
HiRDB技術解説
HiRDB設計と構築のポイント

2019/9

株式会社 日立製作所 サービス&プラットフォームビジネスユニット
サービスプラットフォーム事業本部 DB部



Contents

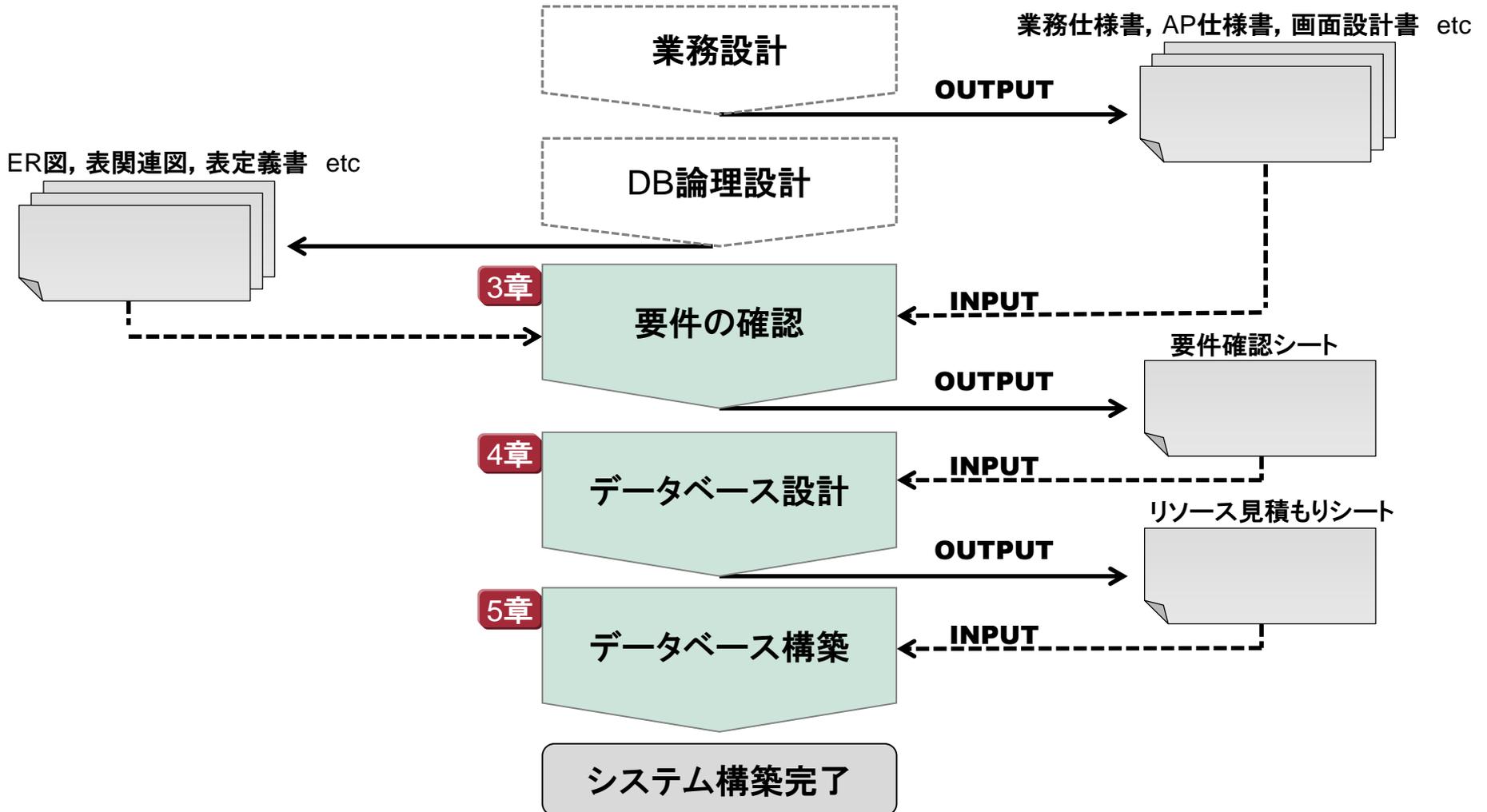
1. はじめに
2. HiRDBのアーキテクチャ
3. 要件の確認
4. データベース設計のポイント
5. データベース構築のポイント
6. 知っておくと役立つポイント
7. おわりに



1. はじめに

1-1 DB構築までの流れ

本資料では、要件の確認からデータベース構築までを解説します。



◆2章:HiRDBのアーキテクチャ

学習する上で基本となるHiRDBのアーキテクチャについて解説します。

◆3章:要件の確認

HiRDBシステムを構築する前に、事前に確認しておく要件について解説します。

◆4章:データベース設計のポイント

ファイル配置、リソース容量見積もりの方法や考え方のポイントについて解説します。

◆5章:データベース構築のポイント

サーバ・ファイル環境の決定、パラメタ設定、構築方法のポイントについて解説します。

◆6章:知っておくと役立つポイント

要件に応じたパラメタ設定や各種設定ファイルを準備する作業を効率的に行う方法について解説します。

2. HiRDBのアーキテクチャ

2.1 概要

2.2 HiRDBの構成要素

2.3 HiRDBファイルの物理構造

この章では、HiRDBの構成要素とそれらの働き、データベース構造などについて説明します。

2. HiRDBのアーキテクチャ

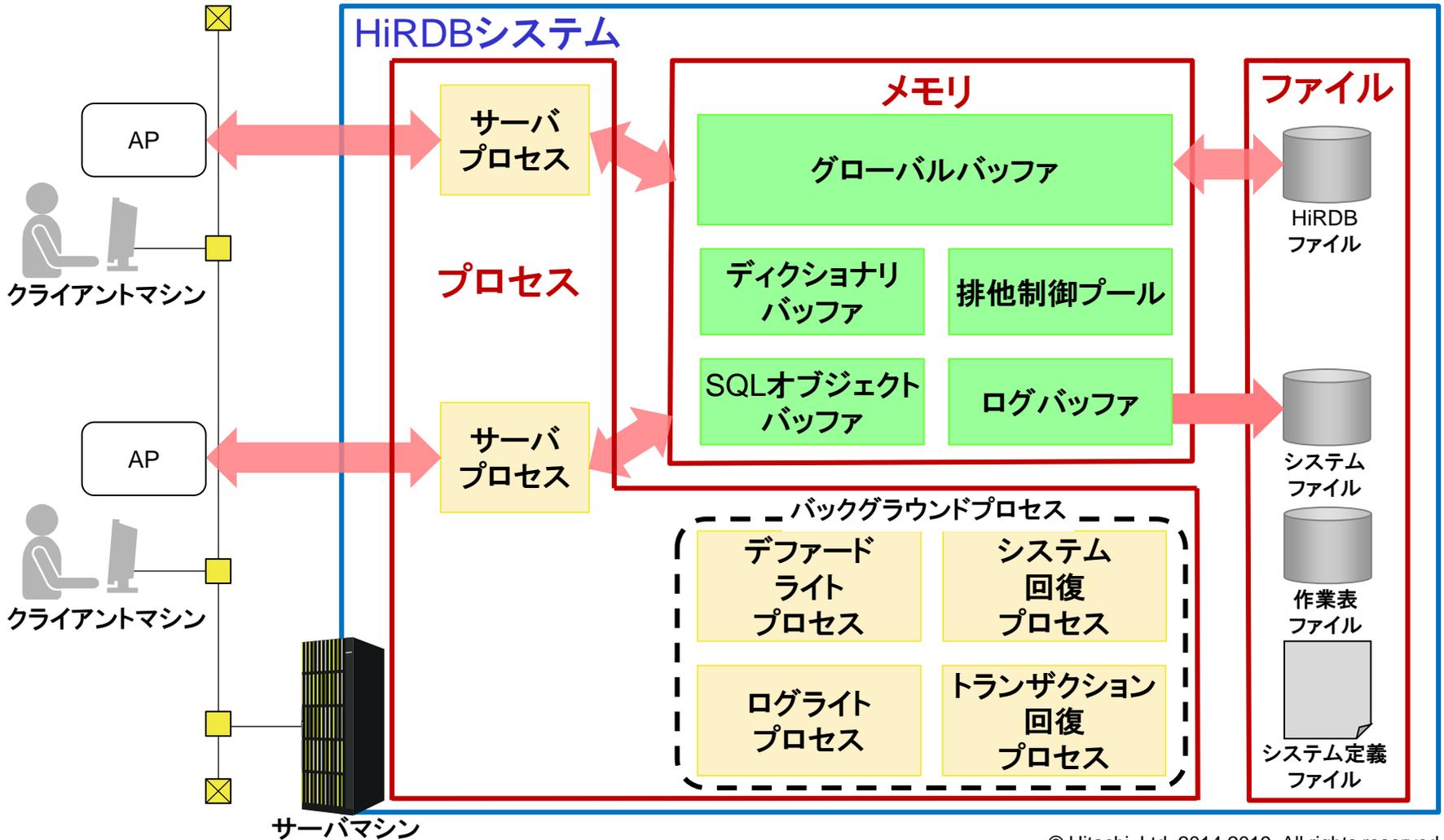
2.1 概要

2.2 HiRDBの構成要素

2.3 HiRDBファイルの物理構造

2-2-1 HiRDBの構成要素

解説 HiRDBは、プロセス、メモリ、ファイルから構成されており、これらをまとめてHiRDBシステムといいます。



解説

プロセスは、アプリケーションからの要求を処理したり、システムを稼働するために実行されるプログラムです。HiRDBIには、サーバプロセスとバックグラウンドプロセスがあります。

HiRDBシステム

サーバ
プロセス

プロセス

サーバ
プロセス

プロセス	説明
サーバプロセス	アプリケーションからSQLを受け付けて実行します。 一つのアプリケーションに対して、一つのサーバプロセスが起動します。
バックグラウンドプロセス	主に、次の処理を行うプロセス群です。 <ul style="list-style-type: none"> ・データベースの更新内容をファイルに出力する ・システムログをファイルに出力する ・システム障害からデータベースを回復する ・トランザクション障害からデータベースを回復する

バックグラウンドプロセス

デフォード
ライト
プロセス

システム
回復
プロセス

ログライト
プロセス

トランザクション
回復
プロセス

解説

SQLの解析と実行に必要なデータは、一時的にメモリ上に格納されます。HiRDBが使用するメモリの種類を次の表に示します。

メモリ

グローバルバッファ

ディクショナリ
バッファ

排他制御プール

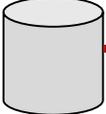
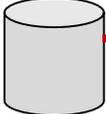
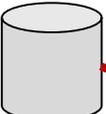
SQLオブジェクト
バッファ

ログバッファ

プロセス	説明
グローバル バッファ	データ入出力時にデータを格納します。 グローバルバッファには、表やインデクスのデータ入出力時に使用するデータバッファと、SQL解析時に必要な情報の入出力時に使用するシステムRDエリアバッファがあります。
ディクショナリ バッファ	SQLの解析時に必要な定義や情報を格納します。
SQLオブジェクト バッファ	解析したSQLオブジェクトを格納します。
排他制御プール	排他情報(対象となる排他資源、排他モードなど)を格納します。
ログバッファ	システムログを一時的に格納します。

解説 ファイルには、HiRDBのデータベースを構成するファイルと、HiRDBの実行環境を定義するファイルがあります。

ファイル ■ データベースを構成する主なファイル HiRDBファイルシステム領域上に作成

ファイル	説明
 HiRDB ファイル	HiRDB ファイル 表やインデクスのデータを格納するファイルです。 表やインデクスの定義情報も格納します。
 システム ファイル	システム ファイル 次の3種類のファイルを総称して、システムファイルといいます。 <ul style="list-style-type: none"> ・ システムログファイル データベースのすべての更新履歴(ログ)が記録されます。 ・ シンクポイントダンプファイル システム障害から回復する場合、システムログファイルのどの時点から回復すればよいかという情報が、シンクポイント※ごとに記録されます。 ・ ステータスファイル HiRDBがどのような状態(ファイルのオープン/クローズなど)であるかが記録されます。システム障害から回復する場合、この情報を基に障害発生前の状態に戻します。
 作業表 ファイル	作業表 ファイル 表を結合したり、データをソートするときに使用する情報を一時的に格納するファイルです。

■ 実行環境を定義するファイル OSのファイルシステム上に作成

ファイル	説明
 システム 定義 ファイル	システム 定義 ファイル HiRDBを開始する場合に必要な、HiRDBの実行環境や構成を定義するファイルです。 (例:ファイル構成、ファイル名称、バッファの大きさ、サーバプロセス数など) このファイルはテキストファイルです。ファイルを開いて、データベース管理者が直接編集します。

※ シンクポイントとは、メモリの更新内容とディスクの内容を一致させるシステムの同期点のことです。

2. HiRDBのアーキテクチャ

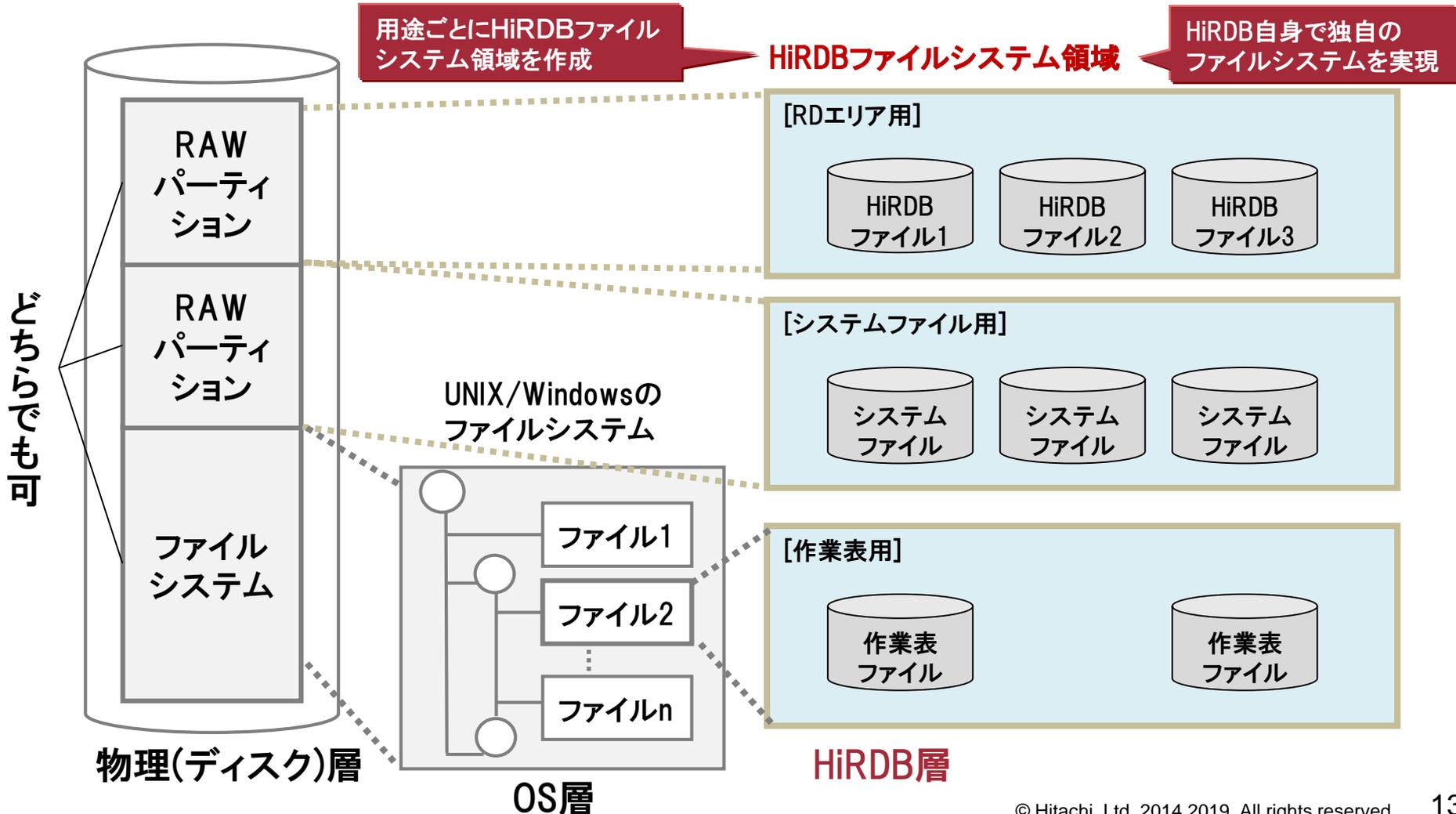
2.1 概要

2.2 HiRDBの構成要素

2.3 HiRDBファイルの物理構造

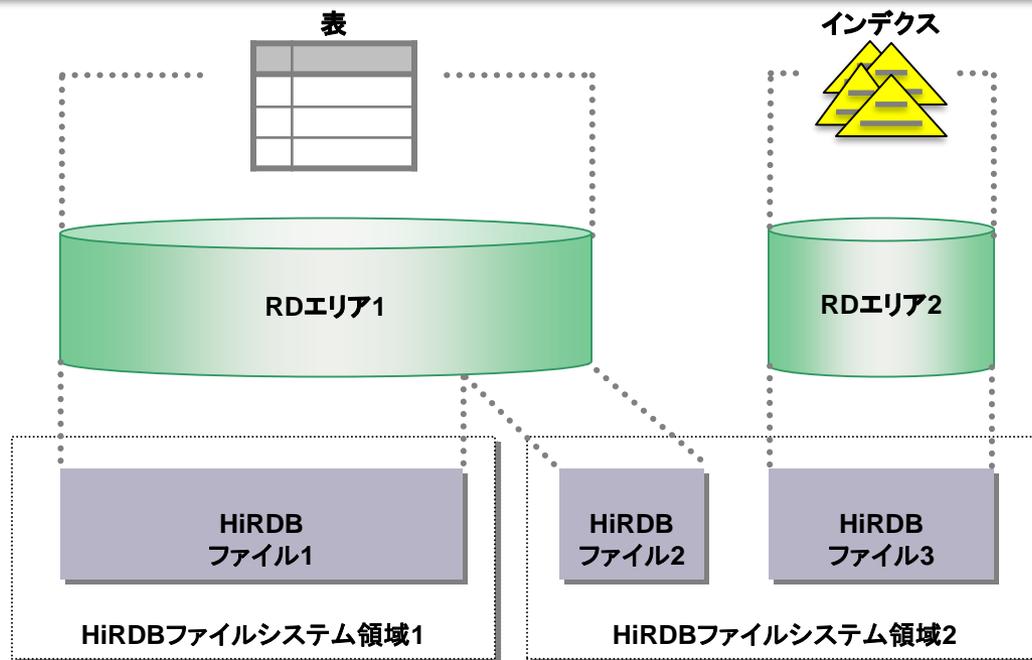
2-3-1 HiRDBファイルシステム領域

解説 ディスクのRAWパーティションまたはOSのファイルシステム上にHiRDBファイルシステム領域を作成し、その中にHiRDBファイル・システムファイル・作業表ファイルを作成します。



解説

RDエリアは、表やインデクスを格納するための、論理的な単位です。RDエリアは、複数のHiRDBファイルから構成されます。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。



RDエリアの最大サイズ

最大1TBのRDエリアを構築することができます。
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。64GB×16=1TB になります。

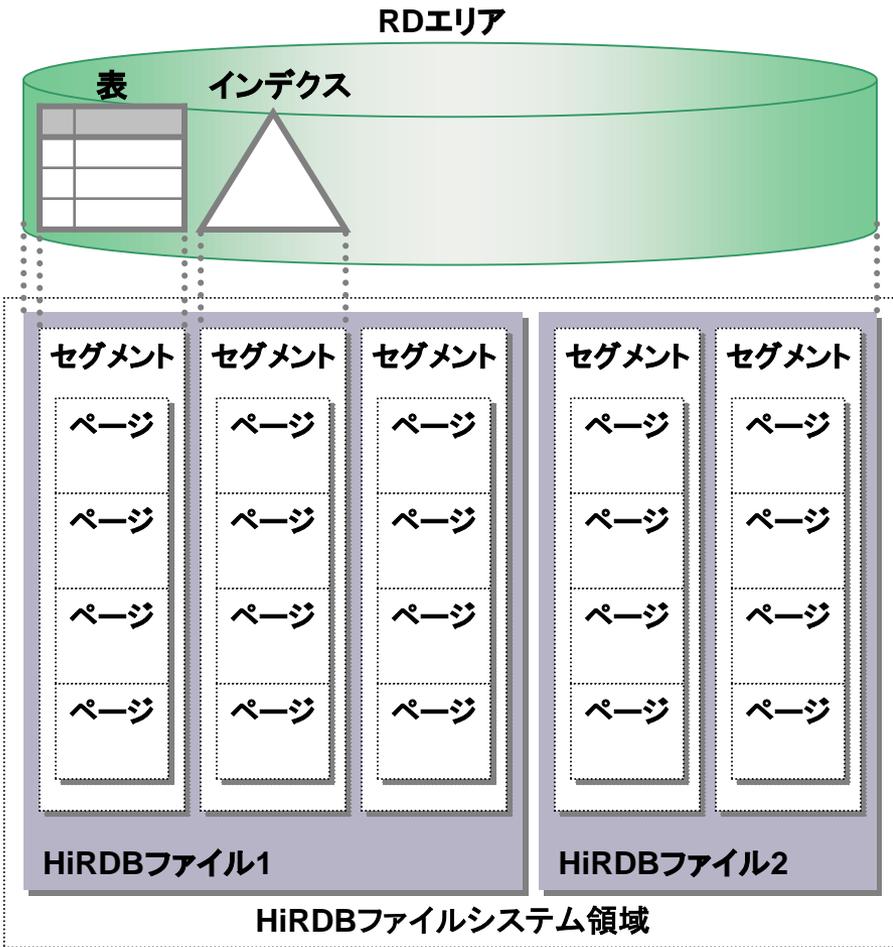
2-3-3 RDエリアの種類

解説 RDエリアの種類を次に示します。

RDエリアの種類		主な管理対象	作成基準
システムRDエリア	マスタディレクトリ RDエリア	HiRDBファイルシステム領域の構成、パスなど	必須
	データディレクトリ RDエリア	RDエリアがどのHiRDBファイルシステム領域にあるかという情報など	必須
	データディクショナリ RDエリア	ディクショナリ表(データディクショナリや、表やインデクスがどのRDエリアにあるかという情報など)	必須
	データディクショナリLOB RDエリア	ストアドプロシジャ、ストアドファンクションのSQL情報	任意
ユーザRDエリア	ユーザRDエリア	表やインデクスを格納	必須
	ユーザLOB RDエリア	文書、画像、音声などの長大な可変長データを格納	任意

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。

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。

3. 要件の確認

3.1 概要

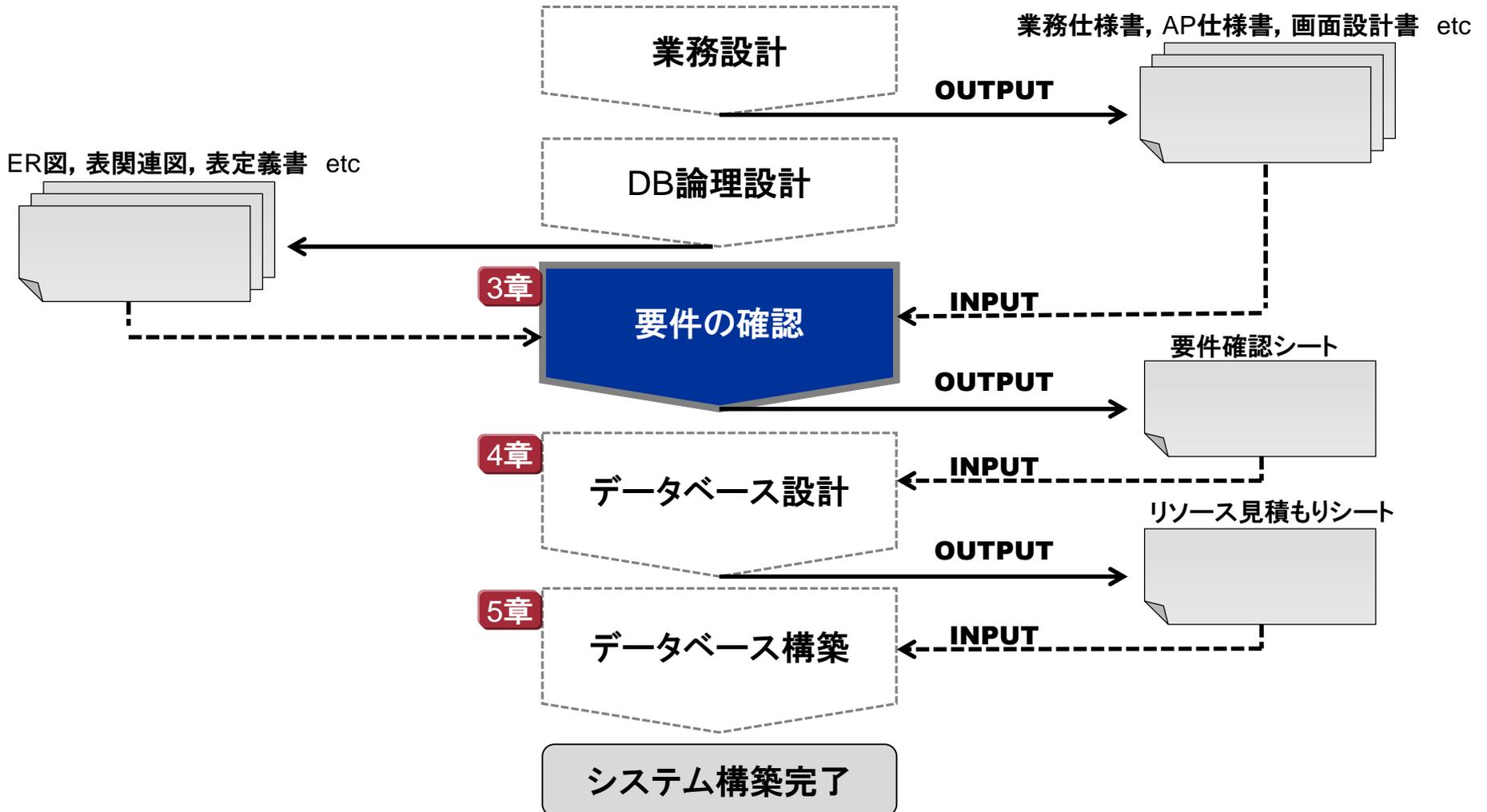
3.2 要件確認項目

3.3 業務要件

3.4 性能要件

3.5 運用要件

解説 本章では要件の確認について説明します。



3. 要件の確認

3.1 概要

3.2 要件確認項目

3.3 業務要件

3.4 性能要件

3.5 運用要件

3-2-1 要件確認項目(1)

解説 確認する項目をまとめた、要件確認シートを次に示します。

カテゴリ	確認項目	単位	変数	確認結果
業務要件(論理設計した結果)	定義する表の数	個	t-cnt	
	各表の列数	列	c-cnt_A	
			c-cnt_B	
			:	
	各表のデータ件数	件	d-cnt_A	
			d-cnt_B	
			:	
	各表の行長	バイト	rowleng_A	
			rowleng_B	
			:	
	定義するインデックスの数	個	i-cnt	
	各インデックスのキー長	バイト	idxleng_A	
			idxleng_B	
			:	
	各インデックスのキー種類数	種類	keycnt_A	
keycnt_B				
:				
各インデックスの重複数	件/キー	cntperkey_A		
		cntperkey_B		
		:		

3-2-2 要件確認項目(2)

カテゴリ	確認項目	単位	変数	確認結果
性能要件	トランザクションごとの更新量の最大値	バイト	max_trn_size	
	トランザクションごとの処理行数の最大値	行	max_suid_cnt	
	1日あたりの平均トランザクション数	件	tpd	
	最大同時接続数	本	maxusers	
運用要件	計画停止時間	時間	stptime	
	RPO(Recovery Point Objective):リカバリポイント目標	時間	rpo	
	RTO(Recovery Time Objective):リカバリ時間目標	時間	rto	

3. 要件の確認

3.1 概要

3.2 要件確認項目

3.3 業務要件

3.4 性能要件

3.5 運用要件

解説 データベース論理設計の成果物である表一覧およびインデクス一覧から次の項目を確認します。これらは、**ユーザRDエリア、作業表ファイルの容量に関係します。**

項番	確認項目	内容
1	定義する表の数	定義する表の数を確認します。
2	各表の列数	各表の列数を確認します。
3	各表のデータ件数	データ保有期間内で、想定されるデータ件数の最大値を表ごとに求めます。
4	各表の行長	行長は、各列の長さの合計値です。列の長さは、各列のデータ型の長さから求めます。可変長列の場合は平均の長さを列の長さとし、各データ型の長さについては、マニュアル「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」-「データ長一覧」を参照してください。
5	定義するインデクスの数	インデクスの数には、自動的に作成されるインデクス(例えば、主キー)も含まれます。
6	各インデクスのキー長	インデクスキー長は、インデクスが定義された列の長さを合算し、インデクスが定義された列数を加えた値です。
7	各インデクスのキー種類数	各インデクスのキー種類数とは、インデクスを作成した列に何種類のデータが格納されているのかのことで、ユニークインデクスの場合は表のデータ件数と等しくなります。
8	各インデクスの重複数	<p>インデクスの重複数とは、インデクスキーごとのデータ件数の平均であり、以下の式により求めることができます。</p> <div style="border: 1px solid black; padding: 5px; display: inline-block; margin: 10px 0;"> $\frac{\text{表のデータ件数}}{\text{インデクスキー種類数}}$ </div> <p>ユニークインデクスの場合は、インデクスの重複数は 1 となります。</p>

解説 ベンダーに対して製品の発注を行う「発注システム」のシステム要件を示します。

■業務要件

1. 発注担当者は、ベンダーに対して発注を行う
2. 1日あたり平均50件発注する
3. 1回で、平均2種類の製品を発注する
4. 1回で、最大20種類の製品を発注可能とする
5. 毎月、月末に発注レポートを作成する
6. 発注レポートには、発注NO、発注日付、発注先のベンダー名、発注した製品名、発注数量を表示する
7. データ保有期間は2年間とする
8. 2年間で、500件のベンダー情報を登録する
9. 2年間で、3000種類の製品情報を登録する
10. 登録されたベンダーは、1件以上製品情報を登録している
11. 登録されたベンダーに対して、1件以上発注している

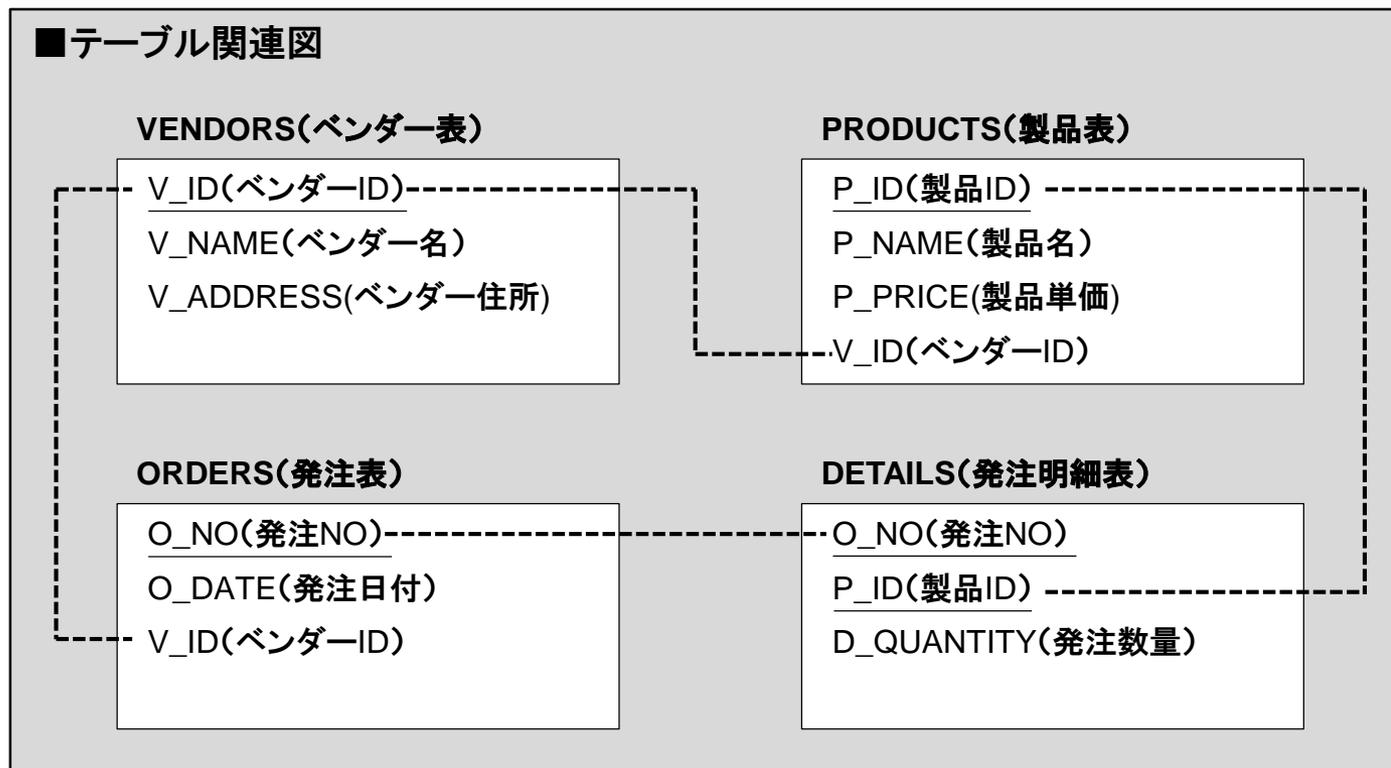
■性能要件

1. ピーク時は、1分間あたり最大6件の発注がある
2. DBMSへの最大同時接続数を10とする

■運用要件

1. 1日4時間はメンテナンス時間とする
2. 各種障害が発生しても、コミットされたトランザクション内のデータ更新を失ってはならない
3. 各種障害発生時、最低2時間以内に回復できなければならない

解説 システム要件から論理設計を行った結果のテーブル関連図、列定義を示します。



3-3-4 例題 列定義

■列定義

VENDORS(ベンダー表)

実装上の列名	データ型	主キー	NOT NULL	制約(参照制約/検査制約)	インデクス
V_ID	CHAR(5)	○	○	—	V_PK_IDX(仮)
V_NAME	CHAR(30)	—	○	—	—
V_ADDRESS	CHAR(30)	—	○	—	—

PRODUCTS(製品表)

実装上の列名	データ型	主キー	NOT NULL	制約(参照制約/検査制約)	インデクス
P_ID	CHAR(5)	○	○	—	P_PK_IDX(仮)
P_NAME	CHAR(30)	—	○	—	—
P_PRICE	INTEGER	—	○	—	—
V_ID	CHAR(5)	—	○	参照制約(VENDORS 表V_ID 列)	P_VID_IDX

ORDERS(発注表)

実装上の列名	データ型	主キー	NOT NULL	制約(参照制約/検査制約)	インデクス
O_NO	INTEGER	○	○	—	O_PK_IDX(仮)
O_DATE	DATE	—	○	—	—
V_ID	CHAR(5)	—	○	参照制約(VENDORS 表V_ID 列)	O_VID_IDX

DETAILS(発注明細表)

実装上の列名	データ型	主キー	NOT NULL	制約(参照制約/検査制約)	インデクス
O_NO	INTEGER	○	○	参照制約(ORDERS 表O_NO 列)	D_PK_IDX(仮)
P_ID	CHAR(5)		○	参照制約(PRODUCTS表P_ID列)	
D_QUANTITY	INTEGER	—	○	—	—

解説 システム要件、テーブル関連図、列定義から確認した確認項目の結果を示します。

項番	確認項目	確認内容	根拠
1	定義する表の数	テーブル関連図より、定義する表の数は4個です。	3-3-3項 テーブル関連図
2	各表の列数	列定義より、VENDORS表は3列、PRODUCTS表は4列、ORDERS表は3列、DETAILS表は3列となります。	3-3-4項 列定義
3	各表のデータ件数	<p>データ保有期間は2年間のため、VENDORS表は500件、PRODUCTS表は3000件となります。ORDERS表は、平均1日あたり50件の発注があるため、データ件数は次の通りになります。また、DETAILS表は、1回の発注で、平均2種類の製品を発注するため、次の通りになります。</p> <ul style="list-style-type: none"> ・ORDERS表: $50[\text{件/日}] \times 365[\text{日}] \times 2[\text{年}] = 36500\text{件}$ ・DETAILS表: $36500[2\text{年間の発注回数}] \times 2[\text{種類/1回の発注}] = 73000\text{件}$ 	3-3-2項 業務要件の項番2、3、7、8、9
4	各表の行長	<p>行長は各列の長さの合計なので、まず各列の長さを求めます。各列の長さは、データ型がCHARの場合は定義長と同じ、INTEGERの場合は4なので、例えばPRODUCTS表の行長は次のようになります。VENDORS表、ORDERS表、DETAILS表についても同様に求めます。</p> <p>$5 + 30 + 4 + 5 = 44$ バイト</p>	3-3-4項 列定義
5	定義するインデックスの数	列定義より、定義するインデックスの数は、主キーを含み6個となります。	3-3-4項 列定義

3-3-6 例題 業務要件詳細まとめ(2)

項番	確認項目	確認内容	根拠
6	各インデクスのキー長	<p>インデクスキー長は、インデクスが定義された列の長さを合算し、インデクスが定義された列数を加えた値となります。例えばVENDORS表の主キーに対するインデクスのキー長は次のようになります。他のインデクスも同様に求めます。</p> <p>5+1=6 バイト</p>	3-3-4項 列定義
7	各インデクスのキー種類数	<p>インデクスのキー種類数は、インデクスを作成した列に何種類のデータが格納されているのかになります。例えばVENDORS表の主キーに対するインデクスは、ユニークインデクスのため、表のデータ件数つまり500がキー種類数となります。また、PRODUCTS表のV_ID上のインデクスのキー種類数はVENDORS表のデータ件数と一致するため、500となります。一致するのは「VENDORS表に登録済みのベンダーは製品情報が1件以上PRODUCTS表に登録されている」ためです。他のインデクスも同様に求めます。</p>	項番3の表のデータ件数と 3-3-4項 列定義
8	各インデクスの重複数	<p>インデクスの重複数は、インデクスキーごとのデータ件数の平均のため、例えばPRODUCTS表のV_ID列上のインデクスの重複数は3000/500で求められ、6となります。他のインデクスも同様に求めます。</p>	項番3の表のデータ件数と 3-3-4項 列定義

解説 確認項目の結果を要件確認シートにまとめると、次のようになります。

カテゴリ	確認項目	単位	変数	確認結果
業務要件(論理設計した結果)	定義する表の数	個	t-cnt	4
	各表の列数	列	c-cnt_VENDORS	3
			c-cnt_PRODUCTS	4
			c-cnt_ORDERS	3
			c-cnt_DETAILS	3
	各表のデータ件数	件	d-cnt_VENDORS	500
			d-cnt_PRODUCTS	3000
			d-cnt_ORDERS	36500
			d-cnt_DETAILS	73000
	各表の行長	バイト	rowleng_VENDORS	65
			rowleng_PRODUCTS	44
			rowleng_ORDERS	13
			rowleng_DETAILS	13
	定義するインデックスの数	個	i-cnt	6

3-3-8 要件確認シート(業務用件)まとめ(2)

カテゴリ	確認項目	単位	変数	確認結果
業務要件(論理設計した結果)	各インデクスのキー長	バイト	idxleng_V_PK_IDX	6
			idxleng_P_PK_IDX	6
			idxleng_P_VID_IDX	6
			idxleng_O_PK_IDX	5
			idxleng_O_VID_IDX	6
			idxleng_D_PK_IDX	11
	各インデクスのキー種類数	種類	keycnt_V_PK_IDX	500
			keycnt_P_PK_IDX	3000
			keycnt_P_VID_IDX	500
			keycnt_O_PK_IDX	36500
			keycnt_O_VID_IDX	500
			keycnt_D_PK_IDX	73000
	各インデクスの重複数	件/キー	cntperkey_V_PK_IDX	1
			cntperkey_P_PK_IDX	1
			cntperkey_P_VID_IDX	6
			cntperkey_O_PK_IDX	1
			cntperkey_O_VID_IDX	73
			cntperkey_D_PK_IDX	1

3. 要件の確認

3.1 概要

3.2 要件確認項目

3.3 業務要件

3.4 性能要件

3.5 運用要件

解説

性能要件である「トランザクションごとの更新量の最大値」について、解説と求め方を以下に示します。

項番	確認項目	解説／求め方
1	トランザクションごとの更新量の最大値	トランザクションごとの更新系(更新・追加・削除)処理の更新量の最大値を確認します。この値は、 システムログファイルの容量に関係します。

大まかに値を求めるには、まずトランザクションごとに更新・追加・削除する行数をそれぞれ確認し、次の計算式にあてはめます。計算した結果が最も大きい値が、求める値です。

$$((\text{更新行数} \times 2) + \text{追加行数} + \text{削除行数}) \times \text{行長}$$

詳細な値を求める場合は、マニュアル「システム導入・設計ガイド」の「表データ更新時に出力されるシステムログ量」-「基本行ログ量の見積もり」の計算式で求めてください。

解説

性能要件である「トランザクションごとの処理行数の最大値」について、解説と求め方を以下に示します。

項番	確認項目	解説／求め方
2	トランザクションごとの処理行数の最大値	トランザクションごとの処理行数の最大値を確認します。この値は、 排他制御プールの容量に関係します。

この値を求めるには、まずトランザクションごとに参照・更新・追加・削除する行数をそれぞれ確認し、次の計算式にあてはめます。計算した結果が最も大きい値が、求める値です。

参照行数＋更新行数＋追加行数＋削除行数

解説 性能要件である「1日あたりの平均トランザクション数」について、以下に解説します。

項番	確認項目	解説／求め方
3	1日あたりの平均トランザクション数	1日あたりのトランザクション数の平均を確認します。ただし、ここでは、更新系処理をするトランザクションのみを対象とし、参照のみのトランザクションは対象外とします。 更新系処理をするトランザクションはシステムログを出力するため、この数が多いければ、必要とするシステムログファイルの容量も大きくなります。

解説 「最大同時接続数」について、以下に解説します。

項番	確認項目	内容
4	最大同時接続数 (pd_max_users)	HiRDBに同時に接続するアプリケーションの最大数を確認します。この値は、ステータスファイルの容量、作業表ファイル用HiRDBファイルシステム領域の容量、システムRDエリアバッファのページ数、メモリ容量に関係します。

HiRDBコマンドおよびユティリティ、SQL Executer、Datareplicator反映側も同時接続数を消費します。これらを考慮した設計を行って下さい。なお、同時接続数ライセンスの場合は、ご購入頂いたライセンス数以内を設定して下さい。

pd_max_users指定値が大きいと、プロセス固有メモリを多く消費します。HiRDBが消費するメモリサイズの見積もりを行ってください(マニュアル「システム導入・設計ガイド」の「メモリ所要量の計算式」の計算式で求めてください)。

3-4-5 要件確認シート(性能要件)まとめ

解説 確認項目の結果を要件確認シートにまとめると、次のようになります。

カテゴリ	確認項目	単位	変数	確認結果	根拠
性能要件	トランザクションごとの更新量の最大値	バイト	max_trn_size	273	3-3-2項 業務要件の項番4「1回の発注で、最大20種類の製品を発注できる」より、発注処理を行うトランザクションでは、ORDERS表で13バイト追加、DETAILS表に260バイト追加。バッチの特異ケースは除外した。
	トランザクションごとの処理行数の最大値	行	max_suid_cnt	8150	3-3-2項 業務要件の項番5「毎月、月末に発注レポートを作成する」、項番6「発注レポートには、発注NO、発注日付、発注先のベンダー名、発注した製品名、発注数量を表示する」より、発注レポート処理を行うトランザクションでは、最大で、VENDORS表を500行、PRODUCTS表を3000行、ORDERS表を1550行(50[件/日]×31日)、DETAILS表を3100行(1550行×2種類)検索。
	1日あたりの平均トランザクション数	件	tpd	50	3-3-2項 業務要件の項番2「1日あたり平均50件発注する」より
	最大同時接続数	本	maxusers	10	3-3-2項 性能要件の項番2「DBMS への最大同時接続数を10とする」より

3. 要件の確認

3.1 概要

3.2 要件確認項目

3.3 業務要件

3.4 性能要件

3.5 運用要件

解説 運用要件である「計画停止時間」、「RPO／RTO」について、解説します。

項番	確認項目	内容
1	計画停止時間	データベースシステムの計画停止時間を時間単位で確認します。 計画停止時間とは、データベースのバックアップやメンテナンスのために、計画的に業務(サービス)を停止する時間です。
2	RPO／RTO	各種障害からHiRDBを回復する場合を想定して、次の点を確認します。 ■障害発生時、どの時点でデータベースを回復できるか(RPO) ■障害からの回復にどれだけの時間を要するか(RTO) これらは、データベースのバックアップ計画、システムログの運用方法、障害時の回復方法に関係します。

解説 確認項目の結果を要件確認シートにまとめると、次のようになります。

カテゴリ	確認項目	単位	変数	確認結果	根拠
運用要件	計画停止時間	時間	stptime	4	3-3-2項 運用要件の項番1「1日4時間はメンテナンス時間とする」より
	RPO	時間	rpo	0	3-3-2項 運用要件の項番2「各種障害が発生しても、コミットされたトランザクション内のデータ更新を失ってはならない」より
	RTO	時間	rto	2	3-3-2項 運用要件の項番3「各種障害発生時、最低2時間以内に回復できなければならない」より

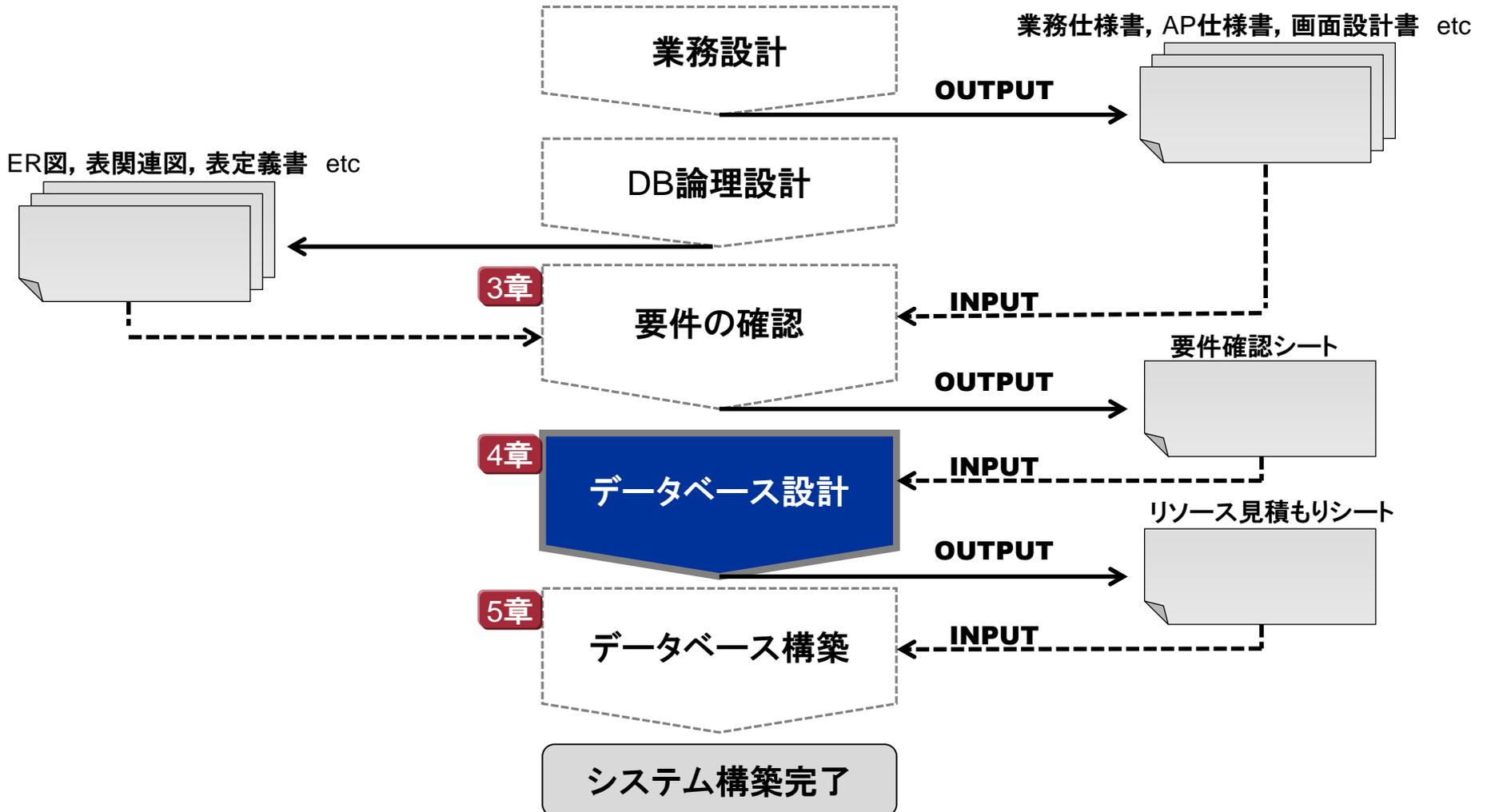
4. データベース設計のポイント

4.1 概要

4.2 データベース設計

4.3 リソース容量見積もり

解説 本章ではデータベース設計について説明します。



解説

データベースを構成するファイルやバッファの配置や容量を業務ごとに見積もり、ディスクやメモリの構成や必要容量を決定します。

■ HiRDBの物理設計の手順とポイント

1. ファイル配置の検討 **4.2節**

- ①HiRDBファイルシステム領域の配置
- ②システムファイルの配置
- ③RDエリアの配置
- ④グローバルバッファの配置

2. ディスク容量見積もり **4.3節**

- ①システムファイルの容量
- ②作業表ファイルの容量
- ③RDエリアの容量
- ④HiRDBファイルシステム領域の容量

3. メモリ容量見積もり **4.3節**

- ①グローバルバッファの容量
- ②排他制御プールの容量

4. データベース設計のポイント

4.1 概要

4.2 データベース設計

4.3 リソース容量見積もり

解説 パフォーマンス、信頼性の観点でのHiRDBのデータベース設計の方針を次に示します。

- パフォーマンスを向上させるためには
 - ディスクが競合しないように、ファイル配置をします。
 - I/O回数が少なくなるようにバッファ配置をします。
- 信頼性を向上させるためには
 - 複数のディスクを用意し、多重化を検討します。
 - 障害の影響が局所化するように配置を検討します。

解説 HiRDBファイルシステム領域の配置の方針を次に示します。

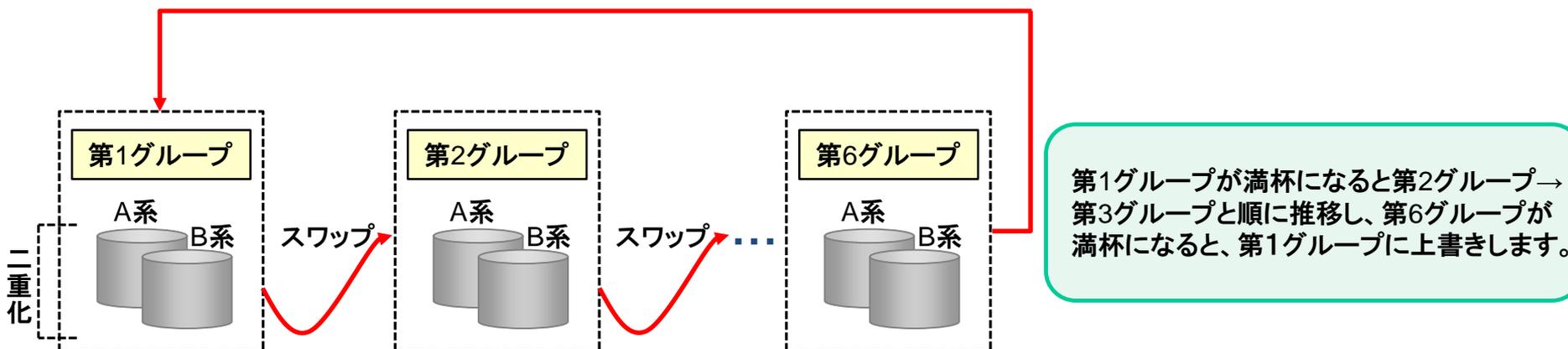
- HiRDBファイルシステム領域は、次に示す用途ごとに作成します。
 - ・RDエリア用
 - ・システムファイル用
 - ・作業表ファイル用
- RDエリア用のHiRDBファイルシステム領域は、システムRDエリア用とユーザRDエリア用をそれぞれ別に作成してください。
- システムファイルを二重化する場合は、異なるディスクに作成してください。
- 作業表ファイルは常に領域を確保する訳ではないため、作業表ファイル用のHiRDBファイルシステム領域はどのディスクに作成しても構いません。

上記を踏まえ、容量のバランスを考慮したHiRDBファイルシステム領域の配置を以下の図に示します。
なお、ここでは、OSはディスクAにインストールされているとします。



解説

システムログファイルは、一つのシステムログファイルが満杯になったら次のシステムログファイルを使用する、というように循環利用しています。そのため、6グループ作成している場合は、次の図のようになります。



システムログファイルが1グループだけの場合、システムログファイルが満杯になり、次に書き込むシステムログファイルがないのでHiRDBは異常終了します。2グループでも運用は可能ですが、一巡して上書きされる前に余裕を持ってアンロード処理(6章)ができるように、システムログファイルは**少なくとも6グループ以上用意してください。**

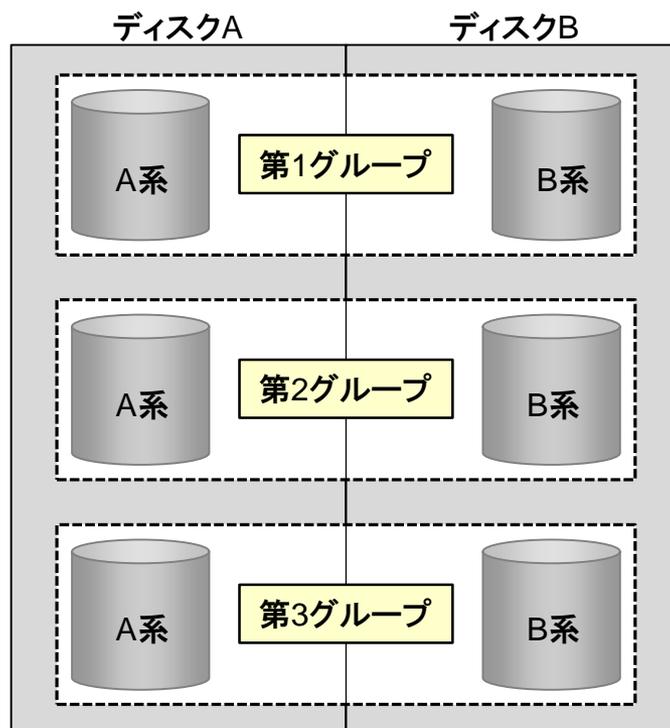
システムファイルは、二重化したファイルをまとめて「グループ(世代)」といい、二重化したそれぞれのファイルをA系、B系といいます。

システムログファイルが満杯になり、次のシステムログファイルを使用することを「システムログファイルのスワップ」といいます。

解説 システムログファイルは、障害回復時に使用する重要なファイルのため、二重化することをお勧めします。

二重化したシステムログファイルには、同じシステムログが書き込まれます。

ディスク障害が発生した場合、正常なディスクにあるシステムログファイルから回復できるため、二重化したシステムログファイルを異なるディスクに作成してください。



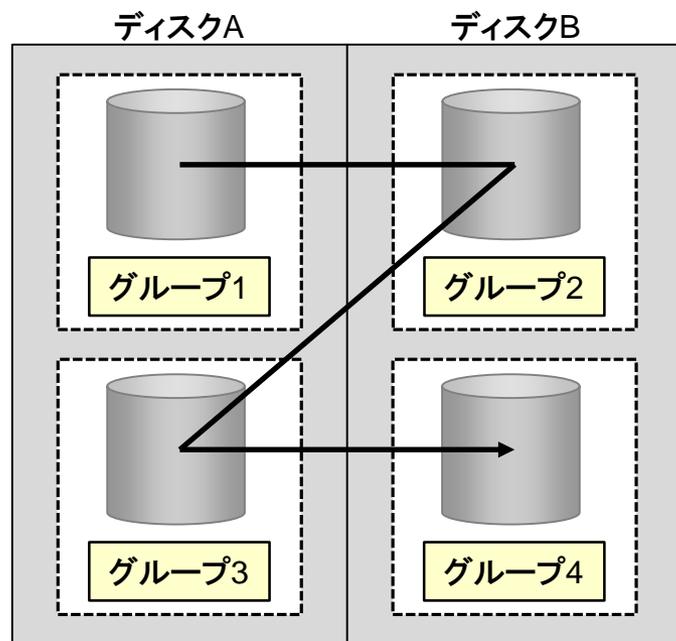
ディスクAに障害が発生しても、ディスクBに同じ内容のシステムログファイルがあるため、障害回復時に使用できます。

解説

ディスク障害に備えて、シンクポイント情報が続けて同じディスクに取られないように配置してください。

シンクポイントダンプ取得時点で実行中の更新系トランザクションがある場合、トランザクションがコミットするまで、取得したシンクポイントダンプファイルは無効です。このトランザクションがコミットすると、シンクポイントダンプファイルは有効になります。障害が起きた時点で有効なシンクポイントダンプファイルを確保するために、シンクポイントダンプファイルは4グループ以上用意します。

二つのディスクに配置する場合は、次の図に示すように、シンクポイントダンプファイルを作成します。



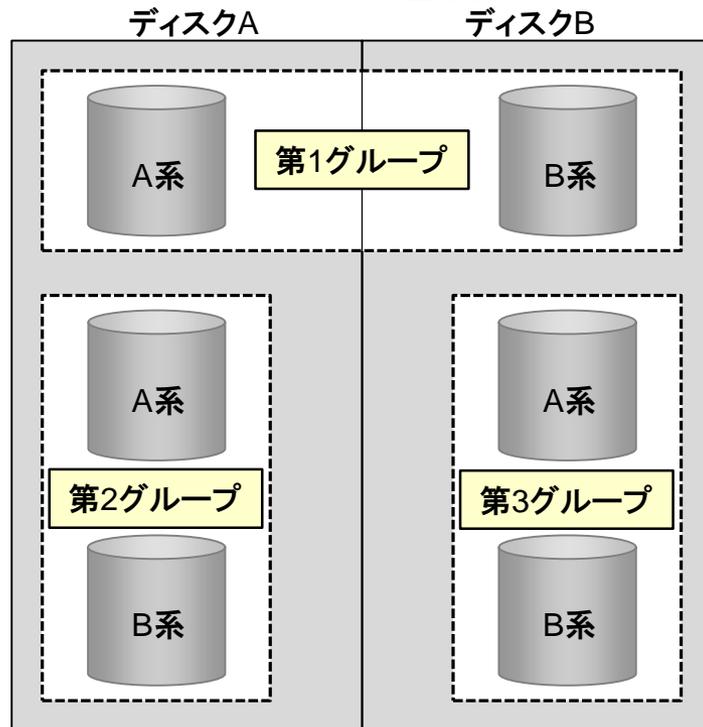
ディスクAに障害が発生しても、ディスクBに2回目、4回目のシンクポイント時の情報があるため、その時点からの回復ができます。4回目のシンクポイントダンプファイルが有効になっていない場合でも、2回目のシンクポイントダンプファイルが有効になっているため、2回目のシンクポイントダンプファイル取得時点からの回復ができます。

シンクポイントダンプファイルも、二重化できます。二重化したシンクポイントダンプファイルには、同じ情報が書き込まれます。

解説 ステータスファイルの配置について解説します。

ステータスファイルは、二重化が必須です。A系とB系のステータスファイルに、同時に同じ情報が書き込まれます。そのため、A系とB系の両方が正常な状態でないと書き込みできません。
ディスク障害が発生しても、正常なA系とB系のステータスファイルが残るように配置します。
ステータスファイルは、**少なくとも3グループ以上**用意してください。

ディスクが2個の場合は、次の図のように配置します。

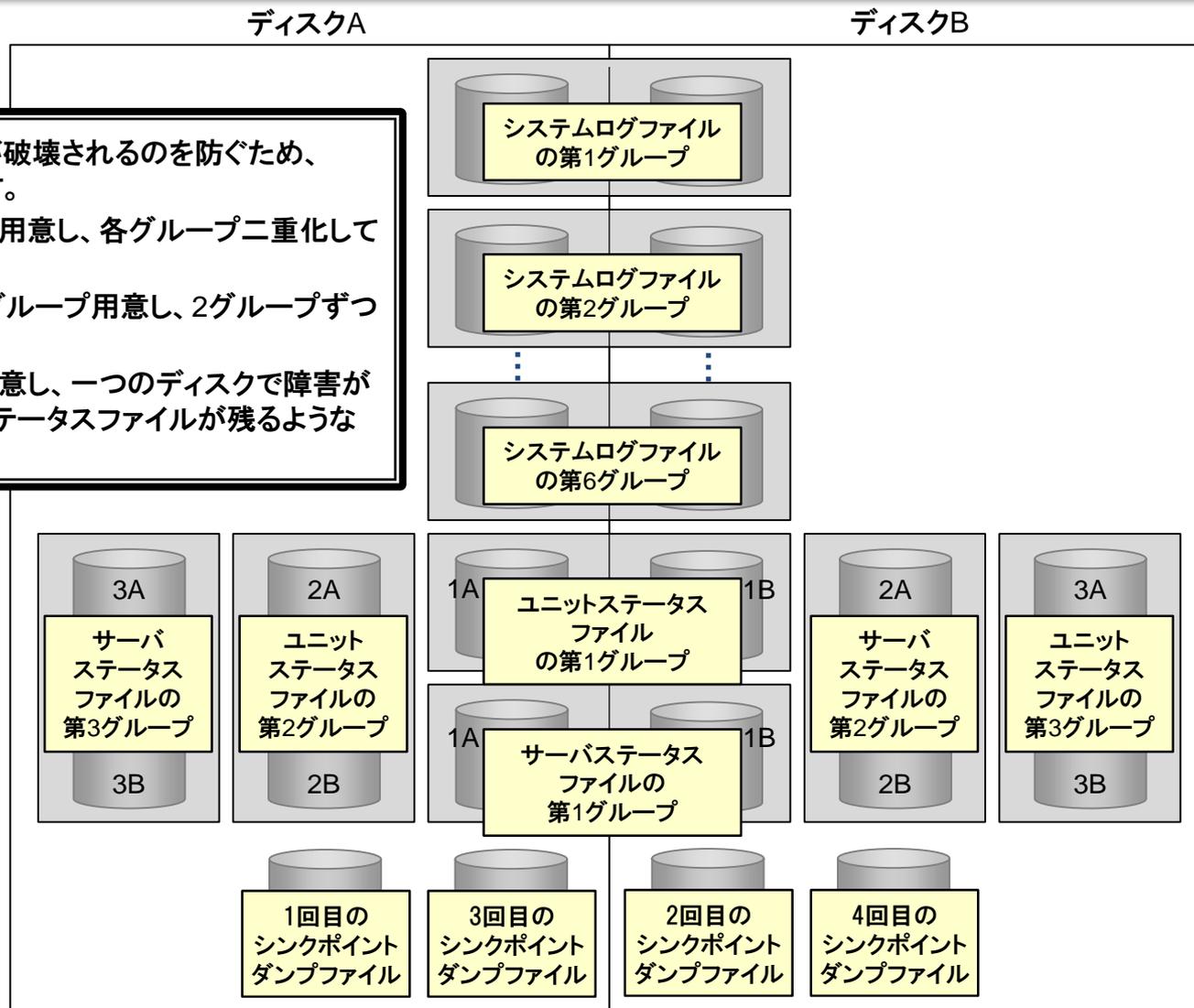


ディスクAに障害が発生しても、ディスクBに第3グループのA系とB系のステータスファイルがあるので、HiRDBは異常終了しません。

4-2-7 システムファイルの構成

解説 システムファイルの配置方針を次にまとめます。

- ディスク障害でシステムファイルが破壊されるのを防ぐため、ディスクを少なくとも2個用意します。
- システムログファイルは6グループ用意し、各グループ二重化して異なるディスクに作成します。
- シンクポイントダンプファイルは4グループ用意し、2グループずつ異なるディスクに作成します。
- ステータスファイルは3グループ用意し、一つのディスクで障害が発生しても、正常なA系とB系のステータスファイルが残るような配置で作成します。

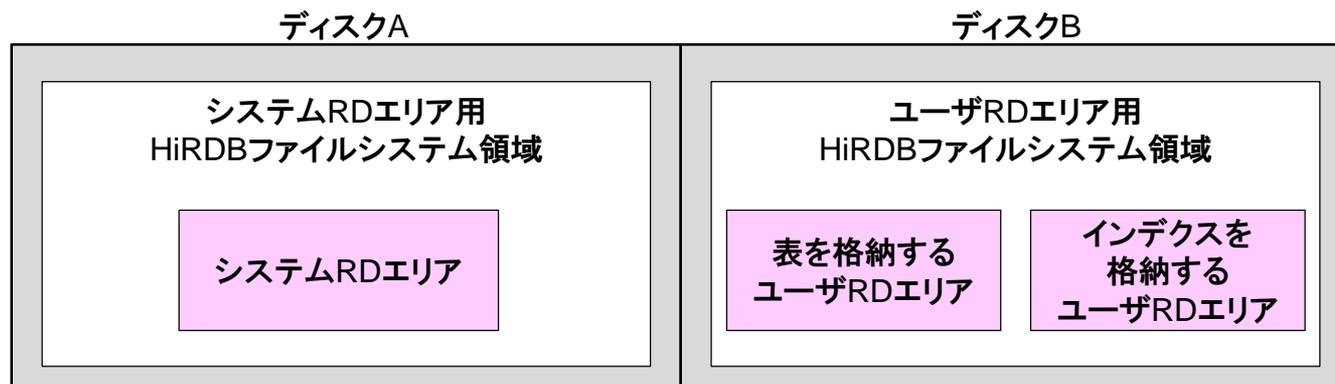


解説

4-2-2項のHiRDBファイルシステム領域の配置方針を考慮した、RDエリア用のHiRDBファイルシステム領域に作成するRDエリアの配置の方針を次に示します。

- 表とインデクスは異なるRDエリアに格納します。
- アクセスが多い表が複数ある場合は、別のRDエリアに格納することを検討してください。

上記の方針を踏まえたRDエリアの配置を図に示します。



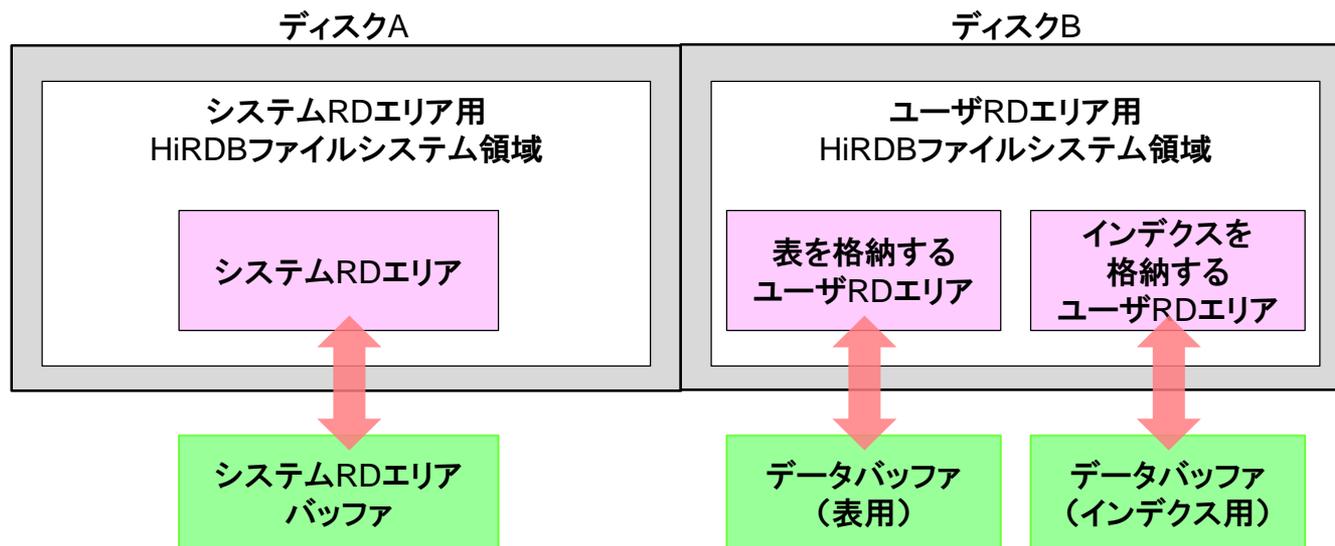
4-2-9 グローバルバッファの配置

解説

グローバルバッファには、システムRDエリアのデータ入出力に使用するシステムRDエリアバッファと、ユーザRDエリアのデータ入出力に使用するデータバッファがあります。グローバルバッファの配置の方針を次に示します。

- 表とインデクスは異なるRDエリアに格納するため、別々にデータバッファ(表用とインデクス用)を作成します。
- システムRDエリアバッファ専用、システムRDエリアバッファを作成します。

上記の方針を踏まえたグローバルバッファの配置を図に示します。



4. データベース設計のポイント

4.1 概要

4.2 データベース設計

4.3 リソース容量見積もり

4-3-1 リソース見積もり項目(1)

解説

これまでに決定した要件やデータベース設計を基に、リソースの容量を見積もります。見積もる項目をまとめた、見積もりシートを次に示します。

カテゴリ	リソース	単位	変数	見積もり値	
システム ファイル	システムログファイル	容量	キロバイト	LOG_VOL	
		レコード長	キロバイト	LOG_RECLENG	
		レコード数	個	LOG_REC_CNT	
	シンクポイントダンプファイル	容量	キロバイト	SPD_VOL	
		レコード数	個	SPD_REC_CNT	
	ユニットステータスファイル	容量	キロバイト	UTSTS_VOL	
		レコード数	個	UTSTS_REC_CNT	
	サーバステータスファイル	容量	キロバイト	STS_VOL	
		レコード数	個	STS_REC_CNT	
	作業表ファイル	容量	キロバイト	WRK_VOL	

4-3-2 リソース見積もり項目(2)

カテゴリ	リソース		単位	変数	見積もり値	
RDエリア	システム RDエリア	マスタ ディレクトリ用	容量	メガバイト	RDMAST_VOL	
			セグメント長	ページ	RDMAST_SEGLENG	
			セグメント数	個	RDMAST_SEGCNT	
		データ ディレクトリ用	容量	メガバイト	RDDIR_VOL	
			セグメント長	ページ	RDDIR_SEGLENG	
			セグメント数	個	RDDIR_SEGCNT	
		データ ディクショナリ用	容量	メガバイト	RDDICT_VOL	
			セグメント長	ページ	RDDICT_SEGLENG	
			セグメント数	個	RDDICT_SEGCNT	
	ユーザRDエリア(表用)		容量	キロバイト	RDTBL_VOL_1	
			セグメント長	ページ	RDTBL_SEGLENG_1	
			セグメント数	個	RDTBL_SEGCNT_1	
			容量	キロバイト	RDTBL_VOL_2	
			セグメント長	ページ	RDTBL_SEGLENG_2	
			セグメント数	個	RDTBL_SEGCNT_2	
			容量	キロバイト	RDTBL_VOL_3	
			セグメント長	ページ	RDTBL_SEGLENG_3	
			セグメント数	個	RDTBL_SEGCNT_3	
容量			キロバイト	RDTBL_VOL_4		
セグメント長			ページ	RDTBL_SEGLENG_4		
セグメント数			個	RDTBL_SEGCNT_4		

4-3-3 リソース見積もり項目(3)

カテゴリ	リソース		単位	変数	見積もり値
RDエリア	ユーザRDエリア (インデクス用)	容量	キロバイト	RDIDX_VOL	
		セグメント長	ページ	RDIDX_SEGLENG	
		セグメント数	個	RDIDX_SEGCNT	
HiRDB ファイル システム 領域	システムファイル用	容量	メガバイト	HiAREA_SYSFIL_1	
				HiAREA_SYSFIL_2	
				HiAREA_SYSFIL_3	
				HiAREA_SYSFIL_4	
				HiAREA_SYSFIL_5	
				HiAREA_SYSFIL_6	
	作業表ファイル用	容量	メガバイト	HiAREA_WRKFILE	
	システムRDエリア用	容量	メガバイト	HiAREA_SYSRD	
	ユーザRDエリア用	容量	メガバイト	HiAREA_USRRD_1	
				HiAREA_USRRD_2	
HiAREA_USRRD_3					
HiAREA_USRRD_4					
グローバル バッファ	システムRDエリアバッファ	容量	ページ	SYSBUFF	
	データバッファ(表用)			TBLBUFF	
	データバッファ(インデクス用)			IDXBUFF	
排他制御プール		容量	キロバイト	LCK_POOL	

解説 リソースの容量見積もり方法について解説します。

■容量見積もりの方法

(1)マニュアルの計算式

マニュアル「システム導入・設計ガイド」の各リソースの容量見積もりの計算式を使用して計算してください。

(2)本資料で紹介する計算式

簡易的に見積もりたい場合は、以降で紹介する計算式を使用して計算してください。
詳細に見積もりたい場合は、マニュアルの計算式で見積もりしてください。

4-3-5 リソースごとの容量見積もり方法

解説

リソースごとの容量見積もり方法を示します。
見積もるリソースの順序は以下の表の項番の順になります。

項番	カテゴリ	リソース	見積もり方法
1	システムファイル	システムログファイル	本資料で紹介する計算式(4-3-8項)または詳細に見積もりたい場合は、マニュアルの計算式
		シンクポイントダンプファイル	マニュアルの計算式
		ユニットステータスファイル	
		サーバステータスファイル	
2	作業表ファイル		マニュアルの計算式
3	RDエリア	システムRDエリア	マニュアルの計算式
		ユーザRDエリア	マニュアルの計算式
4	HiRDBファイルシステム領域	システムファイル用	各システムファイル容量から求める
		作業表ファイル用	マニュアルの計算式
		システムRDエリア用	システムRDエリア容量から求める
		ユーザRDエリア用	ユーザRDエリア容量から求める
5	グローバルバッファ	システムRDエリアバッファ	本資料で紹介する計算式(4-3-10項)または詳細に見積もりたい場合は、マニュアルの計算式
		データバッファ(表用)	本資料で紹介する計算式(4-3-11項)または詳細に見積もりたい場合は、マニュアルの計算式
		データバッファ(インデクス用)	
6	排他制御プール		本資料で紹介する計算式(4-3-13項)または詳細に見積もりたい場合は、マニュアルの計算式

4-3-6 リソース見積もりでの入力と出力(1)

解説 リソース見積もり時に用意しておくもの(入力)と見積もりツールや計算式で得られるもの(出力)を、以下に示します。

カテゴリ	リソース	入力	出力
システムファイル	システムログファイル	<ul style="list-style-type: none"> ・表の行長(業務要件項番4) ・トランザクションごとの更新量の最大値(性能要件項番1) ・1日あたりの平均トランザクション数(性能要件項番3) 	容量(LOG_VOL) レコード長 (LOG_RECLENG) レコード数(LOG_REC_CNT) レコード長とレコード数の設計の考え方を4-3-8項で解説。
	シンクポイントダンプファイル	<ul style="list-style-type: none"> ・最大同時実行数(性能要件項番4) ・シンクポイントダンプファイルのグループ数 (4-2-5項 シンクポイントダンプファイルの配置結果) 	容量(SPD_VOL) レコード数(SPD_REC_CNT)
	ユニットステータスファイル	<ul style="list-style-type: none"> ・最大同時実行数(性能要件項番4) ・グローバルバッファ数 (4-2-8項 RDエリアの配置および4-2-9項 グローバルバッファの配置結果) 	容量(UTSTS_VOL) レコード数 (UTSTS_REC_CNT)
	サーバステータスファイル	<ul style="list-style-type: none"> ・システムログファイルのグループ数 (4-2-3項,4-2-4項 システムログファイルの配置結果) ・シンクポイントダンプファイルのグループ数 (4-2-5項 シンクポイントダンプファイルの配置結果) 	容量(STS_VOL) レコード数(STS_REC_CNT)
作業表ファイル		<ul style="list-style-type: none"> ・表の行長(業務要件項番4) ・表のデータ件数(業務要件項番3) 	容量(WRK_VOL)
RDエリア	システムRDエリア	システム規模(ユーザ数、RDエリア数、表の数(業務要件項番1))など)	容量(RDMASTER_VOL, RDDDIR_VOL, RDDICT_VOL) セグメント数 (RDMASTER_SEGCNT, RDDDIR_SEGCNT, RDDICT_SEGCNT)

4-3-7 リソース見積もりでの入力と出力(2)

カテゴリ	リソース	入力	出力
RDエリア	ユーザRDエリア	<ul style="list-style-type: none"> ・表の行長(業務要件項番4) ・表の列数(業務要件項番2) ・表のデータ件数(業務要件項番3) ・インデクスのキー長(業務要件項番6) ・インデクスのキー種類数(業務要件項番7) ・インデクスの重複数(業務要件項番8) ・ページ長(ページ長の考え方を4-3-9項で解説。) ・セグメント長(セグメント長の考え方を4-3-9項で解説。) 	容量(RDTBL_VOL_xxx, RDIDX_VOL_xxx) セグメント数 (RDTBL_SEGCNT_xxx, RDIDX_SEGCNT_xxx)
HiRDB ファイル システム 領域	システム ファイル用	<ul style="list-style-type: none"> ・システムログファイルの容量(LOG_VOL) ・シンクポイントダンプファイルの容量(SPD_VOL) ・ユニットステータスファイルの容量(UTSTS_VOL) ・サーバステータスファイルの容量(STS_VOL) 	容量 (HiAREA_SYSFILE_xxx)
	作業表 ファイル用	<ul style="list-style-type: none"> ・作業表ファイルの容量(WRK_VOL) ・最大同時実行数(性能要件項番4) 	容量(HiAREA_WRKFILE)
	システム RDエリア用	<ul style="list-style-type: none"> ・マスタディレクトリ用RDエリアの容量(RDMAST_VOL) ・データディレクトリ用RDエリアの容量(RDDIR_VOL) ・データディクショナリ用RDエリアの容量(RDDICT_VOL) 	容量(HiAREA_SYSRD)
	ユーザ RDエリア用	<ul style="list-style-type: none"> ・ユーザRDエリアの容量(RDTBL_VOL_xxxまたはRDIDX_VOL) 	容量 (HiAREA_USRRD_xxx)
グローバル バッファ	システム RDエリアバッファ	<ul style="list-style-type: none"> ・最大同時実行数(性能要件項番4) 簡易的に見積もる場合の計算式を4-3-10項で解説。	容量(SYSBUFF)
	データバッファ (表用)	<ul style="list-style-type: none"> ・ユーザRDエリアの容量(RDTBL_VOL_xxx) ・ユーザRDエリアの容量(RDIDX_VOL) 	容量(TBLBUFF)
	データバッファ (インデクス用)	簡易的に見積もる場合の計算式を4-3-11項で解説。 バッファ割当ての調整方法について4-3-12項で解説。	容量(IDXBUFF)
排他制御プール		<ul style="list-style-type: none"> ・トランザクションごとの処理行数の最大値(性能要件項番2) 簡易的に見積もる場合の計算式を4-3-13項で解説。	容量(LCK_POOL)

解説 システムログファイルの容量は、システムログファイルのレコード長とレコード数から見積もります。

■システムログファイルのレコード長(LOG_RECLENG)

システムログファイルのレコード長は、1024 バイトを指定することを推奨します。
レコード長が短い場合でも、サイズの大きいデータのファイルへの入出力回数は増加しません。
レコード単位の切り上げによる空き領域が小さくなるため、ファイルの使用効率は良くなります。

■システムログファイルの容量

システムログファイルのレコード数を得るのに、HiRDBを運用するために必要な1日分のシステムログファイルの容量を求めます。

↑トランザクションごとの更新量の最大値 × 1日あたりの平均トランザクション数 × 1.5 ↑

性能要件 項番1 max_trn_size

性能要件 項番3 tpd

安全係数

■システムログファイルのレコード数(LOG_REC CNT)

次の計算式から見積もれます。

↑システムログファイルの容量 / システムログファイルのレコード長 ↑

解説 ユーザRDエリアの容量見積もりで必要になるページ長、セグメント長の考え方を解説します。

■ ページ長の考え方

(1) 表のページは、表の行長とページアクセスの競合を考慮し、決定してください。

・検索を優先する場合、ページサイズを大きくしてください。

1ページの格納行数を増やすことで、I/O回数を減らします。ただし、1ページの格納行の最大は255行のため、行長が100バイト未満の表をページ長が30720バイトのRDエリアに格納しないでください。

・同一ページへの更新が多い場合、ページ長は8KB程度と1ページの格納件数を少なくすることでアクセス時の競合を減らしてください。100多重以上のアクセスなど、同一ページへの更新頻度が高いと競合待ちが発生するためです。

(2) インデクスのページは、入出力の効率の良い4096～8192バイト程度としてください。

■ セグメント長の考え方

(1) 通常、RDエリア容量の総ページ数の1/10～1/100程度のサイズがお勧めです。

また、複数のRDエリアで似たサイズは揃えておくとう管理しやすいです。

(2) 大量のデータ挿入をする表では、出来るだけセグメント長を大きく(最大は16000)し、セグメント確保回数を減らすと効率が良いです。

■ RDエリア設計の考え方

RDエリアを設計する際、1RDエリアに1つの表またはインデクス、1HiRDBファイルシステム領域に1RDエリアを配置する構成とすると、監視やチューニングをきめ細かく行えるため、お勧めです。

しかし、RDエリア数、ファイル数が多いとファイル数等の制限あるいは、運用面の都合により、実現が難しいこともあります。そのような場合は、性質の似たものでまとめると良いです(競合の考慮も忘れないでください)。

解説

グローバルバッファは入出力時にデータを格納する領域です。そのため、グローバルバッファの容量は、入出力の最小単位であるページ単位で見積もります。マスタディレクトリRDエリア、データディレクトリRDエリア、データディクショナリRDエリアに割り当てるグローバルバッファのページ数は、それぞれ次の計算式から見積もります。

■システムRDエリアバッファの容量(SYSBUFF)

システムRDエリアバッファのページ数 = 最大同時接続数 × 50
性能要件 項番4 maxusers (単位: ページ)

解説

表を格納しているユーザ用RDエリアのグローバルバッファ容量、インデクスを格納しているユーザ用RDエリアのグローバルバッファ容量は次の計算式から見積もります。
単位をページに変更するには、見積もったデータバッファの容量をページ長で割ります。

- **表を格納しているユーザ用RDエリアの容量(TBLBUFF)**
グローバルバッファの容量は、データベース容量の1%を目安として見積もります。
これを最低ラインとして、性能要件にあわせて増やします。

表を格納しているユーザ用RDエリアのグローバルバッファ容量
＝ユーザ用RDエリア(表用)の容量×0.01

【例】 ユーザ用RDエリアの容量25ギガバイトの場合
表を格納しているユーザ用RDエリアのグローバルバッファ容量
＝25ギガバイト×0.01＝250メガバイト

- **インデクスを格納しているユーザ用RDエリアの容量(IDXBUFF)**
インデクスを格納しているユーザ用RDエリアに割り当てるグローバルバッファの容量は、次の計算式から見積もります。

インデクスを格納しているユーザ用RDエリアのグローバルバッファ容量
＝インデクス容量※×0.15

※ インデクス容量を見積れない場合は、インデクス容量を「ユーザ用RDエリア(表用)の容量×0.1」としてください。

解説

グローバルバッファの割り当ては、データの特徴や業務性能、データのアクセス方法によって調整してください。調整する場合の考え方を示します。

- アクセス頻度が高い表を格納したユーザRDエリアのデータのヒット率が上がるように、割り当てるグローバルバッファのページ数を多くします。
- アクセス頻度が低い表、またはアクセス時間が長くても問題がない表を格納したユーザRDエリアに割り当てるグローバルバッファのページ数を少なくします。
- インデクスに割り当てるグローバルバッファのページ数を多くします。

解説 排他制御プールの容量は、次の計算式から見積もります。

排他制御プール容量

= ↑トランザクションごとの処理行数の最大値^{※1} / 4^{※2} ↑ × 最大同時接続数 / 1024

性能要件 項番2 max_suid_cnt

性能要件 項番4 maxusers

(単位:メガバイト)

※1 但し、排他を取得しない検索は除外する。

※2 32ビットモードのHiRDBの場合、6で割ってください。

4-3-14 リソース見積もりシート(1)

解説

計算式で見積もった容量をリソース見積もりシートにまとめます(見積もり値は仮定値)。これらの値を入力してHiRDBを構築します。

カテゴリ	リソース		単位	変数	見積もり値	
システム ファイル	システムログファイル	容量	キロバイト	LOG_VOL	1589	
		レコード長	キロバイト	LOG_RECLENG	1024	
		レコード数	個	LOG_REC CNT	1589	
	シンクポイントダンプファイル	容量	キロバイト	SPD_VOL	108	
		レコード数	個	SPD_REC CNT	27	
	ユニットステータスファイル	容量	キロバイト	UTSTS_VOL	140	
		レコード数	個	UTSTS_REC CNT	35	
	サーバステータスファイル	容量	キロバイト	STS_VOL	256	
レコード数		個	STS_REC CNT	64		
作業表ファイル		容量	キロバイト	WRK_VOL	927	
RDエリア	システム RDエリア	マスタ ディレクトリ用	容量	メガバイト	RDMAST_VOL	20
			セグメント長	ページ	RDMAST_SEGLENG	50
			セグメント数	個	RDMAST_SEGCNT	100
		データ ディレクトリ用	容量	メガバイト	RDDIR_VOL	10
			セグメント長	ページ	RDDIR_SEGLENG	50
			セグメント数	個	RDDIR_SEGCNT	50
	データ ディクショナリ用	容量	メガバイト	RDDICT_VOL	100	
		セグメント長	ページ	RDDICT_SEGLENG	5	
		セグメント数	個	RDDICT_SEGCNT	5000	
	ユーザRDエリア(表用)		容量	キロバイト	RDTBL_VOL_1	7576
			セグメント長	ページ	RDTBL_SEGLENG_1	110
			セグメント数	個	RDTBL_SEGCNT_1	17
ユーザRDエリア(インデクス用)		容量	キロバイト	RDTBL_VOL_2	10736	
		セグメント長	ページ	RDTBL_SEGLENG_2	140	
		セグメント数	個	RDTBL_SEGCNT_2	19	

4-3-15 リソース見積もりシート(2)

カテゴリ	リソース		単位	変数	見積もり値
HiRDB ファイル システム 領域	システムファイル用	容量	メガバイト	HiAREA_SYSFILE_1	3
				HiAREA_SYSFILE_2	3
				HiAREA_SYSFILE_3	4
				HiAREA_SYSFILE_4	4
				HiAREA_SYSFILE_5	4
				HiAREA_SYSFILE_6	4
	作業表ファイル用	容量	メガバイト	HiAREA_WRKFILE	12
	システムRDエリア用	容量	メガバイト	HiAREA_SYSRD	130
	ユーザRDエリア用	容量	メガバイト	HiAREA_USRRD_1	8
				HiAREA_USRRD_2	11
グローバル バッファ	システムRDエリアバッファ	容量	ページ	SYSBUFF	20
	データバッファ(表用)			TBLBUFF	256
	データバッファ(インデクス用)			IDXBUFF	124
排他制御プール		容量	キロバイト	LCK_POOL	20375

5. データベース構築のポイント

5.1 概要

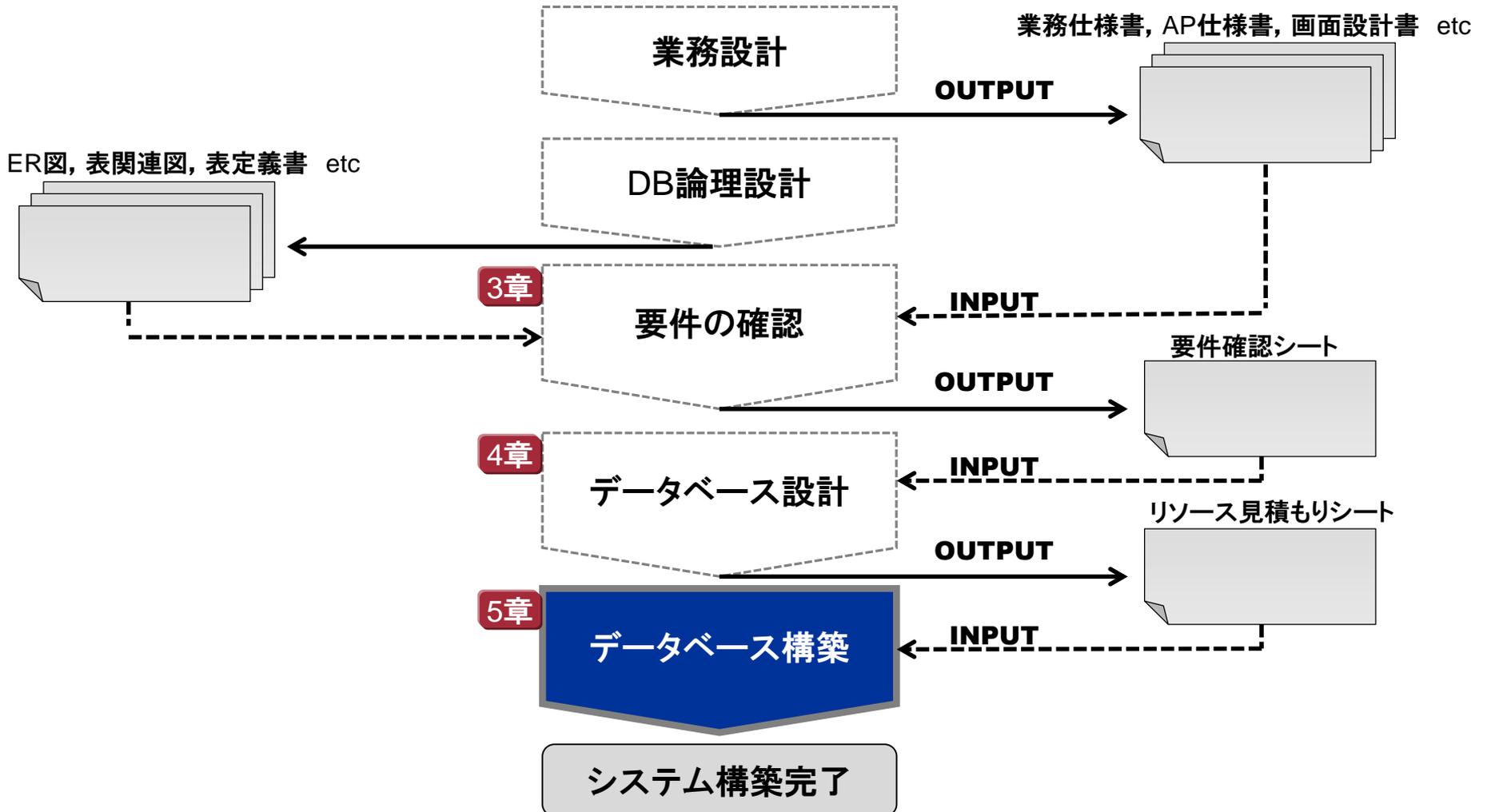
5.2 データベース構築手順

5.3 サーバ・ファイル環境の決定

5.4 パラメタ設定

5.5 データベース構築

解説 本章ではデータベース構築について説明します。



5. データベース構築のポイント

5.1 概要

5.2 データベース構築手順

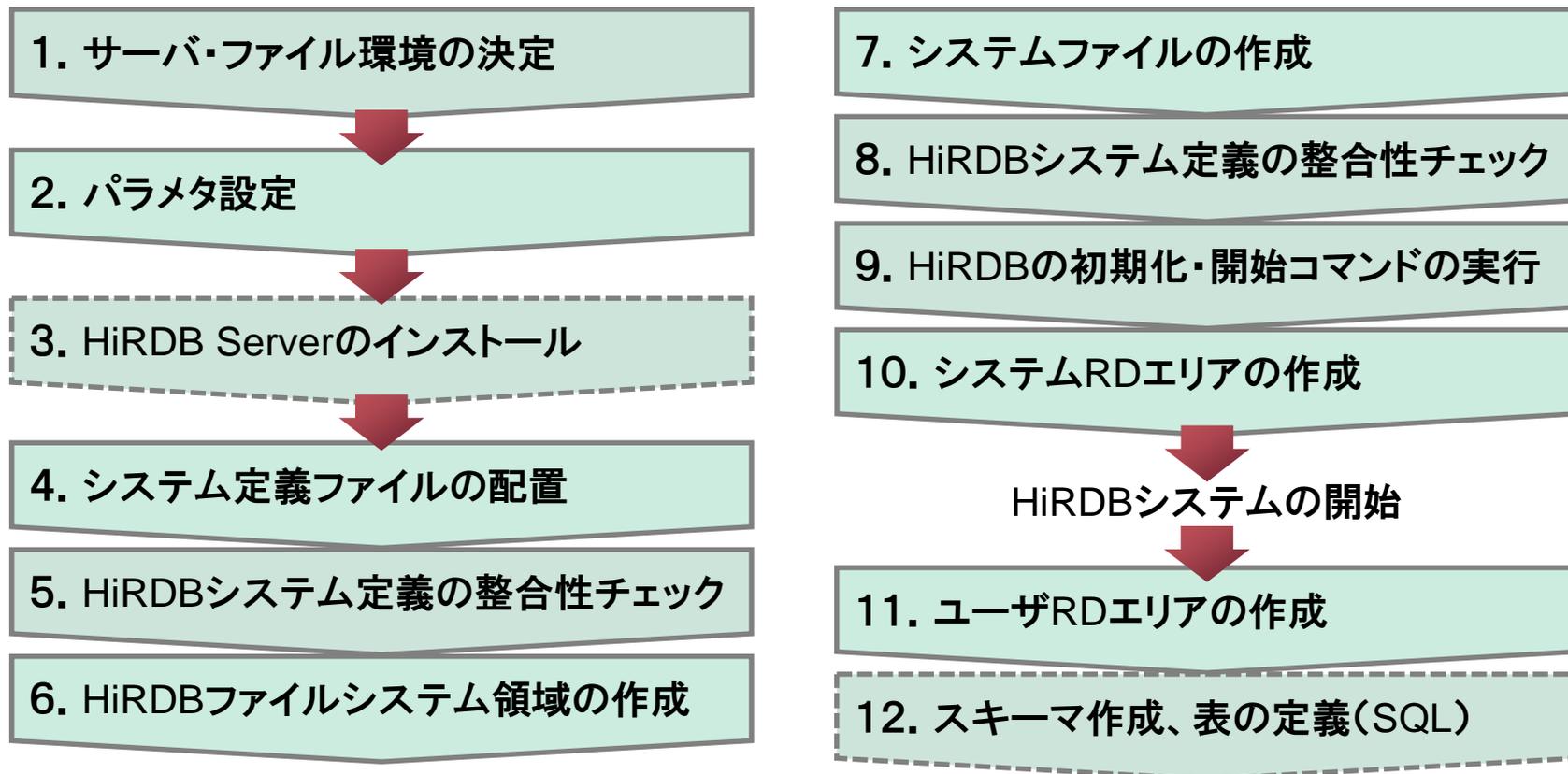
5.3 サーバ・ファイル環境の決定

5.4 パラメタ設定

5.5 データベース構築

解説 HiRDBにおけるデータベースの構築手順を示します。

■HiRDBの構築手順



※本資料では、実線枠で囲んだ手順について解説します。

5. データベース構築のポイント

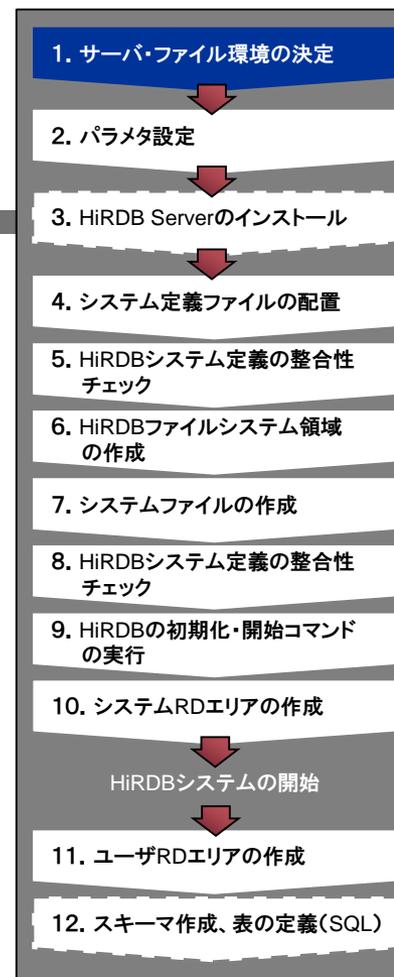
5.1 概要

5.2 データベース構築手順

5.3 サーバ・ファイル環境の決定

5.4 パラメタ設定

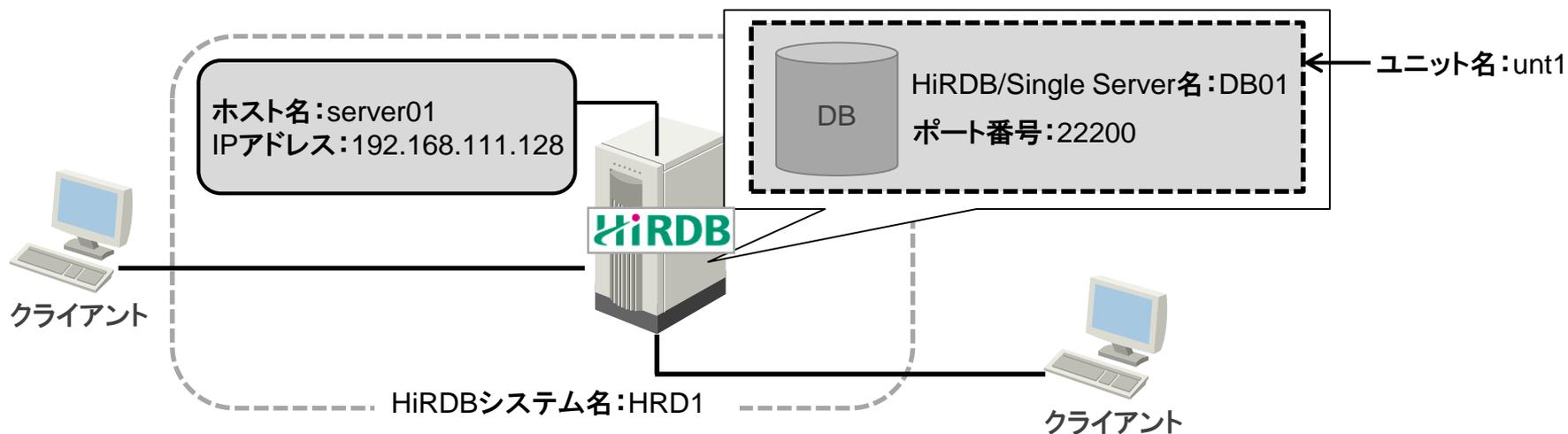
5.5 データベース構築



解説

サーバマシン環境に関して以下の項目を確認または決定しておきます。
確認・決定結果例を以下に示します。

カテゴリ	確認/決定項目	例
サーバマシン環境	ホスト(コンピュータ)名	Server01
	IPアドレス	192.168.111.128
	ポート番号	22200
HiRDBサーバ環境	HiRDBシステム名	HRD1
	ユニット名	unt1
	HiRDB/Single Server名(サーバ名)	DB01



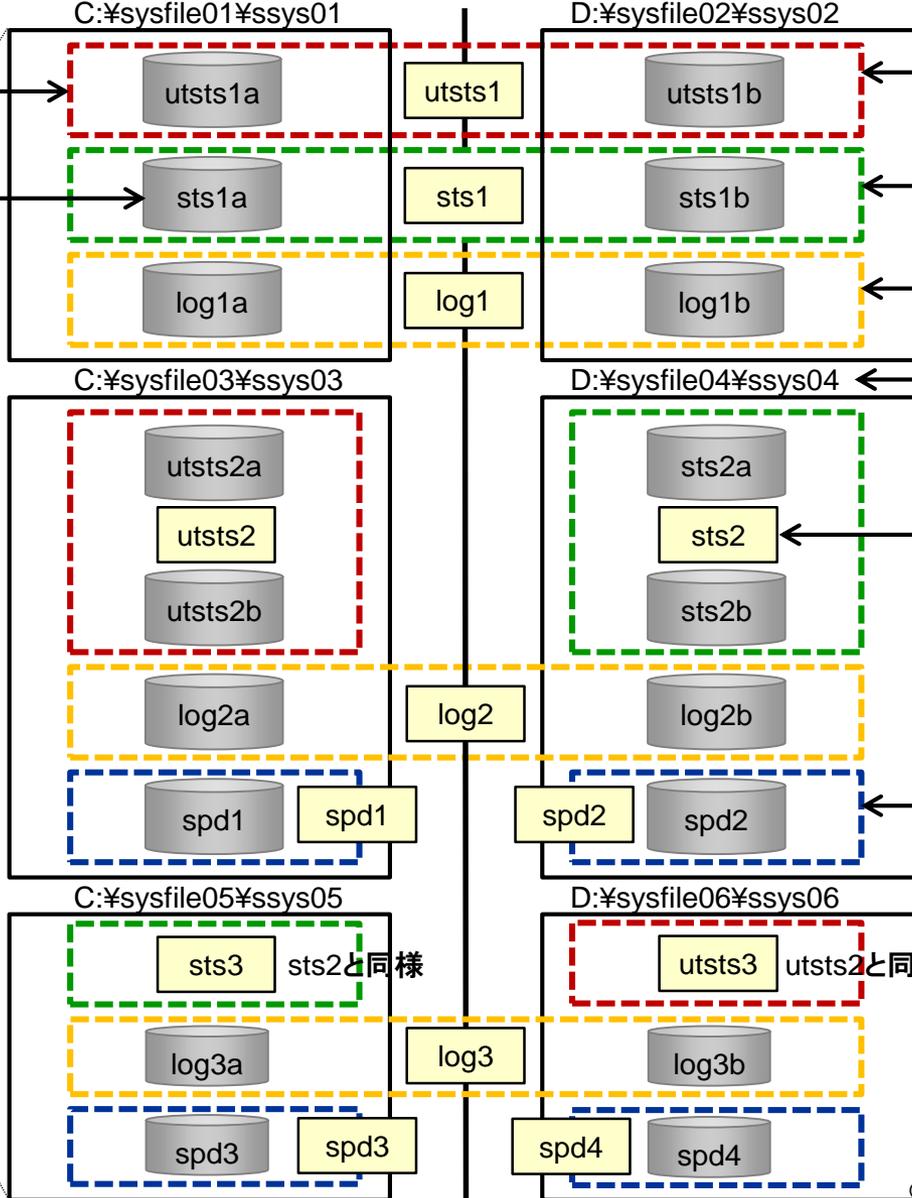
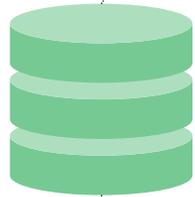
5-3-2 ファイル環境の決定(1)

ファイルの名称や配置を決定します。決定結果例を以下に示します。

■システムファイルグループ

システムファイル名

C:¥



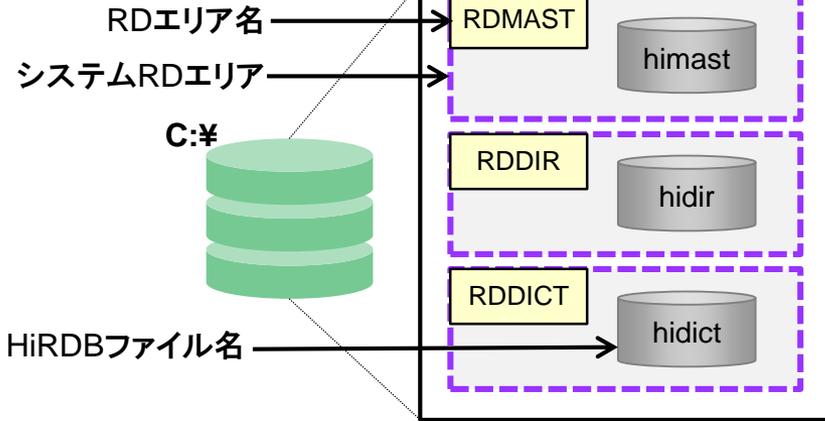
※システムログファイルは、図を見易くするために3グループにしています。実際は、6グループ以上としてください。

5-3-3 ファイル環境の決定(2)

■RDエリア

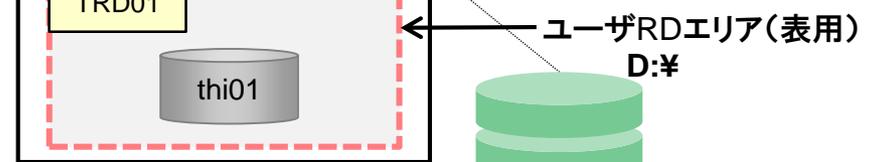
システムRDエリア

C:¥sysarea¥srdsys

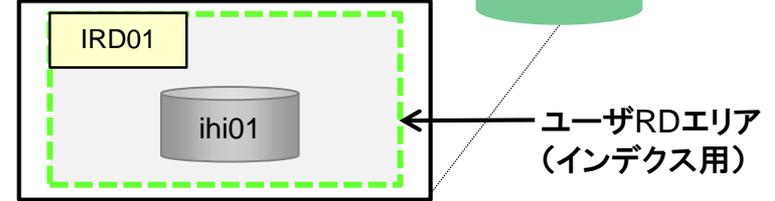


ユーザRDエリア

D:¥tblarea01¥tsys01 ← HiRDBファイルシステム領域 (RDエリア用)

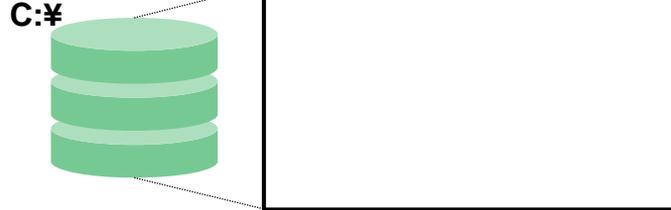


D:¥idxarea01¥isys01

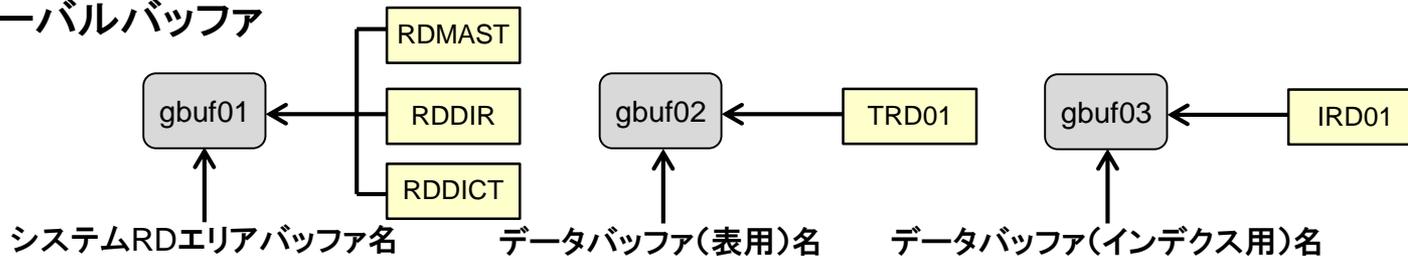


■作業表ファイル

C:¥workarea¥wrksys ← HiRDBファイルシステム領域 (作業表ファイル用)



■グローバルバッファ



5. データベース構築のポイント

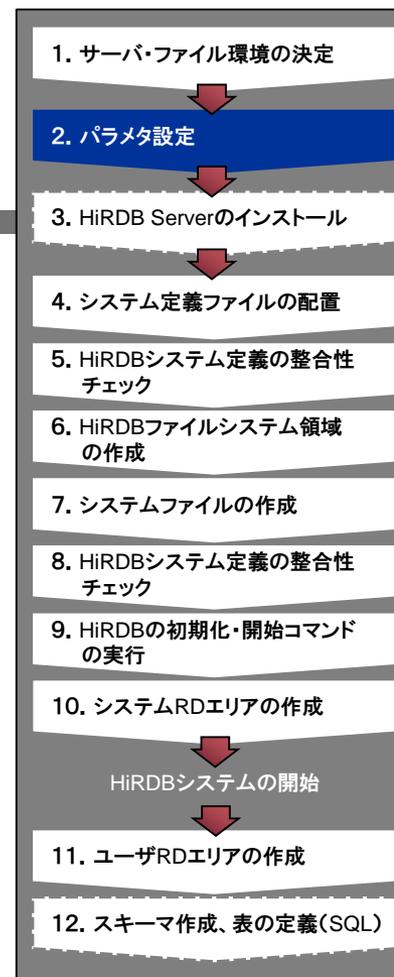
5.1 概要

5.2 データベース構築手順

5.3 サーバ・ファイル環境の決定

5.4 パラメタ設定

5.5 データベース構築



解説

リソース見積もりシートの値とサーバ・ファイル環境で決定した値、運用設計※、信頼性設計※、セキュリティ設計※をした結果を用意しておいて、以下の項目の作成時に定義ファイル、コマンドの引数や制御文ファイルに値を設定します。

運用ディレクトリ

※本資料では解説していませんが、これらの設計を行ってください。

- ① **システム定義ファイル [%PDDIR%¥CONF¥]に作成**
 システム定義ファイルとは、HiRDBの実行環境を定義するテキストファイルです。
 システム定義ファイルには、次の3種類があります。

定義ファイルの種類	ファイル名	定義内容
システム共通定義ファイル	pdsys	<ul style="list-style-type: none"> ・システム全体に共通なパラメータ定義 (システム名称、IPアドレス、ポート番号、最大同時接続数、HiRDB起動モード、ユニット構成/サーバ構成の定義ほか) ・グローバルバッファの定義(グローバルバッファ名称、対象RDエリア)
ユニット制御情報定義ファイル	pdutsys	<ul style="list-style-type: none"> ・ユニット名 ・ユニットステータスファイルの定義(グループ名、ファイル名)
シングルサーバ定義ファイル	サーバ名 と同名	<ul style="list-style-type: none"> ・サーバステータスファイルの定義 ・システムログの定義 ・システムログファイルの定義 ・シンクポイントダンプファイルの定義

- ② **HiRDBファイルシステム領域作成に必要な情報 [コマンド(pdfmkfs)の引数に指定]**
 HiRDBファイルシステム領域のサイズ
 格納先とファイル名、使用目的(システムファイル用・RDエリア用・作業表用) ほか
- ③ **システムファイル作成に必要な情報 [コマンド(pdloginit, pdstsinit)の引数に指定]**
 システムログファイル、シンクポイントダンプファイル、ステータスファイルの
 ファイル名、レコード長、レコード数
- ④ **RDエリア作成に必要な情報 [コマンド(pdinit)の引数に指定する制御ファイルの中に指定]**
 RDエリア名、RDエリア種別、ページ長、セグメント長、HiRDBファイル名、セグメント数 ほか

定義例

```

=====
# pdsys : システム共通定義ファイル
=====
set pd_system_id=HRD1                ←システム名
set pd_name_port=22200                ←ポート番号
set pd_master_file_name= "C:¥sysarea¥srdsys¥himast " ←マスタディレクトリRDエリアの先頭に定義したHiRDBファイル名
set pd_max_users=10                  ←最大同時接続数
set pd_mode_conf=MANUAL2             ←HiRDBの開始方法

#ユニット構成の定義
pdunit -x server01 -u unt1           ←ホスト名・ユニット名

#サーバ構成の定義
pdstart -t SDS -s DB01 -x server01   ←サーバ種別・サーバ名・ホスト名

#グローバルバッファの定義
pdbuffer -a gbuf01 -n 20 ¥           ←システムRDバッファ名・バッファ面数 (SYSBUFFの値) ・割当先RDエリア名
-r RDMAST, RDDIR, RDDICT

pdbuffer -a gbuf02 -n 256 ¥          ←データバッファ (表用) 名・バッファ面数 (TBLBUFFの値) ・割当先RDエリア名
-r TRD01

pdbuffer -a gbuf03 -n 29 ¥           ←データバッファ (インデクス用) 名・バッファ面数 (IDXBUFFの値) ・割当先RDエリア名
-r IRD01
    
```

手動開始(MANUAL1 | MANUAL2)
自動開始(AUTO)

サーバ種別
 シングルサーバの場合
 SDS:シングルサーバ
 パラレルサーバの場合
 MGR:システムマネージャ
 FES:フロントエンドサーバ
 BES:バックエンドサーバ
 DIC:ディスクジョナリサーバ

定義例

```
#=====
# pdutsys : ユニット制御情報ファイル
#=====
# ユニット名の定義
set pd_unit_id=unt1

# ユニットステータスファイルの定義
set pd_syssts_file_name_1 = "utsts1", ¥ ←第1グループ名
"C:¥sysfile01¥ssys01¥utsts1a", ¥ ←A系ファイル名
"D:¥sysfile02¥ssys02¥utsts1b" ←B系ファイル名

set pd_syssts_file_name_2 = "utsts2", ¥ ←第2グループ名
"C:¥sysfile03¥ssys03¥utsts2a", ¥ ←A系ファイル名
"C:¥sysfile03¥ssys03¥utsts2b" ←B系ファイル名

set pd_syssts_file_name_3 = "utsts3", ¥ ←第3グループ名
"D:¥sysfile06¥ssys06¥utsts3a", ¥ ←A系ファイル名
"D:¥sysfile06¥ssys06¥utsts3b" ←B系ファイル名
```

定義例

```

=====
# DB01 シングルサーバ定義ファイル
=====
# 【パラメータによる定義】
# 《サーバステータスファイルの定義》
set pd_sts_file_name_1 = "sts1", ¥          ←第1グループ名
"C:¥sysfile01¥ssys01¥sts1a", ¥          ←A系ファイル名
"D:¥sysfile02¥ssys02¥sts1b"            ←B系ファイル名
—<省略>—
set pd_sts_file_name_3 = "sts3", ¥          ←第3グループ名
"C:¥sysfile05¥ssys05¥sts3a", ¥          ←A系ファイル名
"C:¥sysfile05¥ssys05¥sts3b"            ←B系ファイル名

# 《排他制御プールの定義》
set pd_lck_pool_size=20375                ←排他制御プールのサイズ

# 《システムログファイル/シンクポイントダンプファイルの定義》
set pd_log_dual =Y                        ←システムログファイルの二重化の有無
set pd_log_rec_leng = 1024                ←システムログファイルのレコード長
set pd_spd_assurance_count = 2           ←シンクポイントダンプファイルの有効保証世代数

#HiRDB起動時に使用可能（オンライン）となるロググループの設定
pdlogadfg -d sys -g log1 ONL             ←第1グループ
pdlogadfg -d sys -g log2 ONL             ←第2グループ
pdlogadfg -d sys -g log3 ONL             ←第3グループ
    
```

推奨値

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。

シングルサーバ定義ファイルのつづき

#システムログファイルの定義

pdlogadpf -d sys -g log1 ¥ ←第1グループ名

-a "C:¥sysfile01¥ssys01¥log1a" ¥ ←A系ファイル名

-b "D:¥sysfile02¥ssys02¥log1b" ←B系ファイル名

—<省略>—

pdlogadpf -d sys -g log3 ¥ ←第3グループ名

-a "C:¥sysfile05¥ssys05¥log3a" ¥ ←A系ファイル名

-b "D:¥sysfile06¥ssys06¥log3b" ←B系ファイル名

HiRDB起動時に使用可能となる（オンライン）シンクポイントダンプファイルの設定

pdlogadfg -d spd -g spd1 ONL ←第1グループ名

—<省略>—

pdlogadfg -d spd -g spd4 ONL ←第4グループ名

シンクポイントダンプファイルの定義

pdlogadpf -d spd -g spd1 ¥ ←第1グループ名

-a "C:¥sysfile03¥ssys03¥spd1" ←ファイル名

—<省略>—

pdlogadpf -d spd -g spd4 ¥ ←第4グループ名

-a "D:¥sysfile06¥ssys06¥spd4" ←ファイル名

作業表ファイル用HiRDBファイルシステム領域の定義

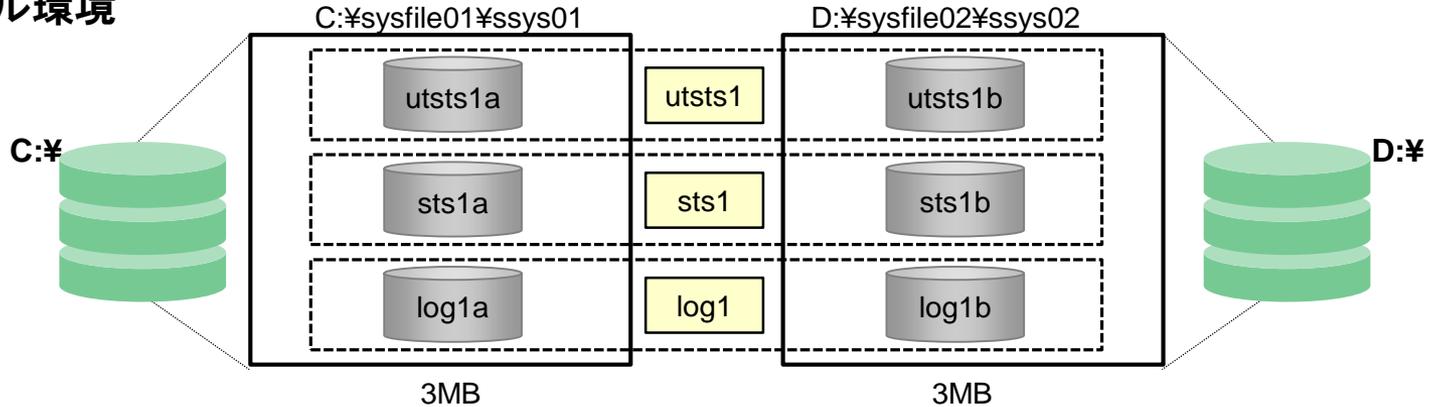
pdwork -v ¥

"C:¥workarea¥wrksys" ←作業表ファイル用HiRDBファイルシステム領域名

解説

HiRDBファイルシステム領域を作成するためには、**pdfmkfs**コマンドを用います。以下にシステムファイル用のHiRDBファイルシステム領域のコマンド設定例を示します。

■ファイル環境



■設定項目(見積もりした値やファイル環境で決定した値)

- 領域サイズ :3MB
- システムファイルの最大数 :3
- 使用目的 :システムファイル
- 領域名 :C:\%sysfile01%\%ssys01

- 領域サイズ :3MB
- システムファイルの最大数 :3
- 使用目的 :システムファイル
- 領域名 :D:\%sysfile02%\%ssys02

■コマンド設定例

使用目的

```
pdfmkfs -n 3 -l 3 -k SYS -i C:\%sysfile01%\%ssys01
```

```
pdfmkfs -n 3 -l 3 -k SYS -i D:\%sysfile02%\%ssys02
```

-i: -n オプションで指定したサイズ分の領域を、コマンド実行時に確保する場合に指定。省略した場合、HiRDB ファイルシステム領域の管理情報を格納するだけの領域を確保。

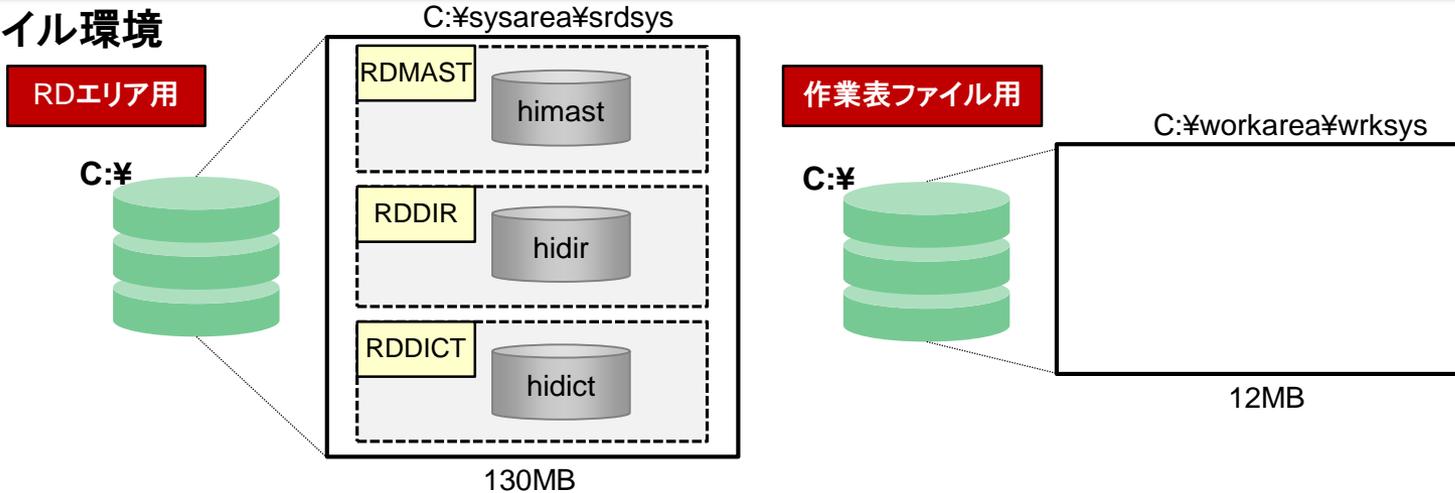
領域サイズ

システムファイルの最大数

領域名

解説 以下にRDエリア用、作業表ファイル用のHiRDBファイルシステム領域のコマンド設定例を示します。

■ファイル環境



■設定項目(見積もりした値やファイル環境で決定した値)

- 領域サイズ :130MB
- システムファイルの最大数 :3
- 使用目的 :RDエリア
- 領域名 :C:¥sysarea¥srdsys

- 領域サイズ :12MB
- システムファイルの最大数 :10
- 使用目的 :作業表ファイル
- 領域名 :C:¥workarea¥wrksys

■コマンド設定例

```

pdfmkfs -n 130 -l 3 -k DB -i C:¥sysarea¥srdsys
pdfmkfs -n 12 -l 10 -e 230 -k WORK -i C:¥workarea¥wrksys
    
```

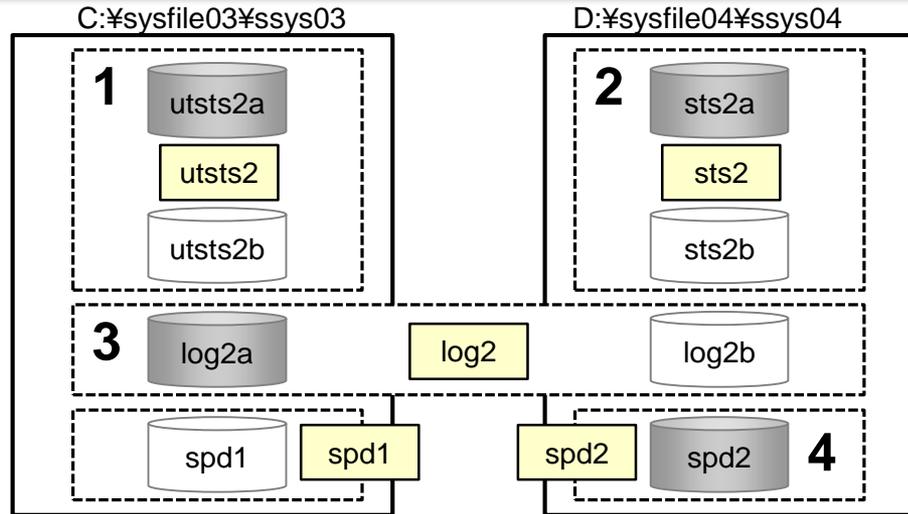
Labels for the commands: '使用目的' (Use Purpose) points to 'DB' and 'WORK'; '領域名' (Area Name) points to 'C:¥sysarea¥srdsys' and 'C:¥workarea¥wrksys'; '領域サイズ' (Area Size) points to '130' and '12'; 'システムファイルの最大数' (Maximum Number of System Files) points to '3' and '10'.

-e: HiRDB ファイルシステム領域に含まれる容量不足となったHiRDBファイル、作業表ファイルの自動増分回数(何回増分できるか)の総数を指定。

解説

システムファイルを作成するためには、**pdstsinit**および**pdloginit**コマンドを用います。以下にシステムファイルのコマンド設定例を示します。

■ファイル環境



■設定項目(見積もりした値やファイル環境で決定した値)

1 ユニットステータスファイル	
■ユニット名	:unt1
■レコード長	:4096バイト(デフォルト値)
■レコード数	:35
■ファイル名	:C:¥sysfile03¥ssys03¥utsts2a

2 サーバステータスファイル	
■サーバ名	:DB01
■レコード長	:4096バイト(デフォルト値)
■レコード数	:64
■ファイル名	:D:¥sysfile04¥ssys04¥sts2a

3 システムログファイル	
■ファイル種別名	:sys
■レコード長	:1024バイト
■レコード数	:1589
■ファイル名	:C:¥sysfile03¥ssys03¥log2a

4 シンクポイントダンプファイル	
■ファイル種別名	:spd
■レコード長	:4096バイト(デフォルト値)
■レコード数	:27
■ファイル名	:D:¥sysfile04¥ssys04¥spd2

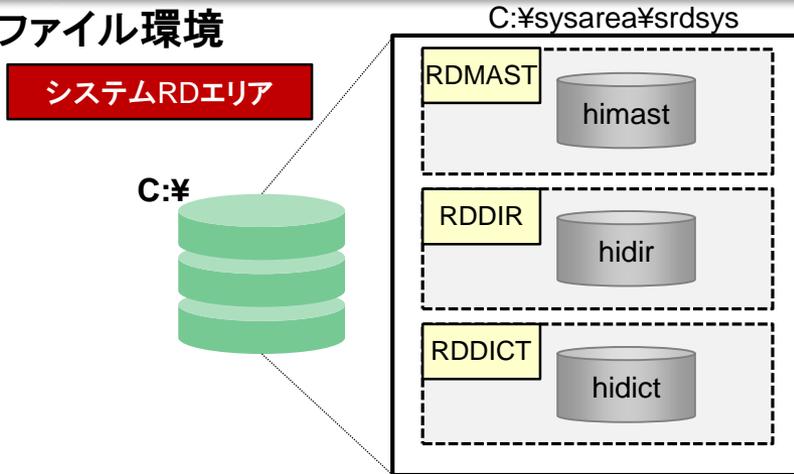
■コマンド設定例

	ユニット名	サーバ名	ファイル名	レコード数
1	pdstsinit	-u unt1	-f C:¥sysfile03¥ssys03¥utsts2a	-c 35
2	pdstsinit	-s DB01	-f D:¥sysfile04¥ssys04¥sts2a	-c 64
3	pdloginit	-d sys	-f C:¥sysfile03¥ssys03¥log2a	-l 1024 -n 1589
4	pdloginit	-d spd	-f D:¥sysfile04¥ssys04¥spd2	-n 27
	ファイル種別名	ファイル名	レコード数	レコード長

解説

システムRDエリアは、HiRDBを初期開始するときにデータベース初期設定ユーティリティ (pdinit) で作成する必要があります。
引数として指定する制御文ファイル(create rdarea文)の定義例とコマンド設定例を以下に示します。

■ファイル環境



■RDエリアの作成順序

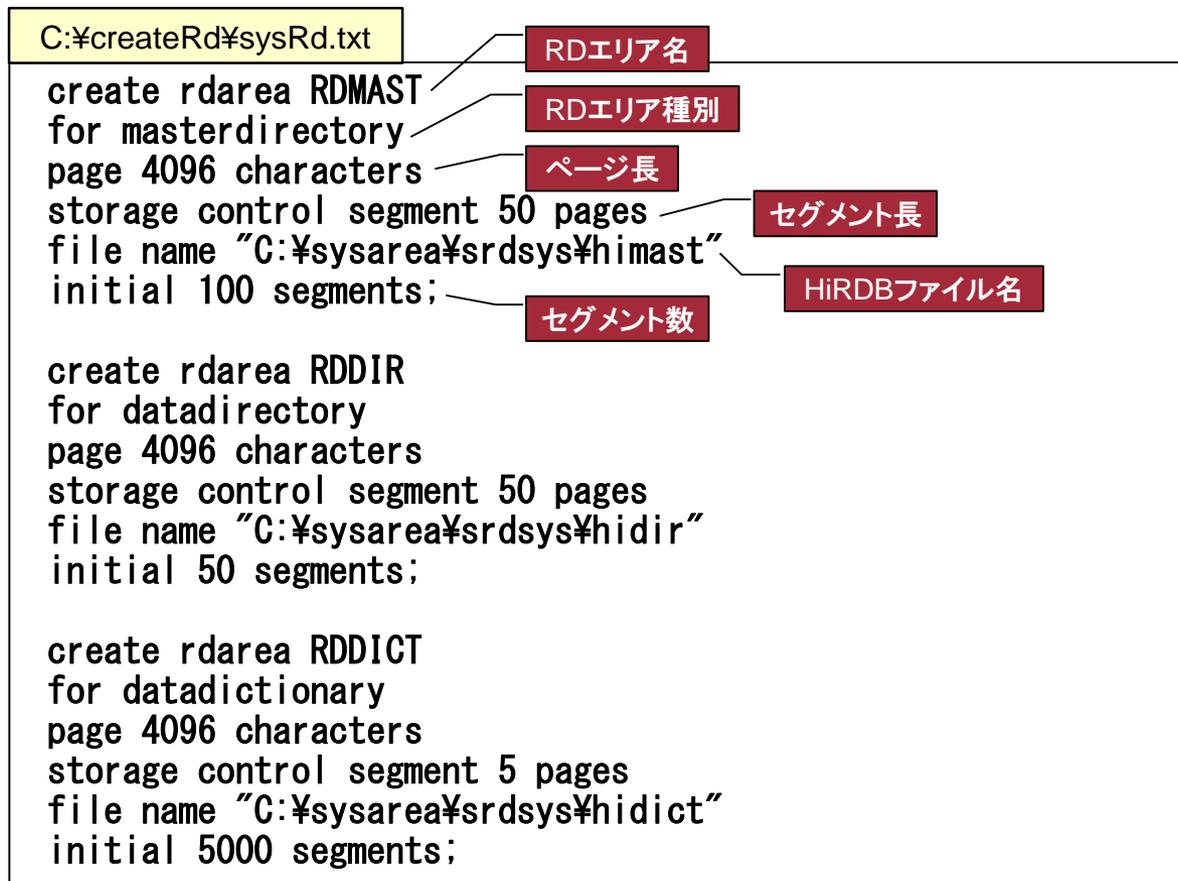
RDエリアは以下の順で作成する必要があります。
制御文ファイルでは、以下の順でcreate rdarea文を記述してください。

作成順序	RDエリアの種類
1	マスタディレクトリRDエリア
2	データディレクトリRDエリア
3	データディクショナリRDエリア
4	データディクショナリLOB RDエリア
5	ユーザRDエリア
	ユーザLOB RDエリア

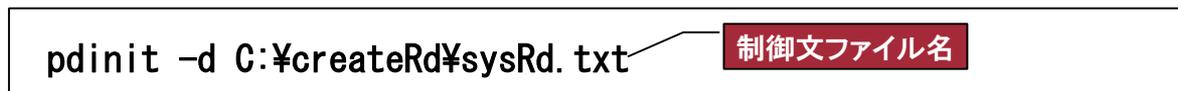
■設定項目(見積もりした値やファイル環境で決定した値)

<p>マスタディレクトリRDエリア</p> <ul style="list-style-type: none"> ■RDエリア名 :RDMAST ■ページ長 :4096(デフォルト値) ■セグメント長 :50 ■HiRDBファイル名 :C:¥sysarea¥srdsys¥himast ■セグメント数 :100 	<p>データディクショナリRDエリア</p> <ul style="list-style-type: none"> ■RDエリア名 :RDDICT ■ページ長 :4096(デフォルト値) ■セグメント長 :5 ■HiRDBファイル名 :C:¥sysarea¥srdsys¥hidict ■セグメント数 :5000
<p>データディレクトリRDエリア</p> <ul style="list-style-type: none"> ■RDエリア名 :RDDIR ■ページ長 :4096(デフォルト値) ■セグメント長 :50 ■HiRDBファイル名 :C:¥sysarea¥srdsys¥hidir ■セグメント数 :50 	

■create rdarea文定義例



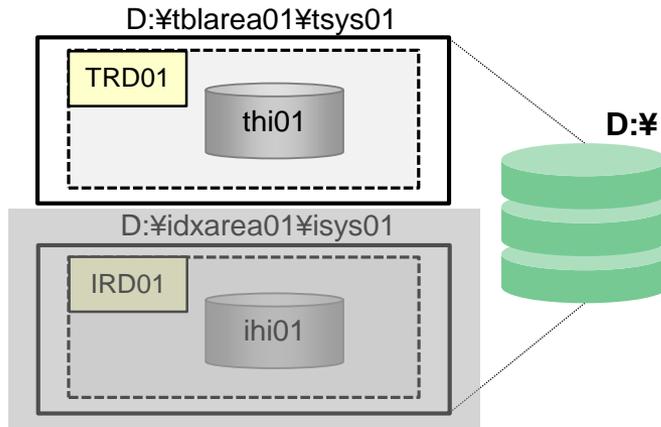
■コマンド設定例



解説

ユーザRDエリアは、HiRDBシステムの開始後にデータベース構成変更ユーティリティ(pdmod)で作成します。
引数として指定する制御文ファイル(create rdarea文)の定義例とコマンド設定例を以下に示します。

■ファイル環境 ユーザRDエリア



■設定項目(見積もりした値やファイル環境で決定した値)

■RDエリア名	:TRD01
■ページ長	:4096(デフォルト値)
■セグメント長	:110
■HiRDBファイル名	:D:\tblarea01\tsys01\thi01
■セグメント数	:17

■create rdarea文定義例

C:\createRd\tblRd.txt

```
create rdarea TRD01
for user used by PUBLIC
page 4096 characters
storage control segment 110 pages
file name "D:\tblarea01\tsys01\thi01"
initial 17 segments;
```

RDエリア名

RDエリア種別

ページ長

セグメント長

HiRDBファイル名

セグメント数

■コマンド設定例

pdmod -a C:\createRd\tblRd.txt

制御文ファイル名

5. データベース構築のポイント

5.1 概要

5.2 データベース構築手順

5.3 サーバ・ファイル環境の決定

5.4 パラメタ設定

5.5 データベース構築



解説 パラメタ設定で作成したものを配置またはコマンドの実行で作成します。

■ HiRDBの構築手順とポイント

5-3節

1. サーバマシン環境・データベース環境の決定

5-4節

2. パラメタ設定

3. HiRDB Serverのインストール

パラメタ設定段階で準備

4. システム定義ファイルの配置

システム共通定義、ユニット制御情報定義ファイル
シングルサーバ定義ファイルを運用ディレクトリ下のCONFディレクトリ(%PDDIR%¥CONF¥)に配置

5-5-3項

5. HiRDBシステム定義の整合性チェック

HiRDBシステム定義の内容チェック: pdconfchk -nコマンド

5-4-6項~5-4-7項

6. HiRDBファイルシステム領域の作成

HiRDBファイルシステム領域作成: pdfmkfsコマンド

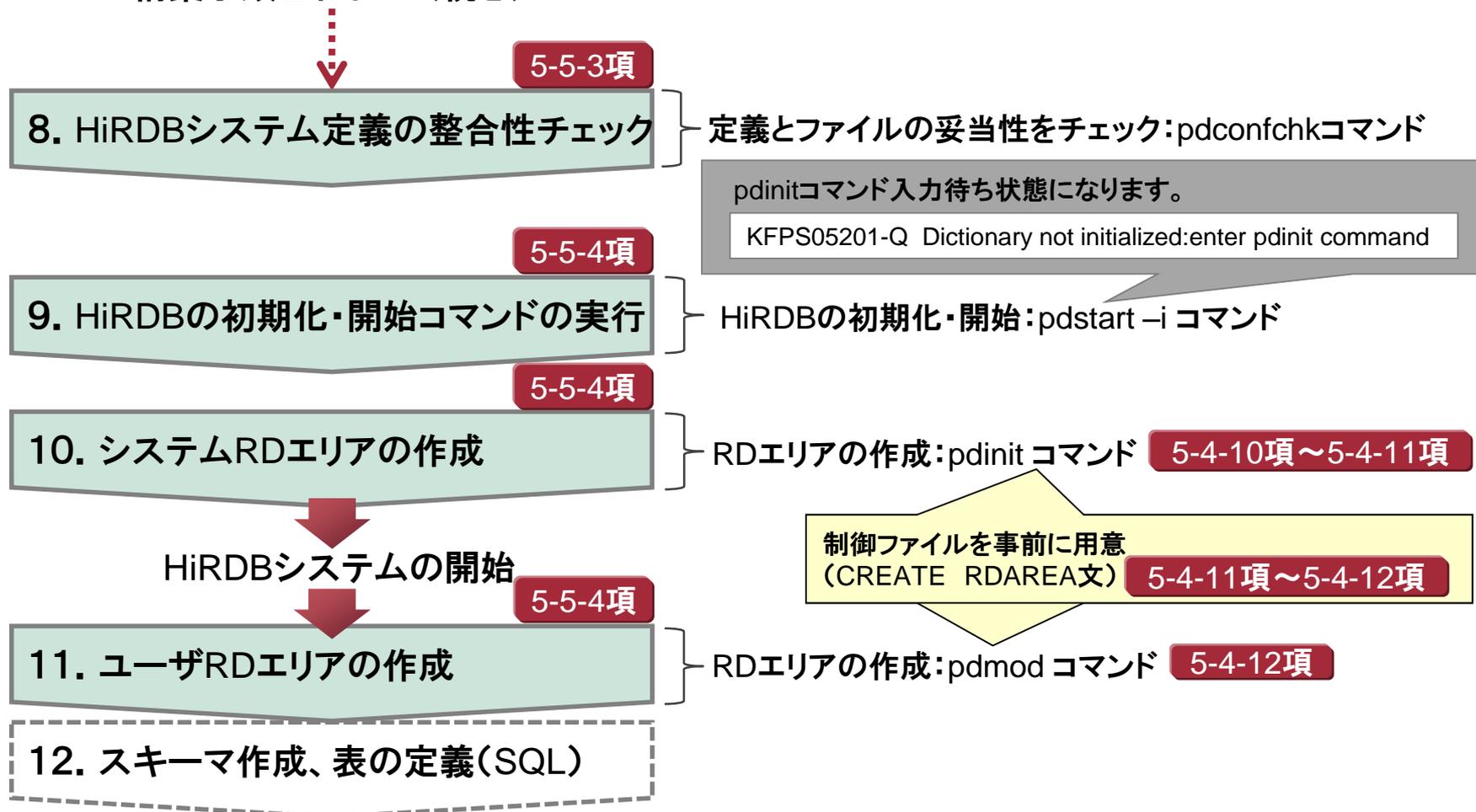
7. システムファイルの作成

システムログファイル : pdloginit -d sysコマンド
シンクポイントダンプファイル : pdloginit -d spdコマンド
ステータスファイル : pdstsinitコマンド

5-4-8項~5-4-9項

次へ続く

■HiRDBの構築手順とポイント(続き)



解説

HiRDBを起動する前に、HiRDBシステム定義の内容を確認するコマンドpdconfchkを使って、内容に誤りがないことを確認してください。内容確認は、2回に分けて実行します。

(1)システムファイル(*1)作成前

pdconfchkコマンドに-nオプションを付けて実行し、HiRDBシステム定義の内容のみチェックします。HiRDBシステム定義の内容に誤りがあると、システムファイルの作成でエラーになる可能性がありますので、必ず確認してください。

【コマンド実行例】 pdconfchk -n

(2)システムファイル(*1)作成後

pdconfchkコマンド(オプションなし)を実行し、HiRDBシステム定義の内容と、システムファイルの存在およびアクセス権限についてチェックします。

【コマンド実行例】 pdconfchk

(*1):システムログファイル、シンクポイントダンプファイル、ステータスファイル

■注意事項

HiRDBシステム定義のシステム共通定義(pdsys)のpdunit文で指定するホスト名および運用ディレクトリは、間違わないようにしてください。指定されたホスト名および運用ディレクトリの環境をチェックしますので、ここに誤りがあると、エラーメッセージが出力されず正しいチェックができません。ホスト名および運用ディレクトリは、コマンド実行前に確認してください。

解説 システムRDエリア、ユーザRDエリアの作成手順とHiRDBの開始の関係を以下に示します。

■作成手順

1. pdstart -i コマンドの実行(HiRDBシステムの初期化および開始)

```
C:¥>pdstart -i  
KFPS05201-Q Dictionary not initialized:enter pdinit command
```

HiRDBからのpdinitコマンドの実行要求(入力待ち)

2. 新規コマンドプロンプトでpdinit コマンドの実行(システムRDエリアの作成)

```
C:¥>pdinit -d C:¥createRd¥sysRd.txt  
KFPX24000-I DB initialize ended, return code=0
```

create rdarea文のテキストファイル

pdinitが正常終了

```
C:¥>pdstart -i  
KFPS05201-Q Dictionary not initialized:enter pdinit command  
KFPS05210-I HiRDB system initialization process complete
```

HiRDBシステムの初期化完了

-----HiRDB起動状態-----

3. pdmod コマンドの実行(ユーザRDエリアの作成)

```
C:¥>pdmod -a C:¥createRd¥tblRd.txt  
KFPX14250-I Processing of create rdarea statement ended ...  
KFPX24213-I DB modification for ended return code=0 ...  
KFPX24200-I DB modification ended, return code=0
```

create rdarea文のテキストファイル

pdmodが正常終了

6. 知っておくと役立つポイント

6.1 要件に応じたパラメタ設定

6.2 設計と構築を効率的に行うために

解説

5章では、HiRDBの起動に最低限必要なHiRDBサーバのパラメタ設定のみご紹介しました。本番環境においては、さらに要件に合わせた見直しが必要なパラメタがあります。

■ HiRDBのパラメタ設定項目

システム要件に合わせて、必ず下記の設定を見直ししてください。HiRDBには多くのパラメタがありますが、以下のように分類されます。

- ① 全システム共通で検討を推奨するパラメタ
- ② 特定のケースにおいて検討を推奨するパラメタ

分類	設定の種類	設定箇所
サーバ側の設定	HiRDBサーバの設定	システム定義ファイル
	OSの設定	OSのパラメタ
クライアント側の設定	HiRDBクライアントの設定	クライアント環境変数 UNIX: profile, .login, .cshrcなど Windows: ユーザ環境変数、HiRDB.iniなど

本節では、この中から①の代表的な項目についてご紹介します。

※その他の項目については、技術資料「ユーザ事例に学ぶ HiRDB設計のコツとトラブルシュート」でご紹介しています。

HiRDB技術資料のURL

https://www.hitachi.co.jp/Prod/comp/soft1/hirdb/files/tech_info/index.html#const opr

6-1-2 検討を推奨するパラメタ

分類	パラメタと設定値※	説明	検討事項
HiRDBの開始方法	pd_mode_conf = <u>MANUAL1</u> <u>MANUAL2</u> <u>AUTO</u>	HiRDBの起動モード	省略値は、手動開始(異常終了時も手動開始)になっています。 OSの起動時にHiRDBを自動的に開始したい場合は <u>AUTO</u> を設定してください。
通信処理	pd_service_port = <u>なし</u>	スケジューラプロセスのポート番号 (高速接続を適用する場合に指定)	高速接続機能の適用を検討してください。 (⇒6-1-3項)
システムログファイル	pd_log_unload_check = <u>Y</u> <u>N</u>	ログファイルのアンロードチェック 有無	システムログの運用方法について検討してください。 (⇒6-1-4~5項)
	pd_log_auto_unload_path = <u>なし</u>	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のアンロードログファイルの出力先ディレクトリ名	
	pd_log_remain_space_check = <u>warn</u> <u>safe</u>	システムログファイルの空き容量監視機能の動作を指定	
SQL最適化オプション	pd_optimize_level	SQL最適化オプション	SQL最適化オプションを見直してください。 (⇒6-1-6項)
	pd_additional_optimize_level	SQL拡張最適化オプション	
統計情報	pd_statistics = <u>A</u> <u>Y</u> <u>N</u> pdstbegin <設定されない> pd_stj_file_size = <u>102400</u>	統計情報の取得	09-50以降は、デフォルトで統計情報が出力されます。通常は、デフォルトの出力設定にて統計情報を取得してください。統計ログファイルの容量が十分かを確認して、pd_stj_file_sizeの指定値を調整してください。
拡張SQLエラー情報出力機能	pd_uap_exerror_log_use = <u>AUTO</u> <u>YES</u> <u>NO</u>	拡張SQLエラー情報出力機能の使用有無	サーバ側にもSQLエラー情報を出力します。エラーとなったSQLの調査が容易になり、さらに複数クライアントのSQLエラー情報を一元管理できます。 <u>YES</u> を指定すると、エラー発生時刻がマイクロ秒単位でわかります。

※ 設定値の下線は省略値を意味します。

解説

高速接続機能は、アプリケーション(クライアント)とHiRDBサーバ間の通信回数を削減して接続する機能です。下記のメリットがありますので、適用を検討してください。

■ 通常の接続の場合

- 接続のために、①～③の通信が発生します。

■ 高速接続の場合

- 高速接続を適用すると、
①のポート番号割り当て処理を省略して、
②③の通信のみとなるため、
通常の接続に比べて通信回数を削減できます。

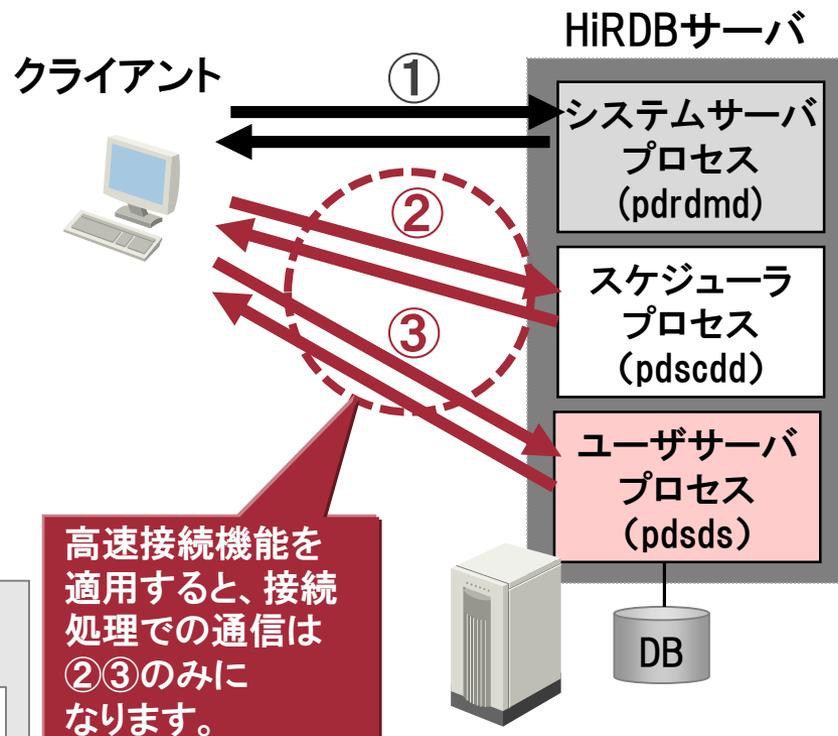
【設定するパラメタと推奨値】

pd_service_port = スケジューラプロセスのポート番号

クライアント側で、クライアント環境変数の設定も必要です。

- ・PDSERVICEPORT
- ・PDSERVICEGRP
- ・PDSRVTYPE (接続先のサーバマシンOSがWindowsおよびLinuxの場合)

【シングルサーバ構成の場合】



解説

システムログファイルが全て満杯になると、ユニットダウンします。これを防ぐために、検討が推奨されるパラメタについて説明します。

■ システムログのアンロード種別の指定

自動ログアンロードの適用を推奨します。

... 手動でログのアンロードを行う場合は、ログスワップ契機などでpdlogunloadコマンドを実行する運用が必要になります。

【設定するパラメタ】

パラメタ	pd_log_unload_check	pd_log_auto_unload_path
アンロード種別		
自動ログアンロード	Y(デフォルト値)	アンロードログファイルを格納するディレクトリ名※
手動ログアンロード	Y(デフォルト値)	省略
ログのアンロードを行わない	N	省略

推奨値

※【アンロードログファイルの格納先に関する注意事項】

- アンロードログファイル出力先ディレクトリの容量不足が発生した場合は、自動ログアンロード機能は停止します。ディスク容量に、注意してください。
- ディスク障害発生時に回復できるよう、RDエリアを格納するディスクとは異なるディスクに配置して下さい。

■ システムログファイルの空き容量監視機能の適用

予期しない長大トランザクションでシステムログが満杯になりユニットダウンするのを防ぐために、下記の設定をお奨めします。

【設定するパラメタと推奨値】

pd_log_remain_space_check=safe [省略値:warn]

省略値(warn)は、警告メッセージの表示のみです。推奨値(safe)を指定すると、さらに新規トランザクションの受付を抑制して、実行中のトランザクションを強制終了します。これにより、システムログファイルの空き容量を確保します。

解説

SQL最適化関連のパラメタの省略値は、推奨値とは異なっています。
新規システムの場合は、推奨値を設定するようにしてください。

■ VRアップ、システム更改、移行の場合

現行と同じ値を指定して下さい。

… 本パラメタの指定値を変更する場合は、SQLの性能に影響があるため、必ず動作確認をしてください。

■ 新規システムの場合

下記の**推奨値(v09-50以降のデフォルト値と同じ)**を設定してください。

● SQL最適化オプション

【設定するパラメタと推奨値】

```
pd_optimize_level=  
    "PRIOR_NEST_JOIN","PRIOR_OR_INDEXES","DETER_AND_INDEXES",¥  
    "RAPID_GROUPING","DETER_WORK_TABLE_FOR_UPDATE",¥  
    "APPLY_ENHANCED_KEY_COND","MOVE_UP_DERIVED_COND"
```

※: パラレルサーバの場合は、上記(single省略値)に加えて
"SORT_DATA_BES" "FLTS_ONLY_DATA_BES"を指定

● SQL拡張最適化オプション

【設定するパラメタと推奨値】

```
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"
```

6. 知っておくと役立つポイント

6.1 要件に応じたパラメタ設定

6.2 設計と構築を効率的に行うために

解説

システム定義のパラメタなど、各種設定ファイルを準備する作業を効率的に行うために、設計／構築支援ツールを提供しています。

■GUIで設計と構築作業を行えます。

- 簡易セットアップツール (⇒6-2-2項)
- HiRDB Configuration Assistant (⇒6-2-3項)

6-2-2 簡易セットアップツール

解説

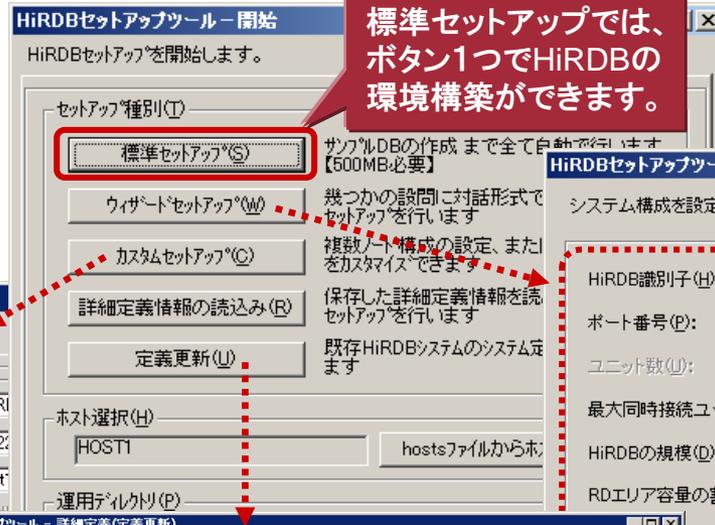
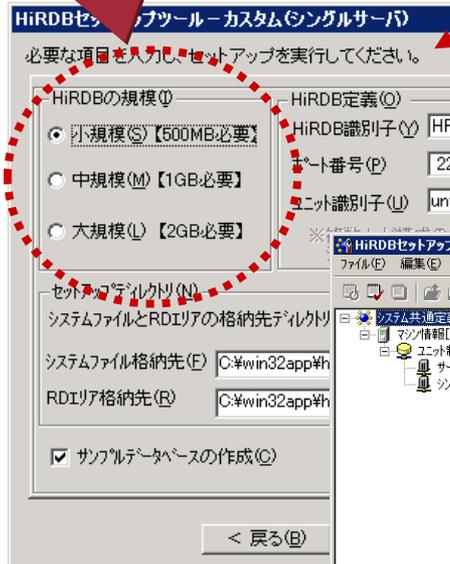
簡易セットアップツールは、HiRDBの環境構築を行うツールです。
HiRDB/Single Server、HiRDB/Parallel Serverの両方に対応しています。

全OSの
HiRDBサーバに
標準付属！

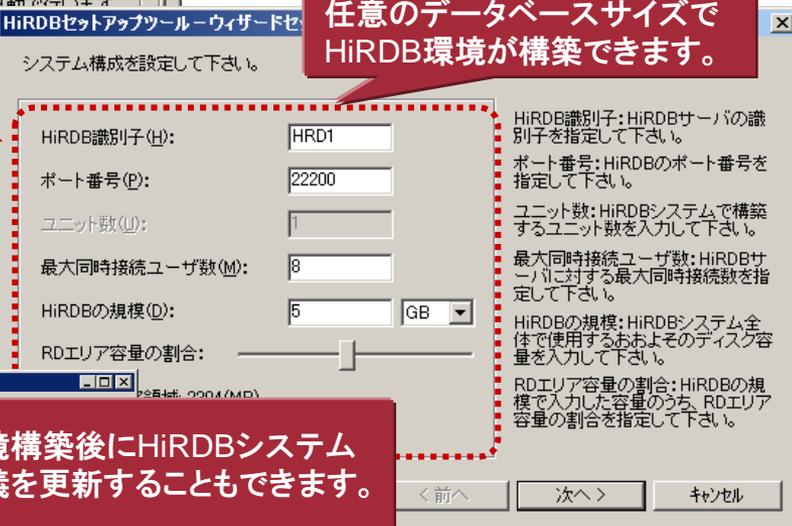
カスタムセットアップでは、小規模・中規模・大規模の3種類のテンプレートを選択できます。
【注意】HiRDBの規模が2GBを超える場合は、ウィザードセットアップをご使用ください。

標準セットアップでは、ボタン1つでHiRDBの環境構築ができます。

ウィザードセットアップでは、任意のデータベースサイズでHiRDB環境が構築できます。



環境構築後にHiRDBシステム定義を更新することもできます。



カスタマイズした環境を保存し、別マシンへの配布できます。

UNIX上のHiRDBに対しても、Windows上のツールから環境構築できます。(*1)

(*1) ツールが存在するWindowsマシンから、HiRDBが存在するUNIXマシンへリモートシェルが実行できる環境を、事前に設定してください。

7. おわりに

HiRDB

これからもHiRDBに
ご期待ください！



■本資料での表記

本資料では製品名称および名称について次のように表記しています。

ただし、それぞれのプログラムについての表記が必要な場合はそのまま表記しています。

- ・HP-UX, AIX, およびLinuxを総称してUNIXと表記します。
- ・Red Hat Enterprise LinuxをLinuxと表記します。
- ・Windows, Windows Serverを総称してWindowsと表記します。

■商標類

- ・ IBM, AIXは、世界の多くの国で登録されたInternational Business Machines Corporationの商標です。
- ・ Red Hat, and Red Hat Enterprise Linux are registered trademarks of Red Hat, Inc. in the United States and other countries. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.
- ・ 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。
- ・ Windowsおよび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。
- ・ その他記載の会社名、製品名は、それぞれの会社の商号、商標もしくは登録商標です。

■対象となる製品

記載の仕様は、HiRDB Version 9及びVersion 10です。

製品の改良により予告なく記載されている仕様が変更になることがあります。

END

HiRDB設計と構築のポイント

株式会社 日立製作所 サービス&プラットフォームビジネスユニット
サービスプラットフォーム事業本部 DB部

HITACHI
Inspire the Next 