36×nPPI＋20×nPPO＋200×nPPA＋8×nPPP＋20×nAR＋48×nARA＋16×nRPAT
    ＋20×nCAT＋28×（nRPAF＋nRCAF）＋20×（nRPAA＋nRCAA）＋287×nRSFF
```

```
＋8×nRSFP＋813×nPJA＋20×nPJP＋813×nRJFC＋20×nRJFP
トリガを使用する場合に加算します。
＋28×（nTSN×2＋nTSO）
｝÷1024↑
```

(b)　HiRDB が自動的に作成する場合

表定義時，参照動作に CASCADE を指定した場合，HiRDB が制約制御のためにトリガを作成したときの 1 ルーチンのルーチン制御用オブジェクト長は次に示す計算式から求めます。

〈計算式〉

```
1ルーチンのルーチン制御用オブジェクト長（単位：キロバイト）＝
    ↑｛
    608×RCC＋（5120＋100×RDi＋256×RIi）×RCP×RCT
    ｝÷1024↑
```

(3)　計算式で使用する変数

変数名	説明
RCN	SQL オブジェクトで使用する表，及びインデクスの合計数
Si	SQL 文中にある検索項目数（SQL 文で指定する列がインデクス列の場合はその列数）
Pi	SQL 文中にある埋込み変数又はパラメタ数
Ti	SQL 文中にある表名数
Ci	SQL 文中にある列名数 表定義時に次に示す定義をした列数も加算してください（SQL 文中にない列であっても加算します）。 • 列定義に DEFAULT を指定している列 • 非ナル値制約指定に WITH DEFAULT を指定している列
TCi	SQL 文中にある表の構成列数
Wi	SQL 文中にある論理演算子数※
Ki	SQL 文中にある定数の数※
Li	SQL 文中にある定数の合計長※（単位：バイト）
Ii	SQL 文実行時に使用するインデクス数（SQL 文で指定する表のうち，検索条件に指定するインデクス数）
Di	SQL 文中の表に定義された格納条件の総数（マトリクス分割表は 2 倍する）
QX	問合せ指定数
Ai	SQL 文中のスカラ演算の総数
SiT	SQL 文中にある選択式の抽象データ型数
SiS	SQL 文中にある選択式の抽象データ型のスーパタイプ数
SiN	SQL 文中にある選択式のサブタイプである抽象データ型のスーパタイプの総数

変数名	説明
SiTA	SQL 文中にある選択式の抽象データ型の属性数
SiSA	SQL 文中にある選択式の抽象データ型のスーパタイプの属性数
SiNA	SQL 文中にある選択式のサブタイプである抽象データ型のコンポネント指定総数
nSPA	SQL 文中にある手続き文の呼び出し数
nSPP	SQL 文中にある手続き文の引数の総数
nFF	SQL 文中にある関数の呼び出し数※
nFP	SQL 文中にある関数の引数の総数※
nFC	SQL 文中にある関数の総関数定義候補数（関数呼出し数 nFF に，引数が抽象データ型の各関数呼出しに対して，サブタイプを引数とする関数定義数を加算する）
nPFF	SQL オブジェクトが使用するプラグイン関数呼出し数（SQL 文中にあるプラグイン関数呼出し数＋ SELECT の場合は 1，INSERT，UPDATE，DELETE の場合は 6）
nSFF	SQL 文中にあるシステム定義スカラ関数の呼び出し数※
nSFP	SQL 文中にあるシステム定義スカラ関数の引数の総数※
nJFC	SQL 文中にある外部 Java 関数の呼び出し数
nJFP	SQL 文中にある外部 Java 関数の引数の総数
nAT	SQL 文中にあるコンポネント指定で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を除く）
nAA	SQL 文中にあるコンポネント指定で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を含む）
nAF	SQL 文中にあるコンポネント指定で使用する属性総数
nPAT	SQL オブジェクトが使用するプラグイン関数の引数で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を除く）
nPAA	SQL オブジェクトが使用するプラグイン関数の引数で使用する抽象データ型数（スーパタイプ，サブタイプを含む）
nCAT	SQL 文中にあるコンストラクタ関数の呼び出し数
nCAA	SQL 文中にあるコンストラクタ関数の抽象データ型数（スーパタイプを含む）
nCAF	SQL 文中にあるコンストラクタ関数の抽象データ型の属性総数
nTR	SQL 文の実行によって起動されるトリガ数
nTSN	SQL 文の実行によって起動される各トリガのトリガ SQL 文中の新値相関名によって修飾された列の総数
nTSO	SQL 文の実行によって起動される各トリガのトリガ SQL 文中の旧値相関名によって修飾された列の総数
nTCN	SQL 文の実行によって起動される各トリガのトリガ動作条件中の新値相関名によって修飾された列の総数
nTCO	SQL 文の実行によって起動される各トリガのトリガ動作条件中の旧値相関名によって修飾された列の総数
RCC	SQL 文中で，更新対象の表を参照する表の外部キーの構成列数と，主キーの構成列数の総数

変数名	説明
RCT	SQL 文中で，更新対象の表を参照する表，及び更新対象の表が参照する表の総数
RCP	参照表定義時に，参照動作に指定した CASCADE の総数
RIi	参照表定義時，参照指定をする被参照表に定義されたインデクスの総数
RDi	参照表定義時，参照指定をする被参照表に定義された分割格納条件の総数（マトリクス分割表の場合は，2 倍する）
Ui	集合演算の数＋ 1
Tmi	マトリクス分割した表数
Wri	行値構成子要素を 2 以上指定した行値構成子を用いた条件数
QXs	複数の選択式を指定した表副問合せ数
CDi	検査制約の条件数
PIX	複数インデクス利用を行うインデクス数
ALP	リストから取得する受渡し値の受信関数数
XQX	XQuery 数
nURI	XML 名前空間 URI の指定数
TCCi	文字集合指定のある列数
nSQ	順序数生成子数
nUOC	データ圧縮に使用する UOC 数
nTT	一時表数
sRi	手続き及び関数中の SQL パラメタ数（INOUT 指定の SQL パラメタ数は 2 倍する）
sRUi	手続き及び関数中の SQL パラメタの総数（又は，トリガ定義中のトリガ SQL 文中の新旧値相関名によって修飾された列の総数）
sDi	手続き，関数，及びトリガ SQL 文中の SQL 変数（declare）の総数
sSXi	手続き，関数，及びトリガ SQL 文中の SQLCODE，SQLCOUNT 変数の総数
sCUi	手続き，関数，及びトリガ SQL 文中の CURRENT_TIME，CURRENT_DATE 定数の総数
sSi	手続き及びトリガ SQL 文中の操作系 SQL の数（カーソル宣言を除く SQL 文：OPEN，FETCH，CLOSE，UPDATE，DELETE，INSERT 文など）
sPi	手続き，関数，及びトリガ SQL 文中のルーチン制御 SQL 数（BEGIN，SET，IF，ELSEIF，WHILE など）
sLA	手続き，関数，及びトリガ SQL 文中のラベル数
sKi	手続き，関数，及びトリガ SQL 文中の定数の数（手続き及びトリガ SQL 文中に記述された操作系 SQL 文の定数を除く）
sL	手続き，関数，及びトリガ SQL 文中の定数の長さの合計（手続き及びトリガ SQL 文中に記述された操作系 SQL 文の定数を除く）
sWi	手続き，関数，及びトリガ SQL 文中の条件述語数

変数名	説明
sCM	手続き，関数，及びトリガ SQL 文中の複合文の数
sCCR	手続き及びトリガ SQL 文のカーソル宣言を記述した複合文の数
sDCR	手続き及びトリガ SQL 文のカーソル宣言の数
sCHD	手続き，関数，及びトリガ SQL 文のハンドラ宣言を記述した複合文の数
sDHD	手続き，関数，及びトリガ SQL 文のハンドラ宣言の数
sHCN	手続き，関数，及びトリガ SQL 文のハンドラ宣言中に記述された条件値の数
nRFF	ルーチン中にある関数呼出し数
nRFP	ルーチン中にある関数の引数の総数
nRFC	ルーチン中にある関数の関数定義候補の総数（関数呼出し数 nFF に，引数が抽象データ型の各関数呼出しに対して，サブタイプを引数とする関数定義数を加算する）
nPRFF	ルーチンの SQL オブジェクトが使用するプラグイン関数呼出し数
nPRFP	ルーチンの SQL オブジェクトが使用するプラグイン関数のプラグインパラメタ総数
nPA	ルーチン中にある手続き呼び出し数
nPP	ルーチン中にある手続きのパラメタ総数
nPPI	ルーチン中にある手続きの入力パラメタ総数（入出力パラメタを含む）
nPPO	ルーチン中にある手続きの出力パラメタ総数（入出力パラメタを含む）
nPPA	ルーチンの SQL オブジェクトが使用するプラグイン手続き呼び出し数
nPPP	ルーチンの SQL オブジェクトが使用するプラグイン手続きのプラグインパラメタ総数
nRSFF	ルーチン中にあるシステム定義スカラ関数の呼び出し数
nRSFP	ルーチン中にあるシステム定義スカラ関数の引数の総数
nPJA	ルーチン中にある外部 Java 手続きの呼び出し数
nPJP	ルーチン中にある外部 Java 手続きの引数の総数
nRJFC	ルーチン中にある外部 Java 関数の呼び出し数
nRJFP	ルーチン中にある外部 Java 関数の引数の総数
nAR	ルーチン中にあるコンポネント指定で使用する抽象データ型数（スーパタイプ，抽象データ型属性によって現れる抽象データ型を除く）
nARA	ルーチン中にあるコンポネント指定で使用する属性総数
nRPAT	ルーチンの SQL オブジェクトが使用するプラグインルーチンのパラメタで使用する抽象データ型数（ただし，スーパタイプ，抽象データ型属性の抽象データ型を除く）
nRPAA	ルーチンの SQL オブジェクトが使用するプラグインルーチンのパラメタで使用する抽象データ型数（スーパタイプを含む）
nRPAF	ルーチンの SQL オブジェクトが使用するプラグインルーチンのパラメタで使用する抽象データ型の属性総数

変数名	説明
nRCAT	ルーチン中にあるコンストラクタ関数の呼び出し数
nRCAA	ルーチン中にあるコンストラクタ関数の抽象データ型数（スーパタイプを含む）
nRCAF	ルーチン中にあるコンストラクタ関数の抽象データ型の属性総数

注※

トリガを使用する場合は，SQL 文の実行によって起動される各トリガのトリガ動作条件についても数える必要があります。

(4)　更新可能なオンライン再編成を実行する場合

更新可能なオンライン再編成を実行する場合，SQL オブジェクト用バッファ長を見積もり直す必要があります。SQL オブジェクト用バッファ長は次に示す計算式から求めてください。

SQL オブジェクト用バッファ長＝現在の SQL オブジェクト用バッファ長＋更新可能なオンライン再編成で必要な SQL オブジェクト用バッファ長

参考

更新可能なオンライン再編成の追い付き反映処理では，更新ログから SQL を組み立ててオリジナル RD エリアの表に対して SQL を実行します。追い付き反映処理時に実行される SQL は，UAP が実行した SQL とは異なることがあるため，その分の SQL オブジェクト用バッファ長を見積もる必要があります。

(a)　1SQL 文の SQL オブジェクト長の計算式

更新可能なオンライン再編成時の 1SQL 文の SQL オブジェクト長は，「1SQL 文の SQL オブジェクト長の計算式」で説明した計算式から求めてください。ただし，計算式中の変数の値については，次に示す表を参照してください。

変数名	説明
RCN	2 1SQL が発行されるときは，1 表かつ 1 インデクスのため，合計 2 となります。
Si	0 UAP の検索（SELECT）は追い付き反映の対象外のため，0 となります。
Pi	SQL 種別によって異なります。 • INSERT の場合：UAP が更新する表の構成列数※2 • UPDATE 表名 SET の場合：UAP が更新する表の構成列数※2 ＋マッピングキー構成列数 上記の式に，次のように更新している繰返し列の要素数を加算します。 ・列名［添え字］ ・列名［*］

変数名	説明
	• UPDATE 表名 ADD の場合：UAP が更新する繰返し列数＋マッピングキー構成列数 • UPDATE 表名 DELETE の場合：マッピングキー構成列数 • DELETE の場合：マッピングキー構成列数 なお，Pi の計算結果が 30001 以上になる場合は，Pi を 30000 としてください。
Ti	1 追い付き反映処理で実行される SQL の場合は，表名は一つのため，1 となります。
Ci	SQL 種別によって異なります。 • INSERT の場合：表の構成列数※2 • UPDATE 表名 SET の場合：UAP が更新する表の構成列数※2 ＋マッピングキー構成列数 上記の式に，更新している繰返し列の要素数を加算します。 • UPDATE 表名 ADD の場合：UAP が更新する繰返し列数＋マッピングキー構成列数 • UPDATE 表名 DELETE の場合：マッピングキー構成列数 • DELETE の場合：マッピングキー構成列数 なお，Ci の計算結果が 30001 以上になる場合は，Ci を 30000 としてください。
TCi	UAP が更新する表の構成列数
Wi	0
Ki	0
Li	0
Ii	1 更新可能なオンライン再編成の場合は，ユニークインデクスを一つ決定して検索条件に指定するため，1 となります。
Di	表の格納条件数 マトリクス分割の場合は 2 倍してください。
SiT	0
SiS	0
SiN	0
SiTA	0
SiSA	0
SiNA	0
nSPA	0
nSPP	0
nFF	表に定義されている抽象データ型の列数※1 ＋更新対象の BLOB 又は BINARY 列で SUBSTR 演算をしている列数
nFP	Σ 表に定義されている抽象データ型列※1 ＋ Σ {表に定義されている BLOB 又は BINARY 列※1 で SUBSTR 演算更新の数×3}
nFC	0

変数名	説明
nPFF	表に定義されている抽象データ型列数※1 ＋ 6 表に抽象データ型が定義されていない場合は 0 とします。
nSFF	BLOB 又は BINARY 列を SUBSTR 演算で更新している列数
nSFP	BLOB 又は BINARY 列を SUBSTR 演算で更新している列数×3
nJFC	0
nJFP	0
nAT	0
nAA	0
nAF	0
nPAT	0
nPAA	0
nCAT	表に定義されている抽象データ型列数※1
nCAA	表に定義されている抽象データ型の種類の数※1
nCAF	Σ（表に定義されている抽象データ型の属性数※1×n） n：SGMLTEXT 型及び XML 型の場合は 1，FREEWORD 型の場合は 2 になります。
nTR	0
nTSN	0
nTSO	0
nTCN	0
nTCO	0
RCC	SQL 文中で，更新対象の表を参照する表の外部キーの構成列数と，主キーの構成列数の総数
RCT	SQL 文中で，更新対象の表を参照する表，及び更新対象の表が参照する表の総数

注※1

pdorbegin コマンドに-e オプションを指定して，更新された列だけを反映する場合は，次に示すように読み替えてください。

読み替え前	読み替え後
表に定義されている抽象データ型の列数	表に定義されている抽象データ型の列のうち，更新対象となる抽象データ型列数
表に定義されている BLOB 又は BINARY 列	表に定義されている BLOB 又は BINARY 列のうち，更新対象となる BLOB 又は BINARY 列 ただし，UAP で明示的に更新していない BLOB 又は BINARY 列のうち，表定義時に次の定義をした列数も加算してください。

読み替え前	読み替え後
	• 列定義に DEFAULT 句を指定している列 • 非ナル値制約指定に WITH DEFAULT 句を指定している列
表に定義されている抽象データ型の種類の数	表に定義されている抽象データ型の種類のうち，更新対象となる抽象データ型の種類の数
表に定義されている抽象データ型の属性数	表に定義されている抽象データ型の属性のうち，更新対象となる抽象データ型の属性数

注※2

追い付き反映処理で発行した SQL 種別が INSERT 及び UPDATE の形式 2 となる場合，又は更新列だけ追い付き反映処理をする場合は，次に示すように読み替えてください。

読み替え前	SQL 種別	読み替え後
UAP が更新する表の構成列数	INSERT （形式 1）	−
	UPDATE （形式 1）	UAP で明示的に更新した列数 ただし，UAP で明示的に更新していない列のうち，表定義時に次の定義をした列数も加算してください。 • 列定義に DEFAULT 句を指定している列 • 非ナル値制約指定に WITH DEFAULT 句を指定している列
	INSERT （形式 2）	1
	UPDATE （形式 2）	

（凡例）

−：読み替え不要です。

(b)　必要最低限の SQL オブジェクト用バッファ長の求め方

追い付き反映処理時に SQL オブジェクト用バッファ不足が発生しない必要最低限の，SQL オブジェクト用バッファ長の見積もり式を次に示します。

〈計算式〉

```
MAX {ORG_SQL_OBJ_MAX_SIZE , ORG_SQL_OBJ_AVE_SIZE×ORG_PROC_NUM×
          n
MIN (100 , Σ ORG_SQL_KIND)  }
          i=1
```

変数の説明

ORG_SQL_OBJ_MAX_SIZE：追い付き反映処理中に実行される SQL の SQL オブジェクト長を個々に求め，その中で最大となる SQL オブジェクト長

ORG_SQL_OBJ_AVE_SIZE：追い付き反映処理中に実行される SQL の SQL オブジェクト長の平均値

ORG_PROC_NUM：追い付き反映プロセス数（pdorend コマンドの-m オプションの指定値）

-m オプションの省略時は 2 となります。

ORG_SQL_KIND：該当表に対して実行される SQL 種別数（INSERT，UPDATE，及び DELETE のうち，該当表に対して実行される SQL 種別数）

n：更新可能なオンライン再編成の対象表数

(c)　SQL オブジェクト用バッファ長の再見積もりが必要なとき

次に示す場合は，SQL オブジェクト用バッファ長を見積もり直してください。

- 「1SQL 文の SQL オブジェクト長の計算式」で説明している変数の値が大きくなる場合

  例えば，更新可能なオンライン再編成の対象表に，現在対象となっている表よりも列数が多い表を追加した場合などが該当します。
- 「必要最低限の SQL オブジェクト用バッファ長の求め方」で説明している変数の値が大きくなる場合

  新しい SQL 種別を実行するようになった場合が該当します。例えば，今までは INSERT だけを実行していたが，UPDATE も実行するようになった場合などが該当します。

付録 C.3　ビュー解析情報用バッファ長（pd_view_def_cache_size）の見積もり式

(1)　一つのビュー表当たりのビュー解析情報用バッファ長の計算式

一つのビュー表当たりのビュー解析情報用バッファ長は，次に示す計算式から求めます。計算式で使用する変数については，(2)及び(3)を参照してください。

〈32 ビットモードの HiRDB の計算式〉

```
1ビュー表当たりのビュー解析情報用バッファ長（単位：キロバイト）＝
    ↑1ビュー表当たりのビュー解析情報長÷1024↑

1ビュー表当たりのビュー解析情報長（単位：バイト）＝
    ↑（1024＋LCNST＋15）÷16↑×16
    ＋↑（LPTREE＋15）÷16↑×16
    ＋↑（28×NINCC＋15）÷16↑×16
    ＋↑（28×NINCP＋15）÷16↑×16
    ＋↑（24＋512×NTBL＋40×（NTBL＋NDTBL）
    ＋128×NCLM＋15）÷16↑×16
    ＋↑（16×MAX（↑NADTL÷50↑，1）
    ＋200×NADTL）÷16↑×16
    ＋↑（16×MAX（↑NATTL÷50↑，1）
    ＋144×NATTL）÷16↑×16
    ＋16＋16×（3＋NINCC＋NINCP）
```

〈64 ビットモードの HiRDB の計算式〉

```
1ビュー表当たりのビュー解析情報用バッファ長（単位：キロバイト）＝
    ↑1ビュー表当たりのビュー解析情報長÷1024↑

1ビュー表当たりのビュー解析情報長（単位：バイト）＝
    ↑（1600＋LCNST＋15）÷16↑×16
    ＋↑（LPTREE＋15）÷16↑×16
    ＋↑（40×NINCC＋15）÷16↑×16
    ＋↑（40×NINCP＋15）÷16↑×16
    ＋↑（32＋768×NTBL＋48×（NTBL＋NDTBL）
    ＋184×NCLM＋15）÷16↑×16
    ＋↑（20×MAX（↑NADTL÷50↑，1）
    ＋224×NADTL）÷16↑×16
    ＋↑（20×MAX（↑NATTL÷50↑，1）
    ＋160×NATTL）÷16↑×16
    ＋16＋24×（3＋NINCC＋NINCP）
```

注意事項

ビュー定義に，内部導出表を生成するビュー表を含む場合，そのビュー表のビュー解析情報用バッファ長の計算式も加算してください。

内部導出表を生成しないビュー表の場合は，そのビュー表を介さずに直接指定した SQL 文で計算した値を加算してください。

(2)　LPTREE の計算式

LPTREE は，次に示す計算式から求めます。計算式で使用する変数については，(3)を参照してください。

〈32 ビットモードの HiRDB の計算式〉

```
LPTREE（単位：バイト）＝
    276×NQRY
    ＋12×（2×NQRY＋NSBQ＋NSTOP）
    ＋12×（2×NQRY＋NSLST＋NSLAS）
    ＋12×（4×NTBL＋4×NJTBL＋5×NDTBL＋NDCLM）
    ＋12×（NWHRC＋NGPHV）
    ＋12×（NVLCM＋NEXCM＋NNLCM
    ＋3×NRANG＋3×NLKCM＋3×NSMCM＋NETCM）
    ＋12×NRVCL
    ＋12×NTVCL
    ＋12×NLGEX
    ＋12×（NCNST＋NSREG＋NPRCS）
    ＋12×4×NCSCV
    ＋12×（NARTH＋NCNCT）
    ＋12×NSFNC
    ＋12×NLBLD
    ＋12×（3×NCLM）
    ＋12×（NCASE＋NWHEN）
    ＋12×2×（NSCLF＋2×NCSSP）
    ＋12×（2×NPOS＋3×NDTV）
    ＋12×（NEXTR＋2×NSBST＋3×NVALU＋NBTEX＋NCAST）
    ＋12×（4×NFCSP＋NFPRM）
    ＋12×（3×NATTL＋NATNM）
    ＋28×NOPTL＋20×NOPTJ＋32×NOPTT＋20×NOPTIX
    ＋8×NSLAS
```

```
    ＋120×NJTBL
    ＋480×NCSET
    ＋84×NSPDT
    ＋72×（NSCLF＋NCASE＋NCSSP×2）
    ＋（144＋20＋32）×NFCSP＋（300＋136）×NRTNL
    ＋260×（NEXRTN＋NCLASS＋NJAR）＋32×NPVOW
    ＋140×NPLGL＋172×NPPRL
    ＋（260＋257＋257）×NPLGL
    ＋32×NCUD
    ＋512×NDTBL
    ＋40×NDTBL
    ＋128×NDCLM
    ＋182
    ＋376
    ＋1384
    ＋356×NVCLM
    ＋24×NVCSC
```

〈64 ビットモードの HiRDB の計算式〉

```
LPTREE（単位：バイト）＝
    496×NQRY
    ＋24×（2×NQRY＋NSBQ＋NSTOP）
    ＋24×（2×NQRY＋NSLST＋NSLAS）
    ＋24×（4×NTBL＋4×NJTBL＋5×NDTBL＋NDCLM）
    ＋24×（NWHRC＋NGPHV）
    ＋24×（NVLCM＋NEXCM＋NNLCM
    ＋3×NRANG＋3×NLKCM＋3×NSMCM＋NETCM）
    ＋24×NRVCL
    ＋24×NTVCL
    ＋24×NLGEX
    ＋24×（NCNST＋NSREG＋NPRCS）
    ＋24×4×NCSCV
    ＋24×（NARTH＋NCNCT）
    ＋24×NSFNC
    ＋24×NLBLD
    ＋24×（3×NCLM）
    ＋24×（NCASE＋NWHEN）
    ＋24×2×（NSCLF＋2×NCSSP）
    ＋24×（2×NPOS＋3×NDTV）
    ＋24×（NEXTR＋2×NSBST＋3×NVALU＋NBTEX＋NCAST）
    ＋24×（4×NFCSP＋NFPRM）
    ＋24×（3×NATTL＋NATNM）
    ＋56×NOPTL＋40×NOPTJ＋64×NOPTT＋40×NOPTIX
    ＋16×NSLAS
    ＋208×NJTBL
    ＋480×NCSET
    ＋96×NSPDT
    ＋96×（NSCLF＋NCASE＋NCSSP×2）
    ＋（176＋24＋40）×NFCSP＋（328＋144）×NRTNL
    ＋260×（NEXRTN＋NCLASS＋NJAR）＋32×NPVOW
    ＋168×NPLGL＋184×NPPRL
    ＋（260＋264＋257）×NPLGL
    ＋40×NCUD
    ＋768×NDTBL
    ＋48×NDTBL
    ＋184×NDCLM
```

＋226
＋568
＋1496
＋480×NVCLM
＋32×NVCSC

(3)　計算式で使用する変数

変数名	説明
LCNST	次に示す式の演算結果の定数の合計長※1，※2，※3 • ビュー定義中にある定数の合計長＋ビュー定義中にある文字集合変換 • 対象となる定数の変換後の合計長＋ビュー定義中にある数値文字変換 • 対象となる定数の変換後の合計長＋ビュー定義中にあるスカラ演算
NINCC	• LCNST ＞ 11516 バイトの場合 32 ビットモード：↑（LCNST-11516）÷4020↑ 64 ビットモード：↑（LCNST-11516）÷4024↑ • LCNST≦11516 バイトの場合 0
NINCP	• LTREE ＞ 11516 バイトの場合 32 ビットモード：↑（LTREE−11516）÷4020↑ 64 ビットモード：↑（LTREE−11516）÷4024↑ • LCNST≦11516 バイトの場合 0
NQRY	ビュー定義中にある問合せ指定の数
NSBQ	ビュー定義中にある副問合せの数
NSTOP	ビュー定義中にある集合演算の数
NSLST	ビュー定義中にある全問合せ指定の選択項目数の合計※4
NWHRC	ビュー定義中にある WHERE 句の数
NGPHV	ビュー定義中にある GROUP BY 句の数＋ビュー定義中にある HAVING 句の数
NVLCM	ビュー定義中にある比較述語の数＋ビュー定義中にある限定述語の数
NEXCM	ビュー定義中にある IN 述語の数
NNLCM	ビュー定義中にある NULL 述語の数
NRANG	ビュー定義中にある BETWEEN 述語の数
NLKCM	ビュー定義中にある LIKE 述語と XLIKE 述語の数
NSMCM	ビュー定義中にある SIMILAR 述語の数
NETCM	ビュー定義中にある EXISTS 述語の数
NRVCL	ビュー定義中にある行値構成子の要素の数
NTVCL	ビュー定義中にある IN 述語右側の行値構成子の要素の数

変数名	説明
NLGEX	ビュー定義中にある論理演算子の数
NCNST	ビュー定義中にある定数の延べ数＋ビュー定義中にあるパターン文字の延べ数
NSREG	ビュー定義中にある USER 値関数，CURRENT DATE 値関数，CURRENT TIME 値関数，又は CURRENT TIMESTAMP 値関数の総数
NPRCS	ビュー定義中にある CURRENT TIMESTAMP の小数秒精度指定の数
NCSCV	ビュー定義中にある文字集合変換対象となる定数の数※2
NARTH	ビュー定義中にある四則演算，日付演算，及び時刻演算の総数
NCNCT	ビュー定義中にある連結演算の数
NLBLD	ビュー定義中にあるラベル付き間隔の数
NSFNC	ビュー定義中にある集合関数の数
NSCLF	ビュー定義中にあるシステム組込みスカラ関数の数
NFCSP	ビュー定義中にある関数呼出し，又はスカラ関数の数
NCLM	ビュー定義中にある列の延べ数
NCASE	ビュー定義中にある CASE 式の数＋ビュー定義中にある CASE 略式 NULLIF の数
NWHEN	ビュー定義中にある CASE 式の WHEN の数＋ 1
NCSSP	ビュー定義中にある CAST 指定の数
NPOS	ビュー定義中にあるシステム組込みスカラ関数 POSITION の数
NDTV	ビュー定義中にあるシステム組込みスカラ関数（DATE，TIME，TIMESTAMP，又は VARCHAR_FORMAT）の数
NEXTR	ビュー定義中にあるシステム組込みスカラ関数（YEAR，MONTH，DAY，HOUR，MINUTE，又は SECOND）の数
NSBST	ビュー定義中にあるシステム組込みスカラ関数 SUBSTR の数
NVALU	ビュー定義中にあるシステム組込みスカラ関数 VALUE の数＋ビュー定義中にある CASE 略式 COALESCE の数
NBTEX	ビュー定義中にあるシステム組込みスカラ関数 BIT_AND_TEST の数
NCAST	ビュー定義中にあるシステム組込みスカラ関数（INTEGER，又は DECIMAL）の数
NOPTL	ビュー定義中にある SQL 最適化指定の数
NOPTJ	ビュー定義中にある結合方式の SQL 最適化指定の数
NOPTT	ビュー定義中にある表の SQL 最適化指定の数
NOPTIX	ビュー定義中にあるインデクスの SQL 最適化指定の数
NSLAS	ビュー定義中にある選択式の AS 句の数
NJTBL	ビュー定義中にある結合表の数
NCSET	HiRDB で指定できる文字集合の種別の数＋ 1※5

変数名	説明
NSPDT	ビュー定義中にある「AS データ型」指定の数
NFPRM	ビュー定義中にある関数呼出しの引数とシステム定義スカラ関数の引数の総数
NRTNL	ビュー定義中にある関数呼出しの呼び出し候補となる関数と，システム定義スカラ関数の呼び出し候補となる関数の総数
NEXRTN	ビュー定義中にある呼び出し候補となる関数のうち外部ルーチンの総数
NCLASS	ビュー定義中にある JAVA 関数のクラスの総数
NJAR	ビュー定義中にある JAVA 関数の JAVA アーカイブの総数
NPVOW	ビュー定義中にあるパブリックビューの所有者名の総数
NDTBL	ビュー定義中にある FROM 句の導出表の数＋ビュー定義中にある内部導出表を生成するビュー表の数
NTBL	• ビュー定義中にある表の延べ数が 15 個以下で，かつビュー定義中にある相関名の数と NDTBL の合計が 15 個以下の場合 15 • ビュー定義中にある表の延べ数が 16 個以上か，又はビュー定義中にある相関名の数と NDTBL の合計が 16 個以上の場合 Max（ビュー定義中にある表の延べ数，ビュー定義中にある相関名の数＋ NDTBL の値）
NDCLM	ビュー定義中にある FROM 句の導出表の合計列数＋ビュー定義中にある内部導出表を生成するビュー表の合計列数
NVCLM	ビュー定義によって導出される列数
NVCSC	ビュー定義によって導出される列で，既定文字集合以外の文字集合を指定した列
NADTL	ビュー定義中にあるすべての抽象データ型の継承関係※6 を含めた数
NATTL	ビュー定義中にあるコンポネント指定の数
NATNM	ビュー定義中にあるすべてのコンポネント指定の属性名数
NPLGL	ビュー定義中にあるプラグイン提供関数の数
NPPRL	ビュー定義中にあるプラグイン提供関数のパラメタ数
NCUD	ビュー定義中にある抽象データ型の列数

注※1

定数の種別によって次のように長さを計算します。

定数の種別	長さ（単位：バイト）	境界調整によって必要となる長さの最大値（単位：バイト）
整数定数	4	7
10 進数定数	↑（10 進数定数の精度＋ 1）÷2↑	↑（10 進数定数の精度＋ 1）÷2↑＋ 7
浮動小数点数定数	8	15
文字列定数	2 ＋文字列定数の文字数	2 ＋文字列定数の文字数＋ 3

定数の種別	長さ（単位：バイト）	境界調整によって必要となる長さの最大値（単位：バイト）
16 進文字列定数	4 ＋（16 進文字列定数の文字列長）÷2	4 ＋（16 進文字列定数の文字列長）÷2 ＋ 3
各国文字列定数	2 ＋各国文字列定数の文字数×2	2 ＋各国文字列定数の文字数×2 ＋ 3
混在文字列定数	2 ＋混在文字列定数の半角文字の数＋混在文字列定数の全角文字の文字数×2	2 ＋混在文字列定数の半角文字の数＋混在文字列定数の全角文字の文字数×2 ＋ 3

注※2

文字集合変換対象となる定数とは，次の定数です。

- 比較対象が文字集合の項目である文字列定数
- CAST 指定で変換後のデータ型に文字集合指定をした文字列定数

注※3

数値文字変換対象となる定数とは，次の定数です。

- 比較対象が数値属性の項目である文字列定数
- 連結演算のオペランドに指定された数値定数
- 四則演算のオペランドに指定された文字列定数

注※4

選択項目に＊を指定した場合は，FROM 句に指定したすべての表の列数の合計を加算します。また，選択式に表名.＊を指定した場合は，その表の列数を加算します。

注※5

HiRDB で指定できる文字集合は，EBCDIK だけです。

注※6

プラグイン提供の抽象データ型の継承関係は，各プラグインのマニュアルを参照してください。

付録 C.4　表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式

(1)　計算式で使用する変数

ａ：ディクショナリ表（SQL_TABLES 表）の DEFINITION_CACHE_SIZE の値（単位：バイト）

DEFINITION_CACHE_SIZE の値が分からない場合は，「表定義キャッシュサイズの求め方」を参照してください。

ｂ：表の列数

追い付き状態管理表の場合は 8 を代入してください。

c：表のインデクス数

追い付き状態管理表の場合は 1 を代入してください。

d：表分割条件数

追い付き状態管理表の場合は pdorcreate コマンドの-r オプション，及び-o オプションに指定した RD エリアが属するサーバ数を代入してください。

e：表格納 RD エリア数

追い付き状態管理表の場合は pdorcreate コマンドの-r オプションに指定した RD エリア数＋ 1 を代入してください。

f：インデクス格納 RD エリア数

追い付き状態管理表の場合は pdorcreate コマンドの-r オプションに指定した RD エリア数＋ 1 を代入してください。

g：BLOB 列数

h：BLOB 属性を含む抽象データ型数

i：抽象データ型中の BLOB 属性の総数

j：抽象データ型数

k：プラグインオプション数

n：インデクス除外キー値数

p：ディクショナリ表（SQL_TABLES 表）の STATISTICS_CACHE_SIZE の値（単位：バイト）

pdgetcst コマンドで表の最適化情報を取得した場合に加算します。単位がバイトなのでキロバイトに変換して代入してください。STATISTICS_CACHE_SIZE の値が分からない場合は次に示す計算式で求めます。

（2.6×q $_1$ ＋ 3.0×q $_2$ ＋ 0.04×c ＋ 0.02）×1024（単位：バイト）

q $_1$：列のデータ型が DECIMAL 型以外の列最適化情報数，及び精度 31 けたまでの DECIMAL 型の列最適化情報数の合計

q $_2$：列のデータが精度 32 けた以上の DECIMAL 型の列最適化情報数

r：CREATE TABLE の DEFAULT オペランドの指定列数

s：CREATE TABLE の DEFAULT オペランドに指定する既定値長の合計値（単位：バイト）

既定値を指定するすべての列数分加算します。既定値長を長く変更することがある場合は，変更後の長さを加算してください。

t：パブリックビュー表の表名と同じ表識別子を所有する認可識別子数

u：表に定義しているトリガ数

ｖ：新旧値相関名で修飾して使用している列数

ｗ：トリガ動作条件数

ｘ：トリガ動作条件の総解析ツリー長（単位：バイト）

トリガ動作条件の解析ツリー長の見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」の「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

ｙ：UPDATE トリガのトリガ契機列数

ｚ：表に定義している外部キー数

ａａ：表の主キーを参照している外部キー数

ａｂ：表に定義している検査制約数

ａｃ：検査制約用解析ツリーの合計長（単位：バイト）

検査制約用解析ツリー長の見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」の「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

ａｄ：インデクス構成部分構造パス数

ａｅ：該当するインデクスの部分構造パスの合計長

ａｆ：該当するインデクスの部分構造パス用解析ツリーの合計長（単位：バイト）

部分構造パス用解析ツリー長の見積もり式については，マニュアル「HiRDB システム導入・設計ガイド」の「通常のデータディクショナリ用 RD エリアの容量の見積もり」を参照してください。

ａｇ：表に定義している圧縮列の数

(2)　1 表当たりの表定義情報バッファサイズ（単位：キロバイト）

1 表当たりの表定義情報バッファサイズは，次に示す概算式で求めてください。なお，ビュー表の場合は，ビュー表の基となる実表についても計算してください。

表の種類	1 表当たりの表定義情報バッファサイズ の計算式（単位：キロバイト）
1 実表の表定義情報バッファサイズ	↓（4 ＋↑ａ÷1024↑＋ 0.01×ｂ＋↑ｐ÷1024↑＋ 7）÷8↓×8
1 ビュー表の表定義情報バッファサイズ	↓（4 ＋↑ａ÷1024↑＋ 0.01×ｂ＋ 7）÷8↓×8
追い付き状態管理表（HiRDB Staticizer Option）の表定義情報バッファサイズ	↓（4 ＋↑ａ÷1024↑＋ 0.01×8 ＋ 7）÷8↓×8

(3)　表定義キャッシュサイズの求め方

(a)　DEFINITION_CACHE_SIZE の値が分からない場合

次のどれかに該当する場合は，ディクショナリ表を検索しても DEFINITION_CACHE_SIZE の値が分かりません。

- まだ表を定義していない場合
- HiRDB Version 4.0 からバージョンアップした場合※1
- 32 ビットモードの HiRDB から 64 ビットモードの HiRDB に移行した場合※1，※2

注※1

この場合，DEFINITION_CACHE_SIZE に正しい値が入っていません。

注※2

32 ビットモードから 64 ビットモードに移行した表であるかどうかは，pdvrup コマンドの実行時間と表の作成時間を比較してください。表の作成時間の方が古い表が，32 ビットモードから 64 ビットモードに移行した表です。pdvrup コマンドの実行時間は，ディクショナリ表（SQL_TABLES 表）の MASTER.SQL_TABLES の CHANGE_TIME で分かります。表の作成時間は，ディクショナリ表（SQL_TABLES 表）の作成した表の CREATE_TIME で分かります。これらの列は，次に示す SQL で検索できます。

- **pdvrup コマンド実行時間**

```
select CHANGE_TIME from MASTER.SQL_TABLES where
    TABLE_SCHEMA='MASTER' and TABLE_NAME='SQL_TABLES'
```

- **表の作成時間**

```
select CREATE_TIME from MASTER.SQL_TABLES where
    TABLE_SCHEMA='認可識別子' and TABLE_NAME='表名'
```

(b)　表定義キャッシュサイズの計算式

表定義キャッシュサイズは次に示す計算式で求めてください。

- 実表の場合：（計算式 1 ＋計算式 2）×1024（単位：バイト）
- ビュー表の場合：（計算式 1 ＋計算式 3）×1024（単位：バイト）
- 追い付き状態管理表の場合：計算式 1×1024（単位：バイト）

計算式の種類	計算式
計算式 1	**●HiRDB が 32 ビットモードの場合** 0.6 ＋ 0.13×ｂ＋ 0.25×ｃ＋ 0.18×（ｄ＋ 1）＋ 0.01×（ｅ＋ｆ） **●HiRDB が 64 ビットモードの場合** 0.9 ＋ 0.19×ｂ＋ 0.28×ｃ＋ 0.18×（ｄ＋ 1）＋ 0.01×（ｅ＋ｆ）

計算式の種類	計算式
計算式 2	**●LOB 列を定義した表の場合に加算します。** （0.02 ＋ 0.01×ｅ）×ｇ **●BLOB 属性を含む抽象データ型の列を定義した表の場合に加算します。** 0.02×ｈ＋（0.02 ＋ 0.01×ｅ）×ｉ **●抽象データ型の列を定義した表の場合に加算します。** 0.3×ｊ＋0.3×ｋ **●インデクス除外キー値を定義した表の場合に加算します。** 0.3×ｎ **●SEGMENT REUSE を指定（SEGMENT REUSE NO を除く）した表の場合に加算します。** 0.01×ｅ **●DEFAULT オペランドに既定値を指定した表の場合に加算します。** 0.02×ｒ＋↑ｓ÷1024↑ **●トリガを定義した表の場合に加算します。** 0.3×ｕ＋0.2×ｖ＋0.1×ｗ＋↑ｘ÷1024↑＋0.2×ｙ **●外部キーを定義した表の場合に加算します。** 1.3×ｚ **●被参照表の場合に加算します。** 0.2 ＋ 0.1×ａａ **●検査制約を定義した表の場合に加算します。** 0.1×ａｂ＋↑ａｃ÷1024↑ **●部分構造インデクスを定義した表の場合に，部分構造インデクスごとに加算します。** 0.1×ａｄ＋↑ａｅ÷1024↑＋↑ａｆ÷1024↑ **●圧縮表の場合に，加算します。** 0.08×ａｇ
計算式 3	**●パブリックビュー表の場合に加算します。** 0.1 ＋（0.1×ｔ） **●ビュー表の構成列が抽象データ型となる場合に加算します。** 0.3×ｊ

付録 C.5　1 サーバ当たりの UNTIL DISCONNECT 指定の排他表と RD エリア数の総和（pd_lck_until_disconnect_cnt）の見積もり式

1 サーバ当たりの UNTIL DISCONNECT 指定排他の表と RD エリアの総和（pd_lck_until_disconnect_cnt）の見積もり式を次に示します。なお，計算式はサーバで異なります。サーバ共通定義の見積もり式は，バックエンドサーバと同じです。

〈計算式〉

サーバの種類	計算式
シングルサーバ	{a＋b＋c＋d＋o＋p＋q＋（t×3）＋（w×2）＋u＋v＋x}×2＋{2×（f＋g＋i＋1）＋（g＋y＋z）＋A＋B}×e
ディクショナリサーバ	（a＋b＋c＋d）×2＋{2×（f＋g＋i＋1）＋（g＋y＋z）＋A＋B}×e
バックエンドサーバ	{a＋b＋c＋d＋o＋p＋q＋r＋s＋（t×3）＋（w×2）＋u＋v＋x}×2＋{2×（f＋g＋i＋1）＋（g＋y＋z）＋A＋B}×e

a：同時実行する UNTIL DISCONNECT 指定の LOCK TABLE 文で指定する表の数

b：同時実行する UNTIL DISCONNECT 指定の LOCK TABLE 文で指定する表を格納する RD エリア数

c：1
インナレプリカ構成管理数です。インナレプリカ機能を使用する場合に加算します。

d：同時実行する UNTIL DISCONNECT 指定の LOCK TABLE 文で指定する表を格納する RD エリア数
レプリカグループ構成管理数です。インナレプリカ機能を使用する場合に加算します。

e：同時実行するユティリティの数

f：同時実行するユティリティの処理対象表を格納する LOB 用 RD エリアの数

g：同時実行するユティリティの処理対象インデクスを格納する RD エリアの数

i：同時実行するユティリティの処理対象表を格納する RD エリアの数

o：同時に使用するデータ用ローカルバッファが割り当てられた表の格納 RD エリアの総数
データ用ローカルバッファを使用する場合に加算します。

p：同時に使用するインデクス用ローカルバッファが割り当てられたインデクスの格納 RD エリアの総数
インデクス用ローカルバッファを使用する場合に加算します。

q：同時に使用するインデクス用ローカルバッファが割り当てられたインデクスの対象の表の総数
インデクス用ローカルバッファを使用する場合に加算します。

r：同時実行する IN EXCLUSIVE MODE 指定の LOCK 文で指定する共用表格納 RD エリアの総数
共用 RD エリアを使用する場合に加算します。

s：同時実行する IN EXCLUSIVE MODE 指定の LOCK 文で指定する共用表に定義されているすべての共用インデクス格納 RD エリアの総数
共用 RD エリアを使用する場合に加算します。

t：同時に実体化する SQL セッション固有一時表数
SQL セッション固有一時表を使用する場合に加算します。

u：同時に実体化する SQL セッション固有一時表を格納する一時表用 RD エリアの総数
SQL セッション固有一時表を使用する場合に加算します。

v：同時に実体化する SQL セッション固有一時表に定義されている一時インデクスを格納する一時表用 RD エリアの総数

SQL セッション固有一時表を使用する場合に加算します。

w：同時に実体化する SQL セッション固有一時インデクス数

SQL セッション固有一時表を使用する場合に加算します。

x：同時に実体化する SQL セッション固有一時表及び SQL セッション固有一時インデクスが格納されている RD エリア内の使用セグメント数の総数

SQL セッション固有一時表を使用する場合に加算します。

y：使用する順序数生成子数

データベース作成ユティリティを実行する場合に加算します。

z：使用する順序数生成子格納 RD エリア数

データベース作成ユティリティを実行する場合に加算します。

A：処理対象表の外部キー数

整合性チェックユティリティを実行する場合に加算します。

B：処理対象表の外部キーが参照する被参照表の格納 RD エリアの総数

整合性チェックユティリティを実行する場合に加算します。

注　ここでいうユティリティとは次に示すユティリティのことです。

- データベース作成ユティリティ
- データベース再編成ユティリティ
- 空きページ解放ユティリティ
- グローバルバッファ常駐化ユティリティ
- リバランスユティリティ（占有モード）
- 整合性チェックユティリティ

付録 C.6　ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式

ルーチン定義情報用バッファ長は次に示す計算式から求めます。

```
　使用頻度が高いルーチンの定義情報長の合計値
＋使用するプラグインのプラグイン関数の定義情報長の合計値
＋使用頻度が高いシステム定義スカラ関数の定義情報長の合計値
```

(1)　1 ルーチン当たりのルーチン定義情報長の求め方

1 ルーチン当たりのルーチン定義情報長は，次に示す概算式で求めてください。

〈計算式〉

↑（1.3 ＋ 0.2× a）↑× b　　（単位：キロバイト）

a：使用頻度が高いルーチンの総パラメタ数

b：使用頻度が高いルーチンの定義数

(2)　プラグイン関数の定義情報長の求め方

プラグイン関数の定義情報長は，次に示す概算式で求めてください。

〈計算式〉

0.6 ＋ c ＋ 0.2× d　　（単位：キロバイト）

c：一つのプラグインが DML で利用するプラグイン関数の合計数※

d：一つのプラグインが DML で利用するプラグイン関数のパラメタの合計数※

注

上記の計算式は１プラグイン当たりの計算式です。複数のプラグインをインストールしている場合は，インストールしたプラグインの数だけ計算して加算してください。

注※

DML で利用するプラグイン関数，及び DML で利用するプラグイン関数のパラメタの合計数は次に示す SQL で求めます。

```
SELECT COUNT(*),SUM(N_PARAM) FROM MASTER.SQL_PLUGIN_ROUTINES
  WHERE PLUGIN_NAME = 'プラグイン名称'
  AND (TIMING_DESCRIPTOR = 'ADT_FUNCTION'
    OR TIMING_DESCRIPTOR IS NULL
    OR TIMING_DESCRIPTOR = 'BEFORE_INSERT'
    OR TIMING_DESCRIPTOR = 'AFTER_INSERT'
    OR TIMING_DESCRIPTOR = 'BEFORE_UPDATE'
    OR TIMING_DESCRIPTOR = 'AFTER_UPDATE'
    OR TIMING_DESCRIPTOR = 'BEFORE_DELETE'
    OR TIMING_DESCRIPTOR = 'AFTER_DELETE'
    OR TIMING_DESCRIPTOR = 'BEFORE_PURGE_TABLE'
    OR TIMING_DESCRIPTOR = 'AFTER_PURGE_TABLE'
    OR TIMING_DESCRIPTOR = 'INDEX_SEARCH'
    OR TIMING_DESCRIPTOR = 'INDEX_COUNT'
    OR TIMING_DESCRIPTOR = 'INDEX_INSERT'
    OR TIMING_DESCRIPTOR = 'INDEX_BEFORE_UPDATE'
    OR TIMING_DESCRIPTOR = 'INDEX_AFTER_UPDATE'
    OR TIMING_DESCRIPTOR = 'INDEX_DELETE'
    OR TIMING_DESCRIPTOR = 'PURGE_INDEX'
    OR TIMING_DESCRIPTOR = 'INDEX_MAINTENANCE_DEFERRED'
    OR TIMING_DESCRIPTOR = 'BEFORE_INSERT_DC'
    OR TIMING_DESCRIPTOR = 'BEFORE_UPDATE_DC'
    OR TIMING_DESCRIPTOR = 'BEFORE_DATA_CHECK'
    OR TIMING_DESCRIPTOR = 'AFTER_DATA_CHECK')
```

(3)　一つ当たりのシステム定義スカラ関数の定義情報長の求め方

一つ当たりのシステム定義スカラ関数の定義情報長を次の表に示します。

表 C–1　一つ当たりのシステム定義スカラ関数の定義情報長

関数名	定義情報長 （単位：キロバイト）
ACOS，ADD_INTERVAL，ASCII，ASIN，ATAN，ATAN2，CENTURY，COS，COSH，CHR，DATE_TIME，DAYNAME，DAYOFWEEK，DAYOFYEAR，DEGREES，EXP，INTERVAL_DATETIMES，LAST_DAY，LN，LOG10，MIDNIGHTSECONDS，MONTHNAME，MONTHS_BETWEEN，NEXT_DAY，PI，RADIANS，SIN，SINH，SQRT，TAN，TANH，WEEK，WEEKOFMONTH，YEARS_BETWEEN	2
POWER，IS_DBLBYTES，IS_SNGLBYTES，ISDIGITS，ROUNDMONTH，TRANSL_LONG	4
CEIL，FLOOR，HALF，INSERTSTR，INSERTSTR_LONG，LEFTSTR，LTRIMSTR，NUMEDIT，QUARTER，REPLACE_LONG，REVERSESTR，RIGHTSTR，RTRIMSTR，SIGN，STRTONUM，TRUNCYEAR	6
LTRIM，REPLACE，RTRIM，TRANSL，TRUNC	12
POSSTR，ROUND	18
GREATEST，LEAST	32

付録 D　排他資源数の見積もり

SQL 文，ユティリティ，及びコマンド実行時や HiRDB 起動時に必要となる排他資源数の概算式を示します。これらの値から，最大の排他資源数を算出し余裕値を加算して，次に示すオペランドにプールサイズを設定してください。

- 各サーバのサーバ定義の pd_lck_pool_size オペランド
- フロントエンドサーバ定義の pd_fes_lck_pool_size オペランド

注意事項

1. ここで示す排他資源数は，トランザクション内で有効となります。一つのトランザクションで複数の SQL を実行する場合，その排他資源の総和が必要となりますが，既に排他が掛かっている場合は必要としません。排他制御については，マニュアル「HiRDB UAP 開発ガイド」を参照してください。
2. HiRDB/パラレルサーバの場合，各バックエンドサーバのリソース数は，対象バックエンドサーバで管理されるリソース数（RD エリア数，インデクス数，行数など）で算出する必要があります。
3. pd_inner_replica_control オペランドを指定した場合に加算される排他資源数は，トランザクション実行時 1 回だけ要求されます。

付録 D.1　定義系 SQL

(1)　ALTER INDEX

(a)　HiRDB/シングルサーバの場合

```
7＋（インデクス構成列数×2）＋オブジェクトが無効となる手続き，及びトリガ数
オブジェクトが無効となる手続きがある場合に加算します。
＋オブジェクトが無効となる手続き数×3
＋オブジェクトが無効となる手続きが使用する資源数×5
オブジェクトが無効となるトリガがある場合に加算します。
＋オブジェクトが無効となるトリガ数×5
＋UPDATEトリガのトリガ契機に指定した列数×3
＋インデクスが定義されている表を参照するトリガ動作手続き数×5
  新旧値相関名がある場合にさらに加算します。
  ＋4
インデクスに関する統計情報を出力している場合に加算します。
＋2
分割キーインデクスの場合に加算します。
＋分割条件数×2
インデクス型を使用しているインデクスの場合に加算します。
＋1
＋インデクス型を定義している型が使用する抽象データ型数×2
＋インデクス型が使用する関数数×2
部分構造インデクスの場合に加算します。
＋構成部分構造パス数×2
```

除外値指定がある場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクトが無効となる手続き，及びトリガ数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6＋（インデクス構成列数×2）
オブジェクトが無効となる手続きがある場合に加算します。
＋オブジェクトが無効となる手続き数×3
＋オブジェクトが無効となる手続きが使用する資源数×5
オブジェクトが無効となるトリガがある場合に加算します。
＋オブジェクトが無効となるトリガ数×5
＋UPDATEトリガのトリガ契機に指定した列数×3
＋インデクスが定義されている表を参照するトリガ動作手続き数×5
　新旧値相関名がある場合にさらに加算します。
　＋4
インデクスに関する統計情報を出力している場合に加算します。
＋2
分割キーインデクスの場合に加算します。
＋分割条件数×2
インデクス型を使用しているインデクスの場合に加算します。
＋1
＋インデクス型を定義している型が使用する抽象データ型数×2
＋インデクス型が使用する関数数×2
部分構造インデクスの場合に加算します。
＋構成部分構造パス数×2
除外値指定がある場合に加算します。
＋2

(2)　ALTER PROCEDURE

(a)　HiRDB/シングルサーバの場合

5
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
実行者が所有者以外の場合に加算します。
＋2
手続き名称指定の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数＋2
　ALL指定の場合にさらに加算します。
　＋対象者が所有する全ルーチン数
　＋対象者が定義した全PUBLICルーチン数＋1
　INDEX USING指定の場合にさらに加算します。
　＋指定した表をリソースとして使用するルーチン数×2＋3
AUTHORIZATION指定なしの場合に加算します。

＋システム内の無効な全ルーチン数
ALL指定の場合にさらに加算します。
＋システム内の全ルーチン数
INDEX USING指定の場合にさらに加算します。
＋指定した表をリソースとして使用するルーチン数×2＋3
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5
実行者が所有者以外の場合に加算します。
＋2
手続き名称指定の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数
ALL指定の場合にさらに加算します。
＋対象者が所有する全ルーチン数
＋対象者が定義した全PUBLICルーチン数
INDEX USING指定の場合にさらに加算します。
＋指定した表をリソースとして使用するルーチン数×2＋3
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
ALL指定の場合にさらに加算します。
＋システム内の全ルーチン数
INDEX USING指定の場合にさらに加算します。
＋指定した表をリソースとして使用するルーチン数×2＋3
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1

使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1

(3)　ALTER ROUTINE

(a)　HiRDB/シングルサーバの場合

5
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
トリガがある場合に加算します。
＋1
実行者が所有者以外の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数＋1
　ALL指定の場合にさらに加算します。
　＋対象者が所有する全ルーチン数
　＋対象者が定義した全PUBLICルーチン数
　対象者の所有するトリガがある場合にさらに加算します。
　＋対象者が所有するトリガ数×4＋3
　＋対象者が所有するトリガを定義している表数
　＋トリガの契機列数×2＋1
　＋トリガの動作条件中で使用している資源情報数×2＋1
　＋トリガで使用するパラメタ数×2＋1
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
　ALL指定の場合にさらに加算します。
　＋システム内の全ルーチン数
　システム内にトリガがある場合にさらに加算します。
　＋全トリガ数×4＋3
　＋トリガを定義している全表数
　＋トリガの契機列数×2＋1
　＋トリガの動作条件中で使用している資源情報数×2＋1
　＋トリガで使用するパラメタ数×2＋1
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1

使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
トリガがある場合に加算します。
＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5
実行者が所有者以外の場合に加算します。
＋2
AUTHORIZATION指定の場合に加算します。
＋対象者が所有する無効なルーチン数
＋対象者が定義した無効なPUBLICルーチン数
　ALL指定の場合にさらに加算します。
　＋対象者が所有する全ルーチン数
　＋対象者が定義した全PUBLICルーチン数
　対象者の所有するトリガがある場合にさらに加算します。
　＋対象者が所有するトリガ数×4＋3
　＋対象者が所有するトリガを定義している表数＋1
　＋トリガの契機列数×2＋1
　＋トリガの動作条件中で使用している資源情報数×2＋1
　＋トリガで使用するパラメタ数×2＋1
AUTHORIZATION指定なしの場合に加算します。
＋システム内の無効な全ルーチン数
　ALL指定の場合にさらに加算します。
　＋システム内の全ルーチン数
　システム内にトリガがある場合にさらに加算します。
　＋全トリガ数×4＋3
　＋トリガを定義している全表数＋1
　＋トリガの契機列数×2＋1
　＋トリガの動作条件中で使用している資源情報数×2＋1
　＋トリガで使用するパラメタ数×2＋1
ルーチンに使用するリソースがある場合に加算します。
＋使用するルーチン数×2
使用するリソースに表がある場合に加算します。
＋使用する表数＋1
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数＋1
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数＋1

使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型＋1
パラメタを指定している場合に加算します。
＋パラメタ数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1

(4)　ALTER TABLE

(a)　HiRDB/シングルサーバの場合で，主キーの追加，削除以外のとき

6＋表を基に定義してあるビュー表数
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×3＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2
＋定義トリガの契機列数＋1
＋定義トリガで使用するパラメタ数＋1
インナレプリカ機能を使用している場合に加算します。
＋表で使用しているRDエリア数＋1
　インデクスが定義してある場合にさらに加算します。
　＋表に定義してあるインデクス数＋1
　BLOB列が定義してある場合にさらに加算します。
　＋表に定義してあるBLOB列数＋1
　表が分割表の場合にさらに加算します。
　＋表の分割数＋1
　　分割キーインデクスが定義してある場合にさらに加算します。
　　＋表の分割数×分割キーインデクス数＋1
　　BLOB列が定義してある場合にさらに加算します。
　　＋表の分割数×BLOB列数＋1
ADD列名を指定する場合に加算します。
＋5
　表を基に定義したビュー表がある場合にさらに加算します。
　＋表を基に定義したビュー表数
　BLOB列を追加する場合にさらに加算します。
　＋表に追加するRDエリア数＋1
　＋表で使用しているRDエリア数＋1
　＋表のBLOB列数
　＋2
　＋（1＋表に定義してあるインデクス数
　＋表に定義してあるBLOB列数
　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
　　表をリソースとして使用するルーチンがある場合にさらに加算します。
　　＋表をリソースとするルーチン数＋1
　　表を基に定義したビュー表がある場合にさらに加算します。
　　＋表を基に定義したビュー表数×2＋2
　　インデクスが定義してある場合にさらに加算します。
　　＋表に定義してあるインデクス数＋1
　　表が分割表の場合にさらに加算します。
　　＋表の分割数×2＋2
　　　分割キーインデクスが定義してある場合にさらに加算します。
　　　＋分割キーインデクス数×表の分割数＋1

BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
インナレプリカ機能を使用している場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
ユーザ定義型列を追加する場合にさらに加算します。
＋4
＋ユーザ定義型で使用する属性数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
BLOB属性が定義されているユーザ定義型の場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
プラグインの抽象データ型を使用している場合にさらに加算します。
＋使用するプラグイン数＋1
＋使用するプラグインのルーチン数＋1
＋使用するプラグインルーチンのパラメタ数＋1
インナレプリカ機能を使用している場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
表がFIX表の場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
非FIX表にNOT NULL列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。

＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
DEFAULT句の指定がある列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
列を切り出す場合にさらに加算します。
＋1
ADD RDAREAの場合に加算します。
＋9
＋表の構成列数
＋2
＋表に定義してあるインデクス数
＋4
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数
FIXハッシュ表の場合に加算します。
＋表で使用している表用RDエリア数
＋表の分割数
表で既に使用しているRDエリアを追加する場合にさらに加算します。
＋既に使用しているRDエリア数
表で未使用のRDエリアを追加する場合にさらに加算します。
＋追加するRDエリア数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクスの使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×RDエリア追加後の分割数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋2
＋プラグイン列数＋1
＋表で使用しているプラグインの属性数＋1
＋表で使用しているプラグインのルーチン数＋1
＋表で使用しているプラグインルーチンのパラメタ数＋1
＋表で使用しているプラグインのBLOB属性数×RDエリア追加後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
＋表に定義してあるBLOB列数×RDエリア追加後の分割数＋1
＋表に定義してあるBLOB列の使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるBLOB列数
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
＋表で使用しているユーザ定義型の属性数＋1
ユーザ定義型にBLOB属性が定義してある場合にさらに加算します。

＋表で使用しているユーザ定義型のBLOB属性数＋1
＋表で使用しているユーザ定義型のBLOB属性の使用RDエリア数×2＋1
＋表で使用しているユーザ定義型のBLOB属性数×RDエリア追加後の分割数＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるユーザ定義型列数
インナレプリカ機能を使用している場合にさらに加算します。
＋追加するRDエリア数×世代数
＋2
＋（表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数＋2
＋表に定義してあるユーザ定義型のBLOB属性数）×（表で使用しているRDエリア数＋1）×（世代数＋1）
CHANGE 列名の場合に加算します。
＋11
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表にトリガが定義してある場合にさらに加算します。
＋列をリソースとして使用するトリガ数×2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
＋列をパラメタとして使用するルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
列のデータ型がVARCHAR，NVARCHAR，MVARCHARの場合にさらに加算します。
＋表の構成列数
＋表で使用しているインデクス用RDエリア数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
列がBLOB列の場合にさらに加算します。
＋1
列がユーザ定義型列の場合にさらに加算します。
＋2
＋列で使用しているユーザ定義型の属性数＋1
NO SPLIT指定の場合又はRECOVERY指定の場合にさらに加算します。
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数
データ長の変更の場合にさらに加算します。
表にインデクスが定義されている場合にさらに加算します。
＋1
列回復制約が定義してある場合にさらに加算します。
列がBLOB列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
列がBLOB属性を持つユーザ定義型列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
インナレプリカ機能を使用している場合にさらに加算します。
列がBLOB列の場合にさらに加算します。
＋2

＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
列がBLOB属性を持つユーザ定義型列の場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
CHANGE CLUSTER KEYの場合に加算します。
＋10
＋クラスタキーインデクスの構成列数×2＋1
＋表で使用しているRDエリア数
オブジェクトが無効となるルーチンがある場合にさらに加算します。
＋オブジェクトが無効となるルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
分割表の場合にさらに加算します。
＋表の分割数＋1
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋2
FIX表の場合にさらに加算します。
＋表の構成列数
CHANGE LOCKの場合に加算します。
＋5
＋表で使用しているRDエリア数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
CHANGE HASHの場合に加算します。
＋5
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
FIXハッシュ分割表の場合にさらに加算します。
＋表の分割数＋1
＋表で使用しているRDエリア数＋1
CHANGE SEGMENT REUSEの場合に加算します。
＋4
＋2

＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数
＋表の分割条件数×2
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
SEGMENT REUSE指定の有無を変更する場合にさらに加算します。
＋表で使用しているRDエリア数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
インナレプリカ機能を使用している場合にさらに加算します。
＋2
＋（1＋表に定義してあるインデクス数
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているRDエリア数×（世代数＋1）
CHANGE INSERT ONLYの場合に加算します。
＋5
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数＋1
行削除期間指定ありの場合にさらに加算します。
＋1
DROP列名の場合に加算します。
＋18
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×4
＋表を基に定義されているビュー表の構成列数
＋表を基に定義されているビュー表の使用資源数
＋表を基に定義されているビュー表にアクセス権限を持つユーザ数
表を基に定義されているパブリックビュー表がある場合にさらに加算します。
＋4
表にトリガが定義してある場合にさらに加算します。
＋列を契機列として使用するトリガ数×2
＋表に定義してあるトリガ数×4
＋表に定義してあるトリガの使用するリソース数×2
＋表に定義してあるトリガの使用するルーチン数
＋表に定義してあるトリガの使用するパラメタ数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2
＋列をパラメタとして使用するルーチン数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
＋列を構成列として使用しているインデクス数×2＋2
＋表で使用しているインデクス用RDエリア数＋1

＋列を構成列として使用しているインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋列を構成列として使用しているインデクス数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋列を構成列として使用しているインデクスの除外値数＋1
インナレプリカ機能を使用している場合にさらに加算します。
＋表で使用しているインデクス用RDエリア数×世代数
表にBLOB列が定義されている場合にさらに加算します。
＋表に定義されているBLOB列数
分割表の場合にさらに加算します。
＋表に定義されているBLOB列数×表の分割数
分割表の場合にさらに加算します。
＋表の分割数＋1
マトリクス分割表の場合にさらに加算します。
＋2
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったインデクス用RDエリア数×2＋2
＋表もインデクスも格納しなくなったRDエリア数×2
RENAME TABLEの場合に加算します。
＋7
＋表の構成列数
＋表へのアクセス権限を持つユーザ数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
表に関する統計情報を出力している場合にさらに加算します。
＋2
列に関する統計情報を出力している場合にさらに加算します。
＋統計情報が存在する列数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
横分割表の場合にさらに加算します。
＋表の分割条件数×2＋1
マトリクス分割表の場合にさらに加算します。
＋表の分割条件数＋4
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋表の分割キー数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
部分構造インデクスが定義してある場合にさらに加算します。
＋部分構造インデクス数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋表に定義してあるインデクスの除外値数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
表にユーザ定義型列が定義してある場合にさらに加算します。

＋表に定義してあるユーザ定義型列数＋1
表にインデクス型を使用するインデクスが定義してある場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
変更前の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
変更後の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
RENAME COLUMNの場合に加算します。
＋7
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2＋1
＋列をパラメタとして使用するルーチン数×2
マトリクス分割表の場合にさらに加算します。
＋2
列がインデクス構成列の場合にさらに加算します。
＋列を構成列とするインデクス数＋1
列がプラグインインデクスの構成列の場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
列がBLOB列の場合にさらに加算します。
＋2
分割表の場合にさらに加算します。
＋分割数
列がユーザ定義型列の場合にさらに加算します。
列がインデクス型を使用するインデクスの構成列の場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
CHANGE RDAREA横分割表の場合に加算します。
＋8
＋表の構成列数
＋変更前の分割条件数×2＋変更後の分割条件数×2
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋4
＋表に定義してあるインデクス数
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表から削除するRDエリア数）
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1

＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2
インナレプリカ機能を使用している場合にさらに加算します。
＋変更後の表で使用しているRDエリア数×世代数
＋2
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表から削除するRDエリア数）×（世代数＋1）
CHANGE RDAREAマトリクス分割表の場合に加算します。
＋11
＋表の構成列数
＋変更前の分割条件数＋変更後の分割条件数
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋4
＋表に定義してあるインデクス数
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表から削除するRDエリア数）
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数

表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2
インナレプリカ機能を使用している場合にさらに加算します。
＋変更後の表で使用しているRDエリア数×世代数
＋2
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表から削除するRDエリア数）×（世代数＋1）

(b)　HiRDB/シングルサーバの場合で，主キーの追加，削除のとき

(i) 主キーの追加の場合

17＋対象表の構成列数＋主キーの構成列数×2
＋インデクス格納RDエリア数※×3＋表格納RDエリア数×2
＋5＋2×インデクス格納RDエリア数
対象表がマトリクス分割表の場合に加算します。
＋3＋表の分割キー情報数
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数×3＋1
対象表を参照するトリガ動作手続きがある場合に加算します。
＋対象表を参照するトリガ動作手続き数＋1
インナレプリカ機能を使用している場合に加算します。
対象表にLOB列又はLOB属性が定義されている場合にさらに加算します。
＋LOB列格納用RDエリア数
＋LOB属性格納用RDエリア数
＋1＋（LOB列格納用RDエリア数×LOB列数）×2
＋（LOB属性格納用RDエリア数×LOB属性列数）×2
対象表にインデクスが定義されている場合にさらに加算します。
＋インデクス数
＋（インデクス格納用RDエリア数×インデクス数）×2

(ii) 主キーの削除の場合

17＋主キーの構成列数×2＋インデクス格納RDエリア数※×4
＋インデクス格納RDエリア数×7＋インデクスセグメント数
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数×3＋1
対象表を参照するトリガ動作手続きがある場合に加算します。
＋対象表を参照するトリガ動作手続き数＋1
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×2＋1
＋無効となるルーチンの持つ資源数×無効となるルーチン数
無効となるトリガがある場合に加算します。
＋無効となるトリガ数＋1
＋無効となるトリガ動作条件中で参照している資源情報数＋1
＋無効となるトリガのトリガ動作手続きのパラメタ数＋1

＋無効となるトリガのトリガ契機列情報数＋1
インデクスの最適化情報を取得している場合に加算します。
＋2
インナレプリカ機能を使用している場合に加算します。
＋表格納RDエリア数
　インデクスを定義している表にLOB列又はLOB属性が定義されている場合にさらに加算します。
　＋LOB列格納用RDエリア数
　＋LOB属性格納用RDエリア数
　対象表にLOB列又はLOB属性が定義されている場合にさらに加算します。
　＋1＋（LOB列格納用RDエリア数×LOB列数）×2
　＋（LOB属性格納用RDエリア数×LOB属性列数）×2
　対象表にインデクスが定義されている場合にさらに加算します。
　＋インデクス数
　＋（インデクス格納用RDエリア数×インデクス数）×2
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(c)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）で，主キーの追加，削除以外のとき

1＋オブジェクトが無効となるルーチン数
＋表を基に定義してあるビュー表数
DROP COLUMNの場合に加算します。
＋表に定義してあるトリガ数

(d)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）で，主キーの追加，削除のとき

(i) 主キーの追加の場合

1
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数

(ii) 主キーの削除の場合

1
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数

(e)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，主キーの追加，削除以外のとき

3
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×3＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2
＋定義トリガの契機列数＋1

＋定義トリガで使用するパラメタ数＋1
DROP COLUMNの場合にさらに加算します。
＋表に定義してあるトリガ数
インナレプリカ機能を使用している場合に加算します。
＋表で使用しているRDエリア数＋1
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスで使用するRDエリア数
BLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
＋表に定義してあるBLOB列で使用するRDエリア数
表が分割表の場合にさらに加算します。
＋表の分割数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋表の分割数×分割キーインデクス数＋1
BLOB列が定義してある場合にさらに加算します。
＋表の分割数×BLOB列数＋1
ADD列名の場合に加算します。
＋5
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数
BLOB列を追加する場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数
マトリクス分割表の場合にさらに加算します。
＋3
ユーザ定義型列を追加する場合にさらに加算します。
＋4
＋ユーザ定義型で使用する属性数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
BLOB属性が定義されているユーザ定義型の場合にさらに加算します。
＋表に追加するRDエリア数＋1
＋表で使用しているRDエリア数＋1
＋表のBLOB列数
インデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表のBLOB列数×表の分割数

マトリクス分割表の場合にさらに加算します。
＋3
プラグインの抽象データ型を使用している場合にさらに加算します。
＋使用するプラグイン数＋1
＋使用するプラグインのルーチン数＋1
＋使用するプラグインルーチンのパラメタ数＋1
表がFIX表の場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
非FIX表にNOT NULL列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
列を切り出す場合にさらに加算します。
＋1
DEFAULT句の指定がある列を追加する場合にさらに加算します。
＋表で使用しているRDエリア数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数＋1
表を基に定義したビュー表がある場合にさらに加算します。
＋表を基に定義したビュー表数×2＋2
表が分割表の場合にさらに加算します。
＋表の分割数×2＋2
ADD RDAREAの場合に加算します。
＋9
＋表の構成列数
＋2
＋表に定義してあるインデクス数
FIXハッシュ表の場合にさらに加算します。
＋表で使用している表用RDエリア数
＋表の分割数
表で既に使用しているRDエリアを追加する場合にさらに加算します。
＋既に使用しているRDエリア数
表で未使用のRDエリアを追加する場合にさらに加算します。
＋追加するRDエリア数＋1
インナレプリカ機能を使用している場合にさらに加算します。
＋追加するRDエリア数×世代数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクスの使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×RDエリア追加後の分割数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋2

＋プラグイン列数＋1
＋表で使用しているプラグインの属性数＋1
＋表で使用しているプラグインのルーチン数＋1
＋表で使用しているプラグインルーチンのパラメタ数＋1
＋表で使用しているプラグインのBLOB属性数×RDエリア追加後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
＋表に定義してあるBLOB列数×RDエリア追加後の分割数＋1
＋表に定義してあるBLOB列の使用RDエリア数×2＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるBLOB列数
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
＋表で使用しているユーザ定義型の属性数＋1
ユーザ定義型にBLOB属性が定義してある場合にさらに加算します。
＋表で使用しているユーザ定義型のBLOB属性数＋1
＋表で使用しているユーザ定義型のBLOB属性の使用RDエリア数×2＋1
＋表で使用しているユーザ定義型のBLOB属性数×RDエリア追加後の分割数＋1
表で未使用のRDエリアを追加する場合にさらに加算します。
＋表に定義してあるユーザ定義型列数
CHANGE列名の場合に加算します。
＋11
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表にトリガが定義してある場合にさらに加算します。
＋列をリソースとして使用するトリガ数×2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
＋列をパラメタとして使用するルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
列のデータ型がVARCHAR，NVARCHAR，MVARCHARの場合にさらに加算します。
＋表の構成列数
＋表で使用しているインデクス用RDエリア数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
列がBLOB列の場合にさらに加算します。
＋1
列がユーザ定義型列の場合にさらに加算します。
＋2
＋列で使用しているユーザ定義型の属性数＋1
NO SPLIT指定の場合又はRECOVERY指定の場合にさらに加算します。
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
BLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数
データ長の変更の場合にさらに加算します。
表にインデクスが定義されている場合にさらに加算します。
＋1
CHANGE CLUSTER KEYの場合に加算します。
＋10
＋クラスタキーの構成列数×2＋1
＋表で使用しているRDエリア数

オブジェクトが無効となるルーチンがある場合にさらに加算します。
＋オブジェクトが無効となるルーチン数×2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
分割表の場合にさらに加算します。
＋表の分割数＋1
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋2
FIX表の場合にさらに加算します。
＋表の構成列数
CHANGE LOCKの場合に加算します。
＋5
＋表で使用しているRDエリア数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
CHANGE HASHの場合に加算します。
＋5
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
FIXハッシュ分割表の場合にさらに加算します。
＋表の分割数＋1
＋表で使用しているRDエリア数＋1
CHANGE SEGMENT REUSEの場合に加算します。
＋4
分割表の場合にさらに加算します。
＋表の分割数
＋表の分割条件数×2
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
SEGMENT REUSE指定の有無を変更する場合にさらに加算します。
＋表で使用しているRDエリア数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1

CHANGE INSERT ONLYの場合に加算します。
＋5
＋表で使用しているRDエリア数
分割表の場合にさらに加算します。
＋表の分割数＋1
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×3＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数＋1
行削除期間指定ありの場合にさらに加算します。
＋1
DROP列名の場合に加算します。
＋18
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×4
＋表を基に定義されているビュー表の構成列数
＋表を基に定義されているビュー表の使用資源数
＋表を基に定義されているビュー表にアクセス権限を持つユーザ数
表を基に定義されているパブリックビュー表がある場合にさらに加算します。
＋4
表にトリガが定義してある場合にさらに加算します。
＋列を契機列として使用するトリガ数×2
＋表に定義してあるトリガ数×4
＋表に定義してあるトリガの使用するリソース数×2
＋表に定義してあるトリガの使用するルーチン数
＋表に定義してあるトリガの使用するパラメタ数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2
＋列をパラメタとして使用するルーチン数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表にインデクスが定義してある場合にさらに加算します。
＋表に定義されているインデクス数
列がインデクス構成列の場合にさらに加算します。
＋列を構成列として使用しているインデクス数×2＋2
＋表で使用しているインデクス用RDエリア数＋1
＋列を構成列として使用しているインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×表の分割数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋列を構成列として使用しているインデクス数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋列を構成列として使用しているインデクスの除外値数＋1
インナレプリカ機能を使用している場合にさらに加算します。
＋表で使用しているインデクス用RDエリア数×世代数
表にBLOB列が定義されている場合にさらに加算します。
＋表に定義されているBLOB列数
分割表の場合にさらに加算します。
＋表に定義されているBLOB列数×表の分割数
分割表の場合にさらに加算します。
＋表の分割数＋1
マトリクス分割表の場合にさらに加算します。
＋2
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったインデクス用RDエリア数×2＋2
＋表もインデクスも格納しなくなったRDエリア数×2
RENAME TABLEの場合に加算します。

＋7
＋表の構成列数
＋表へのアクセス権限を持つユーザ数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
表に関する統計情報を出力している場合にさらに加算します。
＋2
列に関する統計情報を出力している場合にさらに加算します。
＋統計情報が存在する列数＋1
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数
分割表の場合にさらに加算します。
＋表の分割数＋1
横分割表の場合にさらに加算します。
＋表の分割条件数×2＋1
マトリクス分割表の場合にさらに加算します。
＋表の分割条件数＋4
ハッシュ混在マトリクス分割表の場合にさらに加算します。
＋表の分割キー数＋1
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
分割キーインデクスが定義してある場合にさらに加算します。
＋分割キーインデクス数×分割数＋1
部分構造インデクスが定義してある場合にさらに加算します。
＋部分構造インデクス数＋1
プラグインインデクスが定義してある場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
インデクスに除外値指定がある場合にさらに加算します。
＋表に定義してあるインデクスの除外値数＋1
インデクスに関する統計情報を出力している場合にさらに加算します。
＋表に定義してあるインデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数＋1
分割表の場合にさらに加算します。
＋表に定義してあるBLOB列数×分割数＋1
表にユーザ定義型列が定義してある場合にさらに加算します。
＋表に定義してあるユーザ定義型列数＋1
表にインデクス型を使用するインデクスが定義してある場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
変更前の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
変更後の表名と同名称のパブリックビュー表が存在する場合にさらに加算します。
＋3
RENAME COLUMNの場合に加算します。
＋7
＋表の構成列数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数×2
列に関する統計情報を出力している場合にさらに加算します。
＋2
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとして使用するルーチン数×2＋1
＋列をパラメタとして使用するルーチン数×2

マトリクス分割表の場合にさらに加算します。
＋2
列がインデクス構成列の場合にさらに加算します。
＋列を構成列とするインデクス数＋1
列がプラグインインデクスの構成列の場合にさらに加算します。
＋プラグインインデクスが使用する関数数＋1
列がBLOB列の場合にさらに加算します。
＋2
分割表の場合にさらに加算します。
＋分割数
列がユーザ定義型列の場合にさらに加算します。
列がインデクス型を使用するインデクスの構成列の場合にさらに加算します。
＋インデクスで使用するユーザ定義型数＋1
ユーザ定義型にBLOB属性を定義している場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数＋1
分割表の場合にさらに加算します。
＋ユーザ定義型で使用するBLOB属性数×分割数
CHANGE RDAREA 横分割表の場合に加算します。
＋8
＋表の構成列数
＋変更前の分割条件数×2＋変更後の分割条件数×2
＋変更前の分割数＋変更後の分割数
＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋2
＋表に定義してあるインデクス数
インナレプリカ機能を使用している場合にさらに加算します。
＋変更後の表で使用しているRDエリア数×世代数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2
CHANGE RDAREA マトリクス分割表の場合に加算します。
＋11
＋表の構成列数
＋変更前の分割条件数＋変更後の分割条件数
＋変更前の分割数＋変更後の分割数

＋変更前の表で使用しているRDエリア数
＋変更後の表で使用しているRDエリア数×2
＋2
＋表に定義してあるインデクス数
インナレプリカ機能を使用している場合にさらに加算します。
＋変更後の表で使用しているRDエリア数×世代数
表をリソースとして使用するルーチンがある場合にさらに加算します。
＋表をリソースとするルーチン数
表を基に定義されているビュー表がある場合にさらに加算します。
＋表を基に定義されているビュー表数
表にインデクスが定義してある場合にさらに加算します。
＋表に定義してあるインデクス数＋1
＋表に定義してあるインデクスの構成列数＋1
＋表に定義してあるインデクス数×変更前の分割数＋1
＋表に定義してあるインデクス数×変更後の分割数
部分構造インデクスが定義してある場合にさらに加算します。
＋表に定義してある部分構造インデクス数＋1
表にBLOB列が定義してある場合にさらに加算します。
＋表に定義してあるBLOB列数×変更前の分割数＋1
＋表に定義してあるBLOB列数×変更後の分割数
表を被参照表とする参照制約が定義してある場合にさらに加算します。
＋表を被参照表とする参照制約数＋1
＋表を被参照表とする参照表数
表を参照表とする参照制約が定義してある場合にさらに加算します。
＋表の参照制約数＋1
表に検査制約が定義してある場合にさらに加算します。
＋表の検査制約数＋1
再編成時期予測機能を使用している場合にさらに加算します。
＋表で使用しなくなったRDエリア数×2＋2
＋データを削除されるRDエリア数
＋表もインデクスも格納しなくなったRDエリア数×2

(f) HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，主キーの追加，削除のとき

(i) 主キーの追加の場合

11＋対象表の構成列数＋主キーの構成列数
＋インデクス格納RDエリア数※×3＋表格納RDエリア数×2
対象表がマトリクス分割表の場合に加算します。
＋3＋表の分割キー情報数
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数×3＋1
対象表を参照するトリガ動作手続きがある場合に加算します。
＋対象表を参照するトリガ動作手続き数＋1
インナレプリカ機能を使用している場合に加算します。
対象表にLOB列又はLOB属性が定義されている場合にさらに加算します。
＋1＋（LOB列格納用RDエリア数×LOB列数）×2
＋（LOB属性格納用RDエリア数×LOB属性列数）×2
対象表にインデクスが定義されている場合にさらに加算します。
＋インデクス数
＋（インデクス格納用RDエリア数×インデクス数）×2

(ii) 主キーの削除の場合

11＋主キーの構成列数×2＋インデクス格納RDエリア数※×4
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数×2＋1
対象表を参照するトリガ動作手続きがある場合に加算します。
＋対象表を参照するトリガ動作手続き数＋1
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×2＋1
＋無効となるルーチンの持つ資源数×無効となるルーチン数
無効となるトリガがある場合に加算します。
＋無効となるトリガ数＋1
＋無効となるトリガ動作条件中で参照している資源情報数＋1
＋無効となるトリガのトリガ動作手続きのパラメタ数＋1
＋無効となるトリガのトリガ契機列情報数＋1
インデクスの最適化情報を取得している場合に加算します。
＋2
インナレプリカ機能を使用している場合に加算します。
　対象表にLOB列又はLOB属性が定義されている場合にさらに加算します。
　＋1＋（LOB列格納用RDエリア数×LOB列数）×2
　＋（LOB属性格納用RDエリア数×LOB属性列数）×2
　対象表にインデクスが定義されている場合にさらに加算します。
　＋インデクス数
　＋（インデクス格納用RDエリア数×インデクス数）×2
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(g)　HiRDB/パラレルサーバの場合（バックエンドサーバ）で，主キーの追加，削除以外のとき

オブジェクトが無効となるルーチン数
DROP COLUMNの場合に加算します。
＋表に定義してあるトリガ数
ADD 列名の場合に加算します。
　BLOB列を追加する場合にさらに加算します。
　＋2
　＋（1＋表に定義してあるインデクス数
　＋表に定義してあるBLOB列数
　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
　BLOB属性を持つユーザ定義型列を追加する場合にさらに加算します。
　＋2
　＋（1＋表に定義してあるインデクス数
　＋表に定義してあるBLOB列数
　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
　インナレプリカ機能を使用している場合にさらに加算します。
　　BLOB列を追加する場合にさらに加算します。
　　＋2
　　＋（1＋表に定義してあるインデクス数
　　＋表に定義してあるBLOB列数
　　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
　　BLOB属性を持つユーザ定義型列を追加する場合にさらに加算します。
　　＋2
　　＋（1＋表に定義してあるインデクス数
　　＋表に定義してあるBLOB列数
　　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）

ADD RDAREAの場合に加算します。
＋4
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数
＋表に定義してあるユーザ定義型のBLOB属性数
　インナレプリカ機能を使用している場合にさらに加算します。
　＋2
　＋（表に定義してあるインデクス数×2
　＋表に定義してあるBLOB列数＋2
　＋表に定義してあるユーザ定義型のBLOB属性数）×（表で使用しているRDエリア数＋1）×（世代数＋1）
CHANGE 列名の場合に加算します。
列回復制約が定義してある場合に加算します。
　列がBLOB列の場合にさらに加算します。
　＋2
　＋（1＋表に定義してあるインデクス数
　＋表に定義してあるBLOB列数
　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
　列がBLOB属性を持つユーザ定義型列の場合にさらに加算します。
　＋2
　＋（1＋表に定義してあるインデクス数
　＋表に定義してあるBLOB列数
　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数
　インナレプリカ機能を使用している場合にさらに加算します。
　　列がBLOB列の場合にさらに加算します。
　　＋2
　　＋（1＋表に定義してあるインデクス数
　　＋表に定義してあるBLOB列数
　　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
　　列がBLOB属性を持つユーザ定義型列の場合にさらに加算します。
　　＋2
　　＋（1＋表に定義してあるインデクス数
　　＋表に定義してあるBLOB列数
　　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているLOB用RDエリア数×（世代数＋1）
CHANGE SEGMENT REUSEの場合に加算します。
＋2
＋表で使用するRDエリア数
　インナレプリカ機能を使用している場合にさらに加算します。
　＋2
　＋（1＋表に定義してあるインデクス数
　＋表に定義してあるBLOB列数
　＋表に定義してあるユーザ定義型のBLOB属性数）×表で使用しているRDエリア数×（世代数＋1）
CHANGE RDAREAの場合に加算します。
＋2
＋削除対象インデクス数
＋（2
＋表に定義してあるインデクス数×2
＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表から削除するRDエリア数）
　インナレプリカ機能を使用している場合にさらに加算します。
　＋2
　＋削除対象インデクス数
　＋（2
　＋表に定義してあるインデクス数×2

＋表に定義してあるBLOB列数）×（表に追加するRDエリア数
＋表から削除するRDエリア数）×（世代数＋1）

(h)　HiRDB/パラレルサーバの場合（バックエンドサーバ※）で，主キーの追加，削除のとき

(i) 主キーの追加の場合

5＋インデクス格納RDエリア数×2
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数
インナレプリカ機能を使用している場合に加算します。
対象表にLOB列又はLOB属性が定義されている場合にさらに加算します。
＋LOB列格納用RDエリア数
＋LOB属性格納用RDエリア数

(ii) 主キーの削除の場合

5＋インデクス格納RDエリア数×4＋インデクスセグメント数
対象表を参照するルーチンがある場合に加算します。
＋対象表を参照するルーチン数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数
インナレプリカ機能を使用している場合に加算します。
＋表格納RDエリア数
インデクスを定義している表にLOB列又はLOB属性が定義されている場合にさらに加算します。
＋LOB列格納用RDエリア数
＋LOB属性格納用RDエリア数

(i)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ※）

2

注※　インナレプリカ機能を使用する場合は世代数分必要になります。

(5)　ALTER TRIGGER

(a)　HiRDB/シングルサーバの場合

19
実行者が所有者以外の場合に加算します。
＋2
トリガ契機列がある場合に加算します。
＋トリガの契機列数×2＋1
トリガ動作条件指定がある場合に加算します。
＋トリガの動作条件中で使用している資源情報数×2＋1
トリガが使用するリソースがある場合に加算します。
＋使用するリソース数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数
使用するリソースにビュー表がある場合に加算します。

＋使用するビュー表数
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型数＋1
パラメタを指定している場合に加算します。
＋パラメタ数×2＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

17
実行者が所有者以外の場合に加算します。
＋2
トリガ契機列がある場合に加算します。
＋トリガの契機列数×2＋1
トリガ動作条件指定がある場合に加算します。
＋トリガの動作条件中で使用している資源情報数×2＋1
トリガが使用するリソースがある場合に加算します。
＋使用するリソース数×2＋1
使用するリソースに表がある場合に加算します。
＋使用する表数
使用するリソースにビュー表がある場合に加算します。
＋使用するビュー表数
＋使用するビュー表で使用する利用資源数＋1
使用するリソースにインデクスがある場合に加算します。
＋使用するインデクス数＋1
使用するリソースにルーチンがある場合に加算します。
＋使用するルーチン数
使用するリソースにユーザ定義型がある場合に加算します。
＋使用するユーザ定義型数＋1
パラメタを指定している場合に加算します。
＋パラメタ数×2＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1

(6)　COMMENT

(a)　HiRDB/シングルサーバの場合

5
コメントが列の場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5
コメントが列の場合に加算します。
＋2

(7)　CREATE AUDIT

(a)　HiRDB/シングルサーバの場合

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(8)　CREATE CONNECTION SECURITY

(a)　HiRDB/シングルサーバの場合

FOR CONNECT, PASSWORD指定の場合に加算します。
12＋ディクショナリ表SQL_USERSに登録されているユーザ数
FOR CONSTRAINT指定の場合に加算します。
＋3
FOR INTERVAL PASSWORD指定の場合に加算します。
＋5

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

FOR CONNECT, PASSWORD指定の場合に加算します。
12＋ディクショナリ表SQL_USERSに登録されているユーザ数
FOR CONSTRAINT指定の場合に加算します。
＋3
FOR INTERVAL PASSWORD指定の場合に加算します。
＋5

(9)　CREATE 〔PUBLIC〕FUNCTION

(a)　HiRDB/シングルサーバの場合

CREATE 〔PUBLIC 〕PROCEDUREの排他資源数＋オブジェクトが無効となるルーチン数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2
＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びバックエンドサーバ）

CREATE 〔PUBLIC 〕PROCEDUREの排他資源数＋オブジェクトが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

CREATE 〔PUBLIC 〕PROCEDUREの排他資源数＋オブジェクトが無効となるルーチン数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2
＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(10)　CREATE INDEX（プラグインインデクスではない場合）

(a)　HiRDB/シングルサーバの場合※

22＋インデクスが無効となるルーチン数＋構成列数×4
一時インデクスではない場合に加算します。
＋5＋2×インデクス格納RDエリア数
＋インデクス格納RDエリア数×４
インデクスを定義する表を参照するルーチン数がある場合に加算します。
＋インデクスを定義する表を参照するルーチン数×4＋2
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクス格納RDエリアが二つ以上の場合に加算します。
＋インデクス格納RDエリア数×3＋1
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場

合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋インデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

21＋構成列数×4
一時インデクスではない場合に加算します。
＋インデクス格納RDエリア数×４
インデクスを定義する表を参照するルーチン数がある場合に加算します。
＋インデクスを定義する表を参照するルーチン数×4＋2
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクス格納RDエリアが二つ以上の場合に加算します。
＋インデクス格納RDエリア数×3＋1＋インデクスが無効となるルーチン数
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ※）で一時インデクスではないとき

5＋インデクス格納RDエリア数×2＋インデクスが無効となるルーチン数
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数

(e)　HiRDB/パラレルサーバの場合（バックエンドサーバ）で一時インデクスのとき

インデクスが無効となるルーチン数

(f)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ※）

2

注※　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

(11)　CREATE INDEX（プラグインインデクスの場合）

(a)　HiRDB/シングルサーバの場合※

21
＋インデクスが無効となるルーチン数
＋インデクス格納RDエリア数×8
＋構成列数×4
＋適用関数数×5

＋Σ適用関数パラメタ数
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋インデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

15
＋インデクス格納RDエリア×6
＋構成列数×4
＋適用関数数×5
＋Σ適用関数パラメタ数
＋インデクスが無効となるルーチン数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ※）

5＋インデクス格納RDエリア数×2＋インデクスが無効となるルーチン数
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数

注※　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

(12)　CREATE 〔PUBLIC〕PROCEDURE

(a)　HiRDB/シングルサーバの場合

15
＋SQL文前処理でアクセスする表数
＋SQL文前処理でアクセスするビュー表数
＋SQL文前処理でアクセスするビュー表の基となる実表数
使用するリソースがある場合に加算します。
＋使用するリソース数×7＋3
パラメタを指定している場合に加算します。
＋パラメタ数×3＋1
パラメタのデータ型がユーザ定義型の場合に加算します。
＋ユーザ定義型のパラメタ数×2＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3
＋SQL文前処理でアクセスする表数
＋SELECT文前処理でアクセスするビュー表数
＋SELECT文前処理でアクセスするビュー表の基となる実表数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

17
使用するリソースがある場合に加算します。
＋使用するリソース数×7＋3
パラメタを指定している場合に加算します。
＋パラメタ数×3＋1
パラメタのデータ型がユーザ定義型の場合に加算します。
＋ユーザ定義型のパラメタ数×2＋1

(13)　CREATE ROLE

(a)　HiRDB/シングルサーバの場合

12

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

12

(14)　CREATE SCHEMA

(a)　HiRDB/シングルサーバの場合

5

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5

(15)　CREATE SEQUENCE

(a)　HiRDB/シングルサーバの場合

11＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

11

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

2

(16)　CREATE TABLE

(a)　HiRDB/シングルサーバの場合※で，表を格納するRDエリアを省略し，かつ非横分割表（一時表を含む）のとき

22＋列数×3
一時表ではない場合に加算します。
＋9
＋公用RDエリア数×3
＋LOB列数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6
＋LOB属性数×分割RDエリア数×4
＋ユーザ定義型属性数×2
＋5
クラスタキーを定義している場合に加算します。
＋8
＋構成列数×3
　一時表ではない場合にさらに加算します。
　＋5
　＋インデクス格納RDエリア数×2
　＋分割RDエリア数×6
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
　＋被参照表を参照する関数の総数
検査制約を定義している場合に加算します。
＋検査制約数×9

(b)　HiRDB/シングルサーバの場合※で，表を格納するRDエリアを指定し，かつ非横分割表のとき

30＋列数×4＋LOB列数×4＋表格納RDエリア数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6
＋LOB属性数×分割RDエリア数×4
＋ユーザ定義型属性数×2
＋5
クラスタキーを定義している場合に加算します。
＋5
＋インデクス格納RDエリア数×2
＋分割RDエリア数×6
＋8
＋構成列数×3
参照制約を定義している場合に加算します。

＋被参照表数
＋外部キー数×10
参照制約動作にCASCADEが定義されている場合にさらに加算します。
＋CASCADE数×17
＋CASCADEで指定した被参照数×4
参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
＋CASCADEを指定した被参照表の主キー構成列の合計数×8
＋CASCADEで指定した被参照数×3
無効となる関数がある場合にさらに加算します。
＋オブジェクトが無効となる関数の数
＋Σ（2＋プロシジャ内リソース数×5）
＋被参照表を参照する関数の総数
検査制約を定義している場合に加算します。
＋検査制約数×9

(c)　HiRDB/シングルサーバの場合※で，横分割表のとき

31＋列数×4＋分割RDエリア数×21＋LOB列数×分割RDエリア数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6
＋LOB属性数×分割RDエリア数×4
＋ユーザ定義型属性数×2
＋5
クラスタキーを定義している場合に加算します。
＋5
＋インデクス格納RDエリア数×2
＋分割RDエリア数×6
＋8
＋構成列数×3
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
参照制約動作にCASCADEが定義されている場合にさらに加算します。
＋CASCADE数×17
＋CASCADEで指定した被参照数×4
参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
＋CASCADEを指定した被参照表の主キー構成列の合計数×8
＋CASCADEで指定した被参照数×3
無効となる関数がある場合にさらに加算します。
＋オブジェクトが無効となる関数の数
＋Σ（2＋プロシジャ内リソース数×5）
＋被参照表を参照する関数の総数
検査制約を定義している場合に加算します。
＋検査制約数×9

(d)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3
参照制約を定義している場合に加算します。
＋被参照表数
＋被参照表を参照する総ルーチン数

(e) HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，表を格納するRDエリアを省略し，かつ非横分割表（一時表を含む）のとき

```
22＋列数×3
一時表ではない場合に加算します。
＋公用RDエリア数×3
＋LOB列数×分割RDエリア数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6
＋LOB属性数×分割RDエリア数×4
＋ユーザ定義型属性数×2
＋5
クラスタキーを定義している場合に加算します。
＋8＋構成列数×3
  一時表ではない場合にさらに加算します。
  ＋分割RDエリア数×6
参照制約を定義している場合に加算します。
＋外部キー数×10
  参照制約動作にCASCADEが定義されている場合にさらに加算します。
  ＋CASCADE数×17
  ＋CASCADEで指定した被参照数×4
  参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
  ＋CASCADEを指定した被参照表の主キー構成列の合計数×8
  ＋CASCADEで指定した被参照数×3
  無効となる関数がある場合にさらに加算します。
  ＋オブジェクトが無効となる関数の数
  ＋Σ（2＋プロシジャ内リソース数×5）
検査制約を定義している場合に加算します。
＋検査制約数×9
```

(f) HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，表を格納するRDエリアを指定し，かつ非横分割表のとき

```
27＋列数×4＋LOB列数×4
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6
＋LOB属性数×分割RDエリア数×4
＋ユーザ定義型属性数×2
＋5
クラスタキーを定義している場合に加算します。
＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋外部キー数×10
  参照制約動作にCASCADEが定義されている場合にさらに加算します。
  ＋CASCADE数×17
  ＋CASCADEで指定した被参照数×4
  参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
  ＋CASCADEを指定した被参照表の主キー構成列の合計数×8
  ＋CASCADEで指定した被参照数×3
  無効となる関数がある場合にさらに加算します。
  ＋オブジェクトが無効となる関数の数
  ＋Σ（2＋プロシジャ内リソース数×5）
検査制約を定義している場合に加算します。
＋検査制約数×9
```

(g)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）で，横分割表のとき

マトリクス分割表ではない場合に加算します。
26＋列数×4＋分割RDエリア数×17＋LOB列数×分割RDエリア数×4
マトリクス分割表の場合に加算します。
23＋列数×4
＋分割RDエリア数（重複排除しない数）×16
＋LOB列数×分割RDエリア数×4
＋分割キー数×2
＋（格納条件数＋2）×2
＋2
ユーザ定義型を使用した場合に加算します。
＋ユーザ定義型数×6＋LOB属性数×分割RDエリア数×4＋ユーザ定義型属性数×2＋5
クラスタキーを定義している場合に加算します。
＋分割RDエリア数×6＋8＋構成列数×3
参照制約を定義している場合に加算します。
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
　無効となる関数がある場合にさらに加算します。
　＋オブジェクトが無効となる関数の数
　＋Σ（2＋プロシジャ内リソース数×5）
検査制約を定義している場合に加算します。
＋検査制約数×9

(h)　HiRDB/パラレルサーバの場合（バックエンドサーバ※）で，一時表ではないとき

5＋表格納RDエリア数×4
クラスタキーを定義している場合に加算します。
＋5＋インデクス格納RDエリア数×2

(i)　HiRDB/パラレルサーバの場合（バックエンドサーバ）で，一時表のとき

0

(j)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ※）

2
クラスタキーを定義している場合に加算します。
＋1

注※　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

(17)　CREATE TRIGGER

(a)　HiRDB/シングルサーバの場合

23
＋オブジェクトが無効になるルーチン数×5
＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋オブジェクトが無効になるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

20＋オブジェクトが無効になるルーチン数＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効になるルーチン数

(18)　CREATE TYPE

(a)　HiRDB/シングルサーバの場合

24＋無効となるルーチン数＋属性数×3
属性にユーザ定義型がある場合に加算します。
＋（ユーザ定義型数×6＋2）
サブタイプ定義の場合に加算します。
＋11
無効となるルーチンがある場合に加算します。

＋（無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1）
無効となるルーチンをビュー表で使用している場合にさらに加算します。
＋ビュー表で使用している無効となるルーチン数＋1
＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数又は手続き定義をする場合に加算します。
＋関数・手続き定義分排他資源数※

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋無効となるルーチン数
関数又は手続き定義をする場合に加算します。
＋関数・手続き定義数分排他資源数※

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

23＋属性数×3
属性にユーザ定義型がある場合に加算します。
＋（ユーザ定義型数×6＋2）
サブタイプ定義の場合に加算します。
＋11
無効となるルーチンがある場合に加算します。
＋（無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1）
無効となるルーチンをビュー表で使用している場合にさらに加算します。
＋ビュー表で使用している無効となるルーチン数＋1
＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数又は手続き定義をする場合に加算します。
＋関数又は手続き定義数分排他資源数※

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

無効になるルーチン数

注※　関数定義又はプロシジャ定義の排他要求数を参照してください。

(19)　CREATE 〔PUBLIC〕VIEW

(a)　HiRDB/シングルサーバの場合

33
＋SELECT文前処理でアクセスする表数
＋SELECT文前処理でアクセスするビュー表数
＋SELECT文前処理でアクセスするビュー表の基となる実表数
＋列数×4
＋ビュー表の基表数×4
＋SELECT文前処理の数
パブリックビュー表の場合に加算します。
＋2
＋パブリックビュー表と同じ名称の表数×2
＋パブリックビュー表と同じ名称のビュー表数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3
＋SELECT文前処理でアクセスする表数
＋SELECT文前処理でアクセスするビュー表数
＋SELECT文前処理でアクセスするビュー表の基となる実表数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

30＋列数×4＋ビュー表の基表数×4＋SELECT文前処理の数
パブリックビュー表の場合に加算します。
＋2
＋パブリックビュー表と同じ名称の表数×2
＋パブリックビュー表と同じ名称のビュー表数×2

(20)　DROP AUDIT

(a)　HiRDB/シングルサーバの場合

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7
オブジェクト指定ありの場合に加算します。
＋該当するオブジェクトの定義数
オブジェクト指定なしの場合に加算します。
＋オブジェクト指定のない定義数

(21)　DROP CONNECTION SECURITY

(a)　HiRDB/シングルサーバの場合

FOR CONNECT，PASSWORD指定の場合に加算します。
12＋ディクショナリ表SQL_USERSに登録されているユーザ数
FOR CONSTRAINT指定の場合に加算します。
＋3
FOR INTERVAL PASSWORD指定の場合に加算します。
＋5

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

FOR CONNECT，PASSWORD指定の場合に加算します。
12＋ディクショナリ表SQL_USERSに登録されているユーザ数
FOR CONSTRAINT指定の場合に加算します。

＋3
FOR INTERVAL PASSWORD指定の場合に加算します。
＋5

(22)　DROP DATA TYPE

(a)　HiRDB/シングルサーバの場合

26＋無効となるルーチン数＋属性数×3
サブタイプの場合に加算します。
＋6
ユーザ定義型の属性がある場合に加算します。
＋ユーザ定義型属性数×4
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数定義がある場合に加算します。
＋関数定義数×7＋1＋関数のパラメタ数×4＋1
＋関数で使用する手続き数×5＋1
手続き定義がある場合に加算します。
＋手続き定義数×5＋1＋手続きのパラメタ数×4＋1
＋手続きで使用するリソース数×5＋1
トリガ動作条件に指定している場合に加算します。
＋15
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

23＋属性数×3＋オブジェクトが無効となるルーチン数
サブタイプの場合に加算します。
＋6
ユーザ定義型の属性がある場合に加算します。
＋ユーザ定義型属性数×4
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×4＋2＋無効となるルーチンのリソース数×5＋1
　無効となるルーチンをビュー表で使用している場合にさらに加算します。
　＋ビュー表で使用している無効となるルーチン数＋1
　＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1
関数定義がある場合に加算します。
＋関数定義数×7＋1＋関数のパラメタ数×4＋1＋関数で使用する手続き数×5＋1
手続き定義がある場合に加算します。
＋手続き定義数×5＋1＋手続きのパラメタ数×4＋1＋手続きで使用するリソース数×5＋1
トリガ動作条件に指定している場合に加算します。
＋15
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数

(23)　DROP 〔PUBLIC〕FUNCTION

(a)　HiRDB/シングルサーバの場合

DROP 〔PUBLIC 〕PROCEDUREの排他資源数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
トリガ動作条件に指定している場合に加算します。
＋15
無効となるルーチンをビュー表で使用している場合に加算します。
＋ビュー表で使用している無効となるルーチン数＋1＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ，及びバックエンドサーバ）

DROP 〔PUBLIC 〕PROCEDUREと同じ

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

DROP 〔PUBLIC 〕PROCEDUREの排他資源数
パラメタがある場合に加算します。
＋パラメタ数×2
パラメタにユーザ定義型がある場合に加算します。
＋ユーザ定義型数×2
トリガ動作条件に指定している場合に加算します。
＋15
無効となるルーチンをビュー表で使用している場合に加算します。
＋ビュー表で使用している無効となるルーチン数＋1＋ビュー表で使用している無効となるルーチン数×パラメタ数＋1

(24)　DROP INDEX（プラグインインデクスではない場合）

(a)　HiRDB/シングルサーバの場合※

22
＋オブジェクト又はインデクスが無効となるルーチン数
＋構成列数×3
一時インデクスではない場合に加算します。
＋インデクス格納RDエリア数×７
＋インデクスセグメント数
一時インデクスの場合に加算します。
＋3
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×7＋1

インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクスの最適化情報を取得している場合に加算します。
＋4
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクト又はインデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

16
＋構成列数×3
＋オブジェクト又はインデクスが無効となるルーチン数
一時インデクスではない場合に加算します。
＋インデクス格納RDエリア数×3
無効となるルーチンがある場合に加算します。
無効となるルーチン数×7＋1
インデクスを定義した表を参照するトリガ動作手続きがある場合に加算します。
＋インデクスを定義した表を参照するトリガ動作手続き数×5
インデクスの最適化情報を取得している場合に加算します。
＋4
インデクスが部分構造インデクスの場合に加算します。
＋インデクス構成部分構造パス数×3
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ※）で一時インデクスではないとき

5
＋インデクス格納RDエリア数×4
＋インデクスセグメント数
＋オブジェクト又はインデクスが無効となるルーチン数
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数

(e)　HiRDB/パラレルサーバの場合（バックエンドサーバ）で一時インデクスのとき

オブジェクト又はインデクスが無効となるルーチン数

(f)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ※）

2

注※　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

(25)　DROP INDEX（プラグインインデクスの場合）

(a)　HiRDB/シングルサーバの場合※

15
＋オブジェクト又はインデクスが無効となるルーチン数
＋インデクス格納RDエリア数×10
＋構成列数×4
＋適用関数数×3
＋インデクスセグメント数
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクト又はインデクスが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

9
＋インデクス格納RDエリア数×6
＋構成列数×4
＋適用関数数×3
＋オブジェクト又はインデクスが無効となるルーチン数
再編成時期予測機能を使用している場合に加算します。
＋インデクス格納RDエリア数×62＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ※）

5
＋インデクス格納RDエリア数×4
＋インデクスセグメント数
＋オブジェクト又はインデクスが無効となるルーチン数
インナレプリカ機能を使用していて，インデクスを定義した表にLOB列又はLOB属性が定義されている場合に加算します。
＋LOB列格納RDエリア数＋LOB属性格納RDエリア数

注※　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

(26) DROP 〔PUBLIC〕PROCEDURE

(a) HiRDB/シングルサーバの場合

19＋パラメタ数×5＋リソース数×10＋オブジェクトが無効になるルーチン数
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×4＋2＋無効になるルーチンのリソース数×5＋1
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b) HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋オブジェクトが無効となるルーチン数

(c) HiRDB/パラレルサーバの場合（ディクショナリサーバ）

18＋パラメタ数×3＋リソース数×10＋オブジェクトが無効となるルーチン数
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×4＋2＋無効になるルーチンのリソース数×5＋1
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d) HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数

(27) DROP ROLE

(a) HiRDB/シングルサーバの場合

8＋指定したロールのロール利用権限を与えた認可識別子の総数＋指定したロールに付与したアクセス権限の総数

(b) HiRDB/パラレルサーバの場合（ディクショナリサーバ）

8＋指定したロールのロール利用権限を与えた認可識別子の総数＋指定したロールに付与したアクセス権限の総数

(28) DROP SCHEMA

(a) HiRDB/シングルサーバの場合※1

11
＋スキーマ内の表数
＋スキーマ内のビュー表数
＋スキーマ内のルーチン数
＋スキーマ内の表又はビュー表を基表にしたほかのスキーマのビュー表数

＋削除する表，ビュー表，ルーチン，又はデータ型を参照するほかのスキーマのインデクス数
スキーマ内の表削除分を加算します（マトリクス分割表ではない場合）。
＋Σ｛表格納RDエリア数×3
＋分割条件指定数×9
＋列数×4
＋（LOB列数×4）×表格納RDエリア数
＋1
＋8｝
＋2
＋Σ（表格納RDエリア数×10
＋10
＋表データの使用中セグメント数）
スキーマ内の表削除分を加算します（マトリクス分割表の場合）。
＋Σ｛表格納RDエリア数×3
＋表格納RDエリア指定数（重複排除しない数）×12
＋列数×4
＋（LOB列数×4）×表格納RDエリア数
＋1
＋8
＋削除表の権限定義数
＋分割キー数×2
＋（格納条件指定数＋2）×2
＋2｝
＋3
＋Σ（表格納RDエリア数×10
＋10
＋表データの使用中セグメント数）
スキーマ内のインデクス削除分を加算します。
＋3
＋Σ（インデクス格納RDエリア数×7
＋構成列数×3
＋インデクス構成部分構造パス数×3※2）
＋Σ（インデクス格納RDエリア数×8
＋10
＋インデクスの使用中セグメント数）
＋8
最適化情報を収集している場合に加算します。
＋最適化情報収集表数×2＋最適化情報収集列数×3＋最適化情報収集インデクス数×3
表削除に伴い削除されるビュー表分を加算します。
＋2＋Σ（12＋ビュー表の基表数×4＋列数×4）
スキーマ内の表，ビュー表，ルーチン，又はデータ型を使用しているルーチン分を加算します。
＋3＋Σ（5＋プロシジャ内リソース数×5＋パラメタ定義数×3）
無効となるプロシジャがある場合に加算します。
＋無効となるプロシジャ数＋Σ（2＋リソース数×5）＋スキーマ内に定義されたデータ型数分※3
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
トリガを定義した表がある場合に加算します。
＋スキーマ内トリガ定義数×8
＋スキーマ内UPDATEトリガ定義の総指定列数×3
＋スキーマ内トリガ動作条件があるトリガ数×3
＋スキーマ内トリガ動作条件中で参照する資源種類数×5
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋Σ｛（表格納RDエリア数

＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62}
＋Σ｛（インデクス格納RDエリア数
＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62}
＋1
スキーマ内に順序数生成子がある場合に加算します。
＋2＋スキーマ内の順序数生成子数×3
＋スキーマ内の順序数生成子数×2
削除するスキーマ操作権限がある場合に加算します。
＋1＋削除対象のスキーマ操作権限を持っているユーザ数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3
＋スキーマ内の表数
＋スキーマ内のビュー表数
＋スキーマ内のルーチン数
＋スキーマ内の順序数生成子数
＋スキーマ内の表，又はビュー表を基表にしたほかのスキーマのビュー表数
＋削除する表，ビュー表，ルーチン，又はデータ型を参照するほかのスキーマのルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

11＋ほかのスキーマのインデクス数
スキーマ内の表削除分を加算します（マトリクス分割表ではない場合）。
＋Σ｛表格納RDエリア数×3
＋分割条件指定数×9
＋列数×4
＋（LOB列数×4）×表格納RDエリア数
＋削除する表への権限定義数×3
＋6}
＋11
スキーマ内の表削除分を加算します（マトリクス分割表の場合）。
＋Σ｛表格納RDエリア数×3
＋表格納RDエリア指定数（重複排除しない数）×12
＋列数×4
＋（LOB列数×4）×表格納RDエリア数
＋削除表の権限定義数×3
＋1
＋6
＋分割キー数×2
＋（格納条件指定数＋2）×2
＋2}
＋3
スキーマ内のインデクス削除分を加算します。
＋4
＋Σ（インデクス格納RDエリア数×7
＋構成列数×7
＋インデクス構成部分構造パス数×3※2)
＋インデクス数×5
最適化情報を収集している場合に加算します。
＋最適化情報収集表数×2
＋表の最適化情報収集列数×3
＋最適化情報収集インデクス数×3
表削除に伴い削除されるビュー表分を加算します。
＋2

＋Σ（18＋ビュー表の基表数×4＋列数×4＋ビュー表への権限定義数×3）
スキーマ内の表，ビュー表，ルーチン，データ型を使用しているルーチン分を加算します。
＋3
＋Σ（5＋プロシジャ内リソース数×10＋パラメタ定義数×3）
無効となるプロシジャがある場合に加算します。
＋無効となるプロシジャ数
＋Σ（5＋リソース数×5）
＋スキーマ内に定義されたデータ型数分※3
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
トリガを定義した表がある場合に加算します。
＋スキーマ内トリガ定義数×8
＋スキーマ内UPDATEトリガ定義の総指定列数×3
＋スキーマ内トリガ動作条件があるトリガ数×3
＋スキーマ内トリガ動作条件中で参照する資源種類数×5
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋Σ（（表格納RDエリア数＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62）
＋Σ（（インデクス格納RDエリア数＋（LOB列数＋LOB属性数）×表格納用RDエリア数）×62）＋1
スキーマ内に順序数生成子がある場合に加算します。
＋2＋スキーマ内の順序数生成子数×2
削除するスキーマ操作権限がある場合に加算します。
＋1＋削除対象のスキーマ操作権限を持っているユーザ数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ※1）

8
＋オブジェクト又はインデクスオブジェクトが無効となるルーチン数
一時表及び一時インデクスではない場合に加算します。
＋Σ（表格納RDエリア数×10＋10＋表データの使用中セグメント数）
＋Σ（インデクス格納RDエリア数×8＋10＋インデクスの使用中セグメント数）
スキーマ内に順序数生成子が存在する場合に加算します。
＋スキーマ内の順序数生成子数×2

(e)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ※1）

表数＋インデクス格納RDエリア数

注※1　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

注※2　インデクスが部分構造インデクスの場合に繰り返し加算します。

注※3　ユーザ定義型削除時の排他要求数を参照してください。

(29)　DROP SEQUENCE

(a)　HiRDB/シングルサーバの場合

9＋2
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×3＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2
＋2
＋定義トリガ契機列数
＋1
＋定義トリガで使用するパラメタ数
＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

8
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数×2＋2
トリガが定義されている場合に加算します。
＋トリガ定義数×2＋2＋定義トリガ契機列数＋1＋定義トリガで使用するパラメタ数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

2

(30)　DROP TABLE

(a)　HiRDB/シングルサーバの場合※1

マトリクス分割表ではない場合に加算します。
24
＋表を基に定義したビュー表数
＋削除する表又はビュー表を参照するルーチン数
＋列数×4
　一時表ではない場合にさらに加算します。
　＋5
　＋表格納RDエリア数×4
　＋表データの使用中セグメント数
　＋表格納RDエリア数×3
　＋分割条件指定数×9
マトリクス分割表の場合に加算します。
22

＋表を基に定義したビュー表数
＋削除する表又はビュー表を参照するルーチン数
＋表格納RDエリア数×3
＋表格納RDエリア数（重複排除しない数）×8
＋列数×4
＋5
＋表格納RDエリア数×4
＋表データの使用中セグメント数
＋分割キー数×2
＋（格納条件指定数＋2）×2
＋2
インデクスを定義している場合に加算します。
＋Σ（構成列数×3）＋3
　一時インデクスではない場合に加算します。
　＋Σ（インデクス格納RDエリア数×2）
　＋インデクスの使用中セグメント数
　＋Σ｛（インデクス格納RDエリア数×7）＋（インデクス構成部分構造パス数×3）※2｝
LOB列を定義している場合に加算します。
＋（LOB列数×4）×表格納RDエリア数＋1
最適化情報を収集している場合に加算します。
＋2＋最適化情報収集列数×3＋最適化情報収集インデクス数×3
ビュー表の基表の場合に加算します。
＋削除対象ビュー表数＋Σ｛（12＋ビュー表の基表数）＋列数×4｝＋2
無効となるルーチンがある場合に加算します。
＋オブジェクトが無効となるルーチン数＋Σ（2＋プロシジャ内リソース数×5）
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
ユーザ定義型の列が定義されている場合に加算します。
＋ユーザ定義型列数×4＋LOB属性数×分割RDエリア数×5＋抽象データ型属性数＋4
トリガが定義されている場合に加算します。
＋トリガ定義数×8
＋UPDATEトリガ定義の総指定列数×3
＋トリガ動作条件があるトリガ数×3
＋トリガ動作条件中で参照する資源種類数×5
LOB列又はLOB属性を含むユーザ定義型を定義している場合に加算します。
＋Σ（↑RDエリア確保セグメント数÷64000↑＋1）※3
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
＋制約削除によって影響する被参照表の関数の総数
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋｛表格納用RDエリア数＋インデクス格納RDエリア数
＋（LOB列数＋LOB属性数）×表格納用RDエリア数｝×62＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋表を基に定義したビュー表数＋削除する表又はビュー表を参照するルーチン数
参照制約を定義している場合に加算します。

＋被参照表数×4
＋制約削除によって影響する被参照表の関数の総数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

マトリクス分割表ではない場合に加算します。
21＋列数×4
＋オブジェクトが無効となるルーチン数
　一時表ではない場合にさらに加算します。
　＋表格納RDエリア数×3
　＋分割条件指定数×9
マトリクス分割表の場合に加算します。
19＋表格納RDエリア数×3＋分割条件指定数×9＋列数×4＋オブジェクトが無効となるルーチン数
インデクスを定義している場合に加算します。
＋Σ（構成列数×3）＋3
　一時表ではない場合にさらに加算します。
　＋Σ｛（インデクス格納RDエリア数×7）＋（インデクス構成部分構造パス数×3※2）｝
LOB列を定義している場合に加算します。
＋（LOB列数×4）×表格納RDエリア数＋1
最適化情報を収集している場合に加算します。
＋2＋最適化情報収集列数×3＋最適化情報収集インデクス数×3
ビュー表の基表の場合に加算します。
＋削除対象ビュー表数＋Σ｛12＋ビュー表の基表数）＋列数×4｝＋2
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数＋Σ（2＋プロシジャ内リソース数×5）
削除する表を参照するトリガ動作手続きがある場合に加算します。
＋削除する表を参照するトリガ動作手続き数×5
ユーザ定義型の列が定義されている場合に加算します。
＋ユーザ定義型列数×4＋LOB属性数×分割RDエリア数×5＋抽象データ型属性数＋4
トリガが定義されている場合に加算します。
＋トリガ定義数×8
＋UPDATEトリガ定義の総指定列数×3
＋トリガ動作条件があるトリガ数×3
＋トリガ動作条件中で参照する資源種類数×5
参照制約を定義している場合に加算します。
＋被参照表数
＋外部キー数×10
　参照制約動作にCASCADEが定義されている場合にさらに加算します。
　＋CASCADE数×17
　＋CASCADEで指定した被参照数×4
　参照制約動作の指定がUPDATE ON CASCADEの場合にさらに加算します。
　＋CASCADEを指定した被参照表の主キー構成列の合計数×8
　＋CASCADEで指定した被参照数×3
検査制約を定義している場合に加算します。
＋検査制約数×9
再編成時期予測機能を使用している場合に加算します。
＋｛表格納用RDエリア数＋インデクス格納RDエリア数
＋（LOB列数＋LOB属性数）×表格納用RDエリア数｝×62＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ※1）で一時表ではないとき

5＋表格納RDエリア数×4＋表データの使用中セグメント数＋オブジェクトが無効となるルーチン数
インデクスを定義している場合に加算します。
＋Σ（インデクス格納RDエリア数×2）＋インデクスの使用中セグメント数

LOB列又はLOB属性を含むユーザ定義型を定義している場合に加算します。
＋Σ（↑RDエリア確保セグメント数÷64000↑＋1）※3

(e)　HiRDB/パラレルサーバの場合（バックエンドサーバ）で一時表のとき

オブジェクト又はインデクスが無効となるルーチン数

(f)　HiRDB/パラレルサーバの場合（参照専用バックエンドサーバ※1）

2
共用インデクスを定義している場合に加算します。
＋インデクス格納RDエリア数

注※1　インナレプリカ機能を使用する場合は世代数分の排他要求数が必要になります。

注※2　インデクスが部分構造インデクスの場合に繰り返し加算します。

注※3　LOB 格納 RD エリアの数だけ繰り返し加算します。

(31)　DROP TRIGGER

(a)　HiRDB/シングルサーバの場合

19＋オブジェクトが無効になるルーチン数＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5
無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3＋オブジェクトが無効になるルーチン数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

16＋オブジェクトが無効になるルーチン数＋トリガ動作手続きで使用している資源数×5
トリガ動作条件指定がある場合に加算します。
＋トリガ動作条件指定中で使用している資源数×5＋3
トリガ動作手続き中に新旧値相関名で修飾した列名を使用している場合に加算します。
＋新値相関名で修飾した列種類数×3＋旧値相関名で修飾した列種類数×3＋3
トリガ契機列を指定する場合に加算します。
＋トリガ契機列数×3
無効になるルーチンがある場合に加算します。
＋無効になるルーチン数×5＋ルーチンが使用する資源数×5

無効になるルーチンにトリガ動作手続きがある場合に加算します。
＋トリガ動作手続き数×5

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効になるルーチン数

(32)　DROP 〔PUBLIC〕VIEW

(a)　HiRDB/シングルサーバの場合

10＋削除されるビュー表数＋無効となるプロシジャ数＋Σ（15＋列数×4＋ビューの基表数×4）
無効となるプロシジャがある場合に加算します。
＋無効となるプロシジャ数＋Σ（2＋プロシジャ内リソース数×5）
削除するビュー表を参照するトリガ動作手続きがある場合に加算します。
＋削除するビュー表を参照するトリガ動作手続き数×5
パブリックビュー表の場合に加算します。
＋1
＋パブリックビュー表と同じ名称の表数
＋パブリックビュー表と同じ名称のビュー表数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

削除されるビュー表数＋無効となるプロシジャ数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

10＋Σ（15＋列数×4＋ビューの基表数×4）
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数＋Σ（2＋ルーチン内リソース数×5）
削除するビュー表を参照するトリガ動作手続きがある場合に加算します。
＋削除するビュー表を参照するトリガ動作手続き数×5
パブリックビュー表の場合に加算します。
＋1
＋パブリックビュー表と同じ名称の表数
＋パブリックビュー表と同じ名称のビュー表数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

オブジェクトが無効となるルーチン数

(33)　GRANT AUDIT

(a)　HiRDB/シングルサーバの場合

3
パスワード文字制限機能を使用している場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3
パスワード文字制限機能を使用している場合に加算します。
＋2

(34)　GRANT CONNECT

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(35)　GRANT DBA

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数
パスワード文字制限機能を使用している場合に加算します。
＋2

(36)　GRANT RDAREA

(a)　HiRDB/シングルサーバの場合

5＋指定したユーザ数
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5＋指定したユーザ数
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(37)　GRANT SCHEMA

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数

(38)　GRANT SCHEMA OPERATION

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数

(39)　GRANT アクセス権限

(a)　HiRDB/シングルサーバの場合

4＋指定した認可識別子数
指定した表が監査証跡表の場合に加算します。
＋指定したユーザ数＋１

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

4＋指定した認可識別子数
指定した表が監査証跡表の場合に加算します。
＋指定したユーザ数＋１

(40)　GRANT ロール利用権限

(a)　HiRDB/シングルサーバの場合

6＋（指定したロール数）×（指定した認可識別子数）×2

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6＋（指定したロール数）×（指定した認可識別子数）×2

(41)　REVOKE CONNECT

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数
IPアドレスによる接続制限機能を使用している場合に加算します。
＋2＋ディクショナリ表SQL_ACCESS_SECURITYの行数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数
IPアドレスによる接続制限機能を使用している場合に加算します。
＋2＋ディクショナリ表SQL_ACCESS_SECURITYの行数

(42)　REVOKE DBA

(a)　HiRDB/シングルサーバの場合

3＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

3＋指定したユーザ数

(43)　REVOKE RDAREA

(a)　HiRDB/シングルサーバの場合

10
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

10
PUBLIC指定の場合に加算します。
＋指定したRDエリア数
ユーザ指定の場合に加算します。
＋指定したRDエリア数×指定したユーザ数

(44)　REVOKE SCHEMA

(a)　HiRDB/シングルサーバの場合

5＋指定したユーザ数
削除するスキーマ操作権限がある場合に加算します。
＋1＋削除対象のスキーマ操作権限を持っているユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

5＋指定したユーザ数
削除するスキーマ操作権限がある場合に加算します。
＋1＋削除対象のスキーマ操作権限を持っているユーザ数

(45)　REVOKE SCHEMA OPERATION

(a)　HiRDB/シングルサーバの場合

2＋指定したユーザ数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

2＋指定したユーザ数

(46)　REVOKE アクセス権限

(a)　HiRDB/シングルサーバの場合

7＋指定した認可識別子数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×2＋1
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのシングルサーバの排他資源数
指定した表が監査証跡表の場合に加算します。
＋指定したユーザ数＋ 1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのフロントエンドサーバの排他資源数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6＋指定した認可識別子数
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数×2＋1
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのディクショナリサーバの排他資源数
指定した表が監査証跡表の場合に加算します。
＋指定したユーザ数＋１

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

0
無効となるルーチンがある場合に加算します。
＋無効となるルーチン数
表を基に定義したビュー表がある場合に加算します。
＋DROP 〔PUBLIC 〕VIEWのバックエンドサーバの排他資源数

(47)　REVOKE ロール利用権限

(a)　HiRDB/シングルサーバの場合

4＋指定したロール数×指定した認可識別子数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

4＋指定したロール数×指定した認可識別子数

付録 D.2　操作系 SQL

(1)　SELECT（一時表以外で LOCK TABLE なし及び WITHOUT LOCK 指定なしの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋検索対象表格納RDエリア数

＋表数
＋検索ヒット行数※3
＋利用インデクス格納RDエリア数※2
＋検索で利用したインデクスキー数※5
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
リストを介した表データ検索の場合，次の点に留意してください。
・式中の前処理の排他要求数は，リストの基表に適用して求めてください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象表格納RDエリア数＋利用インデクス格納RDエリア数
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象表格納RDエリア数
＋表数
＋検索ヒット行数※3
＋利用インデクス格納RDエリア数※2
＋検索で利用したインデクスキー数※5
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
リストを介した表データ検索の場合，次の点に留意してください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象表格納RDエリア数＋利用インデクス格納RDエリア数
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

検索条件式中の，AND，OR で指定した列にインデクスが定義してある場合は，基本的にそのインデクスは使用されます。

注※3

- 複数の表から検索する場合は，個々の表の検索ヒット行数を求めて加算してください。
- インデクス定義列に対する条件を二つ以上（FOR UPDATE がある場合は一つ以上）指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。
- ページ排他指定の場合は検索ヒットページ数となります。

注※4

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※5

インデクスキー値排他指定の場合は 1，インデクスキー値無排他指定の場合 0 になります。

(2) SELECT（一時表以外で LOCK TABLE なし及び WITHOUT LOCK WAIT 指定ありの場合）

(a) HiRDB/シングルサーバの場合

［前処理］※1
＋検索対象RDエリア数
＋表数
＋1
＋利用インデクス格納RDエリア数※2
＋検索で利用したインデクスキー数※4
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。
・式中の前処理の排他要求数は，リストの基表に適用して求めてください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象RDエリア数＋利用インデクス格納RDエリア数
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b) HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象RDエリア数
＋表数
＋2
＋利用インデクス格納RDエリア数※2
＋検索で利用したインデクスキー数※4
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象RDエリア数＋利用インデクス格納RDエリア数
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

検索条件式中の AND，OR で指定した列にインデクスが定義してある場合は，基本的にそのインデクスは使用されます。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

インデクスキー値排他指定の場合は 1，インデクスキー値無排他指定の場合 0 になります。

(3)　SELECT（一時表以外で LOCK TABLE なし及び WITHOUT LOCK NOWAIT 指定ありの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋検索対象RDエリア数
＋表数
＋利用インデクス格納RDエリア数※2
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。
・式中の前処理の排他要求数は，リストの基表に適用して求めてください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1

pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象RDエリア数＋利用インデクス格納RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※1
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象RDエリア数＋表数＋利用インデクス格納RDエリア数※2
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象RDエリア数＋利用インデクス格納RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

検索条件式中の AND，OR で指定した列にインデクスが定義してある場合は，基本的にそのインデクスは使用されます。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(4)　SELECT（一時表の場合，又は一時表以外で LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2＋検索対象RDエリア数＋利用インデクス格納RDエリア数
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。
・式中の前処理の排他要求数は，リストの基表に適用して求めてください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象RDエリア数＋利用インデクス格納RDエリア数
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※2
リストを介した表データ検索の場合，式中の前処理の排他要求数はリストの基表に適用して求めてください。
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

検索対象RDエリア数＋利用インデクス格納RDエリア数
インデクス型プラグインを使用して検索する場合，又は参照する列にデータ型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
リストを介した表データ検索の場合，次の点に留意してください。
・式中の検索対象表格納RDエリア数や表数は，リストの基表について求めてください。
・次の値を加算してください。
＋1
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋検索対象RDエリア数＋利用インデクス格納RDエリア数
ルーチンを使用して検索する場合に加算します。
＋使用するルーチン数＋1

注※1

FOR UPDATE 指定の SELECT で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(5) INSERT（一時表以外で INSERT～VALUES 句指定，LOCK TABLE なしの場合）

(a) HiRDB/シングルサーバの場合

［前処理］※1
＋5
＋挿入対象RDエリア数※3
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数
＋1ページに格納できないBINARY型の挿入列数×分岐行数
＋インデクス数
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※2
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

(b) HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c) HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d) HiRDB/パラレルサーバの場合（バックエンドサーバ）

5
＋挿入対象RDエリア数※3
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数
＋1ページに格納できないBINARY型の挿入列数×分岐行数
＋インデクス数
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※2
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※3

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(6) INSERT（一時表の場合，又は INSERT～VALUES 句指定及び LOCK TABLE ありの場合※1）

(a) HiRDB/シングルサーバの場合

［前処理］※2＋1＋挿入対象RDエリア数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋5
＋2×一時インデクス数
＋3×シングルサーバ内の一時表用RDエリア数
＋一時表と一時インデクスで使用するセグメント数の総和

(b) HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2

(c) HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1＋挿入対象RDエリア数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋5
＋2×一時インデクス数
＋3×バックエンドサーバ内の一時表用RDエリア数
＋一時表と一時インデクスで使用するセグメント数の総和

注※1

一時表以外で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(7)　INSERT（INSERT～SELECT 節指定ありの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋挿入対象RDエリア数※5
＋4＋挿入行数※2，※6
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入行の列数×挿入行数※2，※6×分岐行数
＋1ページに格納できないBINARY型の挿入列数×挿入行数※2，※6×分岐行数
＋挿入行数※6×インデクス数
＋1
＋検索対象RDエリア数※5
＋検索対象表数＋検索ヒット行数※3，※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※4
＋検索に利用するプラグインが使用する論理ファイル数※4
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数＋検索対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。

＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋2＋2×一時インデクス数
＋シングルサーバ内の一時表用RDエリア数×3
＋一時表と一時インデクスで使用するセグメント数の総和

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

4＋挿入対象RDエリア数※5
＋挿入行数※2，※6
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入行の列数×挿入行数※6×分岐行数
＋1ページに格納できないBINARY型の挿入列数×挿入行数※6×分岐行数
＋挿入行数※6×インデクス数
＋1
＋検索対象RDエリア数※5
＋検索対象表数
＋検索ヒット行数※3，※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※4
＋検索に利用するプラグインが使用する論理ファイル数※4
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数＋検索対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋2＋2×一時インデクス数
＋バックエンドサーバ内の一時表用RDエリア数×3
＋一時表と一時インデクスで使用するセグメント数の総和

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

ページ排他指定の場合は挿入ページ数となります。

注※3

- 複数の表から検索する場合は，個々の表の検索ヒット行数を求めて加算してください。
- インデクス定義列に対する条件を二つ以上指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。
- ページ排他指定の場合は検索ヒットページ数となります。

注※4

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※5

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※6

一時表の場合は 0 です。

(8) INSERT（一時表以外で INSERT～VALUES 句指定，LOCK TABLE なし，及びインデクスキー値無排他方式の場合）

(a) HiRDB/シングルサーバの場合

```
［前処理］※1
＋5
＋挿入対象RDエリア数※2
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数
＋1ページに格納できないBINARY型の挿入列数×分岐行数
＋1
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
```

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

5＋挿入対象RDエリア数※2
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×分岐行数
＋1ページに格納できないBINARY型の挿入列数×分岐行数
＋1
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

(9)　INSERT（一時表の場合，又は INSERT～VALUES 句指定，LOCK TABLE あり※4，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋1＋挿入対象RDエリア数※2
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数

一時表に挿入する場合に加算します。
＋2
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋5
＋2×一時インデクス数
＋3×シングルサーバ内の一時表用RDエリア数
＋一時表と一時インデクスで使用するセグメント数の総和

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

1＋挿入対象RDエリア数※2
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋5
＋2×一時インデクス数
＋3×バックエンドサーバ内の一時表用RDエリア数
＋一時表と一時インデクスで使用するセグメント数の総和

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

一時表以外で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

(10) INSERT（INSERT～SELECT 節指定，及びインデクスキー値無排他方式の場合）

(a) HiRDB/シングルサーバの場合

［前処理］※1
＋挿入対象RDエリア数※2
＋4
＋挿入行数※3，※6
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×挿入行数※6×分岐行数
＋1ページに格納できないBINARY型の挿入列数×挿入行数※6×分岐行数
＋インデクス数
＋1
＋検索対象RDエリア数※2
＋検索対象表数＋検索ヒット行数※4，※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※5
＋検索に利用するプラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数＋検索対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋2
＋2×一時インデクス数
＋3×サーバ内一時表用RDエリア数
＋一時表と一時インデクスで使用するセグメント数の総和

(b) HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
SQLセッション固有一時表に挿入する場合に加算します。
＋使用するSQLセッション固有一時表数
一時表に挿入する場合に加算します。
＋2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

挿入対象RDエリア数※2
＋4
＋挿入行数※3，※6
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の挿入列数×挿入行数※6×分岐行数
＋1ページに格納できないBINARY型の挿入列数×挿入行数※6×分岐行数
＋インデクス数
＋1
＋検索対象RDエリア数※2
＋検索対象表数＋検索ヒット行数※4，※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋挿入対象プラグインが使用する論理ファイル数※5
＋検索に利用するプラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋挿入対象RDエリア数＋検索対象RDエリア数
ルーチンを使用して挿入する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して挿入する場合に加算します。
＋使用する順序数生成子数×2
一時表に挿入する場合に加算します。
＋2
＋2×一時インデクス数
＋3×バックエンドサーバ内の一時表用RDエリア数
＋一時表と一時インデクスで使用するセグメント数の総和

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※3

ページ排他指定の場合は挿入ページ数となります。

注※4

- 複数の表から検索する場合は，個々の表の検索ヒット行数を求めて加算してください。
- インデクス定義列に対する条件を二つ以上指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。
- ページ排他指定の場合は検索ヒットページ数となります。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※6

一時表の場合は 0 です。

(11)　UPDATE（一時表以外で LOCK TABLE なしの場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋更新対象RDエリア数※5
＋4
＋更新行数※2
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数
＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数
＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数
＋更新対象インデクス数×2×更新行数
＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※5
＋4
＋更新行数※2
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数

＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数
＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数
＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数
＋更新対象インデクス数×2×更新行数
＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて更新行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を更新行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を更新行数にしてください。
- ページ排他指定の場合は更新ページ数となります。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列及び LOB 属性の数だけ繰り返し加算します。

注※5

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(12)　UPDATE（一時表の場合，又は LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2＋更新対象RDエリア数※5＋1＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数

pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2
一時表を使用して更新する場合に加算します。
＋一時表と一時インデクスで使用するセグメント数の総和

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※5＋1＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2
一時表を使用して更新する場合に加算します。
＋一時表と一時インデクスで使用するセグメント数の総和

注※1

一時表以外で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列及び LOB 属性の数だけ繰り返し加算します。

注※5

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

(13)　UPDATE（一時表以外で LOCK TABLE なし，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋更新対象RDエリア数※2
＋4
＋更新行数※3
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数
＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数
＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数
＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2
pd_unique_check_modeオペランドに1を指定し，表にプライマリ指定又はユニーク指定のインデクスを定義している場合，次の値を加算します。
＋Σ（↑更新前インデクスキー長÷6↑）※6×更新前行数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※2
＋4
＋更新行数※3
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新前列数×更新前行数×更新前分岐行数
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の更新後列数×更新後行数×更新後分岐行数

＋1ページに格納できないBINARY型の更新前列数×更新前行数×更新前分岐行数
＋1ページに格納できないBINARY型の更新後列数×更新後行数×更新後分岐行数
＋Σ（更新前LOBデータ格納セグメント数）※4
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2
pd_unique_check_modeオペランドに1を指定し，表にプライマリ指定又はユニーク指定のインデクスを定義している場合，次の値を加算します。
＋Σ（↑更新前インデクスキー長÷6↑）※6×更新前行数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて更新行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を更新行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を更新行数にしてください。
- ページ排他指定の場合は更新ページ数となります。表にユニークインデクスが定義されている場合は，更新する行数分を更に加算してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※6

対象となるインデクスの数だけ繰り返し加算します。

(14)　UPDATE（一時表の場合，又は LOCK TABLE あり※5，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1＋更新対象RDエリア数※2＋1＋Σ（更新前LOBデータ格納セグメント数）※3
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2
一時表を使用して更新する場合に加算します。
＋一時表と一時インデクスで使用するセグメント数の総和

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

更新対象RDエリア数※2＋1＋Σ（更新前LOBデータ格納セグメント数）※3
更新対象列にデータ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※4
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋更新対象RDエリア数
pd_nowait_scan_optionオペランドにLOCKを指定した場合に加算します（非FIX表のときだけ加算します）。
＋1
ルーチンを使用して更新する場合に加算します。
＋使用するルーチン数＋1
順序数生成子（NEXT VALUE式）を使用して更新する場合に加算します。
＋使用する順序数生成子数×2
一時表を使用して更新する場合に加算します。
＋一時表と一時インデクスで使用するセグメント数の総和

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※4

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※5

一時表以外で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

(15)　DELETE（一時表以外で LOCK TABLE なしの場合）

(a)　HiRDB/シングルサーバの場合

```
［前処理］※1
＋削除対象RDエリア数※5
＋1
＋削除行数※2
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数
＋1ページに格納できないBINARY型の削除列数×分岐数×削除行数
＋インデクス数×削除行数
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1
```

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

```
［前処理］※1
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1
```

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

```
［前処理］※1
```

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

```
削除対象RDエリア数※5
＋1
＋削除行数※2
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数
＋1ページに格納できないBINARY型の削除列数×分岐数×削除行数
```

＋インデクス数×削除行数
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて削除行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を削除行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を削除行数にしてください。
- ページ排他指定の場合は削除ページ数となります。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

(16)　DELETE（一時表の場合，又は LOCK TABLE ありの場合※1）

(a)　HiRDB/シングルサーバの場合

［前処理］※2
＋削除対象RDエリア数※5
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
＋解放データページ数※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※2
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※2

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※5
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
＋解放データページ数※6
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※3
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

一時表以外で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

注※2

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※3

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

対象となるインデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※6

ページ内の全データが削除されるページ数です。

(17)　DELETE（一時表以外で LOCK TABLE なし，及びインデクスキー値無排他方式の場合）

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋削除対象RDエリア数※2
＋1
＋削除行数※3
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数
＋1ページに格納できないBINARY型の削除列数×分岐数×削除行数
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1
pd_unique_check_modeオペランドに1を指定し，表にプライマリ指定又はユニーク指定のインデクスを定義している場合，次の値を加算します。
＋Σ（↑インデクスキー長÷6↑）※6×削除行数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※2
＋1
＋削除行数※3
＋VARCHAR，NVARCHAR，MVARCHARの256バイト以上の削除列数×分岐数×削除行数
＋1ページに格納できないBINARY型の列数×分岐行数
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1
pd_unique_check_modeオペランドに1を指定し，表にプライマリ指定又はユニーク指定のインデクスを定義している場合，次の値を加算します。
＋Σ（↑インデクスキー長÷6↑）※6×削除行数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となる表，インデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

- 別の表からの検索が必要になる条件式がある場合は，その表の検索ヒット行数を求めて削除行数に加算してください。
- インデクス定義列に対する条件を指定している場合は，各インデクスに対する条件だけで検索される行の総和を求めてください。また，検索ヒット行数を削除行数にしてください。
- 複数の表の列を使用した条件や，関数を用いた条件を除いて検索ヒット行数を求めてください。また，検索ヒット行数を削除行数にしてください。
- ページ排他指定の場合は削除ページ数となります。表にユニークインデクスが定義されている場合は，削除する行数分を更に加算してください。

注※4

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※6

対象となるインデクスの数だけ繰り返し加算します。

(18) DELETE（一時表の場合，又は LOCK TABLE あり※6，及びインデクスキー値無排他方式の場合）

(a) HiRDB/シングルサーバの場合

［前処理］※1
＋削除対象RDエリア数※2
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※3
＋解放データページ数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

削除対象RDエリア数※2
＋Σ（削除したLOBデータ数＋削除したLOBデータ格納セグメント数）※3
＋解放データページ数※4
データ型プラグイン又はインデクス型プラグインが定義されている場合に加算します。
＋プラグインが使用する論理ファイル数※5
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋削除対象RDエリア数
ルーチンを使用して削除する場合に加算します。
＋使用するルーチン数＋1

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

対象となるインデクス，LOB 列，及び LOB 属性格納 RD エリアの総数です。

注※3

LOB 列と LOB 属性の数だけ繰り返し加算します。

注※4

ページ内の全データが削除されるページ数です。

注※5

プラグインが使用する論理ファイル数については，プラグインに付いているマニュアルを参照してください。

注※6

一時表以外で LOCK TABLE IN SHARE MODE がある場合，LOCK TABLE なしと同じ計算式となります。

(19)　PURGE TABLE

(a)　HiRDB/シングルサーバの場合

［前処理］※1
＋対象RDエリア数※3
＋3

＋表の使用中セグメント数×2
＋Σ（（インデクスの使用中セグメント数×2）＋2）※2
＋Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋対象RDエリア数
pd_check_pendingオペランドにUSEを指定，又は省略した場合，かつ対象表が被参照表の場合にさらに加算します。
＋Σ（参照表の対象RDエリア数※3）※5
pd_check_pendingオペランドにUSEを指定，又は省略した場合に加算します。
＋4
対象表が被参照表の場合にさらに加算します。
＋3
＋対象表の表格納用RDエリア数
＋対象表を参照する参照表数×2
＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
＋Σ（参照表の対象RDエリア数※3＋2）※5
対象表が分割表の場合にさらに加算します。
＋1＋対象表の表格納用RDエリア数
対象表を参照する参照表が分割表の場合にさらに加算します。
＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
対象表が参照表，又は検査制約を定義した表，かつ対象表の表格納用RDエリアにインナレプリカ機能を適用しているRDエリアを含まない場合にさらに加算します。
＋1＋対象表に定義した参照制約の数
＋1＋対象表に定義した検査制約の数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

［前処理］※1

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

［前処理］※1
pd_check_pendingオペランドにUSEを指定，又は省略した場合に加算します。
＋4
対象表が被参照表の場合にさらに加算します。
＋3
＋対象表の表格納用RDエリア数
＋対象表を参照する参照表数×2
＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
対象表が分割表の場合にさらに加算します。
＋1＋対象表の表格納用RDエリア数
対象表を参照する参照表が分割表の場合にさらに加算します。
＋Σ（1＋対象表を参照する参照表の表格納用RDエリア数）※5
対象表が参照表，又は検査制約を定義した表，かつ対象表の表格納用RDエリアにインナレプリカ機能を適用しているRDエリアを含まない場合にさらに加算します。
＋1＋対象表に定義した参照制約の数
＋1＋対象表に定義した検査制約の数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

対象RDエリア数※3
＋3
＋表の使用中セグメント数×2

＋Σ（（インデクスの使用中セグメント数×2）＋2）※2
＋Σ（↑HiRDBファイルにLOBデータを格納しているセグメント数÷64000↑）※4
pd_inner_replica_controlオペランドを指定した場合に加算します。
＋対象RDエリア数
pd_check_pendingオペランドにUSEを指定，又は省略した場合，かつ対象表が被参照表の場合にさらに加算します。
＋Σ（参照表の対象RDエリア数※3）※5
pd_check_pendingオペランドにUSEを指定，又は省略した場合，かつ対象表が被参照表の場合に加算します。
＋Σ（参照表の対象RDエリア数※3＋2）※5

注※1

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

注※2

インデクスの数だけ繰り返し加算します。

注※3

対象となる表，インデクス，LOB 列，及び LOB 属性格納用 RD エリアの総数です。

注※4

LOB 列と LOB 属性の LOB 用 RD エリアの HiRDB ファイル数だけ繰り返し加算します。

注※5

対象表を参照する参照表数だけ繰り返し加算します。

(20)　ASSIGN LIST 文（実表からのリスト作成）

(a)　HiRDB/シングルサーバの場合

基表SELECT排他要求数※1＋7＋2
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数＋3＋再作成前のリストデータの使用中セグメント数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

前処理※2

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

前処理※2＋7
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

基表SELECT排他要求数※1＋2
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋3＋再作成前のリストデータの使用中セグメント数

注※1

基表 SELECT 排他要求数については，「SELECT（一時表以外で LOCK TABLE なし及び WITHOUT LOCK 指定なしの場合）」，「SELECT（一時表以外で LOCK TABLE なし及び WITHOUT LOCK WAIT 指定ありの場合）」，「SELECT（一時表以外で LOCK TABLE なし及び WITHOUT LOCK NOWAIT 指定ありの場合）」，又は「SELECT（一時表の場合，又は一時表以外で LOCK TABLE ありの場合※1）」のうち，基表の条件に合う SELECT 文を参照してください。

注※2

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

(21)　ASSIGN LIST 文（リストからのリスト作成）

(a)　HiRDB/シングルサーバの場合

前処理※＋7＋リストの基表格納RDエリア数＋4
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数＋3＋再作成前のリストデータの使用中セグメント数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

前処理※

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

前処理※＋7
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

リストの基表格納RDエリア数＋4
作成するリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋3＋再作成前のリストデータの使用中セグメント数

注※

前処理の排他要求数については，「前処理の排他要求数」を参照してください。

(22)　ASSIGN LIST 文（リストの名前の変更）

(a)　HiRDB/シングルサーバの場合

7
変更後のリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7
変更後のリストの名前として，既にあるリストを指定した場合には次の値を加算します。
＋再作成前のリストの基表格納RDエリア数

(23)　DROP LIST 文（リストの名前を指定した削除：DROP LIST～）

(a)　HiRDB/シングルサーバの場合

7＋リストの基表格納RDエリア数＋3＋リストデータの使用中セグメント数

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

7＋リストの基表格納RDエリア数

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

3＋リストデータの使用中セグメント数

(24)　DROP LIST 文（自分の所有するすべてのリストの削除：DROP ALL LIST～）

(a)　HiRDB/シングルサーバの場合

1＋リスト用RDエリア数＋削除リスト数×2＋削除リストデータの使用中セグメント数の合計

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

1

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

リスト用RDエリア数＋削除リスト数×2＋削除リストデータの使用中セグメント数の合計

(25)　前処理の排他要求数

(a)　HiRDB/シングルサーバの場合

1
＋使用する表数
＋使用するビュー表数
＋Σ※1（2＋列数×2＋インデクス数×2＋表分割数×4
＋Σ※2（インデクス分割数×2）＋2）
＋13
ユーザ定義型，又は関数を使用する場合に加算します。
＋1
最適化情報を収集している場合に加算します（１表当たり）。
＋1＋最適化情報作成列数×4＋1
ビュー表を使用する場合に加算します（１ビュー表当たり）。
＋Σ（列数×2＋4＋ビュー表を構成する列数＋2）
LOB列を定義している場合に加算します。
＋LOB列数×2＋表分割数
ユーザ定義型の列を定義している場合に加算します。
ユーザ定義型列数＋Σ（4＋属性数×2）
LOB属性がある場合に加算します。
＋LOB属性数×2＋表分割数
上位型がある場合に加算します。
＋2
関数を使用している場合に，使用関数分加算します。
＋Σ（2＋名称及びパラメタ構成の同じ関数数×2＋パラメタ数×2）
プラグインを使用する場合に，プラグイン関数数分加算します。
＋Σ（2＋パラメタ数×2）＋3＋コンテキスト数×4＋2
ルーチンを使用する場合に加算します。
＋使用するルーチン数＋1
順序数生成子を使用する場合に加算します。
＋使用する順序数生成子数＋1＋使用する順序数生成子数

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

1＋使用する表数＋使用するビュー表数
ユーザ定義型，又は関数を使用する場合に加算します。
＋1
ルーチンを使用する場合に加算します。
＋使用するルーチン数＋1
順序数生成子を使用する場合に加算します。
＋使用する順序数生成子数

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

Σ※1（2＋列数×2＋インデクス数×2＋表分割数×4＋Σ※2（インデクス分割数×2）＋2）＋13
最適化情報を収集している場合に加算します（1表当たり）。
＋1＋最適化情報作成列数×4＋1
ビュー表を使用する場合に加算します（1ビュー表当たり）。
＋Σ（列数×2＋4＋ビュー表を構成する列数＋2）
LOB列を定義している場合に加算します。
＋LOB列数×2＋表分割数
ユーザ定義型の列を定義している場合に，ユーザ定義型列数分加算します。
＋Σ（4＋属性数×2）

LOB属性がある場合に加算します。
＋LOB属性数×2＋表分割数
上位型がある場合に加算します
＋2
関数を使用している場合に，使用関数分加算します。
＋Σ（2＋名称及びパラメタ構成の同じ関数数×2＋パラメタ数×2）
プラグインを使用する場合に，プラグイン関数数分加算します。
＋Σ（2＋パラメタ数×2）＋3＋コンテキスト数×4＋2
順序数生成子を使用する場合に加算します。
＋1＋使用する順序数生成子数

注※1　表数分加算します。

注※2　表内分割インデクス数分加算します。

付録 D.3　制御系 SQL

(1)　LOCK

(a)　HiRDB/シングルサーバの場合

［前処理］※＋表横分割数＋1

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ及びディクショナリサーバ）

［前処理］※

(c)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

表横分割数＋1

注※

前処理時の排他資源数については，「前処理の排他要求数」を参照してください。

(2)　SET ROLE

(a)　HiRDB/シングルサーバの場合

6

(b)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

6

付録 D.4　ユティリティ及びコマンド

(1)　データベース作成ユティリティ（pdload）

(a)　HiRDB/シングルサーバの場合

209＋α＋表格納用RDエリアの使用セグメント数＋インデクス格納RDエリアの使用セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

206＋α

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

表格納用RDエリアの使用セグメント数＋インデクス格納RDエリアの使用セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）

α：MAX（表の列数，表のインデクス数，表の格納 RD エリア数）

使用セグメント数はデータ格納予定セグメント数で求めます。なお，-d オプション指定時に，既にデータが格納されているセグメント数（使用中セグメント数）の方がデータ格納予定セグメント数より多い場合は，使用中セグメント数で計算してください。

(2)　データベース再編成ユティリティ（pdrorg）

ディクショナリ表のメンテナンスを実行する場合は(8)を参照してください。

(a)　HiRDB/シングルサーバの場合

209＋α＋表格納用RDエリアの使用中セグメント数＋インデクス格納RDエリアの使用中セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

3

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

206＋α

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

表格納用RDエリアの使用中セグメント数＋インデクス格納RDエリアの使用中セグメント数＋（↑LOB用RDエリアの使用セグメント数÷64000↑）

α：MAX（表の列数，表のインデクス数，表の格納 RD エリア数）

(3)　リバランスユティリティ（pdrbal）

(a)　HiRDB/シングルサーバの場合で，共有モード（-k share）のとき

開始終了処理※1＋前処理※2＋リバランス処理※3＋2

(b)　HiRDB/シングルサーバの場合で，占有モード（-k exclusive）のとき

開始終了処理※1＋前処理※2＋リバランス処理※3

(c)　HiRDB/パラレルサーバの場合で，共有モード（-k share）のとき

- フロントエンドサーバ

開始終了処理※1＋前処理※2＋1

- ディクショナリサーバ

開始終了処理※1＋前処理※2

- バックエンドサーバ

開始終了処理※1＋リバランス処理※3

(d)　HiRDB/パラレルサーバの場合で，占有モード（-k exclusive）のとき

- フロントエンドサーバ

開始終了処理※1＋前処理※2

- ディクショナリサーバ

開始終了処理※1＋前処理※2

- バックエンドサーバ

開始終了処理※1＋リバランス処理※3

注※1

開始終了処理の排他要求数＝ΣAi ＋ ΣBi

Ai：1 関数の排他要求数

関数の排他要求数については，「CREATE 〔PUBLIC〕FUNCTION」を参照してください。

Bi：1 手続きの排他要求数

手続きの排他要求数については，「CREATE 〔PUBLIC〕PROCEDURE」を参照してください。

注※2

前処理の排他要求数は次のとおりです。

- HiRDB/シングルサーバの場合：225
- HiRDB/パラレルサーバの場合（フロントエンドサーバ）：5
- HiRDB/パラレルサーバの場合（ディクショナリサーバ）：220

注※3

リバランス処理の排他要求数＝表格納用 RD エリアの使用中セグメント数＋インデクス格納用 RD エリアの使用中セグメント数＋（↑ユーザ LOB 用 RD エリアの使用セグメント数÷64000↑）

(4) pdacunlck コマンド

(a) HiRDB/シングルサーバの場合

```
0
認可識別子指定の場合に加算します。
＋128
ALL指定の場合に加算します。
＋ディクショナリ表SQL_USERSに登録されているユーザ数
```

(b) HiRDB/パラレルサーバの場合（ディクショナリサーバ）

```
0
認可識別子指定の場合に加算します。
＋128
ALL指定の場合に加算します。
＋ディクショナリ表SQL_USERSに登録されているユーザ数
```

(5) 整合性チェックユティリティ（pdconstck）

(a) HiRDB/シングルサーバの場合

```
↑{209＋A＋f＋(a＋c＋(d×b))}÷x↑
```

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

↑(2＋f)÷x↑

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

↑{209＋A＋(a＋c＋(d×b))}÷x↑

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

↑(d×b)÷x↑

a：表格納用 RD エリア数

b：インデクス格納用 RD エリア数

c：LOB 列格納用 RD エリア数

d：インデクス数

f：処理対象表及び外部キーが参照する主キーの表に定義されたプラグイン数

x：32 ビットモードの HiRDB の場合は 6，64 ビットモードの HiRDB の場合は 4

A：MAX（表の列数，表のインデクス数，表格納用 RD エリア数，表の外部キー数，表の検査制約数）

(6)　データベース構成変更ユティリティ（pdmod）

(a)　HiRDB/シングルサーバの場合

(i)RD エリアの再初期化の場合

20
格納表ありの場合に加算します。
＋格納表数＋インデクス格納用RDエリア数※1＋LOB列格納用RDエリア数
ほかのRDエリアに参照関係にある参照表ありの場合に加算します。
＋参照表数×2＋参照表格納RDエリア数
HiRDBファイルの構成変更ありの場合に加算します。
＋HiRDBファイル数※2

(ii)RD エリアの削除の場合

20＋HiRDBファイル数
削除するRDエリアが解析情報表，及び運用履歴表を格納するデータディクショナリ用RDエリアの場合に加算します。
＋100

(iii)HiRDB/シングルサーバ構成から HiRDB/パラレルサーバ構成への RD エリアの変更の場合

20＋RDエリア定義数＋HiRDBファイル数＋ルーチン数
分割表ありの場合に加算します。
＋分割表格納RDエリア数＋分割条件数

(iv)ディクショナリ表の属性定義変更の場合

20＋ディクショナリ表数×3
dicinfオペランド指定の場合に加算します。
＋120＋ルーチン数×2＋トリガ定義数×2＋トリガ動作条件の指定中で使用している資源数＋トリガ動作手続き中に新旧値相関名で修飾した列名数
mcharオペランド指定の場合に加算します。
＋600

(v)RD エリアの属性変更の場合

20＋HiRDBファイル数
renameオペランド指定の場合に加算します(LOB用RDエリア以外)。
＋格納表数＋LOB列数＋インデクス数＋分割表数×2＋分割インデクス数＋順序数生成子数

(vi)その他の場合

40

(b)　HiRDB/パラレルサーバの場合

(i)RD エリアの再初期化の場合

- ディクショナリサーバ

10
格納表ありの場合に加算します。
＋インデクス格納用RDエリア数※1＋LOB列格納用RDエリア数
ほかのRDエリアに参照関係にある参照表ありの場合に加算します。
＋参照表数×2＋参照表格納RDエリア数
HiRDBファイルの構成変更ありの場合に加算します。
＋HiRDBファイル数※2

- バックエンドサーバ

10※3
格納表ありの場合に加算します。
＋格納表数

(ii)RD エリアの削除の場合

- ディクショナリサーバ

10＋HiRDBファイル数
削除するRDエリアが解析情報表，及び運用履歴表を格納するデータディクショナリ用RDエリアの場合に

加算します。
＋100

- バックエンドサーバ

10※3

(iii)ディクショナリ表の属性定義変更の場合

- ディクショナリサーバ

20＋ディクショナリ表数＋ディクショナリビュー表数×2
dicinfオペランド指定の場合に加算します。
＋120＋ルーチン数×2＋トリガ定義数×2＋トリガ動作条件の指定中で使用している資源数＋トリガ動作手続き中に新旧値相関名で修飾した列名数
mcharオペランド指定の場合に加算します。
＋600

(iv)RD エリアの属性変更の場合

- ディクショナリサーバ

10＋HiRDBファイル数
renameオペランド指定の場合に加算します(LOB用RDエリア以外)。
＋格納表数＋LOB列数＋インデクス数＋分割表数×2＋分割インデクス数＋順序数生成子数

- バックエンドサーバ

10※3

(v)RD エリアの移動の場合

- ディクショナリサーバ

10＋RDエリアの指定数※4＋指定したRDエリアのHiRDBファイル数
格納表を参照するルーチンがある場合に加算します。
＋ルーチン数×2
格納表を参照するトリガ動作手続きがある場合に加算します。
＋トリガ定義数＋トリガ動作条件の指定中で使用している資源数＋トリガ動作手続き中に新旧値相関名で修飾した列名数
norestartオペランド指定の場合に加算します。
＋格納表数
分割表ありの場合に加算します。
＋分割表の分割数×2
＋分割表に定義したインデクス数

- バックエンドサーバ

10×RDエリアの指定数※5

(vi)その他の場合

- ディクショナリサーバ

```
30
```

- バックエンドサーバ

```
10※3
```

注※1

ほかの RD エリアにインデクス定義がある場合に加算します。

注※2

with reconstructions オペランド指定の場合は×2 になります。

注※3

共用 RD エリアの場合はすべてのバックエンドサーバで加算します。

注※4

norestart オペランド指定の場合は×2 になります。

注※5

非共用 RD エリアの場合は移動元バックエンドサーバと移動先バックエンドサーバで加算します。共用 RD エリアの場合はすべてのバックエンドサーバで加算します。

(7) pdchpathn コマンド

(a) HiRDB/シングルサーバの場合

```
6
```

(b) HiRDB/パラレルサーバの場合

- ディクショナリサーバ

```
6
```

(8) ディクショナリ表のメンテナンス（pdrorg -k maintenance -c dic）

(a) HiRDB/シングルサーバの場合

```
39＋SQL_RDAREAS表の行数
```

(b)　HiRDB/パラレルサーバの場合

- ディクショナリサーバ

```
39＋SQL_RDAREAS表の行数
```

(9)　JP1/NETM/Audit 用監査ログ出力ユティリティ（pdaudput）

(a)　HiRDB/シングルサーバの場合

```
MAX{(537),(102＋監査証跡表のインデクス数×2＋監査証跡表の利用インデクス格納RDエリア数＋監査証跡表の検索で利用したインデクスキー数)}
```

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

```
3
```

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

```
MAX{(531),(97＋監査証跡表のインデクス数×2)}
```

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

```
3＋監査証跡表の利用インデクス格納RDエリア数＋監査証跡表の検索で利用したインデクスキー数※1
```

注※1

-w オプションで指定した検索条件で使用する全インデクスキー数の合計値のことを指します。

付録 D.5　HiRDB 起動時に必要となる排他資源数

(1)　HiRDB の起動

HiRDB 再開始時にディレードリランによる回復が発生する場合，回復のためのトランザクション実行に必要とする排他資源数は見積もり式に含みません。その場合の排他資源数は，この項で示す排他資源数と同時実行するトランザクションの排他要求数の総和を合計した数になります。

(a)　HiRDB/シングルサーバの場合

```
12
pd_max_tmp_table_rdarea_noオペランドに1以上を指定した場合に加算します。
+一時表用RDエリアの最大数（pd_max_tmp_table_rdarea_noの値）
```

(b)　HiRDB/パラレルサーバの場合（フロントエンドサーバ）

```
0
```

(c)　HiRDB/パラレルサーバの場合（ディクショナリサーバ）

```
12
```

(d)　HiRDB/パラレルサーバの場合（バックエンドサーバ）

```
0
```

付録 E　pdconfchk コマンドでチェックできるオペランド

pdconfchk コマンドでチェックできるオペランドを次の表に示します。オペランド名はアルファベット順に記載しています。なお，次に示す定義のオペランドについては pdconfchk コマンドでチェックしません。

- UAP 環境定義

表 E–1　pdconfchk コマンドでチェックできるオペランド

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
HORCMINST	○	×	×	×	×	○
pd_additional_optimize_level	○	×	×	×	×	○
pd_alv_port	○	×	×	×	×	○
pd_assurance_index_no	○	×	×	×	×	○
pd_assurance_table_no	○	×	×	×	×	○
pd_audit	○	×	×	×	×	○
pd_aud_async_buff_count	○	×	×	×	×	○
pd_aud_async_buff_retry_intvl	○	×	×	×	×	×
pd_aud_async_buff_size	○	×	×	×	×	○
pd_aud_auto_loading	○	×	×	×	×	○
pd_aud_file_name	○	×	×	×	×	○
pd_aud_file_wrn_pnt	○	×	×	×	×	○
pd_aud_max_generation_num	○	×	×	×	×	○
pd_aud_max_generation_size	○	×	×	×	×	○
pd_aud_no_standby_file_opr	○	×	×	×	×	○
pd_aud_sql_data_size	○	×	×	×	×	×
pd_aud_sql_source_size	○	×	×	×	×	×
pd_audit_def_buffer_size	○	×	×	×	×	×
pd_auth_cache_size	○	×	×	×	×	○
pd_auto_vrup	○	×	×	×	×	○
pd_bes_connection_hold	○	×	×	×	×	×
pd_bes_conn_hold_trn_interval	○	×	×	×	×	×
pd_bes_shmpool_size	○	×	×	×	×	○
pd_c_library_directory	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_cancel_down_msgchange	○	×	×	×	×	○
pd_cancel_dump	○	×	×	×	×	○
pd_change_clt_ipaddr	○	×	×	×	×	○
pd_check_pending	○	×	×	×	×	×
pd_client_waittime_over_abort	○	×	×	×	×	○
pd_clt_waittime_over_dump_level	○	×	×	×	×	○
pd_cmd_exec_time	○	×	×	×	×	○
pd_cmd_process_conwaittime	○	×	×	×	×	○
pd_cmdhold_precheck	○	×	×	×	×	○
pd_command_deadlock_priority	○	×	×	×	×	○
pd_connect_auth_type	○	×	×	×	×	○
pd_connect_errmsg_hide	○	×	×	×	×	○
pd_constraint_name	○	×	×	×	×	×
pd_crypto_certificate_file	○	×	×	×	×	○
pd_crypto_private_key	○	×	×	×	×	○
pd_crypto_service_port	○	×	×	×	×	○
pd_cwaittime_report_dir	○	×	×	×	×	×
pd_cwaittime_report_size	○	×	×	×	×	×
pd_cwaittime_wrn_pnt	○	×	×	×	×	×
pd_dbbuff_attribute	○	×	×	×	×	○
pd_dbbuff_binary_data_lru	○	×	×	×	×	○
pd_dbbuff_dev_sector_size	○	×	×	×	×	○
pd_dbbuff_lru_option	○	×	×	×	×	○
pd_dbbuff_modify	○	×	×	×	×	○
pd_dbbuff_rate_updpage	○	×	×	×	×	○
pd_dbbuff_read_retry_count	○	×	×	×	×	○
pd_dbbuff_read_retry_intvl	○	×	×	×	×	○
pd_dbbuff_trace_level	○	×	×	×	×	○
pd_dbreuse_remaining_entries	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_dbsync_altwrite_skip	○	×	×	×	×	○
pd_dbsync_point	○	×	×	×	×	○
pd_db_access_error_action	○	×	×	×	×	○
pd_db_hold_action	○	×	×	×	×	○
pd_db_io_error_action	○	×	×	×	×	×
pd_db_timestamp_invalid_action	○	×	×	×	×	○
pd_ddl_idx_pctfree	○	×	×	×	×	○
pd_ddl_tbl_fix_pctfree	○	×	×	×	×	○
pd_ddl_tbl_pctfree	○	×	×	×	×	○
pd_ddl_tbl_recovery	○	×	×	×	×	○
pd_deadlock_priority_use	○	×	×	×	×	○
pd_debug_info_netstat	○	×	×	×	×	○
pd_dec_sign_normalize	○	×	×	×	×	○
pd_def_buf_control_area_assign	×	×	×	×	×	×
pd_delete_reserved_word_file	○	×	×	×	×	○
pd_dfw_awt_process	○	×	×	×	×	○
pd_dfw_bulk_write	○	×	×	×	×	○
pd_dfw_syncpoint_skip_limit	○	×	×	×	×	×
pd_dic_shmpool_size	○	×	×	×	×	○
pd_down_watch_proc	○	×	×	×	×	○
pd_dump_suppress_watch_time	○	×	×	×	×	○
pd_duplicated_fs_suffix	○	×	×	×	×	○
pd_fes_lck_pool_partition	○	×	×	×	×	○
pd_fes_lck_pool_size	○	×	×	×	×	○
pd_floatable_bes	○	×	×	×	×	×
pd_ha	○	×	×	×	×	○
pd_ha_acttype	○	×	×	×	×	×
pd_ha_agent	○	×	×	×	×	×
pd_ha_ipaddr_inherit	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_ha_max_act_guest_servers	○	×	×	×	×	○
pd_ha_max_server_process	○	×	×	×	×	○
pd_ha_mgr_rerun	○	×	×	×	×	○
pd_ha_observe_timeout	○	×	×	×	×	○
pd_ha_prc_cleanup_check	○	×	×	×	×	○
pd_ha_process_count	○	×	×	×	×	○
pd_ha_resource_act_wait_time	○	×	×	×	×	○
pd_ha_restart_failure	○	×	×	×	×	×
pd_ha_server_process_standby	○	×	×	×	×	×
pd_ha_switch_error	○	×	×	×	×	○
pd_ha_switch_timeout	○	×	×	×	×	○
pd_ha_transaction	○	×	×	×	×	○
pd_ha_trn_queuing_initial	○	×	×	×	×	○
pd_ha_trn_queuing_wait_time	○	×	×	×	×	○
pd_ha_trn_restart_retry_time	○	×	×	×	×	○
pd_ha_unit	○	×	×	×	×	×
pd_hash_table_size	○	×	×	×	×	○
pd_hostname	○	×	×	×	×	○
pd_host_watch_interval	○	×	×	×	×	○
pd_idx_without_rollback	○	×	×	×	×	○
pd_indexlock_mode	○	×	×	×	×	○
pd_inner_replica_control	○	×	×	×	×	×
pd_inner_replica_hold_status	○	×	×	×	×	○
pd_inner_replica_lock_shift	○	×	×	×	×	×
pd_ipc_clt_conn_nblock	○	×	×	×	×	○
pd_ipc_clt_conn_nblock_time	○	×	×	×	×	○
pd_ipc_conn_count	○	×	×	×	×	○
pd_ipc_conn_interval	○	×	×	×	×	○
pd_ipc_conn_nblock	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_ipc_conn_nblock_time	○	×	×	×	×	○
pd_ipc_file_dir	○	×	×	×	×	○
pd_ipc_recv_count	○	×	×	×	×	○
pd_ipc_send_count	○	×	×	×	×	○
pd_ipc_send_retrycount	○	×	×	×	×	○
pd_ipc_send_retrysleeptime	○	×	×	×	×	○
pd_ipc_so_keepalive	○	×	×	×	×	○
pd_ipc_tcp_keepcnt	○	×	×	×	×	○
pd_ipc_tcp_keepidle	○	×	×	×	×	○
pd_ipc_tcp_keepintvl	○	×	×	×	×	○
pd_isolation_level	○	×	×	×	×	○
pd_java_archive_directory	○	×	×	×	×	○
pd_java_castoff	○	×	×	×	×	○
pd_java_classpath	○	×	×	×	×	○
pd_java_libpath	○	×	×	×	×	○
pd_java_option	○	×	×	×	×	○
pd_java_routine_stack_size	○	×	×	×	×	○
pd_java_runtimepath	○	×	×	×	×	○
pd_java_stdout_file	○	×	×	×	×	○
pd_key_resource_type	○	×	×	×	×	○
pd_lck_deadlock_check	○	×	×	×	×	○
pd_lck_deadlock_check_interval	○	×	×	×	×	○
pd_lck_deadlock_check_timeout	○	×	×	×	×	○
pd_lck_deadlock_check_pause	○	×	×	×	×	○
pd_lck_deadlock_info	○	×	×	×	×	○
pd_lck_deadlock_sql_info	○	×	×	×	×	○
pd_lck_hash_entry	○	×	×	×	×	○
pd_lck_pool_partition	○	×	×	×	×	○
pd_lck_pool_size	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_lck_queue_limit	○	×	×	×	×	○
pd_lck_until_disconnect_cnt	○	×	×	×	×	○
pd_lck_wait_timeout	○	×	×	×	×	○
pd_list_initialize_timing	○	×	×	×	×	×
pd_list_rowcount_in_rename	○	×	×	×	×	○
pd_lock_uncommited_delete_data	○	×	×	×	×	○
pd_log_auto_expand_size	○	×	×	×	×	○
pd_log_auto_unload_path	○	×	×	×	×	○
pd_log_auto_unload_restart	○	×	×	×	×	○
pd_log_block_check	○	×	×	×	×	○
pd_log_dual	○	×	×	×	×	○
pd_log_dual_write_method	○	×	×	×	×	×
pd_log_fg_warning_point	○	×	×	×	×	○
pd_log_org_no_standby_file_opr	○	×	×	×	×	×
pd_log_org_reflected_logpoint	○	×	×	×	×	×
pd_log_remain_space_check	○	×	×	×	×	×
pd_log_rerun_reserved_file_open	○	×	×	×	×	○
pd_log_rerun_swap	○	×	×	×	×	○
pd_log_rpl_no_standby_file_opr	○	×	×	×	×	○
pd_log_sdinterval	○	×	×	×	×	○
pd_log_singleoperation	○	×	×	×	×	○
pd_log_swap_timeout	○	×	×	×	×	○
pd_log_unload_check	○	×	×	×	×	○
pd_log_write_buff_count	○	×	×	×	×	○
pd_log_full_write_async	○	×	×	×	×	○
pd_lv_mirror_use	○	×	×	×	×	○
pd_master_file_name	○	×	×	×	×	○
pd_max_access_tables	○	×	×	×	×	○
pd_max_access_tables_wrn_pnt	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_max_add_dbbuff_no	○	×	×	×	×	○
pd_max_add_dbbuff_shm_no	○	×	×	×	×	○
pd_max_ard_process	○	×	×	×	×	×
pd_max_bes_process	○	×	×	×	×	○
pd_max_dic_process	○	×	×	×	×	○
pd_max_file_no	○	×	×	×	×	○
pd_max_file_no_wrn_pnt	○	×	×	×	×	○
pd_max_list_users	○	×	×	×	×	○
pd_max_list_count	○	×	×	×	×	○
pd_max_list_users_wrn_pnt	○	×	×	×	×	○
pd_max_list_count_wrn_pnt	○	×	×	×	×	○
pd_max_open_fds	○	×	×	×	×	○
pd_max_open_holdable_cursors	○	×	×	×	×	○
pd_max_rdarea_no	○	×	×	×	×	○
pd_max_rdarea_no_wrn_pnt	○	×	×	×	×	○
pd_max_recover_process	○	×	×	×	×	○
pd_max_reflect_process_count	○	×	×	×	×	○
pd_max_resident_rdarea_no	○	×	×	×	×	○
pd_max_resident_rdarea_shm_no	○	×	×	×	×	○
pd_max_server_process	○	×	×	×	×	○
pd_max_temporary_object_no	○	×	×	×	×	○
pd_max_users	○	×	×	×	×	○
pd_max_users_wrn_pnt	○	×	×	×	×	○
pd_max_utl_ios_file_no	○	×	×	×	×	○
pd_mlg_file_size	○	×	×	×	×	○
pd_mlg_msg_log_unit	○	×	×	×	×	○
pd_mlg_port	○	×	×	×	×	○
pd_mode_conf	○	×	×	×	×	○
pd_module_trace_max	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_module_trace_timer_level	○	×	×	×	×	○
pd_name_fixed_port_lookup	○	×	×	×	×	○
pd_name_port	○	×	×	×	×	○
pd_non_floatable_bes	○	×	×	×	×	×
pd_nowait_scan_option	○	×	×	×	×	○
pd_oltp_holdcr	○	×	×	×	×	○
pd_optimize_level	○	×	×	×	×	○
pd_os_authenticate	○	×	×	×	×	○
pd_overflow_suppress	○	×	×	×	×	○
pd_pageaccess_mode	○	×	×	×	×	○
pd_plugin_ixmk_dir	○	×	×	×	×	×
pd_prf_file_count	○	×	×	×	×	○
pd_prf_file_size	○	×	×	×	×	○
pd_prf_level	○	×	×	×	×	○
pd_prf_output_directory	○	×	×	×	×	○
pd_prf_trace	○	×	×	×	×	○
pd_process_count	○	×	×	×	×	○
pd_process_terminator	○	×	×	×	×	○
pd_process_terminator_max	○	×	×	×	×	○
pd_pth_trace_max	○	×	×	×	×	○
pd_queue_watch_time	○	×	×	×	×	○
pd_queue_watch_timeover_action	○	×	×	×	×	○
pd_rdarea_expand_format	○	×	×	×	×	○
pd_rdarea_extension_file	○	×	×	×	×	○
pd_rdarea_extension_timing	○	×	×	×	×	○
pd_rdarea_list_no_wrn_pnt	○	×	×	×	×	○
pd_rdarea_open_attribute	○	×	×	×	×	○
pd_rdarea_open_attribute_use	○	×	×	×	×	○
pd_rdarea_warning_point	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_rdarea_warning_point_msgout	○	×	×	×	×	○
pd_read_replica_max_trn_count	○	×	×	×	×	○
pd_read_replica_mode	○	×	×	×	×	○
pd_read_replica_use	○	×	×	×	×	○
pd_read_replica_workdir	○	×	×	×	×	○
pd_redo_allpage_put	○	×	×	×	×	○
pd_reduced_check_time	○	×	×	×	×	○
pd_registered_port	○	○	○	○	○	○
pd_registered_port_check	○	○	○	○	○	○
pd_registry_cache_size	○	×	×	×	×	○
pd_rise_disaster_mode	○	×	×	×	×	○
pd_rise_fence_level	○	×	×	×	×	○
pd_rise_pairvolume_combination	○	×	×	×	×	○
pd_rise_use	○	×	×	×	×	○
pd_rorg_predict	○	×	×	×	×	○
pd_routine_def_cache_size	○	×	×	×	×	○
pd_rpc_trace	○	×	×	×	×	○
pd_rpc_trace_name	○	×	×	×	×	○
pd_rpc_trace_size	○	×	×	×	×	○
pd_rpl_func_control	○	×	×	×	×	○
pd_rpl_hdepath	○	×	×	×	×	×
pd_rpl_init_start	○	×	×	×	×	○
pd_rpl_reflect_mode	○	×	×	×	×	○
pd_scd_port	○	×	×	×	×	○
pd_sds_shmpool_size	○	×	×	×	×	○
pd_security_host_group	○	×	×	×	○	○
pd_security_local_constraint	○	×	×	×	×	○
pd_server_cleanup_interval	○	×	×	×	×	○
pd_server_process_adjust_intvl	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_server_process_adjust_limit	○	×	×	×	×	○
pd_service_port	○	×	×	×	×	○
pd_shared_rdarea_use	○	×	×	×	×	○
pd_shmpool_attribute	○	×	×	×	×	○
pd_shmpool_control	○	×	×	×	×	○
pd_space_level	○	×	×	×	×	○
pd_spd_assurance_count	○	×	×	×	×	○
pd_spd_assurance_msg	○	×	×	×	×	○
pd_spd_dual	○	×	×	×	×	○
pd_spd_max_data_size	○	×	×	×	×	○
pd_spd_reduced_mode	○	×	×	×	×	○
pd_spd_reserved_file_auto_open	○	×	×	×	×	○
pd_spd_syncpoint_skip_limit	○	×	×	×	×	○
pd_spool_cleanup	○	×	×	×	×	×
pd_spool_cleanup_interval	○	×	×	×	×	×
pd_spool_cleanup_interval_level	○	×	×	×	×	×
pd_spool_cleanup_level	○	×	×	×	×	×
pd_sql_command_exec_users	○	×	×	×	×	○
pd_sql_dec_op_maxprec	○	×	×	×	×	○
pd_sql_mode	○	×	×	×	×	○
pd_sql_object_cache_size	○	×	×	×	×	○
pd_sqlobject_stat_timing	○	×	×	×	×	○
pd_start_level	○	×	×	×	×	○
pd_start_skip_unit	○	×	×	×	×	○
pd_start_time_out	○	×	×	×	×	○
pd_statistics	○	×	×	×	×	○
pd_stj_file_size	○	×	×	×	×	○
pd_stj_buff_size	○	×	×	×	×	○
pd_standard_sqlstate	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_stored_lob_data_wrn_pnt	○	×	×	×	×	○
pd_sts_file_name_1～7	○	○	○	○	×	×
pd_sts_last_active_file	○	×	×	×	×	×
pd_sts_last_active_side	○	×	×	×	×	×
pd_sts_last_active_side_sub	○	×	×	×	×	×
pd_sts_last_active_subfile	○	×	×	×	×	×
pd_sts_subfile_name_1～7	○	○	○	○	×	×
pd_substr_length	○	×	×	×	×	○
pd_svr_castoff_size	○	×	×	×	×	○
pd_syssts_file_name_1～7	○	×	×	×	×	○
pd_syssts_last_active_file	○	×	×	×	×	○
pd_syssts_last_active_side	○	×	×	×	×	○
pd_syssts_last_active_side_sub	○	×	×	×	×	×
pd_syssts_last_active_subfile	○	×	×	×	×	×
pd_syssts_subfile_name_1～7	○	○	○	○	×	×
pd_system_complete_wait_time	○	×	×	×	×	○
pd_system_dbsync_point	○	×	×	×	×	○
pd_system_expand_unit	○	×	×	×	×	○
pd_system_id	○	×	×	×	×	○
pd_table_def_cache_size	○	×	×	×	×	○
pd_term_watch_count	○	×	×	×	×	○
pd_thdlock_wakeup_lock	○	×	×	×	×	○
pd_thread_max_stack_size	○	×	×	×	×	○
pd_thread_stack_expand_size	○	×	×	×	×	○
pd_tmp_directory	○	○	○	×	×	×
pd_tmp_table_initialize_timing	○	×	×	×	×	○
pd_trn_commit_optimize	○	×	×	×	×	○
pd_trn_port	○	×	×	×	×	○
pd_trn_rcvmsg_store_buflen	○	×	×	×	×	○

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pd_trn_rerun_branch_auto_decide	○	×	×	×	×	○
pd_trn_rollback_watch_time	○	×	×	×	×	○
pd_trn_send_decision_interval	○	×	×	×	×	○
pd_trn_send_decision_intval_sec	○	×	×	×	×	○
pd_trn_send_decision_retry_time	○	×	×	×	×	○
pd_trn_watch_time	○	×	×	×	×	○
pd_type_def_cache_size	○	×	×	×	×	○
pd_uap_exerror_log_dir	○	×	×	×	×	○
pd_uap_exerror_log_param_size	○	×	×	×	×	○
pd_uap_exerror_log_size	○	×	×	×	×	○
pd_uap_exerror_log_use	○	×	×	×	×	○
pd_unique_check_mode	○	×	×	×	×	○
pd_unique_indexlock_info	○	×	×	×	×	○
pd_unit_id	○	×	×	×	×	○
pd_utl_exec_time	○	×	×	×	×	○
pd_view_def_cache_size	○	×	×	×	×	○
pd_watch_pc_client_time	○	×	×	×	×	○
pd_watch_resource	○	×	×	×	×	○
pd_work_buff_expand_limit	○	×	×	×	×	○
pd_work_buff_mode	○	×	×	×	×	○
pd_work_buff_size	○	×	×	×	×	○
pd_work_table_option	○	×	×	×	×	○
pdaudload	○	×	×	×	×	○
pdbuffer	○	×	×	×	×	○
pdcltgrp	○	×	×	×	×	○
PDCONFPATH	○	×	×	×	×	○
pdhagroup	○	×	×	○	×	○
pdhibegin	○	×	×	×	×	○
pdlogadfg -d rrp	○	×	×	×	×	×

オペランド名	文法のチェック	ファイルのチェック	アクセス権限のチェック	重複指定のチェック	ホスト名のチェック	サーバマシン間のチェック
pdlogadfg -d spd	○	×	×	×	×	×
pdlogadfg -d ssp	○	×	×	×	×	×
pdlogadfg -d sys	○	×	×	×	×	×
pdlogadpf -d rrp	○	○	○	○	×	×
pdlogadpf -d spd	○	○	○	○	×	×
pdlogadpf -d ssp	○	○	○	○	×	×
pdlogadpf -d sys	○	○	○	○	×	×
pdmlgput	○	×	×	×	×	○
pdplgprm	○	×	×	×	×	×
pdplugin	○	×	×	×	×	○
pdstart	○	×	×	×	○	○
pdstbegin	○	×	×	×	×	○
pdunit	○	×	×	×	○	○
pdwork	○	×	×	×	×	×
pdwork_wrn_pnt	○	×	×	×	×	○

（凡例）

○：チェック対象です。

×：チェック対象外です。

文法のチェック：

オペランドの文法が正しいかチェックします。

ファイルのチェック：

システムログファイル，シンクポイントダンプファイル，及びステータスファイルの有無をチェックします。pdconfchk コマンドで-n オプションを指定した場合はファイルのチェックをしません。

アクセス権限のチェック：

HiRDB 管理者にファイルのアクセス権があるかを確認します。

HiRDB 管理者に hosts ファイルのアクセス権があるかを確認します。

pdconfchk コマンドで-n オプションを指定した場合はアクセス権限のチェックをしません。

pd_tmp_directory オペランドで指定したディレクトリにアクセス権があるかを確認します。
pd_tmp_directory オペランドと TMPDIR 環境変数の指定を共に省略した場合は，アクセス権限のチェックをしません。

重複指定のチェック：

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

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

ホスト名のチェック：

ホスト名が hosts ファイルに記述されているかをチェックします。

サーバマシン間のチェック（HiRDB/パラレルサーバ限定）：

システムマネジャのサーバマシンを基準に，サーバマシン間の妥当性をチェックします。

付録 F　適用 OS ごとのオペランド指定可否

適用 OS ごとのオペランド指定可否を次の表に示します。オペランド名はアルファベット順に記載しています。

表 F–1　適用 OS ごとのオペランド指定可否

オペランド名	AIX	Linux
HORCMINST	○	△
pd_additional_optimize_level	○	○
pd_alv_port	○	○
pd_assurance_index_no	○	○
pd_assurance_table_no	○	○
pd_audit	○	○
pd_aud_async_buff_count	○	○
pd_aud_async_buff_retry_intvl	○	○
pd_aud_async_buff_size	○	○
pd_aud_auto_loading	○	○
pd_aud_file_name	○	○
pd_aud_file_wrn_pnt	○	○
pd_aud_max_generation_num	○	○
pd_aud_max_generation_size	○	○
pd_aud_no_standby_file_opr	○	○
pd_aud_sql_data_size	○	○
pd_aud_sql_source_size	○	○
pd_audit_def_buffer_size	○	○
pd_auth_cache_size	○	○
pd_auto_vrup	○	○
pd_bes_connection_hold	○	○
pd_bes_conn_hold_trn_interval	○	○
pd_bes_shmpool_size	○	○
pd_c_library_directory	○	○
pd_cancel_down_msgchange	○	○
pd_cancel_dump	○	○

オペランド名	AIX	Linux
pd_change_clt_ipaddr	○	○
pd_check_pending	○	○
pd_client_waittime_over_abort	○	○
pd_clt_waittime_over_dump_level	○	○
pd_cmd_exec_time	○	○
pd_cmd_process_conwaittime	○	○
pd_cmd_rconfig	○	○
pd_cmd_rlog	○	○
pd_cmd_rmode	○	○
pd_cmdhold_precheck	○	○
pd_command_deadlock_priority	○	○
pd_connect_auth_type	○	○
pd_connect_errmsg_hide	○	○
pd_constraint_name	○	○
pd_crypto_certificate_file	○	○
pd_crypto_private_key	○	○
pd_crypto_service_port	○	○
pd_cwaittime_report_dir	○	○
pd_cwaittime_report_size	○	○
pd_cwaittime_wrn_pnt	○	○
pd_dbbuff_attribute	△	○
pd_dbbuff_binary_data_lru	○	○
pd_dbbuff_dev_sector_size	×	○
pd_dbbuff_lru_option	○	○
pd_dbbuff_modify	○	○
pd_dbbuff_rate_updpage	○	○
pd_dbbuff_read_retry_count	○	○
pd_dbbuff_read_retry_intvl	○	○
pd_dbbuff_trace_level	○	○
pd_dbreuse_remaining_entries	○	○
pd_dbsync_altwrite_skip	○	○

オペランド名	AIX	Linux
pd_dbsync_point	○	○
pd_db_access_error_action	○	○
pd_db_hold_action	○	○
pd_db_io_error_action	○	○
pd_db_timestamp_invalid_action	○	○
pd_ddl_idx_pctfree	○	○
pd_ddl_tbl_fix_pctfree	○	○
pd_ddl_tbl_pctfree	○	○
pd_ddl_tbl_recovery	○	○
pd_deadlock_priority_use	○	○
pd_debug_info_netstat	○	○
pd_dec_sign_normalize	○	○
pd_def_buf_control_area_assign	○	○
pd_delete_reserved_word_file	○	○
pd_dfw_awt_process	○	○
pd_dfw_bulk_write	○	○
pd_dfw_syncpoint_skip_limit	○	○
pd_dic_shmpool_size	○	○
pd_down_watch_proc	○	○
pd_dump_suppress_watch_time	○	○
pd_duplicated_fs_suffix	×	○
pd_fes_lck_pool_partition	○	○
pd_fes_lck_pool_size	○	○
pd_floatable_bes	○	○
pd_ha	○	○
pd_ha_acttype	○	○
pd_ha_agent	○	○
pd_ha_ipaddr_inherit	○	○
pd_ha_max_act_guest_servers	○	○
pd_ha_max_server_process	○	○
pd_ha_mgr_rerun	○	○

オペランド名	AIX	Linux
pd_ha_observe_timeout	○	○
pd_ha_prc_cleanup_check	○	○
pd_ha_process_count	○	○
pd_ha_resource_act_wait_time	○	○
pd_ha_restart_failure	○	○
pd_ha_server_process_standby	○	○
pd_ha_switch_error	○	○
pd_ha_switch_timeout	○	○
pd_ha_transaction	○	○
pd_ha_trn_queuing_initial	○	○
pd_ha_trn_queuing_wait_time	○	○
pd_ha_trn_restart_retry_time	○	○
pd_ha_unit	○	○
pd_hash_table_size	○	○
pd_hostname	○	○
pd_host_watch_interval	○	○
pd_idx_without_rollback	○	○
pd_indexlock_mode	○	○
pd_inner_replica_control	○	○
pd_inner_replica_hold_status	○	○
pd_inner_replica_lock_shift	○	○
pd_ipc_clt_conn_nblock	○	○
pd_ipc_clt_conn_nblock_time	○	○
pd_ipc_conn_count	○	○
pd_ipc_conn_interval	○	○
pd_ipc_conn_nblock	○	○
pd_ipc_conn_nblock_time	○	○
pd_ipc_file_dir	○	○
pd_ipc_recv_count	○	○
pd_ipc_send_count	○	○
pd_ipc_send_retrycount	○	○

オペランド名	AIX	Linux
pd_ipc_send_retrysleeptime	○	○
pd_ipc_so_keepalive	○	○
pd_ipc_tcp_keepcnt	○	○
pd_ipc_tcp_keepidle	○	○
pd_ipc_tcp_keepintvl	○	○
pd_isolation_level	○	○
pd_java_archive_directory	△	○
pd_java_castoff	○	○
pd_java_classpath	△	○
pd_java_libpath	△	○
pd_java_option	△	○
pd_java_routine_stack_size	△	○
pd_java_runtimepath	△	○
pd_java_stdout_file	△	○
pd_key_resource_type	○	○
pd_lck_deadlock_check	○	○
pd_lck_deadlock_check_interval	○	○
pd_lck_deadlock_check_timeout	○	○
pd_lck_deadlock_check_pause	○	○
pd_lck_deadlock_info	○	○
pd_lck_deadlock_sql_info	○	○
pd_lck_hash_entry	○	○
pd_lck_pool_partition	○	○
pd_lck_pool_size	○	○
pd_lck_queue_limit	○	○
pd_lck_until_disconnect_cnt	○	○
pd_lck_wait_timeout	○	○
pd_list_initialize_timing	○	○
pd_list_rowcount_in_rename	○	○
pd_lock_uncommited_delete_data	○	○
pd_log_auto_expand_size	○	○

オペランド名	AIX	Linux
pd_log_auto_unload_path	○	○
pd_log_auto_unload_restart	○	○
pd_log_block_check	×	○
pd_log_dual	○	○
pd_log_dual_write_method	○	○
pd_log_fg_warning_point	○	○
pd_log_full_write_async	○	○
pd_log_org_no_standby_file_opr	○	○
pd_log_org_reflected_logpoint	○	○
pd_log_remain_space_check	○	○
pd_log_rerun_reserved_file_open	○	○
pd_log_rerun_swap	○	○
pd_log_rpl_no_standby_file_opr	○	○
pd_log_sdinterval	○	○
pd_log_singleoperation	○	○
pd_log_swap_timeout	○	○
pd_log_unload_check	○	○
pd_log_write_buff_count	○	○
pd_lv_mirror_use	○	○
pd_master_file_name	○	○
pd_max_access_tables	○	○
pd_max_access_tables_wrn_pnt	○	○
pd_max_add_dbbuff_no	○	○
pd_max_add_dbbuff_shm_no	○	○
pd_max_ard_process	○	○
pd_max_bes_process	○	○
pd_max_dic_process	○	○
pd_max_file_no	○	○
pd_max_file_no_wrn_pnt	○	○
pd_max_list_users	○	○
pd_max_list_count	○	○

オペランド名	AIX	Linux
pd_max_list_users_wrn_pnt	○	○
pd_max_list_count_wrn_pnt	○	○
pd_max_open_fds	○	○
pd_max_open_holdable_cursors	○	○
pd_max_rdarea_no	○	○
pd_max_rdarea_no_wrn_pnt	○	○
pd_max_recover_process	○	○
pd_max_reflect_process_count	○	○
pd_max_resident_rdarea_no	○	○
pd_max_resident_rdarea_shm_no	○	○
pd_max_server_process	○	○
pd_max_temporary_object_no	○	○
pd_max_users	○	○
pd_max_users_wrn_pnt	○	○
pd_max_utl_ios_file_no	×	○
pd_mlg_file_size	○	○
pd_mlg_msg_log_unit	○	○
pd_mlg_port	○	○
pd_mode_conf	○	○
pd_module_trace_max	○	○
pd_module_trace_timer_level	○	○
pd_name_fixed_port_lookup	○	○
pd_name_port	○	○
pd_non_floatable_bes	○	○
pd_nowait_scan_option	○	○
pd_oltp_holdcr	○	○
pd_optimize_level	○	○
pd_os_authenticate	○	○
pd_overflow_suppress	○	○
pd_pageaccess_mode	○	○
pd_plugin_ixmk_dir	○	○

オペランド名	AIX	Linux
pd_prf_file_count	○	○
pd_prf_file_size	○	○
pd_prf_level	○	○
pd_prf_output_directory	○	○
pd_prf_trace	○	○
pd_process_count	○	○
pd_process_terminator	○	○
pd_process_terminator_max	○	○
pd_pth_trace_max	○	○
pd_queue_watch_time	○	○
pd_queue_watch_timeover_action	○	○
pd_rdarea_expand_format	○	○
pd_rdarea_extension_file	○	○
pd_rdarea_extension_timing	○	○
pd_rdarea_list_no_wrn_pnt	○	○
pd_rdarea_open_attribute	○	○
pd_rdarea_open_attribute_use	○	○
pd_rdarea_warning_point	○	○
pd_rdarea_warning_point_msgout	○	○
pd_read_replica_max_trn_count	×	○
pd_read_replica_mode	×	○
pd_read_replica_use	×	○
pd_read_replica_workdir	×	○
pd_redo_allpage_put	○	○
pd_reduced_check_time	○	○
pd_registered_port	○	○
pd_registered_port_check	○	○
pd_registry_cache_size	○	○
pd_rise_disaster_mode	○	△
pd_rise_fence_level	○	△
pd_rise_pairvolume_combination	○	△

オペランド名	AIX	Linux
pd_rise_use	○	△
pd_rorg_predict	○	○
pd_routine_def_cache_size	○	○
pd_rpc_trace	○	○
pd_rpc_trace_name	○	○
pd_rpc_trace_size	○	○
pd_rpl_func_control	○	○
pd_rpl_hdepath	○	○
pd_rpl_init_start	○	○
pd_rpl_reflect_mode	○	○
pd_scd_port	○	○
pd_sds_shmpool_size	○	○
pd_security_host_group	○	○
pd_security_local_constraint	○	○
pd_server_cleanup_interval	○	○
pd_server_process_adjust_intvl	○	○
pd_server_process_adjust_limit	○	○
pd_service_port	○	○
pd_shared_rdarea_use	○	○
pd_shmpool_attribute	△	○
pd_shmpool_control	○	○
pd_space_level	○	○
pd_spd_assurance_count	○	○
pd_spd_assurance_msg	○	○
pd_spd_dual	○	○
pd_spd_max_data_size	○	○
pd_spd_reduced_mode	○	○
pd_spd_reserved_file_auto_open	○	○
pd_spd_syncpoint_skip_limit	○	○
pd_spool_cleanup	○	○
pd_spool_cleanup_interval	○	○

オペランド名	AIX	Linux
pd_spool_cleanup_interval_level	○	○
pd_spool_cleanup_level	○	○
pd_sql_command_exec_users	○	○
pd_sql_dec_op_maxprec	○	○
pd_sql_mode	○	○
pd_sql_object_cache_size	○	○
pd_sqlobject_stat_timing	○	○
pd_standard_sqlstate	○	○
pd_start_level	○	○
pd_start_skip_unit	○	○
pd_start_time_out	○	○
pd_statistics	○	○
pd_stj_file_size	○	○
pd_stj_buff_size	○	○
pd_stored_lob_data_wrn_pnt	○	○
pd_sts_file_name_1～7	○	○
pd_sts_last_active_file	○	○
pd_sts_last_active_side	○	○
pd_sts_last_active_side_sub	△	△
pd_sts_last_active_subfile	△	△
pd_sts_subfile_name_1～7	△	△
pd_substr_length	○	○
pd_svr_castoff_size	○	○
pd_syssts_file_name_1～7	○	○
pd_syssts_last_active_file	○	○
pd_syssts_last_active_side	○	○
pd_syssts_last_active_side_sub	△	△
pd_syssts_last_active_subfile	△	△
pd_syssts_subfile_name_1～7	△	△
pd_system_complete_wait_time	○	○
pd_system_dbsync_point	○	○

オペランド名	AIX	Linux
pd_system_expand_unit	×	○
pd_system_id	○	○
pd_table_def_cache_size	○	○
pd_term_watch_count	○	○
pd_thdlock_wakeup_lock	○	×
pd_thread_max_stack_size	○	○
pd_thread_stack_expand_size	○	○
pd_tmp_directory	○	○
pd_tmp_table_initialize_timing	○	○
pd_trn_commit_optimize	○	○
pd_trn_port	○	○
pd_trn_rcvmsg_store_buflen	○	○
pd_trn_rerun_branch_auto_decide	○	○
pd_trn_rollback_watch_time	○	○
pd_trn_send_decision_interval	○	○
pd_trn_send_decision_intval_sec	○	○
pd_trn_send_decision_retry_time	○	○
pd_trn_watch_time	○	○
pd_type_def_cache_size	○	○
pd_uap_exerror_log_dir	○	○
pd_uap_exerror_log_param_size	○	○
pd_uap_exerror_log_size	○	○
pd_uap_exerror_log_use	○	○
pd_uap_wait	○	○
pd_unique_check_mode	○	○
pd_unique_indexlock_info	○	○
pd_unit_id	○	○
pd_utl_exec_time	○	○
pd_view_def_cache_size	○	○
pd_watch_pc_client_time	○	○
pd_watch_resource	○	○

オペランド名	AIX	Linux
pd_work_buff_expand_limit	○	○
pd_work_buff_mode	○	○
pd_work_buff_size	○	○
pd_work_table_option	○	○
pdaudload	○	○
pdbuffer	○	○
pdcltgrp	○	○
PDCONFPATH	○	○
pdhagroup	○	○
pdhibegin	○	○
pdlbuffer	○	○
pdlogadfg -d rrp	×	○
pdlogadfg -d spd	○	○
pdlogadfg -d ssp	△	△
pdlogadfg -d sys	○	○
pdlogadpf -d rrp	×	○
pdlogadpf -d spd	○	○
pdlogadpf -d ssp	△	△
pdlogadpf -d sys	○	○
pdmlgput	○	○
pdplgprm	○	○
pdplugin	○	○
pdstart	○	○
pdstbegin	○	○
pdunit	○	○
pdwork	○	○
pdwork_wrn_pnt	○	○
SHMMAX	○	○
TZ	○	○

（凡例）

○：指定できるオペランドです。

×：指定できないオペランドです。

△：条件によって，指定できる場合と指定できない場合があります。詳しくは，各オペランドの説明を参照してください。

付録 G　リードレプリカ機能適用時の参照 DB の定義変更可否

リードレプリカ機能を適用している場合の，参照 DB での HiRDB 定義変更可否を次の表に示します。

表 G–1　参照 DB での HiRDB 定義変更可否

オペランド名	参照 DB での HiRDB 定義変更可否
HORCMINST	○
pd_additional_optimize_level	○
pd_alv_port	−
pd_apply_search_ats_num	×
pd_assurance_index_no	○
pd_assurance_table_no	○
pd_audit	○
pd_aud_async_buff_count	○
pd_aud_async_buff_retry_intvl	○
pd_aud_async_buff_size	○
pd_aud_auto_loading	○
pd_aud_file_name	○
pd_aud_file_wrn_pnt	○
pd_aud_max_generation_num	○
pd_aud_max_generation_size	○
pd_aud_no_standby_file_opr	○
pd_aud_sql_data_size	○
pd_aud_sql_source_size	○
pd_audit_def_buffer_size	○
pd_auth_cache_size	○
pd_auto_vrup	×
pd_bes_connection_hold	−
pd_bes_conn_hold_trn_interval	−
pd_bes_shmpool_size	−
pd_c_library_directory	×
pd_cancel_down_msgchange	×
pd_cancel_dump	○

オペランド名	参照 DB での HiRDB 定義変更可否
pd_change_clt_ipaddr	×
pd_check_pending	×
pd_client_waittime_over_abort	○
pd_clt_waittime_over_dump_level	−
pd_cmd_exec_time	○
pd_cmd_process_conwaittime	○
pd_cmd_rconfig	×
pd_cmd_rlog	×
pd_cmd_rmode	×
pd_cmdhold_precheck	×
pd_command_deadlock_priority	×
pd_connect_auth_type	×
pd_connect_errmsg_hide	×
pd_constraint_name	×
pd_crypto_certificate_file	×
pd_crypto_private_key	×
pd_crypto_service_port	○
pd_cwaittime_report_dir	×
pd_cwaittime_report_size	×
pd_cwaittime_wrn_pnt	○
pd_dbbuff_attribute	×
pd_dbbuff_binary_data_lru	×
pd_dbbuff_dev_sector_size	×
pd_dbbuff_lru_option	×
pd_dbbuff_modify	×
pd_dbbuff_rate_updpage	×
pd_dbbuff_read_retry_count	×
pd_dbbuff_read_retry_intvl	×
pd_dbbuff_trace_level	×
pd_dbreuse_remaining_entries	×
pd_dbsync_altwrite_skip	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_dbsync_point	×
pd_db_access_error_action	×
pd_db_hold_action	×
pd_db_io_error_action	×
pd_db_timestamp_invalid_action	×
pd_ddl_idx_pctfree	×
pd_ddl_tbl_fix_pctfree	×
pd_ddl_tbl_pctfree	×
pd_ddl_tbl_recovery	×
pd_deadlock_priority_use	×
pd_debug_info_netstat	×
pd_dec_sign_normalize	×
pd_def_buf_control_area_assign	×
pd_delete_reserved_word_file	○
pd_dfw_awt_process	×
pd_dfw_bulk_write	×
pd_dfw_syncpoint_skip_limit	×
pd_dic_shmpool_size	−
pd_down_watch_proc	×
pd_dump_suppress_watch_time	○
pd_duplicated_fs_suffix	○
pd_ekey	×
pd_fes_lck_pool_partition	−
pd_fes_lck_pool_size	−
pd_floatable_bes	−
pd_ha	○
pd_ha_acttype	○
pd_ha_agent	○
pd_ha_ipaddr_inherit	○
pd_ha_max_act_guest_servers	−
pd_ha_max_server_process	−

オペランド名	参照 DB での HiRDB 定義変更可否
pd_ha_mgr_rerun	−
pd_ha_observe_timeout	○
pd_ha_prc_cleanup_check	○
pd_ha_process_count	−
pd_ha_resource_act_wait_time	−
pd_ha_restart_failure	○
pd_ha_server_process_standby	○
pd_ha_switch_error	○
pd_ha_switch_timeout	○
pd_ha_transaction	○
pd_ha_trn_queuing_initial	−
pd_ha_trn_queuing_wait_time	○
pd_ha_trn_restart_retry_time	○
pd_ha_unit	○
pd_hash_table_size	○
pd_hashjoin_hashing_mode	×
pd_hostname	○
pd_host_watch_interval	−
pd_idx_without_rollback	×
pd_indexlock_mode	×
pd_inet_unix_bufmode	×
pd_inet_unix_utl_bufmode	×
pd_inner_replica_control	○
pd_inner_replica_hold_status	○
pd_inner_replica_lock_shift	○
pd_ipc_clt_conn_nblock	×
pd_ipc_clt_conn_nblock_time	×
pd_ipc_conn_count	×
pd_ipc_conn_interval	×
pd_ipc_conn_nblock	×
pd_ipc_conn_nblock_time	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_ipc_file_dir	×
pd_ipc_inet_bufsize	×
pd_ipc_recv_count	×
pd_ipc_send_count	×
pd_ipc_send_retrycount	×
pd_ipc_send_retrysleeptime	×
pd_ipc_so_keepalive	○
pd_ipc_tcp_keepcnt	○
pd_ipc_tcp_keepidle	○
pd_ipc_tcp_keepintvl	○
pd_ipc_unix_bufsize	×
pd_isolation_level	○
pd_java_archive_directory	×
pd_java_castoff	×
pd_java_classpath	×
pd_java_libpath	×
pd_java_option	×
pd_java_routine_stack_size	×
pd_java_runtimepath	×
pd_java_stdout_file	×
pd_key_resource_type	×
pd_lck_deadlock_check	×
pd_lck_deadlock_check_interval	×
pd_lck_deadlock_check_timeout	×
pd_lck_deadlock_check_pause	×
pd_lck_deadlock_info	×
pd_lck_deadlock_sql_info	×
pd_lck_hash_entry	×
pd_lck_pool_partition	○
pd_lck_pool_size	○
pd_lck_queue_limit	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_lck_until_disconnect_cnt	○
pd_lck_wait_timeout	○
pd_leap_second	×
pd_list_initialize_timing	×
pd_list_rowcount_in_rename	×
pd_listen_socket_bufset	×
pd_lock_uncommited_delete_data	×
pd_log_auto_expand_size	×
pd_log_auto_unload_path	×
pd_log_auto_unload_restart	×
pd_log_block_check	○
pd_log_dual	×
pd_log_dual_write_method	×
pd_log_fg_warning_point	○
pd_log_full_write_async	×
pd_log_max_data_size	×
pd_log_org_no_standby_file_opr	○
pd_log_org_reflected_logpoint	○
pd_log_rec_leng	×
pd_log_remain_space_check	×
pd_log_rerun_reserved_file_open	×
pd_log_rerun_swap	×
pd_log_rollback_buff_count	×
pd_log_rpl_no_standby_file_opr	○
pd_log_sdinterval	○
pd_log_singleoperation	×
pd_log_swap_timeout	×
pd_log_unload_check	×
pd_log_write_buff_count	○
pd_lv_mirror_use	○
pd_master_file_name	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_max_access_tables	○
pd_max_access_tables_wrn_pnt	○
pd_max_add_dbbuff_no	×
pd_max_add_dbbuff_shm_no	×
pd_max_ard_process	×
pd_max_bes_process	−
pd_max_dic_process	−
pd_max_file_no	○※1
pd_max_file_no_wrn_pnt	○
pd_max_list_users	○
pd_max_list_count	○
pd_max_list_users_wrn_pnt	○
pd_max_list_count_wrn_pnt	○
pd_max_open_fds	×
pd_max_open_holdable_cursors	○
pd_max_rdarea_no	○※1
pd_max_rdarea_no_wrn_pnt	○
pd_max_recover_process	×
pd_max_reflect_process_count	○
pd_max_resident_rdarea_no	○
pd_max_resident_rdarea_shm_no	○
pd_max_server_process	○
pd_max_temporary_object_no	○
pd_max_tmp_table_rdarea_no	○
pd_max_users	○
pd_max_users_wrn_pnt	○
pd_max_utl_ios_file_no	○
pd_mlg_file_size	×
pd_mlg_msg_log_unit	−
pd_mlg_port	−
pd_mode_conf	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_module_trace_max	×
pd_module_trace_timer_level	×
pd_name_fixed_port_lookup	−
pd_name_port	○
pd_non_floatable_bes	−
pd_nowait_scan_option	×
pd_oltp_holdcr	×
pd_optimize_level	○
pd_os_authenticate	×
pd_overflow_suppress	×
pd_pageaccess_mode	×
pd_plugin_ixmk_dir	×
pd_prf_file_count	○
pd_prf_file_size	○
pd_prf_level	○
pd_prf_output_directory	×
pd_prf_trace	×
pd_process_count	○
pd_process_terminator	×
pd_process_terminator_max	×
pd_pth_trace_max	×
pd_queue_watch_time	×
pd_queue_watch_timeover_action	×
pd_rdarea_expand_format	×
pd_rdarea_extension_file	×
pd_rdarea_extension_timing	×
pd_rdarea_list_no_wrn_pnt	○
pd_rdarea_open_attribute	×
pd_rdarea_open_attribute_use	×
pd_rdarea_warning_point	×
pd_rdarea_warning_point_msgout	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_read_replica_max_trn_count	○
pd_read_replica_mode	○
pd_read_replica_use	○
pd_read_replica_workdir	○
pd_redo_allpage_put	×
pd_reduced_check_time	－
pd_registered_port	○
pd_registered_port_check	×
pd_registered_port_level	×
pd_registry_cache_size	○
pd_rise_disaster_mode	○
pd_rise_fence_level	○
pd_rise_pairvolume_combination	○
pd_rise_use	○
pd_rorg_predict	×
pd_routine_def_cache_size	○
pd_rpc_trace	×
pd_rpc_trace_name	×
pd_rpc_trace_size	×
pd_rpl_func_control	○
pd_rpl_hdepath	○
pd_rpl_init_start	○
pd_rpl_reflect_mode	－
pd_scd_port	－
pd_sds_shmpool_size	○
pd_security_host_group	○
pd_security_local_constraint	×
pd_server_cleanup_interval	×
pd_server_process_adjust_intvl	×
pd_server_process_adjust_limit	×
pd_service_port	○

オペランド名	参照 DB での HiRDB 定義変更可否
pd_shared_rdarea_use	−
pd_shmpool_attribute	×
pd_shmpool_control	−
pd_space_level	○
pd_spd_assurance_count	×
pd_spd_assurance_msg	×
pd_spd_dual	×
pd_spd_max_data_size	×
pd_spd_reduced_mode	×
pd_spd_reserved_file_auto_open	×
pd_spd_syncpoint_skip_limit	×
pd_spool_cleanup	×
pd_spool_cleanup_interval	×
pd_spool_cleanup_interval_level	×
pd_spool_cleanup_level	×
pd_sql_command_exec_users	×
pd_sql_dec_op_maxprec	×
pd_sql_mode	×
pd_sql_object_cache_size	○
pd_sql_simple_comment_use	×
pd_sqlobject_stat_timing	×
pd_standard_sqlstate	×
pd_start_level	−
pd_start_skip_unit	−
pd_start_time_out	×
pd_statistics	×
pd_stj_file_size	○
pd_stj_buff_size	○
pd_stored_lob_data_wrn_pnt	×
pd_sts_file_name_1〜7	○※2
pd_sts_initial_error	×

オペランド名	参照 DB での HiRDB 定義変更可否
pd_sts_last_active_file	○
pd_sts_last_active_side	○
pd_sts_last_active_side_sub	○
pd_sts_last_active_subfile	○
pd_sts_singleoperation	×
pd_sts_subfile_name_1～7	○
pd_substr_length	×
pd_svr_castoff_size	○
pd_syssts_file_name_1～7	○※2
pd_syssts_initial_error	×
pd_syssts_last_active_file	○
pd_syssts_last_active_side	○
pd_syssts_last_active_side_sub	○
pd_syssts_last_active_subfile	○
pd_syssts_singleoperation	×
pd_syssts_subfile_name_1～7	○
pd_system_complete_wait_time	×
pd_system_dbsync_point	×
pd_system_expand_unit	−
pd_system_id	×
pd_table_def_cache_size	○
pd_tcp_inet_bufsize	×
pd_tcp_unix_bufsize	×
pd_term_watch_count	×
pd_thdlock_wakeup_lock	×
pd_thread_max_stack_size	×
pd_thread_stack_expand_size	×
pd_tmp_directory	×
pd_tmp_table_initialize_timing	○
pd_tpyrced_key	×
pd_trn_commit_optimize	−

オペランド名	参照 DB での HiRDB 定義変更可否
pd_trn_port	−
pd_trn_rcvmsg_store_buflen	×
pd_trn_rerun_branch_auto_decide	−
pd_trn_rollback_msg_interval	×
pd_trn_rollback_watch_time	−
pd_trn_send_decision_interval	−
pd_trn_send_decision_intval_sec	−
pd_trn_send_decision_retry_time	−
pd_trn_watch_time	−
pd_type_def_cache_size	○
pd_uap_exerror_log_dir	×
pd_uap_exerror_log_param_size	○
pd_uap_exerror_log_size	×
pd_uap_exerror_log_use	○
pd_unique_check_mode	×
pd_unique_indexlock_info	×
pd_unit_id	○
pd_utl_buff_size	×
pd_utl_exec_mode	×
pd_utl_exec_time	×
pd_view_def_cache_size	○
pd_watch_pc_client_time	○
pd_watch_resource	○
pd_work_buff_expand_limit	○
pd_work_buff_mode	×
pd_work_buff_size	○
pd_work_table_option	×
pdaudload	○
pdbuffer	○
pdcltgrp	×
PDCONFPATH	×

オペランド名	参照 DB での HiRDB 定義変更可否
pdhagroup	−
pdhibegin	×
pdlogadfg -d rrp	○
pdlogadfg -d spd	○
pdlogadfg -d ssp	○
pdlogadfg -d sys	○
pdlogadpf -d rrp	○
pdlogadpf -d spd	○※2
pdlogadpf -d ssp	○
pdlogadpf -d sys	○※2
pdmlgput	×
pdplgprm	×
pdplugin	×
pdstart	○
pdstbegin	○
pdunit	○
pdwork	×
pdwork_wrn_pnt	○
SHMMAX	○
TZ	×

（凡例）

○：変更可能なオペランドです。

×：変更できないオペランドです。

−：未サポートオペランドのため，指定できないオペランドです。

注※1

オペランドの指定値を小さくすると，HiRDB を開始できない場合があります。

注※2

オペランドの追加，削除ができます。オペランドの変更はできません。

索引

数字

1：1 スタンバイレス型系切り替え機能　422
1 回の要求で起動，停止するサーバプロセス数の上限値　215
1 ビュー表当たりのビュー解析情報長　578
1 文字を表現する最大バイト数　480

A

aio ライブラリ　345
AND の複数インデクス利用　237

D

DECIMAL 型の符号正規化機能　228

H

HA グループ　179, 438
HDEPATH　440
HiRDB Datareplicator〔HiRDB システム定義の変更〕　21, 29
HiRDB Datareplicator 運用ディレクトリ　440
HiRDB Datareplicator に関するオペランド　439
HiRDB Non Recover FES　179
HiRDB Staticizer Option　444
HiRDB Text Search Plug-in　292
HiRDB 運用ディレクトリ名　162
HiRDB 識別子　161
HiRDB システム定義の作成方法〔HiRDB/シングルサーバ〕　16
HiRDB システム定義の作成方法〔HiRDB/パラレルサーバ〕　23
HiRDB システム定義の種類〔HiRDB/シングルサーバ〕　16
HiRDB システム定義の種類〔HiRDB/パラレルサーバ〕　22
HiRDB システム定義の変更手順〔HiRDB/シングルサーバ〕　19
HiRDB システム定義の変更手順〔HiRDB/パラレルサーバ〕　27
HiRDB システム定義ファイル〔HiRDB/シングルサーバ〕　17
HiRDB システム定義ファイル〔HiRDB/パラレルサーバ〕　23
HiRDB システム定義ファイルの共用化　30
HiRDB システム定義ファイルの共用化に関するオペランド　453
HiRDB システム定義ファイルの構成〔HiRDB/シングルサーバ〕　17
HiRDB システム定義ファイルの構成〔HiRDB/パラレルサーバ〕　24
HiRDB の開始方法に関するオペランド　202
HiRDB の処理方式に関するオペランド　207
HiRDB のポート番号　161, 162
HiRDB 予約ポート機能　461
HORCMINST　444

I

IP アドレス　420

J

JAR ファイル　475
Java Runtime Environment のルートディレクトリ　475
Java 仮想マシンの起動オプション　474
Java 仮想マシンのライブラリ　476
Java クラスパス　475
Java ストアドファンクション　474
Java ストアドプロシジャ　474
Java に関するオペランド　474

K

KFPA20009-W　478
KFPS01861-E　204
KFPS05078-I　204
KFPS05623-I　427

L

LOB 用グローバルバッファ 326
LRU 管理方式 319

O

OLTP に関するオペランド 450

P

PAUSE 状態 205
pd_additional_optimize_level 234
pd_alv_port 460
pd_apply_search_ats_num 504
pd_assurance_index_no 341
pd_assurance_table_no 341
pd_aud_async_buff_count 412
pd_aud_async_buff_retry_intvl 412
pd_aud_async_buff_size 410
pd_aud_auto_loading 414
pd_aud_file_name 408
pd_aud_file_wrn_pnt 413
pd_aud_max_generation_num 409
pd_aud_max_generation_size 409
pd_aud_no_standby_file_opr 410
pd_aud_sql_data_size 413
pd_aud_sql_source_size 413
pd_audit 408
pd_audit_def_buffer_size 417
pd_auth_cache_size 289
pd_auto_vrup 457
pd_bes_conn_hold_trn_interval 407
pd_bes_connection_hold 406
pd_bes_shmpool_size 298
pd_c_library_directory 477
pd_cancel_down_msgchange 214
pd_cancel_dump 393
pd_change_clt_ipaddr 461
pd_check_pending 342
pd_client_waittime_over_abort 393
pd_clt_waittime_over_dump_level 394
pd_cmd_exec_time 246
pd_cmd_process_conwaittime 250
pd_cmd_rconfig 217
pd_cmd_rlog 218
pd_cmd_rmode 217
pd_cmdhold_precheck 211
pd_command_deadlock_priority 265
pd_connect_auth_type 451
pd_connect_errmsg_hide 214
pd_constraint_name 342
pd_crypto_certificate_file 471
pd_crypto_private_key 471
pd_crypto_service_port 470
pd_cwaittime_report_dir 479
pd_cwaittime_report_size 479
pd_cwaittime_wrn_pnt 478
pd_db_access_error_action 314
pd_db_hold_action 316
pd_db_io_error_action 212
pd_db_timestamp_invalid_action 318
pd_dbbuff_attribute 295
pd_dbbuff_binary_data_lru 320
pd_dbbuff_dev_sector_size 321
pd_dbbuff_lru_option 319
pd_dbbuff_modify 321
pd_dbbuff_rate_updpage 322
pd_dbbuff_read_retry_count 317
pd_dbbuff_read_retry_intvl 317
pd_dbbuff_trace_level 323
pd_dbreuse_remaining_entries 276
pd_dbsync_altwrite_skip 501
pd_dbsync_point 207
pd_ddl_idx_pctfree 231
pd_ddl_tbl_fix_pctfree 231
pd_ddl_tbl_pctfree 230
pd_ddl_tbl_recovery 231
pd_deadlock_priority_use 265
pd_debug_info_netstat 398
pd_dec_sign_normalize 228

pd_def_buf_control_area_assign 287
pd_delete_reserved_word_file 241
pd_dfw_awt_process 195
pd_dfw_bulk_write 338
pd_dfw_syncpoint_skip_limit 256
pd_dic_shmpool_size 297
pd_down_watch_proc 252
pd_dump_suppress_watch_time 395
pd_duplicated_fs_suffix 483
pd_fes_lck_pool_partition 270
pd_fes_lck_pool_size 268
pd_floatable_bes 238
pd_ha 420
pd_ha_acttype 421
pd_ha_agent 422
pd_ha_ipaddr_inherit 420
pd_ha_max_act_guest_servers 424
pd_ha_max_server_process 425
pd_ha_mgr_rerun 434
pd_ha_observe_timeout 436
pd_ha_prc_cleanup_check 434
pd_ha_process_count 426
pd_ha_resource_act_wait_time 427
pd_ha_restart_failure 421
pd_ha_server_process_standby 422
pd_ha_switch_error 436
pd_ha_switch_timeout 433
pd_ha_transaction 428
pd_ha_trn_queuing_initial 431
pd_ha_trn_queuing_wait_time 429
pd_ha_trn_restart_retry_time 430
pd_ha_unit 421
pd_hash_table_size 235
pd_hashjoin_hashing_mode 503
pd_host_watch_interval 254
pd_hostname 180
pd_idx_without_rollback 281
pd_indexlock_mode 274
pd_inet_unix_bufmode 507
pd_inet_unix_utl_bufmode 519
pd_inner_replica_control 444
pd_inner_replica_hold_status 447
pd_inner_replica_lock_shift 445
pd_ipc_clt_conn_nblock 468
pd_ipc_clt_conn_nblock_time 469
pd_ipc_conn_count 468
pd_ipc_conn_interval 466
pd_ipc_conn_nblock 464
pd_ipc_conn_nblock_time 465
pd_ipc_file_dir 469
pd_ipc_inet_bufsize 515
pd_ipc_recv_count 464
pd_ipc_send_count 464
pd_ipc_send_retrycount 463
pd_ipc_send_retrysleeptime 464
pd_ipc_so_keepalive 472
pd_ipc_tcp_keepcnt 473
pd_ipc_tcp_keepidle 472
pd_ipc_tcp_keepintvl 473
pd_ipc_unix_bufsize 515
pd_isolation_level 281
pd_java_archive_directory 475
pd_java_castoff 477
pd_java_classpath 475
pd_java_libpath 476
pd_java_option 474
pd_java_routine_stack_size 474
pd_java_runtimepath 475
pd_java_stdout_file 476
pd_key_resource_type 266
pd_lck_deadlock_check 279
pd_lck_deadlock_check_interval 280
pd_lck_deadlock_check_pause 285
pd_lck_deadlock_check_timeout 283
pd_lck_deadlock_info 264
pd_lck_deadlock_sql_info 283
pd_lck_hash_entry 273
pd_lck_pool_partition 269

pd_lck_pool_size 266
pd_lck_queue_limit 265
pd_lck_until_disconnect_cnt 271
pd_lck_wait_timeout 264
pd_leap_second 520
pd_list_initialize_timing 244
pd_list_rowcount_in_rename 244
pd_listen_socket_bufset 517
pd_lock_uncommited_delete_data 274
pd_log_auto_expand_size 351
pd_log_auto_unload_path 346
pd_log_auto_unload_restart 347
pd_log_block_check 352
pd_log_dual 345
pd_log_dual_write_method 345
pd_log_fg_warning_point 352
pd_log_full_write_async 350
pd_log_max_data_size 531
pd_log_org_no_standby_file_opr 450
pd_log_org_reflected_logpoint 449
pd_log_rec_leng 532
pd_log_remain_space_check 346
pd_log_rerun_reserved_file_open 348
pd_log_rerun_swap 349
pd_log_rollback_buff_count 533
pd_log_rpl_no_standby_file_opr 440
pd_log_sdinterval 358
pd_log_singleoperation 348
pd_log_swap_timeout 349
pd_log_unload_check 349
pd_log_write_buff_count 350
pd_lv_mirror_use 446
pd_master_file_name 161
pd_max_access_tables 186
pd_max_access_tables_wrn_pnt 260
pd_max_add_dbbuff_no 335
pd_max_add_dbbuff_shm_no 336
pd_max_ard_process 194
pd_max_bes_process 187
pd_max_dic_process 188
pd_max_file_no 302
pd_max_file_no_wrn_pnt 261
pd_max_list_count 243
pd_max_list_count_wrn_pnt 262
pd_max_list_users 243
pd_max_list_users_wrn_pnt 262
pd_max_open_fds 193
pd_max_open_holdable_cursors 272
pd_max_rdarea_no 301
pd_max_rdarea_no_wrn_pnt 260
pd_max_recover_process 218
pd_max_reflect_process_count 448
pd_max_resident_rdarea_no 339
pd_max_resident_rdarea_shm_no 339
pd_max_server_process 183
pd_max_temporary_object_no 343
pd_max_tmp_table_rdarea_no 505
pd_max_users 181
pd_max_users_wrn_pnt 259
pd_max_utl_ios_file_no 484
pd_mlg_file_size 377
pd_mlg_msg_log_unit 376
pd_mlg_port 460
pd_mode_conf 202
pd_module_trace_max 405
pd_module_trace_timer_level 405
pd_name_fixed_port_lookup 458
pd_name_port 161
pd_non_floatable_bes 238
pd_nowait_scan_option 264
pd_oltp_holdcr 450
pd_optimize_level 232
pd_os_authenticate 451
pd_overflow_suppress 226
pd_pageaccess_mode 211
pd_plugin_ixmk_dir 456
pd_prf_file_count 390
pd_prf_file_size 391

pd_prf_level 388
pd_prf_output_directory 392
pd_prf_trace 388
pd_process_count 188
pd_process_terminator 209
pd_process_terminator_max 210
pd_pth_trace_max 406
pd_queue_watch_time 249
pd_queue_watch_timeover_action 250
pd_rdarea_expand_format 308
pd_rdarea_extension_file 310
pd_rdarea_extension_timing 308
pd_rdarea_list_no_wrn_pnt 263
pd_rdarea_open_attribute 311
pd_rdarea_open_attribute_use 310
pd_rdarea_warning_point 303
pd_rdarea_warning_point_msgout 306
pd_read_replica_max_trn_count 487
pd_read_replica_mode 485
pd_read_replica_use 485
pd_read_replica_workdir 486
pd_redo_allpage_put 219
pd_reduced_check_time 206
pd_registered_port 461
pd_registered_port_check 463
pd_registered_port_level 506
pd_registry_cache_size 292
pd_rise_disaster_mode 443
pd_rise_fence_level 444
pd_rise_pairvolume_combination 443
pd_rise_use 442
pd_rorg_predict 407
pd_routine_def_cache_size 291
pd_rpc_trace 387
pd_rpc_trace_name 387
pd_rpc_trace_size 387
pd_rpl_func_control 441
pd_rpl_hdepath 440
pd_rpl_init_start 439
pd_rpl_reflect_mode 439
pd_scd_port 459
pd_sds_shmpool_size 296
pd_security_host_group 417
pd_security_local_constraint 419
pd_server_cleanup_interval 191
pd_server_process_adjust_intvl 217
pd_server_process_adjust_limit 215
pd_service_port 457
pd_shared_rdarea_use 314
pd_shmpool_attribute 293
pd_shmpool_control 294
pd_space_level 227
pd_spd_assurance_count 356
pd_spd_assurance_msg 356
pd_spd_dual 356
pd_spd_max_data_size 358
pd_spd_reduced_mode 357
pd_spd_reserved_file_auto_open 357
pd_spd_syncpoint_skip_limit 255
pd_spool_cleanup 402
pd_spool_cleanup_interval 399
pd_spool_cleanup_interval_level 400
pd_spool_cleanup_level 403
pd_sql_command_exec_users 242
pd_sql_dec_op_maxprec 228
pd_sql_mode 230
pd_sql_object_cache_size 286
pd_sql_simple_comment_use 503
pd_sqlobject_stat_timing 379
pd_standard_sqlstate 242
pd_start_level 206
pd_start_skip_unit 206
pd_start_time_out 204
pd_statistics 377
pd_stj_buff_size 378
pd_stj_file_size 378
pd_stored_lob_data_wrn_pnt 306
pd_sts_file_name_1 369

pd_sts_initial_error 526
pd_sts_last_active_file 372
pd_sts_last_active_side 373
pd_sts_last_active_side_sub 375
pd_sts_last_active_subfile 373
pd_sts_singleoperation 530
pd_sts_subfile_name_1 370
pd_substr_length 480
pd_svr_castoff_size 191
pd_syssts_file_name_1 361
pd_syssts_initial_error 521
pd_syssts_last_active_file 365
pd_syssts_last_active_side 366
pd_syssts_last_active_side_sub 368
pd_syssts_last_active_subfile 366
pd_syssts_singleoperation 525
pd_syssts_subfile_name_1 363
pd_system_complete_wait_time 204
pd_system_dbsync_point 208
pd_system_expand_unit 161
pd_system_id 161
pd_table_def_cache_size 288
pd_tcp_inet_bufsize 516
pd_tcp_unix_bufsize 517
pd_term_watch_count 205
pd_thdlock_wakeup_lock 210
pd_thread_max_stack_size 287
pd_thread_stack_expand_size 288
pd_tmp_directory 482
pd_tmp_table_initialize_timing 344
pd_trn_commit_optimize 224
pd_trn_port 460
pd_trn_rcvmsg_store_buflen 222
pd_trn_rerun_branch_auto_decide 220
pd_trn_rollback_msg_interval 502
pd_trn_rollback_watch_time 225
pd_trn_send_decision_interval 221
pd_trn_send_decision_intval_sec 220
pd_trn_send_decision_retry_time 221
pd_trn_watch_time 221
pd_type_def_cache_size 290
pd_uap_exerror_log_dir 239
pd_uap_exerror_log_param_size 240
pd_uap_exerror_log_size 240
pd_uap_exerror_log_use 239
pd_uap_wait 490
pd_unique_check_mode 282
pd_unique_indexlock_info 281
pd_unit_id 180
pd_utl_buff_size 518
pd_utl_exec_mode 500
pd_utl_exec_time 245
pd_view_def_cache_size 289
pd_watch_pc_client_time 254
pd_watch_resource 258
pd_work_buff_expand_limit 199
pd_work_buff_mode 196
pd_work_buff_size 197
pd_work_table_option 236
pdacunlck コマンド 685
pdaudload 415
pdbuffer 324
pdchgconf コマンド 20, 27
pdcltgrp 454
pdconfchk コマンド 17, 24
pdconfchk コマンドでチェックできるオペランド 692
PDCONFPATH 453
PDCWAITTIME オーバ 252
PDDBACCS 492
pdhagroup 438
pdhibegin 387
pdlbuffer 490
pdlogadfg -d rrp 354
pdlogadfg -d spd 359
pdlogadfg -d ssp 360
pdlogadfg -d sys 353
pdlogadpf -d rrp 355
pdlogadpf -d spd 359

pdlogadpf -d ssp 361
pdlogadpf -d sys 353
pdmlgput 452
pdplgprm 455
pdplugin 455
pdstart 172
pdstart コマンドの完了待ち時間 204
pdstbegin 380
pdunit 162
pdvrup コマンドの自動起動 457
pdwork 201
pdwork_wrn_pnt 261
PRF トレース情報に関するオペランド 388
PRF トレース情報ファイルサイズ 391
PRF トレース情報ファイル世代数 390
PRF トレース情報ファイルを出力するディレクトリ 392
putenv 形式 33

R

RD エリアに関するオペランド 301
RD エリアのオープン契機 310
RD エリアの最大数 301
RD エリアの閉塞状態のチェック 211
RPC トレース情報に関するオペランド 387

S

set 形式 33
SHMMAX 299
SQLSTATE の細分化に関するオペランド 242
SQL オブジェクト長の計算式 568
SQL オブジェクト用バッファ長 286
SQL オブジェクト用バッファ長（pd_sql_object_cache_size）の見積もり式 568
SQL 拡張最適化オプション 234
SQL からのコマンド実行に関するオペランド 242
SQL 最適化オプション 232
SQL 実行時間警告出力機能に関するオペランド 478
SQL 実行時間警告情報ファイル 478
SQL の仕様に関するオペランド 226
SQL 予約語削除機能に関するオペランド 241
SQL 予約語削除ファイル 241
SSH クライアント設定ファイル 217

T

TIMEOUT 204
TZ 451

U

UAP 環境定義 488
UAP 環境定義の追加又は変更手順〔HiRDB/シングルサーバ〕 21
UAP 環境定義の追加又は変更手順〔HiRDB/パラレルサーバ〕 29

W

WITHOUT LOCK NOWAIT 指定の検索 264

あ

空き領域の再利用機能 341
後処理プロセス 209
アンロード状態のチェックを解除する運用 349
アンロードログファイルの出力先ディレクトリ 346

い

一時表に関するオペランド 343
一括入力最大ページ数 331
一相コミット 224
一相最適化 224
インデクスキー値の排他資源の作成方法 266
インデクスキー値無排他 274
インデクス情報ファイル 456
インデクス用グローバルバッファ 327
インデクス用のローカルバッファ 491
インデクス予約数 341
インナレプリカ機能に関するオペランド 444
インナレプリカグループ 444
インナレプリカ最大グループ数 444
インメモリ RD エリア 339
インメモリデータ処理に関するオペランド 339

う

受け入れ可能状態　424
受け入れユニット　425
運用コマンド及びユティリティの実行監視時間　246
運用コマンド・ユティリティ実行におけるプロセス割り当て監視時間　250

え

影響分散スタンバイレス型系切り替え機能　422
演算中のエラー抑止　226

お

追い付き反映処理時に確保するプロセス数　448
オープン契機　310
オプション　33
オペランド指定値の見積もり式　562
オペランドの指定形式　33
オペランドの詳細　119
オンライン再編成上書き可能状態　449
オンライン再編成上書き禁止状態　449

か

開始準備処理の最大待ち時間　204
開始処理完了の連絡待ち時間　206
開始方法　203
開始モード　203
外部 C ストアドルーチンに関するオペランド　477
回復不要 FES　179
拡張 SQL エラー情報出力機能に関するオペランド 239
片系運転〔システムログファイル〕　348
監査証跡に出力する SQL データのサイズ　413
監査証跡に出力する SQL 文のサイズ　413
監査証跡ファイルの最大数　409
監査証跡ファイルの最大容量　409
監査証跡ファイル用の HiRDB ファイルシステム領域名　408
監視時間〔ユティリティ〕　245

き

行の継続　34
共用 RD エリア　314
共用化　30, 453
共用化の手順　32
共用ディレクトリ　32
共用メモリセグメントサイズの上限値　299
共用メモリセグメントの最大数　339
共用メモリに関するオペランド　293

く

空白変換機能　227
空白変換レベル　227
クライアント環境定義との関係〔HiRDB/シングルサーバ〕　18
クライアント環境定義との関係〔HiRDB/パラレルサーバ〕　26
クライアントグループに関するオペランド　454
クライアントグループの接続枠保証機能　454
クライアントの最大待ち時間　393
クラスパス　475
グローバルバッファが使用する共用メモリ　295
グローバルバッファに関するオペランド　319
グローバルバッファの LRU 管理方式　320
グローバルバッファの動的変更　321

け

系切り替え機能に関するオペランド　420
警告値　346
警告メッセージ　258, 265
ゲスト BES　425
検査制約　342
検査保留状態　342

こ

更新可能なオンライン再編成　448
更新バッファ量抑制機能　257
高速系切り替え機能　422
互換モード適用時に検討が必要なオペランド　494

コマンド形式　33
コマンド引数　33
コミットメント制御　224
コメントの記述　33

さ

サーバ共通定義　83
サーバ障害監視時間　433
サーバのプロセス数　183
サーバプロセスの異常終了回数の上限値　252
サーバプロセスの沈み込み　249, 250
サーバプロセスの分割起動，停止要求間隔　217
サーバプロセスのメモリサイズ監視機能　191
サーバ名　172
サーバモード　421
サーバ用ステータスファイル（障害発生時）に関するオペランド　372
サーバ用ステータスファイルに関するオペランド　369
最大起動プロセス数〔ディクショナリサーバ〕　188
最大起動プロセス数〔バックエンドサーバ〕　187
最大常駐プロセス数　189
最大同時起動サーバプロセス数　183
最大同時接続数　181
最大待ち時間　255
再編成時期予測機能に関するオペランド　407
作業表に関するオペランド　196
作業表用バッファ長　197
作業表用バッファの自動増分　199
作業表用ファイル用の HiRDB ファイルシステム領域　201, 261
参照 DB を使用するかどうか　485
参照制約　342
参照制約及び検査制約に関するオペランド　342

し

識別子　161
システム監視に関するオペランド　245
システム共通定義　55
システム構成に関するオペランド　161
システム構成変更コマンド　20, 27
システム定義スカラ関数の定義情報長の求め方　591
システムログの並列出力機能　345
システムログファイルに関するオペランド　345
システムログファイルの空き容量監視機能　346
システムログファイルのアンロード状態　349
システムログファイルの片系運転　348
システムログファイルの自動オープン　348
システムログファイルのスワップ　349
システムログファイルのスワップ完了待ち時間　349
システムログファイルの二重化　345
システムログファイルのファイルグループ　353
システムログファイルの両系運転　348
実体のないステータスファイル　362, 369
実体のない副ステータスファイル　364, 371
指定不要になったオペランド　50
自動オープン〔システムログファイル〕　348
自動オープン〔シンクポイントダンプファイル〕　357
自動開始　202
自動増分〔作業表用バッファ〕　199
自動ログアンロード機能　346
絞込み検索に関するオペランド　243
終了モード　203
縮退運転　357
縮退起動に関するオペランド　206
手動開始　202
常駐プロセス数　189
シンクポイントダンプの取得間隔　358
シンクポイントダンプの有効化　356
シンクポイントダンプファイルに関するオペランド 356
シンクポイントダンプファイルの自動オープン　357
シンクポイントダンプファイルの縮退運転　357
シンクポイントダンプファイルの世代数　356
シンクポイントダンプファイルの二重化　356
シンクポイントダンプファイルのバッファ長　358
シンクポイントダンプファイルのファイルグループ 359
シンクポイントダンプ有効化のスキップ回数監視機能　255
シングルサーバ定義　89

シングルサーバ用共用メモリサイズ　296

す

スケジューラプロセスのポート番号　162
スタックサイズ　287
スタック領域長〔Java〕　474
スタンバイ型とスタンバイレス型を混合した構成〔HiRDB システム定義例〕　558
ステータスファイル　362, 369
スナップショット方式　211
スレッド間ロックの解放通知方式　210

せ

整合性チェックユティリティ　685
セキュリティに関するオペランド　408
セグメント使用率　303
接続枠保証機能　454
全面回復処理に関するオペランド　218
全面回復処理の並列実行プロセス数　218

た

タイムアウト情報　264
タイムゾーン　451

つ

通信処理に関するオペランド　457

て

定義の関係〔HiRDB/シングルサーバ〕　18
定義の関係〔HiRDB/パラレルサーバ〕　25
定義例　540
ディクショナリサーバ定義　104
ディクショナリサーバ用共用メモリサイズ　297
データベース構成変更ユティリティ　686
データベース再編成ユティリティ〔排他資源数〕　683
データベース作成ユティリティ〔排他資源数〕　683
データ用のローカルバッファ　491
適用 OS ごとのオペランド指定可否　706
デッドロック監視処理中断時の監視処理休止時間　285
デッドロック監視処理の最大処理時間　283
デッドロック情報　264
デッドロックの優先順位　265
デファードライト処理の並列 WRITE 機能　196
デファードライトトリガでの更新ページ出力比率　331

と

統計情報に関するオペランド　377
統計ログ　377
統計ログバッファ長　378
統計ログファイル（pd_stj_file_size）の見積もり式　562
統計ログファイルの最大容量　378
同時アクセス可能実表数　186, 260
同時実行最大数に関するオペランド　181
動的追加用共用メモリセグメント数の上限値　336
動的追加用グローバルバッファ数の上限値　335
トラブルシュート情報に関するオペランド　393
トラブルシュート情報の削除　402
トラブルシュート情報の取得　393
トラブルシュート情報の取得〔ネットワーク情報〕　399
トランザクションキャンセル時のプロセスダウンメッセージ変更機能　214
トランザクションキューイング機能　428
トランザクション決着処理に関するオペランド　220
トランザクションサーバプロセスのポート番号　162
トランザクションの同期点処理　222

に

二相コミット　225
入出力エラー　212

は

バージョンアップに関するオペランド　457
バージョンによって省略値が異なるオペランド　48
排他資源数　271
排他資源数の見積もり　593
排他制御で使用する共用メモリ領域　266, 268
排他制御に関するオペランド　264
排他制御用プール　266, 268
排他制御用プールサイズ　266, 268

排他待ち限界経過時間　264
排他待ち時間　264
排他待ちユーザ数　265
バックエンドサーバ接続保持機能に関するオペランド　406
バックエンドサーバ接続保持時間　407
バックエンドサーバ定義　111
バックエンドサーバ用共用メモリサイズ　298
ハッシュ表サイズ　235
バッファサイズ　330
バッファに関するオペランド　286
バッファの確保方式　196
バッファ面数〔グローバルバッファ〕　328
バッファ面数〔ローカルバッファ〕　492

ひ

日付・時刻に関するオペランド　451
非同期 READ 機能　194
非同期 READ プロセス数　194
ビュー解析情報用バッファ長　289
表定義キャッシュサイズの求め方　587
表定義情報バッファサイズ　586
表定義情報用バッファ長　288
表定義情報用バッファ長（pd_table_def_cache_size）の見積もり式　584
表又はインデクスの予約数に関するオペランド　341
表予約数　341

ふ

ファイルグループ〔システムログファイル〕　353
ファイルグループ〔シンクポイントダンプファイル〕　359
ファイルグループ〔副シンクポイントダンプファイル〕　360
副ステータスファイル　363, 370
プラグイン　292
プラグインが使用する共用メモリ　456
プラグイン関数の定義情報長の求め方　591
プラグインに関するオペランド　455
フラグ引数　33
プリフェッチ機能　330, 492
フロータブルサーバ　238
プロセス数の最大値　183
プロセスに関するオペランド　187
プロセスの異常終了回数監視機能　252
フロントエンドサーバ定義　97
文法規則　33

へ

閉塞状態のチェック　211
ページアクセス方式　211

ほ

ポート番号　161
ポート番号の範囲を指定　461
ホスト BES　425
ホスト間監視時間間隔　254
ホスト名　162, 172

ま

前処理の排他要求数　681
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称　161
マルチコネクションアドレス機能　174, 177

み

未決着状態のトランザクション　220

む

無応答障害　245

め

メッセージキュー監視機能　249
メッセージキュー監視時間　249
メッセージの出力抑止機能に関するオペランド　452
メッセージログサーバプロセスのポート番号　162
メッセージログファイルに関するオペランド　376
メッセージログファイルの最大容量　377

も

文字コードに関するオペランド　480
モニタモード　421

ゆ

有効化処理　255
有効保証世代数　356
ユーザ権限情報用バッファ長　289
ユーザサーバホットスタンバイ　422
ユーザ定義型　290
ユーザ定義型情報用バッファ長　290
ユーザ認証に関するオペランド　451
ユティリティの実行監視時間　245
ユニット監視プロセスのポート番号　162
ユニット識別子　162, 172, 180
ユニット制御情報定義　74
ユニット用ステータスファイル（障害発生時）に関するオペランド　365
ユニット用ステータスファイルに関するオペランド 361

り

リアルタイム SAN レプリケーションに関するオペランド　442
リードレプリカ機能適用時の参照 DB の定義変更可否　719
リスト　243
リスト作成数　262, 263
リスト作成ユーザ数　262
リストの初期化時期　244
リソースの使用率　258
リバランスユティリティ〔排他要求数〕　684

る

ルーチン制御用オブジェクト長の計算式　569
ルーチン定義情報長の求め方　590
ルーチン定義情報用バッファ長　291
ルーチン定義情報用バッファ長（pd_routine_def_cache_size）の見積もり式　590

れ

レジストリ情報用バッファ長　292
レプリカ RD エリアのオープン属性　446
連続異常終了回数の上限　205

ろ

ローカルバッファ　490
ロールバック完了応答の最大待ち時間　225
ロールバック指示再送限界時間　225
ログ出力バッファ面数　350
論理ファイル名　362, 363, 369, 370

わ

ワークファイル出力先ディレクトリの変更　482