
スケーラブルデータベースサーバ
HiRDB ファーストステップガイド
(UNIX(R) 用)

解説・手引・操作書

3000-6-254-10

マニュアルの購入方法

このマニュアル，および関連するマニュアルをご購入の際は，
巻末の用紙をご利用ください。

HITACHI

対象製品

適用 OS : HP-UX 11.0 , HP-UX 11i

P-1B62-1171 HiRDB/Single Server Version 7 07-00
P-1B62-1571 HiRDB/Single Server Version 7(64) 07-00
P-1B62-1B71 HiRDB/Run Time Version 7 07-00
P-1B62-1D71 HiRDB/Run Time Version 7(64) 07-00
P-1B62-1C71 HiRDB/Developer's Kit Version 7 07-00
P-1B62-1E71 HiRDB/Developer's Kit Version 7(64) 07-00
R-18434-31 HiRDB Control Manager - Agent 01-01

適用 OS : HI-UX/WE2

P-1662-117 HiRDB/Single Server Version 7 07-00
P-1662-1B7 HiRDB/Run Time Version 7 07-00
P-1662-1C7 HiRDB/Developer's Kit Version 7 07-00
適用 OS : Solaris 2.6 , Solaris 7 , Solaris 8 , Solaris 9
P-9D62-1171 HiRDB/Single Server Version 7 07-00
P-9D62-1B71 HiRDB/Run Time Version 7 07-00
P-9D62-1C71 HiRDB/Developer's Kit Version 7 07-00

適用 OS : Solaris 7 , Solaris 8 , Solaris 9

P-9D62-1571 HiRDB/Single Server Version 7(64) 07-00
P-9D62-1D71 HiRDB/Run Time Version 7(64) 07-00
P-9D62-1E71 HiRDB/Developer's Kit Version 7(64) 07-00

適用 OS : Solaris 8 , Solaris 9

R-19434-31 HiRDB Control Manager - Agent 01-01

適用 OS : AIX 5L V5.1 , AIX 5L V5.2

P-1M62-1171 HiRDB/Single Server Version 7 07-00
P-1M62-1571 HiRDB/Single Server Version 7(64) 07-00
P-1M62-1B71 HiRDB/Run Time Version 7 07-00
P-1M62-1D71 HiRDB/Run Time Version 7(64) 07-00
P-1M62-1C71 HiRDB/Developer's Kit Version 7 07-00
P-1M62-1E71 HiRDB/Developer's Kit Version 7(64) 07-00

R-1M434-31 HiRDB Control Manager - Agent 01-01

適用 OS : Red Hat Linux 7.1 , Red Hat Linux 7.2 , Red Hat Linux AS 2.1

P-9S62-1171 HiRDB/Single Server Version 7 07-00
P-9S62-1B71 HiRDB/Run Time Version 7 07-00
P-9S62-1C71 HiRDB/Developer's Kit Version 7 07-00

適用 OS : Red Hat Linux 7.2 , Red Hat Linux AS 2.1

R-1S434-31 HiRDB Control Manager - Agent 01-01

適用 OS : Windows 2000 , Windows XP Professional , Windows Server 2003

R-15434-11 HiRDB Control Manager - Console 01-01

R-15434-21 HiRDB Control Manager - Server 01-01

これらのプログラムプロダクトのほかにもこのマニュアルをご利用になれる場合があります。詳細は「ソフトウェア添付資料」または「ご使用になる前に」でご確認ください。

これらの製品は、ISO9001 および TickIT の認証を受けた品質システムで開発されました。

輸出時の注意

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

なお、ご不明な場合は、弊社担当営業にお問い合わせください。

商標類

AIX は、米国における米国 International Business Machines Corp. の登録商標です。

HP-UX は、米国 Hewlett-Packard Company のオペレーティングシステムの名称です。

Java 及びすべての Java 関連の商標及びロゴは、米国及びその他の国における米国 Sun Microsystems, Inc. の商標または登録商標です。

JDK は、米国およびその他の国における米国 Sun Microsystems, Inc. の商標または登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における登録商標あるいは商標です。

Microsoft は、米国およびその他の国における米国 Microsoft Corp. の登録商標です。

ODBC は、米国 Microsoft Corp. が提唱するデータベースアクセス機構です。

PC/AT は、米国 International Business Machines Corp. の商品名称です。

Red Hat は、米国およびその他の国で Red Hat, Inc. の登録商標若しくは商標です。

Solaris は、米国 Sun Microsystems, Inc. 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。

Sun Microsystems は、米国 Sun Microsystems, Inc. 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。

UNIX は、X/Open Company Limited が独占的にライセンスしている米国ならびに他の国における登録商標です。

Windows は、米国およびその他の国における米国 Microsoft Corp. の登録商標です。

Windows NT は、米国およびその他の国における米国 Microsoft Corp. の登録商標です。

Windows Server は、米国およびその他の国における米国 Microsoft Corp. の商標です。

X/Open は、X/Open Company Limited の英国ならびに他の国における登録商標です。

プログラムプロダクト「P-9D62-1171, P-9D62-1571, P-9D62-1B71, P-9D62-1C71, P-9D62-1D71, P-9D62-1E71, R-19434-31」には、米国 Sun Microsystems, Inc. が著作権を有している部分が含まれています。

プログラムプロダクト「P-9D62-1171, P-9D62-1571, P-9D62-1B71, P-9D62-1C71, P-9D62-1D71, P-9D62-1E71, R-19434-31」には、UNIX System Laboratories, Inc. が著作権を有している部分が含まれています。

発行

2002 年 11 月 (第 1 版) 3000-6-254

2004 年 3 月 (第 2 版) 3000-6-254-10

著作権

All Rights Reserved. Copyright (C) 2002, 2004, Hitachi, Ltd.

変更内容

変更内容 (3000-6-254-10)

追加・変更機能	変更箇所
HiRDB/Single Server Version 6 を中心とした記述から , HiRDB/Single Server Version 7 を中心とした記述に変更した。	全体
HiRDB Assist を使った運用を , HiRDB Control Manager を使った運用に変更した。	2.2.5 , 2.5 , 5
コマンドを使ったセットアップ手順を追加した。	2.3

単なる誤字・脱字などはお断りなく変更しました。

はじめに

このマニュアルは、次に示す製品の基本的な使い方について説明したものです。

- HiRDB/Single Server Version 7
- HiRDB/Single Server Version 7(64)
- HiRDB SQL Executer
- HiRDB/Run Time Version 7
- HiRDB/Run Time Version 7(64)
- HiRDB/Developer's Kit Version 7
- HiRDB/Developer's Kit Version 7(64)
- HiRDB Control Manager - Console
- HiRDB Control Manager - Server
- HiRDB Control Manager - Agent

対象読者

このマニュアルは、UNIX 版 HiRDB を初めて使用される方、UNIX 版 HiRDB の基本的な使い方を一とおり実行してみたいという方、または C 言語、COBOL 言語、もしくは Java 言語によるアプリケーションを作成する方を対象にしています。

マニュアルの構成

このマニュアルは、次に示す章と付録から構成されています。

第 1 章 ご使用になる前に

このマニュアルの使い方、使用するプログラムプロダクト、および構築する HiRDB システムについて説明しています。

第 2 章 HiRDB システムを構築してみましょう

HiRDB のインストール前の作業から、表にデータロードをするまでの作業について説明しています。

第 3 章 表を操作してみましょう

HiRDB SQL Executer を使用して表を操作する方法について説明しています。

第 4 章 HiRDB を運用してみましょう

HiRDB の運用方法について説明しています。

第 5 章 HiRDB Contol Manager で HiRDB を運用してみましょう

HiRDB Contol Manager を使用した HiRDB の運用方法について説明しています。HiRDB Contol Manager を操作するには Windows を使用します。

第 6 章 クライアントを接続してみましょう

クライアントから表を操作する方法について説明しています。

はじめに

第7章 アプリケーションを作成してみましょう

C 言語, COBOL 言語, または Java 言語を使用してアプリケーションを作成する方法について説明しています。

付録 A Q&A

よくある質問について Q&A 形式で説明しています。

付録 B HiRDB のアンインストール

HiRDB のアンインストールについて説明しています。

付録 C 用語解説

このマニュアルで使用している用語について説明しています。

関連マニュアル

このマニュアルの関連マニュアルを次に示します。必要に応じてお読みください。

- HiRDB Version 7 解説 (UNIX(R) 用)(3000-6-271)
- HiRDB Version 7 システム導入・設計ガイド (UNIX(R) 用)(3000-6-272)
- HiRDB Version 7 システム定義 (UNIX(R) 用)(3000-6-273)
- HiRDB Version 7 システム運用ガイド (UNIX(R) 用)(3000-6-274)
- HiRDB Version 7 コマンドリファレンス (UNIX(R) 用)(3000-6-275)
- HiRDB Version 7 UAP 開発ガイド (UNIX(R)/Windows(R) 用)(3000-6-276)
- HiRDB Version 7 SQL リファレンス (UNIX(R)/Windows(R) 用)(3000-6-277)
- HiRDB Version 7 メッセージ (UNIX(R)/Windows(R) 用)(3000-6-278)
- HiRDB Version 6 解説 (UNIX(R) 用)(3000-6-231)
- HiRDB Version 6 システム導入・設計ガイド (UNIX(R) 用)(3000-6-232)
- HiRDB Version 6 システム定義 (UNIX(R) 用)(3000-6-233)
- HiRDB Version 6 システム運用ガイド (UNIX(R) 用)(3000-6-234)
- HiRDB Version 6 コマンドリファレンス (UNIX(R) 用)(3000-6-235)
- HiRDB Version 6 UAP 開発ガイド (UNIX(R)/Windows(R) 用)(3000-6-236)
- HiRDB Version 6 SQL リファレンス (UNIX(R)/Windows(R) 用)(3000-6-237)
- HiRDB Version 6 メッセージ (UNIX(R)/Windows(R) 用)(3000-6-238)

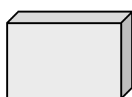
図中で使用する記号

このマニュアルの図中で使用する記号を次のように定義します。

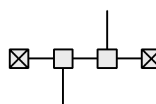
●PC/WSなどのマシン



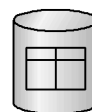
●プログラム



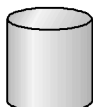
●バス形のLAN



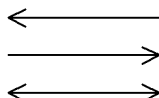
●表



●ファイル



●制御の流れ



●データの流れ



マニュアルの表記

このマニュアルでは、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、特に、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せん。

製品名称または名称	表記
HiRDB/Single Server Version 7, または HiRDB/Single Server Version 6	HiRDB/ シングルサーバ
HiRDB/Developer's Kit Version 7, または HiRDB/Developer's Kit Version 6	HiRDB/Developer's Kit
HiRDB/Run Time Version 7, または HiRDB/Run Time Version 6	HiRDB/Run Time
HiRDB/Developer's Kit HiRDB/Run Time	HiRDB/ クライアント
Java Development Kit	JDK
Microsoft(R) Windows(R) 2000 Server Operating System Microsoft(R) Windows(R) 2000 Advanced Server Operating System Microsoft(R) Windows(R) 2000 Professional Operating System Microsoft(R) Windows(R) 2000 Datacenter Server Operating System	Windows 2000
Microsoft(R) Windows(R) 95 Operating System	Windows 95
Microsoft(R) Windows(R) 98 Operating System	Windows 98
Microsoft(R) Windows(R) Millennium Edition Operating System	Windows Me
Microsoft(R) Windows NT(R) Workstation Operating System Version 4.0 Microsoft(R) Windows NT(R) Server Network Operating System Version 4.0	Windows NT

製品名称または名称	表記
Microsoft(R) Windows Server(TM) 2003, Standard Edition Microsoft(R) Windows Server(TM) 2003, Enterprise Edition	Windows Server 2003
Microsoft(R) Windows(R) XP Professional Operating System Microsoft(R) Windows(R) XP Home Edition Operating System	Windows XP
パーソナルコンピュータ (Personal Computer)	PC
ワークステーション (Workstation)	WS
PC または WS	マシン
ユーザアプリケーションプログラム (User Application Program)	UAP

常用漢字以外の漢字の使用について

このマニュアルでは、常用漢字を使用することを基本としていますが、次に示す用語については、常用漢字以外の漢字を使用しています。

- 個所 (かしょ)
- 同梱 (どうこん)
- 閉塞 (へいそく)

KB (キロバイト) などの単位表記について

1KB (キロバイト), 1MB (メガバイト), 1GB (ギガバイト), 1TB (テラバイト) はそれぞれ 1,024 バイト, 1,024² バイト, 1,024³ バイト, 1,024⁴ バイトです。

目次

1	ご使用になる前に	1
1.1	このマニュアルの使い方	2
1.1.1	マニュアルの目的	2
1.1.2	マニュアルの読み方	2
1.1.3	マシン環境	2
1.2	使用するプログラムプロダクト	4
1.3	構築する HiRDB システムの構成	7
1.4	このマニュアルを読むときの注意事項	9
2	HiRDB システムを構築してみましょう	11
2.1	この章を読むに当たって	12
2.2	HiRDB をインストールする	15
2.2.1	インストールする前にマシンの設定などを確認する	16
2.2.2	インストール前の準備をする	17
2.2.3	HiRDB/ シングルサーバをインストールする	18
2.2.4	HiRDB SQL Executer をインストールする	21
2.2.5	HiRDB Control Manager をインストールする	21
2.2.6	HiRDB/Run Time をインストールする (Windows マシン)	23
2.2.7	HiRDB SQL Executer をインストールする (Windows マシン)	24
2.2.8	HiRDB 運用ディレクトリを設定する	24
2.3	HiRDB の環境設定をする (コマンドでの実行)	26
2.3.1	HiRDB ファイルシステム領域を作成する	27
2.3.2	HiRDB システム定義を作成する	30
2.3.3	システムファイルを作成する	34
2.3.4	HiRDB を開始する	39
2.3.5	RD エリアを作成する	39
2.3.6	HiRDB の稼働状況を確認する	44
2.4	HiRDB の環境設定をする (システムジェネレータでの実行)	46
2.4.1	HiRDB の環境変数を設定する	46
2.4.2	システムジェネレータ (pdgen) を実行する	47
2.5	HiRDB Control Manager - Agent の環境設定をする	53
2.6	HiRDB SQL Executer の環境設定をする (Windows マシン)	57
2.6.1	hosts ファイルの設定をする	57

2.6.2	クライアント環境定義の設定をする	57
2.6.3	環境変数 PATH の設定をする	58
2.7	表を作成する	59
2.7.1	ユーザ名を登録する	59
2.7.2	表を定義する	60
2.7.3	表ヘデータロードをする	63

3

	表を操作してみましよう	71
3.1	表を操作するためには	72
3.2	HiRDB SQL Executer を起動する	73
3.3	表を操作する	74
3.3.1	演習 1 (表を検索する)	74
3.3.2	演習 2 (表へ行を挿入する)	74
3.3.3	演習 3 (表の行を更新する)	75
3.3.4	演習 4 (表の行を削除する)	76
3.4	HiRDB SQL Executer を終了する	77

4

	HiRDB を運用してみましよう	79
4.1	この章を読むに当たって	80
4.2	HiRDB を開始, 終了する	81
4.2.1	基礎知識	81
4.2.2	演習 1 (HiRDB を正常終了する)	83
4.2.3	演習 2 (HiRDB を正常開始する)	84
4.3	システムログを操作する	86
4.3.1	基礎知識	86
4.3.2	演習 (システムログをアンロードする)	91
4.4	バックアップを取得する	94
4.4.1	基礎知識	94
4.4.2	演習 1 (全 RD エリアのバックアップを取得する)	98
4.4.3	演習 2 (RD エリア単位のバックアップを取得する)	100
4.4.4	演習 3 (バックアップを DAT に取得する)	103
4.5	データベースを回復する	106
4.5.1	基礎知識	106
4.5.2	演習 1 (全 RD エリアをバックアップ取得時点に回復する)	108
4.5.3	演習 2 (RD エリアを最新の状態に回復する)	111

4.5.4	演習 3 (DAT のバックアップを使用して回復する)	116
4.6	表を再編成する	120
4.6.1	基礎知識	120
4.6.2	演習 (表を再編成する)	121
4.7	RD エリアを拡張する	124
4.7.1	基礎知識	124
4.7.2	演習 (RD エリアを拡張する)	125
4.8	RD エリアを追加する	128
4.8.1	基礎知識	128
4.8.2	演習 (RD エリアを追加する)	128
4.9	ユーザ権限を与える (ユーザを登録する)	136
4.9.1	基礎知識	136
4.9.2	演習 1 (新規ユーザを登録する)	137
4.9.3	演習 2 (アクセス権限がある表を検索する)	138
4.9.4	演習 3 (アクセス権限がない表を検索する)	139
4.9.5	演習 4 (権限がないユーザで表を検索する)	139
4.9.6	演習 5 (表の所有者を調べる)	140

5

	HiRDB Control Manager で HiRDB を運用してみましょう	143
5.1	この章を読むに当たって	144
5.2	HiRDB Control Manager を開始, 終了する	145
5.2.1	HiRDB Control Manager を開始する	145
5.2.2	HiRDB Control Manager を終了する	146
5.3	HiRDB を登録する	148
5.4	HiRDB を開始, 終了する	150
5.4.1	演習 1 (HiRDB を正常終了する)	150
5.4.2	演習 2 (HiRDB を正常開始する)	151
5.5	システムログを操作する	153
5.5.1	演習 (システムログをアンロードする)	154
5.6	バックアップを取得する	157
5.6.1	演習 1 (全 RD エリア (HiRDB システム) のバックアップを取得する)	158
5.6.2	演習 2 (RD エリアのバックアップを取得する)	166
5.7	データベースを回復する	175
5.7.1	演習 1 (全 RD エリアをバックアップ取得時点に回復する)	175
5.7.2	演習 2 (RD エリアを最新の状態に回復する)	178
5.8	表を再編成する	186

5.8.1 演習（表を再編成する）	186
5.9 RD エリアを拡張する	192
5.9.1 演習（RD エリアを拡張する）	192
5.10 RD エリアを追加する	198
5.10.1 演習（RD エリアを追加する）	198
5.11 ユーザ権限を与える（ユーザを登録する）	209
5.11.1 演習 1（新規ユーザを登録する）	209
5.11.2 演習 2（権限を追加する）	210

6

クライアントを接続してみましょう	215
------------------	-----

6.1 この章を読むに当たって	216
6.2 クライアントの環境設定をする	217
6.2.1 HiRDB/ クライアントをインストールする	217
6.2.2 HiRDB SQL Executer をインストールする	218
6.2.3 hosts ファイルの設定をする	219
6.2.4 クライアント環境定義の設定をする	219
6.2.5 環境変数 PATH の設定をする	220
6.3 クライアントから表を操作する	221
6.3.1 演習 1（表を定義する）	221
6.3.2 演習 2（表へ行を挿入する）	222
6.3.3 演習 3（ほかの所有者の表を検索する）	222

7

アプリケーションを作成してみましょう	225
--------------------	-----

7.1 この章を読むに当たって	226
7.2 作成するアプリケーションの仕様	230
7.2.1 サンプル UAP の機能	230
7.2.2 サンプル UAP の処理の流れ	231
7.3 C 言語でアプリケーションを作成する	233
7.3.1 基礎知識	233
7.3.2 HiRDB/ クライアントの環境設定をする	234
7.3.3 開発環境を準備する	234
7.3.4 UAP のソースプログラムを作成する	234
7.3.5 SQL プリプロセッサでソースプログラムを変換する	238
7.3.6 コンパイルを実行する	238
7.3.7 作成した UAP を実行する	239

7.4	COBOL 言語でアプリケーションを作成する	241
7.4.1	基礎知識	241
7.4.2	HiRDB/ クライアントの環境設定をする	242
7.4.3	開発環境を準備する	242
7.4.4	UAP のソースプログラムを作成する	243
7.4.5	SQL プリプロセサでソースプログラムを変換する	246
7.4.6	コンパイルを実行する	247
7.4.7	作成した UAP を実行する	247
7.5	Java 言語 (SQLJ) でアプリケーションを作成する	249
7.5.1	基礎知識	249
7.5.2	HiRDB/ クライアントの環境設定をする	250
7.5.3	開発環境を準備する	250
7.5.4	UAP のソースプログラムを作成する	250
7.5.5	SQL トランスレータでソースプログラムを変換する	254
7.5.6	コンパイルを実行する	254
7.5.7	作成した UAP を実行する	254

付録

	付録	257
	付録 A Q & A	258
	付録 A.1 環境設定に関する質問	258
	付録 A.2 HiRDB のシステム構成に関する質問	259
	付録 A.3 運用に関する質問	260
	付録 A.4 トラブルに関する質問	260
	付録 B HiRDB のアンインストール	263
	付録 B.1 HiRDB/ シングルサーバのアンインストール手順 (サーバ側)	263
	付録 B.2 HiRDB/Developer's Kit のアンインストール手順 (クライアント側)	263
	付録 B.3 HiRDB SQL Executer のアンインストール手順 (サーバ, クライアント両方)	264
	付録 B.4 HiRDB Control Manager のアンインストール手順	264
	付録 C 用語解説	266

索引

目次

図 1-1	学習用の HiRDB システムを構築するときのマシン環境	3
図 1-2	使用するプログラムプロダクト	4
図 1-3	構築する HiRDB システムの構成	7
図 2-1	HiRDB システムの構築手順	13
図 2-2	この節でインストールするプログラムプロダクトとインストール順	15
図 2-3	HiRDB ファイルシステム領域の構成	27
図 2-4	HiRDB/ シングルサーバの構成	31
図 2-5	RD エリアの構成	41
図 2-6	定義する表	59
図 2-7	CUSTOM 表へのデータロードの概要	64
図 2-8	VENDOR 表へのデータロードの概要	67
図 4-1	システムログとデータベース回復の関係	87
図 4-2	システムログファイルの状態変化	89
図 4-3	システムログをアンロードする手順	91
図 4-4	データベース回復の概要	94
図 4-5	pdcopy コマンドの処理結果リストの出力例	96
図 4-6	バックアップを取得する手順	97
図 4-7	自動ログアンロード機能を使用する場合のバックアップを取得する手順	98
図 4-8	データベースを回復する手順	107
図 4-9	表の再編成	120
図 4-10	表を再編成する手順	121
図 4-11	RD エリアを拡張する手順	124
図 4-12	RD エリアを追加する手順	128
図 6-1	クライアントの接続手順	216
図 7-1	UAP 作成の流れ	227
図 7-2	サンプル UAP の機能の概要	231
図 7-3	サンプル UAP の流れ	232
図 7-4	UAP 生成の流れ (C 言語の場合)	233
図 7-5	UAP 生成の流れ (COBOL 言語の場合)	241
図 7-6	UAP 生成の流れ (Java 言語の場合)	249

表目次

表 1-1	使用するプログラムプロダクト	5
表 4-1	HiRDB の開始モード	81
表 4-2	HiRDB の終了モード	82
表 4-3	システムログの運用の種類	88
表 4-4	バックアップの取得単位	95
表 4-5	バックアップ取得モード	95
表 4-6	ユーザ権限の種類	136
表 7-1	使用できる言語と開発環境	228

1

ご使用になる前に

この章では、このマニュアルの使い方、使用するプログラムプロダクト、および構築する HiRDB システムの構成について説明します。

1.1 このマニュアルの使い方

1.2 使用するプログラムプロダクト

1.3 構築する HiRDB システムの構成

1.4 このマニュアルを読むときの注意事項

1.1 このマニュアルの使い方

このマニュアルの目的，読み方，およびこのマニュアルで行う演習のマシン環境について説明します。

1.1.1 マニュアルの目的

このマニュアルは，マシンを操作しながら，HiRDB システムの基本的な操作を学習するためのものです。

HiRDB の基本的な操作として，このマニュアルでは次の項目について説明しています。

- HiRDB のインストールから表の操作まで（2，3 章）
- 基礎的な運用（4，5 章）
- 別マシンのクライアントの接続（6 章）
- C 言語，COBOL 言語，または Java 言語による UAP の作成（7 章）

作成する HiRDB システムは，HP-UX 版の HiRDB/ シングルサーバです。

なお，このマニュアルで作成する HiRDB システムは学習用のものです。したがって，本番用の HiRDB システムを作成する場合は，HiRDB の各マニュアルを参照してください。本番用の HiRDB システムとは，実際に業務で使用する HiRDB システムです。

1.1.2 マニュアルの読み方

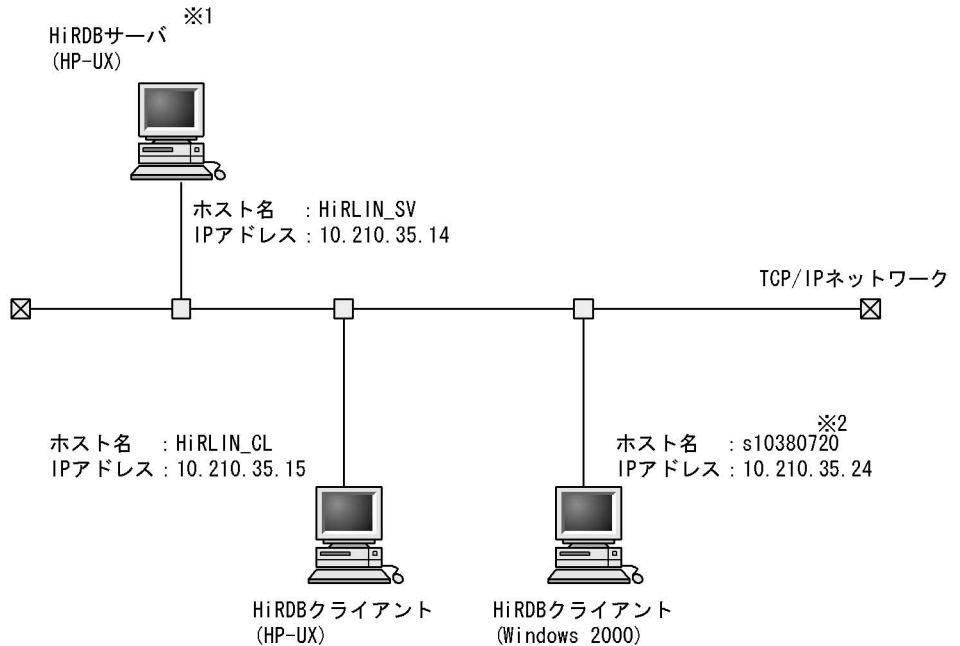
このマニュアルは，通読型のマニュアルです。

マニュアルの構成に従って，実際にマシンを操作しながらご利用ください。

1.1.3 マシン環境

このマニュアルでは，実際に学習用の HiRDB システムを構築します。学習用の HiRDB システムを構築するときのマシン環境を図 1-1 に示します。なお，使用する OS については，HiRDB サーバは HP-UX，HiRDB クライアントは HP-UX です。HiRDB Control Manager を利用する場合，Windows のクライアントも使用します。

図 1-1 学習用の HiRDB システムを構築するときのマシン環境



注※1

バックアップをDATに取得する場合、HiRDBサーバにDATドライブが必要です。

注※2

HiRDB Control Managerを運用する場合に使用します。HiRDB Control Managerの使用方法については、「5. HiRDB Control ManagerでHiRDBを運用してみましょう」を参照してください。

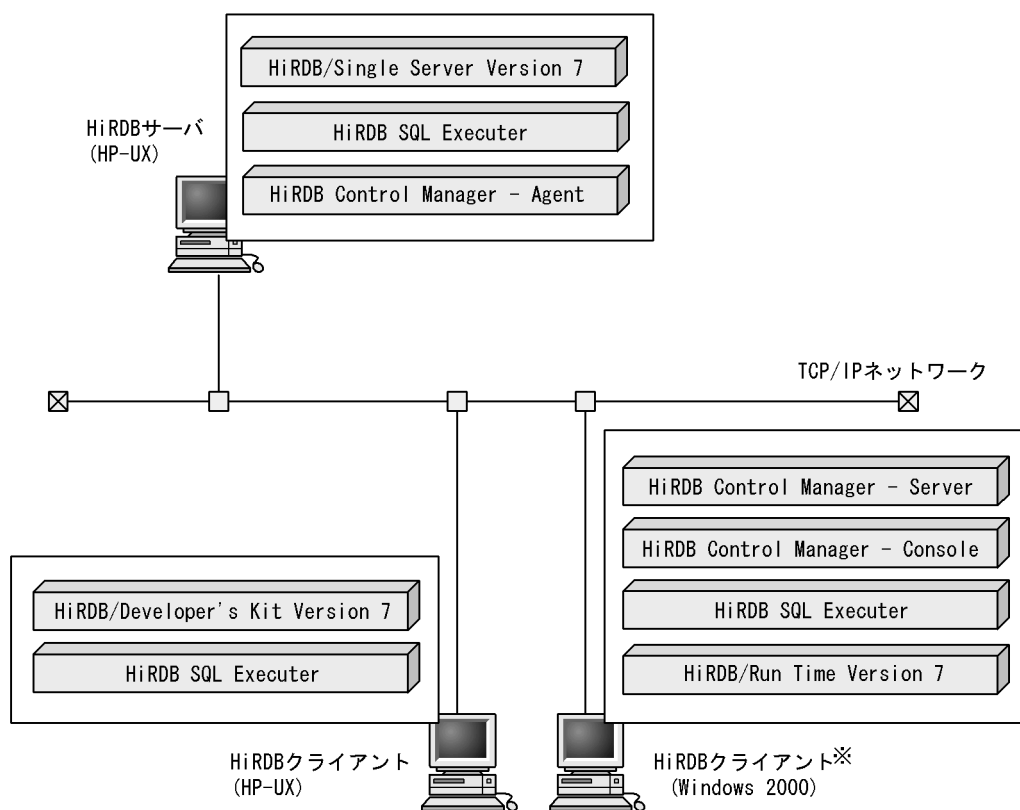
なお、HI-UX/WE2では、Java言語 (SQLJ) によるUAPの作成はできません。また、HiRDB Version 7以外をご使用の場合に必要な環境については、別途ご確認ください。

1.2 使用するプログラムプロダクト

このマニュアルでは、HiRDB/Single Server Version 7 07-00 を使用して HiRDB システムを構築しています（そのほかのプログラムプロダクトのバージョンについては、表 1-1 を参照してください）。HiRDB/Single Server Version 6 を使用する場合でも、基本的な操作などは変わりません。ただし、インストールや環境設定などで使用できるプログラミング言語に一部分差異があります。

使用するプログラムプロダクトを図 1-2 および表 1-1 に示します。

図 1-2 使用するプログラムプロダクト



注※

HiRDB Control Manager を運用する場合に使用します。HiRDB Control Manager の使用方法については、「5. HiRDB Control Manager で HiRDB を運用してみましょう」を参照してください。

表 1-1 使用するプログラムプロダクト

プログラムプロダクト名称	バージョン	説明	インストール先のマシン		
			サーバ	HP-UX クライアント	Windows クライアント
HiRDB/Single Server Version 7	07-00	HiRDB のサーバです。HiRDB システムを管理するプログラムプロダクトです。 このマニュアルでは、HiRDB/Single Server Version 7 を使用します。		-	-
HiRDB SQL Executer	02-04	会話形式で SQL を実行するプログラムプロダクトです。			1
HiRDB Control Manager - Agent	01-01	HiRDB Control Manager のエージェントです。HiRDB Control Manager - Server を経由して、HiRDB Control Manager - Console の命令を受け、同一マシンで稼働している HiRDB サーバを制御します。	2	-	-
HiRDB/Developer's Kit Version 7	07-00	HiRDB のクライアントです。HiRDB サーバがあるマシンとは別のマシンから HiRDB にアクセスするとき、および HiRDB サーバがあるマシンとは別のマシンで UAP を作成するとき使用するプログラムプロダクトです。 このマニュアルでは、HiRDB/Developer's Kit Version 7 を使用します。	-	3	-
HiRDB Control Manager - Console	01-01	HiRDB Control Manager のクライアントです。HiRDB Control Manager を使うと、HiRDB サーバのシステムログファイルの操作やバックアップの取得などの運用を GUI で実行できます。 HiRDB Control Manager - Console は Windows マシンで使用します。	-	-	2

1. ご使用になる前に

プログラムプロダクト名称	バージョン	説明	インストール先のマシン		
			サーバ	HP-UX クライアント	Windows クライアント
HiRDB Control Manager - Server	01-01	HiRDB Control Manager のサーバです。HiRDB Control Manager - Console からの処理命令を HiRDB Control Manager - Agent (HiRDB サーバ) に送ります。HiRDB Control Manager - Console は Windows マシンで使用します。	-	-	2
HiRDB/Run Time Version 7	07-00	HiRDB のクライアントです。HiRDB サーバがあるマシンとは別のマシンから HiRDB にアクセスするときに使用するプログラムプロダクトです。このマニュアルでは、Windows マシンから HiRDB SQL Executer で HiRDB サーバにアクセスするために使用します。	-	-	1

(凡例)

: インストールが必要です。

- : インストールは不要です。

注 1

HiRDB Control Manager を使用した運用で、HiRDB Control Manager から Windows 版 HiRDB SQL Executer を起動する場合に必要です。

注 2

HiRDB Control Manager を使用して HiRDB を運用する場合に必要です。コマンドだけで HiRDB の運用を行う場合には必要ありません。

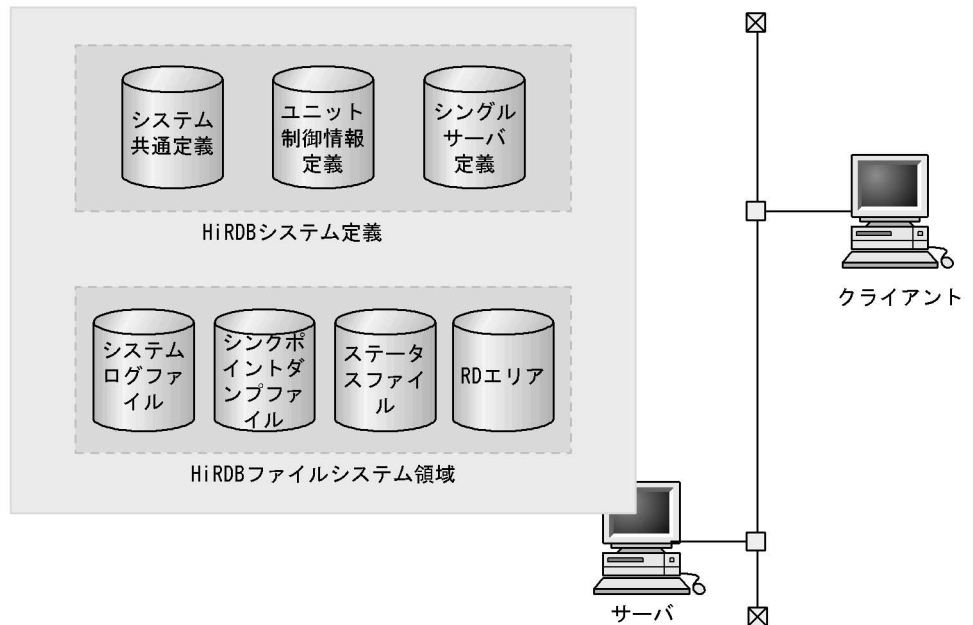
注 3

クライアントで UAP を作成する場合に必要です。UAP を作成しない場合には、このプログラムプロダクトの代わりに HiRDB/Run Time Version 7 を使用することができます。

1.3 構築する HiRDB システムの構成

構築する HiRDB システムの構成を図 1-3 に示します。

図 1-3 構築する HiRDB システムの構成



(1) HiRDB システム定義

HiRDB システムの構成や環境を定義したものを、HiRDB システム定義といいます。HiRDB システム定義には、システム共通定義、ユニット制御情報定義、およびシングルサーバ定義があります。

(2) HiRDB ファイルシステム領域

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、および RD エリアを作成する領域のことを HiRDB ファイルシステム領域といいます。また、HiRDB ファイルシステム領域内に作成されるファイルを HiRDB ファイルといいます。

HiRDB ファイルシステム領域は、それぞれシステムファイル用、作業表用ファイル用、および RD エリア用の用途ごとに作成します。

(3) システムログファイル

データベースの更新履歴情報を格納するためのファイルを、システムログファイルといいます。また、このデータベースの更新履歴情報をシステムログといいます。システムログは、一般的にはジャーナルとも呼ばれています。

1. ご使用になる前に

HiRDBまたはUAPが異常終了した場合、HiRDBがデータベースを回復するときに使用されます。また、ユーザがデータベースを回復するときの入力情報にも使用します。

(4) シンクポイントダンプファイル

HiRDBが異常終了した場合、システムログだけで回復処理をすると、HiRDB開始からのすべてのシステムログが必要となり、回復に多大な時間が掛かります。そこで、HiRDB稼働中に一定の間隔でポイントを設けて、そのポイントで回復する必要のあるHiRDB管理情報を保存することで、ポイント以前のシステムログは不要になり、回復時間を短縮できます。このポイントで取得するHiRDB管理情報を格納するファイルを、シンクポイントダンプファイルといいます。

(5) ステータスファイル

HiRDBがシステムを再開するときに必要なとする、システムステータス情報を格納するファイルを、ステータスファイルといいます。

ステータスファイルには、ユニット単位に作成するユニット用ステータスファイルと、サーバ単位に作成するサーバ用ステータスファイルがあります。

(6) RD エリア

表、インデクスなどの情報を格納する領域をRDエリアといいます。

このマニュアルで構築するHiRDBシステムでは、次のRDエリアを作成します。

マスタディレクトリ用RDエリア：

次の情報を格納します。

- ディクショナリ表、およびユーザが作成した表やインデクスを格納するRDエリアの情報
- RDエリアの登録場所（サーバ）の情報

データディレクトリ用RDエリア：

インデクスに関する情報（列名、データ型など）を、HiRDBのデータ形式で格納します。

データディクショナリ用RDエリア：

定義系SQLの解析結果を管理するディクショナリ表、およびディクショナリ表のインデクスを格納します。

ユーザ用RDエリア：

ユーザが作成する表とインデクスを格納します。

1.4 このマニュアルを読むときの注意事項

HiRDB システムの構築を始める前に、次のことに留意してください。

(1) コマンドの実行

HiRDB のコマンドは、コマンドラインから入力します。

このマニュアルでは、Bourne シェルを使用していると仮定しています。また、コマンド実行例ではプロンプトを「\$」としています。

(2) OS ごとの操作の差異

このマニュアルでは、HP-UX の環境に HiRDB/ シングルサーバを構築するため、HP-UX での操作方法を記述しています。したがって、OS パラメタ（カーネルパラメタ）の設定方法のように、ほかの OS で操作方法が異なる場合があります。HP-UX 以外の OS をご使用の場合は、適宜ご使用の OS の操作方法に読み替えてください。

(3) HiRDB が出力するメッセージ

HiRDB のメッセージ（運用コマンドの出力メッセージを除く）は、syslog ファイルに出力されます。障害が発生したときなどは、この syslog ファイルを見るようにしてください。

(4) ホームディレクトリ（\$HOME）の記述

このマニュアルでは、コマンドの入力画面例などでユーザのホームディレクトリを \$HOME と表記している箇所があります。これは、出力例などでは /home/ ユーザ ID という形式で展開されます。例えば、HiRDB 管理者「hirman」が操作している場合、/home/hirman となります。

(5) 製品の画面や設定値について

このマニュアルに記載されている製品の画面は、ご使用のバージョンによっては、実際の画面と違う部分がある場合があります。デフォルトの設定値についても同様です。

ご使用に当たっては、ご使用のプログラムのマニュアルやヘルプを合わせてご参照ください。

2

HiRDB システムを構築してみましよう

この章では、HiRDB のインストール前の作業から、表にデータロードをするまでの作業について説明します。

-
- 2.1 この章を読むに当たって

 - 2.2 HiRDB をインストールする

 - 2.3 HiRDB の環境設定をする（コマンドでの実行）

 - 2.4 HiRDB の環境設定をする（システムジェネレータでの実行）

 - 2.5 HiRDB Control Manager - Agent の環境設定をする

 - 2.6 HiRDB SQL Executer の環境設定をする（Windows マシン）

 - 2.7 表を作成する
-

2.1 この章を読むに当たって

この章では、HiRDB システムの構築について説明しています。図 2-1 の手順に従ってお読みください。

図 2-1 HiRDB システムの構築手順



注※1 HiRDB Control Managerを使用して運用する場合に必要です。HiRDB Control Managerを使用しない場合は必要ありません。

注※2 どちらの方法で環境設定しても、同じ内容になります。

2. HiRDB システムを構築してみましょう

HiRDB システムを構築するときのユーザ名について

HiRDB システムを構築するときのユーザ名について説明します。各ユーザがどのような作業をしているのかについて次に示します。

root (スーパーユーザ)

使用する OS の管理者 (root) 権限を持っている, OS のユーザです。HiRDB のインストールから, HiRDB の環境設定の前までを行います。

hirman

HiRDB の管理のために OS に登録するユーザです。root (スーパーユーザ) が HiRDB を管理するために必要な権限を設定します。HiRDB の環境設定や HiRDB のコマンドを実行します。

このユーザ「hirman」が, 表の作成演習をするユーザ「MANAGER」に対して DBA 権限を与えます。

MANAGER

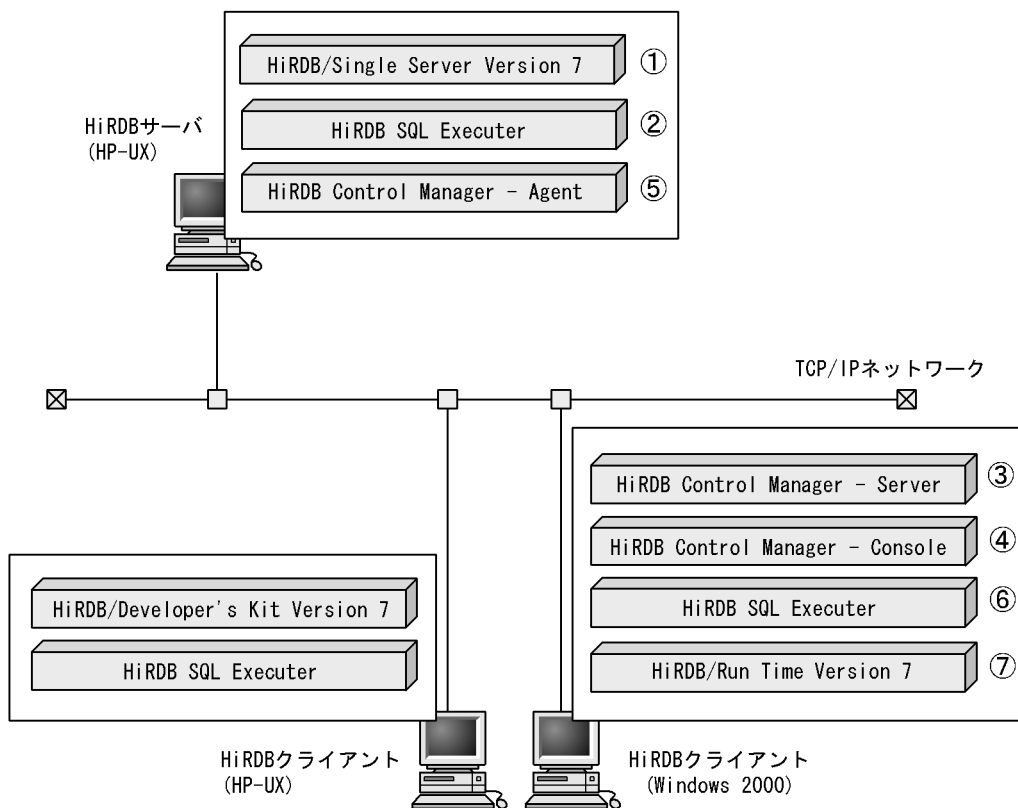
表の作成 (表の定義, データロード), および表の操作演習をする HiRDB のユーザです。

2.2 HiRDB をインストールする

HiRDB サーバ、および HiRDB Control Manager を起動する HiRDB クライアント (Windows 2000) へのプログラムプロダクトのインストールについて説明します。HiRDB クライアント (HP-UX) へのプログラムプロダクトのインストールについては、「6. クライアントを接続してみましょう」を参照してください。

この節でインストールするプログラムプロダクトとインストール順を図 2-2 に示します。

図 2-2 この節でインストールするプログラムプロダクトとインストール順



次の順番でインストールしてください。

- ① 「2.2.3 HiRDB/シングルサーバをインストールする」で説明します。
- ② 「2.2.4 HiRDB SQL Executerをインストールする」で説明します。
- ③ 「2.2.5 HiRDB Control Managerをインストールする」で説明します。
- ④ 「2.2.5 HiRDB Control Managerをインストールする」で説明します。
- ⑤ 「2.2.5 HiRDB Control Managerをインストールする」で説明します。
- ⑥ 「2.2.6 HiRDB/Run Timeをインストールする(Windowsマシン)」で説明します。
- ⑦ 「2.2.7 HiRDB SQL Executerをインストールする(Windowsマシン)」で説明します。

なお、インストールは、root (スーパーユーザ)、または Windows 2000 ログイン時のユー

2. HiRDB システムを構築してみましょう

ザ「Administrator」で行います。

2.2.1 インストールする前にマシンの設定などを確認する

HiRDB をインストールする前に、次の項目について確認してください。

- OS パラメタ (カーネルパラメタ)
- 空きディスク容量
- 共用メモリ

(1) OS パラメタ (カーネルパラメタ)

HiRDB を構築するためには、共有メモリなどの OS パラメタ (カーネルパラメタ) をカスタマイズする必要があります。カスタマイズの手順を次に示します。

1. スーパユーザとしてログインします。
2. OS パラメタ (カーネルパラメタ) を次に示す推奨値よりも大きな値に設定してください。HP-UX の場合、値の変更については、システム管理マネージャ (SAM) を利用できます。SAM の使用方法および OS パラメタについては、HP-UX のマニュアルを参照してください。

! 注意事項

この値は、ファーストステップガイドで行う簡易導入用の値です。データベースを詳細に設計する場合、または HP-UX 以外の OS で使用する場合、マニュアル「HiRDB Version 7 システム導入・設計ガイド (UNIX(R) 用)」の「オペレーティングシステムパラメタの見積もり」に関する記述を参照してください。

OS パラメタ	推奨値
maxuprc	MAX (pd_max_server_process の値 + e , 512) 以上
maxusers	128
nfile	MAX { 1600 , 320 × (h - g) + [a + (b × c) + 320] × g + 848 × i + h × 2 + 227 + k × m } 以上
nproc	MAX (pd_max_server_process の値 + 20 , 576)
semnmi	1,024
semnms	7,200
semnmu	512
semume	512
shmmax	200 [MB]

OS パラメタ	推奨値
shmmni	1,000
shmseg	120

(2) 空きディスク容量

このマニュアルの演習を実施するためには、インストールのためのディスク容量、および HiRDB を運用するためのディスク容量が必要になります。

インストールするために必要なディスク容量は、220MB です。また、このマニュアルでは、システムファイルがインストールされているのとは別のパーティションである / HiRDB に、HiRDB ファイルシステム領域を作成して作業します。それらの操作を行うため必要な /HiRDB のディスク容量は 500MB です。それぞれのパーティションの空きディスク容量を確認してください。

(3) 共用メモリ

このマニュアルで想定している HiRDB システムを構築する場合、50MB 以上の共用メモリが必要となります。

2.2.2 インストール前の準備をする

HiRDB をインストールする前に、root (スーパーユーザ) で次のことを行います。

- (1) IP アドレスの設定
- (2) HiRDB 管理者専用グループの作成
- (3) HiRDB 管理者の登録

(1) IP アドレスの設定

ネットワーク上で HiRDB にアクセスできるようにするために、使用するすべてのマシンの IP アドレスおよびホスト名を /etc/hosts ファイルに設定します。すでに設定されている場合は必要ありません。

/etc/hosts ファイル

```
127.0.0.1      localhost localhost.localdomain
10.210.35.14   HiRLIN_CL.soft.hitachi.co.jp HiRLIN_SV
10.210.35.15   HiRLIN_SV.soft.hitachi.co.jp HiRLIN_CL
10.210.35.24   s10380720.soft.hitachi.co.jp s10380720
```

2. HiRDB システムを構築してみましょう

(2) HiRDB 管理者専用グループの作成

HiRDB 管理者の専用グループを OS に登録します。このマニュアルでは、次のように HiRDB 管理者用グループを設定します。

HP-UX の場合、システム管理マネージャ (SAM) を利用できます。SAM の使用方法およびグループ登録については、HP-UX のマニュアルを参照してください。

項目	設定値
HiRDB 管理者専用グループ	hirdb
グループ ID	505
グループのメンバー (HiRDB 管理者)	hirman

(3) HiRDB 管理者の登録

HiRDB を管理するユーザの情報を OS に登録します。このマニュアルでは、次のように HiRDB 管理者用グループを設定します。

HP-UX の場合、システム管理マネージャ (SAM) を利用できます。SAM の使用方法およびユーザ登録については、HP-UX のマニュアルを参照してください。

項目	設定値
HiRDB 管理者ユーザ	hirman
ユーザ ID	503
所属するグループ ID	505
ホームディレクトリ	/home/hirman
ログイン時に実行されるシェル	/bin/sh

2.2.3 HiRDB/ シングルサーバをインストールする

HiRDB/ シングルサーバをインストールします。

HiRDB/ シングルサーバのインストールは、root 権限があるユーザが実行してください。root 権限がないユーザがインストールすると、エラーとなり、インストーラを中止します。

なおここでは、HP-UX にインストールする手順を示しますが、その他の OS にインストールする場合も基本的な操作は変わりません。

インストール手順

1. CD-ROM をマウントしてください (以降、マウント先を /cdrom とします)。
2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してくだ

2. HIRDB システムを構築してみましょう

さい。
次の画面が表示されます。

```
Hitachi PP Installer 03-00

L) List Installed Software.
I) Install Software.
D) Delete Software.
Q) Quit.

Select Procedure ==>

+-----+
CAUTION!
YOU SHALL INSTALL AND USE THE SOFTWARE PRODUCT LISTED IN THE
"List Installed Software." UNDER THE TERMS AND CONDITION OF
THE SOFTWARE LICENSE AGREEMENT ATTACHED TO SUCH SOFTWARE PRODUCT.
+-----+

All Rights Reserved, Copyright (C) 1994, 2001, Hitachi, Ltd.
```

3. < I > キーを押下してください。
インストールする PP の選択画面が表示されます。

```
PP-No.          VR      PP-NAME
001 P-1B62-1171 0700/G HiRDB/Single Server Version 7

F) Forward B) Backward J) Down K) Up Space) Select/Unselect I) Install Q) Quit
```

4. < スペース > キーを押下してください。
HiRDB/Single Server Version 7 の行頭に、その PP を選択したことを示す
「<@>」が表示されます。

2. HiRDB システムを構築してみましょう

```
PP-No.          VR      PP-NAME
<@>001 P-1B62-1171 0700/G HiRDB/Single Server Version 7

F) Forward B) Backward J) Down K) Up Space) Select/Unselect I) Install Q) Quit
```

5. < I >キーを押下してください。
インストール確認のメッセージが出力されます。

```
PP-No.          VR      PP-NAME
<@>001 P-1B62-1171 0700/G HiRDB/Single Server Version 7

F) Forward B) Backward J) Down K) Up Space) Select/Unselect I) Install Q) Quit
Install PP? (y: install, n: cancel) ==>
```

6. < Y >キーを押下してください。
インストールが開始されます。インストールが終了すると、「Installation completed.」のメッセージが表示されます。

```
PP-No.          VR      PP-NAME
<@>001 P-1B62-1171 0700/G HiRDB/Single Server Version 7

F) Forward B) Backward J) Down K) Up Space) Select/Unselect I) Install Q) Quit
Installation completed.
```

7. < Q > キーを 2 回押下してください。

2.2.4 HiRDB SQL Executer をインストールする

HiRDB SQL Executer は、会話形式で SQL を実行するプログラムです。HiRDB で使用できる SQL のほかに、独自のコマンドなども用意しています。

HiRDB/ シングルサーバをインストールしたマシンに、HiRDB SQL Executer をインストールします。HiRDB SQL Executer のインストール手順を次に示します。

なおここでは、HP-UX にインストールする手順を示しますが、その他の OS にインストールする場合も基本的な操作は変わりません。

1. CD-ROM をマウントしてください（以降、マウント先を /cdrom とします）。
2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してください。
3. 「Install Software」から「HiRDB SQL Executer」を選択して、インストールしてください。

2.2.5 HiRDB Control Manager をインストールする

HiRDB Control Manager をインストールします。HiRDB Control Manager を使って HiRDB を運用しない場合は、この作業を行う必要はありません。

HiRDB Control Manager は、システムログファイルの操作やバックアップの取得などの運用を GUI で実行するための運用支援プログラムです。HiRDB Control Manager は、次の三つのプログラムプロダクトで構成されています。

HiRDB Control Manager - Console

HiRDB Control Manager のクライアントです。GUI を使って HiRDB の開始、終了、バックアップの取得などの操作ができます。

HiRDB Control Manager - Console は Windows 専用プログラムです。したがって、HiRDB Control Manager を使う場合、UNIX 版 HiRDB を Windows から操作する形態になります。このマニュアルでは、Windows マシンを用意して、HiRDB Control Manager 用の操作端末にします。

HiRDB Control Manager - Server

HiRDB Control Manager のサーバです。HiRDB Control Manager - Console からの処理命令を HiRDB Control Manager - Agent (HiRDB サーバ) に送ります。

HiRDB Control Manager - Server は Windows 専用プログラムです。このマニュアルでは、HiRDB Control Manager - Console と同じ Windows マシンにインストールします。

HiRDB Control Manager - Agent

2. HiRDB システムを構築してみましょう

HiRDB Control Manager のエージェントです。HiRDB Control Manager - Server を経由して、HiRDB Control Manager - Console の命令を受け、同一マシンで稼働している HiRDB サーバに実行させます。

HiRDB Control Manager - Console で操作したい HiRDB サーバがインストールされているマシンにインストールします。

HiRDB Control Manager のインストール手順を次に示します。HiRDB Control Manager は、HiRDB Control Manager - Server、HiRDB Control Manager - Console、HiRDB Control Manager - Agent の順序でインストールする必要があります。

(1) HiRDB Control Manager - Server のインストール方法 (Windows マシン)

1. CD-ROM を挿入して、「10_CMS¥DISK1」の「SETUP.EXE」をダブルクリックしてください。
インストーラが起動されます。
2. 以降は、画面の内容に従って操作してください。インストール時の設定項目を次に示します。

画面	説明	このマニュアルでの設定値
パスワードの設定	HiRDB Control Manager - Console から HiRDB Control Manager - Server に接続するときのパスワードを設定します。	hircms
ポート番号の設定	HiRDB Control Manager - Console から接続を受けるポート番号を設定します。内部的には、ここで設定した番号とその次の番号を使用します。連続して空いているポート番号の最初の番号を指定してください。	20291 (内部的には、20292 も使用)
最大接続数	HiRDB Control Manager - Server に同時に接続できる HiRDB Control Manager - Console の数を設定します。	2
HiRDB Control Manager - Agent の自動探索機能の設定	次の条件を満たす HiRDB Control Manager - Agent を HiRDB Control Manager - Server に自動登録するかどうかを指定します。 <ul style="list-style-type: none">• HiRDB Control Manager - Server と同一セグメント内• HiRDB Control Manager - Agent のインストール時に設定したポート番号が 20291• 接続可能な状態にある	自動探索を行わない
管理規模の選択	HiRDB Control Manager で管理する HiRDB の規模を指定します。	小規模

3. [セットアップの完了] 画面では、「はい、直ちにコンピュータを再起動します。」を選択してください。

(2) HiRDB Control Manager - Console のインストール方法 (Windows マシン)

1. CD-ROM を挿入して、「11_CMC¥DISK1」の「SETUP.EXE」をダブルクリックしてください。
インストーラが起動されます。
2. 以降は、画面の内容に従って操作してください。

画面	説明	このマニュアルでの設定値
HiRDB Control Manager - Server 情報の入力	HiRDB Control Manager - Console がデフォルトで接続する HiRDB Control Manager - Server の情報を指定します。	<ul style="list-style-type: none"> • IP アドレス • 10.210.35.24
ポート番号の設定	HiRDB Control Manager - Console がデフォルトで接続する HiRDB Control Manager - Server のポート番号を設定します。	20291
タイムアウトの設定	HiRDB Control Manager - Console からの 1 回の操作に対する待ち時間を設定します。	300

(3) HiRDB Control Manager - Agent のインストール方法 (HiRDB サーバマシン)

1. CD-ROM をマウントしてください (以降、マウント先を /cdrom とします)。
2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してください。
3. 「Install Software」から「HiRDB Control Manager - Agent」を選択して、インストールしてください。

2.2.6 HiRDB/Run Time をインストールする (Windows マシン)

HiRDB Control Manager がインストールされている Windows マシンに、HiRDB/Run Time をインストールします。HiRDB Control Manager を使用しない場合や、HiRDB Control Manager を使って Windows 版の HiRDB SQL Executer を起動しない場合は、この作業は必要ありません。

1. CD-ROM を挿入して、「09_PDSQL¥DISK1」の「SETUP.EXE」をダブルクリックしてください。
インストーラが起動されます。
2. 以降は、画面の内容に従って操作してください。

2.2.7 HiRDB SQL Executer をインストールする (Windows マシン)

HiRDB Control Manager がインストールされている Windows マシンに、HiRDB SQL Executer をインストールします。HiRDB Control Manager を使用しない場合や、HiRDB Control Manager を使って Windows 版の HiRDB SQL Executer を起動しない場合は、この作業は必要ありません。

1. CD-ROM を挿入して、「09_PDSQL¥DISK1」の「SETUP.EXE」をダブルクリックしてください。
インストーラが起動されます。
2. 以降は、画面の内容に従って操作してください。

2.2.8 HiRDB 運用ディレクトリを設定する

HiRDB を運用するためのディレクトリの権限を変更して、OS に登録します。このマニュアルでは、システムファイルがインストールされているパーティション (/opt/HiRDB_S) とは別のパーティションである、/HiRDB ディレクトリを HiRDB 運用ディレクトリとします。

HiRDB 運用ディレクトリの設定手順を次に示します。

(1) HiRDB 運用ディレクトリの権限の変更方法

HiRDB 運用ディレクトリの権限を HiRDB 管理者「hirman」に変更します。

1. chown コマンドを入力して、ディレクトリの所有者を「hirman」に変更してください。

【コマンド形式】

```
$ chown hirman /HiRDB
```

2. chgrp コマンドを実行して、ディレクトリのグループを「hirdb」に変更してください。

【コマンド形式】

```
$ chgrp hirdb /HiRDB
```

3. chmod コマンドを実行して、ディレクトリのモードを変更してください。

【コマンド形式】

```
$ chmod 0755 /HiRDB
```


(2) HiRDB 運用ディレクトリの OS への登録方法

pdsetup コマンドを入力して、HiRDB 運用ディレクトリを OS に登録します。

【コマンド形式】

```
$ /opt/HiRDB_S/bin/pdsetup /HiRDB
```

2.3 HiRDB の環境設定をする（コマンドでの実行）

HiRDB の環境設定をコマンドで実行する場合、次の手順で行います。

1. HiRDB ファイルシステム領域の作成
2. HiRDB システム定義の作成
3. システムファイルの作成
4. HiRDB の開始
5. RD エリアの作成
6. HiRDB の稼働状況の確認

なお、各項では基本的なことだけ説明しています。各項の詳細について知りたい場合は、次のマニュアルを参照してください。

内容	参照先マニュアル
HiRDB ファイルシステム領域の作成， HiRDB システム定義の作成， システムファイルの作成， RD エリアの作成	「HiRDB Version 7 システム導入・設計ガイド（UNIX(R) 用）」
HiRDB の開始， HiRDB の稼働状況の確認	「HiRDB Version 7 システム運用ガイド（UNIX(R) 用）」

環境設定を始める前に

hosts ファイルの設定：

/etc/hosts ファイルに、HiRDB/ シングルサーバをインストールしたマシンの IP アドレス、およびホスト名があるか確認してください。ない場合には、IP アドレス、およびホスト名を追加してください。ただし、DNS サーバを使ってホスト名を管理する場合は必要ありません。

（hosts ファイルの設定例）

```
：  
10.210.35.14 HiRLIN_SV.soft.hitachi.co.jp HiRLIN_SV  
10.210.35.15 HiRLIN_CL.soft.hitachi.co.jp HiRLIN_CL  
：
```

コマンド実行時に作成するファイルについて：

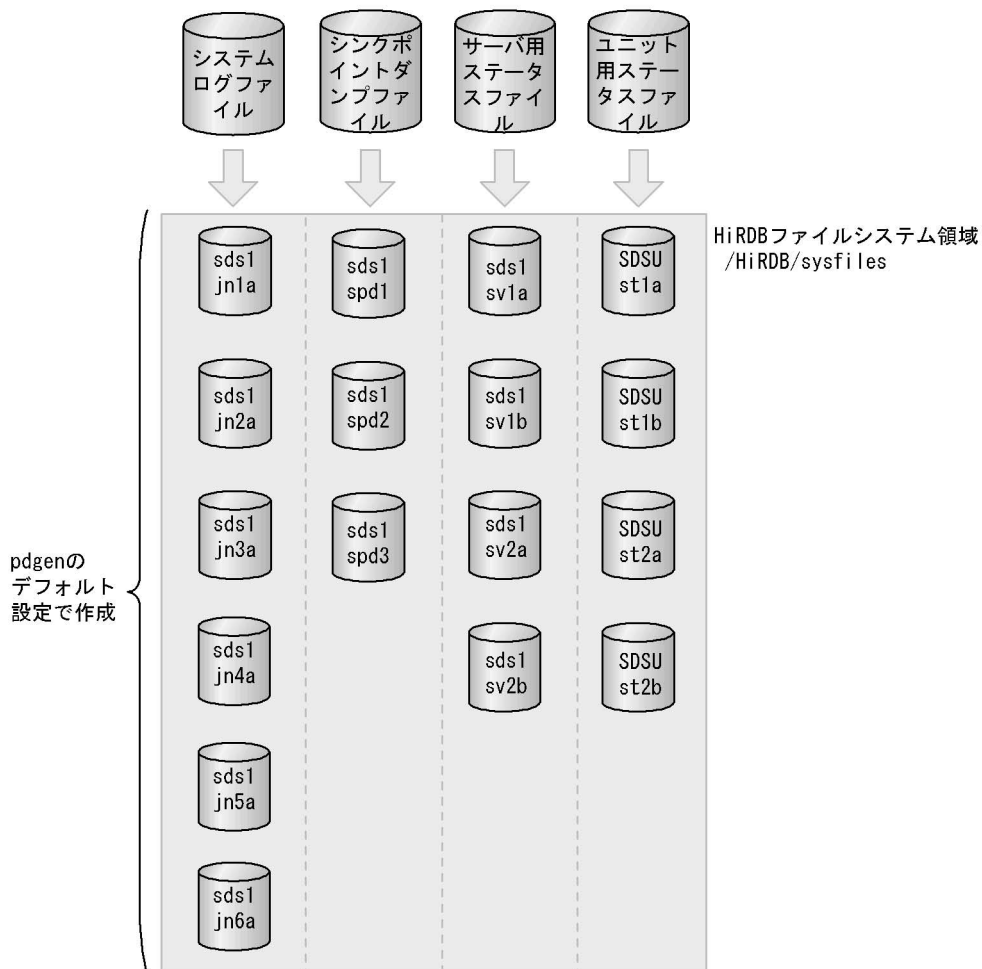
システム定義ファイルやユティリティの制御文ファイルなどは、システムジェネレータが呼び出しているファイルを流用すれば、作成時間（入力時間）を短縮できます。バッチファイルが呼び出しているファイルについては、「2.4 HiRDB の環境設定をする（システムジェネレータでの実行）」を参照してください。

2.3.1 HiRDB ファイルシステム領域を作成する

pdfmkfs コマンドで、HiRDB ファイルシステム領域を作成します。HiRDB ファイルシステム領域の構成を図 2-3 に示します。

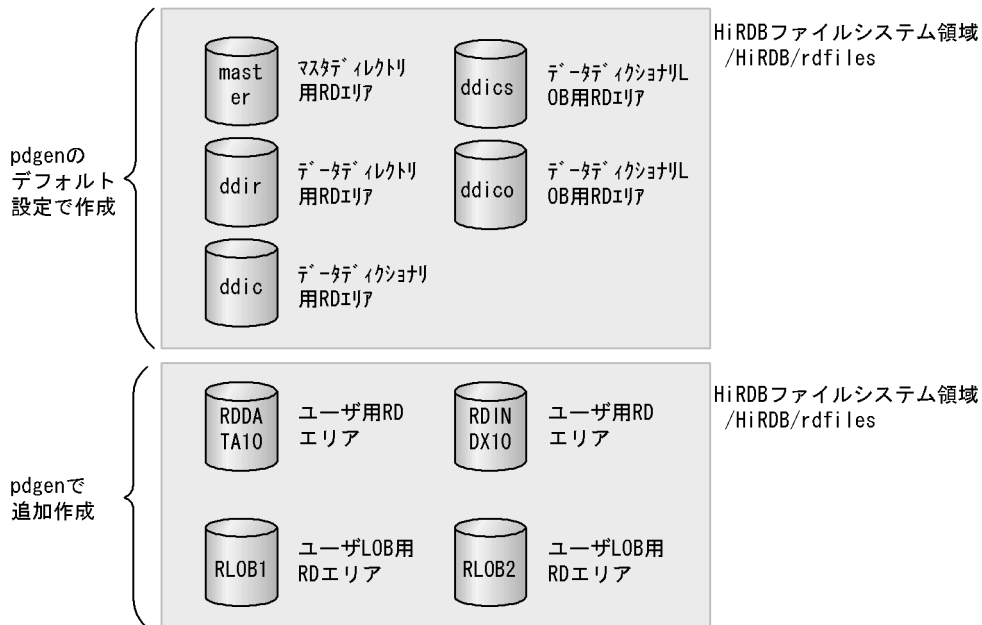
図 2-3 HiRDB ファイルシステム領域の構成

【システムファイル用のHiRDBファイルシステム領域】



2. HiRDB システムを構築してみましょう

【RDエリア用のHiRDBファイルシステム領域】



本番用システムでHiRDBファイルシステム領域を作成する場合

この学習用のHiRDBシステムでは、HiRDBファイルシステム領域は使用目的に関係なくすべて一つのディスクに作成していますが、本番用のシステムでは使用目的に合わせて別ディスクに作成することをお勧めします。

(1) 「area」の作成

コマンドを実行する前に、\$HOME/HiRDBディレクトリ下に「area」というディレクトリを作成します。

(2) pdfmkfs コマンドの実行

コマンドプロンプトから、次に示すコマンドを実行します。

次のコマンド形式では実行するコマンドをまとめて記述しています。実際に実行するときは、1コマンドずつ実行してください。

【コマンド形式】

```
$ pdfmkfs -r -n 119 -l 400 -k SVR -e 100 -i /HiRDB/sysfiles  
$ pdfmkfs -r -n 81 -l 100 -k SVR -e 100 -i /HiRDB/rdfiles
```

〔説明〕

-r :

確認メッセージを表示しないで初期設定するときのオプションです。

-n :

HiRDB ファイルシステム領域サイズ (単位 : MB) です。

-l :

HiRDB ファイルシステム領域に作成する HiRDB ファイルの数の上限です。

-k :

HiRDB ファイルシステム領域の使用目的の指定です。

SVR は、ユティリティ用以外のすべての目的に使用できます。

1 行目のコマンド (.../HiRDB/sysfiles) では、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、または監査証跡ファイル用の HiRDB ファイルシステム領域として使用します。2 行目のコマンド (.../HiRDB/rdfiles) では、RD エリア用の HiRDB ファイルシステム領域として使用します。

したがって、それぞれを SYS, DB と指定することもできます。

-e :

HiRDB ファイルシステム領域の HiRDB ファイルの増分回数の上限です。

-i

-n オプションで指定した HiRDB ファイルシステム領域サイズ分の領域を最初から初期化する場合に指定するときのオプションです。

/HiRDB/... :

HiRDB ファイルシステム領域の名称です。

(3) コマンドの実行結果の確認

pdfmkfs コマンドが正しく実行されたかどうかを確認するために、pdfstatfs コマンドを実行します。なお、ここでは HiRDB ファイルシステム領域 /HiRDB/sysfiles についてだけ説明します。その他の HiRDB ファイルシステム領域についても同様に確認してみてください。

【コマンド形式】

```
$ pdfstatfs /HiRDB/sysfiles
```

【コマンドの実行結果】

2. HiRDB システムを構築してみましょう

```
user area capacity          121642 [KB]
remain user area capacity  2390 [KB]
available file size        2390 [KB]
available file count       400
current file count         17
remain file count          383
free area count            0
available expand count     100
current expand count       0
initialize area kind       SVR
initialize user id         hirman
initialize time            Fri Jan 30 13:50:08 2004
```

〔説明〕

サイズが 119MB の、システムファイル用の HiRDB ファイルシステム領域が作成されています。また、作成者は hirman です。

補足事項

HiRDB ファイルシステム領域は、通常ファイルとしても確認できます。/HiRDB ディレクトリ下に、作成した各 HiRDB ファイルシステム領域の名称のファイルがあります。

2.3.2 HiRDB システム定義を作成する

HiRDB システム定義を作成します。HiRDB システム定義は、vi などのテキストエディタを使用してテキストファイルで作成してください。

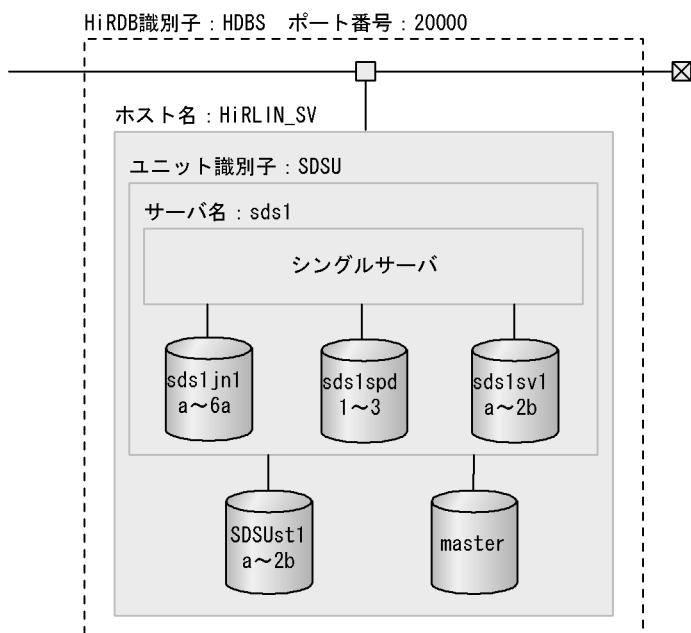
各定義のファイル名称は、システム共通定義は pdsys、ユニット制御情報定義は pdutsys、シングルサーバ定義は sds1 という名称で作成します。

HiRDB システム定義を作成するときの注意事項

1. HiRDB システム定義の各行には、必ず改行を入れてください。
2. 作成した pdsys、pdutsys、および sds1 は、/HiRDB/conf ディレクトリに格納してください。
3. ファイル名には拡張子 (.txt など) は付けしないでください。

HiRDB/シングルサーバの構成を図 2-4 に示します。なお、図中の指定値は pdsys、pdutsys、および sds1 の内容と対応しています。

図 2-4 HiRDB/ シングルサーバの構成



- (凡例) sds1jn1a~6a : システムログファイル
 sds1spd1~3 : シンクポイントダンプファイル
 sds1sv1a~2b : サーバ用ステータスファイル
 SDSUst1a~2b : ユニット用ステータスファイル
 master : マスタディレクトリ用RDエリアの先頭のHiRDBファイル

[説明]

- HDBS : システム共通定義pd_system_idオペランドでの指定値
 20000 : システム共通定義pd_name_portオペランドでの指定値
 HiRLIN_SV : システム共通定義pdstartオペランドの-xでの指定値
 (HiRDB/シングルサーバをインストールしたマシンのホスト名)
 を指定します。
 SDSU : ユニット制御情報定義のpd_unit_idオペランドでの指定値
 sds1 : システム共通定義pdstartオペランドの-sでの指定値、およびシングルサーバ定義のファイル名

注 グローバルバッファプールについては、図2-5を参照してください。

(1) システム共通定義 (pdsys)

システム共通定義 (pdsys) の内容を次に示します。

2. HIRDB システムを構築してみましょう

```
set pd_system_id = HDBS
set pd_name_port = 20000
set pd_mode_conf = AUTO
set pd_max_users = 5
set pd_watch_time = 3600
set pd_rpc_trace = N
set pd_rpc_trace_size = 1000000
set pd_rpc_trace_name = "/tmp"
set pd_master_file_name ¥
    = /HiRDB/rdfiles/master
set pd_sql_object_cache_size = 2000
set pd_combuf_hardware = N
set pd_ha = nouse
set pd_ha_ipaddr_inherit = N
set pd_host_watch_interval = 10
set pd_jpl_use = N
set pd_lck_deadlock_info = Y
set pd_max_recover_process = 1
set pd_mlg_file_size = 1024
set pd_overflow_suppress = N
set pd_shmpool_attribute = fixed

set pd_statistics = N
set pd_utl_buff_size = 32
set pd_optimize_level = "PRIOR_NEST_JOIN", "PRIOR_OR_INDEXES", ¥
    "DETER_AND_INDEXES", "RAPID_GROUPING", ¥
    "DETER_WORK_TABLE_FOR_UPDATE", ¥
    "APPLY_ENHANCED_KEY_COND"
set pd_additional_optimize_level = "COST_BASE_2"
set pd_large_file_use = Y
set pd_lck_release_detect = pipe
set pd_stj_buff_size = 128
pdbuffer -a bp01 -o -n 350

pdunit -x HIRLIN_SV -u SDSU ¥
    -d /HiRDB
pdstart -t SDS -s sds1 -x HIRLIN_SV
pdbuffer -a RDINDX10 -r RDINDX10 -n 700
pdbuffer -a RDDATA10 -r RDDATA10 -n 700
pdbuffer -a RLOB1 -b RLOB1 -n 200
pdbuffer -a RLOB2 -b RLOB2 -n 500
```

(2) ユニット制御情報定義 (pdutsys)

ユニット制御情報定義 (pdutsys) の内容を次に示します。

```
set pd_hostname = HIRLIN_SV
set pd_unit_id = SDSU
set pd_syssts_file_name_1 = "syssts01", ¥
    "/HiRDB/sysfiles/SDSUst1a", ¥
    "/HiRDB/sysfiles/SDSUst1b"
set pd_syssts_file_name_2 = "syssts02", ¥
    "/HiRDB/sysfiles/SDSUst2a", ¥
    "/HiRDB/sysfiles/SDSUst2b"
set pd_syssts_initial_error = stop
set pd_syssts_singleoperation = stop
```


(3) シングルサーバ定義 (sds1)

シングルサーバ定義 (sds1) の内容を次に示します。

シングルサーバ定義には、システムログファイルの運用に関する設定があります。デフォルト値では、「アンロード状態をチェックする運用」になっています (pd_log_unload_check=Y)。システムログファイルは、その状態を監視しておく必要があります。運用を誤ると HiRDB が動作しなくなることもあります。ただし、自動ログアンロード機能を使った運用を行うと、システムログファイルの監視の煩わしさを軽減できます。自動ログアンロード機能を使って運用したい場合は、シングルサーバ定義に、次の設定を追加してください。

- pd_log_auto_unload_path：アンロードログの出力先を指定します。

システムログの運用については「4.3 システムログを操作する」を参照してください。

```

set pd_table_def_cache_size      = 1000
set pd_view_def_cache_size       = 0
set pd_auth_cache_size           = 1
set pd_alias_cache_size          = 0
set pd_routine_def_cache_size    = 100
set pd_log_dual                   = N
set pd_log_sdinterval            = 1000,60
set pd_log_rec_leng               = 1024
set pd_log_rerun_reserved_file_open = Y
set pd_log_max_data_size         = 32000
set pd_log_rerun_swap             = N
set pd_log_singleoperation        = N
set pd_log_write_buff_count      = 3
set pd_log_unload_check          = Y

set pd_lck_until_disconnect_cnt  = 256
set pd_process_count             = 3
set pd_spd_assurance_count       = 1
set pd_spd_max_data_size         = 32000
set pd_sts_initial_error         = stop
set pd_sts_singleoperation       = stop

set pd_work_buff_size            = 1024
set pd_log_remain_space_check    = safe
set pd_spd_dual                  = N
set pd_sts_file_name_1 = "sts01",¥
                        "/HiRDB/sysfiles/sds1sv1a",¥
                        "/HiRDB/sysfiles/sds1sv1b"
set pd_sts_file_name_2 = "sts02",¥
                        "/HiRDB/sysfiles/sds1sv2a",¥
                        "/HiRDB/sysfiles/sds1sv2b"

```

2. HiRDB システムを構築してみましょう

```
pdlogadfg -d spd -g spdfg01 ONL
pdlogadfg -d spd -g spdfg02 ONL
pdlogadfg -d spd -g spdfg03 ONL
pdlogadpf -d spd -g spdfg01 ¥
-a /HiRDB/sysfiles/sds1spd1
pdlogadpf -d spd -g spdfg02 ¥
-a /HiRDB/sysfiles/sds1spd2
pdlogadpf -d spd -g spdfg03 ¥
-a /HiRDB/sysfiles/sds1spd3

pdlogadfg -d sys -g logfg01 ONL
pdlogadfg -d sys -g logfg02 ONL
pdlogadfg -d sys -g logfg03 ONL
pdlogadfg -d sys -g logfg04 ONL
pdlogadfg -d sys -g logfg05 ONL
pdlogadfg -d sys -g logfg06 ONL
pdlogadpf -d sys -g logfg01 ¥
-a /HiRDB/sysfiles/sds1jn1a
pdlogadpf -d sys -g logfg02 ¥
-a /HiRDB/sysfiles/sds1jn2a
pdlogadpf -d sys -g logfg03 ¥
-a /HiRDB/sysfiles/sds1jn3a
pdlogadpf -d sys -g logfg04 ¥
-a /HiRDB/sysfiles/sds1jn4a
pdlogadpf -d sys -g logfg05 ¥
-a /HiRDB/sysfiles/sds1jn5a
pdlogadpf -d sys -g logfg06 ¥
-a /HiRDB/sysfiles/sds1jn6a
```

2.3.3 システムファイルを作成する

システムファイルとは次のファイルのことをいいます。

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

ここで作成するシステムファイルは、HiRDB システム定義で指定したシステムファイルと対応しています。システムファイルの構成については図 2-3 を参照してください。

(1) システムログファイルの作成

システムログファイルは、pdloginit -d sys コマンドで作成します。

(a) pdloginit -d sys コマンドの実行

次のコマンド形式は実行するコマンドをまとめて記述しています。実際に実行するときは、1 コマンドずつ実行してください。

【コマンド形式】

```

$ pdloginit -d sys -s sds1 -f /HiRDB/sysfiles/sds1jn1a -n 17408 -l 1024
$ pdloginit -d sys -s sds1 -f /HiRDB/sysfiles/sds1jn2a -n 17408 -l 1024
$ pdloginit -d sys -s sds1 -f /HiRDB/sysfiles/sds1jn3a -n 17408 -l 1024
$ pdloginit -d sys -s sds1 -f /HiRDB/sysfiles/sds1jn4a -n 17408 -l 1024
$ pdloginit -d sys -s sds1 -f /HiRDB/sysfiles/sds1jn5a -n 17408 -l 1024
$ pdloginit -d sys -s sds1 -f /HiRDB/sysfiles/sds1jn6a -n 17408 -l 1024

```

〔説明〕

-d sys :

システムログファイルを作成する指定です。

-s sds1 :

シングルサーバ sds1 にシステムログファイルを作成します。

-f /HiRDB/sysfiles/... :

システムログファイルの物理ファイルの指定です。pdfmkfs コマンドで作成した HiRDB ファイルシステム領域にファイルを作成します。

-n 17408 :

システムログファイルとして初期設定する HiRDB ファイルのレコード数を指定します。なお、1 レコード当たりのレコード長は、-l オプションで指定します。

-l 1024

システムログファイルとして初期設定する HiRDB ファイルのレコード長を指定します。

(b) コマンドの実行結果の確認

pdloginit -d sys コマンドが正しく実行されたかどうかを確認するために、pdfls コマンドを実行します。なお、ここではシステムログファイル「/HiRDB/sysfiles/sds1jn1a」についてだけ説明します。その他のシステムログファイルについても同様に確認してみてください。

【コマンド形式】

```

$ pdfls /HiRDB/sysfiles/sds1jn1a

```

【コマンドの実行結果】

```

rwr-r- hirman users          4096      17408 18:10 Feb 03 2004 sds1jn1a

```

〔説明〕

17408 レコードのシステムログファイル (sds1jn1a) が作成されています。また、

2. HiRDB システムを構築してみましょう

作成者は hirman です。

(2) シンクポイントダンプファイルの作成

シンクポイントダンプファイルは、pdloginit -d spd コマンドで作成します。

(a) pdloginit -d spd コマンドの実行

次のコマンド形式は実行するコマンドをまとめて記述しています。実際に実行するときは、1 コマンドずつ実行してください。

【コマンド形式】

```
$ pdloginit -d spd -s sds1 -f /HiRDB/sysfiles/sds1spd1 -n 31
$ pdloginit -d spd -s sds1 -f /HiRDB/sysfiles/sds1spd2 -n 31
$ pdloginit -d spd -s sds1 -f /HiRDB/sysfiles/sds1spd3 -n 31
```

〔説明〕

-d spd :

シンクポイントダンプファイルを作成する指定です。

-s sds1 :

シングルサーバ sds1 にシンクポイントダンプファイルを作成します。

-f /HiRDB/sysfiles/... :

シンクポイントダンプファイルの物理ファイルの指定です。pdfmkfs コマンドで作成した HiRDB ファイルシステム領域にファイルを作成します。

-n 31 :

システムログファイルとして初期設定する HiRDB ファイルのレコード数を指定します。なお、1レコード当たりのレコード長は、4096 バイトです。

(b) コマンドの実行結果の確認

pdloginit -d spd コマンドが正しく実行されたかどうかを確認するために、pdfls コマンドを実行します。なお、ここではシンクポイントダンプファイル「/HiRDB/sysfiles/sds1spd1」についてだけ説明します。その他のシンクポイントダンプファイルについても同様に確認してみてください。

【コマンド形式】

```
$ pdfls /HiRDB/sysfiles/sds1spd1
```

【コマンドの実行結果】

```
rwr-r- hirman users          4096      31 18:10 Feb 03 2004 sds1jnl1
```

〔説明〕

31 レコードのシンクポイントダンプファイル (sds1jn1a) が作成されています。また、作成者は hirman です。

(3) ステータスファイルの作成

ステータスファイルは、ユニット用とサーバ用に 2 種類作成する必要があります。

(a) ユニット用ステータスファイル

ユニット用ステータスファイルは、pdstsinit コマンドで作成します。

pdstsinit コマンドの実行

コマンド形式は実行するコマンドをまとめて記述しています。実際に実行するときには、1 コマンドずつ実行してください。

【コマンド形式】

```
$ pdstsinit -f /HiRDB/sysfiles/SDSUst1a -c 32
$ pdstsinit -f /HiRDB/sysfiles/SDSUst1b -c 32
$ pdstsinit -f /HiRDB/sysfiles/SDSUst2a -c 32
$ pdstsinit -f /HiRDB/sysfiles/SDSUst2b -c 32
```

〔説明〕

-f /HiRDB/sysfiles/... :

初期設定をする物理ファイルの名称を絶対パス名で指定します。物理ファイル名は、HiRDB ファイルシステム領域名 /HiRDB ファイル名で指定してください。

また、HiRDB ファイル名に pl で始まる名称は指定できません。すでに初期設定している物理ファイル名を指定した場合は、再度初期設定されます。

-c 32 :

ステータスファイルのレコード数を指定します。

コマンドの実行結果の確認

pdstsinit コマンドが正しく実行されたかどうかを確認するために、pdfls コマンドを実行します。なお、ここではユニット用ステータスファイル「/HiRDB/sysfiles/SDSUst1a」についてだけ説明します。その他のユニット用ステータスファイルについても同様に確認してみてください。

【コマンド形式】

```
$ pdfls /HiRDB/sysfiles/SDSUst1a
```

【コマンドの実行結果】

2. HiRDB システムを構築してみましょう

```
rwr-r- hirman users          4096          32 18:12 Feb 03 2004 SDSUst1a
```

〔説明〕

32レコードのユニット用ステータスファイル (SDSUst1a) が作成されています。
また、作成者は hirman です。

(b) サーバ用ステータスファイル

サーバ用ステータスファイルは、pdstsinic コマンドで作成します。

pdstsinic コマンドの実行

コマンド形式は実行するコマンドをまとめて記述しています。実際に実行するときには、1コマンドずつ実行してください。

【コマンド形式】

```
$ pdstsinic -s sds1 -f /HiRDB/sysfiles/sds1sv1a -c 870
$ pdstsinic -s sds1 -f /HiRDB/sysfiles/sds1sv1b -c 870
$ pdstsinic -s sds1 -f /HiRDB/sysfiles/sds1sv2a -c 870
$ pdstsinic -s sds1 -f /HiRDB/sysfiles/sds1sv2b -c 870
```

〔説明〕

-s sds1 :

サーバ用ステータスファイルに対応するサーバの名称 (sds1) を指定します。

-f /HiRDB/sysfiles/... :

初期設定をする物理ファイルの名称を絶対パス名で指定します。物理ファイル名は、「HiRDB ファイルシステム領域名/HiRDB ファイル名」の形式で指定してください。

また、HiRDB ファイル名に pl で始まる名称は指定できません。すでに初期設定している物理ファイル名を指定した場合は、再度初期設定されます。

-c 870 :

ステータスファイルのレコード数を指定します。

コマンドの実行結果の確認

pdstsinic コマンドが正しく実行されたかどうかを確認するために、pdfls コマンドを実行します。なお、ここではサーバ用ステータスファイル「/HiRDB/sysfiles/sds1sv1a」についてだけ説明します。その他のサーバ用ステータスファイルについても同様に確認してみてください。

【コマンド形式】

```
$ pdfls /HiRDB/sysfiles/sds1sv1a
```

【コマンドの実行結果】

```
rwr-r- hirman users      4096      870 18:12 Feb 03 2004 sds1sv1a
```

〔説明〕

870レコードのサーバ用ステータスファイル (sds1sv1a) が作成されています。また、作成者は hirman です。

2.3.4 HiRDB を開始する

HiRDB を開始します。HiRDB を開始する場合、pdstart コマンドを実行してください。pdstart コマンドを実行すると、HiRDB が初期設定終了待ち状態になり、次のメッセージが表示されます。

```
KFPS05201-Q Dictionary not initialized; enter pdinit command
```

これは、データベース初期設定ユーティリティ (pdinit) を実行してディクショナリを初期設定してくださいというメッセージなので、pdinit を実行します。pdinit については、「2.3.5 RD エリアを作成する」を参照してください。pdinit が正常に終了すると、HiRDB が開始されます。

なお、通常の運用時の開始方法、および終了方法については、「4.2 HiRDB を開始、終了する」、または「5.4 HiRDB を開始、終了する」を参照してください。

注

HiRDB Version 5.0 からバージョンアップした場合は、pdstart を実行する前に、サービスで HiRDB/ シングルサーバを開始する必要があります。

[コントロールパネル] - [管理ツール] - [サービス] のリストボックスから「HiRDB/Single Server」を選択し、[操作] - [開始] を選択するか、[サービスの開始] ボタンをクリックする必要があります。

2.3.5 RD エリアを作成する

データベース初期設定ユーティリティ (pdinit) で、RD エリアを作成します。

作成する RD エリアを次に示します。

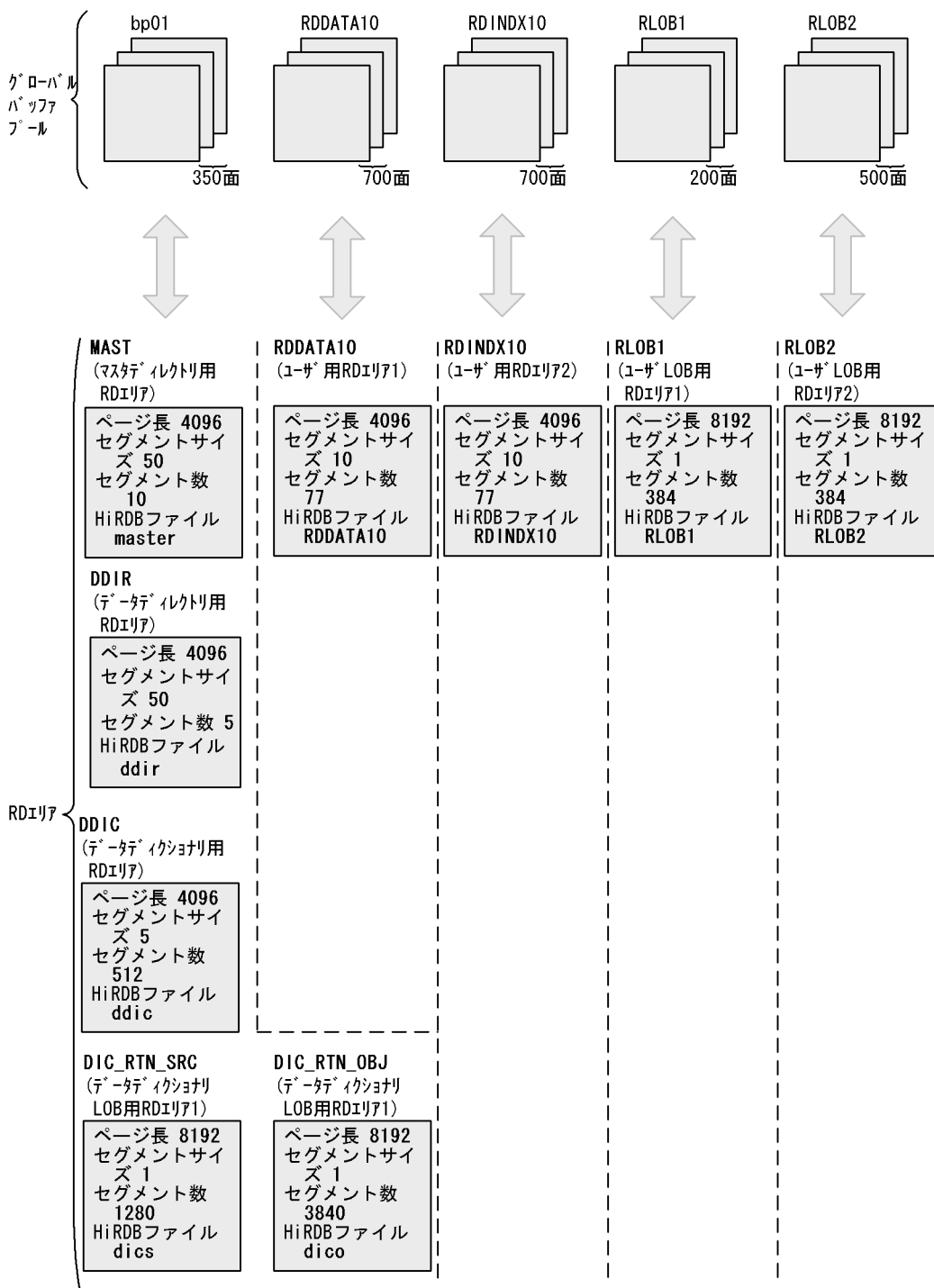
- マスタディレクトリ用 RD エリア (MAST)

2. HiRDB システムを構築してみましょう

- データディレクトリ用 RD エリア (DDIR)
- データディクショナリ LOB 用 RD エリア (DIC_RTN_SRC, および DIC_RTN_OBJ)
- ユーザ LOB 用 RD エリア (RLOB1, および RLOB2)
- データディクショナリ用 RD エリア (RDDICT)
- ユーザ用 RD エリア (RDDATA10, および RDINDX10)

これらの RD エリアは、HiRDB システム定義で定義したグローバルバッファプールと図 2-5 のように対応します。

図 2-5 RD エリアの構成



データベース初期設定ユーティリティで RD エリアを作成する方法について説明します。

2. HiRDB システムを構築してみましょう

(1) 「\$HOME/HiRDB/pdinit/initfile.txt」の作成

まず、制御文ファイル「\$HOME/HiRDB/pdinit/initfile.txt」を作成します。なお、
・・・n (n は 1 ~ 5) は説明上の文字列ですので、実際には入力しないでください。

【C:\¥hirdb¥pdinit¥initfile.txt の内容】

```
create rdarea MAST for masterdirectory .....1
page 4096 characters .....2
storage control segment 50 pages .....3
file name "/HiRDB/rdfiles/master" .....4
initial 10 segments ; .....5

create rdarea DDIR for datadirectory
page 4096 characters
storage control segment 50 pages
file name "/HiRDB/rdfiles/ddir"
initial 5 segments ;

create rdarea DDIC for datadictionary
page 4096 characters
storage control segment 5 pages
file name "/HiRDB/rdfiles/ddic"
initial 512 segments ;

create rdarea DIC_RTN_SRC for LOB used by HiRDB(SQL_ROUTINES)
page 8192 characters
storage control segment 1 pages
file name "/HiRDB/rdfiles/ddics"
initial 1280 segments ;

create rdarea DIC_RTN_OBJ for LOB used by HiRDB(SQL_ROUTINES)
page 8192 characters
storage control segment 1 pages
file name "/HiRDB/rdfiles/ddico"
initial 3840 segments ;

create rdarea RDDATA10 for user used by PUBLIC
page 4096 characters
storage control segment 10 pages
file name "/HiRDB/rdfiles/RDDATA10"
initial 77 segments ;

create rdarea RDINDX10 for user used by PUBLIC
page 4096 characters
storage control segment 10 pages
file name "/HiRDB/rdfiles/RDINDX10"
initial 77 segments ;

create rdarea RLOB1 for LOB used by PUBLIC
page 8192 characters
storage control segment 1 pages
file name "/HiRDB/rdfiles/RLOB1"
initial 384 segments ;

create rdarea RLOB2 for LOB used by PUBLIC
page 8192 characters
storage control segment 1 pages
file name "/HiRDB/rdfiles/RLOB2"
initial 384 segments ;
```

〔説明〕

1. RD エリア名の指定です。ユーザ用 RD エリアの場合は、さらに公用 RD エリア (PUBLIC) の指定もします。
2. RD エリアを構成する HiRDB ファイルのページ長の指定です (単位はバイト)。
3. RD エリアを構成する HiRDB ファイルのセグメントサイズの指定です (単位はページ数)。
4. RD エリアに割り当てる、HiRDB ファイルシステム領域の HiRDB ファイル名の指定です。ここで、指定する HiRDB ファイルシステム領域は pdfmkfs コマンドですすでに作成しています。
5. RD エリアを構成する HiRDB ファイルのセグメント数の指定です。

注

create rdarea 文のオペランド extension を指定すると、RD エリアの自動増分を設定することができます。自動増分とは、RD エリアの容量が不足したとき自動的に RD エリアを拡張する機能です。詳細については、マニュアル「HiRDB Version 7 コマンドリファレンス (UNIX(R) 用)」の pdinit の個所を参照してください。

(2) pdinit コマンドの実行

作成した制御ファイルを入力にして、データベース初期設定ユーティリティ (pdinit) を実行します。

【コマンド形式】

```
$ pdinit -d $HOME/HiRDB/pdinit/initfile.txt
```

〔説明〕

-d \$HOME/HiRDB/pdinit/initfile.txt :
データベース初期設定ユーティリティの制御文ファイルの指定です。

(3) コマンドの実行結果の確認

データベース初期設定ユーティリティが正常に終了すると、次のメッセージが表示されず。

```
KFPX24000-I DB initialize ended, return code=0
```

実際に RD エリアが作成されたかどうかを確認するために、pddbls -r ALL コマンドを実行します。

【コマンド形式】

2. HiRDB システムを構築してみましょう

```
$ pddbls -r ALL
```

【コマンドの実行結果】

```
STATE OF RDAREA
RDAREA      ID      STATUS      TYPE
RDAREA      1      OPNMODE
MAST        1      OPEN        MAST
           INITIAL
DDIR        2      OPEN        DDIR
           INITIAL
DDIC        3      OPEN        DDIC
           INITIAL
DIC_RTN_SRC 4      OPEN        DLOB
           INITIAL
DIC_RTN_OBJ 5      OPEN        DLOB
           INITIAL
RDINDX10    6      OPEN        USER
           INITIAL
RDDATA10    7      OPEN        USER
           INITIAL
RLOB1       8      OPEN        ULOB
           INITIAL
RLOB2       9      OPEN        ULOB
           INITIAL
RDAREA01    10     OPEN        USER
           INITIAL
RDAREA02    11     OPEN        USER
           INITIAL
```

【説明】

制御文ファイル「\$HOME/HiRDB/pdinit/initfile.txt」で指定した 11 個の RD エリアが作成されています。

(4) 補足事項

pdinit を実行すると、実行ユーザに対して DBA 権限が与えられます。ただし、pdinit 実行者に root 権限がある場合は、ユーザ名が「root」になります。実行ユーザは必要に応じて、各ユーザにアクセス権限を与える (SQL の GRANT) 必要があります。

2.3.6 HiRDB の稼働状況を確認する

「2.3.5 RD エリアを作成する」まで正常に終了すると、HiRDB が稼働しているはずですが、実際に HiRDB が稼働しているかどうかを、pdls コマンドで確認してみます。

【コマンド形式】

```
$ pdls
```

【コマンドの実行結果】

HOSTNAME (192039)	UNITID	SVID	STATUS	STARTTIME
HIRLIN_SV	SDSU	*****	ACTIVE	174639
HIRLIN_SV	SDSU	sds1	ACTIVE	174639

〔説明〕

STATUS が ACTIVE になっていれば、HiRDB が稼働中です。

2.4 HiRDB の環境設定をする (システムジェネレータでの実行)

インストール終了後は、HiRDB の環境設定をします。環境設定には、HiRDB の実行に必要な環境変数の設定と HiRDB システムの構築があります。これらの作業は、HiRDB 管理者「hirman」が行います。

HiRDB システムの構築には、システムジェネレータを使用します。システムジェネレータ「pdgen」を使用すると、次の作業を対話形式で行えます。

HiRDB ファイルシステム領域の作成

HiRDB システム定義の作成

システムファイルの作成

HiRDB の開始

RD エリアの作成

「pdgen」を実行して作成される、HiRDB ファイルシステム領域の構成、HiRDB/ シングルスルサーバの構成、および RD エリアの構成については、それぞれ図 2-3、図 2-4、図 2-5 を参照してください。

2.4.1 HiRDB の環境変数を設定する

HiRDB 管理者「hirman」の環境に環境変数を設定します。設定するファイルは、シェルによって異なりますが、ここでは Bourne シェルの場合の設定方法を示します。

1. \$HOME/.profile に次の記述を追加してください。

```
PDDIR=/HiRDB
PDCONFPATH=$PDDIR/conf
PATH=$PDDIR/bin:$PDDIR/client/utl:$PATH
SHLIB_PATH=$PDDIR/lib
PDHOST=HiRLIN_SV
PDUSER="hirman"/"hirman"
PDNAMEPORT=20000
export PDDIR PDCONFPATH PATH SHLIB_PATH PDHOST PDUSER PDNAMEPORT
```

〔説明〕

PDDIR

HiRDB 運用ディレクトリの絶対パスです。

PDCONFPATH

HiRDB システム定義を格納するディレクトリです。

PATH

HiRDB のコマンド、ユティリティ、および HiRDB/ クライアントのライブラリを格納するディレクトリです。

SHLIB_PATH

HiRDB のプログラムなどが格納されているディレクトリです。Solaris 版や Linux 版の HiRDB の場合は、環境変数 LD_LIBRARY_PATH になります。また、AIX 5L 版の場合は、LIBPATH になります。

PDHOST

サーバマシンのホスト名です。

PDUSER

スキーマの所有者です。今のところ、HiRDB 管理者を設定しておきます。

PDNAMEPORT

サーバマシンのポート番号です。

2. 次のコマンドを実行してください。
環境変数が更新されます。

```
$ . $HOME/.profile
```

2.4.2 システムジェネレータ (pdgen) を実行する

「pdgen」の使用方法を次に示します。

(1) システムジェネレータを起動する

HiRDB 運用ディレクトリに移動して、コマンドラインから「bin/pdgen」を実行してください。

【コマンド形式】

```
$ bin/pdgen
```

(2) ライセンス形態を入力する

セットアップ中の HiRDB のライセンス形態を入力してください。このマニュアルでは、接続ユーザ数ライセンスと仮定して、「1」を入力します。

【入力】

2. HiRDB システムを構築してみましょう

```
-- HiRDB/Single Server License selection menu --
1 : Connection License
2 : CPU License
q : Quit system generation
Select 1/2 or q >1
```

(3) 接続ユーザ数を入力する

契約した接続ユーザ数を入力してください。このマニュアルでは、「5」を入力します。

【入力】

```
Enter number of your licenses (2- [licenses]) >5
```

(4) 項目を設定する

セットアップ項目を設定してください。設定したい項目の項番を入力すると、各設定値を聞かれるので、そこに値を入力してください。

本番用システムで HiRDB ファイルシステム領域を作成する場合

この学習用の HiRDB システムでは、HiRDB ファイルシステム領域は使用目的に関係なくすべて一つのディスクに作成しますが、本番用のシステムでは使用目的に合わせて別ディスクに作成することをお勧めします。

【設定画面】

```
-- Master menu --
1 : HiRDB directory                : /HiRDB
2 : HiRDB port number / system id   : 20000 /HDBS
3 : Number of concurrently execute user : 5[user]
4 : Number of access tables per user  : 16 [tables]
5 : Database update size per day     : 100 [MB]
- System file information (status/syncpoint dump/log) -
6 : File type (UNIX/Raw)            : UNIX
7 : Log options
8 : Path name of HiRDB file system   1 : /HiRDB/sysfiles (68 [MB])
9 : Path name of HiRDB file system   2 : Not assign
10 : Path name of HiRDB file system  3 : Not assign
- RDAREA information -
11 : File type (UNIX/Raw)            : UNIX
12 : Path name of HiRDB file system   : /HiRDB/rdfiles (71(Sys:61/User:10)
[MB])
o : Detail options
g : Start system generation
r : Add rdarea (Select this after system generated)
p : Setup Plugin (Select this after system generated)
t : Stop HiRDB
csh or sh : Enter Shell mode
q : Quit system make
Enter 1-12 or o-q >
```


このマニュアルでは、次の項目を設定します。それぞれの項目を設定したあと、< G > キーを押下してください。システムの構築が始まります。

設定する項目	変更前	変更後
5	・ 100	・ 10
12	・ /HiRDB/rdfiles ・ 71	・ /HiRDB/rdfiles ・ 81

【入力】

```
Enter 1-12 or o-q >g
```

【実行結果】

```
Start HiRDB system generation
Creating HiRDB definition files (/HiRDB/conf)
Definition files are created
:
KFPX24000-I DB initialize ended, return code=0
Rdarea initialize completed
4345 12:18:05 SDSU _rdm      KFPS05210-I HiRDB system initialization process
complete
Push enter key to main menu >
```

注

これまでに説明した準備で実行していない箇所があったり、設定に誤りがあったりした場合、システムの構築に失敗することがあります。

その場合は、エラーメッセージが出力されます。メッセージの内容に従って対処して、pdgen を再実行してください。

(5) ユーザ用 RD エリアを作成する - その 1

ユーザ用 RD エリアを作成します。次に示す内容の RD エリアを作成してください。

2. HIRDB システムを構築してみましょう

RD エリア名	エリアサイズ (MB)	セグメント サイズ (ページ)	ページサイズ (KB)	RD エリア タイプ	グローバルバッファサ イズ (ページ)
RDDATA10	3	10	4	Table	700
RDINDX10	3	10	4	Table	700
RLOB1	3	1	8	Blob	200
RLOB2	3	1	8	Blob	500

注

設定時には「for Binary large object data.」と表示されます。

RDDATA10 の設定例を示します。システムジェネレータ起動時の画面で < R > キーを押下して、次のように設定してください。

【設定例】

```
-- Create new rdarea menu --
1 : Rdarea name           : RDDATA10
2 : Area size              : 3 [MB]
3 : Segment size          : 10 [pages]
4 : Page size              : 4 [kB]
5 : Rdarea type (Table/Blob/List) : Table
6 : Global buffer size    : 700 [buffer pages]
7 : HiRDB-file-system-area name : /HiRDB/rdfiles
a : Add Rdarea to system
l : List of rdarea already exist
q : return to main menu
Enter 1-7 or a-q >
```

(6) ユーザ用 RD エリアを作成する - その2

作成する RD エリアを設定したあと、< A > キーを押下してください。エラーメッセージが出力された場合は、メッセージの内容に従って前の設定をやり直してください。

【入力】

```
Enter 1-7 or a-q >a
```

【実行結果】

```
Rdarea RDDATA10 creating
+ pdmod -a /HiRDB/conf/RDDATA10
KFPX14250-I Processing of create rdarea statement ended return code=0,
RDAREA name="RDDATA10"KFPX24213-I DB modification for
ended return code=0 at 13:43:26 on 2004/01/30
KFPX24200-I DB modification ended, return code=0
Enter to return menu >
```

(7) ユーザ用 RD エリアを作成する - その3

ユーザ用 RD エリア作成画面に戻り、< Q > キーを押下してください。

【入力】

```
Enter 1-7 or a-q >q
```

【実行結果】

```
Rdarea are added. You have to restart HiRDB to use the area.
Restart now ? (Y/N) >
```

(8) HiRDB を再起動する

新しい RD エリアを使用するために HiRDB を再起動するかどうかを尋ねるメッセージが出力されます。

作成する RD エリアが残っている場合

- < N > キーを押下してください。すべてのユーザ用 RD エリアを作成するまで、(5) ~ (7) の作業を繰り返します。

すべてのユーザ用 RD エリアを作成した場合

- < Y > キーを押下してください。

【入力】

```
Restart now ? (Y/N) >y
```

2. HiRDB システムを構築してみましょう

【実行結果】

```
HiRDB terminating
5068 12:28:39 SDSU _rdm      KFPS01841-I HiRDB unit SDSU terminated.
mode = NORMAL
5068 12:28:39 SDSU _rdm      KFPS01850-I HiRDB system terminated.
mode = NORMAL
5489 12:29:24 SDSU _rdm      KFPS05210-I HiRDB system initialization
process complete
HiRDB restart completed . Enter >
```

(9) システムジェネレータを終了する

< Q > キーを押下してください。

【入力】

```
Enter 1-12 or o-q >q
```

【実行結果】

```
Really terminate system generation ? (Y/N)>y
```

2.5 HiRDB Control Manager - Agent の環境設定をする

HiRDB Control Manager を使用できるようにするため、HiRDB サーバにインストールした HiRDB Control Manager - Agent の環境設定をします。HiRDB Control Manager を使用しない場合、この操作は必要ありません。

環境設定は、root (スーパーユーザ) で実行してください。

環境設定の手順を次に示します。

1. セットアップツールを実行する
セットアップツール「cmasetup」を実行します。

【コマンドの形式】

```
# /opt/hirdbCMA/bin/cmasetup /HiRDB
```

〔説明〕

/HiRDB

HiRDB の運用ディレクトリです。

【コマンドの実行結果】

```
KFPE34910-I HiRDB_CMA setup start, func = add, HiRDB directory = /HiRDB  
KFPE34935-I Create files necessary for execution
```

2. HiRDB Control Manager - Agent の機能を指定する
使用する HiRDB Control Manager - Agent の機能を指定します。このマニュアルでは、通常 Agent として使用するので、「Y」を指定します。

【入力】

```
Enter the type of CM Agent being setup [ Y : normal Agent / N : NetBackup Agent,  
default=Y, Q : quit ].  
==>Y
```

3. HiRDB Control Manager - Agent がインストールされているマシンの IP アドレスを指定する
HiRDB Control Manager - Agent がインストールされているマシンの IP アドレスを指定します。このマニュアルでは、「10.210.35.14」を指定します。

2. HiRDB システムを構築してみましょう

【入力】

```
Enter the IP address of the host where this CM Agent will run (for example
177.177.177.177) [ Q : quit ].
==>10.210.35.14
```

4. HiRDB Control Manager - Agent が使用するポート番号を指定する

HiRDB Control Manager - Agent が HiRDB Control Manager - Server からの要求を受け付けるために使用するポート番号を指定します。HiRDB Control Manager - Agent がインストールされているマシンで使われていないポート番号を指定してください。このマニュアルでは、「20293」を指定します。

【入力】

```
Enter a port number to be used by this CM Agent [ 5001 to 65535, default=20293,
Q : quit ].
==>20293
```

5. HiRDB Control Manager - Agent のサービス名を指定する

HiRDB Control Manager - Agent は、HiRDB Control Manager - Server からの要求を受け付けるために inetd デーモンを使用します。inetd がこの Agent を識別するためのサービス名称を指定します（12文字以内）。ここで指定したサービス名称は、`/etc/services` および `/etc/inetd.conf` にサービス名称として登録されます。これらのファイルに登録されているサービス名と重複しないようにしてください。このマニュアルでは、「hirdbcma」を指定します。

【入力】

```
Enter a name of the inetd service to be used by this CM Agent [ Q : quit ].
==>hirdbcma
```

6. HiRDB ユニットのノード種別を指定する

この HiRDB Control Manager - Agent が対象とする HiRDB ユニットのノード種別を指定します。このマニュアルでは、対象が HiRDB/ シングルサーバなので、「Y」を指定します。

【入力】

```
Enter the type of the HiRDB unit for which this CM Agent is being setup [ Y :  
system manager or single server / N : non-manager, default = Y, Q : quit ].  
==>Y
```

7. Rocket Punch 機能の使用するかしないかを指定する

Rocket Punch 機能を使用するかしないかを指定します。Rocket Punch 機能とは、管理対象の HiRDB を HiRDB Control Manager - Server に自動登録する機能です。自動登録すると、HiRDB Control Manager - Console で初めて HiRDB ユニットを使用するときに「HiRDB の追加」する必要がなくなります。

このマニュアルでは、「HiRDB の追加」操作も演習で行うので、「N」を指定します。

【入力】

```
Do you want HiRDB CM to automatically register this HiRDB unit to a CM Server?  
[ Y/N, default=N, Q : quit ]  
==>N
```

8. DBA 権限を持つユーザの認可識別子を指定する

HiRDB Control Manager - Agent が対象とする HiRDB の DBA 権限を持つユーザの認可識別子を指定します。このマニュアルでは、「MANAGER」を指定します。

【入力】

```
Enter a user name that has DBA privilege in the HiRDB system for which this CM  
Agent is being setup [ Q : quit ].  
==>MANAGER
```

9. DBA 権限を持つユーザのパスワードを指定する

項番 8 で指定したユーザのパスワードを指定します。このマニュアルでは、「MANAGER」を指定します。

【入力】

```
Enter the password of the user name that has the DBA privilege in the HiRDB  
system for which this CM Agent is being setup [ Q : quit ].  
==>MANAGER
```

10. HiRDB の文字コードを指定する

HiRDB Control Manager - Agent が対象とする HiRDB をセットアップしたときに指定した文字コード (PDLANG) を指定します。セットアップしたときに文字コードを指定しない場合は、「5」を指定します。このマニュアルでは、「5」を指定します。

2. HiRDB システムを構築してみましょう

【入力】

```
Choose the type of character codes that have been specified for the HiRDB system
[ 1 : sjis, 2 : ujis, 3 : utf-8, 4 : lang-c, 5 : default, Q : quit ].
==>5
```

11. HiRDB マシンの文字コードを指定する

HiRDB Control Manager - Agent が対象とする HiRDB が動作するマシンの文字コード（環境変数 LANG）を指定します。このマニュアルでは、「1」を指定します。

【入力】

```
Choose the LANG value of the server machine for which this CM Agent is being
setup [ 1 : ja_JP.eucJP, 2 : ja_JP, 3 : Ja_JP.ujis, 4 : others, default=1, Q :
quit ].
==>1
```

12. バックアップソフトウェアに、NetBackup を使うかどうかを指定する

バックアップソフトウェアに、NetBackup を使うかどうかを指定します。このマニュアルでは、「N」を指定します。

メッセージ「KFPE34911-I HiRDB_CMA setup ended, return code=0」が出力されれば、セットアップは終了です。

【入力】

```
Is NetBackup used? [ Y/N, default=N, Q : quit ]
==>N
```

【出力】

```
KFPE34911-I HiRDB_CMA setup ended, return code=0
```


2.6 HiRDB SQL Executer の環境設定をする (Windows マシン)

Windows のクライアントから HiRDB SQL Executer を使用できるようにするための環境設定を行います。HiRDB Control Manager を使用しない場合、HiRDB Control Manager を使って Windows 版の HiRDB SQL Executer を起動しない場合、この作業は必要ありません。

この作業は、Administrator で実行してください。

2.6.1 hosts ファイルの設定をする

Windows マシンの C:\Windows ディレクトリ下の hosts ファイルに、自マシン、および HiRDB サーバをインストールしたマシンの IP アドレスおよびホスト名を設定します。すでに情報が hosts ファイルに設定されている場合、または DNS サーバを使用する場合には、不要です。

(hosts ファイルの設定例)

```

:
10.210.35.14      HiRLIN_SV.soft.hitachi.co.jp HiRLIN_SV
10.210.35.24      s10380720      #HiRDBサーバ
:

```

2.6.2 クライアント環境定義の設定をする

クライアントの環境変数に、接続する HiRDB/ シングルサーバの情報、およびクライアントのユーザ名 / パスワードを設定します。環境変数に値が設定されている場合、HiRDB.ini ファイルの設定よりも優先されるので注意してください。

C:\Windows ディレクトリ下の HiRDB.ini ファイルに次の情報を設定してください。

なお、クライアント環境定義について知りたい場合は、マニュアル「HiRDB Version 7 UAP 開発ガイド (Windows(R) 用)」を参照してください。

```

PDHOST=HiRLIN_SV
PDNAMEPORT=20000
PDUSER=MANAGER/MANAGER
PDSQLLICENSE=YES

```

[説明]

2. HiRDB システムを構築してみましょう

PDHOST=HiRLIN_SV

接続する HiRDB のホスト名 (システム共通定義の pdstart オペランドの -x の指定値)

PDNAMEPORT=20000

接続する HiRDB のポート番号 (システム共通定義の pd_name_port オペランドの指定値)

PDUSER=MANAGER/MANAGER

HiRDB に接続するときのユーザ名, およびパスワード

PDSQLLICENSE=YES

HiRDB SQL Executer を使用するための設定

2.6.3 環境変数 PATH の設定をする

インストール時, 環境変数 PATH のバイト数が大きいと値が設定されない場合があります。そのような場合, ここで値を設定します。

C:\¥直下の AUTOEXEC.BAT ファイルに次の行を追加してください。

```
SET PATH=%PATH%;C:\¥PROGRA~1¥Hitachi¥HiRDB¥Utl
```

追加したら, Windows 2000 を再起動してください。

また, HiRDB/ クライアント環境変数登録ツールを使用すると, 接続ごとに環境変数の設定を変更することができるようになります。HiRDB/ クライアント環境変数登録ツールは, [スタート] - [プログラム] - [HiRDB Client] - [HiRDB クライアント環境変数登録ツール] から起動します。詳細については, マニュアル「HiRDB Version 7 UAP 開発ガイド (Windows(R) 用)」を参照してください。

2.7 表を作成する

ここでは、表の定義、および表へのデータロードについて説明します。図 2-6 に示すような表を二つ定義します。そして、それぞれ CUSTOM 表、VENDOR 表にデータロードをします。

図 2-6 定義する表

CUSTOM表

CHAR (5)	CHAR (30)	CHAR (12)	CHAR (3)	CHAR (30)
CUSTOM_CD	CUSTOM_NAME	TELNO	ZIPCD	ADDRESS

↑
インデクス
CUSTOMX

VENDOR表

CHAR (5)	CHAR (30)	CHAR (12)	CHAR (3)	CHAR (30)
VENDOR_CD	VENDOR_NAME	TELNO	ZIPCD	ADDRESS

↑
インデクス
VENDORX

2.7.1 ユーザ名を登録する

システムジェネレータ (pdgen) 実行直後、HiRDB を利用できるユーザは「hirman」だけです。

表の作成、および以降の運用は、DBA 権限を持つユーザ「MANAGER」で行います。「MANAGER」への DBA 権限付与は「hirman」が GRANT 文を使って行います。

権限付与の手順を次に示します。

権限付与の手順

1. HiRDB SQL Executer を起動します。/opt/pdsql/pdsql を実行してください。
2. 次の SQL 文を入力し、実行してください。

2. HiRDB システムを構築してみましょう

```
GRANT DBA TO MANAGER IDENTIFIED BY MANAGER;
```

3. 次のメッセージが出力され、DBA 権限を持ったユーザ「MANAGER」が作成されます。

```
KFPX270001-I Processing of SQL completed.
```

4. 以降、HiRDB SQL Executer を使った作業は、表の作成、および運用では、この「MANAGER」が操作するので、「2.4.1 HiRDB の環境変数を設定する」で設定した環境変数 PDUSER を、次のように変更してください。

```
PDUSER=MANAGER/MANAGER
```

5. HiRDB.ini ファイルに次の情報を設定します。HiRDB.ini ファイルは、通常 C:\¥Winnt ディレクトリ下にあります。
なお、環境変数 PDUSER に値が設定されている場合、HiRDB.ini ファイルの設定よりも優先されるので注意してください。

2.7.2 表を定義する

表を定義する場合、定義系 SQL を使用します。また、定義系 SQL はデータベース定義ユーティリティ (pddef) から実行します。

なお、この項では、表の定義に関する基本的なことだけ説明しています。詳細について知りたい場合は、マニュアル「HiRDB Version 7 システム導入・設計ガイド (UNIX(R) 用)」を参照してください。

(1) 「\$HOME/HiRDB/pddef/CreateT.txt」の作成

まず、定義系 SQL を入力したファイルを作成します。なお、セミコロン (;) の後ろには、空白を入れないでください。また、... n (n は 1 ~ 5) は説明上の文字列ですので、実際には入力しないでください。

なお、CreateT.txt には、インストール時に作成されるサンプルファイル /opt/HiRDB_S/sample/sampleDB/tblecreate_e を変更して使用しています。

【\$HOME/HiRDB/pddef/CreateT.txt の内容】

```

CREATE SCHEMA; .....1
CREATE FIX TABLE CUSTOM
(
  CUSTOM_CD          CHAR(5),
  CUSTOM_NAME       CHAR(30),
  TELNO             CHAR(12),
  ZIPCD             CHAR(3),
  ADDRESS           CHAR(30))
IN RDDATA10 PCTFREE=(10,0)
UNIQUE CLUSTER KEY
(
  CUSTOM_CD          ASC)
IN RDINDX10 PCTFREE=10; .....2
CREATE INDEX CUSTOMX ON CUSTOM
(
  CUSTOM_NAME       ASC)
IN RDINDX10 PCTFREE=10; .....3
CREATE FIX TABLE VENDOR
(
  VENDOR_CD         CHAR(5),
  VENDOR_NAME       CHAR(30),
  TELNO             CHAR(12),
  ZIPCD             CHAR(3),
  ADDRESS           CHAR(30))
IN RDDATA10 PCTFREE=(10,0)
UNIQUE CLUSTER KEY
(
  VENDOR_CD         ASC)
IN RDINDX10 PCTFREE=10; .....4
CREATE INDEX VENDORX ON VENDOR
(
  VENDOR_NAME       ASC)
IN RDINDX10 PCTFREE=10; .....5

```

〔説明〕

1. 表、およびインデクスを定義するので、先にスキーマを定義します。
2. CUSTOM 表を定義します。
表の格納先の RD エリアは、RDDATA10 です。
ページ内の未使用領域の比率は 10%、セグメント内の空きページ比率は 0%にします。
列「CUSTOM_CD」をクラスタキー（昇順）にします。
3. CUSTOM 表の列「CUSTOM_NAME」にインデクス CUSTOMX を定義します。
インデクスを格納する RD エリアは、RDINDX10 です。
ページ内の未使用領域の比率は 10%です。
4. VENDOR 表を定義します。
表の格納先の RD エリアは、RDDTATA10 です。
ページ内の未使用領域の比率は 10%、セグメント内の空きページ比率は 0%とします。
列「VENDOR_CD」をクラスタキー（昇順）にします。
5. VENDOR 表の列「VENDOR_NAME」にインデクス VENDORX を定義します。
インデクスを格納する RD エリアは、RDINDX10 です。
ページ内の未使用領域の比率は 10%です。

2. HiRDB システムを構築してみましょう

(2) pddef コマンドの実行

作成した定義系 SQL を入力にして、データベース定義ユティリティ (pddef) を実行します。

【コマンド形式】

```
$ pddef < $HOME/HiRDB/pddef/CreateT.txt
```

【説明】

「\$HOME/HiRDB/pddef/CreateT.txt」というファイルから、定義系 SQL を入力して実行しています。

(3) コマンドの実行結果の確認

データベース定義ユティリティが正常に終了すると、次のメッセージが表示されます。

```
KFPA12000-I Processing of SQL completed  
KFPA12000-I Processing of SQL completed  
KFPA12000-I Processing of SQL completed  
KFPA12000-I Processing of SQL completed  
KFPA12000-I Processing of SQL completed
```

(4) 表の確認方法

実際に表が定義されたかどうかを確認するために、HiRDB SQL Executer でディクショナリ表を検索します。

「2.7.1 ユーザ名を登録する」から HiRDB SQL Executer を運用し続けている場合、ユーザが「hirman」になっています。その場合、ユーザ名「MANAGER」、パスワード「MANAGER」で接続し直す必要があります。HiRDB SQL Executer 上で EXIT コマンドを実行して HiRDB サーバと切り離れたあと、/opt/pdsql/pdsql コマンドで接続してください。

【入力 SQL】

```
SELECT TABLE_SCHEMA, TABLE_NAME, TABLE_TYPE, N_COLS  
FROM MASTER.SQL_TABLES WHERE TABLE_SCHEMA='MANAGER';
```

【SQL の実行結果】

TABLE_SCHEMA	TABLE_NAME	TABLE_TYPE	N_COLS
MANAGER	CUSTOM	BASE TABLE	3
MANAGER	VENDOR	BASE TABLE	3

KFPX27010-I 2 rows selected

(5) インデクスの確認方法

表と同様に、実際にインデクスが定義されたかどうかを確認するために、HiRDB SQL Executer でディクショナリ表を検索します。

【入力 SQL】

```
SELECT TABLE_SCHEMA, TABLE_NAME, INDEX_NAME
FROM MASTER.SQL_INDEXES WHERE TABLE_SCHEMA='MANAGER';
```

【SQL の実行結果】

TABLE_SCHEMA	TABLE_NAME	INDEX_NAME
MANAGER	CUSTOM	(CLUSTER0000131193)
MANAGER	CUSTOM	CUSTOMX
MANAGER	VENDOR	(CLUSTER0000131194)
MANAGER	VENDOR	VENDORX

KFPX27010-I 4 rows selected

2.7.3 表ヘータロードをする

定義した表にデータロードをします。データロードをする場合、データベース作成ユーティリティ (pdload) を使用します。入力データファイルには、HiRDB インストール時に作成されたサンプルのデータを使用します。

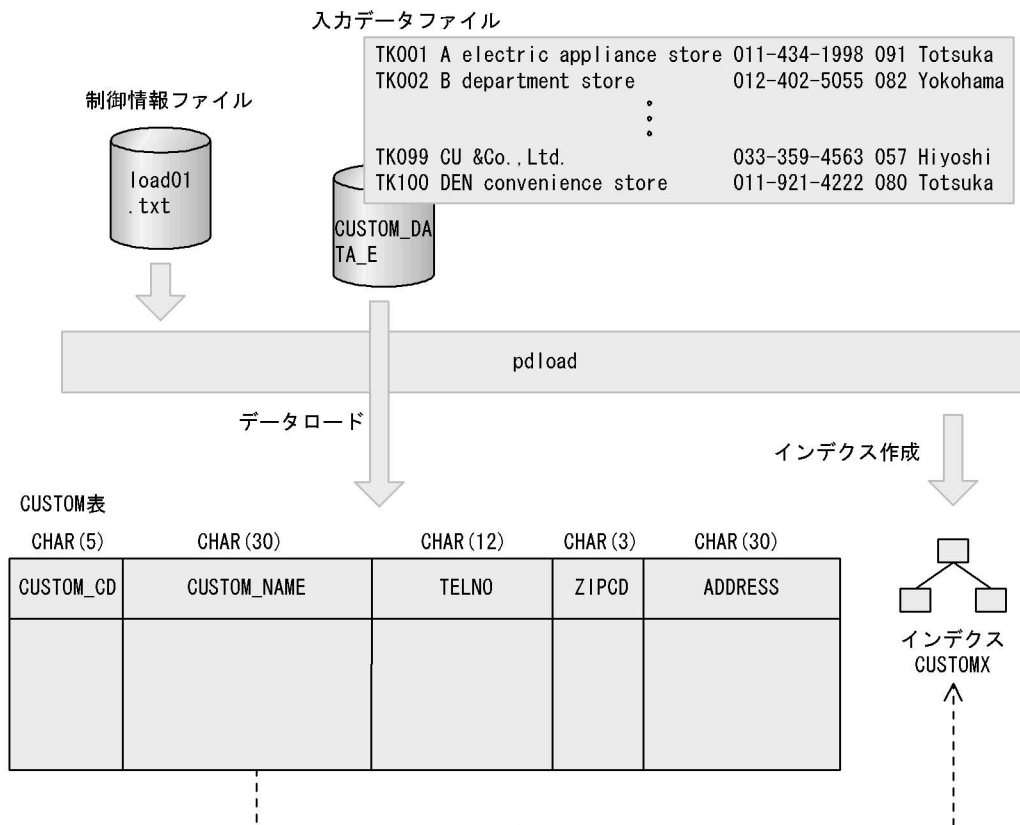
なお、この項では、データロードに関する基本的なことだけ説明しています。詳細について知りたい場合は、マニュアル「HiRDB Version 7 システム導入・設計ガイド (UNIX(R) 用)」を参照してください。

(1) CUSTOM 表へのデータロード

CUSTOM 表へのデータロードの概要を図 2-7 に示します。

2. HIRDB システムを構築してみましょう

図 2-7 CUSTOM 表へのデータロードの概要



(a) 「\$HOME/hirdb/pdload/load01.txt」の作成

まず、制御情報ファイル「\$HOME/hirdb/pdload/load01.txt」を作成します。

【\$HOME/hirdb/pdload/load01.txtの内容】

```
source /opt/HirDB_S/sample/sampleDB/loaddata/CUSTOM_DATA_E
```

〔説明〕

/opt/HirDB_S/sample/sampleDB/loaddata/CUSTOM_DATA_E :
 入力データファイルの名称です。

(b) pdload コマンドの実行

【コマンド形式】

```
$ pdload -u MANAGER -n 10 -i c -b CUSTOM $HOME/HirDB/pdload/load01.txt
```


注

コマンド実行後に、パスワードの入力要求が表示されるので、"MANAGER" と入力してください。

〔説明〕

-u MANAGER :

pdload を実行するユーザ名です。

-n 10 :

10 ページ単位で表に一括出力をする指定です。

-i c :

インデクスの作成方法の指定です。ここでは、インデクス一括作成モード (c) を指定しています。

-b :

入力ファイルのデータ形式です。ここでは、バイナリ形式を指定しています。

CUSTOM :

データロードする表の名称です。

\$HOME/HiRDB/pdload/load01.txt :

制御情報ファイルです。

(c) データベース作成ユティリティの実行結果の確認

データベース作成ユティリティが正常に終了すると、次のメッセージが表示されます。

2. HiRDB システムを構築してみましょう

```
2025 17:10:51 SDSU sds1      KFPL00709-I Error information file
was created, file=/tmp/ERROR-file8aGtAH
2025 17:10:51 SDSU sds1      KFPL00702-I Pdlod started,
table=MANAGER."CUSTOM"
2025 17:10:51 SDSU sds1      KFPL00710-I Index information
file assigned, index=MANAGER."(CLUSTER0000131193)",
RDAREA="RDINDEX10",
file=/tmp/INDEX-(CLUSTER0000131193)-RDINDEX10-filezutmsA
2025 17:10:51 SDSU sds1      KFPL00710-I Index information
file assigned, index=MANAGER."CUSTOMX", RDAREA="RDINDEX10",
file=/tmp/INDEX-CUSTOMX-RDINDEX10-filetnrikt
2025 17:10:52 SDSU sds1      KFPL00723-I 100 rows loaded,
table=MANAGER."CUSTOM", RDAREA="RDDATA10"
2198 17:10:52 SDSU 0mload0   KFPL00703-I 100 rows loaded
2199 17:10:52 SDSU sds1      KFPL00715-I Index load started
at sds1, index=MANAGER."(CLUSTER0000131193)", RDAREA="RDINDEX10"
2199 17:10:52 SDSU sds1      KFPL00716-I Index load ended
at sds1, index=MANAGER."(CLUSTER0000131193)",
RDAREA="RDINDEX10", return code=0
2199 17:10:52 SDSU sds1      KFPL00720-I Index file deleted,
server=sds1,
file=/tmp/INDEX-(CLUSTER0000131193)-RDINDEX10-filezutmsA
2200 17:10:52 SDSU sds1      KFPL00715-I Index load started
at sds1, index=MANAGER."CUSTOMX", RDAREA="RDINDEX10"
2200 17:10:52 SDSU sds1      KFPL00716-I Index load ended
at sds1, index=MANAGER."CUSTOMX", RDAREA="RDINDEX10",
return code=0
2200 17:10:52 SDSU sds1      KFPL00720-I Index file deleted,
server=sds1, file=/tmp/INDEX-CUSTOMX-RDINDEX10-filetnrikt
2197 17:10:52 SDSU _pd0load KFPL00704-I Pdlod terminated,
return code=0
```

(d) データロードした表の検索

実際に表にデータが格納されたかどうかを確認するために、HiRDB SQL Executer で CUSTOM 表を検索します。

【入力 SQL】

```
SELECT * FROM CUSTOM;
```

【SQL の実行結果】

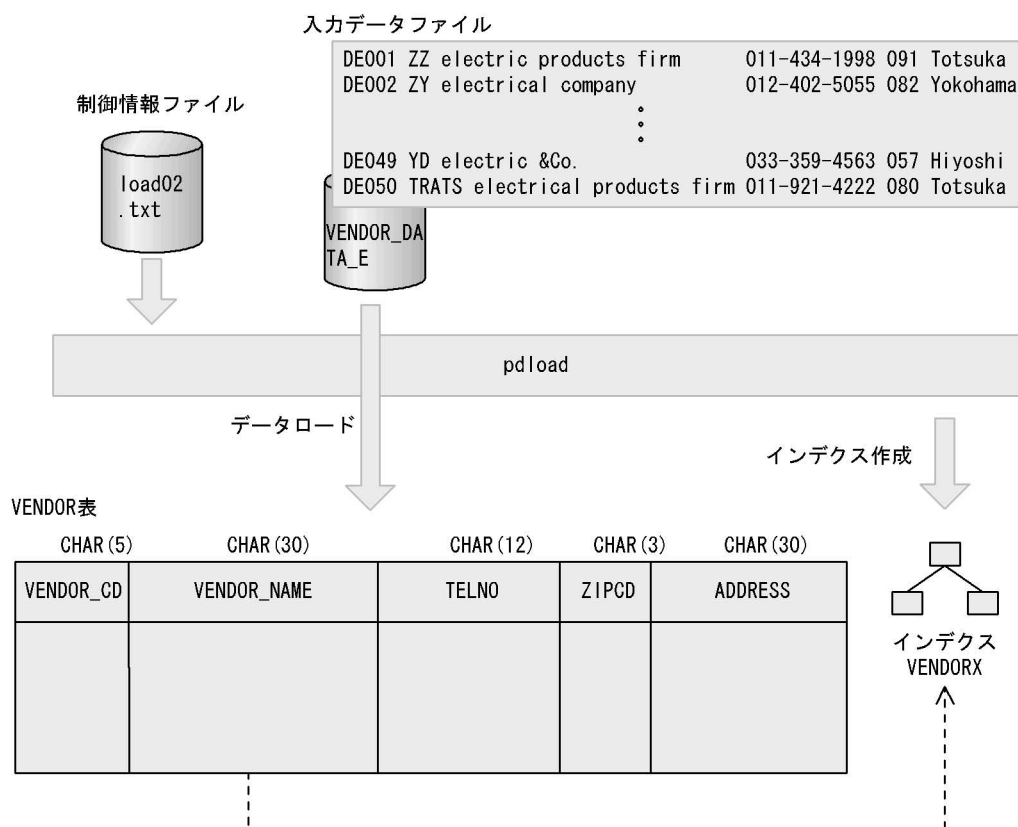
CUSTOM_CD	CUSTOM_NAME	TELNO	ZIPCD	ADDRESS
TK001	A electric appliance store	011-434-1998	091	Totsuka
TK002	B department store	012-402-5055	082	Yokohama
TK003	C products company	014-252-1555	043	Kanazawa
	⋮			
TK098	CT electric appliance store	052-476-5624	066	Fujiyama
TK099	CU &Co.,Ltd.	033-359-4563	057	Hiyoshi
TK100	DEN convenience store	011-921-4222	080	Totsuka

KFPX27010-I 100 rows selected

(2) VENDOR 表へのデータロード

VENDOR 表へのデータロードの概要を図 2-8 に示します。

図 2-8 VENDOR 表へのデータロードの概要



(a) 「\$HOME/HirDB/pdload/load02.txt」の作成

まず、制御情報ファイル「\$HOME/HirDB/pdload/load02.txt」を作成します。

【\$HOME/HirDB/pdload/load02.txt の内容】

2. HiRDB システムを構築してみましょう

```
source /opt/HiRDB_S/sample/sampleDB/loaddata/VENDOR_DATA_E
```

〔説明〕

/opt/HiRDB_S/sample/sampleDB/loaddata/VENDOR_DATA_E :
入力データファイルの名称です。

(b) pdload コマンドの実行

【コマンド形式】

```
$ pdload -u MANAGER -n 10 -i c -b VENDOR $HOME/HiRDB/pdload/load02.txt
```

注

コマンド実行後に、パスワードの入力要求が表示されるので、"MANAGER" と入力してください。

〔説明〕

-u MANAGER :

pdload を実行するユーザ名です。

-n 10 :

10 ページ単位で表に一括出力をする指定です。

-i c :

インデクスの作成方法の指定です。ここでは、インデクス一括作成モード (c) を指定しています。

-b :

入力ファイルのデータ形式です。ここでは、バイナリ形式を指定しています。

VENDOR :

データロードする表の名称です。

\$HOME/HiRDB/pdload/load02.txt :

制御情報ファイルです。

(c) データベース作成ユーティリティの実行結果の確認

データベース作成ユーティリティが正常に終了すると、次のメッセージが表示されます。

```

2201 17:15:06 SDSU sds1      KFPL00709-I Error information
file was created, file=/tmp/ERROR-filejYXDOB
2201 17:15:06 SDSU sds1      KFPL00702-I Pdload started,
table=MANAGER."VENDOR"
2201 17:15:06 SDSU sds1      KFPL00710-I Index information
file assigned, index=MANAGER."(CLUSTER0000131194)",
RDAREA="RDINDX10",
file=/tmp/INDEX-(CLUSTER0000131194)-RDINDX10-fileBnELOX
2201 17:15:06 SDSU sds1      KFPL00710-I Index information
file assigned, index=MANAGER."VENDORX",
RDAREA="RDINDX10", file=/tmp/INDEX-VENDORX-RDINDX10-fileBgJWOj
2201 17:15:06 SDSU sds1      KFPL00723-I 50 rows loaded,
table=MANAGER."VENDOR", RDAREA="RDATA10"
2215 17:15:06 SDSU Omload0   KFPL00703-I 50 rows loaded
2216 17:15:06 SDSU sds1      KFPL00715-I Index load started
at sds1, index=MANAGER."(CLUSTER0000131194)",
RDAREA="RDINDX10"
2216 17:15:06 SDSU sds1      KFPL00716-I Index load ended
at sds1, index=MANAGER."(CLUSTER0000131194)",
RDAREA="RDINDX10", return code=0
2216 17:15:06 SDSU sds1      KFPL00720-I Index file deleted,
server=sds1,
file=/tmp/INDEX-(CLUSTER0000131194)-RDINDX10-fileBnELOX
2217 17:15:06 SDSU sds1      KFPL00715-I Index load started
at sds1, index=MANAGER."VENDORX", RDAREA="RDINDX10"
2217 17:15:06 SDSU sds1      KFPL00716-I Index load ended
at sds1, index=MANAGER."VENDORX", RDAREA="RDINDX10",
return code=0
2217 17:15:06 SDSU sds1      KFPL00720-I Index file deleted,
server=sds1, file=/tmp/INDEX-VENDORX-RDINDX10-fileBgJWOj
2214 17:15:06 SDSU _pd0load KFPL00704-I Pdload terminated,
return code=0

```

(d) データロードした表の検索

実際に表にデータが格納されているかどうかを確認するために、HiRDB SQL Executer で VENDOR 表を検索します。

【入力 SQL】

```
SELECT * FROM VENDOR;
```

【SQL の実行結果】

2. HIRDB システムを構築してみましょう

VENDOR_CD	VENDOR_NAME	TELNO	ZIPCD	ADDRESS
DE001	ZZ electric products firm	011-434-1998	091	Totsuka
DE002	ZY electrical company	012-402-5055	082	Yokohama
DE003	ZX machinery shop	014-252-1555	043	Kanazawa
	:			
DE048	YE machinery shop	052-476-5624	066	Fujiyama
DE049	YD electric &Co.	033-359-4563	057	Hiyoshi
DE050	TRATS electrical products firm	011-921-4222	080	Totsuka
KFPX27010-I				50 rows selected

3

表を操作してみましよう

この章では、HiRDB SQL Executer を使用して、2 章で作成した表を操作する方法について説明します。

3.1 表を操作するためには

3.2 HiRDB SQL Executer を起動する

3.3 表を操作する

3.4 HiRDB SQL Executer を終了する

3.1 表を操作するためには

通常は、C 言語や COBOL 言語などのソース中に SQL を埋め込んで UAP を作成したり、または別のプログラムプロダクトを使用したりして表にアクセスしますが、ここでは対話形式で簡単に表にアクセスできる HiRDB SQL Executer を使用します。UAP を作成して表を操作する方法については、「7. アプリケーションを作成してみよう」を参照してください。

表のデータを操作する場合は、操作系 SQL を使用します。この章では、次の操作系 SQL を使用します。

操作系 SQL の種類

SQL 文	内容
SELECT 文	表からデータを検索します。
INSERT 文	表に行を挿入します。
UPDATE 文	表のデータを更新します。
DELETE 文	表から行を削除します。

注

ここで使用する操作系 SQL は、SELECT 文、INSERT 文、UPDATE 文、および DELETE 文だけです。ほかの操作系 SQL について知りたい場合は、マニュアル「HiRDB Version 7 SQL リファレンス (UNIX(R) 用)」を参照してください。

3.2 HiRDB SQL Executer を起動する

HiRDB SQL Executer を起動します。/opt/pdsql/pdsql を実行してください。

なお、ここでの演習問題は認可識別子（ユーザ名）「MANAGER」、パスワード「MANAGER」で実行します。すでに、環境変数に認可識別子（ユーザ名）「MANAGER」、パスワード「MANAGER」が設定されているので、SQL 実行時に認可識別子（ユーザ名）、パスワードを意識する必要はありません。

3. 表を操作してみましょう

3.3 表を操作する

この節では、表の操作に関する基本的なことだけ説明しています。詳細について知りたい場合は、マニュアル「HiRDB Version 7 SQL リファレンス (UNIX(R) 用)」を参照してください。

3.3.1 演習 1 (表を検索する)

(1) LIKE 述語を使用した検索

CUSTOM 表から、CUSTOM_NAME 中に「shop」が含まれる CUSTOM_NAME とその CUSTOM_CD を検索します。

【SQL 文】

```
SELECT CUSTOM_NAME, CUSTOM_CD FROM CUSTOM WHERE CUSTOM_NAME LIKE '%shop%';
```

【実行結果】

CUSTOM_NAME	CUSTOM_CD
AK House shop	TK037
AV shopping store	TK048
BB shop	TK054
BH shop	TK059
CG food shop	TK085
CH nattou shop	TK086
CM food shop	TK091
CR shop	TK096
Z shop	TK026

KFPX27010-I 9 rows selected

3.3.2 演習 2 (表へ行を挿入する)

VENDOR 表に、'DE051', 'H Software div.', '011-888-8110', '080','Totsuka' のデータを挿入します。

【SQL 文】

```
INSERT INTO VENDOR VALUES ('DE051', 'H Software div.', '011-888-8110', '080', 'Totsuka');
```

【実行結果】

KFPX27010-I 1 rows inserted

"1 行挿入しました" という意味のメッセージが表示されます。実際に値が挿入されているかどうかを確認するために、次の SQL 文を実行します。

```
SELECT * FROM VENDOR;
```

SQL 文を実行すると、次の検索結果が表示されます。

VENDOR_CD	VENDOR_NAME	TELNO	ZIPCD	ADDRESS
DE001	ZZ electric products firm	011-434-1998	091	Totsuka
DE002	ZY electrical company	012-402-5055	082	Yokohama
DE003	ZX machinery shop	014-252-1555	043	Kanazawa
	⋮			
DE043	YJ machinery shop	091-158-4235	076	Nakayama
DE051	H Software div.	011-888-8110	080	Totsuka
DE044	YI electrical construction	034-358-6453	084	Fuchuu
DE045	YH industrial company	011-753-6342	025	Funabashi
DE046	YG electric appliance store	013-536-2221	045	Heiwajima
DE047	YF electric transmission	021-159-9876	035	Dorogame
DE048	YE machinery shop	052-476-5624	066	Fujiyama
DE049	YD electric &Co.	033-359-4563	057	Hiyoshi
DE050	TRATS electrical products firm	011-921-4222	080	Totsuka

KFPX27010-I 51 rows selected

3.3.3 演習 3 (表の行を更新する)

VENDOR 表の、VENDOR_CD が DE001 の ZIPCD を 080 に変更します。

【SQL 文】

```
UPDATE VENDOR SET ZIPCD='080' WHERE VENDOR_CD='DE001';
```

【実行結果】

KFPX27010-I 1 rows updated

"1 行更新しました" という意味のメッセージが表示されます。実際に値が更新されているかどうかを確認するために、次の SQL 文を実行します。

```
SELECT * FROM VENDOR;
```

SQL 文を実行すると、次の検索結果が表示されます。

VENDOR_CD	VENDOR_NAME	TELNO	ZIPCD	ADDRESS
DE001	ZZ electric products firm	011-434-1998	080	Totsuka
DE002	ZY electrical company	012-402-5055	082	Yokohama
DE003	ZX machinery shop	014-252-1555	043	Kanazawa
	⋮			

3. 表を操作してみましょう

3.3.4 演習 4 (表の行を削除する)

VENDOR 表の、VENDOR_CD が DE049 の行を削除します。

【SQL 文】

```
DELETE FROM VENDOR WHERE VENDOR_CD='DE049';
```

【実行結果】

```
KFPX27010-I          1 rows deleted
```

"1 行削除しました" という意味のメッセージが表示されます。実際に値が削除されているかどうかを確認するために、次の SQL 文を実行します。

```
SELECT * FROM VENDOR;
```

SQL 文を実行すると、次の検索結果が表示されます。

VENDOR_CD	VENDOR_NAME	TELNO	ZIPCD	ADDRESS
DE001	ZZ electric products firm	011-434-1998	080	Totsuka
DE002	ZY electrical company	012-402-5055	082	Yokohama
DE003	ZX machinery shop	014-252-1555	043	Kanazawa
	⋮			
DE047	YF electric trancemission	021-159-9876	035	Dorogame
DE048	YE machinery shop	052-476-5624	066	Fujiyama
DE050	TRATS electrical products firm	011-921-4222	080	Totsuka

KFPX27010-I 50 rows selected

SQL 文を実行すると、次の検索結果が表示されます。

3.4 HiRDB SQL Executer を終了する

HiRDB SQL Executer を使用した表の操作を終了します。EXIT コマンドで終了してください。

4

HiRDB を運用してみましょ う

この章では、HiRDB の基礎的な運用方法について説明します。

-
- 4.1 この章を読むに当たって

 - 4.2 HiRDB を開始，終了する

 - 4.3 システムログを操作する

 - 4.4 バックアップを取得する

 - 4.5 データベースを回復する

 - 4.6 表を再編成する

 - 4.7 RD エリアを拡張する

 - 4.8 RD エリアを追加する

 - 4.9 ユーザ権限を与える（ユーザを登録する）
-

4.1 この章を読むに当たって

この章を読むに当たって、知っておいていただきたいことについて説明します。

(1) この章で説明している項目

この章では、HiRDB の基礎的な運用方法について説明しています。説明している項目は次のとおりです。

- HiRDB の開始，終了
- システムログのアンロード
- バックアップの取得
- データベースの回復
- 表の再編成
- RD エリアの拡張
- RD エリアの追加
- ユーザ権限の付与（ユーザの登録）
- なお，各節で説明している運用方法は基礎的なことだけです。運用方法の詳細を知りたい方は，マニュアル「HiRDB Version 7 システム運用ガイド（UNIX(R) 用）」を参照してください。

(2) この章の読み方について

この章は，章全体で一つの演習になっています。したがって，途中から読まないで，4.2 から順次読み進んでください。

4.2 HiRDB を開始，終了する

HiRDB の開始と終了について説明します。

4.2.1 基礎知識

HiRDB の開始には `pdstart` コマンド (HiRDB システム・ユニット・サーバの開始) を，HiRDB の終了には `pdstop` コマンド (HiRDB システム・ユニット・サーバの終了) を使います。

(1) HiRDB の開始

HiRDB の開始には，次に示す二つの方法があります。

1. HiRDB のコマンド (`pdstart` コマンド) で開始する方法
2. OS の起動と同時に自動で開始する方法

通常は，1. の方法で HiRDB を開始するようにしてください。

(a) HiRDB の開始方法の決め方

HiRDB を 1. の方法で開始するか，または 2. の方法で開始するかは，システム共通定義 (`pdsys`) の `pd_mode_conf` オペランドで指定します。

1. の方法で開始する場合

`pd_mode_conf = MANUAL2` を指定します。

2. の方法で開始する場合

`pd_mode_conf = AUTO` を指定します。このマニュアルの手順に従って HiRDB の環境設定をした場合は，`pd_mode_conf = AUTO` が指定されています。

(b) 開始モード

HiRDB には開始モードという概念があります。開始モードによって，`pdstart` コマンドに指定するオプションが異なります。HiRDB の開始モードを表 4-1 に示します。

表 4-1 HiRDB の開始モード

開始モード	実行コマンド	開始モードの説明	前回の終了モード
正常開始	<code>pdstart</code>	通常の開始モードです。正常開始では，前回稼働時の情報を引き継ぎません。ただし，障害閉塞している RD エリアの状態は引き継ぎます。	正常終了
再開始		前回の終了モードが右記の場合，HiRDB は (自動的に) 再開始となります。再開始では，前回稼働時の情報を引き継ぎません。	<ul style="list-style-type: none"> • 計画停止 • 強制終了 • 異常終了

4. HiRDB を運用してみましょう

開始モード	実行コマンド	開始モードの説明	前回の終了モード
強制開始	pdstart dbdestroy	通常は使用しないでください。HiRDB を再開できないときに使用します。HiRDB を強制的に開始します。	-
-	pdstart -i	通常は使用しないでください。データベースを再度初期設定するときの開始方法です。	-
-	pdstart -r	通常は使用しないでください。マスタディレクトリ用 RD エリアに障害が発生したときの開始方法です。	-

(凡例)

- : モードに依存しません。

(2) HiRDB の終了

HiRDB の終了は、pdstop コマンドで行います。

(a) 終了モード

HiRDB には終了モードという概念があります。終了モードによって、pdstop コマンドに指定するオプションが異なります。HiRDB の終了モードを表 4-2 に示します。

表 4-2 HiRDB の終了モード

終了モード	入力コマンド	終了モードの説明
正常終了	pdstop	通常の終了モードです。CONNECT 要求を禁止し、ユティリティを含むすべてのユーザが DISCONNECT したあとに、HiRDB を終了します。
計画停止	pdstop -P	トランザクションの受け付けを禁止し、ユティリティを含むすべてのトランザクションが終了したあとに、HiRDB を終了します。
強制終了	pdstop -f	処理中のトランザクションの完了を待たずに、HiRDB を直ちに終了します。処理中のトランザクションは、再開時にロールバックの対象となります。
異常終了	-	何らかの異常によって HiRDB が終了する場合の終了モードです。処理中のトランザクションの完了を待たずに、HiRDB は直ちに終了します。処理中のトランザクションは、再開時にロールバックの対象となります。

(b) 注意事項

- HiRDB に接続中のユーザがいると HiRDB を終了できません。接続中のユーザがいるかどうかは、pdls コマンドでわかります。
- OS をシャットダウンする場合は、HiRDB が終了しているかどうかを確認してください。HiRDB が終了しているかどうかは、pdls コマンドでわかります。

4.2.2 演習 1 (HiRDB を正常終了する)

稼働中の HiRDB を正常終了します。

(1) pdls コマンドで HiRDB の稼働状況を調べます

HiRDB のシステム状態を表示して、稼働中かどうかを調べます。次のコマンドを実行してください。

```
$ pdls
```

(2) pdls コマンドの実行結果が表示されます

HOSTNAME (174751)	UNITID	SVID	STATUS	STARTTIME
HIRLIN_SV	SDSU	*****	ACTIVE	164914
HIRLIN_SV	SDSU	sds1	ACTIVE	164914

〔説明〕

STATUS に ACTIVE が表示されているので、HiRDB が稼働中です。

(3) pdls コマンドで HiRDB に接続中のユーザがいるかどうかを調べます

サーバのプロセスの状態を表示して、HiRDB に接続しているユーザがいるかどうかを調べます。HiRDB に接続中のユーザがいると、HiRDB を終了できません。次のコマンドを実行してください。

```
$ pdls -d prc
```

(4) pdls コマンドの実行結果が表示されます

HOSTNAME : HIRLIN_SV (174948)	STATUS	PID	UID	GID	SVID	TIME	PROGRAM	C-PID	C-GRP
	L	2024	500	501	sds1	999999			
	L	2218	500	501	sds1	999999			
	L	2219	500	501	sds1	999999			

〔説明〕

- HiRDB に接続中のユーザがいると、PROGRAM に UAP の識別名称が表示されます。したがって、現在 HiRDB に接続中のユーザはいません。HiRDB に接続中のユーザがいると、HiRDB を終了できません。

4. HiRDB を運用してみましょう

例えば、3章の演習で HiRDB SQL Executer を使用したままにしていると (HiRDB に接続したままにしていると), HiRDB を終了できません。HiRDB を終了するには、HiRDB SQL Executer を終了してください。

- PROGRAM には、クライアント環境定義の PDCLTAPNAME オペランドに指定した UAP の識別名称が表示されます。PDCLTAPNAME オペランドを省略すると、Unknown が表示されます。

(5) pdstop コマンドで HiRDB を正常終了します

HiRDB を正常終了します。次のコマンドを実行してください。

```
$ pdstop
```

(6) pdstop コマンドの実行結果が表示されます

```
2009 17:50:19 SDSU _rdm      KFPS01841-I HiRDB unit SDSU terminated.
mode = NORMAL
2009 17:50:19 SDSU _rdm      KFPS01850-I HiRDB system terminated.
mode = NORMAL
```

4.2.3 演習 2 (HiRDB を正常開始する)

HiRDB を正常開始します。

(1) pdstart コマンドで HiRDB を正常開始します

HiRDB を正常開始します。次のコマンドを実行してください。

```
$ pdstart
```

(2) pdstart コマンドの実行結果が表示されます

```
2624 17:50:46 SDSU _rdm      KFPS05210-I HiRDB system initialization
process complete
```

(3) pdls コマンドで HiRDB の稼働状況を調べます

HiRDB のシステム状態を表示して、稼働中かどうかを調べます。次のコマンドを実行してください。

```
$ pdls
```

(4) pdls コマンドの実行結果が表示されます

HOSTNAME(175215)	UNITID	SVID	STATUS	STARTTIME
HIRLIN_SV	SDSU	*****	ACTIVE	175045
HIRLIN_SV	SDSU	sds1	ACTIVE	175045

〔説明〕

STATUS に ACTIVE が表示されているので、HiRDB が開始しました。

4.3 システムログを操作する

バックアップの取得，およびデータベースの回復の演習を行う前に，システムログの運用方法について理解しておく必要があります。ここでは，システムログの操作について説明します。

4.3.1 基礎知識

データベースの障害が発生したとき，障害発生前の最新の状態に回復するためにはバックアップデータ以外にシステムログファイルが必要になります。このシステムログの運用によって，データベースをどの時点まで回復できるか，バックアップをどのように取得するかなどが異なります。

(1) システムログとは

システムログとは，一般的にはジャーナルとも呼ばれているデータベースの更新履歴情報のことです。HiRDBはこのシステムログを次に示す目的のためにシステムログファイルに取得しています。

(a) HiRDB または UAP が異常終了した場合に HiRDB がデータベースを回復するときの入力情報になります

HiRDB または UAP が異常終了すると，実行中の UAP が更新したデータベースの内容をロールバックする必要が生じます。この処理は HiRDB が行います。このときの入力情報がシステムログとなります。

(b) pdrstr コマンド（データベース回復ユティリティ）の入力情報になります

システムログにはデータベースの更新履歴情報が格納されているので，データベースに障害が発生したときでも，バックアップファイルとそのバックアップを取得した時点以降の更新履歴を利用して，データベースを最新の状態に回復できます。

ディスク障害など，データベースを回復する必要が生じた場合，pdrstr コマンドでデータベースを回復します。このとき，次に示す入力情報が必要になります。

- バックアップファイル
- システムログファイル，またはアンロードログファイル

注

バックアップ取得時点に回復する場合は必要ありません。また，バックアップ取得時のオプションによっては，使えないことがあります。

アンロードログファイル

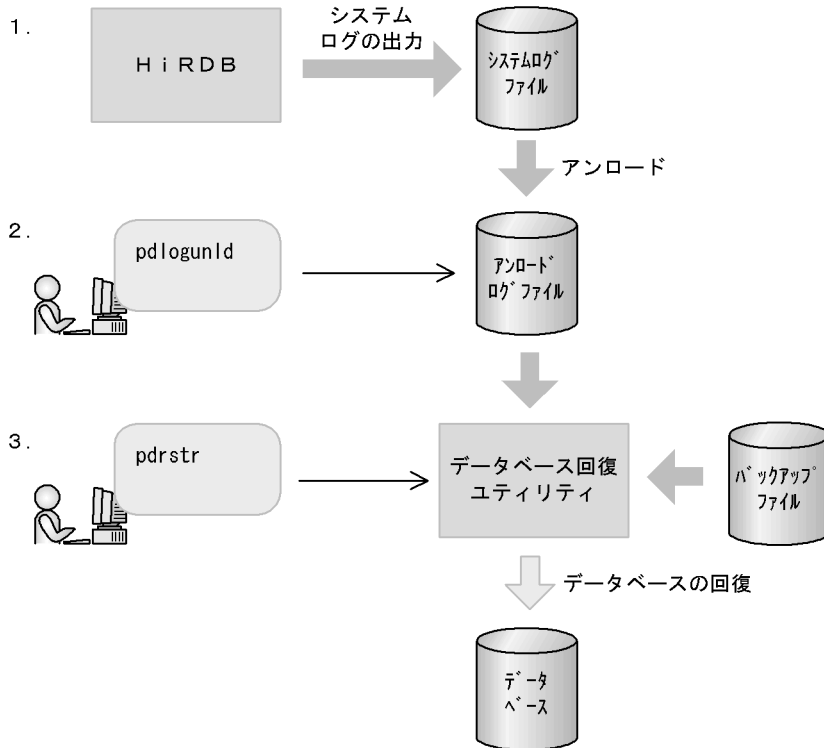
アンロードログファイルは，システムログをアンロードすることで作成することができます。システムログのアンロードは，pdlogunld コマンド（ログ関係ファイルの

アンロード)で行います。

アンロードログファイルを使用すると、データベースに障害が発生したときに、RD エリアごとに障害を回復することができます。システムログファイルを使用したデータベースの回復では、システム全体の回復しかできません。

このアンロードログファイルはシステムログをアンロードして作成します。システムログとデータベース回復の関係を図 4-1 に示します。

図 4-1 システムログとデータベース回復の関係



〔説明〕

1. HiRDB がシステムログファイルにシステムログを出力します。
2. pdlogunld コマンドで、システムログをアンロードします。
3. pdrstr コマンド（データベース回復ユーティリティ）でデータベースを回復します。

(2) システムログの運用の種類

システムログの運用の種類には、表 4-3 に示す方法があります。このマニュアルでは、システムログをアンロードする運用で HiRDB を運用します。「2.3.2(3) シングルサーバ定義 (sds1)」で紹介した自動ログアンロード機能は、システムログをアンロードする運用で HiRDB を運用する場合に利用できます。それぞれの運用方法の詳細については、「HiRDB Version 7 システム運用ガイド (UNIX(R) 用)」を参照してください。

4. HiRDB を運用してみましょう

表 4-3 システムログの運用の種類

運用の種類	運用方法	データベースの回復方法	利点
システムログをアンロードする運用	<ul style="list-style-type: none"> アンロード待ち状態のシステムログをアンロードします。 	<ul style="list-style-type: none"> バックアップおよびアンロードログファイルを入力情報にしてデータベースを回復します。 データベースは、バックアップ取得時点、およびバックアップ取得時点以降の任意の同期点に回復できます。 	<ul style="list-style-type: none"> データベースを最新の状態で回復できます。 RD エリア単位でデータベースを回復できます。 システムログのアンロードを自動化できます。
アンロードレスシステムログ運用	<ul style="list-style-type: none"> アンロード待ち状態のシステムログを解放します（アンロードする必要はありません）。 サーバ単位にバックアップを取得する必要があります。 	<ul style="list-style-type: none"> バックアップおよびバックアップ取得以降のシステムログを入力情報にしてデータベースを回復します。 データベースは、バックアップ取得時点、およびバックアップ取得時点以降の任意の同期点に回復できます。 	<ul style="list-style-type: none"> システムログのアンロード操作がなくなるため、CPU および入出力の負荷を削減でき、HiRDB 管理者の運用が簡単になります。 アンロードログファイルを保管するためのディスク容量が必要なくなります。
アンロード状態のチェックを解除する運用	<ul style="list-style-type: none"> システムログをアンロードする必要はありません。 	<ul style="list-style-type: none"> バックアップを入力情報にしてデータベースを回復します。 データベースは、バックアップ取得時点にしか回復できません。 	<ul style="list-style-type: none"> システムログファイルのアンロード操作または解放操作がなくなるため、運用が簡単になります。 アンロードログファイルを保管するためのファイル容量が必要なくなります。

この節には、システムログをアンロードする運用の場合の演習を記載しています。

(3) システムログをアンロードする運用

システムログをアンロードする運用の場合、(1) の (b) で説明したように、データベースの障害に備えてアンロードログファイルを保管しておく必要があります。

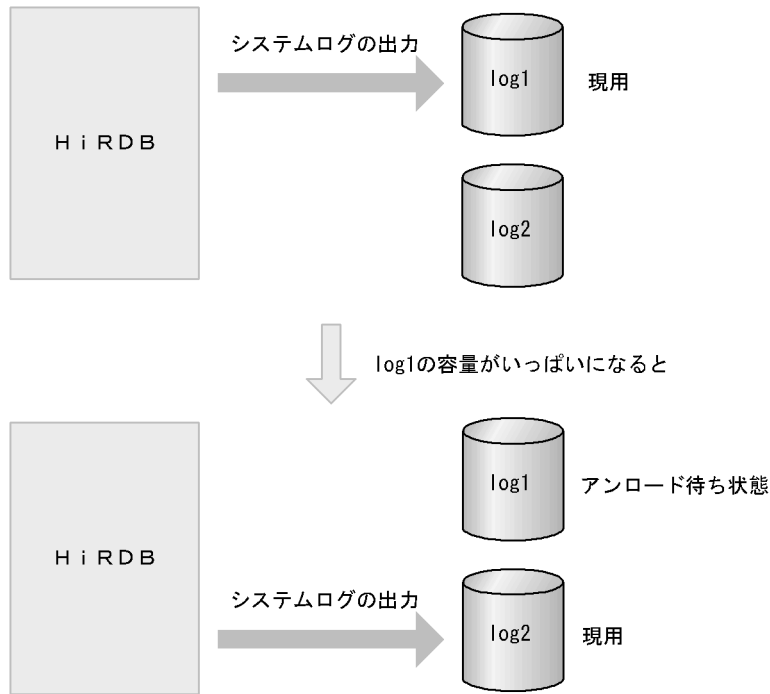
(a) 「アンロード待ち状態」のファイルをアンロードします

システムログが出力されているファイルを「現用」ファイルといいます。この「現用」ファイルの容量いっぱいシステムログが出力されると、システムログの出力先がほかのシステムログファイルに変更されます。これをシステムログファイルのスイッチングといいます。

「現用」であったファイルは「アンロード待ち状態」のファイルとなります。HiRDB 管理者は、この「アンロード待ち状態」のファイルを `pdlogunld` コマンドでアンロードします。

システムログファイルの状態変化を図 4-2 に示します。

図 4-2 システムログファイルの状態変化



システムログファイルの状態は、pdlogls コマンドで確認できます。pdlogls 出力例を次に示します。

```

HOSTNAME : host(180621)
Group   Type Server  Gen No.  Status  Run ID   Block No.  Ex-Status
logfg01 sys  sds1     1  osu---u 4019e4a7   1         6  -----
logfg02 sys  sds1     1  oc-d-u 401f5271   1        40  -----
logfg03 sys  sds1     1  os----- 401f60de   1         8  -----
logfg04 sys  sds1     2  os----- 401f60de   9         9  -----
logfg05 sys  sds1     0  os----- 00000000   0         0  -----
logfg06 sys  sds1     0  os----- 00000000   0         0  -----

```

システムログファイルの状態はここでチェックします。
この2カラム目と3カラム目を見ます。

〔説明〕

- 3カラム目が「u」になっているファイル (logfg01) が、「アンロード待ち状態」のファイルです。
- 2カラム目が「c」になっているファイル (logfg02) が、「現用」のファイルです。

4. HiRDB を運用してみましょう

(b) アンロードをしないと HiRDB が異常終了します

「アンロード待ち状態」のファイルにはシステムログを出力できません。全システムログファイルが「アンロード待ち状態」になると、システムログが出力できなくなり、HiRDB が異常終了します。したがって、システムログのアンロードは忘れずに実行してください。

(c) 自動ログアンロード機能

システムログをアンロードする運用では、pdlogunld コマンドでアンロード待ち状態のシステムログファイルをアンロードする必要があります。この作業を忘れるとスワップ先にできる状態のシステムログファイルがなくなり、HiRDB が異常終了します。これを回避するための機能として、自動ログアンロード機能があります。自動ログアンロード機能を使うと、システムログファイルのアンロード作業を HiRDB が自動的に行います。アンロード作業が頻繁に発生する場合は、システムログのアンロード作業の自動化を検討してください。

HiRDB が自動的に作成するアンロードログファイルの名称には次に示すような規則があり、アンロードされた順序がわかるようになっています。したがって、バックアップ取得時の最新のアンロードログファイルを調べておくと、データベースの回復時にはそれ以降に作成されたアンロードログファイルを使うことでデータベースを最新の状態に回復できます。

名称規則

サーバ名_サーバラン ID と世代番号_ファイルグループ名称

(例)

```
sds01_404d793d0001_log1   バックアップ直前にアンロードしたファイル
sds01_404d793d0002_log2
sds01_404d793d0003_log3
sds01_404d793d0004_log4
```

通常は、サーバを停止するときはバックアップを取得するときなので、サーバラン ID が異なる複数のアンロードログファイルがデータベースの回復に必要なことはありません。バックアップ取得時に、次にアンロードされる予定のアンロードログファイル名を調べておけば、世代番号に注目することでバックアップ取得以降に作成されたアンロードログファイルを把握することができます。

上記の例の場合、「sds01_404d793d0001_log1」がバックアップ直前にアンロードしたファイルとすると、取得したバックアップを使ってデータベースを回復するには、「sds01_404d793d0002_log2」以降の三つのファイルが必要になります。

自動ログアンロード機能の詳細については、マニュアル「HiRDB Version 7 システム運用ガイド (UNIX(R) 用)」を参照してください。

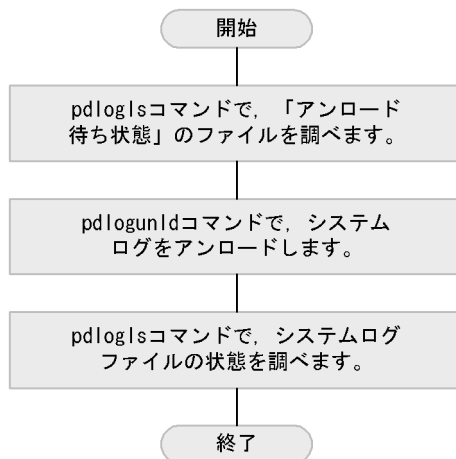
(d) アンロードログファイルの格納場所

このあとに行う演習ではアンロードログファイルをデータベースと同じディスクに格納していますが、本番のシステムではアンロードログファイルをデータベースと異なるディスクに格納してください。同じディスクに格納すると、そのディスクに障害が発生した場合、データベースを回復できなくなります。

(4) システムログのアンロード手順

システムログをアンロードする手順を図 4-3 に示します。

図 4-3 システムログをアンロードする手順



(5) 演習を始める前に

「アンロード待ち状態」のファイルがない場合は、pdlogswap コマンドでシステムログファイルをスワップさせて、「アンロード待ち状態」のファイルを作成してください。

```
$ pdlogswap -d sys
```

4.3.2 演習 (システムログをアンロードする)

「アンロード待ち状態」のシステムログファイルをアンロードします。

(1) pdlogls コマンドでシステムログファイルの状態をチェックします

```
$ pdlogls -d sys
```

4. HIRDB を運用してみましょう

(2) pdlogls コマンドの実行結果が表示されます

```
HOSTNAME : HIRLIN_SV(180621)
Group    Type Server  Gen No.  Status  Run ID      Block No.  Ex-Status
logfg01  sys  sds1     1  osu---u  4019e4a7    1    6  -----
logfg02  sys  sds1     1  osu---u  401f5271    1   40  -----
logfg03  sys  sds1     1  osu---u  401f60de    1    8  -----
logfg04  sys  sds1     2  oc-d--u  401f60de    9    9  -----
logfg05  sys  sds1     0  os----- 00000000    0    0  -----
logfg06  sys  sds1     0  os----- 00000000    0    0  -----
```

〔説明〕

- Status の 3 カラム目が u になっている logfg01, logfg02, および logfg03 が「アンロード待ち状態」のファイルです。
- Status の 2 カラム目が c になっている logfg04 が「現用」のファイルです。

(3) アンロードログファイルを格納するディレクトリを作成します

アンロードログファイルを格納するディレクトリを mkdir コマンドで作成します。

\$HOME/HirDB/pdlogunld ディレクトリを作成します。

(4) pdlogunld コマンドで logfg01, logfg02, および logfg03 のシステムログをアンロードします

```
$ pdlogunld -d sys -g logfg01 -o $HOME/HirDB/pdlogunld/unlog01
$ pdlogunld -d sys -g logfg02 -o $HOME/HirDB/pdlogunld/unlog02
$ pdlogunld -d sys -g logfg03 -o $HOME/HirDB/pdlogunld/unlog03
```

〔説明〕

- g :
アンロードするシステムログファイル (logfg01, logfg02, および logfg03) を指定します。
- o :
アンロードログファイルの名称を指定します。
(3) で作成した \$HOME/hirdb/pdlogunld ディレクトリ下に unlog01, unlog02, および unlog03 ファイルが作成されます。

(5) pdlogunld コマンドの実行結果が表示されます

```

2840 18:07:36 SDSU          KFPS01271-I Unloading completed. generation number=1,
start block number=1, end block number=6

2843 18:07:43 SDSU          KFPS01271-I Unloading completed. generation number=1,
start block number=1, end block number=40

2846 18:07:48 SDSU          KFPS01271-I Unloading completed. generation number=1,
start block number=1, end block number=8

```

〔説明〕

システムログファイルのアンロードが完了しました。

(6) pdlogls コマンドでシステムログファイルの状態をチェックします

```
$ pdlogls -d sys
```

(7) pdlogls コマンドの実行結果が表示されます

```

HOSTNAME : HIRLIN_SV(180806)
Group    Type Server  Gen No.  Status  Run ID      Block No.  Ex-Status
logfg01  sys  sds1      1        os----u  4019e4a7    1          6  -----
logfg02  sys  sds1      1        os----u  401f5271    1         40  -----
logfg03  sys  sds1      1        os----u  401f60de    1          8  -----
logfg04  sys  sds1      2        oc-d--u  401f60de    9          9  -----
logfg05  sys  sds1      0        os----- 00000000    0          0  -----
logfg06  sys  sds1      0        os----- 00000000    0          0  -----

```

〔説明〕

logfg01, logfg02, および logfg03 の Status の 3 カラム目が u (アンロード待ち状態) から - (アンロード済み状態) に変わりました。

注

システムログをアンロードすると、システムログファイルの状態が「アンロード待ち状態」から「アンロード済み状態」に変わります。

4.4 バックアップを取得する

バックアップの取得について説明します。

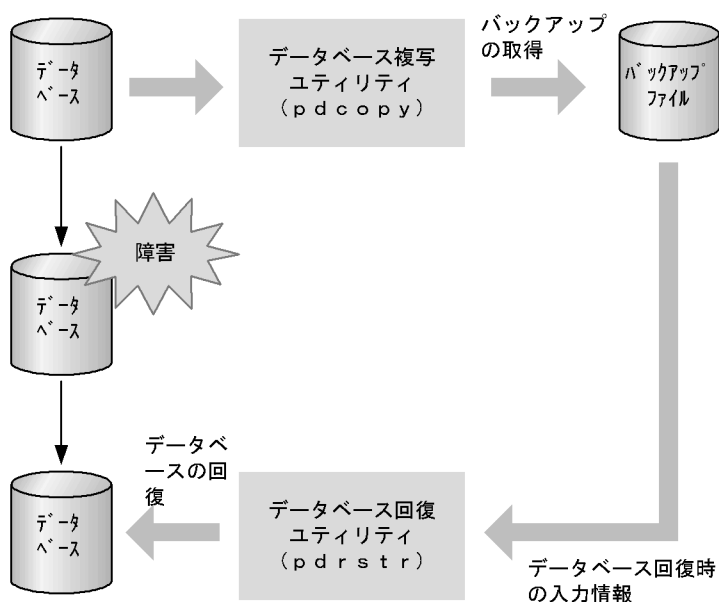
4.4.1 基礎知識

データベースの障害に備えて、定期的にデータベースのバックアップを取得する必要があります。バックアップは `pdcopy` コマンド（データベース複写ユーティリティ）で取得します。

（1）バックアップを取得する目的

ディスク障害などでデータベースを回復する必要が生じた場合、ここで取得したバックアップを `pdrstr` コマンド（データベース回復ユーティリティ）の入力情報にしてデータベースを回復します。データベース回復の概要を図 4-4 に示します。

図 4-4 データベース回復の概要



（2）バックアップの取得単位

バックアップは次に示す単位で取得できます。バックアップの取得単位は `pdcopy` コマンドのオプションで指定できます。バックアップの取得単位を表 4-4 に示します。

表 4-4 バックアップの取得単位

バックアップの取得単位	説明	pdcopy コマンドのオプションの指定
システム単位	全 RD エリアを対象としてバックアップを取得します。次に示す RD エリアのバックアップも取得されます。 <ul style="list-style-type: none"> • マスタディレクトリ用 RD エリア • データディレクトリ用 RD エリア • データディクショナリ用 RD エリア 	-a
RD エリア単位	RD エリアごとにバックアップを取得します。	-r RD エリア名

(3) バックアップ取得モード

pdcopy コマンドの -M オプションでバックアップ取得モードを選択します。バックアップ取得モードを表 4-5 に示します。

表 4-5 バックアップ取得モード

バックアップ取得モード (-M オプション の指定値)	モードの説明	データベースの 回復方法の違い
x	バックアップ取得中、バックアップ対象 RD エリアの参照および更新ができません。バックアップ取得対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。別のモードでバックアップすると、アクセスが多いことが原因でバックアップに時間がかかる場合に使用すると便利です。	バックアップだけでデータベースをバックアップ取得時点に回復できます。
r	バックアップ取得中、バックアップ取得対象の RD エリアの参照はできますが、更新はできません。システムログを使わないでデータベースの回復を行う運用の場合に便利です。	
s	バックアップ取得中、バックアップ取得対象 RD エリアの参照および更新ができます。RD エリアの参照業務および更新業務を続けながらバックアップを取得したい場合に便利です。	データベースを回復するときに、バックアップおよびバックアップ取得中のシステムログが必要になります。

注

pdcopy コマンドの処理結果リストに、RD エリアを回復するときに必要なシステムログファイルの名称および世代番号が出力されます。詳細については、「(4)pdcopy コマンドの処理結果リスト」を参照してください。

注意事項

HiRDB の稼働中に全 RD エリアのバックアップを取得する場合 (システム単位で

4. HiRDB を運用してみましょう

バックアップを取得する場合は、バックアップ取得モード (-M オプション) に *r* または *s* を指定してください。 *x* は指定できません。

理由：

x を指定する場合は、バックアップ取得対象 RD エリアを `pdhold` コマンドで閉塞かつクローズ状態にする必要があります。しかし、マスタディレクトリ用 RD エリアは閉塞かつクローズ状態にできません。このため、-M オプションに *x* を指定して、HiRDB の稼働中に全 RD エリアのバックアップを取得できません。

(4) pdcopy コマンドの処理結果リスト

`pdcopy` コマンドを実行すると処理結果リストが出力されます。処理結果リストの出力先は、`pdcopy` コマンドの `-p` オプションで指定します。`pdcopy` コマンドの処理結果リストの出力例を図 4-5 に示します。

図 4-5 `pdcopy` コマンドの処理結果リストの出力例

```
pdcopy (VV-RR) ***** DB COPY *****   YYYY-MM-DD hh:mm:ss   HDBS
-----
*** DB BACKUP INFORMATION LIST ***          BACKUPMODE : SHARED ..... 1
<<LOG FILE INFORMATION>>
  SERVER NAME   : sds01                      } ..... 2
  FILE NAME     : log2
  GENERATION NO : 1
<<BACKUP FILE INFORMATION>>
  FILE NAME     : host:/home/hirman/hirdb/pdcopy/backup01 } ..... 3
  FILE KIND     : u
  STARTED AT    : YYYY-MM-DD hh:mm:ss
  ENDED AT      : YYYY-MM-DD hh:mm:ss
<<RDAREA INFORMATION>>
  UNIT NAME     : unt1      SERVER NAME : sds01 } ..... 4
  RDAREA NAME   : MAST
  RDAREA ID     :          1 ATTRIBUTE : MASTERDIRECTORY PAGE SIZE : 4096
  STARTED AT    : YYYY-MM-DD hh:mm:ss
  ENDED AT      : YYYY-MM-DD hh:mm:ss
<FILE INFORMATION>
  1 /HiRDB/rfiles/master } ..... 5
  EXTENT COUNT : 1
  :
  :
```

〔説明〕

- バックアップ取得モードに関する情報です。
SHARED : バックアップ取得モードに *s* を指定
EXCLUSIVE : バックアップ取得モードに *x* または *r* を指定
- システムログファイルに関する情報です。

バックアップ取得モードに `s` を指定した場合に表示されます。

FILE NAME に表示されているシステムログファイルがデータベースの回復時に必要なシステムログファイルです。

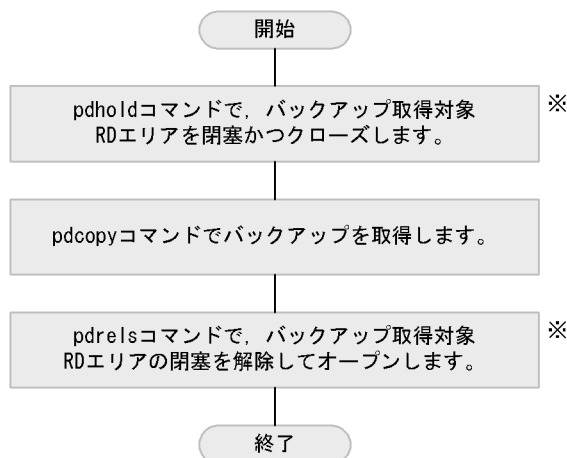
3. バックアップファイルに関する情報です。
4. バックアップ取得対象 RD エリアに関する情報です。
5. バックアップ取得対象 RD エリアのファイル構成に関する情報です。

4 および 5 の情報はバックアップを取得した RD エリアの数だけ表示されます。

(5) バックアップの取得手順

バックアップを取得する手順を図 4-6 に示します。

図 4-6 バックアップを取得する手順



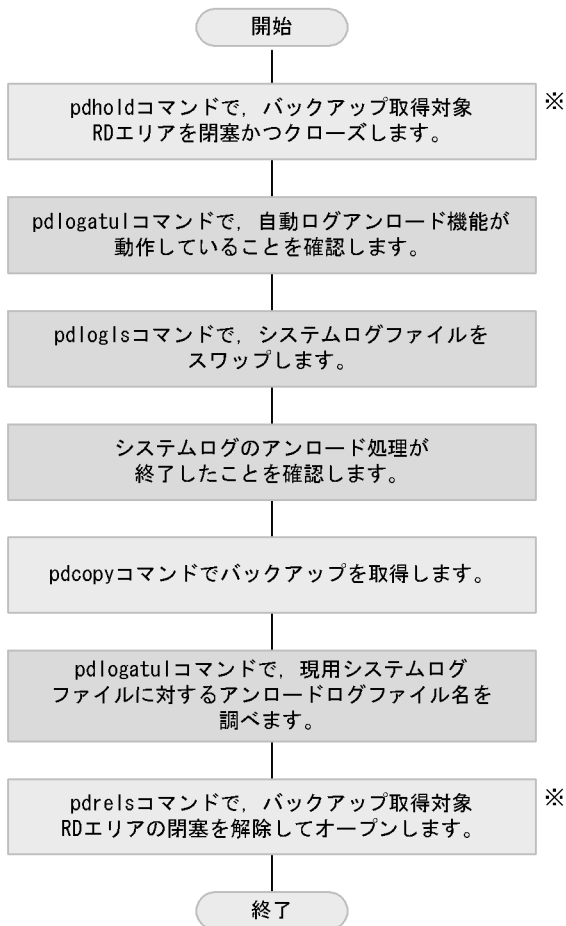
注※ バックアップ取得モードに `x` を指定する場合に必要な操作です。

(6) 自動ログアンロード機能を使用する場合のバックアップの取得手順

自動ログアンロード機能を使用する場合は、バックアップの取得手順が「(5) バックアップの取得手順」とは異なります。自動ログアンロード機能を使用する場合のバックアップの取得手順を図 4-7 に示します。

4. HIRDB を運用してみましょう

図 4-7 自動ログアンロード機能を使用する場合のバックアップを取得する手順



注※ バックアップ取得モードに x を指定する場合に必要な操作です。

自動ログアンロード機能を使用している場合は、システムログをスワップし、それが自動的にアンロードされるのを確認してからバックアップを取得します。

バックアップを取得したあとは、現用システムログに対するアンロードログを調べて、データベース回復時に使用するアンロードログをあらかじめ確認しておきます。

4.4.2 演習 1 (全 RD エリアのバックアップを取得する)

全 RD エリア (マスタディレクトリ用 RD エリア、データディレクトリ用 RD エリア、およびデータディクショナリ用 RD エリアを含む) のバックアップを取得します。バックアップ取得モードは r とします。

なお、ここで取得したバックアップは「4.5 データベースを回復する」の演習で使用するもので、保存しておいてください。

(1) バックアップファイルを格納するディレクトリを作成します

バックアップファイルを格納するディレクトリを `mkdir` コマンドで作成します。
`$HOME/HiRDB/pdcopy` ディレクトリを作成してください。

(2) 全 RD エリアのバックアップを pdcopy コマンドで取得します

全 RD エリアのバックアップを取得（システム全体のバックアップを取得）するため、
`-a` を指定してデータベース複写ユティリティを実行します。次のコマンドを実行してください。

```
$ pdcopy -m /HiRDB/rdfiles/master -a -M r
-b $HOME/HiRDB/pdcopy/backup01
-p $HOME/HiRDB/pdcopy/list01.txt
```

〔説明〕**-m :**

マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称 (`/HiRDB/rdfiles/master`) を指定します。

マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称は、システムジェネレータで作成しています。「2.4.2 システムジェネレータ (`pdgen`) を実行する」を参照してください。

-a :

全 RD エリアのバックアップを取得するオプションです。

-M :

バックアップ取得モードを指定します。

-b :

バックアップファイル名を指定します。

`$HOME/HiRDB/pdcopy` ディレクトリ下に `backup01` ファイルが作成されます。

-p :

`pdcopy` コマンドの処理結果リストの出力先を指定します。`$HOME/HiRDB/pdcopy` ディレクトリ下に `list01.txt` ファイルが作成されます。

(3) pdcopy コマンドの実行結果が表示されます

```
1330 15:24:03 SDSU _pd0copy KFPR00754-I Pdcopy started .....1
1331 15:24:07 SDSU 0bcpy00 KFPR26109-I Backup completed,
file=/home/hirman/hirdb/pdcopy/backup01 .....2
1330 15:24:04 SDSU _pd0copy KFPR26022-I Output result of pdcopy to
/home/hirman/hirdb/pdcopy/list01.txt .....3
1330 15:24:07 SDSU _pd0copy KFPR00756-I Pdcopy terminated,
return code=0 .....4
```

4. HIRDB を運用してみましょう

〔説明〕

1. バックアップの取得を開始しました。
2. バックアップを取得しました。
バックアップファイル名は、/home/hirman/HiRDB/pdcopy/backup01 です。
3. pdcopy コマンドの処理結果リストが /home/hirman/HiRDB/pdcopy/
list01.txt に出力されました。
4. pdcopy コマンドが正常終了しました。

(4) pdcopy コマンドが失敗した場合 (バックアップの取得に失敗した場合)

pdcopy コマンドが失敗すると、リターンコードに 0 が返ってきません。この場合は、画面および syslog に出力されたメッセージを参照して対策してください。

```
1216 11:47:35 unt1 _pd0copy KFPR00754-I Pdcopy started
1216 11:47:35 unt1 _pd0copy KFPR26111-E To copy Master
RDAREA with -M x option, pdstart should be executed
with -r option
1216 11:47:35 unt1 _pd0copy KFPR00756-I Pdcopy terminated,
return code=12
```

〔説明〕

バックアップ取得モード (-M オプション) に x を指定して、システム単位のバックアップを取得すると、このメッセージが表示されます。

4.4.3 演習 2 (RD エリア単位のバックアップを取得する)

RD エリア (RDDATA10) のバックアップをバックアップ取得モード x で取得します。

なお、ここで取得したバックアップは「4.5 データベースを回復する」の演習で使用するので、保存しておいてください。

(1) pdhold コマンドで RDDATA10 を閉塞かつクローズします

対象 RD エリアの参照および更新処理を禁止するため、RD エリアを閉塞かつクローズ状態にします。次のコマンドを実行してください。

```
$ pdhold -r RDDATA10 -c
```

(2) pdhold コマンドの実行結果が表示されます

```

2638 18:02:31 SDSU sds1      KFPH00130-I RDAREA held,
RDAREA = "RDDATA10"      .....1
2638 18:02:31 SDSU sds1      KFPH00130-I RDAREA closed,
RDAREA = "RDDATA10"      .....2
2638 18:02:31 SDSU sds1      KFPH00110-I pdhold command completed ...3

```

〔説明〕

1. RDDATA10 を閉塞しました。
2. RDDATA10 をクローズしました。
3. pdhold コマンドが完了しました。

(3) pdcopy コマンドで RDDATA10 のバックアップを取得します

対象 RD エリアのバックアップを取得します。次のコマンドを実行してください。

```

$ pdcopy -m /HiRDB/rdfiles/master -r RDDATA10 -M x
-b $HOME/HiRDB/pdcopy/backup02
-p $HOME/HiRDB/pdcopy/list02.txt

```

〔説明〕

- m :
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称 (/HiRDB/
rdfiles/master) を指定します。
- r :
バックアップを取得する RD エリアを指定します。
- M :
バックアップ取得モードを指定します。
- b :
バックアップファイル名を指定します。
\$HOME/HiRDB/pdcopy ディレクトリ下に backup02 ファイルが作成されます。
- p :
pdcopy コマンドの処理結果リストの出力先を指定します。\$HOME/HiRDB/
pdcopy ディレクトリ下に list02.txt ファイルが作成されます。

4. HIRDB を運用してみましょう

(4) pdcopy コマンドの実行結果が表示されます

```
2793 18:03:51 SDSU _pd0copy KFPR00754-I Pdcopy started .....1
2794 18:03:51 SDSU 0bcpy00 KFPR26109-I Backup completed,
file=/home/hirman/HiRDB/pdcopy/backup02 .....2
2793 18:03:51 SDSU _pd0copy KFPR26022-I Output result of pdcopy to
/home/hirman/HiRDB/pdcopy/list02.txt .....3
2793 18:03:51 SDSU _pd0copy KFPR00756-I Pdcopy terminated,
return code=0 .....4
```

[説明]

1. バックアップの取得を開始しました。
2. バックアップを取得しました。
バックアップファイル名は、/home/hirman/HiRDB/pdcopy/backup02 です。
3. pdcopy コマンドの処理結果リストが /home/hirman/HiRDB/pdcopy/
list02.txt に出力されました。
4. pdcopy コマンドが正常終了しました。

(5) pdrels コマンドで RDDATA10 の閉塞を解除してオープンします

参照および更新処理を行えるように、対象 RD エリアの閉塞を解除します。次のコマンドを実行してください。

```
$ pdrels -r RDDATA10 -o
```

(6) pdhold コマンドの実行結果が表示されます

```
2639 18:04:29 SDSU sds1 KFPH00130-I RDAREA released,
RDAREA = "RDDATA10" .....1
2639 18:04:29 SDSU sds1 KFPH00130-I RDAREA opened,
RDAREA = "RDDATA10" .....2
2639 18:04:29 SDSU sds1 KFPH00110-I pdrels command completed ...3
```

[説明]

1. RDDATA10 の閉塞を解除しました。
2. RDDATA10 をオープンしました。
3. pdrels コマンドが完了しました。

(7) pddbls コマンドで RDDATA10 の状態を確認します

閉塞が解除されたかどうかを確認します。次のコマンドを実行してください。

```
$ pddbls -r RDDATA10
```

(8) pddbils コマンドの実行結果が表示されます

STATE OF RDAREA	ID	STATUS	TYPE	OPNMODE
RDAREA	7	OPEN	USER	INITIAL
RDDATA10				

〔説明〕

STATUS が OPEN になっているので、閉塞が解除されてオープン状態になっています。

4.4.4 演習 3 (バックアップを DAT に取得する)

全 RD エリア (マスタディレクトリ用 RD エリア, データディレクトリ用 RD エリア, およびデータディクショナリ用 RD エリアを含む) のバックアップを DAT に取得します。バックアップ取得モードは, r とします。

なお, ここで取得したバックアップは「4.5 データベースを回復する」の演習で使用するので, 保存しておいてください。

! 注意事項

バックアップを DAT に取得するためには, HiRDB 管理者が DAT の書き込み権限を持っている必要があります。必要に応じて, DAT のデバイスファイルの権限を変更してください。

(1) バックアップファイルを格納する DAT を用意します

バックアップファイルを格納するための DAT を DAT ドライブにセットします。

なお, 一つの DAT にバックアップファイルは一つしか保存できません。バックアップファイルが存在する DAT にバックアップを取得すると, 上書きされるので注意してください。

(2) 全 RD エリアのバックアップを pdcopy コマンドで取得します

DAT にバックアップを取得するため, -b オプションに DAT のデバイス (/dev/st0) を指定してデータベース複写ユティリティを実行します。次のコマンドを実行してください。

```
$ pdcopy -m /HiRDB/rdfiles/master -a -M r -b /dev/st0
-p $HOME/HiRDB/pdcopy/list03.txt
```

〔説明〕

4. HiRDB を運用してみましょう

-m :

マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称 (/HiRDB/ rdfsfiles/master) を指定します。マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称は、pdinit コマンドの create rdarea 文でマスタディレクトリ用 RD エリアを定義したときに指定しています (file name オペランドに指定したファイル名称です)

-a :

全 RD エリアのバックアップを取得するオプションです。

-M :

バックアップ取得モードを指定します。

-b :

DAT のデバイスファイル名を示す /dev/st0 を指定します。デバイスファイル名は、ご使用のデバイスファイルにあわせて変更してください。
DAT にバックアップを取得する場合、バックアップファイル名を指定できません。また、一つの DAT に複数のバックアップを取得することはできません。

-p :

pdcopy コマンドの処理結果リストの出力先を指定します。\$HOME/HiRDB/ pdcopy ディレクトリ下に list03.txt ファイルが作成されます。

(3) pdcopy コマンドの実行結果が表示されます

```
13079 13:12:28 SDSU _pd0copy KFPR00754-I Pdcopy started .....1
13080 13:13:09 SDSU 0bcopy00 KFPR26109-I Backup completed,
file=/dev/st0 .....2
13079 13:12:28 SDSU _pd0copy KFPR26022-I Output result of pdcopy to
/home/hirman/HiRDB/pdcopy/list03.txt .....3
13079 13:13:09 SDSU _pd0copy KFPR00756-I Pdcopy terminated,
return code=0 .....4
```

〔説明〕

1. バックアップの取得を開始しました。
2. バックアップを取得しました。
バックアップファイル (デバイスファイル) 名は /dev/st0 です。
3. pdcopy コマンドの処理結果リストが /home/hirman/HiRDB/pdcopy に出力されました。
4. pdcopy コマンドが正常終了しました。

(4) pdcopy コマンドが失敗した場合 (バックアップの取得に失敗した場合)

pdcopy コマンドが失敗すると、リターンコードが 0 で返ってきません。この場合は、画

面および syslog に出力されたメッセージを参照して対策してください。

```
1616 11:54:16 unt1 _pd0copy KFPR00754-I Pdcopy started
384 11:55:17 unt1 Orcopy0 KFPR16003-E Open error occurred,
  file=/dev/st5
384 11:54:17 unt1 Orcopy0 KFPR16101-E File I/O error occurred,
  reason=Invalid-permission, func=open, errno=13, (utlfmkbh.c:0243)
1448 11:54:17 unt1 Obcpy0 KFPR16110-I Unable to get
  a backup file /dev/st5
1616 11:54:17 unt1 _pd0copy KFPR26022-I Output result of
  pdcopy to /home/hirman/HiRDB/pdcopy/list03.txt
1616 11:54:17 unt1 _pd0copy KFPR00756-I Pdcopy terminated,
  return code=12
```

〔説明〕

DAT ドライブの指定に存在しないドライブ /dev/st5 を指定すると、このメッセージが表示されます。

4.5 データベースを回復する

データベースの回復について説明します。

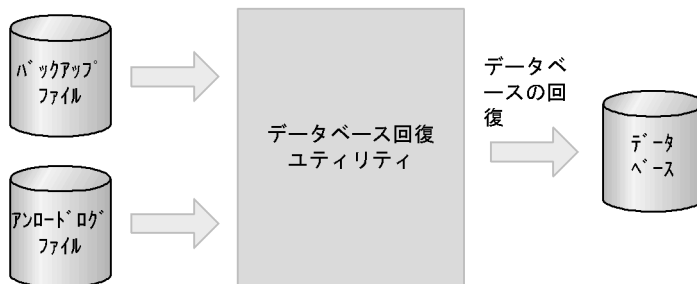
4.5.1 基礎知識

データベースに障害が発生した場合、事前取得したバックアップファイルからデータベースを回復します。データベースの回復は、pdrstr コマンド(データベース回復ユーティリティ)で行います。

(1) pdrstr コマンド (データベース回復ユーティリティ) でデータベースを回復します

ディスク障害など、データベースを回復する必要がある場合、pdrstr コマンドでデータベースを回復します。このとき、次に示す入力情報が必要になります。

- バックアップファイル
- システムログファイル, またはアンロードログファイル



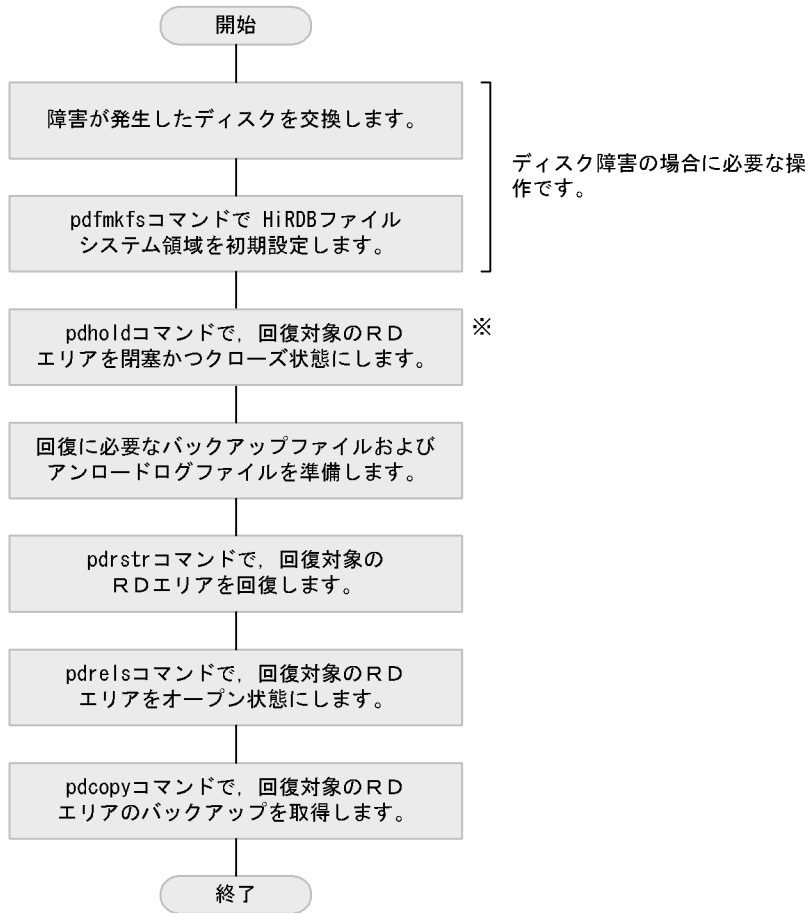
データベースは次に示す時点 (状態) に回復できます。

- バックアップ取得時点
バックアップ取得時点に回復する場合は、システムログやアンロードログファイルは必要ありません。バックアップファイルだけがが必要です。
- 最新の状態 (最新の同期点)
最新の状態に回復する場合は、バックアップファイルのほかに、バックアップ取得以降に出力されたシステムログ、またはそれに対応するアンロードログファイルが必要になります。

(2) データベースの回復手順

データベースを回復する手順を図 4-8 に示します。

図 4-8 データベースを回復する手順



(3) 自動ログアンロード機能を使用する場合のデータベースの回復手順

自動ログアンロード機能を使用する場合も、データベースの回復手順は「(2) データベースの回復手順」と同じです。ただし、自動ログアンロード機能を使用している場合、指定するアンロードログファイル名がわからなくならないように注意する必要があります。アンロードログの時系列リストを自分で作成するなどして、バックアップファイルと指定するアンロードログファイルの対応を管理することをお勧めします。

(4) 注意事項

データベースを回復する際の注意事項を次に示します。

4. HiRDB を運用してみましょう

(a) マスタディレクトリ用 RD エリアの回復

RD エリアを回復するとき、回復対象 RD エリアを `pdhold` コマンドで閉塞かつクローズ状態にする必要があります。しかし、マスタディレクトリ用 RD エリアは閉塞かつクローズ状態にできません。このため、マスタディレクトリ用 RD エリアを回復する場合は、いったん HiRDB を終了して、そのあと `pdstart -r` コマンドで HiRDB を開始する必要があります。

したがって、全 RD エリアを回復する場合はマスタディレクトリ用 RD エリアの回復も含まれるため、`pdstart -r` コマンドで HiRDB を開始する必要があります。

(b) システム用 RD エリアのバックアップ取得時点への回復

システム用 RD エリア（マスタディレクトリ用 RD エリア、データディレクトリ用 RD エリア、およびデータディクショナリ用 RD エリア）をバックアップ取得時点に回復する場合は、回復の前に次の作業をする必要があります。なお、システム用 RD エリアを最新の状態に回復する場合は、次の作業をする必要はありません。

- マスタディレクトリ用 RD エリアを構成する HiRDB ファイルの削除
- データディレクトリ用 RD エリアを構成する HiRDB ファイルの削除
- データディクショナリ用 RD エリアを構成する HiRDB ファイルの削除

これらの HiRDB ファイルは `pdfrm` コマンドで削除できます。

したがって、全 RD エリアをバックアップ取得時点に回復する場合はシステム用 RD エリアの回復も含まれるため、前記の HiRDB ファイルの削除作業をする必要があります。

(c) 最新の状態への回復

RD エリアを最新の状態に回復するには、バックアップおよびバックアップ取得以降に出力されたシステムログが必要です。したがって、現用のシステムログファイルに格納されているシステムログが必要になります。

システムログをアンロードする運用の場合は、`pdlogswap` コマンドで現用ファイルをいったんスワップしてください。そのあと、スワップしたシステムログファイルをアンロードしてください。

4.5.2 演習 1（全 RD エリアをバックアップ取得時点に回復する）

データベースを格納しているディスクに障害が発生しました。全 RD エリア（マスタディレクトリ用 RD エリア、データディレクトリ用 RD エリア、およびデータディクショナリ用 RD エリアを含む）をバックアップ取得時点の状態に回復します。

(1) ディスクを交換します

障害が発生したディスクを交換します。この演習では実際に交換はしません。交換した

と仮定して以降の作業を行います。

(2) pdfmkfs コマンドで HiRDB ファイルシステム領域を初期設定します

pdfmkfs コマンドで HiRDB ファイルシステム領域を初期設定します。pdfmkfs コマンドで指定するオプションは、RD エリア用の HiRDB ファイルシステム領域を前回作成したときと同じにしてください。システムジェネレータによって実行された pdfmkfs コマンドについては、\$PDDIR/conf/pdgen_cmd.[HiRDB サーバ名] のファイルに記録されています。

この演習の場合、これは「2.3.1(2) pdfmkfs コマンドの実行」で実行した、次のコマンドに相当します。

```
$ pdfmkfs -r -n 119 -l 400 -k SVR -e 100 -i /HiRDB/sysfiles
$ pdfmkfs -r -n 81 -l 100 -k SVR -e 100 -i /HiRDB/rdfiles
```

なお、この演習では実際に pdfmkfs コマンドを実行しません。実行したと仮定して以降の作業を行います。

(3) pdrstr コマンドの処理結果リストを格納するディレクトリを作成します

pdrstr コマンドの処理結果リストを格納するディレクトリを mkdir コマンドで作成します。\$HOME/HiRDB/pdrstr ディレクトリを作成します。

(4) RD エリアの回復に必要なファイルを準備します

RD エリアの回復に必要なファイル（データベース回復ユティリティの入力情報ファイル）は次のとおりです。

- システム単位で取得したバックアップファイル
「4.4 バックアップを取得する」の演習 1 で取得したバックアップファイルを使用します。バックアップファイル名は、\$HOME/HiRDB/pdcopy/backup01 です。

(5) pdstop コマンドで HiRDB を正常終了します

HiRDB が稼働中の場合、正常終了してください。障害が発生すると、HiRDB/ シングルサーバは停止しますが、pd_mode_conf オペランドを「AUTO」または「MANUAL1」に設定していると、停止したあとに自動開始することがあります。

```
$ pdstop
```

4. HiRDB を運用してみましょう

(6) pdfrm コマンドで HiRDB ファイルを削除します

全 RD エリアの回復なので、マスタディレクトリ用 RD エリアを構成する HiRDB ファイル、データディレクトリ用 RD エリアを構成する HiRDB ファイル、およびデータディクショナリ用 RD エリアを構成する HiRDB ファイルを削除する必要があります。次のコマンドを実行してください。

```
$ pdfrm /HiRDB/rdfiles/master .....1
$ pdfrm /HiRDB/rdfiles/d_dir .....2
$ pdfrm /HiRDB/rdfiles/ddic .....3
```

[説明]

1. マスタディレクトリ用 RD エリアを構成する HiRDB ファイルを削除します。
2. データディレクトリ用 RD エリアを構成する HiRDB ファイルを削除します。
3. データディクショナリ用 RD エリアを構成する HiRDB ファイルを削除します。

(7) pdstart -r コマンドで HiRDB を開始します

マスタディレクトリ用 RD エリアが使えない場合のデータベースの回復時には、`-r` オプションを指定して HiRDB を開始する必要があります。次のコマンドを実行してください。

```
$ pdstart -r
```

(8) pdrstr コマンドで全 RD エリアをバックアップ取得時点で回復します

「4.4 バックアップを取得する」の演習 1 で取得したバックアップファイルを指定して、データベース回復ユティリティを実行してください。

```
$ pdrstr -m /HiRDB/rdfiles/master -a
-b $HOME/HiRDB/pdcopy/backup01
-p $HOME/HiRDB/pdrstr/list01.txt
```

[説明]

- m :
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。
- a :
全 RD エリアの回復を指定します。

-b :

バックアップファイルの名称を指定します。「4.4 バックアップを取得する」の演習 1 で取得した全 RD エリアのバックアップファイル (\$HOME/hirdb/pdcopy/backup01) を指定します。

-p :

pdrstr コマンドの処理結果出力リストの出力先を指定します。\$HOME/hirdb/pdrstr ディレクトリ下に list01.txt ファイルが作成されます。

(9) pdrstr コマンドの実行結果が表示されます

```
2928 18:11:41 SDSU _pd0rstr KFPR00764-I Pdrstr started .....1
2928 18:11:53 SDSU _pd0rstr KFPR26222-I Output result of pdrstr
to /home/hirman/HiRDB/pdrstr/list01.txt .....2
2928 18:11:55 SDSU _pd0rstr KFPR00765-I Pdrstr terminated,
return code=0 .....3
```

〔説明〕

1. pdrstr コマンドの処理が開始しました。
2. 処理結果リストが /home/hirman/HiRDB/pdrstr/list01.txt に出力されました。
3. pdrstr コマンドの処理が正常終了しました。

(10) pdstop コマンドで HiRDB を正常終了します

HiRDB を通常の稼働状態 (-r 指定なし pdstart) にするため、一度 HiRDB を正常終了してください。

```
$ pdstop
```

(11) pdstart コマンドで HiRDB を正常開始します

HiRDB を正常開始してください。

```
$ pdstart
```

4.5.3 演習 2 (RD エリアを最新の状態に回復する)

ユーザ用 RD エリア (RDDATA10) を最新の状態に回復します。

(1) pdhold コマンドで RDDATA10 を閉塞、クローズ状態にします

回復する RD エリアを閉塞、クローズ状態にします。次のコマンドを実行してください。

4. HIRDB を運用してみましょう

回復する RD エリアが障害閉塞している場合は、pdclose コマンドでクローズしてください。

```
$ pdhold -r RDDATA10 -c
```

(2) pdhold コマンドの実行結果が表示されます

```
2991 18:13:23 SDSU sds1      KFPH00130-I RDAREA held, RDAREA = "RDDATA10"
2991 18:13:23 SDSU sds1      KFPH00130-I RDAREA closed, RDAREA = "RDDATA10"
2991 18:13:23 SDSU sds1      KFPH00110-I pdhold command completed
```

(3) 必要なバックアップファイルとアンロードログファイルを特定します

バックアップファイルには、「4.4 バックアップを取得する」の演習 2 で取得した RDDATA10 のバックアップを格納したバックアップファイルを使用します。バックアップファイル名は、\$HOME/hirdb/pdcopy/backup02 です。

さらに、このバックアップを取得したとき以降に出力されたシステムログを格納したアンロードログファイルが必要になります。バックアップを取得したとき以降に出力されたシステムログは、アンロード待ちのシステムログファイルおよび現用のシステムログファイルに格納されています。したがって、これらのシステムログファイルの内容をアンロードする必要があります。

留意事項

バックアップは、「4.4 バックアップを取得する」の演習 1 で取得した全 RD エリアのバックアップでもかまいません。全 RD エリアのバックアップからでも、特定の RD エリアを回復できます。

(4) pdlogls コマンドでシステムログファイルを調べます

現用のシステムログファイルとアンロード待ちファイルを調べます。次のコマンドを実行してください。

```
$ pdlogls -d sys
```


(5) pdlogls コマンドの実行結果が表示されます

```

HOSTNAME : HIRLIN_SV(181347)
Group    Type Server  Gen No.  Status  Run ID      Block No.  Ex-Status
logfg01  sys  sds1      1  oc-d--u  401f65f0    1    5  -----
logfg02  sys  sds1      1  os----u  401f5271    1   40  -----
logfg03  sys  sds1      1  os----u  401f60de    1    8  -----
logfg04  sys  sds1      2  osu---u  401f60de    9    a  -----
logfg05  sys  sds1      0  os----- 00000000    0    0  -----
logfg06  sys  sds1      0  os----- 00000000    0    0  -----

```

〔説明〕

Status の 2 カラム目が c になっている logfg01 が現用のファイルです。また、Status の 3 カラム目が u になっている logfg04 がアンロード待ちファイルです。

(6) pdlogswap コマンドで現用ファイル (logfg01) をスワップします

現用ファイルをスワップします。次のコマンドを実行してください。

```
$ pdlogswap -d sys
```

〔説明〕

RD エリアを最新の状態に回復するには、バックアップ取得以降に出力されたシステムログが必要です。したがって、現用ファイルに格納されているシステムログが必要になります。現用ファイルはアンロードできないため、pdlogswap コマンドで現用ファイルをいったんスワップします。

(7) pdlogunld コマンドで logfg04 および logfg01 の内容をアンロードします

pdlogunld コマンドで logfg04 および logfg01 の内容をアンロードします。次の例では、実行するコマンドをまとめて記述しています。実際に実行するときは、1 コマンドずつ実行してください。

```

sh-2.05$ pdlogunld -d sys -g logfg04 -o $HOME/HiRDB/pdlogunld/unlog04
sh-2.05$ pdlogunld -d sys -g logfg01 -o $HOME/HiRDB/pdlogunld/unlog05

```

4. HiRDB を運用してみましょう

(8) pdlogunld コマンドの実行結果が表示されます

```
3048 18:15:05 SDSU          KFPS01271-I Unloading completed.
generation number=2, start block number=9, end block number=a
3053 18:15:31 SDSU          KFPS01271-I Unloading completed.
generation number=1, start block number=1, end block number=6
```

(9) pdrstr コマンドで RDDATA10 を回復します

バックアップファイルとバックアップ取得以降の情報が格納されたアンロードログファイルを指定して、RD エリアを回復します。次のコマンドを実行してください。

```
$ pdrstr -m /HiRDB/rdfiles/master -r RDDATA10
-b $HOME/HiRDB/pdcopy/backup02
-l $HOME/HiRDB/pdlogunld/unlog01
  $HOME/HiRDB/pdlogunld/unlog02
  $HOME/HiRDB/pdlogunld/unlog03
  $HOME/HiRDB/pdlogunld/unlog04
  $HOME/HiRDB/pdlogunld/unlog05
-w /tmp -p $HOME/HiRDB/pdrstr/list02.txt
```

[説明]

- m :
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称 (/HiRDB/rdfiles/master) を指定します。
- r :
回復対象の RD エリアを指定します。
- b :
バックアップファイル名を指定します。「4.4 バックアップを取得する」の演習 2 で取得した RDDATA10 のバックアップファイル (\$HOME/HiRDB/pdcopy/backup02) を指定します。
- l :
アンロードログファイル名を指定します。-b で指定したバックアップファイルを取得したあとに取得したシステムログファイルのアンロードログファイルを古い順に指定します。
- w :
ソート用ワークディレクトリの名称を指定します。
- p :
pdrstr コマンドの処理結果リストの出力先を指定します。\$HOME/HiRDB/pdrstr ディレクトリ下に list02.txt ファイルが作成されます。

(10) pdrstr コマンドの実行結果が表示されます

```
3085 18:19:46 SDSU _pd0rstr KFPR00764-I Pdrstr started
3085 18:19:46 SDSU _pd0rstr KFPR26222-I Output result of
pdrstr to /home/hirman/HiRDB/pdrstr/list02.txt
3085 18:19:46 SDSU _pd0rstr KFPR00765-I Pdrstr terminated,
return code=0
```

〔説明〕

リターンコードに 0 が返されたので、RDDATA10 の回復に成功しました。

(11) pdcopy コマンドで RDDATA10 のバックアップを取得します

データベース回復直後の最新の状態で、バックアップを取得します。次のコマンドを実行してください。

```
$ pdcopy -m /HiRDB/rdfiles/master -r RDDATA10 -M x
-b $HOME/HiRDB/pdcopy/backup03
-p $HOME/HiRDB/pdcopy/list03.txt
```

〔説明〕

-m :

マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称 (/HiRDB/rdfiles/master) を指定します。

-r :

バックアップを取得する RD エリアを指定します。

-M :

バックアップ取得モードを指定します。

-b :

バックアップファイル名を指定します。
\$HOME/HiRDB/pdcopy ディレクトリ下に backup03 ファイルが作成されます。

-p :

pdcopy コマンドの処理結果リストの出力先を指定します。\$HOME/HiRDB/pdcopy ディレクトリ下に list03.txt ファイルが作成されます。

4. HiRDB を運用してみましょう

(12) pdcopy コマンドの実行結果が表示されます

```
2672 18:20:48 unt1 _pd0copy KFPR00754-I Pdcopy started
2104 18:20:49 unt1 0bcpy00 KFPR26109-I Backup completed,
file=/home/hirman/HiRDB/pdcopy/backup03
2672 18:20:49 unt1 _pd0copy KFPR26022-I Output result of
pdcopy to /home/hirman/HiRDB/pcopy/list03.txt
2672 18:20:49 unt1 _pd0copy KFPR00756-I Pdcopy terminated,
return code=0
```

〔説明〕

バックアップが取得されました。

(13) pdrels コマンドで RDDATA10 の閉塞を解除してオープンします

回復処理が終了したので、RD エリアの閉塞を解除します。次のコマンドを実行してください。

```
$ pdrels -r RDDATA10 -o
```

(14) pdrels コマンドの実行結果が表示されます

```
2991 18:22:13 SDSU sds1 KFPH00130-I RDAREA released, RDAREA = "RDDATA10"
2991 18:22:13 SDSU sds1 KFPH00130-I RDAREA opened, RDAREA = "RDDATA10"
2991 18:22:13 SDSU sds1 KFPH00110-I pdrels command completed
```

4.5.4 演習 3 (DAT のバックアップを使用して回復する)

データベースを格納しているディスクに障害が発生しました。DAT に取得したバックアップを使用して全 RD エリアを回復します。

(1) ディスクを交換します

障害が発生したディスクを交換します。この演習では実際に交換はしません。交換したと仮定して以降の作業を行います。

(2) pdfmkfs コマンドで HiRDB ファイルシステム領域を初期設定します

pdfmkfs コマンドで HiRDB ファイルシステム領域を初期設定します。pdfmkfs コマンドで指定するオプションは、RD エリア用の HiRDB ファイルシステム領域を前回作成したときと同じにしてください。システムジェネレータによって実行された pdfmkfs コマンドについては、\$PDDIR/conf/pdgen_cmd.[HiRDB サーバ名] のファイルに記録されています。

この演習の場合、これは「2.3.1(2) pdfmkfs コマンドの実行」で実行した、次のコマンド

に相当します。

```
$ pdfmkfs -r -n 119 -l 400 -k SVR -e 100 -i /HiRDB/sysfiles
$ pdfmkfs -r -n 81 -l 100 -k SVR -e 100 -i /HiRDB/rdfiles
```

なお、この演習では実際に pdfmkfs コマンドを実行しません。実行したと仮定して以降の作業を行います。

(3) RD エリアの回復に必要なファイルを準備します

RD エリアの回復に必要なファイル（データベース回復ユティリティの入力情報ファイル）は次のとおりです。

- ・システム単位で取得したバックアップファイル
「4.4 バックアップを取得する」の演習 3 で DAT に取得したバックアップファイルを使用します。DAT に取得したバックアップファイルに対してファイル名を指定することはできません。ファイルの指定には、DAT のデバイスファイル名を示す /dev/st0 を使用します。

(4) pdstop コマンドで HiRDB を正常終了します

HiRDB が稼働中の場合、正常終了してください。障害が発生すると、HiRDB/ シングルサーバは停止しますが、pd_mode_conf オペランドを「AUTO」または「MANUAL1」に設定していると、停止したあとに自動開始することがあります。

```
$ pdstop
```

(5) pdfrm コマンドで HiRDB ファイルを削除します

全 RD エリアの回復なので、マスタディレクトリ用 RD エリアを構成する HiRDB ファイル、データディレクトリ用 RD エリアを構成する HiRDB ファイル、およびデータディクショナリ用 RD エリアを構成する HiRDB ファイルを削除する必要があります。次のコマンドを実行してください。

```
$ pdfrm /HiRDB/rdfiles/master .....1
$ pdfrm /HiRDB/rdfiles/ddir .....2
$ pdfrm /HiRDB/rdfiles/ddic .....3
```

〔説明〕

1. マスタディレクトリ用 RD エリアを構成する HiRDB ファイルを削除します。

4. HiRDB を運用してみましょう

2. データディレクトリ用 RD エリアを構成する HiRDB ファイルを削除します。
3. データディクショナリ用 RD エリアを構成する HiRDB ファイルを削除します。

(6) pdstart -r コマンドで HiRDB を開始します

マスタディレクトリ用 RD エリアが使えない場合のデータベースの回復時には、`-r` オプションを指定して HiRDB を開始する必要があります。次のコマンドを実行してください

```
$ pdstart -r
```

(7) バックアップファイルが格納されている DAT を用意します

「4.4 バックアップを取得する」の演習 3 でバックアップファイルを格納した DAT を DAT ドライブにセットします。

(8) pdrstr コマンドで全 RD エリアをバックアップ取得時点に回復します

DAT デバイスファイル名を指定して、データベース回復ユティリティを実行してください。

```
$ pdrstr -m /HiRDB/rdfiles/master -a -b /dev/st0  
-p $HOME/HiRDB/pdrstr/list03.txt
```

[説明]

- m :
マスタディレクトリ用 RD エリアの先頭の HiRDB ファイル名称を指定します。
- a :
全 RD エリアの回復を指定します。
- b :
バックアップファイルの名称を指定します。「4.4 バックアップを取得する」の演習 3 で全 RD エリアのバックアップを取得した DAT のデバイスファイル名 (`/dev/st0`) を指定します。
- p :
pdrstr コマンドの処理結果出力リストの出力先を指定します。`$HOME/HiRDB/pdrstr` ディレクトリ下に `list03.txt` ファイルが作成されます。

(9) pdrstr コマンドの実行結果が表示されます

```

13424 16:07:21 SDSU _pd0rstr KFPR00764-I Pdrstr started .....1
13424 16:07:52 SDSU _pd0rstr KFPR26222-I Output result of
pdrstr to /home/hirm/hirdb/pdrstr/list03.txt .....2
13424 16:07:52 SDSU _pd0rstr KFPR00765-I Pdrstr terminated,
return code=0 .....3

```

〔説明〕

1. pdrstr コマンドの処理が開始しました。
2. 処理結果リストが /home/hirm/HiRDB/pdrstr/list03.txt に出力されました。
3. pdrstr コマンドの処理が正常終了しました。

(10) pdstop コマンドで HiRDB を正常終了します

HiRDB を通常稼働状態 (-r 指定なし pdstart) にするため、一度 HiRDB を正常終了してください。

```
$ pdstop
```

(11) pdstart コマンドで HiRDB を正常開始します

HiRDB を正常開始してください。

```
$ pdstart
```

4.6 表を再編成する

表の再編成について説明します。

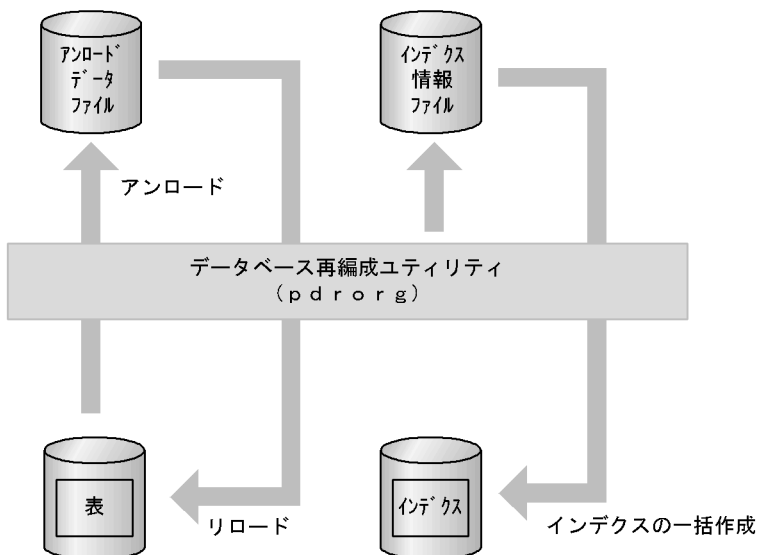
4.6.1 基礎知識

データの検索性能が悪くならないように、表は定期的に再編成する必要があります。表の再編成は `pdrorg` コマンド（データベース再編成ユーティリティ）で行います。

（1）表の再編成とは

データの追加および削除を繰り返すと、データの格納効率が悪くなり、データを検索するときの性能が低下します。これを防ぐために、定期的に `pdrorg` コマンド（データベース再編成ユーティリティ）で表の再編成を実施してください。表の再編成を図 4-9 に示します。

図 4-9 表の再編成



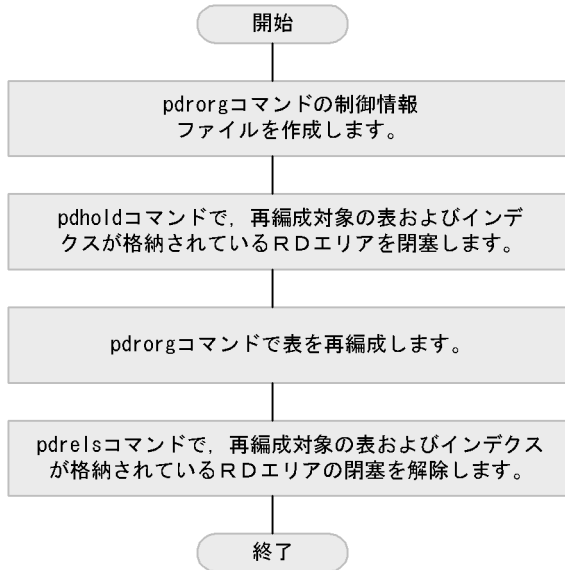
〔説明〕

- 表データを取り出し、専用のファイルに保存します。これを表データのアンロードといいます。また、専用のファイルをアンロードデータファイルといいます。そのあと、表にデータを格納し直します。これを表データのリロードといいます。これら全体の処理を表の再編成といいます。
- 表にインデクスが定義されていると、データをリロードするときにインデクス情報ファイルにインデクス情報が出力されます。その情報を基にして HiRDB がインデクスを一括作成します。これによって、インデクスも再編成されます。

(2) 表の再編成手順

表を再編成する手順を図 4-10 に示します。

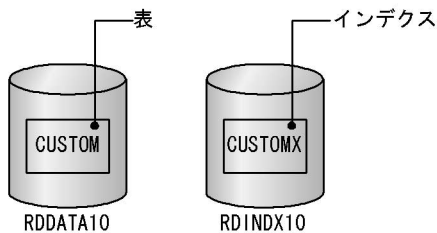
図 4-10 表を再編成する手順



4.6.2 演習（表を再編成する）

CUSTOM 表を再編成します。

- CUSTOM 表にはインデクス（CUSTOMX）が定義されています。
- CUSTOM 表はユーザ用 RD エリア（RDDATA10）に格納されています。
- インデクス（CUSTOMX）はユーザ用 RD エリア（RDINDX10）に格納されていま
す。



(1) 表の再編成時に必要なディレクトリを準備します

次に示すディレクトリが表の再編成時に必要になります。

- アンロードデータファイルを作成するディレクトリ

4. HiRDB を運用してみましょう

- インデクス情報ファイルを作成するディレクトリ

アンロードデータファイルを作成するディレクトリとして、\$HOME/HiRDB/pdrorg を作成してください。さらに、インデクス情報ファイルを作成するディレクトリとして、\$HOME/HiRDB/idxdir を作成してください。

(2) pdrorg コマンドの制御情報ファイルを作成します

制御情報ファイルをテキスト形式のファイルに作成します。この演習では、次に示す内容のファイル \$HOME/HiRDB/pdrorgseigy01.txt を作成してください。

```
unload /home/hirman/HiRDB/pdrorg/unld01 .....1
idxwork /home/hirman/HiRDB/idxdir .....2
sort /tmp .....3
```

[説明]

1. アンロードデータファイル名を指定します。
/home/hirman/HiRDB/pdrorg ディレクトリ下に unld01 ファイルが作成されます。/home/hirman/HiRDB/pdrorg は (1) で作成したディレクトリです。
2. インデクスの一括作成時に使用する作業用ワークディレクトリの名称を指定します。ここで指定したディレクトリ下にインデクス情報ファイルが作成されます。/home/hirman/irdb/idxdir は (1) で作成したディレクトリです。
3. ソート用ワークディレクトリの名称を指定します。

(3) pdhold コマンドで RDDATA10 および RDINDX10 を閉塞します

再編成する表を含む RD エリアを閉塞状態にします。次のコマンドを実行してください。

```
$ pdhold -r RDDATA10,RDINDX10
```

(4) pdhold コマンドの実行結果が表示されます

```
2992 18:22:23 SDSU sds1      KFPH00130-I RDAREA held, RDAREA = "RDDATA10"
2992 18:22:23 SDSU sds1      KFPH00130-I RDAREA held, RDAREA = "RDINDX10"
2992 18:22:23 SDSU sds1      KFPH00110-I pdhold command completed
```

(5) pdrorg コマンドで CUSTOM 表を再編成します

ユーザ「MANAGER」で表を再編成します。次のコマンドを実行してください。

```
$ pdrorg -k rorg -t CUSTOM -u MANAGER $HOME/hirdb/pdrorg/seigy01.txt
```

〔説明〕

- k : 再編成をするので rorg を指定します。
- t : 再編成対象の表 (CUSTOM) を指定します。
- u : pdrorg コマンドを実行するユーザ名を指定します。

\$HOME/hirdb/pdrorg/seigy01.txt :

- (2) で作成した制御情報ファイルの名称を指定します。

(6) パスワードを入力します

パスワードの入力を求められたら「MANAGER」を入力してください。

```
Password:
```

(7) pdrorg コマンドの実行結果が表示されます

```

:
:
3223 18:38:24 SDSU sds1      KFPL00715-I Index load started at sds1,
index=MANAGER."CUSTOMX", RDAREA="RDINDEX10"
3223 18:38:24 SDSU sds1      KFPL00716-I Index load ended at sds1,
index=MANAGER."CUSTOMX", RDAREA="RDINDEX10", return code=0
3223 18:38:24 SDSU sds1      KFPL00720-I Index file deleted, server=sds1,
file=/home/hirman/HiRDB/idxdir/INDEX-CUSTOMX-RDINDEX10-filenQN8QW
3226 18:38:24 SDSU _pd0rorg KFPL00719-I Pdrorg terminated,
return code=0

```

〔説明〕

pdrorg コマンドが正常終了しました。表の再編成が完了しました。

(8) pdrels コマンドで RDDATA10 および RDINDEX10 の閉塞を解除します

再編成が終わったので、表を含む RD エリアを閉塞を解除します。次のコマンドを実行してください。

```
$ pdrels -r RDDATA10,RDINDEX10
```

(9) pdrels コマンドの実行結果が表示されます

```

2990 18:39:03 SDSU sds1      KFPH00130-I RDAREA released, RDAREA = "RDDATA10"
2990 18:39:03 SDSU sds1      KFPH00130-I RDAREA released, RDAREA = "RDINDEX10"
2990 18:39:03 SDSU sds1      KFPH00110-I pdrels command completed

```

4.7 RD エリアを拡張する

RD エリアの拡張について説明します。

4.7.1 基礎知識

データを格納していくうちに、セットアップ時に用意した RD エリアの容量が足りなくなることがあります。このような場合、RD エリアを拡張することでデータを格納する領域を増やすことができます。RD エリアの拡張は、pdmod コマンド（データベース構成変更ユティリティ）で行います。

(1) pdmod コマンド（データベース構成変更ユティリティ）で RD エリアを拡張します

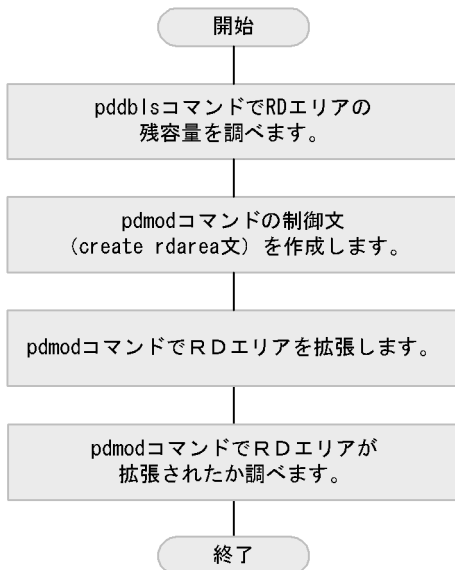
表にデータを追加していくと、RD エリアの残容量が少なくなっていくます。残容量がなくなったら、pdmod コマンドの expand rdarea 文で RD エリアを拡張します。

なお、pdinit コマンドの create rdarea 文でオペランド extension を指定すると、RD エリアの自動増分を設定することができます。自動増分とは、RD エリアの容量が不足したとき自動的に RD エリアを拡張する機能です。詳細については、マニュアル「HiRDB Version 7 コマンドリファレンス（UNIX(R) 用）」の pdinit の個所を参照してください。

(2) RD エリアの拡張手順

RD エリアを拡張する手順を図 4-11 に示します。

図 4-11 RD エリアを拡張する手順



4.7.2 演習 (RD エリアを拡張する)

ユーザ用 RD エリア (RDDATA10) を拡張します。

(1) pddbpls コマンドで RDDATA10 の残容量を調べます

拡張する前に RD エリアの残容量を調べます。次のコマンドを実行してください。

```
$ pddbpls -r RDDATA10 -a
```

(2) pddbpls コマンドの実行結果が表示されます

```
STATE OF RDAREA
RDAREA      ID      STATUS      TYPE      OPENMODE
RDDATA10    7       OPEN        USER      INITIAL
  SEGMENT 75 / 77
```

〔説明〕

未使用セグメント数および全セグメント数が SEGMENT に表示されます。未使用セグメント数が RD エリアの残容量となります。

この例では、未使用セグメント数が 75 で、全セグメント数が 77 となっています。未使用セグメント数が残り少なくなったら、RD エリアを拡張します。現在、未使用セグメント数が多いので拡張する必要はありませんが、演習のため RD エリアを拡張します。

なお、pdinit コマンドの create rdarea 文でオペランド extension を指定することで、未使用セグメント数が少なくなったとき自動的に RD エリアを拡張するように設定することができます。pdinit コマンドの詳細については、マニュアル「HiRDB Version 7 コマンドリファレンス (UNIX(R) 用)」を参照してください。

(3) pdmod コマンドの制御文を格納するディレクトリを作成します

pdmod コマンドの制御文ファイルを格納するディレクトリを mkdir コマンドで作成します。\$HOME/HiRDB/pdmod ディレクトリを作成してください。

(4) pdmod コマンドの制御文ファイルを作成します

pdmod コマンドの expand rdarea 文で RD エリアを拡張します。この演習では、次に示す内容の制御文ファイル \$HOME/hirdb/pdmod/expand.txt を作成してください。

```
expand rdarea RDDATA10 .....1
file name "/HiRDB/rdfiles/rddata10a" .....2
initial 20 segments; .....3
```

4. HiRDB を運用してみましょう

〔説明〕

1. 拡張する RD エリア (RDDATA10) の名称を指定します。
2. 追加する HiRDB ファイルの名称を指定します。
/HiRDB/rdfiles ディレクトリ下に rddata10a ファイルが作成されます。
3. 追加する HiRDB ファイルのセグメント数を指定します。

(5) pdmod コマンドで RD エリアを拡張します

作成した制御文ファイルを入力にして、データベース構成変更ユティリティ (pdmod) を実行します。次のコマンドを実行してください。

```
$ pdmod -a $HOME/HiRDB/pdmod/expand.txt
```

〔説明〕

-a : (4) で作成した制御文ファイルの名称を指定します。

(6) pdmod コマンドの実行結果が表示されます

```
KFPX14250-I Processing of expand rdarea statement ended  
  return code=0, RDAREA name="RDDATA10"  
KFPX24213-I DB modification for ended  
  return code=0 at 18:40:49 on 2004/02/03  
KFPX24200-I DB modification ended, return code=0
```

〔説明〕

リターンコードに 0 が返されたので、pdmod コマンドが正常終了しました。

pdmod コマンドが失敗すると (RD エリアの拡張に失敗すると)

pdmod コマンドが失敗すると、リターンコードに 0 が返ってきません。この場合は、syslog に出力されたメッセージを参照して対策してください。次の例は、指定した RDAREA 名に誤りがある場合に出力されたメッセージです。

```
KFPX24217-E HiRDB file "create" failed, return code=-1535,  
  HiRDB file name="/HiRDB/rdfiles/rddata11a"  
KFPX14250-I Processing of expand rdarea statement ended return code=8,  
  RDAREA name="RDDATA10"  
KFPX24213-I DB modification for ended return code=12  
  at 15:47:56 on 2004/01/29  
KFPX24200-I DB modification ended, return code=12
```

(7) pddbls コマンドで、RDDATA10 が拡張されたことを確認します

結果を確認するため、RDDATA10 の状態を表示します。次のコマンドを実行してください。

```
$ pddbls -r RDDATA10 -a
```

(8) pddbls コマンドの実行結果が表示されます

```
STATE OF RDAREA
RDAREA      ID      STATUS      TYPE      OPNMODE
RDDATA10    7      OPEN        USER      INITIAL
  SEGMENT 95 / 97
```

〔説明〕

全セグメント数が 77 から 97 に変わっています。それに伴い未使用セグメント数も 75 から 95 に変わっています。したがって、RDDATA10 が拡張されています。

4.8 RD エリアを追加する

RD エリアの追加について説明します。

4.8.1 基礎知識

運用を続けているうちに、新しい表を定義する必要がある出てくることがあります。このような場合、RD エリアを追加することで新しい表を定義する領域を確保します。RD エリアの追加は、`pdmod` コマンド（データベース構成変更ユティリティ）で行います。

(1) `pdmod` コマンド（データベース構成変更ユティリティ）で RD エリアを追加します

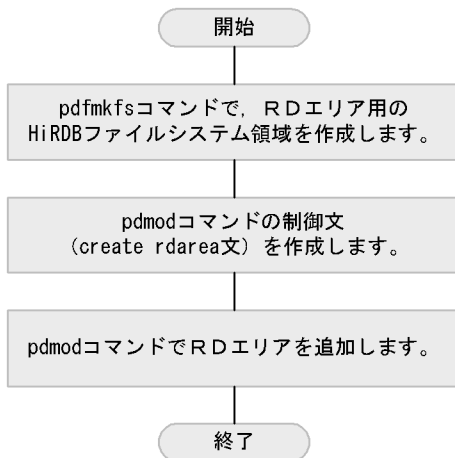
`pdmod` コマンドの `create rdarea` 文で、RD エリアを追加します。新規の表を作成する場合などに RD エリアを追加します。

なお、追加した RD エリアを使用するには、グローバルバッファを割り当てる必要があります。したがって、定義されているグローバルバッファを `pdbufls` コマンドで調べる必要があります。

(2) RD エリアの追加手順

RD エリアを追加する手順を図 4-12 に示します。

図 4-12 RD エリアを追加する手順



4.8.2 演習（RD エリアを追加する）

ユーザ用 RD エリア（RDAREA01）を作成して、そこに社員表（表名：EMP）を作成します。

(1) HiRDB ファイルシステム領域を作成するディレクトリを作成します

RD エリア用の HiRDB ファイルシステム領域を作成するディレクトリを `mkdir` コマンドで作成します。`$HOME/HiRDB/rdarea` ディレクトリを作成してください。

(2) `pdfmkfs` コマンドで、HiRDB ファイルシステム領域を作成します

追加する RD エリア用の HiRDB ファイルシステム領域を作成します。次のコマンドを実行してください。

```
$ pdfmkfs -n 20 -l 5 -k DB -i $HOME/HiRDB/rdarea/area01
```

〔説明〕

RD エリア用の HiRDB ファイルシステム領域を `$HOME/HiRDB/rdarea/area01` に作成します。

`-n` :

HiRDB ファイルシステム領域の大きさを MB 単位で指定します。この演習では 20MB の大きさの HiRDB ファイルシステム領域を作成します。

`-l` :

HiRDB ファイルシステム領域に作成する HiRDB ファイル数の最大値を指定します。

`-k` :

RD エリア用の HiRDB ファイルシステム領域を作成するので、DB を指定します。

`-i` :

HiRDB ファイルシステム領域の全領域を初期設定することを指定します。

`$HOME/HiRDB/rdarea/area01` :

作成する HiRDB ファイルシステム領域の名称を指定します。`$HOME/HiRDB/rdarea` ディレクトリ下に HiRDB ファイルシステム領域 `area01` が作成されます。

`pdfmkfs` コマンドが成功すると

メッセージは何も表示されません。

`pdfmkfs` コマンドが失敗すると

メッセージが表示されます。このメッセージを参照して対策してください。また、エラーの内容によっては `syslog` にメッセージが出力されています。そのメッセージも参照して対策してください。

4. HiRDB を運用してみましょう

(3) pdfstatfs コマンドで、HiRDB ファイルシステム領域が作成されたかを確認します

HiRDB ファイルシステム領域が作成されたかを確認します。次のコマンドを実行してください。

```
$ pdfstatfs $HOME/HiRDB/rdarea/area01
```

(4) pdfstatfs コマンドの実行結果が表示されます

```
user area capacity          20472 [kB]      .....1
remain user area capacity  20472 [kB]      .....2
available file size        20472 [kB]      .....3
available file count        5                .....4
current file count          0                .....5
remain file count           5                .....6
free area count             0                .....7
available expand count      0                .....8
current expand count        0                .....9
initialize area kind        DB                .....10
initialize user id          hirman          .....11
initialize time             Tue Feb  3 18:47:32 2004 .....12
```

[説明]

1. HiRDB ファイルシステム領域の大きさです。
2. 未使用領域の大きさです。
3. 1 ファイルとして確保できる大きさの最大値です。
4. この領域中に作成できる HiRDB ファイルの上限数です。pdfmkfs コマンドの `-l` オプションで指定した値です。
5. この領域中に作成した HiRDB ファイルの数です。
6. この領域中に作成できる HiRDB ファイルの数です (あと幾つ作成できるかです)。
7. 不連続な空き領域の総数です。
8. この領域の増分回数の上限值です。
9. この領域の増分回数合計値です。
10. この領域の種別です。pdfmkfs コマンドの `-k` オプションで指定した値です。
11. この領域を初期化したユーザのログイン名称です。
12. この領域を初期化した時刻です。この例では、2004 年 2 月 3 日 火曜日 18 時 47 分 32 秒に領域が初期化されています。

補足事項 Thu Jan 29 13:52:30 2004

`$HOME/HiRDB/rdarea` ディレクトリ下に `area01` ファイルが作成されています。確認してください。

(5) pdbufls コマンドで、定義されているグローバルバッファを調べます

追加する RD エリアに割り当てられるグローバルバッファを調べます。次のコマンドを実行してください。

```
$ pdbufls -k def
```

〔説明〕

-k def : グローバルバッファの定義情報を表示する指定です。

(6) pdbufls コマンドの実行結果が表示されます

```
DEFINE OF GLOBAL BUFFER
EDIT TIME 2004-02-03 18:48:22
BUFFNAME      SVID      TYPE  SIZE  NUM  WRATIO  RDAREA/INDEX NAME
                PRMAX PRNUM  CSIZE  MAPS
RDINDX10      sds1      R     4k    700   20      "RDINDX10"
                0    32   ***** **
RDDATA10      sds1      R     4k    700   20      "RDDATA10"
                0    32   ***** **
RLOB1         sds1      B     8k    200   0       "RLOB1"
                0    0    ***** **
RLOB2         sds1      B     8k    500   0       "RLOB2"
                0    0    ***** **
bp01          sds1      O     8k    350   20
                0    32   ***** **
```

〔説明〕

定義されているグローバルバッファは RD エリア用 (TYPE=R) の RDINDX , RDDATA10 , 10 , LOB 用 (TYPE=B) の RLOB1 , RLOB2 , およびその他の RD エリア用 (TYPE=O) の bp01 です。

(7) pdmod コマンドの制御文ファイルを作成します

pdmod コマンドの create rdarea 文で RD エリアを作成します。

この演習では、次に示す内容の制御文ファイル \$HOME/HiRDBd/create.txt を作成してください。

```
create rdarea RDAREA01 .....1
globalbuffer RDDATA10 .....2
for user used by PUBLIC .....3
page 4096 characters .....4
storage control segment 10 pages .....5
file name "/home/hirman/HiRDB/rdarea/area01/rdarea01" .....6
initial 10 segments; .....7
```

4. HiRDB を運用してみましょう

〔説明〕

1. 作成する RD エリア (RDAREA01) を指定します。
2. RDAREA01 に割り当てるグローバルバッファ (RDDATA10) を指定します。
(6) で調べた RD エリア用 (TYPE=R) グローバルバッファをここに指定します。
なお、次回の HiRDB 開始以降はここで指定したグローバルバッファに割り当てられません。HiRDB を終了したら、システム共通定義の pbuffer オペランドでグローバルバッファを割り当ててください。
(例)

```
pbuffer -a RDDATA10 -n100 -r RDDATA10,RDAREA01 -w 20
```
3. RD エリアの種類を指定します。この演習では、RDAREA01 を公用 RD エリア (PUBLIC) にします。
4. ページ長を指定します。この演習では、ページ長を 4096 バイトにします。
5. セグメントサイズを指定します。この演習では、1 セグメントの大きさを 10 ページにします。
6. RD エリアを構成する HiRDB ファイルを指定します。\$HOME/HiRDB/rdarea/area01 は、(2) で作成した HiRDB ファイルシステム領域です。pdmod コマンドを実行すると、\$HOME/HiRDB/rdarea/area01 ディレクトリ下に rdarea01 ファイルが作成されます。
7. HiRDB ファイルのセグメント数を指定します。この演習では、10 セグメントとします。

(8) pdmod コマンドで RDAREA01 を追加します

作成した制御文ファイルを入力にして、データベース構成変更ユティリティ (pdmod) を実行します。次のコマンドを実行してください。

```
$ pdmod -a $HOME/HiRDB/pdmod/create.txt
```

〔説明〕

-a : (7) で作成した制御文ファイルの名称を指定します。

(9) pdmod コマンドの実行結果が表示されます

```
KFPX14250-I Processing of create rdarea statement ended return code=0,  
RDAREA name="RDAREA01"  
KFPX24213-I DB modification for ended return code=0 at 18:54:09 on 2004/02/03  
KFPX24200-I DB modification ended, return code=0
```

〔説明〕

リターンコードに 0 が返されたので、pdmod コマンドが正常終了しました。

pdmod コマンドが失敗すると (RD エリアの作成に失敗すると)

pdmod コマンドが失敗すると、リターンコードが0で返ってきません。この場合は、syslog に出力されたメッセージを参照して対策してください。下の例は、create rdarea 文で存在する RDAREA を指定したときに出力されたメッセージです。

```
KFPX14220-E Line 1:"rdarea" operand value "RDAREA01" duplicate
KFPX14220-E Line 6:"file" operand value
"C:/hirdb/rdarea/area01/rdarea01" duplicate
KFPX14250-I Processing of create rdarea statement ended
return code=8, RDAREA name="RDAREA01"
KFPX24213-I DB modification for ended return code=12
at 16:06:16 on 2001/09/10
KFPX24200-I DB modification ended, return code=12
```

(10) pddbls コマンドで、RDAREA01 が作成されたことを確認します
RD エリアが作成されたかどうかを確認します。次のコマンドを実行してください。

```
$ pddbls -r ALL
```

(11) pddbls コマンドの実行結果が表示されます

```
STATE OF RDAREA
RDAREA      ID      STATUS      TYPE      OPNMODE
RDMAST      1       OPEN        MAST      INITIAL
RDDIRT      2       OPEN        DDIR      INITIAL
RDDICT      3       OPEN        DDIC      INITIAL
DIC_RTN_SRC 4       OPEN        DLOB      INITIAL
DIC_RTN_OBJ 5       OPEN        DLOB      INITIAL
RDDATA10    6       OPEN        USER      INITIAL
RDINDX10    7       OPEN        USER      INITIAL
RLOB1       8       OPEN        ULOB      INITIAL
RDAREA01   9       OPEN       USER     INITIAL
```

〔説明〕

RDAREA01 が追加されています。

(12) pddef コマンドの入力ファイルを作成します

pddef コマンドの CREATE TABLE 文で作成した RD エリアに表を定義します。この演習では、次に示す内容の入力ファイル \$HOME/HiRDB/pddef/c_table.txt を作成してください。

```
CREATE FIX TABLE EMP (SNO CHAR(6), SNAME CHAR(12), SYEAR CHAR(4))
IN RDAREA01;      ここに改行を入れてください。
```

4. HIRDB を運用してみましょう

〔説明〕

- セミコロン (;) のあとに改行 (Enter) を入れてください。
- この CREATE TABLE で次に示す列を持つ社員表 (表名: EMP) が作成され
ます。
 - SNO (従業員番号)
 - SNAME (社員名)
 - SYEAR (入社年度)

(13) pddef コマンドで社員表を定義します

作成した入力ファイルを使って、データベース定義ユーティリティ (pddef) を実行しま
す。

```
$ pddef < $HOME/hirdb/pddef/c_table.txt
```

(14) pddef コマンドの実行結果が表示されます

```
KFPA12000-I Processing of SQL completed
```

〔説明〕

社員表が定義されました。

(15) 社員表が定義されたかを確認します

表が定義されたかをディクショナリ表を検索して確認します。HIRDB SQL Executer を起
動してください。

【SQL 入力画面】

```
SELECT TABLE_SCHEMA, TABLE_NAME FROM MASTER.SQL_TABLES  
WHERE TABLE_NAME='EMP';
```

【検索結果画面】

```
TABLE_SCHEMA          TABLE_NAME  
-----  
MANAGER              EMP  
KFPA27010-I          1 rows selected
```

〔説明〕

社員表 (EMP) が定義されています。

(16) データを挿入します

INSERT 文で社員表にデータを挿入します。次の SQL 文を順に実行してください。

【SQL 入力画面】

```
INSERT INTO EMP VALUES ('189218', 'NAKATA', '1989');
INSERT INTO EMP VALUES ('192365', 'SUZUKI', '1992');
INSERT INTO EMP VALUES ('185995', 'TANAKA', '1985');
```

【検索結果画面】

```
KFPX27010-I          1 rows inserted
KFPX27010-I          1 rows inserted
KFPX27010-I          1 rows inserted
```

〔説明〕

データが挿入されました。

(17) 社員表を検索します

データが正しく入力されているか確認します。次の SQL 文を実行してください。

【SQL 入力画面】

```
SELECT * FROM EMP;
```

【検索結果画面】

```

SNO      SNAME      SYEAR
-----
189218   NAKATA     1989
192365   SUZUKI     1992
185995   TANAKA     1985

KFPX27010-I          3 rows selected
```

4.9 ユーザ権限を与える（ユーザを登録する）

ユーザ権限の付与について説明します。

4.9.1 基礎知識

HiRDB はユーザ権限を使ってデータベースの機密保護を実現しています。このため、HiRDB のユーザを登録する場合、そのユーザにユーザ権限を与える必要があります。ユーザ権限がないと、データベースにアクセスできません。

ユーザ権限には表 4-6 に示す種類があります。

表 4-6 ユーザ権限の種類

ユーザ権限の種類	説明	この権限を持っているとできること
DBA 権限	ユーザ権限を与えるおよび取り消すために必要な権限です。権限を管理する人が持つ権限です。HiRDB 管理者は DBA 権限を持っています。DBA 権限を持つユーザは、次に示す権限を持っています。 <ul style="list-style-type: none"> CONNECT 権限 すべての HiRDB ユーザに対するスキーマ定義権限 	<ul style="list-style-type: none"> ほかの人に、ユーザ権限を与えることができます。 付与したユーザ権限を取り消せます。 ほかの人のスキーマを定義できます。 ほかの人のスキーマを削除できます。 表およびインデクスを作成できます。
CONNECT 権限	HiRDB を利用するために必要な権限です。CONNECT 権限を持たないユーザが HiRDB を利用しようとするエラーになります。	<ul style="list-style-type: none"> HiRDB に接続（CONNECT）できるようになります。
スキーマ定義権限	スキーマを定義するために必要な権限です。	<ul style="list-style-type: none"> 自分のスキーマを定義できます。 自分のスキーマを削除できます。 表およびインデクスを作成できます。
RD エリア利用権限	私用 RD エリアを利用するために必要な権限です。公用 RD エリアに表またはインデクスを作成する場合は、RD エリア利用権限は不要です。	私用 RD エリアに表およびインデクスを作成できます。
アクセス権限	表にアクセスするために必要な権限です。アクセス権限は表単位に設定し、次に示す 4 種類があります。	ほかの人の表にアクセスできます。
	SELECT 権限	表の検索（SELECT）ができます。
	INSERT 権限	表にデータを追加（INSERT）できます。
	DELETE 権限	表のデータを削除（DELETE）できます。
	UPDATE 権限	表のデータを更新（UPDATE）できます。

(1) 表をアクセスする人に与える権限は

表をアクセスする人には、次に示す権限を定義系 SQL の GRANT 文で与えてください。

- CONNECT 権限
- 表のアクセス権限

(2) 表を定義する人に与える権限は

表を定義する人には、次に示す権限を定義系 SQL の GRANT 文で与えてください。

- CONNECT 権限
- スキーマ定義権限
- RD エリア利用権限

なお、RD エリア利用権限は、私用 RD エリアに表またはインデクスを定義する場合にだけ必要です。公用 RD エリアに表またはインデクスを定義する場合は不要です。

4.9.2 演習 1 (新規ユーザを登録する)

新規の HiRDB ユーザ (ユーザ名「USER01」、パスワード「PASS01」) に次に示す権限を与えます。HiRDB SQL Executer を起動していなければ、HiRDB SQL Executer を起動してください。

GRANT 文で USER01 に次に示す権限を与えます。

- CONNECT 権限
- CUSTOM 表の SELECT 権限
- スキーマ定義権限

次の SQL 文を入力してください。

【SQL 入力画面】

```
GRANT CONNECT TO USER01 IDENTIFIED BY PASS01;          .....1
GRANT SELECT ON CUSTOM TO USER01;                      .....2
GRANT SCHEMA TO USER01;                                .....3
```

〔説明〕

1. USER01 (パスワード: PASS01) に CONNECT 権限を与えます。
2. USER01 に CUSTOM 表の SELECT 権限を与えます。
3. USER01 にスキーマ定義権限を与えます。

【検索結果画面】

4. HiRDB を運用してみましょう

```
KFPX27001-I Processing of SQL completed  
KFPX27001-I Processing of SQL completed  
KFPX27001-I Processing of SQL completed
```

【説明】

USER01 に各種の権限が与えられました。

4.9.3 演習 2 (アクセス権限がある表を検索する)

ユーザ名「USER01」で、CUSTOM 表を検索します。

(1) ユーザ名「USER01」で HiRDB に接続します

現在、HiRDB SQL Executer を使用して HiRDB に接続しているユーザは MANAGER です。EXIT コマンドで HiRDB への接続を切断してください。

その後、次のコマンドを実行して「USER01」で HiRDB に接続してください。パスワードは、「PASS01」です。

```
$ /opt/pdsq1/pdsq1 -u USER01
```

(2) CUSTOM 表を検索します

HiRDB に接続したら、CUSTOM 表を検索します。次の SQL 文を入力してください。

【SQL 入力画面】

```
SELECT * FROM MANAGER.CUSTOM;
```

【説明】

CUSTOM 表の所有者は MANAGER のため、「認可識別子・表識別子」の形式で表名を指定します。

【検索結果画面】

CUSTOM_CD	CUSTOM_NAME	TELNO	ZIPCD	ADDRESS
TK001	A electric appliance store	011-434-1998	091	Totsuka
TK002	B department store	012-402-5055	082	Yokohama
TK003	C products company	014-252-1555	043	Kanazawa
	⋮			
TK098	CT electric appliance store	052-476-5624	066	Fujiyama
TK099	CU &Co.,Ltd.	033-359-4563	057	Hiyoshi
TK100	DEN convenience store	011-921-4222	080	Totsuka
KFPX27010-I	100 rows selected			

〔説明〕

CUSTOM 表の検索結果が表示されます。

4.9.4 演習 3 (アクセス権限がない表を検索する)

アクセス権限がない VENDOR 表をアクセスすると、アクセスを拒否されます。USER01 で、次の SQL 文を入力してください。

【SQL 入力画面】

```
SELECT * FROM MANAGER.VENDOR;
```

〔説明〕

VENDOR 表の所有者は MANAGER のため、「認認識別子・表識別子」の形式で表名を指定します。

【検索結果画面】

```
KFPA11548-E Unable to execute SELECT:no SELECT privilege for
table MANAGER.VENDOR
```

〔説明〕

VENDOR 表に対する SELECT 権限がないため、VENDOR 表を検索できません。

4.9.5 演習 4 (権限がないユーザで表を検索する)

権限を与えていないユーザ名「USER02」で表を検索します。

(1) ユーザ名「USER02」で HiRDB に接続します

現在、HiRDB SQL Executer を使用して HiRDB に接続しているユーザは USER01 で

4. HiRDB を運用してみましょう

す。EXIT コマンドで HiRDB への接続を切断してください。

そのあと、次のコマンドを実行して「USER02」で HiRDB に接続してください。パスワードは、「PASS02」です。

```
$ /opt/pdsql/pdsql -u USER02
```

(2) USER02 は HiRDB への接続を拒否されます

権限のないユーザ「USER02」では、HiRDB に接続できません。次に示すメッセージが表示されます。

```
KFPA11561-E Specified authorization identifier USER02 has  
no connect privilege
```

【説明】

CONNECT 権限がないため、HiRDB に接続できません。

4.9.6 演習 5 (表の所有者を調べる)

表の所有者を調べるにはディクショナリ表を検索します。

(1) ユーザ名「MANAGER」で HiRDB に接続します

現在、HiRDB SQL Executer を使用して HiRDB に接続しているユーザは USER02 です。EXIT コマンドで HiRDB への接続を切断してください。

そのあと、HiRDB SQL Executer を起動して「MANAGER」で HiRDB に接続してください。

(2) ディクショナリ表を検索します

ディクショナリ表を検索します。次の SQL 文を実行してください。

【SQL 入力画面】

```
SELECT TABLE_SCHEMA, TABLE_NAME FROM MASTER.SQL_TABLES;
```

【検索結果画面】

```
TABLE_SCHEMA          TABLE_NAME
-----
:
HiRDB                 SQL_VIEW_DEF
HiRDB                 SQL_VIEW_TABLE_USAGE
MANAGER               CUSTOM
MANAGER               EMP
MANAGER               VENDOR
MASTER               SQL_ALIASES
MASTER               SQL_COLUMNS
:
:
KFPX27010-I          97 rows selected
```

〔説明〕

- TABLE_NAME に表示されている情報が表の名称です。
- TABLE_SCHEMA に表示されている情報が表の所有者です。

5

HiRDB Control Manager で HiRDB を運用してみましょ う

この章では、HiRDB Control Manager を使用した HiRDB の基礎的な運用方法について説明します。HiRDB Control Manager を使用するには、操作端末として Windows マシンが必要です。

-
- 5.1 この章を読むに当たって

 - 5.2 HiRDB Control Manager を開始，終了する

 - 5.3 HiRDB を登録する

 - 5.4 HiRDB を開始，終了する

 - 5.5 システムログを操作する

 - 5.6 バックアップを取得する

 - 5.7 データベースを回復する

 - 5.8 表を再編成する

 - 5.9 RD エリアを拡張する

 - 5.10 RD エリアを追加する

 - 5.11 ユーザ権限を与える（ユーザを登録する）
-

5.1 この章を読むに当たって

この章を読むに当たって、知っておいていただきたいことについて説明します。

(1) HiRDB Control Manager について

HiRDB Control Manager を使用するには、操作端末として Windows マシンが必要です。Windows マシンには、HiRDB Control Manager - Console および HiRDB Control Manager - Server がインストールされている必要があります。

GUI 運用支援 HiRDB Control Manager を使用すると、システムログファイルの操作やバックアップの取得などの運用をマウス操作で実行できます。したがって、コマンドの実行が不要になるため、HiRDB のシステム運用が簡単になります。

なお、この章の演習は、4 章の各節の基礎知識を読んだあとに実施してください。

また、4 章の演習を実施したあとにこの章の演習を実施すると、コマンドでの運用と HiRDB Control Manager での運用を比較できます。このため、4 章の演習を実施したあとにこの章の演習を実施することをお勧めします。

(2) この章で説明している項目

この章では、HiRDB の基礎的な運用方法について説明しています。説明している項目は次のとおりです。

- HiRDB Control Manager の開始，終了
- HiRDB の登録
- HiRDB の開始，終了
- システムログの操作
- バックアップの取得
- データベースの回復
- 表の再編成
- RD エリアの拡張
- RD エリアの追加
- ユーザ権限の付与（ユーザの登録）

注

HiRDB Control Manager で DAT にバックアップを取得するには、NetBackup などの付加プログラムが必要です。このマニュアルでは付加プログラムをインストールしていないので DAT へのバックアップ演習を行いません。

(3) この章の読み方について

この章は、章全体で一つの演習になっています。したがって、途中から読まないで、5.2 から順次読み進んでください。

5.2 HiRDB Control Manager を開始 , 終了する

HiRDB Control Manager の開始と終了について説明します。

5.2.1 HiRDB Control Manager を開始する

HiRDB Control Manager の開始について説明します。

(1) HiRDB Control Manager - Server を開始します

HiRDB Control Manager - Console を開始するには、HiRDB Control Manager - Server のサービスが起動されている必要があります。サービスが起動されていない場合は、[コントロールパネル] - [管理ツール] - [サービス] で「HiRDB CM - Server」のサービスを開始してください。

(2) HiRDB Control Manager - Agent を開始します

HiRDB Control Manager - Console で HiRDB サーバを操作するには、対象の HiRDB サーバと同じマシンにインストールされている HiRDB Control Manager - Agent のサービスが起動されている必要があります。サービスが起動されていない場合は、HiRDB Control Manager - Agent が正しくセットアップされていない可能性があります。「2.5 HiRDB Control Manager - Agent の環境設定をする」を参照して、サービス名の指定を中心にセットアップ内容を見直してください。

(3) HiRDB Control Manager - Console を開始します

次に示す手順で HiRDB Control Manager を開始します。

手順

1. [スタート] - [プログラム] - [HiRDB CM - Console] - [HiRDB CM - Console] を選択してください。
2. 【HiRDB Control Manager のログイン】ボックスが表示されます。パスワードの欄に HiRDB Control Manager - Server のインストール時に指定したパスワード「hircms」を入力して、「OK」をクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



3. HiRDB Control Manager が開始します。



5.2.2 HiRDB Control Manager を終了する

HiRDB Control Manager の終了について説明します。

(1) HiRDB Control Manager - Console を終了します

HiRDB Control Manager - Console を終了します。[オプション] - [コンソールの終了] を選択してください。

(2) HiRDB Control Manager - Server を終了します

次に示す手順で HiRDB Control Manager - Server を終了します。

手順

1. [コントロールパネル] - [管理ツール] - [サービス] をダブルクリックしてください。
2. 【サービス】に表示されているリストボックスから、「HiRDB Control Manager - Server」を選択し、[操作] - [停止] を選択するか、[サービスの停止] ボタンをクリックしてください。
3. HiRDB Control Manager - Server が終了したら、【サービス】を閉じてください。

5.3 HiRDB を登録する

HiRDB Control Manager - Console が初めて起動したとき、HiRDB Control Manager - Console で操作できる HiRDB (これを管理 HiRDB といいます) はありません。したがって、管理 HiRDB の登録を行う必要があります。管理 HiRDB を登録する手順を次に示します。

なお、HiRDB の登録は、登録対象の HiRDB の稼働中に実行する必要があります。

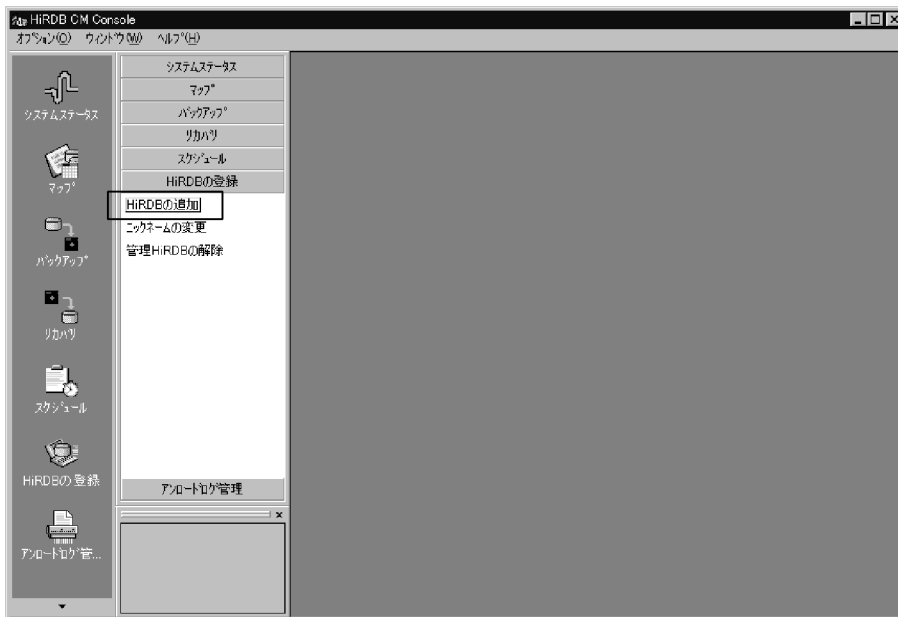
注

HiRDB Control Manager のセットアップで次の設定をしている場合、初めての起動でも管理 HiRDB が存在することもあります。

セットアップ対象	設定
HiRDB Control Manager - Server	HiRDB Control Manager - Agent の自動探索を行う
HiRDB Control Manager - Agent	Rocket Punch 機能を使用する

手順

1. [HiRDB の登録] - [HiRDB の追加] を選択してください。



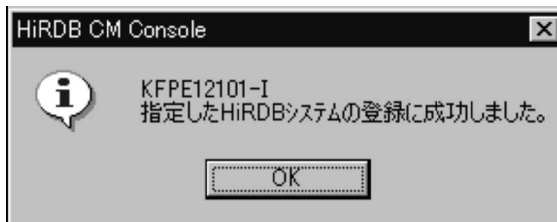
2. 【HiRDB システムの新規登録】ボックスが表示されます。操作したい HiRDB の IP アドレス「10.210.35.14」とその HiRDB に対応する HiRDB Control Manager - Agent が使用するポート番号「20293」を指定して、「OK」ボタンを

5. HiRDB Control Manager で HiRDB を運用してみましょう

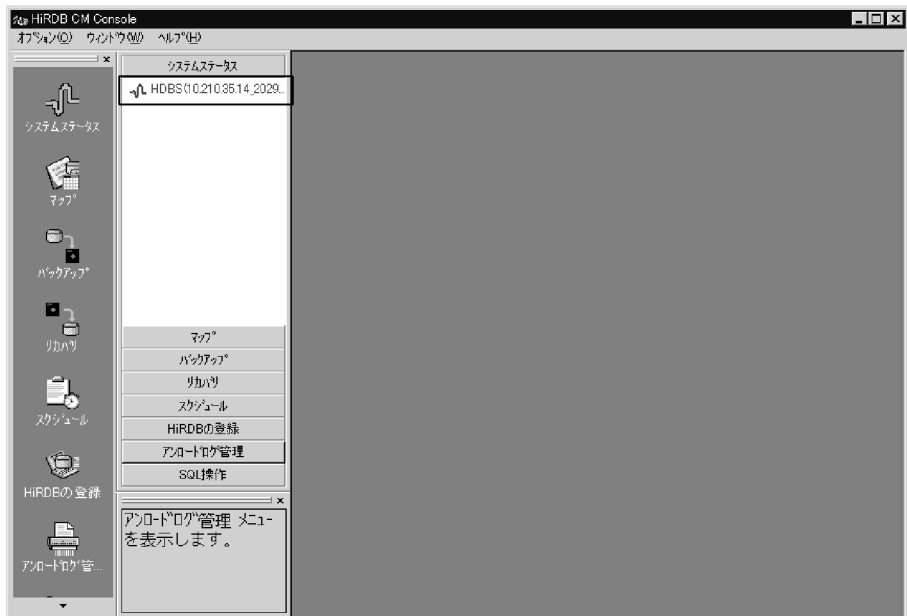
クリックしてください。



3. HiRDB の登録に成功すると、次のメッセージが表示されます。



4. 登録された HiRDB は、[システムステータス] の欄に表示されます。



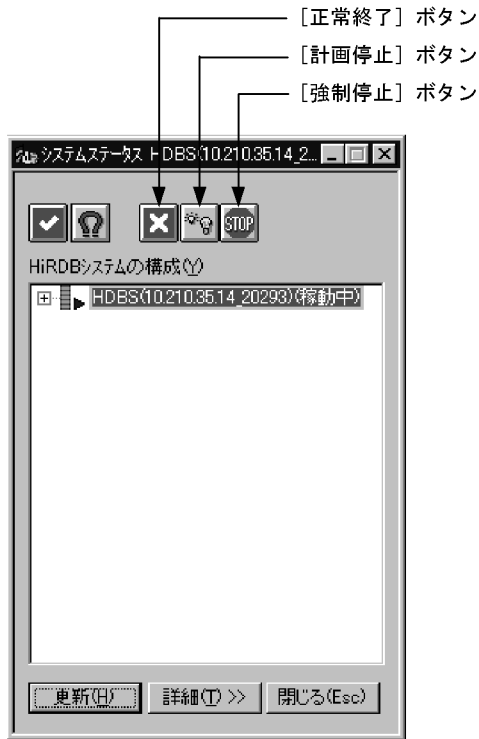
5.4 HiRDB を開始 , 終了する

[システムステータス] 欄の管理 HiRDB をクリックすると,[システムステータス] ボックスが表示されます。[システムステータス] ボックスでは, 管理 HiRDB の開始, 終了や状態表示などの操作が実行できます。



5.4.1 演習 1 (HiRDB を正常終了する)

現在稼働中の HiRDB を正常終了します。[正常終了] ボタンをクリックしてください。



〔説明〕

- [正常終了] ボタンをクリックすると、HiRDB が正常終了します。
- [計画停止] ボタンをクリックすると、HiRDB が計画停止します。
- [強制停止] ボタンをクリックすると、HiRDB が強制終了します。

5.4.2 演習 2 (HiRDB を正常開始する)

[システムステータス] ボックスを操作して、HiRDB を正常開始します。[正常開始] ボタンをクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



〔説明〕

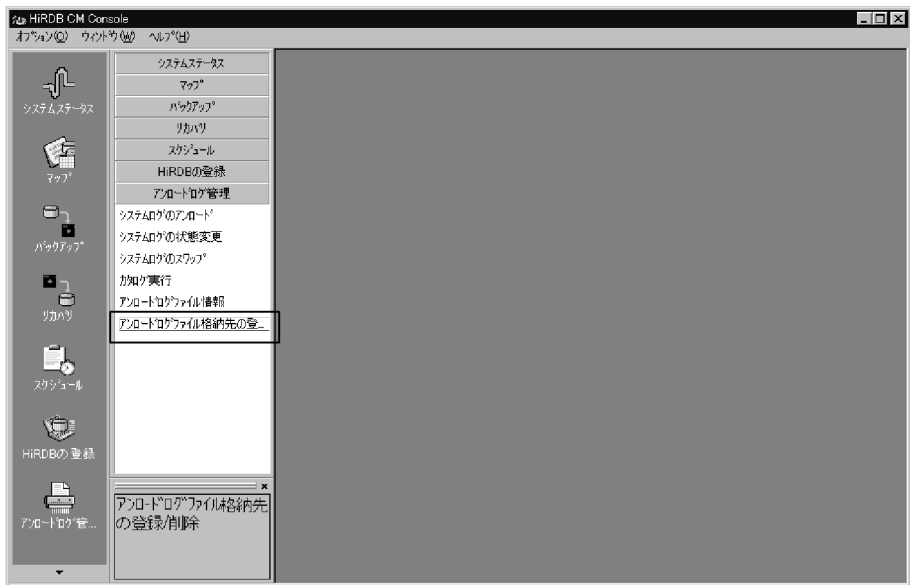
- [正常開始] ボタンをクリックすると、HiRDB が正常開始します。
- [複写・回復専用開始] ボタンをクリックすると、pdstart -r コマンド相当で HiRDB が開始します。

5.5 システムログを操作する

HiRDB Control Manager を使ってアンロードログを取得するには、あらかじめアンロードログファイルの格納先を登録しておく必要があります。アンロードログファイルの格納先を登録していない場合は、次の手順でアンロードログファイルの格納先を登録します。

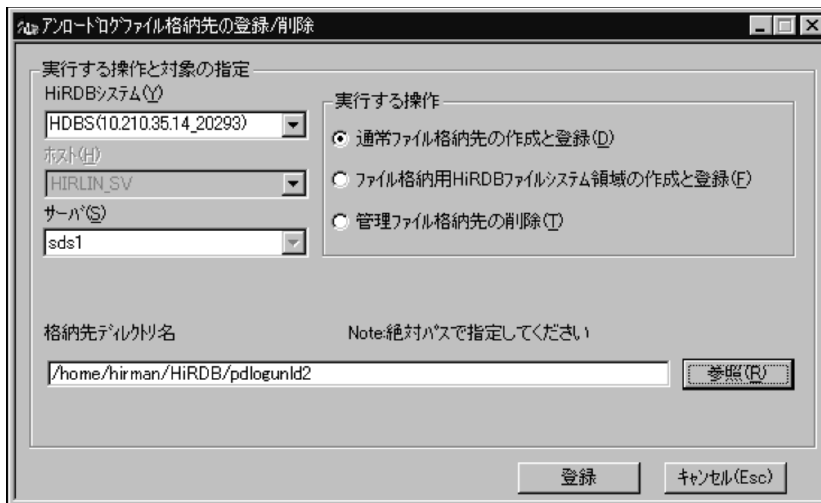
手順

1. [アンロードログ管理] - [アンロードログファイル格納先の登録 / 削除] を選択してください。



2. 【アンロードログファイル格納先の登録 / 削除】ボックスが表示されます。操作対象と操作内容を次のように指定してください。
 HiRDB システム：HDBS (10.210.35.14_20293)
 サーバ：sds1
 実行する操作：通常ファイル格納先の作成と登録
 格納先ディレクトリ名：/home/hirman/HiRDB/pdlogunld2

5. HiRDB Control Manager で HiRDB を運用してみましょう



3. 「登録」をクリックしてください。バックアップファイルを格納するディレクトリが作成されます。

5.5.1 演習（システムログをアンロードする）

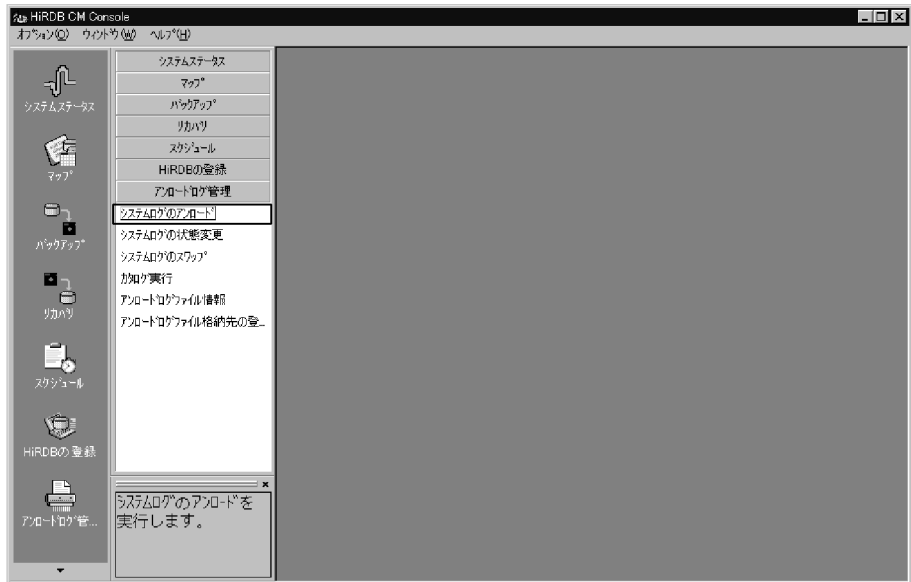
「アンロード待ち状態」のシステムログファイルをアンロードします。

- (1) 「アンロード待ち状態」のシステムログファイルをアンロードします

手順

1. [アンロードログ管理] - [システムログのアンロード] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【システムログのアンロード】ボックスが表示されます。

各設定値を次のように指定してください。

HiRDB システム：HDBS (10.210.35.14_20293)

サーバ：sds1

アンロードログファイルの最大保持数：256

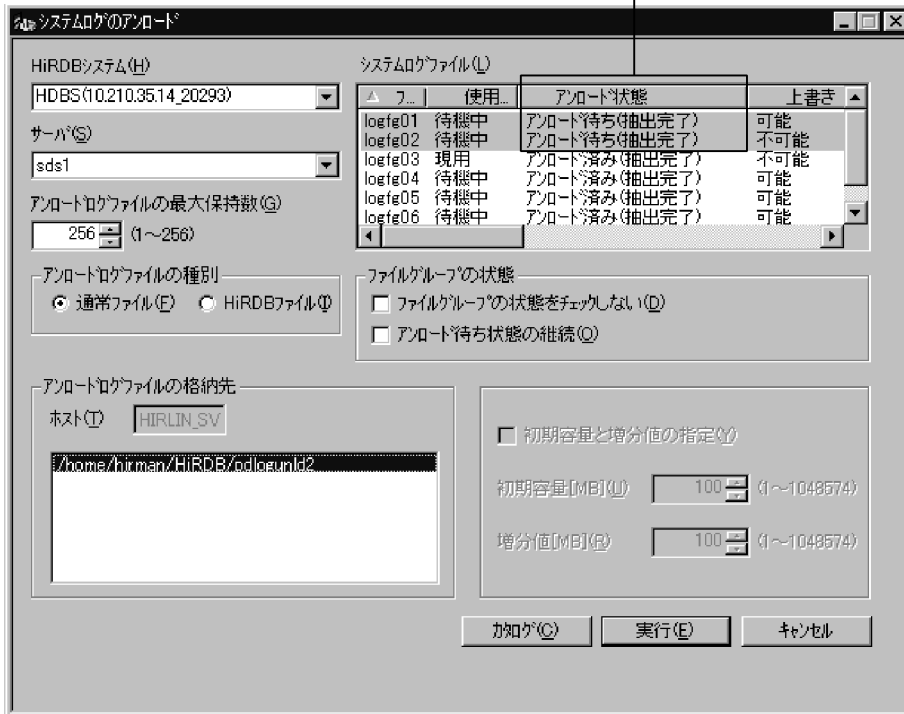
アンロードログファイルの種別：通常ファイル

アンロードログファイルの格納先：/home/hirman/HiRDB/pdlogunld2

システムログファイル：アンロード状態が「アンロード待ち」になっているファイルを選択

5. HIRDB Control Manager で HIRDB を運用してみましょう

アンロード待ち状態のログを選択する。



3. 「実行」をクリックしてください。システムログがアンロードされます。

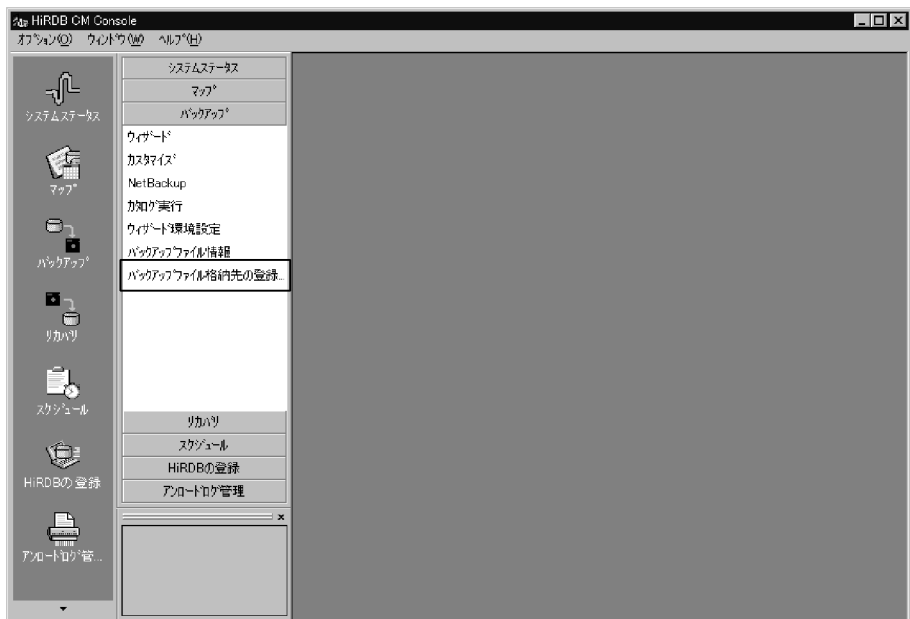
5.6 バックアップを取得する

HiRDB Control Manager を使ってバックアップを取得するには、あらかじめバックアップファイルの格納先を登録しておく必要があります。バックアップの格納先を登録していない場合は、次の手順でバックアップの格納先を登録します。

また、HiRDB Control Manager で DAT にバックアップを取得するには、NetBackup などの付加プログラムが必要です。このマニュアルでは付加プログラムをインストールしていないので DAT へのバックアップ演習を行いません。

手順

1. [バックアップ] - [バックアップ格納先の登録 / 削除] を選択してください。



2. 【バックアップファイル格納先の登録 / 削除】ボックスが表示されます。

操作対象と操作内容を次のように指定してください。

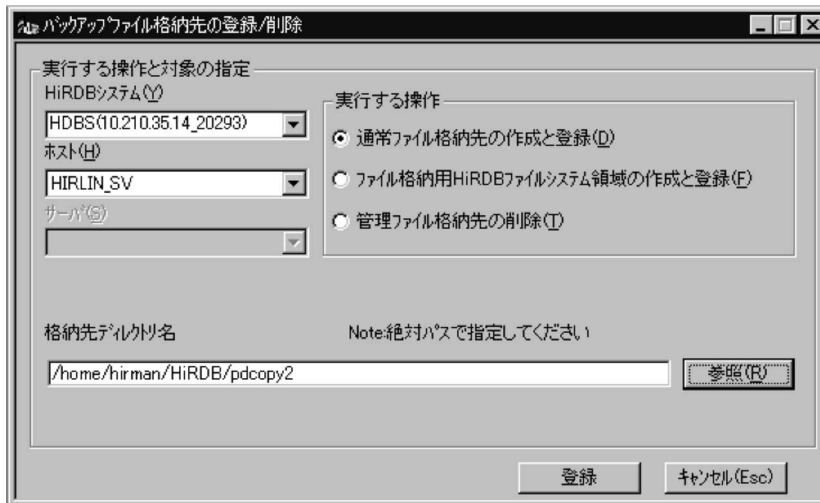
HiRDB システム：HDBS (10.210.35.14_20293)

ホスト：HiRLIN_SV

実行する操作：通常ファイル格納先の作成と登録

格納先ディレクトリ名：/home/hirman/HiRDB/pdcopy2

5. HiRDB Control Manager で HiRDB を運用してみましょう



3. 「登録」をクリックしてください。バックアップファイルを格納するディレクトリが作成されます。

5.6.1 演習 1 (全 RD エリア (HiRDB システム) のバックアップを取得する)

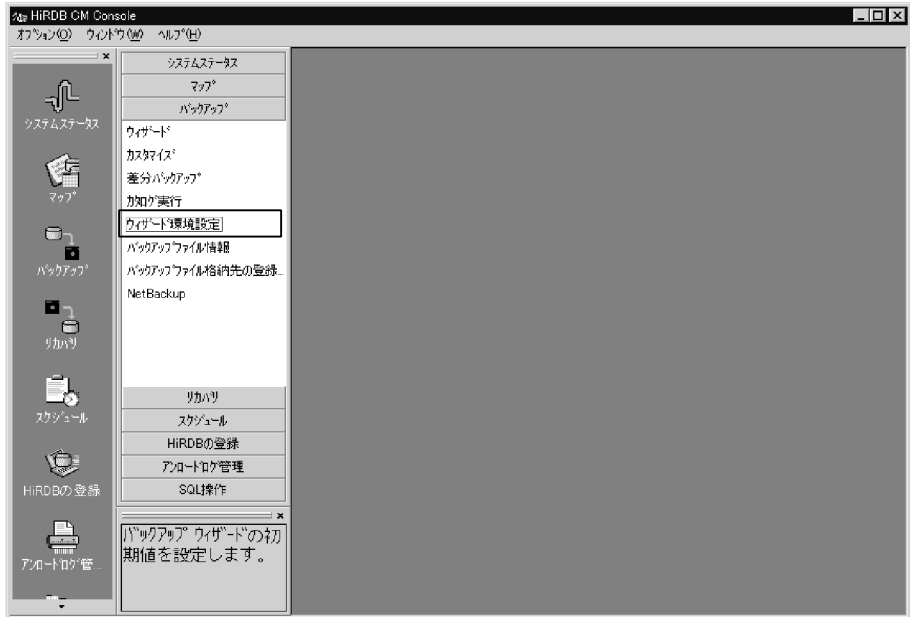
全 RD エリア (HiRDB システム) のバックアップを一括して取得します。バックアップ取得モードは `r` とします。

なお、ここで取得したバックアップは「5.7 データベースを回復する」の演習で使用するので、保存しておいてください。

手順

1. [バックアップ] - [ウィザード環境設定] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【バックアップウィザード環境設定】ボックスが表示されます。
各項目を次のように設定して、「保存」ボタンをクリックしてください。
HiRDB システム：HDBS (10.210.35.14_20293)
バックアップ取得モード：参照可能 / 更新不可能モード

5. HiRDB Control Manager で HiRDB を運用してみましょう

バックアップウィザード環境設定

HiRDBシステム(H)

HiRDBS102103514_202303

バックアップ取得モード

参照・更新不可能モード(N)

参照可能/更新不可能モード(R)

参照・更新可能モード(L)

JP1/OmniBack II

パーストファイル名(P)

追加(A)

削除(D)

デバイスエージェントのあるホスト名(Q)

ファイル種別

通常ファイル(F)

HiRDBファイルシステム領域(A)

EasyMT(E)

EasyMT, MTguide(G)

JP1/OmniBack II(J)

EasyMT

属性定義ファイル名(O)

追加(A)

削除(D)

MTのボリューム名(M)

追加(A)

削除(D)

入出力バックアップ面数(B)

1 (1~255)

処理結果出力ファイル

バックアップの処理結果ファイル名(C)

RDエリア使用状況の出力(D)

排他待ち限界経過時間(秒)(W)

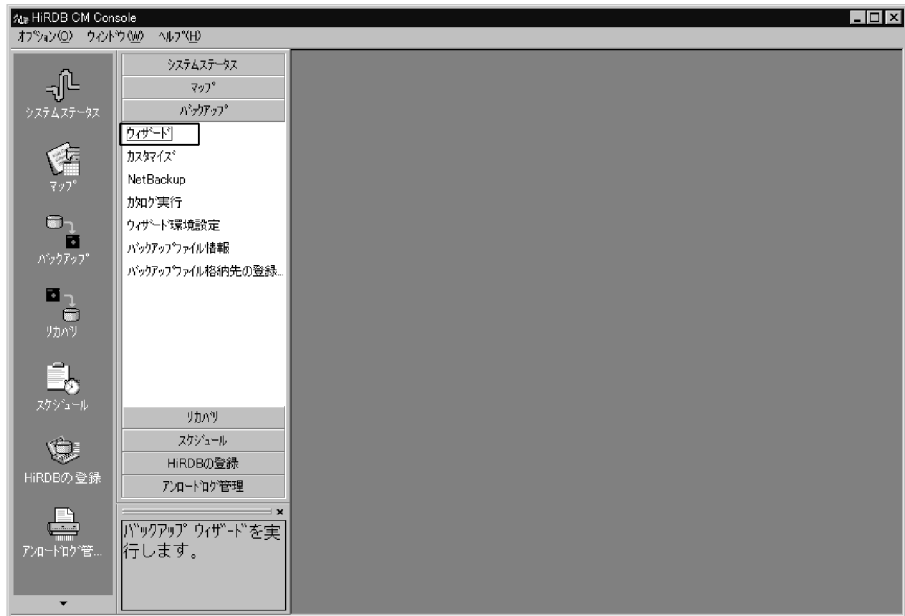
180 (0~200000)

スキップ対象エラーを検知した場合は処理を続行する(Y)

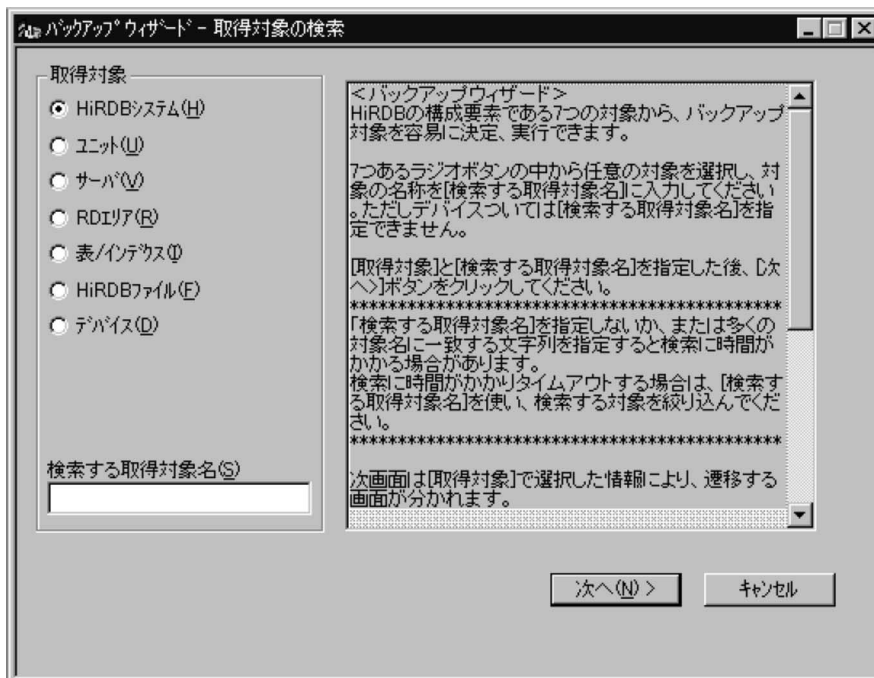
保存(S) キャンセル(Esc)

3. [バックアップ] - [ウィザード] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



4. 【バックアップウィザード - 取得対象の検索】ボックスが表示されます。
取得対象に「HiRDBシステム」を選択して、「次へ」ボタンをクリックしてください。

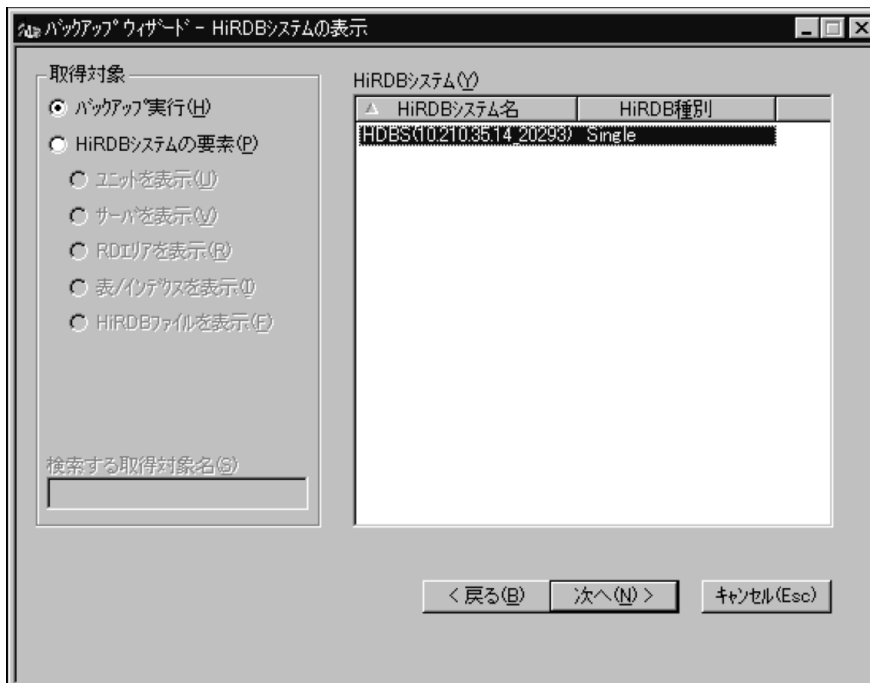


5. 【バックアップウィザード - HiRDB システムの表示】ボックスが表示されます。
各項目を次のように指定して、「次へ」ボタンをクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう

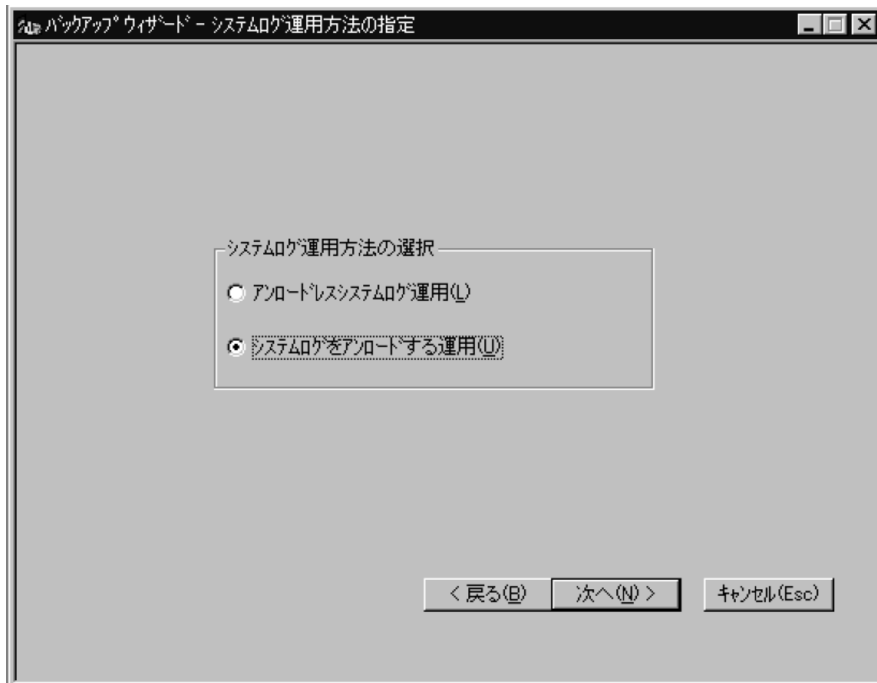
取得対象：バックアップ実行

HiRDB システム：HDBS (10.210.35.14_20293)



6. 【バックアップウィザード - システムログ運用方法の指定】ボックスが表示されます。
- システムログ運用方法に「システムログをアンロードする運用」を選択して、「次へ」ボタンをクリックしてください。

5. HIRDB Control Manager で HIRDB を運用してみましょう



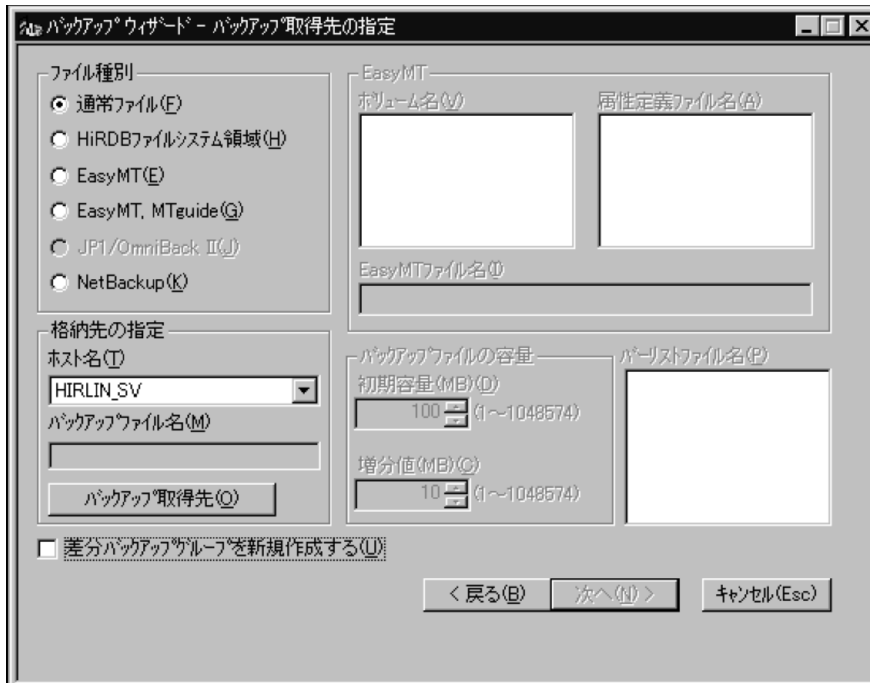
7. 【バックアップウィザード - バックアップ取得先の指定】ボックスが表示されま
す。

各項目を次のように指定して、「バックアップ取得先」ボタンをクリックしてく
ださい。

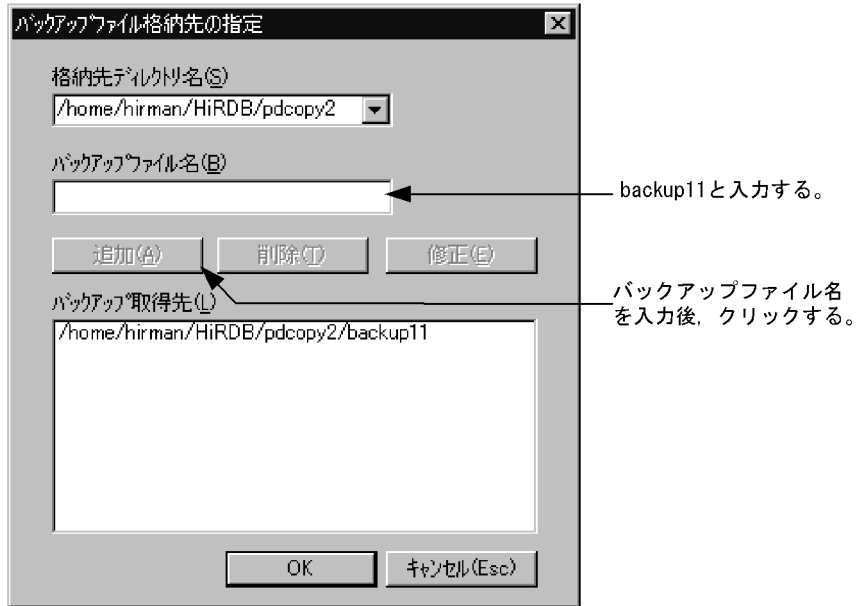
ファイル種別：通常ファイル

ホスト名：HIRLIN_SV

5. HiRDB Control Manager で HiRDB を運用してみましょう



8. 【バックアップファイル格納先の指定】ダイアログが表示されます。
バックアップ先の情報を次のように入力してから、「追加」ボタンをクリックしてください。バックアップ取得先に情報が表示されたら、「OK」ボタンをクリックしてください。
格納先ディレクトリ名：/home/hirman/HiRDB/pdcopy2
バックアップファイル名：backup11



9. 【バックアップウィザード - バックアップ取得先の指定】ボックスが表示されま
す。
「次へ」ボタンをクリックしてください。



10. 【バックアップウィザード - 確認】ボックスが表示されます。
設定した条件を確認して、問題がなければ「実行」ボタンをクリックしてくださ

5. HiRDB Control Manager で HiRDB を運用してみましょう

い。バックアップ処理が始まります。



5.6.2 演習 2 (RD エリアのバックアップを取得する)

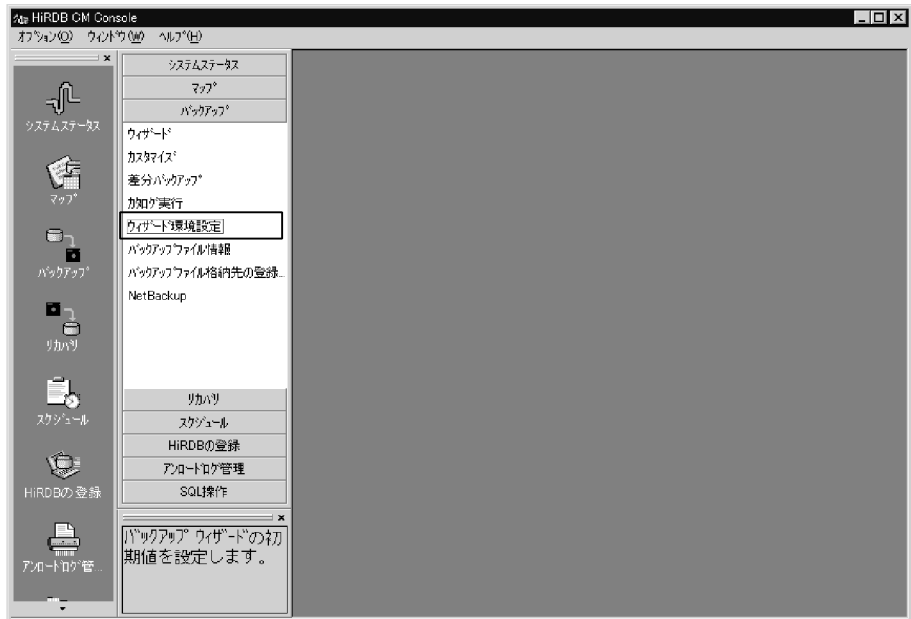
RD エリア (RDDDATA10) のバックアップを取得します。バックアップ取得モードは x とします。

なお、ここで取得したバックアップは「5.7 データベースを回復する」の演習で使用する
ので、保存しておいてください。

手順

1. [バックアップ] - [ウィザード環境設定] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【バックアップウィザード環境設定】ボックスが表示されます。
各項目を次のように設定して、「保存」ボタンをクリックしてください。
HiRDB システム：HDBS (10.210.35.14_20293)
バックアップ取得モード：参照・更新不可能モード

5. HiRDB Control Manager で HiRDB を運用してみましょう

バックアップウィザード環境設定

HiRDBシステム(H)

HiRDBS102103514_202303

バックアップ取得モード

参照・更新不可能モード(N)

参照可能/更新不可能モード(R)

参照・更新可能モード(L)

JP1/OmniBack II

パーストファイル名(P)

追加(A)

削除(D)

デバイスエージェントのあるホスト名(O)

ファイル種別

通常ファイル(F)

HiRDBファイルシステム領域(A)

EasyMT(E)

EasyMT, MTguide(G)

JP1/OmniBack II(L)

EasyMT

属性定義ファイル名(D)

追加(A)

削除(D)

MTのボリューム名(M)

追加(A)

削除(D)

入出力バッファ面数(B)

1 (1~255)

処理結果出力ファイル

バックアップの処理結果ファイル名(C)

RDエリア使用状況の出力(O)

排他待ち限界経過時間(秒)(W)

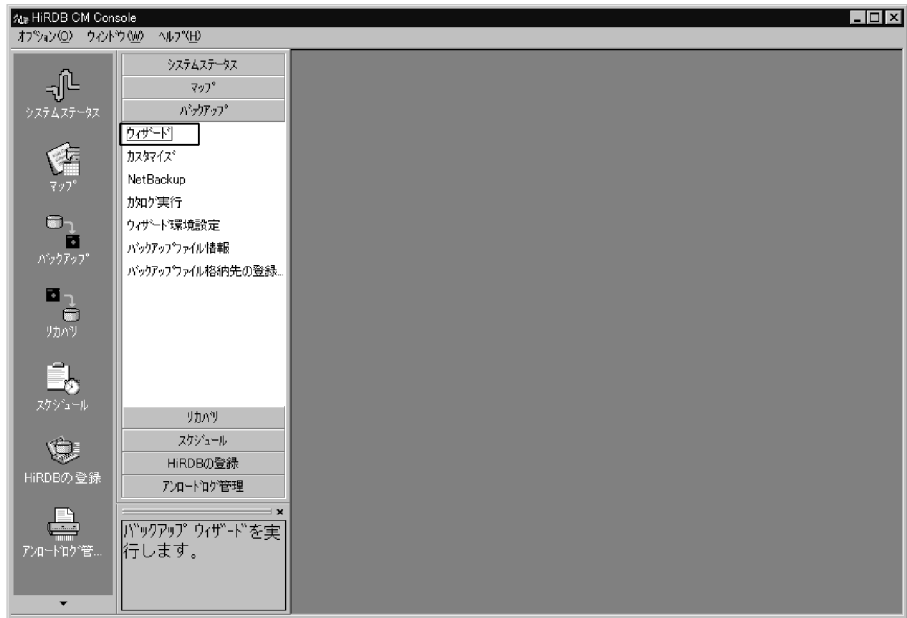
180 (0~200000)

スキップ対象エラーを検知した場合は処理を続行する(Y)

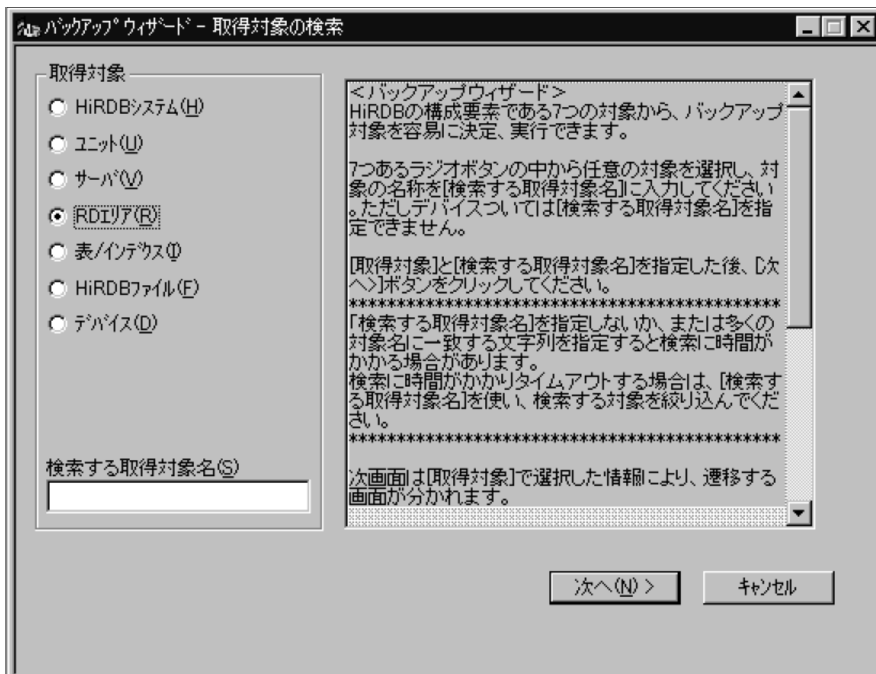
保存(S) キャンセル(Esc)

3. [バックアップ] - [ウィザード] を選択してください。

5. HIRDB Control Manager で HIRDB を運用してみましょう



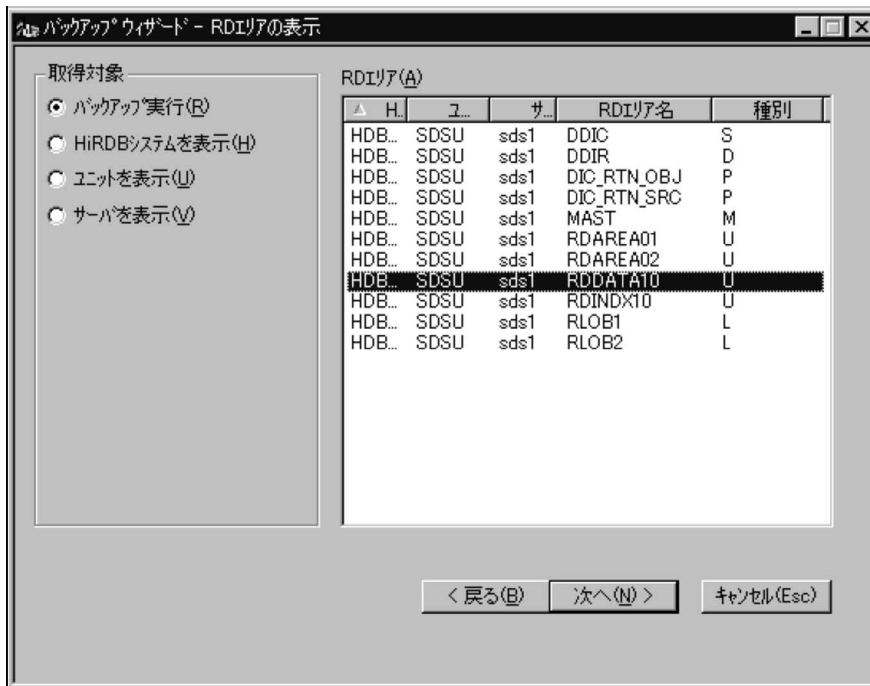
4. 【バックアップウィザード - 取得対象の検索】ボックスが表示されます。
取得対象に「RD エリア」を選択して、「次へ」ボタンをクリックしてください。



5. 【バックアップウィザード - RD エリアの表示】ボックスが表示されます。
取得対象に「バックアップ実行」、RD エリアに「RDDATA10」を選択して、「次へ」ボタンをクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう

このボックスが表示されたとき、「RD エリア名」は表示領域の外にあります。「HiRDB システム名」「ユニット名」「サーバ名」などの表示幅を短くして「RD エリア名」を表示してください。

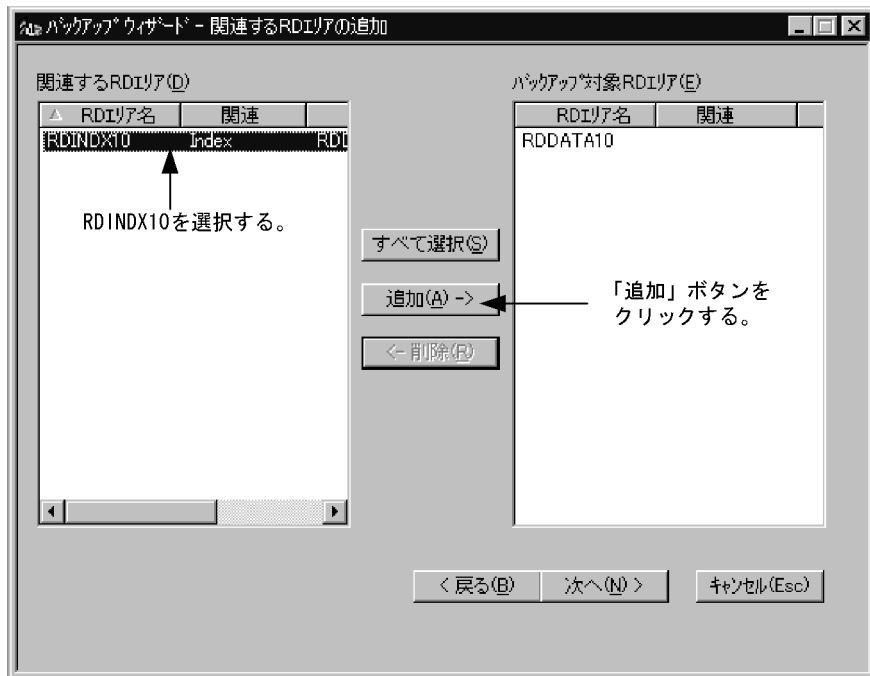


6. 【バックアップウィザード - 関連する RD エリアの追加】ボックスが表示されま

す。
関連する RD エリアから「RDINDX10」を選択して、「追加」ボタンをクリックしてください。バックアップ対象 RD エリアに RDINDX10 が移動したら、「次へ」ボタンをクリックしてください。

このボックスは、選択した RD エリアと論理的な関係がある RD エリアがある場合に表示されます。論理的な関係がある RD エリアは同時にバックアップを取得しておかないと、データベース回復時に整合性が合わなくなる可能性があります。

5. HIRDB Control Manager で HIRDB を運用してみましょう



7. 【バックアップウィザード - バックアップ取得先の指定】ボックスが表示されま

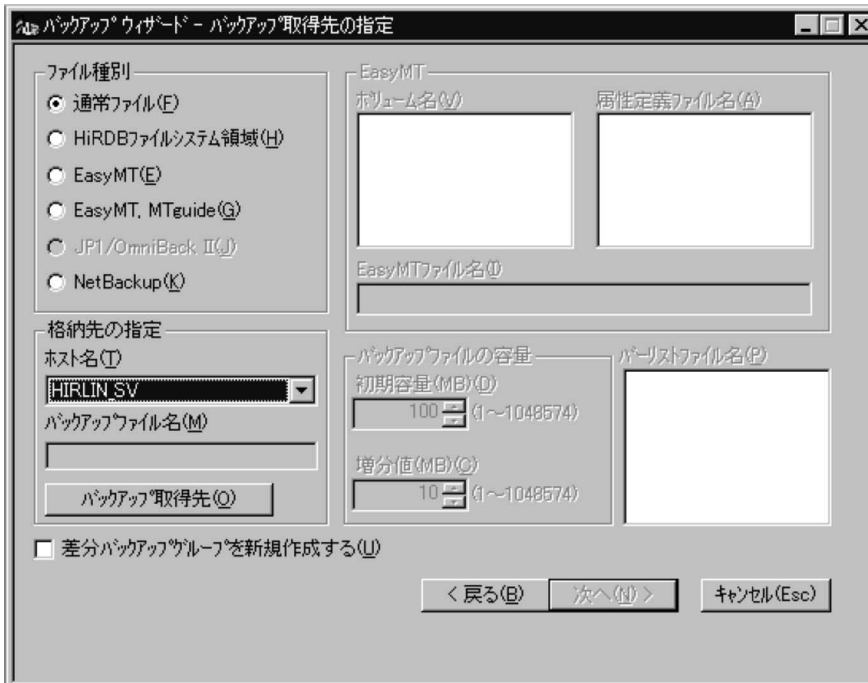
5. HiRDB Control Manager で HiRDB を運用してみましょう

す。

バックアップ取得先の情報を次のように指定して、「バックアップ取得先」ボタンをクリックしてください。

ファイル種別：通常ファイル

ホスト名：HIRLIN_SV



8. 【バックアップ格納先の指定】ボックスが表示されます。

バックアップ先の情報を次のように入力してから、「追加」ボタンをクリックしてください。バックアップ取得先に情報が表示されたら、「OK」ボタンをクリックしてください。

格納ディレクトリ：HIRLIN_SV:/home/hirman/HiRDB/pdcopy2

バックアップファイル名：backup12

5. HIRDB Control Manager で HIRDB を運用してみましょう



9. 【バックアップウィザード - バックアップ取得先の指定】ボックスが表示されます。
- バックアップファイル名に、前の画面で指定した値が表示されていることを確認して、「次へ」ボタンをクリックしてください。



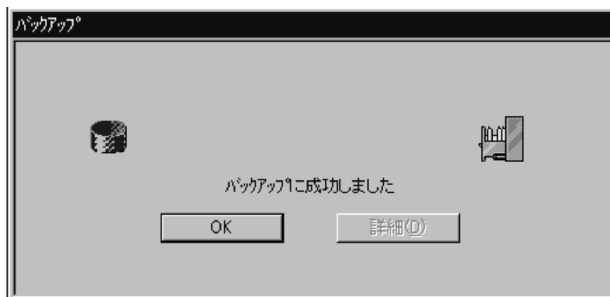
- 10.【バックアップウィザード - 確認】ボックスが表示されます。

5. HIRDB Control Manager で HIRDB を運用してみましょう

バックアップの条件を確認して、「実行」ボタンをクリックしてください。



11. バックアップに成功すると、【バックアップ】ボックスが表示されます。
「OK」ボタンをクリックしてください。



5.7 データベースを回復する

HiRDB Control Manager を使ったデータベースの回復では、回復前の HiRDB の終了や複写・回復専用起動が自動的に行われるため、それらの作業を行う必要はありません。

HiRDB Control Manager では、DAT に取得したバックアップからデータベースを回復するには、NetBackup などの付加プログラムが必要です。このマニュアルでは付加プログラムをインストールしていないので DAT を使った演習は行いません。

5.7.1 演習 1 (全 RD エリアをバックアップ取得時点に回復する)

データベースを格納しているディスクに障害が発生しました。全 RD エリアをバックアップ取得時点の状態に回復します。

(1) ディスクを交換します

障害が発生したディスクを交換します。この演習では実際に交換はしません。交換したと仮定して以降の作業を行います。

(2) pdfmkfs コマンドで HiRDB ファイルシステム領域を初期設定します

pdfmkfs コマンドで HiRDB ファイルシステム領域を初期設定します。pdfmkfs コマンドで指定するオプションは、RD エリア用の HiRDB ファイルシステム領域を前回作成したときと同じにしてください。システムジェネレータによって実行された pdfmkfs コマンドについては、/HiRDB/conf/pdgen_cmd.[HiRDB サーバ名] のファイルに記録されています。

この演習の場合、次のコマンドになります。

```
pdfmkfs -r -n 68 -l 400 -k SVR -e 100 -i /HiRDB/sysfiles
pdfmkfs -r -n 81 -l 100 -k SVR -e 100 -i /HiRDB/rdfiles
```

なお、この演習では実際に pdfmkfs コマンドを実行しません。実行したと仮定して以降の作業を行います。

(3) RD エリアの回復に必要なファイルを準備します

RD エリアの回復に必要なファイル (データベース回復ユーティリティの入力情報ファイル) は次のとおりです。

- システム単位で取得したバックアップファイル
「5.6 バックアップを取得する」の演習 1 で取得したバックアップファイルを使用しま

5. HiRDB Control Manager で HiRDB を運用してみましょう

す。バックアップファイル名は、/home/hirman/hirdb/pdcopy2/backup11 です。

(4) 全 RD エリアをバックアップ取得時点に回復します

手順

1. [リカバリ] - [ウィザード] を選択してください。



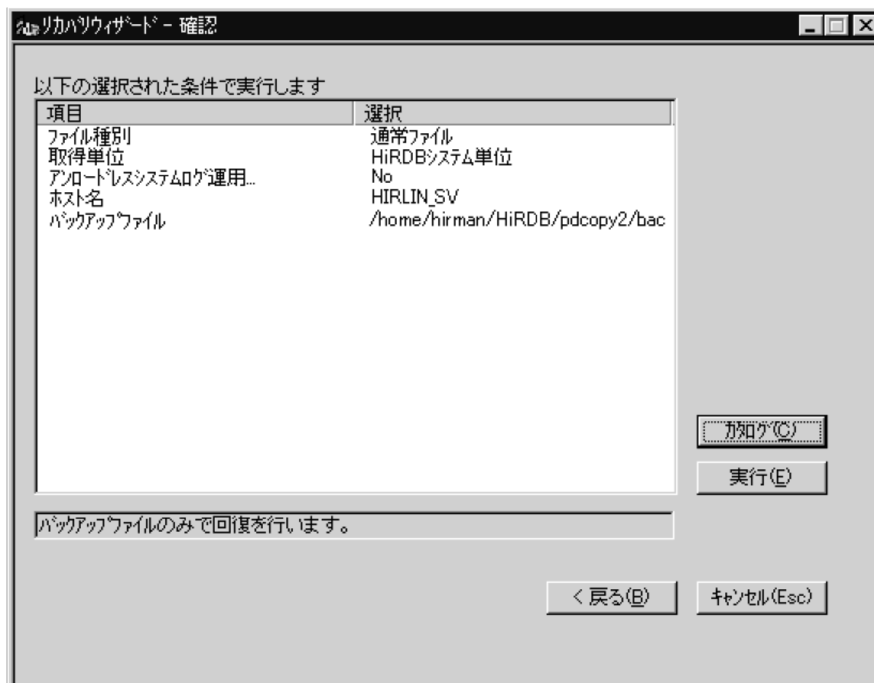
2. 【リカバリウィザード - 回復対象の指定】ボックスが表示されます。
各項目を次のように設定して、「次へ」ボタンをクリックしてください。
回復単位：HiRDB システム単位
HiRDB システム：HDBS (10.210.35.14_20293)

5. HiRDB Control Manager で HiRDB を運用してみましょう



3. 【リカバリウィザード - 確認】ボックスが表示されます。

設定した条件を確認して、問題がなければ「実行」ボタンをクリックしてください。リカバリ処理が始まります。



5.7.2 演習 2 (RD エリアを最新の状態に回復する)

ユーザ用 RD エリア (RDDATA10) を最新の状態に回復します。アンロードログファイルを使ってデータベースを回復する場合、ウィザードではなくカスタマイズでリカバリを行います。

(1) 必要なバックアップファイルとアンロードログファイルを特定します

(a) 必要なバックアップファイル

バックアップファイルには、「5.6 バックアップを取得する」の演習 2 で取得した RDDATA10 のバックアップを格納したバックアップファイルを使用します。バックアップファイル名は、/home/hirman/hirdb/pdcopy2/backup12 です。

留意事項

バックアップは、「5.6 バックアップを取得する」の演習 1 で取得した全 RD エリアのバックアップでもかまいません。全 RD エリアのバックアップからでも、特定の RD エリアを回復できます。

(b) 必要なアンロードログファイル

(a) のバックアップを取得したとき以降に出力されたシステムログを格納したアンロードログファイルが必要になります。バックアップを取得したとき以降に出力されたシステムログをすべてアンロードしてください。したがって、次に示すシステムログファイルをアンロードする必要があります。

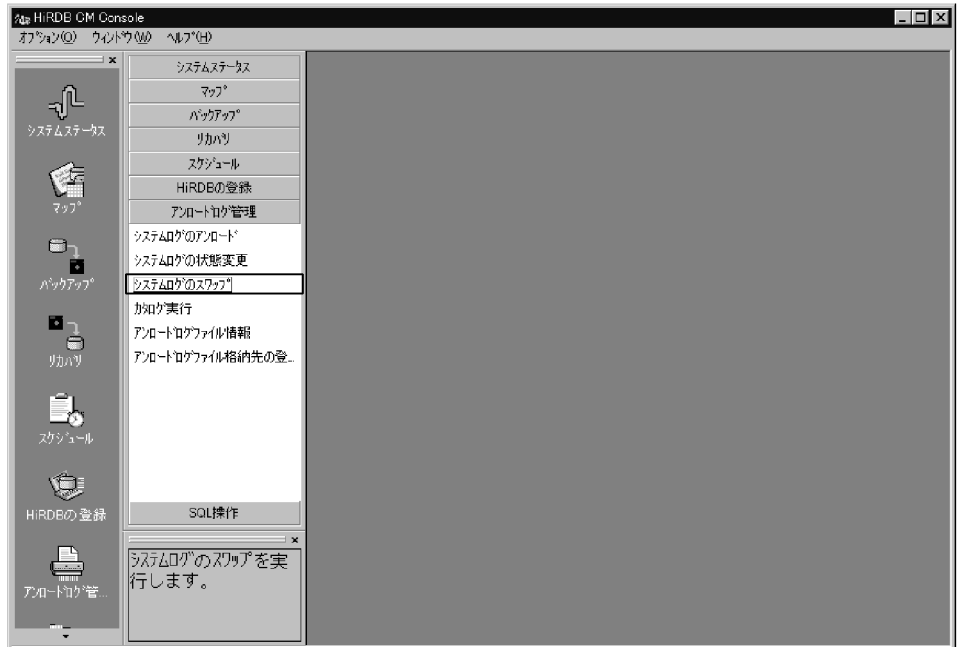
- 「アンロード待ち状態」のファイル
- 現用のファイル

(2) 現用ファイルをスワップします

手順

1. [アンロードログ管理] - [システムログのスワップ] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【システムログのスワップ】ボックスが表示されます。

各項目を次のように設定して、「実行」ボタンをクリックしてください。

HiRDB システム : HDBS (10.210.35.14_20293)

サーバ : sds1



3. 現用ファイルのスワップが終了したら、「キャンセル」ボタンをクリックしてください。

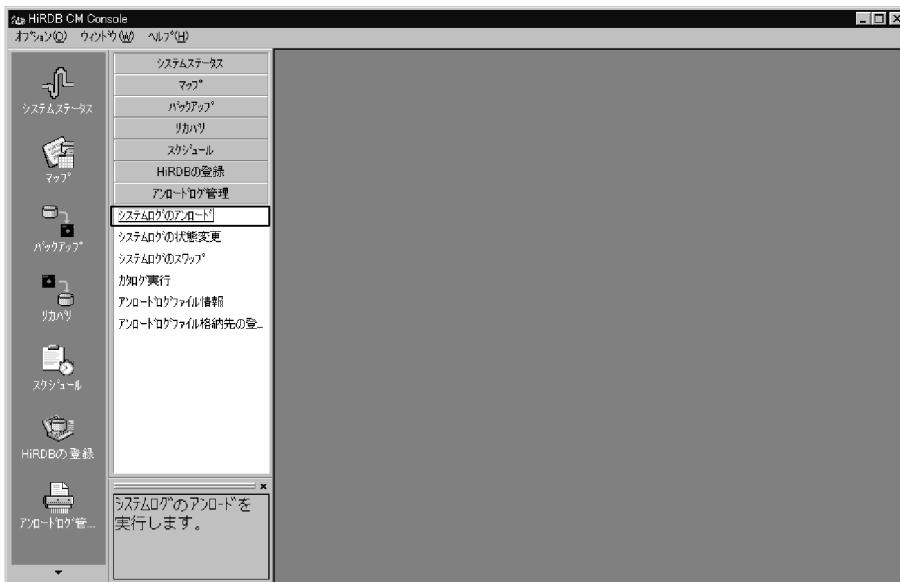
(3) 「ダウンロード待ち状態」のファイルをダウンロードします

「ダウンロード待ち状態」のファイルをダウンロードします。

5. HiRDB Control Manager で HiRDB を運用してみましょう

手順

1. [アンロードログ管理] - [システムログのアンロード] を選択してください。



2. 【システムログのアンロード】ボックスが表示されます。

各設定値を次のように指定してください。

HiRDB システム : HDBS (10.210.35.14_20293)

サーバ : sds1

アンロードログファイルの最大保持数 : 256

アンロードログファイルの種別 : 通常ファイル

アンロードログファイルの格納先 : /home/hirman/HiRDB/pdlogunld2

システムログファイル : アンロード状態が「アンロード待ち」になっているファイル (logfg01 ~ logfg04) を選択

5. HIRDB Control Manager で HIRDB を運用してみましょう

「アンロード待ち」状態のファイルを選択する。



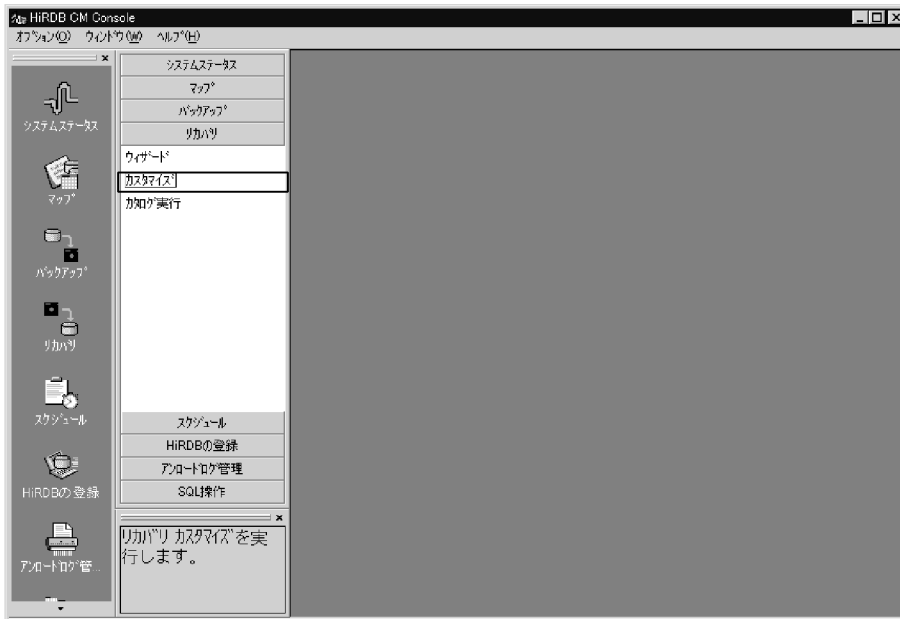
3. 選択したシステムログファイルがアンロードされます。

(4) RDDATA10 を回復します

手順

1. [リカバリ] - [カスタマイズ] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【リカバリ カスタマイズ】ボックスが表示されます。

「回復単位」タブ、および「ファイル種別」タブのページで次のように設定して、「確認画面へ」ボタンをクリックしてください。

回復単位

回復単位：RD エリア単位

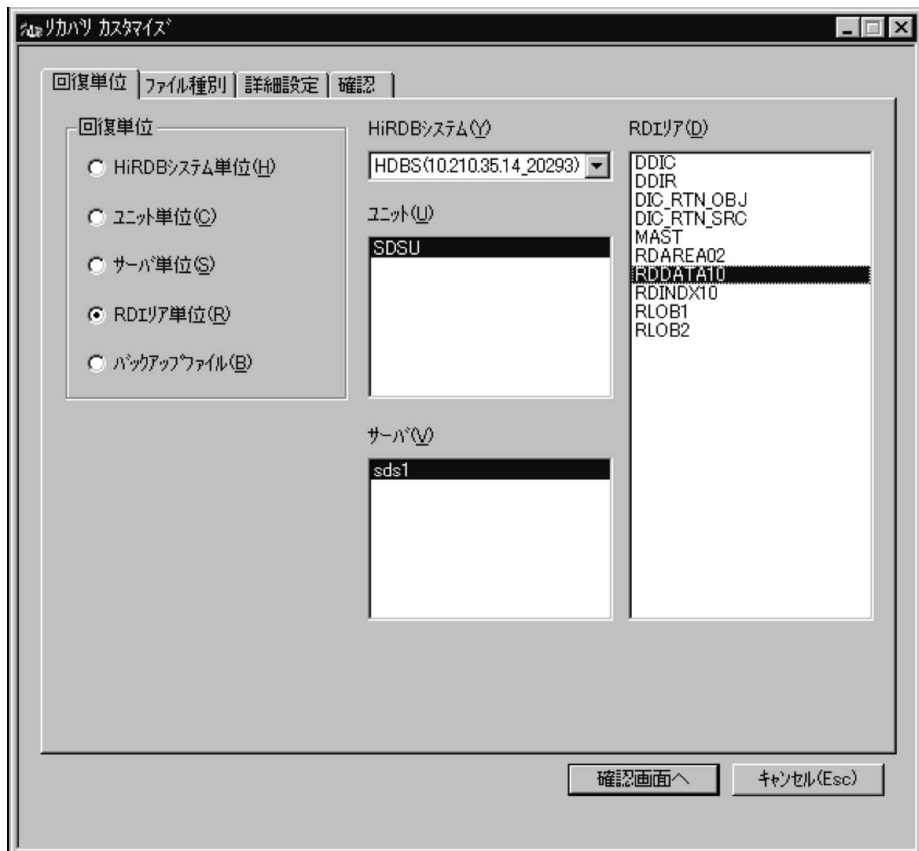
HiRDB システム：HDBS (10.210.35.14_20293)

ユニット：SDSU

サーバ：sds1

RD エリア：RDDATA10

5. HIRDB Control Manager で HIRDB を運用してみましょう



ファイル種別

ファイル種別：通常ファイル

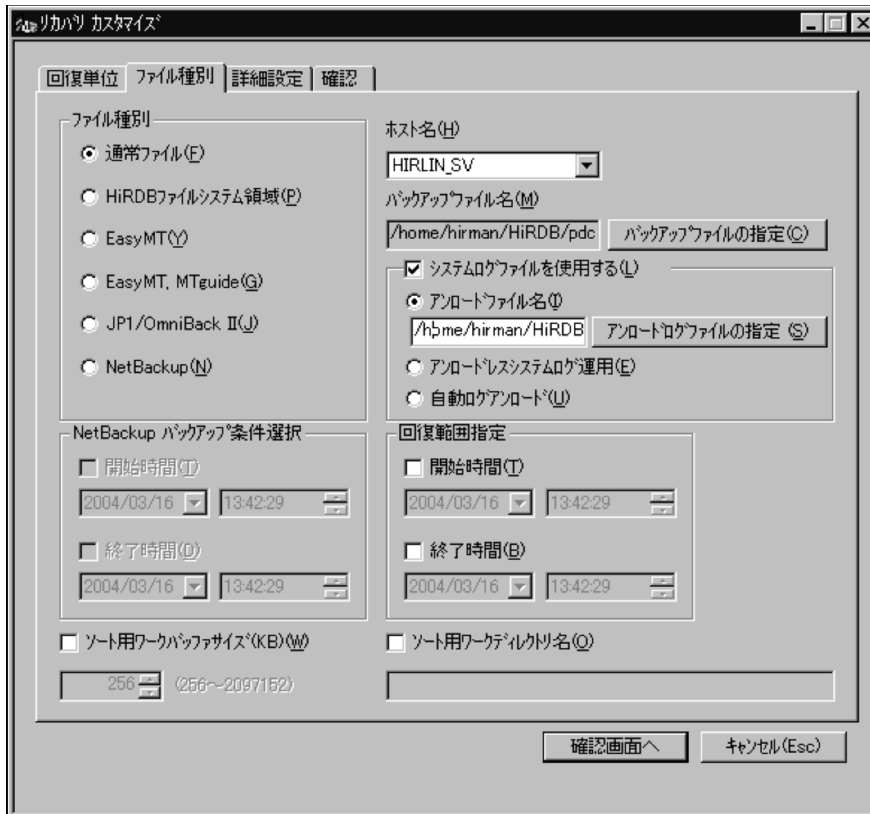
ホスト名：HIRLIN_SV

バックアップファイル名：/home/hirman/hirdb/pdcopy2/backup12

システムログファイルを使用する：チェック

アンロードログファイル名：logfg01 ~ logfg04

5. HIRDB Control Manager で HIRDB を運用してみましょう



3. 指定した条件を確認して問題なければ、「実行」ボタンをクリックしてください。
リカバリ処理が始まります。

5. HiRDB Control Manager で HiRDB を運用してみましょう



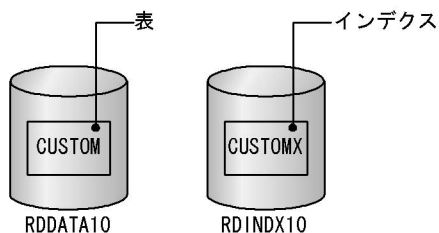
4. RDDATA10 が回復されます。

5.8 表を再編成する

5.8.1 演習（表を再編成する）

CUSTOM 表を再編成します。

- CUSTOM 表にはインデクス（CUSTOMX）が定義されています。
- CUSTOM 表はユーザ用 RD エリア（RDDATA10）に格納されています。
- インデクス（CUSTOMX）はユーザ用 RD エリア（RDINDX10）に格納されています。



「4.6 表を再編成する」の演習を行った方は、(1) ~ (2) の操作は必要ありません。すでに「4.6 表を再編成する」の演習で実施しています。(3) から始めてください。

(1) 表の再編成時に必要なディレクトリを準備します

次に示すディレクトリが表の再編成時に必要になります。

- アンロードデータファイルを作成するディレクトリ
- インデクス情報ファイルを作成するディレクトリ

アンロードデータファイルを作成するディレクトリとして、`/home/hirman/HiRDB/pdrorg` を作成してください。さらに、インデクス情報ファイルを作成するディレクトリとして、`/home/hirman/HiRDB/idxdir` を作成してください。

(2) データベース再編成ユーティリティの制御情報ファイルを作成します

制御情報ファイルを `/home/hirman/HiRDB/pdrorg/seigo.01.txt` に作成してください。

```
unload /home/hirman/HiRDB/pdrorg/unld01 .....1
idxwork /home/hirman/HiRDB/idxdir .....2
sort /tmp .....3
```

〔説明〕

1. アンロードデータファイル名を指定します。

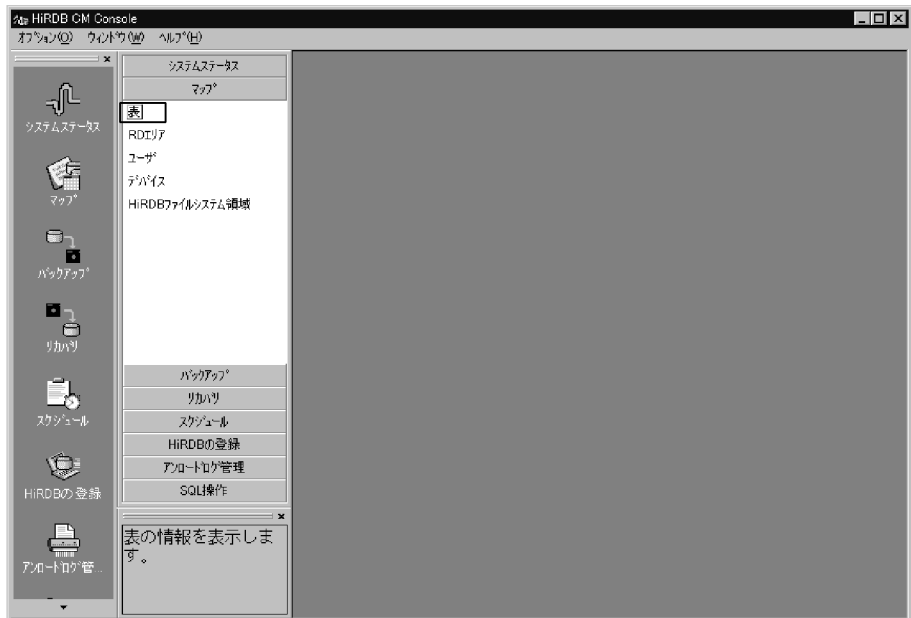
/home/hirman/HiRDB/pdrorg ディレクトリ下に unld01 ファイルが作成されま
す。/home/hirman/HiRDB/pdrorg は (1) で作成したディレクトリです。

2. インデックスの一括作成時に使用する作業用ワークディレクトリの名称を指定しま
す。ここで指定したディレクトリ下にインデックス情報ファイルが作成されます。/
home/hirman/HiRDB/idxdir は、(1) で作成したディレクトリです。
3. ソート用ワークディレクトリの名称を指定します。

(3) CUSTOM 表を再編成します

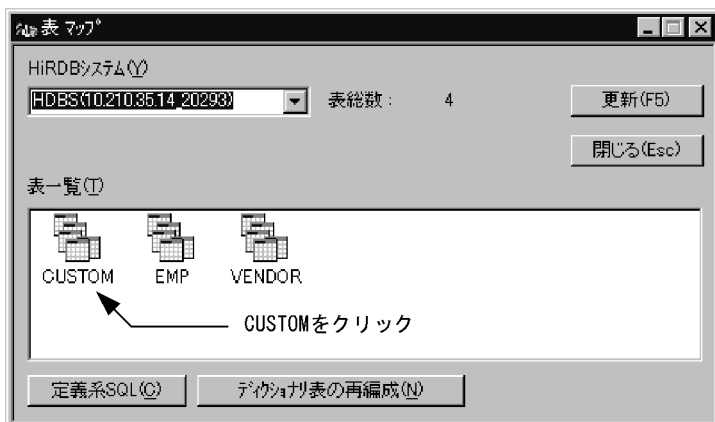
手順

1. [マップ] - [表] を選択してください。

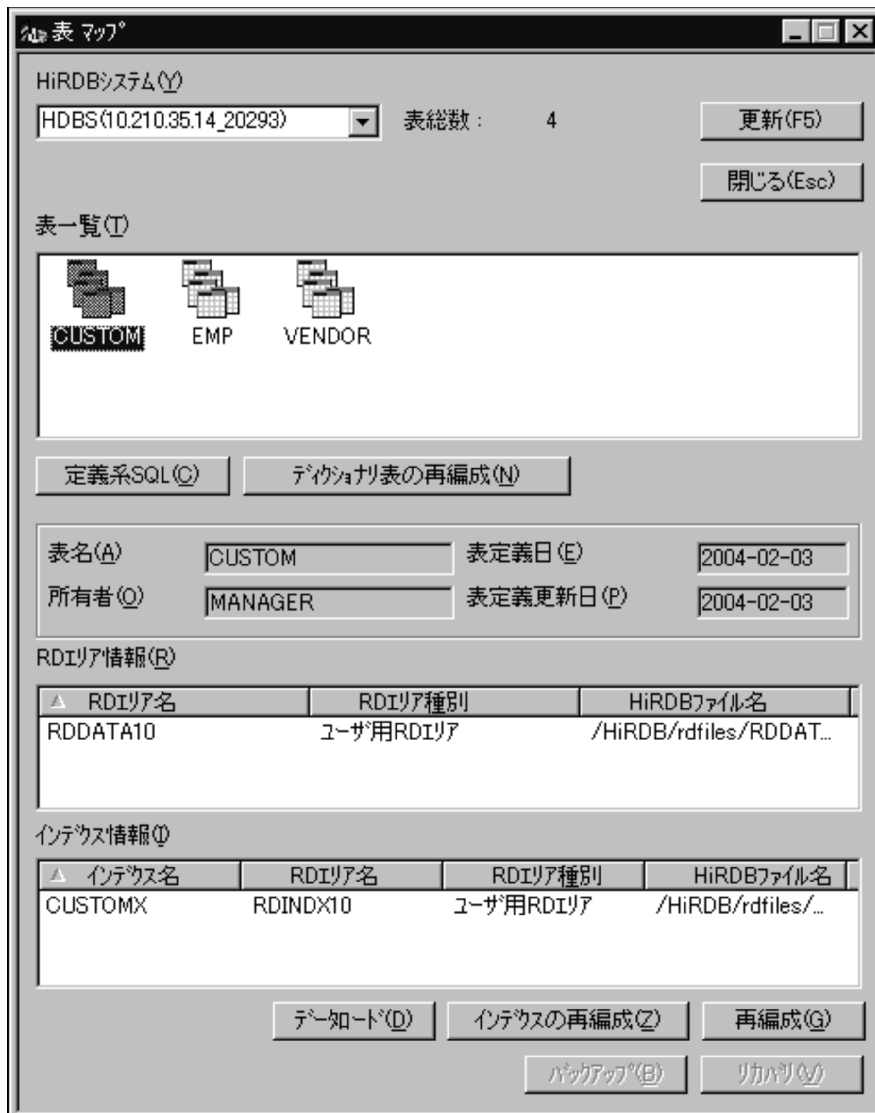


2. 【表 マップ】ボックスが表示されます。
HiRDB システムに「HDBS (10.210.35.14_20293)」を選択したあと、表一覧の
「CUSTOM」をクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



3. 【表 マップ】ボックスが拡大されて、CUSTOM 表の情報が表示されます。[再編成] ボタンをクリックしてください。



4. 【データベース再編成ユーティリティの実行】ボックスが表示されます。ここで再編成の情報を次のように指定して、「詳細設定」ボタンをクリックしてください。
- パスワード：MANAGER
 処理方針：表のアンロード&リロード
 制御情報ファイル名：/home/hirman/HiRDB/pdrorg/seigy01.txt

5. HIRDB Control Manager で HIRDB を運用してみましょう



5. 【データベース再編成ユーティリティの実行】ボックスで [詳細設定] をクリックすると、【データベース再編成ユーティリティ 詳細設定】ダイアログが表示されます。ここで表の再編成時の条件を指定します。
アンロード指定で「クラスタ順」を指定して、「OK」をクリックしてください。



6. 【データベース再編成ユーティリティの実行】ボックスで「実行」をクリックしてください。

5. HIRDB Control Manager で HIRDB を運用してみましょう



7. 【データベース再編成ユーティリティ】ダイアログに実行結果が表示されます。
「閉じる」ボタンをクリックしてください。



5.9 RD エリアを拡張する

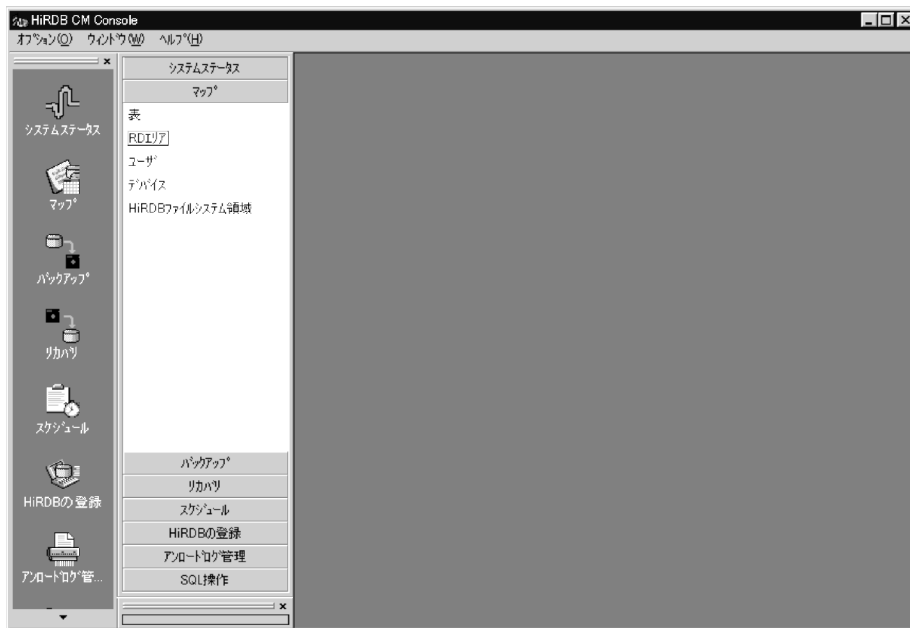
5.9.1 演習（RD エリアを拡張する）

ユーザ用 RD エリア（RDDATA10）を拡張します。

（1）RDDATA10 の残容量を調べます

手順

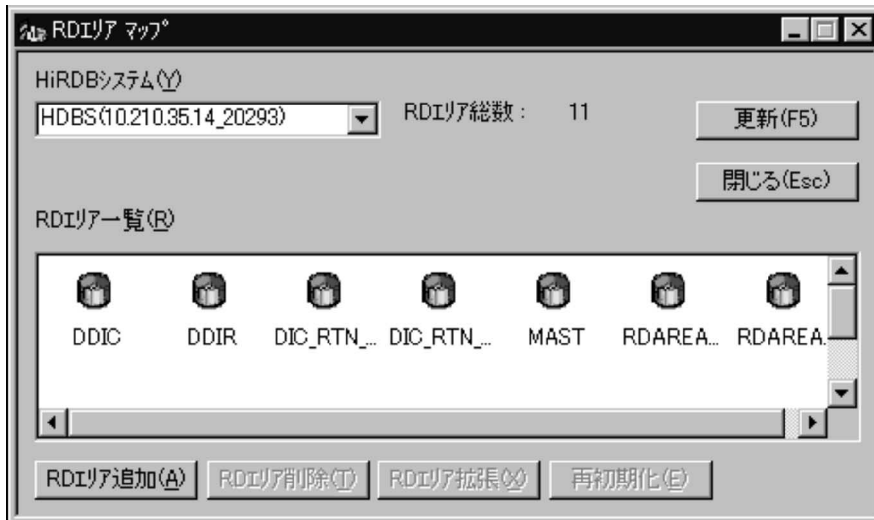
1. 【マップ】 - 【RD エリア】を選択してください。



2. 【RD エリア マップ】ボックスが表示されます。

HiRDB システムに「HDBS (10.210.35.14_20293)」を選択したあと、RD エリア一覧の「RDDATA10」をクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



3. 【RD エリア マップ】ボックスが拡大されて、RDDATA10 の情報が表示されます。

5. HiRDB Control Manager で HiRDB を運用してみましょう



〔説明〕

この例では、全セグメント数が 97 で、使用率が 4% になっています。

(2) RD エリアを拡張します

手順

1. [RD エリア拡張] をクリックしてください。



2. 【RD エリアの拡張】ボックスが表示されます。

追加する HiRDB ファイルを次のように指定して、「実行」ボタンをクリックしてください。

HiRDB ファイルシステム領域名: /HiRDB/rfiles

HiRDB ファイル名: rddata10b

セグメント数: 10

5. HiRDB Control Manager で HiRDB を運用してみましょう



〔説明〕

「HiRDB ファイルシステム領域名」「HiRDB ファイル名」「セグメント数」を入力したあとに「追加」ボタンをクリックすると、「HiRDB ファイルシステム領域名 / HiRDB ファイル名」に情報が入力されます。

HiRDB ファイルシステム領域名が長くて「HiRDB ファイルシステム領域名」欄では確認できない場合、「追加」ボタンをクリックしてください。選択した HiRDB ファイルシステム領域名が「HiRDB ファイルシステム領域名 / HiRDB ファイル名」欄に表示されます。HiRDB ファイルシステム領域名に誤りがあれば、「削除」ボタンで削除してから選択し直してください。

3. RD エリア拡張のメッセージが出力されます。「OK」ボタンをクリックしてください。

(3) RDDATA10 が拡張されたことを確認します

【RD エリア マップ】ボックスの「更新」ボタンをクリックしてから、RD エリア一覧の「RDDATA10」をクリックしてください。RDDATA10 の情報が表示されます。

5. HiRDB Control Manager で HiRDB を運用してみましょう



〔説明〕

全セグメント数が、97 から 107 に変わっています。したがって、RDDATA10 が拡張されています。

5.10 RD エリアを追加する

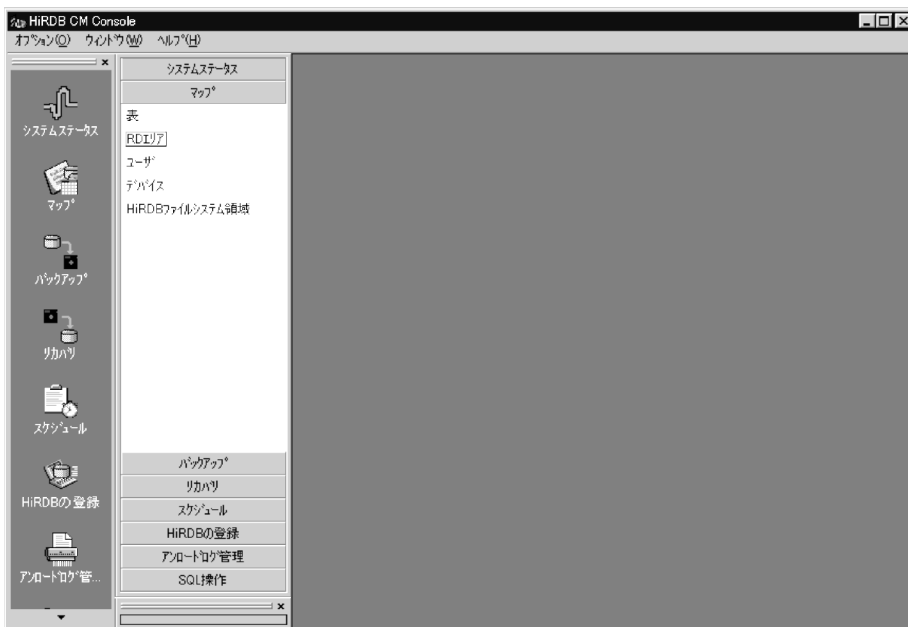
5.10.1 演習（RD エリアを追加する）

新しい HiRDB ファイルシステム領域（area02）とユーザ用 RD エリア（RDAREA02）を作成して、そこに PC 割り当て表（表名：PCLIST）を作成します。

（1）RDAREA02 を追加します

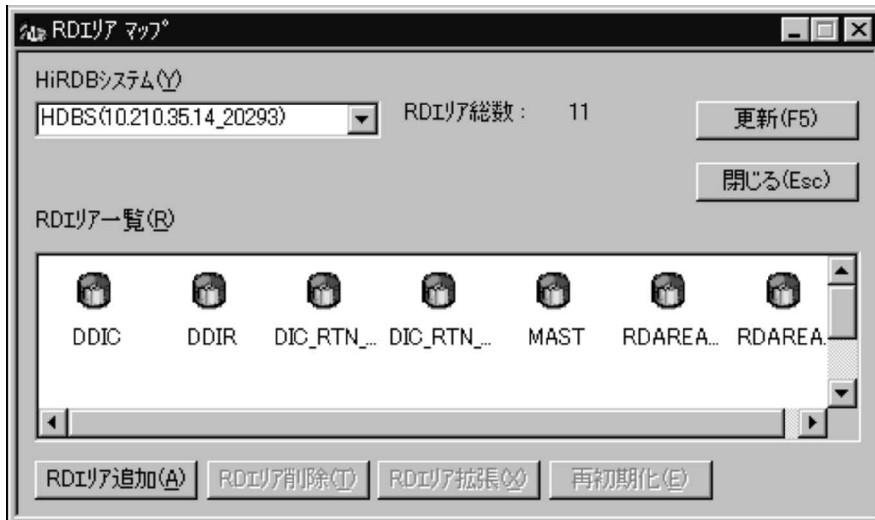
手順

1. [マップ] - [RD エリア] を選択してください。



2. 【RD エリア マップ】ボックスが表示されます。

HiRDB システムに「HDBS (10.210.35.14_20293)」を選択したあと、「RD エリア追加」ボタンをクリックしてください。



3. 【RD エリアの追加】ボックスが表示されます。

追加する HiRDB ファイルを次のように指定して、「システム領域の初期設定」ボタンをクリックしてください。

RD エリア名：RDAREA02

サーバ：sds1

種類：ユーザ用 RD エリア

一時割り当てするグローバルバッファ名：RDDATA10

ページ長：4096

セグメントサイズ：10

5. HiRDB Control Manager で HiRDB を運用してみましょう



〔説明〕

RDAREA02 に割り当てる HiRDB システム領域「area02」は新規に作成するので、まだ HiRDB ファイルシステム領域名から選択できません。「システム領域の初期設定」でシステム領域「area02」を登録する必要があります。

4. 【HiRDB ファイルシステム領域の初期設定】ダイアログが表示されます。

次のように指定して、「登録」ボタンをクリックしてください。

HiRDB ファイルシステム領域名：/home/hirman/HiRDB/rdarea/area02

HiRDB ファイルシステム領域サイズ：20

作成する HiRDB ファイル数：1

HiRDB ファイルの増分回数：5

HiRDB ファイルシステム領域を初期化する：チェック

使用目的：すべて（ユティリティを除く）

ファイル種別：通常ファイル

5. HiRDB Control Manager で HiRDB を運用してみましょう

HiRDBファイルシステム領域の初期設定

実行する操作と対象の指定

HiRDBシステム(Y)
HDBS (10.210.35.14_20293)

ホスト(H)
HIRLIN_SV

サーバ(S)
sds1

HiRDBファイルシステム領域名 Note:絶対パスで指定してください
/home/hirman/HiRDB/rdarea/area02

参照(F)

ファイル格納用HiRDBファイルシステム領域の設定

HiRDBファイルシステム領域サイズ(N) 20 MB(1~1048575)

作成するHiRDBファイル数(L) 1 個(1~4096)

HiRDBファイルの増分回数(E) 5 回(0~60000)

HiRDBファイルシステム領域を初期化する(O)

使用目的

すべて(ユティリティを除く)(A)

RDエリア外用RDエリアを除く(R)

システムウオッチポイントのマスターファイル(W)

リスト用RDエリア・作業表用ファイル(W)

ユティリティ用(U)

ファイル種別

通常ファイル(Q)

キャラクタ型スペシャルファイル(Q)

登録 キャンセル(Esc)

〔説明〕

HiRDB ファイルシステム領域名

新しく作成する HiRDB ファイルシステム領域名を指定します。

HiRDB ファイルシステム領域を初期化する

「HiRDB ファイルシステム領域サイズ」で指定したサイズ分の領域を最初から初期化する場合に指定します。このオプションを選択しない場合、HiRDB ファイルシステム領域の管理情報だけを作成します。

5. 【RD エリアの追加】ボックスで追加する HiRDB ファイルを次のように指定して、「実行」ボタンをクリックしてください。

HiRDB ファイルシステム領域名 : /home/hirman/HiRDB/rdarea/area02

HiRDB ファイル名 : rdarea02

セグメント数 : 10

5. HiRDB Control Manager で HiRDB を運用してみましょう



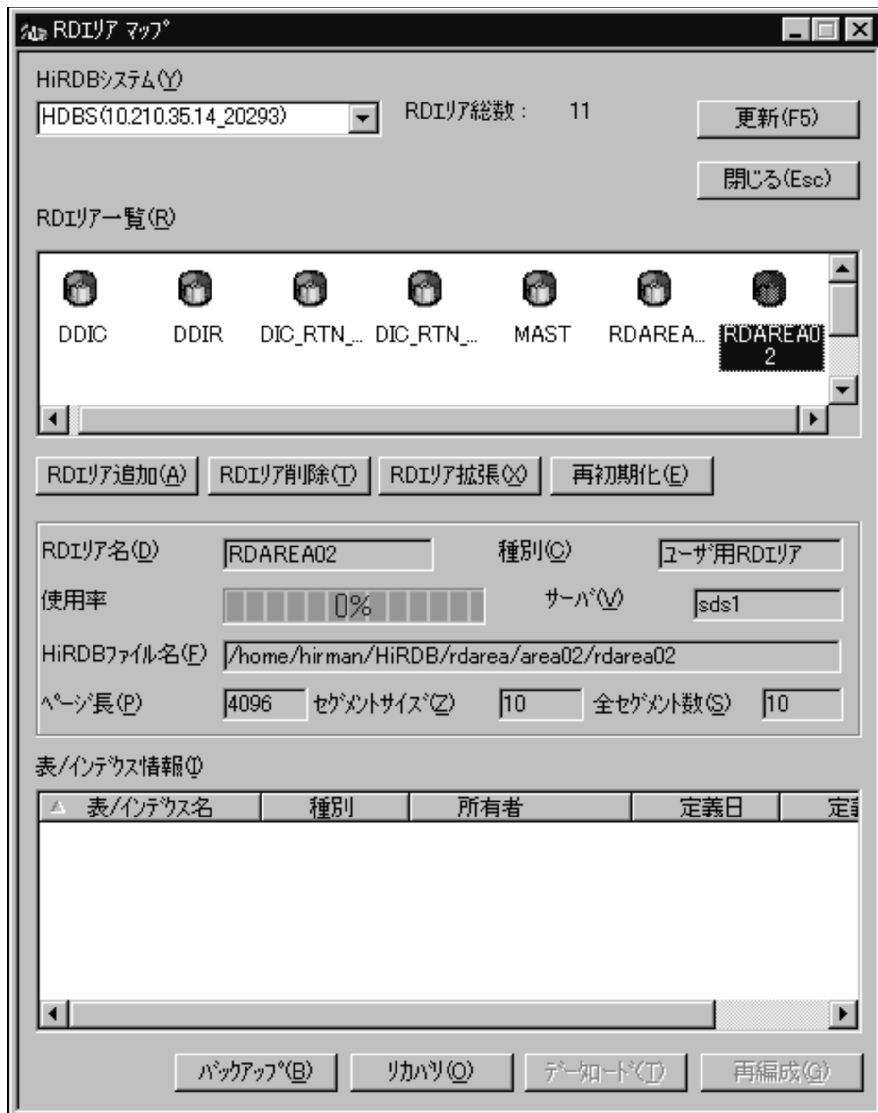
〔説明〕

「HiRDB ファイルシステム領域名」「HiRDB ファイル名」「セグメント数」を入力したあとに「追加」ボタンをクリックすると、「HiRDB ファイルシステム領域名 / HiRDB ファイル名」に情報が入力されます。

6. RD エリア追加のメッセージが出力されます。「OK」ボタンをクリックしてください。

(2) RDAREA02 が追加されたことを確認します

【RD エリア マップ】ボックスの「更新」ボタンをクリックしてから、RD エリア一覧の「RDAREA02」をクリックしてください。RDAREA02 の情報が表示されます。



〔説明〕

「(1) RDAREA02 を追加します」で指定したとおりに RD エリアが作成されています。使用率は 0% です。

(3) RDAREA02 に表を定義します

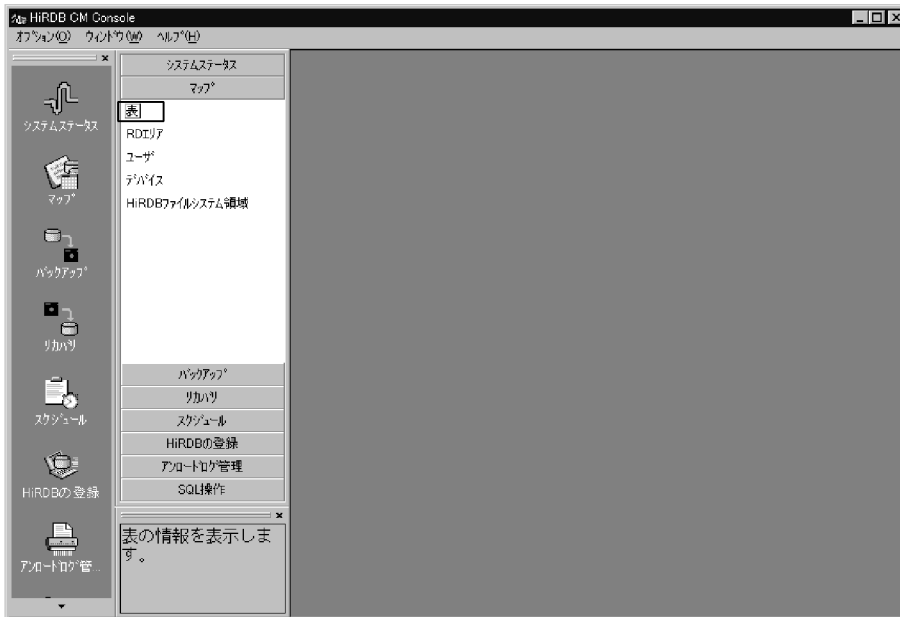
手順

1. CREATE TABLE をテキスト形式のファイル (\$HOME/HiRDB/pddef/c_table2.txtC:¥hirdb¥pddef¥c_table2.txt) に作成してください。

5. HIRDB Control Manager で HIRDB を運用してみましょう

```
CREATE FIX TABLE PCLIST (PCID CHAR(8), USR CHAR(12), SINCE INTEGER)
IN RDAREA02;     ここに改行を入れてください。
```

2. [マップ] - [表] を選択してください。



3. 【表 マップ】ボックスが表示されます。
「定義系 SQL」ボタンをクリックしてください。



4. 【定義系 SQL の実行】ボックスが表示されます。
実行する SQL と実行者を次のように指定して、「実行」ボタンをクリックしてください。
認可識別子 : MANAGER

5. HiRDB Control Manager で HiRDB を運用してみましょう

パスワード : MANAGER

ファイルを指定 : /home/hirman/HiRDB/pddef/c_table2.txt



5. SQL の実行結果を示すメッセージボックスが表示されます。「閉じる」ボタンをクリックしてください。
6. 「サーバの管理情報を更新しますか？」というメッセージが表示されます。「はい」ボタンをクリックしてください。

(4) PC 割り当て表が定義されたかを確認します

【RD エリア マップ】ボックスで [RDAREA02] をクリックしてください。定義された PCLIST が表示されます。

5. HiRDB Control Manager で HiRDB を運用してみましょう



(5) データを挿入してみましょう

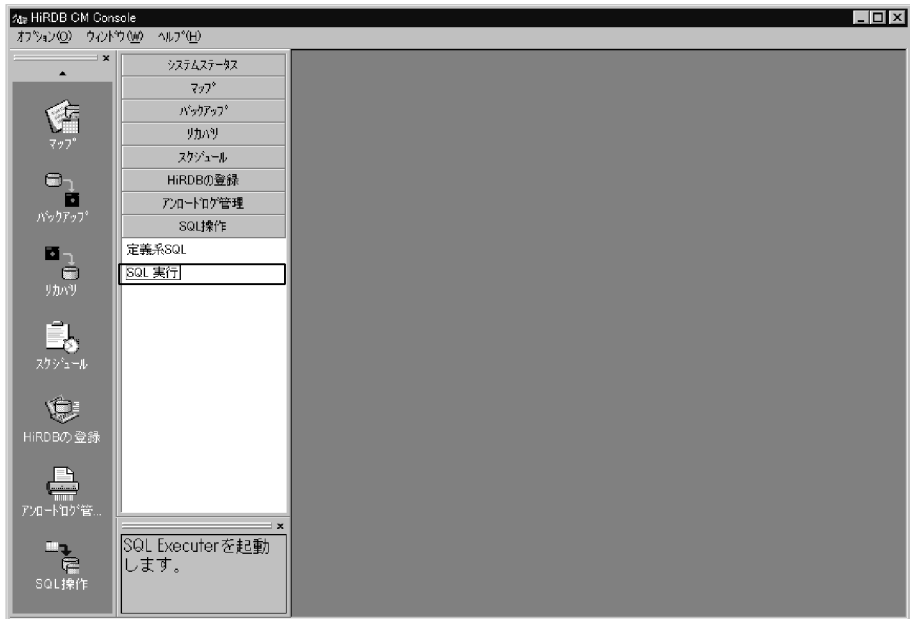
HiRDB Control Manager - Console をインストールしているマシンに、HiRDB SQL Executer および HiRDB/ クライアントがインストールされている場合、HiRDB Control Manager - Console から HiRDB SQL Executer を起動することができます。

ここでは、HiRDB Control Manager - Console から HiRDB SQL Executer を起動して、データを挿入する手順を説明します。

手順

1. [SQL 操作] - [SQL 実行] を選択してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【SQL の実行】ボックスが表示されます。

HiRDB SQL Executer による HiRDB サーバへの接続情報を次のように指定して、「実行」ボタンをクリックしてください。

HiRDB システム：「HDBS (10.210.35.14_20293)」

認可識別子：MANAGER

パスワード：MANAGER



3. HiRDB SQL Executer 上で INSERT 文を実行し、データを挿入してください。

【SQL 入力画面】

5. HiRDB Control Manager で HiRDB を運用してみましょう

```
INSERT INTO PCLIST VALUES('PC0001','TARO',1995);  
INSERT INTO PCLIST VALUES('PC0011','HANAKO',2000);  
INSERT INTO PCLIST VALUES('PC0012','ICHIRO',1998);
```

【検索結果画面】

```
KFPX27010-I          1 rows inserted  
KFPX27010-I          1 rows inserted  
KFPX27010-I          1 rows inserted
```

〔説明〕

データが挿入されました。

(6) PC 割り当て表を検索してみましょう

データが正しく挿入されているか確認するため、表を検索します。HiRDB SQL Executer 上で次の SQL 文を実行してください。

【SQL 入力画面】

```
SELECT * FROM PCLIST;
```

【検索結果画面】

```
PCID      USR      SINCE  
-----  
PC0001    TARO      1995  
PC0011    HANAKO    2000  
PC0012    ICHIRO    1998  
  
KFPX27010-I          3 rows selected
```

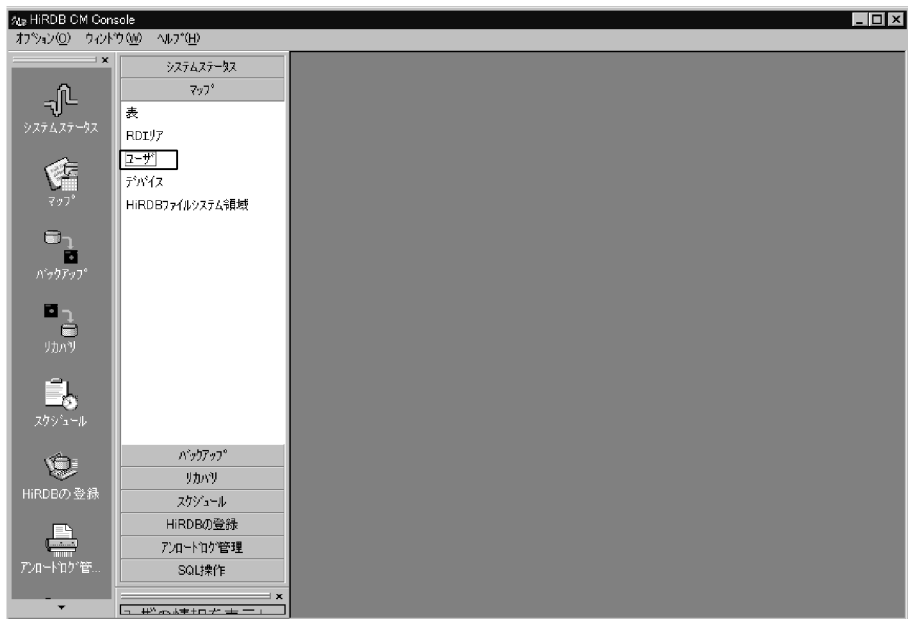

5.11 ユーザ権限を与える（ユーザを登録する）

5.11.1 演習 1（新規ユーザを登録する）

新規の HiRDB ユーザ（ユーザ名「USER02」、パスワード「PASS02」）を登録します。

手順

1. [マップ] - [ユーザ] を選択してください。



2. 【ユーザ マップ】ボックスが表示されます。

HiRDB システムに「HDBS (10.210.35.14_20293)」を選択したあと、「ユーザ追加」ボタンをクリックしてください。



5. HiRDB Control Manager で HiRDB を運用してみましょう

3. 【ユーザの追加】ボックスが表示されます。

ユーザ情報を次のように設定して、「実行」ボタンをクリックしてください。

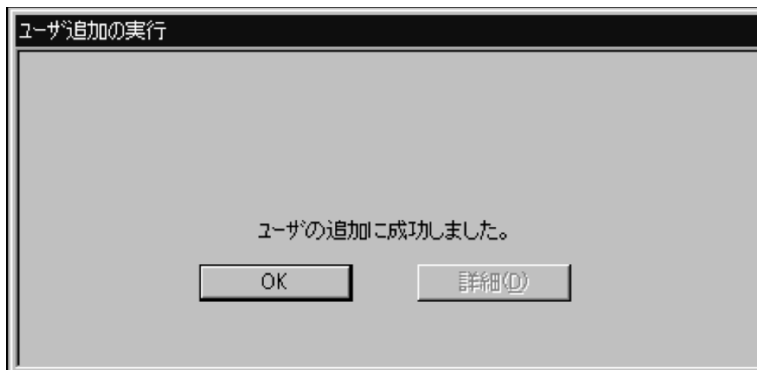
ユーザ名：USER02

パスワードの設定 / 変更：PASS02



4. 【ユーザ追加の実行】ボックスが表示され、ユーザが作成されます。

「OK」をクリックしてください。



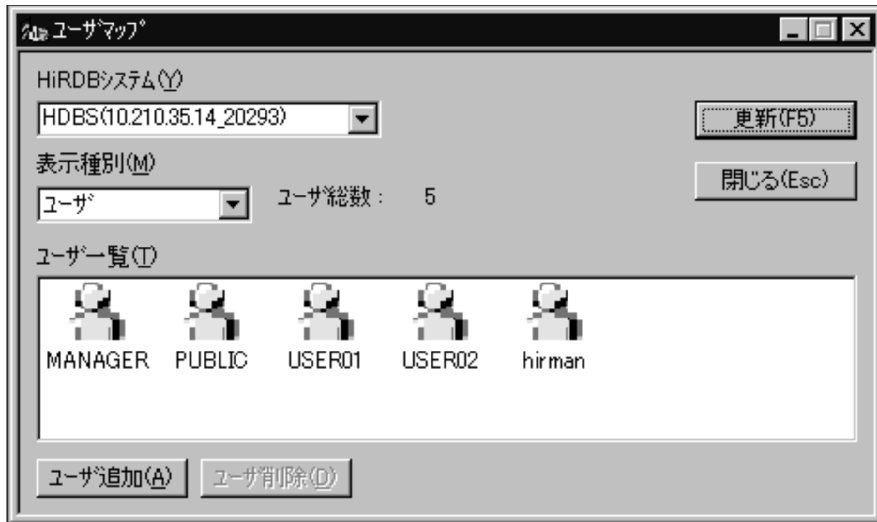
5.11.2 演習 2 (権限を追加する)

HiRDB ユーザ (ユーザ名 「 USER01 」) に DBA 権限を与えます。

手順

1. 【ユーザマップ】ボックスで,[USER01] をクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



2. 【ユーザマップ】ボックスが拡大されて、USER01 の情報が表示されます。
「ユーザ権限」ボタンをクリックしてください。

5. HIRDB Control Manager で HIRDB を運用してみましょう

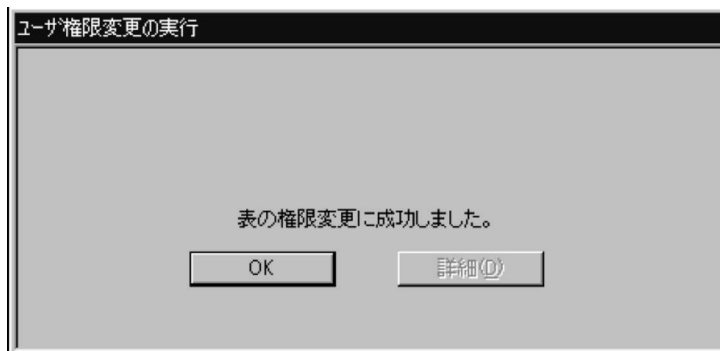


3. 【ユーザー権限】ボックスが表示されます。パスワードに「PASS02」を入力して、
[DBA 権限] をチェックして「実行」をクリックしてください。

5. HiRDB Control Manager で HiRDB を運用してみましょう



4. 【ユーザ権限変更の実行】ボックスが表示され、USER01 に DBA 権限が与えられます。



6

クライアントを接続してみ ましょう

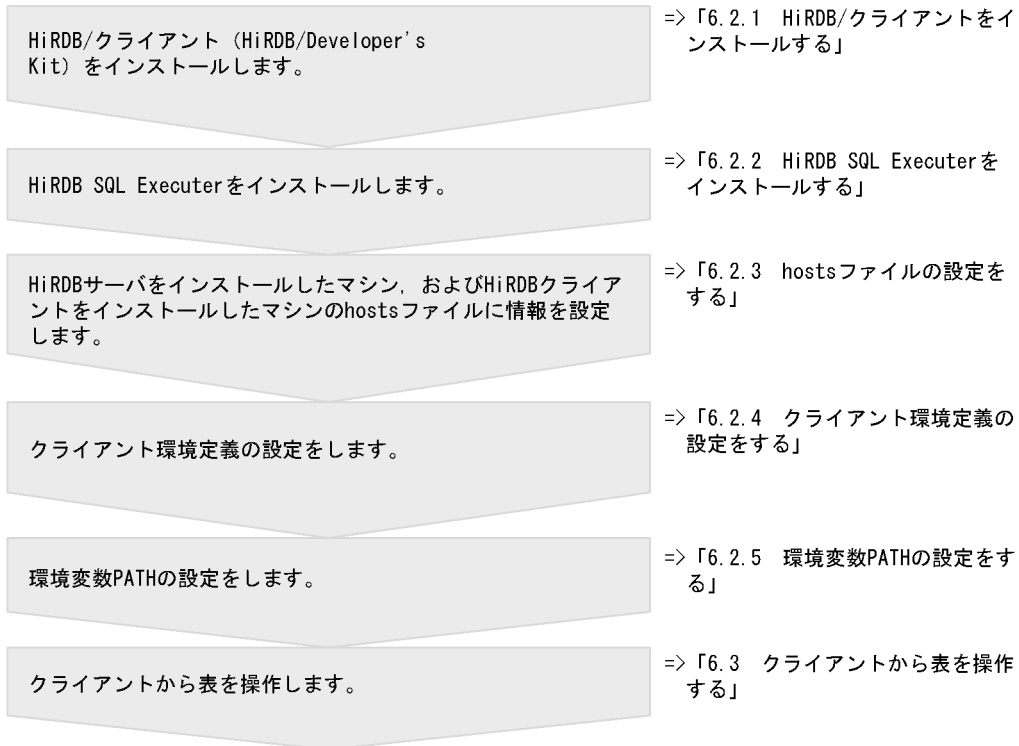
この章では、サーバとは別の HP-UX マシンにクライアントをインストールし、そのクライアントから表をアクセスする方法について説明します。

-
- 6.1 この章を読むに当たって
 - 6.2 クライアントの環境設定をする
 - 6.3 クライアントから表を操作する
-

6.1 この章を読むに当たって

この章では、クライアントの接続について説明しています。クライアントの接続手順を次に示します。

図 6-1 クライアントの接続手順



6.2 クライアントの環境設定をする

クライアントの環境設定手順を次に示します。

1. HiRDB/クライアントのインストール
2. HiRDB SQL Executer のインストール
3. hosts ファイルの設定
4. クライアント環境定義の設定
5. 環境変数 PATH の設定

6.2.1 HiRDB/クライアントをインストールする

HiRDB/クライアント (HiRDB/Developer's Kit) をインストールします。HiRDB/シングルサーバをインストールしたマシンではない、別のマシンにインストールします。「7. アプリケーションを作成してみましょう」を実施しないときは、インストールする HiRDB/クライアントは HiRDB/Run Time でもかまいません。

なお、HiRDB/クライアントが動作する OS は HP-UX を仮定します。

インストール手順を次に示します。

インストール手順

1. CD-ROM をマウントしてください (以降、マウント先を /cdrom とします)。
 2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してください。
- 次の画面が表示されます。

```

Hitachi PP Installer  03-00

L) List Installed Software.
I) Install Software.
D) Delete Software.
Q) Quit.

Select Procedure ===>

+-----+
CAUTION!
YOU SHALL INSTALL AND USE THE SOFTWARE PRODUCT LISTED IN THE
"List Installed Software." UNDER THE TERMS AND CONDITION OF
THE SOFTWARE LICENSE AGREEMENT ATTACHED TO SUCH SOFTWARE PRODUCT.
+-----+

All Rights Reserved, Copyright (C) 1994, 2003, Hitachi, Ltd.

```

3. < I > キーを押下してください。
- インストールする PP の選択画面が表示されます。

6. クライアントを接続してみましょう

```
PP-No.          VR      PP-NAME
001 P-1B62-1C71 0700/G HiRDB/Developer's Kit Version 7

F) Forward B) Backward J) Down K) Up Space) Select/Unselect I) Install Q) Quit
```

4. <スペース>キーを押下してください。

HiRDB/Developer's Kit Version 7 の行頭に、その PP を選択したことを示す「<@>」が表示されます。

```
PP-No.          VR      PP-NAME
<@>001 P-1B62-1C71 0700/G HiRDB/Developer's Kit Version 7

F) Forward B) Backward J) Down K) Up Space) Select/Unselect I) Install Q) Quit
```

5. <I>キーを押下してください。

インストールが開始されます。インストールが終了すると、「Installation completed.」のメッセージが表示されます。

6. <Q>キーを2回押下してください。

6.2.2 HiRDB SQL Executer をインストールする

クライアントから会話形式で SQL を発行するため、HiRDB SQL Executer を使用しません。

HiRDB SQL Executer のインストール手順を次に示します。

1. CD-ROM をマウントしてください (以降、マウント先を /cdrom とします)。
2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してください。
3. 「Install Software」から「HiRDB SQL Executer」を選択して、インストールしてください。

6.2.3 hosts ファイルの設定をする

HiRDB/クライアントをインストールしたマシンの、/etc/hosts ファイルに、自マシン、および HiRDB サーバをインストールしたマシンの IP アドレスおよびホスト名を設定します。すでに情報が hosts ファイルに設定されている場合、または DNS サーバを使用する場合には、不要です。

(hosts ファイルの設定例)

```

:
10.210.35.14      HiRLIN_SV.soft.hitachi.co.jp HiRLIN_SV
10.210.35.15      HiRLIN_CL.soft.hitachi.co.jp HiRLIN_CL
:

```

6.2.4 クライアント環境定義の設定をする

クライアントの環境変数に、接続する HiRDB/シングルサーバの情報、およびクライアントのユーザ名/パスワードを設定します。

C:\¥Windows ディレクトリ下の HiRDB.ini ファイルに次の情報を設定してください。

なお、クライアント環境定義について知りたい場合は、マニュアル「HiRDB Version 7 UAP 開発ガイド (UNIX(R)/Windows(R) 用)」を参照してください。

```

PDHOST=HiRLIN_SV
PDNAMEPORT=20000
PDUSER=USER01/PASS01
PDSQLLICENSE=YES

```

[説明]

```
PDHOST=HiRLIN_SV
```

接続する HiRDB のホスト名 (システム共通定義の pdstart オペランドの -x の指定値)

```
PDNAMEPORT=20000
```

接続する HiRDB のポート番号 (システム共通定義の pd_name_port オペランドの指定値)

```
PDUSER=USER01/PASS01
```

HiRDB に接続するときのユーザ名、およびパスワード

```
PDSQLLICENSE=YES
```

HiRDB SQL Executer を使用するための設定

注

6. クライアントを接続してみましょう

PDUSER のユーザ名、およびパスワードは、「4.9.2 演習 1（新規ユーザを登録する）」の演習問題で権限を与えられたユーザを想定しています。

6.2.5 環境変数 PATH の設定をする

コマンド、およびユーティリティを実行するために、クライアントの環境変数 PATH に次のディレクトリを追加してください。

```
/opt/pdsql:$PATH
```

6.3 クライアントから表を操作する

「3. 表を操作してみましょう」ですでに表を操作してみましたが、今度は別マシンのクライアントから表を操作してみます。表を操作するには HiRDB SQL Executer を使用します。

ここでは、ユーザ名「USER01」で HiRDB に CONNECT します。USER01 には、CONNECT 権限、スキーマ定義権限、および CUSTOM 表に対する SELECT 権限が与えられています。

なお、表の操作について知りたい場合は、マニュアル「HiRDB Version 7 SQL リファレンス (UNIX(R)/Windows(R) 用)」を参照してください。

6.3.1 演習 1 (表を定義する)

次の列から構成される ZAIKO 表を定義します。

- SHOHIN_CD (CHAR(4))
- SHOHIN_NAME (CHAR(8))
- COLOR (CHAR(5))
- TANKA (INTEGER)
- ZAIKORYO (INTEGER)

【SQL 文】

表を定義する前に、まずスキーマを定義します。

```
CREATE SCHEMA;
```

"KFPX27001-I Processing of SQL completed" と表示されたら、次に表を定義します。

```
CREATE TABLE ZAIKO  
(SHOHIN_CD CHAR(4), SHOHIN_NAME CHAR(8), COLOR CHAR(5),  
TANKA INTEGER, ZAIKORYO INTEGER) IN RDDATA10;
```

"KFPX27001-I Processing of SQL completed" と表示されると、表が定義されています。

【実行結果】

実際に ZAIKO 表が定義されているかどうかを確認します。次の SQL を実行してください。

6. クライアントを接続してみましょう

```
TABLES USER01;
```

実行すると、次の結果が表示されます。

```
TABLE_SCHEMA TABLE_NAME          N_COLS N_INDEX CREATE_TIME
-----
USER01        ZAIKO                5       0 20040204203756
KFPX27010-I          1 rows selected
```

6.3.2 演習 2 (表へ行を挿入する)

定義した ZAIKO 表に、次の行データを挿入します。
'777Z', 'boots', 'black', 10000, 3

【SQL 文】

```
INSERT INTO ZAIKO VALUES('777Z', 'boots', 'black', 10000, 3);
```

【実行結果】

正常に実行された場合、"KFPX27001-I Processing of SQL completed" と表示されます。実際に ZAIKO 表に行データが挿入されたかどうかを確認するために、次の SQL 文を実行します。

```
SELECT * FROM ZAIKO;
```

SQL 文を実行すると、次の検索結果が表示されます。

```
SHOHIN_CD SHOHIN_NAME COLOR TANKA          ZAIKORYO
-----
777Z      boots      black      10000          3
KFPX27010-I          1 rows selected
```

6.3.3 演習 3 (ほかの所有者の表を検索する)

MANAGER が定義した CUSTOM 表を検索します。
CUSTOM_NAME の値に shop を含む行を検索します。

【SQL文】

```
SELECT * FROM MANAGER.CUSTOM WHERE CUSTOM_NAME LIKE'%shop%';
```

【実行結果】

CUSTOM_CD	CUSTOM_NAME	TELNO	ZIPCD	ADDRESS
TK037	AK House shop	092-921-9463	099	Sapporo
TK048	AV shopping store	052-476-5624	066	Fujiyama
TK054	BB shop	015-713-6833	084	Mito
TK059	BH shop	011-123-4586	089	Koyasu
TK085	CG food shop	037-357-5372	042	Okayama
TK086	CH nattou shop	098-159-4532	073	Mito
TK091	CM food shop	067-741-7878	043	Roppongi
TK096	CR shop	013-536-2221	045	Heiwajima
TK026	Z shop	056-578-2387	092	Kawasaki

KFPX27010-I 9 rows selected

7

アプリケーションを作成してみましよう

この章では、HiRDB のユーザアプリケーションプログラム (UAP) を作成する方法について説明します。

-
- 7.1 この章を読むに当たって
 - 7.2 作成するアプリケーションの仕様
 - 7.3 C 言語でアプリケーションを作成する
 - 7.4 COBOL 言語でアプリケーションを作成する
 - 7.5 Java 言語 (SQLJ) でアプリケーションを作成する
-

7.1 この章を読むに当たって

一般の業務でデータベースを利用するとき、通常、アプリケーションプログラムを使用します。この章では、簡単なユーザアプリケーションプログラムを作成して、HiRDBに接続する手順について説明します。以降は、ユーザアプリケーションプログラムのことを、UAP と表記します。

システム管理者やオペレータの方など、アプリケーションを開発する必要がない方は、この章を読み飛ばしてかまいません。

(1) この章で説明している項目

この章では、HiRDB/Developer's Kit を使用して、UAP を作成・実行する手順について説明しています。

HiRDB の UAP 開発では、ソースプログラム中に SQL を直接記述できるという特長があります。これを、埋込み型 UAP といいます。ここでは、HiRDB の埋込み型 UAP をコンパイルして実行するまでの、一とおりの手順を理解していただくことを目的としています。なお、Java 言語での埋込み型 UAP には、静的埋込み型 UAP である SQLJ を使用します。

埋込み型 UAP のほかに、HiRDB が提供する ODBC ドライバや JDBC ドライバを使用する方法があります。ODBC ドライバや JDBC ドライバを使用した UAP 開発については、マニュアル「HiRDB Version 7 UAP 開発ガイド (UNIX(R)/Windows(R) 用)」を参照してください。

! 注意事項

言語仕様や開発環境など、一般的なプログラミングの知識については、ここでは説明していません。また、HiRDB のプログラミングインターフェースについても説明していません。HiRDB のプログラミングインターフェースについては、マニュアル「HiRDB Version 7 UAP 開発ガイド (UNIX(R)/Windows(R) 用)」を参照してください。

(2) 必要な知識について

HiRDB の UAP を作成するには、C 言語、COBOL 言語、または Java 言語を使用します。どれかの言語仕様を知っていると、よりスムーズにこの章を理解できます。

この章では手順を中心に説明していますので、言語仕様を知らない方でも、一とおりの作業ができるようになっていきます。ただし、アプリケーション開発環境のインストールやコンパイルの方法に関しては、該当製品のマニュアルを参照していただくことになります。

(3) この章の読み方について

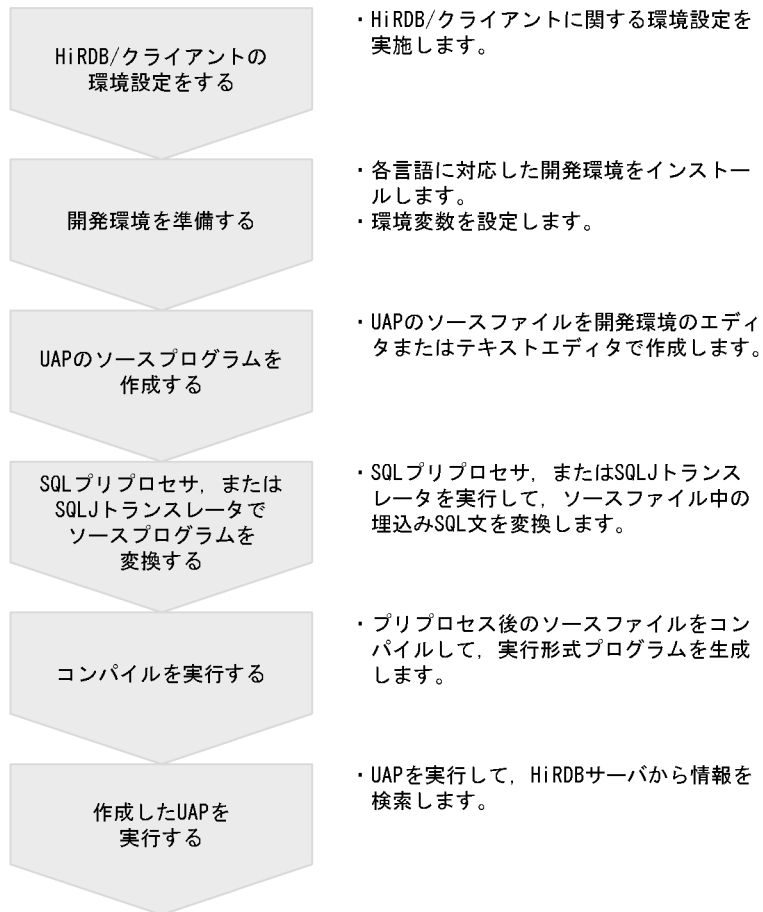
ここでは、HiRDB/クライアントで UAP を開発します。「6. クライアントを接続してみ

ましよう」でクライアントの環境設定が終了していることが前提になります。

以降の節では、作成する UAP の仕様（7.2 節）と、言語ごとの UAP 作成方法（7.3 節，7.4 節，7.5 節）を説明しています。UAP の作成方法に関しては、C 言語を使用する方は「7.3 C 言語でアプリケーションを作成する」を、COBOL 言語を使用する方は「7.4 COBOL 言語でアプリケーションを作成する」を、Java 言語を使用する方は「7.5 Java 言語（SQLJ）でアプリケーションを作成する」をお読みください。

HiRDB の UAP を作成する手順を、図 7-1 に示します。どのプログラム言語を使用した場合も、同様の手順でコンパイル・実行できます。

図 7-1 UAP 作成の流れ



（4）留意事項

UAP をコンパイルするには、コンパイラ（開発環境）が必要です。UAP 開発には、表 7-1 の開発環境を使用できます。

7. アプリケーションを作成してみましょう

表 7-1 使用できる言語と開発環境

OS	言語	開発環境
HP-UX	C 言語 / C++ 言語	ANSI 規格に準拠した C/C++ コンパイラ
	COBOL 言語	HP-UX11.0 COBOL85 Version 3.0 以降 HP-UX11i COBOL85 Version 3.0 03-03-/A 以降
	OOCOBOL 言語	HP-UX11.0 OOCOBOL Version 3.0 03-01 以降 HP-UX11i OOCOBOL Version 3.0 03-03 以降
	Java 言語	JDK 1.2.2-04 以降
HI-UX/WE2	C 言語 / C++ 言語	ANSI 規格に準拠した C/C++ コンパイラ
	COBOL 言語	COBOL85
	OOCOBOL 言語	OOCOBOL
Solaris	C 言語 / C++ 言語	ANSI 規格に準拠した C/C++ コンパイラ
	COBOL 言語	他社 COBOL Solaris 2.6 COBOL85 Version 3.0 03-00 以降 Solaris 7 COBOL85 Version 3.0 03-01 以降 Solaris 8 COBOL85 Version 3.0 03-02 以降 Solaris 9 COBOL85 Version 3.0 03-05-/A 以降
	Java 言語	JDK 1.3.0-01 以降
AIX 5L	C 言語 / C++ 言語	ANSI 規格に準拠した C/C++ コンパイラ
	COBOL 言語	AIX 5L V5.1 COBOL85 Version 3.0 03-03 以降 AIX 5L V5.2 COBOL85 Version 3.0 03-05-/D 以降
	OOCOBOL 言語	OOCOBOL Version 3.0 03-03 以降
	Java 言語	JDK 1.3 以降
Linux	C 言語 / C++ 言語	ANSI 規格に準拠した C/C++ コンパイラ, gcc, g++
	COBOL 言語	COBOL85 Version 3.0 03-04 以降
	Java 言語	JDK 1.3.0-01 以降

使用する言語に応じて開発環境を選択し、インストールしてください。インストールの方法に関しては、該当製品のマニュアルを参考にしてください。なお、このマニュアルでは次の開発環境を使用します。

- 最適化 C (C 言語の場合)

- COBOL85 (COBOL 言語の場合)
- JDK 1.4.0 (Java 言語の場合)

このマニュアルでは、CUI (C_haracter U_ser I_nterface) ベースのアプリケーションをコマンドでコンパイルする方法を紹介します。このため、開発環境のバージョンが異なる場合でも、同様の手順でコンパイルおよび実行できます。

7.2 作成するアプリケーションの仕様

作成するサンプル UAP について説明します。

7.2.1 サンプル UAP の機能

HiRDB サーバに接続して、データベースの情報を検索する UAP を作成してみましょう。具体的には、次の機能を持たせます。

- 「2. HiRDB システムを構築してみましょう」で構築した HiRDB システムに接続します。ユーザ名とパスワードは、標準入力から受け取ります。
- UAP から SQL の SELECT 文を発行して、CUSTOM 表の内容を検索します。このとき、コマンドライン引数で CUSTOM_CD を二つ指定して、指定した CUSTOM_CD 間の CUSTOM を取り出すように検索条件を設定します。コマンドラインでの指定のしかたは次のようになります。

```
$ program aaa bbb
```

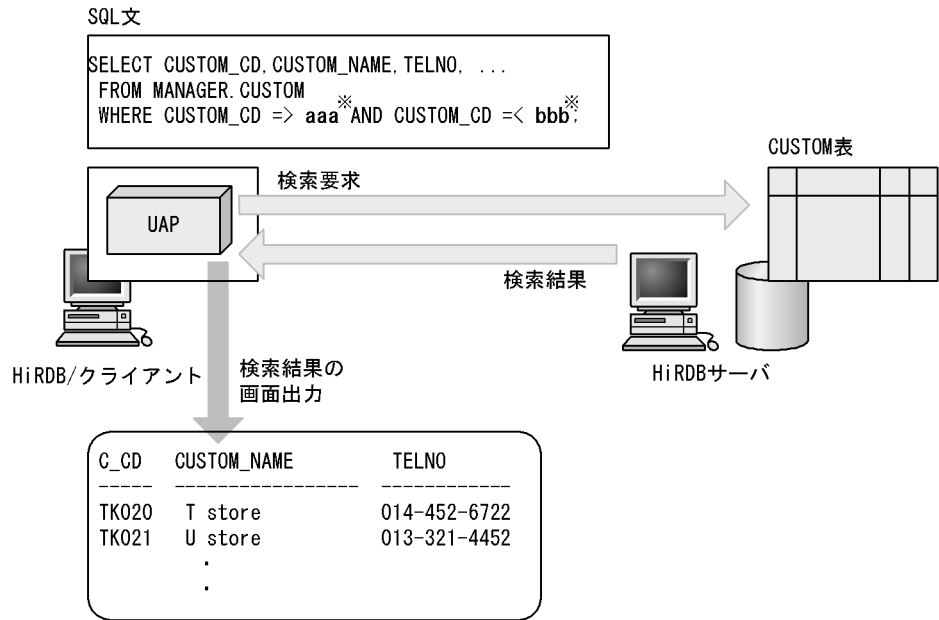
上記のように指定すると、CUSTOM_CD が aaa ~ bbb の CUSTOM の情報が取り出されます。

- HiRDB サーバから検索結果を受け取り、書式を整えて標準出力に表示します。

上記の機能を実現する CUI ベースの UAP を作成します。UAP 実行者からの入力、コマンドライン引数または標準入力から受け取ります。また、検索結果は、標準出力に出力することにします。

UAP の機能の概要を、図 7-2 に示します。

図 7-2 サンプル UAP の機能の概要



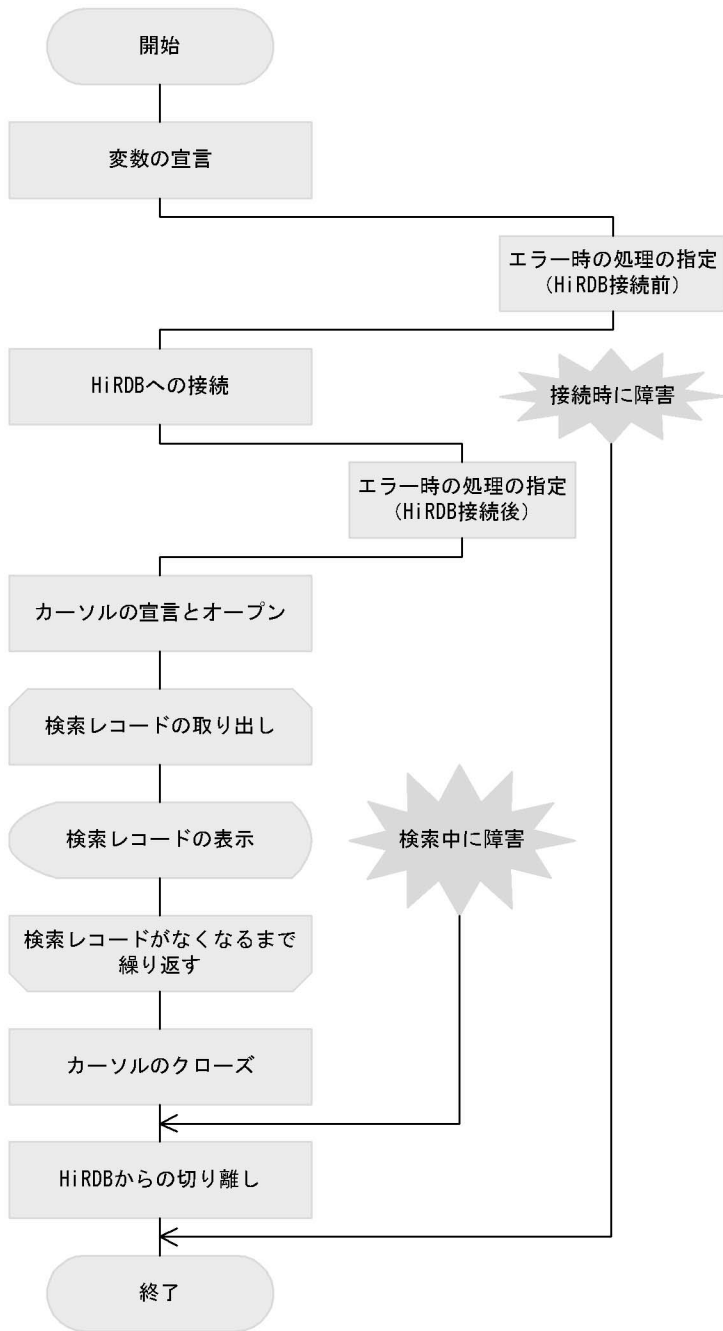
注※ aaa, bbbは, UAPの実行者がコマンドライン引数で指定した値です。

7.2.2 サンプル UAP の処理の流れ

サンプル UAP の処理の流れを図 7-3 に示します。コーディングの内容を知りたい方は、図 7-3 とコーディング中のコメントを参照してください。

7. アプリケーションを作成してみましょう

図 7-3 サンプル UAP の流れ



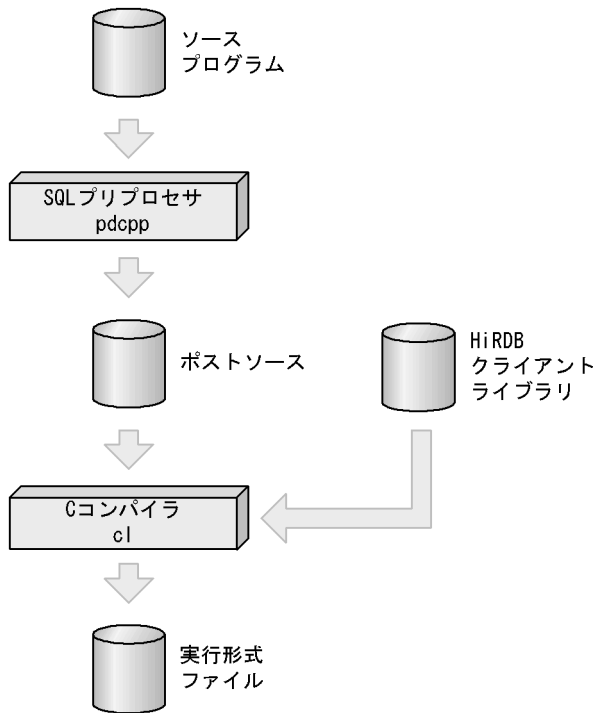
7.3 C 言語でアプリケーションを作成する

C 言語を使用して UAP を作成する方法を説明します。

7.3.1 基礎知識

C 言語の埋込み型 UAP から実行形式ファイルを生成するためには、埋込み型 UAP をいったんポストソースに変換する必要があります。UAP 生成の流れを図 7-4 に示します。

図 7-4 UAP 生成の流れ (C 言語の場合)



また、この演習で作成するファイル名を次に示します。

ソースプログラム

```
$HOME/csample/csample.ec
```

ポストソース

```
$HOME/csample/csample.c
```

実行形式ファイル

```
$HOME/csample/csample
```

7. アプリケーションを作成してみましょう

7.3.2 HiRDB/クライアントの環境設定をする

HiRDBのUAPを開発するには、HiRDB/Developer's Kitが必要です。

HiRDB/クライアントの環境設定については、「6.2 クライアントの環境設定をする」で実施済みです。環境設定を行っていない場合は、「6.2 クライアントの環境設定をする」の手順に従って設定を行ってください。

! 注意事項

HiRDB/Run Time ではUAPを開発できません。HiRDB/Developer's Kitをインストールしてください。

7.3.3 開発環境を準備する

この演習を行うには、次の開発環境が必要です。

- 最適化 C

(1) 環境変数を設定する

\$HOME/.profile に次の記述を追加して、「. \$HOME/.profile」を実行してください。

```
PDDIR=/opt/HiRDB
PATH=/opt/HiRDB/client/utl:$PATH
LANG=ja_JP.SJIS
export PDDIR PATH LANG
```

【説明】

- PDDIR
クライアントマシンのHiRDB/Developer's Kitのインストールディレクトリの絶対パスです。
- PATH
プリプロセサファイルへのパスです。
- LANG
サーバの文字コードの種別を設定します。

7.3.4 UAPのソースプログラムを作成する

UAPのソースプログラムを作成します。

(1) 「csample」ディレクトリの作成

サンプルUAPを格納するディレクトリを作成します。ログインユーザのディレクトリ

に、次のディレクトリを作成してください。

```
$HOME/csample
```

(2) 「csample.ec」の作成

開発環境のエディタまたはテキストエディタを使って、埋込み型 UAP のソースプログラムを作成します。次のテキストファイルを作成してください。

```
$HOME/csample/csample.ec
```

ソースプログラム (csample.ec) の内容を次に示します。

7. アプリケーションを作成してみましょう

```
/* **** */
/* [Abstract] */
/* This is the sample program of the HiRDB First Step Guide. */
/* This program searches the MANGER.CUSTOM table and displays the range of */
/* CUSTOM_NAME items between two entered C_CD code numbers. */
/* **** */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void connectError();
void abnormalEnd();

int main( int argc, char **argv )
{
/* Declare variables **** */
EXEC SQL BEGIN DECLARE SECTION;
char xUserId[9]; /* User ID */
char xPswd[9]; /* Password */
char gCCode[6]; /* CUSTOM_CD */
char gCName[31]; /* CUSTOM_NAME */
char gTelno[13]; /* TELNO */
char gZipcd[4]; /* ZIPCD */
char gAddress[10]; /* ADDRESS */
char aSmall[6]; /* Argument1 */
char aLarge[6]; /* Argument2 */
EXEC SQL END DECLARE SECTION;

/* Display start message and read input **** */
if( argc == 3 ){
strcpy(aSmall, argv[1]);
strcpy(aLarge, argv[2]);
}

printf("Processing has started.\n");

/* Error process before connecting to HiRDB **** */
EXEC SQL WHENEVER SQLERROR PERFORM connectError;

/* Connect to HiRDB **** */
printf("USERID:");
scanf("%s", &xUserId);
printf("PASSWORD:");
scanf("%s", &xPswd);
EXEC SQL CONNECT USER :xUserId USING :xPswd;
printf("HiRDB was successfully connected.\n");

/* Error process after connecting to HiRDB **** */
EXEC SQL WHENEVER SQLERROR PERFORM abnormalEnd;
}
```

```

/* Declare and open CURSOR *****/
if( argc != 3 ){
    EXEC SQL DECLARE CR1 CURSOR FOR
        SELECT CUSTOM_CD, CUSTOM_NAME, TELNO, ZIPCD, ADDRESS FROM MANAGER.CUSTOM;
    EXEC SQL OPEN CR1;
}else{
    EXEC SQL DECLARE CR2 CURSOR FOR
        SELECT CUSTOM_CD, CUSTOM_NAME, TELNO, ZIPCD, ADDRESS
        FROM MANAGER.CUSTOM WHERE CUSTOM_CD >= :aSmall AND CUSTOM_CD <= :aLarge;
    EXEC SQL OPEN CR2;
}
SQLCODE=0;

/* Display search result *****/
printf(" C_NO    CUSTOM_NAME          TELNO    ZIP    ADDRESS\n");
printf(" -----\n");
while(SQLCODE>=0){
    EXEC SQL WHENEVER NOT FOUND GO TO OWARI;
    if( argc != 3 )
        EXEC SQL FETCH CR1 INTO :gCCode, :gCName, :gTelno, :gZipcd, : gAddress;
    else
        EXEC SQL FETCH CR2 INTO :gCCode, :gCName, :gTelno, :gZipcd, : gAddress;
    EXEC SQL WHENEVER NOT FOUND CONTINUE;

    printf(" %5s %30s %12s %3s %10s\n", gCCode, gCName, gTelno, gZipcd, gAddress);
}

OWARI:
/* Close CURSOR *****/
if( argc != 3 ){
    EXEC SQL CLOSE CR1;
}else{
    EXEC SQL CLOSE CR2;
}
EXEC SQL COMMIT;

/* Disconnect from HiRDB *****/
EXEC SQL WHENEVER SQLERROR CONTINUE;
EXEC SQL WHENEVER NOT FOUND CONTINUE;
EXEC SQL DISCONNECT;
printf("HiRDB was successfully disconnected.\n");

/* End processing *****/
printf("Processing has ended.\n");
return(0);
}

/* Error process before connecting to HiRDB *****/
void connectError(void)
{
    printf("*** Connection to HiRDB failed.\n");
    EXEC SQL WHENEVER SQLERROR CONTINUE;
    EXEC SQL DISCONNECT;
    exit(2);
}

```

7. アプリケーションを作成してみましょう

```
/* Error process after connecting to HiRDB *****/
void abnormalEnd()
{
    int wsqlcode;

    wsqlcode = (-1)*SQLCODE;
    printf("¥n*** SQL error occurred. ¥n");
    printf(" MESSAGE-ID = KFP11%3d-E¥n", wsqlcode);
    printf(" SQLERRMC = %s¥n", SQLERRMC);
    EXEC SQL WHENEVER SQLERROR CONTINUE;
    EXEC SQL ROLLBACK;
    EXEC SQL DISCONNECT;
    exit(2);
}
```

7.3.5 SQL プリプロセサでソースプログラムを変換する

SQL プリプロセサを使用して、UAP のソースプログラムをポストソースに変換します。

【コマンド形式】

```
$ pdcpp $HOME/csample/csample.ec -o $HOME/csample/csample.c
```

【説明】

`$HOME/csample/csample.ec`

UAP のソースプログラム (入力ファイル) を指定します。

`-o $HOME/csample/csample.c`

ポストソース (出力ファイル) を指定します。

コマンドが正常に終了すると、`$HOME/csample` ディレクトリ下に `csample.c` が生成されます。このファイルが生成されているかどうかを確認してください。

7.3.6 コンパイルを実行する

C コンパイラを使用して、先ほど SQL プリプロセサで変換したポストソースを、実行形式プログラムに変換します。このコマンドを実行する前に、環境変数 `PATH` に `cc` コマンドの実行ファイルを保存しているディレクトリが設定されていることを確認してください。

【コマンド形式】

```
$ cc -o $HOME/csample/csample -I $PDDIR/include $HOME/csample/csample.c
-L$PDDIR/client/lib -lzcltk
```

【説明】

```

$HOME/csample/csample.c
    C のソースプログラム（入力ファイル）を指定します。

-o $HOME/csample/csample
    実行ファイル（出力ファイル）を指定します。

-l $PDDIR/include
    HiRDB のインクルードディレクトリを指定します。

-L $PDDIR/client/lib
    HiRDB が提供する共用ライブラリのディレクトリを指定します。

-lzcltk
    HiRDB が提供する共用ライブラリ（libzcltk.sl）を指定します。

```

コマンドが正常に終了すると、\$HOME/csample ディレクトリ下に `csample` が生成されます。このファイルが生成されているかどうかを確認してください。

ここで、環境変数などが正しく設定されていないと、コンパイルエラーが発生します。エラーが発生した場合、設定内容を見直して再度コンパイルしてください。

7.3.7 作成した UAP を実行する

作成したサンプル UAP を実行してみましょう。サンプル UAP を実行する前に、HiRDB サーバが開始されていることを確認してください。

サンプル UAP には二つのコマンド引数を与えることで、CUSTOM_CD の範囲を指定します。ここでは、CUSTOM_CD が TK020 ~ TK030 の情報を CUSTOM 表から検索します。次のコマンドを実行してください。

【コマンド形式】

```
$ $HOME/csample/csample TK020 TK030
```

サンプル UAP が開始されると、次のメッセージが返されます。

【コマンド形式の実行結果】

```
Processing has started.
```

7. アプリケーションを作成してみましょう

それに引き続いて、ユーザ名とパスワードの入力を求めるメッセージが出力されます。
ここでは、ユーザ名「USER01」、パスワード「PASS01」を入力してください。

HiRDB サーバに接続すると、次の結果が表示されます。

【サンプル UAP の実行結果】

```
HiRDB was successfully connected.
C_NO    CUSTOM_NAME          TELNO          ZIP    ADDRESS
-----
TK020   T store              014-452-6722  047    Honmoku
TK021   U store              013-321-4452  077    Isezaki
TK022   V store              012-967-7342  042    Harajuku
TK023   W industry           066-980-4482  034    Yoshida
TK024   X House factory      065-976-2245  064    Hirato
TK025   Y electric appliance store 345-129-8772  088    Motomiya
TK026   Z shop               056-578-2387  092    Kawasaki
TK027   AA store             078-546-3345  037    Tsurumi
TK028   AB department store  089-783-3657  094    Obihiro
TK029   AC &Co.              012-198-4941  022    Akebono
TK030   AD store             015-234-5673  088    Hongou
HiRDB was successfully disconnected.
Processing has ended.
```

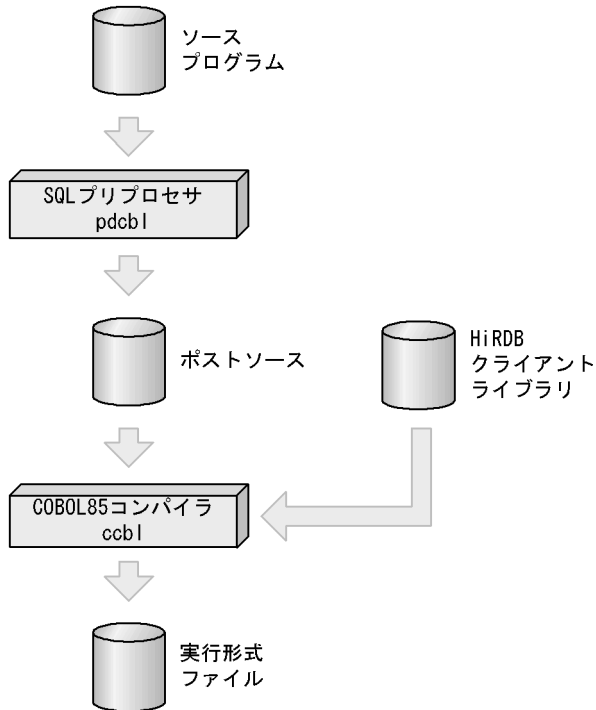

7.4 COBOL 言語でアプリケーションを作成する

COBOL 言語を使用して UAP を作成する方法を説明します。

7.4.1 基礎知識

COBOL 言語の埋込み型 UAP から実行形式ファイルを生成するためには、埋込み型 UAP をいったんポストソースに変換する必要があります。UAP 生成の流れを図 7-5 に示します。

図 7-5 UAP 生成の流れ (COBOL 言語の場合)



また、この演習で作成するファイル名を次に示します。

ソースプログラム

```
$HOME/cblsample/cblsample.ecb
```

ポストソース

```
$HOME/cblsample/cblsample.cbl
```

実行形式ファイル

7. アプリケーションを作成してみましょう

```
$HOME/cblsample/cblsample
```

7.4.2 HiRDB/クライアントの環境設定をする

HiRDBのUAPを開発するには、HiRDB/Developer's Kitが必要です。

HiRDB/クライアントの環境設定については、「6.2 クライアントの環境設定をする」で実施済みです。環境設定を行っていない場合は、「6.2 クライアントの環境設定をする」の手順に従って設定を行ってください。

! 注意事項

HiRDB/Run Time では UAP を開発できません。HiRDB/Developer's Kit をインストールしてください。

7.4.3 開発環境を準備する

この演習を行うには、次の開発環境が必要です。

- COBOL85

(1) COBOL85 をインストール

COBOL85 をインストールしてください。インストール方法については、該当製品のマニュアルを参考にしてください。

(2) 環境変数を設定する

`$HOME/.profile` に次の記述を追加して、「`.$HOME/.profile`」を実行してください。

```
LANG=ja_JP.SJIS
PATH=/opt/HILNGcbl/bin:/opt/HiRDB/client/utl:$PATH
CBLLIB=/opt/HiRDB/include
SHLIB_PATH=/opt/HiRDB/client/lib:/opt/HILNGcbl/lib
export LANG PATH CBLLIB SHLIB_PATH
```

【説明】

- LANG
サーバの文字コードの種別を設定します。
- PATH
COBOL85 の実行ファイルおよびプリプロセサファイルへのパスです。
- CBLLIB
COBOL85 のインクルードディレクトリを指定します。
- SHLIB_PATH

HiRDB で提供している共用ライブラリおよび COBOL85 のライブラリへのパスです。

7.4.4 UAP のソースプログラムを作成する

UAP のソースプログラムを作成します。

(1) 「cblsample」ディレクトリの作成

サンプル UAP を格納するディレクトリを作成します。ログインユーザのディレクトリに、次のディレクトリを作成してください。

```
$HOME/cblsample
```

(2) 「cblsample.ecb」の作成

開発環境のエディタまたはテキストエディタを使って、埋込み型 UAP のソースプログラムを作成します。次のテキストファイルを作成してください。

```
$HOME/cblsample/cblsample.ecb
```

ソースプログラム (cblsample.ecb) の内容を次に示します。

7. アプリケーションを作成してみましょう

```
000100 IDENTIFICATION DIVISION.
000200*****
000300* [Abstract]
000400* This is the sample program of the HiRDB First Step Guide.
000500* This program searches the MANGER.CUSTOM table and
000600* displays the range of CUSTOM_NAME items
000700* between two entered CUSTOM_CD code numbers.
000800*****
000900*
001000 PROGRAM-ID. CBLSAMPLE.
001100*
001200 ENVIRONMENT DIVISION.
001300*
001400 DATA DIVISION.
001500 WORKING-STORAGE SECTION.
001600 01 CMDLINE PIC X(18).
001700 01 ARGNUM PIC 9(1).
001800* Declare variables *****
001900 EXEC SQL
002000 BEGIN DECLARE SECTION
002100 END-EXEC.
002200 77 XUSERID PIC X(8).
002300 77 XPSWD PIC X(8).
002400 77 GCCODE PIC X(5).
002500 77 GCNAME PIC X(30).
002600 77 GTELNO PIC X(12).
002700 77 GZPCD PIC X(5).
002800 77 GADDRESS PIC X(10).
002900 77 ASMAIL PIC X(5).
003000 77 ALARGE PIC X(5).
003100 EXEC SQL
003200 END DECLARE SECTION
003300 END-EXEC.
003400
003500*
003600 PROCEDURE DIVISION.
003700* Display start message and read input *****
003800 DISPLAY 'Processing has started.'
003900 ACCEPT CMDLINE FROM COMMAND-LINE
004000 UNSTRING CMDLINE DELIMITED BY SPACE
004100 INTO ASMAIL
004200 ALARGE
004300 TALLYING IN ARGNUM
004400 END-UNSTRING.
004500
004500* Error process before connecting to HiRDB *****
004700 EXEC SQL
004800 WHENEVER SQLERROR GO TO ERR2
004900 END-EXEC.
005000
```

```

005100* Connect to HirDB *****
005200    DISPLAY 'USERID:' WITH NO ADVANCING
005300    ACCEPT XUSERID
005400    DISPLAY 'PASSWORD:' WITH NO ADVANCING
005500    ACCEPT XPSWD
005600    EXEC SQL
005700        CONNECT :XUSERID IDENTIFIED BY :XPSWD
005800    END-EXEC
005900    DISPLAY 'HirDB was successfully connected.'.
006000
006100* Error process after connecting to HirDB *****
006200    EXEC SQL
006300        WHENEVER SQLERROR GO TO ERR1
006400    END-EXEC.
006500
006600* Declare and open CURSOR *****
006700    IF ASmall IS = "      " THEN
006800        EXEC SQL
006900            DECLARE CR1 CURSOR FOR
007000                SELECT CUSTOM_CD, CUSTOM_NAME, TELNO, ZIPCD, ADDRESS
007100                    FROM MANAGER.CUSTOM
007200            END-EXEC
007300        EXEC SQL
007400            OPEN CR1
007500        END-EXEC
007600    ELSE
007700        EXEC SQL
007800            DECLARE CR2 CURSOR FOR
007900                SELECT CUSTOM_CD, CUSTOM_NAME, TELNO, ZIPCD, ADDRESS
008000                    FROM MANAGER.CUSTOM
008100                    WHERE CUSTOM_CD >= :ASmall AND CUSTOM_CD <= :ALARGE
008200            END-EXEC
008300        EXEC SQL
008400            OPEN CR2
008500        END-EXEC
008600    END-IF.
008700
008800* Display search result *****
008900    DISPLAY 'C_CD    CUSTOM_NAME          ' WITH NO ADVANCING
009000    DISPLAY '          TELNO          ZIP  ADDRESS'
009100    DISPLAY '-----' WITH NO ADVANCING
009200    DISPLAY '-----'
009300    EXEC SQL
009400        WHENEVER NOT FOUND GO TO CONT
009500    END-EXEC.
009600 LOOP.
009700    IF ASmall IS = "      " THEN
009800        EXEC SQL
009900            FETCH CR1 INTO :GCCODE, :GCNAME, :GTELNO, :GZIPCD, :GADDRESS
010000        END-EXEC
010100    ELSE
010200        EXEC SQL
010300            FETCH CR2 INTO :GCCODE, :GCNAME, :GTELNO, :GZIPCD, :GADDRESS
010400        END-EXEC
010500    END-IF

```

7. アプリケーションを作成してみましょう

```
010600    DISPLAY GCCODE WITH NO ADVANCING
010700    DISPLAY ' ' WITH NO ADVANCING
010800    DISPLAY GCNAME WITH NO ADVANCING
010900    DISPLAY ' ' WITH NO ADVANCING
011000    DISPLAY GTELNO WITH NO ADVANCING
011100    DISPLAY ' ' WITH NO ADVANCING
011200    DISPLAY GZIPCD WITH NO ADVANCING
011300    DISPLAY GADDRESS
011400    GO TO LOOP.
011500
011600* Close CURSOR *****
011700 CONT.
011800    EXEC SQL
011900        WHENEVER NOT FOUND CONTINUE
012000    END-EXEC.
012100    IF ASMAIL IS = " " THEN
012200        EXEC SQL
012300            CLOSE CR1
012400        END-EXEC
012500    ELSE
012600        EXEC SQL
012700            CLOSE CR2
012800        END-EXEC
012900    END-IF.
013000    GO TO DISC.
013100
013200* Error process after connecting to HiRDB *****
013300 ERR1.
013400    DISPLAY '*** SQL error occurred.'.
013500
013600* Disconnect from HiRDB *****
013700 DISC.
013800    EXEC SQL
013900        WHENEVER SQLERROR CONTINUE
014000    END-EXEC.
014100    EXEC SQL
014200        DISCONNECT
014300    END-EXEC.
014400    DISPLAY 'HiRDB was successfully disconnected'.
014500    GO TO FIN.
014600
014700* Error process before connecting to HiRDB *****
014800 ERR2.
014900    DISPLAY '*** Connection to HiRDB failed.'.
015000
015100* End processing *****
015200 FIN.
015300    DISPLAY 'Processing has ended.'.
015400    STOP RUN.
015500 END PROGRAM CBLSAMPLE.
```

7.4.5 SQL プリプロセサでソースプログラムを変換する

SQL プリプロセサを使用して、UAP のソースプログラムをポストソースに変換します。

【コマンド形式】

```
$ pdcbl $HOME/cblsample/cblsample.ecb -o $HOME/cblsample/cblsample.cbl
```

【説明】

\$HOME/cblsample/cblsample.ecb

UAP のソースプログラム（入力ファイル）を指定します。

-o \$HOME/cblsample/cblsample.cbl

ポストソース（出力ファイル）を指定します。

コマンドが正常に終了すると、\$HOME/cblsample ディレクトリ下に cblsample.cbl が生成されます。このファイルが生成されているかどうかを確認してください。

7.4.6 コンパイルを実行する

COBOL コンパイラを使用して、先ほど SQL プリプロセサで変換したポストソースを、実行形式プログラムに変換します。

【コマンド形式】

```
$ ccb1 -o $HOME/cblsample/cblsample $HOME/cblsample/cblsample.cbl -L /opt/HiRDB/client/lib -lzclt
```

【説明】

-o \$HOME/cblsample/cblsample

実行ファイル（出力ファイル）を指定します。

\$HOME/cblsample/cblsample.cbl

COBOL のソースプログラム（入力ファイル）を指定します。

-L /opt/HiRDB/client/lib

HiRDB が提供する共用ライブラリのディレクトリを指定します。

-lzclt

HiRDB が提供する共用ライブラリ（libzclt.sl）を指定します。

コマンドが正常に終了すると、\$HOME/cblsampleC:\¥CBLSAMPLE ディレクトリ下に cblsample が生成されます。このファイルが生成されているかどうかを確認してください。

7.4.7 作成した UAP を実行する

作成したサンプル UAP を実行してみましょう。サンプル UAP を実行する前に、HiRDB サーバが開始されていることを確認してください。

7. アプリケーションを作成してみましょう

サンプル UAP には二つのコマンド引数を与えることで、CUSTOM_CD の範囲を指定します。ここでは、CUSTOM_CD が TK020 ~ TK030 の情報を CUSTOM 表から検索します。次のコマンドを実行してください。

【コマンド形式】

```
$ $HOME/cblsample/cblsample TK020 TK030
```

サンプル UAP が開始されると、次のメッセージが返されます。

【コマンド形式の実行結果】

```
Processing has started.
```

それに引き続いて、ユーザ名とパスワードの入力を求めるメッセージが出力されます。ここでは、ユーザ名「USER01」、パスワード「PASS01」を入力してください。クライアント環境定義 PDUSER が設定されていますので、ユーザ名、パスワードを入力しないで < Enter > キーを押しても、HiRDB サーバに接続できます。

HiRDB サーバに接続すると、次の結果が表示されます。

【サンプル UAP の実行結果】

```
HiRDB was successfully connected.
C_NO    CUSTOM_NAME          TELNO          ZIP  ADDRESS
-----
TK020   T store              014-452-6722  047  Honmoku
TK021   U store              013-321-4452  077  Isezaki
TK022   V store              012-967-7342  042  Harajuku
TK023   W industry           066-980-4482  034  Yoshida
TK024   X House factory      065-976-2245  064  Hirato
TK025   Y electric appliance store 345-129-8772  088  Motomiya
TK026   Z shop               056-578-2387  092  Kawasaki
TK027   AA store             078-546-3345  037  Tsurumi
TK028   AB department store  089-783-3657  094  Obihiro
TK029   AC &Co.              012-198-4941  022  Akebono
TK030   AD store             015-234-5673  088  Hongou
HiRDB was successfully disconnected.
Processing has ended.
```

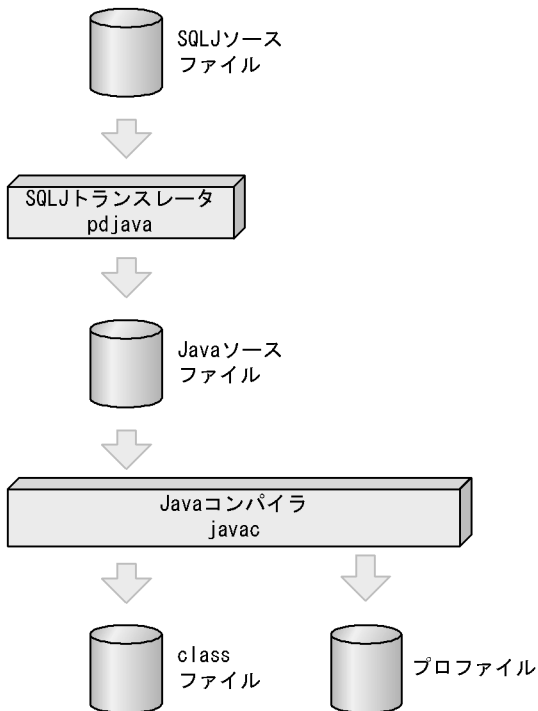

7.5 Java 言語 (SQLJ) でアプリケーションを作成する

Java 言語 (SQLJ) を使用して UAP を作成する方法を説明します。

7.5.1 基礎知識

Java 言語の埋込み型 UAP から class ファイルを生成するためには、埋込み型 UAP をいったん Java ソースに変換する必要があります。UAP 生成の流れを図 7-6 に示します。

図 7-6 UAP 生成の流れ (Java 言語の場合)



また、この演習で作成するファイル名を次に示します。

ソースプログラム

```
$HOME/javasample/javasample.sqlj
```

ポストソース

```
$HOME/javasample/javasample.java
```

class ファイル

```
$HOME/javasample/javasample.class
```

7.5.2 HiRDB/ クライアントの環境設定をする

HiRDB の UAP を開発するには、HiRDB/Developer's Kit が必要です。

HiRDB/ クライアントの環境設定については、「6.2 クライアントの環境設定をする」で実施済みです。環境設定を行っていない場合は、「6.2 クライアントの環境設定をする」の手順に従って設定を行ってください。

! 注意事項

HiRDB/Run Time では UAP を開発できません。HiRDB/Developer's Kit をインストールしてください。

7.5.3 開発環境を準備する

この演習を行うには、次の開発環境が必要です。

- JDK 1.2.2-04 以降

(1) JDK をインストールする

JDK をインストールしてください。インストール方法については、該当製品のマニュアルを参考にしてください。

(2) 環境変数を設定する

\$HOME/.profile に次の記述を追加して、「. \$HOME/.profile」を実行してください。

```
PATH=/HiRDB/utl:$PATH
CLASSPATH=/HiRDB/utl/pdsq1j.jar:/HiRDB/client/utl/pdjdbc.jar:$CLASSPATH
export PATH CLASSPATH
```

【説明】

- PATH
Java トランスレータファイルへのパスです。
- CLASSPATH
HiRDB が提供する jar ファイルのパスです。

7.5.4 UAP のソースプログラムを作成する

UAP のソースプログラムを作成します。

(1) 「javasample」ディレクトリの作成

サンプル UAP を格納するディレクトリを作成します。ログインユーザのディレクトリに、次のディレクトリを作成してください。

```
$HOME/javasample
```

(2) 「javasample.sqlj」の作成

開発環境のエディタまたはテキストエディタを使って、埋込み型 UAP のソースプログラムを作成します。次のテキストファイルを作成してください。

```
$HOME/javasample/javasample.sqlj
```

ソースプログラム (javasample.sqlj) の内容を次に示します。

7. アプリケーションを作成してみましょう

```
/* **** */
/* [Abstract] */
/* This is the sample program of the HiRDB First Step Guide. */
/* This program searches the MANGER.CUSTOM table and displays the range of */
/* CUSTOM_NAME items between two entered C_CD code numbers. */
/* **** */
import java.awt.*;
import java.lang.*;
import java.io.*;
import java.sql.*;

#sql context ExecutionContext;
#sql iterator ReadIterator(String, String, String, String, String);

public class javasample
{
public static void main (String args[])
{
/* Declare variables **** */
String xUserId = null;      /* User ID      */
String xPswd = null;       /* Password    */
String gCCode = null;      /* CUSTOM_CD   */
String gCName = null;      /* CUSTOM_NAME */
String gTelno = null;      /* TELNO      */
String gZipcd = null;      /* ZIPCD      */
String gAddress = null;    /* CUSTOM_NAME */
String aSmall = null;      /* argument   */
String aLarge = null;     /* argument2  */
BufferedReader inp = null; /* buffer for input */
ExecutionContext execCtx = null; /* connection */
ReadIterator readItr = null; /* cursor     */

/* Display start message and read input **** */
if( args.length == 2 ){
    aSmall = new String(args[0]);
    aLarge = new String(args[1]);
}
System.out.println("Processing has started.");
try {

/* Connect to HiRDB **** */
inp = new BufferedReader(new InputStreamReader(System.in));
System.out.print("USERID:");
xUserId = inp.readLine();
System.out.print("PASSWORD:");
xPswd = inp.readLine();

Class.forName("JP.co.Hitachi.soft.HiRDB.JDBC.PrdbDriver");
execCtx = new ExecutionContext("jdbc:hitachi:PrdbDrive", xUserId,
                               xPswd, false);
System.out.println("HiRDB was successfully connected.");
} catch (Exception e) {

/* Error process before connecting to HiRDB **** */
System.out.println("*** Connection to HiRDB failed.");
System.out.println(e.getMessage());
return;
}
}
}
```

```

try {
/* Declare and open CURSOR *****/
if ( aSmall == null ) {
    #sql [execCtx] readItr =
    { SELECT CUSTOM_CD, CUSTOM_NAME, TELNO, ZIPCD, ADDRESS
      FROM MANAGER.CUSTOM };
} else {
    #sql [execCtx] readItr =
    { SELECT CUSTOM_CD, CUSTOM_NAME, TELNO, ZIPCD, ADDRESS
      FROM MANAGER.CUSTOM
      WHERE CUSTOM_CD >= :aSmall AND CUSTOM_CD <= :aLarge };
}

/* Display search result *****/
System.out.println(" C_NO   CUSTOM_NAME           "+
                  "TELNO       ZIP   ADDRESS");
System.out.println(" -----  -----");
while(true){
    #sql { FETCH :readItr INTO :gCCode, :gCName, :gTelno,
              :gZipcd, :gAddress };
    if(readItr.endFetch()) break;
    System.out.println(" "+gCCode+" "+gCName+" "+gTelno
                      +" "+gZipcd+" "+gAddress);
}

/* Close CURSOR *****/
readItr.close();

/* Disconnect from HiRDB *****/
execCtx.close();
System.out.println("disconnect from HiRDB.");

/* End processing *****/
System.out.println("end process.");

} catch (SQLException e) {

/* Error process after connecting to HiRDB *****/
System.out.println("*** SQL error occurred.");
System.out.println("ErrorCode = "+e.getErrorCode());
System.out.println(e.getMessage());
try {
    execCtx.close();
} catch (Exception ee) {
    ;
}

} catch (Exception e) {
System.out.println("*** Some errors occurred.");
System.out.println(e.getMessage());
}
}
}

```

7. アプリケーションを作成してみましょう

7.5.5 SQL トランスレータでソースプログラムを変換する

SQL トランスレータを使用して、UAP のソースプログラムを java ソースに変換します。

【コマンド形式】

```
$ pdjava $HOME/javasample/javasample.sqlj
```

【説明】

`$HOME/javasample/javasample.sqlj`

UAP のソースプログラム（入力ファイル）を指定します。

コマンドが正常に終了すると、`$HOME/javasample` ディレクトリ下に `javasample.java` およびプロファイル（拡張子 `ser`）が生成されます。このファイルが生成されているかどうかを確認してください。

7.5.6 コンパイルを実行する

Java コンパイラを使用して、先ほど SQL プリプロセサで変換した Java ソースを、`class` ファイルに変換します。このコマンドを実行する前に、環境変数 `PATH` に `javac` コマンドの実行ファイルを保存しているディレクトリが設定されていることを確認してください。

【コマンド形式】

```
$ javac $HOME/javasample/javasample.java
```

【説明】

`$HOME/javasample/javasample.java`

Java のソースプログラム（入力ファイル）を指定します。

コマンドが正常に終了すると、`$HOME/javasample` ディレクトリ下に `javasample.class` を含む四つの `class` ファイルが生成されます。このファイルが生成されているかどうかを確認してください。

ここで、環境変数などが正しく設定されていないと、コンパイルエラーが発生します。エラーが発生した場合、設定内容を見直して再度コンパイルしてください。

7.5.7 作成した UAP を実行する

作成したサンプル UAP を実行してみましょう。サンプル UAP を実行する前に、HiRDB

サーバが開始されていることを確認してください。

サンプル UAP には二つのコマンド引数を与えることで、CUSTOM_CD の範囲を指定します。ここでは、CUSTOM_CD が TK020 ~ TK030 の情報を CUSTOM 表から検索します。次のコマンドを実行してください。

【コマンド形式】

```
$ java $HOME/javasample/javasample TK020 TK030
```

サンプル UAP が開始されると、次のメッセージが返されます。

【コマンド形式の実行結果】

```
Processing has started.
```

それに引き続いて、ユーザ名とパスワードの入力を求めるメッセージが出力されます。ここでは、ユーザ名「USER01」、パスワード「PASS01」を入力してください。

HiRDB サーバに接続すると、次の結果が表示されます。

【サンプル UAP の実行結果】

```
HiRDB was successfully connected.
  C_NO  CUSTOM_NAME          TELNO      ZIP  ADDRESS
-----
TK020  T store                014-452-6722  047  Honmoku
TK021  U store                013-321-4452  077  Isezaki
TK022  V store                012-967-7342  042  Harajuku
TK023  W industry            066-980-4482  034  Yoshida
TK024  X House factory       065-976-2245  064  Hirato
TK025  Y electric appliance  345-129-8772  088  Motomiya
TK026  Z shop                056-578-2387  092  Kawasaki
TK027  AA store              078-546-3345  037  Tsurumi
TK028  AB department store   089-783-3657  094  Obihiro
TK029  AC &Co.               012-198-4941  022  Akebono
TK030  AD store              015-234-5673  088  Hongou
HiRDB was successfully disconnected.
Processing has ended.
```


付録

付録A Q & A

付録B HiRDB のアンインストール

付録C 用語解説

付録 A Q & A

HiRDB の環境設定，運用でよくある質問をまとめています。ここでは，次の項目についての Q&A を記載しています。

1. 環境設定に関する質問
2. HiRDB のシステム構成に関する質問
3. 運用に関する質問
4. トラブルに関する質問

付録 A.1 環境設定に関する質問

(1) システム共通定義の pdstart オペランドに指定するホスト名

質問

システム共通定義の pdstart オペランドに指定するホスト名は，どこの設定に対応しているのですか？

お答えします

コマンドラインから hostname コマンドを実行してください。出力された値がホスト名です。

(2) 通常ファイルとキャラクタ型スペシャルファイルの使い分け

質問

通常ファイルとキャラクタ型スペシャルファイルの違いは何ですか？また，HiRDB はどちらのファイルシステムを使用していますか？

お答えします

通常ファイルはカーネルバッファを経由してデータを入出力しますが，キャラクタ型スペシャルファイルは，HiRDB のバッファから直接データを入出力します。

HiRDB では通常ファイルとキャラクタ型スペシャルファイルの両方を使用できます。キャラクタ型スペシャルファイルの使用を基本と考えていますが，次に示す場合は通常ファイルの方が性能が優れています。

- 大量検索を主体とする表を格納する RD エリアの HiRDB ファイル
- 作業表用ファイル

ただし，通常ファイルはシステムダウンに弱いので，次に示すファイルは必ずキャラクタ型スペシャルファイルに作成してください。

- 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
- システム用 RD エリアを構成する HiRDB ファイル
- 更新頻度の高いユーザ用 RD エリアを構成する HiRDB ファイル

なお、プリフェッチ機能をサポートしているため、大量検索主体の表を格納する RD エリアの構成ファイルもキャラクタ型スペシャルファイルで高速検索できます。

付録 A.2 HiRDB のシステム構成に関する質問

(1) HiRDB/Developer's Kit

質問

HiRDB/Developer's Kit はどんなときに必要ですか？

お答えします

HiRDB サーバがあるマシンで UAP を作成する場合は、HiRDB サーバに HiRDB/ Developer's Kit の機能が含まれているので必要ありません。HiRDB サーバがあるマシンとは別のマシンで UAP を開発する場合に必要です。

また、HiRDB サーバと異なるプラットフォームの UAP を作成する場合にも必要です。

(2) HiRDB に同梱されている製品

質問

HiRDB にバンドルされているプログラムプロダクトを教えてください。

お答えします

バンドルされているプログラムプロダクトを次に示します。

HP-UX, HI-UX/WE2, Solaris, AIX 5L 版の場合

バンドルされているプログラムプロダクトはありません。

Linux 版の場合

- HiRDB/Run Time (JDBC を含む)
- HiRDB Control Manager - Server ¹
- HiRDB Control Manager - Console ¹
- HiRDB Control Manager - Agent
- HiRDB Assist - Server
- HiRDB Assist - Client ¹
- HiRDB SQL Executer
- HiRDB Dataextractor
- HiRDB Datareplicator ²
- SORT

注 1

HiRDB Control Manager - Server, HiRDB Control Manager - Console, および HiRDB Assist - Client については、バンドルされるのは Windows 版です。

注 2

セット製品に含まれる HiRDB Datareplicator では、V7 対応の形名であっても、VR が 06-03 の場合があります。この製品は、HiRDB Datareplicator 06-03 相当の製品です。06-03 の機能の範囲で、HiRDB V7 と組み合わせてご利用になれます。

SORT は HiRDB でだけ使用でき、ほかの製品では使用できません。

付録 A.3 運用に関する質問

(1) JP1 との連携

質問

JP1 と連携はしているのですか？

お答えします

しています。HiRDB の開始、終了などのイベントを JP1/System Event Service (JP1 Version 6 以降では JP1/Base) に登録できます。登録したイベントは、JP1/Automatic Operation Monitor (JP1 Version 6 以降では JP1/Automatic Job Management System 2) で利用できます。ただし、Linux 版ではこれらの製品との連携はできません。

(2) サンプルを使用して環境作成したあとの、システム定義の変更

質問

サンプルを使用して環境を作成したあと、システム定義を変更する場合は、\$PDDIR/conf 下のシステム定義をそのまま変更すれば良いのですか？

お答えします

そのまま変更してください。その場合、HiRDB を正常終了させておく必要があります。

付録 A.4 トラブルに関する質問

(1) システム定義の最後の行に指定したオペランドが認識されない

質問

システム定義の最後の行に指定したオペランドが認識されていません。なぜですか？

お答えします

システム定義ファイルの最終行に改行があるかどうか確認してください。システム定義や、ユティリティの制御文の最終行には改行が必要です。改行がないと、正しく認識されません。

(2) 認可識別子やパスワードが正しく認識されない

質問

認可識別子やパスワードが正しく認識されていません。なぜですか？

お答えします

大文字、小文字が正しく認識されていない可能性があります。小文字を指定する場合は、引用符 (") で囲んでください。引用符で囲まないと、小文字を指定しても大文字として扱われます。

(例)

```
SELECT * FROM "user05".TABLE05;
```

(3) データベース再編成ユーティリティ実行時の KFPO00107-E エラー

質問

データベース再編成ユーティリティを実行すると、KFPO00107-E (write (olktdlr)) エラーとなります。

お答えします

OS の port が不足しています。データベース再編成ユーティリティ以外でも、HiRDB 開始時を含め、特に更新処理が多発する場合は HiRDB の port の使用頻度が上がるため、注意する必要があります。この現象が発生した場合には、少し時間をおいてコマンドを実行してください。

(4) データベース複製ユーティリティ実行時に、ディスク容量に余裕があるのにディスク容量不足となる

質問

データベース複製ユーティリティ実行時に、ディスク容量に余裕があるのにディスク容量不足のエラーとなります。なぜですか？

お答えします

ディスクの残容量に余裕があるのに、データベース複製ユーティリティ実行時にディスク容量不足を示す旨のメッセージが出力された場合は、バックアップファイルの大きさが 2GB を超えている可能性があります。HiRDB が扱えるファイルの大きさの上限は 2GB です。この場合は、次に示すどれかの対処をしてください。

対処方法 1

バックアップファイルを複数個指定してください。例えば、3GB 必要な場合はバックアップファイルを最低 2 個指定してください。

対処方法 2

UNIX ファイルシステム構築時にラージファイルを使用可能とするオプションを指定した UNIX ファイルシステム上の通常ファイルまたは HiRDB ファイルシステム領域を使用してください。HiRDB ファイルシステム領域のラージファイルの作成方法については、マニュアル「HiRDB Version 7 システム導入・設

計ガイド」を参照してください。

対処方法 3

キャラクタ型スペシャルファイル上に作成した HiRDB ファイルシステム領域にバックアップファイルを作成してください。HiRDB ファイルシステム領域のラージファイルの作成方法については、マニュアル「HiRDB Version 7 システム導入・設計ガイド」を参照してください。なお、ラージファイルをサポートしていない一部の HI-UX/WE2 では、最大 4GB の HiRDB ファイルシステム領域に納まる範囲のバックアップファイルを作成できます。

付録 B HiRDB のアンインストール

HiRDB をアンインストールする手順について次に示します。それぞれ、HiRDB/ シングルサーバをインストールしたマシン（サーバ側）、HiRDB/Run Time をインストールしたマシン（クライアント側）でアンインストールします。

なお、アンインストールする前に、必ず HiRDB を終了させてください。

付録 B.1 HiRDB/ シングルサーバのアンインストール手順 （サーバ側）

HiRDB/ シングルサーバをアンインストールする手順を次に示します。

1. CD-ROM をマウントしてください（以降、マウント先を /cdrom とします）。
2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してください。
3. < D > キーを押下してください。
現在インストールされている PP の一覧が表示されます。
4. カーソルで HiRDB/Single Server Version 7 を選択して、< スペース > キーを押下してください。
HiRDB/Single Server Version 7 の行頭に、その PP を選択したことを示す「<@>」が表示されます。
5. < D > キーを押下してください。
アンインストールが開始されます。アンインストールが終了すると、「Delete procedure completed.」のメッセージが表示されます。
6. < Q > キーを 2 回押下してください。

付録 B.2 HiRDB/Developer's Kit のアンインストール手順 （クライアント側）

HiRDB/Developer's Kit をアンインストールする手順を次に示します。

1. CD-ROM をマウントしてください（以降、マウント先を /cdrom とします）。
2. /cdrom/HPUX/SETUP /cdrom を実行して、Hitachi PP Installer を起動してください。
3. < D > キーを押下してください。
現在インストールされている PP の一覧が表示されます。
4. カーソルで HiRDB/Developer's Kit Version 6(32) を選択して、< スペース > キーを押下してください。
HiRDB/Developer's Kit Version 7 の行頭に、その PP を選択したことを示す「<@>」が表示されます。

5. < D > キーを押下してください。
アンインストールが開始されます。アンインストールが終了すると、「Delete procedure completed.」のメッセージが表示されます。
6. < Q > キーを 2 回押下してください。

付録 B.3 HiRDB SQL Executer のアンインストール手順 (サーバ, クライアント両方)

HiRDB SQL Executer をアンインストールする手順を次に示します。

1. CD-ROM をマウントしてください (以降, マウント先を /cdrom とします)。
2. /cdrom/HPUX/SETUP /cdrom を実行して, Hitachi PP Installer を起動してください。
3. < D > キーを押下してください。
現在インストールされている PP の一覧が表示されます。
4. カーソルで HiRDB SQL Executer を選択して, <スペース> キーを押下してください。
HiRDB SQL Executer の行頭に, その PP を選択したことを示す「<@>」が表示されます。
5. < D > キーを押下してください。
アンインストールが開始されます。アンインストールが終了すると、「Delete procedure completed.」のメッセージが表示されます。
6. < Q > キーを 2 回押下してください。

付録 B.4 HiRDB Control Manager のアンインストール手順

HiRDB Control Manager をアンインストールする手順を次に示します。

HiRDB Control Manager をアンインストールする場合, あらかじめ「管理 HiRDB の解除」を行い, サービスを停止しておく必要があります。

HiRDB Control Manager - Agent のアンインストール

1. CD-ROM をマウントしてください (以降, マウント先を /cdrom とします)。
2. /cdrom/HPUX/SETUP /cdrom を実行して, Hitachi PP Installer を起動してください。
3. < D > キーを押下してください。
現在インストールされている PP の一覧が表示されます。
4. カーソルで HiRDB Control Manager - Agent を選択して, <スペース> キーを押下してください。
HiRDB Control Manager - Agent の行頭に, その PP を選択したことを示す「<@>」が表示されます。
5. < D > キーを押下してください。

アンインストールが開始されます。アンインストールが終了すると、「Delete procedure completed.」のメッセージが表示されます。

6. < Q > キーを 2 回押下してください。

HiRDB Control Manager - Server のアンインストール

1. [コントロールパネル] - [アプリケーションの追加と削除] をダブルクリックしてください。
【アプリケーションの追加と削除のプロパティ】ダイアログが表示されます。
2. 「インストールと削除」を選択してください。
3. 一覧の中から「HiRDB Control Manager - Server」を選択し、「追加と削除」をクリックしてください。
【ファイル削除の確認】ダイアログが表示されます。
4. 「はい」をクリックしてください。
アンインストールが始まります。しばらくすると、【コンピュータからプログラムを削除】ダイアログにアンインストールが終わりましたというメッセージが表示され、「OK」がアクティブになります。
5. 「OK」をクリックしてください。
アンインストール終了です。
6. アンインストールしてもインストールディレクトリ「hirdbCMS」下にファイルが残ります。エクスプローラなどを使用して、C:\Program Files\hitachi ディレクトリ下の「hirdbCMS」ディレクトリを削除してください。

HiRDB Control Manager - Console のアンインストール

1. [コントロールパネル] - [アプリケーションの追加と削除] をダブルクリックしてください。
【アプリケーションの追加と削除のプロパティ】ダイアログが表示されます。
2. 「インストールと削除」を選択してください。
3. 一覧の中から「HiRDB Control Manager - Console」を選択し、「追加と削除」をクリックしてください。
【ファイル削除の確認】ダイアログが表示されます。
4. 「はい」をクリックしてください。
アンインストールが始まります。しばらくすると、【コンピュータからプログラムを削除】ダイアログにアンインストールが終わりましたというメッセージが表示され、「OK」がアクティブになります。
5. 「OK」をクリックしてください。
アンインストール終了です。

付録 C 用語解説

(英字)

CONNECT 権限

HiRDB を利用するために必要な権限です。CONNECT 権限を持たないユーザが HiRDB を利用しようとするとうエラーになります。

DBA 権限

ユーザ権限を与えるためおよび取り消すために必要な権限です。権限を管理する人が持つ権限です。HiRDB 管理者は DBA 権限を持っています。DBA 権限を持つユーザは、次に示す権限を持っています。

- CONNECT 権限
- すべての HiRDB ユーザに対するスキーマ定義権限

FIX 属性

行長が固定の表に付ける属性のことです。

HiRDB.ini ファイル

HiRDB/クライアントから実行するユティリティや HiRDB SQL Executer が HiRDB サーバに接続するために必要な情報を設定しておくファイルです。このファイルは、サーバのマシンおよびクライアントのマシンの両方に必要です。

HiRDB Control Manager

HiRDB Control Manager は、システムログファイルの操作やバックアップの取得などの運用を Windows の GUI で実行するための運用支援プログラムです。HiRDB Control Manager は、次の三つのプログラムプロダクトで構成されています。

- HiRDB Control Manager - Console
- HiRDB Control Manager - Server
- HiRDB Control Manager - Agent

HiRDB Control Manager を使用するためには、Windows マシンが必要です。

HiRDB Control Manager - Agent

HiRDB Control Manager のエージェントです。HiRDB Control Manager - Server を経由して、HiRDB Control Manager - Console の命令を受け、同一マシンで稼働している HiRDB サーバに実行させます。

HiRDB Control Manager - Console で操作したい HiRDB サーバがインストールされているマシンにインストールします。

HiRDB Control Manager - Console

HiRDB Control Manager のクライアントです。GUI を使って HiRDB の開始、終了、バックアップの取得などの操作ができます。

HiRDB Control Manager - Server

HiRDB Control Manager のサーバです。HiRDB Control Manager - Console からの処理命令を HiRDB Control Manager - Agent (HiRDB サーバ) に送ります。

HiRDB SQL Executer

会話形式で SQL を実行するプログラムです。さらに、独自のコマンドなども用意しています。

HiRDB 管理者

HiRDB の運用コマンドの実行者で、Administrator 権限を持つユーザ ID でログインしたユーザのことです。HiRDB のディレクトリおよびファイルの所有者です。

HiRDB/ クライアント

HiRDB/Developer's Kit, または HiRDB/Run Time をインストールしたマシンのことをいいます。

HiRDB システム定義

HiRDB システムの構成や環境を定義したものを、HiRDB システム定義といいます。HiRDB システム定義には、システム共通定義、ユニット制御情報定義、およびシングルサーバ定義があります。

HiRDB ファイル

表、インデクス、障害発生時にシステムの状態を回復させるのに必要な情報など、HiRDB のさまざまな情報を格納するための、HiRDB 専用のファイルのことです。

HiRDB ファイルシステム領域

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、および RD エリアを作成する領域のことを HiRDB ファイルシステム領域といいます。また、HiRDB ファイルシステム領域内に作成されるファイルを HiRDB ファイルといいます。HiRDB ファイルシステム領域は、それぞれシステムファイル用、作業表用ファイル用、および RD エリア用の用途ごとに作成します。

RD エリア

データの格納単位の一つで、1 ~ 16 個の HiRDB ファイルから構成されます。RD エリアには、次に示すものがあります。

- マスタディレクトリ用 RD エリア
- データディクショナリ用 RD エリア
- データディレクトリ用 RD エリア
- ユーザ用 RD エリア

なお、使用する機能によっては上記の RD エリアのほかに、別の種類の RD エリアが必要になることがあります。

RD エリアの自動増分

RD エリアの容量が不足したとき自動的に RD エリアを拡張する機能です。

pdinit コマンドの create rdarea 文で設定します。

RD エリア利用権限

私用 RD エリアを利用するために必要な権限です。公用 RD エリアに表またはインデクスを作成する場合は、RD エリア利用権限は不要です。

(ア行)

アクセス権限

表をアクセスするために必要な権限です。アクセス権限は表単位に設定し、次に示す 4 種類があります。

- SELECT 権限
- INSERT 権限
- DELETE 権限
- UPDATE 権限

アンロード済み状態

システムログファイルの状態の一つです。取得したシステムログをアンロードログファイルにアンロードしたあとの状態のことです。

アンロード待ち状態

システムログファイルの状態の一つです。取得したシステムログをアンロードログファイルにアンロードしていない状態のことです。

アンロードログファイル

システムログファイルの内容 (システムログ) を、`pdlogunld` コマンドでアンロードして作成したファイルのことです。

インデクス

表を検索するためのキーとして列に付けた索引のことで、キーとキー値から成ります。キーとは列の内容を示した列名のこと、キー値とは列の値のことです。

(カ行)

グローバルバッファプール

ディスク上の RD エリアに格納されているデータを入出力するためのバッファの集まりのことで、共用メモリ上に確保されます。RD エリアまたはインデクスには、必ずグローバルバッファプールを割り当てます。

現用

システムログファイルの状態の一つです。現在、システムログが出力されているファイルです。

更新前ログ取得モード

UAP またはユティリティを実行するときのデータベースの更新ログ取得方式の一つです。UAP またはユティリティが RD エリアの内容を更新するときに、ロールバックに必要なデータベース更新ログだけを取得する方式のことです。

公用 RD エリア

HiRDB に登録されているすべてのユーザが利用できるユーザ用 RD エリアです。

(サ行)

サーバ

HiRDB のデータベース管理システムの特定の機能の単位です。

再編成

表データをいったんアンロードデータファイルに吸い上げます。これを表データのアンロードといえます。そのあと、表にデータを格納し直します。これを表データのリロードといえます。これら全体の処理を表の再編成といえます。

作業表用ファイル

SQL 文を実行するときに必要とする一時的な情報を格納するファイルのことです。

システムファイル

障害時にシステムの状況を回復するための情報などを格納するためのファイルです。次に示すファイルから構成されています。

- 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

システム用 RD エリア

次に示す RD エリアの総称です。

- マスタディレクトリ用 RD エリア
- データディレクトリ用 RD エリア
- データディクショナリ用 RD エリア

システムログファイル

データベースの更新履歴情報を格納するためのファイルです。一般的にはジャーナルファイルとも呼ばれています。データベースの更新履歴情報をシステムログといえます。システムログは、HiRDB または UAP が異常終了した場合、HiRDB がデータベースを回復するときに使用します。また、ユーザがデータベースを回復するときの入力情報にも使用します。

実表

実際にある表のことです。

自動ログアンロード

システムログファイルのアンロード作業を自動化する機能です。

私用 RD エリア

権限があるユーザだけが使用できるユーザ用 RD エリアです。

シンクポイントダンプファイル

HiRDB が異常終了した場合、システムログだけで回復処理をすると、HiRDB 開始からのすべてのシステムログが必要となり、回復に多大な時間が掛かります。そこで、HiRDB 稼働中に一定の間隔でポイントを設定して、そのポイントで回復する必要がある HiRDB 管理情報を保存することで、ポイント以前のシステムログは不要になり、回復時間を短縮できます。このポイントで取得する HiRDB 管理情報を格納するファイルを、シンクポイントダンプファイルといえます。

シングルサーバ

1 台のマシンで動作するデータベース管理システムのことです。

スキーマ

一人のユーザが持つ、表、インデクス、ストアドプロシジャ、および権限を包括した概念です。

スキーマ定義権限

スキーマを定義するために必要な権限です。

ステータスファイル

HiRDB を再開するとき必要とするシステムステータス情報を格納するファイルのことです。次に示すファイルから構成されます。

- サーバ用ステータスファイル
- ユニット用ステータスファイル

スワップ

現用のシステムログファイルが変わることです。

セグメント

連続した複数のページから構成される、領域の割り当て単位です。

(夕行)

ディクショナリ表

HiRDB が作成して管理する表のことです。ディクショナリ表はデータディクショナリ用 RD エリアに格納されています。ディクショナリ表には、表やインデクスなどの、HiRDB システムが管理する定義情報が格納されています。ディクショナリ表は、ユーザが操作系 SQL で参照し、表、インデクスなどの定義情報を確認するために使用します。

データディクショナリ用 RD エリア

定義系 SQL の解析結果を管理するディクショナリ表、およびディクショナリ表のインデクスを格納するための RD エリアです。

データディレクトリ用 RD エリア

インデクスに関する情報 (列名、データ型など) を HiRDB のデータ形式で格納するための RD エリアです。

データロード

表にデータを格納することをいいます。データベース作成ユーティリティ (pdload) で実行します。

(八行)

バックアップ取得モード

pdcopy コマンドの -M オプションでバックアップ取得モードを選択します。バックアップ取得モードには、次に示す三つの種類があります。

- x
バックアップ取得中、バックアップ対象 RD エリアの参照および更新ができません。バックアップ取得対象 RD エリアを pdhold コマンドで閉塞かつクローズ状態にする必要があります。バックアップだけでデータベースをバックアップ取得時点で回復できます。
- r
バックアップ取得中、バックアップ取得対象の RD エリアの参照はできますが、更新はできません。バックアップだけでデータベースをバックアップ取得時点で回復できます。
- s
バックアップ取得中、バックアップ取得対象 RD エリアの参照および更新ができます。データベースを回復するときに、バックアップおよびバックアップ取得中のシステムログが必要になります。

ページ

データの格納単位の一つで、データベースの入出力動作の最小単位です。ページには次の種類があります。

- データページ : 表の中の行を格納するページです。
- インデクスページ : インデクスのキー値を格納するページです。
- ディレクトリページ : RD エリアの状態の管理情報を格納するページです。

(マ行)

マスタディレクトリ用 RD エリア

ディクショナリ表、ユーザが作成した表やインデクスなどを格納する RD エリアの情報、RD エリアの登録場所（サーバ）の情報などを管理する RD エリアのことです。

マルチ HiRDB

一つのサーバマシンで複数の HiRDB サーバを稼働させる形態のことです。

(ヤ行)

ユーザ権限

データベースを作成したり、検索したりするのに必要な権限のことです。ユーザ権限がないと、データベースにアクセスできません。ユーザ権限には次に示す種類があります。

- DBA 権限
- CONNECT 権限
- スキーマ定義権限
- RD エリア利用権限
- アクセス権限

ユーザ用 RD エリア

ユーザが作成する表とインデクスを格納するための RD エリアのことです。

ユニット

一つの PC 内の、HiRDB の動作環境のことです。

(ラ行)

ログ取得モード

UAPまたはユティリティを実行するときのデータベースの更新ログ取得方式の一つです。UAPまたはユティリティがRDエリアの内容を更新するときに、ロールバックおよびロールフォワードに必要なデータベース更新ログを取得する方式のことです。

ログレスモード

UAPまたはユティリティを実行するときのデータベースの更新ログ取得方式の一つです。UAPまたはユティリティがRDエリアの内容を更新するときに、データベース更新ログを取得しない方式のことです。

索引

C

COBOL言語でアプリケーションを作成する 241

CONNECT 権限 136, 266

C 言語でアプリケーションを作成する 233

D

DAT のバックアップを使用して回復する 116

DBA 権限 136, 266

DELETE 権限 136

F

FIX 属性 266

G

GRANT 文 137

H

HiRDB.ini ファイル 266

HiRDB/Developer's Kitのアンインストール 263

HiRDB/Run Time をインストールする (Windows マシン) 23

HiRDB/Single Server のアンインストール 263

HiRDB/ クライアント 267

HiRDB/ クライアントの環境設定をする [COBOL 言語] 242

HiRDB/ クライアントの環境設定をする [C 言語] 234

HiRDB/ クライアントの環境設定をする [Java 言語] 250

HiRDB/ クライアントをインストールする 217

HiRDB/ シングルサーバのインストール 18

HiRDB/ シングルサーバの構成 31

HiRDB Control Manager 144, 266

HiRDB Control Manager - Agent 21, 266

HiRDB Control Manager - Console 21, 266

HiRDB Control Manager - Server 21, 267

HiRDB Control Manager - Agentの環境設定 53

HiRDB Control Manager のアンインストール 264

HiRDB Control Manager のインストール手順 22

HiRDB Control Manager をインストールする 21

HiRDB Control Manager を開始する 145

HiRDB Control Manager を終了する 146

HiRDB SQL Executer 267

HiRDB SQL Executer のアンインストール 264

HiRDB SQL Executer のインストール 21

HiRDB SQL Executer の環境設定 (Windows マシン) 57

HiRDB SQL Executer をインストールする 218

HiRDB SQL Executer をインストールする (Windows マシン) 24

HiRDB 運用ディレクトリを設定する 24

HiRDB が出力するメッセージ 9

HiRDB 管理者 267

HiRDB 管理者専用グループの作成 18

HiRDB 管理者の登録 18

HiRDB システム・ユニット・サーバの開始 81

HiRDB システム・ユニット・サーバの終了 81

HiRDB システム定義 7, 267

HiRDB システム定義を作成する 30

HiRDB システムの構築手順 13

HiRDB システムを構築するときのユーザ名 14

HiRDB のアンインストール 263

HiRDB の開始 81

HiRDB の稼働状況を確認する 44

HiRDB の環境設定をする (コマンドでの実行) 26
 HiRDB の環境設定をする (システムジェネレータでの実行) 46
 HiRDB の環境変数を設定する 46
 HiRDB のシステム構成に関する質問 259
 HiRDB の終了 82
 HiRDB ファイル 267
 HiRDB ファイルシステム領域 7, 267
 HiRDB ファイルシステム領域の構成 27
 HiRDB ファイルシステム領域を作成する 27
 HiRDB をインストールする 15
 HiRDB を開始する 39
 HiRDB を正常開始する 84, 151
 HiRDB を正常終了する 83, 150
 HiRDB を登録する 148
 hosts ファイルの設定 26, 219

I

INSERT 権限 136
 IP アドレスの設定 17

J

Java 言語 (SQLJ) でアプリケーションを作成する 249

O

OS ごとの操作の差異 9
 OS パラメタ 16

P

pdcopy コマンド 94
 pddef 60
 pdfmkfs コマンド 28
 pdinit 39
 pdload 63
 pdloginit -d spd コマンド 36
 pdloginit -d sys コマンド 34
 pdlogunld コマンド 86, 88
 pdmod コマンド 124, 128
 pdrorg コマンド 120

pdrstr コマンド 94, 106
 pdstart コマンド 81
 pdstop コマンド 82
 pdstsinit コマンド 37, 38
 pdsys 31
 pdutsys 32

R

RD エリア 8, 267
 RD エリア単位のバックアップを取得する 100
 RD エリアの拡張手順 124
 RD エリアの自動増分 267
 RD エリアの追加手順 128
 RD エリアのバックアップを取得する 166
 RD エリア利用権限 136, 267
 RD エリアを拡張する 124, 125, 192
 RD エリアを作成する 39
 RD エリアを追加する 128, 198

S

sds1 33
 SELECT 権限 136
 SQLJ 226
 SQL トランスレータでソースプログラムを変換する [Java 言語] 254
 SQL プリプロセサでソースプログラムを変換する [COBOL 言語] 246
 SQL プリプロセサでソースプログラムを変換する [C 言語] 238

U

UAP 226
 UAP 作成の流れ 227
 UAP のソースプログラムを作成する [COBOL 言語] 243
 UAP のソースプログラムを作成する [C 言語] 234
 UAP のソースプログラムを作成する [Java 言語] 250
 UPDATE 権限 136

あ

空きディスク容量 17
 アクセス権限 136, 268
 アクセス権限がある表を検索する 138
 アクセス権限がない表を検索する 139
 アンロード状態のチェックを解除する運用
 88
 アンロード済み状態 268
 アンロード待ち状態 88, 268
 アンロードレスシステムログ運用 88
 アンロードログファイル 86, 268

い

異常終了 82
 インストール前の準備をする 17
 インデクス 268

う

埋込み型 UAP 226
 運用に関する質問 260

か

カーネルパラメタ 16
 開始 81, 150
 開始モード 81
 開発環境を準備する〔COBOL 言語〕242
 開発環境を準備する〔C 言語〕234
 開発環境を準備する〔Java 言語〕250
 回復 106, 175
 学習用の HiRDB システムを構築するときの
 マシン環境 3
 環境設定に関する質問 258
 環境変数 PATH の設定 220
 管理 HiRDB 148

き

基礎知識〔COBOL 言語〕241
 基礎知識〔C 言語〕233
 基礎知識〔Java 言語〕249
 強制開始 82
 強制終了 82

共用メモリ 17

く

クライアントから表を操作する 221
 クライアント環境定義の設定 57, 219
 クライアントの環境設定 217
 クライアントの接続手順 216
 グローバルバッファプール 268

け

計画停止 82
 権限がないユーザで表を検索する 139
 権限を追加する 210
 現用 88, 268

こ

更新前ログ取得モード 268
 構築する HiRDB システムの構成 7
 公用 RD エリア 137, 268
 コマンドの実行 9
 コンパイルを実行する〔COBOL 言語〕247
 コンパイルを実行する〔C 言語〕238
 コンパイルを実行する〔Java 言語〕254

さ

サーバ 269
 サーバ用ステータスファイル 38
 再開 81
 最新の状態に回復する 111, 178
 最新の状態への回復 108
 再編成 120, 186, 269
 作業表用ファイル 269
 作成した UAP を実行する〔COBOL 言語〕
 247
 作成した UAP を実行する〔C 言語〕239
 作成した UAP を実行する〔Java 言語〕254
 作成するアプリケーションの仕様 230
 サンプル UAP の機能 230
 サンプル UAP の処理の流れ 231

し

システム共通定義 31
 システムジェネレータ (pdgen) を実行する
 47
 システムファイル 269
 システムファイルを作成する 34
 システム用 RD エリア 269
 システム用 RD エリアのバックアップ取得時
 点への回復 108
 システムログ 86, 153
 システムログとデータベース回復の関係 87
 システムログのアンロード手順 91
 システムログの運用の種類 87
 システムログファイル 7, 269
 システムログファイルの作成 34
 システムログファイルの状態変化 89
 システムログファイルのスワップ 88
 システムログをアンロードする 91, 154
 システムログをアンロードする運用 88
 実表 269
 自動ログアンロード 269
 自動ログアンロード機能 90
 自動ログアンロード機能を使用する場合の
 データベースの回復手順 107
 自動ログアンロード機能を使用する場合の
 バックアップの取得手順 97
 終了 81, 150
 終了モード 82
 私用 RD エリア 137, 269
 使用するプログラムプロダクト 4
 使用できる言語と開発環境 228
 新規ユーザを登録する 137, 209
 シンクポイントダンプファイル 8, 269
 シンクポイントダンプファイルの作成 36
 シングルサーバ 270
 シングルサーバ定義 33

す

スキーマ 270
 スキーマ定義権限 136, 270
 ステータスファイル 8, 270
 ステータスファイルの作成 37

スワップ 270

せ

正常開始 81
 正常終了 82
 製品の画面や設定値について 9
 セグメント 270
 全 RD エリア (HiRDB システム) のバック
 アップを取得する 158
 全 RD エリアのバックアップを取得する 98

て

ディクショナリ表 270
 データディクショナリ用 RD エリア 270
 データディレクトリ用 RD エリア 270
 データベース回復の概要 94
 データベース回復ユティリティ 94, 106
 データベース構成変更ユティリティ
 124, 128
 データベース再編成ユティリティ 120
 データベース作成ユティリティ 63
 データベース初期設定ユティリティ 39
 データベース定義ユティリティ 60
 データベースの回復手順 106
 データベース複写ユティリティ 94
 データロード 270

と

トラブルに関する質問 260

は

バックアップ 94, 157
 バックアップ取得時点に回復する 108, 175
 バックアップ取得モード 95, 270
 バックアップの取得単位 94
 バックアップの取得手順 97
 バックアップを DAT に取得する 103

ひ

表の行を更新する 75
 表の行を削除する 76

表の再編成 120, 186
表の再編成手順 121
表の所有者を調べる 140
表へ行を挿入する 74, 222
表ヘデータロードをする 63
表を検索する 74
表を再編成する 121, 186
表を作成する 59
表を操作する 74
表を定義する 60, 221

へ

ページ 271

ほ

ホームディレクトリ (\$HOME) の記述 9
ほかの所有者の表を検索する 222

ま

マスタディレクトリ用 RD エリア 271
マスタディレクトリ用 RD エリアの回復 108
マルチ HiRDB 271

ゆ

ユーザ権限 136, 209, 271
ユーザ名を登録する 59
ユーザ用 RD エリア 271
ユーザを登録する 136
ユニット 271
ユニット制御情報定義 32
ユニット用ステータスファイル 37

ろ

ログ関係ファイルのアンロード 87
ログ取得モード 272
ログレスモード 272

(株) 日和 出版センター 行き

FAX 番号 0120-210-454 (フリーダイヤル)

日立マニュアル注文書

ご注文日	年 月 日
送付先ご住所	〒
お客様名 (団体名, 又は法人名など)	
お名前	
電話番号	()
FAX 番号	()

資料番号	マニュアル名	数量
合計		

マニュアルのご注文について、ご不明な点は
(株) 日和 出版センター (☎03-5281-5054) へお問い合わせください。