

HiRDBデータベース再編成 運用方法の解説

2011/08

株式会社 日立製作所 情報・通信システム社
ITプラットフォーム事業本部 開発統括本部 DB設計部

Contents



1. はじめに
2. **HiRDBのデータベース構造**
3. **HiRDBのデータ格納方法**
4. データベースの断片化を監視する運用
5. データベースの断片化を低減する運用
6. データベースの断片化を解消する運用
7. 運用事例
8. トラブル事例

1. はじめに

「止めない」設計思想を貫く 高信頼ノンストップデータベース

社会基盤を支えるために
日立が自社開発にこだわり続ける純国産RDBMS

ハイアールディービー

HiRDB Version **9**

Highly Scalable Relational DataBase

今まで培った信頼性をベースに
クラウド時代を支える「ワンランク上の」
高性能・高信頼データベースを目指します。

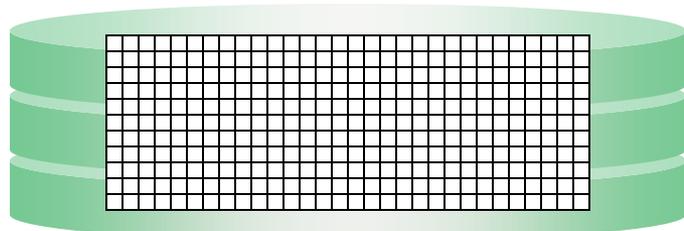
解説

データベースシステムを安定稼働させるためには、データベースを設計する際に、再編成に関する運用も含めて計画することが重要です。データベースの再編成を実施することで、不要な領域の増加を防いだり、データへのアクセス性能を保持することができます。

■データベースの格納状態

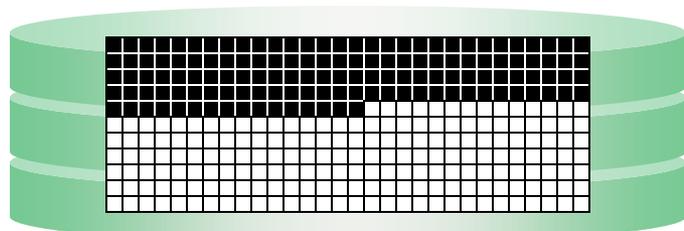
①初期状態

データが格納されていない状態です。



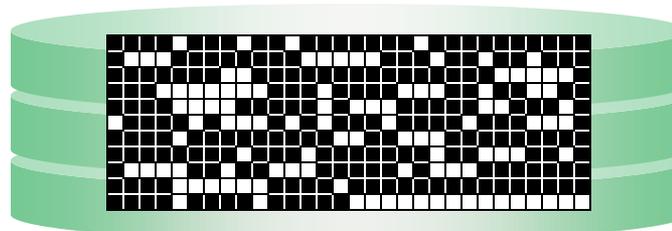
②初期データの登録

初期データの登録直後は、データが連続した状態で格納されています。



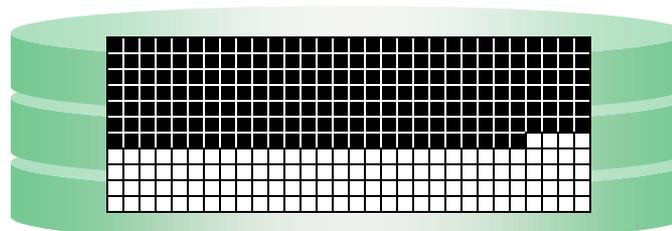
③断片化の発生

データの追加・削除を実行する業務を継続していると、データベースが断片化されます。



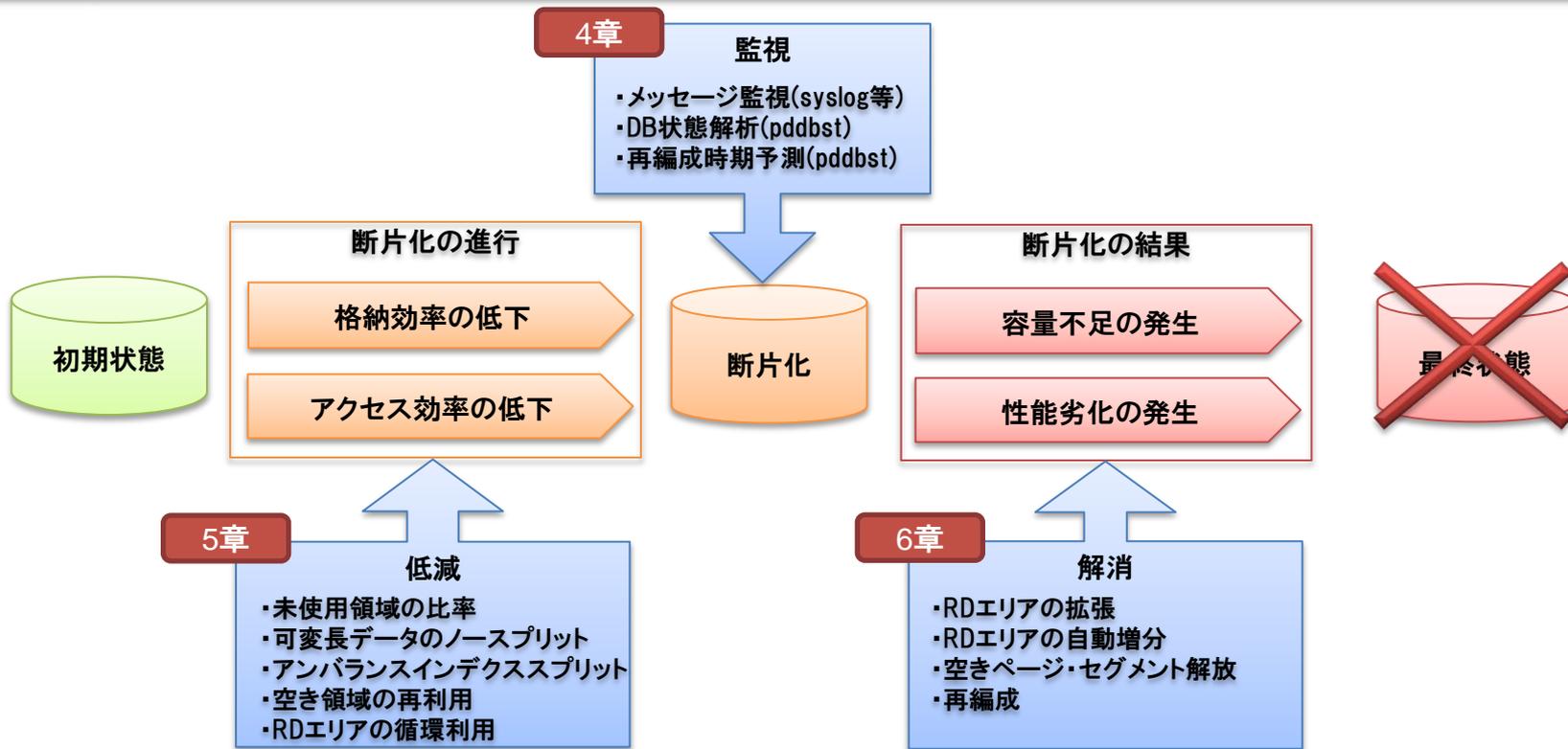
④データベースの再編成

データベースの再編成を実行することで、データベースの断片化を解消します。



解説

データベースの断片化が進行すると、格納効率やアクセス効率の低下が発生します。更に進行すると、容量不足や性能劣化が発生します。HiRDBでは、断片化の状況を監視方法、進行の低減や解消する方法を用意しています。また、これら前提として、HiRDBのデータベース構造や格納方法を理解しておくことも重要となります。



3章	データ格納の方法	・通常データ	・可変長データ	・インデクス
-----------	----------	--------	---------	--------

2章	データベースの構造	・RDエリア	・セグメント	・ページ
-----------	-----------	--------	--------	------

解説 本資料では、再編成運用を検討するにあたり、全体像に示した内容を2章～6章で解説します。更に7章で適用事例や8章でトラブル事例を挙げて、理解を深めます。

■本資料の内容

◆再編成運用を理解するための前提知識を解説します。

2章: HiRDBのデータベース構造

3章: HiRDBのデータ格納方法

◆データベースの断片化への各種対処方法を解説します。

4章: データベースの断片化を監視する方法

5章: データベースの断片化を低減する方法

6章: データベースの断片化を解消する方法

◆事例を示して理解を深めます。

7章: 運用事例

データベースシステムの例を挙げて、それぞれの業務に合ったデータベース再編成運用を説明します。

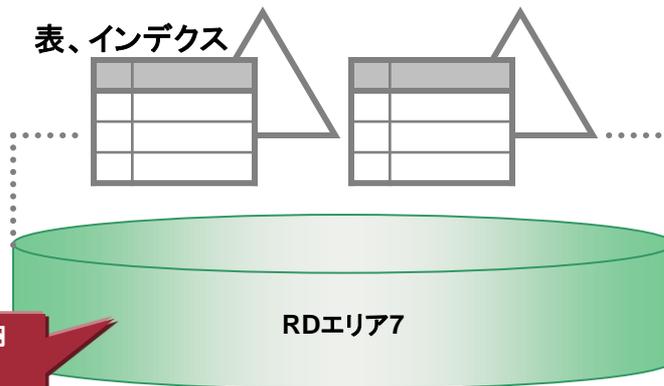
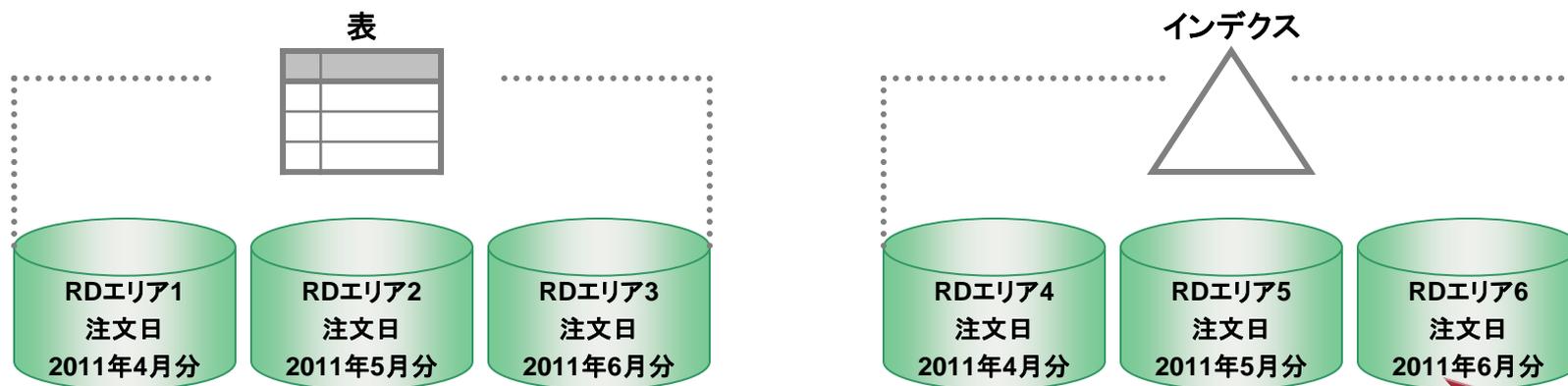
8章: トラブル事例

業務に適していないデータベース再編成運用方法を選択した場合のトラブル事例について、紹介します。

2. HiRDBのデータベース構造

解説

表やインデクスは、RDエリアという論理的な単位に格納します。1つの表もしくはインデクスを、複数のRDエリアに分割して格納できます。また、1つのRDエリアに、複数の表やインデクスを格納することもできます。

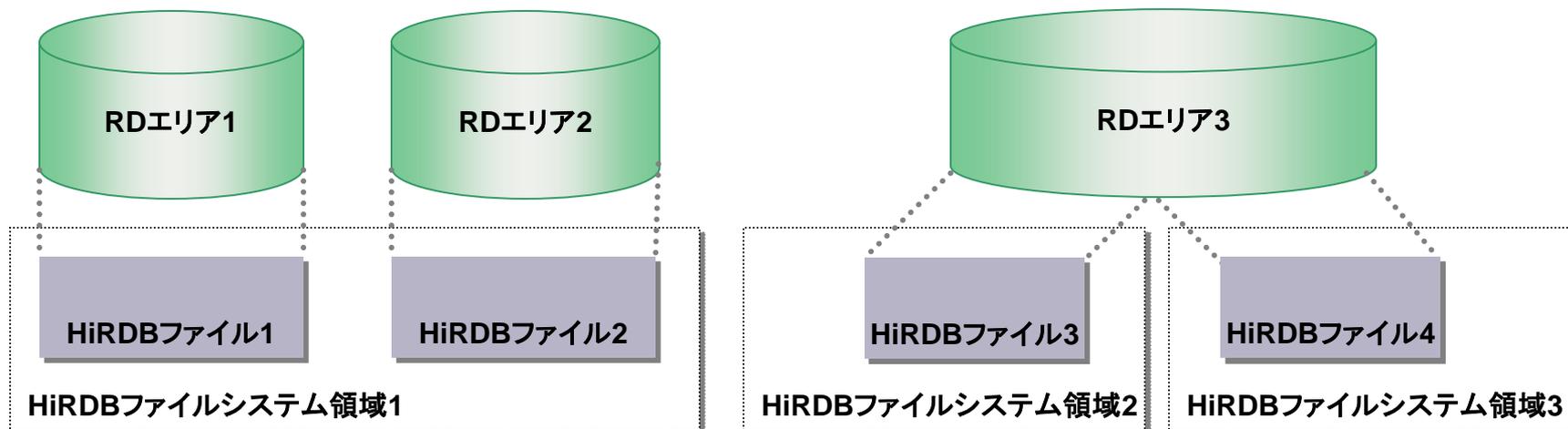


1つの表およびインデクスは、最大1024個のRDエリアに分割して格納できます。

1つのRDエリアに格納できる資源の数
◆表:最大500
◆インデクス:最大500

解説

RDエリアは、HiRDBファイルで構成されています。表やインデクスのデータは、HiRDBファイルに格納します。HiRDBファイルは、HiRDBファイルシステム領域に作成されます。1つのHiRDBファイルの最大サイズは、64GBです。



HiRDBファイルシステム領域とは

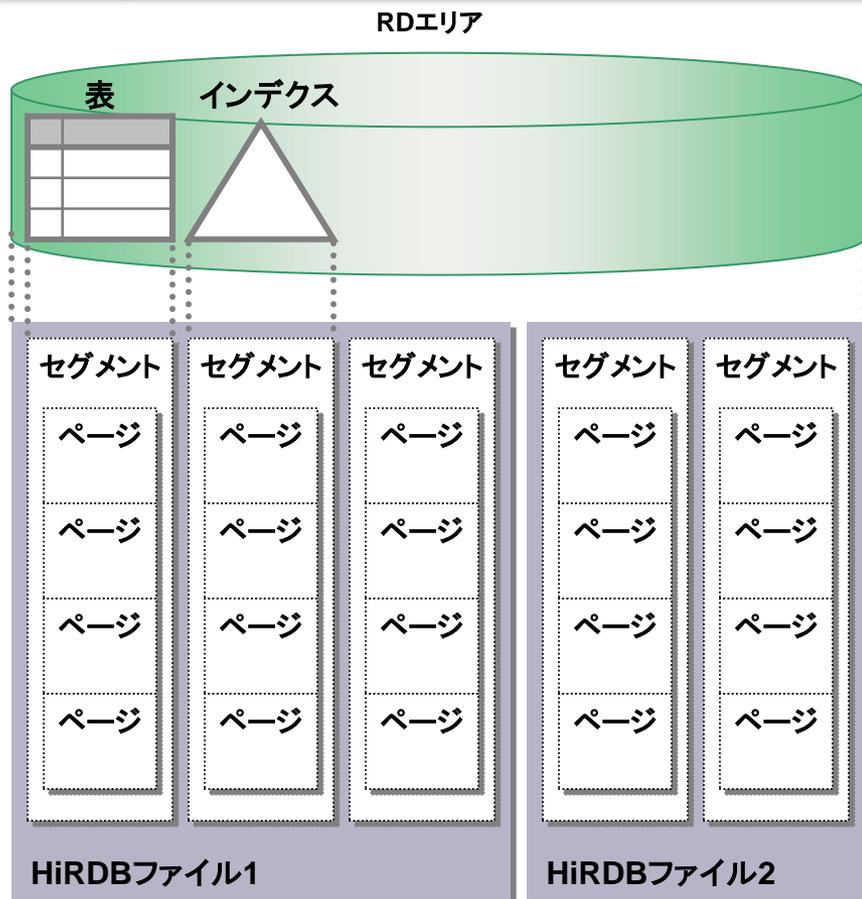
- ◆HiRDBファイルを作成する領域です。HiRDB自身で、独自のファイルシステムを実現しています。
- ◆RAWデバイスや、OSのファイルシステム上のファイルに割り当てます。

RDエリアの最大サイズ

1つのRDエリアを構成するHiRDBファイルの数は、最大16です。1つのHiRDBファイルの最大サイズが、64GBであるため、最大1TBのRDエリアを構築することができます。

解説

表やインデクスのデータを格納するHiRDBファイルは、セグメントとページという単位で構成されています。ページサイズやセグメントサイズ(1セグメント当たりのページ数)は、RDエリアごとに指定できます。



ページとは

- ◆データを入出力する場合の、最小単位です。
- ◆1ページのサイズは、4KBから30KBまでの範囲で指定できます。ただし、2KBの倍数にしてください。

セグメントとは

- ◆連続した複数のページで構成されており、表やインデクスをRDエリアに格納する場合の割り当て単位です。つまり、1つのセグメントには、1つの資源(表もしくはインデクス)だけが格納されます。
- ◆RDエリアの使用率は、セグメントの使用率で判断します。
- ◆セグメントサイズは、1から16,000ページまでの範囲で指定できます。
- ◆複数のHiRDBファイルでRDエリアを構成する場合、HiRDBファイルごとにセグメント数を変更することができます。1つのHiRDBファイルには、ファイルサイズの最大値(64GB)に達するセグメント数まで作成できます。

■ HiRDBファイルシステム領域の作成例 ■

pdfmkfs	: HiRDBファイルシステム領域の初期設定コマンド
-n 1024	: HiRDBファイルシステム領域サイズ(メガバイト単位)
-l 10	: HiRDBファイルシステム領域に作成するHiRDBファイルの最大数
-k DB	: 使用目的、RDエリア用のHiRDBファイルを作成する場合はDBを指定
-i	: HiRDBファイルシステム領域サイズ全体を初期化する
/HiRDB/DB	: HiRDBファイルシステム領域を割り当てるファイル名(パス名)

■ RDエリアの作成例(*1) ■

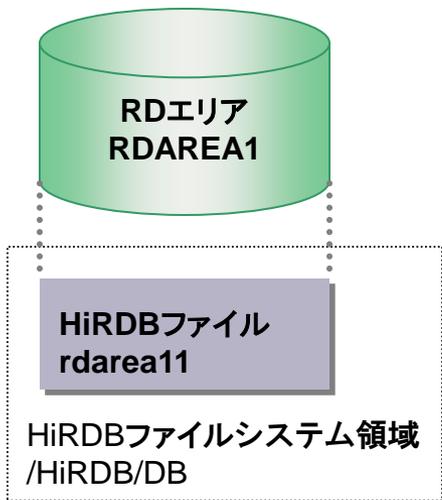
pdmod	: データベース構成変更ユーティリティ
-a /HiRDB/control_file/add	: 制御文ファイル名

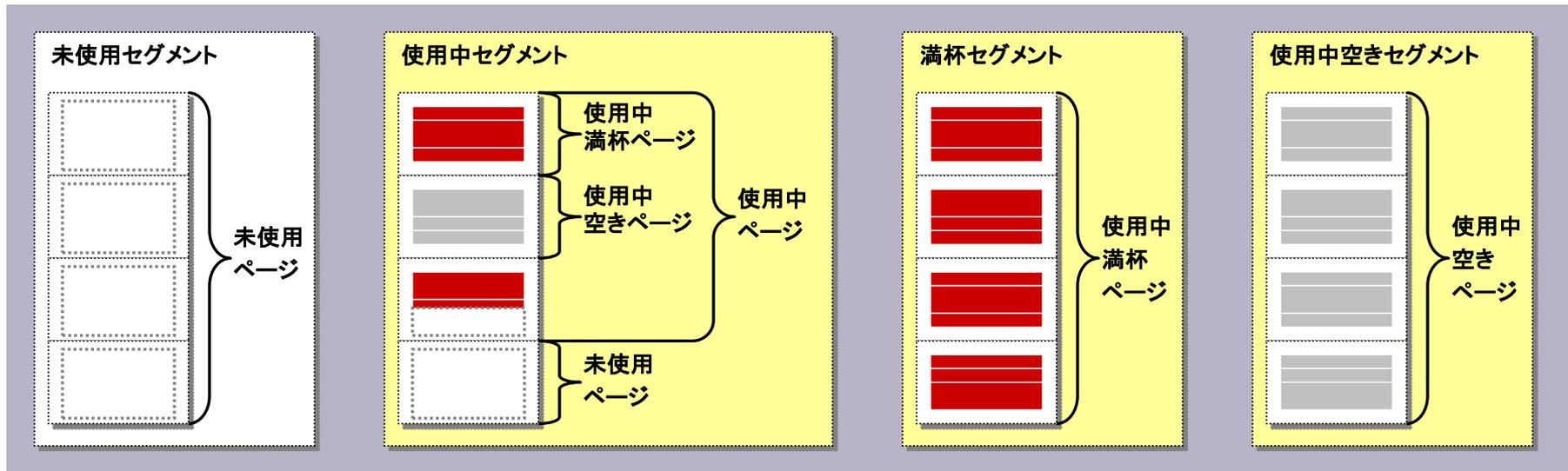
■ 制御文ファイル(/HiRDB/control_file/add)の内容 ■

create rdarea RDAREA1	: RDエリア名
for user used by PUBLIC	: RDエリア属性
page 4096 characters	: ページサイズ
storage control segment 50 pages	: セグメントサイズ
file name "/HiRDB/DB/rdarea11"	: RDエリアを構成するHiRDBファイル名
initial 1000 segments	: 上記HiRDBファイルに作成するセグメント数
;	

(*1): データベース初期構築後にRDエリアを追加作成する場合の例です。

データベース初期構築時にRDエリアを作成するコマンドは、データベース初期設定ユーティリティ(pdinit)を使用します。





■ : データ格納領域

■ : 空き領域(データの格納に割り当てられたことがあるが、データ削除などにより、現在はデータを格納していない領域)

□ : 未使用領域(データの格納に割り当てられていない領域)

セグメントの状態

◆未使用セグメント

表やインデスの格納先として割り当てられていないセグメントです。

◆使用中セグメント

表およびインデスの格納先として、割り当てられているセグメントです。使用中セグメントのうち、セグメント内の全ページが使用中満杯ページであるセグメントを「満杯セグメント」と呼びます。

また、セグメント内の全ページが使用中空きページであるセグメントを「使用中空きセグメント」と呼びます。

ページの状態

◆未使用ページ

データの格納に割り当てられていないページです。未使用セグメント内のページは、すべて未使用ページになります。

◆使用中ページ

データの格納に割り当てられているページです。使用中ページのうち、これ以上データを追加できる空き領域がないページを「使用中満杯ページ」と呼びます。また、データの削除などにより、全体が空き領域になっているページを「使用中空きページ」と呼びます。

3. HiRDBのデータ格納方法

3. HiRDBのデータ格納方法

3. 1 表データ

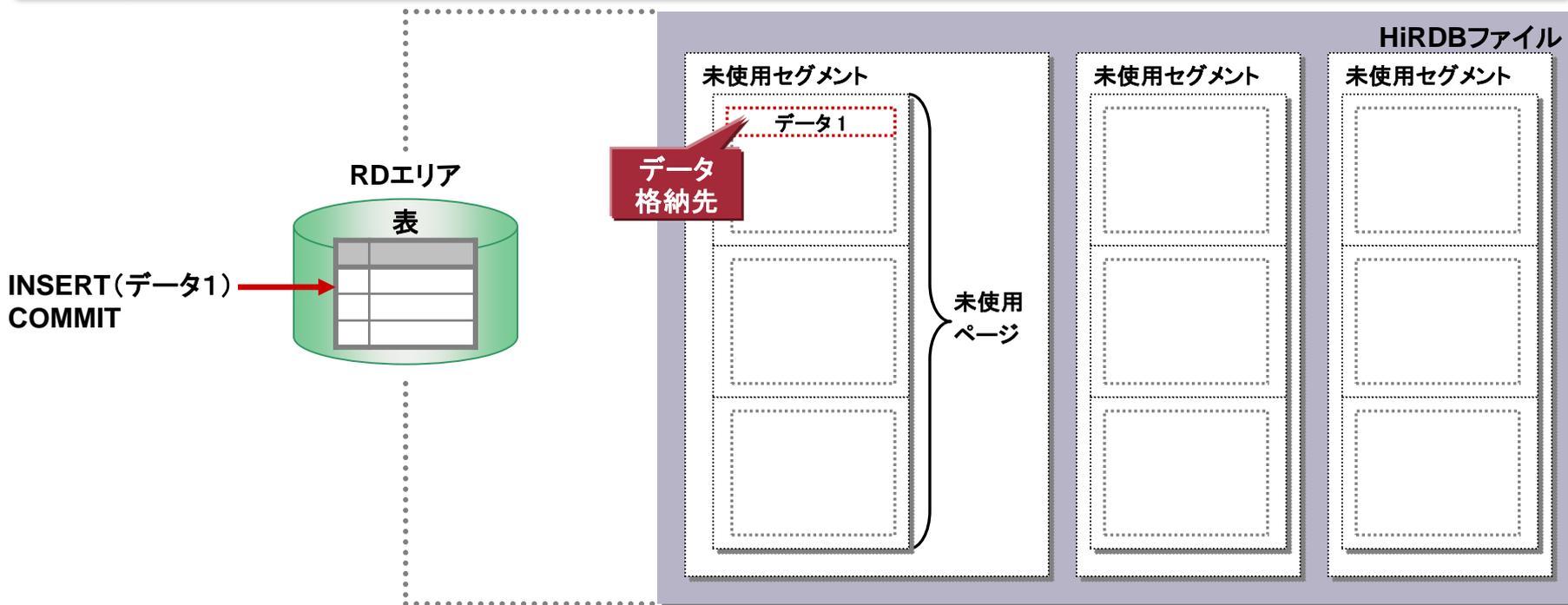
3. 2 表データ(可変長長大データ)

3. 3 インデクス

3-1-1 SQL操作による表データの格納方法 データの追加①

解説

表データの場合、1行目のデータを格納する時に、使用するセグメントを割り当てます。格納先RDエリアの未使用セグメントを1つ割り当て、そのセグメントの未使用ページにデータを格納します。

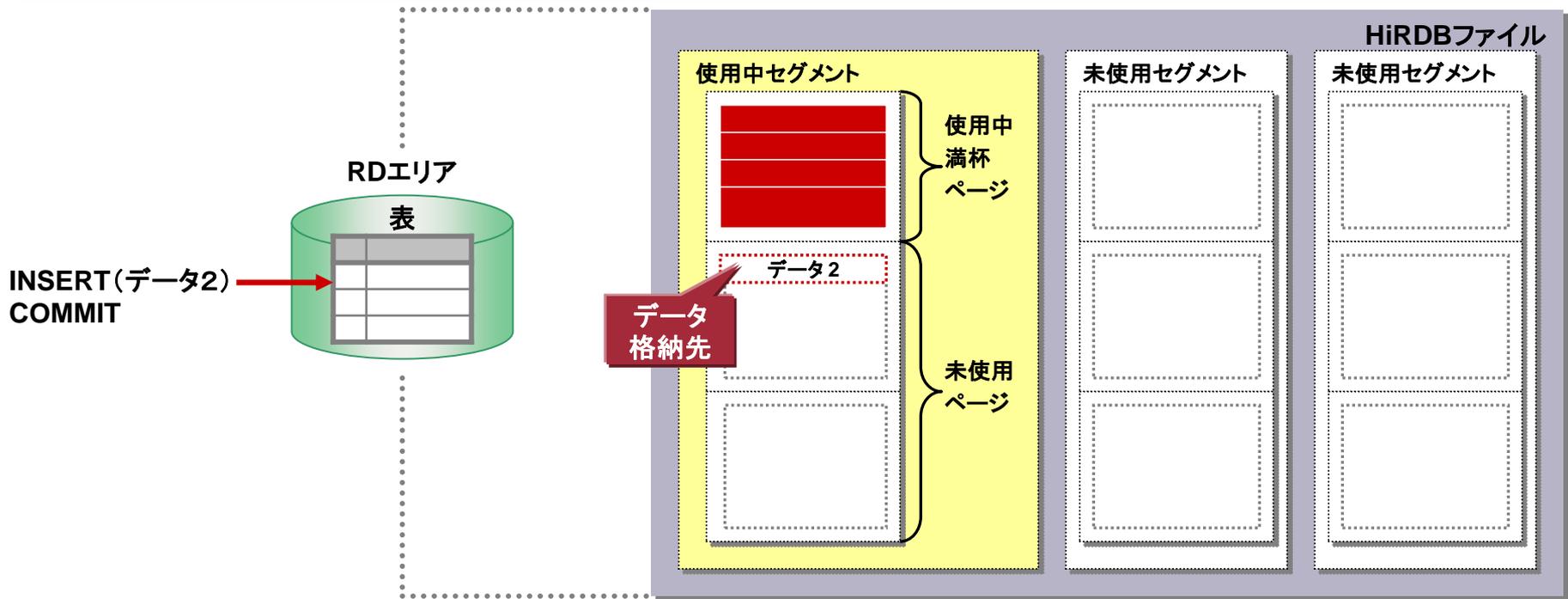


-----: 未使用領域

3-1-2 SQL操作による表データの格納方法 データの追加②

解説

2行目以降のデータは、前回格納したページに格納します。
前回格納したページに、データを格納できる領域がなくなった場合は、そのセグメント内の別のページを割り当て、格納します。

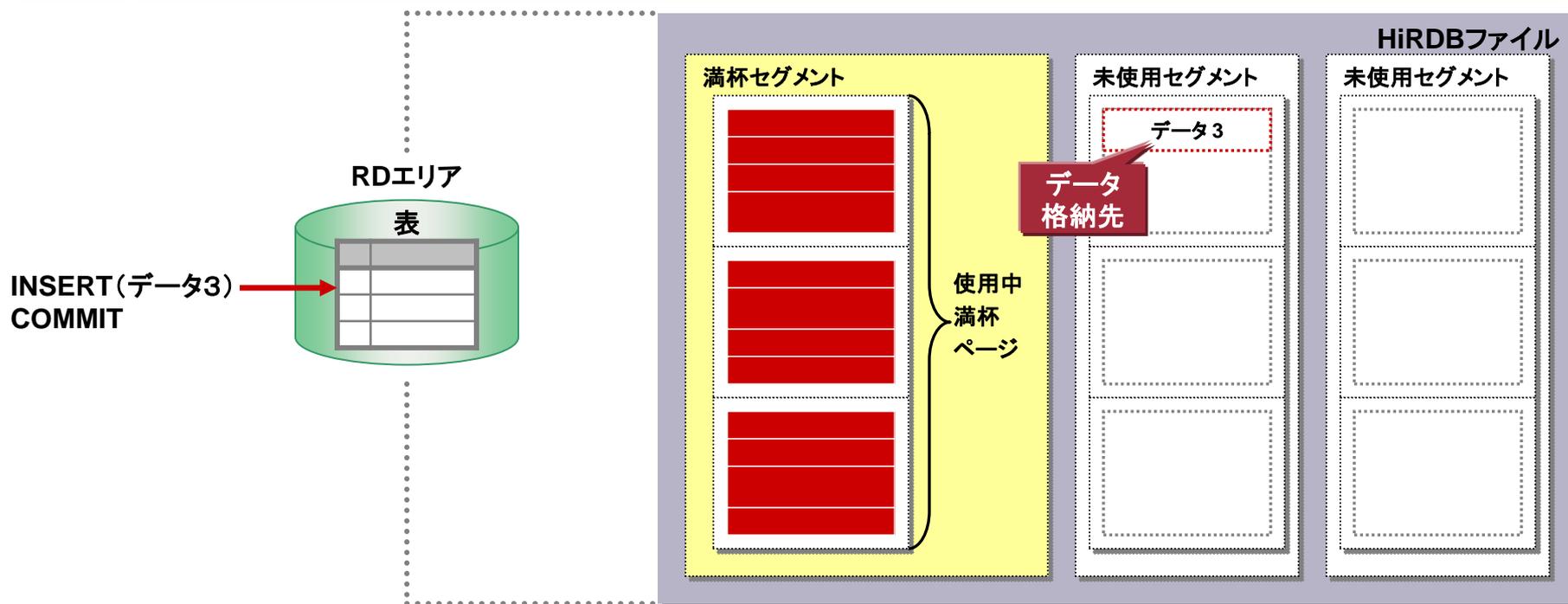


— : データ格納領域
⋯ : 未使用領域

3-1-3 SQL操作による表データの格納方法 データの追加③

解説

前回格納したページに、データを格納できる領域がなくなり、また、そのセグメント内に未使用ページが存在しない場合(満杯セグメント)は、未使用セグメントを新たに割り当て、新しいセグメント内の未使用ページにデータを格納します。

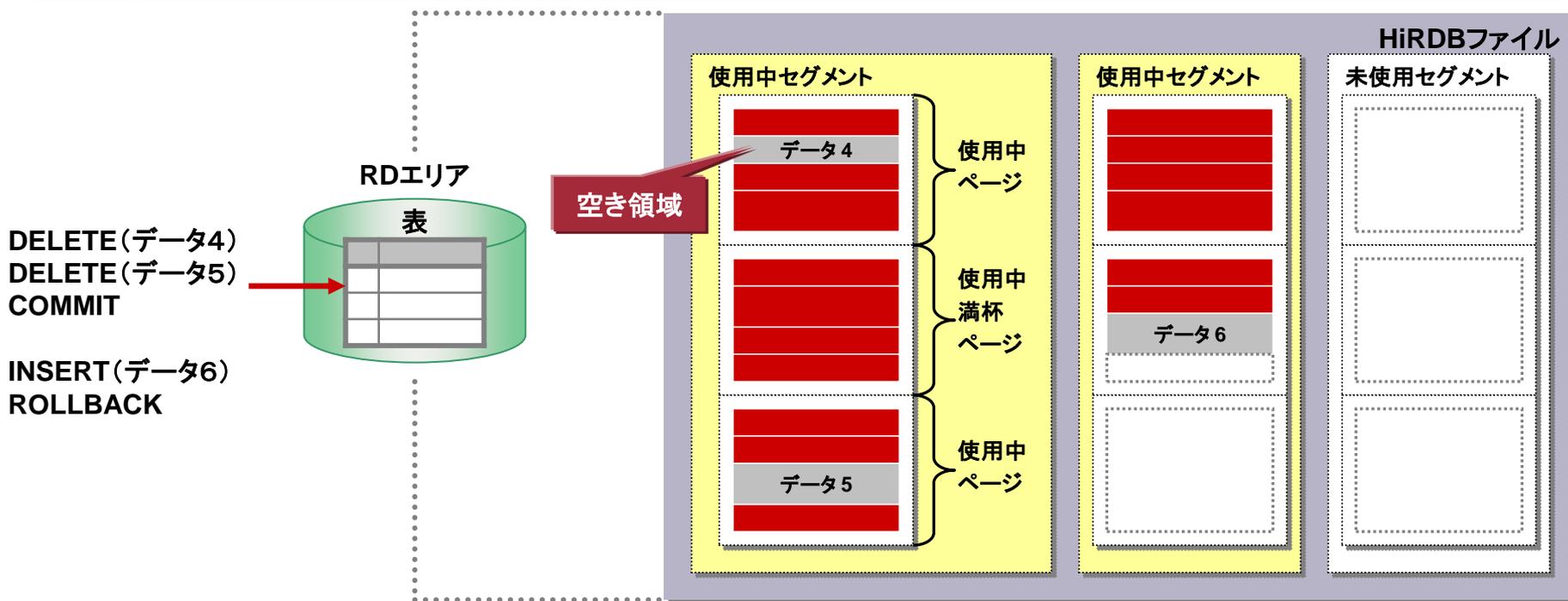


■ : データ格納領域
□ : 未使用領域

3-1-4 SQL操作による表データの格納方法 データの削除

解説

データを削除した場合、そのデータを格納していた領域は、空き領域となります。
また、INSERTを実行したトランザクションがロールバックした場合、そのトランザクションで確保した領域は、空き領域になります。

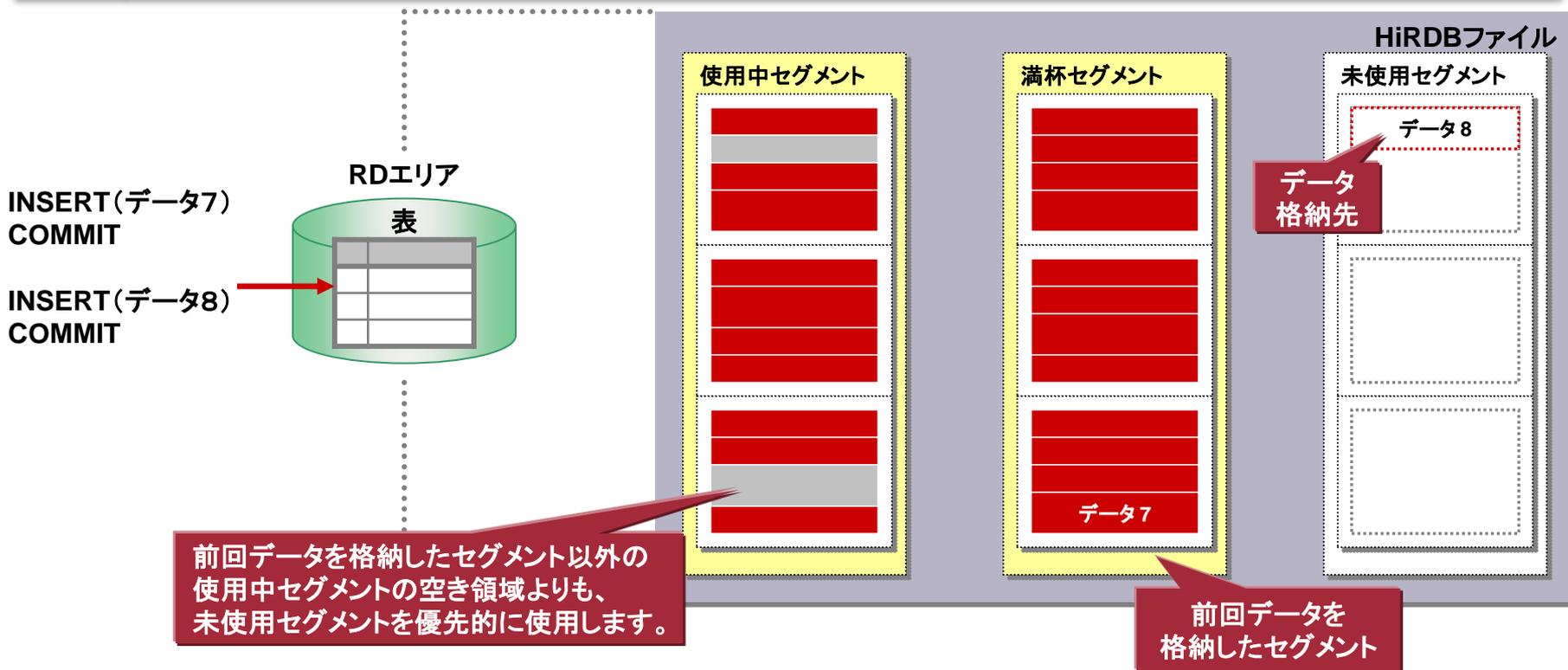


- : データ格納領域
- : 空き領域
- : 未使用領域

3-1-5 SQL操作による表データの格納方法 データの追加④

解説

前回データを格納したセグメントに空き領域が存在する場合は、その空き領域を優先して使用します。満杯セグメントになった時点で、新たな未使用セグメントを割り当てます。

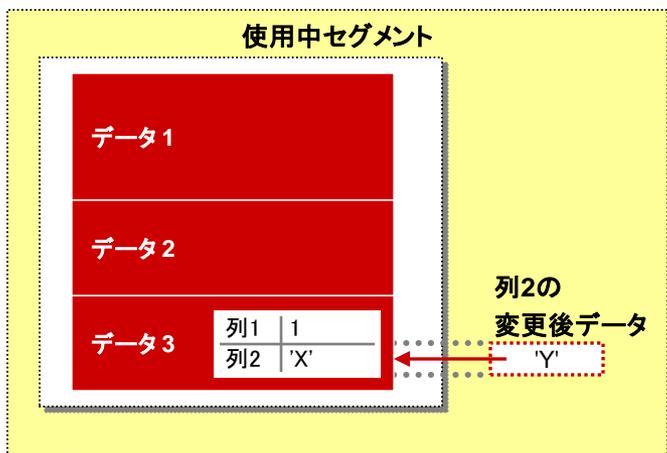


- : データ格納領域
- : 空き領域
- ⋯ : 未使用領域

解説

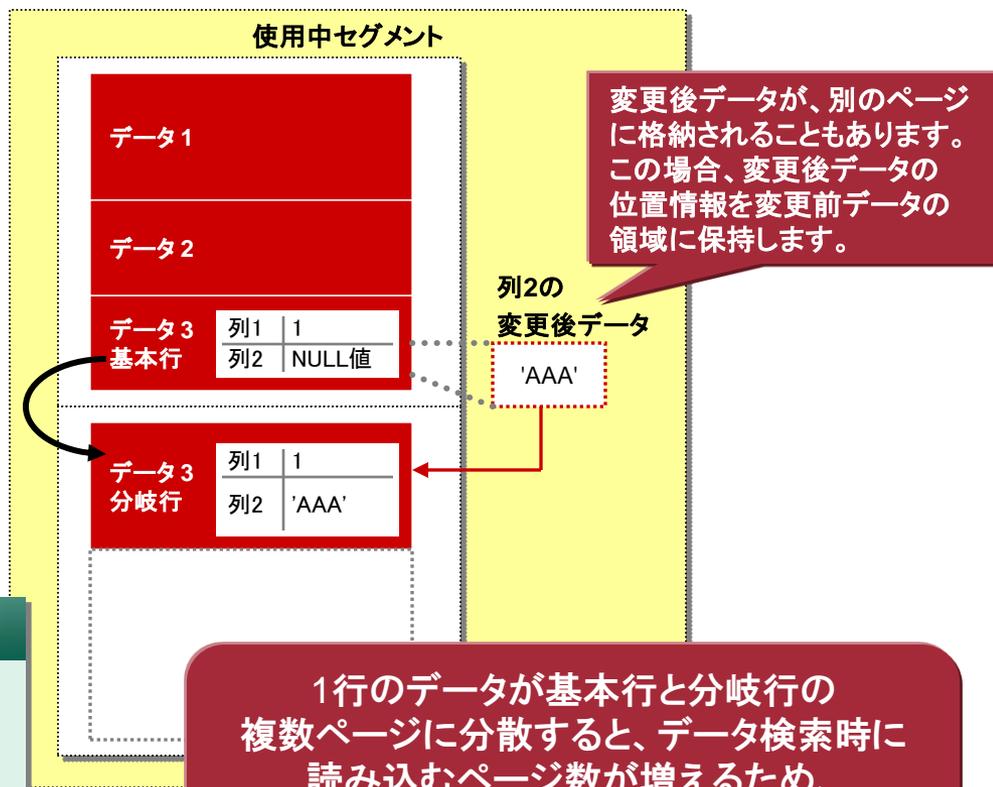
変更前後のデータ長が同じ、もしくは変更後のデータ長が短くなる場合は、データ格納位置は変更せず、データの内容を更新します。変更後のデータ長が長くなる場合は、変更後データを別の新たな領域に格納します。

■変更前後のデータ長が同じ場合



— : データ格納領域
⋯ : 未使用領域

■変更後のデータ長が長い場合



1行のデータが基本行と分岐行の複数ページに分散すると、データ検索時に読み込むページ数が増えるため、検索性能に影響を及ぼすことがあります。

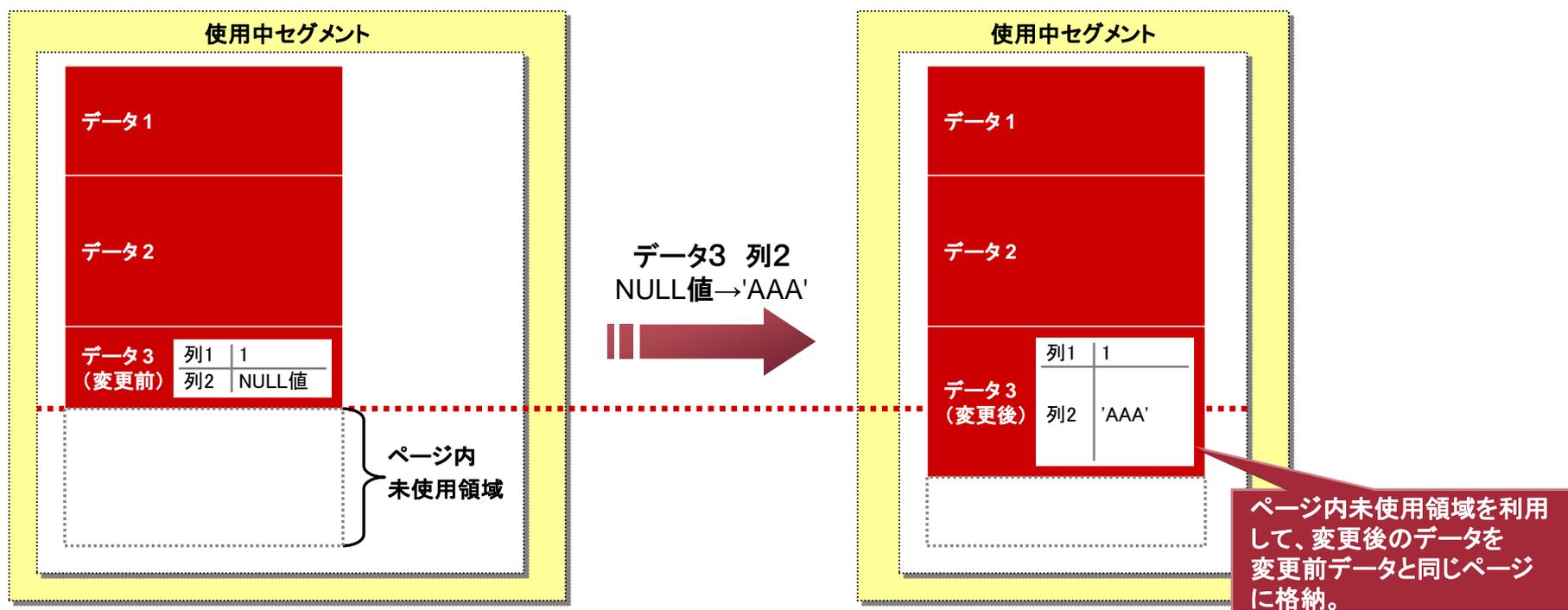
基本行と分岐行

- ◆基本行:すべての列の基本となるデータを格納する行。分岐行が存在する場合は、分岐先の位置情報を保持している。
- ◆分岐行:基本行とは別のページに格納された行。

解説

ページ内未使用領域を設定しておくことで、データ長が長くなる更新をした場合でも、変更後データが別ページに分岐する可能性を抑えることができます。

■ ページ内に未使用領域が存在する場合



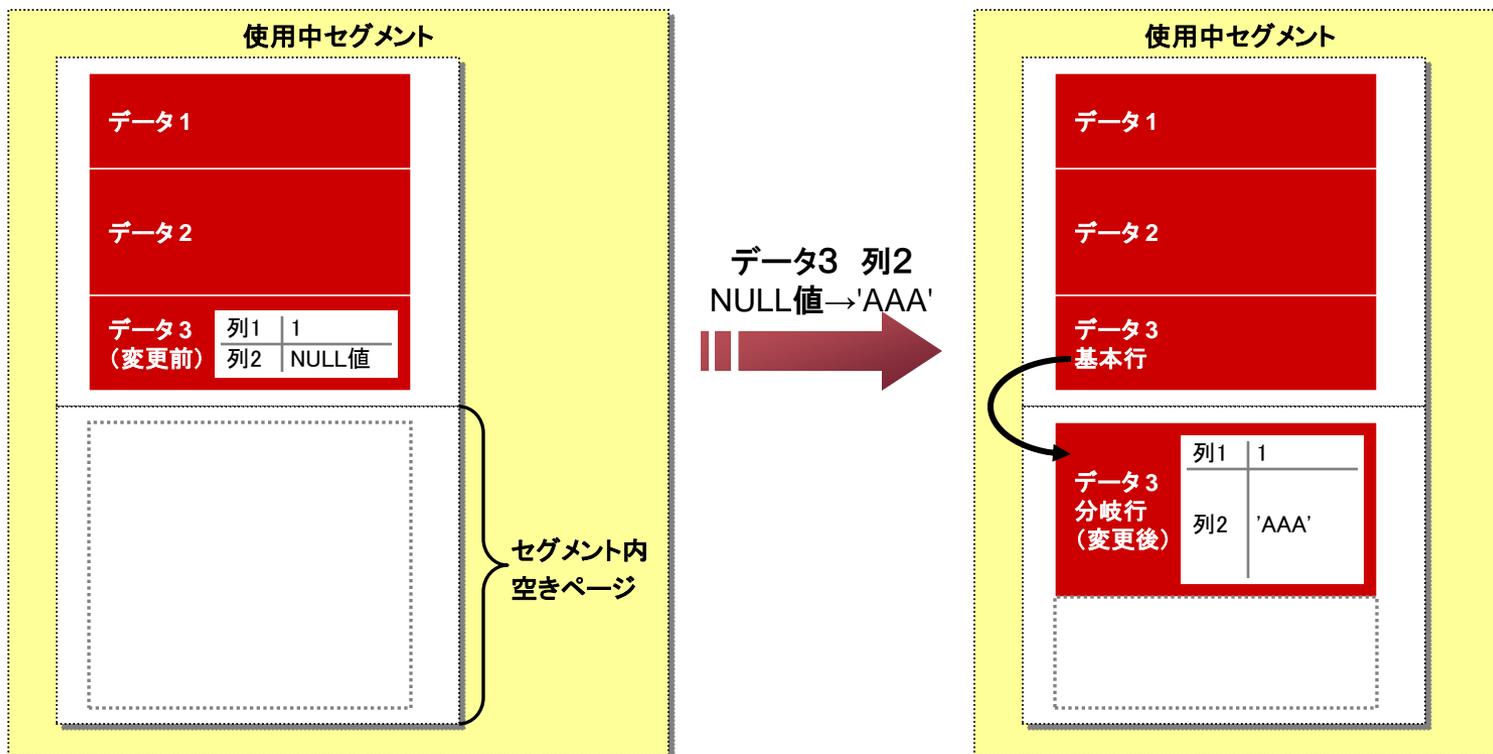
■ : データ格納領域
 : 未使用領域

未使用領域の設定方法については、5章データベースの断片化を低減する運用で解説します。

解説

データの更新を継続して、ページ内未使用領域をすべて使用した場合は、変更データが存在するセグメントの未使用ページに変更後データ(分岐行)を格納します。この場合、セグメント内空きページを設定しておくことで、分岐行を格納するページを確保しやすくなります。

■同一ページ内に格納できない場合



— : データ格納領域
 - - - : 未使用領域

未使用領域の設定方法については、5章データベースの断片化を低減する運用で解説します。

3. HiRDBのデータ格納方法

- 3. 1 表データ
- 3. 2 表データ(可変長長大データ)
- 3. 3 インデクス

解説

可変長の長大データが存在する表の場合は、3.1章で説明した格納方法と異なる点があります。この章では、以下の3種類について説明します。

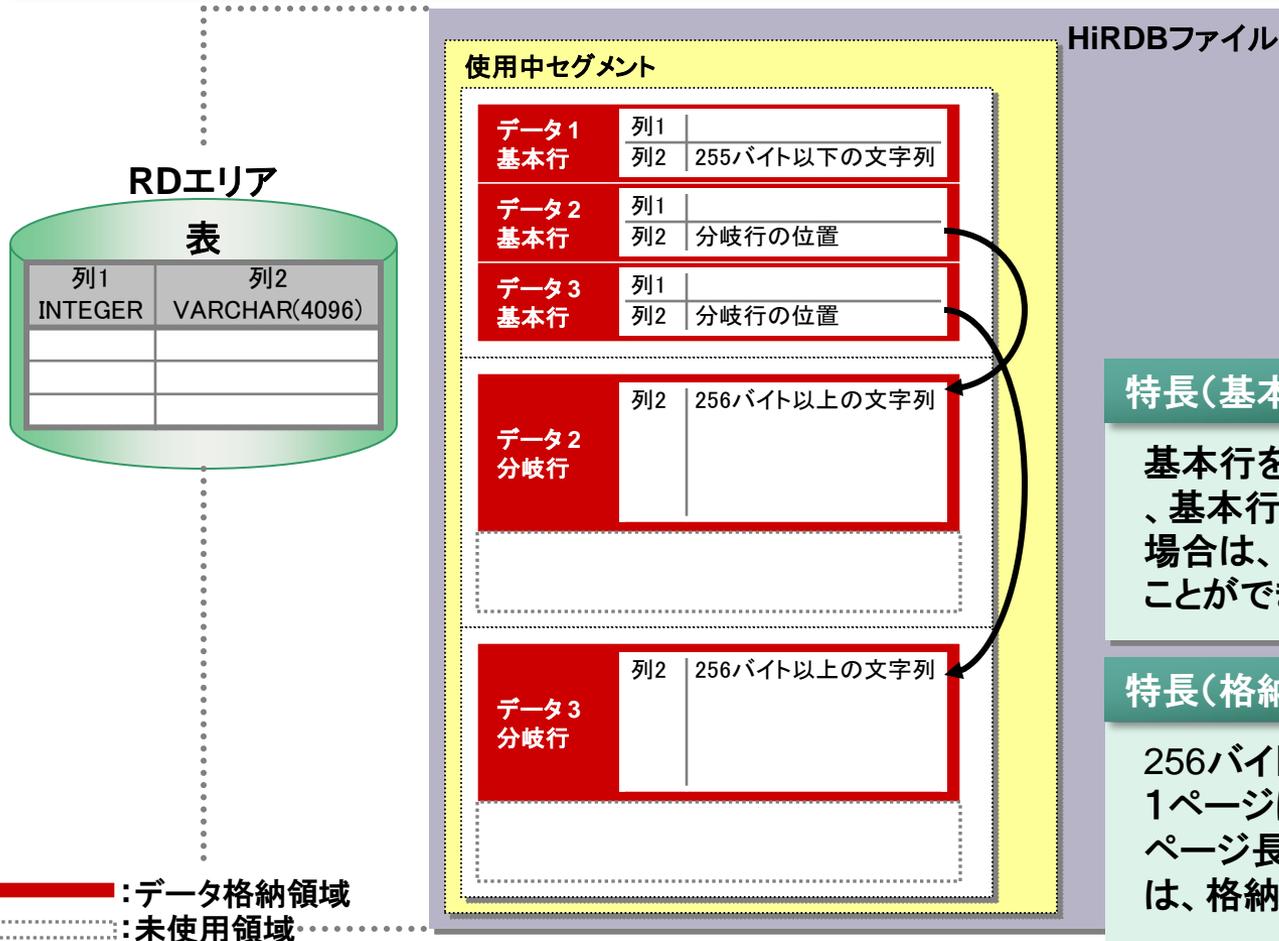
256バイト以上の
VARCHAR、MVARCHAR、NVARCHAR

BLOB

BINARY

解説

可変長文字列(VARCHAR、MVARCHAR、NVARCHAR)に格納するデータの長さが256バイト以上の場合、他の列のデータとは別のページに格納されます。



特長(基本行へのアクセスを優先)

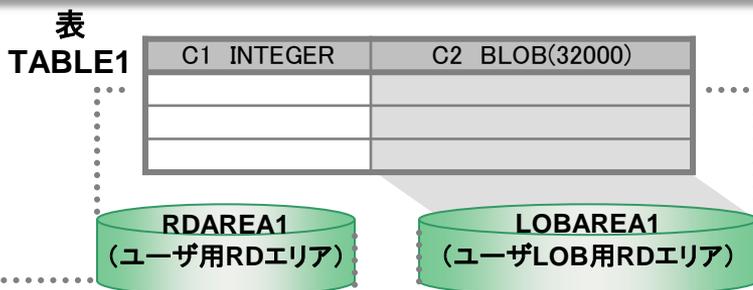
基本行を同じページに格納しているので、基本行部分だけを操作する頻度が高い場合は、アクセスするページを限定することができます。

特長(格納効率について)

256バイト以上の可変長文字列データは1ページに1行しか格納しないため、ページ長に比べてデータ長が短い場合は、格納効率が低下します。

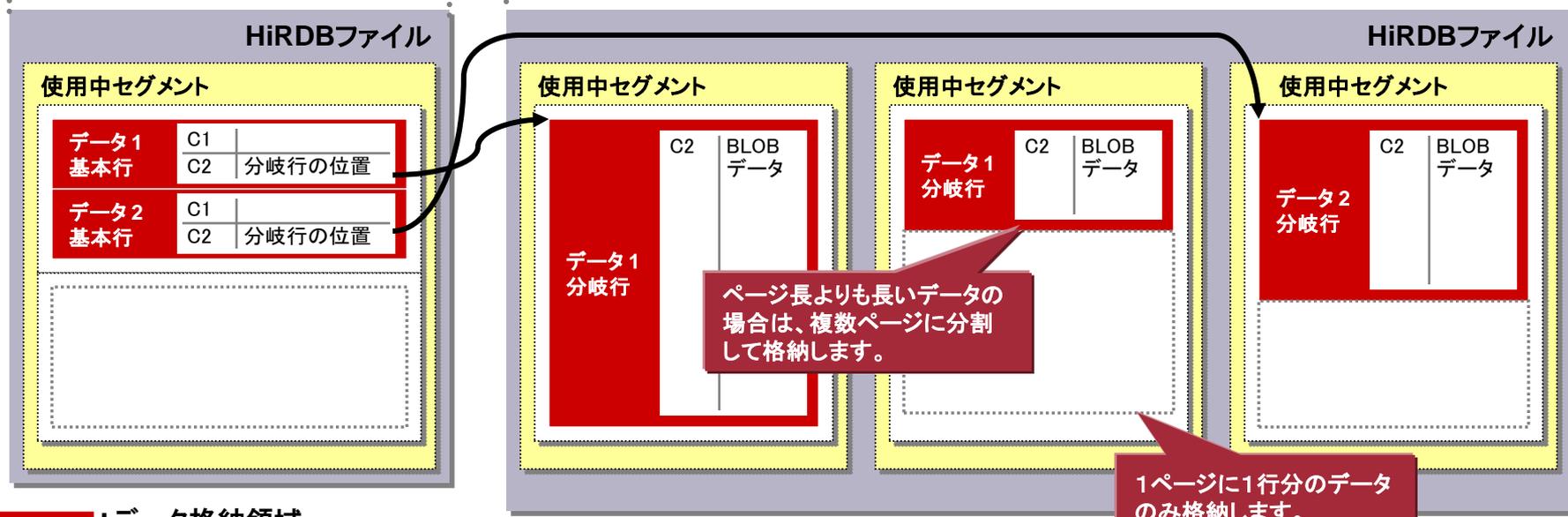
解説

BLOB列のデータは、基本行とは別のRDエリアに格納します。BLOB列を格納するRDエリアは、ユーザLOB用RDエリアという種別になります。ユーザLOB用RDエリアは、ページサイズ(8KB)、セグメントサイズ(1ページ)が固定です。



■BLOB列を含む表定義の例■

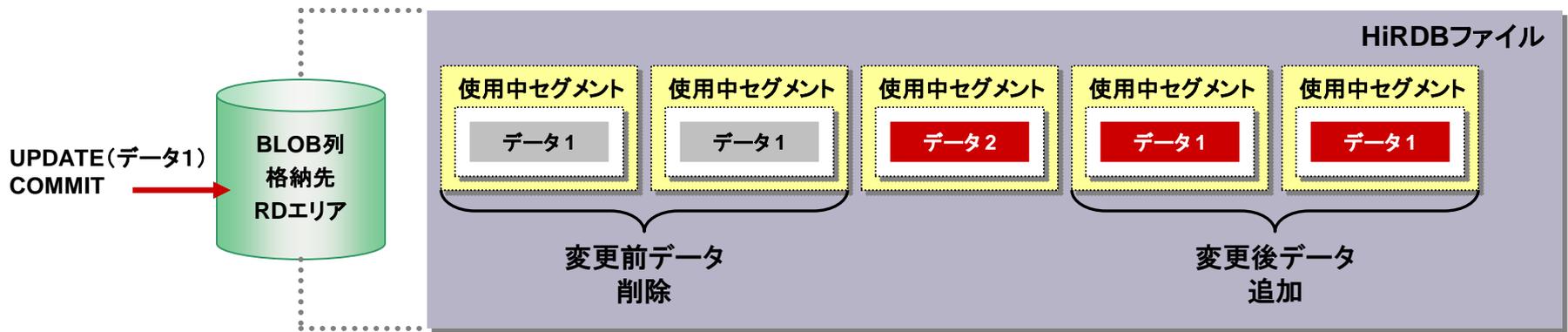
```
CREATE TABLE TABLE1
(C1 INTEGER, C2 BLOB(32000) IN LOBAREA1 )
IN RDAREA1
```



— : データ格納領域
 : 未使用領域

解説

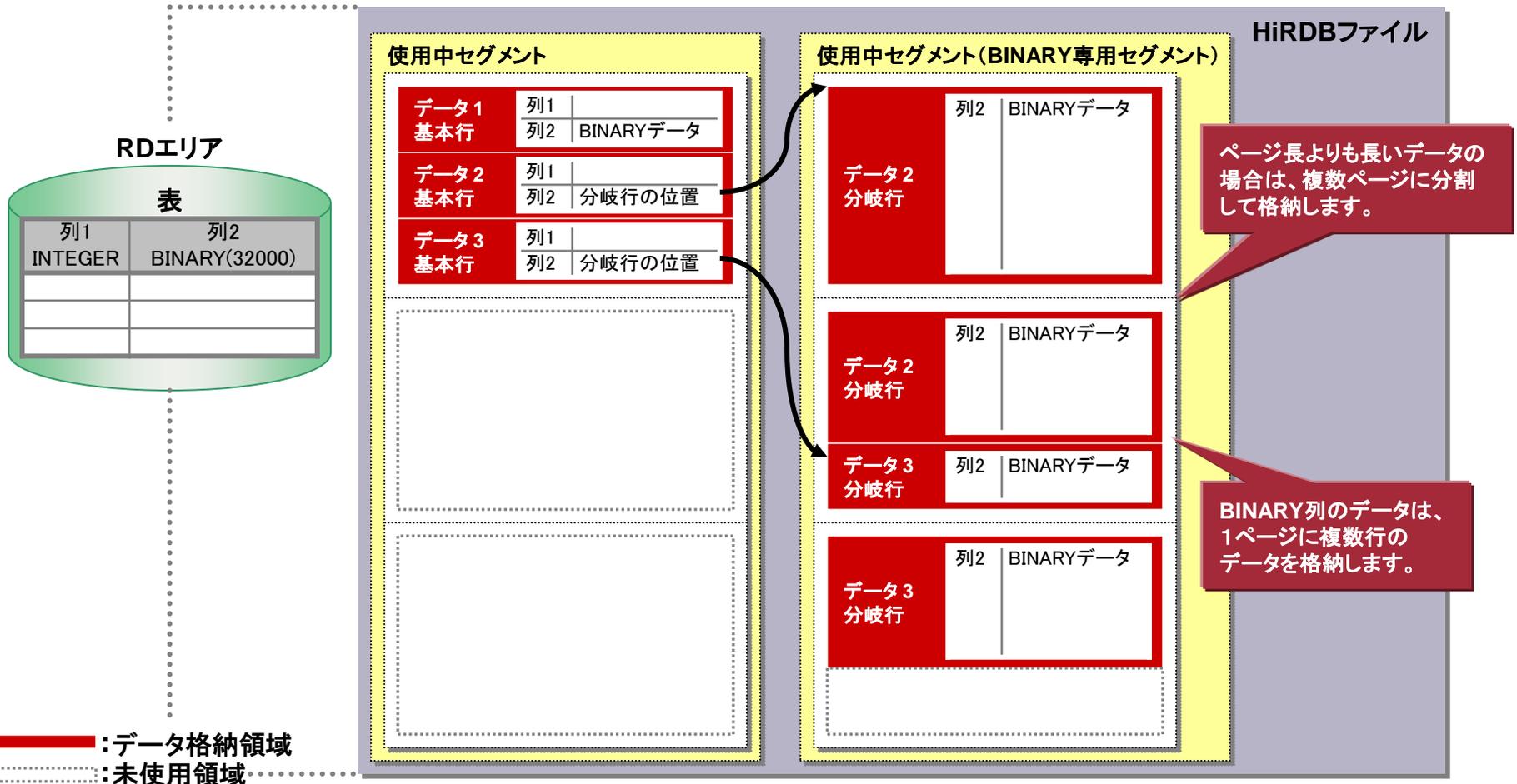
BLOBデータを更新した場合は、変更後のデータ長にかかわらず(データ長が同じ、もしくは短くなる場合でも)、別の領域に変更後データを格納します。



■ : データ格納領域
■ : 空き領域

解説

BINARY列のデータは、表格納先RDエリアと同じRDエリアに格納します。1行のデータ長がページサイズ以下の場合は、1行を1ページに格納します。1行のデータ長がページ長より大きい場合は、BINARY列のデータはBINARY専用セグメントのページに格納します。



3-2-6 BINARY専用セグメントの効果

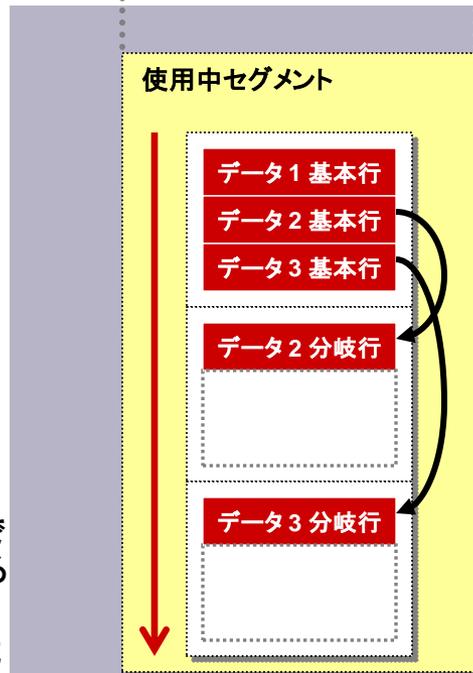
解説

1ページより長いBINARY列のデータは、BINARY専用セグメントに格納されます。
BINARY列へアクセスしないテーブルスキャンを行う場合に、BINARY専用セグメントのページは読み込む対象としません。

■テーブルスキャン時のスキャン対象ページ

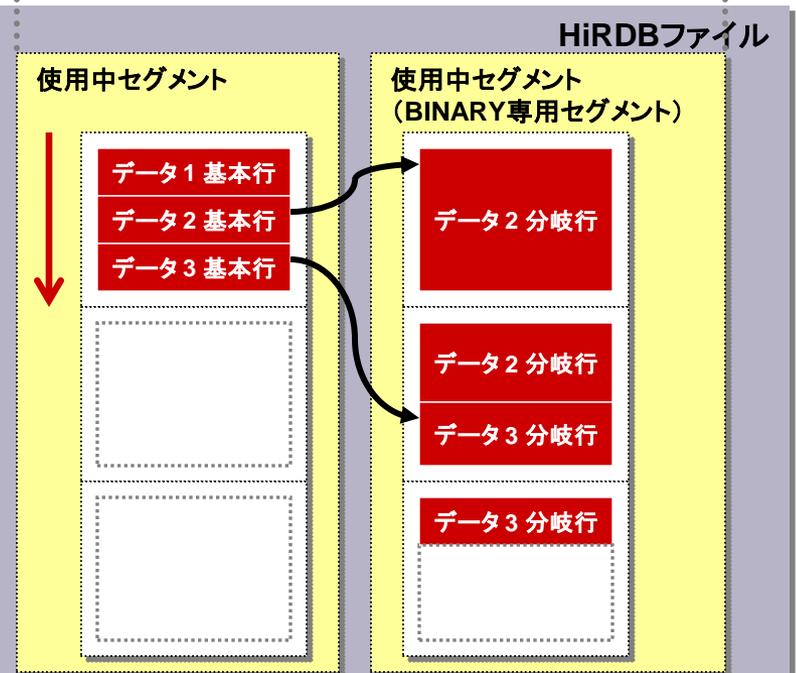
◆VARCHAR(分岐行あり)の場合

列1	列2
INTEGER	VARCHAR(4096)



◆BINARY専用セグメントの場合

列1	列2
INTEGER	BINARY(32000)



→ : テーブルスキャンで
スキャン対象となる
ページ

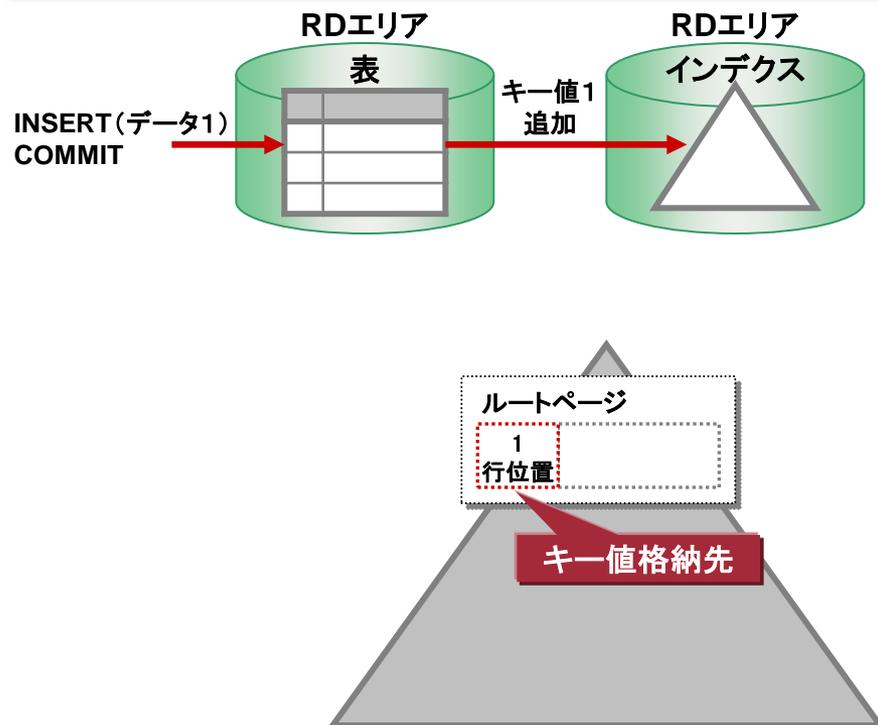
— : データ格納領域
⋯ : 未使用領域

3. HiRDBのデータ格納方法

- 3. 1 表データ
- 3. 2 表データ(可変長長大データ)
- 3. 3 **インデクス**

解説

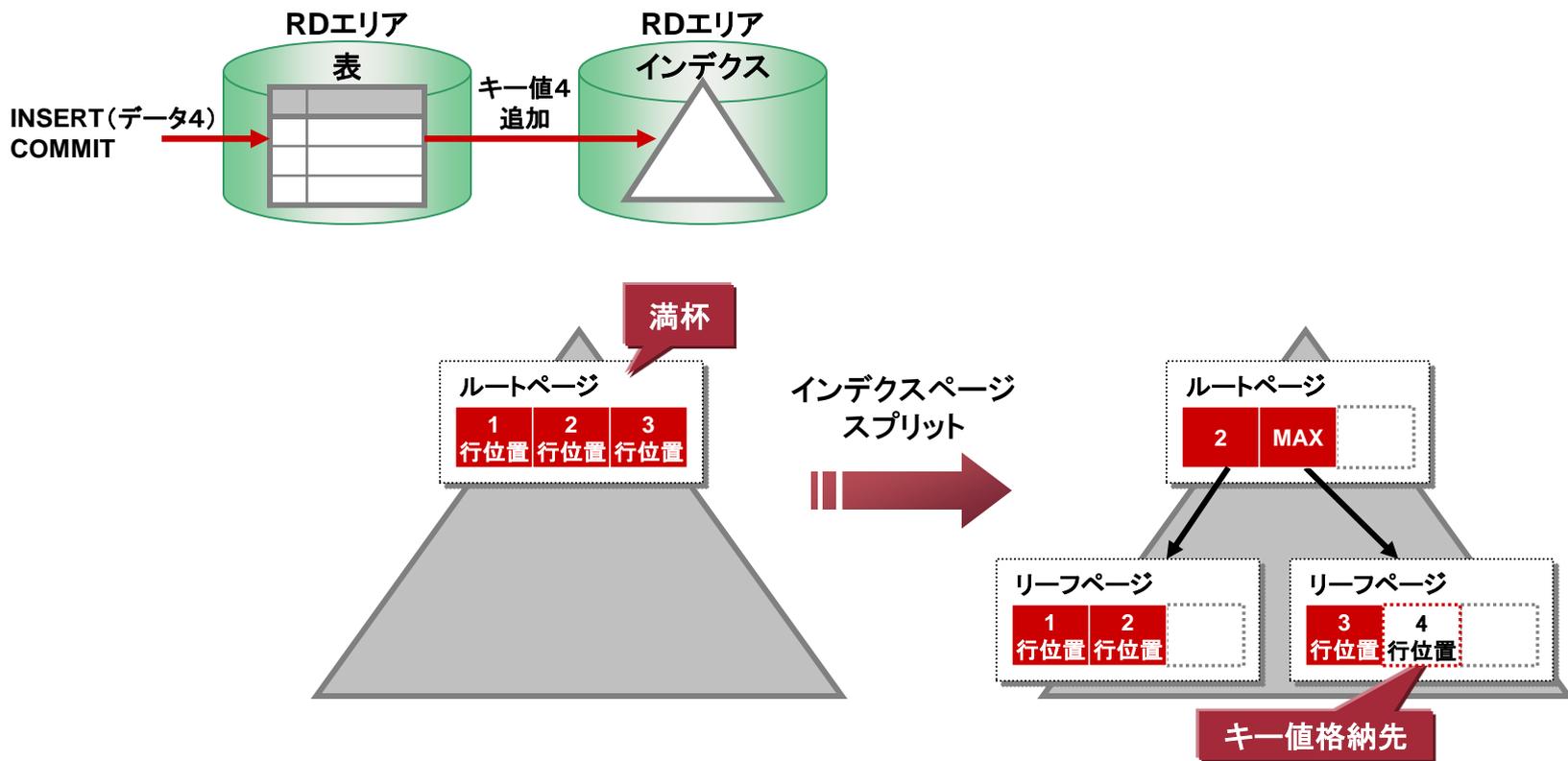
インデクスの場合、インデクス定義時にルートページを確保します。
1行目のキー値は、ルートページに格納します。



.....: 未使用領域

3-3-2 インデクスの格納方法②

解説 2行目以降のキー値も、ルートページに格納します。ルートページに格納できる領域がなくなった場合は、データを2つに分割し、それぞれリーフページに格納します。この動作をインデクスページスプリットと呼びます。リーフページは、ルートページからポイントします。

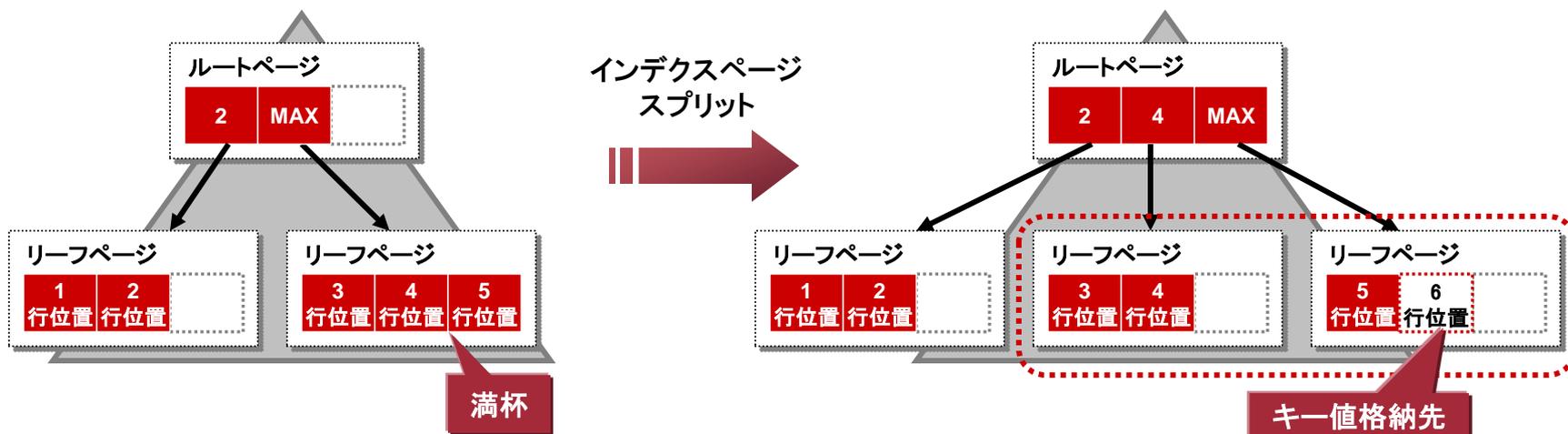
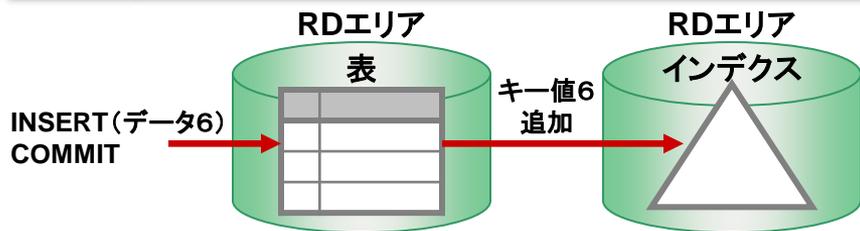


■ : キー値格納領域
□ : 未使用領域

3-3-3 インデクスの格納方法③

解説

キー値を格納するリーフページに領域がなくなった場合、同様にインデクスページスプリットを行います。

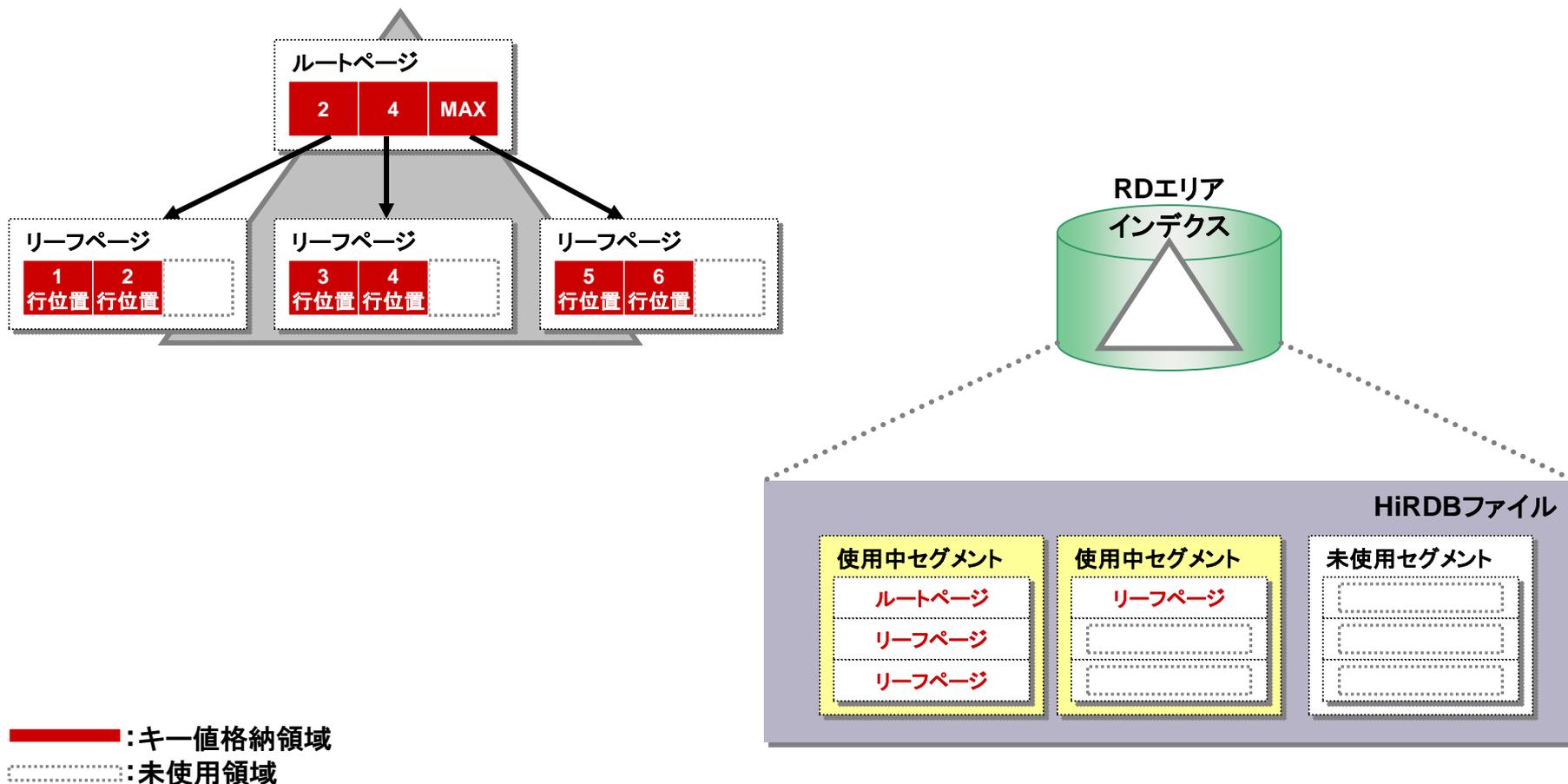


■ : キー値格納領域
□ : 未使用領域

3-3-4 インデクスページの割り当て

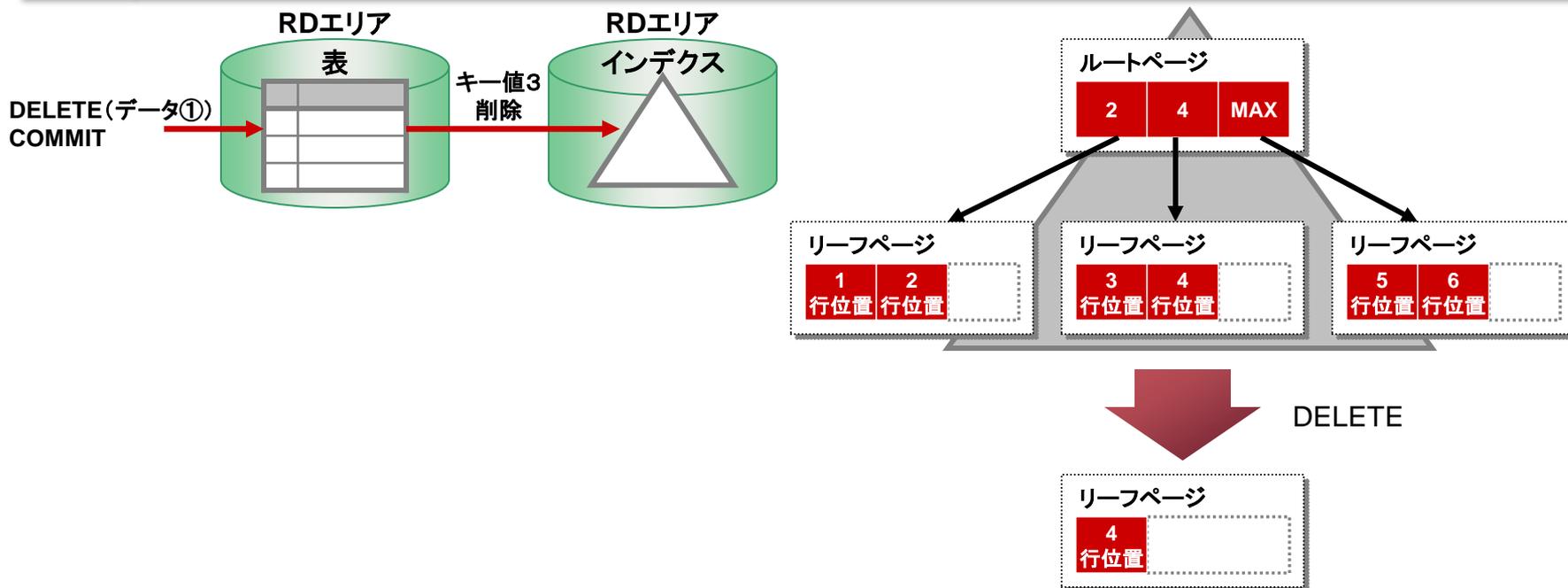
解説

インデクスページは、ルートページが存在するセグメントの未使用ページから割り当めます。ルートページが存在するセグメントに未使用ページがなくなった場合は、新たに未使用セグメントを確保します。



解説

データを削除すると、そのデータを指すキー値も削除されます。通常のインデクスの場合、キー値が削除されると、格納していた領域からそのキー値は削除されます。

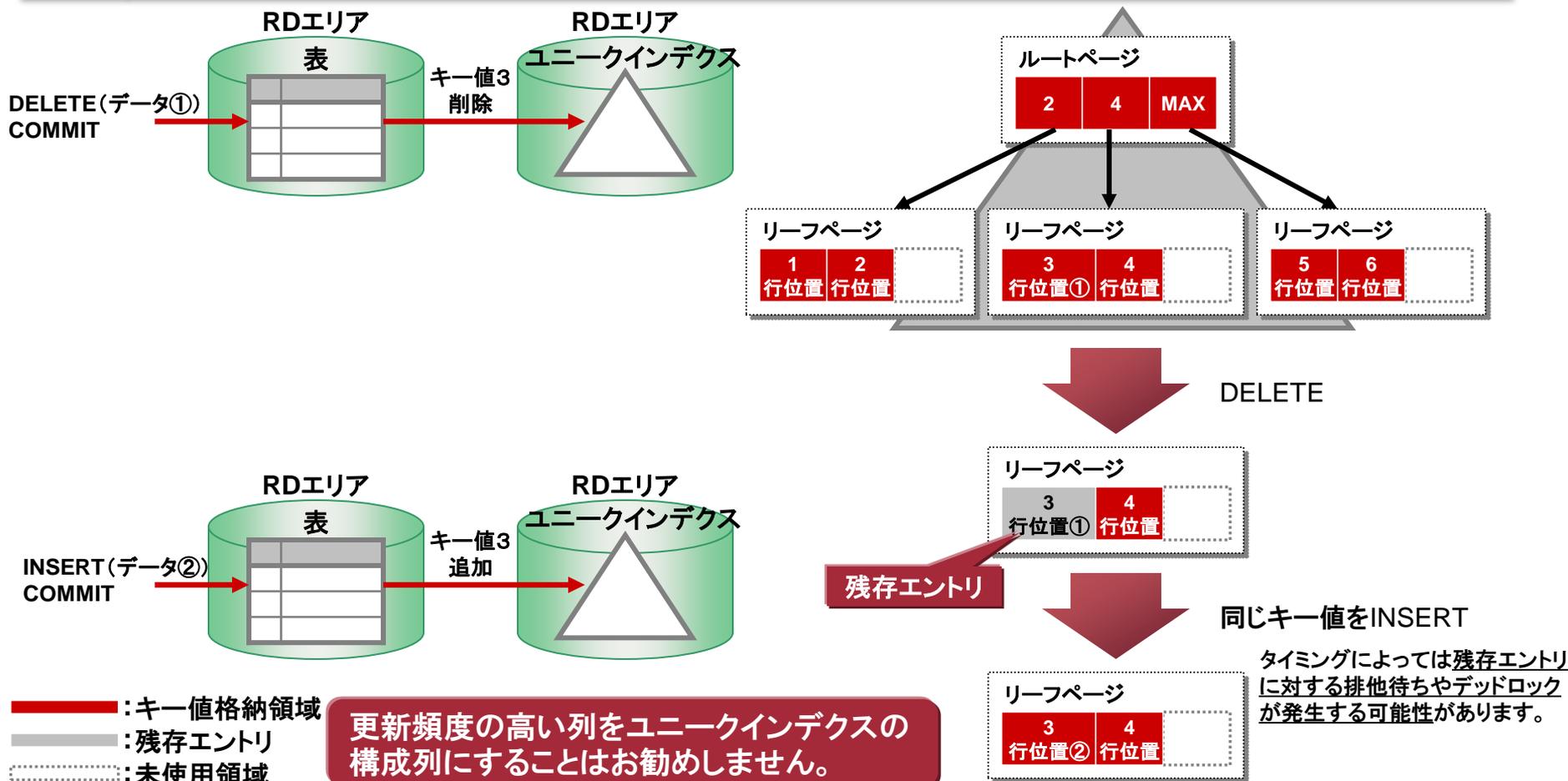


■ : キー値格納領域
 □ : 未使用領域

3-3-6 キー値の削除(ユニークインデクス)

解説

ユニークインデクスの場合、データを削除しても、そのデータを指すキー値の領域は残存エントリとして残ります。同じキー値を持つデータが再び追加された場合は、その残存エントリが利用されますが、別のキー値の格納先としては利用されないことがあります。



4. データベースの断片化を監視する運用

解説

データベースの断片化を監視して、問題が起きる前に対処して、データベースを安定稼働させます。本章では、それぞれの運用について解説します。

メッセージの監視

データベースの状態解析

再編成時期予測

解説

データベースの断片化により格納効率やアクセス効率が低下すると、HiRDBIはsyslog (Windowsの場合はイベントログ)に、KFPH00211-IやKFPH00212-I、KFPH22017-Iのメッセージを出力します。これらを監視することで、素早い対処が可能となります。

KFPH00211-I ~RDエリアの使用率通知メッセージ

RDエリアの使用率が特定の値に達した時点で出力されます。デフォルトの設定では、メッセージ出力契機となる使用率は、80%、90%、100%の3つです。出力契機となる使用率を変更したい場合は、システム共通定義pd_rdarea_warning_pointオペランドに、出力契機となる使用率を指定してください。

データベース状態解析ユティリティでデータベースの状態を確認して、RDエリアの拡張や再編成、空きページ・セグメント解放の対処を行います。

KFPH00212-I ~表の格納効率やアクセス効率の低下メッセージ

RDエリアや表の格納効率やアクセス効率が低下すると出力されます。

データベース状態解析ユティリティでデータベースの状態を確認して、RDエリアの拡張や再編成、空きページ・セグメント解放の対処を行います。

KFPH22017-I ~BLOBデータの格納効率やアクセス効率の低下メッセージ

BLOBデータの格納効率やアクセス効率が低下すると出力されます。

データベース状態解析ユティリティでデータベースの状態を確認して、RDエリアの再編成の対処を行います。

4-3 データベースの状態解析 セグメント使用率の確認

解説

データの追加を継続して行くと、使用中セグメントの数が増加していきます。
RDエリア内のセグメント使用率を確認する方法を解説します。
データの削除を行わないRDエリアは、セグメント使用率が高いと容量不足となっています。



- : データ格納領域
- : 空き領域
- : 未使用領域

■ RDエリアの使用率
使用中セグメント数(4) / 全セグメント数(5)
= 80%

RDエリアの状態表示コマンド(pddb1s)

```
% pddb1s -r RDAREA1 -a
STATE OF RDAREA
RDAREA      ID      STATUS      TYPE
RDAREA1     4      OPNMODE    USER
RDAREA1     4      OPEN
RDAREA1     4      INITIAL
SEGMENT     75 / 700
```

未使用セグメント数 / 全セグメント数

セグメント使用率

データベース状態解析ユーティリティ(pddb1st)

```
% pddb1st -r RDDATA10 -k phys
: (中略)
-----
RD Area Name   : RDDATA10
Server        : sds01
History1 Hold Status :      Hold Code :      0 Hold Time :
History2 Hold Status :      Hold Code :      0 Hold Time :

Unused Segment:      2096
Segment Size       :      10 Pages Page Size       :      4096 Bytes
Used (Full)       :
Segment           :      1% ( 0%)      4 ( 0)      2100
Page              :      1% ( 0%)      8 ( 0)      21000
```

解説

ページ使用率からRDエリアの使用中ページの比率を確認できますが(①)、使用中空きページを含むために、格納効率が低下しているか、データ量に応じた使用率なのか判断できません。その場合は、ページの詳細情報(②)で確認します。

■ ページの詳細情報を含む状態解析結果の例(典型的な例です)

```
% pddbst -r user_rdarea_1 -k logi -d
: (中略)
```

```
Table Name : TBL01
Auth Id    : user1
Status     :
Reference Pending Status :
Check      Pending Status :
Segment Reuse : - segments
<Base row segment>
Search Mode : INS
Reuse Search Failure : 0/ 0
Used (Full) Used ( Full) Sum
Segment 100% ( 0%) 107 ( 0) 107
Page 100% ( 0%) 535 ( 0) 535
Collect On Segment : 0
Collect Prearranged Page : 0
Used Page Ratio Page (Ratio)
0% : 0 ( 0%)
1- 10% : 1 ( 1%)
11- 20% : 0 ( 0%)
21- 30% : 0 ( 0%)
31- 40% : 0 ( 0%)
41- 50% : 0 ( 0%)
51- 60% : 0 ( 0%)
61- 70% : 118 ( 22%)
71- 80% : 416 ( 78%)
81- 90% : 0 ( 0%)
91-100% : 0 ( 0%)
Total 535
```

使用中セグメント
+ 使用中空きセグメント
+ 満杯セグメント

使用中ページ
+ 使用中空きページ
+ 使用中満杯ページ

使用中空きページ

① 使用中空きページが含まれるため。格納効率が低下しているか、データ量に応じた使用量かわからない。

② 使用中空きページが多く、(使用中空きページ+未使用ページ)の比率が高い場合は、ページの格納効率が低下している。

使用中空きページ
+ 未使用ページ

② PCTFREE値を超えるこの辺の比率が高い場合は、格納効率は高く、データ量の増加により容量が不足となる可能性がある。

ページの
詳細情報

解説

データベース状態解析ユーティリティ(pddbst)では、表格納先RDエリアの情報と、BLOB列格納先RDエリアの情報がそれぞれ表示されます。

■ BLOB列を含む表の状態解析結果例

Table Name : TABLE2
Auth Id : USER1
Hash :
Total Segment : 7 Max Sum Segment : 6
Total Page : 16 Max Sum Page : 10
RD Area Count : 2/ 2

RD Area Name : RDAREA01
Server : sds01
Status :
Reference Pending Status :
Check Pending Status :
History1 Hold Status : Hold Code : 0 Hold Time :
History2 Hold Status : Hold Code : 0 Hold Time :
Job Name : Line Count : Index Method :
Unused Segment: 45
Segment Reuse : - segments
<Base row segment>
Search Mode : INS
Reuse Search Failure : 0/ 0
Used(Full) Used(Full) Sum
Segment 100%(0%) 1(0) 1
Page 10%(0%) 1(0) 10

表全体の情報

表格納先RDエリアの情報

【右に続く】

RD Area Name : LOBAREA01
Server : sds01
Status :
Column Name : C2
History1 Hold Status : Hold Code : 0 Hold Time :
History2 Hold Status : Hold Code : 0 Hold Time :
Unused Segment: 9994(9994)
Used Used Sum
Segment 100% 6 6
Segment Over :N Last Segment : 6/ 10000
Lobmap Over :N

BLOB列格納先RDエリアの情報

RDエリアの格納状態を示しています。
この値が'Y'になると、データベース再編成が必要です。

使用中セグメントの最終位置／総セグメント数
を示しています。
最終位置が、総セグメント数に近い値になった
場合は、データベース再編成が必要です。

解説

データベース状態解析ユーティリティ(pddbst)では、基本行を格納するセグメントとBINARY専用セグメントの使用率を区別して確認することができます。
(HiRDB Version 8 08-04以降)

■BINARY専用セグメントが存在する表の状態解析結果例

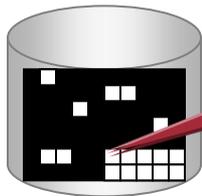
Table Name	: TABLE1				
Auth Id	: USER1				
Hash	:				
Total Segment	: 3	Max Sum Segment	: 3		} 表全体の使用率
Total Page	: 30	Max Sum Page	: 30		
RD Area Count	: 1/	1			

RD Area Name	: RDAREA01				} RDエリアごとの情報
Server	: sds01				
Status	:				
Reference Pending Status	:				
Check Pending Status	:				
History1 Hold Status	: Hold Code : 0	Hold Time :			
History2 Hold Status	: Hold Code : 0	Hold Time :			
Job Name	: Line Count :	Index Method :			
Unused Segment	: 46				
Segment Reuse	: - segments				
<Base row segment>					
Search Mode	: INS				
Reuse Search Failure	: 0/ 0				
	Used(Full)	Used(Full)	Sum		
Segment	100%(0%)	1(0)	1	} 基本行を格納するセグメントの使用率	
Page	10%(0%)	1(0)	10		
<BINARY segment>					
Search Mode	: INS				
Reuse Search Failure	: 0/ 0				
	Used(Full)	Used(Full)	Sum		
Segment	100%(0%)	2(0)	2	} BINARY専用セグメントの使用率	
Page	60%(50%)	12(10)	20		

解説

再編成時期予測機能を利用することで、RDエリアの容量不足に対するメンテナンス方法やメンテナンス時期の通知、表やインデクスの格納効率やアクセス効率の低下に対するメンテナンス方法やメンテナンス時期を通知できます。

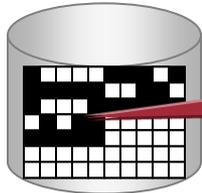
① RDエリアの容量不足の防止



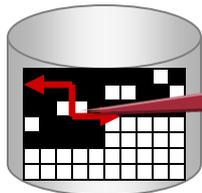
容量が満杯

予測レベル1

② 格納効率の低下の防止

無駄な空き領域が
多い

③ アクセス効率の低下の防止

データの格納状態が
乱れている

予測レベル2

メンテナンス情報例

% pddbst -k pred -e 2			
pddbst VV-RR **** Analysis Item Data *****			
No	: 1		
Date	: 2005/05/25		
Method	: ReclaimS		
Type	: T		
Name	: "k1234567"."table01" ("RDUSER01")		
StateDate	: 2005/05/19 00:10:32		

Information			Date

Empty Page Ratio			2005/05/25
Unused Page Ratio (50)			
Number of Branch Row		(50)	

No	: 2		
Date	: 2005/05/25		
Method	: Reorganize		
Type	: T*		
Name	: "k1234567"."table02" ("RDUSER01")		
StateDate	: 2005/05/19 00:10:32		

Information	Value	Method	Date

Empty Page Ratio	0 (30)		
Unused Page Ratio (50)	100 (50) * Reorganize		2005/05/25
Number of Branch Row	0 (50)		

メンテナンス予定日の
早い順に表示メンテナンス方法を
表示

解説

再編成時期予測を行うためには、予測データを蓄積する表の格納RDエリアを準備し、予測データの蓄積と、予測データの解析を以下の手順で行います。

1. 準備

- ・解析情報表および運用履歴表を格納するデータディクショナリ用RDエリアの作成

```
$ pdmod -a 制御文ファイル
```

```
制御文ファイル: create rdarea rddb for datadictionary of dbmanagement ~
```

- ・システム共通定義のpd_rorg_predictオペランドにYを指定
- ・予測データ更新のためのシステムログファイルの容量見積もり

2. 予測データ蓄積

- ・データベースを解析して、再編成時期の予測データを解析情報表に蓄積

```
$ pddbst -k logi -r ALL -e 1 (予測レベル1の実行例)
```

※予測の精度を高めるために、定期的に(毎日)実行してください。

3. 予測データ解析

- ・予測データを解析して、DBメンテナンス予定日が近いRDエリアと予測日を取得

```
$ pddbst -k pred -r ALL -e 1 (予測レベル1の実行例)
```

※DBメンテナンス時期の見落としがないように、定期的に実行してください。

4. メンテナンス方法解析

- ・DBメンテナンス予定日が近いRDエリアがある場合、更にメンテナンスが必要な表、インデクス、およびRDエリアと、そのメンテナンス方法を調査します。

```
$ pddbst -k pred -r ALL -e 1 -m (予測レベル1の実行例)
```

5. メンテナンス

表示されたメンテナンス方法に従って、表を再編成、インデクスを再編成、RDエリアを拡張。RDエリアを再初期化します。

※予測レベル2は解析に時間を要するので、負荷を分散するために、インターバル解析、マージ解析の方法があります。詳細はマニュアルをご参照。



```
pddbst -k logi
```

```
SQL or コマンド
```

蓄積



データディクショナリ用
RDエリア

解析

```
pddbst -k pred
```

```
pddbst -k pred -m
```

5. データベースの断片化を低減する運用

解説

データベースの断片化を低減して、断片化の影響を抑えて、データベースを安定稼働させます。本章では、それぞれの運用について解説します。

未使用領域比率

ノースプリットオプション

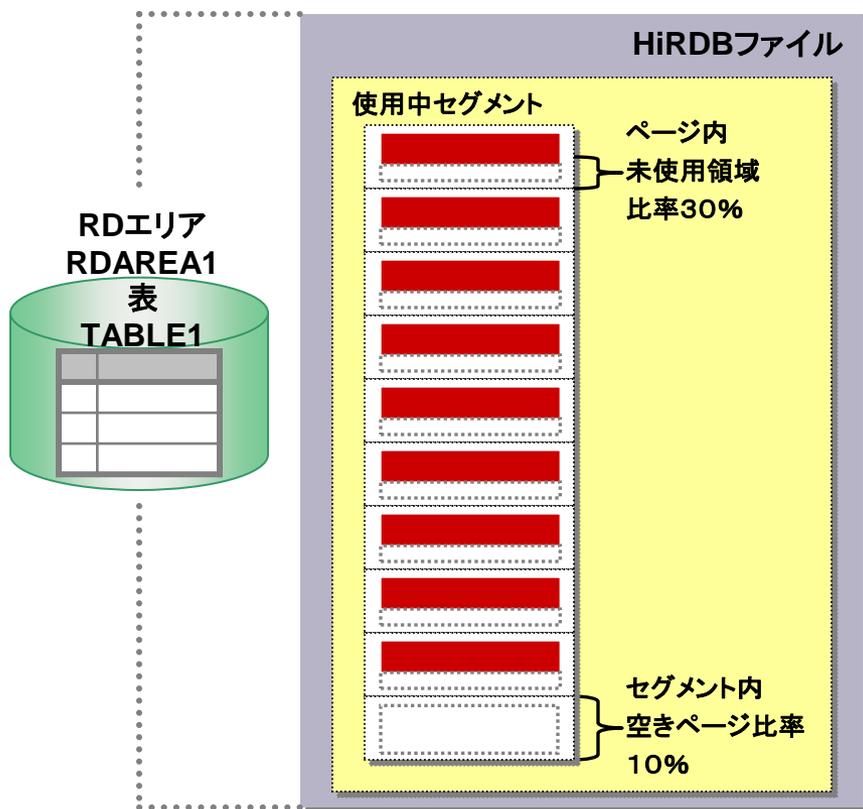
アンバランスインデクススプリット

空き領域の再利用

RDエリアの循環利用

解説

ページ内の未使用領域の比率およびセグメント内の空きページ比率を設定することで、データの長くなる更新により、別ページや別セグメントに分岐するのを低減できます。CREATE TABLE文の表オプションPCTFREEで設定します。PCTFREEの設定は、データロードでデータを格納する場合とデータベース再編成を実行した場合に、有効になります。



■ : データ格納領域
□ : 未使用領域

■ PCTFREE 指定例 ■

```
CREATE TABLE TABLE1  
( C1 INTEGER , C2 CHAR(100) )  
IN RDAREA1  
PCTFREE=(30,10)
```

PCTFREEのデフォルト値

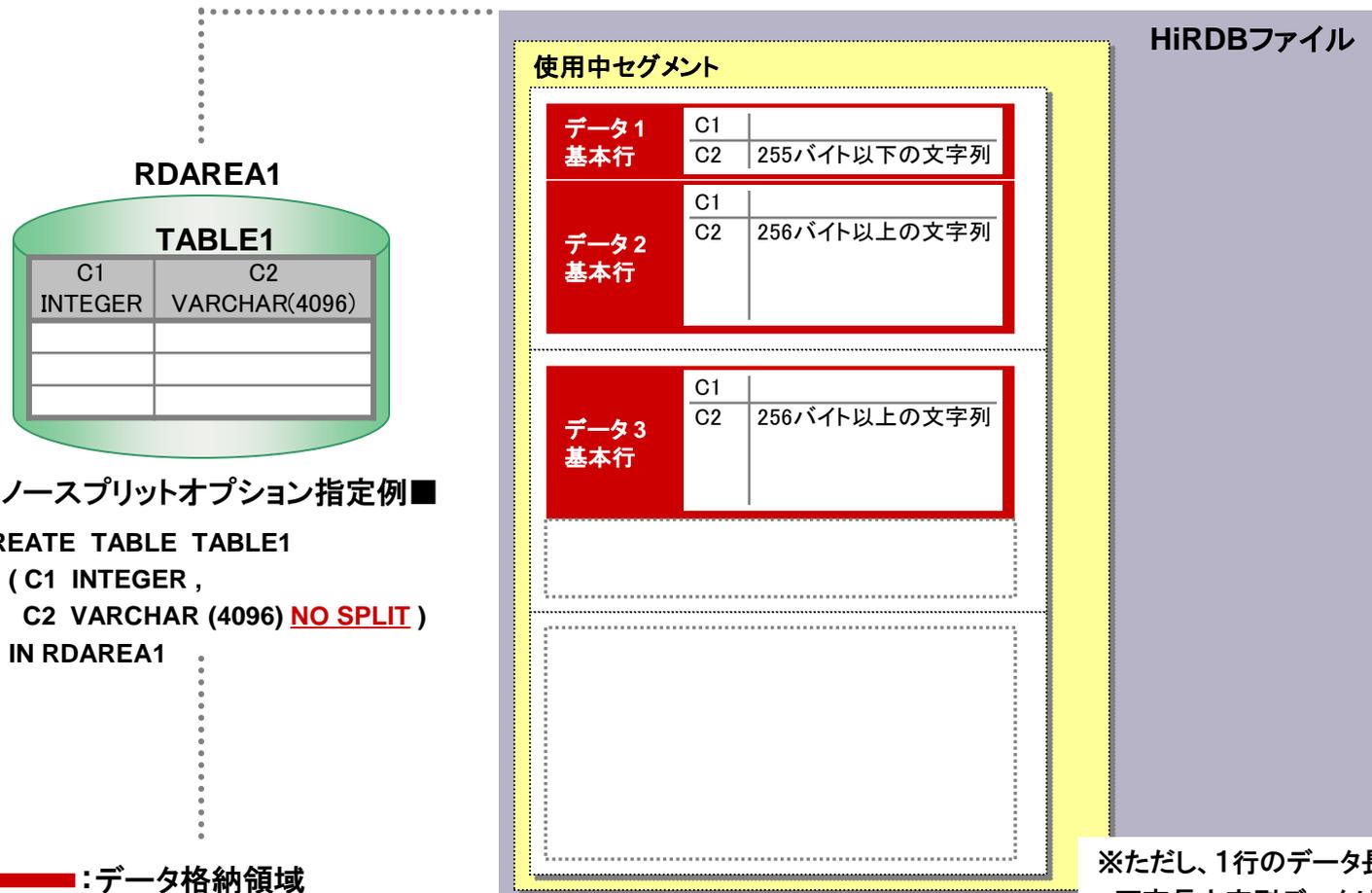
PCTFREEのデフォルト値は、以下の通りです。
ページ内未使用領域比率:30%
セグメント内空きページ比率:10%
上記の指定例は、デフォルトの設定と同じになります。

FIX表の場合

FIX表の場合は、行長が固定であるため、ページ内未使用領域およびセグメント内空きページは不要です。よって、PCTFREEの設定は、両方とも0%を指定してください。

解説

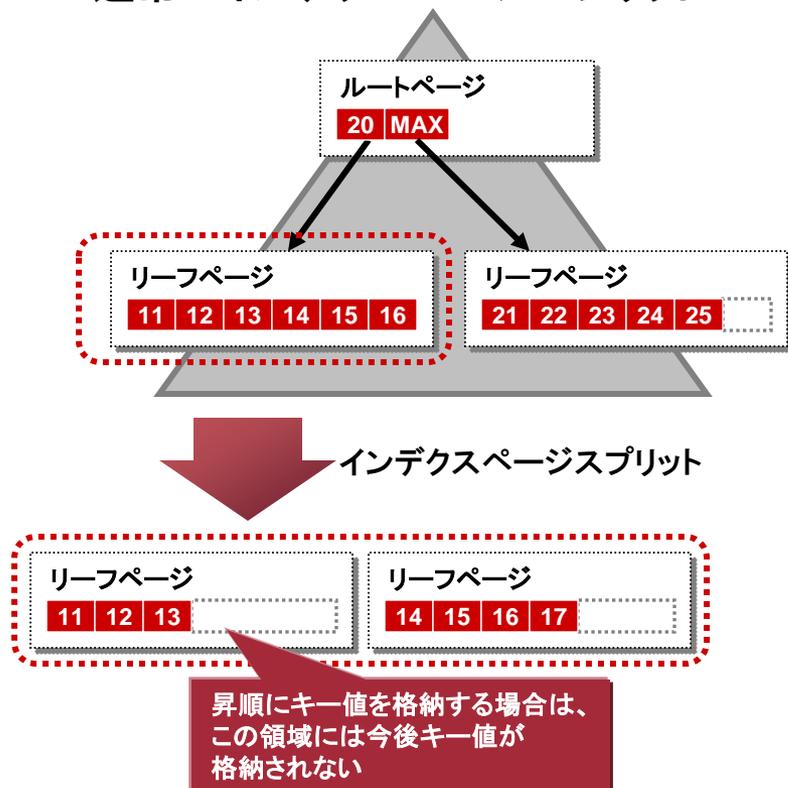
列定義にノースプリットオプションを指定すると、256バイト以上の可変長文字列データの場合でも、1行を1ページに格納します。これにより、格納効率を向上させることができます。



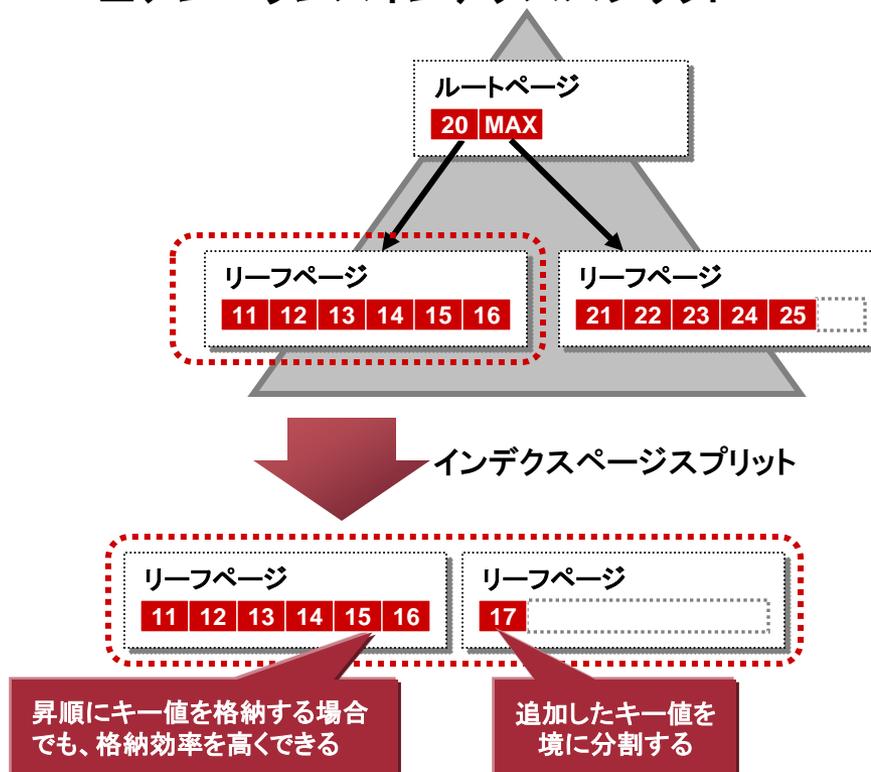
※ただし、1行のデータ長がページ長を超える場合は、可変長文字列データは、別のページに格納します。

解説 通常のインデクスページスプリットでは、キー値を半分ずつに分割して格納します。ただし、連続したキー値を格納する場合は、前半部分のページにその後キー値が追加されないため、格納効率が下がります。これはアンバランスインデクススプリットにより解消できます。

■通常のインデクスページスプリット



■アンバランスインデクススプリット

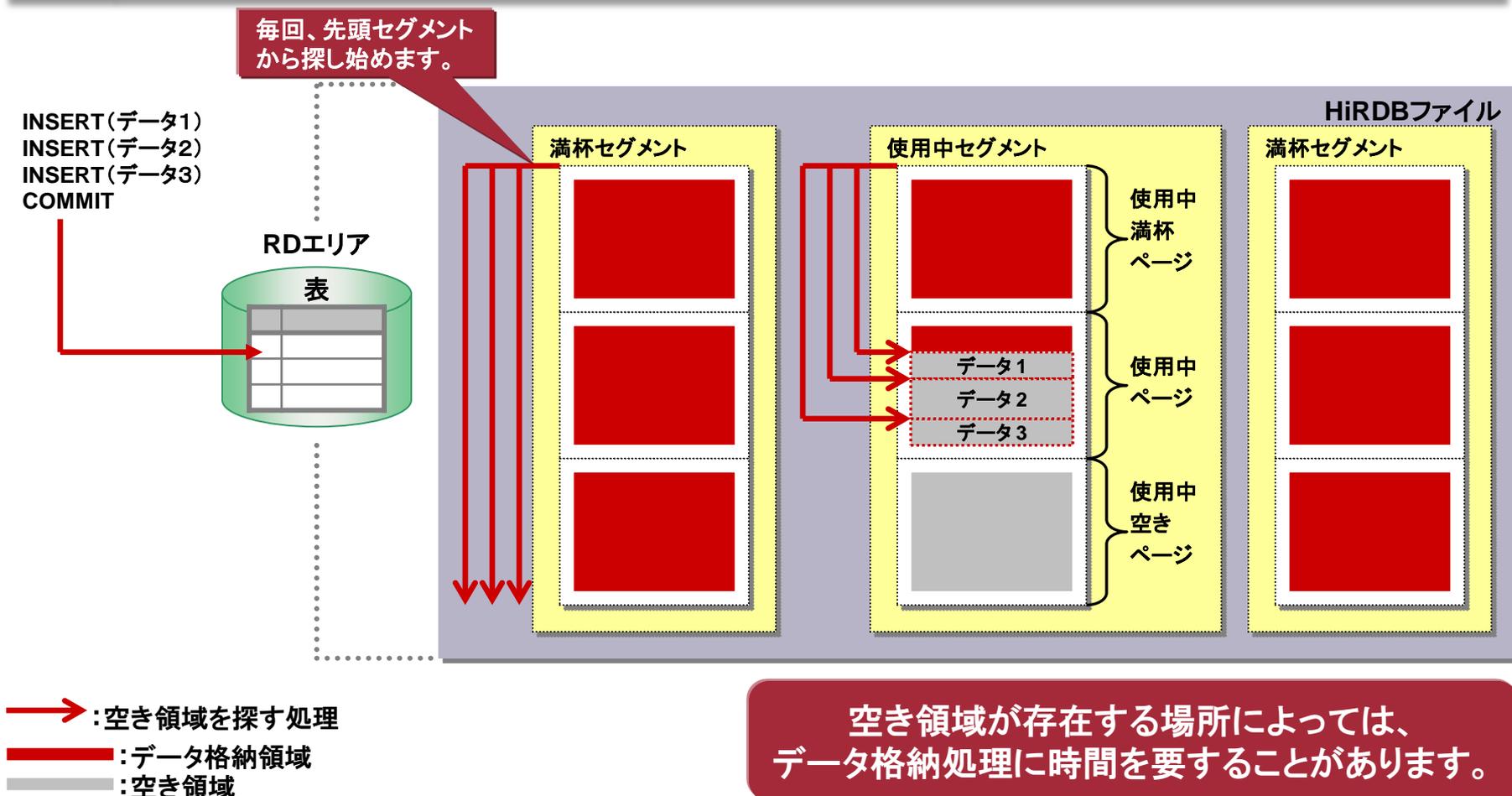


アンバランスインデクススプリットを使用する場合、CREATE INDEX、またはCREATE TABLEのインデクスオプションに、UNBALANCED SPLITを指定します。

5-5-1 空き領域の再利用 セグメント満杯時の通常の空き領域の探索

解説

RDエリアの最終セグメントが満杯になった場合、先頭の使用セグメントの先頭ページから空き領域を探します。基本の動作では、データ格納のたびに、先頭の使用セグメントの先頭ページから探す処理を開始します。

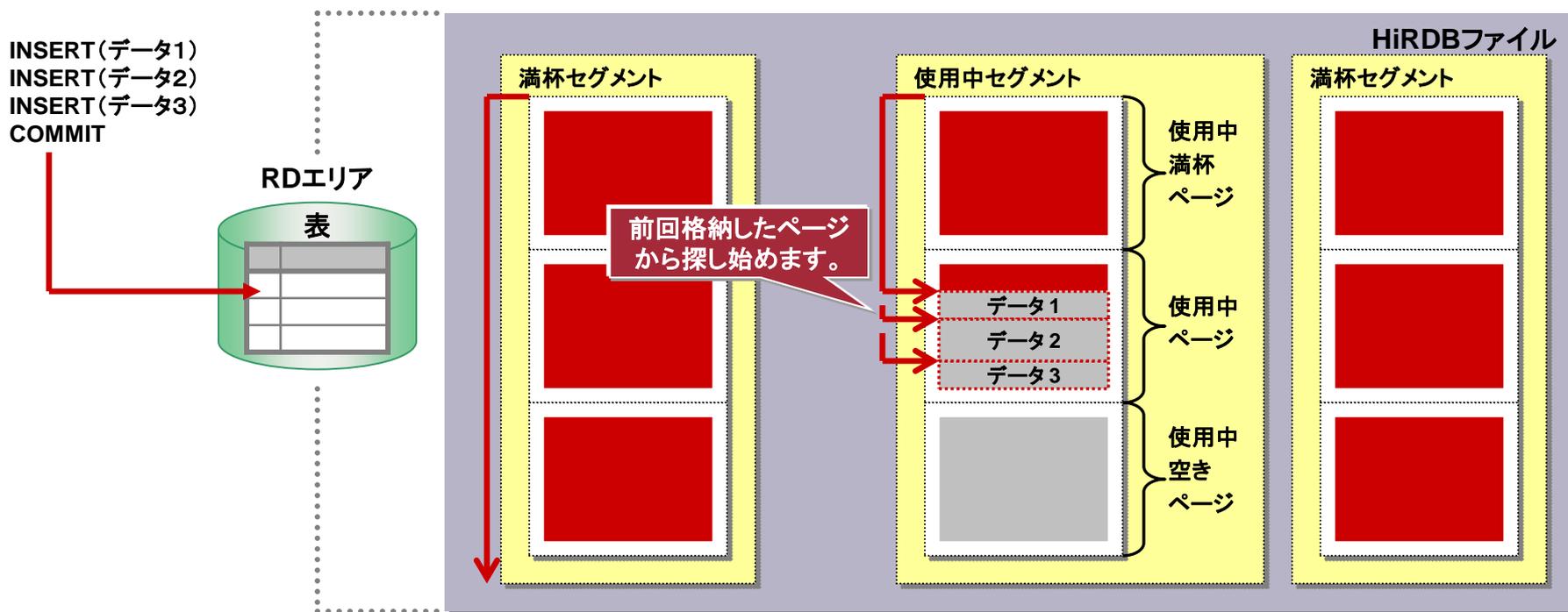


空き領域が存在する場所によっては、データ格納処理に時間を要することがあります。

5-5-2 空き領域の再利用 空き領域の再利用時の空き領域探索

解説

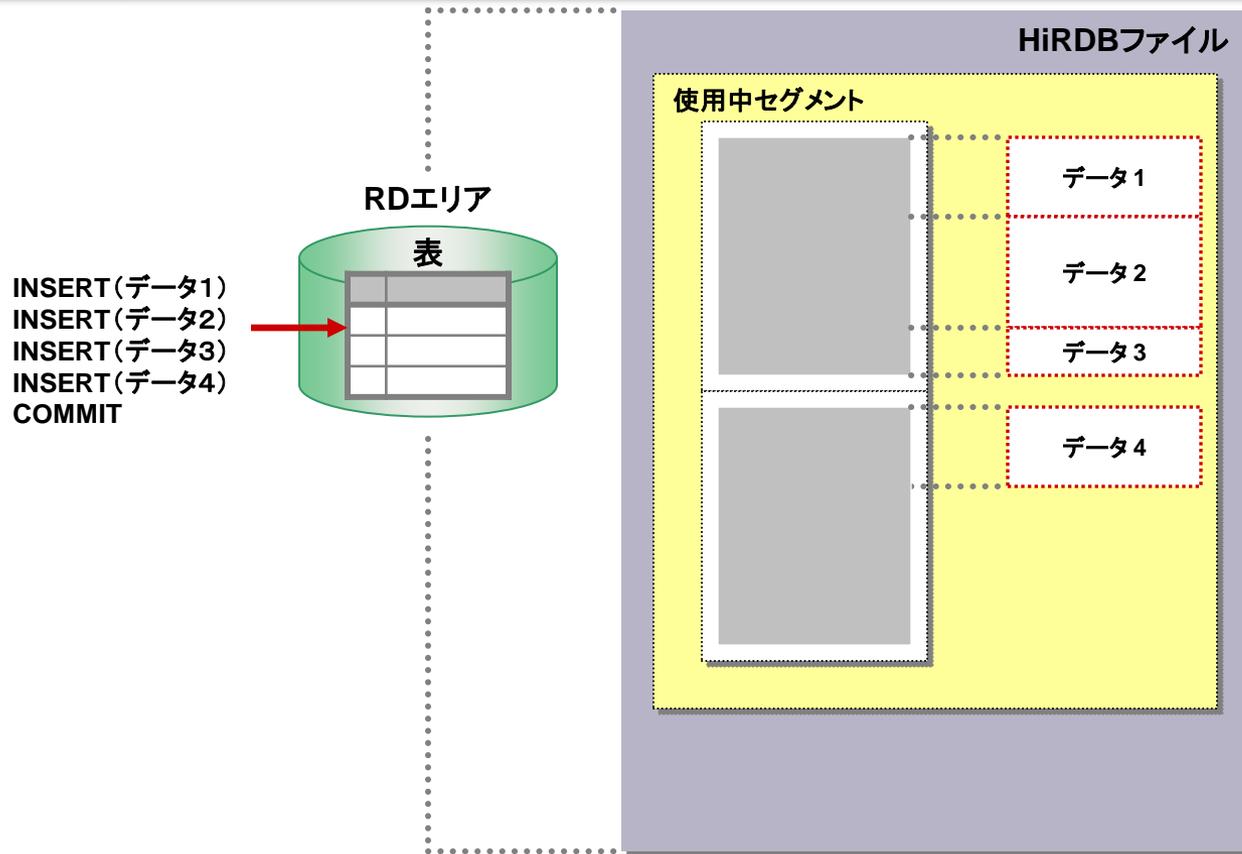
空き領域再利用機能(SEGMENT REUSE)を使用すると、空き領域を探し始める位置が変更になります。毎回先頭の使用済みセグメントの先頭ページから探すのではなく、前回データを格納したページから、探す処理を始めます。



- : 空き領域を探す処理
- : データ格納領域
- : 空き領域

5-5-3 空き領域の再利用 SEGMENT REUSEの有効な場面

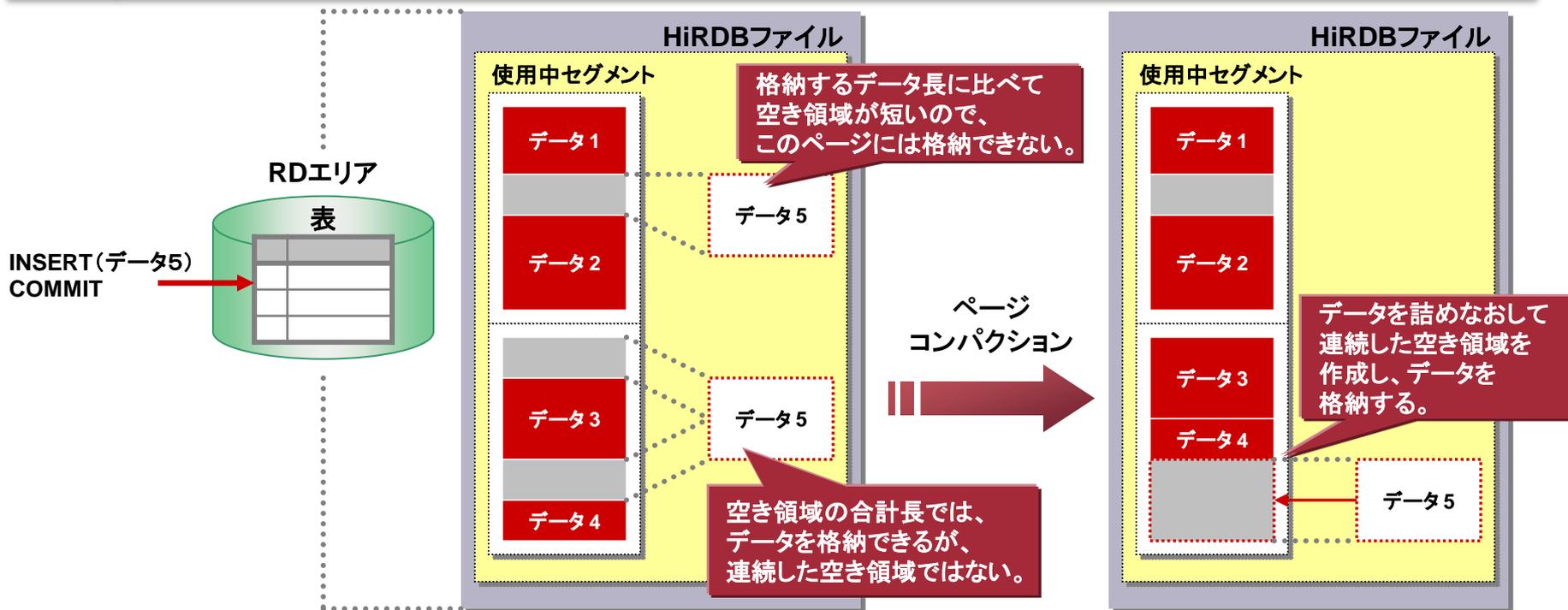
解説 データを格納した順番で削除するなど、連続した空き領域が発生する表の場合は、SEGMENT REUSEを使用することで、空き領域を探す処理を速くすることができます。また、データ配置の乱れも発生しないため、データベース再編成を実施しなくても、運用を継続することができます。



— : 空き領域

5-5-4 空き領域の再利用 断片的な空き領域の場合

解説 空き領域が連続していない場合、格納するデータ長によっては、空き領域を見つけても格納できない場合があります。また、ページ内の空き領域を合計すると格納できる場合は、ページ内のデータを詰めなおし(ページコンパクション)、その後追加するデータを格納します。



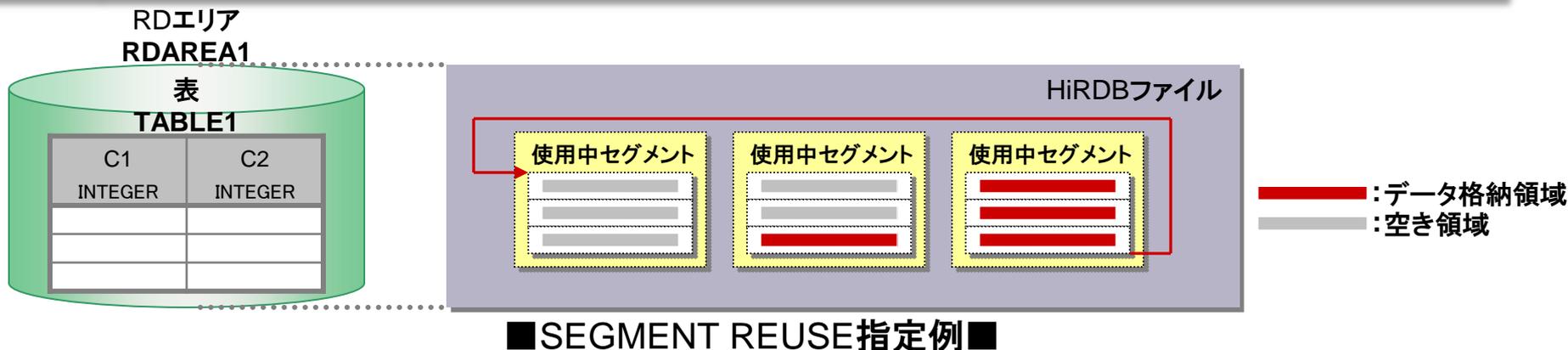
■ : データ格納領域
■ : 空き領域

空き領域が断片的に存在する場合は、
SEGMENT REUSEを使用しても、
データ格納処理性能の劣化を抑えることができません。

5-5-5 空き領域の再利用 SEGMENT REUSE設定方法

解説

空き領域再利用機能(SEGMENT REUSE)を使用する場合は、表定義(CREATE TABLE)の表オプションまたは表定義変更(ALTER TABLE)の列属性変更定義に、SEGMENT REUSEを指定します。



```
CREATE TABLE TABLE1  
( C1 INTEGER , C2 INTEGER )  
IN RDAREA1  
SEGMENT REUSE セグメント数  
※この時指定するセグメント数は表の容量見積も  
りを行った結果になります。
```

```
ALTER TABLE TABLE1  
CHANGE SEGMENT REUSE セグメント数  
※自動増分等により容量が増加した場合、  
ALTER TABLEでSEGMENT REUSEを変更  
します。
```

注意事項

使用セグメント数が指定値に達した時点で、新たに未使用セグメント確保するのではなく、先頭のセグメントから空き領域を探します。ただし、使用中セグメント内に空き領域が存在しなかった場合は、新たに未使用セグメントを確保し、データを格納します。これ以降、新たな未使用セグメントを確保する前には、必ず先頭からすべての使用中セグメントに対して、空き領域を探す処理が動作するため、処理性能が劣化します。

→空き領域が存在しない表に対しては、空き領域再利用機能を使用しないで下さい。

解説

1つのRDエリアに複数の表やインデックスを格納している場合、表ごとに使用するセグメント数を指定することで、データのライフサイクルに合わせて使用する領域を限定することができます。セグメント数を指定する場合は、データ量や削除までの期間などを把握する必要があります。

■ SEGMENT REUSE

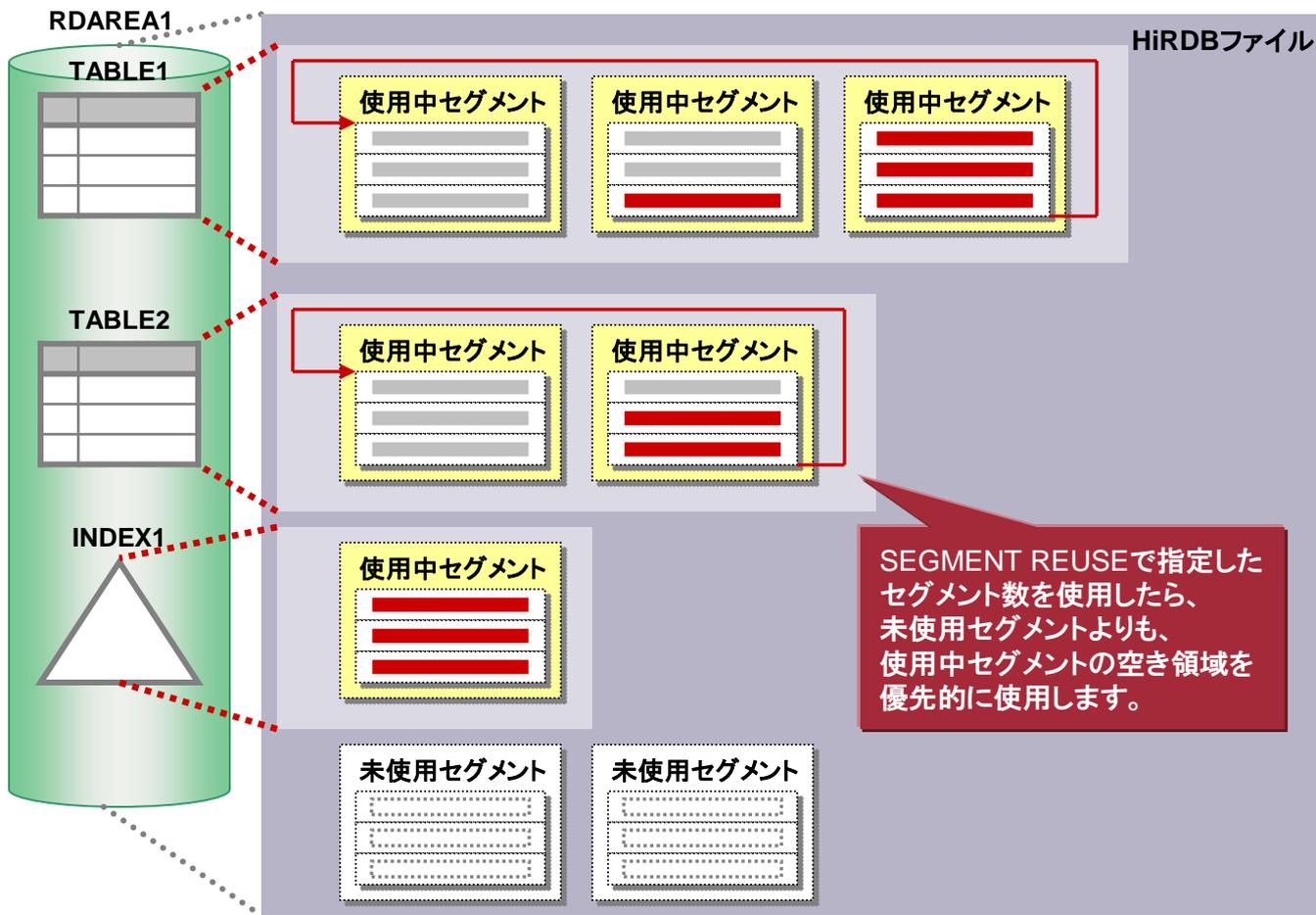
セグメント数指定例■

```
CREATE TABLE TABLE1
(C1 INTEGER,
 C2 INTEGER)
IN RDAREA1
SEGMENT REUSE 3
```

```
CREATE TABLE TABLE2
(C1 INTEGER,
 C2 INTEGER)
IN RDAREA1
SEGMENT REUSE 2
```

※インデックスには、
SEGMENT REUSEの
指定はできません。

■ : データ格納領域
■ : 空き領域
□ : 未使用領域



5-5-7 空き領域の再利用 BINARY専用セグメントが存在する表

解説

SEGMENT REUSEを使用している表で、基本行を格納するセグメントとBINARY専用セグメントの2種類が存在する場合は、同じ種類のセグメントの中で空き領域を再利用します。



■表定義の例■

```
CREATE TABLE TABLE1  
( C1 INTEGER ,  
  C2 BINARY(32000) )  
IN RDAREA1  
SEGMENT REUSE 6
```

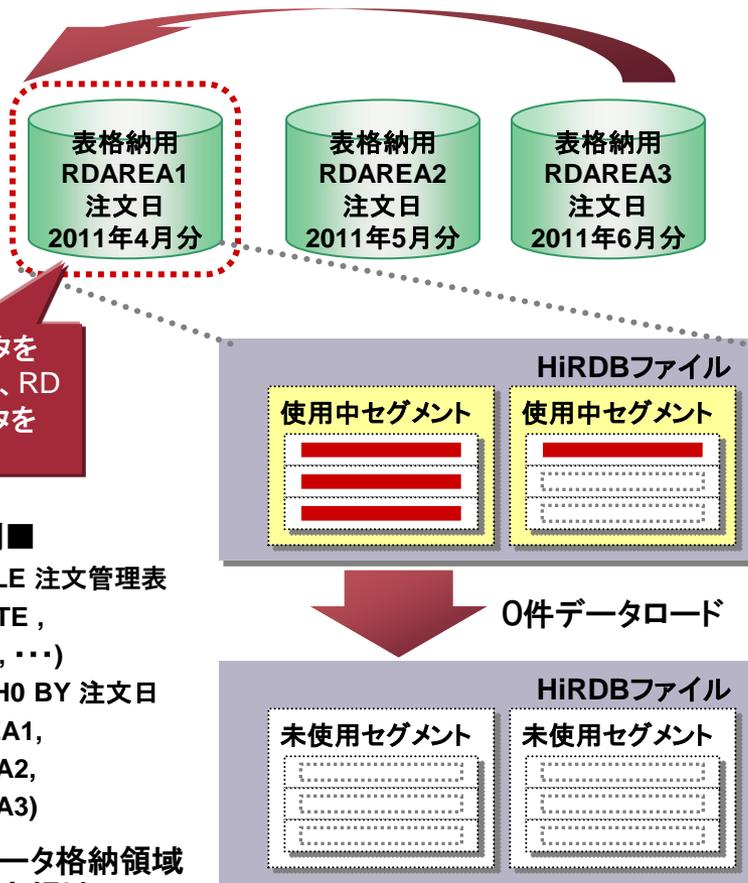
※SEGMENT REUSEでセグメント数を指定する場合は、
基本行を格納するセグメント数とBINARY専用セグメント数の
合計を指定してください。

— : データ格納領域
— : 空き領域

基本行を格納するセグメントに空き領域が存在しても、
BINARY専用セグメントに空き領域がない場合は、
容量不足になりますので、ご注意ください。

解説

RDエリアを循環利用する運用は、データベースの断片化を低減できる場合があります。ハッシュ関数HASH0等で表分割して、データ格納RDエリアを月単位で循環させて割り当てます。最初のRDエリアに新しい月のデータが格納される前に、データベース作成ユーティリティ(pdload)の作成モードで、0件データロードにより、RDエリアのデータを削除します。



7月分のデータを格納する前に、RDエリアのデータを削除する

■表定義の例■

```
CREATE TABLE 注文管理表
(注文日 DATE,
 注文ID INT, ...)
HASH HASH0 BY 注文日
IN (RDAREA1,
  RDAREA2,
  RDAREA3)
```

■ : データ格納領域
■ : 空き領域

■0件データロードの実行例■

```
pdload      : データベース作成ユーティリティ
-d          : 作成モード
-i s        : インデクス作成方法に更新モードを指定(*1)
注文管理表 : 表識別子
/HiRDB/control_file/load : 制御文ファイル名
```

(*1): インデクス作成方法に更新モードを指定することで、インデクス一括作成処理を抑止できます。

■制御文ファイルの内容(/HiRDB/control_file/load)■

```
source RDAREA1 /HiRDB/DATA/DATA0
: 対象RDエリア名と入力データファイル(0バイトファイル)
の名称を指定
```

0件データロードにより、処理対象の表が使用していたセグメントは未使用セグメントになります。次の0件データロードを行うまで、データベースの断片化による影響がなければ、定期的に未使用セグメントに戻るため、再編成は不要となります。

6. データベースの断片化を解消する運用

6. データベースの断片化を解消する運用

6. 1 運用の検討

6. 2 RDエリアの拡張

6. 3 RDエリアの自動増分

6. 4 空き領域の解放

6. 5 データベース再編成

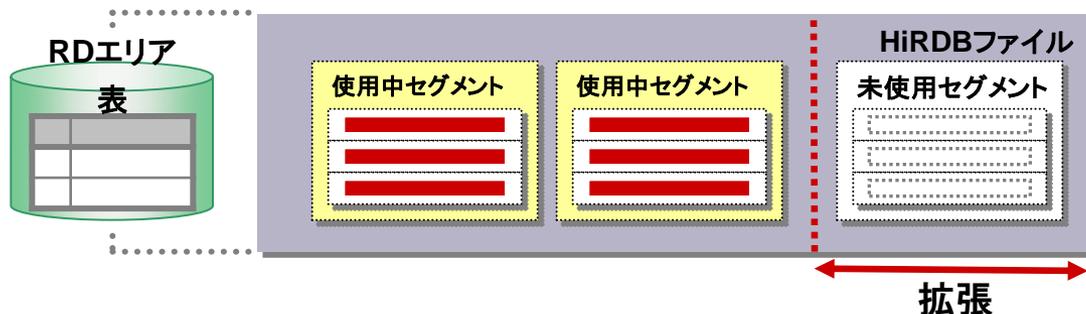
解説

データベースの領域が不足した場合、データベースの領域を拡張するか、断片化の解消をすることで、データベースの安定稼働が可能となります。
本章では、それぞれの運用について解説します。

RDエリアの拡張

◆RDエリアを拡張する

基本的にデータの削除を行わない表は、空き領域が存在しないため、RDエリアの使用率が高くなった時点で、RDエリアを拡張してください。

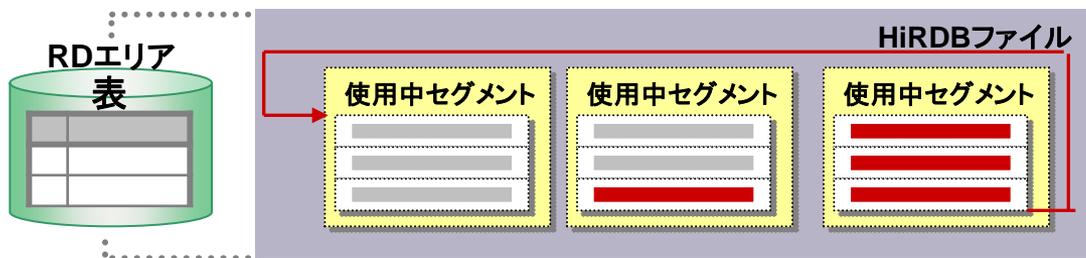


RDエリアの自動増分

空き領域の解放

◆空き領域の再利用、または再編成する

データの追加と削除を繰り返し、格納効率やアクセス効率が低下している場合は、RDエリアを拡張せずに、空き領域の再利用やデータベースの再編成を行います。



データベース再編成

- : データ格納領域
- : 空き領域
- : 未使用領域

6. データベースの断片化を解消する運用

- 6. 1 運用の検討
- 6. 2 **RDエリアの拡張**
- 6. 3 RDエリアの自動増分
- 6. 4 空き領域の解放
- 6. 5 データベース再編成

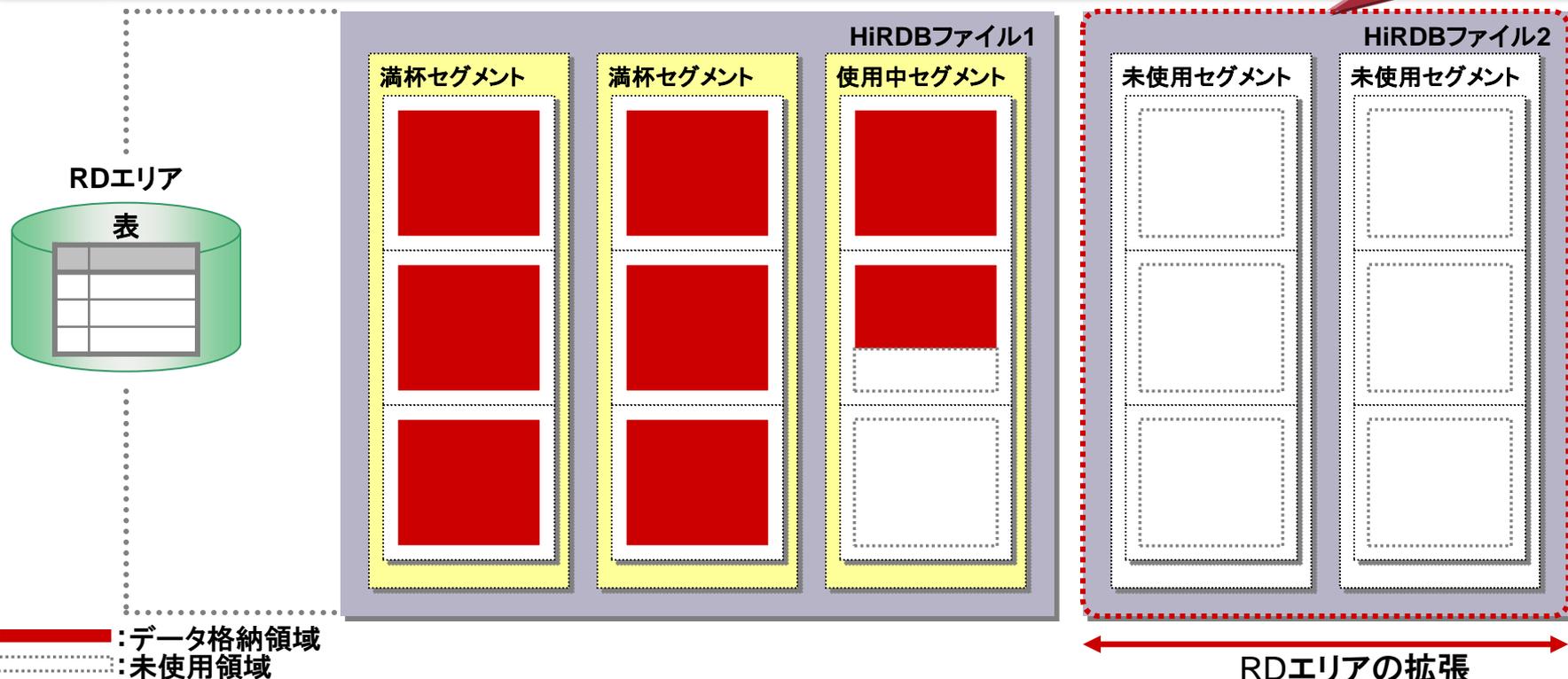
6-2-1 RDエリアの拡張 ユティリティによる拡張

解説

RDエリアの拡張は、データベース構成変更ユティリティ(pdmod)で実行できます。
pdmodでは、RDエリアを構成するHiRDBファイルを追加することで、RDエリアを拡張します。

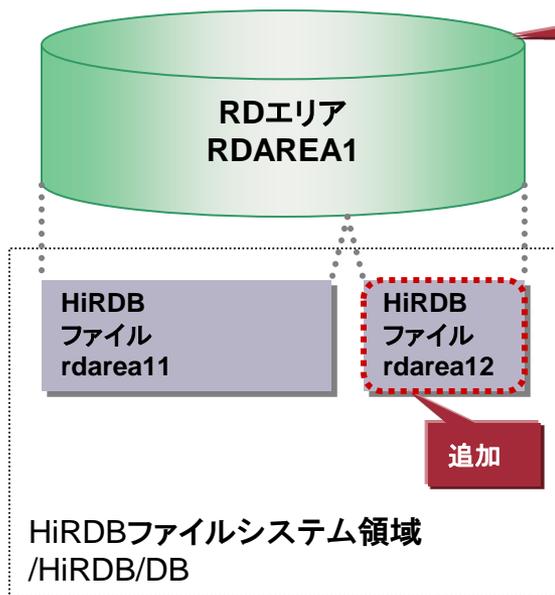
RDエリア拡張の検討のポイント

- ・pdreclaimやDB再編成では容量が不足している
- ・自動増分回数や容量が限界値を超えてしまった
- ・RDエリアを構成するファイルが15ファイル以下である
- ・更新ランザクションが拡張完了まで排他待ちする
- ・システム定義のpd_max_file_noの値に余裕がある



6-2-2 RDエリアの拡張 ユーティリティによる拡張方法

構成変更ユーティリティ(pdmod)実行後の障害に備えて、RDエリア拡張後のRDエリアのバックアップを取得してください。



■RDエリアの拡張例■

pdmod	: データベース構成変更ユーティリティ
-a /HiRDB/control_file/expand	: 制御文ファイル名

■制御文ファイル(/HiRDB/control_file/expand)の内容■

expand rdarea RDAREA1	: RDエリア名
file name "/HiRDB/DB/rdarea12"	: 追加するHiRDBファイル名
initial 500 segments	: 上記HiRDBファイルに作成するセグメント数
;	

RDエリアの拡張は、該当RDエリアにアクセスする業務と、並行して実行できます。(*1)

(*1): 更新系トランザクションと並行実行する場合は、排他待ちが発生することがあります。

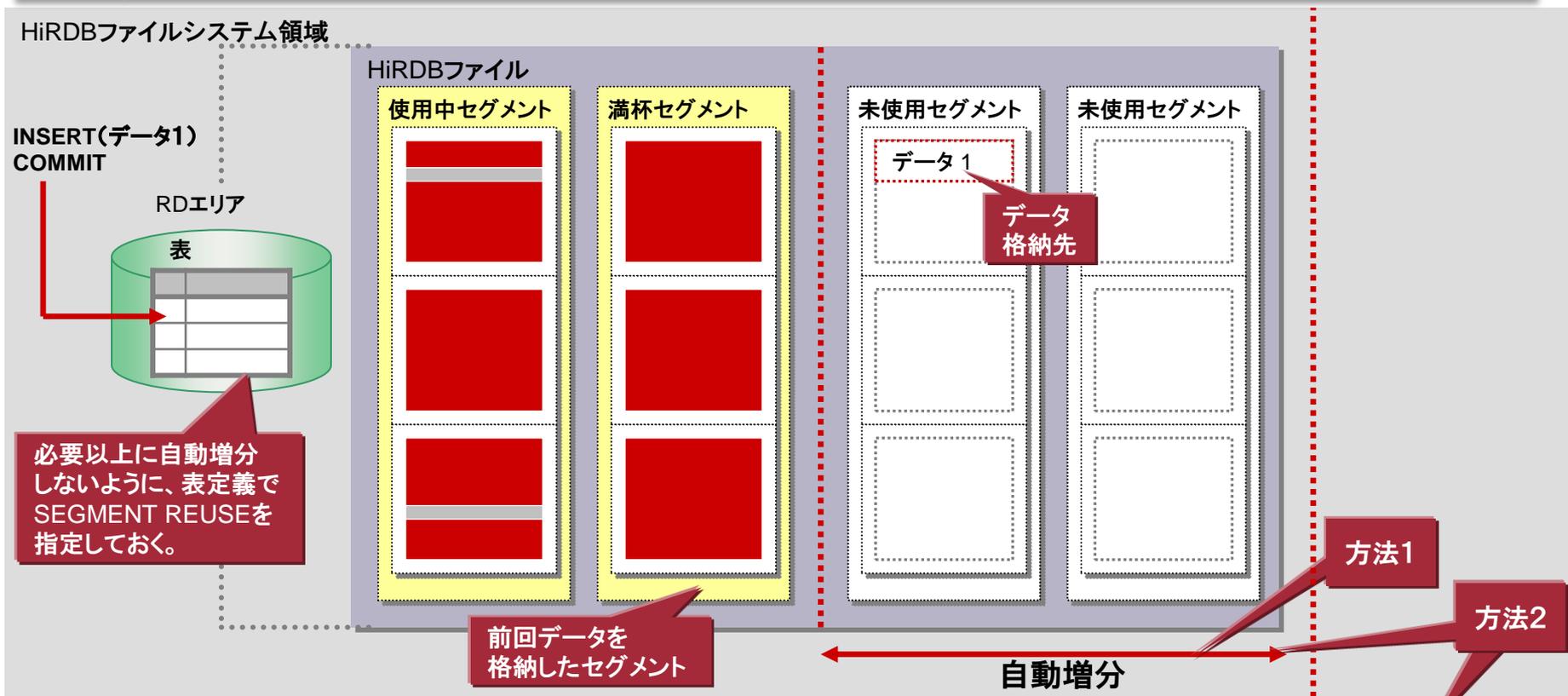
6. データベースの断片化を解消する運用

- 6. 1 運用の検討
- 6. 2 RDエリアの拡張
- 6. 3 **RDエリアの自動増分**
- 6. 4 空き領域の解放
- 6. 5 データベース再編成

6-3-1 RDエリアの自動増分

解説

RDエリアの自動増分機能を使用すると、未使用セグメントを新たに割り当てようとした時に、RDエリア内に未使用セグメントが存在しなかった場合は、HiRDBファイルのサイズを自動的に拡張し、未使用セグメントを追加します。



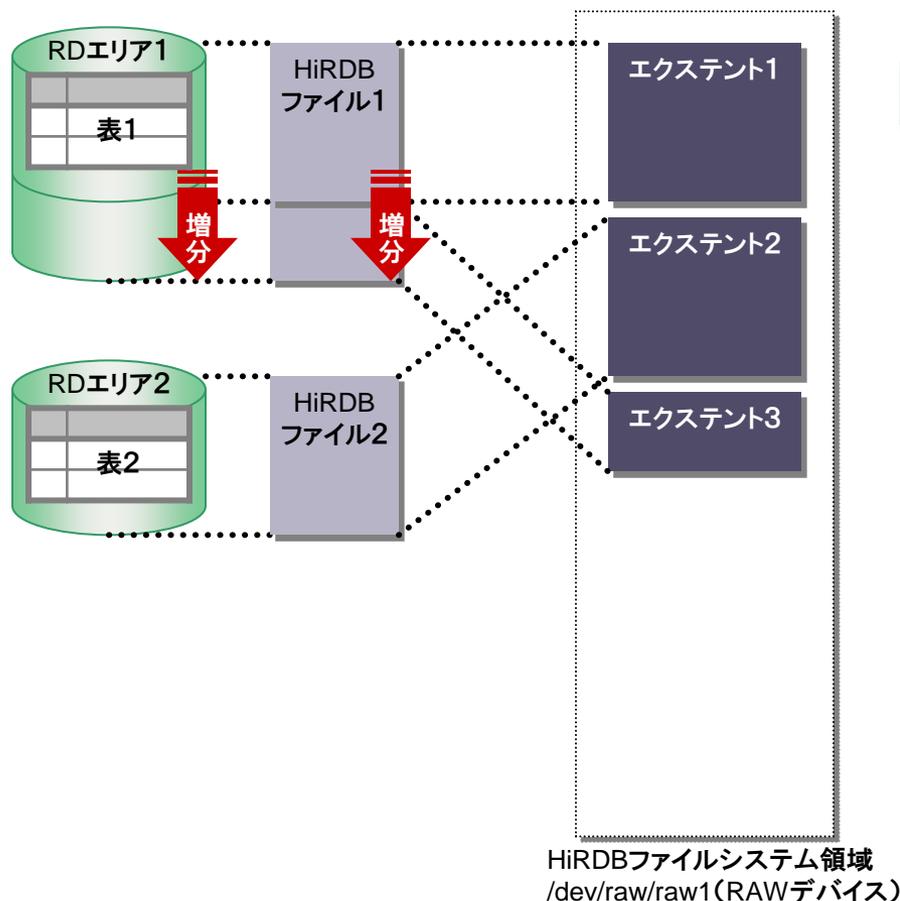
- : データ格納領域
- : 空き領域
- : 未使用領域

HiRDBファイルを自動増分する方法(方法1)と
HiRDBファイルとHiRDBファイルシステム領域を自動増分する方法(方法2)があります。

6-3-2 RDエリアの自動増分 RDエリアの自動増分(方法1)

解説

方法1は、HiRDBファイルシステム領域の容量は増分せず、HiRDBファイルだけを自動増分する方法です。



特長

RAWデバイス全体のサイズで、HiRDBファイルシステム領域を作成します。その中に、複数のHiRDBファイルを作成します。

HiRDBファイルシステム領域内に、拡張するための領域を確保しておきます。

HiRDBファイルが自動増分する時には、エクステント（HiRDBファイルシステム領域内の連続した領域の固まり）を追加して、HiRDBファイルを増分します。

1つのHiRDBファイルを構成するエクステントの最大数は24ですので、1つのHiRDBファイルは、23回まで増分可能です。

6-3-3 RDエリアの自動増分 RDエリアの自動増分指定方法(方法1)

■HiRDBファイルシステム領域の作成例■

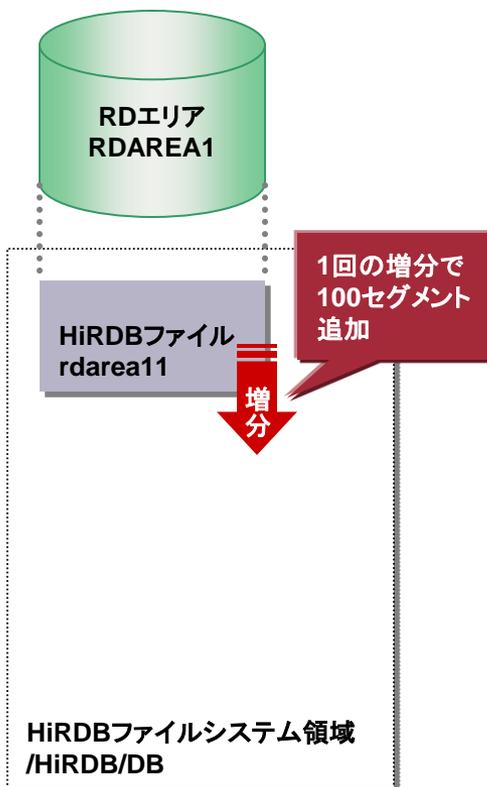
```
pdfmkfs          :HiRDBファイルシステム領域の初期設定コマンド
-n 1024          :HiRDBファイルシステム領域サイズ(メガバイト単位)
-l 10            :HiRDBファイルシステム領域に作成するHiRDBファイルの最大数
-k DB            :使用目的、RDエリア用のHiRDBファイルを作成する場合はDBを指定
-e 230           :最大増分回数、HiRDBファイル数(-l指定値)×23(最大増分回数)を
                  目安に指定する
-i              :HiRDBファイルシステム領域サイズ全体を初期化する
/HiRDB/DB       :HiRDBファイルシステム領域を割り当てるパス名
```

■RDエリアの作成例■

```
pdmod            :データベース構成変更ユーティリティ
-a /HiRDB/control_file/add :制御文ファイル名
```

■制御文ファイル(/HiRDB/control_file/add)の内容■

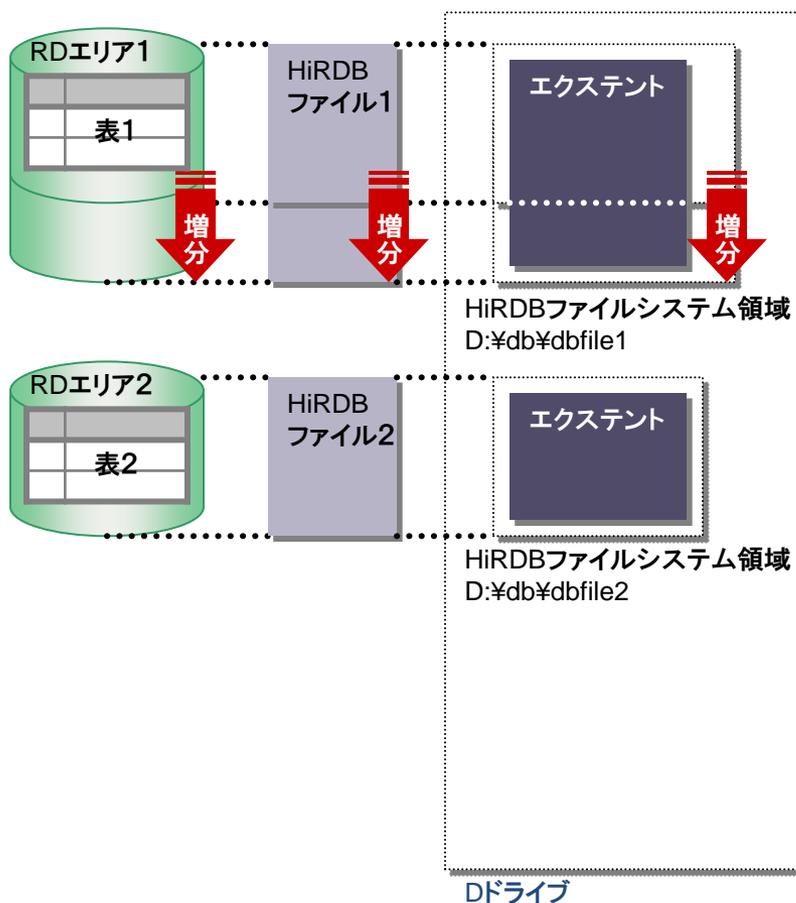
```
create rdarea RDAREA1          :RDエリア名
for user used by PUBLIC        :RDエリア属性
page 4096 characters           :ページサイズ
storage control segment 50 pages :セグメントサイズ
extension use 100 segments     :自動増分で追加するセグメント数
file name "/HiRDB/DB/rdarea11" :RDエリアを構成するHiRDBファイル名
initial 1000 segments         :上記HiRDBファイルに作成するセグメント数
;
```



6-3-4 RDエリアの自動増分 RDエリアの自動増分(方法2)

解説

方法2は、HiRDBファイルシステム領域を自動増分しつつ、HiRDBファイルも自動増分する方法です。



特長

1つのHiRDBファイルシステム領域には、1つのHiRDBファイルだけを作成します。HiRDBファイルの初期サイズに合わせて、HiRDBファイルシステム領域のサイズを決定します。

HiRDBファイルシステム領域を作成したドライブ内に、拡張するための領域を確保してください。

HiRDBファイルが自動増分する時、HiRDBファイルシステム領域内に必要となる容量が足りない場合は、HiRDBファイルシステム領域も同時に増分します。

HiRDBファイルの最大サイズ(64GB)まで増分可能です。増分の回数に制限はありません。

注意事項

ドライブを定期的に容量監視して、ディスクが一杯にならないように、注意してください。

容量不足になるとDB破壊となる場合があります。

6-3-5 RDエリアの自動増分 RDエリアの自動増分指定方法(方法2)

■HiRDBファイルシステム領域の作成例■

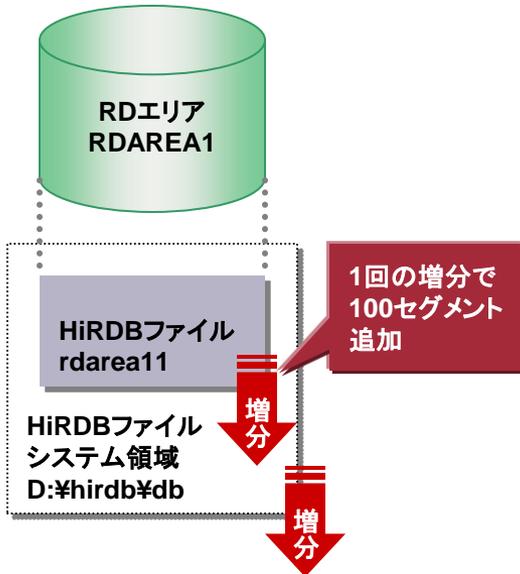
```
pdfmkfs          : HiRDBファイルシステム領域の初期設定コマンド
-n 250           : HiRDBファイルシステム領域サイズ(メガバイト単位)
-k DB            : 使用目的。RDエリア用のHiRDBファイルを作成する場合はDBを指定
-i              : HiRDBファイルシステム領域サイズ全体を初期化する
-a              : HiRDBファイルシステム領域サイズが-nオプションで指定した値に
                  達したとき、自動的にHiRDBファイルシステム領域を拡張する
D:¥hirdb¥db      : HiRDBファイルシステム領域を割り当てるファイル名
```

■RDエリアの作成例■

```
pdmod            : データベース構成変更ユーティリティ
-a D:¥HiRDB¥control_file¥add : 制御文ファイル名
```

■制御文ファイル(D:¥HiRDB¥control_file¥add)の内容■

```
create rdarea RDAREA1          : RDエリア名
for user used by PUBLIC         : RDエリア属性
page 4096 characters            : ページサイズ
storage control segment 50 pages : セグメントサイズ
extension use 100 segments    : 自動増分で追加するセグメント数
file name "D:¥hirdb¥db¥rdarea11" : RDエリアを構成するHiRDBファイル名
initial 1000 segments          : 上記HiRDBファイルに作成するセグメント数
;
```



選択基準

基本的に、HiRDBファイルシステム領域を拡張できる場合には、運用が容易な方式2がお勧めです。それ以外の場合は、方式1の適用を検討してください。

例えば、UNIX^(R)版の場合は、RAWデバイス上にHiRDBファイルを配置することを推奨していますが、この場合、HiRDBファイルシステム領域を拡張する方法は適さないため、方式1を適用します。

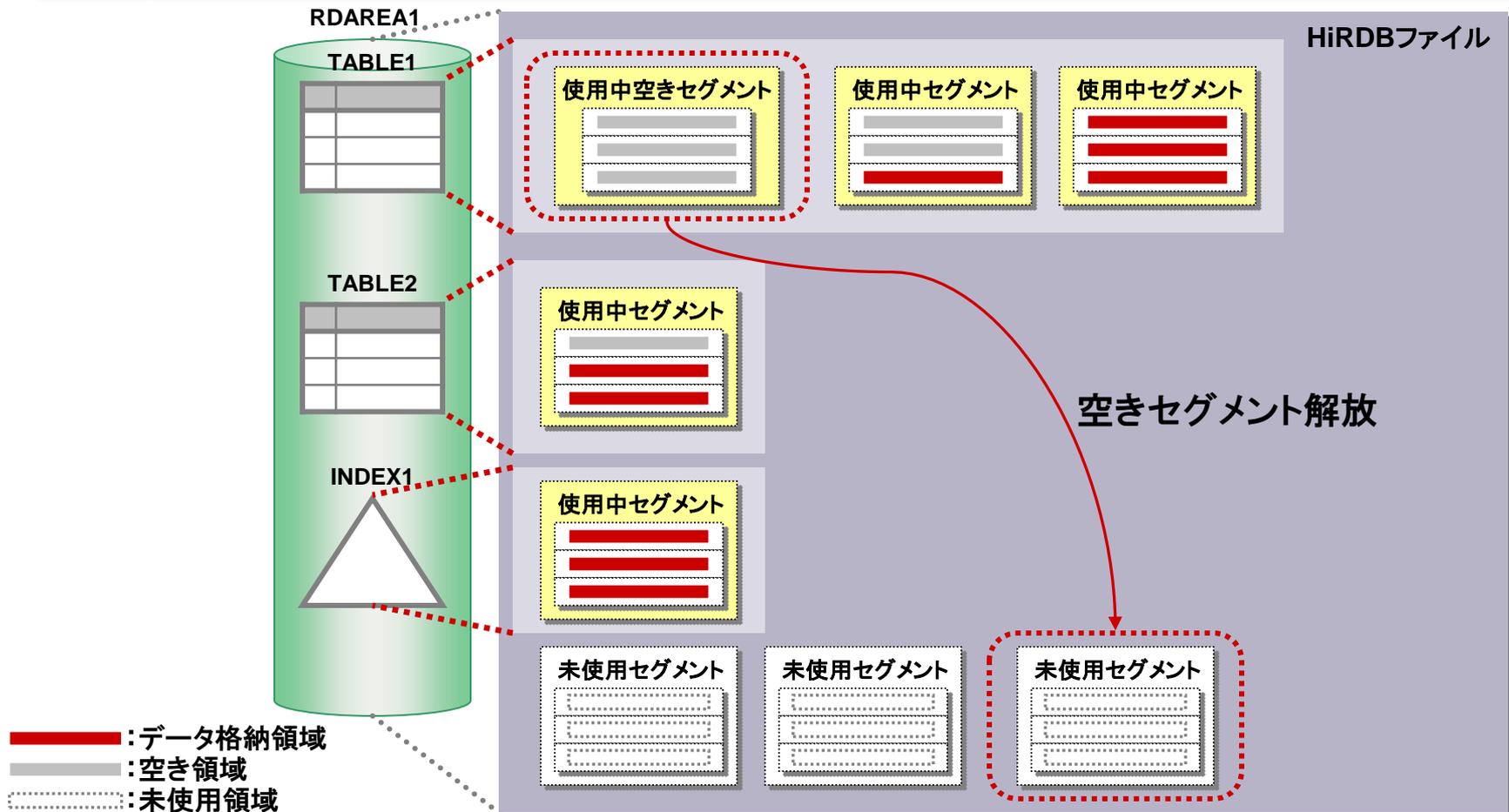
6. データベースの断片化を解消する運用

- 6. 1 運用の検討
- 6. 2 RDエリアの拡張
- 6. 3 RDエリアの自動増分
- 6. 4 空き領域の解放**
- 6. 5 データベース再編成

6-4-1 空き領域の解放 空き領域の解放による領域の再利用

解説

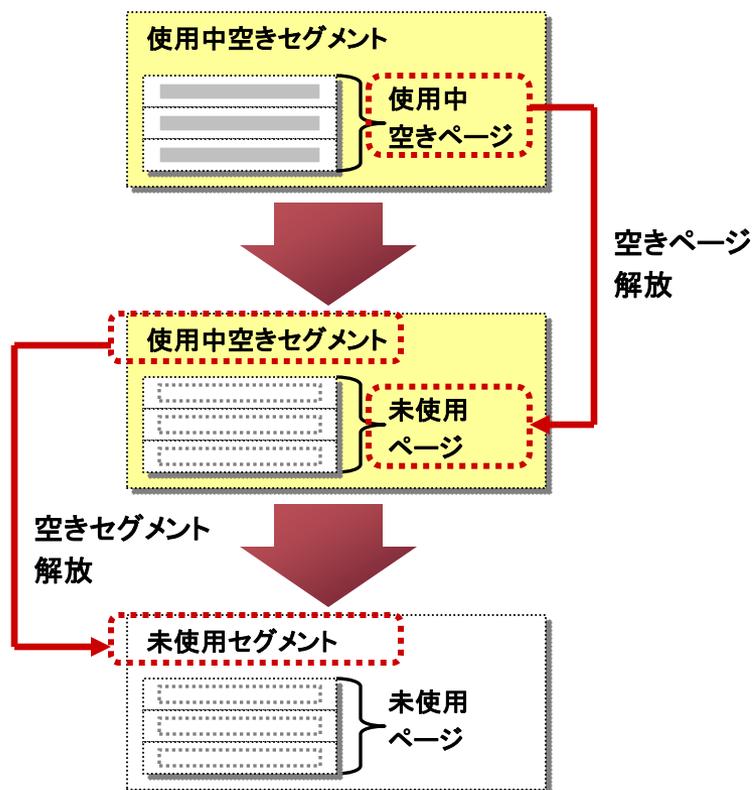
データを大量に削除するなど、セグメント全体が空き領域になる場合は、空きセグメント解放を行い、使用中空きセグメントを未使用セグメントに変更することで、他の表やインデクスがその領域を再利用することができます。



6-4-2 空き領域の解放 空きセグメント解放実行方法

解説

空きセグメント解放は、空きページ解放ユーティリティ(pdreclaim)で実行します。
まず最初に、空きページ解放を行い、使用中空きページを未使用ページに変更します。
その後で、空きセグメント解放を行い、使用中空きセグメントを未使用セグメントに変更します。



■空きセグメント解放の実行例■

(1) 空きページ解放

```
pdreclaim  
-k table      :対象資源の種別。表の場合はtableを指定。  
-t TABLE1    :表識別子。
```

(2) 空きセグメント解放

```
pdreclaim  
-k table      :対象資源の種別。表の場合はtableを指定。  
-t TABLE1    :表識別子。  
-j           :空きセグメント解放を行う場合に指定。
```

(3) 空きページ解放および空きセグメント解放 (HiRDB 08-05以降で使用可能)

```
pdreclaim  
-k table      :対象資源の種別。表の場合はtableを指定。  
-t TABLE1    :表識別子。  
-a         :空きページ解放および空きセグメント解放を  
              行う場合に指定。
```

同時実行性について

◆空きページ解放

空きページ解放は、オンライン業務と並行して実行することができます。ただし、空きページ解放の処理では、ページの使用状況を確認するために、表の使用済みページをすべて読み込みます。よって、オンライン業務と同時実行する場合は、トラフィックが低い時間帯に実行してください。

◆空きセグメント解放

空きページ解放ユーティリティ(pdreclaim)に -j オプションを指定して空きセグメント解放を実行する場合は、RDエリアに対して排他モードの排他制御を行いますので、オンライン業務の停止中に実行してください。

ページコンパクションについて

空きページ解放の延長で、ページコンパクションの処理を実行します。これにより、INSERTの延長でページコンパクションが実行される可能性を低くすることができます。

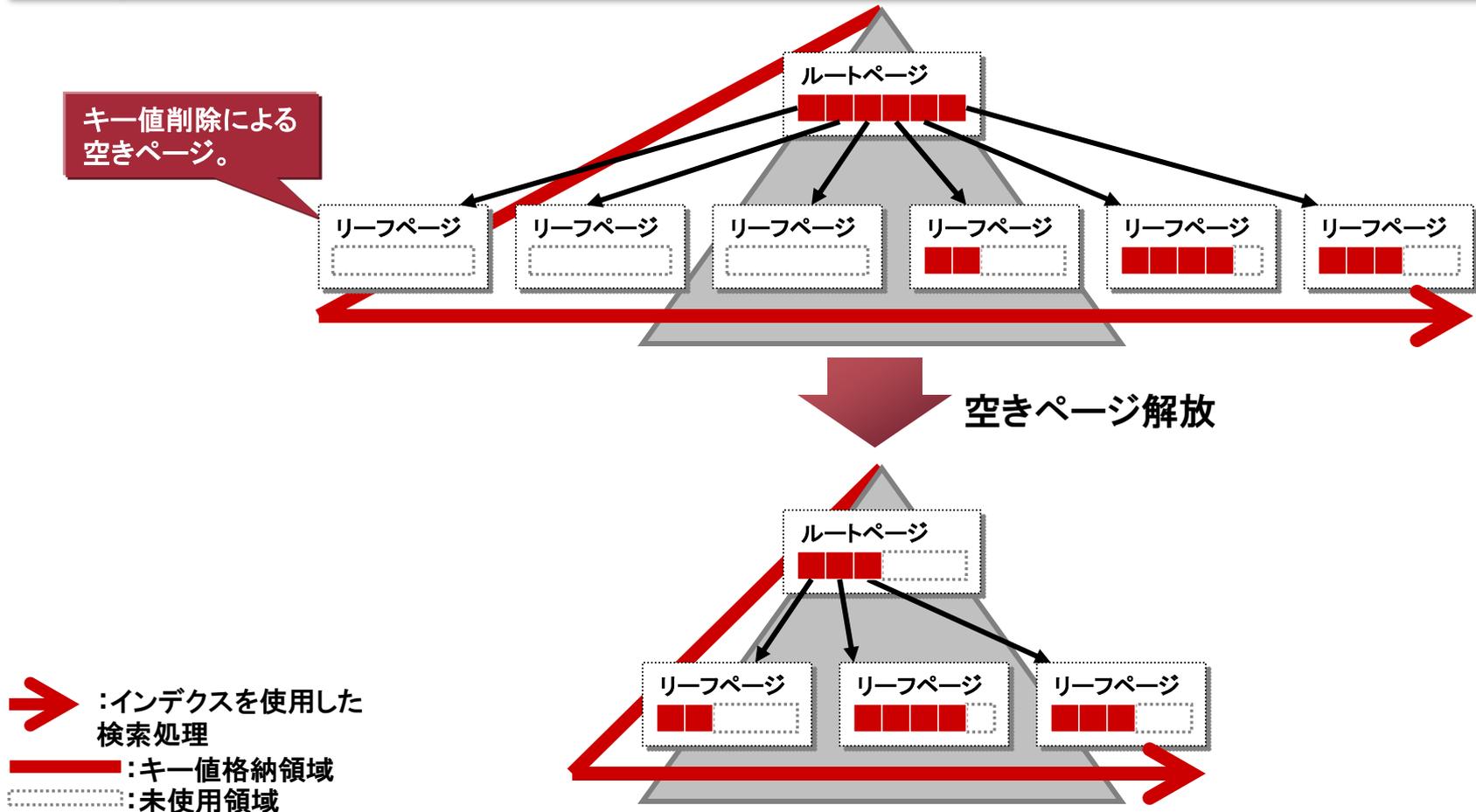
しかし、空きページ解放の延長でページコンパクションを実行すると、ページの書き込み処理や、システムログの出力処理を行います。よって、オンライン業務と並行して実行する場合は、ディスクI/O負荷により、業務性能に影響を与える可能性があります。

使用中空きページの状態を変更するだけが目的の場合は、空きページ解放ユーティリティ(pdreclaim)に -o オプションを指定して、ページコンパクションを行わないようにしてください。

6-4-4 空き領域の解放 インデクスの空きページ解放

解説

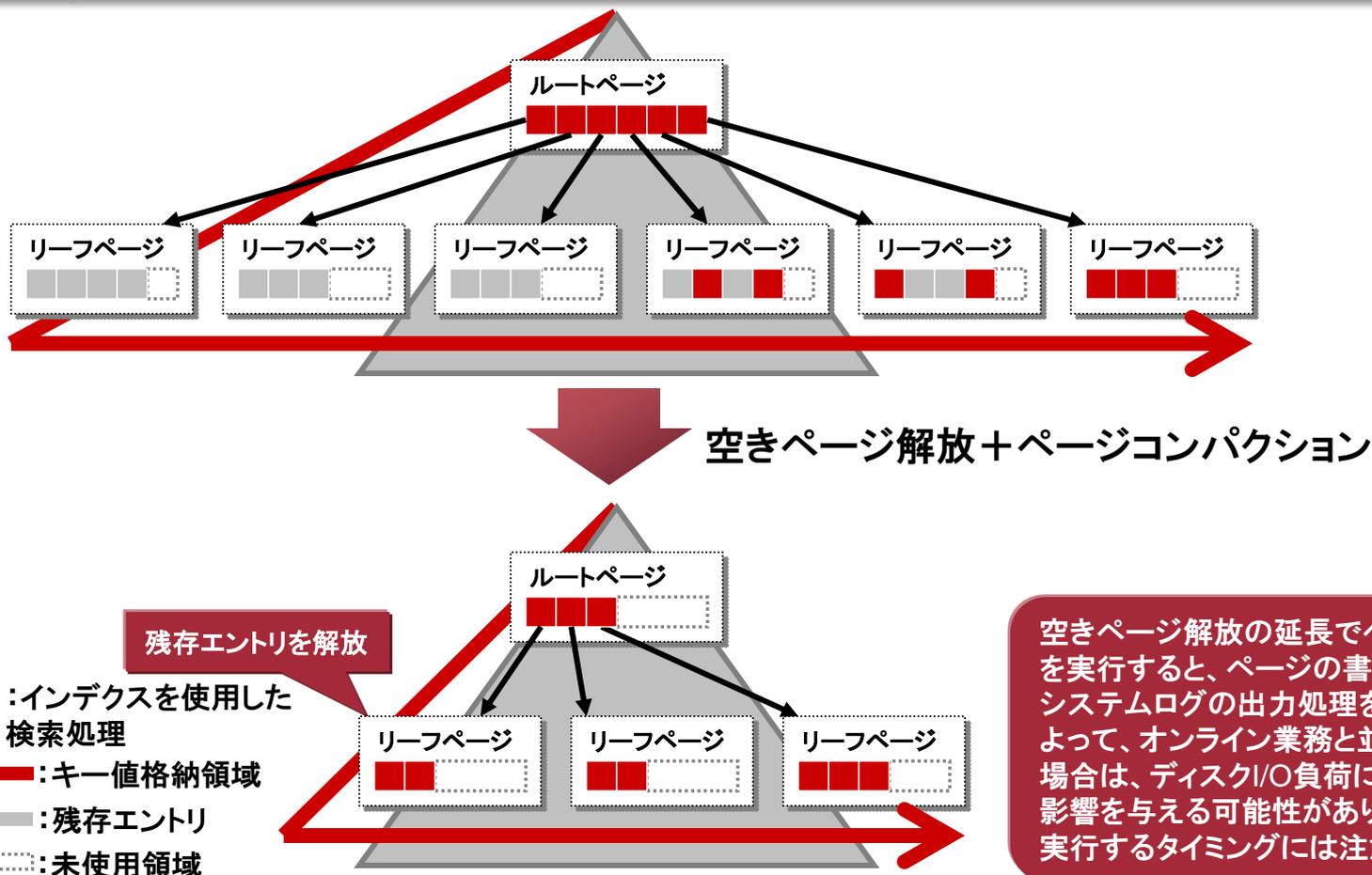
ページ内のすべてのキー値が削除されるようなインデクスの場合、空きページ解放ユーティリティ (pdreclaim) を実行することで、インデクスの構成ページ数を削減することができます。構成ページ数を削減すると、インデクスを使用した検索性能が向上する効果があります。



6-4-5 空き領域の解放 ユニークインデクスの空きページ解放

解説

キー値を頻繁に更新するユニークインデクスの場合、空きページ解放ユーティリティ(pdreclaim)にページコンパクションを行うオプション(-x)を指定して実行し、残存エントリを解放するようにしてください。

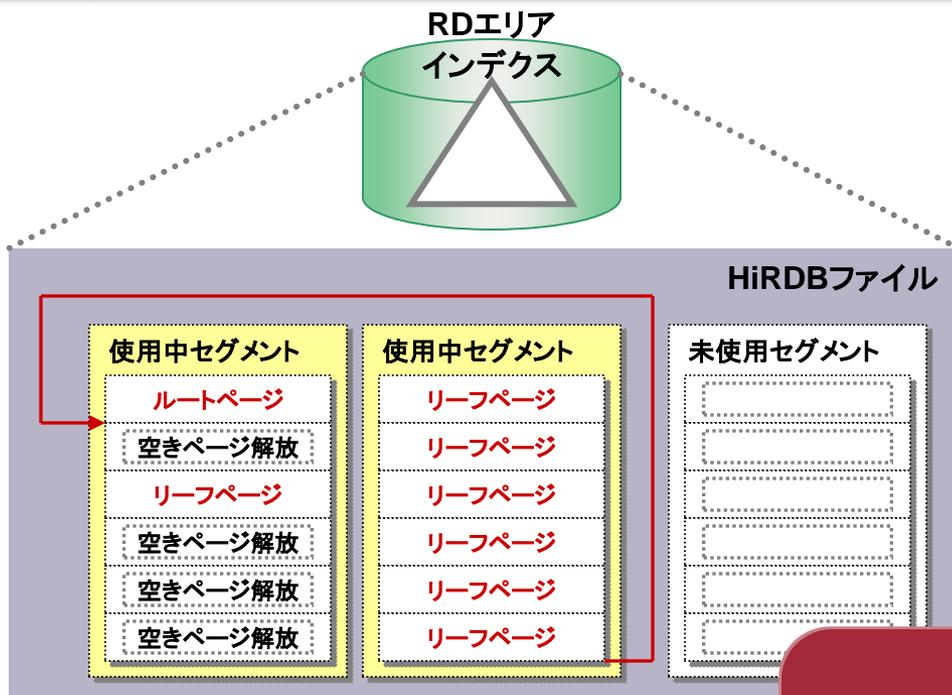


空きページ解放の延長でページコンパクションを実行すると、ページの書き込み処理や、システムログの出力処理を行います。よって、オンライン業務と並行して実行する場合は、ディスクI/O負荷により、業務性能に影響を与える可能性があります。実行するタイミングには注意してください。

6-4-6 空き領域の再利用 インデクスページの割り当て方法への効果

解説

前回ページを割り当てたセグメント内に未使用ページが存在しない場合、通常は新たな未使用セグメントを割り当てますが、空きページ解放を実行すると、すべての使用中セグメント内の未使用ページを優先的に使用します。

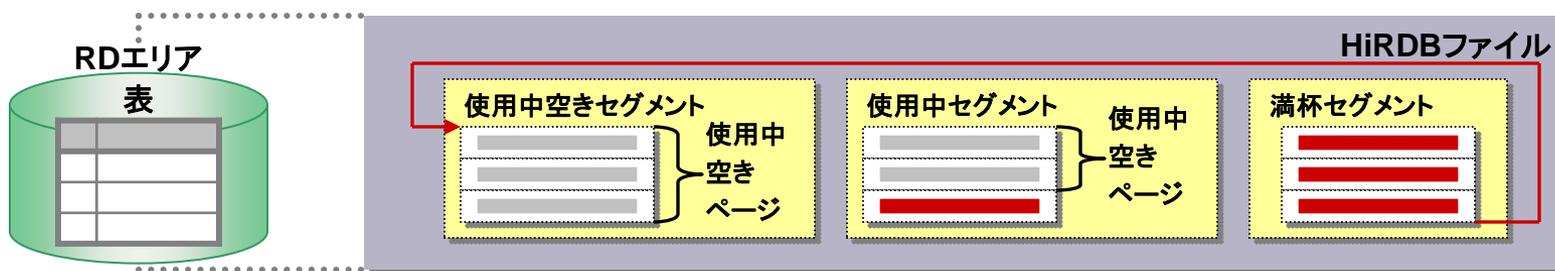


空きページ解放することで、使用中セグメント数を増やさない効果があります。ただし、削除するキー値の配置が、ページ単位で解放されない場合は、インデクスの再編成を実施してください。

.....: 未使用領域

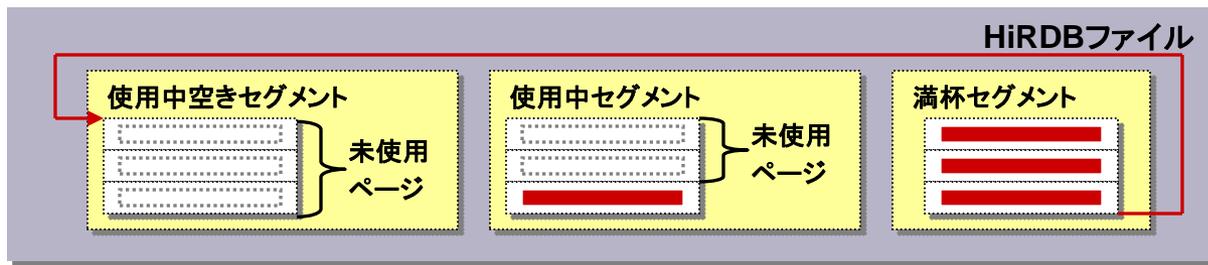
6-4-7 空き領域の解放 分岐行を含む表の空きページ解放

解説 256バイト以上の可変長文字列データの分岐行を含む表に対して、SEGMENT REUSEを使用して空き領域を再利用する場合は、空きページ解放も併せて運用することで、さらに効果があります。(理由は次のスライドで説明します)



```
CREATE TABLE TABLE1
(C1 INTEGER, C2 VARCHAR(4096))
IN RDAREA1
SEGMENT REUSE 3
```

空きページ解放



- : データ格納領域
- : 空き領域
- : 未使用領域

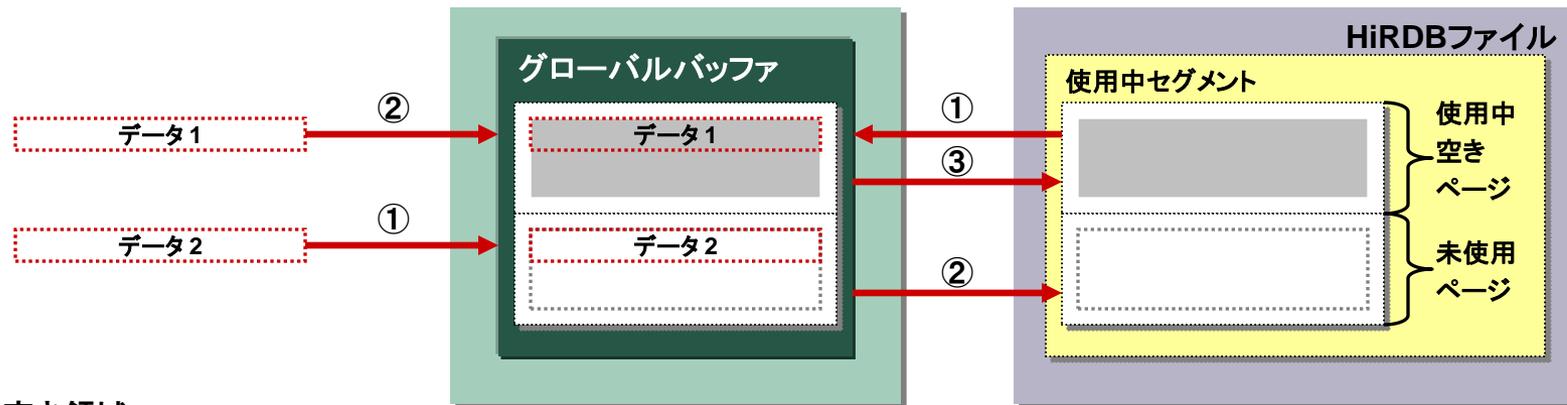
※SEGMENT REUSEを使用していない場合は、分岐行は未使用ページにしか格納できません。

6-4-8 空き領域の解放 分岐行の格納での未使用ページの効果

解説

256バイト以上の可変長文字列データの分岐行を格納するページを割り当てる際に、使用中ページの場合は、ページの内容をグローバルバッファに読み込んで、空き領域のサイズを確認してから、分岐行を格納します。未使用ページの場合は、現在のページを読み込む処理がないため、使用中空きページの状態よりもデータ格納処理を速くすることができます。

■使用中空きページと未使用ページのデータ格納処理の違い



■ : 空き領域
□ : 未使用領域

使用中空きページの場合

- ①現在のページの内容をグローバルバッファに読み込む
- ②空きページであることを確認し、グローバルバッファ上のページにデータを格納する
- ③シンクポイントなどのタイミングで、HiRDBファイル上のページに書き込

未使用ページの場合

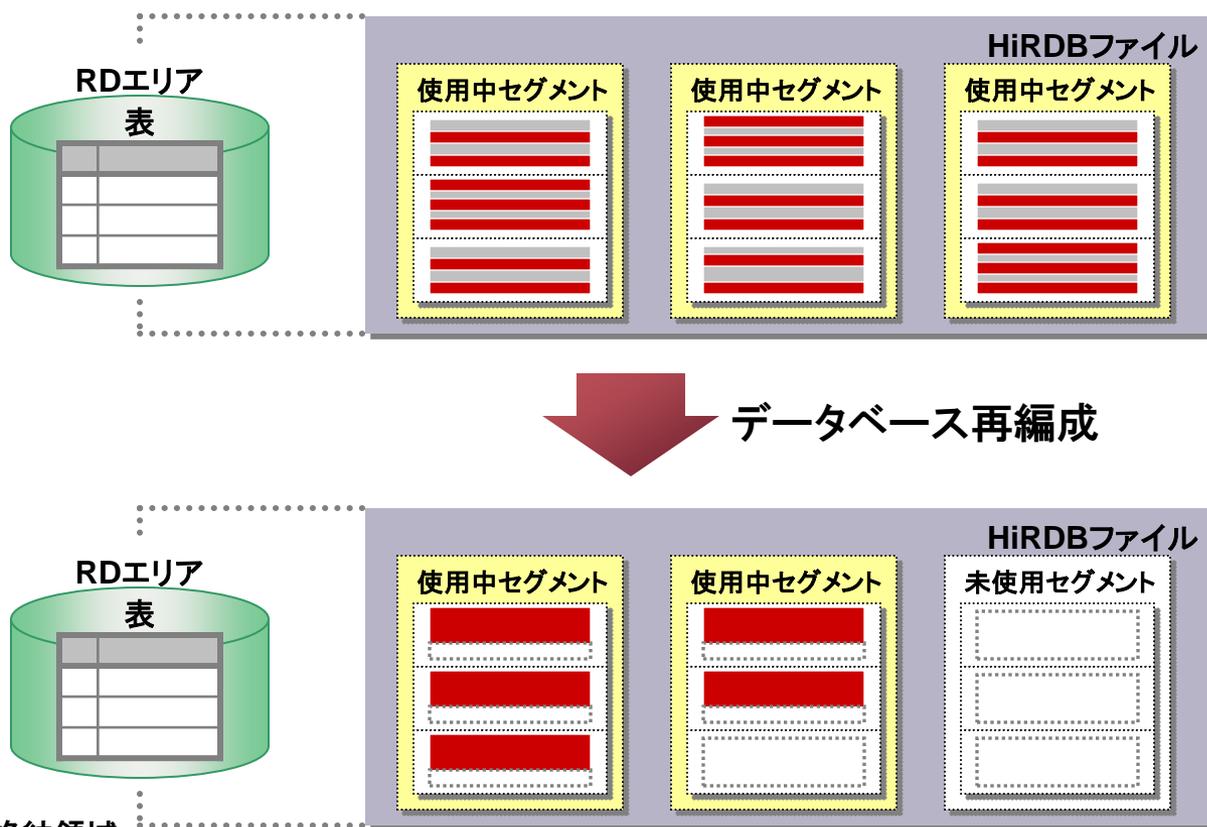
- ①グローバルバッファ上に空きページを確保し、データを格納する
- ②シンクポイントなどのタイミングで、HiRDBファイル上のページに書き込む

6. データベースの断片化を解消する運用

- 6. 1 運用の検討
- 6. 2 RDエリアの拡張
- 6. 3 RDエリアの自動増分
- 6. 4 空き領域の解放
- 6. 5 データベース再編成

解説

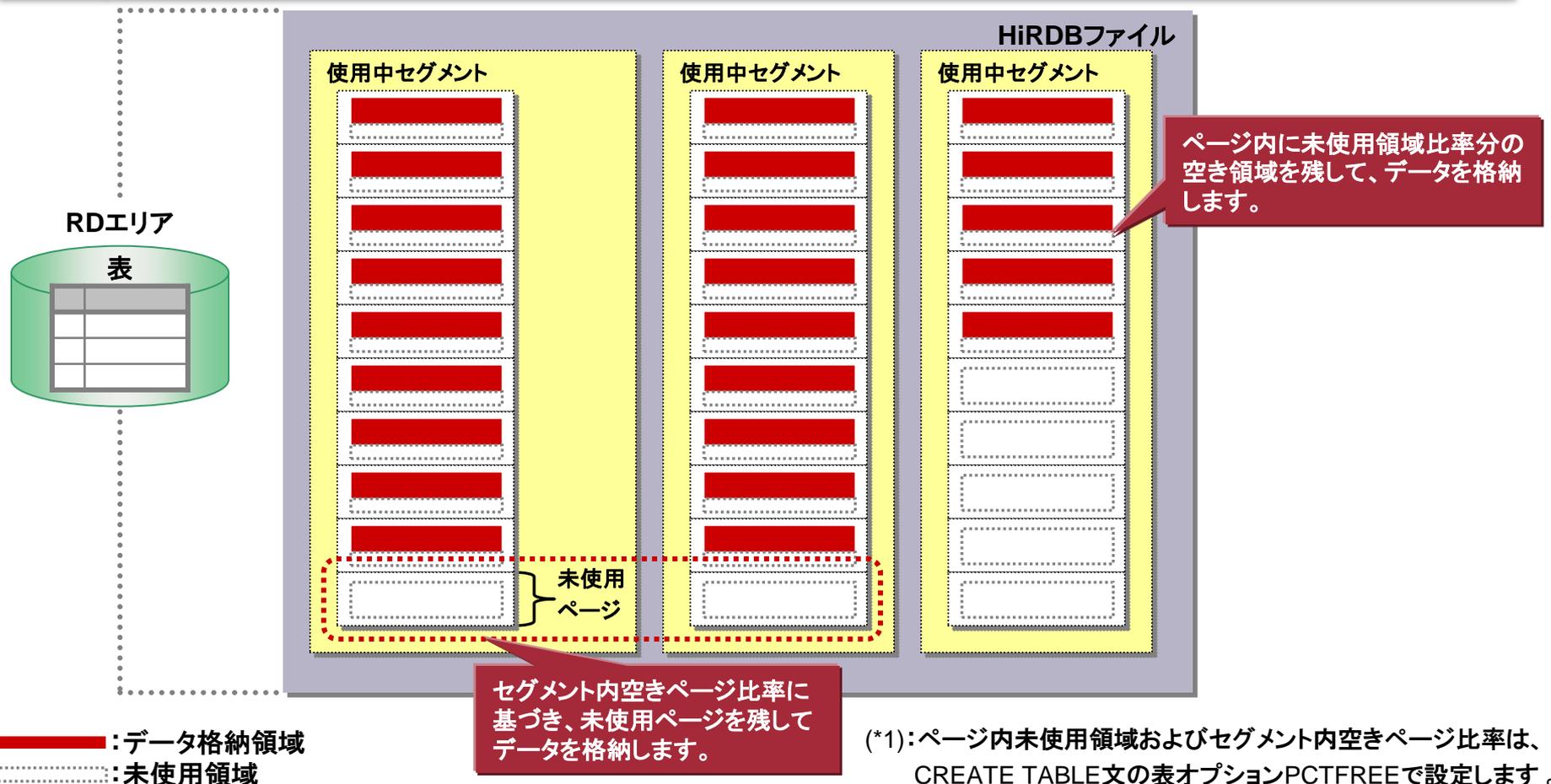
削除するデータが規則的でなく、空き領域が断片的に存在している状態の場合は、データベース再編成を実行して、データを詰めなおし、断片化を解消してください。



- : データ格納領域
- : 空き領域
- : 未使用領域

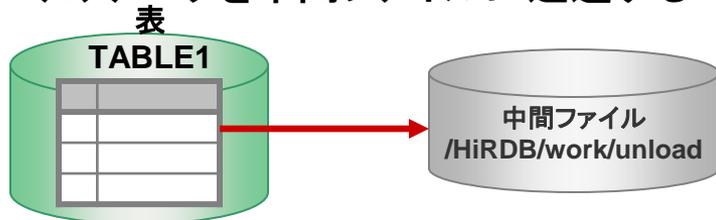
解説

データベース再編成ユーティリティ(pdrorg)を実行すると、表定義時に指定^(*)したページ内未使用領域比率およびセグメント内空きページ比率の設定に従い、データを詰めなおします。一定の空き領域を残しておくことで、データを更新する場合に効果があります。



■データベース再編成処理の流れ

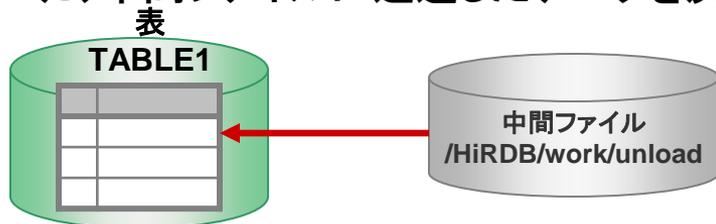
(1)データを中間ファイルに退避する



(2)データを全件削除する



(3)中間ファイルに退避したデータを戻す



■データベース再編成の実行例■

```
pdrorg          :データベース再編成ユーティリティ  
-k rorg         :処理内容。再編成の場合はrorgを指定。  
-t TABLE1     :表識別子  
/HiRDB/control_file/rorg :制御文ファイル名
```

■制御文ファイル(/HiRDB/control_file/rorg)の内容■

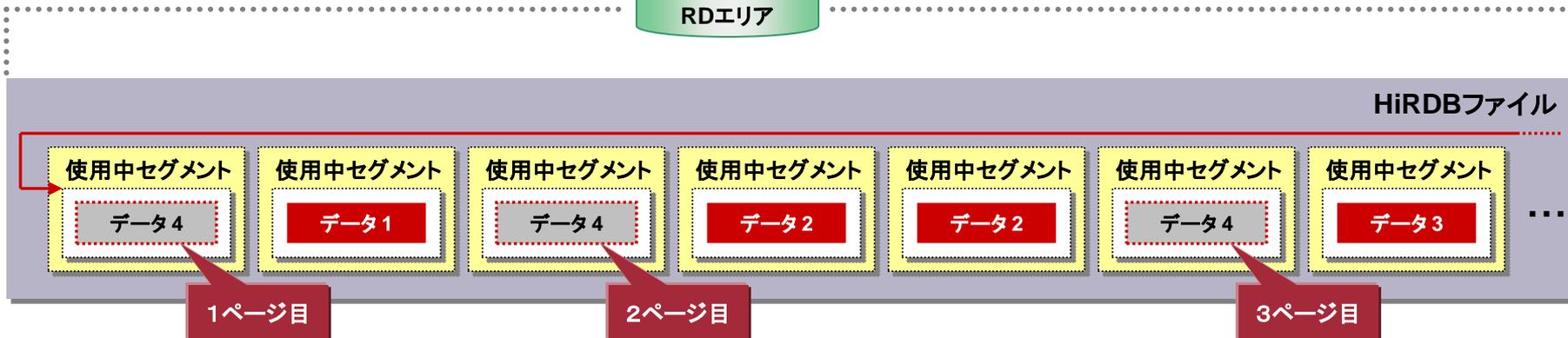
```
unload /HiRDB/work/unload :中間ファイル名を指定
```

解説

BLOB列を格納するユーザLOB用RDエリアの最後のセグメントまで使用した場合、空き領域を再利用します。ただし、空き領域を利用し始めると、業務性能に影響を及ぼす可能性があるため、未使用セグメントを使い切る前に、データベース再編成を実行してください。

■ 空き領域を再利用する場合

INSERT (データ4: 格納するために3ページ分必要)
COMMIT



■ : データ格納領域
■ : 空き領域

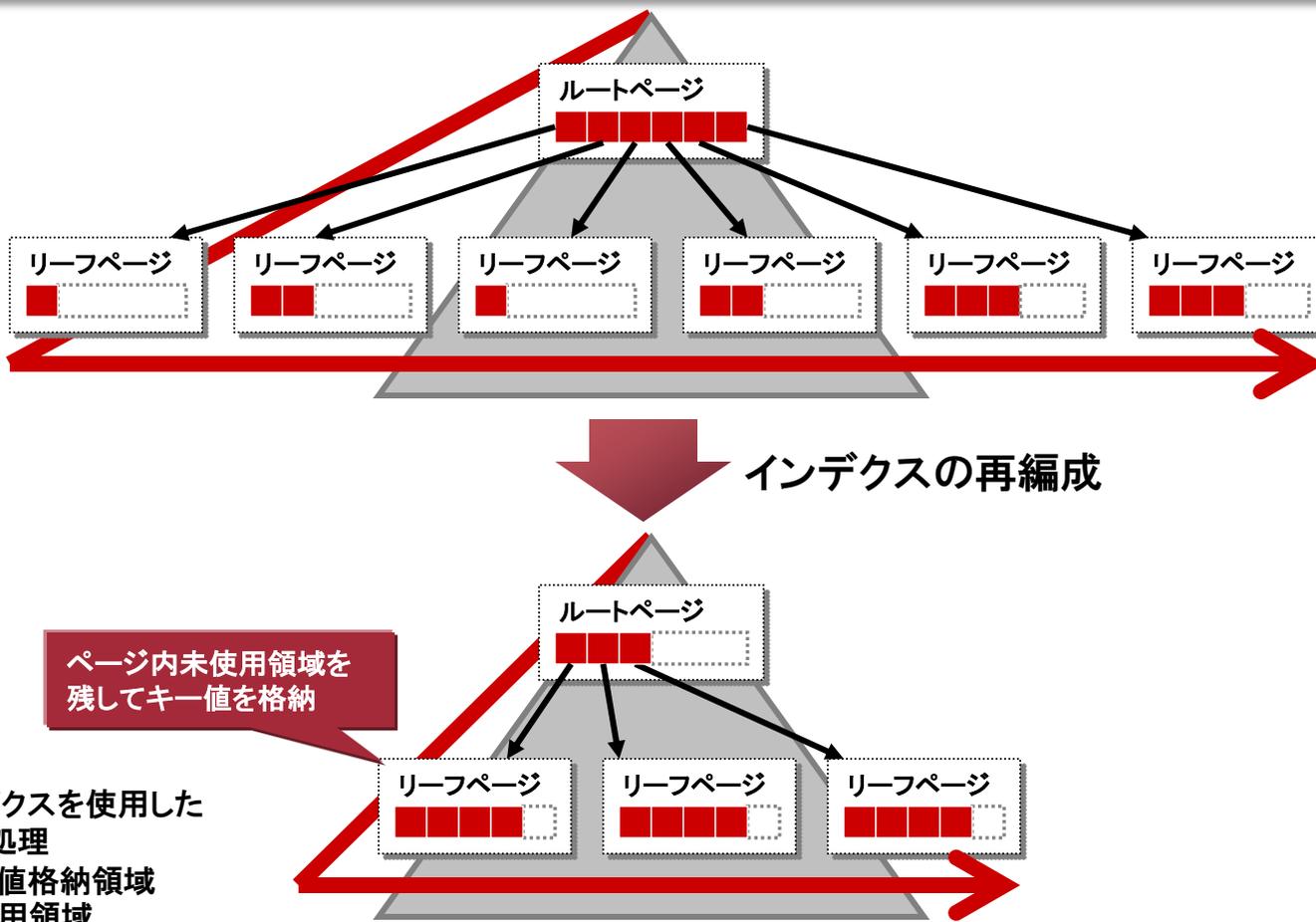
※BLOB列を格納する領域に対して、空きページ解放や
空きセグメント解放は実行できません。

複数ページにまたがる大きさのBLOBデータを
格納する場合、空き領域を再利用すると、
連続したページに格納できず、業務性能に
影響を及ぼす可能性があります。

6-5-5 データベース再編成 インデクスの再編成

解説

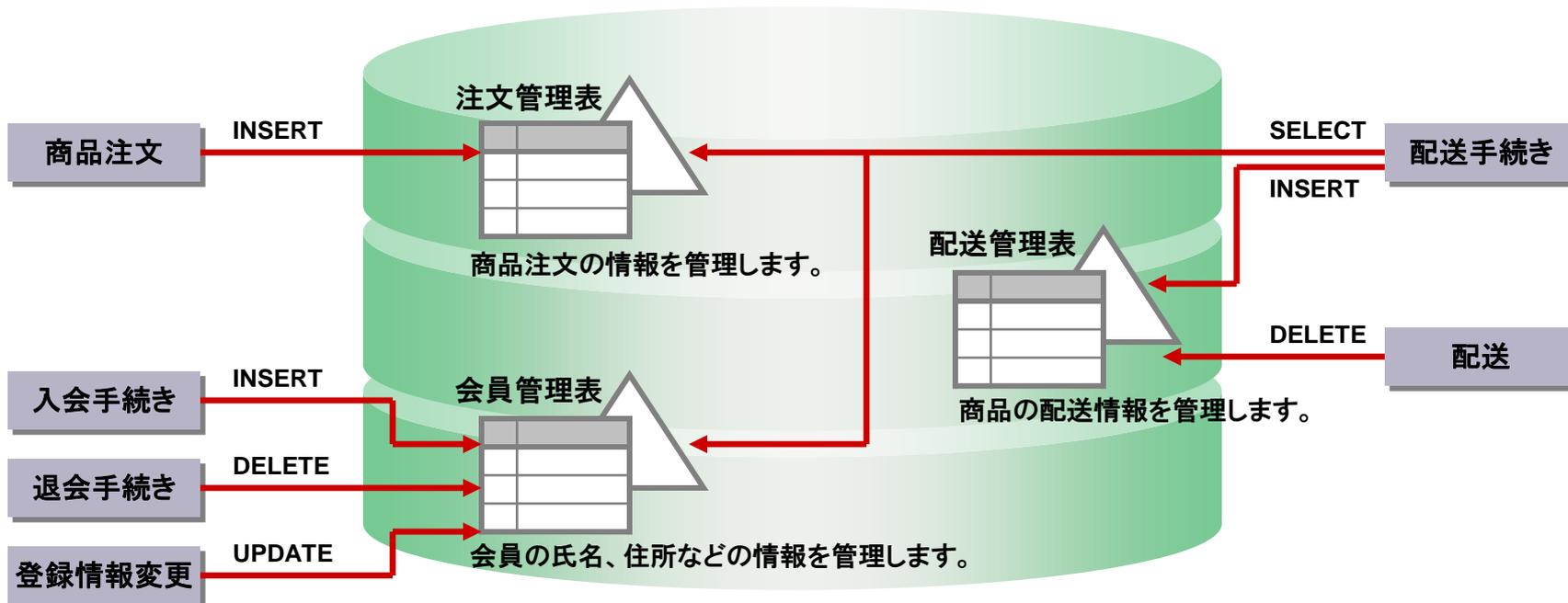
インデクスの再編成では、インデクス定義時のインデクスオプションPCTFREEで指定したページ内未使用領域比率にしたがい、キー値を格納します。データ削除により、キー値の配置が断片的になった場合は、キー値を詰めなおすことで、インデクスの構成ページ数を削減できます。

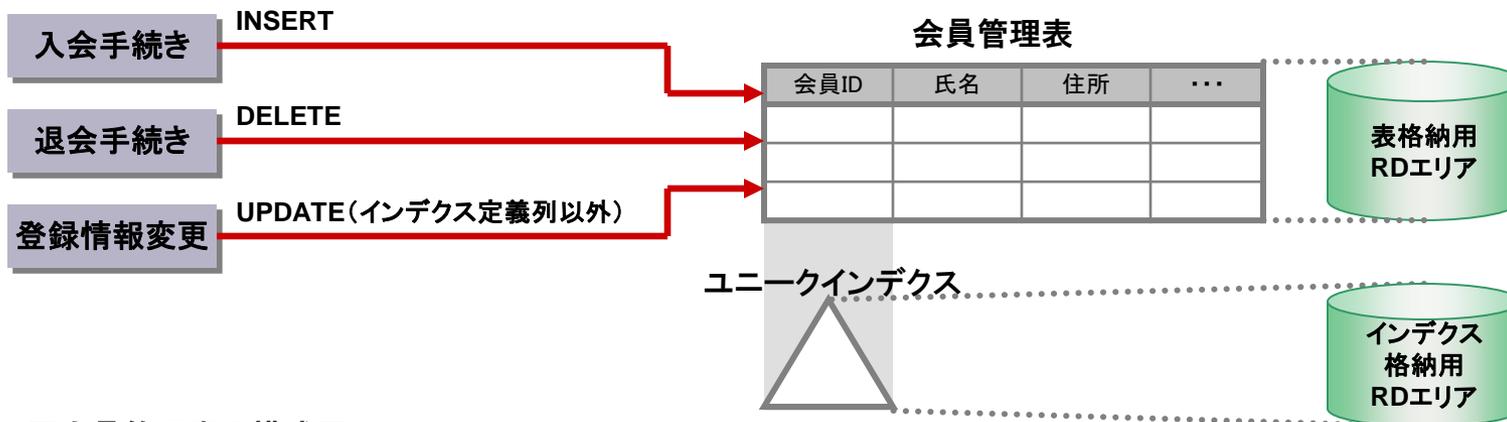


7. 運用事例

解説 商品販売管理システムの例を挙げて、表ごとの再編成運用について説明します。

■会員制の商品販売管理システムの例





■会員管理表の構成■

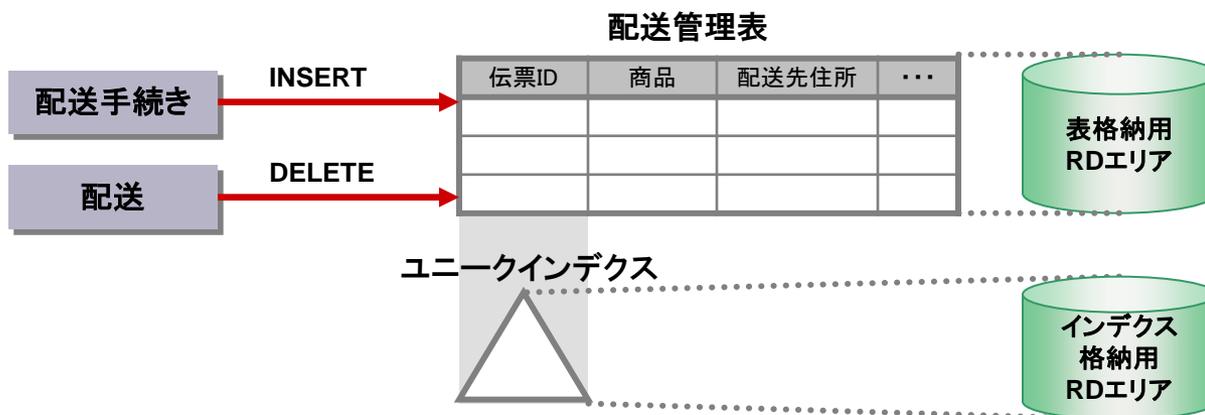
- ◆会員の氏名、住所などの情報を管理する。
- ◆会員ID列にユニークインデクスを定義。
- ◆表とインデクスは、別のRDエリアに格納する。

会員管理表の特長

- ◆入会手続き、退会手続きの業務において、データの追加および削除が発生する。
削除するデータの順序などは規則的ではない。
⇒空き領域を再利用する運用には向いていない。
- ◆住所変更など、登録情報変更の業務において、可変長データの更新が発生する。
⇒定期的にデータベース再編成を実行する必要がある。

会員管理表の再編成運用

- ◆定期的にデータベース再編成を実行する運用とする。
- ◆会員数の増加に対しては、RDエリアを拡張する運用とする。RDエリアの使用率を通知するメッセージを監視する。
- ◆インデクスについては、表の再編成の延長で再編成する。



■ 配送管理表の構成 ■

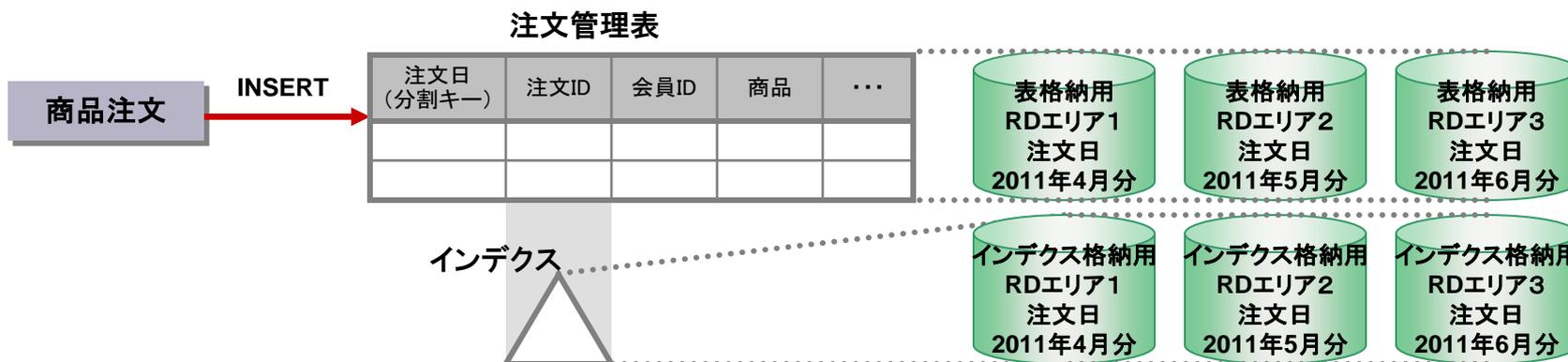
- ◆ 各注文の配送する商品、配送先住所などを管理する。
- ◆ 伝票IDには、昇順の値が登録される。
- ◆ 伝票ID列にユニークインデクスを定義。
- ◆ 表とインデクスは、別のRDエリアに格納する。

配送管理表の特長

- ◆ 配送手続き、配送の業務において、データの追加および削除が発生する。配送手続きで格納したデータは、配送が完了した段階(数日後)で必ず削除されるため、連続した空き領域が発生する。
⇒ 空き領域を再利用する運用に向いている。
- ◆ データの更新は発生しない。

配送管理表の再編成運用

- ◆ 表に対してはSEGMENT REUSEを使用する。
- ◆ インデクスについては、定期的に空きページ解放を実行する。これは、ユニークインデクスの残存エントリを解放することと、使用セグメント数の増加を抑えることが目的。



■ 注文管理表の構成 ■

- ◆ 注文した会員情報や商品の情報など、商品注文に関する情報を管理する。
- ◆ 注文IDには、昇順の値が登録される。
- ◆ ハッシュ関数HASH0^(*)を使用した表の横分割を行い、月ごとに格納するRDエリアを循環利用する運用とする。
- ◆ 過去3ヶ月分のデータを保持するため、格納先RDエリア数を3としている。
- ◆ 注文ID列にインデクスを定義。インデクスも表と同様に横分割する。
- ◆ 表とインデクスは、別のRDエリアに格納する。

(*1): ハッシュ関数HASH0を使用すると、分割列中の年月の値をハッシングし、データを格納するRDエリアを月単位で循環させて割り当てる横分割表を定義できます。

注文管理表の特長

- ◆ 商品注文の業務において、データの追加が発生する。
- ◆ 月ごとに格納するRDエリアを循環利用する運用とし、3ヶ月分の注文履歴を保管する。
- ◆ 保管期間内のデータは削除しない。

注文管理表の再編成運用

- ◆ 保管期間の過ぎたデータは削除し、そのデータを格納していたRDエリアには、次の月のデータを格納する。RDエリアを循環利用する場合、保管期間の過ぎたデータの削除方法については、次スライドを参照のこと。

解説 特定のRDエリアに格納されたデータを削除するためには、データベース作成ユーティリティ (pdload) の作成モードで、対象のRDエリアに0件のデータをデータロードする方法があります。pdloadの作成モードとは、既存のデータを削除してから、入力データを格納する方法です。



0件データロード



— : データ格納領域
- - - : 未使用領域

■0件データロードの実行例■

```
pdload      : データベース作成ユーティリティ
-d          : 作成モード
-i s        : インデクス作成方法に更新モードを指定(*1)
注文管理表 : 表識別子
/HiRDB/control_file/load : 制御文ファイル名
```

(*1): インデクス作成方法に更新モードを指定することで、インデクス一括作成処理を抑止できます。

■制御文ファイルの内容(/HiRDB/control_file/load)■

```
source RDAREA1 /HiRDB/DATA/DATA0
: 対象RDエリア名と入力データファイル(0バイトファイル)
の名称を指定
```

作成モードのデータロードでは、既存のデータを削除した時点で、処理対象の表が使用していたセグメントは未使用セグメントになります。

8. トラブル事例

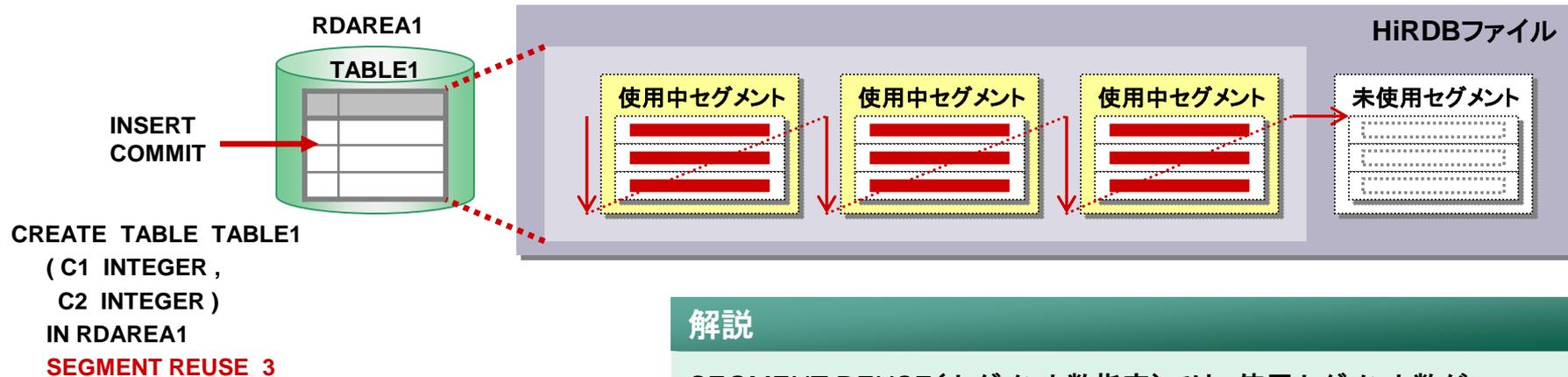
データベース再編成に関する設計や運用方法を誤ると、性能劣化などのトラブルに陥ることがあります。この章では、トラブル事例について紹介します。

現象

データの追加を継続していると、ある時点からデータ格納処理性能が遅延し始めた。

原因

データを削除しない(空き領域が存在しない)表であるにもかかわらず、SEGMENT REUSE (セグメント数指定)を使用していたため、使用するセグメント数が指定した値を超えた段階で新たな未使用セグメントを確保する前に、空き領域を探す処理が動作していた。



- : 空き領域を探す処理
- : データ格納領域
- : 未使用領域

解説

SEGMENT REUSE(セグメント数指定)では、使用セグメント数が指定値に達した時点で、新たに未使用セグメント確保するのではなく、先頭のセグメントから空き領域を探します。ただし、使用中セグメント内に空き領域が存在しなかった場合は、新たに未使用セグメントを確保し、データを格納します。これ以降、新たな未使用セグメントを確保する前には、必ず先頭からすべての使用中セグメントに対して、空き領域を探す処理が動作します。

注意事項 ①

データを削除しない表に対して、SEGMENT REUSEは使用しないでください。

SEGMENT REUSEは空き領域を再利用するための機能です。
空き領域が存在しない表に対しては、使用しないでください。

注意事項 ②

SEGMENT REUSEのセグメント数に、有効なデータ量を格納するためのセグメント数よりも少ない値を指定した場合も、同様の現象となります。

SEGMENT REUSEのセグメント数は、有効なデータ量を見積もった上で設定してください。

注意事項 ③

SEGMENT REUSEを指定した表に対して、データを大量削除した後に、別のデータを挿入するという処理を行う場合、大量削除後に一旦commitしてください。

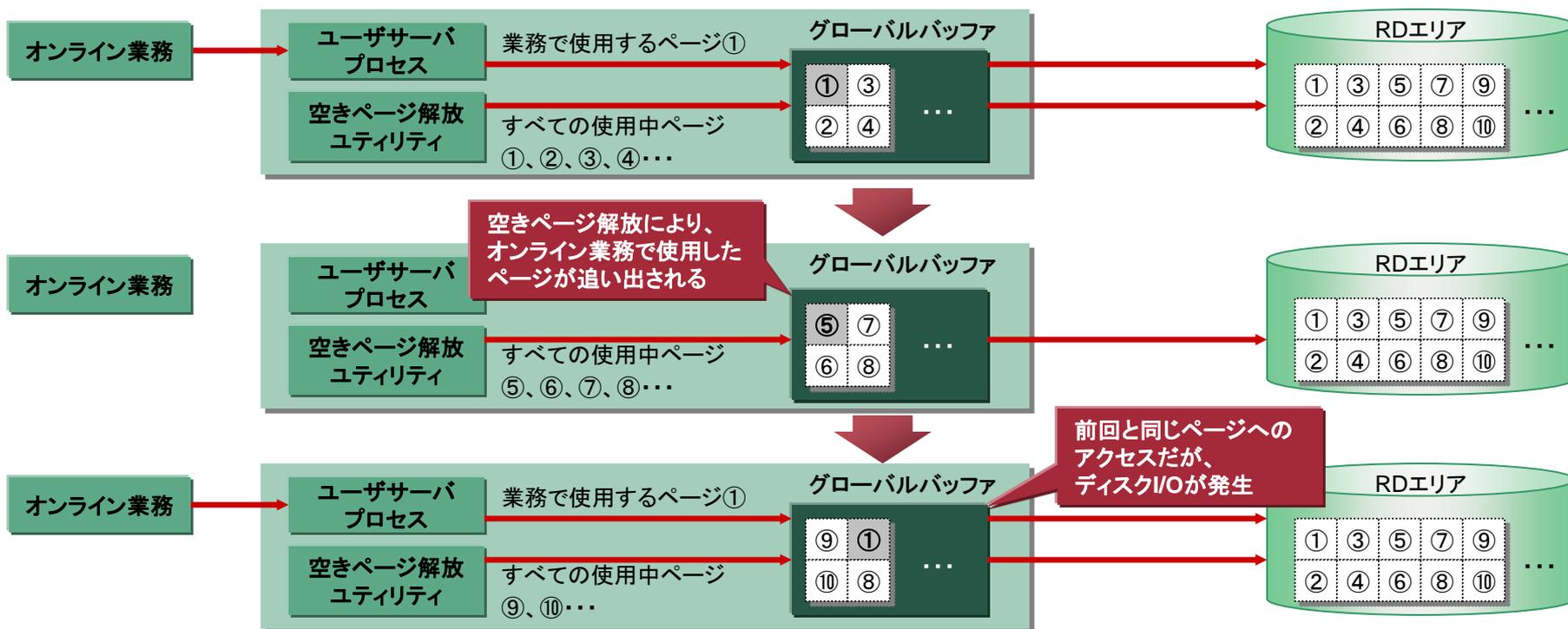
同一トランザクション内の削除により、空き状態となったページは、再利用の対象になりませんので、ご注意ください。

現象

オンライン業務と並行して空きページ解放ユティリティ(pdreclaim)を実行したところ、オンライン業務性能が遅延した。

原因

空きページ解放ユティリティでは、すべての使用中ページを参照するため、オンライン業務で使用するページをグローバルバッファ上から追い出してしまう、オンライン業務の延長で発生するディスクI/O回数が増加したことで、業務性能に影響を及ぼした。



注意
事項
①

空きページ解放と業務アプリケーションを並行実行する場合は、トラフィックの低い時間帯に実行してください。

注意
事項
②

空きページ解放の実行間隔について、検討してください。
空きページ解放の効果が低いにもかかわらず、頻繁に実行することは避けてください。
空きページ解放ユティリティの効果(解放したページ数など)は、ユティリティの実行結果で確認できます。

現象

RDエリアの初期サイズを格納データ量よりも大きく作成してしまったため、RDエリアを縮小して、別のRDエリアにディスク容量を割り当てようとしたが、この作業を行うためには、業務を停止する必要があった。

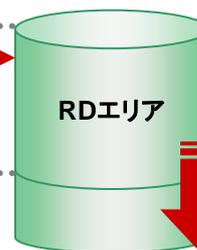
時間

■RDエリアを拡張する場合

オンライン業務



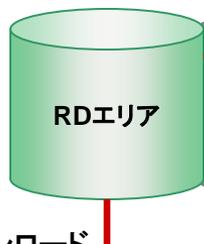
- ◆データベース構成変更ユーティリティ
RDエリアの拡張
- ◆RDエリアの自動増分



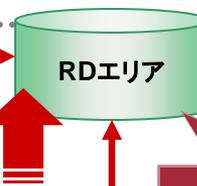
業務と並行実行可能

■RDエリア(初期サイズ)を縮小する場合

オンライン業務



- ②データベース構成変更ユーティリティ
RDエリアの再初期化
(with reconstruction 指定)



オンライン業務

業務停止中に実行可能

①データをアシロード



③データをリロード

注意事項

RDエリアの初期サイズは、データ容量を見積もった上で、検討してください。

他社所有名称に対する表示

- AIXは、米国およびその他の国におけるInternational Business Machines Corporationの商標です。
- SAP BusinessObjectsは、ドイツのSAP AGまたはSAP関連会社の商標または登録商標です。
- CORBAは、Object Management Groupが提唱する分散処理環境アーキテクチャの名称です。
- ER/Studioは、米国Embarcadero Technologies, Inc.の登録商標です。
- HP-UXは、Hewlett-Packard Companyのオペレーティングシステムの名称です。
- IBMおよびCognosは、米国およびその他の国におけるInternational Business Machines Corporationの商標です。
- Java, J2EE, Solaris及びJDBCは、Oracle Corporation 及びその子会社、関連会社の米国 及びその他の国における登録商標または商標です。
- Linuxは、Linus Torvalds氏の日本およびその他の国における登録商標または商標です。
- Microsoft .NET は、お客様、情報、システムおよびデバイスを繋ぐソフトウェアです。
- Microsoft Accessは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- Microsoft OfficeおよびExcelは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- Microsoftは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- ODBCは、米国Microsoft Corporationが提唱するデータベースアクセス機構です。
- OLEは、米国Microsoft Corporationが開発したソフトウェア名称です。
- QlikViewは、QlikTech International AB.の登録商標です。
- SAP, および本文書に記載されたその他の SAP 製品、サービス、ならびにそれぞれのロゴは、ドイツおよびその他の国々における SAP AG の商標または登録商標です。
- UNIXは、The Open Groupの米国ならびに他の国における登録商標です。
- VERITASおよびVERITASロゴは、Symantec Corporation の米国およびその他の国における商標または登録商標です。
- Windows Serverは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- Windows Vistaは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- Windowsは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です。
- X/Openは、The Open Groupの英国ならびに他の国における登録商標です。
- その他記載の会社名、製品名は、それぞれの会社の商標もしくは登録商標です。

END

HiRDB

**HiRDBデータベース再編成
運用方法の解説**

2011/08

株式会社 日立製作所 情報・通信システム社
ITプラットフォーム事業本部 開発統括本部 DB設計部

Human Dreams. Make IT Real.

私たちは、ITと制御技術、そして社会インフラシステムで
人々の夢をかなえるイノベーションを起こしていきます。

HITACHI
Inspire the Next