
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
(Windows(R) 用)

解説・手引・操作書

3020-6-054-30

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 : Windows NT , Windows 2000 , Windows XP Professional , Windows Server 2003

P-2462-7174 HiRDB/Single Server Version 7 07-00

適用 OS : Windows Me , Windows NT , Windows 2000 , Windows XP , Windows Server 2003

P-2662-1174 HiRDB/Run Time Version 7 07-00

P-2662-1274 HiRDB/Developer's Kit Version 7 07-00

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

R-15434-31 HiRDB Control Manager - Agent 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「ご使用になる前に」または「Readme ファイル」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

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。

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。

Microsoft Access 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。

OLE は、米国 Microsoft Corp. が開発したソフトウェア名称です。

OLE は、Object Linking and Embedding の略です。

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

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。

Visual C++ 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 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。

発行

1999 年 7 月 (第 1 版) 3020-6-054

2004 年 3 月 (第 4 版) 3020-6-054-30

著作権

All Rights Reserved. Copyright (C) 1999, 2004, Hitachi, Ltd.

変更内容

変更内容 (3020-6-054-30)

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
演習に使用する OS を Windows 2000 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。	2.2.4 , 5
コマンドを使ったセットアップおよびバッチを使ったセットアップの記述を , 簡易セットアップツールを使ったセットアップの記述に変更した。	2.3
Java 言語 (SQJ) によるアプリケーション開発の記述を追加した。	7.5

単なる誤字・脱字などはお断りなく変更しました。

変更内容 (3020-6-054-20)

追加・変更機能
HiRDB/Single Server Version 5.0 を中心とした記述から , HiRDB/Single Server Version 6 を中心とした記述に変更した。
RD エリアの自動増分に関する記述を追加した。
自動ログアンロードに関する記述を追加した。
演習に使用する OS を Windows NT 4.0 , Windows 95 から , Windows 2000 , Windows 98 に変更した。

変更内容 (3020-6-054-10)

追加・変更機能
HiRDB/Single Server Version 4.0 を中心とした記述から , HiRDB/Single Server Version 5.0 を中心とした記述に変更した。
DAT を使ったバックアップおよびデータベースの回復の演習を追加した。
HiRDB Assist で root 以外のユーザで接続できるようになった。
HiRDB Assist を使用した RD エリアの拡張・追加方法に関する説明を追加した。
UAP の作成に関する説明の追加に伴い , クライアントにインストールするプログラムを HiRDB/Run Time から HiRDB/Developer's Kit に変更した。
クライアント / サーバ環境での UAP の作成に関する説明を追加した。

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

対象読者

このマニュアルは、Windows 版 HiRDB を初めて使用される方、Windows 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 Control Manager で HiRDB を運用してみましょう

HiRDB Control Manager を使用した HiRDB の運用方法について説明しています。

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。

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。

はじめに

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 (Windows(R) 用)(3020-6-271)
- HiRDB Version 7 システム導入・設計ガイド (Windows(R) 用)(3020-6-272)
- HiRDB Version 7 システム定義 (Windows(R) 用)(3020-6-273)
- HiRDB Version 7 システム運用ガイド (Windows(R) 用)(3020-6-274)
- HiRDB Version 7 コマンドリファレンス (Windows(R) 用)(3020-6-275)
- HiRDB Version 7 UAP 開発ガイド (Windows(R) 用)(3020-6-276)
- HiRDB Version 7 SQL リファレンス (Windows(R) 用)(3020-6-277)
- HiRDB Version 7 メッセージ (Windows(R) 用)(3020-6-278)
- HiRDB Version 6 解説 (Windows(R) 用)(3020-6-121)
- HiRDB Version 6 システム導入・設計ガイド (Windows(R) 用)(3020-6-122)
- HiRDB Version 6 システム定義 (Windows(R) 用)(3020-6-123)
- HiRDB Version 6 システム運用ガイド (Windows(R) 用)(3020-6-124)
- HiRDB Version 6 コマンドリファレンス (Windows(R) 用)(3020-6-125)
- HiRDB Version 6 UAP 開発ガイド (Windows(R) 用)(3020-6-126)
- HiRDB Version 6 SQL リファレンス (Windows(R) 用)(3020-6-127)
- HiRDB Version 6 メッセージ (Windows(R) 用)(3020-6-128)

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。

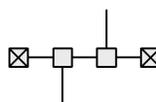
●PC/WSなどのマシン



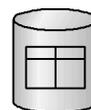
●プログラム



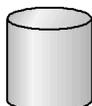
●バス形のLAN



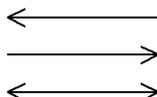
●表



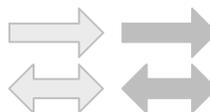
●ファイル



●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す。

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 Access(R) 2000	Microsoft Access
Microsoft(R) Visual C++(R)	Microsoft Visual C++
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

製品名称または名称	表記
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
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 (かしよ)
- 同梱 (どうこん)
- 閉塞 (へいそく)
- 拗音符 (ようおんぷ)

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	6
1.4	このマニュアルを読むときの注意事項	8
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	14
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	14
2.2.2	HiRDB/ シングルサーバをインストールする	18
2.2.3	HiRDB SQL Executer をインストールする	23
2.2.4	HiRDB Control Manager をインストールする	23
2.3	HiRDB の環境設定をする (簡易セットアップツールでの実行)	27
2.3.1	作成する環境	27
2.3.2	簡易セットアップツールを実行する	32
2.3.3	定義ファイルを変更したい場合は	41
2.4	表を作成する	46
2.4.1	ユーザ名を登録する	46
2.4.2	表を定義する	48
2.4.3	表ヘデータロードをする	51
3	表を操作してみましょう	59
3.1	表を操作するためには	60
3.2	HiRDB SQL Executer を起動する	61
3.3	表を操作する	62
3.3.1	演習 1 (表を検索する)	62
3.3.2	演習 2 (表へ行を挿入する)	64
3.3.3	演習 3 (表の行を更新する)	65

3.3.4 演習 4 (表の行を削除する)	66
3.4 HiRDB SQL Executer を終了する	68

4

HiRDB を運用してみよう	69
4.1 この章を読むに当たって	70
4.2 HiRDB を開始, 終了する	71
4.2.1 基礎知識	71
4.2.2 演習 1 (HiRDB を正常終了する)	73
4.2.3 演習 2 (HiRDB を正常開始する)	74
4.3 システムログを操作する	76
4.3.1 基礎知識	76
4.3.2 演習 (システムログをアンロードする)	81
4.4 バックアップを取得する	84
4.4.1 基礎知識	84
4.4.2 演習 1 (全 RD エリアのバックアップを取得する)	89
4.4.3 演習 2 (バックアップを DAT に取得する)	91
4.4.4 演習 3 (自動ログアンロード機能を使っている場合に全 RD エリアのバックアップを取得する)	93
4.5 データベースを回復する	97
4.5.1 基礎知識	97
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)	99
4.5.3 演習 2 (DAT のバックアップを使用して回復する)	103
4.5.4 演習 3 (自動ログアンロード機能を使っている場合に回復する)	107
4.6 表を再編成する	115
4.6.1 基礎知識	115
4.6.2 演習 (表を再編成する)	116
4.7 RD エリアを拡張する	119
4.7.1 基礎知識	119
4.7.2 演習 (RD エリアを拡張する)	120
4.8 RD エリアを追加する	123
4.8.1 基礎知識	123
4.8.2 演習 (RD エリアを追加する)	123
4.9 ユーザ権限を与える (ユーザを登録する)	132
4.9.1 基礎知識	132
4.9.2 演習 1 (新規ユーザを登録する)	133
4.9.3 演習 2 (アクセス権限がある表を検索する)	134

4.9.4	演習 3 (アクセス権限がない表を検索する)	135
4.9.5	演習 4 (権限がないユーザで表を検索する)	136
4.9.6	演習 5 (表の所有者を調べる)	137

5

HiRDB Control Manager で HiRDB を運用してみましよう		139
5.1	この章を読むに当たって	140
5.2	HiRDB Control Manager を開始, 終了する	141
5.2.1	HiRDB Control Manager を開始する	141
5.2.2	HiRDB Control Manager を終了する	142
5.3	HiRDB を登録する	143
5.4	HiRDB を開始, 終了する	145
5.4.1	演習 1 (HiRDB を正常終了する)	145
5.4.2	演習 2 (HiRDB を正常開始する)	146
5.5	システムログを操作する	148
5.5.1	演習 (システムログをアンロードする)	149
5.6	バックアップを取得する	152
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)	153
5.6.2	演習 2 (自動ログアンロード機能を使っている場合に全 RD エリアのバックアップを取得する)	161
5.7	データベースを回復する	163
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)	163
5.7.2	演習 2 (自動ログアンロード機能を使っている場合に回復する)	166
5.8	表を再編成する	174
5.8.1	演習 (表を再編成する)	174
5.9	RD エリアを拡張する	180
5.9.1	演習 (RD エリアを拡張する)	180
5.10	RD エリアを追加する	186
5.10.1	演習 (RD エリアを追加する)	186
5.11	ユーザ権限を与える (ユーザを登録する)	197
5.11.1	演習 1 (新規ユーザを登録する)	197
5.11.2	演習 2 (権限を追加する)	198

6

クライアントを接続してみましよう		203
6.1	この章を読むに当たって	204
6.2	クライアントの環境設定をする	205

6.2.1	HiRDB/クライアントをインストールする	205
6.2.2	HiRDB SQL Executer をインストールする	208
6.2.3	hosts ファイルの設定をする	208
6.2.4	クライアント環境定義の設定をする	209
6.2.5	環境変数 PATH の設定をする	209
6.3	クライアントから表を操作する	211
6.3.1	演習 1 (表を定義する)	211
6.3.2	演習 2 (表へ行を挿入する)	212
6.3.3	演習 3 (ほかの所有者の表を検索する)	212
6.4	ODBC 対応のアプリケーションプログラムからアクセスする	214
6.4.1	ODBC ドライバをインストールする	214
6.4.2	Microsoft Access から HiRDB をアクセスする	216
7	アプリケーションを作成してみましょう	219
7.1	この章を読むに当たって	220
7.2	作成するアプリケーションの仕様	223
7.2.1	サンプル UAP の機能	223
7.2.2	サンプル UAP の処理の流れ	224
7.3	C 言語でアプリケーションを作成する	225
7.3.1	基礎知識	225
7.3.2	HiRDB/クライアントの環境設定をする	226
7.3.3	開発環境を準備する	226
7.3.4	UAP のソースプログラムを作成する	226
7.3.5	SQL プリプロセサでソースプログラムを変換する	229
7.3.6	コンパイルを実行する	230
7.3.7	作成した UAP を実行する	231
7.4	COBOL 言語でアプリケーションを作成する	232
7.4.1	基礎知識	232
7.4.2	HiRDB/クライアントの環境設定をする	233
7.4.3	開発環境を準備する	233
7.4.4	UAP のソースプログラムを作成する	233
7.4.5	SQL プリプロセサでソースプログラムを変換する	237
7.4.6	コンパイルを実行する	237
7.4.7	作成した UAP を実行する	238
7.5	Java 言語 (SQLJ) でアプリケーションを作成する	240
7.5.1	基礎知識	240

7.5.2	HiRDB/ クライアントの環境設定をする	241
7.5.3	開発環境を準備する	241
7.5.4	UAP のソースプログラムを作成する	241
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	245
7.5.6	コンパイルを実行する	245
7.5.7	作成した UAP を実行する	245

付録 247

付録 A	Q & A	248
付録 A.1	インストールに関する質問	248
付録 A.2	環境設定に関する質問	248
付録 A.3	HiRDB のシステム構成に関する質問	249
付録 A.4	運用に関する質問	250
付録 A.5	トラブルに関する質問	251
付録 B	HiRDB のアンインストール	254
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)	254
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)	255
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)	255
付録 B.4	HiRDB Control Manager のアンインストール手順 (サーバ側)	256
付録 C	用語解説	258

索引 265

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	6
図 2-1	HiRDB システムの構築手順	12
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	14
図 2-3	簡易セットアップツールで作成する HiRDB ファイルシステム領域	28
図 2-4	簡易セットアップツールで作成する HiRDB/ シングルサーバの構成	30
図 2-5	簡易セットアップツールで作成する RD エリアの構成	31
図 2-6	定義する表	46
図 2-7	GOODS 表へのデータロードの概要	52
図 2-8	STOCK 表へのデータロードの概要	55
図 4-1	システムログとデータベース回復の関係	77
図 4-2	システムログファイルの状態変化	79
図 4-3	システムログをアンロードする手順	81
図 4-4	データベース回復の概要	84
図 4-5	pdcopy コマンドの処理結果リストの出力例	87
図 4-6	バックアップを取得する手順	88
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	89
図 4-8	データベースを回復する手順	98
図 4-9	表の再編成	115
図 4-10	表を再編成する手順	116
図 4-11	RD エリアを拡張する手順	120
図 4-12	RD エリアを追加する手順	123
図 6-1	クライアントの接続手順	204
図 7-1	UAP 作成の流れ	221
図 7-2	サンプル UAP の機能の概要	223
図 7-3	サンプル UAP の流れ	224
図 7-4	UAP 生成の流れ (C 言語の場合)	225
図 7-5	UAP 生成の流れ (COBOL 言語の場合)	232
図 7-6	UAP 生成の流れ (Java 言語の場合)	240

表目次

表 1-1	使用するプログラムプロダクト	5
表 2-1	システムログに関する運用方法	32
表 4-1	HiRDB の開始モード	72
表 4-2	HiRDB の終了モード	72
表 4-3	システムログの運用の種類	78
表 4-4	バックアップの取得単位	85
表 4-5	バックアップ取得モード	85
表 4-6	ユーザ権限の種類	132
表 7-1	使用できる言語と開発環境	222

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，Windows 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

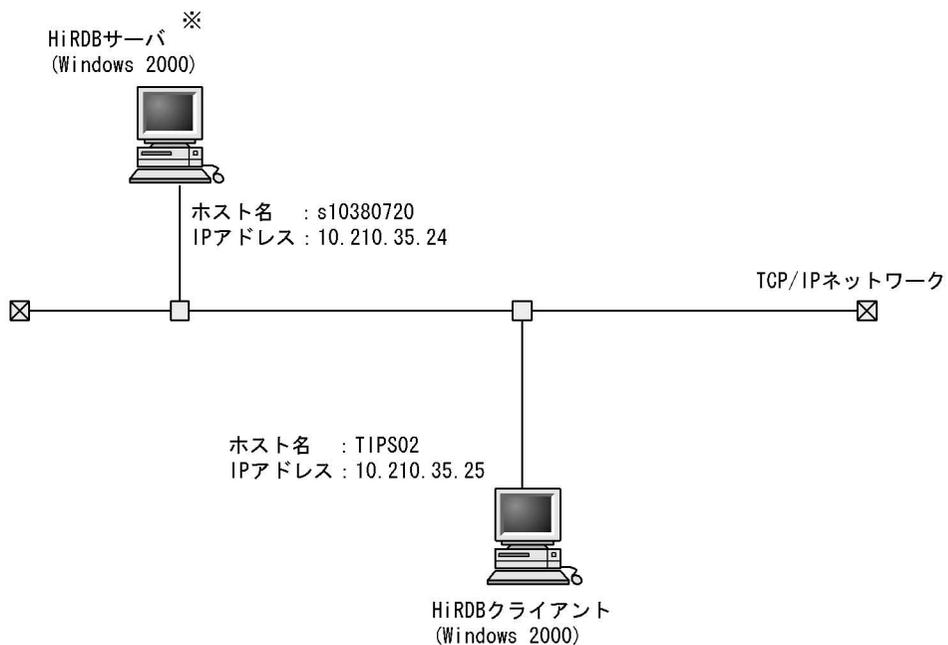
このマニュアルは，通読型のマニュアルです。

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 Windows 2000，HiRDB クライアントは Windows 2000 です。

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※
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。

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

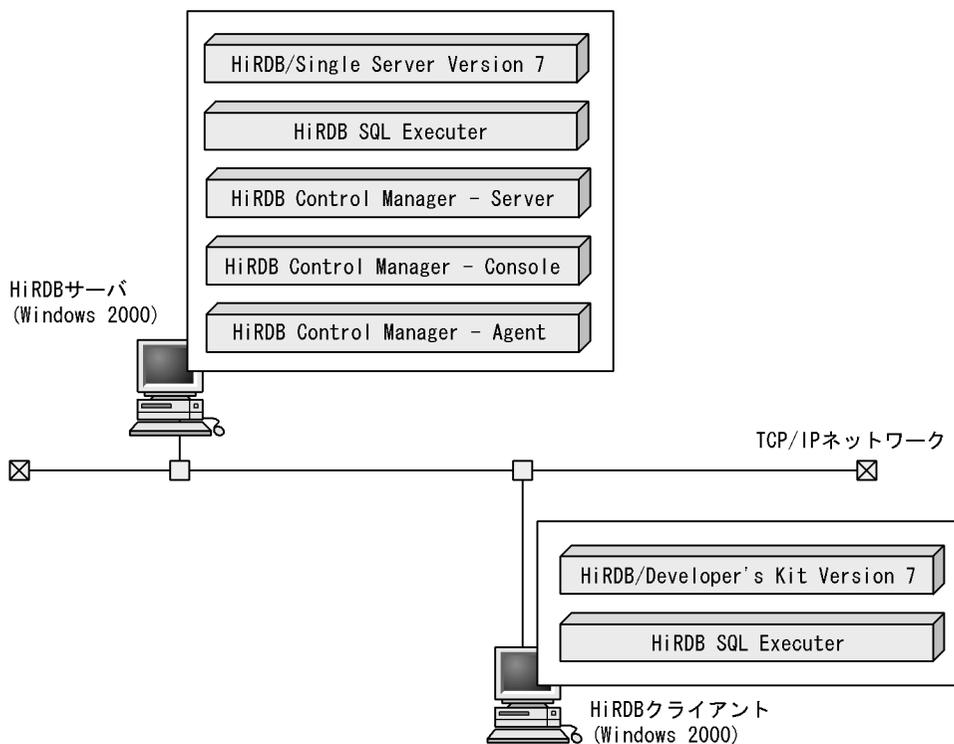


表 1-1 使用するプログラムプロダクト

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	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。		-
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。		
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。	1	-
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。	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。	1	-
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 を作成するときに使用するプログラムプロダクトです。	-	2

(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。

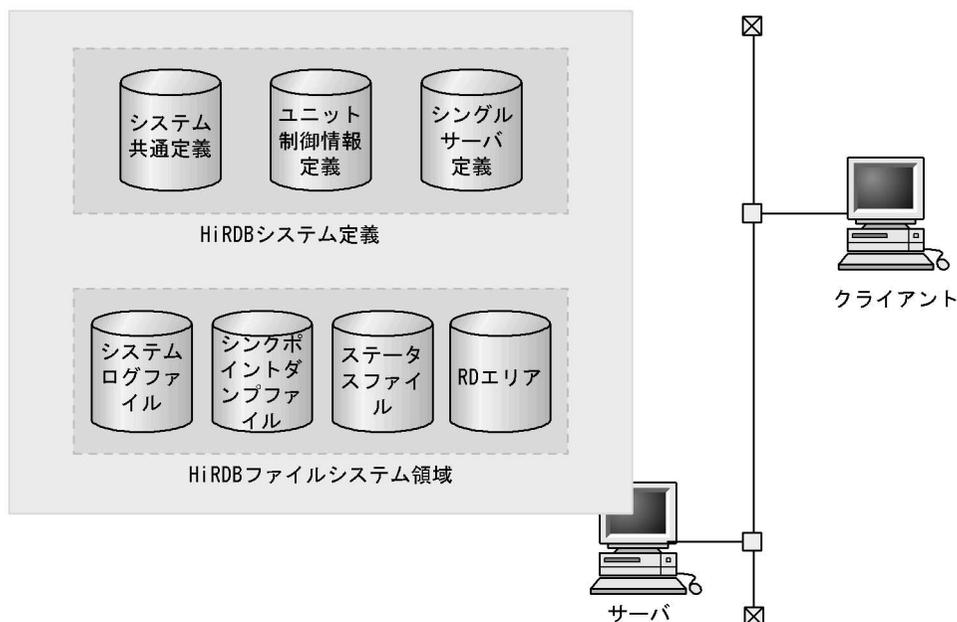
注 2

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。

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定して、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) インストールディレクトリ

このマニュアルでは、使用するプログラムプロダクトを C: ドライブにインストールしています。別のドライブにインストールする場合は、マニュアル中の「C:¥...」をご利用のドライブに読み替えてください。

また、Windows 2000 を C:¥Winnt ディレクトリにインストールしています。別のディレクトリにインストールしている場合は、マニュアル中の「C:¥Winnt¥...」をご利用のディレクトリに読み替えてください。

(2) コマンドの実行

HiRDB のコマンドは、コマンドプロンプトから入力します。

コマンドプロンプトは、[スタート] - [プログラム] - [アクセサリ] - [コマンドプロンプト] で起動してください。

(3) コントロールパネルの開き方

マニュアル中では、[スタート] - [設定] - [コントロールパネル] で [コントロールパネル] を開いて行う作業を [コントロールパネル] と省略して記述しています。

(4) コンピュータの管理の開き方

マニュアル中では、[コントロールパネル] - [管理ツール] - [コンピュータの管理] で [コンピュータの管理] を開いて行う作業を [コンピュータの管理] と省略して記述しています。

(5) イベントビューアの開き方

マニュアル中では、[コントロールパネル] - [管理ツール] - [イベントビューア] で [イベントビューア] を開いて行う作業を [イベントビューア] と省略して記述していません。

(6) OS ごとの操作の差異

このマニュアルでは、Windows 2000 の環境に HiRDB/ シングルサーバを構築するため、Windows 2000 での操作方法を記述しています。Windows 2000 と操作方法が違う OS を使用する場合は、必要に応じて操作部分を読み替えてください。

(7) 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、Windows 2000 のイベントビューアに出力されます。障害が発生したときなどは、この Windows 2000 のイ

イベントビューアを見るようにしてください。

イベントビューアは、[コントロールパネル] - [管理ツール] - [イベントビューア]で見ることができます。「ログ」メニューの「アプリケーション」を選択すると、アプリケーションログの一覧が表示されます。この一覧の「ソース」が「HiRDB/SingleServer」となっているものが、HiRDBが出力したメッセージです。

(8) 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（簡易セットアップツールでの実行）

2.4 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※ 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。

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。

Administrator

使用する OS の管理者 (Administrator) 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 , データベース初期設定ユーティリティ (pdinit) の実行までを行います。

root

HiRDB の管理者ユーザです。データベース初期設定ユーティリティ (pdinit) 実行後に , HiRDB に自動的に作成されます。

このユーザ「root」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

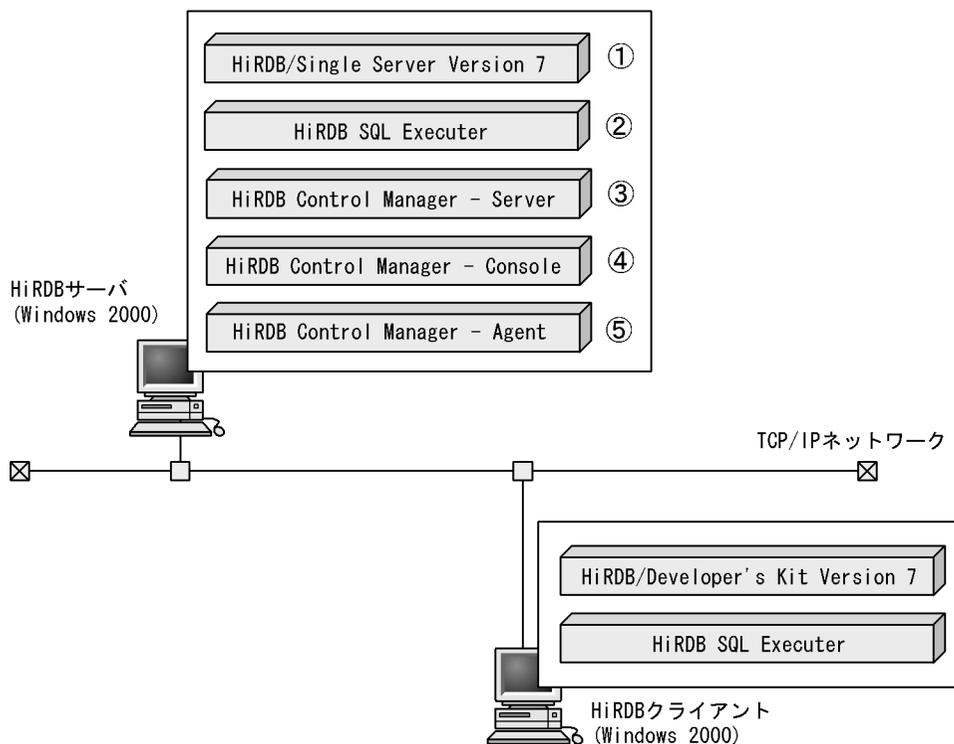
表の作成 (表の定義 , データロード) , および表の操作演習をする HiRDB のユーザです。

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 「2.2.2 HiRDB/シングルサーバをインストールする」で説明します。
- ② 「2.2.3 HiRDB SQL Executerをインストールする」で説明します。
- ③ 「2.2.4 HiRDB Control Managerをインストールする」で説明します。
- ④ 「2.2.4 HiRDB Control Managerをインストールする」で説明します。
- ⑤ 「2.2.4 HiRDB Control Managerをインストールする」で説明します。

なお、インストールは、Windows 2000 ログイン時のユーザ「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。

- 適用機種
- システム環境変数 TZ
- 空きディスク容量
- 仮想メモリ

上記の項目を確認したら、次の項目について設定してください。

(1) 適用機種

HiRDB は、Windows 版の PC/AT 互換機に対応しています。Windows NT、Windows 2000、Windows XP、または Windows Server 2003 がインストールされた PC/AT 互換機であることを確認してください。

(2) システム環境変数 TZ

(a) システム環境変数 TZ の確認方法

システム環境変数の確認方法について説明します。

1. [コントロールパネル] - [システム] を選択してください。
2. [システムのプロパティ] の [詳細] タブから「環境変数」をクリックしてください。
3. システム環境変数の変数「TZ」が JST-9 になっているか確認してください。



変数「TZ」がない場合、または変数「TZ」の値が JST-9 でない場合は、変数「TZ」を追加、または変更する必要があります。

システム環境変数 TZ の値は、システム共通定義の TZ オペランドに指定します。

(b) システム環境変数 TZ の設定方法

システム環境変数に、変数「TZ」、値「JST-9」を設定する例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. [コントロールパネル] - [システム] を選択してください。
2. 【システムのプロパティ】ダイアログの [詳細] タブから「環境変数」をクリックしてください。
3. システム環境変数のいちばん下の行をクリックしてください。



4. 「新規」をクリックしてください。
【新しいシステム変数】ダイアログが表示されます。



5. 変数名に「TZ」、変数値に「JST-9」を入力して、「OK」をクリックしてください。



6. システム環境変数に、変数「TZ」、値「JST-9」が表示されます。「OK」をクリックします。



(3) 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、55MB です。また、このマニュアルで

2. HiRDB システムを構築してみましょう

は、小規模のデータベースを作成して運用します。そのために必要なディスク容量は 500MB です。したがって、合計 555MB 以上空いていることを確認してください。HiRDB Control Manager を使用する場合は、さらに 800MB 程度のディスク容量が必要です。

(4) 仮想メモリ

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、70MB 以上の仮想メモリが必要となります。したがって、HiRDB が起動していない状態で使用している仮想メモリの値 + 70MB の値が、仮想メモリとして必要になります。

仮想メモリの設定方法を次に示します。

1. [コントロールパネル] - [システム] を選択してください。
2. 【システムのプロパティ】ダイアログの [詳細] タブから「パフォーマンスオプション」をクリックしてください。
3. 「仮想メモリ」の「変更」をクリックしてください。
4. 仮想メモリの初期サイズ、および最大サイズに、「HiRDB が起動していない状態で使用している仮想メモリの値 + 70MB」の値を設定してください。
5. 値を設定したら、「設定」をクリックしてから、「OK」をクリックしてください。

注

仮想メモリは、初期サイズと最大サイズを同じ値に設定してください。同じ値に設定しなかった場合、HiRDB がメモリ不足で異常終了することがあります。

また、同時実行ユーザ数が増えると、仮想メモリも増えるので、注意してください。

(5) システムキャッシュの設定

メモリの圧迫を防ぐために、システムキャッシュの設定をします。

1. [マイネットワーク] を選択して「プロパティ」をクリックしてください。
【マイネットワークのプロパティ】ダイアログが表示されます。
2. [ローカルエリア接続] を選択して「プロパティ」をクリックしてください。
【ローカルエリア接続のプロパティ】ダイアログが表示されます。
3. 「Microsoft ネットワーク用ファイルとプリンタ共有」を選択して、「プロパティ」をクリックします。
【Microsoft ネットワーク用ファイルとプリンタ共有のプロパティ】ダイアログが開きます。
4. 「ネットワークアプリケーションのスループットを最大にする」を選択して、「OK」をクリックしてください。

2.2.2 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。

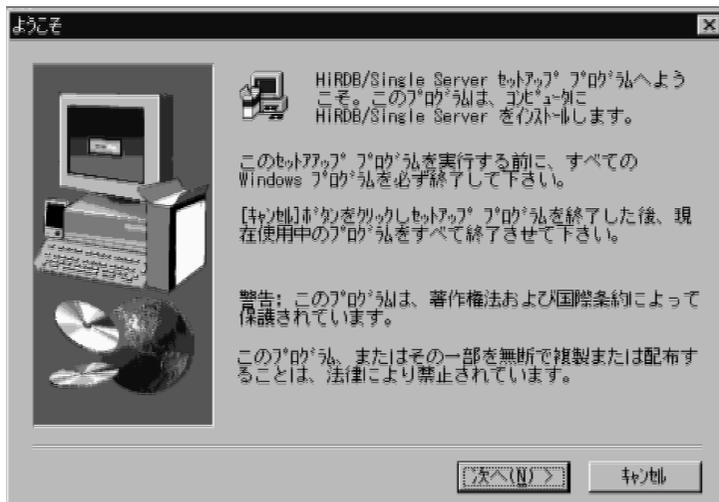
HiRDB/ シングルサーバのインストールは、Administrator 権限があるユーザが実行して

ください。Administrator 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、Windows 2000 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。

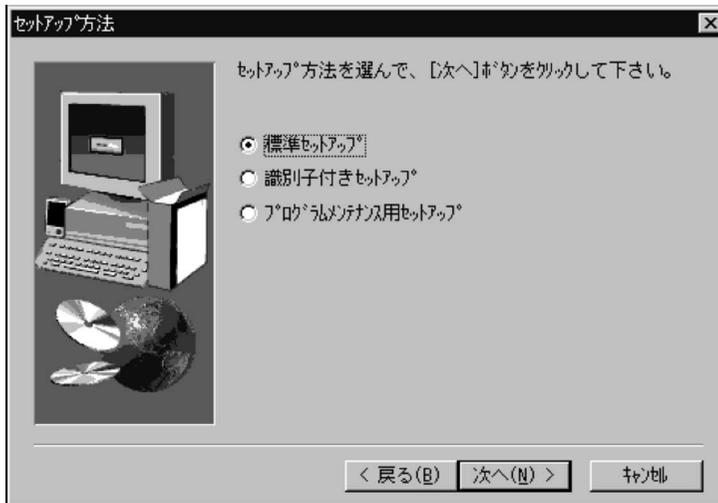
インストール手順

1. インストールする前に、すべての Windows アプリケーションを終了させてください。
2. エクスプローラを使用して、CD-ROM のディレクトリ「01_SRV」の「SETUP.EXE」をダブルクリックしてください。インストールプログラムが起動されるので、画面の指示に従ってインストールしてください。なお、TZ 環境変数に関する警告メッセージが表示されますが、そのまま「OK」をクリックしてください。
3. 「次へ」をクリックしてください。

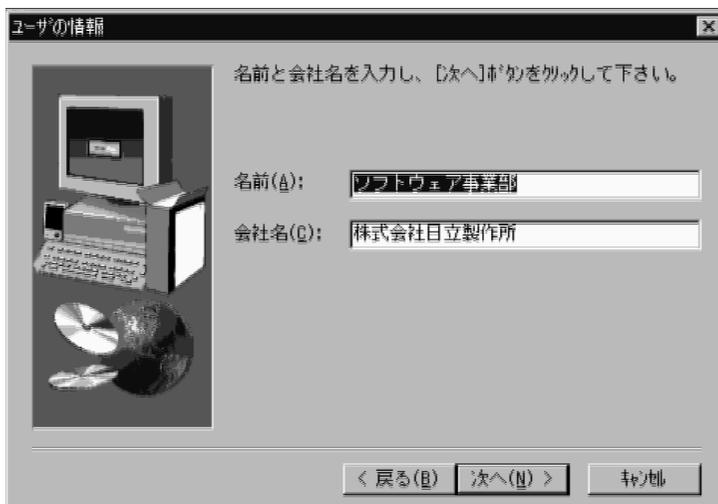


4. 「標準セットアップ」を選択して「次へ」をクリックしてください。
なお、マルチ HiRDB を使用する場合は、「識別子付きセットアップ」を選択します。マルチ HiRDB とは、一つのサーバマシンで複数の HiRDB サーバを稼働させる形態のことです。

2. HIRDB システムを構築してみましょう



5. ユーザの情報として名前と会社名を入力してから、「次へ」をクリックしてください。



6. インストール先ディレクトリ名を指定してください。仮定値は、「ドライブ名 ¥win32app¥hitachi¥hirdb_s」です。仮定値で問題なければ、そのまま「次へ」をクリックしてください。

2. HIRDB システムを構築してみましょう

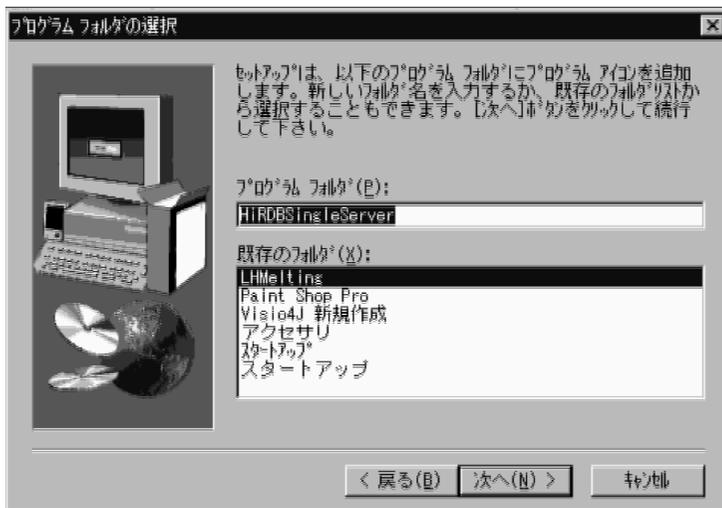


7. インストールするコンポーネントとして「JAVA 実行環境」を選択して、「次へ」をクリックしてください。ただし、「7.5 Java 言語 (SQLJ) でアプリケーションを作成する」の演習を行わない場合は、このコンポーネントは必要ありません。



8. 「次へ」をクリックしてください。

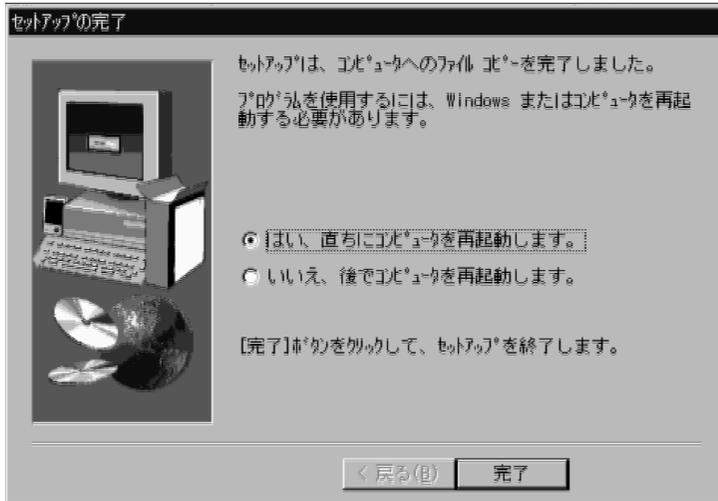
2. HIRDB システムを構築してみましょう



9. 設定した内容を確認し、問題がなければ「次へ」をクリックしてください。インストールが始まります。



10. インストールが終了すると、次の画面が表示されます。



このあとに、HiRDB SQL Executer、および HiRDB Control Manager をインストールするので、「いいえ、後でコンピュータを再起動します。」を選択し、「完了」をクリックしてください。

なお、HiRDB SQL Executer、および HiRDB Control Manager がすでにインストールされている場合には、「はい、直ちにコンピュータを再起動します。」を選択して、「完了」をクリックしてください。

2.2.3 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、Windows 2000 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。

2.2.4 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

2. HiRDB システムを構築してみましょう

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、HiRDB の操作端末として使いたいマシンにインストールします。このマニュアルでは、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 - 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、HiRDB Control Manager のサーバとして使いたいマシンにインストールします。このマニュアルでは、HiRDB Control Manager - Console と同じマシン (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 - 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 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

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。	hirems
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

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

1. CD-ROM を挿入して、「12_CMA¥DISK1」の「SETUP.EXE」をダブルクリックし

2. HIRDB システムを構築してみましょう

てください。

インストーラが起動されます。

- 以降は、画面の内容に従って操作してください。[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。

2.3 HiRDB の環境設定をする（簡易セットアップツールでの実行）

簡易セットアップツールを使って、HiRDB の環境設定を実行します。簡易セットアップツールを使うと、次の作業を実行できます。

1. HiRDB ファイルシステム領域の作成
2. HiRDB システム定義ファイル（pdsys , pdutsys , sds01）の作成
3. システムファイルの作成と初期化
4. データベースの初期化と HiRDB の開始
5. サンプルデータベースの作成

このマニュアルでは、項番 4 までの作業を簡易セットアップツールで実行します。

注意

HiRDB Version 6 で行っていたように、コマンドを使った環境設定やバッチファイル（SPsetup.bat）を使った環境設定も行うことができます。

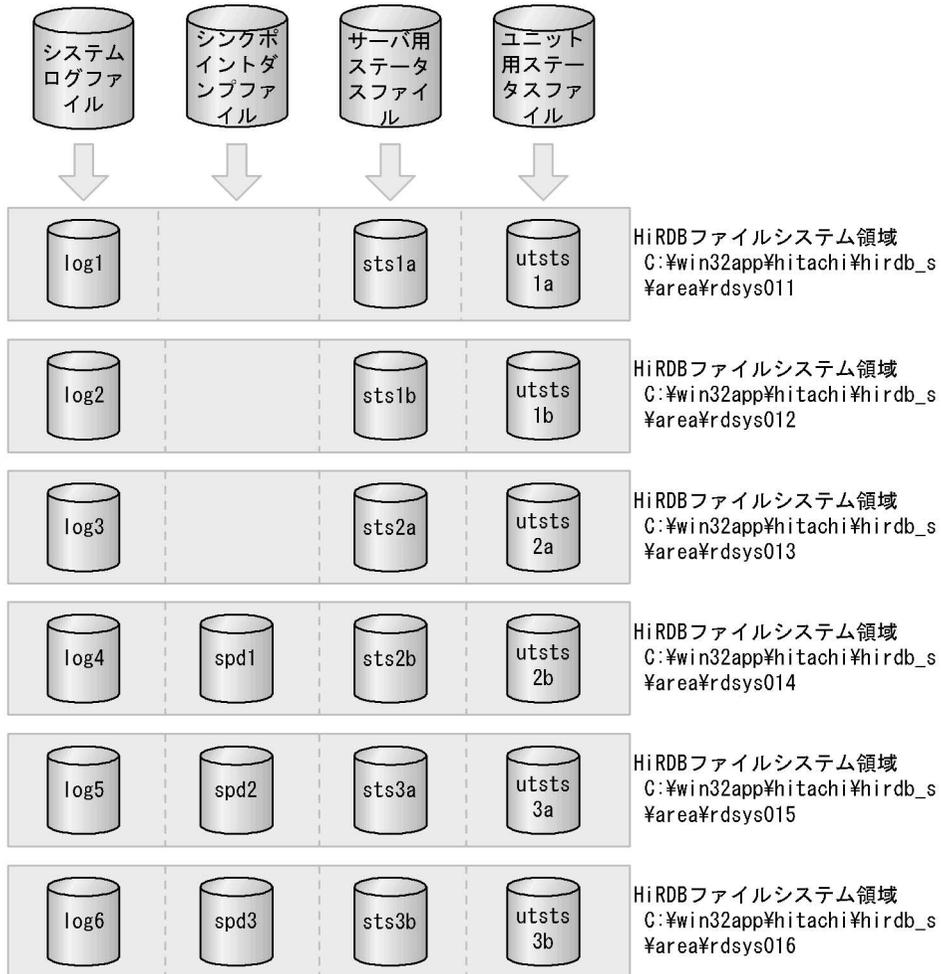
2.3.1 作成する環境

なお、簡易セットアップツールを実行して作成する HiRDB ファイルシステム領域の構成、HiRDB/ シングルサーバの構成、および RD エリアの構成を図 2-3、図 2-4、および図 2-5 に示します。

2. HIRDB システムを構築してみましょう

図 2-3 簡易セットアップツールで作成する HIRDB ファイルシステム領域

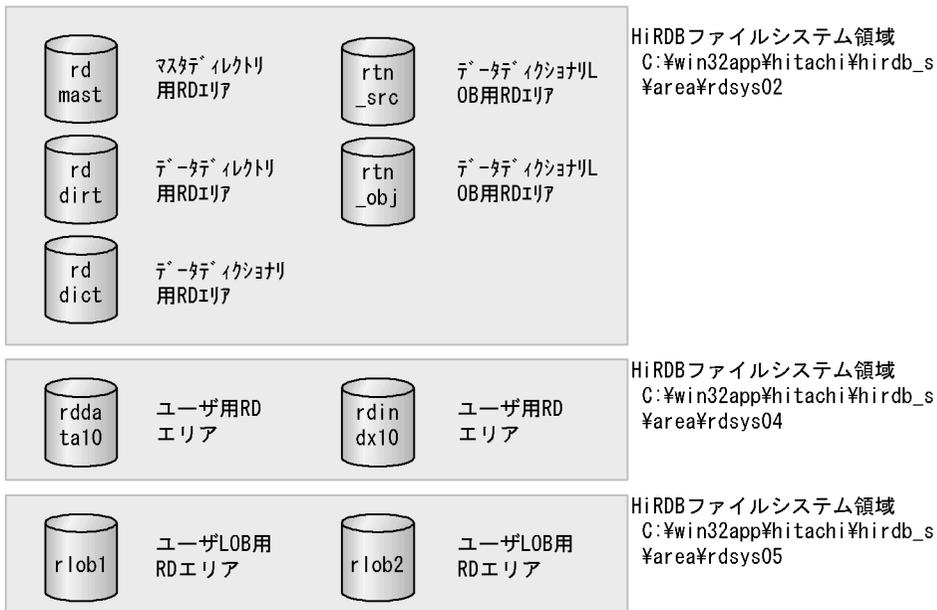
【システムファイル用のHiRDBファイルシステム領域】



【作業表用ファイル用のHiRDBファイルシステム領域】

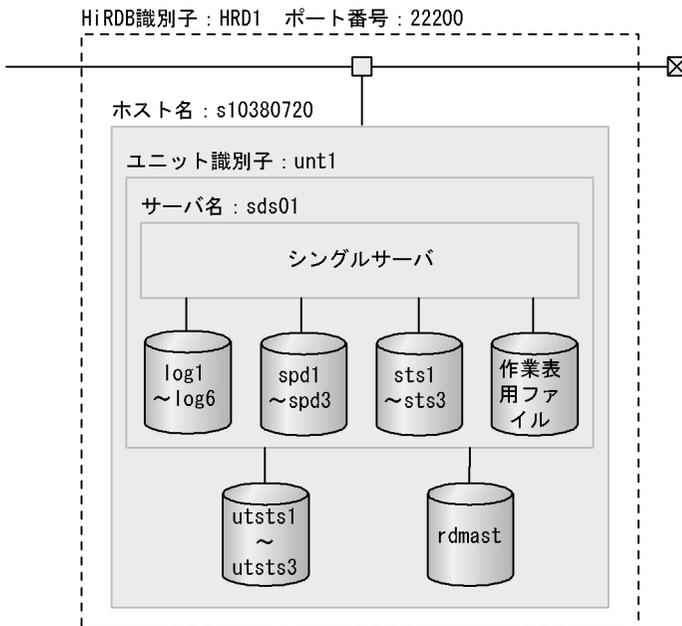


【RDエリア用のHiRDBファイルシステム領域】



2. HiRDB システムを構築してみましょう

図 2-4 簡易セットアップツールで作成する HiRDB/ シングルサーバの構成



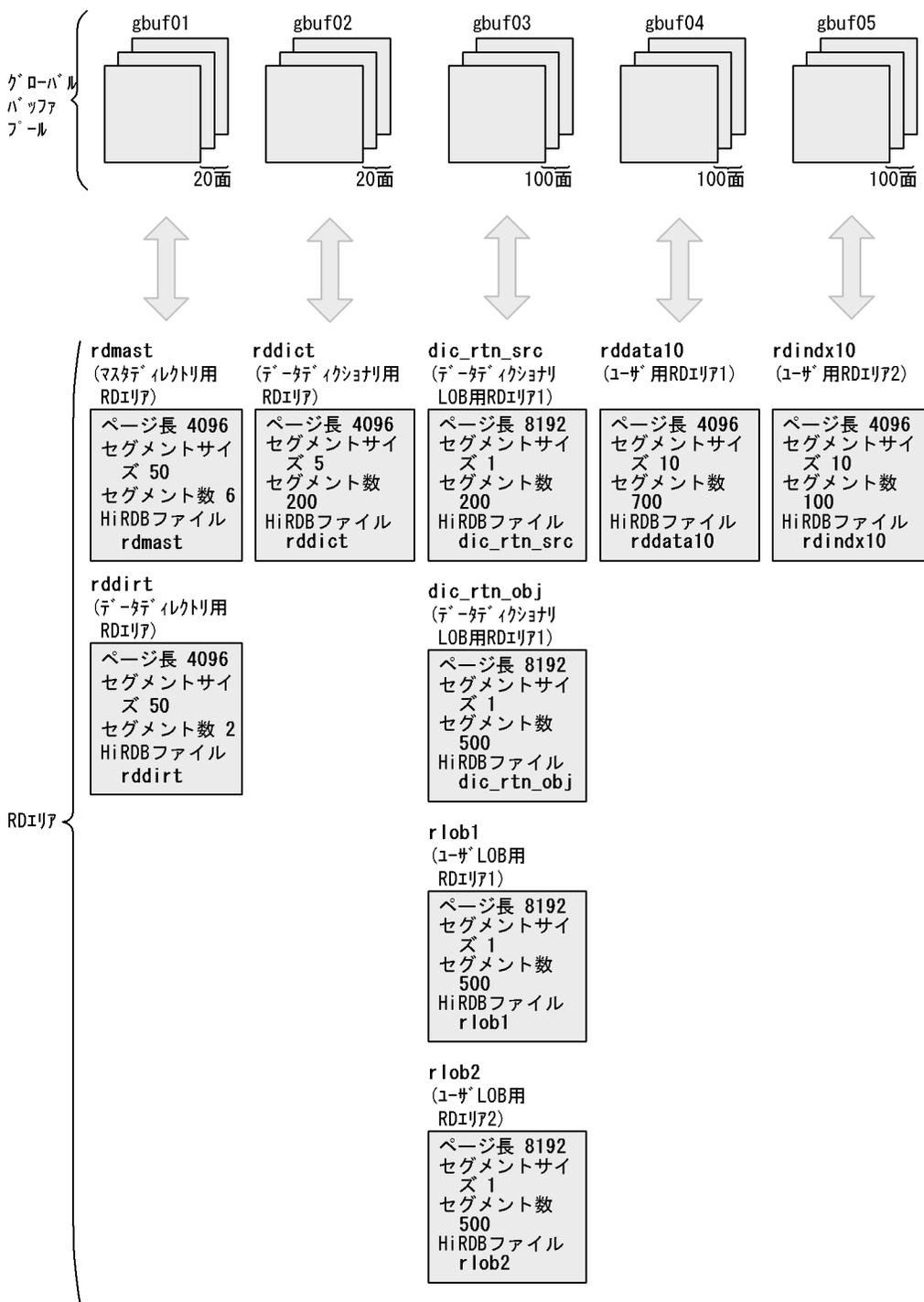
- (凡例) log1~log6 : システムログファイル
spd1~spd3 : シンクポイントダンプファイル
sts1~sts3 : サーバ用ステータスファイル
utsts1~utsts3 : ユニット用ステータスファイル
rdmast : マスタディレクトリ用RDエリアの先頭のHiRDBファイル

[説明]

- HRD1 : システム共通定義pd_system_idオペランドでの指定値
22200 : システム共通定義pd_name_portオペランドでの指定値
s10380720 : システム共通定義pdstartオペランドの-xでの指定値。hostsファイルに設定したホスト名 (HiRDB/Single Serverをインストールしたマシンのホスト名) を指定します。
unt1 : ユニット制御情報定義のpd_unit_idオペランドでの指定値
sds01 : 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。

図 2-5 簡易セットアップツールで作成する RD エリアの構成



2. HiRDB システムを構築してみましょう

システムログファイルの運用方法について

HiRDB には、システムログファイルという HiRDB の操作ログを記録したファイルがあります。このファイルを利用すると、データベースに障害が発生しても、障害が発生する直前の状態まで回復することができます。

システムログに関する運用方法には、大きく分けて表 2-1 に示す二つがあります。

表 2-1 システムログに関する運用方法

システムログファイルの運用方法	説明
システムログをアンロードする運用	データベースの回復にシステムログを利用する運用です。 利点 データベース障害時、障害発生直前の状態まで回復できます。
システムログをアンロードしない運用 (アンロードレスシステムログ運用)	データベースの回復にシステムログを利用しない運用です。 利点 システムログファイルをアンロードする際の CPU 負荷を削減できます。また、アンロードしたシステムログファイル(アンロードログファイル)を格納するためのディスク容量がいらなくなります。

簡易セットアップツールのデフォルト設定では、システムログをアンロードしない運用になります。

システムログをアンロードしない運用の場合、システムログを管理する必要がなくなるので、管理する場合に比べて HiRDB を簡単に運用できます。ただし、この場合、障害が発生したとき、バックアップ取得時点までしかデータベースを回復できません。

障害が発生した場合に障害が発生する直前の状態まで回復したいとき、システムログをアンロードする運用を行います。この際、システムログのアンロードを HiRDB に自動的に実行させる自動ログアンロード機能を使うと、システムログのアンロード処理が HiRDB によって自動的に行われるのでお勧めです。

システムログファイルの詳細については、「4.3 システムログを操作する」を参照してください。

2.3.2 簡易セットアップツールを実行する

簡易セットアップツールの使用手順を次に示します。

(1) 簡易セットアップツールを起動する

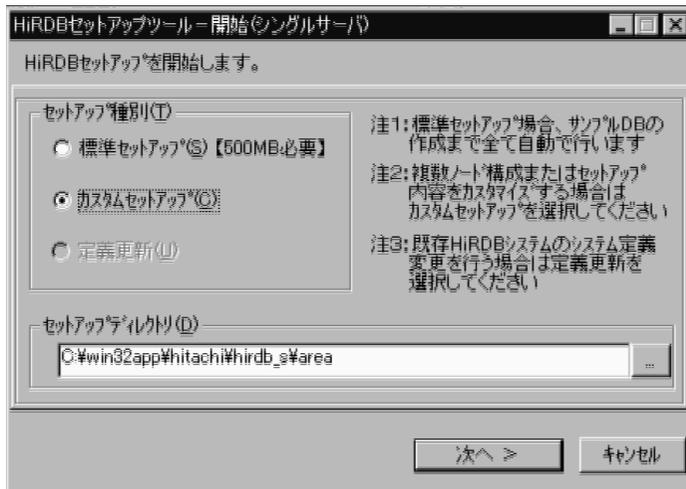
簡易セットアップツールを起動します。

C:\win32app\hitachi\hirdb_s\PDISTUP\bin\pdistup.exe を実行してください。

(2) セットアップ種別を設定する

【HiRDB セットアップツール - 開始 (シングルサーバ)】ウィンドウが表示されます。定義の変更を行うので、次のように設定を変更してから「次へ」ボタンをクリックしてください。

セットアップ種別：カスタムセットアップ



〔説明〕

セットアップ種別

セットアップの種別を指定します。

簡易セットアップツールが用意している標準設定でセットアップする（標準セットアップ）か、標準設定を変更してセットアップする（カスタムセットアップ）か、セットアップ済みの HiRDB の設定を変更する（定義更新）かを選択します。

セットアップディレクトリ

システムファイル、および RD エリアを作成するディレクトリを指定します。

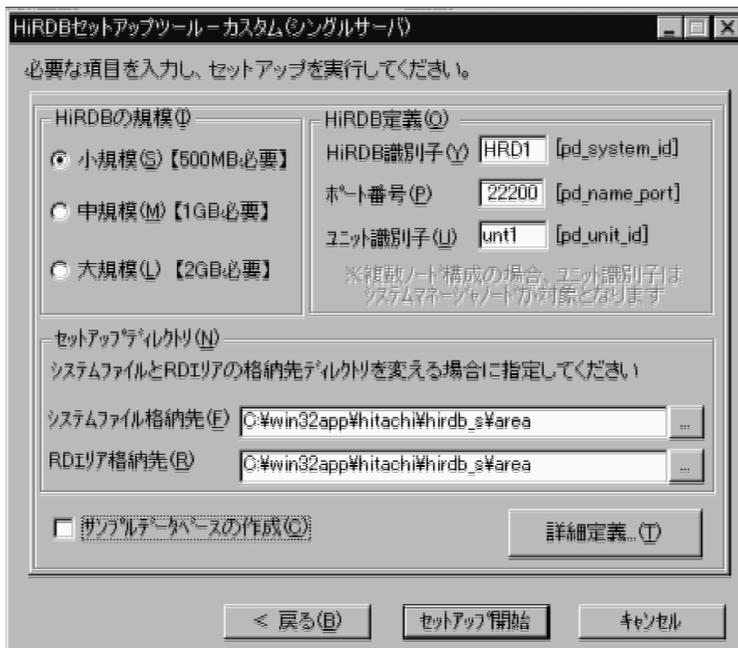
標準値は <インストールディレクトリ> \area です。

(3) HiRDB 規模，HiRDB 定義，セットアップディレクトリを設定する

【HiRDB セットアップツール - カスタム (シングルサーバ)】ウィンドウが表示されます。次のように設定を変更してから「詳細定義」ボタンをクリックしてください。

サンプルデータベースの作成：チェックを外す

2. HiRDB システムを構築してみましょう



[説明]

HiRDB の規模

セットアップするデータベースの規模を指定します。

HiRDB 定義

セットアップする HiRDB の識別子 (4 文字の半角英数字)、使用するポート番号 (5001 ~ 65535)、およびユニット識別子 (4 文字の半角英数字) を指定します。

セットアップディレクトリ

システムファイル、および RD エリアを作成するディレクトリを指定します。
標準値は <インストールディレクトリ> \%area です。

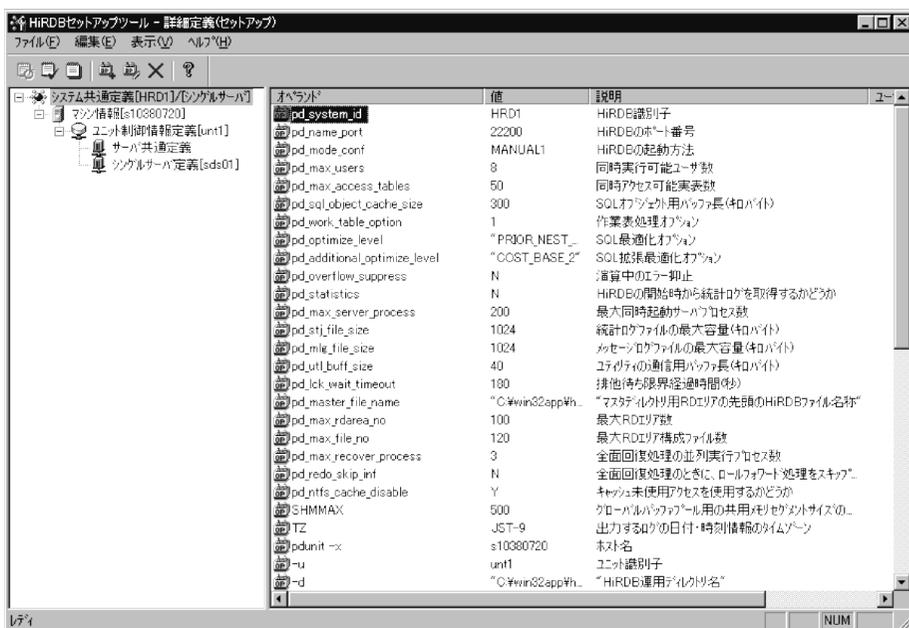
HiRDB の規模を選択するときの注意事項：

大規模、中規模、小規模を選択すると、それぞれ pd_max_users (同時実行可能ユーザ数) に 32, 16, 8 が設定されたシステム共通定義が作成されます。ご購入されているライセンスの形態に合わせて選択してください。

(4) 定義ファイルを確認する

【HiRDB セットアップツール - 詳細定義 (セットアップ)】ウィンドウが表示されます。ここで、「2.3.1 作成する環境」で説明したシステムログファイルの運用に関する設定を行います。左側のペインから「シングルサーバ定義 [sds01]」を選択してください。

2. HIRDB システムを構築してみましょう



〔説明〕

システム共通定義 [HRD1] / [シングルサーバ]

選択すると、右側のペインにシステム共通定義 (pdsys) の内容が表示されます。

マシン情報 [s10380720]

選択すると、右側のペインにホスト名などのマシン情報が表示されます。

ユニット制御情報定義 [unit1]

選択すると、右側のペインにユニット制御情報定義 (pdutys) の内容が表示されます。

サーバ共通定義

HiRDB/ シングルサーバでは使用しません。

シングルサーバ定義 [sds01]

選択すると、右側のペインにシングルサーバ定義 (sds01) の内容が表示されます。

このマニュアルでの設定値

このマニュアルで使用する各定義ファイルの設定値を次に示します。これは、簡易セットアップツールのデフォルトの値をそのまま使用したものです。

システム共通定義 (pdsys)

2. HIRDB システムを構築してみましょう

```
*****
**
# pdsys : HiRDB system common definition (SingleServer:small)
*****
**
#
#-----
--
# set form
#
set pd_system_id = HRD1
set pd_name_port = 22200
set pd_mode_conf = MANUAL1
set pd_max_users = 8
set pd_max_access_tables = 50
set pd_sql_object_cache_size = 300
set pd_work_table_option = 1
set pd_optimize_level = ¥
"PRIOR_NEST_JOIN",¥
"PRIOR_OR_INDEXES",¥
"DETER_AND_INDEXES",¥
"RAPID_GROUPING",¥
"DETER_WORK_TABLE_FOR_UPDATE"
set pd_additional_optimize_level = "COST_BASE_2"
set pd_overflow_suppress = N
set pd_statistics = N

set pd_max_server_process = 200
set pd_stj_file_size = 1024
set pd_mlg_file_size = 1024
set pd_utl_buff_size = 40
set pd_lck_wait_timeout = 180
set pd_master_file_name = "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast"
set pd_max_rdarea_no = 100
set pd_max_file_no = 120
set pd_max_recover_process = 3
set pd_redo_skip_inf = N
set pd_ntfs_cache_disable = Y
#
#-----
--
# putenv form
#
putenv SHMMAX 500
putenv TZ JST-9
#
#-----
--
# command form
#
pdunit -x s10380720 -u unt1 -d "C:¥win32app¥hitachi¥hirdb_s"
pdstart -t SDS -s sds01 -u unt1
pdbuffer -a gbuf01 -n 20 -r rdmast,rddirt -w 20
pdbuffer -a gbuf02 -n 20 -r rddict -w 20
pdbuffer -a gbuf03 -n 10000 -o -w 20
pdbuffer -a gbuf04 -n 10000 -r rddata10 -w 20
pdbuffer -a gbuf05 -n 10000 -r rdindx10 -w 20
```

ユニット制御情報定義 (pdutysys)

```

*****
**
# pdutysys : HiRDB unit control information definition (SingleServer:small)
#*****
**
#
#-----
--
# set form
#
set pd_unit_id = unt1
#set pd_shmpool_attribute = free
set pd_lck_deadlock_info = Y
set pd_lck_wait_timeout = 180
set pd_max_server_process = 100
set pd_stj_file_size = 1024
set pd_max_recover_process = 3
set pd_redo_skip_inf = N

set pd_syssts_file_name_1 = ¥
"utsts1",¥
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys011¥utsts1a",¥
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys012¥utsts1b"
set pd_syssts_file_name_2 = ¥
"utsts2",¥
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys013¥utsts2a",¥
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys014¥utsts2b"
set pd_syssts_file_name_3 = ¥
"utsts3",¥
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys015¥utsts3a",¥
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys016¥utsts3b"
set pd_syssts_initial_error = stop
set pd_syssts_singleoperation = continue
#
#-----
--
# putenv form
#
putenv SHMMAX 500

```

シングルサーバ定義 (sds01)

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、運用方法が単純な「アンロード状態をチェックしない運用」になっています (pd_log_unload_check=N)。しかし、障害対策などにきめ細かい対応をしたい場合は、システムログをアンロードする運用が必要です。そのような場合は、「自動ログアンロード機能を使った運用」になるよう設定を変更してください。

この場合に変更が必要なオペランドは、次の二つです。

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

2. HIRDB システムを構築してみましょう

```
*****
**
# HiRDB single server definition (SingleServer:small)
# --- Server Name : sds01 ---
*****
**
#
#-----
--
# set form
#
set pd_table_def_cache_size = 200
set pd_auth_cache_size = 1
set pd_view_def_cache_size = 1
set pd_process_count = 8
set pd_sds_shmpool_size = 3150
set pd_lck_pool_size = 1000
set pd_work_buff_mode = pool
set pd_work_buff_size = 384
set pd_log_rec_leng = 1024
set pd_log_max_data_size = 32000
set pd_log_write_buff_count = 3
set pd_log_dual = N
set pd_log_rerun_swap = N
set pd_log_unload_check = N
set pd_log_rerun_reserved_file_open = Y
set pd_log_sdinterval = 1000
set pd_spd_assurance_count = 2
set pd_spd_max_data_size = 32000
set pd_spd_reduced_mode = 2
set pd_spd_reserved_file_auto_open = N
set pd_sts_initial_error = stop
set pd_sts_singleoperation = stop
set pd_log_auto_unload_path = "C:\win32app\hitachi\hirdb_s\area\unloadlog"

set pd_sts_file_name_1 = ¥
"sts1", ¥
"C:\win32app\hitachi\hirdb_s\area\rdsys011\sts1a", ¥
"C:\win32app\hitachi\hirdb_s\area\rdsys012\sts1b"
set pd_sts_file_name_2 = ¥
"sts2", ¥
"C:\win32app\hitachi\hirdb_s\area\rdsys013\sts2a", ¥
"C:\win32app\hitachi\hirdb_s\area\rdsys014\sts2b"
set pd_sts_file_name_3 = ¥
"sts3", ¥
"C:\win32app\hitachi\hirdb_s\area\rdsys015\sts3a", ¥
"C:\win32app\hitachi\hirdb_s\area\rdsys016\sts3b"
#
```

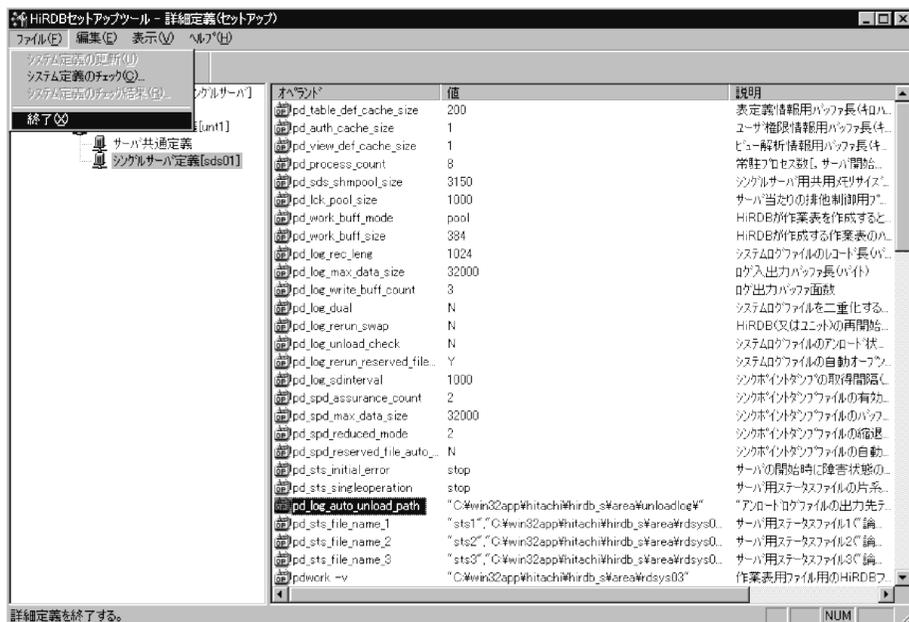
```

#-----
--
# command form
#
pdwork -v "C:\win32app\hitachi\hirdb_s\area\rdsys03"
pdlogadfg -d sys -g log1 ONL
pdlogadfg -d sys -g log2 ONL
pdlogadfg -d sys -g log3 ONL
pdlogadfg -d sys -g log4 ONL
pdlogadfg -d sys -g log5 ONL
pdlogadfg -d sys -g log6 ONL
pdlogadpf -d sys -g log1 -a "C:\win32app\hitachi\hirdb_s\area\rdsys011\log1"
pdlogadpf -d sys -g log2 -a "C:\win32app\hitachi\hirdb_s\area\rdsys012\log2"
pdlogadpf -d sys -g log3 -a "C:\win32app\hitachi\hirdb_s\area\rdsys013\log3"
pdlogadpf -d sys -g log4 -a "C:\win32app\hitachi\hirdb_s\area\rdsys014\log4"
pdlogadpf -d sys -g log5 -a "C:\win32app\hitachi\hirdb_s\area\rdsys015\log5"
pdlogadpf -d sys -g log6 -a "C:\win32app\hitachi\hirdb_s\area\rdsys016\log6"
pdlogadfg -d spd -g spd1 ONL
pdlogadfg -d spd -g spd2 ONL
pdlogadfg -d spd -g spd3 ONL
pdlogadpf -d spd -g spd1 -a "C:\win32app\hitachi\hirdb_s\area\rdsys014\spd1"
pdlogadpf -d spd -g spd2 -a "C:\win32app\hitachi\hirdb_s\area\rdsys015\spd2"
pdlogadpf -d spd -g spd3 -a "C:\win32app\hitachi\hirdb_s\area\rdsys016\spd3"

```

(5) 定義ファイルの確認を終了する

定義ファイルの変更を終了します。[ファイル] - [終了] を選択してください。



(6) セットアップを開始する - その1

【HiRDB セットアップツール - カスタム (シングルサーバ)】ウィンドウが表示されま

2. HIRDB システムを構築してみましょう

す。「セットアップ開始」ボタンをクリックしてください。



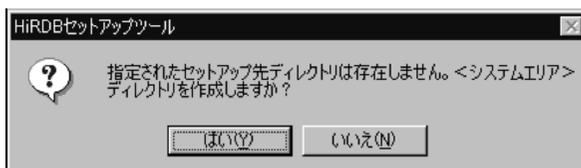
(7) セットアップを開始する - その2

セットアップの開始を確認するダイアログが表示されるので、「OK」ボタンをクリックしてください。



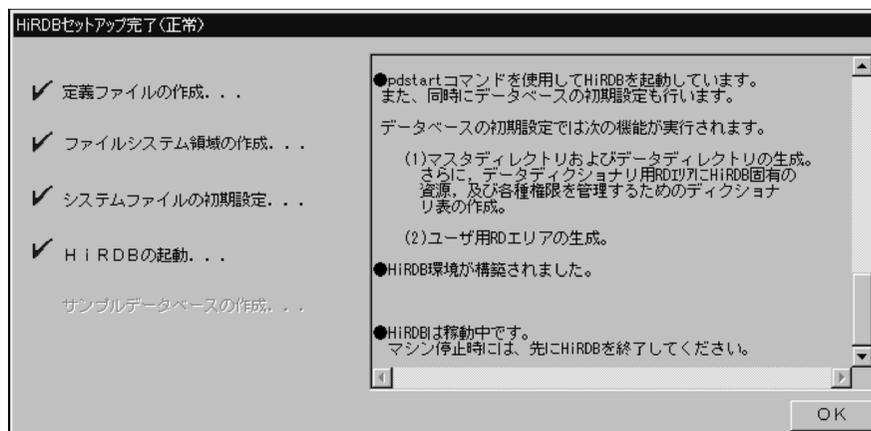
(8) セットアップを開始する - その3

デフォルトで設定されている <システムエリア> ディレクトリが存在していないため、新規作成するかどうかを確認するダイアログが表示されます。「はい」ボタンをクリックしてください。



(9) セットアップを終了する

セットアップが完了すると、【HiRDB セットアップ完了 (正常)】ウィンドウが表示されます。セットアップを終了するため、「OK」ボタンをクリックしてください。



(10) HiRDB の稼働状況を確認する

実際に HiRDB が稼働しているかどうかを、pdls コマンドで確認します。

【コマンド形式】

```
C:¥>pdls
```

【コマンドの実行結果】

HOSTNAME (104901)	UNITID	SVID	STATUS	STARTTIME
s10380720	unt1	*****	ACTIVE	104807
s10380720	unt1	sds01	ACTIVE	104807

〔説明〕

STATUS が ACTIVE になっていれば、HiRDB が稼働中です。

2.3.3 定義ファイルを変更したい場合は

簡易セットアップツールを使ったセットアップで、定義ファイルを変更する手順について説明します。以降の演習では、定義ファイルを変更しないでセットアップした場合を仮定して話を進めますので、読みながら演習を実施したい方は、定義ファイルを変更しないでください。

2. HIRDB システムを構築してみましょう

ここでは、「2.3.1 作成する環境」で説明した自動ログアンロード機能を使えるようにするための定義変更を例に説明します。

自動ログアンロード機能を使えるようにするためには、シングルサーバ定義に次の設定が必要です。

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

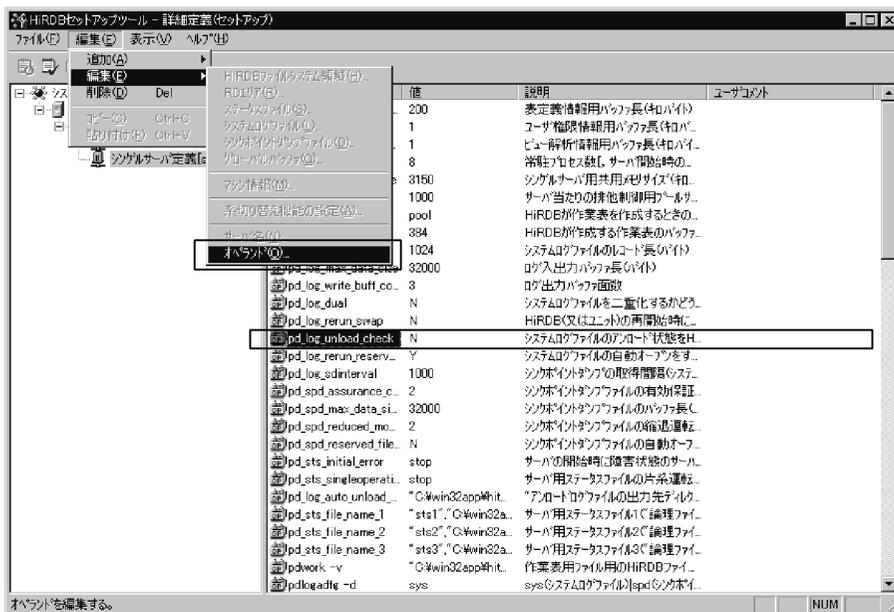
「pd_log_auto_unload_path」にはデフォルトの値が設定されているのでそのまま利用することにして、「pd_log_unload_check」の設定を変えます。設定を変更する手順を次に示します。

(1) シングルサーバ定義を表示する

(4) までと同じ操作を実施してください。

(2) シングルサーバ定義を変更する - その 1

シングルサーバ定義の内容が表示されます。「pd_log_unload_check」を選択してから、[編集] - [編集] - [オペランド] を選択してください。

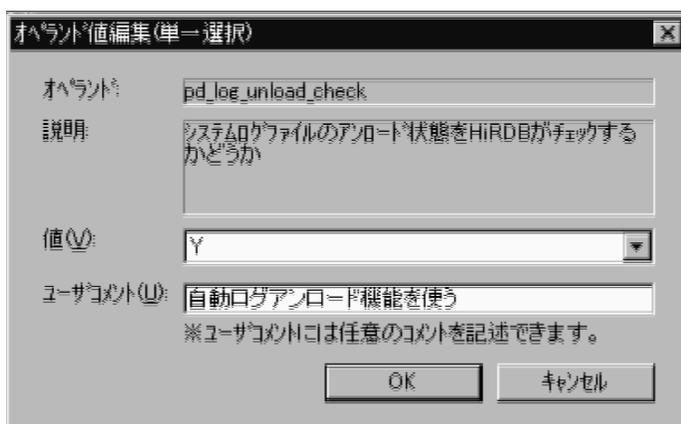


(3) シングルサーバ定義を変更する - その 2

【オペランド値編集(単一選択)】の内容が表示されます。値を次のように指定して、「OK」ボタンをクリックしてください。

値 : Y

2. HIRDB システムを構築してみましょう



(4) 定義が正しく変更されているか確認する

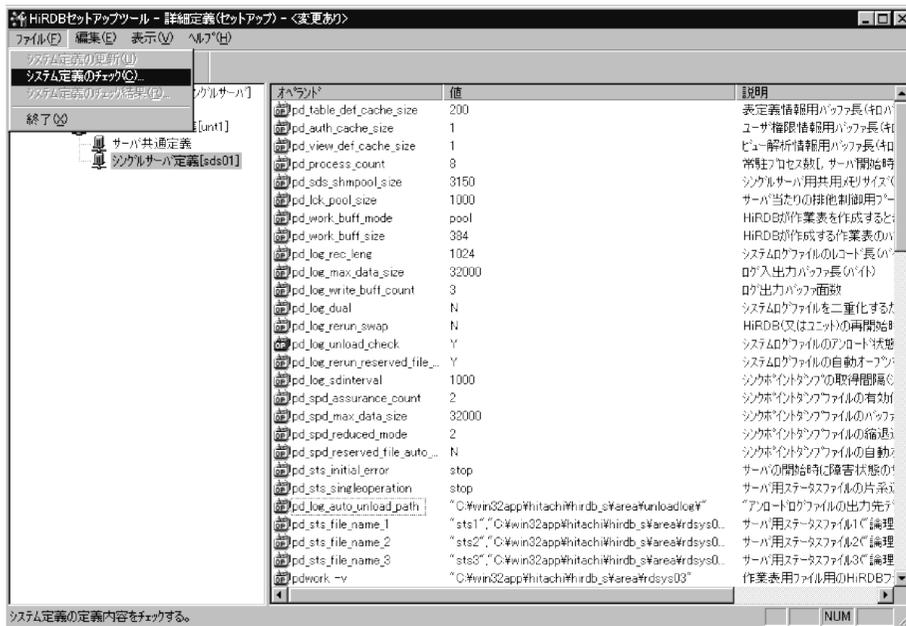
シングルサーバ定義の内容が表示されます。「pd_log_unload_check」の値が「Y」になっていることを確認してください。



(5) システム定義をチェックする

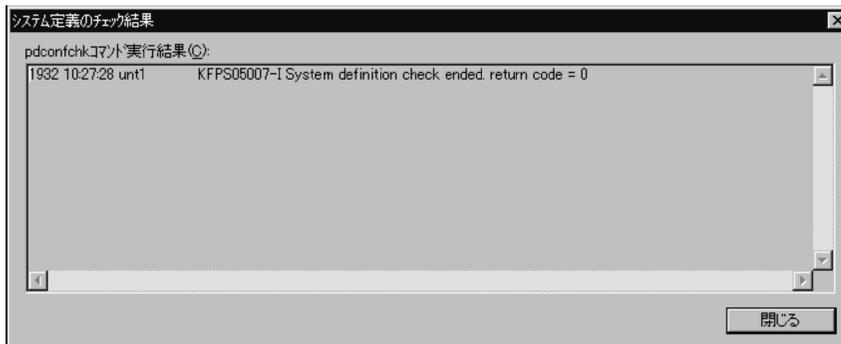
変更した結果、システム定義に矛盾が発生していないか確認します。[ファイル] - [システム定義のチェック]を選択してください。

2. HIRDB システムを構築してみましょう



(6) システム定義チェックを確認する

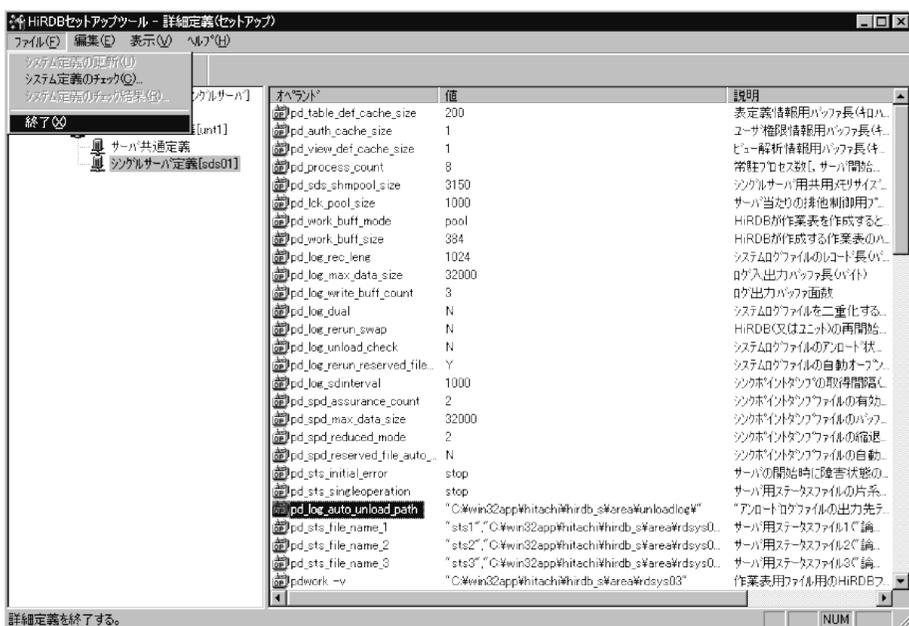
システム定義チェックの結果が表示されます。エラーが出力されていないことを確認して、「閉じる」ボタンをクリックしてください。エラーが出力された場合は、メッセージを基にシステム定義を見直してください。



(7) 定義ファイルの変更を終了する

定義ファイルの変更を終了します。[ファイル] - [終了] を選択してください。

2. HIRDB システムを構築してみましょう



2.4 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 GOODS 表、STOCK 表にデータロードをします。

図 2-6 定義する表

GOODS表

CHAR (6)	CHAR (30)	DECIMAL (7, 0)	CHAR (5)
ショウヒンCD	ショウヒンメイ	タンカ	シレサキCD

↑
インデクス
GOODSX

STOCK表

CHAR (6)	DECIMAL (7, 0)	DECIMAL (7, 0)	DECIMAL (7, 0)	CHAR (5)
ショウヒンCD	ザイコリヨウ	ヒキアテリヨウ	ハツチュウテン	シレサキCD

↑
インデクス
STOCKIDX

2.4.1 ユーザ名を登録する

セットアップツール (PDISTUP.EXE) 実行直後、HiRDB を利用できるユーザは「root」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「root」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.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。

```
PDHOST=s10380720
PDNAMEPORT=22200
PDUSER=' "root"/"root"'
```

〔説明〕

PDHOST=s10380720

...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=22200

...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="root"/"root"

...HiRDB に接続するときのユーザ名, パスワード

2. HiRDB SQL Executer を起動します。[スタート] - [プログラム] - [HiRDB SQL Executer] - [GUI 版 HiRDB SQL Executer] を選択してください。
3. 次の SQL 文を入力し, 実行してください。なお, SQL 文を実行する場合は, SQL 文を入力した後に「F5」キーまたは「Enter」キーを押します。

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

4. 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.
```

5. 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, 項番 1 で設定した HiRDB.ini ファイルの PDUSER を, 次のように変更します。

```
PDUSER=MANAGER/MANAGER
```

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。

2.4.2 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 (Windows(R) 用)」を参照してください。

(1) 「C:¥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 C:¥win32app¥hitachi¥hirdb_s¥SAMPLE¥sampleDB¥tblcreate を変更して使用しています。

【C:¥hirdb¥pddef¥CreateT.txt の内容】

```

CREATE SCHEMA; .....1
CREATE FIX TABLE GOODS
(
  ショウヒンCD    CHAR(6) ,
  ショウヒンメイ  CHAR(30),
  タカ            DECIMAL(7, 0),
  シレサキCD     CHAR(5))
IN RDDATA10 PCTFREE=(10, 0)
UNIQUE CLUSTER KEY
(
  ショウヒンCD    ASC)
IN RDINDX10 PCTFREE=10; .....2
CREATE INDEX GOODSX ON GOODS
(
  ショウヒンメイ  ASC)
IN RDINDX10 PCTFREE=10; .....3
CREATE FIX TABLE STOCK
(
  ショウヒンCD    CHAR(6),
  サイコロヨウ   DECIMAL(7, 0),
  ヒキアテリヨウ DECIMAL(7, 0),
  ハツチュウテン DECIMAL(7, 0),
  シレサキCD     CHAR(5))
IN RDDATA10 PCTFREE=(10, 0)
CLUSTER KEY
(
  ショウヒンCD    ASC)
IN RDINDX10 PCTFREE=10; .....4

```

〔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. GOODS 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「ショウヒン_CD」をクラスタキー（昇順）にします。
3. GOODS 表の列「ショウヒンメイ」にインデクス GOODSX 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. STOCK 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「ショウヒン_CD」をクラスタキー（昇順）にします。

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】

```
C:¥>pddef < C:¥hirdb¥pddef¥CreateT.txt
```

【説明】

「C:¥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
```

(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。

「2.4.1 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「root」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。[ファイル] - [切り離し]で HiRDB サーバと切り離れたあと、[ファイル] - [接続]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	GOODS	BASE TABLE	4
MANAGER	STOCK	BASE TABLE	5
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	GOODS	(CLUSTER0000131195)
MANAGER	GOODS	GOODSX
MANAGER	STOCK	(CLUSTER0000131196)
KFPX27010-I		3 rows selected

2.4.3 表ヘデータロードをする

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。

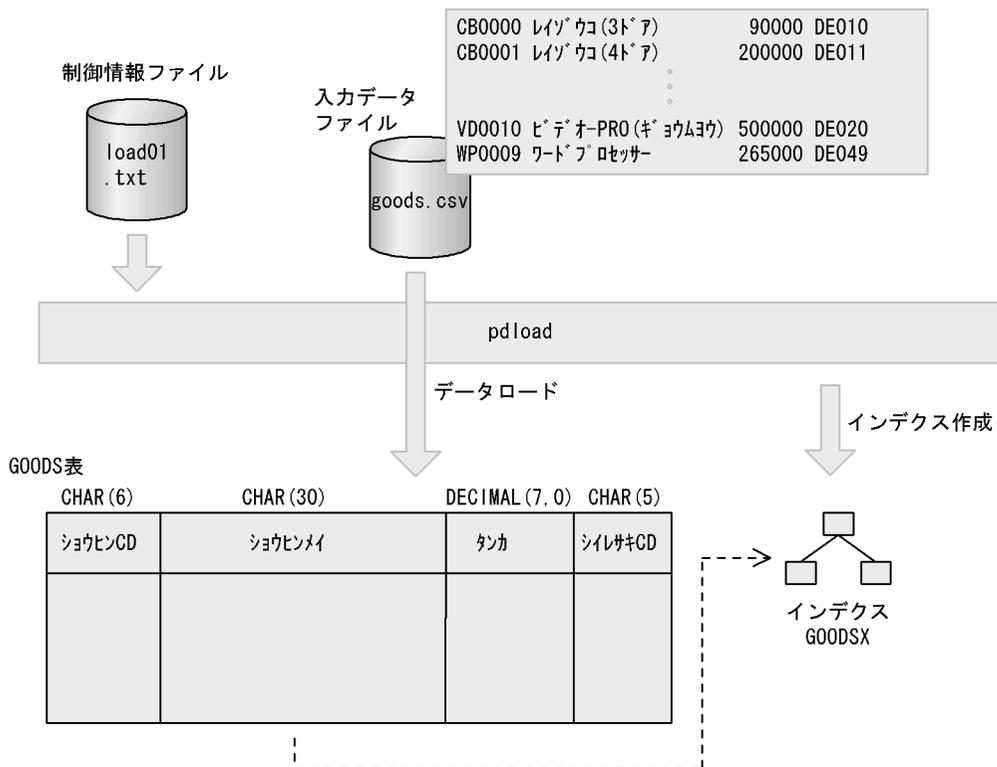
なお、この項では、データロードに関する基本的なことだけ説明しています。詳細について知りたい場合は、マニュアル「HiRDB Version 7 システム導入・設計ガイド (Windows(R) 用)」を参照してください。

(1) GOODS 表へのデータロード

GOODS 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 GOODS 表へのデータロードの概要



(a) 「C:\%hirdb%\pdload\load01.txt」の作成

まず、制御情報ファイル「C:\%hirdb%\pdload\load01.txt」を作成します。

【C:\%hirdb%\pdload\load01.txt の内容】

```
source C:\%win32app%\hitachi%\hirdb_s%\sample%\sampleDB%\loaddata\Goods.csv
```

〔説明〕

C:\%win32app%\hitachi%\hirdb_s%\sample%\sampleDB%\loaddata\Goods.csv :
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】

```
C:\%>pdload -u MANAGER -n 10 -i c GOODS C:\%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。

GOODS :

データロードする表の名称です。

C:¥hirdb¥pdload¥load01.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。

2. HiRDB システムを構築してみましょう

```
2380 13:40:29 unt1 sds01    KFPL00709-I Error information
file was created, file=C:/tmp/ERROR-s2ac.
2380 13:40:29 unt1 sds01    KFPL00702-I Pdlod started,
table=MANAGER."GOODS"
2380 13:40:30 unt1 sds01    KFPL00710-I Index information
file assigned, index=MANAGER."(CLUSTER0000131195)",
RDAREA="RDINDX10",
file=C:/tmp/INDEX-(CLUSTER0000131195)-RDINDX10-s2ac.1
2380 13:40:30 unt1 sds01    KFPL00710-I Index information
file assigned, index=MANAGER."GOODSX",
RDAREA="RDINDX10", file=C:/tmp/INDEX-GOODSX-RDINDX10-s2ac.2
2380 13:40:30 unt1 sds01    KFPL00723-I 100 rows loaded,
table=MANAGER."GOODS", RDAREA="RDATA10"
1572 13:40:30 unt1 0mload0 KFPL00703-I 100 rows loaded
2476 13:40:30 unt1 sds01    KFPL00715-I Index load started
at sds01, index=MANAGER."(CLUSTER0000131195)",
RDAREA="RDINDX10"
2476 13:40:30 unt1 sds01    KFPL00716-I Index load ended
at sds01, index=MANAGER."(CLUSTER0000131195)",
RDAREA="RDINDX10", return code=0
2476 13:40:30 unt1 sds01    KFPL00720-I Index file deleted,
server=sds01,
file=C:/tmp/INDEX-(CLUSTER0000131195)-RDINDX10-s2ac.1
2276 13:40:30 unt1 sds01    KFPL00715-I Index load started
at sds01, index=MANAGER."GOODSX",
RDAREA="RDINDX10"
2276 13:40:31 unt1 sds01    KFPL00716-I Index load ended
at sds01, index=MANAGER."GOODSX",
RDAREA="RDINDX10", return code=0
2276 13:40:31 unt1 sds01    KFPL00720-I Index file deleted,
server=sds01, file=C:/tmp/INDEX-GOODSX-RDINDX10-s2ac.2
1948 13:40:31 unt1 _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 GOODS 表を検索します。

【入力 SQL】

```
SELECT * FROM GOODS;
```

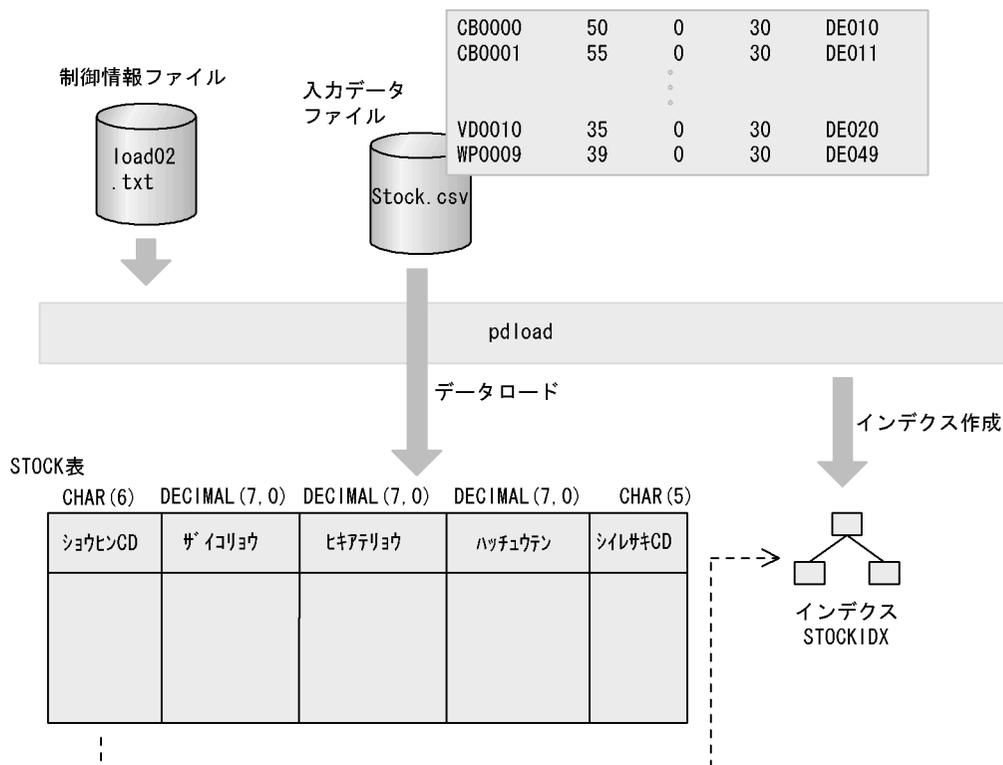
【SQL の実行結果】

ショウヒンCD	ショウヒンメイ	タンカ	シレサキCD
CB0000	レイゾウウコ(3トリア)	0090000.	DE010
CB0001	レイゾウウコ(4トリア)	0200000.	DE011
CB0002	レイゾウウコ(セヒョウキノウツキ)	0250000.	DE012
:			
VD0009	ヒダテオヘヘータ	0140000.	DE019
VD0010	ヒダテオPRO(キョウムヨウ)	0500000.	DE020
WP0009	ワートブプロセッサ	0265000.	DE049
KFPX27010-I		100 rows selected	

(2) STOCK 表へのデータロード

STOCK 表へのデータロードの概要を図 2-8 に示します。

図 2-8 STOCK 表へのデータロードの概要



2. HIRDB システムを構築してみましょう

(a) 「C:¥hirdb¥pload¥load02.txt」の作成

まず、制御情報ファイル「C:¥hirdb¥pload¥load02.txt」を作成します。

【C:¥hirdb¥pload¥load02.txt の内容】

```
source C:¥win32app¥hitachi¥hirdb_s¥sample¥sampleDB¥loaddata¥Stock.csv
```

〔説明〕

C:¥win32app¥hitachi¥hirdb_s¥sample¥sampleDB¥loaddata¥Stock.csv :
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】

```
C:¥>pdload -u MANAGER -n 10 -i c STOCK C:¥hirdb¥pload¥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。

STOCK :
データロードする表の名称です。

C:¥hirdb¥pload¥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。

```

2512 13:47:42 unt1 sds01    KFPL00709-I Error information
file was created, file=C:/tmp/ERROR-s2eg.
2512 13:47:42 unt1 sds01    KFPL00702-I Pdload started,
table=MANAGER."STOCK"
2512 13:47:42 unt1 sds01    KFPL00710-I Index information
file assigned, index=MANAGER."(CLUSTER0000131196)",
RDAREA="RDINDX10",
file=C:/tmp/INDEX-(CLUSTER0000131196)-RDINDX10-s2eg.1
2512 13:47:43 unt1 sds01    KFPL00723-I 100 rows loaded,
table=MANAGER."STOCK", RDAREA="RDDATA10"
2476 13:47:43 unt1 0mload0 KFPL00703-I 100 rows loaded
2404 13:47:43 unt1 sds01    KFPL00715-I Index load started
at sds01, index=MANAGER."(CLUSTER0000131196)",
RDAREA="RDINDX10"
2404 13:47:43 unt1 sds01    KFPL00716-I Index load ended
at sds01, index=MANAGER."(CLUSTER0000131196)",
RDAREA="RDINDX10", return code=0
2404 13:47:43 unt1 sds01    KFPL00720-I Index file deleted,
server=sds01,
file=C:/tmp/INDEX-(CLUSTER0000131196)-RDINDX10-s2eg.1
2396 13:47:43 unt1 _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 STOCK 表を検索します。

【入力 SQL】

```
SELECT * FROM STOCK;
```

【SQL の実行結果】

2. HIRDB システムを構築してみましょう

ショウヒンCD	サイコロヨウ	ヒキアテリョウ	ハツチュウテン	シレサキCD
CB0000	0000050.	0000000.	0000030.	DE010
CB0001	0000055.	0000000.	0000030.	DE011
CB0002	0000040.	0000000.	0000020.	DE012
		:		
VD0009	0000033.	0000000.	0000030.	DE019
VD0010	0000035.	0000000.	0000030.	DE020
WP0009	0000039.	0000000.	0000030.	DE049
KFPX27010-I		100 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 (Windows(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。[スタート] - [プログラム] - [HiRDB SQL Executer] - [GUI 版 HiRDB SQL Executer] を選択してください。

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、HiRDB.ini ファイルに認可識別子 (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。

【SQL文】

```
SELECT ショウビンメイ,タンカ FROM GOODS
WHERE タンカ<30000 AND ショウビンメイ LIKE'ソウジキ%' ;
```

【実行結果】

ショウビンメイ	タンカ
ソウジキ	0020000.
ソウジキ(コミハツクシキ)	0011500.
ソウジキ(フジナー)	0021000.

KFPX27010-I 3 rows selected

(3) 表を結合した検索

GOODS 表と STOCK 表を結合し、タンカが 100,000 円以上で、かつザイコリヨウが 50 個以上のシイレサキ C D, ハツチユウテンを検索します。

【SQL文】

```
SELECT GOODS.シイレサキCD, STOCK.ハツチユウテン FROM GOODS, STOCK
WHERE GOODS.ショウビンCD=STOCK.ショウビンCD AND タンカ>=100000 AND
ザイコリヨウ>=50;
```

【実行結果】

3. 表を操作してみましょう

シレサキCD	ハツユウテン
DE011	0000030.
DE001	0000030.
DE003	0000030.
DE049	0000030.
DE048	0000040.
DE005	0000040.
DE021	0000030.

KFPX27010-I 7 rows selected

3.3.2 演習 2 (表へ行を挿入する)

GOODS 表に, 'TV0008', 'テレビ-15ガタ(テレビデオ)', 49800, 'DE003' のデータを挿入します。

【SQL文】

```
INSERT INTO GOODS  
VALUES('TV0008', 'テレビ-15ガタ(テレビデオ)', 49800, 'DE003');
```

【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 GOODS;
```

SQL 文を実行すると、次の検索結果が表示されます。

ショウヒンCD	ショウヒンメイ	タンカ	シレサキCD
CB0000	レイゾウゴ(3トア)	0090000.	DE010
	:		
TE0002	コードレススハンテング(カイトキノウツキ)	0188000.	DE042
TV0008	テレビ-15カタ(テレビデオ)	0049800.	DE003
TE0006	ルスハンテング	0089000.	DE036
	:		
WP0009	ワートプロセッサ	0265000.	DE049
KFPX27010-I			
	101 rows selected		

3.3.3 演習 3 (表の行を更新する)

GOODS 表の、ショウヒンメイがホットプレート(タコヤキプレート)のタンカを、1 割引にします。

【SQL 文】

```
UPDATE GOODS SET タンカ=タンカ*0.9
WHERE ショウヒンメイ='ホットプレート(タコヤキプレート)';
```

【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 GOODS;
```

3. 表を操作してみましょう

SQL文を実行すると、次の検索結果が表示されます。

シヨウヒンCD	シヨウヒンメイ	タンカ	シレサキCD
CB0000	レイゾ°ウカ(3ト°ア)	0090000.	DE010
	:		
H00003	ホツプ°レート	0012000.	DE043
H00004	ホツプ°レート(タコキキ°レート)	0005400.	DE044
KA0005	カラオケセット	0356000.	DE045
	:		
TV0008	テレビ°-15カ°タ(テレビ°テ°オ)	0049800.	DE003
KFPX27010-I			

101 rows selected

3.3.4 演習 4 (表の行を削除する)

GOODS 表の、シヨウヒン C D が CB0002 の行を削除します。

【SQL文】

```
DELETE FROM GOODS WHERE シヨウヒンCD='CB0002';
```

【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 GOODS;
```

SQL文を実行すると、次の検索結果が表示されます。

ショウビンCD	ショウビンメイ	タンカ	シレサキCD
CB0000	レイゾウウコ(3トア)	0090000.	DE010
CB0001	レイゾウウコ(4トア)	0200000.	DE011
CB0003	レイゾウウコ(チルトキノウツキ)	0260000.	DE013
	:		
TV0008	テレビ-15カタ(テレビデオ)	0049800.	DE003
KFPX27010-I			

100 rows selected

3. 表を操作してみましょう

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。[ファイル] - [終了] を選択してください。

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（Windows(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. Windows 2000 のサービスと HiRDB のコマンド (`pdstart` コマンド) で開始する方法
2. Windows 2000 のサービスだけで開始する方法

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。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 = MANUAL1` が指定されています。

2. の方法で開始する場合

`pd_mode_conf = AUTO` を指定します。

(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。

なお，Windows 2000 のサービスで HiRDB を開始する場合は，正常開始または再開しできません。

4. HiRDB を運用してみましょう

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	pdstart	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
強制開始	pdstart dbdestroy	通常は使用しないでください。HiRDB を再開始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 を終了するには、次の方法があります。

1. HiRDB のコマンド（pdstop コマンド）で終了する方法
2. Windows 2000 のサービスで終了する方法

通常は、1. の方法で HiRDB を終了するようにしてください。

（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。

なお、Windows 2000 のサービスで HiRDB を終了する場合は、正常終了しかできません。

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。

終了モード	入力コマンド	終了モードの説明
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。

```
C:¥>pdls
```

(2) pdls コマンドの実行結果が表示されます

```
HOSTNAME(144716)      UNITID  SVID      STATUS   STARTTIME
s10380720             unt1    ***** ACTIVE   131951
s10380720             unt1    sds01    ACTIVE   131951
```

〔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。

4. HiRDB を運用してみましょう

```
C:¥>pdls -d prc
```

(4) pdls コマンドの実行結果が表示されます

```
HOSTNAME : sl0380720(144817)
STATUS  PID  UID  GID  SVID  TIME  PROGRAM  C-PID  C-GRP
L       2452  0    0    sds01  999999
L       2548  0    0    sds01  999999
L       1848  0    0    sds01  999999
L        868  0    0    sds01  999999
L       2536  0    0    sds01  999999
L       2320  0    0    sds01  999999
L       2440  0    0    sds01  999999
L       2296  0    0    sds01  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。
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。

```
C:¥>pdstop
```

(6) pdstop コマンドの実行結果が表示されます

```
1980 14:50:59 unt1 _rdm      KFPS01841-I HiRDB unit unt1 terminated.
mode = NORMAL
1980 14:50:59 unt1 _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。

```
C:¥>pdstart
```

(2) pdstart コマンドの実行結果が表示されます

```
2012 14:53:46 unt1 _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。

```
C:¥>pdls
```

(4) pdls コマンドの実行結果が表示されます

HOSTNAME (145349)	UNITID	SVID	STATUS	STARTTIME
s10380720	unt1	*****	ACTIVE	145345
s10380720	unt1	sds01	ACTIVE	145345

〔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

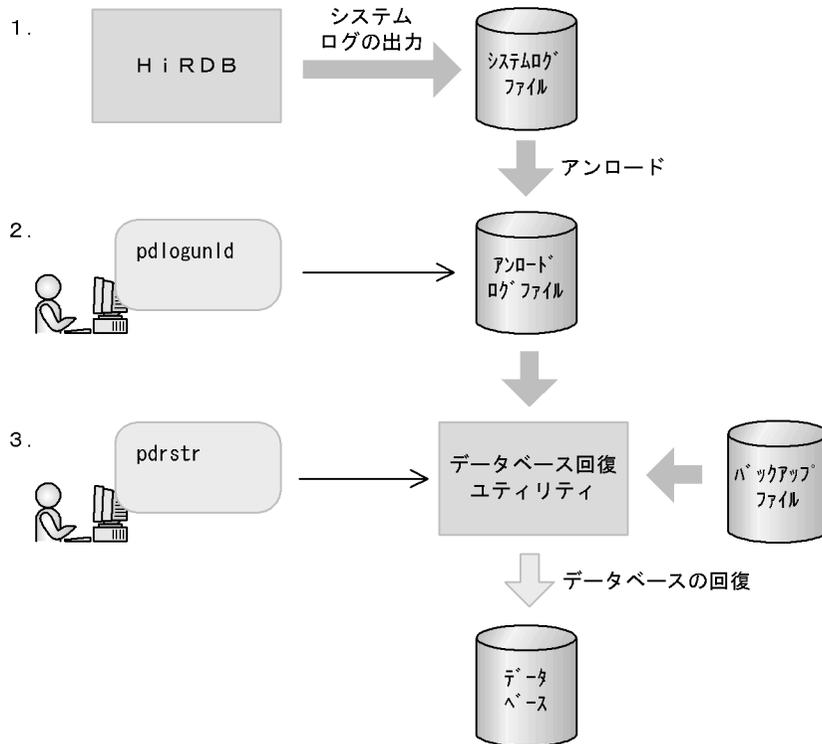
アンロードログファイルは，システムログをアンロードすることで作成することができます。システムログのアンロードは，pdlogunld コマンド（ログ関係ファイルの

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、運用方法が最も単純なアンロード状態のチェックを解除する運用で HiRDB を運用します。「2.3.1 作成する環境」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 (Windows(R) 用)」を参照してく

4. HiRDB を運用してみましょう

ださい。

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に回復できます。 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。このマニュアルではアンロード状態のチェックを解除する運用を利用しているので、この演習は実行する必要はありません。

(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。

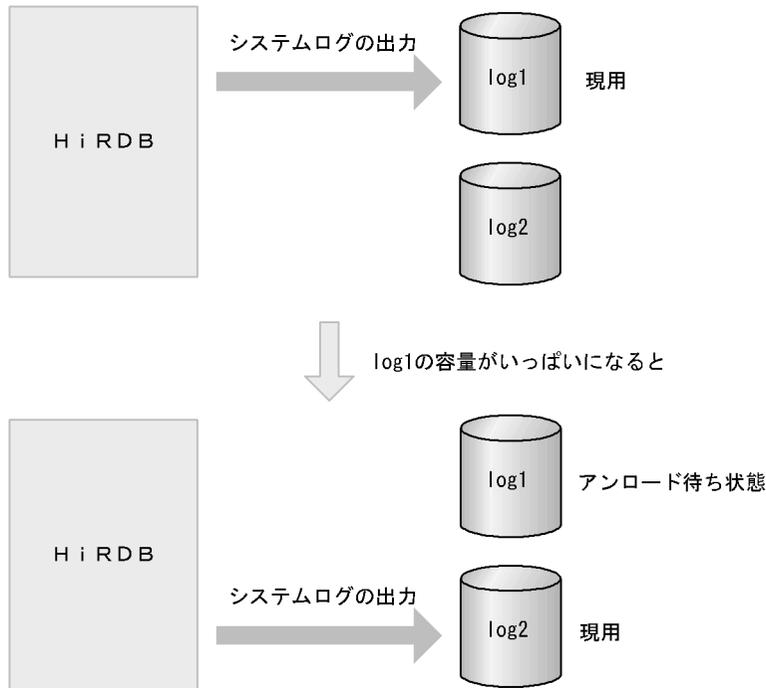
(a) 「アンロード待ち状態」のファイルをアンロードします

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ワップといいます。

「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 (115350)
Group   Type Server  Gen No.  Status  Run ID   Block No.  Ex-Status
log1    sys  sds01    1  osu---u  40174e5a  1          7  -----
log2    sys  sds01    0  oc-d--u  40186bda  0          0  -----
log3    sys  sds01    0  os----- 00000000  0          0  -----
log4    sys  sds01    0  os----- 00000000  0          0  -----
log5    sys  sds01    0  os----- 00000000  0          0  -----
log6    sys  sds01    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 (log1) が、「アンロード待ち状態」のファイルです。

4. HiRDB を運用してみましょう

- 2 カラム目が「c」になっているファイル (log2) が、「現用」のファイルです。

(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の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 (Windows(R) 用)」を参照してください。

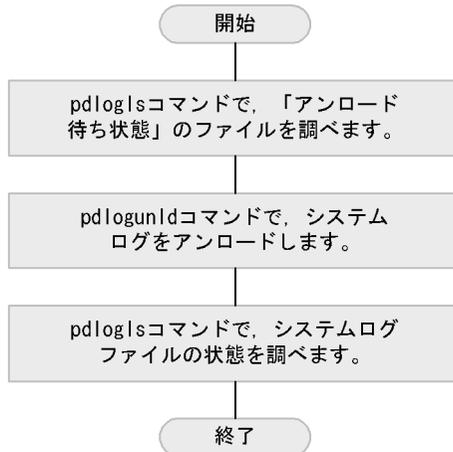
(d) アンロードログファイルの格納場所

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。

```
C:\¥>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。

4. HIRDB を運用してみましょう

(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

```
C:¥>pdlogls -d sys
```

(2) pdlogls コマンドの実行結果が表示されます

```
HOSTNAME : s10380720(111302)
Group    Type  Server  Gen No.  Status  Run ID      Block No.  Ex-Status
log1     sys  sds01   1        osud--u  40186bda    1          1  -----
log2     sys  sds01   2        oc-d--u  40186bda    2          2  -----
log3     sys  sds01   0        os----- 00000000    0          0  -----
log4     sys  sds01   0        os----- 00000000    0          0  -----
log5     sys  sds01   0        os----- 00000000    0          0  -----
log6     sys  sds01   0        os----- 00000000    0          0  -----
```

〔説明〕

- Status の 3 カラム目が u になっている log1 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 log2 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エクスプローラで作成します。
C:¥hirdb¥pdlogunld ディレクトリを作成します。

(4) pdlogunld コマンドで log1 のシステムログをアンロードします

```
C:¥>pdlogunld -d sys -g log1 -o C:¥hirdb¥pdlogunld¥unlog01
```

〔説明〕

- g :
アンロードするシステムログファイル (log1) 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 C:¥hirdb¥pdlogunld ディレクトリ下に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

```
2276 11:21:31 unt1          KFPS01271-I Unloading completed. generation number=1,
start block number=1, end block number=1
```

〔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

```
C:¥>pdlogls -d sys
```

(7) pdlogls コマンドの実行結果が表示されます

```
HOSTNAME : s10380720(112217)
Group    Type Server   Gen No.  Status  Run ID    Block No.  Ex-Status
log1     sys  sds01     1  os-d--u 40186bda  1  1  -----
log2     sys  sds01     2  oc-d--u 40186bda  1  2  -----
log3     sys  sds01     0  os----- 00000000  0  0  -----
log4     sys  sds01     0  os----- 00000000  0  0  -----
log5     sys  sds01     0  os----- 00000000  0  0  -----
log6     sys  sds01     0  os----- 00000000  0  0  -----
```

〔説明〕

log1 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。

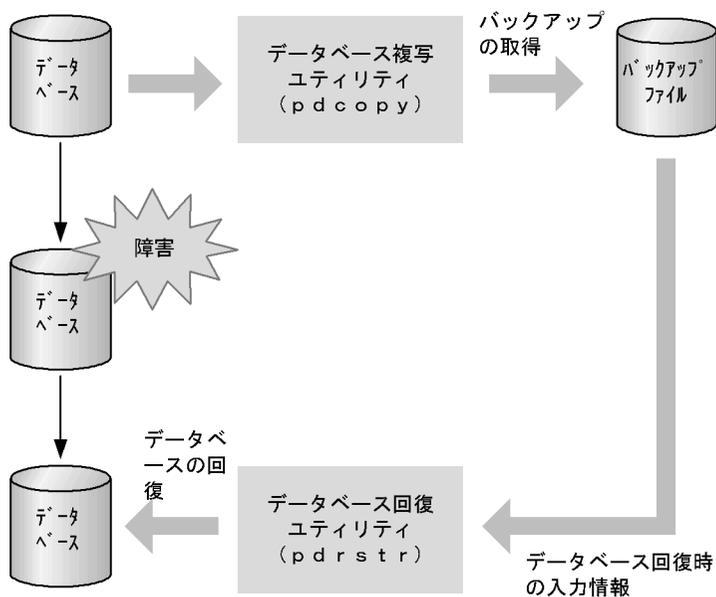
4.4.1 基礎知識

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。 このマニュアルで実施しているアンロード状態のチェックを解除する運用の場合、システム単位のバックアップしか実行できません。 <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。 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、「(5) pdcopy コマンドの処理結果リスト」を参照してください。

4. HiRDB を運用してみましょう

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）バックアップファイルの格納場所

この先行う演習ではバックアップ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、本番用のシステムではバックアップ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。

（5）`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 HRD1
-----
*** 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:C:/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   : RDMAST
  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 C:/win32app/hitachi/hirdb_s/area/rdsys02/rdmast } ..... 5
  EXTENT COUNT : 1
  :
  :
  :

```

〔説明〕

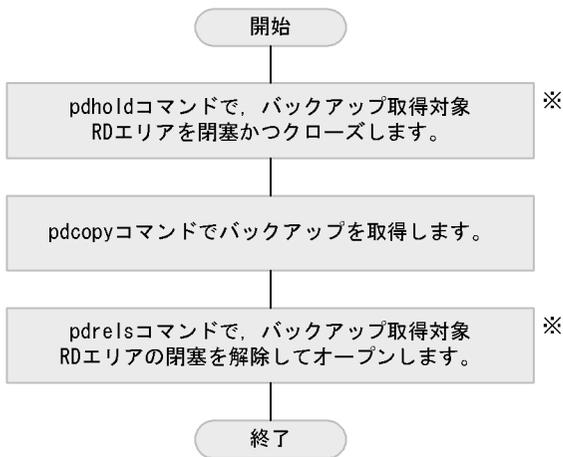
1. バックアップ取得モードに関する情報です。
SHARED : バックアップ取得モードに s を指定
EXCLUSIVE : バックアップ取得モードに x または r を指定
2. システムログファイルに関する情報です。
バックアップ取得モードに s を指定した場合に表示されます。
FILE NAME に表示されているシステムログファイルがデータベースの回復時に必要なシステムログファイルです。
3. バックアップファイルに関する情報です。
4. バックアップ取得対象 RD エリアに関する情報です。
5. バックアップ取得対象 RD エリアのファイル構成に関する情報です。
4 および 5 の情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。

(6) 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。

4. HIRDB を運用してみましょう

図 4-6 バックアップを取得する手順

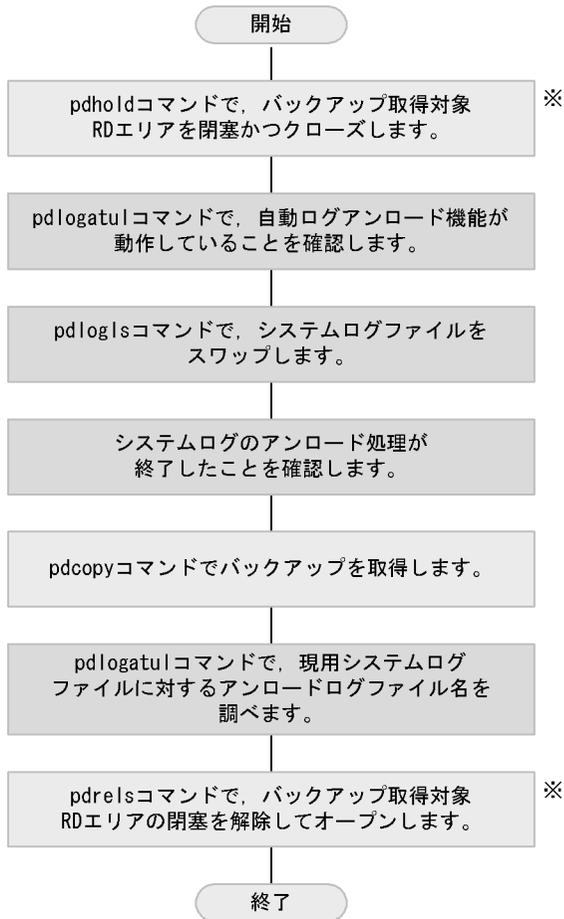


注※ バックアップ取得モードにxを指定する場合に必要な操作です。

(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「(6) 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。

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」の演習で使用するので、保存しておいてください。

4. HiRDB を運用してみましょう

(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エクスプローラで作成します。

C:\¥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。

```
C:\¥>pdcopy -m C:\¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast -a  
-M r -b C:\¥hirdb¥pdcopy¥backup01 -p C:\¥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

(C:\¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast) 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 :

バックアップファイル名を指定します。

C:\¥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。C:\¥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

```
1200 11:47:30 unt1 _pd0copy KFPR00754-I Pdcopy started .....1  
1900 11:47:36 unt1 0bcpy00 KFPR26109-I Backup completed,  
file=c:/hirdb/pdcopy/backup01 .....2  
1200 11:47:31 unt1 _pd0copy KFPR26022-I Output result of  
pdcopy to C:/hirdb/pdcopy/list01.txt .....3  
1200 11:47:36 unt1 _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、C:¥hirdb¥pdcopy¥backup01 です。
3. pdcopy コマンドの処理結果リストが C:¥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、画面およびイベントビューア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 (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。

(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 (¥¥.¥tape0)

4. HiRDB を運用してみましょう

を指定してデータベース複製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。

```
C:¥>pdcopy -m C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast -a -M r
-b ¥¥.¥tape0 -p C:¥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 (C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast) 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 のデバイス名を示す ¥¥.¥tape0 を指定します。DAT にバックアップファイルが作成されます。デバイス名は、[コンピュータの管理] から [記憶域] - [リムーバブル記憶域] - [物理的な場所] を選択したあと、使用している DAT の [プロパティ] - [デバイス情報] を参照することで確認できます。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。C:¥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

```
1692 16:14:20 unt1 _pd0copy KFPR00754-I Pdcopy started .....1
1288 16:14:34 unt1 0bcpy00 KFPR26109-I Backup completed,
file=//./tape0 .....2
1692 16:14:21 unt1 _pd0copy KFPR26022-I Output result of pdcopy to
C:/hirdb/pdcopy/list03.txt .....3
1692 16:14:34 unt1 _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）名は ¥¥.¥tape0 です。
3. pdcopy コマンドの処理結果リストが C:¥hirdb¥pdcopy¥list03.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、リターンコードが 0 で返ってきません。この場合は、画面およびイベントビューア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=../tape9
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 Obcopy0 KFPR16110-I Unable to get
a backup file ../tape9
1616 11:54:17 unt1 _pd0copy KFPR26022-I Output result of
pdcopy to C:/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 ¥¥.¥tape9 を指定すると、このメッセージが表示されます。

4.4.4 演習 3（自動ログアンロード機能を使っている場合に全 RD エリアのバックアップを取得する）

自動ログアンロード機能を使っている場合のバックアップの取得を行います。

バックアップ対象は、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）です。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。

（１）自動ログアンロード機能が動作していることを確認します

自動ログアンロード機能が動作しているか確認します。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

```
C:¥>pdlogatul -d sys
```

〔説明〕

自動ログアンロード機能の動作状態を表示します。

(2) 自動ログアンロード機能の動作状態が表示されます

AUTO_LOG_UNLOAD の値が「ACTIVE」になっていることを確認してください。

```
HOSTNAME : s10380720(170005)
SERVER_NAME:sds01
AUTO_LOG_UNLOAD NOW_UNLOAD_LOG_GROUP CREATE_DIR
      ACTIVE          **** C:/win32app/hitachi/hirdb_s/area/unloadlog/
CURRENT LOG GENERATION INFO.
LOG_GROUP GEN_NO.  SERVER_RUN_ID RUN_ID  UNLOAD_FILE_NAME
log1      1 404d793d      404d7935 sds01_404d793d0001_log1
```

〔説明〕

ACTIVE :

自動ログアンロード機能の動作状態です。表示される値は次のとおりです。

ACTIVE : 動作中

STOP : 停止中

STOPPING : 停止時のアンロード処理待ち監視中

- : 自動ログアンロード機能を使用できない状態

(3) システムログファイルをスワップします

現用のシステムログ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。

```
C:¥>pdlogswap -d sys -w
```

(4) システムログのアンロード処理が終了したことを確認します

スワップしたシステムログファイルがアンロードされたことを確認します。次のコマンドを実行してください。

```
C:¥>pdlogls -d sys
```

(5) pdlogs コマンドの実行結果が表示されます

Status の 3 カラム目に「u」(アンロード待ち状態)のシステムログがないことを確認します。アンロード待ちのシステムログがある場合、少し時間をおいてから確認してください。

```

HOSTNAME : s10380720(170332)
Group    Type Server   Gen No.  Status  Run ID    Block No.  Ex-Status
log1     sys  sds01     1  os-d--u 404d7935    1          1  -----
log2     sys  sds01     2  oc-d--u 404d7935    2          2  -----
log3     sys  sds01     0  os----- 00000000    0          0  -----
log4     sys  sds01     0  os----- 00000000    0          0  -----
log5     sys  sds01     0  os----- 00000000    0          0  -----
log6     sys  sds01     0  os----- 00000000    0          0  -----

```

(6) 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。

```

C:¥>pdcopy -m C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast -a
-M r -b C:¥hirdb¥pdcopy¥backup03 -p C:¥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 (C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast) 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 :

バックアップファイル名を指定します。
C:¥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。C:¥hirdb¥pdcopy

4. HIRDB を運用してみましょう

ディレクトリ下に list03.txt ファイルが作成されます。

(7) 現用システムログファイルに対するアンロードログファイル名を調べます

自動ログアンロード機能の動作状態を表示して、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ときに必要なアンロードログファイルを確認します。次のコマンドを実行してください。

```
C:¥>pdlogatul -d sys
```

〔説明〕

自動ログアンロード機能の動作状態を表示します。

(8) 自動ログアンロード機能の動作状態が表示されます

UNLOAD_FILE_NAME に示されているアンロードログファイル名を確認して、保管してください。取得したバックアップを使って最新の状態にデータベースを回復する場合、このファイル、およびこれ以降にアンロードされるファイルが必要になります。

```
HOSTNAME : s10380720(170005)
SERVER_NAME:sds01
AUTO_LOG_UNLOAD NOW_UNLOAD_LOG_GROUP CREATE_DIR
ACTIVE ***** C:/win32app/hitachi/hirdb_s/area/unloadlog/
CURRENT LOG GENERATION INFO.
LOG_GROUP GEN_NO. SERVER_RUN_ID RUN_ID UNLOAD_FILE_NAME
log1 1 404d793d 404d7935 sds01_404d793d0001_log1
```

sds01_404d793d0001_log1 :

現用世代を自動ログアンロードしたときのアンロードログファイル名です。

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。

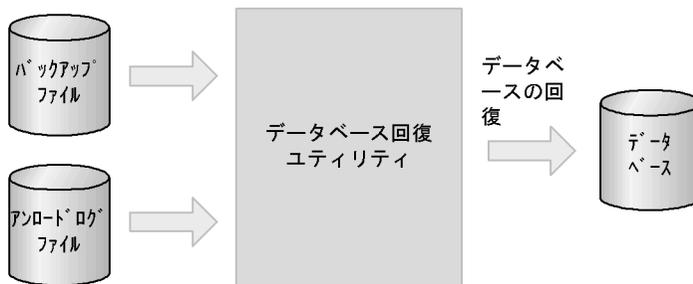
4.5.1 基礎知識

データベースに障害が発生した場合、事前に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



データベースは次に示す時点 (状態) に回復できます。

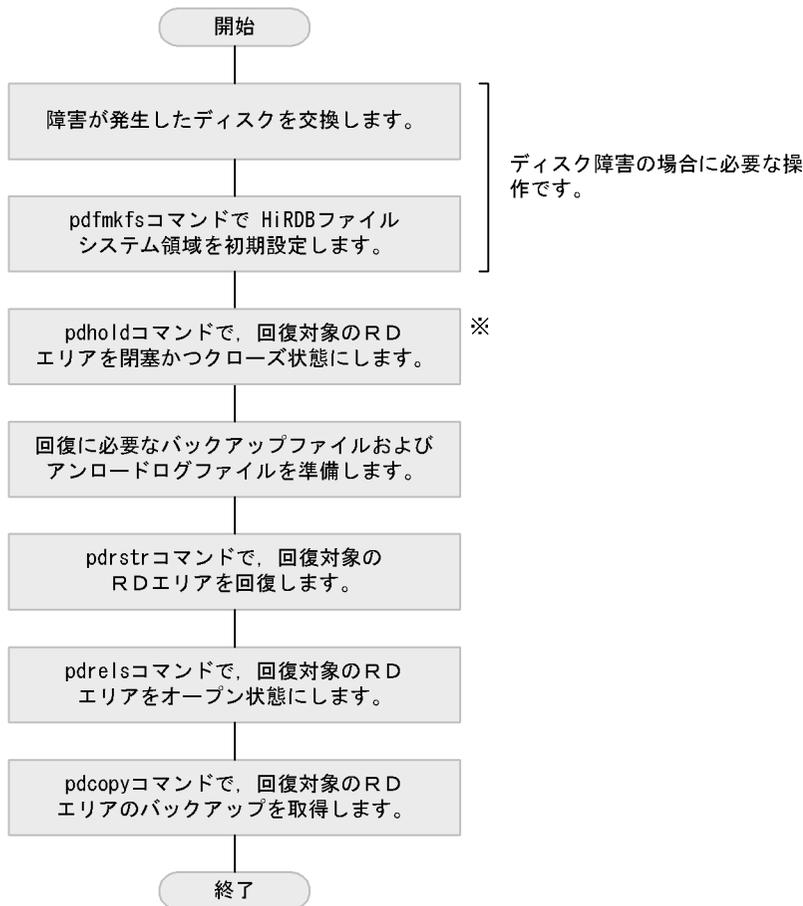
- バックアップ取得時点
バックアップ取得時点に回復する場合は、システムログやアンロードログファイルは必要ありません。バックアップファイルだけがが必要です。
- 最新の状態 (最新の同期点)
最新の状態に回復する場合は、バックアップファイルのほかに、バックアップ取得以降に出力されたシステムログ、またはそれに対応するアンロードログファイルが必要になります。

(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。

4. HiRDB を運用してみましょう

図 4-8 データベースを回復する手順



注※ pdrstrコマンドの実行対象となるRDエリアは、閉塞かつクローズ状態である必要があります。したがって、この操作が必要になります。

(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。

(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

4. HiRDB を運用してみましょう

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。

この演習の場合、次のコマンドになります。

```
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys011"
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys012"
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys013"
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys014"
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys015"
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys016"
pdfmkfs -r -n 50 -l 30 -e 1 -k DB
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02"
pdfmkfs -r -n 50 -l 60 -e 255 -k WORK
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys03"
pdfmkfs -r -n 100 -l 30 -e 1 -k DB
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys04"
pdfmkfs -r -n 20 -l 10 -e 1 -k DB
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys05"
c
```

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エクスプローラで作成します。C:¥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
「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、C:\¥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。

```
C:¥>pdstop
```

(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。

```
C:¥>pdfrm C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast      .....1
C:¥>pdfrm C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rddirt      .....2
C:¥>pdfrm C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rddict      .....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。

```
C:¥>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に回復します

「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。

4. HiRDB を運用してみましょう

```
C:\>pdrstr -m C:\win32app\hitachi\hirdb_s\area\rdsys02\rdmast -a  
-b C:\hirdb\pdcopy\backup01 -p C:\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 (C:\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。
C:\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

```
2016 11:36:43 unt1 _pd0rstr KFPR00764-I Pdrstr started .....1  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"RDMAST" ended normally .....2  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"RDDIRT" ended normally  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"RDDICT" ended normally  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"DIC_RTN_SRC" ended normally  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"DIC_RTN_OBJ" ended normally  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"RDDATA10" ended normally  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"RDINDX10" ended normally  
2224 11:36:59 unt1 0rrstr0 KFPR00766-I Recovery of  
"RLOB1" ended normally  
2016 11:36:59 unt1 _pd0rstr KFPR26222-I Output result of pdrstr  
to C:/hirdb/pdrstr/list01.txt .....3  
2016 11:37:00 unt1 _pd0rstr KFPR00765-I Pdrstr terminated,  
return code=0 .....4
```

〔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 C:\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。

```
C:¥>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。

```
C:¥>pdstart
```

4.5.3 演習 2 (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。

この演習の場合、次のコマンドになります。

4. HiRDB を運用してみましょう

```
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys011"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys012"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys013"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys014"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys015"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys016"

pdfmkfs -r -n 50 -l 30 -e 1 -k DB
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02"

pdfmkfs -r -n 50 -l 60 -e 255 -k WORK
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys03"

pdfmkfs -r -n 100 -l 30 -e 1 -k DB
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys04"

pdfmkfs -r -n 20 -l 10 -e 1 -k DB
-i "C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys05"
```

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
「4.4 バックアップを取得する」の演習 2 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 のデバイス名を示す ¥¥.¥tape0 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。

```
C:¥>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。

```
C:\>pdfrm C:\win32app\hitachi\hirdb_s\area\rdsys02\rdmast .....1
C:\>pdfrm C:\win32app\hitachi\hirdb_s\area\rdsys02\rd dirt .....2
C:\>pdfrm C:\win32app\hitachi\hirdb_s\area\rdsys02\rd dict .....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。

(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

```
C:\>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 2 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。

```
C:\>pdrstr -m C:\win32app\hitachi\hirdb_s\area\rdsys02\rdmast -a
-b ¥¥.¥tape0 -p C:\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。

4. HIRDB を運用してみましょう

-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 2 で全 RD エリアのバックアップを取得した DAT のデバイス名 (¥¥.¥tape0) 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。C:¥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

```
2456 10:58:03 unt1 _pd0rstr KFPR00764-I Pdrstr started .....1
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RDMAST" ended normally .....2
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RDDIRT" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RDDICT" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"DIC_RTN_SRC" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"DIC_RTN_OBJ" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RDATA10" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RDINDX10" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RLOB1" ended normally
2636 10:58:26 unt1 0rrstr0 KFPR00766-I Recovery of
"RDAREA02" ended normally
2456 10:58:29 unt1 _pd0rstr KFPR26222-I Output result of
pdrstr to C:/hirdb/pdrstr/list03.txt .....3
2456 10:58:29 unt1 _pd0rstr KFPR00765-I Pdrstr terminated,
return code=0 .....4
```

〔説明〕

1. pdrstr コマンドの処理が開始しました。
2. 各 RD エリアが回復されました。
3. 処理結果リストが C:¥hirdb¥pdrstr¥list03.txt に出力されました。
4. 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。

```
C:¥>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。

```
C:¥>pdstart
```

4.5.4 演習 3 (自動ログアンロード機能を使っている場合に回復する)

自動ログアンロード機能を使っている場合のデータベースの回復を行います。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) を最新の状態に回復します。

全 RD エリアを最新の状態に回復する場合、幾つかの RD エリアに分けて回復のための操作をする必要があります。詳細については、マニュアル「HiRDB Version 7 システム運用ガイド (Windows(R) 用)」を参照してください。

(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

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 3 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、C:¥hirdb¥pdcopy¥backup03 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、バックアップを取得したときの最後に確認したアンロードログファイル、それ以降にアンロードされたアンロードログファイル、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。HiRDB に自動アンロードされたアンロードログファイルについては、自分で管理しているアンロードログファイルの時系列リストなどで確認してください。

このマニュアルの場合、バックアップ取得後にアンロードされたアンロードログファイルは、次のファイルと仮定します。

- sds01_404d793d0002_log2

4. HiRDB を運用してみましょう

(2) 現用のシステムログファイルをスワップします

最新の更新情報が入っている現用のシステムログファイルの内容をアンロードするためにスワップします。次のコマンドを実行してください。

```
C:¥>pdlogswap -d sys -w
```

(3) システムログのアンロード処理が終了したことを確認します

スワップしたシステムログファイルがアンロードされたことを確認します。次のコマンドを実行してください。

```
C:¥>pdlogls -d sys
```

(4) pdlogls コマンドの実行結果が表示されます

Status の 3 カラム目に「u」(アンロード待ち状態)のシステムログがないことを確認します。アンロード待ちのシステムログがある場合、少し時間をおいてから確認してください。

```
HOSTNAME : s10380720(19075)
Group    Type Server  Gen No.  Status  Run ID      Block No.  Ex-Status
log1     sys  sds01    1  os----u  404d7935   1          1  -----
log2     sys  sds01    2  os----u  404d7935   2          6  -----
log3     sys  sds01    0  os-d--u  404d7935   7          7  -----
log4     sys  sds01    0  oc-d--u  404d7935   8          8  -----
log5     sys  sds01    0  os----- 00000000   0          0  -----
log6     sys  sds01    0  os----- 00000000   0          0  -----
```

(5) 作成されたアンロードログファイルを確認します

(4)でアンロードされたアンロードログファイル名を確認します。アンロードログファイルの出力ディレクトリを確認してください。このマニュアルの場合、ファイル名は「sds01_404d793d0003_log3」になります。

(6) 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。

```
C:¥>pdstop
```

(7) 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。

```
C:¥>pdfrm C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast      .....1
C:¥>pdfrm C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rddirt      .....2
C:¥>pdfrm C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rddict      .....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。

(8) 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

```
C:¥>pdstart -r
```

(9) pdrstr コマンドでマスタディレクトリ用 RD エリアを最新の状態で回復します

バックアップ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。

```
C:¥>pdrstr -m C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast -r rdmast
-b C:¥hirdb¥pdcopy¥backup03
-l C:¥win32app¥hitachi¥hirdb_s¥area¥unloadlog¥sds01_404d793d0002_log2
  C:¥win32app¥hitachi¥hirdb_s¥area¥unloadlog¥sds01_404d793d0003_log3
-p C:¥hirdb¥pdrstr¥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 ファイル名称を指定します。
- r :
マスタディレクトリ用 RD エリアを指定します。
- 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 で取得したバックアップファイル名 (backup03) 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。
- 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。
C:¥hirdb¥pdrstr ディレクトリ下に list03.txt ファイルが作成されます。

(10) pdrstr コマンドでデータディレクトリ用 RD エリアなどをバックアップ取得時点で回復します

次に示す RD エリアをバックアップ取得時点で回復します。データベース回復ユティリティを実行してください。

- データディレクトリ用 RD エリア : RDDIRT
- データディクショナリ用 RD エリア : RDDICT
- データディクショナリ LOB 用 RD エリア (ソース格納用) : DIC_RTN_SRC

```
C:¥>pdrstr -m C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys02¥rdmast  
-r rddirt,rddict,dic_rtn_src  
-b C:¥hirdb¥pdcopy¥backup03  
-p C:¥hirdb¥pdrstr¥list04.t
```

[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。
- r :
データディレクトリ用 RD エリアなどの RD エリアを指定します。
- 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 で取得したバックアップファイル名 (backup03) 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。

C:\¥\hirdb¥\pdrstr ディレクトリ下に list04.txt ファイルが作成されます。

(11) 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。

```
C:\¥>pdstop
```

(12) 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。

```
C:\¥>pdstart
```

(13) pdhold コマンドで RD エリアを閉塞、クローズ状態にします

マスタディレクトリ用 RD エリアおよびデータディクショナリ用 RD エリアを除いた 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。

```
C:\¥>pdhold -r rddirt,dic_rtn_src,dic_rtn_obj,rddata10,rdindx10,rlob1 -c
```

〔説明〕

-r:

閉塞、クローズ状態にする RD エリアを指定します。

(14) pdhold コマンドでデータディクショナリ用 RD エリアを閉塞、クローズ状態にします

データディクショナリ用 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。

```
C:\¥>pdhold -r rddict -c
```

〔説明〕

-r:

データディクショナリ用 RD エリアを指定します。

4. HiRDB を運用してみましよう

(15) pdrstr コマンドでマスタディレクトリ用 RD エリア以外の RD エリアを最新の状態に回復します

マスタディレクトリ用 RD エリアのときと同様に、マスタディレクトリ用 RD エリア以外の 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。

```
C:\>pdrstr -m C:\win32app\hitachi\hirdb_s\area\rdsys02\rdmast
-r rddict,rddirt,dic_rtn_src,dic_rtn_obj,rddata10,rdindx10,rlob1
-b C:\hirdb\pdcopy\backup03
-l C:\win32app\hitachi\hirdb_s\area\unloadlog\sds01_404d793d0002_log2
  C:\win32app\hitachi\hirdb_s\area\unloadlog\sds01_404d793d0003_log3
-p C:\hirdb\pdrstr\list05.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。
- r :
マスタディレクトリ用 RD エリア以外の RD エリアの RD エリアを指定します。
- 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 で取得したバックアップファイル名 (backup03) 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。
- 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。
C:\hirdb\pdrstr ディレクトリ下に list05.txt ファイルが作成されます。

(16) pdrels コマンドでデータディクショナリ用 RD エリアの閉塞を解除し、オープン状態にします

データディクショナリ用 RD エリアの閉塞を解除し、オープン状態にします。次のコマンドを実行してください。

```
C:\>pdrels -r rddic -o
```

〔説明〕

- r :
閉塞、クローズ状態にする RD エリアを指定します。

(17) pdrels コマンドで RD エリアの閉塞を解除し、オープン状態にします

マスタディレクトリ用 RD エリアおよびデータディクショナリ用 RD エリアを除いた RD エリアの閉塞を解除し、オープン状態にします。次のコマンドを実行してください。

```
C:>pdrels -r rddirt,dic_rtn_src,dic_rtn_obj,rddata10,rdindx10,rlob1 -o
```

〔説明〕

-r :

閉塞、クローズ状態にする RD エリアを指定します。

(18) pdcopy コマンドで全 RD エリア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。

```
C:>pdcopy -m C:\win32app\hitachi\hirdb_s\area\rdsys02\rdmast -a  
-M r -b C:\hirdb\pdcopy\backup04 -p C:\hirdb\pdcopy\list06.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

(C:\win32app\hitachi\hirdb_s\area\rdsys02\rdmast) 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 :

バックアップファイル名を指定します。

C:\hirdb\pdcopy ディレクトリ下に backup04 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。C:\hirdb\pdcopy ディレクトリ下に list06.txt ファイルが作成されます。

4. HiRDB を運用してみましょう

(19)pdstop コマンドで HiRDB を正常終了します

HiRDB を正常終了してください。

```
C:¥>pdstop
```

(20)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。

```
C:¥>pdstart
```

4.6 表を再編成する

表の再編成について説明します。

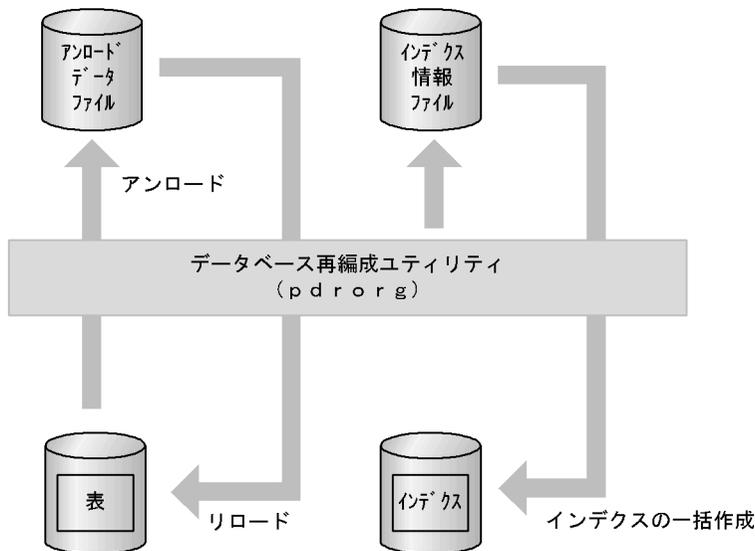
4.6.1 基礎知識

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〕

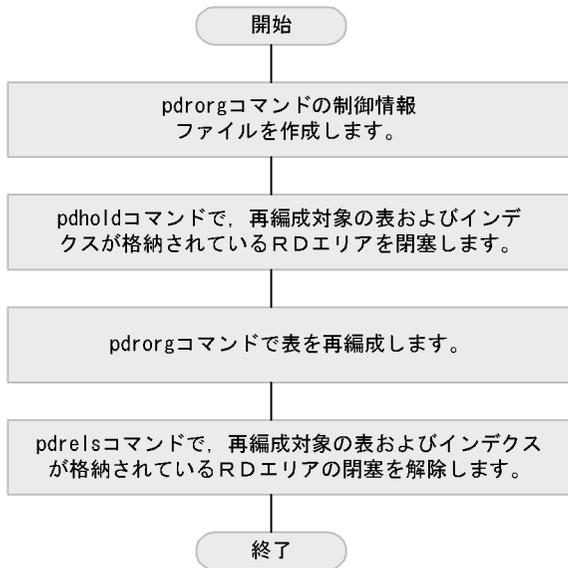
- 表データを取り出し、専用のファイルに保存します。これを表データのアンロードといいます。また、専用のファイルをアンロードデータファイルといいます。その後、表にデータを格納し直します。これを表データのリロードといいます。これら全体の処理を表の再編成といいます。
- 表にインデクスが定義されていると、データをリロードするときにインデクス情報ファイルにインデクス情報が出力されます。その情報を基にして HiRDB がインデクスを一括作成します。これによって、インデクスも再編成されます。

4. HIRDB を運用してみましょう

(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。

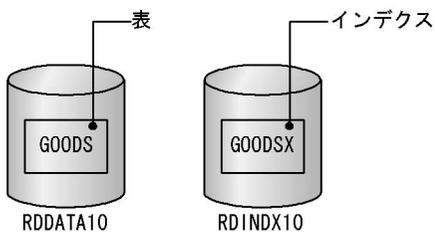
図 4-10 表を再編成する手順



4.6.2 演習（表を再編成する）

GOODS 表を再編成します。

- GOODS 表にはインデクス（GOODSX）が定義されています。
- GOODS 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（GOODSX）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
- 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、C:\%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、C:\%work%\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 C:\%hirdb%\pdrorg\%seigyo01.txt を作成してください。

```
unload C:\%hirdb%\pdrorg%\unld01 .....1
idxwork C:\%work%\idxdir .....2
sort C:\%tmp .....3
```

〔説明〕

1. アンロードデータファイル名を指定します。
C:\%hirdb%\pdrorg ディレクトリ下に unld01 ファイルが作成されます。
C:\%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。
C:\%work%\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。

```
C:\%>pdhold -r RDDATA10,RDINDX10
```

(4) pdhold コマンドの実行結果が表示されます

```
2524 11:54:46 unt1 sds01 KFPH00130-I RDAREA held, RDAREA = "RDDATA10"
2524 11:54:46 unt1 sds01 KFPH00130-I RDAREA held, RDAREA = "RDINDX10"
2524 11:54:46 unt1 sds01 KFPH00110-I pdhold command completed
```

(5) pdrorg コマンドで GOODS 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。

```
C:\%>pdrorg -k rorg -t GOODS -u MANAGER
C:\%hirdb%\pdrorg\%seigyo01.txt
```

4. HIRDB を運用してみましょう

〔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 (GOODS) 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。
- C:¥hirdb¥pdrorg¥seigyo01.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

```
2472 11:55:42 unt1 sds01      KFPL00715-I Index load started at sds01,  
index=MANAGER."GOODSX", RDAREA="RDINDX10"  
2472 11:55:42 unt1 sds01      KFPL00716-I Index load ended at sds01,  
index=MANAGER."GOODSX", RDAREA="RDINDX10", return code=0  
2472 11:55:42 unt1 sds01      KFPL00720-I Index file deleted, server=sds01,  
file=C:/work/idxdir/INDEX-GOODSX-RDINDX10-s2d8.1  
1384 11:55:42 unt1 _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 RDINDX10 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。

```
C:¥>pdrels -r RDDATA10,RDINDX10
```

(9) pdrels コマンドの実行結果が表示されます

```
2440 11:56:30 unt1 sds01      KFPH00130-I RDAREA released, RDAREA = "RDDATA10"  
2440 11:56:30 unt1 sds01      KFPH00130-I RDAREA released, RDAREA = "RDINDX10"  
2440 11:56:30 unt1 sds01      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。残容量が少なくなったら、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（Windows(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。

4. HIRDB を運用してみましょう

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) pddb1s 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。

```
C:¥>pddb1s -r RDDATA10 -a
```

(2) pddb1s 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    6      OPEN       USER     INITIAL
SEGMENT 2098 / 2100
```

〔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 2098 で、全セグメント数が 2100 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 (Windows(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エクスプローラで作成します。C:\%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 C:\%hirdb%\pdmod\%expand.txt を作成してください。

```
expand rdarea RDDATA10          .....1
file name "C:\%win32app%\hitachi%\hirdb_s\%area%\rdsys04%\rddata10a" .....2
initial 100 segments ;          .....3
```

〔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。

C:\%win32app%\hitachi%\hirdb_s\%area%\rdsys04 ディレクトリ下に rddata10a ファイルが作成されます。

C:\%win32app%\hitachi%\hirdb_s\%area%\rdsys04 は Windows 2000 のファイルパスです。

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。

```
C:\%>pdmod -a C:\%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。

4. HIRDB を運用してみましょう

(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 13:48:26 on 2004/01/29
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、イベントビューア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="C:/win32app/hitachi/hirdb_s/area/rdsys04/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。

```
C:¥>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    6      OPEN       USER     INITIAL
SEGMENT 2198 / 2200
```

〔説明〕

全セグメント数が 2100 から 2200 に変わっています。それに伴い未使用セグメント数も 2098 から 2198 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

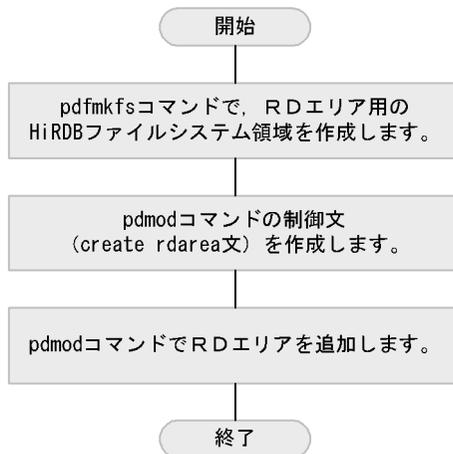
pdmod コマンドの create rdarea 文で、RD エリアを追加します。新規の表を作成する場合などに RD エリアを追加します。

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 pdbufsls 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。

4. HiRDB を運用してみましょう

(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エクスプローラで作成します。C:\%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。

```
C:\%hirdb%\rdarea\%area01
```

〔説明〕

RD エリア用の HiRDB ファイルシステム領域を C:\%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。

C:\%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。C:\%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イベントビューア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。

(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。

```
C:¥>pdfstatfs C:¥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 (NOLOB)          .....10
initialize user id          Administrator       .....11
initialize time              Thu Jan 29 13:52:30 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 1 月 29 日 木曜日 13 時 52 分 30 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

C:¥hirdb¥rdarea ディレクトリ下に area01 ファイルが作成されています。エクスプローラで確認してください。

4. HIRDB を運用してみましょう

(5) pdbufsls 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。

```
C:¥>pdbufsls -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) pdbufsls コマンドの実行結果が表示されます

```
DEFINE OF GLOBAL BUFFER
EDIT TIME 2004-01-29 13:53:49
BUFFNAME      SVID      TYPE  SIZE  NUM  WRATIO  RDAREA/INDEX NAME
              PRMAX  PRNUM  CSIZE  MAPS
gbuf01        sds01      R     4k    20   20      "RDMAST"
              0 32 ***** **
              "RDDIRT"
gbuf02        sds01      R     4k    20   20      "RDDICT"
              0 32 ***** **
gbuf04        sds01      R     4k   10000  20      "RDDATA10"
              0 32 ***** **
gbuf05        sds01      R     4k   10000  20      "RDINDX10"
              0 32 ***** **
gbuf03        sds01      O     8k   10000  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 gbuf01 , gbuf02 , gbuf04 , gbuf05 およびその他の RD エリア用 (TYPE=O) の gbuf03 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 C:¥hirdb¥pdmod¥create.txt を作成してください。

```

create rdarea RDAREA01 .....1
globalbuffer gbuf04 .....2
for user used by PUBLIC .....3
page 4096 characters .....4
storage control segment 10 pages .....5
file name "C:¥hirdb¥rdarea¥area01¥rdarea01" .....6
initial 10 segments; .....7

```

〔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 (gbuf04) 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 pdbuffer 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)
pddbuffer -a gbuf04 -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。C:¥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、C:¥hirdb¥rdarea¥area01 ディレクトリ下に rdarea01 ファイルが作成されます。C:¥hirdb¥rdarea¥area01 は Windows 2000 のファイルパスです。
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ーティリティ (pdmod) を実行します。次のコマンドを実行してください。

```
C:¥>pdmod -a C:¥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。

4. HIRDB を運用してみましょう

(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 13:56:50 on 2004/01/29  
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、イベントビューア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。

```
C:¥>pddbls -r ALL
```

(11) pddbls コマンドの実行結果が表示されます

STATE OF RDAREA	ID	STATUS	TYPE	OPNMODE
RDAREA	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

[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 C:¥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。
```

〔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ティリティ (pddef) を実行します。

```
C:¥>pddef < C:¥hirdb¥pddef¥c_table.txt
```

(14) pddef コマンドの実行結果が表示されます

```
KFPAL2000-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】

4. HIRDB を運用してみましょう

```
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
KFPX27010-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】

4. HiRDB を運用してみましょう

```
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。 表およびインデッ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。 表およびインデッ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ッ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ッ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
- GOODS 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 GOODS 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 GOODS 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、GOODS 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。[ファイル] - [切り離し] を選択して、HiRDB から DISCONNECT してください。これで MANAGER が HiRDB から DISCONNECT されました。

そのあと、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。

[ファイル] - [接続] を選択してください。【CONNECT】ダイアログが表示されます。ユーザ名「USER01」、パスワード「PASS01」、ホスト名「s10380720」およびポート番号「22200」を指定して「OK」をクリックしてください。



(2) GOODS 表を検索します

HiRDB に接続したら、GOODS 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.GOODS;
```

〔説明〕

GOODS 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】

```

      ⋮
VD0006 ビデオ-SVHS (BSタイプ)          0170000. DE016
VD0007 ビデオ-SVHS (G-CODE)            0180000. DE017
VD0008 ビデオ-テープ                    0125000. DE018
VD0009 ビデオ-Eテープ                  0140000. DE019
VD0010 ビデオ-PRO (キョウムヨウ)       0500000. DE020
WP0009 ワードプロセッサ                  0265000. DE049

KFPX27010-I          100 rows selected

```

〔説明〕

GOODS 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 STOCK 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.STOCK;
```

〔説明〕

STOCK 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】

```
KFPA11548-E Unable to execute SELECT:no SELECT privilege for
table MANAGER.STOCK
```

〔説明〕

4. HiRDB を運用してみましょう

STOCK 表に対する SELECT 権限がないため、STOCK 表を検索できません。

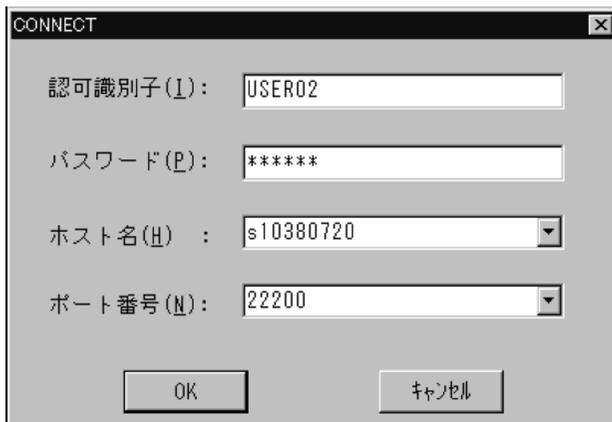
4.9.5 演習 4 (権限がないユーザで表を検索する)

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 です。[ファイル] - [切り離し] を選択して、HiRDB から DISCONNECT してください。これで USER01 が HiRDB から DISCONNECT されました。

[ファイル] - [接続] を選択してください。【CONNECT】ダイアログが表示されます。ユーザ名「USER02」、パスワード「PASS02」、ホスト名「s10380720」およびポート番号「22200」を指定して「OK」をクリックしてください。



(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。



〔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。[ファイル] - [切り離し] を選択して、HiRDB から DISCONNECT してください。これで USER01 が HiRDB から DISCONNECT されました。

そのあと、【CONNECT】ダイアログで、ユーザ名「MANAGER」、パスワード「MANAGER」、ホスト名「s10380720」およびポート番号「22200」を指定して「OK」をクリックしてください。HiRDB に CONNECT されます。



(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】

4. HIRDB を運用してみましょう

TABLE_SCHEMA	TABLE_NAME
:	:
HIRDB	SQL_VIEW_DEF
HIRDB	SQL_VIEW_TABLE_USAGE
MANAGER	EMP
MANAGER	GOODS
MANAGER	STOCK
MASTER	SQL_ALIASES
MASTER	SQL_COLUMNS
:	:
KFPX27010-I	80 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。

-
- 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

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、[コントロールパネル] - [管理ツール] - [サービス] で「HiRDB CM - Agent」のサービスを開始してください。

(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。

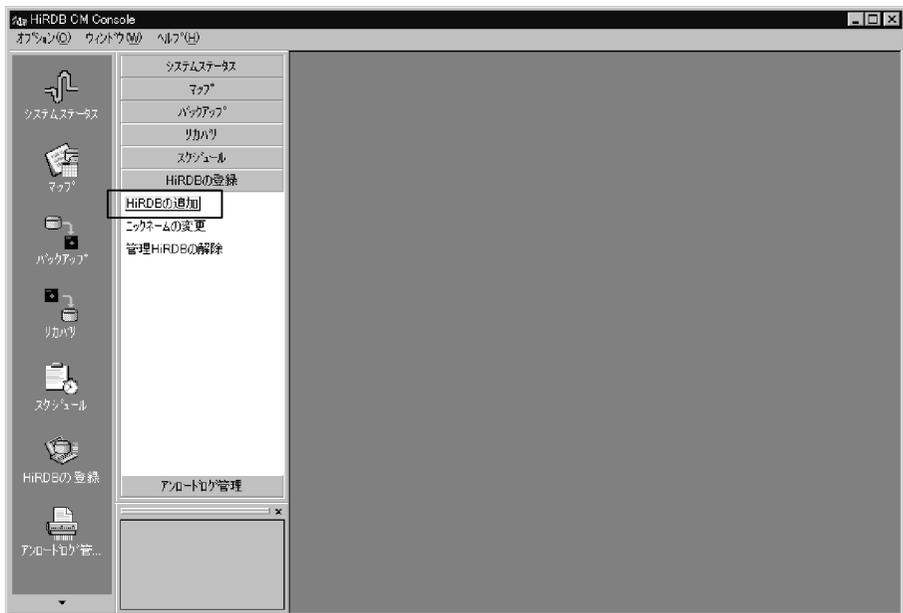
注

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。



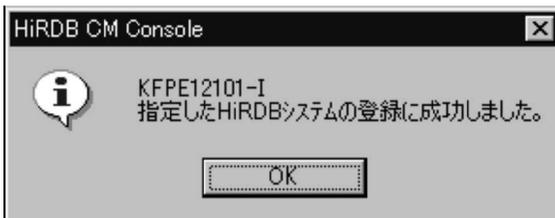
2. 【HiRDB システムの新規登録】ボックスが表示されます。操作したい HiRDB の IP アドレス「10.210.35.24」とその HiRDB に対応する HiRDB Control Manager - Agent が使用するポート番号「20293」を指定して、「OK」ボタンを

5. HiRDB Control Manager で HiRDB を運用してみましょう

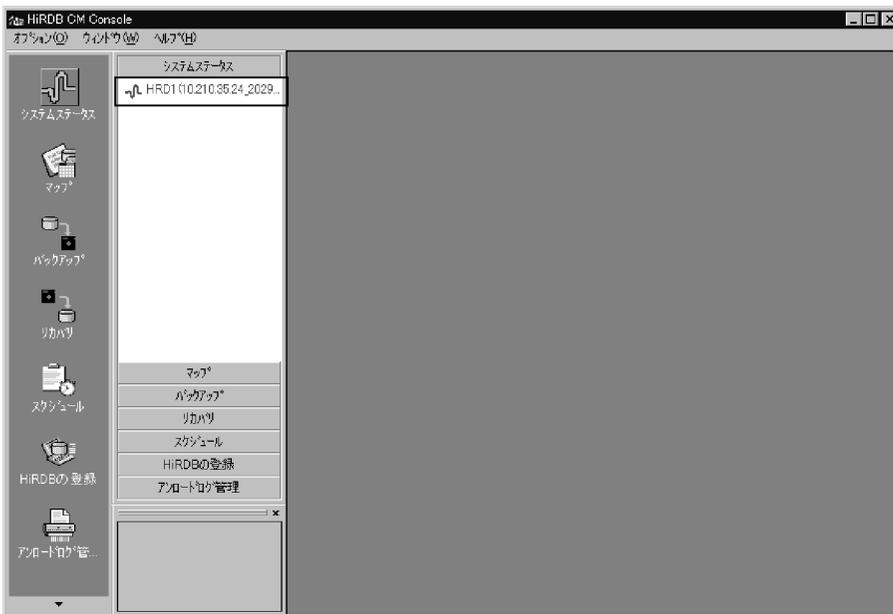
クリックしてください。



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

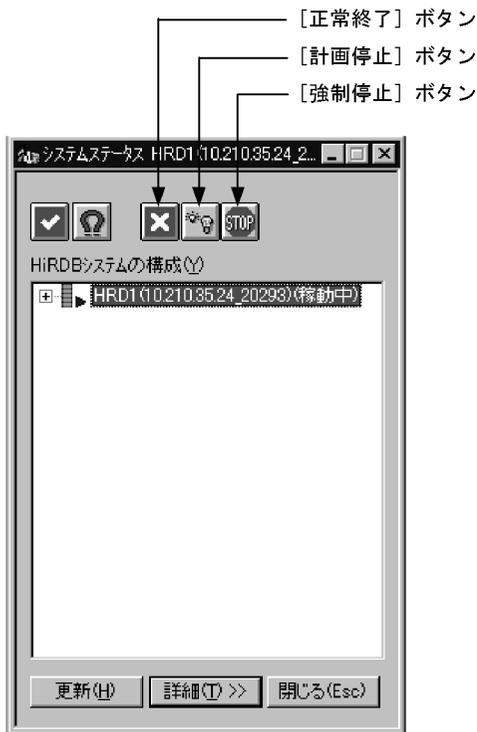
[システムステータス] 欄の管理 HiRDB をクリックすると , [システムステータス] ボックスが表示されます。[システムステータス] ボックスでは , 管理 HiRDB の開始 , 終了や状態表示などの操作が実行できます。



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。

5. HiRDB Control Manager で HiRDB を運用してみましょう



〔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。

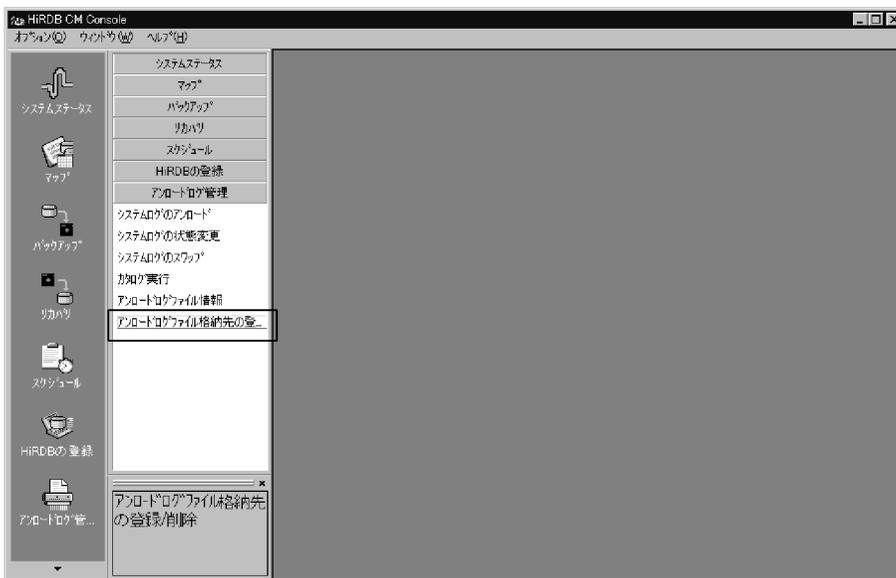
5.5 システムログを操作する

このマニュアルではアンロード状態のチェックを解除する運用を行っているのですが、マニュアルどおりに演習を実施している方はこの演習を実施する必要はありません。

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。

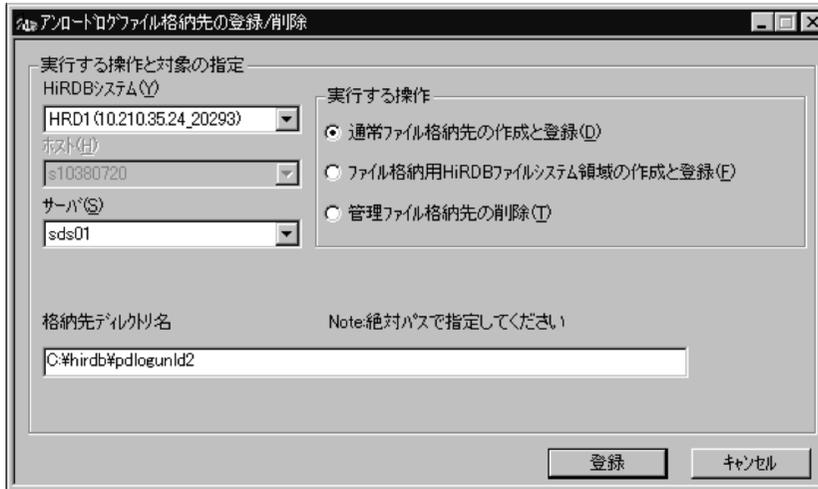
操作対象と操作内容を次のように指定してください。

HiRDB システム : HRD1 (10.210.35.24_20293)

サーバ : sds01

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 : C:\¥hirdb¥pdlogunld2



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）

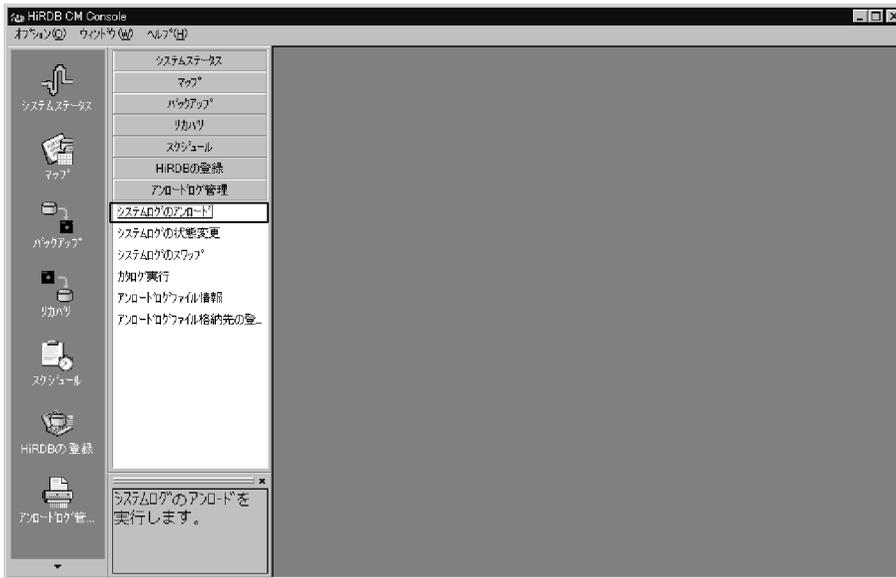
「アンロード待ち状態」のシステムログファイルをアンロードします。

- (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. 【システムログのダウンロード】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 : HRD1 (10.210.35.24_20293)

サーバ : sds01

ダウンロードログファイルの最大保持数 : 256

ダウンロードログファイルの種別 : 通常ファイル

ダウンロードログファイルの格納先 : C:\¥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。

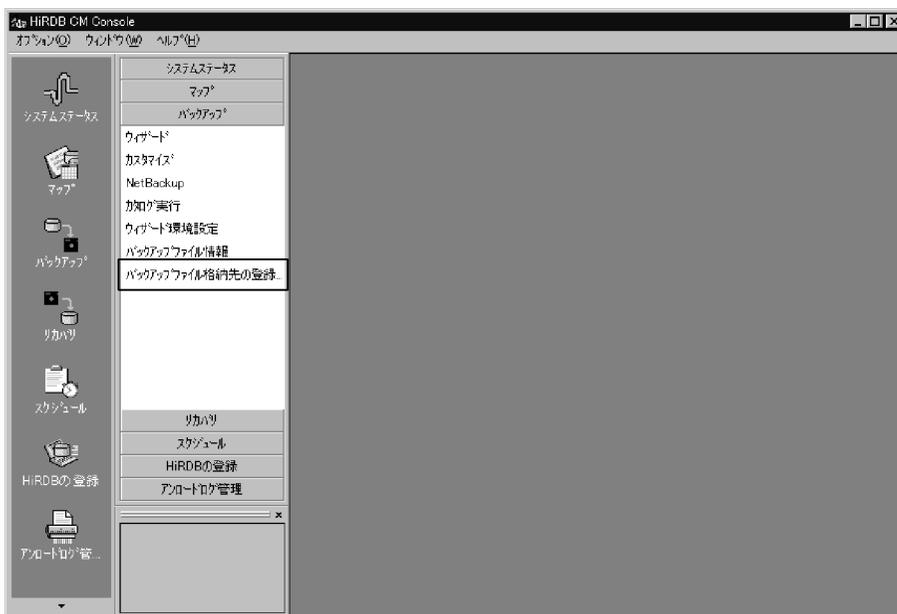
5.6 バックアップを取得する

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。

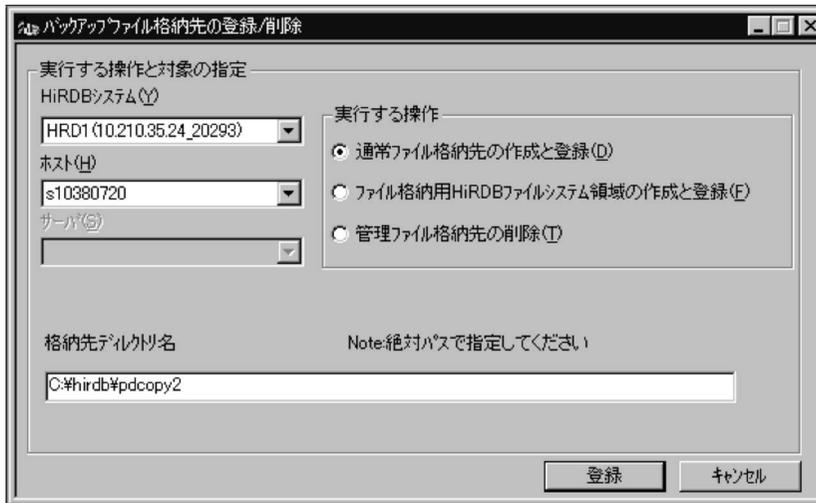
操作対象と操作内容を次のように指定してください。

HiRDB システム : HRD1 (10.210.35.24_20293)

ホスト : s10380720

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 : C:\¥hirdb¥pdcopy2



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)

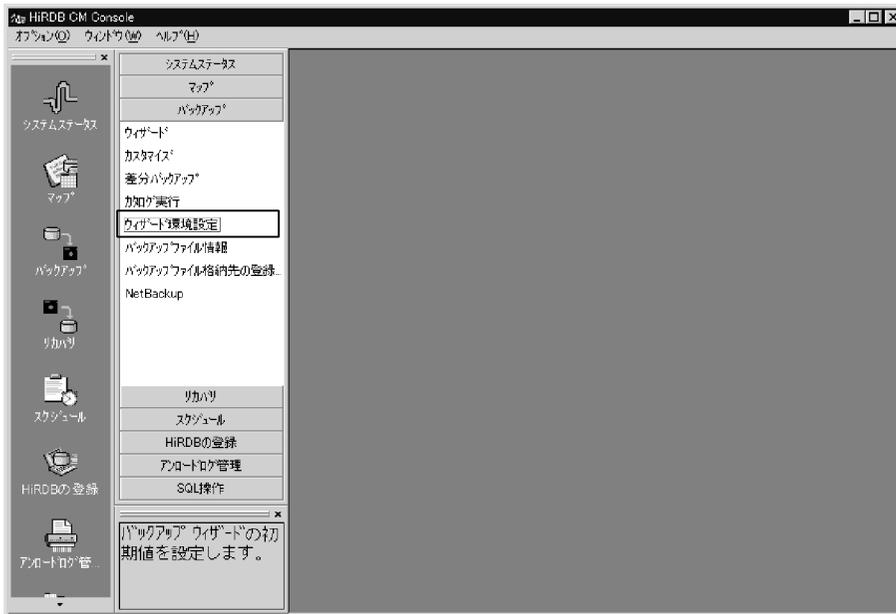
全 RD エリア (HiRDB システム) のバックアップを一括して取得します。バックアップ取得モードは r とします。

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：HRD1 (10.210.35.24_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)

HIRDB102103524_20298

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

バックアップファイル名(B)

追加(A)

削除(D)

ディスクエ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(J)

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)

入出力バックアップ回数(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(W)

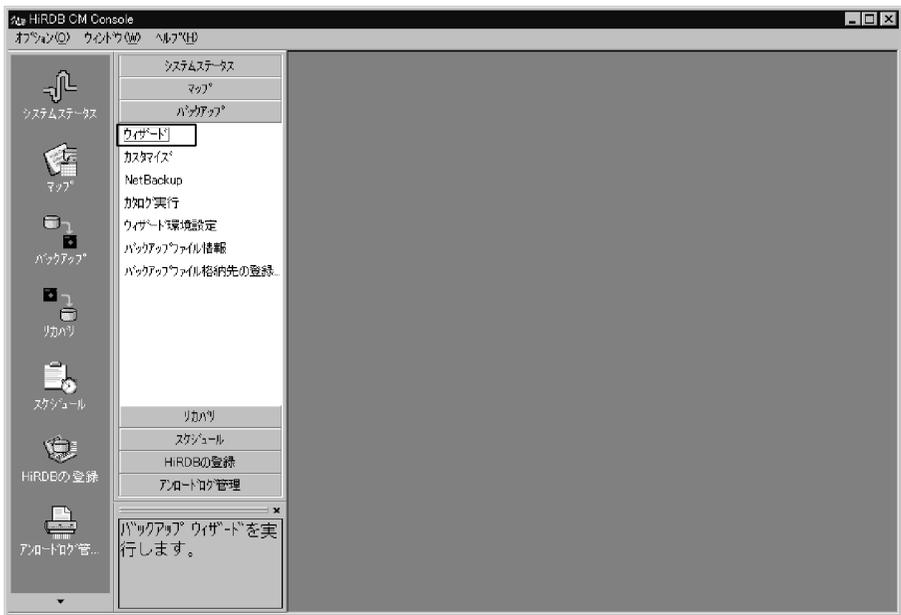
180 (0~200000)

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)

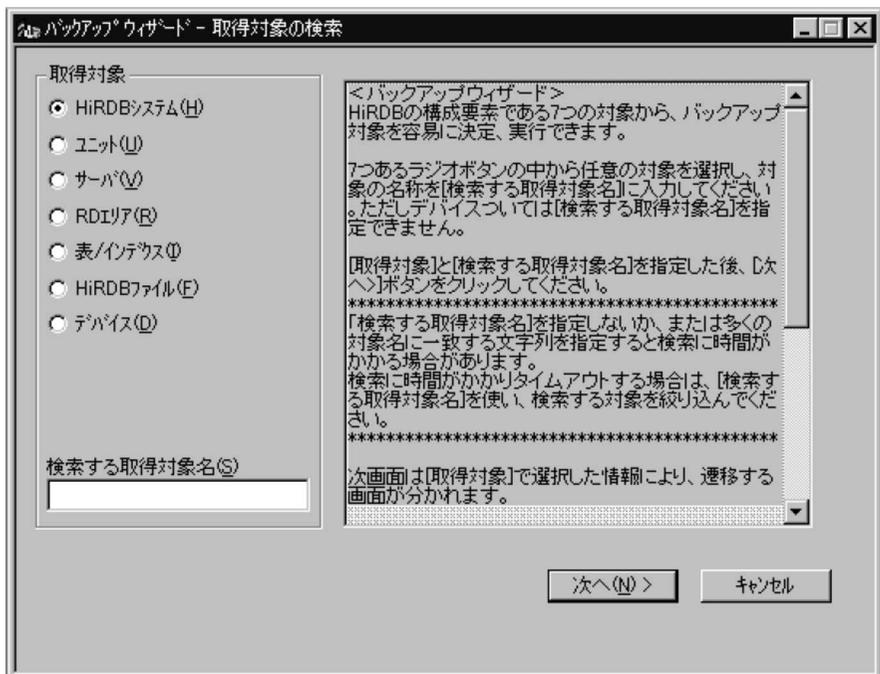
保存(S) キャンセル(Esc)

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

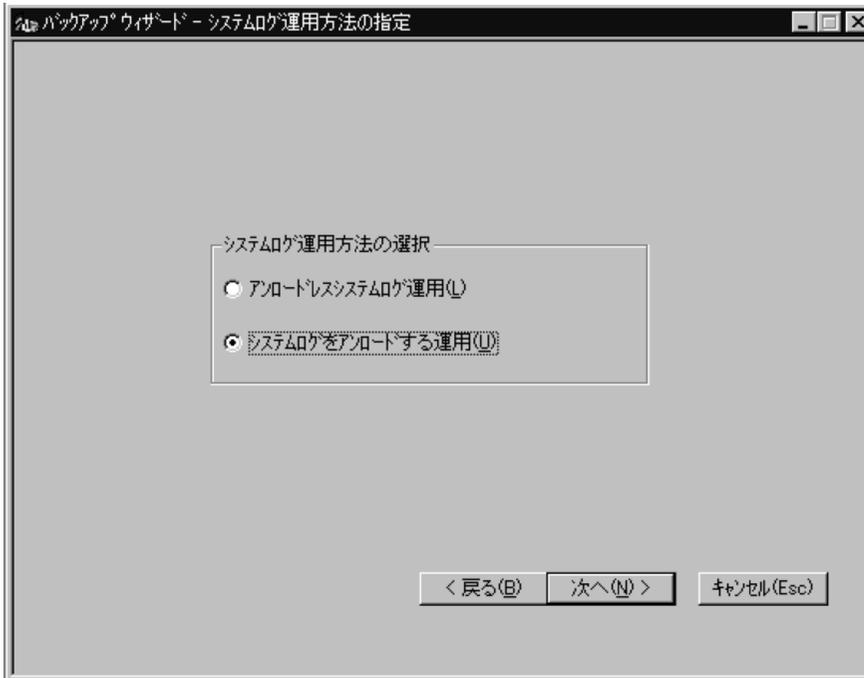
HiRDB システム：HRD1 (10.210.35.24_20293)



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

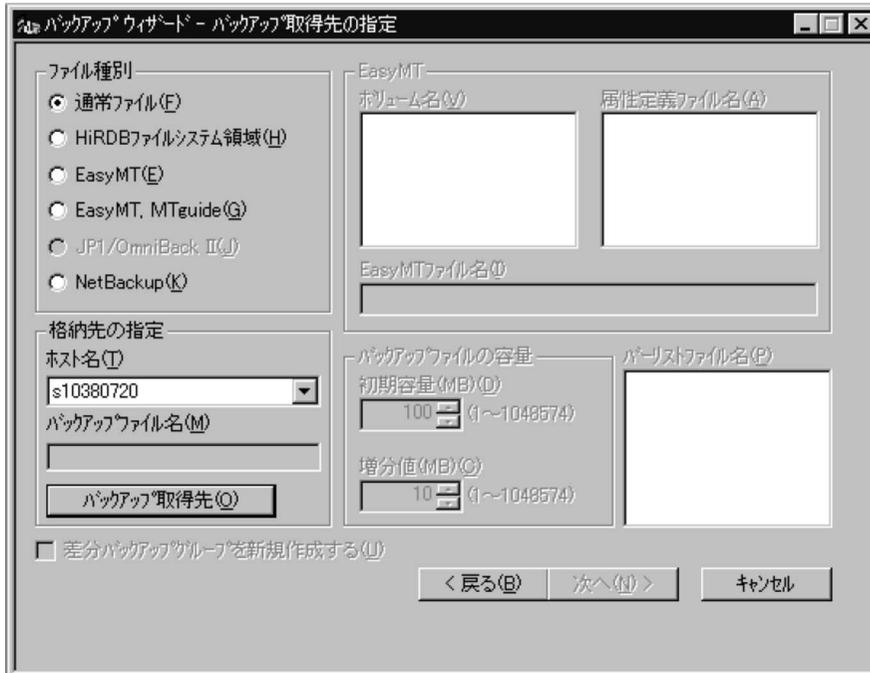


7. 【バックアップウィザード - バックアップ取得先の指定】ボックスが表示されます。

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。

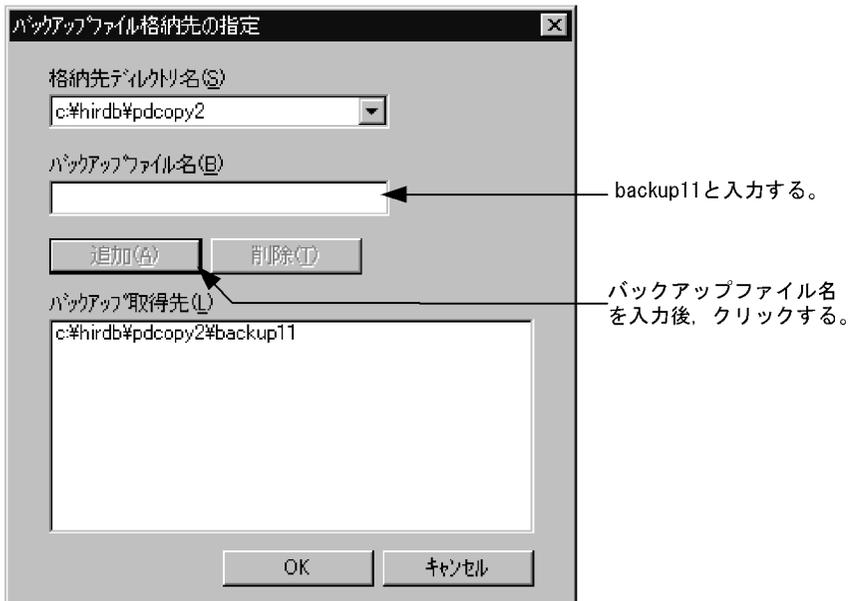
ファイル種別：通常ファイル

ホスト名：s10380720

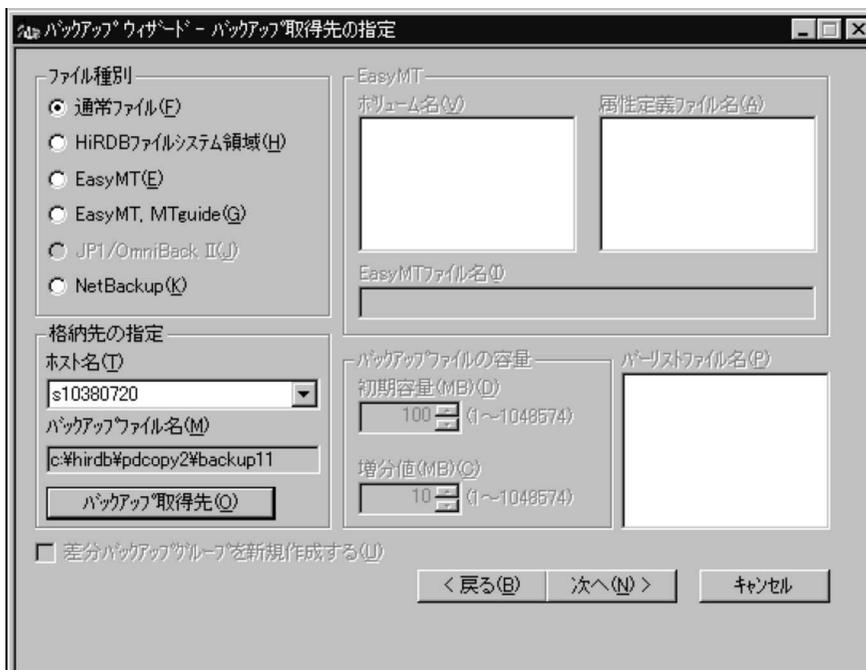


8. 【バックアップファイル格納先の指定】ダイアログが表示されます。
 バックアップ先の情報を次のように入力してから、「追加」ボタンをクリックしてください。バックアップ取得先に情報が表示されたら、「OK」ボタンをクリックしてください。
 格納先ディレクトリ名：C:\%hirdb%\pdcopy2
 バックアップファイル名：backup11

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。
「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い

い。バックアップ処理が始まります。



5.6.2 演習 2（自動ログアンロード機能を使っている場合に全 RD エリアのバックアップを取得する）

自動ログアンロード機能を使っている場合のバックアップの取得を行います。

バックアップ対象は、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）です。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 2 で使用するのので、保存しておいてください。

HiRDB Control Manager を使うと、自動ログアンロード機能を使ったデータベース回復では、システムログファイルは自動的にスワップされます。また、必要なアンロードログファイルは自動的に選択されます。したがって、バックアップを取得するときにアンロードログファイルを確認する必要はありません。

（1）自動ログアンロード機能が動作していることを確認します

自動ログアンロード機能が動作しているか確認します。次のコマンドを実行してください。

5. HiRDB Control Manager で HiRDB を運用してみましょう

```
C:¥>pdlogatul -d sys
```

〔説明〕

自動ログアンロード機能の動作状態を表示します。

(2) 自動ログアンロード機能の動作状態が表示されます

AUTO_LOG_UNLOAD の値が「ACTIVE」になっていることを確認してください。

```
HOSTNAME : s10380720(180219)
SERVER_NAME:sds01
AUTO_LOG_UNLOAD NOW_UNLOAD_LOG_GROUP CREATE_DIR
      ACTIVE          **** C:/win32app/hitachi/hirdb_s/area/unloadlo
g/
CURRENT LOG GENERATION INFO.
LOG_GROUP GEN_NO.  SERVER_RUN_ID RUN_ID  UNLOAD_FILE_NAME
log2      2 404eb711      404eb708 sds01_404eb7110002_log2
```

〔説明〕

ACTIVE :

自動ログアンロード機能の動作状態です。表示される値は次のとおりです。

ACTIVE : 動作中

STOP : 停止中

STOPPING : 停止時のアンロード処理待ち監視中

- : 自動ログアンロード機能を使用できない状態

(3) 全 RD エリア (HiRDB システム) のバックアップを取得します。

バックアップ方法については、演習 1 を参照してください。操作方法は演習 1 と同じです。ただし、バックアップファイル名を backup12 にしてください。

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。

簡易セットアップツールの「小規模」でセットアップした場合、これは次のコマンドになります。

```
pdfmkfs -r -n 45 -l 30 -e 1 -k SYS -i
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys011"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS -i
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys012"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS -i
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys013"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS -i
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys014"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS -i
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys015"

pdfmkfs -r -n 45 -l 30 -e 1 -k SYS -i
"C:¥win32app¥hitachi¥hirdb_s¥area¥rdsys016"
```

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。

5. HiRDB Control Manager で HiRDB を運用してみましょう

(3) RD エリアの回復に必要なファイルを準備します

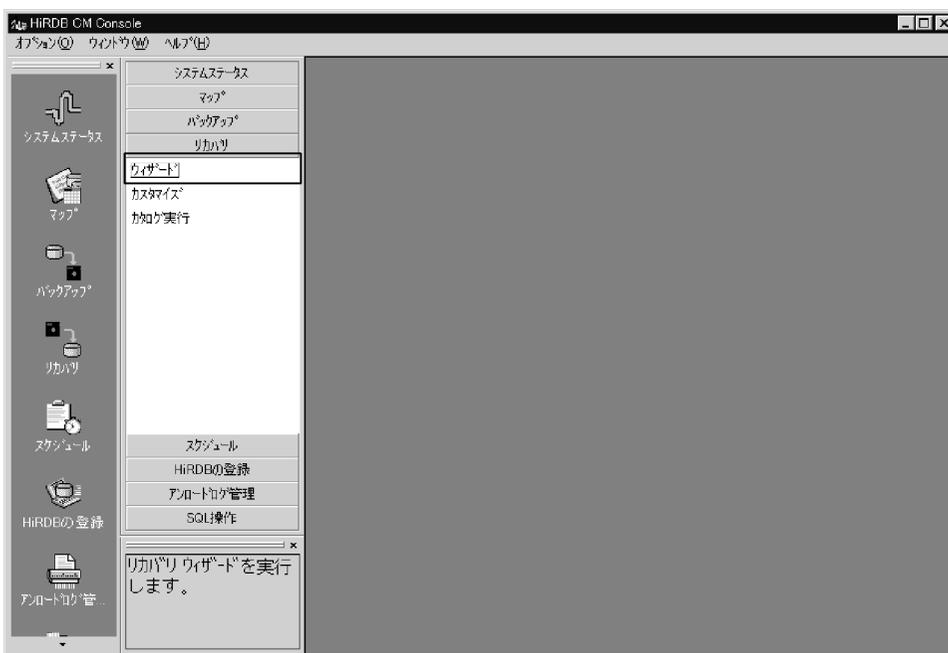
RD エリアの回復に必要なファイル（データベース回復ユーティリティの入力情報ファイル）は次のとおりです。

- 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す。バックアップファイル名は、C:\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：HRD1 (10.210.35.24_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 (自動ログアンロード機能を使っている場合に回復する)

自動ログアンロード機能を使っている場合のデータベースの回復を行います。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) を最新の状態に回復します。

全 RD エリアを最新の状態に回復する場合, 幾つかの RD エリアに分けて回復のための操作をする必要があります。詳細については, マニュアル「HiRDB Version 7 システム運用ガイド (Windows(R) 用)」を参照してください。

(1) 必要なバックアップファイルを特定します

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, C:\%hirdb%\pdcopy2\backup12 です。

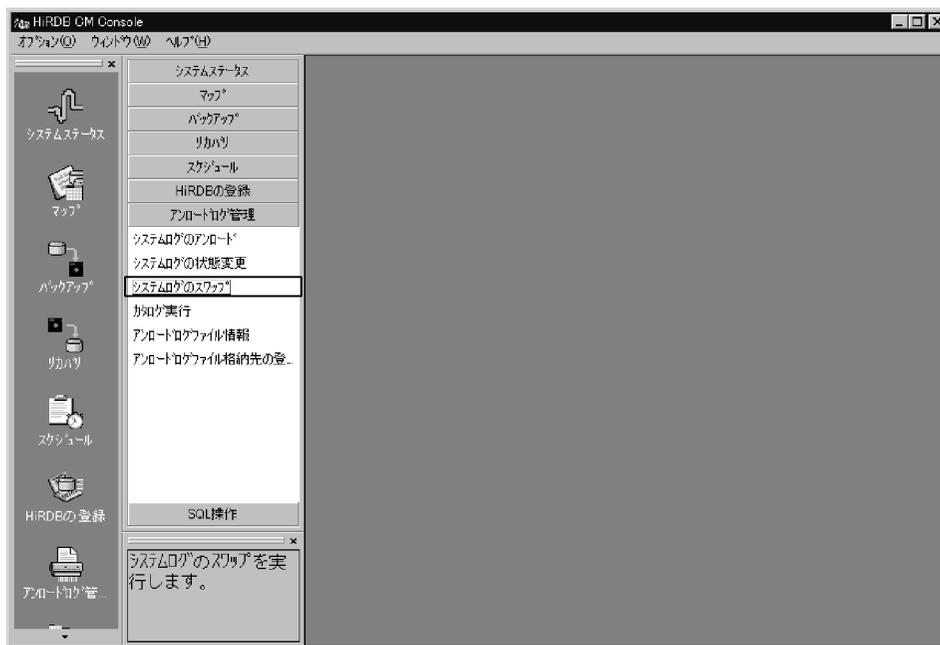
このバックアップを取得したとき以降に出力されたシステムログを格納したアンロードログファイルは, HiRDB Control Manager が自動で選択します。

(2) 現用のシステムログファイルをスワップします

現用のシステムログ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。



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：HRD1 (10.210.35.24_20293)

サーバ：sds01



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。

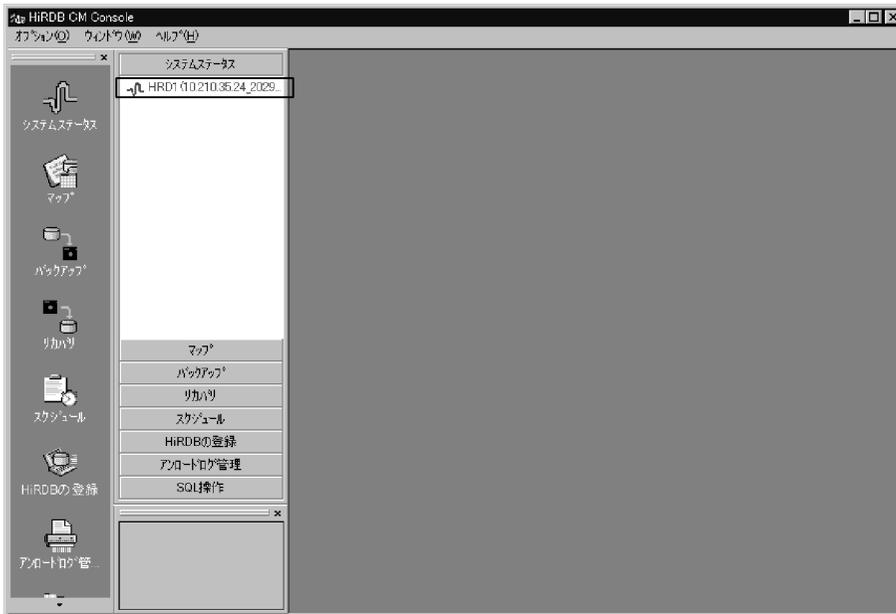
(3) システムログのアップロード処理が終了したことを確認します

スワップしたシステムログファイルがアップロードされたことを確認します。

手順

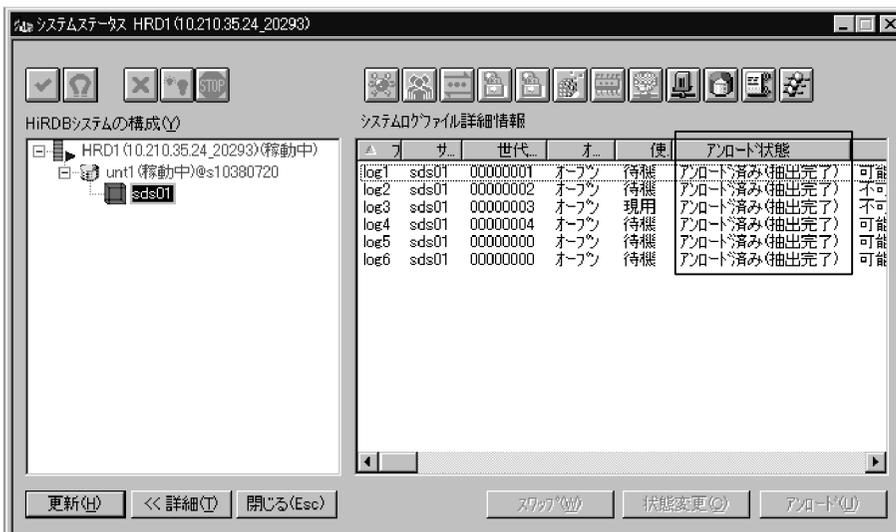
1. [システムステータス] - [HRD1 (10.210.35.24_20293)] を選択してください。

5. HIRDB Control Manager で HIRDB を運用してみましょう



2. 【システムステータス】ボックスが表示されます。

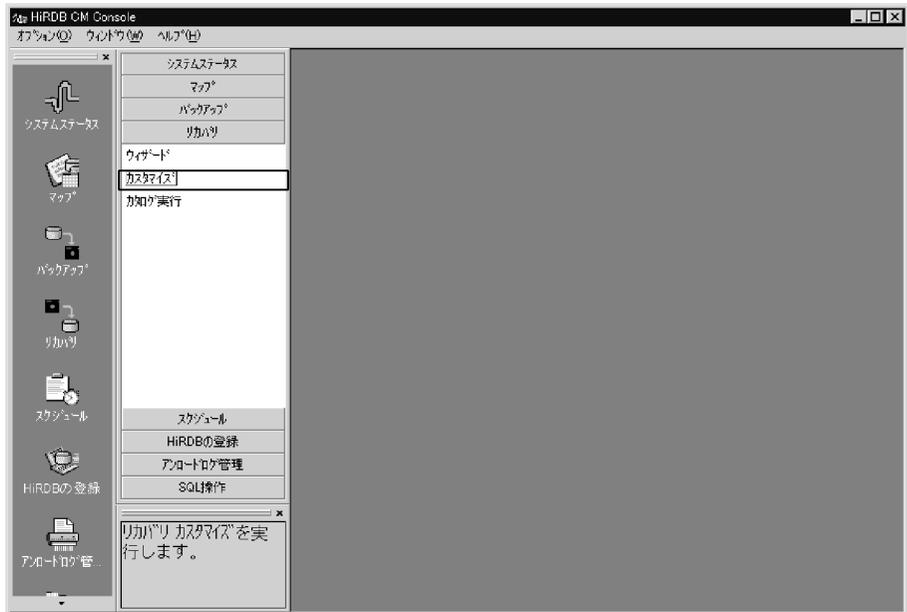
「sds01」の詳細情報から、「システムログファイル詳細」ボタンをクリックして、すべてのシステムログファイルが「アンロード済み」であることを確認してください。アンロード待ちのシステムログがある場合、少し時間をおいてから確認してください。



(4) マスタディレクトリ用 RD エリアを最新の状態に回復します

マスタディレクトリ用 RD エリアを回復します。

1. [リカバリ] - [カスタマイズ] を選択してください。



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：HRD1 (10.210.35.24_20293)

ユニット：unt1

サーバ：sds01

RD エリア：RDMAST

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

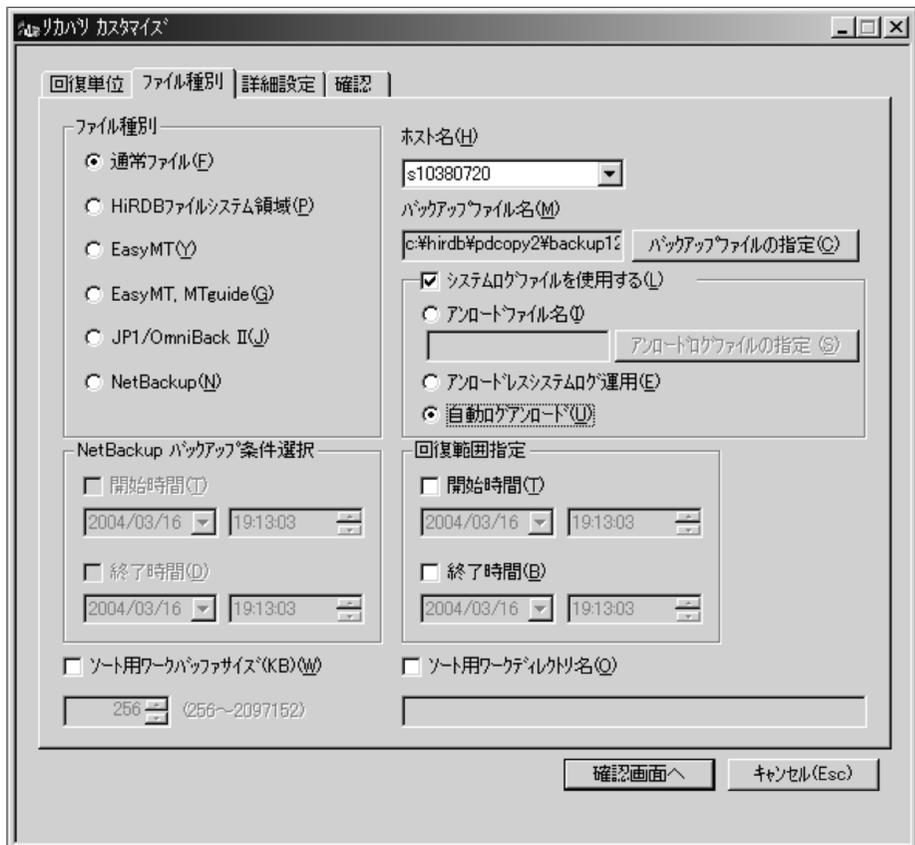
ホスト名：s10380720

バックアップファイル名：c:\¥hirdb¥pdcopy2¥backup12

システムログファイルを使用する：チェック

自動ログアンロード：チェック

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. マスタディレクトリ用 RD エリアが回復されます。

(5) データディレクトリ用 RD エリアなどをバックアップ取得時点に回復します

次に示す RD エリアをバックアップ取得時点に回復します。

- データディレクトリ用 RD エリア : RDDIRT
- データディクショナリ用 RD エリア : RDDICT
- データディクショナリ LOB 用 RD エリア (ソース格納用) : DIC_RTN_SRC

【リカバリ カスタマイズ】ボックスの各ページの指定を次のように変更して、データベースを回復してください。

回復単位

RD エリア : RDDIRT , RDDICT , DIC_RTN_SRC

ファイル種別

自動ログアンロード : チェックを外す

(6) マスタディレクトリ用 RD エリア以外の RD エリアを最新の状態に回復します

マスタディレクトリ用 RD エリア以外の RD エリアを回復します。

【リカバリ カスタマイズ】ボックスの各ページの指定を次のように変更して、データベースを回復してください。

回復単位

RD エリア：RDMAST 以外のすべて

ファイル種別

自動ログアンロード：チェック

(7) データディクショナリ LOB 用 RD エリア（オブジェクト格納用）を最新の状態に回復します

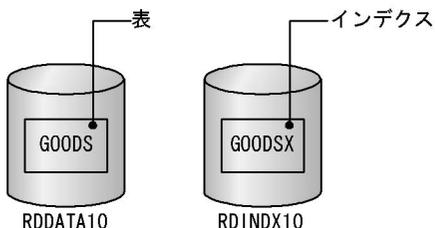
データディクショナリ LOB 用 RD エリア（オブジェクト格納用）を回復します。このマニュアルでは、ストアドプロシジャまたはストアドファンクションを登録していないので、実施する必要はありません。

5.8 表を再編成する

5.8.1 演習（表を再編成する）

GOODS 表を再編成します。

- GOODS 表にはインデクス（GOODSX）が定義されています。
- GOODS 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（GOODSX）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、C:¥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、C:¥work¥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 C:¥hirdb¥pdrorg¥seigyo01.txt に作成してください。

```
unload C:¥hirdb¥pdrorg¥unld01 .....1
idxwork C:¥work¥idxdir .....2
sort C:¥tmp .....3
```

〔説明〕

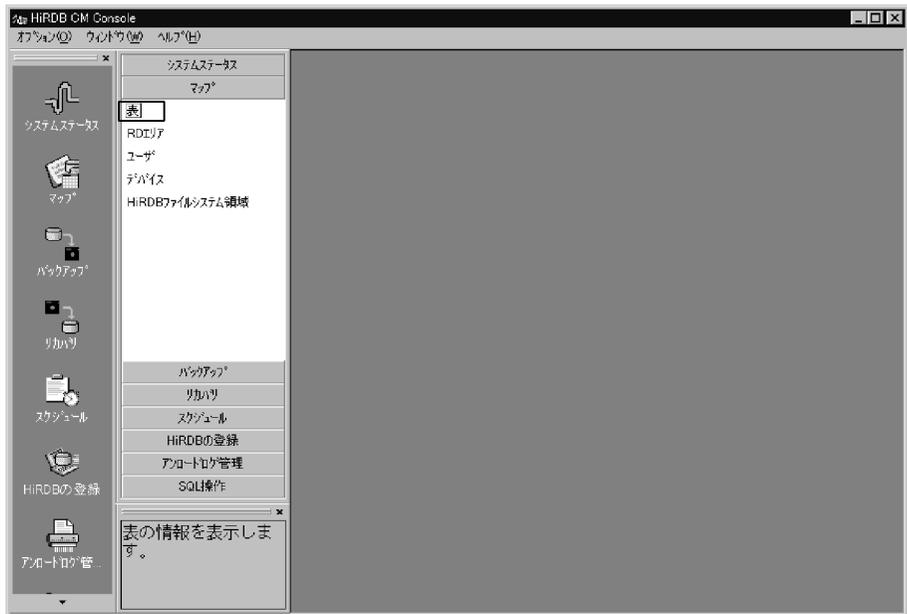
1. アンロードデータファイル名を指定します。
C:¥hirdb¥pdrorg ディレクトリ下に unld01 ファイルが作成されます。

- C:\¥hirdb¥pdorg 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。C:\¥work¥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) GOODS 表を再編成します

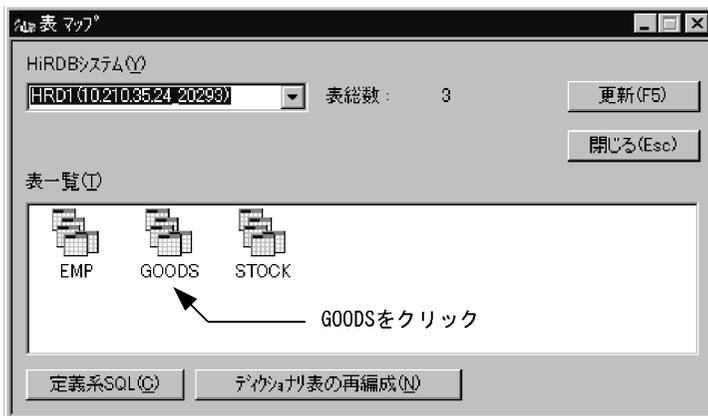
手順

1. [マップ] - [表] を選択してください。

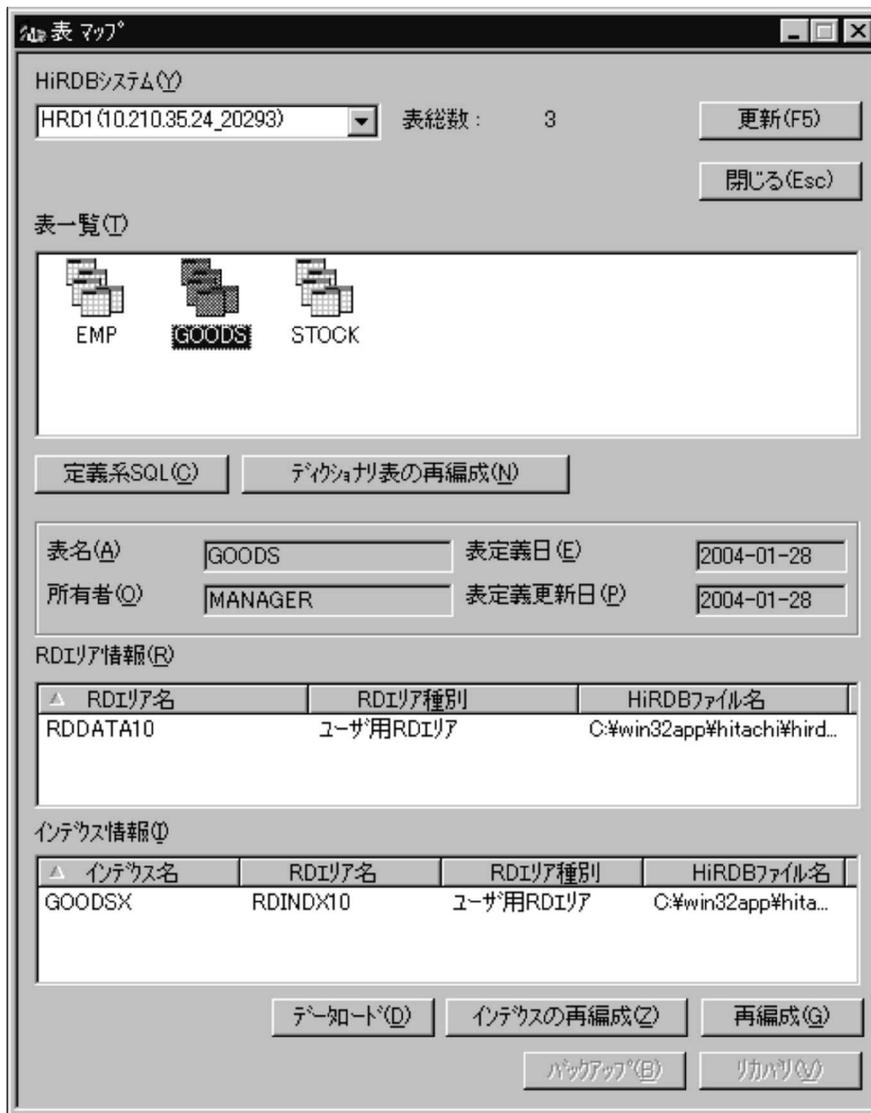


2. 【表 マップ】ボックスが表示されます。
HiRDB システムに「HRD1 (10.210.35.24_20293)」を選択したあと、表一覧の「GOODS」をクリックしてください。

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、GOODS 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：C:\¥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

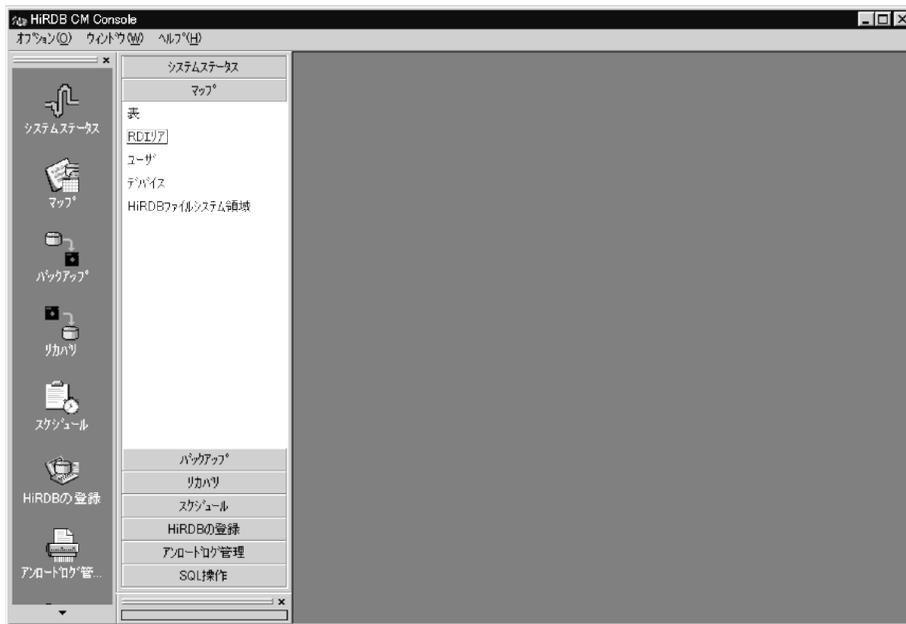
5.9.1 演習（RD エリアを拡張する）

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

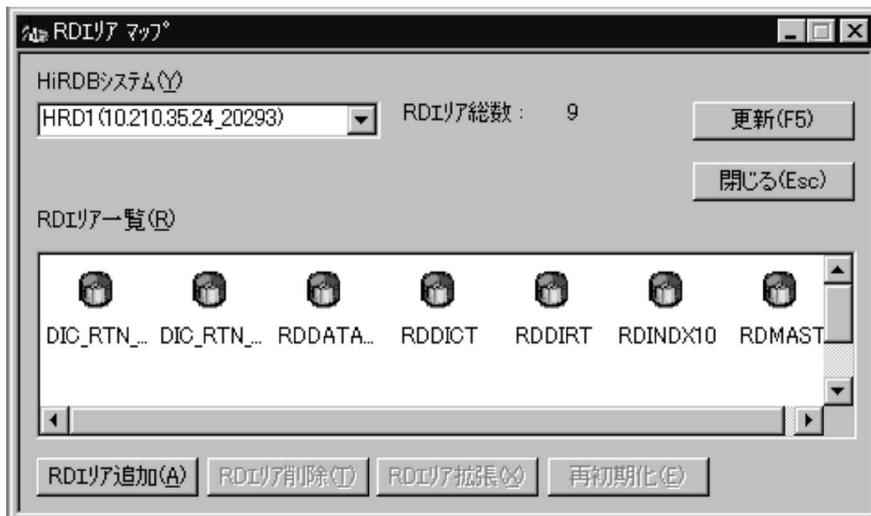
1. 【マップ】 - 【RD エリア】を選択してください。



2. 【RD エリア マップ】ボックスが表示されます。

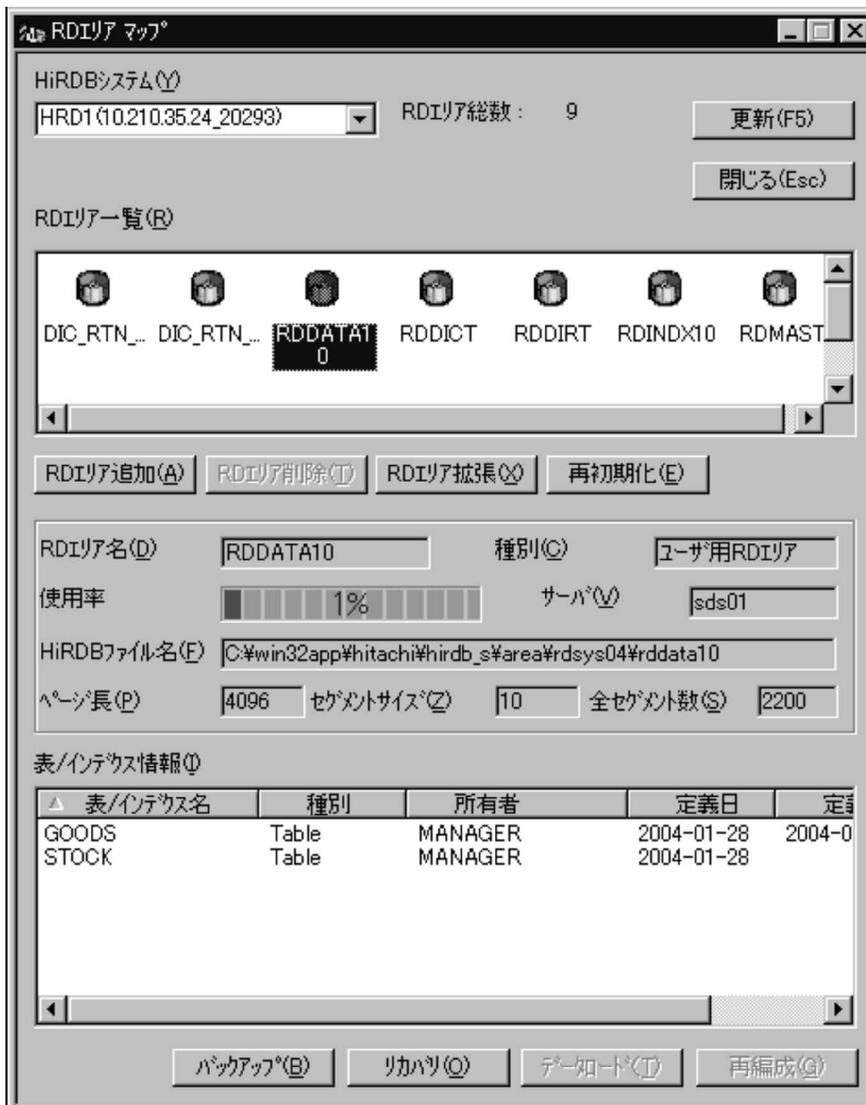
HiRDB システムに「HRD1 (10.210.35.24_20293)」を選択したあと、RD エリア一覧の「RDDATA10」をクリックしてください。

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



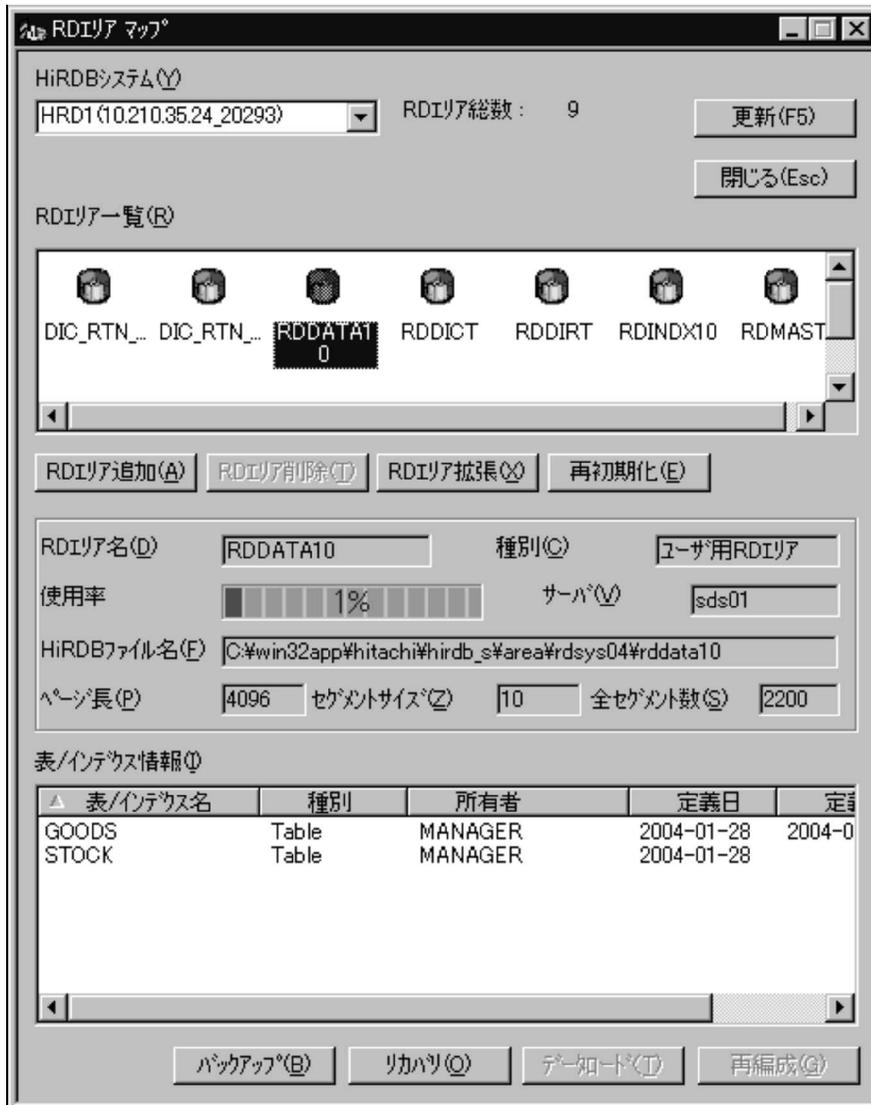
〔説明〕

この例では、全セグメント数が 2200 で、使用率が 1% 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：

C:\win32app\hitachi\hirdb_s\area\rdsys04

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。

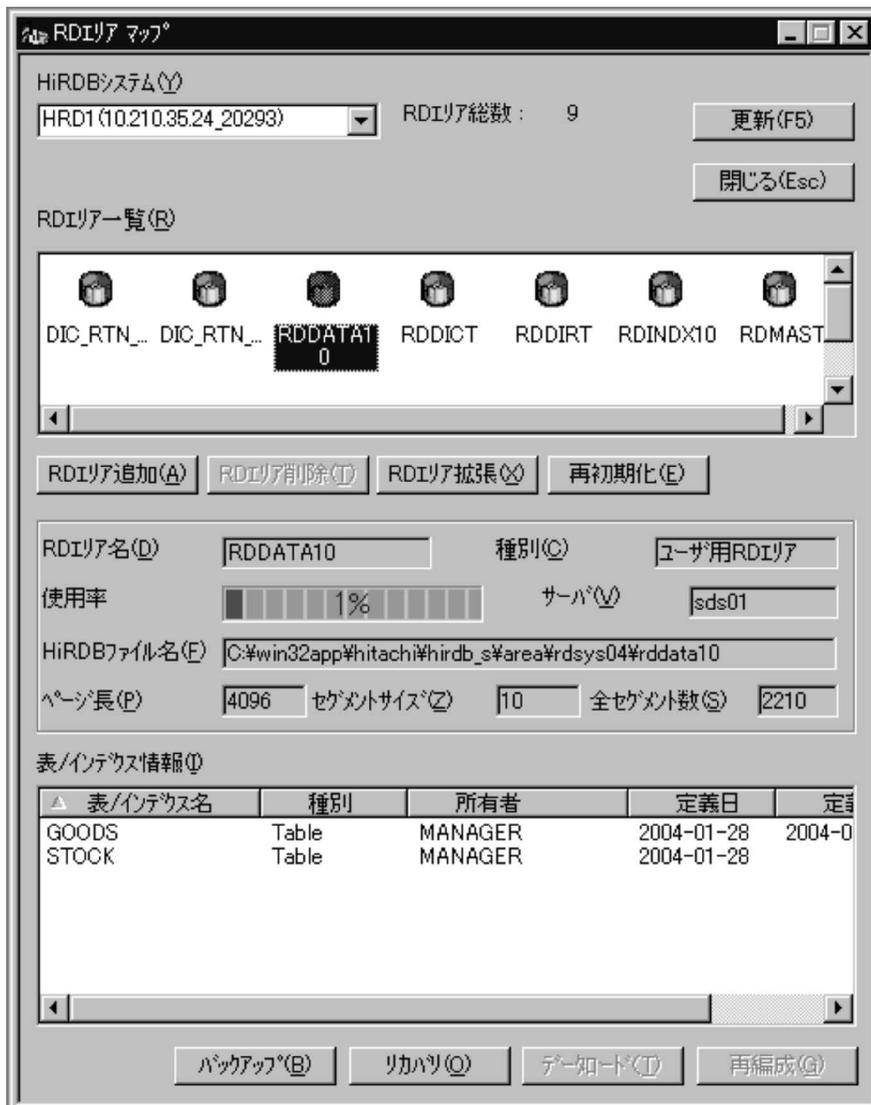
HiRDB ファイルシステム領域名が長くて「HiRDB ファイルシステム領域名」欄では確認できない場合、「追加」ボタンをクリックしてください。選択した HiRDB ファイルシステム領域名が「HiRDB ファイルシステム領域名 / HiRDB ファイル名」欄に表示されます。HiRDB ファイルシステム領域名に誤りがあれば、「削除」ボタンで削除してから選択し直してください。

- 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、2200 から 2210 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

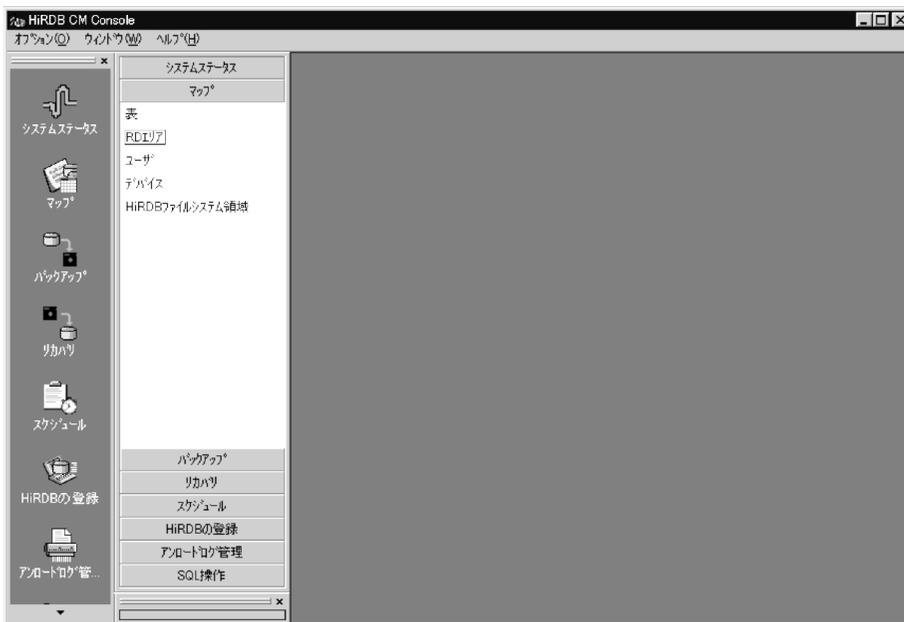
5.10.1 演習（RD エリアを追加する）

新しい HiRDB ファイルシステム領域（rdsys10）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

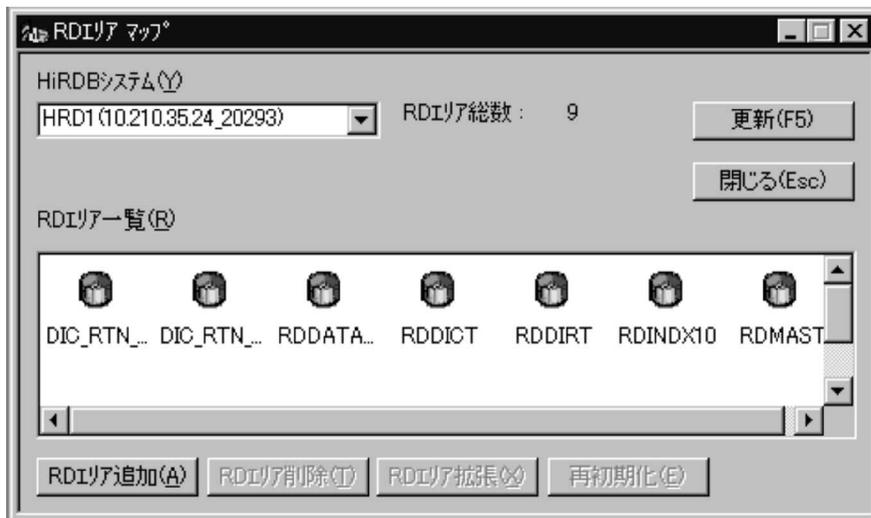
1. [マップ] - [RD エリア] を選択してください。



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「HRD1 (10.210.35.24_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。

5. HiRDB Control Manager で HiRDB を運用してみましょう



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 : sds01

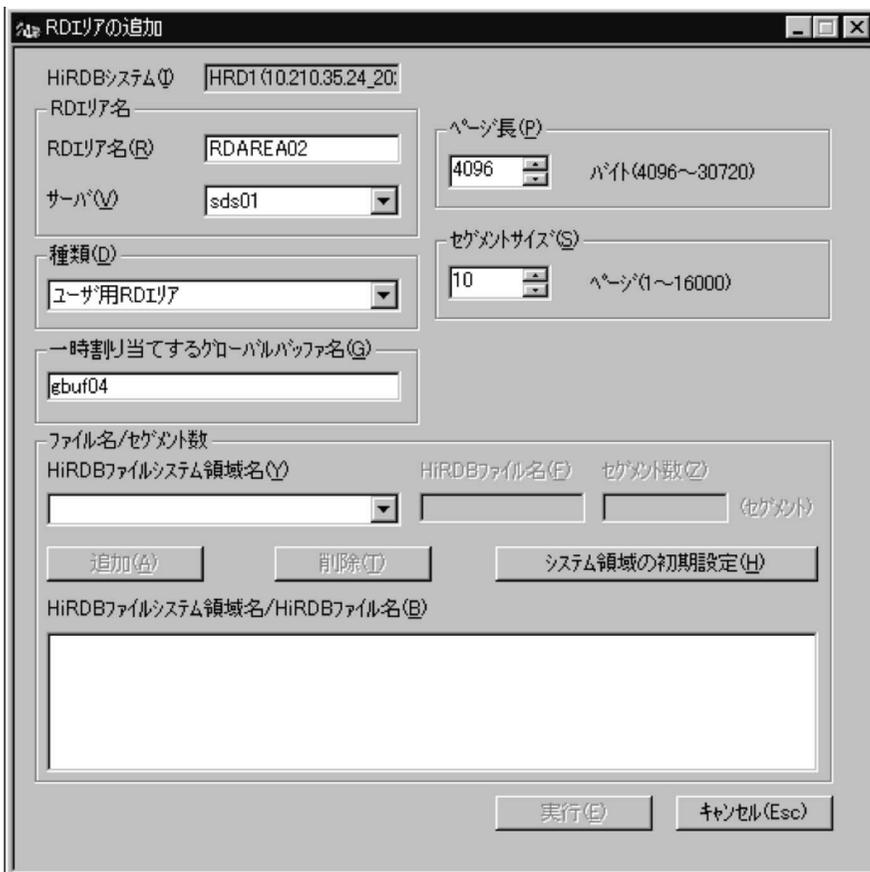
種類 : ユーザ用 RD エリア

一時割り当てするグローバルバッファ名 : gbuf04

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「rdsys10」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「rdsys10」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：

C:\win32app\hitachi\hirdb_s\area\rdsys10

HiRDB ファイルシステム領域サイズ：10

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

〔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：

C:\¥win32app¥hitachi¥hirdb_s¥area¥rdsys10

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

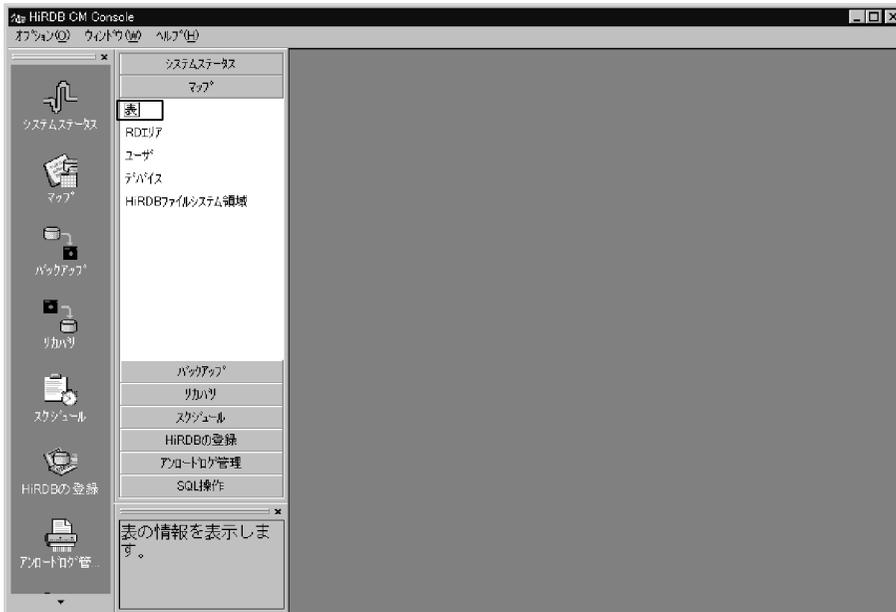
手順

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 : C:\%hirdb%\pdef%c_table2.txt

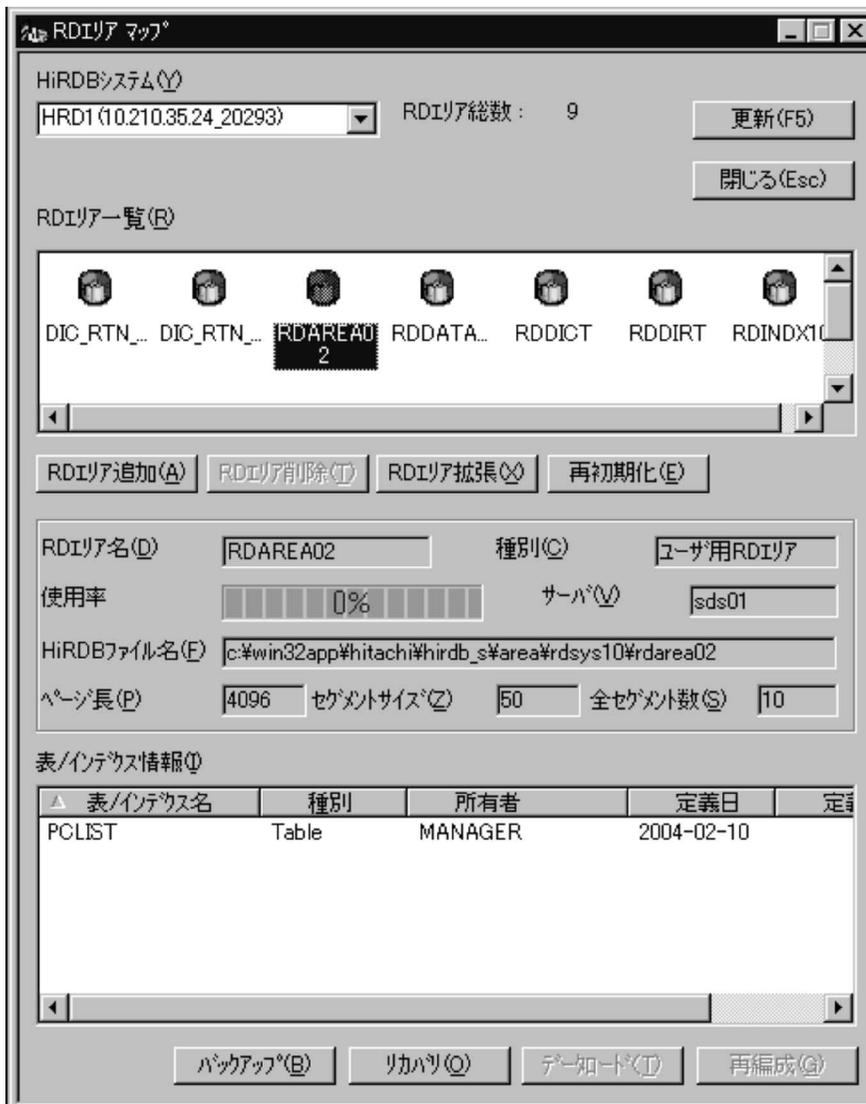


5. SQL の実行結果を示すメッセージボックスが表示されます。「閉じる」ボタンをクリックしてください。
6. 「サーバの管理情報を更新しますか？」というメッセージが表示されます。「はい」ボタンをクリックしてください。

(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：「HRD1 (10.210.35.24_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', 'ヒタチタロウ', 1995);  
INSERT INTO PCLIST VALUES('PC0011', 'ヒタチハナコ', 2000);  
INSERT INTO PCLIST VALUES('PC0012', 'ヒタチイロウ', 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	ヒタチタロウ	1995
PC0011	ヒタチハナコ	2000
PC0012	ヒタチイロウ	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「HRD1 (10.210.35.24_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】ボックスにが表示され、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
 - 6.4 ODBC 対応のアプリケーションプログラムからアクセスする
-

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



注※ 実行するかしないかは任意です。なお、ODBC対応のアプリケーションプログラムとしてMicrosoft Accessを使用しています。

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

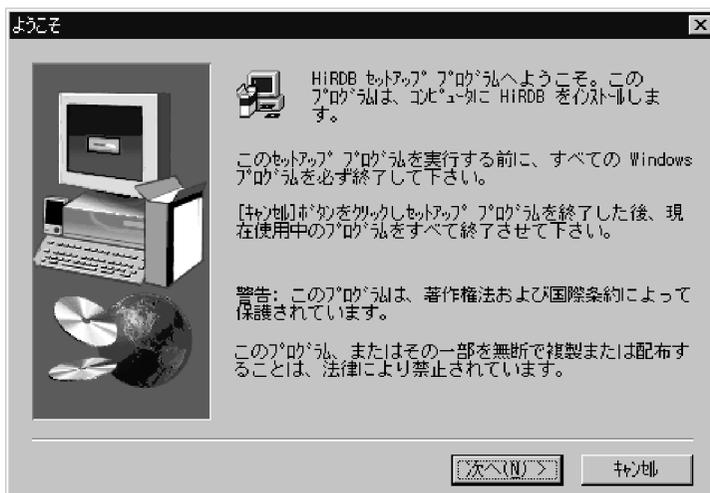
HiRDB/ クライアント (HiRDB/Developer's Kit) をインストールします。HiRDB/ シングルサーバをインストールしたマシンではない、別のマシンにインストールします。「7. アプリケーションを作成してみましょう」を実施しないときは、インストールする HiRDB/ クライアントは HiRDB/Run Time でもかまいません。

なお、HiRDB/ クライアントが動作する OS は Windows 2000 を仮定します。

インストール手順を次に示します。

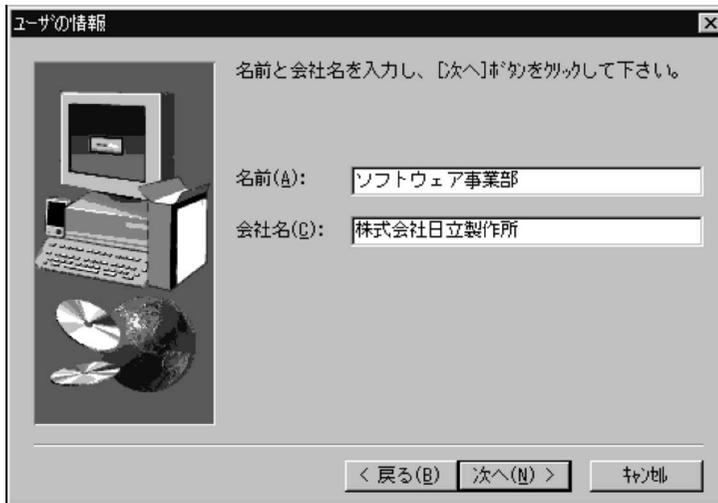
インストール手順

1. エクスプローラを使用して、CD-ROM の「02_LIB32\DISK1」の「SETUP.EXE」を実行してください。インストーラが起動し、次の画面が表示されます。「次へ」をクリックしてください。



2. ユーザの情報として、名前と会社名を入力してから、「次へ」をクリックしてください。

6. クライアントを接続してみましょう



3. インストール先ディレクトリ名を指定してください。既定値で問題なければ、そのまま「次へ」をクリックしてください。



4. セットアップ方法を選択し、「次へ」をクリックしてください。



5. 設定した内容を確認し、問題がなければ「次へ」をクリックしてください。インストールが始まります。



6. インストールが終了すると、次の画面が表示されます。「完了」をクリックしてください。

6. クライアントを接続してみましょう



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 を使用します。

HiRDB SQL Executer のインストール手順を次に示します。

1. CD-ROM を挿入して、「09_pdsq1¥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。

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、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)

```
10.210.35.24      s10380720      #HiRDBサーバ
10.210.35.25      TIPS02         #HiRDB/クライアント
:
```

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。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=s10380720
PDNAMEPORT=22200
PDUSER=USER01/PASS01
```

[説明]

```
PDHOST=s10380720
```

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=22200
```

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

注

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

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

6. クライアントを接続してみましょう

開発ガイド (Windows (R) 用)」を参照してください。

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 GOODS 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 (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。

- 商品コード (CHAR(4))
- 商品名 (NCHAR(8))
- 色 (NCHAR(2))
- 単価 (INTEGER)
- 在庫量 (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  
(商品コード CHAR(4), 商品名 NCHAR(8), 色 NCHAR(2),  
 単価 INTEGER, 在庫量 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 20040205184117
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', N'長靴', N'黒', 10000, 3

【SQL文】

```
INSERT INTO ZAIKO VALUES('777Z',N'長靴',N'黒',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。

```
商品コード 商品名          色  単価      在庫量
-----
777Z        長靴                黒    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 GOODS 表を検索します。
HK で始まるショウヒン CD, およびタンカが 20000 円以下の行を検索します。

【SQL文】

```
SELECT * FROM MANAGER.GOODS
WHERE ショウビンCD LIKE 'HK%' AND タンカ<=20000;
```

【実行結果】

ショウビンCD	ショウビンメイ	タンカ	シレサキCD
HK0006	デンキカーペット(3ショウヨウ)	0012600.	DE036
HK0007	デンキカーペット(4.5ショウヨウ)	0015600.	DE037

KFPX27010-I 2 rows selected

6.4 ODBC 対応のアプリケーションプログラムからアクセスする

ODBC 対応のアプリケーションプログラムから HiRDB のデータベースをアクセスする場合、ODBC ドライバをインストールする必要があります。

ここでは、ODBC 対応のアプリケーションプログラムとして、Microsoft Access を使用する例について説明します。なお、Microsoft Access は、HiRDB/ クライアントをインストールしたマシンにあるものと仮定します。

6.4.1 ODBC ドライバをインストールする

HiRDB/ クライアント (HiRDB/Developer's Kit) をインストールしているマシンに、ODBC ドライバをインストールします。

インストール手順

1. エクスプローラを使用して、CD-ROM のディレクトリ「05_odb32¥DISK1」の「SETUP.EXE」を実行してください。すると、次の画面が表示されるので、「継続」をクリックしてください。



2. 【ドライバのセットアップ】ダイアログの「HiRDB 32bit Driver」を選択し、「OK」をクリックしてください。



3. ファイルのコピーが始まります。
ファイルのコピーが終了すると、【データソース】ダイアログが表示されます。

【データソース】ダイアログの「追加」をクリックしてください。



4. 【データソースの追加】ダイアログが表示されるので、「HiRDB 32bit Driver」を選択し、「OK」をクリックしてください。



5. 【HiRDB Driver セットアップ】ダイアログが表示されるので、次の各項目に情報を入力して、「OK」をクリックしてください。

- ・ データソース名：HiRDBDatabase ... 任意のデータソース名を指定します。
- ・ PDHOST：s10380720 ... アクセスする HiRDB の、システム共通定義 (pdsys) の pdstart オペランドの -x に指定したホスト名を指定します。
- ・ PDNAMEPORT：22200 ... アクセスする HiRDB の、システム共通定義 (pdsys) の pd_name_port オペランドに指定したポート番号を指定します。

6. クライアントを接続してみましょう



6. 【データソース】ダイアログのデータソース名に、「HiRDBDatabase (HiRDB 32bit Driver)」が追加されます。
7. 「閉じる」をクリックしてください。
8. "セットアップが完了しました。" というメッセージが表示されます。「OK」をクリックしてください。

6.4.2 Microsoft Access から HiRDB をアクセスする

Microsoft Access から HiRDB の表をアクセスする方法について説明します。

手順

1. Microsoft Access を起動します。
2. 【新しいデータベースの作成】ダイアログが表示されます。
「データベースの新規作成」を選択して、ファイル名を付けてください。
3. [ファイル] - [開く] を選択し、任意のファイル名（ここでは、例として HiRDBTables.mdb を入力します）を指定し、「開く」をクリックしてください。
4. 【HiRDBTables : データベース】ダイアログが表示されます。
[ファイル] - [外部データの取り込み] - [テーブルのリンク] を選択してください。
5. 【リンク】ダイアログが表示されます。
「ファイルの種類」から「ODBC データベース」を選択してください。
6. 【データソースの選択】ダイアログが表示されます。
「コンピュータ データソース」からデータソース名「HiRDBDatabase」を選択し、「OK」をクリックしてください。
7. 【HiRDB ログイン】ダイアログが表示されます。
[ユーザ ID] に「USER01」、[パスワード] に「PASS01」を入力し、「OK」をクリックしてください。



HiRDB ログイン

データソース : HiRDBDatabase

ユーザID : USER01

パスワード : *****

OK キャンセル

8. 【テーブルのリンク】ダイアログのテーブルに、HiRDBの表の一覧が表示されます。アクセスしたい表を選択し、「OK」をクリックしてください。
9. 【HiRDBTables : データベース】ダイアログの「テーブル」に、選択した表が表示されます。その表を選択して「OK」をクリックしてください。
【固有レコード識別子の選択】ダイアログが表示されます。
10. 「MANAGER_GOODS」テーブルのフィールドから、フィールドを選択して「OK」をクリックしてください。
11. 「テーブル」に、選択した表が追加されます。
あとは、Microsoft Access で行の追加、変更、削除などをしてください。

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 (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 (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

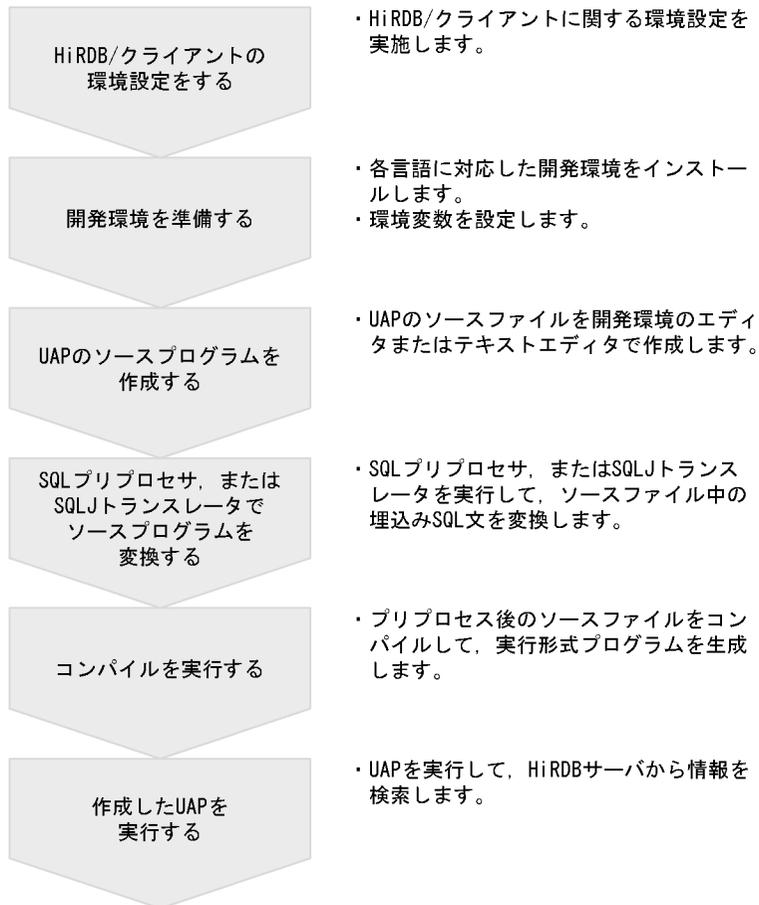
ここでは、HiRDB/クライアントで UAP を開発します。「6. クライアントを接続してみ

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

言語	開発環境
C 言語 / C++ 言語	Microsoft Visual C++ (2.0 以降)
COBOL 言語	COBOL85 Version 3 以降 ¹ ² , COBOL 2002
OOCOBOL 言語	OOCOBOL (Version 3 以降) ²
Java 言語	JDK 1.3.0-01 以降

注 1

UAP を DLL 化する場合は、Version 4.0 (04-02 以降) が必要です。

注 2

使用する OS によって、必要なバージョンが異なります。

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。

- Microsoft Visual C++ Version 6.0 (C 言語の場合)
- COBOL2002 NetClient Suite (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 (Character User Interface) 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

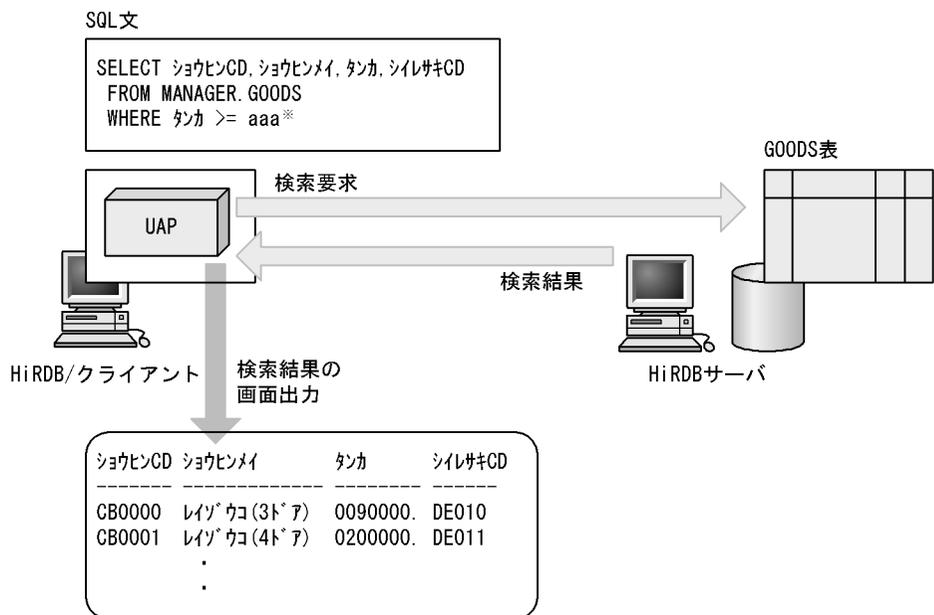
HiRDB サーバに接続して、データベースの情報を検索する UAP を作成してみましょう。具体的には、次の機能を持たせます。

- 「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、GOODS 表の内容を検索します。このとき、コマンドライン引数で指定された価格以上の商品だけ取り出すように、検索条件を指定します。
- 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 実行者からの入力は、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。

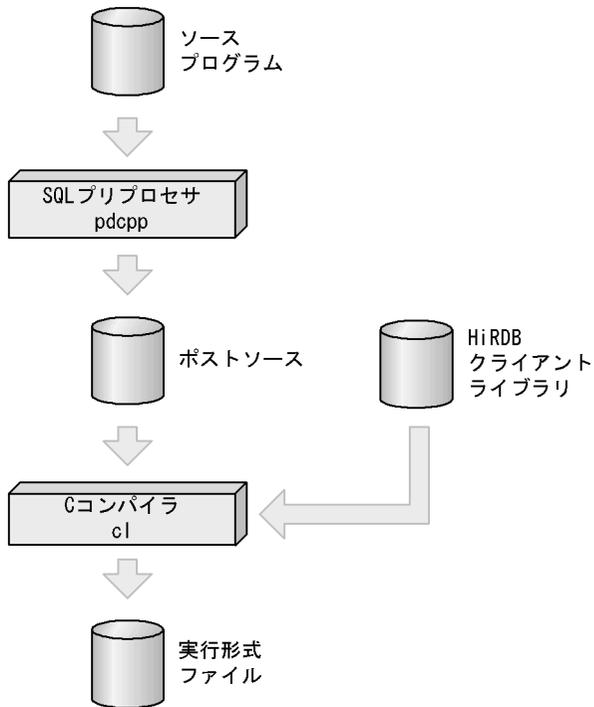
7.3 C 言語でアプリケーションを作成する

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

C:\¥CSAMPLE¥csample.ec

ポストソース

C:\¥CSAMPLE¥csample.c

実行形式ファイル

C:\¥CSAMPLE¥csample.exe

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。

- Microsoft Visual C++ (2.0以降)

(1) Microsoft Visual C++ をインストール

Microsoft Visual C++ 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

Microsoft Visual C++ のコンパイラでHiRDBのライブラリを使用できるように、環境変数を設定します。C:\¥ディレクトリ下のAUTOEXEC.BATファイルに次の行を追加してください。

```
set INCLUDE=%INCLUDE%;C:\¥PROGRA~1\¥HITACHI\¥HIRDB\¥INCLUDE
```

追加したら、Windows 2000 を再起動してください。

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。エクスプローラで次のディレクトリを作成してください。

```
C:\¥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。

C:¥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

```
/* **** */
/* [概要] */
/* HiRDBファーストステップガイドのサンプルプログラムです。 */
/* コマンドライン引数で指定された価格以上のレコードを */
/* MANAGER.GOODS表から検索します。 */
/* **** */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void connectError();
void abnormalEnd();

int main( int argc, char **argv )
{
/* 埋込み変数及び標識変数の宣言 **** */
EXEC SQL BEGIN DECLARE SECTION;
char xUserId[9]; /* ユーザID */
char xPswd[9]; /* パスワード */
char gSCode[7]; /* 商品コード */
char gSName[31]; /* 商品名 */
int gTanka; /* 単価 */
char gShiire[6]; /* 仕入先コード */
int aTanka; /* 検索条件 */
EXEC SQL END DECLARE SECTION;

/* 開始のメッセージの出力とコマンドライン引数の読み取り **** */
if( argc != 2 ){
aTanka = 0;
}else{
aTanka = atoi(argv[1]);
}

printf("処理を開始します。 %n");

/* エラー時の処置の指定 (HiRDB接続前) **** */
EXEC SQL WHENEVER SQLERROR PERFORM connectError;

/* HiRDBとの接続 **** */
printf("USERID:");
scanf("%s",&xUserId);
printf("PASSWORD:");
scanf("%s",&xPswd);
EXEC SQL CONNECT USER :xUserId USING :xPswd;
printf("HiRDBとの接続に成功しました。 %n");

/* エラー時の処置の指定 (HiRDB接続後) **** */
EXEC SQL WHENEVER SQLERROR PERFORM abnormalEnd;

/* カーソルの宣言とオープン **** */
EXEC SQL DECLARE CR1 CURSOR FOR
SELECT ショウヒンCD, ショウヒンメイ, INTEGER(タンカ), シイサキCD
FROM MANAGER.GOODS WHERE INTEGER(タンカ) >= aTanka;
EXEC SQL OPEN CR1;
SQLCODE=0;
```

```

/* 検索結果の出力 *****/
printf(" ショウヒンCD ショウヒンメイ          タンカ   シレサキCD¥n");
printf(" -----¥n");
while(SQLCODE>=0) {
    EXEC SQL WHENEVER NOT FOUND GO TO OWARI;
    EXEC SQL FETCH CR1 INTO :gSCode, :gSName, :gTanka, :gShiire;
    EXEC SQL WHENEVER NOT FOUND CONTINUE;

    printf(" %6s %30s %7d %6s¥n", gSCode, gSName, gTanka, gShiire);
}

OWARI:
/* カーソルのクローズ *****/
EXEC SQL CLOSE CR1;
EXEC SQL COMMIT;

/* サーバとの切り離し *****/
EXEC SQL WHENEVER SQLERROR CONTINUE;
EXEC SQL WHENEVER NOT FOUND CONTINUE;
EXEC SQL DISCONNECT;
printf("HiRDBから切り離しました。¥n");

/* 終了処理 *****/
printf("処理を終了します。¥n");
return(0);
}

/* HiRDBに接続できなかった場合の処置 *****/
void connectError(void)
{
    printf("*** HiRDBとの接続に失敗しました。 ***¥n");
    printf("KFSG001-E : incorrect UserId/Password¥n");
    EXEC SQL DISCONNECT;
    exit(2);
}

/* エラー発生時の処置 *****/
void abnormalEnd()
{
    int wsqlcode;

    wsqlcode = (-1)*SQLCODE;
    printf("¥n*** SQLでエラーが発生しました。 ¥n");
    printf(" MESSAGE-ID = KFPAl1%3d-E¥n", wsqlcode);
    printf(" SQLERRMC = %s¥n", SQLERRMC);
    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】

7. アプリケーションを作成してみましょう

```
C:¥>pdccpp C:¥CSAMPLE¥csample.ec /o C:¥CSAMPLE¥csample.c
```

【説明】

C:¥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 C:¥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、C:¥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 Microsoft Visual C++ 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】

```
C:¥>cl C:¥CSAMPLE¥csample.c "C:¥Program Files¥Hitachi¥HiRDB¥lib¥Cltdll.lib" /MT
```

【説明】

C:¥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。

"C:¥Program Files¥Hitachi¥HiRDB¥lib¥Cltdll.lib"

インポートライブラリを指定します。

/MT

Microsoft Visual C++ のマルチスレッドライブラリの使用を指定します。

コマンドが正常に終了すると、C:¥CSAMPLE ディレクトリ下に csample.exe 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、Microsoft Visual C++ のインクルードパスや LIB パス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。設定方法については、Microsoft Visual C++ のマニュアルを参照してください。

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 にはコマンド引数として価格を指定できます。ここでは、価格が 500000 以上の商品を GOODS 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】

```
C:¥>C:¥CSAMPLE¥csample 500000
```

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】

```
処理を開始します。
```

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との接続に成功しました。
シヨウヒンCD シヨウヒンメイ          タカ          シレサキCD
-----
CP0001 コピーキ(カラー)              0620000. DE031
PR0000 プロジェクター-150インチ(キョウムヨウ) 1000000. DE010
PR0009 プロジェクター-100インチ          0500000. DE009
TV0007 テレビ-34インチ(ハイビジョン)      0500000. DE007
VD0010 ビデオ-PRO(キョウムヨウ)          0500000. DE020
HiRDBから切り離しました。
処理を終了します。
```

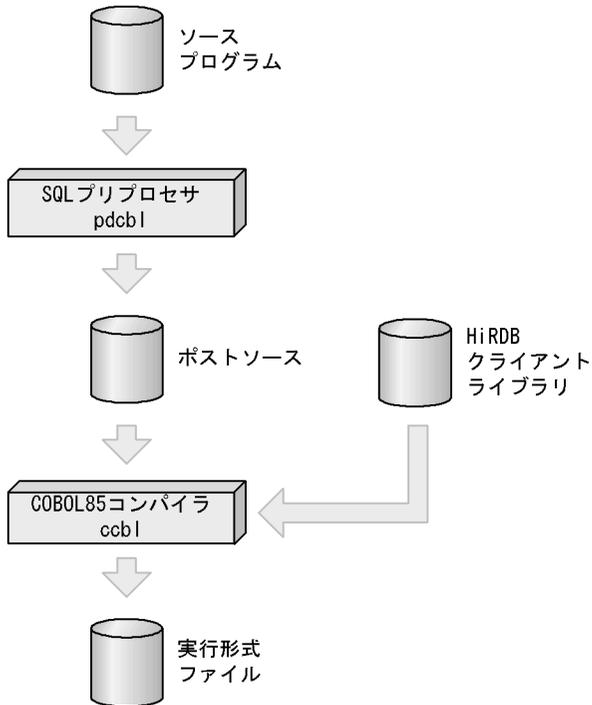
7.4 COBOL 言語でアプリケーションを作成する

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

C:\¥CBLSAMPLE¥cb1sample.ecb

ポストソース

C:\¥CBLSAMPLE¥cb1sample.cbl

実行形式ファイル

C:\¥CBLSAMPLE¥cb1sample.exe

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。

- COBOL2002

(1) COBOL2002 をインストール

COBOL2002 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

COBOL 言語のコンパイラで HiRDB のライブラリを使用できるように、環境変数を設定します。C:¥ディレクトリ下の AUTOEXEC.BAT ファイルに次の行を追加してください。

```
set CBLLIB=C:¥PROGRA~1¥HITACHI¥HIRDB¥INCLUDE
```

追加したら、Windows 2000 を再起動してください。

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。エクスプローラで次のディレクトリを作成してください。

```
C:¥CBLSAMPLE
```

7. アプリケーションを作成してみましょう

(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。

```
C:¥CBLSAMPLE¥cblsample.ecb
```

ソースプログラム (cblsample.ecb) の内容を次に示します。

```

000100 IDENTIFICATION DIVISION.
000200*****
000300* [概要]
000400* HiRDBファーストステップガイドのサンプルプログラムです。
000500* コマンドライン引数で指定された価格以上のレコードを
000600* MANAGER.GOODS表から検索します。
000700*****
000800*
000900 PROGRAM-ID. CBLSAMPLE.
001000*
001100 ENVIRONMENT DIVISION.
001200*
001300 DATA DIVISION.
001400 WORKING-STORAGE SECTION.
001500 01 CMDLINE PIC X(18).
001600 01 ARG PIC X(8) JUSTIFIED RIGHT.
001700 01 TANKA PIC -9(7).
001800* 埋込み変数及び標識変数の宣言 *****
001900 EXEC SQL
002000 BEGIN DECLARE SECTION
002100 END-EXEC.
002200 77 XUSERID PIC X(7).
002300 77 XPSWD PIC X(7).
002400 77 GSCODE PIC X(6).
002500 77 GSNAME PIC X(30).
002600 77 GTANKA PIC S9(9) COMP.
002700 77 GSHIIRE PIC X(5).
002800 77 ATANKA PIC S9(9) COMP.
002900 EXEC SQL
003000 END DECLARE SECTION
003100 END-EXEC.
003200*
003300 PROCEDURE DIVISION.
003400* 開始のメッセージの出力とコマンドライン引数の読み取り *****
003500 DISPLAY '処理を開始します。'
003600 ACCEPT CMDLINE FROM COMMAND-LINE
003700 UNSTRING CMDLINE DELIMITED BY SPACE INTO ARG
003800 MOVE ARG TO ATANKA.
003900
004000* エラー時の処置の指定 (HiRDB接続前) *****
004100 EXEC SQL
004200 WHENEVER SQLERROR GO TO ERR2
004300 END-EXEC.
004400
004500* HiRDBとの接続 *****
004600 DISPLAY 'USERID:' WITH NO ADVANCING
004700 ACCEPT XUSERID
004800 DISPLAY 'PASSWORD:' WITH NO ADVANCING
004900 ACCEPT XPSWD
005000 EXEC SQL
005100 CONNECT :XUSERID IDENTIFIED BY :XPSWD
005200 END-EXEC
005300 DISPLAY 'HiRDBとの接続に成功しました。'
005400

```

7. アプリケーションを作成してみましょう

```
005500* エラー時の処置の指定 (HiRDB接続後) *****
005600 EXEC SQL
005700     WHENEVER SQLERROR GO TO ERR1
005800 END-EXEC.
005900
006000* カーソルの宣言とオープン *****
006100 EXEC SQL
006200     DECLARE CR1 CURSOR FOR
006300     SELECT ショウヘンCD, ショウヘンメイ, タンカ, シレサキCD FROM MANAGER. GOODS
006400     WHERE タンカ >= :ATANKA
006500 END-EXEC.
006600 EXEC SQL
006700     OPEN CR1
006800 END-EXEC.
006900
007000* 検索結果の出力 *****
007100 DISPLAY 'ショウヘンCD ショウヘンメイ' WITH NO ADVANCING
007200 DISPLAY '          タンカ          シレサキCD'
007300 DISPLAY '-----' WITH NO ADVANCING
007400 DISPLAY '-----',
007500 EXEC SQL
007600     WHENEVER NOT FOUND GO TO CONT
007700 END-EXEC.
007800 LOOP.
007900 EXEC SQL
008000     FETCH CR1 INTO :GSCODE, :GSNAME, :GTANKA, :GSHIIRE
008100 END-EXEC
008200 DISPLAY GSCODE WITH NO ADVANCING
008300 DISPLAY ' ' WITH NO ADVANCING
008400 DISPLAY GSNAME WITH NO ADVANCING
008500 DISPLAY ' ' WITH NO ADVANCING
008600 MOVE GTANKA TO TANKA
008700 DISPLAY TANKA WITH NO ADVANCING
008800 DISPLAY '. ' WITH NO ADVANCING
008900 DISPLAY GSHIIRE
009000 GO TO LOOP.
009100
009200* カーソルのクローズ *****
009300 CONT.
009400 EXEC SQL
009500     WHENEVER NOT FOUND CONTINUE
009600 END-EXEC.
009700 EXEC SQL
009800     CLOSE CR1
009900 END-EXEC.
010000 GO TO DISC.
010100
010200* エラー発生時の処置 *****
010300 ERR1.
010400 DISPLAY 'エラーが発生しました。'.
010500
```

```

010600* HiRDBからの切り離し *****
010700 DISC.
010800     EXEC SQL
010900     WHENEVER SQLERROR CONTINUE
011000     END-EXEC.
011100     EXEC SQL
011200     DISCONNECT
011300     END-EXEC.
011400     DISPLAY 'HiRDBから切り離しました。'.
011500     GO TO FIN.
011600
011700* HiRDBに接続できなかった場合の処置 *****
011800 ERR2.
011900     DISPLAY 'HiRDBとの接続に失敗しました。'.
012000
012100* 終了処理 *****
012200 FIN.
012300     DISPLAY '処理を終了します。'
012400     STOP RUN.
012500 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】

```
C:¥>pdcb1 C:¥CBLSAMPLE¥cblsample.ecb /o C:¥CBLSAMPLE¥cblsample.cbl
```

【説明】

C:¥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 C:¥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、C:¥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】

7. アプリケーションを作成してみましょう

```
C:¥>ccbl2002 C:¥CBLSAMPLE¥cblsample.cbl "C:¥Program
Files¥Hitachi¥HiRDB¥lib¥Cltdll.lib"
```

【説明】

C:¥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。

"C:¥Program Files¥Hitachi¥HiRDB¥lib¥Cltdll.lib"

インポートライブラリを指定します。

コマンドが正常に終了すると、C:¥CBLSAMPLE ディレクトリ下に cblsample.exe 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。

サンプル UAP にはコマンド引数として価格を指定できます。ここでは、価格が 500000 以上の商品を GOODS 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】

```
C:¥>C:¥CBLSAMPLE¥cblsample 500000
```

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】

```
処理を開始します。
```

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との接続に成功しました。

シヨウヒンCD	シヨウヒンメイ	タカ	シレサキCD
CP0001	コピーキ(カテー)	0620000.	DE031
PR0000	プロジェクター-150インチ(キョウムヨウ)	1000000.	DE010
PR0009	プロジェクター-100インチ	0500000.	DE009
TV0007	テレビ-34センチ(ハイビジョン)	0500000.	DE007
VD0010	ビデオ-PRO(キョウムヨウ)	0500000.	DE020

HiRDBから切り離しました。

処理を終了します。

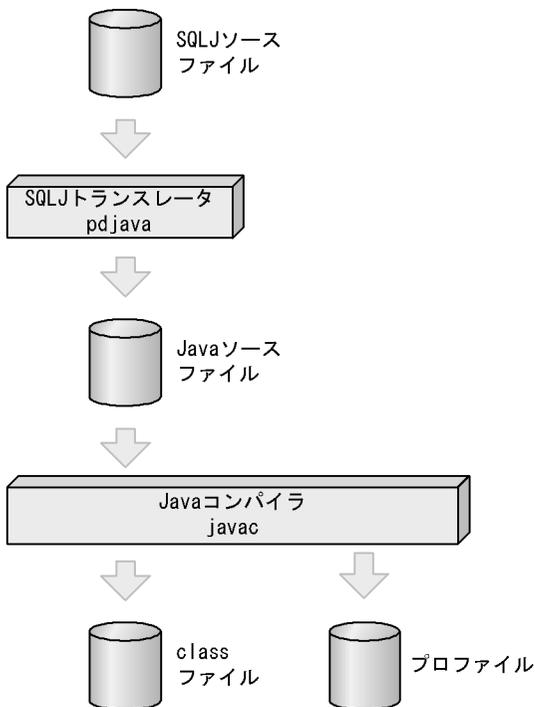
7.5 Java 言語 (SQLJ) でアプリケーションを作成する

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

C:¥JAVASAMPLE¥javasample.sqlj

ポストソース

C:¥JAVASAMPLE¥javasample.java

class ファイル

C:¥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 1.3.0-01 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

Java 言語のコンパイラで HiRDB のライブラリを使用できるように、環境変数を設定します。C:¥ディレクトリ下の AUTOEXEC.BAT に次の行を追加してください。

```
set
CLASSPATH=%CLASSPATH%;C:¥PROGRA~1¥HITACHI¥HiRDB¥utl1¥PDSQLJ.JAR;C:¥
WIN32APP¥HITACHI¥HIRDB_¥S¥CLIENT¥UTL¥PDJDBC.JAR
```

追加したら、Windows 2000 を再起動してください。

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。エクスプローラで次のディレクトリを作成してください。

7. アプリケーションを作成してみましょう

C:¥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。

C:¥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

```
/* **** */
/* [概要] */
/* HiRDBファーストステップガイドのサンプルプログラムです。 */
/* コマンドライン引数で指定された価格以上のレコードを */
/* MANAGER_GOODS表から検索します。 */
/* **** */
import java.awt.*;
import java.lang.*;
import java.io.*;
import java.sql.*;

#sql context ExecutionContext;
#sql iterator ReadIterator(String, String, String, String);

public class javasample
{
public static void main (String args[])
{
/* 埋め込み変数の宣言 **** */
String xUserId = null; /* ユーザID */
String xPswd = null; /* パスワード */
String gSCode = null; /* 商品コード */
String gSName = null; /* 商品名 */
String gTanka = null; /* 単価 */
String gShiire = null; /* 仕入先コード */
String aTanka = null; /* 検索条件 */
BufferedReader inp = null; /* ユーザ認証用バッファ */
ExecutionContext execCtx = null; /* 接続コンテキスト */
ReadIterator readItr = null; /* カーソル */

/* 開始メッセージの出力とコマンドライン引数の読み取り **** */
if( args.length == 1 ){
aTanka = new String(args[0]);
}
System.out.println("処理を開始します。");

try {

/* HiRDBへの接続 **** */
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.PrbdbDriver");
execCtx = new ExecutionContext("jdbc:hitachi:PrbdbDrive", xUserId, xPswd, false);
System.out.println("HiRDBとの接続に成功しました。");
} catch (Exception e) {

/* エラー時の処置の指定 (HiRDB接続前) **** */
System.out.println("*** HiRDBとの接続に失敗しました。 ***");
System.out.println(e.getMessage());
return;
}
}
}
```

7. アプリケーションを作成してみましょう

```
try {
/* カーソルの宣言とオープン *****/
    if ( aTanka == null ) {
        #sql [execCtx]      readItr =
            { SELECT ショウビンCD, ショウビンメイ, タンカ, シレサキCD FROM MANAGER.GOODS };
    } else {
        #sql [execCtx]      readItr =
            { SELECT ショウビンCD, ショウビンメイ, タンカ, シレサキCD FROM MANAGER.GOODS
              WHERE タンカ >= :aTanka };
    }

/* 検索結果の出力 *****/
    System.out.println("   ショウビンCD   ショウビンメイ           "+
                       "   タンカ   シレサキCD");
    System.out.println(" -----");
    while( true ){
        #sql { FETCH :readItr INTO :gSCode, :gSName, :gTanka, :gShiire };
        if( readItr.endFetch() ) break;
        System.out.println(" "+gSCode+" "+gSName+" "+gTanka+" "+gShiire);
    }

/* カーソルのクローズ *****/
    readItr.close();

/* HiRDBからの切り離し *****/
    execCtx.close();
    System.out.println("HiRDBから切り離しました。");

/* 終了処理 *****/
    System.out.println("処理を終了します。");

} catch (SQLException e) {

/* エラー発生時の処置 (HiRDB接続後) *****/
    System.out.println("*** SQLでエラーが発生しました。***");
    System.out.println("ErrorCode = "+e.getErrorCode());
    System.out.println(e.getMessage());
    try {
        execCtx.close();
    } catch (Exception ee) {
        ;
    }

} catch (Exception e) {
    System.out.println("*** なんらかのエラーが発生しました。***");
    System.out.println(e.getMessage());
}
}
```

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】

```
C:¥>pdjava C:¥JAVASAMPLE¥jvasample.sqlj
```

【説明】

```
C:¥JAVASAMPLE¥jvasample.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、C:¥JAVASAMPLE ディレクトリ下に jvasample.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】

```
C:¥>javac C:¥JAVASAMPLE/jvasample.java
```

【説明】

```
C:¥JAVASAMPLE/jvasample.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、C:¥JAVASAMPLE ディレクトリ下に jvasample.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

7. アプリケーションを作成してみましょう

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 にはコマンド引数として価格を指定できます。ここでは、価格が 500000 以上の商品を GOODS 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】

```
C:¥>java C:¥JAVASAMPLE¥jvasample 500000
```

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】

```
処理を開始します。
```

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との接続に成功しました。
```

ショウヒンCD	ショウヒンメイ	タカ	シレサキCD
CP0001	コピーキ(カラー)	620000	DE031
PR0000	プロジェクター-150インチ(キョウムヨウ)	1000000	DE010
PR0009	プロジェクター-100インチ	500000	DE009
TV0007	テレビ-34カタ(ハイビジョン)	500000	DE007
VD0010	ビデオ-PRO(キョウムヨウ)	500000	DE020

```
HiRDBから切り離しました。
```

```
処理を終了します。
```

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. インストールに関する質問
2. 環境設定に関する質問
3. HiRDB のシステム構成に関する質問
4. 運用に関する質問
5. トラブルに関する質問

付録 A.1 インストールに関する質問

(1) インストールディレクトリ

質問

インストールディレクトリについて，何か注意することはありますか？

お答えします

インストールディレクトリの名称を指定するときに次の点に注意してください。なお，インストールディレクトリのパス名は，環境変数 PDDIR に格納されています。

- 空白，全角文字，および特殊記号は使用しないでください。
- 長さは 128 バイト以内にしてください。
- システム共通定義の pdunit オペランドで指定する HiRDB 運用ディレクトリ名と，同じ名称にしてください。

(2) 1073 エラー

質問

インストール時に 1073 エラーになります。

お答えします

この 1073 エラーは，HiRDB のアンインストール後に OS を再起動しないで，再度 HiRDB をインストールしたときに発生します。HiRDB をインストール，またはアンインストールしたあとには，OS を再起動する必要があります。対処方法については，マニュアル「HiRDB Version 7 システム導入・設計ガイド (Windows(R) 用)」を参照してください。

付録 A.2 環境設定に関する質問

(1) 仮想メモリの見積もり方法

質問

仮想メモリの見積もり方法について教えてください。

お答えします

一般的には実メモリの 2, 3 倍程度ですが、仮想メモリはディスクを使用するため、大き過ぎるとディスクを圧迫することになります。これらを考慮して、見積もってください。

なお、実メモリを使用するか、または仮想メモリを使用するかの制御は、OS が管理しています。HiRDB では制御できません。

(2) ネットワークドライブの使用

質問

ネットワークドライブは使用できますか？

お答えします

使用できません。HiRDB のインストールもそうですが、ユティリティのバックアップファイルなどもすべてローカルドライブを使用してください。

(3) 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

コマンドプロンプトから 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。

[コントロールパネル] - [ネットワーク] の [プロトコル] タブの TCP/IP のプロパティをクリックしてください。[DNS] タブのホストに設定されている名称が、ホスト名です。

付録 A.3 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) Windows 版の SORT

質問

UNIX 版の HiRDB では、SORT が前提プログラムプロダクトになっていますが、Windows 版の HiRDB では SORT は必要ですか？

お答えします

Windows 版の HiRDB には SORT が組み込まれているため、SORT を用意する必要はありません。ただし、この組み込まれている SORT は、HiRDB 用なので単体では使用できません。

(3) 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。

- HiRDB/Run Time (JDBC, ODBC, OLE DB プロバイダを含む)
- HiRDB Control Manager - Server
- HiRDB Control Manager - Console
- HiRDB Control Manager - Agent
- HiRDB Assist - Server
- HiRDB Assist - Client
- HiRDB SQL Executer
- HiRDB Access Path Browser
- HiRDB Dataextractor
- HiRDB Datareplicator
- SORT

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.4 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。

(2) pdcancel と pdkill の使い分け

質問

pdcancel コマンドと pdkill コマンドは、どのように使い分ければ良いのですか？

お答えします

次のように使用してください。

- pdcancel コマンド
トランザクション処理中のサーバのプロセスを終了させます。
- pdkill コマンド
トランザクション処理をしていないサーバのプロセスを終了させます。

(3) 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.5 トラブルに関する質問

(1) HiRDB のサービス起動時に、サービスのエラーコード：1067 が出力される

質問

HiRDB のサービス起動時に、サービスのエラーコード：1067 が出力されます。

お答えします

次のどちらかの原因が考えられます。

- 仮想メモリ不足の可能性があります。仮想メモリの指定値は、実メモリの 2, 3 倍程度を推奨しています。また、[コントロールパネル] - [システム] - [仮想メモリ] で、初期値と MAX が同じ値になっているか確認してください。
- %PDDIR%\tmp ディレクトリが、ほかのアプリケーションプログラムからアクセスされたままになっていないか確認してください。HiRDB は、開始時に作業ディレクトリである %PDDIR%\tmp を削除しているため、ほかのアプリケーションプログラムで %PDDIR%\tmp をアクセスしていると排他エラーとなります。

(2) HiRDB のサービスを開始したが応答が返ってこない

質問

HiRDB のサービスを開始しましたが、応答が返ってきません。また、イベントログにも何も出力されていません。

お答えします

システム定義の `pd_mode_conf` オペランドの指定値が `MANUAL1`、または `MANUAL2` になっていないか確認してください。
`MANUAL1`、または `MANUAL2` の場合は、さらにコマンドプロンプトから `pdstart` コマンドを実行する必要があります。
`AUTO` の場合は、サービスを開始すればそのまま `HiRDB` を開始します。

(3) 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。

(4) 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;
```

(5) データベース再編成ユ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。Windows NT の場合は、`port` 解放後 4 分間は使用できなくなります。データベース再編成ユティリティ以外でも、`HiRDB` 起動時を含め、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、ある程度時間をおいてコマンドを実行してください。

(6) データベース複製ユ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上限を超えている可能性があります。バックアップファイルの最大値は次のとおりです。

条件		バックアップファイルの最大値
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する場合	pd_large_file_use = N 指定時 (省略値)	2,047MB
	pd_large_file_use=Y 指定時	1,048,575MB
バックアップファイルを Windows のファイルで作成する場合		$2^{63} - 1$ バイト

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. [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。
2. 【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。
3. 一覧の中から「HiRDB/Single Server」を選択し、「追加と削除」を選択してください。
4. アンインストールが始まります。しばらくすると、「共有ファイルを削除しますか？」という問い合わせのメッセージが表示されることがあります。表示されたら、「いいえ」をクリックしてください。
5. アンインストールが終わると、「HiRDB/Single Server の再インストールは、コンピュータを再起動したあとに行ってください」というメッセージが表示されます。「OK」をクリックしてください。
6. 【コンピュータからプログラムを削除】ダイアログの「OK」をクリックしてください。
7. アンインストール終了です。
8. アンインストールしても、インストール後に作成されたディレクトリ、ファイルなどは、インストールディレクトリ「hirdb_s」下に残ります。エクスプローラなどを使用して、C:\win32app\hitachi ディレクトリ下の「hirdb_s」ディレクトリを削除してください。ただし、HiRDB Control Manager をアンインストールしていない場合には、先に HiRDB Control Manager をアンインストールしてから、「hirdb_s」ディレクトリを削除してください。

注意

[アプリケーションの追加と削除] でアンインストールを実行して [アプリケーションの追加と削除] ダイアログからの応答がなくなった場合は、タスクマネージャで [アプリケーションの追加と削除] ダイアログを終了させてください。この場合、アンインストールは正しく実行されているので問題ありません。

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. [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. 「セットアップと削除」を選択してください。
3. 一覧の中から「HiRDB」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完了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。
6. 「C:\AUTOEXEC.BAT」ファイルに追加した PATH の行を削除するか、または行の先頭に「REM 」を入力してファイルを閉じるかしてください。

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. [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 SQL Executer」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完了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「Pdsq1」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「Pdsq1」ディレクトリを削除してください。

付録 B.4 HiRDB Control Manager のアンインストール手順 (サーバ側)

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. [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 - Agent」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完了し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「CMAgent」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:\win32app\hitachi\hirdb_s ディレクトリ下の「CMAgent」ディレクトリを削除してください。

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完了し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完了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 - 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)

簡易セットアップツール

HiRDB サーバのセットアップを GUI を使ってできるツールです。

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使用できるユーザ用 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。

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います。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います。

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 232

CONNECT 権限 132, 258

C 言語でアプリケーションを作成する 225

D

DAT のバックアップを使用して回復する 103

DBA 権限 132, 258

DELETE 権限 132

F

FIX 属性 258

G

GRANT 文 133

H

HiRDB.ini ファイル 258

HiRDB/Developer's Kitのアンインストール 255

HiRDB/Single Server のアンインストール 254

HiRDB/ クライアント 259

HiRDB/ クライアントの環境設定をする
〔COBOL 言語〕 233

HiRDB/ クライアントの環境設定をする〔C
言語〕 226

HiRDB/ クライアントの環境設定をする
〔Java 言語〕 241

HiRDB/ クライアントをインストールする 205

HiRDB/ シングルサーバのインストール 18

HiRDB Control Manager 140, 258

HiRDB Control Manager - Agent 24, 258

HiRDB Control Manager - Console 24, 258

HiRDB Control Manager - Server 24, 258

HiRDB Control Manager のアンインストール 256

HiRDB Control Manager のインストール手
順 24

HiRDB Control Manager をインストールす
る 23

HiRDB Control Manager を開始する 141

HiRDB Control Manager を終了する 142

HiRDB SQL Executer 259

HiRDB SQL Executer のアンインストール
255

HiRDB SQL Executer のインストール 23

HiRDB SQL Executer をインストールする
208

HiRDB が出力するメッセージ 8

HiRDB 管理者 259

HiRDB システム・ユニット・サーバの開始
71

HiRDB システム・ユニット・サーバの終了
71

HiRDB システム定義 6, 259

HiRDB システムの構築手順 12

HiRDB システムを構築するときのユーザ名
13

HiRDB のアンインストール 254

HiRDB の開始 71

HiRDB の稼働状況を確認する 41

HiRDB の環境設定をする（簡易セットアッ
プツールでの実行） 27

HiRDB のシステム構成に関する質問 249

HiRDB の終了 72

HiRDB ファイル 259

HiRDB ファイルシステム領域 6, 259

HiRDB をインストールする 14

HiRDB を正常開始する 74, 146

HiRDB を正常終了する 73, 145

HiRDB を登録する 143

hosts ファイルの設定 208

I

INSERT 権限 132

J

Java 言語 (SQLJ) でアプリケーションを作成する 240

M

Microsoft Access から HiRDB をアクセスする 216

O

ODBC 対応のアプリケーションプログラムからアクセスする 214

ODBC ドライバをインストールする 214

OS ごとの操作の差異 8

P

pdcopy コマンド 84

pddef 48

pdload 51

pdlogunld コマンド 76, 78

pdmod コマンド 119, 123

pdrorg コマンド 115

pdrstr コマンド 84, 97

pdstart コマンド 71

pdstop コマンド 72

R

RD エリア 7, 259

RD エリアの拡張手順 119

RD エリアの自動増分 259

RD エリアの追加手順 123

RD エリア利用権限 132, 259

RD エリアを拡張する 119, 120, 180

RD エリアを追加する 123, 186

S

SELECT 権限 132

SQLJ 220

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〕 245

SQL プリプロセサでソースプログラムを変換する〔COBOL 言語〕 237

SQL プリプロセサでソースプログラムを変換する〔C 言語〕 229

U

UAP 220

UAP 作成の流れ 221

UAP のソースプログラムを作成する

〔COBOL 言語〕 233

UAP のソースプログラムを作成する〔C 言語〕 226

UAP のソースプログラムを作成する〔Java 言語〕 241

UPDATE 権限 132

あ

空きディスク容量 17

アクセス権限 132, 260

アクセス権限がある表を検索する 134

アクセス権限がない表を検索する 135

アンロード状態のチェックを解除する運用 78

アンロード済み状態 260

アンロード待ち状態 78, 260

アンロードレスシステムログ運用 78

アンロードログファイル 76, 260

い

異常終了 73

イベントビューアの開き方 8

インストールディレクトリ 8

インストールに関する質問 248

インデクス 260

う

埋込み型 UAP 220

運用に関する質問 250

か

開始 71, 145
 開始モード 71
 開発環境を準備する〔COBOL 言語〕 233
 開発環境を準備する〔C 言語〕 226
 開発環境を準備する〔Java 言語〕 241
 回復 97, 163
 学習用の HiRDB システムを構築するときの
 マシン環境 3
 仮想メモリ 18
 簡易セットアップツール 27, 260
 簡易セットアップツールを実行する 32
 環境設定に関する質問 248
 環境変数 PATH の設定 209
 管理 HiRDB 143

き

基礎知識〔COBOL 言語〕 232
 基礎知識〔C 言語〕 225
 基礎知識〔Java 言語〕 240
 強制開始 72
 強制終了 73

く

クライアントから表を操作する 211
 クライアント環境定義の設定 209
 クライアントの環境設定 205
 クライアントの接続手順 204
 グローバルバッファプール 260

け

計画停止 72
 権限がないユーザで表を検索する 136
 権限を追加する 198
 現用 78, 260

こ

更新前ログ取得モード 260
 構築する HiRDB システムの構成 6
 公用 RD エリア 133, 260
 コマンドの実行 8

コントロールパネルの開き方 8
 コンパイルを実行する〔COBOL 言語〕 237
 コンパイルを実行する〔C 言語〕 230
 コンパイルを実行する〔Java 言語〕 245
 コンピュータの管理の開き方 8

さ

サーバ 261
 再開始 72
 最新の状態への回復 99
 再編成 115, 174, 261
 作業表用ファイル 261
 作成した UAP を実行する〔COBOL 言語〕
 238
 作成した UAP を実行する〔C 言語〕 231
 作成した UAP を実行する〔Java 言語〕 245
 作成するアプリケーションの仕様 223
 作成する環境 27
 サンプル UAP の機能 223
 サンプル UAP の処理の流れ 224

し

システム環境変数 TZ 15
 システムキャッシュの設定 18
 システム共通定義 (pdsys) 35
 システムファイル 261
 システム用 RD エリア 261
 システム用 RD エリアのバックアップ取得時
 点への回復 99
 システムログ 76, 148
 システムログとデータベース回復の関係 77
 システムログのアンロード手順 81
 システムログの運用の種類 77
 システムログファイル 6, 261
 システムログファイルの運用方法 32
 システムログファイルの状態変化 79
 システムログファイルのスワップ 78
 システムログをアンロードする 81, 149
 システムログをアンロードする運用 78
 実表 261
 自動ログアンロード 261
 自動ログアンロード機能 80

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 98
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 88
自動ログアンロード機能を使っている場合に
回復する 107, 166
自動ログアンロード機能を使っている場合に
全 RD エリアのバックアップを取得する 161
終了 71, 145
終了モード 72
私用 RD エリア 133, 261
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 222
新規ユーザを登録する 133, 197
シンクポイントダンプファイル 7, 261
シングルサーバ 262
シングルサーバ定義 (sds01) 37

す

スキーマ 262
スキーマ定義権限 132, 262
ステータスファイル 7, 262
スワップ 262

せ

正常開始 72
正常終了 72
製品の画面や設定値について 9
セグメント 262
全 RD エリア (HiRDB システム) のバック
アップを取得する 153
全 RD エリアのバックアップ〔自動ログアン
ロード機能〕93
全 RD エリアのバックアップを取得する 89

て

定義ファイルを変更したい場合 41
ディクショナリ表 262
データディクショナリ用 RD エリア 262
データディレクトリ用 RD エリア 262
データベース回復の概要 84
データベース回復ユーティリティ 84, 97

データベース構成変更ユーティリティ
119, 123
データベース再編成ユーティリティ 115
データベース作成ユーティリティ 51
データベース定義ユーティリティ 48
データベースの回復手順 97
データベース複写ユーティリティ 84
データロード 262
適用機種 15

と

トラブルに関する質問 251

は

バックアップ 84, 152
バックアップ取得時点に回復する 99, 163
バックアップ取得モード 85, 262
バックアップの取得単位 84
バックアップの取得手順 87
バックアップファイルの格納場所 86
バックアップを DAT に取得する 91

ひ

表の行を更新する 65
表の行を削除する 66
表の再編成 115, 174
表の再編成手順 116
表の所有者を調べる 137
表へ行を挿入する 64, 212
表へデータロードをする 51
表を検索する 62
表を再編成する 116, 174
表を作成する 46
表を操作する 62
表を定義する 48, 211

へ

ページ 263

ほ

ほかの所有者の表を検索する 212

ま

マスタディレクトリ用 RD エリア 263
マスタディレクトリ用 RD エリアの回復 99
マルチ HiRDB 263

ゆ

ユーザ権限 132, 197, 263
ユーザ名を登録する 46
ユーザ用 RD エリア 263
ユーザを登録する 132
ユニット 263
ユニット制御情報定義 (pdutysys) 37

ろ

ログ関係ファイルのアンロード 77
ログ取得モード 264
ログレスモード 264

(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。