

HiRDB データ連動機能

HiRDB Datareplicator Version 10

解説・手引・操作書

3020-6-573-10

前書き

■ 対象製品

●適用 OS : HP-UX 11i V3(IPF)

P-1J62-1KA1 HiRDB Datareplicator Version 10 10-00

●適用 OS : AIX V7.1, AIX V7.2

P-1M62-1KA1 HiRDB Datareplicator Version 10 10-04

●適用 OS : Red Hat Enterprise Linux 7 (64-bit x86_64), Red Hat Enterprise Linux 8 (64-bit x86_64)

P-8462-1KA1 HiRDB Datareplicator Version 10 10-04

●適用 OS : Windows Server 2016 Standard, Windows Server 2016 Datacenter, Windows Server 2019 Standard, Windows Server 2019 Datacenter, Windows Server 2022 Standard, Windows Server 2022 Datacenter, Windows 10 Pro (x64), Windows 10 Enterprise (x64), Windows 11 Pro (x64), Windows 11 Enterprise (x64)

P-2962-1KA4 HiRDB Datareplicator Version 10 10-04

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

■ 輸出時の注意

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

なお、不明な場合は、弊社担当営業にお問い合わせください。

■ 商標類

HITACHI, HiRDB, Cosminexus, HA モニタ, OpenTP1, TPBroker, uCosminexus, VOS3/LS, VOS3/US, VOS3/XS, XDM は、株式会社 日立製作所の商標または登録商標です。

Amazon Web Services, AWS, Powered by AWS ロゴ, Amazon EC2, Amazon Route 53 は、Amazon.com, Inc.またはその関連会社の商標です。

AMD は、Advanced Micro Devices, Inc.の商標です。

Ansible is a registered trademark of Red Hat, Inc. in the United States and other countries.

Ansible は、米国およびその他の国における Red Hat, Inc.の登録商標です。

Hibernate is a registered trademark of Red Hat, Inc. in the United States and other countries.

Hibernate は、米国およびその他の国における Red Hat, Inc.の登録商標です。

IBM, AIX, DataStage, DB2, HACMP, MetaBroker, MetaStage, OS/390, PowerHA および QualityStage は、世界の多くの国で登録された International Business Machines Corporation の商標です。

Itanium は、Intel Corporation またはその子会社の商標です。

Jboss is a registered trademark of Red Hat, Inc. in the United States and other countries.

Jboss は、米国およびその他の国における Red Hat, Inc.の登録商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Microsoft, Access, ActiveX, Azure, Excel, Visual Basic, Visual C++, Visual Studio, Windows, Windows Server は、マイクロソフト 企業グループの商標です。

Oracle および Java は、オラクルおよびその関連会社の登録商標です。

Red Hat is a registered trademark of Red Hat, Inc. in the United States and other countries.

Red Hat は、米国およびその他の国における Red Hat, Inc.の登録商標です。

Red Hat Enterprise Linux is a registered trademark of Red Hat, Inc. in the United States and other countries.

Red Hat Enterprise Linux は、米国およびその他の国における Red Hat, Inc.の登録商標です。

UNIX は、The Open Group の商標です。

その他記載の会社名、製品名などは、それぞれの会社の商標もしくは登録商標です。

■ 発行

2022 年 10 月 3020-6-573-10

■ 著作権

All Rights Reserved. Copyright (C) 2018, 2022, Hitachi, Ltd.

変更内容

変更内容 (3020-6-573-10) HiRDB Datareplicator Version 10 10-04

追加・変更内容	変更箇所
抽出側の表をすべて抽出する場合に、抽出定義ファイルへの表数分のオペランドの記載を不要とする機能（全表抽出定義文）をサポートしました。	1.3, 表 4-2, 4.2.2, 4.3.3(1), 5.5.1(1), 5.5.1(2), 5.5.5, 6.8, 6.8.1, 7
次のメッセージを追加しました。 KFRB01116-E, KFRB01117-E, KFRB01333-E, KFRB04571-E, KFRB04582-E, KFRB04583-W, KFRB04584-E, KFRB04585-E, KFRB04586-E, KFRB04587-E, KFRB04588-E, KFRB04589-E	10.2
次のメッセージを変更しました。 KFRB00045-E, KFRB02053-W, KFRB04500-I	10.2

単なる誤字・脱字などはお断りなく訂正しました。

変更内容 (3020-6-573) HiRDB Datareplicator Version 10 10-00

追加・変更内容
マニュアルの体裁を変更しました。

HiRDB Datareplicator Version 8(64) 08-14

追加・変更内容
Windows 版 Datareplicator の場合、hdsstop -t force で反映側 Datareplicator を停止できない場合に、反映側 Datareplicator の状態に関係なく、反映側 Datareplicator の全プロセスを強制終了するコマンド hdsemergencystop をサポートしました。
次のメッセージを変更しました。 KFRB00911-Q, KFRB00912-I, KFRB00913-E, KFRB00914-I

HiRDB Datareplicator Version 8(64) 08-11

追加・変更内容
Windows 版 Datareplicator の場合、稼働トレースの取得による性能への影響を低減するため、プロセスごとに稼働トレースファイルを割り当てる機能（プロセス単位収集方式）をサポートしました。 これに伴い、抽出システム定義および反映システム定義に次のオペランドを追加しました。 <ul style="list-style-type: none">int_trc_unit オペランド int_trc_output_path オペランド

変更内容 (3020-6-360-70) HiRDB Datareplicator Version 8 08-09, HiRDB Datareplicator Version 8 (64) 08-09

追加・変更内容
抽出側 Datareplicator 稼働中に抽出処理が停止していないかどうかを監視する抽出処理監視機能をサポートしました。 これに伴い、次のオペランドを追加しました。 <ul style="list-style-type: none">extwatch_intvl オペランド
HiRDB の暗黙的なロールバックと反映処理のリトライの競合によって、抽出側 DB と反映側 DB が不整合になるおそれがある場合、skip_sqlcode オペランドの指定を無視して、SQL エラーのスキップを抑止できるようにしました。 これに伴い、次のオペランドを追加しました。 <ul style="list-style-type: none">sqlerr_during_retry オペランド
UNIX の場合の、反映ステータスファイルの容量の見積もり式を変更しました。
次のメッセージを追加しました。 KFRB03061-E, KFRB05043-I, KFRB05044-W, KFRB05045-I
次のメッセージを変更しました。 KFRB00033-E, KFRB00034-E, KFRB02003-E, KFRB05019-E

変更内容 (3020-6-360-60) HiRDB Datareplicator Version 8 08-07, HiRDB Datareplicator Version 8 (64) 08-07

追加・変更内容
抽出側 Datareplicator, 及び反映側 Datareplicator で障害が発生した場合、障害の原因調査に必要な情報を取得できるようにしました。 これに伴い、次のコマンドを追加しました。 <ul style="list-style-type: none">hdeinfoget コマンドhdsinfoget コマンド
抽出側, 及び反映側の HiRDB のバージョンが 09-50 以降の場合、抽出定義の extract 文の key, 及び ukey に指定できる列数を最大 64 個にしました。
列データ編集 UOC からの戻り値に 4 を追加しました。
次のメッセージを追加しました。 KFRB00053-W, KFRB03095-W, KFRB09250-E, KFRB09253-I, KFRB09254-I, KFRB09256-E, KFRB09257-I, KFRB09258-I, KFRB09259-E, KFRB09263-E, KFRB09264-E, KFRB09265-E, KFRB09266-E, KFRB09267-I, KFRB09268-I, KFRB09269-W, KFRB09270-I, KFRB09271-I, KFRB09272-I, KFRB09273-I, KFRB09274-W, KFRB09275-W, KFRB09276-W, KFRB09277-W, KFRB09278-I
次のメッセージを変更しました。 KFRB00824-E, KFRB02003-E
HiRDB の適用 OS と使用できる Datareplicator のバージョンに、適用 OS を追加しました。

はじめに

このマニュアルは、HiRDB Datareplicator Version 10（以降、このマニュアルでは、「Datareplicator」と表記します）の機能と使い方について説明したものです。

Datareplicator をインストールできる HiRDB（HiRDB/Single Server, 及び HiRDB/Parallel Server）は、HiRDB Version 10 以降です。

■ 対象読者

Datareplicator を使ってデータ連動システムを構築／運用するシステム管理者、システム設計者、プログラマ、及びオペレータの方々を対象としています。

このマニュアルの記述は、次に示す知識があることを前提としています。

HiRDB から HiRDB へのデータ連動システムを構築／運用する方

 Datareplicator の適用 OS の基礎的な知識

 HiRDB のシステム管理者の知識

XDM/RD E2 から HiRDB へのデータ連動システム、又は HiRDB から XDM/RD E2 へのデータ連動システムを構築／運用する方

 Datareplicator の適用 OS の基礎的な知識

 HiRDB のシステム管理者の知識

 XDM E2 系のシステム管理者の知識

 XDM/RD E2 のシステム管理者の知識

 XDM/DS のシステム管理者の知識

XDM/SD E2 から HiRDB へのデータ連動システムを構築／運用する方

 Datareplicator の適用 OS の基礎的な知識

 HiRDB のシステム管理者の知識

 XDM E2 系のシステム管理者の知識

 XDM/SD E2 のシステム管理者の知識

 XDM/DS のシステム管理者の知識

ADM から HiRDB へのデータ連動システムを構築／運用する方

 Datareplicator の適用 OS の基礎的な知識

 HiRDB のシステム管理者の知識

 ADM のシステム管理者の知識

 XDM/DS のシステム管理者の知識

PDMII E2 から HiRDB へのデータ連動システムを構築／運用する方

Datareplicator の適用 OS の基礎的な知識

HiRDB のシステム管理者の知識

PDMII E2 のシステム管理者の知識

XDM/DS のシステム管理者の知識 (VOS3 PDMII E2 の場合)

TMS-4V/SP から HiRDB へのデータ連動システムを構築／運用する方

Datareplicator の適用 OS の基礎的な知識

HiRDB のシステム管理者の知識

TMS-4V/SP のシステム管理者の知識 (TMS-4V/SP/データ連携支援の知識)

XDM/DS のシステム管理者の知識

RDB1 E2 から HiRDB へのデータ連動システムを構築／運用する方

Datareplicator の適用 OS の基礎的な知識

HiRDB のシステム管理者の知識

RDB1 E2 のシステム管理者の知識

■ 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 (UNIX 用マニュアル)

- HiRDB Version 10 システム導入・設計ガイド (UNIX(R)用) (3020-6-552)
- HiRDB Version 10 システム定義 (UNIX(R)用) (3020-6-554)
- HiRDB Version 10 システム運用ガイド (UNIX(R)用) (3020-6-556)
- HiRDB Version 10 コマンドリファレンス (UNIX(R)用) (3020-6-558)
- インナレプリカ機能 HiRDB Staticizer Option Version 10 (3020-6-563)
- HiRDB Version 10 ディザスタリカバリシステム 構築・運用ガイド (3020-6-564)

HiRDB (Windows 用マニュアル)

- HiRDB Version 10 システム導入・設計ガイド (Windows(R)用) (3020-6-553)
- HiRDB Version 10 システム定義 (Windows(R)用) (3020-6-555)
- HiRDB Version 10 システム運用ガイド (Windows(R)用) (3020-6-557)
- HiRDB Version 10 コマンドリファレンス (Windows(R)用) (3020-6-559)

HiRDB (Windows, UNIX 共通マニュアル)

- HiRDB Version 10 解説 (3020-6-551)
- HiRDB Version 10 UAP 開発ガイド (3020-6-560)
- HiRDB Version 10 SQL リファレンス (3020-6-561)
- HiRDB Version 10 メッセージ (3020-6-562)
- HiRDB Version 10 XDM/RD E2 接続機能 (3020-6-565)
- HiRDB Version 10 バッチ高速化機能 (3020-6-567)
- HiRDB Version 10 パフォーマンスガイド (3020-6-568)
- HiRDB 全文検索プラグイン HiRDB Text Search Plug-in Version 9 (3020-6-481-30)
- HiRDB XML 拡張機能 HiRDB XML Extension Version 9 (3020-6-480-30)
- HiRDB データ連動拡張機能 HiRDB Datareplicator Extension Version 10 (3020-6-574)
- データベース抽出・反映サービス機能 HiRDB Dataextractor Version 10 (3020-6-575)

なお、本文中で使用している HiRDB Version 10 のマニュアル名は、Version 10, (UNIX(R)用), (Windows(R)用) を省略して表記しています。使用しているプラットフォームに応じて UNIX 用又は Windows 用のマニュアルを参照してください。

関連製品

- 通信管理 XNF/S-E2 構成定義編 (3000-3-521)
- 高信頼化システム監視機能 HA モニタ (3000-9-109) ※
- HI-UX/WE2 日立 CSMA/CD ネットワーク CD105 (TCP/IP) (3000-3-200)
- VOS3 データマネジメントシステム XDM E2 系 解説 (6190-6-620)
- VOS3 データマネジメントシステム XDM E2 系 XDM/RD E2 解説 (6190-6-637)
- VOS3 データマネジメントシステム ADM 概説 (6190-6-521)
- VOS3 TMS-4V/SP 解説 (6190-6-104)
- VOS3 TMS-4V/SP データ連携支援機能 TMS-4V/SP/データ連携支援 (6190-6-152)
- VOS3 データベース抽出プログラム XDM/XT (6190-6-641)
- VOS3 XDM データ連動機能 XDM/DS 解説・定義 (6190-6-642)
- VOS3 XDM データ連動機能 XDM/DS 運用 (6190-6-658)
- VOS3 データベースデータ連動機能 Database Datareplicator (6190-6-650)
- VOS3 データマネジメントシステム PDMII, PDMII E2 解説 (6190-6-511)
- VOS3 データマネジメントシステム PDMII, PDMII E2 ユティリティ (6190-6-513)
- VOS3 データマネジメントシステム PDMII, PDMII E2 使用の手引 (6190-6-514)

- VOS1 データマネジメントシステム PDMII E2 解説 (6150-6-511)
- VOS1 データマネジメントシステム PDMII E2 使用の手引 (6150-6-514)
- VOS1 データマネジメントシステム PDMII E2 ユティリティ (6150-6-515)
- VOS1 データマネジメントシステム RDB1 E2 解説 (6150-6-560)
- VOS1 データマネジメントシステム RDB1 E2 使用の手引 (6150-6-561)

注※

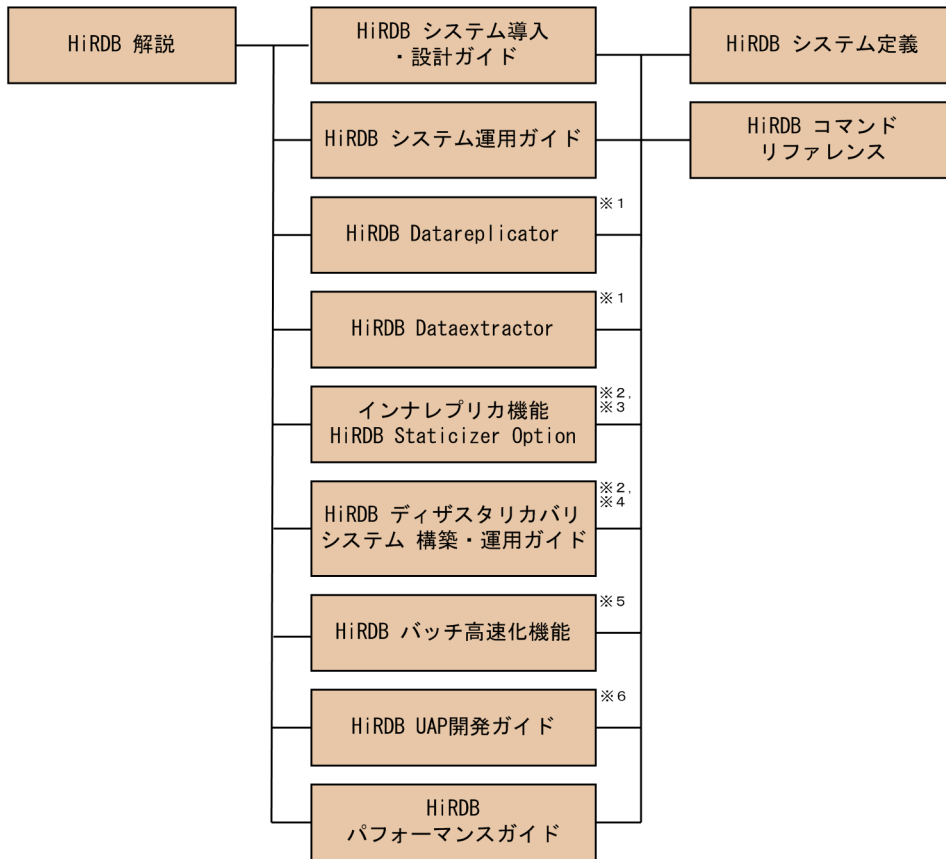
本文中で使用している HA モニタのマニュアル名は、AIX(R)編及びHP-UX(R)編を省略して表記しています。使用しているプラットフォームに応じて AIX 用、HP-UX 用、又は Linux 用のマニュアルを参照してください。

■ 利用者ごとの関連マニュアル

HiRDB のマニュアルをご利用になる場合、利用者ごとに次のようにお読みください。

また、より理解を深めるために、左側のマニュアルから順にお読みいただくことをお勧めします。

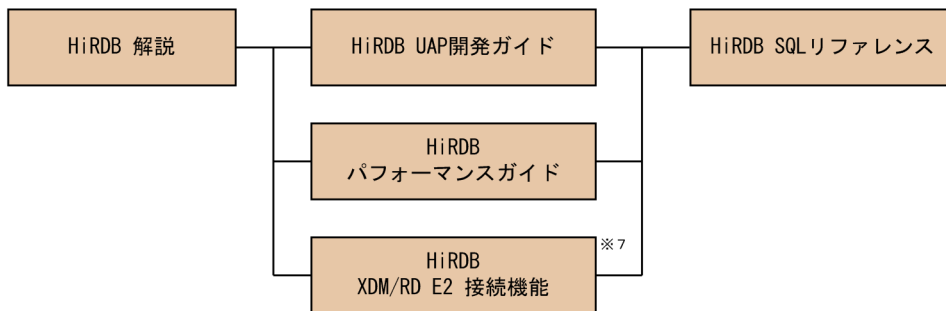
システム管理者が利用するマニュアル



表の作成者が利用するマニュアル



UAP作成者、およびUAP実行者が利用するマニュアル



- 注※1 レプリケーション機能を使用してデータ連携をする場合にお読みください。
- 注※2 UNIX用マニュアルです。Windows用はありません。
- 注※3 インナレプリカ機能を使用する場合にお読みください。
- 注※4 ディザスタリカバリシステムを構築する場合にお読みください。
- 注※5 インメモリデータ処理によるバッチ高速化を行う場合にお読みください。
- 注※6 OLTPシステムと連携する場合は必ずお読みください。
- 注※7 XDM/RD E2接続機能を使用して、XDM/RD E2のデータベースを操作する場合にお読みください。

■ このマニュアルでの表記

このマニュアルでは製品名称及び名称について次のように表記しています。ただし、それぞれのプログラムについての表記が必要な場合はそのまま表記しています。

製品名称または名称	表記	
HiRDB Server Version 10	HiRDB/シングルサーバ	HiRDB または HiRDB サーバ
	HiRDB/パラレルサーバ	
HiRDB/Developer's Kit Version 10	HiRDB/Developer's Kit	HiRDB クライアント
HiRDB/Developer's Kit Version 10(64)		
HiRDB/Run Time Version 10	HiRDB/Run Time	
HiRDB/Run Time Version 10(64)		
HiRDB Accelerator Version 10	HiRDB Accelerator	
HiRDB Advanced High Availability Version 10	HiRDB Advanced High Availability	
HiRDB Control Manager	HiRDB CM	
HiRDB Control Manager Agent	HiRDB CM Agent	
HiRDB Dataextractor Version 10	HiRDB Dataextractor	
HiRDB Datareplicator Version 10	HiRDB Datareplicator	
HiRDB Disaster Recovery Light Edition Version 10	HiRDB Disaster Recovery Light Edition	
HiRDB Non Recover Front End Server Version 10	HiRDB Non Recover FES	
HiRDB Staticizer Option Version 10	HiRDB Staticizer Option	
HiRDB Text Search Plug-in Version 9	HiRDB Text Search Plug-in	
HiRDB XML Extension Version 9	HiRDB XML Extension	
シングルサーバ	SDS	
システムマネージャ	MGR	
フロントエンドサーバ	FES	
ディクショナリサーバ	DS	
バックエンドサーバ	BES	
Microsoft ActiveX	ActiveX	
DataStage(R)	DataStage	
JP1/Magnetic Tape Access	EasyMT	
EasyMT		

製品名称または名称	表記
JP1/Automatic Job Management System 2 - Scenario Operation	JP1/AJS2-SO
JP1/Automatic Job Management System 3	JP1/AJS3
JP1/Automatic Job Management System 2	
JP1/Cm2/Extensible SNMP Agent	JP1/ESA
JP1/Cm2/Extensible SNMP Agent for Mib Runtime	
JP1/NETM/Audit - Manager	JP1/NETM/Audit
JP1/Cm2/Network Node Manager	JP1/NNM
JP1/Performance Management	JP1/PFM
JP1/Performance Management - Agent Option for HiRDB	JP1/PFM-Agent for HiRDB
JP1/Performance Management - Agent Option for Platform	JP1/PFM-Agent for Platform
JP1/Performance Management/SNMP System Observer	JP1/SSO
JP1/VERITAS NetBackup BS V4.5 Agent for HiRDB License	JP1/VERITAS NetBackup Agent for HiRDB License
JP1/VERITAS NetBackup V4.5 Agent for HiRDB License	
JP1/VERITAS NetBackup 5 Agent for HiRDB License	
LifeKeeper for Linux v9.1	LifeKeeper
SIOS DataKeeper Cluster Edition	DataKeeper
Microsoft Office Excel	Microsoft Excel または Excel
DLT(TM)	DLT
PowerBuilder(R)	PowerBuilder
JP1/Magnetic Tape Library	MTguide
JP1/VERITAS NetBackup BS v4.5	NetBackup
JP1/VERITAS NetBackup v4.5	
PowerHA SystemMirror V7.1	PowerHA
OpenTP1/Server Base Enterprise Option	TP1/EE
Hitachi System Information Capture	HSIC
Hitachi TrueCopy	TrueCopy
Hitachi TrueCopy Asynchronous	

製品名称または名称	表記	
Hitachi TrueCopy basic		
Hitachi TrueCopy Software		
TrueCopy		
TrueCopy Asynchronous		
TrueCopy remote replicator		
Hitachi Universal Replicator Software	Universal Replicator	
Universal Replicator		
Microsoft .NET Framework	.NET Framework	
Microsoft Visual C++	Visual C++またはC++言語	
Windows Server Failover Cluster	Microsoft Failover Cluster	MSFC
Microsoft Failover Cluster		
Microsoft Cluster Server		
Virtual-storage Operating System 3/Forefront System Product	VOS3/FS	VOS3
Virtual-storage Operating System 3/Leading System Product	VOS3/LS	
Virtual-storage Operating System 3/Unific System Product	VOS3/US	
Virtual-storage Operating System 3/eXtensible System Product	VOS3/XS	
VOS3 Database Connection Server	DB コネクションサーバ	
Extensible Data Manager/Base Extended Version 2 XDM 基本プログラム XDM/BASE E2	XDM/BASE E2	
XDM/Data Communication and Control Manager 3 XDM データコミュニケーションマネジメントシステム XDM/DCCM3	XDM/DCCM3	
XDM/Relational Database リレーショナルデータベースシステム XDM/RD	XDM/RD	XDM/RD
XDM/Relational Database Extended Version 2 リレーショナルデータベースシステム XDM/RD E2	XDM/RD E2	
HP-UX 11i V3 (IPF)	HP-UX または HP-UX (IPF)	
AIX V7.1	AIX V7.1	AIX

製品名称または名称	表記	
AIX V7.2	AIX V7.2	
Red Hat Enterprise Linux 6 (32-bit x86)	Linux 6	Linux
Red Hat Enterprise Linux 6 (64-bit x86_64)		
Red Hat Enterprise Linux 7 (64-bit x86_64)	Linux 7	
Red Hat Enterprise Linux 8 (64-bit x86_64)	Linux 8	
Red Hat Enterprise Linux 6 (64-bit x86_64)	Linux (EM64T)	
Red Hat Enterprise Linux 7 (64-bit x86_64)		
Red Hat Enterprise Linux 8 (64-bit x86_64)		
Red Hat Enterprise Linux 6 (32-bit x86)	Linux 6 (32-bit x86)	Linux 6
Red Hat Enterprise Linux 6 (64-bit x86_64)	Linux 6 (64-bit x86_64)	
Red Hat Enterprise Linux 7 (64-bit x86_64)	Linux 7 (64-bit x86_64)	Linux 7
Red Hat Enterprise Linux 8 (64-bit x86_64)	Linux 8 (64-bit x86_64)	Linux 8
Microsoft Windows Server 2012 Standard	Windows Server 2012 Standard	Windows Server 2012
Microsoft Windows Server 2012 Datacenter	Windows Server 2012 Datacenter	
Microsoft Windows Server 2012 R2 Standard	Windows Server 2012 R2	
Microsoft Windows Server 2012 R2 Datacenter		
Microsoft Windows Server 2016 Standard	Windows Server 2016 Standard	Windows Server 2016
Microsoft Windows Server 2016 Datacenter	Windows Server 2016 Datacenter	
Microsoft Windows Server 2019 Standard	Windows Server 2019 Standard	Windows Server 2019
Microsoft Windows Server 2019 Datacenter	Windows Server 2019 Datacenter	
Microsoft Windows Server 2022 Standard	Windows Server 2022 Standard	Windows Server 2022
Microsoft Windows Server 2022 Datacenter	Windows Server 2022 Datacenter	
Microsoft Windows 7 Home Premium	Windows 7 Home Premium	Windows 7
Microsoft Windows 7 Professional	Windows 7 Professional	

製品名称または名称	表記	
Microsoft Windows 7 Enterprise	Windows 7 Enterprise	
Microsoft Windows 7 Ultimate	Windows 7 Ultimate	
Microsoft Windows 7 Home Premium (x64)	Windows 7 (x64)	
Microsoft Windows 7 Professional (x64)		
Microsoft Windows 7 Enterprise (x64)		
Microsoft Windows 7 Ultimate (x64)		
Microsoft Windows 7 Professional (x64)	Windows 7 Professional (x64)	
Microsoft Windows 7 Enterprise (x64)	Windows 7 Enterprise (x64)	
Microsoft Windows 7 Ultimate (x64)	Windows 7 Ultimate (x64)	
Windows 8.1 (Core Edition)	Windows 8.1	
Windows 8.1 Pro		
Windows 8.1 Enterprise		
Windows 8.1 (Core Edition) (x64)		
Windows 8.1 Pro (x64)		
Windows 8.1 Enterprise (x64)		
Windows 10 Home	Windows 10	
Windows 10 Pro		
Windows 10 Enterprise		
Windows 10 Home (x64)		
Windows 10 Pro (x64)		
Windows 10 Enterprise (x64)		
Windows 11 Pro (x64)	Windows 11	
Windows 11 Enterprise (x64)		
Microsoft Windows Server 2008 Standard	Windows Server 2008 Standard	Windows (x86)
Microsoft Windows Server 2008 Enterprise	Windows Server 2008 Enterprise	
Microsoft Windows 7 Home Premium	Windows 7 Home Premium	
Microsoft Windows 7 Professional	Windows 7 Professional	

製品名称または名称	表記		
Microsoft Windows 7 Enterprise	Windows 7 Enterprise		
Microsoft Windows 7 Ultimate	Windows 7 Ultimate		
Windows 8 (Core Edition)			
Windows 8 Pro			
Windows 8 Enterprise			
Windows 8.1 (Core Edition)			
Windows 8.1 Pro			
Windows 8.1 Enterprise			
Windows 10 Home			
Windows 10 Pro			
Windows 10 Enterprise			
Microsoft Windows Server 2008 Standard (x64)	Windows Server 2008 (x64)		Windows (x64)
Microsoft Windows Server 2008 Enterprise (x64)			
Microsoft Windows Server 2008 R2 Standard (x64)	Windows Server 2008 R2		
Microsoft Windows Server 2008 R2 Enterprise (x64)			
Microsoft Windows Server 2008 R2 Datacenter (x64)			
Microsoft Windows Server 2012 Standard	Windows Server 2012		
Microsoft Windows Server 2012 Datacenter			
Microsoft Windows Server 2012 R2 Standard	Windows Server 2012 R2		
Microsoft Windows Server 2012 R2 Datacenter			
Microsoft Windows Server 2016 Standard	Windows Server 2016		
Microsoft Windows Server 2016 Datacenter			
Microsoft Windows 7 Home Premium (x64)	Windows 7 (x64)		
Microsoft Windows 7 Professional (x64)			
Microsoft Windows 7 Enterprise (x64)			
Microsoft Windows 7 Ultimate (x64)			
Windows 8 (Core Edition) (x64)			
Windows 8 Pro (x64)			
Windows 8 Enterprise (x64)			

製品名称または名称	表記
Windows 8.1 (Core Edition) (x64)	
Windows 8.1 Pro (x64)	
Windows 8.1 Enterprise (x64)	
Windows 10 Home (x64)	
Windows 10 Pro (x64)	
Windows 10 Enterprise (x64)	
Windows 11 Pro (x64)	
Windows 11 Enterprise (x64)	

- Windows Server 2016, Windows Server 2019, および Windows Server 2022 を総称して Windows Server と表記します。また, Windows Server, Windows 7, Windows 8, Windows 8.1, Windows 10, および Windows 11 を総称して Windows と表記します。
- 本文中では, Windows 8, および Windows 8.1 を分けて説明している個所以外は, Windows 8, および Windows 8.1 を総称して Windows 8 と表記します。
- HP-UX, AIX および Linux を総称して UNIX と表記します。

■ このマニュアルで使用する略語

このマニュアルで使う英略語の一覧を次に示します。

英略語	英字の表記
ADM	<u>A</u> daptable <u>D</u> ata <u>M</u> anager
ADT	<u>A</u> bstract <u>D</u> ata <u>T</u> ype
AP	<u>A</u> pplication <u>P</u> rogram
API	<u>A</u> pplication <u>P</u> rogramming <u>I</u> nterface
BES	<u>B</u> ack <u>E</u> nd <u>S</u> erver
BLOB	<u>B</u> inary <u>L</u> arge <u>O</u> bject
CD-ROM	<u>C</u> ompact <u>D</u> isc - <u>R</u> ead <u>O</u> nly <u>M</u> emory
CSMA/CD	<u>C</u> arrier <u>S</u> ense <u>M</u> ultiple <u>A</u> ccess with <u>C</u> ollision <u>D</u> etection
DAT	<u>D</u> igital <u>A</u> udio <u>T</u> aperecorder
DB	<u>D</u> atab <u>a</u> se
DBM	<u>D</u> atab <u>a</u> se <u>M</u> odule

英略語	英字の表記
DNS	<u>D</u> omain <u>N</u> ame <u>S</u> ervice
DS	<u>D</u> ictionary <u>S</u> erver
DWH	<u>D</u> ata <u>W</u> arehouse
EBCDIC	<u>E</u> xtended <u>B</u> inary <u>C</u> oded <u>D</u> ecimal <u>I</u> nterchange <u>C</u> ode
EBCDIK	<u>E</u> xtended <u>B</u> inary <u>C</u> oded <u>D</u> ecimal <u>I</u> nterchange <u>K</u> ana code
EUC	<u>E</u> xtended <u>U</u> NIX <u>C</u> ode
FD	<u>F</u> loppy <u>D</u> isk
FES	<u>F</u> ront <u>E</u> nd <u>S</u> erver
HD	<u>H</u> ard <u>D</u> isk
HNA	<u>H</u> itachi <u>N</u> etwork <u>A</u> rchitecture
IPF	<u>I</u> tanium(R) <u>P</u> rocessor <u>F</u> amily
JFS	<u>J</u> ournaled <u>F</u> ile <u>S</u> ystem
JFS2	<u>E</u> nhanced <u>J</u> ournaled <u>F</u> ile <u>S</u> ystem
JIS	<u>J</u> apanese <u>I</u> ndustrial <u>S</u> tandard code
KEIS	<u>K</u> anji processing <u>E</u> xtended <u>I</u> nformation <u>S</u> ystem
LAN	<u>L</u> ocal <u>A</u> rea <u>N</u> etwork
MGR	<u>S</u> ystem <u>M</u> anager
MIB	<u>M</u> anagement <u>I</u> nformation <u>B</u> ase
MSCS	<u>M</u> icrosoft <u>C</u> luster <u>S</u> erver
OFIS/POL	<u>O</u> ffice Automation and <u>I</u> ntelligence Support <u>S</u> oftware / <u>P</u> roblem <u>O</u> riented <u>L</u> anguage
OLAP	<u>O</u> nline <u>A</u> nalytical <u>P</u> rocessing
OS	<u>O</u> perating <u>S</u> ystem
OSI	<u>O</u> pen <u>S</u> ystems <u>I</u> nterconnection
PC	<u>P</u> ersonal <u>C</u> omputer
PDMII E2	<u>P</u> ractical <u>D</u> ata <u>M</u> anager <u>I</u> I <u>E</u> xtended <u>V</u> ersion <u>2</u>
PP	<u>P</u> rogram <u>P</u> roduct
RDB1 E2	<u>R</u> elational <u>D</u> atabase <u>M</u> anager <u>1</u> <u>E</u> xtended <u>V</u> ersion <u>2</u>
SCSI	<u>S</u> mall <u>C</u> omputer <u>S</u> ystem <u>I</u> nterface
SDS	<u>S</u> ingle <u>D</u> atabase <u>S</u> erver

英略語	英字の表記
SGML	<u>S</u> tandard <u>G</u> eneralized <u>M</u> arkup <u>L</u> anguage
SNMP	<u>S</u> imple <u>N</u> etwork <u>M</u> anagement <u>P</u> rotocol
TCP/IP	<u>T</u> ransmission <u>C</u> ontrol <u>P</u> rotocol / <u>I</u> nternet <u>P</u> rotocol
TMS-4V/SP	<u>T</u> ransaction <u>M</u> anagement <u>S</u> ystem - <u>4V</u> / <u>S</u> ystem <u>P</u> roduct
UAC	<u>U</u> ser <u>A</u> ccount <u>C</u> ontrol
UAP	<u>U</u> ser <u>A</u> pplication <u>P</u> rogram
UOC	<u>U</u> ser <u>O</u> wn <u>C</u> oding
VOS1	<u>V</u> irtual-storage <u>O</u> perating <u>S</u> ystem <u>1</u>
VOS3	<u>V</u> irtual-storage <u>O</u> perating <u>S</u> ystem <u>3</u>
WS	<u>W</u> orkstation
XDM/BASE E2	<u>E</u> xtensible <u>D</u> ata <u>M</u> anager / <u>B</u> ase <u>E</u> xtended Version <u>2</u>
XDM/DS	<u>E</u> xtensible <u>D</u> ata <u>M</u> anager / <u>D</u> ata <u>S</u> preader
XDM/RD E2	<u>E</u> xtensible <u>D</u> ata <u>M</u> anager / <u>R</u> elational <u>D</u> atabase <u>E</u> xtended Version <u>2</u>
XDM/SD E2	<u>E</u> xtensible <u>D</u> ata <u>M</u> anager / <u>S</u> tructured <u>D</u> atabase <u>E</u> xtended Version <u>2</u>
XDM/XT	<u>E</u> xtensible <u>D</u> ata <u>M</u> anager / <u>D</u> ata <u>E</u> xtract
XML	<u>E</u> xtensible <u>M</u> arkup <u>L</u> anguage
XNF/S-E2	<u>E</u> xtended HNA based communication <u>N</u> etworking <u>F</u> acility / for <u>S</u> erver - <u>E</u> xtended Version <u>2</u>

■ パス名の表記

- パス名の区切りは「/」で表記しています。Windows 版 Datareplicator を使用している場合はマニュアル中の「/」を「¥」に置き換えてください。ただし、Windows 版と UNIX 版でパス名が異なる場合は、それぞれのパス名を表記しています。
- HiRDB 運用ディレクトリのパスを\$PDDIR と表記します。ただし、Windows 版と UNIX 版でパス名が異なるため、それぞれを表記する場合、Windows 版は%PDDIR%と表記します。例を次に示します。

UNIX 版：\$PDDIR/client/lib/

Windows 版：%PDDIR%¥CLIENT¥UTL¥

■ ログの表記

●Windows 版の場合

Windows のイベントビューアで表示されるアプリケーションログをイベントログと表記します。イベントログは、次の方法で参照できます。

〈手順〉

1. [スタート] – [プログラム] – [管理ツール (共通)] – [イベントビューア] を選択します。
2. [ログ] – [アプリケーション] を選択します。

アプリケーションログが表示されます。「ソース」の列が「HiRDB Datareplicator」になっているのが HiRDB Datareplicator が出力したメッセージです。

●UNIX 版の場合

OS のログを syslogfile と表記します。syslogfile は、/etc/syslog.conf でログ出力先に指定しているファイルです。一般的には、次のファイルが syslogfile となります。

OS	ファイル
HP-UX	/var/adm/syslog/syslog.log
AIX	/var/adm/ras/syslog
Linux	/var/log/messages

■ Windows の操作説明で使う表記

Windows の操作説明で使う記号を次に示します。

記号	意味
[]	ボタンやテキストボックスなど、画面に表示されている要素を示します。
[] – []	画面に表示されるメニューやアイコンなどを選択する操作を示します。

Windows の用語「ディレクトリ」と「フォルダ」は、「ディレクトリ」に統一して表記しています。

■ 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。

●ワークステーション、端末



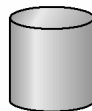
●端末での操作



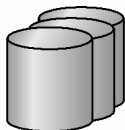
●プログラム



●ファイル 又は データベース



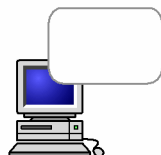
●ファイル(複数)



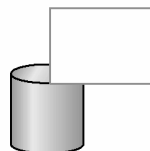
●磁気テープ



●画面の表示



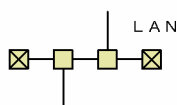
●ファイルの内容



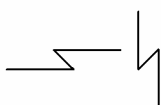
●ネットワーク WAN



●バス形の LAN



●通信回線



●プログラムの流れ



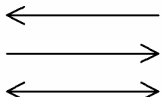
●工程、作業項目の流れ



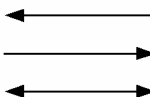
●データの流れ



●制御の流れ



●その他の流れ



■ このマニュアルで使用する記号

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

文法記述記号	意味
[]	この記号で囲んでいる項目は省略できることを示します。 (例) set hirdbusr = HiRDB 接続認可識別子 [/パスワード] これは、hirdbusr オペランドで、/パスワードを省略できることを示します。
	この記号で区切られた項目は、選択できることを示します。 (例) set keepalive = true false これは、keepalive オペランドの設定値として、true 又は false のどちらかを選択できることを示します。
{ }	この記号で囲まれている複数の項目のうち、一つを選択できることを示します。 (例) set startmode = { trn tbl spd } これは、startmode オペランドの設定値として、trn, tbl, 又は spd のどれかを選択できることを示します。
_(下線)	この下線記号で示す項目は、省略したときに仮定される値であることを示します。 (例) set keepalive = true <u>false</u> これは、keepalive オペランドを省略すると、set keepalive = false と仮定されることを示します。

文法記述記号	意 味
{{ }}…	この記号で囲まれた項目を一つの単位として、繰り返して指定できることを示します。 (例) {{ name フィールド名称 [const 初期値] }}… これは, "name フィールド名称 [const 初期値] " を複数回繰り返して指定できることを示します。
~	この記号の後に指定する値の属性を示します。
< >	指定する値の構文要素記号を示します。
《 》	値を指定しなかったときに Datareplicator が仮定する値 (省略時仮定値) を示します。
(())	指定できる値の範囲を示します。

■ このマニュアルで使用する構文要素記号

指定する値の説明で使う記号です。指定する文字列には、基本的に半角文字を使ってください。全角文字を使う場合は、半角文字数で換算してください。

構文要素記号	意 味
<英字>	アルファベット (A~Z, a~z), _ (下線), - (ハイフン)
<英字記号>	アルファベット (A~Z, a~z), #, @, ¥
<英数字>	<英字>, 数字 (0~9)
<英数字記号>	<英字記号>, 数字 (0~9)
<符号なし整数>	数字 (0~9)
<16進数字>	数字 (0~9), A~F, a~f
<識別子>	先頭がアルファベット (A~Z, a~z) の英数字
<記号名称>	先頭が<英字記号>の<英数字記号>
<文字列>	任意の文字の配列
<パス名>	記号名称, ¥又は/, 及び. (ピリオド)
<ファイル名>	アルファベット (A~Z, a~z), 数字 (0~9), . (ピリオド), _ (下線), - (ハイフン), 及び@で構成される文字列 (最大 30 文字) 注 すべての半角文字で指定してください。UNIX の場合は、英字の大文字と小文字は区別されます。 パス名は、Datareplicator をインストールする OS の規則に依存します。

■ このマニュアルで使用する計算式の記号

このマニュアルで使用する計算式の記号の意味を次に示します。

記号	意味
↑ ↑	計算結果の値を小数点以下で切り上げることを示します。 (例) $\uparrow 34 \div 3 \uparrow$ の計算結果は「12」になります。
↓ ↓	計算結果の値を小数点以下で切り下げることを示します。 (例) $\downarrow 34 \div 3 \downarrow$ の計算結果は「11」になります。
MAX	計算結果の最も大きい値を選ぶことを示します。 (例) MAX (3×6, 4 + 7) の計算結果は「18」になります。
MIN	計算結果の最も小さい値を選ぶことを示します。 (例) MIN (3×6, 4 + 7) の計算結果は「11」になります。

■ 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。

■ HiRDB Version 9 と HiRDB Version 10 の製品体系の違い

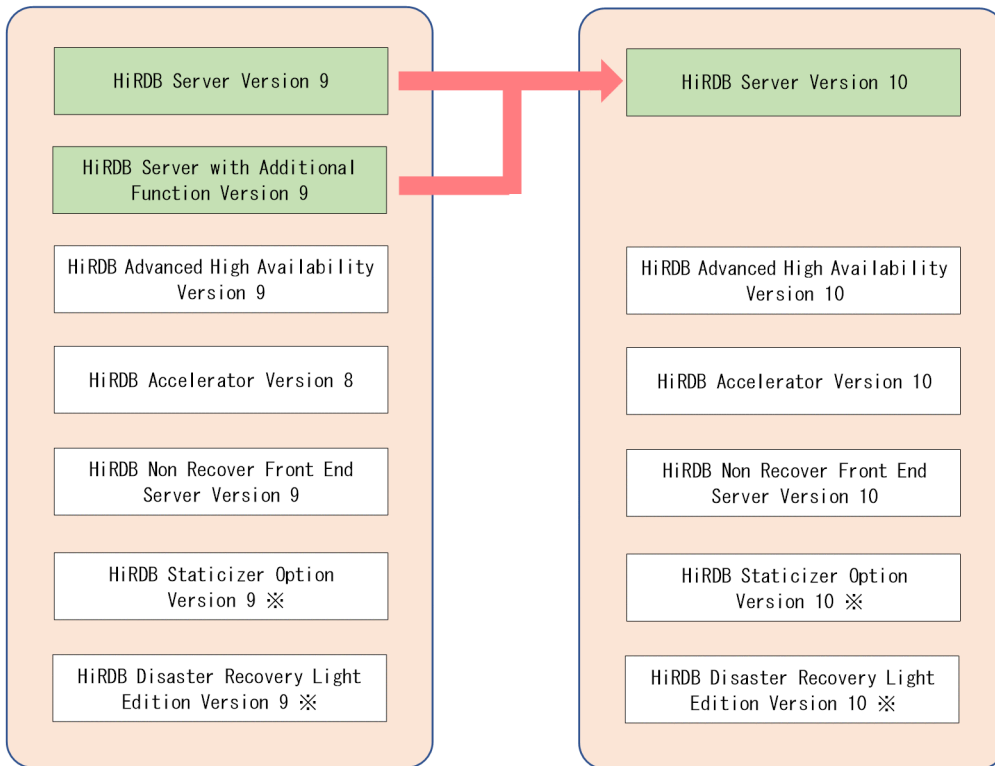
HiRDB Version 10 では、製品体系を次のように変更しました。

- HiRDB Server with Additional Function を廃止し、HiRDB Server with Additional Function の機能を HiRDB Server に統合しました。

HiRDB Version 9 と HiRDB Version 10 の製品体系の違いを次に示します。

●HiRDB Version 9

●HiRDB Version 10



注※ UNIX 版でだけ使用できる製品です。

目次

前書き	2
変更内容	4
はじめに	6

1	概要	36
1.1	特長	37
1.1.1	Datareplicator の目的	37
1.1.2	データ連動システム	39
1.2	データ連動について	43
1.2.1	データ連動システムの組み合わせ	43
1.2.2	データ連動の仕組み	43
1.2.3	抽出側 DB と反映側 DB との対応	49
1.2.4	データ連動できるデータベース	49
1.2.5	データベース用語の対応	52
1.3	データ連動で使える表	53
1.4	データ連動で使える表のデータ型	55
1.4.1	Datareplicator で使えるデータ型	55
1.4.2	抽象データ型を使った表のデータ連動	56
1.4.3	繰返し列を使った表のデータ連動	57
1.4.4	BLOB 型と BINARY 型を使った表のデータ連動	58
1.5	データ連動できる SQL	61
1.6	ソフトウェア構成	62
1.6.1	HiRDB から HiRDB へデータ連動する場合のソフトウェア構成	62
1.6.2	HiRDB からメインフレーム側 DB へデータ連動する場合のソフトウェア構成	64
1.6.3	メインフレーム側 DB から HiRDB へデータ連動する場合のソフトウェア構成	65
1.6.4	SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動する場合のソフトウェア構成	67
1.7	データ連動システムの構築手順	69
2	環境の設定	70
2.1	Datareplicator に関連する製品	71
2.1.1	Datareplicator に必要な製品	71
2.1.2	抽出側システムに必要な製品	72
2.1.3	反映側システムに必要な製品	73
2.2	Datareplicator のインストール (UNIX)	74

2.2.1	インストール前の準備	74
2.2.2	インストールするサーバマシン	74
2.2.3	インストールの実行	75
2.2.4	Datareplicator のアンインストール	75
2.3	ディレクトリ構成 (UNIX)	76
2.3.1	Datareplicator のインストール時に作成されるディレクトリ	76
2.3.2	抽出側 Datareplicator のディレクトリ構成	77
2.3.3	反映側 Datareplicator のディレクトリ構成	80
2.4	環境変数の設定 (UNIX)	83
2.4.1	抽出側 Datareplicator の環境変数	83
2.4.2	反映側 Datareplicator の環境変数	86
2.5	通信環境の設定 (UNIX)	91
2.5.1	抽出側 Datareplicator の通信環境の設定	91
2.5.2	反映側 Datareplicator の通信環境の設定	93
2.6	Datareplicator のインストール (Windows)	95
2.6.1	インストール前の準備	95
2.6.2	インストールするサーバマシン	95
2.6.3	インストールの実行	96
2.6.4	インストール時に登録される内容	97
2.6.5	インストール完了後に必要な作業	98
2.6.6	Datareplicator のアンインストール	99
2.7	ディレクトリ構成 (Windows)	100
2.7.1	Datareplicator のインストール時に作成されるディレクトリ	100
2.7.2	抽出側 Datareplicator のディレクトリ構成	101
2.7.3	反映側 Datareplicator のディレクトリ構成	104
2.8	環境変数の設定 (Windows)	107
2.8.1	抽出側 Datareplicator の環境変数	107
2.8.2	反映側 Datareplicator の環境変数	108
2.9	通信環境の設定 (Windows)	112
2.9.1	サービス名の登録	112
2.9.2	ホスト名の登録	113
2.9.3	Windows Terminal Service の使用	113
2.10	Windows Vista, 及び Windows Server 2008 の場合の動作環境	115
2.10.1	コマンドの実行	115
2.10.2	コマンド名の変更	115
2.10.3	JIS 第 3 水準・第 4 水準の文字のサポート	116
2.11	Datareplicator のバージョンアップ	117
2.11.1	バージョンアップ時の注意	117
2.11.2	バージョンアップの手順	117

2.11.3 バージョンアップ時の注意 117

3 データ連動機能 119

3.1 連動パターン 120

3.1.1 同一形式の表にデータ連動するパターン 120

3.1.2 別形式の表にデータ連動するパターン 120

3.1.3 1 表から n 表にデータ連動するパターン 122

3.1.4 n 表から 1 表にデータ連動するパターン 122

3.1.5 送信行を選択してデータ連動するパターン 123

3.1.6 ユーザOWNコーディングを使ってデータ連動するパターン 123

3.1.7 更新情報の履歴を時系列順に取得するパターン 124

3.2 抽出側 Datareplicator の抽出処理 125

3.2.1 抽出処理の概要 125

3.2.2 抽出処理で使うファイルとプロセス 126

3.2.3 抽出処理の単位 131

3.2.4 抽出側 Datareplicator の情報の取得 132

3.2.5 送信プロセス数の制御 133

3.2.6 ユーザOWNコーディングを使った更新情報のチェック 134

3.2.7 メッセージ出力の抑止 135

3.2.8 連結演算による更新データの連動 136

3.2.9 インナレプリカ機能を使った RD エリアのデータ連動 137

3.3 反映側 Datareplicator の反映処理 141

3.3.1 反映処理の概要 141

3.3.2 反映処理で使うファイルとプロセス 142

3.3.3 反映処理の方式 146

3.3.4 マルチ FES 機能に対応した反映方式 149

3.3.5 反映処理の単位 149

3.3.6 トリガが定義されている表のデータ連動 150

3.3.7 時系列情報の取得 151

3.3.8 反映側 Datareplicator の情報の取得 151

3.3.9 ユーザOWNコーディングを使った更新情報の加工 152

3.3.10 反映時エラーのスキップ 152

3.3.11 マージ表の作成 156

3.3.12 反映処理の同期点処理の指定 158

3.3.13 メッセージ出力の抑止 159

3.3.14 排他エラー時の動作 160

3.4 Datareplicator ファイルシステム領域 161

3.4.1 Datareplicator ファイルシステム領域の目的 161

3.4.2 Datareplicator ファイルシステム領域の作成方法 163

3.4.3	Datareplicator ファイルシステム領域の構造	164
3.4.4	Datareplicator ファイルシステム領域を使うときの注意	167
3.5	滞留監視機能	168
3.5.1	滞留監視機能の概要	168
3.5.2	滞留監視機能の使用	169
3.5.3	注意事項	171
3.6	反映トランザクション同期機能	173
3.6.1	反映トランザクション同期機能の概要	173
3.6.2	反映トランザクション同期機能の準備	173
3.6.3	反映トランザクション同期機能の処理の流れ	177
3.6.4	トランザクションブランチ情報	179
3.6.5	同期反映グループ	180
3.6.6	抽出状態・反映状態の確認方法	183
3.6.7	障害運用	184
3.6.8	注意事項	185
3.6.9	運用例	187
3.7	イベント機能	189
3.7.1	イベントの発行	189
3.7.2	イベントの種類	190
3.7.3	イベントコードの定義	190
3.7.4	イベントを検知するタイミングと動作	191
3.7.5	注意事項	194
3.8	ファイルの二重化	196
3.8.1	二重化できるファイル	196
3.8.2	二重化に関連するファイル	197
3.8.3	二重化する場合の注意事項	198
4	システム設計	200
4.1	システム設計の項目	201
4.2	連動パターンの設計	203
4.2.1	同一形式の表にデータ連動するパターン	204
4.2.2	抽出側 DB のすべての表を同一形式の表にデータ連動するパターン	206
4.2.3	別形式の表にデータ連動するパターン	207
4.2.4	1 表から n 表にデータ連動するパターン	212
4.2.5	n 表から 1 表にデータ連動するパターン	215
4.2.6	送信行を選択してデータ連動するパターン	217
4.2.7	ユーザOWNコーディングを使ってデータ連動するパターン	218
4.2.8	更新情報の履歴を時系列順に取得するパターン	219
4.3	抽出側 DB と反映側 DB との対応の設計	221

- 4.3.1 抽出側 DB の条件 221
- 4.3.2 反映対象表の作成 223
- 4.3.3 マッピングキーの対応の設計 224
- 4.3.4 データ型の対応の設計 227
- 4.3.5 文字コードの対応の設計 235
- 4.3.6 繰返し列の設計 250
- 4.3.7 時系列情報表の作成 252
- 4.4 抽出側システムと反映側システムとの対応の設計 256
- 4.4.1 HiRDB から HiRDB へデータ連動する場合の設計 256
- 4.4.2 HiRDB からメインフレーム側 DB へデータ連動する場合の設計 261
- 4.4.3 メインフレーム側 DB から HiRDB へデータ連動する場合の設計 265
- 4.4.4 SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動する場合の設計 267
- 4.5 データ連動システムの形態の設計 270
- 4.5.1 データ連動システムの形態の種類 270
- 4.5.2 データ連動システムの階層型システムへの応用形態の設計 271
- 4.5.3 複数システム間でのデータ連動システムの注意 272
- 4.6 抽出側 Datareplicator の設計 278
- 4.6.1 抽出側 Datareplicator のファイル構成 278
- 4.6.2 抽出側 Datareplicator で使うファイルの準備 280
- 4.6.3 抽出方法の設計 285
- 4.6.4 送信方法の設計 287
- 4.6.5 抽出処理の開始方式の設計 292
- 4.6.6 抽出処理の停止方式の設計 294
- 4.6.7 イベント制御用テーブルの設計 295
- 4.6.8 抽出側 Datareplicator のリソースの設計 299
- 4.7 反映側 Datareplicator の設計 321
- 4.7.1 反映側 Datareplicator のファイル構成 321
- 4.7.2 反映側 Datareplicator で使うファイルの準備 322
- 4.7.3 反映方法の設計 326
- 4.7.4 反映処理の開始方式の設計 336
- 4.7.5 反映処理の停止方式の設計 339
- 4.7.6 反映処理方式の切り替えの設計 343
- 4.7.7 反映側 Datareplicator のリソースの設計 344
- 4.8 抽出側 HiRDB の設計 361
- 4.8.1 抽出側 Datareplicator が使うディレクトリ名の指定 361
- 4.8.2 HiRDB Datareplicator 連携の開始指定 361
- 4.8.3 HiRDB Datareplicator 連携を開始した後の注意事項 362

5	定義 363	
5.1	Datareplicator の定義の概要	364
5.1.1	Datareplicator の定義の構成	364
5.1.2	定義の規則	367
5.2	抽出システム定義	372
5.2.1	形式	372
5.2.2	定義内容の変更	374
5.2.3	オペランドの説明	377
5.3	抽出環境定義	395
5.3.1	形式	395
5.3.2	定義内容の変更	396
5.3.3	オペランドの説明	397
5.4	送信環境定義	404
5.4.1	形式	404
5.4.2	定義内容の変更	405
5.4.3	オペランドの説明	407
5.5	抽出定義	416
5.5.1	構造と形式	416
5.5.2	定義内容の変更	417
5.5.3	抽出定義文	418
5.5.4	送信定義文	420
5.5.5	全表抽出定義文	424
5.6	抽出側 HiRDB の定義	427
5.6.1	システム共通定義	427
5.6.2	ユニット制御情報定義	428
5.7	二重化定義（抽出側）	430
5.7.1	二重化定義（抽出側）の形式と説明	430
5.8	反映システム定義	431
5.8.1	形式	431
5.8.2	定義内容の変更	432
5.8.3	オペランドの説明	434
5.9	反映環境定義	449
5.9.1	形式	449
5.9.2	定義内容の変更	451
5.9.3	オペランドの説明	454
5.10	反映定義	474
5.10.1	定義の規則	474
5.10.2	構造と形式	475
5.10.3	定義内容の変更	477

- 5.10.4 更新情報フィールド定義 477
- 5.10.5 反映表定義 480
- 5.10.6 反映グループ定義 486
- 5.11 更新情報定義 494
- 5.11.1 構造と形式 494
- 5.11.2 稼働環境定義文 495
- 5.11.3 抽出再定義文 495
- 5.11.4 抽出文 497
- 5.12 二重化定義（反映側） 500
- 5.12.1 二重化定義（反映側）の形式と説明 500
- 5.13 Datareplicator の定義例 501
- 5.13.1 システムの対応とファイルの構成例 501
- 5.13.2 抽出側 Datareplicator の定義の例 503
- 5.13.3 反映側 Datareplicator の定義の例 506
- 5.13.4 抽出定義と反映定義の例 508
- 5.13.5 マルチ FES 機能を使う場合の反映グループ定義の例 514

6 運用 519

- 6.1 データ連動システムの運用の概要 520
- 6.1.1 通常のデータ連動システムの運用手順 520
- 6.1.2 SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動する場合の運用手順 522
- 6.2 環境構築時の初期化手順 525
- 6.3 抽出側 Datareplicator の起動と終了 529
- 6.3.1 抽出側 Datareplicator の起動 529
- 6.3.2 抽出側 Datareplicator の終了 531
- 6.4 抽出側 Datareplicator の運用 533
- 6.4.1 抽出処理の運用 533
- 6.4.2 抽出側 Datareplicator で使うファイルの運用 537
- 6.4.3 抽出側 Datareplicator 運用時の注意 545
- 6.5 抽出側 HiRDB の運用 548
- 6.5.1 HiRDB Datareplicator 連携の開始・終了・中止 548
- 6.5.2 データ連動用連絡ファイル 549
- 6.5.3 システムログファイルの運用 549
- 6.5.4 抽出環境定義の extsuppress の指定と、抽出側 HiRDB 及び抽出側 Datareplicator での処理 555
- 6.5.5 HiRDB Datareplicator 連携の実行状況の確認 556
- 6.5.6 抽出側 HiRDB の運用手順 557
- 6.5.7 syncterm=true を設定したときの運用上の注意 570
- 6.5.8 sendcontrol=sendmst を設定したときの運用上の注意 571
- 6.6 反映側 Datareplicator の起動と終了 572

- 6.6.1 反映側 Datareplicator の起動 572
- 6.6.2 反映側 Datareplicator の終了 574
- 6.7 反映側 Datareplicator の運用 576
 - 6.7.1 反映処理の運用 576
 - 6.7.2 反映側 Datareplicator で使うファイルの運用 581
 - 6.7.3 反映側 Datareplicator 運用時の注意 588
- 6.8 HiRDB 及び Datareplicator の構成変更 592
 - 6.8.1 連動対象表の定義変更 593
 - 6.8.2 抽出側システムの構成変更 596
 - 6.8.3 反映側システムの構成変更 598
 - 6.8.4 反映トランザクション同期機能を使用している場合の構成変更 600
- 6.9 系切り替え機能を使うときの運用 605
 - 6.9.1 系切り替え機能の形態 606
 - 6.9.2 系切り替え機能を使うときの準備 (HA モニタの場合) 608
 - 6.9.3 系切り替え機能を使うときの準備 (Microsoft Cluster Server の場合) 616
 - 6.9.4 系切り替え機能を使うときの運用手順 (HA モニタの場合) 618
 - 6.9.5 系切り替え機能を使うときの注意 619
 - 6.9.6 影響分散スタンバイレス型系切り替え機能での運用 622
 - 6.9.7 系切り替え機能を使うときの準備 (Amazon EC2 環境の HA モニタの場合) 631
- 6.10 ファイルの二重化を使うときの運用 633
 - 6.10.1 ファイルの一重化から二重化への変更 633
 - 6.10.2 二重化するファイルの変更 634
 - 6.10.3 ファイルの二重化から一重化への変更 636
- 6.11 ラージファイルの運用 637
 - 6.11.1 ラージファイル運用の準備 (UNIX 版限定) 637
 - 6.11.2 ラージファイル運用時のコマンド実行時間の見積もり 640
- 6.12 チューニング 643
 - 6.12.1 チューニングの要否 643
 - 6.12.2 チューニングの方法 644
- 6.13 運用時の注意事項 646
 - 6.13.1 OS の時刻変更についての注意事項 646
 - 6.13.2 Datareplicator が作成するファイルのバックアップについて 646

7 コマンドの文法 647

- コマンドの概要 648
- hdechgstatus (抽出側 Datareplicator の状態変更) 653
- hdeevent (抽出側 Datareplicator でのイベント発行) 654
- hdefcopy (抽出側現用ファイルの複写) 656
- hdefstate (抽出側二重化ファイルの状態表示) 657
- hdeinfoget (抽出側障害情報の取得) 659

hdemodq (抽出情報キューファイルの構成変更)	662
hdeprep (抽出定義プリプロセスファイルの作成)	665
hdeshmclean (抽出側 Datareplicator の共用リソースの削除)	667
hdestart (抽出側 Datareplicator の起動)	672
hdestart_n (抽出側 Datareplicator の部分起動)	682
hdestate (抽出側 Datareplicator のステータス情報の取得)	687
hdestop (抽出側 Datareplicator の終了)	695
hdestop_n (抽出側 Datareplicator の部分終了)	698
hdscnvedt (文字コード変換用マッピングテーブルの編集)	701
hdschgstatus (反映側 Datareplicator の状態変更)	706
hdsemergencystop (反映側 Datareplicator の強制終了)	707
hdsfcopy (反映側現用ファイルの複写)	708
hdsfmkfs (Datareplicator ファイルシステム領域の初期化)	709
hdsfstate (反映側二重化ファイルの状態表示)	711
hdsfstats (Datareplicator ファイルシステム領域の状態の表示)	713
hdsinfoget (反映側障害情報の取得)	716
hdsrefinm (更新情報内容の確認)	719
hdsrctl (反映処理の制御)	724
hdssamqin (SAM ファイルからの更新情報の抽出)	727
hdsshmclean (反映側 Datareplicator の共用リソースの削除)	729
hdsstart (反映側 Datareplicator の起動)	732
hdsstate (反映側 Datareplicator のステータス情報の取得)	738
hdsstop (反映側 Datareplicator の終了)	747
hdstrcredit (稼働トレースファイルの編集)	751
pdlogchg (ログ関係のファイルのステータスの変更)	766
pdls (HiRDB システムの状態の表示)	767
pdrplstart (HiRDB Datareplicator 連携の開始)	769
pdrplstop (HiRDB Datareplicator 連携の中止)	770

8 ユーザOWNコーディング 771

8.1 反映情報編集 UOC	772
8.1.1 反映情報編集 UOC の概要	772
8.1.2 反映情報編集 UOC の作成手順 (UNIX)	774
8.1.3 反映情報編集 UOC の作成手順 (Windows)	778
8.1.4 反映情報編集 UOC で使う関数の文法	780
8.1.5 反映情報編集 UOC での抽象データ型の扱い	794
8.1.6 反映情報編集 UOC を作成するときの注意	800
8.1.7 反映情報編集 UOC のサンプル	803
8.2 列データ編集 UOC	807
8.2.1 列データ編集 UOC の概要	807
8.2.2 列データ編集 UOC の作成手順 (UNIX)	808
8.2.3 列データ編集 UOC の作成手順 (Windows)	811
8.2.4 列データ編集 UOC で使う関数の文法	812

- 8.2.5 列データ編集 UOC を作成するときの注意 819
- 8.2.6 列データ編集 UOC のサンプル 820
- 8.3 送信データ UOC 822
- 8.3.1 送信データ UOC の概要 822
- 8.3.2 送信データ UOC の作成手順 (UNIX) 823
- 8.3.3 送信データ UOC の作成手順 (Windows) 826
- 8.3.4 送信データ UOC で使う関数の文法 827
- 8.3.5 送信データ UOC を作成するときの注意 834
- 8.3.6 送信データ UOC のサンプル 836

9 障害時の対処 837

- 9.1 抽出側 Datareplicator の障害時の対処 838
 - 9.1.1 障害の対処の手順 838
 - 9.1.2 エラーの対処方法 840
 - 9.1.3 エラー対処後の処置 843
 - 9.1.4 反映側システムでエラーが発生したときの対処 844
 - 9.1.5 ユーザOWNコーディングの障害時の対処 845
 - 9.1.6 ファイルの二重化運用時のファイル障害への対処 845
- 9.2 反映側 Datareplicator の障害時の対処 847
 - 9.2.1 障害の対処の手順 847
 - 9.2.2 エラーの対処方法 848
 - 9.2.3 エラー対処後の処置 851
 - 9.2.4 抽出側システムでエラーが発生したときの対処 852
 - 9.2.5 ユーザOWNコーディングの障害時の対処 853
- 9.3 障害回復方法の選択基準 854
- 9.4 障害回復時の初期化手順 857
 - 9.4.1 Datareplicator を初期化する必要がある障害 857
 - 9.4.2 障害回復時の Datareplicator の初期化手順 857
- 9.5 システムログファイルによるデータ連動回復 860
 - 9.5.1 システムログファイルによるデータ連動回復の概要 860
 - 9.5.2 システムログファイルによるデータ連動回復の前提条件 861
 - 9.5.3 システムログファイルによる回復手順 865
- 9.6 アンロードログファイルによるデータ連動回復 869
 - 9.6.1 アンロードログファイルによるデータ連動回復の概要 869
 - 9.6.2 アンロードログファイルによるデータ連動回復の前提条件 870
 - 9.6.3 アンロードログファイルによるデータ連動回復を使う前の準備 873
 - 9.6.4 アンロードログファイルによる回復手順の概要 874
 - 9.6.5 アンロードログファイルによるデータ連動回復手順の詳細 878
 - 9.6.6 データ連動回復機能のコマンド 887

- 9.6.7 データ連動回復機能を実行した後の作業 892
- 9.6.8 反映側システムで表単位反映方式で反映処理する場合の反映抑止方法 892
- 9.7 抽出情報キューファイル回復機能 895
 - 9.7.1 抽出情報キューファイル回復機能の概要 895
 - 9.7.2 抽出情報キューファイル回復機能の前提条件 896
 - 9.7.3 抽出情報キューファイル回復機能の回復手順 896
- 9.8 反映障害による未送信情報の取得（更新 SQL 出力機能） 901
 - 9.8.1 使用方法 901
 - 9.8.2 更新 SQL ファイル 902
 - 9.8.3 操作手順 906
 - 9.8.4 注意事項 908

10 メッセージの一覧 909

- 10.1 メッセージの概要 910
 - 10.1.1 メッセージの出力先 910
 - 10.1.2 メッセージの出力形式 911
 - 10.1.3 メッセージの記述形式 912
- 10.2 メッセージの詳細 916
- 10.3 システムコールエラー一覧 1222
- 10.4 要因コード一覧 1224
- 10.5 SQL コードの詳細 1228

付録 1230

- 付録 A HiRDB との対応状況 1231
 - 付録 A.1 HiRDB の適用 OS と使用できる Datareplicator のバージョン 1231
 - 付録 A.2 HiRDB の機能と Datareplicator でのサポート状況 1233
- 付録 B Datareplicator 定義の予約語 1241
- 付録 C UNIX 版 Datareplicator と Windows 版 Datareplicator の機能差 1245
- 付録 D Datareplicator のバージョンダウン 1246
 - 付録 D.1 製品形名によるバージョンダウン手順の違い 1246
 - 付録 D.2 バージョンダウンの手順 1246
 - 付録 D.3 バージョンダウン後の注意 1246
- 付録 E 用語解説 1248

索引 1257

1

概要

この章では、Datareplicator の特長、データ連動の概要、ソフトウェア構成、及びデータ連動システムの構築手順について説明します。

1.1 特長

HiRDB Datareplicator（以降、Datareplicator と表記します）は、HiRDB システムのデータベースに、ほかのデータベースの更新内容を自動的に反映できるようにする製品です。また、HiRDB システムのデータベースの更新内容を、ほかのシステムのデータベースに自動的に反映することもできます。

1.1.1 Datareplicator の目的

Datareplicator の目的について説明します。

(1) 分散配置したデータベースの課題

これまで、企業活動で発生する大量のデータは、メインフレーム側のデータベースに蓄積されてきました。近年はワークステーション（WS）やパーソナルコンピュータ（PC）上で稼働する DBMS（HiRDB）を使って、メインフレーム側のデータを複数の WS や PC に分散して配置する形態が広がっています。このようなシステム形態が広まるとともに、分散配置したデータベースをどのように統合して活用するかが重要な課題になっています。

(2) レプリケーション機能の概要

分散配置したデータベースの内容をほかのデータベースに反映する機能を、**レプリケーション機能**といいます。レプリケーション機能を使うと、一つのシステムのデータベースの情報をほかのシステムのデータベースに反映して、分散システム環境でのデータ管理を支援できます。

レプリケーション機能には、次の 2 種類があります。

データ連動機能

データベースの更新に連動してデータを抽出し、データベースの更新内容をほかのシステムのデータベースに自動的に反映します。ほかのシステムのデータベースに自動的に反映することを**データ連動**といい、データ連動機能を備えたデータベースシステムを**データ連動システム**といいます。

データ連動機能は、データベースの最新データをほかのシステムのデータベースで利用したり、自動的にバックアップを取得したりするのに適しています。

データベース抽出・反映サービス機能

データベースのデータを一括して抽出し、ほかのシステムのデータベースに反映します。

データ抽出・反映サービス機能は、大量データを一括で抽出して格納したり、ほかのシステムのデータベースを初期作成したり再作成したりするのに適しています。

HiRDB のデータベースとメインフレームのデータベースで、レプリケーション機能を実行するために必要な製品を次の表に示します。

表 1-1 レプリケーション機能を実行するために必要な製品

レプリケーション機能の種類	HiRDB のデータ連動製品	メインフレーム DB のデータ連動製品
データ連動機能	HiRDB Datareplicator	XDM/DS*2
データ抽出・反映サービス機能	HiRDB Dataextractor*1	XDM/XT*3

注※1

HiRDB Dataextractor の詳細については、マニュアル「データベース抽出・反映サービス機能 HiRDB Dataextractor」を参照してください。

注※2

XDM/DS の詳細については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

注※3

XDM/XT の詳細については、マニュアル「VOS3 データベース抽出プログラム XDM/XT」を参照してください。

(3) Datareplicator のデータ連動システムの利点

複数のデータベース間でデータを連動させる方法には、一方のデータベースの更新と同期して、そのたびにほかのデータベースを更新する方法もあります。しかし、この方法では同期するための処理や障害時の回復などシステムへの負荷が非常に大きくなります。

これに対して、Datareplicator のデータ連動ではデータベースの更新情報を抽出して、抽出した更新情報を基に非同期にほかのシステムのデータベースを更新するため、システムの負荷を小さくできます。

Datareplicator によるデータ連動は、分散配置しているデータベースの厳密な同期よりも、システム全体の負荷の軽減や性能を重視するシステムの場合に有効です。

そのほかに、Datareplicator のデータ連動機能を使うと次に示す利点があります。

- 更新内容は Datareplicator が自動的に抽出するので、アプリケーションの変更は必要ありません。
- 更新したデータだけを転送するので、転送データ量が最小限で済み、基幹システムに負担を掛けません。
- 更新内容を反映する表や列を選んだり、更新内容を時系列で蓄積したりできるので、データウェアハウスの利用者に応じた使いやすい表を作成できます。

(4) Datareplicator のデータ連動システムの用語

このマニュアルで使用するデータ連動システムの用語を次の表に示します。

表 1-2 データ連動システムの用語

データ連動システムの用語	用語の意味
抽出側システム	反映させる更新内容を抽出するシステム（連動元）
抽出側 Datareplicator	抽出側システムの Datareplicator
抽出側 DB*	抽出側システムのデータベース
反映側システム	抽出された更新内容を反映するシステム（連動先）
反映側 Datareplicator	反映側システムの Datareplicator
反映側 DB*	反映側システムのデータベース

注※

抽出側 DB 又は反映側 DB をシステムごとに区別するときは、HiRDB の DB を抽出側 HiRDB 又は反映側 HiRDB といい、メインフレームの DB をメインフレーム側 DB といいます。

1.1.2 データ連動システム

(1) データ連動できるシステム

HiRDB を使うときは、次に示す組み合わせでデータ連動できます。

- HiRDB から HiRDB へのデータ連動
- HiRDB からメインフレーム側 DB へのデータ連動
- メインフレーム側 DB から HiRDB へのデータ連動
- SAM ファイルを使った、メインフレーム側 DB から HiRDB へのデータ連動

データ連動システムの組み合わせの詳細については、「1.2.1 データ連動システムの組み合わせ」を参照してください。

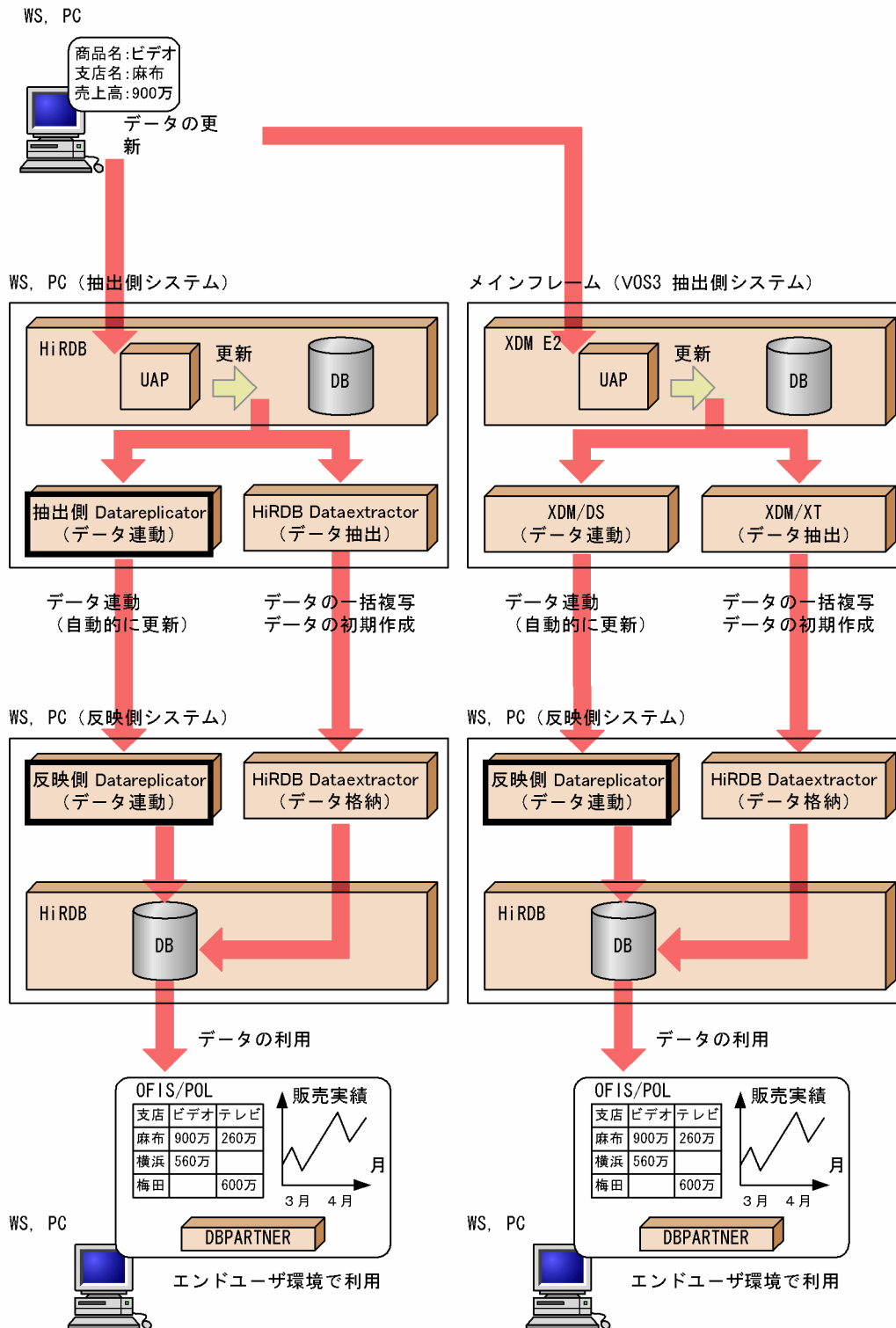
(2) データ連動を適用できる業務システム

データ連動は、次に示す業務システムの運用形態に適用できます。

- 複数部門によって運用される基幹業務（メインフレーム又はサーバに構築）同士のデータ交換
- 基幹業務の履歴を保存するシステムへのデータ追加
- 基幹業務から部門システムへのデータ抽出
- 部門システム間のデータ交換
- 部門システムのバックアップデータの抽出

データ連動を適用した業務システムの例を次の図に示します。

図 1-1 データ連動を適用した業務システムの例



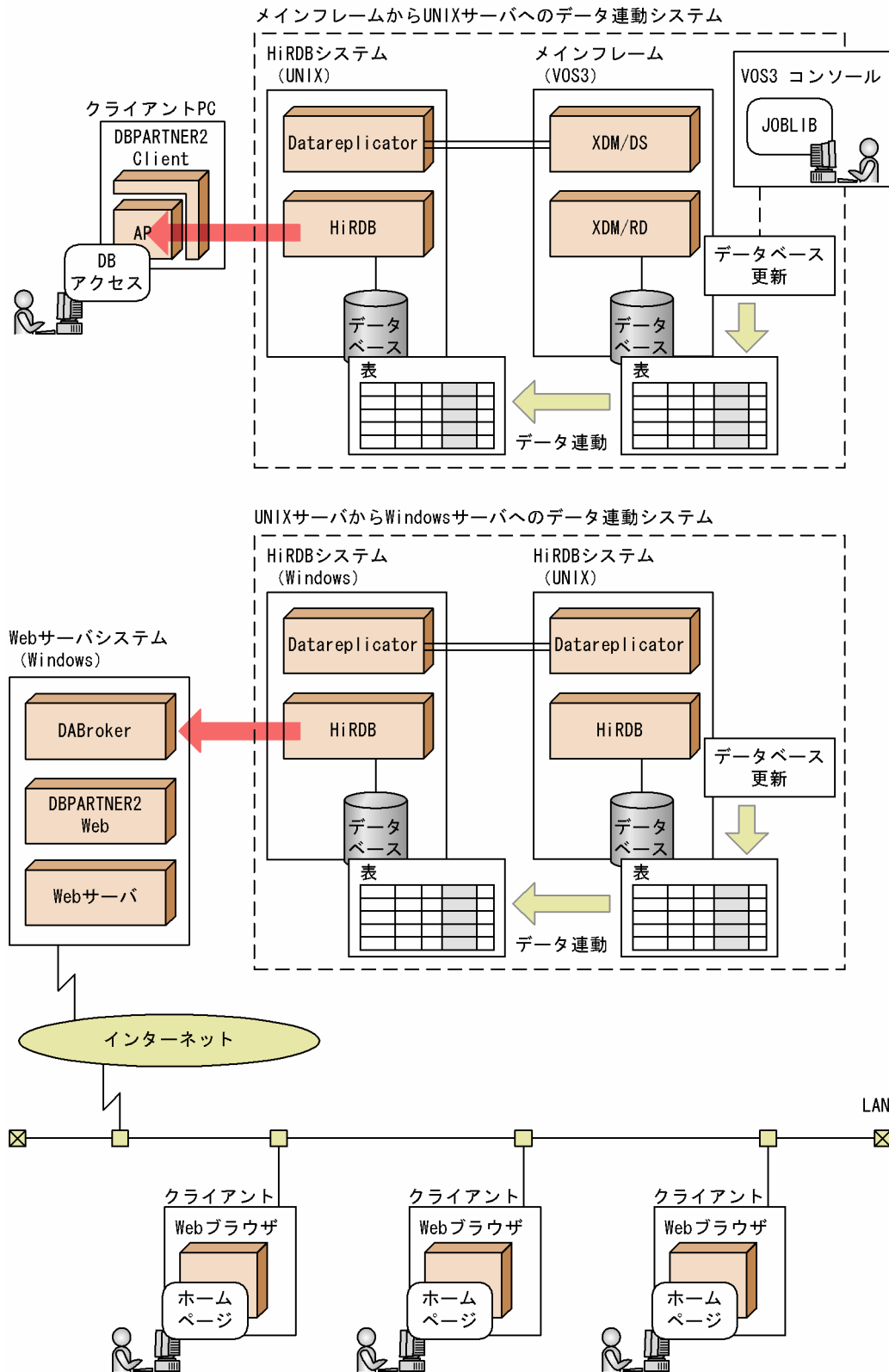
(3) データウェアハウスへの適用

HiRDB のレプリケーション機能を使うと、HiRDB 同士でデータを抽出/反映するだけでなく、メインフレームの DB (資産要素が強いレガシーデータ) を HiRDB に反映して活用できます。これらのレプリケーション機能の特長を生かして、複数のデータベースを連携した統合的なデータベースシステム (データウェア

アハウス)を構築できます。データウェアハウスでは、OLAP ツールを使ってデータベースシステムを分析したり、エンドユーザがPCを使ってHiRDBにデータ連動したデータにアクセスしたりできます。

レプリケーション機能を使ってHiRDBをデータウェアハウスに適用する例を次の図に示します。次の図の例は、基幹業務のデータベースをDatareplicatorでデータ連動し、そのデータをパーソナルコンピュータから参照する形態です。データベースシステムの分析と意思決定システムなどのデータウェアハウスの要素は省略しています。

図 1-2 レプリケーション機能を使ってHiRDBをデータウェアハウスに適用する例



1.2 データ連動について

データ連動システムは、抽出側 Datareplicator と反映側 Datareplicator、又はこれらの製品と XDM/DS で構成されます。

1.2.1 データ連動システムの組み合わせ

Datareplicator を使ってデータ連動できる抽出側 DB と反映側 DB の組み合わせを次の表に示します。

表 1-3 Datareplicator を使ってデータ連動できる抽出側 DB と反映側 DB の組み合わせ

抽出側 DB (適用 OS*)	反映側 DB (適用 OS*)
HiRDB (UNIX, Windows)	HiRDB (UNIX, Windows) XDM/RD E2 (VOS3)
XDM/SD E2 (VOS3)	HiRDB (UNIX, Windows)
XDM/RD E2 (VOS3)	
ADM (VOS3)	
PDMII E2 (VOS3, VOS1)	
TMS-4V/SP (VOS3)	
RDB1 E2 (VOS1)	

注

メインフレーム同士でデータ連動するシステムの組み合わせについては、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」又は各データベースのマニュアルを参照してください。

注※

各データベース製品は、ここで示す適用 OS 以外の OS では使えません。

1.2.2 データ連動の仕組み

データ連動システムの組み合わせ別に、データ連動の仕組みについて説明します。

(1) HiRDB から HiRDB へのデータ連動

HiRDB から HiRDB へのデータ連動の仕組みを説明します。

抽出側 DB が更新されると、抽出側 Datareplicator が更新情報 (UAP がデータベースを更新した情報) を HiRDB のシステムログファイルから抽出して、抽出情報キューファイルに格納します。これを抽出処

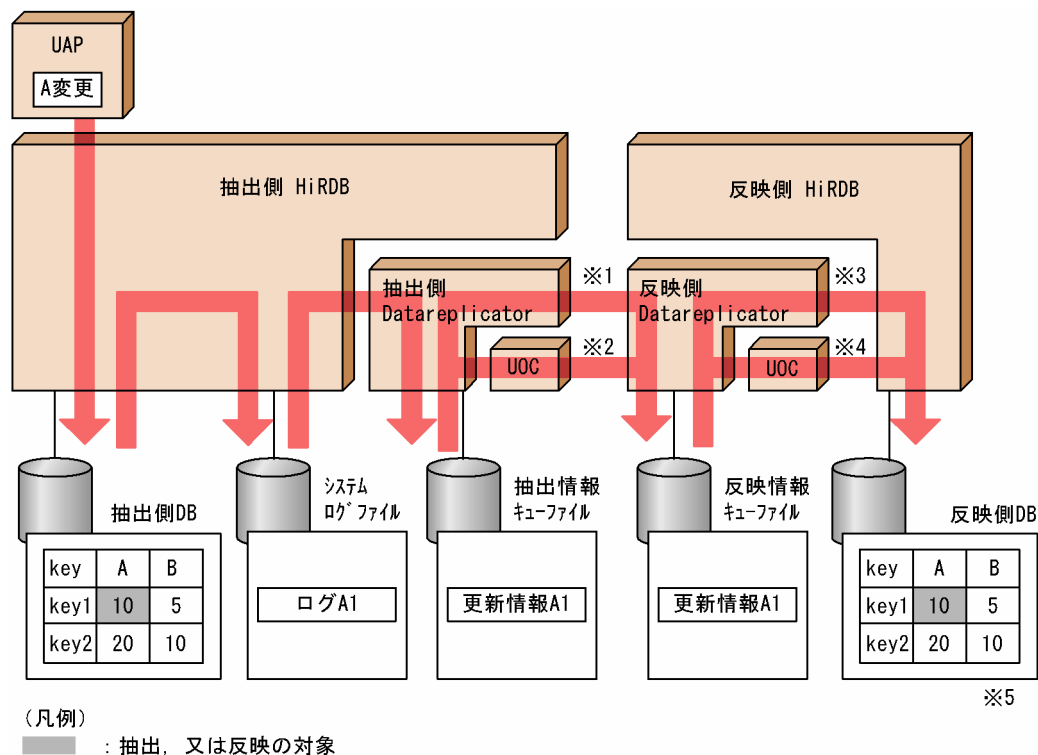
理といいます。ユーザOWNコーディング（UOC）を使うと、更新情報をチェックして、反映側 Datareplicator への送信対象にするかどうかを判定できます。

更新情報は、抽出側 Datareplicator に定義した送信間隔ごとに、反映側 Datareplicator に送信され、反映情報キューファイルに格納されます。反映情報キューファイルに格納された更新情報は、反映側 Datareplicator に定義した抽出側 DB のトランザクション数ごとに、反映側 DB に反映されます。これを反映処理といいます。このとき、反映側 DB への更新は抽出側 DB の更新に同期しません。

反映側 DB への反映には、反映側 Datareplicator が自動的に反映する方法と、ユーザOWNコーディング（UOC）を使って任意に反映する方法の 2 種類があります。

HiRDB から HiRDB へのデータ連動の仕組みを次の図に示します。

図 1-3 HiRDB から HiRDB へのデータ連動の仕組み



注※1

抽出側 Datareplicator が自動的に送信する方法です。

注※2

更新情報をチェックしてから送信するときを使う UOC です。

注※3

反映側 Datareplicator が自動的に反映する方法です。

注※4

更新情報を編集して反映するときを使う UOC です。

注※5

反映側 Datareplicator が自動的に反映する方法の場合です。UOC を経由して反映する方法を使うと、更新情報を任意に加工して反映できます。

(2) HiRDB からメインフレーム側 DB へのデータ連動

HiRDB からメインフレーム側 DB (XDM/RD E2) へのデータ連動の仕組みを説明します。

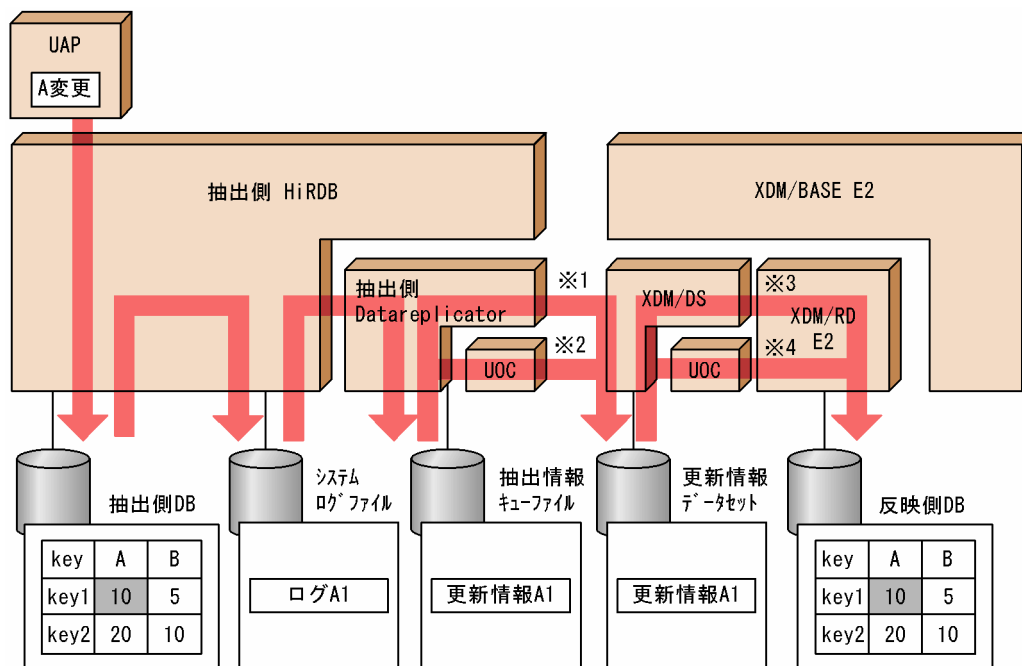
抽出側 DB が更新されると、抽出側 Datareplicator が更新情報を HiRDB のシステムログファイルから抽出して、抽出情報キューファイルに格納します。ユーザオウンコーディング (UOC) を使うと、更新情報をチェックして、反映側 Datareplicator への送信対象にするかどうかを判定できます。

更新情報は、抽出側 Datareplicator に指定した送信間隔ごとにメインフレーム側のデータ連動製品 XDM/DS に送信されて、更新情報データセットに格納されます。格納された更新情報は、XDM/DS に指定した抽出側 DB のトランザクション数ごとに、反映側 DB に反映されます。このとき、反映側 DB への更新は抽出側 DB の更新に同期しません。

反映側 DB への反映方法には、XDM/DS が自動的に反映する方法と、UOC で任意に反映する方法の 2 種類があります。

HiRDB からメインフレーム側 DB へのデータ連動の仕組みを次の図に示します。

図 1-4 HiRDB からメインフレーム側 DB へのデータ連動の仕組み



(凡例)

■ : 抽出, 又は反映の対象

注※1

抽出側 Datareplicator が自動的に送信する方法です。

注※2

更新情報をチェックしてから送信するときに使う UOC です。

注※3

反映側の XDM/DS が自動的に反映する方法です。

注※4

更新情報を編集して反映するときに使う UOC です。

(3) メインフレーム側 DB から HiRDB へのデータ連動

メインフレーム側 DB (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) から HiRDB へのデータ連動の仕組みを説明します。

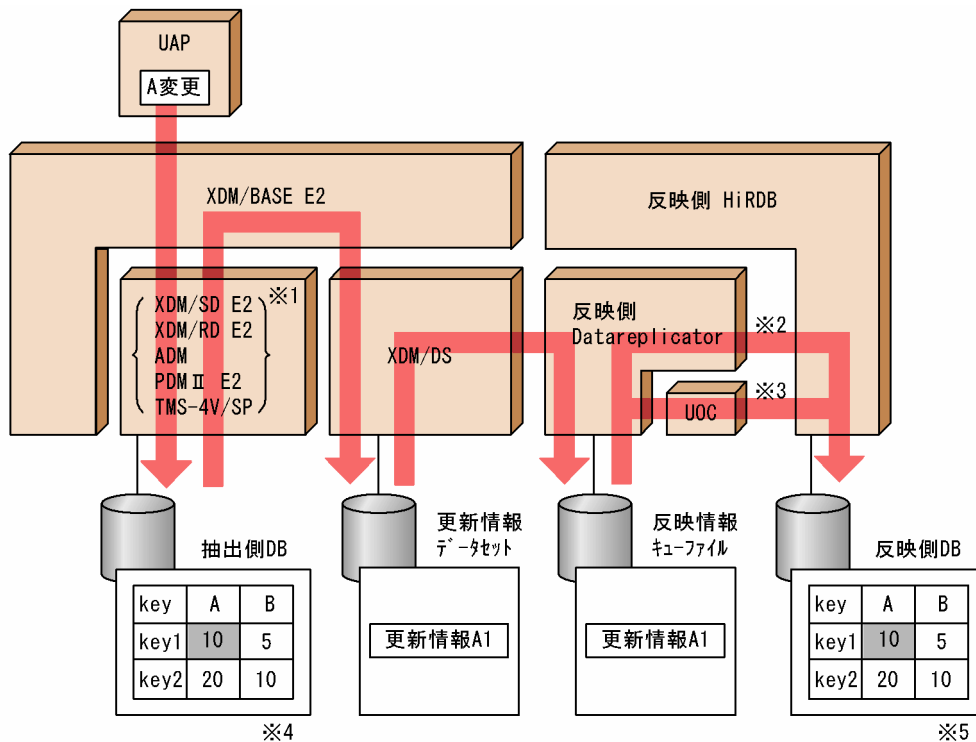
抽出側 DB が更新されると、メインフレーム側のデータ連動製品 XDM/DS が更新情報を抽出して、メインフレーム側の**更新情報データセット**に格納します。

更新情報は、XDM/DS で指定する送信間隔ごとに、反映側 Datareplicator に送信され、**反映情報キューファイル**に格納されます。反映情報キューファイルに格納された更新情報は、反映側 Datareplicator に定義した抽出側 DB のトランザクション数ごとに、反映側 DB に反映されます。このとき、反映側 DB への更新は抽出側 DB の更新に同期しません。

反映側 DB への反映方法には、Datareplicator が自動的に反映する方法と、UOC を使って任意に反映する方法の 2 種類があります。

メインフレーム側 DB から HiRDB へのデータ連動の仕組みを次の図に示します。

図 1-5 メインフレーム側 DB から HiRDB へのデータ連動の仕組み



(凡例)
 : 抽出, 又は反映の対象

注※1

XDM/SD E2, XDM/RD E2, ADM, VOS3 PDM II E2, TMS-4V/SP のどれか一つが必要です。
 TMS-4V/SP の場合は, TMS-4V/SP/データ連携支援が必要です。

注※2

反映側 Datareplicator が自動的に反映する方法です。

注※3

更新情報を編集して反映するときを使う UOC です。

注※4

抽出側 DB が XDM/RD E2 の場合の例です。

注※5

反映側 Datareplicator が自動的に反映する方法の場合の例です。UOC を経由して反映する方法を使うと, 更新情報を任意に加工して反映できます。

(4) SAM ファイルを使った, メインフレーム側 DB から HiRDB へのデータ連動

SAM ファイルとは, ファイルの先頭から順番にデータが格納されているファイルのことです。Datareplicator の場合, SAM ファイルを VOS1 PDMII E2 又は VOS1 RDB1 E2 とのデータ連動のために使います。SAM ファイルを使って, メインフレーム側 DB (PDMII E2, RDB1 E2) から HiRDB へデータ連動する仕組みを説明します。

抽出側 DB が更新されると、メインフレーム側の定義に従って更新情報を抽出して、メインフレーム側の更新情報データセットを経由して、SAM ファイルに格納します。この SAM ファイルは、次に示すファイル転送プログラムを使って、反映側 Datareplicator のシステムにバイナリ形式で転送します。

VOS3 PDMII E2

VOS3 XFIT, XNF/TCP, IFIT-TSS E2 を使ったファイル転送

VOS1 PDMII E2, VOS1 RDB1 E2

VOS1 XFIT, HCAM/TCP, IFIT/IEX を使ったファイル転送

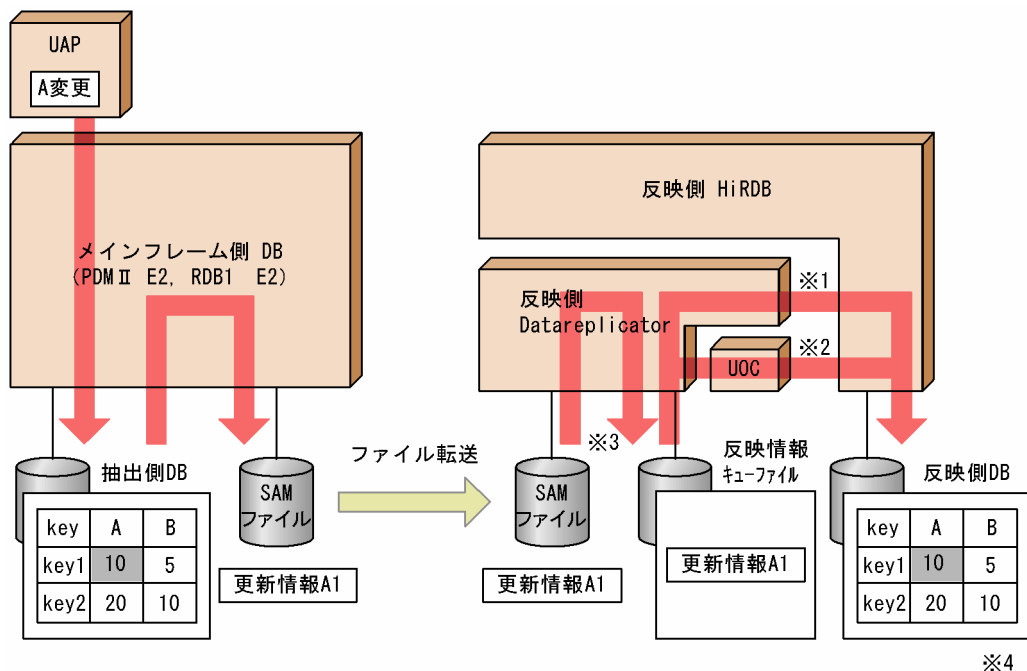
反映側システムのどのディレクトリに SAM ファイルが格納されるかは、ファイル転送プログラムの指定で決まります。

反映側 Datareplicator では、手動で更新情報入力コマンド (hdssamqin コマンド) を実行して、SAM ファイルから反映情報キューファイルに更新情報を格納します。反映情報キューファイルに格納された更新情報は、反映側 Datareplicator に指定した抽出側 DB のトランザクション数ごとに、反映側 DB に反映されます。このとき、反映側 DB への更新は抽出側 DB の更新に同期しません。

反映側 DB への反映方法には、Datareplicator が自動的に反映する方法と、UOC を使って任意に反映する方法の 2 種類があります。

SAM ファイルを使って、メインフレーム側 DB から HiRDB へのデータ連動の仕組みを次の図に示します。

図 1-6 SAM ファイルを使って、メインフレーム側 DB から HiRDB へのデータ連動の仕組み



(凡例)

■ : 抽出, 又は反映の対象

注※1

反映側 Datareplicator が自動的に反映する方法です。

注※2

更新情報を編集して反映するときに使う UOC です。

注※3

SAM ファイルから反映情報キューファイルに更新情報を格納するときは、Datareplicator のコマンド (hdssamqin コマンド) を実行する必要があります。この操作は手動で実行します。

注※4

反映側 Datareplicator が自動的に反映する方法の場合の例です。UOC を経由して反映する方法を使うと、更新情報を任意に加工して反映できます。

1.2.3 抽出側 DB と反映側 DB との対応

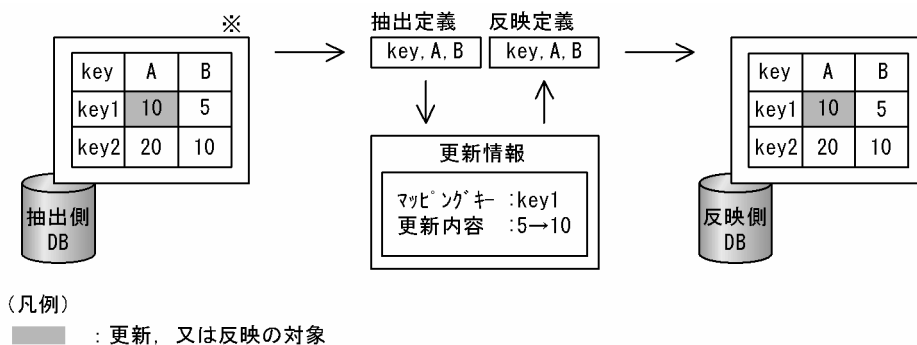
抽出側 DB と反映側 DB とでデータ連動するためには、抽出側 DB と反映側 DB のデータを更新情報で対応付けます。抽出側 DB についての指定を抽出定義、反映側 DB についての指定を反映定義といいます。

それぞれの定義は、データ連動製品 (Datareplicator 又は XDM/DS) で指定します。なお、SAM ファイルを使ったデータ連動の場合、メインフレーム側 DB の定義はファイル転送プログラムに転送するデータセットを作成する前に指定してください。

抽出定義で抽出側 DB のデータと更新情報とを対応付けて、反映定義で更新情報と反映側 DB とを対応付けます。更新情報は、行を特定するためのマッピングキーと更新内容で構成されています。

抽出側 DB と反映側 DB との対応を次の図に示します。

図 1-7 抽出側 DB と反映側 DB との対応



注※

抽出側 DB が HiRDB の場合の例で説明しています。

1.2.4 データ連動できるデータベース

Datareplicator を使ってデータ連動できるデータベースについて説明します。

(1) データベースの種類

Datareplicator でデータ連動できるデータベースには、**構造型データベース**（構造型データベース、階層型データベース、ネットワーク型データベース）と**リレーショナル型データベース**があります。以降、このマニュアルでは構造型データベースを**構造型**と、リレーショナル型データベースを**リレーショナル型**と表記します。

データベースの種類と抽出側 DB、反映側 DB との関係を次の表に示します。

表 1-4 データ連動できるデータベースの種類と抽出側 DB、反映側 DB との関係

抽出側システム		反映側システム	
種類	抽出側 DB	種類	反映側 DB
リレーショナル型	HiRDB XDM/RD E2 TMS-4V/SP※ RDB1 E2	リレーショナル型	HiRDB
構造型	XDM/SD E2		XDM/RD E2
階層型	ADM		
ネットワーク型	PDMII E2		

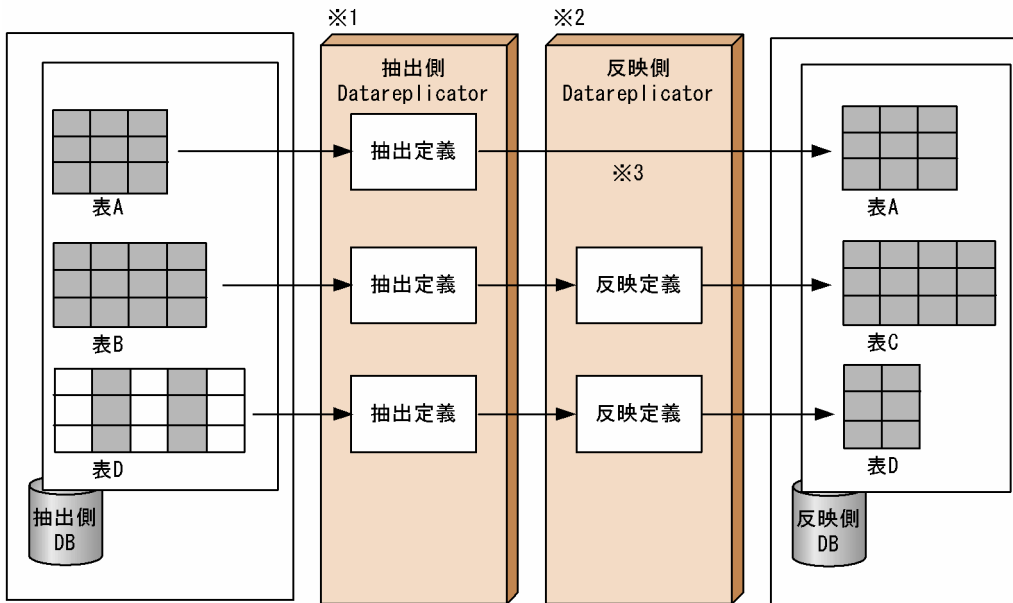
注※

TMS-4V/SP のデータは階層型ですが、TMS-4V/SP/データ連携支援がリレーショナル構造で出力するため、Datareplicator ではリレーショナル型に分類されます。

(2) データベースの種類とデータ連動の関係

リレーショナル型からリレーショナル型へのデータ連動の概念を [図 1-8](#) に、構造型からリレーショナル型へのデータ連動の概念を [図 1-9](#) に示します。

図 1-8 リレーショナル型からリレーショナル型へのデータ連動の概念



(凡例)

■ : 抽出, 反映対象

注※1

抽出側 DB が HiRDB の場合の例です。抽出側 DB が XDM/RD E2 のときは、抽出側 Datareplicator の代わりに XDM/DS が必要です。

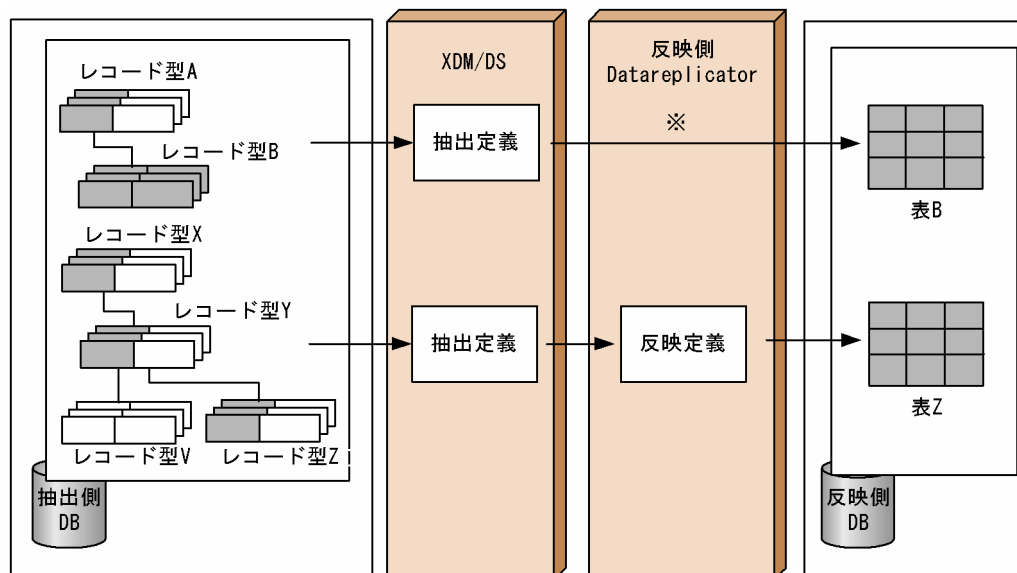
注※2

反映側 DB が HiRDB の場合の例です。反映側 DB が XDM/RD E2 のときは、反映側 Datareplicator の代わりに XDM/DS が必要です。

注※3

抽出側 DB の表を形式、表名、列名などがすべて同一の反映側 DB の表に反映する場合、反映定義を省略できます。

図 1-9 構造型からリレーショナル型へのデータ連動の概念



(凡例)

■ : 抽出, 反映対象

注※

反映定義を省略したときは、反映側 DB の「認可識別子.表識別子」には抽出側 DB の「スキーマ.レコード型」が対応します。

1.2.5 データベース用語の対応

抽出側 DB と反映側 DB とでデータベースの種類が異なる場合、データベース用語も異なります。データベース用語の対応を次の表に示します。

このマニュアルの説明では、リレーショナル型のデータベース用語を使って説明します。

表 1-5 データベース用語の対応

構造型, 階層型, ネットワーク型			リレーショナル型
XDM/SD E2	ADM	PDMII E2	HiRDB, XDM/RD E2, TMS-4V/SP*, RDB1 E2
スキーマ	データベース	DBM	データベース
レコード型	セグメント	データセット	表
レコード実現値	セグメントオカレンス	レコード	行
構成要素	フィールド	フィールド	列

注※

TMS-4V/SP のデータは階層型ですが、TMS-4V/SP/データ連携支援がリレーショナル構造で出力するため、Datareplicator ではリレーショナル型に分類されます。

1.3 データ連動で使える表

Datareplicator でデータ連動できる表を次の表に示します。次の表の内容は抽出側 Datareplicator と反映側 Datareplicator で共通です。

表 1-6 Datareplicator でデータ連動できる表

分類		データ連動の可否
実表	オプションを指定していない場合	○
	SHARE オプションを指定している場合	△※1
	WITHOUT ROLLBACK オプションを指定している場合	△※2
	INSERT ONLY オプションを指定している場合	△※3
	INNER CONSTRUCTOR オプションを指定している場合	△※4
	CHARACTER SET オプションを指定している場合	△※5
	トリガが定義されている場合	△※6
	COMPRESSED オプションを指定している場合	△※7
	上記以外のオプションを指定している場合	○
ビュー表		×
ディクショナリ表		×
一時表		×

(凡例)

- ：データ連動できます。制限はありません。
- △：データ連動できますが、制限があります。※8
- ×

×：データ連動できません。※9

注※1

SHARE オプションを指定している表（共用表）を格納する共用 RD エリアを作成するときに、create rdarea 文の server name オペランドに指定したバックエンドサーバだけデータ連動できます。共用表をデータ連動させる場合は、反映側 Datareplicator で反映定義の load 文に with lock 句を指定する必要があります。抽出側 Datareplicator で定義を指定する必要はありません。反映定義の詳細については、「5.10.5 反映表定義」を参照してください。

注※2

WITHOUT ROLLBACK オプションを指定している表のデータ連動には、前提条件と制限事項があります。詳細については、「[6.7.3\(4\) WITHOUT ROLLBACK オプションを指定している表についての注意](#)」を参照してください。

注※3

抽出表と反映表を同じ構成にしてください。

注※4

このオプションを指定した列を抽出対象としない場合は、データ連動できます。

注※5

CHARACTER SET オプションを指定している列を抽出対象にする場合は、次の制限事項をすべて満たしてください。満たさなかった場合、抽出側 DB と反映側 DB で不整合が発生するおそれがあります。

- 抽出列と、抽出列に対応する反映列の両方に、同じ CHARACTER SET オプションを指定してください。
- 抽出側が XDM/DS で、反映列が CHARACTER SET オプションに EBCDIK を指定している場合、反映定義の format 文でその列に対応するフィールドに nocodecnv を指定してください。

注※6

トリガが定義されている表のデータ連動には、前提条件と制限事項があります。詳細については、「[3.3.6 トリガが定義されている表のデータ連動](#)」を参照してください。

注※7

COMPRESSED オプションを指定している表のデータ連動には、前提条件があります。詳細については、「[6.7.3\(5\) COMPRESSED オプションを指定している表についての注意](#)」を参照してください。

注※8

抽出定義の抽出定義文で指定した場合、抽出定義プリプロセス処理で定義エラーになります。抽出定義の全表抽出定義文を指定した場合、抽出定義プリプロセス処理で KFRB04583-W メッセージを出力し処理を続行します。

注※9

抽出定義の抽出定義文で指定した場合、抽出定義プリプロセス処理で定義エラーになります。抽出定義の全表抽出定義文を指定した場合、データ連動できない表は無視します。

1.4 データ連動で使える表のデータ型

Datareplicator で使えるデータ型は、HiRDB で扱えるデータ型と同じです。ただし、一部のデータ型は Datareplicator で認識できるデータ型に変換されます。また、HiRDB の特殊なデータ型（抽象データ型の列、繰返し列）も、データ連動できます。

1.4.1 Datareplicator で使えるデータ型

Datareplicator で使えるデータ型を次の表に示します。

表 1-7 Datareplicator で使えるデータ型

分類	Datareplicator で使えるデータ型
数値型	INTEGER, SMALLINT, DECIMAL, LARGEDECIMAL, FLOAT, SMALLFLT
文字型	CHAR ^{*3} , VARCHAR ^{*3} , NCHAR, NVARCHAR, MCHAR, MVARCHAR
日付型	DATE, TIME, TIMESTAMP, INTERVAL YEAR TO DAY, INTERVAL HOUR TO SECOND
抽象データ型 ^{*1}	SGMLTEXT, FREEWORD, XML
特殊な型 ^{*2}	BLOB, BINARY

注 1

更新をしていない列についても、同じ値で更新したものとしてデータ連動します。ただし、次のデータ型の列については、更新していない場合データ連動しません。

- VARCHAR 型（256 バイト以上の場合）
- NVARCHAR 型（128 文字以上の場合）
- MVARCHAR 型（256 バイト以上の場合）
- BLOB 型
- BINARY 型（256 バイト以上場合）
- 抽象データ型（SGMLTEXT, FREEWORD, XML）

注 2

- データ型に繰返し列を作成しているときでも抽出できます。繰返し列を使った表のデータ連動については、「[1.4.3 繰返し列を使った表のデータ連動](#)」を参照してください。
- LONG VARCHAR, LONG NVARCHAR, 及び LONG MVARCHAR は、それぞれ VARCHAR, NVARCHAR, 及び MVARCHAR として認識されます。

注 3

JIS 第 3 水準・第 4 水準の文字をデータ連動するには条件があります。詳細については、「[2.10.3 JIS 第 3 水準・第 4 水準の文字のサポート](#)」を参照してください。

注※1

抽象データ型のデータ連動には規則があります。詳細については、「1.4.2 抽象データ型を使った表のデータ連動」を参照してください。

注※2

BLOB 型及び BINARY 型のデータ連動には規則があります。詳細については、「1.4.4 BLOB 型と BINARY 型を使った表のデータ連動」を参照してください。

注※3

文字集合が指定されている場合もデータ連動できます。ただし、抽出列に指定した文字集合と反映列に指定した文字集合が同じである必要があります。文字集合については、「4.3.5 文字コードの対応の設計」の EBCDIK の説明を参照してください。

Datareplicator で抽出した列のデータ型は、反映側システムが HiRDB のときは抽出したデータ型で反映されます。XDM/RD に反映するときのデータ型については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 運用」及び該当するデータベースのマニュアルを参照してください。

抽出側システム／反映側システムと Datareplicator のデータ型の関係については、「4.3.4 データ型の対応の設計」を参照してください。

1.4.2 抽象データ型を使った表のデータ連動

Datareplicator では、HiRDB の抽象データ型 (ADT) の列を対象にしてデータ連動できます。抽象データ型とは、マルチメディアデータなど複雑な構造を持つ長大なデータ型のことです。HiRDB では、マルチメディアデータを抽象データ型として表に格納して、従来のデータベースの表と同じように SQL でアクセスできるようにしています。

SGMLTEXT 型を含む表の例を次に示します。

● SGMLTEXT型を含む表を作成する SQL (CREATE TABLE)

```
CREATE TABLE レポート (題名 CHAR(32), 日付 DATE, 作者 CHAR(32),
                        文献 SGMLTEXT RECOVERY ALL ALLOCATE (sgmltext IN (PDUSER02))
                        PLUGIN '<TEXTTYPE>SGML</TEXTTYPE><DTD>MAN.dtd</DTD>'
                        <NORparm>MANnorm.prm</NORparm>'
                        ) IN (PDUSER01);
```

● CREATE TABLEで作成されるレポート表 (SGMLTEXT型の列名: 文献)

列のデータ型	CHAR(32)	DATE	CHAR(32)	SGMLTEXT
列名	題名	日付	作者	文献
	:	:	:	:

(1) Datareplicator でデータ連動できる抽象データ型

Datareplicator でデータ連動できる抽象データ型は、SGMLTEXT 型、FREEWORD 型、及び XML 型です。抽象データ型は HiRDB 同士のデータ連動システムの場合だけデータ連動できます。

(2) 抽出側システムと反映側システムのデータ型の対応

抽象データ型を使ったデータ連動では、抽出側システムと反映側システムで次に示す条件を満たしている必要があります。

- 抽出側 DB と反映側 DB が共に HiRDB
- 抽出側と反映側の列のデータ型が、両方とも抽象データ型
- 抽象データ型の定義（型名、属性数、属性データ型など）が、抽出側と反映側で一致
- 抽象データ型の所有者名（認可識別子）が、抽出側と反映側で一致
- 抽象データ型に対するコンストラクタ関数の定義が、抽出側と反映側で一致
- コンストラクタ関数に渡す引数の数は一つ
- HiRDB プラグインの抽象データ型に対する固有の定義が、抽出側と反映側で一致
- 文書型定義が抽出側と反映側で一致（SGMLTEXT 型限定）
- 文字コードが抽出側と反映側で一致（XML 型限定）

注意

上記の条件を満たしているかどうかは、Datareplicator ではチェックしません。適切なデータが反映されたかどうかは、反映側 HiRDB のデータ互換性チェックに依存します。反映側 HiRDB への反映処理がエラーになったときは、Datareplicator がエラーメッセージを出力した後、反映処理を停止します。エラーになったときの対処については、既存のデータ型での互換性チェックでエラーになったときと同じです。

1.4.3 繰返し列を使った表のデータ連動

Datareplicator では、HiRDB の繰返し列を対象にしてデータ連動できます。繰返し列とは、一つのセルに複数の要素を持つ列のことです。HiRDB では、複数の要素を一つのセルに入れた列を繰返し列として表に格納して、従来のデータベースの表と同じように SQL でアクセスできるようにしています。

● 繰返し列を含む表を作成する SQL (CREATE TABLE)

```
CREATE TABLE 社員表 (氏名 NVARCHAR(10),
                     性別 NCHAR(1),
                     家族 NVARCHAR(5) ARRAY[10],
                     続柄 NVARCHAR(5) ARRAY[10]);
```

● CREATE TABLEで作成される社員表 (繰返し列名: 家族 と 続柄)

列のデータ型	NVARCHAR(10)	NCHAR(1)	NVARCHAR(5)	NVARCHAR(5)
列名	氏名	性別	家族	続柄
	○×△◇	男	○○○○○ ××××× : △△△△△ ◇◇◇◇◇	父 母 : 弟 妹
	:	:	:	:
	:	:	:	父 母 : 弟 妹

ただし、繰返し列のセル単位にデータ連動できますが、要素単位ではデータ連動できません。

(1) 抽出側システムと反映側システムのデータ型の対応

繰返し列を使ったデータ連動では、抽出側システムと反映側システムで次に示す条件を満たしている必要があります。

- 抽出側と反映側の列に、両方とも繰返し列を定義
- 繰返し列の定義 (型名, 属性数, 属性データ型, 要素データ数, 要素データの順序など) が、抽出側と反映側で一致

ただし、反映情報編集 UOC を使うとデータを編集できるので、上記の条件を満たしていないときでもデータ連動できることがあります。

注意
 上記の条件を満たしているかどうかは、Datareplicator ではチェックしません。適切なデータが反映されたかどうかは、反映側 HiRDB のデータ互換性チェックに依存します。反映側 HiRDB への反映処理がエラーになったときは、Datareplicator がエラーメッセージを出力した後、反映処理を停止します。エラーになったときの対処については、既存のデータ型での互換性チェックでエラーになったときと同じです。

1.4.4 BLOB 型と BINARY 型を使った表のデータ連動

Datareplicator では、BLOB 型と BINARY 型をデータ連動できますが、規則があります。BLOB 型と BINARY 型の規則について説明します。

(1) BLOB 型の規則

BLOB 型の列をデータ連動する場合の規則を次に示します。

- BLOB 型の列をデータ連動する場合は、HiRDB の表定義の RECOVERY オペランドに ALL を指定してください。HiRDB の表定義については、マニュアル「HiRDB SQL リファレンス」を参照してください。
- BLOB 型の列の定義長では、Datareplicator は、BLOB データ実長ではなく BLOB 列定義長に従ってバッファなどを割り当てるため、BLOB 列定義長はできるだけ実データ長に近い値で定義してください。
- 定義長が 2GB 以上又は表作成時に定義長指定を省略（HiRDB が定義長として 2GB を仮定）した BLOB 型の列を抽出対象とすると、定義エラーになります。性能とメモリ所要量を考慮して、定義長が数 MB までの BLOB 型の列をデータ連動対象とすることをお勧めします。

ただし、定義長は 2GB 以上でも実際に扱うデータは小さい BLOB 型の列の場合、抽出側 Datareplicator の環境変数 HDE_BIN_COL_MAXLEN を指定することで、表を再定義することなく BLOB 型の列をデータ連動できるようになります。環境変数 HDE_BIN_COL_MAXLEN の詳細については、次の個所を参照してください。

- [\[2.4.1 抽出側 Datareplicator の環境変数\]](#) (UNIX)
 - [\[2.8.1 抽出側 Datareplicator の環境変数\]](#) (Windows)
 - [\[6.4.3\(4\) 環境変数 HDE_BIN_COL_MAXLEN についての注意\]](#)
- BLOB 型の抽出列はマッピングキーにできません。
 - HiRDB のユティリティの処理対象に抽出対象の BLOB 型の列が含まれている場合、更新ログを出力する更新であっても、BLOB 型の列に対するユティリティの更新内容を Datareplicator は認識しないため、データ連動の対象にはなりません。反映側 DB には、BLOB 型の列に対するユティリティの実行結果は反映されないため、注意してください。
 - 反映側 Datareplicator での BLOB 型データの未反映情報には、次に示すフォーマットで更新データが出力されます。

*BLOB (データ長) *

(2) BINARY 型の規則

BINARY 型の列をデータ連動する場合の規則を次に示します。

- 性能とメモリ所要量を考慮して、定義長が 10MB 以下の BINARY 型の列をデータ連動対象とすることをお勧めします。
- BINARY 型の抽出列はマッピングキーにできません。
- 反映側 Datareplicator での BINARY 型データの未反映情報には、次に示すフォーマットで更新データが出力されます。

*BINARY (データ長) *

(3) BLOB 型と BINARY 型の後方削除更新について

HiRDB のシステム共通定義の `pd_rpl_func_control` オペランドに `BACKWARD_CUTOFF_UPDATE` を指定すると、BLOB 型と BINARY 型を後方削除更新できます。BLOB 型と BINARY 型の後方削除更新の詳細については、マニュアル「HiRDB UAP 開発ガイド」を参照してください。

なお、BLOB 型又は BINARY 型の後方削除更新のデータ連動には、次に示す制限があります。

- マージ表への反映はできない
- 時系列情報表への反映はできない
- 列データ編集 UOC を使用できない
- 送信データ UOC を使用できない

pd_rpl_func_control オペランドの指定に関する注意

BLOB 型又は BINARY 型を後方削除更新しない場合は、HiRDB のシステム共通定義の `pd_rpl_func_control` オペランドに `BACKWARD_CUTOFF_UPDATE` を指定しないでください。

また、`pd_rpl_func_control` オペランドの `BACKWARD_CUTOFF_UPDATE` の指定を変更する場合は、変更する前にすべての更新ログが反映し終わっているかどうかを確認してください。反映されていない更新ログがある状態で `pd_rpl_func_control` オペランドを変更すると、抽出側 DB と反映側 DB で SQL の実行数が一致しないなどの問題が発生します。

1.5 データ連動できる SQL

Datareplicator でデータ連動できる SQL を次の表に示します。

表 1-8 Datareplicator でデータ連動できる SQL

SQL 種別	データ連動の可否	制限
INSERT	○	WITHOUT ROLLBACK オプションを指定している表はデータ連動できません。
UPDATE	○	UPDATE 文の SET 句に指定していない列（更新をしていない列）についても、同じ値で更新したものとしてデータ連動します。ただし、次のデータ型については、更新していない場合データ連動しません。 <ul style="list-style-type: none">• VARCHAR 型（256 バイト以上の場合）• NVARCHAR 型（128 文字以上の場合）• MVARCHAR 型（256 バイト以上の場合）• BLOB 型• BINARY 型（256 バイト以上の場合）• 抽象データ型（SGMLTEXT, FREEWORD, XML）
DELETE	○	WITHOUT ROLLBACK オプションを指定している表はデータ連動できません。
PURGE	○	<ul style="list-style-type: none">• WITHOUT ROLLBACK オプションを指定している表はデータ連動できません。• 抽出対象の表を複数の BES に分割して配置している場合はデータ連動できません。複数の BES に分割して配置している表に対して PURGE 文を実行した場合の動作は、「5.4 送信環境定義」の prg_eventno オペランドを参照してください。
上記以外の SQL	×	—

(凡例)

○：データ連動できます。ただし、制限があります。

×：データ連動できません。

注

抽出側 DB で複数の行を一つの SQL 文で更新した場合、反映側 Datareplicator では、抽出側 DB で更新した行の数だけ SQL 文を発行します（SQL 単位ではなく更新した行単位でデータ連動します）。

1.6 ソフトウェア構成

データ連動システムのシステム構成には、次に示す 4 種類があります。ここでは、それぞれのデータ連動システムのソフトウェア構成について説明します。

- HiRDB から HiRDB へデータ連動するシステム構成
- HiRDB からメインフレーム側 DB へデータ連動するシステム構成
- メインフレーム側 DB から HiRDB へデータ連動するシステム構成
- SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動するシステム構成

抽出側 DB が HiRDB の場合

抽出側 Datareplicator は、抽出対象となる HiRDB のシステムマネージャがあるサーバマシン、及び抽出対象表が存在する HiRDB のバックエンドサーバがあるサーバマシンにインストールしてください。

反映側 DB が HiRDB の場合

反映側 Datareplicator は、反映対象となる HiRDB と同じサーバマシンにインストールしてください。ただし、HiRDB クライアントが同じサーバマシンにインストールされている場合は、反映対象となる HiRDB とは異なるサーバマシンにもインストールできます。

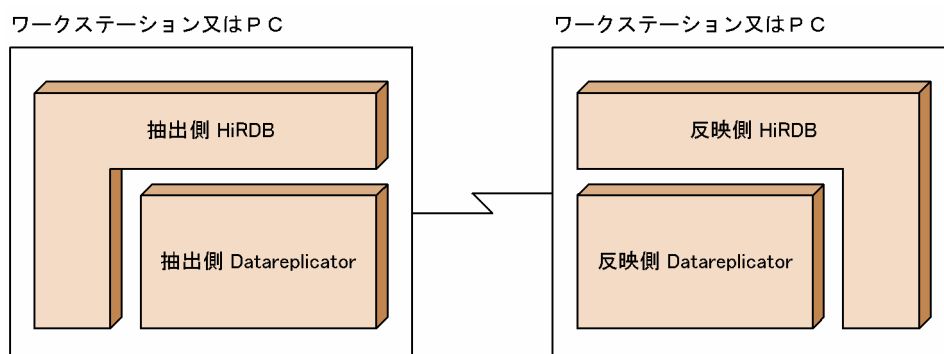
それぞれのシステム構成によって、必要なソフトウェアが異なります。システム構成と必要なソフトウェア構成については、「[2.1 Datareplicator に関連する製品](#)」を参照してください。

ここに示す Datareplicator 側のシステム構成は、UNIX と Windows で共通です。

1.6.1 HiRDB から HiRDB へデータ連動する場合のソフトウェア構成

HiRDB から HiRDB へデータ連動する場合のソフトウェア構成を次の図に示します。

図 1-10 HiRDB から HiRDB へデータ連動する場合のソフトウェア構成



(1) 抽出側システムと反映側システムの対応

HiRDB から HiRDB へデータ連動する場合の、抽出側システムと反映側システムの対応を次に示します。

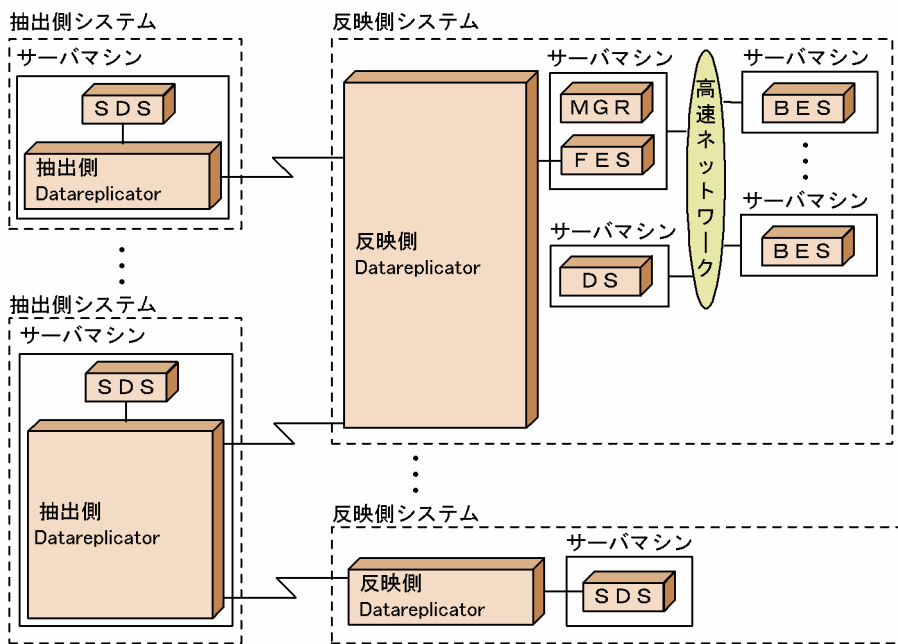
- 抽出側 HiRDB と抽出側 Datareplicator との対応
抽出側 HiRDB (シングルサーバ) : 抽出側 Datareplicator = 1 : 1
抽出側 HiRDB (パラレルサーバ) : 抽出側 Datareplicator = 1 : n
- 抽出側 Datareplicator と反映側 Datareplicator との対応
抽出側 Datareplicator : 反映側 Datareplicator = 1 : m
抽出側 Datareplicator : 反映側 Datareplicator = n : 1
- 反映側 Datareplicator と反映側 HiRDB との対応
反映側 Datareplicator : 反映側 HiRDB = n : 1

(2) 抽出側システムと反映側システムの形態

HiRDB から HiRDB へデータ連動する場合は、抽出側 HiRDB がシングルサーバの場合とパラレルサーバの場合とでソフトウェア構成が異なります。

抽出側 HiRDB がシングルサーバの場合の形態を [図 1-11](#) に、抽出側 HiRDB がパラレルサーバの場合の形態を [図 1-12](#) に示します。

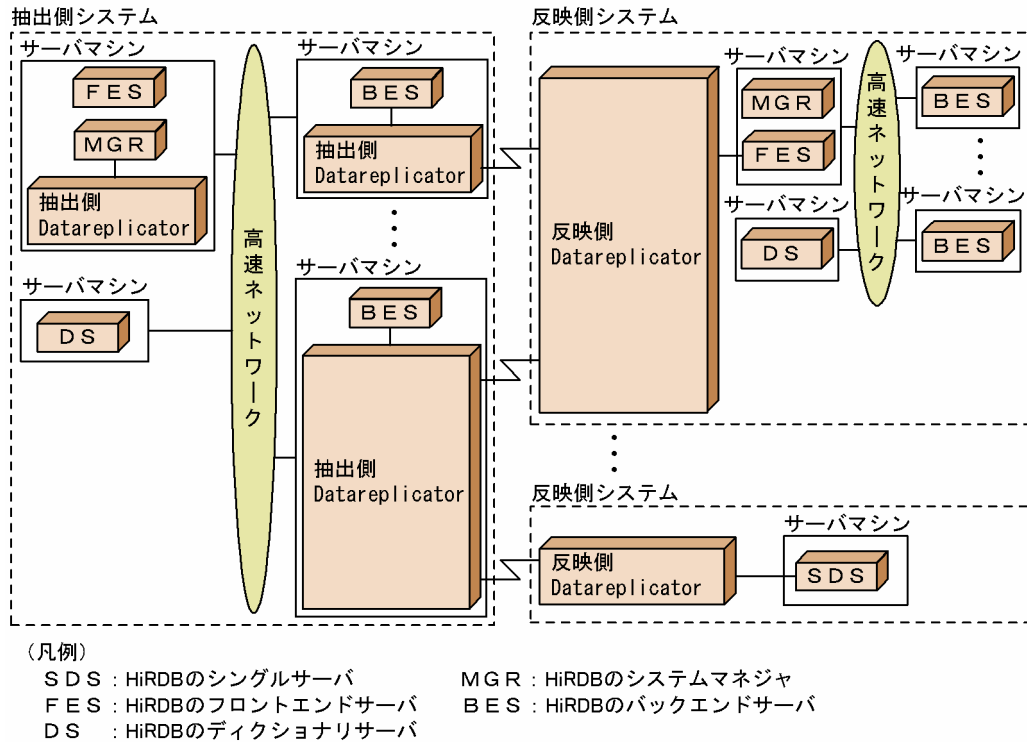
図 1-11 HiRDB から HiRDB へデータ連動する場合の抽出側システムと反映側システムの形態 (抽出側 HiRDB がシングルサーバの場合)



(凡例)

- | | |
|------------------------|-----------------------|
| SDS : HiRDBのシングルサーバ | MGR : HiRDBのシステムマネージャ |
| FES : HiRDBのフロントエンドサーバ | BES : HiRDBのバックエンドサーバ |
| DS : HiRDBのディクショナリサーバ | |

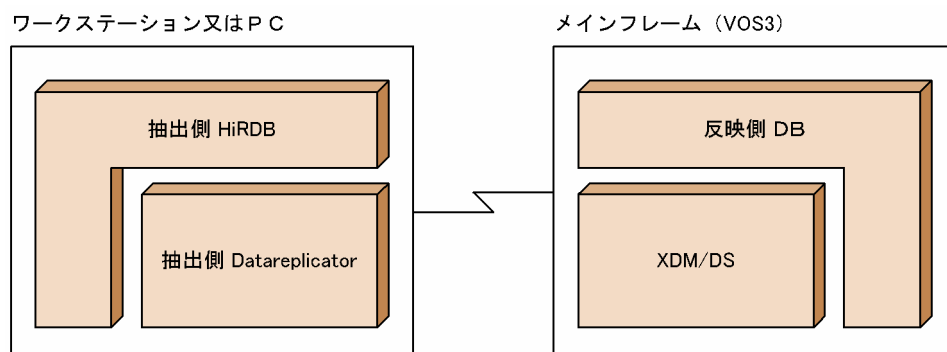
図 1-12 HiRDB から HiRDB へデータ連動する場合の抽出側システムと反映側システムの形態
(抽出側 HiRDB がパラレルサーバの場合)



1.6.2 HiRDB からメインフレーム側 DB へデータ連動する場合のソフトウェア構成

HiRDB からメインフレーム側 DB (XDM/RD E2) へデータ連動する場合のソフトウェア構成を次の図に示します。

図 1-13 HiRDB からメインフレーム側 DB へデータ連動する場合のソフトウェア構成



(1) 抽出側システムと反映側システムの対応

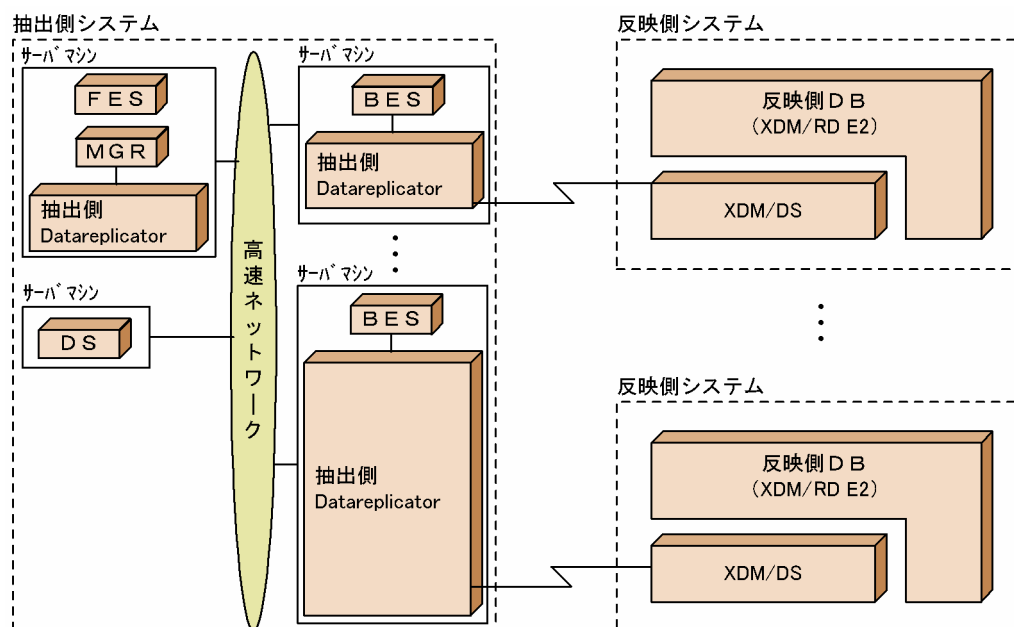
HiRDB からメインフレーム側 DB (XDM/RD E2) にデータ連動する場合の、抽出側システムと反映側システムの対応を次に示します。

- 抽出側 Datareplicator と抽出側 HiRDB との対応
抽出側 HiRDB (シングルサーバ) : 抽出側 Datareplicator = 1 : 1
抽出側 HiRDB (パラレルサーバ) : 抽出側 Datareplicator = 1 : n
- Datareplicator と XDM/DS との対応
抽出側 Datareplicator : XDM/DS = 1 : m
抽出側 Datareplicator : XDM/DS = 1 : 1
- 反映側 DB と XDM/DS との対応
マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

(2) 抽出側システムと反映側システムの形態

HiRDB からメインフレーム側 DB (XDM/RD E2) へデータ連動する場合の、Datareplicator と抽出側、反映側システムとの対応を次の図に示します。

図 1-14 HiRDB からメインフレーム側 DB へデータ連動する場合の抽出側システムと反映側システムの形態

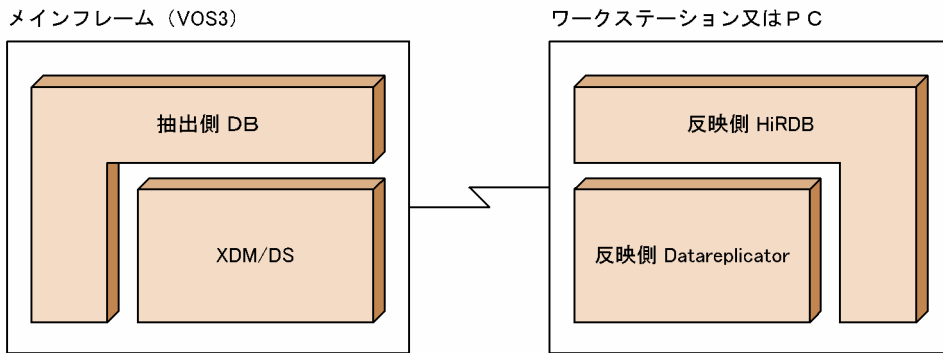


(凡例)
 F E S : HiRDBのフロントエンドサーバ M G R : HiRDBのシステムマネージャ
 D S : HiRDBのディクショナリサーバ B E S : HiRDBのバックエンドサーバ

1.6.3 メインフレーム側 DB から HiRDB へデータ連動する場合のソフトウェア構成

メインフレーム側 DB (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) から HiRDB へデータ連動する場合のソフトウェア構成を次の図に示します。

図 1-15 メインフレーム側 DB から HiRDB へデータ連動する場合のソフトウェア構成



(1) 抽出側システムと反映側システムの対応

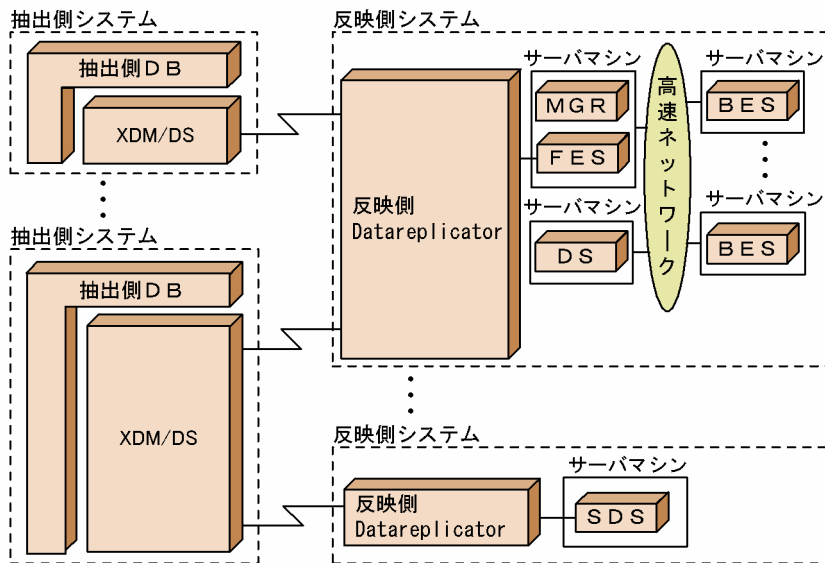
メインフレーム側 DB (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) から HiRDB にデータ連動する場合の、抽出側システムと反映側システムの対応を次に示します。

- 抽出側 DB と XDM/DS との対応
マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。
- XDM/DS と Datareplicator との対応
XDM/DS : 反映側 Datareplicator = 1 : m
XDM/DS : 反映側 Datareplicator = n : 1
- 反映側 Datareplicator と反映側 HiRDB との対応
反映側 Datareplicator : 反映側 HiRDB = n : 1

(2) 抽出側システムと反映側システムの形態

メインフレーム側 DB から HiRDB へデータ連動する場合の、Datareplicator と抽出側、反映側システムとの対応を次の図に示します。

図 1-16 メインフレーム側 DB から HiRDB ヘデータ連動する場合の抽出側システムと反映側システムの形態

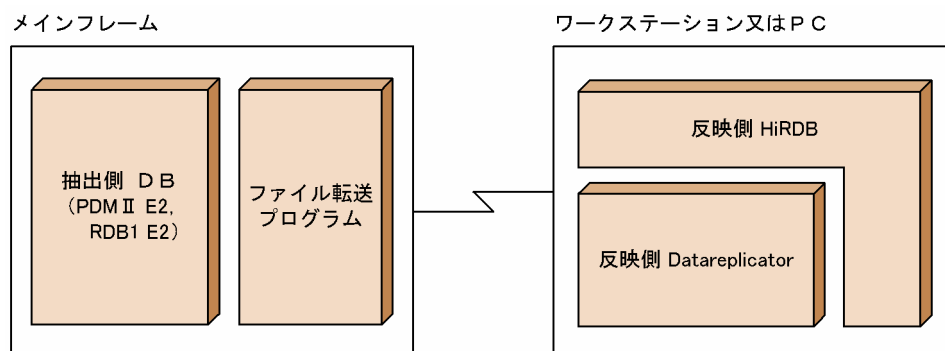


(凡例)
 SDS : HiRDBのシングルサーバ MGR : HiRDBのシステムマネージャ
 FES : HiRDBのフロントエンドサーバ BES : HiRDBのバックエンドサーバ
 DS : HiRDBのディスクジョナリサーバ

1.6.4 SAM ファイルを使って、メインフレーム側 DB から HiRDB ヘデータ連動する場合のソフトウェア構成

SAM ファイルを使って、メインフレーム側 DB (PDMII E2, RDB1 E2) から HiRDB ヘデータ連動する場合のソフトウェア構成を次の図に示します。

図 1-17 SAM ファイルを使って、メインフレーム側 DB から HiRDB ヘデータ連動する場合のソフトウェア構成



(1) 抽出側システムと反映側システムの対応

SAM ファイルを使ってメインフレーム側 DB から HiRDB にデータ連動する場合の、抽出側システムと反映側システムの対応を次に示します。

- 抽出側 DB と Datareplicator との対応

抽出側 DB と反映側 Datareplicator の対応数には、特に制限はありません。

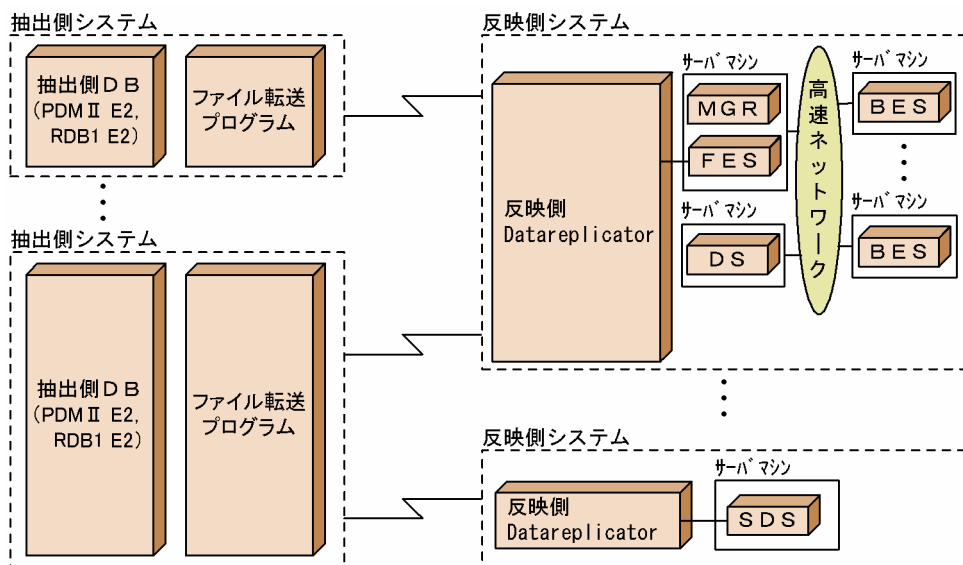
- 反映側 Datareplicator と反映側 HiRDB との対応

反映側 Datareplicator : 反映側 HiRDB = n : 1

(2) 抽出側システムと反映側システムの形態

SAM ファイルを使って、メインフレーム側 DB (PDM II E2, RDB1 E2) から HiRDB へデータ連動する場合の、Datareplicator と抽出側、反映側システムとの対応を次の図に示します。

図 1-18 SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動する場合の抽出側システムと反映側システムの形態



(凡例)

S D S : HiRDBのシングルサーバ

F E S : HiRDBのフロントエンドサーバ

D S : HiRDBのディクショナリサーバ

M G R : HiRDBのシステムマネージャ

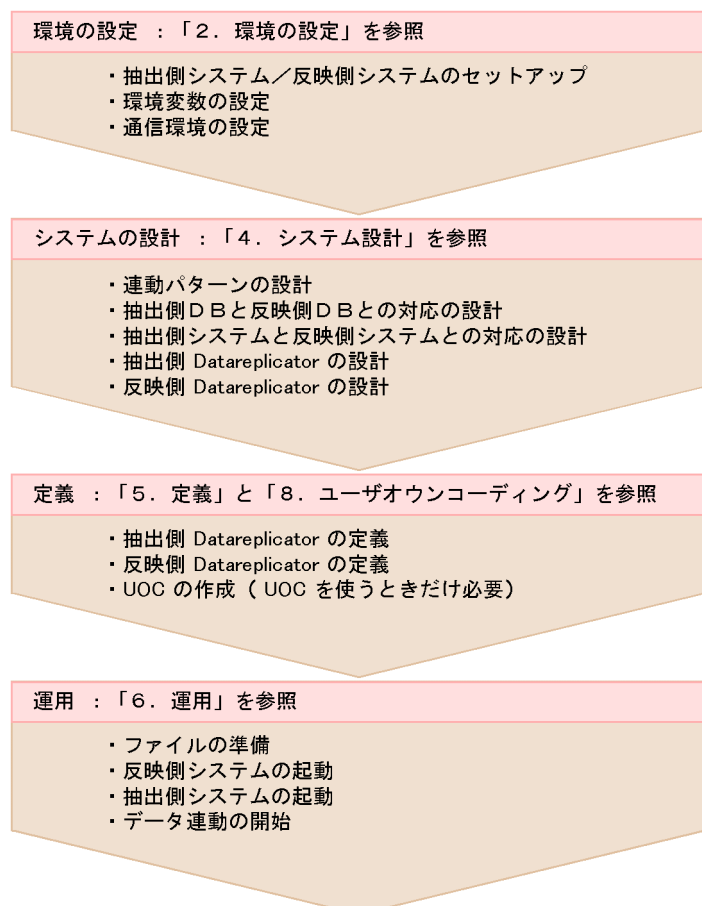
B E S : HiRDBのバックエンドサーバ

1.7 データ連動システムの構築手順

Datereplicator を導入してから、データ連動するまでに必要なシステム構築手順を次の図に示します。

HiRDB システムの構築手順と運用方法については、マニュアル「HiRDB システム導入・設計ガイド」又はマニュアル「HiRDB システム運用ガイド」を参照してください。

図 1-19 データ連動するまでに必要なシステム構築手順



2

環境の設定

この章では、Datareplicator に関連する製品、Datareplicator のインストール手順と作成されるディレクトリ、環境変数の設定、及び通信環境の設定について説明します。環境を設定する方法は、UNIX 版 Datareplicator (2.2~2.5) と Windows 版 Datareplicator (2.6~2.9) に分けて説明します。

2.1 Datareplicator に関連する製品

Datareplicator を使うときに必要な、下記のソフトウェアについて説明します。

- Datareplicator に必要な製品
- 抽出側システムに必要な製品
- 反映側システムに必要な製品

2.1.1 Datareplicator に必要な製品

Datareplicator に必要な製品の一覧を次の表に示します。次の表の内容は、抽出側 Datareplicator と反映側 Datareplicator で共通です。

表 2-1 Datareplicator に必要な製品の一覧

製品名		必要度	備 考
OS (UNIX 版の場合)	HP-UX, AIX, 又は Linux	○	特にありません。
OS (Windows 版の場合)	Windows 2000, Windows Server 2003, Windows XP, Windows Vista, 又は Windows Server 2008		
DBMS (HiRDB)	HiRDB/Single Server	○	前提になる HiRDB です。
	HiRDB/Parallel Server		HiRDB/シングルサーバにするか、HiRDB/パラレルサーバにするかによって、インストールする製品が決まります。
	HiRDB Text Search Plug-in	△	抽象データ型 (SGMLTEXT 型, 又は FREEWORD 型) の列を使うときに必要な HiRDB プラグイン製品です。
	HiRDB XML Extension	△	抽象データ型 (SGMLTEXT 型, FREEWORD 型, 又は XML 型) の列を使うときに必要な HiRDB プラグイン製品です。
周辺 PP	XNF/H/BASE	△	HP-UX 版 Datareplicator で, OSI プロトコル (チャンネル接続) で通信するときが必要です。

(凡例)

- ：どれか一つが必要です。
- △：条件によっては必要です。

2.1.2 抽出側システムに必要な製品

抽出側システムに必要な製品の一覧を次の表に示します。次の表では、OS と周辺 PP は省略します。

表 2-2 抽出側システムに必要な製品の一覧

抽出側 DB	データ連動に必要な製品	備考
HiRDB	HiRDB Datareplicator	特にありません。
XDM/SD E2	XDM/DS	特にありません。
XDM/RD E2	XDM/DS	特にありません。
ADM	ADM/EB	ADM がオンライン環境で稼働する場合に必要です。
	XDM/DS 又は VOS3 Database Datareplicator ^{※1}	特にありません。
PDMII E2 (VOS3)	PDMII/EF	XDM/DS 又は Database Datareplicator を使用してデータ連動する場合に必要です。
	XDM/DS 又は VOS3 Database Datareplicator ^{※1}	XDM/DS 又は VOS3 Database Datareplicator を使用しないで、SAM ファイルを作成してデータ連動することもできます。
PDMII E2 (VOS1) ^{※2}	—	抽出側 DB で SAM ファイルを作成し、次に示すファイル転送プログラムを使用して反映側に転送します。 VOS1 XFIT/FTP, VOS1 HCAM/TCP, 又は VOS1 IFIT/IEX
TMS-4V/SP ^{※3}	TMS-4V/SP/データ連携支援 (VOS3)	TMS-4V/SP を使用してデータ連動する場合に必要です。
	XDM/DS 又は VOS3 Database Datareplicator ^{※1}	特にありません。
RDB1 E2 ^{※2}	—	抽出側 DB で SAM ファイルを作成し、以下のファイル転送プログラムを使用して反映側に転送します。 VOS1 XFIT/FTP, VOS1 HCAM/TCP, 又は VOS1 IFIT/IEX

(凡例)

—：データ連動に必要な製品はありません。

注※1

XDM/DS 以外にも、メインフレームのデータベース向けのデータ連動製品に VOS3 Database Datareplicator があります。VOS3 のデータ連動製品が XDM/DS でも VOS3 Database Datareplicator でも、Datareplicator で定義する内容は変わりません。

注※2

自動的にはデータ連動できないため、反映側 Datareplicator で抽出データを反映する操作が必要になります。

注※3

TMS-4V/SP を抽出側 DB にするときは、ほかの抽出側 DB と比べてデータ形式が異なります。
TMS-4V/SP からの抽出については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」又は「VOS3 TMS-4V/SP データ連携支援機能 TMS-4V/SP/データ連携支援」を参照してください。

2.1.3 反映側システムに必要な製品

反映側システムに必要な製品の一覧を次の表に示します。次の表では、OS と周辺 PP は省略します。

表 2-3 反映側システムに必要な製品の一覧

反映側 DB	データ連動に必要な製品
HiRDB	HiRDB Datareplicator
XDM/RD E2	XDM/DS

2.2 Datareplicator のインストール (UNIX)

UNIX 版 Datareplicator をインストールする方法について説明します。

2.2.1 インストール前の準備

Datareplicator をインストールする前に、次に示す作業が必要になります。

- 稼働環境の確認
- ユーザ権限の確認

(1) 稼働環境の確認

次に示す内容を確認してください。

- Datareplicator をインストールするマシンのディスクの空き容量が十分であることを確認してください。Datareplicator に必要なディスク容量については、「[4.6.8 抽出側 Datareplicator のリソースの設計](#)」又は「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。
- Datareplicator が動作するのに必要な製品がインストールされていることを確認してください。Datareplicator が動作するのに必要な製品については、「[2.1 Datareplicator に関連する製品](#)」を参照してください。

(2) ユーザ権限の確認

インストールを実行する利用者がスーパーユーザであることを確認してください。

2.2.2 インストールするサーバマシン

Datareplicator をインストールするサーバマシンは、HiRDB システムが HiRDB/シングルサーバの場合と HiRDB/パラレルサーバの場合で異なります。さらに HiRDB/パラレルサーバの場合は、抽出側システムと反映側システムで異なります。

Datareplicator をインストールするサーバマシンを次の表に示します。

表 2-4 Datareplicator をインストールするサーバマシン

HiRDB システムの種類	抽出側システムの場合	反映側システムの場合
HiRDB/シングルサーバ	シングルサーバがあるサーバマシンに Datareplicator をインストールします。	シングルサーバがあるサーバマシンに Datareplicator をインストールします。

HiRDB システムの種類	抽出側システムの場合	反映側システムの場合
HiRDB/パラレルサーバ	次のサーバがあるサーバマシンすべてに Datareplicator をインストールします。 <ul style="list-style-type: none"> システムマネージャ (MGR) バックエンドサーバ (BES) 	HiRDB クライアント機能が動作するサーバマシンに、一つの Datareplicator をインストールします。

2.2.3 インストールの実行

「2.1 Datareplicator に関連する製品」を参照して、UNIX 版 Datareplicator を使うために必要な製品をインストールしてください。UNIX 版 Datareplicator のインストールは、日立 PP インストーラで行います。

HiRDB と Datareplicator のどちらを先にインストールしてもかまいませんが、Datareplicator の環境変数を設定するときは、同じサーバマシンに HiRDB がインストールされていることが前提です。

2.2.4 Datareplicator のアンインストール

UNIX 版 Datareplicator のアンインストールは、日立 PP インストーラで実施してください。

なお、インストール時に作成される/etc/hirdbds ディレクトリは、アンインストールを実施しても削除されません。不要な場合※は、手作業で削除してください。

注※

hdeinfoget, 及び hdsinfoget コマンドの実行に必要な情報が、hdestart, hdsstart, 及び hdeprep コマンド実行時に/etc/hirdbds ディレクトリに作成されます。Datareplicator のバージョンアップを目的に、既存バージョンをアンインストールする場合などは、削除しないで残しておくことをお勧めします。なお、誤って削除した場合でも、Datareplicator を再インストールした後に hdestart, hdsstart, 及び hdeprep コマンドを実行することで、ディレクトリ内のファイルは再作成されます。

2.3 ディレクトリ構成 (UNIX)

UNIX 版 Datareplicator をインストールすると、/opt、及び/etc ディレクトリ下のそれぞれに hirdbds/ というディレクトリが作成されて、その下に Datareplicator のディレクトリとファイルが格納されます。また、Datareplicator の動作環境を設定するために、各種の定義ファイルを作成する必要があります。

ここでは、UNIX 版 Datareplicator のディレクトリ構成について説明します。

2.3.1 Datareplicator のインストール時に作成されるディレクトリ

UNIX 版 Datareplicator のインストール時に作成されるディレクトリを図 2-1 に、ファイルの内容を表 2-5 に示します。このディレクトリは、抽出側 Datareplicator と反映側 Datareplicator で共通です。

図 2-1 Datareplicator に関連するディレクトリとファイル

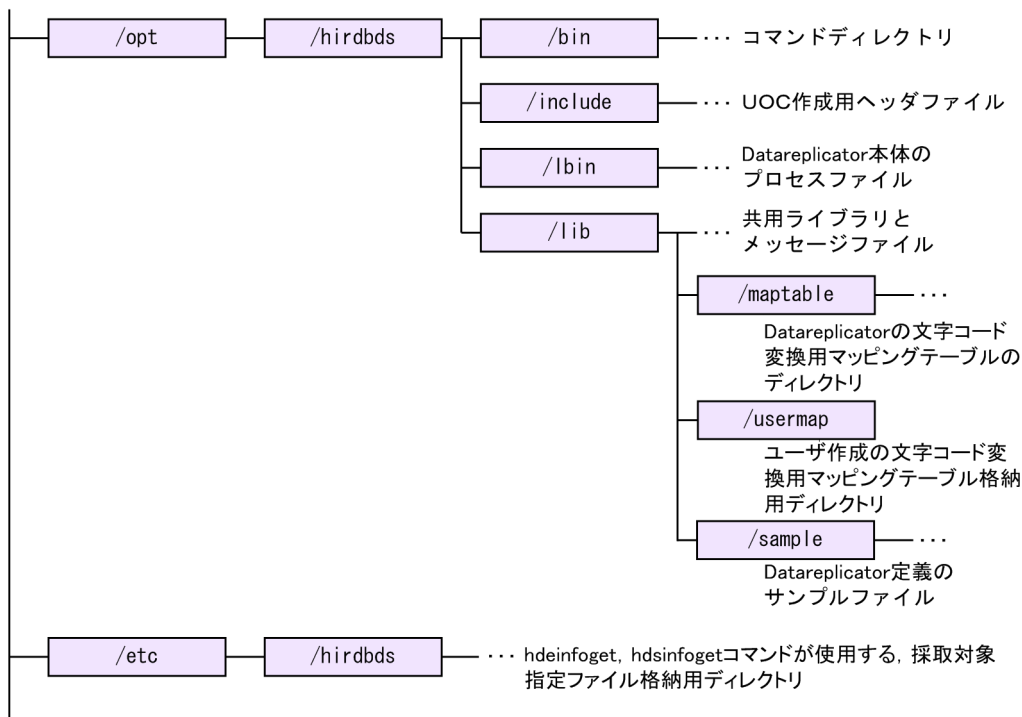


表 2-5 Datareplicator のインストール時に作成されるディレクトリとファイルの内容

作成タイミング	ディレクトリとファイル名	内容
インストール時	/opt/hirdbds/bin/	Datareplicator のコマンドがあるディレクトリ
	/opt/hirdbds/include/	UOC 作成用ヘッダファイル
	/opt/hirdbds/lbin/	Datareplicator 本体のプロセスファイル
	/opt/hirdbds/lib/	Datareplicator の共用ライブラリとメッセージファイル
	/opt/hirdbds/lib/mactable/	Datareplicator の文字コード変換用マッピングテーブルのディレクトリ

作成タイミング	ディレクトリとファイル名	内容
	/opt/hirdbds/lib/usermap/	ユーザが作成した文字コード変換用マッピングテーブルを格納するディレクトリ
	/opt/hirdbds/lib/sample/	Datareplicator 定義とユーザOWNコーディングのサンプルファイル※1
	/etc/hirdbds/	hdeinfoget, 及び hdsinfoget コマンドが使用するトラブルシューティング情報格納ディレクトリ※2

注※1

Datareplicator 定義テンプレートのファイル名については「[5.13.2 抽出側 Datareplicator の定義の例](#)」又は「[5.13.3 反映側 Datareplicator の定義の例](#)」を、ユーザOWNコーディングのサンプルのファイル名については「[8.1.7 反映情報編集 UOC のサンプル](#)」, 「[8.2.6 列データ編集 UOC のサンプル](#)」, 及び「[8.3.6 送信データ UOC のサンプル](#)」を参照してください。

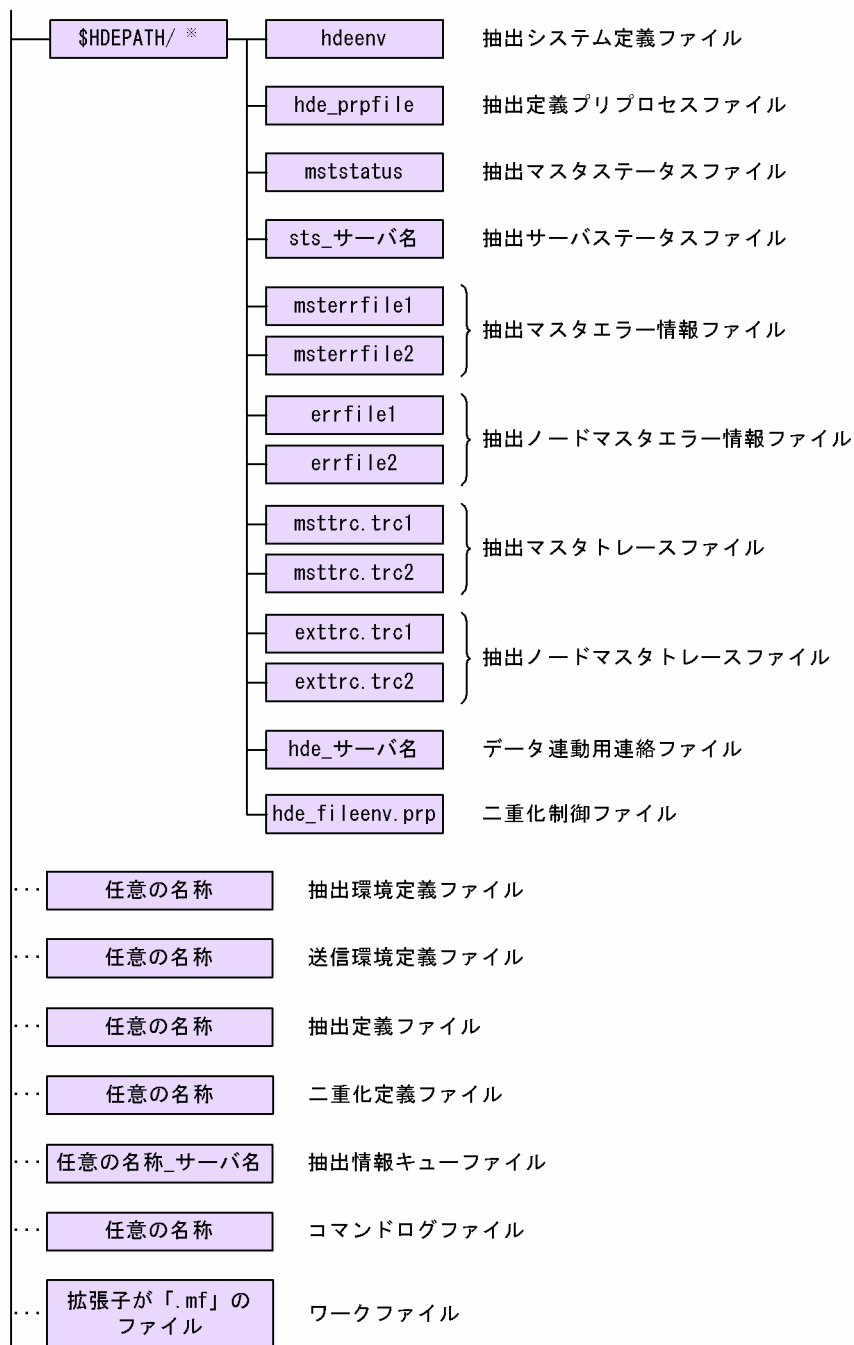
注※2

hdeinfoget, 及び hdsinfoget コマンドで収集するファイルの格納場所などに関する情報が、このディレクトリ内に作成されます。これらのファイルは、hdestart, hdsstart, 及び hdeprep コマンド実行時に自動的に作成されるため、直接修正しないでください。

2.3.2 抽出側 Datareplicator のディレクトリ構成

抽出側 Datareplicator のディレクトリ構成を [図 2-2](#) に、ファイルの内容を [表 2-6](#) に示します。

図 2-2 抽出側 Datareplicator のディレクトリ構成 (UNIX)



注※

抽出側システムで作成する、Datareplicator の定義を格納するディレクトリを示します。

表 2-6 抽出側 Datareplicator で作成するディレクトリとファイルの内容 (UNIX)

作成タイミング	ディレクトリとファイル名※1	内容
ユーザが作成	\$HDEPATH/hdeenv	抽出システム定義ファイル
	任意のディレクトリ/任意の名称	抽出環境定義ファイル
	任意のディレクトリ/任意の名称	送信環境定義ファイル

作成タイミング	ディレクトリとファイル名※1	内容
	任意のディレクトリ/任意の名称	抽出定義ファイル
	任意のディレクトリ/任意の名称	二重化定義ファイル
抽出側 Datareplicator の初期起動時	\$HDEPATH/hde_prpfile	抽出定義プリプロセスファイル
	任意のディレクトリ/任意の名称_サーバ名	抽出情報キューファイル
	\$HDEPATH/mststatus	抽出マスタステータスファイル
	\$HDEPATH/sts_サーバ名	抽出サーバステータスファイル
	\$HDEPATH/msterrfile1 \$HDEPATH/msterrfile2	抽出マスタエラー情報ファイル
	\$HDEPATH/errfile1 ※2 \$HDEPATH/errfile2	抽出ノードマスタエラー情報ファイル
	\$HDEPATH/msttrc.trc1 \$HDEPATH/msttrc.trc2	抽出マスタトレースファイル
	\$HDEPATH/exttrc.trc1 ※3 \$HDEPATH/exttrc.trc2	抽出ノードマスタトレースファイル
	\$HDEPATH/hde_サーバ名	データ連動用連絡ファイル
	任意のディレクトリ/任意の名称※4	コマンドログファイル
	\$HDEPATH/hde_fileenv.prp	二重化制御ファイル
任意のディレクトリ/拡張子が「.mf」のファイル※5	ワークファイル	

注※1

環境変数 HDEPATH は、抽出側 Datareplicator の定義を作成する任意のディレクトリを示します。

注※2

抽出システム定義の errfile_unique オペランドに true を指定すると、ファイル名に「_ホスト名」が付きます（ファイル名が「errfile1_ホスト名」と「errfile2_ホスト名」になります）。

注※3

抽出システム定義の errfile_unique オペランドに true を指定すると、ファイル名に「_ホスト名」が付きます（ファイル名が「exttrc_ホスト名.trc1」と「exttrc_ホスト名.trc2」になります）。

注※4

コマンドログファイル名は、環境変数 hde_command_log_file に絶対パス名又は相対パス名で指定します。相対パス名で指定すると、「\$HDEPATH/相対パス名」が絶対パス名として仮定されます。環境変数 hde_command_log_file の指定を省略すると、コマンドログファイル名として \$HDEPATH/hdecmdlog が仮定されます。

実際のコマンドログファイル名は、指定した名称の最後に「1」又は「2」が付けられます。実際のコマンドログファイル名の長さが「OS のパスの最大長-1」を超えないように指定してください。

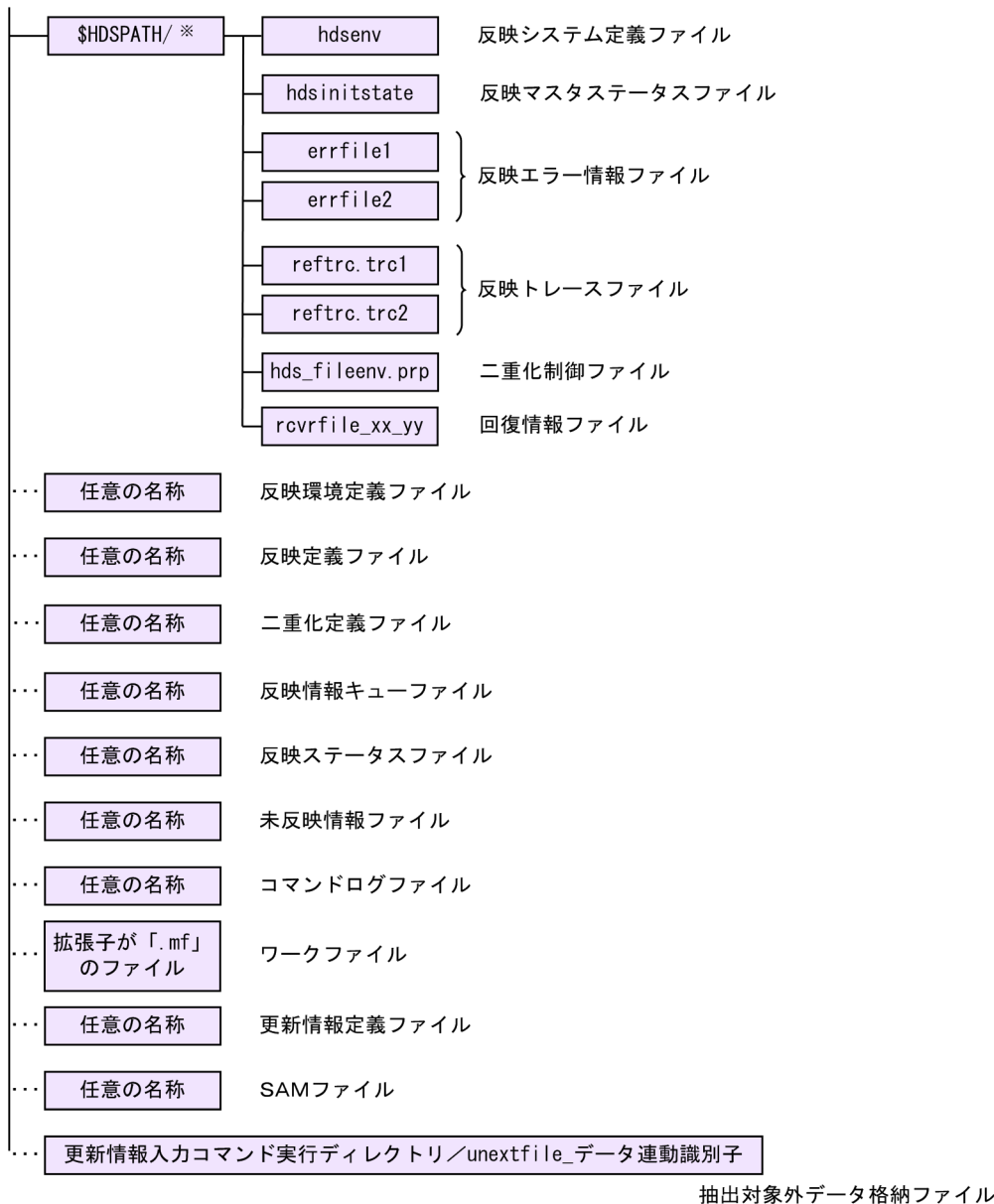
注※5

ファイルを二重化する場合に、Datareplicatorが自動生成するワークファイルです。Datareplicatorの稼働中には削除しないでください。

2.3.3 反映側 Datareplicator のディレクトリ構成

反映側 Datareplicator のディレクトリ構成を図 2-3 に、ファイルの内容を表 2-7 に示します。

図 2-3 反映側 Datareplicator のディレクトリ構成 (UNIX)



注※

反映側システムで作成する、Datareplicatorの定義を格納するディレクトリを示します。

表 2-7 反映側 Datareplicator で作成するディレクトリとファイルの内容 (UNIX)

作成タイミング	ディレクトリとファイル名※1	内容
ユーザが作成	\$HDSPATH/hdsenv	反映システム定義ファイル
	任意のディレクトリ/任意の名称	反映環境定義ファイル
	任意のディレクトリ/任意の名称	反映定義ファイル
	任意のディレクトリ/任意の名称	二重化定義ファイル
抽出側システムとの接続時	\$HDSPATH/rcvfile_xx_yy※2	回復情報ファイル
反映側	任意のディレクトリ/任意の名称	反映情報キューファイル
Datareplicatorの初期起動時	任意のディレクトリ/任意の名称	反映ステータスファイル
	\$HDSPATH/hdsinitstate	反映マスタステータスファイル
	\$HDSPATH/errfile1 \$HDSPATH/errfile2	反映エラー情報ファイル
	\$HDSPATH/reftrc.trc1 \$HDSPATH/reftrc.trc2	反映トレースファイル
	任意のディレクトリ/任意の名称	未反映情報ファイル
	任意のディレクトリ/任意の名称※3	コマンドログファイル
	\$HDSPATH/hds_fileenv.prp	二重化制御ファイル
	任意のディレクトリ/拡張子が [.mf]のファイル※4	ワークファイル
ユーザが作成	任意のディレクトリ/任意の名称	更新情報定義ファイル※5
ユーザが転送	任意のディレクトリ/任意の名称	SAM ファイル※6
更新情報入力コマンドの実行時	更新情報入力コマンドの実行ディレクトリ/unextfile_データ連動識別子	抽出対象外データ格納ファイル※7

注※1

環境変数 HDSPATH は、反映側 Datareplicator の定義を作成する任意のディレクトリを示します。

注※2

xx は反映側 Datareplicator 識別子

yy はデータ連動識別子

注※3

コマンドログファイル名は、環境変数 hds_command_log_file に絶対パス名又は相対パス名で指定します。相対パス名で指定すると、「\$HDSPATH/相対パス名」が絶対パス名として仮定されます。環境変数 hds_command_log_file の指定を省略すると、コマンドログファイル名として\$HDSPATH/hdscmdlog が仮定されます。

実際のコマンドログファイル名は、指定した名称の最後に「1」又は「2」が付けられます。実際のコマンドログファイル名の長さが「OS のパスの最大長-1」を超えないように指定してください。

注※4

ファイルを二重化する場合に、Datareplicator が自動生成するワークファイルです。Datareplicator の稼働中には削除しないでください。

注※5

更新情報定義ファイルは、SAM ファイルを使うメインフレーム側 DB のうち、PDMII E2 のデータを抽出する反映側 Datareplicator で作成します。それ以外の場合は、作成する必要はありません。

注※6

メインフレーム側の更新情報抽出 SAM ファイルが、ファイル転送プログラムで転送されてきます。SAM ファイルは、SAM ファイルを使うメインフレーム側 DB (PDMII E2, RDB1 E2) とデータ連動する反映側 Datareplicator の場合に作成します。それ以外の場合は必要ありません。

注※7

抽出対象外データ格納ファイルは、更新情報入力コマンドを実行するたびに作成／再作成されます。ファイル名は、unextfile_に更新情報入力コマンドの実行時に指定したデータ連動識別子が付いた名称になります。

2.4 環境変数の設定 (UNIX)

製品をインストール後、Datareplicator を実行する前に環境変数を設定する必要があります。ここでは、設定する環境変数を抽出側 Datareplicator と反映側 Datareplicator に分けて説明します。

2.4.1 抽出側 Datareplicator の環境変数

抽出側 Datareplicator の環境変数の設定内容と設定例について説明します。

(1) 抽出側 Datareplicator の環境変数の設定内容

抽出側 Datareplicator の環境変数は、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定します。

抽出側 Datareplicator の環境変数の設定内容を次の表に示します。

表 2-8 抽出側 Datareplicator の環境変数の設定内容

環境変数	設定内容
PATH	抽出側 Datareplicator のコマンドライブラリの名称を指定します。*1
LANG	抽出側 Datareplicator のメッセージや定義で使う文字コードを指定します。 抽出システム定義の dblocale 及び msglocale オペランドに指定した内容を基に、該当する文字コードを指定してください。
TZ	抽出側 Datareplicator のメッセージ出力時のタイムゾーンを指定します。
PDDIR*2*3	抽出側 HiRDB の運用ディレクトリを指定します。
PDCONFPATH*2*3	抽出側 HiRDB のシステム定義ファイルを格納するディレクトリを指定します。
PDNAMEPORT*2	抽出側 HiRDB のポート番号を指定します。
PDHOST*2	抽出側 HiRDB のホスト名を指定します。
PDUSER*2	抽出側 HiRDB の接続ユーザを指定します。
PDLANG*2	抽出側 HiRDB の文字コードが UTF-8 の場合、「UTF-8」を指定します。 抽出側 HiRDB の文字コードが UTF-8 以外の場合は、HiRDB のマニュアル「UAP 開発ガイド」の記載に従ってください。
SHLIB_PATH*2*3*4*5	\$PDDIR/lib を指定します。Linux (IPF) 版、及び HP-UX (IPF) 版を使用する場合には、 \$PDDIR/lib : \$PDDIR/client/lib を指定してください。 列データ編集 UOC を使用する場合は、列データ編集 UOC のライブラリ格納パスも合わせて指定してください。
HDEPATH	抽出側 Datareplicator の運用ディレクトリを指定します。 このディレクトリ下に抽出側 Datareplicator の定義ファイル、ステータスファイル、エラー情報ファイル、稼働トレースファイルを作成します。

環境変数	設定内容
	<p>注意</p> <ul style="list-style-type: none"> 指定する運用ディレクトリのパス名の長さは、64バイト以内にしてください。 指定する運用ディレクトリに表 2-9 で示す権限を設定してください。 抽出側 Datareplicator と反映側 Datareplicator が同じマシン上にある場合には、環境変数 HDEPATH と HDSPATH とで同じディレクトリを指定しないでください。 抽出側 HiRDB が複数のサーバマシンから構成されるパラレルサーバの場合、それぞれのサーバマシン上の抽出側 Datareplicator の運用ディレクトリを同一にしてください。
HDE_BIN_COL_MAXLEN	<p>Datareplicator 内部で扱う BLOB 型の列の定義長を、実際の定義長よりも小さい値で指定します (単位: キロバイト)。定義長は 2GB 以上でも実際に扱うデータは小さい BLOB 型の列の場合、表を再定義することなくデータ連動できるようになります。</p> <p>注意</p> <ul style="list-style-type: none"> この環境変数に指定された値は、hdeprep コマンド実行時に有効になります。 指定する BLOB 型の列の定義長は、262,143KB 以内にしてください。 この環境変数に指定された値は、抽出対象となるすべての BLOB 型の列に対して有効となります。 この環境変数に指定した定義長を超える BLOB 型の列が検出された場合の処理については、「6.4.3(4) 環境変数 HDE_BIN_COL_MAXLEN についての注意」を参照してください。
EXTSHM (AIX 版だけ)	<p>プロセス空間の共有メモリ領域数の制限をなくすことを示す「ON」を設定します。共有メモリが不足したときに設定します。不足していないときの設定は任意です。</p>
PSALLOC (AIX 版だけ)	<p>メモリ確保時に必要なページングスペースをすぐに確保することを示す「early」を設定します。AIX は、通常メモリ確保時にはページングスペースを確保しません。必ず設定してください。</p>
LDR_CNTRL (AIX 版だけ)	<p>MAXDATA=メモリサイズ</p> <p>プログラムを実行するのに必要なメモリサイズを設定します。メモリサイズは、0x20000000~0x80000000 の範囲で 2GB までです。メモリは 256MB 単位で設定します。ただし、メモリサイズは共有メモリと mmap 領域部分と共通なので、バランスを考慮する必要があります。</p> <p>共有メモリが不足したときに設定します。不足していないときの設定は任意です。メモリが不足したときは、0x20000000 を設定して起動します。それでも不足するときは 0x10000000 ずつ増やして再起動します。</p>
NODISCLAIM (AIX 版だけ)	<p>free() に対するコールの処理方法として、nodisclaim() の発行を抑止することを示す「true」を設定します。</p> <p>必ず設定してください。</p>

注※1

コマンドライブラリの名称は、 /opt/hirdbds/bin/ です。

注※2

抽出側 HiRDB の環境変数です。HiRDB の設定内容については、マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

注※3

抽出側 HiRDB がパラレルサーバで、さらにサーバマシンごとに異なる設定をしている場合には、抽出システム定義でサーバマシンごとに個別に定義できます。抽出システム定義での個別の定義を省略すると、そのサーバマシン下の抽出側 Datareplicator では、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定した環境変数の値に従います。

注※4

Linux 版の場合は LD_LIBRARY_PATH です。
AIX 版の場合は LIBPATH です。

注※5

この環境変数に 256 バイト以上の値を指定した場合、抽出側 Datareplicator は指定値不正エラーとなります。256 バイト以上の値を指定する必要がある場合には、抽出システム定義の node_shlibpath オペランドに、255 バイト以内で抽出側 Datareplicator に必要なライブラリパスだけを指定してください。node_shlibpath オペランドを指定することで、環境変数の指定値不正エラーが発生しなくなります。

また、node_shlibpath オペランドの指定値を有効にするためには、抽出側 Datareplicator を初期化する必要があります。将来的に、環境変数の指定値が 256 バイト以上となることが予想される場合には、初期構築時に node_shlibpath オペランドを指定しておくことをお勧めします。

表 2-9 抽出側 Datareplicator の運用ディレクトリに設定する権限

項番	実行ユーザ	設定する権限
1	抽出側 Datareplicator の使用者	読み込み権限 書き込み権限 実行権限
2	抽出側 Datareplicator の使用者と同一グループ	読み込み権限 書き込み権限 実行権限
3	その他のグループ	読み込み権限

(2) 抽出側 Datareplicator の環境変数の設定例

抽出側 Datareplicator の環境変数の設定例を次に示します。下記の例は、HP-UX 版 Datareplicator の場合で説明します。

• Bourne シェル (sh) の場合の例

```
$ PATH=$PATH:/opt/hirdbds/bin  
$ LANG=ja_JP.SJIS※  
$ HDEPATH=/opt/hirdbds/define  
$ export PATH LANG HDEPATH
```

• C シェル (csh) の場合の例

```
% set path=($path /opt/hirdbds/bin)
% setenv LANG ja_JP.SJIS※
% setenv HDEPATH /opt/hirdbds/define
```

注※

環境変数 LANG の設定方法については、該当する OS のマニュアルを参照してください。

この例では、抽出側 Datareplicator 定義を /opt/hirdb/define/ディレクトリに格納していることを想定しています。また、TZ 環境変数と HiRDB の環境変数の指定は省略しています。

TZ 環境変数の設定方法については、該当する OS のマニュアルを参照してください。

抽出側 HiRDB の環境変数 (\$PDDIR, \$PDCONFPATH, \$PDNAMEPORT, \$PDHOST, \$PDUSER, 及び \$SHLIB_PATH) の設定例については、マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

2.4.2 反映側 Datareplicator の環境変数

反映側 Datareplicator の環境変数の設定内容と設定例について説明します。

(1) 反映側 Datareplicator の環境変数の設定内容

反映側 Datareplicator の環境変数は、反映側 Datareplicator のコマンドを実行するユーザの環境に対して設定します。

反映側 Datareplicator の環境変数の設定内容を次の表に示します。

表 2-10 反映側 Datareplicator の環境変数の設定内容

環境変数	設定内容
PATH	反映側 Datareplicator のコマンドライブラリの名称を指定します。*1
LANG	反映側 Datareplicator のメッセージや定義で使う文字コードを指定します。 反映システム定義の dblocale 及び msglocale オペランドに指定した内容を基に、該当する文字コードを指定してください。
TZ	反映側 Datareplicator のメッセージ出力時のタイムゾーンを指定します。
PDDIR*2	反映側 HiRDB の運用ディレクトリを指定します。
PDNAMEPORT*2	反映側 HiRDB のポート番号を指定します。
PDHOST*2	反映側 HiRDB のホスト名を指定します。
PDLANG*2	反映側 HiRDB の文字コードが UTF-8 の場合、「UTF-8」を指定します。 反映側 HiRDB の文字コードが UTF-8 以外の場合は、HiRDB のマニュアル「UAP 開発ガイド」の記載に従ってください。

環境変数	設定内容
SHLIB_PATH ^{※2※3}	\$PDDIR/lib を指定します。Linux (IPF) 版、及び HP-UX (IPF) 版を使用する場合には、\$PDDIR/lib : \$PDDIR/client/lib を指定してください。
HDSPATH	<p>反映側 Datareplicator の運用ディレクトリを指定します。このディレクトリ下に反映側 Datareplicator の定義ファイル、ステータスファイル、エラー情報ファイル、稼働トレースファイルを作成します。</p> <p>注意</p> <ul style="list-style-type: none"> 指定する運用ディレクトリのパス名の長さは、64 バイト以内にしてください。 指定する運用ディレクトリに表 2-11 で示す権限を設定してください。 抽出側 Datareplicator と反映側 Datareplicator が同じマシン上にある場合には、環境変数 HDEPATH と HDSPATH とで同じディレクトリを指定しないでください。
HDS_MST_STDCLOSE	<p>反映側 Datareplicator で、標準入力、標準出力、及び標準エラー出力をクローズするかどうかを指定します。</p> <p>この環境変数を省略又は「TRUE」を指定： 反映側 Datareplicator で、標準入力、標準出力、及び標準エラー出力をクローズします。</p> <p>「FALSE」を指定： 反映側 Datareplicator で、標準入力、標準出力、及び標準エラー出力をクローズしません。</p> <p>「FALSE」を指定した場合、反映情報編集 UOC で標準入力、標準出力、及び標準エラー出力を使用できます。</p> <p>この環境変数を設定しないまま反映情報編集 UOC で標準入力、標準出力、又は標準エラー出力を使用した場合、反映が正常に実行されないおそれがあります。反映情報編集 UOC で標準入力、標準出力、又は標準エラー出力を使用する場合は、必ずこの環境変数に「FALSE」を設定してください。</p>
HDS_RFI_ELANG ^{※4}	<p>hdsrefinfmt コマンドを実行する際の抽出側システムの文字コードを指定します。</p> <p>指定されていない場合は、反映ステータスファイルを基に自動判別します。ただし、反映側の初期化直後に定義情報を表示する場合は、自動判別できません。自動判別できない場合は、ja_JP.SJIS を仮定します。</p>
HDS_RFI_PLANG ^{※4}	<p>hdsrefinfmt コマンドを実行する際の反映側システムの文字コードを指定します。</p> <p>指定されていない場合は、ja_JP.SJIS を仮定します。</p>
HDSCLTWAITTIME	<p>HiRDB のクライアント環境定義 PDCWAITTIME (クライアントの最大待ち時間) と PDSWAITTIME (サーバの最大待ち時間) ^{※5} の設定値を、反映側 Datareplicator で有効にするかどうかを指定します。</p> <p>この環境変数の指定は任意のため、指定する必要がある場合は省略することをお勧めします。</p> <p>「USER」を指定： HiRDB のクライアント環境定義の値を有効にします。</p> <p>「REPL」を指定： HiRDB のクライアント環境定義の値を無効にして、次の値を設定します。</p> <ul style="list-style-type: none"> PDCWAITTIME 0 (応答があるまで待ち続ける) を設定します。 PDSWAITTIME 反映システム定義 commit_wait_time オペランドの値^{※6} + 600 (秒) を設定します。 反映トランザクション同期機能を使用している場合は、反映システム定義 syncwait_limit_time オペランドの値 + 600 (秒) を設定します。

環境変数	設定内容
	<p>なお、設定値が 65,535 を超える場合は 65,535 が設定されます。</p> <p>この環境変数を省略した場合</p> <p>反映情報編集 UOC を実行するときは「USER」の指定値が仮定され、反映 SQL プロセスを実行するときは「REPL」の指定値が仮定されます。</p> <p>「USER」を指定する場合</p> <p>HiRDB のクライアント環境定義 PDCWAITTIME と PDSWAITTIME には、次のオペランドの指定値より大きな値を設定してください。</p> <ul style="list-style-type: none"> • PDCWAITTIME HiRDB の pd_lck_wait_timeout オペランド • PDSWAITTIME 反映システム定義の commit_wait_time オペランド及び syncwait_limit_time オペランド <p>PDCWAITTIME と PDSWAITTIME に設定した値が小さいと、HiRDB の監視時間を超過してしまうことがあります。この場合、反映側 Datareplicator のプロセスは SQL エラーを検知して異常終了します。</p>
EXTSHM (AIX 版だけ)	<p>プロセス空間の共有メモリ領域数の制限をなくすことを示す「ON」を設定します。 共有メモリが不足したときに設定します。不足していないときの設定は任意です。</p>
PSALLOC (AIX 版だけ)	<p>メモリ確保時に必要なページングスペースをすぐに確保することを示す「early」を設定します。 AIX は、通常メモリ確保時にはページングスペースを確保しません。 必ず設定してください。</p>
LDR_CNTRL (AIX 版だけ)	<p>MAXDATA=メモリサイズ プログラムを実行するのに必要なメモリサイズを設定します。メモリサイズは、0x20000000～0x80000000 の範囲で 2GB までです。メモリは 256MB 単位で設定します。ただし、メモリサイズは共有メモリと mmap 領域部分と共通なので、バランスを考慮する必要があります。 共有メモリが不足したときに設定します。不足していないときの設定は任意です。メモリが不足したときは、0x20000000 を設定して起動します。それでも不足するときは 0x10000000 ずつ増やして再起動します。</p>
NODISCLAIM (AIX 版だけ)	<p>free()に対するコールの処理方法として、nodisclaim()の発行を抑止することを示す「true」を設定します。 必ず設定してください。</p>
HDS_SHM_MODE (HP-IPF 版だけ)	<p>反映側 Datareplicator で確保する共用メモリのアドレッシングモードを指定します。 この環境変数は、反映側 Datareplicator のコマンドを実行するユーザの環境に対して設定してください。反映情報編集 UOC を使用しない場合、又は 64 ビット版の反映情報編集 UOC だけを使用する場合は「64」を指定してください。*7 32 ビット版の反映情報編集 UOC を使用する場合は、省略又は「64」以外を指定してください。 「64」を指定した場合 反映側 Datareplicator で使用する共用メモリを 64 ビットアドレッシングモードの空間に確保します。 省略又は「64」以外を指定 反映側 Datareplicator で使用する共用メモリを 32 ビット/64 ビット互換アドレッシングモードの空間に確保します。</p>

注※1

コマンドライブラリの名称は、 /opt/hirdbds/bin/ です。

注※2

反映側 HiRDB の環境変数です。環境変数の設定内容については、マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

注※3

Linux 版の場合は LD_LIBRARY_PATH です。
AIX 版の場合は LIBPATH です。

注※4

HDS_RFI_ELANG, 及び HDS_RFI_PLANG の指定値を次に示します。

環境変数	文字コード			
	JIS8/シフト JIS	EUC	Unicode (UTF-8)	EBCDIC/KEIS EBCDIK/KEIS
HDS_RFI_ELANG	ja_JP.SJIS	ja_JP.UJIS	ja_JP.UTF8	EBCDIK
HDS_RFI_PLANG	ja_JP.SJIS	ja_JP.UJIS	ja_JP.UTF8	—

(凡例)

— : 該当しません。

注※5

HiRDB のクライアント環境定義 PDSWATCHTIME は対象外です。HiRDB のクライアント環境定義に指定した値が常に有効となります。

注※6

反映環境定義の commit_wait_time オペランドを指定している場合は、反映環境定義に指定した値が有効になります。

注※7

HDS_SHM_MODE を省略, 又は「64」以外を指定した場合, 共用メモリを 32 ビット/64 ビット互換アドレッシングモードの空間に確保します。この空間は 2GB 以下の領域に確保され, ほかの 32 ビット/64 ビット互換 PP や AP がその領域で動作する環境では, OS 全体のメモリに余裕があってもメモリ不足が発生することがあります。

表 2-11 反映側 Datareplicator の運用ディレクトリに設定する権限

項番	実行ユーザ	設定する権限
1	抽出側 Datareplicator の使用者	読み込み権限 書き込み権限 実行権限
2	抽出側 Datareplicator の使用者と同一グループ	読み込み権限
3	その他のグループ	読み込み権限

(2) 反映側 Datareplicator の環境変数の設定例

反映側 Datareplicator の環境変数の設定例を次に示します。

下記の例は、HP-UX 版 Datareplicator の場合で説明します。

- Bourne シェル (sh) の場合の例

```
$ PATH=$PATH:/opt/hirdbds/bin
$ LANG=ja_JP.SJIS※
$ HDSPATH=/opt/hirdbds/define
$ HDSCLTWAITTIME=USER
$ export PATH LANG HDSPATH HDSCLTWAITTIME
```

- C シェル (csh) の場合の例

```
% set path=($path /opt/hirdbds/bin)
% setenv LANG ja_JP.SJIS※
% setenv HDSPATH /opt/hirdbds/define
% setenv HDSCLTWAITTIME USER
```

注※

環境変数 LANG の設定方法については、該当する OS のマニュアルを参照してください。

この例では、反映側 Datareplicator 定義を /opt/hirdb/define/ディレクトリに格納していることを想定しています。また、TZ 環境変数と HiRDB の環境変数の指定は省略しています。

TZ 環境変数の設定方法については、該当する OS のマニュアルを参照してください。

反映側 HiRDB の環境変数 (\$PDDIR, \$PDCONFPATH, \$PDNAMEPORT, \$PDHOST, \$SHLIB_PATH) の設定例については、マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

2.5 通信環境の設定 (UNIX)

Datereplicator を実行する前に、通信環境を設定する必要があります。ここでは、通信環境の設定内容について説明します。

2.5.1 抽出側 Datereplicator の通信環境の設定

抽出側 Datereplicator の通信環境の設定内容と設定例を説明します。

(1) 抽出側 Datereplicator の通信環境の設定内容

抽出側 Datereplicator の通信環境の設定内容を次の表に示します。

表 2-12 抽出側 Datereplicator の通信環境の設定内容

通信環境	設定内容	設定するサーバマシン	
		抽出側 HiRDB がシングルサーバの場合	抽出側 HiRDB がパラレルサーバの場合
/etc/services	反映側システムとの通信で使うサービス名とポート番号 ^{*1} を指定します。ここで指定したサービス名を送信環境定義の hdeservice オペランドで指定します。ポート番号は、反映側システムと同じにしてください。	SDS	BES
	抽出マスタプロセスと抽出ノードマスタプロセスとの通信で使うサービス名とポート番号 ^{*1} を指定します。ここで指定したサービス名を抽出システム定義の mstservice オペランドで指定します。ポート番号は、システムマネージャのサーバとバックエンドサーバのサーバとで同じにしてください。	SDS	MGR, BES ^{*2}
/etc/hosts	反映側システムとの通信で使う IP アドレスとホスト名を指定します。ここで指定したホスト名を送信環境定義の hdehost オペランドで指定します。	SDS	BES
/etc/inetd.conf	抽出ノードマスタプロセスを起動するためのエントリを登録します。	SDS	MGR, BES

(凡例)

SDS：抽出側 HiRDB でシングルサーバを定義したサーバマシン

MGR：抽出側 HiRDB でシステムマネージャを定義したサーバマシン

BES：抽出側 HiRDB でバックエンドサーバ（抽出対象の DB がないバックエンドサーバも含む）を定義したサーバマシン

注※1

既に services ファイルに登録してあるポート番号や、ほかのソフトウェアで使うポート番号は指定できません。

注※2

抽出側 HiRDB でシステムマネージャを定義したサーバマシン、及びバックエンドサーバを定義したすべてのサーバマシンに対して設定します。

(2) 抽出側 Datareplicator の通信環境の設定例

抽出側 Datareplicator の通信環境の設定例を説明します。

(a) 抽出側 Datareplicator での/etc/services の設定例

/etc/services にサービス名とポート番号を設定する例を次に示します。

```
サービス名 ポート番号/tcp
```

注 下線部分は固定の値です。

- サービス名
ユーザ任意の名称を指定します。
- ポート番号
通信環境内で統一した値を指定します。ただし、ほかのサービスで使っていない値を指定してください。

(b) 抽出側 Datareplicator での/etc/hosts の設定例

/etc/hosts に IP アドレスとホスト名を設定する例を次に示します。

```
IPアドレス ホスト名
```

- IP アドレス
ユーザ任意のアドレスを指定します。
- ホスト名
ユーザ任意の名称を指定します。

(c) 抽出側 Datareplicator での/etc/inetd.conf の設定例

/etc/inetd.conf に抽出ノードマスタプロセスを起動するためのエントリを登録する例を次に示します。

HP-UX 版の場合

```
hdemaster stream tcp nowait user1 /opt/hirbds/lbin/hdenodemst hdenodemst
```

注

下線部分は固定の値です。上記のコマンド例は HP-UX 版 Datareplicator の場合の例です。なお、/etc/inetd.conf の設定の変更を有効にするには、次のコマンドを実行してください。

- kill -HUP inetd のプロセス ID

AIX 版の場合

```
hdemaster stream tcp nowait user1 /bin/env env [EXTSHM=ON] PSALLOC=early
NODISCLAIM=true
[LDR_CNTRL=MAXDATA=0x.....] /opt/hirdbds/lbin/hdenodemst
```

注

下線部分は固定の値です。

なお、/etc/inetd.conf の設定の変更を有効にするには、次のコマンドを実行してください。

- kill -HUP inetd のプロセス ID

Linux 版の場合

Linux 版の場合、/etc/inetd.conf を/etc/xinetd.conf に、inetd プロセスを xinetd プロセスに読み替えてください。

Linux 版の場合、次のどちらかの方法で設定します。

- /etc/xinetd.conf ファイルに記述する。
- /etc/xinetd.d ディレクトリにファイルを作成し登録する。

次に設定例を示します。下線部分をお使いの環境に合わせて修正してください。

```
service hdemaster
{
    socket_type      = stream
    protocol        = tcp
    wait            = no
    user            = user1
    server          = /opt/hirdbds/lbin/hdenodemst
}
```

注

設定の変更を有効にするには、次のコマンドを実行してください。

- /sbin/service xinetd reload
- **hdemaster**
抽出システム定義の mstservice で指定するサービス名を指定します。
- **user1**
抽出側 HiRDB のユーザと同じグループに属する、抽出側 Datareplicator の初期化 (hdestart -i コマンド) を実行するユーザ名を指定します。

2.5.2 反映側 Datareplicator の通信環境の設定

反映側 Datareplicator の通信環境の設定内容と設定例を説明します。

反映側 Datareplicator の通信環境は、反映側 Datareplicator のコマンドを実行するサーバマシンに対して設定します。

(1) 反映側 Datareplicator の通信環境の設定内容

反映側 Datareplicator の通信環境の設定内容を次の表に示します。

表 2-13 反映側 Datareplicator の通信環境の設定内容

通信環境	設定内容
/etc/services	抽出側システムとの通信で使うサービス名とポート番号を指定します。ここで指定したサービス名を反映システム定義の hdsservice オペランドで指定します。ポート番号は、抽出側システムと同じにしてください。
XNF のネットワーク定義	OSI プロトコル（チャンネル接続）で通信する場合※に、抽出側システムとの通信で使う UCE 名と T セレクタ値を指定します。ここで指定した T セレクタ値を反映システム定義の reflect_tselector オペランドに指定します。T セレクタ値は、抽出側システムと同じにしてください。

注※

通信プロトコルに OSI プロトコル（チャンネル接続）を使えるのは、OS が HP-UX の場合だけです。OS が AIX, Linux, 又は Windows の場合は、使用できる通信プロトコルは TCP/IP だけです。

(2) 反映側 Datareplicator の通信環境の設定例

反映側 Datareplicator の通信環境の設定例を説明します。

TCP/IP のネットワーク環境の設定については、マニュアル「HI-UX/WE2 日立 CSMA/CD ネットワーク CD105 (TCP/IP)」を参照してください。XNF のネットワーク定義については、マニュアル「通信管理 XNF/S-E2 構成定義編」を参照してください。

(a) 反映側 Datareplicator での/etc/services の設定例

/etc/services にサービス名とポート番号を設定する例を次に示します。

サービス名 <u>ポート番号</u> /tcp

注 下線部分は固定の値です。

- サービス名
任意の名称を指定します。
- ポート番号
通信環境内で統一した値を指定します。ただし、ほかのサービスで使っていない値を指定してください。

2.6 Datareplicator のインストール (Windows)

Windows 版 Datareplicator をインストールする方法について説明します。

2.6.1 インストール前の準備

Datareplicator をインストールする前に、次に示す作業が必要になります。

- 稼働環境の確認
- ユーザ権限の確認

(1) 稼働環境の確認

次に示す内容を確認してください。

- Datareplicator をインストールするマシンのディスクの空き容量が十分であることを確認してください。Datareplicator に必要なディスク容量については、「[4.6.8 抽出側 Datareplicator のリソースの設計](#)」又は「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。
- Datareplicator が動作するのに必要な製品がインストールされていることを確認してください。Datareplicator が動作するのに必要な製品については、「[2.1 Datareplicator に関連する製品](#)」を参照してください。

(2) ユーザ権限の確認

インストールを実行するユーザに Administrator の権限があることを確認してください。

2.6.2 インストールするサーバマシン

Datareplicator をインストールするサーバマシンは、HiRDB システムが HiRDB/シングルサーバの場合と HiRDB/パラレルサーバの場合で異なります。さらに HiRDB/パラレルサーバの場合は、抽出側システムと反映側システムで異なります。

Datareplicator をインストールするサーバマシンを次の表に示します。

表 2-14 Datareplicator をインストールするサーバマシン

HiRDB システムの種類	抽出側システムの場合	反映側システムの場合
HiRDB/シングルサーバ	シングルサーバがあるサーバマシンに Datareplicator をインストールします。	シングルサーバがあるサーバマシンに Datareplicator をインストールします。

HiRDB システムの種類	抽出側システムの場合	反映側システムの場合
HiRDB/パラレルサーバ	次のサーバがあるサーバマシンすべてに Datareplicator をインストールします。 <ul style="list-style-type: none"> システムマネージャ (MGR) バックエンドサーバ (BES) 	HiRDB クライアント機能が動作するサーバマシンに、一つの Datareplicator をインストールします。

2.6.3 インストールの実行

「[2.1 Datareplicator に関連する製品](#)」を参照して、Windows 版 Datareplicator を使うために必要な製品をインストールしてください。

HiRDB と Datareplicator のどちらを先にインストールしてもかまいませんが、Datareplicator の環境変数を設定するときは、同じサーバマシンに HiRDB がインストールされていることが前提です。

Datareplicator のインストールが完了した後は、いったん Windows を再起動する必要があります。

Datareplicator をインストールする前に Datareplicator の環境変数を設定しておく、Windows の再起動が 1 回で済みます。Datareplicator の環境変数を設定する方法については、「[2.8 環境変数の設定 \(Windows\)](#)」を参照してください。

(1) インストールの実行手順

インストール用の統合 CD-ROM からインストールする手順を次に示します。インストールを実行する前に、Datareplicator とすべての Windows アプリケーションを終了させてください。

1. HiRDB の統合 CD-ROM を挿入して、hcd_inst.exe を実行します。

日立総合インストーラが起動します。

2. 画面の指示に従って [次へ>] ボタンをクリックします。

3. 名前と会社名を入力して、[次へ>] ボタンをクリックします。

表示されている名前と会社名をそのまま使う場合は、入力不要です。

4. インストール先ディレクトリを指定し、[次へ>] ボタンをクリックします。

標準のインストール先は、次に示すディレクトリです。

Windows のインストール先ドライブ：¥Program Files¥HITACHI¥hirdbds

上記のディレクトリ以外にインストールする場合は、[参照...] ボタンをクリックして、インストール先のディレクトリ^{※1}を指定します。指定したディレクトリがない場合、作成するかどうかを確認するメッセージが表示されます。

5. セットアップ方法を選択し、[次へ>] ボタンをクリックします。

- [標準] を選択すると、反映側 Datareplicator、抽出側 Datareplicator、運用管理^{※2}がインストールされます。

- [コンパクト] を選択すると、反映側 Datareplicator, 抽出側 Datareplicator がインストールされます。
 - [カスタム] を選択すると、インストールするコンポーネントを選べます。
6. 登録するプログラムフォルダ名又はグループ名として [HiRDB Datareplicator] が表示されるので、[次へ>] ボタンをクリックします。
- 表示されている名前以外のグループを登録する場合は、名前を指定します。また、既存のグループに登録する場合は、既存のグループの中から選択します。
7. ファイルのコピーを開始する前に現在の設定が表示されるので、[次へ>] ボタンをクリックします。Datareplicator のインストールが開始します。
8. Datareplicator のインストールが完了したら、いったん Windows を再起動します。

注※1

指定できるドライブは、ローカルドライブだけです。ネットワークドライブは指定しないでください。

注※2

既に運用管理をインストールしている状態でインストールすると、ファイルのコピーでエラーになることがあります。エラーになったときは、SNMP サービスをいったん停止してから再インストールしてください。

2.6.4 インストール時に登録される内容

インストール時に登録される内容を次に示します。

(1) サービス

インストール時に登録されるサービスの種類、及び [コントロールパネル] の [サービス] アイコンを開いたときに表示される名称とスタートアップの設定を次の表に示します。

表 2-15 インストール時に登録されるサービス

種 類	表示される名称	スタートアップの種類※	ログオン※
抽出サービス	HiRDB Datareplicator (Source Site)	手動	システムアカウント
ノードマスタプロセス起動サービス	HiRDB Datareplicator (Source Site NMT)	手動	システムアカウント
反映サービス	HiRDB Datareplicator (Target Site)	手動	システムアカウント
状態監視サービス	HiRDB Datareplicator (Agent)	手動	システムアカウント

注※

インストール後にコントロールパネルの [サービス] アイコンを開いて変更できます。システムアカウント以外に Administrators 権限を持つユーザのアカウントに変更できます。

(2) アイコン

[HiRDB Datareplicator] フォルダ又はグループに、はじめにお読みいただくファイル (Datareplicator のインストールディレクトリ下の¥Readme.txt) のアイコンが登録されます。

アンインストールは、[コントロールパネル] の [アプリケーションの追加と削除] アイコンを使うため、アンインストールプログラムのアイコンは登録されません。

(3) システムの環境変数

環境変数 PATH に Datareplicator のコマンドライブラリの名称が追加されます。コマンドライブラリの名称を次に示します。

Datareplicatorのインストールディレクトリ¥bin

2.6.5 インストール完了後に必要な作業

抽出側 Datareplicator の場合、インストール完了後に次に示す作業をしてください。

- ノードマスタプロセス起動サービスのサービス名 [hdenmserv] を、Windows のシステムディレクトリ¥drivers¥etc¥services (これ以降、services ファイルと表記します) に設定します。
設定方法と形式については、「[2.9 通信環境の設定 \(Windows\)](#)」を参照してください。
- ノードマスタプロセス起動サービスが自動で開始するように、次に示す手順で設定します。
 1. [コントロールパネル] にある [サービス] アイコンを開きます。
 2. [サービス] から「HiRDB Datareplicator (Source Site NMT)」を選択します。
 3. [スタートアップ] ボタンをクリックします。
 4. [サービス] の [スタートアップの種類] で [自動] を選択します。
 5. [OK] ボタンをクリックします。

■ 注意事項

services ファイルは、Windows を終了又は再起動する前に必ず設定してください。設定しないと、Windows の起動時にノードマスタプロセス起動サービスがエラーになります。

また、抽出側 HiRDB が HiRDB/パラレルサーバのときは、抽出側 Datareplicator が動作するすべての HiRDB ノード上で、ノードマスタ起動サービスの自動起動を実行してください。

2.6.6 Datareplicator のアンインストール

Datareplicator をアンインストールする手順を次に示します。

アンインストールを実行する前に、[HiRDB Datareplicator] のサービスはすべて終了させてください。

1. [コントロールパネル] にある [アプリケーションの追加と削除] アイコンを開きます。
2. [インストールと削除] タブを選択します。
3. インストールされているアプリケーションの一覧から [HiRDB Datareplicator] を選択して、[追加と削除...] ボタンをクリックします。
4. 画面の指示に従って、[はい] ボタンをクリックします。
5. アンインストールが終了したら、[OK] ボタンをクリックします。

2.7 ディレクトリ構成 (Windows)

Windows 版 Datareplicator が使うファイルには、インストール時に自動的に作成されるファイル、システム環境を設定する定義ファイル、及び Datareplicator の初期起動時に自動的に作成されるファイルがあります。

ここでは、Windows 版 Datareplicator のディレクトリ構成について説明します。

2.7.1 Datareplicator のインストール時に作成されるディレクトリ

Windows 版 Datareplicator のインストール時に作成されるディレクトリを図 2-4 に、ファイルの内容を表 2-16 に示します。このディレクトリは、抽出側 Datareplicator と反映側 Datareplicator で共通です。

図 2-4 Datareplicator に関連するディレクトリとファイル

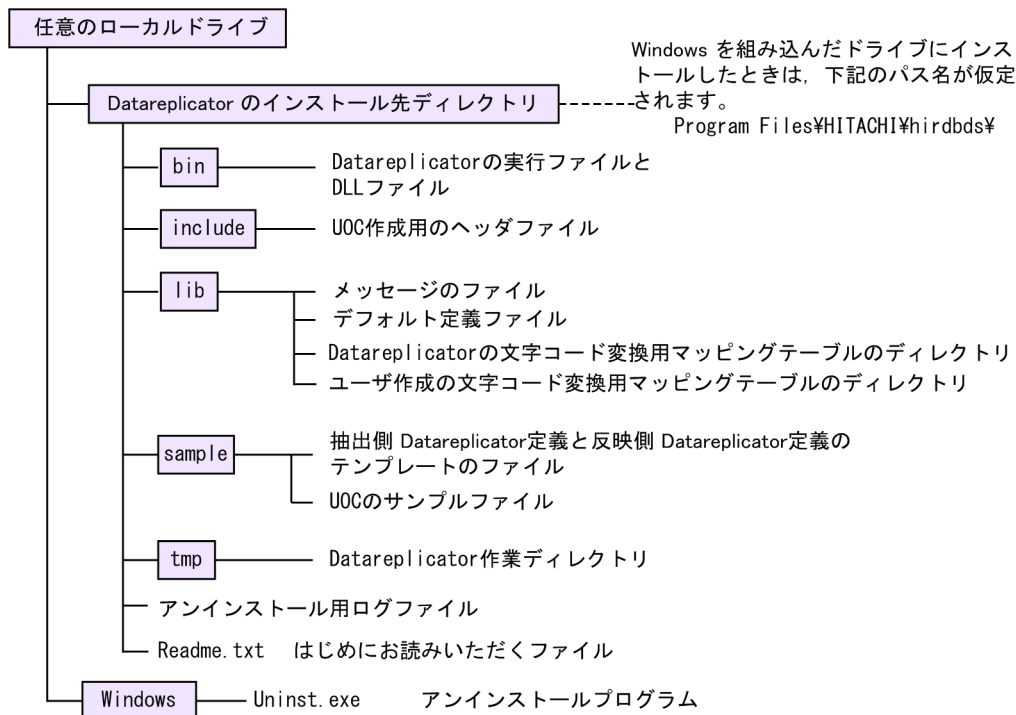


表 2-16 Datareplicator のインストール時に作成されるディレクトリとファイルの内容

作成タイミング	ディレクトリとファイル名	内容
インストール時	¥bin¥	Datareplicator のコマンドがあるディレクトリ
	¥include¥	UOC 作成用ヘッダファイル
	¥lib¥	Datareplicator の共用ライブラリとメッセージファイル
	¥lib¥maptable¥	Datareplicator の文字コード変換用マッピングテーブルのディレクトリ

作成タイミング	ディレクトリとファイル名	内容
	¥lib¥usermap¥	ユーザが作成した文字コード変換用マッピングテーブルを格納するディレクトリ
	¥sample¥	Datareplicator 定義のサンプルとユーザOWNコーディングのサンプル※
	¥tmp¥	Datareplicator の作業用ディレクトリ

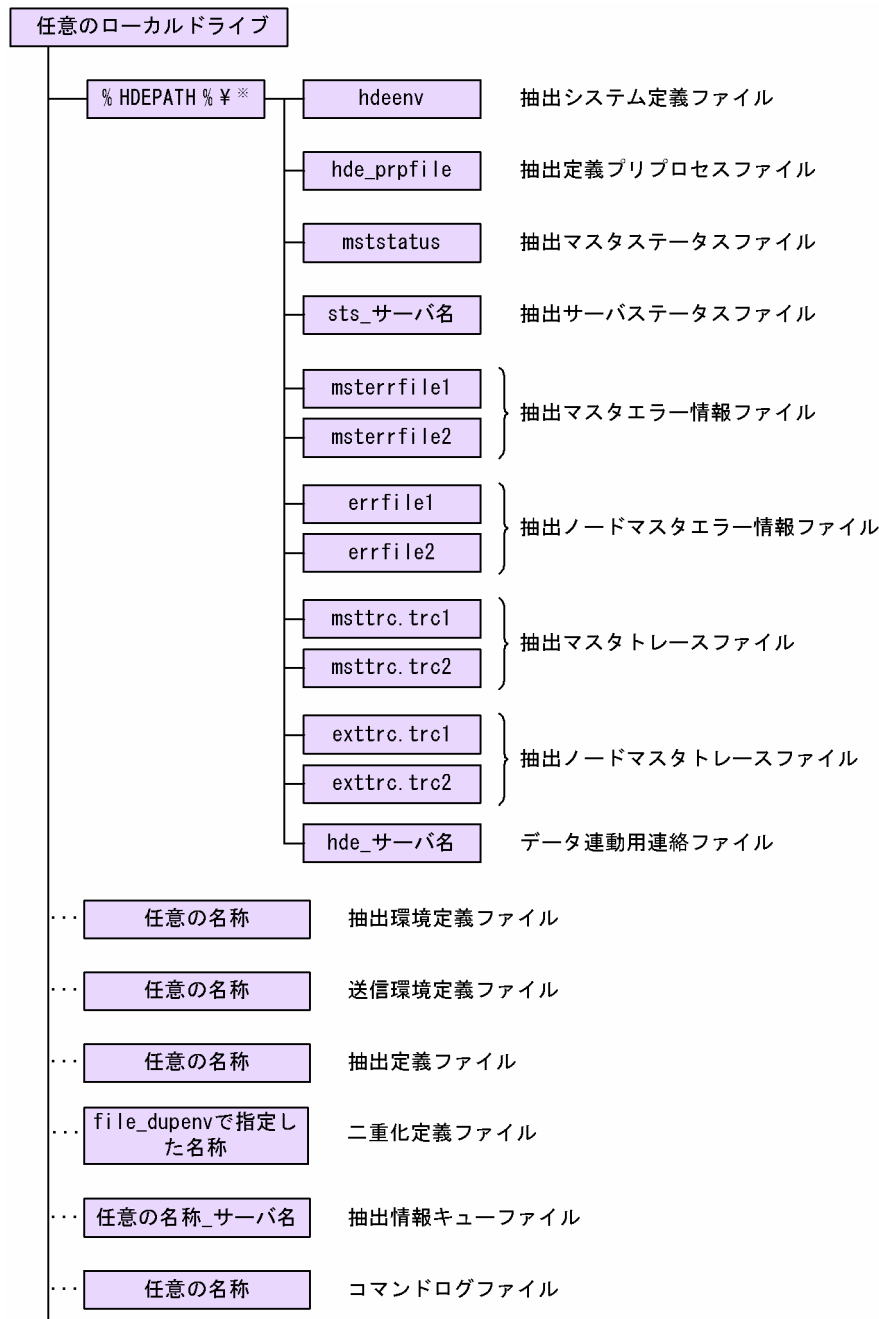
注※

Datareplicator 定義テンプレートのファイル名については「[5.13.2 抽出側 Datareplicator の定義の例](#)」又は「[5.13.3 反映側 Datareplicator の定義の例](#)」を、ユーザOWNコーディングのサンプルのファイル名については「[8.1.7 反映情報編集 UOC のサンプル](#)」, 「[8.2.6 列データ編集 UOC のサンプル](#)」, 及び「[8.3.6 送信データ UOC のサンプル](#)」を参照してください。

2.7.2 抽出側 Datareplicator のディレクトリ構成

抽出側 Datareplicator のディレクトリ構成を [図 2-5](#) に、ファイルの内容を [表 2-17](#) に示します。

図 2-5 抽出側 Datareplicator のディレクトリ構成 (Windows)



注※

Datareplicator の定義を格納する、任意のディレクトリを示す環境変数です。

表 2-17 抽出側 Datareplicator のディレクトリとファイルの内容

作成タイミング	ディレクトリとファイル名※1※2	内容
ユーザが作成	%HDEPATH%\hdeenv	抽出システム定義ファイル
	任意の名称	抽出環境定義ファイル
	任意の名称	送信環境定義ファイル

作成タイミング	ディレクトリとファイル名※1※2	内容
	任意の名称	抽出定義ファイル
	抽出システム定義ファイルの file_dupenv オペランドに指定したファイル名	二重化定義ファイル
抽出側 Datareplicator の初期起動時	%HDEPATH%¥hde_prpfile	抽出定義プリプロセスファイル
	任意の名称_サーバ名	抽出情報キューファイル
	%HDEPATH%¥mststatus	抽出マスタステータスファイル
	%HDEPATH%¥sts_サーバ名	抽出サーバステータスファイル
	%HDEPATH%¥msterrfile1 %HDEPATH%¥msterrfile2	抽出マスタエラー情報ファイル
	%HDEPATH%¥errfile1 ※3 %HDEPATH%¥errfile2	抽出ノードマスタエラー情報ファイル
	%HDEPATH%¥msttrc.trc1 %HDEPATH%¥msttrc.trc2	抽出マスタトレースファイル
	%HDEPATH%¥exttrc.trc1 ※4 %HDEPATH%¥exttrc.trc2	抽出ノードマスタトレースファイル
	%HDEPATH%¥hde_サーバ名	データ連動用連絡ファイル
任意の名称 ※5	コマンドログファイル	

注※1

各ディレクトリは、任意のローカルドライブの下に作成します。

注※2

環境変数 HDEPATH は、Datareplicator の定義を作成する任意のディレクトリを示します。

注※3

抽出システム定義の errfile_unique オペランドに true を指定すると、ファイル名に「_ホスト名」が付きます（ファイル名が「errfile1_ホスト名」と「errfile2_ホスト名」になります）。

注※4

抽出システム定義の errfile_unique オペランドに true を指定すると、ファイル名に「_ホスト名」が付きます（ファイル名が「exttrc_ホスト名.trc1」と「exttrc_ホスト名.trc2」になります）。

注※5

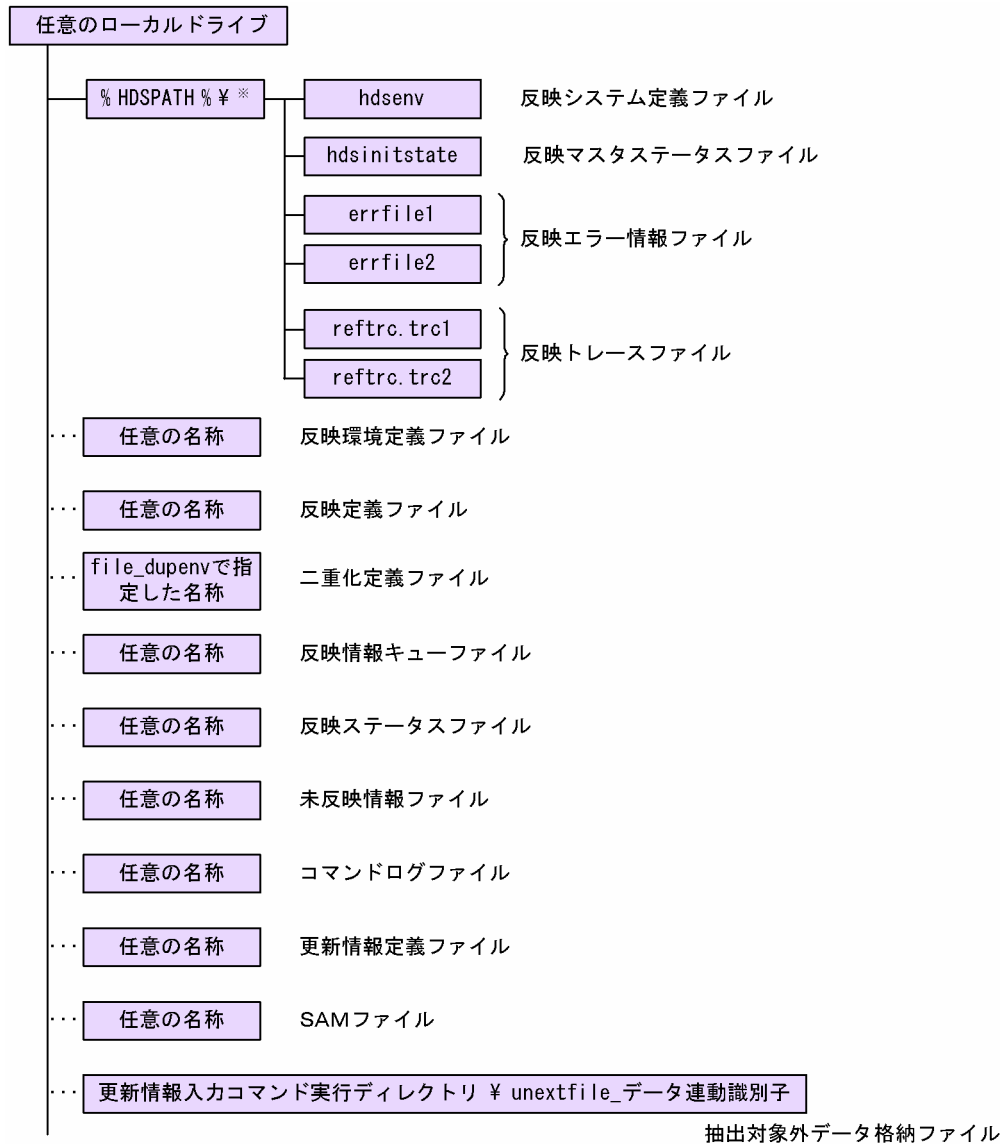
コマンドログファイル名は、環境変数 hde_command_log_file に絶対パス名又は相対パス名で指定します。相対パス名で指定すると、「%HDEPATH%¥相対パス名」が絶対パス名として仮定されます。環境変数 hde_command_log_file の指定を省略すると、コマンドログファイル名として%HDEPATH%¥hdecmdlog が仮定されます。

実際のコマンドログファイル名は、指定した名称の最後に「1」又は「2」が付けられます。実際のコマンドログファイル名の長さが「OS のパスの最大長-1」を超えないように指定してください。

2.7.3 反映側 Datareplicator のディレクトリ構成

反映側 Datareplicator のディレクトリ構成を図 2-6 に、ファイルの内容を表 2-18 に示します。

図 2-6 反映側 Datareplicator のディレクトリ構成 (Windows)



注※

Datareplicator の定義を格納する、任意のディレクトリを示す環境変数です。

表 2-18 反映側 Datareplicator のディレクトリとファイルの内容

作成タイミング	ディレクトリとファイル名※1※2	内容
ユーザが作成	%HDSPATH%\hdsenv	反映システム定義ファイル
	任意の名称	反映環境定義ファイル
	任意の名称	反映定義ファイル

作成タイミング	ディレクトリとファイル名※1※2	内容
	抽出システム定義ファイルの file_dupenv オペランドに指定したファイル名	二重化定義ファイル
反映側 Datareplicator の初期起動時	任意の名称	反映情報キューファイル
	任意の名称	反映ステータスファイル
	%HDSPATH%hdsinitstate	反映マスタステータスファイル
	%HDSPATH%¥errfile1 %HDSPATH%¥errfile2	反映エラー情報ファイル
	%HDSPATH%reftrc.trc1 %HDSPATH%reftrc.trc2	反映トレースファイル
	任意の名称	未反映情報ファイル
	任意の名称※3	コマンドログファイル
ユーザが作成	任意の名称	更新情報定義ファイル※4
ユーザが転送	任意の名称	SAM ファイル※5
更新情報入力コマンド の実行時	更新情報入力コマンドの実行ディレクトリ¥unextfile_データ連動識別子	抽出対象外データ格納ファイル※6

注※1

各ディレクトリは、任意のローカルドライブの下に作成します。

注※2

環境変数 HDSPATH は、Datareplicator の定義を作成する任意のディレクトリを示します。

注※3

コマンドログファイル名は、環境変数 hds_command_log_file に絶対パス名又は相対パス名で指定します。相対パス名で指定すると、「%HDSPATH%¥相対パス名」が絶対パス名として仮定されます。環境変数 hds_command_log_file の指定を省略すると、コマンドログファイル名として%HDSPATH%¥hdscmdlog が仮定されます。

実際のコマンドログファイル名は、指定した名称の最後に「1」又は「2」が付けられます。実際のコマンドログファイル名の長さが「OS のパスの最大長-1」を超えないように指定してください。

注※4

更新情報定義ファイルは、SAM ファイルを使うメインフレーム側 DB のうち、PDMII E2 のデータを抽出する反映側 Datareplicator で作成します。それ以外の場合は、作成する必要はありません。

注※5

メインフレーム側の更新情報抽出 SAM ファイルが、ファイル転送プログラムで転送されてきます。SAM ファイルは、SAM ファイルを使うメインフレーム側 DB (PDMII E2, RDB1 E2) とデータ連動する反映側 Datareplicator の場合に作成します。それ以外の場合は必要ありません。

注※6

抽出対象外データ格納ファイルは、更新情報入力コマンドを実行するたびに作成／再作成されます。ファイル名は、unextfile_に更新情報入力コマンドの実行時に指定したデータ連動識別子が付いた名称になります。

2.8 環境変数の設定 (Windows)

抽出側 Datareplicator と反映側 Datareplicator の環境変数をシステム環境変数に設定する方法と設定する内容について説明します。

環境変数は、次に示す手順で設定してください。

1. [コントロールパネル] にある [システム] アイコンを開きます。
2. [システム] ダイアログボックス中の [環境] タブを選んでから [変数] と [値] を設定して、[設定] ボタンをクリックします。環境変数は、システム環境変数にだけ設定してください。

Datareplicator をインストールしてから環境変数を設定した場合は、いったん Windows を再起動する必要があります。この手間を防ぐため、Datareplicator をインストールする前に環境変数を設定しておくと、再起動が 1 回で済みます。

環境変数を確認するときは、コントロールパネルなどでシステム環境変数を確認してください。

2.8.1 抽出側 Datareplicator の環境変数

抽出側 Datareplicator の環境変数の設定内容を次の表に示します。

表 2-19 抽出側 Datareplicator の環境変数の設定内容

環境変数	設定内容
PATH	次に示すパスを追加します。 <ul style="list-style-type: none">• 抽出側 Datareplicator のインストールディレクトリ¥bin^{*1}• %PDDIR%¥bin^{*2}• %PDDIR%client¥utl^{*2}
TZ	抽出側 Datareplicator のメッセージ出力時のタイムゾーンを指定します。HiRDB のシステム定義に指定した内容と合わせてください。
PDDIR ^{*3*4}	抽出側 HiRDB の運用ディレクトリを指定します。
PDCONFPATH ^{*3*4}	抽出側 HiRDB のシステム定義ファイルを格納するディレクトリを指定します。
PDNAMEPORT ^{*3}	抽出側 HiRDB のポート番号を指定します。
PDHOST ^{*3}	抽出側 HiRDB のホスト名を指定します。
PDUSER	抽出側 HiRDB の接続ユーザを指定します。
HDEPATH	抽出側 Datareplicator の運用ディレクトリを指定します。このディレクトリ下に抽出側 Datareplicator の定義ファイル、ステータスファイル、エラー情報ファイルを作成します。さらに、このディレクトリは抽出側 Datareplicator のコアファイル（プロセス全体のダンプファイル）の出力先になります。 なお、Windows の場合は運用ディレクトリ名に丸括弧を使用できます。

環境変数	設定内容
	<p>注意</p> <ul style="list-style-type: none"> 抽出側 Datareplicator と反映側 Datareplicator が同じマシン上にある場合には、環境変数 HDEPATH と HDSPATH とで同じディレクトリを指定しないでください。 抽出側 HiRDB が複数のサーバマシンから構成されるパラレルサーバの場合、それぞれのサーバマシン上の抽出側 Datareplicator の運用ディレクトリを同一にしてください。
HDE_BIN_COL_MAXLEN	<p>Datareplicator 内部で扱う BLOB 型の列の定義長を、実際の定義長よりも小さい値で指定します (単位: キロバイト)。定義長は 2GB 以上でも実際に扱うデータは小さい BLOB 型の列の場合、表を再定義することなくデータ連動できるようになります。</p> <p>注意</p> <ul style="list-style-type: none"> この環境変数に指定された値は、hdeprep コマンド実行時に有効になります。 指定する BLOB 型の列の定義長は、262,143KB 以内にしてください。 この環境変数に指定された値は、抽出対象となるすべての BLOB 型の列に対して有効となります。 この環境変数に指定した定義長を超える BLOB 型の列が検出された場合の処理については、「6.4.3(4) 環境変数 HDE_BIN_COL_MAXLEN についての注意」を参照してください。

注※1

コマンドライブラリの名称として、インストール時に設定されます。詳細については、「2.6.4 インストール時に登録される内容」を参照してください。

注※2

抽出側 HiRDB を標準セットアップしている場合は、インストール時に設定されます。抽出側 HiRDB を識別子付きセットアップしている場合は、必ず設定してください。

注※3

抽出側 HiRDB の環境変数です。HiRDB の設定内容については、マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

注※4

抽出側 HiRDB がパラレルサーバで、さらにサーバマシンごとに異なる設定をしている場合には、抽出システム定義でサーバマシンごとに個別に定義できます。抽出システム定義での個別の定義を省略すると、そのサーバマシン下の抽出側 Datareplicator では、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定した環境変数の値に従います。

2.8.2 反映側 Datareplicator の環境変数

反映側 Datareplicator の環境変数の設定内容を次の表に示します。

表 2-20 反映側 Datareplicator の環境変数の設定方法

環境変数	設定内容
PATH	反映側 Datareplicator のコマンドライブラリの名称を指定します。*1

環境変数	設定内容
TZ	反映側 Datareplicator のメッセージ出力時のタイムゾーンを指定します。HiRDB のシステム定義に指定した内容と合わせてください。
PDDIR*2	反映側 HiRDB の運用ディレクトリを指定します。
PDNAMEPORT*2	反映側 HiRDB のポート番号を指定します。
PDHOST*2	反映側 HiRDB のホスト名を指定します。
HDSPATH	<p>反映側 Datareplicator の運用ディレクトリを指定します。このディレクトリ下に反映側 Datareplicator の定義ファイル、ステータスファイル、エラー情報ファイルを作成します。さらに、このディレクトリは反映側 Datareplicator のコアファイル（プロセス全体のダンプファイル）の出力先になります。</p> <p>なお、Windows の場合は運用ディレクトリ名に丸括弧を使用できます。</p> <p>注意</p> <p>抽出側 Datareplicator と反映側 Datareplicator が同じマシン上にある場合は、環境変数 HDEPATH と HDSPATH とで同じディレクトリを指定しないでください。</p>
HDS_MST_STDCLOSE	<p>反映側 Datareplicator で、標準入力、標準出力、及び標準エラー出力をクローズするかどうかを指定します。</p> <p>この環境変数を省略又は「TRUE」を指定： 反映側 Datareplicator で、標準入力、標準出力、及び標準エラー出力をクローズします。</p> <p>「FALSE」を指定： 反映側 Datareplicator で、標準入力、標準出力、及び標準エラー出力をクローズしません。</p> <p>「FALSE」を指定した場合、反映情報編集 UOC で標準入力、標準出力、及び標準エラー出力を使用できます。</p> <p>この環境変数を設定しないまま反映情報編集 UOC で標準入力、標準出力、又は標準エラー出力を使用した場合、反映が正常に実行されないおそれがあります。反映情報編集 UOC で標準入力、標準出力、又は標準エラー出力を使用する場合は、必ずこの環境変数に「FALSE」を設定してください。</p>
HDS_RFI_ELANG*3	<p>hdsrefinfmt コマンドを実行する際の抽出側システムの文字コードを指定します。</p> <p>指定されていない場合は、反映ステータスファイルを基に自動判別します。ただし、反映側の初期化直後に定義情報を表示する場合は、自動判別できません。自動判別できない場合は、ja_JP.SJIS を仮定します。</p>
HDS_RFI_PLANG*3	<p>hdsrefinfmt コマンドを実行する際の反映側システムの文字コードを指定します。</p> <p>指定されていない場合は、ja_JP.SJIS を仮定します。</p>
HDSCLTWAITTIME	<p>HiRDB のクライアント環境定義 PDCWAITTIME（クライアントの最大待ち時間）と PDSWAITTIME（サーバの最大待ち時間）*4 の設定値を、反映側 Datareplicator で有効にするかどうかを指定します。</p> <p>この環境変数の指定は任意のため、指定する必要がない場合は省略することをお勧めします。</p> <p>「USER」を指定： HiRDB のクライアント環境定義の値を有効にします。</p> <p>「REPL」を指定： HiRDB のクライアント環境定義の値を無効にして、次の値を設定します。</p> <ul style="list-style-type: none"> • PDCWAITTIME

環境変数	設定内容
	<p>0 (応答があるまで待ち続ける) を設定します。</p> <ul style="list-style-type: none"> • PDSWAITTIME <p>反映システム定義 commit_wait_time オペランドの値^{※5} + 600 (秒) を設定します。 反映トランザクション同期機能を使用している場合は、反映システム定義 syncwait_limit_time オペランドの値 + 600 (秒) を設定します。 なお、設定値が 65,535 を超える場合は 65,535 が設定されます。</p> <p>この環境変数を省略した場合</p> <p>反映情報編集 UOC を実行するときは「USER」の指定値が仮定され、反映 SQL プロセスを実行するときは「REPL」の指定値が仮定されます。</p> <p>「USER」を指定する場合</p> <p>HiRDB のクライアント環境定義 PDCWAITTIME と PDSWAITTIME には、次のオペランドの指定値より大きな値を設定してください。</p> <ul style="list-style-type: none"> • PDCWAITTIME HiRDB の pd_lck_wait_timeout オペランド • PDSWAITTIME 反映システム定義の commit_wait_time オペランド及び syncwait_limit_time オペランド <p>PDCWAITTIME と PDSWAITTIME に設定した値が小さいと、HiRDB の監視時間を超過してしまふことがあります。この場合、反映側 Datareplicator のプロセスは SQL エラーを検知して異常終了します。</p>

注※1

コマンドライブラリの名称は、インストール時に設定されます。詳細については、「[2.6.4 インストール時に登録される内容](#)」を参照してください。

注※2

反映側 HiRDB の環境変数です。環境変数の設定内容については、マニュアル「HiRDB システム導入・設計ガイド」を参照してください。

注※3

HDS_RFI_ELANG, 及び HDS_RFI_PLANG の指定値を次に示します。

環境変数	文字コード			
	JIS8/シフト JIS	EUC	Unicode (UTF-8)	EBCDIC/KEIS EBCDIK/KEIS
HDS_RFI_ELANG	ja_JP.SJIS	ja_JP.UJIS	ja_JP.UTF8	EBCDIK
HDS_RFI_PLANG	ja_JP.SJIS	ja_JP.UJIS	ja_JP.UTF8	—

(凡例)

— : 該当しません。

注※4

HiRDB のクライアント環境定義 PDSWATCHTIME は対象外です。HiRDB のクライアント環境定義に指定した値が常に有効となります。

注※5

反映環境定義の `commit_wait_time` オペランドを指定している場合は、反映環境定義に指定した値が有効になります。

2.9 通信環境の設定 (Windows)

Datareplicator を実行する前に、通信環境を設定する必要があります。ここでは、通信環境の設定内容について説明します。

2.9.1 サービス名の登録

services ファイル (Windows のシステムディレクトリ¥drivers¥etc¥services) に、次に示す形式でサービス名とポート番号を指定します。

サービス名 ポート番号/tcp

(1) 抽出側 Datareplicator の場合

抽出側 Datareplicator で services ファイルに指定する内容を次の表に示します。

表 2-21 抽出側 Datareplicator で services ファイルに指定する内容

サービス名	ポート番号※1	設定するサーバマシン	
		抽出側 HiRDB がシングルサーバの場合	抽出側 HiRDB がパラレルサーバの場合
反映側システムとの通信で使うサービス名を任意の名称で指定します。 送信環境定義の hdeservice オペランドにも、同じ名称を指定してください。	システム内で使える任意の番号を指定します。 反映側システムと同じ番号を指定してください。	シングルサーバを定義したサーバマシン	バックエンドサーバを定義したすべてのサーバマシン※2
抽出マスタプロセスと抽出ノードマスタプロセスとの通信で使うサービス名「hdenmserv」を指定します。 名称は変更できません。 インストール完了後、Windows を終了又は再起動する前に指定してください。	システム内で使える任意の番号を指定します。 ポート番号は、システムマネージャのサーバとバックエンドサーバのサーバとで同じにしてください。	シングルサーバを定義したサーバマシン	<ul style="list-style-type: none">システムマネージャを定義したサーバマシンバックエンドサーバを定義したすべてのサーバマシン※2

注※1

既に services ファイルに登録してあるポート番号や、ほかのソフトウェアで使うポート番号は指定できません。

注※2

抽出側 DB がないバックエンドサーバも含まれます。

(2) 反映側 Datareplicator の場合

反映側 Datareplicator で services ファイルに指定する内容を次の表に示します。

表 2-22 反映側 Datareplicator で services ファイルに指定する内容

サービス名	ポート番号*
抽出側システムとの通信で使うサービス名を任意の名称で指定します。 反映システム定義の hdsservice オペランドにも同じ名称を指定してください。	システム内で使える、任意の番号を指定します。 抽出側システムと同じ番号を指定してください。

注※

既に services ファイルに登録してあるポート番号や、ほかのソフトウェアで使っているポート番号は指定できません。

2.9.2 ホスト名の登録

反映側システムとの通信で使うホスト名を抽出側システムの hosts ファイル（Windows のシステムディレクトリ %drivers%etc%hosts）に登録します。

登録する形式を次に示します。シングルサーバを定義したサーバマシン、又はバックエンドサーバを定義したすべてのサーバマシンで登録してください。

IPアドレス ホスト名

IP アドレスには任意のアドレスを、ホスト名には任意の名称を指定します。ただし、既に hosts ファイルに登録してある IP アドレスやホスト名は指定できません。

2.9.3 Windows Terminal Service の使用

Windows Terminal Service を使用して Datareplicator の運用・操作ができます。これによって、遠隔地のマシンや、コンソールがないマシンに対しても Datareplicator の運用・操作が可能となります。

Windows Terminal Service の詳細については、OS のマニュアルを参照してください。

Windows Terminal Service には、コンソールセッションと仮想セッションがあります。

• コンソールセッション

操作するサーバの入出力デバイス（ディスプレイ、キーボード、マウスなど）をターミナルサーバのクライアントのデバイスに切り替えるセッションです。主な注意事項を次に示します。

- サーバへリモートデスクトップ接続するコマンドに、/console オプションを付けて起動します。
- クライアント及びサーバの中で 1 個のセッションしか作成できません。

(元の画面は操作不可の状態 (ログオフ状態) となります)

- Windows 2000, Windows Vista, 及び Windows Server 2008 ではサポートしていません。
- **仮想セッション**
操作するサーバをバックグラウンドで操作する (クライアント操作用の画面をバックグラウンドに作成する) セッションです。主な注意事項を次に示します。
 - サーバへリモートデスクトップで接続するコマンドをオプションなしで起動します。
 - 複数作成可能です。

(1) Datareplicator のバージョンと Windows Terminal Service の使用可否

Datareplicator のバージョンによって、Windows Terminal Service を使用できない場合があります。Datareplicator のバージョンと Windows Terminal Service の使用可否を次の表に示します。

表 2-23 Datareplicator のバージョンと Windows Terminal Service の使用可否

Datareplicator のバージョン	コンソールセッション	仮想セッション
08-00 以前	△	×
08-01 以降	○	○

(凡例)

- ：使用できます。
- △：Windows Server 2003 の場合だけ使用できます。
- ×：使用できません。

(2) セキュリティ上の注意事項

Windows Terminal Service を使用した場合、サーバのコンソールがログインしていない状態であっても、クライアント側の画面がログインした状態場合があります。このため、サーバ及びクライアントに次の設定又は運用を行うなどの注意が必要となります。

- スクリーンセーバーのパスワードによる保護を付ける
- クライアントからの操作が終了した場合、ログオフする

2.10 Windows Vista, 及び Windows Server 2008 の場合の動作環境

Windows Vista, 及び Windows Server 2008 の場合, 動作環境がほかの Windows とは異なることがあります。この節では, Windows Vista, 及び Windows Server 2008 の場合の Datareplicator の動作環境について説明します。

2.10.1 コマンドの実行

Windows Vista, 及び Windows Server 2008 には, UAC (User Account Control) というセキュリティ機能があります。UAC を利用している場合, 普段は必要最低限の権限で作業し, システム設定などの管理者権限が必要な作業をするときだけ, 権限を昇格させます。UAC の詳細については, OS のマニュアルを参照してください。

UAC を利用している場合, Datareplicator でコマンドを実行するとき, 「管理者として実行」したコマンドプロンプトかどうかによって, 動作が異なります。

「管理者として実行」したコマンドプロンプトの場合

コマンドプロンプト起動時に, 操作の続行を許可するかどうかの確認ダイアログが表示されます。確認ダイアログで操作の続行を許可すると, コマンドプロンプトが起動します。その後は, 従来と同じようにコマンドプロンプト上でコマンドを実行できます。

「管理者として実行」したコマンドプロンプトではない場合

コマンドプロンプト起動時には確認ダイアログは表示されません。代わりに, コマンドプロンプト上でコマンドを実行するたびに, プログラムのアクセスを許可するかどうかの確認ダイアログが表示されます。確認ダイアログでプログラムのアクセスを許可すると, コマンドプロンプト上でコマンドを実行できます※。

注※

一部のコマンドは, 処理結果のコマンドプロンプトが消えてしまうなどして, 正常に実行できません。このため, 通常は「管理者として実行」したコマンドプロンプトでコマンドを実行することをお勧めします。

2.10.2 コマンド名の変更

Windows Vista, 及び Windows Server 2008 の場合, ほかの OS とコマンド名が異なるコマンドがあります。OS によってコマンド名が異なるコマンドを次の表に示します。

表 2-24 OS によってコマンド名が異なるコマンド

Windows Vista 及び Windows Server 2008 以外の場合	Windows Vista 及び Windows Server 2008 の場合
setup_tool1	set_tool1

Windows Vista 及び Windows Server 2008 以外の場合	Windows Vista 及び Windows Server 2008 の場合
setup_tool2	set_tool2
unsetup_tool	unset_tool

2.10.3 JIS 第 3 水準・第 4 水準の文字のサポート

Windows Vista, 及び Windows Server 2008 でサポートされている JIS 第 3 水準・第 4 水準の文字をデータ連動するには, 次を示す条件をすべて満たす必要があります。

表 2-25 JIS 第 3 水準・第 4 水準の文字をデータ連動する条件 (Windows 限定)

環境	条件
OS	Windows Vista, 及び Windows Server 2008
DB	HiRDB バージョン 08-02 以降
Datareplicator	Datareplicator バージョン 08-01 以降
抽出システム定義の dblocale オペランドの指定値	utf-8
反映システム定義の dblocale オペランドの指定値	

注

上記の条件を満たしていない場合は, 未定義の文字として空白に変換されたり, 正しく表示されなかったりします。

2.11 Datareplicator のバージョンアップ

Datareplicator をバージョンアップする方法について説明します。バージョンアップ時は、Datareplicator を初期化する必要があります。

なお、Datareplicator をバージョンダウンする場合の方法については、「[付録 D Datareplicator のバージョンダウン](#)」を参照してください。

2.11.1 バージョンアップ時の注意

HiRDB/パラレルサーバで Datareplicator を使うときは、システムマネージャ及びすべてのバックエンドサーバの Datareplicator でバージョンを合わせてください。

2.11.2 バージョンアップの手順

Datareplicator をバージョンアップする手順について説明します。ここで示す手順は、HiRDB のバージョンアップが済んでいることが前提です。

1. HiRDB と Datareplicator を正常終了します。HiRDB/パラレルサーバのときは、すべてのサーバを正常終了してください。
2. バージョンアップする Datareplicator をインストールします。Datareplicator の各種ファイル（定義ファイル、ステータスファイルなど）の内容を引き継いで使うときは、定義の内容を確認してください。

2.11.3 バージョンアップ時の注意

バージョンアップ後に注意することを次に示します。

- バージョンアップ後に初めて Datareplicator を開始するときに必ず初期開始を実行してください。
- 32 ビット版 Datareplicator から 64 ビット版 Datareplicator にバージョンアップした場合、UOC の修正が必要です。
 - UNIX 版の場合、long 型は 8 バイトの整数として認識されます。そのため、long 型は int 型に修正してください。
 - ポインタは 8 バイトとなります。UOC 内でポインタの長さを特に意識した処理がなければ、修正の必要はありません。
 - long 型を使用した埋込み変数の宣言はできません。埋込み変数の long 型は int 型に修正してください。

- 反映情報編集 UOC を使用している場合には、バージョンアップ後、反映情報編集 UOC を再リンクしてください。リンク手順の詳細は、「[8.1.2 反映情報編集 UOC の作成手順 \(UNIX\)](#)」を参照してください。

3

データ連動機能

この章では、Datareplicator で実現できるデータ連動のパターン、データ連動の抽出処理と反映処理、及び Datareplicator が提供する機能について説明します。

3.1 連動パターン

Datereplicator でデータ連動できるパターンについて説明します。ここで説明する連動パターンは、抽象データ型を使ったデータ連動の場合を含みます。連動パターンの設計については「4.2 連動パターンの設計」を参照してください。

ここでは、リレーショナル型からリレーショナル型へ反映する場合の例で説明しています。

3.1.1 同一形式の表にデータ連動するパターン

同一形式の表にデータ連動するパターンの例について説明します。

(1) すべて同一の表に反映する例

抽出対象表を、形式、表名、列名などがすべて同一の表に反映する例を次の図に示します。

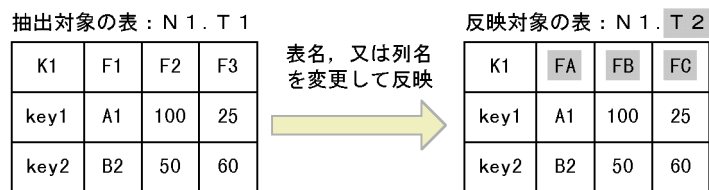
図 3-1 すべて同一の表に反映する例



(2) 表名、又は列名を変更して反映する例

抽出対象表の表名、又は列名を変更して反映する例を次の図に示します。

図 3-2 表名、又は列名を変更して反映する例



(凡例)

■ : 抽出対象の表と反映対象の表とで異なる部分

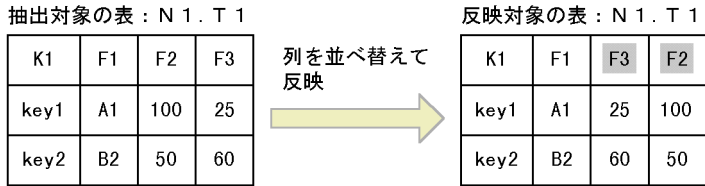
3.1.2 別形式の表にデータ連動するパターン

別形式の表にデータ連動するパターンの例について説明します。

(1) 列を並べ替えて反映する例

抽出対象表の列を並べ替えて反映するパターンを次の図に示します。列を並べ替える場合、抽出側システム、又は反映側システムのどちらかで定義します。

図 3-3 列を並べ替えて反映する例



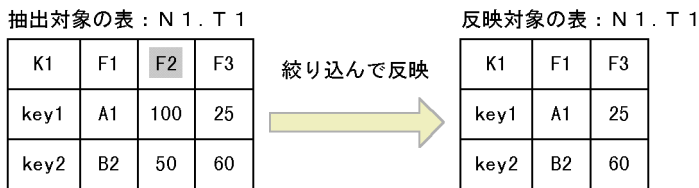
(凡例)

■ : 抽出対象の表と反映対象の表とで異なる部分

(2) 一部の列を反映する例

抽出対象表の一部の列を反映する例を次の図に示します。一部の列を反映する場合、抽出側システム、又は反映側システムのどちらかで定義します。

図 3-4 一部の列を反映する例



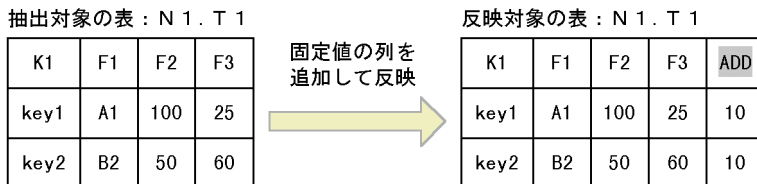
(凡例)

■ : 抽出対象の表と反映対象の表とで異なる部分

(3) 固定値の列を追加して反映する例

固定値の列を追加して反映する例を次の図に示します。固定値の列を追加して反映する場合、反映側システムで定義します。

図 3-5 固定値の列を追加して反映する例



(凡例)

■ : 抽出対象の表と反映対象の表とで異なる部分

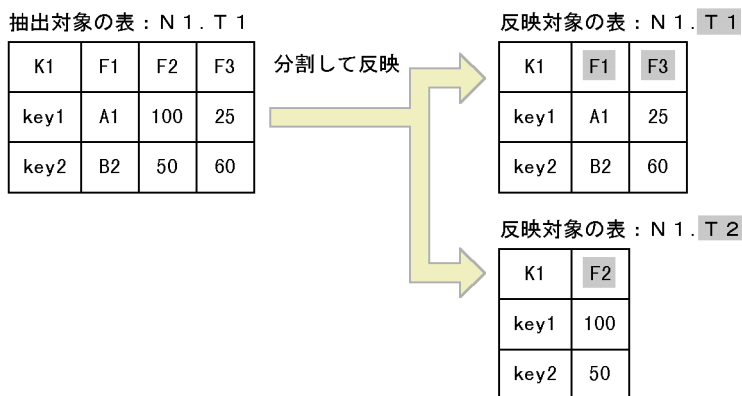
3.1.3 1表からn表にデータ連動するパターン

1表からn表にデータ連動するパターンの例について説明します。

(1) 複数の表に分割して反映する例

抽出対象表を複数の表に分割して反映する例を次の図に示します。複数の表に分割して反映する場合、抽出側システム、又は反映側システムのどちらかで定義します。また、抽出側システムの定義によって、別の反映先に分割して反映することもできます。

図 3-6 複数の表に分割して反映する例



(凡例)

■ : 抽出対象の表と反映対象の表とで異なる部分

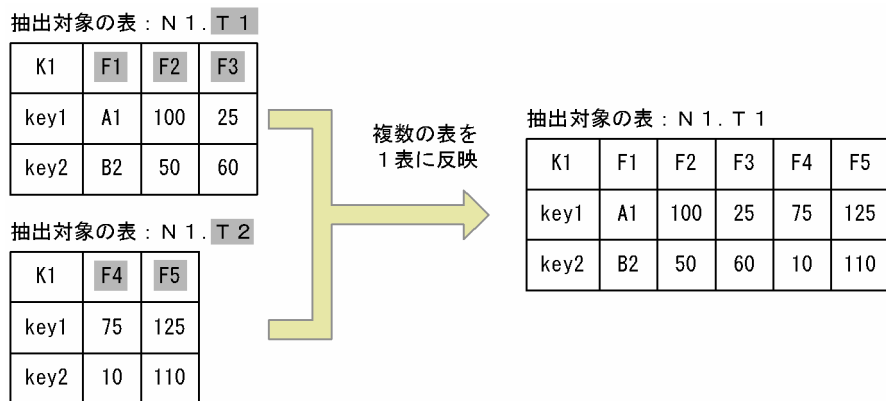
3.1.4 n表から1表にデータ連動するパターン

n表から1表にデータ連動するパターンの例について説明します。

(1) 複数の表を一つの表に反映する例

複数の表を一つの表に反映する例を次の図に示します。複数の表を一つの表に反映する場合、反映側システムで定義します。

図 3-7 複数の表を一つの表に反映する例



(凡例)

■ : 抽出対象の表と反映対象の表とで異なる部分

3.1.5 送信行を選択してデータ連動するパターン

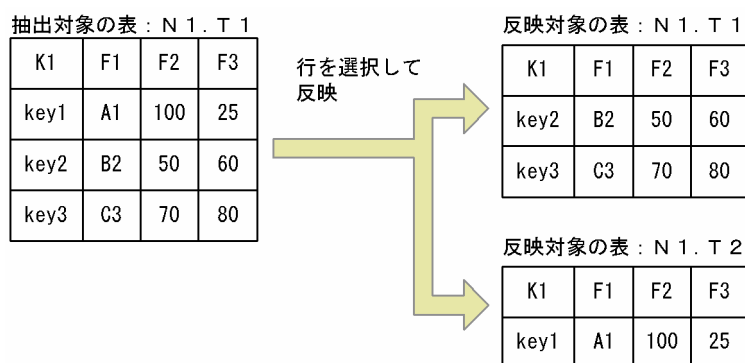
送信行を選択してデータ連動するパターンの例について説明します。

(1) 送信行を選択して反映する例

抽出側システムの更新情報のうち、条件に該当する更新情報だけを選んで反映するパターンです。複数の反映側システムごとに、反映する更新情報の条件を指定できます。反映する更新情報の条件は、抽出定義の送信文 (send 文) に指定します。

送信行を選択して反映する例を次の図に示します。

図 3-8 送信行を選択して反映する例



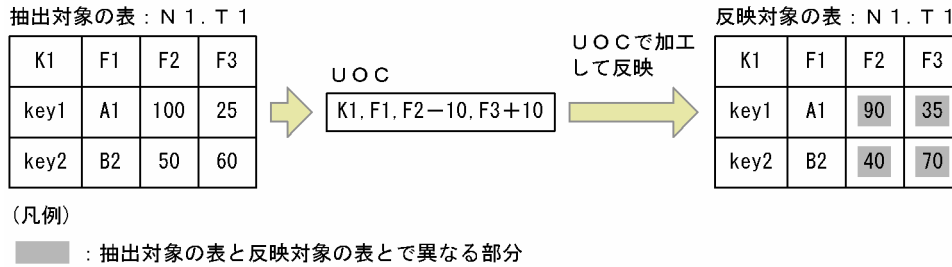
3.1.6 ユーザOWNコーディングを使ってデータ連動するパターン

UOC でデータを編集してから反映するデータ連動のパターン例について説明します。UOC については、[\[8. ユーザOWNコーディング\]](#) を参照してください。

(1) UOC で更新情報を任意に加工して反映する例

反映情報編集 UOC で更新情報を任意に加工して反映する例を次の図に示します。

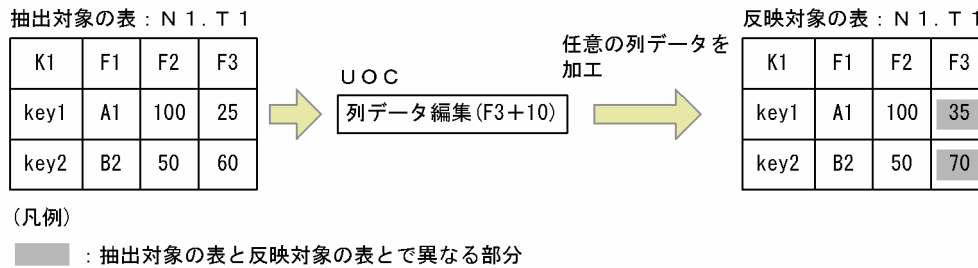
図 3-9 UOC で更新情報を加工して反映する例



(2) UOC で列データを編集して反映する例

列データ編集 UOC を使って、任意の列の更新情報を加工して反映する例を次の図に示します。

図 3-10 UOC で列を加工して反映する例



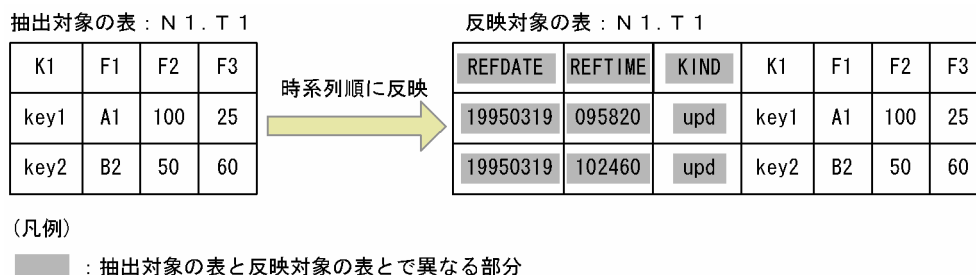
3.1.7 更新情報の履歴を時系列順に取得するパターン

更新情報の履歴を時系列順に取得するパターンの例について説明します。

(1) 更新情報のほかに反映日付、反映時間などを時系列順に反映する例

更新情報のほかに反映日付、反映時間などを時系列順に反映する例を次の図に示します。時系列情報を取得する場合、反映側システムで定義します。

図 3-11 更新情報のほかに反映日付、反映時間などを時系列順に取得する例



3.2 抽出側 Datareplicator の抽出処理

抽出側 Datareplicator の抽出処理について説明します。

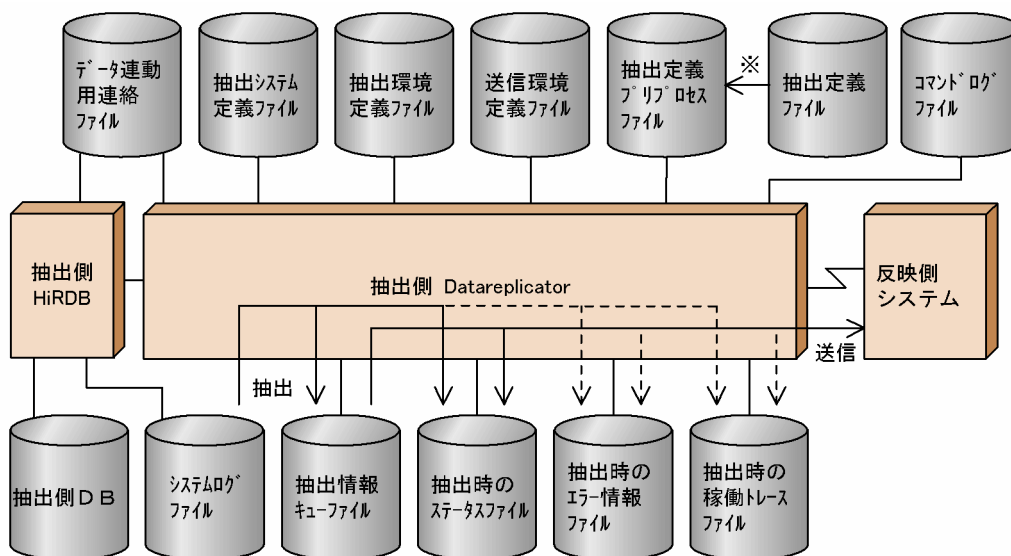
抽出側 DB がメインフレーム側 DB のときの抽出処理については、次に示すマニュアルを参照してください。

- XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/S の場合
マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。
- SAM ファイルを使うメインフレーム側 DB の場合
VOS3 PDMII E2, VOS1 PDMII E2, VOS1 RDB1 E2 のうち、該当する製品のマニュアルを参照してください。

3.2.1 抽出処理の概要

抽出処理は「抽出」と「送信」に分かれます。抽出処理の概要を次の図に示します。

図 3-12 抽出処理の概要



(凡例)

→ : データを格納

--> : エラーのときにデータを格納

注※ hdeprepコマンド

(1) 抽出

抽出側 HiRDB のシステムログファイルから更新情報を抽出して、抽出情報キューファイルに格納します。抽出側 HiRDB がパラレルサーバの場合には、抽出対象になるバックエンドサーバごとに HiRDB のシステムログファイルが存在するため、バックエンドサーバごとに抽出処理を実行します。このとき、抽出の状態をデータ連動用連絡ファイルと抽出時のステータスファイルに格納します。

抽出中にエラーになったときは、エラーの内容が抽出時のエラー情報ファイルに出力されます。

(2) 送信

抽出情報キューファイルに入っている更新情報を読み出し、指定された送信先の反映側システムに送信します。抽出側 HiRDB がパラレルサーバの場合には、抽出対象になるバックエンドサーバごとに送信処理を実行します。このとき、送信の状態を抽出時のステータスファイルに格納します。

送信中にエラーになったときは、エラーの内容が抽出時のエラー情報ファイルに出力されます。

3.2.2 抽出処理で使うファイルとプロセス

抽出処理で使うファイルとプロセスについて説明します。

(1) 抽出処理で使うファイル

抽出処理で使うファイルを次に示します。

(a) HiRDB システムログファイル

HiRDB でのデータベースの更新情報を格納するファイルです。抽出側 Datareplicator では、このファイルから更新情報を抽出します。

(b) 抽出情報キューファイル

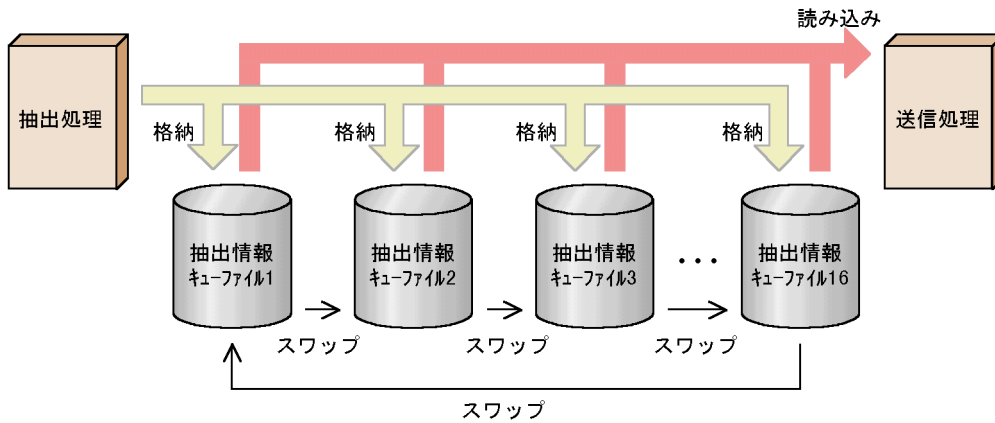
抽出処理によって HiRDB システムログファイルから抽出した更新情報が格納されるファイルです。

抽出処理によって抽出された更新情報は順次、抽出情報キューファイルに格納されます。一つの抽出情報キューファイルが満杯になると、格納対象が次のファイルに切り替えられます。これを、スワップといいます。抽出側 Datareplicator では、このスワップによって大量の更新情報を格納できます。スワップの順番は、抽出環境定義の qufile001~016 オペランドの指定順になります。

また、すべての抽出情報キューファイルが満杯になると、先頭のファイルにスワップされます。スワップ先のファイルの更新情報の送信が終了していない場合はスワップできません。この場合、キューファイルが満杯であることを示すメッセージが出力され、スワップ先のファイルの更新情報の送信が終了するまで、システムログファイルからの更新情報の抽出を停止します。

抽出情報キューファイルへのデータの格納方法を次の図に示します。

図 3-13 抽出情報キューファイルへのデータの格納方法



(c) データ連動用連絡ファイル

HiRDB での更新情報のシステムログファイルへの格納状態や、抽出側 Datareplicator での更新情報のシステムログファイルからの読み込み状態など、抽出処理を実行するために必要な HiRDB との連絡情報を格納、又は読み込むためのファイルです。

(d) 抽出時のステータスファイル

障害発生時に回復に必要な抽出/送信の状態などが格納されるファイルです。抽出時のステータスファイルには、抽出マスタステータスファイルと、抽出サーバステータスファイルがあります。

(e) 抽出時のエラー情報ファイル

抽出処理や送信処理がエラーになったときに、エラーの詳細情報が出力されるファイルです。抽出時のエラー情報ファイルには、抽出マスタエラー情報ファイルと抽出ノードマスタエラー情報ファイルがあります。

なお、エラー情報ファイルに出力される情報と同様の情報を、syslog ファイルにも出力できます。syslog ファイルにも出力するかどうかは、抽出システム定義の syslogout オペランドで指定します。

(f) 抽出時の稼働トレースファイル

Datareplicator の稼働状況を取得するファイルです。Datareplicator の動作や性能についての情報を取得しています。稼働トレースファイルには、抽出マスタトレースファイルと抽出ノードマスタトレースファイルがあります。

(g) 抽出システム定義ファイル

抽出側 Datareplicator 識別子や、送信先識別子など、抽出側 Datareplicator 全体の稼働環境の情報を定義するファイルです。

(h) 抽出環境定義ファイル

抽出情報キューファイル名や抽出情報キューファイルのサイズなど、抽出処理の稼働環境の情報を定義するファイルです。

(i) 送信環境定義ファイル

通信のためのサービス名やホスト名など、送信処理の稼働環境の情報を定義するファイルです。

(j) 抽出定義ファイル

抽出対象表、列と更新情報との対応や、更新情報の送信先など、抽出、送信処理の詳細な情報を定義するファイルです。

(k) 抽出定義プリプロセスファイル

hdeprep コマンドを実行して、抽出定義ファイルを内部形式に変換したファイルです。抽出定義プリプロセスファイルへの変換は、抽出側 Datareplicator を起動する前に実行する必要があります。

抽出定義プリプロセスファイルの有効性チェック

抽出側 Datareplicator を起動するとき、抽出定義プリプロセスファイルが有効かどうか自動的にチェックします。次の場合は抽出側 Datareplicator の起動処理が中断されます。

- 抽出定義プリプロセスファイルの作成日付が抽出マスタステータスファイルの作成日時よりも古い場合 (KFRB00713-E メッセージが出力される)
- hdeprep コマンドを実行した後に、抽出対象表の表定義を変更している場合 (KFRB00866-E メッセージが出力される)

ただし、環境変数の PDUSER を省略したり、パスワードの指定を省略したりして HiRDB に接続できない場合は、KFRB00868-W メッセージが出力され、抽出定義プリプロセスファイルの有効性をチェックしないまま抽出側 Datareplicator が起動されます。

(l) コマンドログファイル

Datareplicator のコマンドを実行した日付と時間の履歴を格納するファイルです。

(2) 抽出処理のプロセス構成

抽出側 HiRDB がシングルサーバの場合の抽出処理のプロセス構成を [図 3-14](#) に、抽出側 HiRDB がパラレルサーバの場合の抽出処理のプロセス構成を [図 3-15](#) に示します。

図 3-14 抽出処理のプロセス構成（抽出側 HiRDB がシングルサーバの場合）

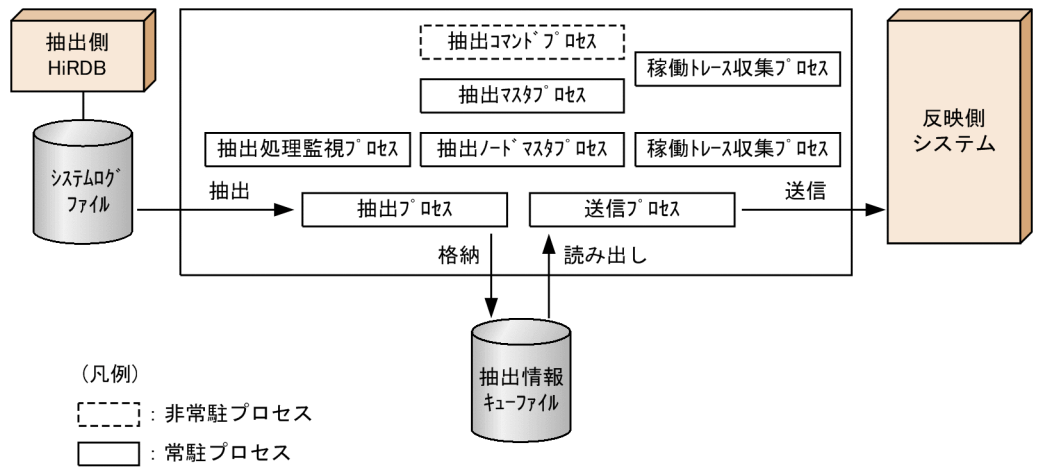
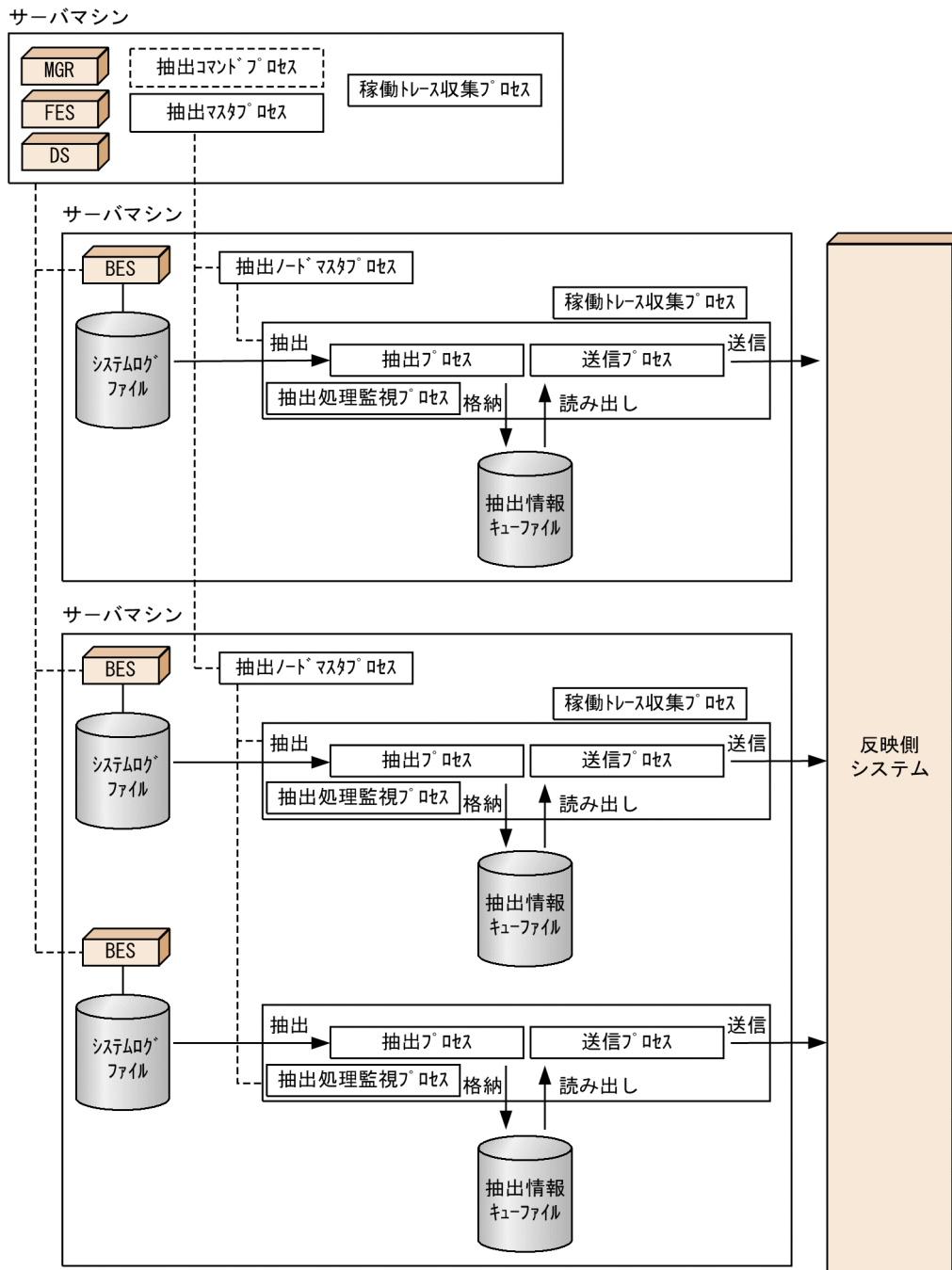


図 3-15 抽出処理のプロセス構成 (抽出側 HiRDB がパラレルサーバの場合)



(凡例)

- | | |
|-----------------------|------------------------|
| MGR : HiRDBのシステムマネージャ | FES : HiRDBのフロントエンドサーバ |
| DS : HiRDBのディクショナリサーバ | BES : HiRDBのバックエンドサーバ |
| □ : 常駐プロセス | □ : 非常駐プロセス |
| ----- : 制御 | |

(a) 抽出コマンドプロセス

抽出側 Datareplicator のコマンドを処理し、抽出マスタプロセスに指示を出すプロセスです。抽出側 HiRDB がパラレルサーバの場合には、システムマネージャ下で一つ稼働します。

(b) 抽出マスタプロセス

抽出ノードマスタプロセスを管理するプロセスです。抽出側 HiRDB がパラレルサーバの場合には、システムマネージャ下で一つ稼働します。

(c) 抽出ノードマスタプロセス

抽出プロセスと送信プロセスを管理するプロセスです。抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバのある各サーバマシン下で一つ稼働します。抽出側 Datareplicator では、このサーバマシンのことをノードと呼びます。

(d) 抽出プロセス

システムログファイルから更新情報を抽出し、抽出情報キューファイルに格納するプロセスです。抽出側 HiRDB がパラレルサーバの場合には、抽出対象の各バックエンドサーバ下で一つ稼働します。

(e) 送信プロセス

抽出情報キューファイルに格納されている更新情報を読み出し、反映側システムに送信するプロセスです。送信プロセスは、送信先の数分稼働します。抽出側 HiRDB がパラレルサーバの場合には、抽出対象の各バックエンドサーバ下で、送信先の数だけ稼働します。ただし、抽出システム定義の sendcontrol オペランドに sendmst を指定したときは、抽出システム定義の sendprocnum オペランドに設定した値以内の送信プロセスが稼働します。

以降、抽出システム定義の sendcontrol オペランドに sendmst を指定したときを **sendmst 方式**、nodemst を指定したときを **nodemst 方式**と呼びます。

(f) 送信マスタプロセス

送信先数が多いときに、起動する送信プロセス数を制御／抑制するため、送信プロセスの起動、停止、スケジュールをするプロセスです。送信マスタプロセスは、送信プロセス数を制御する定義をしたときに生成されます。送信プロセス数の制御については、「[3.2.5 送信プロセス数の制御](#)」を参照してください。

(g) 稼働トレース収集プロセス

稼働トレース情報を取得するためのプロセスです。

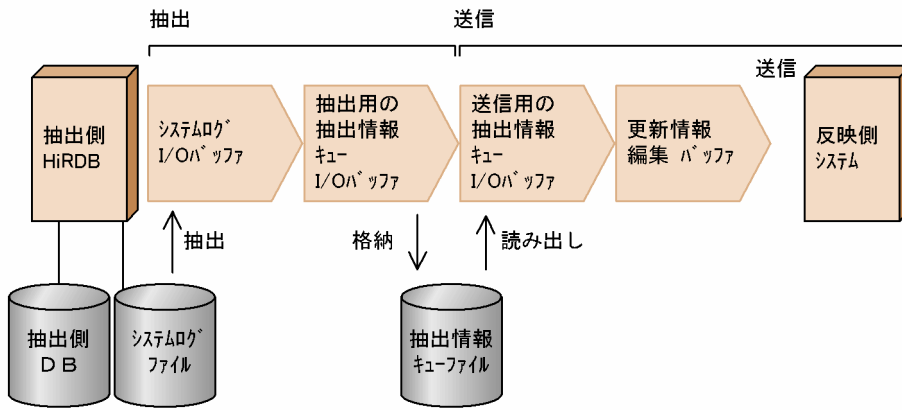
(h) 抽出処理監視プロセス

抽出処理が停止していないかどうかを監視するためのプロセスです。

3.2.3 抽出処理の単位

抽出処理の単位を次の図に示します。

図 3-16 抽出処理の単位



(1) 抽出

システムログファイルに格納されている更新情報を読み込み、抽出定義に従って抽出情報キューファイルに格納します。このとき、システムログ I/O バッファ、抽出用の抽出情報キュー I/O バッファを使います。抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバ単位に抽出を実行します。

抽出で使う各バッファについては、「4.6.3 抽出方法の設計」を参照してください。

(2) 送信

送信先ごとに抽出情報キューファイルに格納されている更新情報を読み出し、トランザクションが完了している更新情報だけを編集し、反映側システムに送信します。このとき、送信用の抽出情報キュー I/O バッファ、更新情報編集バッファを使います。

送信で使う各バッファについては、「4.6.4 送信方法の設計」を参照してください。

3.2.4 抽出側 Datareplicator の情報の取得

コマンドやシステム定義の指定で、次に示す抽出側 Datareplicator の状態を取得できます。

(1) ステータス情報の取得

ステータス情報取得コマンド (hdestate コマンド) を実行すると、抽出側 Datareplicator の状態を確認するためのステータス情報を標準出力に取得できます。ステータス情報は、抽出側 Datareplicator が稼働中のときだけ取得できます。

ステータス情報として、抽出側 Datareplicator 識別子、ノードの情報、抽出情報キューファイルの使用状況などを取得できます。取得例については、「7. コマンドの文法」の hdestate コマンド (抽出側 Datareplicator のステータス情報の取得) の文法を参照してください。

(2) 稼働トレースの取得

Datareplicator の稼働状況を取得して性能を把握するために、稼働トレースを取得できます。稼働トレースを取得すると、次に示す運用ができます。

- 抽出処理、送信処理が遅く感じるときに、システムの設定を変更する目安にする（チューニングとして、抽出バッファのサイズを改善するか、送信バッファ/送信間隔を改善するか、又は通信回線容量を大きくするかを決める）。
- 送信データ UOC の性能が出ないときに、Datareplicator 側での問題か UOC 側での問題かを切り分ける

抽出側 Datareplicator で稼働トレースを取得するときは、抽出システム定義の `int_trc_lvl` オペランドと `int_trc_filesz` オペランドを指定します。稼働トレース収集方式は、抽出システム定義の `int_trc_unit` オペランドで指定します。さらに、バックエンドサーバごとの稼働トレースを取得する場合は抽出環境定義の `int_trc_getv` オペランドを、相手システム識別子ごとの稼働トレースを取得する場合は送信環境定義の `int_trc_getv` オペランドを指定します。

取得した稼働トレースを出力するときは、`hdstrcredit` コマンドを使います。

稼働トレースの定義については「[5.2 抽出システム定義](#)」, 「[5.3 抽出環境定義](#)」, 及び「[5.4 送信環境定義](#)」を、`hdstrcredit` コマンドについては「[7. コマンドの文法](#)」の `hdstrcredit` コマンドの文法を参照してください。

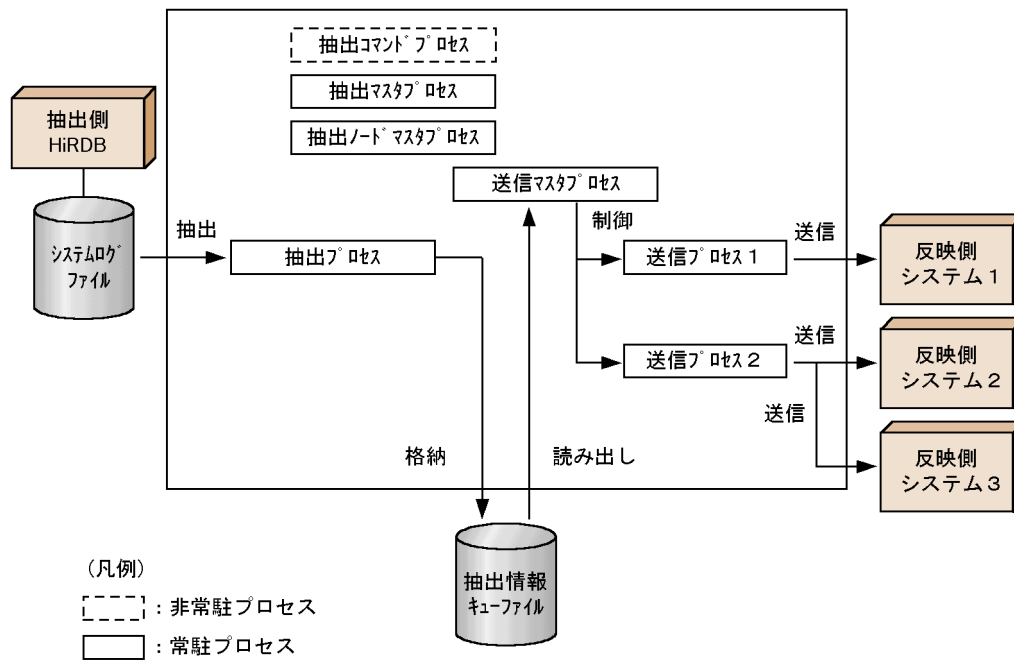
3.2.5 送信プロセス数の制御

Datareplicator では、送信プロセスを送信先単位に起動しています。そのため、送信先が多くなると送信プロセスが増えて、抽出側システムのメモリが不足したりキューファイルの入出力が遅くなったりします。このような場合に備えて、起動する送信プロセスを抑制して、限られた送信プロセス数内ですべての送信処理を実行するように指定できます。

送信プロセス数を制御するときは、抽出システム定義の `sendcontrol` オペランドに `sendmst` を指定します。送信プロセス数を制御する指定をすると、ノードマスタープロセスと送信プロセスの間に**送信マスタープロセス**が生成されます。この送信マスタープロセスが送信プロセスの起動、停止、及び送信処理のスケジュールを制御します。

送信プロセスの制御の概要を次の図に示します。次の図では、2 個の送信プロセスで 3 件の送信先への送信処理を実行する例で示します。

図 3-17 送信プロセスの制御の概要



(1) 送信マスタープロセスの動作

送信マスタープロセスは、送信間隔に到達するまでの間に抽出情報キューファイルから更新情報を読み込んで、有効なトランザクションの管理情報を各送信先単位に生成します。送信間隔に到達したときに、有効なトランザクション管理情報が生成された場合、送信マスタープロセスは送信プロセスに送信要求を発行します。このとき、最大起動プロセス数の指定に従って、送信マスタープロセスは稼働中の送信プロセスをスケジューリングします。

(2) 送信プロセス数を制御する場合の送信プロセスの動作

送信プロセス数の制御を指定したときは、送信プロセスの処理は次に示す点で異なります。

- 抽出情報キューファイルから更新情報を読み込むタイミングは、送信マスタープロセスからトランザクション管理情報を受け取ったときです。
- 更新情報を送信した後で、送信が完了したことを送信マスタープロセスに通知して、その後、送信マスタープロセスの送信要求待ちのまま、プロセスは起動した状態になります。

3.2.6 ユーザOWNコーディングを使った更新情報のチェック

データ連動の業務を補助するため、抽出側 Datareplicator ではユーザOWNコーディング (UOC) を作成できます。抽出側 Datareplicator の業務で作成できる UOC を次に示します。

- 送信データ UOC
更新情報を送信する前に、送信対象にするかどうかをチェックできます。

UOC については、「[8. ユーザOWNコーディング](#)」を参照してください。

3.2.7 メッセージ出力の抑止

Datareplicator 稼働時には、syslog ファイル (Windows の場合はイベントログ)、及びエラー情報ファイルに稼働状況を伝えるためのメッセージが出力されます。

これらのメッセージは稼働状況を知るための重要な情報ですが、稼働環境によっては大量にメッセージが出力されてリソースを多く消費する場合があります。

このような場合に備えて、ユーザが情報として不要と判断したメッセージの出力を抑止するように指定できます。

メッセージの出力を抑止するためのオペランドを次に示します。

オペランド	抑止対象出力先
syslogout	syslog ファイル (又はイベントログ)
syslog_message_suppress	
node_syslogout	
info_message_out*	<ul style="list-style-type: none">• syslog ファイル (又はイベントログ)• エラー情報ファイル

注

各オペランドの詳細については、「[5.2 抽出システム定義](#)」を参照してください。

注※

info_message_out オペランドに suppress を指定して出力を抑止したメッセージのうち、抑止対象から除外してエラー情報ファイルに出力したいメッセージ番号がある場合は、except_suppress オペランドで指定できます。

syslog ファイル (又はイベントログ) へのメッセージの出力を抑止するためのオペランドを、混在して指定した場合のメッセージの出力結果を次に示します。

オペランド			メッセージの出力先	
info_message_out	syslogout 又は node_syslogout	syslog_message_suppress	syslog ファイル (又はイベントログ)	エラー情報ファイル
suppress	指定不可	指定不可	×	×
nosuppress	true	指定あり	×	○
		指定なし	○	○

オペランド			メッセージの出力先	
info_message_out	syslogout 又は node_syslogout	syslog_message_suppress	syslog ファイル (又はイベントログ)	エラー情報ファイル
	false	指定不可	×	○

(凡例)

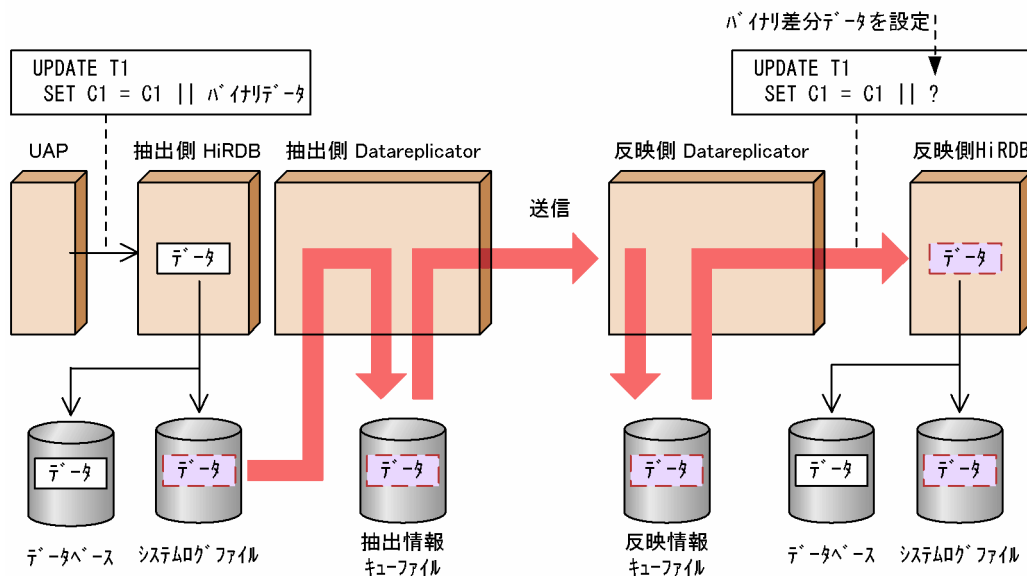
- ：出力します。
- ×

3.2.8 連結演算による更新データの連動

連結演算を使った更新情報も，Datareplicator でデータ連動できます。連結演算の更新では，更新ログが連結演算で追加されたデータ部分だけになります。そのため，従来の BLOB 列や BINARY 列のデータ連動のように列全体を抽出して反映するのではなく，差分データだけを抽出して反映します。

連結演算の更新データのデータ連動を次の図に示します。

図 3-18 連結演算の更新データのデータ連動



(凡例)

- データ : 連結演算データ (列データ全体)
- データ : 連結演算データ (追加した差分データだけ)

注意事項

連結演算とは，Datareplicator Version 6 で BLOB 列の分割取得と呼ばれていたものです。

3.2.9 インナレプリカ機能を使った RD エリアのデータ連動

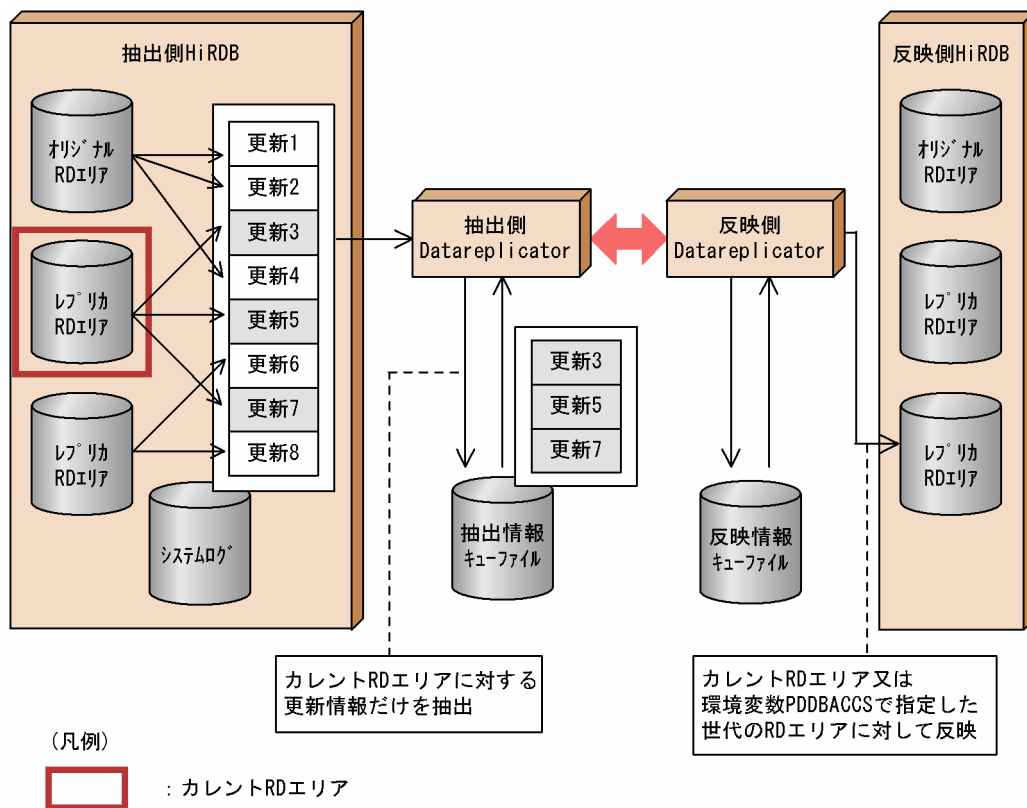
HiRDB のインナレプリカ機能を使った RD エリアも、Datareplicator でデータ連動できます。

インナレプリカ機能では、オリジナル RD エリアとレプリカ RD エリアのどちらの更新でもシステムログファイルに更新ログが出力されます。この更新ログから、抽出対象表 ID に該当する更新ログを Datareplicator が抽出します。

反映側 DB がインナレプリカ機能を使っているときは、データ連動対象のカレント RD エリア（オリジナル RD エリア又はレプリカ RD エリア）又は特定世代のデータベースへ反映できます。ただし、表ごとに反映先を切り替えることはできません。特定世代のデータベースへ反映する場合は、HiRDB の環境変数 Pddbaccs で指定してください。

インナレプリカ機能を使った RD エリアのデータ連動を次の図に示します。

図 3-19 インナレプリカ機能を使った RD エリアのデータ連動



(1) インナレプリカのデータ連動の定義

更新ログの抽出方法は、「5.3 抽出環境定義」の extract_level オペランドに指定します。extract_level オペランドの指定値と抽出する更新ログの関係は次のとおりです。

- current_gen : 抽出対象の HiRDB がインナレプリカ機能を使っているときに、カレント RD エリアに対する更新だけを抽出の対象にします。

- all_gen：抽出対象のHiRDBがインナレプリカ機能を使っているときに、すべてのレプリカRDエリア（オリジナルRDエリアを含む）に対する更新を抽出対象にします。

インナレプリカの運用に従って、次に示す指針で extract_level オペランドを指定してください。

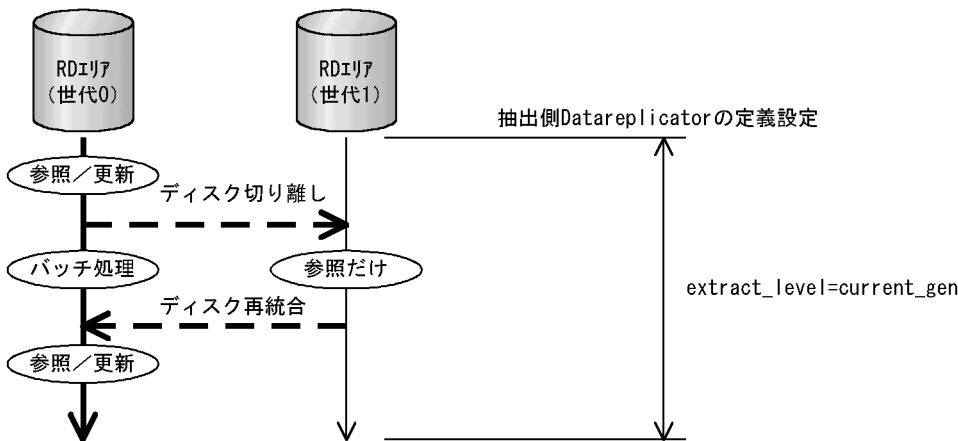
インナレプリカの運用	extract_level オペランドの指定値
常にカレント RD エリアがマスタ（抽出対象）になる運用	extract_level=current_gen を指定
カレント RD エリア以外がマスタ（抽出対象）になる状態がある運用	ディスクの切り離し時に extract_level=all_gen を指定*

注※

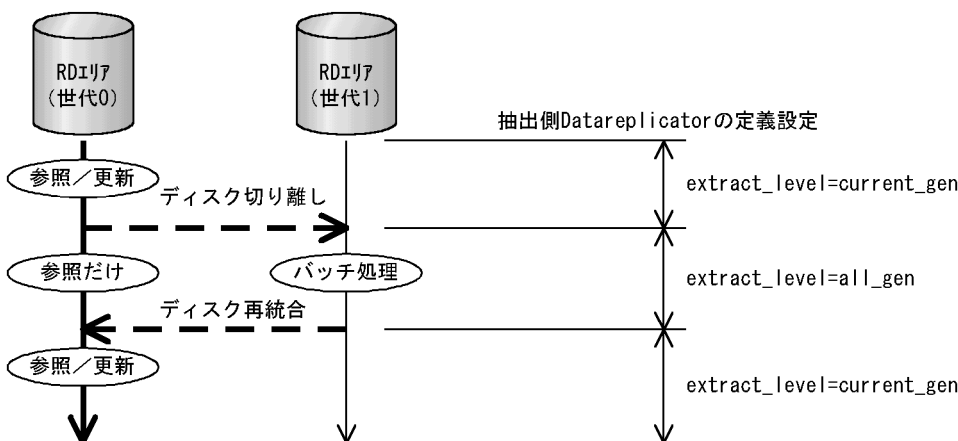
extract_level=all_gen を指定する場合には、制限事項があります。詳細は「(2) 制限事項」を参照してください。

extract_level オペランドを指定する指針を次に示します。

[例] 常にカレントRDエリアがマスタ（抽出対象）となる運用例



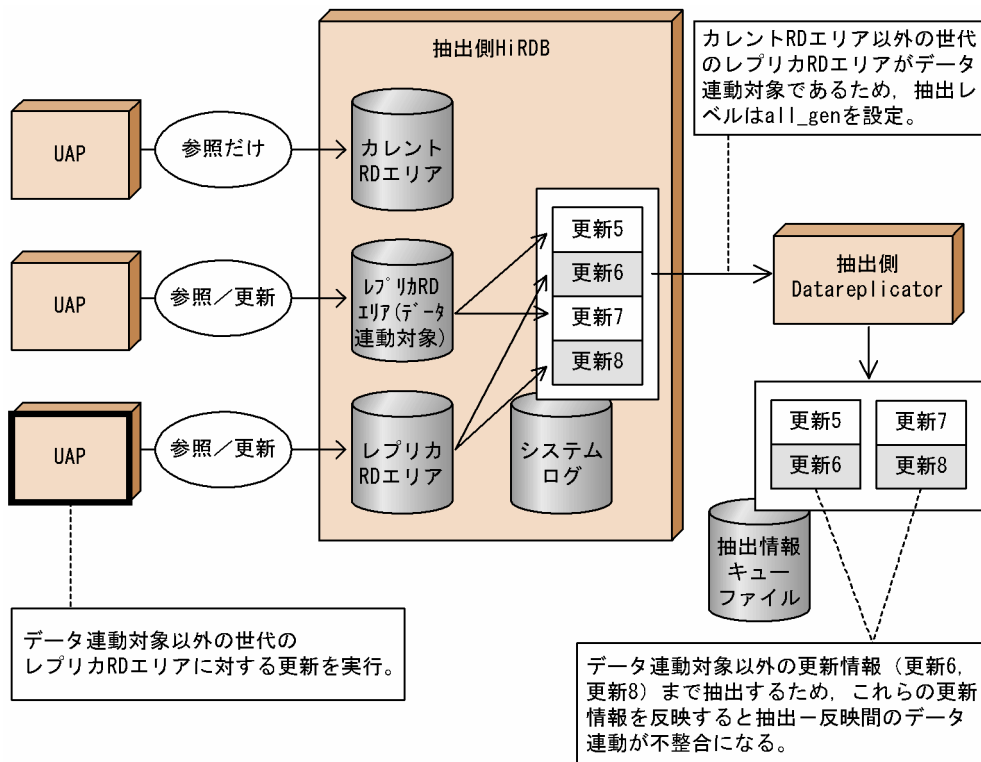
[例] カレントRDエリア以外がマスタ（抽出対象）となる状態がある運用例



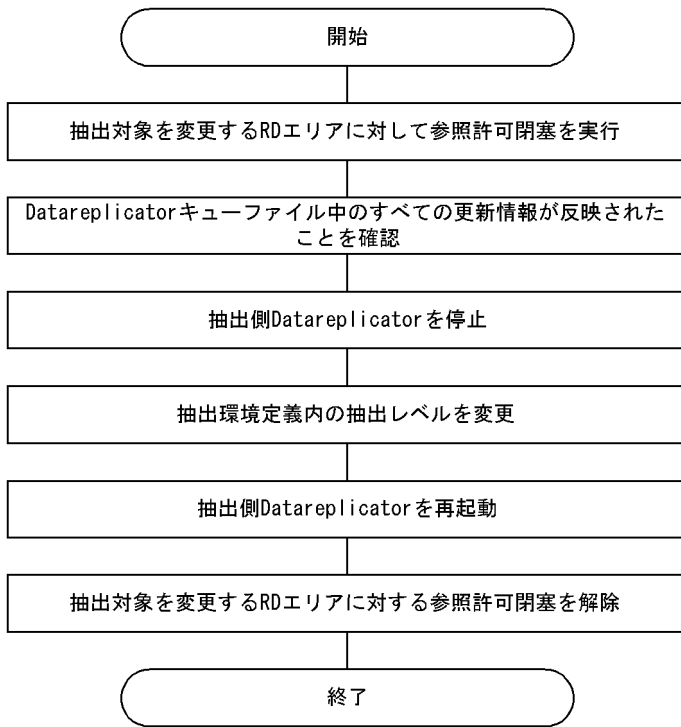
(2) 制限事項

- イベント制御用テーブルに対する更新（イベント発行）については、抽出側 Datareplicator の定義に関係なく、すべての世代に対する更新を抽出します。
- PURGE TABLE については、extract_level=current_gen の場合、操作対象の RD エリアがすべてカレント RD エリアである PURGE TABLE だけを抽出します。
- データ連動対象の RD エリアは、できるだけカレント RD エリアとして、抽出レベルには current_gen を指定してください。

データ連動対象のレプリカ RD エリアをカレントでない RD エリアにするときは、抽出レベルには all_gen を指定してください。このとき、データ連動対象かどうかに関係なく、すべてのレプリカ RD エリアに対する更新が抽出対象になるため、データ連動対象外のレプリカ RD エリアを更新すると、データ連動に不整合が起きます。そのため、抽出レベルを all_gen に設定したときは、データ連動対象のレプリカ RD エリアだけを更新してください。抽出レベルを all_gen に設定したときに、データ連動に不整合が起こる例を次に示します。



- 抽出対象とするレプリカ RD エリアを変更するときは、変更先のレプリカ RD エリアが反映側 DB と同期が取れていなければなりません。抽出対象とするレプリカ RD エリアの変更手順を次に示します。



3.3 反映側 Datareplicator の反映処理

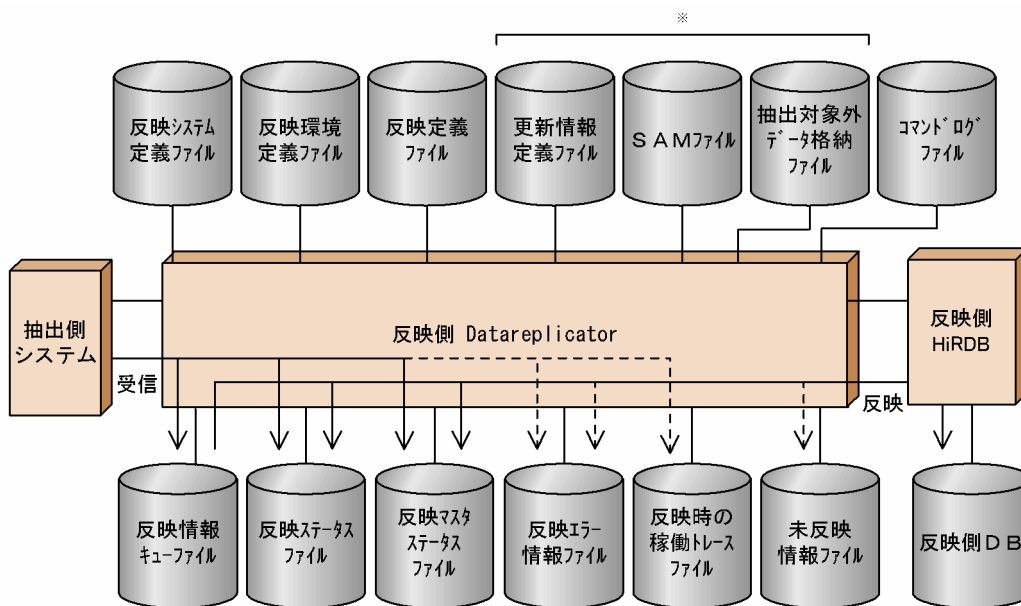
反映側 Datareplicator の反映処理について説明します。

反映側 DB がメインフレーム側 DB (XDM/RD E2) のときの反映処理については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

3.3.1 反映処理の概要

反映処理は「受信」と「反映」に分かれます。反映処理の概要を次の図に示します。

図 3-20 反映処理の概要



(凡例)

→ : データを格納

--> : エラーのときにデータを格納

注※ SAMファイルを使うメインフレーム側DB (PDM II E2, RDB1 E2) とデータ連動するときに必要なファイルです。ただし、更新情報定義ファイルは、RDB1 E2の場合は必要ありません。

(1) 受信

抽出側システムから送信された抽出定義と更新情報を受信し、反映情報キューファイルに格納します。このとき、受信の状態を反映ステータスファイルに格納します。

受信中にエラーになったときは、エラーの内容がエラー情報ファイルに出力されます。

(2) 反映

反映情報キューファイルに格納されている更新情報を読み出し、マッピングキーで対応する行にSQL文を発行して、反映側DBを更新します。このとき、反映の状態を反映ステータスファイルに格納します。

反映中にエラーになったときは、エラーの内容がエラー情報ファイルに出力されます。

発行した SQL 文が HiRDB でエラーになって反映できなかった場合には、その SQL 文を未反映情報ファイルに出力します。

3.3.2 反映処理で使うファイルとプロセス

反映処理で使うファイルとプロセスについて説明します。

(1) 反映処理で使うファイル

反映処理で使うファイルを次に示します。

(a) 反映情報キューファイル

抽出側システムから送信される抽出定義と更新情報を格納するためのファイルです。

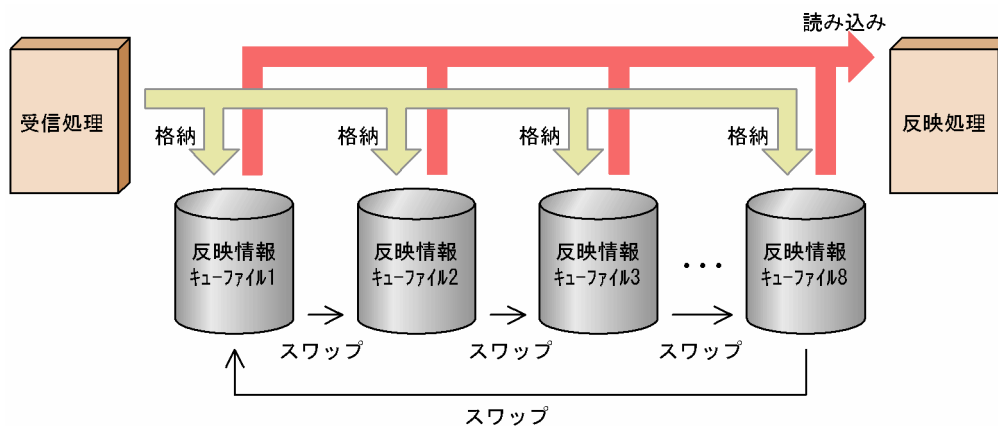
受信した更新情報は順次、反映情報キューファイルに格納されます。一つの反映情報キューファイルが満杯になると、格納対象が次のファイルに切り替えられます。これをスワップといいます。反映側 Datareplicator では、このスワップによって大量の更新情報を格納でき、大量の更新情報を一括して反映することもできます。

すべての反映情報キューファイルが満杯になると、先頭のファイルにスワップされます。スワップ先のファイルの更新情報の反映処理が終了 (COMMIT) していない場合はスワップできません。この場合、抽出側システムとの間の通信が切断され、次の送信間隔まで更新情報の格納が停止します。次の送信間隔時にスワップ先のファイルの更新情報が、すべて反映処理に読み込まれていれば、更新情報の格納が再開されます。

送信間隔については、「4.6.4 送信方法の設計」を参照してください。XDM/DS で指定する送信間隔については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

反映情報キューファイルへのデータの格納方法を次の図に示します。

図 3-21 反映情報キューファイルへのデータの格納方法



(b) 反映時のステータスファイル

障害発生時に回復に必要な受信／反映の状態や、抽出側システムの抽出定義の状態などが格納されるファイルです。

(c) 反映マスタステータスファイル

初期開始時の実行結果が格納されるファイルです。

(d) 反映時のエラー情報ファイル

受信処理や反映処理がエラーになったときに、エラーの詳細情報が出力されるファイルです。エラー情報ファイルに出力される情報と同様の情報を、syslog ファイルにも出力できます。syslog ファイルにも出力するかどうかは、反映システム定義の syslogout オペランドで指定します。

(e) 反映時の稼働トレースファイル

Datareplicator の稼働状況を取得するファイルです。Datareplicator の動作と性能についての情報を取得しています。反映時の稼働トレースファイルには、**反映トレースファイル**があります。

(f) 未反映情報ファイル

反映処理で発行した SQL 文がエラーになったときに、その SQL 文が出力されるファイルです。未反映情報ファイルを確認し、反映処理終了後にエラーになった SQL 文を再実行して、障害を回復できます。

未反映情報ファイルは、抽出側システムごとに 2 個作成されます。一つの未反映情報ファイルが満杯になると、出力対象をもう一方のファイルに切り替えます。これをスワップといいます。スワップした場合、スワップ先のファイルの状態に関係なく、ファイルの先頭から SQL 文が出力されます。

(g) 反映システム定義ファイル

抽出側システムとの接続方法又は反映側 HiRDB へのアクセス方法など、反映側 Datareplicator 全体の稼働環境の情報を定義するファイルです。

(h) 反映環境定義ファイル

反映情報キューファイル名又は反映ステータスファイル名など、反映処理の稼働環境の情報を定義するファイルです。

(i) 反映定義ファイル

抽出対象表、列と反映対象表、列との対応や反映グループなど、反映処理の詳細な情報を定義するファイルです。

(j) コマンドログファイル

Datareplicator のコマンドを実行した日付と時間の履歴を格納するファイルです。

(k) 更新情報定義ファイル

SAM ファイルを使ってデータ連動する抽出側 DB が PDMII E2 の場合に、反映処理の情報を定義するファイルです。SAM ファイル中の抽出対象表、列情報の指定と列の再定義情報を指定します。

(l) SAM ファイル

SAM ファイルを使ってデータ連動する場合の、ファイル転送プログラムで転送された PDMII E2 又は RDB1 E2 の更新情報抽出 SAM ファイルです。

PDMII E2 で作成する情報抽出 SAM ファイルは、PDMII E2 の稼働統計解析ユーティリティ (PDMJANL) で作成します。このユーティリティの実行時には、制御文 EXTRACT ステートメントで EXCHANGE = SWAP を指定するか又は EXCHANGE を省略してください。この指定以外でユーティリティを実行した場合は、SAM ファイル中に終了情報が出力されないことがあり、正しくデータ連動できません。

RDB1 E2 で作成する情報抽出 SAM ファイルは、RDB1 E2 の更新情報抽出機能で作成します。

SAM ファイルに使うデータセットは、**可変長ブロック形式**にしてください。固定長ブロック形式のデータセットは、データ連動には使えません。

PDMII E2 の稼働統計解析ユーティリティ (PDMJANL) については、マニュアル「VOS3 データベースマネジメントシステム PDMII, PDMII E2 ユティリティ」又は「VOS1 データベースマネジメントシステム PDMII E2 ユティリティ」を参照してください。

(m) 抽出対象外データ格納ファイル

SAM ファイルを使って抽出側 DB とデータ連動する場合に、更新情報定義ファイルの抽出文に指定していない表の更新データ、又は反映できない更新データ情報を出力するファイルです。

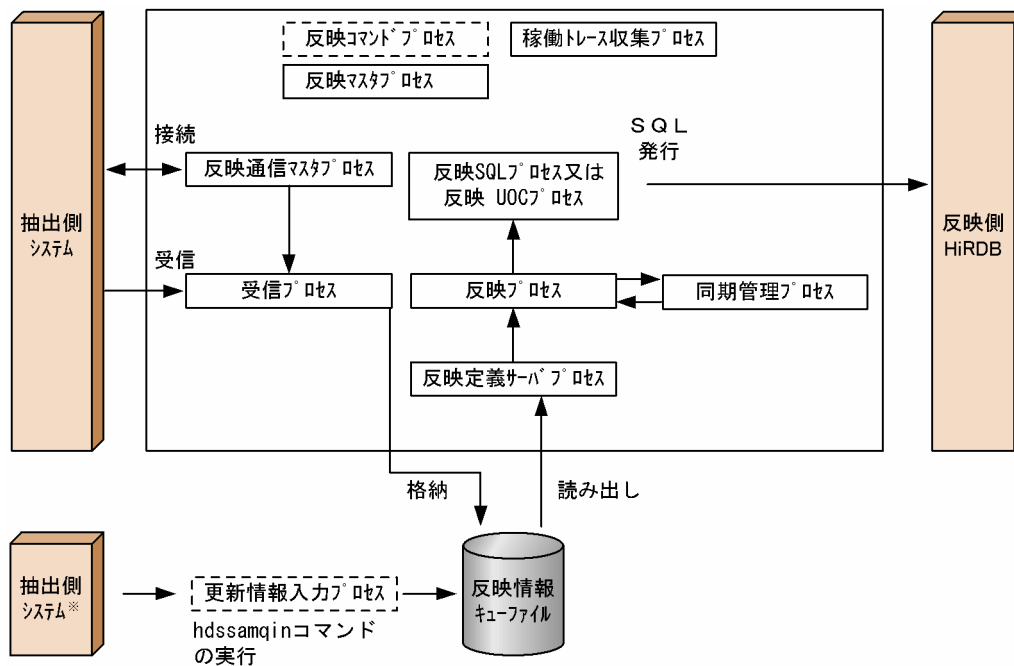
(n) 反映抑止対象リストファイル

反映時に、反映抑止によるエラーのスキップを行う場合に、作成します。反映抑止によるエラーのスキップについては、「3.3.10(1) 反映抑止によるスキップ」を参照してください。

(2) 反映処理のプロセス構成

反映処理のプロセス構成を次の図に示します。

図 3-22 反映処理のプロセス構成



(凡例)

□ : 常駐プロセス
 □ : 非常駐プロセス

注※ SAMファイルを使うメインフレーム側DBを示します。

(a) 反映コマンドプロセス

反映側 Datareplicator のコマンドを処理し、反映マスタープロセスに指示を出すプロセスです。

(b) 反映マスタープロセス

反映側 Datareplicator 全体を管理するプロセスです。

(c) 反映通信マスタープロセス

抽出側システムからの接続要求を受け付けるプロセスです。

(d) 受信プロセス

抽出側システムからの更新情報を受信し、反映情報キューファイルに格納するプロセスです。

(e) 反映定義サーバプロセス

反映プロセスを管理するプロセスです。

(f) 反映プロセス

反映 SQL プロセスを管理するプロセスです。

(g) 反映 SQL プロセス

更新情報を基に SQL 文を生成し、反映側 DB に SQL 文を発行するプロセスです。

(h) 反映 UOC プロセス

UOC を使って更新情報を加工して反映する場合に、反映側 DB に SQL 文を発行するプロセスです。

(i) 更新情報入力プロセス

SAM ファイルを使ってデータ連動するときに、メインフレーム側 DB (PDMII E2, RDB1 E2) から転送されてきた SAM ファイルを hdssamqin コマンドで反映情報キューファイルに格納するときに稼働するプロセスです。

(j) 稼働トレース収集プロセス

稼働トレース情報を取得するためのプロセスです。

(k) 同期管理プロセス

反映トランザクション同期機能を使用する際に、グローバルトランザクションを管理するプロセスです。

3.3.3 反映処理の方式

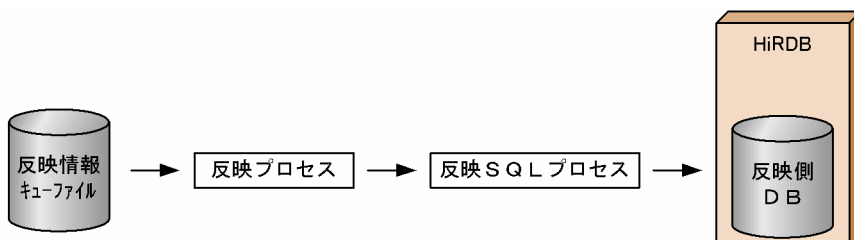
反映処理の方式には、トランザクション単位反映方式と表単位反映方式の二つがあります。どちらの方式にするかは、反映環境定義で指定します。システム設計の詳細については、「4.7.3(1) 反映処理の方式の設計」を参照してください。

(1) トランザクション単位反映方式

反映処理に反映プロセスと反映 SQL プロセスを一つずつ割り当てて反映する方式をトランザクション単位反映方式といいます。トランザクション単位反映方式では、反映情報キューファイルから一つずつトランザクションを読み込んで、逐次 HiRDB に反映します。

トランザクション単位反映方式のプロセス構成を次の図に示します。

図 3-23 トランザクション単位反映方式のプロセス構成



(2) 表単位反映方式

一つ又は複数の反映対象表ごとに反映グループを作り、各グループに並行に反映する方式を表単位反映方式といいます。

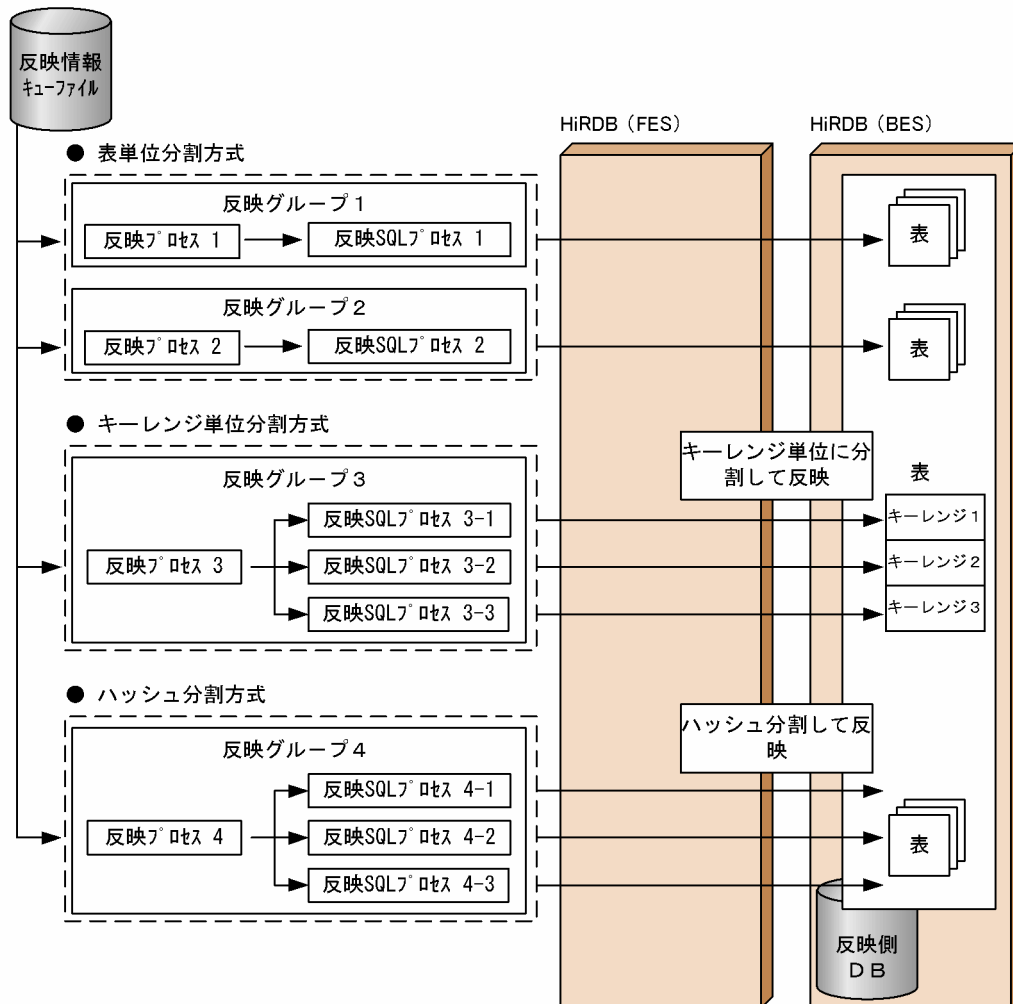
各反映対象表の更新量が均等であれば、反映処理の性能向上が期待できます。一方で、起動プロセス数が増えてメモリの使用量が多くなるため、メモリ容量を考慮した上で選択してください。

表単位反映方式には、次に示す種類があります。

- 表単位分割方式
- キーレンジ単位分割方式
- ハッシュ分割方式

表単位反映方式のプロセス構成を次の図に示します。

図 3-24 表単位反映方式のプロセス構成



(a) 表単位分割方式

一つ又は複数の反映対象表ごとに反映グループを作成します。この場合、反映グループごとに反映プロセスと反映 SQL プロセスが一つずつ割り当てられます。各反映プロセスは、反映情報キューファイルの更新情報を逐次読み込み、割り当てられた反映グループの表に対してだけ SQL 文を発行します。

反映側 HiRDB がパラレルサーバで、サーバ間横分割していない表の場合に、スループットの向上が期待できます。

(b) キーレンジ単位分割方式

一つの反映対象表で一つの反映グループを作成し、さらに反映グループの中でキーレンジ分割条件を指定します。このとき、反映グループに対して一つの反映プロセスが割り当てられて、さらにキーレンジ分割条件数分の反映 SQL プロセスが割り当てられます。反映プロセスは反映情報キューファイルの更新情報を逐次読み込んで、指定した条件に合致する反映 SQL プロセスだけが SQL 文を発行します。

反映側 HiRDB がパラレルサーバで、さらに一つの表を複数のユニットに横分割している場合に、その横分割と同様に反映側 Datareplicator でキーレンジ分割条件を定義すると、スループットの向上が期待できます。

ただし、1 回で反映するデータの量が少なかったり、又は反映対象表が大きくなかったりする場合など、反映側 HiRDB で負荷が掛からないデータ連動の場合には、キーレンジ分割をしない方がレンジの判定処理を実行しない分、反映処理の性能が良い場合があります。

また、次に示す HiRDB の構成のときは、キーレンジ単位分割方式を使ってもフロントエンドサーバとバックエンドサーバの通信が増えて性能が上がらないことがあります。

- フロントエンドサーバとバックエンドサーバが異なるマシン上にあるとき
- HiRDB の表がハッシュ分割されたとき

さらに、キーレンジ分割の条件指定が「一致」又は「範囲指定」だけなので、分割のキーになる列に対して連続した値が送信されると、特定のフロントエンドサーバに処理が集中してしまいます。

上記の理由でキーレンジ分割でも性能が上がらないときは、ハッシュ分割方式を使ってください。

(c) ハッシュ分割方式

一つの反映対象表で一つの反映グループを作成します。このとき、反映グループに対して一つの反映プロセスが割り当てられて、さらにハッシュ分割数分の反映 SQL プロセスが割り当てられます。反映プロセスは反映情報キューファイルの更新情報を逐次読み込んで、指定した条件に合致する反映 SQL プロセスだけが SQL 文を発行します。

キーレンジ分割でも性能が上がらないときに、ハッシュ分割方式を使ってください。ハッシュ分割方式を使うと、キーレンジ分割では 1 か所に集中していた連続データをハッシュ法で分割できます。特にマルチ FES 機能を使った HiRDB/パラレルサーバに反映するときは、ハッシュ分割方式の効果があります。

3.3.4 マルチ FES 機能に対応した反映方式

マルチ FES 機能とは、HiRDB/パラレルサーバで、一つのフロントエンドサーバに SQL 処理が集中してスループットが低下する場合に、複数のフロントエンドサーバを定義することで負荷を分散し、スループットを向上させるための HiRDB の機能です。

反映側 Datareplicator では表単位反映方式で反映処理を実行する場合に、このマルチ FES 機能に対応して、スループットを向上できます。

表単位分割方式では、反映グループごとに一つの反映先のフロントエンドサーバを割り当てることができます。キーレンジ単位分割方式では、キーレンジごとに反映先のフロントエンドサーバを割り当てることができます。マルチ FES 機能に対応する定義については、「5.10 反映定義」を参照してください。

ハッシュ分割方式で HiRDB サーバ間の通信の効率を上げる方法

ハッシュ分割方式では、分割されるエリア別に、反映するデータをどのフロントエンドサーバで処理するかをあらかじめ設定できます。これによってフロントエンドサーバの負荷を分散でき、またフロントエンドサーバと RD エリアの属するバックエンドサーバが同じサーバ上にあるときには、フロントエンドサーバとバックエンドサーバの間の通信オーバーヘッドを削減できます。

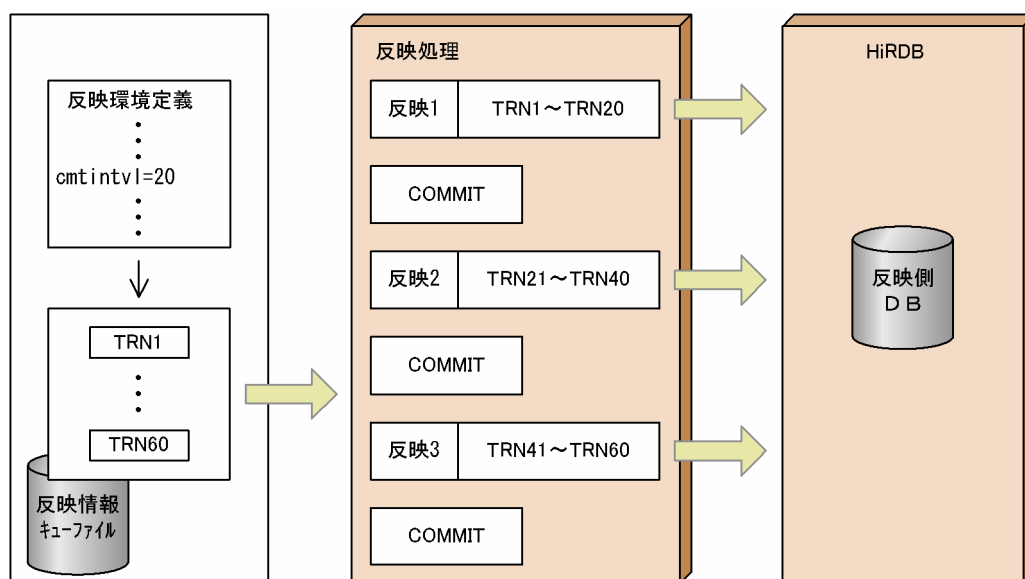
3.3.5 反映処理の単位

反映側 Datareplicator では、抽出側システムで実行された複数個のトランザクションを一つの単位として反映できます。複数のトランザクションを一つの単位として反映することで、性能面での向上が期待できます。

反映処理の単位は、反映環境定義の `cmtintvl`、`trncmtintvl`、`tblcmtintvl` オペランドで定義します。

反映処理の単位を次の図に示します。

図 3-25 反映処理の単位



3.3.6 トリガが定義されている表のデータ連動

表に定義されるトリガには、ユーザが定義するトリガと、HiRDBによって自動的に定義されているトリガがあります。トリガが定義されている表をデータ連動する場合に必要な作業について説明します。

(1) ユーザによるトリガが定義されている表のデータ連動

ユーザによるトリガが定義された表をデータ連動すると、抽出側 DB でトリガ SQL が実行されたにもかかわらず、データ連動後、反映側 DB でもトリガが実行されるため、DB に不整合が発生することがあります。

DB に不整合が発生することがあるのは、次のような場合です。

- BEFORE トリガでルーチン制御 SQL の代入文を使用した場合
- レプリケーション対象となる反映先の表に AFTER トリガを定義した場合

これらの条件に当てはまる場合は、反映環境定義で control_trigger オペランドに not_execute を設定してください。

注意

- 反映対象となる表へルーチン制御 SQL のファイル出力文を AFTER トリガに定義した場合、control_trigger オペランドに not_execute を指定するとファイルに出力されません。
- 反映対象表にトリガを定義して反映情報編集インタフェースの関数内で HiRDB にアクセスする場合、反映環境定義の control_trigger オペランドは無効になります。

(2) HiRDB によるトリガが定義されている表のデータ連動

参照制約が定義された表で、参照制約動作に CASCADE を指定している場合、その表には HiRDB によるトリガが定義されています。抽出元 HiRDB、反映先 HiRDB のデータ連動対象の表に参照制約動作に CASCADE が指定されている場合、次に示す条件を満たしてください。

前提条件

- 反映環境定義で control_reference_trigger オペランドに not_execute を指定する
- 反映環境定義で check_pending オペランドに use を指定する
- 参照制約が定義された表をすべてデータ連動の対象とする
- 抽出側と反映側の被参照表と参照表の制約を同じ構成にする

制限事項

反映情報編集 UOC には、参照制約を定義した表は指定できません。

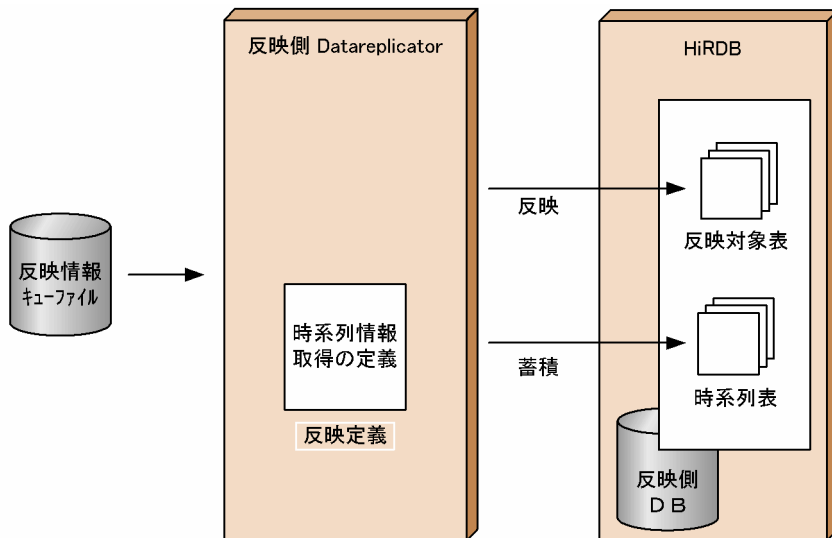
これらの条件を満たした上でデータ連動をすると、反映対象表に参照制約が定義されていた場合、該当する表が検査保留状態になります。表単位に整合性チェックユーティリティを実行して検査保留状態を解除してください。

3.3.7 時系列情報の取得

反映定義に時系列情報を取得する指定をすると、反映処理の実行時に反映側システムに対する更新情報を時系列情報表に蓄積できます。更新情報を時系列に取得することで、更新台帳として更新の履歴を検索したり、更新の頻度を分析したりできます。また、更新情報以外にも反映日付、反映時間、更新種別などの情報も取得できます。取得できる時系列情報については、「4.3.7 時系列情報表の作成」を参照してください。

時系列情報の取得の概念を次の図に示します。

図 3-26 時系列情報の取得の概念



3.3.8 反映側 Datareplicator の情報の取得

コマンドやシステム定義の指定で、次に示す反映側 Datareplicator の状態を取得できます。

(1) ステータス情報の取得

ステータス情報取得コマンド (`hdsstate` コマンド) を実行すると、反映側 Datareplicator の状態を確認するためのステータス情報を標準出力に取得できます。ステータス情報は、反映側 Datareplicator が稼働中のときだけ取得できます。

ステータス情報として、反映情報キューファイルの使用状況、受信処理の情報、反映処理の情報などを取得できます。取得例については、「7. コマンドの文法」の `hdsstate` コマンドの文法を参照してください。

(2) 稼働トレースの取得

Datareplicator の稼働状況を取得して性能を把握するために、稼働トレースを取得できます。稼働トレースを取得すると、次に示す運用ができます。

- 反映処理が遅く感じるときに、HiRDB の性能を最大限に引き出すようにシステムの設定を変更する目安にする（チューニングとして、HiRDB テーブル定義のインデクスの付与などで改善するべきか、Datareplicator の反映処理を表単位モードで実行して並列度を上げることで改善するべきかを決める）。
- 反映処理編集 UOC の性能が出ないときに、Datareplicator 側での問題か UOC 側での問題かを切り分ける。

反映側 Datareplicator で稼働トレースを取得するときは、反映システム定義の `int_trc_lvl` オペランドと `int_trc_filesz` オペランドを指定します。稼働トレース収集方式は、反映システム定義の `int_trc_unit` オペランドで指定します。さらに、データ連動識別子ごとの稼働トレースを取得するときは、反映環境定義の `int_trc_getl` オペランドを指定します。

取得した稼働トレースを出力するときは、`hdstrcredit` コマンドを使います。

稼働トレースの定義については「[5.8 反映システム定義](#)」及び「[5.9 反映環境定義](#)」を、`hdstrcredit` コマンドについては「[7. コマンドの文法](#)」の `hdstrcredit` コマンドの文法を参照してください。

3.3.9 ユーザOWNコーディングを使った更新情報の加工

データ連動の業務を補助するため、反映側 Datareplicator ではユーザOWNコーディング (UOC) を作成できます。反映側 Datareplicator の業務で作成できる UOC を次に示します。

- 反映情報編集 UOC
更新情報を任意に加工して反映できます。
- 列データ編集 UOC
任意の列の更新情報を、任意に加工して反映できます。

UOC については、「[8. ユーザOWNコーディング](#)」を参照してください。

3.3.10 反映時エラーのスキップ

反映処理中にエラーが発生しても、そのエラーを無視して反映処理を続行できます。これによって、反映不要なエラーが発生した場合、エラー対策する必要なく反映処理を続行できます。

反映時エラーのスキップには、次の 2 種類があります。

種別	説明
反映抑止によるスキップ (指定したエラーのスキップ)	エラーが発生してから、エラー情報 (更新情報識別子) を基に、スキップするエラーを指定します。 SQL エラーも含めて、すべてのエラーに適用できます。
SQL エラーのスキップ	あらかじめスキップする SQL エラーを SQLCODE で指定しておきます。

種別	説明
	一部の SQL エラーにだけ適用できます。適用できる SQL エラーについては、「5.9 反映環境定義」の skip_sqlcode オペランドを参照してください。

(1) 反映抑止によるスキップ

反映処理中に発生したエラーのメッセージの information に出力される更新情報識別子[※]を指定することで、そのエラーを無視して反映処理を続行できます。更新情報識別子の指定は、反映抑止対象リストファイルで行います。

メッセージの詳細情報については、「10.1.3(4) 詳細情報の説明」を参照してください。

注※

更新情報識別子は、hdsrefinfmt コマンドを実行することで確認することもできます。詳細については、「7. コマンドの文法」の hdsrefinfmt コマンドを参照してください。

(a) 前提条件

反映抑止によるスキップを使用できる抽出側システムの Datareplicator のバージョンを次の表に示します。

表 3-1 反映抑止によるスキップを使用できる Datareplicator

抽出製品	バージョン
HiRDB Datareplicator	04-00-/O 以降
HiRDB Datareplicator Version 5.0	05-03-/D 以降
HiRDB Datareplicator Version 6	06-01-/A 以降
HiRDB Datareplicator Version 7 以降	07-00 以降
XDM/DS (TMS/4V 抽出)	07-01 以降
XDM/DS (TMS/4V 以外の抽出)	08-03 以降
VOS3 Database Datareplicator	01-00 以降

注

次に示すシステム構成の場合、表 3-1 のバージョンでも反映抑止によるスキップを使用できません。

- 抽出側システムに HiRDB Datareplicator Extension を使用している。
- 反映側 Datareplicator への入力として、hdssamqin コマンドを使用している。

(b) 定義方法

反映抑止は、反映抑止対象リストファイルで指定します。反映抑止対象リストファイルについての詳細を次の表に示します。

表 3-2 反映抑止対象リストファイルについての詳細

項目	説明
ファイル名	HDSPATH/reflect_pass_list_データ連動識別子※ (例) HDSPATH/reflect_pass_list_a1
ファイル種別	UNIX 通常ファイル, 又は Windows ファイル
属性	hdsstart コマンド実行ユーザに対して, read 権限があります。

注※

- 連動識別子は, すべて小文字で指定する必要があります。
- 連動識別子が 00~09 の場合, 0~9 を指定してください。
0a~0f の場合は, a~f を指定してください。

反映抑止対象リストファイルでの指定形式を次に示します。

指定形式

```
制御コード = Trn [: Pos] [, Grp] #コメント
:
```

制御コード

運用に合わせて, 次から制御コードを指定します。

制御コード	説明
SKIP_TYPE_ONLY	通常の反映抑止の場合に指定します。 指定した更新情報識別子を抑止します。
SKIP_TYPE_UNTIL	データ連動回復の場合に指定します。

Trn [: Pos] [, Grp]

エラーメッセージの information に出力される更新情報識別子を指定します。Trn, Pos, 及び Grp の間にスペースやタブは記述できません。

指定	説明
Trn	更新情報が格納されたトランザクションの ID です。 メッセージの information 中の Extract Id の「:」までに 16 進数 (24 けた) として表示されます。
Pos	Trn が示すトランザクション内での更新情報の格納順位です。 メッセージの information 中の Extract Id の「:」以降に 10 進数で (1~4294967295) 表示されます。
Grp	反映定義の group 文で指定した反映グループ名称です。 Grp によって指定される抑止対象は, 反映処理方式によって異なります。反映処理方式ごとの Grp の抑止対象の違いを表 3-3 に示します。

表 3-3 反映処理方式ごとの Grp の抑止対象の違い

反映処理方式	Grp の抑止対象	
	Grp 指定あり	Grp 指定なし
表単位反映方式	Grp で指定した反映グループ内の Trm : Pos を抑止します。	すべての反映グループ内の Trm : Pos を抑止します。
トランザクション単位反映方式	すべての反映グループ内の Trm : Pos を抑止します (Grp の指定を無視します)。	すべての反映グループ内の Trm : Pos を抑止します。

規則

- 指定は 1 行につき一つです。
- 一つの制御コードから更新識別子までは、同じ行に指定します。
- 1 行は 512 バイト以内です。
- 行中に「#」を記述すると、「#」以降はコメントとして扱われます。

指定例

更新情報識別子の指定例を示します。

表単位反映方式で反映中に、次のメッセージが出力された場合を考えます。

メッセージの information

```
Fri May 23 16:29:42 2003 process: hdssqle[trngroup](2032) function: hds_sqe_hexeis
errorcode: KFRB03033-E 02 a1 (hdssqle.exe[trngroup]) HiRDB data base EXECUTE error occ
urred, table name = "k896201"."R003", SQLCODE = -803.
information: "C1"=1
KFPA11803-E Duplicate key value in unique index id=196861
Additional Transaction Info = 3e681b950000000000000012f.
Extract Id = 3e681b9500000000000000012f:1
```

反映グループ「trngroup」に対する、トランザクション「3e681b950000000000000012f」（格納順位「1」）の反映だけを抑止するとき、更新情報識別子は次のように指定します。

反映抑止対象リストファイルの内容

```
SKIP_TYPE_ONLY=3e681b9500000000000000012f:1, trngroup
```

(2) SQL エラーのスキップ

あらかじめ指定した SQLCODE のエラーが発生した場合に、そのエラーを無視して反映処理を続行できます。SQL エラーしかスキップできませんが、あらかじめスキップするエラーを指定することができるため、反映抑止によるスキップより手間が掛かりません。

スキップ対象とする SQLCODE は、反映環境定義の skip_sqlcode オペランドに指定します。ただし、暗黙的にロールバックされる SQL エラーに該当する SQLCODE を指定しても、スキップ指定は無視されて反映処理は停止します。

また、HiRDB が暗黙的なロールバックを行う SQL エラーのうち、通信エラー（SQLCODE が-722、又は-723）が発生した場合、その後の反映処理リトライ中に発生した SQL エラーをスキップすると、抽出側 DB と反映側 DB の不整合が発生するおそれがあります。これは、HiRDB の暗黙的なロールバック処理と反映処理のリトライが競合するためです。

SQL エラーをスキップさせないために、skip_sqlcode オペランドを指定する場合は、sqlerr_during_retry オペランドに stop を指定、又は指定を省略してください。これによって、抽出側 DB と反映側 DB の不整合が発生するおそれがあるときは、スキップ指定が無視されて、反映処理は KFRB03061-E メッセージを出力して停止します。KFRB03061-E メッセージを出力して停止したときは、HiRDB の暗黙的なロールバック処理が完了するのを待った後、反映処理を再起動してください。sqlerr_during_retry の指定と通信エラー後の反映処理リトライ中に発生した SQL エラーのスキップの関係について次の表に示します。

sqlerr_during_retry の指定	通信エラー後の反映処理リトライ中に発生した SQL エラーの SQLCODE	スキップの有無
stop 又は省略	skip_sqlcode オペランドに指定した SQLCODE	×※
	上記以外の SQLCODE	×
enable_skip	skip_sqlcode オペランドに指定した SQLCODE	○
	上記以外の SQLCODE	×

(凡例)

- ：スキップされます。
- ×：スキップされません。

注※

KFRB03061-E メッセージを出力します。

反映 SQL エラーのスキップの指定については、「[5.9 反映環境定義](#)」の skip_sqlcode オペランド及び sqlerr_during_retry オペランドの説明を参照してください。

3.3.11 マージ表の作成

複数の表を抽出側システムから抽出した更新情報を反映側の一つの表に反映（n 表を 1 表に反映）するとき、又は抽出側システムが構造型でもリレーショナル型データベースでもないときは、マッピングキーの制約でキー重複や該当行なしになって反映側システムでエラーになります。これを解決するため反映側システムでは、更新情報で INSERT するキーが重複したときに UPDATE に変換したり、UPDATE で該当行なしのときに INSERT に変換して格納したりできます。これをマージ表の作成といいます。

マージ表を作成するかどうかは、反映表定義に指定します。反映表定義の指定については、「[5.10 反映定義](#)」の load 文の説明を参照してください。

(1) マージ表の前提条件

反映処理で作成できるマージ表には、次に示す条件があります。

- 一般表であること。時系列情報表は対象外です。
- 非 FIX 表であること。FIX 表のときは、DELETE で null 値格納オプションを指定しないこと。ただし、n 表からのマージ表でないときは、この条件は前提ではありません。
- SQL プロセスを使った反映処理であること。反映情報編集 UOC で編集した反映は、マージ表を作成できません。
- マッピングキーが同じであること (n 表からのマージ表のとき)。また、マッピングキーの全体又は一部の構成列にユニークインデックスがあること。かつ、非マッピングキー列にユニークインデックスの指定がないこと。
- 抽出表に対する PURGE TABLE が実行されないこと (n 表からのマージ表のとき)。
- マッピングキーに対して更新されないこと (n 表からのマージ表のとき)。

(2) マージ表の処理

次の規則に従って、マージ表は作成されます。

- UPDATE したデータが Not Found になったときは、INSERT データとして格納されます。
- INSERT したデータがキー重複になったときは、UPDATE データとして格納されます。
- DELETE 対象のデータのときは、反映定義のオプションによって 行を削除するか、対象列を null 値にして格納されます。

(3) 繰返し列を含む表へのマージ表反映

繰返し列を含む表に要素番号を指定した UPDATE を実行すると、列内の要素データすべてではなく、更新された要素のデータだけが更新情報として送られます。このときマージ表反映の INSERT で指定するデータが不足しないように、NULL 値で補完されます。補完される方式を次に示します。

SQL		要素# < 最小要素#	最小要素# < 要素# < 最大要素#	最大要素# < 要素#
INSERT		補完不要		
UPDATE SET	列指定	補完不要		
	要素指定	NULL で補完	NULL で補完	補完しない
UPDATE ADD		NULL で補完	ありえない	補完しない
UPDATE DELETE		列値 NULL として反映		

注

要素 # は要素番号を示します。

最大要素#／最小要素#は、該当する繰返し列に対する更新情報にある要素の中で最大／最小の要素番号を示します。

(4) マージ表へ反映するデータの注意事項

- 連結演算データをマージ表へ反映する場合に、反映の UPDATE が該当行なし (SQLCODE = 100) になったときの INSERT で反映するデータは、連結演算データだけになります。
- 後方削除更新をしている BLOB 型又は BINARY 型のデータは、マージ表には反映しないでください。抽出側 DB と反映側 DB で不整合が発生します。

3.3.12 反映処理の同期点処理の指定

Datereplicator が反映処理を反映側 DB に要求すると、反映処理が完了したキューファイルの情報がステータスファイルに書き込まれます。この処理はトランザクション処理であり、トランザクションを正常にするかエラーにするか (コミット又はロールバック) を決めることを同期点処理といいます。同期点処理が完了した更新情報は、反映処理を再開始 (リラン) しても変更されません。

(1) 反映側 DB への同期点処理方式の種類

反映側 DB への同期点処理方式には、1 相コミット方式と 2 相コミット方式があります。

- 1 相コミット方式
反映側 DB に対して 1 回のコミットで同期点処理を実行する方式です。
- 2 相コミット方式
反映側 DB へのコミット要求を 2 回 (2 相) に分けて実行する方式です。

同期点処理方式は、反映システム定義の `commitment_method` オペランドで指定します。

同期点処理方式を変更したときは、反映側 Datereplicator を初期開始する必要があります。

(2) 1 相コミット方式と 2 相コミット方式の違い

反映側 DB へのコミットが正常にリターンすると、コミットまでの更新情報のキューファイルの情報を書き込んだ後、次のトランザクションを開始して更新情報を処理します。

反映側 DB がエラーになると、反映処理をいったん終了させて反映処理の再開始を待ちます。再開始の処理では、1 相コミット方式で再起動すると、前回コミット処理の実行中にエラーになったトランザクションから再実行します。この方式だと、反映側 DB のコミットの完了を Datereplicator が正しく認識できないことがあります (例: 反映側 DB からのリターン途中で通信エラーになって Datereplicator に正常にリターンしなかった)。このとき Datereplicator はエラーと認識して、HiRDB の再開始 (リラン) 時にトランザクションが再実行されます。もし、マッピングキーがユニークになっていないと、データが二重に挿入されるおそれがあります。

2相コミット方式のときは、上記のようなデータの二重挿入を防げます。反映側 Datareplicator が 2相コミットを定義できるバージョンのときは、2相コミット方式を定義することをお勧めします。

3.3.13 メッセージ出力の抑止

Datareplicator 稼働時には、syslog ファイル (Windows の場合はイベントログ)、及びエラー情報ファイルに稼働状況を伝えるためのメッセージが出力されます。

これらのメッセージは稼働状況を知るための重要な情報ですが、稼働環境によっては大量にメッセージが出力されてリソースを多く消費する場合があります。

このような場合に備えて、ユーザが情報として不要と判断したメッセージの出力を抑止するように指定できます。

メッセージの出力を抑止するためのオペランドを次に示します。

オペランド	抑止対象出力先
syslogout	syslog ファイル (又はイベントログ)
syslog_message_suppress	
info_message_out*	<ul style="list-style-type: none"> • syslog ファイル (又はイベントログ) • エラー情報ファイル

注

各オペランドの詳細については、「[5.8 反映システム定義](#)」を参照してください。

注※

info_message_out オペランドに suppress を指定して出力を抑止したメッセージのうち、抑止対象から除外してエラー情報ファイルに出力したいメッセージ番号がある場合は、except_suppress オペランドで指定できます。

syslog ファイル (又はイベントログ) へのメッセージの出力を抑止するためのオペランドを、混在して指定した場合のメッセージの出力結果を次に示します。

オペランド			メッセージの出力先	
info_message_out	syslogout	syslog_message_suppress	syslog ファイル (又はイベントログ)	エラー情報ファイル
suppress	指定不可	指定不可	×	×
nosuppress	true	指定あり	×	○
		指定なし	○	○
	false	指定不可	×	○

(凡例)

○：出力します。

×：出力しません。

3.3.14 排他エラー時の動作

反映処理が排他エラーとなった場合、反映側 Datareplicator はエラーとなったトランザクションの反映処理を内部的に再実行します。再実行を 3 回しても排他エラーが解消されない場合は、反映処理がエラー終了します。

3.4 Datareplicator ファイルシステム領域

Datareplicator ファイルシステム領域は、Datareplicator が使うファイルをキャラクタ型スペシャルファイルに効率良く格納するために使います。

注意

Datareplicator ファイルシステム領域を使えるのは、UNIX 版 Datareplicator のときだけです。Windows 版 Datareplicator では、Datareplicator ファイルシステム領域を設定できません。

3.4.1 Datareplicator ファイルシステム領域の目的

UNIX のキャラクタ型スペシャルファイルの最低長は 1MB です。抽出側 Datareplicator のファイルをキャラクタ型スペシャルファイルに割り当てると使用領域よりも空き領域の方が多くなって、格納効率が悪くなります。Datareplicator ファイルシステム領域を使うと、このような格納効率の悪化を防いで、ファイル領域を効率良く管理できます。

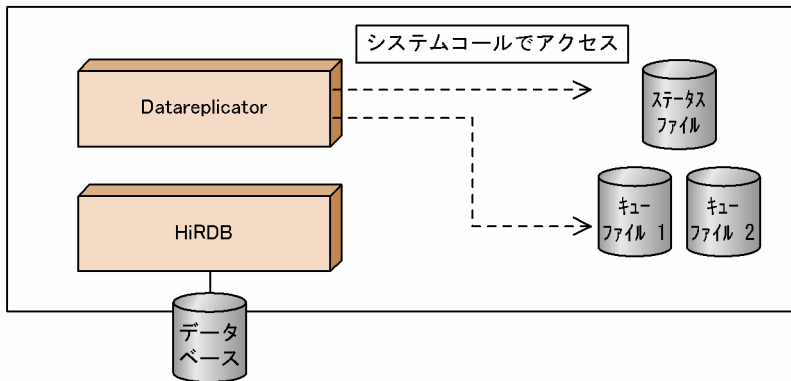
Datareplicator ファイルシステム領域を設定すると、Datareplicator が使う複数のファイル（システムファイル）を、一つのキャラクタ型スペシャルファイルにまとめることができます。

Datareplicator ファイルシステム領域と Datareplicator のファイルとの関係を次の図に示します。

図 3-27 Datareplicator ファイルシステム領域と Datareplicator のファイルとの関係

● 通常の Datareplicator のシステムファイルの場合

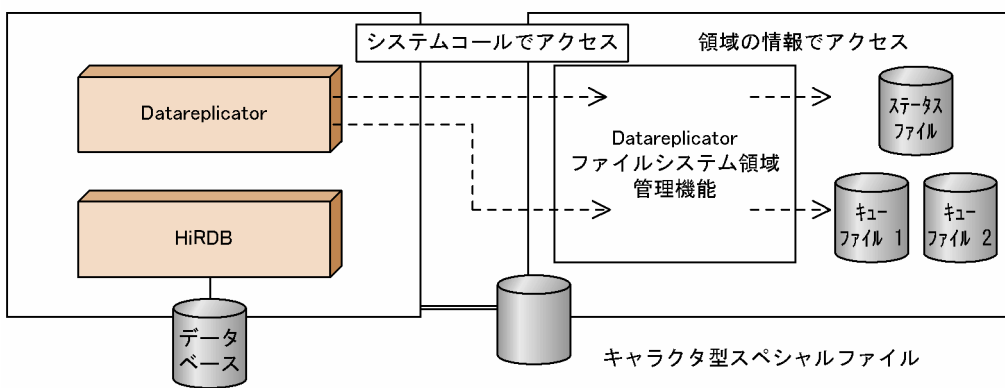
Datareplicator システム



● Datareplicator ファイルシステム領域を使う場合

Datareplicator システム

Datareplicator ファイルシステム領域



(1) Datareplicator ファイルシステム領域に格納できるファイル

システムファイルのうち、Datareplicator ファイルシステム領域に格納できるファイルを次の表に示します。

表 3-4 Datareplicator ファイルシステム領域に格納できるファイル

Datareplicator の種類	格納できるファイル
抽出側 Datareplicator	抽出サーバステータスファイル
	抽出情報キューファイル
	データ連動用連絡ファイル
反映側 Datareplicator	反映ステータスファイル
	反映情報キューファイル

3.4.2 Datareplicator ファイルシステム領域の作成方法

Datareplicator ファイルシステム領域を作成する手順を次に示します。

(1) 新規で作成する場合

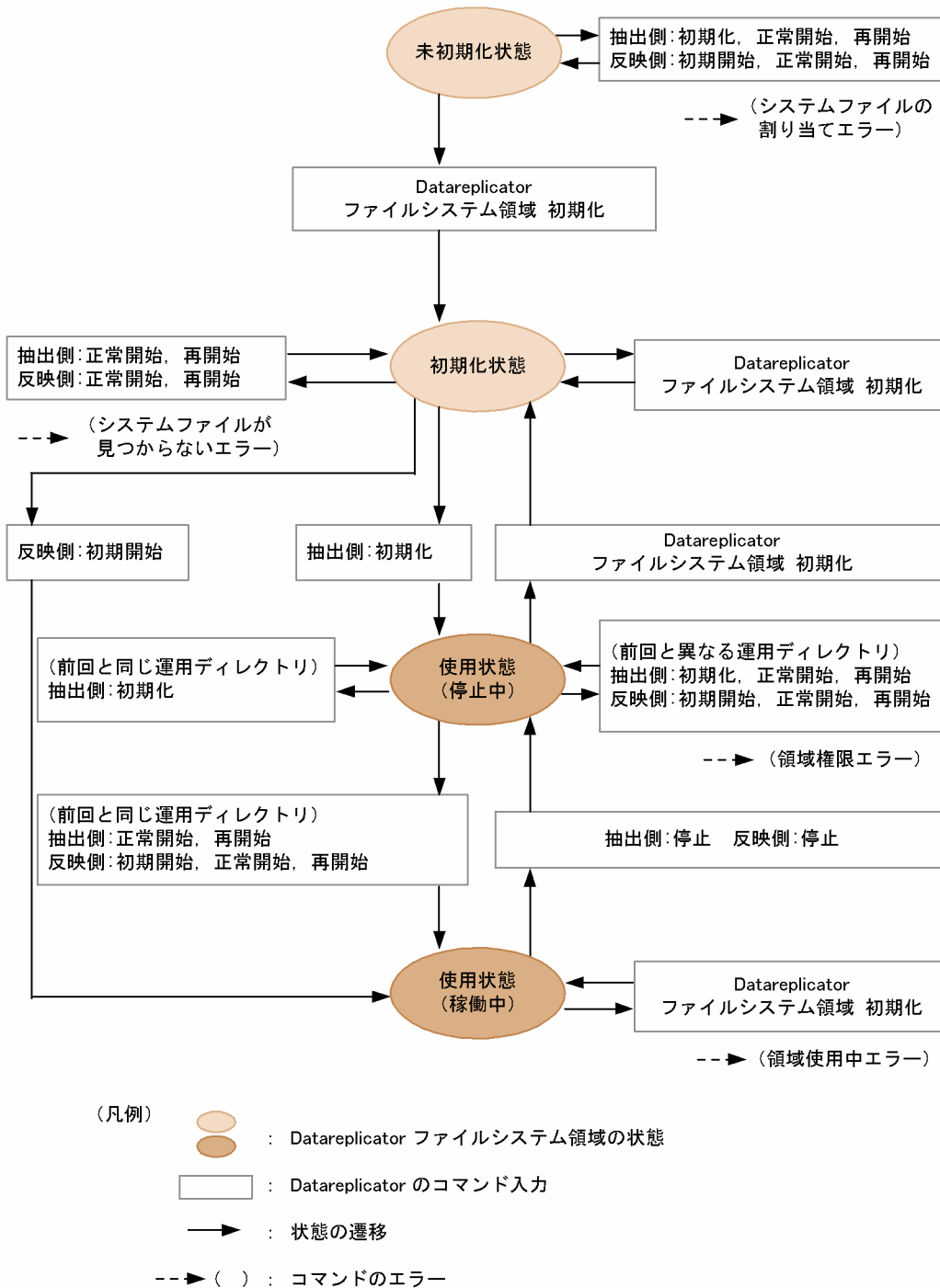
1. Datareplicator ファイルシステム領域として割り当てるキャラクタ型スペシャルファイルを準備します。
2. キャラクタ型スペシャルファイルを Datareplicator ファイルシステム領域として初期化します。初期化するときは、`hdsfmkfs` コマンドを使います。
3. Datareplicator の抽出環境定義 又は反映環境定義に `devicexx` オペランドを指定します。
4. システムファイルを Datareplicator ファイルシステム領域に割り当てます。

(2) 変更する場合

- 前回と同じ運用ディレクトリの Datareplicator のとき
Datareplicator を初期開始 (`hdsstart -i` コマンドを実行) してください。
- 前回とは異なる運用ディレクトリの Datareplicator のとき
Datareplicator ファイルシステム領域を `hdsfmkfs` コマンドで初期化した後、Datareplicator を初期開始 (`hdsstart -i` コマンドを実行) してください。

Datareplicator ファイルシステム領域の状態遷移を次の図に示します。

図 3-28 Datareplicator ファイルシステム領域の状態遷移

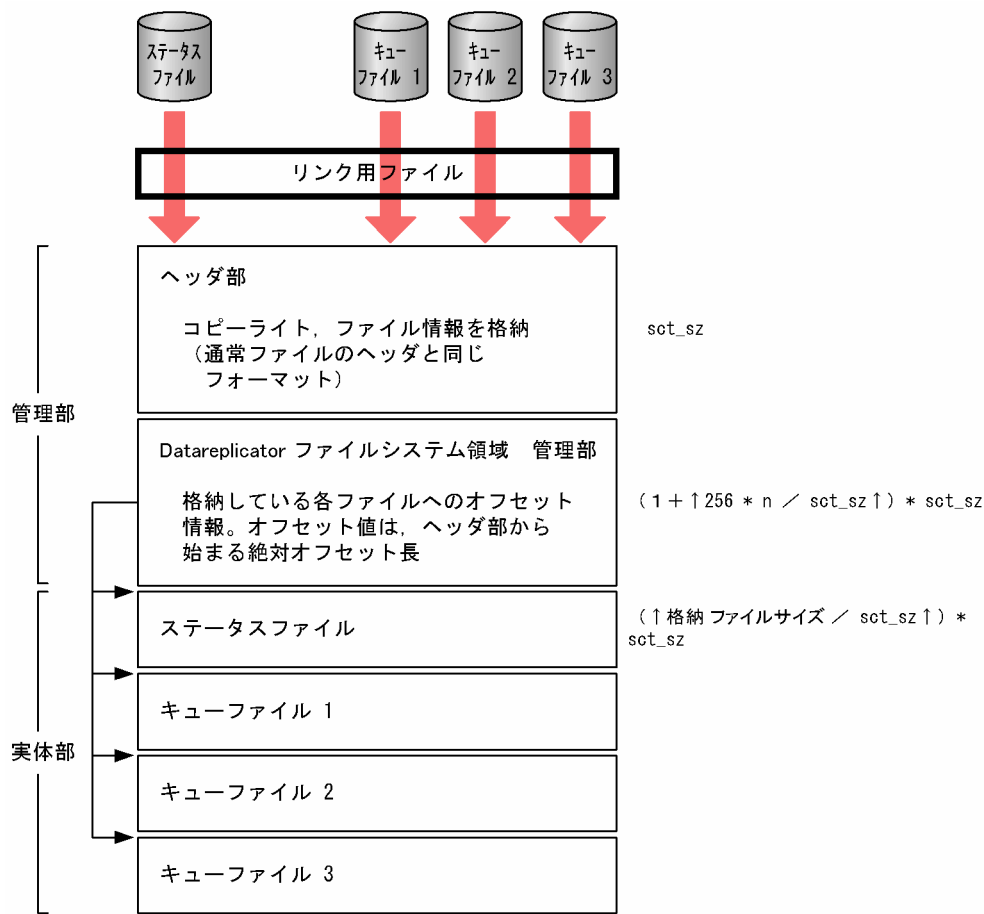


3.4.3 Datareplicator ファイルシステム領域の構造

Datareplicator ファイルシステム領域は、領域を管理する管理部、割り当てたファイルを格納する実体部、Datareplicator ファイルシステム領域に対してリンクしているファイルから構成されます。Datareplicator ファイルシステム領域に割り当てたシステムファイルにアクセスする各プロセスやコマンドは、このリンクしたファイル名を使います。

Datareplicator ファイルシステム領域の構造を次の図に示します。

図 3-29 Datareplicator ファイルシステム領域の構造



(凡例)

sct_sz : Datareplicator ファイルシステム領域のセクタ長

(1) Datareplicator ファイルシステム領域を割り当てるときの規則

Datareplicator ファイルシステム領域の実体部にシステムファイルを割り当てるときの規則を次に示します。

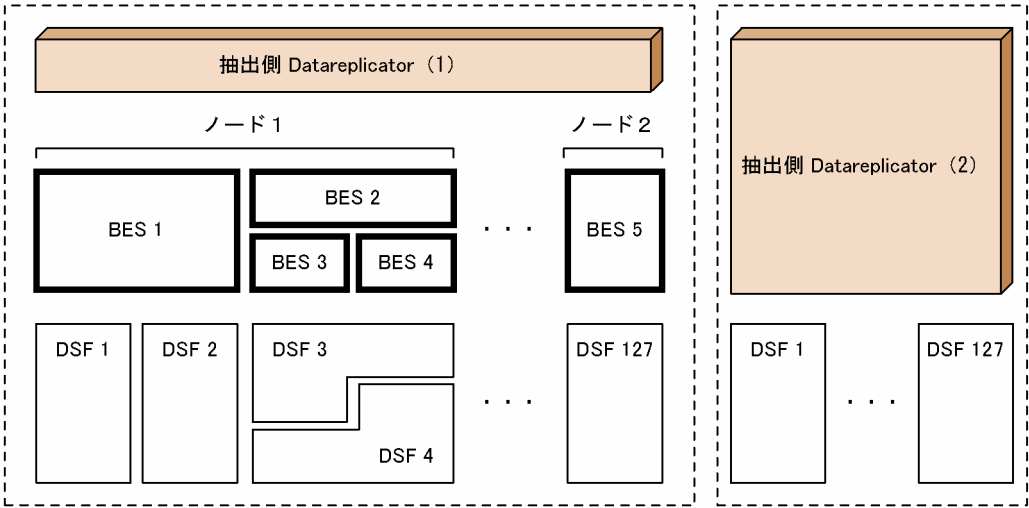
- hdsfmkfs コマンドの `-l` オプションで指定したファイル数を超えるシステムファイルを割り当てることはできません。
- 一つの Datareplicator ファイルシステム領域に割り当てられるファイルサイズは、次に示す算出式の範囲内であればなりません。

$$\begin{aligned}
 & \text{Datareplicator ファイルシステム領域サイズ} \geq \text{sct_sz} && \text{(ヘッダ部)} \\
 & + (1 + \lceil 256 * n / \text{sct_sz} \rceil) * \text{sct_sz} && \text{(管理部)} \\
 & + \sum n (\lceil \text{格納ファイルサイズ} / \text{sct_sz} \rceil) * \text{sct_sz} && \text{(割り当てファイルサイズ累計)}
 \end{aligned}$$

(凡例)
 sct_sz : Datareplicator ファイルシステム領域のセクタ長
 n : 格納するファイル数

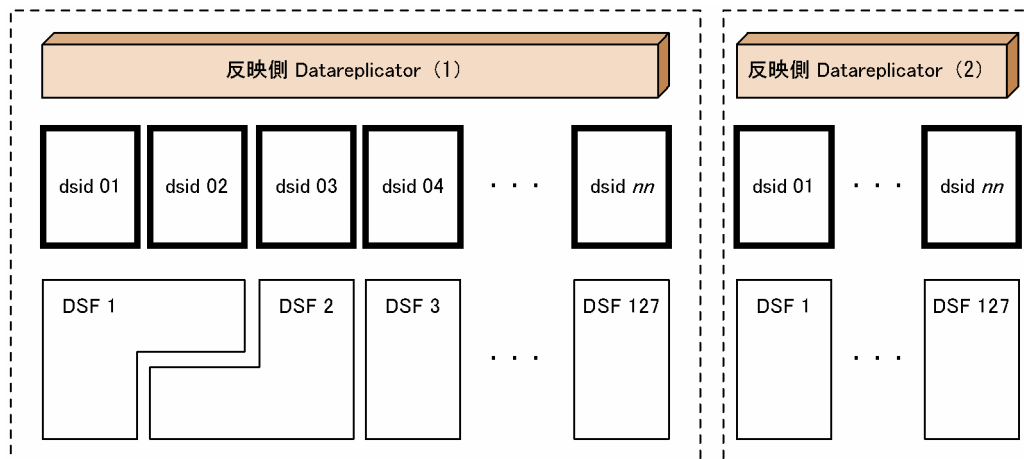
- 一つのシステムファイルを複数の Datareplicator ファイルシステム領域に分割して割り当てることはできません。
- 異なる運用ディレクトリの Datareplicator のファイルを、同じ Datareplicator ファイルシステム領域に割り当てることはできません。
- 同じ Datareplicator 内で、かつ同じマシン内で使うシステムファイルは、同じ Datareplicator ファイルシステム領域に割り当てることができます。割り当てる例を図 3-30 と図 3-31 に示します。

図 3-30 抽出側システムでの Datareplicator ファイルシステム領域への格納例



(凡例)
 BES : バックエンドサーバ
 DSF : Datareplicatorファイルシステム領域

図 3-31 反映側システムでの Datareplicator ファイルシステム領域への格納例



(凡例)

DSF : Datareplicatorファイルシステム領域

- 反映側 Datareplicator で複数のデータ連動識別子 (dsid) を定義している場合に部分初期開始をするためには、Datareplicator ファイルシステム領域をデータ連動識別子ごとに割り当てる必要があります。一つの Datareplicator ファイルシステム領域を複数のデータ連動識別子で共用しているときには、部分初期開始はエラーになります。

3.4.4 Datareplicator ファイルシステム領域を使うときの注意

Datareplicator ファイルシステム領域を使うときに注意する項目を次に示します。

- 反映側 Datareplicator で複数のデータ連動識別子 (dsid) を定義しているときは、Datareplicator ファイルシステム領域はデータ連動識別子に対応させて、各領域は異なるディスクに作成することをお勧めします。このようにファイルを割り当てると、ディスクの入出力を分散できます。
- 抽出側 Datareplicator で同じノード内に複数のバックエンドサーバがあるときは、Datareplicator ファイルシステム領域はバックエンドサーバに対応させて、各領域は異なるディスクに作成することをお勧めします。このようにファイルを割り当てると、ディスクの入出力を分散できます。

3.5 滞留監視機能

滞留監視機能を使用して、抽出側 Datareplicator、又は反映側 Datareplicator で更新の時間差を監視できます。

3.5.1 滞留監視機能の概要

滞留監視機能とは、更新データの滞留時間を監視する機能です。滞留時間とは、抽出側 DB のトランザクションがコミットして更新情報がシステムログファイルに格納された時刻と、任意のデータ連動処理が完了した時刻との差のことです。滞留監視機能の概念を図 3-32 に、滞留監視する項目を表 3-5 に示します。

図 3-32 滞留監視機能の概念

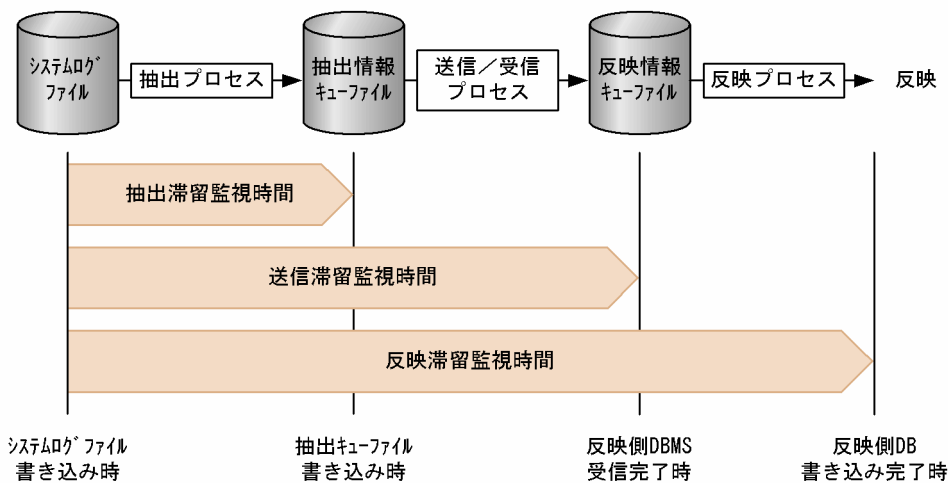


表 3-5 滞留監視する項目

監視項目	説明	監視プロセス
抽出滞留時間	更新情報がシステムログファイルに格納された時刻と、その更新情報が抽出情報キューファイルに書き込まれる時刻との差のことです。 更新情報が抽出情報キューファイルに書き込まれるときに監視します。抽出用の抽出情報キュー I/O バッファ内に複数の更新情報が存在する場合、それらのうち、最初にコミットした更新情報が書き込まれる時刻との差が、抽出滞留時間になります。	抽出プロセス
送信滞留時間	更新情報がシステムログファイルに格納された時刻と、その更新情報が送信されて反映側 DBMS で受信が完了した時刻との差のことです。 送信プロセスの送信間隔ごとに監視します。複数の更新情報が一度に送信される場合は、それらのうち、最初に送信された更新情報の受信が	送信プロセス

監視項目	説明	監視プロセス
	完了した時刻との差が、送信滞留時間になります。	
反映滞留時間	更新情報がシステムログファイルに格納された時刻と、反映側 DB でその更新情報が反映された時刻との差のことです。 反映側 DB でのコミット間隔ごとに監視します。	反映プロセス

なお、Datareplicator Extension の場合は抽出滞留時間の監視をサポートしていません。Datareplicator Extension の滞留監視機能については、マニュアル「HiRDB データ連動拡張機能 HiRDB Datareplicator Extension」を参照してください。

3.5.2 滞留監視機能の使用

(1) オペランドの指定とメッセージ出力の関係

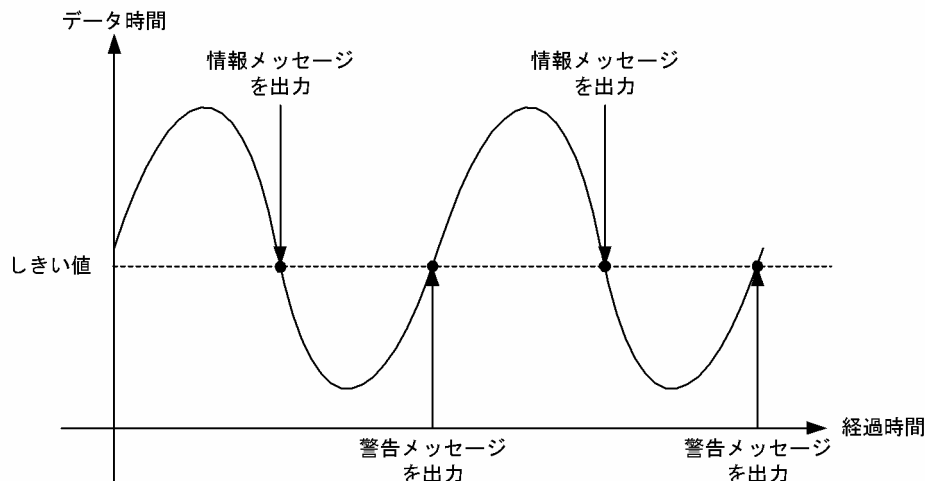
滞留監視機能を使用する場合は、次のオペランドを指定してください。

オペランドの詳細については、「5.3 抽出環境定義」、「5.4 送信環境定義」及び「5.9 反映環境定義」を参照してください。

- extract_delay_limit_time オペランド
- send_delay_limit_time オペランド
- reflect_delay_limit_time オペランド

滞留監視機能を使用すると、滞留時間が上記のオペランドに指定されているしきい値を超えたタイミングで 1 回、しきい値内に戻ったタイミングで 1 回警告メッセージが出力されます。しきい値を超えている間、及び範囲内にある間に継続して出力されることはありません。メッセージの出力タイミングを次の図に示します。

図 3-33 メッセージの出力タイミング



(2) 抽出側 watchintvl オペランドによる滞留監視

抽出側 DB では、watchintvl オペランドに指定した滞留監視のタイミングで、抽出マスタプロセスが各ノードのその時点での滞留状態を出力します。しきい値を超えた状態（滞留超過）であれば、KFRB00700-W (00066) メッセージが出力されます。一度、滞留超過した状態からしきい値内に戻った状態であれば、KFRB00700-W (00067) メッセージが出力されます。滞留超過したまま、又はしきい値内のままであれば、メッセージは出力されません。

オペランドの詳細については、「5.2 抽出システム定義」を参照してください。

なお、滞留状態と watchintvl のタイミングによっては、同一のメッセージが連続して出力されたり、KFRB00700-W (00066) メッセージは出力されなくても、KFRB00700-W (00067) メッセージが出力されたりする場合があります。

(3) 警告メッセージの出力先

滞留監視機能の警告メッセージは、エラー情報ファイル及び syslog ファイルに出力されます。監視項目ごとの警告メッセージの出力先とメッセージ番号について次の表に示します。出力されるメッセージについては、「10.2 メッセージの詳細」を参照してください。

表 3-6 監視項目ごとの警告メッセージ出力先とメッセージ番号

監視項目	出力場所	出力先	メッセージ番号
抽出滞留時間 送信滞留時間	滞留が発生したサーバマシン	<ul style="list-style-type: none"> syslog ファイル※1 エラー情報ファイル 	KFRB00066-W
	抽出対象の HiRDB のマネージャが稼働するサーバマシン	<ul style="list-style-type: none"> syslog ファイル マスタエラー情報ファイル 	KFRB00700-W
反映滞留時間	反映処理が稼働するサーバマシン	<ul style="list-style-type: none"> syslog ファイル※2 エラー情報ファイル 	KFRB00066-W

注※1

抽出システム定義の `node_syslogout` オペランドに「true」を指定した場合に出力されます。

注※2

反映システム定義の `syslogout` オペランドに「true」を指定した場合に出力されます。

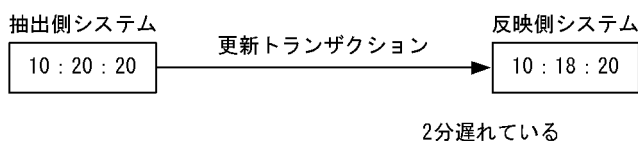
3.5.3 注意事項

滞留監視機能を使用する上での注意事項を次に示します。

- 滞留時間は、ステータス情報取得コマンド (`hdestate`, `hdsstate`) で参照できます。このため、滞留時間を参照するには、Datareplicator が稼働している必要があります。
- 滞留監視機能は、抽出プロセス、送信プロセス及び反映プロセスが稼働している場合だけ有効です。各プロセスが停止している場合は監視しません。
- 滞留監視機能は、日本標準時 (JST-9) だけに対応しています。
- PDMII の SAM ファイルを入力として反映を行う場合は、`hdssamqin` コマンドの実行時間を起点として反映滞留時間の監視を行います。
- 最後に更新した SQL の更新時間と、コミットの処理時刻の差が大きいトランザクションがある環境では、滞留監視機能を使用しないでください。抽出滞留時間では滞留超過 (オペランドのしきい値を超えている状態) が発生していなくても、送信滞留時間又は反映滞留時間では滞留超過になることがあります。
- 抽出側システムと反映側システムとで時刻に差がある場合、滞留時間を正確に監視できないため、注意してください。

滞留監視機能の運用上の注意事項について、反映滞留時間のしきい値に 1 分を指定した場合の例を、[図 3-34](#) と [図 3-35](#) に示します。

図 3-34 反映側システムの時刻が遅れている場合



更新情報がシステムログファイルに格納されてから反映処理が完了するまでに 1 分以上掛かると、警告メッセージが出力されます。ただし、[図 3-34](#) の環境の場合、反映側システムの時刻が 2 分遅れているため、反映処理が完了するまでに 3 分以上掛かったときに、警告メッセージが出力されることになります。この場合、反映側システムの時刻を抽出側システムに合わせることで設定したしきい値どおりの動作をしますが、時刻を変更するには、稼働中の関連製品をいったん停止させる必要があります。

図 3-35 反映側システムの時刻が進んでいる場合

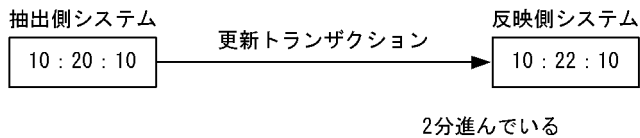


図 3-34 の環境と同じように、更新情報がシステムログファイルに格納されてから反映処理が完了するまでに 1 分以上掛かると、警告メッセージが出力されます。しかし、反映側システムの時刻が 2 分進んでいるため、反映処理が完了する前に警告メッセージが出力されてしまいます。この場合、反映側システムの時刻を抽出側システムに合わせることで、回避できます。ただし、反映側システムの時間をバックデートした場合は、バックデートする前の時間に追い付いたことを確認した後に、関連する製品を起動する必要があります。

補足

上記のような事例を防ぐためにも、抽出側システムと反映側システムで時間差がある場合、あらかじめ、両方のシステムの時刻を合わせておくことをお勧めします。合わせない場合は、抽出側システムと反映側システムとの時刻の差を考慮したしきい値を指定してください。

3.6 反映トランザクション同期機能

反映トランザクション同期機能について説明します。

この機能を使用する前に、必ず「3.6.8 注意事項」を参照してください。

3.6.1 反映トランザクション同期機能の概要

HiRDB で発生するトランザクションには、各バックエンドサーバで発生するトランザクション（トランザクションブランチ）と、それらをまとめるトランザクション（グローバルトランザクション）があります。通常、Datareplicator はトランザクションブランチ単位で連動するため、トランザクションブランチ反映直後は、グローバルトランザクション単位でのデータの整合性は保証していません。

反映トランザクション同期機能を使用すると、抽出側 HiRDB で発生したグローバルトランザクション単位でのデータの整合性を、トランザクションブランチの反映時に保証できます。

3.6.2 反映トランザクション同期機能の準備

(1) 前提条件

反映トランザクション同期機能は、抽出側システム及び反映側システムが、どちらも Datareplicator の場合にだけ使用できます。

この機能を使用するためには、次の条件をすべて満たす必要があります。

- 抽出側 HiRDB のバージョンが 07-03 以降で、かつパラレルサーバである。
- 抽出側 Datareplicator のバージョンが 07-04 以降である。
- 反映側 Datareplicator のバージョンが 07-04 以降である。

なお、次に示す OS のどれかを使用している場合、反映側 HiRDB のバージョンは 07-03 以降である必要があります。

- HP-UX (IPF)
- Linux (IPF)
- Windows

また、HiRDB Datareplicator Extension を使用するシステムでは、XA インターフェースを使用できないため、グローバルトランザクション単位でのデータの整合性は保証できません。

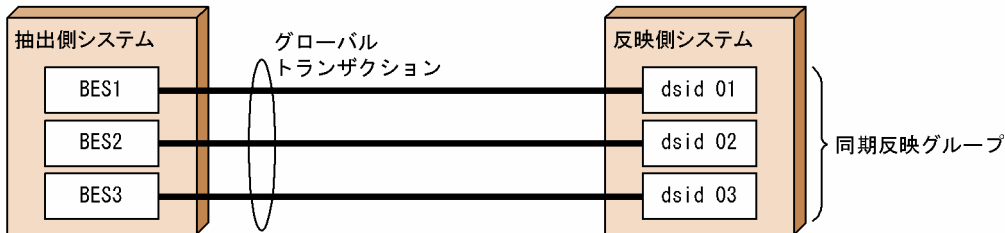
(2) システム構成

(a) 抽出側システムのバックエンドサーバと反映側システムのデータ連動識別子との対応

この機能を使用する場合、抽出側 HiRDB のバックエンドサーバごとに対応するデータ連動識別子を用意します。このとき、データ連動識別子は同期反映グループとしてまとめておく必要があります。また、イベント制御用テーブル (hde_dtbl) を抽出対象のバックエンドサーバに配置しておきます。

反映トランザクション同期機能を使用する場合の構成例を、次の図に示します。

図 3-36 反映トランザクション同期機能を使用する場合の構成例



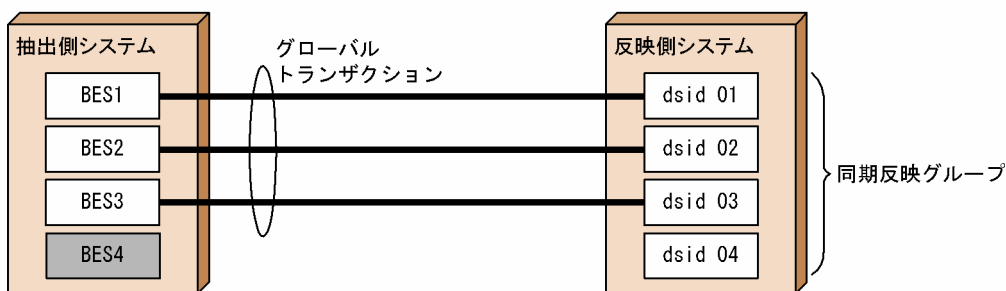
(凡例)

BES : バックエンドサーバ
dsid : データ連動識別子

なお、一つの抽出側 HiRDB から抽出するグローバルトランザクションは、一つの反映側システムで受信する必要があります。このため、データ連動対象の表を配置するバックエンドサーバ数の最大は、一つの反映側 Datareplicator で受信できる最大数となります (反映側システムが UNIX の場合は 128, Windows の場合は 63)。

抽出対象外のバックエンドサーバについては、グローバルトランザクションの対象になりません。次の図に示すように、抽出対象外のバックエンドサーバに対応したデータ連動識別子を同期反映グループにすると、同期を待ち続けるため、反映処理が進まなくなります。

図 3-37 同期を待ち続ける構成の例



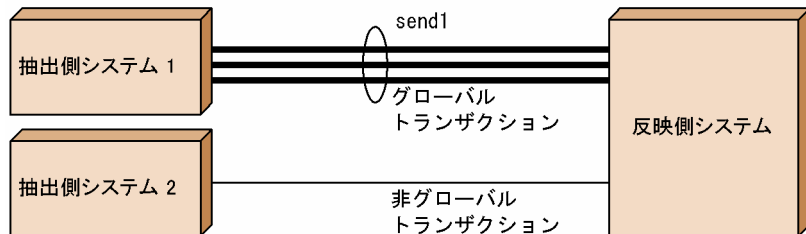
(凡例)

BES : バックエンドサーバ
dsid : データ連動識別子
■ : 抽出対象外のバックエンドサーバ

(b) 抽出側システムと反映側システムの構成

この機能を使用する場合、一つの反映側システムで受信できるグローバルトランザクションは一つだけです。また、一つの反映側システムで、グローバルトランザクションと非グローバルトランザクションを混在して受信できます。構成例を次の図に示します。

図 3-38 抽出側システムと反映側システムの構成例 1

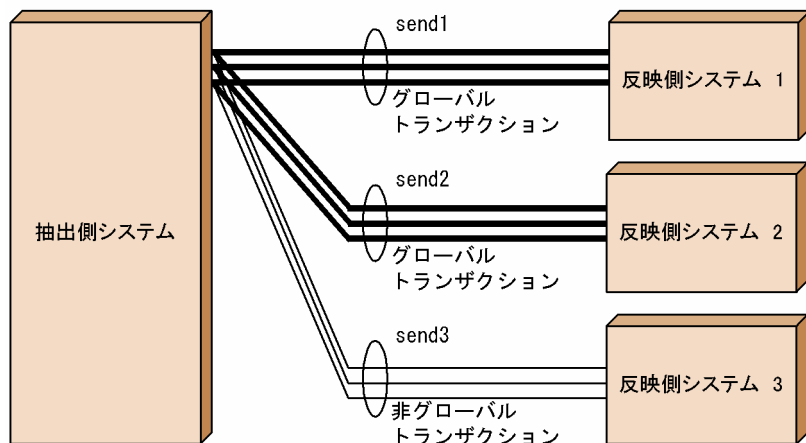


説明

抽出側システム 1 からグローバルトランザクションを、抽出側システム 2 から非グローバルトランザクションを、一つの反映側システムに送信します。

また、一つのグローバルトランザクションを複数の送信先識別子に送信することもできます。グローバルトランザクションの送信の有無は、送信先識別子ごとに指定できます。構成例を次の図に示します。

図 3-39 抽出側システムと反映側システムの構成例 2

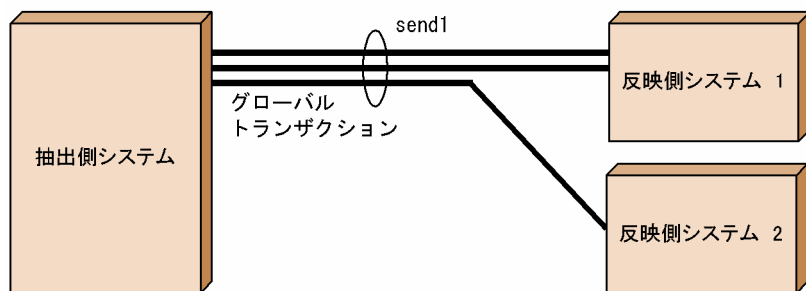


説明

一つの抽出側システムから、複数の反映側システムにグローバルトランザクション及び非グローバルトランザクションを送信します。

次の図に示すように、一つの送信先識別子のセットを複数の反映側システムへ分散する構成にはできません。このような構成の場合、同期は保証しません。

図 3-40 同期できない構成の例



(3) 指定するオペランド

反映トランザクション同期機能を使用する際に指定しておく必要があるオペランドを次の表に示します。

表 3-7 反映トランザクション同期機能を使用する際に指定するオペランド

PP	定義ファイル	オペランド	説明
HiRDB	システム共通定義	pd_rpl_reflect_mode = uap ^{*1}	トランザクションブランチ情報 (Datareplicator がこの機能を実現するために必要な情報) を、システムログファイルへ出力するためのオペランドです。詳細については、マニュアル「HiRDB システム定義」を参照してください。
Datareplicator	反映システム定義	syncgroup001 ^{*2}	同期反映グループを構成するためのオペランドです。
	送信環境定義	reflect_mode = uap ^{*3}	トランザクションブランチ情報を、反映側 Datareplicator へ送信するためのオペランドです。
		eventsync	同期イベント (同期反映グループが COMMIT を発行する契機となるイベント) の番号を指定します。

注※1

データ連動中に指定値を uap から server, 又は server から uap に変更すると、反映処理実行時に KFRB03312-E メッセージが出力されて、反映処理は停止します。また、トランザクションブランチ情報は送信されなくなります。回避するためには、同期反映グループを解除して再起動する必要があります。

注※2

トランザクションブランチ情報を送信する状態で syncgroup001 オペランドが定義されていないと、KFRB02066-W メッセージが出力されて、非同期で反映が行われます。トランザクションブランチ情報を送信しない状態で syncgroup001 オペランドが定義されていると、ポートチェックエラー (詳細コード = 26) になります。

注※3

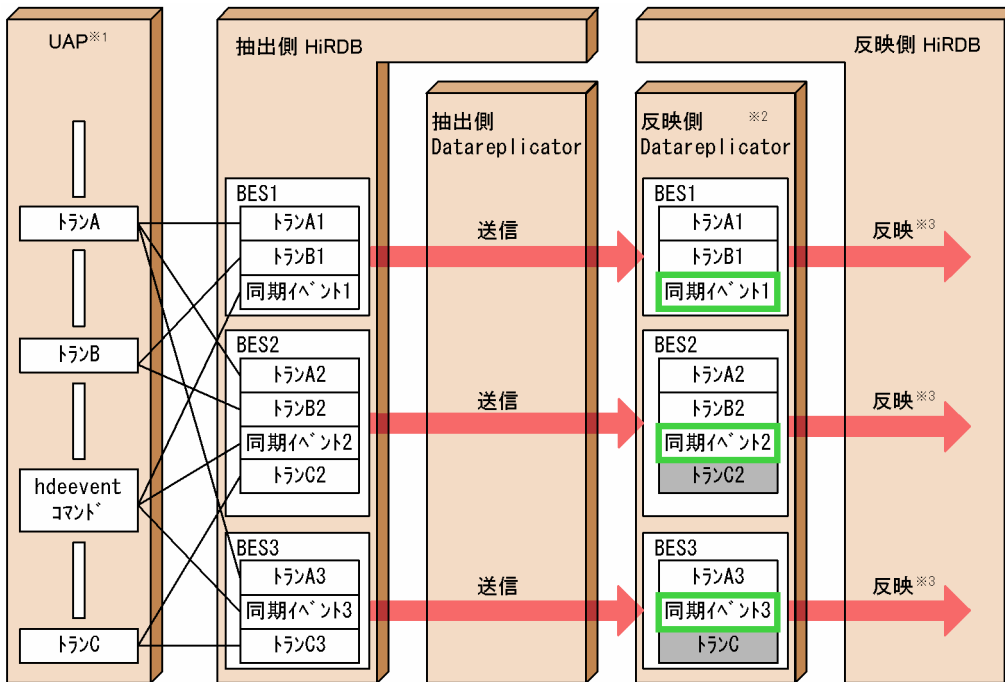
HiRDB の pd_rpl_reflect_mode オペランドの指定値が server の場合は、reflect_mode オペランドに uap を指定しても、hdestart コマンド実行時にエラーとなり、KFRB00726-E メッセージが出力されます。

また、reflect_mode オペランドに uap を指定する場合は、送信環境定義の overwrite オペランドに false を指定してください。false 以外を指定すると、KFRB00846-E メッセージが出力されてエラーになります。

3.6.3 反映トランザクション同期機能の処理の流れ

反映トランザクション同期機能は、イベント機能を利用して実現します。処理の流れを次の図に示します。

図 3-41 反映トランザクション同期機能の処理の流れ



(凡例)

- トラン A : BES1~3に対して更新するトランザクション
- トラン B : BES1~2に対して更新するトランザクション
- トラン C : BES2~3に対して更新するトランザクション
- BES : バックエンドサーバ
- 同期イベント : COMMITを発行するタイミング
- トラン : 反映の対象にならないトランザクション

注※1

UAP は次の順番で実行します。

1. トラン A
2. トラン B
3. hdeevent コマンド
4. トラン C

hdeevent コマンドには、送信環境定義の eventsync オペランドに指定した同期イベントの番号をイベントコードとして指定します。

注※2

BES1, BES2, 及び BES3 の更新情報の反映処理は、それぞれ異なる反映プロセスによって実行されます。そのため、トラン A 及びトラン B の反映処理は非同期に実行されます。また、トラン A 及びトラン B の反映処理を実行した時点では、まだ COMMIT を発行しません。

注※3

反映プロセスは、同期イベントを反映した時点でほかの反映プロセスを待機します。BES1 から BES3 までの反映プロセスすべてが同期イベントを反映して、全バックエンドサーバで整合性が取れた時点（全グローバルトランザクションを決着させた後）で、全反映プロセスは一斉に COMMIT を発行します。したがって、トラン C に対する COMMIT は、次に同期イベントが実行されるまで発行されません。

注意事項

この機能を使用すると、反映側 Datareplicator から発生するトランザクション（反映トランザクション）を COMMIT する契機は、同期イベントだけになります。このため、定期的に同期イベントを発生させないと反映トランザクションが長大化し、HiRDB のリソース不足などが発生します。反映側 HiRDB のリソースを考慮して、反映トランザクションがエラーにならない間隔で、定期的に同期イベントを実行する必要があります。

定期的に同期イベントを発生させるためのシェルスクリプトのサンプルを次に示します。次のサンプルでは、5 秒間隔で同期イベントを発生させます。

```
#!/bin/sh
if [ -z "$1" ]
then
    echo "Please specify syncevent number."
    exit 0
fi
if [ $1 -le 0 -o $1 -ge 129 ]
then
    echo "Please specify numerical value from 1 to 128. "
    exit 0
fi
if [ -z $PDUSER ]
then
    echo "Please set PDUSER."
    exit 0
fi
echo "Start event of synchronization."
execf=OK
while [ -n $execf ]
do
    /opt/hirdbds/bin/hdeevent -n $1
    if [ $? = 0 ]
    then
        echo "event $1 0.K"
    else
        echo "Execution of hdeevent failed."
        exit 0
    fi
done
```

sleep 5

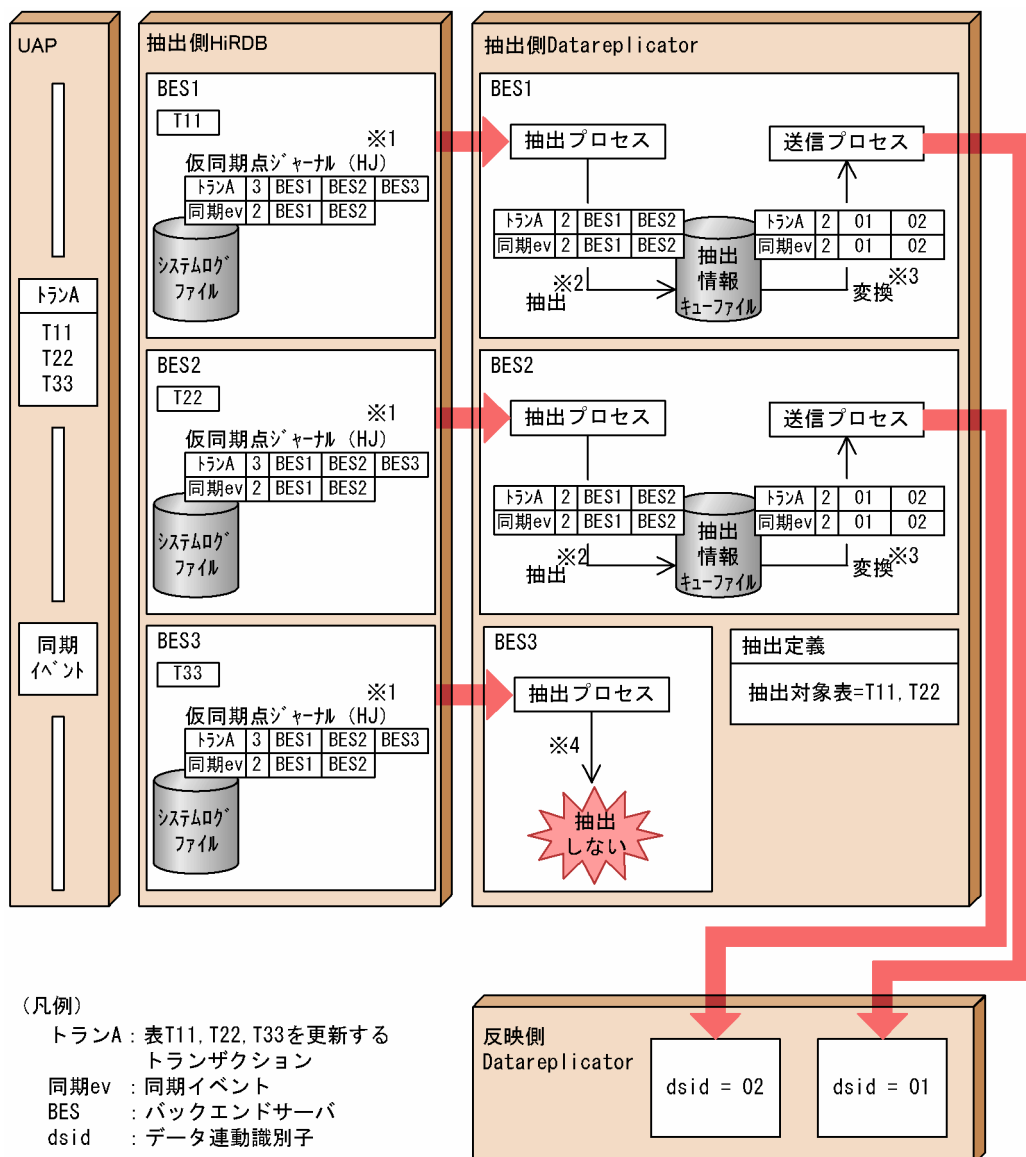
```
done  
echo "Stop event of synchronization."  
exit 1
```

3.6.4 トランザクションブランチ情報

トランザクションブランチ情報とは、レプリケーション対象のバックエンドサーバ名に関する情報のことです。反映トランザクション同期機能を使用する場合、抽出側 HiRDB のシステムログファイルにトランザクションブランチ情報が格納されます。抽出側 Datareplicator はその情報を抽出して、データの整合性を保証しています。

トランザクションブランチ情報の連動を次の図に示します。

図 3-42 トランザクションブランチ情報の連動



注※1

抽出側 HiRDB に対してトランザクションを実行すると、HiRDB のシステムログファイルに、トランザクション単位にトランザクションブランチ情報が出力されます。なお、この時点でのトランザクションブランチ情報には、該当するトランザクションの分岐先バックエンドサーバ名がすべて格納されています。

注※2

システムログファイル中にトランザクションブランチ情報が出力されている場合、抽出側 Datareplicator の抽出プロセスが無条件にトランザクションブランチ情報を抽出して、抽出情報キューファイルに格納します。このとき、抽出対象表以外の表を更新するバックエンドサーバ名の情報は削除されます。

注※3

抽出側 Datareplicator の送信プロセスがトランザクションブランチ情報内のバックエンドサーバ名をデータ連動識別子に変換して、反映側 Datareplicator に送信します。

注※4

抽出対象表以外の表を更新するバックエンドサーバの場合、トランザクションブランチ情報を抽出しません。

なお、HiRDB の正常停止ログは、同期イベントに置換して送信されます。これによって、HiRDB の正常停止を検知した場合でも反映トランザクションを同期します。ただし、同期イベントを抽出するタイミングは HiRDB が正常起動したときです。HiRDB が正常停止した時点では、反映トランザクションの同期は行われないため、注意してください。

3.6.5 同期反映グループ

反映トランザクション同期機能を使用する場合、抽出側 Datareplicator の各バックエンドサーバが送信するトランザクションを一つの反映側 Datareplicator で受信するために、対応する反映側 Datareplicator のデータ連動識別子を同期反映グループとしてまとめておく必要があります。

(1) 同期反映グループの起動

同期反映グループを起動するには、必要なオペランドを正しく指定して、次のどちらかのコマンドを入力します。

1. `hdsstart`
2. `hdsrftl -g 同期反映グループ名 -m start`

なお、2.は同期管理プロセスが停止中の場合にだけ実行できます。同期管理プロセスが稼働中の場合に実行すると、KFRB00161-E メッセージが出力されて、コマンドは無視されます。プロセスの稼働状況は、`hdsstate` コマンドによって確認できます。

同期反映グループを解除した状態で起動するには、次のコマンドを入力します。

hdsstart -c 同期反映グループ名

一度このコマンドを入力すると、次回以降 1.又は 2.のコマンドを入力しても、同期反映グループを解除した状態で起動します。再度同期反映グループを有効にするためには、反映側 Datareplicator を初期化する必要があります。

注意事項

同期反映グループを構成するデータ連動識別子単位での起動はできません。同期反映グループを構成するデータ連動識別子に対して hdsrftcl -d コマンドを実行すると、KFRB03304-E メッセージが出力されて、コマンドは無視されます。

(2) 同期反映グループの停止

同期反映グループを停止するには、次のどちらかのコマンドを入力します。

- hdsstop*
- hdsrftcl -g 同期反映グループ名 -m immediate

注※

指定できるオプションについては、[hdsstop](#) コマンドを参照してください。

コマンドの指定ごとの停止方法を次の表に示します。

表 3-8 同期反映グループの停止方法

コマンドの指定	停止方法
hdsstop	受信プロセスが停止していて、かつ反映情報キューファイルの終端を検知している。反映情報キューファイルの終端検知時にトランザクション実行中であった場合は、そのトランザクションをロールバックして終了する。
hdsstop -t event	
hdsstop -q 待ち時間	
hdsstop -t immediate	コマンドを受け付けた後の、最初の同期イベントでの同期完了、又は反映情報キューファイルの終端を検知している。反映情報キューファイルの終端検知時にトランザクション実行中であった場合は、そのトランザクションをロールバックして終了する。
hdsrftcl -g 同期反映グループ名 -m immediate	
hdsstop -t force	コマンドを受け付けた時点で、実行中の反映トランザクションをロールバックして終了する。反映トランザクションが発生していない場合はそのまま終了する。

注意事項

同期反映グループを構成するデータ連動識別子単位での停止はできません。同期反映グループを構成するデータ連動識別子に対して `hdsrctl -d` コマンドを実行すると、KFRB03304-E メッセージが出力されて、コマンドは無視されます。

(3) 同期反映グループに関する留意点

同期反映グループに関する留意点を次に示します。

(a) オペランドの指定に関する留意点

同期反映グループに指定したデータ連動識別子では、次の表に示すオペランドの指定に制限があります。

表 3-9 同期反映グループに指定したデータ連動識別子の制限

定義ファイル名	オペランド	備考
反映システム定義	<code>commit_wait_time</code>	無視されます。
	<code>commitment_method</code>	<code>fxa_sqlc</code> を指定してください。 <code>fxa_sqlc</code> 以外を指定すると、KFRB00847-E メッセージが出力されてエラーになります。
	<code>discintvl</code>	無視されます。
反映環境定義	<code>eventtrn</code>	無視されます。
	<code>eventtbl</code>	無視されます。
	<code>eventretm</code>	無視されます。
	<code>eventretbl</code>	無視されます。
	<code>eventspd</code>	無視されます。
	<code>eventcntreset</code>	無視されます。
	<code>cmtintvl</code>	無視されます。
	<code>trncmtintvl</code>	無視されます。
	<code>tblcmtintvl</code>	無視されます。
	<code>reflect_trn_max_sqlnum</code>	無視されます。
	<code>startmode</code> *	<code>trn</code> を指定してください。 <code>trn</code> 以外を指定すると、KFRB00847-E メッセージが出力されてエラーになります。
	<code>ujcodekind</code>	<code>rcv</code> を指定してください。 <code>rcv</code> 以外を指定すると、KFRB00847-E メッセージが出力されてエラーになります。

定義ファイル名	オペランド	備考
反映定義	load 文	by 'uoc 名'を指定できません。by 'uoc 名'を指定すると、KFRB03315-E メッセージが出力されてエラーになります。

注※

同期反映グループに指定しているデータ連動識別子の反映方式は、startmode オペランドで指定します。イベント機能や hdsrfctl コマンドでの変更はできません。

なお、同期反映グループを解除して起動すると、無視されていたパラメタも有効になるため、指定値は検討しておく必要があります。

(b) イベント機能に関する留意点

同期反映グループに対するイベントのうち、同期イベント以外は機能しなくなります。同期イベント以外を使用しても、KFRB03316-W メッセージの出力後、無視されます。

接続単位終了イベントについても、-1 イベントとしては扱われなくて、無視されます。

(c) 同期反映グループ内のエラーに関する留意点

同期反映グループを構成するデータ連動識別子の一部でエラーが発生した場合、同期反映グループを構成するすべてのデータ連動識別子がエラー停止します。

同期反映グループ内でエラーが発生すると、同期管理プロセスが停止して、同期管理プロセスの停止を検知した各反映プロセスが KFRB03311-E メッセージを出力します。

同期管理プロセスの停止を検知するタイミングは次のどちらかです。

- SQL 発行時、又は発行前
- 反映情報キューファイルの終端検知後、反映環境定義の ref_wait_interval オペランドの指定値経過時

(d) hdsstate コマンドで表示されるトランザクションの件数に関する留意点

hdsstate コマンドによってデータ連動識別子ごとに表示されるトランザクションの件数は、トランザクションブランチの件数です。抽出側で発生した、実際のグローバルトランザクションの件数とは一致しません。

3.6.6 抽出状態・反映状態の確認方法

反映トランザクション同期機能を使用する場合、抽出状態及び反映状態の確認方法が変わります。次の表に確認方法を示します。

表 3-10 抽出状態及び反映状態の確認方法

運用	この機能を使用しない場合	この機能を使用する場合
抽出元 DB の更新内容がどこまで反映されているか確認したい。	業務の切れ目で hdevent コマンドを実行し、反映側システムに出力される KFRB03211-I メッセージで最終業務まで反映済みであることを確認してください。	「3.6.9 運用例」に説明する運用をしてください。
システムログからすべてのログが抽出済みか確認したい。	pdlogsync -d sys コマンドの実行後、pdls -d rpl -j コマンドを実行し、(2)より(1)の方が大きいことを確認してください。 SYSTEMID : HRD1(183346) Data replication : Y UNITID : unt1(183346) Data replication : Y SERVER NAME : sds01 Extract Database : Y Extract Status : C System Log Extract Point : Run ID Group Gen No. BLock No. 41418a08 log5 2 4e …(1) System Log Sync Info : Run ID Group Gen No. BLock No. 41418a08 log5 2 4c …(2)	この機能を使用しない場合と同様です。
抽出情報キューファイルからすべてのログを送信した後で、抽出側 Datareplicator を停止したい。	抽出対象表の更新ができない状態にした後で hdestate コマンドを入力し、各バックエンドサーバの read position と write position が一致していることを確認してから、hdestop コマンドを実行してください。	抽出対象表の更新ができない状態にした後で同期イベントを実行し、各バックエンドサーバの read position と write position が一致していることを確認してから、hdestop コマンドを実行してください。
反映情報キューファイルからすべてのログを反映した後で、反映側 Datareplicator を停止したい。	抽出側 Datareplicator が停止している状態で hdsstate コマンドを入力し、各データ連動識別子の read position と write position が一致していることを確認してから、hdsstop コマンドを実行してください。	1. 抽出対象表の更新ができない状態にした後で同期イベントを実行し、各バックエンドサーバの read position と write position が一致していることを確認してから、hdestop コマンドで抽出側 Datareplicator を停止します。 2. 1.の状態 hdsstop コマンドを実行してください。

3.6.7 障害運用

反映トランザクション同期機能使用時に想定される障害とその対応方法を次の表に示します。

表 3-11 反映トランザクション同期機能使用時の障害運用

障害内容	対応策
hdevent コマンドの実行漏れのため同期が取れないで、抽出情報キューファイル及び反映情報キューファイルが満杯になる。	同期反映グループを解除して対応してください。
長大なトランザクションなどが原因で、抽出情報キューファイルが満杯になる。	データ連動回復機能で回復してください。
ディスク障害などのため、一部のデータ連動識別子の更新情報や同期イベントが失われる。	データ連動回復機能で回復してください。
通信障害のため、一部のバックエンドサーバの同期イベントが受け取れない。	通信障害の回復を待ち、続きを受信してください。
反映情報キューファイルが満杯で、同期イベントが受け取れない。	同期反映グループを解除して反映側システムを再起動し、反映情報キューファイルに空きを作ります。すべてのログの反映が完了した時点で、再度同期反映グループを構成してください。
反映側 DB のリソース不足や、タイムアウトのため、反映できない。	次のどちらかの方法で対策します。 <ul style="list-style-type: none"> DB の設定を変更した後、再実行してください。 同期反映グループを解除して反映側システムを再起動し、反映情報キューファイルに空きを作ります。すべてのログの反映が完了した時点で、再度同期反映グループを構成してください。
一部のデータ連動識別子の反映実行エラー（SQL エラーなど）のため、同期が取れない。	次のどちらかの方法で対策します。 <ul style="list-style-type: none"> エラー原因を取り除いて、再実行してください。 反映環境定義の skip_sqlcode オペランド又は反映抑止を使用して、エラーとなっている更新情報をスキップしてください。
同期イベントが失われたり、シーケンスがずれたりする。	同期反映グループを解除して対応してください。

注意事項

抽出側システムで障害回復機能を使用した場合、トランザクションブランチ情報が抽出されないため、反映トランザクション同期機能は使用できません。障害回復するには、同期反映グループの定義を削除するか、又は同期反映グループを解除した状態で起動してください。

3.6.8 注意事項

反映トランザクション同期機能使用時の注意事項について説明します。

(1) UOC に関する制限

反映トランザクション同期機能では、反映情報編集 UOC は使用できません。

(2) データ連動識別子の部分初期化

同期反映グループを構成するデータ連動識別子ごとの部分初期化はできません。同期反映グループを構成するデータ連動識別子の部分初期化を実行した場合、KFRB04344-E メッセージが出力されて、初期化処理を中断します。ただし、次のどちらかの条件に該当する場合は、同期反映グループ全体を初期化します。

- 同期反映グループを構成するデータ連動識別子が一つである。
- 部分初期化を指定したデータ連動識別子以外にデータ連動識別子の定義がない、又は、ほかのデータ連動識別子がすべて欠番である。

例えば、次の場合、KFRB04344-E メッセージは出力されません。反映側 Datareplicator の初期化として動作します。

コマンドの指定

dsid003 の初期化を指定して反映側 Datareplicator を開始する。

```
hdsstart -i -D 003
```

反映システム定義の内容

dsid003 以外のデータ連動識別子が欠番となっている。

```
dsid001 = **  
dsid002 = **  
dsid003 = c1  
dsid004 = **  
dsid005 = **  
syncgroup001 = Grp001, c1
```

(3) トランザクションの長大化

反映トランザクション同期機能の処理中に、各バックエンドサーバの同期イベントがそろわないと、トランザクションが長大化して、反映側 HiRDB でリソースが不足する可能性があります。また、長大なトランザクションが最終的に決着できないでロールバックする場合、多量のロールバックログを出力します。

このような問題を回避するために、反映側 HiRDB の排他制御用プールサイズ、及びシステムログファイルサイズを超えないように、反映システム定義の syncwait_limit_tran_count オペランドで同期待ち限界トランザクション数を見積もってください。

(4) 全行削除 (PURGE TABLE) の実行

抽出側 HiRDB がパラレルサーバで、サーバ間横分割していない表の場合は、PURGE TABLE を実行してもデータ連動ができます。

注意

PURGE TABLE を実行すると、反映トランザクションは自動的に COMMIT されます。そのため、抽出側 DB に更新トランザクションが実行されていない状態で、次の運用をしなければなりません。

1. 同期イベントを実行して同期を取り、ほかのトランザクションが実行されていない状態にします。
2. PURGE TABLE を実行します。PURGE TABLE の実行条件は次のとおりです。
 - ・実行するトランザクション中に、PURGE TABLE は一つだけである。
 - ・PURGE TABLE よりも前に、INSERT, UPDATE, 及び DELETE を含んでいない。
3. 再度、同期イベントを実行します。

これ以外の運用をした場合、PURGE TABLE を実行したデータ連動識別子の反映トランザクションだけが COMMIT され、他データ連動識別子との反映トランザクションの同期は保証できなくなります。その場合、速やかに同期イベントを実行して、反映トランザクションを同期する必要があります。

(5) 抽出定義の変更

抽出定義を変更するためには、反映側 Datareplicator を初期化する必要があります。反映側 Datareplicator の初期化手順は、「6.8.4 反映トランザクション同期機能を使用している場合の構成変更」を参照してください。

反映側 Datareplicator を初期化しないで抽出定義を変更すると、反映側 Datareplicator の実行中に KFRB03317-E メッセージが出力されて、エラーになります。エラーを回避するためには、同期反映グループを解除して再起動する必要があります。

(6) 反映側 Datareplicator 実行環境下の HiRDB のクライアント環境定義

反映側 Datareplicator の環境変数 HDSCLTWAITTIME に「USER」を指定した場合、HiRDB のクライアント環境定義 PDSWAITTIME, 及び PDSWATCHTIME には 0 を指定してください。0 以外を指定した場合、反映 SQL プロセスが接続タイムアウトとなることがあります。

3.6.9 運用例

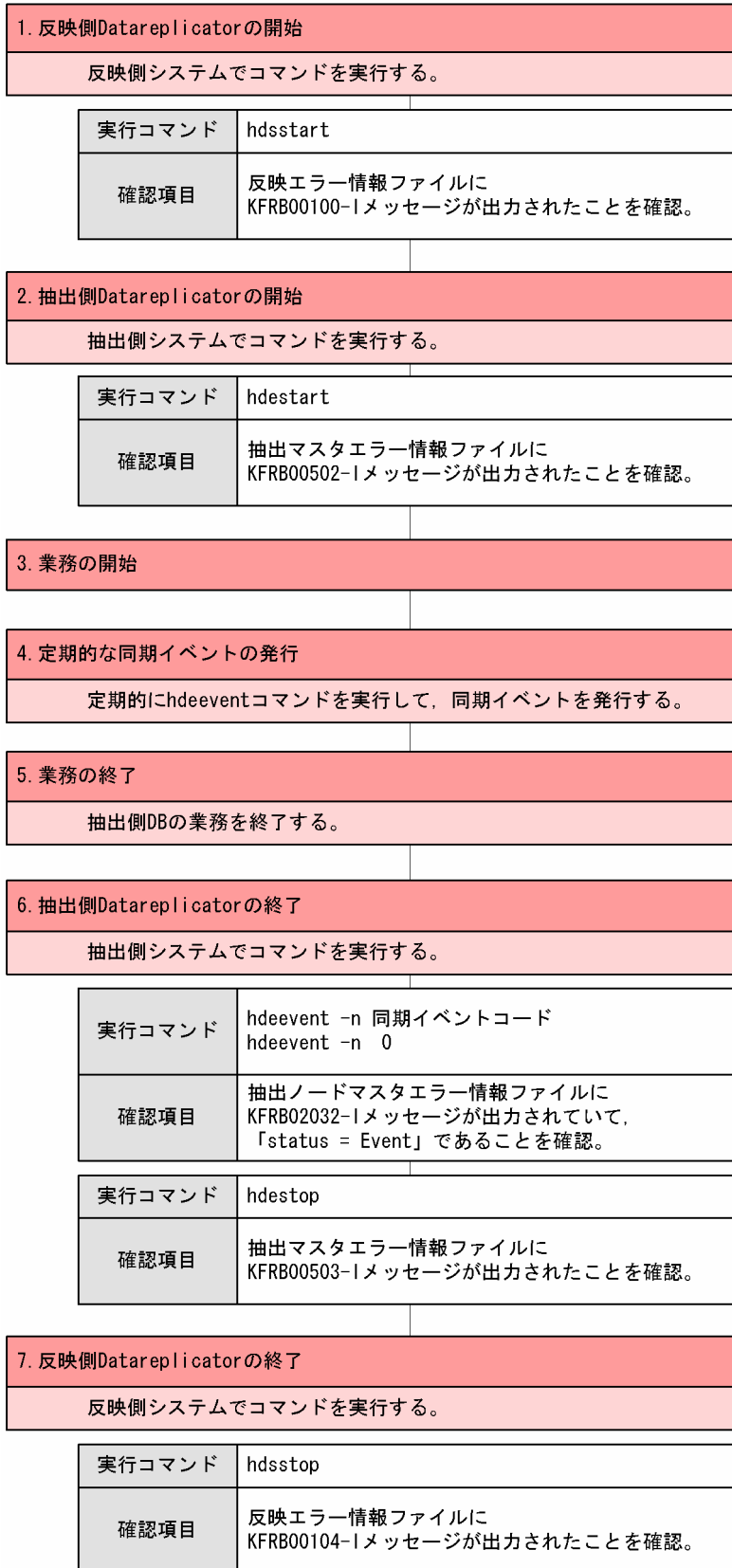
反映トランザクション同期機能を使用する際の運用例について説明します。

運用ごとの手順を次に示します。

運用	手順
環境構築時の初期化手順	「6.2 環境構築時の初期化手順」の手順に従ってください。
通常業務	「通常業務の開始から終了まで」の手順に従ってください。
<ul style="list-style-type: none"> ・抽出対象表の構成変更 ・抽出側 HiRDB のサーバ構成変更 	「通常業務の開始から終了まで」の手順に従った後、抽出定義及び反映定義を変更してください。その後、「6.8.4 反映トランザクション同期機能を使用している場合の構成変更」の手順に従ってください。
抽出側 Datareplicator の部分初期化	反映トランザクション同期機能を使用している送信先は、部分初期化できません。 「6.8.4 反映トランザクション同期機能を使用している場合の構成変更」の手順に従ってください。

通常業務の開始から終了まで

通常業務の開始から終了までの運用の流れを次の図に示します。

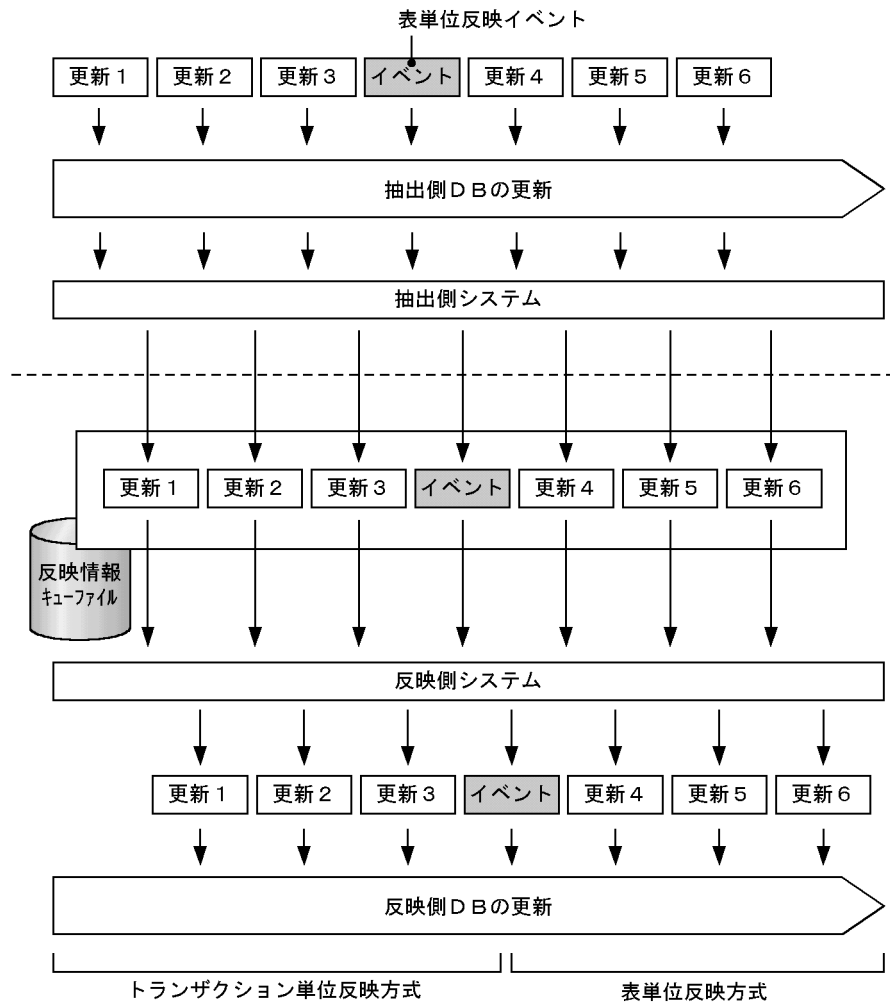


3.7 イベント機能

イベント機能とは、抽出側システムで発行したイベントに従って、反映処理の動作を切り替える機能です。イベント機能を使用すると、反映処理の停止、反映処理の再開始、及び稼働中の反映処理の反映方式の切り替えができます。

イベント機能の動作例を次の図に示します。

図 3-43 イベント機能の動作例



注 更新 1～更新 3 はトランザクション単位反映方式で反映しています。
更新 4 の前に抽出側システムで表単位反映イベントを発行すると、更新 4～更新 6 は表単位反映方式で反映されます。

3.7.1 イベントの発行

抽出側 Datareplicator で hdevent コマンドを実行すると、イベントを発行します。hdevent コマンドには、送信環境定義又は反映環境定義で指定したイベントコードを指定します。イベントコードの定義については、「3.7.3 イベントコードの定義」を参照してください。

なお、イベント機能を使う場合は、抽出側 HiRDB にイベント制御用テーブルを作成してください。イベント制御用テーブルを UAP で更新すると、UAP からイベントを発行することもできます。イベント制御用テーブルについては、「4.6.7 イベント制御用テーブルの設計」を参照してください。

3.7.2 イベントの種類

イベントの種類を次の表に示します。

表 3-12 イベントの種類

イベント種類	内容
送信データ件数リセットイベント	抽出側 Datareplicator の送信データ件数 (hdestate コマンド実行結果の Send data Transmission count に表示される件数) をリセットするイベントです。
同期イベント	バックエンドサーバ間での反映トランザクションの同期を取って、COMMIT するイベントです。このイベントは、反映トランザクション同期機能を使用する場合にだけ有効です。
トランザクション単位反映イベント	トランザクション単位での反映処理を要求するイベントです。
表単位反映イベント	表単位での反映処理を要求するイベントです。
トランザクション単位反映再起動イベント	停止状態にある反映処理を、トランザクション単位での反映処理方式で再起動するイベントです。
表単位反映再起動イベント	停止状態にある反映処理を、表単位での反映処理方式で再起動するイベントです。
停止イベント	反映処理の停止を要求するイベントです。
反映処理数リセットイベント	実行した反映処理の件数 (hdsstate コマンド実行結果の Reflection count に表示する件数) をリセットするイベントです。
接続単位終了イベント	抽出側システムからの更新情報の転送が終了したときに、抽出側システムから送信されるイベントです。この接続単位終了イベントのイベントコードは -1 (固定) です (このイベントコードは反映環境定義では指定できません)。

3.7.3 イベントコードの定義

それぞれのイベントは、イベントコードで識別されます。イベントコードを定義するオペランドを次の表に示します。

表 3-13 イベントコードを定義するオペランド

イベントの種類	イベントコード (既定値)	オペランド名	定義の種類
送信データ件数リセットイベント	-	eventcntreset	送信環境定義

イベントの種類	イベントコード (既定値)	オペランド名	定義の種類
同期イベント	—	eventsync	
トランザクション単位反映イベント	1	eventtrn	反映環境定義
表単位反映イベント	2	eventtbl	
トランザクション単位反映再起動イベント	3	eventretrn	
表単位反映再起動イベント	4	eventretbl	
停止イベント	5	eventspd	
反映処理数リセットイベント	—	eventcntreset	
接続単位終了イベント	−1 (固定)	(指定できません)	

(凡例)

—：イベントコードの規定値はありません。

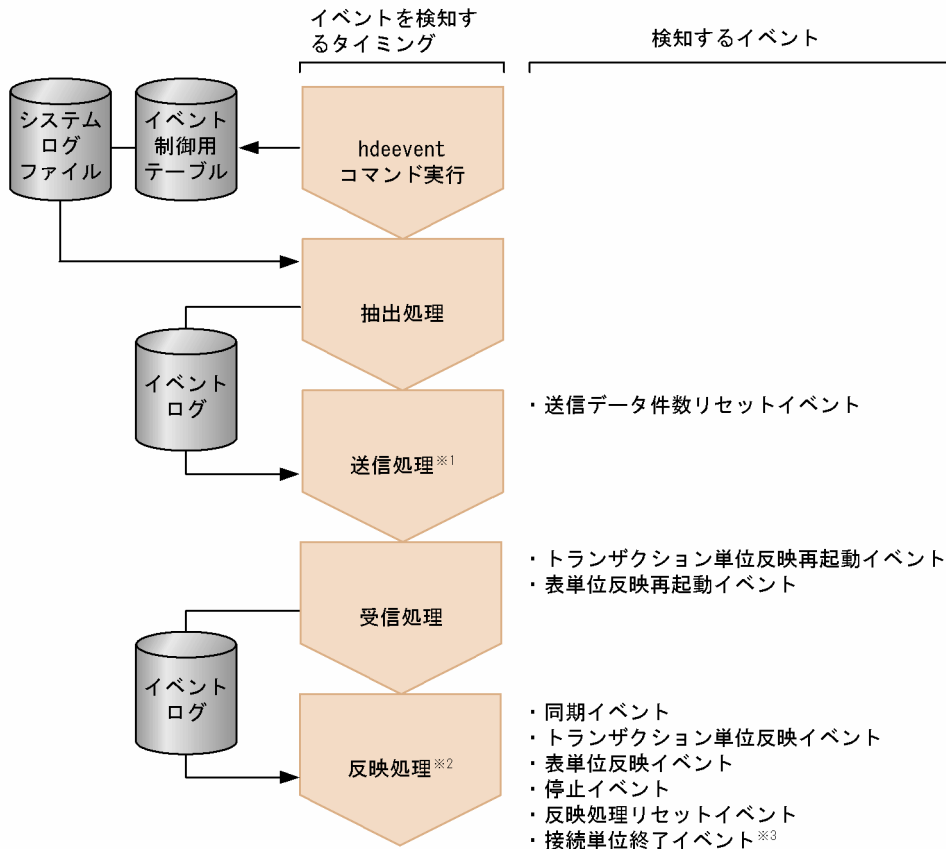
3.7.4 イベントを検知するタイミングと動作

イベントを検知するタイミングと、イベント検知時の抽出機能又は反映機能の動作について説明します。

(1) イベントを検知するタイミング

イベントを検知し、反映処理の動作を切り替えるタイミングを次の図に示します。

図 3-44 イベントを検知するタイミング



注※1

KFRB02036-I メッセージが出力されます。

注※2

KFRB03006-I 又は KFRB03211-I メッセージが出力されます。

注※3

hdsstop コマンドの実行によって検知されます。

(2) イベントを検知したときの動作

イベント検知時、イベントの種類によって、抽出機能又は反映機能の対応するプロセスが動作します。イベント検知時の抽出機能又は反映機能の動作を次の表に示します。

表 3-14 イベント検知時の抽出機能又は反映機能の動作

イベント種別	プロセス種別					
	送信プロセス	反映マスタプロセス	受信プロセス	反映定義サーバプロセス	反映プロセス	同期管理プロセス
送信データ件数リセットイベント	送信データ件数をリセットし	—	—	—	—	—

イベント種別	プロセス種別					
	送信プロセス	反映マスタプロセス	受信プロセス	反映定義サーバプロセス	反映プロセス	同期管理プロセス
	て、続行します。					
同期イベント	—	—	—	—	同期反映グループに属するすべての反映プロセスで、一斉にCOMMITします。	同期反映グループに属する反映プロセス間の同期を取ります。
トランザクション単位反映イベント	—	—	—	反映プロセスをtrnモードで再起動します。	メッセージを出力して、プロセスを終了します(終了モード: イベント終了)。	—
表単位反映イベント	—	—	—	反映プロセスをtblモードで再起動します。	メッセージを出力して、プロセスを終了します(終了モード: イベント終了)。	—
トランザクション単位反映再起動イベント	—	反映定義サーバプロセスを再起動します。ただし、既に起動している場合は、エラーメッセージを出力します。	反映マスタプロセスに通知します。	反映プロセスをtrnモードで起動します。その後、すべての反映プロセスを、このイベントコードを検知して起動した後で、メッセージを出力して処理を続行します。	メッセージを出力して、処理を続行します。	—
表単位反映再起動イベント	—	反映定義サーバプロセスを再起動します。ただし、既に起動している場合は、エラーメッセージを出力します。	反映マスタプロセスに通知します。	反映プロセスをtblモードで起動します。その後、すべての反映プロセスを、このイベントコードを検知して起動した後で、メッセージを出力して処理を続行します。	メッセージを出力して、処理を続行します。	—
停止イベント	—	—	—	すべての反映プロセスのイベント終了を待ちます。	メッセージを出力して、プロセスを終了します。	—

イベント種別	プロセス種別					
	送信プロセス	反映マスタプロセス	受信プロセス	反映定義サーバプロセス	反映プロセス	同期管理プロセス
				す。すべての反映プロセスがイベント終了した後、メッセージを出力してプロセスを終了します。	(終了モード：イベント終了)。	
反映処理数リセットイベント	—	—	—	—	反映処理数をリセットして、処理を続行します。	—
接続単位終了イベント	—	—	—	すべての反映プロセスがこのイベントコードを検知した時点でメッセージを出力し、処理を続行します。	メッセージを出力して、処理を続行します。	—

(凡例)

—：動作しません。

trn モード：トランザクション単位反映方式

tbl モード：表単位反映方式

注

イベントの種別に関係なく、イベント検知時には反映側データベースに対して同期点処理を実行します。

3.7.5 注意事項

イベント機能を使用する場合の注意事項を次に示します。

- 抽出側システムから送られたイベントコードが、反映環境定義に指定したイベントコード、又は-1（接続単位終了イベント）以外の場合、反映側 Datareplicator はイベントが通知されたことを認識してメッセージを出力しますが、イベント通知前の反映処理を継続します（イベントは無視されます）。
- 反映処理が既に動作しているときは、再起動イベント（トランザクション単位反映再起動イベント、表単位反映再起動イベント）を検知しても無視されます。
- 再起動イベント（トランザクション単位反映再起動イベント、表単位反映再起動イベント）を処理する前に Datareplicator を停止すると、再起動イベントの情報は破棄されます。

- 反映定義サーバプロセスのイベント監視間隔は1秒であり、また、同時に管理できるイベントの数は16個までです。このため、1秒間に16個を超えるイベントが発行された場合には、メッセージにイベントコードが表示されません。反映定義サーバプロセスが出力するメッセージのイベントコードを使って運用する場合には、1秒間に16個を超えるイベントを発行しないでください。

3.8 ファイルの二重化

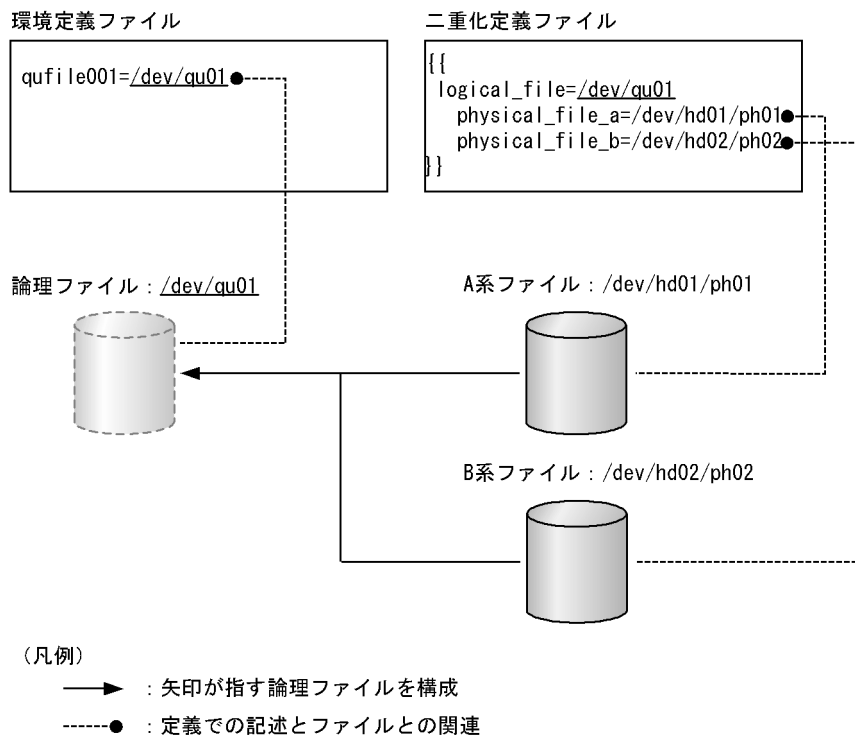
Datareplicator では、データ連動に使用するファイルを二重化することで、ファイル障害時の信頼性を向上することができます。

注意事項

ファイルを二重化すると、二つのファイルに書き込みをするため、データ連動の処理時間が約 2 倍長くなります。データ連動の処理時間を考慮して、ファイルを二重化してください。

ファイルの二重化を行う場合、環境定義で指定するファイルを論理ファイルにします。そして、その論理ファイルを構成する二つの物理ファイルを二重化定義ファイルで定義します。これらのファイルは、一方を A 系ファイル、他方を B 系ファイルと呼びます。A 系ファイルと B 系ファイルの内容は同じです。論理ファイルと A 系 B 系の物理ファイルの関係を次の図に示します。

図 3-45 論理ファイルと A 系 B 系物理ファイルの関係



Datareplicator は、この二つの物理ファイルに対して同じ情報を出力します。そして、どちらかのファイルに障害が発生すると、Datareplicator は障害が発生したファイルを切り離し、残りのファイルを使用してデータ連動を続行します。

3.8.1 二重化できるファイル

二重化できるファイルを次の表に示します。

表 3-15 二重化できるファイル

対象システム	定義ファイル
抽出システム	抽出マスタステータスファイル
	抽出サーバステータスファイル
	抽出情報キューファイル
	データ連動用連絡ファイル
反映システム	反映マスタステータスファイル
	反映ステータスファイル
	反映情報キューファイル

注意事項

- 物理ファイルに割り当てるファイル種別（キャラクタ型スペシャルファイル，又は通常ファイル）は，すべての物理ファイルで統一してください。
- セクタ長はすべての物理ファイルで統一してください。
- データ連動用連絡ファイルに関しては，論理ファイル名と同じ名称の物理ファイル名を A 系， B 系のどちらかに指定してください。
- Datareplicator ファイルシステム領域を使用するファイルは二重化できません。二重化する論理ファイル名をファイルシステム領域に割り当てないでください。

3.8.2 二重化に関連するファイル

二重化に関連するファイルを次の表に示し，それぞれのファイルについて説明します。

表 3-16 二重化に関連するファイル

ファイル	使用目的
抽出システム定義ファイル， 又は反映システム定義ファイル	二重化定義ファイルのファイル名を定義します。
抽出環境定義ファイル， 又は反映環境定義ファイル	二重化するファイルの論理ファイル名を定義します。
二重化定義ファイル	二重化するファイルの物理ファイル名を定義します。
二重化制御ファイル	ファイルの二重化を制御するためのファイルです。Datareplicator が初期化処理時に生成します。

(1) 抽出システム定義ファイル，又は反映システム定義ファイル

二重化定義ファイルのファイル名を指定します。

抽出側システムを対象とするファイルを二重化する場合は抽出システム定義ファイルに、反映側システムを対象とするファイルを二重化する場合は反映システム定義ファイルに file_dupenv オペランドを定義してください。file_dupenv オペランドについては、「5.2 抽出システム定義」又は「5.8 反映システム定義」を参照してください。このオペランドが省略された場合、ファイルの二重化は使用できません。

(2) 抽出環境定義ファイル, 又は反映環境定義ファイル

二重化するファイルを論理ファイル名で定義します。

抽出側システムを対象とするファイルを二重化する場合は抽出環境定義ファイルに、反映側システムを対象とするファイルを二重化する場合は反映環境定義ファイルに論理ファイル名を定義してください。

(3) 二重化定義ファイル

論理ファイル名と物理ファイル名の対応を定義します。定義の詳細については、「5.7 二重化定義 (抽出側)」又は「5.12 二重化定義 (反映側)」を参照してください。

(4) 二重化制御ファイル

ファイルの二重化を制御するためのファイルです。Datareplicator が初期化処理時に次のディレクトリに生成します。

システム	作成ディレクトリ	ファイル名
抽出側	すべての HiRDB サーバマシンの抽出側 Datareplicator 運用ディレクトリ	hde_fileenv.prp
反映側	反映側 Datareplicator 運用ディレクトリ	hds_fileenv.prp

Datareplicator 初期化後は、二重化制御ファイルのバックアップを取得しておいてください。このファイルの障害時には、バックアップを利用して回復する必要があります。

3.8.3 二重化する場合の注意事項

(1) 系切り替え機能を使用している場合

- Datareplicator 初期化後、対応する待機系サーバの Datareplicator の運用ディレクトリに、二重化制御ファイルをコピーしてください。ただし、次の場合はコピーする必要はありません。
 - HiRDB/パラレルサーバで相互系切り替え構成の場合に、待機系のホストに実行系の BES があるとき実行系の BES があるところに自動的に二重化制御ファイルが作成されます。
 - 2 回目以降の系切り替えで、かつ二重化制御ファイルの内容に一切変更がないとき

- 拡張子が「.mf」のファイルは待機系にコピーする必要はありません。拡張子が「.mf」のファイルは、ファイルを二重化する場合に、Datareplicatorが自動生成するワークファイルです。Datareplicatorの稼働中には削除しないでください。

(2) ファイル障害が発生した場合

二重化しているファイルに障害が発生した場合、ファイルの回復作業が必要になります。詳細については、「[9.1.6 ファイルの二重化運用時のファイル障害への対処](#)」を参照してください。

4

システム設計

この章では、業務形態に合わせてデータ連動システムを設計する方法について説明します。さらに、抽出側 Datareplicator の設計方法と反映側 Datareplicator の設計方法についても説明します。

4.1 システム設計の項目

設計する項目と関連する定義とコマンドの一覧を次の表に示します。

表 4-1 設計する項目と関連する定義とコマンドの一覧

設計する項目	説明	関連する定義とコマンド※
連動パターンの設計	どのようなパターンでデータ連動するかを設計します。	抽出側システムの定義 ・抽出定義 反映側システムの定義 ・反映定義
抽出側 DB と反映側 DB との対応の設計	データ連動の対象になる抽出側 DB と反映側 DB のマッピングキー、データ型、文字コードなどの対応を設計します。 反映対象表のデータがない場合は、HiRDB のデータベース作成ユティリティ、HiRDB Dataextractor などで作成する必要があります。	—
抽出側システムと反映側システムとの対応の設計	抽出側システムと反映側システムとの対応数、対応するときに必要になるデータ連動識別子などを設計します。	抽出側システムの定義 ・抽出システム定義 ・抽出環境定義 ・送信環境定義 ・抽出定義 反映側システムの定義 ・反映システム定義 ・反映環境定義
データ連動システムの形態の設計	データ連動システムを単一方向更新システムか双方向更新システムのどちらにするか、及び階層型システムへの応用形態にするかどうかを設計します。	抽出側システムの定義 ・抽出システム定義 ・抽出環境定義 ・送信環境定義 ・抽出定義 反映側システムの定義 ・反映システム定義 ・反映環境定義
抽出側 Datareplicator の設計※	抽出側 Datareplicator で使うファイルの対応や、抽出/送信方法、起動/終了方法などを設計します。	抽出側システムの定義 ・抽出システム定義 ・抽出環境定義 ・送信環境定義 ・コマンド ・hdestart ・hdestop ・hdeevent
反映側 Datareplicator の設計※	反映側 Datareplicator で使うファイルの対応や、反映方法、起動・終了方法などを設計します。	反映側システムの定義 ・反映システム定義

設計する項目	説明	関連する定義とコマンド※
		<ul style="list-style-type: none"> ・反映環境定義 ・コマンド <ul style="list-style-type: none"> ・hdsstart ・hdsstop ・hdsrctl

注※

抽出側/反映側システムがメインフレーム (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) の場合は, XDM/DS に定義します。XDM/DS の定義とコマンドについては, マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」, 及びマニュアル「VOS3 XDM データ連動機能 XDM/DS 運用」を参照してください。

抽出側システムが SAM ファイルを使うメインフレーム側 DB (PDMII E2, RDB1 E2) の場合は, 該当する製品のマニュアルを参照してください。

4.2 連動パターンの設計

Datareplicator のデータ連動には、7 種類のパターンがあります。これらからデータ連動の目的に合ったパターンを選択します。また、これら複数のパターンを組み合わせることもできます。

データ連動のパターンを次の表に示します。

表 4-2 データ連動のパターン

データ連動のパターン	内容
同一形式の表にデータ連動するパターン	抽出対象表を、すべて同一の表に反映します。
	抽出対象表の表名、列名だけを変更して反映します。
抽出側 DB のすべての表を同一形式の表にデータ連動するパターン	抽出側 DB のすべての表を抽出対象とし、全表抽出定義文を使用して、すべて同一の表に反映します。
別形式の表にデータ連動するパターン	抽出対象表の列を抽出側で並べ替えて抽出し、反映します。
	抽出対象表の列を反映側で並べ替えて反映します。
	抽出対象表の一部の列を抽出側で選択して抽出し、反映します。
	抽出対象表の一部の列を反映側で選択し、反映します。
	抽出対象表に、反映側で固定値の列を追加して反映します。
1 表から n 表にデータ連動するパターン	一つの抽出対象表を抽出側で複数に分割して抽出し、一つの反映側で反映します。
	一つの抽出対象表を抽出側で複数に分割して抽出し、複数の反映側で反映します。
	一つの抽出対象表を反映側で複数に分割して反映します。
n 表から 1 表にデータ連動するパターン	複数の抽出対象表を反映側で一つにまとめ、一つの表に反映します。
送信行を選択して反映するパターン	送信行を選択して反映します。
ユーザOWNコーディングを使ってデータ連動するパターン	更新情報を反映側の UOC でユーザ任意に加工して反映します。
更新情報の履歴を時系列順に取得するパターン	更新情報のほかに反映日付、反映時間などを、時系列順に取得します。

以降、データ連動のパターン別に、抽出側システムの抽出定義の内容について説明します。

ここでは Datareplicator の抽出側システムの定義で説明します。

抽出側 DB がメインフレーム (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) の場合には、抽出定義は XDM/DS で指定します。XDM/DS の抽出定義については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

抽出側 DB が SAM ファイルを使うメインフレーム (PDMII E2, RDB1 E2) の場合は、製品ごとに定義に相当する指定をする場合としない場合があります。詳細は、各製品の該当するマニュアルを参照してください。

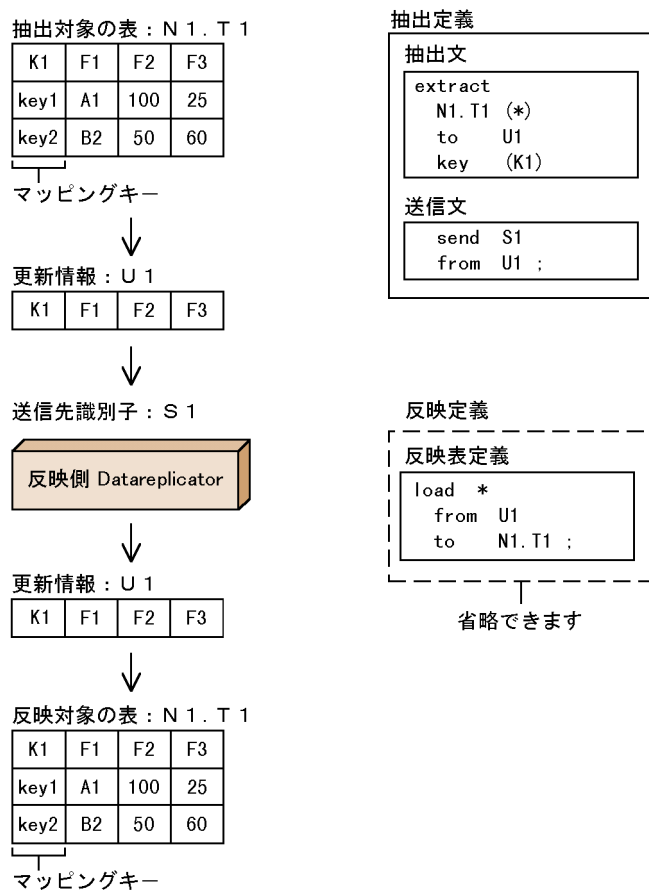
4.2.1 同一形式の表にデータ連動するパターン

同一形式の表にデータ連動するパターンについて説明します。

(1) 抽出対象表を、すべて同一の表に反映する例

抽出対象表を、形式、表名、列名などがすべて同一の表に反映する例を次の図に示します。

図 4-1 すべて同一の表に反映する例



(説明)

- 対象とする表

抽出側では、抽出対象とする表を抽出定義の抽出文で「認可識別子.表識別子」の形式で指定します。
反映側では、反映対象とする表を反映表定義の to 句で「認可識別子.表識別子」の形式で指定します。

- 抽出対象にする列

抽出側では、抽出対象にする列を、抽出文で指定します。すべての列を、並び順を替えないで抽出する場合、「*」で列名の指定を省略できます。

- **更新情報の名称**

一つの表に対する抽出条件を、一つの抽出文で定義します。この抽出条件で抽出された更新情報の名称を抽出文の to 句で定義します。この名称を**更新情報名**といいます。一つの表から複数の抽出条件で抽出する場合は、抽出条件の数だけ、抽出文を指定します。

- **抽出した更新情報の送信先**

抽出定義の送信文で、抽出した更新情報の送信先を指定します。送信文では、送信先ごとに送信する更新情報名を一つ指定します。一つの送信先に対して複数の更新情報名を送信する場合は、送信する更新情報名の数だけ、送信文を指定します。

- **マッピングキー**

抽出／反映の対象とする行は、マッピングキーで特定します。このため、抽出対象表と反映対象表とで、マッピングキーを対応させる必要があります。

マッピングキーは、抽出文の key 句で指定します。

- **反映定義の省略**

同一形式の表で、さらに表名、列名も同じであるため、反映定義を省略できます。反映定義を省略した場合、抽出定義を反映定義と仮定して、データ連動を実行します。ただし、認可識別子、表識別子、及び列名として仮定される値は、抽出側 DB によって異なります。反映定義を省略したときに認可識別子、表識別子、及び列名に仮定される値を次の表に示します。

表 4-3 反映定義を省略したときに認可識別子、表識別子、及び列名に仮定される値

抽出側 DB							反映側 DB
HiRDB	XDM/SD E2*	XDM/RD E2	ADM	PDMII E2	TMS-4V/SP	RDB1 E2	HiRDB
認可識別子	スキーマ名	認可識別子	データベース名	DBM 名	認可識別子	認可識別子	認可識別子
表識別子	最下位のレコード型名	表識別子	セグメント	データセット名	表識別子	表識別子	表識別子
列名	構成要素名	列名	フィールド名	フィールド名	列名	列名	列名

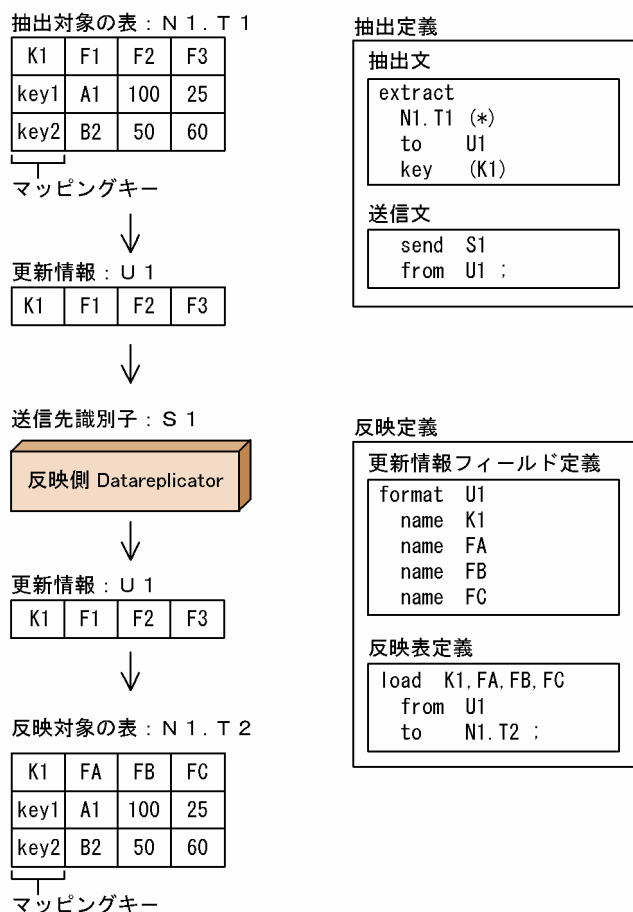
注※

1 個の抽出定義内で、構成要素名と抽出再定義 RESTRUCT 文で指定した再定義項目名がユニークでない場合は、再定義項目名に対応する反映表の定義ができないため、反映定義は省略できません。

(2) 抽出対象表の表名、又は列名を変更して反映する例

抽出対象表の表名、又は列名を変更して反映する例を次の図に示します。

図 4-2 表名, 又は列名を変更して反映する例



(説明)

• 列名の定義の変更

更新情報フィールド定義の name 句で, 変更後の列名を定義します。更新情報フィールド定義の name 句には, 更新情報のすべてのフィールドの名称を定義します。ここで定義したフィールド名称のうち, 反映対象とするフィールドを, 反映表定義の load 文で指定します。ただし, すべてのフィールドの並び順を替えないで反映する場合は, 更新情報フィールド定義を省略できます。

• 表名, 又は列名の変更

反映表定義の load 文に変更後の列名を指定し, to 句で変更後の表名を指定します。

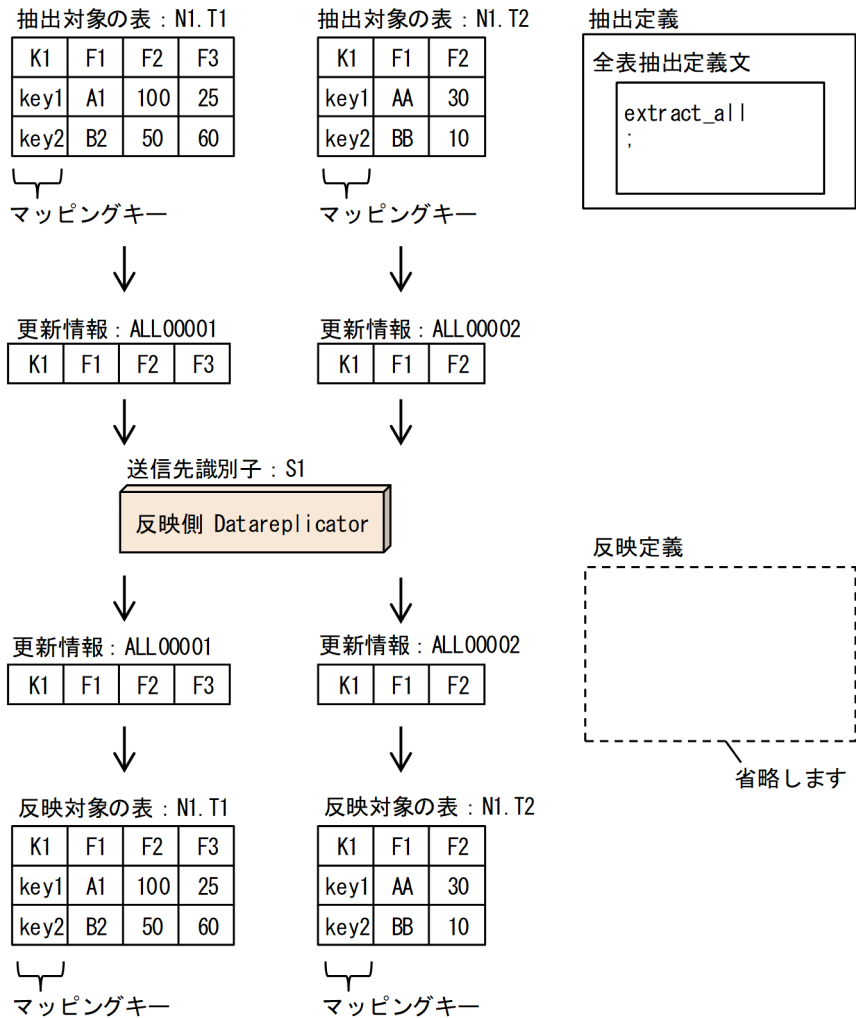
4.2.2 抽出側 DB のすべての表を同一形式の表にデータ連動するパターン

抽出側 DB のすべての表を同一形式の表にデータ連動するパターンの例について説明します。

(1) 抽出側 DB のすべての表を抽出対象とし、全表抽出定義文を使用して、すべて同一の表に反映する例

抽出側 DB のすべての表を抽出対象とし、全表抽出定義文を使用して、形式、表名、列名などがすべて同一の表に反映する例を次の図に示します。

図 4-3 すべての表を抽出対象とし、すべて同一の表に反映する例



(説明)

- 対象とする表の指定
抽出側では、抽出定義の全表抽出定義文を指定します。
反映側では、反映定義を省略します。

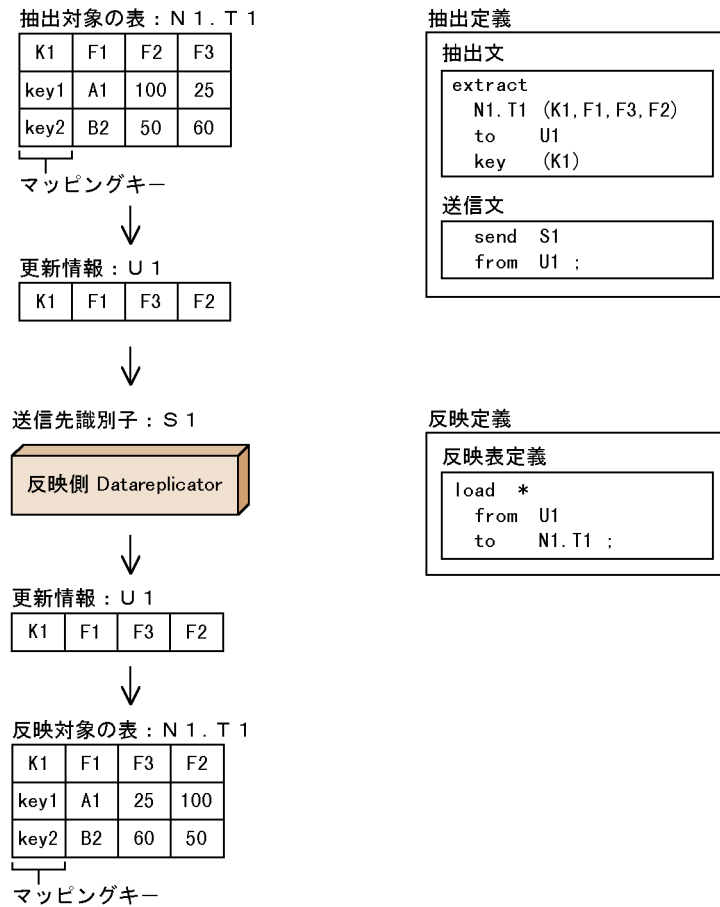
4.2.3 別形式の表にデータ連動するパターン

別形式の表にデータ連動するパターンの例について説明します。

(1) 抽出対象表の列を抽出側で並べ替えて抽出して反映する例

抽出対象表の列を抽出側で並べ替えて抽出して反映する例を次の図に示します。

図 4-4 抽出側で列を並べ替えて抽出して反映する例



(説明)

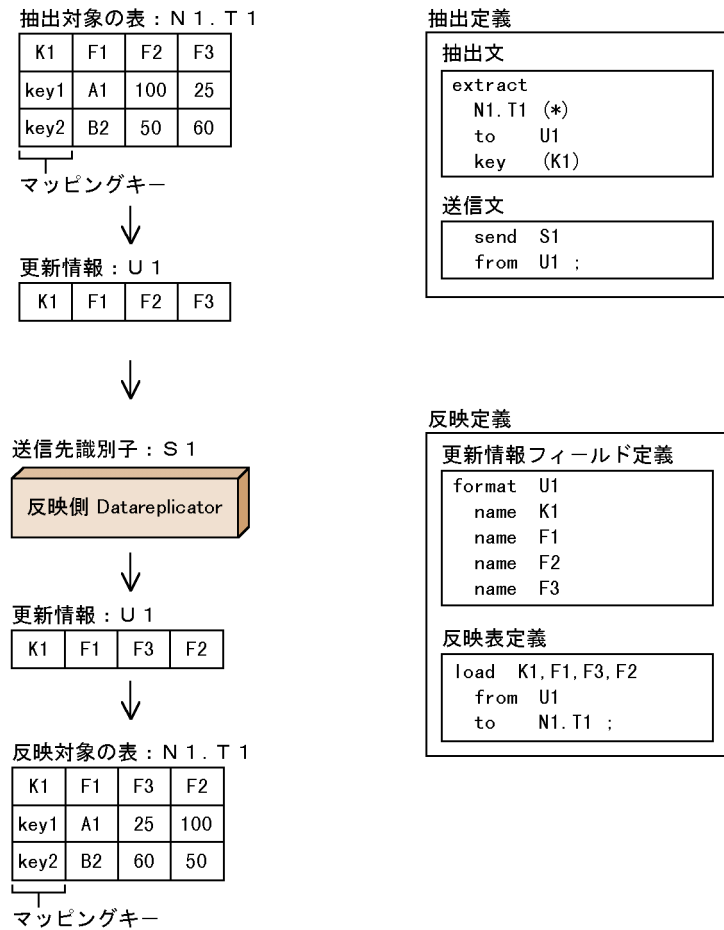
- 抽出側での列の並べ替え

抽出側で列を並べ替える場合には、並べ替える順番に抽出対象表の列名を抽出文で指定します。

(2) 抽出対象表の列を反映側で並べ替えて反映する例

抽出対象表の列を反映側で並べ替えて反映する例を次の図に示します。

図 4-5 反映側で列を並べ替えて反映する例



(説明)

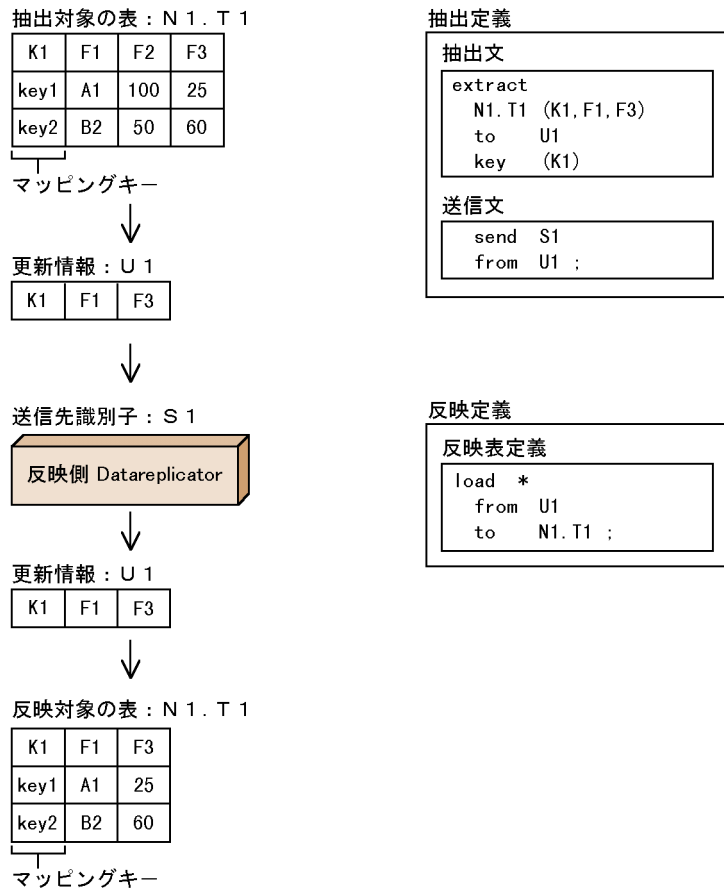
• 反映側での列の並べ替え

反映側で列を並べ替える場合には、まず更新情報フィールド定義の name 句で、更新情報のフィールド名称を定義します。ここで定義したフィールドを、並べ替える順番に反映表定義の load 文で指定します。

(3) 抽出対象表の一部の列を抽出側で選択して抽出して反映する例

抽出対象表の一部の列を抽出側で選択して抽出して反映する例を次の図に示します。

図 4-6 抽出側で列の一部を選択して抽出して反映する例



(説明)

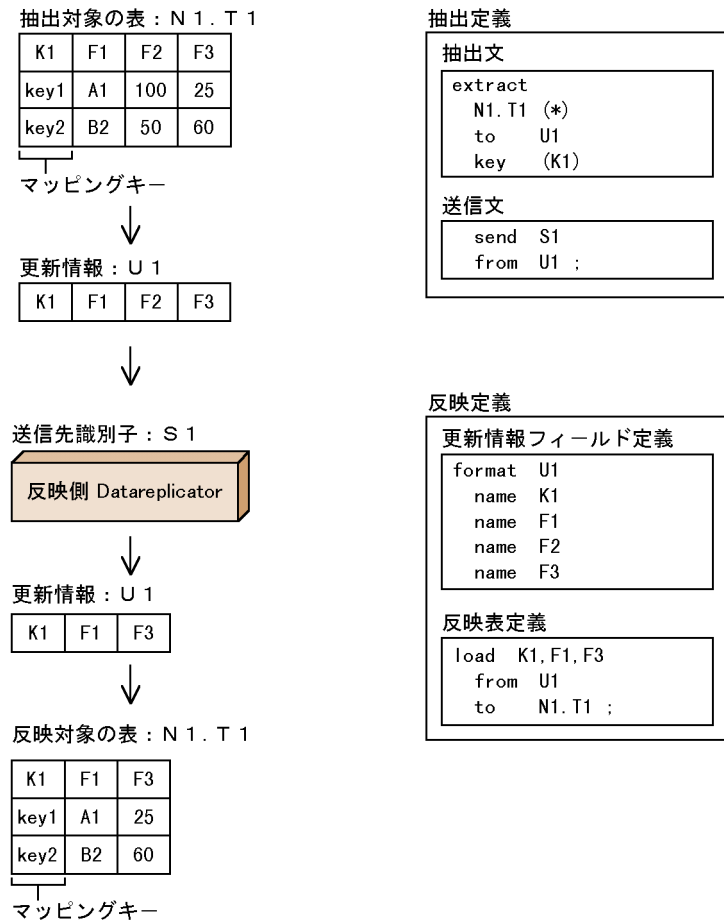
- 抽出側での列の選択

抽出側で一部の列を選択して抽出する場合には、抽出する列だけを抽出文で指定します。

(4) 抽出対象表の一部の列を反映側で選択して反映する例

抽出対象表の一部の列を反映側で選択して反映する例を次の図に示します。

図 4-7 反映側で列の一部を選択して反映する例



(説明)

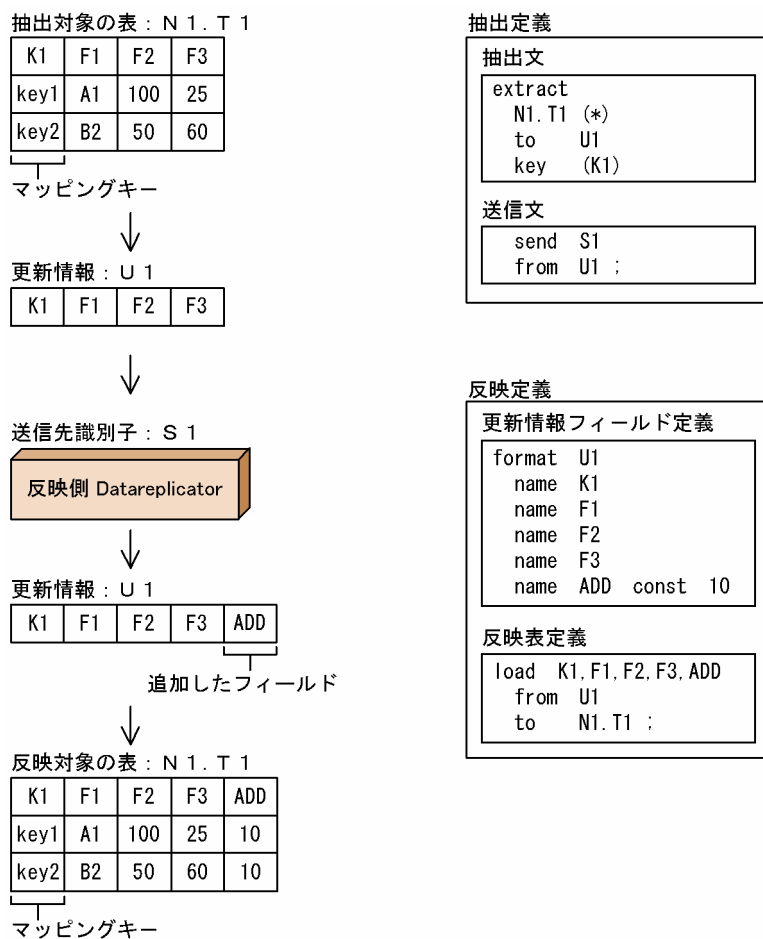
- 反映側での列の選択

反映側で一部の列を選択して反映する場合には、まず更新情報フィールド定義の name 句で、更新情報のフィールド名称を定義します。ここで定義したフィールドのうち、反映するフィールドだけを反映表定義の load 文で指定します。

(5) 抽出対象表に、反映側で固定値の列を追加して反映する例

抽出対象表に、反映側で固定値の列を追加して反映する例を次の図に示します。

図 4-8 反映側で固定値の列を追加して反映する例



(説明)

• 固定値の列を追加する場合の更新情報のフィールド名称

固定値の列を追加する場合には、更新情報フィールド定義でユーザ任意のフィールド名を指定します。さらに指定したフィールド名称の後に、const と追加する固定値を指定します。

また反映表定義の load 文では、ユーザ任意のフィールド名も指定します。

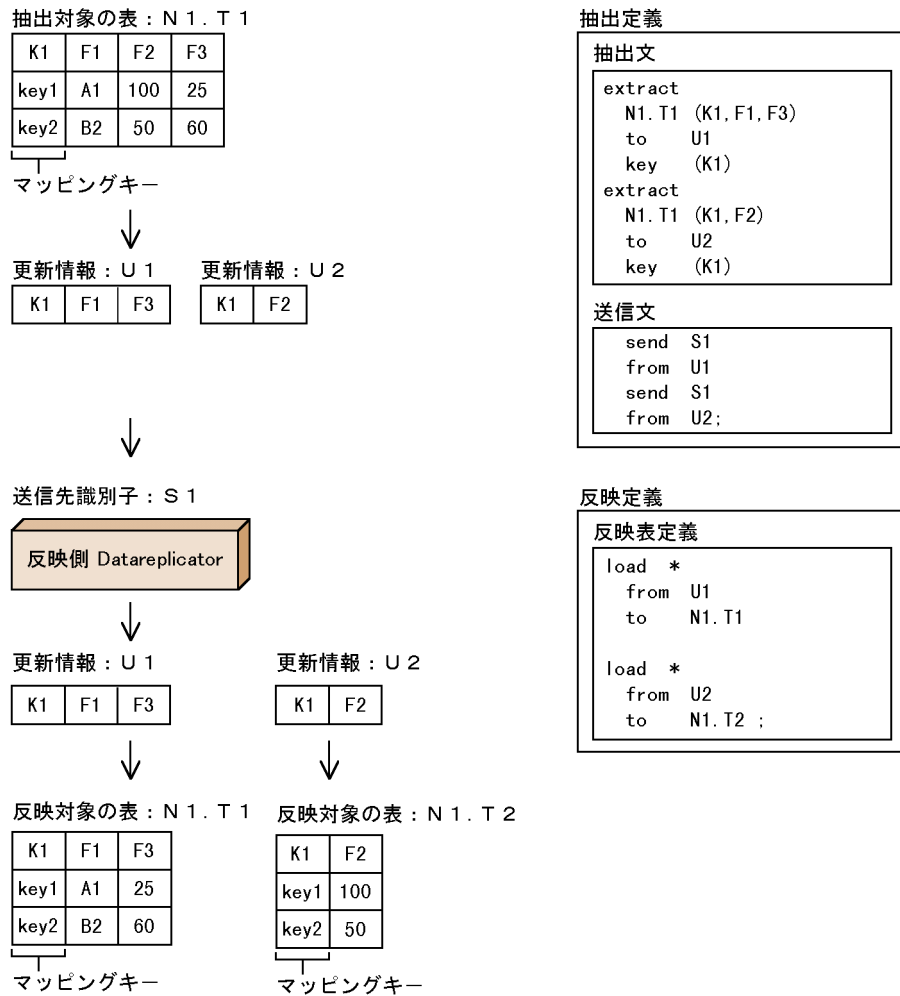
4.2.4 1 表から n 表にデータ連動するパターン

1 表から n 表にデータ連動するパターンについて説明します。

(1) 一つの抽出対象表を抽出側で複数に分割して抽出して一つの反映側で反映する例

一つの抽出対象表を抽出側で複数に分割して抽出して一つの反映側で反映する例を次の図に示します。

図 4-9 抽出側で 1 表を n 表に分割して抽出して一つの反映側システムで反映する例



(説明)

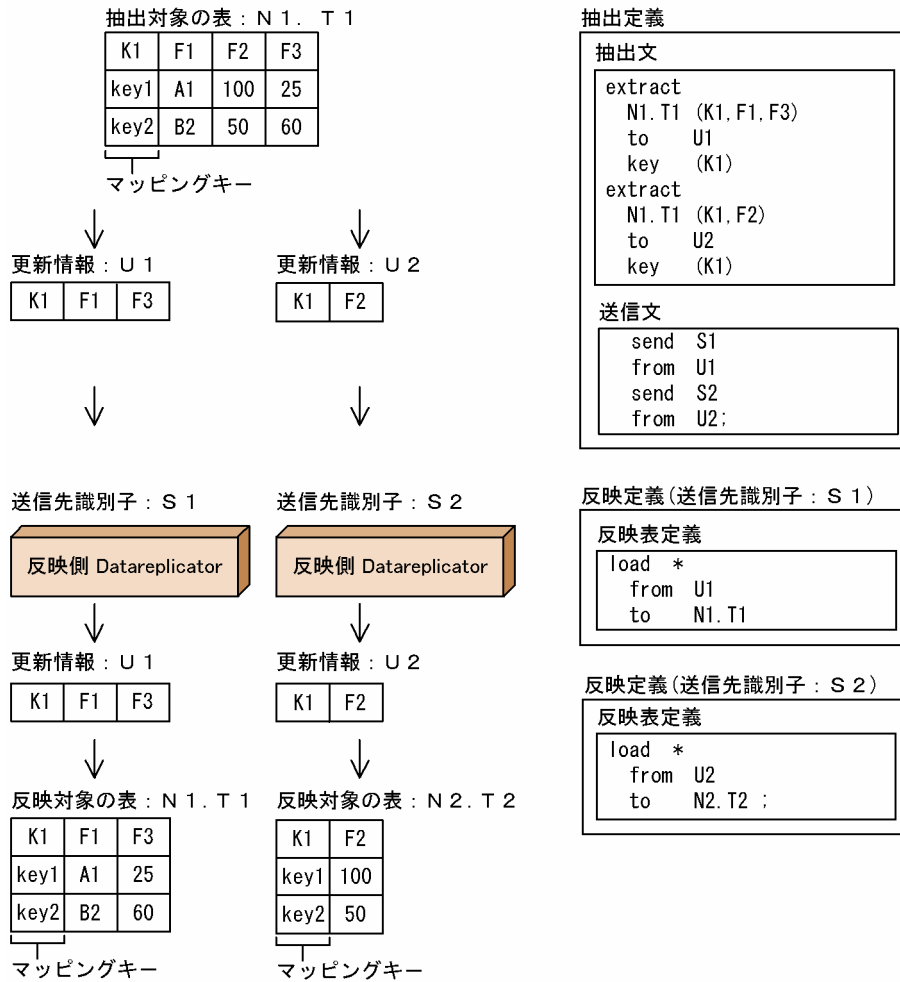
• 抽出側での表の分割

抽出側で 1 表を n 表に分割する場合には、分割する数だけ、抽出文を指定します。

(2) 一つの抽出対象表を抽出側で複数に分割して抽出して複数の反映側で反映する例

一つの抽出対象表を抽出側で複数に分割して抽出して複数の反映側で反映する例を次の図に示します。

図 4-10 抽出側で 1 表を n 表に分割して抽出して複数の反映側で反映する例



(説明)

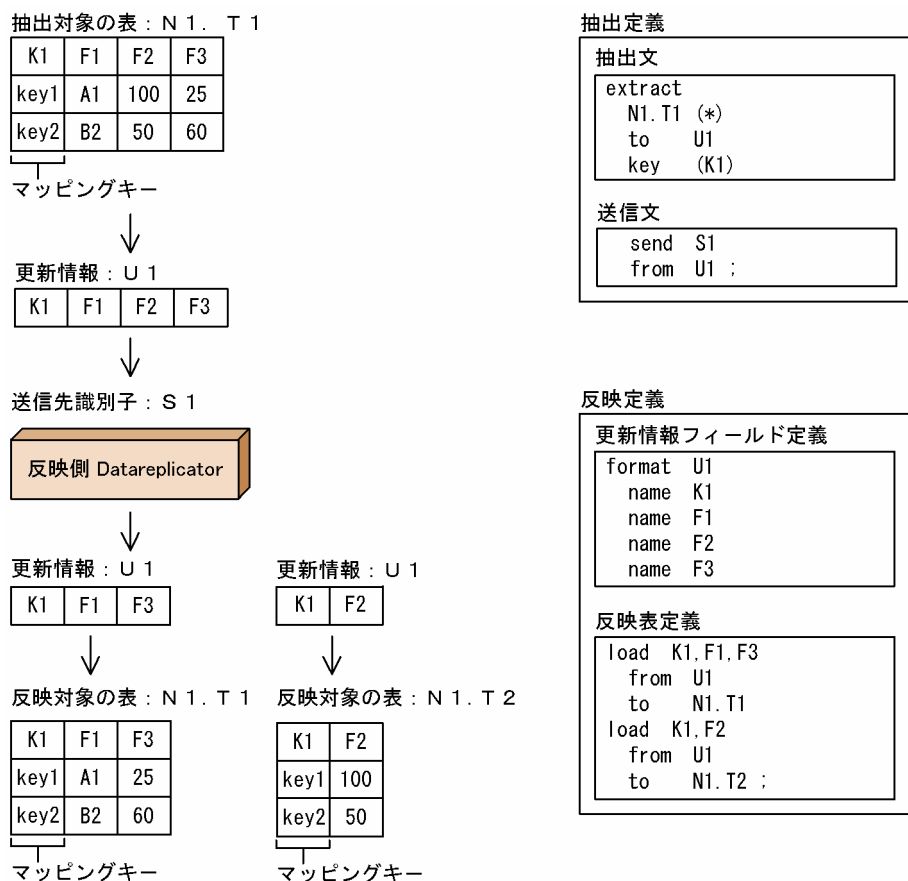
- 複数の反映側への送信

複数の反映側へ送信するには、送信する反映側の数だけ、送信文を指定します。

(3) 一つの抽出対象表を反映側で複数に分割して反映する例

一つの抽出対象表を反映側で複数に分割して反映する例を次の図に示します。

図 4-11 一つの抽出対象表を反映側で複数に分割して反映する例



(説明)

• 反映側での表の分割

反映側で一つの抽出対象表を複数に分割する場合には、分割する数だけ、反映表定義を指定します。抽出対象表と反映対象表は、マッピングキーで対応する必要があるため、すべての反映表定義でマッピングキーになるフィールドを指定します。

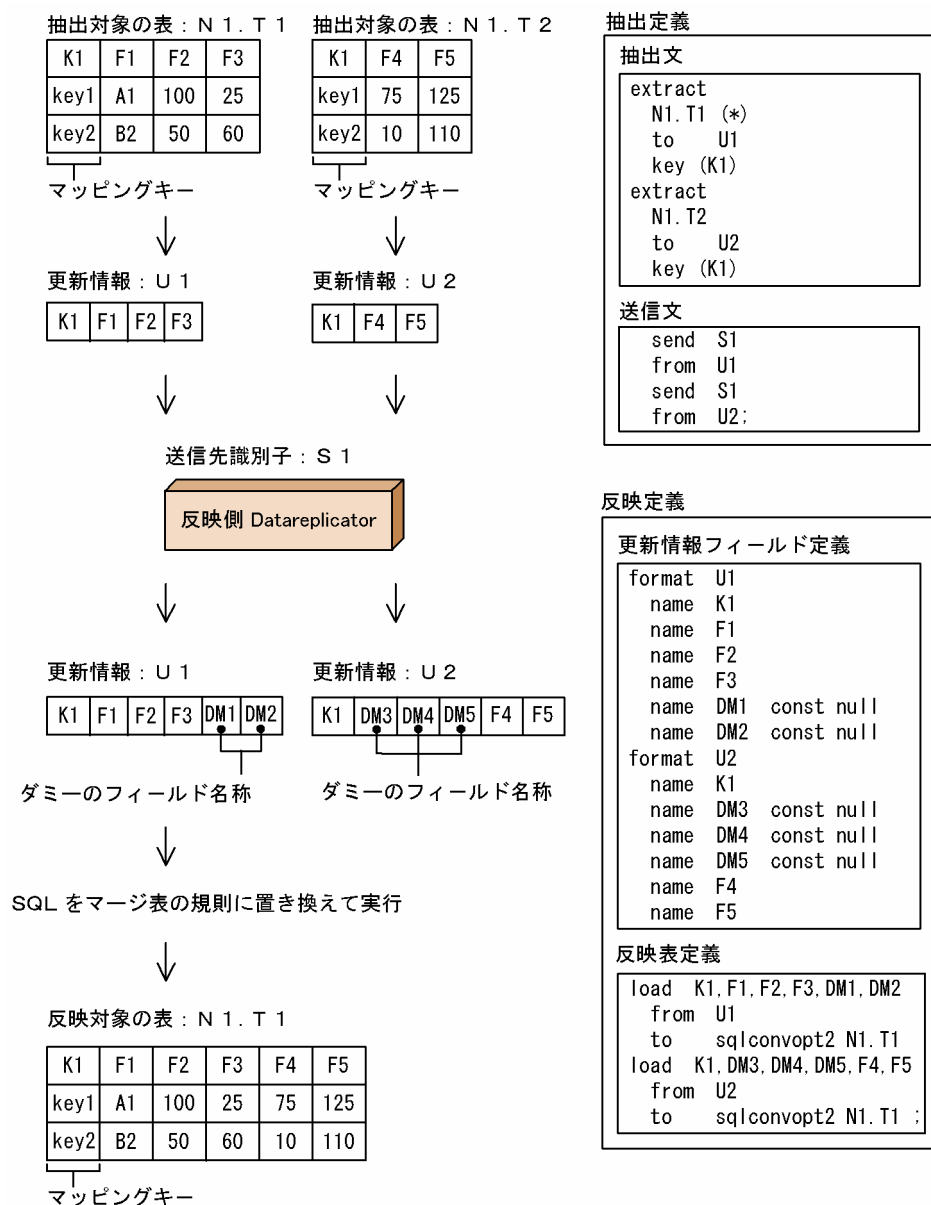
4.2.5 n表から1表にデータ連動するパターン

n表から1表にデータ連動するパターンについて説明します。

複数の抽出対象表を反映側で一つにまとめ、一つの表に反映するときは、**マージ表**を使います。マージ表を反映側システムで生成して、複数の抽出対象表を反映側で一つにまとめて一つの表に反映できます。INSERT、DELETE 及び PURGE TABLE などによる変更は、マージ表の規則又は反映表定義 load 文の指定に従って変換されます。

マージ表を使って、複数の抽出対象表を反映側で一つにまとめて一つの表に反映する例を次の図に示します。

図 4-12 マージ表を使って、複数の抽出対象表を反映側で一つにまとめて一つの表に反映する例



(説明)

• 抽出側での複数の表の抽出、送信

抽出側で複数の表の更新情報を抽出する場合には、抽出する表の数だけ、抽出文を指定します。

• 反映側での複数の表から一つの表への反映

反映側で複数の表から一つの表に反映する場合には、抽出対象表の数だけ、更新情報フィールド定義、及び反映表定義を指定します。また、この場合、抽出対象表と反映対象表とで列の数を対応させる必要があるため、更新情報フィールド定義でユーザ任意のダミーのフィールド名称を指定します。ダミーのフィールド名称には、const 句を指定します。

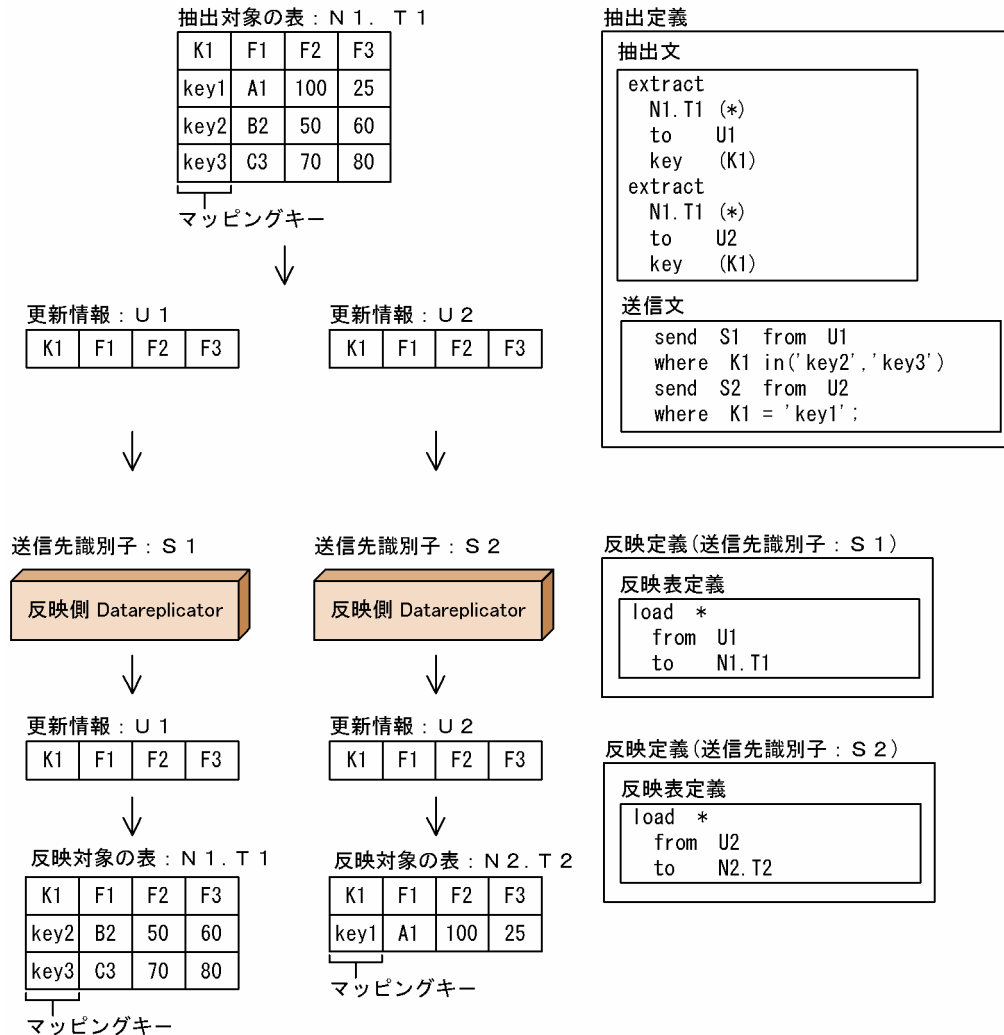
反映表定義では、更新情報フィールド定義で指定したすべてのフィールド名称を指定します。なお、マージ表を作成するときは、反映表定義の to 句で表名の前に sqlconvopt1 又は sqlconvopt2 を指定します。

4.2.6 送信行を選択してデータ連動するパターン

抽出側システムの更新情報のうち、条件に該当する更新情報だけを選んで反映するパターンです。複数の反映側システムごとに、反映する更新情報の条件を指定できます。

送信行を選択してデータ連動する例を次の図に示します。

図 4-13 送信行を選択してデータ連動する例



(説明)

- 抽出側 Datareplicator の指定

抽出側 Datareplicator の抽出定義の送信文 (send 文) に、マッピングキーに対する選択条件を指定します。

- 複数の反映側システムへの送信

送信先の反映側システムの数だけ send 文を指定すると、条件を付けた更新情報を複数の反映側システムに送信できます。

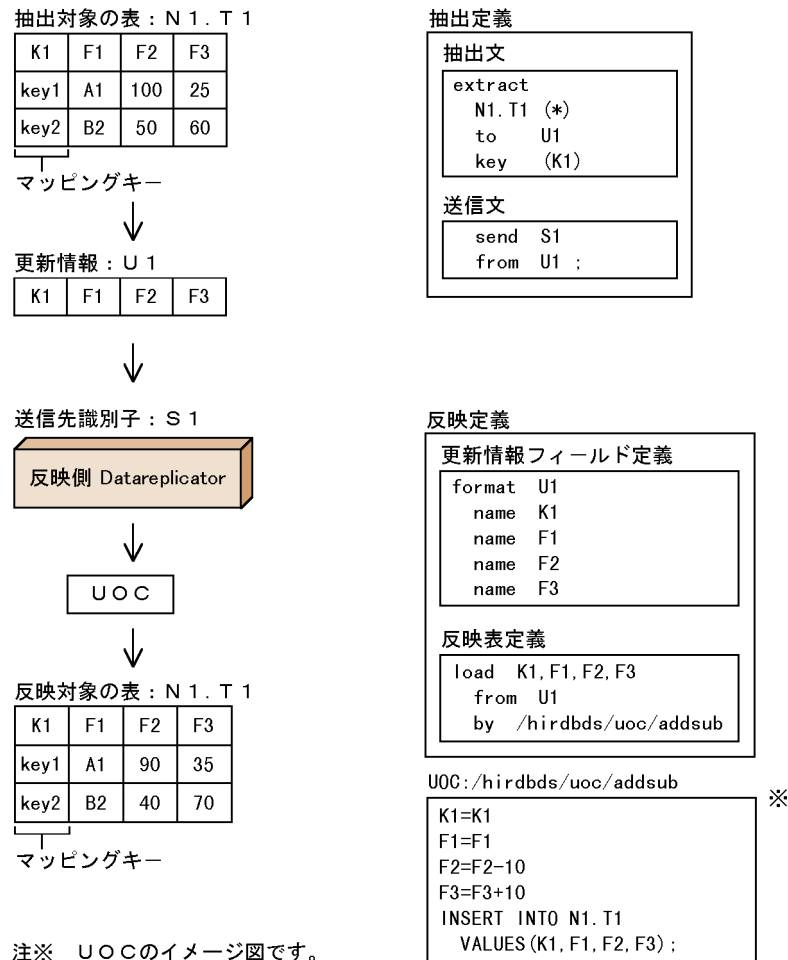
4.2.7 ユーザOWNコーディングを使ってデータ連動するパターン

UOC でデータを編集してから反映するデータ連動のパターンの例について説明します。UOC については、「8. ユーザOWNコーディング」を参照してください。

(1) 更新情報を反映側の UOC で任意に加工して反映する例

UOC を使うと、更新情報を編集・加工してから反映できます。更新情報を反映側の UOC で任意に加工して反映する例を次の図に示します。

図 4-14 反映側の UOC を使って任意に加工して反映する例



(説明)

- UOC によって加工して反映する場合の UOC 名

UOC によって加工して反映する場合は、反映表定義の by 句に UOC 名を指定します。

- UOC での処理

UOC に対しては、反映表定義の load 文で指定した更新情報のフィールドのデータを渡します。UOC では、受け取ったデータを加工して、反映処理をするための SQL 文の生成と発行が必要になります。

UOC は、反映側 Datareplicator で用意する関数を使って作成し、コンパイルとリンケージを実行する必要があります。

• UOC のサンプル

反映側 Datareplicator では UOC のサンプルを用意しています。この UOC を利用して、任意の UOC を作成できます。

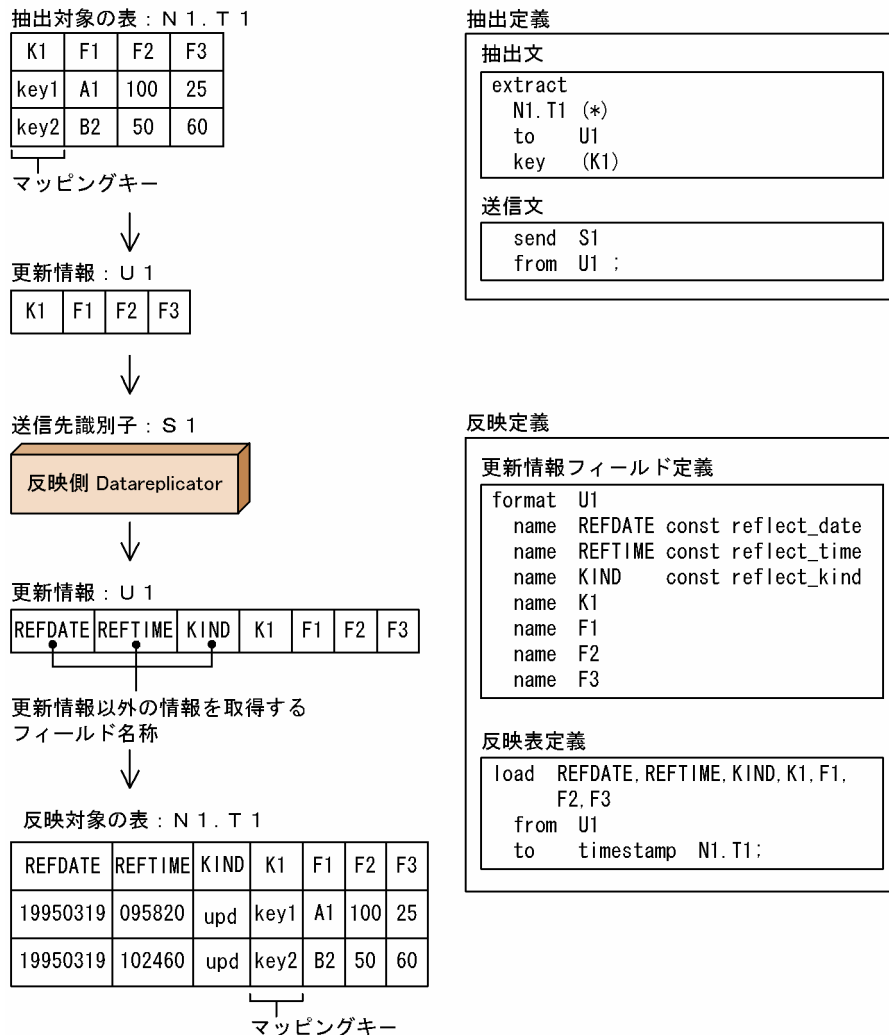
4.2.8 更新情報の履歴を時系列順に取得するパターン

更新情報の履歴を時系列順に取得するパターンを説明します。

(1) 更新情報のほかに反映日付、反映時間などを時系列順に取得する例

更新情報のほかに反映日付、反映時間などを、時系列順に取得する例を次の図に示します。

図 4-15 更新情報のほかに反映日付、反映時間などを時系列順に取得する例



(説明)

- **時系列順に取得する表**

時系列順に取得する場合、反映表定義の to 句で、表名の前に timestamp と指定します。更新情報以外の情報を取得する場合、反映対象表には、あらかじめ更新情報以外の情報を取得する列を作成しておく必要があります。

- **更新情報以外の取得情報**

更新情報以外に反映日付、反映時間、反映種別などの情報を取得する場合には、更新情報フィールド定義の name 句にユーザ任意のフィールド名称を指定します。また、フィールド名称の後に、const と取得する情報を表す定数を指定します。

ここで指定した名称のフィールドのうち、取得の対象とするフィールドを反映表定義の load 文で指定します。

4.3 抽出側 DB と反映側 DB との対応の設計

抽出側 DB と反映側 DB を対応させたシステム設計について説明します。

4.3.1 抽出側 DB の条件

HiRDB のデータベースからデータを抽出する場合の条件について説明します。

メインフレーム側 DB (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) からデータを抽出する場合の条件については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。SAM ファイルを使うメインフレーム側 DB (PDMII E2, RDB1 E2) からデータを抽出する場合の条件については、各製品の該当するマニュアルを参照してください。

(1) 抽出対象の表

抽出対象の表については、「1.3 データ連動で使える表」を参照してください。

(2) 抽出対象のデータ型

抽出対象のデータ型については、「1.4.1 Datareplicator で使えるデータ型」を参照してください。

抽出側 DB と反映側 DB とのデータ型の対応については、「4.3.4 データ型の対応の設計」を参照してください。

(3) 抽出対象の SQL

抽出対象の SQL については、「1.5 データ連動できる SQL」を参照してください。

(4) 抽出対象の UAP

抽出側 Datareplicator は、システムログ中のデータベースの更新ログを基にデータ連動を実行します。このため、データベースの更新ログを出力する UAP での、データ連動対象 DB の更新は、すべて抽出対象になります。

データ連動の実行時、データ連動対象 DB を更新する UAP を実行する場合には、注意が必要な環境変数と SQL 文があります。UAP 実行時の注意を次の表に示します。

表 4-4 UAP 実行時の注意

項目	注意
環境変数でのログレスモードの指定	HiRDB のクライアント環境定義の環境変数 PDDBLOG に、NO を指定して実行しないでください。

項目	注意
SQL 文での定義の変更	<p>データ連動の実行中に、データ連動対象 DB に対して、次の SQL 文を発行しないでください。</p> <ul style="list-style-type: none"> ・ PURGE TABLE (複数の BES にわたる分割表に対する発行※) ・ ALTER TABLE ・ DROP TABLE <p>データ連動の実行時の定義変更については、「6.5.6 抽出側 HiRDB の運用手順」を参照してください。</p>

注※

抽出側 Datareplicator が複数の BES にわたる分割表に対する PURGE TABLE を検知すると、エラー情報ファイルに警告メッセージを出力して、該当する更新情報を抽出しません。このような状態になったときには、送信環境定義の prg_eventno オペランドの指定で、送信処理を停止したり、反映側 Datareplicator に対してイベントを発行したりできます。

(5) 抽出対象のユーティリティ

HiRDB Datareplicator が抽出対象として対応しているのはデータベース作成ユーティリティ (pdload) だけです。ただし、オプションを正しく指定しないと抽出-反映間の不整合を引き起こす可能性があります。BLOB 型のデータを抽出できないという制約もあります。

また、その他のユーティリティのうち更新ログを出力するものについてはデータ連動中に実行すると抽出-反映間で不整合を引き起こすため、オプション指定により抽出対象外として実行する必要があります。

ユーティリティ実行時の注意を次の表に示します。

表 4-5 ユティリティ実行時の注意

ユーティリティ名		注意
データベース定義ユーティリティ	pddef	<p>データ連動の実行中に、データ連動対象 DB に対して、次の SQL 文を発行しないでください。</p> <ul style="list-style-type: none"> ・ ALTER TABLE ・ DROP TABLE <p>データ連動の実行時の定義変更については、「6.5.6 抽出側 HiRDB の運用手順」を参照してください。</p>
データベース作成ユーティリティ	pdload	<ul style="list-style-type: none"> • 既存の表の場合 データ連動の実行中に、作成モードで実行しないでください (-d オプションは指定できません)。追加モードで、ログ取得モードを指定して実行してください (-l オプションには a を指定してください)。 • 新規の表の場合 データ連動の実行中に連動対象の表を新規に追加するときは、ログ取得モードで実行してください (-l オプションには a を指定してください)。

ユーティリティ名		注意
		なお、pdload をデータ連携する場合、「4.7.3(9)(a) 排他資源数」の章も参照してください。
データベース構成変更ユーティリティ	pdmod	データ連動の実行中は、抽出対象表が格納してある RD エリアを再初期化しないでください。
データベース再編成ユーティリティ	pdrorg	データ連動の実行中は、ログレスモード又は更新前ログ取得モードで実行してください (-l オプションに n 又は p を指定してください)。ログ取得モードで実行すると、再編成したデータベースの一部の更新情報だけがシステムログに出力されるので、抽出側 DB と反映側 DB との間で不整合が発生します。
データベース回復ユーティリティ	pdrstr	データベースのバックアップとバックアップ取得以降のログを基にして回復してください。それ以外の方法で回復すると、抽出側 DB と反映側 DB との間で不整合が発生します。
リバランスユーティリティ	pdrbal	データ連動対象表に対してリバランスユーティリティを実行する場合、表定義の変更 (RD エリアの追加) をする必要があります。表定義と抽出定義変更時の運用手順については、「6.5.6(3) 抽出側の表定義と抽出定義変更時の運用手順」を参照してください。なお、リバランスユーティリティによる DB 更新は、抽出対象外となります。

注

ユーティリティの処理対象に BLOB 列が含まれている場合は、ユーティリティが実行した DB 更新を抽出側 Datareplicator が無視するため、反映側 DB にはユーティリティが更新した行は反映されません。

(6) 抽出対象の更新情報長

抽出側 Datareplicator が抽出対象にできる更新情報の最大長 (抽出対象列の定義長の総和) は 256MB です。更新情報長が 256MB より大きい場合は抽出定義プリプロセスコマンドの実行でエラーになります。

ただし、SGMLTEXT 型及び XML 型の列はチェック対象外のため、SGMLTEXT 型及び XML 型を抽出対象とする場合は、それぞれの型で扱うデータ長を考慮した上で 256MB 以内となるように抽出対象列を定義してください。

4.3.2 反映対象表の作成

反映対象表の作成手順について説明します。

(1) 反映対象表の作成方法

反映対象表を作成します。反映対象表のデータの作成方法の種類を次に示します。

- HiRDB のデータベース作成ユーティリティによる作成

- HiRDB の SQL 文による作成
- HiRDB Dataextractor による作成

HiRDB での表の作成方法については、マニュアル「HiRDB システム運用ガイド」を参照してください。HiRDB Dataextractor での作成方法については、マニュアル「データベース抽出・反映サービス機能 HiRDB Dataextractor」を参照してください。

(2) 反映対象表の条件

表を作成するときには、次に示す条件を満たす必要があります。

- 作成する表は、実表にします。
- マッピングキーを格納する列を設定しておきます。
- 表を構成する列の数を 4,000 以内にします。

4.3.3 マッピングキーの対応の設計

マッピングキーとは、更新情報が反映対象表のどの行に対応するのかを特定するためのキーのことです。このため、マッピングキーは抽出対象表と反映対象表とで対応させる必要があります。マッピングキーは 1 個以上の列で構成します。

注意 1

マッピングキーは、反映対象のどの行に反映するかを特定するためのものです。マッピングキーが抽出側と反映側の間で特定できないときは、該当する行が複数になって、残りの反映処理で不都合が起きます。

マッピングキーで行が特定できない場合：

データを反映するときに、行が存在しないエラーになって終了することがあります。この場合は、反映環境定義の skip_sqlcode オペランドにスキップする SQLCODE を指定して再実行してください。

注意 2

反映対象表には、すべてのマッピングキー列を構成列とするユニークキーインデックスを定義してください。ユニークキーインデックスを定義していないと、キー重複エラーを検知できなくなります。さらに、UPDATE と DELETE の反映性能が著しく低下します。

マッピングキーは、抽出側 DB によって指定方法が異なります。抽出側 DB ごとのマッピングキーについて説明します。

(1) 抽出側 DB が HiRDB の場合のマッピングキー

マッピングキーに指定できるデータ型を次に示します。

表 4-6 マッピングキーに指定できるデータ型（抽出側 DB が HiRDB）

分類	マッピングキーに指定できるデータ型	列定義長
数値型	INTEGER	—
	SMALLINT	—
	DECIMAL(m,n)	$1 \leq m \leq 38, 0 \leq n \leq 38, n \leq m$
	FLOAT	—
	SMALLFLT	—
文字型	CHAR(n)	$n \leq 255$
	VARCHAR(n)	$n \leq 255$
	NCHAR(n)	$n \leq 127$
	NVARCHAR(n)	$n \leq 127$
	MCHAR(n)	$n \leq 255$
	MVARCHAR(n)	$n \leq 255$
日付型	DATE	—
	TIME	—
	TIMESTAMP(p)	$p = 0, 2, 4, 6$
	INTERVAL YEAR TO DAY	—
	INTERVAL HOUR TO SECOND	—

(凡例)

—：該当しません。

注

繰返し列はマッピングキーに指定できません。

- 抽出定義で抽出定義文を指定する場合、マッピングキーは key 句 又は ukey 句で指定します。抽出定義で全表抽出定義文を指定する場合、マッピングキーは Datareplicator が決定します。マッピングキーの決定方法については、「表 5-14 マッピングキーのユニーク条件とインデクス条件」を参照してください。
- マッピングキーには、最大 64 個の列が指定できます。
- 抽出定義の抽出定義文に key 句で列を指定した場合は、マッピングキーは更新しないでください。マッピングキーを更新すると反映側 DB では更新後のキーで反映されるので、抽出側 DB での更新とは異なる行に対する更新になります。このため、抽出側 DB と反映側 DB との間に不整合が発生することがあります。マッピングキーを更新する場合は、ukey 句で列を指定してください。
- 反映側 Datareplicator には、INSERT、UPDATE、及び DELETE 時に操作したすべての行のマッピングキーに対応するデータが送信されます。

(2) 抽出側 DB が XDM/SD E2 の場合のマッピングキー

- 抽出側 DB が XDM/SD E2 の場合のマッピングキーの指定方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。
- 反映側 Datareplicator には、次のデータが送信されます。

抽出対象レコードの ERASE 時

抽出対象レコードと、それより上の階層のマッピングキーに対応するデータ

抽出対象レコードより上位のレコードの ERASE 時

削除したレコードと、それより上の階層のマッピングキーに対応するデータ

抽出対象レコードの STORE, 又は MODIFY 時

抽出対象レコードと、それより上の階層のマッピングキーに対応するデータ

(3) 抽出側 DB が XDM/RD E2 の場合のマッピングキー

- 抽出側 DB が XDM/RD E2 の場合のマッピングキーの指定方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。
- 反映側 Datareplicator には、INSERT, UPDATE, 及び DELETE 時に操作したすべての行のマッピングキーに対応するデータが送信されます。

(4) 抽出側 DB が ADM の場合のマッピングキー

- 抽出側 DB が ADM の場合のマッピングキーの指定方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。
- 反映側 Datareplicator には、次のデータが送信されます。
ISRT, REPL, DLET コールのマッピングキーに対応するデータ

(5) 抽出側 DB が PDMII E2 の場合のマッピングキー

(a) XDM/DS を使ってデータ連動する場合

- 抽出側 DB が PDMII E2 の場合のマッピングキーの指定方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

(b) SAM ファイルを使ってデータ連動する場合

- 抽出側 DB が PDMII E2 の場合のマッピングキーは、更新情報定義で指定します。更新情報定義の指定については、「5.11 更新情報定義」を参照してください。
- 反映側 Datareplicator の反映情報キューファイルには、抽出側 DB から転送された SAM ファイルの内容が格納されます。

(6) 抽出側 DB が TMS-4V/SP の場合のマッピングキー

- 抽出側 DB が TMS-4V/SP の場合のマッピングキーの指定方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。
- 反映側 Datareplicator には、INSERT、UPDATE、及び DELETE 時に操作したすべての行のマッピングキーに対応するデータが送信されます。

(7) 抽出側 DB が RDB1 E2 の場合のマッピングキー

- 抽出側 DB が RDB1 E2 の場合のマッピングキーは、RDB1 E2 の更新情報抽出機能で生成されます。
- 反映側 Datareplicator の反映情報キューファイルには、抽出側 DB から転送された SAM ファイルの内容が格納されます。

(8) SMALLFLT, FLOAT 型の抽出列について

マッピングキーが SMALLFLT, FLOAT の概数型の場合、抽出システム別でデータ形式が異なります。さらに、概数型は SQL 実行時に SQLCODE 100 (条件に合う行がありません) になる可能性があります。FLOAT, SMALLFLT 型の列はマッピングキーに指定しないでください。

4.3.4 データ型の対応の設計

表のデータ型と Datareplicator で反映するデータ形式の関係を次の表に示します。

表 4-7 表のデータ型と Datareplicator で反映するデータ形式の関係

データ型	データ形式, データ長	文字コード変換の可否
INTEGER	HiRDB のデータ形式, データ長と同じ	×
SMALLINT		×
DECIMAL		×
LARGE DECIMAL		×
FLOAT		×
SMALLFLT		×
CHAR		○
VARCHAR	○	
LONG VARCHAR	VARCHAR として認識	○
NCHAR	HiRDB のデータ形式, データ長と同じ	○
NVARCHAR		○

データ型	データ形式, データ長	文字コード変換の可否
LONG NVARCHAR	NVARCHAR として認識	○
MCHAR	HiRDB のデータ形式, データ長と同じ	○
MVARCHAR		○
LONG MVARCHAR	MVARCHAR として認識	○
DATE	HiRDB のデータ形式, データ長と同じ	×
TIME		×
TIMESTAMP		×
INTERVAL YEAR TO DATE		×
INTERVAL HOUR TO SECOND		×
SGMLTEXT		○
FREWORD		○
XML		×
BLOB		×
BINARY		×
繰返し列		×

(凡例)

- ：文字コード変換できます。
- ×

抽出側 DB によって、抽出側と反映側とでデータ型が異なることがあります。このため、反映対象表の作成時には、抽出対象表と反映対象表のデータ型を対応させる必要があります。データ型が対応していないと、反映処理がエラーになります。

データ型の対応について、抽出側 DB ごとに説明します。

(1) 抽出側 DB が HiRDB の場合のデータ型の対応

抽出側 DB が HiRDB の場合の、抽出対象表と反映対象表のデータ型の対応を次の表に示します。

表 4-8 HiRDB の場合の抽出対象表と反映対象表のデータ型の対応

抽出対象表のデータ型	反映対象表のデータ型
INTEGER	INTEGER
SMALLINT	SMALLINT
DECIMAL	DECIMAL

抽出対象表のデータ型	反映対象表のデータ型
LARGE DECIMAL	LARGE DECIMAL
FLOAT	FLOAT
SMALLFLT	SMALLFLT* ¹
CHAR	CHAR 又は MCHAR
VARCHAR	VARCHAR 又は MVARCHAR
NCHAR	NCHAR
NVARCHAR	NVARCHAR
MCHAR	CHAR 又は MCHAR
MVARCHAR	VARCHAR 又は MVARCHAR
DATE	DATE
TIME	TIME
TIMESTAMP	TIMESTAMP
INTERVAL YEAR TO DAY	INTERVAL YEAR TO DAY
INTERVAL HOUR TO SECOND	INTERVAL HOUR TO SECOND
SGMLTEXT* ²	SGMLTEXT* ²
FREWORD	FREWORD
XML	XML
BLOB* ²	BLOB* ²
BINARY	BINARY

注

- 反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。
- 繰返し列を指定できます。ただし、繰返し列の要素単位では抽出できません。

注※1

データ連動できますが、概数のため丸め誤差が発生します。

注※2

データ連動する場合は、HiRDB の表定義の RECOVERY オペランドに ALL を指定してください。HiRDB の表定義については、マニュアル「HiRDB SQL リファレンス」を参照してください。

(2) 抽出側 DB が XDM/SD E2 の場合のデータ型の対応

抽出側 DB が XDM/SD E2 の場合の、抽出対象のデータと反映対象表のデータ型の対応を次の表に示します。

表 4-9 XDM/SD E2 の場合の抽出対象のデータと反映対象表のデータ型の対応

抽出対象のデータ※1		反映対象表のデータ型
データ属性	データの長さ	
CHAR(n)	$1 \leq n \leq 30000$	CHAR(n)
CHAR(n)	$n > 30000$	VARCHAR(n) ※3
NCHAR(n)	$1 \leq n \leq 15000$	NCHAR(n)
NCHAR(n)	$n > 15000$	NVARCHAR(n) ※3
PACK(m,n)	—	DECIMAL(m + n,n) ※4
UNPACK(m,n)※2	—	DECIMAL(m + n,n) ※4
COMP(m,n)	$1 \leq m \leq 4, n = 0$	SMALLINT
COMP(m,n)	$5 \leq m \leq 9, n = 0$	INTEGER
COMP(m,n)	$m = 0$ 又は $m > 10$ 又は $n \neq 0$	DECIMAL(29,n) ※6
BINARY(m,n)	$1 \leq m \leq 15, n = 0$	SMALLINT
BINARY(m,n)	$16 \leq m \leq 31, n = 0$	INTEGER
BINARY(m,n)	$m = 0$ 又は $m > 32$ 又は $n \neq 0$	DECIMAL(29,n) ※5

(凡例)

—：該当しません。

注

反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。

注※1

XDM/DS で RESTRUCT 文を指定している場合は、RESTRUCT 文で定義した属性と長さになります。

注※2

UNPACK 型のデータ変換で、次の条件のどれかを満たすとエラーになります。

- ゾーン部の値が (F)₁₆ 以外の場合
- 符号部の値が (F)₁₆, (C)₁₆, (D)₁₆ 以外の場合
- 数値部の値が (0)₁₆ 以上 (9)₁₆ 以下 以外の場合

エラーになると、エラーメッセージと未反映情報が出力されて、エラーになった反映グループの反映処理が停止します。UOCを使った反映処理では、UNPACK型のデータ変換をしないで、抽出データをそのままUNPACK型データとしてUOCに渡し、反映処理を続行します。

注※3

空白サプレスはしません。

注※4

符号の正規化はしません。

注※5

BINARY型は、小数部けた数が0以外のときには、DECIMAL型に変換されます。そのとき、XDM/DSで扱えるDECIMAL型の精度の最大値は29のため、小数点以下30けた以降は切り捨てられます。

注※6

XDM/DSで扱えるDECIMAL型の精度の最大値は29になります。

(3) 抽出側DBがXDM/RD E2の場合のデータ型の対応

抽出側DBがXDM/RD E2の場合の、抽出対象表と反映対象表のデータ型の対応を次の表に示します。

表 4-10 XDM/RD E2の場合の抽出対象表と反映対象表のデータ型の対応

抽出対象表のデータ型	反映対象表のデータ型
INTEGER	INTEGER
SMALLINT	SMALLINT
DECIMAL	DECIMAL
LARGE DECIMAL	LARGE DECIMAL
FLOAT	FLOAT
SMALLFLT	SMALLFLT※1
CHAR	CHAR 又は MCHAR
VARCHAR	VARCHAR
LONG VARCHAR	
NCHAR	NCHAR
NVARCHAR	NVARCHAR
LONG NVARCHAR	
MCHAR	CHAR 又は MCHAR
MVARCHAR	VARCHAR 又は MVARCHAR
LONG MVARCHAR	

抽出対象表のデータ型	反映対象表のデータ型
DATE	DATE
TIME*3	TIME
INTERVAL YEAR TO DAY	INTERVAL YEAR TO DAY
INTERVAL HOUR TO SECOND*3	INTERVAL HOUR TO SECOND
BLOB*2	BLOB*2

注

- 反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。
- 繰返し列と配列列は指定できません。

注※1

データ連動できますが、概数のため丸め誤差が発生します。

注※2

BLOB 型の列をデータ連動する場合は、HiRDB の表定義の RECOVERY オペランドに ALL を指定してください。HiRDB の表定義については、マニュアル「HiRDB SQL リファレンス」を参照してください。

注※3

小数部を指定した列はデータ連動できません。

(4) 抽出側 DB が ADM の場合のデータ型の対応

抽出側 DB が ADM の場合の、抽出対象のデータと反映対象表のデータ型の対応を次の表に示します。

表 4-11 ADM の場合の抽出対象のデータと反映対象表のデータ型の対応

ADM でのデータ型		Datareplicator でのデータ型	反映対象表のデータ型
Cn		CHAR(n)	CHAR(n)
Xn		CHAR(n)	CHAR(n)
Pn	n ≤ 15	PACK(2n-1,0)	DECIMAL(2n-1,0)
	n > 15	CHAR(n)	CHAR(n)

注 1

反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。

注 2

XDM/DS の RESTRUCT 文で ADM のデータ型を UNPACK 属性とした場合に、抽出された更新情報に次に示すデータが含まれるときは、反映処理が停止します。

- ゾーン部の値が、(F)₁₆ 以外の場合
- 符号部の値が (F)₁₆, (C)₁₆, (D)₁₆ 以外の場合
- 数値部の値が (0)₁₆ 以上 (9)₁₆ 以下 以外の場合

エラーになると、エラーメッセージと未反映情報が出力されて、エラーになった反映グループの反映処理が停止します。UOC を使った反映処理では、UNPACK 型のデータ変換をしないで、抽出データをそのまま UNPACK 型データとして UOC に渡し、反映処理を続行します。

(5) 抽出側 DB が PDMII E2 の場合のデータ型の対応

抽出側 DB が PDMII E2 の場合の、抽出再定義フィールドと反映対象表のデータ型の対応を表 4-12 に、PDMII E2 のフィールド属性と反映対象表のデータ型の対応を表 4-13 に示します。

表 4-12 抽出再定義フィールドと反映対象表のデータ型の対応

抽出再定義文のフィールド属性	データの長さ	反映対象表のデータ型
CHAR(n)	$n \leq 30000$	CHAR(n)
NCHAR(n)	$n \leq 15000$	NCHAR(n)
PACK(m,n)	—	DECIMAL(m + n,n)
PACKNS(m,n)	—	DECIMAL(m + n,n)
UNPACK(m,n)	—	DECIMAL(m + n,n)
UNPACKNS(m,n)	—	DECIMAL(m + n,n)
COMP(m,n)	$1 \leq m \leq 4, n = 0$	SMALLINT
	$5 \leq m \leq 9, n = 0$	INTEGER
	$m = 0$ 又は $m > 10$ 又は $n \neq 0$	DECIMAL(29,n)*
BINARY(m,n)	$1 \leq m \leq 15, n = 0$	SMALLINT
	$16 \leq m \leq 31, n = 0$	INTEGER
	上記以外	FLOAT
FLOAT(n)	$n = 4, 8$	FLOAT

注

反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。

注※

XDM/DS で扱える DECIMAL 型の精度の最大値は 29 になります。

表 4-13 PDMII E2 のフィールド属性と反映対象表のデータ型の対応

PDMII E2 のフィールド属性	データの長さ	反映対象表のデータ型
X(n)	$n \leq 30000$	CHAR(n)
P(m,n), AP(m,n)	$n > 0, 2m-1 > n$	DECIMAL(2m-1,n)
	$n > 0, 2m-1 \leq n$	DECIMAL(n,n)
	$n \leq 0$	DECIMAL(2m-1-n,0)
Z(m,n), AZ(m,n)	$n > 0, m > n$	DECIMAL(m,n)
	$n > 0, m \leq n$	DECIMAL(n,n)
	$n \leq 0$	DECIMAL(m-n,0)
B(m,n), AB(m,n)	$n > 0, 2.5m > n$	DECIMAL(2.5m,n)
	$n > 0, 2.5m \leq n$	DECIMAL(n,n)
	$n \leq 0$	DECIMAL(2.5m-n,0)
F(n)	$n = 4, 8$	FLOAT
N(n)	$n \leq 30000$	NCHAR ($\uparrow n \div 2 \uparrow$)

注

反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。

(6) 抽出側 DB が TMS-4V/SP の場合のデータ型の対応

抽出側 DB が TMS-4V/SP の場合は、TMS-4V/SP/データ連携支援のユーザOWNコーディング (UOC) でデータ型を設定します。抽出側 DB が TMS-4V/SP の場合のデータ型については、マニュアル「VOS3 TMS-4V/SP データ連携支援機能 TMS-4V/SP/データ連携支援」を参照してください。

(7) 抽出側 DB が RDB1 E2 の場合のデータ型の対応

抽出側 DB が RDB1 E2 の場合の、抽出対象表と反映対象表のデータ型の対応を次の表に示します。

表 4-14 RDB1 E2 の場合の抽出対象表と反映対象表のデータ型の対応

抽出対象表のデータ型	反映対象表のデータ型
INTEGER	INTEGER
SMALLINT	SMALLINT
DECIMAL	DECIMAL
FLOAT	FLOAT
SMALLFLT	SMALLFLT*

抽出対象表のデータ型	反映対象表のデータ型
CHAR	CHAR 又は MCHAR
VARCHAR	VARCHAR
LONG VARCHAR	—
NCHAR	NCHAR
NVARCHAR	NVARCHAR

(凡例)

—：反映の対象にできません。

注

- 反映対象表の列よりも長いデータや、数値がオーバーフローするデータを反映しようとする、反映側 HiRDB でエラーになります。
- 繰返し列は指定できません。

注※

データ連動できますが、概数のために丸め誤差が発生します。

4.3.5 文字コードの対応の設計

文字コードの対応の設計方法について説明します。

(1) 文字コードの種類

抽出側 DB と反映側 DB の文字コードの種類を次に示します。

- EBCDIC/KEIS, EBCDIK/KEIS

抽出側 DB がメインフレームの場合の、抽出側 DB の文字コードです。

EBCDIC/KEIS, EBCDIK/KEIS には、次の 4 種類があります。

- EBCDIC/KEIS78
- EBCDIK/KEIS78
- EBCDIC/KEIS83
- EBCDIK/KEIS83

このマニュアルでは、以降この 4 種類のコードを EBCDIK/KEIS と表記します。

- EBCDIK

メインフレームの 1 バイトコードです。

- JIS8/シフト JIS

抽出側 DB 又は反映側 DB が HiRDB の場合の文字コードです。

HiRDB の pdsetup コマンドでシフト JIS 漢字コードを指定している場合に使われる文字コードです。抽出側 Datareplicator では抽出システム定義の dblocale オペランドで、反映側 Datareplicator では反映システム定義の dblocale オペランドで、対応する HiRDB の文字コードを指定します。

- EUC

抽出側 DB 又は反映側 DB が HiRDB の場合の文字コードです。

HiRDB の pdsetup コマンドで EUC 日本語漢字コードを指定している場合に使われる文字コードです。抽出側 Datareplicator では抽出システム定義の dblocale オペランドで、反映側 Datareplicator では反映システム定義の dblocale オペランドで、対応する HiRDB の文字コードを指定します。

- UCS2

UCS2 は、Unicode UCS2 文字セットです。Datareplicator では、UTF-8 へのエンコードルールだけを意識し、コードと文字のマッピングについては意識していません。使用できる Unicode の範囲は UCS2 の BMP 面（基本多言語面）で、UTF-8 へのエンコードは UCS2 を UTF-8 形式にエンコードしたものです。

- UTF-8

抽出側 DB 又は反映側 DB が HiRDB の場合の文字コードです。HiRDB の pdsetup コマンドで UTF-8 を指定している場合に使われる文字コードです。UTF-8 は、Unicode の文字を表現する文字エンコーディングスキームの一つで 8 ビット単位の情報の組み合わせで文字を表記します。抽出側 Datareplicator では抽出システム定義の dblocale オペランド、反映側 Datareplicator では反映システム定義の dblocale オペランドで、対応する HiRDB の文字コードを指定します。

(2) 文字コードの変換

抽出側 DB と反映側 DB との文字コードが異なっている場合、Datareplicator は文字コードを定義に従って変換します。抽出側 DB と反映側 DB で同じ文字コードを使っている場合は、文字コードを変換する必要はありません。

(a) 反映側 Datareplicator での文字コードの変換

反映側 Datareplicator では、抽出側 DB の文字コードを反映側 DB の文字コードに対応させて変換します。反映側 DB の文字コードは、反映システム定義の dblocale オペランドで指定します。

抽出側 DB がメインフレーム側 DB の場合、抽出側 DB の文字コードの EBCDIK/KEIS の種類は、反映環境定義の ebcdic_type オペランドで指定します。

抽出側 DB と反映側 DB の文字コードの対応を次の表に示します。

表 4-15 抽出側 DB と反映側 DB の文字コードの対応

抽出側 DB の文字コード	反映側 DB の文字コード				
	EBCDIK/KEIS*1	EBCDIK	JIS8/シフト JIS	EUC*2	UTF-8
EBCDIK/KEIS*1	—	—	○	○	○

抽出側 DB の文字コード	反映側 DB の文字コード				
	EBCDIK/KEIS*1	EBCDIK	JIS8/シフト JIS	EUC*2	UTF-8
EBCDIK	—	—	○	—	—
JIS8/シフト JIS	○	○	—	○	○
EUC*2	○	—	○	—	○
UTF-8	○	—	○	○	—

(凡例)

- ：抽出側 DB の文字コードを反映側 DB の文字コードに変換できます。
- ：文字コードを変換する必要はありません。

注※1

EBCDIK/KEIS を扱えるのは XDM/DS と VOS3 Database Datareplicator だけです。

注※2

コードセット 0~2 以外の動作は保証しません。

(3) 文字コード変換の規則

Datereplicator での文字コードの変換規則について説明します。

文字コードの変換方法

文字コード変換用マッピングテーブルを使って文字コードを変換します。

外字の変換方法の定義

外字の変換方法は、hdscnvedt コマンドで文字コード変換用マッピングテーブルを編集して定義します。

(4) EBCDIK/KEIS から JIS8/シフト JIS への文字コードの変換規則

(a) 1 バイトコード

対応するシフト JIS の文字コードに変換されます。

(b) 2 バイトコード (標準文字コード)

- 対応するシフト JIS の文字コードに変換されます。
- 機能キャラクタ ((0A42)₁₆, (0A41)₁₆) は削除して詰められます。これによって余りができた場合は、空白文字 ((20)₁₆) に変換されます。
- 機能キャラクタで囲まれた文字列中の (00)₁₆ ~ (40)₁₆ は、対応するシフト JIS の文字コードに変換されます。
- 最後の文字が 2 バイトコードの先頭バイトの場合は、空白文字 ((20)₁₆) に変換されます。

(c) 2バイトコード (外字)

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。

未定義の外字，及び文字コード変換用マッピングテーブルを作成していない場合は，反映環境定義の `undefcode_cnv` オペランドの指定値に従って，空白文字に変換されます。

(d) 空白文字

シフトコード中の空白文字は，変換されます。空白文字の変換規則 (EBCDIK/KEIS から JIS8/シフト JIS) を次の表に示します。

表 4-16 空白文字の変換規則 (EBCDIK/KEIS から JIS8/シフト JIS)

EBCDIK/KEIS の文字コード	JIS8/シフト JIS の文字コード
全角の空白文字 (A1A1) ₁₆	全角の空白文字 (8140) ₁₆
二つの連続した半角の空白文字 (40) ₁₆ (40) ₁₆	反映環境定義の <code>shiftspace_cnv</code> オペランドの指定値に従って，全角の空白文字又は二つの連続した半角の空白文字に変換します。
半角の空白文字 (40) ₁₆	半角の空白文字 (20) ₁₆

(5) EBCDIK/KEIS から EUC への文字コードの変換規則

(a) 1バイトコード

- 仮名文字を除く 1 バイトコードは，対応する EUC の文字コードに変換されます。
- 仮名文字は，2 バイトコードに変換されます。

(b) 2バイトコード (標準文字コード)

- 対応する EUC の文字コードに変換されます。
- 機能キャラクタ ((0A42)₁₆, (0A41)₁₆) は削除して詰められます。これによって余りができた場合は，空白文字 ((20)₁₆) に変換されます。
- 機能キャラクタで囲まれた文字列中の (00)₁₆ ~ (40)₁₆ は，対応する EUC の文字コードに変換されません。
- 最後の文字が 2 バイトコードの先頭バイトの場合，空白文字 ((20)₁₆) に変換されます。

(c) 2バイトコード (外字)

9024 文字中の外字領域の後ろ 8836 文字を，EUC 文字コードの外字領域に固定的に割り当てて変換されます。

それ以外のコードは、ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。未定義の外字、及び文字コード変換用マッピングテーブルを作成していない場合は、反映環境定義の `undefcode_cnv` オペランドの指定値に従って、空白文字に変換されます。

ただし、コードセット 3 は SQL 文の発行時にエラーになることがあります。

(d) 空白文字

空白文字は、変換されます。空白文字の変換規則 (EBCDIK/KEIS から EUC) を次の表に示します。

表 4-17 空白文字の変換規則 (EBCDIK/KEIS から EUC)

EBCDIK/KEIS の文字コード	EUC の文字コード
全角の空白文字 (A1A1) ₁₆	全角の空白文字 (A1A1) ₁₆
二つの連続した半角の空白文字 (40) ₁₆ (40) ₁₆	反映環境定義の <code>shiftspace_cnv</code> オペランドの指定値に従って、全角の空白文字又は二つの連続した半角の空白文字に変換します。
半角の空白文字 (40) ₁₆	半角の空白文字 (20) ₁₆

(e) オーバフローが発生した場合の対処

半角の仮名文字は 1 バイトから 2 バイトに変換されるため、抽出対象表で半角の仮名文字を使っている場合、変換後のデータ長が長くなってオーバフローが発生することがあります。オーバフローが発生した場合の対処 (EBCDIK/KEIS から EUC) を次の表に示します。

表 4-18 オーバフローが発生した場合の対処 (EBCDIK/KEIS から EUC)

発生場所	Datareplicator の動作	ユーザの対処
定数, 更新データ	SQL エラー (全バイト数の変換後に、データが定義長を超えた場合)	定義長を変更して、反映処理を再実行してください。

注

識別子が 30 バイトを超えた場合、又は更新データが 32,000 バイトを超えた場合には、Datareplicator は処理を続行できなくなります。このため、データ連動を開始する前に、抽出側システムをチェックして、識別子と更新データが制限値を超えないことを確認してください。

(6) EBCDIK/KEIS から UTF-8 への文字コードの変換規則

(a) 1 バイトコード

- 仮名文字を除く 1 バイトコードは、対応する UTF-8 の文字コードに変換されます。
- 仮名文字は、3 バイトコードに変換されます。

(b) 2 バイトコード (標準文字コード)

- 対応する UTF-8 の文字コードに変換されます。
- 機能キャラクタ((0A42)₁₆, (0A41)₁₆)は削除して詰められます。これによって余りができた場合は、空白文字((20)₁₆)に変換されます。
- 機能キャラクタで囲まれた文字列中の(00)₁₆~(40)₁₆ は、対応する UTF-8 の文字コードに変換されま
す。
- 最後の文字が 2 バイトコードの先頭バイトの場合、空白文字(20)₁₆ に変換されます。

(c) 2 バイトコード (外字)

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。

未定義の外字、及び文字コード変換用マッピングテーブルを作成していない場合は、反映環境定義の
undefcode_cnv オペランドの指定に従って、空白文字に変換されます。

(d) 空白文字

シフトコード中の空白文字は、変換されます。空白文字の変換規則 (EBCDIK/KEIS から UTF-8) を次
の表に示します。

表 4-19 空白文字の変換規則 (EBCDIK/KEIS から UTF-8)

EBCDIK/KEIS の文字コード	UTF-8 の文字コード
全角の空白文字 (A1A1) ₁₆	全角の空白文字 (E38080) ₁₆
二つの連続した半角の空白文字 (40) ₁₆ (40) ₁₆	反映環境定義の shiftspace_cnv オペランドの指定値に従って、 全角の空白文字又は二つの連続した半角の空白文字に変換しま す。
半角の空白文字 (40) ₁₆	半角の空白文字 (20) ₁₆

(e) オーバフローが発生した場合の対処

半角の仮名文字は 1 バイトから 3 バイトに変換され、標準漢字は 2 バイトから 3 バイトに変換されるた
め、抽出対象表で半角仮名文字や標準漢字を使用している場合、変換後のデータ長が長くなってオーバフ
ローが発生することがあります。

オーバフローが発生した場合の対処 (EBCDIK/KEIS から UTF-8) を次の表に示します。

表 4-20 オーバフローが発生した場合の対処 (EBCDIK/KEIS から UTF-8)

発生場所	Datareplicator の動作	ユーザの対処
定数, 更新データ	SQL エラー (全バイト数の変換後に, データが定義長を超えた場合)	定義長を変更して, 反映処理を再実行してください。

注

識別子が 30 バイトを超えた場合, 又は更新データが 32,000 バイトを超えた場合には, Datareplicator は処理を続行できなくなります。このため, データ連動を開始する前に, 抽出側システムをチェックして, 識別子と更新データが制限値を超えないことを確認してください。

(f) 注意事項

反映側 DB が HiRDB で UTF-8 の場合, 表に NCHAR・NVARCHAR 型の列を作成できません。そのため, MCHAR など別のデータ型に格納する場合は, 列データ編集 UOC を使用して格納してください。

(7) EBCDIK から JIS8 への文字コードの変換規則

(a) 1 バイトコード

JIS8 コード (1 バイトコード) に変換します。文字集合指定で EBCDIK を指定した列には 1 バイトコードしか指定できないため, 次の現象が起こる可能性があります。

- すべて 1 バイトコードとみなして変換するため, 2 バイトコードが変換対象となる場合, 文字化けが発生します。
- 1 バイトコードの変換では, 変換前のデータ長と変換後のデータ長は変わりません。
- すべて 1 バイトコードとして処理されるため, 文字コード変換エラーは発生しません。

(8) JIS8 から EBCDIK への文字コードの変換規則

(a) 1 バイトコード

対応する EBCDIK の文字コードに変換されます。

(b) 2 バイトコード (標準文字コード)

2 バイトコードの 1 バイト目と 2 バイト目をそれぞれ 1 バイトコードとみなし, それぞれを EBCDIK の 1 バイトコード 2 文字に変換します。そのため, 次の現象が起こる可能性があります。

- 2 バイトコードを変換すると文字化けが発生します。
- 1 バイトコードの変換では, 変換前のデータ長と変換後のデータ長は変わりません。
- すべて 1 バイトコードとして処理されるため, 文字コード変換エラーは発生しません。

(9) JIS8/シフト JIS 又は EUC から、EBCDIK/KEIS への文字コードの変換規則

(a) 1 バイトコード

対応する EBCDIK/KEIS の文字コードに変換されます。

(b) 2 バイトコード (標準文字コード)

対応する EBCDIK/KEIS の文字コードに変換されます。

ただし、最後の文字が 2 バイトコードの先頭バイトの場合は、空白文字 ((40)₁₆) に変換されます。

(c) 2 バイトコード (外字)

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。

未定義の外字、及び文字コード変換用マッピングテーブルを作成していない場合は、空白文字 ((4040)₁₆) に変換されます。

(d) 空白文字

全角空白文字 ((8140)₁₆) は、対応する全角空白文字 ((A1A1)₁₆) に変換されます。半角空白文字 ((20)₁₆) は、対応する半角空白文字 ((40)₁₆) に変換されます。

(10) JIS8/シフト JIS から UTF-8 への文字コードの変換規則

(a) 1 バイトコード

- 仮名文字を除く 1 バイトコードは、対応する UTF-8 の文字コードに変換されます。
- 仮名文字は、3 バイトコードに変換されます。

(b) 2 バイトコード (標準文字コード)

対応する UTF-8 の文字コードに変換されます。

(c) 2 バイトコード (外字)

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。

未定義の外字、及び文字コード変換用マッピングテーブルを作成していない場合は、反映環境定義の `undefcode_cnv` オペランドの指定に従って、空白文字に変換されます。

(d) 空白文字

シフトコード中の空白文字は、変換されます。空白文字の変換規則 (JIS8/シフト JIS から UTF-8) を次の表に示します。

表 4-21 空白文字の変換規則 (JIS8/シフト JIS から UTF-8)

JIS8/SJIS の文字コード	UTF-8 の文字コード
全角の空白文字 (8140) ₁₆	全角の空白文字 (E38080) ₁₆
半角の空白文字 (20) ₁₆	半角の空白文字 (20) ₁₆

(e) オーバフローが発生した場合の対処

半角の仮名文字は 1 バイトから 3 バイトに変換され、標準漢字は 2 バイトから 3 バイトに変換されるため、抽出対象表で半角仮名文字や標準漢字を使用している場合、変換後のデータ長が長くなってオーバフローが発生することがあります。オーバフローが発生した場合の対処 (JIS8/シフト JIS から UTF-8) を次の表に示します。

表 4-22 オーバフローが発生した場合の対処 (JIS8/シフト JIS から UTF-8)

発生場所	Datareplicator の動作	ユーザの対処
定数, 更新データ	SQL エラー (全バイト数の変換後に、データが定義長を超えた場合)	定義長を変更して、反映処理を再実行してください。

注

識別子が 30 バイトを超えた場合、又は更新データが 32,000 バイトを超えた場合には、Datareplicator は処理を続行できなくなります。このため、データ連動を開始する前に、抽出側システムをチェックして、識別子と更新データが制限値を超えないことを確認してください。

(f) 注意事項

- 反映側 DB が HiRDB で UTF-8 の場合、表に NCHAR・NVARCHAR 型の列を作成できません。そのため、MCHAR など別のデータ型に格納する場合は、列データ編集 UOC を使用して格納してください。
- SJIS 漢字コードから Unicode (UCS2) へのマッピングには、次の表に示す 2 種類があります。

表 4-23 SJIS 漢字コードと Unicode での文字のマッピング

マッピング方式			内容
方式	変換対象	漢字範囲	
JIS 方式	SJIS~JIS X0221	JIS 第 1 水準	JIS X0221 で規定されたマッピングに従います。
		JIS 第 2 水準	
MS 方式	Windows 符号化文字集合~MS-Unicode	JIS 第 1 水準	Microsoft 社が規定したマッピングに従います。
		JIS 第 2 水準	
		ベンダ拡張文字	

MS 方式では、JIS 方式に比べベンダ拡張文字が追加されており、UCS2 に変換する際 JIS 方式と異なるコードポイントに設定されます。Datareplicator は、MS 方式で行います。

- NCHAR 型のデータを変換する場合、UTF-8 への変換後の長さが奇数のとき、変換後のデータに半角スペースが付与されます。collen メンバ（列データ長）には、変換後の（バイト長+1）/2 の値が設定されます。

UTF-8 への変換後の長さが偶数の場合、変換後のデータはそのままとし、collen メンバ（列データ長）には、変換後のバイト長/2 の値が設定されます。

- 抽出表の NCHAR 型の定義長 n が偶数の場合、反映表の MCHAR 型の定義長は $n*3$ となります。抽出表の NCHAR 型の定義長 n が奇数の場合、反映表の MCHAR 型の定義長は $(n*3) + 1$ となります。

(11) EUC から JIS8/シフト JIS への変換規則

(a) 1 バイトコード

対応するシフト JIS のコードに変換されます。

(b) 2 バイトコード（標準文字コード）

対応するシフト JIS のコードに変換されます。最後の文字が 2 バイトコードの先頭バイトである場合は、空白文字（ $(20)_{16}$ ）に変換されます。

(c) 2 バイトコード（外字）

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。未定義の外字、及び文字コード変換用マッピングテーブルを作成していない場合は、反映環境定義の undefcode_cnv オペランドの指定値に従って、空白文字に変換されます。

(d) 空白文字

全角の空白文字（ $(A1A1)_{16}$ ）は対応するコード（ $(8140)_{16}$ ）に変換されます。2 個連続する半角の空白文字（ $(20)_{16}$ ）は全角空白文字（ $(8140)_{16}$ ）に変換されます。1 個の半角の空白文字（ $(20)_{16}$ ）は対応するコード（ $(20)_{16}$ ）に変換されます。

(12) EUC から UTF-8 への文字コードの変換規則

(a) 1 バイトコード

仮名文字を除く 1 バイトコードは、対応する UTF-8 の文字コードに変換されます。

(b) 2 バイトコード（標準文字コード）

- 仮名文字は、3 バイトコードに変換されます。
- 対応する UTF-8 の文字コードに変換されます。

(c) 3バイトコード (外字)

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。

未定義の外字，及び文字コード変換用マッピングテーブルを作成していない場合は，反映環境定義の `undefcode_cnv` オペランドの指定に従って，空白文字に変換されます。

(d) 空白文字

シフトコード中の空白文字は，変換されます。空白文字の変換規則 (EUC から UTF-8) を次の表に示します。

表 4-24 空白文字の変換規則 (EUC から UTF-8)

EUC の文字コード	UTF-8 の文字コード
全角の空白文字 (A1A1) ₁₆	全角の空白文字 (E38080) ₁₆
半角の空白文字 (20) ₁₆	半角の空白文字 (20) ₁₆

(e) オーバフローが発生した場合の対処

半角の仮名文字は 2 バイトから 3 バイトに変換され，標準漢字は 2 バイトから 3 バイトに変換されるため，抽出対象表で半角仮名文字や標準漢字を使用している場合，変換後のデータ長が長くなってオーバフローが発生することがあります。オーバフローが発生した場合の対処 (EUC から UTF-8) を次の表に示します。

表 4-25 オーバフローが発生した場合の対処 (EUC から UTF-8)

発生場所	Datereplicator の動作	ユーザの対処
定数，更新データ	SQL エラー (全バイト数の変換後に，データが定義長を超えた場合)	定義長を変更して，反映処理を再実行してください。

注

識別子が 30 バイトを超えた場合，又は更新データが 32,000 バイトを超えた場合には，Datereplicator は処理を続行できなくなります。このため，データ連動を開始する前に，抽出側システムをチェックして，識別子と更新データが制限値を超えないことを確認してください。

(f) 注意事項

- 反映側 DB が HiRDB で UTF-8 の場合，表に NCHAR・NVARCHAR 型の列を作成できません。そのため，MCHAR など別のデータ型に格納する場合は，列データ編集 UOC を使用して格納してください。

- NCHAR 型のデータを変換する場合、UTF-8 への変換後の長さが奇数のとき、変換後のデータに半角スペースが付与されます。collen メンバ（列データ長）には、変換後の（バイト長+1）/2 の値が設定されます。

UTF-8 への変換後の長さが偶数の場合、変換後のデータはそのままとし、collen メンバ（列データ長）には、変換後のバイト長/2 の値が設定されます。

- 抽出表の NCHAR 型の定義長 n が偶数の場合、反映表の MCHAR 型の定義長は $n*3$ となります。抽出表の NCHAR 型の定義長 n が奇数の場合、反映表の MCHAR 型の定義長は $(n*3) + 1$ となります。

(13) UTF-8 から EUC 又は JIS8/シフト JIS への文字コードの変換規則

(a) 1 バイトコード

対応する各文字コードに変換されます。

(b) 2 バイトコード及び 3 バイトコード（標準漢字）

対応する UTF-8 の文字コードに変換されます。

(c) 3 バイトコード（外字）

ユーザが作成する文字コード変換用マッピングテーブルに従って変換されます。

未定義の外字、及び文字コード変換用マッピングテーブルを作成していない場合は、反映環境定義の undefcode_cnv オペランドの指定に従って、空白文字に変換されます。

(14) 文字コード変換の抑止

任意の抽出列に対して文字コードの変換抑止を指定すると、反映側 Datareplicator でその列の文字コード変換を抑止できます。文字コード変換の抑止の指定については、「[5.10 反映定義](#)」の更新情報フィールド定義の説明を参照してください。

(15) 各文字コード変換規則詳細

(a) JIS8/シフト JIS コードの変換規則

JIS8/シフト JIS コードの変換規則を次の表に示します。

表 4-26 JIS8/シフト JIS コードの変換規則

1byte	2byte	3byte	コード変換規則
0x00-0x80	—	—	JIS8 と認識し、対応するコードに変換します。
0x81-0x9F	0x40-0xFC (0x7F を除く)	—	SJIS（漢字）と認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。

1byte	2byte	3byte	コード変換規則
	—	—	未完全コードと認識し、変換しないでスキップします。
0xA0-0xDF	—	—	JIS8 と認識し、対応するコードに変換します。
0xE0-0xEF	0x40-0xFC (0x7F を除く)	—	SJIS (漢字) と認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。
	—	—	未完全コードと認識し、変換しないでスキップします。
0xF0-0xFC	0x40-0xFC (0x7F を除く)	—	SJIS (外字) と認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。
	—	—	未完全コードと認識し、変換しないでスキップします。
0xFD-0xFF	—	—	JIS8 と認識し、対応するコードに変換します。

(b) EUC コードの変換規則

EUC コードの変換規則を次の表に示します。

表 4-27 EUC コードの変換規則

1byte	2byte	3byte	コード変換規則
0x00-0x8D	—	—	コードセット 0 と認識し、対応するコードに変換します。
0x8E	0xA0-0xFF	—	コードセット 2 (仮名文字) と認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。
	—	—	未完全コードと認識し、変換しないでスキップします。
0x8F	0xA1-0xFE	0xA1-0xFE	コードセット 3 (外字) と認識し、対応するコードに変換します。
		上記以外	undefcode_cnv の指定によって変換します。
		—	未完全コードと認識し、変換しないでスキップします。
	上記以外	0xA1-0xFE	undefcode_cnv の指定によって変換します。
		上記以外	
		—	未完全コードと認識し、変換しないでスキップします。

1byte	2byte	3byte	コード変換規則
	—	—	
0x90-0x9F	—	—	コードセット 0 と認識し、対応するコードに変換します。
0xA0	—	—	0x20 に変換します。
0xA1-0xFE	0xA1-0xFE	—	コードセット 1 (漢字) と認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。
	—	—	未完全コードと認識し、変換しないでスキップします。

(c) EBCDIK/KEIS コードの変換規則

EBCDIK/KEIS コードの変換規則を次の表に示します。

表 4-28 EBCDIK/KEIS コードの変換規則

1byte	2byte	3byte	コード変換規則		
半角シフト中	0x00-0x09	—	—	対応するコードに変換します。	
	0x0A	0x41	—	—	半角シフト切り替えて、変換しません。
		0x42	—	—	全角シフト切り替えて、変換しません。
		上記以外	—	—	第 1 バイト, 第 2 バイトを 1 バイトコードとして変換します。
		—	—	—	未完全コードと認識し、変換しないでスキップします。
	0x0B-0xFF	—	—	—	対応するコードに変換します。
全角シフト中	0x00-0x40	—	—	—	第 1 バイト, 第 2 バイトを 1 バイトコードとして変換します。
	0x41-0xA0	0xA1-0xFE	—	—	対応するコードに変換します。
		上記以外	—	—	undefcode_cnv の指定によって変換します。
		—	—	—	未完全コードと認識し、変換しないでスキップします。
	0xA1-0xFE	0xA1-0xFE	—	—	対応するコードに変換します。
		上記以外	—	—	undefcode_cnv の指定によって変換します。

1byte		2byte	3byte	コード変換規則
		—	—	不完全コードと認識し、変換しないでスキップします。
	0xFF	0x01-0xFF	—	undefcode_cnv の指定によって変換します。
		—	—	不完全コードと認識し、変換しないでスキップします。

(d) UTF-8 コードの変換規則

UTF-8 コードの変換規則を次の表に示します。

表 4-29 UTF-8 コードの変換規則

1byte	2byte	3byte	コード変換規則
0x00~0x7F	—	—	1 バイトコードと認識し、対応するコードに変換します。
0x80~0xBF	0x00~0xFF	—	0x20 に変換します。
	—	—	
0xC2~0xDE	0x80~0xFF	—	2 バイトコードと認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。
	—	—	不完全コードと認識し、変換しないでスキップします。
0xDF	0x80~0xBF	—	2 バイトコードと認識し、対応するコードに変換します。
	上記以外	—	undefcode_cnv の指定によって変換します。
	—	—	不完全コードと認識し、変換しないでスキップします。
0xE0	0xA0~0xFF	0x80~0xFF	3 バイトコードと認識し、対応するコードに変換します。
		上記以外	undefcode_cnv の指定によって変換します。
		—	不完全コードと認識し、変換しないでスキップします。
	上記以外	0x80~0xFF	undefcode_cnv の指定によって変換します。
		上記以外	
	—	—	不完全コードと認識し、変換しないでスキップします。

1byte	2byte	3byte	コード変換規則
0xE1~0xEE	0x80~0xFF	0x80~0xFF	3バイトコードと認識し、対応するコードに変換します。
		上記以外	undefcode_cnv の指定によって変換します。
		—	未完全コードと認識し、変換しないでスキップします。
	上記以外	0x80~0xFF	undefcode_cnv の指定によって変換します。
		—	未完全コードと認識し、変換しないでスキップします。
		—	—
0xEF	0x80~0xBF	0x80~0xBF	3バイトコードと認識し、対応するコードに変換します。
		上記以外	undefcode_cnv の指定によって変換します。
		—	未完全コードと認識し、変換しないでスキップします。
	上記以外	0x80~0xBF	undefcode_cnv の指定によって変換します。
		上記以外	—
		—	未完全コードと認識し、変換しないでスキップします。
上記以外	—	—	undefcode_cnv の指定によって変換します。

4.3.6 繰返し列の設計

繰返し列の設計方法について説明します。

(1) 繰返し列データの抽出の制限

繰返し列を抽出するときには、抽出側 Datareplicator では特別な定義は必要ありません。ただし、次に示す制限があります。

- 繰返し列を抽出するときは、一つの列として操作してください。繰返し列を構成する要素ごとには抽出できません。
- 繰返し列をマッピングキーの対象にはできません。

(2) 繰返し列データの反映の制限

UPDATE の SET 句で更新された繰返し列データを反映するときに、次に示す条件に該当すると、反映側 Datareplicator で SQL エラー (SQLCODE=-129) になって反映できなくなります。

- (マッピングキーの数+抽出対象表の繰返し列以外の抽出列の数+ UPDATE の SET 句に指定した繰返し列に該当する更新項目数) > 30000

マッピングキーの数は必ず 1 以上なので、反映できる SET 句の繰返し列更新項目数は最大 29999 です。抽出側システムで繰返し列のすべての要素を更新するなど、大量の要素データを 1SQL で更新するときは、各要素単位に更新項目を指定するのではなく、列単位で更新することをお勧めします。

(3) 繰返し列をデータ連動するときの反映定義

繰返し列を反映するときの反映定義は、従来通り load 文で繰返し列を含む表を反映対象として定義します。更新情報フィールド定義で繰返し列に対して定数指定 (format 文の name 句で const 指定) するときは、NULL 定数だけ指定できます。NULL 定数以外の値を指定すると、Datareplicator ではエラーにしないで、指定された定数でそのまま反映されます (反映側 HiRDB でデータ互換性エラーになります)。

反映定義の load 文に繰返し列を含む表を指定すると、timestamp を指定して時系列情報表を作成できません。繰返し列を含む表に対して timestamp を指定すると、定義解析エラーとして該当するデータ連動識別子 (dsid) での反映処理が停止します。

反映定義の load 文に繰返し列を含む表を指定しても、sqlconvopt1, sqlconvopt2 を指定してマージ表を作成できます。ただし、要素指定の UPDATE を INSERT に切り替えるときは、不足している要素データは NULL 値で補完されます (「3.3.11 (3) 繰返し列を含む表へのマージ表反映」を参照してください)。

(4) 繰返し列データの文字コード変換

抽出側 HiRDB の文字コード体系と反映側 HiRDB の文字コード体系が異なるときは、繰返し列データの各要素データに対して文字コード変換が実行されます。文字コード変換については、「4.3.5 文字コードの対応の設計」を参照してください。

(5) 繰返し列データの UOC インタフェース

繰返し列データは、反映情報編集 UOC、列データ編集 UOC で加工できます。ただし、列データ編集 UOC でのデータの受け渡しは要素データごとになります (1 回の列データ編集 UOC 関数の呼び出しで、一つの要素データを列データ編集 UOC 関数へ渡します)。このため、要素数を列データ編集 UOC で増やしたり減らしたりできません。

(6) 要素数が不一致のときの処理

HiRDB では繰返し列に対して UPDATE の SET や DELETE にデータが存在しない要素番号を指定して実行すると、存在しない繰返し列の要素に対する更新が無視されます (このときの SQLCODE は 0, SQLWARN7 に「W」が設定されます)。

通常データ連動では、抽出側と反映側の繰返し列の要素数が不一致になることはありません。ただし、HiRDB の操作でマージ表の反映や反映先の表を更新するなどが原因で要素数が不一致となったときは、反映環境定義 skip_mvcelmwarn オペランドと sqlerr_skip_info オペランドの指定に従って処理が実行されます。

skip_mvcelmwarn		sqlerr_skip_info			
		output	msgoutput	sqloutput	nooutput
true	処理	正常（反映処理を続行）			
	出力メッセージ	Warning	Warning	なし	なし
	未反映出力	出力	なし	出力	なし
false	処理	エラー（トランザクションをロールバックして反映を中止）			
	出力メッセージ	Error			
	未反映出力	出力			

4.3.7 時系列情報表の作成

時系列情報表を作成する手順について説明します。

(1) 取得できる時系列情報

反映表定義の to 句で timestamp を指定すると、更新情報を更新の履歴として時系列順に取得できます。ただし、更新の種別によっては時系列情報を取得できないことがあります。時系列情報を取得できないときは、null 値になります。

取得できる時系列情報の内容と取得可否を次の表に示します。

表 4-30 取得できる時系列情報の内容と取得可否

取得情報	内容	可否			
		更新	挿入	削除	PURGE
抽出日付	抽出側システムで更新情報が更新ジャーナルに出力された日付	○	○	○	○
抽出時間	抽出側システムで更新情報が更新ジャーナルに出力された時間	○	○	○	○
抽出時刻印	抽出側システムで更新情報が更新ジャーナルに出力された日付及び時間	○	○	○	○
反映日付	反映側システムで更新情報が反映側 DB に反映された日付	○	○	○	○
反映時間	反映側システムで更新情報が反映側 DB に反映された時間	○	○	○	○
反映時刻印	反映側システムで更新情報が反映側 DB に反映された日付及び時間	○	○	○	○
反映種別	反映の種別が次のように取得されます。 <ul style="list-style-type: none"> • 更新 : upd • 挿入 : ins 	○	○	○	○

取得情報	内容	可否			
		更新	挿入	削除	PURGE
	<ul style="list-style-type: none"> 削除 : del PURGE : purge 				
マッピングキー	更新対象の特定に使ったキー値	○	○	○	—
更新情報	更新したデータ	○	○	—	—
SQL 通番	時系列情報単位に割り当てたシーケンシャルな番号	○	○	○	○

(凡例)

○：取得できます。

—：NULL 値になります。

(2) 時系列情報を取得する表の定義

更新情報、反映日付、反映時間などの時系列情報を取得するためには、反映対象にする HiRDB のデータベース上に、これらの情報を取得するための表（時系列情報表）を作成しておく必要があります。時系列情報表を作成するときには、表名と列構成（列数、列順序、列名）は任意に指定できます。ただし、列の属性は取得する情報に合わせて定義する必要があります。また、取得する情報の種類によっては、更新情報フィールド定義の const 句で定数を指定します。

時系列情報表の列の属性と const 句での指定を次の表に示します。

表 4-31 時系列情報表の列の属性と const 句での指定

取得する情報	列の属性	更新情報フィールド定義の const 句での指定
抽出日付	DATE	extract_date
抽出時間	TIME	extract_time
抽出時刻印	TIMESTAMP [(p)] *1, p = 0, 2, 4, 6	extract_timestamp
反映日付	DATE	reflect_date
反映時間	TIME	reflect_time
反映時刻印	TIMESTAMP [(p)] *1, p = 0, 2, 4, 6	reflect_timestamp
反映種別	CHAR(5)	reflect_kind
マッピングキー	マッピングキーを格納できる属性	—
更新情報	更新情報を格納できる属性	—
SQL 通番	INTEGER, SMALLINT, 位取り 0 の DECIMAL	sequence [認可識別子*2,]順序数生成子識別子

(凡例)

－：該当しません。

注※1

p は表示する秒数の小数点以下のけた数を表す整数です。例えば、p = 2 の場合、小数点第 2 位の秒数までを時刻として取得します。

抽出時刻印については、p の指定内容に関わらず、抽出側 DB がメインフレームの場合にだけ小数点以下の時刻を取得します。また、そのときに取得する秒数は、小数点第 2 位までになります。

注※2

認可識別子を省略すると、反映システム定義で定義した認可識別子が仮定されます。

時系列情報表を利用する場合には、次の点に注意してください。

- 時系列情報は、HiRDB の検索機能 (SELECT コマンド) によって参照できます。ただし、抽出側 DB の更新順に取得されないことがあるため、抽出側 DB の更新順に検索したい場合は、時系列情報表に SQL 通番列を指定し、SQL 通番情報を取得するようにしてください。
また検索時には、SQL 通番列でソートの指定をしてください。
- 時系列情報表には、更新情報、反映日付、反映時間などの情報が時系列順に蓄積され続けます。このため、不要になった情報を、DELETE 又は PURGE TABLE を発行して、時系列情報表から削除する必要があります。

(3) 時系列情報表へ反映するデータの注意事項

- BLOB 分割取得データを時系列情報表へ反映した場合、反映するデータは BLOB 分割取得データだけになります。
- 後方削除更新をしている BLOB 型又は BINARY 型のデータは、時系列情報表には反映しないでください。抽出側 DB と反映側 DB で不整合が発生します。

(4) SQL 通番を取得する場合の注意事項

- SQL 通番を取得する場合、事前に順序数生成子を作成しておく必要があります。
順序数生成子を作成する際の推奨値を次に示します。

オペランド	推奨値	説明
[認可識別子] 順序数生成子識別子	任意	任意の名称を指定してください。
FOR PUBLIC USAGE	FOR PUBLIC USAGE	Replicator のプロセスが採番要求を行うため、順序数生成子作成者以外でも使用できるように、FOR PUBLIC USAGE を指定してください。
IN 順序数生成子 格納 RD エリア	時系列情報表と同じサーバに存在する RD エリア名	時系列情報表と異なるサーバに順序数生成子を作成すると、採番ごとにサーバ間で通信が発生し、性能が劣化します。そ

オペランド	推奨値	説明
		のため、時系列情報表と同じサーバに存在する RD エリア名を推奨します。
順序数生成子データ型オプション	DECIMAL(38)	循環が発生する要因を少なくするため、整数で最大値が指定できる、DECIMAL(38)を推奨します。
順序数生成子開始オプション	任意	SQL 通番の使用用途に合わせて値を設定してください。
順序数生成子増分オプション	任意	
順序数生成子最大値オプション	任意	
順序数生成子最小値オプション	任意	
順序数生成子循環オプション	NO CYCLE	SQL 通番が重複しないように NO CYCLE を推奨します。
順序数生成子ログ出力間隔オプション	省略	SQL 通番の抜けが発生しないようにオプションを省略または 1 を指定してください。

- データ連携中に SQL 通番のリセット（順序数生成子の削除後，再定義）は行わないでください。データ連携中に SQL 通番のリセットを行うと SQL エラーが発生し，連携が停止します。
- 時系列情報表のクリアを行う際に，SQL 通番の循環を発生させないよう，SQL 通番のリセット（順序数生成子の削除後，再定義）の実施を推奨します。

4.4 抽出側システムと反映側システムとの対応の設計

抽出側システムと反映側システムとの対応を設計する手順について説明します。

4.4.1 HiRDB から HiRDB へデータ連動する場合の設計

HiRDB から HiRDB へデータ連動する場合、抽出側システムと反映側システムの対応は抽出側 HiRDB がシングルサーバの場合とパラレルサーバの場合とで異なります。

抽出側 HiRDB がシングルサーバの場合の対応を図 4-16 に、抽出側 HiRDB がパラレルサーバの場合の対応を図 4-17 に示します。

図 4-16 HiRDB から HiRDB へデータ連動する場合の、抽出側システムと反映側システムとの対応（抽出側 HiRDB がシングルサーバの場合）

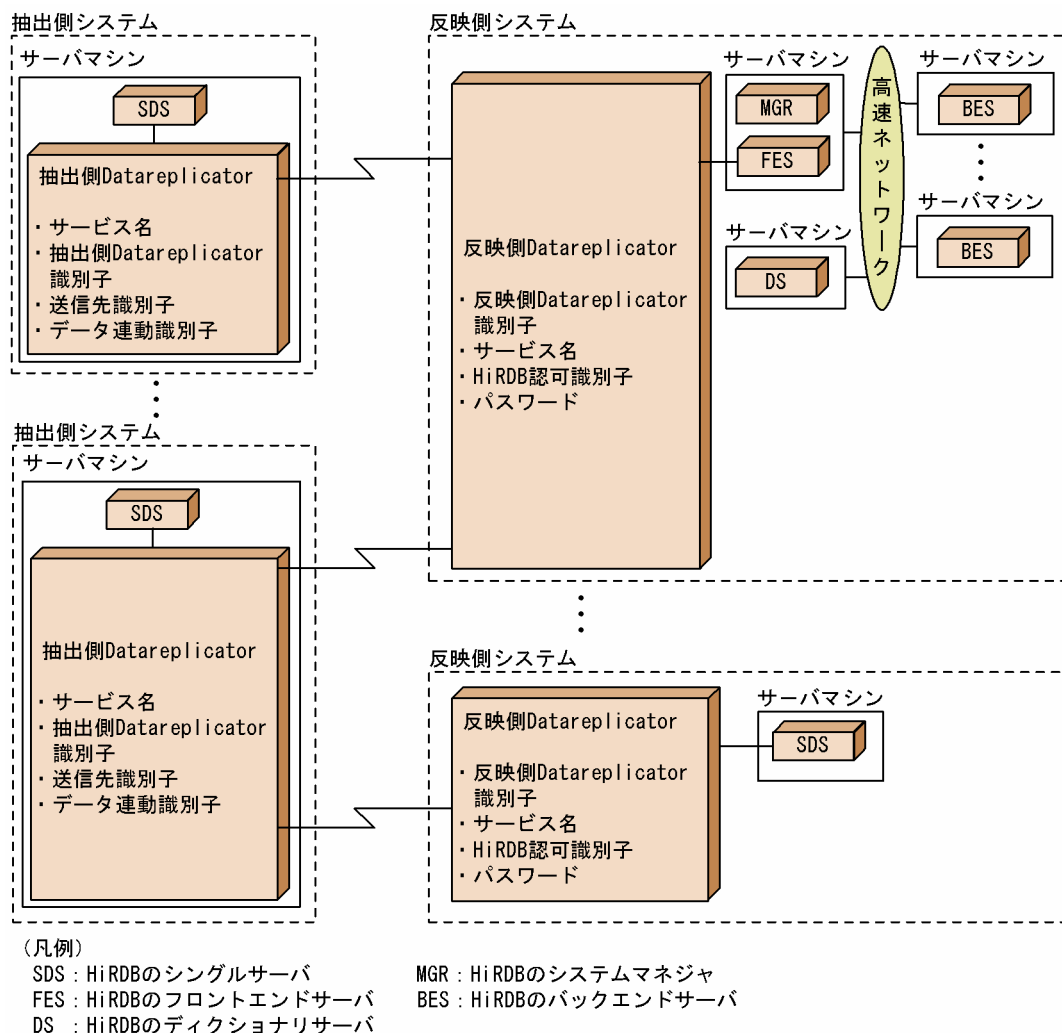
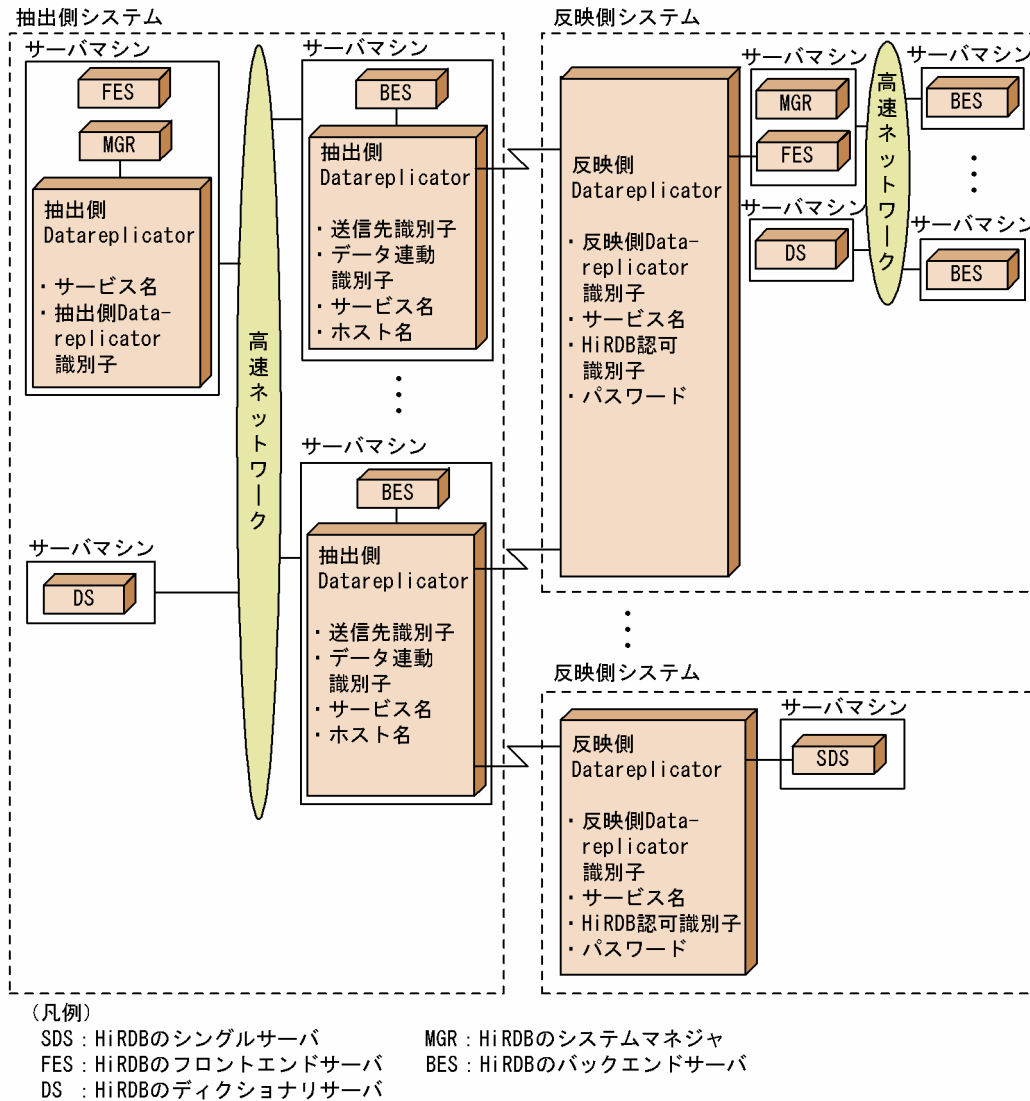


図 4-17 HiRDB から HiRDB へデータ連動する場合の、抽出側システムと反映側システムとの対応（抽出側 HiRDB が平行サーバの場合）



(1) 抽出側 HiRDB と抽出側 Datareplicator との対応

抽出側 HiRDB と抽出側 Datareplicator との対応数について説明します。

(a) 抽出側 HiRDB がシングルサーバの場合の対応数

抽出側 HiRDB（シングルサーバ）：抽出側 Datareplicator = 1 : 1

一つの抽出側 HiRDB に対して、抽出側 Datareplicator は一つだけ稼働できます。また、一つの抽出側 Datareplicator に対して、抽出側 HiRDB は一つだけ稼働できます。

(b) 抽出側 HiRDB がパラレルサーバの場合の対応数

抽出側 HiRDB (パラレルサーバ) : 抽出側 Datareplicator = 1 : n[※]

注※

n は、1 + バックエンドサーバの存在するサーバ数

一つの抽出側 HiRDB に対して、抽出側 Datareplicator は、抽出側 HiRDB のシステムマネージャが存在するサーバで一つと、各バックエンドサーバが存在するサーバで一つずつ稼働します。抽出側 Datareplicator は、抽出対象の DB がないバックエンドサーバが存在するサーバマシンでも、一つずつ稼働します。

逆に、一つの抽出側 Datareplicator に対して、抽出側 HiRDB は一つだけ稼働できます。

(2) 抽出側 Datareplicator と反映側 Datareplicator との対応

抽出側 Datareplicator と反映側 Datareplicator とを対応付けるために、次の項目を設計する必要があります。

(a) 対応数

抽出側 Datareplicator : 反映側 Datareplicator = 1 : m

抽出側 Datareplicator : 反映側 Datareplicator = n : 1

変数	UNIX	Windows
m	1~64 [※]	1~63 [※]
n	1~128	1~63

注※

sendcontrol オペランドに sendmst を指定すると 1~4096 の範囲になります。

一つの抽出側 Datareplicator から、複数の反映側 Datareplicator に更新情報を送信できます。HiRDB がパラレルサーバの場合には、バックエンドサーバが存在するサーバの抽出側 Datareplicator ごとに、複数の反映側 Datareplicator に更新情報を送信できます。

逆に、一つの反映側 Datareplicator では、複数の抽出側 Datareplicator からの更新情報を受信できます。さらに、一つの反映側 HiRDB に複数の反映側 Datareplicator を稼働させると、128 個以上 (Windows 版では 63 個以上) の抽出側システム (データ連動識別子) から更新情報を受信できるようになります。ただし、次の不具合が起こるおそれがあります。

- 反映処理の排他エラーが発生する
- 更新情報を反映する順番が保証されない

(b) 送信先識別子

抽出側 Datareplicator で更新情報の送信先を識別するための識別子を、送信先識別子といいます。抽出システム定義の `sendidxx` オペランド又は `sendidxxxx` オペランドで送信先になるすべての送信先識別子を指定し、抽出定義の送信文で、どの送信先識別子にどの更新情報を送信するのかを指定します。

送信先識別子ごとに送信環境定義を作成し、送信環境定義の `sendhdsid` オペランドに反映側の識別子を指定することで、送信先を識別します。また、抽出側 Datareplicator では、この送信先識別子の単位に送信処理を起動したり、送信処理の縮退を指定したりできます。

なお、`sendidxx` 又は `sendidxxxx` オペランドは、昇順に連続して指定することが必要です。運用開始後に、`sendidxx` 又は `sendidxxxx` オペランドの指定を追加する場合には、抽出側及び反映側の環境の初期化が必要になります。将来の送信先の増加に備えたい場合には、`sendidxx` 又は `sendidxxxx` オペランドで欠番指定（送信先識別子として**を指定する）を定義してください。欠番指定使用時は、該当の送信先を初期化するだけで、該当の送信先とのデータ連動を開始できます（抽出側及び反映側全体の環境の初期化は不要になります）。

(c) データ連動識別子

データ連動する抽出側システムと反映側システムとを対応付けるための識別子を、データ連動識別子といいます。データ連動識別子は、データ連動する抽出側システムと反映側システムとで同じ識別子を指定します。また、反映側システムでは一意になる必要があります。抽出側 Datareplicator では、抽出環境定義の `dsid` オペランドで指定します。反映側 Datareplicator では、反映システム定義の `dsidxxx` オペランドで指定します。

抽出側システムが HiRDB の場合、データ連動識別子は抽出単位に指定します。抽出単位とは、更新情報の抽出、送信を実現する単位です。抽出側 HiRDB がシングルサーバの場合には、一つの抽出側 Datareplicator で抽出、送信を実現しています。このため、抽出側 HiRDB がシングルサーバの場合には、一つの抽出側 Datareplicator が一つの抽出単位になります。一方、抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバごとに存在する抽出側 Datareplicator で抽出、送信を実現しています。このため、抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバごとに存在する抽出側 Datareplicator が、それぞれ抽出単位になります。

なお、`dsidxxx` オペランドは、昇順に連続して指定することが必要です。運用開始後に、`dsidxxx` オペランドの指定を増減する場合には、抽出側及び反映側の環境の初期化が必要になります。将来の抽出元の増加に備えたい場合には、`dsidxxx` で欠番指定（データ連動識別子として**を指定する）を定義してください。欠番指定使用時は、追加する抽出元の環境を初期化するだけで、該当の抽出元とのデータ連動を開始できます（抽出側及び反映側全体の環境の初期化は不要になります）。

(d) 抽出側 Datareplicator 識別子

抽出側 Datareplicator を識別するための識別子を抽出側 Datareplicator 識別子といいます。抽出側 Datareplicator 識別子は、抽出システム定義の `hdeid` オペランドで指定します。

(e) 反映側 Datareplicator 識別子

反映側 Datareplicator を識別するための識別子を反映側 Datareplicator 識別子といいます。反映側 Datareplicator 識別子は、反映システム定義の hdsid オペランドで指定します。

抽出側 Datareplicator では、更新情報の送信先になる反映側 Datareplicator 識別子を、送信環境定義の sendhdsid オペランドで指定します。

(f) サービス名

抽出側 Datareplicator と反映側 Datareplicator とで更新情報を送受信するために、次に示す services ファイルに通信のサービス名を追加する必要があります。

- UNIX 版 Datareplicator の場合：/etc/services
- Windows 版 Datareplicator の場合：Windows のシステムディレクトリ¥drivers¥etc¥services

この通信のサービス名を、抽出側 Datareplicator の送信環境定義の hdeservice オペランド、及び反映側 Datareplicator の hdsservice オペランドで指定します。

(g) ホスト名

更新情報を送信するため、抽出側システムの hosts ファイルにホスト名を登録する必要があります。

- UNIX 版 Datareplicator の場合：/etc/hosts
- Windows 版 Datareplicator の場合：Windows のシステムディレクトリ¥drivers¥etc¥hosts

このホスト名を、送信環境定義の hdehost オペランドで指定します。

(3) 反映側 Datareplicator と反映側 HiRDB との対応

反映側 Datareplicator と反映側 HiRDB とを対応付けるために、次の項目を設計する必要があります。

(a) 対応数

反映側 Datareplicator : 反映側 HiRDB = n : 1

一つの反映側 Datareplicator に対して、反映側 HiRDB は一つだけ稼働できます。

逆に一つの反映側 HiRDB に対して、反映側 Datareplicator は複数稼働できます。

(b) HiRDB 認可識別子とパスワード

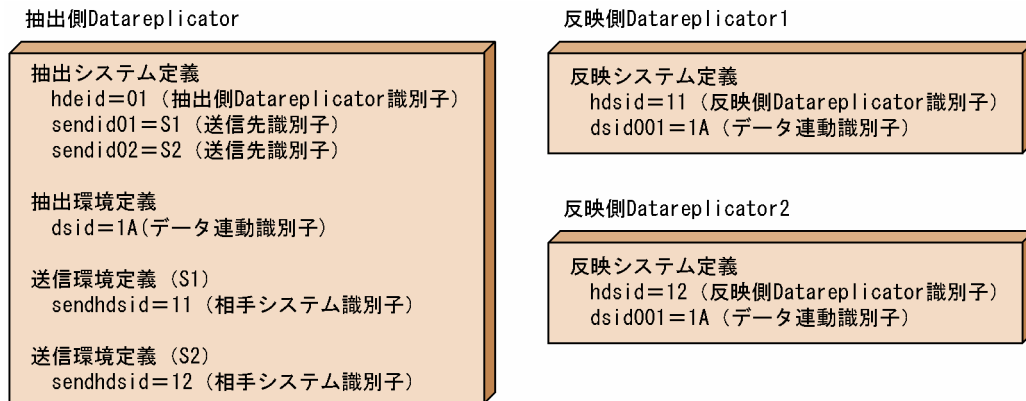
反映側 Datareplicator は反映側 HiRDB の IUAP として、SQL 文を発行し、反映を実現します。このため、反映側 HiRDB と接続するための HiRDB 認可識別子とパスワードの指定が必要になります。HiRDB 認可識別子とパスワードは、反映システム定義の hirdbusr オペランドで指定します。反映システム定義については、「[5.8 反映システム定義](#)」を参照してください。

反映側 Datareplicator の起動時に、パスワードを標準入力から入力する運用もできます。

(4) 識別子の定義例

一つの抽出側 Datareplicator から、二つの反映側 Datareplicator にデータ連動する場合の識別子の定義例を次の図に示します。

図 4-18 識別子の定義例



説明

- 抽出側 Datareplicator 識別子と反映側 Datareplicator 識別子を、それぞれ抽出システム定義の hdeid オペランドと反映システム定義の hdsid オペランドに指定します。
- データ連動識別子を抽出環境定義の dsid オペランドと反映システム定義の hdsid オペランドに指定します。データ連動するシステム同士は、同じデータ連動識別子を指定します。
- 抽出システム定義の sendidxx オペランドに送信先識別子を指定します。送信先識別子ごとに送信環境定義を作成し、sendhdsid オペランドに相手システム識別子 (反映側 Datareplicator 識別子) を指定します。

4.4.2 HiRDB からメインフレーム側 DB へデータ連動する場合の設計

HiRDB からメインフレーム側 DB (XDM/RD E2) へデータ連動する場合、抽出側システムと反映側システムの対応は抽出側 HiRDB がシングルサーバの場合と平行サーバの場合とで異なります。

抽出側 HiRDB がシングルサーバの場合の対応を図 4-19 に、抽出側 HiRDB が平行サーバの場合の対応を図 4-20 に示します。

図 4-19 HiRDB からメインフレーム側 DB へデータ連動する場合の、抽出側システムと反映側システムとの対応（抽出側 HiRDB がシングルサーバの場合）

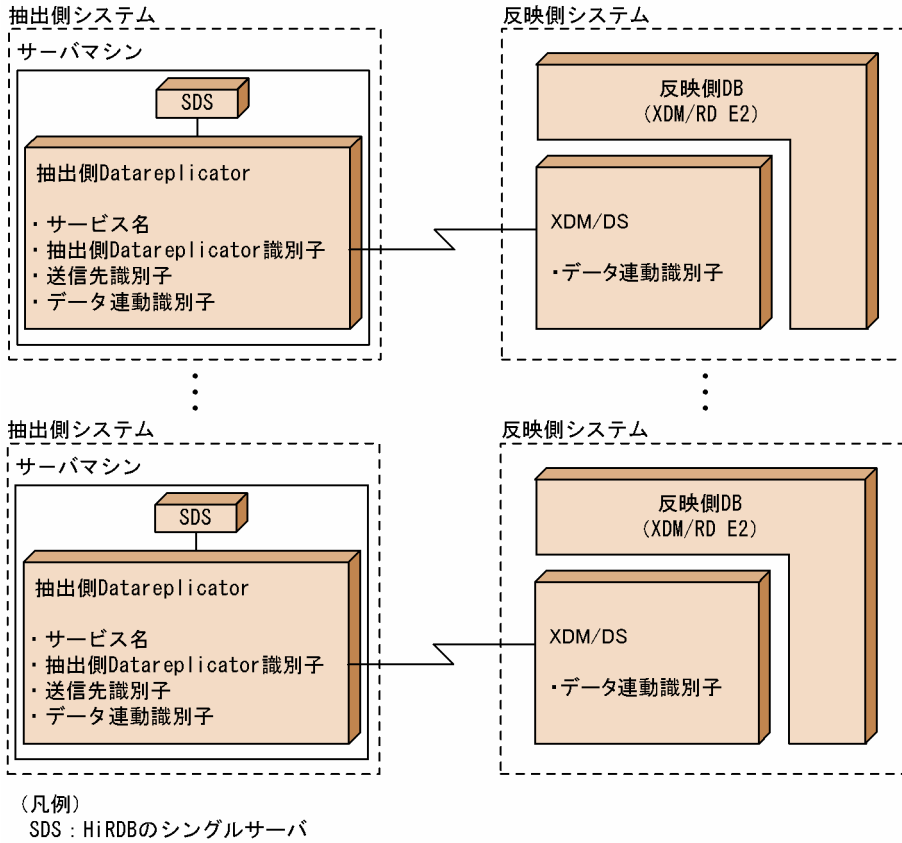
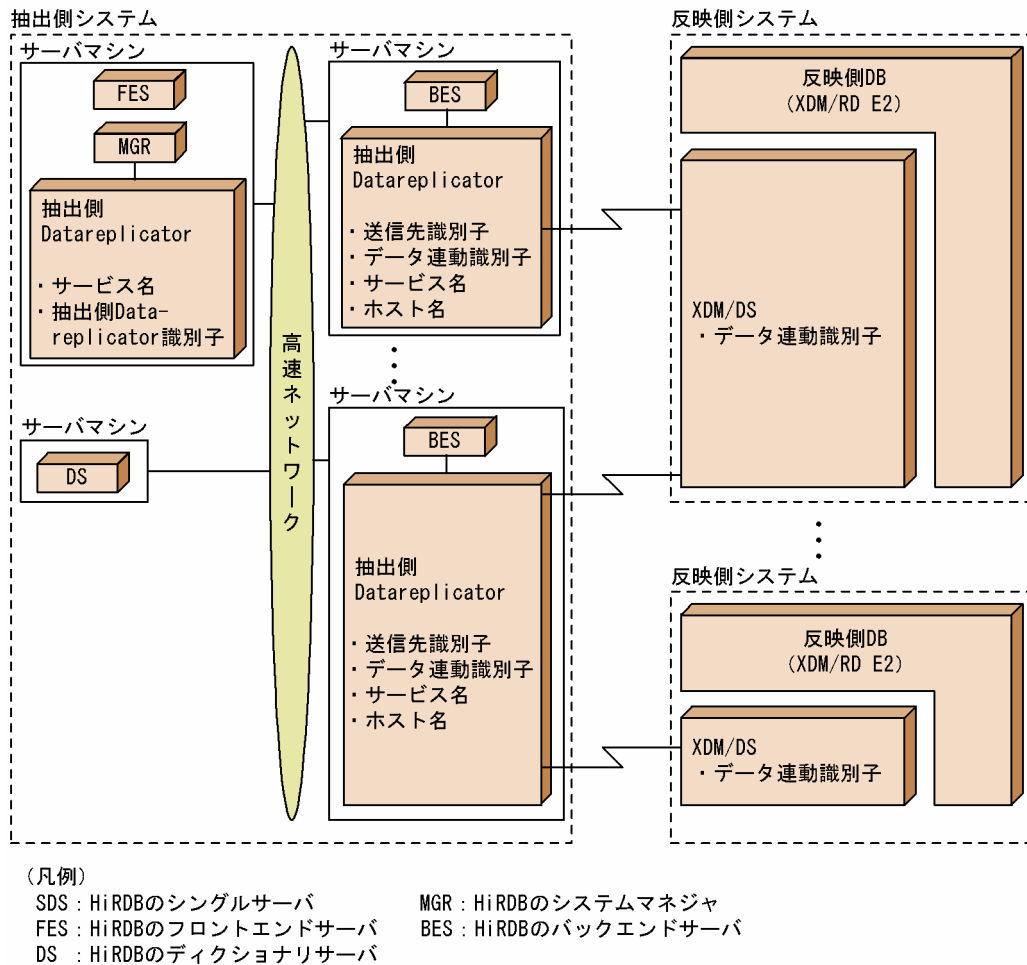


図 4-20 HiRDB からメインフレーム側 DB へデータ連動する場合の、抽出側システムと反映側システムとの対応（抽出側 HiRDB がパラレルサーバの場合）



(1) 抽出側 HiRDB と抽出側 Datareplicator との対応

抽出側 HiRDB と抽出側 Datareplicator との対応については、「4.4.1(1) 抽出側 HiRDB と抽出側 Datareplicator との対応」を参照してください。

(2) 抽出側 Datareplicator と XDM/DS との対応

抽出側 Datareplicator と XDM/DS とを対応付けるために、次の項目を設計する必要があります。

ここでは、抽出側 Datareplicator の指定について説明します。XDM/DS の各項目の指定内容については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

(a) 対応数

抽出側 Datareplicator : XDM/DS = 1 : m

抽出側 Datareplicator : XDM/DS = 1 : 1

変数	UNIX	Windows
m	1~64*	1~63*

注※

sendcontrol オペランドに sendmst を指定すると 1~4096 の範囲になります。

一つの抽出側 Datareplicator から、複数の反映側 XDM/DS に更新情報を送信できます。HiRDB がパラレルサーバの場合には、バックエンドサーバが存在するサーバの抽出側 Datareplicator ごとに、複数の反映側 XDM/DS に更新情報を送信できます。

逆に、一つの反映側 XDM/DS では、1 個の抽出側 Datareplicator からの更新情報を受信できます。さらに、一つの反映側 XDM/DS に複数の XDM/DS を稼働させると、1 個以上の抽出側システム（データ連動識別子）から更新情報を受信できるようになります。ただし、次の不具合が起こるおそれがあります。

- 反映処理の排他エラーが発生する
- 更新情報を反映する順番が保証されない

(b) 送信先識別子

送信先識別子の詳細については、「[4.4.1\(2\)\(b\) 送信先識別子](#)」を参照してください。

(c) データ連動識別子

データ連動識別子の詳細については、「[4.4.1\(2\)\(c\) データ連動識別子](#)」を参照してください。

(d) 抽出側 Datareplicator 識別子

抽出側 Datareplicator を識別するための識別子を抽出側 Datareplicator 識別子といいます。抽出側 Datareplicator 識別子は、抽出システム定義の hdeid オペランドで指定します。

(e) 反映側の識別子

XDM/DS での反映側の識別子の指定については、マニュアル「[VOS3 XDM データ連動機能 XDM/DS 解説・定義](#)」を参照してください。

(f) サービス名

抽出側 Datareplicator と XDM/DS とで更新情報を送受信するために、次に示す services ファイルに通信のサービス名を追加する必要があります。

- UNIX 版 Datareplicator の場合：/etc/services
- Windows 版 Datareplicator の場合：Windows のシステムディレクトリ¥drivers¥etc¥services

この通信のサービス名を、抽出側 Datareplicator の送信環境定義の hdeservice オペランドに指定します。

(g) ホスト名

更新情報を送信するために、抽出側システムの hosts ファイルにホスト名を登録する必要があります。

- UNIX 版 Datareplicator の場合：/etc/hosts
- Windows 版 Datareplicator の場合：Windows のシステムディレクトリ¥drivers¥etc¥hosts

このホスト名を、送信環境定義の hdehost オペランドで指定します。

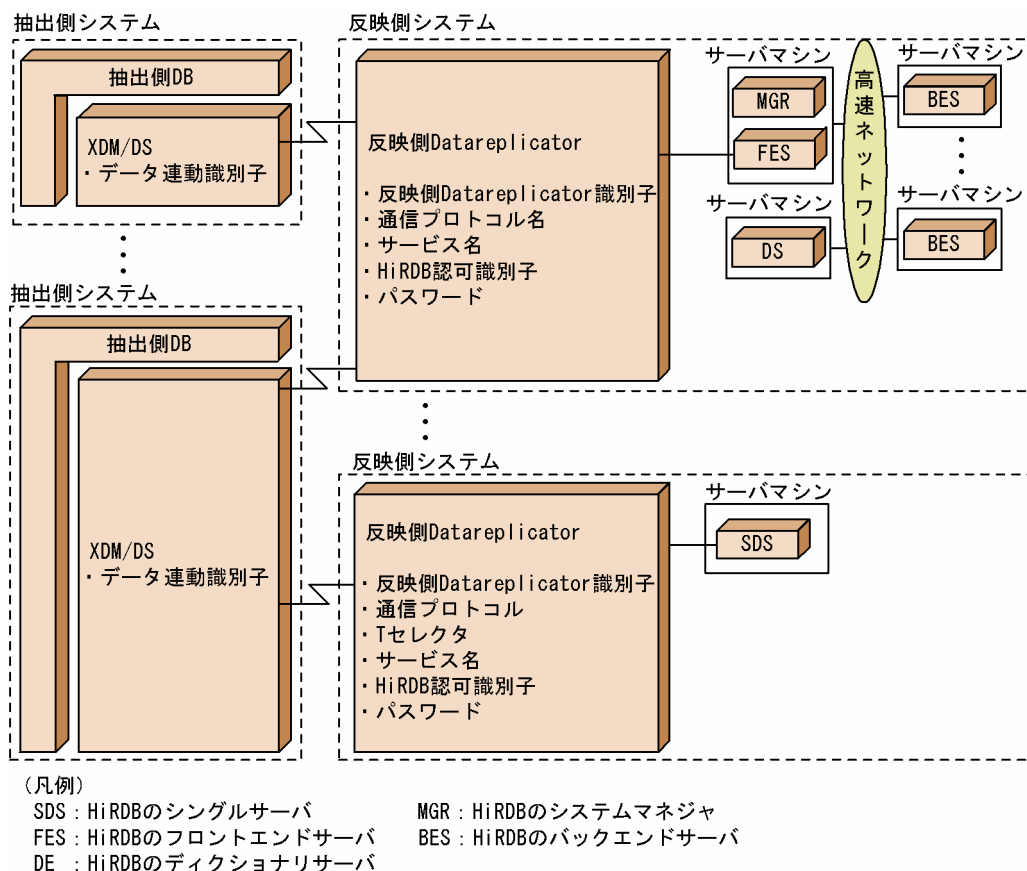
(3) XDM/DS と反映側 DB との対応

XDM/DS と反映側 DB との対応については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

4.4.3 メインフレーム側 DB から HiRDB へデータ連動する場合の設計

メインフレーム側 DB (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) から HiRDB へデータ連動する場合の、抽出側システムと反映側システムの対応を次の図に示します。

図 4-21 メインフレーム側 DB から HiRDB へデータ連動する場合の、抽出側システムと反映側システムとの対応



(1) 抽出側 DB と XDM/DS との対応

抽出側 DB と XDM/DS との対応については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

(2) XDM/DS と反映側 Datareplicator との対応

XDM/DS と反映側 Datareplicator とを対応付けるために、次の項目を設計する必要があります。

ここでは、反映側 Datareplicator の指定について説明します。XDM/DS の各項目の指定内容については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

(a) 対応数

XDM/DS : 反映側 Datareplicator = 1 : m

XDM/DS : 反映側 Datareplicator = n : 1

変数	UNIX	Windows
m	mの範囲については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。	
n	1~128	1~63

一つの XDM/DS から、複数の反映側 Datareplicator に更新情報を送信できます。

逆に、一つの反映側 Datareplicator では、複数の XDM/DS からの更新情報を受信できます。さらに、一つの反映側 HiRDB に複数の反映側 Datareplicator を稼働させると、128 個以上（Windows 版では 63 個以上）の抽出側システム（データ連動識別子）から更新情報を受信できるようになります。ただし、次の不具合が起こるおそれがあります。

- 反映処理の排他エラーが発生する
- 更新情報を反映する順番が保証されない

(b) データ連動識別子

データ連動識別子の詳細については、「4.4.1(2)(c) データ連動識別子」を参照してください。

(c) 反映側 Datareplicator 識別子

反映側 Datareplicator を識別するための識別子を反映側 Datareplicator 識別子といいます。反映側 Datareplicator 識別子は、反映システム定義の hdsid オペランドで指定します。

抽出側 Datareplicator では、更新情報の送信先になる反映側 Datareplicator 識別子を、送信環境定義の sendhdsid オペランドで指定します。

(d) 通信プロトコル

XDM/DS と反映側 Datareplicator とで更新情報を送受信するために使う通信プロトコルの指定が必要になります。反映側 Datareplicator では、反映システム定義の `protocoll`, `protocol2` オペランドで指定します。

(e) T セレクタ

XDM/DS と反映側 Datareplicator とで OSI プロトコル（チャンネル接続）を使って、更新情報の送受信をする場合、T セレクタを指定する必要があります。反映側 Datareplicator では、反映システム定義の `reflect_tselector` オペランドで指定します。

(f) サービス名

XDM/DS と反映側 Datareplicator とで TCP/IP プロトコルを使って更新情報を送受信するために、次に示す `services` ファイルに通信のサービス名を追加する必要があります。

- UNIX 版 Datareplicator の場合：/etc/services
- Windows 版 Datareplicator の場合：Windows のシステムディレクトリ¥drivers¥etc¥services

この通信のサービス名を、反映側 Datareplicator の `hdsservice` オペランドで指定します。

(3) 反映側 Datareplicator と反映側 HiRDB との対応

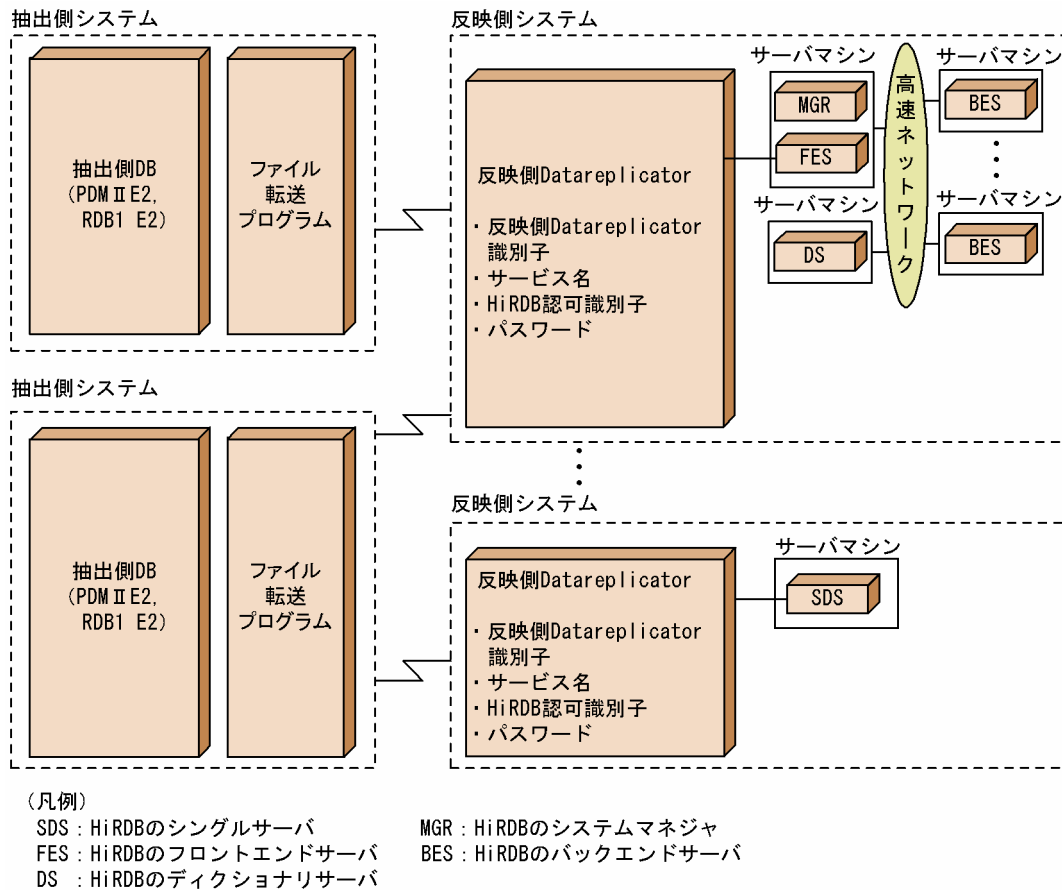
反映側 Datareplicator と反映側 HiRDB との対応については、「[4.4.1\(3\) 反映側 Datareplicator と反映側 HiRDB との対応](#)」を参照してください。

4.4.4 SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動する場合の設計

SAM ファイルを使ってメインフレーム側 DB（PDMII E2, RDB1 E2）から HiRDB へデータ連動する場合の、抽出側システムと反映側システムの対応を次の図に示します。

抽出側 DB には、データ連動製品は必要ありません。ただし抽出側 DB では、作成した SAM ファイルを反映側システムにファイル転送するプログラムが必要です。

図 4-22 メインフレーム側 DB (PDMII E2, RDB1 E2) から HiRDB へデータ連動する場合の、抽出側システムと反映側システムとの対応



(1) メインフレーム側 DB と反映側 Datareplicator との対応

SAM ファイルを使うメインフレーム側 DB (PDMII E2, RDB1 E2) と反映側 Datareplicator とを対応付けるために、次の項目を設計する必要があります。

ここでは、反映側 Datareplicator の指定について説明します。

(a) 対応数

メインフレーム側 DB と反映側 Datareplicator の対応数には、特に制限はありません。ただし、一つの反映側 Datareplicator で同時に実行できる更新情報入力コマンド (hdssamqin コマンド) の数は、UNIX 版の場合は 128 個まで、Windows 版の場合は 63 個までです。

さらに、一つの反映側 HiRDB に複数の反映側 Datareplicator を稼働させると、128 個以上の抽出側システム (データ連動識別子) から更新情報を受信できるようになります。ただし、一つの反映側 HiRDB (DB) に対して複数の反映側 Datareplicator から更新情報を反映する形態では、次の不具合が起こるおそれがあります。

- 反映処理の排他エラーが発生する
- 更新情報を反映する順番が保証されない

(b) データ連動識別子

反映側 Datareplicator では、反映システム定義の dsidxxx オペランドでデータ連動識別子を指定します。ここで指定したデータ連動識別子を、hdssamqin コマンドに指定して実行します。データ連動識別子の詳細については、「[4.4.1\(2\)\(c\) データ連動識別子](#)」を参照してください。

(c) 反映側 Datareplicator 識別子

反映側 Datareplicator を識別するための識別子を反映側 Datareplicator 識別子といいます。反映側 Datareplicator 識別子は、反映システム定義の hdsid オペランドで指定します。

抽出側 Datareplicator では、更新情報の送信先になる反映側 Datareplicator 識別子を、送信環境定義の sendhdsid オペランドで指定します。

(2) 反映側 Datareplicator と反映側 HiRDB との対応

反映側 Datareplicator と反映側 HiRDB との対応については、「[4.4.1\(3\) 反映側 Datareplicator と反映側 HiRDB との対応](#)」を参照してください。

4.5 データ連動システムの形態の設計

データ連動システムの形態の設計について説明します。ここでは、HiRDB から HiRDB へデータ連動する場合の例で説明します。

4.5.1 データ連動システムの形態の種類

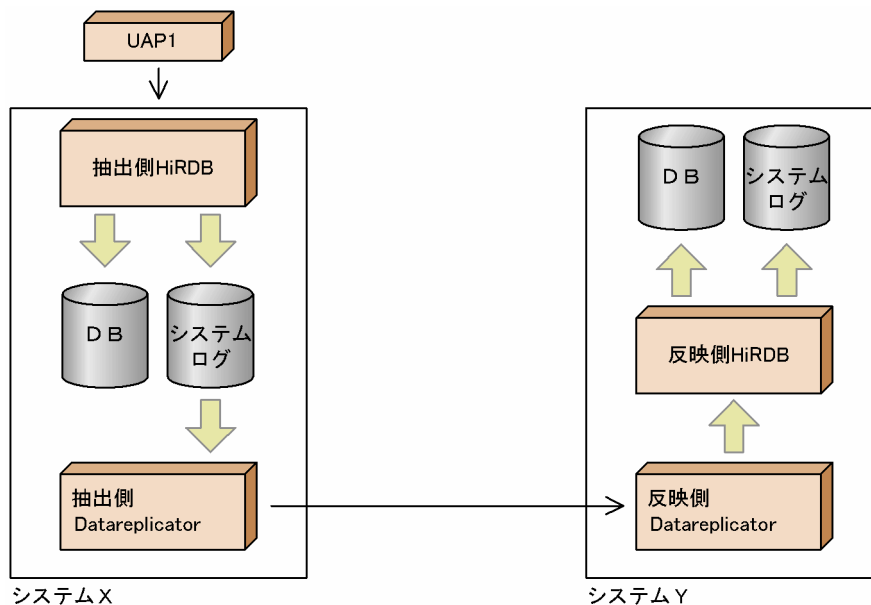
Datareplicator で設計できるデータ連動システムの種類について説明します。

(1) 単一方向更新システム

単一方向更新システムでは、一方が抽出側システムになって、もう一方が反映側システムになります。このため、データ連動の流れは単一方向です。

単一方向更新システムの例を次の図に示します。

図 4-23 単一方向更新システムの例

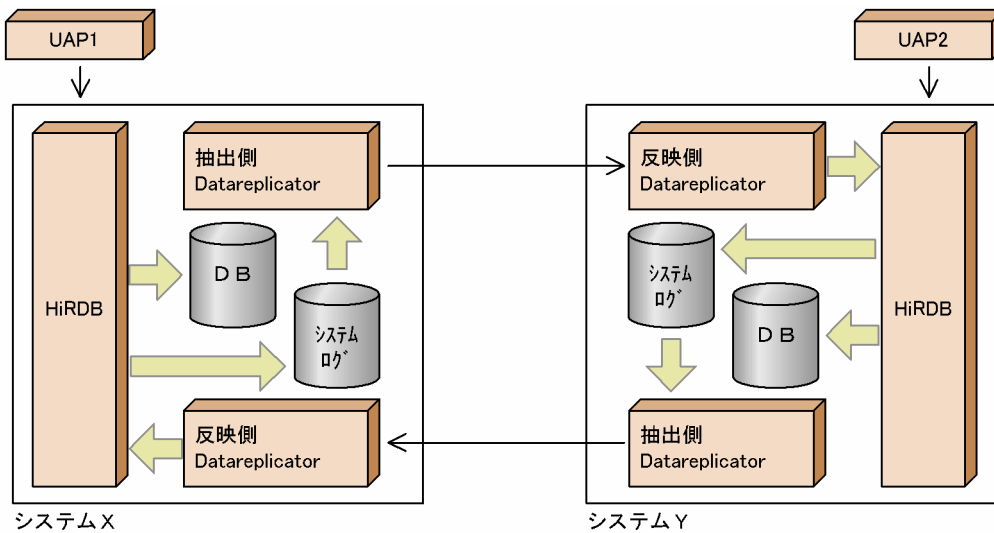


(2) 双方向更新システム

双方向更新システムでは、双方がそれぞれ抽出側システム、及び反映側システムになります。このため、データ連動の流れは双方向です。

双方向更新システムの例を次の図に示します。

図 4-24 双方向更新システムの例



4.5.2 データ連動システムの階層型システムへの応用形態の設計

データ連動システムの階層型システムへの応用形態の設計について説明します。

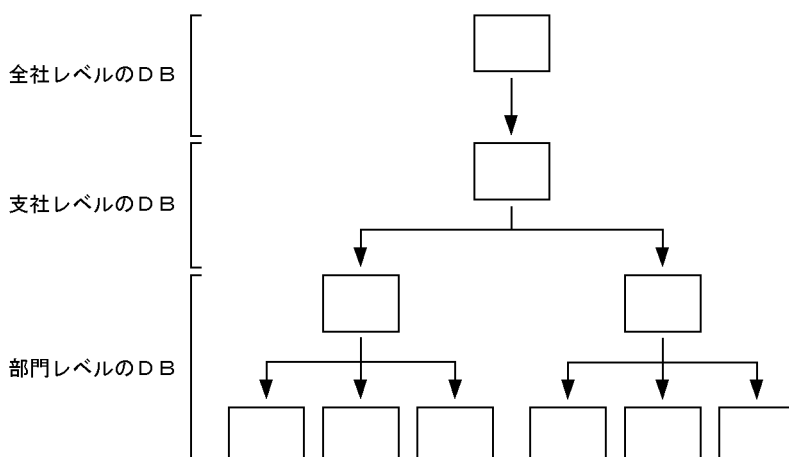
(1) 単一方向更新システムの階層型システムへの応用形態

次の特長を持つ階層型システムに対して、単一方向更新システムを適用できます。

- 上位の DB の更新情報を下位の DB に反映するだけで、下位の DB の更新情報は、上位の DB に反映する必要がない

単一方向更新システムを適用した階層型システムを次の図に示します。

図 4-25 単一方向更新システムを適用した階層型システムの例



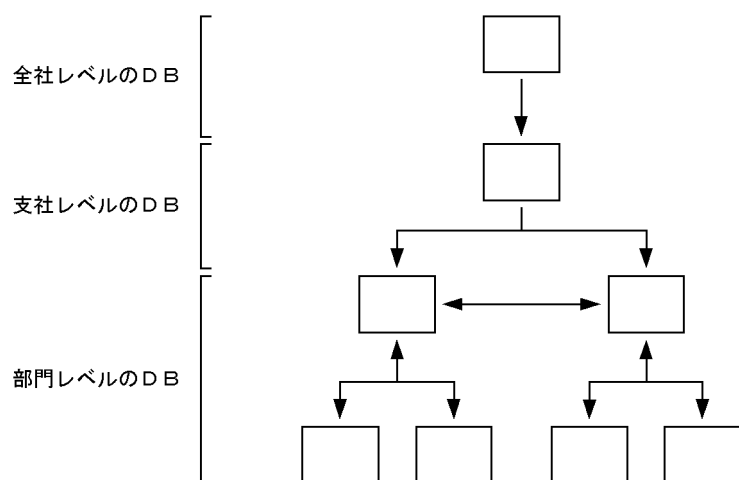
(2) 双方向更新システムの階層型システムへの応用形態

次の二つの特長を持つ階層型システムに対して、双方向更新システムを適用できます。

- 全社レベル、支社レベルのDBのように、規模の大きいDBでは、下位のDBに反映するだけで、下位のDBの更新情報は、上位のDBに反映する必要がありません。この場合、全社レベル、支社レベルでは単一方向更新システムになります。
- 部門レベルやそれ以下のレベルのDBのように、限られた情報を持つDBでは、情報を共有します。この場合、部門レベルとそれ以下のレベルでは、双方向更新システムになります。

双方向更新システムを適用した階層型システムの例を次の図に示します。

図 4-26 双方向更新システムを適用したデータ連動システムの例



4.5.3 複数システム間でのデータ連動システムの注意

複数システム間でのデータ連動システムの注意について説明します。

(1) 複数の抽出側システムからの、同一の反映対象表に対する更新

複数の抽出側システムで同一の反映対象表に対する更新が発生する場合、最終的にどのシステムで更新されたデータであるかが特定できなくなります。例えば、複数の抽出側システムで同一の表に対する全件 UPDATE が発生した場合、最終的にどのシステムで更新されたデータであるかが特定できなくなります。

これを防ぐためには、反映対象表に対してキーレンジ分割をして、各抽出側システムからの更新範囲を限定します。

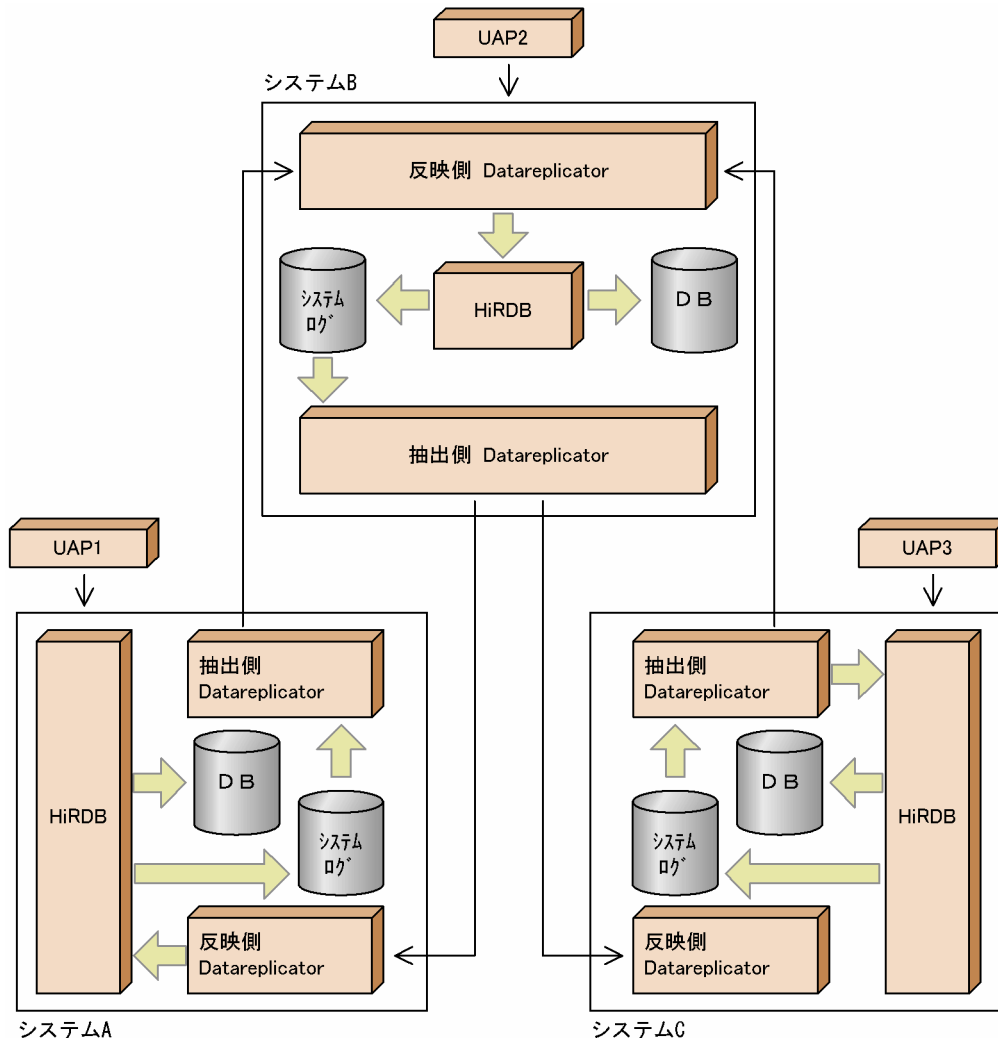
(2) 複数システム間でのデータ連動の順序性

一つの抽出側システムから複数の反映側システムへのデータ連動を実行している場合、各反映側システムでの反映処理の順序性は、各反映側システムの処理速度、負荷などに依存するため、不定になります（ただし、抽出側システムのトランザクションの順序性は保証されます）。

複数システム間でのデータ連動で順序付ける場合には、複数のシステムを一つの線をつないだシステムにしてください。例えば、システム A、システム B、及びシステム C の三つのシステムをデータ連動する場合、システム A とシステム B をデータ連動し、同じくシステム B とシステム C をデータ連動することで、システム A からシステム C までを一つの線をつないだ構成にできます。

複数のシステムを一つの線をつないだシステムの例を次の図に示します。

図 4-27 複数のシステムを一つの線をつないだシステムの例



(3) ループバックの抑止

双方向更新システムでは、複数のシステム間で更新情報を送受信します。このため、送信されてきた更新情報を反映した後、再抽出して送信元に送信し返すことで、複数のシステム間で同一の更新情報が無限に抽出、反映を繰り返す可能性があります。これをループバックといいます。

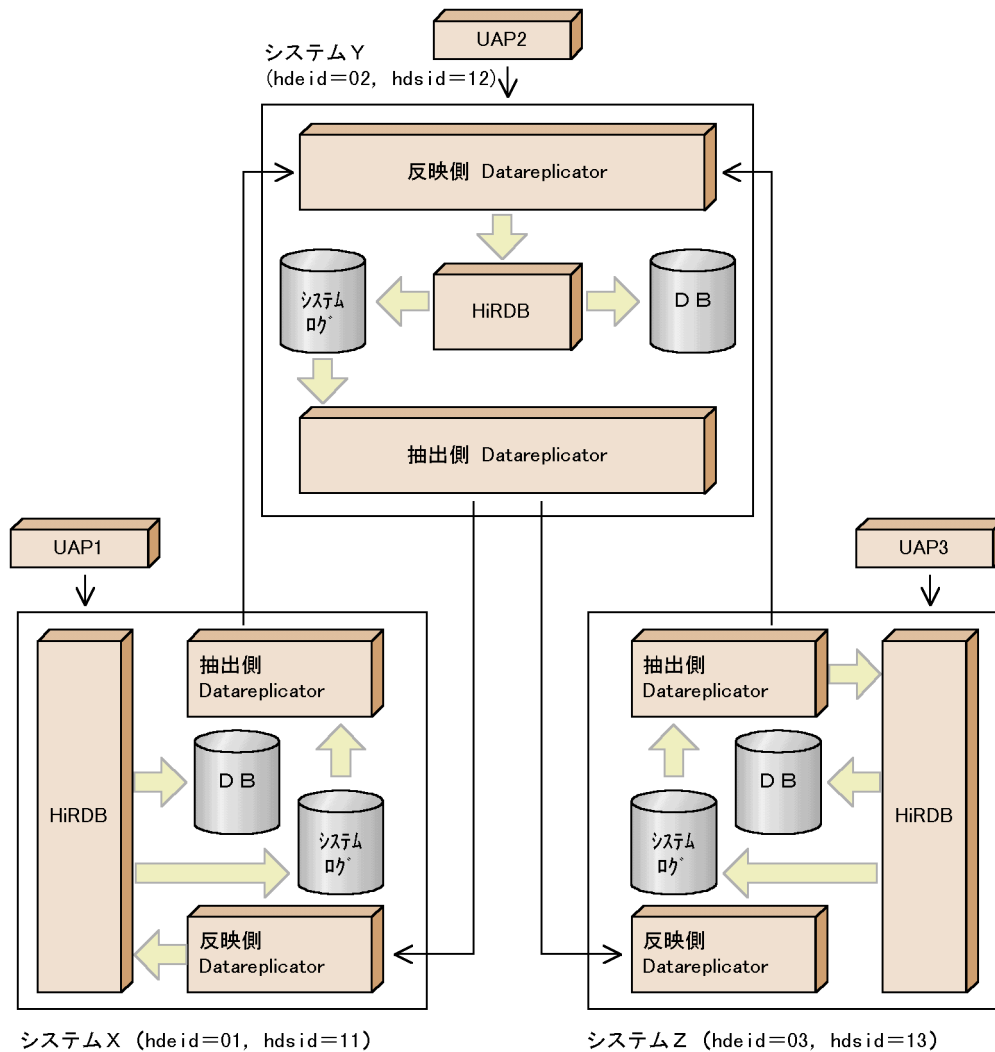
このループバックを抑止するには、送信先識別子ごとに送信環境定義の nsndidxxx (xxx は 001~256 の整数) オペランドで、送信を抑止する更新情報の受信元の識別子 (送信抑止受信元識別子) を指定します。送信抑止受信元識別子には、抑止の対象とするデータ連動システムの、抽出側 Datareplicator 識別子を指定します。例えば、データ連動システム A (hdeid = 01, hdsid = 02) で、データ連動システム B (hdeid

= 11, hdsid = 12) から送信されてくる更新情報を抑止の対象とする場合には、データ連動システム A の送信環境定義で、nsndid001 = 11 と指定します。

送信抑止の指定をしているときは、UAP 名に hdssqle で始まる名称は使わないでください。送信抑止の指定をしていると、hdssqle で始まる UAP 名の更新情報の送信が抑止されることがあります。

ループバックを抑止する双方向更新システムの形態の例を次の図に示します。

図 4-28 ループバックを抑止する双方向更新システムの形態の例



● 送信抑止の設定内容

- システム X : sendhdsid=12, nsndid001=02
- システム Y : sendhdsid=11, nsndid001=01
- sendhdsid=13, nsndid001=03
- システム Z : sendhdsid=12, nsndid001=02

図 4-27 で実行される Datareplicator の処理は次のとおりです。

• 動作内容

UAP1 の更新情報の流れ

1. システム X で、UAP1 の更新情報をシステムログファイルに格納します。

2. システム X では 1.の更新情報を抽出機能で抽出して、システム Y に送信します（システム X の抑止の指定はシステム Y に対して nsndid001 = 02 だが、この段階の UAP1 の更新情報には受信元がないため、抑止しません）。
3. システム Y では 2.で送信された更新情報を反映機能で反映して、更新情報をシステムログファイルに格納します。
4. システム Y では 3.の更新情報を抽出機能で抽出して、システム Z に送信します（システム Y の抑止の指定はシステム X に対しては nsndid001 = 01、システム Z に対しては nsndid001 = 03 だが、この段階の UAP1 の更新情報の受信元はシステム X であるため、システム X への送信だけが抑止されます）。
5. システム Z では 4.で送信された更新情報を反映機能で反映して、更新情報をシステムログファイルに格納します。
6. システム Z では 5.の更新情報を抽出機能で抽出するが、送信しません（システム Z の抑止の指定はシステム Y に対して nsndid001 = 02 だが、この段階の UAP1 の更新情報の受信元はシステム Y であるため、システム Y への送信が抑止されます）。

UAP2 の更新情報の流れ

1. システム Y で、UAP2 の更新情報をシステムログファイルに格納します。
2. システム Y では 1.の更新情報を抽出機能で抽出して、システム X とシステム Z に送信します（システム Y の抑止の指定はシステム X に対しては nsndid001 = 01、システム Z に対しては nsndid001 = 03 だが、この段階の UAP2 の更新情報には受信元がないため、抑止されません）。
3. システム X, Z では 2.で送信された更新情報を反映機能で反映して、更新情報をシステムログファイルに格納します。
4. システム X, Z では、3.の更新情報を抽出機能で抽出するが、送信しません（システム X, 及びシステム Z での抑止の指定はシステム Y に対して nsndid001 = 02 であり、この段階の UAP2 の更新情報の受信元はシステム Y であるため、システム Y への送信が抑止されます）。

UAP3 の更新情報の流れ

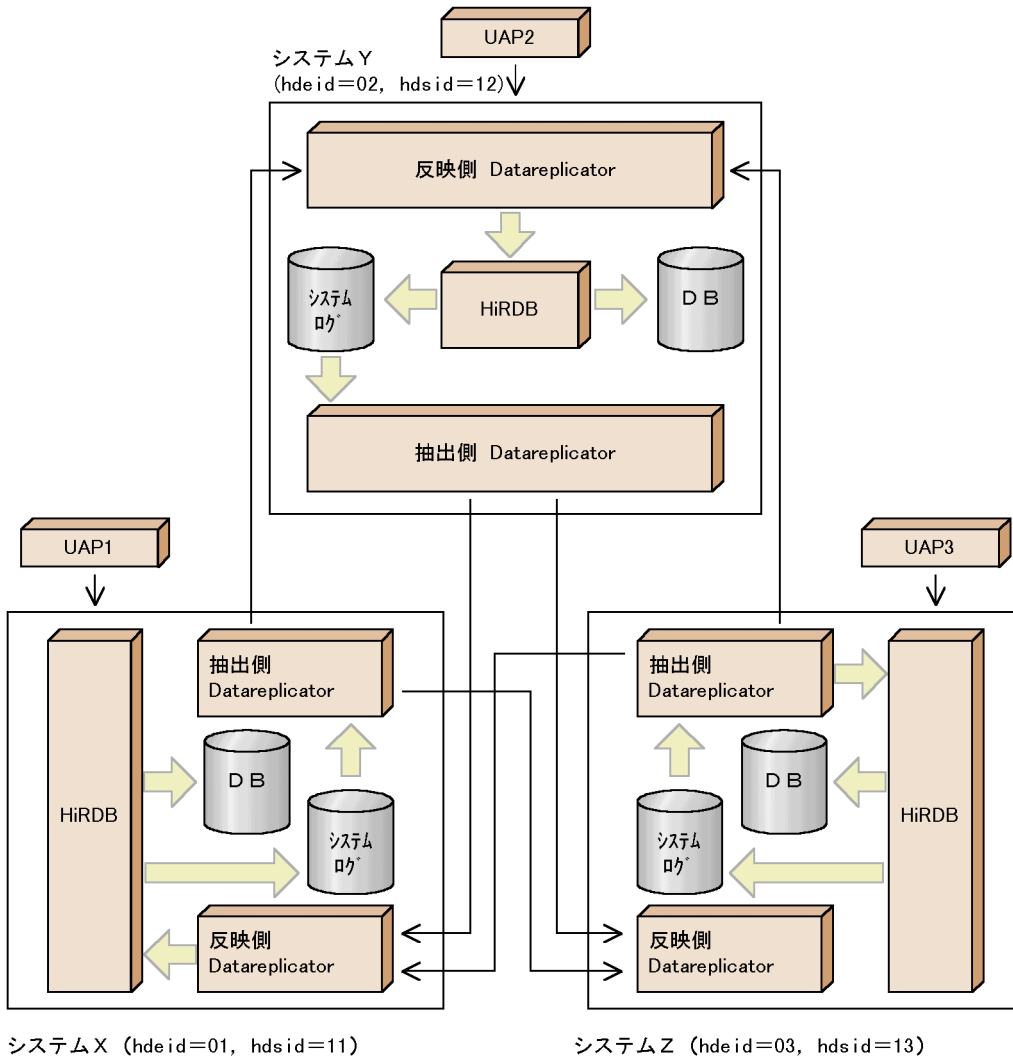
1. システム Z で、UAP3 の更新情報をシステムログファイルに格納します。
2. システム Z では 1.の更新情報を抽出機能で抽出して、システム Y に送信します（システム Z の抑止の指定はシステム Y に対して nsndid001 = 02 だが、この段階の UAP3 の更新情報には受信元がないため、抑止されません）。
3. システム Y では 2.で送信された更新情報を反映機能で反映して、更新情報をシステムログファイルに格納します。
4. システム Y では 3.の更新情報を抽出機能で抽出して、システム X に送信します（システム Y の抑止の指定はシステム X に対しては nsndid001 = 01、システム Z に対しては nsndid001 = 03 だが、この段階の UAP3 の更新情報の受信元はシステム Z であるため、システム Z への送信だけが抑止されます）。
5. システム X では 4.で送信された更新情報を反映機能で反映して、更新情報をシステムログファイルに格納します。

6. システム X では 5. の更新情報を抽出機能で抽出するが、送信しません（システム X の抑止の指定はシステム Y に対して nsendid001 = 02 であり、この段階の UAP3 の更新情報の受信元はシステム Y であるため、システム Y への送信が抑止されます）。

(4) 三つ以上のシステムの間で双方向更新をするデータ連動システム

三つ以上のシステムの間で双方向更新をするの例を次の図に示します。

図 4-29 三つ以上のシステムの間で双方向更新をする例



<設定内容>

システム X : sendhdsid=12, nsendid001=02, nsendid002=03
 sendhdsid=13, nsendid001=02, nsendid002=03
 システム Y : sendhdsid=11, nsendid001=01, nsendid002=03
 sendhdsid=13, nsendid001=01, nsendid002=03
 システム Z : sendhdsid=11, nsendid001=01, nsendid002=02
 sendhdsid=12, nsendid001=01, nsendid002=02

図 4-28 で実行される Datareplicator の処理は次のとおりです。

- 動作内容

UAP1 の更新情報の流れ

1. システム X で、UAP1 の更新情報をシステムログファイルに格納します。
2. システム X では、1.の更新情報を抽出機能で抽出して、システム Y、Z に送信します（システム X では抑止を指定していますが、この段階の UAP1 の更新情報には抽出元がないため、抑止されません）。
3. システム Y、Z では、2.の更新情報を反映機能で反映して、更新情報をシステムログファイルに格納します。
4. システム Y では、システム X、Z に対して共に nsendid001=01 で抽出元がシステム X となる更新情報の送信を抑止しているため、3.の更新情報は送信されません。
5. システム Z でも同様に、3.の更新情報は送信されません。

UAP2, UAP3 についても同様に、ループバックの制御をします。

(5) XDM/DS からデータ連動機能を使う場合のループバック抑止について

XDM/DS から送信された更新情報に対する対応を次に示します。

XDM/DS のバージョン	XDM/DS から送信された更新情報に対する対応	
	ループバックの制御をするとき	UOC で反映するとき
07-00 以前	送信環境定義の nsendidxxx オペランドには「00」を設定してください。設定しないと、その更新情報を他システムへ送信します。	インタフェースブロック (UINTERFACE_BLK) の抽出側 Datareplicator 識別子には「00」が設定されます。
07-01 以降	送信環境定義の nsendidxxx オペランドには XDM/DS 起動定義の XDM/DS 識別子句へ指定した文字コード (EBCDIK) を設定してください。設定しないと、その更新情報を他システムへ送信します。	インタフェースブロック (UINTERFACE_BLK) の抽出側 Datareplicator 識別子には XDM/DS 起動定義の XDM/DS 識別子句へ指定した文字コード (EBCDIK) が設定されます。

4.6 抽出側 Datareplicator の設計

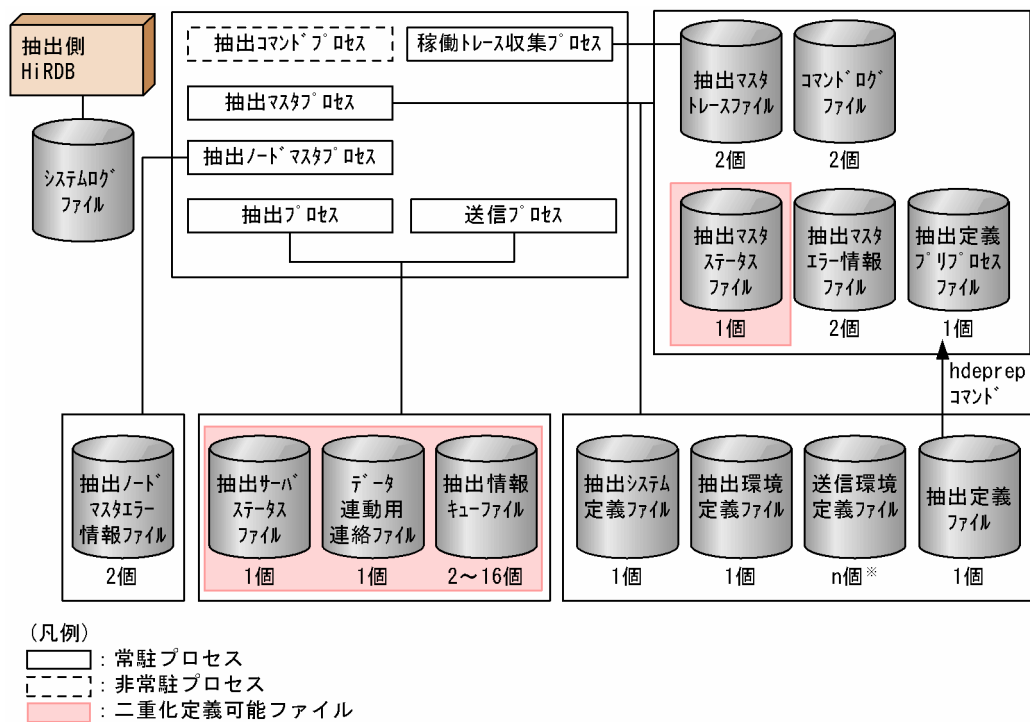
抽出側 Datareplicator のシステム設計について説明します。

4.6.1 抽出側 Datareplicator のファイル構成

抽出側 Datareplicator のファイル構成を図 4-30 と図 4-31 に示します。

抽出側 Datareplicator のディレクトリ構成については、UNIX 版の場合は「2.3.2 抽出側 Datareplicator のディレクトリ構成」を、Windows 版の場合は「2.7.2 抽出側 Datareplicator のディレクトリ構成」を参照してください。

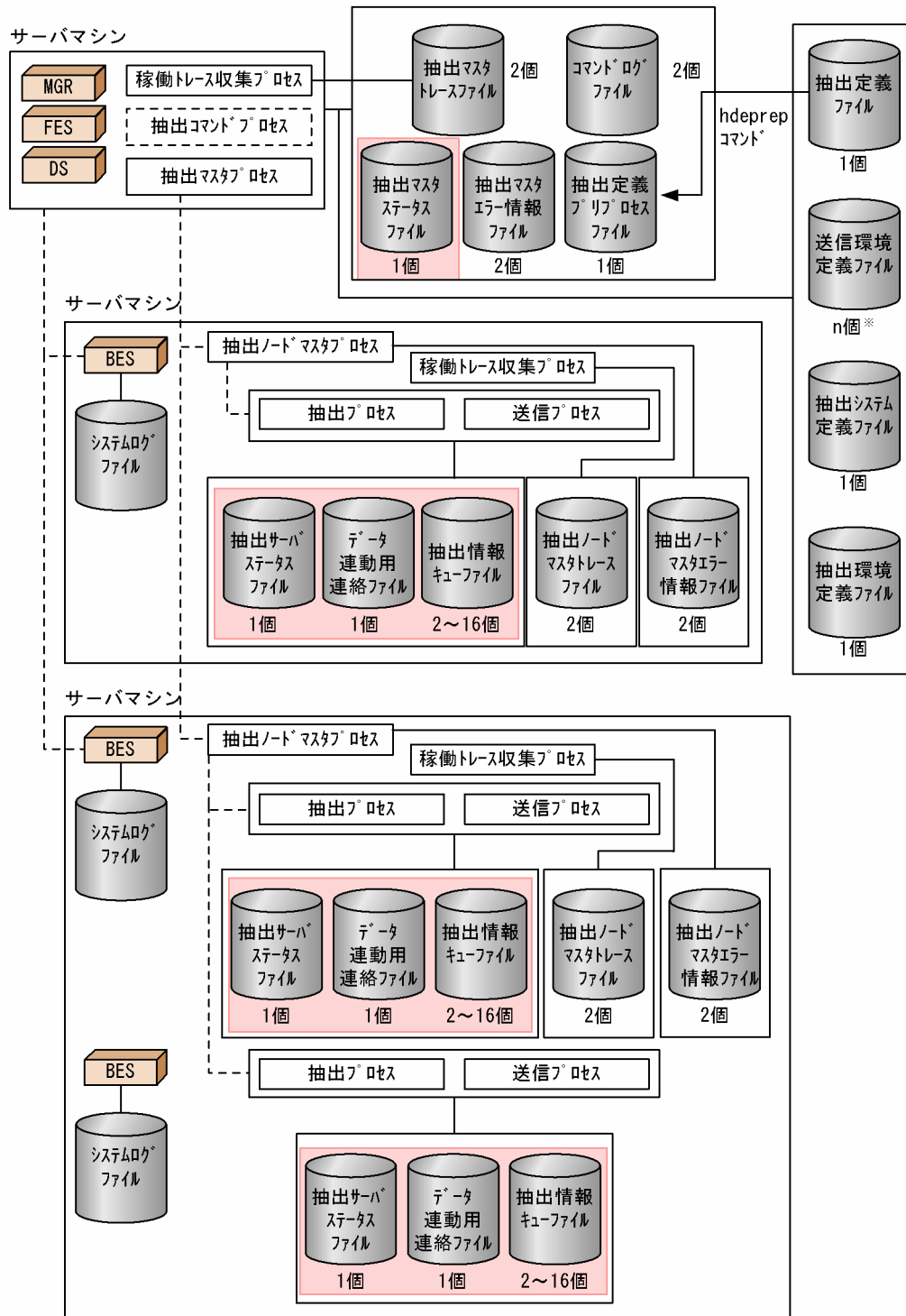
図 4-30 抽出側 HiRDB がシングルサーバの場合の抽出側 Datareplicator のファイル構成



注※

n は送信先となる反映側システムの数です。

図 4-31 抽出側 HiRDB がパラレルサーバの場合の抽出側 Datareplicator のファイル構成



(凡例)

- | | | | |
|--|--------------------|---|--------------------|
| MGR | : HiRDBのシステムマネージャ | FES | : HiRDBのフロントエンドサーバ |
| DS | : HiRDBのディクショナリサーバ | BES | : HiRDBのバックエンドサーバ |
| | : 常駐プロセス | | : 非常駐プロセス |
| | : 二重化定義可能ファイル | - - - - | : 制御 |

注※

n は送信先となる反映側システムの数です。

4.6.2 抽出側 Datareplicator で使うファイルの準備

抽出側 Datareplicator を動作するために必要な、次に示すファイルの準備方法について説明します。

ユーザが作成するファイル	抽出側 Datareplicator の初期起動時に作成されるファイル
<ul style="list-style-type: none"> 抽出システム定義ファイル 抽出環境定義ファイル 送信環境定義ファイル 抽出定義ファイル 二重化定義ファイル 	<ul style="list-style-type: none"> 抽出定義プリプロセスファイル 抽出情報キューファイル 抽出マスタステータスファイル 抽出サーバステータスファイル 抽出マスタエラー情報ファイル 抽出ノードマスタエラー情報ファイル 抽出マスタトレースファイル 抽出ノードマスタトレースファイル データ連動用連絡ファイル コマンドログファイル 二重化制御ファイル

ファイルの内容については、「[3.2.2 抽出処理で使うファイルとプロセス](#)」を参照してください。

ファイルの運用については、「[6.4.2 抽出側 Datareplicator で使うファイルの運用](#)」を参照してください。

抽出側 Datareplicator で使うファイルの作成について次の表に示します。

表 4-32 抽出側 Datareplicator で使うファイルの作成

ファイル名		ファイル種別 ^{※7}		個数	必要・任意
		R ^{※8}	C ^{※9}		
定義ファイル	抽出システム定義ファイル ^{※1}	○	×	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個 ^{※10})	必要
	抽出環境定義ファイル ^{※1}	○	×	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個 ^{※11})	必要
	送信環境定義ファイル ^{※1}	○	×	送信先識別子ごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に送信先識別子ごとに 1 個 ^{※11})	必要
	抽出定義ファイル ^{※2}	○	×	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個)	必要
	二重化定義ファイル ^{※1}	○	×	抽出側システムごとに 1 個	任意

ファイル名		ファイル種別※7		個数	必要・任意
		R※8	C※9		
				(抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個)	
抽出定義プリプロセスファイル※3		○	○	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個)	必要
抽出情報キューファイル※4※12		○	○	抽出側システムごとに 2~16 個 (抽出側 HiRDB がパラレルサーバの場合には、BES ごとに 2~16 個)	必要
ステータスファイル※4※12	抽出マスタステータスファイル	○	○	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個)	必要
	抽出サーバステータスファイル	○	○	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、BES ごとに 1 個)	必要
エラー情報ファイル※5※6	抽出マスタエラー情報ファイル	○	×	抽出側システムごとに 2 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 2 個)	必要
	抽出ノードマスタエラー情報ファイル	○	×	抽出側システムごと 2 個 (抽出側 HiRDB がパラレルサーバの場合には、BES が存在するサーバマシンごとに 2 個)	必要
稼働トレースファイル	抽出マスタトレースファイル※13	○	×	抽出側システムごとに 2 個	任意
	抽出ノードマスタトレースファイル※13	○	×	抽出側システムごとに 2 個 (抽出側 HiRDB がパラレルサーバの場合には、BES ごとに 2 個)	任意
データ連動用連絡ファイル※4※12		○	○	抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、BES ごとに 1 個)	必要
コマンドログファイル※6		○	×	抽出側システムごとに 2 個	任意
二重化制御ファイル※5		○	×	抽出側システムごとに 1 個	任意

(凡例)

MGR：システムマネージャ

BES：バックエンドサーバ（抽出対象の DB がないバックエンドサーバも含む）

R：UNIX の通常ファイル，又は Windows ファイル

C：UNIX のキャラクタ型スペシャルファイル

○：作成できます。

×：作成できません。

注※1

抽出側 Datareplicator を起動する前に、OS のエディタで作成してください。

注※2

hdeprep コマンドを実行する前に、OS のエディタで作成してください。

注※3

抽出定義ファイルを作成後、抽出側 Datareplicator を起動する前に hdeprep コマンドを実行してください。hdeprep コマンドを実行するとファイルが自動的に作成されます。ただし、UNIX のキャラクタ型スペシャルファイルの場合は、hdeprep コマンドを実行する前に、キャラクタ型スペシャルファイルに対するシンボリックリンクを作成してください。

注※4

抽出側 Datareplicator を起動する前に hdestart - i コマンドを実行して抽出側 Datareplicator を初期化してください。hdestart - i コマンドを実行すると、ファイルが自動的に作成されます。ただし、UNIX のキャラクタ型スペシャルファイルの場合は、抽出側 Datareplicator を起動する前に、キャラクタ型スペシャルファイルに対するシンボリックリンクを作成してください。

注※5

抽出側 Datareplicator の初期化時に自動的に作成されます。

注※6

抽出側 Datareplicator の起動時に自動的に作成されます。

注※7

次のファイルはすべて同じファイル種別にしてください。

- 抽出情報キューファイル
- 抽出サーバステータスファイル
- データ連動用連絡キューファイル

注※8

ユーザが作成するファイル（定義ファイル）の場合には、抽出側 Datareplicator の使用者に read 権限を与えてください。write 権限は任意で与えてください。ユーザが作成するファイル（定義ファイル）以外の場合は、権限を変更しないでください。

注※9

キャラクタ型スペシャルファイルは OS のコマンドで作成します。キャラクタ型スペシャルファイルで作成する場合は、次の権限を設定してください。

データ連動用連絡ファイルの場合

以下の権限を設定してください。

項番	実行ユーザ	設定する権限
1	抽出側 Datareplicator の使用者	読み込み権限 書き込み権限 実行権限
2	抽出側 Datareplicator の使用者と同一グループ	読み込み権限 書き込み権限
3	その他のグループ	読み込み権限

データ連動用連絡ファイル以外の場合

抽出側 Datareplicator の使用者に read 権限と write 権限を与えてください。

また、系切り替え機能を使うときは、キャラクタ型スペシャルファイルで作成してください。

抽出側 Datareplicator の稼働中にファイルの容量不足が発生しないよう、環境構築時、hdestart -i コマンドに init を指定して実行し、ファイルのサイズに不足がないかテストすることをお勧めします。

注※10

抽出側 HiRDB がパラレルサーバで、さらに HiRDB の環境変数 (PDDIR, PDCONF_PATH, SHLIB_PATH など) にサーバマシンごとに異なる指定をしている場合には、環境変数に対するオペランドをサーバマシンごとに個別に作成することもできます。

注※11

抽出側 HiRDB がパラレルサーバの場合には、BES ごとに作成することもできます。

注※12

UNIX 版の場合、ファイル種別を OS の通常ファイルとしていると、システムの障害時に、データが出力されないことがあります。抽出情報キューファイル、抽出時のステータスファイル、及びデータ連動用連絡ファイルは、障害回復時に抽出側 Datareplicator が使うため、このファイルにデータが出力されていないと障害を回復できなくなります。このため、抽出情報キューファイル、抽出時のステータスファイル、及びデータ連動用連絡ファイルのファイル種別は、信頼性の高いキャラクタ型スペシャルファイルにすることをお勧めします。

注※13

稼働トレース収集方式を全プロセス一括収集方式とする場合のファイル名です。プロセス単位収集方式とする場合の稼働トレースファイル名については、「5.2 抽出システム定義」の int_trc_unit オペランドの説明を参照してください。

抽出側 Datareplicator で使うファイルの設定について次の表に示します。

表 4-33 抽出側 Datareplicator で使うファイルの設定

ファイルの種類	ファイル名	設定
抽出システム定義ファイル	\$HDEPATH/hdeenv	• 定義する内容については、「5.2 抽出システム定義」を参照してください。
抽出環境定義ファイル	任意のディレクトリ/任意の名称	• 定義する内容については、「5.3 抽出環境定義」を参照してください。

ファイルの種類	ファイル名	設定
		<ul style="list-style-type: none"> 任意の名称を抽出システム定義の extdef オペランドに指定してください。
送信環境定義ファイル	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 定義する内容については、[5.4 送信環境定義]を参照してください。 任意の名称を抽出システム定義の senddef01～senddef64 オペランドに指定してください。
抽出定義ファイル	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 定義する内容については、[5.5 抽出定義]を参照してください。 任意の名称を hdeprep コマンド実行時に指定し、抽出定義プリプロセスファイルを生成してください。
二重化定義ファイル	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 定義する内容については、[5.7 二重化定義 (抽出側)]を参照してください。 任意の名称を抽出システム定義ファイルの file_dupenv オペランドに指定してください。
抽出定義プリプロセスファイル	\$HDEPATH/ hde_prpfile	—
抽出情報キューファイル※ ¹	任意のディレクトリ/任意の名称_サーバ名	<ul style="list-style-type: none"> 任意の名称を抽出環境定義の qufile001～qufile016 オペランドに指定してください。指定した名称に「_サーバ名」を付けたものがファイル名になります。 ファイル名は抽出側システムで一意にしてください。抽出側 HiRDB がパラレルサーバの場合には、各バックエンドサーバで一意にしてください。 抽出環境定義の queuesize オペランドでファイルのサイズを指定してください。
抽出マスタステータスファイル※ ¹	\$HDEPATH/mststatus	—
抽出サーバステータスファイル※ ¹	\$HDEPATH/sts_サーバ名	—
抽出マスタエラー情報ファイル	\$HDEPATH/msterrfile1 \$HDEPATH/msterrfile2	<ul style="list-style-type: none"> 抽出システム定義の errfilesz オペランドでファイルのサイズを指定してください。
抽出ノードマスタエラー情報ファイル	\$HDEPATH/errfile1 \$HDEPATH/errfile2	<ul style="list-style-type: none"> 抽出システム定義の errfile_unique オペランドに true を指定すると、「errfile1_ホスト名」、及び「errfile2_ホスト名」というファイル名になります。 抽出システム定義の errfilesz オペランドでファイルのサイズを指定してください。
抽出マスタトレースファイル※ ²	\$HDEPATH/msttrc.trc1 \$HDEPATH/msttrc.trc2	<ul style="list-style-type: none"> 抽出システム定義の int_trc_filesz オペランドでファイルのサイズを指定してください。 hdstrcredit コマンドで編集、又は参照ができます。
抽出ノードマスタトレースファイル※ ²	\$HDEPATH/exttrc.trc1 \$HDEPATH/exttrc.trc2	<ul style="list-style-type: none"> 抽出システム定義の int_trc_filesz オペランドでファイルのサイズを指定してください。

ファイルの種類	ファイル名	設定
		<ul style="list-style-type: none"> 抽出システム定義の <code>errfile_unique</code> オペランドに <code>true</code> を指定すると、「<code>exttrc.trc1_ホスト名</code>」, 及び「<code>exttrc.trc2_ホスト名</code>」というファイル名になります。 <code>hdstrcredit</code> コマンドで編集, 又は参照ができます。
データ連動用連絡ファイル※1	<code>\$HDEPATH/hde_サーバ名</code>	—
コマンドログファイル	任意のディレクトリ/任意の名称_サーバ名	—
二重化制御ファイル	<code>\$HDEPATH/hde_fileenv.prp</code>	—

(凡例)

— : 特にありません。

注※1

ファイルを二重化できます。ただし、ファイルシステム領域を使用する場合は、二重化できません。

注※2

稼働トレース収集方式を全プロセス一括収集方式とする場合のファイル名です。プロセス単位収集方式とする場合の稼働トレースファイル名については、「[5.2 抽出システム定義](#)」の `int_trc_unit` オペランドの説明を参照してください。

4.6.3 抽出方法の設計

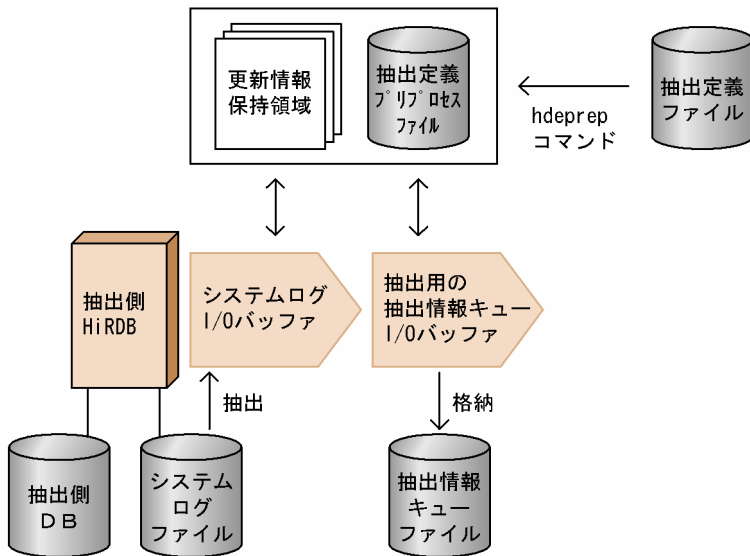
抽出方法の設計手順について説明します。

(1) 抽出で使うバッファの設計

抽出側 `Datareplicator` は、システムログ I/O バッファを使って、システムログファイルからシステムログを読み込みます。また、読み込んだシステムログの中で、抽出定義に指定された抽出対象表の更新情報だけを抽出情報キューファイルに格納します。このとき、抽出用の抽出情報キュー I/O バッファを使いません。

抽出方法の概要を次の図に示します。

図 4-32 抽出方法の概要



(a) システムログ I/O バッファ

抽出側 Datareplicator を起動すると、システムログ I/O バッファはローカルメモリ上に確保されます。ここでは、システムログ I/O バッファサイズの見積もり時の考慮点とサイズの指定について説明します。

システムログ I/O バッファサイズの見積もり時の考慮点

- システムログ I/O バッファは、システムログファイルからシステムログを抽出するときに使います。このため、システムログ I/O バッファのサイズは小さいよりも大きい方が、システムログファイルから 1 回で抽出する更新情報の量がより多くなって、読み書き回数を少なくできます。ただし、抽出側 Datareplicator が異常終了した場合、サイズを大きくした分、再起動時に抽出し直す更新情報の量が多くなるため、再起動時によって多くの時間が掛かります。
- 抽出側 HiRDB の `pd_log_max_data_size` に指定した値以上にする必要があります。
- 抽出側 HiRDB がパレルサーバの場合には、バックエンドサーバごとに見積もります。

システムログ I/O バッファのサイズの指定

システムログ I/O バッファのサイズは、抽出環境定義の `logiosize` オペランドで指定します。ここで指定したサイズで、システムログファイルから更新情報を抽出します。

(b) 抽出用の抽出情報キュー I/O バッファ

抽出側 Datareplicator を起動すると、抽出用の抽出情報キュー I/O バッファはローカルメモリ上に確保されます。ここでは、抽出用の抽出情報キュー I/O バッファサイズの見積もり時の考慮点とサイズの指定について説明します。

抽出用の抽出情報キュー I/O バッファサイズの見積もり時の考慮点

- 抽出用の抽出情報キュー I/O バッファは、抽出した更新情報を抽出情報キューファイルに格納するときに使います。このため、抽出用の抽出情報キュー I/O バッファのサイズは、小さいよりも大きい方が、より多くの更新情報を 1 回で抽出情報キューファイルに格納でき、読み書きの回数を少な

くできます。ただし、抽出側 Datareplicator が異常終了した場合、サイズを大きくした分、再起動時に抽出し直す更新情報の量が多くなるため、再起動時によって多くの時間が掛かります。

- 「抽出環境定義の `queuesize` オペランドの指定値-1」の値以下にする必要があります。
- 「(抽出環境定義の `queuesize` オペランドの指定値-1) ÷ 指定値」で余りがない値にする必要があります。余りの生じる値を指定した場合、抽出情報キューファイルの余りのサイズ分の領域は使われません。
- 抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバごとに見積もります。
- 抽出情報キュー I/O バッファは、抽出プロセスで一つ使います。

抽出用の抽出情報キュー I/O バッファのサイズの指定

抽出用の抽出情報キュー I/O バッファのサイズは、抽出環境定義の `quiosize` オペランドで指定します。ここで指定したサイズで、抽出用の抽出情報キュー I/O バッファと送信用の抽出情報キュー I/O バッファが確保されます。送信用の抽出情報キュー I/O バッファについては、「[4.6.4 送信方法の設計](#)」を参照してください。

(2) 抽出処理のエラー監視間隔の設計

抽出側 Datareplicator では、抽出マスタプロセスから抽出ノードマスタプロセス、抽出プロセス、及び送信プロセス（送信マスタプロセス）でエラーが発生しているかどうかを監視しています。エラーの監視間隔を短くすると、より早くエラーを検知できます。ただし、エラーの監視時間を短くし過ぎると、サーバと通信に負荷が掛かります。

エラー監視間隔は、抽出システム定義の `watchintvl` オペランドで指定します。

(3) 抽出環境定義の単位の設計

抽出側 HiRDB がパラレルサーバの場合、抽出側 Datareplicator はバックエンドサーバ単位に抽出を実行します。この場合、抽出側 Datareplicator の抽出環境定義では、すべてのバックエンドサーバに対しての共通の環境と、特定のバックエンドサーバに対しての環境を分けて定義できます。

共通の環境については、抽出環境定義の `commondef` の下で定義し、特定の環境については、`besdef`（サーバ名）の下で定義します。同一のオペランドを `commondef` と `besdef` とで指定した場合、`besdef` に指定したサーバに対しては、`besdef` での指定を優先します。

4.6.4 送信方法の設計

送信方法の設計手順について説明します。

(1) 送信で使うバッファの設計

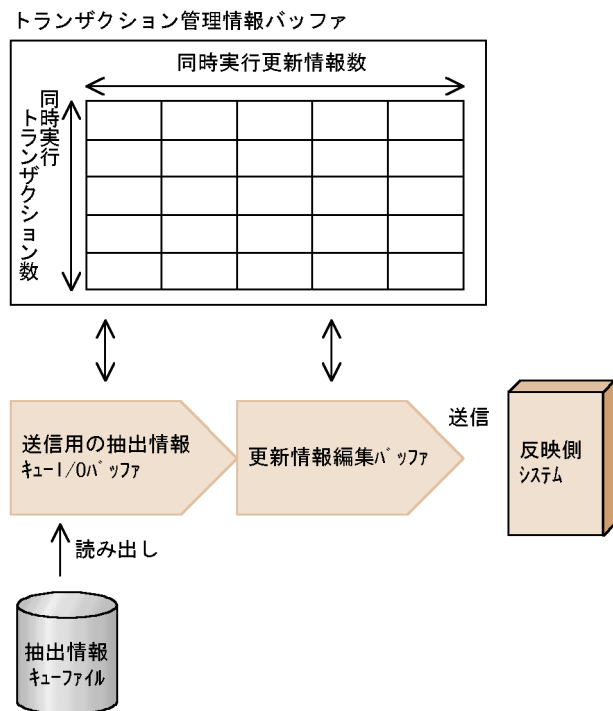
抽出側 Datareplicator は、送信先ごとに次の処理を実行しています。

- 抽出情報キューファイルから、送信対象の更新情報を読み込みます。このとき、送信用の抽出情報キュー I/O バッファを使います。
- 抽出情報キューファイルから読み込んだ更新情報の中で、トランザクションが完了している更新情報だけを抽出して編集して、反映側システムに送信します。このとき、更新情報編集バッファを使います。

また、トランザクションの完了状態を管理するため、トランザクションの管理情報を格納しているトランザクション管理情報バッファを使います。

送信方法の概要を次の図に示します。

図 4-33 送信方法の概要



(a) トランザクション管理情報バッファ

トランザクション管理情報バッファは、抽出側 Datareplicator の起動時に、ローカルメモリ上に確保されます。ここでは、トランザクション管理情報バッファサイズの見積もり時の考慮点とバッファサイズの指定について説明します。

トランザクション管理情報バッファサイズの見積もり時の考慮点

- トランザクション管理情報バッファには、抽出情報キューファイルに格納されているトランザクションの管理情報が格納されます。抽出側 Datareplicator では、更新情報の編集と送信に、このトランザクション管理情報バッファを使います。データの抽出、送信を実行中に、トランザクション管理情報のサイズがバッファサイズよりも大きくなった場合には、抽出側 Datareplicator によって、さらに初期値の 1/5 の領域が追加されます。この場合、バッファの追加のためにシステムに負荷が掛かり、処理速度が低下することがあります。バッファサイズを大きくする方が、バッファの追加が発生する可能性が低くなります。逆にバッファサイズを大きくし過ぎると、メモリが不足する可能

性が大きくなります。このため、トランザクション管理情報バッファのサイズを見積もるときには、領域追加が発生しないで、さらにメモリが不足しないように十分考慮してください。

- 抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバごとに見積もります。

トランザクション管理情報バッファサイズの指定

トランザクション管理情報バッファサイズは、送信環境定義の maxtran オペランドと maxtrandata オペランドで指定します。抽出側 Datareplicator では、maxtran オペランドで指定した同時実行最大トランザクション数と、maxtrandata オペランドで指定したトランザクション内最大更新情報数の積を、トランザクション管理情報バッファサイズの初期値とします。

(b) 送信用の抽出情報キュー I/O バッファ

抽出側 Datareplicator を起動すると、送信用の抽出情報キュー I/O バッファはローカルメモリ上に確保されます。ここでは、送信用の抽出情報キュー I/O バッファサイズ、バッファ数の見積もり時の考慮点、バッファサイズの指定、バッファ数の指定について説明します。

送信用の抽出情報キュー I/O バッファサイズ、バッファ数の見積もり時の考慮点

- 送信用の抽出情報キュー I/O バッファは、抽出情報キューファイルに格納されている更新情報を読み込むときに使います。このときには、システムログファイルから抽出した更新情報を格納したときと同じサイズで、更新情報を読み込む必要があります。このため、送信用の抽出情報キュー I/O バッファのサイズは、抽出用の抽出情報キュー I/O バッファと同じになります。抽出用の抽出情報キュー I/O バッファについては、「[4.6.3 抽出方法の設計](#)」を参照してください。
- 送信用の抽出情報キュー I/O バッファでは、バッファ数が指定できます。送信間隔が指定されている場合、抽出側 Datareplicator は送信間隔で更新情報を再び読み込んで送信します。このため、バッファ数が多く指定されていると、バッファ中の更新情報を使えるため、抽出情報キューファイルからの読み込みを減らせます。

送信用の抽出情報キュー I/O バッファのサイズの指定

送信用の抽出情報キュー I/O バッファのサイズは、抽出用の抽出情報キュー I/O バッファと同じになります。抽出用の抽出情報キュー I/O バッファについては、「[4.6.3 抽出方法の設計](#)」を参照してください。

送信用の抽出情報キュー I/O バッファの数の指定

送信用の抽出情報キュー I/O バッファの数は、送信環境定義の readbufnum オペランドで指定します。

(c) 更新情報編集バッファ

更新情報編集バッファは、抽出側 Datareplicator の起動時に、ローカルメモリ上に確保されます。ここでは、更新情報編集バッファサイズの見積もり時の考慮点とサイズの指定について説明します。

更新情報編集バッファサイズの見積もり時の考慮点

- 更新情報編集バッファを使って、更新情報を編集します。このため、更新情報編集バッファのサイズは、小さいよりも大きい方が、より多くの更新情報を 1 回で編集でき、読み書きの回数を少なくできます。ただし、サイズを大きくし過ぎると、メモリ不足になることもあります。

- ISQL 以上の更新情報が格納できるサイズにする必要があります。
- 抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバごとに見積もります。

更新情報編集バッファサイズの指定

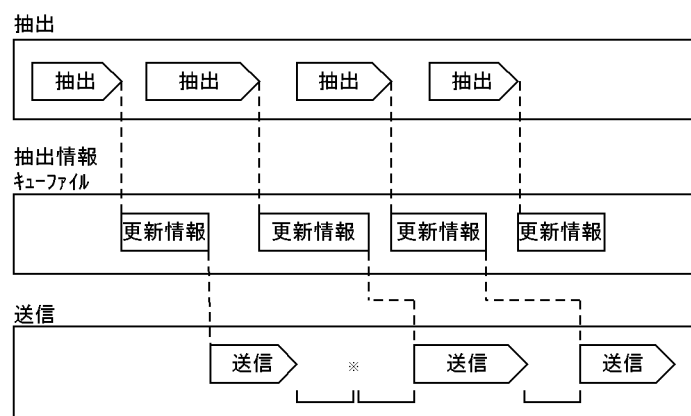
更新情報編集バッファサイズは、送信環境定義の editbufsize オペランドで指定します。

(2) 送信処理の送信間隔の設計

抽出側 Datareplicator では、前回の更新情報を、一定の時間間隔ごとに送信先に送信します。この時間間隔を**送信間隔**といいます。送信間隔は、送信環境定義の sendintvl オペランドで指定します。

送信処理の送信間隔の概要を次の図に示します。

図 4-34 抽出処理の送信間隔の概要



(凡例)

- 抽出又は送信
- 更新情報
- 送信間隔

注※ 1 回目の送信間隔時には、まだ更新情報の抽出が完了していないため、更新情報は送信されません。2 回目の送信間隔時には、更新情報の抽出が完了しているため、更新情報が送信されます。

(3) コネクションリトライ回数

抽出側 Datareplicator では、送信先の反映側システムとの接続に失敗したときに、接続を再実行できます。これを**コネクションリトライ**といいます。送信環境定義の retrynum でコネクションリトライ回数を指定します。コネクションリトライ回数を指定していない場合は、次の送信間隔まで待機し、送信間隔ごとに再接続を実行します。

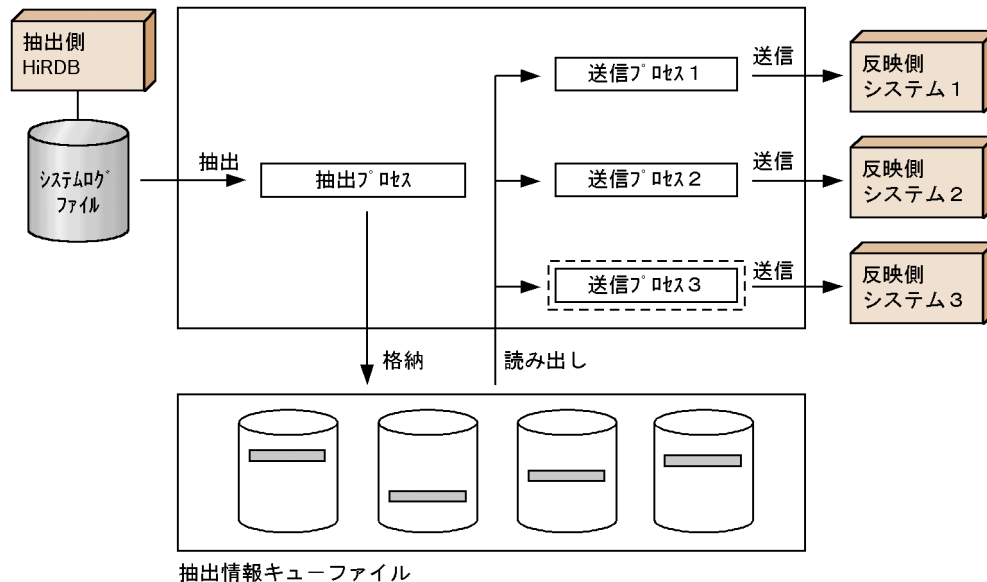
(4) 送信処理の縮退の設計

抽出側 Datareplicator では、一つの抽出環境から複数の送信先に更新情報を送信できます。何らかの原因によって更新情報が抽出情報キューファイルに残ったままの状態が続くと、抽出情報キューファイルのスワップができなくなり、システムログファイルから更新情報を抽出できなくなります。

抽出側 Datareplicator では、抽出情報キューファイルのスワップができない場合に、特定の送信先に対応するデータ連動だけを中止し、送信対象の更新情報をスワップ先になる抽出情報キューファイルから破棄して、抽出情報キューファイルのスワップを可能にします。これを**送信処理の縮退**といいます。送信処理の縮退は、送信環境定義の `overwrite` オペランドで指定します。

送信処理の縮退の概念を次の図に示します。

図 4-35 送信処理の縮退の概念



(凡例)

- ⎓ : 縮退を指定している送信先の送信プロセス。
- : 送信プロセス3で送信の対象となる更新情報。
送信プロセス3は縮退対象のため、抽出情報キューファイルの満杯時には、ほかの更新情報を上書きできるようになります。

(a) 送信処理の縮退を指定した場合の動作

抽出情報キューファイルのスワップができない状態で、さらに送信処理の縮退を指定した送信先の、送信対象の更新情報が、スワップ先になる抽出情報キューファイルに残っている場合、その送信プロセスだけを停止し、対応する更新情報をスワップ先になる抽出情報キューファイルから破棄します。これによって、抽出情報キューファイルのスワップが可能になって、引き続きシステムログファイルからの抽出を続行できます。

送信処理の縮退が発生した場合には、縮退の対象になる送信先の反映側 DB と抽出側 DB との整合性は保証されません。このため、送信先の反映側 DB を再作成する必要があります。

一部の送信先の、抽出側 DB と反映側 DB との整合性よりも、システム全体のデータ連動を優先したい場合には、送信処理の縮退を指定します。

(b) 送信処理の縮退を指定していない場合の動作

抽出情報キューファイルがスワップできなくなった場合には、反映側システムへの更新情報の送信によって、スワップ先になる抽出情報キューファイルに空きができるまで、システムログファイルからの更新情報の抽出を停止します。このため、抽出側 DB と反映側 DB の整合性は保証されます。

データ連動システム全体の抽出側 DB と反映側 DB の整合性を保証したい場合には、縮退を指定しません。

(5) 送信環境定義の単位の設計

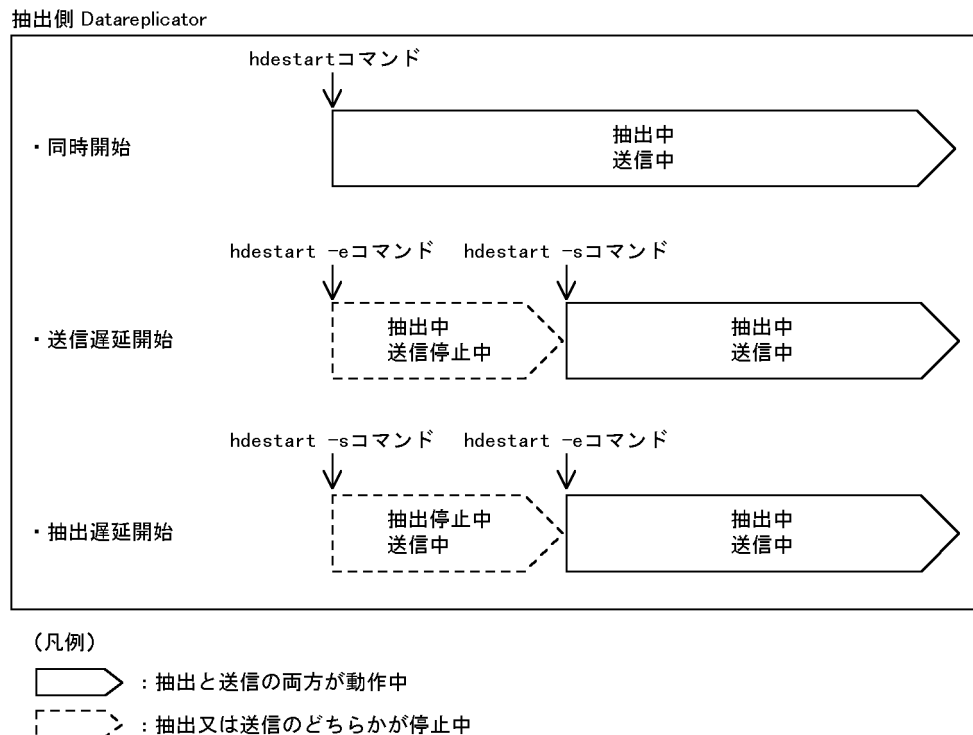
抽出側 HiRDB がパラレルサーバの場合、抽出側 Datareplicator ではバックエンドサーバ単位に送信環境定義を定義します。この場合、送信環境定義では、すべてのバックエンドサーバに対しての共通の環境と、特定のバックエンドサーバに対しての環境を分けて定義できます。

共通の環境については、送信環境定義の `commondef` の下で定義し、特定の環境については、`besdef` (サーバ名) の下で定義します。同一のオペランドを `commondef` と `besdef` とで指定した場合、`besdef` に対応するバックエンドサーバでは、`besdef` での指定を優先します。

4.6.5 抽出処理の開始方式の設計

抽出処理の開始方式を設計します。抽出処理の開始方式には、同時開始、送信遅延開始、及び抽出遅延開始があります。抽出処理の開始方式と開始タイミングの関係を次の図に示します。

図 4-36 抽出処理の開始方式と開始タイミングの関係



(1) 同時開始

抽出側 Datareplicator の起動と同時に、抽出と送信を開始します。同時開始とすると、システムログファイルの更新情報を抽出した順に逐次、送信します。

使用目的

システムログファイルの更新情報を抽出した順に逐次、反映側システムに送信する場合に有効です。

コマンドの実行

hdestart コマンドを実行します。

(2) 送信遅延開始

抽出側 Datareplicator の起動時には抽出だけを開始し、hdestart -s コマンドによって送信の開始が要求された時点で、送信を開始します。反映側システムが起動されていないときや、反映側システムの負荷が高い時間を避けて運用するときに使います。

送信遅延開始とすると、抽出情報キューファイルに格納されている更新情報は送信されないで、抽出した更新情報が格納されていくため、抽出情報キューファイルの容量を十分に確保する必要があります。

使用目的

反映側システムが起動されていないときや、反映側システムの負荷が高い時間を避けて運用するときに有効です。

コマンドの実行

hdestart -e コマンドを実行します。送信の開始時には、hdestart -s コマンドを実行します。hdestart -s コマンドに送信先識別子を指定すると、特定の送信先への送信だけを開始することもできます。

(3) 抽出遅延開始

抽出側 Datareplicator の起動時には送信だけを開始し、hdestart -e コマンドによって抽出の開始が要求された時点で、抽出を開始します。抽出側 HiRDB のオンライン中の負荷を考慮して運用するときに使います。

抽出遅延開始とすると、抽出側 HiRDB のシステムログファイルから更新情報を抽出しないため、システムログファイルが抽出完了になりません（スワップ先になりません）。このため、抽出側 HiRDB で大量のトランザクションを処理することが予想される場合には、システムログファイルの容量を十分確保する必要があります。

使用目的

抽出側 HiRDB のオンライン中の負荷を考慮して運用するときに有効です。

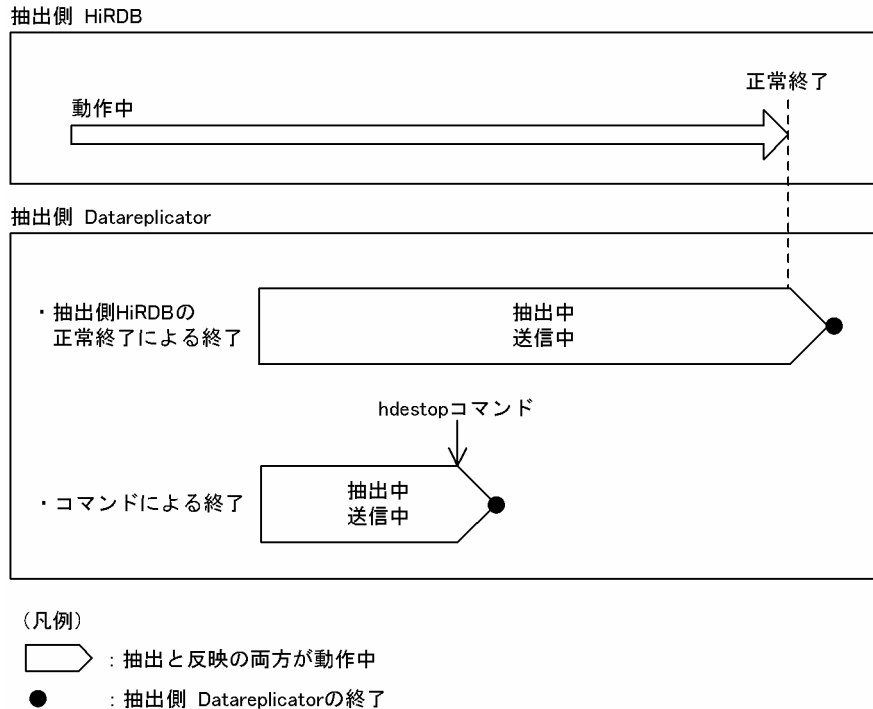
コマンドの実行

hdestart -s コマンドを実行します。hdestart -s コマンドに送信先識別子を指定すると、特定の送信先への送信だけを開始することもできます。抽出の開始時には、hdestart -e コマンドを実行します。

4.6.6 抽出処理の停止方式の設計

抽出処理の停止方式を設計します。抽出処理の停止方式は**同時停止**になります。同時停止とは、抽出処理と送信処理の両方を停止し、抽出側 Datareplicator を終了することです。抽出処理の停止方式と停止タイミングの関係を次の図に示します。

図 4-37 抽出処理の停止方式と停止タイミングの関係



(1) 抽出側 HiRDB の正常終了による停止 (正常終了)

抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator を終了します。

使用目的

コマンドを実行しないで、抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator を終了する場合に使います。

抽出システム定義の設定

syncterm オペランドに true を設定します。

終了する条件

- 抽出側 HiRDB の正常終了を検知した
- 検知した正常終了の後に、抽出側 HiRDB でトランザクションが発生していない
- すべての抽出プロセスが正常に稼働していて、システムログファイルのすべての更新情報の、抽出情報キューファイルへの抽出が完了している
- すべての送信プロセスが正常に稼働していて、抽出情報キューファイルのすべての更新情報の、反映側システムへの送信が完了している

(2) コマンドによる停止 (強制終了)

hdestop コマンドを実行した時点で実行中の、抽出と送信が完了した時点で、抽出側 Datareplicator を終了します。

使用目的

緊急に抽出側 Datareplicator を終了する場合に使います。

コマンドの実行

hdestop コマンドを実行します。

4.6.7 イベント制御用テーブルの設計

抽出側 Datareplicator から反映側 Datareplicator の動作内容を操作する場合には、**イベント機能**を使います。

イベント機能を使う場合は、反映側 Datareplicator では反映環境定義でイベントコードと対応する処理を指定します。反映側 Datareplicator でのイベントコードの指定については、「[4.7.3 反映方法の設計](#)」を参照してください。

イベント機能を使う場合は、抽出側 Datareplicator で、イベントコードを指定した **hdeevent コマンド** を実行します。hdeevent コマンドを実行すると、イベント制御用テーブルに対して SQL を発行して、イベント制御用の更新情報をシステムログファイルに出力します。抽出側 Datareplicator では、このイベント制御用の更新情報を抽出し、反映側 Datareplicator に送信して、反映側 Datareplicator の動作内容を操作します。抽出側 Datareplicator では、イベント制御用の更新情報を検知した時点で、送信間隔を待たないで、反映側 Datareplicator に対する送信処理を開始します。送信処理の完了後は、次の送信間隔まで送信処理待ちになります。

イベント制御用テーブルは、抽出側 Datareplicator を起動する前に、抽出側 HiRDB に作成しておく必要があります。

(1) イベントコードの指定範囲

指定できるイベントコードの範囲と動作を次の表に示します。

表 4-34 指定できるイベントコードの範囲と動作

範囲	動作
0	送信プロセスを停止します (反映側 Datareplicator には送信されません)。
1~255	指定したイベントコードを、反映側 Datareplicator に送信します。
上記以外	反映側 Datareplicator には送信されません。

(2) hdeevent コマンドを実行したときに発行される SQL 文

hdeevent コマンドを実行すると、次の SQL 文が抽出側 HiRDB に対して発行されます。抽出側 HiRDB がパラレルサーバの場合には、すべてのバックエンドサーバに対して、SQL 文が発行されます。

```
LOCK TABLE "hde_dtbl" IN EXCLUSIVE MODE
UPDATE "hde_dtbl" SET EVNO = ユーザ指定のイベントコード
COMMIT WORK
```

hdeevent コマンドを実行する代わりに、同様の SQL 文を発行して、反映側 Datareplicator に対してイベントを発行することもできます。ただし、イベント制御用テーブルに対しては、上記以外の操作をしないでください。

(3) イベント制御用テーブルの条件

抽出側 Datareplicator を起動する前に必要になる、イベント制御用テーブルの条件を説明します。

(a) 作成するユーザ ID

任意のユーザ ID で、イベント制御用テーブルを作成します。

hdeevent コマンドを実行する場合には、イベント制御用テーブルを作成したユーザ ID を指定します。hdeevent コマンドの実行時にユーザ ID の指定を省略すると、PDUSER 環境変数の設定値が仮定されます。

(b) テーブル名

hde_dtbl で固定です。

(c) テーブルの属性

FIX 属性を指定します。

(d) テーブルの構成

イベント制御用テーブルの構成を次の表に示します。

表 4-35 イベント制御用テーブルの構成

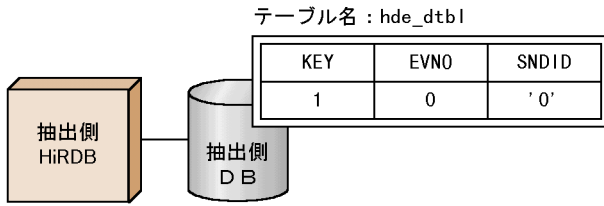
列名	列属性	列長	値
KEY	INTEGER	1	整数
EVNO	INTEGER	1	0
SNDID	CHAR	10	'0'

(e) イベント制御用テーブルの例

- 抽出側 HiRDB がシングルサーバの場合

抽出側 HiRDB がシングルサーバの場合、1 行のテーブルを作成します。抽出側 HiRDB がシングルサーバの場合のイベント制御用テーブルの例を次の図に示します。

図 4-38 抽出側 HiRDB がシングルサーバの場合のイベント制御用テーブルの例



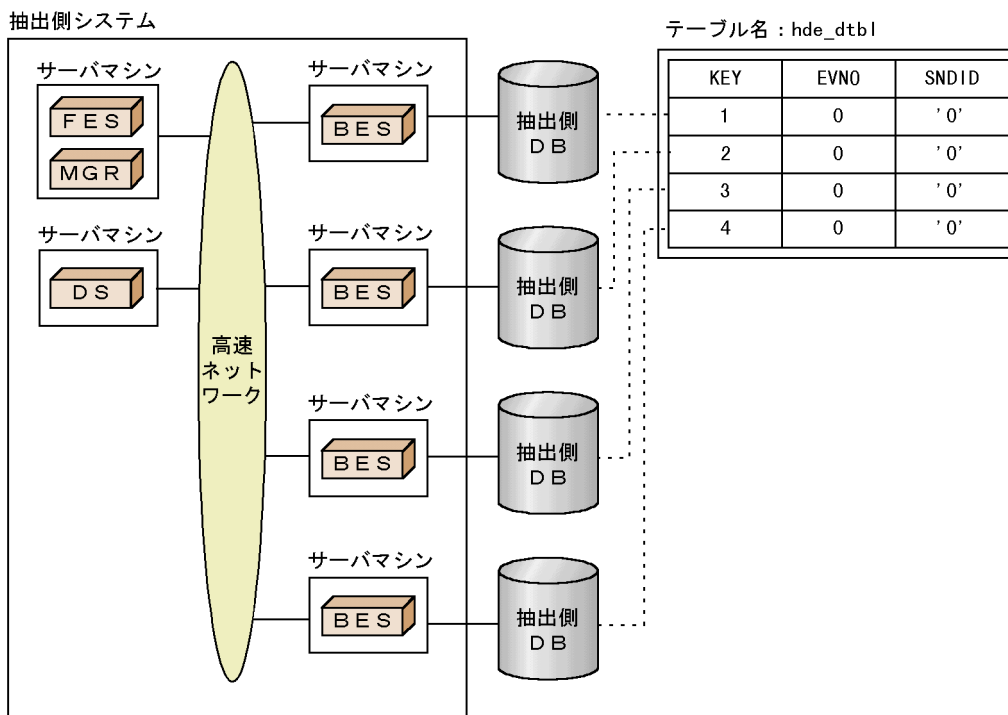
作成時の SQL の例

```
CREATE FIX TABLE "hde_dtbl" ("KEY" INTEGER, "EVNO" INTEGER, "SNDID" CHAR(10))  
INSERT INTO "hde_dtbl" ("KEY", "EVNO", "SNDID") VALUES (1, 0, '0')
```

- 抽出側 HiRDB がパラレルサーバの場合

抽出側 HiRDB がパラレルサーバの場合には、抽出の対象になるすべてのバックエンドサーバに対して、1 行ずつキーレンジ分割してテーブルを作成します。また、RD エリアは、各バックエンドサーバの中から一つ指定します。抽出側 HiRDB がパラレルサーバの場合のイベント制御用テーブルの例を次の図に示します。

図 4-39 抽出側 HiRDB がパラレルサーバの場合のイベント制御用テーブルの例



作成時の SQL の例

```
CREATE FIX TABLE "hde_dtbl" ("KEY" INTEGER, "EVNO" INTEGER, "SNDID" CHAR(10))
  IN ((RDA1)"KEY"=1, (RDA2)"KEY"=2, (RDA3)"KEY"=3, (RDA4)"KEY"=4)
INSERT INTO "hde_dtbl" ("KEY", "EVNO", "SNDID") VALUES (1, 0, '0')
INSERT INTO "hde_dtbl" ("KEY", "EVNO", "SNDID") VALUES (2, 0, '0')
INSERT INTO "hde_dtbl" ("KEY", "EVNO", "SNDID") VALUES (3, 0, '0')
INSERT INTO "hde_dtbl" ("KEY", "EVNO", "SNDID") VALUES (4, 0, '0')
```

(凡例)

MGR : HiRDBのシステムマネージャ FES : HiRDBのフロントエンドサーバ
 BES : HiRDBのバックエンドサーバ DS : HiRDBのディクショナリサーバ

注 RDエリアは、各BESの中から一つ指定します。

(f) 注意

- イベント制御用テーブルは、hdeprep コマンドを実行する前に作成する必要があります。
- 抽出側 HiRDB がパラレルサーバで、バックエンドサーバの追加、削除などの構成変更をした場合、イベント制御用テーブルを再作成する必要があります。イベント制御用テーブルを再作成しなかった場合には、追加したバックエンドサーバからの抽出にイベントが組み込まれないで、抽出側 DB と反映側 DB が不整合になることがあります。
- イベント制御用テーブルを再作成した場合には、hdeprep コマンドを実行する必要があります。
- 抽出側 HiRDB のシステム中に、イベント制御用テーブルの名称である hde_dtbl と、同じ名称のテーブルは作成しないでください。
- イベント制御用テーブルは、抽出対象にできません。
- 一つのトランザクション中に、イベントコードが 1~255 のイベントを複数発行した場合、最後に発行したイベントだけが有効になります。

4.6.8 抽出側 Datareplicator のリソースの設計

抽出側 Datareplicator のディスク資源とメモリ資源の設計方法について説明します。

(1) 抽出側 Datareplicator のディスク資源の設計

抽出側 Datareplicator のディスク資源の一覧を次の表に示します。

表 4-36 抽出側 Datareplicator のディスク資源の一覧

ファイル名		ファイル種別		必要・任意 [個数]	容量の説明先	ファイル満杯時の動作
		R	C			
定義ファイル	抽出システム定義ファイル	○	×	必要 [抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に 1 個)]	(a)	特にありません。
	抽出環境定義ファイル	○	×	必要 [抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に 1 個)]		
	送信環境定義ファイル	○	×	必要 [送信先識別子ごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に送信先識別子ごとに 1 個)]		
	抽出定義ファイル	○	×	必要 [抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に 1 個)]		
	二重化定義ファイル	○	×	任意 [抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に 1 個)]		
抽出定義プリプロセスファイル		○	○※ 2	必要 [抽出側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に 1 個)]	(b)	特にありません。
抽出情報キューファイル※1※3		○	○※ 2	必要 [抽出側システムごとに 2~16 個 (抽出側 HiRDB がパラレルサーバの場合には, BES ごとに 2~16 個)]	(c)	スワップします。※1

ファイル名		ファイル種別		必要・任意 [個数]	容量の説明先	ファイル満杯時の動作
		R	C			
ステータスファイル※3	抽出マスタステータスファイル	○	○※ 2	必要 [抽出側システムごとに1個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に1個)]	(d)	特にありません。
	抽出サーバステータスファイル※1	○	○※ 2	必要 [抽出側システムごとに1個 (抽出側 HiRDB がパラレルサーバの場合には, BES ごとに1個)]		
エラー情報ファイル	抽出マスタエラー情報ファイル	○	×	必要 [抽出側システムごとに2個 (抽出側 HiRDB がパラレルサーバの場合には, MGR に2個)]	(e)	スワップします。
	抽出ノードマスタエラー情報ファイル	○	×	必要 [抽出側システムごと2個 (抽出側 HiRDB がパラレルサーバの場合には, BES が存在するサーバマシンごとに2個)]		
稼働トレースファイル	抽出マスタトレースファイル※4	○	×	任意 [抽出側システムごとに2個]	(f)	スワップします。
	抽出ノードマスタトレースファイル※4	○	×	任意 [抽出側システムごとに2個 (抽出側 HiRDB がパラレルサーバの場合には, BES ごとに2個)]		
データ連動用連絡ファイル※3		○	○※ 2	必要 [抽出側システムごとに1個 (抽出側 HiRDB がパラレルサーバの場合には, BES ごとに1個)]	(g)	特にありません。
コマンドログファイル		○	×	任意 [抽出側システムごとに2個]	(h)	スワップします。

(凡例)

BES : バックエンドサーバ (抽出対象の DB がないバックエンドサーバも含む)

R : UNIX の通常ファイル又は Windows ファイル

C : UNIX のキャラクタ型スペシャルファイル

○ : 作成できます。

× : 作成できません。

注※1

スワップ先のファイルの送信が完了していない場合には、次の送信間隔まで待ちます。次の送信間隔時に送信が完了していれば、更新情報の抽出を開始します。送信間隔については、「4.6.4 送信方法の設計」を参照してください。

注※2

AIX 版でキャラクタ型スペシャルファイルを使用する場合は、各ファイル容量の算出式に 1,024 バイト加算してください。また、抽出環境定義の `queuesize` オペランドには、算出した容量から 1,024 バイト減算した値を指定してください。

注※3

Datereplicator ファイルシステム領域を使用する場合は、「3.4.3(1) Datereplicator ファイルシステム領域を割り当てるときの規則」も参照してください。

注※4

稼働トレース収集方式を全プロセス一括収集方式とする場合のファイル名です。プロセス単位収集方式とする場合の稼働トレースファイル名については、「5.2 抽出システム定義」の `int_trc_unit` オペランドの説明を参照してください。

(a) 定義ファイルの容量

抽出側 Datereplicator の定義ファイル（抽出システム定義ファイル、抽出環境定義ファイル、送信環境定義ファイル、抽出定義ファイル）の容量は、指定した定義オペランドの内容によって異なります。

(b) 抽出定義プリプロセスファイルの容量

抽出定義プリプロセスファイルには、抽出定義で定義した情報が内部形式に変換された情報が格納されます。

抽出定義プリプロセスファイルの容量の見積もり式を次に示します。

- UNIX の通常ファイル、又は Windows ファイルの場合

$$164 + (40 + SV_INFO) \times SV_NUM \text{ (単位: バイト)}$$

- UNIX のキャラクタ型スペシャルファイルの場合

$$\uparrow 164 \div SCT_SIZE \uparrow \times SCT_SIZE + (40 + SV_INFO) \times SV_NUM \text{ (単位: バイト)}$$

- SV_INFO

$$= 48 + (52 + 96 \times SCH_NUM + 64 \times UPD_NUM + 128 \times COL_NUM + 64 + 112 \times TYPE_NUM + 64 \times ATTR_NUM) + 24 + 32 \times TBL_NUM + 24 \times COL_NUM + SND_INFO + LOB_INFO + DEFAULT_INFO$$

- SND_INFO

$$= 16 + 28 \times UPD_NUM + \sum_{SND} (28 \times SND_UPD + 32 \times SND_COND + 272 \times SND_CNST) + 16 + 40 \times SND_NUM + \sum_{SND} (12 \times SND_UPD + 32 \times SND_TBL + 24 \times SND_COL)$$

- DEFAULT_INFO
= 16 + 20×SV_DEF + SUM_DEF
- LOB_INFO
= 16 + 4×LOB_AREA

上記の算出式で使っている変数の意味を次に示します。

- Σ_{SND} ：全送信先数の総和
- SV_NUM：バックエンドサーバ数
- SV_INFO：バックエンドサーバ単位定義情報サイズ
- SCH_NUM：すべての抽出対象スキーマ数
- UPD_NUM：すべての更新情報名数
- TBL_NUM：すべての抽出対象表数
- COL_NUM：すべての抽出対象列数
- TYPE_NUM：抽出対象になる抽象データ型の数。重複分は除きます。抽出対象の抽象データ型が継承している上位データ型と、抽象データ型の属性データタイプとして定義されている抽象データ型の数を含みます。
- ATTR_NUM：TYPE_NUM に計上した抽象データ型の属性数の総和
- SND_INFO：送信先情報サイズ
- SND_NUM：送信先数
- SND_UPD：送信先ごとの送信対象更新情報名数
- SND_TBL：送信先ごとの送信対象表数
- SND_COL：送信先ごとの送信対象列数
- SCT_SIZE：セクタ長
- SND_COND：送信先ごとの送信条件数
- SND_CNST：送信先ごとの送信条件定数指定数
- LOB_INFO：BLOB 抽出情報サイズ
- LOB_AREA：抽出対象 BLOB 列格納 RD エリア数
- SV_DEF：列追加時にデフォルト値を指定したサーバ内の列数
- SUM_DEF：列追加時にデフォルト値を指定したサーバ内の列の DEF_LEN の総和
- DEF_LEN：((デフォルト値の長さ÷4) + 1) × 4

(c) 抽出情報キューファイルの容量

抽出情報キューファイルには、抽出した更新情報が蓄積されます。このため、抽出情報キューファイルの容量は、抽出する更新情報の量より多くする必要があります。

抽出情報キューファイルの容量の見積もり式（更新情報の算出式）を次に示します。ファイルの種別が OS の通常ファイルの場合でも、キャラクタ型スペシャルファイルの場合でも、抽出情報キューファイルの容量の見積もり式は同じです。

- 更新情報の算出式

$$\text{EXT_ALL} + \text{QFL_NUM} \times \text{SCT_SIZE} + (\text{EXT_ALL} \div \text{IO_SIZE}) \times 12 \text{ (単位: バイト)}$$

- EXT_ALL

$$= \Sigma \text{ins} (\text{INS_SIZE}) + \Sigma \text{upd} (\text{UPD_SIZE}) + \Sigma \text{del} (\text{DEL_SIZE}) + 168 \times \text{TRN_NUM} + 122 \times \text{EVT_NUM}$$

- INS_SIZE

$$= 100 + \text{EXT_COL} \times 12 + \text{ROW_SIZE}$$

- UPD_SIZE

$$= 100 + \text{EXT_COL} \times 12 + \text{ROW_SIZE}$$

- DEL_SIZE

$$= 100 + \text{MAP_COL} \times 12 + \Sigma \text{map} (\text{MAP_DATA})$$

- ROW_SIZE

$$= \Sigma \text{ext} (\text{COL_DATA} + \text{LOB_DATA} + \text{VAR_DATA} + \text{ADT_DATA} + \text{MLT_DATA})$$

- COL_DATA

$$= \uparrow \text{列データ長} \div 4 \uparrow \times 4$$

- LOB_DATA

$$= 56 \times \uparrow (\text{BLOB データ長} \div 8192) \uparrow + \text{BLOB データ長}$$

- VAR_DATA

$$= 48 \times \uparrow (\text{VARCHAR データ長} \div \text{PAGE_SIZE}) \uparrow + \text{VARCHAR データ長}$$

- ADT_DATA

$$= 48 \times \text{ADT_NUM} + \text{抽象データ型データ長}$$

抽象データ型データ長については、HiRDB の該当するマニュアルを参照してください

- MLT_DATA

$$= 58 \times \uparrow \uparrow \text{最大要素数} \div 8 \uparrow \div 4 \uparrow \times 4 + 52 \times \uparrow \text{繰返し列データ長} \div \text{PAGE_SIZE} \uparrow + \text{繰返し列データ長}$$

繰返し列データ長：

- ・固定長要素のとき：1 + 要素定義長 × 要素数
- ・可変長要素のとき： $\Sigma \text{elm} (5 + \text{VAR_DATA})$

上記の算出式で使っている変数の意味を次に示します。

- EXT_ALL：全更新情報サイズ
- SCT_SIZE：セクタ長

- IO_SIZE：抽出環境定義の quiosize オペランドの設定値
- INS_SIZE：単位挿入行サイズ
- UPD_SIZE：単位更新行サイズ
- DEL_SIZE：単位削除行サイズ
- ROW_SIZE：1 行当たりの更新データ長
- QFL_NUM：抽出情報キューファイル総数
- TRN_NUM：抽出対象の HiRDB に対する実行トランザクション数（抽出対象外の表に対するトランザクション、無効トランザクション Rollback を含みます）
- EVT_NUM：イベント発行数
- EXT_COL：行単位当たりの抽出対象列数
- MAP_COL：行単位当たりのマッピングキー列数
- LOB_DATA：BLOB 列の更新データ長
- VAR_DATA：256 バイト以上の VARCHAR 列（MVARCHAR, NVARCHAR を含みます）の更新データ長
- PAGE_SIZE：HiRDB の DB ページサイズ
- ADT_DATA：抽象データ型列の更新データ長
- ADT_NUM：抽象データ型の数（このデータ型に関連する継承データ型、ネストデータ型の列を含みます）
- MLT_DATA：繰返し列の更新データ長
- COL_DATA：列単位の更新データ長（BLOB 列、256 バイト以上の VARCHAR 列、抽象データ型列を除きます）
- MAP_DATA：マッピングキーサイズ
- Σ_{ins} ：全挿入行サイズの総和
- Σ_{upd} ：全更新行サイズの総和
- Σ_{del} ：全削除行サイズの総和
- Σ_{map} ：1 行当たりのマッピングキーサイズの総和
- Σ_{ext} ：1 行当たりの抽出対象列データサイズの総和
- Σ_{elm} ：1 行当たりの繰返し列要素データサイズの総和

- **抽出情報キューファイルの容量見積もり時の考慮点**

抽出情報キューファイルの容量見積もり時には、次に示すことを考慮してください。

- 抽出情報キューファイルのサイズは、反映側システムがダウンして復旧に時間が掛かる場合を想定し、3~4 日分の業務で発生する更新情報を格納できるサイズを用意してください。

- 次のようなトランザクションが発生し、抽出情報キューファイルに空きがなくなった場合、抽出側、反映側の DB の内容を一致させた後、Datareplicator の初期化が必要となります。
 - ・ バッチなどのトランザクションを実行し、一つのトランザクション内で発生した更新情報を抽出情報キューファイルに格納できない。
 - ・ 抽出側 DB を更新したトランザクションを決着しないで放置し、ほかの複数のトランザクションで発生したすべての更新情報が抽出情報キューファイルに格納できない。
- 抽出処理の運用方式による容量の違い

抽出処理の運用方式による容量見積りの考慮点を次の表に示します。なお、抽出情報キューファイルは、1 トランザクションの最大サイズ（最も大量に更新されるデータ量の総サイズ）以上の容量になるようにする必要があります。

一つのトランザクションが抽出情報キューファイルに収まらない場合、抽出処理が進まなくなります。このため、HiRDB のシステムログファイルに未抽出のログが溜まり、システムログファイルが満杯になると、HiRDB がサーバダウンします。

表 4-37 抽出処理の運用方式による容量見積りの考慮点

運用方式	考慮点
同時開始, 抽出遅延開始	<ul style="list-style-type: none"> • 抽出側 Datareplicator での抽出の遅延を考慮して、トランザクションのピーク時に蓄積されるデータ量をディスク容量とします。 • ファイルを 2 個用意すれば、運用は可能と考えられます。ただし、何らかの要因で送信処理だけを停止する可能性がある場合には、ファイルの数を増やしておくことをお勧めします。
送信遅延開始	<ul style="list-style-type: none"> • 送信が開始される前に蓄積されるデータ量と送信開始後も蓄積されるデータ量を考慮してディスク容量を算出します。 • 同時開始と送信遅延開始を併用する場合には、ファイルのサイズを小さめにしてファイルの数を多くした方が、ファイルの満杯を回避しやすいこともあります。

(d) ステータスファイルの容量

抽出側 Datareplicator のステータスファイル（抽出マスタステータスファイル、抽出サーバステータスファイル）の容量の算出式を次に示します。

ステータスファイルには、抽出側 Datareplicator の初期化時に必要なサイズが割り当てられます。実際のレプリケーション運用中にファイル容量が増加し満杯となることはありません。

- 抽出マスタステータスファイルの容量の算出式
 - UNIX の通常ファイル, 又は Windows ファイルの場合

$$(2 + \text{SND_NUM} + \text{ND_NUM} \times \text{SV_NUM}) \times 1024 \text{ (単位: バイト)}$$

- UNIX のキャラクタ型スペシャルファイルで、セクタ長が 1KB を超える場合

$$(2 + \text{SND_NUM} + \text{ND_NUM} \times \text{SV_NUM}) \times \text{SCT_SIZE} \text{ (単位: バイト)}$$

ND_NUM : 全ノード数

SV_NUM : 全サーバ数

SND_NUM：全送信先数

SCT_SIZE：セクタ長

- 抽出サーバステータスファイルの容量の算出式

抽出側 HiRDB がパラレルサーバの場合には、バックエンドサーバごとにステータスファイルの容量を見積もります。

- UNIX の場合

$$(22 + \text{SND_NUM} \times 2 + \text{UINF_NUM} + \text{SYS_NUM}) \times 1024 \text{ (単位: バイト)}$$

- Windows の場合

$$(22 + \text{SND_NUM} \times 2 + \text{UINF_NUM}) \times 1024 \text{ (単位: バイト)}$$

- UNIX のキャラクタ型スペシャルファイルで、セクタ長が 1KB を超える場合

$$(22 + \text{SND_NUM} \times 2 + \text{UINF_NUM} + \text{SYS_NUM}) \times \text{SCT_SIZE} \text{ (単位: バイト)}$$

SND_NUM：送信先数

UINF_NUM：抽出システム定義の extinfo_num オペランドの設定値

SCT_SIZE：セクタ長

SYS_NUM：抽出環境定義に指定されたファイルシステム領域数

(e) エラー情報ファイルの容量

抽出側 Datareplicator のエラー情報ファイル（抽出マスタエラー情報ファイル，抽出ノードマスタエラー情報ファイル）の容量の既定値は 16 キロバイトです。長時間のエラー情報を残したいときには，抽出システム定義を変更してファイルの容量を拡張してください。

(f) 稼働トレースファイルの容量

稼働トレースファイルの容量は，抽出システム定義の int_trc_filesz オペランドで指定します。

(g) データ連動用連絡ファイルの容量

データ連動用連絡ファイルの容量の算出式を次に示します。抽出側 HiRDB がパラレルサーバの場合には，バックエンドサーバごとにデータ連動用連絡ファイルの容量を見積もります。

- HP-UX のキャラクタ型スペシャルファイルで，セクタ長が 1KB を超える場合

$$9 \times \text{SCT_SIZE} \text{ (単位: バイト)}$$

- 上記以外のキャラクタ型スペシャルファイル，UNIX の通常ファイル，又は Windows ファイルの場合

$$9 \times 1024 \text{ (単位: バイト)}$$

SCT_SIZE：セクタ長

(h) コマンドログファイルの容量

128 キロバイト固定です。

(i) その他のファイル容量

Windows 版では、インストールディレクトリ下の tmp ディレクトリ下に、ワークファイルを幾つか作成します。このため、ワークファイル分として 4MB の容量を見積もってください。

(2) 抽出側 Datareplicator のメモリ資源の設計

抽出側 Datareplicator のメモリ資源の一覧を次の表に示します。

表 4-38 抽出側 Datareplicator のメモリ資源の一覧

メモリ資源	容量の説明先
抽出マスタプロセス	(a)
抽出ノードマスタプロセス	(b)
抽出プロセス	(c)
送信プロセス	(d)
送信マスタプロセス	(e)
稼働トレース収集プロセス	(f)
抽出処理監視プロセス	(g)
Datareplicator エージェントプロセス	(h)
コマンド連絡用共用メモリサイズ	(i)
ステータス情報連絡用共用メモリサイズ	(j)
プロセス間連絡用共用メモリサイズ	(k)
抽出定義格納用共用メモリサイズ	(l)
メッセージ格納用共用メモリサイズ	(m)
トランザクション管理情報格納用共用メモリサイズ	(n)
セマフォ所要数	(o)
メッセージキュー所要数	(p)
更新情報編集バッファサイズ	(q)

表 4-39 抽出側 Datareplicator のメモリ見積もり式に使用する変数の一覧

変数名	変数の内容
ATTR_NUM	データ連動の対象となる全抽象データ型の属性数の総和
COL_LEN	抽出対象表の抽出対象列の合計長のうち、最大長。

変数名	変数の内容
	ただし、以下に示す列の場合、以下の長さを適用し、合計長を算出する。 <ul style="list-style-type: none"> 繰返し列：1 要素分の定義長 BLOB/BINARY 列： <ul style="list-style-type: none"> 35000 バイト以下の場合：列定義長 35000 バイト超の場合：35000 バイト ADT の場合：0
COL_NUM	全抽出対象表の列数の合計数
COM_FILE	データ連動用連絡ファイルサイズ (バイト)
DATA_NUM	送信間隔中に検知する最大更新情報件数。 バッチなど、処理時間が送信間隔より長いトランザクションが存在する場合、トランザクションが決着されるまで更新情報件数として蓄積する。
EMST_FILE	抽出マスタステータスファイルサイズ (バイト)
EST_FILE	抽出サーバステータスファイルサイズ (バイト)
EXT_FILE	抽出環境定義ファイルサイズ (バイト)
LOB_MAX	抽出対象の BLOB 属性の列のうち、最大更新データ長 (バイト) (列定義長ではなく実データ長)
LOGIOSIZE	抽出環境定義 logiosize 指定値 (注意 (KB)のまま指定する)
MSV_NUM	<抽出元が HiRDB の場合> 抽出対象の各ノード内のバックエンドサーバ数のうち、最大数 <抽出元が HiRDB 以外の場合> 1
ND_NUM	抽出対象の全ノード数
NSV_NUM	該当ノード内のサーバ数
PRP_FILE	抽出定義プリプロセスファイルサイズ (バイト)
QUE_NUM	抽出環境定義 qufileXXX の指定数
QUIOSIZE	抽出環境定義 quiosize の指定値 (注意 (KB)のまま指定する)
RBUF_NUM	送信環境定義 readbufnum の指定値
RDAMAX_ALL	連動対象表を格納している RD エリアの最大ページ長 (バイト)
RDAMAX_BIN	BINARY を格納している RD エリアの最大ページ長 (バイト)
RDAMAX_LOB	LOB を格納している RD エリアの最大ページ長 (バイト)
RDAMAX_REP	繰返し列を含む表を格納している RD エリアの最大ページ長 (バイト)
RDAMAX_VCHR	varchar 型,もしくは nvarchar 型,もしくは mvarchar 型を含む表を格納している RD エリアの最大ページ長 (バイト)

変数名	変数の内容
SCH_NUM	抽出対象の認可識別子の合計数
SND_FILE	送信環境定義ファイルサイズ (バイト)
SND_NUM	抽出システム定義 sendidxx の指定数
SV_NUM	<p><抽出元が HiRDB の場合> 抽出対象の全バックエンドサーバ数</p> <p><抽出元が HiRDB 以外の場合> 1</p>
SYS_FILE	抽出システム定義ファイルサイズ (バイト)
TBL_NUM	抽出対象表の合計数
TRN_NUM	<p>送信間隔中に検知する最大トランザクション数。</p> <p>バッチなど、処理時間が送信間隔より長いトランザクションが存在する場合、トランザクションが決着されるまでトランザクション数として蓄積する。</p>
TYPE_NUM	全抽出対象表の抽象データ型の合計数
UBUF	<p><抽出側 DB が HiRDB の場合></p> <ul style="list-style-type: none"> 抽出システム定義 sendcontrol に nodemst を指定した場合 送信環境定義 editbufsize 指定値 抽出システム定義 sendcontrol に sendmst を指定した場合 抽出システム定義 smt_editbufsize 指定値 <p><抽出側が XDM/DS の場合> XDM/DS 起動定義 REFLECTBUFF 指定値。 詳細は、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。 (注意 (KB)のまま指定する)</p>
UINF_NUM	抽出システム定義 extinfunum の指定値

(a) 抽出マスタプロセス

• プロセス数

抽出側 HiRDB がシングルサーバのときは一つだけ必要です。抽出側 HiRDB がパラレルサーバのときには、システムマネージャが存在するサーバマシンに一つ必要です。

• プロシジャサイズ

163840 バイト

• 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

$$\begin{aligned}
 & (\downarrow \text{ND_NUM} / 8 \downarrow + 1) \times 8800 \\
 & + (\downarrow \text{SV_NUM} / 16 \downarrow + 1) \times 17600
 \end{aligned}$$

+ PRP_FILE × 5
 + MSV_NUM × (39196 + SND_NUM × 3968 + UINF_NUM × 1024)
 + (76 + 32 × SND_NUM) × MSV_NUM
 + (8 + SND_NUM + 18 × SV_NUM) × 132
 + MAX (SYS_FILE , EXT_FILE , SND_FILE)
 + 58658

(可変部)

- 反映トランザクション同期機能を使用する場合に加算する式

48 × UINF_NUM
 + 10 × SV_NUM
 + 1024 × SV_NUM
 + 1494

- ファイル二重化機能を使用する場合に加算する式

1024

変数の説明

表 4-39 を参照してください。

(b) 抽出ノードマスタプロセス

- プロセス数

抽出側 HiRDB がシングルサーバのときには一つだけ必要です。抽出側 HiRDB がパラレルサーバのときには、各バックエンドサーバが存在するサーバマシンに一つずつ必要です。

- プロシジャサイズ

71680 バイト

- 動的メモリサイズ (単位：バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

(100 × NSV_NUM + 744 × SND_NUM × NSV_NUM)
 + (32 × SND_NUM × NSV_NUM)
 + (2 × SND_NUM)
 + QUIOSIZE × 1024
 + (224 + 2040 + MSV_NUM × (39196 + SND_NUM × 3968 + UINF_NUM
 × 1024) + PRP_FILE)
 + (MAX (QUIOSIZE × 1024 + 516 , 66052))
 + (↓ QUIOSIZE × 1024 / 1024 ↓ × 1024)
 + (MAX (EST_FILE , COM_FILE) - 1024)
 + 545086

(可変部)

- ファイル二重化機能を使用する場合に加算する式
(QUIOSIZE × 1024)

変数の説明

表 4-39 を参照してください。

(c) 抽出プロセス

- プロセス数

抽出側 HiRDB がシングルサーバのときには一つだけ必要です。抽出側 HiRDB がパラレルサーバのときには、各バックエンドサーバに一つずつ必要です。

- プロシジャサイズ

153600 バイト

- 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

$$\begin{aligned} & (1024 \times (UINF_NUM + 1)) \\ & + (2944 \times SND_NUM) \\ & + LOGIOSIZE \times 1024 \\ & + (\uparrow \uparrow SND_NUM / 8 \uparrow / 4 \uparrow \times 4) \\ & + (12 \times QUE_NUM) \\ & + (LOGIOSIZE \times 2048) \\ & + (MAX ((4184 + \uparrow \uparrow SND_NUM / 8 \uparrow / 4 \uparrow \times 4 \\ & + 15 \times COL_NUM + COL_LEN + 4096) , (LOGIOSIZE \times 2048))) \\ & + (MAX (QUIOSIZE \times 1024 + 516 , 66052) + (QUIOSIZE \times 1024)) \\ & + (\downarrow QUIOSIZE \times 1024 / 1024 \downarrow \times 1024) \\ & + (MAX (EST_FILE , COM_FILE) - 1024) \\ & + (LOGIOSIZE \times 1024) \\ & + 744992 \end{aligned}$$

(可変部)

- ファイル二重化機能を使用する場合に加算する式
QUIOSIZE × 1024

変数の説明

表 4-39 を参照してください。

(d) 送信プロセス

- 抽出システム定義の sendcontrol オペランドに nodemst を指定した場合

- プロセス数

抽出側 HiRDB がシングルサーバのときには、送信先の数だけ必要です。抽出側 HiRDB がパラレルサーバのときには、各バックエンドサーバに、送信先の数だけ必要です。

- プロシジャサイズ

378880 バイト

- 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

(UBUF × 1024 - 120)
+ (12 × QUE_NUM)
+ MAX (88 + ↑ SND_NUM / 32 ↑ × 4 , 1244)
+ (QUIOSIZE × 1024)
+ ((TRN_NUM × 340 + DATA_NUM × 64)) . . . ※
+ ↑ (SND_NUM / 32) ↑ × 4
+ UBUF × 1024
+ 32 × TBL_NUM
+ 96 × SCH_NUM + 64 × TBL_NUM + 128 × COL_NUM
+ MAX (QUIOSIZE × 1024 + 516 , 66052) + QUIOSIZE × 1024
+ (↓ QUIOSIZE × 1024 / 1024 ↓ × 1024) × RBUF_NUM
+ (EST_FILE - 1024)
+ 18604

(重要)

注※が示す式については、見積もり量より多くのトランザクション、あるいは更新情報が発生すると、送信プロセスが自動的にメモリを拡張します。この拡張回数に制限はないため、抽出側DBに対してバッチ処理など1トランザクションで多量の更新を行なう場合、送信プロセスのサイズが肥大化する場合があります。バッチ処理をデータ連携する場合、定期的に Commit 処理を行い、1トランザクションでの更新件数を数万件程度に抑えるようにしてください。

また、AIXにおいて、見積もりサイズが256メガバイトを超える場合、環境変数 LDR_CNTRL を設定してください。設定方法については、「[2.4.1 抽出側 Datareplicator の環境変数](#)」及び、「[2.5.1 抽出側 Datareplicator の通信環境の設定](#)」を参照してください。

(可変部)

- 抽出対象列に以下のデータ型が含まれる場合に加算する式
 - varchar,nvarchar,mvarchar,blob,binary,繰り返し列
分岐データ数 × 108
※ 分岐データの目安: : 可変長列の全データ長/RD エリアの平均値
 - varchar,nvarchar,mvarchar,BLOB 型を使用する場合は、更に以下の内容を加算してください。

$\text{MAX}(\text{RDAMAX_VCHR}, 8192)$

- ・ BLOB 型を使用する場合は、更に以下の内容を加算してください。

$1024 \times \uparrow \uparrow \text{LOB_MAX} / \text{RDAMAX_LOB} \uparrow / 16 \uparrow$

- ・ 繰り返し列を使用する場合は、更に以下の内容を加算してください。

$\text{RDAMAX_REP} \times 2$

- ・ binary 型を使用する場合は、更に以下の内容を加算してください。

RDAMAX_BIN

- ・ SGML 型を使用する場合は、更に以下の内容を加算してください。

$64 + 112 \times \text{TYPE_NUM} + 64 \times \text{ATTR_NUM}$

- ・ 反映トランザクション同期機能を使用する場合は加算する式

$32 + \text{SV_NUM}$

- ・ 送信 uoc を使用する、もしくは SGML 型を使用する場合は加算する式

$(68 \times \text{TBL_NUM} + 16 \times \text{COL_NUM}) + (64 \times \text{COL_NUM})$

- ・ HDE_BIN_COL_MAXLEN 環境変数を指定する、もしくは送信 UOC を使用する場合は加算する式

$20 \times \text{TBL_NUM} + 8 \times \text{COL_NUM}$

- ・ 抽出定義に ukey を指定している場合は加算する式

RDAMAX_ALL

- ・ 文字集合を使用する場合は加算する式

32032

- ・ ファイルシステム領域を使用する、もしくは AIX でキャラクタ型スペシャルファイルを使用する場合は加算する式

4608

- ・ ファイル二重化機能を使用する場合は加算する式

$29736 + (\text{QUIOSIZE} \times 1024)$

変数の説明

表 4-39 を参照してください。

●抽出システム定義の sendcontrol オペランドに sendmst を指定した場合

- ・ プロセス数

抽出システム定義の sendprocnum オペランドに指定した数だけ必要です。

- ・ プロシジャサイズ

307200 バイト

- ・ 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

(UBUF × 1024 - 120)
 + (12 × QUE_NUM)
 + MAX (88 + ↑ SND_NUM / 32 ↑ × 4 , 1244)
 + (QUIOSIZE × 1024)
 + UBUF × 1024
 + 40 + 32 × TBL_NUM
 + 52 + 96 × SCH_NUM + 64 × TBL_NUM + 128 × COL_NUM
 + 2428 + MAX (QUIOSIZE × 1024 + 516 , 66052)
 + QUIOSIZE × 1024 + 516
 + 11808 + (↓ QUIOSIZE × 1024 / 1024 ↓ × 1024) × RBUF_NUM
 + 18588

(可変部)

- 抽出対象列に以下のデータ型が含まれる場合に加算する式
 - varchar,nvarchar,mvarchar,lob,binary,繰返し列
分岐データ数 × 108
※ 分岐データの目安：：可変長列の全データ長/RD エリアの平均値
 - varchar,nvarchar,mvarchar,BLOB 型を使用する場合は、
更に以下の内容を加算してください。
MAX (RDAMAX_VCHR , 8192)
 - BLOB 型を使用する場合は、更に以下の内容を加算してください。
1024 × ↑ ↑ LOB_MAX / RDAMAX_LOB ↑ / 16 ↑
 - 繰返し列を使用する場合は、更に以下の内容を加算してください。
RDAMAX_REP × 2
 - binary 型を使用する場合は、更に以下の内容を加算してください。
RDAMAX_BIN
 - SGML 型を使用する場合は、更に以下の内容を加算してください。
64 + 112 × TYPE_NUM + 64 × ATTR_NUM
- 反映トランザクション同期機能を使用する場合に加算する式
32 + SV_NUM
- 送信 uoc を使用する、もしくは SGML 型を使用する場合に加算する式
(68 × TBL_NUM + 16 × COL_NUM) + (64 × COL_NUM)
- HDE_BIN_COL_MAXLEN 環境変数を指定する、もしくは送信 UOC を使用する場合に加算する式
20 × TBL_NUM + 8 × COL_NUM
- 抽出定義に ukey を指定している場合に加算する式
RDAMAX_ALL
- 文字集合を使用する場合に加算する式

32032

- ファイルシステム領域を使用する、もしくは AIX でキャラクタ型スペシャルファイルを使用する場合に加算する式

4608

- ファイル二重化機能を使用する場合に加算する式

$29736 + (\text{QUIOSIZE} \times 1024)$

変数の説明

表 4-39 を参照してください。

(e) 送信マスタプロセス

- プロセス数

抽出側 HiRDB がシングルサーバのときには一つだけ必要です。抽出側 HiRDB がパラレルサーバのときには、各バックエンドサーバに一つずつ必要です。

- プロシジャサイズ

256000 バイト

- 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

$(12 \times \text{QUE_NUM})$
 $+ \text{MAX}(88 + \uparrow \text{SND_NUM} / 32 \uparrow \times 4, 1244)$
 $+ (\text{QUIOSIZE} \times 1024)$
 $+ ((\text{TRN_NUM} \times 340 + \text{DATA_NUM} \times 64) \times \text{SND_NUM})$
 $+ \uparrow (\text{SND_NUM} / 32) \uparrow \times 4$
 $+ \text{MAX}(\text{QUIOSIZE} \times 1024 + 516, 66052) + \text{QUIOSIZE} \times 1024$
 $+ \downarrow \text{QUIOSIZE} \times 1024 / 1024 \downarrow \times 1024$
 $+ (\text{EST_FILE} - 1024)$
 $+ 18512$

(可変部)

- ファイルシステム領域を使用する、もしくは AIX でキャラクタ型スペシャルファイルを使用する場合に加算する式

4608

- ファイル二重化機能を使用する場合に加算する式

$11736 + (\text{QUIOSIZE} \times 1024)$

変数の説明

表 4-39 を参照してください。

(f) 稼働トレース収集プロセス

- プロセス数
「抽出マスタプロセス数+抽出ノードマスタプロセス数」だけが必要です。
- プロシジャサイズ
61440 バイト
- 動的メモリサイズ
51200 バイト

(g) 抽出処理監視プロセス

- プロセス数
抽出ノードマスタプロセス数だけが必要です。
- プロシジャサイズ
61440 バイト
- 動的メモリサイズ
51200 バイト

(h) Datareplicator エージェントプロセス

- プロセス数
抽出システム又は反映側システムのどちらか一方で加算します。
- プロシジャサイズ
71680 バイト
- 動的メモリサイズ (単位: バイト)
 $300000※ + (ND_NUM + SV_NUM + SND_NUM + 2) \times 64$

変数の説明

表 4-39 を参照してください。

注※

抽出システム又は反映側システムのどちらか一方で加算します。

(i) コマンド連絡用共用メモリサイズの見積もり方法

コマンド連絡用共用メモリサイズの見積もり式を次に示します。コマンド連絡用共用メモリは、抽出マスタプロセスが稼働するサーバマシン上で使います。

$$\text{コマンド連絡用共用メモリサイズ} = 1360 \text{ (単位: バイト)}$$

(j) ステータス情報連絡用共用メモリサイズの見積もり方法

ステータス情報連絡用共用メモリサイズの見積もり式を次に示します。ステータス情報連絡用共用メモリは、抽出マスタプロセスが稼働するサーバマシン上で使います。

$$1360 + 36 + 64 + ND_NUM + 60 + SV_NUM + 164 \times SV_NUM \times SND_NUM \\ + 8 \times \Sigma_{SND} (NSND_NUM) \times SV_NUM \\ + (ND_NUM + SV_NUM + SV_NUM \times SND_NUM) \times 4 \text{ (単位: バイト)}$$

Σ_{SND} : 全送信先の送付

ND_NUM : 全ノード数

SV_NUM : 全サーバ数

SND_NUM : 全送信先数

NSND_NUM : 送信先当たりの送信抑止受信元識別子数

(k) プロセス間連絡用共用メモリサイズの見積もり方法

プロセス間連絡用共用メモリサイズの見積もり式を次に示します。プロセス間連絡用共用メモリは、抽出ノードマスタプロセスが稼働するサーバマシン上で使います。

$$1048 + (2969 + 4 \times SND_NUM + 16 \times UINF_NUM) \times NSV_NUM \\ + 1600 \times NSV_NUM + SND_NUM \text{ (単位: バイト)}$$

SND_NUM : 全送信先数

NSV_NUM : 該当ノード内サーバ数

UINF_NUM : 抽出システム定義の extinforum オペランドの設定値

(l) 抽出定義格納用共用メモリサイズの見積もり方法

抽出定義格納用共用メモリサイズの見積もり式を次に示します。抽出定義格納用共用メモリは、抽出ノードマスタプロセスが稼働するサーバマシン上で使います。

$$164 + (40 + SV_INFO) \times SV_NUM \text{ (単位: バイト)}$$

- SV_INFO

$$= 48 + (52 + 96 \times SCH_NUM + 64 \times UPD_NUM + 128 \times COL_NUM + 64 + 112 \times \\ TYPE_NUM + 64 \times ATTR_NUM) + 24 + 32 \times TBL_NUM + 24 \times COL_NUM + SND_INFO + \\ LOB_INFO + DEFAULT_INFO$$

- SND_INFO

$$= 16 + 28 \times \text{UPD_NUM} + 40 \times \text{SND_NUM} + \sum_{\text{SND}} (40 \times \text{SND_UPD} + 20 \times \text{SND_TBL} + 24 \times \text{SND_COL})$$

- **DEFAULT_INFO**

$$= 16 + 20 \times \text{SV_DEF} + \text{SUM_DEF}$$

- **LOB_INFO**

$$= 16 + 4 \times \text{LOB_AREA}$$

- \sum_{SND} ：全送信先数の総和
- SV_NUM：該当ノード内のバックエンドサーバ数
- SV_INFO：バックエンドサーバ単位定義情報サイズ
- SCH_NUM：すべての抽出対象スキーマ数
- UPD_NUM：すべての更新情報名数
- TBL_NUM：すべての抽出対象表数
- COL_NUM：すべての抽出対象列数
- TYPE_NUM：抽出対象になる抽象データ型の数。重複分は除きます。抽出対象の抽象データ型が継承している上位データ型と抽象データ型の属性データタイプとして定義されている抽象データ型の数を含みます。
- ATTR_NUM：TYPE_NUM に計上した抽象データ型の属性数の総和
- SND_INFO：送信先情報サイズ
- SND_NUM：送信先数
- SND_UPD：送信先ごとの送信対象更新情報名数
- SND_TBL：送信先ごとの送信対象表数
- SND_COL：送信先ごとの送信対象列数
- LOB_INFO：BLOB 抽象情報サイズ
- LOB_AREA：抽出対象 BLOB 列格納 RD エリア数
- SV_DEF：列追加時にデフォルト値を指定したサーバ内の列数
- SUM_DEF：列追加時にデフォルト値を指定したサーバ内の列の DEF_LEN の総和
- DEF_LEN：((デフォルト値の長さ ÷ 4) + 1) × 4

(m) メッセージ格納用共用メモリサイズの見積もり方法

メッセージ格納用共用メモリサイズの見積もり式を次に示します。メッセージ格納用共用メモリは、抽出ノードマスタプロセスが稼働するサーバマシン上で使います。

メッセージ格納用共用メモリサイズ = 79200 (単位：バイト)

(n) トランザクション管理情報格納用共用メモリサイズの見積もり方法

トランザクション管理情報格納用共用メモリサイズの見積もり式を次に示します。トランザクション管理情報格納用共用メモリは、抽出ノードマスタプロセスが稼働するサーバマシン上で使います。この共用メモリは、抽出システム定義の sendcontrol オペランドの指定が sendmst の場合にだけ必要になります。

$$1048576 \times \text{NSV_NUM} \text{ (単位: バイト)}$$

NSV_NUM: 該当ノード内サーバ数

(o) セマフォ所要数の見積もり方法

抽出側 Datareplicator が使うセマフォ数の見積もり式を次に示します。セマフォは、抽出ノードマスタプロセスが稼働するサーバマシン上で使います。

- 抽出システム定義に sendcontrol=nodemst を指定している場合

$$1 + (2 + 2 \times \text{SND_NUM}) \times \text{NSV_NUM}$$

- 抽出システム定義に sendcontrol=sendmst を指定している場合

$$1 + 4 \times \text{NSV_NUM}$$

SND_NUM: 全送信先数

NSV_NUM: 該当ノード内サーバ数

上記の見積もり式で見積もった数のセマフォを抽出側システムで使えるように、必要に応じてカーネルパラメタのシステム全体のセマフォ最大数 (SEMMNS) を変更してください。カーネルパラメタの更新方法については、該当する OS のマニュアルを参照してください。

(p) メッセージキュー所要数の見積もり方法

抽出マスタプロセスが稼働するサーバマシンに 1 個と、抽出ノードマスタプロセスが稼働するサーバマシンごとに各 1 個ずつ必要です。

(q) 更新情報編集バッファサイズの見積もり方法

更新情報編集バッファサイズの見積もり式を次に示します。抽出対象表ごとに更新情報編集バッファサイズを見積もり、最大値を「5.2 抽出システム定義」の smt_editbufsize オペランド、又は「5.4 送信環境定義」の editbufsize オペランドに指定してください。

- 抽出定義に key 句を指定している場合

$$244 + 16 \times \text{列数} + \text{データ長}^{\ast} \text{ (単位: バイト)}$$

- 抽出定義に ukey 句を指定している場合

$$244 + 16 \times \text{列数} + \text{データ長}^{\ast} \\ + (32 + 16 \times \text{マッピングキーの列数} + \text{マッピングキーのデータ長の合計}) \\ (\text{単位: バイト})$$

注※

連動対象の列の全データ長を合計した値です (単位: バイト)。

4.7 反映側 Datareplicator の設計

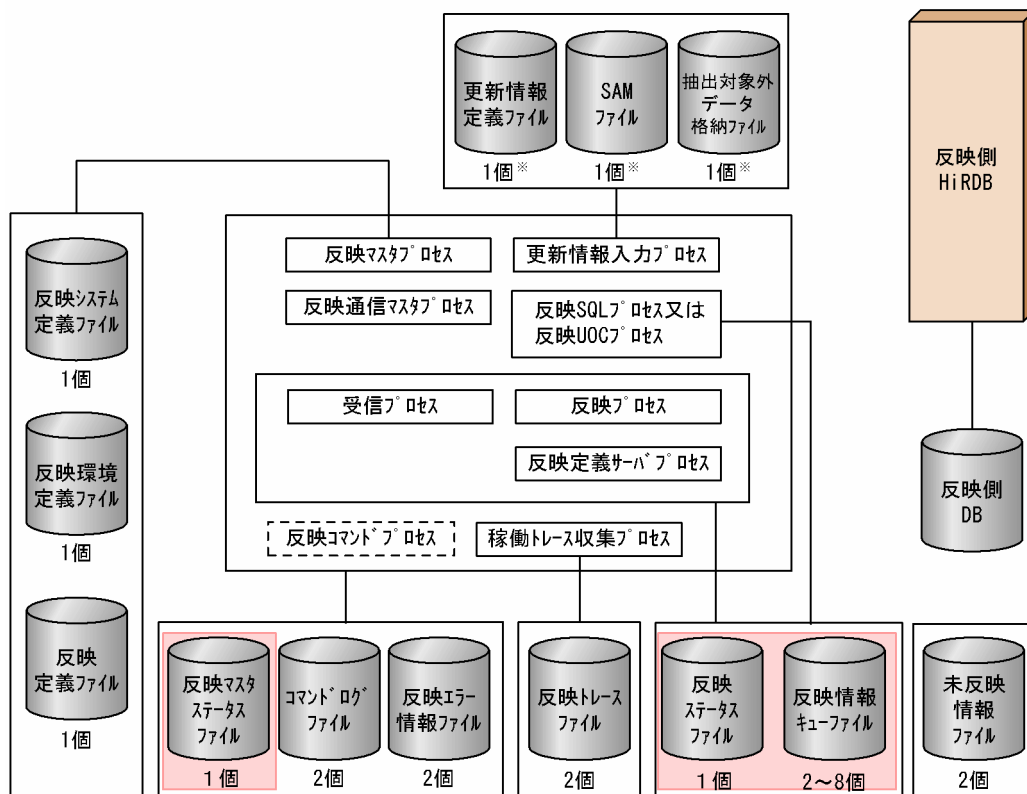
反映側 Datareplicator のシステム設計について説明します。

4.7.1 反映側 Datareplicator のファイル構成

反映側 Datareplicator のファイル構成を次の図に示します。

反映側 Datareplicator のディレクトリ構成については、UNIX 版の場合は「2.3.3 反映側 Datareplicator のディレクトリ構成」を、Windows 版の場合は「2.7.3 反映側 Datareplicator のディレクトリ構成」を参照してください。

図 4-40 反映側 Datareplicator のファイル構成



(凡例)

- : 常駐プロセス
- : 非常駐プロセス
- : 二重化定義可能ファイル

注※

SAM ファイルを使う抽出側システムからデータを受け取るときに必要です。ただし、RDB1 E2 の場合は更新情報定義ファイルは必要ありません。

4.7.2 反映側 Datareplicator で使うファイルの準備

反映側 Datareplicator を動作させるために必要な、次に示すファイルの準備方法について説明します。

ユーザが作成するファイル	反映側 Datareplicator の初期起動時に作成されるファイル
<ul style="list-style-type: none"> 反映システム定義ファイル 反映環境定義ファイル 反映定義ファイル 二重化定義ファイル 	<ul style="list-style-type: none"> 反映情報キューファイル 反映ステータスファイル 反映マスタステータスファイル 反映エラー情報ファイル 稼働トレースファイル 反映トレースファイル 未反映情報ファイル コマンドログファイル 二重化制御ファイル 回復情報ファイル

SAM ファイルを使う抽出側 DB (PDMII E2, RDB1 E2) の場合は、上記のファイルに加えて、次に示すファイルを準備する必要があります。

- 更新情報定義ファイル
- SAM ファイル
- 抽出対象外データ格納ファイル

ファイルの内容については、「[3.3.2 反映処理で使うファイルとプロセス](#)」を参照してください。

ファイルの運用については、「[6.7.2 反映側 Datareplicator で使うファイルの運用](#)」を参照してください。

反映側 Datareplicator で使うファイルの作成について次の表に示します。

表 4-40 反映側 Datareplicator で使うファイルの作成

ファイル名		ファイル種別 ^{※8}		個数	必要・任意
		R ^{※9}	C ^{※10}		
定義ファイル	反映システム定義ファイル ^{※1}	○	×	反映側システムごとに 1 個	必要
	反映環境定義ファイル ^{※1}	○	×	データ連動識別子ごとに 1 個	必要
	反映定義ファイル ^{※1}	○	×	データ連動識別子ごとに 1 個	任意 ^{※11}
	更新情報定義ファイル ^{※2}	○	×	更新情報入力コマンド実行ごとに 1 個	任意 ^{※12}
	二重化定義ファイル ^{※1}	○	×	反映側システムごとに 1 個 (抽出側 HiRDB がパラレルサーバの場合には、MGR に 1 個)	任意
反映情報キューファイル ^{※3※13}		○	○	データ連動識別子ごとに 2~8 個	必要

ファイル名		ファイル種別※8		個数	必要・任意
		R※9	C※10		
ステータスファイル	反映ステータスファイル※3※13	○	○	データ連動識別子ごとに1個	必要
	反映マスタステータスファイル※3※13	○	○	反映側システムごとに1個	必要
未反映情報ファイル※4※5		○	×	データ連動識別子ごとに2個	必要
反映エラー情報ファイル※4※5		○	×	反映側システムごとに2個	必要
稼働トレースファイル（反映トレースファイル）※14		○	×	反映側システムごとに2個	任意
コマンドログファイル※5		○	×	反映側システムごとに2個	任意
SAM ファイル※6		○	×	更新情報入力コマンド実行ごとに1個	任意※12
抽出対象外データ格納ファイル※7		○	×	データ連動識別子ごとに1個	任意※12
二重化制御ファイル※4		○	×	反映側システムごとに1個	任意
回復情報ファイル		○	○	データ連動識別子ごとに1個	任意

(凡例)

MGR：システムマネージャ

R：UNIX の通常ファイル，又は Windows ファイル

C：UNIX のキャラクタ型スペシャルファイル

○：作成できます。

×：作成できません。

注※1

反映側 Datareplicator を起動する前に，OS のエディタで作成してください。

注※2

更新情報入力処理（hdssamqin コマンドの実行）と反映処理を開始する前に，OS のエディタで作成してください。

注※3

反映側 Datareplicator を起動する前に hdsstart - i コマンドを実行して反映側 Datareplicator を初期化してください。hdsstart - i コマンドを実行すると，ファイルが自動的に作成されます。ただし，UNIX のキャラクタ型スペシャルファイルの場合は，反映側 Datareplicator を起動する前に，キャラクタ型スペシャルファイルに対するシンボリックリンクを作成してください。

注※4

抽出側 Datareplicator の初期化時に自動的に作成されます。

注※5

抽出側 Datareplicator の起動時に自動的に作成されます。

注※6

SAM ファイルを使うメインフレーム側 DB のシステムで作成した SAM ファイルを、メインフレーム側のファイル転送プログラムを使って、反映側 Datareplicator へ転送します。

注※7

hdssamqin コマンドを実行すると、作成又は再作成されます。ファイルが既に存在する場合は再作成されて、サイズは 0 バイトになります。

注※8

次のファイルはすべて同じファイル種別にしてください。

- 反映情報キューファイル
- 反映ステータスファイル

注※9

ユーザが作成するファイル（定義ファイル）の場合には、反映側 Datareplicator の使用者に read 権限を与えてください。write 権限は任意で与えてください。ユーザが作成するファイル（定義ファイル）以外の場合は、権限を変更しないでください。

注※10

キャラクタ型スペシャルファイルは OS のコマンドで作成します。キャラクタ型スペシャルファイルで作成する場合は、反映側 Datareplicator の使用者に read 権限と write 権限を与えてください。また、系切り替え機能を使うときは、キャラクタ型スペシャルファイル形式で作成してください。

なお、反映情報キューファイルと反映ステータスファイルをキャラクタ型スペシャルファイルで作成した場合は、反映側 Datareplicator を hdsstart -i -f コマンドで初期開始してください。

反映側 Datareplicator の稼働中にファイルの容量不足が発生しないよう、環境構築時、hdsstart -i コマンドに init を指定して実行し、ファイルのサイズに不足がないかテストすることをお勧めします。

注※11

抽出対象の表と反映対象の表の形式、表名、列名などすべてが同一の場合は省略できます。

注※12

SAM ファイルを使ってデータ連動する場合に、更新情報入力コマンド（hdssamqin コマンド）を実行するときに使うファイルです。ただし、RDB1 E2 の場合は、更新情報定義ファイルは必要ありません。

注※13

UNIX 版の場合、ファイル種別を OS の通常ファイルとしていると、システムの障害時に、データが出力されないことがあります。反映情報キューファイル、反映ステータスファイル、及び反映マスタステータスファイルは、障害回復時に反映側 Datareplicator が使うため、このファイルにデータが出力されていない場合には障害を回復できなくなります。このため、これらのファイルは、信頼性の高いキャラクタ型スペシャルファイルに作成することをお勧めします。

注※14

稼働トレース収集方式を全プロセス一括収集方式とする場合のファイル名です。プロセス単位収集方式とする場合の稼働トレースファイル名については、「5.8 反映システム定義」の int_trc_unit オペランドの説明を参照してください。

反映側 Datareplicator で使うファイルの設定について次の表に示します。

表 4-41 反映側 Datareplicator で使うファイルの設定

ファイルの種類	ファイル名	備考
反映システム定義ファイル	\$HDSPATH/hdsenv	<ul style="list-style-type: none"> 定義する内容については、「5.8 反映システム定義」を参照してください。
反映環境定義ファイル	\$HDSPATH/任意の名称	<ul style="list-style-type: none"> 定義する内容については、「5.9 反映環境定義」を参照してください。 任意の名称を反映システム定義の refenv001～refenv128 オペランドに指定してください。
反映定義ファイル	\$HDSPATH/任意の名称	<ul style="list-style-type: none"> 定義する内容については、「5.10 反映定義」を参照してください。 任意の名称を反映環境定義の reffile オペランドに指定してください。
更新情報定義ファイル	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 指定する内容については、「5.11 更新情報定義」を参照してください。 任意の名称を hdssamqin -n コマンド実行時に指定してください。
二重化定義ファイル	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 定義する内容については、「5.12 二重化定義 (反映側)」を参照してください。 任意の名称を反映システム定義ファイルの file_dupenv オペランドに指定してください。
反映情報キューファイル※1	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 任意の名称を反映環境定義の qufile001～qufile008 オペランドに指定してください。 反映環境定義の queuesize オペランドでファイルのサイズを指定してください。
反映ステータスファイル※1	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 任意の名称を反映環境定義の statsfile オペランドに指定してください。 反映環境定義の statssize オペランドでファイルのサイズを指定してください。
反映マスタステータスファイル※1	\$HDSPATH/hdsinitstate	—
未反映情報ファイル	任意のディレクトリ/任意の名称	<ul style="list-style-type: none"> 任意の名称を反映環境定義の unreffile1 と unreffile2 オペランドに指定してください。 反映環境定義の unreffilez オペランドでファイルサイズを指定してください。
反映エラー情報ファイル	\$HDSPATH/errfile1 \$HDSPATH/errfile2	<ul style="list-style-type: none"> 反映システム定義の errfilez オペランドでファイルサイズを指定してください。
稼働トレースファイル (反映トレースファイル) ※2	\$HDSPATH/reftrc.trc1 \$HDSPATH/reftrc.trc2	<ul style="list-style-type: none"> 反映システム定義の int_trc_filesz オペランドでファイルサイズを指定します。 hdstrcredit コマンドで編集, 又は参照ができます。

ファイルの種類	ファイル名	備考
コマンドログファイル	任意のディレクトリ/任意の名称	—
SAM ファイル	任意のディレクトリ/任意の名称	• 任意の名称を <code>hdssamqin -n</code> コマンド実行時に指定してください。
抽出対象外データ格納ファイル	任意のディレクトリ/ unextfile_データ連動識別子	—
二重化制御ファイル	<code>\$HDSPATH/ hds_fileenv.prp</code>	—
回復情報ファイル	表 9-8 参照	—

(凡例)

— : 特にありません。

注※1

ファイルを二重化できます。ただし、ファイルシステム領域を使用する場合は、二重化できません。

注※2

稼働トレース収集方式を全プロセス一括収集方式とする場合のファイル名です。プロセス単位収集方式とする場合の稼働トレースファイル名については、「5.8 反映システム定義」の `int_trc_unit` オペランドの説明を参照してください。

4.7.3 反映方法の設計

データの反映方法として、次の項目を設計する必要があります。

- 反映処理の方式
- マルチ FES 機能を使う場合の反映処理の方式
- 反映処理の DISCONNECT 発行間隔
- イベント機能による反映処理の自動制御
- 反映処理の COMMIT 発行間隔
- 反映定義で定義されていない更新情報に対する処理
- 反映対象表の存在チェック
- 定義情報格納用共用メモリサイズ

(1) 反映処理の方式の設計

反映処理の方式には、トランザクション単位反映方式と表単位反映方式の二つがあります。トランザクション単位反映方式と表単位反映方式の詳細については、「3.3.3 反映処理の方式」を参照してください。

反映処理の方式は、反映環境定義の startmode 又は breakmode で指定します。

(a) トランザクション単位反映方式

抽出側 DB で更新されたトランザクション順に反映側 HiRDB のデータベースに反映します。

トランザクション単位反映方式では、同一形式の表（表名、列名、属性がすべて同じ表）に反映する場合は、反映定義を省略できます。

(b) 表単位反映方式

一つ又は複数の反映対象表ごとに反映グループを作り、各グループに並列に反映します。反映グループは、一つの反映処理に最大 128 個定義できます。反映グループは反映グループ定義で指定します。反映グループ定義については、「[5.10.6 反映グループ定義](#)」を参照してください。

表単位反映方式には、次に示す 3 種類があります。

- 表単位分割方式
- キーレンジ単位分割方式
- ハッシュ分割方式

表単位分割方式

表単位分割方式では抽出側 DB で更新されたトランザクションを、ユーザの定義した反映グループごとに分けて、並列に反映します。

表単位分割方式で反映処理を実行する場合に、表同士で参照制約があるときは、参照制約がある表を同一グループとしてください。

キーレンジ単位分割方式

キーレンジ単位分割方式では抽出側 DB で更新されたトランザクションを、ユーザが定義したキーレンジに従って、並列に反映します。キーレンジ分割方式で反映処理を実行する場合には、次の点に注意してください。

- 一つの反映グループに定義できるキーレンジは、最大 8 個です。
- キーレンジ分割条件文に指定できる列名は、抽出対象表のマッピングキーになる列に対応する、反映対象表の列名だけです。
- 一つのキーレンジ分割条件では、条件文を最大 8 個指定できます。一つのキーレンジ分割条件に複数の条件文を指定すると、すべての条件文が AND 結合され、複雑なキーレンジ分割条件も指定できます。ただし、複雑なキーレンジ分割条件を指定すると、条件文の判定のために性能が低下する場合があります。分割よりも性能を重視する場合には、一つのキーレンジ分割条件に指定する条件文を少なくしてください。
- 一つのキーレンジに対して反映処理が集中する場合には、キーレンジ分割しないときより、性能が低下する場合があります。これはキーレンジ分割しないときと同様に処理が分散されていなく、さらにキーレンジ分割の判定処理を実行するためです。この場合には、反映側 Datareplicator を正常

終了、又は即時終了し、反映処理が集中したキーレンジを、さらに分割した後、反映側 Datareplicator を再起動してください。

- CPU 性能が低い状態で多数の反映プロセス、SQL プロセスを起動させた場合には、マシンに掛かる負荷によって反映側 Datareplicator が異常終了することがあります。これを防ぐためには、反映処理の少ないキーレンジを other 条件などでまとめ、1 回で起動するプロセス数を少なくしてください。

ハッシュ分割方式

ハッシュ分割方式では抽出側 DB で更新されたトランザクションを、ハッシュ法で並列に反映します。

(2) マルチ FES 機能を使う場合の反映処理の方式の設計

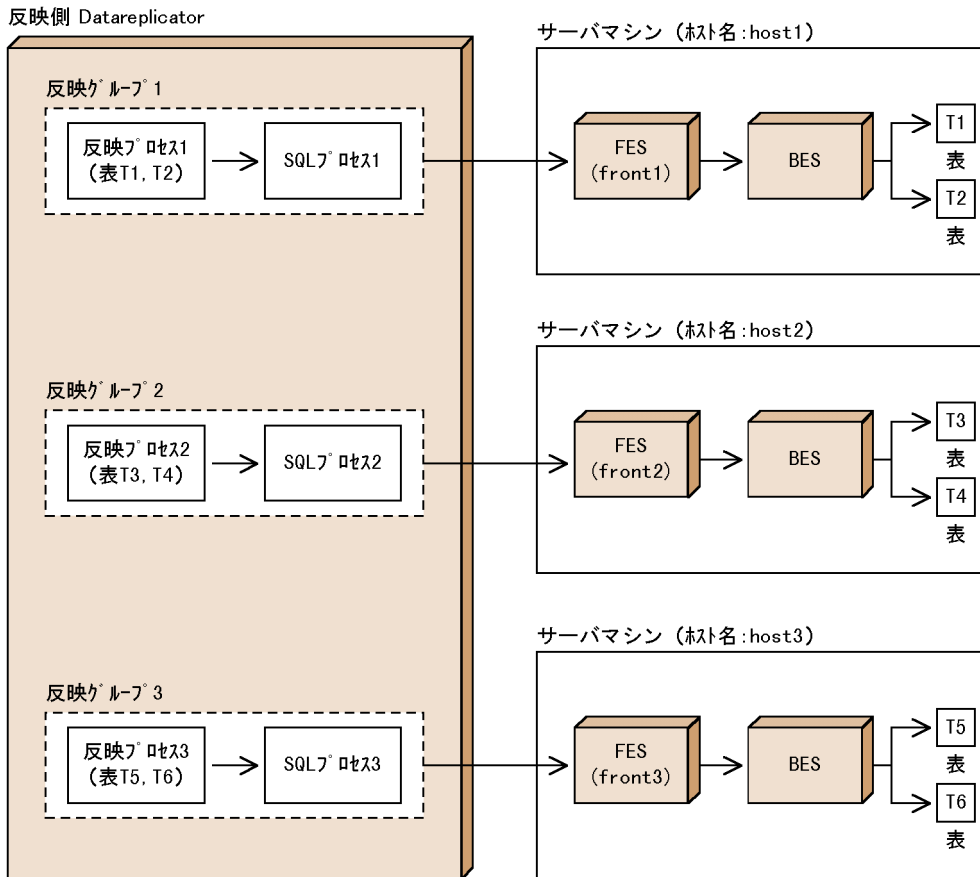
反映側 HiRDB がマルチ FES 機能を使っている場合、反映側 Datareplicator でもマルチ FES 機能に対応した反映処理を実行できます。マルチ FES 機能を使うときは、反映定義で反映グループに対応する反映先のフロントエンドサーバを指定する必要があります。反映側 Datareplicator では、反映定義に従って反映先のフロントエンドサーバごとに SQL プロセスを対応付けます。このため、反映先のフロントエンドサーバごとに並行して SQL 文を発行でき、フロントエンドサーバの負荷を分散できます。

表単位反映方式の分割方式ごとにマルチ FES 機能を使う例と、マルチ FES 機能を使うときの考慮点を次に示します。

(a) 表単位分割方式の場合

表単位分割方式の場合には、反映側 HiRDB でサーバ単位に表をグループ化して格納していると、反映先のフロントエンドサーバごとに並行して SQL 文を発行できるため、最もスループットの向上が期待できます。表単位分割方式でマルチ FES 機能に対応した例を次の図に示します。

図 4-41 表単位分割方式でマルチ FES 機能を使う例



(凡例)

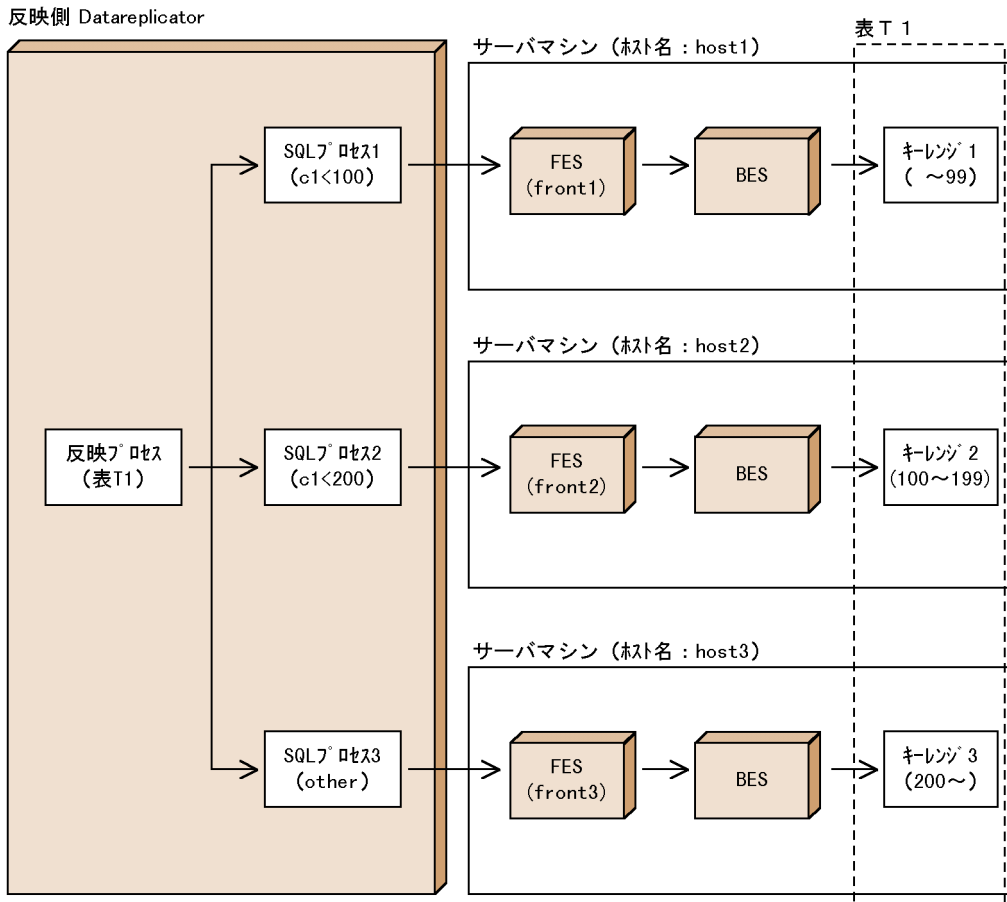
F E S : HiRDBのフロントエンドサーバ B E S : HiRDBのバックエンドサーバ

(b) キーレンジ単位分割方式の場合

キーレンジ単位分割方式では、反映側 HiRDB で横分割したキーレンジ単位に、別のサーバに表を分割して格納しているときに、反映先のフロントエンドサーバごとに並行して SQL 文を発行できるため、スループットの向上が期待できます。

キーレンジ単位分割方式でマルチ FES 機能を使う例を次の図に示します。

図 4-42 キーレンジ単位分割方式でマルチ FES 機能を使う例



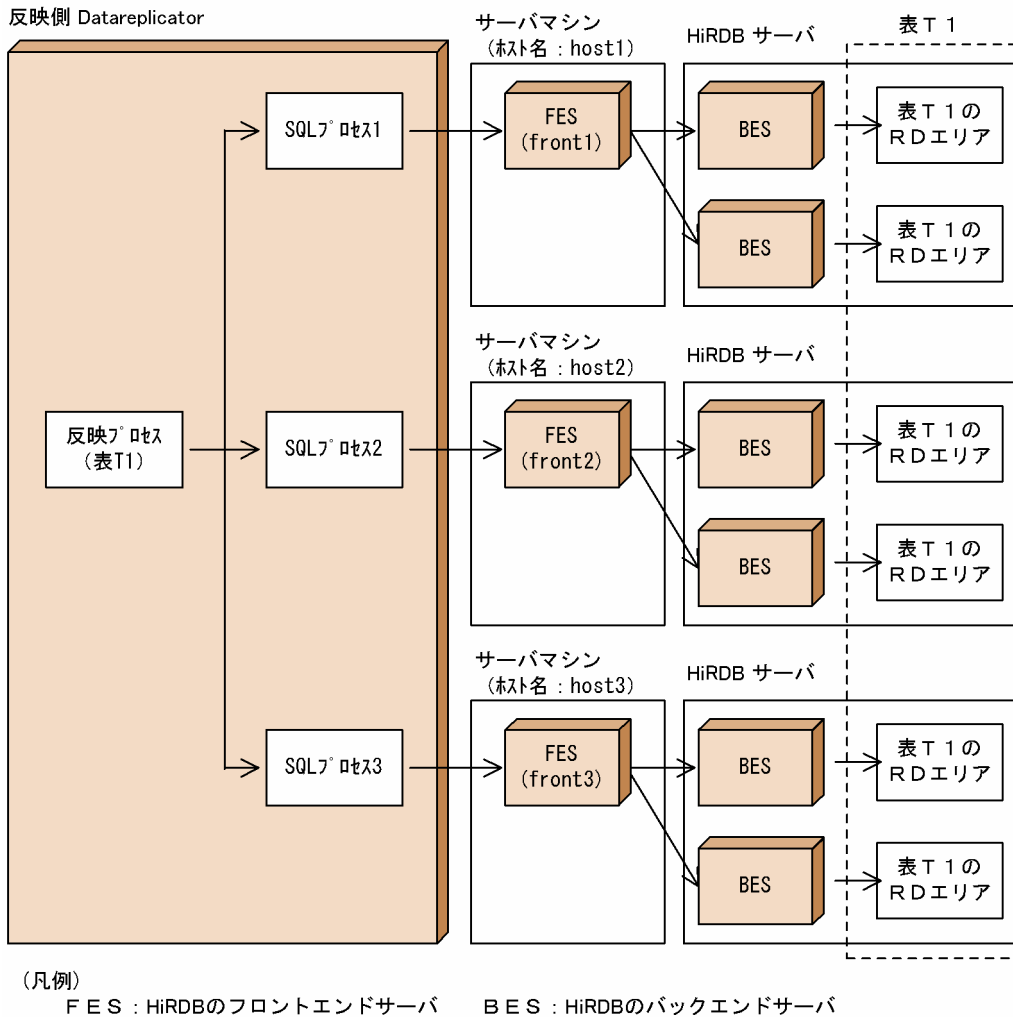
(凡例)
 FES : HiRDBのフロントエンドサーバ BES : HiRDBのバックエンドサーバ

(c) ハッシュ分割方式の場合

ハッシュ分割方式では、反映側 HiRDB で表を分割して格納している場合に、ハッシュ法に従ってフロントエンドサーバごとに並行して SQL 文を実行します。キーレンジ単位分割方式に比べて、マルチ FES 機能でのフロントエンドサーバごとに処理を分割できるので、スループットの向上が期待できます。

ハッシュ分割方式でマルチ FES 機能を使う例を次の図に示します。

図 4-43 ハッシュ分割方式でマルチ FES 機能を使う例



マルチ FES 構成にしているシステムの場合、ハッシュ分割方式では、分割されるエリア別に、反映するデータをどのフロントエンドサーバで処理するかをあらかじめ設定できます。これによってフロントエンドサーバの負荷を分散でき、またフロントエンドサーバと RD エリアの属するバックエンドサーバが同じサーバ上にあるときには、フロントエンドサーバとバックエンドサーバの間の通信オーバーヘッドを削減できます。

(d) 考慮点

反映側 HiRDB のマルチ FES 環境に対して、反映定義で指定した反映先のフロントエンドサーバと実際に反映処理を実行するバックエンドサーバの起動するマシンが異なる場合には、1 回の反映処理ごとにフロントエンドサーバとバックエンドサーバとの間の通信が発生して、効率的な処理が実行できません。このため、反映定義でフロントエンドサーバを指定する場合には、表データを格納しているマシンで起動しているフロントエンドサーバを指定してください。

(3) 反映処理の DISCONNECT 発行間隔の設計

反映処理が反映情報キューファイルに蓄積されている更新情報の終端を検知してから、反映側 HiRDB に対して DISCONNECT を発行するまでの間隔を設計します。反映処理の DISCONNECT 発行間隔は、反映システム定義の `discintvl` オペランドで指定します。

DISCONNECT 発行間隔の考慮点を次に示します。

- DISCONNECT を発行しない場合には、発行間隔に 0 を指定します。
- 抽出側 DB が HiRDB の場合には、抽出側 Datareplicator 定義の送信環境定義の `sendintvl` オペランドの指定値と、抽出側システムのトランザクション発生頻度を考慮して、`discintvl` オペランドを指定してください。抽出側 DB がメインフレーム側 DB の場合には、XDM/DS 起動定義の `RINTERVAL` 句の指定値と抽出側システムのトランザクション発生頻度を考慮して、`discintvl` オペランドを指定してください。
- 抽出側システムの業務終了まで短い間隔かつ高頻度でトランザクションが発生する場合は、0 に近い値を指定すると DISCONNECT が発行される可能性が上がります。トランザクション発生頻度にばらつきがある場合、`sendintvl` オペランド、又は `RINTERVAL` 句を大きく設定している場合はこのオペランドを小さく指定し、小さく設定している場合は、`sendintvl` オペランド、又は `RINTERVAL` 句より大きく指定すると DISCONNECT 発行の最適化が図れます。

(4) イベント機能による反映処理の自動制御の設計

イベント機能を使って、抽出側システムで発行させたイベントに従って、反映処理の動作を切り替えることができます。イベント機能を使う場合、反映側 Datareplicator では、抽出側システムで発行するイベントに対応させるイベントコードを反映環境定義で指定します。ここでは、切り替えができる反映処理の動作と、考慮点について説明します。

(a) イベント機能によって切り替えができる反映処理の動作

イベント機能によって切り替えができる反映処理の動作と反映環境定義との関係を次の表に示します。

表 4-42 切り替えができる反映処理の動作と反映環境定義との関係

イベント種別	反映処理の動作	指定するオペランド
反映処理停止イベント	反映処理を停止します。	<code>eventspd</code>
トランザクション単位反映イベント	反映処理の動作中に、反映方式をトランザクション単位反映方式に切り替えます。	<code>eventtrn</code>
表単位反映イベント	反映処理の動作中に、反映方式を表単位反映方式に切り替えます。	<code>eventtbl</code>
トランザクション単位反映再起動イベント	反映処理の停止中に、反映処理をトランザクション単位反映方式で再起動します。	<code>eventretrn</code>
表単位反映再起動イベント	反映処理の停止中に、反映処理を表単位反映方式で再起動します。	<code>eventretbl</code>

イベント種別	反映処理の動作	指定するオペランド
反映処理数リセットイベント	反映側 Datareplicator の反映処理数をリセットします。	eventcntreset

(b) イベント機能を使う場合の考慮点

- 抽出側システムで発行するイベントコードに対応するイベントコードを、反映環境定義の該当するオペランドに指定してください。
- 反映環境定義で指定されていないイベントコードに対応するイベントを抽出側システムで発行すると、反映側システムの syslog ファイルにメッセージが出力されます。これを利用して、抽出側システムの何らかの動作を、反映側システムに連絡できます。例えば、抽出側システムが終了したときに、反映側システムでは指定されていないイベントを抽出側システムから発行し、抽出側システムが終了したことを反映側システムに連絡することもできます。

(5) 反映処理の COMMIT 発行間隔の設計

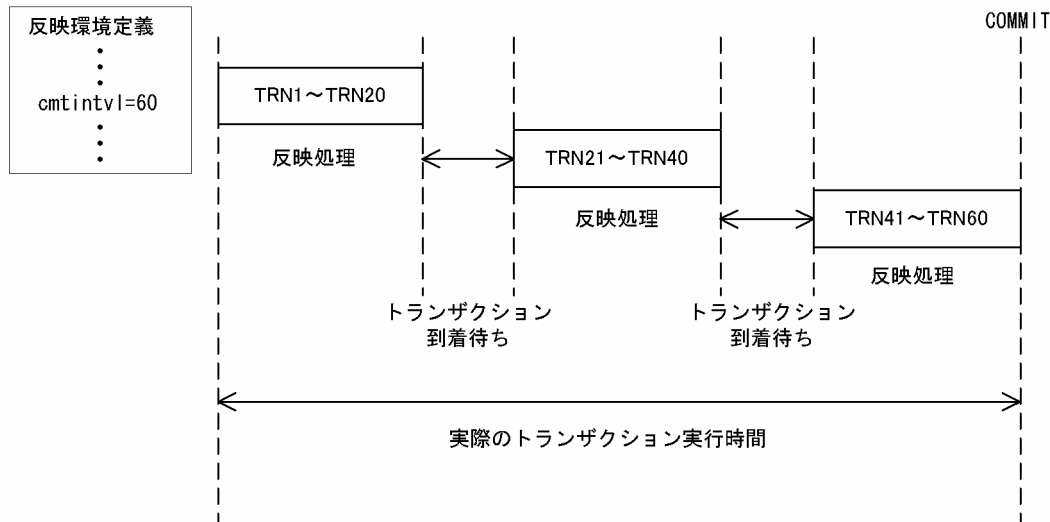
コミットを発行する単位は、抽出側システムのトランザクション数で指定します。反映側 HiRDB にコミットを発行する間隔は、反映環境定義の `cmtintvl`, `trncmtintvl`, 又は `tblcmtintvl` オペランドで指定します。

コミットの発行間隔の設定時の考慮点を次に示します。

- コミットを発行する間隔が小さいよりも大きい方が、より多くのトランザクションに対する SQL 文を 1 回で発行するため、反映側 HiRDB への負荷が減り、性能面での向上が期待できます。ただし、障害が発生したときには、逆に反映できないトランザクションが多くなります。反映できなかったトランザクションは、次回の反映処理の起動時に自動的に反映されますが、コミットを発行する間隔が大きい場合には、回復するのにより長い時間が掛かります。
- 1 コミット間隔の反映処理で出力される反映側 HiRDB のログ量を考慮し、システムログファイルなど、障害回復用のファイルが満杯にならないようにしてください。
- 反映側 HiRDB では SQL 文として PURGE TABLE を発行すると、SQL 文を発行した時点でコミットを発行します。また、イベント検知時には反映側 Datareplicator が同期点を取得するため、コミットを発行します。このため、PURGE TABLE を含む更新情報を反映する場合やイベント検知時には、反映環境定義で定義したコミット間隔と異なることがあります。
- 反映方式ごとに COMMIT 発行間隔を指定できます。トランザクション単位反映方式の COMMIT 発行間隔は `trncmtintvl` オペランドで、表単位反映方式の COMMIT 発行間隔は `tblcmtintvl` オペランドで指定します。両方の反映方式に共通の COMMIT 発行間隔は `cmtintvl` オペランドで指定します。
- `cmtintvl`, `trncmtintvl`, 及び `tblcmtintvl` オペランドの指定値を大きくすると、HiRDB サーバに対して次の影響があります。反映側 Datareplicator から発生するトランザクションの大きさは、反映先 HiRDB の許容範囲を超えないようにしてください。
 - 排他リソース量（行排他）が増加し、DB の排他リソース不足となる。
 - シンクポイントダンプの有効化処理のスキップ回数の上限を超え、反映処理が ROLLBACK される。
 - `commit_wait_time` オペランドの指定値を超えた時点でコミットが発行され、反映処理が終了となる。

注意

反映側 Datareplicator から発生するトランザクションの実行時間は、実際に SQL を実行する時間と、次に反映するトランザクションの到着待ち時間を合計したものです。実際のトランザクション実行時間を次の図に示します。



次に反映するトランザクションの到着待ち時間の限界は、`commit_wait_time` オペランドで指定します。`commit_wait_time` オペランドの指定値を超えても次のトランザクションが到着しない場合は、反映側 Datareplicator が自動的に COMMIT を発行し、トランザクションを決着させます。

(6) 反映定義で定義されていない更新情報に対する処理の設計

反映定義で定義されていない更新情報について、その更新情報が定義された抽出側システムの抽出定義を反映定義と仮定して、反映処理を実行するかどうかを設計します。反映定義で定義されていない更新情報に対する処理は、反映環境定義の `defmerge` オペランドで指定します。

反映定義で定義されていない更新情報に対する処理の考慮点を次に示します。

- 反映定義を省略している場合は、無条件に抽出定義を反映定義と仮定します。

(7) 反映対象表の存在チェックの設計

反映側 Datareplicator の起動時に、反映対象表が反映側 HiRDB に存在するかどうかをチェックするかどうかを設計します。

反映対象表の存在チェックは、反映環境定義の `tblcheck` オペランドで指定します。

- 反映定義を省略している場合には、抽出側システムの抽出定義で定義されている抽出対象表が、反映側システムに存在するかどうかをチェックします。
- 反映定義がある場合は、存在チェックの指定に関係なく、反映定義で指定した反映対象表が、反映側システムに存在するかどうかをチェックします。

(8) 定義情報格納用共用メモリサイズ的设计

反映側 Datareplicator を起動すると、定義情報格納用共用メモリは共用メモリ上に保持されます。定義情報格納用共用メモリサイズは、反映環境定義の defshmsize オペランドで指定します。

定義情報格納用共用メモリサイズの見積もり時の考慮点を次に示します。

- 定義情報格納用共用メモリには、反映情報キューファイルに格納される抽出側システムの抽出定義情報が保持されます。このため、定義情報格納用共用メモリサイズは、抽出側システムの抽出定義情報のサイズよりも大きくしてください。抽出定義情報よりも小さい値を指定した場合、反映側 Datareplicator の起動時にエラーになります。反映情報キューファイルに格納される抽出定義情報については、「[4.7.7 反映側 Datareplicator のリソース的设计](#)」を参照してください。

(9) 反映側 HiRDB のリソース的设计

(a) 排他資源数

次の二つのうち大きい方の排他資源数を、反映先 HiRDB のリソースとして見積もってください。また、抽出側 HiRDB で実行する pdload をデータ連携する場合、反映側ではすべて INSERT 文に置き換えて実行します。よって、INSERT ごとに排他資源が必要になります。そのため、HiRDB の排他資源を多量に消費し、反映側 Datareplicator の反映処理が排他資源不足で、SQL エラーとなる場合があります。pdload をデータ連携する場合は、反映側 DB の排他資源数を pdload による挿入件数以上とするように見積もってください。

- load 文で定義したすべての反映表に対して、SQL 文"select * from 反映表"を PREPARE したときの排他資源数
- 反映処理の COMMIT 間に発行する SQL 文 (INSERT 文, UPDATE 文, DELETE 文) を実行できるだけの排他資源数

排他資源数の見積もりの詳細については、マニュアル「HiRDB システム定義」を参照してください。

(b) HiRDB に同時に接続するデータ連動識別子の数

HiRDB に同時に接続するデータ連動識別子の数を、HiRDB の定義の pd_max_users オペランドの指定値に加算してください。HiRDB に同時に接続するデータ連動識別子の数は、反映システム定義に定義しているデータ連動識別子ごとに、次の数になります。

トランザクション単位反映モードの場合

1 個

表単位反映モードの場合

反映定義に定義している group 文で指定した分割数[※]の合計数

注※

分割数の詳細を次に示します。

- 表単位分割方式：1 個
- キーレンジ単位分割方式：キーレンジ分割数
- ハッシュ分割方式：ハッシュ分割数，又は SQL 分割数

(例)

group G1 by T1 : 1 個

group G2 by T2 : 1 個

group G3 by T3 : 5 個

hash divide into 5

上記の場合，HiRDB に同時に接続するデータ連動識別子の数は 7 個 (1+1+5) になります。

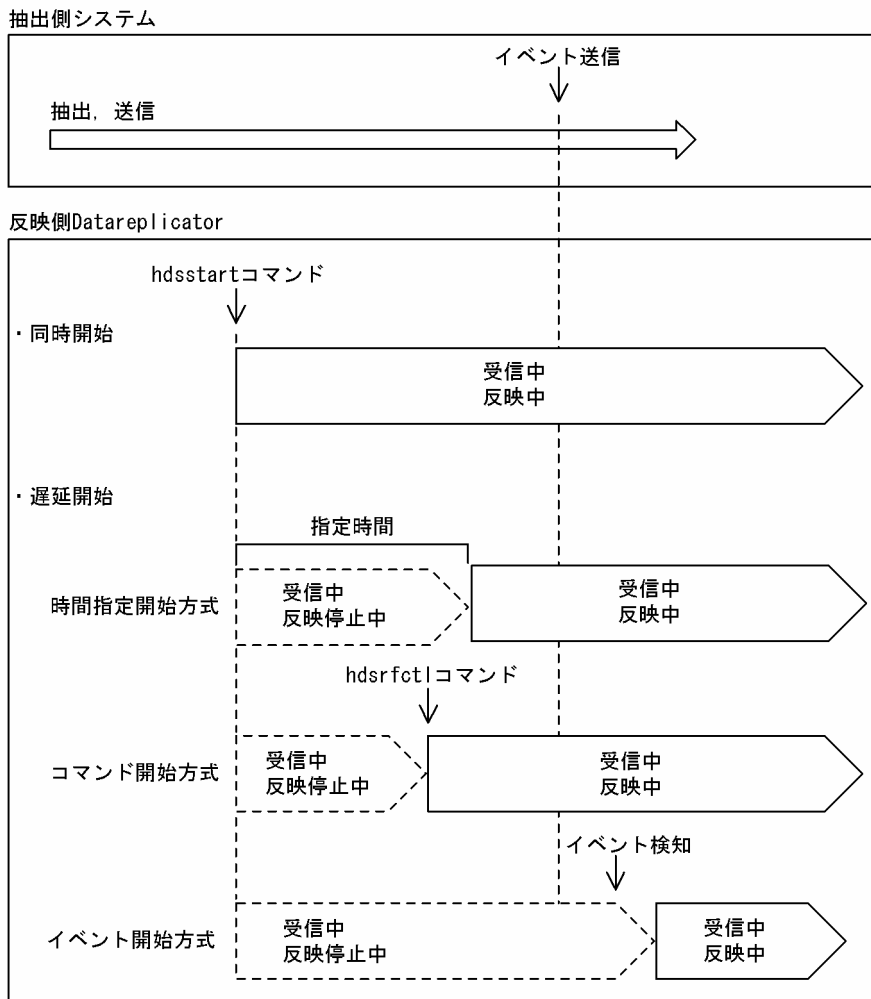
(c) HiRDB に同時にアクセスするテーブル数

HiRDB に同時にアクセスするテーブル数を，HiRDB の定義の `pd_max_access_tables` オペランドの指定値に加算してください。HiRDB に同時にアクセスするテーブル数は，反映定義に指定した load 文の数になります。

4.7.4 反映処理の開始方式の設計

反映処理の開始方式を設計します。反映処理の開始方式には，**同時開始**と**遅延開始**があります。反映処理の開始方式と開始タイミングの関係を次の図に示します。

図 4-44 反映処理の開始方式と開始タイミングの関係



(凡例)

→ : 受信処理と反映処理の両方が動作中

- - - - -> : 反映処理だけ停止中

(1) 同時開始

反映側 Datareplicator を起動すると同時に反映処理を開始します。同時開始とすると、更新情報を受信した順に逐次、反映できます。

使用目的

更新情報を受信した順に逐次、反映する場合に有効です。

反映環境定義の設定

startmode オペランドに trn (トランザクション単位反映方式)、又は tbl (表単位反映方式) を指定します。

コマンドの実行

hdsstart コマンドを実行します。

(2) 遅延開始

反映側 Datareplicator の起動時には、受信処理だけを開始し、任意のタイミングで反映処理を開始（遅延開始）します。反映処理を開始する前に、定義の変更、反映側 DB のメンテナンスなどをする場合に使います。

遅延開始としても受信処理は開始されているため、抽出側システムから送信された更新情報は反映情報キューファイルに格納されます。

遅延開始での開始方式には次の三つがあります。

(a) 時間指定開始方式

ユーザが指定した時間間隔後に反映処理を自動的に開始する方式です。指定する時間は、反映側 Datareplicator の起動時からの時間です。

使用目的

反映側 Datareplicator の起動後に、定期的なデータベースのメンテナンスなどの作業をする場合に有効です。

反映環境定義の設定

- startmode オペランドに spd を指定します。
- breaktime オペランドに反映処理を開始する時間を指定します。

コマンドの実行

hdsstart コマンドを実行します。

(b) コマンド開始方式

コマンドを実行してから反映処理を開始する方式です。この方式で、反映側システムから反映処理の動作を制御できます。

使用目的

日々の運用で、反映を開始する時間が異なる場合に使います。

反映環境定義の設定

startmode オペランドに spd を指定します。

コマンドの実行

hdsrctl コマンドを実行します。

(c) イベント開始方式

抽出側システムから送信されたイベントを検知した時点で、反映処理を開始する方式です。この方式で、抽出側システムから、反映処理の動作を制御できます。

使用目的

抽出側システムの運用によって反映処理を制御する場合に使います。

抽出側システムでの設定

反映処理を開始する直前の、抽出対象の UAP をイベントとして設定します。

反映環境定義の設定

startmode オペランドに spd を指定します。

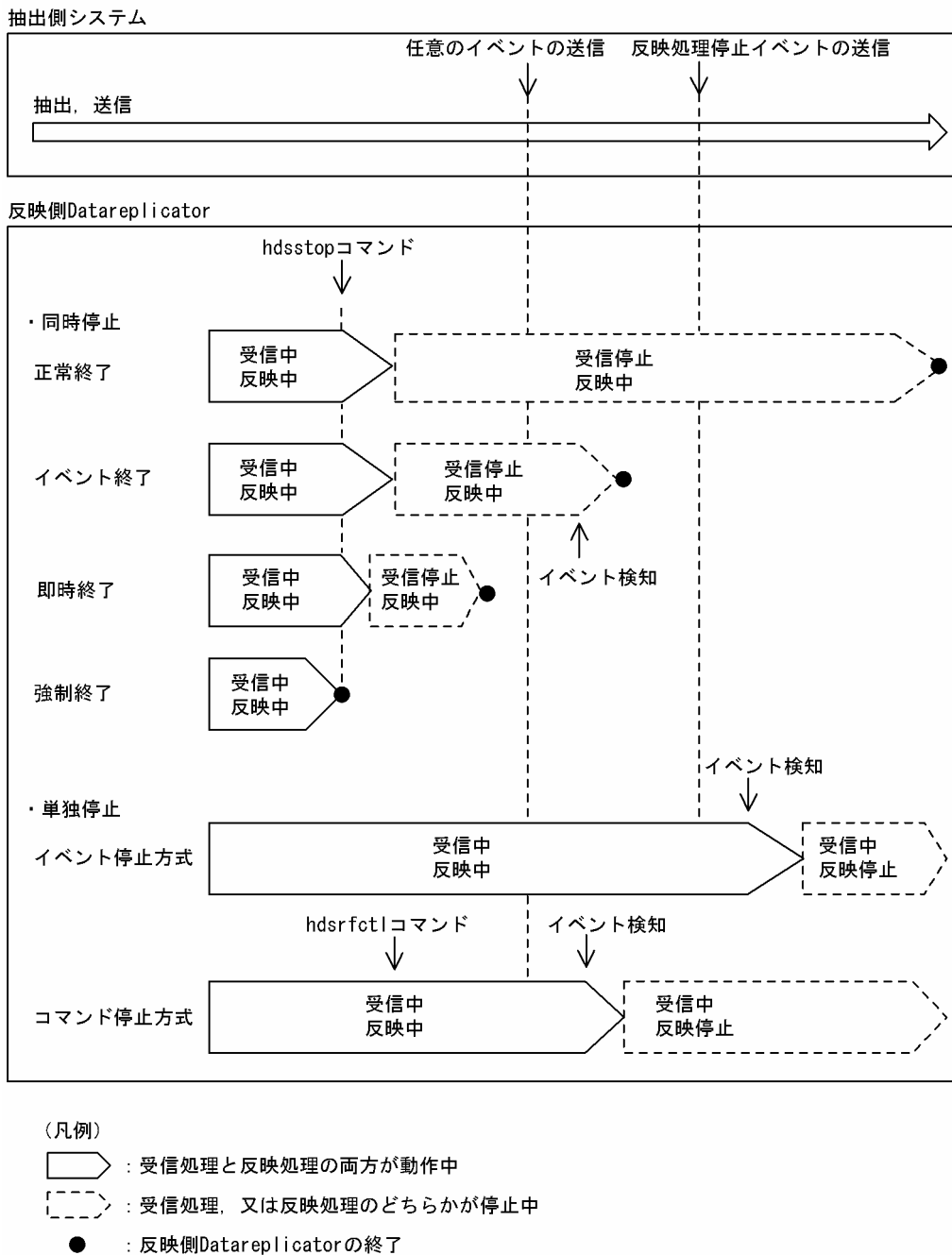
eventretmn 又は eventretbl オペランドで、抽出側システムで設定したイベントコードと、イベント受信後の反映処理の方式との対応付けを指定します。

4.7.5 反映処理の停止方式の設計

反映処理の停止方式を設計します。反映処理の停止方式には、**同時停止**と**単独停止**があります。

反映処理の停止方式と停止タイミングの関係を次の図に示します。

図 4-45 反映処理の停止方式と停止タイミングの関係



(1) 同時停止

受信処理と反映処理の両方を停止し、反映側 Datareplicator を終了します。同時停止をすると、受信処理が停止されるため、抽出側システムから送信される更新情報を受信しません。反映側 Datareplicator が終了するタイミングは、終了モードによって次のように異なります。

(a) 終了モードが正常終了の場合

hdsstop コマンドの実行時に稼働中の、抽出側システムの稼働単位までの反映処理が完了した時点で、反映処理を停止し、反映側 Datareplicator は終了します。

抽出側 DB が HiRDB の場合の稼働単位は、抽出側システム全体の正常開始から正常終了までです。抽出側 DB がメインフレーム側 DB の場合の稼働単位については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

使用目的

抽出側システム又は反映側システムを正常終了する場合に使います。

抽出側システムでの設定、操作

抽出側システムを正常終了します。

コマンドの実行

hdsstop コマンドを実行します。

(b) 終了モードがイベント終了の場合

hdsstop -t event コマンドを実行した後、抽出側システムから送信された何らかのイベントを検知した時点で、反映処理を停止し、反映側 Datareplicator は終了します。ただし、イベントを検知する前に、抽出側システムの稼働単位まで反映が完了した場合には、イベントを待たないで反映処理を停止します。

反映方式が表単位反映方式の場合には、動作中の反映処理のうち、最も遅いグループの反映が完了した時点で、反映処理を停止します。

イベント終了モードで反映側 Datareplicator を終了した場合、次回反映側 Datareplicator を開始すると、前回停止した時点の続きから反映処理を開始します。

使用目的

反映側システムを計画的に終了する場合に使います。

抽出側システムでの設定、操作

抽出側 DB が HiRDB の場合には、抽出側 Datareplicator で hdeevent コマンドを実行します。抽出側 DB がメインフレーム側 DB の場合には、反映処理を停止する直前の、抽出対象の UAP をイベントとして設定します。

コマンドの実行

hdsstop -t event コマンドを実行します。

(c) 終了モードが即時終了の場合

hdsstop -t immediate コマンドを実行した後、現在反映処理中の更新情報の反映が終了した時点で、反映処理を停止し、反映側 Datareplicator は終了します。

反映方式が表単位反映方式の場合には、動作中の反映処理のうち、最も反映の進んでいるグループから停止し、最も反映の遅れているグループの反映が、最も反映の進んでいるグループの停止地点に到達した時点で、反映処理が停止します。

コマンドの実行後から、最も反映の遅れているグループの反映が終了する時点までに更新情報を受信した場合には、更新情報が反映情報キューファイルに格納された時点で、受信処理が終了します。

使用目的

反映側システムを一時的に停止する場合、又は受信処理だけ実行中に反映処理を起動しないで停止する場合に実行します。

コマンドの実行

`hdsstop -t immediate` コマンドを実行します。

(d) 終了モードが強制終了の場合

`hdsstop -t force` コマンドを実行した時点で直ちに、反映処理を停止し、反映側 Datareplicator は終了します。同期点処理を実行中の場合には、同期点処理が完了してから反映処理を停止します。強制終了モードの場合、反映側 DB と抽出側 DB とで不整合が生じることがあります。ただし、次回の反映側 Datareplicator の起動時に、反映側 DB を自動的に回復します。

使用目的

反映側システムを緊急に停止する場合に使います。

コマンドの実行

`hdsstop -t force` コマンドを実行します。

(2) 単独停止

受信処理を停止しないで、反映処理だけを単独で停止します。単独停止としても受信処理は継続されているため、抽出側システムから送信された更新情報は反映情報キューファイルに格納されます。

単独停止での停止方式には次の二つがあります。

(a) コマンド停止方式

`hdsrfctl` コマンドを実行した後、抽出側システムから送信された何らかのイベントを検知した時点で、反映処理を停止する方式です。

使用目的

反映側システムから反映処理の停止を要求する場合に使います。

コマンドの実行

`hdsrfctl` コマンドを実行します。

(b) イベント停止方式

抽出側システムから送信された反映処理停止イベントを検知した時点で、反映処理を停止する方式です。

使用目的

抽出側システムから反映処理の停止を要求する場合に使います。

抽出側システムでの設定, 操作

抽出側 DB が HiRDB の場合には, 抽出側 Datareplicator で hdeevent コマンドを実行します。抽出側 DB がメインフレーム側 DB の場合には, 反映処理を停止する直前の, 抽出対象の UAP をイベントとして設定します。

反映環境定義の設定

eventspd オペランドで, 抽出側システムで設定したイベントコードとの対応付けをします。

コマンドの実行

コマンドの実行は不要です。

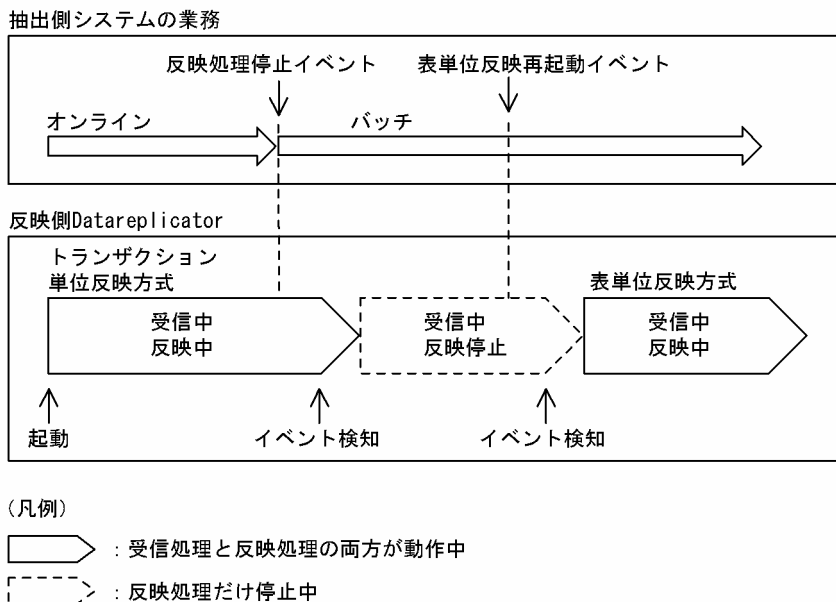
4.7.6 反映処理方式の切り替えの設計

反映側 Datareplicator では, 抽出側システムの業務の運用に合わせて, 反映処理の方式を変更できます。反映処理の方式を切り替えたい場合には, 抽出側システムから送信されるイベントに, 任意の方式を対応付ける必要があります。

また, 表単位反映方式を使っている場合には, イベント検知時に反映定義サーバプロセスが出力するメッセージを利用すれば, すべての反映プロセスでのイベントの検知を確認できます。これによって, 複数の反映プロセスで並行して反映処理をしているときでも, すべての反映プロセスでの処理の終了や開始を待って, 次の処理を始めるような運用ができます。

抽出側システムの業務をオンライン業務からバッチ業務に切り替えたときの反映処理方式の切り替えの設計例を次の図に示します。

図 4-46 反映処理方式の切り替えの設計例



(凡例)

- ➡ : 受信処理と反映処理の両方が動作中
- - - ➡ : 反映処理だけ停止中

(説明)

- 抽出側システムの業務形態がオンライン業務の場合、トランザクション単位反映方式で逐次反映しています。オンライン業務が終了した時点で、抽出側システムから反映処理停止イベントを発行します。
- 反映側システムでは反映処理停止イベントを検知した時点で、反映処理を停止します。
- 抽出側システムでのバッチ処理がある程度進んだ時点で、抽出側システムから表単位反映再起動イベントを発行します。
- 反映側システムでは表単位反映再起動イベントを検知した時点で、表単位反映方式で反映処理を再起動します。

4.7.7 反映側 Datareplicator のリソースの設計

反映側 Datareplicator のディスク資源とメモリ資源の設計方法について説明します。

(1) 反映側 Datareplicator のディスク資源の設計

反映側 Datareplicator のディスク資源の一覧を次の表に示します。

表 4-43 反映側 Datareplicator のディスク資源の一覧

ファイル名		ファイル種別		必要・任意 [個数]	容量の説明先	ファイル満杯時の動作
		R	C			
定義ファイル	反映システム定義ファイル	○	×	必要 [反映側システムごとに 1 個]	(a)	特にありません。
	反映環境定義ファイル	○	×	必要 [データ連動識別子ごとに 1 個]		
	反映定義ファイル	○	×	任意 [データ連動識別子ごとに 1 個]		
	二重化定義ファイル	○	×	任意 [反映側システムごとに 1 個]		
反映情報キューファイル※1※3		○	○ ※2	必要 [データ連動識別子ごとに 2~8 個]	(b)	スワップします。※1
反映ステータスファイル※1※3		○	○ ※2	必要 [データ連動識別子ごとに 1 個]	(c)	抽出側システムに対応した反映処理が停止します。
反映マスタステータスファイル		○	○ ※2	必要 [反映側システムごとに 1 個]	(d)	反映処理が停止します。
未反映情報ファイル		○	×	必要 [データ連動識別子ごとに 2 個]	(e)	スワップします。

ファイル名	ファイル種別		必要・任意 [個数]	容量の説明先	ファイル満杯時の動作
	R	C			
反映エラー情報ファイル	○	×	必要 [反映側システムごとに 2 個]	(f)	スワップします。
稼働トレースファイル (反映トレースファイル) ※4	○	×	任意 [反映側システムごとに 2 個]	(g)	スワップします。
更新情報定義ファイル	○	×	必要 [更新情報入力コマンドの実行ごとに 1 個]	(h)	特にありません。
SAM ファイル	○	×	必要 [更新情報入力コマンドの実行ごとに 1 個]	(i)	特にありません。
抽出対象外データ格納ファイル	○	×	必要 [データ連動識別子ごとに 1 個]	(j)	特にありません。
コマンドログファイル	○	×	任意 [2 個]	(k)	スワップします。
回復情報ファイル	○	○ ※2	任意 [データ連動識別子ごとに 1 個]	表 9-8 参照	特にありません。

(凡例)

R : UNIX の通常ファイル又は Windows ファイル

C : UNIX のキャラクタ型スペシャルファイル

○ : 作成できます。

× : 作成できません。

注※1

スワップ先のファイルの反映が完了していない場合には、次の送信間隔まで更新情報の受信を停止します。次の送信間隔時に反映が完了していれば、更新情報の受信を開始します。

注※2

AIX 版でキャラクタ型スペシャルファイルを使用する場合は、各ファイル容量の算出式に 1,024 バイト加算してください。また、反映環境定義の `queuesize` オペランド及び `statssize` オペランドには、算出した容量から 1,024 バイト減算した値を指定してください。

注※3

Datareplicator ファイルシステム領域を使用する場合は、「[3.4.3\(1\) Datareplicator ファイルシステム領域を割り当てるときの規則](#)」も参照してください。

注※4

稼働トレース収集方式を全プロセス一括収集方式とする場合のファイル名です。プロセス単位収集方式とする場合の稼働トレースファイル名については、「5.8 反映システム定義」の int_trc_unit オペランドの説明を参照してください。

(a) 定義ファイルの容量

反映側 Datareplicator の定義ファイル（反映システム定義ファイル，反映環境定義ファイル，反映定義ファイル）の容量は，指定した定義オペランドによって異なります。

(b) 反映情報キューファイルの容量

反映情報キューファイルには，接続情報，抽出定義情報，及び更新情報の3種類の情報が格納されます。各情報の詳細を次に示します。

- 反映情報キューファイルに格納される情報

- 接続情報

接続情報とは，抽出側システムと反映側 Datareplicator との接続についての情報です。抽出側システムとの接続を開始するときに1回だけ格納されます。

- 抽出定義情報

抽出定義情報とは，抽出側システムでの抽出定義についての情報です。

抽出側システムとの接続時（反映側 Datareplicator を初期化して起動した場合だけ）又は抽出定義を変更した後の抽出側システムとの接続時に格納されます。

- 更新情報

更新情報とは，抽出側システムで抽出した抽出側 DB の更新についての情報です。抽出側システムから更新情報を受信したときに格納されます。

- 反映情報キューファイルの容量の見積もり式

反映情報キューファイルの容量の見積もり式を次の表に示します。

表 4-44 反映情報キューファイルの容量の見積もり式

項目	見積もり式 (単位: バイト)	変数の説明
接続情報	512	なし。
抽出定義情報	$512 \times \uparrow (650 + D + E + F) \div 512 \uparrow$	$D = 32 + 96 \times \text{認可識別子数} + 64 \times \text{抽出定義表数} + 128 \times \text{抽出定義列数}$ $E = 88 \times \uparrow D \div \text{更新情報バッファサイズ} \uparrow$ $F = 32 \times \uparrow (D + E) \div 32453 \uparrow$
更新情報	$\Sigma (\text{TRN} + \Sigma \text{INS} + \Sigma \text{UPD} + \Sigma \text{DEL} + \Sigma \text{PRG})$ TRN = 104 INS = Ir Ic	Σ 最初の Σ は，更新情報の数の和を示します。INS, UPD, DEL, 及び PRG の Σ は，それぞれの行数の和を示します。

項目	見積もり式 (単位: バイト)	変数の説明
	$Ir = 96 + Row$ $Ic = 80 + (8 \times Cni) + Row$ $UPD = Ur Uc$ $Ur = 96 + Row$ $Uc = 90 + (8 \times Cnu) + Cl$ $DEL = 80 + (8 \times Dni) + Dl$ $PRG = 80$	TRN トランザクション情報 INS INSERT 行 UPD UPDATE 行 DEL DELETE 行 PRG PURGE TABLE 行 OR (選択) を示します Ir 行インタフェースの場合 Ic 列インタフェースの場合 Row 行の長さ Cni 挿入対象列の数 Ur 行インタフェースの場合 Ucn 列インタフェースの場合 Cnu 更新対象列の数とマッピングキーの数の和 Cl 更新列の長さとのマッピングキー列の長さの和 Dl マッピングキーの長さ Dni マッピングキー列の数

注※

抽出側 DB が HiRDB の場合には、更新情報バッファサイズは更新情報編集バッファサイズ（「5.2 抽出システム定義」の `smt_editbufsize` オペランド、又は「5.4 送信環境定義」の `editbufsize` オペランドの指定値）になります。抽出側 DB がメインフレーム側 DB の場合の更新情報バッファサイズについては、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

- 反映情報キューファイルの容量見積もり時の考慮点

反映情報キューファイルの容量見積もり時には、次に示すことを考慮してください。

• 反映処理の運用方式による容量の違い

反映処理の運用方式による容量見積もりの考慮点を次の表に示します。

表 4-45 反映処理の運用方式による容量見積もりの考慮点

運用方式	考慮点
同時開始	<ul style="list-style-type: none">トランザクションのピーク時に蓄積されるデータ量をディスク容量とします。ファイルを2個用意すれば、運用は可能と考えられます。ただし、何らかの要因で反映処理だけを停止する可能性がある場合には、ファイルの数を増やしておくことをお勧めします。
遅延開始、反映処理停止	<ul style="list-style-type: none">反映が開始又は再開される前に蓄積されるデータ量と反映開始後も蓄積されるデータ量を考慮してディスク容量を算出します。同時開始と遅延開始又は反映処理停止を併用する場合には、ファイルのサイズを小さめにしてファイルの数を多くした方が、ファイルの満杯を回避しやすいこともあります。

(c) 反映ステータスファイルの容量

反映ステータスファイルの容量の算出式を次に示します。

反映ステータスファイルには、抽出側 DB から送られてくる抽出定義情報が格納されます。抽出定義情報サイズは、抽出定義プリプロセスファイルのサイズで見積もります。したがって、抽出定義情報サイズの見積もりが不十分な場合は、抽出側から更新情報が送られてきたタイミングでファイル容量不足エラーになります。

UNIX の場合

$$293 \times 1024 + \text{抽出定義情報サイズ (単位: バイト)}$$

Windows の場合

$$282 \times 1024 + \text{抽出定義情報サイズ (単位: バイト)}$$

UNIX のキャラクタ型スペシャルファイルで、セクタ長が 1KB を超える場合

$$293 \times \text{SCT_SIZE} + \uparrow \text{抽出定義情報サイズ} \div \text{SCT_SIZE} \uparrow \times \text{SCT_SIZE (単位: バイト)}$$

SCT_SIZE: セクタ長 (単位: バイト)

(d) 反映マスタステータスファイルの容量

反映マスタステータスファイルの容量の算出式を次に示します。

反映マスタステータスファイルには、反映側 Datareplicator の初期化時に必要なサイズが割り当てられます。実際のレプリケーション運用中に、ファイル容量が増加し満杯となることはありません。

UNIX, 又は Windows の場合

$$2 \times 1024 \text{ (単位: バイト)}$$

UNIX のキャラクタ型スペシャルファイルで、セクタ長が 1KB を超える場合

$2 \times \text{SCT_SIZE}$ (単位: バイト)

SCT_SIZE: セクタ長 (単位: バイト)

(e) 未反映情報ファイルの容量

既定値は 16 キロバイトです。発生する可能性のある SQL エラーの容量が大きい場合には、ファイルの容量を拡張してください。

(f) 反映エラー情報ファイルの容量

既定値は 16 キロバイトです。長時間のエラー情報を残したいときは、ファイルの容量を拡張してください。

(g) 稼働トレースファイルの容量

稼働トレースファイルの容量は、反映システム定義の `int_trc_filesz` オペランドで指定します。

(h) 更新情報定義ファイルの容量

更新情報定義ファイルの容量は、指定した定義オペランドによって異なります。

(i) SAM ファイルの容量

PDMII E2 又は RDB1 E2 から転送されたファイルの大きさで決まります。

(j) 抽出対象外データ格納ファイルの容量

抽出対象外データの数によって異なります。容量は、ファイルシステムに余裕がある限り、拡張されます。

(k) コマンドログファイルの容量

128 キロバイト固定です。

(l) その他のファイル容量

Windows 版では、インストールディレクトリ下の `tmp` ディレクトリ下に、ワークファイルを幾つか作成します。このため、ワークファイル分として 4MB の容量を見積もってください。

(2) 反映側 Datareplicator のメモリ資源の設計

反映側 Datareplicator のメモリ資源の一覧を次の表に示します。

表 4-46 反映側 Datareplicator のメモリ資源の一覧

メモリ資源	容量の説明先
反映マスタプロセス	(a)
反映通信マスタプロセス	(b)

メモリ資源	容量の説明先
反映定義サーバプロセス	(c)
反映プロセス	(d)
反映 SQL プロセス	(e)
反映 UOC プロセス	(f)
更新情報入力プロセス	(g)
稼働トレース収集プロセス	(h)
Datareplicator エージェントプロセス	(i)
定義情報格納用共用メモリサイズ	(j)
反映プロセス間連絡用共有メモリサイズ	(k)
BLOB 列反映共有メモリサイズ	(l)
セマフォ所要数	(m)

表 4-47 反映側 Datareplicator のメモリ見積もり式に使用する変数の一覧

変数名	変数の内容
ATTR_NUM	データ連動の対象となる全抽象データ型の属性数の総和
CNST_NUM	反映定義で定義した全 format 文の const 句を指定した name 句の数の総和
COL_MAX	抽出対象表ごとに抽出対象列の実データ長の合計したうち、最大の長さ(バイト) (ただし、BLOB/BINARY 列は加算しない)
COL_MNUM	抽出対象表ごとのうち、最大の列数
COL_NUM	全抽出対象表の列の合計数
CUOC_CLM_NUM	列データ編集 UOC を指定した反映列の合計数
DUP_UPD	1 抽出表に複数定義した更新情報名数の合計数
ELEM_LEN	抽出対象表ごとに全繰返し列要素長の合計長のうち、最大の長さ(バイト) (定義長ではなく、抽出した実長)
ELEM_NUM	抽出対象表ごとに全繰返し列要素の合計数のうち、最大の要素数 (定義要素数ではなく、抽出した実要素数)
EXCS_R	反映対象となる列の総数 (反映環境定義 defmerge オペランドに true を指定して、load 文を省略した場合に仮定された反映列を含む)
FLD_NUM	反映定義で定義した全 load 文のフィールド名指定数の合計数
FOR_NUM	反映定義で定義した format 文の合計数
GRP_NUM	反映定義で定義した group 文の合計数
GRPD_NUM	反映定義で定義した group 文の分割指定数の合計数

変数名	変数の内容
GRPHA_NUM	反映定義 group 文で hash 句を指定した表のハッシュ分割 RD エリア数の合計数
GRPHC_LEN	反映定義 group 文で hash 句を指定した表のハッシュキー列値長の合計長 (バイト)
GRPHC_NUM	反映定義 group 文で hash 句を指定した表のハッシュキー列数の合計数
GRPK_NUM	反映定義で定義した group 文のキーレンジ分割条件数の合計数
GRPT_NUM	反映定義で定義した group 文の表指定数の合計数
IDX_NUM	反映対象表ごとに作成したインデクス数のうち、最大のインデクス数
IDXCLM_NUM	反映対象表ごとに作成したインデクスのインデクス構成列数の合計数のうち、最大数
KEY_MAX	抽出対象表ごとのマッピングキー長の合計長のうち、最大長 (バイト)
KEY_MNUM	抽出対象表ごとのマッピングキーの合計数のうち、最大数
KEY_NUM	全抽出対象表のマッピングキー数の合計数
LOD_NUM	反映定義で定義した load 文の合計数
NAME_NUM	反映定義で定義した全 format 文の name 句の合計数
NCHR_MAX	抽出対象の各国文字列属性 (NCHAR/NVARCHAR/LONG NVARCHAR) の列定義長のうち、最大長 (バイト)
PRP_FILE	抽出定義プリプロセスファイルサイズ (バイト)
RDEF_FILE	反映定義ファイルサイズ (バイト)
REP_NUM	全抽出対象表の繰返し列列数の合計数
RMST_FILE	反映マスタステータスファイルサイズ (バイト)
RST_FILE	反映ステータスファイルサイズ (バイト)
RUOC_ATTR_NUM	反映情報編集 UOC に渡す ADT 内データ属性数の合計数
RUOC_COL_MAX	反映情報編集 UOC に渡す反映列数のうち、最大数
RUOC_KEY_MAX	反映情報編集 UOC に渡すマッピングキー列数のうち、最大数
RUOC_TYPE_NUM	反映情報編集 UOC に渡す ADT 内データ型数の合計数
SKIP_LIST	反映抑止対象リストファイル内の行数
SKIPTYPEONLY	反映抑止対象リストファイルに指定した SKIP_TYPE_ONLY 句の合計数
SV_NUM	<抽出元が HiRDB の場合> 抽出対象の全バックエンドサーバ数 <抽出元が HiRDB 以外の場合> 1
TBL_NUM	抽出対象表の合計数
TBLT_R	反映対象となる表の合計数 (反映環境定義 defmerge オペランドに true を指定して、load 文を省略した場合に仮定された反映表を含む)

変数名	変数の内容
TYPE_NUM	全抽出対象表の抽象データ型の合計数
UBUF	<p><抽出側 DB が HiRDB の場合></p> <ul style="list-style-type: none"> 抽出システム定義 sendcontrol に nodemst を指定した場合 送信環境定義 editbufsize 指定値 抽出システム定義 sendcontrol に sendmst を指定した場合 抽出システム定義 smt_editbufsize 指定値 <p><抽出側が XDM/DS の場合></p> <p>XDM/DS 起動定義 REFLECTBUFF 指定値。詳細については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください（注意 (KB)のまま指定する）。</p>
UPD_NUM	更新情報名の合計数

(a) 反映マスタプロセス

- プロセス数
反映側 Datareplicator に一つ必要です。
- プロシジャサイズ
204800 バイト
- 動的メモリサイズ (単位: バイト)
下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。
(固定部)
1024

(b) 反映通信マスタプロセス

- プロセス数
(対応するデータ連動識別子の数 + 1) 必要です。
- プロシジャサイズ
256000 バイト
- 動的メモリサイズ (単位: バイト)
下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。
(固定部)

$$UBUF \times 1024 + 136$$

$$+ UBUF \times 1024$$

$$+ \uparrow (UBUF \times 1024 + 64) / 1024 \uparrow \times 1024$$

$$+ 216$$

$$+ 281$$

$$+ 1024$$

$$+ 68228$$

+ 5219
 + 10280 + MAX (1024 , PRP_FILE / SV_NUM)
 + MAX (RMST_FILE , RST_FILE) - 1024
 + 512000

(可変部)

- 二重化機能使用時に加算する式
65536

変数の説明

表 4-47 を参照してください。

(c) 反映定義サーバプロセス

- プロセス数

対応するデータ連動識別子の数だけ必要です。

- プロシジャサイズ

409600 バイト

- 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

(↑ (UPD_NUM + LOD_NUM) / 16 ↑ × 16) × 160
 + (↑ (COL_NUM + FLD_NUM) / 16 ↑ × 16 + TBL_NUM) × 128
 + ↑ (PRP_FILE / SV_NUM + 152) / 1024 ↑ × 1024
 + PRP_FILE / SV_NUM
 + ↑ (PRP_FILE / SV_NUM + 64) / 1024 ↑ × 1024
 + PRP_FILE / SV_NUM
 + ↑ (PRP_FILE / SV_NUM + 64) / 1024 ↑ × 1024
 + (↑ COL_NUM / (TBL_NUM × 16) ↑ + TBL_NUM) × 384
 + ↑ TBLT_R / 16 ↑ × 704
 + 260
 + 68228
 + 5219
 + 10280 + MAX (1024 , PRP_FILE / SV_NUM)
 + MAX (RMST_FILE , RST_FILE) - 1024
 + ↑ UPD_NUM / 16 ↑ × 704
 + 512000

(可変部)

- 反映定義を指定した場合に加算する式

RDEF_FILE

- 反映定義に const 句を指定した format 文を定義した場合に加算する式
 $\uparrow \text{CNST_NUM} / 16 \uparrow \times 832$
- 反映定義に format 文を定義した場合に加算する式
(\uparrow 反映定義で定義した format 文の数の総和 / $16 \uparrow * 16$) * 36
- 反映定義に format 文を定義した場合に加算する式
 $\uparrow \text{FOR_NUM} / 16 \uparrow \times 576$
+ $\uparrow \text{NAME_NUM} / 16 \uparrow \times 768$
- 反映定義に load 文を定義した場合に加算する式
 $\uparrow \text{LOD_NUM} / 16 \uparrow \times 832$
+ $\uparrow \text{LOD_NUM} / 8 \uparrow \times 704$
+ $\uparrow \text{FLD_NUM} / 16 \uparrow \times 384$
- load 文のフィールド指定に*を指定していない場合に加算する式
 $\uparrow \text{FLD_NUM} / 16 \uparrow \times 896$
- load 文に指定した反映先表に文字集合列が存在する場合に加算する式
32
- 反映定義に group 文を定義した場合に加算する式
 $\uparrow (\text{GRP_NUM} + 2) / 16 \uparrow \times 2752$
+ ($\uparrow \text{GRPT_NUM} / 16 \uparrow + \text{GRP_NUM}$) $\times 64$
- group 文に分割指定有りの場合に加算する式
 $\uparrow (\text{GRPD_NUM} + 2) / 16 \uparrow \times 1024$
- 次の条件の場合に加算する式
反映環境定義 mapping_key_check 句で not_null_unique 又は unique を指定
反映定義 LOAD 文の check 句で not_null_unique 又は unique を指定
 $8 \times \text{IDX_NUM} + 32 \times \text{IDXCLM_NUM}$
+ $16 + 4 \times \text{IDX_NUM}$
+ 580
- 反映定義の group 文にキーレンジ分割条件を定義した場合に加算する式
($\uparrow \text{GRPK_NUM} / (\text{GRPD_NUM} \times 16) \uparrow + \text{GRPD_NUM}$) $\times 4416$
+ 32
- 文字集合指定列連携時に加算する式
32
- 抽象データ型列連携時に加算する式
 $64 + 112 \times \text{TYPE_NUM} + 64 \times \text{ATTR_NUM}$
- 二重化機能使用時に加算する式

変数の説明

表 4-47 を参照してください。

(d) 反映プロセス

• プロセス数

トランザクション単位反映方式の場合は一つだけ必要です。表単位反映方式の場合は、最大反映グループの数だけ必要です。

• プロシジャサイズ

256000 バイト

• 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

1464
 + MAX (48000 , COL_MAX + 4096)
 + 32
 + 67
 + UBUF × 1024
 + 60
 + 845 + KEY_MAX
 + 19632
 + 12 × ↑ KEY_MNUM / 10 ↑ × 10
 + 12 × ↑ COL_MNUM / 50 ↑ × 50
 + 68228
 + 5219
 + 10280 + MAX (1024 , PRP_FILE / SV_NUM)
 + MAX (RMST_FILE , RST_FILE) - 1024
 + 512000

(可変部)

- 反映定義 group 文で hash 句を指定時に加算する式
 GRPHA_NUM × 4
 + GRPHC_NUM × 4
 + GRPHC_NUM × 4
 + GRPHC_NUM × 256
- 反映定義 group 文で hash 句を指定し、ハッシュキー列に文字集合指定列が一つ以上ある場合に加算する式

GRPHC_NUM × 4

- 反映定義 group 文で hash 句を指定し、ハッシュ値不正時に加算する式

GRPHC_LEN

- 反映抑止機能使用時に加算する式

92

+ 52 × SKIP_LIST

+ ↑ SKIPTYPEONLY / 10 ↑ × 40

- 抽出元が XDM/DS で、NCHAR/NVARCHAR/LONG NVARCHAR 型をデータ連動する場合に加算する式

NCHR_MAX + 2

- 二重化機能使用時に加算する式

65536

変数の説明

表 4-47 を参照してください。

(e) 反映 SQL プロセス

- プロセス数

トランザクション単位反映方式の場合は一つだけ必要です。表単位反映方式の場合は、最大反映グループの数だけ必要です。

- プロシジャサイズ

307200 バイト

- 動的メモリサイズ (単位: バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

1448

+ 88 + 40 × COL_MNUM + 39 × KEY_MNUM

+ 197 + 40 × COL_MNUM + 39 × KEY_MNUM

+ 4096

+ 4096

+ 68 × TBLT_R

+ 32 × TBLT_R + 16 × (EXCS_R - 1) + 2 × EXCS_R

+ 32 × TBLT_R + 16 × (EXCS_R + KEY_NUM - 1) + 2 × (EXCS_R + KEY_NUM)

+ 32 × TBLT_R + 16 × (EXCS_R - 1) + 2 × EXCS_R

+ 2 × EXCS_R

+ 68228

+ 5219

+ 10280 + MAX (1024 , PRP_FILE / SV_NUM) + MAX (RMST_FILE, RST_FILE) - 1024
+ 512000

(可変部)

- 反映定義 LOAD 文で WITH LOCK 句を指定時に加算する式
 $48 \times \uparrow \text{LCKT_NUM} / 10 \uparrow \times 10$
- 文字集合指定列連携時に加算する式
 $96 \times \text{TBLT_R} + 2 \times \text{EXCS_R}$
 $+ 96 \times \text{TBLT_R} + 2 \times (\text{EXCS_R} + \text{KEY_NUM})$
 $+ 96 \times \text{TBLT_R} + 2 \times \text{KEY_NUM}$
- 抽象データ型列連携時に加算する式
 $88 + 35 \times \text{COL_MNUM}$
- 繰り返し列連携時に加算する式
 $16 \times (\text{REP_NUM} + 1)$
 $+ (\text{ELEM_LEN} + 2 \times \text{ELEM_NUM} + 8 \times \text{TBLT_R})$
- 抽象データ型列又は繰り返し列連携時に加算する式
 $84 + 40 \times \text{COL_MNUM} + 39 \times \text{KEY_MNUM}$
- 文字集合指定列連携時に加算する式
 $52 \times \text{CNST_NUM}$
- 列データ UOC 使用時に加算する式
 $\text{CUOC_CLM_NUM} \times 48$
 $+ \text{CUOC_CLM_NUM} \times 32008$
- 1 抽出表に複数定義した更新情報がある時に加算する式
 $20 \times \uparrow \text{DUP_UPD} / 5 \uparrow$
- 抽象データ型列連携時に加算する式
 $80 \times \text{TYPE_NUM}$
- 二重化機能使用時に加算する式
65536

変数の説明

表 4-47 を参照してください。

注意

最大更新行長を算出する場合に、抽出列型として BLOB 型があるときは、その列は 20 バイトで計算してください。

(f) 反映 UOC プロセス

- プロセス数

UOC を使うときに、一つだけ必要です。

- プロシジャサイズ

(102400 + UOC サイズ) バイト

- 動的メモリサイズ (単位：バイト)

下記の固定部分と条件に合致する可変部分で算出した値の合計値が動的メモリサイズとなります。

(固定部)

408
+ MAX (48000 , COL_MAX + 4096)
+ 68228
+ 5219
+ 10280 + MAX (1024 , PRP_FILE / SV_NUM)
+ MAX (RMST_FILE , RST_FILE) - 1024
+ 512000
+ 48 × RUOC_COL_MAX
+ 48 × RUOC_KEY_MAX
+ 824

(可変部)

- 反映情報編集 UOC を使用する更新情報名に ADT 列がある場合に加算する式
 $120 \times \uparrow \text{RUOC_TYPE_NUM} / 10 \uparrow$
 $+ 84 \times \text{RUOC_TYPE_NUM} + 60 \times \text{RUOC_ATTR_NUM}$
- ファイル二重化機能を使用する場合に加算する式
65536

変数の説明

表 4-47 を参照してください。

(g) 更新情報入力プロセス

- プロセス数

SAM ファイルを使ったデータ連動を使うときに、一つだけ必要です。

- プロシジャサイズ

250 キロバイト

- 動的メモリサイズ (単位：キロバイト)

$300 + \text{DBUF} + \text{TBUF}$

変数の説明

DBUF : $0.5 \times$ 更新情報定義の抽出文の総フィールド数

TBUF : 1 トランザクション内更新データの総長 $\times 2$

(h) 稼働トレース収集プロセス

- プロセス数
反映側 Datareplicator に一つ必要です。
- プロシジャサイズ
60 キロバイト
- 動的メモリサイズ
50 キロバイト

(i) Datareplicator エージェントプロセス

- プロセス数
抽出側システム又は反映側システムどちらか一方で加算します。
- プロシジャサイズ
70 キロバイト
- 動的メモリサイズ (単位：キロバイト)

$$\uparrow 300000^{\ast} + (\text{HDS_NUM} + \text{DC_NUM} \times 3 + \text{GRP_NUM}) \times 64 \div 1024 \uparrow$$

変数の説明

HDS_NUM：全反映側システム数

DC_NUM：全データ連動数

GRP_NUM：全反映グループ数

注※

抽出側システム又は反映側システムどちらか一方で加算します。

(j) 定義情報格納用共用メモリサイズの見積もり方法

反映環境定義の defshmsize オペランドで指定する定義情報格納用共用メモリサイズの見積もり式を次に示します。

$$\uparrow 192 \times (\text{GRPT} + 2) + 160 \times \text{TBLT_E} + 152 \times \text{EXCS_E} + 156 \times \text{TBLT_r} \\ + 128 \times \text{EXCS_r} + 88 \times \text{TBLT_R} + 24 \times \text{EXCS_R} + 52 \times \text{CNST} \uparrow \text{ (単位：バイト)}$$

GRPT：反映定義ファイル内の反映グループ定義数（反映定義の group 文の数）

TBLT_E：抽出表数

EXCS_E：抽出列数

TBLT_r：反映定義ファイル内の反映表定義数（反映定義の load 文の数）

EXCS_r：反映表定義で指定した反映表の列数の総数

TBLT_R：反映表数（実際に反映対象となる表数。「5.9 反映環境定義」の defmerge オペランドに true を指定して、load 文を省略した場合に仮定された反映表を含む）

EXCS_R：反映列数（実際に反映対象となる列数。「5.9 反映環境定義」の defmerge オペランドに true を指定して、load 文を省略した場合に仮定された反映列を含む）

CNST：反映定義ファイル内の更新情報フィールド定義で指定した const 句の総数

(k) 反映プロセス間連絡用共有メモリサイズの見積もり方法

$$65089 + 290592 \times UJ \text{ (単位 : バイト)}$$

UJ：抽出側システム数

(l) BLOB 列反映共有メモリサイズの見積もり方法

反映グループごとに必要になります。

$$(LOB_SIZE + 4) \times 4 + 80 \text{ (単位 : バイト)}$$

LOB_SIZE：反映グループ内の（各更新情報名に含まれる BLOB 列の定義長の合計）の最大値

(m) セマフォ所要数の見積もり方法

反映側 Datareplicator が使うセマフォ数の見積もり式を次に示します。

$$5 + 10 \times UJ$$

UJ：抽出側システム数

上記の見積もり式で見積もった数のセマフォを反映側システムで使えるように、必要に応じてカーネルパラメタのシステム全体のセマフォ最大数 (SEMMNS) を変更してください。カーネルパラメタの更新方法については、該当する OS のマニュアルを参照してください。

4.8 抽出側 HiRDB の設計

抽出側 HiRDB のシステム設計について説明します。

HiRDB のデータベースからデータを抽出する場合、抽出側 HiRDB は抽出側 Datareplicator が必要とするデータ連動情報を、データベースの更新ログに付けてシステムログファイルに出力します。抽出側 Datareplicator は、抽出側 HiRDB が出力したシステムログファイル中のデータベースの更新ログを基に、データ連動を実行します。抽出側 HiRDB でのデータ連動情報のシステムログファイルへの出力機能を、HiRDB Datareplicator 連携機能といいます。

抽出側 HiRDB で HiRDB Datareplicator 連携を開始すると、データ連動情報が付けられたシステムログがシステムログファイルに出力されます。抽出側 Datareplicator を起動して抽出処理を開始すると、抽出側 Datareplicator はシステムログファイルからのシステムログの抽出を開始します。

HiRDB のデータベースからデータを抽出するためには、抽出側 HiRDB に次の指定が必要です。

- 抽出側 Datareplicator が使うディレクトリ名の指定
- HiRDB Datareplicator 連携の開始指定

4.8.1 抽出側 Datareplicator が使うディレクトリ名の指定

抽出側 HiRDB のユニット制御情報定義で、抽出側 Datareplicator が使うディレクトリ名を指定します。抽出側 HiRDB は、このディレクトリ下にあるデータ連動用連絡ファイルを、抽出側 Datareplicator との抽出状態の連絡に使います。この指定がない場合、HiRDB Datareplicator 連携機能は使えません。指定方法を次に示します。

```
pd_rpl_hdepath=抽出側HiRDB Datareplicator運用ディレクトリ名
```

ユニット制御情報定義での pd_rpl_hdepath の指定方法については、「[5.6 抽出側 HiRDB の定義](#)」を参照してください。

4.8.2 HiRDB Datareplicator 連携の開始指定

抽出側 HiRDB でデータ連動情報をシステムログに付けて出力するには、次のどちらかを指定して、HiRDB Datareplicator 連携を開始する必要があります。

- 抽出側 HiRDB のシステム共通定義での pd_rpl_init_start = Y 指定

抽出側 HiRDB 開始時から HiRDB Datareplicator 連携を開始する場合に指定します。この場合、抽出側 HiRDB 開始時から、データ連動に必要なデータ連動情報がシステムログファイルに出力されます。システム共通定義での pd_rpl_init_start の指定方法については、「[5.6 抽出側 HiRDB の定義](#)」を参照してください。

- pdrplstart コマンドの実行（システム共通定義で pd_rpl_init_start = Y を指定していない場合又は pdrplstop コマンドで HiRDB Datareplicator 連携を中止している場合）

システム共通定義で pd_rpl_init_start = Y を指定していない場合は、HiRDB の pdrplstart コマンドを実行して HiRDB Datareplicator 連携を開始します。

pdrplstart コマンドを実行した場合、抽出側 HiRDB は pdrplstart コマンド実行後に実行するトランザクションの更新情報からデータ連動情報の出力を開始します。pdrplstart コマンド実行前に、データ連動対象 DB を更新するトランザクションを実行している場合、抽出側 DB と反映側 DB との整合性は保証されません。このため、pdrplstart コマンドを入力する場合は、データ連動対象 DB を更新するトランザクションが実行できない状態で入力してください。

HiRDB の pdrplstop コマンドを実行して HiRDB Datareplicator 連携を中止している場合に、データ連動を再開したいときも pdrplstart コマンドを実行します。pdrplstop コマンドを実行して HiRDB Datareplicator 連携を中止した場合は、抽出側 DB と反映側 DB との間に不整合が発生していますので、pdrplstart コマンドの実行前に反映側 DB の再作成が必要です。

pdrplstart コマンド又は pdrplstop コマンドを実行する場合の運用手順と反映側 DB の再作成時の運用手順については、「[6.5.6 抽出側 HiRDB の運用手順](#)」を参照してください。

pdrplstart コマンドと pdrplstop コマンドの形式については、「[7. コマンドの文法](#)」を参照してください。

4.8.3 HiRDB Datareplicator 連携を開始した後の注意事項

HiRDB Datareplicator 連携を開始した後、長時間にわたり抽出側 Datareplicator の抽出処理を停止しておくと、HiRDB のシステムログファイルが満杯となり、HiRDB が停止することがあります。そのため、HiRDB の稼働中は、抽出側 Datareplicator の抽出処理を停止しない運用をお勧めします。

5

定義

この章では、抽出側 Datareplicator と反映側 Datareplicator にデータ連動システムの設計内容を定義する方法と、定義例について説明します。

5.1 Datareplicator の定義の概要

Datareplicator の定義の概要について説明します。

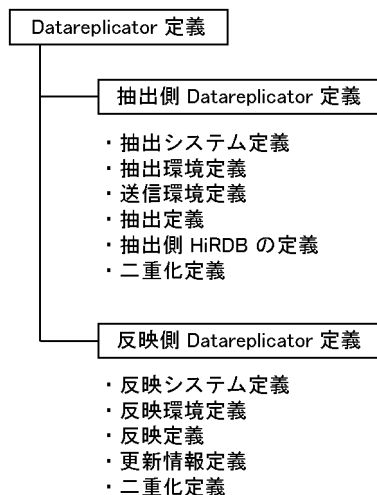
メインフレーム側のデータ連動製品 (XDM/DS) に定義する内容については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

5.1.1 Datareplicator の定義の構成

Datareplicator の定義は、抽出側 Datareplicator と反映側 Datareplicator で別々の定義ファイルを作成します。どちらの定義も、データ連動システムを起動する前に作成しておく必要があります。さらに、抽出側 HiRDB のシステム定義には、Datareplicator を使ってデータ連動することを指定しておく必要があります。

Datareplicator の定義の体系を次の図に示します。

図 5-1 Datareplicator の定義の体系



Datareplicator の定義ファイルの作成方法については、「4.6.2 抽出側 Datareplicator で使うファイルの準備」又は「4.7.2 反映側 Datareplicator で使うファイルの準備」を参照してください。

定義ファイルのサンプル

Datareplicator では、定義ファイルのテンプレート (サンプル) を提供しています。テンプレートの内容を定義ファイルのディレクトリ (HDSPATH) 又は任意のディレクトリ下にコピーして内容を流用すれば、最初から定義ファイルを作成する手間が省けます。Datareplicator の定義ファイルのサンプルについては、「5.13.2 抽出側 Datareplicator の定義の例」又は「5.13.3 反映側 Datareplicator の定義の例」を参照してください。

抽出側 Datareplicator の定義の一覧を表 5-1 に、反映側 Datareplicator の定義の一覧を表 5-2 に示します。

表 5-1 抽出側 Datareplicator の定義の一覧

抽出側 Datareplicator の定義名	定義ファイル名	必要・任意	内容
抽出システム定義	\$HDEPATH/hdeenv	必要	抽出側 Datareplicator の稼働環境についての情報を定義します。定義する情報には、抽出側 Datareplicator 識別子、送信先識別子などがあります。
抽出環境定義	任意の名称	必要	抽出処理の稼働環境についての情報を定義します。定義する情報には、抽出情報キューファイル名、抽出情報キューファイルサイズなどがあります。
送信環境定義	任意の名称	必要	送信処理の稼働環境についての情報を定義します。定義する情報には、通信のためのサービス名、ホスト名などがあります。
抽出定義	任意の名称	必要	抽出、送信処理の内容を定義します。定義する情報には、抽出対象表と更新情報の対応付け、更新情報の送信先識別子などがあります。
抽出側 HiRDB の定義	HiRDB の次の定義ファイル <ul style="list-style-type: none"> システム共通定義ファイル ユニット制御情報定義ファイル 	必要	抽出側 HiRDB で抽出側 Datareplicator を使うために必要な情報を定義します。定義する情報には、データ連動の開始方法、抽出側 Datareplicator が使うディレクトリ名などがあります。
二重化定義	抽出システム定義ファイルの file_dupenv オペランドに指定したファイル名	任意	抽出側システムで使用する定義ファイルを二重化する場合に定義します。二重化できるファイルには、抽出マスターステータスファイル、抽出サーバステータスファイル、抽出情報キューファイル、及びデータ連動用連絡ファイルがあります。

表 5-2 反映側 Datareplicator の定義の一覧

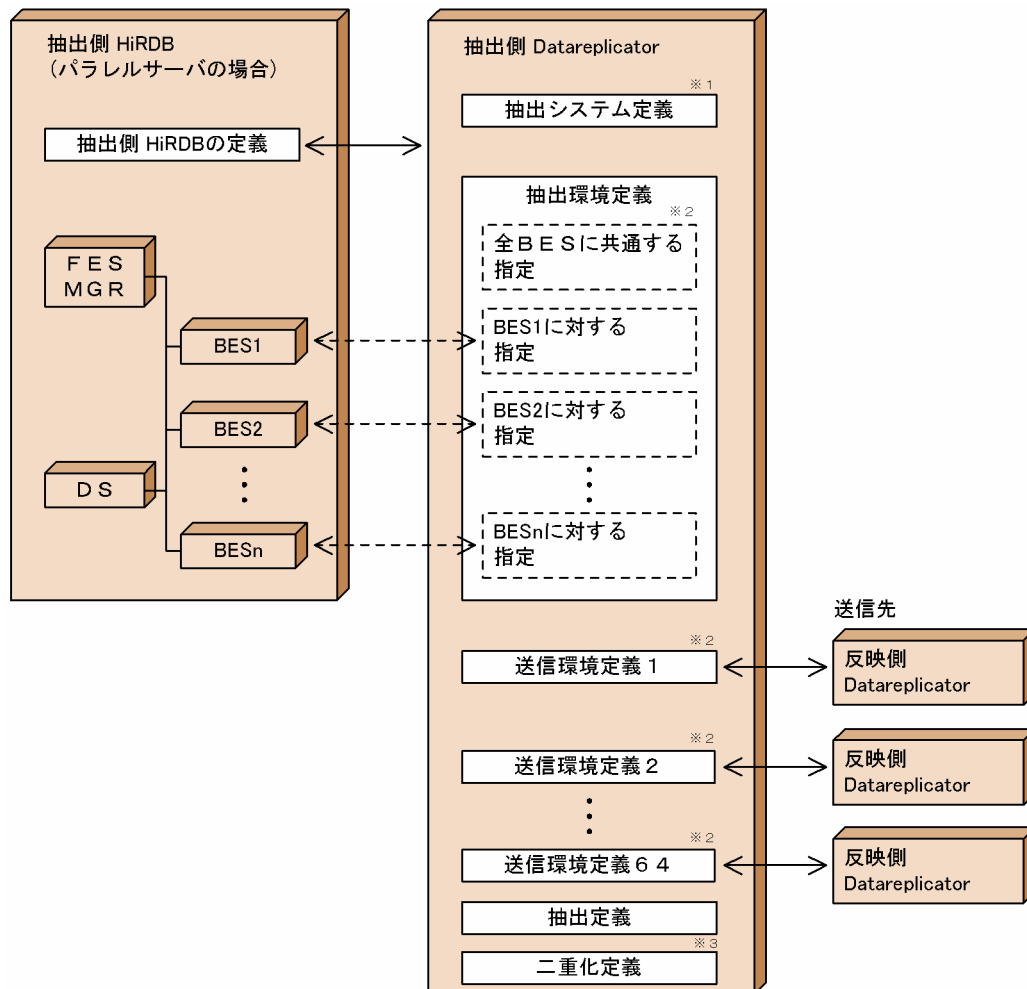
反映側 Datareplicator の定義名	定義ファイル名	必要・任意	内容
反映システム定義	\$HDSPATH/hdsenv	必要	反映側 Datareplicator の稼働環境についての情報を定義します。定義する情報には、通信プロトコル、抽出側システムのデータ連動識別子などがあります。
反映環境定義	任意の名称	必要	反映処理の稼働環境についての情報を定義します。定義する情報には、反映情報キューファイル名、反映ステータスファイル名、反映処理の方式などがあります。
反映定義	任意の名称	任意	反映処理の内容を定義します。定義する情報には、更新情報と反映対象表との対応付け、反映先のフロントエンドサーバ、キーレンジ分割条件、ハッシュ分割条件、UOC 名などがあります。
更新情報定義	任意の名称	任意	SAM ファイルを使ってデータ連動する場合に、抽出側 DB が PDMII E2 のときに定義します。

反映側 Datareplicator の定義名	定義ファイル名	必要・任意	内容
二重化定義	反映システム定義ファイルの file_dupenv オペランドに指定したファイル名	任意	反映側システムで使用する定義ファイルを二重化する場合に定義します。二重化できるファイルには、抽出マスタステータスファイル、抽出サーバステータスファイル、抽出情報キューファイル、及びデータ連動用連絡ファイルがあります。

(1) 抽出側 Datareplicator の定義とデータ連動システムの関係

抽出側 Datareplicator の定義とデータ連動システムの関係を次の図に示します。

図 5-2 抽出側 Datareplicator の定義とデータ連動システムの関係



(凡例)
MGR : HiRDBのシステムマネージャ FES : HiRDBのフロントエンドサーバ
BES : HiRDBのバックエンドサーバ DS : HiRDBのディクショナリサーバ

注※1

抽出側 HiRDB がパラレルサーバで、さらに環境変数に対して、サーバマシンごとに異なる指定をしている場合には、環境変数に対するオペランドをサーバマシンごとに個別に指定できます。

注※2

抽出側 HiRDB のすべての BES に対して共通の指定をすることも、また各 BES に対して必要に応じて個別に指定することもできます。

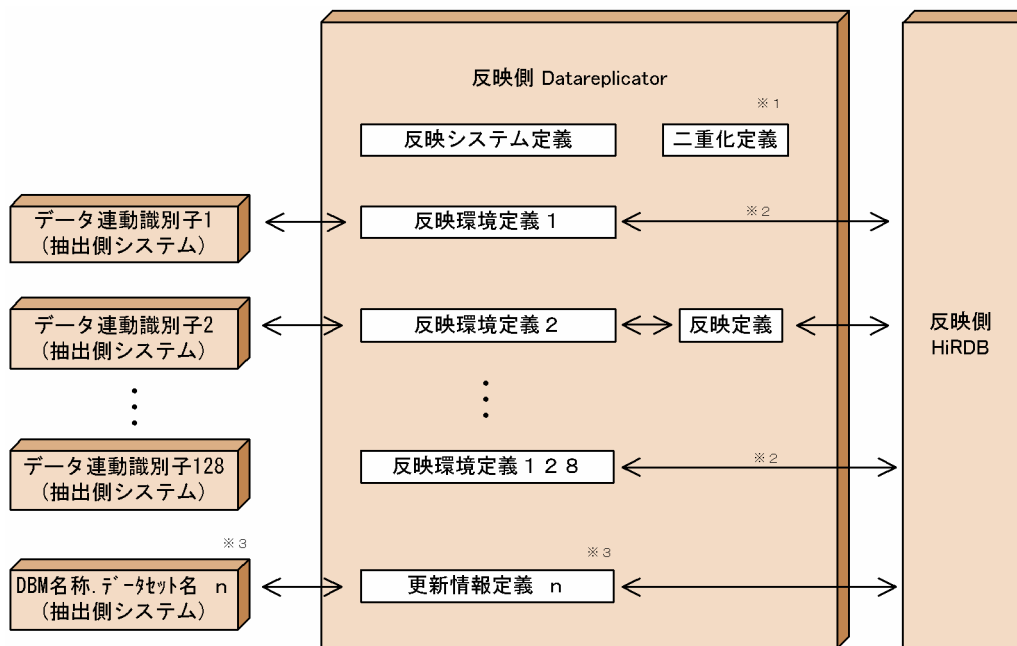
注※3

二重化定義は、ファイルの二重化を使用する場合に必要です。

(2) 反映側 Datareplicator の定義とデータ連動システムの関係

反映側 Datareplicator の定義とデータ連動システムの関係を次の図に示します。

図 5-3 反映側 Datareplicator の定義とデータ連動システムの関係



注※1

二重化定義は、ファイルの二重化を使用する場合に必要です。

注※2

抽出対象の表と反映対象の表とで、表の形式、表名、及び列名がすべて同じ場合には、反映定義を省略できます。

注※3

SAM ファイルを使った抽出側システムのうち、PDM II E2 の場合に更新情報定義が必要です。

5.1.2 定義の規則

Datareplicator の定義は、次の三つの形式で記述します。

set 形式

抽出システム定義, 抽出環境定義, 送信環境定義, 反映システム定義, 反映環境定義

構文形式

抽出定義, 反映定義, 更新情報定義

ファイル独自の形式

二重化定義

抽出側 HiRDB がパラレルサーバで、さらに HiRDB の環境変数 (PDDIR, PDCONFPATH, SHLIB_PATH など) にサーバマシンごとに異なる指定をしている場合には、抽出側 Datareplicator の定義の抽出システム定義では、環境変数に対するオペランドをサーバマシンごとに個別に定義できます。

抽出側 HiRDB がパラレルサーバの場合には、抽出側 Datareplicator の定義の抽出環境定義と送信環境定義では、バックエンドサーバごとに環境を定義できます。

(1) set 形式の定義の規則

- 定義文の 1 行の長さは、半角で 80 文字以内にしてください。80 文字を超える場合は、改行の直前に継続符号「¥」を記述し、複数行に分けて記述してください。また、継続行は 1 カラム目から記述してください。
- コメントを記述する場合は、コメントの先頭に「#」を記述します。「#」以降から行末までがコメントになります。コメントは複数行に継続して記述できません。
- 各オペランドの記述順序に制限はありません。
- 同じオペランドを複数個指定すると、定義ファイルの中の最後に出現するオペランドが有効になります。
- パラメタ名称だけを指定して、パラメタ値を指定していない場合は、パラメタ値不正のエラーとなります。

(2) 構文形式の定義の規則

- コメントを記述する場合は、コメントの先頭に「/*」, 末尾に「*/」を記述します。
- コメントは複数行にわたって記述できます。ただし、「/と*」又は「*と/」は連続して記述してください。
- 文字列定数として「'」(アポストロフィ) を記述する場合は、「'」とアポストロフィを二つ記述することで一つの「'」を表します。
- Datareplicator の予約語と同名の識別子を指定する場合は、識別子を「"」(ダブルクォーテーション) と「"」(ダブルクォーテーション) で囲んで指定してください。Datareplicator の予約語については、「付録 B Datareplicator 定義の予約語」を参照してください。
- 最後に記述した定義文の文末には「;」(セミコロン) を指定してください。

(3) ファイル独自の形式の定義の規則

二重化定義の定義形式については、「5.7 二重化定義（抽出側）」又は「5.12 二重化定義（反映側）」を参照してください。

(4) サーバマシンごとの定義（抽出システム定義）

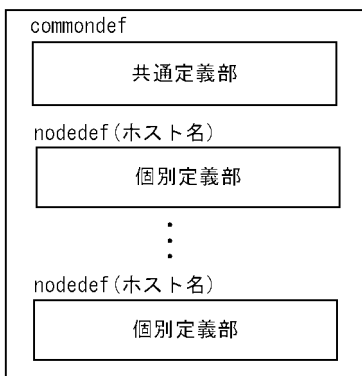
抽出側 HiRDB がパラレルサーバで、さらに HiRDB の環境変数（PDDIR, PDCONFPATH, SHLIB_PATH など）にサーバマシンごとに異なる指定をしている場合には、環境変数に対するオペランドをサーバマシンごとに個別に定義できます。サーバマシンごとの個別の定義を省略すると、そのサーバマシン下の抽出側 Datareplicator は、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定した環境変数に従います。サーバマシンごとの定義の形式と規則を次に示します。

抽出側 Datareplicator は定義ファイルの解析時に、HDEPATH 下に「定義ファイル名_tmp」という名称のワークファイルを、一時的に作成します。

(a) サーバマシンごとの定義形式

サーバマシンごとの定義形式を次の図に示します。

図 5-4 サーバマシンごとの定義形式



(b) サーバマシンごとの定義の規則

サーバマシンごとの定義の規則を次に説明します。

- commondef で定義を開始する共通定義部と nodedef（ホスト名）で定義を開始する個別定義部で構成してください。共通定義部、個別定義部に定義できるオペランドについては、「5.2 抽出システム定義」を参照してください。
- 抽出側 HiRDB がパラレルサーバで、さらに HiRDB の環境変数にサーバマシンごとに異なる指定をしている場合には、サーバマシンごとの環境変数に対するオペランドを個別定義部で定義してください。それ以外の場合は、個別定義部を省略できます。
- 共通定義部は定義ファイルの先頭に記述してください。
- commondef は省略できます。commondef を指定しても省略しても、定義ファイルの先頭から最初の nodedef の前までが、共通定義部とみなされます。nodedef（ホスト名）の指定がない場合は、すべ

てが共通定義部になります。ただし、commondef を省略しても、省略できないオペランドは共通定義部に指定する必要があります。

- 同じ nodedef (ホスト名) を指定した場合には、先に指定した nodedef (ホスト名) が有効になります。
- nodedef (ホスト名) に、抽出側 HiRDB の構成 (系切り替え構成では、現用系の構成) に含まれないホスト名を指定した場合には、定義解析時にエラーになります。
- commondef と nodedef (ホスト名) は、小文字の半角にしてください。
- nodedef (ホスト名) のホスト名には、抽出対象 HiRDB のシステム共通定義に指定されている pdunit 文、又は pdstart 文の -x オプションに指定されているホスト名のうち、個別定義の対象とするホスト名を指定してください。nodedef (ホスト名) を指定するときに、スペースを使わないでください。

(5) バックエンドサーバごとの定義 (抽出環境定義, 送信環境定義)

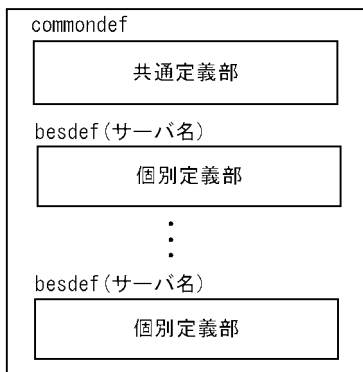
抽出側 HiRDB がパラレルサーバの場合には、抽出環境定義と送信環境定義でバックエンドサーバごとに環境を定義できます。バックエンドサーバごとの定義の形式と規則を次に示します。

抽出側 Datareplicator は、定義ファイルの解析時に HDEPATH 下に「定義ファイル名_tmp」という名称のワークファイルを、一時的に作成します。

(a) バックエンドサーバごとの定義の形式

バックエンドサーバごとの定義形式を次の図に示します。

図 5-5 バックエンドサーバごとの定義形式



(b) バックエンドサーバごとの定義の規則

バックエンドサーバごとの定義の規則を次に説明します。

- commondef で定義を開始する共通定義部と besdef (サーバ名) で定義を開始する個別定義部で構成してください。共通定義部では、各バックエンドサーバで共通の指定値にするオペランドを定義し、個別定義部では、バックエンドサーバごとに個別の指定値にするオペランドを定義してください。
- すべてのバックエンドサーバに対する定義を同じにする場合には、共通定義部だけを指定してください。
- 共通定義部は定義ファイルの先頭に記述してください。

- `commondef` は省略できます。`commondef` を指定しても省略しても、定義ファイルの先頭から最初の `besdef` (サーバ名) の前までが、共通定義部になります。`besdef` (サーバ名) の指定がない場合は、すべてが共通定義部になります。
- 共通定義部と個別定義部の両方で、すべてのオペランドを定義できます。共通定義部と個別定義部の両方で同じオペランドを定義した場合、個別定義部に対応するバックエンドサーバは、個別定義部の指定値に従います。個別定義部に対応しないバックエンドサーバは、共通定義部の指定値に従います。個別定義部で定義されていないオペランドが共通定義部で定義されている場合、個別定義部に対応するバックエンドサーバでは、共通定義部での指定値に従います。
- 同じ `besdef` (サーバ名) を指定した場合には、先に指定した `besdef` (サーバ名) が有効になります。
- `besdef` (サーバ名) に、抽出側 HiRDB の構成に含まれないサーバ名を指定した場合には、定義解析時にエラーになります。
- `commondef` と `besdef` (サーバ名) は、小文字の半角にしてください。
- `besdef` (サーバ名) のサーバ名には、定義の対象とするバックエンドサーバのサーバ名を指定してください。`besdef` (サーバ名) を指定するときに、スペースを使わないでください。

5.2 抽出システム定義

抽出システム定義では、抽出側 Datareplicator 全体の稼働環境についての情報を定義します。

5.2.1 形式

- 共通定義部でだけ定義する項目

set hdeid = 抽出側 Datareplicator 識別子

[set extdef = "抽出環境定義ファイル名"]

[set sendcontrol = nodemst | sendmst]

set sendid01 = 送信先識別子

[… [set sendid64 = 送信先識別子]] ※1

[set senddef01 = "送信環境定義ファイル名"]

[… [set senddef64 = "送信環境定義ファイル名"]] ※1

set sendid0001 = 送信先識別子

[… [set sendid4096 = 送信先識別子]]

[set senddef0001 = 送信環境定義ファイル名]

[… [set senddef4096 = 送信環境定義ファイル名]]

[set errfile_unique = true | false]

[set errfilesz = エラー情報ファイルの最大ファイルサイズ]

[set syslogout = true | false]

[set syslog_message_suppress = メッセージ番号 [, メッセージ番号] …]

[set dblocale = { sjis | euc | utf-8 }]

[set msglocale = { english | sjis-japanese | euc-japanese }]

[set watchintvl = エラー監視間隔]

[set cmwaittime = 通信待ち時間]

[set mstservice = マスターノードマスタ間通信サービス名] ※2

[set extinfo_num = 最大更新情報名数]

[set sync_term = true | false]

[set term_level = { normal | plan | both }]

[set info_message_out = nosuppress | suppress]

[set except_suppress = メッセージ番号 [, メッセージ番号] ...]

[set int_trc_lvl = 稼働トレース取得レベル [, 稼働トレース取得範囲]]

[set int_trc_filesz = 稼働トレースファイルのサイズ]

[set int_trc_unit = system | process] ※3

[set int_trc_output_path = 稼働トレースファイル出力先] ※3

[set int_trc_rintvl = 稼働トレース情報取得間隔]

[set sendprocnum = 送信プロセス最大起動数]

[set smt_sendintvl = 送信マスタプロセス送信間隔]

[set smt_sendintvl_scale = minute | second]

[set smt_editbufsize = 更新情報編集バッファ長]

[set smt_readbufnum = 更新情報読み込み用 I/O バッファ数]

[set smt_queue_read_wait_interval = 送信プロセスの抽出情報キューファイル読み込み間隔]

[set file_dupenv = 二重化定義ファイル名]

[set nodecontrol = unit | server]

[set node_connection_accept = true | false]

[set connection_accept_hostname = 抽出ノードマスタプロセス接続要求受け付けホスト名]

[set connection_accept_service = 抽出ノードマスタプロセス接続要求受け付けサービス名]

[set connection_accept_waittime = 抽出ノードマスタプロセス接続要求待ち時間]

[set connection_retry_time = 抽出ノードマスタプロセス再接続処理時間]

[set node_syslogout = true | false]

[set send_counter_reset = true | false]

[set hirdb_audit_trail = all | none]

[set recover_info_send = true | false]

[set recover_info_send_interval = 回復情報送信間隔]

[set cm_errno_check = true | false]

[set extwatch_intvl = 抽出処理監視間隔]

- 個別定義部でだけ定義する項目

[set node_pddir = 抽出側 HiRDB の PDDIR 環境変数の設定値]

[set node_pdconfpath = 抽出側 HiRDB の PDCONFPATH 環境変数の設定値]

[set node_shlibpath = 抽出側 HiRDB の SHLIB_PATH 環境変数の設定値]

[set node_host = ノードマスターホスト名]

注※1

Windows 版 Datareplicator のときは、送信先識別子と対応する送信環境定義ファイルは 63 個までしか指定できないため、sendid64 と senddef64 は指定しないでください。これらのオペランドを指定するとエラーになります。

注※2

Windows 版 Datareplicator のときは、マスターノードマスター間通信サービス名は「hdenmserv」固定で、Datareplicator のインストール完了後に services ファイルに指定します。そのため、mstservice オペランドは指定しないでください。このオペランドを指定しても無効になります。

注※3

Windows 版 Datareplicator のときにだけ指定できます。UNIX 版 Datareplicator のときに、このオペランドを指定するとエラーになります。

5.2.2 定義内容の変更

定義内容を変更する場合には、次のように操作してください。

1. 抽出側 Datareplicator を終了します。

初期開始をする場合は、抽出側 Datareplicator を正常終了させてください。抽出側 Datareplicator が異常終了している場合には、一度正常開始してから定義内容を変更してください。

2. テキストエディタで定義内容を変更します。

3. 次の表に示す、変更後の定義内容が有効になる開始モードで開始します。

表 5-3 抽出システム定義の変更後の定義内容が有効になる開始モード

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始※1	部分初期開始※1	正常開始※1
hdeid	○	—	—
extdef	○	—	—
sendcontrol	○	—	—
sendidxx※2	○	○	—
senddefxx※2	○	○	—
sendidxxx※3	○	○	—
senddefxxx※3	○	○	—
errfile_unique	○	—	—
errfilesz	○	○	○
syslogout	○	○	○
syslog_message_suppress	○	○	○
dblocale	○	—	—
msglocale	○	○	○
watchintvl	○	○	○
cmwaittime	○	○	○
mstservice	○	○	○
extinforum	○	—	—
syncterm	○	○	○
termlevel	○	○	○
info_message_out	○	○	○
except_suppress	○	○	○
int_trc_lvl	○	○	○
int_trc_filesz	○	○	○
int_trc_unit	○	○	○
int_trc_output_path	○	○	○
int_trc_rintvl	○	○	○
sendprocnum	○	○	○
smt_sendintvl	○	○	○

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始※1	部分初期開始※1	正常開始※1
smt_sendintvl_scale	○	○	○
smt_editbufsize	○	○	○
smt_readbufnum	○	○	○
smt_queue_read_wait_interval	○	○	○
file_dupenv	○	—	—
nodecontrol	○	—	—
node_connection_accept	○	○	○
connection_accept_hostname	○	○	○
connection_accept_service	○	○	○
connection_accept_waittime	○	○	○
connection_retry_time	○	○	○
node_syslogout	○	○	○
send_counter_reset	○	○	○
hirdb_audit_trail	○	○	○
recover_info_send	○	—	—
recover_info_send_interval	○	—	—
cm_errno_check	○	○	○
extwatch_intvl	○	○	○
node_pddir	○	—	—
node_pdconfpath	○	—	—
node_shlibpath	○	—	—
node_host	○	○	○

(凡例)

○：対応する動作によって変更内容が有効になることを示します。複数の動作に対して記載されている場合は、どれか一つの動作を実行することで変更内容が有効になることを示します。

—：該当しません。

注※1

抽出側 Datareplicator を起動するときの開始モードです。

注※2

xx には、01～64 が対応します。正常開始する場合は、sendid01～sendid64 オペランドと senddef01～senddef64 オペランドを変更しないでください。

注※3

xxxx には、0001～4096 が対応します。正常開始する場合は、sendid0001～sendid4096 オペランドと senddef0001～senddef4096 オペランドを変更しないでください。

5.2.3 オペランドの説明

- hdeid = 抽出側 Datareplicator 識別子

～ 〈16 進数字〉 ((00～FF))

個々の抽出側システムを識別するための識別子を指定します。抽出側 HiRDB がパラレルサーバの場合には、システムマネージャ下の抽出側 Datareplicator とすべてのバックエンドサーバ下の抽出側 Datareplicator を合わせて一つの抽出側システムになります。このため、抽出側 HiRDB がパラレルサーバの場合には、抽出側システムに一つだけ抽出側 Datareplicator 識別子が必要になります。

- extdef = "抽出環境定義ファイル名"

～ 〈1～64 バイトの [パス名/] ファイル名〉

《すべてのオペランドの指定を省略した抽出環境定義》

抽出処理の稼働環境を定義した抽出環境定義ファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDEPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが 125 バイト以下になるように指定してください。

extdef オペランドを省略すると、すべてのオペランドの指定を省略した抽出環境定義が仮定されます。

- sendcontrol = nodemst | sendmst

抽出データを送信するときの処理方式を指定します。

nodemst

各送信先ごとに送信プロセスを起動して、送信処理を送信先ごとに並列処理します。送信先は、sendidxx オペランドと senddefxx オペランドに指定します。送信先は、64 件まで指定できます。送信先の数が少なくトラフィックが高い業務でデータ連動するときには、nodemst を指定することをお勧めします。

sendmst

送信マスタプロセスで送信プロセス数を制御します。送信マスタプロセスは、sendprocnum オペランドに指定した数の送信プロセスを使って送信処理をスケジュールします。送信先は、sendidxxxx オペランドと senddefxxx オペランドに指定します。送信先は、4096 件まで指定できます。送信先の数が多くトラフィックが低い業務でデータ連動するときには、sendmst を指定することをお勧めします。

- sendid01 = 送信先識別子 [… [sendid64 = 送信先識別子]]

～ 〈1～8 文字の記号名称, 又は**〉

sendidxx には、更新情報の送信先を識別する名称（送信先識別子）を指定します。ここで指定する送信先識別子は、抽出定義の送信文やコマンド指定時に、処理対象の送信先システムを特定するために使います。また、メッセージの出力時にこのオペランドで指定する送信先識別子が表示されます。

sendcontrol オペランドに nodemst を指定又は sendcontrol オペランドを省略したときは、このオペランドを必ず指定してください。このオペランドの指定値は sendcontrol オペランドに nodemst を指定又は sendcontrol オペランドを省略した場合だけ有効です。

sendid01～sendid64 オペランドに指定する 01～64 の数字は、01 から昇順に連続して指定してください。昇順に連続して指定していない場合は、正しい順序で指定されているオペランドまでが有効になります。

送信先識別子に ** を指定すると、欠番として扱われ、この送信先識別子での送信は行いません。また、対応する senddefxx は無視されます。** は複数の送信先識別子に指定できますが、すべての sendidxx に ** を指定した場合は、送信先識別子が指定されていないものとして扱われます。

Windows 版 Datareplicator のときは、送信先識別子と対応する送信環境定義ファイルは 63 個までしか指定できないため、sendid64 は指定しないでください。このオペランドを指定するとエラーになります。

- senddef01 = "送信環境定義ファイル名" [… [senddef64 = "送信環境定義ファイル名"]]
～ 〈1～64 バイトの [パス名/] ファイル名〉

《すべてのオペランドの指定を省略した送信環境定義》

senddefxx には、送信処理の稼働環境を定義した送信環境定義ファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDEPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが 125 バイト以下になるように指定してください。このオペランドの指定値は sendcontrol オペランドに nodemst を指定又は sendcontrol オペランドを省略した場合だけ有効です。

ファイル名は、抽出側システムで一意になるようにしてください。

senddef01～senddef64 オペランドに指定する 01～64 の数字は、sendid01～sendid64 オペランドと対応する必要があります。

senddef01～senddef64 に対応する sendid01～sendid64 がない場合は、対応していない senddefxx は無視されます。sendid01～sendid64 に対応する senddef01～senddef64 がない場合には、すべてのオペランドを省略した送信環境定義が仮定されます。

Windows 版 Datareplicator のときは、送信先識別子と対応する送信環境定義ファイルは 63 個までしか指定できないため、senddef64 は指定しないでください。このオペランドを指定するとエラーになります。

- sendid0001 = 送信先識別子 [… [sendid4096 = 送信先識別子]]
～ 〈1～8 文字の記号名称〉

sendidxxxx には、更新情報の送信先を識別する名称（送信先識別子）を指定します。ここで指定する送信先識別子は、抽出定義の送信文やコマンド指定時に、処理対象の送信先システムを特定するために使います。また、メッセージの出力時にこのオペランドで指定する送信先識別子が表示されます。

sendcontrol オペランドに sendmst を指定したときは、このオペランドを必ず指定してください。このオペランドの指定値は sendcontrol オペランドに sendmst を指定した場合だけ有効です。

sendid0001～sendid4096 オペランドに指定する 0001～4096 の数字は、0001 から昇順に連続して指定してください。昇順に連続して指定していない場合は、正しい順序で指定されているオペランドまでが有効になります。

送信先識別子に**を指定すると、欠番として扱われ、この送信先識別子での送信は行いません。また、対応する senddefxxxx は無視されます。**は複数の送信先識別子に指定できますが、すべての sendidxxxx に**を指定した場合は、送信先識別子が指定されていないものとして扱われます。

- senddef0001 = "送信環境定義ファイル名" [… [senddef4096 = "送信環境定義ファイル名"]]
～ 〈1～125 バイトの [パス名/] ファイル名〉

《すべてのオペランドの指定を省略した送信環境定義》

senddefxxxx には、送信処理の稼働環境を定義した送信環境定義ファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDEPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが 125 バイト以下になるように指定してください。

このオペランドの指定値は sendcontrol オペランドに sendmst を指定した場合だけ有効です。

ファイル名は、抽出側システムで一意になるようにしてください。

senddef0001～senddef4096 オペランドに指定する 0001～4096 の数字は、sendid0001～sendid4096 オペランドと対応する必要があります。

senddef0001～senddef4096 に対応する sendid0001～sendid4096 がない場合は、対応していない senddefxxxx は無視されます。sendid0001～sendid4096 に対応する senddef0001～senddef4096 がない場合には、すべてのオペランドを省略した送信環境定義が仮定されます。

- errfile_unique = true | false

HiRDB/パラレルサーバで相互系切り替え構成のシステムに Datareplicator を使えるように、抽出ノードマスタエラー情報ファイルの名称を固有にするかどうかを指定します。

さらに、抽出ノードマスタトレースファイルの名称を固有にするかどうかを指定します。

true

抽出側 Datareplicator で固有な名称になるように、エラー情報ファイル名と稼働トレースファイルに各ノードのホスト名が付けられます。

- 抽出ノードマスタエラー情報ファイル名：errfile1_ホスト名, errfile2_ホスト名
- 抽出ノードマスタトレースファイル名：exttrc_ホスト名.trc1, exttrc_ホスト名.trc2

false

HiRDB/パラレルサーバのすべてのノードで同じ名称のエラー情報ファイル (errfile1, errfile2), 抽出ノードマスタトレースファイル (exttrc.trc1, exttrc.trc2) になります。

errfile_unique オペランドは、HiRDB/パラレルサーバで相互系切り替え構成の Datareplicator を使うときにだけ「true」を指定します。それ以外のときは、「false」を指定 又は errfile_unique オペランドを省略してかまいません。

- errfilesz = エラー情報ファイルの最大ファイルサイズ
～ 〈符号なし整数〉 ((1～2097151)) 《16》 (単位：KB)

抽出マスタエラー情報ファイルと抽出ノードマスタエラー情報ファイルの上限サイズを指定します。抽出マスタエラー情報ファイルと抽出ノードマスタエラー情報ファイルのサイズについては、「4.6.8 抽出側 Datareplicator のリソースの設計」を参照してください。

- `syslogout = true | false`

抽出マスタエラー情報ファイルに出力される情報を、`syslog` ファイルにも出力するかどうかを指定します。

true

抽出マスタエラー情報ファイルに出力される情報を、`syslog` ファイルに出力します。

false

抽出マスタエラー情報ファイルに出力される情報を、`syslog` ファイルに出力しません。

- `syslog_message_suppress = メッセージ番号 [, メッセージ番号] ...`

`syslog` ファイル (Windows の場合はイベントログ) への出力を抑止したいメッセージ番号を指定します。

- メッセージ番号は 64 個まで指定できます。
 - 同じメッセージ番号を重複して指定すると、1 件目のメッセージ番号だけを有効にして、2 件目以降の重複したメッセージ番号は無視されます。
 - このオペランドに指定したメッセージは、メッセージの重要度 (E, W, I, Q) が何であっても、出力が抑止されます。
 - `syslog` ファイル (又はイベントログ) にだけ出力されるメッセージは、このオペランドを指定しても出力されます。
- `dblocale = { sjis | euc | utf-8 }`

抽出側 Datareplicator が動作する文字コード体系を指定します。抽出定義解析でも、このオペランドで指定した文字コードで抽出定義が記述されていると認識し、抽出定義を解析します。

このオペランドに指定する文字コード体系は、抽出側 HiRDB の文字コード体系と同じにしてください。

sjis

JIS8/シフト JIS コード体系で動作します。

euc

EUC コード体系で動作します。

utf-8

utf-8 コード体系で動作します。

このオペランドの省略時仮定値は、OS によって異なります。次の表で確認してください。

OS	省略時仮定値
HP-UX 版	sjis
AIX 版	
Windows 版	

OS	省略時仮定値
Solaris 版	euc
Linux 版	

Windows 版 Datareplicator のときは、抽出側 Datareplicator が動作する文字コード体系は JIS8/シフト JIS コード体系と Unicode 体系だけなので、「sjis」又は「utf-8」のどちらかを指定します。「sjis」、「utf-8」以外の値を dblocale オペランドに指定するとエラーになります。

- `msglocale = { english | sjis-japanese | euc-japanese }`

抽出側 Datareplicator が出力するメッセージの文字コードを指定します。

Windows 版 Datareplicator の場合、`english` 又は `sjis-japanese` のどちらかを指定してください。`euc-japanese` を指定するとエラーになります。

`english`

英文のメッセージを出力します。

`sjis-japanese`

JIS8/シフト JIS コード体系で和文のメッセージを出力します。

`euc-japanese`

EUC コード体系で和文のメッセージを出力します。

- `watchintvl = エラー監視間隔`

～ 〈符号なし整数〉 ((10~32767)) 《60》 (単位：秒)

抽出マスタプロセスによって、抽出処理と送信処理のエラーを監視する間隔、又は抽出プロセスと送信プロセスの滞留監視の間隔を指定します。

- `cmwaittime = 通信待ち時間`

～ 〈符号なし整数〉 ((10~32767)) 《60》 (単位：秒)

抽出マスタプロセスと抽出ノードマスタプロセスとの間の通信待ち時間を指定します。

- `mstservice = マスターノードマスタ間通信サービス名`

～ 〈1~64 文字の識別子〉 《hdemaster》

抽出側 Datareplicator の抽出マスタプロセスと抽出ノードマスタプロセスとの間の通信エントリを `services` ファイルに追加するときに指定したサービス名を指定します。

Windows 版 Datareplicator の場合、マスターノードマスタ間通信サービス名は「`hdenmserv`」固定で、Datareplicator のインストール完了後に `services` ファイルに指定します。そのため、`mstservice` オペランドは指定しないでください。このオペランドを指定しても無効になります。

- `extinforum = 最大更新情報名数`

～ 〈符号なし整数〉 《50》 ((1~4096))

抽出定義で指定した更新情報名数を指定します。抽出定義で指定した更新情報名数よりも少ない値を指定した場合には、`hdeprep` コマンドの実行時にエラーになります。

- `syncterm = true | false`

抽出側 HiRDB の正常終了に連動して、抽出側 Datareplicator も自動的に正常終了するかどうかを指定します。

なお、系切り替え構成の場合は、系切り替えのタイミングと抽出側 Datareplicator の停止タイミングとの順序性が保証できないため、このオペランドに「true」を指定しても、すべての更新情報が反映側システムに送信されることは保証されません。したがって、系切り替え構成の場合は、このオペランドに「true」を指定しないでください。すべての更新情報が反映側システムに送信された後に計画的に系切り替えする場合は、hdestate コマンドで事前に確認するようにしてください。

true

抽出側 HiRDB の正常終了に連動して、抽出側 Datareplicator を自動的に正常終了します。次の条件をすべて満たす必要があります。

- ・抽出側 HiRDB の正常終了を検知しています。
- ・検知した正常終了の後に、抽出側 HiRDB でトランザクションが発生していません。
- ・抽出プロセスが正常に稼働し、システムログファイルのすべての更新情報の、抽出情報キューファイルへの抽出が完了しています。
- ・送信プロセスが正常に稼働し、抽出情報キューファイルのすべての更新情報の、反映側システムへの送信が完了しています。

false

抽出側 HiRDB が正常終了した場合でも、抽出側 Datareplicator は正常終了しません。

- ・ true を指定した場合、抽出側 Datareplicator は、システムログファイル中のすべてのシステムログを抽出して、抽出したシステムログ中の送信対象の更新情報を、すべて反映側システムへ送信した時点で終了します。

送信プロセスだけを起動していない場合、抽出プロセスが抽出側 HiRDB の正常終了を検知する前に送信プロセスを起動する必要があります。抽出プロセスが抽出側 HiRDB の正常終了を検知するまでに送信プロセスを起動しない場合、抽出プロセスが抽出側 HiRDB の正常終了を検知した時点で、抽出側 Datareplicator は正常終了します。このため、抽出情報キューファイルに更新情報が残ったままになることがあります。この場合には、次回 hdestart コマンド又は hdestart -s コマンドで送信プロセスを起動すると、抽出情報キューファイルの更新情報が、反映側システムに送信されます。

- ・ 抽出情報キューファイル回復機能を使用する場合は、false を指定してください。true を指定すると抽出情報キューファイル回復機能を使用できません。
- termlevel = { normal | plan | both }

HiRDB が停止したときに、抽出側 Datareplicator を自動的に同期停止させる場合、同期停止の対象とする HiRDB の停止方法を指定します。このオペランドは、syncterm オペランドに true を指定したときだけ有効です。

normal

HiRDB が正常停止したときに、抽出側 Datareplicator は正常終了します。

plan

HiRDB が計画停止したときに、抽出側 Datareplicator は正常終了します。

both

HiRDB が正常停止 又は計画停止したときに、抽出側 Datareplicator は正常終了します。

• info_message_out = nosuppress | suppress

情報通知レベルのメッセージを、syslog ファイル (Windows の場合はイベントログ)、及びエラー情報ファイルに出力しないようにするかどうかを指定します。

nosuppress

情報通知レベルのメッセージの出力を抑止しません。

suppress

情報通知レベルのメッセージの出力を抑止します。出力抑止の対象になるメッセージ番号は次のとおりです。

00551, 00552, 00553, 00554, 00555, 00557, 00581, 00583, 02031, 02032, 02033, 02034, 02036, 02037, 05001, 05002, 05008, 05012, 05013, 05018

• except_suppress = メッセージ番号 [, メッセージ番号] ...

～ 〈5 けたの符号なし整数〉

info_message_out オペランドに suppress を指定して出力を抑止したメッセージのうち、抑止対象から除外して syslog ファイルやエラー情報ファイルに出力したいメッセージ番号を指定します。メッセージ番号は 63 個まで指定できます。

出力抑止の対象でないメッセージ番号を指定しても、無効になります (特に意味を持ちません)。同じメッセージ番号を重複して指定すると、1 件目のメッセージ番号だけを有効にして、2 件目以降の重複したメッセージ番号は無視されます。

このオペランドは、info_message_out オペランドに suppress を指定したときだけ有効です。

• int_trc_lvl = 稼働トレース取得レベル [, 稼働トレース取得範囲]

稼働トレースファイルで採取する項目を変更するときに、次に示す表の値を指定します。このオペランドを省略すると、すべての機能の共通情報 (最小限必要な情報) だけを取得します。

稼働トレース取得レベルに na を指定すると、稼働トレースは取得されません。さらに、稼働トレース取得範囲に指定した値は無視されます (ただし、構文エラーや指定範囲のエラーはチェックされます)。

• 稼働トレース取得レベルに指定する値

指定する値	取得する情報		
	共通情報	性能概要	性能詳細
p1	○	○	×
p2	○	○	○
na	×	×	×
int_trc_lvl オペランドを省略	○	×	×

(凡例)

○ : 採取します。

× : 採取しません。

共通情報：開始・終了情報，エラー情報，プロセスレベルでの変局点を示す大まかなチェックポイント情報の総称。

注意

p2 を指定したときは，稼働トレースファイルのサイズを大きく設定しておかないと，すぐにラップアラウンドして必要な情報が消えてしまうことがあります。また，トレース採取のためのオーバーヘッドが大きくなることもあります。

• 稼働トレース取得範囲に指定する値

指定する値	取得する情報		
	MST (制御機能)	CAP (抽出機能)	SND (送信機能)
c1	○	○	×
c2	○	×	○
c3	○	×	×
nc	○	×*	×*
int_trc_lvl オペランドを省略 又は稼働トレース取得範囲を省略	○	○	○

(凡例)

- ：採取します。
- ×：採取しません。

注※

抽出システム定義で nc を指定したときは，抽出環境定義と送信環境定義の int_trc_getv オペランドの指定で稼働トレースを個別に取得できます。

int_trc_lvl オペランドの指定の目安

int_trc_lvl オペランドは，次のように指定することをお勧めします。

1. 本番運用のとき

本番運用では，int_trc_lvl オペランドを省略することをお勧めします。

この指定で性能が出ないときは，一時的に稼働トレースのオペランドを変更して性能状況を見てください。このとき int_trc_lvl オペランドの第 1 パラメタに「p1」又は「p2」，第 2 パラメタに「nc」を指定してから，特定の（性能が出ていない）機能に int_trc_getl 又は int_trc_getv オペランドを指定します。int_trc_lvl オペランドの第 1 パラメタに「p2」を指定したときは，int_trc_filesz オペランドには余裕を持った値（1MB 以上を推奨）を指定してください。

2. テスト運用のとき

テストの段階では，int_trc_lvl オペランドの第 1 パラメタに「p1」を指定して，第 2 パラメタを省略することをお勧めします。こうしておけば，性能の問題が起こったときに hdstrcredit コマンドを実行していつでも稼働トレースを参照できます。コマンドの実行結果を参照して，HiRDB の表定義や反映グループ数をチューニングしてください。

より詳細な情報 (SQL の実行単価など) が必要なときは、int_trc_lvl オペランドの第 2 パラメータを「p2」に変更してください。「p2」を指定したときは、int_trc_filesz オペランドには余裕を持った値 (1MB 以上を推奨) を指定してください。

- int_trc_filesz = 稼働トレースファイルのサイズ

～ 〈符号なし整数〉 ((32~2097120)) 《128》 (単位: KB)

稼働トレースファイル 1 個当たりの最大格納サイズを指定します。

int_trc_lvl オペランドに na を指定すると、このオペランドを指定しても無視されます。

このオペランドに指定した値を 32KB の単位で切り上げた値が、ファイル 1 個当たりの最大容量になります。そのため、指定する値は 32 の倍数にすることをお勧めします。

稼働トレースファイルは、2 個のファイルを相互にスワップ/ラップしながら使います。

- int_trc_unit オペランドに system を指定、または省略した場合

抽出マスタトレースファイル:

\$HDEPATH/msttrc.trc1 と \$HDEPATH/msttrc.trc2

抽出ノードマスタトレースファイル:

\$HDEPATH/exttrc.trc1 と \$HDEPATH/exttrc.trc2

errfile_unique オペランドに true を指定すると、ファイル名に「_ホスト名」が付きます (「exttrc_ホスト名.trc1」と「exttrc_ホスト名.trc2」になります)。

- int_trc_unit オペランドに process を指定した場合

表「抽出側 Datareplicator の稼働トレースファイル名」を参照してください。

- int_trc_rintvl = 稼働トレース情報取得間隔

～ 〈符号なし整数〉 ((5~30000)) 《50》 (単位: ミリ秒)

稼働トレース情報を取得する間隔を指定します。

int_trc_lvl オペランドに na を指定すると、このオペランドを指定しても無視されます。

稼働トレース情報取得間隔を小さくすると、欠落する稼働トレース情報は減らすことができますが、一秒当たりの監視回数が増加するため、CPU 利用率は高くなります。

次の場合は、稼働トレース情報取得間隔を小さくしてください。

- 稼働トレース情報に欠落が発生している場合
- 稼働トレース取得レベルで、取得情報の多いレベルが指定されている場合
- Datareplicator の稼働頻度が高い (抽出データが多い) 場合

また、次の場合は、稼働トレース情報取得間隔を大きくしてください。

- Datareplicator の稼働頻度が低く (抽出データが少なく)、Datareplicator による CPU 利用率を削減したい場合

- int_trc_unit = system | process

稼働トレース収集方式を指定します。このオペランドは、Windows 版 Datareplicator の場合だけ指定できます。

system を指定すると複数プロセスから同時に 1 つのファイルに書き込みを行うため、排他制御を行います。したがって、複数プロセスからの稼働トレース出力が集中すると、排他制御がボトルネックとなり、Datareplicator 全体の処理性能が劣化することがあります。抽出システム定義の int_trc_lvl オペランドに p2 を指定し、かつ、送信プロセスが複数ある環境では、送信プロセス数に比例して性能劣化の度合いが大きくなる傾向があります。process を指定すると、プロセスごとに稼働トレースファイルを割り当てるため、排他制御が不要となり、性能への影響を抑えることができます。process を指定すると、稼働トレース収集プロセスは起動しません。

system

抽出側 Datareplicator の抽出マスタープロセスの稼働トレースを抽出マスタートレースファイルに、抽出マスタープロセス以外の稼働トレースを抽出ノードマスタートレースファイルに出力します（全プロセス一括収集方式）。

process

抽出側 Datareplicator の各プロセスの稼働トレースを別々の稼働トレースファイルに出力します（プロセス単位収集方式）。

process を指定した場合の稼働トレースファイル名を、表「抽出側 Datareplicator の稼働トレースファイル名」に示します。稼働トレースファイルの出力先ディレクトリは、int_trc_output_path オペランドに指定します。

表 5-4 抽出側 Datareplicator の稼働トレースファイル名

プロセス名	稼働トレースファイル名
抽出マスタープロセス	hdemaster.trc1 hdemaster.trc2
抽出ノードマスタープロセス	<ul style="list-style-type: none"> 抽出システム定義の nodecontrol オペランドに unit を指定した場合 hdenodemst_ホスト名.trc1 hdenodemst_ホスト名.trc2 (例：hdenodemst_host1.trc1) 抽出システム定義の nodecontrol オペランドに server を指定した場合 hdenodemst_サーバ名.trc1 hdenodemst_サーバ名.trc2 (例：hdenodemst_BES1.trc1)
抽出プロセス	hdecapture_サーバ名.trc1 hdecapture_サーバ名.trc2 (例：hdecapture_BES1.trc1)
送信プロセス*	<ul style="list-style-type: none"> 抽出システム定義の sendcontrol オペランドに nodemst を指定した場合 hdesender_サーバ名_送信先識別子.trc1 hdesender_サーバ名_送信先識別子.trc2 (例：hdesender_BES1_send01.trc1)

プロセス名	稼働トレースファイル名
	<ul style="list-style-type: none"> 抽出システム定義の sendcontrol オペランドに sendmst を指定した場合 hdesndprc_サーバ名_送信先識別子.trc1 hdesndprc_サーバ名_送信先識別子.trc2 (例：hdesndprc_BES1_send01.trc1)
送信マスタプロセス	<ul style="list-style-type: none"> 抽出システム定義の sendcontrol オペランドに sendmst を指定した場合だけ作成 hdesendmst_サーバ名.trc1 hdesendmst_サーバ名.trc2 (例：hdesendmst_BES1.trc1)

注※

送信先識別子に¥が含まれる場合、¥を@に置き換えます。

• **int_trc_output_path = 稼働トレースファイル出力先**

プロセス単位収集方式の稼働トレースファイルの出力先ディレクトリを絶対パス名または相対パス名で指定します。このオペランドは、Windows 版 Datareplicator の場合だけ指定できます。相対パス名で指定した場合は、%HDEPATH%からの相対パスです。指定できる長さは、絶対パス名で 125 バイトまでとなります。125 バイトを超えた場合、KFRB00916-E メッセージを出力し、起動処理を停止します。

このオペランドは、int_trc_unit オペランドに process を指定した場合だけ有効です。

このオペランドで指定した稼働トレースファイル出力先ディレクトリは、Datareplicator 起動までにユーザが作成してください。その際、Datareplicator 管理者には read 権限と write 権限を付与してください。Datareplicator 起動時に指定したディレクトリがない場合、KFRB00865-E メッセージを出力し、起動処理を中止します。

このオペランドを省略すると、%HDEPATH%/trace を仮定します。該当するディレクトリがない場合、抽出マスタプロセス、抽出ノードマスタプロセス起動時に Datareplicator がディレクトリを作成します。

稼働トレースファイルの合計最大サイズは次の式のとおりとなります。稼働トレースファイル出力先には、この値以上の空き容量のあるディレクトリを指定してください。

$$\text{int_trc_filesz オペランド指定値} \times ((\text{サーバ数}^{\ast} \times (\text{送信先識別子数} + 3)) + 1) \times 2$$

注※

サーバ数は、抽出対象の HiRDB がシングルサーバの場合は 1、パラレルサーバの場合はバックエンドサーバの数となります。

• **sendprocnum = 送信プロセス最大起動数**

UNIX 版 Datareplicator の場合 ～((1~4096)) 《1》

Windows 版 Datareplicator の場合 ～((1~63)) 《1》

送信マスタプロセスが起動する送信プロセスの最大起動数を指定します。

このオペランドは、sendcontrol オペランドに sendmst を指定した場合に有効です。

- **smt_sendintvl = 送信マスタプロセス送信間隔**

～((0~1440)) 《5》

送信マスタプロセスの送信間隔を指定します。

送信間隔の単位は `smt_sendintvl_scale` オペランドで指定します。

このオペランドは、`sendcontrol` オペランドに `sendmst` を指定した場合に有効です。

0 を指定すると、トランザクション単位で更新情報を送信します。ただし、反映までの時間を短くしたい場合は、1 秒からチューニングを始めてください。0 を指定すると送信処理が遅くなる場合があります。

`sendcontrol` オペランドに `sendmst` を指定すると、送信環境定義の `sendintvl` オペランドの指定値は無視されて、このオペランドの指定値に従って送信処理が実行されます。

- **smt_sendintvl_scale = minute | second**

`smt_sendintvl` オペランドの指定値の単位を指定します。

minute

`smt_sendintvl` オペランドの指定値の単位を分にします。

second

`smt_sendintvl` オペランドの指定値の単位を秒にします。

このオペランドは、`smt_sendintvl` オペランドを指定した場合に有効です。

- **smt_editbufsize = 更新情報編集バッファ長**

～((1~2097151)) 《300》 (単位 : KB)

送信マスタプロセスを使った送信プロセスの制御中に、送信プロセスが更新情報を反映側システムで受信できる形式に編集するためのバッファ長を指定します。

このオペランドは、`sendcontrol` オペランドに `sendmst` を指定した場合に有効です。`sendcontrol` オペランドに `sendmst` を指定すると、送信環境定義の `editbufsize` オペランドの指定値は無視されて、このオペランドの指定値に従って送信処理が実行されます。なお、このオペランドの指定値を大きくすると通信回数を削減できます。`smt_sendintvl` オペランドで指定した時間内に抽出情報キューファイルに書き込まれるデータ量 (更新情報量) より大きい値を指定することを推奨します。

抽出情報キューファイルに書き込まれるデータ量を計算する場合は、`hdestate` コマンドを定期的に行い、Queue write position が示すオフセット情報の増加量を計測してください。

- **smt_readbufnum = 更新情報読み込み用 I/O バッファ数**

～((1~255)) 《1》

送信マスタプロセスを使った送信プロセスの制御中に、送信マスタプロセスが抽出情報キューファイルから更新情報を読み込むときの I/O バッファ数を指定します。

このオペランドは、`sendcontrol` オペランドに `sendmst` を指定した場合に有効です。`sendcontrol` オペランドに `sendmst` を指定すると、送信環境定義の `readbufnum` オペランドの指定値は無視されて、このオペランドの指定値に従って送信処理が実行されます。

- **smt_queue_read_wait_interval = 送信プロセスの抽出情報キューファイル読み込み間隔**

～ 〈符号なし整数〉 ((100~60000)) 《2000》 (単位 : ミリ秒)

送信マスタプロセスによる送信制御時に、抽出情報キューファイルの終端を検知してから次の読み込みを再開するまでの間隔を指定します。

このオペランドの値を小さくすれば、抽出情報キューファイルの終端を検知した後の待ち時間が短くなるので、送信処理の即時性を向上させることができます。

ただし、値を小さくすると CPU の利用率が高くなるため、逆に性能が悪くなる可能性があります。このため、更新処理の発生頻度や CPU の性能を考慮した上で値を指定する必要があります。

このオペランドは、sendcontrol オペランドに sendmst を指定した場合に有効です。sendcontrol オペランドに sendmst を指定すると、送信環境定義の queue_read_wait_interval オペランドの指定値は無視されて、このオペランドの指定値に従って送信処理が実行されます。

- **file_dupenv = 二重化定義ファイル名**
 ~ 〈1~125 バイトのファイル名〉

二重化定義ファイル名を絶対パス名、又は相対パス名で指定します。相対パス指定の場合は、抽出側 Datareplicator 運用ディレクトリからの相対パスとします。

このオペランドを省略した場合、二重化機能を使用しないものとして動作します。

- **nodecontrol = unit | server**

抽出ノードマスタプロセスの制御方式を指定します。

unit

抽出ノードマスタプロセスを抽出対象 HiRDB のユニット単位に起動します。

server

抽出ノードマスタプロセスを抽出対象 HiRDB のバックエンドサーバ単位に起動します。

抽出対象 HiRDB が影響分散スタンバイレス型系切り替え構成の場合、このオペランドの指定を server とする必要があります。

また、hdestart コマンドで抽出マスタプロセスを起動する場合は、抽出ノードマスタプロセスの起動ホスト情報を抽出対象 HiRDB から取得するため、抽出対象 HiRDB が稼働している必要があります (hdestart_n コマンドで起動する場合は、抽出対象 HiRDB は稼働している必要はありません)。

このオペランドの指定内容によって名称、及び作成単位が変更となるリソースを次に示します。

リソース種別	nodecontrol=unit 指定		nodecontrol=server 指定※	
	名称	作成単位	名称	作成単位
抽出ノードマスタエラー情報ファイル	errfile1, errfile2	ノード単位に 2 個	errfile1_サーバ名, errfile2_サーバ名	サーバ単位に 2 個
抽出ノードマスタ稼働トレースファイル	exttrc.trc1, exttrc.trc2		exttrc_サーバ 名.trc1, exttrc_ サーバ名.trc2	

注※

抽出システム定義の errfile_unique オペランドの指定は無効とし、上表の名称でリソースを作成します。

- **node_connection_accept = true | false**

抽出ノードマスタプロセスから抽出マスタプロセスへの回線接続をするかどうかを指定します。抽出ノードマスタプロセスは、次に示す契機で抽出マスタプロセスへの回線接続を行います。

- 抽出ノードマスタプロセスを `hdestart_n` コマンドで起動した場合
- 抽出マスタプロセスと抽出ノードマスタプロセス間の回線切断後、抽出ノードマスタプロセスが回線接続リトライをする場合

これらの各契機について、`node_connection_accept` オペランドを指定することによって抽出マスタプロセス、抽出ノードマスタプロセスの動作を制御できます。

`node_connection_accept` オペランドを指定による抽出マスタプロセス、抽出ノードマスタプロセスの動作制御を次に示します。

事象	true 指定	false 指定
抽出ノードマスタプロセスを <code>hdestart_n</code> コマンドで起動	<ul style="list-style-type: none"> • 抽出マスタプロセスの動作 抽出ノードマスタプロセスからの接続要求を受け付けます。 • 抽出ノードマスタプロセスの動作 抽出マスタプロセスと接続した後、レプリケーションを開始します。 	<ul style="list-style-type: none"> • 抽出マスタプロセスの動作 抽出ノードマスタプロセスからの接続要求を受け付けません。 • 抽出ノードマスタプロセスの動作 <code>hdestart_n</code> コマンドで指定した接続待ち時間に到達した時点でタイムアウトによってプロセスを終了します。
抽出マスタプロセスと抽出ノードマスタプロセス間の回線が切断	<ul style="list-style-type: none"> • 抽出マスタプロセスの動作 すべての抽出ノードマスタプロセスとの回線が切断した場合、抽出ノードマスタプロセスからの接続要求待ちを行います。 • 抽出ノードマスタプロセスの動作 抽出マスタプロセスに対して回線の再接続を行います。 	<ul style="list-style-type: none"> • 抽出マスタプロセスの動作 すべての抽出ノードマスタプロセスとの回線が切断した場合、プロセスを終了します。 • 抽出ノードマスタプロセスの動作 回線切断検知後、プロセスを終了します。

`node_connection_accept` オペランドの指定値は、`nodecontrol` オペランドに `server` を指定した場合だけ有効となります。抽出対象 HiRDB が影響分散スタンバイレス型系切り替え構成の場合、このオペランドの指定を `true` とする必要があります。

また、`node_connection_accept` オペランドに `true` を指定した場合、`connection_accept_hostname` オペランド及び `connection_accept_service` オペランドの指定が必要となります。

- `connection_accept_hostname` = 抽出ノードマスタプロセス接続要求受け付けホスト名
～ 〈1～32 文字の識別子〉

抽出ノードマスタプロセスから抽出マスタプロセスへの回線接続をする場合に、抽出マスタプロセスが抽出ノードマスタプロセスからの回線接続要求を受け付けるためのホスト名を指定します。

`node_connection_accept` オペランドに `true` を指定した場合、このオペランドの指定が必要となります。

影響分散スタンバイレス型系切り替え構成の場合は、系切り替え発生後に待機系で起動した抽出マスタプロセスに対して、現用系と同じホスト名で抽出ノードマスタプロセスが接続要求を発行するため、IP アドレス引き継ぎありのホスト名をこのオペランドで指定しておく必要があります。

また、抽出ノードマスタプロセスを `hdestart_n` コマンドで起動する運用の場合は、`-x` オプションで指定する抽出マスタプロセス稼働ホスト名に、このオペランドで指定したホスト名を指定する必要があります。

なお、このオペランドに指定するホスト名は、OS の `hosts` ファイルに追加しておく必要があります。

- **connection_accept_service = 抽出ノードマスタプロセス接続要求受け付けサービス名**

～ 〈1～64 文字の識別子〉

抽出ノードマスタプロセスから抽出マスタプロセスへの回線接続をする場合に、抽出マスタプロセスが抽出ノードマスタプロセスからの回線接続要求を受け付けるためのサービス名を指定します。

`node_connection_accept` オペランドに `true` を指定した場合は、このオペランドの指定が必要となります。

このオペランドに指定するサービス名（及びポート番号）は、抽出システム定義の `mstservice` オペランド及び送信環境定義の `hdeservice` オペランドに指定したサービス名（及びポート番号）とは異なるものを指定します。

抽出ノードマスタプロセスを `hdestart_n` コマンドで起動する運用の場合は、`-n` オプションで指定するサービス名に、このオペランドで指定したサービス名を指定する必要があります。

なお、このオペランドに指定するサービス名は、OS の `services` ファイルに追加しておく必要があります。

- **connection_accept_waittime = 抽出ノードマスタプロセス接続要求待ち時間**

～((1～3600)) 《300》 (単位：秒)

すべての抽出ノードマスタプロセスとの回線が切断した場合、抽出マスタプロセスが抽出ノードマスタプロセスからの再接続要求を受け付けるための待ち時間を指定します。

どの抽出ノードマスタプロセスからも接続要求がないまま、このオペランドに指定した待ち時間が経過した場合は、抽出マスタプロセスが停止します。

このオペランドは、`node_connection_accept` オペランドで `true` を指定した場合にだけ有効です。

- **connection_retry_time = 抽出ノードマスタプロセス再接続処理時間**

～((1～3600)) 《300》 (単位：秒)

抽出マスタプロセスとの回線が切断した場合、抽出ノードマスタプロセスが抽出マスタプロセスに対して回線の再接続処理をする時間を指定します。

抽出マスタプロセスと回線の再接続ができないまま、このオペランドに指定した待ち時間が経過した場合は、抽出ノードマスタプロセスが停止します。

このオペランドは、`node_connection_accept` オペランドで `true` を指定した場合にだけ有効です。

- **node_syslogout = true | false**

抽出ノードマスタエラー情報ファイルに出力される情報を、該当するマシンの `syslog` ファイルにも出力するかどうかを指定します。

true

抽出ノードマスタエラー情報ファイルに出力される情報を、`syslog` ファイルに出力します。

false

抽出ノードマスタエラー情報ファイルに出力される情報を、syslog ファイルに出力しません。

なお、Windows 版については、ノード単位エラー情報の出力有無は syslogout オペランドの指定によって決定するため、このオペランドはサポートしていません。このオペランドを指定しても無視されます。

注意

このオペランドに「true」を指定した場合、ノード単位エラー情報をすべて出力します。稼働環境によっては情報が大量に出力されて、syslog ファイルのリソースを多く消費する場合がありますので、「true」を指定しないことをお勧めします。

- `send_counter_reset = true | false`

抽出側システム起動時に送信データ件数をリセットするかどうかを指定します。

true

抽出側システム起動時に送信データ件数をリセットします。

false

抽出側システム起動時に送信データ件数をリセットしません。

- `hirdb_audit_trail = all | none`

監査証跡の取得対象になる操作（監査対象イベント）が発生した場合に、監査証跡を取得するかどうかを指定します。

監査対象イベントとなるのは、hdeevent コマンド、hdeprep コマンド、抽出マスタプロセスです。

all

監査証跡を取得します。

none

監査証跡を取得しません。

抽出側 Datareplicator の処理による監査証跡取得のオーバーヘッドを抑止したい場合は、none を指定することをお勧めします。

- `recover_info_send = true | false`

反映側 Datareplicator に回復情報を送信するかどうかを指定します。

true

回復情報を送信します。

false

回復情報を送信しません。

注意

回復情報を送信しないと、システムログファイルによるデータ連動回復を実行できないため、「true」を指定することをお勧めします。システムログファイルによるデータ連動回復については、「[9.5 システムログファイルによるデータ連動回復](#)」を参照してください。

- `recover_info_send_interval = 回復情報送信間隔`

～((1～32767)) 《1》

反映側 Datareplicator に回復情報を送信する間隔を、抽出側システムでのトランザクション数で指定します。このオペランドは recover_info_send オペランドに「true」を指定した場合だけ有効となります。ただし、イベント送信時はこのオペランドの指定に関係なく、回復情報を送信します。

指定値の目安については、「9.5.2(3)(a) 回復情報ファイルの作成」を参照してください。

- **cm_errno_check = true | false**

通信障害を検知した際、障害内容をチェックして、再接続の可能性のある障害の場合だけ接続リトライするか、障害内容に関係なく無条件に接続リトライするかを指定します。

true

再接続の可能性のある障害の場合だけ接続リトライします。

ただし、ネットワークの構成や機器の組み合わせによっては、一時的な障害の場合でも接続リトライの対象とならない場合があります。再接続できない障害の場合は、送信プロセスをエラー停止します。

false

障害内容に関係なく、無条件に接続リトライします。

注意

- ・「false」を指定した場合、「true」指定時に、抽出側 Datareplicator がチェックする障害内容についても、無条件に接続リトライします。このため、「true」指定時に早期に検出できていた通信障害が、送信環境定義の retrynum オペランドに指定した回数の接続リトライを終えるまで検出されません。通信障害を早期に検知するために、接続リトライ回数（送信環境定義の retrynum オペランドで指定）を無限にしないことをお勧めします。

- ・「false」を指定した場合、再接続の可能性のある障害以外で接続リトライするとき、KFRB02053-W メッセージが出力されます（回線切断理由は other）。また、付加情報としてシステムコール種別、及び errno を、抽出ノードマスタエラー情報ファイルに出力します。

- **extwatch_intvl = 抽出処理監視間隔**

～((0, 2～60)) 《0》

抽出側 Datareplicator 稼働中に抽出処理が停止していないかどうかを監視する間隔を分単位で指定します。

0 を指定、又はこのオペランドを省略すると、抽出処理を監視しません。

2～60 の値を指定すると、指定した時間間隔で抽出処理を監視します。抽出処理が停止していることを検知した場合、KFRB05044-W メッセージを出力します。

なお、抽出処理の監視は hdestart -s コマンドや hdestop -e コマンドによって意図的に抽出処理を停止している場合にも行われます。

- **node_pddir = 抽出側 HiRDB の PDDIR 環境変数の設定値**

～ 〈1～255 バイトのパス名〉

抽出側 HiRDB がパラレルサーバで、さらに環境変数 PDDIR に対してサーバマシンごとに異なる設定をしている場合に、サーバマシンごとに環境変数 PDDIR に対して設定した、抽出側 HiRDB の運用ディレクトリを絶対パス名で指定します。

このオペランドは個別定義部でだけ定義してください。

node_pddir オペランドの指定を省略すると、そのサーバマシン下の抽出側 Datareplicator は、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定した環境変数 PDDIR に従います。

- **node_pdconfpath = 抽出側 HiRDB の PDCONFPATH 環境変数の設定値**

～ 〈1～255 バイトのパス名〉

抽出側 HiRDB がパラレルサーバで、さらに環境変数 PDCONFPATH に対してサーバマシンごとに異なる設定をしている場合に、サーバマシンごとに環境変数 PDCONFPATH に対して設定した、抽出側 HiRDB のシステム定義ファイルを格納するディレクトリを絶対パス名で指定します。

このオペランドは個別定義部でだけ定義してください。

node_pdconfpath オペランドの指定を省略すると、そのサーバマシン下の抽出側 Datareplicator は、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定した環境変数 PDCONFPATH に従います。

- **node_shlibpath = 抽出側 HiRDB の SHLIB_PATH 環境変数の設定値**

～ 〈1～255 バイトのパス名〉

抽出側 HiRDB がパラレルサーバで、さらに環境変数 SHLIB_PATH に対してサーバマシンごとに異なる設定をしている場合に、サーバマシンごとに環境変数 SHLIB_PATH に対して設定した、\$PDDIR/lib を絶対パス名で指定します。

このオペランドは個別定義部でだけ定義してください。

node_shlibpath オペランドの指定を省略すると、そのサーバマシン下の抽出側 Datareplicator は、抽出側 Datareplicator のコマンドを実行するユーザの環境に対して設定した環境変数 SHLIB_PATH に従います。

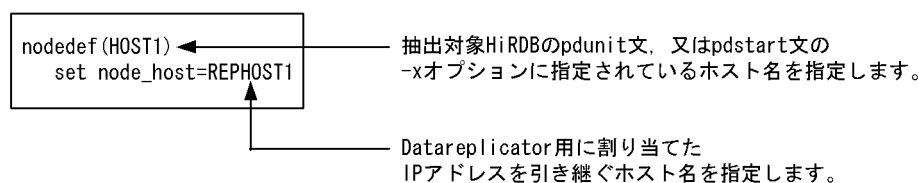
- **node_host = ノードマスタホスト名**

～ 〈1～32 文字の英数字〉

系切り替え構成で次の条件を共に満たさない場合は、Datareplicator 用に IP アドレスを引き継ぐホスト名を割り当て、このオペランドに割り当てたホスト名を指定する必要があります。

- 抽出対象 HiRDB のシステム共通定義の pdunit 文又は pdstart 文の -x オプションに指定されているホスト名が、IP アドレスを引き継ぐホスト名である場合
- HA モニタのリソースサーバ機能を使用し、抽出対象 HiRDB 及び抽出側 Datareplicator の停止完了後に上記 IP アドレスの切り替えが行われるように設定している場合

オペランドの指定例を次に示します。



5.3.2 定義内容の変更

定義内容を変更する場合には、次のように操作してください。抽出側 Datareplicator が異常終了している場合には、一度正常開始してから定義内容を変更してください。

1. 抽出側 Datareplicator を正常終了します。
2. テキストエディタで定義内容を変更します。
3. 次の表に示す、変更後の定義内容が有効になる開始モードで開始します。

表 5-5 抽出環境定義の変更後の定義内容が有効になる開始モード

オペランド名	変更後の定義内容が有効になる開始モード	
	初期開始 ^{※1}	正常開始 ^{※1}
dsid	○	—
qufile ^{※2}	○	—
queuesize	○	—
logiosize	○	○
quiosize	○	—
extsuppress	○	—
ext_wait_interval	○	○
extract_level	○	○
int_trc_getv	○	○
qufullwarn	○	○
extract_delay_limit_time	○	○
devicex ^{※3}	○	—

(凡例)

○：対応する動作によって変更内容が有効になることを示します。複数の動作に対して記載されている場合は、どれか一つの動作を実行することで変更内容が有効になることを示します。

—：該当しません。

注※1

抽出側 Datareplicator を起動するときの開始モードです。

注※2

xxx には、001～016 が対応します。

注※3

xx には、01～18 が対応します。

5.3.3 オペランドの説明

- dsid = データ連動識別子

～ 〈16 進数字〉 ((00～FF)) 《00》

反映側システムに対して、更新情報の抽出元を識別させるためのデータ連動識別子を指定します。指定するデータ連動識別子は、対応する反映側システム内で一意にしてください。

- qufile001 = ”抽出情報キューファイル名”

～ 〈1～64 バイトの [パス名/] ファイル名〉 《\$HDEPATH/qufile001》

qufile002 = ”抽出情報キューファイル名”

～ 〈1～64 バイトの [パス名/] ファイル名〉 《\$HDEPATH/qufile002》

:

qufile016 = ”抽出情報キューファイル名”

～ 〈1～64 バイトの [パス名/] ファイル名〉

qufilexxx には、抽出情報キューファイル名を絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDEPATH/相対パス名が絶対パス名に仮定されます。抽出側 Datareplicator では、指定した名称に「_サーバ名」を付けた名称のファイルを作成します。UNIX 版の場合、ファイルの種別がキャラクタ型スペシャルファイルのときには、指定した名称に「_サーバ名」を付けた名称のファイルを作成する必要があります。

絶対パス名で指定する場合には、「_サーバ名」を含んだファイル名の長さが 64 バイト以下になるようにしてください。相対パス名で指定する場合には、仮定される絶対パス名に「_サーバ名」を含んだファイル名の長さが 125 バイト以下になるようにしてください。

ファイル名は、抽出側システムで一意にしてください。抽出側 HiRDB がパラレルサーバの場合には、各バックエンドサーバで一意にしてください。

抽出情報キューファイルはファイル単位のスワップ方式で情報が格納されるため、2 個以上必要で、16 個まで指定できます。qufile001～qufile016 をすべて省略すると、qufile001 には \$HDEPATH/qufile001 が、qufile002 には \$HDEPATH/qufile002 が仮定されます。

qufile001～qufile016 に指定する 001～016 は、001 から昇順に連続して指定してください。昇順に連続して指定していない場合には、正しい順序で指定されている抽出情報キューファイルまでを更新情報の格納の対象にします。ここで指定した順番に、抽出情報キューファイルはスワップされます。

- queuesize = 抽出情報キューファイルサイズ

～ 〈符号なし整数〉 ((33～1000000000)) 《65》 (単位 : KB)

qufile001～qufile016 オペランドで定義した抽出情報キューファイルのサイズを指定します。queuesize オペランドで指定したファイルサイズは、qufile001～qufile016 オペランドで指定したすべてのファイルの、一つ当たりのサイズになります。

抽出情報キューファイルの見積もり式については、「[4.6.8 抽出側 Datareplicator のリソースの設計](#)」を参照してください。

また、抽出情報キューファイルをラージファイルとして運用する場合、2097152 (2GB) 以上に指定します。ラージファイルとして運用する場合は、OS や Datareplicator ファイルシステム領域を、ラー

ジファイルに対応するように事前に設定しておく必要があります。詳細については、「[6.11 ラージファイルの運用](#)」を参照してください。

注意

既存の抽出情報キューファイルのサイズを変更した場合、抽出側 Datareplicator を初期化する必要があるため、レプリケーションが完了していることを事前に確認してから、ファイルサイズを変更してください。

- **logiosize = システムログ I/O バッファサイズ**

～ 〈符号なし整数〉 ((32~510)) 《510》 (単位 : KB)

抽出側 HiRDB のシステムログファイルから更新情報を読み込むときに使う、システムログ I/O バッファのサイズを指定します。実際のバッファサイズは、このオペランドを使用した次の式で算出されず。

$$4 \times \uparrow \text{logiosize} \div 4 \uparrow$$

システムログ I/O バッファのサイズについては、「[4.6.3 抽出方法の設計](#)」を参照してください。

- **quiosize = 抽出情報キュー I/O バッファサイズ**

～ 〈符号なし整数〉 ((32~510)) 《32》 (単位 : KB)

抽出情報キューファイルに更新情報を格納するとき、又は抽出情報キューファイルから更新情報を読み込むときに使う、抽出情報キュー I/O バッファのサイズを指定します。更新情報キュー I/O バッファのサイズについては、「[4.6.3 抽出方法の設計](#)」を参照してください。

- **extsuppress = true | false**

抽出対象表がないサーバに対して抽出側 Datareplicator を動作させるかどうかを指定します。

extsuppress オペランドは、抽出側 HiRDB がパラレルサーバの場合にだけ指定してください。抽出対象表がないサーバの負荷を軽減したい場合には、true を指定します。サーバごとに抽出対象表があるかどうかを意識したくない場合には、false を指定します。

extsuppress の指定と抽出側 HiRDB 及び抽出側 Datareplicator での処理については、「[6.5.4 抽出環境定義の extsuppress の指定と、抽出側 HiRDB 及び抽出側 Datareplicator での処理](#)」を参照してください。

true

true を指定したサーバに対しては、抽出側 Datareplicator は動作しません。このため、true を指定したサーバの表が抽出定義で定義されていても、更新情報は抽出されません。

false

false を指定したサーバに対しては、抽出側 Datareplicator は動作します。このため false を指定したサーバの表が抽出定義で定義されている場合だけ、更新情報が抽出されます。false を指定したサーバの表が抽出定義で定義されていない場合には、更新情報は抽出されません。

false を指定時に抽出側 Datareplicator を起動しない状態で、抽出側 HiRDB だけで HiRDB Datareplicator 連携機能を実行し続けると、抽出対象表がないサーバでも、システムログファイルが満杯になる可能性があります。

- **ext_wait_interval = 抽出終端検知後の抽出再開始間隔**

～ 〈符号なし整数〉 ((100~60000)) 《5000》 (単位 : ミリ秒)

抽出対象 DBMS からの更新情報の終端を検知してから次回の抽出を再開するまでの間隔を指定します。このオペランドの値を小さくすれば、システムログファイルの終端を検知した後の待ち時間が短くなるので、抽出処理の即時性を向上させることができます。

ただし、値を小さくすると CPU の利用率が高くなるため、逆に性能が悪くなる可能性があります。このため、更新処理の発生頻度や CPU の性能を考慮した上で値を指定する必要があります。

- `extract_level = current_gen | original_gen | all_gen`

抽出対象の HiRDB がインナレプリカ機能を使っているときの抽出レベルを指定します。

`current_gen`

抽出対象 HiRDB がインナレプリカ機能を使っているときに、カレント RD エリアに対する更新だけを抽出対象にします。

`original_gen`

抽出対象 HiRDB がインナレプリカ機能を使っているときに、オリジナル RD エリアに対する更新だけを抽出対象にします。

`all_gen`

抽出対象 HiRDB がインナレプリカ機能を使っているときに、すべてのレプリカ RD エリア（オリジナル RD エリアを含む）に対する更新を抽出対象にします。

- `int_trc_getv = true | false`

このノード上（共通定義部 `commondef` のとき）又はバックエンドサーバ上（個別定義部 `besdef` のとき）の抽出関連プロセスについての稼働トレースを取得するかどうかを指定します。

`true`

抽出関連プロセスごとの稼働トレースを取得します。

`false`

抽出関連プロセスごとの稼働トレースを取得しません。

このオペランドは、抽出システム定義の `int_trc_lvl` オペランドの第 2 パラメタに `nc` を指定したときだけ有効になります。

- `qufullwarn = 抽出情報キューファイル満杯警告値`

～ 〈符号なし整数〉 ((1~8)) (単位：ファイル数)

抽出情報キューファイルが満杯になる前に警告メッセージを出力するときに指定します。空いている抽出情報キューファイルの数がここで指定した値以下になった場合、抽出ノードマスタエラー情報ファイルに `KFRB05031-W` メッセージが、抽出マスタエラー情報ファイルに `KFRB00701-E` (メッセージ番号=05031) が出力されます。

この指定値が抽出情報キューファイル数以上の場合、抽出情報キューファイルがスワップするごとに、警告メッセージが出力されます。

- `extract_delay_limit_time = 抽出滞留時間しきい値`

～ 〈符号なし整数〉 ((0~86400)) 《0》 (単位：秒)

更新情報がシステムログファイルに格納された時刻と、抽出情報キューファイルに抽出情報が書き込まれる時刻との差（抽出滞留時間）のしきい値を指定します。抽出滞留時間が、ここで指定したしきい値よりも大きくなった場合、警告メッセージが出力されます。

抽出環境定義の `ext_wait_interval` オペランドの指定値も、滞留時間に考慮する必要があります。なお、0 を指定した場合、抽出プロセスでの滞留監視機能は無効になります。

注意

このオペランドは、レプリケーション対象となる更新量や、送信間隔、反映間隔などを考慮した上で、値を調整してください。初めは、10 分（600 秒）を目安に設定することをお勧めします。

- `device01 = Datareplicator` ファイルシステム領域名 `[[, 割り当てファイル種別] ...]`
 ~ 〈1~125 バイトの絶対パス名〉
- `device02 = Datareplicator` ファイルシステム領域名 `[[, 割り当てファイル種別] ...]`
 ~ 〈1~125 バイトの絶対パス名〉
- ：
- `device18 = Datareplicator` ファイルシステム領域名 `[[, 割り当てファイル種別] ...]`
 ~ 〈1~125 バイトの絶対パス名〉

`devicexx` オペランドには、Datareplicator ファイルシステム領域とその領域に指定した割り当てファイル種別を指定します。2 けたの数字部分 (`xx`) は、必ず通し番号にしてください。割り当てファイル種別とファイルの対応を次に示します。

割り当てファイル種別	各オペランドに指定するファイル名
<code>qufilexxx</code> (<code>xxx</code> は 001~016 までの整数)	<code>set qufilexxx</code> で指定した抽出情報キューファイル用リンクファイル名。作成するリンクファイルには、サーバ名が付与されます。
<code>hde_file</code>	該当するバックエンドサーバのデータ連動用連絡ファイル用リンクファイル名
<code>sts_file</code>	該当するバックエンドサーバのステータスファイル用リンクファイル名

割り当てファイル種別を省略すると、該当するバックエンドサーバのすべてのファイルを割り当て対象にします。

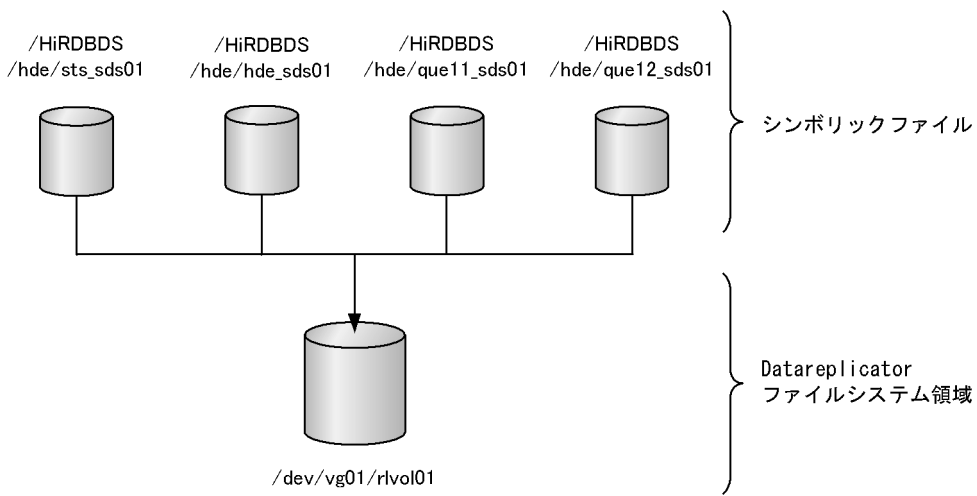
`devicexx` オペランドで Datareplicator ファイルシステム領域として割り当てたファイルは、自動的に Datareplicator ファイルシステムにリンクするファイルとして作成されます。割り当てファイル種別に対応するファイルの名称は、既にキャラクタ型スペシャルファイルとして割り当てられていないようにしてください。

`devicexx` オペランドは個別定義部でだけ指定できます。共通定義部で指定すると抽出定義の解析でエラーになって、以降の処理が中断されます。抽出対象 HiRDB がシングルサーバ構成の場合は、個別定義部の開始 (`besdef`) で指定するサーバ名にシングルサーバ名を指定してください。

`devicexx` オペランドの定義例を次に示します。

● 定義例1：シングルサーバの場合

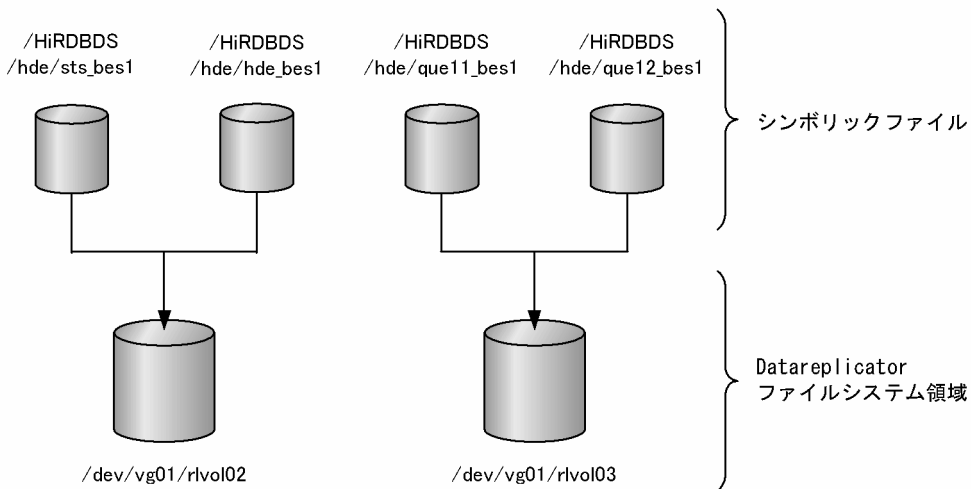
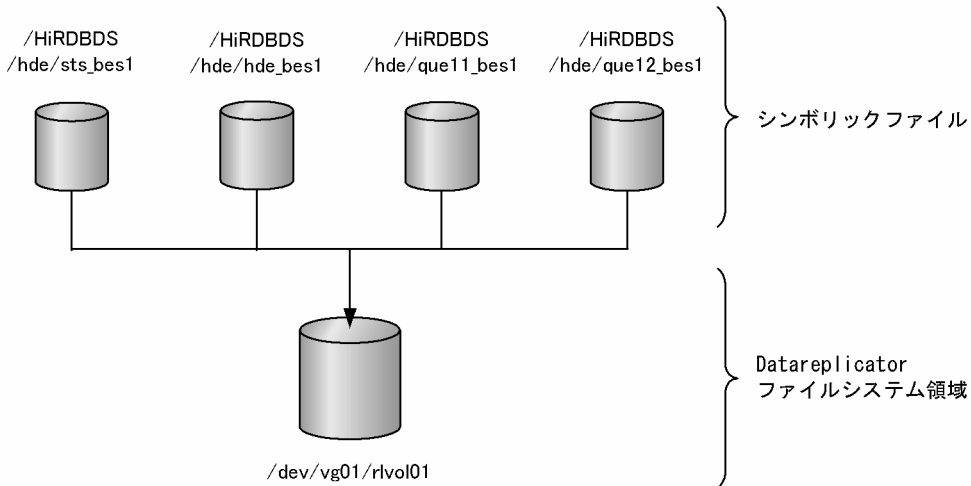
```
set qufile001 = /HiRDBDS/hde/que11
set qufile002 = /HiRDBDS/hde/que12
set device01  = /dev/vg01/rlvol01
```



● 定義例2：パラレルサーバの場合

```

besdef (bes1)
set qufile001 = /HiRDBDS/hde/que11
set qufile002 = /HiRDBDS/hde/que12
set device01 = /dev/vg01/rlvol01
besdef (bes2)
set qufile001 = /HiRDBDS/hde/que11
set qufile002 = /HiRDBDS/hde/que12
set device01 = /dev/vg02/rlvol02,hde_file,sts_file
set device02 = /dev/vg03/rlvol03,qufile001,qufile002
    
```



注意

devicexx オペランドを指定して Datareplicator を運用した後で、devicexx オペランドを定義から省略するときは、次に示す手順で作業を実施してください。

1. Datareplicator を停止します。
2. devicexx オペランドを抽出環境定義から削除します。

3. 削除した `devicexx` オペランドに指定していた Datareplicator ファイルシステム領域を、`hdsfmkfs` コマンドで初期化します。

4. Datareplicator を初期開始します。

上記の手順を踏まないで `devicexx` オペランドを定義から削除して Datareplicator を初期開始すると、実行時にファイル不正でエラーになることがあります。

5.4 送信環境定義

送信処理を稼働するために必要な情報を定義します。送信環境定義ファイルは送信先識別子ごとに作成する必要があります。抽出側 HiRDB がパラレルサーバの場合には、一つの送信環境定義ファイルでバックエンドサーバごとに送信環境を定義できます。

5.4.1 形式

[set sendhdsid =相手システム識別子]

[set hdeservice =サービス名]

[set hdehost =送信先ホスト名]

[set protocol = tcp | osi] ※

[set extract_tselector = T セレクタ] ※

[set nsap_address = NSAP アドレス] ※

[set senduoc = use | nouse]

[set sendintvl =送信間隔]

[set sendintvl_scale = minute | second]

[set nsndid001 =送信抑止受信元識別子
[… [set nsndid256 =送信抑止受信元識別子]]]

[set keepalive = true | false]

[set retrynum =コネクションリトライ回数]

[set retry_interval =コネクションリトライ時のリトライ間隔]

[set overwrite = true | false]

[set overwrite_continue = true | false]

[set maxtran =同時実行最大トランザクション数]

[set maxtrandata =トランザクション内最大更新情報数]

[set readbufnum =送信用の抽出情報キュー I/O バッファ数]

[set editbufsize =更新情報編集バッファサイズ]

[set prg_eventno = イベント番号]

[set int_trc_getv = true | false]

[set queue_read_wait_interval = 送信プロセスの抽出情報キューファイル読み込み間隔]

[set recvwatchtime = データ受信用回線監視時間]

[set send_delay_limit_time = 送信滞留時間しきい値]

[set reflect_mode = server | uap]

[set eventsync = 同期イベントコード]

[set eventcntreset = 送信データ件数リセットイベントコード]

注※

OSI プロトコルで通信を実行する場合に指定します。ただし、このオペランドは HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

5.4.2 定義内容の変更

定義内容を変更する場合には、次のように操作してください。

1. 抽出側 Datareplicator を終了します。

初期開始をする場合は、抽出側 Datareplicator を正常終了させてください。抽出側 Datareplicator が異常終了している場合には、一度正常開始してから定義内容を変更してください。

2. テキストエディタで定義内容を変更します。

3. 次の表に示す、変更後の定義内容が有効になる開始モードで開始します。

表 5-6 送信環境定義の変更後の定義内容が有効になる開始モード

オペランド名	変更後の定義内容が有効になる開始モード	
	初期開始又は部分初期開始※1	正常開始※1
sendhdsid	○	—
hdeservice	○	—
hdehost	○	○
protocol	○	—
extract_tselector	○	—
nsap_address	○	—
senduoc	○	○

オペランド名	変更後の定義内容が有効になる開始モード	
	初期開始又は部分初期開始※1	正常開始※1
sendintvl	○	○
sendintvl_scale	○	○
nsndidxxx※2	○	—
keepalive	○	○
retrynum	○	○
retry_interval	○	○
overwrite	○	○
overwrite_continue	○	○
maxtran	○	○
maxtrandata	○	○
readbufnum	○	○
editbufsize	○	○
prg_eventno	○	○
int_trc_getv	○	○
queue_read_wait_interval	○	○
recvwatchtime	○	○
send_delay_limit_time	○	○
reflect_mode	○	—
eventsync	○	—
eventcntreset	○	○

(凡例)

○：対応する動作によって変更内容が有効になることを示します。複数の動作に対して記載されている場合は、どれか一つの動作を実行することで変更内容が有効になることを示します。

—：該当しません。

注※1

抽出側 Datareplicator を起動するときの開始モードです。

注※2

xxx には、001～256 が対応します。

5.4.3 オペランドの説明

- sendhdsid = 相手システム識別子

～ 〈16 進数字〉 ((00～FF)) 《00》

データ連動の相手システムの識別子を指定します。

相手システムが反映側 Datareplicator のときは、反映側 Datareplicator 識別子を指定します。

相手システムが XDM/DS のときは、反映 XDM/DS 識別子を EBCDIK で 16 進変換した値を指定します。

- hdeservice = サービス名

～ 〈1～64 文字の識別子〉 《hirdbds》

抽出側 Datareplicator の通信エントリを services ファイルに追加するときに指定したサービス名を指定します。

運用例

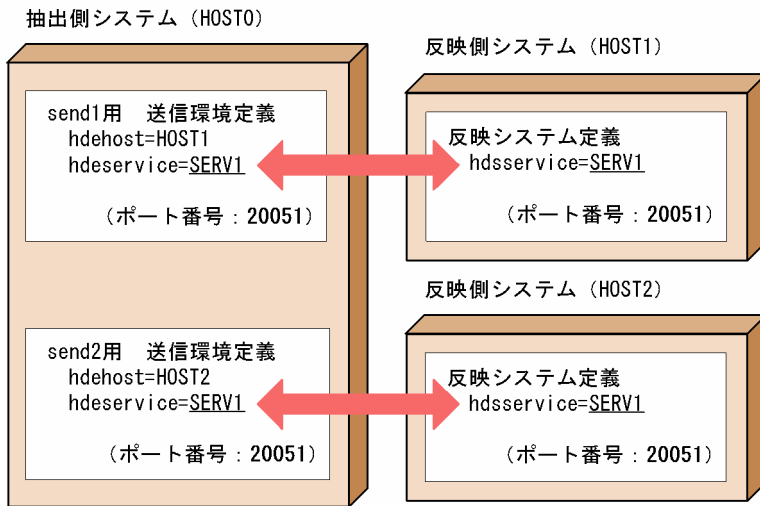
反映側システムが 2 台のサーバマシンで構築されている場合について説明します。

この場合、それぞれのサーバマシンに対して、送信環境定義を定義してください。サービス名には、抽出側システムの通信環境の設定時に指定したサービス名を指定します。このとき、ポート番号が反映側のサーバマシンと同じ番号であることを確認してください。抽出側システムと反映側システムの間にはファイアウォールを設置している場合、抽出側から反映側への通信方向に対して、このオペランドに指定したポート番号を通過する指定を行ってください。

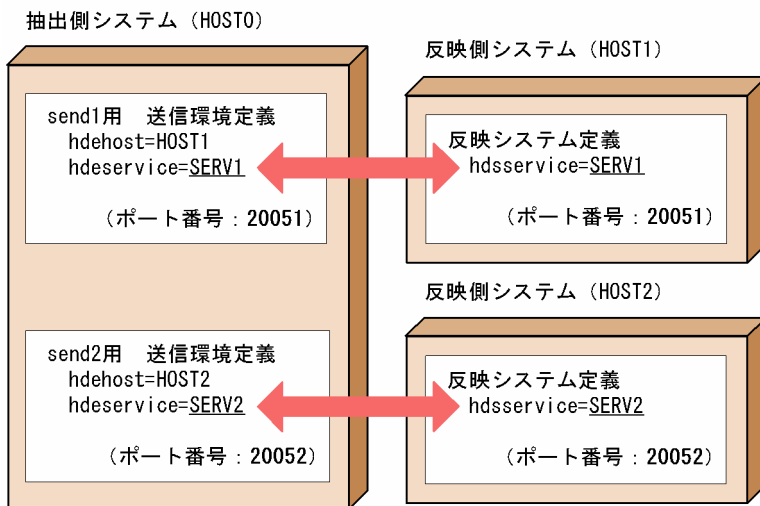
なお、反映側のサーバマシンの送信先ホスト名がそれぞれ異なる場合、サービス名が同じ名称でも問題ありません。

反映側システムが 2 台のサーバマシンで構築されている場合の、送信環境定義の指定例を次の図に示します。

●ポート番号が反映側のサーバマシンと同じ場合



●ポート番号が反映側のサーバマシンと異なる場合



• hdehost = 送信先ホスト名

～ 〈1～255 文字の識別子〉《hdehost》

hosts ファイル又は DNS に対して接続先エントリを追加するときに指定したホスト名を指定します。ホスト名は、先頭が英字で始まる英数字、又は「-」、「_」、「.」を指定してください。

• protocol = tcp | osi

反映側システムとのプロトコルを指定します。

tcp

TCP/IP プロトコルで通信を実行します。

osi

OSI プロトコルで通信を実行します。ただし、抽出システム定義の sendcontrol に sendmst を指定した場合は、通信プロトコルに OSI は指定できません。

このオペランドは、HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

• extract_tselector = T セレクタ

～ 〈2～64 けたで偶数けたの 16 進数字〉《00》

接続先の反映側 Datareplicator の受信プロセスと同じ T セレクタを指定します。また、OSI プロトコルで通信を実行する場合に指定します。ただし、このオペランドは HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

- `nsap_address = NSAP アドレス`

～ 〈2～40 けたで偶数けたの 16 進数字〉《00》

接続先の反映側 Datareplicator が動作するマシンの NSAP アドレスを指定します。また、OSI プロトコルで通信を実行する場合に指定します。ただし、このオペランドは HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

- `senduoc = use | nouse`

送信データ UOC を使うかどうかを指定します。

use

送信データ UOC を使います。

UNIX 版 Datareplicator で `SHLIB_PATH` の下に `libsenduoc.sl` がない場合は、`nouse` を仮定して動作します。Windows 版 Datareplicator で `PATH` の下に `senduoc.dll` がない場合は、DLL のローディングエラーになって送信が停止します。

nouse

送信データ UOC を使いません。

- `sendintvl = 送信間隔`

～ 〈符号なし整数〉 ((0～1440)) 《5》

抽出した更新情報を反映側システムに送信するときの送信間隔を指定します。

送信間隔の単位は `sendintvl_scale` オペランドで指定します。

0 を指定すると、トランザクション単位で更新情報を送信します。ただし、反映までの時間を短くしたい場合は、1 秒からチューニングを始めてください。0 を指定すると送信処理が遅くなる場合があります。

WITHOUT ROLLBACK オプションを指定している表をデータ連動する場合

WITHOUT ROLLBACK オプションを指定している表は、1 回の更新を 1 トランザクションとして扱うため、このオペランドに 0 を指定すると、WITHOUT ROLLBACK オプションを指定している表を更新するたびに送信処理が発生して、送信処理のオーバーヘッドが大きくなります。このため、WITHOUT ROLLBACK オプションを指定している表をデータ連動する場合は、このオペランドに 0 以外を指定することをお勧めします。

- `sendintvl_scale = minute | second`

`sendintvl` オペランドの指定値の単位を指定します。

minute

`sendintvl` オペランドの指定値の単位を分にします。

second

`sendintvl` オペランドの指定値の単位を秒にします。

このオペランドは、sendintvl オペランドを指定した場合に有効です。

個別定義部でこのオペランドを指定する場合は、同じ besdef 内で再度 sendintvl オペランドを指定する必要があります。sendintvl オペランドを省略すると、このオペランドは無視されます。

sendintvl オペランド及び sendintvl_scale オペランドの組み合わせと、有効となる値との関係を次に示します。

個別定義部の指定		共通定義部の指定		
		sendintvl 指定あり		sendintvl 指定なし (sendintvl_scale は無視される)
		sendintvl_scale 指定あり	sendintvl_scale 指定なし	
sendintvl 指定あり	sendintvl_scale 指定あり	時間：個別定義部の指定が有効となる 単位：個別定義部の指定が有効となる	時間：個別定義部の指定が有効となる 単位：個別定義部の指定が有効となる	時間：個別定義部の指定が有効となる 単位：個別定義部の指定が有効となる
	sendintvl_scale 指定なし	時間：個別定義部の指定が有効となる 単位：分が仮定される	時間：個別定義部の指定が有効となる 単位：分が仮定される	時間：個別定義部の指定が有効となる 単位：分が仮定される
sendintvl 指定なし (sendintvl_scale は無視される)		時間：共通定義部の指定が有効となる 単位：共通定義部の指定が有効となる	時間：共通定義部の指定が有効となる 単位：分が仮定される	時間：5 が仮定される 単位：分が仮定される

- **nsndid001 = 送信抑止受信元識別子 [… [nsndid256 = 送信抑止受信元識別子]]**
 ~ 〈16 進数字〉 ((00~FF))

nsndidxxx には、更新情報の送信を抑止する受信元の識別子を指定します。一つのシステムに抽出側と反映側の両方が存在するシステムを複数組み合わせさせたデータ連動システムで、ループバックを発生させないために指定します。

送信抑止受信元識別子には、この抽出側 HiRDB 下で稼働している反映側 Datareplicator が受信した更新情報の、受信元のシステム（この反映側システムへ送信した抽出側システム）での抽出側 Datareplicator 識別子を指定します。ループバックの抑止については、「[4.5 データ連動システムの形態の設計](#)」を参照してください。

nsndid001~nsndid256 オペランドに指定する 001~256 の数字は、001 から昇順に連続して指定してください。昇順に連続して指定していない場合は、正しい順序で指定されているオペランドまでが有効になります。

nsndid オペランドの指定がない場合には、sendhdsid オペランドで指定した相手システムに対して、該当する更新情報をすべて送信します。

- **keepalive = true | false**

ソケットにキープアライブオプションを設定するかどうかを指定します。

true

キープアライブオプションを設定します。

false

キーブアライブオプションを設定しません。

- **retrynum = コネクションリトライ回数**

～ 〈符号なし整数〉 ((0~256)) 《0》

送信処理で反映側システムとの通信接続に失敗したときのリトライ回数を指定します。送信処理では、通信接続に失敗した反映側システムに対して、指定した回数まで再接続を試みます。0を指定した場合、通信接続に成功するまで、又は停止要求を受け付けるまで、再接続を試みます。

再接続の間隔は、sendcontrol オペランドの指定値によって異なります。再接続の間隔を次の表に示します。

sendcontrol オペランドの指定値	再接続の間隔
nodemst	コネクションリトライ時のリトライ間隔 (retry_interval オペランドの指定値)
sendmst	送信間隔 (smt_sendintvl オペランドの指定値)

- **retry_interval = コネクションリトライ時のリトライ間隔**

～ 〈符号なし整数〉 ((1~60)) 《60》 (単位: 秒)

反映側システムで系切り替えが発生した場合、早急に再接続したいときコネクションリトライ時のリトライ間隔を指定します。このオペランドは sendcontrol オペランドに nodemst を指定した場合だけ有効です。

また、反映情報キューファイルが満杯になり接続が切断された場合も、このオペランドに指定した間隔でコネクションリトライします。このため、極端に小さい値を指定した場合、反映情報キューファイルが満杯になると、反映情報キューファイルが満杯であることを知らせるメッセージが、リトライ間隔ごとに出力されるので注意してください。

- **overwrite = true | false**

抽出情報キューファイルの満杯時に、sendhdsid で指定した相手システムに対する送信処理を縮退して、抽出情報キューファイルの満杯を回避するかどうかを指定します。

送信環境定義の reflect_mode オペランドに uap を指定している場合は、false を指定してください。false 以外を指定すると、KFRB00847-E メッセージが出力されます。

true

相手システムに対する送信処理を、縮退の対象にします。true 指定時、抽出情報キューファイルの満杯が発生した時点で、該当する送信処理は中止され、抽出情報キューファイルから送信対象の更新情報は破棄されます。この場合、該当する送信処理の送信先の反映側システムでは、抽出側 DB と反映側 DB との不整合が発生するため、反映側 DB の再作成が必要です。

false

相手システムに対する送信処理を、縮退の対象にしません。

- **overwrite_continue = true | false**

overwrite オペランドで指定した縮退状態を、次の送信処理再開始時に引き継ぐかどうかを指定します。

true

縮退状態で送信処理を再開始した場合、送信プロセス起動エラーとすることで、縮退状態を継続します。

false

縮退状態で送信処理を再開始した場合、縮退状態を解除して、再開始以降に抽出したトランザクションから送信処理を再開始します。

- **maxtran = 同時実行最大トランザクション数**

～ 〈符号なし整数〉 ((5~17800000)) 《100》

更新情報の編集、送信時に使うトランザクション管理情報バッファの初期値を算出するために、同時に実行されるトランザクション数を指定します。トランザクション管理情報バッファの初期値は、maxtran オペランドで指定する同時実行最大トランザクション数と maxtrandata オペランドで指定するトランザクション内最大更新情報数の積で内部的に算出します。maxtran オペランドの指定値と maxtrandata の指定値の積が 89000000 以内になるように指定してください。

トランザクション管理情報バッファのサイズについては、「[4.6.4 送信方法の設計](#)」を参照してください。

- **maxtrandata = トランザクション内最大更新情報数**

～ 〈符号なし整数〉 ((5~17800000)) 《500》

更新情報の編集、送信時に使うトランザクション管理情報バッファの初期値を算出するために、一つのトランザクション内で発生する更新情報の最大数を指定します。ただし、抽出定義の extract 文に ukey 句を指定したときは、更新前データを格納する更新情報数を考慮して、2 倍にして計算してください。

トランザクション管理情報バッファの初期値は、maxtran オペランドで指定する同時実行最大トランザクション数と maxtrandata オペランドで指定するトランザクション内最大更新情報数の積で内部的に算出します。maxtran オペランドの指定値と maxtrandata の指定値の積が 89000000 以内になるように指定してください。トランザクション管理情報バッファのサイズについては、「[4.6.4 送信方法の設計](#)」を参照してください。

- **readbufnum = 送信用の抽出情報キュー I/O バッファ数**

～ 〈符号なし整数〉 ((1~255)) 《1》

更新情報の送信時に、更新情報を抽出情報キューファイルから読み込むときに使う、送信用の抽出情報キュー I/O バッファの数を指定します。送信用の抽出情報キュー I/O バッファのサイズは、抽出環境定義の quiosize で指定します。送信用の抽出情報キュー I/O バッファについては、「[4.6.4 送信方法の設計](#)」を参照してください。

- **editbufsize = 更新情報編集バッファサイズ**

～ 〈符号なし整数〉 ((1~2097151)) 《300》 (単位 : KB)

抽出情報キューファイルから読み出した更新情報を、反映側システムが受信できる送信形式に編集するために使う更新情報編集バッファのサイズを指定します。

更新情報編集バッファのサイズについては、「[4.6.4 送信方法の設計](#)」を参照してください。

なお、このオペランドの指定値を大きくすると通信回数を削減できます。sendintvl オペランドで指定した時間内に抽出情報キューファイルに書き込まれるデータ量（更新情報量）より大きい値を指定することを推奨します。

抽出情報キューファイルに書き込まれるデータ量を計算する場合は、hdestate コマンドを定期的に行い、Queue write position が示すオフセット情報の増加量を計測してください。

- **prg_eventno = イベント番号**

～ 〈符号なし整数〉 ((0~255))

送信プロセスが複数の BES にわたる分割表に対する PURGE TABLE の更新情報を検知したときに、反映側システムに送信するイベントのイベント番号を指定します。指定するイベント番号によって、送信プロセスの動作が異なります。イベント番号と送信プロセスの動作を次に示します。

指定するイベント番号	送信プロセスの動作
指定なし	該当する PURGE TABLE を無視して、送信処理を続けます。
0	反映側システムに正常停止ログを送信した後、送信処理を終了します（送信プロセス停止）。
上記以外	反映側システムに prg_eventno オペランドに指定した番号のイベントを送信します。イベントを送信した後は、送信処理を続けます。

- **int_trc_getv = true | false**

このノード上（共通定義部 commondef のとき）又はバックエンドサーバ上（個別定義部 besdef のとき）の送信プロセスについての稼働トレースを取得するかどうかを指定します。

true

送信プロセスごとの稼働トレースを取得します。

false

送信プロセスごとの稼働トレースを取得しません。

このオペランドは、抽出システム定義の int_trc_lvl オペランドの第 2 パラメタに nc を指定したときだけ有効になります。

- **queue_read_wait_interval = 送信プロセスの抽出情報キューファイル読み込み間隔**

～ 〈符号なし整数〉 ((100~60000)) 《2000》 (単位：ミリ秒)

送信プロセスによる送信制御時に、抽出情報キューファイルの終端を検知してから次の読み込みを再開するまでの間隔を指定します。

このオペランドの値を小さくすれば、抽出情報キューファイルの終端を検知した後の待ち時間が短くなるので、送信処理の即時性を向上させることができます。

ただし、値を小さくすると CPU の利用率が高くなるため、逆に性能が悪くなる可能性があります。このため、更新処理の発生頻度や CPU の性能を考慮した上で値を指定する必要があります。

- **recvwatchtime = データ受信用回線監視時間**

～ 〈符号なし整数〉 ((1~35791394)) 《10》 (単位：分)

抽出-反映間の回線監視時間を指定します。このオペランドに指定した時間内に応答がない場合、回線をクローズし、接続確立をリトライします。

このオペランドの指定値を小さくすると、回線障害を早期に検出できるようになります。しかし、指定値を小さくし過ぎると次のような場合に、不要に回線の切断及び再接続を行うことになるため、設定値を調節する必要があります。

- 反映システムの負荷が高く、送信データを反映情報キューファイルに格納するのに時間が掛かる場合
 - 送信環境定義内の editbufsize オペランドの指定値が大きく、反映情報キューファイルに対する 1 回の書き込み量が多い場合
 - 通信回線の負荷が高く、反映側からの応答に時間が掛かる場合
- `send_delay_limit_time = 送信滞留時間しきい値`
～ 〈符号なし整数〉 ((0~86400)) 《0》 (単位：秒)

更新情報がシステムログファイルに格納された時刻と、更新データが送信されて反映側 DBMS で受信が完了した時刻との差 (送信滞留時間) のしきい値を指定します。送信滞留時間が、ここで指定したしきい値よりも大きくなった場合、警告メッセージが出力されます。

送信環境定義の `queue_read_wait_interval` オペランドや `sendintvl` オペランドの指定値も、滞留時間に考慮する必要があります。なお、0 を指定した場合、送信プロセスでの滞留監視機能は無効になります。

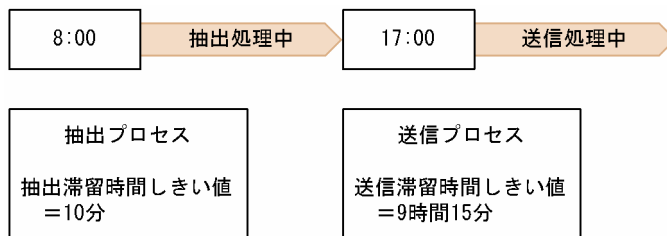
注意

このオペランドは、レプリケーション対象となる更新量や、送信間隔、反映間隔などを考慮した上で、値を調整してください。初めは、15 分 (900 秒) を目安に設定することをお勧めします。

運用例

送信処理を遅らせて開始する場合の運用例を次の図に示します。

図 5-6 送信処理を遅らせて開始する場合の運用例



例では、8 時~17 時まで抽出処理だけを実行し、17 時以降から送信処理を開始しています。このような運用環境下で送信滞留時間を監視する場合、更新データが反映システムに 9 時間送信されないうで、滞留したままとなるため、送信滞留時間しきい値に 9 時間分を足す必要があります。

- `reflect_mode = server | uap`
送信先の反映側 Datareplicator で、反映トランザクション同期機能を使用するかどうかを指定します。

server

反映トランザクション同期機能を使用しない場合に指定します。

uap

反映トランザクション同期機能を使用する場合に指定します。

このオペランドに uap を指定する場合、eventsync オペランドを指定する必要があります。また、overwrite オペランドに false を指定する必要があります。

- eventsync =同期イベントコード

～ 〈符号なし整数〉 ((1~128))

同期イベントのイベントコードを指定します。

このオペランドを指定する場合、reflect_mode オペランドに uap を指定する必要があります。

このオペランドには、eventcntreset オペランド及び prg_eventno オペランドに指定するイベントコードとは異なる値を指定してください。同じ値を指定するとエラーになります。

- eventcntreset =送信データ件数リセットイベントコード

～ 〈符号なし整数〉 ((1~128))

送信データ件数をリセットするためのイベントコードを指定します。抽出側 Datareplicator と反映側 Datareplicator でリセットを同期させる場合、反映環境定義の eventcntreset オペランドと同じ値にする必要があります。

5.5 抽出定義

抽出側 Datareplicator の抽出，送信処理に必要な情報を定義します。

5.5.1 構造と形式

(1) 構造

(a) 抽出対象の表を指定する場合

抽出定義の構造を「[図 5-7 抽出定義の構造](#)」に，抽出定義の内容と繰り返し指定できる指定数を「[表 5-7 抽出定義の内容と指定数](#)」に示します。

図 5-7 抽出定義の構造

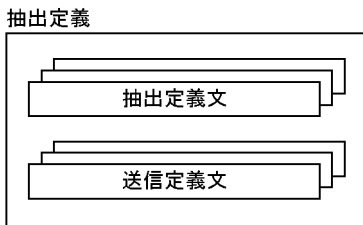


表 5-7 抽出定義の内容と指定数

定義名	定義文	指定数	内容
抽出定義文	extract 文	1~4096	抽出対象にする表，列，マッピングキーを定義します。
送信定義文	send 文	0~4096	更新情報の送信先を定義します。

(b) すべての表を抽出対象とする場合

抽出定義の構造を，「[図 5-8 抽出定義の構造（すべての表を抽出対象とする場合）](#)」に，抽出定義の内容と指定数を，「[表 5-8 抽出定義の内容と指定数（すべての表を抽出対象とする場合）](#)」に示します。

図 5-8 抽出定義の構造（すべての表を抽出対象とする場合）

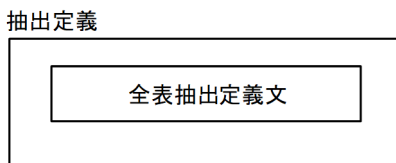


表 5-8 抽出定義の内容と指定数（すべての表を抽出対象とする場合）

定義名	定義文	指定数	内容
全表抽出定義文	extract_all 文	1	抽出側 DB の全実表をデータ連動の対象とし、連動対象外とする表を定義します。

(2) 形式

(a) 抽出対象の表を指定する場合

/* 抽出定義文 */

```
{{ extract 認可識別子.表識別子 ( {列名 [{{, 列名 }}...] | * } )
```

```
to 更新情報名
```

```
{key | ukey} (列名 [{{, 列名 }}...]) [check {not_null_unique | unique | none}]]...
```

/* 送信定義文 */

```
{{ send 送信先識別子 from 更新情報名
```

```
[ where 列名 {比較演算子 定数 | in (定数 [, 定数] ...)
```

```
| flike (比較開始位置, 定数)}
```

```
[ and 列名 {比較演算子 定数 | in (定数 [, 定数] ...)
```

```
| flike (比較開始位置, 定数)}}]
```

```
}}...]
```

```
;
```

(b) すべての表を抽出対象とする場合

/* 全表抽出定義文 */

```
extract_all [except 認可識別子.表識別子 {{, 認可識別子.表識別子}}...]
```

```
;
```

5.5.2 定義内容の変更

定義内容を変更する場合には、次のように操作してください。抽出側 Datareplicator が異常終了している場合には、一度正常開始してから定義内容を変更してください。

1. 抽出側 Datareplicator を正常終了します。
2. テキストエディタで定義内容を変更します。
3. 抽出側システムで hdeprep コマンドを実行します。
4. 抽出側 Datareplicator を正常開始します。

5.5.3 抽出定義文

データ連動の対象にする表の抽出条件を定義します。

(1) 形式

```
{ extract 認可識別子.表識別子 ( { 列名 [ { { , 列名 } } … ] | * } )
```

```
to 更新情報名
```

```
key | ukey ( 列名 [ { { , 列名 } } … ] ) [ check { not_null_unique | unique | none } ] } …
```

(2) オペランドの説明

- extract 認可識別子.表識別子 ({ 列名 [{ { , 列名 } } …] | * })

抽出する表とその表から抽出する列を指定します。

一つの抽出条件に対して、一つの extract 文を指定します。一つの extract 文で指定できる認可識別子.表識別子は 1 種類だけです。

一つの表を複数の抽出条件で抽出する場合には、複数の extract 文を指定します。一つの extract 文で指定できる列名は、4000 個までです。

認可識別子.表識別子

指定する列が含まれている表の、認可識別子.表識別子を指定します。指定する表は実表でなければなりません。

認可識別子

更新情報を抽出する表の、認可識別子を指定します。

表識別子

更新情報を抽出する表の、表識別子を指定します。

列名

抽出する列名を指定します。列名は、最大 4000 列 指定できます。

繰返し列は、要素を指定しないときにだけ指定できますが、配列列は指定できません。

同じ列名を重複して指定することもできます。ただし、key 句で指定する列名は、重複して指定できません。extract 文で指定した列の順序が、更新情報のフィールドの順序になります。更新情報のフィールドとは、更新情報中にあり、抽出した HiRDB のフィールドを格納するための領域を指します。

*

*を指定すると、抽出対象表のすべての列は、並び順を変えないでそのまま抽出されます。

• to 更新情報名

～ 〈1～8 文字の記号名称〉

抽出した更新情報に対する更新情報名を指定します。

更新情報名は、抽出定義中のすべての抽出文で一意に指定してください。

• {key | ukey} (列名 [{, 列名 }…])

マッピングキーにする列名を指定します。更新される可能性があるかどうかで、key 句か ukey 句にするかを決定します。

マッピングキーにできる列のデータ型については、「4.3.3(1) 抽出側 DB が HiRDB の場合のマッピングキー」を参照してください。

key 列名

マッピングキーにする列名を指定します。key 句で指定した列に対しては、更新しないでください。

ukey 列名

更新される可能性があるマッピングキーにする列名を指定します。

なお、マッピングキーが更新されるとデータ連動回復機能は使用できません。

列名は、抽出側及び反映側の HiRDB のバージョンが 09-50 以降の場合は最大 64 個、その他の場合は最大 16 個指定できます。ただし、同じ列名を重複して指定できません。

指定する列名は、extract 文で指定されている列名でなければなりません。extract 文で*を指定している場合は、抽出対象表の構成列のどれかを指定してください。

• check {not_null_unique | unique | none}

マッピングキー列に対するユニークチェックを行うときの条件を指定します。ユニークチェックの内容を次の表に示します。条件を満たしていなかった場合、抽出定義プリプロセス処理で定義エラーが発生します。この句を省略した場合は、hdeprep コマンドの-k オプションの指定に従ってチェックされます。

表 5-9 ユニークチェックの内容

チェック項目	チェック内容
インデクス種別	次のどれかのインデクス種別であること。 <ul style="list-style-type: none">• ユニークインデクス• ユニーククラスタインデクス• プライマリインデクス• プライマリクラスタインデクス
インデクス構成列	マッピングキー構成列だけであること。

not_null_unique

抽出対象表に対して、「表 5-8 ユニークチェックの内容」の内容を満たすインデクスが定義されていること、及びそのインデクスの構成列が NOT NULL 属性であることを確認します。

unique

抽出対象表に対して、「表 5-8 ユニークチェックの内容」の内容を満たすインデックスが定義されていることを確認します。NULL 値に関するチェックは行わないため、データの一意性は抽出側 DB を更新するユーザプログラムなどで保証する必要があります。

none

チェックを行いません。データの一意性は抽出側 DB を更新するユーザプログラムなどで保証する必要があります。

(3) 注意

- key 句を指定した場合は、extract 文にマッピングキー以外の列も指定してください。key 句を指定したマッピングキーは更新されないため、マッピングキー列だけを指定すると抽出できません。ukey 句を指定した場合は、extract 文にマッピングキー列だけを指定することもできます。
- null 値データの抽出はできます。
- 認可識別子、表識別子、列名、更新情報名及び送信先識別子は、大文字と小文字を区別して認識します。
- 認可識別子、表識別子、列名、更新情報名及び送信先識別子で抽出側 Datareplicator の予約語と同じ名称を指定する場合、名称を「"」（ダブルクォーテーション）で囲んで指定してください。抽出側 Datareplicator の予約語については、「付録 B Datareplicator 定義の予約語」を参照してください。
- extract 文で指定する列名には、マッピングキーになる列名を重複して指定できません。
- key 句を指定しているときにマッピングキーを更新すると、更新後のデータをキーとして反映側で処理されるため、反映処理でエラー（キーが一致するデータなし）又は意図したものとは異なる行に対して反映が実行されることとなります。マッピングキーは更新しないでください。

5.5.4 送信定義文

抽出側 DB から抽出した更新情報を、送信先識別子ごとに選択して送信するための定義をします。送信文は省略できます。送信文の指定方法によって、送信先へ送信される更新情報は異なります。送信文の指定と送信される更新情報を次の表に示します。

表 5-10 送信文の指定と送信される更新情報

送信文の指定	送信される更新情報	
	送信文で指定した送信先	送信文で指定していない送信先
送信文の指定なし（全省略）	—	全送信先に、すべての更新情報名の更新情報が送信されます。
一部の送信先に対して送信文を指定	送信文に指定した更新情報名の更新情報だけが送信されます。	すべての更新情報名の更新情報が送信されます。
すべての送信先に対して送信文を指定		—

(1) 形式

```
{ { send 送信先識別子 from 更新情報名  
  [ where 列名 {比較演算子 定数 | in (定数 [, 定数] …)  
    | flike (比較開始位置, 定数)}  
  [ and 列名 {比較演算子 定数 | in (定数 [, 定数] …)  
    | flike (比較開始位置, 定数)} ] ]  
} } … ]
```

(2) オペランドの説明

- send 送信先識別子

～ 〈1～8 文字の記号名称〉

抽出システム定義で指定した送信先識別子の中から、更新情報の送信対象にする**送信先識別子**を指定します。

- from 更新情報名

～ 〈1～8 文字の記号名称〉

send 句で指定した送信先識別子への送信対象にする**更新情報名**を、抽出文で指定した更新情報名の中から指定します。

更新情報名の先頭文字列が同じである複数の更新情報名を同一の送信先に送信する場合、半角の「%」によって前方一致指定で指定できます。半角の「%」は、0 文字以上の任意の文字列を示します。

(例) 前方一致指定の例

ABC%

上記の指定の場合、ABC, ABCAA, ABCABC など先頭の 3 文字が ABC の更新情報名が、送信の対象になります。

- where 句

where 句を指定すると、指定した条件に一致した更新情報だけを送信できます。where 句に指定する内容について説明します。

- 列名

送信対象とする更新データのマッピングキー列を指定します。ただし、選択条件列として指定できるマッピングキー列の属性は、次の表に示す属性に限ります。マッピングキー以外の列、又は次の表の属性以外の列を指定すると、定義エラーになります。

表 5-11 選択条件に指定できるマッピングキー列属性

列属性	長さ, 精度	位取り	指定できる長さ
char(n)	$n \leq 255$	—	$1 \leq n \leq 255$

列属性	長さ, 精度	位取り	指定できる長さ
mchar(n)	n ≤ 255	—	1 ≤ n ≤ 255
nchar(n)	n ≤ 127	—	1 ≤ n ≤ 127
varchar(n)	n ≤ 255	—	1 ≤ n ≤ 255
nvarchar(n)	n ≤ 127	—	1 ≤ n ≤ 127
[large] decimal(m,n)	1 ≤ m ≤ 38	0 ≤ n ≤ 38	1 ≤ m ≤ 38, m ≥ n
integer	4 バイト	—	4 バイト
smallint	2 バイト	—	2 バイト

- 比較演算子

比較演算子として、次の 6 種類が指定できます。

=, <>, >, >=, <, <=

- 定数

選択条件で比較する定数を指定します。選択条件に指定できる定数を「表 5-11 選択条件に指定できる定数」に、選択条件列の属性と定数との関連を「表 5-12 選択条件列属性と定数との関連」に示します。

表 5-12 選択条件に指定できる定数

定数種別	説明	指定例
文字列定数	アポストロフィで囲んだ文字列。 文字列定数としてアポストロフィを指定するときは、二つ続けて指定することで 1 文字のデータとして認識されます。1 から 255 バイトまで指定できます。	'AB"CD' '日本'
真数定数 又は符号なし整数	符号 (+, -), 数字, 小数点から構成される文字列。 10 進数で 29 けたまで指定できます。	-100 12.3

表 5-13 選択条件列属性と定数との関連

列属性	定数	
	文字列定数	真数定数 又は符号なし整数
char(n)	○※1	×
mchar(n)	○※1	×
nchar(n)	○※1	×
varchar(n)	○※2	×
nvarchar(n)	○※2	×
nvarchar(n)	○※2	×

列属性	定数	
	文字列定数	真数定数 又は符号なし整数
[large] decimal(m,n)	×	○
integer	×	○※3
smallint	×	○※4

(凡例)

○：指定できる定数

×：指定できない定数

注※1

n < 文字列定数長の場合：

定義エラー。

n > 文字列定数長の場合：

文字列定数に空白文字を補い、列長と同じ長さに調整後、比較します。文字集合指定がある列の場合は、補う空白文字はその列の文字集合の空白文字となります。文字集合指定がない列の場合は、抽出システム定義の dblocale オペランドに指定したロケールの空白文字となります。また、nchar の場合は 2 バイトコードの空白文字を付けます。nchar 属性列に対して奇数バイトの文字列定数を指定すると定義エラーになります。

注※2

n < 文字列定数長の場合：

定義エラー。

n > 文字列定数長の場合：

文字列定数長の分だけ左側から比較し、結果が等しければさらに文字列長を比較します。

注※3

-2147483648 から 2147483647 までの整数値だけが指定できます。そのほかは定義エラーになります。

注※4

-32768 から 32767 までの整数値だけが指定できます。そのほかは定義エラーになります。

- **in (定数 [, 定数…])**

選択条件列データが、右辺で指定した定数のどれかと一致した場合に、真と判定する条件を指定します。この条件で指定できる定数は、最大 16 個です。指定する定数については、比較演算子の場合に指定する定数の仕様に従います。

- **flike (比較開始位置, 定数)**

文字列属性の部分一致判定をする条件を指定します。選択条件列データに対して、比較開始位置から始まる文字列が定数と一致した場合に真と判定されます。比較開始位置の指定範囲は 0 から 254 までです。指定する定数については、比較演算子の場合に指定する文字列定数の仕様に従います。ただし、比較時に定数に対して空白文字は付けられません。

選択条件列の定義長が（比較開始位置+定数長）よりも短い場合は、定義エラーになります。また、選択条件列が可変長属性の場合、選択条件列の実データ長が（比較開始位置+定数長）よりも短い場合は、偽と判定されます。

(3) 注意

- 同一の送信先識別子、更新情報名の送信文を複数指定した場合は、定義エラーになります。
- 抽出文で定義した各更新情報名に対して、少なくとも一つ以上の送信先が対応するように定義してください。送信先が特定できない更新情報が存在する場合は、定義エラーになります。
- where 句に指定できる選択条件の数は、最大 256 個です。
- from 句で更新情報名を前方一致指定で指定している場合は、where 句を指定できません。
- from 句で、同一の抽出対象表より複数の抽出（複数の更新情報名を定義）する更新情報名を指定している場合は、where 句は指定できません。
- where 句で、文字集合指定のある列を選択条件にした場合、大小比較の結果は、該当する列の文字集合内での大小関係によって決まります。
- 選択条件列データが null 値の場合は、判定結果は偽になります。
- send 文に指定した更新情報名を定義した extract 文で ukey 句を指定している場合は、更新前データで条件判定します。そのため、送信条件に指定したマッピングキー列に対して送信条件に該当しない値に更新をすると、送信先データが送信対象外のキーに更新されてしまいます。更新されたデータは不当になって、更新後の値に対応する送信先に、あるべきデータが存在しないという矛盾状態になります。送信条件に指定したマッピングキー列を送信条件の範囲外の値に更新する場合は、単純に UPDATE を実行するのではなく、DELETE と INSERT を実行してデータの存在の整合を取る必要があります。

5.5.5 全表抽出定義文

抽出側 DB の全実表をデータ連動の対象とし、連動対象外とする表を定義します。

(1) 形式

`extract_all [except 認可識別子.表識別子 {{, 認可識別子.表識別子}}…]`

(2) オペランドの説明

- `extract_all`

`extract_all` 句を指定すると、抽出側 DB の全実表の全列が抽出対象となります。抽出側 DB のすべての表を同一構成で反映側へ連動する場合、`extract_all` 句を使用すると抽出定義の記載を簡略化できます。`extract_all` 句を指定する場合、更新情報名とマッピングキーは次のように決定します。

- 更新情報名

抽出対象表に対応する更新情報名を Datareplicator で作成します。作成する更新情報名は、ALLxxxxx (xxxxxx は 5 桁の通番) となります。このため、反映側のエラーメッセージに出力された更新情報名から、エラーの対象となる表などを特定できない場合があります。その場合、hdsrefinfmt コマンドで抽出定義情報を出力して、更新情報名やマッピングキーに対応する表と列の情報を確認します。

• マッピングキー

マッピングキーに該当する列は、各表に定義されたユニークインデックスの構成列とします。マッピングキーに使用するユニークインデックスは、hdeprep コマンドの -k オプションの指定値から次の表のとおり決定します。-k オプションに none を指定した場合は、KFRB04587-E メッセージを出力してエラー終了します。マッピングキーに使用できるユニークインデックスが定義されていない表は抽出対象外とし KFRB04583-W を出力します。

マッピングキーに該当する列は更新できます。

表 5-14 マッピングキーのユニーク条件とインデックス条件

hdeprep コマンドの -k オプション指定値	マッピングキーに使用するインデックス	
	インデックス種別と優先順	インデックス構成列の条件
not null unique 又は省略	<p>下記インデックスの内、インデックス構成列の条件を満たすものを 1~3 の優先順で決定します。</p> <ol style="list-style-type: none"> 1. ユニーククラスタインデックス 2. 主キーインデックス 3. ユニークインデックス (インデックス ID が小さい順) 	<ul style="list-style-type: none"> • すべての列が、マッピングキーとして使用できるデータ型及びデータ長 ([4.3.3(1) 抽出側 DB が HiRDB の場合のマッピングキー] 参照) であること。 <p>すべての列が NOT NULL 属性であること。</p>
unique		<p>すべての列が、マッピングキーとして使用できるデータ型及びデータ長 ([4.3.3(1) 抽出側 DB が HiRDB の場合のマッピングキー] 参照) であること。</p>

• except 認可識別子.表識別子 {{, 認可識別子.表識別子}}...

extract_all 句を指定する場合に、抽出対象外とする表を指定します。except 句は最大 2048 個指定できます。指定数が最大数を越えた場合は KFRB04585-E メッセージを出力してエラー終了します。また、抽出側 DB に定義されていない表を指定すると KFRB04584-E メッセージを出力してエラー終了します。同じ表を重複して指定してもエラーにはなりません。

(3) 注意

- 抽出対象の表数が 0 の場合には KFRB04586-E メッセージを出力してエラー終了します。
- 抽出対象の表数が 4097 以上ある場合には KFRB04582-E メッセージを出力してエラー終了します。
- 各制限事項によって抽出対象外とした表の場合、KFRB04583-W メッセージを出力して定義解析を続行します。KFRB04583-W メッセージは、抽出対象外とした表数分出力します。
- 抽出側 DB へ抽出対象表の追加、および表や列、インデックスに関する定義を変更する場合は、抽出定義の変更は不要ですが、hdeprep コマンドの再実行が必要です。手順については、「6.8 HiRDB 及び

[Datareplicator の構成変更](#)」を参照してください。抽出側 DB へ抽出対象外の表を追加する場合は、次回 hdeprep コマンドの実行前に、抽出対象外の表を except 句に指定する必要があります。

- 反映側 Datareplicator では、反映定義を省略するか、又は、反映定義で反映グループ定義だけを指定してください。

5.6 抽出側 HiRDB の定義

HiRDB のデータベースからデータを抽出する場合は、抽出側 HiRDB の定義に、抽出側 Datareplicator を使うために必要な情報を定義します。

抽出側 Datareplicator を使うために必要な HiRDB の定義を次の表に示します。HiRDB の定義の記述形式とこのほかの HiRDB の定義については、マニュアル「HiRDB システム定義」を参照してください。

表 5-15 抽出側 Datareplicator を使うために必要な HiRDB の定義

定義名	記述形式	オペランド	内容	再開始時の変更可否
システム共通定義	set 形式	pd_rpl_init_start	抽出側 HiRDB でのデータ連動の開始方法を指定します。	×
		pd_log_rpl_no_standby_file_opr	システムログファイルがスワップできなかった場合の、データ連動に対する処置を指定します。	○
ユニット制御情報定義	set 形式	pd_rpl_hdepath	抽出側 Datareplicator で使うディレクトリ名を指定します。	×

(凡例)

- ：HiRDB の再開始時にオペランドを変更できます。
- ×：HiRDB の再開始時にオペランドを変更できません。

5.6.1 システム共通定義

(1) 形式

:

```
[ set pd_rpl_init_start = Y | N ]
```

```
[ set pd_log_rpl_no_standby_file_opr = stop | continue ]
```

:

(2) オペランドの説明

- pd_rpl_init_start = Y | N

HiRDB Datareplicator 連携を、HiRDB 開始時から開始するかどうかを指定します。

Y

HiRDB 開始時から、HiRDB Datareplicator 連携を開始します。Y 指定時は、HiRDB Datareplicator 連携に必要なデータ運動情報が、HiRDB 開始時からシステムログに出力されます。

N

HiRDB 開始時から、HiRDB Datareplicator 連携を開始しません。N 指定時に、HiRDB Datareplicator 連携を開始したい場合には、HiRDB の `pdrplstart` コマンドを使ってください。`pdrplstart` コマンド指定時の運用方法については、「[6.5 抽出側 HiRDB の運用](#)」を参照してください。

- `pd_log_rpl_no_standby_file_opr = stop | continue`

HiRDB Datareplicator 連携を実行している場合に、抽出側 Datareplicator でのシステムログの抽出が完了していないために、すべてのシステムログファイルがスワップ先にできない状態でスワップ要求が発生した場合の HiRDB Datareplicator 連携に対する処置を指定します。

このオペランドは、`pd_rpl_init_start = Y` 又は `pdrplstart` コマンドを使って、HiRDB Datareplicator 連携を実行している場合にだけ有効です。

このオペランドは、HiRDB/パラレルサーバの場合のフロントエンドサーバ、及びディクショナリサーバに対しては無効になります。

stop

HiRDB のユニットを強制停止します。

強制停止した HiRDB のユニットを再開始する場合の運用方法については、「[6.5 抽出側 HiRDB の運用](#)」を参照してください。

continue

HiRDB Datareplicator 連携を中止して、HiRDB での処理だけを続行します。この場合、データ運動の対象になっている抽出側 DB と反映側 DB との間で不整合が発生するため、反映側 DB の再作成が必要です。反映側 DB の再作成については、「[6.5 抽出側 HiRDB の運用](#)」を参照してください。

5.6.2 ユニット制御情報定義

(1) 形式

:

set `pd_rpl_hdepath` = 抽出側 Datareplicator 運用ディレクトリ名

:

(2) オペランドの説明

- `pd_rpl_hdepath` = 抽出側 Datareplicator 運用ディレクトリ名
～ 〈パス名〉

抽出側 Datareplicator が使うディレクトリ名を指定します。抽出側 Datareplicator の、環境変数 HDEPATH で指定したディレクトリ名を指定してください。

このオペランドを指定していないと、HiRDB Datareplicator 連携機能は使えません。このオペランドは、システム共通定義での pd_rpl_init_start = Y 指定又は pdrplstart コマンドの指定によって、HiRDB Datareplicator 連携を実行している場合にだけ有効です。

5.7 二重化定義（抽出側）

5.7.1 二重化定義（抽出側）の形式と説明

抽出側システムの二重化定義で使用する論理ファイル名と物理ファイル名の対応を定義します。

(1) 形式

```
{ { logical_file = 論理ファイル名
    physical_file_a = 物理ファイル名1
    physical_file_b = 物理ファイル名2
# コメント行
} }
:
```

定義文の 1 行の長さは半角 1,024 文字以内です。コメントを記述する場合は、行の先頭に「#」を記述してください。

(2) オペランドの説明

`logical_file` = 論理ファイル名

二重化するファイルの論理ファイル名を指定します。ファイルごとの論理ファイル名を次に示します。

対象システム	定義ファイル	論理ファイル名
抽出側システム	抽出マスタステータスファイル	mststatus
	抽出サーバステータスファイル	sts_サーバ名
	抽出情報キューファイル	抽出環境定義に指定した名称_サーバ名
	データ連動用連絡ファイル	hde_サーバ名

論理ファイル名を重複して定義した場合は、定義解析エラーが発生します。また、抽出環境定義ファイル中にない論理ファイル名称を定義した場合、その論理ファイル名に関する定義は無視されます。

`physical_file_a` = 物理ファイル名 1, `physical_file_b` = 物理ファイル名 2

～((1～125)) (単位: バイト)

`logical_file` で指定した論理ファイルを構成する物理ファイル名を絶対パスで指定します。データ連動用連絡ファイルを二重化する場合、物理ファイル名 1, 又は物理ファイル名 2 のどちらか一方に論理ファイル名を指定してください。物理ファイル名を重複して定義した場合は、定義解析エラーが発生します。キャラクタ型スペシャルファイルを使用する場合、ノード間で名称が重複しないようシンボリックリンクを作成して、すべての物理ファイル名を一意の名称にしてください。

5.8 反映システム定義

反映システム定義では、反映側 Datareplicator の稼働環境についての情報を定義します。

5.8.1 形式

```
[ set hdsid = 反映側 Datareplicator 識別子 ]
[ set hirdbusr = HiRDB 接続認可識別子 [ /パスワード ] ]
[ set protocol1 = tcp | osi ] ※1
[ set protocol2 = tcp | osi ] ※1
set dsid001 = データ連動識別子
[ … [ set dsid128 = データ連動識別子 ] ] ※2
set refenv001 = "反映環境定義ファイル名"
[ … [ set refenv128 = "反映環境定義ファイル名" ] ] ※2
[ set hdsservice = サービス名 ]
[ set reflect_tselector = T セレクタ ] ※1
[ set hirdb_audit_trail = all | uoc | none_cont | none_stop ]
[ set keepalive = true | false ]
[ set errfilesz = 反映エラー情報ファイルサイズ ]
[ set syslogout = true | false ]
[ set syslog_message_suppress = メッセージ番号 [, メッセージ番号] … ]
[ set dblocale = { sjis | euc | utf-8 | unknown } ]
[ set msglocale = { english | sjis-japanese | euc-japanese } ]
[ set discintvl = disconnect 発行間隔 ]
[ set info_message_out = nosuppress | suppress ]
[ set except_suppress = メッセージ番号 [, メッセージ番号] … ]
[ set commitment_method = fxa_none | fxa_sqle ]
```

[set int_trc_lvl = 稼働トレース取得レベル [, 稼働トレース取得範囲]]

[set int_trc_filesz = 稼働トレースファイルのサイズ]

[set int_trc_rintvl = 稼働トレース情報取得間隔]

[set int_trc_unit = system | process] ※3

[set int_trc_output_path = 稼働トレースファイル出力先] ※3

[set use_convertlib = true | false] ※3

[set ref_wait_interval = 反映プロセスの反映情報キューファイル読み込み間隔]

[set commit_wait_time = COMMIT 発行間隔]

[set file_dupenv = 二重化定義ファイル名]

[set syncgroup001 = 同期反映グループ名, データ連動識別子 [{ , データ連動識別子 } …]]

[set syncgrp_discintvl = disconnect 発行待ち時間]

[set syncwait_limit_tran_count = 同期待ち限界トランザクション数]

[set syncwait_limit_time = 同期待ち限界時間]

[set reflect_counter_reset = true | false]

注※1

OSI プロトコルで通信を実行する場合に指定します。ただし、このオペランドは HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

注※2

Windows 版 Datareplicator ではデータ連動識別子とそれに対応する反映環境定義ファイルは 1~63 個までなので、dsid064~dsid128 と refenv064~ refenv 128 オペランドは指定できません。これらのオペランドを指定するとエラーになります。

注※3

Windows 版 Datareplicator のときにだけ指定できます。UNIX 版 Datareplicator のときに、このオペランドを指定するとエラーになります。

5.8.2 定義内容の変更

定義内容を変更する場合には、次のように操作してください。

1. 反映側 Datareplicator を終了します。

初期開始をする場合は、終了モードは正常終了、即時終了、又はイベント終了にする必要があります。反映側 Datareplicator が異常終了している場合には、一度正常開始してから定義内容を変更してください。

2. テキストエディタで定義内容を変更します。

3. 次の表に示す、変更後の定義内容が有効になる開始モードで開始します。

表 5-16 反映システム定義の変更後の定義内容が有効になる開始モード

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始※1	部分初期開始※1	正常開始※1
hdsid	○	—	—
hirdbusr	○	○	○
protocol1	○	—	—
protocol2	○	—	—
dsidxxx※2	○	○	—
refenvxxx※2	○	○	—
hdsservice	○	○	○
reflect_tselector	○	○	○
hirdb_audit_trail	○	○	○
keepalive	○	○	○
errfilesz	○	○	○
syslogout	○	○	○
syslog_message_suppress	○	○	○
dblocale	○	—	—
msglocale	○	○	○
discintvl	○	○	○
info_message_out	○	○	○
except_suppress	○	○	○
commitment_method	○	—	—
int_trc_lvl	○	○	○
int_trc_filesz	○	○	○
int_trc_rintvl	○	○	○
int_trc_unit	○	○	○
int_trc_output_path	○	○	○

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始※1	部分初期開始※1	正常開始※1
use_convertlib	○	○	○
ref_wait_interval	○	○	○
commit_wait_time	○	○	○
file_dupenv	○	—	—
syncgroup001	○	—	—
syncgrp_discintvl	○	○	○
syncwait_limit_tran_count	○	○	○
syncwait_limit_time	○	○	○
reflect_counter_reset	○	○	○

(凡例)

○：対応する動作によって変更内容が有効になることを示します。複数の動作に対して記載されている場合は、どれか一つの動作を実行することで変更内容が有効になることを示します。

—：該当しません。

注※1

反映側 Datareplicator を起動するときの開始モードです。

注※2

xxx には、001～128 が対応します。正常開始する場合は、dsid001～dsid128 オペランドと refenv001～refenv128 オペランドを変更しないでください。

5.8.3 オペランドの説明

- hdsid = 反映側 Datareplicator 識別子

～ 〈16 進数字〉 ((00～FF)) 《00》

複数の反映側 Datareplicator を動作させる場合には、反映側 Datareplicator を識別するための識別子を指定します。反映側 Datareplicator が一つだけの場合は、省略できます。

- hirdbusr = HiRDB 接続認可識別子 [/パスワード]

《反映側 HiRDB の環境変数「PDUSER」の設定値》

反映側 HiRDB へ接続するときを使う認可識別子とパスワードを指定します。

- HiRDB 接続認可識別子

～ 〈1～8 文字の識別子〉

反映側 HiRDB へ接続するときを使う認可識別子を指定します。不正な認可識別子を指定すると、エラーになります。

- /パスワード

～ 〈1～28 文字の記号名称〉

反映側 HiRDB へ接続するときを使うパスワードを指定します。不正なパスワードを指定すると、エラーになります。また、「/パスワード」を省略すると、パスワードなしとして反映側 HiRDB に接続します。

UNIX 版の場合は、パスワードに「*」を指定すると、反映側 Datareplicator の起動時に、標準入力へのパスワードの入力要求があります。ここで入力したパスワードをそのまま使って、反映側 HiRDB へ接続します。

Windows 版の場合は、パスワードに「*」を指定すると、パスワードなしとして反映側 HiRDB に接続します。

このオペラントで指定した内容については、大文字又は小文字を区別するために「"」で指定値を囲む必要はありません。「"」で指定値を囲んだ場合は、「"」もユーザ ID 又はパスワードの一部として認識されるため、エラーとなります。

- protocol1 = tcp | osi

抽出側システムとの通信プロトコルを指定します。

tcp

TCP/IP プロトコルで通信を実行します。

osi

OSI プロトコルで通信を実行します。

このオペラントは、HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

- protocol2 = tcp | osi

抽出側システムとの通信プロトコルを複数 使う場合に指定します。protocol1 で指定した通信プロトコルと同じプロトコルを指定した場合は、protocol2 は無視されます。

tcp

TCP/IP プロトコルで通信を実行します。

osi

OSI プロトコルで通信を実行します。

このオペラントは、HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

- dsid001 = データ連動識別子

… dsid128 = データ連動識別子

～ 〈16 進数字((00～FF)), 又は**〉

dsidxxx には、反映処理の対象とする抽出側システムのデータ連動識別子を指定します。

データ連動識別子は、反映システム定義内で一意にしてください。

複数のデータ連動識別子をデータ連動の対象とする場合、dsid001～dsid128 オペラントに指定する 001～128 の数字は、001 から昇順に連続して指定してください。昇順に連続して指定していない場合は、正しい順序で指定されているオペラントまでが有効になります。

データ連動識別子に**を指定すると、欠番として扱われ、このデータ連動識別子でのデータ連動（受信プロセスや定義サーバプロセスの起動）は行いません。

は複数のデータ連動識別子に指定できますが、すべての dsidxxx にを指定した場合は、データ連動識別子が指定されていないものとして扱われます。

Windows 版 Datareplicator ではデータ連動識別子とそれに対応する反映環境定義ファイルは 1~63 個までなので、dsid064~dsid128 オペランドは指定できません。これらのオペランドを指定するとエラーになります。

- **refenv001 = "反映環境定義ファイル名"**

… refenv128 = "反映環境定義ファイル名"

～ 〈1~64 バイトの [パス名/] ファイル名〉

refenvxxx には、反映処理の稼働環境を定義した反映環境定義ファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDSPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが 125 バイト以下になるように指定してください。

ファイル名は、システムで一意になるようにしてください。

refenv001~refenv128 オペランドに指定する 001~128 の数字は、dsid001~dsid128 オペランドと対応する必要があります。

dsid001~dsid128 に対応する refenv001~refenv128 がない場合はエラーになります。また、refenv001~refenv128 に対応する dsid001~dsid128 がない場合は、対応していない refenv は無視されます。

Windows 版 Datareplicator ではデータ連動識別子とそれに対応する反映環境定義ファイルは 1~63 個までなので、refenv064~refenv 128 オペランドは指定できません。これらのオペランドを指定するとエラーになります。

- **hdsservice = サービス名**

～ 〈1~64 文字の識別子〉 《hirdbds》

反映側 Datareplicator の通信エントリを services ファイルに追加するときに指定したサービス名を指定します。

- **reflect_tselector = T セレクタ**

～ 〈2~64 けたの範囲の偶数けたの 16 進数字〉

protocol1 又は protocol2 で osi を指定した場合に指定が必要になります。

反映側 Datareplicator と抽出側システムに割り当てた T セレクタを指定します。反映側 Datareplicator と抽出側システムで指定する T セレクタは同じでなければなりません。抽出側システムで指定する T セレクタが文字列の場合、16 進文字列に変換したものを指定してください。

このオペランドは、HP-UX 版 Datareplicator (IPF 版は除きます) だけで有効です。

- **hirdb_audit_trail = all | uoc | none_cont | none_stop**

監査証跡の取得対象になる操作（監査対象イベント）が発生した場合に、監査証跡を取得するかどうかを指定します。

監査対象イベントとなるのは、反映定義サーバプロセス、反映 SQL プロセス、CONNECT 処理以外の反映 UOC プロセスです。反映 UOC プロセスでは、HiRDB にアクセスする反映情報編集 UOC に

対してだけ、このオペランドが有効になります。なお、反映情報編集 UOC を使用する場合に監査証跡を取得しないときは、必ず「8.1.6(7) 反映処理に関する監査証跡を取得しない場合の注意」を参照してください。

all

反映側 Datareplicator の処理による監査証跡を取得します。

uoc

反映情報編集 UOC の処理による監査証跡を取得します。

none_cont

監査証跡を取得しません。ただし、反映情報編集 UOC で、KFRB00864-W、又は KFRB03094-W メッセージが出力された場合は、その後の監査証跡を取得して処理を続行します。

none_stop

監査証跡を取得しません。ただし、反映情報編集 UOC で、KFRB00865-E、又は KFRB03094-W メッセージが出力された場合は、処理を停止します。

反映側 Datareplicator の処理による監査証跡取得のオーバーヘッドを抑止したい場合は、none_cont、又は none_stop を指定することをお勧めします。

- **keepalive = true | false**

ソケットにキープアライブオプションを設定するかどうかを指定します。

true

キープアライブオプションを設定します。

false

キープアライブオプションを設定しません。

protocol1 オペランド又は protocol2 オペランドで osi を指定した場合には、keepalive オペランドの指定は無効になります。

- **errfilesz = 反映エラー情報ファイルサイズ**

～ 〈符号なし整数〉 ((1~2097151)) 《16》 (単位:KB)

反映エラー情報ファイルの上限サイズを指定します。

- **syslogout = true | false**

反映エラー情報ファイルに出力される情報を、syslog ファイルにも出力するかどうかを指定します。

true

反映エラー情報ファイルに出力される情報を、syslog ファイルに出力します。

false

反映エラー情報ファイルに出力される情報を、syslog ファイルに出力しません。

- **syslog_message_suppress = メッセージ番号 [, メッセージ番号] …**

syslog ファイル (Windows の場合はイベントログ) への出力を抑止したいメッセージ番号を指定します。

- メッセージ番号は 64 個まで指定できます。
 - 同じメッセージ番号を重複して指定すると、1 件目のメッセージ番号だけを有効にして、2 件目以降の重複したメッセージ番号は無視されます。
 - このオペランドに指定したメッセージは、メッセージの重要度 (E, W, I, Q) が何であっても、出力が抑止されます。
 - syslog ファイル (又はイベントログ) にだけ出力されるメッセージは、このオペランドを指定しても出力されます。
- `dblocale = { sjis | euc | utf-8 | unknown }`

抽出側システムから送信された更新情報を、反映側システムの格納文字コードに変換するときの文字コードを指定します。また、反映定義解析でもこのオペランドで指定した文字コードで反映定義が記述されていると認識し、反映定義を解析します。

sjis

JIS8/シフト JIS コード体系に更新情報を変換します。

euc

EUC コード体系に更新情報を変換します。ただし、コードセット 3 については SQL エラーになることがあります。

utf-8

UTF-8 文字コード体系にデータを変換します。

unknown

文字コードを変換しません。UOC で文字コードを変換するときに指定します。unknown を指定した場合は、反映定義解析では、LANG 環境変数の指定に従って次のように反映定義の文字コードを認識して、反映定義を解析します。

LANG=ja_JP.SJIS : JIS8/シフト JIS コード体系

LANG=上記以外 : EUC コード

このオペランドの省略時仮定値は、OS によって異なります。次の表で確認してください。

OS	省略時仮定値
HP-UX 版	sjis
AIX 版	
Windows 版	
Solaris 版	euc
Linux 版	

Windows 版 Datareplicator のときは、抽出側 Datareplicator が動作する文字コード体系は JIS8/シフト JIS コード体系と Unicode 体系だけなので、「sjis」、「utf-8」又は「unknown」のどれかを指定します。これ以外の値を dblocale オペランドに指定するとエラーになります。

- `msglocale = { english | sjis-japanese | euc-japanese }`

反映側 Datareplicator が出力するメッセージの文字コードを指定します。

Windows 版 Datareplicator の場合、english 又は sjis-japanese のどちらかを指定してください。euc-japanese を指定するとエラーになります。

english

英文のメッセージを出力します。

sjis-japanese

JIS8/シフト JIS コード体系で和文のメッセージを出力します。

euc-japanese

EUC コード体系で和文のメッセージを出力します。

- **discintvl = disconnect 発行間隔**

～ 〈符号なし整数〉 ((0~65535)) 《180》 (単位：秒)

反映処理が反映情報キューファイルに蓄積されている更新情報の終端を検知してから、反映側 HiRDB に対して disconnect を発行するまでの間隔を指定します。

0 を指定した場合は、反映処理は更新情報の終端を検知しても、反映側 HiRDB に対して disconnect を発行しません。disconnect の発行契機は、commit_wait_time オペラントと同じです。

- **info_message_out = nosuppress | suppress**

情報通知レベルのメッセージを、syslog ファイル (Windows の場合はイベントログ)、及びエラー情報ファイルに出力しないようにするかどうかを指定します。

nosuppress

情報通知レベルのメッセージの出力を抑止しません。

suppress

情報通知レベルのメッセージの出力を抑止します。出力抑止の対象になるメッセージの番号は次のとおりです。

00100, 00103, 00104, 02019, 02020, 02021, 02022, 03001, 03002, 03008, 03009,
03011, 03012, 03013, 03022, 03201, 03202, 03204, 03209, 03028, 03058, 03301,
03302

- **except_suppress = メッセージ番号 [, メッセージ番号] …**

～ 〈5 けたの符号なし整数〉

info_message_out オペラントに suppress を指定して出力を抑止したメッセージのうち、抑止対象から除外して syslog ファイルやエラー情報ファイルに出力したいメッセージ番号を指定します。メッセージ番号は 63 個まで指定できます。

出力抑止の対象でないメッセージ番号を指定しても、無効になります (特に意味を持ちません)。同じメッセージ番号を重複して指定すると、1 件目のメッセージ番号だけを有効にして、2 件目以降の重複したメッセージ番号は無視されます。

このオペラントは、info_message_out オペラントに suppress を指定したときだけ有効です。

- **commitment_method = fxa_none | fxa_sql**

反映処理の同期点処理方式（1相コミット方式又は2相コミット方式）を指定します。反映処理の同期点処理方式の詳細については、「3.3.12 反映処理の同期点処理の指定」を参照してください。

なお、このオペランドを変更したときは、反映側 Datareplicator を初期開始してください。

fxa_none

同期点処理を1相コミット方式で実行します。反映先 RDBMS が HiRDB 以外のときは、この値以外を指定しても無視されて、1相コミット方式で同期点処理が実行されます。反映情報編集 UOC を使用する場合、fxa_none を指定してください。

fxa_sqlc

反映 SQL プロセスでの同期点処理を2相コミット方式で実行します。ただし、反映 UOC プロセスでの同期点処理は、1相コミット方式で実行します。

反映トランザクション同期機能を使用する場合（syncgroup001 オペランドを指定する場合は、fxa_sqlc を指定してください。

また、反映列に繰返し列を含む場合は、fxa_sqlc を指定してください。

一つの HiRDB に対して複数の反映側 Datareplicator で2相コミット方式を使って反映するときは、反映側 Datareplicator 識別子に同じ値は指定できません。このようにして一つの HiRDB に反映すると、同期点処理を正常に実行できません。

[重要]

このオペランドに fxa_sqlc を指定している場合、反映側 Datareplicator を初期化する前に、反映先 HiRDB に未決着のトランザクションが残っていないことを必ず確認した後、初期化してください。

未決着のトランザクションが残っている場合は、pdcmnt、もしくは pdrbk コマンドでトランザクションを決着させた後に初期化してください。

• int_trc_lvl = 稼働トレース取得レベル [、稼働トレース取得範囲]

稼働トレースファイルで採取する項目を変更するときに、次に示す表の値を指定します。このオペランドを省略すると、すべての機能の共通情報（最小限必要な情報）だけを取得します。

稼働トレース取得レベルに na を指定すると、稼働トレースは取得されません。さらに、稼働トレース取得範囲に指定した値は無視されます（ただし、構文エラーや指定範囲のエラーはチェックされます）。

• 稼働トレース取得レベルに指定する値

指定する値	取得する情報		
	共通情報	性能概要	性能詳細
p1	○	○	×
p2	○	○	○
na	×	×	×
int_trc_lvl オペランドを省略	○	×	×

(凡例)

○：採取します。

×：採取しません。

共通情報：開始・終了情報，エラー情報，プロセスレベルでの変局点を示す大まかなチェックポイント情報の総称。

注意

p2 を指定したときは，稼働トレースファイルのサイズを大きく設定しておかないと，すぐにラップアラウンドして必要な情報が消えてしまうことがあります。また，トレース採取のためのオーバーヘッドが大きくなることもあります。

• 稼働トレース取得範囲に指定する値

指定する値	取得する情報			
	MST (制御)	RCV (受信)	RFC (反映)	SQE (SQL 実行)
c1	○	○	○	×
c2	○	○	×	○
c3	○	○	×	×
c4	○	×	○	○
c5	○	×	○	×
c6	○	×	×	○
c7	○	×	×	×
nc	○	×※	×※	×※
int_trc_lvl オペランドを省略又は稼働トレース取得範囲を省略	○	○	○	○

(凡例)

- ：採取します。
- ×：採取しません。

注※

反映システム定義で nc を指定したときは，反映環境定義の int_trc_getl オペランドの指定で稼働トレースを個別に取得できます。

int_trc_lvl オペランドの指定の目安

int_trc_lvl オペランドは，次のように指定することをお勧めします。

1. 本番運用のとき

本番運用では，int_trc_lvl オペランドを省略することをお勧めします。

この指定で性能が出ないときは，一時的に稼働トレースのオペランドを変更して性能状況を見てください。このとき int_trc_lvl オペランドの第 1 パラメタに「p1」又は「p2」，第 2 パラメタに「nc」を指定してから，特定の（性能が出ていない）機能に int_trc_getl 又は int_trc_getv オペランドを指定します。int_trc_lvl オペランドの第 1 パラメタに「p2」を指定したときは，int_trc_filesz オペランドには余裕を持った値（1MB 以上を推奨）を指定してください。

2. テスト運用のとき

テストの段階では、int_trc_lvl オペランドの第 1 パラメタに「p1」を指定して、第 2 パラメタを省略することをお勧めします。こうしておけば、性能の問題が起こったときに hdstrcredit コマンドを実行していつでも稼働トレースを参照できます。コマンドの実行結果を参照して、HiRDB の表定義や反映グループ数をチューニングしてください。

より詳細な情報（SQL の実行単価など）が必要なときは、int_trc_lvl オペランドの第 2 パラメタを「p2」に変更してください。「p2」を指定したときは、int_trc_filesz オペランドには余裕を持った値（1MB 以上を推奨）を指定してください。

- int_trc_filesz = 稼働トレースファイルのサイズ

～ 〈符号なし整数〉 ((32~2097120)) 《128》 (単位：KB)

稼働トレースファイル 1 個当たりの最大格納サイズを指定します。

int_trc_lvl オペランドに na を指定すると、このオペランドを指定しても無視されます。

このオペランドに指定した値を 32KB の単位で切り上げた値が、ファイル 1 個当たりの最大容量になります。そのため、指定する値は 32 の倍数にすることをお勧めします。

稼働トレースファイルは、2 個のファイルを相互にスワップ/ラップしながら使います。

- int_trc_unit オペランドに system を指定、または省略した場合

反映トレースファイル：

\$HDSPATH/reftrc.trc1 と \$HDSPATH/reftrc.trc2

- int_trc_unit オペランドに process を指定した場合

表「[反映側 Datareplicator の稼働トレースファイル名](#)」を参照してください。

- int_trc_rintvl = 稼働トレース情報取得間隔

～ 〈符号なし整数〉 ((5~30000)) 《50》 (単位：ミリ秒)

稼働トレース情報を取得する間隔を指定します。

int_trc_lvl オペランドに na を指定すると、このオペランドを指定しても無視されます。

稼働トレース情報取得間隔を小さくすると、欠落する稼働トレース情報は減らすことができますが、一秒当たりの監視回数が増加するため、CPU 利用率は高くなります。

次の場合は、稼働トレース情報取得間隔を小さくしてください。

- 稼働トレース情報に欠落が発生している場合
- 稼働トレース取得レベルで、取得情報の多いレベルが指定されている場合
- Datareplicator の稼働頻度が高い（反映データが多い）場合

また、次の場合は、稼働トレース情報取得間隔を大きくしてください。

- Datareplicator の稼働頻度が低く（反映データが少なく）、Datareplicator による CPU 利用率を削減したい場合

- int_trc_unit = system | process

稼働トレース収集方式を指定します。このオペランドは、Windows 版 Datareplicator の場合だけ指定できます。

system を指定すると、複数プロセスから同時に 1 つのファイルに書き込みを行うため、排他制御を行います。したがって、複数プロセスからの稼働トレース出力が集中すると、排他制御がボトルネックとなり、Datareplicator 全体の処理性能が劣化することがあります。反映システム定義の int_trc_lvl オペランドに p2 を指定し、かつ、反映プロセスが複数ある環境では、反映プロセス数に比例して性能劣化の度合いが大きくなる傾向があります。process を指定すると、プロセスごとに稼働トレースファイルを割り当てるため排他制御が不要となり、性能への影響を抑えることができます。

system

反映側 Datareplicator の全プロセスの稼働トレースを同一の稼働トレースファイルに出力します (全プロセス一括収集方式)。

process

反映側 Datareplicator の各プロセスの稼働トレースを別々の稼働トレースファイルに出力します (プロセス単位収集方式)。

process を指定した場合の稼働トレースファイル名を、表「[反映側 Datareplicator の稼働トレースファイル名](#)」に示します。稼働トレースファイルの出力先ディレクトリは、int_trc_output_path オペランドに指定します。

表 5-17 反映側 Datareplicator の稼働トレースファイル名

プロセス名	稼働トレースファイル名
反映マスタプロセス	hdsmaster.trc1 hdsmaster.trc2
反映通信マスタプロセス	hdstcpmst.trc1 hdstcpmst.trc2
受信プロセス	hdstcpmst_データ連動識別子.trc1 hdstcpmst_データ連動識別子.trc2 (例：hdstcpmst_01.trc1)
反映定義サーバプロセス	hdsdefserv_データ連動識別子.trc1 hdsdefserv_データ連動識別子.trc2 (例：hdsdefserv_01.trc1)
反映プロセス※	hdsreflect_データ連動識別子_反映グループ名.trc1 hdsreflect_データ連動識別子_反映グループ名.trc2 (例：hdsreflect_01_trngroup.trc1)
反映 SQL プロセス※	hdssqle_データ連動識別子_反映グループ名_SQL 分割番号.trc1 hdssqle_データ連動識別子_反映グループ名_SQL 分割番号.trc2 (例：hdssqle_01_trngroup_1.trc1) SQL 分割番号については、反映グループ定義に having または hash の指定がない場合は、1 (固定) になります
反映 UOC プロセス※	uoc_データ連動識別子_反映グループ名.trc1 uoc_データ連動識別子_反映グループ名.trc2

プロセス名	稼働トレースファイル名
	(例：uoc_01_uoc001.trcl)

注※

反映グループ名に¥が含まれる場合、¥を@に置き換えます。

- **int_trc_output_path = 稼働トレースファイル出力先**

プロセス単位収集方式の稼働トレースファイルの出力先ディレクトリを絶対パス名または相対パス名で指定します。このオペランドは、Windows 版 Datareplicator の場合だけ指定できます。相対パス名で指定した場合は、%HDSPATH%からの相対パスです。指定できる長さは、絶対パス名で 125 バイトまでとなります。125 バイトを超えた場合、KFRB00916-E メッセージを出力し、起動処理を停止します。

このオペランドは、int_trc_unit オペランドに process を指定した場合だけ有効です。

このオペランドで指定した稼働トレースファイル出力先ディレクトリは、Datareplicator 起動までにユーザが作成してください。その際、Datareplicator 管理者には read 権限と write 権限を付与してください。Datareplicator 起動時に指定したディレクトリがない場合、KFRB00865-E メッセージを出力し、起動処理を中止します。

このオペランドを省略すると、%HDSPATH%/trace を仮定します。該当するディレクトリがない場合、反映マスタプロセス起動時に Datareplicator がディレクトリを作成します。

稼働トレースファイルの合計最大サイズは次の式のとおりとなります。稼働トレースファイル出力先には、この値以上の空き容量のあるディレクトリを指定してください。

$$(int_trc_filesz \text{ オペランド指定値} \times (\text{データ連動識別子数} + \text{全反映グループ数} + 1) \times 2) \times 2$$

- **use_convertlib = true | false**

コード変換を使って文字コードを変換するかどうかを指定します。

true

コード変換を使って文字コードを変換します。抽出側 DB で次に示す文字コードを使用している場合にだけ有効になります。

- EBCDIK/KEIS78
- EBCDIK/KEIS83
- SJIS

コード変換がインストールされていない場合、true を指定しても無効になります。

false

コード変換を使わないで文字コードを変換します。この場合、外字は変換されないで空白になります。外字を変換する場合には、hdscnvedt コマンドで外字の変換方法を変更してください。

hdscnvedt コマンドについては、「7. コマンドの文法」の [hdscnvedt](#) コマンドを参照してください。

use_convertlib オペランドは、Windows 版 Datareplicator のときにだけ指定できます。UNIX 版 Datareplicator のときは、指定しないでください。

- **ref_wait_interval = 反映プロセスの反映情報キューファイル読み込み間隔**

～ 〈符号なし整数〉 ((100~60000)) 《5000》 (単位：ミリ秒)

反映プロセスが、反映情報キューファイルの終端を検知してから次の読み込みを再開するまでの間隔を指定します。

このオペランドの値を小さくすれば、反映情報キューファイルの終端を検知した後の待ち時間が短くなるので、反映処理の即時性を向上させることができます。

ただし、値を小さくすると CPU の利用率が高くなるため、逆に性能が悪くなる可能性があります。このため、更新処理の発生頻度や CPU の性能を考慮した上で値を指定する必要があります。

- `commit_wait_time = COMMIT 発行間隔`

～ 〈符号なし整数〉 ((0~300)) 《30》 (単位：秒)

反映プロセスが反映情報キューファイルの終端を検知してから、反映側 HiRDB に対して COMMIT を発行するまでの間隔を指定します。

0 を指定した場合は、反映情報キューファイルの終端を検知した時点で COMMIT を発行します。

このオペランドを指定又は省略すると、`ref_wait_interval` オペランドで指定した間隔で反映情報キューファイルを読み込みながら、このオペランドの指定値を超えるまで COMMIT の発行を保留し、指定値を超えた時点で COMMIT を発行します。

したがって、`ref_wait_interval` オペランドの指定値によっては、このオペランドで指定したとおりに COMMIT が発行されない場合があります。

なお、前回の COMMIT を発行してから反映情報キューファイルの終端を検知するまでに、反映情報キューファイルに更新情報が格納されなかった場合は、このオペランドの指定値を超えても COMMIT を発行しません。

このオペランドの指定値を大きくすると、反映情報キューファイルの終端を検知してから COMMIT を発行するまでの時間が長くなるため、無排他検索時を除き、データを参照できる時間が短くなります。

このオペランドの指定値を小さくすると、COMMIT を発行するまでの時間が短くなるため、データを参照できる時間が長くなりますが、送信間隔の指定によっては、COMMIT を発行した直後に次のデータを受信して反映しなければならなくなります。その結果、COMMIT を何回も発行することになり、スループットが低下する可能性があります。

したがって、このオペランドを指定する場合は、送信間隔が 1 分以内のときは、送信間隔の半分の値、送信間隔が 1 分以上のときは、デフォルト値を指定することをお勧めします。送信間隔が 0 の場合は、実際の送信間隔の平均値を考慮して指定してください。

なお、このオペランドの指定値を大きくする場合は、HiRDB クライアントの時間監視についての環境変数の値も、このオペランドの指定値より大きくなるように指定してください。

このオペランドの指定値が、`discintvl` オペランドの指定値より大きい場合は、`discintvl` オペランドの指定値に達した時点で COMMIT が発行されるので、このオペランドを指定する意味がなくなります。したがって、このオペランドを指定する場合は、`discintvl` オペランドの指定値より小さい値を指定してください。

ただし、このオペランドの指定値が `discintvl` オペランドの指定値より大きくても、エラーチェックはされません。

- `file_dupenv = 二重化定義ファイル名`

～ 〈1~125 バイトのファイル名〉

二重化定義ファイル名を絶対パス名、又は相対パス名で指定します。相対パス指定の場合は、反映側 Datareplicator 運用ディレクトリからの相対パスとします。

file_dupenv オペランドを省略した場合、二重化機能を使用しないものとして動作します。

- syncgroup001 =同期反映グループ名、データ連動識別子 [{, データ連動識別子}] …

反映トランザクション同期機能を使用する場合に指定します。

このオペランドを指定する場合は、commitment_method オペランドに fxa_sqlc を指定（反映処理の同期点処理方式を 2 相コミット方式に指定）してください。

同期反映グループ名

～ 〈1～8 文字の記号名称〉

同期反映グループに付与する名称を指定します。

データ連動識別子

dsidxxx オペランドに指定したデータ連動識別子を指定します。dsidxxx オペランドにないデータ連動識別子、又は**を指定した場合は、エラーになります。

データ連動識別子は重複して指定できません。

一つの同期反映グループで指定できるデータ連動識別子の最大数は、UNIX 版 Datareplicator では 128、Windows 版 Datareplicator では 63 です。また、Windows 版 Datareplicator では、dsid064～dsid128 オペランドは指定できません。これらのオペランドを指定するとエラーになります。

このオペランドに指定するデータ連動識別子について、一部のオペランドの指定に制限があります。詳細は、「3.6.5 同期反映グループ」を参照してください。

- syncgrp_discintvl = disconnect 発行待ち時間

〈符号なし整数〉 ((0～65535)) 《180》 (単位：秒)

反映トランザクション同期機能使用時に、disconnect を発行するまでの待ち時間を指定します。

commit が発行されてから指定した待ち時間が経過しても反映対象の更新情報がない場合、反映側 HiRDB に対して disconnect を発行します。

0 を指定した場合は、反映処理は更新情報の終端を検知しても、反映側 HiRDB に対して disconnect を発行しません。

このオペランドは、同期反映グループに指定されているデータ連動識別子に対してだけ有効となります。

- syncwait_limit_tran_count =同期待ち限界トランザクション数

〈符号なし整数〉 ((2～65535)) 《4096》

同期反映グループに指定したデータ連動識別子の各反映プロセスが、1 反映トランザクションとして処理できる抽出トランザクションの件数を指定します。

このオペランドに指定する値は、反映トランザクションがロールバックしても反映側 HiRDB のシステムログファイルに収まる範囲で、かつ反映トランザクションが消費する排他資源数が反映側 HiRDB の排他制御用プールサイズ内に収まる範囲とする必要があります。

オペランド指定値の範囲内に同期イベントが実行されない場合、KFRB03303-E メッセージが出力されて、同期反映処理がエラー停止します。

- syncwait_limit_time = 同期待ち限界時間

〈符号なし整数〉 ((1~65535)) 《180》 (単位：秒)

同期反映グループに指定したデータ連動識別子の各反映プロセスが、反映トランザクション処理中に同期イベントを検知する前に反映情報キューファイルの終端を検知してから、次の更新情報を受信するまでの限界時間を指定します。

このオペランドに指定する値は、抽出側 HiRDB で実行する hdeevent コマンドの実行間隔より長い時間を指定する必要があります。

オペランド指定値の範囲内に同期イベントが実行されない場合、KFRB03303-E メッセージが出力されて、同期反映処理がエラー停止します。

- reflect_counter_reset= true | false

反映側システム起動時に反映処理数をリセットするかどうかを指定します。

true

反映側システム起動時に反映処理数をリセットします。

false

反映側システム起動時に反映処理数をリセットしません。

注意

反映側 Datareplicator 再起動時の反映方式が前回終了時の反映方式と異なっている場合、false を指定してもリセットされます。前回終了時の反映方式と反映側システム再起動時の反映方式、及びリセット有無の関係について次に示します。

前回終了時の反映方式	再起動時の反映方式	リセットの有無
トランザクション単位	トランザクション単位	×
	表単位	○
表単位	トランザクション単位	○
	表単位	×※

(凡例)

○：リセットされます。

×：リセットされません。

注※ 前回終了時と Datareplicator 再起動時で、次の条件が異なる場合、反映処理件数がリセットされます。

- ・ 反映グループ名
- ・ SQL プロセス数
- ・ 分割方式 (キーレンジ分割, ハッシュ分割)
- ・ キーレンジ分割の場合, キーレンジグループ数
- ・ ハッシュ分割の場合, ハッシュ分割表の RD エリア数

また、再起動時以外に反映処理件数がリセットされる条件を次に示します。

- ・ イベントによって反映方式を切り替えた場合

- ・ 定義情報を変更して、反映プロセスを再起動した場合

5.9 反映環境定義

反映処理を実行するために必要な情報を定義します。複数のデータ連動識別子をデータ連動の対象とする場合、反映処理はデータ連動識別子ごとに実行するため、反映環境定義もデータ連動識別子ごとに必要になります。

5.9.1 形式

```
set qufile001 = "反映情報キューファイル名"
set qufile002 = "反映情報キューファイル名"
    [ … [ set qufile008 = "反映情報キューファイル名" ] ]
set queuesize = 反映情報キューファイルサイズ
[ set reffile = "反映定義ファイル名" ]
set statsfile = "反映ステータスファイル名"
set statssize = 反映ステータスファイルサイズ
set unreffile1 = "未反映情報ファイル名 (プライマリ) "
set unreffile2 = "未反映情報ファイル名 (セカンダリ) "
[ set unreffilesz = 未反映情報ファイルサイズ ]
[ set startmode = { trn | tbl | spd } ]
[ set restartmode = initial | continue ]
[ set breaktime = hh:mm ]
[ set breakmode = trn | tbl ]
[ set event_maxentry = イベントの最大数 ]
[ set eventtrn = トランザクション単位反映イベントコード ]
[ set eventtbl = 表単位反映イベントコード ]
[ set eventretrn = トランザクション単位反映再起動イベントコード ]
[ set eventretbl = 表単位反映再起動イベントコード ]
[ set eventspd = 反映処理停止イベントコード ]
```

```

[ set eventcntreset = 反映処理数リセットイベントコード ]
[ set defmerge = true | false ]
[ set cmtintvl = 反映処理コミット間隔 ]
[ set trncmtintvl = トランザクション単位反映方式での反映処理コミット間隔 ]
[ set tblcmtintvl = 表単位反映方式での反映処理コミット間隔 ]
[ set tblcheck = true | false ]
set defshmsize = 定義情報格納用共有メモリサイズ
[ set ebcdic_type = { eck78 | ekk78 | eck83 | ekk83 } ]
[ set shiftspace_cnv = multi | single ]
[ set undefcode_cnv = multi | single ]
[ set ref_data_backspace = suppress | nosuppress ]
[ set skip_sqlcode = SQLCODE [, SQLCODE ] ... ]
[ set skip_mvcelmwarn = true | false ]
[ set sqlerr_during_retry = stop | enable_skip ]
[ set sqlerr_skip_info = { output | msgoutput | sqloutput | nooutput } ]
[ set extract_init = check | nocheck ]
[ set db_connect_retry_number = DB コネクトリトライ回数 ]
[ set db_connect_retry_interval = DB コネクトリトライ間隔 ]
[ set skip_codecnv_error = true | false ]
[ set int_trc_getl = 稼働トレース取得範囲 ]
[ set device01 = Datareplicator ファイルシステム領域名 [[, 割り当てファイル種別] ...]
[ set device02 = Datareplicator ファイルシステム領域名 [[, 割り当てファイル種別] ...]
    [ ... [ set device09 = Datareplicator ファイルシステム領域名 [[, 割り当てファイル種別]
...]]]]]
[ set ujcodekind = rcv | sam ]
[ set discintvl = disconnect 発行間隔 ]

```

[set ref_wait_interval = 反映プロセスの反映情報キューファイル読み込み間隔]

[set commit_wait_time = COMMIT 発行間隔]

[set mapping_key_check = { not_null_unique | unique | none }]

[set control_trigger = execute | not_execute]

[set control_reference_trigger = execute | not_execute]

[set check_pending = use | nouse]

[set reflect_delay_limit_time = 反映滞留時間しきい値]

[set sql_lockerr_retrynum = 排他エラー発生時のトランザクションリトライ回数]

[set xa_recovery_retry_count = トランザクション回復要求のリトライ回数]

[set xa_recovery_retry_interval = トランザクション回復要求のリトライ間隔]

[set reflect_trn_max_sqlnum = 反映トランザクション内最大更新 SQL 数]

5.9.2 定義内容の変更

定義内容を変更する場合には、次のように操作してください。

1. 反映処理、又は反映側 Datareplicator を終了します。
初期開始をする場合で、反映側 Datareplicator が異常終了しているときには、一度正常開始してから定義内容を変更してください。
2. テキストエディタで定義内容を変更します。
3. 次の表に示す、変更後の定義内容が有効になる開始モードで開始します。

表 5-18 反映環境定義の変更後の定義内容が有効になる開始モード

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始又は部分初期開始※1	正常開始※1	反映処理の開始※2
qufilexxx※3	○	—	—
queuesize	○	—	—
reffile	○	○	○
statsfile※4	○	—	—
statssize	○	—	—
unreffile1	○	—	—

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始又は部分初期開始※1	正常開始※1	反映処理の開始※2
unreffile2	○	—	—
unreffilesz	○	○	—
startmode※5	○	○	—
restartmode※5	○	○	—
breaktime※5	○	○	—
breakmode※5	○	○	—
event_maxentry	○	—	—
eventtrn	○	—	—
eventtbl	○	—	—
eventretrn	○	—	—
eventretbl	○	—	—
eventcntreset	○	○	—
eventspd	○	—	—
defmerge	○	○	○
cmtintvl	○	○	○
trncmtintvl	○	○	○
tblcmtintvl	○	○	○
tblcheck	○	○	○
defshmsize	○	○	○
ebcdic_type	○	○	○
shiftspace_cnv	○	○	○
undefcode_cnv	○	○	○
ref_data_backspace	○	○	○
skip_sqlcode	○	○	○
skip_mvcelmwarn	○	○	○
sqlerr_during_retry	○	○	○
sqlerr_skip_info	○	○	○
extract_init	○	○	—
db_connect_retry_number	○	○	○

オペランド名	変更後の定義内容が有効になる開始モード		
	初期開始又は部分初期開始※1	正常開始※1	反映処理の開始※2
db_connect_retry_interval	○	○	○
skip_codecnv_error	○	○	○
int_trc_getl	○	○	—
devicexx※6	○	—	—
ujcodekind	○	—	—
discintvl	○	○	○
ref_wait_interval	○	○	○
commit_wait_time	○	○	○
mapping_key_check	○	○	○
control_trigger	○	—	—
control_reference_trigger	○	—	—
check_pending	○	—	—
reflect_delay_limit_time	○	○	○
sql_lockerr_retrinum	○	○	○
xa_recovery_retry_count	○	○	○
xa_recovery_retry_interval	○	○	○
reflect_trn_max_sqlnum	○	○	○

(凡例)

○：対応する動作によって変更内容が有効になることを示します。複数の動作に対して記載されている場合は、どれか一つの動作を実行することで変更内容が有効になることを示します。

—：該当しません。

注※1

反映側 Datareplicator を起動するときの開始モードです。

注※2

反映側 Datareplicator の反映処理を起動するときの開始モードです。

注※3

xxx には、001～008 が対応します。

注※4

正常開始する場合は、statsfile オペランドを変更しないでください。

注※5

前回のエラーで終了している場合、又は強制終了モードで終了している場合には、抽出側 DB と反映側 DB の整合性を保つため、オペランドの設定内容は前回の終了時の状態を引き継ぎます。

注※6

xx には、01～09 が対応します。

5.9.3 オペランドの説明

- qufile001 = "反映情報キューファイル名"
… qufile008 = "反映情報キューファイル名"
～ 〈1～64 バイトの [パス名/] ファイル名〉

qufilexxx には、反映情報キューファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDSPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが 125 バイト以下になるように指定してください。

反映情報キューファイルはファイル単位のスワップ方式で情報が格納されるため、2 個以上必要で、8 個まで指定できます。最低でも qufile001 と qufile002 は指定してください。

qufile001～qufile008 に指定する 001～008 は、001 から昇順に連続して指定してください。昇順に連続して指定していない場合には、正しい順序で指定されている反映情報キューファイルまでを反映の対象にします。

ファイル名は、反映側システムで一意にしてください。

- queuesize = 反映情報キューファイルサイズ
～ 〈符号なし整数〉 ((2～1000000000)) (単位：KB)

qufile001～qufile008 オペランドで定義した反映情報キューファイルのサイズを指定します。queuesize オペランドで指定したファイルサイズは、qufile001～qufile008 オペランドで指定したすべてのファイルの、一つ当たりのサイズになります。

反映情報キューファイルの見積もり式については、「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。

また、反映情報キューファイルをラージファイルとして運用する場合、2097152 (2GB) 以上に指定します。ラージファイルとして運用する場合は、OS や Datareplicator ファイルシステム領域を、ラージファイルに対応するように事前に設定しておく必要があります。詳細については、「[6.11 ラージファイルの運用](#)」を参照してください。

注意

既存の反映情報キューファイルのサイズを変更した場合、反映側 Datareplicator を初期化する必要があるため、レプリケーションが完了していることを事前に確認してから、ファイルサイズを変更してください。

- reffile = "反映定義ファイル名"
～ 〈1～64 バイトの [パス名/] ファイル名〉

反映処理の内容を定義した反映定義ファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDSPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが125バイト以下になるように指定してください。

抽出対象表と反映対象表の形式、表名、及び列名がすべて同一の場合には、reffile オペランドを省略できます。ファイル名は、反映側システムで一意にしてください。

- **statsfile = "反映ステータスファイル名"**

～ 〈1～64 バイトの [パス名/] ファイル名〉

障害発生時の回復処理の実行に必要な情報が格納されるステータスファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDSPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが125バイト以下になるように指定してください。ファイル名は、反映側システムで一意にしてください。

なお、ステータスファイルの名称を変更する場合に、変更するステータスファイルと同じ名称のファイルが既に存在しているときは、反映側 Datareplicator を `hdsstart -i -f` コマンドで初期開始してください。

- **statssize = 反映ステータスファイルサイズ**

～ 〈符号なし整数〉 ((72～2000000)) (単位:KB)

反映ステータスファイルのサイズを指定します。反映ステータスファイルのサイズについては、「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。

- **unreffile1 = "未反映情報ファイル名 (プライマリ) "**

unreffile2 = "未反映情報ファイル名 (セカンダリ) "

～ 〈1～64 バイトの [パス名/] ファイル名〉

未反映情報ファイルの名称を、絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDSPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが125バイト以下になるように指定してください。

未反映情報ファイルにはプライマリファイル (unreffile1) とセカンダリファイル (unreffile2) があり、ファイル単位のスワップ方式で情報が蓄積されます。

ファイル名は、反映側システムで一意にしてください。

- **unreffileesz = 未反映情報ファイルサイズ**

～ 〈符号なし整数〉 ((1～2097151)) 《16》 (単位:KB)

未反映情報ファイルの上限サイズを指定します。unreffileesz オペランドで指定するサイズは、unreffile1 と unreffile2 の両方に対応します。

未反映情報ファイルのサイズについては、「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。

- **startmode = { trn | tbl | spd }**

反映側 Datareplicator 起動時の反映方式を指定します。

startmode オペランドの指定は、前回の終了モードが正常終了、イベント終了、又は即時終了の場合にだけ有効です。hdsstop -t force コマンドによる強制終了や、反映処理のエラーによる終了の場合、終了前の反映方式が引き継がれます。

trn

トランザクション単位反映方式で反映処理を開始します。

tbl

表単位反映方式（表単位分割方式、キーレンジ単位分割方式、ハッシュ分割方式）で反映処理を開始します。tbl を選択すると、各表の反映グループごとにトランザクションが発生します。そのため、抽出側では複数の表を一つのトランザクションで更新した場合でも、反映側ではそれぞれ別のトランザクションで更新されることになり同期が保証されません。同期を取る必要がある表は、同一の反映グループに定義してください。

spd

反映側 Datareplicator の起動時に反映処理を停止したまま、受信処理だけを開始します。

• restartmode = initial | continue

反映側 Datareplicator 起動時に、前回終了時の反映処理稼働状態（反映処理が停止状態であるかないか）を引き継ぐかどうかを指定します。

hdsrftcl -m immediate コマンドによって反映処理を停止した場合、又は反映処理停止イベントの検知によって反映処理を停止した場合に、前回終了時の反映処理稼働状態を引き継ぎます。hdsstop コマンドによる終了や、反映処理のエラーによる終了の場合、前回終了時の反映処理稼働状態は引き継ぎません。

initial

前回終了時の反映処理稼働状態を引き継がないで、startmode オペランドの指定に従って反映処理を起動します。

continue

前回終了時の反映処理稼働状態を引き継いで起動します。

• breaktime = hh:mm

反映側 Datareplicator の起動時から停止状態（startmode オペランドの設定が spd の場合）にある反映処理を開始する時間を、反映側 Datareplicator の起動時を起点とした相対時間で指定します。

00:00~23:59 以外の時間を指定すると、オペランドの設定エラーになります。breaktime オペランドを省略、又は「breaktime = 00:00」を指定すると、hdsrftcl コマンドによって反映処理が開始されるまで停止したままになります。

hh ~ 〈符号なし整数〉 ((00~23))

mm ~ 〈符号なし整数〉 ((00~59))

hh 又は mm の値が 1 けたの場合は、前方に 0 を指定する必要があります。例えば、1 時間 1 分後に反映処理を開始したい場合は、01:01 と指定します。

• breakmode = trn | tbl

反映側 Datareplicator の起動時から停止状態にある反映処理を開始する場合の反映方式を指定します。breakmode オペランドは、次の二つの条件が満たされている場合だけ有効です。この場合、breakmode オペランドを必ず指定してください。

- startmode オペランドの設定が spd です。

- breaktime オペランドの設定が有効です。

trn

トランザクション単位反映方式で反映処理を開始します。

tbl

表単位反映方式（表単位分割方式，キーレンジ単位分割方式，ハッシュ分割方式）で反映処理を開始します。

- event_maxentry = イベント同時管理数

～ 〈符号なし整数〉 ((16~1024)) 《16》

表単位反映方式で反映処理を行う場合に，処理中にイベントを検知したときに同時に管理するイベントの数を指定します。反映定義サーバプロセスは，1 秒間隔でイベントの完了を監視し，処理の完了したイベントに対して KFRB03211-I メッセージを出力します。イベント同時管理数が不足した場合，イベントに対する処理は行われますが，イベント処理完了を示すメッセージ KFRB03211-I が出力されないことがあります。

KFRB03211-I メッセージの出力を契機に何らかの業務を開始しているなど，KFRB03211-I メッセージの出力が必須となる場合は，イベント同時管理数が不足することがないように，1 秒間に発生するイベント数以上の値をこのパラメタに指定してください。

注意

1 秒間に検知したイベントの数が，ここで指定したイベント同時管理数より少ない場合でも，どれかの反映プロセスで処理が遅延していると，他の遅延していない反映プロセスは次の処理を進め次のイベントを検知します。そのため，管理中のイベント数が増加し，イベント同時管理数が不足することがあります。

- eventtrn = トランザクション単位反映イベントコード

～ 〈符号なし整数〉 ((1~128)) 《1》

トランザクション単位反映方式で反映処理を稼働するためのイベントコードを指定します。トランザクション単位反映イベントコードは，抽出側システムと対応付ける必要があります。

- eventtbl = 表単位反映イベントコード

～ 〈符号なし整数〉 ((1~128)) 《2》

表単位反映方式（表単位分割方式，キーレンジ単位分割方式，ハッシュ分割方式）で反映処理を稼働するためのイベントコードを指定します。表単位反映イベントコードは，抽出側システムと対応付ける必要があります。

- eventretrn = トランザクション単位反映再起動イベントコード

～ 〈符号なし整数〉 ((1~128)) 《3》

停止中の反映処理を，トランザクション単位反映方式で再起動するためのイベントコードを指定します。トランザクション単位反映再起動イベントコードは，抽出側システムと対応付ける必要があります。

- eventretbl = 表単位反映再起動イベントコード

～ 〈符号なし整数〉 ((1~128)) 《4》

停止中の反映処理を、表単位反映方式（表単位分割方式、キーレンジ単位分割方式、ハッシュ分割方式）で再開始するためのイベントコードを指定します。表単位反映再起動イベントコードは、抽出側システムと対応付ける必要があります。

- **eventspd = 反映処理停止イベントコード**

～ 〈符号なし整数〉 ((1~128)) 《5》

反映処理を停止させるためのイベントコードを指定します。反映処理停止イベントコードは、抽出側システムと対応付ける必要があります。

- **eventcntreset = 反映処理数リセットイベントコード**

～ 〈符号なし整数〉 ((1~128))

反映処理数をリセットするためのイベントコードを指定します。抽出側 Datareplicator と反映側 Datareplicator でリセットを同期させる場合、送信環境定義の eventcntreset オペランドと同じ値にする必要があります。

なお、このオペランドは反映環境定義の次のオペランドと同じ指定はできません（省略時仮定値を含む）。

- eventtrn オペランド
- eventtbl オペランド
- eventspd オペランド
- eventretm オペランド
- eventretbl オペランド

- **defmerge = true | false**

反映定義で定義されていない更新情報について、その更新情報が定義された抽出側システムの抽出定義を反映定義と仮定して、反映処理を実行するかどうかを指定します。defmerge オペランドは、反映定義がある場合に有効になります。

true

抽出定義に対応するすべての表に反映処理を実行します。抽出定義に対応する反映定義がない場合には、抽出定義を反映定義と仮定して反映を実行します。

false

反映定義で定義された表だけに反映処理を実行します。

- **cmtintvl = 反映処理コミット間隔**

～ 〈符号なし整数〉 ((1~32767)) 《100》

反映側 Datareplicator が反映側 HiRDB に対してコミットを発行する間隔を、抽出側システムでのトランザクション数で指定します。

次の場合には、cmtintvl オペランドの指定に関係なく、反映側 HiRDB に対してコミットが発行されます。

- イベントを検知した場合
- 反映処理で PURGE TABLE を実行した場合

cmtintvl, trncmtintvl, 及び tblcmtintvl オペランドの指定値を大きくすると、HiRDB サーバに対して影響があります。詳細については、「4.7.3(5) 反映処理の COMMIT 発行間隔の設計」を参照してください。

また、cmtintvl オペランドの指定値を大きくすると、反映側 Datareplicator の 1 トランザクションで実行する SQL 件数が増加するため、次に示す HiRDB のリソースが多く消費されます。

- pd_lck_pool_size
- pd_max_access_tables
- pd_log_sdinterval
- pd_sql_object_cache_size

排他リソースが不足しエラーが発生する場合は、cmtintvl オペランドの指定値を小さくしてください。

• trncmtintvl = トランザクション単位反映方式での反映処理コミット間隔

～ 〈符号なし整数〉 ((1~32767)) 《100》

トランザクション単位反映方式で反映処理を実行中に、反映側 Datareplicator が反映側 HiRDB に対してコミットを発行する間隔を、抽出側システムでのトランザクション数で指定します。

次の場合には trncmtintvl の指定に関係なく、反映側 HiRDB に対してコミットが発行されます。

- イベントを検知した場合
- 反映処理で PURGE TABLE を実行した場合

反映方式がトランザクション単位反映方式の場合、trncmtintvl オペランドの指定値は、cmtintvl オペランドの指定値よりも優先されます。

cmtintvl, trncmtintvl, 及び tblcmtintvl オペランドの指定値を大きくすると、HiRDB サーバに対して影響があります。詳細については、「4.7.3(5) 反映処理の COMMIT 発行間隔の設計」を参照してください。

また、trncmtintvl オペランドの指定値を大きくすると、反映側 Datareplicator の 1 トランザクションで実行する SQL 件数が増加するため、次に示す HiRDB のリソースが多く消費されます。

- pd_lck_pool_size
- pd_max_access_tables
- pd_log_sdinterval
- pd_sql_object_cache_size

排他リソースが不足しエラーが発生する場合は、trncmtintvl オペランドの指定値を小さくしてください。

• tblcmtintvl = 表単位反映方式での反映処理コミット間隔

～ 〈符号なし整数〉 ((1~32767)) 《100》

表単位反映方式で反映処理を実行中に、反映側 Datareplicator が反映側 HiRDB に対してコミットを発行する間隔を、抽出側システムでのトランザクション数で指定します。

次の場合には、tblcmtintvl の指定に関係なく、反映側 HiRDB に対してコミットが発行されます。

- イベントを検知した場合

- 反映処理で PURGE TABLE を実行した場合

反映方式が表単位反映方式の場合、tblcmtintvl オペランドの指定値は、cmtintvl オペランドの指定値よりも優先されます。

cmtintvl, trncmtintvl, 及び tblcmtintvl オペランドの指定値を大きくすると、HiRDB サーバに対して影響があります。詳細については、「[4.7.3\(5\) 反映処理の COMMIT 発行間隔の設計](#)」を参照してください。

また、tblcmtintvl オペランドの指定値を大きくすると、反映側 Datareplicator の 1 トランザクションで実行する SQL 件数が増加するため、次に示す HiRDB のリソースが多く消費されます。

- pd_lck_pool_size
- pd_max_access_tables
- pd_log_sdinterval
- pd_sql_object_cache_size

排他リソースが不足しエラーが発生する場合は、tblcmtintvl オペランドの指定値を小さくしてください。

- `tblcheck = true | false`

反映側 Datareplicator の起動時に、反映対象表が反映側システムにあるかどうかをチェックするために指定します。

反映定義がある場合は、tblcheck オペランドの設定に関係なく、反映定義で指定した反映対象表が、反映側システムにあるかどうかをチェックします。反映定義を省略している場合には、tblcheck オペランドを true と指定しているときだけ、抽出側システムから送信された抽出定義に定義されている抽出対象表が、反映側システムにあるかどうかをチェックします。

true

反映対象表が反映側システムにあるかどうかをチェックします。true を指定すると、反映側 Datareplicator の起動時に反映側 HiRDB に負荷が掛かる可能性があります。

false

チェックしません。

- `defshmsize = 定義情報格納用共有メモリサイズ`
～ 〈符号なし整数〉 ((1~2000000)) (単位:KB)

UNIX 版 Datareplicator の場合

定義情報の解析結果を格納する共有メモリのサイズを指定します。共有メモリのサイズについては、「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。

Windows 版 Datareplicator の場合

定義情報の解析結果を格納するメモリマップトファイルの容量を指定します。メモリマップトファイルの容量については、「[4.7.7 反映側 Datareplicator のリソースの設計](#)」を参照してください。

- `ebcdic_type = { eck78 | ekk78 | eck83 | ekk83 }`

抽出側システムの文字コード体系が EBCDIK/KEIS の場合、又は抽出側 Datareplicator が EBCDIK/KEIS に変換して送信する場合に、送信されるデータの EBCDIK/KEIS の種別を指定します。

ebcdic_type オペランドは、送信されたデータの文字コード体系が EBCDIK/KEIS の場合だけ有効になります。

eck78

EBCDIC/KEIS78 の場合に指定します。

ekk78

EBCDIK/KEIS78 の場合に指定します。

eck83

EBCDIC/KEIS83 の場合に指定します。

ekk83

EBCDIK/KEIS83 の場合に指定します。

- **shiftspace_cnv = multi | single**

全角モード中に、二つの連続した半角の空白文字が存在する場合に変換する、空白文字の種別（全角又は半角）を指定します。

shiftspace_cnv オペランドは、送信されたデータの文字コード体系が EBCDIK/KEIS の場合だけ有効になります。

multi

二つの連続した半角の空白文字を、一つの全角の空白文字に変換します。

single

二つの連続した半角の空白文字を、二つの連続した半角の空白文字に変換します。

- **undefcode_cnv = multi | single**

定義されていない外字を検出した場合に変換する、空白文字の種別（全角又は半角）を指定します。

multi

定義されていない外字を、一つの全角の空白文字に変換します。

single

定義されていない外字を、二つの連続した半角の空白文字に変換します。

- **ref_data_backspace = suppress | nosuppress**

次の条件をすべて満たす場合は、suppress を指定してください。

suppress を指定しないで、抽出側と反映側のシステムの定義長を同じにしている場合には、SQL 発行時にエラーとなります。

抽出側 DB の文字コード体系と反映側 DB の文字コード体系によって、SQL 発行時にエラーとなる条件を次に示します。

抽出ロケール	反映ロケール			
	EBCDIK(C)/KEIS	SJIS	EUC	UTF-8
EBCDIK(C)/KEIS	—	○	×	×
SJIS	—	—	• A1 かつ B2	• A1 かつ B3

抽出ロケール	反映ロケール			
	EBCDIK(C)/KEIS	SJIS	EUC	UTF-8
			• A2 かつ B1	• A2 かつ B3 • A3 かつ B3
EUC	—	○	—	
UTF-8	—	○	○	—

(凡例)

○：SQL 発行時にエラーになりません。

×：条件によって、SQL 発行時にエラーとなります。

A1：反映側 DB のデータに半角カナを含む

A2：反映側 DB のデータに外字を含む

A3：反映側 DB のデータに ASCII コード以外を含む

B1：反映側 DB の列定義長が抽出側の 1.5 倍 (3/2) 以下

B2：反映側 DB の列定義長が抽出側の 2 倍 (2/1) 以下

B3：反映側 DB の列定義長が抽出側の 3 倍 (3/1) 以下

suppress を指定すると、文字コードの変換後に空白文字の削除処理が実行されるため、CPU 処理時間が増加する可能性があります。このため、反映側 DB の文字コード体系が EUC 及び UTF-8 以外の場合は、できるだけ nosuppress を指定することを推奨します。すべての空白文字の場合には、1 文字分の空白文字と仮定して反映処理を実行します。

suppress

文字列の後ろの空白文字を取り除きます。

nosuppress

文字列の後ろの空白文字を取り除きません。

• skip_sqlcode = SQLCODE [, SQLCODE] …

反映処理中に DB の更新で SQL エラーが発生した場合、その SQL エラーが発生した更新情報をスキップさせる対象の SQLCODE を指定します。

スキップできるのは、次の二つの SQL エラーです。

- 更新実行時 (EXECUTE) に発生した SQL エラー
- INSERT, 又は UPDATE の前処理時 (PREPARE) に発生した SQL エラー

ただし、バージョン 05-00 よりも前の Datareplicator では、SQL の前処理時 (PREPARE) に発生した SQL エラーはスキップできません。

SQLCODE は、最大 32 個指定できます。

負の SQLCODE を指定するときは、マイナス符号を付けて指定します。ただし、暗黙的にロールバックされる SQL エラーに該当する SQLCODE を指定しても、スキップ指定は無視されて反映処理は停止します。

また、HiRDB の暗黙的なロールバック発生後の反映処理のリトライ中に発生した SQL エラーについては、sqlerr_during_retry オペランドの指定によってスキップ指定が無視されて反映処理が停止することがあります。

このオペランドにはデフォルト値はありません。オペランドを省略すると、SQL エラー発生時に SQLCODE の内容に関係なく、反映処理は停止します。

SQLCODE については、マニュアル「HiRDB UAP 開発ガイド」を参照してください。

- skip_mvcelmwarn = true | false

反映処理中に、繰返し列を含む表に対する UPDATE を実行した場合に、SQLWARN7 に「W」が設定された（UPDATE の SET 句 又は DELETE 句で、更新する行に存在しない要素番号を指定した）ときの Datareplicator の処理を指定します。

true

SQLWARN7 をスキップして、処理を続けます。

false

エラーとして反映処理を停止します。

- sqlerr_during_retry = stop | enable_skip

反映処理中に通信エラー（SQLCODE が-722, 又は-723）が発生した場合、その後の反映処理リトライ中に SQL エラーが発生したときに、skip_sqlcode オペランドの指定値を有効とするかどうかを指定します。

HiRDB が暗黙的なロールバックを行う SQL エラーのうち、通信エラー（SQLCODE が-722, 又は-723）が発生した場合、その後の反映処理リトライ中に発生した SQL エラーをスキップすると、抽出側 DB と反映側 DB の不整合が発生するおそれがあります。これは、HiRDB の暗黙的なロールバック処理と反映処理のリトライが競合するためです。

SQL エラーをスキップさせないために、skip_sqlcode オペランドを指定する場合は、sqlerr_during_retry オペランドに stop を指定、又は指定を省略してください。これによって、抽出側 DB と反映側 DB の不整合が発生するおそれがあるときは、スキップ指定が無視されて、反映処理は KFRB03061-E メッセージを出力して停止します。KFRB03061-E メッセージを出力して停止したときは、HiRDB の暗黙的なロールバック処理が完了するのを待った後、反映処理を再起動してください。sqlerr_during_retry の指定と、通信エラー後の反映処理リトライ中に発生した SQL エラーのスキップの関係を次の表に示します。

sqlerr_during_retry の指定	通信エラー後の反映処理リトライ中に発生した SQL エラーの SQLCODE	スキップの有無
stop 又は省略	skip_sqlcode オペランドに指定した SQLCODE	×※
	上記以外の SQLCODE	×
enable_skip	skip_sqlcode オペランドに指定した SQLCODE	○
	上記以外の SQLCODE	×

(凡例)

○：スキップされます。

×：スキップされません。

注※

KFRB03061-E メッセージを出力します。

stop

skip_sqlcode の指定を無効とし、反映処理を停止します。

enable_skip

skip_sqlcode の指定を有効とし、skip_sqlcode オペランドの指定に従います。

- sqlerr_skip_info = { output | msgoutput | sqloutput | nooutput }

次に示すオペランドの指定で、スキップ又はエラーを知らせる警告メッセージを出力するかどうか、及びその SQL エラーを未反映情報を出力するかどうかを指定します。

- skip_sqlcode オペランドに指定した SQLCODE の更新情報の反映をスキップ
- skip_mvcelmwarn オペランドに true を指定 (SQLWARN7 に「W」が設定されたときにスキップして処理を続ける)

このオペランドは、skip_sqlcode を指定したとき 又は skip_mvcelmwarn オペランドに true を指定したときにだけ有効です。

output

警告メッセージと未反映情報の両方を出力します。

msgoutput

警告メッセージだけを出力します。

sqloutput

未反映情報だけを出力します。

nooutput

警告メッセージも未反映情報も出力しません。

- extract_init = check | nocheck

初期化直後の抽出側システムと接続するときに、更新情報の受信シーケンスをチェックするかどうかを指定します。

check

初期化直後の抽出側システムと接続するときに、反映側システムも初期化されているかどうかをチェックします。

nocheck

初期化直後の抽出側システムと接続するときに、無条件に接続を許可します。

- db_connect_retry_number = DB コネクトリトライ回数

～ 〈0～255〉《0》(単位：回数)

反映側 Datareplicator がターゲット DB にコネクトしようとしてエラーになった場合の、コネクトリトライ回数を指定します。

- `db_connect_retry_interval` = DB コネクトリトライ間隔
～ 〈0～3600〉《30》(単位：秒)

反映側 Datareplicator がターゲット DB に対してコネクトリトライする場合のリトライ間隔を、秒単位で指定します。

このオペランドは、`db_connect_retry_number` に 1 以上の値を指定したときにだけ有効です。

- `skip_codecvt_error` = `true` | `false`

反映処理中に抽出データの文字コード変換でエラーになったときに、反映処理を停止するか、その抽出データを含む更新情報をスキップして反映処理を続けるかどうかを指定します。

`true`

エラーになったときに、抽出データを含む更新情報をスキップして反映処理を続けます。

`false`

エラーになったときに、反映処理を停止します。

- `int_trc_getl` = 稼働トレース取得範囲

このシステムのデータ連動識別子ごとに取得する稼働トレースの取得範囲を指定します。

このオペランドは、反映システム定義の `int_trc_lvl` オペランドの第 2 パラメタに `nc` を指定したときだけ有効になります。

- 稼働トレース取得範囲に指定する値

指定する値	取得する情報			
	MST (制御)	RCV (受信)	RFC (反映)	SQE (SQL 実行)
c1	○	○	○	×
c2	○	○	×	○
c3	○	○	×	×
c4	○	×	○	○
c5	○	×	○	×
c6	○	×	×	○
c7	○	×	×	×
稼働トレース取得範囲を省略	○	×	×	×

(凡例)

○：採取します。

×：採取しません。

- `device01` = Datareplicator ファイルシステム領域名 [(, 割り当てファイル種別) …]
～ 〈1～125 バイトの絶対パス名〉
- `device02` = Datareplicator ファイルシステム領域名 [(, 割り当てファイル種別) …]
～ 〈1～125 バイトの絶対パス名〉

:

device09 = Datareplicator ファイルシステム領域名 [(, 割り当てファイル種別) …]

～ 〈1~125 バイトの絶対パス名〉

devicexx オペランドには、Datareplicator ファイルシステム領域とその領域に指定した割り当てファイル種別を指定します。2けたの数字部分 (xx) は、必ず通し番号にしてください。割り当てファイル種別とファイルの対応を次に示します。

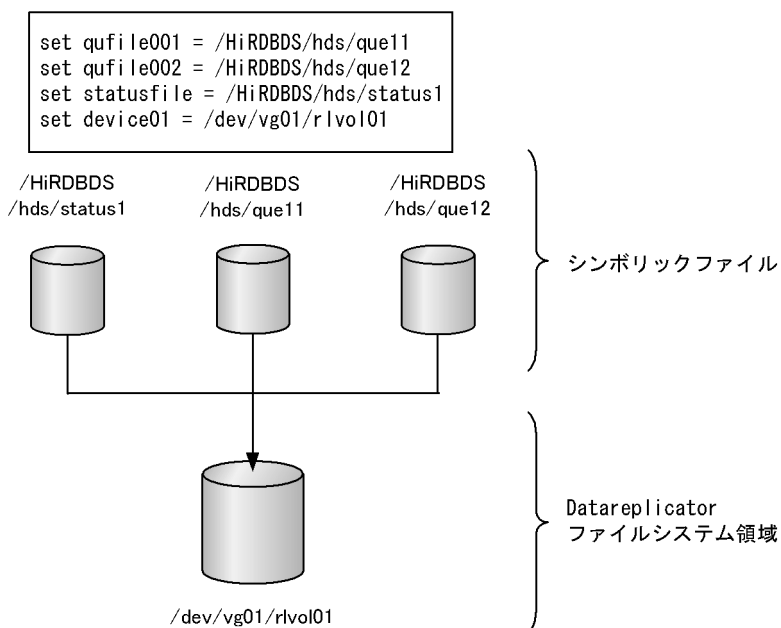
割り当てファイル種別	各オペランドに指定するファイル名
qufilexxx (xxx は 001~008 までの整数)	set qufilexxx で指定した反映情報キューファイル用リンクファイル名
statsfile	set statsfile で指定した反映ステータスファイル用リンクファイル名

割り当てファイル種別を省略すると、該当する dsid のすべてのファイルを割り当て対象にします。

devicexx オペランドで Datareplicator ファイルシステム領域として割り当てたファイルは、自動的に Datareplicator ファイルシステムにリンクするファイルとして作成されます。割り当てファイル種別に対応するファイルの名称は、既にキャラクタ型スペシャルファイルとして割り当てられていないようにしてください。

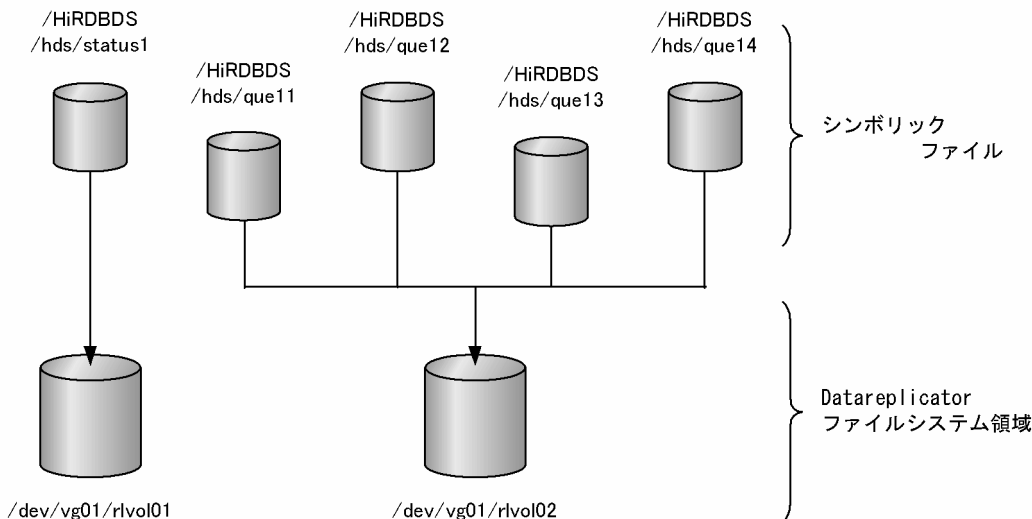
devicexx オペランドの定義例を次に示します。

● 定義例1：シングルサーバの場合



● 定義例2 : パラレルサーバの場合

```
set qufile001 = /HiRDBS/hds/que11
set qufile002 = /HiRDBS/hds/que12
set qufile003 = /HiRDBS/hds/que13
set qufile004 = /HiRDBS/hds/que14
set statusfile = /HiRDBS/hds/status1
set device01 = /dev/vg01/rlvol01, statusfile
set device02 = /dev/vg01/rlvol02, qufile001, qufile002, qufile003, qufile004
```



注意

devicexx オペランドを指定して Datareplicator を運用した後で、devicexx オペランドを定義から省略するときは、次に示す手順で作業を実施してください。

1. Datareplicator を停止します。
2. devicexx オペランドを反映環境定義から削除します。
3. 削除した devicexx オペランドに指定していた Datareplicator ファイルシステム領域を、hdsfmkfs コマンドで初期化します。
4. Datareplicator を初期開始します。

上記の手順を踏まないで devicexx オペランドを定義から削除して Datareplicator を初期開始すると、実行時にファイル不正でエラーになることがあります。

• ucodekind = rcv | sam

データ連動識別子を通常の受信機能で使うか、更新情報入力として使うかを指定します。SAM ファイルを使うメインフレーム側 DB (PDMII E2, RDB1 E2) とデータ連動する反映側 Datareplicator では、ucodekind オペランドに sam を指定します。

rcv

通常の Datareplicator の受信機能で使います。

sam

更新情報入力として使います。

- **discintvl = disconnect 発行間隔**

～ 〈符号なし整数〉 ((0~65535)) 《180》 (単位：秒)

このオペランドの意味については、反映システム定義の discintvl オペランドを参照してください。

このオペランドを指定した場合は、反映システム定義の discintvl オペランドの指定値よりも優先されます。

このオペランドを省略すると、反映システム定義の discintvl オペランドの指定値が有効になります。

反映システム定義の discintvl オペランドも省略すると、反映システム定義の discintvl オペランドのデフォルト値が仮定されます。

- **ref_wait_interval = 反映プロセスの反映情報キューファイル読み込み間隔**

～ 〈符号なし整数〉 ((100~60000)) 《5000》 (単位：ミリ秒)

このオペランドの意味については、反映システム定義の ref_wait_interval オペランドを参照してください。

このオペランドを指定した場合は、反映システム定義の ref_wait_interval オペランドの指定値よりも優先されます。

このオペランドを省略すると、反映システム定義の ref_wait_interval オペランドの指定値が有効になります。

反映システム定義の ref_wait_interval オペランドも省略すると、反映システム定義の ref_wait_interval オペランドのデフォルト値が仮定されます。

- **commit_wait_time = COMMIT 発行間隔**

～ 〈符号なし整数〉 ((0~300)) 《30》 (単位：秒)

このオペランドの意味については、反映システム定義の commit_wait_time オペランドを参照してください。

このオペランドを指定した場合は、反映システム定義の commit_wait_time オペランドの指定値よりも優先されます。

このオペランドを省略すると、反映システム定義の commit_wait_time オペランドの指定値が有効になります。

反映システム定義の commit_wait_time オペランドも省略すると、反映システム定義の commit_wait_time オペランドのデフォルト値が仮定されます。

- **mapping_key_check = { not null unique | unique | none }**

マッピングキー列のユニークチェックを行う場合のユニーク条件を指定します。ここで指定する値は、反映定義の check 句を省略した場合の規定値になります。反映定義に check 句を指定した場合、check 句の指定が優先されます。ユニークチェックの内容については、「5.10.5 反映表定義」の check 句を参照してください。

not_null_unique

反映対象表に対して、「表 5-20 ユニークチェックの内容」の内容を満たすインデックスが定義されていること、及びそのインデックスの構成列が NOT NULL 属性であることを確認します。

unique

反映対象表に対して、「表 5-20 ユニークチェックの内容」の内容を満たすインデックスが定義されていることを確認します。NULL 値に関するチェックは行わないため、完全なデータの一意性は、抽出側 DB を更新するユーザプログラムなどで保証する必要があります。

none

チェックを行いません。データの一意性は、抽出側 DB を更新するユーザプログラムなどで保証する必要があります。

注

反映対象表が反映側システムにあるかどうかをチェックしない場合、このオペランドの指定値に関係なくマッピングキーのチェックは行いません。反映対象表が反映側システムにあるかどうかのチェックに関しては、反映環境定義の `tblcheck` オペランドの内容に従います。

• `control_trigger = execute | not_execute`

反映対象表の、ユーザが設定したトリガを実行するかどうかを指定します。

次のような場合、トリガを実行すると抽出側 DB と反映側 DB に不整合が発生することがあります。

- BEFORE トリガでルーチン制御 SQL の代入文を使用した場合
- レプリケーション対象となる反映先の表に AFTER トリガを定義した場合

execute

反映対象となる表のトリガを実行します。

not_execute

反映対象となる表のトリガを実行しません。

トリガが定義されている表のデータ連動については、「3.3.6 トリガが定義されている表のデータ連動」を参照してください。

• `control_reference_trigger = execute | not_execute`

反映対象表の、HiRDB が設定したトリガを実行するかどうかを指定します。

execute

反映対象となる表のトリガを実行します。

not_execute

反映対象となる表のトリガを実行しません。

参照制約動作に CASCADE を指定した参照表がある場合は、`not_execute` を指定してください。

トリガが定義されている表のデータ連動については、「3.3.6 トリガが定義されている表のデータ連動」を参照してください。

• `check_pending = use | nouse`

反映対象表に参照制約を定義した表がある場合、参照制約を定義した表を検査保留状態とするかどうかを指定します。このオペランドは、`control_reference_trigger` オペランドに `not_execute` を指定した場合だけ有効です。

use

検査保留状態にします。

nouse

検査保留状態にしません。

- `reflect_delay_limit_time` = 反映滞留時間しきい値
～ 〈符号なし整数〉 ((0~86400)) 《0》 (単位：秒)

更新情報がシステムログファイルに格納された時刻と、反映側 DB でその更新情報が反映された時刻との差（反映滞留時間）のしきい値を指定します。反映滞留時間が、ここで指定したしきい値よりも大きくなった場合、警告メッセージが出力されます。

反映環境定義の次のオペランドでの指定値も、滞留時間として考慮する必要があります。

- `cmtintvl` オペランド
- `trncmtintvl` オペランド
- `tblcmtintvl` オペランド
- `ref_wait_interval` オペランド
- `commit_wait_time` オペランド

なお、0 を指定した場合、反映プロセスでの滞留監視機能は無効になります。

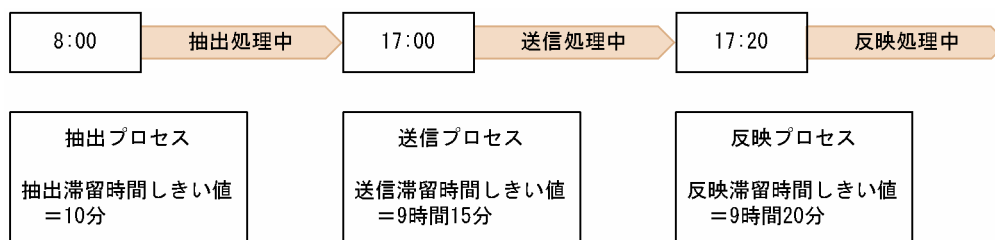
注意

このオペランドは、レプリケーション対象となる更新量や、送信間隔、反映間隔などを考慮した上で、値を調整してください。初めは、20 分（1200 秒）を目安に設定することをお勧めします。

運用例

送信処理を遅らせて開始する場合の運用例を次の図に示します。

図 5-9 送信処理を遅らせて開始する場合の運用例



例では、8 時~17 時まで抽出処理だけを実行し、17 時以降から送信処理を開始しています。このような運用環境下で反映滞留時間を監視する場合、更新データが反映システムに 9 時間送信されずに滞留したままとなるため、反映滞留時間しきい値に 9 時間分を足す必要があります。

- `sql_lockerr_retrynum` = 排他エラー発生時のトランザクションリトライ回数
～ 〈符号なし整数〉 ((0~255)) 《3》

次の場合にロールバックしたトランザクションを、リトライする回数を指定します。0 を指定した場合、リトライしません。

- 反映処理が排他エラーとなった場合

- デッドロックが発生した場合
- 通信障害が発生した場合

HiRDB のシステム共通定義の `pd_lck_wait_timeout` オペランドで指定する HiRDB の排他待ち時間があるため、Datareplicator 内でのリトライ間隔はありません。このオペランドの指定値は、1 回の反映側トランザクションについての再試行回数であるため、反映側トランザクションがコミットを発行した場合はリセットされます。

次に、このオペランドに 10 を指定した場合について説明します。

- あるトランザクションで排他エラーが 4 回発生すれば、4 回リトライします。
- 次のトランザクションは 6 回リトライをするのではなく、先のトランザクション分についてはリセットします。したがって、毎回、リトライが成功するまで、最大 10 回リトライします。
- 最初の排他エラーはリトライ回数に含まれません。したがって、10 を指定した場合、11 回分排他エラーにならないければ、リトライ回数はオーバーしません。

次に、リトライする場合を対象 DBMS ごとに示します。

反映対象 DBMS	リトライする場合	リトライ対象
HiRDB	エラー時の SQLCODE が次の場合 -770 (排他タイムアウトエラー) -911 (デッドロックエラー) -722 (通信エラー) -723 (通信エラー)	エラーとなった更新処理を含むトランザクション
Oracle (Windows 版) SQL Server	ODBC 関数の戻り値が 40001 (デッドロックエラー) の場合	エラーとなった更新処理を含むトランザクション
Oracle (Windows 版以外)	OCI 関数の戻り値が 60 (デッドロックエラー) の場合	エラーとなった更新処理

- `xa_recovery_retry_count` = トランザクション回復要求のリトライ回数

～ 〈符号なし整数〉 ((0~1024)) 《180》

HiRDB にトランザクションの回復を要求した場合に、HiRDB が開始していなかったり、HiRDB の受付プロセスの負荷が大きかったりすると、トランザクションを回復できないことがあります。このオペランドには、トランザクションを回復できなかったときに、再度トランザクションの回復を要求する回数を指定します。

このオペランドは、反映システム定義の `commitment_method` オペランドに `fxa_sqlc` を指定した場合に有効になります。

Datareplicator の反映グループの SQL プロセス数 (HiRDB に対して同時に回復を要求する SQL プロセス数の最大値) を目安に指定してください。なお、0 を指定した場合は、トランザクションの回復の再要求をしません。

- `xa_recovery_retry_interval` = トランザクション回復要求のリトライ間隔

～ 〈符号なし整数〉 ((1~180000)) 《1000》 (単位：ミリ秒)

HiRDB にトランザクションの回復を要求した場合に、HiRDB が開始していなかったり、HiRDB の受付プロセスの負荷が大きかったりすると、トランザクションを回復できないことがあります。このオペランドには、トランザクションを回復できなかったときに、再度トランザクションの回復を要求する間隔をミリ秒単位で指定します。

このオペランドは、反映システム定義の `commitment_method` オペランドに `fxa_sqlc` を指定した場合に有効になります。

このオペランドの指定値を小さくすると、反映処理の時間を短縮できますが、SQL プロセスの CPU 利用率が上がり、システム全体の負荷が高くなるおそれもあります。このため、CPU の性能を考慮した上で値を指定してください。

特に、Datareplicator の反映グループの SQL プロセス数 (HiRDB に対して同時に回復を要求する SQL プロセス数の最大値) が多い場合は、HiRDB のリカバリ処理の負荷が高くなるため、リトライ間隔を長めに指定してください。

• **reflect_trn_max_sqlnum = 反映トランザクション内最大更新 SQL 数**
 ~ 〈符号なし整数〉 ((1~10000000))

複数の抽出側トランザクションを一つの反映トランザクションとして反映する場合、反映トランザクションの最大更新 SQL 数を指定します。この値を基準として、一つの反映トランザクションとして扱われる抽出側トランザクション数が決まります。このオペランドを省略した場合、`cmtintvl` オペランド、`trncmtintvl` オペランド、又は `tblcmtintvl` オペランドの指定値に従います。

なお、`reflect_trn_max_sqlnum` オペランドの指定値を大きくすると、反映側 Datareplicator の 1 トランザクションで実行する SQL 件数が増加するため、次に示す HiRDB のリソースが多く消費されます。

- `pd_lck_pool_size`
- `pd_max_access_tables`
- `pd_log_sdinterval`
- `pd_sql_object_cache_size`

排他リソースが不足しエラーが発生する場合は、`reflect_trn_max_sqlnum` オペランドの指定値を小さくしてください。

一つの反映トランザクションとして扱われる抽出側トランザクション数は、次のように決まります。

反映トランザクションの対象となる更新 SQL 数	反映トランザクションの内容
<code>reflect_trn_max_sqlnum</code> オペランド指定値以下の場合	<code>cmtintvl</code> オペランド、 <code>trncmtintvl</code> オペランド、又は <code>tblcmtintvl</code> オペランドに従い、決定されます。
<code>reflect_trn_max_sqlnum</code> オペランド指定値を超えた場合	更新 SQL 数の合計値が指定値を超えるまでの抽出側トランザクション群を、一つの反映トランザクションとします。 一つの抽出側トランザクションだけで指定した更新 SQL 数を超える場合は、そのトランザクションを一つの反映トランザクションとします。

`cmtintvl = 2`, `reflect_trn_max_sqlnum = 20` という条件での例を、次の図に示します。

図 5-10 抽出側トランザクションと反映側トランザクションの関係

抽出側 トランザクション	T1 更新SQL=5	T2 更新SQL=10	T3 更新SQL=15	T4 更新SQL=100	T5 更新SQL=5
	↓		↓	↓	↓
反映側 トランザクション	T1+T2 更新SQL=15		T3 更新SQL=15	T4 更新SQL=100	T5 更新SQL=5

(凡例)

T:トランザクション

1. T1 の更新 SQL 数は `reflect_trn_max_sqlnum` オペランドの指定値以下であり、`cmtintvl` オペランドの指定値が 2 であることから、T2 を T1 と合わせて一つの反映側トランザクションとして扱えるかどうか判定します。

T1 と T2 の更新 SQL 数の合計値は `reflect_trn_max_sqlnum` オペランドの指定値以下です。したがって、T1 と T2 を合わせて、一つの反映側トランザクションとすることができます。

2. T3 の更新 SQL 数は `reflect_trn_max_sqlnum` オペランドの指定値以下であり、`cmtintvl` オペランドの指定値が 2 であることから、T4 を T3 と合わせて一つの反映側トランザクションとして扱えるかどうか判定します。

しかし、T3 と T4 の更新 SQL 数の合計値は、`reflect_trn_max_sqlnum` オペランドの指定値を超えてしまいます。したがって、T3 と T4 は、別の反映側トランザクションとなります。

3. 残った T5 は、そのまま一つの反映側トランザクションとなります。

注意

`reflect_trn_max_sqlnum` オペランドの指定値は、反映先 DBMS が一つのトランザクション処理に必要なリソースの上限値を目安に設定してください。

5.10 反映定義

反映定義では、反映側 Datareplicator の反映処理に必要な情報を定義します。次に示す機能を使わないで同一形式の表に反映する場合には、反映定義を省略できます。

- 反映処理をグループ単位に並列実行する表単位反映機能
- UOC によって加工して反映する UOC を使った反映機能
- 時系列情報表に更新情報を出力する時系列情報取得機能

反映定義を省略したときに仮定される値を次の表に示します。

表 5-19 反映定義を省略したときに仮定される値

抽出側 DB							反映側 DB
HiRDB	XDM/SD E2*	XDM/RD E2	ADM	PDMII E2	TMS-4V/SP	RDB1 E2	HiRDB
認可識別子	スキーマ名	認可識別子	データベース名	DBM 名	認可識別子	認可識別子	認可識別子
表識別子	最下位のレコード型名	表識別子	セグメント	データセット名	表識別子	表識別子	表識別子
列名	構成要素名	列名	フィールド名	フィールド名	列名	列名	列名

注※

1 個の抽出定義内で、構成要素名、及び抽出再定義 RESTRUCT 文で指定した再定義項目名がユニークでない場合は、再定義項目名に対応する反映表の定義ができないため、反映定義は省略できません。

5.10.1 定義の規則

反映定義中に指定するフィールド名称、認可識別子、表識別子、及び順序数生成子識別子は、次に示す規則に従って指定してください。

- 1～30 バイト（認可識別子は 1～8 バイト）の間で指定してください。
- 使用できる文字は次のとおりです（認可識別子は、英大文字、英小文字、及び数字だけ使用できます）。
 - 英大文字 (A～Z, #, @, ¥)
 - 英小文字 (a～z)
 - 数字 (0～9)
 - 仮名文字
 - _(下線)
 - 空白

- - (ハイフン)
- 全角文字
- 半角文字と全角文字は、混在して指定できます。
- 先頭文字には、半角英大文字、半角英小文字、半角仮名文字 (ア～ン, ヲ), 又は全角文字を指定してください。
- 全角空白文字は指定できません。
- 半角空白, 又は半角ハイフンを含む場合は, 「"」(ダブルクォーテーション) と 「"」(ダブルクォーテーション) で囲んで指定してください。

5.10.2 構造と形式

(1) 構造

反映定義の構造を次の図に、反映定義文の内容と繰り返し指定できる指定数を次の表に示します。

図 5-11 反映定義の構造

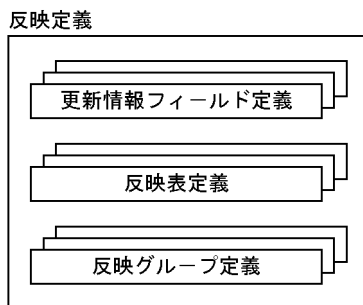


表 5-20 反映定義の内容と指定数

定義名	定義文	指定数	内容
更新情報フィールド定義	format 文	0~4095	抽出したデータと更新情報のフィールドを対応付けます。
反映表定義	load 文	0~4095	更新情報を反映する方法や、時系列情報として取得する内容を指定します。
反映グループ定義	group 文	0~128	表単位反映方式で反映する場合の反映処理のグループ分けを指定します。

(2) 形式

/* 更新情報フィールド定義 */

{ format 更新情報名

name フィールド名称

```

[ { const 初期値 | by 列データ編集 UOC 関数識別子 [ nocodecnv ] | [ nocodecnv ] }
  [ { name フィールド名称
    [ { const 初期値 | by 列データ編集 UOC 関数識別子 [ nocodecnv ] | [ nocodecnv ] }
      ... } ]
  } ... ]
/* 反映表定義 */
[ { load { フィールド名称 [ {, フィールド名称 } ... ] | * }
  from 更新情報名
  { to [ { timestamp | sqlconvopt1 | sqlconvopt2 }
    [ 認可識別子. ] 表識別子
    [ check { not_null_unique | unique | none } ]
    [ with lock ]
    | by 'uoc 名' }
  } ... ]
/* 反映グループ定義 */
[ { group 反映グループ名称
  by [ 認可識別子. ] 表識別子 [ {, [ 認可識別子. ] 表識別子 } ... ]
/*** キーレンジ単位分割方式の定義 ****/
[ { [ in '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [ / '反映先 FES のポート番号' ]
  [ having キーレンジ分割条件文 [ {, キーレンジ分割条件文 } ... ] | other ]
  } ... ]
/*** ハッシュ分割方式の定義 ****/
[ { hash
  [ in '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [ / '反映先 FES のポート番号' ]
  [ (RD エリア名 [ {, RD エリア名 } ... | other) ]
  [ {, '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [ / '反映先 FES のポート番号' ]

```

```
[(RD エリア名 {{, RD エリア名}} … | other)]
```

```
}}… ]
```

```
| divide into SQL プロセス分割数 }
```

```
}}…]]
```

```
;
```

5.10.3 定義内容の変更

定義内容を変更する場合には、次のように操作してください。

1. 抽出側の業務を停止します。
2. 反映側にすべての更新データが反映されていることを確認します。
3. 抽出側及び反映側 Datareplicator を停止します。
4. テキストエディタで抽出定義及び反映定義の内容を変更します。
5. 抽出側 Datareplicator の環境を初期化します。
6. hdeprep コマンドで抽出定義のプリプロセスを実行します。
7. 反映側 Datareplicator を初期開始します。
8. 抽出側 Datareplicator を起動します。
9. 抽出側 HiRDB の業務を再開します。

5.10.4 更新情報フィールド定義

抽出側システムから送信される更新情報の名称と、各フィールドの名称を定義します。

更新情報のすべてのフィールドを、並び順を変えないで反映表定義の load 文で指定した列名の順に対応付ける場合、更新情報フィールド定義は省略できます。この場合、反映表定義の load 文では、フィールド名称を省略して、「*」を指定します。「*」を指定すると、更新情報フィールド定義があっても、その内容は無視されます。

更新情報フィールド定義は、一つの更新情報に対して一つだけ定義できます。

(1) 形式

```
{{ format 更新情報名
```

```
name フィールド名称
```

```
[{ const 初期値 | by 列データ編集 UOC 関数識別子 [ nocodecnv ] | [ nocodecnv ]}
  {{{ name フィールド名称
    [{ const 初期値 | by 列データ編集 UOC 関数識別子 [ nocodecnv ] | [ nocodecnv ]}
      ...}}]
  ...}]
}}...
```

(2) オペランドの説明

- format 更新情報名

～ 〈1～8 文字の記号名称〉

抽出側システムの抽出定義で定義されている更新情報名を指定します。

- name フィールド名称 [{ const 初期値 | by 列データ編集 UOC 関数識別子 [nocodecnv] | [nocodecnv]}

```
[[{ name フィールド名称 [{ const 初期値 | by 列データ編集 UOC 関数識別子 [ nocodecnv ] | [ nocodecnv ]}
  ...}}]
```

format 文で指定した更新情報のフィールド名称, 又は追加するフィールド名称を指定します。フィールド名称は 4000 まで指定できます。

- name フィールド名称

反映するかどうかに関係なく, 更新情報のすべてのフィールド名称を, 更新情報の先頭のフィールド名称から順に指定します。更新情報にはない任意のフィールドを追加する場合には, 任意の名称を指定し, 次に説明する「const 初期値」を指定します。

フィールド名称は, 1 個の更新情報フィールド定義内で一意になるように指定してください。

フィールド名称の指定方法は, 「5.10.1 定義の規則」を参照してください。

- const 初期値

更新情報にはない任意のフィールドを追加する場合に, 追加するフィールドの初期値を指定します。なお, const に指定した値を反映側 DB に反映できるのは, 更新種別が挿入 (INSERT) のときだけです。

初期値として指定できる定数を次の表に示します。

表 5-21 const 句に指定できる定数

定数	列属性	説明	指定内容	反映内容
文字列定数 ^{*1}	CHARACTER	アポストロフィ (') で囲んだ文字列	(例) 'abc'	(例) abc
日本語文字列定数 ^{*1}	NCHAR	アポストロフィ (') で囲んだ日本語文字列	(例) '平和'	(例) 平和
真数定数 ^{*2} 又は 符号なし整数 ^{*2}	INTEGER, SMALLINT,	符号 (+, -), 数字, 小数点から構成される文字列	(例) -1.23	(例) -1.23

定数	列属性	説明	指定内容	反映内容
	DECIMAL, FLOAT, SMALLFLT			
null 定数	※3	null 値を示す定数	null	null 値
反映種別	CHAR	反映処理の更新種別を示す定数	reflect_kind	※4
反映日付定数	DATE	反映日付を示す定数	reflect_date	(例) 19990328※5
反映時間定数	TIME	反映時間を示す定数	reflect_time	(例) 221530※6
反映時刻印定数	TIMESTAMP	反映時刻印を示す定数	reflect_timestamp	(例) 19990328221530※7
反映順序数 生成子定数	INTEGER, SMALLINT, 位 取り0の DECIMAL	SQL 通番を取得する順序数生成子名	(例) sequence[認可 識別子.] 順序数生成子識別 子, または認可識 別子を省略する と, 反映システム 定義で定義した認 可識別子が假定さ れます。	(例) 1※8
抽出日付定数	DATE	抽出日付を示す定数	extract_date	(例) 19990328※5
抽出時間定数	TIME	抽出時間を示す定数	extract_time	(例) 214016※6
抽出時刻印定数	TIMESTAMP	抽出時刻印を示す定数	extract_timestamp	(例) 19990328214016※7

注

抽象データ型の列と繰返し列には、null 定数だけを指定できます。null 定数以外を指定すると、反映時に HiRDB でデータ互換性エラーになります。

注※1

30 バイトまで指定できます。

注※2

10 進数で 29 けたまで指定できます。

注※3

すべての列属性に対応します。

注※4

反映処理の更新種別によって、次の 4 種類のどれかが反映されます。

- ・ upd：更新種別が更新 (UPDATE) の場合

- ・ ins : 更新種別が挿入 (INSERT) の場合
- ・ del : 更新種別が削除 (DELETE) の場合
- ・ purge : 更新種別が全行削除 (PURGE TABLE) の場合

注※5

反映日付と抽出日付は、年月日の形式で反映されます。

注※6

反映時間と抽出時間は、時分秒の形式で反映されます。

注※7

反映時刻印と抽出時刻印は、年月日時分秒 (秒は小数点第 0, 2, 4, 6 位までを選択) の形式で反映されます。

ただし、抽出時刻印については、小数点以下の時刻は抽出側 DB がメインフレームの場合にだけ取得します。また、そのときに取得する秒数は、小数点第 2 位までになります。

注※8

順序数生成子が生成した値です。

• by 列データ編集 UOC 関数識別子

更新情報のフィールドに対応する抽出データを、列データ編集 UOC でデータ編集する場合に、編集に使う列データ編集 UOC 関数識別子を指定します。

列データ編集 UOC 関数識別子は、typeX (X は 1 から 8 まで指定でき、それぞれ UOC 関数 hds_ucoledit1() から hds_ucoledit8() に対応) を指定します。

ただし、マッピングキーに対応するフィールドに対しては指定できません。列データ編集 UOC で扱えないデータ型及び列属性については、「8.2.5(4) データ型の制限」を参照してください。

• nocodecnv

更新情報のフィールドに対応する抽出データの文字コード変換処理を抑止させる場合に指定します。ただし、次に示す抽出列属性に対応するフィールドには指定できません。

- 文字型以外に対応するフィールド
- 抽象データ型 (SGMLTEXT, FREEWORD, XML) に対応するフィールド
- マッピングキーに対応するフィールド

反映情報編集 UOC に対しての反映表定義 (load 文中) に、このオプションを指定したフィールドが含まれていない場合は、このオプションは無視して (文字コード変換処理を実行して) 反映処理を実行します。

5.10.5 反映表定義

更新情報に対する反映対象表を定義します。1 個の更新情報を、複数の反映対象表に反映する場合には、反映表定義を反映対象表の数だけ繰り返し定義します。フィールド名称、認可識別子、及び表識別子の指定方法については、「5.10.1 定義の規則」を参照してください。

(1) 形式

```
{ { load { フィールド名称 [ {, フィールド名称 } … ] | * }  
  from 更新情報名  
  { to [ { timestamp | sqlconvopt1 | sqlconvopt2 } ]  
    [ 認可識別子. ] 表識別子  
    [ check { not_null_unique | unique | none } ]  
    [ with lock ]  
  | by ' uoc 名' }  
  } … }
```

(2) オペランドの説明

- load { フィールド名称 [{, フィールド名称 } …] | * }

更新情報のフィールドを、反映対象表にどのように対応付けるのかを定義します。マッピングキーに該当するフィールド名称は、反映対象表のマッピングキーを格納した列に対応するように定義してください。

なお、HiRDB の pd_max_access_tables オペランドには、指定した load 文の個数以上の値を指定してください。pd_max_access_tables オペランドの指定値が、指定した load 文の個数よりも少ない場合は、HiRDB で KFPA11931-E メッセージが出力されエラーになります。pd_max_access_tables オペランドの詳細については、マニュアル「HiRDB システム定義」を参照してください。

フィールド名称

更新情報フィールド定義で指定したフィールド名称のうち、反映対象とするフィールドを、反映対象表の左の列から順に対応付けます。フィールド名称は、最大 4000 個指定できます。

ここで指定するフィールド名称の数は、反映対象表の列数と一致するように指定してください。

*

「*」を指定すると、対応する更新情報に関する更新情報フィールド定義は無視されます (const, 列データ編集 UOC 関数識別子, 及び nocodecnv 指定も無視されます)。

更新情報のすべてのフィールドは、並び順を変えないでそのまま反映対象表に対応付けられます。

- from 更新情報名

～ 〈1~8 文字の記号名称〉

反映対象にする更新情報名を指定します。更新情報名には抽出定義で指定した名称を指定します。

- to [{ timestamp | sqlconvopt1 | sqlconvopt2 }] [認可識別子.] 表識別子
 [check { not_null_unique | unique | none }] [with lock] | by ' uoc 名'

反映対象表の認可識別子と表識別子、又は UOC 名を定義します。認可識別子と表識別子については、マニュアル「HiRDB SQL リファレンス」を参照してください。

なお、反映対象表の表識別子を指定する場合は、更に次に示すオプションの指定ができます。

- 時系列情報を取得するかどうか。
- マージ表を作成するかどうか。
- ユニークチェックを行うかどうか。
- 反映対象表に対して LOCK TABLE 文を発行するかどうか。

timestamp

時系列情報を取得するときに指定します。timestamp を指定すると、load 句の指定は時系列情報格納表と認識して格納されます。

繰返し列が含まれている表には、timestamp は指定できません。指定すると定義解析エラーになって、このデータ連動識別子での反映処理が停止します。

sqlconvopt1 | sqlconvopt2

更新だけでなく、追加や削除を実行できるマージ表を作成するときに指定します。この値を指定するときは、反映対象表が次に示す条件を満たしていることが前提です。

1. マッピングキー列の一部又は全部に、ユニークインデックスが定義されている。
2. マッピングキーでない列には、ユニークインデックスの定義がない。
3. 反映対象表は、時系列情報表以外である。
4. マッピングキーが更新されない。

SQL 文の実行と指定値による動作の違いを次に示します。

SQL 文	sqlconvopt1	sqlconvopt2
INSERT でキー重複が発生。	UPDATE に置き換えて実行されます。	UPDATE に置き換えて実行されます。
UPDATE で該当行がない。	INSERT に置き換えて実行されます。	INSERT に置き換えて実行されます。
DELETE	該当行が削除されます。該当行がないときは、無視されます。	該当行があるときは、マッピングキー以外で CONST 句指定のない反映対象列に NULL 値を設定してから、UPDATE に置き換えて実行されます。 該当行がないときは、無視されます。
PURGE	すべての行が削除されます。	すべての行が削除されます。

マージ表を作成するときの注意

1. マージ表を作成するときは、sqlconvopt1 又は sqlconvopt2 を指定します。sqlconvopt1 と sqlconvopt2 では、DELETE データに対する処理が異なります。sqlconvopt1 では DELETE を行削除として反映しますが、sqlconvopt2 では DELETE を null 値更新として反映します。null 値更新の対象になる列は、format 文で const 句を指定していない列だけです。
2. マージ表を作成する指定をすると、SQL を 2 回発行することになります。そのため、変換が大量に発生すると反映処理の性能に影響が出ることがあります。

3. 列データ編集 UOC を使うときは、変換前 SQL 発行直前と変換後発行直前の 2 回 UOC が呼び出されます。また、sqlconvopt2 指定で DELETE を実行したときも UOC が呼び出されます。このときのデータの値は、NULL 値になります。

認可識別子

反映対象表の認可識別子を指定します。timestamp を指定したときは、時系列情報格納表の認可識別子を指定します。認可識別子を省略すると、反映システム定義で定義した認可識別子が仮定されます。

表識別子

反映対象表の表識別子を指定します。timestamp を指定したときは、時系列情報格納表の表識別子を指定します。

check {not_null_unique | unique | none}

マッピングキー列に対するユニークチェックを行うときの条件を指定します。ユニークチェックの内容を次の表に示します。この句を指定すると、反映環境定義の mapping_key_check オペランドの値に関係なく、ここで指定した値に従ってユニークチェックを行います。

この句を省略した場合は、mapping_key_check オペランドの指定値に従います。

なお、timestamp オプションが指定されている表には、指定値に関係なくユニークチェックは行われません。

また、マージ表作成オプションを使用する場合には、not_null_unique を指定することをお勧めします。

表 5-22 ユニークチェックの内容

チェック項目	チェック内容
インデクス種別	次のどれかのインデクス種別であること。 <ul style="list-style-type: none"> ユニークインデクス ユニーククラスタインデクス プライマリインデクス プライマリクラスタインデクス
インデクス構成列	マッピングキー構成列だけであること。

not_null_unique

反映対象表に対して、「表 5-20 ユニークチェックの内容」の内容を満たすインデクスが定義されていること、及びそのインデクスの構成列が NOT NULL 属性であることを確認します。

unique

反映対象表に対して、「表 5-20 ユニークチェックの内容」の内容を満たすインデクスが定義されていることを確認します。NULL 値に関するチェックは行わないため、完全なデータの一意性は抽出側 DB を更新するユーザプログラムなどで保証する必要があります。

none

チェックを行いません。データの一意性は抽出側 DB を更新するユーザプログラムなどで保証する必要があります。

mapping_key_check オペランドと check 句では、指定した際の反映側 Datareplicator の動作が異なります。

mapping_key_check オペランド (not_null_unique 指定) の例と、反映側 Datareplicator の動作を次の表に示します。

指定例	動作
load * from t1 to T1 check not_null_unique	not_null_unique のチェックを行います。
load * from t1 to T1 check none load * from t1 to T1	チェックを行いません。

mapping_key_check オペランドを省略した場合 (check 句の指定が優先されます) の例と、反映側 Datareplicator の動作を次の表に示します。

指定例	動作
load * from t1 to T1 check not_null_unique	not_null_unique のチェックを行います。
load * from t1 to T1 check none	チェックを行いません。
load * from t1 to T1	not_null_unique のチェックを行います。

with lock

load 文で指定した表に対して LOCK TABLE 文を発行後、反映処理を実行する場合に指定します。この句の指定は、HiRDB への反映にだけ有効となります。HiRDB 以外への反映の場合、指定は無視されます。

with lock 句は次に示す場合に指定します。

- ・ HiRDB の共用 RD エリア中に作成した共用表に対して反映処理を実行する場合 (必須※)

注※ 指定がない場合、SQL 実行エラーとなります。

- ・ 反映処理で排他制御によるオーバヘッドを削減したい場合 (任意)

LOCK TABLE 文の実行がエラーとなった場合の動作を次の表に示します。

表 5-23 LOCK TABLE 文の実行がエラーとなった場合の動作

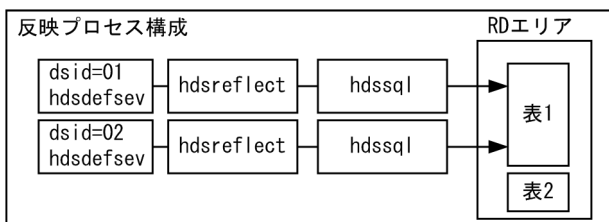
SQLCODE	反映環境定義の skip_sqlcode オペランド指定なし	反映環境定義の skip_sqlcode オペランド指定あり
-770, -911, -722, -723	実行中のトランザクションをロールバックし、同じトランザクションを更に 2 回までリトライします。2 回目のリトライでもエラーが解消されない場合、KFRB03034-E メッセージ及びエラーとなった LOCK TABLE 文を未反映情報ファイルへ出力し、反映処理を停止します。	LOCK TABLE 文をスキップし、LOCK TABLE 文が発行されていない状態で反映処理を続行します。出力する情報種別は反映環境定義の sqlerr_skip_info オペランドに依存します (output, 又は msgoutput が指定されている場合、KFRB03043-W メッセージを出力します)。
上記以外	KFRB03034-E メッセージ及びエラーとなった LOCK TABLE 文を未反映情報ファイルへ出力し、反映処理を停止します。	

注意

・ with lock 句を指定する場合、反映対象表は一つの RD エリアに対して、一つの表を格納する配置を推奨します。

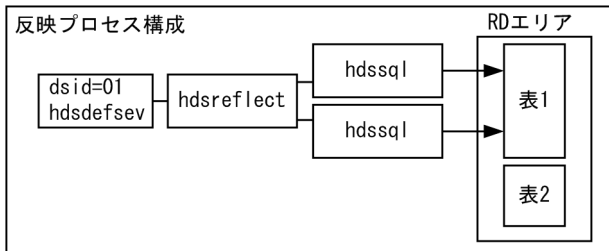
・ 次に示す反映プロセス構成ではデッドロックが発生する可能性があるため適用できません。

(1)



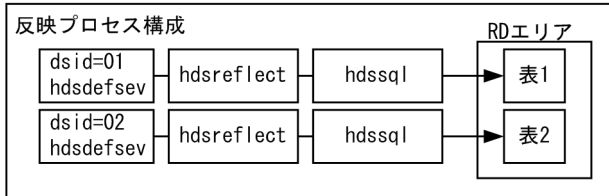
- ・ RDエリア種別
共用・非共用
- ・ 定義内容
異なるデータ連動識別子から同一表へ反映する。

(2)



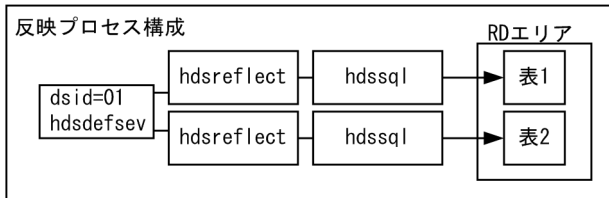
- ・ RDエリア種別
共用・非共用。
- ・ 定義内容
表単位反映方式でハッシュ分割、キーレンジ分割を使用して反映する。

(3)



- ・ RDエリア種別
共用。
- ・ 定義内容
異なるデータ連動識別子から同一共用RDエリア上の共用表へ反映する。

(4)



- ・ RDエリア種別
共用。
- ・ 定義内容
表単位反映方式で異なる反映グループから同一共用RDエリア上の共用表へ反映する。

・ 共用表を反映対象表とする場合、共用表への反映トランザクションが発生しているときに、非反映対象表への参照・更新ができなくなるため、反映対象表と非反映対象表を異なる共用 RD エリアに格納しておく必要があります。

by 'uoc 名'

～ 〈1～64 バイトのパス名〉

ユーザが独自に作成したプログラム (UOC) で反映処理を実行する場合に、そのプログラム名を絶対パス名で指定します。パス名を省略してファイル名だけを指定できません。また、by 句を指定した場合は load 句に定数フィールドを指定できません。

5.10.6 反映グループ定義

表単位反映方式 (表単位分割方式、キーレンジ単位分割方式、又はハッシュ分割方式) で反映処理を並列に実行するときの実行単位にする反映グループを定義します。反映グループは、128 個まで繰り返し定義できます。フィールド名称、認可識別子、及び表識別子の指定方法については、「5.10.1 定義の規則」を参照してください。

(1) 形式

{ { group 反映グループ名

by [認可識別子.] 表識別子 [{, [認可識別子.] 表識別子 } …]

/**** キーレンジ単位分割方式の定義 ****/

{ { { in '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [/ '反映先 FES のポート番号']

[having キーレンジ分割条件文 [{, キーレンジ分割条件文 } …] | other]

} …]

/**** ハッシュ分割方式の定義 ****/

{ { hash

{ in '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [/ '反映先 FES のポート番号']

[(RD エリア名 {, RD エリア名} … | other)]

{ {, '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [/ '反映先 FES のポート番号']

[(RD エリア名 {, RD エリア名} … | other)]

} …]

| divide into SQL プロセス分割数 }

}}…}}

(2) オペランドの説明

- group 反映グループ名

～ 〈1～8 文字の記号名称〉

反映を実行するグループの名称を指定します。反映グループ名は、反映定義中のすべての反映グループ定義で一意になるように指定してください。

次に示す文字列は、大文字、小文字に関係なく、反映グループ名として指定できません。

- trngroup
- othergrp
- uocxxx*

注※ xxx は 3 けたの符号なし整数を表します。

例えば、trngroup の場合は trngroup, TRNGROUP, TmGroup などが指定できません。

- by (認可識別子.) 表識別子 [{, [認可識別子.] 表識別子 }…]

グループを構成する反映対象表を指定します。

認可識別子

グループを構成する反映対象表の認可識別子を指定します。認可識別子を省略すると、反映システム定義で設定した認可識別子が仮定されます。

表識別子

グループを構成する反映対象表の表識別子を指定します。ただし、同一の表識別子を複数のグループに指定できません。キーレンジ分割 又はハッシュ分割をする場合には、その反映グループ定義で指定できる表識別子の数は一つだけです。

- in '反映先 FES のホスト名' / '反映先 FES のサーバ識別子' [/ '反映先 FES のポート番号']

反映側 HiRDB でマルチ FES 機能を使ってフロントエンドサーバが複数起動している場合に、反映先のフロントエンドサーバのホスト名、サーバ識別子、ポート番号を指定します。マルチ FES 機能を使っている場合に in 句の指定を省略すると、反映グループに対応する反映先のフロントエンドサーバは、反映側 HiRDB のシステムマネージャによって決められます。マルチ FES 機能を使っていない場合に in 句を指定しても、ただ一つ起動しているフロントエンドサーバに対して、すべての反映グループが対応付けられます。

一つの表に対する反映処理を、キーレンジ単位に分割して実行する場合には、in 句と having 句を、分割するキーレンジ数分指定します。このとき、having 句で分割する条件も指定します。

in 句は、一つの反映グループ定義内で最大 8 個定義できます。一つの反映グループ定義内で複数の in 句を指定する場合、having 句は省略できません。また、一つの反映グループ定義内で同一の反映先のフロントエンドサーバを複数指定することもできます。

反映先 HiRDB がマルチフロントエンドサーバの場合に、接続する HiRDB サーバのフロントエンドサーバのホスト名を指定します。HiRDB システム定義の pdunit で-p ポート番号を指定しているホス

トへ接続する場合（系切り替え機能を使用している場合）は、そのポート番号を指定する必要があります。指定フォーマットは次のとおりです。

反映先 FES のホスト名：反映先 FES のポート番号

反映先 FES のホスト名

～ 〈1～32 文字の識別子〉

反映先のフロントエンドサーバの PDFESHOST 環境変数に設定したホスト名を指定します。

反映先 FES のサーバ識別子

～ 〈1～8 文字の識別子〉

反映先のフロントエンドサーバの PDSERVICEGRP 環境変数に設定したサーバ識別子を指定します。

反映先 FES のポート番号

～ 〈1～5 文字の識別子〉

反映先のフロントエンドサーバの PDNAMEPORT 環境変数に設定したポート番号を指定します。

上記の環境変数の内容については、マニュアル「HiRDB UAP 開発ガイド」を参照してください。

in 句で指定できる反映先のフロントエンドサーバの数を次に示します。

表単位分割方式：1 グループ当たり 1 個

キーレンジ単位分割方式：一つのレンジで 1 個

ハッシュ分割方式：1 グループ当たり 2～8 個

- **having キーレンジ分割条件文** [{[, キーレンジ分割条件文]}...] | other

一つの表に対する反映処理をキーレンジ単位に分割して実行する場合に、分割するための条件文を指定します。having 句は、一つの in 句に対して一つだけ定義できます。in 句と having 句の組み合わせを複数指定した場合には、先頭の in 句と having 句から順に検証していき、最初に合致した条件に対する反映処理が実行されます。

なお、固定長文字列（CHAR, MCHAR, 又は NCHAR）を指定する場合、条件に合致させるために、条件値長＝定義長となるように、後ろにスペースを付加して指定してください。

反映データがすべての条件文に合致しない場合には、反映処理は実行されません。すべての条件文に合致しない場合にも反映処理を実行する場合には、各反映グループ定義の最後に「having other」を指定してください。

having キーレンジ分割条件文

キーレンジ分割をするときの条件文を指定します。条件文は次の形式で指定します。

列名 = 定数

列の値が指定した定数と等しい

列名 < > 定数

列の値が指定した定数と等しくない

列名 > 定数

列の値が指定した定数より大きい

列名 >= 定数

列の値が指定した定数以上

列名 < 定数

列の値が指定した定数より小さい

列名 <= 定数

列の値が指定した定数以下である

列名 IS NULL

列の値が NULL 値である

列名 IS NOT NULL

列の値が NULL 値でない

条件文は、一つの having 句に最大 8 個定義できます。一つの having 句に条件文を複数指定した場合は、指定した条件がすべて AND 結合されます。例えば、「having c1=100, c2=200」と指定した場合には、「c1=100」と「c2=200」の両方の条件を満たすデータの場合だけ、反映処理が実行されます。

having other

キーレンジ分割条件文に合致しない場合でも、データを反映する場合に指定します。other は、各反映グループ定義の最後に一つだけ指定できます。other を複数指定した場合には、反映定義の解析時にエラーになります。また、other の後に having 句を指定するとエラーになります。

• hash

一つの表に対する反映処理で反映 SQL プロセスを複数個 並列して処理するときに、更新情報を各プロセスに分散するときに指定します。分散するときは、HiRDB のハッシュ分散関数を使います。ハッシュキー列は HiRDB から検索しますが、マッピングキー以外の列が検索されると、反映定義の解析でエラーになります。

ハッシュ分割は、HiRDB でハッシュ分割した表に対してだけ指定できます。HiRDB でハッシュ分割していない表に hash オプションを指定すると、反映定義の解析でエラーになります。

抽出側の列と反映先のハッシュキー列との対応については、次に示す条件を満たしてください。

- 列のデータ型が抽出側と反映側で一致している。
- 列のデータ型が DECIMAL 型のときは、精度と位取りが一致している。
- 列のデータ型が文字型のときは、「抽出側の列長 ≤ 反映側の列長」である。
- 列のデータ型が TIMESTAMP 型のときは、小数部のけた数が一致している。

上記の条件が一つでも満たされないと、反映定義の解析でエラーになります。

hash オプションを指定してハッシュ分割するときは HiRDB のクライアントライブラリを使うので、環境変数 SHLIB_PATH に \$PDDIR/client/lib を追加してください。

• RD エリア名

ハッシュ分割は RD エリア単位でデータ格納先を振り分けるため、ハッシュ分割するときの反映先 FES 指定ごとの振り分け条件として、RD エリア名を指定します。指定がないときは、Datareplicator で自動的に振り分け先を決定します。

RD エリア名は省略できますが、一つの group 文の中で RD エリア名の指定がある反映先 FES 指定と指定がない反映先 FES 指定が混在させないでください。混在していると、定義解析エラーになって、反映処理が終了します。

一つの group 文の中で同じ RD エリア名を複数指定できません。複数指定していると、定義解析エラーになって、反映処理が終了します。

条件として指定していないすべての RD エリアのデータをすべて反映させるときは、RD エリア名に other を指定します。other を省略すると、group 文で最後に指定した反映先 FES 指定に other を指定したと仮定されます。other を指定するときは、次の規則に従ってください。

1. other は、複数指定する in 句の最後に指定してください。

[正しい例]

```
in 'host1'/'fes1' (rdarea1, rdarea2),  
   'host2'/'fes2' (rdarea3, rdarea4),  
   'host3'/'fes3' (other)
```

[エラーになる例]

```
in 'host1'/'fes1' (rdarea1, rdarea2),  
   'host2'/'fes2' (other),  
   'host3'/'fes3' (rdarea5, rdarea6)
```

エラーになる例の場合、3 行目と 2 行目を入れ換えて指定するとエラーを回避できます。

2. other は、RD エリア名の指定で、ほかの RD エリアと同時に指定できません。

[エラーになる例 その 1]

```
in 'host1'/'fes1' (rdarea1, rdarea2),  
   'host2'/'fes2' (rdarea3, rdarea4),  
   'host3'/'fes3' (rdarea5, other)
```

エラーになる例 その 1 の場合、other を削除するとエラーを回避できます。

[エラーになる例 その 2]

```
in 'host1'/'fes1' (rdarea1, rdarea2),  
   'host2'/'fes2' (rdarea3, other),  
   'host3'/'fes3' (rdarea5, rdarea6)
```

エラーになる例 その 2 の場合、2 行目と 3 行目を入れ換えて other を削除するとエラーを回避できます。

なお、hash オプションでは、キーレンジとは異なり、指定していないレンジを反映対象としないようにはできません。省略した RD エリアは、最後の FES 指定のレンジで反映されます。そのため、RD エリアを省略しないで、すべての RD エリア名をどれかの FES 指定で定義することをお勧めします。

- divide into SQL プロセス分割数
 ~ 〈符号なし整数〉 ((2~8))

HiRDB がシングルサーバの表に対する反映処理を、ハッシュ分割などで複数の SQL プロセスで並列に実行するときの SQL プロセス数を指定します。2~8 以外の値を指定すると、反映定義の解析でエラーになります。

(3) 反映グループ定義による反映グループ分け

反映グループ定義を定義している場合、反映側 Datareplicator では、反映処理をグループ単位に分けて並列処理します。反映グループは反映グループ定義に従って、128 グループまで生成できます。反映グループ定義による反映グループの生成を次の表に示します。なお、キーレンジ単位分割方式又はハッシュレンジ分割方式の反映グループを定義する場合、PURGE TABLE が実行されない表でなければなりません。

表 5-24 反映グループ定義による反映処理のグループの生成

反映処理の分類	反映処理実行時の処理グループの生成
反映グループ定義で指定した表に対する反映	反映グループ定義に従って、グループを生成します。
反映グループ定義で指定しなかった表に対する反映	指定されなかったすべての反映対象表で一つのグループを生成します。グループ名は othergrp と設定されます。
UOC による反映	UOC ごとにグループを生成します。グループ名は、UOC001 から UOC128 までの範囲で自動的に設定されます。

(4) キーレンジ分割を指定するときの規則

キーレンジ分割条件を指定するときの規則を説明します。

- **キーレンジ分割 (in 句と having 句の組み合わせ) の最大数**
一つの反映グループ定義では、キーレンジ分割 (in 句と having 句の組み合わせ) を 8 個まで指定できます。8 個以上のキーレンジ分割を指定すると、定義解析時にエラーになります。
- **キーレンジ分割の条件文の最大数**
一つのキーレンジ分割では、条件文を半角のコンマ (,) で区切ることによって、8 個まで指定できます。8 個以上の条件文を指定すると、定義解析時にエラーになります。
- **条件文に指定できる列**
キーレンジ分割の条件文に指定できる列は、反映処理でマッピングキーとしている列だけです。マッピングキーではない列をキーレンジ分割の条件文に指定している場合には、定義解析時にエラーになります。
- **条件文に指定した列に指定できる定数**
キーレンジ分割の条件文には、すべてのデータ型の列を指定できます。ただし、データ型ごとに、指定できる定数に制限があります。キーレンジ分割条件文内の列に指定できる定数を次の表に示します。

表 5-25 キーレンジ分割条件文内の列に指定できる定数

列のデータ型	指定できる定数	指定例
INTEGER	-2147483648~2147483647 の整数値	123456
SMALLINT	-32768~32767 の整数値	12345
DECIMAL(n)	n けたの整数値 (n : 1~38)	1234567890
DECIMAL(n, m)	整数部が 0~ (n-m) けたの固定小数点表現形式	123.456789
SMALLFLT	固定小数点表現形式, 及び単精度浮動小数点表現形式	1.23E10
FLOAT	固定小数点表現形式, 及び倍精度浮動小数点表現形式	1.2345E20
CHAR	1~255 文字の, シングルクォーテーション (') で囲まれた文字列	'ABCDEF'
VARCHAR		
NCHAR		
NVARCHAR		
MCHAR		
MVARCHAR		
DATE		
TIME	反映側 HiRDB での文字列表現のフォーマットである 'HH-MI-SS' だけ HH : 00~23 (時) MI : 00~59 (分) SS : 00~59 (秒)	'12:00:00'
INTERVAL YEAR TO DAY	精度 8, 位取り 0 の固定小数点数である ±yyyymmdd. yyyy : 0000~9999 (か年) mm : 00~99 (か月) dd : 00~99 (か日)	0000010101.
INTERVAL HOUR TO SECOND	精度 6, 位取り 0 の固定小数点数である ±hhmmss. hh : 00~99 (時間) mm : 00~99 (分) ss : 00~99 (秒)	-010101.
TIMESTAMP	反映側 HiRDB での文字列表現のフォーマットである 'YY-MM-DD HH-MI-SS [.NN...N]' だけ※ YYYY : 0001~9999 (年) MM : 01~12 (月) DD : 01~31 (日) HH : 00~23 (時)	'2001-01-01 10:45:30.152 3'

列のデータ型	指定できる定数	指定例
	MI : 00~59 (分) SS : 00~59 (秒) NN...N : 0~999999 (最大 6 けたの小数秒)	

注※

日データではその月の最終日付を超えるかだけをチェックします。うるう年のチェックはしません。

- **固定長文字列データ型の後ろの空白文字の扱い**

固定長文字列データ型を条件文に指定する場合には、反映環境定義の `ref_data_backspace` オペランドの指定値によって、次のように指定してください。

- **suppress を指定した場合**
固定長文字列データ型の後ろの空白文字を除いた値を指定します。
- **nosuppress を指定した場合**
固定長文字列データ型の後ろの空白文字を含んだ値を指定します。

(5) 注意

反映処理が正常終了又はイベント終了以外で停止している場合に、反映グループ定義を変更すると、前回までの反映処理と定義変更後の反映処理との間で不整合が発生する場合があります。

- **UOC による反映がある場合の反映処理の方式**

UOC による反映がある場合、反映システム定義、イベントでの指定、及び `hdsrfctl` コマンドでの指定に関係なく、表単位反映方式で反映処理が実行されます。

5.11 更新情報定義

SAM ファイルを使う抽出側 DB が PDMII E2 (VOS3, VOS1) のときは、反映側 Datareplicator に更新情報定義を指定します。

抽出側 DB が RDB1 E2 のときは、更新情報定義は不要です。

PDMII E2 ログ形式 SAM ファイルからの更新情報の抽出に必要な情報を定義します。

5.11.1 構造と形式

(1) 構造

各定義文は、稼働環境定義文、抽出再定義文、抽出文の順序で記述します。更新情報定義の構造を次の図に示します。また、更新情報定義の内容と指定数を次の表に示します。

図 5-12 更新情報定義の構造

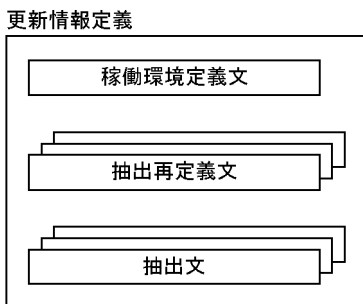


表 5-26 更新情報定義の内容と指定数

定義名	定義文	指定数	内容
稼働環境定義文	filetype 文	1	対象になるファイル種別を指定します。
抽出再定義文	restruct 文	0~4096	複数フィールドを1フィールド、又は1フィールドを複数フィールドとする場合に再定義フィールド名、データ開始位置、フィールド属性、長さをデータセット単位に指定します。
抽出文	extract 文	1~4096	抽出対象 DBM、データセット、フィールド、更新情報名及びマッピングキーを指定します。

(2) 形式

/* 稼働環境定義文 */

filetype PDM

/* 抽出再定義文 */

```

{{{ restrict DBM 名称.データセット名
      {{ field 再定義フィールド名称 [ position データ開始位置 ]
          attr 再定義フィールド属性 (長さ) [ not null ]}}...}}...
/* 抽出文 */
{{ extract DBM 名称.データセット名
      ( { { フィールド名称 | 再定義フィールド名称 }
        [ {{, { フィールド名称 | 再定義フィールド名称 }} ...] | * } )
      to 更新情報名
      key ( { フィールド名称 | 再定義フィールド名称 }
           [ {{, { フィールド名称 | 再定義フィールド名称 }} ...] )}}...
;

```

5.11.2 稼働環境定義文

更新情報入力の稼働環境についての情報を指定します。

(1) 形式

```
filetype PDM
```

(2) オペランドの説明

- filetype PDM

入力する SAM ファイルの種別を指定します。filetype オペランドは、1 件の更新情報定義に対して 1 件だけ指定します。指定できるのは、PDMII E2 ログ形式 SAM ファイルだけです。

PDM

PDMII E2 ログ形式 SAM ファイル

5.11.3 抽出再定義文

PDMII E2 ログ形式 SAM ファイルの場合、PDMII E2 の定義に指定した 1 フィールドを複数フィールド、複数フィールドを 1 フィールド、又はフィールドから部分的にデータを抽出する場合のフィールドを指定します。抽出再定義文はデータセット単位に必要です。指定する定義は順不同です。

抽出再定義文を省略すると、PDMII E2 で定義したフィールドをそのまま更新情報定義に使う更新処理を実行します。

(1) 形式

```

{{{ restruct DBM 名称.データセット名

    {{ field 再定義フィールド名称 [ position データ開始位置 ]

        attr 再定義フィールド属性 (長さ) [ not null ]}}...}}...
    
```

(2) オペランドの説明

DBM 名称、データセット名、再定義フィールド名称は、それぞれ反映側 HiRDB の認可識別子、表名、列名に対応させてください。

- restruct DBM 名称.データセット名
～ 〈1～6 文字の記号名称〉

PDMII E2 ログ形式の場合に DBM 名称とデータセット名を指定します。データセットがコーディッドレコード機能を使ったバリアブルデータセットの場合のデータセット名は、データセット名の後ろにレコードコードの値を付けた名称を指定します。

- field 再定義フィールド名称
～ 〈1～30 文字の記号名称〉

PDMII E2 ログ形式の場合に、再定義フィールド名称を指定します。

- position データ開始位置
～ 〈0〉

再定義フィールドに対応する行データの先頭 (= 0) からの位置を設定します。

- attr 再定義フィールド属性 (長さ) [not null]
再定義フィールド属性と長さを指定します。

not null は、更新データが '4040...' の場合にこのデータを初期値 (null データ) としないで実データとして扱うときに指定します。ただし、パック形式とゾーン形式のデータには、この指定は使えません。指定できるフィールド属性と長さを次の表に示します。

表 5-27 再定義フィールドに指定できるフィールド属性と長さ

フィールド属性 (長さ)	説明	フィールド (バイト)
CHAR(n)	文字列データ n : 文字数 (1 ≤ n ≤ 30000)	n
NCHAR(n)	日本語文字列データ n : 文字数 (1 ≤ n ≤ 15000)	n × 2
PACK(m [, n])	符号ありパック形式 10 進データ	(m + [n] + 1) ÷ 2

フィールド属性 (長さ)	説明	フィールド (バイト)
	m: 整数部 n: 小数部 ($0 \leq m \leq 29, 0 \leq n \leq 29, 1 \leq m + n \leq 29$)	小数点以下切り上げ
PACKNS(m [, n])	符号なしパック形式 10 進データ m: 整数部 n: 小数部 ($0 \leq m \leq 29, 0 \leq n \leq 29, 1 \leq m + n \leq 29$)	$(m + [n] + 1) \div 2$ 小数点以下切り上げ
UNPACK(m, [, n])	符号ありゾーン形式 10 進データ m: 整数部 n: 小数部 ($0 \leq m \leq 29, 0 \leq n \leq 29, 1 \leq m + n \leq 29$)	m + [n]
UNPACKNS(m [, n])	符号なしゾーン形式 10 進データ m: 整数部 n: 小数部 ($0 \leq m \leq 29, 0 \leq n \leq 29, 1 \leq m + n \leq 29$)	m + [n]
COMP(m [, n])	COBOL 2 進固定小数点データ m: 整数部 n: 小数部 ($0 \leq m \leq 18, 0 \leq n \leq 18, 1 \leq m + n \leq 18$)	$1 \leq m [+n] \leq 4$ の場合 : 2 $5 \leq m [+n] \leq 9$ の場合 : 4 $10 \leq m [+n] \leq 18$ の場合 : 8
BINARY(m [, n])	PL/I 2 進固定小数点データ m: 精度 n: 位取り ($1 \leq m \leq 31, 0 \leq n \leq 31, m \geq n$)	$1 \leq m \leq 15$ の場合: 2 $16 \leq m \leq 31$ の場合: 4
FLOAT(n)	浮動小数点データ n: バイト数 (n = 4, 8)	n

5.11.4 抽出文

PDMII E2 ログ形式 SAM ファイルの場合、抽出対象になる DBM 名称、データセット名 及びフィールド名称を指定します。抽出対象の DBM 名称とデータセット名が複数存在する場合は、抽出文を複数指定します。1 データセットに対しては、抽出文を 1 件だけ指定します。

1 件のデータセットを複数の表に反映する場合は、反映定義で指定します。

(1) 形式

```
{ extract DBM 名称.データセット名
```

```
( { { フィールド名称 | 再定義フィールド名称 }
```

[{{, {フィールド名称 | 再定義フィールド名称 }} …] | * }

to 更新情報名

key ({フィールド名称 | 再定義フィールド名称 }

{{, {フィールド名称 | 再定義フィールド名称 }} …)}}…

(2) オペランドの説明

- extract DBM 名称.データセット名

～ 〈1～6 文字の記号名称〉

PDMII E2 ログ形式の場合に DBM 名称とデータセット名を指定します。データセットがコーディッドレコード機能を使ったバリアブルデータセットの場合のデータセット名は、データセット名の後ろにレコードコードの値を付けた名称を指定します。

- フィールド名称 | 再定義フィールド名称

フィールド名称 ～ 〈1～30 文字の記号名称〉

フィールド名称と再定義フィールド名称を指定します。指定順は順不同です。ただし、同じ名称は複数回指定できません。

全列指定 (* 指定) できるのは、抽出再定義文がない場合だけです。

- to 更新情報名

～ 〈1～8 文字の記号名称〉

抽出した更新情報に対する更新情報名を指定します。更新情報名は、更新情報定義中のすべての抽出文で一意になる名称を指定してください。

- key フィールド名称 | 再定義フィールド名称

フィールド名称 ～ 〈1～30 文字の記号名称〉

マッピングキーに該当するフィールドと再定義フィールドを指定します。key 句で指定するフィールドと再定義フィールドは、extract 文で指定している名称でなければなりません。extract 文での指定が全列指定 (* 指定) の場合は、PDMII E2 の定義中の一意になるフィールドを指定してください。

(3) 注意

- マッピングキー列だけの抽出はできません。
- DBM 名称、データセット名、及びフィールド名称は必ず大文字で指定してください。
- 再定義フィールド名称と更新情報名は、英大文字と小文字を区別します。
- DBM 名称、データセット名、フィールド名称、及び再定義フィールド名称に Datareplicator の予約語と同じ名称を指定する場合は、名称を「"」(ダブルクォーテーション) で囲んで指定してください。Datareplicator の予約語については、「付録 B Datareplicator 定義の予約語」を参照してください。
- extract オペランドで指定するフィールドと再定義フィールドは、同じ列名を重複して指定できません。また、key オペランドで指定するフィールド、再定義フィールドも、重複した名称を指定できません。

- マッピングキーに対して更新した場合、更新後のデータをキーとして反映側で処理が実行されるため、反映処理でエラー（キーが一致するデータなし）又は意図したものとは異なる行に対して反映されることとなります。そのため、マッピングキーに対する更新は実行しないでください。
- key オペランドで指定できるマッピングキー列数は、最大 16 個です。
- 次の属性の列をマッピングキーに指定する場合、列の長さは 255 バイト以下にしてください。
 - CHAR
 - NCHAR

5.12 二重化定義（反映側）

5.12.1 二重化定義（反映側）の形式と説明

反映側システムの二重化定義で使用する論理ファイル名と物理ファイル名の対応を定義します。

(1) 形式

```
{ { logical_file = 論理ファイル名
  physical_file_a = 物理ファイル名1
  physical_file_b = 物理ファイル名2
# コメント行
} }
:
```

定義文の1行の長さは半角1,024文字以内です。コメントを記述する場合は、行の先頭に「#」を記述してください。

(2) オペランドの説明

`logical_file` = 論理ファイル名

二重化するファイルの論理ファイル名を指定します。ファイルごとの論理ファイル名を次に示します。

対象システム	定義ファイル	論理ファイル名
反映側システム	反映マスタステータスファイル	hdsinitstate
	反映ステータスファイル	反映環境定義に指定した名称
	反映情報キューファイル	反映環境定義に指定した名称

論理ファイル名を重複して定義した場合は、定義解析エラーが発生します。また、反映環境定義ファイル中にない論理ファイル名称を定義した場合、その論理ファイル名に関する定義は無視されます。

`physical_file_a` = 物理ファイル名1, `physical_file_b` = 物理ファイル名2

～((1～125)) (単位: バイト)

`logical_file` で指定した論理ファイルを構成する物理ファイル名を絶対パスで指定します。物理ファイル名を重複して定義した場合は、定義解析エラーが発生します。

5.13 Datareplicator の定義例

Datareplicator の定義例について説明します。

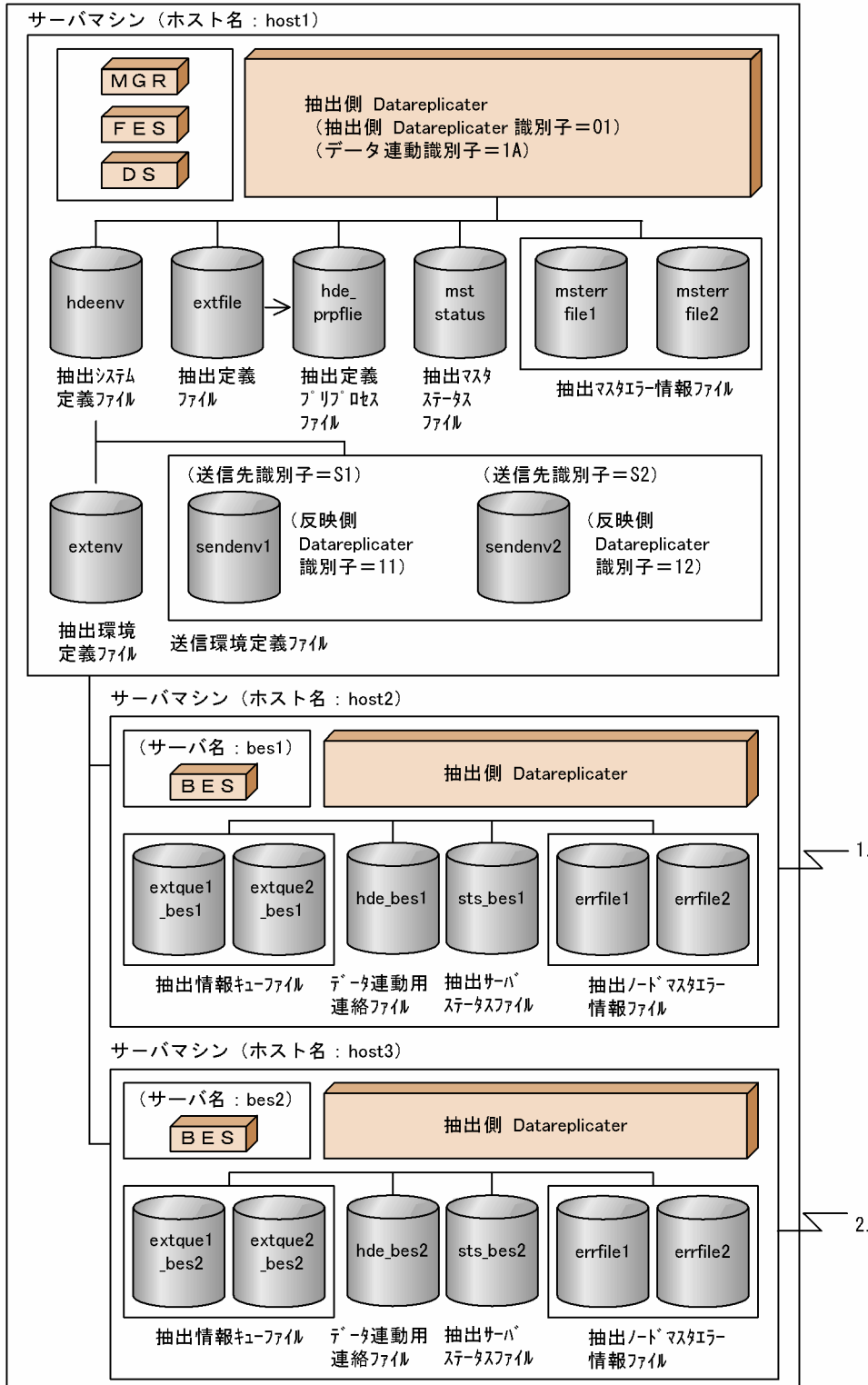
メインフレーム側のデータ連動システムの定義例については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

5.13.1 システムの対応とファイルの構成例

システムの対応とファイルの構成例を次の図に示します。この例では、抽出側システムと反映側システムの両方が Datareplicator の例で示します。

図 5-13 システムの対応とファイルの構成例 1

抽出側システム



(凡例)

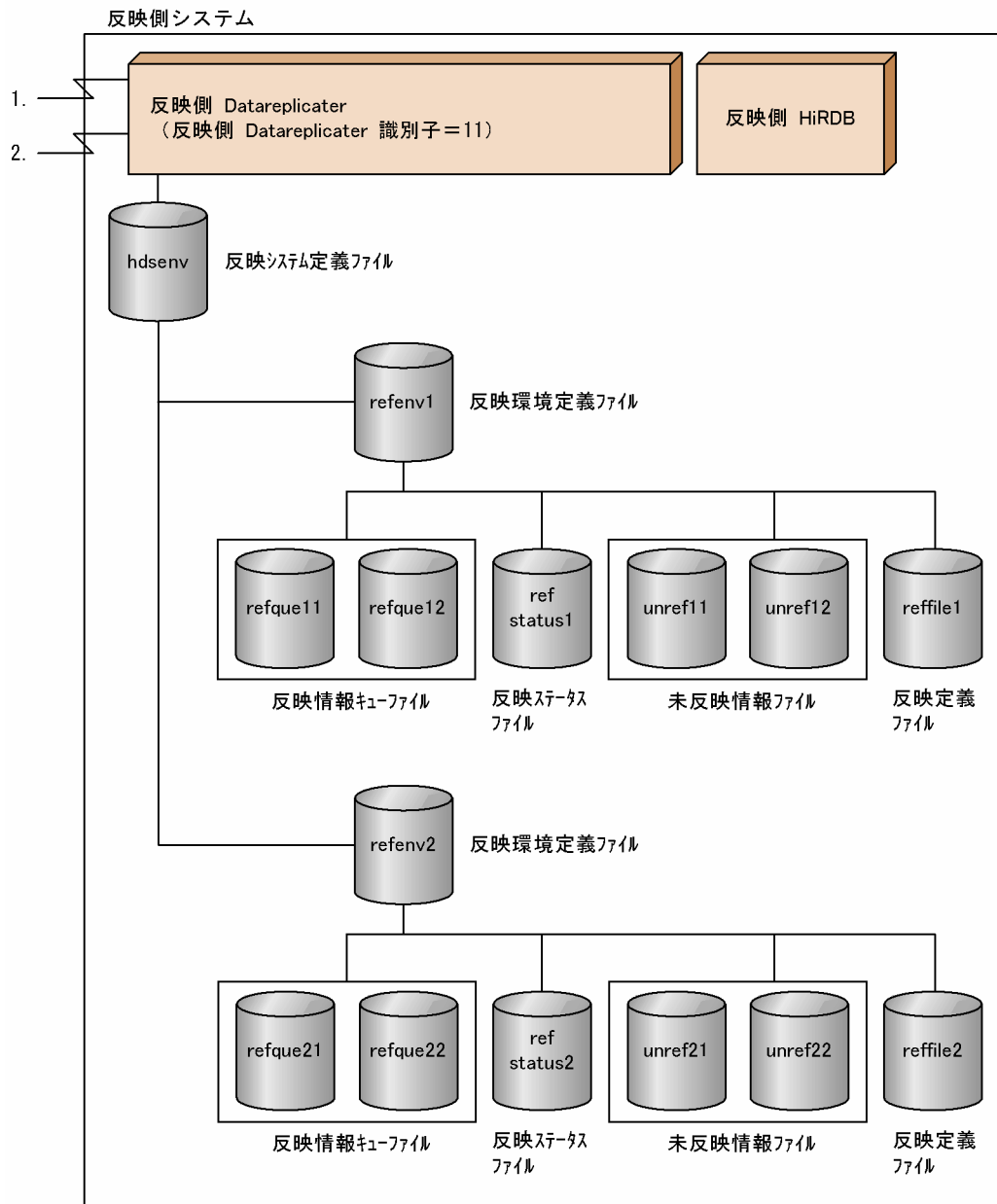
MGR : HiRDBのシステムマネージャ

FES : HiRDBのフロントエンドサーバ

DS : HiRDBのディクショナリサーバ

BES : HiRDBのバックエンドサーバ

図 5-14 システムの対応とファイルの構成例 2



5.13.2 抽出側 Datareplicator の定義の例

抽出側 Datareplicator の定義のうち、抽出システム定義、抽出環境定義、及び送信環境定義の例を説明します。抽出定義の例については、「5.13.4 抽出定義と反映定義の例」を参照してください。

抽出側 Datareplicator では、抽出側 Datareplicator の定義のテンプレートを用意しています。テンプレートを \$HDEPATH/ディレクトリ下にコピーして、ユーザの環境用に修正して利用できます。

UNIX システムで EUC コード体系を使ってテンプレートを利用する場合には、文字コードを変換するための OS コマンド (iconv など) を使う必要があります。文字コードを変換するための OS コマンドについては、該当する OS のリファレンスマニュアルを参照してください。

抽出側 Datareplicator の定義のテンプレートは、次に示すディレクトリに格納してあります。

- UNIX 版の場合： /opt/hirdbds/lib/sample/ディレクトリ
- Windows 版の場合：Datareplicator をインストールしたディレクトリ¥sample¥ディレクトリ

抽出側 Datareplicator の定義のテンプレートのファイル名を次の表に示します。

表 5-28 抽出側 Datareplicator の定義のテンプレートのファイル名

定義名	ファイル名	記述方式
抽出システム定義	hdeenv	<ul style="list-style-type: none"> • コメントには、JIS8/シフト JIS コード体系の文字を使っています。 • set で始まる行は、省略できないオペランドです。 • #set で始まる行は、省略できるオペランドです。 • 指定範囲と省略時仮定値は、次のように記述しています。 《指定範囲；省略時仮定値》 • 省略時仮定値のあるオペランドには、設定値として省略時仮定値を記述しています。省略時仮定値のないオペランドには、設定値を記述していません。 • オペランドの制限長が 80 文字までのため、ルーラを付けています。
抽出環境定義	extenv	
送信環境定義	sendenv	
抽出定義	extfile	<ul style="list-style-type: none"> • コメントには、JIS8/シフト JIS コード体系の文字を使っています。 • 抽出定義では、抽出文は省略できませんが、ユーザの定義に依存するため、定義例はすべてコメントとして記述しています。ただし、定義の終了を示す「;」だけは、コメントとして記述していません。 • 全般規則を先頭に記述しています。

(1) 抽出システム定義の例

抽出システム定義の例を次の図に示します。

図 5-15 抽出システム定義の例

抽出システム定義

```

set hdeid=01          #チュウシュツカ°リンキヘ°ツシ
set extdef=extenv    #チュウシュツカンキョウテイ°
set sendid01=S1      #ソウシンサキキヘ°ツシ
set sendid02=S2
set senddef01=sendenv1  #ソウシンカンキョウテイ°
set senddef02=sendenv2
set errfilesz=32     #エラーファイルサイズ°
set syslogout=true   #システムログ°ファイル
set watchintvl=30    #エラーカンジ°カン
set mstservice=extmst #ツウシンサービ°スメイ
set extinfunum=1000  #サイタ°イコウシンジ°ヨウホウメイスウ
set syncterm=true    #ジ°ト°ウテイ

```

(2) 抽出環境定義の例

抽出環境定義の例を次の図に示します。

図 5-16 抽出環境定義の例

抽出環境定義

```
commondef                #キョウウツウテイキ°
set  qufile001=extque1   #チュウシユツシ° ヨウホウキョウフファイルメイ
set  qufile002=extque2
besdef(bes1)              #bes1ヘノコヘ° ツテイキ°
set  dsid=11             #テ°-ダレント° ウシキヘ° ツシ
set  queuesize=4096      #キョウフファイルサイズ°
set  logiosize=128       #システムログ° ioハ° ッファサイズ°
set  quiosize=128        #キョウ-ioハ° ッファサイズ°
besdef(bes2)              #bes2ヘノコヘ° ツテイキ°
set  dsid=12             #テ°-ダレント° ウシキヘ° ツシ
set  queuesize=2048      #キョウフファイルサイズ°
set  logiosize=64        #システムログ° ioハ° ッファサイズ°
set  quiosize=64         #キョウ-ioハ° ッファサイズ°
```

(3) 送信環境定義の例

(a) 送信先識別子 S1 の場合の送信環境定義の例

送信先識別子 S1 の場合の送信環境定義の例を次の図に示します。

図 5-17 送信環境定義の例 (送信先識別子 S1 の場合)

送信環境定義 (送信先識別子 S1)

```
commondef                #キョウウツウテイキ°
set  sendhdsid=11        #アイテシステムシキヘ° ツシ
set  hdeservice=hdesserv1 #ツウシンサービ° スメイ
set  hdehost=hdeshst1   #ツウシンサキホストメイ
set  keepalive=true     #キープ° アライブ°
set  retrynum=5         #コネクションリトライカイスウ

besdef(bes1)              #bes1ヘノコヘ° ツテイキ°
set  sendintvl=20        #ツウシンカンカク
set  overwrite=false    #チュウシユツシ° ヨウホウキョウウリカ° キ
set  maxtran=200         #ト° ウシ° トランサ° クションスウ
set  maxtrandata=1000    #トランサ° クションナイサイタ° イコウシンジ° ヨウホウスウ
set  editbufsize=1024    #ヘンシユウハ° ッファサイズ°
set  readbufnum=8        #キョウ-ioハ° ッファスウ

besdef(bes2)              #bes2ヘノコヘ° ツテイキ°
set  sendintvl=10        #ツウシンカンカク
set  overwrite=true     #チュウシユツシ° ヨウホウキョウウリカ° キ
set  maxtran=100         #ト° ウシ° トランサ° クションスウ
set  maxtrandata=1000    #トランサ° クションナイサイタ° イコウシンジ° ヨウホウスウ
set  editbufsize=512     #ヘンシユウハ° ッファサイズ°
set  readbufnum=4        #キョウ-ioハ° ッファスウ
```

(b) 送信先識別子 S2 の場合の送信環境定義の例

送信先識別子 S2 の場合の送信環境定義の例を次の図に示します。

図 5-18 送信環境定義の例 (送信先識別子 S2 の場合)

送信環境定義(送信先識別子 S2)

```

commondef                #キョウウツテイキ
set sendhdsid=12         #アイチシステムシキヘツシ
set hdeservice=hdeserv2  #ツウシンサービスマイ
set hdehost=hdesht2     #ツウシンサキホストマイ
set keepalive=true      #キーフアライフ
set retrynum=5          #コネクションリトライカイスウ

besdef(bes1)             #bes1ヘノコヘツテイキ
set sendintvl=20        #ツウシンカンカク
set overwrite=false     #チュウシユツシヨウホウキョウウリカキ
set maxtran=200         #トウジトランザクシヨンスウ
set maxtranda=1000     #トランザクシヨナイサイダイコウシンジヨウホウスウ
set editbufsize=1024   #ヘンシユウハツファサイス
set readbufnum=8       #キョーイオハツファサウス

besdef(bes2)             #bes2ヘノコヘツテイキ
set sendintvl=10        #ツウシンカンカク
set overwrite=true     #チュウシユツシヨウホウキョウウリカキ
set maxtran=100         #トウジトランザクシヨンスウ
set maxtranda=1000     #トランザクシヨナイサイダイコウシンジヨウホウスウ
set editbufsize=512   #ヘンシユウハツファサイス
set readbufnum=4       #キョーイオハツファサウス
    
```

5.13.3 反映側 Datareplicator の定義の例

反映側 Datareplicator の定義のうち、反映システム定義と反映環境定義の例を説明します。反映定義の例については、「5.13.4 抽出定義と反映定義の例」を参照してください。

反映側 Datareplicator では、反映側 Datareplicator の定義のテンプレートを用意しています。テンプレートを \$HDSPATH/ディレクトリ下にコピーして、ユーザの環境用に修正して利用できます。

UNIX システムで EUC コード体系を使ってテンプレートを利用する場合には、OS の stou コマンドを使う必要があります。stou コマンドについては、該当する OS のリファレンスマニュアルを参照してください。

反映側 Datareplicator の定義のテンプレートは、次に示すディレクトリに格納してあります。

- UNIX 版の場合： /opt/hirdbds/lib/sample/ディレクトリ
- Windows 版の場合：Datareplicator をインストールしたディレクトリ¥sample¥ディレクトリ

反映側 Datareplicator の定義のテンプレートのファイル名を次の表に示します。

表 5-29 反映側 Datareplicator の定義のテンプレートのファイル名

定義名	ファイル名	記述方式
反映システム定義	hdsenv	<ul style="list-style-type: none"> • コメントには、JIS8/シフト JIS コード体系の文字を使っています。 • set で始まる行は、省略できないオペランドです。 • #set で始まる行は、省略できるオペランドです。
反映環境定義	refenv	

定義名	ファイル名	記述方式
		<ul style="list-style-type: none"> 指定範囲と省略時仮定値は、次のように記述しています。 《指定範囲；省略時仮定値》 省略時仮定値のあるオペランドには、設定値として省略時仮定値を記述しています。省略時仮定値のないオペランドには、設定値を記述していません。 オペランドの制限長が 80 文字までのため、ルーラを付けています。
反映定義	reffile	<ul style="list-style-type: none"> コメントには、JIS8/シフト JIS コード体系の文字を使っています。 反映定義はすべて省略できるため、定義例はすべてコメントとして記述しています。ただし、定義の終了を示す「;」だけは、コメントとして記述していません。 全般規則を先頭に記述しています。

(1) 反映側システム 1 の場合

(a) 反映システム定義の例

反映側システム 1 の場合の反映システム定義の例を次の図に示します。

図 5-19 反映システム定義の例（反映側システム 1 の場合）

反映システム定義（反映側システム 1）

```

set hdsid=11          #ハンエイカ°ワシキハ°ツシ
set hirdbusr=ref11   #hirdbニカシキハ°ツシ
set dsid001=1A       #テ°タレント°ウシキハ°ツシ
set refenv001=refenv #ハンエイカシキョウテイキ°
set hdsservice=hdsserv1 #ツウシサ°ヒ°スメイ
set keepalive=true   #キープ°アライフ°
set errfilesz=32     #エラー°ファイル
set syslogout=true   #システムログ°ファイル

```

(b) 反映環境定義の例

反映側システム 1 の場合の反映環境定義の例を次の図に示します。

図 5-20 反映環境定義の例（反映側システム 1 の場合）

反映環境定義（反映側システム 1）

```

set qufile001=refque1 #ハンエイシ°ヨウホウキュー°ファイルメイ
set qufile002=refque2
set qufilesize=5000   #キュー°ファイルサイズ°
set reffile=reffile   #ハンエイテイキ°ファイルメイ
set statsfile=refstatus #ハンエイステータス°ファイルメイ
set unreffile1=unref1 #ミハンエイシ°ヨウホウ°ファイルメイ
set unreffile2=unref2
set unreffilesz=150   #ミハンエイシ°ヨウホウ°ファイルサイズ°
set startmode=trn     #スタート°モード°
set defmerge=true     #チュウシツク°テイキ°カテイ
set cmtintvl=20       #コミット°カンカク
set tblcheck=true     #テーブル°チェック
set defshmsize=500    #テイキ°シ°ヨウホウ°カクノウ°キョウヨウ°メモリ°サイズ°

```

(2) 反映側システム 2 の場合

(a) 反映システム定義の例

反映側システム 2 の場合の反映システム定義の例を次の図に示します。

図 5-21 反映システム定義の例 (反映側システム 2 の場合)

反映システム定義 (反映側システム 2)

```
set hdsid=12          #ハンエイジ`ウシキヘ`ツシ
set hirdbusr=ref12   #hirdbニシカシキヘ`ツシ
set dsid001=1A       #デー`タレント`ウシキヘ`ツシ
set refenv001=refenv #ハンエイジ`キョウタイキ`
set hdsservice=hdsserv2 #ツシ`サ`ハ`ヒ`スメイ
set keepalive=true   #キープ`アライブ`
set errfilesz=32     #エラー`ファイル
set syslogout=true   #システムログ`ファイル
```

(b) 反映環境定義の例

反映側システム 2 の場合の反映環境定義の例を次の図に示します。

図 5-22 反映環境定義の例 (反映側システム 2 の場合)

反映環境定義 (反映側システム 2)

```
set qufile001=refque1 #ハンエイジ`ヨウホウ`キュー`ファイル`メイ
set qufile002=refque2
set qufilesize=5000   #キュー`ファイル`サイズ`
set reffile=reffile   #ハンエイジ`テキ`ファイル`メイ
set statsfile=refstatus #ハンエイジ`ステータス`ファイル`メイ
set unreffile1=unref1 #ハンエイジ`ヨウホウ`ファイル`メイ
set unreffile2=unref2
set unreffilesz=150   #ハンエイジ`ヨウホウ`ファイル`サイズ`
set startmode=trn     #スタート`モード`
set defmerge=true     #チェウ`シユ`ツタイキ`カタイ
set cmtintvl=20       #コミット`カンカク
set tblcheck=true     #テーブル`チェック
set defshmsize=500    #テキ`シ`ヨウホウ`カクノウ`キョウ`ヨウ`メモリ`サイズ`
```

5.13.4 抽出定義と反映定義の例

次に示す抽出定義と反映定義の例について説明します。

- 抽出対象表の列を抽出側で選択して並べ替えてデータ連動する場合
- 抽出対象表の列を反映側で選択して並べ替えてデータ連動する場合
- 抽出対象表に反映側で列 (固定値) を追加してデータ連動する場合
- 一つの表を反映側で二つに分けてデータ連動する場合
- 一つの表を抽出側で二つに分けてデータ連動する場合
- 二つの表を反映側で一つにしてデータ連動する場合

ここで示す例では、抽出対象表の伝票番号（DNO）をマッピングキーとしています。

(1) 抽出側での列の選択と並べ替え

抽出対象表の列を抽出側で選択して並べ替えてデータ連動する場合の抽出定義と反映定義の例を次の図に示します。

図 5-23 抽出定義と反映定義の例（抽出側での列の選択と並べ替え）

抽出対象の表：N1. JUCYU

DNO	TCODE	SCODE	JSURYO
伝票番号	得意先コード	商品コード	受注量
026551	TT002	101M	10
026552	TT002	591M	25
026553	TH001	353M	8

⋮

抽出定義

```
extract
N1. JUCYU (SCODE, JSURYO, DNO) /* センタナラビ`カエ */
to UJUCYU /* コウシンジ`ヨウホウ */
key (DNO) /* マッピ`ング`キー */

send S1 /* ソウシ`ン`サキ */
from UJUCYU ; /* コウシンジ`ヨウホウ */
```

更新情報：UJUCYU

SCODE	JSURYO	DNO
-------	--------	-----

送信先識別子：S1



更新情報：UJUCYU

SCODE	JSURYO	DNO
-------	--------	-----

反映対象の表：N1. JUCYU1

商品コード	受注量	伝票番号
101M	10	026551
591M	25	026552
353M	8	026553

⋮

反映定義

```
load
from UJUCYU /* コウシンジ`ヨウホウ */
to N1. JUCYU1 ; /* ハソエ`ヒョウ */
```

(2) 反映側での列の選択と並べ替え

抽出対象表の列を反映側で選択して並べ替えてデータ連動する場合の抽出定義と反映定義の例を次の図に示します。

図 5-24 抽出定義と反映定義の例（反映側での列の選択と並べ替え）

抽出対象の表：N1. JUCYU

DNO	TCODE	SCODE	JSURYO
伝票番号	得意先コード	商品コード	受注量
026551	TT002	101M	10
026552	TT002	591M	25
026553	TH001	353M	8

⋮

抽出定義

```

extract
N1. JUCYU(*)           /* スペテチュウシュウ */
to UJUCYU             /* コウシンジヨウホウ */
key (DNO)             /* マツビンカキ */

send S1               /* ソウシキ */
from UJUCYU ;        /* コウシンジヨウホウ */
    
```

更新情報：UJUCYU

DNO	TCODE	SCODE	JSURYO
-----	-------	-------	--------

送信先識別子：S1



更新情報：UJUCYU

SCODE	JSURYO	DNO
-------	--------	-----

反映対象の表：N1. JUCYU1

商品コード	受注量	伝票番号
101M	10	026551
591M	25	026552
353M	8	026553

⋮

反映定義

```

format UJUCYU         /* ジェチュウコウシンジヨウホウ */
name DNO              /* テンビヨウファイル */
name TCODE            /* トクイファイル */
name SCODE            /* ショウビンファイル */
name JSURYO          /* ジェチュウリヨウファイル */

load SCODE, JSURYO, DNO /* センタナラヘカエ */
from UJUCYU
to N1. JUCYU1 ;      /* ハンエイヨウ */
    
```

(3) 反映側での列（固定値）の追加

抽出対象表に反映側で新しく列（固定値）を追加してデータ連動する場合の抽出定義と反映定義の例を次の図に示します。

図 5-25 抽出定義と反映定義の例（反映側での列（固定値）の追加）

抽出対象の表：N1.JUCYU

DNO	TCODE	SCODE	JSURYO
伝票番号	得意先コード	商品コード	受注量
026551	TT002	101M	10
026552	TT002	591M	25
026553	TH001	353M	8

⋮

抽出定義

```

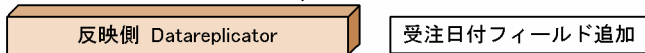
extract
N1.JUCYU(*)           /* スペテチュウシュツ */
to UJUCYU             /* コウシンジヨウホウ */
key(DNO)              /* マツヒンクキ */

send S1              /* ソウシホキ */
from UJUCYU ;        /* コウシンジヨウホウ */
    
```

更新情報：UJUCYU

DNO	TCODE	SCODE	JSURYO
-----	-------	-------	--------

送信先識別子：S1



更新情報：UJUCYU

DNO	TCODE	SCODE	JSURYO	JDATE
-----	-------	-------	--------	-------

反映対象の表：N1.JUCYU2

伝票番号	得意先コード	商品コード	受注量	受注日付
026551	TT002	101M	10	051015
026552	TT002	591M	25	051015
026553	TH001	353M	8	051015

⋮

反映定義

```

format UJUCYU          /* ジュチュウコウシンジヨウホウ */
name DNO               /* テンヒョウフィールド */
name TCODE             /* トクイフィールド */
name SCODE             /* ショウヒンフィールド */
name JSURYO            /* ジュチュウリョウフィールド */
name JDATE const '051015' /* ジュチュウヒョウフィールドツイカ */

load DNO, TCODE, SCODE, JSURYO, JDATE /* ナラヘカエナシ */
from UJUCYU
to N1.JUCYU2 ;
    
```

(4) 反映側で1表を2表に分割

一つの抽出対象表を反映側で二つの表に分けてデータ連動する場合の抽出定義と反映定義の例を次の図に示します。

図 5-26 抽出定義と反映定義の例（反映側で 1 表を 2 表に分割）

抽出対象の表：N1. JUCYU

DNO	TCODE	SCODE	JSURYO
伝票番号	得意先コード	商品コード	受注量
026551	TT002	101M	10
026552	TT002	591M	25
026553	TH001	353M	8

⋮

抽出定義

```

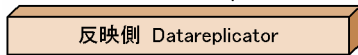
extract
N1. JUCYU(*)           /* スペテチュウシュツ */
to UJUCYU              /* コウシンジヨウホク */
key(DNO)               /* マッピングキー */

send S1                /* ソウシキ */
from UJUCYU ;         /* コウシンジヨウホク */
    
```

更新情報：UJUCYU

DNO	TCODE	SCODE	JSURYO
-----	-------	-------	--------

送信先識別子：S1



更新情報：UJUCYU

DNO	TCODE	SCODE	JSURYO
-----	-------	-------	--------

二つの表に反映

DNO	JSURYO
-----	--------

DNO	TCODE
-----	-------

反映対象の表：N1. RYO

伝票番号	受注量
026551	10
026552	25
026553	8

⋮

反映対象の表：N1. TOKUI

伝票番号	得意先コード
026551	TT002
026552	TT002
026553	TH001

⋮

反映定義

```

format UJUCYU
name DNO
name TCODE
name SCODE
name JSURYO
/* RYOへ/ハソエイ */
load DNO, JSURYO
from UJUCYU
to N1. RYO
/* TOKUIへ/ハソエイ */
load DNO, TCODE
from UJUCYU
to N1. TOKUI
group REFGROUP
by N1. RYO, N1. TOKUI ;
    
```

(5) 抽出側で 1 表を 2 表に分割

一つの抽出対象表を抽出側で二つの表に分けてデータ連動する場合の抽出定義と反映定義の例を次の図に示します。

図 5-27 抽出定義と反映定義の例 (抽出側で 1 表を 2 表に分割)

抽出対象の表 : N1. JUCYU

DNO	TCODE	SCODE	JSURYO
伝票番号	得意先コード	商品コード	受注量
026551	TT002	101M	10
026552	TT002	591M	25
026553	TH001	353M	8

⋮

二つに分割して抽出

更新情報 : UJUCYU1

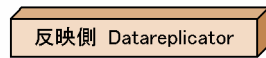
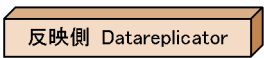
DNO	JSURYO
-----	--------

更新情報 : UJUCYU2

DNO	TCODE
-----	-------

送信先識別子 : S1

送信先識別子 : S2



更新情報 : UJUCYU1

DNO	JSURYO
-----	--------

更新情報 : UJUCYU2

DNO	TCODE
-----	-------

反映対象の表 : N1. RYO

反映対象の表 : N1. TOKUI

伝票番号	受注量
026551	10
026552	25
026553	8

伝票番号	得意先コード
026551	TT002
026552	TT002
026553	TH001

⋮

⋮

抽出定義

```

/* UJUCYU1へ/チュウシュツ */
extract
  N1. JUCYU (DNO, JSURYO)
  to UJUCYU1
  key (DNO)
/* UJUCYU2へ/チュウシュツ */
extract
  N1. JUCYU (DNO, TCODE)
  to UJUCYU2
  key (DNO)
/* S1へ/ワザシ */
send S1
  from UJUCYU1
/* S2へ/ワザシ */
send S2
  from UJUCYU2 ;
    
```

反映定義 (S1)

```

load *
  from UJUCYU1
  to N1. RYO;
    
```

反映定義 (S2)

```

load *
  from UJUCYU2
  to N1. TOKUI;
    
```

(6) 反映側で 2 表を 1 表に反映

二つの抽出対象表を反映側で一つにしてデータ連動する場合の抽出定義と反映定義の例を次の図に示します。

図 5-28 抽出定義と反映定義の例 (反映側で 2 表を 1 表に反映)

抽出対象の表 : N1. JUCYU

DNO	TCODE	SCODE	JSURYO
伝票番号	得意先コード	商品コード	受注量
026551	TT002	101M	10
026552	TT002	591M	25
026553	TH001	353M	8

抽出対象の表 : N1. ZAIKO

DNO	SCODE	FZSURYO
伝票番号	商品コード	在庫量
026551	101M	85
026552	591M	62
026553	353M	29

抽出定義

```

/* JUCYU */
extract
N1. JUCYU(*)
to UJUCYU
key (DNO)
/* ZAIKO */
extract
N1. ZAIKO(*)
to UZAIKO
key (DNO)
/* UJUCYU */
send S1
from UJUCYU
/* UZAIKO */
send S1
from UZAIKO;
    
```

更新情報 : UJUCYU

DNO	TCODE	SCODE	JSURYO
-----	-------	-------	--------

更新情報 : UZAIKO

DNO	SCODE	FZSURYO
-----	-------	---------

送信先識別子 : S1



更新情報 : UJUCYU

DNO	SCODE	JSURYO	DMYDATA1
-----	-------	--------	----------

更新情報 : UZAIKO

DNO	SCODE	DMYDATA2	FZSURYO
-----	-------	----------	---------

反映対象の表 : N1. JZHYO

伝票番号	商品コード	受注量	在庫量
026551	101M	10	85
026552	591M	25	62
026553	353M	8	29

反映定義

```

format UJUCYU
name DNO
name TCODE
name SCODE
name JSURYO
name DMYDATA1 const null
format UZAIKO
name DNO
name SCODE
name DMYDATA2 const null
name FZSURYO
/* JZHYO */
load DNO, SCODE, JSURYO, DMYDATA1
from UJUCYU
to N1. JZHYO
load DNO, SCODE, DMYDATA2, FZSURYO
from UZAIKO
to N1. JZHYO ;
    
```

5.13.5 マルチ FES 機能を使う場合の反映グループ定義の例

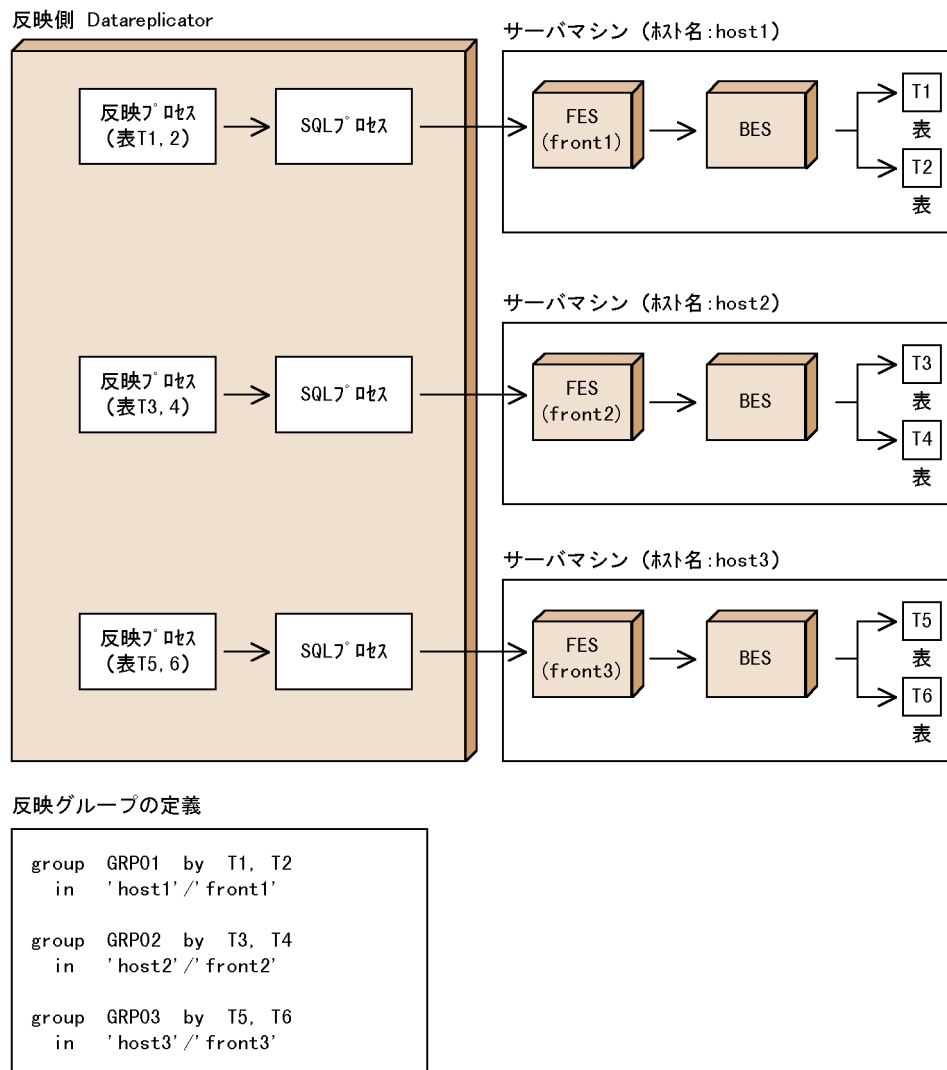
マルチ FES 機能を使う場合の、次に示す反映グループ定義の例について説明します。

- 表単位分割方式でマルチ FES 機能を使う場合
- キーレンジ単位分割方式でマルチ FES 機能を使う場合
- ハッシュ分割方式でマルチ FES 機能を使う場合

(1) 表単位分割方式でマルチ FES 機能を使う場合

表単位分割方式でマルチ FES 機能を使うときの反映グループ定義の例を次の図に示します。

図 5-29 反映グループ定義の例 (表単位分割方式)



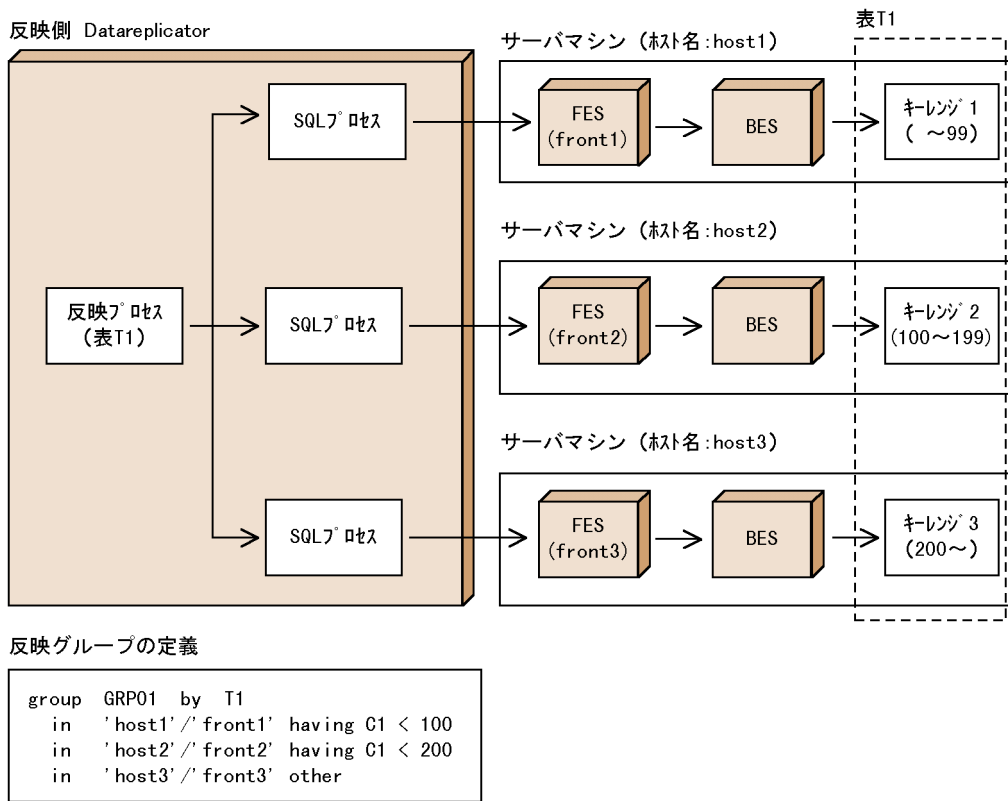
(凡例)

F E S : HiRDBのフロントエンドサーバ B E S : HiRDBのバックエンドサーバ

(2) キーレンジ単位分割方式でマルチ FES 機能を使う場合

キーレンジ単位分割方式でマルチ FES 機能を使うときの反映グループ定義の例を次の図に示します。

図 5-30 反映グループ定義の例 (キーレンジ単位分割方式)



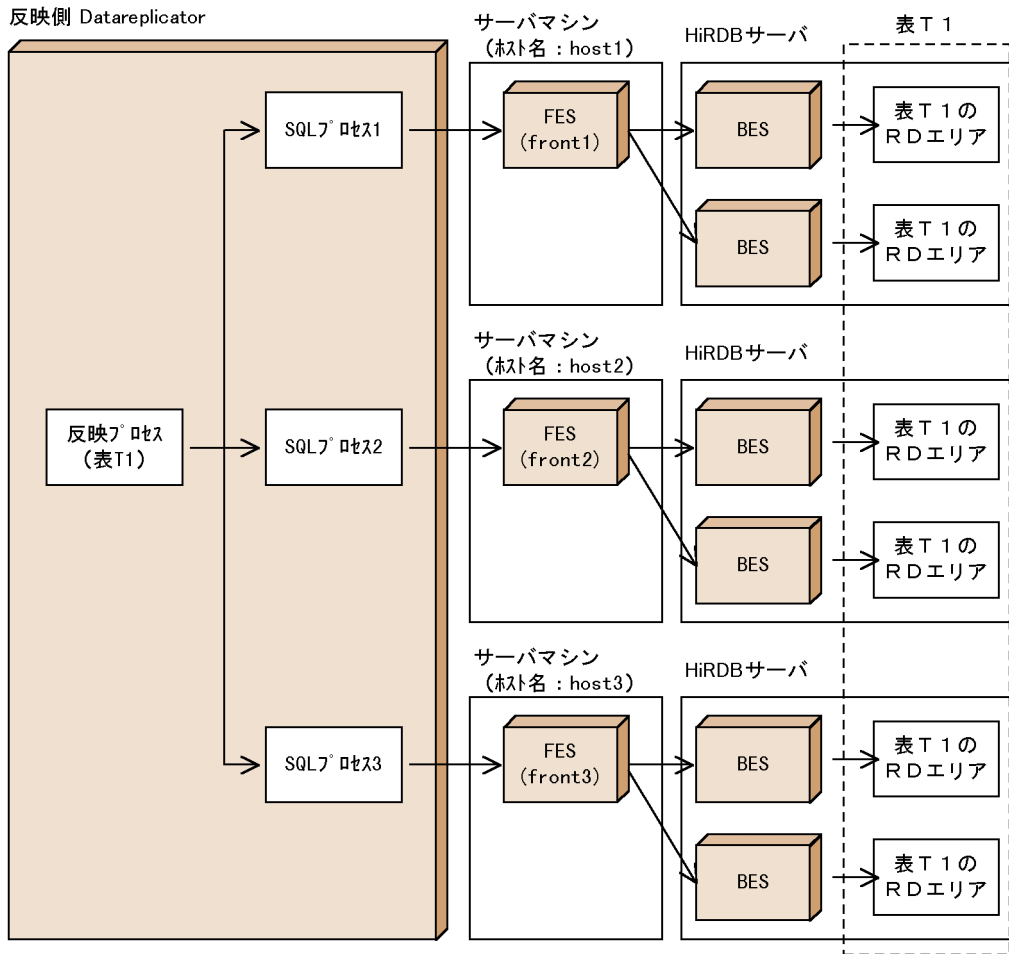
(凡例)

F E S : HiRDBのフロントエンドサーバ B E S : HiRDBのバックエンドサーバ

(3) ハッシュ分割方式でマルチ FES 機能を使う場合

ハッシュ分割方式でマルチ FES 機能を使うときの反映グループ定義の例を次の図に示します。

図 5-31 反映グループ定義の例 (ハッシュ分割方式)



反映グループの定義

```
group GRP02 by T1
  hash in 'host1'/'front1',
         'host2'/'front2',
         'host3'/'front3'
```

(凡例)

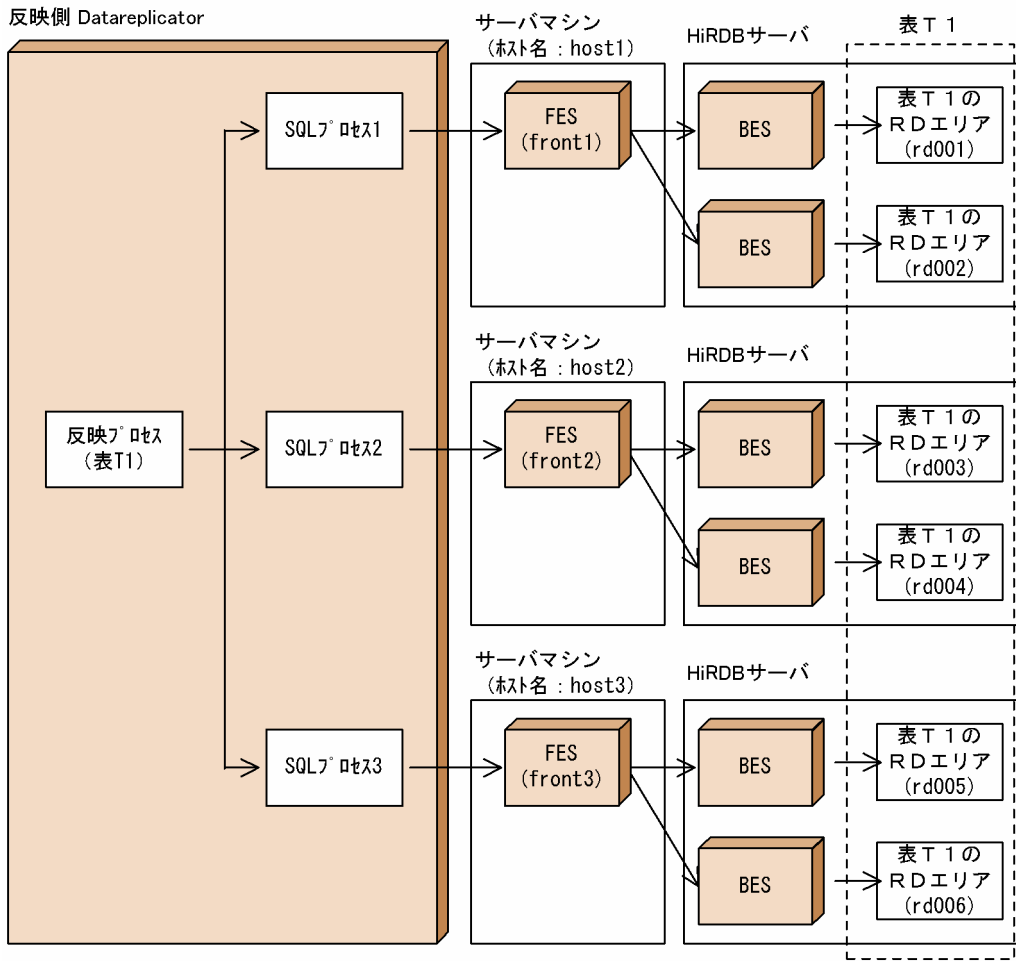
FES : HiRDBのフロントエンドサーバ BES : HiRDBのバックエンドサーバ

(4) ハッシュ分割方式で、FES を指定する場合

ハッシュ分割でデータを別々のサーバに格納するパラレルサーバの表に対して、Datareplicatorの反映グループでFESを指定します。このとき、HiRDBの表と同様にハッシュ分割をします。FESを指定すると、ハッシュ分割した数だけ並行に反映処理すると同時に、FESとBES間の通信オーバーヘッドを削減してFESの負荷を減らせます。これによって、データ連動をより効率的に実行できるようになります。

反映グループ定義の例 (ハッシュ分割方式でFESを指定する場合) を次の図に示します。

図 5-32 反映グループ定義の例 (ハッシュ分割方式で FES を指定する場合)



反映グループの定義

```
group GRP01 by T1
  hash in 'host1'/'front1' (rd001, rd002),
         'host2'/'front2' (rd003, rd004),
         'host3'/'front3' (other)
```

(凡例)

FES : HiRDBのフロントエンドサーバ BES:HiRDBのバックエンドサーバ

6

運用

この章では、データ連動システムを運用するための手順、Datareplicator の起動と終了、及び Datareplicator の運用について説明します。

6.1 データ連動システムの運用の概要

データ連動システムを運用する手順の概要について説明します。Datareplicator を使ったデータ連動システムの組み合わせを次に示します。

データ連動の形式	6.1 の参照先
<ul style="list-style-type: none">HiRDB から HiRDB へのデータ連動メインフレーム側 DB (XDM/SD E2, XDM/RD E2, ADM, PDMII E2, TMS-4V/SP) から HiRDB へのデータ連動HiRDB からメインフレーム側 DB (XDM/RD E2) へのデータ連動	6.1.1
<ul style="list-style-type: none">SAM ファイルを使ったメインフレーム側 DB から HiRDB へのデータ連動	6.1.2

6.1.1 通常のデータ連動システムの運用手順

通常のデータ連動システムの運用手順を次の図に示します。

図 6-1 通常のデータ連動システムの運用手順



(1) Datareplicator の初期化 (初めてデータ連動を開始する場合)

環境を構築した後に初めてデータ連動を開始する場合は、Datareplicator を初期化します。詳細については、「6.2 環境構築時の初期化手順」を参照してください。

(2) 反映側システムの起動

反映側システムの起動手順を次に示します。

1. 反映側 DB (HiRDB 又は XDM E2) を起動します。
2. 反映側のデータ連動製品 (Datareplicator 又は XDM/DS) を起動します。

抽出側システムからのデータ送信時に、送信先の反映側システムが起動していない場合、抽出側システムは送信間隔ごとに反映側システムの起動を確認して、起動が確認できた時点から送信を開始します。このため、抽出側システムからデータを送信できないと、抽出側システムの資源 (抽出情報キューファイルやシステムログファイル) が満杯になることがあります。システムを開始するときは、反映側システムが起動を完了してから、抽出側システムを起動することをお勧めします。

(3) 抽出側システムの起動

抽出側システムの起動手順を次に示します。

1. 抽出側のデータ連動製品 (Datareplicator 又は XDM/DS) を起動します。
2. 抽出側 DB (HiRDB 又は XDM E2) を起動します。

抽出側 Datareplicator は、抽出側 HiRDB が稼働しているかどうかに関係なく起動できます。ただし、抽出側 HiRDB の稼働中に抽出側 Datareplicator が稼働していないと、システムログファイルが満杯になって、抽出側 DB が強制停止されたりデータ連動ができなくなったりすることがあります。

(4) データ連動処理の実行

反映側システムと抽出側システムが正常に起動すると、システム定義に指定したタイミングでデータ連動が開始されます。

Datareplicator の抽出側システムでのデータ連動処理の運用については、「6.4.1 抽出処理の運用」を参照してください。また、反映側システムでのデータ連動処理の運用については、「6.7.1 反映処理の運用」を参照してください。

XDM/DS の抽出側システムと反映側システムでのデータ連動処理の運用については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 運用」を参照してください。

(5) 抽出側システムの終了

データ連動システムの業務を終了するときは、まず抽出側システムを終了します。抽出側システムは、次の順序で終了してください。

1. 抽出側 DB (HiRDB 又は XDM E2) を終了します。
2. 抽出側データ連動製品 (Datareplicator 又は XDM/DS) を終了します。

HiRDB の終了方法についてはマニュアル「HiRDB システム運用ガイド」を、抽出側 Datareplicator の終了については「6.3 抽出側 Datareplicator の起動と終了」を参照してください。

XDM/DS の終了方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 運用」を参照してください。

(6) 反映側システムの終了

反映側システムを終了します。反映側システムは、次の順序で終了してください。

1. 反映側データ連動製品 (Datareplicator 又は XDM/DS) を終了します。
2. 反映側 DB (HiRDB 又は XDM E2) を終了します。

反映側 Datareplicator の終了方法については、「6.6 反映側 Datareplicator の起動と終了」を、HiRDB の終了方法については、マニュアル「HiRDB システム運用ガイド」を参照してください。

XDM/DS の終了方法については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 運用」を参照してください。

6.1.2 SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動する場合の運用手順

SAM ファイルを使って、メインフレーム側 DB (PDMII E2, RDB1 E2) から HiRDB へデータ連動するときの運用手順を次の図に示します。

図 6-2 SAM ファイルを使って、メインフレーム側 DB から HiRDB へデータ連動するときの運用手順



注※

手順 2 と手順 3, 及び手順 5 と手順 6 は手順を入れ替えても運用できます。

(1) Datareplicator の初期化（初めてデータ連動を開始する場合）

環境を構築した後に初めてデータ連動を開始する場合は、Datareplicator を初期化します。詳細については、「6.2 環境構築時の初期化手順」を参照してください。

(2) 反映側システムの起動

反映側システムの起動手順を次に示します。

1. 反映側 DB（HiRDB）を起動します。
2. 反映側のデータ連動製品（Datareplicator）を起動します。

(3) 抽出側システムの起動

抽出側システムの起動手順を次に示します。

1. 抽出側 DB（PDMII E2 又は RDB1 E2）を起動します。

2. ファイル転送プログラムを起動します。

抽出側システムのプログラムの起動順序は、各製品の仕様に従います。

(4) データ連動処理の実行

抽出側システムではファイル転送プログラムを使って、更新情報を格納した SAM ファイルを反映側システムに送信します。送信された SAM ファイルは、抽出側システムのファイル転送プログラムが指定した反映側 Datareplicator のディレクトリに格納されます。

反映側 Datareplicator の操作者は、更新情報を SAM ファイルから反映情報キューファイルに格納するため、hdssamqin コマンドを実行します。反映情報キューファイルへの更新情報の格納が完了すると、反映側システムは反映処理を実行します。

(5) 抽出側システムの終了

抽出側 DB (PDMII E2 又は RDB1 E2) を終了し、抽出側システムを終了します。

抽出側システムのプログラムの終了順序は、各製品の仕様に従います。抽出側システムのプログラムの終了方法については、該当するマニュアルを参照してください。

(6) 反映側システムの終了

反映側システムを終了します。反映側システムは、次の順序で終了してください。

1. 反映側データ連動製品 (Datareplicator) を終了します。
2. 反映側 DB (HiRDB) を終了します。

反映側 Datareplicator の終了方法については、「6.6 反映側 Datareplicator の起動と終了」を、HiRDB の終了方法については、マニュアル「HiRDB システム運用ガイド」を参照してください。

6.2 環境構築時の初期化手順

Datareplicator の環境を構築した後に初めてデータ連動を開始する場合は、Datareplicator を初期化してください。

手順の流れを次の図に示します。

図 6-3 環境構築時の初期化手順



注

手順 6 以降を実施するには、反映側 DB を起動してください。

環境構築時の初期化手順を図に基づいて説明します。説明の番号は図中の番号に対応しています。

1. 次に示す Datareplicator の定義ファイルを作成します。
 - 抽出システム定義ファイル
 - 送信環境定義ファイル
 - 抽出環境定義ファイル
 - 抽出定義ファイル
 - 二重化定義ファイル（抽出側 Datareplicator 用）（任意）
 - 反映システム定義ファイル
 - 反映環境定義ファイル

- 反映定義ファイル
- 二重化定義ファイル（反映側 Datareplicator 用）（任意）

2. 抽出側 HiRDB を停止します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 HiRDB を正常停止します。	pdstop	syslog ファイル（Windows の場合はイベントログ）に KFPS01850-I メッセージが出力されていることを確認してください。

注

HiRDB が正常停止していないと、HiRDB の定義ファイルの変更が有効になりません。

3. 次に示す抽出側 HiRDB の定義ファイルを修正します。

詳細については、「5.6 抽出側 HiRDB の定義」を参照してください。

- システム共通定義の pd_rpl_init_start
（pd_rpl_init_start オペランドには Y を指定することを推奨します）
- システム共通定義の pd_log_rpl_no_standby_file_opr
- ユニット制御情報定義の pd_rpl_hdepath

4. 抽出側システムを初期化します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplicator を初期化します。	hdestart -i (応答メッセージに Y を入力します)	msterrfile1 又は msterrfile2 に KFRB00504-I メッセージが出力されていることを確認してください。KFRB00504-I メッセージが出力されていない、又は msterrfile1 及び msterrfile2 が作成されていない場合は、定義解析エラー、又はファイル生成に失敗しているおそれがあります。syslog ファイル（Windows の場合はイベントログ）を参照してエラー内容を確認し、対処した後に再度初期化を実行してください。

5. 抽出側 HiRDB を起動します。

抽出側 HiRDB を起動したら、手順 8 が完了するまで抽出側 DB に対する更新業務を行わないでください。

項番	作業内容	実行コマンド	確認項目
1	抽出側 HiRDB を起動します。	pdstart	syslog ファイル（Windows の場合はイベントログ）に KFPS05210-I メッセージが出力されていることを確認してください。
2	HiRDB Datareplicator 連携を開始します。 HiRDB のシステム共通定義の pd_rpl_init_start オペランドに N を	pd_rplstart	標準出力に KFPS05140-I メッセージが出力されていることを確認してください。

6. 運用

項番	作業内容	実行コマンド	確認項目
	指定しているか、指定を省略している場合に実施してください。		
3	HiRDB が Datareplicator とのデータ連携が可能な状態になっているか確認します。	pdls -d rpl -j	表示結果の内容※を確認してください。

注※

次の例を参考に、表示結果の(1)と(2)が Y となっていることを確認してください。また、(3)と(4)が示す Gen No と Block No が 0 となっていないことを確認してください。

```

SYSTEMID      : HRD1(183346)
Data replication : Y          ... (1)
UNITID       : unt1(183346)
Data replication : Y          ... (2)
SERVER NAME   : sds01
Extract Database : Y
Extract Status : C
System Log Extract Point :
Run ID  Group  Gen No.  BLock No.
4740e7a9 log24  18      2dd8      ... (3)
System Log Sync Info :
Run ID  Group  Gen No.  BLock No.
4740e7a9 log24  18      2dba      ... (4)

```

6. 抽出側 DB と反映側 DB のデータを一致させます。

HiRDB Dataextractor などを使用して、抽出対象表から反映対象表へデータの一括複写（データの初期作成）を行い、抽出側 DB と反映側 DB のデータを一致させてください。

7. 抽出定義プリプロセスファイルを作成します。

項番	作業内容	実行コマンド	確認項目
1	抽出定義プリプロセスファイルを作成します。	hdeprep -f 抽出定義ファイル	標準出力に KFRB04500-I メッセージが出力されていることを確認してください。

注

hdeprep コマンドには、必要に応じてオプションを付けてください。

8. 反映側 Datareplicator を初期化します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を初期化します。	hdsstart -i -f -q	標準出力（Windows の場合はイベントビューア）に KFRB04216-I メッセージが出力されていることを確認してください。 KFRB04216-I メッセージが出力されていない場合は、定義解析エラー、又はファイル生成に失敗しているおそれがあります。 標準出力（Windows の場合はイベントビューア）を参照してエラー内容を確認し、対処した後に再度初期化を実行してください。

注

反映側 Datareplicator を初めて初期化する場合は、hdsstart コマンドに-f オプションを付けて実行することを推奨します。また、ほかのオプションも必要に応じて付けてください。

9. 抽出側 Datareplicator 及び反映側 Datareplicator を起動します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を起動します。	hdsstart	errfile1 又は errfile2 に KFRB00100-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を起動します。	hdestart*	msterrfile1 又は msterrfile2 に KFRB00502-I メッセージが出力されていることを確認してください。

注※

hdestart コマンドには、必要に応じてオプションを付けてください。

《反映トランザクション同期機能を使用する場合》

反映トランザクション同期機能を使用する場合は、手順 1~9 を実施した後、反映側 Datareplicator の同期反映グループの全プロセスを稼働状態にさせるために、同期イベントを実行してください。

項番	作業内容	実行コマンド	確認項目
1	同期イベントを実行します。	hdeevent* -n 同期イベントコード	<ul style="list-style-type: none">標準エラー出力にメッセージが出力されていないことを確認してください。反映側システムのエラー情報ファイルに KFRB03009-I メッセージが出力され、「reason」が「SYNC EVENT」であることを確認してください。

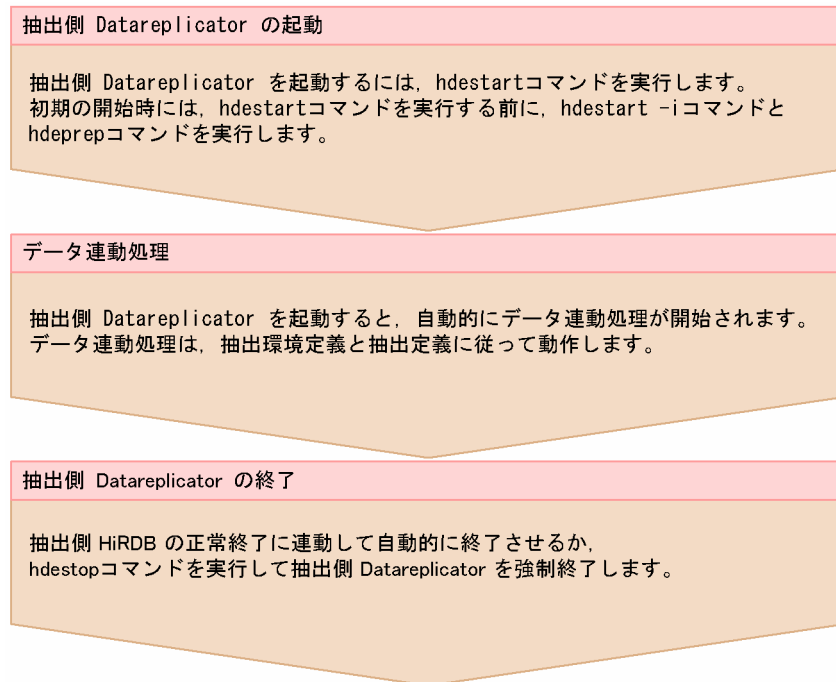
注※

hdeevent コマンドには、必要に応じてオプションを付けてください。

6.3 抽出側 Datareplicator の起動と終了

抽出側 Datareplicator の起動と終了の手順を次の図に示します。

図 6-4 抽出側 Datareplicator の起動と終了の手順



6.3.1 抽出側 Datareplicator の起動

抽出側 Datareplicator の起動方法と開始モードについて説明します。

抽出処理の開始方式の詳細については、「4.6.5 抽出処理の開始方式の設計」を参照してください。

(1) 抽出側 Datareplicator の起動方法

抽出側システムで hdestart コマンドを実行すると、抽出システム定義、抽出環境定義、及び送信環境定義に従って抽出側 Datareplicator を起動します。Windows 版 Datareplicator の場合は、コマンドプロンプトでコマンドを実行してください。

抽出側 Datareplicator の環境を初期化するときには、hdestart -i コマンドを実行します。特定の送信先に対する送信環境だけを初期化したい場合は、hdestart -i -S コマンドを使って、該当する送信先の環境だけを初期化します。

また、HiRDB の影響分散スタンバイレス型系切り替え機能を使用する場合、hdestart_n コマンドによる部分起動（抽出マスタプロセスと抽出ノードマスタプロセスを別々に起動）を行うこともできます。

詳細については、「7. コマンドの文法」の hdestart コマンドを参照してください。

なお、Windows 版 Datareplicator の場合、次の二つの方法でも起動できます。

- 抽出サービスを手動で開始する方法
- Windows の起動時に抽出サービスを自動で開始する方法

抽出サービスを手動で開始する方法

1. [コントロールパネル] にある [サービス] アイコンを開きます。
2. ノードマスタプロセス起動サービスが開始していることを確認します。
3. [サービス] から「HiRDB Datareplicator (Source Site)」を選びます。
4. [スタートアップパラメータ] に開始時のオプション (hdestart コマンドのオプション) を指定します。
5. [開始] ボタンをクリックします。

開始時のオプションに「-i」を指定して初期開始する場合の注意

- [サービス] から初期開始した場合、ファイルを初期化するかどうかの確認メッセージは表示されません。このため、初期開始する前に、抽出側 HiRDB が HiRDB Datareplicator 連携を実行していないことを確認してください。
- ファイルの初期化をした後に抽出側 Datareplicator が終了するので、[サービス] ダイアログボックスでエラーメッセージが表示されることがあります。このときは、イベントビューア又はエラーログファイルで、初期化完了のメッセージが出力されているかどうかを確認してください。初期化完了のメッセージが出力されていれば、初期開始が正常に実行されています。

Windows の起動時に抽出サービスを自動で開始する方法

Windows を起動したときに、抽出サービスを自動で開始する設定にする手順を次に示します。この場合、開始時のオプション (hdestart コマンドのオプション) は指定できません。

1. [コントロールパネル] にある [サービス] アイコンを開きます。
2. ノードマスタプロセス起動サービスが自動で開始する設定になっていることを確認します。
3. [サービス] から「HiRDB Datareplicator (Source Site)」を選びます。
4. [スタートアップ] ボタンをクリックします。
5. [サービス] ダイアログボックスの [スタートアップの種類] で [自動] を選びます。
6. [OK] ボタンをクリックします。
7. [閉じる] ボタンをクリックします。

次に Windows を起動したときに、抽出サービスが自動で開始します。

(2) 抽出側 Datareplicator の開始モード

抽出側 Datareplicator の開始モードには、次の 4 種類があります。

開始モードの詳細については、「7. コマンドの文法」の [hdestart](#) コマンドの文法を参照してください。

- **初期開始**

次に示すファイルを初期化します。抽出側 Datareplicator は起動しません。

- 抽出情報キューファイル
- 抽出マスタステータスファイル
- 抽出サーバステータスファイル
- 抽出マスタエラー情報ファイル
- 抽出ノードマスタエラー情報ファイル
- 稼働トレースファイル
- データ連動用連絡ファイル

- **部分初期開始**

指定された送信先に対する抽出システム定義、及び送信環境定義に従って、指定された送信先への送信環境だけを初期化します（部分初期化）。指定された送信先以外に対する送信環境は変更しません。また、抽出側 Datareplicator は起動しません。

- **正常開始**

前回の稼働時の動作内容を引き継がないで、抽出システム定義、抽出環境定義、及び送信環境定義に従って抽出側 Datareplicator が開始します。

- **再開**

前回、抽出処理又は送信処理がエラーで終了している場合には、抽出側 DB と反映側 DB の整合性を保証するため、前回の稼働時と同じ動作内容で抽出側 Datareplicator が開始します。

6.3.2 抽出側 Datareplicator の終了

抽出側 Datareplicator の終了方法と終了モードについて説明します。

抽出処理の停止方式の詳細については、「[4.6.6 抽出処理の停止方式の設計](#)」を参照してください。

(1) 抽出側 Datareplicator の終了方法

抽出システム定義の `syncterm` オペランドに `true` を指定すると、抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator が終了します（正常終了）。

また、抽出側システムで `hdestop` コマンドを実行すると、抽出側 Datareplicator を終了します（強制終了）。Windows 版 Datareplicator の場合は、コマンドプロンプトでコマンドを実行してください。

また、HiRDB の影響分散スタンバイレス型系切り替え機能を使用する場合、`hdestop_n` コマンドによる部分終了（抽出マスタプロセスと抽出ノードマスタプロセスを別々に終了）を行うこともできます。

なお、抽出側 Datareplicator を終了する場合には、必ず `hdestop` コマンドを使ってください。OS のコマンドでプロセスを消して、抽出側 Datareplicator を終了した場合には、動作は保証できません。

Windows 版 Datareplicator の場合は、次の方法でも終了（強制終了）できます。

抽出サービスを手動で終了する方法

1. [コントロールパネル] にある [サービス] アイコンを開きます。
2. [サービス] から「HiRDB Datareplicator (Source Site)」を選びます。
3. [停止] ボタンをクリックします。

抽出側 Datareplicator の起動中に Windows を停止させると、抽出側 Datareplicator は自動的に強制終了します。

(2) 抽出側 Datareplicator の終了モード

抽出側 Datareplicator の終了モードには、次の 2 種類があります。

終了モードの詳細については、「7. コマンドの文法」の `hdestop` コマンドの文法を参照してください。

- **正常終了**

抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator が終了します。正常終了するには、抽出システム定義の `syncterm` オペランドに `true` を指定している必要があります。

- **強制終了**

`hdestop` コマンドを実行した時点実行中の抽出処理と送信処理が完了すると、抽出側 Datareplicator が終了します。

強制終了すると反映側 DB と抽出側 DB とで不整合が生じることがあります。強制終了した場合には、次回の起動時に開始モードを再開始とすると、反映側 DB との整合性が保証されます。

6.4 抽出側 Datareplicator の運用

抽出側 Datareplicator の運用方法について説明します。

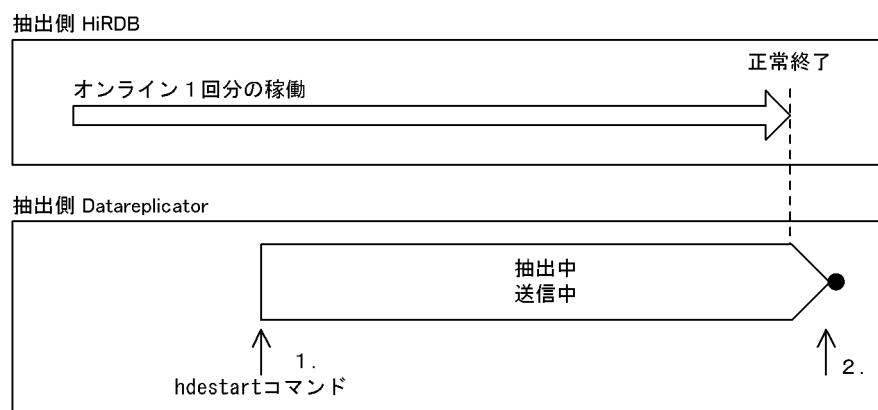
6.4.1 抽出処理の運用

抽出処理の運用方法について説明します。

(1) 抽出側 Datareplicator を同時開始する

抽出側 Datareplicator の起動と同時に、抽出と送信を開始します。抽出側 Datareplicator を同時開始する場合の運用例を図 6-5 に、同時開始の場合の運用手順と抽出側 Datareplicator の動作を表 6-1 に示します。

図 6-5 抽出側 Datareplicator の同時開始の運用例



(凡例)

➡ : 抽出と送信の両方が動作中

● : 抽出側 Datareplicator の終了

注 図中の 1, 2 は、次の表の操作に対応しています。

表 6-1 運用手順と抽出側 Datareplicator の動作 (同時開始)

順序	運用手順	抽出側 Datareplicator の動作
1	hdestart コマンドを実行します (初期起動時は hdestart コマンドを実行する前に、hdestart - i コマンドと hdeprep コマンドを実行します)。	抽出側 Datareplicator が起動して、抽出側 HiRDB のシステムログファイルからの更新情報の抽出、及び反映側システムへの更新情報の送信を開始します。
2	—	syncterm オペランドに true を指定している場合： 抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator が終了します。 syncterm オペランドに false を指定している場合： 抽出側 HiRDB が正常終了しても、抽出側 Datareplicator は正常終了しません。false を指定している場合に抽出側

順序	運用手順	抽出側 Datareplicator の動作
		Datareplicator を終了させたいときは、pdls コマンドや hdestate コマンドを実行して状態を確認した後に、hdestop コマンドを実行してください。

(凡例)

— : 操作は必要ありません。

(2) 抽出側 Datareplicator を送信遅延開始する

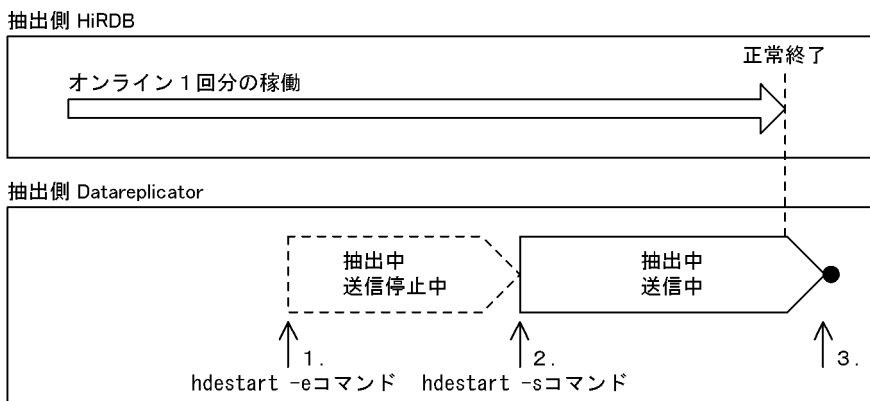
抽出側 Datareplicator の起動時には抽出だけを開始し、hdestart -s コマンドによって送信の開始が要求された時点で、送信を開始します。

送信遅延開始するには、次のように操作します。

- 抽出側システムで hdestart -e コマンドを実行して、抽出側 Datareplicator を起動します。
抽出処理だけが開始されます。
- 抽出側システムで hdestart -s コマンドを実行します。
送信処理が開始されます。hdestart -s コマンドに送信先識別子を指定して、特定の送信先への送信処理だけを開始することもできます。

抽出側 Datareplicator を送信遅延開始する場合の運用例を [図 6-6](#) に、送信遅延開始の場合の運用手順と抽出側 Datareplicator の動作を [表 6-2](#) に示します。

図 6-6 抽出側 Datareplicator の送信遅延開始の運用例



(凡例)

- : 抽出と送信の両方が動作中
- : 抽出又は送信のどちらかが停止中
- : 抽出側 Datareplicator の終了

注 図中の 1~3 は、次の表の操作に対応しています。

表 6-2 運用手順と抽出側 Datareplicator の動作（送信遅延開始）

順序	運用手順	抽出側 Datareplicator の動作
1	hdestart -e コマンドを実行します。	抽出側 Datareplicator が起動して、抽出側 HiRDB のシステムログファイルからの更新情報の抽出だけを開始します。
2	抽出プロセスが、抽出側 HiRDB の正常終了を検知する前に、hdestart -s コマンドを実行します。 抽出プロセスが抽出側 HiRDB の正常終了を検知するまでに送信プロセスを起動していない場合、抽出情報キューファイル内の更新情報は反映側システムに送信されません。	反映側システムへの更新情報の送信を開始します。
3	—	syncterm オペランドに true を指定している場合： 抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator が終了します。 syncterm オペランドに false を指定している場合： 抽出側 HiRDB が正常終了しても、抽出側 Datareplicator は正常終了しません。false を指定している場合に抽出側 Datareplicator を終了させたいときは、pdls コマンドや hdestate コマンドを実行して状態を確認した後に、hdestop コマンドを実行してください。

(凡例)

—：操作は必要ありません。

(3) 抽出側 Datareplicator を抽出遅延開始する

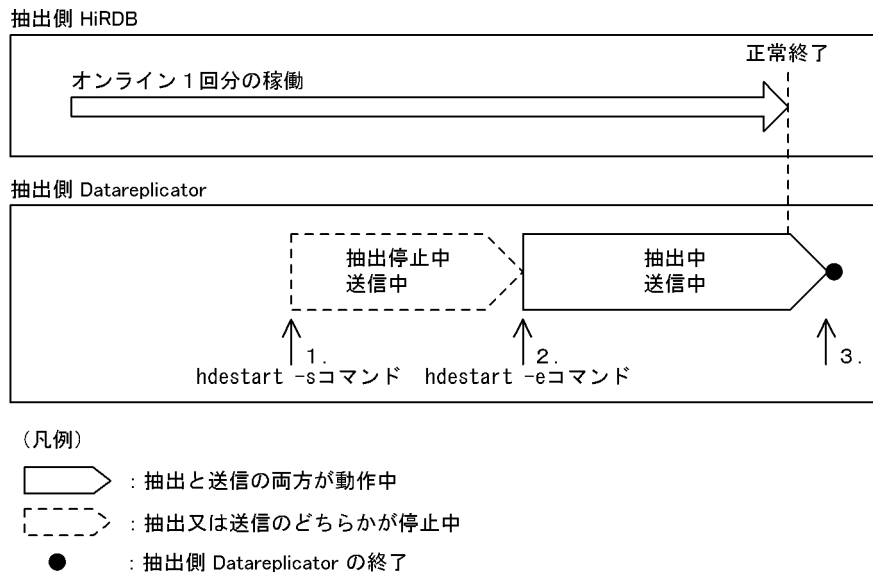
抽出側 Datareplicator の起動時には送信だけを開始し、hdestart -e コマンドによって抽出の開始が要求された時点で、抽出を開始します。

抽出遅延開始するには、次のように操作します。

- 抽出側システムで hdestart -s コマンドを実行します。
送信処理だけが開始されます。hdestart -s コマンドに送信先識別子を指定して、特定の送信先への送信処理だけを開始することもできます。
- 抽出側システムで hdestart -e コマンドを実行します。
抽出処理が開始されます。

抽出側 Datareplicator を抽出遅延開始する場合の運用例を図 6-7 に、抽出遅延開始の場合の運用手順と抽出側 Datareplicator の動作を表 6-3 に示します。

図 6-7 抽出側 Datareplicator の抽出遅延開始の運用例



注 図中の 1~3 は、次の表の操作に対応しています。

表 6-3 運用手順と抽出側 Datareplicator の動作 (抽出遅延開始)

順序	運用手順	抽出側 Datareplicator の動作
1	hdestart -s コマンドを実行します。	抽出側 Datareplicator が起動して、反映側システムへの更新情報の送信だけを開始します。
2	hdestart -e コマンドを実行します。	抽出側 HiRDB のシステムログファイルからの更新情報の抽出を開始します。
3	—	<p>syncterm オペランドに true を指定している場合：</p> <p>抽出側 HiRDB の正常終了に連動して、自動的に抽出側 Datareplicator が終了します。</p> <p>syncterm オペランドに false を指定している場合：</p> <p>抽出側 HiRDB が正常終了しても、抽出側 Datareplicator は正常終了しません。false を指定している場合に抽出側 Datareplicator を終了させたいときは、pdls コマンドや hdestate コマンドを実行して状態を確認した後、hdestop コマンドを実行してください。</p>

(凡例)

— : 操作は必要ありません。

(4) エラー停止中の一部の送信処理だけを再開始する

エラー停止中の一部の送信処理だけを再開始するには、次のように操作します。

1. 抽出側システムで hdestate コマンドを実行し、エラー停止している送信処理を調べます。
2. 抽出側システムで hdestate -s コマンドにエラーになっている送信処理の送信先識別子を指定して、エラーの原因を調査します。

3. エラーの原因を取り除いた後、抽出側システムで、停止している送信処理の送信先識別子を指定した `hdestart -s` コマンドを実行します。

前回のモードを引き継いで、指定した送信先識別子の送信処理が開始されます。

(5) 抽出側 Datareplicator の送信先を追加する

送信先を追加するには、いったんデータ連動を停止して、抽出側又は反映側 Datareplicator 定義を変更します。定義を変更した送信先は初期化が必要ですが、部分初期化を実行することもできます。

送信先を追加する手順については、「[6.5.6 抽出側 HiRDB の運用手順](#)」を参照してください。

6.4.2 抽出側 Datareplicator で使うファイルの運用

抽出側 Datareplicator で使うファイルの運用方法について説明します。

ファイルの内容については、「[3.2.2 抽出処理で使うファイルとプロセス](#)」を参照してください。

ファイルの準備については、「[4.6.2 抽出側 Datareplicator で使うファイルの準備](#)」を参照してください。

注意事項

Datareplicator の稼働中／停止中に関係なく、ファイルを入れ替えないでください。Datareplicator が誤動作するおそれがあります。

(1) 抽出システム定義ファイルの運用方法

抽出システム定義ファイルの運用方法については、「[5.2.2 定義内容の変更](#)」を参照してください。

(2) 抽出環境定義ファイルの運用方法

抽出環境定義ファイルの運用方法については、「[5.3.2 定義内容の変更](#)」を参照してください。

(3) 送信環境定義ファイルの運用方法

送信環境定義ファイルの運用方法については、「[5.4.2 定義内容の変更](#)」を参照してください。

(4) 抽出定義ファイルの運用方法

抽出定義ファイルの運用方法については、「[5.5.2 定義内容の変更](#)」を参照してください。

(5) 抽出定義プリプロセスファイルの運用方法

抽出定義プリプロセスファイルの運用方法について説明します。

(a) 抽出定義ファイルの再変換

抽出側 HiRDB で抽出対象の表の定義変更をした場合、又は抽出側 Datareplicator の抽出定義を変更した場合には、抽出定義ファイルを内部形式に再変換する必要があります。抽出定義ファイルを内部形式に再変換する場合には、次のように操作してください。

1. 抽出側 Datareplicator を正常終了します。
2. 抽出対象の表の定義変更、又は抽出定義を変更します。
3. 抽出側システムで hdeprep コマンドを実行します。
4. 抽出側 Datareplicator を正常開始します。

(6) 抽出情報キューファイルの運用方法

抽出情報キューファイルの運用方法について説明します。抽出情報キューファイルの満杯時の対処については、「[9.1.2 エラーの対処方法](#)」を参照してください。

(a) ファイル名、サイズ、又は個数の変更

抽出情報キューファイルの名称、サイズ、又は個数を変更する場合は、次のように操作します。

UNIX の通常ファイル、又は Windows ファイルの場合の操作

1. 抽出側システムを終了します。
2. 抽出側 Datareplicator を正常終了します。
3. テキストエディタで、抽出環境定義の対応するオペランドの設定を変更します。
4. 抽出側 Datareplicator を初期開始します。
5. 抽出側 Datareplicator を正常開始します。
6. 抽出側システムを起動します。

UNIX のキャラクタ型スペシャルファイルの場合の操作

1. 抽出側システムを終了します。
2. 抽出側 Datareplicator を正常終了します。
3. OS のコマンドで、キャラクタ型スペシャルファイル形式の抽出情報キューファイルを再作成します。

注意事項

HiRDB のサーバ追加、又は削除などによって、ステータスファイルのサイズ変更や Disk 配置の変更が発生した場合にだけ実施が必要

4. テキストエディタで、抽出環境定義の対応するオペランドの設定を、作成した抽出情報キューファイルに合わせて変更します。

注意事項

HiRDB のサーバ追加, 又は削除などによって, ステータスファイルのサイズ変更や Disk 配置の変更が発生した場合にだけ実施が必要

- 抽出側 Datareplicator を初期開始します。
- 抽出側 Datareplicator を正常開始します。
- 抽出側システムを起動します。

(b) 抽出情報キューファイルの構成を変更するコマンド (hdemodq コマンド)

hdemodq コマンドを使うと, 抽出側 Datareplicator を初期開始しなくても, 抽出情報キューファイルの構成を変更できます。hdemodq コマンドでできる操作を次に示します。

hdemodq コマンドの操作	説明
抽出情報キューファイルの情報の表示	オフライン時の抽出情報キューファイルの情報を表示できます。
抽出情報キューファイルの追加登録	満杯状態を解消するために, コマンドで抽出情報キューファイルを追加できます。
抽出情報キューファイルの登録解除	不要になった抽出情報キューファイルを削除できます。

hdemodq コマンドを実行する前提条件

hdemodq コマンドを実行できる条件を次に示します。

- 抽出側 Datareplicator が停止中。
- 環境変数「HDEPATH」を設定済み。
- 構成を変更する抽出情報キューファイルが, コマンドを実行するノードにある。
- このコマンドを実行するユーザは, UNIX のときは抽出ノードマスタープロセス実行ユーザ (inetd.conf で指定), Windows のときは Administrators 権限があるユーザである。ただし, hdemodq コマンドの -l オプション指定 (ファイル情報の表示) に限り, 一般ユーザも実行可能。

同じ抽出側システムに対して hdemodq コマンドを多重実行できるのは, 情報を表示するときだけです。追加登録と登録解除は多重実行できません (どれか一つを除いて, KFRB09302-E メッセージを出力してエラーになります)。

抽出情報キューファイル情報の表示

抽出情報キューファイルの情報を, 標準出力に表示できます。表示形式については, 「7. コマンドの文法」の hdemodq コマンドを参照してください。

抽出情報キューファイルの追加登録

抽出情報キューファイルのサイズ分だけ初期化して, Datareplicator に登録できます。追加登録は, 抽出情報キューファイルへの書き込みでディスク容量オーバー (Disk Full) になったときは適用できません。抽出情報キューファイルは 16 個まで追加できます。

追加できる抽出情報キューファイルの種類は、UNIX の通常ファイル又はキャラクタ型スペシャルファイル（Windows の場合は、Windows ファイル）です。Datareplicator ファイルシステム領域へは、抽出情報キューファイルを追加できません。

追加する抽出情報キューファイルのサイズは、割り当て済みの抽出情報キューファイルと同じサイズでなければなりません。

- **抽出情報キューファイルの登録解除**

Datareplicator に登録してある抽出情報キューファイルを解除します。解除できるのは、抽出情報キューファイルの情報表示で未送信抽出情報の有無が「e」と表示されているファイルだけです。なお、hdemodq コマンドでは登録の解除だけを実行します。抽出情報キューファイル自体は削除しません。抽出情報キューファイルは、最小で 2 個になるまで解除できます。Datareplicator に登録されている抽出情報キューファイルが 2 個のときは、登録を解除できません。

登録を解除できる抽出情報キューファイルの種類は、UNIX の通常ファイル又はキャラクタ型スペシャルファイル（Windows の場合は、Windows ファイル）、又は Datareplicator ファイルシステム領域上の抽出情報キューファイルです。

注意

hdemodq コマンドで抽出情報キューファイルの構成を変更するときは、次の項目に注意してください。

- **抽出サーバステータスファイルのバックアップ**

hdemodq コマンドでは、抽出情報キューファイルの状態を変更するため、抽出サーバステータスファイルを更新します。コマンド実行時の障害に備えて、コマンド実行前に抽出サーバステータスファイルのバックアップを取得しておいてください。抽出サーバステータスファイルを Datareplicator ファイルシステム領域に格納しているときは、その Datareplicator ファイルシステム領域をすべてバックアップしてください。バックアップ方法については、「[6.4.2\(7\)\(b\) 抽出サーバステータスファイルのバックアップ](#)」を参照してください。

- **構成変更後の初期化**

hdemodq コマンドを実行して抽出情報キューファイルの構成を変更した後に hdestart -i コマンドを実行して抽出側 Datareplicator の環境を初期化すると、抽出環境定義に従って環境が再構築されます。そのため、hdemodq コマンドで変更した構成は無効となるので注意してください。

(7) 抽出のステータスファイルの運用方法

抽出側 Datareplicator のステータスファイル（抽出マスタステータスファイル、抽出サーバステータスファイル）の運用方法について説明します。

(a) 抽出のステータスファイルの初期化

抽出のステータスファイル（抽出マスタステータスファイル、抽出サーバステータスファイル）を初期化するときは、次のように操作します。

UNIX の通常ファイル、又は Windows ファイルの場合の操作

1. 抽出側システムを終了します。

2. 抽出側 Datareplicator を正常終了します。
3. 抽出側 Datareplicator を初期開始します。
4. 抽出側 Datareplicator を正常開始します。
5. 抽出側システムを起動します。

UNIX のキャラクタ型スペシャルファイルの場合の操作

1. 抽出側システムを終了します。
2. 抽出側 Datareplicator を正常終了します。
3. OS のコマンドで、前回のステータスファイルを削除します。
HiRDB のサーバ追加、又は削除などによって、ステータスファイルのサイズ変更や Disk 配置の変更が発生した場合にだけ実施が必要です。
4. OS のコマンドで、キャラクタ型スペシャルファイル形式のステータスファイルを再作成します。
HiRDB のサーバ追加、又は削除などによって、ステータスファイルのサイズ変更や Disk 配置の変更が発生した場合にだけ実施が必要です。
5. 抽出側 Datareplicator を初期開始します。
6. 抽出側 Datareplicator を正常開始します。
7. 抽出側システムを起動します。

(b) 抽出サーバステータスファイルのバックアップ

次の場合、抽出サーバステータスファイルをバックアップしてください。

- hdemodq コマンドで抽出情報キューファイルの構成を変更する場合
- 抽出情報キューファイルが満杯の場合
- 抽出情報キューファイル回復機能を使用する場合

抽出サーバステータスファイルのバックアップを取得するためのコマンドを次の表に示します。

表 6-4 抽出サーバステータスファイルのバックアップを取得するためのコマンド

抽出サーバステータスファイルの種別	使用するコマンド	コマンド実行例
UNIX の通常ファイル	cp	cp \$HDEPATH/sts_sds01 sts_sds01_backup
UNIX のキャラクタ型スペシャルファイル	dd	dd if="\$HDEPATH/sts_sds01" of=sts_sds01_backup bs=1024 count=sts_sds01 のキロバイト数
Windows ファイル	copy	copy "%HDEPATH%\sts_sds01" sts_sds01_backup

注 1

抽出サーバステータスファイル名が sts_sds01 の場合の例です。各コマンドの詳細については、該当する OS のマニュアルを参照してください。

注 2

抽出サーバステータスファイルをバックアップから回復する場合は、抽出サーバステータスファイル名とバックアップファイル名を入れ替えて、コマンドを実行してください。

(8) 抽出のエラー情報ファイルの運用方法

抽出のエラー情報ファイル（抽出マスタエラー情報ファイル、抽出ノードマスタエラー情報ファイル）の運用方法について説明します。

(a) 最大サイズの変更

抽出のエラー情報ファイルの最大サイズを変更するときは、次のように操作します。

1. 抽出側システムを終了します。
2. 抽出側 Datareplicator を正常終了します。
3. テキストエディタで、抽出システム定義の対応するオペランドの設定を変更します。
4. 抽出側 Datareplicator を正常開始します。
5. 抽出側システムを起動します。

(b) 抽出のエラー情報ファイルの保存

蓄積の対象になるエラー情報ファイルが満杯になるとスワップされます。スワップ先のエラー情報ファイルはスワップ時に再作成されます。再作成される前にエラー情報ファイルの内容を保存したいときは、次のように操作してください。

1. OS のコマンドで、エラー情報ファイルの更新日時を調べます。
2. 更新日時の古いエラー情報ファイルを、OS のコマンドで任意の名称のファイルにコピーします。

Datareplicator の稼働中にエラー情報ファイルがスワップ又はクローズされると、メッセージ (KFRB00051-I, KFRB00052-I) が出力されます。このメッセージを契機に、必要に応じてバックアップを取得してください。

(c) エラー情報の標準出力への出力

蓄積の対象になるエラー情報ファイルが満杯になるとスワップされます。スワップ先のエラー情報ファイルはスワップ時に再作成されます。再作成される前にエラー情報ファイルの内容を標準出力に出力したいときは、次のように操作してください。

1. OS のコマンドで、抽出マスタエラー情報ファイルの更新日時を調べます。
2. 更新日時の古い抽出マスタエラー情報ファイルを、OS のコマンドで標準出力に出力します。

(d) エラー情報ファイルの出力例

抽出マスタエラー情報ファイルの出力例を図 6-8 に、抽出ノードマスタエラー情報ファイルの出力例を図 6-9 に示します。

図 6-8 抽出マスタエラー情報ファイルの出力例

```
*****
1. Fri Mar 29 21:15:39 2002 process: hdemaster(1324) function:hde_mst_main
2. errorcode: KFRB00701-E .. 00 Error occurred on node-master process,
   hostname = host01, server name = bes01, senderid = snd01, message number = 00703
3. information:
   *****
   Fri Mar 29 21:16:21 2002 process: hdemaster(1301) function:hde_com_main
   errorcode: KFRB00601-E .. 00 Create socket failed, error, errno = 12.
   information:
```

1. Fri...2002, process, function
Fri...2002 : エラーが発生した曜日, 月日, 時分秒, 及び西暦です。
process : エラーが発生した内部プロセス名とプロセス番号です。
function : エラーが発生した内部関数名です。
2. errorcode ~ message number=00703
エラーメッセージです。メッセージの内容については、「10. メッセージの一覧」を参照してください。
3. information
発生したエラーの詳細情報です。

図 6-9 抽出ノードマスタエラー情報ファイルの出力例

```
*****
1. Fri Mar 29 21:15:39 2002 process: hdcapture(1324) function: hds_com_read
2. serverid: SERVER01 senderid:SENDER01
3. errorcode: EFRB05011-E aa 00 HiRDB overwrites unextracted system-log.
4. information:
   *****
   Fri Mar 29 21:16:21 2002 process: hdesender(1301) function: hds_com_main
   serverid: SERVER01 senderid:SENDER02
   errorcode: EFRB02013-E aa 00 Socket option set error, errno = 12.
   information:
```

1. Fri...2002, process, function
Fri...2002 : エラーが発生した曜日, 月日, 時分秒, 及び西暦です。
process : エラーが発生した内部プロセス名とプロセス番号です。
function : エラーが発生した内部関数名です。
2. serverid, senderid
serverid : エラーが発生したサーバ名です。
senderid : エラーが発生した送信先識別子です。
3. errorcode ~ system-log
エラーメッセージです。メッセージの内容については、「10. メッセージの一覧」を参照してください。
4. information
発生したエラーの詳細情報です。

(e) ファイル以外への出力

エラー情報ファイルの内容は、エラー情報ファイルの障害回避や自動化運転の情報取得のために、syslog ファイルに出力されます。

syslog ファイルに出力するときには、抽出システム定義の `syslogout` オペランドを `true` にする必要があります。

抽出マスタエラー情報ファイルの内容の出力先を次の表に示します。

表 6-5 抽出マスタエラー情報ファイルの内容の出力先

抽出マスタエラー情報ファイルの状態	出力先	
	syslog ファイル	抽出マスタエラー情報ファイル
正常 (出力可)	△	○
障害あり (出力不可)	△	×

(凡例)

- ：出力します。
- △：抽出システム定義の `syslogout` オペランドが `true` の場合だけ出力します。
- ×：出力しません。

(9) 稼働トレースファイルの運用方法

稼働トレースファイルは、`hdstrcredit` コマンドで編集して参照できます。`hdstrcredit` コマンドの使い方については、「7. コマンドの文法」の `hdstrcredit` コマンドを参照してください。

(10) データ連動用連絡ファイルの運用方法

データ連動用連絡ファイルの運用方法について説明します。

(a) データ連動用連絡ファイルの初期化

データ連動用連絡ファイルを初期化するには、次のように操作します。

UNIX の通常ファイル、又は Windows ファイルの場合の操作

1. 抽出側システムを終了します。
2. 抽出側 Datareplicator を正常終了します。
3. 抽出側 Datareplicator を初期開始します。
4. 抽出側 Datareplicator を正常開始します。
5. 抽出側システムを起動します。

UNIX のキャラクタ型スペシャルファイルの場合の操作

1. 抽出側システムを終了します。
2. 抽出側 Datareplicator を正常終了します。
3. OS のコマンドで、前回のデータ連動用連絡ファイルを削除します。

4. OS のコマンドで、キャラクタ型スペシャルファイル形式のデータ連動用連絡ファイルを再作成します。
5. 抽出側 Datareplicator を初期開始します。
6. 抽出側 Datareplicator を正常開始します。
7. 抽出側システムを起動します。

(11) コマンドログファイルの運用方法

コマンドログファイルの運用方法について説明します。

コマンドログファイルは、Datareplicator のコマンドを実行した履歴を格納するファイルです。Datareplicator の起動時に自動的に作成されます。コマンドを実行した履歴を確認したいときに、任意に参照してください。コマンドログに出力される情報については、「7. コマンドの文法」の「コマンドの概要」を参照してください。

コマンドログファイルの内容の例を次の図に示します。

図 6-10 コマンドログファイルの内容の例

```
Fri Aug 18 09:36:52 2006 pid=596 SYSTEM hdestart : END : status=0, arg= -i.  
Fri Aug 18 09:37:15 2006 pid=912 administrator hdeprep : END : status=0, arg= -f extfile.  
Fri Aug 18 09:40:18 2006 pid=952 SYSTEM hdestart : END : status=0, arg=.
```

6.4.3 抽出側 Datareplicator 運用時の注意

抽出側 Datareplicator を運用するときの注意を次に示します。

(1) 更新情報についての注意

- 複数の抽出側システムから同一の反映側システムの表を更新する場合、更新順序は保証されません。更新順序を制御する必要がある場合には、運用で実行してください。例えば、二つの抽出側システム A、B から、反映側システムの表 (T1) とをデータ連動する場合、抽出側システム A のマッピングキーの値「100」の行に対する更新の次に、抽出側システム B のキー値 100 の行に対する更新が実行されても、反映側システムのキー値 100 の行の値は抽出側システム B と同一になっていることは保証されません。
- 抽出側システムで抽出定義を変更する場合は、抽出側システムの更新情報がすべて反映側 Datareplicator に送信されたことを確認してから、実行してください。反映側 Datareplicator に送信する更新情報が残っているまま抽出定義を変更した場合、反映処理中にエラーになることがあります。この場合反映側 Datareplicator では、エラーを回復できないことがあり、HiRDB Dataextractor を使った反映側 DB の再作成が必要になることがあります。
- 抽出側 Datareplicator は、送信間隔内でトランザクションが完結している更新情報を対象に、トランザクションのコミット順に更新情報を反映側システムに送信し、反映側システムで反映されます。この

ため、送信間隔内で完結していないトランザクションについては、後に発生したトランザクションよりも後に、反映側システムに送信されることがあります。このときは、抽出側 DB に対する更新処理の順序と反映側 DB に対する更新処理の順序が異なることがあります。

- 抽出側 HiRDB がパラレルサーバの場合、抽出側 Datareplicator はバックエンドサーバ単位に並行して、更新情報の抽出、送信を実行します。このためバックエンドサーバ間での更新順序は保証されません。ただし、全バックエンドサーバ上のシステムログファイル内の、すべての更新情報が抽出、送信され、さらに反映側 DB での反映が完了した時点で、抽出側 DB と反映側 DB の整合性が保証されます。

(2) 初期開始、部分初期開始についての注意

抽出側システムを初期化した状態で起動したときは、反映側 Datareplicator も `hdsstart -i` 又は `hdsstart -i -D` で起動してください。部分初期開始時は、部分初期開始の対象にした送信先に対応する反映側 Datareplicator を `hdsstart -i` 又は `hdsstart -i -D` で起動してください。

抽出側システムだけを初期状態で起動した場合には、KFRB02003-E（詳細コード 5）でエラーになります。

ただし、反映環境定義で `extract_init` に `nocheck` を指定している場合は、反映側 Datareplicator を初期開始又は部分初期開始する必要はありません。

(3) イベントコードについての注意

抽出側システムから送信されたイベントのコードが反映環境定義で定義されていない場合、反映処理の動作には何も影響ありません。ただし、反映側 Datareplicator は、イベントの発生としては認識します。

(4) 環境変数 HDE_BIN_COL_MAXLEN についての注意

抽出側 Datareplicator の環境変数 `HDE_BIN_COL_MAXLEN` を指定（単位：キロバイト）すると、定義長は 2GB 以上でも実際に扱うデータは小さい BLOB 型の列を、表を再定義することなくデータ連動できるようになります。ただし、環境変数 `HDE_BIN_COL_MAXLEN` に指定した定義長を超える BLOB 型の列が検出された場合、抽出側 Datareplicator は次の処理を実施します。

- 該当する BLOB 型の列のデータを NULL 値に置換して、反映側 Datareplicator に反映します。
- NULL 値に置換した更新情報から、マッピングキー列の情報をファイルに出力します。

出力されるファイルについて説明します。

ファイル名及び出力先

「warn_keyinfo.BES 名.送信先名」という名称で、抽出側 Datareplicator 運用ディレクトリ（`HDEPATH` 環境変数に指定したディレクトリ）に作成されます。抽出側 DB が HiRDB/パラレルサーバの場合は、環境変数 `HDE_BIN_COL_MAXLEN` に指定した定義長を超える BLOB 型の列が検出されたバックエンドサーバが存在するマシンの、抽出側 Datareplicator 運用ディレクトリに作成されます。

出力内容

ファイルに出力される内容の例を次に示します。

```
Skip info [SQL : insert]
Table name : USR1.TBL1
Column name : C1
  [DATA : length=4]
  0001                                     *...      *
Column name : C2
  [DATA : length=24]
  30303030303030303030303030303030      *0000000000000000*
  3030303030303031                       *00000001      *
```

注意事項

- 抽出側 Datareplicator の開始前に上記のファイルを手動で削除してください。抽出側 Datareplicator の運用終了時にファイルが作成されていた場合は、データ連動中に、環境変数 HDE_BIN_COL_MAXLEN に指定した定義長を超える BLOB 型の列が検出されたことを示します。
- マッピングキー列の情報は送信処理のタイミングで出力されるため、送信処理中にエラーが発生した場合、送信処理の再開後に、一度出力したマッピングキー列の情報を再度出力する場合があります。

6.5 抽出側 HiRDB の運用

HiRDB のデータベースからデータを抽出するときの、抽出側 HiRDB の運用について説明します。

6.5.1 HiRDB Datareplicator 連携の開始・終了・中止

(1) HiRDB Datareplicator 連携の開始と終了

HiRDB Datareplicator 連携は、次のタイミングで開始されて、抽出側 HiRDB の終了時に終了します。

- 抽出側 HiRDB のシステム共通定義での `pd_rpl_init_start = Y` 指定

正常開始時に引き続き HiRDB Datareplicator 連携を続行するには、抽出側 HiRDB のシステム共通定義の `pd_rpl_init_start` オペランドに `Y` を指定してください。

HiRDB Datareplicator 連携時に `pd_rpl_init_start = Y` と指定してあれば、抽出側 HiRDB の開始/再開に関係なく、前回の抽出側 HiRDB でのシステムログへのデータ連動情報の出力状態と抽出側 Datareplicator でのシステムログの抽出状態を引き継いで実行します。

ただし、次の場合は前回までの状態を引き継がないで、抽出状態を初期化して HiRDB Datareplicator 連携を開始します。このため、抽出側 DB と反映側 DB との間に不整合が発生するので、HiRDB Datareplicator 連携を再開する前に反映側 DB の再作成が必要です。

- 強制正常開始 (`pdstart dbdestroy`)
 - データベースの初期化開始 (`pdstart -i`)
- `pdrplstart` コマンドの実行

`pdrplstart` コマンドで HiRDB Datareplicator 連携を開始すると、再開時には前回までの状態を引き継ぎますが、正常開始時には引き継ぎません。HiRDB を正常開始した後、再度 `pdrplstart` コマンドを実行する必要があります。そのため、抽出側 HiRDB のシステム共通定義の `pd_rpl_init_start` オペランドに `Y` を指定する運用をお勧めします。

(2) HiRDB Datareplicator 連携の中止

HiRDB Datareplicator 連携は、次の指定によって中止されます。

- `pdrplstop` コマンドの実行
- 抽出側 HiRDB のシステム共通定義の `pd_rpl_init_start` オペランドの変更 (Y から N への変更、又は `pd_rpl_init_start` オペランドの省略)

HiRDB Datareplicator 連携を中止した場合は、抽出側 DB と反映側 DB との間に不整合が発生するので、HiRDB Datareplicator 連携を再開する前に反映側 DB の再作成が必要です。`pdrplstop` コマンドを実行する場合の運用手順については、「[6.5.6 抽出側 HiRDB の運用手順](#)」を参照してください。

pdrplstop コマンドでの HiRDB Datareplicator 連携停止は、再開時には引き継がれますが、正常開始時には引き継がれません。

pd_rpl_init_start オペランドの変更によってデータ連動を打ち切る場合は、pd_rpl_hdepath オペランドの指定値は変更しないでください。

6.5.2 データ連動用連絡ファイル

データ連動用連絡ファイルとは、HiRDB での更新情報のシステムログファイルへの格納状態や、抽出側 Datareplicator での更新情報のシステムログファイルからの読み込み状態など、抽出処理を実行するために必要な情報を管理するファイルです。

このファイルは、抽出側 HiRDB、及び抽出側 Datareplicator の起動時に、双方から参照または更新できる状態にしておく必要があります。

データ連動用連絡ファイルの運用では、次に示す制約があります。

- 抽出側 Datareplicator の初期化によってデータ連動用連絡ファイルを初期化した場合や、データ連動用連絡ファイルに障害が発生した場合、それまでの抽出側 Datareplicator でのデータ連動の続きからデータ連動を継続できないため、抽出側 DB と反映側 DB との間に不整合が発生する場合があります。この場合、反映側 DB の再作成が必要です。

反映側 DB を再作成する場合の運用手順については、「[6.5.6 抽出側 HiRDB の運用手順](#)」を参照してください。

なお、抽出側 HiRDB は、シンクポイントダンプ取得時にデータ連動用連絡ファイルの状態を確認しますが、その時点でデータ連動用連絡ファイルが初期化されている場合は、HiRDB Datareplicator 連携を中止して、抽出側 HiRDB の処理だけを続行します。

- 系切り替え環境の場合、データ連動用連絡ファイルを格納する共用ディスクは抽出側 HiRDB のサーバで管理してください。またサーバの起動順序は、抽出側 HiRDB の起動後、抽出側 Datareplicator を起動してください。

詳細は、「[6.9 系切り替え機能を使うときの運用](#)」を参照してください。

6.5.3 システムログファイルの運用

抽出側 HiRDB は、HiRDB Datareplicator 連携を開始するとデータ連動の状態を管理するために、システムログファイルの待機の状態に、抽出側 Datareplicator でのシステムログファイルの抽出が完了しているかどうかの状態を新たに追加します。この状態を**抽出状態**といいます。

ここでは、次の項目について説明します。

- システムログファイルの状態
- 抽出状態

- 抽出状態の設定
- 抽出状態の解消
- 抽出未完了状態のためにスワップ先にできない場合の指定
- 抽出未完了状態のために抽出側 HiRDB のユニットが強制停止した場合の処置
- システムログファイルの操作

(1) システムログファイルの状態

HiRDB Datareplicator 連携機能を使っているときのシステムログファイルの待機状態を次の表に示します。システムログファイルの状態は、HiRDB の pdlogls コマンド及び pdls コマンドで確認できます。

表 6-6 HiRDB Datareplicator 連携機能を使っているときのシステムログファイルの待機状態

状態の種類			備考
待機	スワップ先にできる状態	上書きできる状態	すべての状態を満たしている場合にだけ、スワップ先にできます。
		アンロード済み状態	
		抽出完了状態※	
スワップ先にできない状態	上書きできない状態	アンロード待ち状態	どれか一つの状態であれば、スワップ先にできません。
		抽出未完了状態※	

注※

HiRDB Datareplicator 連携を実行している場合にだけ追加される状態

注意事項

HiRDB Datareplicator 連携を開始した後、長時間にわたり抽出側 Datareplicator の抽出処理を停止しておく、HiRDB のシステムログファイルが満杯となり、HiRDB が停止することがあります。そのため、HiRDB の稼働中は、抽出側 Datareplicator の抽出処理を停止しない運用をお勧めします。

(2) 抽出状態

システムログファイルの抽出状態には、次の二つの状態があります。

抽出完了状態

抽出側 Datareplicator での抽出が完了している状態です。システムログファイル中のシステムログを抽出側 Datareplicator がすべて読み込むと、そのシステムログファイルは抽出完了状態になります。この状態で、上書きできる状態かつアンロード済み状態のシステムログファイルはスワップ先にできません。

- **抽出未完了状態**

抽出側 Datareplicator での抽出が完了していない状態です。抽出側 Datareplicator でまだシステムログを抽出していないシステムログファイルと抽出側 Datareplicator でシステムログを抽出中のシステムログファイルが、抽出未完了状態になります。この状態の場合は、上書きできる状態かつアンロード済み状態のシステムログファイルであってもスワップ先にはなりません。抽出未完了状態のシステムログファイルを強制的にスワップ先にできる状態にしたり、初期化したりした場合、抽出側 DB の更新情報の一部が反映側 DB に反映できないため、抽出側 DB と反映側 DB との間に不整合が発生します。この場合、反映側 DB の再作成が必要になります。

pdlogls コマンドは、システムログ内に未抽出の更新情報がない場合でも、HiRDB の内部制御のために現用のファイルグループを抽出未完了状態として出力する場合があります。このため、システムログ内に未抽出の更新情報があるかどうかを確認する場合は pdls -d rpl -j コマンドを実行して、その出力結果で判定します。

出力結果の System Log Extract Point として出力されている「Run ID」、 「Gen No」、 「Block No」の値のすべてが System Log Sync info として出力されている同項目の値より大きくなっていけば、システムログ内の更新情報はすべて抽出されていることを示します。

(3) 抽出状態の設定

システムログファイルの抽出状態は、次の指定によって HiRDB Datareplicator 連携が開始されたときに設定されます。

- 抽出側 HiRDB のシステム共通定義での pd_rpl_init_start = Y
- pdrplstart コマンド

抽出側 HiRDB は、シンクポイントダンプ取得時に各システムログファイルの抽出状態を確認します。シンクポイントダンプ取得時に、抽出側 Datareplicator での抽出が完了しているシステムログファイルの抽出状態が、抽出未完了状態から抽出完了状態に変更されます。このため、抽出側 Datareplicator での抽出が完了しているシステムログファイルであっても、シンクポイントダンプ取得時までは抽出完了状態にはなりません。

(4) 抽出状態の解消

システムログファイルの抽出状態は、システムログファイルがスワップ先として選択可能な状態になった時点で解消されます。スワップ先になるまでは抽出状態は解消されないため、pdlogls コマンドで表示した場合、まだスワップしていないシステムログファイルでは抽出状態が表示されます。また、この状態のシステムログファイルは、抽出側 Datareplicator の抽出対象になります。

ただし、次の場合は強制的に抽出状態が解消されます。

- pdrplstop コマンドの実行
- システム共通定義の pd_log_rpl_no_standby_file_opr = continue 指定時に、システムログファイルが抽出未完了状態のためにスワップ先にできない場合

- 抽出側 HiRDB のシステム共通定義の `pd_rpl_init_start` オペランドの変更 (Y から N への変更, 又は `pd_rpl_init_start` オペランドの省略)
- `-i` オプション指定の `hdestart` コマンドの実行
- 抽出未完了状態のシステムログファイルの初期化と抽出未完了状態のシステムログファイルの抽出状態の強制変更
- データ連動用連絡ファイルの初期化

強制的に抽出状態が解消された場合, 次に HiRDB Datareplicator 連携を開始したときに, 前回の抽出側 Datareplicator でのデータ連動の続きからデータ連動を継続できないため, 抽出側 DB と反映側 DB との間に不整合が発生する場合があります。この場合, 反映側 DB の再作成が必要です。

(5) 抽出未完了状態のためにスワップ先にできない場合の指定

すべてのシステムログファイルがスワップ先にできない場合に, 抽出未完了状態のためにスワップ先にできないシステムログファイルがあったときのデータ連動に対する処置について指定できます。

この指定は, 抽出側 HiRDB のシステム共通定義で指定します。指定方法を次に示します。

```
pd_log_rpl_no_standby_file_opr = stop | continue
```

システム共通定義での `pd_log_rpl_no_standby_file_opr` の指定方法については, 「5.6 抽出側 HiRDB の定義」を参照してください。

システムログファイルがスワップ先にできない場合の運用手順については, 「6.5.6 抽出側 HiRDB の運用手順」を参照してください。

- データ連動を継続して続行するために抽出側 HiRDB の該当ユニットを強制停止する (stop 指定時又は `pd_log_rpl_no_standby_file_opr` 省略)

データ連動を継続して続行したい場合は, システム共通定義で HiRDB のユニットの強制停止を指定します。この場合, 抽出未完了状態のためにスワップ先にできるファイルがなくなった時点で, 抽出側 HiRDB は該当ユニットを強制停止します。

強制停止した抽出側 HiRDB のユニットを再開する場合は, 抽出側 Datareplicator で抽出中のログファイルが別のファイルに移った (抽出完了のファイルが一つ以上できた) ことを確認してから, 抽出側 HiRDB のユニットを再開してください。

システムログファイルの抽出状態は, `pdlogls` コマンドでそれぞれのシステムログファイルの状態を確認してから, `pdls -d rpl -j` コマンドで抽出がどこまで進んでいるかを確認することで判定できます。

抽出側 HiRDB のユニットの強制停止時に, データ連動を中止して抽出側 HiRDB の処理だけを続行したい場合については, 「(6) 抽出未完了状態のために抽出側 HiRDB のユニットが強制停止した場合の処置」を参照してください。

- データ連動を中止して抽出側 HiRDB での処理だけを続行する (continue 指定)

データ連動を中止して抽出側 HiRDB での処理だけを続行したい場合は, システム共通定義で HiRDB Datareplicator 連携の中止を指定します。この場合, 抽出未完了状態のためにスワップ先にできるファ

イルがなくなった時点で、抽出側 HiRDB はすべてのバックエンドサーバでの HiRDB Datareplicator 連携を中止し、抽出側 HiRDB だけで運用を続行します。抽出側 HiRDB は、スワップ先にできるファイルがなくなった時点ですべてのシステムログファイルの抽出状態を解消します。これによって、抽出未完了状態のためにスワップ先にできなかったシステムログファイルがスワップ先にできる状態になるため、抽出側 HiRDB の処理が続行できます。

抽出状態を解消した場合は、抽出側 DB と反映側 DB との間に不整合が発生するので、HiRDB Datareplicator 連携を再開する前に反映側 DB の再作成が必要です。

(6) 抽出未完了状態のために抽出側 HiRDB のユニットが強制停止した場合の処置

システム共通定義の指定によって、システムログファイルが抽出未完了状態のために抽出側 HiRDB のユニットが強制停止した場合に、抽出側 HiRDB の処理を続行するためには次のどちらかを実行してください。

- データ連動を続行する場合

抽出側 Datareplicator が停止している場合は、抽出側 Datareplicator を起動してください。抽出プロセス、送信プロセスの両方とも起動している場合は、送信先の反映側 Datareplicator の状態を確認し、停止している場合は起動してください。

- データ連動を中止して、抽出側 HiRDB の処理だけを続行する場合

抽出側 Datareplicator 又は反映側 Datareplicator の障害などによって、システムログファイルが抽出完了状態にならない場合に抽出側 HiRDB の処理を続行するためには、HiRDB Datareplicator 連携を中止して抽出側 HiRDB を再開します。この場合は、次のどちらかを実行してから、抽出側 HiRDB を再開してください。

- システム共通定義の `pd_log_rpl_no_standby_file_opr` オペランドの指定を `continue` に変更します。
- `pdlogchg` コマンドですべてのシステムログファイルのステータスを強制的に抽出完了状態にします。

HiRDB Datareplicator 連携を中止した場合は、抽出側 DB と反映側 DB との間に不整合が発生するので、HiRDB Datareplicator 連携を再開する前に反映側 DB の再作成が必要です。

抽出側 HiRDB のユニット強制停止時にデータ連動を中止する場合の運用手順については、「[6.5.6 抽出側 HiRDB の運用手順](#)」を参照してください。

(7) システムログファイルの操作

システムログファイルの操作について、データ連動する場合に特に注意が必要な次の項目について説明します。

- システムログファイルの抽出状態の確認 (`pdlogls` コマンド)
- システムログファイルの抽出状態の強制変更 (`pdlogchg` コマンド)
- 抽出未完了状態のシステムログファイルの削除 (`pdlogrm` コマンド)

このほかの操作については、マニュアル「HiRDB システム運用ガイド」を参照してください。

- システムログファイルの抽出状態の確認 (pdlogls コマンド)

システムログファイルの抽出状態は、HiRDB の pdlogls コマンドで確認できます。HiRDB Datareplicator 連携機能を使っている場合は、ファイルグループ、要素ファイル、及び物理ファイルの状態のアンロード状態の表示に、次の表示が追加されます。

a：アンロード待ち状態，かつ抽出未完了状態

アンロード待ち状態とは、アンロードする必要があるログを含んでいる状態です。

抽出未完了状態とは、HiRDB Datareplicator によるデータ連動情報の抽出が完了していない状態です。

u：アンロード待ち状態，かつ抽出完了状態

e：アンロード済み状態，かつ抽出未完了状態

-：アンロード済み状態，かつ抽出完了状態

システムログファイルの抽出状態は、抽出側 HiRDB の該当するユニットが停止している場合には変更されません。この場合に抽出側 Datareplicator での抽出が完了しても、pdlogls コマンドで表示すると、該当するシステムログファイルの状態は抽出未完了状態が表示されます。

このため、該当するユニットが停止している場合にシステムログファイルの抽出状態を確認するときは、HiRDB の pdls コマンドで、システムログファイルでの抽出側 Datareplicator の抽出状況を確認してください。

- システムログファイルの抽出状態の強制変更 (pdlogchg コマンド)

抽出側 Datareplicator 又は反映側 Datareplicator の障害などによって、システムログファイルが抽出完了状態にならない場合に抽出側 HiRDB の処理を続行するために、HiRDB の pdlogchg コマンドを使ってシステムログファイルの抽出状態を強制的に変更します。

pdlogchg コマンドで -R オプションを指定すると、システムログファイルの抽出状態を抽出未完了状態から抽出完了状態に変更できます。

pdlogchg コマンドで抽出状態を強制的に変更した場合、抽出側 DB と反映側 DB との間に不整合が発生します。この場合は HiRDB Datareplicator 連携を中止して、反映側 DB を再作成してから、HiRDB Datareplicator 連携を再開してください。

pdlogchg コマンド実行時の運用手順については、「[6.5.6 抽出側 HiRDB の運用手順](#)」を参照してください。pdlogchg コマンドの -R オプションの指定方法については、「[7. コマンドの文法](#)」を参照してください。

- 抽出未完了状態のシステムログファイルの削除 (pdlogrm コマンド)

抽出未完了状態のシステムログファイルを強制的に削除する場合は、HiRDB の pdlogrm コマンドに -u オプションを指定します。-u オプションの指定がない場合は、抽出未完了状態のシステムログファイルは削除できません。抽出未完了状態のシステムログファイルを削除した場合、抽出側 DB と反映側 DB との間に不整合が発生するので、次に HiRDB Datareplicator 連携を開始する前に反映側 DB の再作成が必要です。

6.5.4 抽出環境定義の extsuppress の指定と、抽出側 HiRDB 及び抽出側 Datareplicator での処理

抽出環境定義の extsuppress の指定によって、抽出側 HiRDB 及び抽出側 Datareplicator の処理方法が異なります。抽出環境定義の extsuppress の指定と、抽出側 HiRDB 及び抽出側 Datareplicator での処理を次の表に示します。

表 6-7 抽出環境定義の extsuppress の指定と、抽出側 HiRDB 及び抽出側 Datareplicator での処理

HiRDB のサーバ種別	抽出側 Datareplicator での指定		抽出側 HiRDB の処理	抽出側 Datareplicator 起動後の処理
	抽出環境定義 extsuppress	抽出定義での指定		
MGR FES DS	—	—	—	—
BES SDS	true	抽出対象	抽出状態を設定しません。	定義矛盾のメッセージを出力後、該当サーバでの処理を停止します。 「(1) true 指定時に定義を変更して抽出対象にする場合の運用方法」を参照してください。
		抽出対象外		抽出対象外のメッセージを出力後、該当サーバでの処理を停止します。
	false	抽出対象	抽出状態を設定します。	データ連動処理を開始します。
		抽出対象外	抽出側 Datareplicator の起動前 抽出状態を設定します。 抽出側 Datareplicator の起動後 抽出状態を解消します。	抽出対象の表がないことを示すメッセージを出力後、該当サーバでの処理を停止します。 「(2) false 指定時の抽出対象の表がないサーバでの注意」を参照してください。

(凡例)

MGR：システムマネージャ

FES：フロントエンドサーバ

DS：ディクショナリサーバ

BES：バックエンドサーバ

SDS：シングルサーバ

—：該当しない。

抽出対象：該当サーバに、抽出定義で抽出対象に指定した表がある場合

抽出対象外：該当サーバに、抽出定義で抽出対象に指定した表がない場合

(1) true 指定時に定義を変更して抽出対象にする場合の運用方法

true 指定時に定義矛盾で抽出側 Datareplicator の処理が停止した場合に、定義を変更して抽出対象にしたい場合は、抽出環境定義を修正してから、抽出側 Datareplicator を再初期化 (hdestart -i コマンド実行) し、その後で抽出側 Datareplicator を起動してください。

ただし、抽出側 Datareplicator の再初期化の前に、抽出側 HiRDB で HiRDB Datareplicator 連携を開始している場合は、抽出側 DB と反映側 DB との間に不整合が発生します。この場合は、HiRDB Datareplicator 連携を中止して、抽出側 DB を基に反映側 DB を再作成[※]してから、抽出側 Datareplicator の再初期化をしてください。

注※

抽出側 HiRDB のデータ連携対象表を更新していない場合は、反映側 DB を再作成する必要はありません。HiRDB Datareplicator 連携を中止して、抽出側 Datareplicator の再初期化をしてください。

(2) false 指定時の抽出対象の表がないサーバでの注意

false 指定時、抽出側 Datareplicator の初期化後に抽出側 Datareplicator を起動しないで HiRDB Datareplicator 連携機能を実行し続けると、抽出対象の表がないサーバでも、システムログファイルの満杯が発生する可能性があります。このため、HiRDB Datareplicator 連携を開始する場合は、必ず抽出側 Datareplicator を起動してください。

抽出対象の表がないサーバで、システムログファイルの満杯が発生した場合のユーザの取る処置は、システム共通定義の pd_log_rpl_no_standby_file_opr の指定によって異なります。

指定値ごとのユーザの取る処置を次に示します。

- stop 指定時

抽出対象の表がないサーバでは、抽出側 Datareplicator を起動するとシステムログファイルの抽出状態が解消されます。このため、抽出側 Datareplicator を起動してから強制停止したサーバを再起動してください。これによって、ほかのサーバでのデータ連動は続行できます。

- continue 指定時

抽出未完了状態のためにスワップ先にできるファイルがなくなった時点で、抽出側 HiRDB はすべてのバックエンドサーバでの HiRDB Datareplicator 連携を中止します。この場合は、抽出側 DB と反映側 DB との間に不整合が発生するので、HiRDB Datareplicator 連携を再開する前に反映側 DB の再作成が必要です。

6.5.5 HiRDB Datareplicator 連携の実行状況の確認

HiRDB Datareplicator 連携の実行状況は、HiRDB の pdls コマンドで確認できます。表示対象種別に rpl を指定すると、HiRDB Datareplicator 連携の状態を表示します。これによって、HiRDB Datareplicator 連携の次の状況が確認できます。

- HiRDB Datareplicator 連携機能を使っているかどうか
- 抽出側 Datareplicator のシステムログファイルからのシステムログの抽出状況

pdls コマンドの rpl オプションの指定方法については、「[7. コマンドの文法](#)」を参照してください。

6.5.6 抽出側 HiRDB の運用手順

抽出側 HiRDB の運用手順について説明します。運用手順を運用の目的と状況に分けて説明します。抽出側 HiRDB の運用手順と参照先を次の表に示します。

表 6-8 抽出側 HiRDB の運用手順と参照先

運用の目的と状況	参照先	
	項番	タイトル
抽出側 Datareplicator の環境を初期化する場合 (hdestart -i コマンド実行)	(1)	反映側 DB の再作成時の運用手順
反映側 Datareplicator の環境を初期化する場合 (hdsstart -i コマンド実行)		
反映側 DB を再作成する場合		
データ連動を中止した場合		
システムログファイル又はデータ連動用連絡ファイルの初期化		
データ連動の継続実行	(2)	データ連動の継続実行時の運用手順
抽出対象 DB の定義変更	(3)	抽出側の表定義と抽出定義変更時の運用手順
抽出定義の変更		
pdrplstart コマンドでの HiRDB Datareplicator 連携の開始	(4)	pdrplstart コマンド実行時の運用手順
pdrplstop コマンドでの HiRDB Datareplicator 連携の中止	(5)	pdrplstop コマンド実行時の運用手順
pdlogchg コマンドでの抽出状態の変更	(6)	pdlogchg コマンド実行時の運用手順
抽出側 DB の再作成	(7)	抽出対象 DB の再作成時の運用手順
抽出側 HiRDB でのシステム再構築	(8)	抽出側 HiRDB でのシステム再構築時の運用手順
抽出側 HiRDB でのシステム定義の変更	(9)	抽出側 HiRDB でのシステム定義の変更時の運用手順
システムログファイルの満杯発生	(10)	システムログファイルがスワップ先にできない場合の運用手順
システムログファイルの抽出未完了状態での HiRDB のユニット強制停止時にデータ連動を中止する場合	(11)	抽出側 HiRDB のユニット強制停止時にデータ連動を中止する場合の運用手順
抽出側 HiRDB 異常終了	(12)	抽出側 HiRDB 異常終了時の運用手順

運用の目的と状況	参照先	
	項番	タイトル
送信先の追加と削除	(13)	送信先の追加／削除の運用手順

(1) 反映側 DB の再作成時の運用手順

すべてのデータが連動し終わる前に次の操作をした場合は、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化し、反映側 DB を再作成してください。

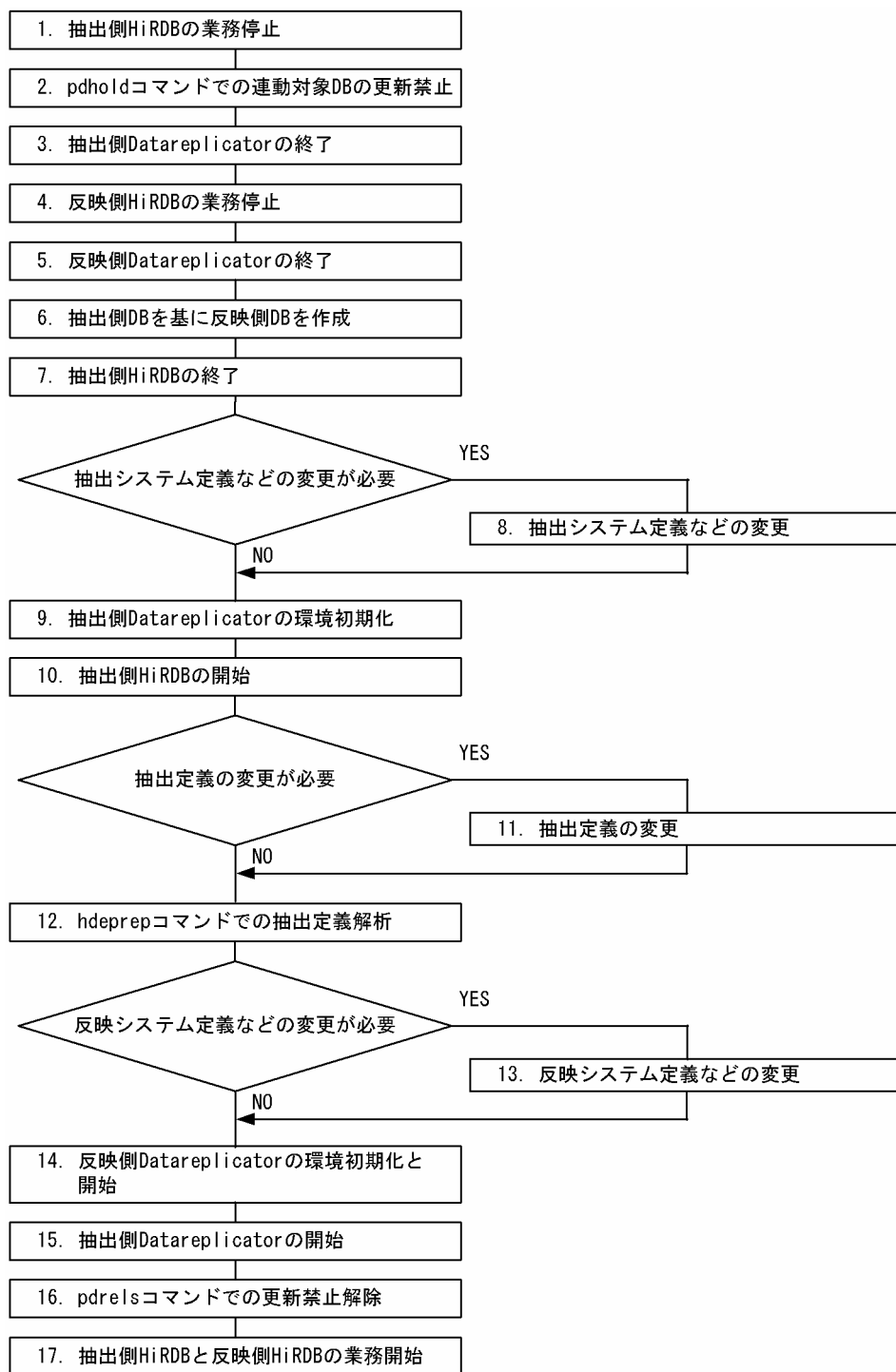
- 抽出側 Datareplicator の環境を初期化 (hdestart -i コマンドを実行)
- 反映側 Datareplicator の環境を初期化 (hdsstart -i コマンドを実行)
- 反映側 DB を再作成
- データ連動を中止
- システムログファイル又はデータ連動用連絡ファイルを初期化

なお、すべてのデータが連動し終わったことを確認してから初期化する場合で、反映側 DB を再作成する必要がないときの初期化手順については、「[6.8 HiRDB 及び Datareplicator の構成変更](#)」を参照してください。

また、エラーの対処方法については、「[9.2.2 エラーの対処方法](#)」を参照してください。

反映側 DB の再作成時の運用手順を次の図に示します。

図 6-11 反映側 DB の再作成時の運用手順



運用手順を図に基づいて説明します。説明の 1.~17. は、図中の番号に対応しています。なお、抽出側 HiRDB がパラレルサーバの場合には、hdeprep コマンド、及び hdestart コマンドは、システムマネージャのあるサーバから実行してください。

1. 連動対象の抽出側 DB と反映側 DB との整合性を保つために、抽出側 HiRDB の業務を停止します。
2. 抽出側のデータ連動対象 DB を pdhold コマンドで閉塞して、データ連動対象 DB の更新を抑止します。
3. 抽出側 Datareplicator を停止します。

4. 反映側 HiRDB の業務を停止します。
5. 反映側 Datareplicator を停止します。
6. 抽出側 DB を基に反映側 DB を作成します。この場合、HiRDB Dataextractor を使うと、抽出側 DB から反映側 DB を効率的に作成できます。
7. 抽出側 HiRDB を停止します。
8. 次の定義ファイルの変更が必要な場合は変更します。
 - 抽出システム定義ファイル
 - 送信環境定義ファイル
 - 抽出環境定義ファイル
9. 抽出側 Datareplicator の環境を初期化します (hdestart -i コマンド実行)。
10. 抽出側 HiRDB のシステム共通定義で pd_rpl_init_start = Y を指定して抽出側 HiRDB を起動します。
11. 抽出定義の変更が必要な場合は変更します。
12. 抽出定義を hdeprep コマンドで解析して、抽出定義プリプロセスファイルを作成します。
13. 次の定義ファイルの変更が必要な場合は変更します。
 - 反映システム定義ファイル
 - 反映環境定義ファイル
 - 反映定義ファイル
14. 反映側 Datareplicator の環境を初期化して、反映側 Datareplicator を起動します (hdsstart -i コマンド実行)。
15. 抽出側 Datareplicator を起動します。
16. 抽出側のデータ連動対象 DB の閉塞を pdrels コマンドで解除します。
17. 抽出側 HiRDB と反映側 HiRDB の業務を再開します。

(2) データ連動の継続実行時の運用手順

HiRDB のシステム共通定義に pd_rpl_init_start = Y を指定している場合は、抽出側 HiRDB の開始又は再開に関係なく、前回の抽出側 HiRDB でのデータ連動情報のシステムログへの出力状態と抽出側 Datareplicator でのシステムログの抽出状態を引き継いで、HiRDB Datareplicator 連携を実行します。このため、pd_rpl_init_start = Y 指定時は、抽出側 Datareplicator を hdestart コマンドで起動すると、前回のデータ連動の続きからデータ連動が継続して実行されます。

ただし、pdrplstop コマンドを実行した場合は抽出状態が取り消されるため、pd_rpl_init_start = Y を指定していても、DB の再作成が必要になります。

データ連動を継続して実行している場合 (システム共通定義に pd_rpl_init_start = Y 指定時) の運用手順を次に示します。

1. 抽出側 HiRDB を起動します（システム共通定義に `pd_rpl_init_start = Y` を指定）。
2. 抽出側 Datareplicator を `hdestart` コマンドで起動します。
3. 必要に応じて、抽出側 HiRDB を終了します。
4. 抽出側 Datareplicator は、抽出システム定義で `syncterm=true` を指定している場合は、抽出側 HiRDB の正常終了時に同期して終了します。この指定がない場合に抽出側 Datareplicator を終了させたいときは、`hdestop` コマンドを実行します。

(3) 抽出側の表定義と抽出定義変更時の運用手順

抽出側 HiRDB の連動対象 DB の表定義と、抽出側 Datareplicator の抽出定義を変更する場合の運用手順について説明します。連動対象 DB の表定義の内容を変更しなくても、`CREATE TABLE` をやり直す場合には、ここで示す手順を実施する必要があります。定義の変更時は、すべての抽出対象のバックエンドサーバで、抽出側 HiRDB のシステムログをすべて抽出して、反映側へ送信する（抽出情報キューファイル中の更新情報もすべて送信する）ことが必要です。

定義変更時の運用手順には、次の二つの場合があります。それぞれの場合に分けて説明します。

- 抽出システム定義に `syncterm=true` 指定時で抽出側 HiRDB を正常終了できる場合
- 抽出システム定義で `syncterm=true` を指定していない場合、及び抽出側 HiRDB を正常終了できない場合

抽出システム定義に `syncterm=true` 指定時で抽出側 HiRDB を正常終了できる場合

1. 抽出側 HiRDB を正常終了します。
抽出側 Datareplicator は、抽出側 HiRDB のシステムログをすべて抽出して反映側へ送信して、送信が完了した時点で自動的に終了します。
2. 抽出側 HiRDB を正常開始します。
システム共通定義に `pd_rpl_init_start = N` を指定して正常開始します。この場合、HiRDB Datareplicator 連携を開始する前に、データ連動対象 DB を更新するトランザクションを実行しないでください。
3. 抽出側の定義を変更します。
表定義の場合：`ALTER TABLE` や、定義系 SQL で定義を変更します。
抽出定義の場合：抽出定義を変更します。
4. 抽出側 HiRDB がパラレルサーバの場合に、抽出対象表を追加又は削除したときは、「[5.3.3 オペランドの説明](#)」の抽出環境定義の「`extsuppress`」の指定値を見直してください。
なお、抽出側 HiRDB にある連動対象 DB の表定義を、一つのバックエンドサーバ内だけで変更する場合には、この手順は必要ありません。
5. 抽出側 Datareplicator を `hdestart -i` コマンドで初期化した後、`hdeprep` コマンドで、抽出定義プリプロセスファイルを再作成します。
抽出側の表定義だけを変更する場合も、`hdeprep` コマンドは必ず実行してください。`hdeprep` コマンドは、抽出側 HiRDB 稼働中にだけ実行できます。

6. 抽出側 Datareplicator を hdestart コマンドで起動します。

7. 一度抽出側 HiRDB を正常終了し、pd_rpl_init_start の指定を N から Y に変更して正常開始します。

抽出システム定義で syncterm=true を指定していない場合、又は抽出側 HiRDB を正常終了できない場合

1. データ連動対象 DB を pdhold コマンドで閉塞して、データ連動対象 DB の更新を抑止します。

2. 抽出側 Datareplicator で、抽出側 HiRDB のシステムログをすべて抽出して、反映側へ送信します（抽出情報キューファイル中の更新情報もすべて送信してください）。

抽出状態と送信状態は、次の方法で確認します。

- ・ pdls コマンド
- ・ hdestate コマンド
- ・ 抽出側 Datareplicator の出力するメッセージ

3. 抽出側 HiRDB のすべてのシステムログの抽出完了と送信完了を確認してから、抽出側 Datareplicator を hdestop コマンドで終了します。

システムログの抽出完了と送信完了の確認手順については、「[6.8.2 抽出側システムの構成変更](#)」の手順 2 を参照してください。

4. 次の変更をする場合は、抽出側 HiRDB での HiRDB Datareplicator 連携を pdrplstop コマンドで中止します。

- ・ 抽出対象の表がないバックエンドサーバに抽出対象の表を追加する場合
- ・ 抽出対象の表を削除する場合

5. 抽出側の定義を変更します。

表定義の場合：ALTER TABLE や、定義系 SQL で定義を変更します。

抽出定義の場合：抽出定義を変更します。

6. 抽出側 HiRDB がパラレルサーバの場合に、抽出対象表を追加又は削除したときは、「[5.3.3 オペランドの説明](#)」の抽出環境定義の「extsuppress」の指定値を見直してください。

なお、抽出側 HiRDB にある連動対象 DB の表定義を、一つのバックエンドサーバ内だけで変更する場合には、この手順は必要ありません。

7. 抽出側 Datareplicator を hdestart -i コマンドで初期化した後、hdeprep コマンドで、抽出定義プリプロセスファイルを再作成します。

抽出側の表定義だけを変更する場合も、hdeprep コマンドは必ず実行してください。hdeprep コマンドは、抽出側 HiRDB 稼働中にだけ実行できます。

8. 抽出側 Datareplicator を hdestart コマンドで起動します。

9. 抽出側 HiRDB での HiRDB Datareplicator 連携を中止している場合は、pdrplstart コマンドで再開始します。

10. データ連動対象 DB の閉塞を pdrels コマンドで解除します。

(4) pdrplstart コマンド実行時の運用手順

pdrplstart コマンド実行時の運用手順を次に示します。

1. データ連動対象 DB を pdhold コマンドで閉塞して、データ連動対象 DB の更新を抑止します。
2. 反映側 DB を作成していない場合、又は抽出側 DB と反映側 DB との間に不整合が発生している場合は、抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。
3. 抽出側 HiRDB で、pdrplstart コマンドを実行してデータ連動を開始します。
4. 抽出側 Datareplicator を hdestart コマンドで起動します。
5. データ連動対象 DB の閉塞を pdrels コマンドで解除します。

(5) pdrplstop コマンド実行時の運用手順

pdrplstop コマンドを実行して、HiRDB Datareplicator 連携を中止する場合の運用手順を次に示します。

1. pdrplstop コマンドで HiRDB Datareplicator 連携を中止します。
2. 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。
3. 抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(6) pdlogchg コマンド実行時の運用手順

pdlogchg コマンドを使って、システムログファイルの抽出状態を強制的に変更する場合の運用手順を次に示します。HiRDB Datareplicator 連携中に pdlogchg コマンドで抽出状態を変更した場合、反映側 DB の再作成が必要です。

1. pdrplstop コマンドで HiRDB Datareplicator 連携を中止します。
2. 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。
3. -R オプションを指定した pdlogchg コマンドで、システムログファイルの抽出未完了状態を抽出完了状態に変更します。抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(7) 抽出対象 DB の再作成時の運用手順

抽出対象 DB の再作成時の運用手順を次に示します。

1. pdrplstop コマンドで HiRDB Datareplicator 連携を中止します。

- 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。
- 抽出対象 DB を再作成します。
- データ連動対象 DB を pdhold コマンドで閉塞して、データ連動対象 DB の更新を抑止します。
- 抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(8) 抽出側 HiRDB でのシステム再構築時の運用手順

抽出側 HiRDB でのシステム再構築時の運用手順を次に示します。

- 抽出側 HiRDB を正常終了します。
- 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。
- 抽出側 HiRDB のシステムを再構築して、システム共通定義に pd_rpl_init_start = N を指定して抽出側 HiRDB を正常開始します (-i オプション指定)。
- 抽出側 DB を再作成します。
- 抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(9) 抽出側 HiRDB でのシステム定義の変更時の運用手順

抽出側 HiRDB でシステム定義を変更するときの運用手順を次に示します。

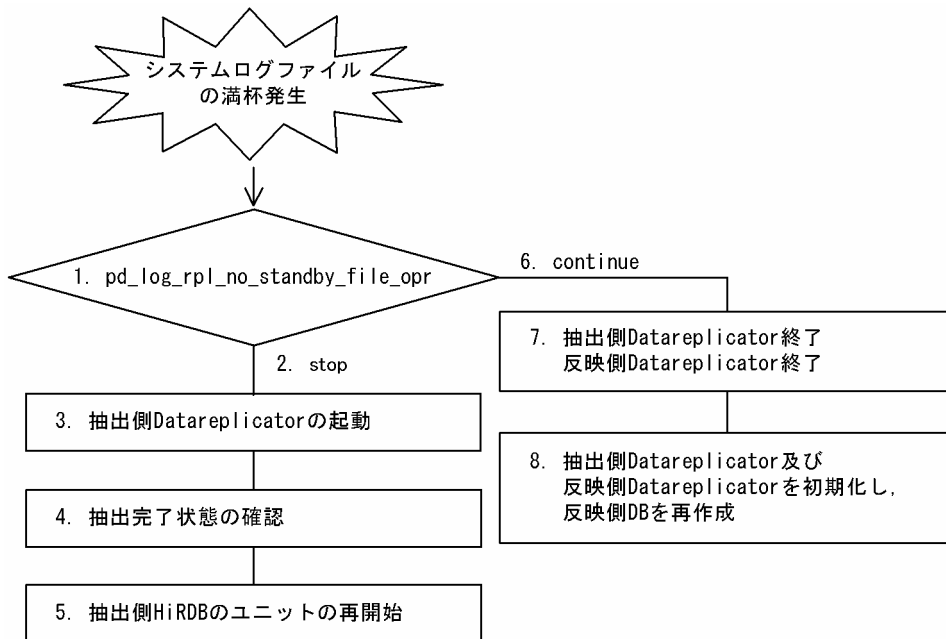
- 抽出側 HiRDB を正常終了します。
- 抽出側 Datareplicator を hdestop コマンドで終了します。
- 抽出側 HiRDB のシステム定義を変更します。
- 抽出側 HiRDB を開始します。
- 抽出側 Datareplicator を hdestart コマンドで起動します。

(10) システムログファイルがスワップ先にできない場合の運用手順

すべてのシステムログファイルがスワップ先にできない場合に、抽出未完了状態のためにスワップ先にできないシステムログファイルがあったときの、運用手順について説明します。

すべてのシステムログファイルがスワップ先にできない場合の運用手順を次の図に示します。

図 6-12 すべてのシステムログファイルがスワップ先にできない場合の運用手順



運用手順を図に基づいて説明します。説明の 1.~8. は、図中の番号に対応しています。

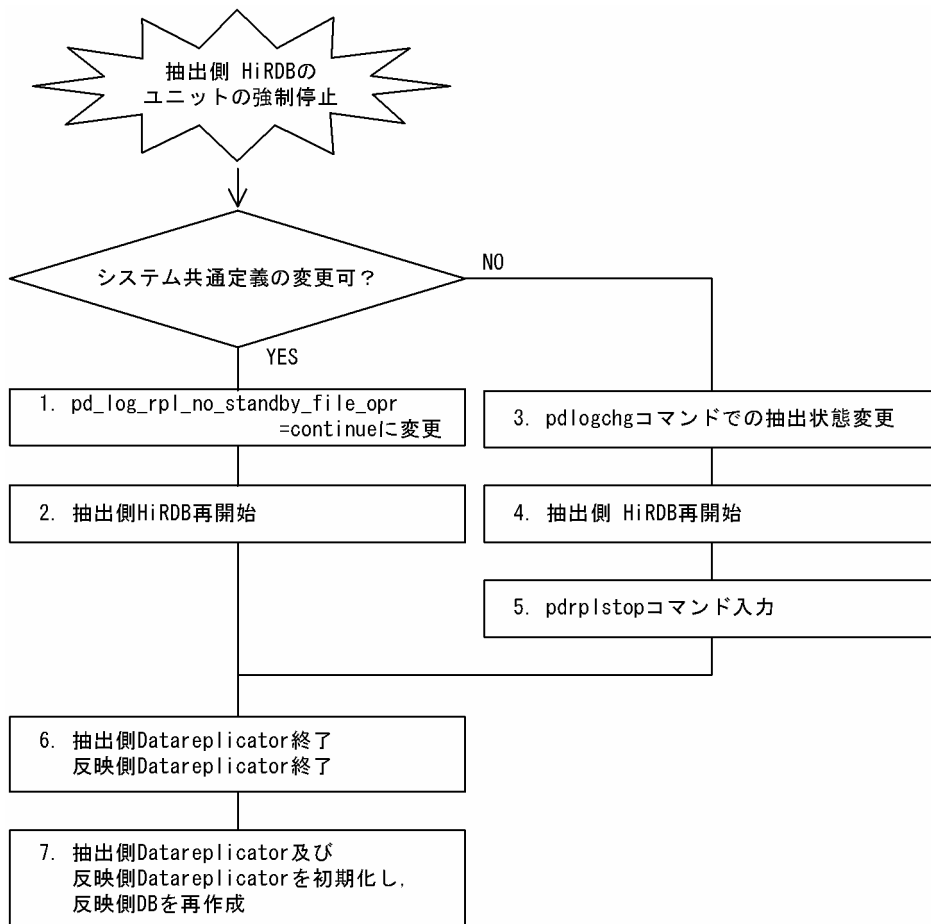
1. 抽出側 HiRDB は、システム共通定義の pd_log_rpl_no_standby_file_opr の指定に従って対応します。
2. stop 指定時又は pd_log_rpl_no_standby_file_opr 省略時は、抽出側 HiRDB の該当ユニットが強制停止します。
3. 抽出側 Datareplicator が停止している場合は起動します。
4. 抽出側 Datareplicator で抽出中のログファイルが別のファイルに移った（抽出完了のファイルが一つ以上できた）ことを確認します。システムログファイルの抽出状態は、pdlogls コマンドでそれぞれのシステムログファイルの状態を確認してから、pdls -d rpl -j コマンドで抽出がどこまで進んでいるかを確認すると判定できます。
5. 抽出側 HiRDB のユニットを再開始します。
6. continue 指定時は、HiRDB Datareplicator 連携を中止して抽出側 HiRDB での処理だけを続行します。抽出未完了状態のためにスワップ先にできるファイルがなくなった時点で、抽出側 HiRDB は HiRDB Datareplicator 連携を中止して、抽出側 HiRDB だけで運用を続行します。抽出側 HiRDB は、スワップ先にできるファイルがなくなった時点ですべてのシステムログファイルの抽出状態を解消します。
7. 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。
8. 抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(11) 抽出側 HiRDB のユニット強制停止時にデータ連動を中止する場合の運用手順

HiRDB のシステム共通定義の指定によって、システムログファイルが抽出未完了状態のために抽出側 HiRDB のユニットが強制停止した場合に、抽出側 Datareplicator を起動しても問題を解決できないとき、データ連動を中止して抽出側 HiRDB の処理だけを続行する運用手順について説明します。

抽出側 HiRDB のユニット強制停止時にデータ連動を中止する場合の運用手順を次の図に示します。

図 6-13 抽出側 HiRDB のユニット強制停止時にデータ連動を中止する場合の運用手順



運用手順を図に基づいて説明します。説明の 1.~7. は、図中の番号に対応しています。

1. 定義を変更できる場合は、システム共通定義の `pd_log_rpl_no_standby_file_opr` オペランドの指定を `continue` に変更します。
2. 抽出側 HiRDB を再開始します。
3. 定義を変更できない場合は、`-R` オプションを指定した `pdlogchg` コマンドをすべてのログファイルに対して実行して、抽出未完了状態を抽出完了状態に強制的に変更します。
4. 抽出側 HiRDB を再開始します。
5. `pdrplstop` コマンドで HiRDB Datareplicator 連携を中止します。

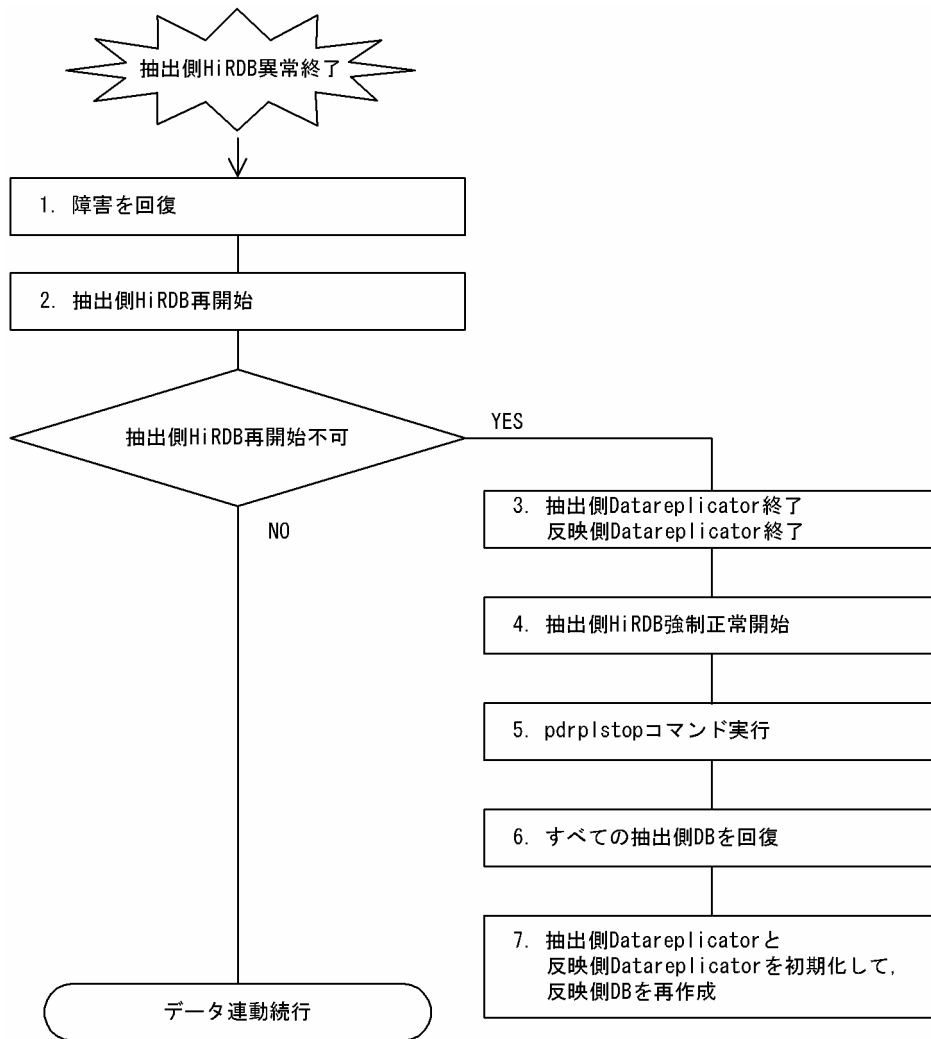
6. 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。

7. 抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(12) 抽出側 HiRDB 異常終了時の運用手順

抽出側 HiRDB が異常終了した場合の運用手順を次の図に示します。

図 6-14 抽出側 HiRDB が異常終了した場合の運用手順



運用手順を図に基づいて説明します。説明の 1.~7. は、図中の番号に対応しています。

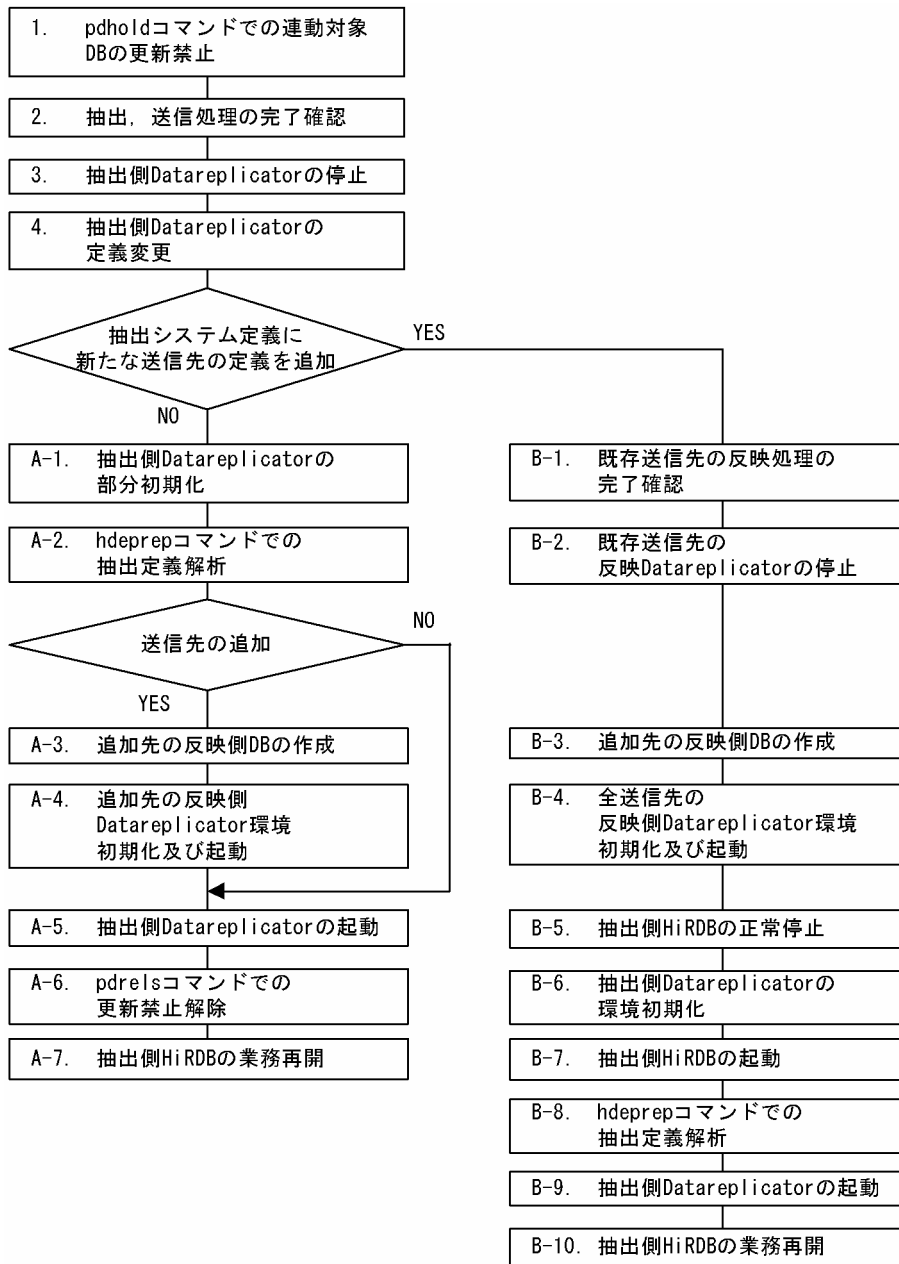
1. 抽出側 HiRDB の異常終了の原因を取り除きます。
2. 抽出側 HiRDB を再開始します。
3. 抽出側 Datareplicator を hdestop コマンドで終了して、反映側 Datareplicator を hdsstop コマンドで終了します。抽出側と反映側 Datareplicator の終了は、データ連動初期化までの任意の時期に実行してかまいません。

4. 抽出側 HiRDB を強制正常開始します。
5. pdrplstop コマンドで HiRDB Datareplicator 連携を中止します。
6. 抽出側 DB を再作成します。
7. 抽出側と反映側の両方のデータ連動の環境を、同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成します。詳細は、「(1)反映側 DB の再作成時の運用手順」を参照してください。

(13) 送信先の追加／削除の運用手順

抽出側 Datareplicator に対して、送信先を追加又は削除する場合の運用手順を次の図に示します。

図 6-15 送信先を追加又は削除するときの運用手順



運用手順を図に基づいて説明します。説明の 1.~4. は、図中の番号に対応しています。

1. データ連動対象 DB を pdhold コマンドで閉塞して、データ連動対象 DB の更新を抑止します。
2. 抽出側 Datareplicator で、抽出側 HiRDB のシステムログをすべて抽出し、反映側へ送信されていることを確認します（抽出情報キューファイル中の更新情報もすべて送信していることを確認してください）。

抽出状態と送信状態は、次の方法で確認します。

- pdls コマンド
 - hdestate コマンド
 - 抽出側 Datareplicator の出力するメッセージ
3. 抽出側 HiRDB のすべてのシステムログの抽出完了と送信完了を確認してから、抽出側 Datareplicator を hdestop コマンドで終了します。

システムログの抽出完了と送信完了の確認手順については、「[6.8.2 抽出側システムの構成変更](#)」の手順 2 を参照してください。

4. 抽出側の定義を変更します。

送信先を追加する場合

抽出システム定義で送信先の欠番を定義している場合は、欠番になっている定義を追加する送信先に変更します。欠番を定義していない場合は、新しい送信先に関する定義（sendid オペランド及び senddef オペランド）を追加します。

送信先を削除する場合

抽出システム定義の送信先定義を、欠番に変更します。

以降の手順は、手順 4 で抽出システム定義に対して新しい送信先を追加（抽出システム定義に対して sendid オペランド及び senddef オペランドを新たに追加）したかどうかで異なります。

新しい送信先を追加しなかった場合は手順 A を、追加した場合は手順 B に従ってください。

手順 A

1. 抽出側 Datareplicator に対して定義変更を行った送信先に対する部分初期化を実行します。
2. hdeprep コマンドを実行し、抽出定義プリプロセスを作成します。
3. 抽出側 DB を基に、追加する送信先の反映側 DB を作成します。
4. 追加する送信先の反映側 Datareplicator の環境を初期化します。
反映 Datareplicator 環境を初期化する際は、事前に次の定義ファイルを作成しておく必要があります。
 - 反映システム定義ファイル
 - 反映環境定義ファイル
 - 反映定義ファイル
5. hdestart コマンドを実行し、抽出側 Datareplicator を起動します。
6. pdrels コマンドを実行し、抽出側のデータ連動対象 DB の閉塞を解除します。

7. 抽出側 HiRDB の業務を再開します。

手順 B

1. 既存の全送信先の反映側 Datareplicator について、抽出側から送信されたデータがすべて反映されていることを確認します。

反映情報キューファイルの更新情報がすべて反映されていることを、hdsstate コマンドを実行して確認してください。hdsstate コマンドを実行して取得したステータス情報の、`<COMMUNICATION INFORMATION>`の write position と`<REFLECTION INFORMATION>`の read position が同じになっていれば、すべての更新情報が反映されています。

2. 既存の全送信先の反映側 Datareplicator を hdsstop コマンドで終了します。

3. 抽出側 DB を基に、追加する送信先の反映側 DB を作成します。

4. 既存の全送信先の反映側 Datareplicator 及び追加する送信先の反映側 Datareplicator の環境を初期化します。

追加する送信先の反映側 Datareplicator 環境を初期化する際は、事前に次の定義ファイルを作成しておく必要があります。

- ・ 反映システム定義ファイル
- ・ 反映環境定義ファイル
- ・ 反映定義ファイル

5. pdstop コマンドを実行し、抽出側 HiRDB を正常停止します。

6. hdestart -i コマンドを実行し、抽出側 Datareplicator の環境を初期化します。

7. pdstart コマンドを実行し、抽出側 HiRDB を起動します。

8. hdeprep コマンドを実行し、抽出定義プリプロセスを作成します。

9. hdestart コマンドを実行し、抽出側 Datareplicator を起動します。

10. 抽出側 HiRDB の業務を再開します。

6.5.7 syncterm=true を設定したときの運用上の注意

抽出システム定義に syncterm=true を設定した場合に、Datareplicator を操作するときの注意について説明します。

(1) HiRDB との同期停止前での Datareplicator の停止

syncterm=true を指定すると、Datareplicator は同期点（HiRDB の正常停止ログ検知）まで処理を続けて、その後自動的に停止します。ただし、抽出対象 HiRDB の構成がパラレルサーバ構成で、かつ Datareplicator が同期点に到達する前にユーザからの停止コマンド実行や異常停止などで停止した場合、各サーバ間で同期点に到達して停止したものと、同期点に未到達のまま停止したものが混在する場合があります。このような状態のまま syncterm=true の指定でデータ連動を再開すると、認識する同期点が各サーバ間で異なって最新の同期点に到達する前に抽出処理を停止してしまう可能性があります。

syncterm=true を指定した場合は、必ず Datareplicator を同期点まで実行させてください (Datareplicator だけを再起動して、全サーバを最新の同期点まで到達させてください)。

(2) 同期停止後の Datareplicator の再起動

syncterm=true を指定したときに HiRDB と同期して停止した Datareplicator を syncterm=true の指定で再起動するためには、事前に HiRDB を起動し、HiRDB Datareplicator 連携を開始しておく必要があります。HiRDB 側のデータ連動を開始しないまま、Datareplicator を再起動すると、Datareplicator は自動的に停止します。

(3) 反映側 Datareplicator との通信未接続時の動作

反映側 Datareplicator との通信が未接続の場合、syncterm=true を指定していると、HiRDB が正常停止しても抽出情報キューファイル内の更新情報を反映側 Datareplicator に送信できないため、送信プロセスは停止しません。このようなときは、反映側 Datareplicator を起動して通信できる状態にすると、送信プロセスは全更新情報を送信完了した後に自動停止します。

6.5.8 sendcontrol=sendmst を設定したときの運用上の注意

抽出システム定義で sendcontrol=sendmst を設定した場合の、Datareplicator を操作するときの注意について説明します。

(1) 反映側 Datareplicator との通信の接続/切断のタイミング

sendcontrol=sendmst を指定した場合は、送信間隔に指定した単位で、反映側 Datareplicator との通信の接続/切断が実行されます。ただし、送信間隔に到達した時点で、送信する更新情報が存在しない送信先との接続/切断は実行されません。このため、sendcontrol=nodemst を指定した場合に比べて、次に示す点で動作が異なります。

sendcontrol=sendmst を指定したとき

- 送信間隔中 (反映側 Datareplicator との通信切断中) に発生した通信回線状態については、反映側 Datareplicator は検知しません。
- 送信処理の開始要求後に一度も更新情報を送信していない状態で送信対象の反映側 Datareplicator に対して正常終了コマンド (hdsstop コマンド) を実行すると、抽出側 Datareplicator の停止を待たないで反映側 Datareplicator は停止します。

(2) 送信プロセスの再利用抑止

送信処理中に送信プロセスでメモリ不足などのエラーが発生した場合、その送信先に対する送信処理を中断するとともに、使っていた送信プロセスの再利用も抑止されます。抑止された送信プロセスを再利用するときは、抽出側 Datareplicator を再起動してください。

6.6 反映側 Datareplicator の起動と終了

反映側 Datareplicator の起動と終了の手順を次の図に示します。

図 6-16 反映側 Datareplicator の起動と終了の手順



6.6.1 反映側 Datareplicator の起動

反映側 Datareplicator の起動方法と開始モードについて説明します。

反映処理の開始方式の詳細については、「4.7.4 反映処理の開始方式の設計」を参照してください。

(1) 反映側 Datareplicator の起動方法

反映側システムで hdsstart コマンドを実行すると、反映システム定義と反映環境定義に従って反映側 Datareplicator を起動します。Windows 版 Datareplicator の場合は、コマンドプロンプトでコマンドを実行してください。

hdsstart コマンドを実行する前に、反映システム定義ファイルに反映側 HiRDB に接続するときのパスワードを指定しておいてください。指定していないと、hdsstart コマンドを実行したときに、「パスワードなし」として反映側 HiRDB に接続されます。

反映側 Datareplicator の環境を初期化するときには、hdsstart -i コマンドを実行します。特定の抽出元に対する環境だけを初期化したい場合は、hdsstart -i -D コマンドを使って、該当する抽出元の環境だけを初期化します。

詳細については、「7. コマンドの文法」の hdsstart コマンドを参照してください。

なお、Windows 版 Datareplicator の場合、次の二つの方法でも起動できます。

- 反映サービスを手動で開始する方法
- Windows の起動時に反映サービスを自動で開始する方法

反映サービスを手動で開始する方法

1. [コントロールパネル] にある [サービス] アイコンを開きます。
2. [サービス] から「HiRDB Datareplicator (Target Site)」を選びます。
3. [スタートアップパラメータ] に開始時のオプション (hdsstart コマンドのオプション) を指定します。
4. [開始] ボタンをクリックします。

Windows の起動時に反映サービスを自動で開始する方法

Windows を起動したときに、反映サービスを自動で開始する設定にする手順を次に示します。この場合、開始時のオプション (hdsstart コマンドのオプション) は指定できません。

1. [コントロールパネル] にある [サービス] アイコンを開きます。
2. [サービス] から「HiRDB Datareplicator (Target Site)」を選びます。
3. [スタートアップ] ボタンをクリックします。
4. [サービス] ダイアログボックスの [スタートアップの種類] で [自動] を選びます。
5. [OK] ボタンをクリックします。
6. [閉じる] ボタンをクリックします。

次に Windows を起動したときに、反映サービスが自動で開始します。

(2) 反映側 Datareplicator の開始モード

反映側 Datareplicator の開始モードには、次の 4 種類があります。

開始モードの詳細については、「7. コマンドの文法」の [hdsstart](#) コマンドを参照してください。

• 初期開始

前回の稼働時の動作内容を引き継がないで、反映システム定義に従って反映側 Datareplicator が開始します。初期開始のときは、反映側 Datareplicator は次に示すファイルを初期化します。

- 反映情報キューファイル
- 反映ステータスファイル
- 反映マスタステータスファイル
- 未反映情報ファイル
- 反映エラー情報ファイル
- 稼働トレースファイル

- **部分初期開始**

指定されたデータ連動識別子に対する反映システム定義に従って、指定されたデータ連動識別子に対する反映環境だけを初期化し（部分初期化）、前回の稼働時の動作内容を引き継がないで反映側 Datareplicator は処理を開始します。

部分初期開始のときは、反映側 Datareplicator は指定されたデータ連動識別子に対する次のファイルを初期化します。

- 反映情報キューファイル
- 反映ステータスファイル
- 未反映情報ファイル

指定されたデータ連動識別子以外に対する反映側 Datareplicator の処理は、正常開始時又は再開始時と同じです。

- **正常開始**

前回の稼働時の動作内容を引き継がないで、反映システム定義に従って反映側 Datareplicator が開始します。

- **再開始**

前回、反映処理がエラーで終了している場合には、抽出側 DB と反映側 DB の整合性を保証するため、前回の稼働時と同じ動作内容で反映側 Datareplicator が開始します。

6.6.2 反映側 Datareplicator の終了

反映側 Datareplicator の終了方法と終了モードについて説明します。

反映処理の停止方式の詳細については、「[4.7.5 反映処理の停止方式の設計](#)」を参照してください。

(1) 反映側 Datareplicator の終了方法

抽出側システムが終了した後に反映側システムで `hdsstop` コマンドを実行すると、反映環境定義に従って反映側 Datareplicator が終了します。Windows 版 Datareplicator の場合は、コマンドプロンプトでコマンドを実行してください。

なお、反映側 Datareplicator を終了する場合には、必ず `hdsstop` コマンドを使ってください。OS のコマンドでプロセスを消して、反映側 Datareplicator を終了した場合には、動作は保証できません。

Windows 版 Datareplicator の場合は、次の方法でも終了（即時終了）できます。

反映サービスを手動で終了する方法

1. [コントロールパネル] にある [サービス] アイコンを開きます。
2. [サービス] から「HiRDB Datareplicator (Target Site)」を選びます。
3. [停止] ボタンをクリックします。

反映側 Datareplicator の起動中に Windows を停止させると、反映側 Datareplicator は自動的に強制終了します。また、反映サービスを終了させた後、反映サービスが停止する前に強制終了のコマンドを実行しても、強制終了は実行できません。

(2) 反映側 Datareplicator の終了モード

反映側 Datareplicator の終了モードには、次の 4 種類があります。

終了モードの詳細については、「7. コマンドの文法」の `hdsstop` コマンドを参照してください。

- **正常終了**

`hdsstop` コマンドの実行後、抽出側システムの稼働単位までの反映処理を終了した時点で反映側 Datareplicator が終了します。抽出側 DB が HiRDB の場合の稼働単位は、抽出側システム全体の正常開始から正常終了までです。抽出側 DB がメインフレーム側 DB の場合の稼働単位については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

- **イベント終了**

`hdsstop` コマンドの実行後、最初に抽出側システムから送信されたイベントを反映処理が検知した時点で反映側 Datareplicator が終了します。ただし、イベントを検知する前に、抽出側システムの稼働単位までの反映が終了した場合には、イベントを待たないで反映側 Datareplicator が終了します。

- **即時終了**

`hdsstop` コマンドの実行後、現在反映処理中の更新情報の反映が終了した時点で反映側 Datareplicator が終了します。

- **強制終了**

更新情報の反映状態に関係なく、`hdsstop` コマンドの実行後、強制的に反映側 Datareplicator が終了します。同期点処理中の場合には、同期点処理が終了した後に、反映側 Datareplicator が終了します。強制終了すると反映側 DB と抽出側 DB とで不整合が生じることがあります。ただし、強制終了した場合には、次の起動時に開始モードを再開とすると、反映側 DB が自動的に回復されます。

6.7 反映側 Datareplicator の運用

反映側 Datareplicator の運用方法について説明します。

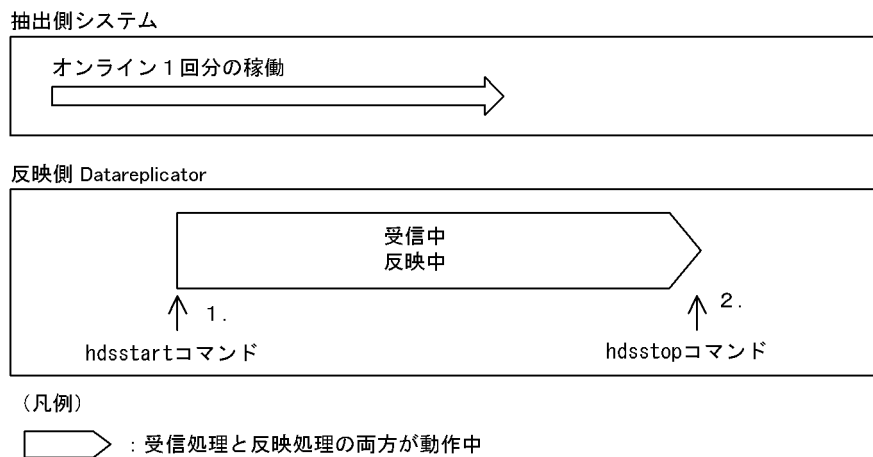
6.7.1 反映処理の運用

反映処理の運用方法について説明します。

(1) 反映側 Datareplicator を同時開始する

反映側 Datareplicator の起動時と同時に、反映処理を開始します。同時開始にすると、更新情報を受信した順に逐次、反映されます。反映側 Datareplicator を同時開始する場合の運用例を図 6-17 に、同時開始の場合の運用手順と反映側 Datareplicator の動作を表 6-9 に示します。

図 6-17 反映側 Datareplicator の同時開始の運用例



注 図中の 1, 2 は、表 6-9 の順序に対応しています。

表 6-9 運用手順と反映側 Datareplicator の動作 (同時開始)

順序	運用手順	反映側 Datareplicator の動作
1	hdsstart コマンドを実行 (初期起動時は -i オプションを付けます)。	反映側 Datareplicator が起動して、抽出側システムから送信される更新情報の受信と反映を開始します。
2	hdsstop コマンドを実行	抽出側システムのオンライン業務 (1 回分の稼働) での更新情報をすべて反映した後、反映側 Datareplicator は終了します。

定義例

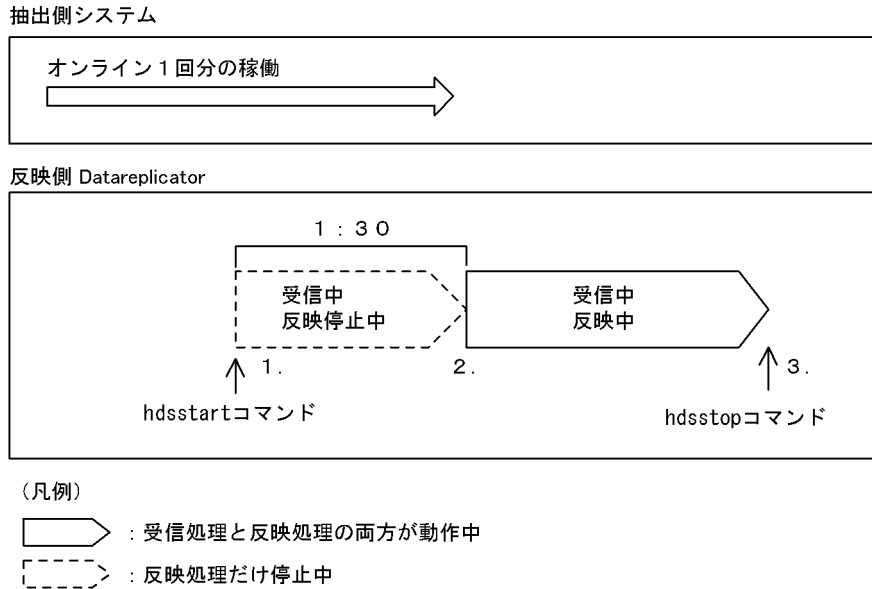
反映側 Datareplicator を同時開始する場合の反映環境定義の定義例を次に示します。

```
set startmode=trn
```


(2) 反映側 Datareplicator を遅延開始（時間指定開始方式）する

反映側 Datareplicator の起動時からユーザが指定した時間後に反映処理を開始します。反映側 Datareplicator を遅延開始（時間指定開始方式）する場合の運用例を図 6-18 に、遅延開始（時間指定開始方式）の場合の運用手順と反映側 Datareplicator の動作を表 6-10 に示します。

図 6-18 反映側 Datareplicator の遅延開始（時間指定開始方式）の運用例



注 図中の 1~3 は、表 6-10 の順序に対応しています。

表 6-10 運用手順と反映側 Datareplicator の動作（遅延開始（時間指定開始方式））

順序	運用手順	反映側 Datareplicator の動作
1	hdsstart コマンドを実行	反映側 Datareplicator が起動して、抽出側システムから送信される更新情報の受信だけを開始します。
2	—	反映側 Datareplicator の起動の、1 時間 30 分後に反映処理を自動的に開始します。
3	hdsstop コマンドを実行	抽出側システムのオンライン業務（1 回分の稼働）での更新情報をすべて反映した後、反映側 Datareplicator は終了します。

(凡例)

— : 操作は必要ありません。

定義例

反映側 Datareplicator を遅延開始（時間指定開始方式）する場合の反映環境定義の定義例を次に示します。この例では、反映側 Datareplicator が起動してから 1 時間 30 分後に、表単位反映方式で反映処理が開始されます。

```
set startmode=spd
set breaktime=01:30
set breakmode=tbl
```

(3) 反映側 Datareplicator を遅延開始（コマンド開始方式）する

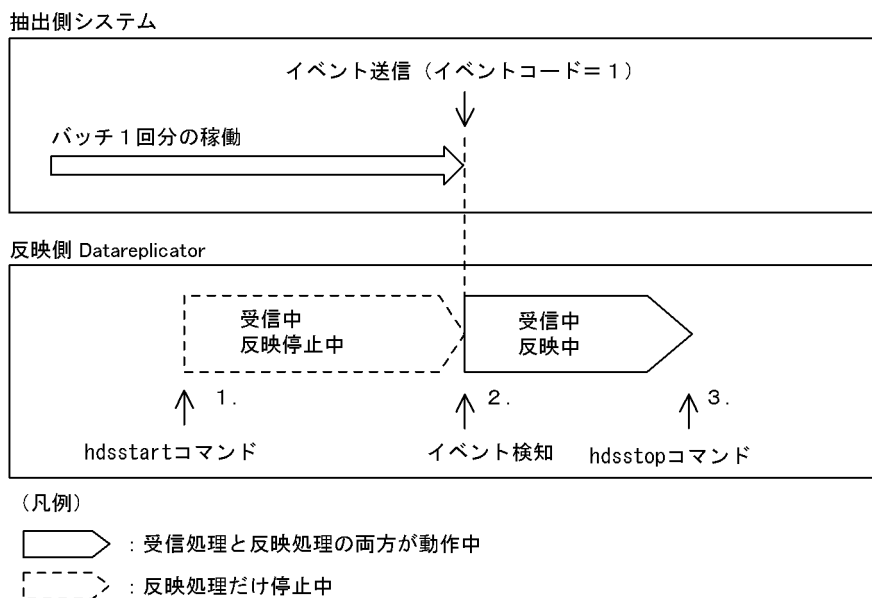
抽出側システムでコマンドを実行してから、反映処理を開始します。遅延開始（コマンド開始方式）するには次のように操作します。

1. 反映環境定義の startmode オペランドに spd を指定します。
2. 反映側システムで hdsstart コマンドを実行して、反映側 Datareplicator を起動します。反映環境定義に従って、受信処理だけが開始されます。
3. 反映側システムで hdsrftcl -m trn 又は hdsrftcl -m tbl コマンドを実行します。指定した反映方式で反映処理が開始されます。

(4) 反映側 Datareplicator を遅延開始（イベント開始方式）する

抽出側システムからのイベントを検知した時点で反映処理を開始します。反映側 Datareplicator を遅延開始（イベント開始方式）する場合の運用例を図 6-19 に、遅延開始（イベント開始方式）の場合の運用手順と反映側 Datareplicator の動作を表 6-11 に示します。

図 6-19 反映側 Datareplicator の遅延開始（イベント開始方式）反映の運用例



注 図中の 1~3 は、表 6-11 の順序に対応しています。

表 6-11 運用手順と反映側 Datareplicator の動作（遅延開始（イベント開始方式））

順序	運用手順	反映側 Datareplicator の動作
1	hdsstart コマンドを実行	反映側 Datareplicator は起動して、抽出側システムから送信される更新情報の受信だけを開始します。
2	—	反映処理再起動イベントに対応するイベントを検知した後、反映処理を開始します。
3	hdsstop コマンドを実行	抽出側システムのバッチ業務 (1 回分の稼働) での更新情報をすべて反映した後、反映側 Datareplicator は終了します。

(凡例)

－：操作は必要ありません。

定義例

反映側 Datareplicator を遅延開始（イベント開始方式）する場合の反映環境定義の定義例を次に示します。この例では、イベントコード=1 のイベントを受信した後にトランザクション単位反映方式で反映処理が開始されます。

```
set startmode=spd
set eventretrn=1
```

(5) 反映処理を停止する

受信処理を動作させたまま、反映処理だけを停止するには、次のように操作します。

1. 反映側システムで `hdsrfctl -m spd` コマンドを実行します。
2. 抽出側システムで、任意のイベントを発生させます。抽出側システムで発生させたイベントを反映側システムで検知した時点で、反映処理が停止します。抽出側 DB が HiRDB の場合の、抽出側システムでのイベントの発行については、「7. コマンドの文法」の `hdeevent` コマンド（抽出側 Datareplicator でのイベント発行）の文法を参照してください。抽出側 DB がメインフレーム側 DB の場合の、抽出側システムでのイベントの発行については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

又は反映側システムで `hdsrfctl -m immediate` コマンドを実行します。

(6) 反映処理方式を指定して停止中の反映処理を再開する

反映定義エラー、SQL エラーなどによって反映処理が停止している場合、又は起動時に受信処理だけを起動している場合には、反映処理方式を指定して反映処理だけを再開できます。反映処理方式を指定して停止中の反映処理を再開するには、次のように運用します。

1. 反映側システムで `hdsrfctl -m trn` 又は `hdsrfctl -m tbl` コマンドを実行します。
2. 抽出側システムで反映環境定義の `eventretrn` 又は `eventretbl` オペランドに対応するイベントを発生させます。抽出側システムで発生させたイベントを検知した時点で、指定した反映方式で反映処理が再開されます。

(7) 稼働中の反映処理方式を切り替える

反映環境定義に従ってデータ連動処理をしている途中に、反映処理をいったん停止し、その後、任意のタイミングで反映処理方式（トランザクション単位反映方式又は表単位反映方式）を切り替えて再開します。

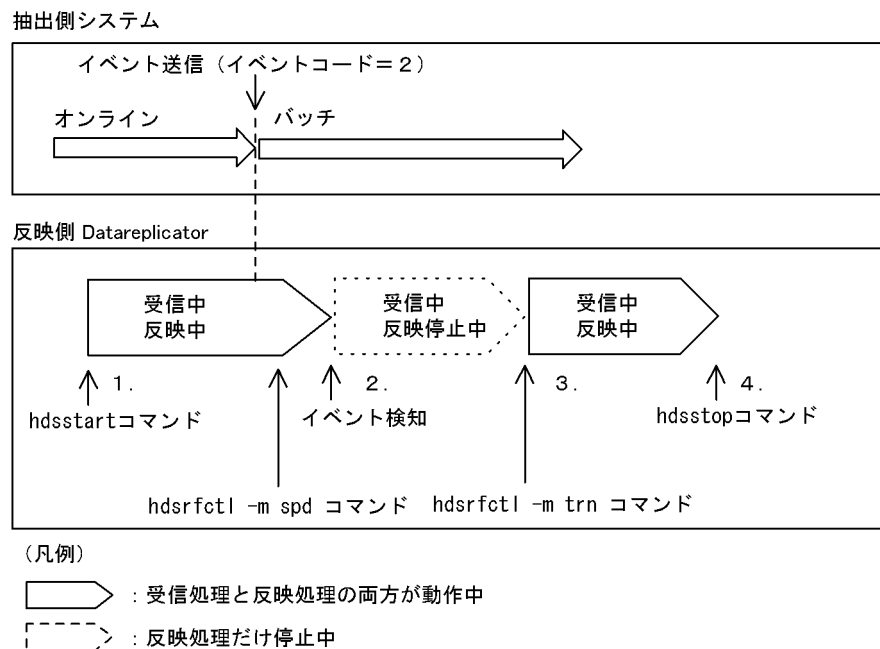
稼働中の反映処理方式を切り替えるには、次のように操作します。

1. 反映側システムで `hdsrfctl -m spd` コマンドを実行して、反映処理を停止します。
2. 反映側システムで `hdsrfctl -m trn` 又は `hdsrfctl -m tbl` コマンドを実行します。

3. 抽出側システムで、反映環境定義の eventtrn オペランド又は eventtbl オペランドに対応するイベントを発生させます。抽出側システムで発生させたイベントを反映側システムで検知した時点で、反映方式が変更されます。

反映処理方式の切り替えの運用例を図 6-20 に、反映処理方式を切り替える場合の運用手順と反映側 Datareplicator の動作を表 6-12 に示します。

図 6-20 反映処理方式の切り替えの運用例



注 図中の 1~4 は、表 6-12 の順序に対応しています。

表 6-12 運用手順と反映側 Datareplicator の動作 (反映処理方式の切り替え)

順序	運用手順	反映側 Datareplicator の動作
1	hdsstart コマンドを実行	反映側 Datareplicator は起動して、抽出側システム (オンライン業務) から送信される更新情報の受信と反映を開始します。
2	オンライン業務での更新情報の反映を終了する前に、hdsrfctl -m spd コマンドを実行 (反映環境定義では、オンライン業務での更新情報の反映が終了した後に、反映方式を表単位に変更し、反映処理を続行する予定だった)	イベント (イベントコード=2) を検知した後、反映処理だけをいったん終了します。 (hdsrfctl コマンドを実行した場合、反映環境定義の eventtbl=2 の指定よりも、コマンドの指定が有効になります)
3	バッチ業務の更新情報の受信が終了する前に、hdsrfctl -m trn を実行	コマンドを実行した時点から、バッチ業務での更新情報の反映処理を再開します。
4	hdsstop コマンドを実行	バッチ業務での更新情報の反映処理が終了した後、反映側 Datareplicator は終了します。

定義例

稼働中の反映処理方式を切り替える場合の反映環境定義の定義例を次に示します。この例では、hdsrfctl コマンドで反映処理を制御しているため、反映環境定義の eventtbl は無効になります。

```
set eventtbl=2
```

(8) 稼働中の反映処理方式を変更する

反映側 Datareplicator では、稼働中に反映処理方式を変更できます。

稼働中の反映処理方式を変更するには、次のように操作します。

1. 反映側システムで `hdsrfctl -m spd` コマンドを実行して、反映処理を停止します。
2. 反映側システムで `hdsrfctl -m trn` 又は `hdsrfctl -m tbl` コマンドを実行します。
3. 抽出側システムで、反映環境定義の `eventtrn` オペランド又は `eventtbl` オペランドに対応するイベントを発生させます。抽出側システムで発生させたイベントを反映側システムで検知した時点で、反映方式が変更されます。

(9) エラー停止中の一部の反映グループだけを再開始する

反映側 Datareplicator は、一部の反映グループだけを再開始できません。表単位反映方式で反映処理を稼働している場合に、一部の反映グループが SQL エラーなどで停止した場合には、次のように操作します。

1. 反映側システムで `hdsrfctl -m immediate` コマンド又は `hdsstop -t immediate` コマンドで、ほかで動作中の反映グループの反映処理を停止します。
2. エラーの原因を取り除いた後、反映側システムで `hdsrfctl -m tbl` コマンド又は `hdsstart` コマンドを実行します。前回のモードを引き継いだ反映方式で反映処理全体が開始されます。

(10) SAM ファイルを使う抽出側 DB からデータ連動する

抽出側システムではファイル転送プログラムを使って、更新情報を格納した SAM ファイルを反映側システムに送信します。送信された SAM ファイルは、抽出側システムのファイル転送プログラムが指定した反映側 Datareplicator のディレクトリに格納されます。

反映側 Datareplicator の操作者は、更新情報を SAM ファイルから反映情報キューファイルに格納するために、`hdssamqin` コマンドを実行します。コマンド実行によって、反映情報キューファイルに更新情報を格納していきます。反映側システムの反映処理が起動状態であれば、更新情報を反映情報キューファイルから読み込んで、反映側 DB を更新します。

6.7.2 反映側 Datareplicator で使うファイルの運用

反映側 Datareplicator で使うファイルの運用方法について説明します。

ファイルの内容については、「[3.3.2 反映処理で使うファイルとプロセス](#)」を参照してください。

ファイルの準備については、「[4.7.2 反映側 Datareplicator で使うファイルの準備](#)」を参照してください。

注意事項

Datereplicator の稼働中／停止中にかかわらず、ファイルを入れ替えないでください。
Datereplicator が誤動作するおそれがあります。

(1) 反映システム定義ファイルの運用方法

反映システム定義ファイルの運用方法については、「[5.8.2 定義内容の変更](#)」を参照してください。

(2) 反映環境定義ファイルの運用方法

反映環境定義ファイルの運用方法については、「[5.9.2 定義内容の変更](#)」を参照してください。

(3) 反映定義ファイルの運用方法

反映定義ファイルの運用方法については、「[5.10.3 定義内容の変更](#)」を参照してください。

(4) 反映情報キューファイルの運用方法

反映情報キューファイルの運用方法について説明します。

(a) ファイル名、サイズ、又は個数の変更

反映情報キューファイルの名称、サイズ、又は個数を変更する場合は、次のように操作します。

UNIX の通常ファイル、又は Windows ファイルの場合の操作

1. 抽出側システムを終了します。
2. 反映側 Datereplicator を正常終了します。
3. テキストエディタで、反映環境定義の対応するオペランドの設定を変更します。
4. 反映側 Datereplicator を初期開始又は部分初期開始します。
5. 抽出側システムを起動します。

UNIX のキャラクタ型スペシャルファイルの場合の操作

1. 抽出側システムを終了します。
2. 反映側 Datereplicator を正常終了します。
3. OS のコマンドで、キャラクタ型スペシャルファイル形式の反映情報キューファイルを再作成します。
4. テキストエディタで、反映環境定義の対応するオペランドの設定を、作成した反映情報キューファイルに合わせて変更します。
5. 反映側 Datereplicator を初期開始又は部分初期開始します。
6. 抽出側システムを起動します。

(5) 反映ステータスファイルの運用方法

反映ステータスファイルの運用方法について説明します。

(a) ファイル名又はサイズの変更

反映ステータスファイルの名称又はサイズを変更する場合には、次のように操作します。

UNIX の通常ファイル，又は Windows ファイルの場合の操作

1. 抽出側システムを終了します。
2. 反映側 Datareplicator を正常終了します。
3. テキストエディタで，反映環境定義の対応するオペランドの設定を変更します。
4. 反映側 Datareplicator を初期開始又は部分初期開始します。
5. 抽出側システムを起動します。

UNIX のキャラクタ型スペシャルファイルの場合の操作

1. 抽出側システムを終了します。
2. 反映側 Datareplicator を正常終了します。
3. OS のコマンドで，前回の反映側ステータスファイルを削除します。
4. OS のコマンドで，キャラクタ型スペシャルファイル形式の反映ステータスファイルを再作成します。
5. テキストエディタで，反映環境定義の対応するオペランドの設定を，作成した反映ステータスファイルに合わせて変更します。
6. 反映側 Datareplicator を初期開始又は部分初期開始します。
7. 抽出側システムを起動します。

(6) 反映マスタステータスファイルの運用方法

反映マスタステータスファイルの運用方法について説明します。

反映マスタステータスファイルは，初期開始時の実行結果を格納するファイルです。Datareplicator の起動時に自動的に作成されます。

(7) 反映エラー情報ファイルの運用方法

反映エラー情報ファイルの運用方法について説明します。

(a) 最大サイズの変更

反映エラー情報ファイルの最大サイズを変更する場合には，次のように操作します。

1. 抽出側システムを終了します。
2. 反映側 Datareplicator を終了します。

3. テキストエディタで、反映システム定義の対応するオペランドの設定を変更します。
4. 反映側 Datareplicator を正常開始します。
5. 抽出側システムを起動します。

(b) 反映エラー情報ファイルの保存

蓄積の対象になる反映エラー情報ファイルが満杯になるとスワップされます。スワップ先の反映エラー情報ファイルはスワップ時に再作成されます。再作成される前に反映エラー情報ファイルの内容を保存したい場合には、次のように操作してください。

1. OS のコマンドで、反映エラー情報ファイルの更新日時を調べます。
2. 更新日時の古い反映エラー情報ファイルを、OS のコマンドで任意の名称のファイルにコピーします。

Datareplicator の稼働中にエラー情報ファイルがスワップ又はクローズされると、メッセージ (KFRB00051-I, KFRB00052-I) が出力されます。このメッセージを契機に、必要に応じてバックアップを取得してください。

(c) 反映エラー情報の標準出力への出力

蓄積の対象になる反映エラー情報ファイルが満杯になるとスワップされます。スワップ先の反映エラー情報ファイルはスワップ時に再作成されます。再作成される前に反映エラー情報ファイルの内容を標準出力に出力したい場合には、次のように操作してください。

1. OS のコマンドで、反映エラー情報ファイルの更新日時を調べます。
2. 更新日時の古い反映エラー情報ファイルを、OS のコマンドで標準出力に出力します。

(d) 反映エラー情報ファイルの出力例

反映エラー情報ファイルの出力例を次の図に示します。

図 6-21 反映エラー情報ファイルの出力例

```

*****
1. Fri Mar 29 14:33:14 2002 process: hdsreflect [othergrp] (10714)
function: hds_rfc_comctl
2. errorcode: KFRB03009-I 99 c2 (hdsreflect [othergrp]) 反映プロセスは同期点処理を
完了しました。理由コード = END OF CONNECTION, ファイル情報 = [0, 5120]
3. information: SQL count = [Insert:4(4), Update:5(5), Delete:4(4), Purge:1(2),
timestamp:0(0), Commit:1(2)]
*****
Fri Mar 29 14:33:14 2002 process: hdsreflect [othergrp] (10714)
function: hds_rfc_main
errorcode: KFRB03006-I 99 c2 (hdsreflect [othergrp]) 反映プロセスがイベントを
検知しました。イベントコード = -1
information:
*****

```

(説明)

1. Fri...2002, process, function

Fri...2002：エラーが発生した曜日，月日，時分秒，及び西暦です。

process：エラーが発生した内部プロセス名とプロセス番号です。

function：エラーが発生した内部関数名です。

2. errorcode：発生したエラーのエラーコードです。

3. information：発生したエラーの詳細情報です。

(e) ファイル以外への出力

反映エラー情報ファイルの内容は，反映エラー情報ファイルの障害時の回避策，又は自動化運転の情報取得のために出力されます。

UNIX 版 Datareplicator の場合は，**syslog** ファイルにも出力されます。ただし，syslog ファイルに出力する場合には，反映システム定義の **syslogout** オペランドを **true** とする必要があります。反映エラー情報ファイルの内容の出力先を次の表に示します。

表 6-13 反映エラー情報ファイルの内容の出力先

反映エラー情報ファイルの状態	出力先	
	syslog ファイル	反映エラー情報ファイル
正常（出力可）	△	○
障害があり（出力不可）	△	×

(凡例)

○：出力します。

△：反映システム定義の **syslogout** オペランドが **true** の場合だけ出力します。

×：出力しません。

(8) 稼働トレースファイルの運用方法

稼働トレースファイルは，**hdstrcredit** コマンドで編集して参照できます。**hdstrcredit** コマンドの使い方については，「7. コマンドの文法」の **hdstrcredit** コマンドを参照してください。

(9) 未反映情報ファイルの運用方法

未反映情報ファイルの運用方法について説明します。

(a) 最大サイズの変更

未反映情報ファイルの最大サイズを変更する場合には，次のように操作します。

1. 抽出側システムを終了します。

2. 反映側 Datareplicator を終了します。

3. テキストエディタで，反映環境定義の対応するオペランドの設定を変更します。

4. 反映側 Datareplicator を初期開始又は部分初期開始します。
5. 抽出側システムを起動します。

(b) 未反映情報ファイルの保存

蓄積の対象になる未反映情報ファイルが満杯になるとスワップされます。スワップ先の未反映情報ファイルはスワップ時に再作成されます。ファイルが再作成される前に未反映情報ファイルの内容を保存したい場合には、次のように操作してください。

1. OS のコマンドで、未反映情報ファイルの更新日時を調べます。
2. 更新日時の古い未反映情報ファイルを、OS のコマンドで任意の名称のファイルにコピーします。

(c) 未反映の更新情報を標準出力に出力

蓄積の対象になる未反映情報ファイルが満杯になるとスワップされます。スワップ先の未反映情報ファイルはスワップ時に再作成されます。ファイルが再作成される前に未反映情報ファイルの内容を標準出力に出力したい場合には、次のように操作してください。

1. OS のコマンドで、未反映情報ファイルの更新日時を調べます。
2. 更新日時の古い未反映情報ファイルを、OS のコマンドで標準出力に出力します。

(d) 未反映情報ファイルの出力例

未反映情報ファイルの出力例を次の図に示します。

図 6-22 未反映情報ファイルの出力例

```
Fri Mar 29 09:29:38 2002 INSERT INTO "SHINYA"."SNDF1" ("CKEY1", "CKEY2", "CKEY3",
"CCHAR1", "CNCHAR1", "CMCHAR1", "CINT1", "CSINT1", "CDEC1", "CLDEC1", "CFLT1", "CSFLT1",
"CDATE1", "CTIME1", "CIYD1", "CIHS1", "CBLOB1", "CADT1", "CMCOL1")
VALUES ('CKEY1CKEY1', 1234567890, +19951003.,
'CHAR1 AAAA', 'NCHAR', 'MCHAR1 AAA', 305419896, 4660, 123456789012345,
12345678901234567890123456789, 1.317774742903815E-82, 1.317774742903815E-82,
'2001-12-24', '14:55:11', +12340102., +112233., **BLOB(4096)**
SGMLTEXT (**BLOB(1024)**), ARRAY[10, 20] )
```

出力形式の注意

- 列属性が抽象データ型のときは、コンストラクタ関数を使って反映するため、更新情報値の部分は次の形式で出力されます。

```
列名 = FUNC (更新情報値)
```

FUNC：コンストラクタ関数名

- 列属性が繰返し列で列指定の更新のときは、(実際の更新では ARRAY を使わないで繰返し構造で指定していますが)、表示形式として分かりやすいように ARRAY を使った次の形式で出力されます。

```
列名 = ARRAY [更新情報値, 更新情報値, ..., 更新情報値]
```

- 列属性が繰返し列で要素指定の更新のときは、次の形式で出力されます。

列名 [要素番号] = 更新情報値

- BLOB 連結演算データの更新のときは、次の形式で出力されます。

列名=列名 | | *BLOB (データバイト数) *

- BINARY 連結演算データの更新のときは、次の形式で出力されます。

列名=列名 | | *BINARY (データバイト数) *

- 時系列情報表の SQL 通番取得対象列のときは、次の形式で出力されます。

NEXT VALUE FOR "認可識別子"."順序数生成子識別子"

(10) コマンドログファイルの運用方法

コマンドログファイルの運用方法について説明します。

コマンドログファイルは、Datareplicator のコマンドを実行した履歴を格納するファイルです。Datareplicator の起動時に自動的に作成されます。コマンドを実行した履歴を確認したいときに、任意に参照してください。コマンドログに出力される情報については、「7. コマンドの文法」の「コマンドの概要」を参照してください。

コマンドログファイルの内容の例を次の図に示します。

図 6-23 コマンドログファイルの内容の例

```
Fri Aug 18 09:36:52 2006 pid=596 SYSTEM hdsstart : END : status=0, arg= -i.  
Fri Aug 18 09:37:15 2006 pid=912 administrator hdsrfctl : END : status=0, arg= -d 01 -m spd.  
Fri Aug 18 09:40:18 2006 pid=952 SYSTEM hdsstop : END : status=0, arg= -t force.
```

(11) 更新情報定義ファイルの運用方法

更新情報定義ファイルの運用方法について説明します。

(a) 定義内容の変更

定義内容を変更するときは、次のように操作してください。反映側 Datareplicator の状態に関係なく、hdssamqin コマンドを実行する前に修正できます。

1. テキストエディタで定義内容を変更します。
2. hdssamqin コマンドを実行します。

ただし、前回コマンドが正常終了しなかった場合には、定義内容は変更しないでください。もし変更した場合には、hdssamqin コマンドに -c オプションを指定して実行してください。

(12) SAM ファイルの運用方法

SAM ファイルを使うメインフレーム側で作成した SAM ファイルを、反映側 Datareplicator 又は更新情報入力コマンド (hdssamqin コマンド) を実行できるシステムにファイル転送してください。

(13) 抽出対象外データ格納ファイルの運用方法

抽出対象外データ格納ファイルの運用方法について説明します。

(a) 抽出対象外データ格納ファイルの確認

抽出対象外データ格納ファイルにデータを格納した場合は、hdssamqin コマンドの終了時にメッセージを出力します。この場合、ファイルの内容を確認してください。

(b) 抽出対象外データ格納ファイルの保存

抽出対象外データ格納ファイルは、hdssamqin コマンドを実行するたびに再作成されます。ファイルの内容を保存する場合は、次回 hdssamqin コマンドを実行する前に、OS のコマンドで任意の名称のファイルに退避してください。

(c) 抽出対象外データ格納ファイルの出力例

抽出対象外データ格納ファイルの出力例を次の図に示します。

図 6-24 抽出対象外データ格納ファイルの出力例

```
DBM name = aaa, Dataset name = bbb, SAM file offset = xx, information = Not defined
DBM name = aaa, Dataset name = bbb, SAM file offset = xx, information = Invalid data length
DBM name = aaa, Dataset name = bbb, SAM file offset = xx, information = Not defined
```

aaa : DBM名称
bbb : データセット名
xx : SAMファイル中の更新データ位置 (ファイルの先頭からのバイト位置)
Not defined : 抽出文に指定していないデータセットの更新データ情報
Invalid data length : 可変長レコードで抽出できない更新データ情報

注
抽出できないデータとは、更新データが定義又は再定義されたフィールドの途中までしかないデータのことで、

6.7.3 反映側 Datareplicator 運用時の注意

反映側 Datareplicator を運用するときの注意を次に示します。

(1) 更新についての注意

- 反映処理中の表に対して、排他制御をしないでください。反映処理の排他待ちやデッドロックが発生する場合があります。

- 複数の抽出側システムから同一の反映側システムの表を更新する場合、更新順序は保証されません。更新順序を制御する必要がある場合には、運用で実行してください。例えば、二つの抽出側システム A、B から、反映側システムの表 (T1) とをデータ連動する場合、抽出側システム A のマッピングキーの値「100」の行に対する更新の次に、抽出側システム B のキー値 100 の行に対する更新が実行されても、反映側システムのキー値 100 の行の値は抽出側システム B と同一になっていることは保証されません。
- 抽出側システムで抽出定義を変更する場合は、抽出側システムの更新情報がすべて反映側 Datareplicator に送信されたことを確認してから、実行してください。反映側 Datareplicator に送信する更新情報が残っているまま抽出定義を変更した場合、反映処理中にエラーになることがあります。この場合、反映側 Datareplicator ではエラーを回復できないことがあり、HiRDB Dataextractor などを使ったテーブル再作成が必要になることがあります。
- UOC による反映がある場合、反映処理は常に表単位反映方式で動作します。

(2) 初期開始、部分初期開始についての注意

- 特定のデータ連動識別子に対する部分だけを初期化する場合は、部分初期開始を実行してください。すべてのデータ連動識別子に対して初期化する場合には、すべてのデータ連動識別子に対する反映処理を終了させてから、全体に対して初期開始を実行してください。
- 抽出側システムを初期化した状態で起動したときは、反映側 Datareplicator も `hdsstart -i` 又は `hdsstart -i -D` で起動してください。

抽出側システムだけを初期状態で起動した場合、KFRB02003-E (詳細コード 5) でエラーになります。ただし、反映環境定義で `extract_init` に `nocheck` を指定している場合は、反映側 Datareplicator を初期開始又は部分初期開始する必要はありません。

(3) イベントについての注意

- 抽出側システムから送信されたイベントのコードが反映環境定義で定義されていない場合、反映処理の動作には何も影響ありません。ただし、反映側 Datareplicator は、イベントの発生としては認識します。
- 反映処理が動作している場合にトランザクション単位反映再起動イベント又は表単位反映再起動イベントを検知しても、反映側 Datareplicator は無視します。

(4) WITHOUT ROLLBACK オプションを指定している表についての注意

WITHOUT ROLLBACK オプションを指定している表のデータ連動には、前提条件と制限事項があります。

前提条件

次の前提条件をすべて満たす必要があります。前提条件を満たさない場合は、抽出側 DB と反映側 DB で不整合が発生することがあります。

- 抽出側 Datareplicator と反映側 Datareplicator のバージョンが 08-02 以降である
- 抽出側 DB と反映側 DB が共に HiRDB である
- 抽出システム定義の `sendcontrol` オペランドに `nodemst` を指定している

- 抽出側 DB と反映側 DB がパラレルサーバの場合は、WITHOUT ROLLBACK オプションを指定している表とそれに関連する表を、同じバックエンドサーバに格納している

制限事項

WITHOUT ROLLBACK オプションを指定している表のデータ連動には、次の制限事項があります。

- 次の表は WITHOUT ROLLBACK オプションが指定されていると、データ連動できません。
 - マージ表
 - 時系列情報表
 - サーバ間横分割表
 - イベント制御用テーブル
- データ連動できる SQL は UPDATE 文だけです。INSERT 文はデータ連動できないため、反映側 DB を初期作成するときは HiRDB Dataextractor などを使用してください。
- WITHOUT ROLLBACK オプションを指定している表では、マッピングキーに指定している列を更新できません。
- 反映時エラーのスキップはできません。SQL コードを指定してスキップしてください。
- データ連動回復機能は使用できません。個別に回復してください。

(a) 反映側 DB を参照する場合の注意

WITHOUT ROLLBACK オプションを指定している表は、1 回の更新を 1 トランザクションとして扱います。このため、WITHOUT ROLLBACK オプションを指定している表と指定していない表を同じトランザクション内で更新しても、反映側 Datareplicator には別々のトランザクションとして送信・反映されます。このとき、トランザクションがすべて反映される前に反映側 DB を参照すると、抽出側 DB と反映側 DB で不整合が発生することがあります。このため、反映処理が完了したことを確認※してから、反映側 DB を参照してください。

また、反映側 Datareplicator で障害が発生してロールバックをする場合、WITHOUT ROLLBACK オプションを指定している表では、更新情報の反映処理を実行中に、表の内容が一時的に古くなることがあります。このため、反映処理が完了したことを確認※してから、反映側 DB を参照してください。

注※

hdsstate コマンドを実行して表示される read position と write position が一致していることを確認してください。

(b) 送信環境定義 sendintvl オペランドの指定

WITHOUT ROLLBACK オプションを指定している表は、1 回の更新を 1 トランザクションとして扱うため、送信環境定義の sendintvl オペランドに 0 (トランザクション単位で更新情報を送信) を指定すると、WITHOUT ROLLBACK オプションを指定している表を更新するたびに送信処理が発生して、送信処理のオーバーヘッドが大きくなります。このため、WITHOUT ROLLBACK オプションを指定している表をデータ連動する場合は、送信環境定義の sendintvl オペランドに 0 以外を指定することをお勧めします。

(5) COMPRESSED オプションを指定している表についての注意

COMPRESSED オプションを指定している表のデータ連動には、前提条件があります。

前提条件

次の前提条件をすべて満たす必要があります。前提条件を満たさない場合は、抽出側 DB と反映側 DB で不整合が発生する、又は、反映処理で SQL 実行エラーとなることがあります。

- 抽出側 Datareplicator と反映側 Datareplicator のバージョンが 08-06 以降である
- 抽出側 DB と反映側 DB が共に HiRDB である
- 抽出側 DB と反映側 DB のエンディアンが同じである
- 抽出側と反映側の表の圧縮に関する指定が同じである

6.8 HiRDB 及び Datareplicator の構成変更

HiRDB 及び Datareplicator の構成を変更する場合の手順について説明します。

Datareplicator では、次の表に示すシステムの構成変更をする場合に、Datareplicator の初期化が必要です。初期化をしないと、Datareplicator 内で保持している更新情報や構成情報と実体が一致しなくなり、ユーザが意図する反映ができなくなるおそれがあります。

Datareplicator の初期化が必要な構成変更を次の表に示します。

表 6-14 初期化が必要な構成変更

項目	内容	記載箇所	
抽出側システムの構成変更	抽出側 HiRDB のサーバ構成変更	抽出側 HiRDB のシステム共通定義に指定する pdstart コマンドや pdunit コマンドの定義内容を変更する場合	6.8.2
	抽出側 HiRDB のシステムログの変更	<ul style="list-style-type: none"> システムログファイル数を変更する場合 システムログファイルを初期化する場合 HiRDB の pdlogchg -R コマンドでシステムログの状態を強制的に変更する場合 	
	連動対象表の定義変更	<ul style="list-style-type: none"> 連動対象表を追加 (CREATE TABLE) する場合 連動対象表を削除 (DROP TABLE) する場合 連動対象表を変更 (ALTER TABLE) する場合 連動対象表のユニークインデクスを削除する場合^{※4} 	
	抽出側 Datareplicator の定義変更 ^{※1}	<ul style="list-style-type: none"> 正常開始をして定義内容が有効になるオペランド以外^{※2}を変更する場合 二重化定義ファイルの定義内容を変更する場合 	
反映側システムの構成変更	連動対象表の定義変更	<ul style="list-style-type: none"> 連動対象表を追加 (CREATE TABLE) する場合 連動対象表を削除 (DROP TABLE) する場合 連動対象表を変更 (ALTER TABLE) する場合 	6.8.3
	反映側 Datareplicator の定義変更	<ul style="list-style-type: none"> 正常開始をして定義内容が有効になるオペランド以外^{※2}を変更する場合 二重化定義ファイルの定義内容を変更する場合 	
	反映先 HiRDB のサーバ構成変更 ^{※3}	反映先 HiRDB のシステム共通定義に指定する pdstart オペランドや pdunit オペランドの定義内容を変更する場合	

注

反映トランザクション同期機能を使用している場合の構成変更については、「[6.8.4 反映トランザクション同期機能を使用している場合の構成変更](#)」を参照してください。

注※1

hdemodq コマンドによって、抽出情報キューファイルの追加又は削除をする場合には、抽出側 Datareplicator 及び反映側 Datareplicator の初期化は必要ありません。

注※2

詳細なオペランドについては、「5. 定義」の各定義に記載されている「定義内容の変更」を参照してください。

注※3

反映定義の反映グループ定義で、キーレンジ単位分割方式、又はハッシュ分割方式の定義をしている場合に初期化が必要です。そのほかの場合は初期化する必要はありません。

注※4

抽出定義で全表抽出定義文を指定している場合で、かつ、マッピングキーとして使用しているユニークインデックスを削除する場合は該当します。マッピングキーに使用するインデックスについては「5.5.5 全表抽出定義文」を参照してください。

なお、すべてのデータが連動し終わる前に初期化・反映側 DB を再作成する必要がある場合は、「6.5.6(1) 反映側 DB の再作成時の運用手順」を参照してください。

6.8.1 連動対象表の定義変更

(1) 抽出定義で全表抽出定義文を指定している場合

抽出定義の修正は必要ありません。表の変更後、Datareplicator を初期化してください。

(2) 抽出定義で抽出定義文を指定している場合

表の変更内容と Datareplicator 定義の修正内容を次の表に示します。なお、連動対象ではない表を変更する場合は、Datareplicator の初期化、及び定義の修正は必要ありません。

表 6-15 Datareplicator の初期化が必要となる表の変更内容と定義の修正内容

表の変更内容	抽出定義の修正内容	反映定義の修正内容
表を追加する※1	extract 文を追加します。	load 文を追加します。
表を削除する	extract 文を削除します。	load 文を削除します。
表を削除した後、再度同じ表を作成する（テーブル ID の変更）	—	—
非マッピングキーの列を連動対象に追加する	extract 文の抽出する列の定義に"*"と指定している場合： 定義の修正は必要ありません。 extract 文の抽出する列の定義に列名を指定している場合： 列名を追加します。	load 文の反映対象とするフィールドの定義に"*"と指定している、又は load 文を省略している場合： 定義の修正は必要ありません。 load 文の抽出する列の定義にフィールド名称を指定している場合： フィールド名称を追加します。

表の変更内容	抽出定義の修正内容	反映定義の修正内容
マッピングキーの列を連動対象に追加する	extract 文の抽出する列の定義に"*"と指定している場合： key 句又は ukey 句に追加します。また、送信条件にする場合は、send 文の where 句に追加します。 extract 文の抽出する列の定義に列名を指定している場合： 抽出する列の定義に列名を追加し、key 句又は ukey 句にも追加します。また、送信条件にする場合は、send 文の where 句に追加します。	load 文の反映対象とするフィールドの定義に"*"と指定している、又は load 文を省略している場合： 定義の修正は必要ありません。 load 文の抽出する列の定義にフィールド名称を指定している場合： フィールド名称を追加します。
非マッピングキーの列を連動対象から削除する	extract 文の抽出する列の定義に"*"と指定している場合： 定義の修正は必要ありません。 extract 文の抽出する列の定義に列名を指定している場合： 列名を削除します。	load 文の反映対象とするフィールドの定義に"*"と指定している、又は load 文を省略している場合： 定義の修正は必要ありません。 load 文の抽出する列の定義にフィールド名称を指定している場合： フィールド名称を削除します。
マッピングキーの列を連動対象から削除する	extract 文の抽出する列の定義に"*"と指定している場合： key 句又は ukey 句から列名を削除します。また、送信条件にしている場合は、send 文の where 句から削除します。 extract 文の抽出する列の定義に列名を指定している場合： 抽出する列の定義から列名を削除し、key 句又は ukey 句からも削除します。また、送信条件にしている場合は、send 文の where 句から削除します。	load 文の反映対象とするフィールドの定義に"*"と指定している、又は load 文を省略している場合： 定義の修正は必要ありません。 load 文の抽出する列の定義にフィールド名称を指定している場合： フィールド名称を削除します。
RD エリアを追加するまたは、RD エリアを移動する	【連動対象の BES が増減する場合】 抽出環境定義の dsid オペランドを BES の増減内容に合わせて修正してください。 [注意事項] HiRDB 稼働中に連動対象の BES が増えるような RD エリアの移動を行わないでください。 移動した場合、該当する RD エリアに配置している表の更新情報は抽出しません。 【連動対象の BES に変更がない場合】 定義変更は不要です。 [注意事項] HiRDB 稼働中に連動対象の BES が増減しない RD エリアの移動を行う場合、Datareplicator の初期化は不要となります※2。	【連動対象の BES が増減する場合】 定義反映システム定義の dsid オペランドを BES の増減内容に合わせて修正してください。 更新情報を受信するデータ連動識別子に対する反映環境定義を作成します。 【連動対象の BES に変更がない場合】 定義変更は不要です。
連動対象ではない列を削除する	—	—
表名を変更する	extract 文の表識別子を変更します。	load 文の表識別子を変更します。

表の変更内容	抽出定義の修正内容	反映定義の修正内容
列名を変更する	extract 文の抽出する列の定義に"*"と指定している場合： 定義の修正は必要ありません。 extract 文の抽出する列の定義に列名を指定している場合： 列名を変更します。	—
列のデータ型（列属性）を変更する	送信条件にしている場合は、send 文の where 句の条件を変更します。	—
分割格納条件を変更する	—	—

(凡例)

—：定義を修正する必要はありません。

注※1

抽出対象表を追加する場合、抽出システム定義の extinforum オペランドの指定値を見直してください。extinforum オペランドに、extract 文の数よりも少ない値を指定していると、hdeprep コマンドの実行時にエラーになります。

注※2

HiRDB 稼働中に連動対象の BES が増減しない RD エリアの移動を行う場合、以下の手順で RD エリアを移動することで Datareplicator の初期化は不要となります。

1. 移動対象の RD エリアを閉塞する。
2. hdeevent コマンドを実行する。指定するイベントコードは、以下のオペランドに指定している以外のコードを指定する。

送信環境定義

- eventcntreset
- eventsync

反映環境定義

- eventtrn
- eventtbl
- eventretrn
- eventretbl
- eventspd
- eventcntreset

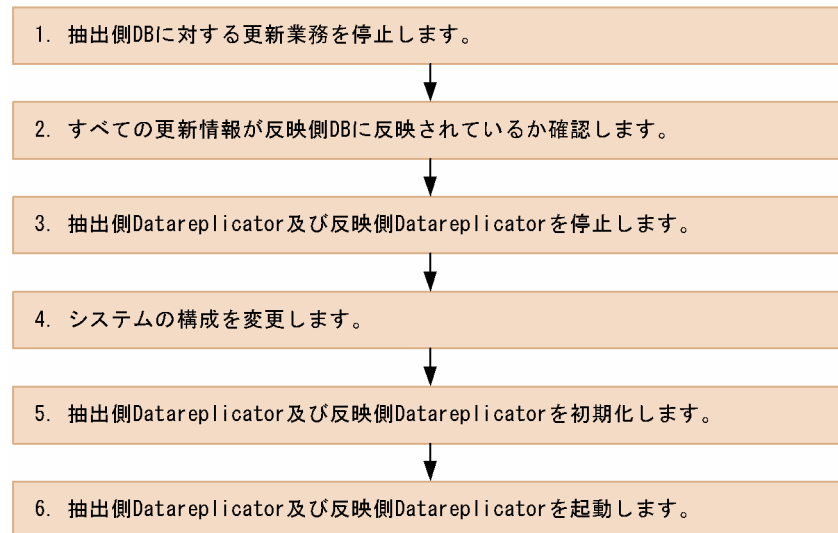
3. 手順 2 実行後、Datareplicator のエラー情報ファイルに KFRB02036-I が出力され、イベントが反映側に送信完了したことを確認する。
4. RD エリアを移動する。
5. 移動対象の RD エリアの閉塞を解除する。

6.8.2 抽出側システムの構成変更

抽出側システムの構成を変更する場合の初期化手順について説明します。なお、抽出側システムと反映側システムの両方の構成を変更する場合も、この項で説明している手順で初期化してください。

手順の流れを次の図に示します。

図 6-25 抽出側システムの構成変更の手順



注

手順 5 以降を実施するには、抽出側 DB 及び反映側 DB を起動してください。

抽出側システムの構成変更の手順を図に基づいて説明します。説明の番号は図中の番号に対応しています。

1. 抽出側 DB に対する更新業務を停止します。
2. すべての更新情報が反映側 DB に反映されているか確認します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplicator のすべてのプロセスが稼働していることを確認します。 停止しているプロセスがあれば、再起動します。	hdestate	表示結果の「Status」が次の状態になっていないことを確認してください※1。 <ul style="list-style-type: none">• init• hold• not active• not active (error)
2	反映側 Datareplicator のすべてのプロセスが稼働していることを確認します。 停止しているプロセスがあれば、再起動します。	hdsstate	表示結果の「PID」が-1 になっていないことを確認してください※2。
3	すべてのシステムログの内容をシステムログファイルに出力します。	pdlogsync -d sys※3	syslog ファイル (Windows の場合はイベントログ) に KFPS02183-I メッセージが出力されていることを確認してください。

項番	作業内容	実行コマンド	確認項目
4	すべての更新情報が抽出されたか確認します。	pdls -d rpl -j	表示結果の内容を確認してください※4。
5	すべての更新情報が送信されたか確認します。	hdestate	表示結果の「current used ratio」が0%になっていることを確認してください。又は、表示結果の「Queue write position」と「Queue read position」が同じ位置を示していることを確認してください。
6	すべての更新情報が反映されたか確認します。	hdsstate	表示結果の「current used ratio」が0%になっていることを確認してください。又は、表示結果の「Queue write position」と「Queue read position」が同じ位置を示していることを確認してください。

注※1

抽出対象外の BES については、「Status」に「not active」と表示されます。

注※2

抽出システム定義の sendcontrol オペランドに sendmst を指定している場合は、<COMMUNICATION INFORMATION>の「PID:hdstcpmst」に-1 が表示されることがあります。

注※3

HiRDB/パラレルサーバの場合は、-s オプションにサーバ名を指定してください。BES が複数ある場合は、抽出対象表が存在するすべての BES についてコマンドを実行してください。

注※4

次の例を参考に、表示結果の(2)より(1)の方が大きいことを確認してください。

```

SYSTEMID      : HRD1(183346)
Data replication : Y
UNITID       : unt1(183346)
Data replication : Y
SERVER NAME  : sds01
Extract Database : Y
Extract Status : C
System Log Extract Point :
Run ID  Group  Gen No.  BLock No.
4740e7a9 log24  18      2dd8      ... (1)
System Log Sync Info :
Run ID  Group  Gen No.  BLock No.
4740e7a9 log24  18      2dba      ... (2)

```

3. 抽出側 Datareplicator 及び反映側 Datareplicator を停止します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplicator を停止します。	hdestop	—
2	抽出側 Datareplicator の停止を確認します。	hdestate	標準エラー出力に KFRB04411-E メッセージが出力されていることを確認してください。
3	反映側 Datareplicator を停止します。	hdsstop	—

項番	作業内容	実行コマンド	確認項目
4	反映側 Datareplicator の停止を確認します。	hdsstate	標準エラー出力に KFRB04302-E メッセージが出力されていることを確認してください。

(凡例)

－：特にありません。

4. システムの構成を変更します。

5. 抽出側 Datareplicator 及び反映側 Datareplicator を初期化します。

項番	作業内容	実行コマンド	確認項目
1	HiRDB とのデータ連携を中止します。	pdrplstop -f	標準出力に KFPS05141-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を初期化します。	hdestart -i (応答メッセージに Y を入力します)	msterrfile1 又は msterrfile2 に KFRB00504-I メッセージが出力されていることを確認してください。
3	抽出定義プリプロセスファイルを作成します。	hdeprep -f 抽出定義ファイル*	標準出力に KFRB04500-I メッセージが出力されていることを確認してください。
4	反映側 Datareplicator を初期化します。	hdsstart -i -q*	標準出力 (Windows の場合はイベントビューア) に KFRB04216-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

6. 抽出側 Datareplicator 及び反映側 Datareplicator を起動します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を起動します。	hdsstart*	errfile1 又は errfile2 に KFRB00100-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を起動します。	hdestart*	msterrfile1 又は msterrfile2 に KFRB00502-I メッセージが出力されていることを確認してください。
3	HiRDB とのデータ連携を開始します。	pdrplstart	標準出力に KFPS05140-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

6.8.3 反映側システムの構成変更

反映側システムの構成を変更する場合の初期化手順について説明します。なお、抽出側システムと反映側システムの両方の構成を変更する場合の初期化手順は、「6.8.2 抽出側システムの構成変更」を参照してください。

手順の流れを次の図に示します。

図 6-26 反映側システムの構成変更の手順



注

手順 5 以降を実施するには、抽出側 DB 及び反映側 DB を起動してください。

反映側システムの構成変更の手順を図に基づいて説明します。説明の番号は図中の番号に対応しています。

1. 抽出側 Datareplicator を停止します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplicator を停止します。	hdestop	—
2	抽出側 Datareplicator の停止を確認します。	hdestate	標準エラー出力に KFRB04411-E メッセージが出力されていることを確認してください。

(凡例)

—：特にありません。

2. 反映側 Datareplicator 中の更新情報がすべて反映されているか確認します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator のすべてのプロセスが稼働していることを確認します。 停止しているプロセスがあれば、再起動します。	hdsstate	hdstcpmst の「PID」には"-1"が表示され、ほかの「PID」が-1 になっていないことを確認してください。
2	すべての更新情報が反映されたか確認します。	hdsstate	表示結果の「current used ratio」が0%になっていることを確認してください。又は、表示結果の「Queue write position」と「Queue read position」が同じ位置を示していることを確認してください。

3. 反映側 Datareplicator を停止します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を停止します。	hdsstop	—
2	反映側 Datareplicator の停止を確認します。	hdsstate	標準エラー出力に KFRB04302-E メッセージが出力されていることを確認してください。

(凡例)

—：特にありません。

4. システムの構成を変更します。

5. 反映側 Datareplicator を初期化します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を初期化します。	hdsstart -i -q*	標準出力（Windows の場合はイベントビューア）に KFRB04216-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

6. 抽出側 Datareplicator 及び反映側 Datareplicator を起動します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を起動します。	hdsstart*	errfile1 又は errfile2 に KFRB00100-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を起動します。	hdestart*	msterrfile1 又は msterrfile2 に KFRB00502-I メッセージが出力されていることを確認してください。

注※

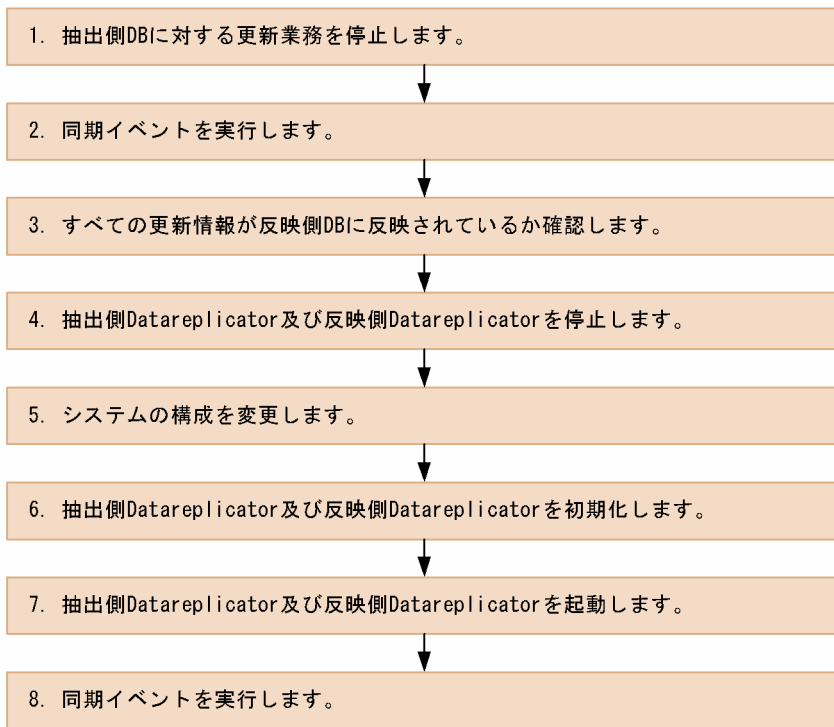
必要に応じてオプションを付けてください。

6.8.4 反映トランザクション同期機能を使用している場合の構成変更

反映トランザクション同期機能を使用している場合の初期化手順について説明します。

手順の流れを次の図に示します。

図 6-27 反映トランザクション同期機能を使用している場合の構成変更の手順



注

手順 6 以降を実施するには、抽出側 DB 及び反映側 DB を起動してください。

反映トランザクション同期機能を使用している場合の構成変更の手順を図に基づいて説明します。説明の番号は図中の番号に対応しています。

1. 抽出側 DB に対する更新業務を停止します。
2. 同期イベントを実行します。

項番	作業内容	実行コマンド	確認項目
1	同期を取って反映を終了させるために、同期イベントを実行します。	hdeevent -n イベントコード※	標準エラー出力にメッセージが出力されていないことを確認してください。

注※

必要に応じてオプションを付けてください。

3. すべての更新情報が反映側 DB に反映されているか確認します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplicator のすべてのプロセスが稼働していることを確認します。 停止しているプロセスがあれば、再起動します。	hdestate	表示結果の「Status」が次の状態になっていないことを確認してください※1。 <ul style="list-style-type: none"> • init • hold • not active • not active (error)

項番	作業内容	実行コマンド	確認項目
2	反映側 Datareplicator のすべてのプロセスが稼働していることを確認します。 停止しているプロセスがあれば、再起動します。	hdsstate	表示結果の「PID」が-1 になっていないことを確認してください※2。
3	すべてのシステムログの内容をシステムログファイルに出力します。	pdlogsync -d sys※3	syslog ファイル（Windows の場合はイベントログ）に KFPS02183-I メッセージが出力されていることを確認してください。
4	すべての更新情報が抽出されたか確認します。	pdls -d rpl -j	表示結果の内容を確認してください※4。
5	すべての更新情報が送信されたか確認します。	hdestate	表示結果の「current used ratio」が0%になっていることを確認してください。又は、表示結果の「Queue write position」と「Queue read position」が同じ位置を示していることを確認してください。
6	すべての更新情報が反映されたか確認します。	hdsstate	表示結果の「current used ratio」が0%になっていることを確認してください。又は、表示結果の「Queue write position」と「Queue read position」が同じ位置を示していることを確認してください。

注※1

抽出対象外の BES については、「Status」に「not active」と表示されます。

注※2

抽出システム定義の sendcontrol オペランドに sendmst を指定している場合は、<COMMUNICATION INFORMATION>の「PID:hdstcpmst」に-1 が表示されることがあります。

注※3

HiRDB/パラレルサーバの場合は、-s オプションにサーバ名を指定してください。BES が複数ある場合は、抽出対象表が存在するすべての BES についてコマンドを実行してください。

注※4

次の例を参考に、表示結果の(2)より(1)の方が大きいことを確認してください。

```

SYSTEMID      : HRD1(183346)
Data replication : Y
UNITID       : unt1(183346)
Data replication : Y
SERVER NAME   : sds01
Extract Database : Y
Extract Status : C
System Log Extract Point :
Run ID  Group  Gen No.  BLock No.
4740e7a9 log24  18       2dd8      ... (1)
System Log Sync Info :
Run ID  Group  Gen No.  BLock No.
4740e7a9 log24  18       2dba      ... (2)

```

4. 抽出側 Datareplicator 及び反映側 Datareplicator を停止します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplicator を停止します。	hdestop	—
2	抽出側 Datareplicator の停止を確認します。	hdestate	標準エラー出力に KFRB04411-E メッセージが出力されていることを確認してください。
3	反映側 Datareplicator を停止します。	hdsstop	—
4	反映側 Datareplicator の停止を確認します。	hdsstate	標準エラー出力に KFRB04302-E メッセージが出力されていることを確認してください。

(凡例)

—：特にありません。

5. システムの構成を変更します。

6. 抽出側 Datareplicator 及び反映側 Datareplicator を初期化します。

項番	作業内容	実行コマンド	確認項目
1	HiRDB とのデータ連携を中止します。	pdrplstop -f	標準出力に KFPS05141-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を初期化します。	hdestart -i (応答メッセージに Y を入力します)	msterrfile1 又は msterrfile2 に KFRB00504-I メッセージが出力されていることを確認してください。
3	抽出定義プリプロセスファイルを作成します。	hdeprep -f 抽出定義 ファイル*	標準出力に KFRB04500-I メッセージが出力されていることを確認してください。
4	反映側 Datareplicator を初期化します。	hdsstart -i -q*	標準出力 (Windows の場合はイベントビューア) に KFRB04216-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

7. 抽出側 Datareplicator 及び反映側 Datareplicator を起動します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を起動します。	hdsstart*	errfile1 又は errfile2 に KFRB00100-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を起動します。	hdestart*	msterrfile1 又は msterrfile2 に KFRB00502-I メッセージが出力されていることを確認してください。
3	HiRDB とのデータ連携を開始します。	pdrplstart	標準出力に KFPS05140-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

8. 同期イベントを実行します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator の同期反映グループの全プロセスを稼働状態にするために、同期イベントを実行します。	hdevent -n イベントコード※	<ul style="list-style-type: none"> 標準エラー出力にメッセージが出力されていないことを確認してください。 反映側システムのエラー情報ファイルに KFRB03009-I メッセージが出力され、「reason」が「SYNC EVENT」であることを確認してください。

注※

必要に応じてオプションを付けてください。

6.9 系切り替え機能を使うときの運用

Datareplicator では HiRDB の連動切り替えを使用して、系切り替えできます。これは、HiRDB の系が切り替わったときに Datareplicator の系も切り替えるものです。Datareplicator で障害が発生したときに、Datareplicator を単独で系切り替えすることはできません。なお、HiRDB での系切り替え機能の運用については、マニュアル「HiRDB システム運用ガイド」を参照してください。

系切り替え機能は、次の表に示すクラスタソフトウェアで使用できます。

表 6-16 サポートしているクラスタソフトウェア

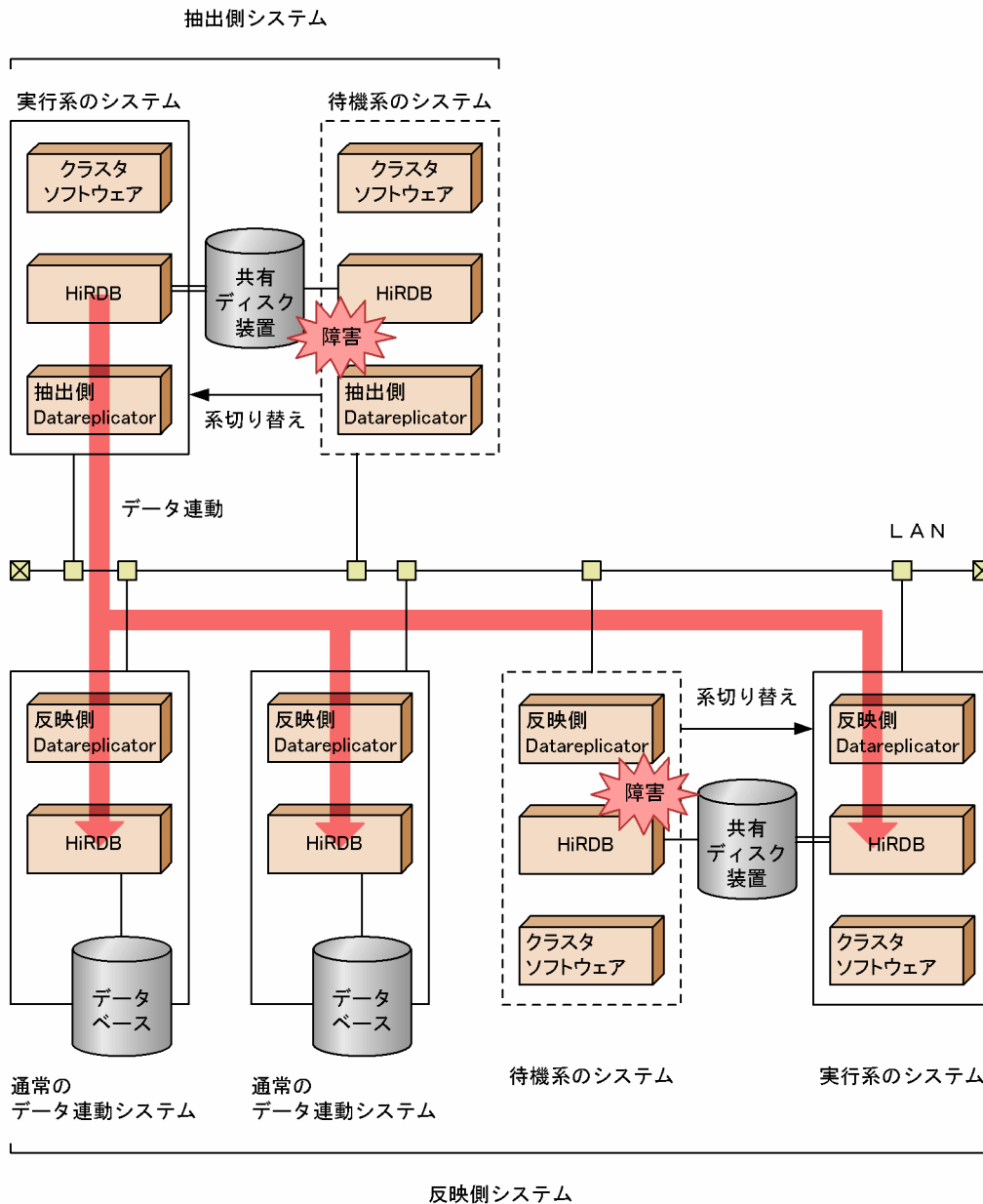
クラスタソフトウェア	対象 OS					
	HP-UX	Solaris	AIX	Linux	Linux (Amazon EC2)	Windows
HA モニタ	○	×	○	○	○	×
MC/ServiceGuard	○	×	×	×	×	×
HACMP	×	×	○	×	×	×
Microsoft Cluster Server	×	×	×	×	×	○

(凡例)

- ：使用できます。
- ×

系切り替え機能を使ったデータ連動システムの形態を次の図に示します。

図 6-28 系切り替え機能を使ったデータ連動システムの形態



6.9.1 系切り替え機能の形態

系切り替え機能の形態について説明します。

(1) 系切り替えの種類

系切り替えをする場合、抽出側 Datareplicator は必ず HiRDB サーバと連動系切り替え（複数の製品をグループ化して、一括して切り替えるようにした系切り替え形態）をしてください。反映側は必要に応じて連動系切り替えをしてください。

HiRDB の系切り替え機能の形態を次に示します。なお、それぞれの系切り替え機能の形態については、関連するクラスタソフトウェアのマニュアルを参照してください。

自動系切り替え

実行系に障害が起こったときに、自動的に系が切り替わる形態です。

計画系切り替え

実行系でクラスタソフトウェアのコマンドを実行して、意図的に系を切り替える形態です。

上記のどちらの形態でも連動系切り替えができます。

ほかに、HiRDB の系切り替えの時間短縮のための機能として、高速系切り替え、ユーザーバホットスタンバイ、及びトランザクションキューイングがあります。

(2) システムの構成

系切り替え機能には、次に示すシステム構成があります。

1：1 系切り替え構成

実行系と待機系が 1：1 に対応している構成です。

2：1 系切り替え構成

実行系と待機系が 2：1 に対応している構成です。

相互系切り替え構成

実行系として動作しながら、同じサーバマシンに互いの待機系を持つ構成です。

上記のシステム形態のうち相互系切り替えで運用するときは、エラー情報ファイル名を固有にするための定義が必要です。

系切り替えの構成ごとによる説明の参照先を次の表に示します。

表 6-17 系切り替えの構成ごとによる説明の参照先

使用するクラスタソフトウェア	系切り替えの構成	参照先
HA モニタ	1:1 系切り替え構成 2:1 系切り替え構成	6.9.2 系切り替え機能を使うときの準備 (HA モニタの場合) 6.9.4 系切り替え機能を使うときの運用手順 (HA モニタの場合)
	相互系切り替え構成	6.9.2 系切り替え機能を使うときの準備 (HA モニタの場合) 6.9.4 系切り替え機能を使うときの運用手順 (HA モニタの場合) 6.9.5 系切り替え機能を使うときの注意
	影響分散スタンバイレス型系切り替え構成	6.9.4 系切り替え機能を使うときの運用手順 (HA モニタの場合)

使用するクラスタソフトウェア	系切り替えの構成	参照先
		6.9.6 影響分散スタンバイレス型系切り替え機能での運用
Microsoft Cluster Server	1:1 系切り替え構成	6.9.3 系切り替え機能を使うときの準備 (Microsoft Cluster Server の場合)

6.9.2 系切り替え機能を使うときの準備 (HA モニタの場合)

HA モニタによる系切り替え機能を使うときに準備する項目を次に示します。ここで説明している項目は、「6.9.1(1) 系切り替えの種類」で説明した系切り替えすべてで有効です。

HA モニタについては、マニュアル「高信頼化システム監視機能 HA モニタ」を参照してください。

(1) HA モニタの準備

HA モニタの server 定義文で、Datareplicator の動作環境を設定します。Datareplicator は、HiRDB と連動して系切り替えを行います。server 定義文の指定例を次に示します。

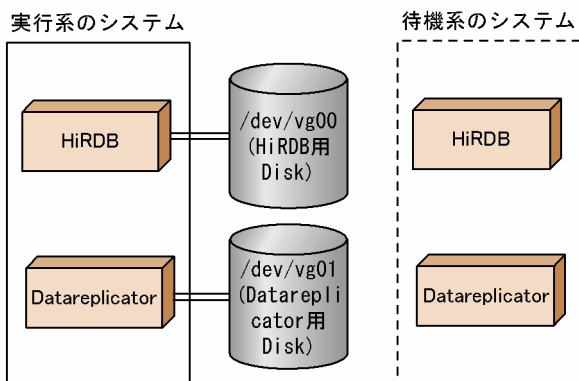
HA モニタのリソースサーバ機能を使用する場合の定義例

```

/**** Resource   ***/
resource alias   REPres
                group      groupA
                initial    online
                disk       /dev/vg00:/dev/vg01
                lan_updown use
;

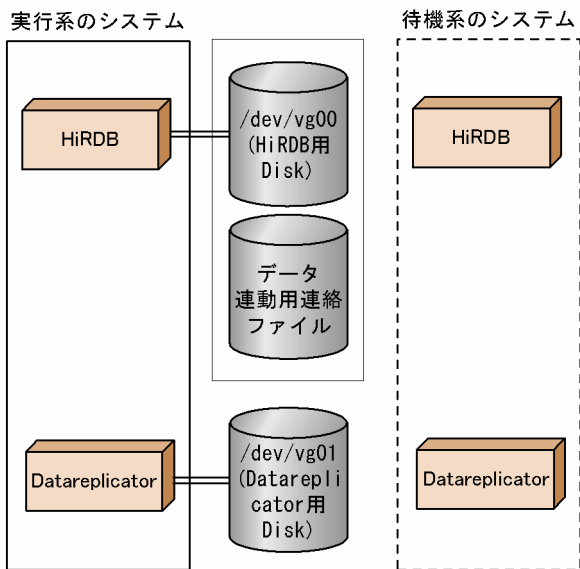
/**** Replicator ***/
server name     /HAMon/etc/replica.up
        alias    repl
        acttype  monitor
        initial  online
        termcommand /HAMon/etc/replica.down
        lan_updown nouse
        parent   REPres
        group    groupA
;

```



HA モニタのリソースサーバ機能を使用しない場合の定義例

```
/***/ Replicator  ***/
server  name      /HAmon/etc/replica.up  ,
        alias     repl      ,
        acttype   monitor   ,
        initial   online    ,
        termcommand /HAmon/etc/replica.down ,
        disk      /dev/vg01  ,
        lan_updown use      ,
        group     groupA    ;
```



抽出側 Datareplicator の系切り替えでは、データ連動用連絡ファイルを HiRDB 本体と共用しています。そのため、データ連動用連絡ファイルのボリューム配置や、HiRDB 本体と Datareplicator との起動タイミングを調整する必要があります。

HA モニタがリソースサーバ機能をサポートしているバージョンである場合、抽出側 Datareplicator の系切り替えでは、HA モニタのリソースサーバ機能を使用することで、起動タイミングの調整が不要になります。系切り替え環境の構築を簡略化できる、リソースサーバ機能を使用することをお勧めします。

次に、Datareplicator に関する server 定義文の各オペランドの指定について説明します。詳細はマニュアル「高信頼化システム監視機能 HA モニタ」を参照してください。

(a) リソースサーバの定義

リソースサーバの定義に関する各オペランドの指定について説明します。

alias

HA モニタで使うコマンドや、出力するメッセージのためのリソースサーバの識別名を指定します。

group

HiRDB 本体及び Datareplicator と連動系切り替えをするように、HiRDB 本体及び Datareplicator と同じサーバグループ名を指定します。

initial

サーバ起動時の状態を指定します。現用系では online を、予備系では standby を指定します。

disk

HiRDB 本体及び Datareplicator について、切り替えが必要なファイルを格納した共有ディスク装置の、ボリュームグループのパス名を指定します。

共有ディスク装置に作成するファイルについては、「表 6-18 系切り替え機能を使うときのファイルの準備」を参照してください。

lan_updown

use を指定します。これは、系切り替え後、抽出側 Datareplicator に抽出マスタプロセスと抽出ノードマスタプロセスとの通信を行うために IP アドレスの切り替えが必要となるためです。

また、上記の IP アドレス切り替えを行うための LAN の状態設定ファイル（サーバ識別名.up ファイル、及びサーバ識別名.down ファイル）を作成する必要があります。

(b) Datareplicator に関する定義

Datareplicator の定義に関する各オペランドの指定について説明します。

name

Datareplicator を開始するコマンドを記述したシェルを絶対パス名で指定します。シェルの記述例を次に示します。

```
sleep 5※ 1.  
su - hirtbdba -c /opt/hirdbds/bin/hdestart 2> /tmp/start_log 2.
```

注※

抽出側 Datareplicator 又は反映側 Datareplicator でリソースサーバ機能を使用する場合は、sleep は必要ありません。

1. HiRDB 本体と Datareplicator で共用するデータ連動用連絡ファイルのディスクの切り替えは、HiRDB 本体の server 定義文で指定します。そのため、HiRDB 本体の切り替えが完了してデータ連動用連絡ファイルを参照できるまでのタイムラグを考慮して、Datareplicator を起動する前に 5～10 秒程度の待ち時間を設定します。

2. シェルは HA モニタからスーパーユーザで実行されますが、Datareplicator の起動コマンドは Datareplicator 管理者で実行する必要があるため、su コマンドを使って Datareplicator 管理者（上記の例では hirtbdba）で Datareplicator を起動します。

起動コマンドの指定は、抽出側 Datareplicator と反映側 Datareplicator で異なります。

抽出側 Datareplicator の場合：/opt/hirdbds/bin/hdestart

反映側 Datareplicator の場合：/opt/hirdbds/bin/hdsstart

なお、起動コマンド実行時のエラーメッセージは、標準エラー出力に対して出力されます。シェル内で起動コマンドを実行する場合は、標準エラー出力をワークファイル（上記例では/tmp/start_log）に出力するようにしてください。

alias

HA モニタで使うコマンドや、出力するメッセージのための識別名を指定します。

acttype

monitor を指定します。

initial

サーバ起動時の状態を指定します。現用系では online を、予備系では standby を指定します。

termcommand

HA モニタの monend コマンドだけでシステムを終了させたり、monswap コマンドだけでシステムを計画切り替えしたりするときに、Datareplicator を停止するコマンドを記述したシェルを絶対パス名で指定します。

シェルの記述例を次に示します。

```
1. su - hirbdbba -c "/opt/hirbdbds/bin/hdestop -t sendterm -w" 2> /tmp/stop_log
```

1. Datareplicator が完全に停止するのを確認する必要があるため、hdestop コマンドを-w で実行します。シェルは HA モニタからスーパーユーザで実行されますが、Datareplicator の停止コマンドは Datareplicator 管理者で実行する必要があるため、su コマンドを使って Datareplicator 管理者（上記の例では hirbdbba）で Datareplicator を停止します。

停止コマンドの指定は、抽出側 Datareplicator と反映側 Datareplicator で異なります。

抽出側 Datareplicator の場合：/opt/hirbdbds/bin/hdestop -t sendterm -w[※]

反映側 Datareplicator の場合：/opt/hirbdbds/bin/hdsstop -t force -w[※]

注※ -t sendterm 及び-t force オプションについては、計画系切り替えの場合に-t immediate を指定するなど、目的に応じてオプションの指定を変更してください。

なお、停止コマンド実行時のエラーメッセージは、標準エラー出力に対して出力されます。シェル内で停止コマンドを実行する場合は、標準エラー出力をワークファイル（上記例では/tmp/stop_log）に出力するようにしてください。

disk

共有ディスク装置に作成するファイルを指定します。共有ディスク装置に作成するファイルについては、「表 6-18 系切り替え機能を使うときのファイルの準備」を参照してください。

リソースサーバ機能を使用しない場合に、切り替えが必要なファイルを格納した共有ディスク装置のボリュームグループのパス名を指定します。ただし、データ連動用連絡ファイルを格納するディスクは HiRDB 本体からもアクセスされるため、HiRDB 本体に関する server 定義文の disk オペランドに指定してください。

リソースサーバ機能を使用する場合、リソースサーバの定義で指定しているので、指定する必要はありません。

lan_updown

抽出側と反映側のシステムで指定は同じです。

抽出側システム

use を指定します。これは、系切り替え後、抽出マスタープロセスと抽出ノードマスタープロセスとの通信が行えるように IP アドレスの切り替えが必要となるためです。

また、上記の IP アドレス切り替えを行うための LAN の状態設定ファイル（サーバ識別名.up ファイル、及びサーバ識別名.down ファイル）を作成する必要があります。LAN の状態設定ファイルで、IP アドレスの切り替え制御を行うホスト名を Datareplicator の抽出システム定義の node_host オペランドに指定してください。

反映側システム

use を指定します。これは、系切り替え後、抽出側 Datareplicator から再接続できるように IP アドレスの切り替えが必要となるためです。

また、上記の IP アドレス切り替えを行うための LAN の状態設定ファイル（サーバ識別名.up ファイル、及びサーバ識別名.down ファイル）を作成する必要があります。

parent

リソースサーバの識別名（alias オペランドに指定した値）を指定します。

group

HiRDB 本体と連動系切り替えをするように HiRDB 本体と同じサーバグループ名を指定します。

Datareplicator はモニタモードのため、切り替え種別の指定は必要ありません。

(2) ハードウェアの準備

系切り替え機能を使うためには、Datareplicator のファイルを共有ディスク装置に割り当てる必要があります。割り当てが必要な Datareplicator のファイルについては、「表 6-18 系切り替え機能を使うときのファイルの準備」を参照してください。

「表 6-18 系切り替え機能を使うときのファイルの準備」で示したファイルを現用系と予備系とで共有する外付けハードディスク（キャラクタ型スペシャルファイル）に作成し、互いの Datareplicator から同じパスでこれらのファイルを参照できるように設定してください。なお、共有ディスク装置として、Datareplicator ファイルシステム領域を使えます。

(3) ネットワークの準備

次のネットワーク構成は、IP アドレスを引き継ぐ構成にしてください。

- 抽出側 Datareplicator の抽出マスタープロセスと抽出ノードマスタープロセス間
- 抽出側 Datareplicator と反映側 Datareplicator 間

(4) ファイルの準備

系切り替え機能を使うためには、共有ディスク装置に割り当てなければならないファイルや、現用系と予備系とで同じディレクトリ名下に複製しておかなければならないファイルがあります。

共有ディスク装置に割り当てるファイルは、現用系及び予備系の HiRDB Datareplicator から同じパス名で参照できるように、シンボリックリンクを作成しておく必要があります。シンボリックリンクの作成方法については、「4.6.2 抽出側 Datareplicator で使うファイルの準備」及び「4.7.2 反映側 Datareplicator で使うファイルの準備」を参照してください。

系切り替え機能を使うときのファイルの準備を次の表に示します。

表 6-18 系切り替え機能を使うときのファイルの準備

区分	ファイル名	共有ディスク装置	同一ディレクトリ名
抽出側 Datareplicator	抽出システム定義ファイル	—	同一
	抽出環境定義ファイル	—	同一
	送信環境定義ファイル	—	同一
	抽出定義ファイル※	—	—
	二重化定義ファイル	—	同一
	二重化制御ファイル	—	同一
	抽出定義プリプロセスファイル	D	—
	抽出情報キューファイル	D	—
	抽出マスタステータスファイル	D	—
	抽出サーバステータスファイル	D	—
	抽出マスタエラー情報ファイル	—	—
	抽出ノードマスタエラー情報ファイル	—	—
	稼働トレースファイル	—	—
	データ連動用連絡ファイル	H	—
コマンドログファイル	—	—	
反映側 Datareplicator	反映システム定義ファイル	—	同一
	反映環境定義ファイル	—	同一
	反映定義ファイル	—	同一
	二重化定義ファイル	—	同一
	二重化制御ファイル	—	同一
	反映情報キューファイル	D	—
	反映ステータスファイル	D	—
	反映マスタステータスファイル	D	—
	反映エラー情報ファイル	—	—
	稼働トレースファイル	—	—

区分	ファイル名	共有ディスク装置	同一ディレクトリ名
	未反映情報ファイル	—	—
	コマンドログファイル	—	—
	SAM ファイル	—	同一
	更新情報定義ファイル	—	同一
	抽出対象外データ格納ファイル	—	同一
	ユーザ作成の文字コード変換用マッピングテーブル	—	同一
	回復情報ファイル	D	—

(凡例)

D :

共有ディスク装置に割り当てる必要があります。

HiRDB Datareplicator の系切り替えと連動して切り替えることができる共有ディスク装置上のボリューム・グループに割り当てる必要があります。切り替えタイミングが異なるため、H で示されるファイルとは異なるボリューム・グループに割り当てる必要があります。

また、割り当てたボリューム・グループは、HiRDB Datareplicator の系切り替えと連動して切り替えられるように HA モニタのサーバ対応環境設定で設定しておく必要があります。

H :

HiRDB の系切り替えと連動して切り替えることができる共有ディスク装置上のボリューム・グループに割り当てる必要があります。

切り替えタイミングが異なるため、D で示されるファイルとは異なるボリューム・グループに割り当てる必要があります。ただし、リソースサーバ機能を使用する場合は、D で示されるファイルと同じボリューム・グループに割り当ててもかまいません。

また、割り当てたボリューム・グループは、HiRDB の系切り替えと連動して切り替えられるように HA モニタのサーバ対応環境設定で設定しておく必要があります。

同一 :

現用系マシン上のファイルを、予備系マシン上の現用系マシンと同じディレクトリ名下にあらかじめ複製しておく必要があります。また、ファイルの内容を更新した場合も、更新後のファイルをあらかじめ予備系マシンに複製しておく必要があります。

— :

該当しません。

注※

現用系マシン上のファイルを、予備系マシン上にあらかじめ複製しておく必要があります。

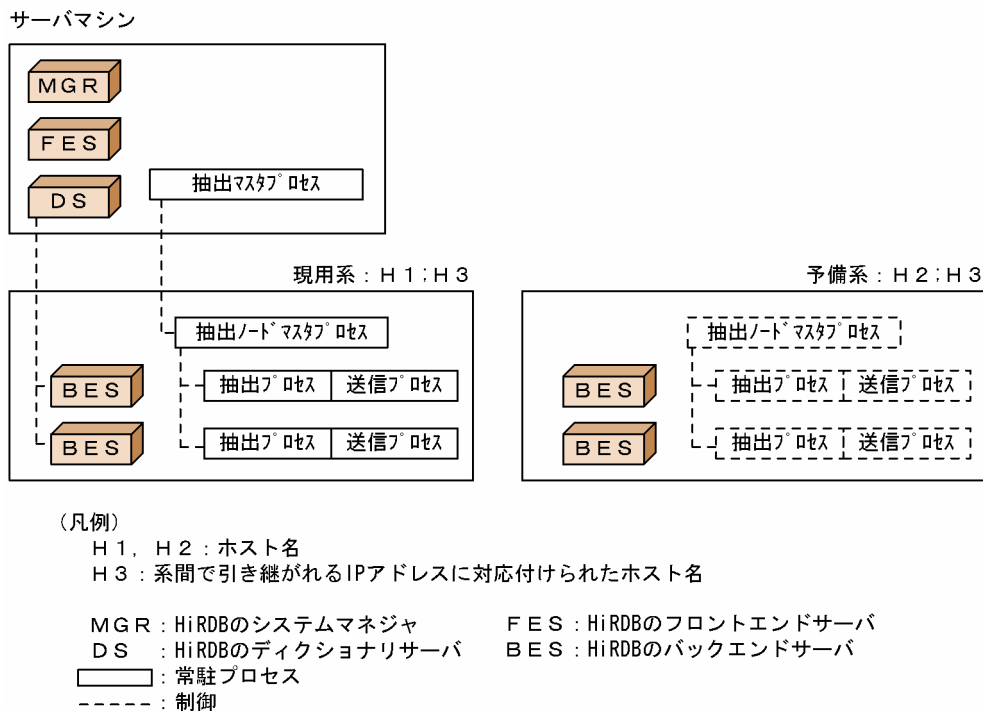
(5) HiRDB で高速系切り替え機能を使う場合の Datareplicator の定義方法

抽出側 Datareplicator では、抽出マスタプロセスから抽出ノードマスタプロセスへの通信用に IP アドレスを引き継ぐホスト名を割り当てる必要があります。このとき、HiRDB で使用するホスト名とは別のホスト名を割り当てる必要があります。

抽出側 Datareplicator 用に割り当てた IP アドレスを引き継ぐホスト名は、抽出システム定義の node_host オペランドに、抽出対象 HiRDB のシステム共通定義の pdstart コマンド、又は pdunit コマンドの -x で指定したホスト名と対応付けて指定する必要があります。

抽出側システムのサーバと主要なプロセスの関係を次の図に示します。

図 6-29 抽出側システムのサーバと主要なプロセスの関係



抽出対象 HiRDB のシステム共通定義の pdstart コマンド、又は pdunit コマンドの -x で指定したホスト名と、抽出ノードマスタプロセスを起動するホスト名との対応付けを定義に指定します。抽出マスタプロセスは、抽出ノードマスタプロセスを起動するホストに対して接続します。

pdsys (システム共通定義)

```
pdunit -x H1 -c H2 -u unt1
```

hdeenv (抽出システム定義)

```
nodedef (H1)
set node_host = H3
```

H3を使用してノードマスタに接続。

6.9.3 系切り替え機能を使うときの準備 (Microsoft Cluster Server の場合)

Microsoft Cluster Server (MSCS) による系切り替え機能を使うときに準備する項目を次に示します。ここで説明している項目は、「6.9.1(1) 系切り替えの種類」で説明した系切り替えすべてで有効です。

MSCS については、MSCS のマニュアルを参照してください。

(1) 制限事項

MSCS による系切り替え機能では、次の場合、実行系と待機系共に、同時に抽出対象となるユニットは一つだけです。

- 一つのサーバマシン上で複数の HiRDB のユニットが稼働する場合
- 相互系切り替えの形態で、一つのサーバマシン上で複数の HiRDB のユニットが稼働する可能性がある場合

(2) Datareplicator の準備

現用系及び待機系のサーバマシンの HDEPATH 又は HDSPATH に、共用ディスク上のパス名称を指定してください。共用ディスク上のパス名称及びディレクトリ名称に使用できる文字の制限を次に示します。

- 大文字、小文字の使用を統一してください。
- ディレクトリ名称には半角英数字だけ使用できます。
- 空白、全角文字、及び特殊記号は使用できません。

次に、システム環境変数を指定します。詳細について、次の表に示します。

表 6-19 システム環境変数の設定値 (MSCS による系切り替え機能使用時)

区分	システム環境変数	現用系及び待機系での設定値
抽出側 Datareplicator	HDEPATH	抽出システム定義ファイル (hdeenv) の格納先フォルダへのパス
	PDDIR	MGR が存在するユニットの PDDIR
	PDCONFPATH	MGR が存在するユニットの PDCONFPATH
	PATH	抽出側 Datareplicator のコマンドライブラリへのパス※
反映側 Datareplicator	HDSPATH	反映システム定義ファイル (hdsenv) の格納先フォルダへのパス
	PDDIR	反映側 HiRDB の PDDIR
	PDNAMEPORT	反映側 HiRDB のポート番号
	PDHOST	反映側 HiRDB のホスト名

区分	システム環境変数	現用系及び待機系での設定値
	PATH	反映側 Datareplicator のコマンドライブラリへのパス※

注※

通常は Datareplicator のインストール時に自動設定されますが、PATH 環境変数の長さによって自動設定がエラーとなった場合、手動で設定してください。

なお、抽出側 Datareplicator で hdeprep コマンド又は hdeevent コマンドを実行するには、クライアント環境定義が必要です。クライアント環境定義については、マニュアル「HiRDB UAP 開発ガイド」を参照してください。

(3) サービスの設定

系切り替え機能を使用するときの、Datareplicator のサービスの設定について次の表に示します。

表 6-20 Datareplicator のサービスの設定

区分	サービス名称	スタートアップの種類	MSCS への登録
抽出側 Datareplicator	HiRDB Datareplicator (Source site)	手動	登録する
	HiRDB Datareplicator (Source site NMT)	自動	登録しない
反映側 Datareplicator	HiRDB Datareplicator (Target Site)	手動	登録する

(4) ハードウェアの準備

系切り替え機能を使うためには、Datareplicator のファイルを共有ディスク装置に割り当てる必要があります。

Datareplicator 運用ディレクトリ (HDEPATH 及び HDSPATH 環境変数で指定したディレクトリ) を含め、「表 6-18 系切り替え機能を使うときのファイルの準備」で示したすべてのファイルを現用系と予備系とで共有する外付けハードディスクに作成し、互いの Datareplicator から同じパスでファイルを参照できるように設定してください。

(5) MSCS の準備

MSCS に登録する資源を次の表に示します。登録手順については MSCS のマニュアルを参照してください。

表 6-21 MSCS に登録する資源

区分	定義項目	リソース	名称	依存関係
抽出側 Datareplicator	抽出側 Datareplicator で使用する共有ディスク※1	共有ディスク	任意	なし
	ネットワーク名称※2	ネットワーク名		
	IP アドレス※2	IP アドレス		
	抽出側 Datareplicator の起動サービス	汎用サービス	HiRDB Datareplicator extract	HiRDB の起動サービス
反映側 Datareplicator	反映側 Datareplicator で使用する共有ディスク※3	共有ディスク	任意	なし
	ネットワーク名称※2	ネットワーク名		
	IP アドレス※2	IP アドレス		
	反映側 Datareplicator の起動サービス	汎用サービス	HiRDB Datareplicator reflect	<ul style="list-style-type: none"> • HiRDB の起動サービス • 共有ディスク

注※1

共有ディスクを「HiRDB の起動サービス」の依存関係に追加してください。

注※2

HiRDB で設定したものと同一ものを使用してください。

注※3

共有ディスクを「反映側 Datareplicator の起動サービス」の依存関係に追加してください。

(6) ネットワークの準備

次のネットワーク構成は、IP アドレスを引き継ぐ構成にしてください。

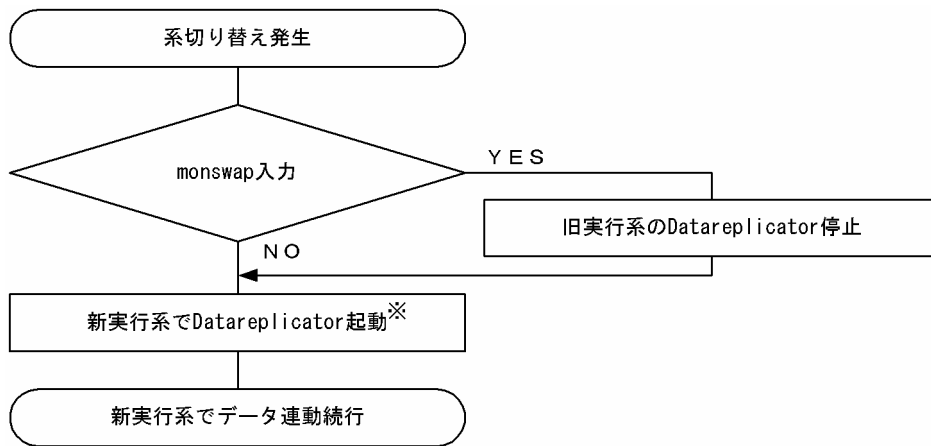
- 抽出側 Datareplicator の抽出マスタプロセスと抽出ノードマスタプロセス間
- 抽出側 Datareplicator と反映側 Datareplicator 間

6.9.4 系切り替え機能を使うときの運用手順 (HA モニタの場合)

HA モニタによる系切り替え機能を使うときの運用手順を次の図に示します。

MSCS による系切り替え機能を使うときの運用手順については、MSCS のマニュアルを参照してください。

図 6-30 HA モニタによる系切り替え機能を使うときの運用手順



注※

抽出側 Datareplicator のコマンドは、HiRDB のシステムマネージャユニット上で稼働する抽出マスタープロセス (hdemaster) に対して発行する必要があります。

このため、抽出側 HiRDB がパラレルサーバの場合に、非システムマネージャユニットが系切り替えするときは、Datareplicator の起動コマンド及び停止コマンドが、システムマネージャユニットに対して発行されるように、他のサーバ上でコマンドを実行するためのコマンド (リモートシェル, セキュアシェル等) をクラスタソフトウェアに登録する必要があります。

6.9.5 系切り替え機能を使うときの注意

系切り替え機能を使った HiRDB システムでデータ連動するときの注意について説明します。

(1) 抽出側 HiRDB がパラレルサーバで相互系切り替えの形態のときの注意

抽出側 HiRDB がパラレルサーバで相互系切り替えの形態にすると、系切り替えで 1 台のサーバマシン上で、一つの抽出対象 HiRDB に属するノードマスタープロセスが複数稼働する状態になることがあります。このとき、各ノードマスタープロセスの運用ディレクトリが一つのサーバマシン上で重複するので、運用ディレクトリ下のファイル名がノードマスタープロセス間で重複するおそれがあります。

抽出側 HiRDB がパラレルサーバで相互系切り替えの形態の例を図 6-31 に、図 6-31 のサーバマシン 3 に着目した場合の、運用ディレクトリ下の重複するファイルを表 6-22 に示します。

図 6-31 抽出側 HiRDB がパラレルサーバで相互系切り替えの形態の例

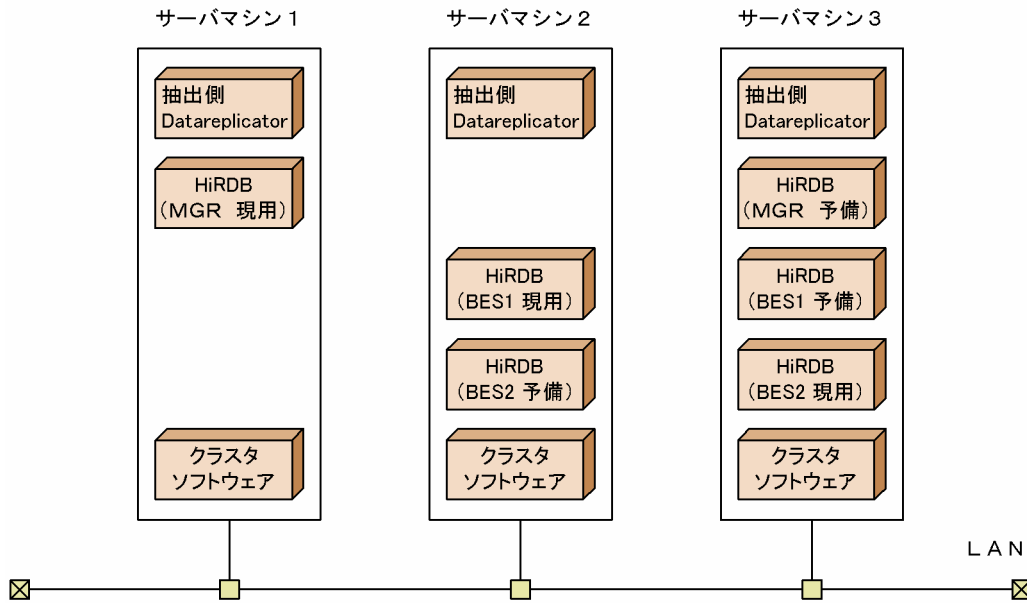


表 6-22 サーバマシン 3 に着目した運用ディレクトリ下の重複するファイル

番号	稼働サーバ			Datareplicator 運用ディレクトリ下のファイル
	MGR	BES1	BES2	
1	—	—	○	<ul style="list-style-type: none"> キューファイル ステータスファイル データ連動用連絡ファイル エラー情報ファイル 稼働トレースファイル
2	○	—	○	<ul style="list-style-type: none"> キューファイル ステータスファイル データ連動用連絡ファイル エラー情報ファイル 稼働トレースファイル
3	—	○	○	<ul style="list-style-type: none"> マスタステータスファイル マスタエラー情報ファイル キューファイル ステータスファイル データ連動用連絡ファイル <u>エラー情報ファイル</u> <u>稼働トレースファイル</u>
4	○	○	○	<ul style="list-style-type: none"> マスタステータスファイル マスタエラー情報ファイル キューファイル ステータスファイル データ連動用連絡ファイル

番号	稼働サーバ			Datareplicator 運用ディレクトリ下のファイル
	MGR	BES1	BES2	
				<ul style="list-style-type: none"> • エラー情報ファイル • 稼働トレースファイル

(凡例)

- ：稼働しているサーバ
- ：稼働していないサーバ
- _____：ノードマスタプロセス間で重複して使うファイル

ノードマスタプロセスが複数稼働すると、エラー情報ファイルと稼働トレースファイルが重複して使われることがあります。このような系切り替え構成では、抽出システム定義の `errfile_unique` オペランドに `true` を設定して、エラー情報ファイルと稼働トレースファイルが重複して使われるのを避けるようにしてください。 `errfile_unique` オペランドに `false` を指定すると、この形態で系切り替えが実行された後に、待機系のノードマスタプロセスの再起動がエラーになります。

(2) Datareplicator の状態を監視するときの注意

Datareplicator の状態を監視する場合は、Datareplicator が出力するメッセージを監視してください。

(3) 手動でディスクを活性化させる必要がある場合

系切り替えの稼働中、ディスクの活性化及び非活性化はクラスタソフトウェアが制御します。しかし、系切り替えをいったん停止して初期化するときは、ユーザ自身で活性化させる必要があります。ディスクを初期化する前には、ディスクを活性化させておいてください。ただし、初期化前にディスクを活性化してアクセスできるようにしないと、ディスクを初期化できないので注意してください。

(4) マシンの時刻に関する注意

系切り替え構成の場合は、実行系、待機系それぞれのマシンの時刻を合わせてください。実行系、待機系それぞれのマシンの時刻が異なると、系切り替えが発生したとき、抽出プロセスがシステムログファイルから更新情報を抽出できません。

(5) 抽出対象 HiRDB が一つのサーバ上に複数のユニットが稼働する構成の場合

系切り替えの過程で抽出ノードマスタプロセスが系切り替え元ユニットで再起動し、系切り替え元ユニットでファイル I/O エラー、又は切り替え先ユニットで起動エラーとなる場合があります。これを回避するために、次の設定をしてください。

- 抽出ノードマスタプロセスのユニット系切り替え時、ディスクの切り離しをする前に再度起動されたノードマスタプロセスを `hdeshmclean` コマンドで強制停止する。
- 切り替え先サーバで Datareplicator が起動エラーになった場合は、再度 Datareplicator を起動する。

(6) 抽出対象 HiRDB がパラレル構成の場合のサーバ定義作成単位

抽出側 HiRDB がパラレル構成の場合の、サーバ定義の作成単位を次に示します。

- スタンバイ型系切り替え, 及び 1:1 スタンバイレス型系切り替えの場合
HiRDB のシステムマネージャ, 及びバックエンドサーバがあるユニットごとにサーバ定義を作成します。
- 影響分散スタンバイレス型系切り替えの場合
HiRDB のシステムマネージャ, 及びバックエンドサーバごとにサーバ定義を作成します。

6.9.6 影響分散スタンバイレス型系切り替え機能での運用

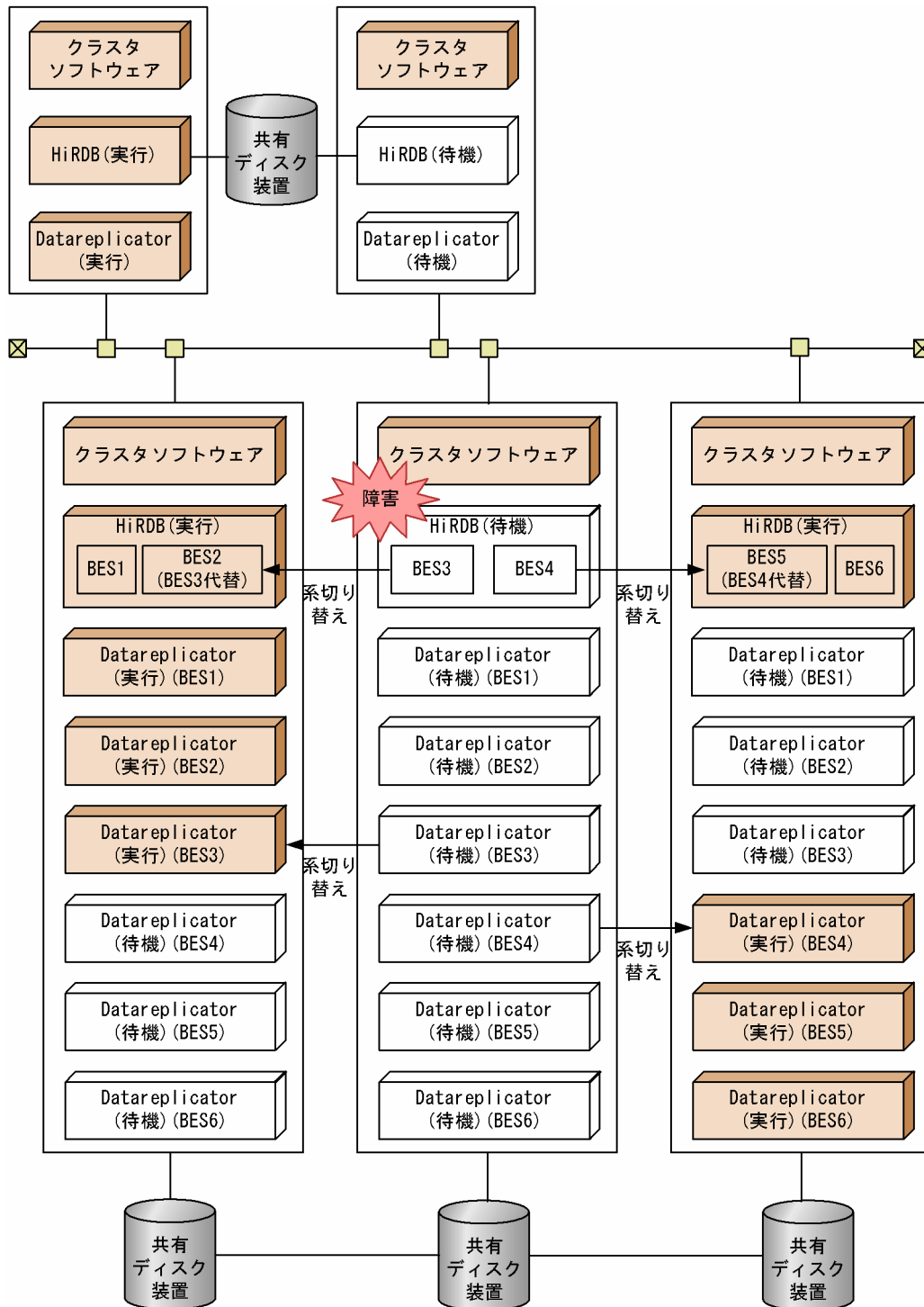
影響分散スタンバイレス型系切り替え機能を使用する場合に注意する点について説明します。

(1) 影響分散スタンバイレス型系切り替え機能でのデータ連動システムの形態

抽出対象 HiRDB が影響分散スタンバイレス型系切り替え構成の場合, バックエンドサーバ単位に系が切り替わります。このため, 抽出側 Datareplicator もバックエンドサーバ単位に系を切り替えるための設定が必要となります。

影響分散スタンバイレス型系切り替え機能でのデータ連動システムの形態を次の図に示します。

図 6-32 影響分散スタンバイレス型系切り替え機能でのデータ連動システムの形態



(2) 影響分散スタンバイレス型系切り替え機能を使うときの準備

ここでは、影響分散スタンバイレス型系切り替え機能に関する準備（通常の系切り替え機能を使う場合の準備と異なる点）について説明します。通常の系切り替え機能を使うときの準備については、「6.9.2 系切り替え機能を使うときの準備（HA モニタの場合）」を参照してください。

(a) HA モニタの準備

通常の系切り替えを行う場合と同様に、HA モニタの server 定義文で Datareplicator の動作環境を定義します。ただし、影響分散スタンバイレス型系切り替えの場合は、抽出対象 HiRDB がバックエンドサーバ単位の切り替えとなるため、HiRDB と連動して系切り替えが行われるように Datareplicator の定義も HiRDB のバックエンドサーバに対応して定義する必要があります。

詳細については、「[6.9.2 系切り替え機能を使うときの準備 \(HA モニタの場合\)](#)」を参照してください。

(b) 抽出側 Datareplicator のホスト名の割り当て

影響分散スタンバイレス型系切り替え機能を使う場合、次のようにホスト名を割り当ててください。

抽出マスタプロセスが稼働するホスト：

IP アドレス引き継ぎありのホスト名を割り当ててください。なお、割り当てるホスト名は、抽出対象 HiRDB が使用するホスト名とは別に、Datareplicator 用として新たに割り当てる必要があります。

抽出ノードマスタプロセスが稼働するホスト：

HiRDB が使用する IP アドレス引き継ぎなしのホスト名を Datareplicator も使用するため、Datareplicator 用として新たにホスト名を割り当てる必要はありません。

(c) Datareplicator の起動、及び停止のシェルの作成

現用系 HiRDB の Datareplicator を停止し、待機系 HiRDB の Datareplicator を起動するためのシェルを作成してください。シェルの作成例を次に示します。

起動のシェル作成例

- システムマネージャがあるユニットの場合

```
sleep 5※  
su -ユーザ名 -c "/opt/hirdbds/bin/hdestart_n" 2> /tmp/start_log
```

- システムマネージャがないユニットの場合

```
sleep 5※  
su -ユーザ名 -c "/opt/hirdbds/bin/hdestart_n -b BES名 -x ホスト名 -n サービス名 " 2> /  
tmp/start_log
```

注

起動コマンド実行時のエラーメッセージは、標準エラー出力に対して出力されます。シェル内で起動コマンドを実行する場合は、標準エラー出力をワークファイル（上記例では/tmp/start_log）に出力するようにしてください。

注※

データ連動用連絡ファイルを格納するボリュームを活性化した後（HiRDB の起動が完了した後）に Datareplicator が起動されるよう、sleep で調整してください。ただし、抽出側 Datareplicator の運用を簡略化する場合は、sleep は必要ありません。抽出側 Datareplicator の運用を簡略化する場

合の詳細については、「[6.9.6\(3\) 抽出側 Datareplicator の運用の簡略化 \(HA モニタ限定\)](#)」を参照してください。

停止のシェル作成例

- システムマネージャがあるユニットの場合

```
su -ユーザ名 -c "/opt/hirdbds/bin/hdestop_n -w" 2> /tmp/stop_log
```

- システムマネージャがないユニットの場合

```
su -ユーザ名 -c "/opt/hirdbds/bin/hdestop_n -b BES名 -t sendterm -c continue -w" 2> /tmp/stop_log
```

注

停止コマンド実行時のエラーメッセージは、標準エラー出力に対して出力されます。シェル内で停止コマンドを実行する場合は、標準エラー出力をワークファイル（上記例では/tmp/stop_log）に出力するようにしてください。

(d) 抽出システム定義

影響分散スタンバイレス型系切り替え機能を使う場合、次の抽出システム定義のオペランドを設定する必要があります。

nodecontrol=server

影響分散スタンバイレス型系切り替え機能を使う場合、HiRDB はバックエンドサーバ単位で系を切り替えます。そのため、抽出側 Datareplicator も、HiRDB と同様にバックエンドサーバ単位で系を切り替えるように設定します。

node_connection_accept=true

抽出ノードマスタプロセスからの接続要求の受け付け、及び系切り替え時の抽出マスタプロセスと抽出ノードマスタプロセス間の回線再接続を行うように設定します。

connection_accept_hostname=抽出ノードマスタプロセスからの接続要求受け付けホスト名

抽出ノードマスタプロセスからの接続要求を受け付けるための、抽出マスタプロセスが稼働するサーバマシンに割り当てた、IP アドレス引き継ぎありのホスト名を指定します。

connection_accept_service=抽出ノードマスタプロセスからの接続要求受け付けサービス名

抽出ノードマスタプロセスからの接続要求を受け付けるためのサービス名を指定します。

connection_accept_waittime =抽出ノードマスタプロセス接続要求待ち時間

すべての抽出ノードマスタプロセスとの回線が切断した場合、抽出マスタプロセスが抽出ノードマスタプロセスからの再接続要求を受け付けるための待ち時間を指定します。

connection_retry_time=抽出ノードマスタプロセス再接続処理時間

抽出マスタプロセスとの回線が切断した場合、抽出ノードマスタプロセスが抽出マスタプロセスに対して回線の再接続処理を行う時間を指定します。

(3) 抽出側 Datareplicator の運用の簡略化 (HA モニタ限定)

次の条件を二つとも満たす場合、HA モニタが提供するユーザコマンドインタフェースを使用できます。

- クラスタソフトウェアが HA モニタである。
- 抽出対象 HiRDB が影響分散スタンバイレス型系切り替え機能を使用している。

HA モニタが提供するユーザコマンドインタフェースを使用すると、HiRDB の系切り替え制御に抽出側 Datareplicator の系切り替え制御が組み込まれるため、系切り替えの準備及び運用が簡略化できます。

簡略化の準備と運用を次に示します。

(a) 準備

1. ユーザコマンドの作成

HiRDB の系切り替え中に行われる、Datareplicator の起動、停止を実行するためのユーザコマンドを作成します。ユーザコマンドのインタフェースについては、マニュアル「高信頼化システム監視機能 HA モニタ」を参照してください。

ユーザコマンドの作成例については、下記「[ユーザコマンド作成例](#)」を参照してください。

2. HA モニタ環境設定に対するユーザコマンドの登録

HiRDB の系切り替え中にユーザコマンドが実行されるように、HA モニタ環境設定の `usrcommand` オペランドに作成したユーザコマンドを登録します。

3. 共有リソースの割り当て

- IP アドレス引き継ぎありのホスト名の割り当て
稼働中のプロセスと HiRDB が使用するホスト名によって、割り当てられるかどうか異なります。割り当ての有無について、次に示します。

HiRDB が使用するホスト名	ホストで稼働中のプロセス	
	抽出マスタープロセス	抽出ノードマスタープロセス※
IP アドレス引き継ぎあり	×	×
IP アドレス引き継ぎなし	○	×

(凡例)

○ : Datareplicator 用として、新たに IP アドレス引き継ぎありのホスト名を割り当てる必要があります。

× : Datareplicator 用として、新たに IP アドレス引き継ぎありのホスト名を割り当てる必要はありません。

注※

抽出側 Datareplicator と反映側 Datareplicator 間のネットワークについては、簡略化しない場合と同様に、反映側 Datareplicator のホスト名に IP アドレス引き継ぎありのホスト名を割り当てる必要があります。

- 共有ボリュームの割り当て

データ連動用連絡ファイルだけでなく、すべての Datareplicator 関連ファイルを、HiRDB の系切り替え制御で切り替えられる共有ボリュームに割り当てる必要があります。

4. Datareplicator の起動、及び停止のシェルの作成

運用を簡略化する場合、Datareplicator の起動、及び停止のシェルのユーザコマンドから実行してください。シェルの作成の詳細については、「[6.9.6\(2\)\(c\) Datareplicator の起動、及び停止のシェルの作成](#)」を参照してください。

5. クラスタソフトウェアの定義作成

Datareplicator の切り替え制御は、HiRDB の系切り替え中にユーザコマンドで行われるため、Datareplicator の系切り替えのための HA モニタサーバ定義は作成する必要はありません。

6. Datareplicator 定義

運用簡略をしない場合と同じです。

ユーザコマンド作成例

- case コマンドの各パターンで指定する値は、御使用になる HA モニタの server 定義に合わせて修正してください。また、パターンの数も御使用になる HiRDB のサーバ数に合わせて修正してください。
- Datareplicator の起動・停止シェル名は、実際に御使用になるシェル名に修正してください（_sh で終わっているコマンドが Datareplicator の起動・停止シェルになります）。

ユーザコマンドの作成例を次に示します。

```
#!/bin/sh
HA_SERVER_ID=$2
SERVER_KIND=$4
SERVER_STATUS=$5
START_PARAM=$6

#####
# convert from HAmontor's server name to HiRDB's server name #
#####
case "$HA_SERVER_ID" in
"REPmng")
  HiRDB_SERVER_ID="MST"
  ;;
"REPb11")
  HiRDB_SERVER_ID="bes11"
  ;;
"REPb12")
  HiRDB_SERVER_ID="bes12"
  ;;
;;
```

```

"REpb21")
  HiRDB_SERVER_ID="bes21"
;;
"REpb22")
  HiRDB_SERVER_ID="bes22"
;;
"REpb31")
  HiRDB_SERVER_ID="bes31"
;;
"REpb32")
  HiRDB_SERVER_ID="bes32"
;;
*)
  exit 0
;;
esac

#####
# issue HiRDB Datareplicator command #
#####
if [ $SERVER_KIND = "online" ]
then
  case "$SERVER_STATUS" in
    "-s")
      #####
      # CASE : online : Normal Start #
      #####
      if [ $START_PARAM = "end" ]
      then
        if [ $HiRDB_SERVER_ID = "MST" ]
        then
          /HiRDBDS/SH/StartMST_sh
        else
          /HiRDBDS/SH/StartNMT_sh $HiRDB_SERVER_ID "initial"
        fi
      fi
      ;;
    "-e")
      #####
      # CASE : online : Normal End #
      #####
      if [ $START_PARAM = "start" ]
      then
        if [ $HiRDB_SERVER_ID = "MST" ]
        then
          /HiRDBDS/SH/StopMST_sh
        else
          /HiRDBDS/SH/StopNMT_sh $HiRDB_SERVER_ID "initial"
        fi
      fi
      ;;
    "-p")
      #####
      # CASE : online : Planning End #
      #####
      if [ $START_PARAM = "start" ]
      then

```

```

if [ $HiRDB_SERVER_ID = "MST" ]
then
  /HiRDBDS/SH/StopMST_sh
else
  /HiRDBDS/SH/StopNMT_sh $HiRDB_SERVER_ID "initial"
fi
fi
;;
"-a")
#####
# CASE : online : Switch by Server down #
#####
if [ $START_PARAM = "start" ]
then
  if [ $HiRDB_SERVER_ID = "MST" ]
  then
    /HiRDBDS/SH/StopMST_sh
  else
    /HiRDBDS/SH/StopNMT_sh $HiRDB_SERVER_ID "continue" "-t sendterm"
  fi
fi
;;
"-w")
#####
# CASE : online : Switch by Planning #
#####
if [ $START_PARAM = "start" ]
then
  if [ $HiRDB_SERVER_ID = "MST" ]
  then
    /HiRDBDS/SH/StopMST_sh
  else
    /HiRDBDS/SH/StopNMT_sh $HiRDB_SERVER_ID "continue"
  fi
fi
;;
esac
else
case "$SERVER_STATUS" in
"-a")
#####
# CASE : standby : Switch by Server down #
#####
if [ $START_PARAM = "end" ]
then
  if [ $HiRDB_SERVER_ID = "MST" ]
  then
    /HiRDBDS/SH/StartMST_sh
  else
    /HiRDBDS/SH/StartNMT_sh $HiRDB_SERVER_ID "continue"
  fi
fi
;;
"-w")
#####
# CASE : standby : Switch by Planning #
#####
if [ $START_PARAM = "end" ]

```

```
then
  if [ $HiRDB_SERVER_ID = "MST" ]
  then
    /HiRDBDS/SH/StartMST_sh
  else
    /HiRDBDS/SH/StartNMT_sh $HiRDB_SERVER_ID "continue"
  fi
fi
;;
esac
fi
```

(b) 運用

1. 抽出側 Datareplicator の初期化

抽出側 Datareplicator の初期化手順を次に示します。

1. 抽出対象 HiRDB を正常終了します。
2. HA モニタを終了します。
3. HA モニタ環境設定の usrcommand オペランドの、Datareplicator 制御用コマンドの指定を削除します。
4. HA モニタを起動します。
5. 抽出対象 HiRDB を非連動モード (pd_rpl_init_start=N) で開始します。
6. 抽出側 Datareplicator の環境の初期化 (hdestart -i コマンド) を実行します。
7. 抽出定義プリプロセスファイルの作成 (hdeprep コマンド) を実行します。
8. 抽出対象 HiRDB を正常終了します。
9. HA モニタを終了します。
10. HA モニタ環境設定の usrcommand オペランドに、Datareplicator 制御用コマンドを登録します。
11. HA モニタを起動します。
12. 抽出対象 HiRDB を連動モード (pd_rpl_init_start=Y) で開始します。HiRDB の開始と連動して、抽出側 Datareplicator が起動されます。

2. 抽出側 Datareplicator の起動, 停止

抽出対象 HiRDB の開始, 終了と連動して, 抽出側 Datareplicator が起動, 停止するため, 操作は不要です。

3. HiRDB 終了状態での抽出側 Datareplicator の起動

抽出対象 HiRDB が未抽出ログ満杯などによって開始できない場合, 抽出側 Datareplicator だけを起動することができます。その手順を次に示します。

1. Datareplicator の起動に必要な, システムマネージャのユニットの共有リソースを手動で活性化します。
2. システムマネージャのユニットで抽出マスタプロセスを起動します (hdestart_n コマンド)。

3. 起動対象の抽出ノードマスタプロセスに必要な、共有リソースを手動で活性化します。
4. バックエンドサーバのユニットで、抽出ノードマスタプロセスを起動します (hdestart_n -b サーバ名)。

6.9.7 系切り替え機能を使うときの準備 (Amazon EC2 環境の HA モニタの場合)

Amazon EC2 環境で、HA モニタによる系切り替え機能を使うときに準備する項目を次に示します。

この章に記載がない項目については、「[6.9.2 系切り替え機能を使うときの準備 \(HA モニタの場合\)](#)」を参照してください。

(1) 前提ソフトウェア

Amazon EC2 環境上にクラスタ構成を適用する場合、次の製品を用意します。

- HA モニタ 01-70 以降
- DRBD (Distributed Replicated Block Device) v9

また、各 Amazon EC2 インスタンスごとに次のストレージを配置します。

- Amazon EBS (Amazon Elastic Block Store)

(2) サポートする系切り替え構成

次の表に示す構成で系切り替えできます。

系切り替えの構成		抽出側	反映側
一つのリージョン内で系切り替えする	一つの AZ 内	○	○
	複数の AZ 間	○	○
複数のリージョン間で系切り替えする		×	×

(3) DRBD に関する設定

リソース名.res ファイルに次のオプションを設定します。

(a) device オプション

DRBD でレプリケーションするブロックデバイスを指定します。

(b) disk オプション

Amazon EBS のブロックデバイスを指定します。

(4) HA モニタに関する設定項目

「6.9.2 系切り替え機能を使うときの準備 (HA モニタの場合)」の「(1)HA モニタの準備」の設定内容の他に、servers 定義ファイルに次のオペランドを設定します。

(a) rep_device オペランド

ディスクをレプリケーションする構成を使用する場合に、DRBD リソース名を指定します。

6.10 ファイルの二重化を使うときの運用

ファイルの二重化に関するファイル構成変更の方法について説明します。次の三つの構成変更の手順について説明します。この節では、ファイルの二重化に対して、通常の運用のことをファイルの一重化と表記します。

- ファイルの一重化から二重化への変更
- 二重化するファイルの変更
- ファイルの二重化から一重化への変更

なお、Datareplicator Extension を使用している場合の変更については、マニュアル「HiRDB データ連動拡張機能 HiRDB Datareplicator Extension」を参照してください。

6.10.1 ファイルの一重化から二重化への変更

ファイルの一重化から二重化に変更するときの手順を次に示します。太字部分は、二重化のための変更を行っている部分です。

(1) 抽出側 Datareplicator での変更手順

抽出側 Datareplicator での変更手順を次に示します。

1. 抽出側 HiRDB を正常停止します。
2. `hdestate` コマンドで抽出情報キューファイルの read 位置と write 位置が同じであることを確認します。
3. `hdestop` コマンドで Datareplicator を停止します。
4. **二重化定義ファイルを作成します。**
5. **二重化するファイルの物理ファイルを用意します**※。
6. **二重化するファイルの論理ファイルと同じ名称のファイルが存在する場合、そのファイルを削除します。**
7. `hdestart -i` コマンドで Datareplicator を初期化します。
8. 各ノードの `$HDEPATH` 下に作成された二重化制御ファイルのバックアップを取得します。
9. 系切り替え構成の場合、切り替え先の `$HDEPATH` 下に二重化制御ファイルをコピーします。
10. 抽出側 HiRDB を起動します。
11. `hdeprep` コマンドを実行します。
12. `hdestart` コマンドで Datareplicator を起動します。

注※

物理ファイルの用意とは、キャラクタ型スペシャルファイルを使用する場合のパーティション割り当て、及びシンボリックリンクの作成を意味します。

(2) 反映側 Datareplicator での変更手順

反映側 Datareplicator での変更手順を次に示します。

抽出側 Datareplicator での変更手順との基本的な違いは、HiRDB を停止する必要がないことです。

1. hdsstate コマンドで反映情報キューファイルの read 位置と write 位置が同じであることを確認します。
2. hdsstop コマンドで Datareplicator を停止します。
3. 二重化定義ファイルを作成します。
4. 二重化するファイルの物理ファイルを用意します※。
5. 二重化するファイルの論理ファイルと同じ名称のファイルが存在する場合、そのファイルを削除します。
6. hdsstart -i コマンドで Datareplicator を初期起動します。
7. \$HDSPATH 下に作成された二重化制御ファイルのバックアップを取得します。
8. 系切り替え構成の場合、切り替え先の \$HDSPATH 下に二重化制御ファイルをコピーします。

注※

物理ファイルの用意とは、キャラクタ型スペシャルファイルを使用する場合のパーティション割り当て、及びシンボリックリンクの作成を意味します。

なお、反映情報キューファイルと反映ステータスファイルをキャラクタ型スペシャルファイルで作成した後初めて初期化する場合は、反映側 Datareplicator を hdsstart -i -f コマンドで初期開始してください。2 回目以降は -f オプションを指定する必要はありません。

6.10.2 二重化するファイルの変更

二重化するファイルを追加、変更、削除するときの手順を次に示します。太字部分は、二重化のための変更を行っている部分です。

手順は、基本的にファイルの一重化から二重化への変更と同じです。

(1) 抽出側 Datareplicator での変更手順

抽出側 Datareplicator での変更手順を次に示します。

1. 抽出側 HiRDB を正常停止します。
2. hdestate コマンドで抽出情報キューファイルの read 位置と write 位置が同じであることを確認します。
3. hdestop コマンドで Datareplicator を停止します。
4. **二重化定義ファイルを変更します。**
5. **二重化するファイルの物理ファイルを用意します※。**
6. **未使用になる物理ファイルを削除します。**

7. 二重化するファイルの論理ファイルと同じ名称のファイルが存在する場合、そのファイルを削除します。
8. `hdestart -i` コマンドで Datareplicator を初期化します。
9. 各ノードの `$HDEPATH` 下に作成された二重化制御ファイルのバックアップを取得します。
10. 系切り替え構成の場合、切り替え先の `$HDEPATH` 下に二重化制御ファイルをコピーします。
11. 抽出側 HiRDB を起動します。
12. `hdeprep` コマンドを実行します。
13. `hdestart` コマンドで Datareplicator を起動します。

注※

物理ファイルの用意とは、キャラクタ型スペシャルファイルを使用する場合のパーティション割り当て、及びシンボリックリンクの作成を意味します。

(2) 反映側 Datareplicator での変更手順

反映側 Datareplicator での変更手順を次に示します。

抽出側 Datareplicator での変更手順との基本的な違いは、HiRDB を停止する必要がないことです。

1. `hdsstate` コマンドで反映情報キューファイルの read 位置と write 位置が同じであることを確認します。
2. `hdsstop` コマンドで Datareplicator を停止します。
3. 二重化定義ファイルを変更します。
4. 二重化するファイルの物理ファイルを用意します※。
5. 未使用になる物理ファイルを削除します。
6. 二重化するファイルの論理ファイルと同じ名称のファイルが存在する場合、そのファイルを削除します。
7. `hdsstart -i` コマンドで Datareplicator を初期起動します。
8. `$HDSPATH` 下に作成された二重化制御ファイルのバックアップを取得します。
9. 系切り替え構成の場合、切り替え先の `$HDSPATH` 下に二重化制御ファイルをコピーします。

注※

物理ファイルの用意とは、キャラクタ型スペシャルファイルを使用する場合のパーティション割り当て、及びシンボリックリンクの作成を意味します。

なお、反映情報キューファイルと反映ステータスファイルをキャラクタ型スペシャルファイルで作成した後に初めて初期化する場合は、反映側 Datareplicator を `hdsstart -i -f` コマンドで初期開始してください。2 回目以降は `-f` オプションを指定する必要はありません。

6.10.3 ファイルの二重化から一重化への変更

ファイルの二重化から一重化に変更するときの手順を次に示します。太字部分は、一重化に戻すための変更を行っている部分です。

(1) 抽出側 Datareplicator での変更手順

抽出側 Datareplicator での変更手順を次に示します。

1. 抽出側 HiRDB を正常停止します。
2. `hdestate` コマンドで抽出情報キューファイルの read 位置と write 位置が同じであることを確認します。
3. `hdestop` コマンドで Datareplicator を停止します。
4. 二重化で使用していた物理ファイルを削除します。
5. 各ノードの `$HDEPATH` 下に作成された二重化制御ファイルを削除します。
6. 系切り替え構成の場合、切り替え先の `$HDEPATH` 下の二重化制御ファイルを削除します。
7. 抽出システム定義からファイル二重化指定を削除します。
8. `hdestart -i` コマンドで Datareplicator を初期化します。
9. 抽出側 HiRDB を起動します。
10. `hdeprep` コマンドを実行します。
11. `hdestart` コマンドで Datareplicator を起動します。

(2) 反映側 Datareplicator での変更手順

反映側 Datareplicator での変更手順を次に示します。

1. `hdsstate` コマンドで反映情報キューファイルの read 位置と write 位置が同じであることを確認します。
2. `hdsstop` コマンドで Datareplicator を停止します。
3. 二重化で使用していた物理ファイルを削除します。
4. `$HDSPATH` 下に作成された二重化構成定義ファイルを削除します。
5. 反映システム定義からファイル二重化指定を削除します。
6. `$HDSPATH` 下に作成された二重化制御ファイルを削除します。
7. 系切り替え構成の場合、切り替え先の `$HDEPATH` 下の二重化制御ファイルを削除します。
8. `hdsstart -i` コマンドで Datareplicator を初期起動します。

6.11 ラージファイルの運用

抽出情報キューファイル、及び反映情報キューファイルの最大値を 2GB 以上（ラージファイル）として運用する方法について説明します。この節では、抽出情報キューファイル、及び反映情報キューファイルを、キューファイルと表記します。

Windows 版の場合

Windows 版でラージファイルを運用する場合、抽出環境定義の `queuesize` オペランド、及び反映環境定義の `queuesize` オペランドに 2097152（2GB）以上を指定します。

UNIX 版の場合

UNIX 版でラージファイルを運用する場合、抽出環境定義の `queuesize` オペランド、及び反映環境定義の `queuesize` オペランドに 2097152（2GB）以上を指定します。また、OS の設定や Datareplicator ファイルシステム領域の最大長を設定する必要があります。詳細は、「[6.11.1 ラージファイル運用の準備（UNIX 版限定）](#)」を参照してください。

注意事項

既存のキューファイルのサイズを変更した場合、Datareplicator を初期化する必要があります。そのため、レプリケーションが完了していることを事前に確認してから、ファイルサイズを変更してください。

6.11.1 ラージファイル運用の準備（UNIX 版限定）

UNIX 版でラージファイルを運用するための準備について説明します。

(1) システムで使用できるファイルサイズの最大値の設定

システムで使用できるファイルサイズは OS で制限されています。そのため、次のどちらかの方法で、システムで使用できるファイルサイズの最大値をラージファイルに対応させる必要があります。

- Datareplicator のプロセスが使用するファイルサイズの最大値より大きな値にする。
- 最大値を設定しない（無制限にする）。

なお、最大値を変更するために必要な OS のコマンド、及び操作手順については、OS のマニュアル、及びシェルのマニュアルを参照してください。

特に、AIX 版は、デフォルトのファイルサイズ制限が 1GB のため注意してください。各 OS で使用できるファイルサイズの最大値は、`limit` コマンド又は `ulimit` コマンドで確認できます。詳細については、各 OS のマニュアル、及びシェルのマニュアルを参照してください。

注意

AIX 版では、inetd から起動するプロセスが使用するファイルサイズの最大値は、管理者権限（スーパーユーザ）と同じになります。抽出マスタプロセスは inetd から起動されるため、次の手順で、管理者権限（スーパーユーザ）の最大値を変更してください。通信設定の詳細については、「[2.5.1 抽出側 Datareplicator の通信環境の設定](#)」を参照してください。

手順	設定内容	コマンド
1	抽出マスタプロセスと抽出ノードプロセスとの通信で使うサービス名とポート番号を追加します。	# vi* /etc/services
2	inetd の通信環境を設定します。	# vi* /etc/inetd.conf
3	管理者権限（スーパーユーザ）のファイルサイズの最大値を設定します。	# vi* /etc/security/limits
4	inetd を停止します。	# stopsrc -s inetd
5	inetd を起動します。	# startsrc -s inetd

注※

テキストエディタに制限はありません。

(2) Datareplicator ファイルシステム領域の最大長の設定

2GB 以上のキューファイルを Datareplicator ファイルシステム領域に格納できるようにするために、hdsfmkfs -n コマンドに 2GB 以上の値を設定します。ただし、設定できる最大長が OS ごとに異なるため、次の表の値を確認してください。

表 6-23 ラージファイル運用時の Datareplicator ファイルシステム領域の最大長

Datareplicator の種類	条件		Datareplicator ファイルシステム領域の最大長 (単位：MB)
HP-UX 版	ラージファイル未使用	通常ファイル	2,047
		キャラクタ型スペシャルファイル	
	ラージファイル使用	通常ファイル	131,071
		キャラクタ型スペシャルファイル	
Solaris 版	ラージファイル未使用	通常ファイル	2,047
		キャラクタ型スペシャルファイル	
	ラージファイル使用	通常ファイル	1,048,575
		キャラクタ型スペシャルファイル	

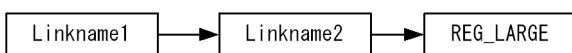
Datareplicator の種類	条件		Datareplicator ファイルシステム領域の最大長 (単位：MB)
AIX 版※	ラージファイル未使用	通常ファイル	2,047
		キャラクタ型スペシャルファイル	
	ラージファイル使用	通常ファイル (JFS)	65,411
		通常ファイル (JFS2)	1,048,575
		キャラクタ型スペシャルファイル	
Linux 版	ラージファイル未使用	通常ファイル	2,047
		キャラクタ型スペシャルファイル	
	ラージファイル使用	通常ファイル	1,048,575
		キャラクタ型スペシャルファイル	

注※

ラージファイルを運用する場合、JFS と JFS2 で Datareplicator ファイルシステム領域として指定できる最大長が異なります。そのため、smit コマンドで、作成先ファイルシステムでラージファイルを使用できるように設定してください。

(3) リンクファイル設定時の注意点

リンク名称を用いてラージファイルを運用する場合、リンクのネストは一つまでしか指定できません。次の図のように、二つ以上ネストした場合、作成対象ファイルをラージファイルとして作成できません。



(凡例)

→ : シンボリックリンク

Linkname1 : リンク名1

Linkname2 : リンク名2

REG_LARGE : 作成対象ファイル

この例の場合、Linkname1 ではなく、Linkname2 を使用します。

6.11.2 ラージファイル運用時のコマンド実行時間の見積もり

キューファイルをラージファイルとして運用すると、次に示すコマンドの実行時間が長くなります。

- hdestart (抽出側 Datareplicator の起動)
- hdemodq (抽出情報キューファイルの構成変更)
- hdsstart (反映側 Datareplicator の起動)
- hdefcopy, hdsfcopy (現用ファイルの複写)

(1) hdestart (抽出側 Datareplicator の起動) の見積もり式

-i init オプションを指定した場合にコマンド実行時間が長くなります。

$$\begin{aligned} \text{抽出情報キューファイル初期作成時間 (マイクロ秒)} &= \sum \text{unit} (T_UNIT) \\ T_UNIT &= \sum \text{svr} (T_SVR) \\ T_SVR &= \sum \text{que} (T_QUE) \\ T_QUE &= \text{DiskW} \times \text{キューファイルサイズ (KB)} \div 1024 \end{aligned}$$

- $\sum \text{unit}$
ユニット内の抽出情報キューファイル作成時間の総和
- T_UNIT
一つのユニット内の全抽出情報キューファイルの初期作成時間
- $\sum \text{svr}$
バックエンドサーバの抽出情報キューファイル作成時間の総和
- T_SVR
一つのバックエンドサーバ用の全抽出情報キューファイルの初期作成時間
- $\sum \text{que}$
抽出情報キューファイル作成時間の総和
- T_QUE
一つの抽出情報キューファイルの初期作成時間
- DiskW
ファイルを配置するディスクに対して、1KB のデータの書き込みに掛かる時間

注意

初期化実行時には、抽出システム定義の通信待ち時間 (cmwaittime オペランドに指定した値) に、各ユニットで算出したユニット内の全抽出情報キューファイル初期作成時間 (T_UNIT) の最大値を指定してください。

抽出システム定義の通信待ち時間 (cmwaittime オペランドに指定した値) 内に処理が終了しないでエラー (KFRB00607-E) となる場合は、通信待ち時間を大きくし、コマンドを再実行してください。

(2) hdemodq (抽出情報キューファイルの構成変更) の見積もり式

-a オプションを指定した場合にコマンド実行時間が長くなります。

$$\text{抽出情報キューファイル追加登録時間 (マイクロ秒)} = \text{DiskW} \times \text{キューファイルサイズ (KB)} \div 1024$$

- DiskW

ファイルを配置するディスクに対して、1KB のデータの書き込みに掛かる時間

(3) hdsstart (反映側 Datareplicator の起動) の見積もり式

-i init オプションを指定した場合にコマンド実行時間が長くなります。

$$\begin{aligned} \text{反映情報キューファイル初期作成時間 (マイクロ秒)} &= \sum \text{que} (T_QUE) + \sum \text{sts} (T_STS) \\ T_QUE &= \text{DiskW} \times \text{キューファイルサイズ (KB)} \div 1024 \\ T_STS &= \text{DiskW} \times \text{ステータスファイルサイズ (KB)} \div 1024 \end{aligned}$$

- $\sum \text{que}$

反映情報キューファイル作成時間の総和

- T_QUE

一つの反映情報キューファイルの初期作成時間

- $\sum \text{sts}$

ステータスファイル作成時間の総和

- T_STS

一つのステータスファイルの初期作成時間

- DiskW

ファイルを配置するディスクに対して、1KB のデータの書き込みに掛かる時間

(4) hdefcopy, hdsfcopy (現用ファイルの複写) の見積もり式

$$\begin{aligned} \text{現用ファイル複写時間 (マイクロ秒)} &= \text{DiskR} \times \text{複写元ファイルサイズ (KB)} \div 1024 \\ &+ \text{DiskW} \times \text{複写先ファイルサイズ (KB)} \div 1024 \end{aligned}$$

- DiskR

ファイルを配置するディスクから、1KB のデータの読み込みに掛かる時間

- DiskW

ファイルを配置するディスクに対して、1KB のデータの書き込みに掛かる時間

注意

複写されるファイルのサイズは、次のように決まります。

ファイル種別	複製対象ファイル	サイズ
通常ファイル	すべて	コマンド実行時の複製元ファイルのサイズ
キャラクタ型スペシャルファイル	抽出情報キューファイル	抽出環境定義の <code>queuesize</code> オペランドに指定した値
	反映情報キューファイル	反映環境定義の <code>queuesize</code> オペランドに指定した値
	上記以外	コマンド実行時の複製元ファイルのサイズ

6.12 チューニング

Datareplicator のチューニングについて説明します。

6.12.1 チューニングの要否

次の観点から、チューニングの要否を判断してください。

- 抽出側 Datareplicator の送信性能
- 反映側 Datareplicator の反映性能

(1) 抽出側 Datareplicator の送信性能

次のように情報を取得し、送信性能のチューニング要否を判断してください。

1. 定期的に hdestate コマンドを実行し、Queue write position, 及び Queue read position の情報を取得してください。
2. Queue write position のオフセット位置から Queue read position のオフセット位置を引いた値を算出してください。
3. 経過時間に比例して、2.で算出した値が増加している場合は、次のチューニングを検討してください。
 - 送信間隔
 - 更新情報編集バッファサイズ
 - 送信マスタプロセス送信間隔
 - 更新情報編集バッファ長
 - 送信プロセスの抽出情報キューファイル読み込み間隔
 - 抽出情報キュー I/O バッファサイズ

ただし、反映情報キューファイルが満杯の場合や、通信回線に障害が発生した場合も、経過時間に比例して2.で算出した値が増加します。その場合は、障害要因を取り除いた後で、再度オフセット位置を取得し、経過時間とオフセット差分の推移を見直してください。

(2) 反映側 Datareplicator の反映性能

次のように情報を取得し、反映性能のチューニング要否を判断してください。

1. 定期的に hdsstate コマンドを実行し、Queue write position, 及び Queue read position の情報を取得してください。
2. Queue write position のオフセット位置から Queue read position のオフセット位置を引いた値を算出してください。

3. 経過時間に比例して、2.で算出した値が増加している場合は、次のチューニングを検討してください。

- SQL の性能
- 反映処理コミット間隔
- トランザクション単位反映方式での反映処理コミット間隔
- 表単位反映方式での反映処理コミット間隔
- 反映トランザクション内最大更新 SQL 数
- COMMIT 発行間隔
- 反映処理の同時実行性
- 反映プロセスの反映情報キューファイル読み込み間隔

SQL の性能は、HiRDB の SQL トレースを取得することで判断できます。SQL トレースを取得する場合、環境変数 PDSQLTRACE を設定する必要があります。環境変数 PDSQLTRACE は、UNIX の場合は反映側 Datareplicator の環境変数に設定してください。Windows の場合はシステム環境変数又は hirdb.ini に設定してください。PDSQLTRACE の詳細については、マニュアル「HiRDB システム定義」を参照してください。

6.12.2 チューニングの方法

チューニングの方法について次の表に示します。どのチューニングが必要か判断した上で方法を選択してください。

表 6-24 チューニングの方法

処理性能	チューニングの要素	チューニングの方法	推奨度
送信性能	<ul style="list-style-type: none"> • 送信間隔 • 更新情報編集バッファサイズ • 送信マスタプロセス送信間隔 • 更新情報編集バッファ長 	<p>次のオペランドの値を調整してください。</p> <ul style="list-style-type: none"> • sendcontrol オペランドに nodemst を指定した場合 <ul style="list-style-type: none"> • sendintvl オペランド • sendintvl_scale オペランド • editbufsize オペランド <p>詳細は、「5.4 送信環境定義」を参照してください。</p> <ul style="list-style-type: none"> • sendcontrol オペランドに sendmst を指定した場合 <ul style="list-style-type: none"> • smt_sendintvl オペランド • smt_sendintvl_scale オペランド • smt_editbufsize オペランド <p>詳細は、「5.2 抽出システム定義」を参照してください。</p>	中

処理性能	チューニングの要素	チューニングの方法	推奨度
	<ul style="list-style-type: none"> 送信プロセスの抽出情報キューファイル読み込み間隔 抽出情報キュー I/O バッファサイズ 	<p>次のオペランドの値を調整してください。</p> <ul style="list-style-type: none"> quiosize オペランド queue_read_wait_interval オペランド <p>quiosize オペランドの詳細は「5.3 抽出環境定義」を、queue_read_wait_interval オペランドの詳細は、「5.4 送信環境定義」を参照してください。</p>	低
反映性能	SQL の性能	<p>反映側 HiRDB で次の作業を実施してください。</p> <ul style="list-style-type: none"> 反映対象表に複数のインデクスを定義している場合 不要なインデクスを削除 反映処理に掛かる時間が増加した場合 反映対象表を格納している RD エリアを再編成 	高
	<ul style="list-style-type: none"> 反映処理コミット間隔 トランザクション単位反映方式での反映処理コミット間隔 表単位反映方式での反映処理コミット間隔 反映トランザクション内最大更新 SQL 数 	<p>次のオペランドの値を調整してください。</p> <ul style="list-style-type: none"> cmtintvl オペランド trncmtintvl オペランド tblcmtintvl オペランド reflect_trn_max_sqlnum オペランド <p>詳細は、「5.9 反映環境定義」を参照してください。</p>	高
	COMMIT 発行間隔	commit_wait_time オペランドを調整してください。詳細は、「 5.9 反映環境定義 」を参照してください。	高
	反映処理の同時実行性	反映グループを定義し、表単位反映方式を適用してください。詳細は、「 5.10 反映定義 」を参照してください。	中
	反映プロセスの反映情報キューファイル読み込み間隔	ref_wait_interval オペランドの値を調整してください。詳細は、「 5.9 反映環境定義 」を参照してください。	低

6.13 運用時の注意事項

6.13.1 OS の時刻変更についての注意事項

HiRDB Datareplicator が稼働しているマシンの OS の時刻を戻した場合、正しく動作できません。OS の時刻を戻す場合、HiRDB のマニュアル「システム運用ガイド」を参照してください。

6.13.2 Datareplicator が作成するファイルのバックアップについて

Datareplicator の初期化時に作成する次のファイルについては、バックアップからの回復を行わないでください。バックアップから回復した場合、ファイル破壊と同じ状態になり、動作を保証できなくなります。

- 抽出マスタステータスファイル
- 抽出サーバステータスファイル
- 抽出情報キューファイル
- データ連動用連絡ファイル
- 反映情報キューファイル
- 反映ステータスファイル
- 反映マスタステータスファイル

ファイル障害などで回復が必要な場合は、「9. 障害時の対処」に従って対処してください。

7

コマンドの文法

この章では、Datareplicator のコマンドの文法について説明します。なお、コマンドはアルファベット順に記載しています。

コマンドの概要

Datereplicator でデータ連動の業務を実行するときに使うコマンドの文法について説明します。

Datereplicator のコマンドを実行する前に、**環境変数**を設定する必要があります。設定する必要がある環境変数については、「[2.4 環境変数の設定 \(UNIX\)](#)」又は「[2.8 環境変数の設定 \(Windows\)](#)」を参照してください。

Windows Vista, 及び Windows Server 2008 でのコマンドの実行

UAC 機能を利用している場合、Datereplicator でコマンドを実行するとき、「管理者として実行」したコマンドプロンプトかどうかによって、動作が異なります。詳細については、「[2.10.1 コマンドの実行](#)」を参照してください。

Datereplicator でデータ連動するときに使うコマンドの一覧を次の表に示します。また、コマンドログに出力される情報について「[表 7-2 コマンドログに出力される情報](#)」に示します。

表 7-1 Datereplicator でデータ連動するときに使うコマンドの一覧

コマンドの種類	コマンド名	機能	稼働中での実行	
			HiRDB 稼働中	Datereplica tor 稼働中
抽出側 Datereplicator のコマンド	hdestart ^{*1}	抽出側 Datereplicator を起動します。 -i オプションを指定すると抽出側 Datereplicator 環境を初期化できます。この場合、起動はしません。	—	△ ^{*4}
	hdestop ^{*2}	抽出側 Datereplicator を停止します。	—	○
	hdeevent ^{*2}	イベントを発行します。	○	—
	hdestate ^{*2}	抽出側 Datereplicator の状態を標準出力に出力し ます。	—	○
	hdeprep ^{*2}	抽出定義を解析して、抽出定義プリプロセスファ イルを作成します	○	△ ^{*5}
	hdemodq ^{*3}	抽出情報キューファイルの構成を変更します	—	×
	hdefcopy ^{*2}	ファイルの二重化で障害が発生したときに、障害 ファイルに現用ファイルを複写します。	—	×
	hdefstate ^{*2}	ファイルの二重化で使用している、物理ファイルの 状態を表示します。	—	△
	hdeshmclean	抽出側 Datereplicator の共用リソースを削除しま す。	—	×
	hdestart_n ^{*2}	抽出側 Datereplicator を部分起動します。	—	○
	hdestop_n ^{*2}	抽出側 Datereplicator を部分停止します。	—	○

コマンドの種類	コマンド名	機能	稼働中での実行	
			HiRDB 稼働中	Datareplica tor 稼働中
	hdechgstatus ^{※2}	抽出側 Datareplicator が停止と起動をわたっても継続して保持する状態、及び情報を変更します。	—	○
	hdeinfoget ^{※2, ※9}	抽出側 Datareplicator で発生した障害の、原因調査に必要な情報を取得します。	—	△
反映側 Datareplicator のコマンド	hdsstart	反映側 Datareplicator を起動します。 - i オプションを指定すると、反映側 Datareplicator 環境を初期化して起動できます。	—	×
	hdsstop ^{※6}	反映側 Datareplicator を終了します。	—	○
	hdsrctl ^{※6}	反映処理方式を制御したり、反映処理だけを再起動したりします。	—	○
	hdsstate ^{※6}	反映側 Datareplicator の状態を標準出力に出力します。	—	○
	hdssamqin ^{※6, ※7}	SAM ファイルを使うメインフレーム側 DB から抽出した更新情報を、反映情報キューファイルに出力します。	—	○
	hdsfcopy ^{※6}	ファイルの二重化で障害が発生したときに、障害ファイルに現用ファイルを複製します。	—	×
	hdsfstate ^{※6}	ファイルの二重化で使用している、物理ファイルの状態を表示します。	—	△
	hdsrefinm ^{※6}	反映情報ステータスファイルから、各更新情報を抽出して、解析結果出力ファイルに出力します。	—	△
	hdschgstatus ^{※6}	反映側 Datareplicator が停止と起動をわたっても継続して保持する状態、及び情報を変更します。	—	○
	hdsshmclean	反映側 Datareplicator の共用リソースを削除します。	—	×
	hdsinfoget ^{※6, ※9}	反映側 Datareplicator で発生した障害の、原因調査に必要な情報を取得します。	—	△
	hdsemergencystop ^{※10}	反映側 Datareplicator の全プロセスを強制終了します。	—	△
HiRDB の コマンド	pdrplstart	抽出側 HiRDB が HiRDB Datareplicator 連携を開始します。	○	—
	pdrplstop	抽出側 HiRDB が HiRDB Datareplicator 連携を中止します。	○	—
	pdls (-d rpl 指定時)	抽出側 HiRDB での HiRDB Datareplicator 連携の状態を表示します。	—	—

コマンドの種類	コマンド名	機能	稼働中での実行	
			HiRDB 稼働中	Datareplica tor 稼働中
	pdlogchg (-R 指定時)	システムログファイルの抽出未完了状態を、抽出完了状態に変更します。	—	—
文字コード変換用マッピングテーブルを編集するコマンド	hdscnvedt	文字コード変換用マッピングテーブルを編集します。文字コード変換用マッピングテーブルを更新する場合は、及び外字マッピングファイルの移行をする場合は、スーパーユーザでコマンドを実行してください。	—	—
稼働トレースファイルを編集するコマンド	hdstrcedit※2, ※6	稼働トレースファイルを編集します。	—	—
Datareplicator ファイルシステム領域で使うコマンド※8	hdsfmkfs※2, ※6, ※9	Datareplicator ファイルシステム領域を初期化します。	—	×
	hdsfstatfs※2, ※6, ※9	Datareplicator ファイルシステム領域の状態を表示します。	—	—

(凡例)

- ：稼働中のときだけ実行できます。
- ×
- △：停止中のときでも実行できます。
- ：関連しません。

注※1

HiRDB 管理者と同じユーザグループに属しているユーザ名で実行してください。

注※2

抽出側 Datareplicator の初期化 (hdestart -i コマンド) を実行したユーザ名で実行してください。

注※3

抽出ノードマスタプロセス実行ユーザ (inetd.conf で指定) と同じユーザ名で実行してください。

注※4

-i オプション指定は、抽出側 Datareplicator の稼働中には実行できません。

注※5

稼働中に実行しても、コマンドの実行結果が抽出側 Datareplicator に反映されるのは、抽出側 Datareplicator を停止してから、再開始した時点です (コマンド実行中に稼働している抽出側 Datareplicator には反映されません)。

注※6

反映側 Datareplicator の初期化 (hdsstart -i コマンド) を実行したユーザ名で実行してください。

注※7

抽出側 DB が SAM ファイルを使うときにだけ使うコマンドです。XDM/DS を使ってデータ連動する場合の反映側 Datareplicator では使いません。

注※8

Datareplicator ファイルシステム領域を設定している Datareplicator で使うコマンドです。スーパーユーザだけが実行できます。これらのコマンドは、抽出側 Datareplicator と反映側 Datareplicator で共通です。Datareplicator ファイルシステム領域を使ったファイル管理については、「[3.4 Datareplicator ファイルシステム領域](#)」を参照してください。

注※9

Windows 版 Datareplicator ではサポートしていないコマンドです。

注※10

Windows 版 Datareplicator だけでサポートしているコマンドです。

表 7-2 コマンドログに出力される情報

コマンド名	出力される情報			
	コマンド引数	コマンド戻り値	エラー発生時の詳細情報	正常終了時の詳細情報
hdechgstatus	×	×	×	×
hdeevent hdeeventO hdeeventS	○	○	×	×
hdefcopy	○	○	×	×
hdefstate	×	×	×	×
hdeinfoget	×	×	×	×
hdemodq	○	○	×	×
hdeprep (抽出定義で抽出定義文を指定する場合)	○	○	×	×
hdeprep (抽出定義で全表抽出定義文を指定する場合)	○	○	○	○
hdeprepO hdeprepS	○	○	×	×
hdeshmclean	○	○	×	×
hdestart hdestartO hdestartS	○	○	○	×

コマンド名	出力される情報			
	コマンド引数	コマンド戻り値	エラー発生時の詳細情報	正常終了時の詳細情報
hdestart_n	○	○	○	×
hdestate	×	×	×	×
hdestop hdestopO hdestopS	○	○	○	×
hdestop_n	○	○	○	×
hdscnvedt	×	×	×	×
hdschgstatus	×	×	×	×
hdsemergencystop	×	×	×	×
hdsfcopy	○	○	×	×
hdsfmkfs	×	×	×	×
hdsfstate	×	×	×	×
hdsfstatfs	×	×	×	×
hdsinfoget	×	×	×	×
hdsrefinfm	×	×	×	×
hdsrfctl	○	○	×	×
hdssamqin	○	○	×	×
hdsshmclean	○	○	×	×
hdsstart	○	○	○	×
hdsstate	×	×	×	×
hdsstop	○	○	○	×
hdstrcredit	×	×	×	×
pdlogchg	×	×	×	×
pdl	×	×	×	×
pdrplstart	×	×	×	×
pdrplstop	×	×	×	×

(凡例)

○：出力されます。

×：出力されません。

hdechgststatus (抽出側 Datareplicator の状態変更)

機能

抽出情報キューファイルの最大使用率を変更します。このコマンドで変更した情報は、抽出側 Datareplicator を停止、又は起動しても保持されます。

形式

```
hdechgststatus -c reset -k max_ratio
```

オプション

- -c reset
-k で指定した情報をリセットするときに指定します。
- -k max_ratio
抽出情報キューファイルの最大使用率を変更するときに指定します。

規則

- hdechgststatus コマンドを実行後、正常終了すると 0 が返されます。異常終了すると 1 が返されます。また、次の条件を満たした場合は 11 が返されます。
 1. 抽出環境定義の extsuppress オペランドに true を指定したバックエンドサーバがある。
 2. 上記バックエンドサーバに抽出対象表が存在する。
 3. 1., 及び 2. の条件に該当するバックエンドサーバだけで構成されるユニットである。

注意

- hdechgststatus コマンドは、抽出側 Datareplicator が稼働中のときだけ実行できます。停止中に実行すると、エラーになります。
- hdechgststatus コマンド実行時に稼働していないノードがある場合、又は hdechgststatus コマンド実行中にノードがダウンした場合、稼働中のノードに対してだけ状態を変更します。変更できなかったノードについては、すべてのノードが稼働している状態で、再度実行してください。

hdeevent (抽出側 Datareplicator でのイベント発行)

Windows 版 Datareplicator の場合、HiRDB が 08-02 以降のときは、このコマンドは抽出対象の HiRDB コマンドプロンプトから実行してください。

機能

hdeevent のオプションに指定したイベントコードを反映側システムに送信し、反映処理の動作を制御します。hdeevent を実行する場合には、イベント制御用テーブルを抽出側 HiRDB に作成する必要があります。イベント制御用テーブルについては、「4.6.7 イベント制御用テーブルの設計」を参照してください。

形式

```
hdeevent -n イベントコード  
          [ -u ユーザID/パスワード ] [ -s 送信先識別子 ]
```

オプション

- -n イベントコード

イベントコードを指定します。指定できるイベントコードの範囲は次のとおりです。

0

送信プロセスを停止します。

1~255

指定したイベントコードを反映側 Datareplicator、もしくは XDM/DS に送信します。

256~32767

指定したイベントコードを反映側 Datareplicator に送信します。

ただし、反映側 Datareplicator のバージョンが 08-10 以上の場合に限りです。

反映側 Datareplicator のバージョンが 08-09 以前、又は XDM/DS に対して送信しようとした場合、送信プロセスで KFRB02073-W を出力し^{※1}、該当するイベントコードの送信は行われません。

注※1 hdeevent コマンドは正常に終了し、実際に送信を行うときに警告メッセージを出力します。

- -u ユーザ ID/パスワード

ユーザ ID ~ 〈1~8 文字の記号名称〉

パスワード ~ 〈1~28 文字の記号名称〉

イベント制御用テーブルを作成したユーザ ID とパスワードを指定します。ユーザ ID、パスワードともに大文字、小文字は区別されます。また、区切り記号 (/) とユーザ ID、パスワードの間に空白を入れないでください。ユーザ ID は 8 文字、パスワードは 28 文字までの記号名称で指定してください。

-u オプションを省略すると、抽出側 HiRDB の環境変数 PDUSER の値が仮定されます。PDUSER を設定していないと、hdeevent の実行時に抽出側 HiRDB との接続処理でエラーになります。

- -s 送信先識別子

イベントの送信先を、送信先識別子で指定します。送信先識別子に**を指定した場合は、引数不正エラーとなります。

-s オプションを省略すると、すべての送信先に対してイベントを送信します。

規則

- hdeevent コマンドを実行すると、抽出側 HiRDB に作成したイベント制御用テーブルに対して SQL 文を発行します。ここで発行した SQL 文の更新ログがシステムログファイルに格納されます。抽出側 Datareplicator では、システムログファイルからこの更新情報を抽出し、反映側システムにイベントとして送信します。
- イベントコードと反映処理の動作については、反映側 Datareplicator の反映環境定義で設定します。反映環境定義で設定されていないイベントコードのイベントを発行した場合には、反映処理の動作に影響はありません。ただし、反映側システムの syslog ファイルにメッセージが出力されます。
- hdeevent コマンドを実行後、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- イベントコードが 1~255 以外のイベントを発行した場合、反映側 Datareplicator には送信されません。

注意

- hdeevent コマンドは、抽出側 HiRDB が稼働中のときだけ実行できます。停止中に実行すると、エラーになります。
- hdeevent コマンドは、抽出側システムでのトランザクション停止時に実行してください。トランザクション実行中に hdeevent コマンドを実行した場合、どのトランザクションからイベントが有効になるかの判断ができないことがあります。この場合、反映側 Datareplicator のイベント検知時点では、hdeevent コマンド実行時に実行中であったトランザクションは反映されていないことがあります。
- 抽出側 HiRDB がパラレルサーバの場合、システムマネージャの存在するサーバマシンから hdeevent コマンドを実行してください。システムマネージャの存在しないサーバマシンから hdeevent コマンドを実行した場合、-s オプションを指定しているときは、エラーが発生して KFRB04001-E メッセージが出力されます。

また、監査証跡の取得を抑止していた場合、hirdb_audit_trail オペランドの指定を無視して監査証跡が取得されます。

hdefcopy (抽出側現用ファイルの複写)

機能

抽出側システムのファイルの二重化で障害が発生して一方の物理ファイルが使用できない状態 (HOLD 状態) になった場合、正常なファイルを障害の発生したファイルに複写するコマンドです。

これによって、障害が発生したファイルが使用可能な状態 (ACTIVE 状態) に回復します。

形式

```
hdefcopy -f 複写元物理ファイル名 -t 複写先物理ファイル名
```

オプション

- -f 複写元物理ファイル名
複写元の (正常な) 物理ファイル名を絶対パスで指定します。
- -t 複写先物理ファイル名
複写先の (障害の発生した) 物理ファイル名を絶対パスで指定します。

規則

- 現用ファイル複写コマンドは、回復対象のファイルが存在するノード上で実行します。
- 現用ファイル複写コマンドは、Datareplicator が停止している時に実行可能です。データ連動用連絡ファイルを回復する場合には、HiRDB も停止している必要があります。
- 複写元物理ファイル名、又は複写先物理ファイル名に二重化定義で正しく定義されていない場合、コマンドエラーになります。
- 複写元の物理ファイルの状態が Active 状態でない場合、コマンドエラーになります※。
- 複写先の物理ファイルの状態が Hold 状態でない場合、コマンドエラーとなります。※
- 複写するファイルをラージファイルとして運用している場合、コマンドの実行時間が長くなることがあります。詳細については、「[6.11.2 ラージファイル運用時のコマンド実行時間の見積もり](#)」を参照してください。

注※

物理ファイルの状態は、hdefstate コマンドで確認できます。詳細については、[hdefstate](#) コマンド (抽出側二重化ファイルの状態表示) を参照してください。

hdefstate (抽出側二重化ファイルの状態表示)

機能

抽出側システムの二重化したファイル群 (物理ファイル) の状態を表示するコマンドです。

形式

```
hdefstate [-f 論理ファイル名]
```

オプション

- -f 論理ファイル名

状態を表示する論理ファイル名を絶対パスで指定します。

このオペランドを省略した場合、コマンドを実行したサーバマシン上のすべての二重化ファイルに関する状態を表示します。

出力形式

hdefstate コマンドで抽出情報キューファイルの状態を出力する形式を次に示します。

```
*****
**          HiRDB Datareplicator file status information          **
**          Sun Aug 25 12:58:52 2002                             **
*****
Logical file name = /users/repli/quefile001_bes1                . . . . . 1.

sys status  type Physical file name                            . . . . . 2.
a  a----- equ /hd01/hde/qufile001_bes1_1
b  a----- equ /hd01/hde/qufile001_bes1_2
-----
Logical file name = /users/repli/quefile002_bes1

sys status  type Physical file name
a  a----- equ /hd01/hde/qufile002_bes1_1
b  h----- equ /hd01/hde/qufile002_bes1_2

Number of Logical file Information = 2                          . . . . . 3.
```

1. 論理ファイル名を示します。

2. 物理ファイル状態情報を示します。

sys : 物理ファイルの系です。

A 系の場合 a, B 系の場合は b が表示されます。

出力	説明
a	A 系ファイル
b	B 系ファイル

status : 物理ファイルの状態です。1 列目の表示から状態を判断します。

出力	説明
a	現用使用状態 (ACTIVE)
h	障害発生状態 (HOLD)
*	状態を取得できません。

type：物理ファイルの種別コードです。

出力	説明
emt	抽出マスタステータスファイル
sst	サーバステータスファイル
rpl	データ連動用連絡ファイル
equ	抽出情報キューファイル
---	上記以外のファイル
***	ファイル種別を取得できません。

Physical file name：物理ファイル名です。

3. 出力された論理ファイル情報の件数を示します。

hdeinfoget (抽出側障害情報の取得)

機能

抽出側 Datareplicator で障害が発生した場合、障害の原因を調査するために定義情報、及び障害情報を取得します。

UNIX 版 Datareplicator だけで使用できるコマンドです。Windows 版 Datareplicator では使用できません。

このコマンドは Datareplicator 管理者が実行してください。

形式

```
hdeinfoget [ -n 抽出側Datareplicator識別子 ]
            -e 初期情報出力先ディレクトリ
```

オプション

- -n 抽出側 Datareplicator 識別子

取得対象とする抽出側 Datareplicator 識別子を指定します。

このオプションを省略した場合、サーバマシン上のすべての抽出側 Datareplicator 識別子が取得対象となります。

- -e 初期情報出力先ディレクトリ

初期情報ファイル（障害原因の切り分けに必要な情報をまとめたファイル）を出力する既存のディレクトリ名を絶対パスで指定します。

指定したディレクトリが存在しない、又は書き込み権限がない場合は、エラーメッセージを出力して処理を打ち切ります。

初期情報出力先ディレクトリには、初期情報ファイルと取得処理結果ログが出力されます。初期情報ファイルはアーカイブファイルであり、複数のファイルが格納されています。それぞれのファイル名を次に示します。

出力ファイル	ファイル名
初期情報ファイル	hdeinfoget_<ホスト名*1>_init_<年月日時分秒*2>.tar.gz
取得処理結果ログ	hdeinfoget_comlog_<ホスト名*1>_<年月日時分秒*2>

注※1

ホスト名の長さが 32 バイト以上の場合、先頭から 32 バイト分の文字を使用します。

注※2

年月日時分秒は「YYYYMMDDhhmmss」（YYYY：西暦年，MM：月，DD：日，hh：時，mm：分，ss：秒）です。

初期情報ファイルに取得される情報を次に示します。

分類		取得する情報	取得の有無
定義情報	Datareplicator 定義ファイル	抽出システム運用ディレクトリパス情報	○
		抽出システム定義ファイル	○
		抽出環境定義ファイル	○
		送信環境定義ファイル	○
		抽出定義ファイル	○※1
		二重化定義ファイル	○
障害情報	Datareplicator 運用ファイル	抽出定義プリプロセスファイル	△
		抽出情報キューファイル	×
		抽出マスタステータスファイル	△
		抽出サーバステータスファイル	△
		抽出マスタエラー情報ファイル	○
		抽出ノードマスタエラー情報ファイル	○
		抽出マスタトレースファイル	○
		抽出ノードマスタトレースファイル	○
		データ連動用連絡ファイル	△
		コマンドログファイル	○※2
		二重化制御ファイル	○
		UXPL ログファイル	×
		Datareplicator 運用情報	Datareplicator バージョン情報
	hdestate コマンド実行結果		○
	hdefstate コマンド実行結果		○
	hdsfstatfs コマンド実行結果		○
	共用メモリ情報		×
	pdls -d rpl -j コマンド実行結果		○
	抽出側 Datareplicator 運用ディレクトリ下のファイルリスト		○
	データ連動対象の表, インデックス定義情報	×	
	OS 情報	Datareplicator インストールディレクトリ下のファイルリスト	○
		プロセスリスト	○
		ipcs -a コマンド実行結果	○

分類	取得する情報	取得の有無
	netstat -n コマンド実行結果	○
	hosts ファイル	○
	services ファイル	○
	inetd 情報	○
	syslog ファイル	○

(凡例)

○：この情報を取得します。

△：UNIX のキャラクタ型スペシャルファイルである場合は取得しません。レギュラーファイルの場合だけ取得します。

×：この情報を取得しません。

注※1

hdeprep コマンドを実行していないサーバマシンでは取得しません。

注※2

環境変数 hde_command_log_file の指定によって任意のファイル名が付与されている場合は、この情報を取得しません。

規則

- 抽出側 Datareplicator 起動時のファイル格納場所を基に情報を取得します。そのため、抽出側 Datareplicator を一度でも正常に起動したことのある場合だけ、このコマンドで情報を取得できます。なお、このコマンド実行時に抽出側 Datareplicator が起動している必要はありません。
- コマンドを実行するユーザにアクセス権限がないファイルは取得できません。KFRB09275-W メッセージを出力して、取得はスキップされます。
- コマンドの結果は、コマンド実行時のリターンコード、又は取得処理結果ログ中のメッセージで確認できます。リターンコードの意味と対処方法を次に示します。

リターンコード	終了状態	意味と対処方法
0	正常終了	コマンドは正常に終了しました。 コンソールや取得処理結果ログに警告メッセージが出力されている場合、一部の情報が取得できていないおそれがあります。メッセージを参照して、個別に取得してください。
8	異常終了	コマンドは異常終了しました。 コンソールや取得処理結果ログに出力されたエラーメッセージを参照して、エラーの原因を取り除いてから、コマンドを再度実行するか、個別に取得してください。
12	割り込み（シグナル発生）による終了	割り込み（シグナル発生）によって処理を打ち切りました。

hdemodq (抽出情報キューファイルの構成変更)

機能

抽出情報キューファイルの構成をオフラインで変更します。次の項目を実行できます。

- 抽出情報キューファイルを追加登録
- 不要になった抽出情報キューファイルの登録を解除
- オフラインでの抽出情報キューファイルについての情報を表示

形式

```
hdemodq  [ -x ホスト名 ] -b HiRDBサーバ名
          { -l
            | -a 抽出情報キューファイル名 [-n 追加キュー二重化定義ファイル]
            | -d 抽出情報キューファイル名 }
```

オプション

- -x ホスト名 ~ 〈32文字の記号名称〉

該当するノードのホスト名を指定します。

このオプションの指定の要否は、抽出システム定義の `nodecontrol` オペランド、及び `errfile_unique` オペランドの指定値によって決まります。

抽出システム定義の `nodecontrol`、及び `errfile_unique` オペランドの指定値による -x オプション指定との関係を次に示します。

nodecontrol 及び errfile_unique オペランドの指定値	-x オプション
<ul style="list-style-type: none">• nodecontrol = unit• errfile_unique = false	指定しても、無効となります。
<ul style="list-style-type: none">• nodecontrol = server• errfile_unique = false	
<ul style="list-style-type: none">• nodecontrol = unit• errfile_unique = true	有効となります。省略した場合は、エラーとなります。
<ul style="list-style-type: none">• nodecontrol = server• errfile_unique = true	指定しても、無効となります。

- -b HiRDB サーバ名 ~ 〈8文字の記号名称〉

hdemodq コマンドの処理対象となる抽出環境を HiRDB サーバ名で指定します。指定したサーバ名に対応する抽出側 Datareplicator の環境に、キューファイルの情報表示、追加登録、登録解除が実行されます。抽出側 HiRDB がシングルサーバの場合も指定してください。

- -l
抽出情報キューファイルの情報を表示するときに指定します。

• -a 抽出情報キューファイル名

- 抽出情報キューファイルを追加登録するときに、追加する抽出情報キューファイルのファイル名を絶対パス又は相対パスで指定します。相対パスで指定したときは、「\$HDEPATH/相対パス名」が絶対パスとして仮定されます。ここでは、抽出側システムで一意になる名称を指定してください。抽出側 HiRDB がパラレルサーバのときは、各バックエンドサーバで一意になる名称を指定してください。実際の抽出情報キューファイルは、指定した名称に「_サーバ名」を付けた名称になります。絶対パスを指定するときは、指定値に「_サーバ名」を付けた名称が 125 バイト以下になるようにしてください。相対パスを指定するときは、仮定される絶対パス名に「_サーバ名」を付けた名称が 125 バイト以下になるようにしてください。
- 追加登録するキューファイルのファイルタイプは、割り当て済みのキューファイルと同じにしてください (UNIX 版の場合、割り当て済みキューファイルが Datareplicator ファイルシステム領域のときは、キャラクタ型スペシャルファイルにしてください)。ファイル種別が UNIX の通常ファイル又は Windows ファイルのときは、hdemodq コマンドがファイルを作成します。ファイル種別が UNIX のキャラクタ型スペシャルファイルのときは、hdemodq コマンドの実行前に、あらかじめファイルを作成しておいてください。ファイル名は、-a オプションに指定した名称に「_サーバ名」を付けた名称で、ファイルサイズは抽出情報キューファイルのファイルサイズ以上にしてください。
- ラージファイル運用時にこのオプションを指定している場合、コマンド実行時間が長くなります。詳細は「6.11.2 ラージファイル運用時のコマンド実行時間の見積もり」を参照してください。

-n

二重化した抽出情報キューファイルを追加する場合に指定するオペランドです。追加する抽出情報キューファイルの構成を定義した追加キュー二重化定義ファイル[※]の名称を絶対パスで指定します。このオペランドが指定されている場合、-a で指定した抽出情報キューファイル名は論理ファイル名として認識されます。

注※ 追加キュー二重化定義ファイルの構文は、二重化定義ファイルと同じです。二重化定義ファイルについては「5.7 二重化定義 (抽出側)」を参照してください。

二重化したキューファイルの追加/削除を行った場合、コマンド処理の延長で二重化制御ファイル(「3.8 ファイルの二重化」を参照)も更新されます。このため、系切り替え構成の場合はキュー追加/削除処理を行った後、二重化制御ファイルを待機系にコピーする必要があります。また、今後行われる Datareplicator の初期化以降もキュー追加/削除を有効としたい場合は、二重化定義に対しても定義内容を追加/削除しておく必要があります。

• -d 抽出情報キューファイル名

抽出情報キューファイルの登録を解除するときに、登録を解除する抽出情報キューファイルのファイル名を絶対パス又は相対パスで指定します。相対パスで指定したときは、\$HDEPATH/相対パス名が絶対パスとして仮定されます。

実際の抽出情報キューファイルの名称は、指定した名称に「_サーバ名」を付けた名称になります。

出力形式

hdemodq コマンドで抽出情報キューファイルの状態を出力する形式を次に示します。

```

hdeid = fa                               ..... 1.
server name = pbes01                     ..... 2.
wrap count = 0                           ..... 3.
status      size create date             file path ..... 4.
qufile001  u----- 8388000 Fri Sep 2 14:25:09 2005 /.../extque01_pbes1
qufile002  u----- 8388000 Fri Sep 2 14:25:09 2005 /.../extque02_pbes1
qufile003  e----- 8388000 Fri Sep 2 14:25:09 2005 /.../extque03_pbes1

```

1. 抽出側 Datareplicator 識別子を示します。
2. 抽出側 HiRDB のサーバ名を示します。
3. 抽出情報キューファイルのラップアラウンド回数を出力します。wrap count 値は、4294967295 を超えると 0 に戻ります。
4. 抽出情報キューファイルの情報を示します。

status : 抽出情報キューファイルの状態を表示します。表示内容を次に示します。

項番	コード位置	内容	値	状態
1	1 列目	未送信抽出情報の有無	u	反映側に送信されていない抽出情報あり
2			e	反映側に送信されていない抽出情報なし
3	2~7 列目	未定義	-	-

size : 抽出情報キューファイルのファイルサイズを KB 単位で表示します。

create date : 抽出情報キューファイルの作成日時を表示します。

file path : 抽出情報キューファイルのファイルパスを表示します。

注意

抽出ノードマスタエラー情報ファイルが存在しない旨のエラーが発生した場合には、次の点を見直してください。

見直しの観点	問題がある場合の対処方法
抽出ノードマスタエラー情報ファイルが不当に削除されていないか	次の手順でこのコマンドを再実行してください。 1. 抽出側 Datareplicator を再起動します。 2. 抽出側 Datareplicator を停止します。 3. hdemodq コマンドを再実行します。
-x オプションを指定している場合、その指定値が正しいか	指定値を正しい値に修正してください。

hdeprep (抽出定義プリプロセスファイルの作成)

Windows 版 Datareplicator の場合、HiRDB が 08-02 以降のときは、このコマンドは抽出対象の HiRDB コマンドプロンプトから実行してください。

機能

抽出定義ファイルの内容を解析し、内部形式に変換した抽出定義プリプロセスファイルを作成します。

形式

```
hdeprep -f 抽出定義ファイル名 [ -u ユーザID/パスワード ]  
        [ -k {not_null_unique | unique | none} ]
```

オプション

- -f 抽出定義ファイル名

～ 〈1～64 バイトのパス名〉

抽出定義ファイルの名称を絶対パス名又は相対パス名で指定します。相対パス名で指定した場合は、\$HDEPATH/相対パス名が絶対パス名に仮定されます。この場合、仮定された絶対パス名の長さが 125 バイト以下になるように指定してください。

- -u ユーザ ID/パスワード

ユーザ ID ～ 〈8 文字の記号名称〉

パスワード ～ 〈28 文字の記号名称〉

抽出側 Datareplicator に対して DBA 権限を持っているユーザの、ユーザ ID とパスワードを指定します。ユーザ ID、パスワードともに大文字、小文字は区別されます。また、区切り記号 (/) とユーザ ID、パスワードの間に空白を入れしないでください。ユーザ ID は 8 文字、パスワードは 28 文字までの記号名称で指定してください。

-u オプションを省略すると、抽出側 HiRDB の環境変数 PDUSER の値が仮定されます。PDUSER を設定していないと、hdeprep コマンドの実行時に HiRDB との接続処理でエラーになります。

- -k {not_null_unique | unique | none}

マッピングキー列に対するユニークチェックを行うときの条件を指定します。

ここで指定する値は、抽出定義で check 句を省略した場合の規定値になります。指定値とチェック内容は、抽出定義の check 句と同じです。「[5.5 抽出定義](#)」のオペランドの説明で、check 句を参照してください。

指定した条件を満たしていない場合、コマンドは異常終了します。このオプションを省略した場合は、not_null_unique が仮定されます。

抽出定義で全表抽出文を指定している場合、none を指定することはできません。指定すると KFRB04587-E メッセージを出力し、コマンドは異常終了します。

規則

- hdeprep コマンドは、hdestart -i コマンドの実行後に実行してください。
- hdeprep コマンドは、抽出側 HiRDB が稼働状態の場合だけ実行できます。また、抽出側 Datareplicator が停止状態の場合に実行してください。抽出側 Datareplicator が稼働中でも hdeprep コマンドは実行できますが、コマンドの実行結果が抽出側 Datareplicator に反映されるのは、抽出側 Datareplicator を停止してから、再開始した時点です（コマンド実行中に稼働している抽出側 Datareplicator には反映されません）。
- 抽出定義プリプロセスファイルは、\$HDEPATH 下に hde_prpfile として作成されます。
- hdeprep コマンドを実行後、正常終了すると 0 が返されます。異常終了すると 1 が返されます。

注意

- hdeprep コマンドは、次の場合に実行する必要があります。
 - hdestart -i コマンドを実行した場合
 - 抽出側 HiRDB で抽出対象の表の定義変更をした場合
 - 抽出側 Datareplicator の抽出定義変更をした場合
 - イベント制御用テーブルを作成した場合
- 抽出定義の解析時には、抽出システム定義の dblocale オペランドで設定した文字コード体系に従います。

hdeshmclean (抽出側 Datareplicator の共用リソースの削除)

機能

抽出側 Datareplicator が異常終了した場合に、抽出側 Datareplicator の共用リソース（プロセス、共用メモリ、又はセマフォ）が残っているとき、残っている共用リソースを削除します。このコマンドは UNIX 版限定のコマンドです。このコマンドは Datareplicator 管理者が実行してください。

形式

```
hdeshmclean [ -x ホスト名 ]  
            [ -l clean [ -q {resp | noresp} ] [ -w ] [ -t {mst | nmt | both} ] ]
```

オプション

- -x ホスト名 ~ (1~32 文字の識別子)

削除対象とする共用リソースが存在するホスト名を指定します。このオプションを指定する必要がある場合を次に示します。

条件	このオプションに指定する値
抽出システム定義の errfile_unique オペランドに true を指定している場合	削除対象とする共用リソースが存在するサーバのホスト名を指定します (該当するサーバの \$HDEPATH 下に作成されている errfile1_XXXXX ファイルの XXXXX と同じ名称を指定してください)。 ただし、抽出マスタプロセスだけが稼働するサーバで、-t mst オプションを指定してコマンドを実行する場合は、このオプションを指定する必要はありません。
抽出システム定義の nodecontrol オペランドに server を指定している場合	削除対象とする共用リソースが存在するバックエンドサーバ名を指定します (該当するバックエンドサーバの \$HDEPATH 下に作成されている errfile1_XXXXX ファイルの XXXXX と同じ名称を指定してください)。

- -l clean

共用リソースを削除する場合に指定します。このオプションを指定しない場合は、残っている共用リソースの表示だけ行います。

- -q {resp | noresp}

共用リソースを削除する前に、削除してよいかを確認する KFRB04331-Q メッセージを出力するかどうかを指定します。

resp

KFRB04331-Q メッセージを出力します。y を入力した場合だけ共用リソースを削除します。

noresp

KFRB04331-Q メッセージを出力しないで、共用リソースを削除します。

- -w

共用リソースの削除が完了するまで、このコマンドの終了を待ち合わせる場合に指定します。このオプションを指定しない場合、コマンドが終了した時点で共用リソースの削除も完了しているかは保証できません。

- -t {mst | nmt | both}

抽出マスタプロセスと抽出ノードマスタプロセスの両方が稼働しているサーバでコマンドを実行する場合に、どのリソースを削除対象とするかを指定します。

mst

抽出マスタプロセスが管理するリソースを削除します。

nmt

抽出ノードマスタプロセスが管理するリソースを削除します。

both

抽出マスタプロセスが管理するリソースと、抽出ノードマスタプロセスが管理するリソースの両方を削除します。

出力形式

抽出システム定義の sendcontrol オペランドに nodemst を指定

<<<*** Common information ***>>>	1
Shmid : statinf	= 13467653 2
Semid : mstcmd	= 0 3
PID : hdemaster	= 7241 4
PID : hdstrcrv	= 25754 5
<<<*** Node information (drum2) ***>>>	6
Shmid : extcom	= 13500422 7
extdef	= 13533191 8
errtxt	= 13565960 9
nmtcmd	= 0 10
Semid : extcom	= 8912925 11
nmtcmd	= 0 12
PID : hdenodemst	= 7244 13
PID : hdstrcrv	= 25757 14
<<<*** Server information (bes1) ***>>>	15
Shmid : trninf(ticb)	= 0 16
trninf(excb)	= 0 17
trninf(d2cb)	= 0 18
sndprcinf	= 0 19
Semid : svrcom	= 8945694 20
PID : hdecapture	= 7247 21
PID : hdesender (SND01)	= 7248 22
PID : hdeextwatch	= 25756 24
<<<*** Server information (bes2) ***>>>	
Shmid : trninf(ticb)	= 0	
trninf(excb)	= 0	
trninf(d2cb)	= 0	
sndprcinf	= 0	
Semid : svrcom	= 8945698	
PID : hdecapture	= 7252	
PID : hdesender (SND01)	= 7253	
⋮		

抽出システム定義の sendcontrol オペランドに sendmst を指定

```

<<<*** Common information ***>>> ..... 1
Shmid : statinf      = 13467653 ..... 2
Semid  : mstcmd      = 0 ..... 3
PID    : hdemaster   = 7241 ..... 4
PID    : hdstrcrvcv  = 25754 ..... 5

<<<*** Node information (drum2) ***>>> ..... 6
Shmid  : extcom      = 13500422 ..... 7
        extdef       = 13533191 ..... 8
        errtxt       = 13565960 ..... 9
        nmtcmd       = 0 ..... 10
Semid  : extcom      = 8912925 ..... 11
        nmtcmd       = 0 ..... 12
PID    : hdenodemst  = 7244 ..... 13
PID    : hdstrcrvcv  = 25757 ..... 14

<<<*** Server information (bes1) ***>>> ..... 15
Shmid  : trninf(ticb) = 2333 ..... 16
        trninf(excb)  = 3432 ..... 17
        trninf(d2cb)  = 4522 ..... 18
        sndprcinf     = 1123 ..... 19
Semid  : svrcom       = 8945694 ..... 20
PID    : hdecapture   = 7247 ..... 21
PID    : hdesendmst   = 7248 ..... 22
PID    : hdesndprc    = 7249 ..... 23
PID    : hdesndprc    = 7250 ..... 23
PID    : hdesndprc    = 7251 ..... 23
PID    : hdeextwatch  = 25756 ..... 24

<<<*** Server information (bes2) ***>>>
Shmid  : trninf(ticb) = 344
        trninf(excb)  = 566
        trninf(d2cb)  = 879
        sndprcinf     = 112
Semid  : svrcom       = 8945698
PID    : hdecapture   = 7252
PID    : hdesendmst   = 7253
        .
        .
        .
    
```

出力情報の説明

番号	出力情報	備考
1	抽出マスタプロセス管理リソースヘッダ	抽出マスタプロセスが管理するリソースが存在しない場合は、番号 1~5 までの情報は表示されません。
2	コマンド連絡用共用メモリ ID	—
3	ステータスコマンド排他用セマフォ ID	—
4	抽出マスタプロセスのプロセス ID	—
5	稼働トレース収集プロセスのプロセス ID	—
6	抽出ノードマスタプロセス管理リソースヘッダ	抽出ノードマスタプロセスが管理するリソースが存在しない場合は、番号 6 以降の情報は表示されません。
7	プロセス間連絡用共用メモリ ID	—
8	抽出定義格納用共用メモリ ID	—

番号	出力情報	備考
9	メッセージテキスト格納用共用メモリ ID	—
10	コマンド連絡用共用メモリ ID	抽出システム定義の nodecontrol オペランドに server を指定していない場合は、0 が出力されます。
11	ノード間排他用セマフォ	—
12	コマンド排他用セマフォ ID	抽出システム定義の nodecontrol オペランドに server を指定していない場合は、0 が出力されます。
13	抽出ノードマスタプロセスのプロセス ID	—
14	稼働トレース収集プロセスのプロセス ID	—
15	サーバ単位リソースヘッダ	—
16	トランザクション情報管理用共用メモリ ID	抽出システム定義の nodecontrol オペランドに server を指定していない場合は、0 が出力されます。
17	トランザクション内更新情報管理用共用メモリ ID	
18	トランザクション内分岐情報管理用共用メモリ ID	
19	送信プロセス管理用共用メモリ ID	
20	サーバ間排他用セマフォ	
21	抽出プロセスのプロセス ID	—
22	送信プロセスのプロセス ID	送信プロセス名は次のように表示されます。 抽出システム定義の sendcontrol オペランドに nodemst を指定している場合は、「hdesender (送信先識別子)」。 抽出システム定義の sendcontrol オペランドに sendmst を指定している場合は、「hdesendmst」。
23	送信プロセス (hdesndprc) のプロセス ID	抽出システム定義の sendcontrol オペランドに sendmst を指定している場合に出力されます。起動しているプロセスの数だけ表示されます。ただし、送信プロセス (hdesndprc) が起動してない場合は、出力されません。
24	抽出処理監視プロセス ID	抽出システム定義の extwatch_intvl オペランドを指定していない場合は、0 が出力されます。

(凡例)

— : 特にありません。

規則

残った共有リソースが原因で抽出側 Datareplicator を起動できない場合は、hdestop コマンドを実行して共有リソースを削除してください。hdestop コマンドを実行しても共有リソースを削除できない場合に、このコマンドを実行してください。

hdestart (抽出側 Datareplicator の起動)

機能

抽出システム定義に従って、抽出側 Datareplicator を起動します。抽出システム定義については、「5.2 抽出システム定義」を参照してください。

形式

```
hdestart { [ -i [ init ] [ -S 送信先識別子番号 ] ] |
           [ -e ] [ -r ]
           [ -s [ 送信先識別子 [{{, 送信先識別子}}...] ] [ -L [ -H ] ] ] |
           [ -R -k queue -b HiRDBサーバ名 ] |
           [ -v [ -e ] [ -r ] ] }
```

オプション

• -i [init] [-S 送信先識別子番号]

抽出側 Datareplicator の環境を初期化する場合に指定します。-i オプション指定時は、抽出側 Datareplicator は起動しません。

-S オプションを指定した場合は、特定の送信先への送信環境だけを初期化します。

-S オプションを指定しない場合は、次に示すファイルを初期化して、抽出側 Datareplicator の環境全体を初期化します (-S オプション指定時には、ファイルは初期化されません)。

- 抽出情報キューファイル
- 抽出マスタステータスファイル
- 抽出サーバステータスファイル
- 抽出マスタエラー情報ファイル
- 抽出ノードマスタエラー情報ファイル
- 稼働トレースファイル
- データ連動用連絡ファイル

init

抽出環境定義で指定したすべての抽出情報キューファイルを、抽出環境定義の queuesize オペランドに指定したサイズで初期作成します。

ただし、初期作成するファイルのサイズによっては、抽出システム定義の cmwaittime オペランドに指定した通信待ち時間内に処理が終了しないでメッセージ KFRB00607-E でエラーになることがあります。このときには、通信待ち時間を大きくしてから、初期化処理を再実行してください。

このオプションは、-S オプション指定時には無効になります。

ラージファイル運用時にこのオプションを指定している場合、コマンド実行時間が長くなります。詳細は「6.11.2 ラージファイル運用時のコマンド実行時間の見積もり」を参照してください。

-S 送信先識別子番号

指定した送信先への送信環境の初期化（部分初期化）をします。指定されなかった送信先に対しては何もしません。

送信先は、送信先識別子番号で指定します。送信先識別子番号には、抽出システム定義の送信先識別子（sendidxx 又は sendidxxxx）の xx 及び xxxx 部分の数字を指定します。

送信先識別子番号に指定できるのは、前回初期開始した時点で定義されていた送信先識別子番号だけで、新たに追加した送信先識別子番号は指定できません。

このオプションで部分初期化の対象になるのは、抽出システム定義の sendidxx（又は sendidxxxx）及び senddefxx（又は senddefxxxx）で指定された送信環境定義です。

部分初期化された送信先識別子の送信プロセスは、次回起動したときに、抽出プロセスが抽出キューへ書き込んだ更新情報から送信を開始します。

部分初期化以外の定義の変更は、正常開始時と同様に扱います。

部分初期化する送信先識別子以外の定義変更は行なわないでください。複数の送信先識別子の部分初期化を行う場合、対象の送信先識別子の定義変更→部分初期化を繰り返してください。

- -e

起動時に、送信処理を開始しないで、抽出処理だけを開始する場合に指定します。また、前回の hdestart 指定時に抽出処理を開始していない場合に、抽出処理を開始するときにも指定します。

- -r

抽出側 Datareplicator 起動時に送信データ件数をリセットする場合に指定します。

このオプションは、抽出システム定義の send_counter_reset オペランドに false を指定した場合に有効となります。

- -s [送信先識別子 [{{, 送信先識別子}} …]] [-L [-H]]

起動時に、抽出処理を開始しないで、送信処理又は更新 SQL 出力処理を開始する場合に指定します。また、前回の hdestart 指定動時に送信処理を開始していない場合に、送信処理を開始するときにも指定します。

- 送信先識別子

特定の送信先識別子の送信処理又は更新 SQL 出力処理だけを開始する場合に、抽出システム定義で指定した送信先識別子を指定します。複数の送信先識別子を指定する場合には、送信先識別子と送信先識別子の間に、空白を入力しないでください。送信先識別子に ** を指定した場合は、引数不正エラーとなります。

送信先識別子を省略すると、すべての送信先識別子の送信処理又は更新 SQL 出力処理が開始します。

- -L

起動時に、更新 SQL 出力処理を開始します。

- -H

このオプションは、-L オプションを指定したときだけ有効です。

更新 SQL 出力処理で CHAR, VARCHAR, MCHAR, MVARCHAR, NCHAR, 及び NVARCHAR の列の列値を 16 進形式で出力します。出力する列値の形式や内容については、「9.8.2 更新 SQL ファイル」を参照してください。

このオプションは、更新 SQL ファイルに出力された列値が文字データとして読み取れなかった場合に指定してください。列値に制御コード (0x00~0x1F) などの表示できない文字が含まれていたり、列値に文字データではなく 16 進数が格納されていたりすると、文字データとして読み取れません。

- -R

抽出環境を回復します。

- -k queue

抽出情報キューファイル回復機能を使用して、抽出情報キューファイルを回復します。抽出情報キューファイル回復機能の詳細については、「9.7 抽出情報キューファイル回復機能」を参照してください。

- -b HiRDB サーバ名 ~ (8 文字の記号名称)

回復対象とする抽出環境を HiRDB サーバ名で指定します。

抽出側 HiRDB がシングルサーバの場合は、シングルサーバのサーバ名 (HiRDB の pdstart -s コマンドに指定するサーバ名) を指定してください。

抽出側 HiRDB がパラレルサーバの場合は、回復対象のバックエンドサーバ名を指定してください。サーバ名は、バックエンドサーバの抽出環境定義に指定しているサーバ名と同じにしてください。サーバ名が異なる場合は、KFRB00711-E メッセージを出力してエラー終了します。

なお、サーバ名は一つだけ指定してください。回復対象のサーバが複数ある場合は、回復対象のサーバの数だけコマンドを実行してください。

- -V

システムログファイルによるデータ連動回復を実行します。

システムログファイルによるデータ連動回復については、「9.5 システムログファイルによるデータ連動回復」を参照してください。

規則

- hdestart コマンドを実行後、正常終了すると、0 が返されます。異常終了すると、1 が返されます。
- 開始モードは、前回の終了モードによって異なります。開始モードと前回の終了モードとの関係を次の表に示します。

表 7-3 抽出側 Datareplicator の開始モードと終了モードとの関係

開始モード	前回の終了モード	起動時の動作内容
初期開始	—※	初期開始のときには、抽出側 Datareplicator は「表 7-4 -i オプション指定時 (-S オプションは指定していないとき) の各ファイルに対する処理の内容」に示すようにファイルを初期化します。抽出側 Datareplicator は起動しません。
部分初期開始	—※	部分初期開始のときには、抽出側 Datareplicator は指定された送信先への送信環境だけを初期化します。抽出側 Datareplicator は起動しません。
正常開始	正常終了	前回の稼働時の動作内容を引き継がないで、抽出側 Datareplicator 定義に従って、抽出側 Datareplicator が開始します。

開始モード	前回の終了モード	起動時の動作内容
再開始	異常終了又は強制終了	抽出側 DB と反映側 DB の整合性を保証するため、前回の稼働時と同じ動作内容で抽出側 Datareplicator が開始します。

注※

前回の終了モードに関係なく、-i オプションを付けて hdestart コマンドを実行すると、初期開始又は部分初期開始されます。

- -i オプション指定時 (-S オプションは指定していないとき) の各ファイルに対する処理の内容を次の表に示します。

表 7-4 -i オプション指定時 (-S オプションは指定していないとき) の各ファイルに対する処理の内容

ファイル名	既存のファイルなし	既存のファイル種別		Datareplicator ファイルシステム領域※1	
		OS の通常ファイル	キャラクタ型スペシャルファイル※1	初期化状態	使っている状態
抽出情報キューファイル	OS の通常ファイル形式で作成される	再作成	何もしない※2	ファイルが割り当てられます。	ファイルが再割り当てされます。
抽出マスタステータスファイル			ヘッダ初期化※2	—	—
抽出サーバステータスファイル			ヘッダ初期化※2	ファイルが割り当てられます。	ファイルが再割り当てされます。
抽出マスタエラー情報ファイル			キャラクタ型スペシャルファイルでは作成できません。	—	—
抽出ノードマスタエラー情報ファイル			—	—	—
稼働トレースファイル			—	—	—
データ連動用連絡ファイル			ヘッダ初期化※2	ファイルが割り当てられます。	ファイルが再割り当てされます。

(凡例)

— : Datareplicator ファイルシステム領域に格納できません。

注※1

キャラクタ型スペシャルファイルと Datareplicator ファイルシステム領域は、UNIX 版 Datareplicator で使えます。

Windows 版 Datareplicator では「OS の通常ファイル」で示す処理が実行されます。

注※2

再作成する場合は、OSのコマンドを使ってください。

- -i オプションを指定しないで hdestart コマンドを実行すると、指定したオプションに従って、抽出側 Datareplicator のプロセスが起動します。オプションと起動プロセスの関係については、「表 7-7 実行する hdestart コマンドの一覧」を参照してください。既に抽出側 Datareplicator が稼働している場合に -i オプションを指定しないで hdestart コマンドを実行すると、指定したオプションに対応するプロセスのうち、停止しているプロセスだけが起動します。
- 抽出側 Datareplicator の状態によっては、hdestart コマンドを実行できない場合があります。抽出側 Datareplicator の状態と実行できる hdestart コマンドの関係を次の表に示します。

表 7-5 抽出側 Datareplicator の状態と実行できる hdestart コマンドの関係

抽出側 Datareplicator の状態	オプションの指定						
	なし	-i	-e	-s	-s -L	-R	-v
起動処理中	×	×	×	×	×	×	×
稼働中※	○	×	○	○	×	×	×
更新 SQL 出力中	×	×	×	×	○	×	×
停止処理中	×	×	×	×	×	×	×
停止中	○	○	○	○	○	○	○

(凡例)

- ：実行できます。
 - ×
- ×：実行できません（エラーになります）。

注※

抽出処理，又は送信処理が停止しているときに，hdestart -i 以外の hdestart コマンドを実行した場合，抽出処理と送信処理は，前回の続きから処理を再開します。

- hdestart コマンドを実行するとき，複数のオプションを組み合わせて指定できます。hdestart コマンドのオプションの組み合わせの関係を次の表に示します。

表 7-6 起動コマンドのオプションの組み合わせの関係

コマンドのオプション	-i	-e	-s	-s -L	-r	-R	-k	-v
-i	×	×	×	×	×	×	×	×
-e	×	×	○	×	○	×	×	○
-s	×	○	×	×	○	×	×	×
-s -L	×	×	×	×	○	×	×	×
-r	×	○	○	○	×	×	×	○

コマンドのオプション	-i	-e	-s	-s -L	-r	-R	-k	-v
-R	×	×	×	×	×	×	○	×
-k	×	×	×	×	×	○	×	×
-v	×	○	×	×	○	×	×	×

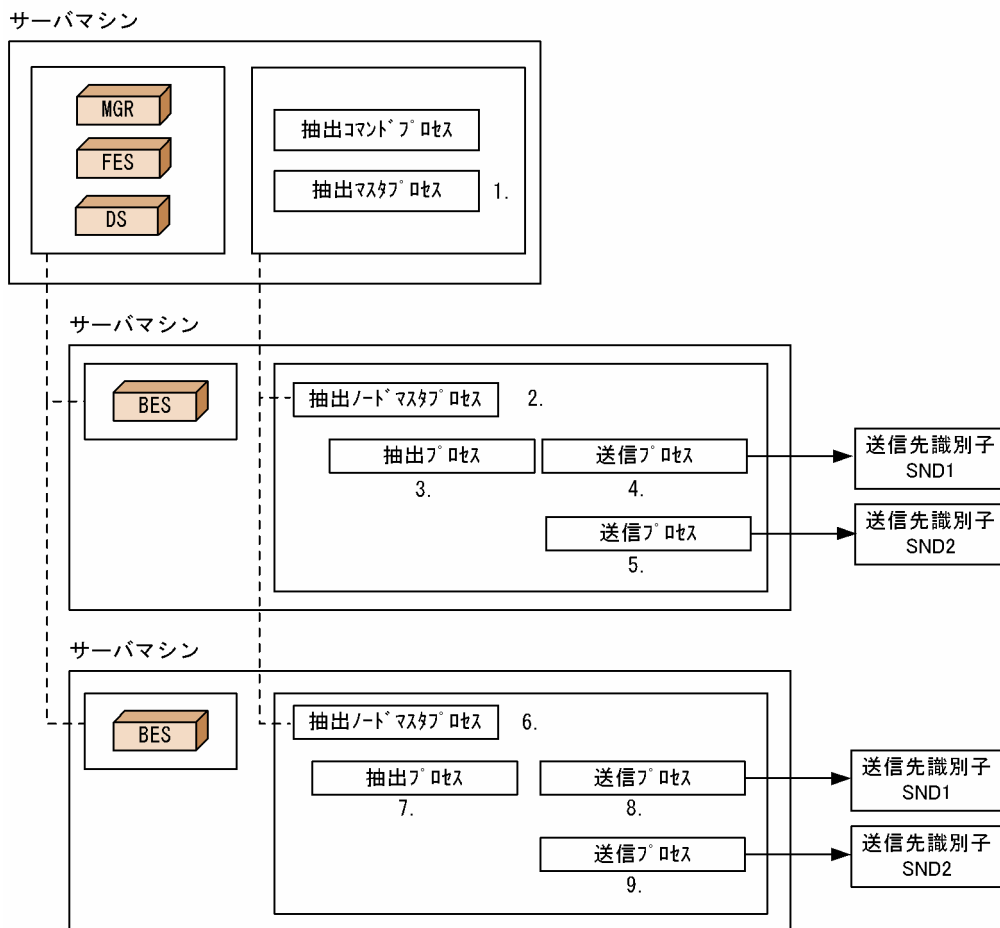
(凡例)

○：実行できます。

×：実行できません（エラーになります）。

- 障害が発生して、一部の抽出側のプロセスが停止した場合には、起動時と同じコマンドを実行してください。起動時と同じコマンドを実行することで、エラーで停止している抽出側のプロセスだけを再起動できます。この場合、正常に稼働しているプロセスに対しては、何も影響がありません。
- 抽出側 HiRDB がパラレルサーバの場合の、hdestart コマンドによる動作を「[図 7-1 抽出側 HiRDB がパラレルサーバの場合の、hdestart コマンドによる動作](#)」に、実行する hdestart コマンドの一覧を「[表 7-7 実行する hdestart コマンドの一覧](#)」に示します。「[図 7-1 抽出側 HiRDB がパラレルサーバの場合の、hdestart コマンドによる動作](#)」の数字は、「[表 7-7 実行する hdestart コマンドの一覧](#)」の数字と対応しています。

図 7-1 抽出側 HiRDB がパラレルサーバの場合の、hdestart コマンドによる動作



(凡例)

- MGR : HiRDBのシステムマネージャ FES : HiRDBのフロントエンドサーバ
- DS : HiRDBのディクショナリサーバ BES : HiRDBのバックエンドサーバ
- : 制御

表 7-7 実行する hdestart コマンドの一覧

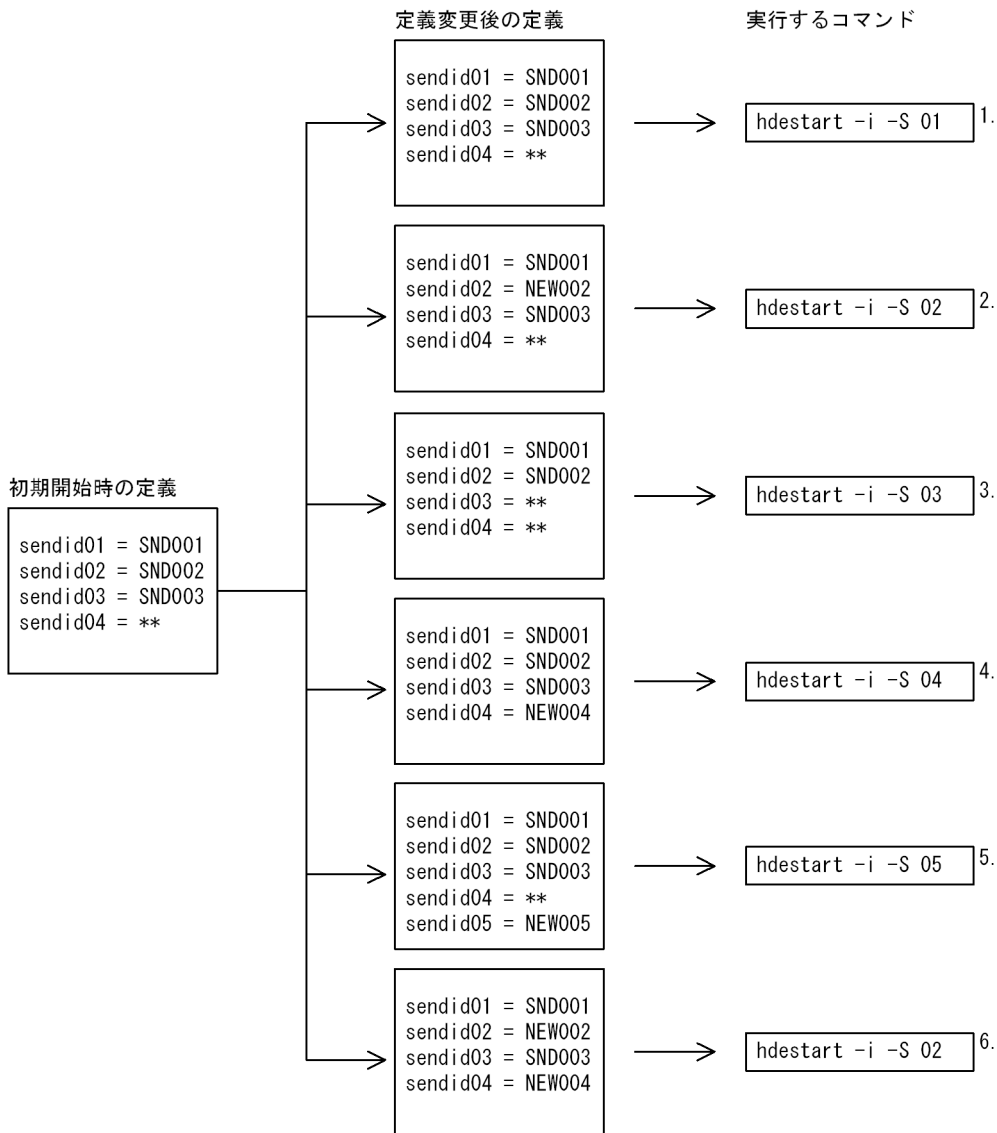
実行するコマンド	動作内容
hdestart -i	抽出側 Datareplicator の環境を初期化
hdestart	すべて (1~9) を起動
hdestart -e	抽出マスタプロセス (1), 抽出ノードマスタプロセス (2, 6), 抽出プロセス (3, 7) を起動
hdestart -s	抽出マスタプロセス (1), 抽出ノードマスタプロセス (2, 6), すべての送信プロセス ※ (4, 5, 8, 9) を起動
hdestart -s SND1	抽出マスタプロセス (1), 抽出ノードマスタプロセス (2, 6), 送信先識別子が SND1 の送信プロセス※ (4, 8) を起動
hdestart -e -s SND1	抽出マスタプロセス (1), 抽出ノードマスタプロセス (2, 6), 抽出プロセス (3, 7), 送信先識別子が SND1 の送信プロセス※ (4, 8) を起動

注※

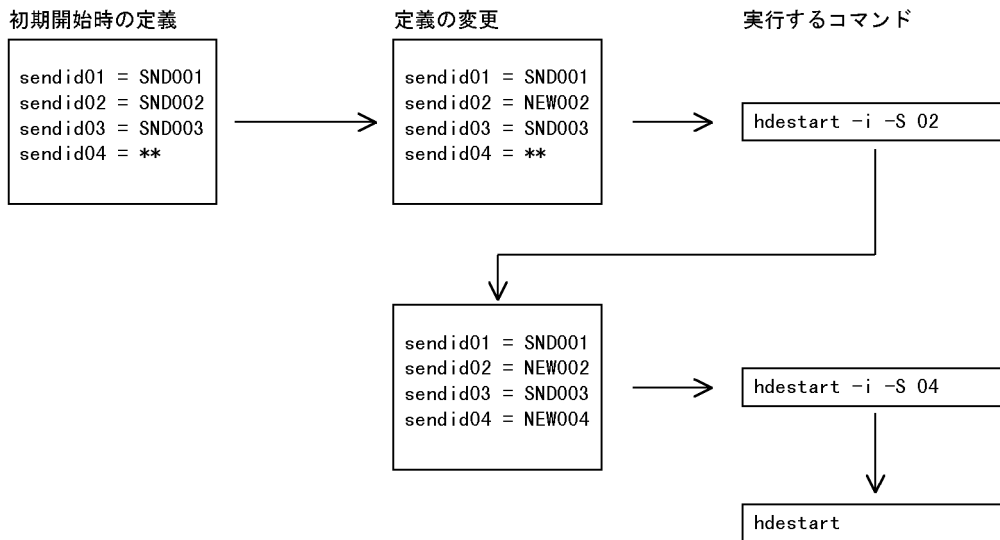
sendcontrol オペランドに sendmst を指定している場合、送信マスタプロセスと送信プロセスが起動します。

注意

- -e オプション、又は -s オプションを指定しない場合は、抽出処理と送信処理を連動して実行します。
- hdestart コマンドを実行する前に、/etc/inetd.conf という通信の環境変数を設定する必要があります。通信環境の設定方法については、「2.4 環境変数の設定 (UNIX)」又は「2.8 環境変数の設定 (Windows)」を参照してください。
- 次の場合は、hdestart コマンドを実行する前に hdeprep コマンドを実行する必要があります。
 - 初めてデータ連動を開始する場合
 - hdestart -i コマンドを実行した場合
 - 抽出側の表定義又は抽出定義を変更する場合
- hdestart コマンドの実行ユーザは、抽出側 HiRDB の実行ユーザと同じユーザグループに属しているユーザでなければなりません。
- hdestart -i コマンドは、抽出側 HiRDB が HiRDB Datareplicator 連携を実行していないときに指定してください。抽出側 HiRDB が HiRDB Datareplicator 連携を実行しているかどうかは、pdls コマンドで確認できます。HiRDB Datareplicator 連携を実行している場合には、pdrplstop コマンドで、HiRDB Datareplicator 連携を中止してください。
- 部分初期開始をしているときにエラーが発生したままの状態、別の送信先識別子番号を指定した部分初期開始、抽出定義のプリプロセス、及び正常起動はできません。
部分初期開始のエラー状態を解消するためには、エラーとなった送信先識別子番号を指定した部分初期開始を、エラーの原因を取り除いた後、再度実行して正常終了させるか、抽出環境全体を初期開始する必要があります。
- -i オプションの指定（部分初期開始の場合は除きます）で実行した場合には、処理を開始するかどうかの確認を応答する必要があります。
- 部分初期開始を指定できるのは、前回初期開始した時点で定義されていた送信先識別子番号だけで、新たに追加した送信先識別子番号は指定できません。
あらかじめ送信先が増えることが分かっているときは、将来使用する送信先識別子に**を指定しておくようにします。
部分初期開始時の指定例を次に示します。



1. 送信先識別子番号 01（送信先識別子 SND001）の部分初期開始をします。
 2. 定義変更後の送信先識別子番号 02（送信先識別子 NEW002）の部分初期開始をします。
 3. 定義変更によって欠番になった送信先識別子番号 03（送信先識別子 **）の部分初期開始をします。
 4. 新たに定義された送信先識別子番号 04（送信先識別子 NEW004）の部分初期開始をします。
 5. 定義変更によって追加された送信先識別子番号 05（送信先識別子 NEW005）は、初期開始時に定義されていないため、コマンドエラーになります。
 6. 指定された送信先識別子番号（送信先識別子 NEW002）以外の送信先識別子番号 004（送信先識別子 NEW004）が変更されているため、コマンドエラーになります。
- 2 か所以上の送信先識別子の部分初期開始をする場合は、1 か所ずつ定義の変更をして部分初期開始をしてください。すべての部分初期化が終了した後、正常開始をしてください。
- この例の場合の実行手順を次に示します。



- 部分初期開始時の実行手順を次に示します。
 1. 抽出側 Datareplicator が停止していることを確認します。
 2. テキストエディタで定義内容を変更します。
 3. hdestart -i -S コマンドで初期化対象の送信先識別子番号を指定して、部分初期開始をします。
 4. hdeprep コマンドを実行します。
 5. 抽出側 Datareplicator を正常開始します。
- 系切り替え構成の場合、初期化又は起動をする前に HiRDB Datareplicator が使用する資源（共用ディスク装置、ネットワーク）を活性化しておく必要があります。
- hdestart -i コマンド、及び hdestart コマンドが正常に完了したかどうかは、コマンドのリターンコードではなく、抽出マスタエラー情報ファイルに次のメッセージが出力されているかどうかで確認してください。
 - hdestart -i コマンドの場合：KFRB00504-I メッセージ
 - hdestart コマンドの場合：KFRB00502-I メッセージ

hdestart_n (抽出側 Datareplicator の部分起動)

機能

抽出側 Datareplicator を部分起動します。

このコマンドの動作概要を次に示します。

-b オプション	起動するプロセス			
	抽出マスタプロセス	抽出ノードマスタプロセス	抽出プロセス	送信プロセス
指定省略	○※1, ※2	—	—	—
指定あり	—	○※1, ※3, ※4	○※4	○※4

注※1

プロセスが既に稼働中の場合、コマンドは警告メッセージを出力し、正常終了します。

注※2

抽出マスタプロセス起動後に全抽出ノードマスタプロセス未起動状態が待ち時間 (-t オプション指定値) 経過した場合、抽出マスタプロセスは自動停止します。

注※3

抽出ノードマスタプロセスは起動後、抽出マスタプロセスと通信が可能になるまで待機します。指定の待ち時間 (-t オプション指定値) の間待機しても抽出マスタプロセスと通信できなかった場合、抽出ノードマスタプロセスは自動停止します。

注※4

起動するのは、-b オプションに指定したバックエンドサーバに対応した抽出ノードマスタプロセス、及びその配下の抽出プロセスと送信プロセスです。その他のバックエンドサーバに対応したプロセスに対しては何もしません。

形式

```
hdestart_n  [ -b サーバ名 -x 抽出マスタプロセス稼働ホスト名  
            -n サービス名 [ -c initial | continue ] [ -r ] ]  
            [ -t 待ち時間 ]
```

オプション

- -b サーバ名 -x 抽出マスタプロセス稼働ホスト名 -n サービス名 [-c initial | continue]

抽出側 Datareplicator の抽出ノードマスタプロセスを起動対象とする場合に指定します。このオプションを省略した場合、抽出マスタプロセスが起動対象となります。

-b サーバ名

～ 〈1～8文字の識別子〉

起動対象とする抽出ノードマスタプロセスに対応するバックエンドサーバ名を指定します。

-x 抽出マスタプロセス稼働ホスト名

～ 〈1～32 文字の識別子〉

抽出マスタプロセスに対して起動要求を発行するため、抽出マスタプロセスが稼働するホスト名（抽出側 Datareplicator 用に割り当てた IP アドレス引き継ぎありのホスト名）を指定します。指定するホスト名は、抽出システム定義の connection_accept_hostname オペランドで指定したホスト名と同じにする必要があります。

正しいホスト名を指定していない場合、hdestart_n コマンドは正常終了しますが、このコマンドによって起動した抽出ノードマスタプロセスは通信タイムアウトでエラー終了します。

-n サービス名

～ 〈1～64 文字の識別子〉

抽出マスタプロセスと抽出ノードマスタプロセス間で通信をするためのサービス名を指定します。指定するサービス名は、抽出システム定義の connection_accept_service オペランドで指定したサービス名と同じにする必要があります。

異なるサービス名を指定した場合、次に示す動作をします。

-n オプション指定値	-x オプション指定値	動作内容
抽出システム定義の mstservice オペランドと同じサービス名	抽出側 Datareplicator が存在するホスト名	-x オプションで指定したホスト、及び hdestart_n コマンドを実行したホストで、いったん抽出ノードマスタプロセスが起動し、このコマンドは正常終了します。その後、両者とも通信エラーでエラー終了します。
送信環境定義の hdeservice オペランドと同じサービス名	反映側 Datareplicator が稼働中のホスト名	hdestart_n コマンドが正常終了後、抽出ノードマスタプロセスは通信タイムアウトでエラー終了します。反映側 Datareplicator はポートチェック（詳細コード=16）でエラーとし、受信を拒否します。
上記以外		hdestart_n コマンドが正常終了後、抽出ノードマスタプロセスは通信タイムアウトでエラー終了します。

-c initial | continue

前回の抽出側 Datareplicator 停止時に、コマンド要求によって稼働状態に合った抽出、送信プロセスだけを起動する場合に指定します。

-c オプション指定による抽出、送信プロセスの動作を次に示します。

プロセス	前回の停止状態	-c オプション指定値	
		initial	continue
抽出プロセス	hdestop_n コマンドの-c オプションで continue を指定して停止	起動する	起動する
	エラー停止		起動する※1
	異常終了		
	上記以外		起動しない

プロセス	前回の停止状態	-c オプション指定値	
		initial	continue
送信プロセス	hdestop_n コマンドの-c オプションで continue を指定して停止		起動する
	エラー停止		起動する※1
	異常終了		
	上記以外		起動しない
更新 SQL 出力プロセス	—	起動しない※2	起動しない※2

注※1

エラー停止，異常終了については系切り替え処理の過程でエラーとなる場合を考慮し，-c オプションの指定に関係なく起動対象とします。

注※2

更新 SQL 出力プロセスは起動対象としないで，抽出，送信プロセスを起動対象とします。

-r

抽出側 Datareplicator 起動時に送信データ件数をリセットする場合に指定します。

このオプションは，抽出システム定義の send_counter_reset オペランドに false を指定した場合に有効となります。

• -t 待ち時間

～〈符号なし整数〉((1~3600))《300》(単位：秒)

最初の接続を確立するまでの待ち時間を指定します。抽出システム定義の cmwaittime オペランドの指定値以上の待ち時間を指定してください。このオプションは，-b オプションを指定するかどうかで内容が異なります。

-b オプション指定の有無による-t オプションの内容を次に示します。

-b オプション指定の有無	-t オプションの内容
あり	hdestart_n コマンドによって起動した抽出ノードマスタプロセスが，抽出マスタプロセスとの接続を確立するまでの待ち時間を指定します。指定時間が経過しても接続が確立できなかった場合，抽出ノードマスタプロセスはタイムアウトで終了します。
なし	hdestart_n コマンドによって起動した抽出マスタプロセスが，抽出ノードマスタプロセスのうちどれか一つと接続できるまでの待ち時間を指定します。指定時間が経過しても接続が確立できなかった場合，抽出マスタプロセスはタイムアウトで終了します。

このオプションの指定値は，プロセス起動後の最初の接続時だけ有効となります。接続が完了した後でほかのプロセスが停止した場合にも接続をリトライしますが，この場合には抽出システム定義の connection_retry_time オペランドで指定した時間リトライします。

規則

- hdestart_n コマンドを実行する場合、抽出システム定義の nodecontrol オペランドで server を指定する必要があります。nodecontrol オペランドで unit を指定した場合次に示す動作となります。
 - **-b オプション指定あり**
 hdestart_n コマンドは正常終了しますが、起動した抽出ノードマスタプロセスはタイムアウトで終了します。この場合、hdestop_n コマンドを使用することによってタイムアウトまで待たないで、抽出ノードマスタプロセスを終了させることができます。
 - **-b オプション指定なし**
 hdestart_n コマンドは正常終了しますが、抽出マスタプロセスが定義解析エラーでエラー終了します。
- hdestart_n コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- -b オプションの指定値と hdestart_n コマンド実行後の動作を次の表に示します。

表 7-8 -b オプションの指定値と hdestart_n コマンド実行後の動作

コマンド実行マシン		-b オプション指定値	hdestart_n コマンド実行後の動作
システムマネージャ ユニット	現用	指定なし	抽出マスタプロセスだけを起動し、コマンドは正常終了します。
		抽出対象 HiRDB に存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。 抽出マスタプロセスが要求受付エラー（サーバ名不正）のメッセージを出力し、抽出ノードマスタプロセスがエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。 該当する抽出ノードマスタプロセスは起動後、抽出サーバステータスファイルの入出力エラーでエラー終了します。
	抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	指定されたサーバ名に対応する抽出ノードマスタプロセスだけを起動し、コマンドは正常終了します。 抽出ノードマスタプロセスは抽出プロセス及び送信プロセスを起動します。	
待機	指定なし	コマンドは正常終了します。 抽出マスタプロセスは抽出マスタステータスファイルの入出力エラーでエラー終了します。	
	抽出対象 HiRDB に存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。 抽出マスタプロセスが要求受付エラー（サーバ名不正）のメッセージを出力し、抽出ノードマスタプロセスがエラー終了します。	

コマンド実行マシン		-b オプション指定値	hdestart_n コマンド実行後の動作
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。
		抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	該当する抽出ノードマスタプロセスは起動後、抽出サーバステータスファイルの入出力エラーでエラー終了します。
非システムマネージャユニット	現用	指定なし	抽出システム定義ファイルが見付からないため、コマンドはエラー終了します。
		抽出対象 HiRDB に存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。 抽出マスタプロセスが要求受付エラー（サーバ名不正）のメッセージを出力し、抽出ノードマスタプロセスがエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。 抽出マスタプロセスが起動し、該当する抽出ノードマスタプロセスに対して開始要求を発行した時点で、抽出ノードマスタプロセスは抽出サーバステータスファイルの入出力エラーでエラー終了します。
		抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	指定されたサーバ名に対応する抽出ノードマスタプロセスだけを起動し、コマンドは正常終了します。 抽出ノードマスタプロセスは抽出プロセス及び送信プロセスを起動します。
	待機	指定なし	抽出システム定義ファイルが見付からないため、コマンドはエラー終了します。
		抽出対象 HiRDB に存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。 抽出マスタプロセスが要求受付エラー（サーバ名不正）のメッセージを出力し、抽出ノードマスタプロセスがエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	抽出ノードマスタプロセスをいったん起動し、コマンドは正常終了します。該当する抽出ノードマスタプロセスは起動後、抽出サーバステータスファイルの入出力エラーでエラー終了します。

hdestate (抽出側 Datareplicator のステータス情報の取得)

機能

コマンドを実行した時点の抽出側 Datareplicator の状態を標準出力に出力します。出力される情報は、次のとおりです。

- **共通情報**
抽出側 Datareplicator 識別子, 総ノード数, 接続ノード数など, 共通な情報です。
- **ノード情報**
ノードホスト名, 抽出ノードマスタプロセスの状態など, ノードの状態を表す情報です。
- **抽出情報キューファイル情報**
抽出情報キューファイルの使用状態を表す情報です。
- **抽出処理情報**
抽出プロセスの状態, 抽出情報キューファイルの書き込み位置など, 抽出処理の状態を表す情報です。
- **送信処理情報**
送信先識別子, 送信プロセスの状態など, 送信処理の状態を表す情報です。

形式

```
hdestate [ -b HiRDBサーバ名 ] [ -s 送信先識別子 ]
```

オプション

- **-b HiRDB サーバ名** ~ 〈8文字の記号名称〉
hdestate コマンドの処理対象となる抽出環境を HiRDB サーバ名で指定します。指定したサーバ名に対応する抽出側 Datareplicator のステータス情報を取得します。抽出側 HiRDB がシングルサーバの場合も指定してください。
- **-s 送信先識別子**
特定の送信先識別子に対する抽出側 Datareplicator のステータス情報を取得する場合には、取得の対象とする送信先識別子を指定します。送信先識別子に**を指定した場合は、引数不正エラーとなります。

特定のサーバ名の中の、特定の送信先識別子に対するステータス情報を取得する場合には、-b オプションと-s オプションの両方を指定します。オプションの指定順序に制限はありません。

オプションをすべて省略すると、すべてのサーバ識別子とすべての送信先識別子に対するステータス情報を取得します。

規則

- hdestate コマンドを実行後、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- hdestate コマンドは、抽出側 Datareplicator が稼働中のときだけ実行できます。

- 抽出側 Datareplicator では、hdestate コマンドを実行した時点でステータス情報を取得し、標準出力に出力します。このため出力される情報は、実際の抽出側 Datareplicator の状態とは多少ずれが発生することがあります。
- 欠番指定された送信先識別子の情報は出力されません。

出力形式

```

*****
**      HiRDB Datareplicator extraction status information      **
**              Mon Mar 29 16:32:05 2004                      **
*****
----- Common information -----
hdeid = fa .....1.
Shmid : 3502 .....2.
PID   : hdemaster = 4579 .....3.
Total node count = 1   Connecting node count = 1 .....4.
nodecontrol : server .....5.

----- Node information -----
host name = host01 .....1.
current host name = : host001 .....2.
Communication status : online .....3.
Shmid : common = 1162  definf = 1163  msginf = 1164 .....4.
Semid : 923 .....5.
PID   : hdenodemst = 2247 .....6.

```



```

----- Server information -----
Server name = pbes01 .....1.
  Replication-node id : 01
  Semid                : 925
<TRANSMISSION QUEUE FILE INFORMATION>
wrap count = 0 .....2.
qufile001 status : using .....3.
qufile002 status : using
qufile003 status : using
qufile004 status : unused
qufile005 status : unused
max used ratio   : 2005/03/24 09:21:18 70% .....4.
current used ratio : 2005/03/24 09:51:18 10% .....5.

<EXTRACTION INFORMATION>
Status : active .....1.
PID    : hdecapture = 2248 .....2.
HiRDB system log overwrite : not overwrite .....3.
Queue write position : qufile003, offset = 4096 .....4.
Extract delay times  : 0000:00:05/0000:10:00 .....5.
<COMMUNICATION INFORMATION>
Send name = send01 .....1.
Status      : queue read .....2.
PID         : hdesender = 2249 .....3.
Service name : serv01 .....4.
Send host name : shost01 .....5.
Send system id : aa .....6.
Send delay times : 0000:00:10/0000:10:00 .....7.
Send count     : 3 .....8.
Send data Transmission count :
  transmission transaction count = 50 .....9.
  transaction count2 = 10 .....10.
  undetermined transaction count = 5 .....11.
  ins = 50, upd = 120, del = 30, purge = 2, event = 0 .....12.
  skipped update information count
  ins = 0, upd = 0, del = 0, purge = 0, event = 0 .....13.
Queue read position : qufile001, offset = 8192 .....14.
Queue current pos   : qufile001, offset = 12288 .....15.
Transmission inhibition information
  system id = a1, count = 5 .....16.
  system id = a2, count = 0
  system id = b1, count = 2

```

共通情報

1. hdeid = fa

抽出側 Datareplicator 識別子が出力されます。

2. Shmid : 3502

共用メモリ id が出力されます。抽出側 HiRDB がパラレルサーバの場合には、システムマネージャ側で取得した共用メモリ id が出力されます。

3. PID : hdemaster = 4579

抽出マスタープロセス id が出力されます。

4. Total node count = 1 ~ Connecting node count = 1

Total node count には、抽出マスタープロセスが管理する全サーバマシン数が出力されます。Connecting node count には、接続中のサーバマシンの数が出力されます。

5. nodecontrol : server

ノードマスタプロセスの制御方式（抽出システム定義の nodecontrol オペランド指定値）が出力されます。ノードマスタプロセスの制御方式は次のように出力されます。

unit : ユニット単位にノードマスタプロセスを起動

server : サーバ単位にノードマスタプロセスを起動

ノード情報

1. host name = host01

ノードに対応するホスト名が出力されます。

2. current host name = host0001

nodecontrol での出力が server の場合だけ、Communication status で出力される通信の状態によって次のように出力されます。

- Communication status の通信の状態が online の場合
current host name : ノードマスタプロセス起動先として現在認識しているホスト名
- Communication status の通信の状態が offline の場合
Server name : サーバ名

3. Communication status : online

抽出マスタプロセスと抽出ノードマスタプロセスとの間の通信状態が出力されます。通信の状態は次のように出力されます。

online : 接続中

offline : エラー切断中

状態がエラー切断中の場合には、これ以降の情報は出力されません。

4. Shmid : common = 1162 ~ msginf = 1164

抽出ノードマスタプロセスが取得した共用メモリ id が出力されます。共用メモリ id の種類は次のとおりです。

common : プロセス間連絡用共用メモリ id

definf : 定義情報格納用共用メモリ id

msginf : メッセージ情報格納用共用メモリ id

5. Semid : 923

抽出ノードマスタプロセスが取得したメッセージ出力排他用セマフォ id が出力されます。

6. PID : hdenodemst = 2247

抽出ノードマスタプロセスのプロセス id が出力されます。

抽出情報キューファイル情報

1. Server name = pbes01 ~ Semid : 925

Server name には、抽出側 HiRDB のサーバ名が出力されます。Replication-node id には、データ連動識別子が出力されます。Semid には、抽出ノードマスタプロセスが取得したセマフォ id が出力されます。

2. wrap count = 0

抽出側 Datareplicator の初期開始時から、抽出情報キューファイルを何順したかを出力します。

3. qufile001 ~ status : unused

抽出環境定義の qufile001 ~ 016 オペランドで指定した抽出情報キューファイルごとに、使用状況を出力します。使用状況は次のよう出力されます。

using : 使用中 (更新情報が格納されている場合は常に using)

unused : 未使用

4. max used ratio : 2005/03/24 09:21:18 70%

抽出情報キューファイル全体の使用率が最大値を示した日時と、その使用率が表示されます。

抽出情報キューファイルを初期化した時点、又は hdechgstatus コマンドで最大使用率をリセットした時点の表示形式は次のとおりです。

```
max used ratio      : ****/**/** **:***:*** 0%
```

5. current used ratio : 2005/03/24 09:51:18 10%

hdestate コマンドを実行した日時と、hdestate コマンドを実行した時点の抽出情報キューファイル全体の使用率が表示されます。

すべての送信先が縮退している状態のときは、使用率が「--」と表示されます。

抽出処理情報

1. Status : active

抽出プロセスの状態が出力されます。状態は次のよう出力されます。

active : 稼働中

not active : 停止中

not active (error) : エラー停止中

停止中、又はエラー停止中の場合には、抽出処理情報の 2. は出力されません。

2. PID : hdecapture = 2248

抽出プロセス id が出力されます。

3. HiRDB system log overwrite : not overwrite

抽出側 HiRDB のシステムログファイルでの、未抽出ログ上書き発生の有無が出力されます。

overwrite : 上書きあり

not overwrite : 上書きなし

4. Queue write position : ~ offset = 1024

抽出情報キューファイルの書き込み位置が出力されます。

qufilexxx には抽出環境定義で指定した qufile001 ~ 016 オペランドのうちの、更新情報を書き込んでいる抽出情報キューファイルに対応するものが出力されます。offset には、更新情報の書き込み位置が、ファイルの先頭からのオフセットで出力されます。

5. Extract delay times : 0000:00:05/0000:10:00

抽出側の HiRDB でコミットした時点から、その更新データを抽出情報キューファイルに書き込むまでの経過時間、及び抽出定義に指定した値が出力されます。表示形式は次のとおりです。

表示形式	説明
0000:00:00/定義値	抽出プロセスから何も抽出されていません。
****.***/****.**	抽出環境定義の extract_delay_limit_time オペランドに 0 が指定されています。
--:---:--/定義値	9999:59:59 を超えました。

送信処理情報

1. Send name = send01

抽出システム定義の sendid01 ~ 64 オペランドで指定した送信先識別子が出力されます。

2. Status : queue read

送信プロセスの状態が出力されます。状態は次のように出力されます。

init : 初期化処理中

queue read : 抽出情報キューファイルの読み込み中

data transmission : データ送信中

hold : 送信処理閉塞中

not active : 停止中

not active (error) : エラー停止中

状態が送信処理閉塞中、又はエラー停止中の場合、送信処理情報の 3. は出力されません。

3. PID : hdesender = 2249

送信プロセス id が出力されます。ただし、抽出システム定義の sendcontrol オペランドの設定が sendmst の場合は、常に 0 になります。更新 SQL 出力処理が起動している場合には、更新 SQL 出力プロセス id が出力されます。

4. Service name : serv01

送信サービス名が出力されます。

5. Send host name : shost01

送信先ホスト名が出力されます。

6. Send system id : aa

送信先の反映側 Datareplicator 識別子が出力されます。

7. Send delay times : 0000:00:10/0000:10:00

抽出側の HiRDB でコミットした時点から、その更新データを反映側システムに送信して受信が完了するまでの経過時間、及び送信定義に指定した値が出力されます。表示形式は次のとおりです。

表示形式	説明
0000:00:00/定義値	送信プロセスから何も送信されていません。

表示形式	説明
****.***/****.**	送信環境定義の send_delay_limit_time オペランドに 0 が指定されています。
--:---:--/定義値	9999:59:59 を超えました。
表示なし	更新 SQL 出力機能を使用しています。

8. send count : 3

送信処理の開始時からの送信処理の回数が出力されます。

9. transmission transaction count = 50

送信処理の開始時からの送信トランザクション数が出力されます。抽出システム定義の send_sounter_reset オペランドに false を指定している場合、抽出側 Datareplicator を再開しても前回起動時の内容が引き継がれます。

10. transaction count2 = 10

抽出側 DB を更新したときのトランザクションとは別に、抽出側 Datareplicator が作成・送信したトランザクション数が出力されます。transmission transaction count の送信トランザクション数には含まれません。

例えば、WITHOUT ROLLBACK オプションを指定している表は、1 回の更新を 1 トランザクションとして扱います。このため、WITHOUT ROLLBACK オプションを指定している表と指定していない表を同じトランザクション内で更新しても、反映側 Datareplicator には別々のトランザクションとして送信・反映されます。このときに、抽出側 Datareplicator が作成・送信した、WITHOUT ROLLBACK オプションを指定している表のトランザクション数が出力されます。

11. undetermined transaction count = 5

送信処理の開始時からの未解決トランザクション数が出力されます。

12. ins = 50 ~ event = 0

送信処理開始時からの操作別の送信データ数が出力されます。抽出システム定義の send_counter_reset オペランドに false を指定している場合、抽出側 Datareplicator を再開しても前回起動時の内容が引き継がれます。

操作別の送信データ数は、次のよう出力されます。

ins : insert データ数

upd : update データ数

del : delete データ数

purge : purge table データ件数

event : イベント発行数

13. ins = 0 ~ event = 0

WITHOUT ROLLBACK オプションを指定している表のデータ連動時、update 以外の更新については送信をスキップします。このスキップに関する、送信処理開始時からの操作別のスキップデータ数が出力されます。

操作別のスキップデータ数は、次のよう出力されます。

ins : insert データ数
upd : update データ数
del : delete データ数
purge : purge table データ件数
event : イベント発行数

14. Queue read position : ~ offset = 33792

抽出情報キューファイルの読み込み位置が出力されます。

qufilexxx には抽出環境定義で指定した qufile001 ~ 016 オペランドのうち、更新情報を読み込んである抽出情報キューファイルに対応するものが出力されます。offset には、更新情報の読み込み位置が、ファイルの先頭からのオフセットで出力されます。

更新 SQL 出力処理の場合、Queue read position の値は更新されません。

15. Queue current pos

抽出情報キューファイルの読み込み位置が出力されます。更新 SQL 出力処理が起動中の場合に表示されます。

16. system id = a1 ~ count = 5

更新情報の送信抑止情報が出力されます。system id には、更新情報の受信元システム識別子が出力されます。count には、送信抑止データ数が出力されます。

補 足

- 抽出情報キューファイル情報の 2., 送信処理情報の 8., 11. で出力される値は、4294967295 を超えると 0 に戻ります。
 - 送信処理情報の 9., 12., 13. で出力される値は、18446744073709551615 を超えると 0 に戻ります。
 - 送信処理情報の 7., 8., 9., 11., 15. で出力される情報は、抽出側 Datareplicator の起動単位でリセットされます。
 - 送信処理情報の 9., 10., 12., 13. で出力される情報は、次の契機でリセットされます。
 - 抽出側 Datareplicator の初期化（部分初期化を含む）又は起動※
 - 送信環境定義の eventcntreset オペランドに指定したイベントの検知
- 注※ 抽出システム定義の send_counter_reset オペランドに false を指定している場合
- 繰返し列に対して、SET 句、ADD 句又は DELETE 句が混在する UPDATE の更新情報を送信すると、送信処理情報の 12. で出力される update データ数は実際の更新件数よりも多くなります。繰返し列の場合で、12. で出力される update データ数と実際の更新件数を一致させたいときは、句ごとに分けて UPDATE で更新してください。

hdestop (抽出側 Datareplicator の終了)

機能

抽出側 Datareplicator を強制終了モードで終了します。正常終了モードで終了する場合には、抽出システム定義の `syncterm` オペランドに `true` を設定します。

形式

```
hdestop  [ -t sendterm ]
          [ -e ]
          [ -s [ 送信先識別子 [ { {, 送信先識別子 } } ...] ] ]
          [ -w ]
```

オプション

- `-t sendterm`

送信処理を停止する場合に、強制的に処理を打ち切って停止するときに指定します。このオプションは、送信処理が稼働中で、かつ稼働中の送信処理の停止を指示するオプション指定の場合 (`-e` 及び `-s` オプションのどちらの指定もないとき、又は `-s` オプションの指定があるとき) にだけ有効になります。`-t` オプションを省略すると、抽出側 Datareplicator は、送信間隔中に抽出情報キューファイルから読み込んだ更新情報を送信し終わってから、送信処理を停止します。

- `-e`

HiRDB システムログファイルからの更新情報の抽出処理だけを停止するときに指定します。

- `-s [送信先識別子 [{ {, 送信先識別子 } } ...]]`

送信先への送信処理だけを停止するときに指定します。

- **送信先識別子**

特定の送信先識別子の送信処理だけを停止するときに、抽出システム定義で指定した送信先識別子を指定します。複数の送信先識別子を指定する場合には、送信先識別子と送信先識別子の間に、空白を入力しないでください。送信先識別子に `**` を指定した場合は、引数不正エラーとなります。送信先識別子を省略すると、すべての送信先識別子への送信処理が終了します。

- `-w`

抽出側 Datareplicator のマスタプロセスが停止するまで待った後で、`hdestop` コマンドの処理を終了するときに指定します。

計画系切り替えで使用する Datareplicator 停止シェルを使用するときのように Datareplicator 本体が完全に停止するのを待ってコマンドを終了させる必要がある場合に、このオプションを指定してください。

`-w` オプションなしのときは、`hdestop` コマンドはマスタプロセスに停止要求を送信完了すると終了します (マスタプロセスはコマンドと非同期に終了します)。

`-w` オプションありのときは、`hdestop` コマンドはマスタプロセスに停止要求を送信完了後、マスタプロセスの終了まで待ってから終了します (コマンド終了時には、マスタプロセスも終了しています)。

hdestop コマンドにオプションを複数付けるときに使える組み合わせを次に示します。

hdestop コマンド のオプション	-t	-e	-s	-w
-t	×	○	○	○
-e	○	×	○	×
-s	○	○	×	×
-w	○	×	×	×

(凡例)

○：オプションを組み合わせて実行できます。

×：オプションを組み合わせて実行できません（エラーメッセージが出力されます）。

規則

- 抽出側 Datareplicator の終了方法には、抽出側 HiRDB の正常終了に連動して自動的に終了する方法と、hdestop コマンドによって終了する方法の 2 種類があります。抽出側 Datareplicator の終了方法と終了モードとの関係を次の表に示します。

表 7-9 抽出側 Datareplicator の終了方法と終了モードとの関係

終了方法	終了モード	終了時の動作内容
抽出側 HiRDB の正常終了に連動した終了	正常終了	次の条件をすべて満たした時点で、抽出側 Datareplicator が終了します。 <ul style="list-style-type: none"> 抽出側 HiRDB の正常終了を検知しました。 検知した正常終了の後に、抽出側 HiRDB でのトランザクションが発生していません。 抽出プロセスが正常に稼働し、システムログファイルのすべての更新情報の、抽出情報キューファイルへの抽出が完了しています。 送信プロセスが正常に稼働し、抽出情報キューファイルのすべての更新情報の、反映側システムへの送信が完了している（送信遅延開始の場合は、抽出プロセスが抽出側 HiRDB の正常終了を検知する前に、送信プロセスを開始しないと、抽出情報キューファイルの更新情報は送信されない）抽出側 HiRDB の正常終了に連動して抽出側 Datareplicator を終了する場合には、抽出システム定義の syncterm オペランドを true に設定する必要があります。
hdestop コマンドの実行	強制終了	hdestop コマンドを実行した時点で実行中の抽出と送信が完了してから、抽出側 Datareplicator が終了します。

- 抽出システム定義の syncterm オペランドを true と設定している場合に hdestop コマンドを実行すると、抽出側 Datareplicator は hdestop コマンドに従い、強制終了モードで終了します。
- hdestop コマンドは、抽出側 Datareplicator が稼働中のときだけ実行できます。抽出側 Datareplicator が稼働中以外の人に hdestop コマンドを実行すると、エラーになります。
- hdestop コマンドを実行後、正常終了すると 0 が返されます。異常終了すると 1 が返されます。

注意

- 既に停止状態の抽出処理又は送信処理に対して停止要求があった場合、停止処理は実行されないで、稼働中の抽出処理又は送信処理に対してだけ停止処理が実行されます。なお、この場合 `hdestop` コマンドは正常終了することとします（ただし、既に停止状態であった内容をエラー情報ファイルに出力します）。
- 抽出処理又は送信処理の個別停止を繰り返し実行した結果、すべての抽出処理及び送信処理が停止した場合は、抽出側 `Datareplicator` を停止します。
- 抽出システム定義の `sendcontrol` オペランドで `sendmst` を指定して、特定の送信先だけの個別停止をした場合、`hdestop` コマンドの受け付け時点で対象外の送信先に対する送信間隔が打ち切れ、送信処理を開始します。
- 抽出システム定義の `nodecontrol` オペランドでの `server` 指定時、抽出ノードマスタプロセスが稼働中であつ抽出マスタプロセスが停止中の場合、`hdestop` コマンドは `KFRB04411-E` メッセージを出力しエラー終了します。稼働中の抽出ノードマスタプロセスは停止しません。

hdestop_n (抽出側 Datareplicator の部分終了)

機能

抽出側 Datareplicator を部分終了します。

このコマンドの動作概要を次に示します。

-b オプション	停止するプロセス			
	抽出マスタプロセス	抽出ノードマスタプロセス	抽出プロセス	送信プロセス
指定省略	○*1*2	—	—	—
指定あり	—	○*1*3*4	○*4	○*4

注※1

プロセスが稼働中でない場合、コマンドはエラー終了します。

注※2

抽出マスタプロセス停止後、稼働中の抽出ノードマスタプロセスは、connection_retry_time オペランドに指定された待ち時間の間、抽出マスタプロセスの再起動を待機します。指定時間経過しても抽出マスタプロセスとの通信が行えなかった場合、抽出ノードマスタプロセスは自動停止します。

注※3

全抽出ノードマスタプロセスが停止した後、稼働中の抽出マスタプロセスは connection_accept_waittime オペランドに指定された待ち時間の間、抽出ノードマスタプロセスの再起動を待機します。指定時間経過しても、どの抽出ノードマスタプロセスとも通信が行えなかった場合、抽出マスタプロセスは自動停止します。

注※4

停止するのは、-b オプションに指定したバックエンドサーバに対応した抽出ノードマスタプロセス及びその配下の抽出プロセスと送信プロセスです。その他のバックエンドサーバに対応したプロセスに対しては何もしません。

形式

```
hdestop_n [-b サーバ名 [-t sendterm] [-c initial | continue]] [-w]
```

オプション

- -b サーバ名 [-t sendterm] [-c initial | continue]

抽出側 Datareplicator の抽出ノードマスタプロセスを停止対象とする場合に指定します。このオプションを省略した場合、抽出マスタプロセスが停止対象となります。

-b サーバ名

～ 〈1～8 文字の識別子〉

停止対象とする抽出ノードマスタプロセスに対応するバックエンドサーバ名を指定します。

停止対象とする抽出マスタプロセス、又は抽出ノードマスタプロセスがない場合、`hdestop_n` コマンドはエラー終了します。

-t sendterm

強制的に送信処理を打ち切って停止するときに指定します。このオプションは、停止対象となる抽出ノードマスタプロセス配下の送信プロセスが稼働中である場合にだけ有効となります。

このオプションを省略した場合、送信間隔中に抽出情報キューファイルから読み込んだ更新情報の送信が終了してから停止します。

-c initial | continue

次の抽出側 Datareplicator 起動時に、現在コマンド要求によって稼働状態にある抽出、送信プロセスだけを起動する場合にこのオプションを指定して、停止します。

-c オプション指定による抽出、送信プロセスの動作については、`hdestart_n` コマンド（抽出側 Datareplicator の部分起動）の -c オプションを参照してください。

• -w

停止対象のプロセスの停止後、`hdestop_n` コマンドの処理を終了するときに指定します。

計画系切り替えで使用する Datareplicator 停止シェルを使用するときのように、Datareplicator が完全に停止してから `hdestop_n` コマンドを終了させる場合に指定します。このオプションを指定した場合、`hdestop_n` コマンド終了時には停止対象のプロセスも終了しています。

このオプションを省略した場合、`hdestop_n` コマンドは停止対象のプロセスに停止を要求した直後に終了します（停止対象のプロセスは、`hdestop_n` コマンドとは非同期に終了します）。

規則

- `hdestop_n` コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- `hdestop_n` コマンドを実行する場合、抽出システム定義の `nodecontrol` オペランドに `server` を設定する必要があります（ただし、正常実行できる場合もあります。詳細については、`hdestart_n` コマンド（抽出側 Datareplicator の部分起動）の規則を参照してください）。
- -b オプションの指定値と `hdestop_n` コマンド実行後の動作を次の表に示します。

表 7-10 -b オプションの指定値と `hdestop_n` コマンド実行後の動作

コマンド実行マシン		-b オプション指定値	<code>hdestop_n</code> コマンド実行後の動作
システムマネージャ ユニット	現用	指定なし	抽出マスタプロセスだけを停止し、 <code>hdestop_n</code> コマンドは正常終了します。
		抽出対象 HiRDB に存在しないサーバ名	エラー情報ファイルが見付からないため、 <code>hdestop_n</code> コマンドはエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	

コマンド実行マシン		-b オプション指定値	hdestop_n コマンド実行後の動作
	待機	抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	該当する抽出ノードマスタプロセスだけを停止し、hdestop_n コマンドは正常終了します。
		指定なし	停止対象となる抽出マスタプロセス、又は抽出ノードマスタプロセスが存在しないため、hdestop_n コマンドはエラー終了します。
		抽出対象 HiRDB に存在しないサーバ名	エラー情報ファイルが見付からないため、hdestop_n コマンドはエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	
		抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	エラー情報ファイルが見付からない、又は停止対象となるプロセスが存在しないため hdestop_n コマンドはエラー終了します。
非システムマネージャユニット	現用	指定なし	抽出システム定義ファイルが見付からないため、コマンドはエラー終了します。
非システムマネージャユニット	現用	抽出対象 HiRDB に存在しないサーバ名	エラー情報ファイルが見付からないため、hdestop_n コマンドはエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	
		抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	
		待機	指定なし
	待機	抽出対象 HiRDB に存在しないサーバ名	エラー情報ファイルが見付からないため、hdestop_n コマンドはエラー終了します。
		抽出対象 HiRDB に存在するが、システムマネージャユニットには存在しないサーバ名	
		抽出対象 HiRDB に存在し、かつシステムマネージャユニットに存在するサーバ名	

hdscnvedt (文字コード変換用マッピングテーブルの編集)

機能

文字コード変換用マッピングテーブルを編集します。外字の変換方法を変更するときに使うコマンドです。

hdscnvedt コマンドには、次の機能があります。

- **文字コード変換用マッピングテーブルの更新**
文字コード変換用マッピングテーブルに定義されている外字の変換方法を変更します。
- **文字コード変換用マッピングテーブルの参照**
文字コード変換用マッピングテーブルに定義されている外字の変換方法を出力します。この出力結果を利用して、文字コード変換用マッピングテーブルを更新できます。
- **外字マッピングファイルの移行**
Datareplicator 05-02 以前で作成した外字マッピングファイルの内容を、文字コード変換用マッピングテーブルに反映します。

形式

文字コード変換用マッピングテーブルの更新

```
hdscnvedt  -w  
            -f 変換元文字コードセット  
            -t 変換先文字コードセット  
            -d 変換定義ファイル名
```

文字コード変換用マッピングテーブルの参照

```
hdscnvedt  -r  
            -f 変換元文字コードセット  
            -t 変換先文字コードセット  
            -o 出力ファイル名  
            [-s 出力開始コード]  
            [-e 出力終了コード]
```

外字マッピングファイルの移行

```
hdscnvedt  -c  
            -f 変換元文字コードセット  
            -t 変換先文字コードセット  
            -g 外字マッピングファイル名
```

オプション

- **-w**
文字コード変換用マッピングテーブルを更新するときに指定します。
このオプションによって、文字コード変換用マッピングテーブル中の外字に相当する範囲の文字コードの対応を変更できます。

- -r

文字コード変換用マッピングテーブルを参照するときに指定します。

- -C

既存の外字マッピングファイルの定義内容を、文字コード変換用マッピングテーブルに移行するときに指定します。Datareplicator 05-02 以前で作成した外字マッピングファイルの内容を、文字コード変換用マッピングテーブルに反映します。

- -f 変換元文字コードセット

変換元の文字コードセットを次のニモニックで指定します。

eck78 : EBCDIC/KEIS78

eck83 : EBCDIC/KEIS83

ekk78 : EBCDIK/KEIS78

ekk83 : EBCDIK/KEIS83

sjis : JIS8/シフト JIS

euc : EUC

ucs2 : UCS2*

注※ 文字コードセットが UTF-8 の場合は、UCS2 コードセットを指定してください (UTF-8 の場合、Datareplicator のマッピングテーブルとしては UCS2 コードセットとして管理しています。UCS2 から UTF-8 へのエンコードは実際のコード変換時に実行します)。

- -t 変換先文字コードセット

変換先の文字コードセットを次のニモニックで指定します。

ekk83 : EBCDIK/KEIS83

sjis : JIS8/シフト JIS

euc : EUC

ucs2 : UCS2*

注※ 文字コードセットが UTF-8 の場合は、UCS2 コードセットを指定してください (UTF-8 の場合、Datareplicator のマッピングテーブルとしては UCS2 コードセットとして管理しています。UCS2 から UTF-8 へのエンコードは実際のコード変換時に実行します)。

-f オプションと-t オプションで指定できる文字コードセットの組み合わせを次に示します。

-f オプション	-t オプション						
	eck78	eck83	ekk78	ekk83	sjis	euc	ucs2
eck78	×	×	×	×	○	○	○
eck83	×	×	×	×	○	○	○
ekk78	×	×	×	×	○	○	○
ekk83	×	×	×	×	○	○	○
sjis	○	○	○	○	×	○	○

-f オプション	-t オプション						
	eck78	eck83	ekk78	ekk83	sjis	euc	ucs2
euc	○	○	○	○	○	×	○
ucs2	×	×	×	×	○	○	×

(凡例)

○：指定できます。

×：指定できません（指定した場合は、KFRB04104-E メッセージが出力されます）。

- **-d 変換定義ファイル名**

～ 〈1～127 バイトのパス名〉

文字コード変換用マッピングテーブルの更新内容を定義した変換定義ファイル名を、絶対パス名又は相対パス名で指定します。相対パス名で指定する場合は、カレントディレクトリからの相対パスとして認識されます。

変換定義ファイルの定義内容については、「変換定義ファイルの形式」を参照してください。

- **-o 出力ファイル名**

～ 〈1～127 バイトのパス名〉

文字コード変換用マッピングテーブルの参照結果を出力するファイル名を、絶対パス名又は相対パス名で指定します。相対パス名で指定する場合は、カレントディレクトリからの相対パスとして認識されません。

文字コード変換用マッピングテーブルの参照結果は、変換定義ファイルの定義フォーマットと同じ形式で、変換元文字コードの昇順に出力ファイルに対して出力します。

出力対象は変換先コードが定義されているコードだけとし、変換先が定義されていないコードについては出力されません。

- **-s 出力開始コード**

-r オプションを指定したときだけ指定できるオプションで、参照を開始する変換元文字コードを 16 進数値文字列で指定します。

このオプションを省略した場合は、文字コード変換用マッピングテーブルの先頭から参照を開始します。出力開始コードは、次の規則に従って指定してください。

- 1 バイトコードは 2 文字、2 バイトコードは 4 文字、3 バイトコードは 6 文字（変換元文字コードセットが EUC、又は EUC-HJ の場合だけ）で指定してください。それ以外の文字数の場合はエラーになります。
- 変換元文字コードセットが EUC、又は EUC-HJ の 3 バイトコードを指定するときは、先頭を必ず 8f にしてください。それ以外の文字を指定した場合はエラーになります。
- -e オプションで指定した出力終了コードよりも小さい値を指定してください。出力終了コードよりも大きい値を指定した場合はエラーになります。

- **-e 出力終了コード**

-r オプションを指定したときだけ指定できるオプションで、参照を終了する変換元文字コードを 16 進数値文字列で指定します。

このオプションを省略した場合は、文字コード変換用マッピングテーブルの終端まで参照します。

出力終了コードは、次の規則に従って指定してください。

- 1 バイトコードは 2 文字、2 バイトコードは 4 文字、3 バイトコードは 6 文字（変換元文字コードセットが EUC の場合だけ）で指定してください。それ以外の文字数を指定した場合はエラーになります。
- 3 バイトコードを指定するときは、先頭を必ず 8f にしてください。それ以外の文字を指定した場合はエラーになります。
- -s オプションで指定した出力開始コードよりも大きい値を指定してください。出力開始コードよりも小さい値を指定した場合はエラーになります。
- -g 外字マッピングファイル名
～ 〈1～127 バイトのパス名〉

移行対象とする外字マッピングファイル名を絶対パス名又は相対パス名で指定します。相対パス名で指定する場合は、カレントディレクトリからの相対パスとして認識されます。

変換定義ファイルの形式

変換定義ファイルには、文字コード変換用マッピングテーブル中の更新したい変換元文字コードと変換先文字コードを次の形式で定義します。

変換元文字コード，変換先文字コード

変換定義ファイルは、次の規則に従って指定してください。

- 1 行に 1 エントリで定義してください。
- 変換元文字コード及び変換先文字コードは、16 進数値文字列（a～f については、大文字と小文字のどちらでも指定できます）で指定してください。
- #以降はコメントとみなされます。変換の定義の後にコメントを書く場合は、定義と#の間に一つ以上の空白文字、又はタブを入れる必要があります。
- 文字コードの指定は、1 バイトコードは 2 文字、2 バイトコードは 4 文字、3 バイトコードは 6 文字（変換元文字コードセットが EUC、又は EUC-HJ の場合だけ）で指定してください。それ以外の文字数を指定した場合はエラーになります。
- 3 バイトコードを指定するときは、先頭を必ず 8f にしてください。それ以外の文字を指定した場合はエラーになります。
- 変換先文字コードの指定以降に出現する最初の空白文字以降は無視されます。
- 各コードの前後にある空白文字及びタブは無視されます。

規則

- 指定できる外字の範囲は、次のとおりです。

- KEIS コード：第 1 バイトが 41～A0，第 2 バイトが A1～FE
- シフト JIS コード：第 1 バイトが F0～FC，第 2 バイトが 40～FC（ただし，7F を除く）
- EUC：第 1 バイトが 8F，第 2 バイトが A1～FE，第 3 バイトが A1～FE
- UCS2：第 1 バイトが E0～F8，第 2 バイトが 00～FF
- このコマンドは，スーパーユーザで実行してください。
- 同じオプションを複数指定した場合は，オプション指定エラーになります。
- 変換元と変換先に同じ文字コードセットを指定した場合はエラーになり，文字コード変換用マッピングテーブルは更新されません。
- コマンドの実行途中でエラーが発生した場合は，文字コード変換用マッピングテーブルは更新されません。
- 文字コード変換用マッピングテーブルが更新された場合，更新後の文字コード変換用マッピングテーブルは，ユーザ作成の文字コード変換用マッピングテーブルとして，インストールディレクトリ/lib/usermap に格納されます。文字コード変換用マッピングテーブルは削除しないでください。インストールディレクトリ/lib/usermap 下に文字コード変換用マッピングテーブルがない場合には，Datareplicator は文字コードの変換規則に従って外字コードも変換します。Datareplicator の文字コードの変換規則については，「4.3.5 文字コードの対応の設計」を参照してください。
- データ連動の開始後に，文字コード変換用マッピングテーブルを更新した場合，更新によってマッピングキーの文字コードの変換結果が異なるようなときには，反映処理が正常に実行できなくなります。このため，データ連動開始後に文字コード変換用マッピングテーブルを更新する場合には，十分な注意が必要です。
- 外字マッピングファイルからの移行時（-c オプション）に指定できる変換元及び変換先文字コードセットの組み合わせを次の表に示します。これ以外の組み合わせを指定した場合はエラーになります。

表 7-11 外字マッピングファイルからの移行時（-c オプション）に指定できる変換元及び変換先文字コードセットの組み合わせ

変換元文字コードセット	変換先文字コードセット
EBCDIC/KEIS78 EBCDIC/KEIS83 EBCDIK/KEIS78 EBCDIK/KEIS83	JIS8/シフト JIS
JIS8/シフト JIS	EBCDIC/KEIS78 EBCDIC/KEIS83 EBCDIK/KEIS78 EBCDIK/KEIS83
	EUC
EUC	JIS8/シフト JIS

hdschgstatus (反映側 Datareplicator の状態変更)

機能

反映情報キューファイルの最大使用率を変更します。このコマンドで変更した情報は、反映側 Datareplicator を停止、又は起動しても保持されます。

形式

```
hdschgstatus -d データ連動識別子
              -c reset -k max_ratio
```

オプション

- -d データ連動識別子
変更対象のデータ連動識別子を指定します。
- -c reset
-k で指定した情報をリセットするときに指定します。
- -k max_ratio
反映情報キューファイルの最大使用率を変更するときに指定します。

規則

hdschgstatus コマンドを実行後、正常終了すると 0 が返されます。異常終了すると 1 が返されます。

注意

hdschgstatus コマンドは、反映側 Datareplicator が稼働中のときだけ実行できます。停止中に実行するとエラーになります。

hdsemergencystop (反映側 Datareplicator の強制終了)

機能

反映側 Datareplicator の状態に関係なく、反映側 Datareplicator の全プロセスを強制終了するコマンドです。

hdsemergencystop コマンドは、hdsstop -t force コマンドで反映側 Datareplicator を停止できなかった場合に使用してください。

Windows 版 Datareplicator だけで使用できるコマンドです。UNIX 版 Datareplicator では使用できません。

形式

```
hdsemergencystop
```

規則

- hdsemergencystop コマンドを実行後、正常終了すると、0 が返されます。異常終了すると、1 が返されます。
- hdsemergencystop コマンドが異常終了した場合は、再度 hdsemergencystop コマンドを実行してください。数回実行しても異常終了する場合は、OS がコマンドを受け付けられない状況になっていると考えられます。この場合は、OS を再起動してください。

hdsfcopy (反映側現用ファイルの複写)

機能

反映側システムのファイルの二重化で障害が発生して一方の物理ファイルが使用できない状態 (HOLD 状態) になった場合、正常なファイルを障害の発生したファイルに複写するコマンドです。

これによって、障害が発生したファイルが使用可能な状態 (ACTIVE 状態) に回復します。

形式

```
hdsfcopy -f 複写元物理ファイル名 -t 複写先物理ファイル名
```

オプション

- -f 複写元物理ファイル名
複写元の (正常な) 物理ファイル名を絶対パスで指定します。
- -t 複写先物理ファイル名
複写先の (障害の発生した) 物理ファイル名を絶対パスで指定します。

規則

- 現用ファイル複写コマンドは、回復対象のファイルが存在するノード上で実行します。
- 現用ファイル複写コマンドは、Datareplicator が停止している時に実行可能です。
- 複写元物理ファイル名、又は複写先物理ファイル名に二重化定義で正しく定義されていない場合、コマンドエラーになります。
- 複写元の物理ファイルの状態が Active 状態でない場合、コマンドエラーになります※。
- 複写先の物理ファイルの状態が Hold 状態でない場合、コマンドエラーとなります※。

注※

物理ファイルの状態は、反映側二重化ファイルの状態表示コマンド (hdsfstate) で確認できます。
詳細については、[hdsfstate](#) コマンドを参照してください。

- 複写するファイルをラージファイルとして運用している場合、コマンドの実行時間が長くなることがあります。詳細については、「[6.11.2 ラージファイル運用時のコマンド実行時間の見積もり](#)」を参照してください。

hdsfmkfs (Datareplicator ファイルシステム領域の初期化)

機能

Datareplicator ファイルシステム領域を初期化します。

Datareplicator ファイルシステム領域は、UNIX 版 Datareplicator で使えます。Windows 版 Datareplicator ではこのコマンドは使えません。

Datareplicator で使ったキャラクタ型スペシャルファイルを再初期化するときは、再初期化するかどうかを確認するメッセージが表示されます。この確認メッセージに「Y」を入力すると、Datareplicator ファイルシステム領域が再初期化されます。「Y」以外を入力すると、コマンド処理は中止されます。初期化していないキャラクタ型スペシャルファイルを指定してコマンドを実行すると、確認メッセージを表示しないで初期化します。

形式

```
hdsfmkfs -l 最大ファイル数 -f キャラクタ型スペシャルファイル名
          [ -n Datareplicator ファイルシステム領域サイズ ] ※
          [ -q resp | noresp ]
```

注※

このオプションは、HP-UX の場合だけ省略できます。

オプション

- -l 最大ファイル数

～ 〈符号なし整数〉 ((1～255))

Datareplicator ファイルシステム領域に格納するファイル数の上限を設定します。

- -f キャラクタ型スペシャルファイル名

～ 〈パス名〉

初期化するキャラクタ型スペシャルファイル名を、パス名で設定します。ここに指定したパス名が、Datareplicator ファイルシステム領域名になります。

Datareplicator ファイルシステム領域として使えるファイルサイズは、キャラクタ型スペシャルファイルのファイルサイズに依存します。

- -n Datareplicator ファイルシステム領域サイズ

～ 〈符号なし整数〉 ((1～1048575)) (単位：MB)

Datareplicator ファイルシステム領域として割り当てる容量を、MB 単位で指定します。-f オプションに指定したキャラクタ型スペシャルファイルの容量よりも大きい値を指定すると、KFRB04001-E (errno = 28 : disk に空きがない) でエラーになって、処理が中断されます。

このオプションは、HP-UX の場合にだけ省略できます。HP-UX の場合にこのオプションを省略すると、-f オプションで指定したキャラクタ型スペシャルファイルの全容量が仮定されます。

なお、このオプションに 2048 (2GB) 以上を指定する場合は、Datareplicator ファイルシステム領域を作成するファイルシステムがラージファイルに対応している必要があります。ラージファイルの運用については、「6.11 ラージファイルの運用」を参照してください。

- `-q resp | noresp`

hdsfmkfs コマンドの動作を指定します。

resp

確認メッセージを出力します。

noresp

確認メッセージを出力しません。確認メッセージに「Y」の入力を仮定して動作します。

規則

- Datareplicator ファイルシステム領域の状態と hdsfmkfs コマンドのオプションの関係を次に示します。

Datareplicator ファイルシステム領域の状態	hdsfmkfs コマンドのオプションの指定		
	なし	-q resp	-q noresp
未初期化状態	A	A	A
初期化状態	A	A	A
使用状態 (停止中)	B	B	A
使用状態 (稼働中)	C	C	C

A：確認メッセージを出力しないで、初期化処理を続けます。

B：確認メッセージ KFRB04809-Q が出力されて、「Y」を入力すると初期化処理を続けます。

C：KFRB04807-E が出力されて、初期化処理が中断されます。

- hdsfmkfs コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- Datareplicator ファイルシステム領域が次に示す状態のときは、hdsfmkfs コマンドの処理が中止されます。

領域が初期化されていないことを示すメッセージが出力される理由：

- 指定したキャラクタ型スペシャルファイルが割り当てられていません。
- キャラクタ型スペシャルファイル以外のファイルを指定しています。
- Datareplicator を実行したユーザに、キャラクタ型スペシャルファイルへの read/write 権限がありません。

領域が使用中であることを示すメッセージが出力される理由：

- Datareplicator の実行中に hdsfmkfs コマンドを実行しています。

ファイルアクセスエラーのメッセージが出力される理由：

- Datareplicator ファイルシステム領域のディスクに障害が起きました。

hdsfstate (反映側二重化ファイルの状態表示)

機能

反映側システムの二重化したファイル群 (物理ファイル) の状態を表示するコマンドです。

形式

```
hdsfstate [-f 論理ファイル名]
```

オプション

- -f 論理ファイル名

状態を表示する論理ファイル名を絶対パスで指定します。

このオペランドを省略した場合、コマンドを実行したサーバマシン上のすべての二重化ファイルに関する状態を表示します。

出力形式

hdsfstate コマンドで抽出情報キューファイルの状態を出力する形式を次に示します。

```
*****
**          HiRDB Datareplicator file status information          **
**          Sun Aug 25 12:58:52 2002                             **
*****
Logical file name = /users/repli/quefile001_bes1                . . . . . 1.

sys status  type Physical file name                            . . . . . 2.
a  a----- rqu /hd01/hds/qufile001_bes1_1
b  a----- rqu /hd01/hds/qufile001_bes1_2
-----
Logical file name = /users/repli/quefile002_bes1

sys status  type Physical file name
a  a----- rqu /hd01/hds/qufile002_bes1_1
b  h----- rqu /hd01/hds/qufile002_bes1_2

Number of Logical file Information = 2                          . . . . . 3.
```

1. 論理ファイル名を示します。

2. 物理ファイル状態情報を示します。

sys : 物理ファイルの系です。

A 系の場合 a, B 系の場合は b が表示されます。

出力	説明
a	A 系ファイル
b	B 系ファイル

status : 物理ファイルの状態です。1 列目の表示から状態を判断します。

出力	説明
a	現用使用状態 (ACTIVE)
h	障害発生状態 (HOLD)
*	状態を取得できません。

type：物理ファイルの種別コードです。

出力	説明
rmt	反映マスタステータスファイル
rst	反映ステータスファイル
rqu	反映情報キューファイル
---	上記以外のファイル
***	ファイル種別を取得できません。

Physical file name：物理ファイル名です。

3. 出力された論理ファイル情報の件数を示します。

hdsfstats (Datareplicator ファイルシステム領域の状態の表示)

機能

Datareplicator ファイルシステム領域の状態を表示します。

Datareplicator ファイルシステム領域は、UNIX 版 Datareplicator で使えます。Windows 版 Datareplicator ではこのコマンドは使えません。

hdsfstats コマンドは、Datareplicator が起動していても起動していなくても実行できます。ただし、Datareplicator ファイルシステム領域の初期化の実行中にコマンドを実行すると、コマンドを入力した時点の領域の状態がそのまま表示されます。

形式

```
hdsfstats -f Datareplicatorファイルシステム領域名
```

オプション

- -f Datareplicator ファイルシステム領域名
～ 〈パス名〉

状態を表示する Datareplicator ファイルシステム領域名を指定します。

規則

- hdsfstats コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- Datareplicator ファイルシステム領域が次に示す状態のときは、hdsfstats コマンドの処理が中止されます。

領域が初期化されていないことを示すメッセージが出力される理由：

- Datareplicator ファイルシステム領域が初期化されていません。

領域が見付からないことを示すメッセージが出力される理由：

- コマンドに指定した Datareplicator ファイルシステム領域がありません。

ファイルアクセスエラーのメッセージが出力される理由：

- Datareplicator ファイルシステム領域のディスクに障害が起きました。

出力形式

コマンドの実行結果は、標準出力に出力されます。出力形式を次に示します。

```

*****
**      HiRDB Datareplicator file system area information      **
**              Fri Mar 29 16:32:05 2002                      **
*****
file system area      /HiRDBDS/sysfile/sysarea05      .....1.
status                used                          .....2.
initialize time       Mon Oct 22 19:40:01 2001      .....3.
Datareplicator directory /HiRDBDS/system01/hde      .....4.
Datareplicator type   hde                          .....5.
area capacity         204800 [KB]                  .....6.
remain area capacity  44691 [KB]                   .....7.
sector size           1024 [B]                     .....8.
available file count  20                            .....9.
used file count       18                            .....10.

[ 1] file name = /HiRDBDS/system01/hde/sts_bes2      .....11.
    file kind = DS_STATS      file size = 77 [KB]    .....12.
[ 2] file name = /HiRDBDS/system01/hde/sts_bes2.
    file kind = DS_HDEFL      file size = 9 [KB] .
[ 3] file name = /HiRDBDS/system01/hde/que2/qufile001_bes2.
    file kind = DS_QUEUE      file size = 10001 [KB]

      :
      :
[ 17] file name = /HiRDBDS/system01/hde/que2/qufile015_bes2
     file kind = DS_QUEUE      file size = 10001 [KB]
[ 18] file name = /HiRDBDS/system01/hde/que2/qufile016_bes2.
     file kind = DS_QUEUE      file size = 10001 [KB]

```

(凡例)

1. file system area

Datareplicator ファイルシステム領域名が表示されます。

2. status

Datareplicator ファイルシステム領域の状態が表示されます。

init : hdsfmkfs コマンドで初期化した状態

used : Datareplicator が使った状態

error : hdsfmkfs コマンドで初期化中に障害が起こった状態

3. initialize time

hdsfmkfs コマンドで初期化した時刻が表示されます。

4. Datareplicator directory

使った Datareplicator 運用ディレクトリ名が表示されます。

5. Datareplicator type

Datareplicator の種類が表示されます。

hde : 抽出側 Datareplicator

hds : 反映側 Datareplicator

6. area capacity

Datareplicator ファイルシステム領域の総容量が表示されます (単位: キロバイト)。

7. remain area capacity

Datareplicator ファイルシステム領域の未使用領域の容量が表示されます (単位: キロバイト)。

8. sector size

Datareplicator ファイルシステム領域のセクタ長が表示されます (単位: キロバイト)。

9. available file count

hdsfmkfs コマンドに指定したシステムファイルの上限が表示されます。

10. used file count

使用中のシステムファイルの数が表示されます。

11. [nn] file name = aa...aa :

nn : Datareplicator ファイルシステム領域内でのシステムファイルの位置が表示されます。

aa...aa : システムファイル名が表示されます。

12. file kind = bb...bb file kind = cc...cc [KB]

bb...bb : システムファイル種別が表示されます。

DS_STATS : ステータスファイル

DS_HDEFL : HiRDB 連絡ファイル

DS_QUEUE : キューファイル

cc...cc : システムファイル容量が表示されます (単位: キロバイト)。

hdsinfoget (反映側障害情報の取得)

機能

反映側 Datareplicator で障害が発生した場合、障害の原因を調査するために定義情報、及び障害情報を取得します。

UNIX 版 Datareplicator だけで使用できるコマンドです。Windows 版 Datareplicator では使用できません。

このコマンドは Datareplicator 管理者が実行してください。

形式

```
hdsinfoget [ -n 反映側Datareplicator識別子 ]  
           -e 初期情報出力先ディレクトリ
```

オプション

- -n 反映側 Datareplicator 識別子

取得対象とする反映側 Datareplicator 識別子を指定します。

このオプションを省略した場合、サーバマシン上のすべての反映側 Datareplicator 識別子が取得対象となります。

- -e 初期情報出力先ディレクトリ

初期情報ファイル（障害原因の切り分けに必要な情報をまとめたファイル）を出力する既存のディレクトリ名を絶対パスで指定します。

指定したディレクトリが存在しない、又は書き込み権限がない場合は、エラーメッセージを出力して処理を打ち切ります。

初期情報出力先ディレクトリには、初期情報ファイルと取得処理結果ログが出力されます。初期情報ファイルはアーカイブファイルであり、複数のファイルが格納されています。それぞれのファイル名を次に示します。

出力ファイル	ファイル名
初期情報ファイル	hdsinfoget_<ホスト名※1>_init_<年月日時分秒※2>.tar.gz
取得処理結果ログ	hdsinfoget_comlog_<ホスト名※1>_<年月日時分秒※2>

注※1

ホスト名の長さが 32 バイト以上の場合、先頭から 32 バイト分の文字を使用します。

注※2

年月日時分秒は「YYYYMMDDhhmmss」（YYYY：西暦年、MM：月、DD：日、hh：時、mm：分、ss：秒）です。

初期情報ファイルに取得される情報を次に示します。

分類		取得する情報	取得の有無
定義情報	Datareplicator 定義ファイル	反映システム運用ディレクトリパス情報	○
		反映システム定義ファイル	○
		反映環境定義ファイル	○
		反映定義ファイル	○
		更新情報定義ファイル	×
		二重化定義ファイル	○
		反映抑止対象リストファイル	○
障害情報	Datareplicator 運用ファイル	反映情報キューファイル	×
		反映ステータスファイル	△
		反映マスタステータスファイル	△
		未反映情報ファイル	○
		反映エラー情報ファイル	○
		反映トレースファイル	○
		コマンドログファイル	○※
		SAM ファイル	×
		抽出対象外データ格納ファイル	×
		二重化制御ファイル	○
		UXPL ログファイル	×
	Datareplicator 運用情報	Datareplicator バージョン情報	○
		hdsstate 実行結果	○
		hdsfstate 実行結果	○
		hdsfstatfs コマンド実行結果	○
		データ連動対象の表、インデックス定義	×
	OS 情報	反映側 Datareplicator 運用ディレクトリ下のファイルリスト	○
		Datareplicator インストールディレクトリ下のファイルリスト	○
		プロセスリスト	○
		ipcs -a コマンド実行結果	○
netstat -n コマンド実行結果		○	
hosts ファイル		○	
services ファイル		○	
inetd 情報		○	

分類	取得する情報	取得の有無
	syslog ファイル	○

(凡例)

○：この情報を取得します。

△：UNIX のキャラクタ型スペシャルファイルである場合は、取得しません。レギュラーファイルの場合だけ取得します。

×：この情報を取得しません。

注※

環境変数 `hds_command_log_file` の指定によって任意のファイル名が付与されている場合は、この情報を取得しません。

規則

- 反映側 Datareplicator 起動時のファイル格納場所を基に情報を取得します。そのため、反映側 Datareplicator を一度でも正常に起動したことのある場合だけ、このコマンドで情報を取得できます。なお、このコマンド実行時に反映側 Datareplicator が起動している必要はありません。
- コマンドを実行するユーザにアクセス権限がないファイルは取得できません。KFRB09275-W メッセージを出力して、取得はスキップされます。
- コマンドの結果は、コマンド実行時のリターンコード、又は取得処理結果ログ中のメッセージで確認できます。リターンコードの意味と対処方法を次に示します。

リターンコード	終了状態	意味と対処方法
0	正常終了	コマンドは正常に終了しました。 コンソールや取得処理結果ログに警告メッセージが出力されている場合、一部の情報が取得できていないおそれがあります。メッセージを参照して、個別に取得してください。
8	異常終了	コマンドは異常終了しました。 コンソールや取得処理結果ログに出力されたエラーメッセージを参照して、エラーの原因を取り除いてから、コマンドを再度実行するか、個別に取得してください。
12	割り込み (シグナル発生) による終了	割り込み (シグナル発生) によって処理を打ち切りました。

hdsrefinfm (更新情報内容の確認)

機能

反映情報ステータスファイルから、各更新情報を抽出して、解析結果出力ファイルに出力するコマンドです。次の目的で使用します。

- 反映抑止対象リストファイル中に記述した更新情報識別子が、反映情報キューファイル中に存在するかを確認するために使用します。
- 反映済みの更新情報、又は未反映の更新情報の内容を確認するために使用します。

形式

```
hdsrefinfm -f 反映ステータスファイル名
{-l 9 | -m trn [-t 反映済みトランザクション件数, 未反映トランザクション件数 ] }
[-g 反映グループ名 | -c ]
-p 解析結果出力ファイル名 [-k 出力ファイルの最大値 ]
```

オプション

抽出対象を指定するオプション

- -f 反映ステータスファイル名
情報の抽出対象である反映ステータスファイル名を絶対パスで指定します。

更新情報の出力内容を指定するオプション

- -l 9
「出力形式 1」に示す形式で出力します。
- -m trn
「出力形式 2」に示す形式で出力します。
- -t 反映済みトランザクション件数, 未反映トランザクション件数
～ 〈英数字〉 ((0~999999999,all) 《1,1》

-m trn オプションを指定する場合に、出力する更新情報を抽出側のトランザクション件数で指定します。反映済みトランザクションの件数と、未反映トランザクションの件数を、コンマ (,) で区切って指定してください。全件出力する場合は、all を指定します。

更新情報出力の基点を指定するオプション

- -g 反映グループ名
更新情報出力の基点を、指定した反映グループの開始位置からに設定します。反映グループ名を絶対パスで指定してください。
- -c
更新情報出力の基点を、反映プロセスの開始位置からに設定します。

これらのオプションを省略した場合は、カレントの反映プロセス実行位置を基点とします。

解析結果出力に関するオプション

- -p 解析結果出力ファイル名

結果を出力するファイル名を絶対パスで指定します。

- -k 出力ファイルの最大値

～〈符号なし整数〉((0~1048576))《1024》(単位:KB)

解析結果出力ファイルの最大値を指定します。0を指定すると、解析結果出力ファイルの最大値は制限されません。

なお、解析結果出力ファイルは、このオプションの指定値よりも1KB程度大きくなる場合があるため、注意してください。

出力形式 1

hdsrefinfmt コマンドの出力形式 (-l 9 オプション指定時) を次に示します。

解析結果出力ファイルの反映情報キューファイル解析結果の項目「Extract-id」に更新情報識別子が、「Additional Transaction information」に抽出トランザクション情報が表示されます。

なお、「Extract-id」及び「Additional Transaction information」以外の項目については保守情報であり内容については保証しません。


```

***** HiRDB Datareplicator Reflection Infomation <Fri Nov 21 11:33:34 2003> ****
***** Status File Information *****
<Common Information[MINF]>
  Datareplicator-id:01 Target-DB:HiRDB DB-Locale:SJIS
  Communication protocol:TCP/IP
<Replication Node Information[UJMF]>
  ReplicationNode-id:c1 ReplicationType:RECIEVE EventDefinition:1, 2, 5, 3, 4
<Recieving Information[RVST]>
  Port Information[Internal]:(c4c1e3c1d9c5d7d3c9c34040, 2, 01, H, S, S,
  2003112111284100, 0, 0, 0000000000000000000000000000000000,
  53455149fe78bd3f010000001031121f11312300, ((0, 0), (0, 0)),
  ((0, 4096), (0, 4096)), ((0, 4096), (0, 4096)), 2097152, 2097152, 0, 280)
  WriteInformation:((0, 4096), (0, 4096)) WrapCount:0
<Reflect Control Information[FSTS]>
  Read Information:(0, 3072)
  Control Information[Internal]:SJ, SJ, HiR, Sfm, (0, 3072)
<Reflect Entry Information[STAT]>
  HeadInformation:GroupNo=000 RunMode=Y
  Read Information:(0, 4096)
  GroupInformation:GroupName=trngroup KeyRangeGroup=No
  ReflectStatusInformation:01, 00, 00, 00, 00, 01
  ReflectDetailInformation:Extract System-id:01 CommitInfo:1(0), 1
  Locale=(SJ, SJ) Source-DB=HiR Format=Sfm
  TransactionInfo=(2, 1) EventInfo=(1, 100)
  Additional Transaction Info:3fbd771e00000000000000e3
  ReflectSQLexeInformation:SQL-count<ins:1, upd:0, del:0, prg:0, cmt:1, tms:0>
***** Queue File Information *****
<Definition Information in Status File>
  Schema-name : kxxxxxx
  UpdateInfomation-name : ext01
  Table : id(131504) name(EXT01)
  Column : id(1) name(C1) option(Key, Norm)
  data-type(INTEGER)
  Column : id(2) name(C2) option(Norm, Nullable)
  data-type(INTEGER) char-set(EBCDIK)
<Update Information in Queue File>
  Que-Pos(0, 3072) Block-Size(1024) Type(Data)
  Tran-count(2, 1) Block-Type(S) Event(Y, 100)
  Transaction-Info[Internal](53455149fe78bd3f010000001031121f11312300)
  Tran Data(1) UpdDataCount(1)
  FirstExtractDate(1031121F, 11312300)
  Transaction Branch information=(4e4b4442756e30310000003e, 2, 11 21 )
  UAP-Info[Internal](48524431, 0000000000000000, 00000000000000)
  Additional Transaction information=(Tx, TranInfo, 3fbd771e00000000000000df)
  Updating Data(1) ExtractDate(1031121F, 11312300)
  Extract-Id = 3FBD771E00000000000000DF:1
  Table-id(131504) Kind(INSERT) DataType(Col)
  Column-id(1) Data-Info(Key, Norm)
  After Data : 00000001 *... *
  Column-id(2) Data-Info(Norm)
  After Data : 00000001 *... *
  Tran Data(2) UpdDataCount(0)
  Additional Transaction information=(Tx, TranInfo, 3fbd771e00000000000000e3)
***** Summary Information *****
  Printing Range = (0, 3072) - (0, 4096)
  Block Information
  PortInfo DefInfo EndInfo DataInfo Others
  0 0 0 1 0
  Transaction Information
  TRNcount DataTotal INScout UPDcount DELcount PRGcount EVTcount
  2 1 1 0 0 0 0
*****

```

出力形式 2

hdsrefinfm コマンドの出力形式 (-m trn オプション指定時) を次に示します。

解析結果出力ファイルの反映情報キューファイル解析結果の項目「Extract-id」に更新情報識別子が、「Additional Transaction information」に抽出トランザクション情報が表示されます。

なお、「Extract-id」及び「Additional Transaction information」以外の項目については保守情報であり内容については保証しません。

```

**** HiRDB Datareplicator Reflection Infomation <Wed Mar 9 20:49:28 2005> ****
*****
**** Reflected Information ****
*****
+-----+
| Tran Data(-2) |
+-----+
Transaction Branch information=(4e4b4442756e30310000004f, 1, 00 )
Additional Transaction information=(Tx,TranInfo, 423a43590000000000000079)
Updating Data(1) ExtractDate(2005/03/09 20:48:07)
Extract Id = 422E5A8A00000000000011EF:1
Table(131193, hitachi.T1) Kind(PURGE) DataType(Row)
+-----+
| Tran Data(-1) |
+-----+
Transaction Branch information=(4e4b4442756e303100000053, 1, 00 )
Additional Transaction information=(Tx,TranInfo, 423a43590000000000000008a)
Updating Data(1) ExtractDate(2005/03/09 20:48:08)
Extract Id = 422E5A8A0000000000001200:1
Table(131193, hitachi.T1) Kind(INSERT) DataType(Col)
Column(1, C1) Data-Info(Key, Norm)
After Data : 00000000 *... *
Column(2, C2) Data-Info(Norm)
After Data : 30202020 20202020 20202020 20202020 *0 *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 * *
**** Summary Information *****
Transaction Information
TRNcount      INScount      UPDcount      DELcount      PRGcount      EVTcount
          2             1             0             0             1             0
*****
**** Un-reflect Information ****
*****
+-----+
| Tran Data(1) |
+-----+
Transaction Branch information=(4e4b4442756e303100000054, 2, 00 21 )
Additional Transaction information=(Tx,TranInfo, 423a43590000000000000008d)
Updating Data(1) ExtractDate(2005/03/09 20:48:09)
Extract Id = 422E5A8A0000000000001204:1
Table(131193, hitachi.T1) Kind(UPDATE) DataType(Col)
Column(1, C1) Data-Info(Key, Norm)
Before Data : 00000000 *... *
Column(1, C1) Data-Info(Key, Norm)
After Data : 00000064 *...d *
Column(2, C2) Data-Info(Norm)
After Data : 31303020 20202020 20202020 20202020 *100 *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 * *

```

```

+-----+
| Tran Data(2) |
+-----+
Transaction Branch information=(4e4b4442756e30310000005a, 2, 00 21 )
Additional Transaction information=(Tx,TranInfo, 423a43590000000000000008e)
Updating Data(1) ExtractDate(2005/03/09 20:48:10)
Extract Id = 422E5A8A0000000000001212:1
Table(131193, hitachi.T1) Kind(INSERT) DataType(Col)
Column(1, C1) Data-Info(Key, Norm)
After Data : 00000001 *... *
Column(2, C2) Data-Info(Norm)
After Data : 31202020 20202020 20202020 20202020 *1 *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 * *
Updating Data(2) ExtractDate(2005/03/09 20:48:11)
Extract Id = 422E5A8A0000000000001212:2
Table(131193, hitachi.T1) Kind(INSERT) DataType(Col)
Column(1, C1) Data-Info(Key, Norm)
After Data : 00000002 *... *
Column(2, C2) Data-Info(Norm)
After Data : 32202020 20202020 20202020 20202020 *2 *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 20202020 20202020 20202020 * *
                20202020 * *
Event(201)
+-----+
| Tran Data(3) |
+-----+
Transaction Branch information=(4e4b4442756e30310000005b, 2, 00 21 )
Additional Transaction information=(Tx,TranInfo, 423a435900000000000000090)
Updating Data(1) ExtractDate(2005/03/09 20:48:11)
Extract Id = 422E5A8A0000000000001216:1
Table(131193, hitachi.T1) Kind(DELETE) DataType(Col)
Column(1, C1) Data-Info(Key, Norm)
Before Data : 00000001 *... *
**** Summary Information ****
Transaction Information
TRNcount    INScount    UPDcount    DELcount    PRGcount    EVTcount
      3           2           1           1           0           1
*****

```

注意

- hdsrefinm コマンド実行環境の文字コードと抽出システム定義の dblocale オペランドに指定した文字コードが異なる場合は、環境変数 HDS_RFI_ELANG を指定してください。
- hdsrefinm コマンド実行環境の文字コードと反映システム定義の dblocale オペランドに指定した文字コードが異なる場合は、環境変数 HDS_RFI_PLANG を指定してください。

hdsrfctl (反映処理の制御)

機能

hdsrfctl コマンドには、次の機能があります。

- 反映処理の再開始

停止中の反映処理を、反映方式を指定して再開始できます。hdsrfctl コマンドを実行すると、直ちに反映処理が再開始されます。

- 反映処理の停止

受信処理は稼働したままで、反映処理だけを停止できます。反映処理の停止タイミングは、指定するオプションによって異なります。

- 反映方式の変更

稼働中の反映処理の反映方式を変更する場合は、次の手順で変更してください。

1. 反映処理を停止します。
2. トランザクション単位反映方式と表単位反映方式のどちらかを指定して、反映処理を再開始します。

- 反映処理のコミット間隔の変更

稼働中の反映処理のコミット間隔を変更します。停止中の反映処理に対しては、コミット間隔を指定して再開始します。

形式 1

```
hdsrfctl -d データ連動識別子 -m { trn | tbl | spd | immediate }  
        [ -c 反映処理コミット間隔]
```

形式 2

```
hdsrfctl -g 同期反映グループ名  
        -m { start | immediate }
```

オプション (形式 1)

- -d データ連動識別子

制御の対象とする反映処理に対応する抽出側システムのデータ連動識別子を指定します。データ連動識別子に**を指定した場合は、引数不正エラーとなります。

抽出側 DB が HiRDB の場合のデータ連動識別子については、[\[5.2 抽出システム定義\]](#)を参照してください。抽出側 DB がメインフレーム側 DB の場合のデータ連動識別子については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。

- -m { trn | tbl | spd | immediate }

trn

トランザクション単位反映方式で反映処理を実行します。

tbl

表単位反映方式（表単位分割方式，キーレンジ単位分割方式，ハッシュ分割方式）で反映処理を実行します。

spd

コマンドを実行した後に，抽出側システムからの何らかのイベントを反映処理で検知した時点で，反映処理を停止します。

immediate

コマンドを実行したときに処理中の反映処理が終了した時点で，反映処理を停止します。コマンドを実行したときに受信中の更新情報がある場合には，反映情報キューファイルへの格納が完了した時点で停止します。

- -c 反映処理コミット間隔

～〈符号なし整数〉《反映環境定義の cmtintvl オペランドの指定値》((1~32767))

制御の対象とする反映処理が，反映側 HiRDB に対してコミットを発行する間隔を指定します。ここで指定した値は，hdsrfctl コマンドを実行した後に開始される反映処理から有効になります。hdsstop コマンドを実行，又は hdsrfctl コマンドを再実行した場合には，-c オプションで前回指定した値は無効になります。

-c オプションを指定した hdsrfctl コマンドを実行して，反映処理を再開始中又はコミット間隔変更中にエラーになった場合には，指定したコミット間隔は無効になります。

オプション (形式 2)

- -g 同期反映グループ名

制御の対象とする同期反映グループを指定します。

- -m { start | immediate }

start

同期反映グループを起動します。ただし，前回終了時に同期反映グループを解除した状態で起動した場合は，前回と同様に解除した状態で起動します。

immediate

反映処理を停止します。停止方法を次に示します。

- ・トランザクションの処理中に同期イベントを検知した場合，トランザクションを COMMIT して反映処理を停止します。
- ・トランザクションの処理中に反映情報キューファイルの終端を検知した場合，トランザクションをロールバックして反映処理を停止します。
- ・処理中のトランザクションがない場合は即時停止します。

規則

- hdsrfctl コマンドは，特定のデータ連動識別子に対する反映処理を対象とするコマンドです。複数のデータ連動識別子に対して反映処理を制御する場合は，データ連動識別子ごとに hdsrfctl コマンドを実行する必要があります。

- hdsrfctl コマンドは、データ連動識別子を指定して抽出側システムごとに実行します。
- hdsrfctl コマンドで反映処理の方式を変更する前には、hdsrfctl -m spd コマンドで、いったんイベント終了モードで反映処理を終了する必要があります。
- 反映側 Datareplicator はイベントでの要求よりも、hdsrfctl コマンドでの要求を優先します。例えば、反映処理の方式の変更を要求したイベントを実行しても、hdsrfctl コマンドで反映処理の停止を要求した場合、コマンドに従って反映処理を停止します。
- hdsrfctl コマンドを実行すると、反映側 Datareplicator はイベント待ち状態になります。このイベント待ち状態のときに hdsrfctl コマンドを再実行すると、後で実行した hdsrfctl コマンドをイベント待ち状態の対象とします。
- hdsstop コマンドを実行して反映側 Datareplicator が停止処理中のときは、hdsrfctl コマンドを実行するとコマンドエラーになります。
- hdsrfctl コマンドを実行してイベント待ち状態のとき、反映処理が異常終了した場合、イベント待ち状態の hdsrfctl コマンドは取り消されます。
- 遅延時間 (breaktime オペランド) を指定して、反映処理を遅延開始している場合には、hdsrfctl -m spd コマンドを実行すると、指定時間後に反映処理を開始しないように遅延時間を無効とし、遅延開始を取り消すことができます。
- 表単位反映モードで反映処理を稼働しているときに一部の反映グループだけがエラーで停止している場合には、エラー停止している一部の反映グループだけを再開始できません。一部の反映グループだけがエラーで停止しているときの運用方法については、「[6.7.1 反映処理の運用](#)」を参照してください。
- hdsrfctl コマンドの -m オプションで spd 又は immediate を指定している場合、-c オプションでコミット間隔を指定してもコミット間隔は無視されます。
- hdsstop コマンド又は hdsrfctl を複数回実行した場合の動作は、コマンドに指定しているオプションによって異なります。hdsstop コマンド又は hdsrfctl を複数回実行した場合の動作については、hdsstop コマンドの文法説明の「[表 7-16 hdsstop コマンド又は hdsrfctl コマンドを複数回実行した場合の動作](#)」を参照してください。
- hdsrfctl コマンドの戻り値、終了区分、及びその要因について次の表に示します。

戻り値	区分	要因
0	正常終了	—
1	異常終了	主な要因は次のとおりです。 <ul style="list-style-type: none"> • 引数不正 • コマンド実行環境不正 (環境変数不正, 定義ファイル不正など) • システムコールエラー • 反映側 Datareplicator 未起動
11	コマンド受け付け不可	反映処理が稼働済みです。
12		反映処理の開始要求に対する応答がありません。

hdssamqin (SAM ファイルからの更新情報の抽出)

機能

SAM ファイルを使う抽出側 DB (PDMII E2, RDB1 E2) の場合に、転送されてきた SAM ファイルの更新情報を反映情報キューファイルに蓄積します。蓄積した更新情報は、Datareplicator の反映処理で HiRDB に反映されます。

形式

```
hdssamqin -d データ連動識別子 -f 入力SAMファイル名
          -n 更新情報定義ファイル名 [ -l 更新情報編集バッファ長 ]
          [ -c ] [ -o ]
```

オプション

- -d データ連動識別子

データ連動識別子を指定します。同時に実行する更新情報入力プロセスや受信プロセスと重複しないように指定してください。データ連動識別子に**を指定した場合は、引数不正エラーとなります。

- -f 入力 SAM ファイル名

～ 〈絶対パス名〉

入力する SAM ファイルを指定します。抽出側 DB からバイナリモードで転送した SAM ファイルを指定してください。ファイル名だけを指定すると、ファイルの場所はコマンドを実行したカレントディレクトリが仮定されます。

- -n 更新情報定義ファイル名

～ 〈絶対パス名〉

データ抽出のための更新情報定義を記述したファイル名を指定します。ファイル名だけを指定すると、ファイルの場所はコマンドを実行したカレントディレクトリが仮定されます。

-n オプションは、PDMII E2 の場合にだけ指定できます。RDB1 E2 の場合は更新情報定義ファイルを作成する必要がないので、-n オプションは指定できません。

- -l 更新情報編集バッファ長

～ 〈符号なし整数〉 ((1~2097151)) 《300》 (単位 : KB)

PDMII E2 から抽出した更新情報を編集するバッファのサイズを指定します。

-l オプションは、PDMII E2 から抽出された SAM ファイルにだけ有効で、RDB1 E2 から抽出した SAM ファイルには無効です。

- -c

前回のコマンド実行時に処理した SAM ファイルの情報を無視して処理を続けるときに指定します。前回の hdssamqin コマンドの実行が正常終了したか異常終了したかに関係なく、コマンドに指定した SAM ファイルを処理します。

- -o

前回のコマンド実行時に障害が起こった場合に、前回の続きから処理を再開するとき指定します。

規則

hdssamqin コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。

注意

SAM ファイルの入力処理が異常終了して、入力ファイルの処理がすべて終了していない場合は、Datareplicator の初期起動コマンド (hdsstart -i) は実行しないでください。初期起動すると、SAM ファイル入力の同期点の情報がなくなって正しくデータ連動できなくなります。万一初期起動した場合は、HiRDB の反映対象表を再作成してから hdssamqin コマンドを実行してください。

hdsshmclean (反映側 Datareplicator の共用リソースの削除)

機能

反映側 Datareplicator が異常終了した場合に、反映側 Datareplicator の共用リソース（プロセス、共用メモリ、又はセマフォ）が残っているとき、残っている共用リソースを削除します。このコマンドは UNIX 版限定のコマンドです。このコマンドは Datareplicator 管理者が実行してください。

形式

```
hdsshmclean [ -l clean [ -q {resp | noresp} ] [ -w ] ]
```

オプション

- **-l clean**

共用リソースを削除する場合に指定します。このオプションを指定しない場合は、残っている共用リソースの表示だけ行います。

- **-q {resp | noresp}**

共用リソースを削除する前に、削除してよいかを確認する KFRB04331-Q メッセージを出力するかどうかを指定します。

resp

KFRB04331-Q メッセージを出力します。y を入力した場合だけ共用リソースを削除します。

noresp

KFRB04331-Q メッセージを出力しないで、共用リソースを削除します。

- **-w**

共用リソースの削除が完了するまで、このコマンドの終了を待ち合わせる場合に指定します。このオプションを指定しない場合、コマンドが終了した時点で共用リソースの削除も完了しているかは保証できません。

出力形式

```

<<<*** Common information ***>>> ..... 1
Shmid : common      = 13467653 ..... 2
Semid : common      = 8912925 ..... 3
PID   : hdsmaster   = 7044 ..... 4
PID   : hdstrcrv    = 7046 ..... 5
PID   : hdstcpmst   = 7047 ..... 6
PID   : hdsosimst   = 7048 ..... 7

<<<*** Sync group information(grpname) ***>>> ..... 8
Shmid : refsync     = 6422538 ..... 9
Shmid : TBI-list    = 6684690 6684691 ..... 10
Semid : refsync     = 10453040 ..... 11
PID   : hdsrefsync  = 22707 ..... 12

<<<*** Replication information (01) ***>>> ..... 13
Shmid : defserv     = 13533191 ..... 14
PID   : hdstcpmst   = 27921 ..... 15
PID   : hdsdefserv  = 27922 ..... 16

<<<*** Group information (trngroup) ***>>> ..... 17
Shmid : lobshm      = 0 ..... 18
PID   : hdsreflect  = 27958 ..... 19
PID   : hdssqlle   = 27959 ..... 20

<<<*** Replication information (02) ***>>>
Shmid : defserv     = 14254130
PID   : hdsosimst   = 27931
PID   : hdsdefserv  = 27932

<<<*** Group information (trngroup) ***>>>
Shmid : lobshm      = 0
PID   : hdsreflect  = 27960
PID   : hdssqlle   = 27963
:
:

```

出力情報の説明

番号	出力情報	備考
1	反映マスタプロセス管理リソースヘッダ	—
2	反映側プロセス間連絡用共用メモリ ID	—
3	反映側プロセス間排他用セマフォ ID	—
4	反映マスタプロセスのプロセス ID	—
5	稼働トレース収集プロセスのプロセス ID	—
6	通信マスタプロセスのプロセス ID (TCP/IP)	—
7	通信マスタプロセスのプロセス ID (OSI)	—
8	同期反映グループリソースヘッダ	反映システム定義に syncgroup001 を指定している場合に出力されます。
9	同期反映グループ用共用メモリ ID	
10	TBI リスト用共用メモリ ID	反映システム定義に syncgroup001 を指定している場合に出力されます。TBI リスト用共用メモリが複数存在する場合は、スペースで区切って出力されます。

番号	出力情報	備考
11	同期反映グループ内排他用セマフォ ID	反映システム定義に syncgroup001 を指定している場合に出力されます。
12	同期管理プロセスのプロセス ID	
13	データ連動識別子単位リソースヘッダ	—
14	反映定義格納用共用メモリ ID	—
15	データ受信プロセスのプロセス ID	プロセス名は、使用プロトコルによって hdstcpmst 又は hdsosimst のどちらかが出力されます。
16	反映定義サーバプロセスのプロセス ID	—
17	反映グループ単位リソースヘッダ	反映グループごとに、番号 17～20 までの情報が出力されます。
18	BLOB データ格納用共用メモリ ID	—
19	反映プロセスのプロセス ID	反映環境定義の startmode オペランドを spd に変更して反映側 Datareplicator を起動した場合、前回起動時のプロセス ID が表示されることがあります。
20	反映 SQL プロセス、又は反映 UOC プロセスのプロセス ID	

(凡例)

—：特にありません。

規則

残った共用リソースが原因で反映側 Datareplicator を起動できない場合は、hdsstop コマンドを実行して共用リソースを削除してください。hdsstop コマンドを実行しても共用リソースを削除できない場合に、このコマンドを実行してください。

hdsstart (反映側 Datareplicator の起動)

機能

反映システム定義に従って、反映側 Datareplicator を起動します。反映システム定義については、「5.8 反映システム定義」を参照してください。

形式

```
hdsstart [ -i [ init ] [ -D データ連動識別子番号 ] [ -f ] [ -q ] [ -r ] ]
         | -c 同期反映グループ名
```

オプション

- -i
反映側 Datareplicator の環境を初期化してから、反映側 Datareplicator を起動（初期開始）する場合に指定します。
-f オプションを省略した場合、初期化対象の反映環境が既に存在して、その反映環境の反映情報キューファイルに反映されていない更新情報が残っていると、エラーになります。
-D オプションを指定した場合は、特定の抽出元に対応する反映環境だけを初期化して、反映側 Datareplicator を起動します。
-D オプションを指定しない場合は、反映システム定義又は反映環境定義で指定した次に示すファイルを初期化し、反映側 Datareplicator の環境全体を初期化して、反映側 Datareplicator を起動します（-D オプション指定時には、指定された抽出元のデータ連動識別子に対応するファイルだけが初期化されます）。
 - 反映情報キューファイル
 - 反映ステータスファイル
 - 反映マスタステータスファイル
 - 反映エラー情報ファイル
 - 稼働トレースファイル
 - 未反映情報ファイル

init

反映環境定義で指定したすべての反映情報キューファイルと反映ステータスファイルを、反映環境定義の `queuesize`、`statssize` オペランドに指定したそれぞれのサイズで初期作成します。

`init` を指定してコマンドを実行すると、`-i` オプションだけを指定したときに比べて初期化処理に時間が掛かります。この処理時間は、ファイルサイズや個数によって異なります。

ラージファイル運用時にこのオプションを指定している場合、コマンド実行時間が長くなります。詳細は「6.11.2 ラージファイル運用時のコマンド実行時間の見積もり」を参照してください。

-D オプションを指定した場合は、指定された抽出元のデータ連動識別子に対応するファイルだけを初期作成します。

-D データ連動識別子番号

指定した抽出元に対応する反映環境に対して初期化（部分初期化）をします。指定されなかった抽出元に対応する反映環境は通常の起動をします。

初期開始処理でエラーが発生した場合は、処理を中断し、反映側 Datareplicator は起動しません。抽出元は、データ連動識別子番号で指定します。データ連動識別子番号には、反映システム定義のデータ連動識別子（dsidxxx）の xxx 部分の数字を指定します。

データ連動識別子番号に指定できるのは、前回初期開始した時点で定義されていたデータ連動識別子番号だけで、新たに追加したデータ連動識別子番号は指定できません。

このオプションで部分初期化の対象になるのは、反映システム定義の dsidxxx, refenvxxx, refenvxxx で指定された反映環境定義、及び反映環境定義の reffile で指定された反映定義です。

部分初期化以外の定義の変更は、正常開始時と同様に扱います。

部分初期化するデータ連動識別子以外の定義変更は行なわないでください。複数のデータ連動識別子の部分初期化を行う場合、対象のデータ連動識別子の定義変更→部分初期化を繰り返してください。

-f

反映情報キューファイル内に未反映の更新情報が残っているかどうかに関係なく、強制的に初期開始を実行します。反映情報キューファイル内に未反映の更新情報が残っていた場合、未反映の更新情報は初期開始によって破棄されるため、このオプションを指定して初期開始する場合は必ず未反映の更新情報を破棄してよいかどうかを確認してから実行してください。

なお、次のどれかに該当する場合はこのオプションを指定してください。

- ・ hdsstop -t force コマンドによって強制終了した場合
- ・ 反映情報キューファイルと反映ステータスファイルをキャラクタ型スペシャルファイルで作成した後に初めて初期化する場合（2回目以降は必要ありません）
- ・ 反映環境定義の statsfile オペランドに指定したステータスファイルの名称を変更する場合に、変更するステータスファイルと同じ名称のファイルが既に存在しているとき

-q

初期開始又は部分初期開始した後、反映 Datareplicator を起動しないで終了します。

-r

反映側 Datareplicator 起動時に反映処理数をリセットします。

このオプションは、反映システム定義の reflect_counter_reset オペランドで false を指定した場合に有効になります。

-c 同期反映グループ名

指定した同期反映グループ名の同期反映をキャンセルして、反映側 Datareplicator を正常開始します。このオプションは、hdeevent コマンドの実行漏れによって反映トランザクションが COMMIT できなくなったり、反映トランザクションが反映側 DB のリソース不足によって COMMIT できなくなったりした場合に使用します。

キャンセルした同期反映グループを構成していた各データ連動識別子は、以降、データ連動識別子ごとに反映処理を実行します。また、同期反映グループを指定したことによって無効になっていたオペランドが有効となります。

一度キャンセルした同期反映グループは、初期化するまで再構成できません。定義されていない同期反映グループ名、又は既にキャンセルした同期反映グループ名を指定した場合、反映側 Datareplicator の正常起動はエラーになります。

規則

- hdsstart コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- 開始モードは、前回の終了モードによって異なります。開始モードと前回の終了モードとの関係を次の表に示します。

表 7-12 開始モードと前回の終了モードとの関係

開始モード	前回の終了モード	起動時の動作内容
初期開始	—※1	前回の稼働時の動作内容を引き継がないで、反映システム定義に従って反映側 Datareplicator が開始します。 初期開始のときには、反映側 Datareplicator は「表 7-13 -i オプションを指定したときの各ファイルに対する処理の内容」に示すようにファイルを初期化します。
部分初期開始	—※2	部分初期開始のときには、反映側 Datareplicator は指定された抽出元に対応する反映環境だけを初期化して、反映側 Datareplicator が開始します。
正常開始	正常終了、イベント終了、又は即時終了	前回の稼働時の動作内容を引き継がないで、反映システム定義に従って反映側 Datareplicator が開始します。
再開始	異常終了又は強制終了	抽出側 DB と反映側 DB の整合性を保証するため、反映環境定義の一部のオペランド※3 の設定内容を無効とし、前回の稼働時と同じ動作内容で反映側 Datareplicator が開始します。

注※1

前回の終了モードに関係なく、-i オプションを付けて hdsstart コマンドを実行すると、初期開始モードで反映側 Datareplicator が起動されます。

注※2

-D オプションで初期化対象に指定されたデータ連動識別子に対する処理は初期開始になります。それ以外のデータ連動識別子に対する処理は前回の終了モードによって正常開始又は再開始になります。

注※3

再開始のときには、反映環境定義のオペランドのうち、startmode, breaktime, breakmode, eventspd, eventtrn, eventtbl, cmtintvl, trncmtintvl, tblcmtintvl オペランドの設定内容が無効になります。ただし、cmtintvl, trncmtintvl, tblcmtintvl オペランドについては、抽出側 DB と反映側 DB の整合性が保証できた時点で、自動的にオペランドの設定内容が有効になります。

- -i オプションを指定したときの各ファイルに対する処理の内容を次の表に示します。

表 7-13 -i オプションを指定したときの各ファイルに対する処理の内容

ファイル名	既存のファイル なし	既存のファイル種別		Datareplicator ファイルシステム領域 ^{※1}	
		OS の 通常ファイル	キャラクタ型スペ シャルファイル ^{※1}	初期化状態	使っている状態
反映情報キュー ファイル ^{※2}	OS の通常ファイ ル形式で作成され ます。	再作成されま す。	何もしません ^{※3}	ファイルが割り当てら れます。	ファイルが再割り 当てされます。
反映ステータス ファイル ^{※2}			ヘッダ初期化 ^{※3}	ファイルが割り当てら れます。	ファイルが再割り 当てされます。
反映マスタステー タスファイル			ヘッダ初期化 ^{※3}	—	—
未反映情報ファ イル			—	—	—
反映エラー情報 ファイル			—	—	—
稼働トレースファ イル			—	—	—

注※1

キャラクタ型スペシャルファイルと Datareplicator ファイルシステム領域は、UNIX 版 Datareplicator で使えます。

Windows 版 Datareplicator では「OS の通常ファイル」で示す処理が実行されます。

注※2

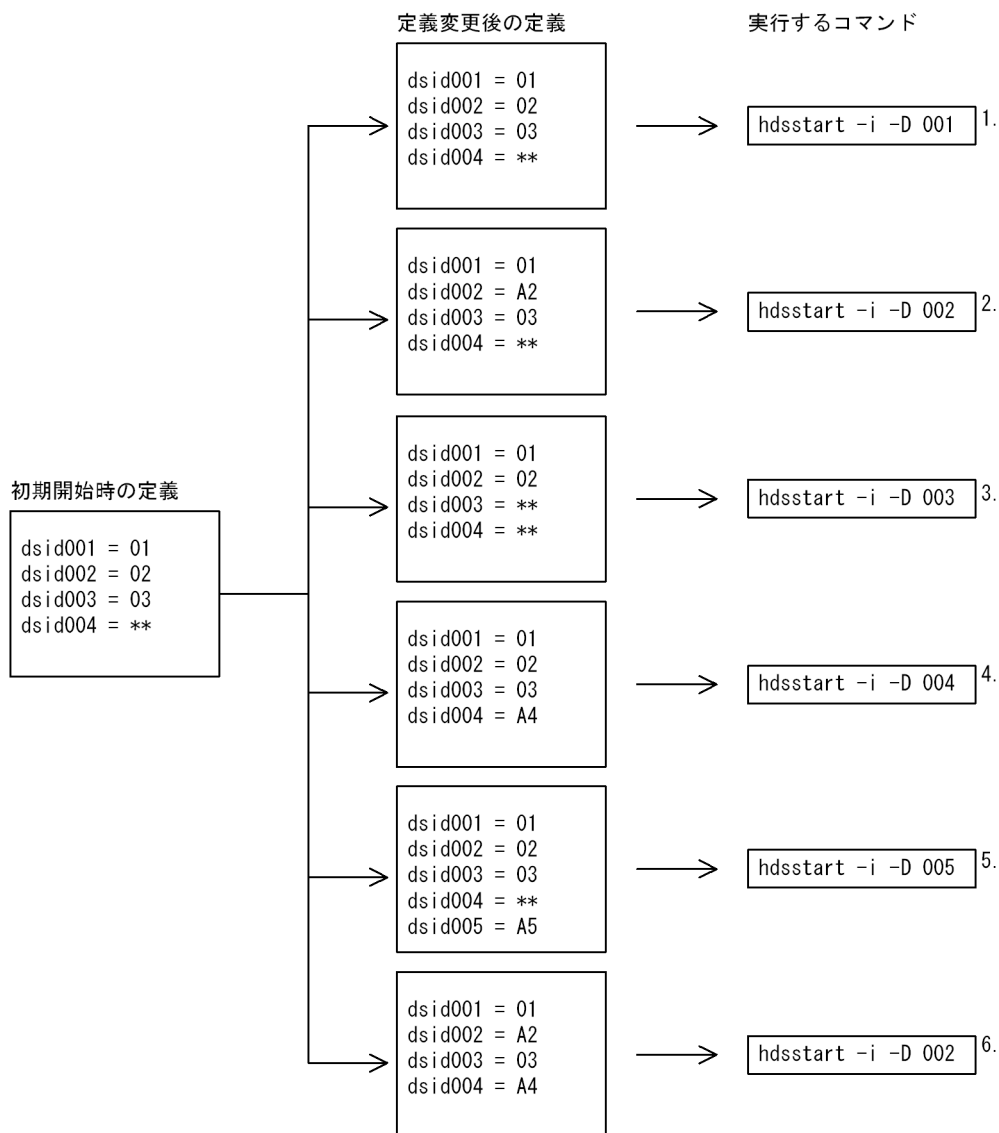
init を付けて hdsstart -i コマンドを実行したときは、指定したサイズで初期作成します。

注※3

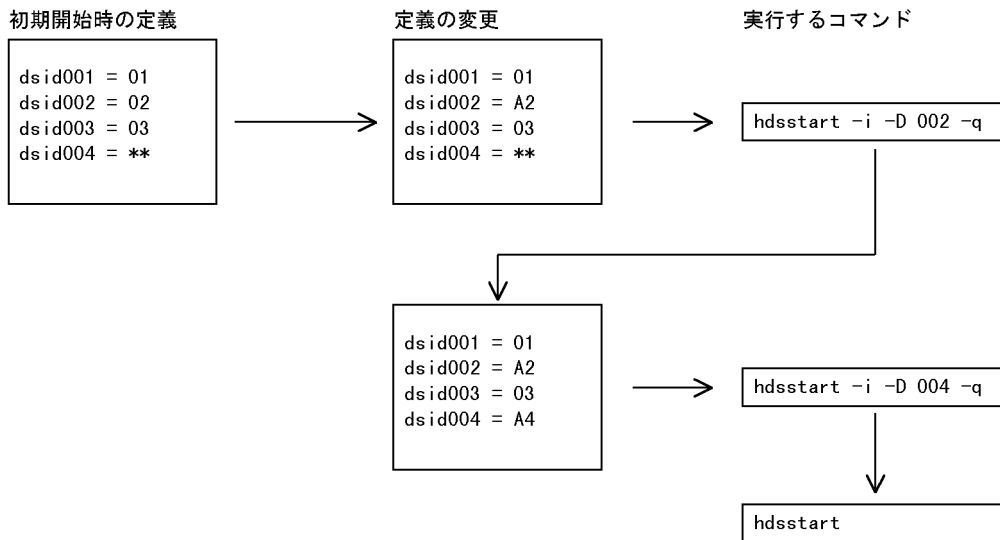
再作成する場合は、OS のコマンドを使ってください。

注意

- 部分初期開始をしているときにエラーが発生したままの状態、別のデータ連動識別子番号を指定した部分初期開始、及び正常起動はできません。
部分初期開始のエラー状態を解消するためには、エラーとなったデータ連動識別子番号を指定した部分初期開始を、エラーの原因を取り除いた後、再度実行して正常終了させるか、反映環境全体を初期開始する必要があります。
- 部分初期開始を指定できるのは、前回初期開始した時点で定義されていたデータ連動識別子番号だけで、新たに追加したデータ連動識別子番号は指定できません。
あらかじめ送信先が増えることが分かっているときは、将来使用するデータ連動識別子に**を指定しておくようにします。
部分初期開始時の指定例を次に示します。



1. データ連動識別子番号 001（データ連動識別子 01）の部分初期開始をします。
 2. 定義変更後のデータ連動識別子番号 002（データ連動識別子 A2）の部分初期開始をします。
 3. 定義変更によって欠番になったデータ連動識別子番号 003（データ連動識別子 **）の部分初期開始をします。
 4. 新たに定義されたデータ連動識別子番号 004（データ連動識別子 A4）の部分初期開始をします。
 5. 定義変更によって追加されたデータ連動識別子番号 005（データ連動識別子 A5）は、初期開始時に定義されていないため、コマンドエラーになります。
 6. 指定されたデータ連動識別子番号（データ連動識別子 A2）以外のデータ連動識別子番号 004（データ連動識別子 A4）が変更されているため、コマンドエラーになります。
- 2 か所以上のデータ連動識別子の部分初期開始をする場合は、1 か所ずつ定義の変更及び-q オプションの指定をして部分初期開始をしてください。すべての部分初期化が終了した後、正常開始をしてください。
- この例の場合の実行手順を次に示します。



- 部分初期開始時の実行手順を次に示します。
 1. 反映側 Datareplicator が停止していることを確認します。
 2. テキストエディタで定義内容を変更します。
 3. hdsstart -i -D コマンドで初期化対象のデータ連動識別子番号を指定して、部分初期開始をします。
- 系切り替え構成の場合、初期化又は起動をする前に HiRDB Datareplicator が使用する資源（共用ディスク装置）を活性化しておく必要があります。

hdsstate (反映側 Datareplicator のステータス情報の取得)

機能

コマンドを実行した時点の反映側 Datareplicator の状態を標準出力に出力します。出力される情報は、次のとおりです。

- **共通情報**
反映側システムに対するすべての反映処理に共通な情報です。
- **同期反映グループ情報**
同期反映グループに関する情報です。
- **反映情報キューファイル情報**
反映情報キューファイルの使用状態を表す情報です。
- **受信処理情報**
プロトコル種別、受信イベントの有無など受信処理の状態を表す情報です。
- **反映処理情報**
反映処理の動作モード、コミット間隔など反映処理の状態を表す情報です。

形式

```
hdsstate [-d データ連動識別子 | -g 同期反映グループ名]
```

オプション

- **-d データ連動識別子**
特定の抽出側システムに対する反映側 Datareplicator のステータス情報を取得する場合に、取得の対象とする抽出側システムのデータ連動識別子を指定します。データ連動識別子に**を指定した場合は、引数不正エラーとなります。
抽出側 DB が HiRDB の場合のデータ連動識別子については、「[5.2 抽出システム定義](#)」を参照してください。抽出側 DB がメインフレーム側 DB の場合のデータ連動識別子については、マニュアル「[VOS3 XDM データ連動機能 XDM/DS 解説・定義](#)」を参照してください。
-d オプションを省略すると、すべての抽出側システムに対応する反映側 Datareplicator のステータス情報が取得されます。
- **-g 同期反映グループ名**
反映トランザクション同期機能使用時、同期反映グループを構成するデータ連動識別子のステータス情報を取得する場合に、その同期反映グループ名を指定します。
-g オプションを省略すると、すべての抽出側システムに対応する反映側 Datareplicator のステータス情報が取得されます。

規則

- hdsstate コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- hdsstate コマンドは、反映側 Datareplicator が稼働中のときだけ実行できます。
- 反映側 Datareplicator では hdsstate コマンドを実行した時点でステータス情報を取得し、標準出力に出力します。このため出力される情報は、実際の反映側 Datareplicator の状態とは多少ずれが発生することがあります。
- 欠番指定されたデータ連動識別子の情報は出力されません。

出力形式

```
*****
**          HiRDB Datareplicator status information          **
**          Thu Nov 10 13:47:41 2005                        **
*****

----- Common information -----

hdsid = c0   shmid = 3504   semid = 907           ..... 1.
PID : hdsmaster = 4692,           ..... 2.
      hdstcpmst = 4693,       hdsosimst = 4694

----- Sync-Group information -----

sync-group = Grp001           ..... 1.
shmid = 4                     ..... 2.
semid = 1                     ..... 3.
PID : hdsrefsync = 27266      ..... 4.

syncwait tran count = 11      ..... 5.
```

```

----- Replication information -----
dsid = a1 [sync-group:Grp001] ..... 1.

<EXTRACTED DATA QUEUE FILE INFORMATION>

wrap count = 1 ..... 2.
qufile001 status : using ..... 3.
qufile002 status : using
qufile003 status : using
qufile004 status : unused
qufile005 status : unused

max used ratio : 2005/11/10 13:47:41 70% ..... 4.
current used ratio : 2005/11/10 13:47:41 10% ..... 5.

<COMMUNICATION INFORMATION>

PID : hdsosimst = 4709 protocol = OSI ..... 6.
receiving event = exist ..... 7.
write position : qufile003, offset =2048 ..... 8.

<REFLECTION INFORMATION>

PID : hdsdefserv = 4695 definition-shmid = 1305 ..... 9.
reflection mode = tbl commit interval = 1 .....10.

GROUP = AAAAA PID :hdsreflect = 4712, .....11.
hdssqle = 4715
status :data-wait1(last event = retbl) .....12.
DB-status :normal .....13.

read transaction = 4 .....14.
read position : qufile003, offset = 1536 .....15.
uocname = none .....16.

Reflection count : .....17.
ins = 4, upd = 0, del = 0, purge = 0,
commit = 4, timestamp = 0
commit(extract transaction) = 4
reflect delay times = 0000:00:05/0000:10:00.....18.

SQL Distribute information : .....19.
hdssqle[1]:
ins = 2, upd = 0, del = 0, purge = 0,
commit = 2, timestamp = 0
hdssqle[2]:
ins = 1, upd = 0, del = 0, purge = 0,
commit = 1, timestamp = 0

GROUP = BBBBB
[グループBBBBBの反映処理情報]

dsid = a2 [sync-group:Grp001]
[データ連動識別子a2の抽出側システムに対する反映処理のステータス情報]

```

共通情報

1. hdsid ~ semid = 907

hdsid には反映側 Datareplicator 識別子が出力されます。shmid には共用メモリ ID が出力されます。
semid にはセマフォ ID が出力されます。

2. PID : hdsmaster ~ hdsosimst = 4694

hdsmaster には反映マスタプロセス ID が出力されます。hdstcpmst 又は hdsosimst には、指定した通信プロトコルに対応する反映通信マスタプロセス ID が出力されます。

反映マスタプロセスが存在しない場合には、hdsstate コマンドを実行したときにエラーになります。反映通信マスタプロセスが存在しない場合には、反映通信マスタプロセス ID に 0 が出力されます。

同期反映グループ情報

1. sync-group = Grp001

sync-group には同期反映グループ名称が出力されます。

2. shmid = 4

shmid には同期反映グループで使用する共用メモリ ID が出力されます。

3. semid = 1

semid には同期反映グループで使用するセマフォ ID が出力されます。

4. PID : hdsrefsync = 27266

hdsrefsync には同期管理プロセスのプロセス ID が出力されます。

5. syncwait tran count = 11

前回のコミット以降、反映中の抽出側トランザクション数が出力されます。

反映情報

反映情報ファイル情報

1. dsid = a1 [sync-group:Grp001]

-d オプションで指定したデータ連動識別子が出力されます。指定したデータ連動識別子が存在しない場合には、hdsstate コマンドを実行したときにエラーになります。

同期反映グループに指定されていないデータ連動識別子の場合の出力形式は次のとおりです。

```
dsid=xx
```

同期反映グループに指定されているデータ連動識別子の場合の出力形式は次のとおりです。

```
dsid=xx [sync-group:同期反映グループ名]
```

2. wrap count = 1

反映側 Datareplicator の初期開始時から、反映情報キューファイルを何順したかを出力します。

3. qufile001 ~ status : unused

反映環境定義の qufile001 ~ 008 オペランドで指定した反映情報キューファイルごとに、使用状況を出力します。使用状況は次のよう出力されます。

using : 使用中 (更新情報が格納されている場合は常に using)

unused : 未使用

4. max used ratio : 2005/11/10 13:47:41 70%

反映情報キューファイル全体の使用率が最大値を示した日時と、その使用率が表示されます。

反映情報キューファイルを初期化した時点、又は hdschgstatus コマンドで最大使用率をリセットした時点の表示形式は次のとおりです。

```
max used ratio      : ****/**/** **:***:**      0%
```

5. current used ratio : 2005/11/10 13:47:41 10%

hdsstate コマンドを実行した日時と、hdsstate コマンドを実行した時点の反映情報キューファイル全体の使用率が表示されます。

受信処理情報

6. PID : hdsosimst ~ protocol = OSI

hdsosimst には受信プロセス ID が出力されます。protocol には通信プロトコルが出力されます。通信プロトコルは次のように出力されます。

TCP/IP : TCP/IP プロトコル

OSI : OSI プロトコル

受信プロセスが存在しない場合には、受信プロセス ID には -1 が出力されます。

7. receiving event = exist

抽出側システムと接続した受信イベントの有無が出力されます。受信イベントの有無は次のように出力されます。

exist : 受信イベントあり

not exist : 受信イベントなし

N/A : 該当しない (受信プロセスが停止中のため)

8. write position ~ offset = 2048

更新情報の書き込み位置が出力されます。

qufilexxx には反映環境定義で指定した qufile001 ~ 008 オペランドのうちの、更新情報を書き込んでいる反映情報キューファイルに対応するものが出力されます。offset には、更新情報の書き込み位置が、ファイルの先頭からのオフセットで出力されます。

反映処理情報

9. PID : hdsdefserv ~ definition-shmid = 1305

hdsdefserv には反映定義サーバプロセス ID が出力されます。definition-shmid には定義情報格納用共用メモリ ID が出力されます。

反映定義サーバプロセスが存在しない場合、反映定義サーバプロセス ID には -1 が出力されます。

10. reflection mode ~ commit interval = 1

reflection mode には反映処理モードが出力されます。commit interval には反映処理コミット間隔が出力されます。反映処理モードは次のように出力されます。

init : 反映処理初期化中 (Datareplicator の初期化後、更新情報を受信していない、又は抽出側の正常停止を検知していない状態を示します)

trn : トランザクション単位反映方式で処理中

tbl : 表単位反映方式で処理中、又は UOC 使用時

spd : 反映処理のサスペンド中 (この場合 12. ~ 18. は表示されません)

N/A : 該当しない (定義サーバプロセスが停止中のため)

反映処理コミット間隔には、コミットを発行する間隔が抽出側システムでのトランザクション数で出力されます。

反映処理コミット間隔に出力される値は、反映環境定義の `cmtintvl`, `trncmtintvl`, `tblcmtintvl` オペランドの指定値、又は `hdsrftcl` コマンドの `-c` オプションの指定値です。ただし、反映プロセスの起動処理中の場合には、0 が出力されます。

11. **GROUP ~ hdssqle = 4715**

GROUP には反映グループ名が出力されます。`hdsreflect` には反映プロセス ID が出力されます。

hdssqle には反映 SQL プロセス (UOC プロセス) ID が出力されます。ただし、反映処理の停止後はプロセス ID が -1 で表示されます。

反映グループ名に出力される名称は、次のとおり異なります。

trngroup : reflection mode が `trn` 又は `init` の場合

反映グループ名 : reflection mode が `tbl` の場合

othergrp : 反映グループの指定がない場合

uocxxx : UOC を使っている場合 (`xxx` は 001 から順に出力されます)

反映 SQL プロセス (UOC プロセス) ID には、反映 SQL プロセスの場合には `hdssqle` と出力され、反映 UOC プロセスの場合にはユーザが定義した反映 UOC プロセス名が出力されます。ただし、定義情報格納用共用メモリが存在しないときには、"`uoc [unknown]`" と出力されます。

12. **status ~ retbl**

status には反映処理の状態が出力されます。`last event` には反映処理が最後に検知したイベントの種別が出力されます。

反映処理の状態は次のよう出力されます。

run : 動作中

definfo-wait : 定義情報待ち

data-wait1 : 更新情報受信待ち (送信終了通知反映前)

data-wait2 : 更新情報受信待ち (送信終了通知反映後)

event-stop : イベント停止

normal-stop : 正常停止

abnormal-stop : 異常停止

midway-stop : 強制停止

イベントの種別は次のよう出力されます。

none : イベント未検知

trn : トランザクション単位反映イベント

tbl : 表単位反映イベント

spd : 反映処理停止イベント

retrn : トランザクション単位反映再起動イベント

retbl : 表単位反映再起動イベント
cntreset : 反映処理数リセットイベント
end : 終了イベント
other : その他

13. DB-status : normal

反映側 DB の状態が次のよう出力されます。

normal : 正常な状態又は反映処理初期化中

critical : 反映処理の同期点処理中に、反映処理が異常終了した状態

14. read transaction = 4

反映処理を起動した時点からの、反映情報キューファイルから読み込んだトランザクション数が出力されます。

ここで出力される値は、コミットに関係なく、反映情報キューファイルに格納された更新情報の合計値です。コミットされたトランザクションの件数については、17.に出力されます。

15. read position ~ 1536

更新情報の読み込み位置が出力されます。

qufilexxx には反映環境定義で指定した qufile001 ~ 008 オペランドのうちの、更新情報を読み込んでいる反映情報キューファイルに対応するものが出力されます。offset には、更新情報の読み込み位置が、ファイルの先頭からのオフセットで出力されます。

16. uocname = none

uoc を使っている場合、UOC 名が出力されます。uoc を使っていない場合には、none が出力されます。ただし、定義情報格納用共用メモリが存在しないときには、"N/A"と出力されます。

17. Reflection count ~ commit(extract transaction)= 4

反映処理を起動した時点から実行した反映処理の数が出力されます。

反映処理に UOC を使っている場合には、uoc に引き渡した反映情報数が出力されます。UOC を使わない反映処理の場合には、SQL の実行数が出力されます。SQL の実行数は次のよう出力されます。

ins : insert 数

upd : update 数

del : delete 数

purge : purge 数

commit : commit 数

timestamp : 時系列情報表に対する insert 数

UOC を使っている場合には 0

commit(extract transaction) : COMMIT した抽出元のトランザクション数

18. reflect delay times = 0000:00:05/0000:10:00

抽出側の HiRDB でコミットした時点から、その更新データを反映側 HiRDB で反映するまでの経過時間、及び反映定義に指定した値が出力されます。表示形式は次のとおりです。

表示形式	説明
0000:00:00/定義値	反映プロセスから何も出力されていません。
.**:*//**.*:*	反映環境定義の reflect_delay_limit_time オペランドに 0 が指定されています。
--:---:--/定義値	9999:59:59 を超えました。
-xxxx:xx:xx/定義値	抽出側の HiRDB でコミットした時点からと、その更新データが反映されるまでの経過時間がマイナスです。抽出側のマシンの時刻が、反映側のマシンの時刻より進んでいます。

19. SQL distribute information ~ timestamp = 0

反映処理の SQL プロセスごとの commit 済み反映処理の数が出力されます。

この反映処理の数は、キーレンジ分割やハッシュ分割を実行したときの SQL プロセスの実行数を示します。SQL の実行数は次のように出力されます (ins, upd, del, purge, timestamp は、17. と一致します)。

ins : insert 数

upd : update 数

del : delete 数

purge : purge 数

commit : commit 数

timestamp : 時系列情報表に対する insert 数

上記の値は、反映処理が起動したときだけ 0 に戻ります。

補足

- 共通情報の 1.~2.は、反映側 Datareplicator が保持するすべての Replication node に共通する情報です。
- 反映情報の 1.~19.は、データ連動識別子ごとに保持する情報です。このため、hdsstate コマンドの -d オプションを省略すると、すべてのデータ連動識別子の数だけ出力されます。-d オプションでデータ連動識別子を指定すると、指定したデータ連動識別子の数だけ出力されます。
- 反映情報の 9.~19.は、反映処理のグループの数だけ出力されます。
- 反映情報の 14.は、処理中又は最後に処理したトランザクションを示します。17.は、反映が完了した (反映側 DB に COMMIT した) トランザクション数を示します。
- hdsstart -i コマンドで起動して、かつ reflection mode が spd の場合は、commit interval に 0 が表示されます。
- SAM ファイルからの更新情報を使ったデータ連動の場合、受信処理情報は次のようになって、反映情報の 6.~8.は出力されません。
Communication process not effective
- 反映情報の 19.は、キーレンジ分割やハッシュ分割などの一つの反映プロセスに対して複数の SQL プロセスが起動するときだけに出力されます。

- 反映情報の 2.は、値が 2147483647 を超えると 0 に戻ります。
- 反映情報の 14., 17., 19.は、値が 18446744073709551615 を超えると 0 に戻ります。
- 反映情報の 14., 17., 19.で出力される情報は、反映側 Datareplicator の起動単位でリセットされます。
- 反映情報の 17.及び 19.で出力される情報は、次の契機でリセットされます。
 - 反映側 Datareplicator の初期化（部分初期化を含む）又は起動*
 - 反映環境定義の eventcntreset オペランドに指定したイベントの検知
注※ 反映システム定義の reflect_counter_reset に true を指定している場合
- 次の場合、反映処理件数はリセットされます。
 - 前回稼働時の反映処理方式と異なる方式で反映側 Datareplicator を起動した場合
 - 表単位反映方式で、かつ前回稼働時の反映グループについて、次の内容を変更した場合
 - 反映グループ名
 - SQL プロセス数
 - 分割方式（キーレンジ分割、ハッシュ分割）
 - キーレンジ分割の場合、キーレンジグループ数
 - ハッシュ分割の場合、ハッシュ分割表の RD エリア数

なお、詳細については、反映システム定義の reflect_counter_reset オペランドを参照してください。

hdsstop (反映側 Datareplicator の終了)

機能

反映側 Datareplicator を終了します。

形式

```
hdsstop [ -t { event | immediate | force } | -q 待ち時間 ]
        [ -w ]
```

オプション

- -t { event | immediate | force }

-t event

イベント終了する場合に指定します。

-t immediate

即時終了する場合に指定します。

-t force

強制終了する場合に指定します。

- -q 待ち時間

～ 〈符号なし整数〉 ((5~600)) (単位：秒)

受信プロセスを終了させ、かつ、受信したデータの分まで反映を完了させて終了したい場合に、受信プロセスの終了待ち時間とともに指定します。

このパラメタは、抽出側システムのマシン電源断や通信回線断の発生で受信プロセスが回線の切断を検知できない場合や、計画系切り替えのために反映側 Datareplicator を強制的に正常終了させたい場合に指定することを前提としています。このため、正常運用（抽出側システムからの送信完了通知があり、かつ、送信完了のところまで反映を完了させて終了させる）のシェルなどには、このオプションを指定しないようにしてください。また、このオプション (-q) を指定して hdsstop コマンドを実行した後は、-t force オプション以外は指定できなくなります。

なお、このパラメタは、-t オプションと同時に指定できません。

- -w

反映側 Datareplicator のマスタプロセスが停止するまで待った後で、hdsstop コマンドの処理を終了するときに指定します。

計画系切り替えで使用する Datareplicator 停止シェルを使用するときのような Datareplicator 本体が完全に停止するのを待ってコマンドを終了させる必要がある場合に、このオプションを指定してください。

-w オプションなしのときは、hdsstop コマンドはマスタプロセスに停止要求を送信完了すると終了します（マスタプロセスはコマンドと非同期に終了します）。

-w オプションありのときは、hdsstop コマンドはマスタプロセスに停止要求を送信完了後、マスタプロセスの終了まで待ってから終了します（コマンド終了時には、マスタプロセスも終了しています）。

規則

- hdsstop コマンドを実行して、正常終了すると 0 が返されます。異常終了すると 1 が返されます。
- -t オプションの指定によって、終了モードを選択できます。-t オプションの指定と終了モードとの関係を次の表に示します。

表 7-14 -t オプションと終了モードとの関係

オプション	終了モード	終了時の動作内容
なし	正常終了	hdsstop コマンドの実行後、抽出側システムの稼働単位まで反映処理を終了した時点で、反映側 Datareplicator が終了します。抽出側 DB が HiRDB の場合の稼働単位は、抽出側システム全体の正常開始から正常終了までです。抽出側 DB がメインフレーム側 DB の場合の稼働単位については、マニュアル「VOS3 XDM データ連動機能 XDM/DS 解説・定義」を参照してください。 なお、抽出システムとの接続状態や、hdsstop コマンドに指定したオプションによっては、上記以外の場合でも正常終了になる場合があります。詳細は「表 7-15 正常終了の条件」を参照してください。
-t event	イベント終了	hdsstop コマンドの実行後、最初に抽出側システムから送信されたイベントを反映処理が検知した時点で、反映側 Datareplicator が終了します。ただし、イベントを検知する前に、抽出側システムの稼働単位までの反映が終了した場合には、イベントを待たないで、反映側 Datareplicator が終了します。 表単位反映方式で反映処理が稼働している場合は、最も遅いグループの反映が終了した時点で反映処理が終了します。 このオプションは、計画的に停止する場合にだけ使ってください。 急に停止させたい場合や受信分の反映を完了させて終了させたい場合には、このオプションは使わないでください。
-t immediate	即時終了	hdsstop コマンドの実行後、現在反映処理中の更新情報の反映が終了した時点で反映側 Datareplicator が終了します。 表単位反映方式で反映処理が稼働している場合は、最も進んでいるグループから停止し、最も反映の遅れているグループの反映が、最も進んでいるグループの停止地点に到達した時点で反映側 Datareplicator が終了します。hdsstop コマンドの実行後から、最も進んでいるグループの反映が終了する時点までに更新情報を受信した場合には、更新情報が反映情報キューファイルに格納された時点で受信処理が終了します。
-t force	強制終了	更新情報の反映状態に関係なく、hdsstop コマンドの実行後、強制的に反映側 Datareplicator が終了します。同期点処理中の場合には、同期点処理が終了した後に、反映側 Datareplicator が終了します。 強制終了すると、反映側 DB と抽出側 DB とで一時的な不整合が生じることがあります。この場合、次回の起動時に開始モードを再開とすると、反映側 DB が自動的に回復されます。

表 7-15 正常終了の条件

抽出システムとの接続状態	反映状態	正常終了の可否
接続中※1	受信分反映完了	×

抽出システムとの接続状態		反映状態	正常終了の可否
		受信分反映未完了	×※3
未接続	送信完了通知あり (抽出側の正常停止)	受信分反映完了	○
		受信分反映未完了	×※4
	送信完了通知なし (抽出側強制停止※2)	受信分反映完了	○
		受信分反映未完了	×※4

(凡例)

- ：正常終了します。
- ×：正常終了しません。

注※1

通信障害によって実際には回線切断が発生していても、keep alive 未設定などのために認識できない場合を含みます。

注※2

抽出側の強制停止 (-t force) のほか、pdrplstop による連動中止ログの検知や抽出側のマシン断・通信障害による通信回線切断を含みます。

注※3

hdsstop コマンドの-q オプションによって、強制的に正常終了できます。この場合、通信回線を強制的にクローズし、反映未完了分をすべて反映完了させてから停止します。なお、この終了方法は、回線切断が発生していても受信プロセスが認識できなかった場合に強制的に正常終了させるときに使います。

注※4

反映未完了分をすべて反映完了させた後は、受信分反映完了と同じです。

- 複数の抽出側システムを反映対象にしている場合には、すべての抽出側システムの反映処理を実行した後、反映側 Datareplicator は終了します。ただし、終了モードが強制終了になっている場合には、直ちに反映側 Datareplicator は終了します。
- -t オプションの指定に関係なく、抽出側システムからの送信データがない場合で更新情報の終端まで反映が完了しているときは、反映側 Datareplicator が終了します。

注意

- hdsstop コマンドを実行して終了処理中のときには、抽出側システムからの接続要求を受け付けません。
- hdsstop コマンドを実行してから終了処理が完了するまでの間に反映側システムが異常終了した場合、終了前に受け付けた停止コマンドの指示は無効になります。
- 抽出側システムからの更新情報の送信がなく、さらに受信しているすべての更新情報の反映処理が終了している状態で hdsstop コマンドを実行した場合には、コマンドのオプションの指定に関係なく、反映側 Datareplicator は終了します。ただし、オプションを指定しない hdsstop コマンド (正常終了) を実行した場合は、前回の停止が正常終了だった場合だけ、反映側 Datareplicator は終了します。

- hdsstop コマンド又は hdsrfctl を複数回実行した場合の動作は、コマンドに指定しているオプションによって異なります。hdsstop コマンド又は hdsrfctl を複数回実行した場合の動作を次の表に示します。

表 7-16 hdsstop コマンド又は hdsrfctl コマンドを複数回実行した場合の動作

後に実行したコマンド		先に実行したコマンド							
		hdsstop コマンドの -t オプションの指定				hdsrfctl コマンドの -m オプションの指定			
		なし	event	immediate	force	trn	tbl	spd	immediate
hdsstop コマンドの -t オプション の指定	なし	×	×	×	×	○	○	○	○
	event	×	×	×	×	○	○	○	○
	immediate	○	○	×	×	○	○	○	○
	force	○	○	○	×	○	○	○	○
hdsrfctl コマンドの -m オプション の指定	trn	×	×	×	×	×	×	△	△
	tbl	×	×	×	×	×	×	△	△
	spd	×	×	×	×	○	○	○	×
	immediate	×	×	×	×	○	○	○	○

(凡例)

- ：先に実行したコマンドの処理を中断して、後に実行したコマンドの処理を開始します。
- △：先に実行したコマンドの処理が終了しているときは、後に実行したコマンドの処理を開始します。先に実行したコマンドの処理が終了していないときは、後のコマンドを実行した時点でエラーになって、先に実行したコマンドの処理を続けます。
- ×：後のコマンドを実行した時点でエラーになって、先に実行したコマンドの処理を続けます。

- Datareplicator のバージョンが 05-02 以前の場合、pdrplstop コマンドを実行して抽出処理を強制終了、又は通信障害などによって抽出-反映間の通信が切断されているときに、hdsstop コマンドで反映側 Datareplicator を停止できません。この状態で反映側 Datareplicator を停止するときは、hdsstop -t immediate 又は hdsstop -t force コマンドを実行してください。
- Windows 版 Datareplicator で、hdsstop -t force コマンドを実行しても反映側 Datareplicator を停止できなかった場合は、hdsemergencystop コマンドで反映側 Datareplicator を強制終了してください。

hdstrcredit (稼働トレースファイルの編集)

機能

抽出側 Datareplicator／反映側 Datareplicator の稼働中に出力されたトレースファイルを編集して、別ファイル（又は標準出力）に出力します。オプションの指定に従って、稼働トレースファイルの内容すべてを解析して出力します。

指定したオプションのエラーが複数あるときは、チェックできるだけのエラーメッセージがすべて出力されます。また、オプションの解析結果は、hdstrcredit コマンドのローカルメモリに保持されます。

形式

```
hdstrcredit -f稼働トレースファイル名 [, 稼働トレースファイル名]
            [ -l { glbl | perf1 | perf2 } ]
            [ -pプロセスID [ -pプロセスID ] ... ] | [ -t 開始日時 [, 終了日時] ]
            [ -o出力先ファイル名 ] [ -O shortfmt ]
```

オプション

- -f稼働トレースファイル名 [, 稼働トレースファイル名]
 ~ 〈1~255 バイトの文字列〉

hdstrcredit コマンドに入力する稼働トレースファイル名を指定します。ここで指定できるのは、次に示す Datareplicator の稼働トレースファイルだけです。

抽出側 Datareplicator のとき：

- 抽出システム定義の int_trc_unit オペランドに system を指定、または省略した場合
 抽出マスタトレースファイル、抽出ノードマスタトレースファイル
- 抽出システム定義の int_trc_unit オペランドに process を指定した場合
 表「抽出側 Datareplicator の稼働トレースファイル名」に示すファイル

反映側 Datareplicator のとき：

- 反映システム定義の int_trc_unit オペランドに system を指定または省略した場合
 反映トレースファイル
- 反映システム定義の int_trc_unit オペランドに process を指定した場合
 表「反映側 Datareplicator の稼働トレースファイル名」に示すファイル

ファイル名は、相対パス名でも絶対パス名でもかまいません。また、「\$HDSPATH/ファイル名」のように環境変数を使っても指定できます。ただし、相対パス名を指定するときは、コマンドをカレントディレクトリで実行してください。

ファイル名は、最大 2 個まで指定できます。ファイル名を複数指定するときは、「,」（コンマ）で区切ります。2 個指定するときは、同じ稼働範囲（Datareplicator の開始コマンドから終了コマンドまでの間で取得した情報）にしてください。ファイル名を 2 個指定するときに、ファイル名とコンマの間

又はコンマと次のファイル名の間に空白を入れると正しく認識できません。ファイルを 2 個指定したときは、作成日付を比較して古い日付の方から入力を開始します。

次に示す指定をすると、コマンドがエラーになって処理を中止します。

- -f オプションを省略
- -f オプションに指定したファイルが存在しない（ファイルを 2 個指定したときに 1 個だけが存在するときもエラー）
- 指定した値が 255 バイトを超えた
- -f オプションを複数指定
- -l { gbl | perf1 | perf2 }
 ～ 《gbl》

トレースファイルの編集結果として、どの情報を出力するかを指定します。値は、英小文字で指定してください。英大文字で指定すると、エラーになります。

-l オプションの指定と出力される情報の関係を次に示します。

トレースファイルの情報	-l オプションの指定値		
	gbl	perf1	perf2
スタートアップ情報 起動時又は停止時の処理実行を示す情報。通常のプロセス起動ごとに 1 回だけ実行される情報で、処理の開始と終了 及びその状態を示します。	○	○	○
グローバル情報 大きな処理の、あるポイントで取得される情報。例えば、送受信の単位や、トランザクションの単位などのように定期的に実行されるが頻度は比較的少ない処理を示します。	○	○	○
ローカルイベント情報 Datareplicator の内部処理での変化を示す情報。グローバル情報に比べるとより小さな処理で実行されるが、全体処理の中では比較的実行頻度が少ない、処理の変局点を示します。	○	○	○
パフォーマンス (1) 情報 比較的大きな範囲（グローバル情報+ローカルイベントの一部）で取得する、性能解析のための情報。概要レベルでの性能を把握するのに有効です。	×	○	○
パフォーマンス (2) 情報 SQL1 件のような、より小さな範囲で取得する性能解析情報。情報量が非常に多くなるため、取得するときは注意が必要だが、ほぼ完全に性能情報を取得できます。	×	×	○
エラー情報 エラーになったときに取得する情報です。	○	○	○

(凡例)

- ：出力する情報
- ×：出力しない情報

ただし、該当する情報を、トレースファイルを採取するときに指定していることが前提です。トレースファイルを採取するときの指定については、抽出システム定義又は反映システム定義を参照してください。

次に示す指定をすると、コマンドがエラーになって処理を中止します。

- 指定できる値以外の値を指定
- 複数のオプションを指定
- -p プロセス ID [-p プロセス ID] …
 - ～ 〈数字の文字列〉

トレースファイルの編集結果を出力する対象のプロセス ID を指定します。-p プロセス ID -p プロセス ID …の形式で、最大 8 個まで指定できます。

出力したいプロセスのプロセス ID は、エラー情報ファイルを参照してください。

プロセス ID が妥当かどうかは、コマンドではチェックしません。また、指定値の途中で数字でない文字があるときは、その直前までの数値がプロセス ID と仮定されます。

繰り返して指定できる最大数を超過してプロセス ID を指定すると、コマンドがエラーになって処理を中止します。

- -t 開始日時 [, 終了日時]
 - ～ 〈YYYYMMDDHHMMSS〉
 - 年 (YYYY) 最 (MM) 日 (DD) 時 (HH) 分 (MM) 秒 (SS)
 - 《すべてを編集対象》

トレースファイルを編集出力する対象範囲の日時を年 (4 けた) 月 (2 けた) 日 (2 けた) 時 (2 けた) 分 (2 けた) 秒 (2 けた) の形式で指定します。指定する順序は、開始日時 [, 終了日時] でなければなりません。

開始日時を指定するとトレースの採取日時が開始日時以上の情報を、開始日時と終了日時の両方を指定するとトレースの採取日時が開始日時以上でかつ終了日時以下の情報が、編集の対象になります。このとき、うるう年・うるう秒はチェックしません。

開始・終了日時を両方とも指定するとき、開始日時とコンマの間 又はコンマと終了時刻の間に空白を入れると正しく認識できません。

次に示す指定をすると、コマンドがエラーになって処理を中止します。

- 指定形式が間違っている
- 開始日時 >= 終了日時である
- -t オプションを複数指定
- -o 出力先ファイル名
 - ～ 〈1~255 バイトの文字列〉 《標準出力》

編集したトレースファイルを出力するファイル名を指定します。-o オプションを省略すると、標準出力に出力されます。ファイル名は存在していても存在していなくてもかまいませんが、存在しているときはファイルは上書きされます。

ファイル名は、相対パス名でも絶対パス名でもかまいません。また、「\$HDSPATH/ファイル名」のように環境変数を使っても指定できます。ただし、相対パス名を指定するときは、コマンドをカレントディレクトリで実行してください。

次に示す指定をすると、コマンドがエラーになって処理を中止します。

- 指定した値が 255 バイトを超えた
- -o オプションを複数指定
- -O shortfmt
 トレース情報の編集を、1 件 1 行形式の簡略フォーマットで出力することを指定します。
 -O オプションに「shortfmt」以外の値を指定すると、指定した値は無視されます（警告メッセージが出力されます）。

規則

- hdstrcredit コマンドの実行結果は、コマンド実行時の戻り値、又はエラーメッセージの有無で確認してください。コマンド実行時の戻り値と、その意味を次に示します。

戻り値	終了状態	内容
0	正常終了	入力したトレースファイルのすべてのデータを入力・編集し終わりました。
1	異常終了	<ul style="list-style-type: none"> • 入力したトレースファイルが入出力エラーになりました。又は、編集結果の出力で入出力エラーになりました。 • 入力したトレースファイルの内容が間違っています。 • 指定したオプションで、コマンドの処理を続けられない間違いがあります。
不定	異常終了	シグナルに割り込まれました（hdstrcredit コマンドではシグナルをハンドリングしません）。

注意

- 入力する稼働トレースファイルの終端がレコードの途中であるときは、次のように編集します。
 - f オプションで指定した稼働トレースファイル名が 1 個のとき：
 - 途中のレコードは無視して、データの終了と判断されます。
 - f オプションで指定した稼働トレースファイル名が 2 個のとき：
 - 途中のレコードは、次のファイルの先頭レコードと連結されます。
- レコードの入力中に OS（システムコール）からエラーが返されたときの動作を次に示します。
 - OS からエラーが返されたとき
 - エラーメッセージを出力して、コマンドは終了します。
- 抽出側 Datareplicator の初期化（hdestart -i コマンド）を実行したユーザ名で実行してください。

出力形式

稼働トレースファイルで取得したレコードが正常のときは、編集されて出力されます。トレースファイルの編集結果の例を次に示します。

```
*****
**            HiRDB Datareplicator Internal trace information            ** 1.
**            Fri Mar 29 16:32:05 2002                               **
*****
No. 99999999 Lvl:zz Pid:99999,zzzz Cpn:zzzzzz Date:yyyy-mm-dd hh:mm:ss.ttttt 2.
Unsendable-count:99999 Add-strings:zzzzzzzzzzzzzzzzzzzzzzzzzzzz 3.
Perf-info:yyyy-mm-dd hh:mm:ss.ttttt,yyyy-mm-dd hh:mm:ss.ttttt 4.
Sndid:zzzzzzzz Ujid:zz Seqn:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx 5.
Functional-type:zzz(zzzzzzzzzzzzzzzzzz) Values:99999,99999,99999,z,z 6.
Additional-data:
      xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx *zzzzzzzzzzzzzzzzzz*
      xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx *zzzzzzzzzzzzzzzzzz*
      :                                     :
      xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx *zzzzzzzzzzzzzzzzzz*
No. 99999999 Lvl:zz Pid:99999,zzzz Cpn:zzzzzz Date:yyyy-mm-dd hh:mm:ss.ttttt 8.
Unsendable-count:99999 Add-strings:zzzzzzzzzzzzzzzzzzzzzzzzzzzz
Perf-info:yyyy-mm-dd hh:mm:ss.ttttt,yyyy-mm-dd hh:mm:ss.ttttt
Sndid:zzzzzzzz Ujid:zz Seqn:xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Functional-type:zzz(zzzzzzzzzzzzzzzzzz) Values:99999,99999,99999,z,z
Additional-data:
      zzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
      zzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
      :                                     :
      zzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzz
```

(凡例)

- 99...9 : 数字の文字列
- zz...z : 文字列
- xx...x : 16進文字列

出力項目の説明

-O shortfmt オプションを指定したときは、3. ~ 8. は出力されません。

1. ヘッダ情報 (稼働トレースであることを示す文字列とコマンド実行日付)
2. レコード通番, トレースレベル (Lvl), プロセス ID (Pid), 識別名 (Cpn), トレース取得日時
トレースレベル (Lvl) に表示される値 (zz) の意味

- ST : スタートアップ情報
- GL : グローバル情報
- LE : ローカルイベント情報
- P1 : パフォーマンス (1) 情報
- P2 : パフォーマンス (2) 情報

プロセス ID (Pid) に続いて表示されるプロセスコード (, zzzz) の一覧については、「表 7-17 プロセスコードの一覧」を参照してください。

識別名 (Cpn) に表示される値 (zzzzzz) と表示内容の関係については、「表 7-18 プロセスコードと識別名の関係」を参照してください。

日時は、年（4けた）-月（2けた）-日（2けた）時（2けた，24時間）：分（2けた）：秒（2けた）.マイクロ秒（6けた）の形式で出力されます（以降も同様）。

3. 送信できなかった件数，付加文字列情報

Add-strings は、稼働トレースの出力内容を簡易的に示す情報です。稼働トレースの目的，状態，出力元情報（内部情報）が 24 バイト以内の文字列として表示されます。

4. 開始時刻・終了時刻

この情報は、性能についての情報のときだけ出力されます。

5. 送信先識別子 (Sndid)，データ連動識別子 (Ujid)，データシーケンス ID (Seqn)

送信先識別子は、反映側システムの場合は「-----」になります。この情報は、性能についての情報のときだけ出力されます。

6. リターン情報

この情報は、関数のリターン情報 又は SQL 実行についての内容が出力されます。関数のリターン情報の場合にはリターン値が，SQL 実行の場合には SQLCODE 及び SQLWARN6，SQLWARN7 が出力されます。

7. 詳細情報があって，バイナリデータ形式のときは，この形式で出力されます。

8. 詳細情報があって，文字列形式のときは，この形式で出力されます。

Additional-data は、稼働トレースの出力内容を詳細に示す情報です。これらは内部情報で，障害が起こったときの資料として参照します。

表 7-17 プロセスコードの一覧

プロセスコード	プロセス名
TRCR *	トレース情報収集プロセス
MSTR	抽出マスタプロセス，又は反映マスタプロセス
NMST	抽出ノードマスタプロセス
SNDR	送信プロセス
SNDM	
SNDC	
CAPT	抽出プロセス
RCVT	受信プロセス (TCP/IP 通信)
RCVO	受信プロセス (OSI 通信)
DEFS	反映定義サーバプロセス
RFCT	反映プロセス
SQLE	反映 SQL プロセス
UOCG	反映 UOC プロセス

注※

起動直後の 1 回だけ出力されます。

表 7-18 プロセスコードと識別名の関係

プロセスコード	Cpn に表示される値	トレースレベル	意味	同時に出力される付加情報
TRCR	QRCVST	GL	稼働トレース取得の開始	メッセージキューの ID
MSTR	R_INIT	GL	INITENV (初期化) 要求発行トレース取得	要求種別, 要求対象
	R_STRT	GL	START 要求発行トレース取得	要求種別, 要求対象
	R_STOP	GL	STOP 要求発行トレース取得	要求種別, 要求対象
	R_STTE	GL	STATE 要求発行トレース取得	要求種別, 要求対象
	R_WTCH	GL	WATCH (監視) 要求発行トレース取得	要求種別, 要求対象
NMST	R_INIT	GL	INITENV (初期化) 要求受付トレース取得	—
	R_STRT	GL	START 要求受付トレース取得	要求種別, 要求対象
	R_STOP	GL	STOP 要求受付トレース取得	要求種別, 要求対象
	R_STTE	GL	STATE 要求受付トレース取得	要求種別, 要求対象
	R_WTCH	GL	WATCH (監視) 要求受付トレース取得	要求種別, 要求対象
SNDR	SNDSTR	ST	送信プロセスの開始	[hdesender start] を表示
	SNDPPT	P1	ポートチェック送信性能の取得	[Send performance: port check send] を表示
	SNDPDF	P1	抽出定義送信性能の取得	[Send performance: extract definition send] を表示
	SNDPTW	P2	トランザクション管理性能の取得	[Send performance: transaction watch] を表示
	SNDPEX	P1	更新情報送信性能 (送信間隔中の全送信) の取得	[Send performance: send interval] を表示
	SNDEND	ST	送信プロセスの終了	[hdesender stop] を表示
	SNCPSD	P1	更新情報送信性能 (1 回分のデータ送信だけ) の取得	送信分割種別, 送信トランザクション数, 更新情報数, バイト数を表示
	SNCPAU	P2	ADT コンストラクタ情報生成 UOC の性能取得	[ADT UOC call] を表示
	SNCPLB	P2	キューファイル格納形式の組み立て性能の取得	[EB Log] を表示
	SNCPLC	P2	キューファイル格納形式の組み立て性能の取得	[EC Log] を表示
SNCPUUC	P2	送信判定 UOC の性能取得	UOC からの戻り値を表示	

プロセスコード	Cpn に表示される値	トレースレベル	意味	同時に出力される付加情報
	SNCPCV	P2	送信データ編集の処理性能取得（繰返し列）	[hde_edt_extcnv2() call] を表示
	SNCPCV	P2	送信データ編集の処理性能取得	[hde_edt_extcnv() call] を表示
	SNDLYT	P1	送信滞留時間の取得	滞留時間
SNDM	SNMSTR	ST	抽出ノードマスタプロセスの開始	[hdesenmst start] を表示
	SNMEND	ST	抽出ノードマスタプロセスの終了	[hdesenmst end] を表示
SNDC	SNPSTR	ST	送信プロセスの開始	[hdesndprc start] を表示
	SNPPEX	P1	更新情報送信性能（送信間隔中の全送信）の取得	[Send performance: send interval] を表示
	SNPEND	ST	送信プロセスの終了	[hdesndprc end] を表示
	SNPPPT	P1	ポートチェック送信性能の取得	[Send performance: port check send] を表示
	SNPPDF	P1	抽出定義送信性能の取得	[Send performance: extract definition send] を表示
	SNDLYT	P1	送信滞留時間の取得	滞留時間
CAPT	REDINI	ST	プロセス起動直後の READ 開始ポイントの取得	READ ポイント情報（バイナリ形式）
	SYNCQU SYNCRD	LC	READ, WRITE ポイント同期処理トレース	システムログ READ ポイント、キューファイル WRITE ポイント
	CMDTRC	LC	コマンド受付トレース	コマンド要求内容
	CPDLYT	P1	抽出滞留時間の取得	滞留時間
RCVT	RCVDP1	P1	データの受信	データ受信状態
	TCPSOP	ER	socket オープンエラー	関数リターン値, errno
	TCPSOP	GL	socket オープン終了	関数リターン値
	TCPSBN	ER	bind エラー	関数リターン値, errno
	TCPSBN	GL	bind 終了	関数リターン値
	TCPSLI	GL	listen 結果	関数リターン値
	TCPSCP	GL	accept 待ち	—
	TCPSCP	GL	accept リターン	関数リターン値, errno
	TCPFRK	GL	fork エラー	関数リターン値, errno
RCVO	OSITOP	GL	t_open 結果	関数リターン値, errno

プロセスコード	Cpn に表示される値	トレースレベル	意味	同時に出力される付加情報
	OSITBN	GL	t_bind 終了	関数リターン値, errno
	OSITAL	GL	t_alloc 終了	関数リターン値, errno
	OSITLI	GL	t_listen 待ち	—
	OSITLI	GL	t_listen 結果	関数リターン値, errno
	OSITOP	GL	t_open 結果	関数リターン値, errno
	OSITBN	GL	t_bind 結果	関数リターン値, errno
	OSITAC	GL	t_accept 結果	関数リターン値, errno
	OSIFRK	GL	fork 結果	関数リターン値, errno
DEFS	DFINIT	ST	プロセス起動	データ連動識別子
	DFEXIT	ST	プロセス停止	—
	DEFMKP	LE	子プロセス起動 (INIT)	イベントコード, プロセス ID
	DEFMKP	LC	子プロセス起動 (NORMAL)	イベントコード, プロセス ID
	DEFMKP	LC	子プロセス起動 (RERUN)	イベントコード, プロセス ID
	DEFCOM	LC	コマンド入力検知	コマンド名
	DEFMKP	LC	子プロセス起動 (イベント)	イベントコード, プロセス ID
	DEFMKP	LC	子プロセス起動 (hdsstop)	イベントコード, プロセス ID
RFCT	RFINIT	ST	反映プロセス開始	Datareplicator 識別子, データ連動識別子
	EVSTOP	GL	前回イベント停止による停止	イベント ID
	IMSTOP	GL	開始処理中 Immediate 停止	—
	RFSTRT	GL	反映処理開始	開始モード, キューファイルオフセット
	RFSIGS	LE	シグナルの受信	受信シグナル
	RFQGET	P1	キューファイルの終端を検知	キューファイルオフセット
	RFQGET	P1	キューファイルデータを入力	キューファイルオフセット, 更新情報 (先頭 128 バイト)
	RFTRNC	P1	更新情報中のトランザクションエントリ表示	トランザクションカウンタ, トランザクション内更新情報数
	EVENTC	LE	イベントの検知	イベント ID, プロセス状態コード 1,2
	RFTERM	ST	反映プロセスの停止	終了モード, キューファイルオフセット

プロセスコード	Cpnに表示される値	トレースレベル	意味	同時に出力される付加情報
	RFPIPR	P2	SQL 実行結果の受信	送信先エントリ, read サイズ又はリターンコード
	RFPIPW	P2	SQL 実行指示の反映 SQL プロセス又は UOC プロセスへの送信	送信先エントリ, write サイズ, SQL 組み立て情報 (先頭 128 バイト)
	RFRCVR	P1	リラン実行時のトランザクション回復実行結果 (2 相コミットだけ)	リターンコード
	RFCOMT	P1	COMMIT 結果	COMMIT 発生要因, リターンコード
	RFENDC	P1	END/DISCONNECT 結果	種別, リターンコード
	RFEXIT	ST	反映プロセスの停止	種別, リターンコード
	RFDLTY	P1	反映滞留時間の取得	滞留時間
SQLE	SQINIT	ST	プロセスの起動	開始モード (dsid, グループ名, 分割番号)
	SQEXIT	ST	プロセスの停止	終了モード (NORMAL, FORCE, ERROR)
	SQEREQ	GL	停止要求の受付	—
	XARERN	LC	リカバリ実行 (2 相コミット方式)	リカバリ種別
	TRETRY	LC	トランザクションリトライ	—
	CNCT_S	P2	connect 実行前	—
	CNCT_E	P2	connect 実行後	SQLCODE
	PINS_S	P2	prepare (insert) 実行前	SQL 文 (最大 127 バイト)
	PINS_E	P2	prepare (insert) 実行後	SQLCODE
	PUPD_S	P2	prepare (update) 実行前	SQL 文 (最大 127 バイト)
	PUPD_E	P2	prepare (update) 実行後	SQLCODE
	PDEL_S	P2	prepare (delete) 実行前	SQL 文 (最大 127 バイト)
	PDEL_E	P2	prepare (delete) 実行後	SQLCODE
	EINS_S	P2	execute (insert) 実行前	表名
	EINS_E	P2	execute (insert) 実行後	SQLCODE
	EUPD_S	P2	execute (update) 実行前	表名
	EUPD_E	P2	execute (update) 実行後	SQLCODE
	EDEL_S	P2	execute (delete) 実行前	表名
	EDEL_E	P2	execute (delete) 実行後	SQLCODE

プロセスコード	Cpnに表示される値	トレースレベル	意味	同時に出力される付加情報
	EXEI_S	P2	execute immediate 実行前	SQL 文 (最大 127 バイト)
	EXEI_E	P2	execute immediate 実行後	SQLCODE
	CMIT_S	P2	commit 実行前	—
	CMIT_E	P2	commit 実行後	SQLCODE
	RLBK_S	P2	rollback 実行前	—
	RLBK_E	P2	rollback 実行後	SQLCODE
	XAOP_S	P2	xa_open()実行前	—
	XAOP_E	P2	xa_open()実行後	xa_open()戻り値
	XAPR_S	P2	xa_prepare()実行前	—
	XAPR_E	P2	xa_prepare()実行後	xa_prepare()戻り値
	XACM_S	P2	xa_commit()実行前	—
	XACM_E	P2	xa_commit()実行後	xa_commit()戻り値
	XARL_S	P2	xa_rollback()実行前	—
	XARL_E	P2	xa_rollback()実行後	xa_rollback()戻り値
	DISC_S	P2	disconnect 実行前	—
	DISC_E	P2	disconnect 実行後	SQLCODE
	SQPIPR	P2	更新情報読み込み	要求種別,read サイズ
	PWRT_A	P2	PRE-C 結果送信	実行結果,write サイズ
	SQPIPW	P2	COMMIT 結果送信	実行結果,write サイズ
	CUOC_S	P2	列 UOC 実行前	列 UOC 関数種別
	CUOC_E	P2	列 UOC 実行後	列 UOC 関数種別,列 UOC 関数実行結果
	CUOC_S	P2	列 UOC 実行前	列 UOC 関数種別
	CUOC_E	P2	列 UOC 実行後	列 UOC 関数種別,列 UOC 関数実行結果
	SQPIPW	P2	更新 (同期点まで) 実行結果の送信	実行結果,write サイズ
UOCG	UCINIT	ST	プロセスの開始	開始モード
	UCEXIT	ST	プロセスの停止	終了モード (NORMAL,FORCE,ERROR)
	UBE1_S	GL	hds_ubegin()呼び出し	—
	UEDT_S	GL	hds_uedit()呼び出し	—

プロセスコード	Cpn に表示される値	トレースレベル	意味	同時に出力される付加情報
	UEND_S	GL	hds_uend()呼び出し	—
	UCPIPR	P2	SQL 実行指示読み込み	—
	UBE1_E	P2	hds_ubegin()戻り	—
	UEDT_E	P2	hds_uedit()戻り	—
	UEND_E	P2	hds_uend()戻り	—
	UCPIPW	P2	更新（同期点まで）実行結果送信	—
	UCEREQ	GL	停止要求受付	—

出力例

反映側 Datareplicator で稼働トレースを取得した例を次に示します。ここで示す稼働トレースは、次に示す条件で取得したとします。

- 反映システム定義で、「int_trc_lvl = p1」を設定
- 「hdstrcredit -f reftrc.trcl -l perf1 -o reftrc.res」を実行

```
*****
**          HiRDB Datareplicator Internal trace information          **
**          Fri Mar 29 17:48:09 2002                                **
*****
No. 00000001 Lvl:GL Pid:02774,TRCR Cpn:QRCVST Date:2002-03-29 17:38:04.975462
  Unsendable-count:00000 Add-strings:Rcv Start Qid=00118
No. 00000002 Lvl:ST Pid:02776,DEFS Cpn:DFINIT Date:2002-03-29 17:38:05.169038
  Unsendable-count:00000 Add-strings:Defserv process start
  Additional-data:
    hdsdefserv dsid=BF
No. 00000003 Lvl:PI Pid:02888,RCVT Cpn:RCVDP1 Date:2002-03-29 17:43:39.776223
  Unsendable-count:00000 Add-strings:hds_rcv_rcvcnt
  Perf-info:2002-03-29 17:43:39.770903,2002-03-29 17:43:39.776223
  Sndid:----- Ujid:bf Seqn:000000000000000000000000000000000000000000000000000
  Additional-data:
    Recv Data : Port(def)
No. 00000004 Lvl:PI Pid:02888,RCVT Cpn:RCVDP1 Date:2002-03-29 17:43:39.781997
  Unsendable-count:00000 Add-strings:hds_rcv_rcvcnt
  Perf-info:2002-03-29 17:43:39.780873,2002-03-29 17:43:39.781997
  Sndid:----- Ujid:bf Seqn:000000000000000000000000000000000000000000000000000
  Additional-data:
    Recv Data : Definition(Single)
No. 00000005 Lvl:PI Pid:02888,RCVT Cpn:RCVDP1 Date:2002-03-29 17:46:46.366779
  Unsendable-count:00000 Add-strings:hds_rcv_rcvcnt
  Perf-info:2002-03-29 17:46:45.876897,2002-03-29 17:46:46.366779
  Sndid:----- Ujid:bf Seqn:534551493897eef400000011000202f17454100
  Additional-data:
    Recv Data : Extract data(Single)
No. 00000006 Lvl:LE Pid:02776,DEFS Cpn:DEFMKP Date:2002-03-29 17:47:05.341889
  Unsendable-count:00000 Add-strings:
  Additional-data:
    reflect start(mode=INIT,id=2896)
No. 00000007 Lvl:ST Pid:02896,RFCT Cpn:RFINIT Date:2002-03-29 17:47:05.452806
  Unsendable-count:00000 Add-strings:RFCT-PROC STRT(ca,bf)
No. 00000008 Lvl:GL Pid:02896,RFCT Cpn:RFSTRT Date:2002-03-29 17:47:05.463339
  Unsendable-count:00000 Add-strings:RFCT Mode=Initial,N
  Additional-data:
    Queue file position = [0, 0]
No. 00000009 Lvl:PI Pid:02896,RFCT Cpn:RFQGET Date:2002-03-29 17:47:05.483200
  Unsendable-count:00000 Add-strings:Ofst(Po)=0,512
  Sndid:----- Ujid:bf Seqn:000000000000000000000000000000000000000000000000000
  Additional-data:
    00000d8 e2000000 00000d8 e2e20000 *.
    00000000 00000000 00000000 00000000 *.
    00000000 00000000 00000001 00000001 *.
    00000000 00000000 00000000 00000000 *.
    00000000 00000000 *.
No. 00000010 Lvl:PI Pid:02896,RFCT Cpn:RFQGET Date:2002-03-29 17:47:05.495048
  Unsendable-count:00000 Add-strings:Ofst(Df)=0,1024
No. 00000011 Lvl:ST Pid:02896,RFCT Cpn:RFTERM Date:2002-03-29 17:47:05.521948
  Unsendable-count:00000 Add-strings:Term mode=Defserv
  Additional-data:
    Queue file position = [0, 1024]
No. 00000012 Lvl:ST Pid:02896,RFCT Cpn:RFEXIT Date:2002-03-29 17:47:05.533897
  Unsendable-count:00000 Add-strings:RFCT-PROC EXIT,RTN=0
```

```

No. 00000013 Lvl:LE Pid:02776, DEFS Cpn:DEFMKP Date:2002-03-29 17:47:06.754307
  Unsendable-count:00000 Add-strings:
  Additional-data:
    reflect start(mode=NORMAL, id=2897)
No. 00000014 Lvl:ST Pid:02897, RFCT Cpn:RFINIT Date:2002-03-29 17:47:06.804736
  Unsendable-count:00000 Add-strings:RFCT-PROC STRT(ca, bf)
No. 00000015 Lvl:GL Pid:02897, RFCT Cpn:RFSTRT Date:2002-03-29 17:47:06.826876
  Unsendable-count:00000 Add-strings:RFCT Mode=Normal, N
  Additional-data:
    Queue file position = [0, 2048]
No. 00000016 Lvl:P1 Pid:02897, RFCT Cpn:RFQGET Date:2002-03-29 17:47:06.827741
  Unsendable-count:00000 Add-strings:Ofst(Dt)=0, 2048
  Sndid:----- Ujid:bf Seqn:534551493897eef4000000011000202f17454100
  Additional-data:
    000001a4 e2100000 000001a4 d9e20000 *. . . . . *
    53455149 3897eef4 00000001 1000202f *SEQ18. . . . . /*
    17454100 d5000000 00000001 00000003 *.EA. . . . . *
    00000000 00000000 00000000 00000000 *. . . . . *
    00000000 00000000 *. . . . . *
No. 00000017 Lvl:P1 Pid:02897, RFCT Cpn:RFTRNC Date:2002-03-29 17:47:06.827987
  Unsendable-count:00000 Add-strings:Transaction entry loop
  Sndid:----- Ujid:bf Seqn:534551493897eef4000000011000202f17454100
  Additional-data:
    Current entry = 0, Data count = 3
No. 00000018 Lvl:ST Pid:02898, SQLE Cpn:SQINIT Date:2002-03-29 17:47:07.037222
  Unsendable-count:00000 Add-strings:SQLE process start
  Additional-data:
    hdssqle[trngroup] dsid=BF DistNum=1
No. 00000019 Lvl:P1 Pid:02897, RFCT Cpn:RFCOMT Date:2002-03-29 17:47:07.616639
  Unsendable-count:00000 Add-strings:Commit, Retrn=1
  Sndid:----- Ujid:bf Seqn:534551493897eef4000000011000202f17454100
No. 00000020 Lvl:P1 Pid:02897, RFCT Cpn:RFQGET Date:2002-03-29 17:47:07.637376
  Unsendable-count:00000 Add-strings:No data in QUEUE
  Additional-data:
    Queue offset=0, 2560
No. 00000021 Lvl:P1 Pid:02897, RFCT Cpn:RFQGET Date:2002-03-29 17:47:12.631778
  Unsendable-count:00000 Add-strings:No data in QUEUE
  Additional-data:
    Queue offset=0, 2560
No. 00000022 Lvl:P1 Pid:02897, RFCT Cpn:RFQGET Date:2002-03-29 17:47:17.631355
  Unsendable-count:00000 Add-strings:No data in QUEUE
  Additional-data:
    Queue offset=0, 2560
No. 00000023 Lvl:P1 Pid:02897, RFCT Cpn:RFQGET Date:2002-03-29 17:47:22.631228
  Unsendable-count:00000 Add-strings:No data in QUEUE
  Additional-data:
    Queue offset=0, 2560
No. 00000024 Lvl:P1 Pid:02897, RFCT Cpn:RFQGET Date:2002-03-29 17:47:27.631219
  Unsendable-count:00000 Add-strings:No data in QUEUE
  Additional-data:
    Queue offset=0, 2560
No. 00000025 Lvl:P1 Pid:02888, RCVT Cpn:RCVDP1 Date:2002-03-29 17:47:28.302308
  Unsendable-count:00000 Add-strings:hds_rcv_rcvcnt
  Perf-info:2002-03-29 17:47:28.126569, 2002-03-29 17:47:28.302308
  Sndid:----- Ujid:bf Seqn:000000000000000000000000000000000000000000000000000
  Additional-data:
    Recv Data : End
No. 00000026 Lvl:P1 Pid:02888, RCVT Cpn:RCVDP1 Date:2002-03-29 17:47:28.305020
  Unsendable-count:00000 Add-strings:hds_rcv_rcvcnt
  Perf-info:2002-03-29 17:47:28.304990, 2002-03-29 17:47:28.305020
  Sndid:----- Ujid:bf Seqn:000000000000000000000000000000000000000000000000000
  Additional-data:
    Recv Data : Close

```

```

No. 00000027 Lvl:LE Pid:02897,RFCT Cpn:RFSIGS Date:2002-03-29 17:47:31.089582
  Unsendable-count:00000 Add-strings:Recieve Signal=01
No. 00000028 Lvl:LE Pid:02776,DEFS Cpn:DEFCOM Date:2002-03-29 17:47:31.091007
  Unsendable-count:00000 Add-strings:Command accept
  Additional-data:
    command = hdsstop(normal)
No. 00000029 Lvl:P1 Pid:02897,RFCT Cpn:RFQGET Date:2002-03-29 17:47:31.091196
  Unsendable-count:00000 Add-strings:Ofst (En)=0, 2560
  Sndid:----- Ujid:bf Seqn:ffffffffffffffffffffffffffffffffffff
  Additional-data:
    00000040 e2200000 *...@...*
No. 00000030 Lvl:P1 Pid:02897,RFCT Cpn:RFQGET Date:2002-03-29 17:47:31.112715
  Unsendable-count:00000 Add-strings:No data in QUEUE
  Additional-data:
    Queue offset=0, 3072
No. 00000031 Lvl:ST Pid:02897,RFCT Cpn:RFTERM Date:2002-03-29 17:47:31.144381
  Unsendable-count:00000 Add-strings:Term mode=Normal
  Additional-data:
    Queue file position = [0, 2560]
No. 00000032 Lvl:P1 Pid:02897,RFCT Cpn:RFENDC Date:2002-03-29 17:47:31.144612
  Unsendable-count:00000 Add-strings:End Req,Retrn=1
  Sndid:----- Ujid:bf Seqn:ffffffffffffffffffffffffffffffffffff
No. 00000033 Lvl:GL Pid:02898,SQLE Cpn:SQREQ Date:2002-03-29 17:47:31.144853
  Unsendable-count:00000 Add-strings:END request accept
No. 00000034 Lvl:ST Pid:02898,SQLE Cpn:SQEXIT Date:2002-03-29 17:47:31.153876
  Unsendable-count:00000 Add-strings:SQLE process end
  Additional-data:
    exit code=1
No. 00000035 Lvl:ST Pid:02897,RFCT Cpn:RFEXIT Date:2002-03-29 17:47:31.176136
  Unsendable-count:00000 Add-strings:RFCT-PROC EXIT,RTN=0
No. 00000036 Lvl:ST Pid:02776,DEFS Cpn:DFEXIT Date:2002-03-29 17:47:32.204934
  Unsendable-count:00000 Add-strings:Defserv process end

```

(凡例)

No.00000002, 00000007, 00000014, 00000018 など

: 反映処理で各プロセスが起動されたことを示します。

No.00000003, 00000004, 00000025, 00000026

: 受信処理を実行したことを示します。Perf-info で受信動作に入った時刻とデータの受信が完了した時刻を表示します。

No.00000016, 00000019

: 反映キューファイルからデータを読み込んだ時刻と、トランザクションを完結 (COMMIT) した時刻を示します。識別名 (Cpn) で反映キューファイルの読み込みを示す「RFQGET」とコミットを示す「RFCOMT」の間のレコードが、反映処理のトランザクションの範囲です。この時刻の差分が、反映側システムでのトランザクション処理時間です。

No.00000031 以降

: 反映側システムの各プロセスの終了処理を示します。個々のプロセス ID がそれぞれの終了処理の時刻を示します。

稼働トレース取得レベルに「p2」を指定して、hdstrcredit コマンド実行時に-l perf2 を指定すると、上記の情報に加えて SQL 単位の発行・完了時刻が出力されます。

pdlogchg (ログ関係のファイルのステータスの変更)

このコマンドは HiRDB のコマンドです。規則や注意などの詳細については、マニュアル「HiRDB コマンドリファレンス」を参照してください。

機能

指定したファイルグループのステータスを強制的にアンロード済み状態又は抽出側 Datareplicator の抽出完了状態にします。

形式

```
pdlogchg -d sys [ -s サーバ名 ] -g ファイルグループ名 [ -R ]
```

オプション

抽出側 Datareplicator でデータ連動する場合に指定できるオプションについて説明します。このほかのオプションについては、マニュアル「HiRDB コマンドリファレンス」を参照してください。

- -R

HiRDB Datareplicator 連携をしている場合に、抽出未完了状態のファイルグループを、強制的に抽出完了状態に変更します。

このオプションを指定すると、抽出状態だけを抽出未完了状態から抽出完了状態に変更します。このオプションを省略すると、アンロード状態だけをアンロード待ち状態からアンロード済み状態に変更します。

pdls (HiRDB システムの状態の表示)

このコマンドは HiRDB のコマンドです。規則や注意などの詳細については、マニュアル「HiRDB コマンドリファレンス」を参照してください。

機能

指示した表示対象種別ごとに HiRDB システムの状態を表示します。

形式

```
pdls -d rpl [ -j ] [ -u ユニット識別子 ] [ -s サーバ名]
```

オプション

- -d 表示対象種別
～《svr》

状態を表示する対象種別を指定します。

rpl

HiRDB Datareplicator 連携の状態の表示

HiRDB Datareplicator 連携機能を使っているかどうかを表示します。また、システムログファイルでの抽出側 Datareplicator のシステムログの抽出状況を表示します。

- -j
システムログファイルでの抽出側 Datareplicator のシステムログの抽出状況を表示する場合に指定します。このオプションを指定した場合は、次の情報を表示します。
 - HiRDB Datareplicator 連携機能を使っているかどうか
 - 抽出側 Datareplicator のシステムログファイルからのシステムログの抽出状況
 - 抽出側 HiRDB のシステムログファイルへのシステムログの出力状況

このオプションを省略すると、HiRDB Datareplicator 連携機能を使っているかどうかだけを表示します。システムログの抽出状況と出力状況は表示されません。

- -u ユニット識別子
～〈4文字の識別子〉

あるユニットでの、抽出側 Datareplicator のシステムログの抽出状況を表示する場合に、ユニット識別子を指定します。

-u オプションを省略すると、システム内のすべてのサーバが表示対象になります。-j オプションを省略して -u オプションを指定すると、システム及びこのユニットが HiRDB Datareplicator 連携機能を使っているかどうかだけを表示します。

- -s サーバ名
～〈1～8文字の識別子〉

あるサーバでの、抽出側 Datareplicator のシステムログの抽出状況を表示する場合に、サーバ名を指定します。

-s オプションを省略すると、システム内のすべてのサーバが表示対象になります。

-j オプションを省略して -s オプションを指定すると、システム及びこのサーバの属するユニットが HiRDB Datareplicator 連携機能を使っているかどうかだけを表示します。

pdrplstart (HiRDB Datareplicator 連携の開始)

このコマンドは HiRDB のコマンドです。規則や注意などの詳細については、マニュアル「HiRDB コマンドリファレンス」を参照してください。

機能

HiRDB Datareplicator 連携を開始します。

形式

```
pdrplstart
```

pdrplstop (HiRDB Datareplicator 連携の中止)

このコマンドは HiRDB のコマンドです。規則や注意などの詳細については、マニュアル「HiRDB コマンドリファレンス」を参照してください。

機能

HiRDB Datareplicator 連携を中止します。

注意事項

このコマンドは、Datareplicator によるデータ連動を中止する場合にだけ使います。一時的な中断の目的では使わないでください。pdrplstop コマンドを実行すると、抽出側 DB と反映側 DB は不整合になります。

形式

```
pdrplstop [-f]
```

オプション

- -f

HiRDB Datareplicator のシステムログの抽出状況に関係なく、HiRDB Datareplicator 連携を強制終了する場合に指定します。停止中のユニットがある場合でも、HiRDB Datareplicator 連携を終了します。

8

ユーザOWNコーディング

ユーザOWNコーディング（UOC）を作成すると、Datareplicator のデータ連動の業務を補助できます。

この章では、Datareplicator で使えるユーザOWNコーディング（UOC）の概要、作成方法、関数の文法、及び UOC のサンプルについて説明します。

8.1 反映情報編集 UOC

反映情報編集 UOC を作成すると、更新情報を編集して反映側 DB の表に任意に反映できます。反映側 Datareplicator は表単位反映方式で反映処理を実行します。

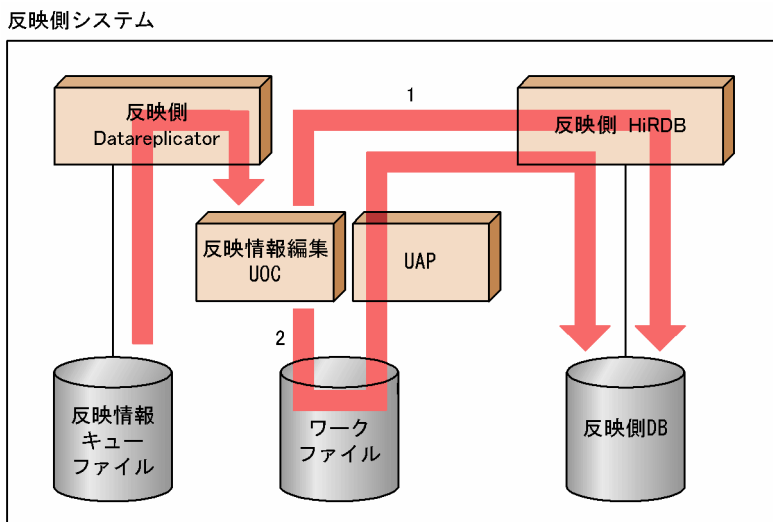
8.1.1 反映情報編集 UOC の概要

反映情報編集 UOC を使って反映側 DB に更新情報を反映する方法には、次の二つがあります。

- SQL の発行
- ワークファイルへの出力

更新情報を反映する手順を次の図に示します。

図 8-1 更新情報を反映する手順



1. SQL の発行

反映側 Datareplicator から受け取った更新情報を基に反映情報編集 UOC で SQL を発行し、反映側 DB に反映します。

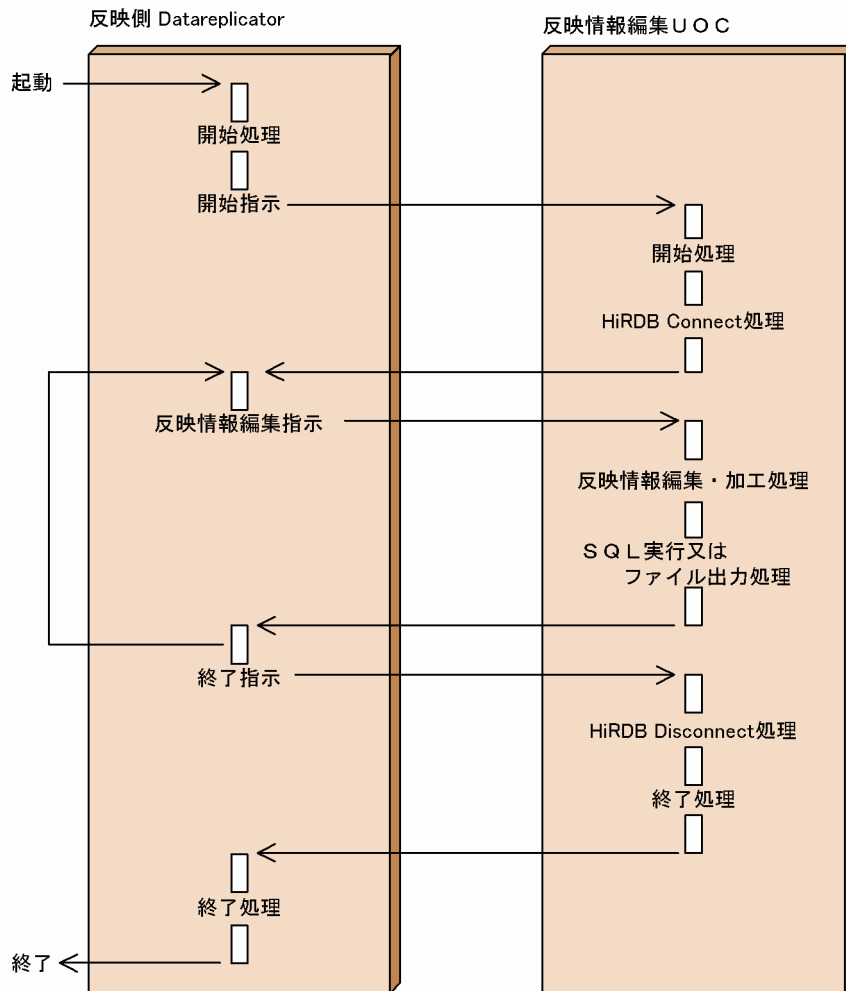
2. ワークファイルへの出力

反映側 Datareplicator から受け取った更新情報を反映情報編集 UOC で編集した後、ワークファイルへ出力します。UAP を使ってワークファイルから更新情報を取り出し、反映側 DB へ反映します。

(1) 反映情報編集 UOC の制御の流れ

反映情報編集 UOC と反映側 Datareplicator との制御の流れを次の図に示します。

図 8-2 反映情報編集 UOC の制御の流れ



(凡例)

□ : 処理中を表します。

開始指示

反映側 Datareplicator を起動すると、反映処理起動の延長で UOC に対して開始指示の発行 (UOC の `hds_ubegin()` の呼び出し) を行います。

反映情報編集指示

抽出側から送られてきた更新情報を反映プロセスが読み込み、読み込んだ更新情報 1 件ごとに更新情報編集指示の発行 (UOC の `hds_uedit()` の呼び出し) を行います。

終了指示

反映側 Datareplicator を停止すると、反映処理停止の延長で UOC に対して終了指示の発行 (UOC の `hds_uend()` の呼び出し) を行います。

反映情報編集 UOC の各関数は、次のタイミングで呼び出されます。

- `hds_ubegin()`

反映処理の起動時 (hdsstart, hdsrfctl コマンドによる反映プロセスの起動, 又は定義変更検知※, モード変更イベント検知による反映プロセスの再起動) にだけ呼び出されます。

- hds_uedit()

更新情報 1 件ごとに呼び出されます。

- hds_uend()

反映処理の停止時 (hdsstop, hdsrfctl コマンドによる反映プロセスの停止, 又は定義変更検知※, モード変更イベント検知による反映プロセスの一時停止) にだけ呼び出されます。

注※ 定義変更検知は, XDM/DS で選択定義を使用している場合, 定義変更していない場合でも, 反映側システムと再接続するたびに定義が送信され hds_ubegin(), hds_uend() を呼び出します。

つまり, 通常の処理 (反映側 Datareplicator を 1 回起動してから停止するまでの間) での各関数の実行回数は, 次のようになります (n は更新回数を表します)。

- hds_ubegin() : 1 回
- hds_uedit() : 1 ~ n 回
- hds_uend() : 1 回

(2) 更新情報が渡される単位

更新情報は 1SQL 単位で, 反映側 Datareplicator から反映情報編集 UOC に渡されます。

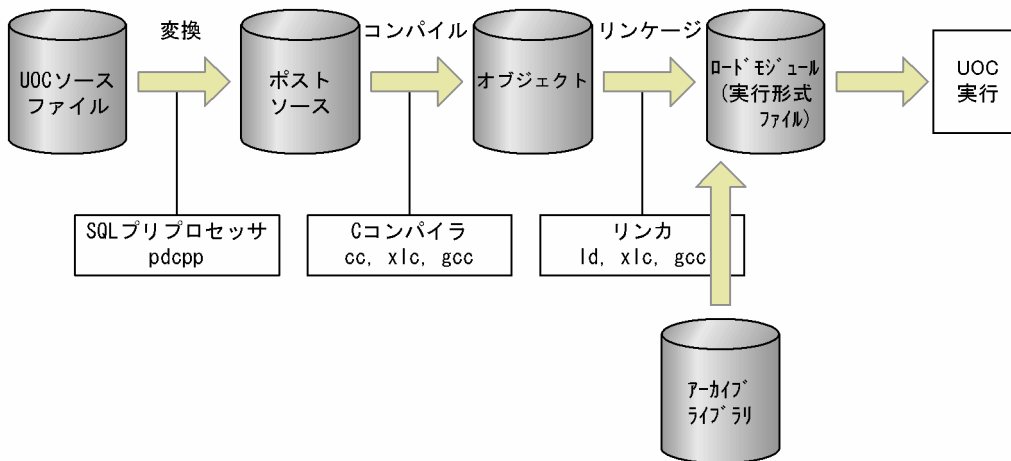
同期点処理の COMMIT 要求は, 単独で渡されます。

8.1.2 反映情報編集 UOC の作成手順 (UNIX)

UNIX 版 Datareplicator で反映情報編集 UOC を作成する手順について説明します。

反映情報編集 UOC では SQL を使う場合, 直接コンパイルを実行できません。いったん C コンパイラがコンパイルできるソースに変換した後, コンパイルする必要があります。UOC を実行するまでの手順を次の図に示します。

図 8-3 反映情報編集 UOC の実行手順



(1) 作成方法

反映情報編集 UOC インタフェース共有ライブラリの作成方法を次に示します。

1. 反映情報編集 UOC インタフェースを使う反映列を決定します。
2. 反映情報編集 UOC インタフェースの関数を作成します。
3. 反映情報編集 UOC で SQL を使用する場合、反映側 HiRDB の SQL プリプロセッサを使って、C コンパイラがコンパイルできるソースファイルに変換します。
SQL プリプロセッサについては、マニュアル「HiRDB UAP 開発ガイド」を参照してください。
4. 反映情報編集 UOC インタフェースの関数をコンパイル／リンケージします。

(2) コンパイル及びリンケージ

コンパイル及びリンケージについて OS ごとに説明します。

なお、リンケージの未解決エラーが発生する場合は、リンケージのときに必要なライブラリすべてを指定してください。

(a) HP-UX 版の場合

- UOC のコンパイル方法

```
cc -c +DD64 -I /HiRDB/include -I /opt/hirdbds/include UOCソースファイル名
```

-c :

オブジェクト作成オプション

-I /HiRDB/include :

下線部分に、反映側 HiRDB のインストールディレクトリを指定します。

-I /opt/hirdbds/include :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。
UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

UOC ソースファイル名 :

SQL プリプロセサを使って、C コンパイラがコンパイルできるソースファイルに変換した名称を指定
します。

コンパイルコマンドの文法については、該当する OS のリファレンスマニュアルを参照してください。

• UOC のリンケージ方法

```
ld -o UOC名 -u main UOCオブジェクト名  
/opt/hirdbds/lib/libhdsuif.a /opt/hirdbds/lib/libhdsulb.so  
/opt/hirdbds/lib/libhdscom.so /opt/hirdbds/lib/libhdspsc.so  
-lc -L /HiRDB/client/lib -lzcltk64
```

注 1

UOC 名には、実行形式ファイル名を指定します。この実行形式ファイル名を、反映定義に指定しま
す。UOC の実行時には、実行形式ファイルを格納したディレクトリのパス名を環境変数 PATH に設
定しておいてください。

UOC オブジェクト名には、コンパイルして作成したオブジェクトファイル名を指定します。

/opt/hirdbds/lib 下のライブラリは、反映側 Datareplicator が用意しているライブラリです。これら
のライブラリは必ず指定してください。

注 2

リンケージのコマンドに指定する実行ファイル名は、hds で始まる名称 又はシステムにインストール
済みのプログラムで禁止している名称を指定しないでください。

注 3

-L オプションの行は、UOC 内から SQL を実行するとき指定するライブラリです。下線で示す部分
には、反映側 HiRDB をインストールしたディレクトリを指定します。

(b) AIX 版の場合

• UOC のコンパイル方法

```
xlc -q64 -c -I /HiRDB/include -I /opt/hirdbds/include UOCソースファイル名
```

-c :

オブジェクト作成オプション

-I /HiRDB/include :

下線部分に、反映側 HiRDB のインストールディレクトリを指定します。

-I /opt/hirdbds/include :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。
UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

UOC ソースファイル名：

SQL プリプロセサを使って、C コンパイラがコンパイルできるソースファイルに変換した名称を指定します。

コンパイルコマンドの文法については、該当する OS のリファレンスマニュアルを参照してください。

• UOC のリンケージ方法

```
xlc -q64 -o UOC名 -u main UOCオブジェクト名  
/opt/hirdbds/lib/libhdspsc.a /opt/hirdbds/lib/libhdsulb.a  
/opt/hirdbds/lib/libhdscom.a /opt/hirdbds/lib/libhdsuif.a  
-lm -lc -L /HiRDB/client/lib -WL,-bldpath:./usr/lib:/lib:/HiRDB/client/lib -lzcltk64
```

注 1

UOC 名には、実行形式ファイル名を指定します。この実行形式ファイル名を、反映定義に指定します。UOC の実行時には、実行形式ファイルを格納したディレクトリのパス名を環境変数 PATH に設定しておいてください。

UOC オブジェクト名には、コンパイルして作成したオブジェクトファイル名を指定します。

/opt/hirdbds/lib 下のライブラリは、反映側 Datareplicator が用意しているライブラリです。これらのライブラリは必ず指定してください。

注 2

リンケージのコマンドに指定する実行ファイル名は、hds で始まる名称 又はシステムにインストール済みのプログラムで禁止している名称を指定しないでください。

注 3

-L オプションの行は、UOC 内から SQL を実行するとき指定するライブラリです。下線で示す部分には、反映側 HiRDB をインストールしたディレクトリを指定します。

(c) Linux 版の場合

• UOC のコンパイル方法

```
gcc -c -I /HiRDB/include -I /opt/hirdbds/include UOCソースファイル名
```

-c：

オブジェクト作成オプション

-I /HiRDB/include：

下線部分に、反映側 HiRDB のインストールディレクトリを指定します。

-I /opt/hirdbds/include：

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

UOC ソースファイル名：

SQL プリプロセサを使って、C コンパイラがコンパイルできるソースファイルに変換した名称を指定します。

コンパイルコマンドの文法については、該当する OS のリファレンスマニュアルを参照してください。

- UOC のリンケージ方法

```
gcc -o UOC名 -u main UOCオブジェクト名
/opt/hirdbds/lib/libhdsuif.a /opt/hirdbds/lib/libhdsulb.so
/opt/hirdbds/lib/libhdscom.so /opt/hirdbds/lib/libhdspsc.so
-lpthread -ldl
-L/HiRDB/client/lib -lzclt64 -WL,-R/HiRDB/client/lib
```

注 1

-R オプションは省略できます。ただし、省略した場合は、LD_LIBRARY_PATH に「HiRDB をインストールしたディレクトリ/client/lib」を追加しておく必要があります。

注 2

-l は、反映側 HiRDB のリンケージライブラリ名を指定します。

注 3

-L オプションの行は、UOC 内から SQL を実行するときに指定するライブラリです。下線で示す部分には、反映側 HiRDB をインストールしたディレクトリを指定します。

8.1.3 反映情報編集 UOC の作成手順 (Windows)

Windows 版 Datareplicator で反映情報編集 UOC を作成する手順について説明します。Windows 版 Datareplicator の UOC は、DLL ファイル (xxx.dll) として作成します。

(1) 作成方法

Windows 版の場合、反映情報編集 UOC は Visual Studio で作成します。

次に示す手順で、DLL ファイルを作成してください。

1. 反映情報編集 UOC を使う反映表を決定します。
2. Windows 版 HiRDB のアプリケーションプログラムを作成する C 言語で反映情報編集 UOC の関数を作成します。
UOC から SQL を発行するときは、SQL を各言語に展開するために HiRDB のプリプロセサを使います。Datareplicator の UOC 用のヘッダファイルは、Datareplicator のインストール先ディレクトリの下に `¥include` に格納してあります。
3. 作成した関数をコンパイル／リンケージして、DLL ファイルを作成します。Datareplicator から UOC を呼び出すため、インタフェース関数は必ず `_cdecl` 呼び出し規約で作成して、エクスポートしておいてください。
4. 反映情報編集 UOC の名称を、反映定義に絶対パス名で指定します。

(2) コンパイル及びリンケージ

開発環境でのコンパイル，及びリンケージ方法について説明します。

Visual Studio を使用してコンパイル，リンケージをする場合，オプションの設定は，プロジェクトメニューから「プロパティ」を選択します。

設定する項目を次の表に示します。

表 8-1 Visual Studio の「プロパティ」で設定する項目

項目	カテゴリ	カテゴリの設定	設定値
プラットフォーム	—	—	x64
構成プロパティ	詳細	共通言語ランタイムサポート	共通言語ランタイムサポートを使用しない
C/C++	コード生成	構造体メンバのアラインメント	8 バイト (/Zp8)
		ランタイムライブラリ	マルチスレッド DLL (/MD)
	詳細設定	呼び出し規約	__cdecl (/Gd)
	全般	追加のインクルードディレクトリ	Datareplicator インストールディレクトリ¥include
リンカー	全般	追加のライブラリディレクトリ	HiRDB インストールディレクトリ¥CLIENT¥LIB (UOC から SQL を発行する場合に指定)
	入力	追加の依存ファイル	PDCLTM140X.LIB (UOC から SQL を発行する場合に指定)

(凡例)

—：該当しません。

リンケージの未解決エラーが発生する場合は，リンケージのときに必要なライブラリすべてを指定してください。

(3) 注意事項

- 反映側 Datareplicator 又はほかのシステムで使っているメモリマップトファイルは，UOC では使わないでください。
- UOC の処理がエラーになったときは，エラーログファイルのプログラム名には DLL ファイルの名称が出力されます。
- UOC から SQL を発行する場合は，HiRDB が提供するインクルードファイル，インポートライブラリを，それぞれコンパイル，リンクの際に指定してください。

8.1.4 反映情報編集 UOC で使う関数の文法

反映情報編集 UOC は、C 言語を使ってプログラミングします。反映側 Datareplicator には、反映情報編集 UOC を作成するときを使う関数を備えています。

反映情報編集 UOC で使える関数名と機能の概要を次の表に示します。

表 8-2 反映情報編集 UOC で使える関数と機能の概要

関数名	機能
hds_ubegin() (編集開始指示)	反映情報の編集を開始することを指示します。ユーザは、反映側 HiRDB への接続処理又は環境の初期設定の処理を実行します。
hds_uedit() (編集・加工指示)	反映情報の編集・加工及び同期点処理を指示します。ユーザは、反映情報を編集・加工して SQL を発行したり一般ファイルへ出力したりします。
hds_uend() (編集終了指示)	反映情報の編集を終了することを指示します。ユーザは、反映側 HiRDB からの切り離し処理又は環境の終了処理を実行します。

(1) hds_ubegin() (編集開始指示関数)

(a) 関数の形式

```
#include<hds_ucommon.h>
#include<hds_ureflect.h>
int hds_ubegin (UINTERFACE_BLK *インタフェースブロック名,
int *ステータス) ;
```

(b) パラメタの説明

- UINTERFACE_BLK *インタフェースブロック名

hds_uedit()関数又は hds_uend()関数に情報を引き継ぎたい場合に設定する情報を格納するための領域の名称を指定します。UINTERFACE_BLK は呼び出し元が確保します。インタフェースブロックの内容を次の表に示します。

表 8-3 インタフェースブロック (UINTERFACE_BLK) の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
auth_id	8	char	認可識別子	呼び出し元	反映システム定義で指定した接続認可識別子
password	30	char	パスワード	呼び出し元	反映システム定義で指定したパスワード
exsysid	2	char	抽出側 Datareplicator 識別子	抽出元	抽出側 Datareplicator の抽出システム定義で指定した抽出側 Datareplicator 識別子。小文字の 16 進数字が文字列として設定されます。 XDM/DS との連動時は、'00'が設定されます。
repid1	2	char	識別子 1	呼び出し元	反映システム定義で指定したデータ連動識別子

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
repid2	2	char	識別子 2	呼び出し元	反映システム定義で指定した反映側 Datareplicator 識別子
inherinf1	8	char	引き継ぎ情報 1	呼び出し先	hds_uedit()関数と hds_uend()関数に引き継ぐ領域のアドレス
inherinf2	4	int	引き継ぎ情報 2	呼び出し先	hds_uedit()関数と hds_uend()関数に引き継ぐ情報
stopinf	4	int	終了モード	呼び出し元	UOC 終了時の終了モード この領域は、hds_uend()関数の呼び出し時だけ有効です。 HDS_FLG_NORMAL 正常終了 HDS_FLG_FORCE 強制停止

注 1

反映側 Datareplicator では、UINTERFACE_BLK のヘッダファイルを用意しているため、それを使ってください。ヘッダファイルについては、「表 8-13 反映情報編集 UOC のヘッダファイルの一覧」を参照してください。

注 2

ユーザは、設定者が呼び出し元になっている領域を書き換えしないでください。

• int *ステータス

編集開始指示関数の中でエラーを検知した場合、ステータスに値を設定できます。ステータスに値を設定すると、呼び出し元である Datareplicator からメッセージが出力され、その中にこのステータス値が表示されます。

(c) 戻り値

呼び出し元に戻るときに戻り値を設定します。設定できる戻り値を次の表に示します。

表 8-4 hds_ubegin()からの戻り値

コード	状態	ニモニック	呼び出し元の処置
0	正常終了	HDS_RET_NORM	処理を続行します。
4	軽度 (継続できるレベル) のエラー検知	HDS_RET_WARN	メッセージを出力します。処理は続行します。
0, 4 以外	重度 (継続できないレベル) のエラー検知	HDS_RET_ERR	メッセージを出力します。処理は中止し、プロセスを終了します。

(d) 抽出側 Datareplicator を使うときの注意

抽出側 Datareplicator の送信環境定義の nsndidxxx オペランド (xxx は 001~256 の整数) に、送信抑止受信元識別子を指定している場合、HiRDB の CONNECT 処理の直前に、HiRDB の環境変数である \$PDCLTAPNAME に、hdssqle で始まり抽出側 Datareplicator 識別子で終わる 20 バイト以内の文字列を設定してください。

抽出側 Datareplicator 識別子として設定される文字列は、インタフェースブロック内に設定される文字列と同じです (2 バイトの 16 進数字の文字列)。

(2) hds_uedit() (編集・加工指示関数)

(a) 関数の形式

```
#include<hds_uchcommon.h>
#include<hds_ureflect.h>
int hds_uedit (UINTERFACE_BLK *インタフェースブロック名,
              UREFLECT_BLK *反映制御ブロック名,
              UDATA_BLK *キー情報ブロック名,
              UDATA_BLK *反映データブロック名,
              int *ステータス);
```

(b) パラメタの説明

- UINTERFACE_BLK *インタフェースブロック名

hds_begin()関数からの情報を引き継ぎたい場合に設定する情報を格納するための領域の名称を指定します。インタフェースブロック名については、「表 8-3 インタフェースブロック (UINTERFACE_BLK) の内容」を参照してください。

- UREFLECT_BLK *反映制御ブロック名

反映情報の共通情報を格納するための領域の名称を指定します。反映制御ブロックは呼び出し元が確保します。反映制御ブロックの内容を次の表に示します。

表 8-5 反映制御ブロック (UREFLECT_BLK) の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
updkind	8	char	更新種別	呼び出し元	抽出側 DB を更新するときの種別。次に種別を示します。種別を示す文字列は、8 バイト固定です。8 バイトに満たない部分は、空白で埋められます。 INSERT INSERT を示します。 UPDATE UPDATE を示します。 DELETE DELETE を示します。

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
					PURGE PURGE を示します。 COMMIT COMMIT を示します。* 注※ この値の格納時は、認可識別子以降の項目の値は意味がありません。
auth_id	30	char	認可識別子	呼び出し元	抽出側 DB を更新するときの認可識別子
tbl_id	30	char	表識別子	呼び出し元	抽出側 DB を更新するときの表識別子
upd_date	4	char	更新日付	呼び出し元	抽出側 DB を更新するときの日付 (更新情報をジャーナルに出力した日付)。日付は 10 進符号なしパック形式 (YYYYMMDD) で出力します。 YYYY：西暦年 MM：月 DD：日
upd_time	4	char	更新時刻	呼び出し元	抽出側 DB 更新時の時刻 (更新情報をジャーナルに出力した時刻) 時刻は 10 進符号なしパック形式 (HHMMSSTT) で出力します。 HH：時 MM：分 SS：秒 TT：1/100 秒
uapname	8	char	抽出側の UAP 名	呼び出し元	抽出側 DB を更新するときのユーザアプリケーションプログラムの名称
extsysid	4	char	抽出側 DB のシステム ID	呼び出し元	抽出側 DB が HiRDB の場合 抽出側 DB の HiRDB 識別子 抽出側 DB がメインフレーム (XDM/DS, PDMII E2, RDB1 E2) の場合 抽出側 DB のサブシステム ID
infflag1	1	unsigned char	予備領域 1	—	予備
infflag2	1	unsigned char	予備領域 2	—	予備
infflag3	1	unsigned char	予備領域 3	—	予備
reserve1	1	char	予備領域 4	—	予備
infflag4	4	int	予備領域 5	—	予備

注 1

反映側 Datareplicator では、UREFLECT_BLK のヘッダファイルを用意しているのので、それを使ってください。ヘッダファイルについては、「表 8-13 反映情報編集 UOC のヘッダファイルの一覧」を参照してください。

注 2

ユーザは、設定者が呼び出し元になっている領域を書き換えしないでください。

- UDATA_BLK *キー情報ブロック名
- UDATA_BLK *反映データブロック名

反映情報を格納するための領域の名称を指定します。hds_uedit() の第 3 パラメタのキー情報ブロックと第 4 パラメタの反映データブロックの両方でこの構造体を使います。

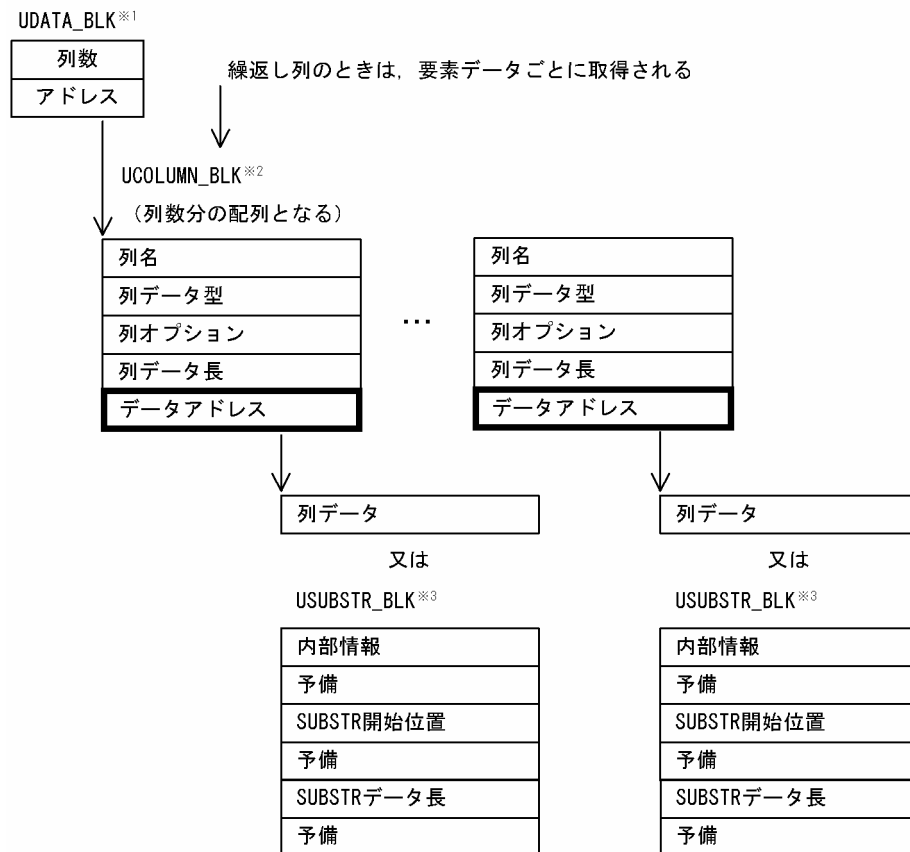
キー情報ブロックでは、UDATA_BLK にマッピングキーの情報を格納します。反映データブロックでは、UDATA_BLK に反映データの情報を格納します。

キー情報ブロックは、INSERT、PURGE 及び COMMIT の場合には列数に 0 が設定されます。また、キー値が変更された場合には、キー情報ブロックには更新前のキー値が設定されます。更新後のキー値及び INSERT 時のキー値は、反映データブロックの中に設定されます。

DELETE、PURGE、COMMIT の場合は、反映データブロックの列数に 0 が設定されます。

キー情報及び反映データ情報は、UDATA_BLK と UDATA_BLK からポイントされる構造体（これを UCOLUMN_BLK といいます）とスカラデータから構成されます。UDATA_BLK の列数に 0 が設定されている場合、UCOLUMN_BLK は参照しないでください。UDATA_BLK とその関連構造を次の図に示します。

図 8-4 UDATA_BLK とその関連構造



注※1

UDATA_BLK の内容については、「表 8-6 UDATA_BLK の内容」を参照してください。

注※2

UCOLUMN_BLK の内容については、「表 8-7 UCOLUMN_BLK の内容」を参照してください。

注※3

BLOB 型又は BINARY 型の後方削除更新をしている場合のデータ構造です。USUBSTR_BLK の内容については、「表 8-8 USUBSTR_BLK の内容」を参照してください。

表 8-6 UDATA_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
colnum	4	int	列数	呼び出し元	マッピングキーの列数又は反映データの列数が設定されます。
colinfptr	8	UCOLUMN_BLK	UCOLUMN_BLK*アドレス	呼び出し元	UCOLUMN_BLK のアドレスが設定されます。

注

Datereplicator では、UDATA_BLK のヘッダファイルを用意しているので、それを使ってください。ヘッダファイルについては、「表 8-13 反映情報編集 UOC のヘッダファイルの一覧」を参照してください。

表 8-7 UCOLUMN_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
colname	30	char	列名	呼び出し元	列の名前が設定されます。
coltype	1	unsigned char	列データ型	呼び出し元	列のデータ型が設定されます。設定内容については、「表 8-9 列データ型のニモニック」を参照してください。
coloption	1	unsigned char	列オプション	呼び出し元	列のオプションが設定されます。オプションの種類を次に示します。 HDS_COL_NOOPT (0x00) : オプションなし HDS_COL_NULL (0x01) : NULL データ HDS_COL_MAPK (0x02) : マッピングキー 反映データブロックだけ設定されます。キー情報ブロックには設定されません。 HDS_COL_ARRAY (0x04) : 配列列 HDS_COL_REPET (0x08) : 繰返し列 HDS_COL_ELMNL (0x10) : 繰返し列の要素の値が NULL HDS_COL_REPTCOL (0x20) : 繰返し構造指定 HDS_COL_BLOBSUB (0x40) : 連結演算データ HDS_COL_SUBSTR (0x80) : SUBSTR 演算後方削除更新ログ
elementnum	4	union	繰返し列の有効要素番号 又は抽象データ型の数	呼び出し元	繰返し列のとき： 前 2 バイト (short) には有効要素番号が設定されます。*指定のときは -2 が設定されます。 後 2 バイト (short) には該当する列の要素データの数が設定されます。 抽象データ型の列のとき： 継承関係を含めたデータ型の数が設定されます。継承がないときは 1 が設定されます。 配列列のとき： 前 2 バイトには抽出側配列列の要素番号が設定されます。 後 2 バイトには 0 が設定されます。

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
mltcolkind	1	unsigned char	繰返し列の更新種別	呼び出し元	繰返し列の場合の更新種別を次に示します。更新種別が UPDATE のときだけ有効になります。 HDS_COL_ADDV (0x01) : ADD HDS_COL_DELV (0x02) : DELETE HDS_COL_SETV (0x04) : SET
adtfunc	1	char	抽象データ型の反映方式	呼び出し元	「0x01」で固定です。
charset	1	char	文字集合種別	呼び出し元	UOC に渡す列データの文字集合種別が設定されます。 文字集合指定なし : X'00' (HDS_CSET_DEFAULT) 文字集合指定あり (EBCDIK) : X'01' (HDS_CSET_EBCDIK)
reserve1	2	char	予備領域	—	予備
collen	4	int	列データ長	呼び出し元	列のデータ長が設定されます。列オプションが NULL データの場合、0 が設定されます。
dataptr	4	union	データアドレス	呼び出し元	列データのアドレスが設定されます。列オプションが NULL データの場合、NULL ポインタが設定されます。設定内容については、「表 8-10 データアドレスのニモニック」を参照してください。

注

反映側 Datareplicator では、UCOLUMN_BLK のヘッダファイルを用意しているため、それを使用してください。ヘッダファイルについては、「表 8-13 反映情報編集 UOC のヘッダファイルの一覧」を参照してください。

表 8-8 USUBSTR_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
reserve1	4	int	内部情報 1	呼び出し元	内部情報
data_len	4	int	内部情報 2	呼び出し元	内部情報
reserve2	4	int	予備領域 1	呼び出し元	予備
substr_spos	4	unsigned int	SUBSTR 開始位置	呼び出し元	スカラ関数 SUBSTR 使用時の、切り出すデータ列の開始位置
reserve3	4	int	予備領域 2	呼び出し元	予備
substr_len	4	unsigned int	SUBSTR データ長	呼び出し元	スカラ関数 SUBSTR 使用時の、切り出すデータ列の長さ
reserve4	16	char	予備領域 3	呼び出し元	予備

列データ型のニモニックを「表 8-9 列データ型のニモニック」に、データアドレスのニモニックを「表 8-10 データアドレスのニモニック」に示します。

表 8-9 列データ型のニモニック

列データ型	ニモニック	データコード
INTEGER	HDS_T_INT	(0xF1)
SMALLINT	HDS_T_SINT	(0xF5)
LARGE DECIMAL	HDS_T_DEC	(0xE5)
FLOAT	HDS_T_FLT	(0xE1)
DOUBLE PRECISION	HDS_T_DBL	(0xE1)
SMALL FLOAT	HDS_T_SFLT	(0xE3)
REAL	HDS_T_REAL	(0xE3)
CHARACTER	HDS_T_CHAR	(0xC5)
VARCHAR	HDS_T_VCHAR	(0xC1)
NCHAR	HDS_T_NCHAR	(0xB5)
NVARCHAR	HDS_T_NVCHAR	(0xB1)
MCHAR	HDS_T_MCHAR	(0xA5)
MVCHAR	HDS_T_MVCHAR	(0xA1)
DATE	HDS_T_DATE	(0x71)
TIME	HDS_T_TIME	(0x79)
TIMESTAMP	HDS_T_TIMESTAMP	(0x7D)
INTERVAL YEAR TO DAY	HDS_T_YTD	(0x65)
INTERVAL HOUR TO SECOND	HDS_T_HTS	(0x6F)
BLOB	HDS_T_BLOB	(0x93)
BINARY	HDS_T_BINARY	(0x91)
UNPACK	HDS_T_UNPACK	(0xFF)

表 8-10 データアドレスのニモニック

データ型	ニモニック	アドレスタイプ
INTEGER	HDS_A_INT	int*
SMALLINT	HDS_A_SINT	short*
LARGE DECIMAL	HDS_A_DEC	char*
FLOAT	HDS_A_FLT	double*

データ型	ニモニック	アドレスタイプ
DOUBLE PRECISION	HDS_A_DBL	double*
SMALL FLOAT	HDS_A_SFLT	float*
REAL	HDS_A_REAL	float*
CHARACTER	HDS_A_CHAR	char*
VARCHAR	HDS_A_VCHAR	char*
NCHAR	HDS_A_NCHAR	char*
NVARCHAR	HDS_A_NVCHAR	char*
MCHAR	HDS_A_MCHAR	char*
MVCHAR	HDS_A_MVCHAR	char*
DATE	HDS_A_DATE	char*
TIME	HDS_A_TIME	char*
INTERVAL YEAR TO DAY	HDS_A_YTD	char*
INTERVAL HOUR TO SECOND	HDS_A_HTS	char*
BLOB	HDS_A_BLOB	char*
UNPACK	HDS_A_UNPACK	char*
TIMESTAMP	HDS_A_TIMESTAMP	char*
BINARY	HDS_A_BINARY	char*
SUBSTR 演算後方削除更新ログ	HDS_A_SUBSTR	USUBSTR_BLK*

• int *ステータス

編集・加工指示関数の中でエラーを検知した場合、ステータスに値を設定できます。ステータスに値を設定すると、関数から戻った後に呼び出し元であるインタフェースプロセスからメッセージを出力し、その中にステータス値を表示します。

(c) 戻り値

呼び出し元に戻るときに戻り値を設定します。設定できる戻り値を次の表に示します。

表 8-11 hds_uedit()からの戻り値

コード	状態	ニモニック	呼び出し元の処置
0	正常終了	HDS_RET_NORM	処理を続行します。
4	軽度（継続できるレベル）のエラー検知	HDS_RET_WARN	メッセージを出力します。処理は続行されます。
0, 4 以外	重度（継続できないレベル）のエラー検知	HDS_RET_ERR	メッセージを出力します。処理は中止して、プロセスを終了します。

(3) hds_uend() (編集終了指示関数)

(a) 関数の形式

```
#include<hds_ucommon.h>
#include<hds_ureflect.h>
int hds_uend (UINTERFACE_BLK *インタフェースブロック名,
             int *ステータス);
```

(b) パラメタの説明

- UINTERFACE_BLK *インタフェースブロック名

hds_begin()関数又はhds_edit()関数からの情報を引き継ぎたい場合に設定する情報を格納するための領域の名称を指定します。インタフェースブロック名については、「表 8-3 インタフェースブロック (UINTERFACE_BLK) の内容」を参照してください。

hds_ubegin()関数で動的に領域を割り当てた場合には、hds_uend()関数の中で解放してください。また、オープンしたファイルがある場合にはクローズしてください。hds_uend()関数から戻った後は、UINTERFACE_BLK で設定されているユーザ設定項目に対しては何もしないで、構造体の領域を解放してください。

データベースアクセスしている UOC で終了モードが強制停止のときは、DBMS に ROLLBACK を実行してから DISCONNECT を実行してください。

- int *ステータス

編集終了指示関数の中でエラーを検知した場合、ステータスに値を設定できます。ステータスに値を設定すると、関数から戻った後に呼び出し元であるインタフェースプロセスからメッセージを出力し、その中にこのステータス値を表示します。

(c) 戻り値

呼び出し元に戻るときに戻り値を設定します。設定できる戻り値を次の表に示します。

表 8-12 hds_uend()関数からの戻り値

コード	状態	二モニック	呼び出し元の処置
0	正常終了	HDS_RET_NORM	正常にプロセスを終了させます。
4	軽度 (継続できるレベル) のエラー検知	HDS_RET_WARN	メッセージを出力します。プロセスは正常に終了させます。
0, 4 以外	重度 (継続できないレベル) のエラー検知	HDS_RET_ERR	メッセージを出力します。プロセスは正常に終了させます。

(4) 反映情報編集 UOC で使うヘッダファイル

反映側 Datareplicator では、hds_ubegin(), hds_uedit(), hds_uend()の関数のインタフェースになる構造体及び二モニックコードをヘッダファイルで用意しています。

反映情報編集 UOC で使うヘッダファイルは、次に示すディレクトリに格納してあります。

UNIX 版の場合： /opt/hirdbds/include/

Windows 版の場合：Datareplicator をインストールしたディレクトリ¥include¥

反映情報編集 UOC のヘッダファイルの一覧を次の表に示します。

表 8-13 反映情報編集 UOC のヘッダファイルの一覧

ファイル名	内容	反映情報編集 UOC に関連するメンバ
hds_ucommon.h	反映情報編集 UOC の共通の構造体及びニモニックコード	<ul style="list-style-type: none">• UINTERFACE_BLK• 戻り値のニモニック
hds_ureflect.h	反映情報編集 UOC の構造体及びニモニックコード	<ul style="list-style-type: none">• UREFLECT_BLK• UREFLECT_BLK のニモニック• UDATA_BLK• UCOLUMN_BLK• データ型のニモニック• データアドレスのニモニック

(5) 規則

(a) ファイル

反映情報編集 UOC の関数の中で、操作できるファイルはユーザ独自のファイル（オープンしたファイル）だけです。ファイルの規則を次に示します。

- 標準入力、標準出力、標準エラー出力は、使わないでください。
ただし、反映側 Datareplicator の環境変数 HDS_MST_STDCLOSE に「FALSE」を指定した場合は、標準入力、標準出力、標準エラー出力を使用できます。
- 自身でオープンしたファイル以外を書き込んだり読み込んだりしないでください。
- 反映側 Datareplicator のファイルを操作しないでください。

(b) 定義との関連

- UOC を実行する場合、反映方式は表単位反映方式となります。このため、Datareplicator は次のように動作します。
 - 反映環境定義の startmode パラメタは無効になります。
 - hdsstate コマンドの表示内容の<REFLECTION INFORMATION>中の reflection mode は"tbl"となります。
- UOC に対しては、control_trigger 指示をしません。このため、UOC で実行する場合には、反映環境定義の control_trigger パラメタは無効になります。

- UOC に対しては、disconnect 指示をしません。このため、UOC で実行する反映の場合には、反映システム定義の discintvl オペランドの指定は無効になります。
- UOC 内部で使う表の存在チェックをしません。このため、UOC で実行する反映の場合には、反映環境定義の tblcheck オペランドの指定は無効になります。
- UOC で反映処理を実行する場合には、反映定義でユーザが作成した UOC の名称を設定します。反映処理の実行時には、指定した名称の UOC が自動的に起動されます。

(c) SQL 記述領域への設定

SQL 記述領域を使って反映側 DB を更新する場合、必要な内容を UOC インタフェーステーブルから SQL 記述領域に設定して、UOC を作成します。

UOC で直接、UOC インタフェーステーブルの列データを参照する場合は、データ型によっては C 言語のキャストをし、参照する必要があります。UOC インタフェーステーブルの列データアドレスが指すデータのデータ形式、実データ長と C 言語での表記の関係を次の表に示します。

表 8-14 UOC の列データ型、データ長と C 言語での表記の関係

UOC インタフェーステーブル		データ領域長 (バイト)	C 言語での表記
ニモニック	列データ長		
HDS_T_INT	4	4	int
HDS_T_SINT	2	2	short
HDS_T_DEC	先頭 2 バイト：精度 後ろ 2 バイト：位取り	精度 ÷ 2 + 1	char データエリア [実データ長]
HDS_T_FLT	8	8	double
HDS_T_DBL			
HDS_T_SFLT	4	4	float
HDS_T_REAL			
HDS_T_CHAR	文字列長 (バイト数)	UOC の列データ長	char データエリア [実データ長]
HDS_T_MCHAR			
HDS_T_VCHAR	データ部文字列長 (バイト数)	UOC の列データ長 + 2	struct { unsigned short 実データ長 char データエリア [実データ長] }
HDS_T_NCHAR	文字列長 (文字数)	UOC の列データ長 * 2	char データエリア [実データ長]
HDS_T_NVCHAR	データ部文字列長 (文字数)	UOC の列データ長 * 2 + 2	struct { unsigned short 実データ長 char データエリア [実データ長] }

UOC インタフェーステーブル		データ領域長 (バイト)	C 言語での表記
ニモニック	列データ長		
			}
HDS_T_DATE	4	4	char データエリア [実データ長]
HDS_T_TIME	3	3	char データエリア [実データ長]
HDS_T_YTD	先頭 2 バイト：精度 (8) 後ろ 2 バイト：位取り (0)	5	char データエリア [実データ長]
HDS_T_HTS	先頭 2 バイト：精度 (6) 後ろ 2 バイト：位取り (0)	4	char データエリア [実データ長]
HDS_T_BLOB (列オプションが 0x80 以外の場合)	データ部データ長 (バイト数)	UOC の列データ長 + 8	struct { int 予備領域 unsigned int 実データ長 char データエリア [実データ長] } *
HDS_T_BLOB (列オプションが 0x80 の場合)	32	UOC の列データ長 + 8	[表 8-8 USUBSTR_BLK の内容] を参照し てください。
HDS_T_UNPACK	UNPACK データ長 (バイト 数)	整数部けた数+小数 部けた数	char データエリア [実データ長]
HDS_T_TIMESTAMP P	7 + ↑p ÷ 2 ↑ (バイト数) p : 小数部のけた数	UOC 列データ長	char データエリア [実データ長]
HDS_T_BINARY (列オプションが 0x80 以外の場合)	データ部データ長 (バイト数)	UOC の列データ長 + 4	struct { int 実データ長 char データエリア [実データ長] } *
HDS_T_BINARY (列オプションが 0x80 の場合)	32	UOC の列データ長 + 8	[表 8-8 USUBSTR_BLK の内容] を参照し てください。

注

繰返し列のデータは、一つの要素のデータ情報が一つの UCOLUMN_BLK に格納されます。

注※

連結演算のときも受け渡し方法は同じです。

(d) データ型

- 抽出側 DB がメインフレーム側 DB の場合、データ変換が必要なときには、データ変換をしたデータが UOC に渡されます。データ変換については、「4.3.4 データ型の対応の設計」を参照してください。

- DATE 型, TIME 型のデータ形式は, 符号なしパック形式で格納されています。SQL 記述領域を使った UAP を作成する場合には, データが格納されているエリアのアドレスを, 直接 SQL 記述領域に設定して処理できます。SQL 記述領域を使わない UAP を作成する場合, DATE 型, TIME 型のデータを文字列の日付データ, 時刻データに変換した後, 埋込み変数に設定して処理する必要があります。
- SMALLFLOAT 型のデータは, 表現範囲の相違によるオーバーフローを回避するため, FLOAT 型のデータとして渡されます。
- 抽出列型が UNPACK 型の場合, DECIMAL 型へデータ変換した後に DECIMAL データ形式で UOC に渡します。UNPACK データが DECIMAL 型に変換できない場合は, UNPACK データ形式で UOC に渡します。UOC に渡される UNPACK 型のニモニックについては, 「表 8-9 列データ型のニモニック」又は「表 8-10 データアドレスのニモニック」を参照してください。

8.1.5 反映情報編集 UOC での抽象データ型の扱い

反映情報編集 UOC に抽象データ型を渡す場合, 抽象データ型のコンストラクタ関数の入力データは, 抽象データ型の属性データとして渡されます。

抽象データ型の属性データは, 抽象データ型固有の情報を含んでいるため, 反映情報編集 UOC でも, 抽象データ型のコンストラクタ関数にそのまま反映してください (抽象データ型の更新データを編集及び加工して反映することはできません)。

(1) 抽象データ型のニモニックと列データ構造

抽象データ型のニモニックと列データ構造について説明します。

(a) 抽象データ型のニモニック

抽象データ型のニモニックを「表 8-15 列データ型のニモニック(UCOLUMN_BLK.coltype)」と「表 8-16 データアドレスのニモニック(UCOLUMN_BLK.dataptr)」に示します。抽象データ型以外のニモニックについては, 「表 8-9 列データ型のニモニック」と「表 8-10 データアドレスのニモニック」を参照してください。

表 8-15 列データ型のニモニック(UCOLUMN_BLK.coltype)

列データ型	ニモニック	データコード
ADT (抽象データ型)	HDS_T_ADT	(0x83)

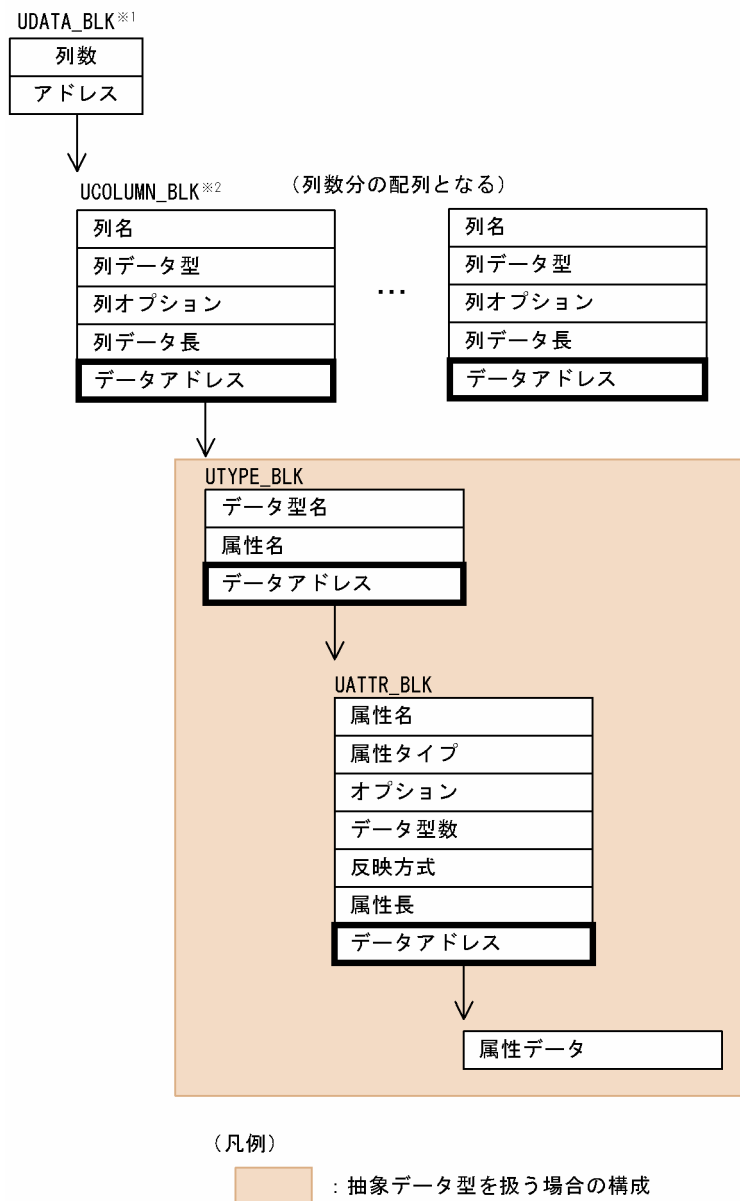
表 8-16 データアドレスのニモニック(UCOLUMN_BLK.dataptr)

データ型	ニモニック	アドレスタイプ
ADT (抽象データ型)	HDS_A_ADT	UTYPE_BLK *

(b) 抽象データ型の列データ構造

抽象データ型の列データ構造を次の図に示します。抽象データ型以外の列データ構造については、「[図 8-4 UDATA_BLK とその関連構造](#)」を参照してください。

図 8-5 抽象データ型の列データ構造



注※1

UDATA_BLK の内容については、「[表 8-6 UDATA_BLK の内容](#)」を参照してください。

注※2

UCOLUMN_BLK の内容については、「[表 8-7 UCOLUMN_BLK の内容](#)」を参照してください。

UTYPE_BLK (抽象データ型情報) の内容を次の表に示します。

表 8-17 UTYPE_BLK (抽象データ型情報) の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
schema_name	31	char	認可識別子	呼出元	抽象データ型の認可識別子 (終端記号を含む)
type_name	31	char	抽象データ型名	呼出元	抽象データ型の型名 (終端記号を含む)
reserve1	2	char	予備領域 1	呼出元	予備
def_attrnum	4	int	定義属性数	呼出元	抽象データ型の定義属性数
set_attrnum	4	int	属性データ数	呼出元	データ型中に格納されている属性データ数
data_addr	8	UATTR_BLK *	UATTR_BLK*アドレス	呼出元	属性管理テーブル (UATTR_BLK) のアドレス
reserve2	8	char	予備領域 2	呼出元	予備

UATTR_BLK (属性データ情報) の内容を次の表に示します。

表 8-18 UATTR_BLK (属性データ情報) の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
attr_name	31	char	属性名	呼出元	属性名 (終端記号を含む)
attr_type	1	unsigned char	属性のデータ型	呼出元	属性のデータ型 データ型のニモニックについては、「表 8-9 列データ型のニモニック」を参照してください。
null_flag	1	unsigned char	属性の NULL フラグ	呼出元	属性の NULL フラグ <ul style="list-style-type: none"> • 0x01 : NULL 値 • 0x00 : 非 NULL 値
adtfunc	1	unsigned char	抽象データ型の反映方式	呼出元	属性抽象データ型反映方式 <ul style="list-style-type: none"> • 0x01 : コンストラクタ関数反映方式 • 0x00 : 属性のデータ型が抽象データ型 (0x83) 以外の場合
reserve1	3	char	予備領域 1	呼出元	予備
attr_len	4	int	属性データ長	呼出元	属性データ長 次の場合は 0 になります。 <ul style="list-style-type: none"> • 属性のデータ型が抽象データ型の場合 • 属性の NULL フラグが NULL 値の場合
adtelnum	4	int	抽象データ型継承数	呼出元	属性のデータ型が抽象データ型の場合に、継承関係にあるデータ型数

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
					<ul style="list-style-type: none"> 1: 継承なしの場合 0: 属性のデータ型が抽象データ型 (0x83) 以外の場合
data_addr	8	UNION *	属性データのアドレス	呼出元	属性データのアドレス (抽出側で入力した抽象データ型のコンストラクタ関数の入力データ) <ul style="list-style-type: none"> UTYPE_BLK のアドレス: 属性のデータ型が抽象データ型の場合 NULL: 属性の NULL フラグが NULL 値の場合 データアドレスのニモニックについては、「表 8-10 データアドレスのニモニック」を参照してください。
reserve2	8	char	予備領域 2	呼出元	予備

(c) 各抽象データ型の UTYPE_BLK と UATTR_BLK の設定内容

各抽象データ型の UTYPE_BLK の設定内容を「表 8-19 各抽象データ型の UTYPE_BLK の設定内容」に、UATTR_BLK の設定内容を「表 8-20 各抽象データ型の UATTR_BLK の設定内容」に示します。

表 8-19 各抽象データ型の UTYPE_BLK の設定内容

メンバ名	抽象データ型		
	SGMLTEXT	FREWORD	XML
schema_name	MASTER	MASTER	MASTER
type_name	SGMLTEXT	FREWORD	XML
reserve1	予備	予備	予備
def_attrnum	0	0	0
set_attrnum	1	1	1
data_addr	UATTR_BLK のアドレス	UATTR_BLK のアドレス	UATTR_BLK のアドレス
reserve2	予備	予備	予備

表 8-20 各抽象データ型の UATTR_BLK の設定内容

メンバ名	抽象データ型		
	SGMLTEXT	FREWORD	XML
attr_name	未設定	未設定	未設定
attr_type	HDS_T_BLOB(0x93)*	HDS_T_VCHAR(0xC1)*	HDS_T_BINARY(0x91)*
null_flag	0x00	0x00	0x00
adtfunc	0x00	0x00	0x00

メンバ名	抽象データ型		
	SGMLTEXT	FREEWORD	XML
reserve1	予備	予備	予備
attr_len	実際の属性データ長 (最大 2,147,483,647)	実際の属性データ長 (最大 32,000)	実際の属性データ長 (最大 2,147,483,647)
adtelnum	0	0	0
data_addr	属性データのアドレス	属性データのアドレス	属性データのアドレス
reserve2	予備	予備	予備

注※

反映情報編集 UOC に渡した抽象データ型の属性データを参照する場合は、通常の列データ型と同じように C 言語のキャストをしてください。属性データのアドレスが指すデータ構造を属性データ型にキャストするときは、「表 8-14 UOC の列データ型、データ長と C 言語での表記の関係」を参照してください。

(2) 抽象データ型を扱うときの注意事項

抽象データ型の場合、反映情報編集 UOC に渡す UCOLUMN_BLK の collen メンバ (列データ長) には値を設定しません。そのため、抽象データ型を扱うときは、UCOLUMN_BLK の collen メンバの値を参照しないでください。

(3) 抽象データ型を扱う反映情報編集 UOC のサンプル

抽出対象表及び反映対象表の構成

抽出対象表の構成を次の図に示します。反映対象表も同じ構成になります。

列のデータ型	INTEGER	CHAR(64)	SGMLTEXT	FREEWORD	XML
列名	KEYNO	KEYNAME	COL_SGMLTEXT	COL_FREEWORD	COL_XML
	:	:	:	:	:

コーディング例

SGMLTEXT 型、FREEWORD 型、及び XML 型の列を含む表に対して INSERT を実行するコーディング例を次の図に示します。

図 8-6 抽象データ型の列を含む表に対して INSERT を実行するコーディング例

```

/*****
/* 機能 = 抽象データ型を使った反映情報編集UOCのサンプル */
*****/
#include <hds_ucommon.h>
#include <hds_ureflect.h>
#include <stdio.h>

#define SQL_INSERT      "INSERT "
                        :
                        :

/*****
/* 機能 = 編集・加工指示 */
*****/
int hds_uedit(
    UINTERFACE_BLK *pstUiBlock,
    UREFLECT_BLK *pstRefBlock,
    UDATA_BLK *pstKeyBlock,
    UDATA_BLK *pstRefDataBlock,
    long *piStatus)
    /* IPF版の場合は、longではなく int */
{
    EXEC SQL BEGIN DECLARE SECTION;
        int iKeyNo; /* キー番号(INT) */
        char szKeyName[64]; /* キー名称(CHAR(64)) */
        SQL TYPE IS BLOB(100000) stSgmltext; /* SGMLTEXT列 ※1 */
        SQL TYPE IS BINARY(100000) stXml; /* XML列 ※1 */
        SQL TYPE IS VARCHAR(32000) stFreeWord; /* FREEWORD列 ※1 */
        /* ※1:サイズは、必要サイズ分確保してください */
    EXEC SQL END DECLARE SECTION;

    int iRc; /* 当関数リターンコード */
    UCOLUMN_BLK *pstColBlk; /* 列情報 */
    UTYPE_BLK *pstTypeBlk; /* 抽象データ型情報 */
    UATTR_BLK *pstAttrBlk; /* 抽象データ型属性情報 */

    if ( memcmp( pstRefBlock->updkind, SQL_INSERT, strlen(SQL_INSERT)) == 0 ) {
        /* KEYNO */
        pstColBlk = &(pstRefDataBlock->colinfptr[0]);
        iKeyNo = *(pstColBlk->dataptr.intptr);

        /* KEYNAME */
        pstColBlk = &(pstRefDataBlock->colinfptr[1]);
        memcpy(szKeyName, pstColBlk->dataptr.cptr, 64);

        /* COL_SGMLTEXT */
        /* 反映情報編集UOCに渡ってきた更新データをそのまま反映する */
        pstColBlk = &(pstRefDataBlock->colinfptr[2]);
        pstTypeBlk = (UTYPE_BLK *)pstColBlk->dataptr.adtptr;
        pstAttrBlk = (UATTR_BLK *) (pstTypeBlk->data_addr);
        stSgmltext.stSgmltext_reserved = 0;
        stSgmltext.stSgmltext_length =
            *(unsigned long *) (pstAttrBlk->data_addr.blobptr + sizeof(long));
        memcpy(stSgmltext.stSgmltext_data,
            (pstAttrBlk->data_addr.blobptr + sizeof(long) + sizeof(unsigned long)),
            stSgmltext.stSgmltext_length);
    }
}

```

```

/*----- Comment Begin (for IPF) -----
IPF版の場合は、longではなくint
stSgmltext.stSgmltext_length =
    *(unsigned int*)(pstAttrBlk->data_addr.blobptr + sizeof(int));
memcpy(stSgmltext.stSgmltext_data,
    (pstAttrBlk->data_addr.blobptr + sizeof(int) + sizeof(unsigned int) ),
    stSgmltext.stSgmltext_length);
----- Comment End -----*/

/* COL_XML */
/* 反映情報編集UOCに渡ってきた更新データをそのまま反映する */
pstColBlk = &(pstRefDataBlock->colinfptr[3]);
pstTypeBlk = (UTYPE_BLK *)pstColBlk->dataptr.adtpr;
pstAttrBlk = (UATTR_BLK *) (pstTypeBlk->data_addr);
stXml.len = *(long *) (pstAttrBlk->data_addr.binaryptr);
memcpy(stXml.str, (pstAttrBlk->data_addr.binaryptr + sizeof(long)), stXml.len);
/*----- Comment Begin (for IPF) -----
IPF版の場合は、longではなくint
stXml.len = *(int *) (pstAttrBlk->data_addr.binaryptr);
memcpy(stXml.str, (pstAttrBlk->data_addr.binaryptr + sizeof(int)), stXml.len);
----- Comment End -----*/

/* COL_FREEWORD */
/* 反映情報編集UOCに渡ってきた更新データをそのまま反映する */
pstColBlk = &(pstRefDataBlock->colinfptr[4]);
pstTypeBlk = (UTYPE_BLK *)pstColBlk->dataptr.adtpr;
pstAttrBlk = (UATTR_BLK *) (pstTypeBlk->data_addr);
stFreeWord.len = *(short *) ((unsigned short *) (pstAttrBlk->data_addr.vcptr));
memcpy(stFreeWord.str,
    (pstAttrBlk->data_addr.vcptr + sizeof(stFreeWord.len)),
    stFreeWord.len);

EXEC SQL
    INSERT INTO ADTTBL( KEYNO, KEYNAME,
        COL_SGMLTEXT,
        COL_XML,
        COL_FREEWORD)
    VALUES( :iKeyNo, :szKeyName,
        SGMLTEXT( :stSgmltext AS BLOB(2147483647) ),
        XML( :stXml AS BINARY(2147483647) ),
        FREEWORD( :stFreeWord AS VARCHAR(32000) ) );

        :
        :
        :

```

(凡例)

■ : 抽象データ型の更新データを操作している箇所です。

8.1.6 反映情報編集 UOC を作成するときの注意

(1) シグナルの制限

反映情報編集 UOC の関数の中では、シグナルの操作をしないでください。

(2) システムコールの制限

- alarm(), pause() のシステムコールを発行しないでください。
- fork(), exec() で子プロセスを生成する場合は、ユーザ側で後処理してください。

(3) 共用メモリの制限

反映側 Datareplicator 又はほかのシステムと同じ共用メモリを使わないでください。

(4) 同期点の制限

hds_uedit()で「COMMIT」を渡し、その戻り値が0又は4の場合、1個の同期点が完結されたものとみなされます。障害が発生して反映情報編集 UOC のプロセスがアボートした場合、再開始のポイントは、前回の同期点の次から再実行されます。反映情報編集 UOC の関数の中では、「COMMIT」が渡された場合に同期点処理をするとともに、途中で異常終了して再開始された場合に二重処理が発生しないように考慮した作りをしておく必要があります。

(5) データベースにアクセスする UOC を作成するときの注意

データベースにアクセスする UOC では、編集終了関数 hds_uend()でインタフェースブロック (UINTERFACE_BLK) の終了情報フラグ (stopinf) を参照してください。その終了情報が強制終了の場合は、データベースシステムに対して ROLLBACK を実行した後に DISCONNECT するように UOC を作成してください。この場合、ROLLBACK しないで DISCONNECT すると、強制停止までの更新が DISCONNECT の自動コミットによってコミットされます。そのため、次回の Datareplicator 再開始時に、データベースに矛盾が起こって UOC が SQL エラーになります。

(6) 反映環境定義との関連

- discintvl (disconnect 発行間隔) は、UOC による反映グループでは無効です。
- 反映対象表にトリガを定義して反映情報編集インタフェースの関数内で HiRDB にアクセスする場合、反映環境定義の control_trigger パラメータは無効になります。

(7) 反映処理に関する監査証跡を取得しない場合の注意

反映情報編集 UOC を新規に作成する場合は、ここで示す注意事項に従って、反映情報編集 UOC を作成してください。

既存の反映情報編集 UOC を使用する場合は、反映情報編集 UOC の実行形式ファイルを再度リンケージしてください。その場合、既存の反映情報編集 UOC がここで示す注意事項に反していないことを確認してください。リンケージ方法については「8.1 反映情報編集 UOC」を参照してください。なお、リンケージを行わなかった場合は、監査証跡が取得されます。

- 次の規則に従って反映情報編集 UOC を作成してください。
 - hds_ubegin 関数内で HiRDB への CONNECT を実行する
この規則に反した場合、KFRB03094-W メッセージが出力されます。
その後の処理は、反映システム定義の hirdb_audit_trail オペランドの指定値に従います。
 - hds_ubegin 関数内で CONNECT 以外の SQL を実行しない

この規則に反した場合、hds_ubegin 関数内で指定した CONNECT 以外の SQL は、監査証跡が取得されます。なお、hds_ubegin 関数内でトランザクションが終了していない場合は、そのトランザクションに関する監査証跡が取得される場合があります。

- hds_uend 関数内で HiRDB への DISCONNECT を実行する
この規則に反した場合、DISCONNECT 発行後の監査証跡は取得されます。
- SQL を実行しない反映情報編集 UOC の場合、実行形式ファイル作成時のリンケージコマンドに HiRDB のライブラリを指定しないでください。
この規則に反した場合、KFRB03094-W メッセージが出力されます。
その後の処理は、反映システム定義の hirdb_audit_trail オペランドの指定値に従います。
- 反映情報編集 UOC によって不正なアクセスが行われることを防ぐため、Datareplicator 管理者は、次の点を必ず確認してください。
 - 使用する反映情報編集 UOC が DB に対して不正なアクセスをしないこと
 - 使用する反映情報編集 UOC が悪意を持った第三者に改竄されないこと
 - 反映情報編集 UOC ファイルの所有者及びアクセス権が次のように設定されていること

表 8-21 反映情報編集 UOC ファイルの所有者及びアクセス権

所有者, アクセス権		設定する情報
所有者	ユーザ ID	HiRDB Datareplicator 管理者
	グループ ID	HiRDB 管理者のグループ ID
アクセス権	所有者	r-x (read 権限及び実行権限)
	グループ	--- (アクセス不可)
	そのほか	--- (アクセス不可)

ファイルの書き込み権限がないため、反映情報編集 UOC の実行形式ファイルを再作成する場合は、既存の実行形式ファイルを削除してから、新しい実行形式ファイルを再作成してください。

(8) 参照制約が定義されている表をデータ連動するときの注意

反映情報編集 UOC に、参照制約が定義されている表は指定できません。

(9) 拡張 SYSLOG 機能に関する注意事項

拡張 SYSLOG 機能を使用する場合、「8.1.2 反映情報編集 UOC の作成手順 (UNIX)」に記載されているリンケージ方法で再度リンケージをしてください。

リンケージしなかった場合、反映情報編集 UOC で出力される syslogfile は、拡張 SYSLOG 機能の対象となりません。

8.1.7 反映情報編集 UOC のサンプル

反映側 Datareplicator に格納してある UOC のサンプルについて説明します。

(1) 格納ディレクトリとファイル名

反映側 Datareplicator をインストールすると、サンプルファイルが次のディレクトリに格納されます。

ディレクトリ

UNIX 版の場合：/opt/hirdbds/lib/sample/

Windows 版の場合：Datareplicator をインストールしたディレクトリ¥sample¥

ファイル名

hds_redit.ec

(2) 抽出環境

(a) 抽出対象表

抽出対象表の構成を次の図に示します。

図 8-7 抽出対象表の構成

列のデータ型	INTEGER	CHAR (30)	CHAR (2)	INTEGER	INTEGER
列名	商品コード	商品名	色	単価	在庫量
	:	:	:	:	:

注 更新処理の対象は、「在庫量」だけと想定しています。

(b) 抽出処理の内容

サンプルで抽出する処理の内容を次に示します。

- 更新情報名：ZAIKO
- 抽出対象列：すべての列
- 抽出順序：列の構成順
- マッピングキー：「商品コード」の列

(3) 反映環境

(a) 反映対象表の内容

反映対象表の構成を次の図に示します。反映対象表名は、「在庫表」としています。

図 8-8 反映対象表（在庫表）の構成

列のデータ型	INTEGER	CHAR (30)	INTEGER	INTEGER
列名	商品コード	商品名	単価	在庫量
	:	:	:	:

(b) 反映定義

反映定義の内容を次に示します。反映定義は次の条件に従っています。

- 抽出対象表の列「色」以外を反映対象とします。
- UOC 名は「/users/replicator/sampleuoc」とします。

```
format ZAIKO
NAME FCODE      /* 商品コード */
NAME FNAME      /* 商品名 */
NAME FCOLOR     /* 色 */
NAME FTANKA     /* 単価 */
NAME FZAIKO     /* 在庫量 */
load FCODE,FNAME,FTANKA,FZAIKO from ZAIKO by '/users/replicator/sampleuoc';
```

(4) 処理の内容

編集開始指示関数 (hds_ubegin) は、反映側 HiRDB への CONNECT 処理を実行します。編集・加工指示関数 (hds_uedit) は、更新種別に対応して INSERT, UPDATE, DELETE, PURGE, COMMIT を実行します。

UPDATE, DELETE では、反映対象表に更新対象の行がないエラー (SQLCODE 100) になると、エラーになった更新情報を無視して反映処理を継続します。そのほかの SQL 実行エラーが発生した場合、編集終了指示関数 (hds_uend) で ROLLBACK 処理を実行します。編集終了指示関数 (hds_uend) は、必要に応じて ROLLBACK し、DISCONNECT を実行します。

(a) 編集開始指示関数 (hds_ubegin)

hds_ubegin 関数のパラメタである UINTERFACE_BLK から、認可識別子、パスワードを埋込み変数に設定して、その埋込み変数をパラメタとして反映側 HiRDB に CONNECT を発行します。

(b) 編集・加工指示関数 (hds_uedit)

- SQL 実行処理

hds_uedit 関数のパラメタである UREFLECT_BLK の更新種別で指定された SQL を、反映側 HiRDB に発行します。

INSERT

hds_uedit 関数のパラメタである反映データブロック (UDATA_BLK) を基に反映対象表の各列の更新データアドレスを導き、各データを埋込み変数に設定します。設定した埋込み変数をパラメタとして INSERT を実行します。

UPDATE

マッピングキーの列である「商品コード」の更新データアドレスを、hds_uedit 関数のパラメタであるキー情報ブロックを基に導き、その更新データを埋込み変数に設定します。

抽出対象表の更新対象列は、列「在庫量」だけのため、hds_uedit 関数のパラメタである反映データブロックには、列「在庫量」の更新情報が設定されています。その列「在庫量」の更新データを埋込み変数に設定します。

列「商品コード」の埋込み変数を UPDATE の探索条件値とし、さらに列「在庫量」の埋込み変数を更新値として UPDATE を実行します。

DELETE

マッピングキーの列「商品コード」の更新データアドレスを hds_uedit 関数のパラメタであるキー情報ブロックを基に導き、更新データを埋込み変数に設定します。その埋込み変数を DELETE 文の探索条件として DELETE を実行します。

PURGE

反映対象表の名称「在庫表」を指定し、PURGE TABLE を実行します。

COMMIT

COMMIT WORK を実行します。

• 関数終了処理

SQL の実行結果である SQLCODE によって、リターンコードとステータスコードを決定します。SQLCODE によるリターンコードとステータスコードを次の表に示します。

表 8-22 SQL コードによるリターンコードとステータスコード

SQLCODE	SQL の状態	リターンコード	ステータスコード
0	正常終了	HDS_RET_NORM (正常終了)	—
100	UPDATE, 又は DELETE で反映対象表に、該当する行が存在しないエラー	HDS_RET_WARN (軽度のエラー)	SQLCODE
0, 100 以外※	反映処理を中断するエラー	HDS_RET_ERR (重度のエラー)	SQLCODE

注※

hds_uend 関数で ROLLBACK を実行するフラグとして、hds_uedit 関数のパラメタであるインタフェースブロック (UINTERFACE_BLK) の引き継ぎ情報 2 に「-1」を設定します。

(c) 編集終了指示関数 (hds_uend)

ROLLBACK 及び DISCONNECT を実行します。

ROLLBACK は、hds_uend 関数のパラメタであるインタフェースブロック (UNITERFACE_BLK) の引き継ぎ情報 2 が「-1」のときだけ実行します。

8.2 列データ編集 UOC

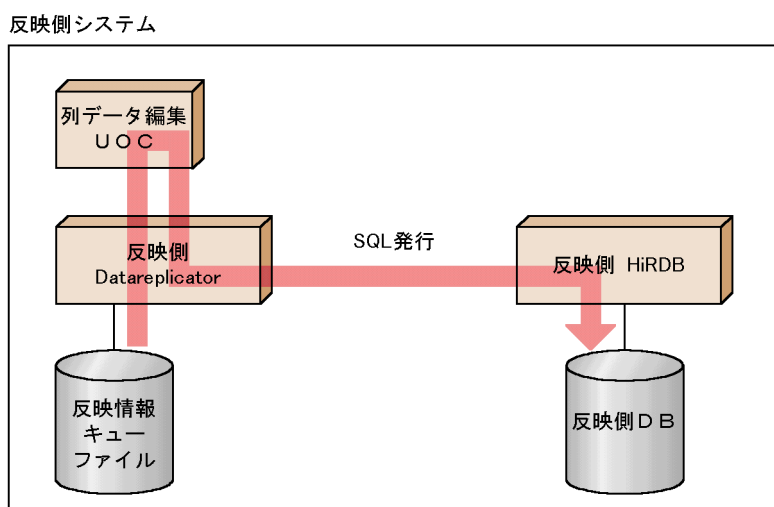
列データ編集 UOC を作成すると、任意の抽出列データを編集して、反映側 DB に反映できます。

8.2.1 列データ編集 UOC の概要

列データ編集 UOC を作成すると、任意の抽出データ情報を Datareplicator が反映側 DB に反映する前に UOC へ制御が渡されます。Datareplicator は UOC から制御が戻った後に、UOC で編集したデータを反映側 DB へ反映します。

列データ編集 UOC を作成した場合の更新情報を反映する手順を次の図に示します。

図 8-9 列データ編集 UOC を使った場合の更新情報を反映する手順

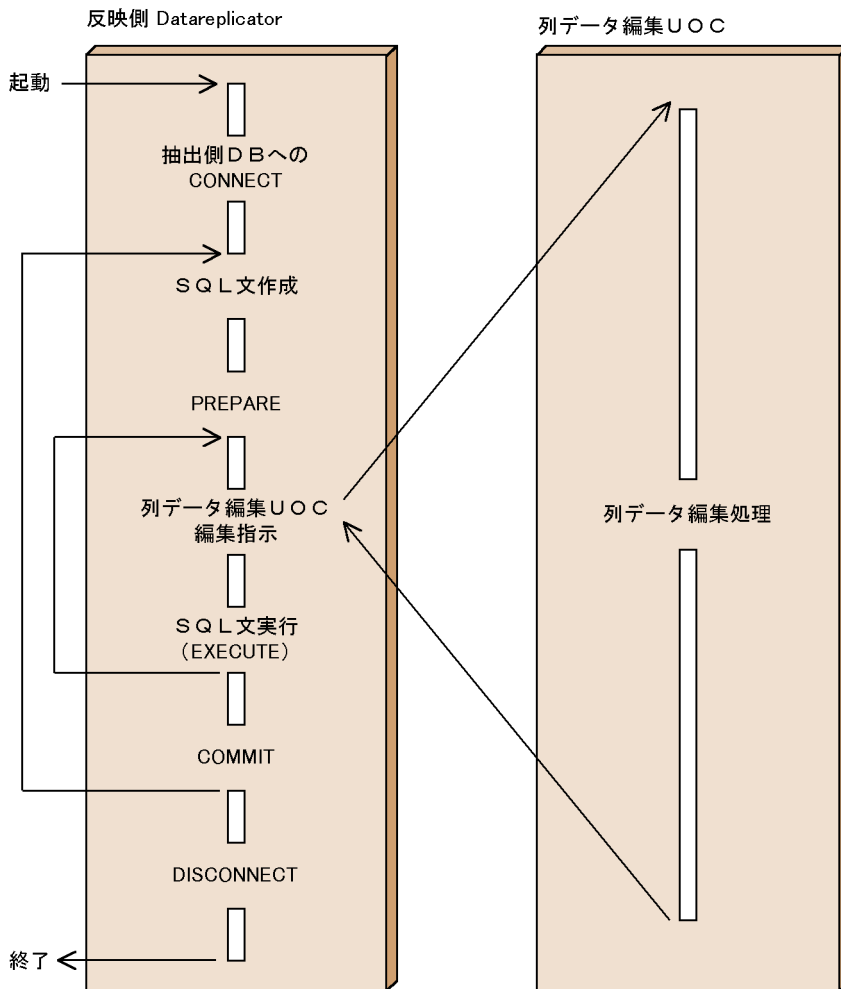


(1) 列データ編集 UOC が呼び出されるタイミング

列データ編集 UOC は、列データ編集 UOC の対象抽出列を含む表の INSERT と列データ編集 UOC の対象反映列の UPDATE 反映処理が発生したときに、Datareplicator から呼び出されます。

列データ編集 UOC の制御の流れを次の図に示します。

図 8-10 列データ編集 UOC の制御の流れ



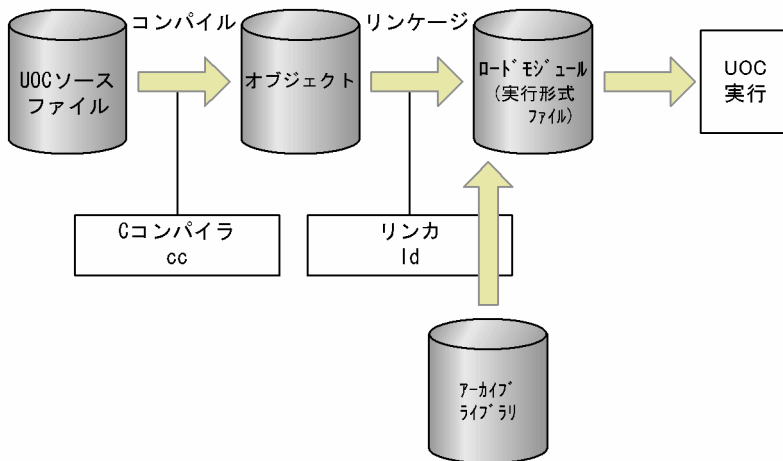
(凡例)

□ : 処理中を表します。

8.2.2 列データ編集 UOC の作成手順 (UNIX)

UNIX 版 Datareplicator で列データ編集 UOC を作成する手順について説明します。列データ編集 UOC を実行するまでの手順を次の図に示します。

図 8-11 列データ編集 UOC の実行手順



(1) 作成方法

列データ編集 UOC インタフェース共有ライブラリの作成方法を次に示します。

1. 列データ編集 UOC インタフェースを使う反映列を決定します。
2. 列データ編集 UOC インタフェースの関数を作成します。
3. 列データ編集 UOC インタフェースの関数をコンパイル／リンケージします。

(2) コンパイル及びリンケージ

コンパイル及びリンケージについて OS ごとに説明します。

なお、リンケージの未解決エラーが発生する場合は、リンケージのときに必要なライブラリすべてを指定してください。

(a) HP-UX 版の場合

- UOC のコンパイル方法

```
cc -c +DD64 +z -I/opt/hirdbds/include UOCソースファイル名 [UOCソースファイル名] ...
```

-c :

オブジェクト作成オプション

+z :

位置独立コード作成オプション

-I :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。
UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

- UOC のリンケージ方法

```
ld -b -o libhdscuoc.sl UOCオブジェクトファイル名 [UOCオブジェクトファイル名] ...
```

(b) AIX 版の場合

- UOC のコンパイル方法

```
xlc -q64 -c -I /opt/hirdbds/include UOCソースファイル名 [UOCソースファイル名] ...
```

-c :

オブジェクト作成オプション

-I :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

- UOC のリンケージ方法

```
xlc -q64 -bM:SRE -bnoentry -bE:/opt/hirdbds/lib/libhdscuoc.exp -o libhdscuoc.a  
UOCオブジェクトファイル名 [UOCオブジェクトファイル名] ...
```

(c) Linux 版の場合

- UOC のコンパイル方法

```
gcc -c -fPIC -I/opt/hirdbds/include UOCソースファイル名 [UOCソースファイル名] ...
```

-c :

オブジェクト作成オプション

-fPIC :

位置独立コード作成オプション

-I :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

- UOC のリンケージ方法

```
gcc -shared -o libhdscuoc.so UOCオブジェクトファイル名 [UOCオブジェクトファイル名] ...
```

(3) 注意事項

作成した列データ編集 UOC のライブラリは、任意のディレクトリに格納できます。

格納先のディレクトリは SHLIB_PATH 環境変数で指定してください。

ただし、/opt/hirdbds/lib 下にある libhdscuoc.sl (Linux 版, 又は HP-UX (IPF) 版の場合は libhdscuoc.so, AIX 版の場合は libhdscuoc.a) とは入れ替えないでください。

8.2.3 列データ編集 UOC の作成手順 (Windows)

Windows 版 Datareplicator で列データ編集 UOC を作成する手順について説明します。Windows 版 Datareplicator の UOC は、DLL ファイル (xxx.dll) として作成します。

(1) 作成方法

Windows 版の場合、列データ編集 UOC は Visual Studio で作成します。

次に示す手順で、DLL ファイルを作成してください。

1. Windows 版 HiRDB のアプリケーションプログラムを作成する C 言語で列データ編集 UOC の関数を作成します。列データ編集 UOC の DLL ファイル名は、**hdscuoc.dll** にしてください。
Datareplicator の UOC 用のヘッダファイルは、Datareplicator のインストール先ディレクトリの下に `¥include` に格納してあります。
2. 作成した関数をコンパイル/リンケージして、DLL ファイルを作成します。Datareplicator から UOC を呼び出すため、インタフェース関数は必ず `__cdecl` 呼び出し規約で作成して、エクスポートしておいてください。

(2) コンパイル及びリンケージ

Visual Studio を使用してコンパイル、リンケージをする場合、オプションの設定は、プロジェクトメニューから「プロパティ」を選択します。

設定する項目を次の表に示します。

表 8-23 Visual Studio の「プロパティ」で設定する項目

項目	カテゴリ	カテゴリの設定	設定値
プラットフォーム	—	—	x64
構成プロパティ	詳細	共通言語ランタイムサポート	共通言語ランタイムサポートを使用しない
C/C++	コード生成	構造体メンバのアラインメント	8 バイト (/Zp8)
		ランタイムライブラリ	マルチスレッド DLL (/MD)
	詳細設定	呼び出し規約	<code>__cdecl</code> (/Gd)
	全般	追加のインクルードディレクトリ	Datareplicator のインストールディレクトリ¥include

(凡例)

— : 該当しません。

リンケージの未解決エラーが発生する場合は、リンケージのときに必要なライブラリすべてを指定してください。

(3) 列データ編集 UOC を作成するときの注意

- 反映側 Datareplicator 又はほかのシステムで使っているメモリマップトファイルは、UOC では使わないでください。
- UOC の処理がエラーになったときは、エラーログファイルのプログラム名には DLL ファイルの名称が出力されます。
- UOC を配置したフォルダをシステム環境変数の path に追加してください。

8.2.4 列データ編集 UOC で使う関数の文法

列データ編集 UOC は、C 言語でプログラミングします。反映側 Datareplicator には、列データ編集 UOC を作成するときを使う関数を備えています。

列データ編集 UOC で使える関数名と機能の概要を次の表に示します。

表 8-24 列データ編集 UOC で使える関数と機能の概要

関数名	機能
hds_ucoleditX() (列データの編集)	列データの編集前のデータを受け取って、編集後のデータを格納します。ユーザは、編集前のデータを編集して、編集後のデータを格納します。

(1) hds_ucoleditX (列データの編集関数)

列データ編集 UOC で使う関数は、hds_ucoleditX (X は 1~8) です。一つの Datareplicator システム内に 8 個の UOC 関数を作成できます。

(a) 関数の形式

```
#include<hds_ucommon.h>
#include<hds_ureflect.h>
int hds_ucoleditX (UCOLENV_BLK *列データ編集インタフェース環境ブロック,
                  UCOLUMN_BLK *編集前反映列情報,
                  UCOLUMN_BLK *編集後反映列情報,
                  int *ステータス);
```

(b) パラメタの説明

パラメタの内容を次に示します。

関数の構造体は、列データ編集 UOC のヘッダファイルで定義しています。構造体の設定者が呼び出し元の領域は、更新しないでください。ヘッダファイルについては、「表 8-31 列データ編集 UOC で使うヘッダファイルの一覧」を参照してください。

- UCOLENV_BLK

列データ編集インタフェースの環境情報を格納するブロックです。
UCOLENV_BLK の内容を次の表に示します。

表 8-25 UCOLENV_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
interface_blk	56	UINTERFACE_BLK	インタフェースブロック	呼び出し元	反映システム定義の情報
reflect_blk	96	UREFLECT_BLK	反映制御ブロック	呼び出し元	反映データの抽出側 DB 更新情報
auth_id	8	char	認可識別子	呼び出し元	反映側 DB 反映時の認可識別子
tbl_id	30	char	表識別子	呼び出し元	反映側 DB 反映時の表識別子
reserve	2	char	予備	呼び出し元	—

• UINTERFACE_BLK, UREFLECT_BLK

UINTERFACE_BLK と UREFLECT_BLK の構造は、反映情報編集 UOC と構造は同じですが、設定内容や使う領域が異なります。

UINTERFACE_BLK の内容を「表 8-26 UINTERFACE_BLK の内容」に、UREFLECT_BLK の内容を「表 8-27 UREFLECT_BLK の内容」に示します。

表 8-26 UINTERFACE_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
auth_id	8	char	認可識別子	呼び出し元	反映システム定義で指定した接続認可識別子
password	30	char	パスワード	呼び出し元	反映システム定義で指定したパスワード
exsysid	2	char	抽出側 Datareplicator 識別子	呼び出し元	抽出側 Datareplicator の抽出システム定義で指定した抽出側 Datareplicator 識別子。小文字の 16 進数字が文字列として設定されます。 XDM/DS との連動時は、'00'が設定されます。
repid1	2	char	識別子 1	呼び出し元	反映システム定義で指定したデータ連動識別子
repid2	2	char	識別子 2	呼び出し元	反映システム定義で指定した反映側 Datareplicator 識別子
*inherinf1	4	char *	—	—	列データ編集 UOC では使いません。
inherinf2	4	int	—	—	列データ編集 UOC では使いません。
stopinf	4	int	—	—	列データ編集 UOC では使いません。

表 8-27 UREFLECT_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
updkind	8	char	更新種別	呼び出し元	抽出側 DB を更新するときの種別。次に種別を示します。 INSERT INSERT 文を示します。 UPDATE UPDATE 文を示します。 注 種別は 8 バイト固定です。8 バイトに満たない部分は、空白で埋められます。
auth_id	30	char	認可識別子	呼び出し元	抽出側 DB を更新するときの認可識別子
tbl_id	30	char	表識別子	呼び出し元	抽出側 DB を更新するときの表識別子
upd_date	4	char	更新日付	呼び出し元	抽出側 DB を更新するときの日付 (更新情報をジャーナルに出力した日付)。日付は 10 進符号なしパック形式 (YYYYMMDD) で出力します。 YYYY：西暦年 MM：月 DD：日
upd_time	4	char	更新時刻	呼び出し元	抽出側 DB 更新時の時刻 (更新情報をジャーナルに出力した時刻) 時刻は 10 進符号なしパック形式 (HHMMSSTT) で出力します。 HH：時 MM：分 SS：秒 TT：1/100 秒
uapname	8	char	抽出側の UAP 名	呼び出し元	抽出側 DB を更新するときのユーザアプリケーションプログラム名
extsysid	4	char	抽出側 DB のシステム ID	呼び出し元	抽出側 DB が HiRDB の場合 抽出側 DB の HiRDB 識別子 抽出側 DB がメインフレーム (XDM/DS, PDMII E2, RDB1 E2) の場合 抽出側 DB のサブシステム ID
infflag1	1	unsigned char	フラグ 1	呼び出し元	未使用 HDS_NULL_FLAG (0x00) 固定値
infflag2	1	unsigned char	フラグ 2	呼び出し元	未使用 HDS_NULL_FLAG (0x00) 固定値

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
infflag3	1	unsigned char	フラグ 3	呼び出し元	未使用 (0x00) 固定値
reserve1	1	char	—	—	(境界調整用)
infflag4	4	int	フラグ 4	呼び出し元	未使用 (0x00) 固定値

• UCOLUMN_BLK

UCOLUMN_BLK は、列データについての情報を格納する領域です。

編集前反映列情報は、呼び出し元が更新情報の更新データについての情報を格納し、列データ編集 UOC へ渡します。一方、編集後反映列情報は、UOC で編集した情報を格納する領域です。編集後のデータ型、列オプション (任意)、列データ長の設定、編集後のデータは、編集後反映列情報のデータアドレスが示す領域へ UOC が格納してください。

UCOLUMN_BLK の内容を次の表に示します。

表 8-28 UCOLUMN_BLK の内容

メンバ名	長さ (バイト)	属性	領域名	編集前 設定者	編集後 設定者	内容
colname	30	char	列名	呼び出し元	呼び出し元	編集前反映列情報 抽出元表の列名が設定されます。 編集後反映列情報 反映先表の列名が設定されます。
coltype	1	unsigned char	列タイプ	呼び出し元	呼び出し先	編集前反映列情報として、抽出列の列データ型が設定されます。 編集後反映列情報として、編集後の列データ型を設定してください。 反映情報編集インタフェースと同じモニックを使ってください (ただし、編集後反映列情報には、BLOB, BINARY, UNPACK, 及び ADT (SGMLTEXT, FREEWORD, XML) データ型は指定できません)。
coloption	1	unsigned char	列オプション	呼び出し元	呼び出し先	編集前反映列情報として、抽出列の列オプションが設定されます。 編集後反映列情報として、編集後の列オプションを設定してください。 設定される列オプションを次に示します。 HDS_COL_NOOPT (0x00) オプションなし

メンバ名	長さ (バイト)	属性	領域名	編集前 設定者	編集後 設定者	内容
						HDS_COL_NULL (0x01) NULL データ HDS_COL_ARRAY (0x04) 配列列 (ただし、編集後反映列情報には、配列列オプションは設定できません)
elementnum	4	union	配列列の有効要素番号	呼び出し元	—	編集前反映列情報として、列オプションに配列が指定されている場合、配列列の有効要素番号を設定します (編集後反映列情報には設定できません)。
mltcolkind	1	unsigned char	—	—	—	列データ編集 UOC では使いません。
adtfunc	1	unsigned char	ADT 列反映方式	呼び出し元	—	抽象データ型連動方式として、次に示す値が設定されます。 0x00 抽象データ型以外 HDS_ADTREP_NORM(0x01) コンストラクタ関数反映方式
charset	1	char	文字集合種別	呼び出し元	—	UOC に渡す列データの文字集合種別が設定されます。 編集前と編集後の文字集合種別は変更しないでください。変更すると、編集後の文字集合種別を無視し、編集前の文字集合として以降の処理を続行します。 文字集合指定なし： X'00' (HDS_CSET_DEFAULT) 文字集合指定あり (EBCDIK)： X'01' (HDS_CSET_EBCDIK)
reserve1	1	char	—	—	—	(境界調整用)
collen	4	int	列データ長	呼び出し元	呼び出し先	列のデータ長。内容については、「 表 8-29 列データ長の設定規則 」を参照してください。 ただし、列オプションが NULL データのときは、編集前反映列情報には 0 を設定します。編集後反映列情報に列オプションとして NULL データを設定する場合、データ型に関係なく列データ長を設定する必要はありません。
dataptr	4	union	データアドレス	呼び出し元	—	編集前反映列情報 編集前列データのアドレスが設定されています。

メンバ名	長さ (バイト)	属性	領域名	編集前設定者	編集後設定者	内容
						<p>列オプションが NULL データのときには、データアドレスとして、NULL が設定されています。</p> <p>編集後反映列情報</p> <p>列データ編集領域のアドレスが設定されています。列データ編集領域は、32002 バイトの領域です。列データ編集 UOC では、本アドレスが示す領域へ対して編集データを設定してください。</p> <p>アドレスの型は、反映情報編集インタフェースと同じモニックを使ってください。</p>

表 8-29 列データ長の設定規則

変換前/変換後 列データ型	編集前反映列情報		編集後反映列情報		備考
	設定有無	設定内容	設定要否	設定内容	
NCHAR	有	編集前データ長 [文字数]	要	編集後データ長 [文字数]	—
CHAR MCHAR UNPACK	有	編集前データ長 [バイト数]	要	編集後データ長 [バイト数]	編集後反映列データ型に UNPACK 型は指定できません。
NVARCHAR VARCHAR MVARCHAR	無	—	不要	—	編集前データは、データエリア内の先頭 2 バイトに格納されている実長を参照して編集してください。また、編集後データにはデータエリアの先頭 2 バイトに、編集後データの実長を設定してください。列データ型が NVARCHAR のときは、文字数を設定、VARCHAR、MVARCHAR のときは、バイト数を設定してください。
DECIMAL	有	先頭 2 バイト 精度 (全体のけた数) 後ろ 2 バイト 位取り (小数部けた数)	要	先頭 2 バイト 精度 (全体のけた数) 後ろ 2 バイト 位取り (小数部けた数)	次の場合、精度の最大値は 38 となります。 <ul style="list-style-type: none"> 反映側 HiRDB のバージョンが 08-04 以降の場合 反映側 DBMS が Oracle 又は SQL Server の場合 上記以外の場合は、精度の最大値は 29 となります。
TIMESTAMP	有	先頭 2 バイト 編集前データ長	要	先頭 2 バイト 編集後データ長	p は、位取り (小数部けた数) を表す数値です。

変換前/変換後 列データ型	編集前反映列情報		編集後反映列情報		備考
	設定 有無	設定内容	設定 要否	設定内容	
		7 + ↑p ÷ 2 ↑ [バイト数] 後ろ2バイト 位取り (小数部け た数)		7 + ↑p ÷ 2 ↑ [バイト数] 後ろ2バイト 位取り (小数部け た数)	
上記以外	無	—	不要	—	データ長は、HiRDB のデータ形式を参照してください。

(c) ステータス

列データ編集 UOC 関数をエラーリターンする場合に設定できます。

設定したステータス値は、Datareplicator に制御が戻った後に、エラーメッセージとともに出力されます。

(d) 戻り値

列データ編集 UOC 関数では、呼び出し元にリターンするときに戻り値を設定してください。

設定できる列データ編集 UOC からの戻り値を次の表に示します。

表 8-30 hds_ucoleditX()からの戻り値

状態	コード	ニモニック	リターン後の処置
正常終了	0	HDS_RET_NORM	反映処理を続行します。
エラー検知 (継続できるレベル)	4	HDS_RET_WARN	呼び出し元でメッセージを出力し、反映処理を続行します。
エラー検知 (継続できないレベル)	0, 4 以外	HDS_RET_ERR	呼び出し元でメッセージを出力し、その後、反映処理は中止します。

(2) 列データ編集 UOC で使うヘッダファイル

列データ編集 UOC で使うヘッダファイルは、次に示すディレクトリに格納してあります。

UNIX 版： /opt/hirdbds/include/

Windows 版の場合：Datareplicator をインストールしたディレクトリ¥include

列データ編集 UOC で使うヘッダファイルの一覧を次の表に示します。

表 8-31 列データ編集 UOC で使うヘッダファイルの一覧

ファイル名	内容	列データ編集 UOC に関連するメンバ
hds_ucommon.h	列データ編集 UOC の共通の構造体及びニモニックコード	<ul style="list-style-type: none"> ・ UINTERFACE_BLK ・ 戻り値のニモニック
hds_ureflect.h	列データ編集 UOC の構造体及びニモニックコード	<ul style="list-style-type: none"> ・ UCOLENV_BLK ・ UREFLECT_BLK ・ UREFLECT_BLK で使うニモニック ・ UCOLUMN_BLK ・ データ型のニモニック ・ データアドレスのニモニック

8.2.5 列データ編集 UOC を作成するときの注意

列データ編集 UOC 実行時には、環境変数に、共用ライブラリファイルが格納されているディレクトリパスを指定してください。なお、環境変数と共用ライブラリファイルは OS ごとに異なります。OS、環境変数、及び共用ライブラリファイルの対応を次の表に示します。

表 8-32 OS、環境変数、及び共用ライブラリファイルの対応

OS	環境変数	共用ライブラリファイル
Linux	LD_LIBRARY_PATH	libhdscuoc.so
HP-UX	SHLIB_PATH	libhdscuoc.so
AIX	LIBPATH	libhdscuoc.a

環境変数が指定されていない場合や、環境変数で指定したディレクトリパス下に共用ライブラリファイルが存在しない場合は、列データ編集 UOC を使わない通常の反映処理が実行されます。

(1) シグナルの制限

列データ編集 UOC 関数内では、シグナルの操作をしないでください。

(2) ファイルの制限

列データ編集 UOC 関数内で操作できるファイルは、ユーザ独自のファイルだけ（関数内でオープンしたファイルだけ）です。ファイルの規則を次に示します。

- 標準入力、標準出力、標準エラー出力は使わないでください。
- 自らがオープンしたファイル以外を read, write, close しないでください。
- HiRDB と Datareplicator のファイルを操作しないでください。

(3) 共用メモリの制限

共用メモリは、Datareplicator やほかのシステムと同じ共用メモリを使わないでください。

(4) データ型の制限

- 次のデータ型の列は、列データ編集 UOC では扱えません。
指定した場合、反映定義の解析でエラーになります。
 - 抽象データ型 (SGMLTEXT, FREEWORD, XML)
 - BLOB 型
 - BINARY 型
- SMALLFLOAT 型データは FLOAT 型データとして UOC に渡します (表現範囲の相違によるオーバーフローを避けるため)。
- 抽出列型が UNPACK データの場合、DECIMAL 型へデータ変換した後に列データ編集 UOC へデータを渡します。ただし、UNPACK データが DECIMAL 型変換できない場合は、UNPACK データ形式で列データ編集 UOC へ渡します。

(5) SQL の実行の制限

列データ編集 UOC 関数内、又は列データ編集 UOC 関数の延長で、SQL を実行しないでください。SQL を実行すると、Datareplicator のトランザクション管理が不正になることがあります。

(6) AIX 版の場合の注意事項

AIX 版では、hds_ucoledit1() から hds_ucoledit8() までの関数のすべてを作成しておいてください。すべてを作成していない場合、反映処理の実行時にエラーとなります。

(7) 繰返し列を扱う場合の注意事項

列データ編集 UOC でのデータの受け渡しは要素データごとになります (1 回の列データ編集 UOC 関数の呼び出しで、一つの要素データを列データ編集 UOC 関数へ渡します)。このため、繰返し列の要素数を列データ編集 UOC で増やしたり減らしたりすることはできません。

8.2.6 列データ編集 UOC のサンプル

反映側 Datareplicator に格納してある UOC のサンプルについて説明します。

(1) 格納ディレクトリとファイル名

ディレクトリ

UNIX 版の場合 : /opt/hirdbds/lib/sample/

Windows 版の場合：Datareplicator をインストールしたディレクトリ¥sample¥

ファイル名

hds_cedit.c

8.3 送信データ UOC

送信データ UOC を作成すると、抽出した更新情報を反映側システムに送信する前に、送信対象とすることがどうかをチェックできます。送信データ UOC は、抽出側 Datareplicator で作成できます。

8.3.1 送信データ UOC の概要

送信データ UOC を作成すると、抽出した更新情報を Datareplicator が反映側システムに送信する前に UOC へ制御が渡されます。Datareplicator は UOC から制御が戻った後に、チェック結果に従って反映側システムに更新情報を送信します。

送信データ UOC には、次に示すデータが渡されます。

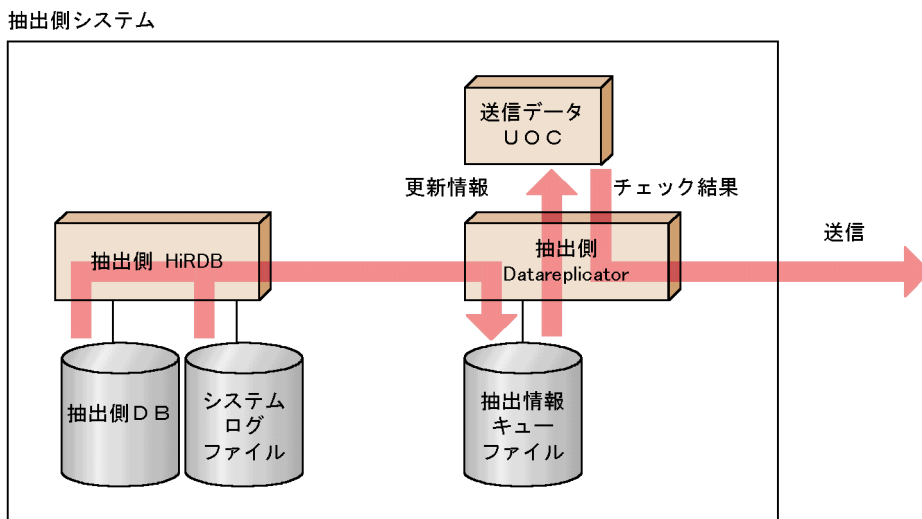
- 抽出定義の send 文の where 句に指定した条件をクリアしたデータ
- ループバック抑止の判定で送信対象になったデータ

送信データ UOC に渡す更新情報は、抽出定義で指定した抽出対象表単位で渡します。したがって、1 表から複数の更新情報を抽出するように指定している場合は、マージされた更新情報 1 件を渡します。

送信データ UOC を使う抽出側 Datareplicator には、送信環境定義の senduoc オペランドに use を指定します。

送信データ UOC を作成した場合の更新情報を送信する手順を次の図に示します。

図 8-12 送信データ UOC を使った場合の更新情報を送信する手順

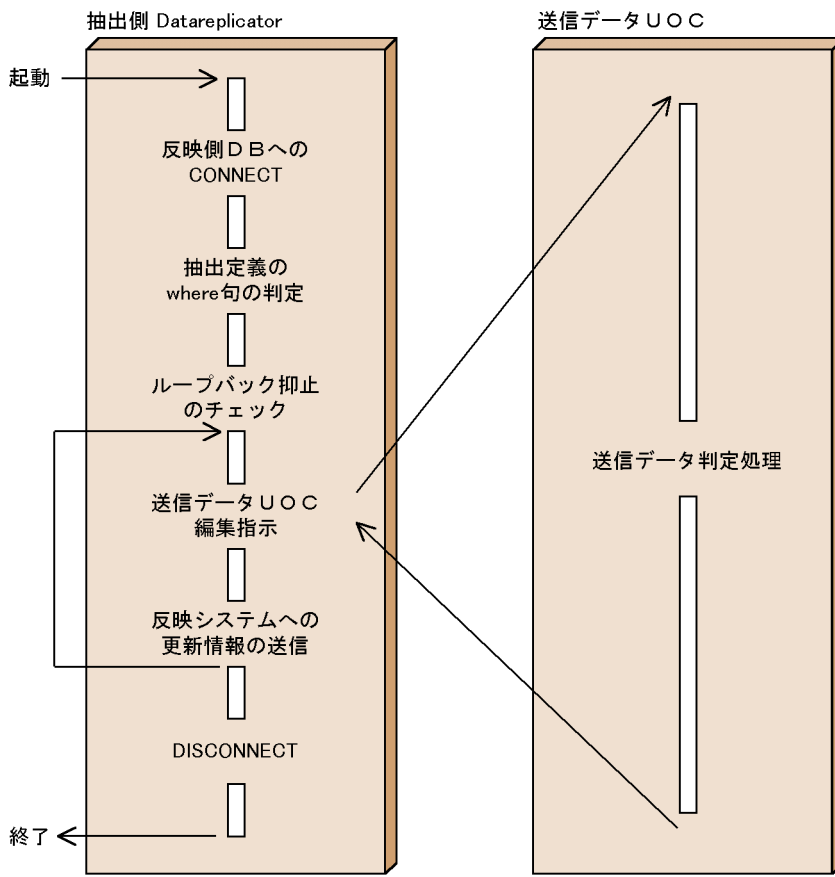


(1) 送信データ UOC が呼び出されるタイミング

送信データ UOC は、抽出定義の where 句の判定が完了して、ループバック抑止のチェックが完了したときに、Datareplicator から呼び出されます。

送信データ UOC の制御の流れを次の図に示します。

図 8-13 送信データ UOC の制御の流れ



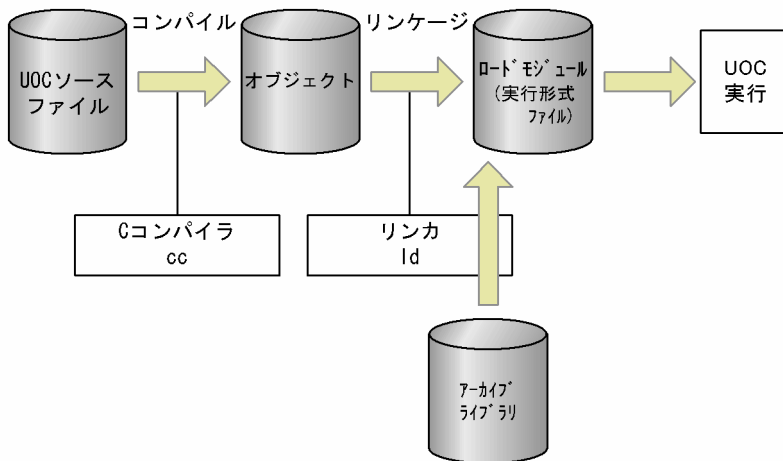
(凡例)

□ : 処理中を表します。

8.3.2 送信データ UOC の作成手順 (UNIX)

UNIX 版 Datareplicator で送信データ UOC を作成する手順について説明します。送信データ UOC を実行するまでの手順を次の図に示します。

図 8-14 送信データ UOC の実行手順



(1) 作成方法

送信データ UOC の共有ライブラリの作成方法を次に示します。

1. 送信データ UOC でチェックする更新情報を決定します。
2. 送信データ UOC の関数を作成します。
3. 送信データ UOC の関数をコンパイル／リンケージします。

(2) コンパイル及びリンケージ

コンパイル及びリンケージについて OS ごとに説明します。

なお、リンケージの未解決エラーが発生する場合は、リンケージのときに必要なライブラリすべてを指定してください。

(a) HP-UX 版の場合

- UOC のコンパイル方法

```
cc -c +DD64 +z -I/opt/hirdbds/include UOCソースファイル名 [UOCソースファイル名] ...
```

-c :

オブジェクト作成オプション

+z :

位置独立コード作成オプション

-I :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。
UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

- UOC のリンケージ方法


```
ld -b -o libhdesuoc.so UOCオブジェクトファイル名 [UOCオブジェクトファイル名] ...
```

(b) AIX 版の場合

- UOC のコンパイル方法

```
xlc -q64 -c -I /opt/hirdbds/include UOCソースファイル名 [UOCソースファイル名] ...
```

-c :

オブジェクト作成オプション

-I :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。
UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

- UOC のリンケージ方法

```
xlc -q64 -bM:SRE -bnoentry -bE:/opt/hirdbds/lib/libhdesuoc.exp -o libhdesuoc.a  
UOCオブジェクトファイル名 [UOCオブジェクトファイル名] ...
```

(c) Linux 版の場合

- UOC のコンパイル方法

```
gcc -c -fPIC -I/opt/hirdbds/include  
UOCソースファイル名 [UOCソースファイル名] ...
```

-c :

オブジェクト作成オプション

-fPIC :

位置独立コード作成オプション

-I :

Datareplicator が提供するヘッダファイル格納ディレクトリを指定 (/opt/hirdbds/include 固定)。
UOC 自身がほかのヘッダファイルを必要とする場合は、複数指定できます。

- UOC のリンケージ方法

```
gcc -shared -o libhdesuoc.so UOCオブジェクトファイル名 [UOCオブジェクトファイル名] ...
```

(3) 注意事項

作成した送信データ UOC のライブラリは、任意のディレクトリに格納できます。ただし、/opt/hirdbds/lib 下にある libhdesuoc.so (HP-UX 版の場合は libhdesuoc.so, Linux 版の場合は libhdesuoc.so, AIX 版の場合は libhdesuoc.exp) とは入れ替えないでください。

UOC を実行する前に、抽出システム定義の `node_shlibpath` オペランドに送信データ UOC ライブラリ (`libhdesuoc.sl` 又は `libhdesuoc.so`) が格納してあるディレクトリを指定しておいてください。
`node_shlibpath` オペランドに指定したディレクトリ下に送信データ UOC ライブラリがないときは、全更新情報を送信対象とします。

8.3.3 送信データ UOC の作成手順 (Windows)

Windows 版 Datareplicator で送信データ UOC を作成する手順について説明します。Windows 版 Datareplicator の UOC は、DLL ファイル (`xxx.dll`) として作成します。

(1) 作成方法

Windows 版の場合、送信データ UOC は Visual Studio で作成します。

次に示す手順で、DLL ファイルを作成してください。

- Windows 版 HiRDB のアプリケーションプログラムを作成する C 言語で送信データ UOC の関数を作成します。送信データ UOC の DLL ファイル名は、`hdesuoc.dll` にしてください。
 Datareplicator の UOC 用のヘッダファイルは、Datareplicator のインストール先ディレクトリの下に `¥include` に格納してあります。
- 作成した関数をコンパイル/リンケージして、DLL ファイルを作成します。Datareplicator から UOC を呼び出すため、インタフェース関数は必ず `__cdecl` 呼び出し規約で作成して、エクスポートしておいてください。

(2) コンパイル及びリンケージ

Visual Studio を使用してコンパイル、リンケージをする場合、オプションの設定は、プロジェクトメニューから「プロパティ」を選択します。

設定する項目を次の表に示します。

表 8-33 Visual Studio の「プロパティ」で設定する項目

項目	カテゴリ	カテゴリの設定	設定値
プラットフォーム	—	—	x64
構成プロパティ	詳細	共通言語ランタイムサポート	共通言語ランタイムサポートを使用しない
C/C++	コード生成	構造体メンバのアラインメント	8 バイト (/Zp8)
		ランタイムライブラリ	マルチスレッド DLL (/MD)
	詳細設定	呼び出し規約	<code>__cdecl</code> (/Gd)

項目	カテゴリ	カテゴリの設定	設定値
	全般	追加のインクルードディレク トリ	Datereplicator のインストー ルディレクトリ%include

(凡例)

－：該当しません。

リンケージの未解決エラーが発生する場合は、リンケージのときに必要なライブラリすべてを指定してください。

(3) 注意事項

- 抽出側 Datereplicator 又はほかのシステムで使っているメモリマップトファイルは、UOC では使わないでください。
- UOC の処理がエラーになったときは、エラーログファイルのプログラム名には DLL ファイルの名称が出力されます。
- UOC を配置したフォルダをシステム環境変数の path に追加してください。

8.3.4 送信データ UOC で使う関数の文法

送信データ UOC は、C 言語でプログラミングします。抽出側 Datereplicator には、送信データ UOC を作成するときに使う関数を備えています。

送信データ UOC で使える関数名と機能の概要を次の表に示します。

表 8-34 送信データ UOC で使える関数と機能の概要

関数名	機能
hde_usendcheck () (送信データの判定)	送信データ (更新情報) を受け取って、チェック結果を格納します。ユーザは、チェックする内容を指定して、結果を返します。

(1) hde_usendcheck (送信データの判定)

送信データ UOC で使う関数は、hde_usendcheck() です。一つの抽出側 Datereplicator システム内に 1 個の UOC 関数を作成できます。

(a) 関数の形式

```
#include<hde_usend.h>
int hde_usendcheck (
HDE_EXT_ENVINFO *envinfo, /* (IN) 送信環境情報ブロック*/
HDE_EXT_TBLINFO *tblinfo, /* (IN) 抽出表情報ブロック */
HDE_EXT_COLINFO *colinfo, /* (IN) 抽出データ情報ブロック*/
int *status ) ; /* (OUT) 送信UOCステータス */
```

(b) パラメタの説明

パラメタの内容を次に示します。

関数の構造体は、送信データ UOC のヘッダファイルで定義しています。構造体の設定者が呼び出し元の領域は、更新しないでください。ヘッダファイルについては、「表 8-42 送信データ UOC のヘッダファイルの一覧」を参照してください。

• HDE_EXT_ENVINFO

送信データ UOC の環境情報を格納するブロックです。

HDE_EXT_ENVINFO の内容を次の表に示します。

表 8-35 HDE_EXT_ENVINFO の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
sendid	9	char	送信先識別子	呼び出し元	該当する更新情報の送信先を示す識別子 (NULL で終了)
hdeid	1	unsigned char	抽出側 Datareplicator 識別子	呼び出し元	抽出システム定義 hdeid オペランドの指定値 識別子 2 文字は、1 バイトで表現しています。 (例) c1 → 0xC1
hdsid	1	unsigned char	反映システム識別子	呼び出し元	送信環境定義 sendhdsid オペランドの指定値 識別子 2 文字は、1 バイトで表現しています。 (例) c1 → 0xC1
dsid	1	unsigned char	データ連動識別子	呼び出し元	送信環境定義 dsid オペランド、又は抽出環境定義の dsid オペランドの指定値 (送信環境定義 dsid オペランドが優先) 識別子 2 文字は、1 バイトで表現しています。 (例) c1 → 0xC1
reserve1	20	char	予備	呼び出し元	予備 (0x00...00)

• HDE_EXT_TBLINFO

HDE_EXT_TBLINFO の内容を次の表に示します。

表 8-36 HDE_EXT_TBLINFO の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
tableid	4	int	抽出表 ID	呼び出し元	該当する更新情報の抽出表 ID
tablename	31	char	抽出表名	呼び出し元	該当する更新情報の抽出表名 (NULL で終了)
schemaname	9	char	認可識別子	呼び出し元	該当する更新情報の抽出表の認可識別子 (NULL で終了)
exttime	8	char	抽出時間	呼び出し元	該当する更新情報の抽出時間

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
					10 進符号なしパック形式 (YYYYMMDDhhmmss00) で出力します。 YYYY：西暦年 MM：月 DD：日 hh：時 mm：分 ss：秒
extkind	1	unsigned char	更新種別	呼び出し元	更新した SQL 種別 INSERT： HDE_EXT_INSERT (0x01) UPDATE： HDE_EXT_UPDATE (0x02) DELETE： HDE_EXT_DELETE (0x03) PURGE： HDE_EXT_PURGE (0x04)
apkind	1	unsigned char	ループバック情報	呼び出し元	ループバック情報 0x00： UAP からの更新情報 0x01： 反映側 Datareplicator (hdesqle オペランド) からの更新情報
reserve1	11	char	予備	呼び出し元	予備 (0x00...00)

• HDE_EXT_COLINFO

HDE_EXT_COLINFO の内容を「[表 8-37 HDE_EXT_COLINFO の内容](#)」に、
HDE_EXT_COLINFO から参照できる HDE_EXT_DATAINFO の内容を「[表 8-38 HDE_EXT_DATAINFO の内容](#)」に示します。

HDE_EXT_TBLINFO の extkind の値が HDE_EXT_PURGE のときは、HDE_EXT_COLINFO の内容は NULL 値になって参照できません。

表 8-37 HDE_EXT_COLINFO の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
colnum	4	int	列情報数	呼び出し元	該当する更新情報にある抽出列の数 (HDE_EXT_DATAINFO のエントリ数)

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
*datainfo	4	HDE_EXT_DATAINFO *	抽出データ情報の先頭アドレス	呼び出し元	抽出データ情報 (HDE_EXT_DATAINFO) の配列の先頭アドレス
reserve	8	char	予備	呼び出し元	予備

表 8-38 HDE_EXT_DATAINFO の内容

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
colid	4	int	抽出列 ID	呼び出し元	抽出列 ID
colname	31	char	抽出列名	呼び出し元	抽出列名 (NULL で終了)
coltype	1	unsigned char	列データ型	呼び出し元	抽出列のデータ型 列データ型の一覧については、「表 8-40 列データ型の一覧」を参照してください。
colopt	1	unsigned char	列オプション	呼び出し元	抽出列のオプション マッピングキー： HDE_EXT_MAPPINGKEY (0x01) 繰返し列： HDE_EXT_MCOL (0x02)
charset	1	unsigned char	文字集合種別	呼び出し元	該当する列の文字集合種別 文字集合指定なし： HDE_CSET_DEFAULT 0x00 文字集合指定あり (EBCDIK)： HDE_CSET_EBCDIK 0x01
reserve1	2	char	予備	呼び出し元	予備 (0x00...00)
bf_dataalen	4	int	更新前の列データ長 ^{*1}	呼び出し元	抽出列の更新前の列データ長 0： 0 バイトデータ, 又は NULL 値を示します。 -1： 更新前の列データが未使用であることを示します。
bf_dataptr	4	union ^{*2}	更新前の列データ ^{*1}	呼び出し元	抽出列の更新前の列データへの先頭アドレス 0： NULL 値を示します。 -1： 更新前の列データが未使用であることを示します。

メンバ名	長さ (バイト)	属性	領域名	設定者	内容
af_dataLEN	4	int	更新後の列データ長 ^{※1}	呼び出し元	抽出列の更新後の列データ長 0: 0バイトデータ, 又は NULL 値を示します。 -1: 更新後の列データが未使用であることを示します。
af_dataptr	4	union ^{※2}	更新後の列データ ^{※1}	呼び出し元	抽出列の更新後の列データへの先頭アドレス 0: NULL 値を示します。 -1: 更新後の列データが未使用であることを示します。
precision	2	short	精度	呼び出し元	coltype が HDE_T_DEC の場合に精度 (全体のけた数) が設定されます。それ以外の場合には 0 が設定されます。
scale	2	short	位取り	呼び出し元	coltype が HDE_T_DEC の場合に位取り (小数点以下のけた数) が設定されます。それ以外の場合には 0 が設定されます。
sec_preci	2	short	小数秒のけた数	呼び出し元	coltype が HDE_T_TIMESTAMP の場合に小数秒のけた数が設定されます。それ以外の場合には 0 が設定されます。
reserve2	2	char	予備	呼び出し元	予備 (0x00...00)

注※1

更新前データと更新後データの組み合わせの一覧を「表 8-39 更新前データと更新後データの組み合わせ」に示します。DATAINFO の並び (配列) については、INSERT のときにはカラム ID でソート、UPDATE、DELETE のときには抽出順になります。

注※2

該当する抽出列の属性が CHAR、又は VARCHAR の場合、渡されるデータは該当する抽出列の文字集合となります。

表 8-39 更新前データと更新後データの組み合わせ

データ種別		更新前 データ長	更新前 データ	更新後 データ長	更新後 データ
INSERT	n バイトデータ	-1	-1	n	addr
	0 バイトデータ	-1	-1	0	addr
	NULL データ	-1	-1	0	0

データ種別			更新前 データ長	更新前 データ	更新後 データ長	更新後 データ
非マッピングキーの UPDATE	n バイト データ から	n バイトデータへ	-1	-1	n	addr
		0 バイトデータへ	-1	-1	0	addr
		NULL データへ	-1	-1	0	0
	0 バイト データ から	n バイトデータへ	-1	-1	n	addr
		0 バイトデータへ	-1	-1	0	addr
		NULL データへ	-1	-1	0	0
	NULL データ から	n バイトデータへ	-1	-1	n	addr
		0 バイトデータへ	-1	-1	0	addr
		NULL データへ	-1	-1	0	0
マッピングキーの UPDATE	n バイト データ から	n バイトデータへ	n	addr	n	addr
		0 バイトデータへ	n	addr	0	addr
		NULL データへ	n	addr	0	0
	0 バイト データ から	n バイトデータへ	0	addr	n	addr
		0 バイトデータへ	0	addr	0	addr
		NULL データへ	0	addr	0	0
	NULL データ から	n バイトデータへ	0	0	n	addr
		0 バイトデータへ	0	0	0	addr
		NULL データへ	0	0	0	0
DELETE*	n バイトデータ		n	addr	-1	-1
	0 バイトデータ		0	addr	-1	-1
	NULL データ		0	0	-1	-1

注※
渡されるデータはマッピングキーのデータだけです。可変長データの同値更新、又は非更新の場合、変更前後のデータ情報はすべて「-1」になります。

列データ型の一覧を次の表に示します。

表 8-40 列データ型の一覧

列データ型	ニモニック	コード
INTEGER	HDE_T_INT	0xF1
SMALLINT	HDE_T_SINT	0xF5

列データ型	ニモニック	コード
LARGE DECIMAL	HDE_T_DEC	0xE5
FLOAT	HDE_T_FLT	0xE1
DOUBLE PRECISION	HDE_T_DBL	0xE1
SMALL FLOAT	HDE_T_SFLT	0xE3
REAL	HDE_T_REAL	0xE3
CHARACTER	HDE_T_CHAR	0xC5
VARCHAR	HDE_T_VCHAR	0xC1
NCHAR	HDE_T_NCHAR	0xB5
NVARCHAR	HDE_T_NVCHAR	0xB1
MCHAR	HDE_T_MCHAR	0xA5
MVARCHAR	HDE_T_MVCHAR	0xA1
DATE	HDE_T_DATE	0x71
TIME	HDE_T_TIME	0x79
TIMESTAMP	HDE_T_TIMESTAMP	0x7D
INTERVAL YEAR TO DAY	HDE_T_YTD	0x65
INTERVAL HOUR TO SECOND	HDE_T_HTS	0x6F
ADT (抽象データ型)	HDE_T_ADT	0x83
BLOB	HDE_T_BLOB	0x93
BINARY	HDE_T_BINARY	0x91

- **int *status**

UOC の状態コードを示します。UOC 内でエラーが起こったときの状態を表示するために使います。関数の呼び出し時は、0 で初期化して渡されます。

ここに設定されたコードは、戻り値が HDE_EXT_SEND (1), HDE_EXT_NOSEND (0) 以外の場合に出力されるエラーメッセージ KFRB02052-E の埋字として表示されますが、設定された値は送信プロセスの動作には影響しません。

(c) 戻り値

送信データ UOC 関数では、呼び出し元にリターンするときに戻り値を設定してください。

設定できる送信データ UOC からの戻り値を次の表に示します。

表 8-41 hde_usendcheck()からの戻り値

状態	コード	ニモニック	リターン後の処置
更新情報を送信	1	HDE_EXT_SEND	該当する更新情報を反映システムに送信します。
更新情報を送信しない	0	HDE_EXT_NOSEND	該当する更新情報は反映システムに送信しません。
送信処理を停止	-1	HDE_EXT_STOP	反映システムへの送信処理を停止します。この値が返ったときは、送信プロセスからエラーメッセージ KFRB02052-E が出力されて、該当する送信先に対する処理を中止します。 送信処理を停止した理由は、hde_usendcheck()の status に返ります。
	上記以外		

(2) 送信データ UOC で使うヘッダファイル

送信データ UOC で使うヘッダファイルは、次に示すディレクトリに格納してあります。

UNIX 版の場合：/opt/hirdbds/include/

Windows 版の場合：Datareplicator をインストールしたディレクトリ¥include¥

送信データ UOC のヘッダファイルを次の表に示します。

表 8-42 送信データ UOC のヘッダファイルの一覧

ファイル名	内容	送信データ UOC に関連するメンバ
hde_usend.h	送信データ UOC の構造体及びニモニックコード	<ul style="list-style-type: none"> ・ HDE_EXT_ENVINFO ・ HDE_EXT_TBLINFO ・ HDE_EXT_COLINFO ・ HDE_EXT_DATAINFO ・ データ型のニモニック ・ データアドレスのニモニック

8.3.5 送信データ UOC を作成するときの注意

送信データ UOC 実行時には、環境変数 SHLIB_PATH (Linux 版の場合は LD_LIBRARY_PATH, AIX 版の場合は LIBPATH) に共用ライブラリファイル (libhdesuoc.so 又は libhdesuoc.exp) が格納されているディレクトリパスを指定してください。

環境変数 SHLIB_PATH (Linux 版の場合は LD_LIBRARY_PATH, AIX 版の場合は LIBPATH) が指定されていない場合や、環境変数で指定したディレクトリパス下に共用ライブラリファイルが存在しない場合は、送信データ UOC を使わない通常の反映処理が実行されます。

(1) シグナルの制限

- 送信データ UOC 関数内では、シグナルの操作をしないでください。
- プロセスを制御する関数 (exit(), abort() など) は使わないでください。
- 送信データ UOC 内で、HDE_EXT_DATAINFO 中の更新前の列データ、更新後の列データのアドレスで free() を実行しないでください。

(2) ファイルの制限

送信データ UOC 関数内で操作できるファイルは、ユーザ独自のファイルだけ (UOC 関数でオープンしたファイルだけ) です。ファイルの規則を次に示します。

- 標準入力、標準出力、標準エラー出力は使わないでください。
- 自らがオープンしたファイル以外を read, write, close しないでください。
- HiRDB と Datareplicator のファイルを操作しないでください。
- 送信データ UOC に到着したデータを加工しないでください。
- 送信データ UOC 内で確保した資源は、次の送信データ UOC の呼び出しに引き継ぎません。

(3) 共用メモリの制限

共用メモリは、Datareplicator やほかのシステムと同じ共用メモリを使わないでください。

(4) データ型の制限

次のデータ型の列は、送信データ UOC では扱えません。

- 抽象データ型 (SGMLTEXT, FREEWORD, XML)
- 追加更新をしている BLOB 型及び BINARY 型
- 後方削除更新をしている BLOB 型及び BINARY 型

(5) SQL の実行の制限

送信データ UOC 関数内、又は送信データ UOC 関数の延長で、SQL を実行しないでください。このように SQL を実行すると、Datareplicator のトランザクション管理が不正になることがあります。

(6) そのほかの制限

- 送信文字コードを EBCDIK 指定にした場合でも、渡されるデータは EBCDIK 変換前の文字コードで渡します。
- 同じ送信先に対する処理でも、複数のプロセスに分けて処理されることを考慮して送信データ UOC を作成してください。

- 引数で渡す表 ID, 列 ID は hdeprep コマンド実行時 ID が渡されます。そのため, UOC の中でこれらの ID を使っているときに, hdeprep コマンド以降に表を再作成すると UOC の修正も必要になります。

8.3.6 送信データ UOC のサンプル

Datereplicator に格納してある UOC のサンプルについて説明します。

(1) 格納ディレクトリとファイル名

ディレクトリ

UNIX 版の場合: /opt/hirdbds/lib/sample/

Windows 版の場合: Datereplicator をインストールしたディレクトリ¥sample¥

ファイル名

hde_uocsample.c

(2) ソースファイルの内容

送信データ UOC のサンプルは, 次に示す仕様で作成しています。

- 在庫表の商品コードと在庫量をチェックして該当するデータを送信します。
- 条件式は次のとおりで, 該当する場合は HDE_EXT_SEND を戻り値として設定します。
商品コード = 10 でかつ在庫量が 100 以上
又は
商品コード = 20 でかつ在庫量が 50 以上
- 在庫表以外の抽出表のデータは, すべて送信対象とします。

(3) サンプルのライブラリ作成手順

次に, サンプルのライブラリ作成手順を HP-UX の場合の例で示します。

```
$ cc -c +DD64 +z -I/opt/hirdbds/include hde_uocsample.c
$ ld -b -o libhdesuoc.so hde_uocsample.o
```

9

障害時の対処

この章では、抽出側 Datareplicator 又は反映側 Datareplicator を運用中に障害が発生した場合の対処について説明します。

9.1 抽出側 Datareplicator の障害時の対処

抽出側 Datareplicator で障害が起こったときの対処方法について説明します。

反映側システムの障害の対処方法については、HiRDB の反映側システムのときは「[9.2 反映側 Datareplicator の障害時の対処](#)」を、HiRDB 以外の反映側システムのときは該当するマニュアルを参照してください。

9.1.1 障害の対処の手順

抽出側 Datareplicator の障害に対処する手順について説明します。

(1) 対処の手順

1. エラーメッセージの内容を確認します。

syslog ファイル（Windows の場合はイベントログ）にメッセージが出力されます。メッセージの内容を確認してください。

2. 次のエラー情報ファイルに、syslog ファイル（Windows の場合はイベントログ）と同じエラーメッセージが出力されているかどうかを確認します。

- 抽出マスタエラー情報ファイル
- 抽出ノードマスタエラー情報ファイル

Datereplicator の稼働中にエラー情報ファイルがスワップ又はクローズされると、メッセージ（KFRB00051-I, KFRB00052-I）が出力されます。このメッセージを契機にバックアップを取得しているときは、その内容を確認してください。

3. 抽出側 Datareplicator の稼働トレースファイルを `hdstrcredit` コマンドで編集し、解析した結果を保存します。

4. メッセージの内容を基に、エラーに対処します。

エラーの対処方法については、「[9.1.2 エラーの対処方法](#)」を参照してください。

5. エラーの対処後の処置を実行します。

エラーの原因によっては、障害が発生したプロセスに対応した処置をする必要があります。障害が発生したプロセスごとの処置については、「[9.1.3 エラー対処後の処置](#)」を参照してください。

エラーの内容によっては、抽出側 Datareplicator のプロセス又は共用メモリ（Windows 版の場合はメモリマップトファイル）が残っていることがあります。

UNIX 版 Datareplicator の場合は、残ったプロセス又は共用メモリは Datareplicator 再起動時に削除されます。

Windows 版 Datareplicator の場合は、残ったプロセス又はメモリマップトファイルを次の手順で削除してください。

手順	操作
1	タスクマネージャを起動し、「プロセス」タブで表示される Datareplicator プロセスのプロセス ID を取得します。
2	手順 1 で取得したプロセス ID に該当するプロセスを停止します。 pdkill プロセス ID*
3	インストールディレクトリ%tmp%hde%spool%shm 下にあるメモリマップトファイルを削除します。

注※

Datareplicator Extension を使用している場合は、pdkill コマンドでなく hdskill コマンドを使用してください。

(2) システム管理者への連絡

エラーのうち、システム管理者に連絡する必要がある場合は、次のように処置してください。

1. 次に示すデータを取得します。

- syslog ファイル (Windows の場合はイベントログ) に出力されたメッセージ
- 抽出側 Datareplicator のステータス情報
hdestate コマンドを実行して、ステータス情報を取得します。
- コアファイル (プロセス全体のダンプファイル)
環境変数 HDEPATH で設定したディレクトリに、次に示す形式で出力されます。

core プロセス ID

プロセス ID はエラーが発生したプロセスを識別する番号です。

ただし、抽出側 Datareplicator を起動したユーザが環境変数 HDEPATH で指定したディレクトリに対して書き込み権限を持っていない場合、コアファイルは作成されません。

- メモリマップトファイル (Windows 版の場合)
- 抽出マスタステータスファイル
- 抽出サーバステータスファイル
- 抽出マスタエラー情報ファイル
- 抽出ノードマスタエラー情報ファイル
- 稼働トレースファイル
- 定義ファイル (変更の可能性がある場合)

2. エラーの内容を調査して対処します。

内部矛盾などのエラーの場合には、保守員に連絡してください。

9.1.2 エラーの対処方法

抽出側システムのエラーの原因と対処方法を次の表に示します。なお、障害が発生して抽出側 DB と反映側 DB の不整合が発生した場合は、抽出側 Datareplicator 及び反映側 Datareplicator を初期化する必要があります。初期化手順の詳細については、「9.4 障害回復時の初期化手順」を参照してください。

表 9-1 エラーの原因と対処方法

エラーの原因の種類		エラーの対処方法
メモリ不足（ローカルメモリ）		<ul style="list-style-type: none"> 動作中の不要なプロセスを終了してください。 メモリを増設してください。
メモリ不足（共用メモリ）		<ul style="list-style-type: none"> 共用メモリを使っている、動作中の不要なプロセスを終了してください。 メモリを増設してください。
ファイルディスクリプタ不足		<ul style="list-style-type: none"> 不要なファイルをクローズしてください。 システムパラメタ（ファイルディスクリプタ数の上限値）を調整してください。
ソケット確立不可		システムコールからの詳細コードを基にエラーの原因を取り除いてください。
ポートチェックエラー		詳細コードを基にエラーの原因を取り除いてください。
通信回線切断		通信回線を再接続してください。
ファイル容量不足	抽出情報キューファイル	抽出情報キューファイルに格納してあるすべての更新情報を反映側システムに送信し終わってから、hdestop コマンドで抽出側 Datareplicator を終了してください。その後、ファイルの数を増やす又は容量を拡張した後、hdestart -i コマンドでファイルを初期化して、hdestart コマンドで再起動してください。 詳細は、「抽出情報キューファイル満杯時の運用手順」のフローチャートに従って対処してください。
	抽出時のステータスファイル	抽出情報キューファイルに格納されているすべての更新情報を反映側システムに送信し終わってから、hdestop コマンドで抽出側 Datareplicator を終了してください。その後、ファイルの容量を拡張した後、hdestart -i コマンドでファイルを初期化して、hdestart コマンドで再起動してください。
媒体不正	抽出情報キューファイル	抽出情報キューファイルにだけ障害が発生した場合 エラーの原因を取り除いた後、抽出情報キューファイル回復機能を使用して、抽出情報キューファイルを回復してください。抽出情報キューファイル回復機能については、「9.7 抽出情報キューファイル回復機能」を参照してください。 抽出情報キューファイル以外にも障害が発生した場合 エラーの原因を取り除いた後、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。
	抽出時のステータスファイル	エラーの原因を取り除いた後、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。
	データ連動用連絡ファイル	
	システムログファイル※	

エラーの原因の種類		エラーの対処方法
	抽出側 DB	抽出側 DB のバックアップとバックアップ取得以降のログを基にして、データベース回復ユーティリティ (pdrstr) で回復してください。それ以外の方法で回復した場合は、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。
	抽出情報キューファイルの満杯	hdestate コマンドで送信プロセスの状態を確認して、正常に稼働していない場合は、エラー原因を取り除いた後、エラーが発生している送信プロセスを再起動してください。 送信プロセスでのエラーが取り除けない場合などに、データ連動を中止するときは、抽出側 Datareplicator を強制停止してください。その後、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。 詳細は、「抽出情報キューファイル満杯時の運用手順」のフローチャートに従って対処してください。 抽出情報キューファイルの満杯時に縮退対象になった送信先がある場合の処置については、「9.1.4 反映側システムでエラーが発生したときの対処」を参照してください。
	更新情報不正	更新情報の不正が起こった直後に抽出したステータスファイルと抽出情報キューファイルを保存して保守員に連絡してください。 エラーの原因を取り除いた後、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。
	定義情報不正	抽出側システム又は抽出側 Datareplicator の定義を修正してください。
	SQL エラー	SQLCODE を基にエラーの原因を取り除いてください。
	内部矛盾	保守員に連絡してください。
	マシン電源切断	マシンを再起動してください。
	バスエラー	保守員に連絡してください。
	不正シグナル受信 (sigkill)	UNIX 版 Datareplicator の場合 不正シグナルを受信した場合は、hdesmclean コマンドを実行してください。 Windows 版 Datareplicator の場合 マシンを再起動してください。

注※

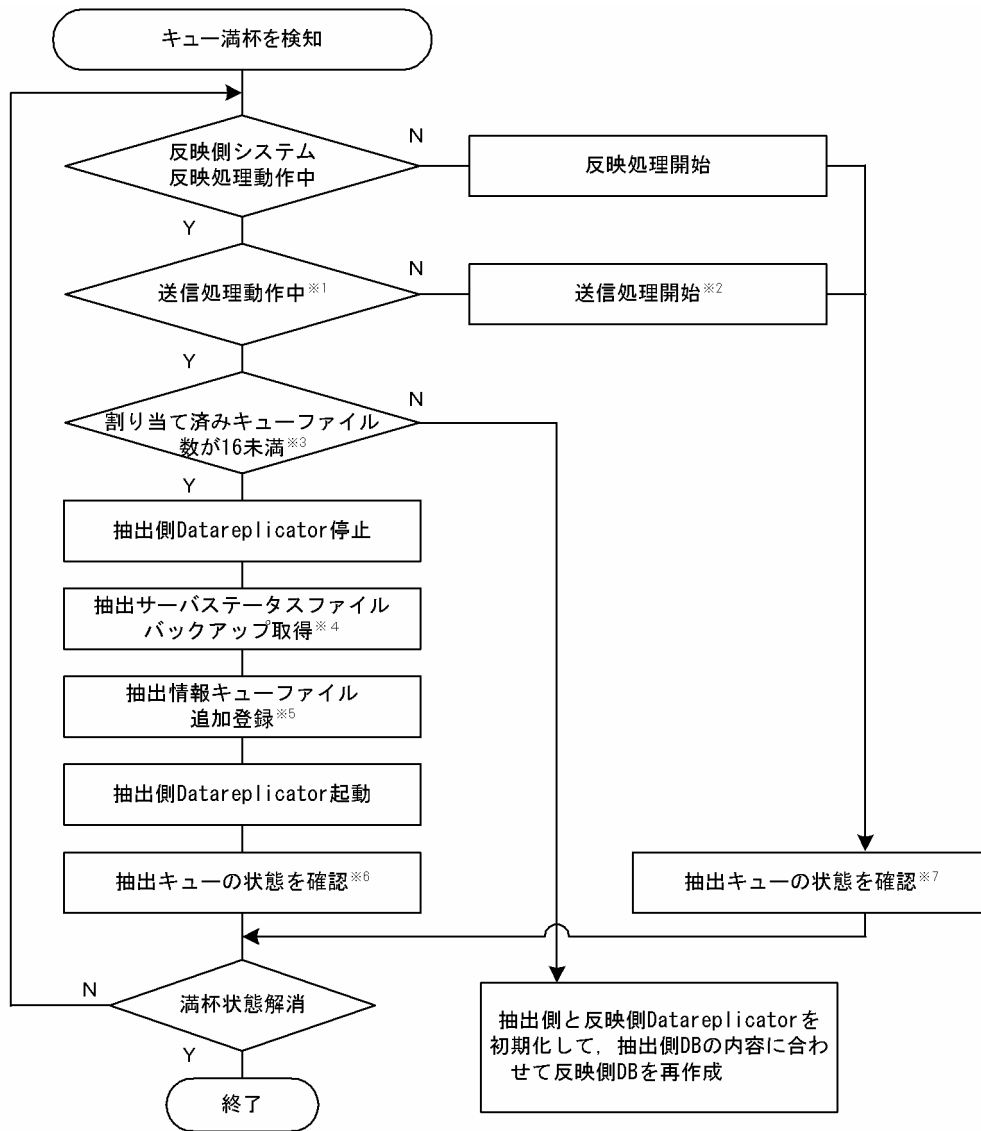
二重化している場合は、両系障害が発生したとき。

抽出情報キューファイル満杯時の運用手順

抽出情報キューファイルが満杯になると、エラーメッセージが出力されます。出力されるメッセージと出力するプロセスとの関係を次に示します。

- 抽出ノードマスタ：KFRB00042-E
- 抽出マスタ：KFRB00701-E

これらのメッセージが出力された場合の運用手順を次に示します。



注※1

hdestate コマンドを実行して、送信処理が動作中かどうかを確認します。

注※2

必要に応じて停止した要因を取り除いて、hdestart -s コマンドで送信処理を開始します。

注※3

hdestate コマンドの実行結果から、抽出情報キューファイルが幾つ割り当てられているかを確認します。

注※4

抽出情報キューファイルの追加登録に失敗したときのために、バックアップを取得してください。バックアップ方法については、「6.4.2(7)(b) 抽出サーバステータスファイルのバックアップ」を参照してください。

注※5

抽出情報キューファイルは、`hdemodq` コマンドで追加登録できます。`hdemodq` コマンドについては、「6.4.2(6)(b) 抽出情報キューファイルの構成を変更するコマンド (`hdemodq` コマンド)」又は「7. コマンドの文法」の `hdemodq` コマンドの文法の説明を参照してください。

注※6

抽出情報キューファイル全体を読み込むのに必要な時間（目安として 1MB 当たり 1 秒）が経過した後、送信間隔（指定値が 0 の場合は 1 分）ごとに数回 `hdestate` コマンドを実行して抽出情報キューファイルへの処理状態を確認します。「Queue read position」が「Queue write position」の次の抽出情報キューファイル示しているときは満杯状態が解消されていないため、抽出情報キューファイルを追加します。

注※7

送信間隔（指定値が 0 の場合は 1 分）が経過した後、送信間隔ごとに数回 `hdestate` コマンドを実行して抽出情報キューに対する処理状態を確認します。「Queue read position」が「Queue write position」の次の抽出情報キューファイル示しているときは満杯状態が解消されていないため、抽出情報キューファイルを追加します。

9.1.3 エラー対処後の処置

障害が発生したプロセスごとの抽出側 Datareplicator の処理と、エラー対処後の処置を次の表に示します。

表 9-2 エラー対処後の処置

障害が発生したプロセス	抽出側 Datareplicator の処理	ユーザーが取る処置
抽出マスタプロセス	抽出側 Datareplicator を終了する（各バックエンドサーバ下の抽出側 Datareplicator も終了します）。 エラーメッセージは、抽出マスタエラー情報ファイルと抽出ノードマスタエラー情報ファイルに出力します。	エラー対処後、 <code>hdestart</code> コマンドで抽出側 Datareplicator を再起動してください。
抽出ノードマスタプロセス	エラーが発生したサーバマシンでのすべての抽出処理と送信処理を停止します。 エラーメッセージは、抽出ノードマスタエラー情報ファイルに出力します。	エラー対処後、エラーが発生したバックエンドサーバのサーバマシンの抽出処理、送信処理を <code>hdestart</code> コマンドで再起動してください。
抽出プロセス	エラーが発生したバックエンドサーバの抽出処理だけを停止します。 エラーメッセージは、抽出ノードマスタエラー情報ファイルに出力します。	エラー対処後、 <code>hdestart -e</code> コマンドで停止している抽出処理を再起動してください。
送信プロセス	エラーが発生した送信処理だけを停止します。 エラーメッセージは、抽出ノードマスタエラー情報ファイルに出力します。	エラー対処後、 <code>hdestart -s</code> コマンドで停止している送信処理を再起動してください。

障害が発生したプロセス	抽出側 Datareplicator の処理	ユーザが取る処置
送信マスタプロセス	送信処理はすべて停止します。 エラーメッセージは、抽出ノードマスタエラー情報ファイルに出力します。	エラー対処後、hdestart -s コマンドで停止している送信処理を再起動してください。
抽出コマンドプロセス	メッセージ出力後、コマンドの実行を中止します。 エラーメッセージは、標準出力に出力します。	エラー対処後、コマンドを再実行してください。
稼働トレース収集プロセス	エラーになるとメッセージが出力されて、稼働トレースは取得されなくなります。 Datareplicator の処理は続行します。	メッセージの内容を参照して、エラーの原因を取り除いてください。

9.1.4 反映側システムでエラーが発生したときの対処

反映側システムでエラーが発生したときの対処について説明します。ここでは、次の項目について説明します。

- 抽出側 DB と反映側 DB との間に不整合が発生した場合
- 複数の反映側システムへ送信している場合の反映側 DB の再作成
- 抽出情報キューファイルの満杯時に縮退対象になった送信先がある場合の処置

(1) 抽出側 DB と反映側 DB との間に不整合が発生した場合

反映側システムでの障害のために、抽出側 DB と反映側 DB との間に不整合が発生した場合は、反映側 DB の再作成が必要です。抽出側と反映側の両方のデータ連動の環境を同期を取って初期化し、抽出側 DB を基に反映側 DB を再作成してください。反映側 DB の再作成については、「6.5 抽出側 HiRDB の運用」を参照してください。

(2) 複数の反映側システムへ送信している場合の反映側 DB の再作成

一つの抽出側システムから複数の反映側システムへ送信している場合に、ある特定の反映側システムで障害が発生したときは、抽出側システムでは部分初期開始を使用すると、障害が発生した反映側システムに対する送信処理だけを初期化できます。この場合、障害が発生した反映側システムに対する送信処理だけを部分初期開始し、障害が発生した反映側システムの反映対象 DB を再作成してください。

この場合の運用手順を次に示します。

1. 連動対象の抽出側 DB と反映側 DB との整合性を保つために、抽出側 HiRDB の業務を停止します。
2. 抽出側のデータ連動対象 DB を pdhold コマンドで閉塞して、データ連動対象 DB の更新を抑止します。
3. 抽出側 Datareplicator を hdestop コマンドで終了します。
4. 反映側 Datareplicator を hdsstop コマンドで終了します。

- 抽出側 DB を基に、障害が発生した反映側システムの反映側 DB だけを再作成します。この場合、HiRDB Dataextractor を使うと、抽出側 DB から反映側 DB を効率的に作成できます。
- 障害が発生した送信先を指定して、抽出側 Datareplicator を部分初期開始します (hdestart -i -S コマンドを実行)。
- 障害が発生した反映側システムの反映側 Datareplicator の環境を初期化します (hdsstart -i コマンドを実行)。
- hdestart コマンドで抽出側 Datareplicator を起動します。
- 抽出側のデータ連動対象 DB の閉塞を pdrels コマンドで解除します。
- 抽出側 HiRDB の業務を再開します。

(3) 抽出情報キューファイルの満杯時に縮退対象になった送信先がある場合の処置

抽出情報キューファイルが満杯になったときに縮退対象になった送信先システムがある場合、縮退対象になった送信先システムの反映側 DB を再作成する必要があります。

この場合の運用手順は、複数の反映側システムへ送信している場合にある特定の反映側システムで障害が発生したときの運用手順と同じです。運用手順については、「(2) 複数の反映側システムへ送信している場合の反映側 DB の再作成」を参照してください。

9.1.5 ユーザOWNコーディングの障害時の対処

UOC の稼働中に障害が発生した場合、抽出側 Datareplicator は UOC が設定した戻り値に対応する処置を実行します。UOC の戻り値については、「8. ユーザOWNコーディング」を参照してください。

UOC が異常終了すると、反映側 Datareplicator はメッセージを出力して異常終了します。

9.1.6 ファイルの二重化運用時のファイル障害への対処

ファイルの二重化して運用しているときにファイル障害が発生した場合、ファイルの回復作業が必要になります。以降、障害の対処方法を説明します。

(1) A 系と B 系の両方に障害が発生した場合

A 系及び B 系のファイルの両方に障害が発生した場合は、Datareplicator を初期化してください。

(2) A 系と B 系のどちらかに障害が発生した場合

A 系又は B 系のファイルのどちらかに障害が発生した場合は、障害の発生したファイルによって対処方法が異なります。

Datareplicator が自動的に回復するファイル

次に示すファイルに障害が発生した場合、Datareplicator が自動的に障害を回復します。回復は、障害発生後のそのファイルを使用するプロセスが起動した時に行われます。

- 抽出マスタステータスファイル
- 抽出サーバステータスファイル
- データ連動用連絡ファイル
- 反映マスタステータスファイル
- 反映ステータスファイル

手動で回復する必要があるファイル

次に示すファイルに障害が発生した場合、現用ファイル複写コマンド (`hdefcopy` 又は `hdsfcopy`) でファイルを回復する必要があります。コマンドの詳細については、「7. コマンドの文法」の `hdefcopy` コマンド又は `hdsfcopy` コマンドを参照してください。

- 抽出情報キューファイル
- 反映情報キューファイル

9.2 反映側 Datareplicator の障害時の対処

反映側 Datareplicator で障害が起こったときの対策方法について説明します。

抽出側システムの障害の対処方法については、HiRDB の抽出側システムのときは「[9.1 抽出側 Datareplicator の障害時の対処](#)」を、HiRDB 以外の抽出側システムのときは該当するマニュアルを参照してください。

9.2.1 障害の対処の手順

反映側 Datareplicator の障害に対処する手順について説明します。

(1) 対処の手順

1. エラーメッセージの内容を確認します。

syslog ファイル（Windows の場合はイベントログ）にメッセージが出力されます。メッセージの内容を確認してください。

2. 反映エラー情報ファイルのエラーログに、syslog ファイル（Windows の場合はイベントログ）と同じエラーメッセージが出力されているかどうかを確認します。

Datareplicator の稼働中にエラー情報ファイルがスワップ又はクローズされると、メッセージ（KFRB00051-I, KFRB00052-I）が出力されます。このメッセージを契機にバックアップを取得しているときは、その内容を確認してください。

3. 反映側 Datareplicator の稼働トレースファイルを `hdstrcredit` コマンドで編集し、解析した結果を保存します。

4. メッセージの内容を基に、エラーに対処します。

エラーの対処については、「[9.2.2 エラーの対処方法](#)」を参照してください。

5. エラーの対処後の処置を実行します。

エラーの原因によっては、障害が発生したプロセスに対応した処置をする必要があります。障害が発生したプロセスごとの処置については、「[9.2.3 エラー対処後の処置](#)」を参照してください。

エラーの内容によっては、反映側 Datareplicator のプロセス又は共用メモリ（Windows 版の場合はメモリマップトファイル）が残っていることがあります。

UNIX 版 Datareplicator の場合は、残ったプロセス又は共用メモリは Datareplicator 再起動時に削除されます。

Windows 版 Datareplicator の場合は、残ったプロセス又はメモリマップトファイルを次の手順で削除してください。

手順	操作
1	タスクマネージャを起動し、「プロセス」タブで表示される Datareplicator プロセスのプロセス ID を取得します。
2	手順 1 で取得したプロセス ID に該当するプロセスを停止します。

手順	操作
	pdkill プロセス ID*
3	インストールディレクトリ%tmp%hde*spool*shm 下にあるメモリマップトファイルを削除します。

注※

Datareplicator Extension を使用している場合は、pdkill コマンドでなく hdskill コマンドを使用してください。

(2) システム管理者への連絡

エラーのうち、システム管理者に連絡する必要がある場合は、次のように処置してください。

1. 次に示すデータを取得します。

- syslog ファイル (Windows の場合はイベントログ) に出力されたメッセージ
- 反映側 Datareplicator のステータス情報
hdsstate コマンドを実行して、ステータス情報を取得します
- エラー情報を示すバイナリファイル
環境変数 HDSPATH で設定したディレクトリに、次の二つのファイルが出力されます。
 - TCPDMP
 - rfc_core

ただし、反映側 Datareplicator を起動したユーザが環境変数 HDSPATH で指定したディレクトリに対して書き込み権限を持っていない場合、rfc_core ファイルは作成されません。

- メモリマップトファイル (Windows 版の場合)
- 反映ステータスファイル
- 反映マスタステータスファイル
- 反映エラー情報ファイル
- 未反映情報ファイル
- 稼働トレースファイル
- 定義ファイル (変更の可能性がある場合)

2. エラーの内容を調査して対処します。

内部矛盾などのエラーの場合には、保守員に連絡してください。

9.2.2 エラーの対処方法

反映側システムのエラーの原因と対処方法を次の表に示します。なお、障害が発生して抽出側 DB と反映側 DB の不整合が発生した場合は、抽出側 Datareplicator 及び反映側 Datareplicator を初期化する必要があります。初期化手順の詳細については、「9.4 障害回復時の初期化手順」を参照してください。

表 9-3 エラーの原因と対処方法

エラーの原因の種類		エラーの対処方法
メモリ不足（ローカルメモリ）		<ul style="list-style-type: none"> 動作中の不要なプロセスを終了してください。 メモリを増設してください。
メモリ不足（共用メモリ）		<ul style="list-style-type: none"> 反映環境定義の defshmsize オペランドの値を調整してください。 共用メモリを使っている、動作中の不要なプロセスを終了してください。 メモリを増設してください。
ファイルディスクリプタ不足		<ul style="list-style-type: none"> 不要なファイルをクローズしてください。 システムパラメタ（ファイルディスクリプタ数の上限値）を調整してください。
ソケット確立不可		システムコールからの詳細コードを基にエラーの原因を取り除いてください。
ポートチェックエラー		詳細コードを基にエラーの発生原因を取り除いてください。
通信回線切断		通信回線を再接続し、抽出側システムを再起動してください。
データ受信シーケンスエラー		抽出側システムの保守員に連絡してください。
ファイル容量不足	反映情報キューファイル	<p>反映情報キューファイルに格納されているすべての更新情報が反映されているかどうかを、hdsstate コマンドで確認してください（COMMUNICATION INFORMATION の write position と REFLECTION INFORMATION の read position が同じになっていれば、すべての更新情報が反映されています）。</p> <p>すべての更新情報が反映されている状態でも、まだ KFRB00005-E（キューファイル容量不足）になるときは、hdsstop -t immediate 又は force コマンドを実行して反映側 Datareplicator を終了してください。その後、ファイル数を増やすか又は容量を拡張してから、hdsstart -i コマンドで再起動してください。このとき、抽出側 Datareplicator は自動的に再送信するので、特に操作は不要です。</p>
	反映ステータスファイル	反映情報キューファイルに格納されているすべての更新情報を反映し終わってから、hdsstop コマンドで反映側 Datareplicator を終了してください。その後、ファイルの容量を拡張してから、hdsstart -i コマンドで再起動してください。
更新情報不正		保守員に連絡してください。
定義情報不正		抽出側システム又は反映側 Datareplicator の定義を修正してください。
媒体不正	反映情報キューファイル	エラー対処後、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化し、抽出側 DB を基に反映側 DB を再作成してください。
	反映ステータスファイル	
	反映マスタステータスファイル	
SQL エラー		SQLCODE を基にエラーの原因を取り除いてください。
更新情報入力コマンド処理中のエラー		<p>抽出処理に障害が発生した場合</p> <p>更新情報入力には影響しないため、処理は続行します。抽出処理は、起動コマンド (hdestart) で再起動できます。</p>

エラーの原因の種類	エラーの対処方法
	<p>既存の反映処理に障害が発生した場合 更新情報入力には影響しないため、処理は続行します。既存の反映処理は再起動コマンド (hdsrfctl) で再起動できます。</p> <p>更新情報入力に障害が発生した場合 更新情報入力用の反映処理は続行します。更新情報入力処理を再起動する場合は、hdssamqin コマンドに -o オプションを指定して実行してください。 -o オプションは障害時にだけ有効です。メッセージを出力できる異常終了時には実行する必要はありません。</p> <p>更新情報入力用反映処理に障害が発生した場合 更新情報入力処理は続行します。反映処理は再起動コマンド (hdsrfctl) で再起動できます。</p>
内部矛盾	保守員に連絡してください。
マシン電源切断	マシンを再起動してください。
バスエラー	保守員に連絡してください。
不正シグナル受信 (sigkill)	<p>UNIX 版 Datareplicator の場合 不正シグナルを受信した場合は、hdsshmclean コマンドを実行してください。</p> <p>Windows 版 Datareplicator の場合 不正シグナルを受信した場合は、hdsemergencystop コマンドを実行してください。</p>

2 相コミット使用時のトランザクション状態の確認

同期点処理要求の実行中に SQL エラーになったときに、同期点処理を 2 相コミット方式にしていると、HiRDB の pdls -d trn -a コマンドを実行して表示される XID から Datareplicator の反映機能で実行しているトランザクションの状態を検知できます。Datareplicator の反映機能のトランザクションの情報として、XID に次の形式で出力されます。

```
XID=aaaaaaaaabbccdddd,eeeeeeeeffgghhhh
```

aaaaaaaa : 実行プロセス名。UOC のときは「hdsuocg」、UOC でないときは「hdssqle」が表示されます。

bb : Datareplicator 識別子 (00~ff)

cc : データ連動識別子 (00~ff)

dddd : 「0000」が表示されます。

eeeeeeee : 反映グループ名が表示されます。

ff : 反映グループ順序番号 (データ連動識別子ごとの反映グループ通番)

gg : SQL プロセス順序番号 (反映グループごとの SQL プロセス通番)

hhhh : 「0000」が表示されます。

2 相コミット使用時のトランザクションの決着

2 相コミット方式で同期点処理をするときは、連携を中止して初期開始をする場合を除いて、コマンドを使ってトランザクションを決着しないでください。

反映側 Datareplicator のトランザクションは、通常は反映側 Datareplicator を再起動した時点で自動的に決着されます。ただし、反映先 HiRDB の異常終了によって反映側 Datareplicator が停止した場合には、トランザクションが決着されないで残るときがあります。このときは、次の手順に従って反映側 Datareplicator を再起動する必要があります。

1. `hdsstop -t force` で反映側 Datareplicator を強制停止させます。
2. 異常終了の原因を取り除いて、反映側 HiRDB を再開させます。
3. `pdls -d trn -a` コマンドを実行して、決着していないトランザクションの一覧を取得し、トランザクションの表示内容に従ってトランザクションを決着させます。その後、反映側 Datareplicator を再起動します。

トランザクションの表示内容とユーザの取る処置を次に示します。

表示内容			ユーザが取る処置
PROGRAM	B-SVID	STATUS	
hdssqle 以外	—	—	Datareplicator 以外のトランザクションのため、トランザクション発生元のプログラムの指示に従ってトランザクションを決着します。
hdssqle	**...*	—	**...* が表示されている TRNGID を持つトランザクションは、すべて反映側 Datareplicator の再起動時に自動的に決着されるため、対策は不要です。
	**...* 以外	FORGETTING(?,w) ?: 任意の文字を示す	この TRNGID を持つトランザクションは、 <code>pdfgt</code> コマンドでトランザクションを決着させます。
		FORGETTING(?,w) 以外	この TRNGID を持つトランザクションの XID を、エラー情報ファイルに出力されている KFRB3072-E メッセージの XID と比較して、次のどちらかの処置を実行します。 同じ XID を持つ KFRB3072-E がある場合： <code>pdcmnt</code> コマンドでトランザクションを決着させます。 同じ XID を持つ KFRB3072-E がない場合： <code>pdrbk</code> コマンドでトランザクションを決着させます。

9.2.3 エラー対処後の処置

障害が発生したプロセスごとの反映側 Datareplicator の処理と、エラー対処後の処置を次の表に示します。

表 9-4 エラー対処後の処置

障害が発生したプロセス	反映側 Datareplicator の処理	ユーザが取る処置
反映マスタプロセス	反映側 Datareplicator を終了します。	エラー対処後、 <code>hdsstart</code> コマンドで反映側 Datareplicator を再起動してください。
反映通信マスタプロセス 受信プロセス	通信及び受信処理をいったん停止します。 抽出側システムから再接続要求が発行されて、通信及び受信処理を自動的に再開します。	—

障害が発生したプロセス	反映側 Datareplicator の処理	ユーザが取る処置
	反映処理は続行します。	
	データの受信を停止した後、通信及び受信処理が回復しません。 反映処理は続行します。	hdsstop コマンドで反映側 Datareplicator をいったん終了してください。その後、エラーの発生原因を取り除き、hdsstart コマンドで反映側 Datareplicator を再起動 ^{*1} してください。
反映定義サーバプロセス 反映プロセス 反映 SQL プロセス	エラーが発生した反映処理を停止します。 受信処理は続行します。	エラー対処後、エラーが発生した反映処理を hdsrftcl コマンドで再起動 ^{*2} してください。
稼働トレース収集プロセス	エラーになるとメッセージが出力されて、稼働トレースは取得されなくなります。 Datareplicator の処理は続行します。	メッセージの内容を参照して、エラーの原因を取り除いてください。
更新情報入力プロセス	hdssamqin コマンド処理を停止します。	hdssamqin コマンドを再実行してください。指定するオプションについては、「7. コマンドの文法」の hdssamqin コマンドの文法を参照してください。

注※1

抽出側 DB と反映側 DB との間に不整合が発生している場合には、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

注※2

抽出側 DB と反映側 DB との間に不整合が発生している場合には、hdsstop コマンドで反映側 Datareplicator を停止してください。その後、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

9.2.4 抽出側システムでエラーが発生したときの対処

抽出側システムでエラーが発生した場合の障害対策について説明します。ここでは、次の項目について説明します。

- 抽出側 DB と反映側 DB との不整合発生時
- 複数の抽出側システムから反映している場合の反映側 DB の再作成

(1) 抽出側 DB と反映側 DB との不整合発生時

抽出側システムでの障害のために、抽出側 DB と反映側 DB との間に不整合が発生した場合は、反映側 DB の再作成が必要です。抽出側と反映側の両方のデータ連動の環境を同期を取って初期化し、抽出側 DB を基に反映側 DB を再作成してください。

(2) 複数の抽出側システムから反映している場合の反映側 DB の再作成

一つの反映側システムで複数の抽出側システムから受信して反映している場合に、ある特定の抽出側システムで障害が発生したときは、反映側システムでは部分初期開始を使用すると、障害が発生した抽出側システムに対する反映処理だけを初期化できます。この場合、障害が発生した抽出側システムに対する反映処理だけを部分初期開始し、障害が発生した抽出側システムの反映対象 DB を再作成してください。

この場合の運用手順を次に示します。

1. 障害が発生した抽出側のデータ連動対象 DB を閉塞して、データ連動対象 DB の更新を抑止します。
2. 抽出側システムが HiRDB の場合は、抽出側 Datareplicator を `hdestop` コマンドで終了します。
抽出側システムがメインフレームの場合は、抽出側の XDM/DS を終了します。
3. 反映側 Datareplicator を `hdsstop` コマンドで終了します。
4. 障害が発生した抽出側システムの反映対象 DB だけを、抽出側 DB を基に再作成します。
この場合、HiRDB Dataextractor を使うと、抽出側 DB から反映側 DB を効率的に作成できます。
5. 抽出側システムのデータ連動の環境を初期化します。
抽出側システムが HiRDB の場合は、抽出側 Datareplicator の環境を初期化します (`hdestart -i` コマンドを実行)。
抽出側システムがメインフレームの場合は、抽出側の XDM/DS の環境を初期化します。
6. 障害が発生した抽出側システムを指定して、反映側 Datareplicator を部分初期開始します (`hdsstart -i -D` コマンドを実行)。
7. 抽出側システムが HiRDB の場合は、抽出側 Datareplicator を `hdestart` コマンドで起動します。
抽出側システムがメインフレームの場合は、抽出側の XDM/DS を起動します。
8. 抽出側のデータ連動対象 DB の閉塞を解除します。

9.2.5 ユーザOWNコーディングの障害時の対処

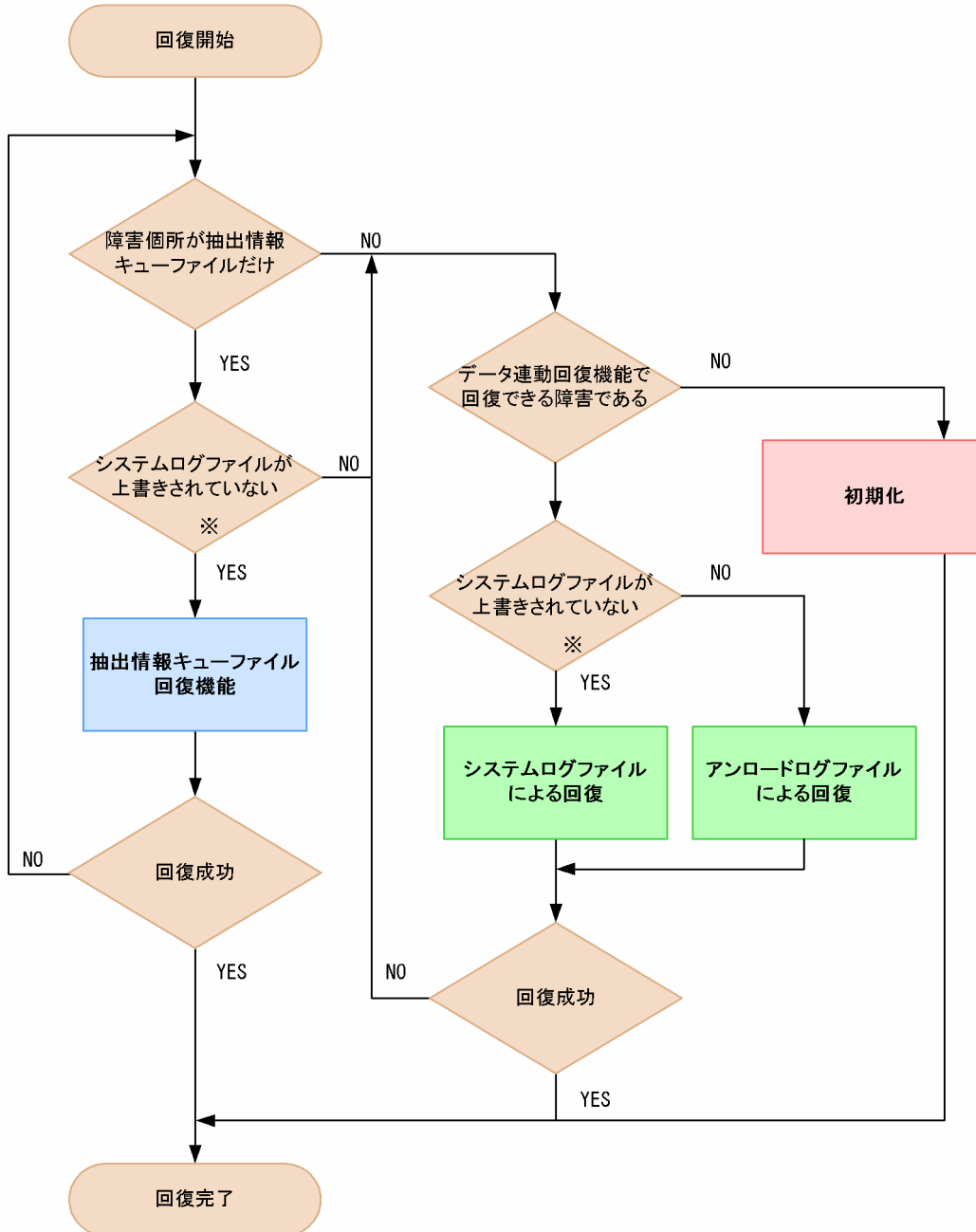
UOC の稼働中に障害が発生した場合、反映側 Datareplicator は UOC が設定した戻り値に対応する処置を実行します。UOC の戻り値については、「8. ユーザOWNコーディング」を参照してください。

UOC が異常終了すると、反映側 Datareplicator はメッセージを出力して異常終了します。

9.3 障害回復方法の選択基準

障害発生時の状況に応じて、実行できる回復方法が異なります。障害回復方法の選択基準を次の図に示します。

図 9-1 回復方法の選択基準



注※

システムログファイルが上書きされているかどうか判断がつかない場合は、一度「YES」に流れ、回復を実行します。実行時にシステムログファイルの上書きによるエラーになったら、「NO」の分岐から再実行します。

抽出情報キューファイルにだけ障害が発生した場合は、抽出情報キューファイル回復機能を使用してください。データ連動回復機能を使用すると、抽出側と反映側の両方を回復する必要がありますが、抽出情報キューファイル回復機能を使用すると、抽出側だけの回復で済みます。

データ連動回復機能で回復できる障害（抽出側システム）を次の表に示します。

表 9-5 データ連動回復機能で回復できる障害（抽出側システム）

障害の発生事象		データ連動回復機能		
		システムログファイルによる回復	アンロードログファイルによる回復	
コマンド誤り	抽出側 Datareplicator を初期化	○	△	
	pdrplstop コマンド実行（データ連動の停止）、又は hdestart -i コマンド実行（初期開始）	○	△	
ディスク障害	抽出情報キューファイル	抽出側 HiRDB のシステムログファイルが上書きされていない	△※	
		抽出側 HiRDB のシステムログファイルが上書きされている	○	
	抽出サーバステータスファイル		○	△
	データ連動用連絡ファイル		○	△
ディスク容量不足	抽出情報キューファイル	○	△	

（凡例）

- ：回復できます。
- △：回復はできますが、この回復方法はお勧めしません。
- ×：回復できません。

注※

抽出情報キューファイル回復機能を使用してください。

データ連動回復機能で回復できる障害（反映側システム）を次の表に示します。

表 9-6 データ連動回復機能で回復できる障害（反映側システム）

障害の発生事象		データ連動回復機能	
		システムログファイルによる回復	アンロードログファイルによる回復
コマンド誤り	反映側 Datareplicator を初期化	○	△
ディスク障害	反映情報キューファイル	○	△
ディスク容量不足	反映ステータスファイル	○	△

(凡例)

○：回復できます。

△：回復はできますが、○の回復方法をお勧めします。

初期化については、「[9.4 障害回復時の初期化手順](#)」を参照してください。

データ連動回復機能の、システムログファイルによる回復については、「[9.5 システムログファイルによるデータ連動回復](#)」を参照してください。また、アンロードログファイルによる回復については、「[9.6 アンロードログファイルによるデータ連動回復](#)」を参照してください。

抽出情報キューファイル回復機能については、「[9.7 抽出情報キューファイル回復機能](#)」を参照してください。

9.4 障害回復時の初期化手順

Datereplicator では、障害が発生して抽出側 DB と反映側 DB の不整合が発生した場合、抽出側 Datereplicator 及び反映側 Datereplicator を初期化する必要があります。この節では、抽出側 DB と反映側 DB の不整合が発生した場合の初期化手順について説明します。

9.4.1 Datereplicator を初期化する必要がある障害

抽出側 DB と反映側 DB の不整合が発生する障害を次に示します。次に示す障害が発生した場合は、抽出側 Datereplicator 及び反映側 Datereplicator を初期化してください。

- 次のファイルでファイル破壊（Disk 破壊、削除など）が発生した

抽出側 Datereplicator のファイル：

- 抽出マスタステータスファイル
- 抽出サーバステータスファイル
- データ連動用連絡ファイル
- 抽出情報キューファイル
- 抽出定義プリプロセスファイル
- 二重化制御ファイル

反映側 Datereplicator のファイル：

- 反映マスタステータスファイル
- 反映ステータスファイル
- 反映情報キューファイル
- 二重化制御ファイル

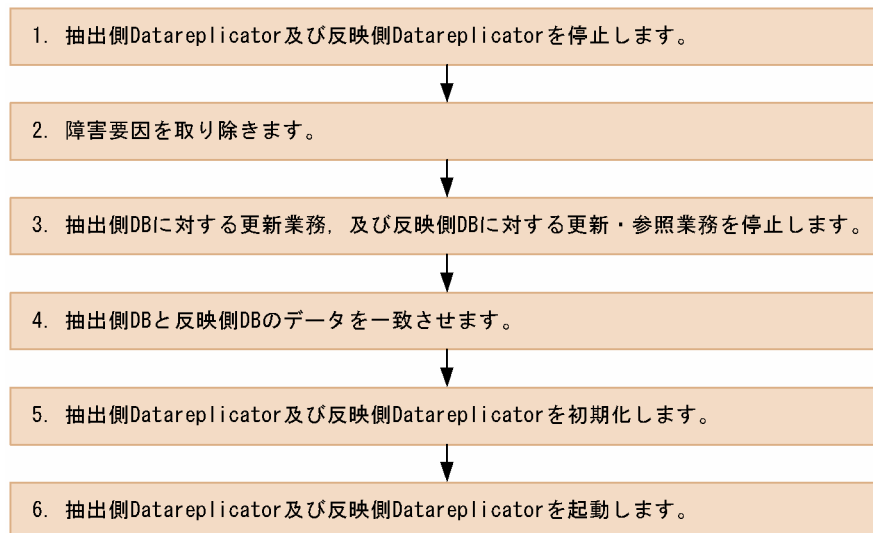
- データの重複や Not Found などの、抽出側 DB と反映側 DB の不整合による SQL エラーが発生した
- 不正データの入力によるコード変換エラーが発生した
- 反映側 Datereplicator では未サポートのデータ型を抽出対象列として反映した
- 未抽出データを含むシステムログファイルの上書きが発生した
- pdrplstop コマンドを実行してデータ連動を中止した
- 反映側、又は抽出側のデータベース（RD エリア、システムログファイル）を再作成、又は初期化した。

9.4.2 障害回復時の Datereplicator の初期化手順

抽出側 DB と反映側 DB の不整合が発生した場合の初期化手順について説明します。

手順の流れを次の図に示します。

図 9-2 障害回復時の初期化手順



注

- 手順 1, 手順 2, 及び手順 3 は順番を入れ替えることができます。
- 手順 4 以降を実施するには、抽出側 DB 及び反映側 DB を起動してください。

障害回復時の初期化手順を図に基づいて説明します。説明の番号は図中の番号に対応しています。

1. 抽出側 Datareplikator 及び反映側 Datareplikator を停止します。

項番	作業内容	実行コマンド	確認項目
1	抽出側 Datareplikator を停止します。	hdestop	—
2	抽出側 Datareplikator の停止を確認します。	hdestate	標準エラー出力に KFRB04411-E メッセージが出力されていることを確認してください。
3	反映側 Datareplikator を停止します。	hdsstop	—
4	反映側 Datareplikator の停止を確認します。	hdsstate	標準エラー出力に KFRB04302-E メッセージが出力されていることを確認してください。

(凡例)

— : 特にありません。

2. 障害要因を取り除きます。

3. 抽出側 DB に対する更新業務、及び反映側 DB に対する更新・参照業務を停止します。

4. 抽出側 DB と反映側 DB のデータを一致させます。

HiRDB Dataextractor などを使用して、抽出対象表から反映対象表へデータの一括複写（データの初期作成）を行い、抽出側 DB と反映側 DB のデータを一致させてください。

5. 抽出側 Datareplikator 及び反映側 Datareplikator を初期化します

項番	作業内容	実行コマンド	確認項目
1	HiRDB とのデータ連携を中止します。	pdrplstop -f	標準出力に KFPS05141-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を初期化します。	hdestart -i (応答メッセージに Y を入力します)	msterrfile1 又は msterrfile2 に KFRB00504-I メッセージが出力されていることを確認してください。
3	抽出定義プリプロセスファイルを作成します。	hdeprep -f 抽出定義ファイル※	標準出力に KFRB04500-I メッセージが出力されていることを確認してください。
4	HiRDB とのデータ連携を開始します。	pdrplstart	標準出力に KFPS05140-I メッセージが出力されていることを確認してください。
5	反映側 Datareplicator を初期化します。	hdsstart -i -q※	標準出力 (Windows の場合はイベントビューア) に KFRB04216-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

6. 抽出側 Datareplicator 及び反映側 Datareplicator を起動します。

項番	作業内容	実行コマンド	確認項目
1	反映側 Datareplicator を起動します。	hdsstart※	errfile1 又は errfile2 に KFRB00100-I メッセージが出力されていることを確認してください。
2	抽出側 Datareplicator を起動します。	hdestart※	msterrfile1 又は msterrfile2 に KFRB00502-I メッセージが出力されていることを確認してください。

注※

必要に応じてオプションを付けてください。

9.5 システムログファイルによるデータ連動回復

データ連動回復機能とは、通常 of データ連動が停止した場合に、障害で消失した未反映の更新情報を再抽出し、その情報を基に抽出情報キューファイルを回復して反映側に送信することで、データ連動の整合性を回復できるようにする機能です。

HiRDB Dataextractor で表全体を回復するには数日掛かってしまう大規模システムで、早急にデータ連動の整合性を回復するときに、データ連動回復機能を使います。

データ連動回復機能には、次の二つの方法があります。

- システムログファイルによる回復
- アンロードログファイルによる回復

ここでは、システムログファイルによるデータ連動回復について説明します。アンロードログファイルによるデータ連動回復については、「[9.6 アンロードログファイルによるデータ連動回復](#)」を参照してください。

注意事項

データ連動回復機能は、通常 of 運用で間違えて実行するとデータ連動不整合を引き起こすおそれがあります。データ連動の障害発生時に限って使うようにしてください。

9.5.1 システムログファイルによるデータ連動回復の概要

システムログファイルによるデータ連動回復には次のメリットがあります。

- 回復手順が容易です。
- 抽出側の業務を停止しないで、次のファイルを回復できます。
 - 抽出マスタステータスファイル
 - データ連動連絡ファイル
 - 抽出サーバステータスファイル
 - 反映マスタステータスファイル
 - 反映情報キューファイル
 - 反映ステータスファイル

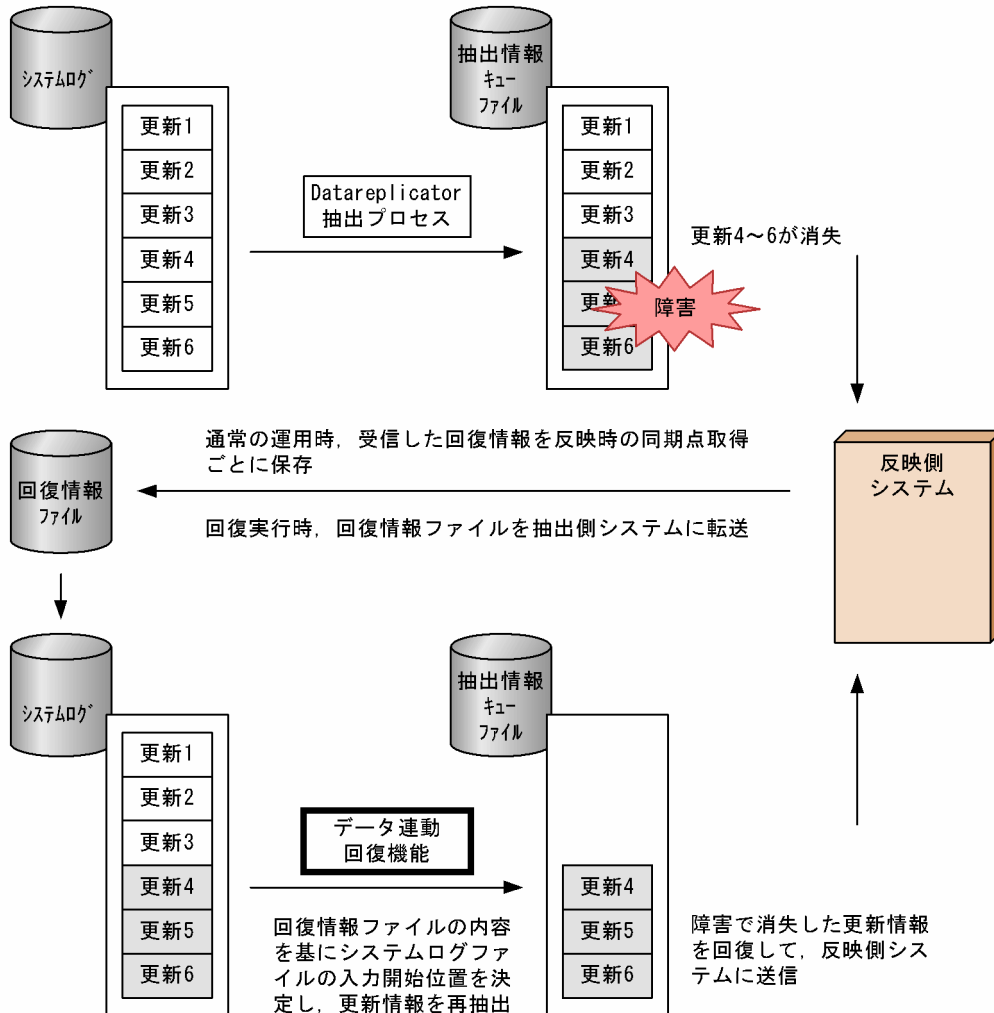
ただし、次の点に注意してください。

- システムログファイルが上書きされていると、この回復方法は使用できません。

- 通常の運用時、更新情報に回復情報が付加されて反映側 Datareplicator に送信されるため、送信データ量が増加します。

抽出情報キューファイルに障害が起こった場合の、システムログファイルによるデータ連動回復の概要を次の図に示します。

図 9-3 システムログファイルによる回復方法の概要



9.5.2 システムログファイルによるデータ連動回復の前提条件

(1) サポートするバージョンと製品の組み合わせ

システムログファイルによるデータ連動回復をできるバージョンと製品の組み合わせを次の表に示します。

表 9-7 システムログファイルによるデータ連動回復を使えるバージョンと製品の組み合わせ

反映側システム		抽出側システム	
		Datareplicator	
		08-03 以前	08-04 以降
Datareplicator	08-03 以前	×	×
	08-04 以降	×	○
XDM/DS		×	×

(凡例)

- ：システムログファイルによるデータ連動回復を使えます。
- ×

(2) 前提条件

システムログファイルによるデータ連動回復を使用するための前提条件を次に示します。

- 抽出側 DBMS が HiRDB であること。
- HiRDB のシステムログファイルが上書きされていないこと。
- 抽出システム定義の recover_info_send オペランドに true を指定していること。
- 回復情報ファイルが残っていること。

(3) システムログファイルによるデータ連動回復を使う前の準備

システムログファイルによるデータ連動回復を使う前には、次の準備をする必要があります。

(a) 回復情報ファイルの作成

抽出システム定義の recover_info_send オペランドに true を指定すると、反映側 Datareplicator の \$HDSPATH 下に、回復情報ファイルが作成されます。このファイルには回復情報が格納されます。回復情報は、回復時にシステムログファイルの入力開始位置を決定するために必要な情報です。回復情報ファイルの内容を次に示します。

表 9-8 回復情報ファイルの内容

項目	内容	備考
ファイル名	rcvrfile_xx_yy	xx：反映側 Datareplicator 識別子 (hdsid) yy：データ連動識別子 (dsidxxx) どちらも 2 けたの 16 進数値文字列で指定します。英字は小文字で指定してください。

項目	内容	備考
作成場所	反映側 Datareplicator の \$HDSPATH 下	—
作成単位	反映側 Datareplicator のデータ連動識別子 (dsidxxx) 単位に 1 個	—
ファイル種別	通常ファイル又はキャラクタ型スペシャルファイル	反映側 Datareplicator が系切り替え環境の場合、共有ディスク装置のキャラクタ型スペシャルファイルを使用してください。
ファイルサイズ	(1 + n)KB	反映処理の方式によって、次のように値が異なります。 <ul style="list-style-type: none"> トランザクション単位反映方式の場合 n : 1 表単位反映方式の場合 n : 反映グループ数 反映グループ単位に回復情報を格納します。
作成タイミング	ポートチェック情報受信時	—
更新タイミング	反映同期点取得時 (反映ステータスファイル更新時)	更新情報は蓄積されないで、上書きされます。

(凡例)

— : 特に備考はありません。

回復情報の送信

回復情報は抽出時のトランザクション単位に反映側 Datareplicator に送信されるため、通常の運用時に、送信データ量、及び反映側 Datareplicator の反映情報キューファイルの使用量が増加します。回復情報を送信する場合に増加するリソースの計算式を次に示します。リソース設計時には、この増加量を考慮してください。

- 送信データの容量

$$(\uparrow \text{TRUN_NUM} / \text{SND_INT} \uparrow) \times 144 \text{ (単位 : バイト)}$$

- 反映情報キューファイルの容量

$$(\uparrow \text{TRUN_NUM} / \text{SND_INT} \uparrow) \times 144 \text{ (単位 : バイト)}$$

- 抽出情報キューファイルの容量

$$144 \times \text{TRN_NUM} \text{ (単位 : バイト)}$$

TRN_NUM : 抽出対象表を更新したトランザクションの数

SND_INT : recover_info_send_interval オペランドに指定した値 (省略時は 1)

回復情報送信間隔

通常の運用時での送信データ量の増加を抑えるため、回復情報を送信する間隔を抽出システム定義の recover_info_send_interval オペランドで指定します。

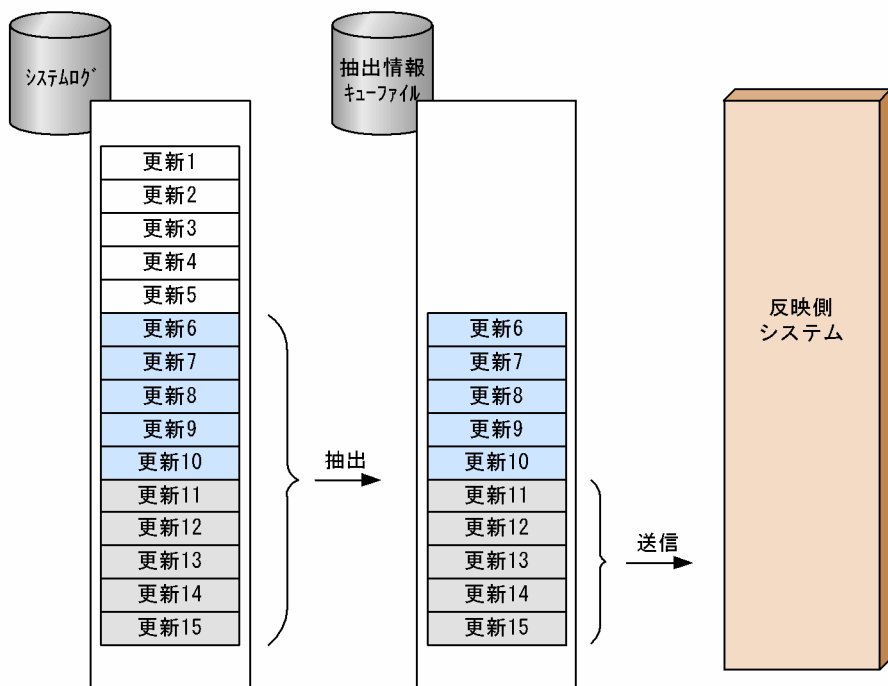
このオペランドには、1~32767の範囲で、1回の送信間隔（送信環境定義の sendintvl 指定時間）中に、回復情報が一度は付加されるような値（1送信間隔中に発生する平均トランザクション数）を指定することをお勧めします。

recover_info_send_interval オペランドに1以外を指定した場合について説明します。

通常の運用時、送信プロセスの起動後、最初に送信するトランザクションの送信データに回復情報を付加し、以降は指定した値のトランザクション数ごとに回復情報を送信します。しかし、回復実行時、回復不要な更新情報も入力及び抽出することになります。

recover_info_send_interval に5を指定した場合の、回復情報の送信例を次の図に示します。

図 9-4 回復情報の送信例



[説明]

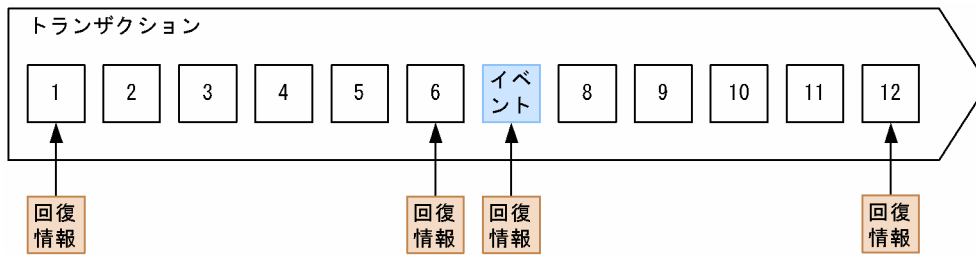
障害で消失し、回復が必要な更新情報は「更新11」から「更新15」ですが、recover_info_send_interval オペランドを指定しているので、「更新6」から入力及び抽出していきます。不要な更新情報は、入力及び抽出はされますが、送信はされません。

注意

回復不要な更新情報が上書きされている場合、回復を実行できません。

なお、このオペランドの指定値に関係なく、イベント送信時は回復情報を送信します。recover_info_send_interval に5を指定した場合の、イベント送信時の回復情報の送信例を、次の図に示します。

図 9-5 イベント送信時の回復情報の送信例



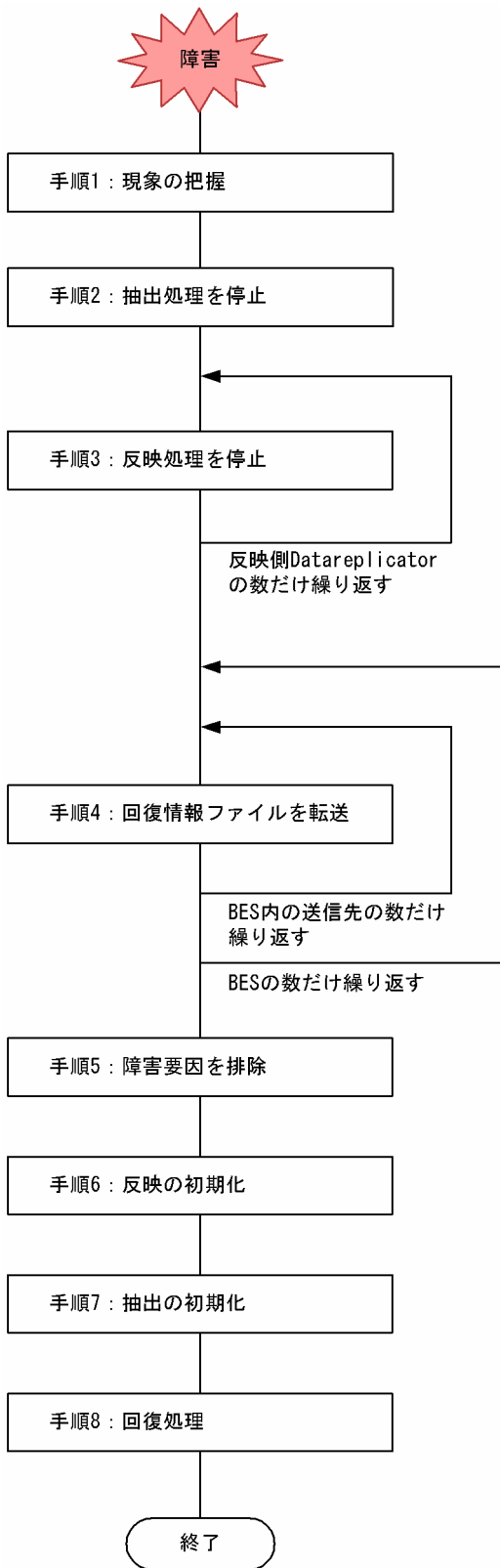
(b) システムログファイルの容量の確認

抽出システム定義の `recover_info_send_interval` オペランドに 1 以外の値を指定した場合、障害発生から回復完了までに生じるシステムログによって、回復に必要な範囲のシステムログが上書きされないように、システムログファイルの容量を十分確保してください。

9.5.3 システムログファイルによる回復手順

システムログファイルによる回復手順の概要を次の図に示します。

図 9-6 システムログファイルによる回復手順の概要



システムログファイルによる回復手順の詳細を次の表に示します。

表 9-9 システムログファイルによる回復手順の詳細

項目	対象	操作内容とコマンド実行例	確認内容
手順 1： 現象の把握	抽出側又は反映側	障害要因を特定します。	—
手順 2： 抽出処理を停止	抽出側	抽出側 Datareplicator を停止します。 コマンド実行例： hdestop	—
手順 3： 反映処理を停止	反映側	反映側 Datareplicator を停止します。 コマンド実行例： hdsstop -t immediate	—
手順 4： 回復情報ファイルを転送	反映側	回復情報ファイルを、抽出側へバイナリモードで転送します。 抽出側システムに対応する回復情報ファイルが複数ある場合、すべての回復情報ファイルを、それぞれ該当する BES（障害の発生していない BES も含む）の \$HDEPATH 下に転送します。 回復情報ファイルが UNIX のキャラクタ型スペシャルファイルの場合、dd コマンドで通常ファイルとしてコピーし、コピーした回復情報ファイルを転送します。 コマンド実行例： [回復情報ファイルの転送] \$ cd \$HDSPATH \$ ftp 抽出側サーバ名 ftp> cd 該当する HDEPATH のパス名 ftp> binary ftp> put 回復情報ファイル名 ftp> bye [回復情報ファイルのコピー] ・ AIX の場合 \$ dd if=\$HDSPATH/回復情報ファイル名 ¥ \$ of=任意のパス/回復情報ファイル名 ¥ \$ bs=1024 skip=1 count=回復情報ファイルサイズ (KB) ・ UNIX の場合 \$ dd if=\$HDSPATH/回復情報ファイル名 ¥ \$ of=任意のパス/回復情報ファイル名 ¥ \$ bs=1024 count=回復情報ファイルサイズ (KB)	—
手順 5： 障害要因を排除	抽出側又は反映側	障害要因を取り除きます。	—
手順 6： 反映の初期化	反映側	反映側 Datareplicator の初期化を実行します。 コマンド実行例： hdsstart -i -f	—
手順 7：	抽出側	抽出側 Datareplicator の初期化を実行します。	—

項目	対象	操作内容とコマンド実行例	確認内容
抽出の初期化		コマンド実行例： pdrplstop -f hdestart - i hdeprep -f 抽出定義ファイル名 pdrplstart	
手順 8： 回復処理	抽出側	データ連動回復で、抽出側 Datareplicator を起動します。 コマンド実行例： hdestart - v	回復を完了したときに出力されるメッセージ (KFRB05041-I) が出力されていることを確認します。*

(凡例)

－：特に確認内容はありません。

注 1

回復実行中に障害が発生した場合の Datareplicator の動作は、通常のデータ連動時に障害が発生した場合と同じです。対処方法については、「9.1 抽出側 Datareplicator の障害時の対処」及び「9.2 反映側 Datareplicator の障害時の対処」を参照してください。

注 2

回復途中の抽出側 Datareplicator を再開する場合、前回稼働時までに回復開始メッセージを出力済みの場合、再開時にメッセージは出力されません。

また、回復情報ファイルが指定されていない場合、又は回復情報ファイルの内容が前回起動時と異なっている場合、回復開始のメッセージ出力時に指定されていた回復情報ファイルの内容で動作します。

注 3

回復情報ファイルに関して次に示すような障害が発生した場合、hdestart - v コマンドによる起動時に KFRB05007-E メッセージが出力され、エラー終了します。この場合は、アンロードログファイルによる回復を実行してください。

1. 回復情報ファイルへの I/O エラー
2. 反映側 Datareplicator が正常終了していない場合の回復情報ファイルを、回復時に抽出側 Datareplicator に割り当てた場合

1.が発生し、エラーメッセージが出力された後も、処理は続行されます。ただし、その後回復情報は更新されません。

2.のように、反映側 Datareplicator が正常終了していない場合（異常終了や電源断など）では、回復情報が正しく更新されていないことがあります。この不完全な回復情報ファイルによって回復を実行した場合、通常のデータ連動時に反映済みであった更新情報を再度送信して反映することがあります。

注※

メッセージ出力後、抽出側 Datareplicator は通常のデータ連動処理を続行し、更新情報を送信します。

9.6 アンロードログファイルによるデータ連動回復

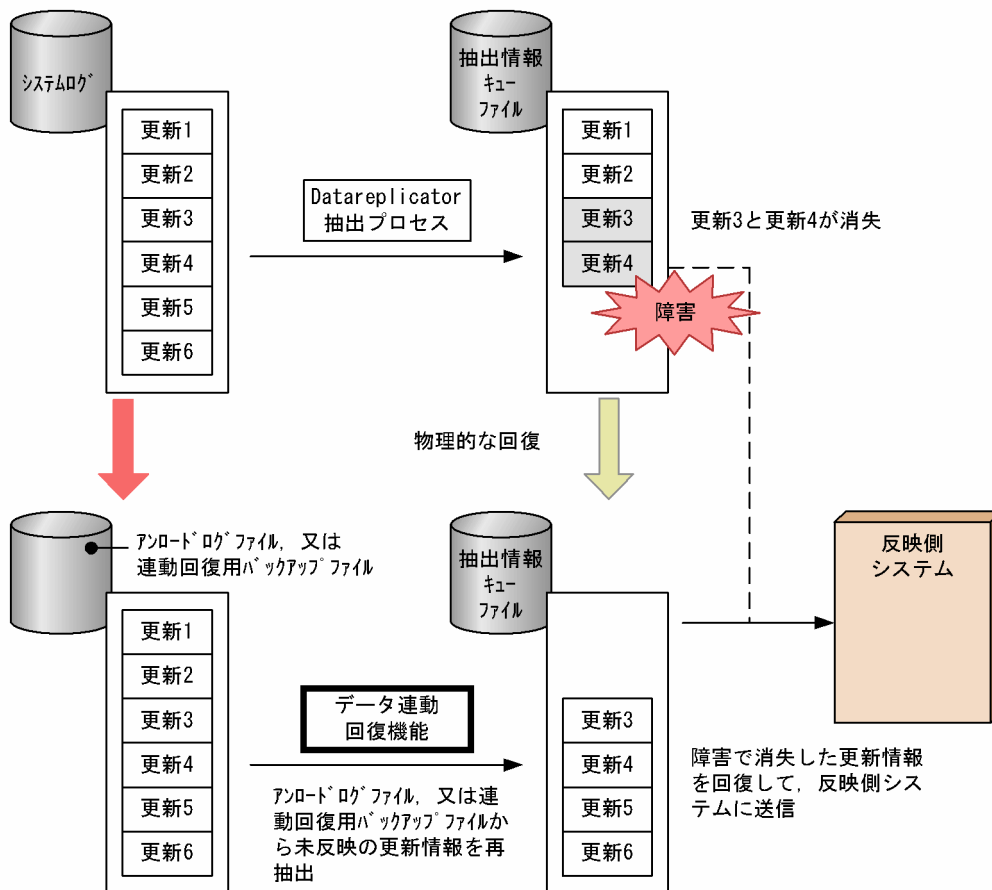
ここでは、アンロードログファイルによるデータ連動回復について説明します。

9.6.1 アンロードログファイルによるデータ連動回復の概要

アンロードログファイルによるデータ連動回復では、アンロードログファイル、又は連動回復用バックアップファイルから障害で消失した未反映の更新情報を再抽出します。連動回復用バックアップファイルとは、アンロードログ情報が格納されている HiRDB ファイルシステム領域内の HiRDB ファイルを、HiRDB ファイルシステムのバックアップコマンド (pdfbkup) で HiRDB ファイル単位に退避したものです。

抽出情報キューファイルに障害が起こった場合の、アンロードログファイルによるデータ連動回復の概要を次の図に示します。

図 9-7 アンロードログファイルによるデータ連動回復の概要



9.6.2 アンロードログファイルによるデータ連動回復の前提条件

(1) サポートするバージョンと製品の組み合わせ

アンロードログファイルによるデータ連動回復を使えるバージョンと製品の組み合わせを次の表に示します。なお、抽出側システムが Datareplicator 以外の製品については、各製品の該当するマニュアルを参照してください。

表 9-10 アンロードログファイルによるデータ連動回復をサポートするバージョンと製品の組み合わせ

反映側システム		抽出側システム	
		Datareplicator	
		06-01 以前	06-01-/A 以降*
Datareplicator	04-00-/O 以降	×	○
	05-03-/D 以降	×	○
	06-01-/A 以降*	×	○
	07-00 以降, 及び 08-00 以降	×	○
	上記以外	×	×
XDM/DS	08-00-A	×	○
	08-02 以降	×	○
	上記以外	×	×

(凡例)

- ：アンロードログファイルによるデータ連動回復を使えます。
- ×

注※

logmrg コマンドの入力対象となる HiRDB のファイルがラージファイルの場合、アンロードログファイルによるデータ連動回復が有効な HiRDB Datareplicator のバージョンは、06-02 以降になります。

(2) 前提条件

HiRDB のアンロードログファイルを使うため、回復対象のログを含むアンロードログファイルがすべてそろっていることが前提条件です。

データ連動に不整合が発生した要因とアンロードログファイルによるデータ連動回復の適用可否（抽出側システム）を、次の表に示します。

表 9-11 データ連動に不整合が発生した要因とアンロードログファイルによるデータ連動回復の適用可否（抽出側システム）

製品	データ連動不整合の発生要因		適用可否	回復範囲	
抽出側 Datareplicator	運用誤り	連動停止指示 (定義変更)	pd_rpl_init_start オペランドを Y から N に変更	適用可	すべての BES
			pd_rpl_hdepath オペランドを無効化	適用不可*	—
			pd_log_rpl_no_standby_file_opr オペランドを continue に設定している状態でのシステムログファイル満杯発生	適用可	満杯が発生した BES が属するユニット内の BES
	連動停止指示 (コマンドの入力間違い)		pd_rplstop コマンド実行	適用可	すべての BES
			pdlogchg -R コマンド実行	適用可	コマンドを実行した BES
		抽出側 Datareplicator を初期化		適用可	すべての BES
		長大なトランザクションを実行したことで、キューファイルが満杯		適用不要 (キューの追加で回復)	—
		抽出対象表の定義変更 (DROP 又は CREATE TABLE) 実行後の抽出定義プリプロセス未実行		適用不可*	—
		ログレスモードでのデータベースの更新		適用不可*	—
	障害	ハードウェア	キューファイルなどのディスク障害	適用可	障害が発生した BES
ソフトウェア		プロセスの停止	適用不要 (リラン実行で回復)	プロセスが停止した BES	

注※

HiRDB Dataextractor, XDM/XT による回復が必要です。

データ連動に不整合が発生した要因とアンロードログファイルによるデータ連動回復の適用可否（反映側システム）を、次の表に示します。

表 9-12 データ連動に不整合が発生した要因とアンロードログファイルによるデータ連動回復の適用可否（反映側システム）

製品	データ連動不整合の発生要因		適用可否	回復範囲
反映側 Datareplicator	運用誤り	反映側 Datareplicator 初期化	適用可	対応する BES
		長大なトランザクションを実行したことで、キューファイルが満杯	適用不要 (初期化によるキューの追加で回復)	—
	障害	ハードウェア	キューファイルなどのディスク障害	適用可

製品	データ連動不整合の発生要因		適用可否	回復範囲	
		ソフトウェア	プロセスの停止	適用不要（リラン実行で回復）	プロセスが停止した BES
反映側 XDM/DS	運用誤り	反映側 XDM/DS 初期化		適用可	対応する BES
		長大なトランザクションを実行したことで、キューファイルが満杯		適用不要（データセットを追加して回復）	—
	障害	ハードウェア	キューファイルなどのディスク障害	適用可	対応する BES
		ソフトウェア	プロセスの停止	適用不要（リラン実行で回復）	—

(3) 制限事項

アンロードログファイルによるデータ連動回復を適用するときの制限事項を次に示します。

- トランザクションについての制限

データ連動回復機能の入力として指定するアンロードログファイル中に回復対象のトランザクションが完全に含まれていないときは、そのトランザクションは回復対象外になります。回復対象トランザクションの開始又は終了が入力アンロードログファイル中に含まれていない場合は、回復を実行する過程で警告ファイル（resfile_BES 名）が出力されるので、内容を確認して、必要であれば入力するアンロードログファイルを追加してください（詳細については、「9.6.3(2)アンロードログファイルの準備」を参照してください）。

HiRDB 側のデータ連動中止範囲を回復対象外にするときは、その範囲を渡るトランザクションは回復対象外になります。

- 反映側での反映方式についての制限

次の条件のどちらかを満たすときは、アンロードログファイルによるデータ連動回復は適用できません。HiRDB Dataextractor や XDM/XT で回復してください。

- マッピングキーを更新している。

- 表単位反映方式で反映処理をしている。ただし、反映側システムで反映抑止機能をサポートしているバージョンは除きます。反映抑止機能の使用については、「9.6.8 反映側システムで表単位反映方式で反映処理する場合の反映抑止方法」を参照してください。

- アンインストールについての制限

アンロードログファイルによるデータ連動回復機能をセットアップした状態で Datareplicator をアンインストールすると、Datareplicator を再インストールしたときにこのデータ連動回復機能をセットアップできなくなります。

Datareplicator のアンインストールは、必ずアンロードログファイルによるデータ連動回復機能をアンセットアップした状態で実行してください。

(4) 注意事項

- マッピングキー更新を行っている場合、反映側 Datareplicator の最後の反映処理中に COMMIT エラー、もしくはシステム障害となった場合、回復する最初のトランザクションで SQL エラー (SQLCODE が 100, もしくは-803) となる場合があります。

この現象が発生した場合、反映環境定義の skip_sqlcode オペランドを使い、処理を継続可能にすることができます。ただし、回復後速やかに、当オペランドを削除してください。

- 反映システム定義に commitment_method = fxa_sqlc を指定している場合、反映側 Datareplicator がエラー停止している、もしくは hdsstop -t force で強制停止している場合、HiRDB に未決着のトランザクションが残る場合があります。HiRDB の pdrbk コマンドでトランザクションを決着させた後、回復作業を実施してください。

9.6.3 アンロードログファイルによるデータ連動回復を使う前の準備

アンロードログファイルによるデータ連動回復を使う前には、次の準備をする必要があります。なお、抽出側 Datareplicator で設定が必要な環境変数は、すべて設定しておいてください。

(1) 環境変数定義ファイルの作成 (障害回復対象のバックエンドサーバのノードに必要)

抽出側 Datareplicator の運用ディレクトリ (\$HDEPATH) 下に、次の環境変数を定義したファイルを hde_toolenv という名称で作成します。なお、環境変数を省略する場合でも、hde_toolenv ファイルは空のファイルとして作成してください。

[TOOL_OUTPUT_DIR=データ連動回復機能の実行結果格納ディレクトリ]

注

環境変数はすべて 1 カラム目から開始してください。

TOOL_OUTPUT_DIR には、データ連動回復機能が出力するファイル (トランザクションリストなど) を格納するためのディレクトリを指定します。この環境変数を省略すると、\$HDEPATH が仮定されます。また、相対パスでディレクトリを指定すると、\$HDEPATH からの相対パスが仮定されます。

(2) アンロードログファイルの準備

データ連動の回復に使う HiRDB のアンロードログファイルを、該当するバックエンドサーバに用意します。現用のシステムログファイルからも、必要に応じて、ログファイルをスワップした後にアンロードログファイルを取得してください。

用意した HiRDB アンロードログファイルは同一ディレクトリ下に格納してください。

(3) 空きディスク容量の確認（障害回復対象のバックエンドサーバのノードに必要）

次に示す項目について、確認してください。

- アンロードログファイルを配置する場所に、アンロードログファイルと同じ容量の空きディスク容量があることを確認してください。また、logmrg コマンドの実行時に指定する出力先ディレクトリ下にアンロードログファイルと同じ容量の空きディスク容量があることを確認してください。
- データ連動回復機能 1 の実行で作成されるトランザクションリスト（抽出 Datareplicator 運用ディレクトリ下に作成）を格納するための空きディスク容量があることを確認してください。トランザクションリスト容量の算出式は次のとおりです。

$$\text{トランザクションリスト容量 (バイト)} = \text{回復対象トランザクション数} \times 64 + 4$$

回復対象トランザクション数は、回復対象とするすべてのアンロードログファイルに対して、次のコマンドを実行した出力結果の総和で求めます。

```
pdlogcat -s -kp アンロードログファイル名 |  
grep "Record header" | wc -l
```

- \$HDEPATH ディレクトリ下に、1MB 程度の空きディスク容量があることを確認してください。

(4) 前回実行時の作成ファイルの削除又は移動（同一バックエンドサーバに対して回復作業を 2 回以上実施する場合だけ）

アンロードログファイルによるデータ連動回復を実行すると、環境変数定義ファイル（hde_toolenv）の TOOL_OUTPUT_DIR に指定したディレクトリ下に、次のファイルが作成されます。

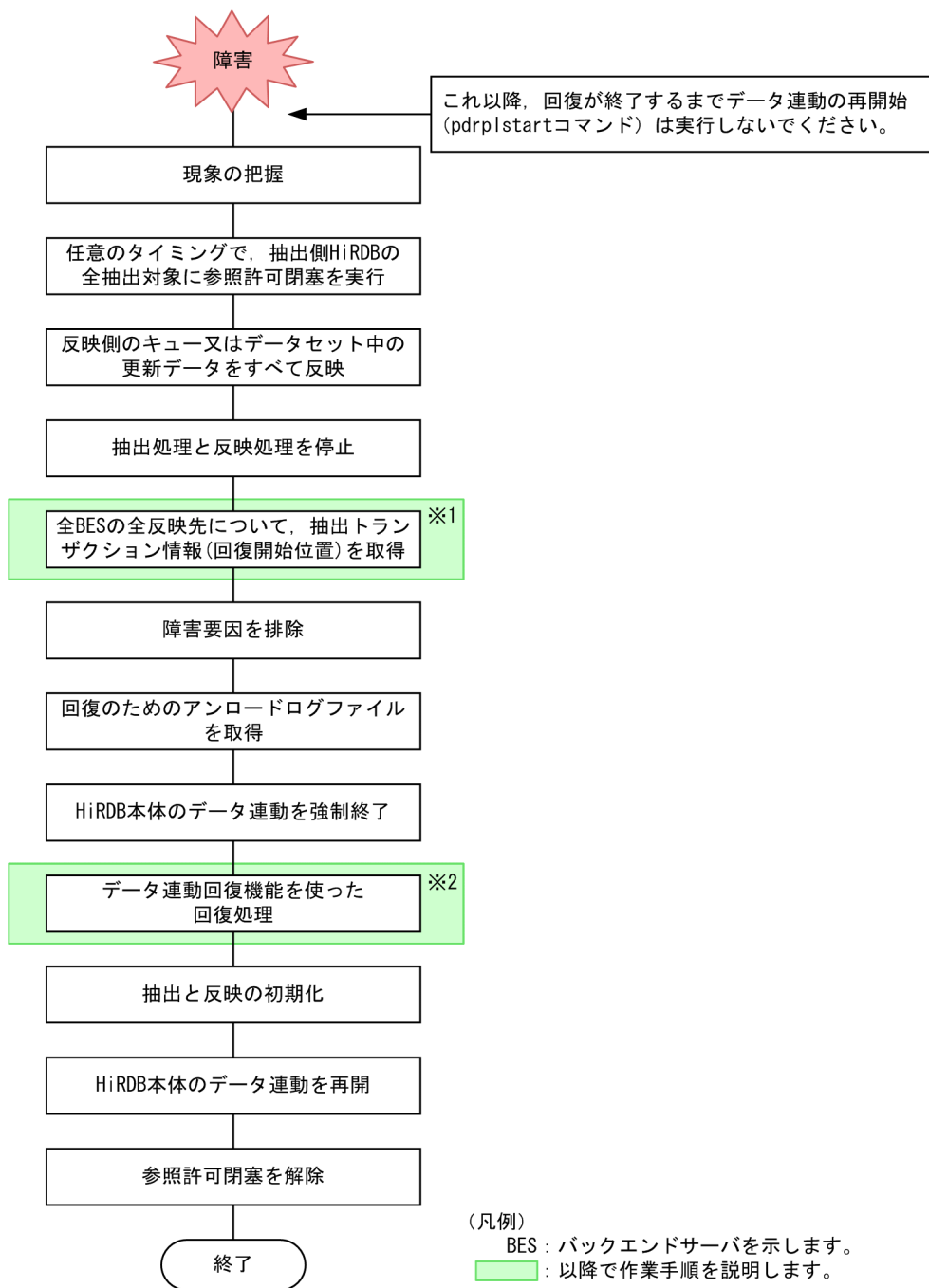
- tranlist_BES 名：回復対象トランザクションリストファイル
- res_file_BES 名：回復区間内未決着トランザクション有無判定ファイル

アンロードログファイルによるデータ連動回復を 2 回以上実行する場合には、前記のファイルが存在しているかどうかを確認してください。ファイルが存在していた場合には削除又は名称を変更してください。

9.6.4 アンロードログファイルによる回復手順の概要

アンロードログファイルによる回復手順の概要を次の図に示します。

図 9-8 アンロードログファイルによる回復手順の概要



注※1

「(1) 抽出トランザクション情報 (回復 開始位置) の取得」を参照してください。

注※2

「(2) データ連動回復機能を使った回復処理」及び「9.6.5 アンロードログファイルによるデータ連動回復手順の詳細」を参照してください。

「図 9-8 アンロードログファイルによる回復手順の概要」中の「全 BES の全反映先について、抽出トランザクション情報 (回復開始位置) を取得」と「データ連動回復機能を使った回復処理」の手順について説明します。

(1) 抽出トランザクション情報（回復開始位置）の取得

抽出トランザクション情報とは、抽出側 HiRDB で発生したトランザクションを一意に識別するための情報です。反映処理が反映をすべて完了した時点で出力する抽出トランザクション情報は、抽出側で実行されたそのトランザクションまで反映が完了したことを示します。

アンロードログファイルによるデータ連動回復では、アンロードログファイル中のどのトランザクションから回復を開始するのかを指定する必要があります。そのとき、反映処理完了時点の抽出トランザクション情報を指定することで、このトランザクションよりも後に完了したトランザクション（反映側に対して未反映のトランザクション）を対象に回復します。

(a) 反映側が Datareplicator のときの、抽出トランザクション情報の取得方法

反映側が Datareplicator のときの、抽出トランザクション情報の取得方法を次に示します。

1. 反映情報キューファイルに格納している更新情報をすべて反映した後、反映側 Datareplicator を停止します。
2. エラー情報ファイルの最後に出力されている KFRB03009-I メッセージを検索して、このメッセージの「Additional Transaction Info =」に出力されている抽出トランザクション情報を取得します。

エラー情報ファイルが上書きされ、KFRB03009-I メッセージの情報がない場合は、次のコマンドを実行して、実行結果から抽出トランザクション情報を取得します。このコマンドは、反映側 Datareplicator を初期化する前に実行してください。

```
hdsrefinfmt -f 反映ステータスファイル名 -l 9 -p 解析結果出力ファイル名
```

注 反映ステータスファイル名を相対パスで指定したときは、\$HDSPATH からの相対パスとして認識します。

コマンドの出力例と回復開始位置の取得方法

反映モードが trn の場合：

```
**** HiRDB Datareplicator Reflection Information <Mon Nov 06 10:55:59 2006> ****
**** Status File Information ****

<Common Information[MINF]>
  Datareplicator-id:f1 Target-DB:HiRDB DB-Locale:SJIS
  Communication protocol:TCP/IP
<Replication Node Information[UJMF]>
  ReplicationNode-id:01 ReplicationType:RECIEVE EventDefinition:1, 2, 5, 3, 4
  :
  <中略>
  :
<Reflect Entry Information[STAT]>
  HeadInformation:GroupNo=000 RunMode=Y
  Read Information:(0, 6144)
  GroupInformation:GroupName=trngroup KeyRangeGroup=No
  ReflectStatusInformation:01, 00, 00, 00, 00, 01
  ReflectDetailInformation:Extract System-id:01 CommitInfo:0(0), 10, 0
                               Locale=(EH, SJ) Source-DB=HiR Format=Sfm
                               TransactionInfo=(3, 2) EventInfo=(0, 0)
  Additional Transaction Info:452348bb0000000000000290
  ReflectSQLExelInformation:SQL-count<ins:1, upd:1, del:0, prg:1, cmt:2, tms:0>
```

「<Reflect Entry Information[STAT]> HeadInformation:GroupNo=000」の「Additional Transaction Info」の情報が回復開始位置となります。

反映モードが tbl の場合：

```

**** HiRDB Datareplicator Reflection Infomation <Fri Jan 25 12:11:22 2002> ****
**** Status File Information ****
<Common Information[MINF]>
Datareplicator-id:f4 Target-DB:HiRDB DB-Locale:SJIS
Communication protocol:TCP/IP
<Replication Node Information[UJMF]>
ReplicationNode-id:c1 ReplicationType:RECIEVE EventDefinition:1,2,5,3,4
:
<中略>
:
<Reflect Entry Information[STAT]>
HeadInformation:GroupNo=000 RunMode=N
Read Information:(0,1024)
GroupInformation:GroupName=trngroup KeyRangeGroup=No
ReflectStatusInformation:10,00,00,00,00,00
ReflectDetailInformation:Extract System-id:04 CommitInfo:0(0),1
                          Locale=(SJ,SJ) Source-DB=HiR Format=Sfm
                          TransactionInfo=(0,0) EventInfo=(0,0)
ReflectSQLexelInformation:SQL-count<ins:0,upd:0,del:0,prg:0,cmt:0,tms:0>
<Reflect Entry Information[STAT]>
HeadInformation:GroupNo=001 RunMode=Y
Read Information:(0,31232)
GroupInformation:GroupName=uoc001 KeyRangeGroup=No
ReflectStatusInformation:01,00,00,00,00,01
ReflectDetailInformation:Extract System-id:04 CommitInfo:3(0),1
                          Locale=(SJ,SJ) Source-DB=HiR Format=Sfm
                          TransactionInfo=(10,6) EventInfo=(1,203)
Additional Transaction Info:3c242b8700000000000000a95
ReflectSQLexelInformation:SQL-count<ins:58,upd:87,del:0,prg:1,cmt:6,tms:0>
<Reflect Entry Information[STAT]>
HeadInformation:GroupNo=002 RunMode=Y
Read Information:(0,23040)
GroupInformation:GroupName=othergrp KeyRangeGroup=No
ReflectStatusInformation:01,00,00,00,00,00
ReflectDetailInformation:Extract System-id:04 CommitInfo:3(4720),1
                          Locale=(SJ,SJ) Source-DB=HiR Format=Sfm
                          TransactionInfo=(8,2) EventInfo=(0,0)
Additional Transaction Info:3c242b8700000000000000094a
ReflectSQLexelInformation:SQL-count<ins:0,upd:0,del:0,prg:2,cmt:2,tms:0>
*****

```

反映状態	回復開始位置の取得方法
「Read Information」がすべて同じ位置の場合	「<Reflect Entry Information[STAT]> HeadInformation:GroupNo=001」の「Additional Transaction Info」の情報が回復開始位置となります。
「Read Information」が異なる位置である場合	「Read Information」が一番遅れている反映グループ*の、「Additional Transaction Info」の情報が回復開始位置となります。

注※

ほかの反映グループについては、反映抑止の制御コードとして SKIP_TYPE_UNTIL を使用します。指定方法については、「3.3.10(1) 反映抑止によるスキップ」を参照してください。

(b) 反映側が XDM/DS のときの、抽出トランザクション情報の取得方法

反映側が XDM/DS のときの、抽出トランザクション情報の取得方法を次に示します。

反映 XDM/DS の状態	取得方法	出力先	抽出トランザクション情報
正常停止又は計画停止	反映 XDM/DS の終了時に出力されるコンソールメッセージ JWD396I の詳細情報を参照して取得	コンソール	JWD396I で出力される「EXTTRN INF : kk...k」で示す kk...k 部分 (16 進形式 24 けた)
	XDM/DS 支援ユーティリティによる処理状態出力実行結果から取得	ABCPRINT	出力結果中の「LAST-EXT-TRN-INF=」以降に示される内容 (16 進形式 24 けた)

(2) データ連動回復機能を使った回復処理

「[図 9-8 アンロードログファイルによる回復手順の概要](#)」に示す回復手順の「データ連動回復機能を使った回復処理」について説明します。

(a) データ連動回復のセットアップ

データ連動回復機能をセットアップします。セットアップを実行すると、Datareplicator の通常の抽出機能がデータ連動回復機能に置き換えられます。このため、次の点に注意してください。

- データ連動回復機能のセットアップは、抽出側 Datareplicator 停止後に実行してください。
- データ連動回復機能のセットアップ中は、Datareplicator の抽出機能は使えません (データ連動回復の実行中に抽出対象とする HiRDB に対して実行した更新は抽出できません)。データ連動回復機能の実行中は、抽出対象表に対して更新はしないでください。

データ連動回復で処理できるのは、**更新情報の抽出まで**です。データ連動回復機能の実行後には、反映側 Datareplicator に再送信して反映までを実行する必要があります。

セットアップでは、次の 2 段階の処理を実行して、データ連動の回復に必要な更新情報を抽出情報キューファイルに抽出します。

• データ連動回復機能 1 (トランザクション検索フェーズ)

logmrg コマンドで作成したアンロードログ分割ファイルを入力して、回復対象トランザクション情報を抽出した後、回復対象トランザクションリストファイルに出力します。

• データ連動回復機能 2 (抽出キュー作成フェーズ)

logmrg コマンドで作成したアンロードログ分割ファイルからデータ連動回復機能 1 で抽出した回復対象トランザクションに該当する更新情報を抽出した後、抽出情報キューファイルに格納します。

複数のバックエンドサーバから抽出している場合は、バックエンドサーバごとに回復処理を繰り返し実行する必要があります。

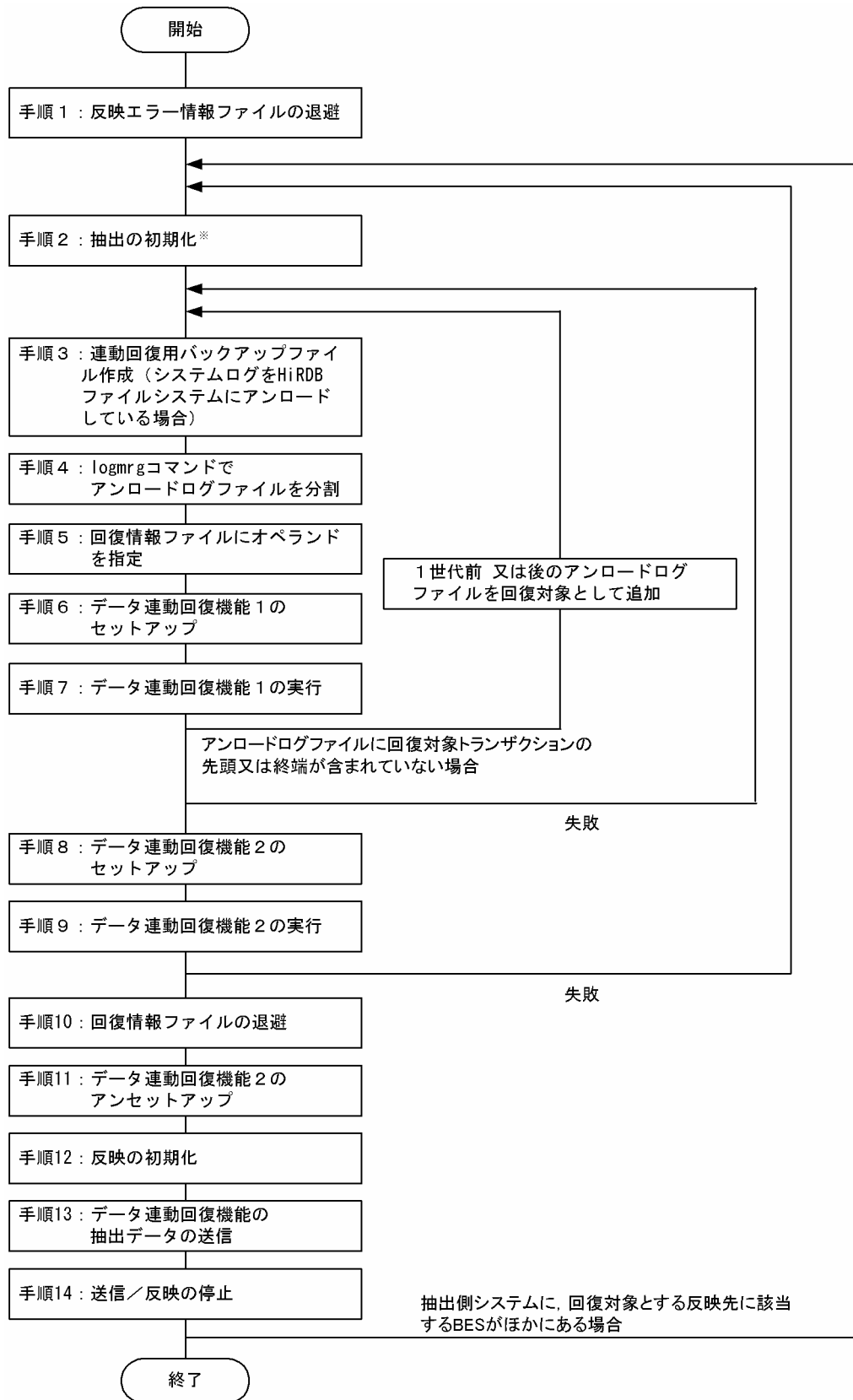
9.6.5 アンロードログファイルによるデータ連動回復手順の詳細

アンロードログファイルによる回復手順の詳細について説明します。

(1) アンロードログファイルによる回復手順

アンロードログファイルによる回復手順を次の図に示します。

図 9-9 アンロードログファイルによる回復手順



注※

抽出の初期化手順には、抽出定義プリプロセスも含まれます。初期化した後に hdeprep コマンドで抽出定義のプリプロセスをしてください。抽出定義のプリプロセスをしないと、回復実行エラーになります。

アンロードログファイルによる回復手順の詳細を次の表に示します。

「操作内容とコマンド実行例」のコマンド実行例を参照するときは、次のことに注意してください。

- コマンドプロンプトの「\$」は Datareplicator 管理者での操作内容を、「#」はスーパーユーザ（Windows 版のときはインストールを実行したユーザ）での操作内容を示します。
- su コマンドは、UNIX のスーパーユーザコマンドです。Windows 版では実行しません。
- vi コマンドは、UNIX の vi エディタを示します。Windows 版のときは、メモ帳（notepad.exe）などの任意のエディタを使います。

表 9-13 アンロードログファイルによる回復手順の詳細

項目	対象	操作内容とコマンド実行例	入力と出力	確認内容
手順 1： 反映エラー情報ファイルの退避	障害回復対象のバックエンドサーバ	反映エラー情報ファイル中に、どこまで反映が完了したかを示す抽出トランザクション情報が格納されているので退避します。 コマンド実行例： \$ cp \$HDSPATH/errfile1 \$HDSPATH/errfile1.sav \$ cp \$HDSPATH/errfile2 \$HDSPATH/errfile2.sav	—	—
手順 2： 抽出の初期化	マネージャノード	抽出の初期化を実行します。抽出の初期化には抽出プリプロセスも含まれます。 コマンド実行例： \$ hdestart -i \$ hdeprep -f extfile	—	—
手順 3： 連動回復用バックアップファイル作成（システムログを HiRDB ファイルシステムにアンロードしている場合）※1	障害回復対象のバックエンドサーバ	世代順に連動回復用バックアップファイルを作成します。 コマンド実行例 \$pdfbkup HiRDB ファイルシステム領域名/HiRDB ファイル名 バックアップファイル名	入力ファイル： • アンロードログファイルを格納した HiRDB ファイルシステム 出力ファイル： • 連動回復用バックアップファイル	—
手順 4： logmrg コマンドでアン	障害回復対象のバック	logmrg コマンドを実行します。	入力ファイル：	\$HDEPATH/caplogparm_BES 名のファイルが作成されている

項目	対象	操作内容とコマンド実行例	入力と出力	確認内容
ロードログファイルを分割※2	エンドサーバ	コマンド実行例： <code>\$ logmrg バックエンドサーバ名</code> 分割ブロック数 出力先ディレクトリ 入力元ディレクトリ {アンロードログファイル名 連動回復用バックアップファイル} [{アンロードログファイル名 連動回復用バックアップファイル}] 注 アンロードログファイル又は連動回復用バックアップファイルは必ず古い順に指定してください。	<ul style="list-style-type: none"> アンロードログファイル，又は連動回復用バックアップファイル 出力ファイル： <ul style="list-style-type: none"> 出力先ディレクトリ/アンロードログファイル名_通番.unlog \$(HDEPATH)/caplogparm_BES 名 	て，かつ作成日付が logmrg コマンドの実行日付と等しいことを確認します。
手順 5： 回復情報ファイルにオペランドを指定※3	障害回復対象のバックエンドサーバ	<code>\$(HDEPATH)/caplogparm_BES</code> 名のファイルに，回復開始位置と HiRDB 側のデータ連動中止区間の回復が必要かどうかのオペランドを追加します。 コマンド実行例： <code>\$ vi \$(HDEPATH)/caplogparm_BES 名</code>	入力ファイル： <ul style="list-style-type: none"> \$(HDEPATH)/caplogparm_BES 名 出力ファイル： <ul style="list-style-type: none"> \$(HDEPATH)/caplogparm_BES 名 	—
手順 6： データ連動回復機能 1 のセットアップ	障害回復対象のバックエンドサーバが存在するノード	データ連動回復機能 1（トランザクション検索フェーズ）のために，プログラムを入れ替えます。 コマンド実行例： <code>\$ su</code> <code># setup_tool1 (Windows Vista, 及び Windows Server 2008 の場合は set_tool1)</code>	—	「Setup for <hdecapture_tool1> complete.」のメッセージが出力されることを確認します。
手順 7： データ連動回復機能 1 の実行	マネージャノード	抽出側 Datareplicator の抽出機能を起動します。このとき，必ず抽出機能だけを起動してください。 コマンド実行例： <code>\$ hdestart -e</code>	入力ファイル： <ul style="list-style-type: none"> \$(HDEPATH)/hde_toolenv 出力先ディレクトリ/アンロードログファイル名_通番.unlog \$(HDEPATH)/caplogparm_BES 名 出力ファイル： <ul style="list-style-type: none"> tranlist_BES 名 res_file_BES 名（不完全なトランザクションが存在するときだけ） 	hdestate コマンドで障害回復対象のバックエンドサーバが存在するノード又は抽出側 Datareplicator 本体が終了状態になるまで監視します。終了後の手順を次に示します。 1. エラー情報ファイルにエラーが出力されていないことを確認します。

項目	対象	操作内容とコマンド実行例	入力と出力	確認内容
				<p>2. 環境変数の TOOL_OUTPUT_ DIR に指定したディ レクトリ下に res_file_BES 名の ファイルが存在しな いことを確認します。</p> <p>3. 障害回復対象のバック エンドサーバが存在 するノードだけが 終了状態になってい るときは hdestop コ マンドを実行し抽出 側 Datareplicator を停止します。</p>
手順 8： データ連動回 復機能 2 の セットアップ ※4※5	障害回復対 象のバック エンドサー バが存在す るノード	<p>データ連動回復機能 2（抽出 キュー作成フェーズ）のために、 プログラムを入れ替えます。</p> <p>コマンド実行例：</p> <pre>\$ su # setup_tool2 (Windows Vista, 及び Windows Server 2008 の場合は set_tool2)</pre>	—	「Setup for <hdecapture_tool2> complete.」のメッセ ージが出力されることを確 認します。
手順 9： データ連動回 復機能 2 の 実行	マネージャ ノード	<p>抽出側 Datareplicator の抽出機 能を起動します。このとき、必ず 抽出機能だけを起動してください。</p> <p>コマンド実行例：</p> <pre>\$ hdestart -e</pre>	<p>入力ファイル：</p> <ul style="list-style-type: none"> • \$HDEPATH/hde_toolenv • 出力先ディレクトリ/アン ロードログファイル名_通 番.unlog • \$HDEPATH/ caplogparm_BES 名 • tranlist_BES 名 <p>出力ファイル：</p> <ul style="list-style-type: none"> • 抽出情報キューファイル（回 復対象データのキュー格納） 	<p>hdestate コマンドで障 害回復対象のバックエン ドサーバが存在するノ ード又は抽出側 Datareplicator 本体が 終了状態になるまで監視 します。終了後の手順を 次に示します。</p> <ol style="list-style-type: none"> 1. エラー情報ファイル にエラーが出力され ていないことを確認 します。 2. 障害回復対象のバック エンドサーバが存在 するノードだけが 終了状態となってい るときは、hdestop コマンドを実行して 抽出側 Datareplicator を停 止します。
手順 10：	障害回復対 象のバック	\$HDEPATH/caplogparm_BES 名のファイルを別名にします。	—	機能のアンインストール を忘れた場合に誤動作し

項目	対象	操作内容とコマンド実行例	入力と出力	確認内容
回復情報ファイルの退避※ 5	エンドサーバ	コマンド実行例： \$ mv \$HDEPATH/ caplogparm_BE S 名 \$HDEPATH/ caplogparm_BE S 名.sav		ないためにする作業です。
手順 11： データ連動回復機能 2 のアンセットアップ	障害回復対象のバックエンドサーバが存在するノード	unsetup_tool コマンドを実行します。 コマンド実行例： \$ su # unsetup_tool (Windows Vista, 及び Windows Server 2008 の場合は unset_tool)	—	「Unsetup for <hdecapture_tool> complete.」のメッセージが出力されることを確認します。
手順 12： 反映の初期化	マネージャノード	反映の初期化をします。 コマンド実行例： \$ hdsstart -i	—	—
手順 13： データ連動回復機能の抽出データの送信	マネージャノード	抽出側 Datareplicator の送信機能を起動します。このとき、必ず送信機能だけを起動してください。 コマンド実行例： \$ hdestart -s \$ hdestate	—	送信機能を起動した後で、hdestate コマンドで送信のキューファイル read オフセットがキューファイル write オフセットに追い付くのを監視します。追い付いたら hdestop コマンドを実行して抽出側 Datareplicator を停止します。
手順 14： 送信/反映の停止	マネージャノード	抽出側送信プロセスと反映プロセスを停止します。 コマンド実行例： \$ hdestop \$ hdsstop	—	—

注※1

HiRDB ファイルの日付を確認するときは、次のコマンドを実行します。

pdlogcat_s -i バックアップ元 HiRDB ファイル | grep "First use"

バックアップ元 HiRDB ファイルとは、連動回復用バックアップファイルの作成元になった HiRDB ファイルのことです。「HiRDB ファイルシステム領域/HiRDB ファイル名」の形式で指定してください。

注※2

アンロードログファイルの日付を確認するときは、次のコマンドを実行します。

pdlogcat_s -i アンロードログファイル名

コマンド実行結果の「First use」の行に出力される日付が、該当するアンロードログファイルに実際に更新ログを格納した日付を示します。logmrg コマンドに指定するアンロードログファイルは、この日付の古い順に指定してください。

注※3

追加するパラメータは、次の内容に従って、ファイルの最後に追加してください。

オペランド名		区分	設定値	説明
RCVR_START	06-02 より前	任意	指定できる RCVR_START オペランドの数は、4096 個までです。 形式 抽出トランザクション情報 • 抽出トランザクション情報 ～〈24 けたの 16 進文字列〉 障害回復の開始位置を示す抽出トランザクション情報を指定します。抽出トランザクション情報の取得方法については、「9.6.4(1)抽出トランザクション情報（回復開始位置）の取得」を参照してください。	このオペランドを省略すると、最初に入力するアンロードログファイルの先頭が障害回復の開始位置と仮定されます。
	06-02 以降	任意	指定できる RCVR_START オペランドの数は、4096 個までです。 形式 {送信先識別子 *} [, 抽出トランザクション情報] • 送信先識別子 抽出システム定義に指定した送信先識別子を指定します。送信先識別子に「*」を指定すると、すべての送信先が対象になります。このオペランドに送信先識別子だけを指定したときは、入力アンロードログファイルの先頭を回復開始位置とします。このオペランドの送信先識別子に「*」を指定したときに、ほかにこのオペランドを指定すると、エラーになります。また、同じ送信先識別子を複数指定すると、エラーになります。 • 抽出トランザクション情報 ～〈24 けたの 16 進文字列〉 抽出トランザクション情報の取得方法については、「9.6.4(1)抽出トランザクション情報（回復開始位置）の取得」を参照してください。	「障害回復対象となる送信先識別子」と「障害回復の回復位置を示す抽出トランザクション情報」を送信先ごとに指定します。このオペランドを省略すると、すべての送信先識別子を回復して、入力アンロードログファイルの先頭を回復開始位置と解釈されます。
RCVR_RPLSTOP		任意	Y : HiRDB 側の連動中止区間を回復対象とします。 N 又は Y 以外の値 : HiRDB 側の連動中止区間を回復対象としません。	障害回復の範囲に HiRDB 側の連動中止区間が存在していた場合にこの区間を回復対象とすることが指定できます。 このオペランドを省略すると、N が仮定されます。

RCVR_START オペランドを指定するときの注意

マッピングキー更新をしていないときだけ、入力アンロードログファイルを先頭から回復できます（マッピングキー更新をしている場合、回復時の二重更新でデータ連動不整合になる可能性があるため）。また、反映側で二重更新によるエラーが発生するため、回復に先立って反映環境定義の skip_sqlcode オペランドにキー不在エラー（100）及びキー重複エラー（-803）の SQLCODE を設定しておく必要があります。

指定例（下線部が追加したパラメタ）

```
INPUT = /HiRDBDS/hirdbb/HDE/work/unldlog2_1.unlog, 10, UNLDLOG
INPUT = /HiRDBDS/hirdbb/HDE/work/unldlog2_2.unlog, 10, UNLDLOG
INPUT = /HiRDBDS/hirdbb/HDE/work/unldlog2_3.unlog, 10, UNLDLOG
BLOCKBUF = 22200
RCVR_START = SND01,3BA6E580000000000000000015
RCVR_RPLSTOP = Y
```

注※4

「res_file_BES 名」のファイルが存在する場合には、ファイルの内容を確認して、下記に従って対処してください（ファイルはテキストファイルです）。

出力内容	説明	対処方法
「N.G(start nothing)」	アンロードログにトランザクションの開始がないデータが存在します。	もう 1 世代前のアンロードログファイルを追加して、手順 3 から再実行してください。 再実行するときには res_file_BES 名のファイルを削除又は別名で保存してください。
「N.G (end nothing)」	アンロードログにトランザクションの終了がないデータが存在します。	抽出対象表を格納するすべての RD エリアに対する参照閉塞が有効となっているかどうか確認してください。 参照閉塞が有効となっている場合： この出力を無視して、以降の回復手順を続行してください。 参照閉塞が無効となっている場合： 参照閉塞を有効とした後、現時点のカレントまでのアンロードログファイルを追加して、手順 3 から再実行してください。 再実行するときには res_file_BES 名のファイルを削除又は別名で保存してください。
「N.G (rplstop found)」 「N.G (start nothing,rplstop)」 「N.G (end nothing,rplstop)」	HiRDB 側の連動中止区間にわたるトランザクションが存在しません。	この出力は HiRDB 側の連動中止区間が回復対象外であることを示します。HiRDB 側の連動中止区間を回復対象外とすることで問題なければ、この出力を無視して以降の回復手順を続行してください。 HiRDB 側の連動中止区間を回復対象とする場合は RCVR_RPLSTOP=Y を指定した後、手順 6 から再実行してください。

注※5

障害回復対象のバックエンドサーバが存在するノードにそのほかのバックエンドサーバが存在する構成の場合、これらのバックエンドサーバに対しても回復ツールが実行されることとなります。したがって、障害回復対象以外のバックエンドサーバについては回復に必要なファイルが存在しないなどのエラー（上記の※3を参照）となりますが、これらのエラーは無視してください。

(2) アンロードログファイルによるデータ連動回復の実行でエラーとなった場合の対処方法

アンロードログファイルによるデータ連動回復の実行でエラーとなった場合、アンロードログファイルによるデータ連動回復固有のエラーはすべて KFRB05009-E メッセージのエラー情報ファイルに出力されます。メッセージ本文の内容は無視して「function:」の部分に出力される情報でエラーの内容を特定してください。「function:」部分の説明を次の表に示します。

表 9-14 KFRB05009-E メッセージの「function:」部分の内容

「function:」部分の内容	説明	対処方法
analyze error	\$HDEPATH/caplogparm_BES 名ファイルの解析でエラーが発生しました。	\$HDEPATH/caplogparm_BES 名が存在するかどうか確認し、存在しない場合は手順 3※から再実行してください。ファイルが存在するときは、該当するファイル内に追加したオペランドの指定が間違っています。オペランドを修正した後、手順 7※から再実行してください。
env open_error	環境変数定義ファイル (hde_toolenv) のオープンでエラーが発生しました。	環境変数定義ファイルが \$ HDEPATH 下に作成されているかどうかを確認し、作成されていない場合は作成後に手順 7※から再実行してください。
intrn error	回復ツールの初期化処理でエラーが発生しました。	このエラーはメモリ不足のときだけ発生します。ほかの稼働中のプログラムを停止させて空きメモリを増やした後、手順 7※から再実行してください。
open_error	アンロードログ分割ファイルのオープン処理でエラーが発生しました。	logmrg コマンドによって作成されるアンロードログ分割ファイルが正しく作成されていない可能性があります。手順 3※から再実行してください。
read_error	アンロードログ分割ファイルの読み込み処理でエラーが発生しました。	logmrg コマンドによって作成されるアンロードログ分割ファイルが正しく作成されていない可能性があります。手順 3※から再実行してください。
trnout error	トランザクションリストの出力処理でエラーが発生しました。	トランザクションリストの出力先ディレクトリの容量が不足した可能性があります。出力先ディレクトリを変更後、手順 7※から再実行してください。
trnget error	トランザクションリストの入力処理でエラーが発生しました。	トランザクションリストが正しく作成されていません。手順 11※を実行後、手順 6※から再実行してください。
blk_invalid	アンロードログファイル中に不正なログブロックを検出しました。	内部矛盾エラーです。開発元に連絡してください。

「function:」部分の内容	説明	対処方法
start point err	抽出トランザクション情報の指定が間違っています。	RCVR_START オペランドで指定された抽出トランザクション情報が不正又は入力に指定したアンロードログファイルが不足しています。 抽出トランザクション情報指定不正の場合は修正後に手順 7*から、アンロードログファイル指定不足の場合は入力とするアンロードログファイルを追加して手順 3*から再実行してください。
seq_invalid	アンロードログファイル中のログブロック出現順序が間違っています。	logmrg コマンド実行時に指定したアンロードログファイル、又は連動回復用バックアップファイルの順序が不正（ログ出力日付順に昇順となっていない）である可能性があります。入力とするアンロードログファイル、又は連動回復用バックアップファイルの指定順序を確認して、手順 3*から再実行してください。
param len error	環境変数の TOOL_OUTPUT_DIR に指定したディレクトリパス名の長さが間違っています。	指定値を修正後、手順 7*から再実行してください。
remain trnentry	データ連動回復機能 1 で作成したトランザクションリストと、データ連動回復機能 2 で回復したトランザクション数が不一致です。	内部矛盾エラーです。開発元に連絡してください。

注※

「(1)アンロードログファイルによる回復手順」を参照してください。

9.6.6 データ連動回復機能のコマンド

アンロードログファイルによるデータ連動回復では、次のコマンドを使います。

コマンド名	説明
logmrg	回復対象のアンロードログファイル又は連動回復用バックアップファイルを入力して、障害回復ツールが入力できる形式及びサイズのファイルに分割するコマンドです。
setup_tool1*	データ連動回復機能 1 を使える状態にセットアップするコマンドです。
setup_tool2*	データ連動回復機能 2 を使える状態にセットアップするコマンドです。
unsetup_tool*	セットアップ中のデータ連動回復機能を無効化して、Datareplicator の抽出機能を使える状態に回復するコマンドです。

注※

Windows のバージョンによって、コマンド名が異なります。次の表に示すコマンド名で実行してください。

Windows XP 及び Windows Server 2003 の場合	Windows Vista 以降及び Windows Server 2008 以降の Windows の場合
setup_tool1	set_tool1
setup_tool2	set_tool2
unsetup_tool	unset_tool

(1) logmrg コマンド

logmrg コマンドは、ログインユーザがアクセス権を持つディレクトリで実行してください。ログインユーザにアクセス権がないディレクトリで実行すると、アクセス権エラーになります。

形式

```
logmrg HiRDBサーバ名
      分割ブロック数
      出力先ディレクトリ
      入力元ディレクトリ
      {HiRDBアンロードログファイル名 | 連動回復用バックアップファイル名}
      [ {HiRDBアンロードログファイル名 | 連動回復用バックアップファイル名} ]
```

オプション

- HiRDB サーバ名

HiRDB サーバ名を指定します。

- 分割ブロック数

HiRDB アンロードログファイルを分割する単位をブロック数で指定します。ここに指定する値とアンロードログ分割ファイルの最大サイズとの関連を次に示します。

$$\text{アンロードログ分割ファイル最大サイズ (単位: バイト)} \\ = \text{回復対象HiRDBのpd_log_max_data_sizeオペランド値} \times \text{分割ブロック数}$$

このオペランドには、100～1000 の範囲で指定することをお勧めします。

なお、アンロードログ分割ファイルのサイズは、2GB-1 を超えないようにしてください。

- 出力先ディレクトリ

logmrg コマンドの出力ファイル（アンロードログ分割ファイルなど）の格納ディレクトリを絶対パスで指定します。指定できる名称長は 128 バイト以内です。空白を含む名称のディレクトリパスは指定しないでください。

- 入力元ディレクトリ

入力する HiRDB アンロードログファイル群の格納ディレクトリを指定します。指定できる名称長は 128 バイト以内です。空白を含む名称のディレクトリパスは指定しないでください。

Datareplicator のバージョンが 07-02 以降では、入力するアンロードログファイル又は連動回復用バックアップファイルとして、ラージファイルを指定できます。ただし、出力するアンロードログ分割ファイルは、ラージファイルではないファイルである必要があります。

• HiRDB アンロードログファイル名 | 連動回復用バックアップファイル名

～ 〈1～32 文字の記号名称〉

入力する HiRDB アンロードログファイル名又は連動回復用バックアップファイルを指定します。ファイル名は世代順に指定してください。なお、アンロードログ分割ファイルは、通常ファイル、又はラージファイルではないファイルである必要があるため、注意してください。

アンロードログファイルの日付を確認するには、次のコマンドを実行します。

`pdlogcat_s -i アンロードログファイル名 | grep "First use"`

連動回復用バックアップファイルの日付を確認するには、次のコマンドを実行します。

`pdlogcat_s -i バックアップ元 HiRDB ファイル | grep "First use"`

バックアップ元 HiRDB ファイルとは、連動回復用バックアップファイルの作成元になった HiRDB ファイルのことです。「HiRDB ファイルシステム領域/HiRDB ファイル名」の形式で指定してください。

メッセージ

メッセージ	対処方法
[aa...aa] File(ファイル名) open error, errno=xx...xx	errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
[aa...a] File size after dividing exceeds 2 GB, filesize = bb...b	アンロードログ分割ファイルのサイズが 2GB を超えています。logmrg コマンド引数の分割ブロック数に、より小さい値を指定して再実行してください。
[aa...aa] Interface error occurred	保守員に連絡してください。
[aa...aa] Invalid file format	コマンドに指定する入力ファイルに、HiRDB のアンロードログファイル、又は連動回復用バックアップファイルを指定して、再実行してください。
[aa...aa] Lseek error, errno= xx...xx	errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
[aa...aa] Malloc error, size= xx...xx	メモリを追加した後で、コマンドを再実行してください。
[aa...aa] Number of HiRDB files in backup file exceeds x	指定された連動回復用バックアップファイル内に x 個を超える HiRDB ファイル情報が存在しています。 HiRDB ファイル単位に作成した連動回復用バックアップファイルを指定してください。
[aa...aa] Read error, errno= xx...xx	errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
[aa...aa] Stat error, errno= xx...xx	errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
[aa...aa]system call error. errno=xx...xx	システムコール名及び errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
[aa...aa] Write error, errno= xx...xx	errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。

メッセージ	対処方法
File open error. errno=xx...xx file=yy...yy	errno に従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
HDEPATH env not set.	HDEPATH 環境変数を設定した後で、コマンドを再実行してください。
HDEPATH env length invalid	HDEPATH 環境変数の設定を修正した後で、コマンドを再実行してください。
Invalid command argument. Usage : logmrg <server name> <block numbers> <output directory> <input directory> <unload log file name>...	オペランドの指定を修正した後で、コマンドを再実行してください。
Invalid logcut output file.	このメッセージの出力以前に出力しているメッセージに従ってエラー要因を特定し排除した後で、コマンドを再実行してください。
Specified output directory is not full path name.	オペランドの指定を修正した後で、コマンドを再実行してください。

(2) setup_tool1 コマンド

setup_tool1 コマンドは、データ連動回復機能 1 の環境をセットアップするコマンドです。セットアップでは、通常運用時に使用するプログラムをリネームして、データ連動回復機能 1 をコピーします。

形式

```
setup_tool1 (Windows Vista, 及びWindows Server 2008の場合はset_tool1)
```

メッセージ

メッセージ	対処方法
Setup for <hdecapture_tool1> complete.	データ連動回復機能 1 のセットアップが完了しました。以降の回復作業を継続してください。
Setup for <hdecapture_tool1> failed.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
<hdecapture> backup failed.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
Setup for <hdecapture_tool1> already execute.	データ連動回復機能 1 はセットアップ済みです。
HiRDB Datareplicator is not installed.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
Please execute by root user.	管理者権限（スーパーユーザ）でコマンドを再実行してください。

(3) setup_tool2 コマンド

setup_tool2 コマンドは、データ連動回復機能 2 の環境をセットアップします。セットアップでは、データ連動回復機能 1 をリネームして、データ連動回復機能 2 をコピーします。

形式

```
setup_tool2 (Windows Vista, 及びWindows Server 2008の場合はset_tool2)
```

メッセージ

メッセージ	対処方法
Setup for <hdecapture_tool2> complete.	データ連動回復機能 2 のセットアップが完了しました。以降の回復作業を継続してください。
Setup for <hdecapture_tool2> failed.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
<hdecapture> backup failed.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
Not setup for <hdecapture_tool1>.	データ連動回復機能 1 のセットアップ完了後に、コマンドを実行してください。
HiRDB Datareplicator is not installed.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
Please execute by root user.	管理者権限（スーパーユーザ）でコマンドを再実行してください。

(4) unsetup_tool コマンド

unsetup_tool コマンドは、データ連動回復機能 1, 及びデータ連動回復機能 2 の環境をアンセットアップします。アンセットアップでは、セットアップ済みの機能をリネームして、通常運用時に使用するプログラムをコピーして戻します。

形式

```
unsetup_tool (Windows Vista, 及びWindows Server 2008の場合はunset_tool)
```

メッセージ

メッセージ	対処方法
Unsetup for <hdecapture_tool> complete.	データ連動回復機能のアンセットアップが完了しました。
Unsetup for <hdecapture_tool> failed.Please re-install HiRDB Datareplicator.	Datareplicator のロードモジュールが不正な状態である可能性があります。Datareplicator を再インストールした後で、コマンドを再実行してください。
Not setup for <hdecapture_tool>.	データ連動回復機能がセットアップされていません。

メッセージ	対処方法
Please execute by root user.	管理者権限（スーパーユーザ）で、コマンドを再実行してください。

9.6.7 データ連動回復機能を実行した後の作業

(1) 回復完了後の作業

アンロードログファイルによるデータ連動回復での回復作業を完了した後は、次のファイルを削除してください。

- アンロードログ分割ファイル
logmrg コマンドを実行したときに作成される「出力先ディレクトリ/入力アンロードログファイル名_通番.unlog」と「\$HDEPATH/caplogparm_BES 名（別名にしてあるときはそのファイル）」を削除してください。
- データ連動回復機能関連ファイル
抽出側 Datareplicator 運用ディレクトリ（\$HDEPATH）下、及び環境変数定義ファイル（hde_toolenv）の環境変数 TOOL_OUTPUT_DIR に指定したディレクトリにある、次のファイルを削除してください。
 - tranlist_BES 名, res_file_BES 名, hde_toolenv

(2) データ連動回復の確認

データ連動回復が正しく実行されたかどうかを、下記の項目で確認してください。

- 抽出表と反映表のデータ件数
抽出表と反映表のデータ件数をそれぞれ取得して、両者に差異がないことを確認してください。
- 抽出表と反映表のデータ内容
抽出表と反映表から回復対象範囲のデータをサンプリングして検索して、両者のデータ内容に差異がないことを確認してください。

9.6.8 反映側システムで表単位反映方式で反映処理する場合の反映抑止方法

反映側システムでは、表単位反映方式で反映処理を行なっている場合、反映抑止によるエラーのスキップを使用することでデータ連動回復できます。

反映抑止によるエラーのスキップは次の場合に使用できます。

- 回復中に、トランザクション単位反映イベント、又は表単位反映イベントを使用していない場合
- 抽出定義の変更（hdeprep コマンドの再実行を含む）を行っていない場合

データ連動回復する場合、反映抑止の制御コードには SKIP_TYPE_UNTIL を使用します。反映抑止によるエラーのスキップの詳細については、「3.3.10(1) 反映抑止によるスキップ」を参照してください。

(1) 反映抑止対象リストファイルの作成手順

反映抑止対象リストファイルの作成手順を次に示します。

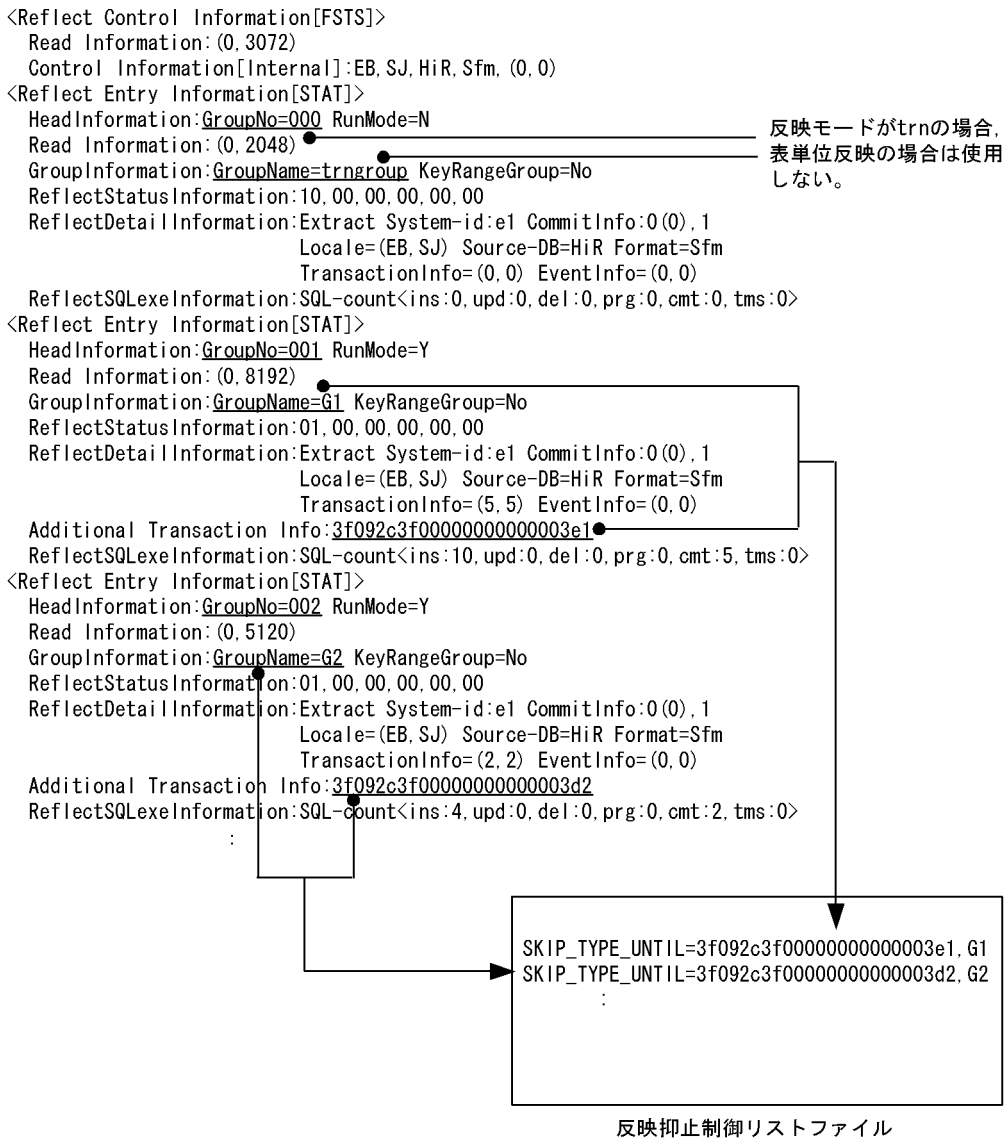
反映情報キューファイルに格納している更新情報をすべて反映したことを確認します。

1. 反映側 Datareplicator を停止します。
2. hdsrefinfmt コマンドを実行して、実行結果から各グループごとの抽出トランザクション情報を取得します。

hdsrefinfmt コマンドは、反映グループ名の指定をしないで、次のように実行します。

```
hdsrefinfmt -f 反映ステータスファイル名 -l 9 -p 解析結果出力ファイル名
```

3. 抽出トランザクション情報を基に、\$HDSPATH 下に反映抑止対象リストファイルを作成します。
hdsrefinfmt コマンドの出力と反映抑止制御リストファイルとの対応は次のようになります。この例では、反映モードは trn とします。



反映抑止リストファイルを作成するとき、パラメタ RCVR_START で指定した抽出トランザクション情報と同じ抽出トランザクション情報を持つグループについては、反映抑止制御リストファイルに記述しないでください。

4. 反映側 Datareplicator を起動します。

(2) 注意事項

- SKIP_TYPE_UNTIL を使用して反映抑止を行った後、反映プロセスを停止したときには、この指定は必ず削除してください。削除しないで反映プロセスを再起動すると、検知するすべての更新情報の反映が抑止されます。
- SKIP_TYPE_UNTIL 指定時には、イベントによる反映モードの切り替え、及び抽出側の定義変更を行なわないでください。反映プロセスが自動停止、起動を行うため、処理済みの SKIP_TYPE_UNTIL が再度有効となり、不当に反映抑止を行なう可能性があります。
- SKIP_TYPE_UNTIL で抑止された更新情報を、障害回復機能を使って回復することはできません。HiRDB Dataextractor や XDM/XT による回復が必要となります。

9.7 抽出情報キューファイル回復機能

ここでは、抽出情報キューファイル回復機能について説明します。

9.7.1 抽出情報キューファイル回復機能の概要

抽出情報キューファイル回復機能とは、ディスク障害などで抽出情報キューファイルだけに障害が発生した場合に、抽出サーバステータスファイルの情報を基に、抽出情報キューファイルを再作成する機能です。

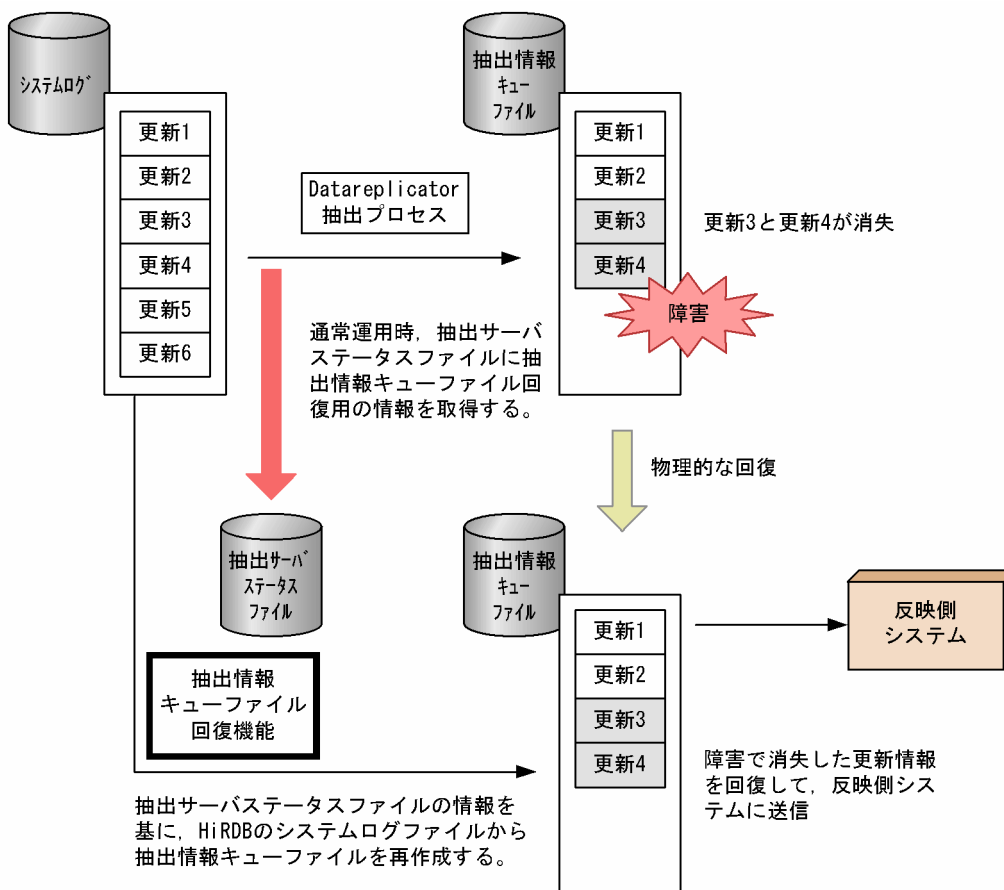
抽出情報キューファイル回復機能には次のメリットがあります。

- 抽出側の業務（更新業務）を停止する必要がありません。
- 回復手順が容易です。

ただし、システムログファイルが上書きされていると、この機能は使用できません。

抽出情報キューファイル回復機能の概要を次の図に示します。

図 9-10 抽出情報キューファイル回復機能の概要



9.7.2 抽出情報キューファイル回復機能の前提条件

抽出情報キューファイル回復機能を使用するための条件について説明します。

(1) 前提条件

次の条件を満たしている必要があります。

- 抽出側 DB が HiRDB であること
- 次のファイルに障害が発生していないこと、またファイルを初期化していないこと
 - 抽出マスタステータスファイル
 - 抽出サーバステータスファイル
 - データ連動用連絡ファイル
- HiRDB のシステムログファイルが上書きされていないこと
- 抽出情報キューファイルが満杯になっていないこと
- 障害が発生する前の抽出処理中に、hdeprep コマンドを実行していないこと
- 回復前の稼働時に、HiRDB の pdrplstop コマンドを実行していないこと
- 抽出システム定義の syncterm オペランドに true を指定していないこと
- Datareplicator ファイルシステム領域を使用していないこと

(2) 抽出側 Datareplicator の初期開始 (バージョンアップ時)

抽出側 Datareplicator を 08-01 より前のバージョンから 08-01 以降にバージョンアップした場合は、抽出側 Datareplicator を初期開始してください※。

08-01 より前のバージョンの抽出情報キューファイルは、抽出情報キューファイル回復機能では回復できません (データ連動回復機能で回復してください)。08-01 より前のバージョンの抽出情報キューファイルが存在する場合は、抽出側 Datareplicator の起動時に、KFRB05037-W メッセージが出力されます。

注※

抽出情報キューファイル回復機能を使用しないシステムの場合は、初期開始は必要ありません。

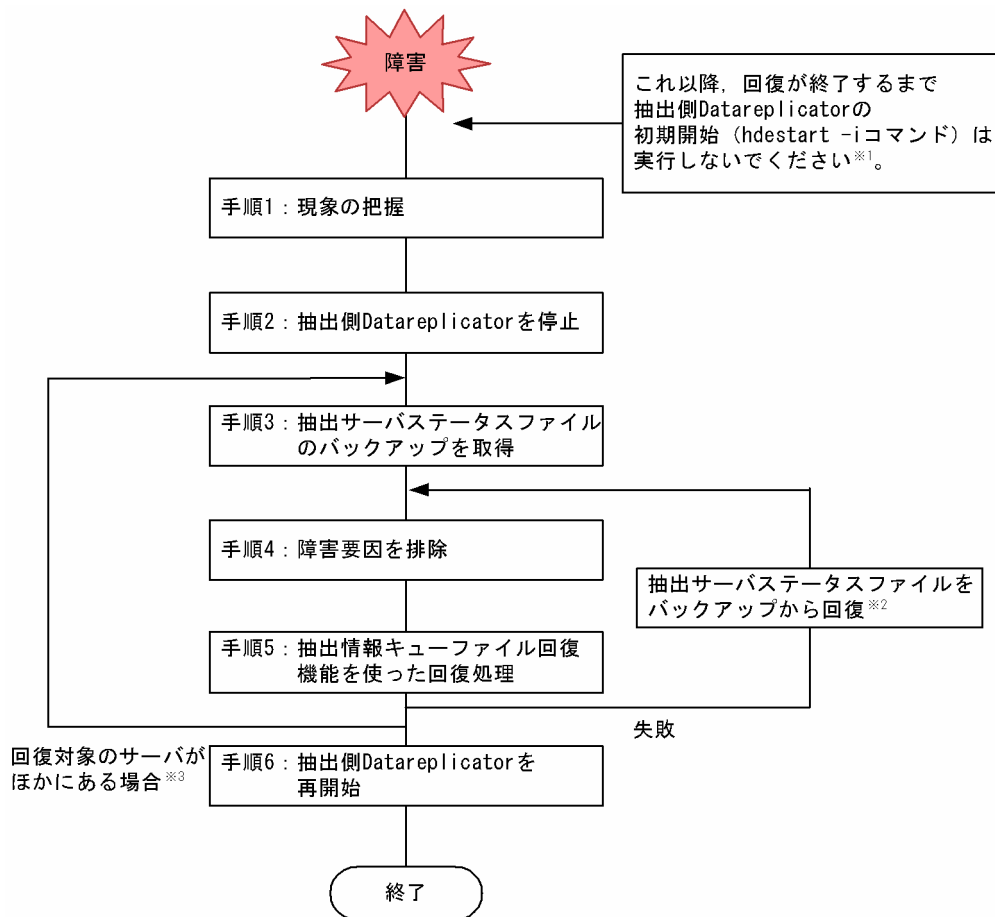
(3) 制限事項

データ連動回復機能を使用しているときに抽出情報キューファイルに障害が発生した場合、抽出情報キューファイル回復機能は使用できません。再度、データ連動回復機能で障害を回復してください。

9.7.3 抽出情報キューファイル回復機能の回復手順

抽出情報キューファイル回復機能を使った回復手順を次の図に示します。

図 9-11 抽出情報キューファイル回復機能を使った回復手順



注※1

初期開始をするとステータスファイルなどが初期化されるので、抽出情報キューファイル回復機能を使用できなくなります。初期開始をした場合は、抽出側 DB と反映側 DB でデータの整合性を合わせ、抽出側 Datareplicator と反映側 Datareplicator を初期開始してください。

注※2

抽出サーバステータスファイルだけをバックアップから回復してください。データ連動用連絡ファイルは HiRDB が使用しているため、回復しないでください。

注※3

抽出側システムが HiRDB/パラレルサーバの場合、バックエンドサーバ単位で回復します。回復対象のバックエンドサーバが複数あるときは、手順 3 から手順 5 までを回復対象のバックエンドサーバの数だけ繰り返してください。

回復手順の詳細について、次の表に示します。

表 9-15 抽出情報キューファイル回復機能を使った回復手順の詳細

項目	対象	操作内容とコマンド実行例	確認内容
手順 1： 現象の把握※ 1	障害回復対象のサーバ	障害の要因を特定します。抽出情報キューファイルにだけ障害が発生している場合は、これ以降の手順に従って回復します。	—
手順 2： 抽出側 Datareplica tor を停止	マネージャノード	抽出側 Datareplicator を停止します。 コマンド実行例： hdestop	—
手順 3： 抽出サーバス テータスファ イルのバック アップを取得	障害回復対象のサーバ	抽出サーバステータスファイルのバックアップを取得します。 コマンド実行例： 「6.4.2(7)(b) 抽出サーバステータスファイルのバックアップ」を参照してください。	—
手順-： 回復に失敗 し、再度回復 する場合	障害回復対象のサーバ	抽出サーバステータスファイルをバックアップから回復します。 コマンド実行例： 「6.4.2(7)(b) 抽出サーバステータスファイルのバックアップ」を参照してください。	—
手順 4： 障害要因を 排除	障害回復対象のサーバ	抽出情報キューファイルの障害の要因を取り除きます。	—
手順 5： 抽出情報 キューファイ ル回復機能 を使った回復 処理	マネージャノード	抽出情報キューファイルを回復します。 コマンド実行例： hdestart -R -k queue -b bes1	回復を開始したときに出力される KFRB05034-I メッセージと、回復を完了したときに出力される KFRB05035-I メッセージの、回復対象ファイルの範囲が同じことを確認してください。 hdestate コマンドで抽出側 Datareplicator が終了状態になるまで監視します※2。又は、マネージャノードの抽出マスタエラー情報ファイルに、抽出側 Datareplicator を終了するメッセージ (KFRB00510-I) が出力されているか確認します。抽出側 Datareplicator の終了後は、エラー情報ファイルにエラーが出力されていないことを確認してください。
手順 6：	マネージャノード	抽出側 Datareplicator を再開始します。	—

項目	対象	操作内容とコマンド実行例	確認内容
抽出側 Datareplicator を再開始		コマンド実行例： hdestart	

(凡例)

－：確認の必要な項目はありません。

注※1

HiRDB のシステムログファイルがラップアラウンドして、回復に必要なログが上書きされていた場合は、抽出情報キューファイル回復機能を使用できません (KFRB05011-E メッセージが出力されます)。データ連動回復機能を使用して回復してください。

注※2

抽出側 Datareplicator が終了するまでは、hdestart コマンドと hdestop コマンドは実行できません。

(1) 抽出情報キューファイル回復機能を使った回復処理

障害が発生する前に抽出処理が完了していて、反映側 Datareplicator に送信していない更新情報がある場合に、抽出情報キューファイルを回復します。

抽出情報キューファイルの回復では、抽出サーバステータスファイルの情報を基に、障害が発生する前に抽出処理が完了していた更新情報をシステムログファイルから抽出して、抽出情報キューファイルに格納します。

(2) 抽出側 Datareplicator を再開始

抽出情報キューファイルを回復した後、抽出側 Datareplicator を再開始します。抽出側 Datareplicator を再開始すると、抽出情報キューファイル回復機能によって再抽出した更新情報が反映側 Datareplicator に送信されます。また、障害が発生した後の更新情報については、抽出処理の続きを実行します。

障害が発生する前に反映側 Datareplicator に更新情報をすべて送信していた抽出情報キューファイルは、初期化された状態から抽出処理を再開します。

(3) 抽出情報キューファイルの回復中に障害が発生した場合

抽出情報キューファイルの回復中に障害が発生した場合は、抽出サーバステータスファイルをバックアップから回復して、「[9.1.2 エラーの対処方法](#)」を参照して障害の要因を取り除いてください。障害の要因を取り除いた後、抽出情報キューファイル回復機能を再度実行してください。

ただし、次のファイルに障害が発生した場合は、抽出情報キューファイル回復機能は使用できなくなります。データ連動回復機能を使用して回復してください。

- 抽出マスタステータスファイル
- 抽出サーバステータスファイル

- データ連動用連絡ファイル

9.8 反映障害による未送信情報の取得（更新 SQL 出力機能）

Datareplicator では、抽出側 DB の更新と反映側 DB の更新には時間差があります。したがって、反映処理の最中に抽出側 DB に障害が発生した場合、抽出側 DB と反映側 DB との間に不整合が発生することがあります。

更新 SQL 出力機能は、このような場合に抽出側 DB からの未送信情報（反映側 DB に反映されていない情報）を SQL 文にしてファイル出力する機能です。これによって、反映されていない情報を知ることができます。

注

更新 SQL 出力機能の使用時に更新情報の抽出、及び反映側システムへの送信はできません。

更新 SQL 出力機能を使用しても、抽出情報キューファイルに格納されている未送信情報がクリアされることはありません。

9.8.1 使用方法

更新 SQL 出力機能の使用方法について説明します。

(1) 前提条件

更新 SQL 出力機能を使用する場合、次に示す条件を満たしている必要があります。

- 抽出側 Datareplicator のプロセスが起動可能である（CPU 障害などから回復している）
- 抽出側システムにある次のファイルが正常な状態である（ディスク障害などから回復している）
 - 抽出システム定義ファイル
 - 抽出環境定義ファイル
 - 送信環境定義ファイル
 - 抽出定義ファイル
 - 抽出定義プリプロセスファイル
 - 抽出情報キューファイル
 - 抽出マスタステータスファイル
 - 抽出サーバステータスファイル
 - データ連動用連絡ファイル
 - 二重化制御ファイル
 - HiRDB システムログ

(2) コマンド

hdestart に -s -L オプションを指定して実行します。コマンドの詳細については、[hdestart コマンド](#)を参照してください。

(3) 使用するプロセス

更新 SQL 出力機能を使用すると、更新 SQL 出力プロセスが起動します。更新 SQL 出力プロセスの詳細を次に示します。

プロセス名	起動方法	終了方法
hdesqlput	hdestart -s -L	hdestop

更新 SQL 出力プロセスは、次に示す点を除いて送信プロセスと同じ動作をします。

- 反映側 Datareplicator への接続及び送信を行いません。
- 抽出サーバステータスファイルの更新を行いません。
- HiRDB の正常停止や連動停止ログ (pdrplstop) を検知した場合でも、停止しません。
- hdeevent -n 0 が実行された場合でも停止しません。
- 抽出システム定義の sendcontrol オペランドに sendmst を指定している場合も、nodemst 方式で動作します。
- 縮退状態の送信先が存在する場合、送信環境定義の overwrite_continue オペランドに false が指定されていても、縮退状態は継続されます。

9.8.2 更新 SQL ファイル

更新 SQL 出力機能によって出力される更新 SQL ファイルについて説明します。

(1) 出力先

更新 SQL ファイルの出力先を次に示します。

項目	出力先
出力先ノード	抽出側 DB の更新対象になる各ノード
出力先ディレクトリ	抽出側 Datareplicator 運用ディレクトリ
ファイル名称	SQLTXT_サーバ名_送信先識別子

同じ名称のファイルが存在している場合、情報がそのファイルに追加出力されます。

また、抽出情報キューファイルに未送信情報が存在しない場合、更新 SQL ファイルは作成されません。

(2) 出力形式

更新 SQL ファイルには、未送信情報が 1 行 1SQL 文の形式で出力されます。SQL 文として出力されるデータは COMMIT されたトランザクションのものだけであり、ROLLBACK されたトランザクションのデータは出力されません。また、トランザクションは COMMIT された順番に並べ替えて出力されます。なお、仕掛かり中の更新情報は出力されません。

出力されるデータの形式を次に示します。

INSERT 文

```
/* 更新日時 */ INSERT INTO 認可識別子.表名(列名,列名,...) VALUES(値※1,値,...);
```

UPDATE 文※²

```
/* 更新日時 */ UPDATE 認可識別子.表名 SET※3 列名=値※1,列名=値,... WHERE※4 列名=値 AND 列名=値 ...;
```

DELETE 文

```
/* 更新日時 */ DELETE FROM 認可識別子.表名 WHERE※4 列名=値※1 AND 列名=値 ...;
```

PURGE TABLE 文※⁵

```
/* 更新日時 */ PURGE TABLE 認可識別子.表名;
```

イベント※⁶

```
/* ****-**-** **:**:** */ /* HDEEVENT イベントコード */
```

トランザクションの終端

```
/* ****-**-** **:**:** */ COMMIT;
```

注※¹

出力する列値の形式と内容を次の表に示します。なお、繰返し列の場合は、NULL 値/非 NULL 値に関係なく、列値は*MCOL*と出力されます。

列属性	文字集合指定	hdestart コマンドの-H オプション	出力される形式	出力される内容	備考
CHAR, 又は VARCHAR	なし	なし	文字形式 (例: 'aaa')	該当する列の列値	制御コードはピリオドとして出力されます。
		あり	16 進形式 (例: X'C3C3C3')		—
	あり	なし	文字形式 (例: 'aaa')	該当する列の列値を DB ロケールに文字コード	制御コードはピリオドとして出力されます。
		あり	16 進形式		—

列属性	文字集合指定	hdestart コマンドの-H オプション	出力される形式	出力される内容	備考
			(例: X'C3C3C3')	変換した後の列値	
MCHAR, 又は MVARC HAR	-	なし	文字形式 (例: M'あいうえお')	該当する列の列値	制御コードはピリオドとして出力されま す。
		あり	16 進形式 (例: X'81C141')		-
NCHAR, 又は NVARC HAR	-	なし	文字形式 (例: N'あいう')		制御コードはピリオドとして出力されま す。
		あり	16 進形式 (例: X'81C181C2')		-
INTEGER, 又は SMALLINT	-	-	10 進形式 (例: 100)		-
DECIMAL	-	-	10 進形式 (例: 100)		パック形式データを 文字列に変換する際 にエラーが発生した 場合, その列は 16 進形式で出力されま す。
FLOAT, 又は SMALLFLT	-	-	浮動小数点形式 (例: 1.0...0E+02)		-
DATE	-	-	文字形式 (例: '2008-04-15')		パック形式データを 文字列に変換する際 にエラーが発生した 場合, その列は 16 進形式で出力されま す。
TIME	-	-	文字形式 (例: '16:15:30')		
TIMESTAM P	-	-	文字形式 (例: '2008-04-15 16:15:30')		
INTERVAL YEAR TO DAY	-	-	10 進形式 (例: +12340102.)		
INTERVAL HOUR TO SECOND	-	-	10 進形式 (例: +112233.)		
BLOB	-	-	固定文字列	**BLOB**	• SUBSTR 演算の 場合も, 該当す
BINARY	-	-		**BINARY**	

列属性	文字集合指定	hdestart コマンドの-H オプション	出力される形式	出力される内容	備考
					る固定文字列 ("*BLOB*"又は"*BINARY*") が出力されます。 • NULL 値/非 NULL 値に関係なく、該当する固定文字列 ("*BLOB*"又は"*BINARY*") が出力されます。
ADT	—	—	固定文字列	"*ADT*"	NULL 値/非 NULL 値に関係なく、該当する固定文字列"*ADT*"が出力されます。

(凡例)

— : 該当しません。

注※2

繰返し列に対する更新情報を出力する場合、次の制限があります。

- UPDATE ADD, UPDATE SET, 及び UPDATE DELETE は、すべて UPDATE SET として出力されます。
- 要素指定による更新の場合でも、列名に対する添え字及び'*'は出力されません。

注※3

定義長が 256 バイト以上の可変長データが非更新の場合、その列は SET 句に出力されません。

注※4

WHERE 句にはマッピングキー列が出力されます。列値が NULL の場合には、「列名 IS NULL」の形式で出力されます。

注※5

複数サーバにわたる分割表に対する PURGE TABLE の更新情報を検知した場合、送信環境定義の prg_eventno オペランド指定の有無に従って動作します。指定値による動作の違いを次に示します。

オペランドの指定	動作
あり	prg_eventno オペランドに指定した番号のイベントを出力します。イベント出力後、処理を続行します。
なし	該当する PURGE TABLE を無視します。

注※6

イベントコード 0 は本来反映側 Datareplicator への送信対象ではありませんが、このファイルには出力されます。

一つのトランザクション中で複数のイベントが発行された場合、最後に発行されたイベントだけ出力されます。また、hdevent コマンドによるイベント発行の場合、イベントが出力された直後に COMMIT が出力されます。

(3) 出力例

更新 SQL ファイルの出力例を次に示します。

```
/* 2004-12-29 19:18:00 */ INSERT INTO "USR1"."T1"("C1","C2") VALUES(1,'a');
/* 2004-12-29 19:18:30 */ UPDATE "USR1"."T1" SET "C1"=2,"C2"=' b' WHERE "C1"=2;
/* 2004-12-29 19:19:00 */ DELETE FROM "USR1"."T1" WHERE "C1" IS NULL;
/* 2004-12-29 19:19:30 */ PURGE TABLE "USR1"."T1";
/* ****-**-** **:**:** */ /* HDEEVENT 200 */
/* ****-**-** **:**:** */ COMMIT;
```

9.8.3 操作手順

更新 SQL 出力を実行するには、次のように操作します。

1. 反映側 Datareplicator が起動している場合、停止します。

これは、項番 3 又は項番 6 でオプションを誤って指定した場合に反映側システムに接続してしまうのを防ぐための操作です。

```
hdsstop -t immediate
```

2. 送信環境定義の overwrite オペランドに true を指定している場合は、抽出側 Datareplicator を停止した後、false に変更します。

これは、項番 3 で抽出情報キューファイルが満杯になった場合に縮退してしまうのを防ぐための操作です。

3. 抽出側 Datareplicator の抽出処理を開始します。

```
hdestart -e
```

4. 次に示す方法で、システムログのすべての更新情報を抽出したことを確認します。

- 抽出側 Datareplicator のエラー情報ファイルに KFRB00042-E (抽出情報キューファイルが満杯) が出ていないことを確認します。

KFRB00042-E が出ている場合、抽出情報キューファイルを追加した後、抽出処理を再開してください※。

```
hdemodq
```

注※

抽出情報キューファイルの最大数は 16 個です。16 個存在する状態で満杯になった場合、一部の更新情報は出力されません。

- HiRDB システムログの抽出状況から、必要な更新情報の抽出が終了していることを確認します。

pdlogls コマンドの実行結果と pdls -d rpl -j コマンド実行結果の System Log Extract Point を比較し、前者を後者が追い越していることを確認します。次に例を示します。

pdlogls の出力例

```
$ pdlogls -d sys -s flora370
HOSTNAME : flora370(151739)
Group    Type Server   Gen No.  Status  Run ID    Block No.
log10    sys  sds01    1        oc-d--u  3e6835a9  1        40
log11    sys  sds01    0        os----- 00000000  0        0
```

pdls の出力例

```
$ pdls -d rpl -j -s flora370
SYSTEMID      : HRD1(150621)
Data replication : Y
UNITID       : unt1(150621)
Data replication : Y
SERVER NAME   : sds01
Extract Database : Y
Extract Status : C
System Log Extract Point :
Run ID  Group   Gen No.  BLock No.
3e6835a9 log10    1        41
System Log Sync Info :
Run ID  Group   Gen No.  BLock No.
3e6835a9 log10    1        13
```

pdlogls の結果では最終ブロック番号が log10 の 40 ブロック目となっていますが、pdls では現在抽出中のシステムログブロック番号が log10 の 41 ブロック目となっています。これによって、必要な更新情報の抽出処理が終了していることが分かります。

- 抽出側 Datareplicator を停止します。

```
hdestop
```

- すべての更新 SQL ファイルを削除した後、抽出側 Datareplicator の更新 SQL 出力処理を開始します。

```
hdestart -s -L
```

- hdestate コマンドを実行して、Queue write position と Queue current pos の値が等しくなったことを確認します。

Queue current pos については [hdestate](#) コマンドを参照してください。

- 抽出側 Datareplicator を停止します。

```
hdestop
```

- テキストエディタなどで更新 SQL ファイルを開き、未送信データを確認してください。

未送信データを反映側へ送信する必要がある場合には、送信処理を開始してください。送信する必要がない場合には、抽出情報キューファイルに格納されている未送信データを削除するために、抽出側 Datareplicator を部分初期化又は初期化してください。

9.8.4 注意事項

- Datareplicator で出力されるメッセージでは、送信プロセスも出力プロセスも同じく「Sender process」と表記されます（日本語メッセージ中では「送信プロセス」）。出力されたメッセージが送信プロセス／出力プロセスのどちらに対するものなのかは、メッセージ中に記述されているメッセージ出力プロセス名で判断してください。
- 抽出システム定義の nodecontrol オペランドに server を指定している場合、更新 SQL 出力機能実行中に系切り替えが発生すると、Datareplicator は次の表のような動作をします。

表 9-16 更新 SQL 出力機能実行中に系切り替えが発生した場合の Datareplicator の動作

系切り替えの発生状況	Datareplicator の動作
MST が存在するホストで系切り替えが発生	系切り替え完了後、MST はいったん起動されますが、更新 SQL 出力状態で稼働中の NMT からの接続要求を拒否します。その結果、すべての NMT 及び MST が停止します。
NMT が存在するホストで系切り替えが発生	系切り替え完了後、NMT はいったん起動されますが [※] 、更新 SQL 出力状態で稼働中の MST は、NMT からの接続要求を拒否します。その結果、系切り替えが発生した NMT は停止します。

(凡例)

MST：抽出マスタプロセス

NMT：抽出ノードマスタプロセス

注※

NMT は hdestart_n コマンドで起動されるため、抽出プロセスと送信プロセスが起動されます。

10

メッセージの一覧

この章では、メッセージの出力形式と対処方法について説明します。

10.1 メッセージの概要

Datareplicator が出力するメッセージについて説明します。

HiRDB が出力するメッセージについては、「HiRDB メッセージ」を参照してください。

10.1.1 メッセージの出力先

Datareplicator のメッセージの出力先を次に示します。

(1) UNIX (HP-UX, Solaris, AIX, Linux) の場合

UNIX 版 Datareplicator のメッセージの出力先を次に示します。

- 標準エラー出力
- エラー情報ファイル

抽出側 Datareplicator の場合：

抽出マスタエラー情報ファイル, 抽出ノードマスタエラー情報ファイル

反映側 Datareplicator の場合：

反映エラー情報ファイル

- syslog ファイル

(2) Windows の場合

Windows 版 Datareplicator のメッセージの出力先を次に示します。

- 標準エラー出力

Datareplicator のコマンドを実行した画面です。

- エラー情報ファイル

抽出側 Datareplicator の場合：

抽出マスタエラー情報ファイル, 抽出ノードマスタエラー情報ファイル

反映側 Datareplicator の場合：

反映エラー情報ファイル

- イベントログ

UNIX 版 Datareplicator で syslog ファイルに出力されるメッセージが、Windows のイベントログに出力されます。

10.1.2 メッセージの出力形式

メッセージの出力形式を、出力先別に説明します。

(1) 出力先が標準エラー出力、syslog ファイル又はイベントログの場合

Datareplicator が出力するメッセージの形式を次に示します。

```
KFRBnnnnn - T XX YY (ZZ...ZZ) MM...MM
```

KFRBnnnnn - T

メッセージ ID (英数字 11 文字)

XX

抽出側 Datareplicator 識別子 又は反映側 Datareplicator 識別子 (2 けたの 16 進文字)

ただし、次の状態のときには、抽出側 Datareplicator 識別子及び反映側 Datareplicator 識別子を表示できません。このため、識別子の代わりに「..」を表示します。

- コマンドからのエラー出力
- 共用メモリアタッチ以前のエラー出力
- メッセージファイルオープン以前のエラー出力

YY

データ連動識別子 (2 けたの 16 進文字)

ただし、次の状態のときには、データ連動識別子を表示できません。このため、データ連動識別子の代わりに「..」を表示します。

- コマンドからのエラー出力
- 共用メモリアタッチ以前のエラー出力
- メッセージファイルオープン以前のエラー出力
- データ連動識別子の単位に依存しない処理でのエラー出力

ZZ...ZZ

メッセージ出力プロセス名、及び反映グループ名

反映グループ名は、反映プロセス以降の処理でのエラー出力の場合だけ表示します。

MM...MM

メッセージテキスト (最大 242 けた)

(2) 出力先がエラー情報ファイルの場合

Datareplicator では、システム内で発生したエラーのログをエラー情報ファイルの中に蓄積します。抽出側 Datareplicator のエラー情報ファイルの出力形式と運用方法については、「[6.4.2 抽出側 Datareplicator で使うファイルの運用](#)」を参照してください。反映側 Datareplicator のエラー情報ファ

イルの出力形式と運用方法については、「6.7.2 反映側 Datareplicator で使うファイルの運用」を参照してください。

10.1.3 メッセージの記述形式

このマニュアルでメッセージを説明する形式を次に示します。

(1) 記述形式

KFRBnnnnn - T

英文メッセージテキスト (Y)

和文メッセージテキスト

メッセージの意味を説明しています。

(S)メッセージを出力した後に、Datareplicator がする処置を示します。

(O)メッセージを受け取ったオペレータが取る処置を示します。

[対策]メッセージを受け取った Datareplicator の管理者が取る処置を示します。

注

- メッセージ中で使用する「保守員に連絡してください。」とは、当社社員、又は当社営業担当部署に連絡することを示します。
- 英文メッセージテキストだけで、和文メッセージテキストがないメッセージもあります。

(2) メッセージの記号の説明

メッセージ ID の記号の意味を次に示します。

KFRB

Datareplicator が出力するメッセージであることを示すコードです。

nnnnn

メッセージ番号を示します。

T

メッセージの重要度を示します。

E：エラーメッセージ

機能を実行できないエラーが起こったことを示します。

W：警告メッセージ

リソースの使用状況などについての警告，又はコマンドの指定が間違っているが，値を仮定して処理を続けることを示します。

I：情報メッセージ

上記の E，W に該当しない，単純な動作状況を示します。

Q：応答待ちメッセージ

出力したメッセージに対するユーザの応答を待っていることを示します。

Y

メッセージの出力先種別を示します。

一つのメッセージが複数の出力先種別を持つ場合は，出力される可能性のある種別を「+」でつないで記述します。

C：エラー情報ファイル，及び syslog ファイル※1

S：標準エラー出力

L：syslog ファイル※1

E：エラー情報ファイル又はイベントログ（Windows の場合）

T：標準出力※2

注※1

Windows の場合は，イベントログに出力されます。

注※2

hdeinfoget，及び hdsinfoget コマンドが標準出力へ出力するメッセージだけが該当します。

(3) メッセージテキストの説明

(a) メッセージテキストで記述している内容

メッセージの意味，現象，及び挿入文字の説明を記述しています。

メッセージテキストの中に使っている記号の意味を次に示します。

{ }

{ } で囲んだ複数のテキストは，そのうちのどれかが出力されます。また，{ } 内の | は区切りを示します。

出力例

{認可識別子 | パスワード}

[]

[] で囲んだテキストは，出力されないことがあります。

(b) 英文メッセージテキストと和文メッセージテキストの切り替え

Datareplicator のメッセージテキストは、英文と和文の 2 種類で記述しています。英文と和文は、抽出システム定義又は反映システム定義の msglocale オペランドで切り替えます。

英文と和文の切り替えができるのは、エラー情報ファイル及び syslog ファイルに出力されるメッセージです。標準出力に出力されるメッセージ、及び英文／和文の切り替えができなかったメッセージは、英文で出力されます。これらのメッセージについては、和文メッセージテキストを記述していません。

(4) 詳細情報の説明

エラー情報ファイルに出力するときに、一部のエラーメッセージに対して付加される詳細情報を記述しています。

(a) 抽出トランザクション情報

抽出トランザクション情報は、データ連動回復機能を使用する場合に入力とする情報で、トランザクションを一意に示します。

抽出トランザクションが出力されない場合

Datareplicator Extension を使って反映する場合、抽出トランザクション情報は出力されません。

抽出トランザクションを詳細情報として出力するメッセージ番号

次に示すメッセージ番号の詳細情報として抽出トランザクション情報を出力します。

03004, 03006, 03009, 03010, 03021, 03022, 03023, 03024, 03025, 03027, 03028,
03029, 03031, 03032, 03033, 03034, 03036, 03038, 03040, 03041, 03042, 03043,
03044, 03045, 03046, 03047, 03048, 03049, 03051, 03052, 03053, 03056, 03057,
03058, 03070, 03071, 03074, 03080, 03081, 03083, 03084, 03085, 03086, 03087,
03088, 03089, 03090, 03091, 03092, 03101, 03102, 03103, 03104, 03105, 03106

抽出トランザクション情報の出力形式

```
<テキスト> Additional Transaction Info = nn...nn.
```

メッセージ番号が 03009 以外のときは、処理中のトランザクション情報になります。03009 のときは、同期点発行前に処理した最後のトランザクションについての情報です。nn...nn には、次に示す情報が出力されます。

XDM/DS が送信元で、抽出元 DBMS が TMS-4V/SP 以外のとき：

アクティビティ ID を 16 進文字形式で出力します。

XDM/DS が送信元で、抽出元 DBMS が TMS-4V/SP のとき：

APJ ジャーナル ID を [a-] bbbbbbbb-ccccccc-ddd の形式で出力します。

a : DMS-ID。ただし、0x00 のときは出力しません。

bbbbbbb : ラン ID (16 進文字形式)

ccccccc : 中央処理通番 (16 進文字形式)

ddd : JSQ (10 進文字形式)

(b) 更新情報識別子

更新情報識別子は、反映抑止機能を使用する場合に入力とする情報で、更新情報を一意に示します。

更新情報識別子の出力条件

更新情報識別子は、次の条件を満たしている場合に出力されます。

- 送信元が次の表に示すバージョンの場合

表 10-1 メッセージに詳細情報を追加出力するプログラムとそのバージョン

抽出側プログラム	条件	バージョン
HiRDB Datareplicator	なし	04-00-/O 以降
HiRDB Datareplicator Version 5.0	なし	05-03-/D 以降
HiRDB Datareplicator Version 6	なし	06-01-/A 以降
HiRDB Datareplicator Version 7	なし	07-00 以降
HiRDB Datareplicator Version 8	なし	08-00 以降
HiRDB Datareplicator Version 10	なし	10-00 以降
XDM/DS	TMS/4V からの抽出	07-01 以降
	上記以外からの抽出	08-03 以降
DBDR	なし	01-00 以降

- Datareplicator Extension を使用しないで反映する場合

更新情報識別子を詳細情報として出力するメッセージ番号

次に示すメッセージ番号の詳細情報として更新情報識別子を出力します。

03004, 03010, 03028, 03029, 03031, 03032, 03033, 03034, 03036, 03038, 03040, 03041, 03042, 03043, 03044, 03045, 03046, 03047, 03048, 03049, 03051, 03052, 03053, 03056, 03057, 03058, 03070, 03071, 03074, 03080, 03081, 03083, 03084, 03085, 03086, 03087, 03088, 03089, 03090, 03091, 03092, 03101, 03102, 03103, 03104, 03105, 03106

更新情報識別子の出力形式

<テキスト> ExtractID = nn...nn.

10.2 メッセージの詳細

KFRB00001-E

File(pipe) open error, file name = aa...aa, errno = xx...xx. (C)

ファイル又はパイプのオープン処理中にエラーが発生しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : ファイル名

ファイル名には実際の名称ではなく、次のキーワードが出力される場合があります。

- StatusFile : 抽出サーバステータスファイル
- RECEIVING QUEUE FILE : 反映情報キューファイル
- STATUS FILE : 反映ステータスファイル
- ERRORLOG FILE : エラー情報ファイル
- UNREFLECTED DATA FILE : 未反映情報ファイル
- START UP PARAMETER FILE : 反映システム定義
- REFLECT ENVIRONMENT FILE : 反映環境定義
- REFLECT DEFINITION FILE : 反映定義
- PIPE : プロセス間連絡用 PIPE
- SOCKET : 通信用ソケット

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00002-E

File(pipe) read error, file name = aa...aa, errno = xx...xx. (C)

ファイル又はパイプからの読み込み処理中にエラーが発生しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : ファイル名

ファイル名には実際の名称ではなく、次のキーワードが出力される場合があります。

- StatusFile : 抽出サーバステータスファイル
- RECEIVING QUEUE FILE : 反映情報キューファイル
- STATUS FILE : 反映ステータスファイル

- ERRORLOG FILE：エラー情報ファイル
- UNREFLECTED DATA FILE：未反映情報ファイル
- START UP PARAMETER FILE：反映システム定義
- REFLECT ENVIRONMENT FILE：反映環境定義
- REFLECT DEFINITION FILE：反映定義
- PIPE：プロセス間連絡用 PIPE
- SOCKET：通信用ソケット

xx...xx：errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

エラー番号が 0 の場合は、ファイルの内容が不正です。

KFRB00003-E

File(pipe) write error, file name = aa...aa, errno = xx...xx. (C)

ファイル又はパイプへの書き込み処理中にエラーが発生しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa：ファイル名

ファイル名には実際の名称ではなく、次のキーワードが出力される場合があります。

- StatusFile：抽出サーバステータスファイル
- RECEIVING QUEUE FILE：反映情報キューファイル
- STATUS FILE：反映ステータスファイル
- ERRORLOG FILE：エラー情報ファイル
- UNREFLECTED DATA FILE：未反映情報ファイル
- START UP PARAMETER FILE：反映システム定義
- REFLECT ENVIRONMENT FILE：反映環境定義
- REFLECT DEFINITION FILE：反映定義
- PIPE：プロセス間連絡用 PIPE
- SOCKET：通信用ソケット

xx...xx：errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

エラー番号が 0 の場合は、ディスク容量が不足しています。

KFRB00004-E

File(pipe) close error, file name = aa...aa, errno = xx...xx. (C)

ファイル又はパイプのクローズ処理中にエラーが発生しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : ファイル名

ファイル名には実際の名称ではなく、次のキーワードが出力される場合があります。

- StatusFile : 抽出サーバステータスファイル
- RECEIVING QUEUE FILE : 反映情報キューファイル
- STATUS FILE : 反映ステータスファイル
- ERRORLOG FILE : エラー情報ファイル
- UNREFLECTED DATA FILE : 未反映情報ファイル
- START UP PARAMETER FILE : 反映システム定義
- REFLECT ENVIRONMENT FILE : 反映環境定義
- REFLECT DEFINITION FILE : 反映定義
- PIPE : プロセス間連絡用 PIPE
- SOCKET : 通信用ソケット

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00005-E

Unable to find update queue file. (C)

空き反映情報キューファイルがないため、更新データは書き込めません。

(S)受信処理を中断します。

(O)Datareplicator が反映情報キューファイルへの書き込みを自動的にリトライ[※]します。

注※

送信環境定義の retry_interval オペランドの指定値の間隔でリトライします。

ただし、このメッセージが繰り返し出力される場合は、hdsstate コマンドを実行し、「<EXTRACTED DATA QUEUE FILE INFORMATION>」の「current used ratio」が示す反映情報キューファイルの使用率を確認し、以下の対処を行ってください。

使用率	要因	対処
0%	抽出側で実行した 1 トランザクションの更新量が反映情報キューファイルのサイズを超えています。	<p>hdsstop -t immediate 又は force コマンドを実行して反映側 Datareplicator を終了してください。その後、次のどちらかの対処をした後、hdsstart -i コマンドで反映側 Datareplicator を初期開始してください。</p> <ul style="list-style-type: none"> 反映情報キューファイルのサイズ（反映環境定義の queuesize オペランドで指定）を大きくする。 反映情報キューファイルの数を増やす。
1%以上	hdsstate コマンドの「hdsreflect」が-1 の場合、反映処理が停止しています。	エラー情報ファイルを確認し、エラーメッセージが出力されている場合は、該当エラーメッセージに従って対処した後、hdsrftl コマンドで反映処理を再起動してください。
	上記以外の場合、反映処理が遅延しているおそれがあります。	「6.12 チューニング」を参照し、Datareplicator の反映処理をチューニングしてください。

KFRB00006-E

Insufficient memory, size = aa...aa, info = bb...bb. (C)

メモリ不足が発生しました。要求サイズ = aa...aa, 詳細 = bb...bb.

aa...aa：要求サイズ（システムコール内部処理でメモリ不足になったときは、-1）

bb...bb：付加情報

(S)処理を中断します。

(O)原因を取り除いた後、再実行してください。

KFRB00007-E

Failed to generate process, code = xx. (C)

プロセスの生成に失敗しました。詳細コード = xx.

xx：発生したプロセス名を示す詳細コード

00：反映マスタプロセス

01：反映通信マスタプロセス

02：受信プロセス

03：反映定義サーバプロセス

04：反映プロセス

05：反映 SQL プロセス

(S)処理を中断します。

(O)再実行してください。その結果、同じエラーが発生した場合には、保守員に連絡してください。

KFRB00008-E

Parent process disappeared, process name = aa...aa. (C)

何らかの原因で親プロセスが消滅しました。プロセス名 = aa...aa.

aa...aa：消滅したプロセス名

(S)処理を中断します。

(O)再実行してください。正常に動作しない場合は、hdsstart -i コマンドで再実行してください。

KFRB00009-E

Internal error, function name = aa...aa, code = xx. (S + L + E)

内部矛盾が発生しました。関数名 = aa...aa, 詳細コード = xx.

aa...aa：発生関数名

xx：詳細コード

(S)処理を中断します。

(O)保守員に連絡してください。

KFRB00010-E

Read/write pointer set error (lseek), file name = aa...aa, errno = xx...xx. (C)

リードポインタ又はライトポインタの設定に失敗しました (シーク)。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa：ファイル名

xx...xx：errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00011-I

No log data in update queue file. (C)

反映情報キューファイルの中にデータがありません。

(S)処理を続行します。

KFRB00012-E

Update queue file is full. (C)

反映情報キューファイルに空きがありません。

(S)受信処理を中断します。

(O)反映処理が終了した後、データを蓄積するのに十分な容量の反映情報キューファイルを指定して、`hdsstart -i` コマンドで再起動してください。反映情報キューファイルの容量の見積もりについては、「[4. システム設計](#)」を参照してください。

KFRB00013-W

Syslog file write error. (C)

syslog ファイルの書き込みに失敗しました。

(S)処理を続行します。

ただし、syslog ファイルへメッセージは出力されません。

KFRB00014-W

Reflect information file write error, function name = aa...aa, errno = xx...xx. (C)

反映エラー情報ファイルの書き込み処理中にエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : エラーが発生した関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。

ただし、反映エラー情報ファイルへメッセージは出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、`errno.h` 又は OS のマニュアルを参照してください。

KFRB00015-W

Unreflected data file write error, function name = aa...aa, errno = xx...xx. (C)

未反映情報ファイルの書き込み処理中にエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : エラー発生関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。

ただし、未反映情報は、ファイルに出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00016-W

```
Reflect information /unreflected data file initialize error, function name = aa...aa, file name = bb...bb, errno = xx...xx. (C)
```

反映エラー情報ファイル又は未反映情報ファイルの初期設定に失敗しました。関数名 = aa...aa, ファイル名 = bb...bb, エラー番号 = xx...xx.

aa...aa : エラー発生関数名

bb...bb : エラー発生ファイルの絶対パス名

xx...xx : エラー番号

(S)処理を続行します。

ただし、反映エラー情報ファイル又は未反映情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00017-W

```
Reflect information /unreflected data file open error, function name = aa...aa, file name = bb...bb, errno = xx...xx. (C)
```

反映エラー情報ファイル又は未反映情報ファイルのオープン処理中にエラーが発生しました。関数名 = aa...aa, ファイル名 = bb...bb, エラー番号 = xx...xx.

aa...aa : エラー発生関数名

bb...bb : エラー発生ファイルの絶対パス名

xx...xx : エラー番号

(S)反映エラー情報ファイルのオープン処理中にエラーが発生した場合、処理を続行します。未反映情報ファイルのオープン処理中にエラーが発生した場合、処理を中断します。反映エラー情報ファイル又は未反映情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00018-E

Exec error, program = aa...aa, errno = xx...xx. (C)

プログラムのローディングに失敗しました。プログラム名 = aa...aa, エラー番号 = xx...xx.

aa...aa : プログラム名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00019-E

Fork error, process = aa...aa, errno = xx...xx. (C)

プロセスの生成に失敗しました。プロセス名 = aa...aa, エラー番号 = xx...xx.

aa...aa : プロセス名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00020-E

Malloc error, errno = xx...xx, size = aa...aa, info = bb...bb. (C)

領域確保処理中にメモリ不足が発生しました。エラー番号 = xx...xx, 要求サイズ = aa...aa, 詳細 = bb...bb.

xx...xx : errno に設定されたエラー番号

aa...aa : 要求サイズ

bb...bb : 詳細情報

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00021-E

Semaphore delete error, errno = xx...xx. (C)

セマフォの消去処理に失敗しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00022-E

Semaphore id get error, errno = xx...xx, Key = aa...aa. (C)

セマフォの割り当てに失敗しました。エラー番号 = xx...xx, セマフォキー = aa...aa.

xx...xx : errno に設定されたエラー番号

aa...aa : セマフォキー

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00023-E

Semaphore operation error, operation = aa...aa, errno = xx...xx. (C)

セマフォの操作に失敗しました。操作 = aa...aa, エラー番号 = xx...xx.

aa...aa : セマフォ操作

xx...xx : エラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00024-E

File information get error (stat), errno = xx...xx. (C + S)

ファイルステータス情報の取得に失敗しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00025-E

Unable to attach shared memory, shared memory type = aa...aa, errno = xx...xx, ID = bb...bb. (C)

共用メモリのアタッチに失敗しました。共用メモリ種別 = aa...aa, エラー番号 = xx...xx, 共用メモリ ID = bb...bb.

aa...aa : 共用メモリ種別

COMMON (反映共通共用メモリ)

DEFSERV (反映定義解析共用メモリ)

ERRTXT (メッセージテキスト共用メモリ)

EXTCOM (抽出共通共用メモリ)

EXTDEF (抽出定義解析共用メモリ)

LOBSHM (BLOB 反映用共用メモリ)

NMTCMD (抽出ノードマスターコマンド間連絡用共用メモリ)

STATINF (ステータス情報共用メモリ)

SYNCINF (同期反映グループ用共用メモリ)

TBILIST (トランザクションブランチ情報管理リスト用共用メモリ)

xx...xx : errno に設定されたエラー番号

bb...bb : 共用メモリ ID

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00026-E

Unreflected data file is full. (C)

未反映情報ファイルが満杯です。

(S)処理を中断します。

(O)反映環境定義の unreffilesz オペランドの設定値を大きくした後、反映処理を再起動してください。

KFRB00027-E

Unable to establish language environment. (C)

文字コード環境の設定に失敗しました。

(S)処理を中断します。

(O)コードマッピングテーブルファイルを読み込むのに十分なメモリを確保してもメッセージが出力される場合は、コードマッピングテーブルを再作成してください。

KFRB00028-E

Unable to set up signal handler, errno = xx...xx. (C)

シグナルハンドラの登録に失敗しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)システムを終了します。

(O)内部エラーです。保守員に連絡してください。

KFRB00029-E

Unable to allocate shared memory, shared memory type = aa...aa, errno = xx...xx, Key = bb...bb, size = cc...cc. (C)

共用メモリの割り当てに失敗しました。共用メモリ種別 = aa...aa, エラー番号 = xx...xx, 共用メモリキー = bb...bb, 要求サイズ = cc...cc.

aa...aa : 共用メモリ種別

COMMON (反映共通共用メモリ)

DEFSERV (反映定義解析共用メモリ)

ERRTXT (メッセージテキスト共用メモリ)

EXTCOM (抽出共通共用メモリ)

EXTDEF (抽出定義解析共用メモリ)

LOBSHM (BLOB 反映用共用メモリ)

NMTCMD (抽出ノードマスターコマンド間連絡用共用メモリ)

STATINF (ステータス情報共用メモリ)

SYNCINF (同期反映グループ用共用メモリ)

TBILIST (トランザクションブランチ情報管理リスト用共用メモリ)

TRNINF (トランザクション管理共用メモリ)

xx...xx : errno に設定されたエラー番号

bb...bb : 共用メモリキー

cc...cc : 要求サイズ

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00030-E

No error message data in message text file, message id = nn...nn. (C)

メッセージファイルの中にメッセージテキストがありません。

nn...nn : メッセージ ID

(S)処理を続行します。

(O)内部エラーです。保守員に連絡してください。

KFRB00031-E

Update queue file sequence error. (C)

反映情報キューファイルの順番が正しくありません。

反映情報キューファイルの順番が正しくないか、又は反映情報キューファイルの内容が不正です。

(S)処理を中断します。

(O)内部エラーです。保守員に連絡してください。

KFRB00032-E

Error occurred in getting file status, file name = aa...aa, errno = xx...xx. (C)

ファイルのステータス取り出し処理中にエラーが発生しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : ファイル名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00033-E

Invalid transmission queue file, file name = aa...aa. (C + S)

抽出情報キューファイルの内容が不正です。ファイル名 = aa...aa.

aa...aa : ファイル名

(S)処理を中断します。

(O)シンボリックリンクを使用している場合は、指定内容が別のファイルと重複していないか確認してください。そのほかの場合は、保守員に連絡してください。

KFRB00034-E

Invalid extract status file, file name = aa...aa. (C + S)

抽出時のステータスファイルの内容が不正です。ファイル名 = aa...aa.

aa...aa : ファイル名

(S)処理を中断します。

(O)シンボリックリンクを使用している場合は、指定内容が別のファイルと重複していないか確認してください。そのほかの場合は、保守員に連絡してください。

KFRB00035-E

Error occurred in getting file-disk information, file name = aa...aa. (C)

ファイルディスク情報の取得に失敗しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : ファイル名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00036-W

Extract master information file initialization error, function name = aa...aa, errno = xx...xx. (C)

抽出マスタエラー情報ファイルの初期設定に失敗しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。ただし、抽出マスタエラー情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。
エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00037-W

Extract slave information file initialization error, function name = aa...aa, errno = xx...xx. (C)

抽出ノードマスタエラー情報ファイルの初期設定に失敗しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。ただし、抽出ノードマスタエラー情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。
エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00038-W

Extract master information file open error, function name = aa...aa, errno = xx...xx. (C)
抽出マスタエラー情報ファイルのオープン処理中にエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。ただし、抽出マスタエラー情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。
エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00039-W

Extract slave information file open error, function name = aa...aa, errno = xx...xx. (C)
抽出ノードマスタエラー情報ファイルのオープン処理中にエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。ただし、抽出ノードマスタエラー情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00040-W

Extract master information file write error, function name = aa...aa, errno = xx...xx. (C)

抽出マスタエラー情報ファイルの書き込み処理中にエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。ただし、抽出マスタエラー情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00041-W

Extract slave information file write error, function name = aa...aa, errno = xx...xx. (C)

抽出ノードマスタエラー情報ファイルの書き込み処理中にエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を続行します。ただし、抽出ノードマスタエラー情報ファイルへ情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00042-E

Transmission queue file is full. (C)

抽出情報キューファイルに空きがありません。

(S)空きができるまで、蓄積処理を一定間隔でリトライします。

(O)hdestate コマンドを実行し、「<TRANSMISSION QUEUE FILE INFORMATION>」の「current used ratio」が示す抽出情報キューファイルの使用率を確認し、以下の対処を行ってください。

使用率	要因	対処
100%	hdestate コマンドの「<COMMUNICATIONINFORMATION>」の「Status」が「not active」又は「not active(error)」の場合、送信処理が停止しています。 上記以外の場合、送信環境定義の retry_interval オペランドの指定値時間を経過しても、hdestate コマンドの「Queue write position」が示す抽出情報キューファイルの書き込み位置が進まないときは、長大トランザクション※のおそれがあります。	抽出ノードマスタエラー情報ファイルを確認し、エラーメッセージが出力されている場合は、該当エラーメッセージに従って対処した後、hdestart -s コマンドで送信処理を再起動してください。 hdemodq コマンドで抽出情報キューファイルを追加した後、抽出側 Datarepictor を再起動してください。 抽出情報キューファイル数を最大まで増やしても KFRB00042-E を出力する場合、抽出環境定義の queuesize オペランドの値を見直してください。その後、抽出側 DB、反映側 DB の整合性を確保した後、抽出側 Datarepicator、反映側 Datareplicator それぞれを初期化してください。
1%~99%	送信処理が遅延しているおそれがあります。	「6.12 チューニング」を参照し、Datarepicator の送信処理をチューニングしてください。

注※

長大トランザクションとは、次のどちらかのトランザクションのことを示します。

- (a)pdload など多量の更新を行っているトランザクション。
- (b)表に更新を行っているが、長時間決着していないトランザクション。

KFRB00043-I

No log data in transmission queue file. (C)
抽出情報キューファイル中に更新情報がありません。

(S)処理を続行します。

KFRB00044-E

Unable to set up signal handler, kind = aa...aa, errno = xx...xx. (C)
シグナルハンドラの登録に失敗しました。シグナル種別 = aa...aa, エラー番号 = xx...xx.

aa...aa : シグナル種別

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00045-E

Invalid extract master status file. (C)

抽出マスタステータスファイルの内容が不正です。

(S)処理を中断します。

(O)内部エラーです。

初期化直後の起動時にこのメッセージが出力された場合は、抽出側 Datareplicator の初期化が失敗しているおそれがあります。抽出マスタエラー情報ファイル、syslog ファイルにエラーメッセージが出力されている場合、メッセージに従ってエラーの原因を取り除いた後、初期化を再実行してください。メッセージが出力されていない場合や、上記以外の場合は、保守員に連絡してください。

KFRB00046-E

Ftok error, file name = aa...aa, errno = xx...xx. (C + S)

ファイルからのキーの作成に失敗しました。ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : ファイル名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00047-E

File lock operation error, file name = aa...aa, operation = bb...bb, errno = xx...xx. (C + S)

ファイルのロック操作に失敗しました。ファイル名 = aa...aa, 操作 = bb...bb, エラー番号 = xx...xx.

aa...aa : エラー発生ファイルの絶対パス名

bb...bb : ファイル操作

xx...xx : エラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00048-E

Process terminated due to node-master abnormal termination. (C)

抽出ノードマスタプロセスが異常終了したため、停止します。

(S)処理を中断します。

(O)抽出ノードマスタプロセスのエラーの原因を確認してください。

KFRB00049-E

Invalid extract coordinate file. (C)

データ連動用連絡ファイルが不正です。

(S)処理を中断します。

(O)内部エラーです。保守員に連絡してください。

KFRB00050-E

Extract data exceeds 32000byte, table id = aa...aa. (C + S)

更新情報が 32000 バイトを超えました。表 ID = aa...aa.

更新情報が 32000 バイトを超えました。反映側 Datareplicator のバージョンが 02-00 より前のバージョンの場合は、32000 バイトを超える更新情報を送信できません。

(S)処理を中断します。

(O)更新情報が 32000 バイト以下になるように抽出定義などを修正してください。hdeprep コマンドでエラーが発生した場合は、修正後、再実行してください。送信プロセスでエラーが発生した場合は、HiRDB Datareplicator 連携を再開始する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化し、抽出側 DB を基に反映側 DB を再作成してください。

KFRB00051-I

Error information/unreflected data file swapped, file kind = aa...aa, closed file = bb...bb. (L)

エラー情報ファイル 又は未反映情報ファイルをスワップしました。

aa...aa：スワップしたファイルの種別

errorfile：反映エラー情報ファイル又は抽出ノードエラー情報ファイル

unreffile：反映側未反映情報ファイル

msterrfile：抽出マスタエラー情報ファイル

bb...bb : スワップによってクローズしたファイルの名称

Datareplicator 識別子とデータ連動識別子は、..で出力します。

(S)処理を続行します。

(O)必要に応じて、ファイルのバックアップを取得してください。

KFRB00052-I

```
Error information/unreflected data file closed, file kind = aa...aa, closed file = bb...bb.  
(L)
```

エラー情報ファイル 又は未反映情報ファイルをクローズしました。

aa...aa : クローズしたファイルの種別

errorfile : 反映エラー情報ファイル 又は抽出ノードエラー情報ファイル

unreffile : 反映側未反映情報ファイル

msterrfile : 抽出マスタエラー情報ファイル

bb...bb : クローズしたファイルの名称

Datareplicator 識別子とデータ連動識別子は、..で出力します。

(S)処理を続行します。

(O)必要に応じて、ファイルのバックアップを取得してください。

KFRB00053-W

```
Error occurred in troubleshooting, process = aa...aa, resource = bb...bb, func = cc...cc, errno  
= dd...dd. (C)
```

トラブルシュート処理でエラーが発生しました。処理種別 = aa...aa, リソース名 = bb...bb, 関数名 = cc...cc, エラー番号 = dd...dd

トラブルシュート処理でエラーが発生しました。このエラーによってトラブルシュート情報の収集ができません。トラブルシュート情報を収集するためには、エラーの要因を取り除いてから、エラーとなったコマンドを再度実行してください。

aa...aa : 処理種別

Collect : トラブルシュート情報収集処理

Delete : 前回のトラブルシュート情報削除処理

Check : ファイル状態取得処理

bb...bb : リソース名

aa...aa が Collect の場合 : トラブルシュート情報ファイル名

aa...aa が Delete の場合 : トラブルシュート情報格納ディレクトリ名

aa...aa が Check の場合：次のどれかが表示されます。

- 抽出マスタステータスファイル名
- 抽出定義プリプロセスファイル名
- 抽出サーバステータスファイル名
- データ連動用連絡ファイル名
- 反映マスタステータスファイル名
- 反映ステータスファイル名

cc...cc：エラーが発生した関数名

dd...dd：errno に設定されたエラー番号

(S)処理を続行します。

[対策]エラーが発生した関数名称，及び errno の値から原因を調査し，対策してください。

次に代表的なエラーを示します。

関数名	errno の値	意味	対策
fopen	ENXIO (6)	ファイルに対するアクセス権限がありません。	bb...bb に対する書き込み権限を付与してから，再度実行してください。
	EAGAIN (11)	他プロセスが保持しているロックによって，ファイルへの操作ができません。	bb...bb に対するロックを解除後，再度実行してください。
	ENOMEM (12)	<p>UNIX 版の場合：</p> <p>次のどちらかの原因が考えられます。</p> <ul style="list-style-type: none"> • スワップ領域不足でプロセス固有メモリを確保できません。 • プロセス数が多過ぎるか，又は一部のプロセスが大量のメモリを消費しています。 <p>Windows 版の場合：</p> <p>次のどちらかの原因が考えられます。</p> <ul style="list-style-type: none"> • 仮想メモリ不足でプロセス固有メモリを確保できません。 • プロセス数が多過ぎるか，又は一部のプロセスが大量のメモリを消費しています。 	<p>UNIX 版の場合：</p> <p>スワップ領域が足りない場合は，スワップ領域を拡張してください。拡張できない場合は，不要なプロセスを停止させてください。</p> <p>一部のプロセスが大量のメモリを消費している場合は，該当するプロセスをいったん停止することを検討してください。</p> <p>Windows 版の場合：</p> <p>仮想メモリが足りない場合は，仮想メモリを拡張してください。拡張できない場合は，不要なプロセスを停止させてください。</p> <p>一部のプロセスが大量のメモリを消費している場合は，該当するプロセスをいったん停止することを検討してください。</p>
	EACCES (13)	ファイルに対するアクセス権限がありません。	bb...bb に対する書き込み権限を付与してから，再度実行してください。
	EEXIST (17)	同一名称のディレクトリが存在しています。	ディレクトリ bb...bb を削除後，再度実行してください。
	ENFILE	ファイルのオープン数がシステムの上限を超えました。	OS のオペレーティングシステムパラメタ（カーネルパラメタ）の，システムでオープンできる

関数名	errno の値	意味	対策
	(23)		ファイル最大数 (HP-UX の場合は nfile, AIX の場合は maxuproc×nofiles×固定ライセンス数, Linux の場合は NR_FILE 又は fs.file-max) の指定値を大きくしてください。
	EMFILE (24)	1 プロセス当たりのファイルオープン数の上限に達しました。	OS パラメタの 1 プロセス当たりのファイルオープン数の見直しを見直してください。
	ENOSPC (28)	ファイルを作成するための空き容量がありません。	ファイルシステム内の不要なファイルを削除するなどして, bb...bb のファイルを格納するための空き容量を大きくしてください。
stat	ENOMEM (12)	UNIX 版の場合： 次のどちらかの原因が考えられます。 <ul style="list-style-type: none"> スワップ領域不足でプロセス固有メモリを確保できません。 プロセス数が多過ぎるか, 又は一部のプロセスが大量のメモリを消費しています。 Windows 版の場合： 次のどちらかの原因が考えられます。 <ul style="list-style-type: none"> 仮想メモリ不足でプロセス固有メモリを確保できません。 プロセス数が多過ぎるか, 又は一部のプロセスが大量のメモリを消費しています。 	UNIX 版の場合： スワップ領域が足りない場合は, スワップ領域を拡張してください。拡張できない場合は, 不要なプロセスを停止させてください。 一部のプロセスが大量のメモリを消費している場合は, 該当するプロセスをいったん停止することを検討してください。 Windows 版の場合： 仮想メモリが足りない場合は, 仮想メモリを拡張してください。拡張できない場合は, 不要なプロセスを停止させてください。 一部のプロセスが大量のメモリを消費している場合は, 該当するプロセスをいったん停止することを検討してください。
	EACCES (13)	ファイルに対するアクセス権限がありません。	bb...bb に対する書き込み権限を付与してから, 再度実行してください。
mkdir	ENOENT (2)	HiRDB Datareplicator のインストール時に作成されるディレクトリ (/etc/hirdbds/) が存在しません (UNIX 版だけに発生)。	HiRDB Datareplicator を再インストールしてください。
	ENOMEM (12)	UNIX 版の場合： 次のどちらかの原因が考えられます。 <ul style="list-style-type: none"> スワップ領域不足でプロセス固有メモリを確保できません。 プロセス数が多過ぎるか, 又は一部のプロセスが大量のメモリを消費しています。 Windows 版の場合： 次のどちらかの原因が考えられます。 <ul style="list-style-type: none"> 仮想メモリ不足でプロセス固有メモリを確保できません。 プロセス数が多過ぎるか, 又は一部のプロセスが大量のメモリを消費しています。 	UNIX 版の場合： スワップ領域が足りない場合は, スワップ領域を拡張してください。拡張できない場合は, 不要なプロセスを停止させてください。 一部のプロセスが大量のメモリを消費している場合は, 該当するプロセスをいったん停止することを検討してください。 Windows 版の場合： 仮想メモリが足りない場合は, 仮想メモリを拡張してください。拡張できない場合は, 不要なプロセスを停止させてください。 一部のプロセスが大量のメモリを消費している場合は, 該当するプロセスをいったん停止することを検討してください。

関数名	errno の値	意味	対策
	EACCES (13)	ディレクトリに対するアクセス権がありません。	bb...bb に対する書き込み権限を付与してから、再度実行してください。
	ENOSPC (28)	ディレクトリを作成するための空き容量がありません。	ファイルシステム内の不要なファイルを削除するなどして、bb...bb のファイルを格納するための空き容量を大きくしてください。

KFRB00060-E

Error occurred in Oracle Call Interface function, func = aa...aa, return = bb...bb, errcode = cc...cc, info = dd...dd. (C + S)

OCI 関数内でエラーが発生しました。関数名 = aa...aa, 戻り値 = bb...bb, エラーコード = cc...cc, 詳細情報 = dd...dd.

aa...aa : OCI 関数名

bb...bb : 関数戻り値

cc...cc : Oracle エラーコード

dd...dd : 付加情報

(S)処理を終了します。

(O)Oracle エラーコードを参照して、エラーの原因を取り除いた後、再実行してください。なお、Oracle エラーコードが 1422 の場合、抽出対象 Oracle 上に複数のイベント制御用テーブル (hde_dtbl) が存在している可能性があります。イベント制御用テーブルが抽出対象 Oracle 上に一つだけ存在するように、残りのイベント制御用テーブルを削除してください。

KFRB00061-E

Unmatch extract DBMS kind, initial = aa...aa, current = bb...bb. (C + S)

抽出対象 DBMS が不一致です。初期化時 = aa...aa, 実行時 = bb...bb.

aa...aa : 初期化時の DBMS 種別コード

0 : HiRDB

1 : Oracle

2 : SQL Server

bb...bb : 実行時の DBMS 種別コード

0 : HiRDB

1 : Oracle

2 : SQL Server

(S)処理を終了します。

(O)初期化時の DBMS に対応するコマンドで再実行してください。

KFRB00062-E

```
Failed to load library, library name = aa...aa.    (E)
ライブラリのロードに失敗しました。ライブラリ名 = aa...aa
```

aa...aa：ライブラリ名

(S)処理を終了します。

(O)information に出力されるエラー情報を基に、エラーの原因を取り除いてください。information に出力される文字列の最大長は 254 バイトです。これを超えた場合、文字列の後ろから 254 バイトが出力されます。

KFRB00063-E

```
Failed to get procedure address, library name = aa...aa, procedure name = bb...bb.    (E)
ライブラリのロードに失敗しました。ライブラリ名 = aa...aa,関数名 = bb...bb
```

関数アドレスの取得に失敗しました。

aa...aa：ライブラリ名

bb...bb：関数名

(S)処理を終了します。

(O)information に出力されるエラー情報を基に、エラーの原因を取り除いてください。information に出力される文字列の最大長は 254 バイトです。これを超えた場合、文字列の後ろから 254 バイトが出力されます。

KFRB00064-E

```
Operation directory access error occurred, operation = nn...nn, directory = aa...aa, errno =
xx...xx.    (S+L)
運用ディレクトリにアクセスできません。操作種別 = nn...nn, 運用ディレクトリ名 = aa...aa, エラー
番号 = xx...xx
```

nn...nn：操作種別（システムコール名）

aa...aa：運用ディレクトリ名

xx...xx：エラー番号

(S)処理を終了します。

(O)次の対処をしてください。

- エラー番号が 2 の場合
環境変数 HDEPATH, 又は HDSPATH に設定されている運用ディレクトリ名を見直してください。
- エラー番号が 13 の場合
運用ディレクトリのアクセス権を見直してください。
その他のエラー番号については, errno.h 又は OS のマニュアルを参照してください。

KFRB00065-E

Shared memory operation failure, type = aa...aa, ID = bb...bb, operation = cc...cc, errno = xx...xx. (C+L)

共有メモリ操作に失敗しました。共有メモリ種別 = aa...aa, 共有メモリ ID = bb...bb, 操作種別 = cc...cc, エラー番号 = xx...xx.

aa...aa : 共有メモリ種別

- COMMON (反映共通共有メモリ)
- DEFSERV (反映定義解析共有メモリ)
- EXTCOM (抽出共通共有メモリ)
- EXTDEF (抽出定義解析共有メモリ)
- ERRTXT (メッセージテキスト共有メモリ)
- LOBSHM (BLOB 反映用共有メモリ)
- NMTCMD (抽出ノードマスター-コマンド間連絡用共有メモリ)
- STATINF (ステータス情報共有メモリ)

bb...bb : 共有メモリ ID

cc...cc : 操作種別

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して, エラーの原因を取り除き, 再度実行してください。

KFRB00066-W

Delay time exceeded definition value of aa...aa, bb...bb. (C+E)

滞留時間は, 定義の aa...aa 値を超えました。詳細 = bb...bb

滞留時間が定義値を超えました。

aa...aa : 定義オペランド

extract_delay_limit_time : 抽出滞留時間

send_delay_limit_time : 送信滞留時間

reflect_delay_limit_time : 反映滞留時間

bb...bb : 滞留時間

DelayTime[xxxx (時) :xx (分) :xx (秒)]の形式で出力します。

(S)処理を続行します。

(O)aa...aa に出力された情報に合わせた対処をしてください。

- extract_delay_limit_time (抽出滞留時間) の場合
滞留時間が大きい場合は、システムログ I/O バッファ、又は抽出用の抽出情報キュー I/O バッファを拡張してください。
- send_delay_limit_time (送信滞留時間) の場合
滞留時間が大きい場合は、抽出用の抽出情報キュー I/O バッファ、又は送信用の抽出情報キュー I/O バッファを拡張してください。
- reflect_delay_limit_time (反映滞留時間) の場合
滞留時間が大きい場合は、抽出側の更新データ量を圧縮してください。

KFRB00067-W

Delay time returned into the definition value of aa...aa. (C + E)

遅延時間は、定義の aa...aa 値内に戻りました。

滞留時間は、定義値内に戻りました。

aa...aa : 定義オペランド

extract_delay_limit_time : 抽出滞留時間

send_delay_limit_time : 送信滞留時間

reflect_delay_limit_time : 反映滞留時間

(S) 処理を続行します。

(O) KFRB00066-W メッセージに従って対処してください。

KFRB00068-E

Process check operation failure, operation = aa...aa, errno = bb...bb, process = cc...cc, pid = dd...dd. (C)

プロセスの存在確認に失敗しました。操作種別= aa...aa, エラー番号= bb...bb, プロセス名= cc...cc, プロセス ID = dd...dd

aa...aa : 操作種別

waitpid

kill

bb...bb : errno に設定されたエラー番号

cc...cc : プロセス名

dd...dd : プロセス ID

(S)チェックしたプロセスは停止しているものとみなして、処理を続行します。

KFRB00069-E

File size exceeded limit, file name = aa...aa, code = bb. (S + L)

指定したファイルのサイズは、ファイルシステムの制限値を超えました。ファイル名 = aa...aa, 詳細コード = bb

指定したファイルのサイズは、ファイルシステムの制限値を超えました。

aa...aa : ファイル名

bb : 詳細コード

1 : 指定したファイルのサイズは 2GB 以上ですが、ファイルシステムがラージファイルシステムになっていません。

2 : ファイルサイズの最大値を超えています。

(S)処理を中断します。

(O)詳細コードに従った対処をしてください。

- 詳細コード : 1

指定したファイルのファイルシステムがラージファイルに対応しているかどうかを、OS のシステム管理コマンドで確認してください。ラージファイルに対応していない場合、ラージファイルに対応するように設定するか、又はファイルサイズの最大値を 2GB 未満に設定してください。

OS のシステム管理コマンドについては、OS のマニュアルを参照してください。

- 詳細コード : 2

指定したファイルのサイズが、ファイルシステムで設定されている最大値を超えないように設定してください。設定できる最大値は OS ごとに異なるため、「6.11 ラージファイルの運用」で確認して設定してください。

KFRB00071-W

Message classification set error, information = aa...aa, code = xx...xx. (C)

メッセージ種別の設定に失敗しました。

aa...aa：保守情報 1

xx...xx：保守情報 2

注

インストール時に作成されるインストールディレクトリ下のファイルが壊れている場合に、このメッセージが出力されます。

(S)システムログのプログラム ID, 又はイベントログのソース名に HiRDB Datareplicator を仮定して、処理を続行します。

(O)HiRDB Datareplicator を再インストールしてください。

KFRB00080-W

Recover information file access error, function name = aa...aa, file name = bb...bb, errno = xx...xx. (C)

回復情報ファイルへのアクセスエラーが発生しました。関数名 = aa...aa, ファイル名 = bb...bb, エラー番号 = xx...xx

aa...aa：エラー発生関数名

bb...bb：ファイル名

xx...xx：errno にセットされたエラー番号

(S)処理を続行します。ただし、回復情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。
エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00081-W

Semaphore operation failure, function name = aa...aa, sem-id = bb...bb, errno = xx...xx. (C)

排他処理の操作に失敗しました。関数名 = aa...aa, セマフォ ID = bb...bb, エラー番号 = xx...xx

aa...aa：エラー発生関数名

bb...bb：セマフォ ID

xx...xx：errno にセットされたエラー番号

(S)処理を続行します。ただし、回復情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00082-W

Memory allocate error, function name = aa...aa, size = bb...bb, errno = xx...xx. (C)

領域確保処理中にメモリ不足が発生しました。関数名 = aa...aa, 確保領域サイズ = bb...bb, エラー番号 = xx...xx

aa...aa : エラー発生関数名

bb...bb : 確保領域サイズ

xx...xx : errno にセットされたエラー番号

(S)処理を続行します。ただし、回復情報は出力されません。

(O)出力されたメッセージを参照してエラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00083-E

Unable to load compression library, name = aa...aa, return code = bb...bb, errno = cc...cc. (C)

圧縮ライブラリがロードできません。ライブラリ名 = aa...aa, リターンコード = bb...bb, エラーコード = cc...cc.

aa...aa : 圧縮ライブラリ名称

- PDZLIB

bb...bb : リターンコード

4 : 表定義時に圧縮ライブラリ名で指定したライブラリ aa...aa が HiRDB にインストールされていません。

8 : 圧縮ライブラリ aa...aa のロード時にエラーが発生しました。

12 : 圧縮ライブラリのロードに成功しましたが、シンボルの解決でエラーが発生しました。

cc...cc : エラー番号 (errno)

bb...bb が 4 の場合 : ****

bb...bb が 8 の場合 : 圧縮ライブラリのロード時の errno

bb...bb が 12 の場合 : 圧縮ライブラリのシンボル解決時の errno

(S)処理を中断します。

(O)リターンコード, エラー番号を参照して, エラー原因を取り除き, 再度コマンドを実行してください。
エラー番号については, errno.h 及びユーザが使用する OS のマニュアルを参照してください。エラー原因を取り除けない場合は, 保守員に連絡してください。

KFRB00084-E

aa...aa return with invalid code bb...bb. (C)

圧縮ライブラリ aa...aa から不正な戻り値 bb...bb が返却されました。

aa...aa : 圧縮ライブラリ名称

- PDZLIB

bb...bb : 戻り値

圧縮ライブラリ aa...aa の戻り値

(S)処理を中断します。

(O)保守員に連絡してください。

KFRB00085-E

Length of Expand data aa...aa bb...bb, compression library name = cc...cc. (C)

圧縮ライブラリでエラーが発生しました。エラー情報 = Length of Expand data aa...aa bb...bb,
圧縮ライブラリ名 = cc...cc.

aa...aa

- 伸張後データ長
- lager than
- 伸張後データ長 less than compressed data length

bb...bb

- Invalid
- 伸張後データ格納領域サイズ
- 伸張前データ長

cc...cc : 圧縮ライブラリ名称

- PDZLIB

(S)処理を中断します。

(O)保守員に連絡してください。

KFRB00086-E

Expand processing error occurred, library = aa...aa, errinf = bb...bb. (C)

圧縮ライブラリ aa...aa の伸張処理で bb...bb で示すエラーが発生しました。

aa...aa : 圧縮ライブラリ名称

- PDZLIB

bb...bb : エラー情報

圧縮ライブラリ aa...aa の伸張処理で発生したエラー情報

(S)処理を中断します。

(O)下記のエラー情報に基づいてエラー原因を取り除き、再度コマンドを実行してください。エラー原因を取り除けない場合は、保守員に連絡してください。

- -1000 より小さい値 (-lnnn) の場合
-1000 より小さい値 : - (エラー情報 + 1000) の値 (nnn) を errno (1~151) として errno.h 及びユーザが使用している OS のマニュアルを参照し、エラー原因を取り除き再度コマンドを実行してください。
代表的な errno については、マニュアル「HiRDB メッセージ」を参照してください。
- -4 の場合
圧縮ライブラリの実行に必要なメモリ約 260KB を追加し、再度コマンドを実行してください。
- 上記以外の場合
保守員に連絡してください。

KFRB00100-I

Target site Datareplicator start completed. (C)

反映側 Datareplicator の起動を完了しました。

(S)処理を続行します。

KFRB00101-I

Reflection restart event was accepted. (C)

反映処理再起動イベントを受け付けました。

(S)処理を続行します。

KFRB00102-I

Command request was accepted. (C)

コマンド要求を受け付けました。

(S)処理を続行します。

KFRB00103-I

Accepted target site Datareplicator stop request. Requiring child processes to terminate. (C)

反映側 Datareplicator の終了処理を受け付けました。子プロセスに停止要求を行います。

(S)処理を続行します。

KFRB00104-I

Target site Datareplicator termination completed. (C)

反映側 Datareplicator の終了処理を完了しました。

(S)反映側 Datareplicator を終了します。

KFRB00150-E

Target site Datareplicator start error. (C)

反映側 Datareplicator の起動に失敗しました。

(S)反映側 Datareplicator を終了します。

(O)エラーの原因を取り除いた後、反映側 Datareplicator を再起動してください。

KFRB00151-E

Illegal replication node-id was specified. (C)

不正なデータ連動識別子が指定されました。

(S)コマンド要求を無視します。

(O)正しいデータ連動識別子を指定したコマンドを再実行してください。

KFRB00152-E

Unable to restore communication master process status, protocol = aa...aa. (C)

反映通信マスタプロセスの状態を回復できません。プロトコル種別 = aa...aa.

aa...aa : プロトコル種別

(S)該当する反映通信マスタプロセスの起動をスキップして、処理を続行します。

(O)反映ステータスファイルからの回復情報を読み込める状態にした後、システムを再起動してください。ただし、媒体に障害が発生した場合には、媒体を再び初期設定してください。

KFRB00153-E

Communication master process not exist, protocol = aa...aa. (C)

反映通信マスタプロセスがありません。プロトコル種別 = aa...aa.

aa...aa : プロトコル種別

(S)反映通信マスタプロセスを再起動します。

(O)再起動が失敗した場合は、反映通信マスタプロセスが起動できない要因を排除した後、反映側 Datareplicator を再起動してください。

KFRB00154-E

Definition server process not exist. (C)

反映定義サーバプロセスがありません。

(S)処理を続行します。

(O)反映処理を再起動してください。

KFRB00155-E

Communication master process retry started, protocol = aa...aa. (C)

反映通信マスタプロセスの再起動を開始しました。プロトコル種別 = aa...aa.

aa...aa : プロトコル種別

(S)処理を続行します。

KFRB00156-E

Communication master process retry error, protocol = aa...aa. (C)

反映通信マスタプロセスの再起動に失敗しました。プロトコル種別 = aa...aa.

aa...aa : プロトコル種別

(S)処理を続行します。

(O)反映通信マスタプロセスが起動できない要因を排除した後、反映側 Datareplicator を再起動してください。

KFRB00157-E

Reflection restart event acceptance error. (C)

反映処理再起動イベントの受け付けに失敗しました。

(S)イベントを無視します。

(O)内部エラーです。このメッセージの直前に出力されたエラーメッセージを参照してください。エラーメッセージが出力されていない場合は、保守員に連絡してください。

KFRB00158-E

Command acceptance error. (C)
コマンド要求の受け付けに失敗しました。

(S)要求を無視します。

(O)内部エラーです。このメッセージの直前に出力されたエラーメッセージを参照してください。エラーメッセージが出力されていない場合は、保守員に連絡してください。

KFRB00159-E

Definition server process is running. (C)
反映定義サーバプロセスは稼働中です。

(S)要求を無視します。

KFRB00160-E

Definition server process has been suspended. (C)
反映定義サーバプロセスは停止中です。

(S)要求を無視します。

KFRB00161-E

Failed to initialize the process, function name = aa...aa, errno = xx...xx (L)

プロセスの初期化に失敗しました。

aa...aa : 関数名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00162-W

Synchronous reflection group is running, group name = aa...aa. (C)

同期反映グループは稼動中です。グループ名 = aa...aa

aa...aa : 同期反映グループ名

(S)処理を続行します。

(O)hdsstate コマンドで同期管理プロセス (hdsrefsync) の稼働状態を確認して、同期管理プロセスが停止中の場合にだけ hdsrfctl コマンドを実行してください。

KFRB00163-W

Synchronous reflection group is not running, group name = aa...aa. (C)

同期反映グループは稼動していません。グループ名 = aa...aa

aa...aa : 同期反映グループ名

(S)処理を中断します。

(O)hdsstate コマンドで同期管理プロセス (hdsrefsync) の稼働状態を確認して、同期管理プロセスが stop 又は not active の場合にだけ hdsrfctl コマンドを実行してください。

KFRB00164-E

Synchronous reflection group doesn't exist or is canceled, group name = aa...aa. (C)

同期反映グループは存在しない、もしくはキャンセルされています。グループ名 = aa...aa

次のどれかの理由によって、hdsrfctl コマンドを受け付けられません。

- syncgroup001 オペランドを指定していない。
- syncgroup001 オペランドに指定していない同期反映グループ名を指定した。
- 指定した同期反映グループは、hdsstart -c コマンドによってキャンセルされている。

aa...aa : hdsrfctl に指定された同期反映グループ名

(S)処理を続行します。ただし、hdsrfctl コマンドは中断します。

(O)syncgroup001 オペランドに指定していない同期反映グループ名を指定した場合は、hdsrfctl コマンドの同期反映グループ名の指定をし直して、再度実行してください。

KFRB00400-E

Failed to connect service control manager, code = aa...aa, errno = xx...xx. (E)

サービス制御マネージャへの接続に失敗しました。コード = aa...aa, エラー番号 = xx...xx.

aa...aa : コード

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

なお、HiRDB Datareplicator Extension の Oracle 抽出機能、又は SQLServer 抽出機能をインストールした場合は、既にインストールされている HiRDB Datareplicator の抽出サービスが無効になります。その状態で hdestart コマンドを実行した場合、code=3, errno=1058 を出力します。

上記要因でこのエラーが発生した場合は、次のように対処してください。

- Oracle 抽出機能又は SQL Server 抽出機能を起動する場合
hdestart コマンドではなく、Oracle 抽出機能使用時は hdestartO を、SQL Server 抽出機能使用時は hdestartS コマンドを実行してください。
- HiRDB 抽出機能を起動する場合
「HiRDB Datareplicator (Source site)」サービスのスタートアップの種別を「手動」に戻すか、又は HiRDB Datareplicator Extension をアンインストールしてください。

KFRB00401-E

Failed to register service handler function, errno = xx...xx. (E)

サービスコントロールハンドラの登録に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00402-E

Failed to create event object, function name = aa...aa, event name = bb...bb, errno = xx...xx. (E)

イベントオブジェクトの作成に失敗しました。関数名 = aa...aa, イベント名 = bb...bb, エラー番号 = xx...xx.

aa...aa : 関数名

bb...bb : イベント名

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00403-E

```
Failed to open event object, function name = aa...aa, event name = bb...bb, errno =
xx...xx. (E)
```

イベントのオープンに失敗しました。関数名 = aa...aa, イベント名 = bb...bb, エラー番号 = xx...xx.

aa...aa : 関数名

bb...bb : イベント名

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00404-E

```
Failed to set state of event object to signaled, function name = aa...aa, event name =
bb...bb, errno = xx...xx. (E)
```

イベントの設定に失敗しました。関数名 = aa...aa, イベント名 = bb...bb, エラー番号 = xx...xx.

aa...aa : 関数名

bb...bb : イベント名

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00405-E

```
Failed to create thread to watch master, errno = xx...xx. (E)
```

マスタ監視スレッドの作成に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00406-E

Failed to create thread to watch receive master, errno = xx...xx. (E)

受信マスタ監視スレッドの作成に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00407-E

Failed to create thread to watch definition server, errno = xx...xx. (E)

定義サーバ監視スレッドの作成に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00408-E

Failed to watch master, errno = xx...xx. (E)

マスタの監視に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00409-E

Failed to watch receive master, errno = xx...xx. (E)

受信マスタの監視に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00410-E

Failed to watch definition server, errno = xx...xx. (E)

定義サーバの監視に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00411-E

Failed to duplicate handle, errno = xx...xx. (E)

ハンドルの複製に失敗しました。エラー番号 = xx...xx.

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00412-E

Failed to create named pipe, pipe name = aa...aa, errno = xx...xx. (E)

名称付きパイプの作成に失敗しました。パイプ名 = aa...aa, エラー番号 = xx...xx.

aa...aa : パイプ名

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00413-E

Failed to connect named pipe, pipe name = aa...aa, errno = xx...xx. (E)

名称付きパイプの接続に失敗しました。パイプ名 = aa...aa, エラー番号 = xx...xx.

aa...aa : パイプ名

xx...xx : GetLastError() で取得したエラー番号

(S) 処理を中止します。

(O) エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00414-E

Failed to open named pipe, pipe name = aa...aa, errno = xx...xx. (E)

名称付きパイプのオープンに失敗しました。パイプ名 = aa...aa, エラー番号 = xx...xx.

aa...aa : パイプ名

xx...xx : GetLastError() で取得したエラー番号

(S) 処理を中止します。

(O) エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00415-E

Target Site HiRDB Datareplicator service start failed. (S)

HiRDB Datareplicator Reflect サービスの起動に失敗しました。

反映サービスの開始に失敗しました。

(S) 処理を中止します。

(O) エラーの原因を取り除いた後、反映サービスを再開してください。

KFRB00416-E

Invalid setup information. (E)

セットアップ情報が不正です。

(S) 処理を中止します。

(O) Datareplicator を再インストールしてください。

KFRB00417-E

Failed to wait for event object, function name = aa...aa, event name = bb...bb, errno = xx...xx. (E)

イベントの待ちに失敗しました。

aa...aa：関数名

bb...bb：イベント名

xx...xx：GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)Datareplicator を再インストールしてください。

KFRB00418-E

Failed to retrieve the process termination status, process name = aa...aa, errno = xx...xx. (E)

プロセスの終了ステータスの取得に失敗しました。

aa...aa：プロセス名

xx...xx：GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00419-E

Failed to get master process handle, errno = xx...xx. (E)

抽出マスタプロセス、又は反映マスタプロセスのハンドルの取得に失敗しました。

xx...xx：GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00420-E

Source Site HiRDB Datareplicator service start failed. (S)

抽出サービスの開始に失敗しました。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、抽出サービスを再開始してください。

KFRB00421-E

Failed to load library, library name = aa...aa, errno = xx...xx. (C)

ライブラリのロードに失敗しました。ライブラリ名 = aa...aa, エラー番号 = xx...xx.

LoadLibrary()でのライブラリのロードに失敗しました。

aa...aa : ライブラリ名

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00422-E

Failed to get procedure address, procedure name = aa...aa, errno = xx...xx. (S)

関数アドレスの取得に失敗しました。関数名 = aa...aa, エラー番号 = xx...xx.

GetProcAddress()での関数アドレスの取得に失敗しました。

aa...aa : 関数名

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00423-E

Failed to generate Target Site HiRDB Datareplicator service stop process, errno = xx...xx. (S)

反映側 Datareplicator の停止プロセスの生成でエラーが起きました。

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00424-E

```
Failed to generate Source Site HiRDB Datareplicator service stop process, errno =  
xx...xx. (S)
```

抽出側 Datareplicator の停止プロセスの生成でエラーが起きました。

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00425-E

```
Failed to create child process, process name=aa...aa, errno = xx...xx. (C)
```

子プロセスの作成に失敗しました。

aa...aa : プロセス名

xx...xx : GetLastError()で取得したエラー番号

(S)処理を続行します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00426-E

```
Failed to create trace receive thread, errno = xx...xx. (C)
```

稼働トレース受信スレッドの作成に失敗しました。

xx...xx : GetLastError()で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00427-E

Failed to create timer watch thread, errno = xx...xx. (C)

タイマ監視スレッドの作成に失敗しました。

xx...xx : GetLastError() で取得したエラー番号

(S)処理を続行します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00428-E

Failed to watch trace receive thread, errno = xx...xx. (C)

稼働トレース受信スレッドの監視に失敗しました。

xx...xx : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00429-W

Activity trace collection process check failure, function name = aa...aa, errno = xx...xx. (C)

稼働トレース収集プロセスの存在の確認に失敗したため、このメッセージの出力以降、稼働トレース情報を取得できません。

aa...aa : エラーになった関数名

xx...xx : GetLastError() で取得したエラー番号

(S)処理を続行します。

(O)稼働トレース情報を取得したい場合は Datareplicator を停止し、エラーになった関数名と GetLastError() で取得したエラー番号を基に、OS のマニュアルを参照してエラー原因を調査してください。エラー原因を取り除いた後、Datareplicator を再起動してください。

KFRB00430-W

Activity trace collection process does not exist. (C)

稼働トレース収集プロセスが存在しないため、このメッセージの出力以降、稼働トレース情報を取得できません。

(S)処理を続行します。

(O)このメッセージより前に稼働トレース収集プロセスでエラーメッセージが出力されている場合は、そのエラーの対処法に従って対処してください。このメッセージより前に稼働トレース収集プロセスでエラーメッセージが出力されていない場合は、保守員に連絡してください。

KFRB00499-E

```
Fatal error occurred. xx...xx. (C)
```

致命的なエラーが発生しました。

xx...xx：エラー情報

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00501-I

```
Command request was accepted. request = aa...a. (C)
```

コマンド要求を受け付けました。 要求種別 = aa...a

aa...a：要求種別

INIT：初期化要求

START：起動要求

NODE_START：ノード単位起動要求

STOP：停止要求

NODE_STOP：ノード単位停止要求

STATE：状態取得要求

CHG_STATUS：状態更新要求

(S)処理を続行します。

KFRB00502-I

```
Source site Datareplicator start completed. (C)
```

抽出側 Datareplicator の起動処理を完了しました。

(S)処理を続行します。

KFRB00503-I

Source site Datareplicator termination completed. (C)
抽出側 Datareplicator の終了処理を完了しました。

(S)抽出側 Datareplicator を終了します。

KFRB00504-I

Source site Datareplicator initialization completed. (C)
抽出側 Datareplicator の初期化処理を完了しました。

(S)抽出側 Datareplicator を終了します。

KFRB00505-I

Command request completed. (C)
コマンド要求処理を完了しました。

(S)処理を続行します。

KFRB00506-W

Source site Datareplicator start is incomplete. (C)
抽出側 Datareplicator の起動処理が不完全です。

(S)処理を続行します。

(O)起動していないユニットのエラー情報を確認してエラーの原因を取り除いた後、再起動してください。

KFRB00507-W

Source site Datareplicator termination is incomplete. (C)
抽出側 Datareplicator の終了処理が不完全です。

(S)処理を続行します。

(O)正常に終了できなかったユニットのエラー情報を確認してください。

KFRB00508-W

Source site Datareplicator initialization is incomplete. (C)
抽出側 Datareplicator の初期化処理が不完全です。

(S)抽出側 Datareplicator を終了します。

(O)初期化が完了していないユニットのエラー情報を確認してエラーの原因を取り除いた後、hdestart -i コマンドを再び初期化してください。

KFRB00509-W

Command request is incomplete. (C)

コマンド要求処理が不完全です。

(S)処理を続行します。

(O)コマンドの要求を完了していないユニットのエラー情報を確認してエラーの原因を取り除いた後、再実行してください。

KFRB00510-I

Terminate source site Datareplicator. (C)

抽出側 Datareplicator を終了します。

(S)抽出側 Datareplicator を終了します。

KFRB00511-I

Preprocess file format conversion started, old format version = aa...aa, new format version = bb...bb. (C)

プリプロセスファイルのフォーマット変換を開始しました。旧フォーマットバージョン = aa...aa, 新フォーマットバージョン = bb...bb.

aa...aa : 旧フォーマットバージョン

bb...bb : 新フォーマットバージョン

(S)処理を続行します。

KFRB00512-W

-S option value of hdestart command is invalid, value = aa...aa. (C)

hdestart コマンドの-S オプションで指定した値が不正です。

aa...aa : -S オプションで指定した送信先識別子番号

(S)処理を中止します。

(O)現在定義されている送信先数より大きい値が指定されているため、値を変更した後に再実行してください。

KFRB00513-W

```
-S option value of hdestart command is not equal to send number of incomplete initialization, previous value = aa...aa, current value = bb...bb. (C)
```

hdestart コマンドの-S オプションで指定した値は、未完了となった部分初期化で指定されていた送信先識別子番号と異なります。

aa...aa : 前回指定した送信先識別子番号

bb...bb : 今回指定した送信先識別子番号

(S)処理を中止します。

(O)前回指定した送信先識別子番号を指定して部分初期化を再実行するか、抽出側 Datareplicator 全体を初期化してください。

KFRB00514-I

```
Connection request from Node-master process was accepted. (C)
```

抽出ノードマスタープロセスからの接続要求を受け付けました。

(S)処理を続行します。

KFRB00515-W

```
Unable to get connection information from Node-master process. (C)
```

抽出ノードマスタープロセスからの接続情報の取得に失敗しました。

このメッセージの出力する時点では、抽出ノードマスタープロセスから接続ホスト名を取得していないため、前後に出力する送受信エラーメッセージのホスト名は"UNKNOWN HOST"を表示します。

(S)処理を続行します。

(O)このメッセージの直前に出力されている受信エラーメッセージに従って対処してください。

KFRB00516-W

```
Invalid connection information from Node-master process, hostname = aa...aa, reason = bb...bb. (C)
```

抽出ノードマスタープロセスからの接続要求が不正です。

aa...aa : 接続ホスト名

bb...bb : 理由コード

1. nodecontrol オペランドで指定している抽出ノードマスタプロセスの制御単位と接続要求のあった抽出ノードマスタプロセスの制御単位が不一致です。
2. 存在しない接続対象名で、抽出ノードマスタプロセスから接続要求がありました。
3. 抽出マスタプロセスが送信プロセス起動要求によって起動されたにもかかわらず、抽出ノードマスタプロセスから更新 SQL 出力プロセス起動での接続要求がありました。
4. 抽出マスタプロセスが更新 SQL 出力プロセス起動要求によって起動されたにもかかわらず、抽出ノードマスタプロセスから送信プロセス起動での接続要求がありました。
5. 抽出ノードマスタプロセスとの接続状態チェック処理でエラーを検出しました。
6. ノード単位初期化処理でエラーを検出しました。
7. 抽出ノードマスタプロセスに対する定義情報送信処理でエラーを検出しました。

(S)処理を続行します。

(O)次に示す表に従って対処してください。

理由コード	意味	対策
1	nodecontrol オペランドで指定している抽出ノードマスタプロセスの制御単位と接続要求のあった抽出ノードマスタプロセスの制御単位が不一致です。	nodecontrol オペランドが unit の場合、抽出ノードマスタプロセスを個別に起動できません。抽出ノードマスタプロセスを起動する場合は、抽出マスタプロセスに対して hdestart コマンドを実行してください。
2	存在しない接続対象名で、抽出ノードマスタプロセスから接続要求がありました。	hdestart_n コマンドに正しいサーバ名を指定した後、コマンドを再実行してください。
3	抽出マスタプロセスが送信プロセス起動要求によって起動されたにもかかわらず、抽出ノードマスタプロセスから更新 SQL 出力プロセス起動での接続要求がありました。	更新 SQL 出力処理を再実行する場合、抽出 Datareplicator をいったん停止した後に、抽出マスタプロセスに対して hdestart コマンドの -s オペランドで -L を指定してください。
4	抽出マスタプロセスが更新 SQL 出力プロセス起動要求によって起動されたにもかかわらず、抽出ノードマスタプロセスから送信プロセス起動での接続要求がありました。	
5	抽出ノードマスタプロセスとの接続状態チェック処理でエラーを検出しました。	このメッセージの直前に出力されているエラーメッセージに従って対処してください。
6	ノード単位初期化処理でエラーを検出しました。	
7	抽出ノードマスタプロセスに対する定義情報送信処理でエラーを検出しました。	このメッセージの直前に出力されているエラーメッセージに従って対処してください。 抽出ノードマスタプロセスがステータスファイルのオープンエラーとなっている場合、hdestart_n コマンドの実行ユニットが不正の可能性があります。正しいユニットでコマンドを再実行してください。

KFRB00517-W

```
Node-master process is already connected, target = aa...aa, current hostname = bb...bb,  
request hostname = cc...cc. (C)
```

接続要求のあった抽出ノードマスタープロセスは既に接続済みです。

aa...aa：接続対象名

bb...bb：現時点で接続中のホスト名

cc...cc：新たに接続要求のあったホスト名

(S)処理を続行します。

KFRB00518-I

```
Terminate Extract master process, because connection wait time from Node-master process  
was over. (C)
```

抽出ノードマスタープロセスからの接続待ち時間に達したため、抽出マスタープロセスを終了します。

(S)システムを終了します。

KFRB00519-I

```
Connection request completed, target = aa...aa, connect hostname = bb...bb. (C)
```

抽出ノードマスタープロセスからの接続処理を完了しました。

aa...aa：接続対象名

bb...bb：接続中ホスト名

(S)処理を続行します。

KFRB00551-I

```
Extract master process request was accepted, request = aa...a. (C)
```

抽出マスタープロセスからの要求を受け付けました。 要求種別 = aa...a

aa...a：要求種別

INITENV：環境作成要求

START：起動要求

STOP：停止要求

STATE：状態取得要求

CHG_STATUS：状態更新要求

(S)処理を続行します。

KFRB00552-I

Node-master process start completed. (C)

抽出ノードマスタプロセスの起動を完了しました。

(S)処理を続行します。

KFRB00553-I

Node-master process termination completed. (C)

抽出ノードマスタプロセスの終了処理を完了しました。

(S)処理を続行します。

KFRB00554-I

Node-master process initialization completed. (C)

抽出ノードマスタプロセスの初期化処理を完了しました。

(S)該当するノードの処理を終了します。

KFRB00555-I

Extract master process request completed. (C)

抽出マスタプロセスの要求処理を完了しました。

(S)処理を続行します。

KFRB00556-W

Extract master process request is incomplete. (C)

抽出マスタプロセスの要求処理が不完全です。

(S)処理を続行します。

(O)事前に出力されているメッセージの情報を確認してください。

KFRB00557-I

Terminate node-master process. (C)

抽出ノードマスタプロセスを終了します。

(S)該当するノードの処理を終了します。

KFRB00558-W

Connection was closed, Node-master process retry connect to Extract master process. (C + L)

回線が切断されました。抽出マスタプロセスへの回線の再接続を行います。

(S)処理を続行します。

(O)通信エラーとなった理由を、事前に出力されているメッセージによって確認してください。また、抽出マスタプロセスの状態を確認し、必要であれば抽出マスタプロセスを再起動してください。

KFRB00559-W

Connection wait time to Extract master process was over. (C + L)

抽出マスタプロセスへの接続待ち時間に到達しました。

(S)該当するノードの処理を中断します。

(O)接続エラーとなった理由を、事前に出力されているメッセージによって確認してください。

KFRB00560-E

Connection request to Extract master process was not accepted, message number = nnnnn. (C + L)

抽出マスタプロセスへの接続要求が受け付けられませんでした。メッセージ番号 = nnnnn.

nnnnn : 抽出マスタプロセスで発生したエラーのメッセージ番号

nnnnn は、Datareplicator が出力するメッセージ ID の KFRBnnnnn-T に対応します。

(S)該当するノードの処理を中断します。

(O)メッセージ番号 nnnnn に対応するメッセージ KFRBnnnnn-T の説明を参照してください。

KFRB00561-E

Node-master process connected to Extract master process, host name = aa...aa, port number = bb...bb. (C + L)

抽出マスタプロセスへの接続処理を完了しました。接続ホスト名 = aa...aa, ポート番号 = bb...bb.

aa...aa : 接続ホスト名

bb...bb : -ポート番号

(S)処理を続行します。

aa...aa Function applied, code = xx. (C)

aa...aa 機能を適用します。

aa...aa : 次に示すどちらかの適用する機能

High-availability :

syslogfile の信頼性向上

High-availability and Code conversion :

syslogfile の信頼性向上, 及び syslogfile の文字コード変換 (SJIS を UTF-8 に変換)

xx : 詳細コード

01 : syslogfile の信頼性向上, 及び syslogfile の文字コード変換 (SJIS を UTF-8 に変換) 機能を使用します。

02 : 次のどちらかの理由で, syslogfile の文字コード変換を使用しないで, syslogfile の信頼性向上だけを実現します。

- 日立コード変換がインストールされていません。
- 日立コード変換の初期化に失敗しました。メモリ不足のおそれがあります。

03 : Datareplicator の抽出システム定義, 又は反映システム定義の dblocale オペランド, 及び msglocale オペランドに指定した値は適切ですが, 拡張 SYSLOG 機能がコード変換機能に対応していないため, syslogfile の文字コード変換を使用しません。syslogfile の信頼性向上だけを実現します。

04 : Datareplicator の抽出システム定義, 又は反映システム定義の dblocale オペランド, 及び msglocale オペランドに指定した値が, コード変換機能で指定できる組み合わせではないため, syslogfile の文字コード変換を使用しません。syslogfile の信頼性向上だけを実現します。

(S)処理を続行します。

[対策]

syslogfile の文字コード変換に対応している OS で, syslogfile の文字コード変換を使用したい場合の, 詳細コードと対処方法を次に示します。

詳細コード	対処方法
01	対処は不要です。
02	日立コード変換がインストールされていない場合 Datareplicator を停止し, 日立コード変換をインストールしてから Datareplicator を再開始してください。 日立コード変換がインストールされている場合 <ul style="list-style-type: none"> • メモリ不足のときは, メモリ不足を解消し, Datareplicator を再開始してください。 • 上記以外のときは, Datareplicator を停止し, 日立コード変換を再インストールしてから Datareplicator を再開始し

詳細コード	対処方法
	てください。解消されない場合は、保守員に連絡してください。
03	Datareplicator を停止し、文字コード変換機能に対応している拡張 SYSLOG 機能をインストールし、再度 Datareplicator を開始してください。
04	Datareplicator を停止し、Datareplicator の抽出システム定義、又は反映システム定義の dblocale オペランド、及び msglocale オペランドの指定値を、文字コード変換機能が適用できる組み合わせに修正してから Datareplicator を開始してください。

KFRB00581-I

Interrupted request of forceful termination. (C)
強制終了要求に割り込まれました。

(S)強制終了処理で抽出側 Datareplicator を終了します。

KFRB00582-I

Communicable node-master process not exist. (C)
通信可能な抽出ノードマスタープロセスが存在しません。

(S)抽出側 Datareplicator を終了します。

KFRB00583-I

Capture process and Sender process not exist. (C)
抽出プロセス及び送信プロセスが存在しません。

(S)抽出ノードマスタープロセスを終了します。

KFRB00584-I

Status update completed, server name = aa...a (C)
状態の更新に成功しました。サーバ名 = aa...a

メッセージ中に出力されているサーバに関する状態の更新に成功しました。

aa...aa：サーバ名

(S)処理を続行します。

KFRB00601-E

```
Create socket failed, errno = xx...xx.    (C)  
ソケットの生成に失敗しました。エラー番号 = xx...xx.
```

xx...xx : errno に設定されたエラー番号

(S)該当するノードの処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00602-E

```
Unable to get host-information, hostname = aa...aa, h_errno = xx...xx.    (C)  
ホスト情報の取得に失敗しました。ホスト名 = aa...aa, エラー番号 = xx...xx.
```

aa...aa : ホスト名

xx...xx : errno に設定されたエラー番号

(S)該当するノードの処理を中断します。

(O)aa...aa のサービスが、OS の services ファイルに正しく登録されていることを確認してください。

KFRB00603-E

```
Connection failed, hostname = aa...aa, errno = xx...xx.    (C)  
コネクション要求に失敗しました。ホスト名 = aa...aa, エラー番号 = xx...xx.
```

aa...aa : ホスト名

xx...xx : errno に設定されたエラー番号

(S)該当するノードの処理を中断します。

(O)抽出マスタプロセスと抽出ノードマスタプロセス間のコネクション確立要求時にエラーを検知しました。

エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

xx...xx が 111 (Linux), 239 (HP-UX), 79 (AIX) の場合、以下の要因が考えられます。

- 抽出システム定義の node_host オペランドに指定したホスト名が誤っている。
この場合、正しいホスト名に修正し、抽出側 Datareplicator を再起動してください。
- 抽出システム定義の mstservice オペランドに指定したサービス名の inetd.conf ファイルへの登録が行われていない。

この場合、inetd.conf ファイルへのサービス名の登録、および inetd プロセスの再起動を行った後、抽出側 Datareplicator を再起動してください。

xx...xx が 110 (Linux), 238 (HP-UX), 70 (AIX), 10060 又は 10061 (Windows) の場合

- 抽出システム定義の node_host オペランドに指定したホスト名が誤っている。
この場合、正しいホスト名に修正し、抽出側 Datareplicator を再起動してください。

KFRB00604-E

```
Send error, hostname = aa...aa, errno = xx...xx.    (C)
送信処理に失敗しました。ホスト名 = aa...aa, エラー番号 = xx...xx.
```

aa...aa : ホスト名

xx...xx : errno に設定されたエラー番号

(S)該当するノードの処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00605-E

```
Receive error, hostname = aa...aa, errno = xx...xx.  (C)
受信処理に失敗しました。ホスト名 = aa...aa, エラー番号 = xx...xx.
```

抽出マスタプロセスと抽出ノードマスタプロセス間で通信が失敗しました。

aa...aa : ホスト名

xx...xx : errno に設定されたエラー番号

(S)該当するノードの処理を中断します。

(O)エラー番号が次のどれかの場合、一時的な障害のおそれもあります。

hdestart コマンドを実行し、抽出側 Datareplicator を再起動してください。

- ECONNRESET
errno の値が HP-UX の場合は 232, AIX の場合は 73, Linux の場合は 104, Windows の場合は 10054
- ETIMEDOUT
errno の値が HP-UX の場合は 238, AIX の場合は 78, Linux の場合は 110, Windows の場合は 10060
- ECONNABORTED
Windows の場合だけに発生し、errno の値が 10053

エラーが解消されない場合、ネットワーク障害の対処をした後、hdestart コマンドを実行し、抽出側 Datareplicator を再起動してください。

KFRB00606-E

Send timeout, hostname = aa...aa. (C)

通信待ち時間内に送信できませんでした。ホスト名 = aa...aa.

aa...aa : ホスト名

(S)該当するノードの処理を中断します。

(O)抽出ノードマスタープロセスが起動しているか、又はホスト間の通信に問題が発生していないかを確認してください。

KFRB00607-E

Receive timeout, hostname = aa...aa. (C)

通信待ち時間内に受信できませんでした。ホスト名 = aa...aa.

aa...aa : ホスト名

(S)該当するノードの処理を中断します。

(O)抽出システム定義の watchintvl オペランドに指定したエラー監視間隔での抽出マスタープロセスと抽出ノードマスタープロセス間の通信処理で、タイムアウトを検知しました。以下の要因が考えられます。

- 系切り替えにより、aa...aa で示すホストの IP アドレスが一時的に非活性になっている。
この場合、系切り替え完了後に自動的に再接続されるため、対処は不要です。エラーは無視してください。
- 系切り替え以外の要因で、aa...aa で示すホストの IP アドレスが非活性になっている。
この場合、IP アドレスを活性化した後、抽出側 Datareplicator を再起動してください。
- aa...aa で示すホストで Datareplicator のプロセスがエラー停止している。
この場合、エラー監視先で出力されるエラーメッセージを確認し、対処してください。

KFRB00608-E

Found end of transmission, hostname = aa...aa. (C)

通信終了を検知しました。ホスト名 = aa...aa.

aa...aa : ホスト名

(S)該当するノードの処理を中断します。

(O)抽出ノードマスタプロセスが起動しているか、又はホスト間の通信に問題が発生していないかを確認してください。

KFRB00609-E

Unable to get service-information, service name = aa...aa, errno = xx...xx. (E)

サービス情報を取得できません。 サービス名 = aa...aa, エラー番号 = xx...xx

aa...aa : サービス名

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00610-E

Bind system call error, errno = xx...xx. (E)

bind システムコールでエラーが発生しました。 エラー番号 = xx...xx

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00611-E

Listen system call error, errno = xx...xx. (C)

listen システムコールでエラーが発生しました。 エラー番号 = xx...xx

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00612-E

Accept system call error, errno = xx...xx. (C)

accept システムコールでエラーが発生しました。 エラー番号 = xx...xx

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00613-E

Setsockopt system call error, errno = xx...xx. (C)

ソケットオプションの設定でエラーが発生しました。エラー番号 = xx...xx

setsockopt システムコールでエラーが発生しました。

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00614-E

Select system call error, errno = xx...xx. (C)

select システムコールでエラーが発生しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00700-W

Warning occurred on node-master process, hostname = aa...aa, server name = bb...bb, senderid = cc...cc, message number = nnnnn. (C + E)

抽出ノードマスタプロセスで警告が発生しています。ホスト名 = aa...aa, サーバ名 = bb...bb, 送信先識別子 = cc...cc, メッセージ番号 = nnnnn

抽出ノードマスタプロセスで警告が発生しています。

aa...aa : ホスト名

bb...bb : サーバ名

cc...cc : 送信先識別子

nnnnn : 抽出ノードマスタプロセスで発生した警告のメッセージ番号

nnnnn は、Datareplicator が出力するメッセージ ID の KFRBnnnnn-T に対応します。

(S) メッセージ番号 nnnnn に対応するメッセージ KFRBnnnnn-T の説明を参照してください。

(O) メッセージ番号 nnnnn に対応するメッセージ KFRBnnnnn-T の説明を参照してください。

KFRB00701-E

Error occurred on node-master process, hostname = aa...aa, server name = bb...bb, senderid = cc...cc, message number = nnnnn. (C)

抽出ノードマスタプロセスでエラーが発生しています。ホスト名 = aa...aa, サーバ名 = bb...bb, 送信先識別子 = cc...cc, メッセージ番号 = nnnnn.

aa...aa : ホスト名

bb...bb : サーバ名

cc...cc : 送信先識別子

nnnnn : 抽出ノードマスタプロセスで発生したエラーのメッセージ番号

nnnnn は、Datareplicator が出力するメッセージ ID の KFRBnnnnn-T に対応します。

(S) メッセージ番号 nnnnn に対応するメッセージ KFRBnnnnn-T の説明を参照してください。

(O) メッセージ番号 nnnnn に対応するメッセージ KFRBnnnnn-T の説明を参照してください。

KFRB00702-E

Missing necessary file, file name = aa...aa, file-kind = bb...bb. (C)

必要なファイルが存在しません。ファイル名 = aa...aa, ファイル種別 = bb...bb.

aa...aa : ファイル名

bb...bb : ファイル種別

EXTDEF (抽出定義プリプロセスファイル)

MSTSTATFILE (抽出マスタステータスファイル)

NMTSTATFILE (抽出サーバステータスファイル)

ERRFILE (抽出側のエラー情報ファイル)

ERRTXT (メッセージテキストファイル)

(S) 処理を中断します。

(O)ファイルを用意して、再実行してください。

メッセージテキストファイルはインストール時に作成されるファイルです。この場合、再インストールが必要です。

KFRB00703-E

File initialization error, file name = aa...aa, file-kind = bb...bb, errno = xx...xx. (C+S)
ファイル初期化処理中にエラーが発生しました。ファイル名 = aa...aa, ファイル種別 = bb...bb, エラー番号 = xx...xx.

aa...aa : ファイル名

bb...bb : ファイル種別

MSTSTATFILE (抽出マスタステータスファイル)

NMTSTATFILE (抽出サーバステータスファイル)

QUEFILE (抽出情報キューファイル)

CNCTFILE (データ連動用連絡ファイル)

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00704-E

Invalid process communication, hostname = aa...aa. (C)
不正なプロセスと通信しています。ホスト名 = aa...aa.

aa...aa : ホスト名

(S)該当するノードの処理を中断します。

(O)ネットワークの設定に誤りがないか確認してください。

KFRB00705-E

Node-master process is running. (C)
抽出ノードマスタプロセスは稼働中です。

(S)要求を無視します。

KFRB00706-E

Invalid request was accepted, hostname = aa...aa. (C)

不正な要求を受け付けました。ホスト名 = aa...aa.

aa...aa : ホスト名

(S)該当するノードの処理を中断します。

(O)内部エラーです。保守員に連絡してください。

KFRB00708-E

Process terminated due to inner error, process = aa...aa. (C)

プロセスが異常終了しています。プロセス名 = aa...aa.

aa...aa : プロセス名

(S)該当するノードの処理を中断します。

(O)内部エラーです。保守員に連絡してください。

KFRB00709-E

Missing necessary environment variable value, variable name = aa...aa. (C)

環境変数値が取得できません。環境変数名 = aa...aa.

aa...aa : 環境変数名

(S)処理を中断します。

(O)環境変数を設定してください。

KFRB00710-E

Invalid environment variable value, variable name = aa...aa. (C)

環境変数値が不正です。環境変数名 = aa...aa.

aa...aa : 環境変数名

(S)処理を中断します。

(O)環境変数の設定値を確認して、修正してください。

KFRB00711-E

Invalid server name specification, server name = aa...aa. (C)

コマンドで不正なサーバ名を指定しています。サーバ名 = aa...aa.

aa...aa : サーバ名

(S)不正なサーバ名に対する処理だけを中断します。

(O)引数を正しく指定して、再実行してください。

KFRB00712-E

Invalid sender-id specification, sender-id = aa...aa. (C)

コマンドで不正な送信先識別子を指定しています。送信先識別子 = aa...aa.

aa...aa : 送信先識別子

(S)不正な識別子に対する処理だけを中断します。

(O)引数を正しく指定して、再実行してください。

KFRB00713-E

Unable to load preprocess file of extract definition, because preprocess file is older than extract master status file. (C)

抽出定義プリプロセスファイルが抽出マスタステータスファイルよりも古いため、抽出定義プリプロセスファイルを読み込むことができません。

(S)処理を中断します。

(O)hdeprep コマンドを実行して、抽出定義プリプロセスファイルを再作成してください。

KFRB00714-E

Unable to use file system, file system = aa...aa, errcode = bb...bb. (L)

ファイルシステム領域を使用できません。ファイルシステム領域名 = aa...aa, 詳細コード = bb...bb.

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中断します。

(O)Datareplicator ファイルシステム領域に指定したファイルが初期化済みかどうかを確認した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードを次に示します。

詳細コード	意味
1	キャラクタ型スペシャルファイル以外のファイル名を指定しました。
2	Datareplicator ファイルシステム領域が初期化されていません。

詳細コード	意味
3	Datareplicator ファイルシステム領域は、既に初期化してあります。
4	Datareplicator ファイルシステム領域は、使用状態（停止中）です。
5	Datareplicator ファイルシステム領域は、使用状態（稼働中）です。
6	Datareplicator ファイルシステム領域に、同じ名称のファイルを複数登録しようとしています。
7	該当する Datareplicator ファイルシステム領域の最大ファイル数を超過して、ファイルを格納しようとしています。
8	該当する Datareplicator ファイルシステム領域のサイズを超過して、ファイルを格納しようとしています。 [3.4.3 Datareplicator ファイルシステム領域の構造]にある領域の構造図を参照して、Datareplicator ファイルシステム領域のサイズを見積もり直してください。
9	使うキャラクタ型スペシャルファイルのセクタ数が足りません。 [3.4.3 Datareplicator ファイルシステム領域の構造]にある領域の構造図を参照して、Datareplicator ファイルシステム領域のサイズを見積もり直してください。
10	システムコールエラーになりました。次に示すことが考えられます。 <ul style="list-style-type: none"> • Datareplicator ファイルシステム領域が存在しません。 • Datareplicator ファイルシステム領域が破壊されています。
11	リンクファイルの作成に失敗しました。次に示すことが考えられます。 <ul style="list-style-type: none"> • Datareplicator ファイルシステム領域に格納しようとしているファイルが、既にファイルとして存在しています。 • リンクファイルを作成するパスが存在しない、あるいはアクセス権がありません。
12	メモリが不足しました。
13	使うキャラクタ型スペシャルファイルのセクタ長が足りません。256 バイト以上必要です。

KFRB00715-E

Unable to initialize file system, file system = aa...aa, errcode = bb...bb. (L)
ファイルシステム領域の初期化に失敗しました。ファイルシステム領域名 = aa...aa, 詳細コード = bb...bb.

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中断します。

(O)Datareplicator ファイルシステム領域に指定したファイルを hdsfmkfs コマンドで初期化した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB00716-E

Unable to prepare file system, file system = aa...aa, errcode = bb...bb. (L)

ファイルシステム領域の準備に失敗しました。ファイルシステム領域名 = aa...aa, 詳細コード = bb...bb.

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中断します。

(O)Datareplicator ファイルシステム領域に指定したファイルが使えるかどうかを確認した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB00717-E

Unable to add file system, file system = aa...aa, add file = bb...bb, add size = cc...cc, errcode = dd...dd. (L)

ファイルシステム領域にファイルを追加できません。ファイルシステム領域名 = aa...aa, 追加ファイル名 = bb...bb, 追加ファイルサイズ = cc...cc, 詳細コード = dd...dd.

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域に追加するファイル名

cc...cc : 追加するファイルのサイズ

dd...dd : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中断します。

(O)Datareplicator ファイルシステム領域に格納するファイルのサイズを確認した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB00718-E

Unable to use both UNIX regular file and Character special file. (S)

UNIX ファイルとキャラクタスペシャルファイルを混在して使用することはできません。

(S)処理を中断します。

(O)ファイル種別の混在を修正した後、再実行してください。

KFRB00719-E

Source site database is not supported HiRDB, HiRDB = aa...aa, Datareplicator = bb...bb. (L)

抽出元データベースは前提ではない HiRDB です。HiRDB = aa...aa, Datareplicator = bb...bb.

aa...aa : 抽出元 HiRDB のアドレスモード

bb...bb : 抽出側 Datareplicator のアドレスモード

(S)処理を中断します。

(O)抽出元になる HiRDB を前提とする Datareplicator がインストールされているかどうかを確認してください。

KFRB00720-E

Failed to get information of source site database, function name = aa...aa, errno = bb...bb. (L)

抽出元データベースの情報取得に失敗しました。関数名 = aa...aa, エラー番号 = bb...bb.

aa...aa : 関数名

bb...bb : エラー番号

(S)処理を中断します。

(O)内部エラーです。エラーの原因を取り除いた後、再実行してください。エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00721-E

Unable to get current host name, target name = aa...aa, information = bb...bb. (C)

現用ホスト名の取得に失敗しました。対象 = aa...aa, 情報 = bb...bb

aa...aa : 対象名

bb...bb : エラー情報

(S)処理を中断します。

(O)エラー情報で原因を取り除いた後、再実行してください。

なお、エラー情報に"Unable to get host name. rtncode=-1854"が出力されている場合は、抽出対象 HiRDB を起動した後、再実行してください。

KFRB00722-E

Command request is invalid, reason = aa...aa. (C)

コマンド要求が不正です。理由 = aa...aa

aa...aa : 理由コード

Unmatch nodecontrol operand : nodecontrol オペランドとコマンドが不一致です。

(S)処理を中断します。

(O)nodecontrol オペランドが unit の場合、hdestart_n コマンドで抽出マスタプロセスを起動できません。この場合、hdestart コマンドで抽出マスタプロセスを起動してください。

KFRB00723-E

Unable to access shared memory, because the exclusion could not get, type = aa...aa.
(C + L)

排他を取得できなかったため、共有メモリにアクセスできません。共有メモリ種別 = aa...aa.

aa...aa : 共有メモリ種別

COMMON (反映共通共用メモリ)

DEFSERV (反映定義解析共用メモリ)

EXTCOM (抽出共通共用メモリ)

STATINF (ステータス情報共用メモリ)

EXTDEF (抽出定義解析共用メモリ)

ERRTXT (メッセージテキスト共用メモリ)

LOBSHM (BLOB 反映用共用メモリ)

NMTCMD (抽出ノードマスタ-コマンド間連絡用共用メモリ)

(S)処理を中断します。

(O)抽出側システムで実行中のコマンドがある場合、そのコマンド終了後に、抽出ノードマスタプロセスを再起動してください。

KFRB00724-W

Status update is incompleted, host name = aa...aa. (C)

状態の更新に失敗しました。ホスト名 = aa...aa

メッセージ中に出力されているホスト名に該当するノードの状態更新に失敗しました。

aa...aa : ホスト名

(S)処理を続行します。

(O)このメッセージの直前に出力されているエラーメッセージに従って対処してください。エラーメッセージが出力されていない場合は、該当するノードの抽出ノードマスタエラー情報ファイルに出力されている KFRB00725-W メッセージに従って対処してください。

KFRB00725-W

Status update is incompletd, server name = aa...aa. (C)

状態の更新に失敗しました。サーバ名 = aa...aa

メッセージ中に出力されているサーバに関する状態の更新に失敗しました。

aa...aa：サーバ名

(S)処理を続行します。

(O)このメッセージの直前に出力されているエラーメッセージに従って対処してください。

KFRB00726-E

Invalid operand pd_rpl_reflect_mode of source HiRDB. (C)

抽出側 H i R D B のオペランド pd_rpl_reflect_mode が不正です。

反映トランザクション同期機能を使用する場合、抽出側 HiRDB のシステム定義には pd_rpl_reflect_mode=uap を指定します。

(S)処理を中断します。

(O)HiRDB のシステム定義に pd_rpl_reflect_mode=uap を指定し、HiRDB を再起動してください。

KFRB00801-E

Invalid HiRDB Datareplicator-id, id = aa, definition file = bb...bb. (C)

抽出側 Datareplicator 識別子の指定が不正です。抽出側 Datareplicator 識別子 = aa, 抽出システム定義ファイル名 = bb...bb.

aa：抽出側 Datareplicator 識別子

bb...bb：抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00802-E

Invalid extract environment statement file name in definition file, file name = aa...aa, definition file = bb...bb. (C)

抽出環境定義ファイル名が不正です。抽出環境定義ファイル名 = aa...aa, 抽出システム定義ファイル名 = bb...bb.

aa...aa : 抽出環境定義ファイル名

bb...bb : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00803-E

Invalid replication sender-id in definition file, sender-id = aa...aa, definition file = bb...bb. (C)

送信先識別子が不正です。送信先識別子 = aa...aa, 抽出システム定義ファイル名 = bb...bb.

aa...aa : 送信先識別子

bb...bb : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00804-E

Duplicate replication sender-id in definition file, sender-id = aa...aa, definition file = bb...bb. (C)

送信先識別子が重複しています。送信先識別子 = aa...aa, 抽出システム定義ファイル名 = bb...bb.

aa...aa : 送信先識別子

bb...bb : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00805-E

Missing transmission environment statement file in definition file, definition file = aa...aa. (C)

送信環境定義ファイルがありません。抽出システム定義ファイル名 = aa...aa.

aa...aa : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00806-E

Invalid transmission environment statement file name in definition file, file name = aa...aa, definition file = bb...bb. (C)

送信環境定義ファイル名が不正です。送信環境定義ファイル名 = aa...aa, 抽出システム定義ファイル名 = bb...bb.

aa...aa : 送信環境定義ファイル名

bb...bb : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00807-E

Missing replication node-id in definition file, definition file = aa...aa. (C)

データ連動識別子が指定されていません。抽出環境定義ファイル名 = aa...aa.

aa...aa : 抽出環境定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00808-E

Service name is not found in services file, service name = aa...aa, definition file = bb...bb. (C)

サービス名が登録されていません。サービス名 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : サービス名

bb...bb : 抽出システム定義ファイル名又は送信環境定義ファイル名

(S)処理を中断します。

(O)サービス名を登録するか、登録されているサービス名を指定してください。

KFRB00809-E

Missing extract environment statement file in definition file, definition file = aa...aa. (C)

抽出環境定義ファイルがありません。抽出システム定義ファイル名 = aa...aa.

aa...aa : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00813-E

Duplicate transmission queue file name in definition file, file name = aa...aa, definition file = bb...bb. (C)

抽出情報キューファイル名が重複しています。抽出情報キューファイル名 = aa...aa, 抽出環境定義ファイル名 = bb...bb.

抽出情報キューファイル名が重複しています。

aa...aa : 抽出情報キューファイル名

bb...bb : 抽出環境定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00814-E

Invalid file name in definition file, file name = aa...aa, definition file = bb...bb. (C)
ファイル名が不正です。不正なファイル名 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : 不正なファイル名

bb...bb : 定義ファイル名

(S)処理を中断します。

(O)定義ファイル内の変数の値を確認して、修正してください。

KFRB00815-E

Transmission queue file must be more than two in definition file, definition file = aa...aa. (C)

抽出情報キューファイルが2個以上指定されていません。抽出環境定義ファイル名 = aa...aa.

aa...aa : 抽出環境定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイル内で抽出情報キューファイルを2個以上指定してください。

KFRB00818-E

Invalid service name in definition file, service name = aa...aa, definition file = bb...bb. (C)

サービス名が不正です。サービス名 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : サービス名

bb...bb : 抽出システム定義ファイル名 又は送信環境定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル 又は送信環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00819-E

Invalid host name in definition file, host name = aa...aa, definition file = bb...bb. (C)

ホスト名が不正です。ホスト名 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : ホスト名

bb...bb : 抽出システム定義ファイル名 又は送信環境定義ファイル名

(S)処理を中断します。

(O)定義ファイル内のホスト名を確認して、修正してください。

KFRB00820-E

Invalid nsnd-id, nsnd-id = aa, definition file = bb...bb. (C)

送信抑止受信元識別子の指定が不正です。送信抑止受信元識別子 = aa, 送信環境定義ファイル名 = bb...bb.

aa : 送信抑止受信元識別子

bb...bb : 送信環境定義ファイル名

(S)処理を中断します。

(O)送信環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00821-E

Duplicate nsnd-id in definition file, definition file = aa...aa. (C)

送信抑止受信元識別子が重複しています。送信環境定義ファイル名 = aa...aa.

aa...aa：送信環境定義ファイル名

(S)処理を中断します。

(O)送信環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00822-E

Missing HiRDB Datareplicator-id, definition file = aa...aa. (C)

抽出側 Datareplicator 識別子が指定されていません。抽出システム定義ファイル名 = aa...aa.

aa...aa：抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00823-E

Queue file I/O buffer size in definition file exceeds transmission queue file size, definition file = aa...aa. (C)

抽出情報キューファイル I/O バッファサイズが抽出情報キューファイルサイズより大きいです。抽出環境定義ファイル名 = aa...aa.

aa...aa：抽出環境定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00824-E

Variable was changed in definition file after last information, definition file = aa...aa, variable = bb...bb. (C)

再起動時に定義ファイル内の内容が変更されています。定義ファイル名 = aa...aa, 変数名 = bb...bb.

aa...aa：定義ファイル名

定義ファイル名に「nodenum」が出力される場合は、HiRDB システム共通定義の pdunit の指定値が変更されています。

bb...bb：変数名

(S)処理を中断します。

(O)hdestart -i コマンドで初期化するか、又は定義ファイルの内容を戻してから、再起動してください。2 か所以上の送信先識別子又はデータ連動識別子の部分初期化をする場合、1 か所ずつ定義変更をして部分初期化してください。

KFRB00831-E

Server Common definition get error, definition file = aa...aa, errno = xx...xx. (C)

共通定義部の切り出しに失敗しました。定義ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 定義ファイル名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00832-E

Server local definition get error, definition file = aa...aa, errno = xx...xx. (C)

個別定義部の切り出しに失敗しました。定義ファイル名 = aa...aa, エラー番号 = xx...xx.

aa...aa : 定義ファイル名

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB00833-E

Missing sender-id in definition file, definition file = aa...aa. (C)

送信先識別子が指定されていません。抽出システム定義ファイル名 = aa...aa.

aa...aa : 抽出システム定義ファイル名

(S)処理を中断します。

(O)抽出システム定義ファイル内の変数の値を確認して、修正してください。

KFRB00834-E

Missing sender system-id in definition file, definition file = aa...aa. (C)

相手システム識別子が指定されていません。送信環境定義ファイル名 = aa...aa.

aa...aa : 送信環境定義ファイル名

(S)処理を中断します。

(O)送信環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00835-E

Invalid header in definition file, definition file = aa...aa. (C)

ヘッダの書き方が不正です。定義ファイル名 = aa...aa.

aa...aa : 定義ファイル名

(S)処理を中断します。

(O)定義ファイル内のヘッダの記述を確認して、修正してください。

KFRB00837-E

Product of maxtran and maxtrandata exceeds maximum value, definition file = aa...aa.

(C)

同時実行最大トランザクション数とトランザクション内最大更新情報数の積が最大値を超えています。
送信環境定義ファイル名 = aa...aa.

aa...aa : 送信環境定義ファイル名

(S)処理を中断します。

(O)送信環境定義ファイル内の変数の値を確認して、修正してください。

KFRB00838-E

Invalid server name in definition file, server name = aa...aa, definition file = bb...bb. (C)

個別定義のヘッダに不正なサーバ名を設定しています。サーバ名 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : サーバ名

bb...bb : 抽出環境定義ファイル名又は送信環境定義ファイル名

(S)処理を中断します。

(O)個別定義部のサーバ名を確認して、修正してください。

KFRB00841-E

Invalid Tselector in definition file, Tselector = aa...aa, definition file = bb...bb. (S)

Tセレクトタが不正です。Tセレクトタ = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : Tセレクトタ

bb...bb : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB00842-E

Invalid NSAP address in definition file, NSAP address = aa...aa, definition file = bb...bb.
(S)

NSAP アドレスが不正です。NSAP アドレス = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : NSAP アドレス

bb...bb : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB00843-E

Cannot specify OSI protocol, definition file = aa...aa. (S)

通信プロトコルに OSI を指定できません。定義ファイル名 = aa...aa.

aa...aa : 定義ファイル名

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB00844-E

Invalid operand value, definition file = aa...aa, operand = bb...bb, value = cc...cc. (S)

オペランド値が不正です。定義ファイル名 = aa...aa, オペランド名 = bb...bb, 値 = cc...cc.

aa...aa : 定義ファイル名

bb...bb : オペランド名

cc...cc : オペランド値

(S)処理を中断します。

(O)エラー原因を取り除いた後、再実行してください。

KFRB00845-E

Not specified Necessary operand, definition file = aa...aa, operand = bb...bb. (S)

必須オペランドが指定されていません。定義ファイル名 = aa...aa, オペランド名 = bb...bb.

aa...aa : 定義ファイル名

bb...bb : オペランド名

(S)処理を中断します。

(O)エラー原因を取り除いた後, 再実行してください。

KFRB00846-E

Duplicate port number, port number = aa...aa, definition file = bb...bb. (S)

ポート番号が重複しています。ポート番号 = aa...aa, 定義ファイル名 = bb...bb

aa...aa : ポート番号

bb...bb : 定義ファイル

(S)処理を中断します。

(O)抽出システム定義の mstservice オペランド, 及び connection_accept_service オペランドに指定したサービス名に該当するポート番号が重複しています。異なるポート番号に定義を修正した後, 再実行してください。

KFRB00847-E

Combination of operand aa...a and bb...b is illegal. (C + S)

aa...a と bb...b オペランドの組合せが不正です。

反映トランザクション同期機能では使用できないオペランドの組み合わせを検知しました。

aa...a : オペランド名

bb...b : オペランド名

(S)処理を中断します。

(O)オペランドの設定を修正した後, 再度初期化してください。

KFRB00848-E

Event code not unique in definition file, definition file = aa...aa, operand = bb...bb, cc...cc.

(L + E)

定義ファイル中のイベントコードが一意ではありません。定義ファイル名 = aa...aa, オペランド名 = bb...bb, cc...cc.

イベントコードが重複しています。

aa...aa：定義ファイル名

bb...bb：オペランド名

cc...cc：オペランド名

(S)処理を中断します。

(O)定義ファイル aa...aa の bb...bb オペランドと cc...cc オペランドに指定したイベントコードが重複しています。異なるイベントコードとなるように定義を修正した後、再実行してください。

KFRB00849-E

Number of the servers which have extraction tables exceeds 128. (C)

抽出対象表の存在するサーバ数が 128 を超えています。

反映トランザクション同期機能を使用する場合に、抽出対象表の存在するサーバ数が上限を超えています。

(S)処理を中断します。

(O)反映トランザクション同期機能を使用する場合、すべての抽出対象表が存在するバックエンドサーバの合計は、一つの反映側 Datareplicator で受信できる最大数となります（反映側システムが UNIX の場合は 128、Windows の場合は 63）。バックエンドサーバ数が上限を超えないように抽出定義を修正した後、hdeprep コマンドを実行し、hdestart コマンドを再実行してください。

KFRB00851-E

Missing command-option in definition file, definition file = aa...aa, command = bb...bb, option = cc...cc. (C)

HiRDB のシステム共通定義で必要なコマンドオプションが指定されていません。定義ファイル = aa...aa, コマンド名 = bb...bb, オプション名 = cc...cc.

aa...aa：定義ファイル

bb...bb：コマンド名

cc...cc：オプション名

(S)処理を中断します。

(O)HiRDB の定義ファイルの内容を確認して、修正してください。

KFRB00861-E

Duplicate file kind appointed in definition file, server name = aa...aa, operand = bb...bb, definition file = cc...cc. (C)

ファイルシステム領域への割り当てファイル種別が重複して指定されました。サーバ名 = aa...aa, オペランド名 = bb...bb, 定義ファイル名 = cc...cc.

aa...aa : サーバ名

bb...bb : オペランド名

cc...cc : 定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイルの内容を確認して、修正してください。

KFRB00862-E

Invalid file kind appointed in definition file, file kind = aa...aa, definition file = bb...bb.
(C)

ファイルシステム領域への割り当てファイル種別指定が不正です。割り当てファイル種別 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : 割り当てファイル種別

bb...bb : 定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイルの内容を確認して、修正してください。

KFRB00863-E

Duplicate file system appointed in definition file, file system = aa...aa, definition file = bb...bb. (C)

ファイルシステム領域名称が重複して指定されました。ファイルシステム領域名 = aa...aa, 定義ファイル名 = bb...bb.

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : 定義ファイル名

(S)処理を中断します。

(O)抽出環境定義ファイルの内容を確認して、修正してください。

KFRB00864-W

System-call(aa...aa) error, param = bb...bb, detail = cc...cc. (L)

システムコールエラーが発生しました。

aa...aa : システムコール名

bb...bb : パラメタ情報※

cc...cc : システムコールエラー文字列

(S)処理を中止します。

(O)エラーの原因を取り除いてから、コマンドを再実行してください。

注※

パラメタ情報とシステムコールエラーの関連については、「[10.3 システムコールエラー一覧](#)」を参照してください。

KFRB00865-E

System-call(aa...aa) error, param = bb...bb, errno = cc...cc. (L)

システムコール(aa...aa)でエラーが発生しました。パラメタ = bb...bb, エラー番号 = cc...cc

aa...aa : システムコール名

bb...bb : パラメタ情報※

cc...cc : エラー番号

(S)処理を中止します。

(O)エラー番号に基づいてエラーの原因を取り除き、再度実行してください。エラー番号については、error.h 又は各 OS のマニュアルを参照してください。

注※

パラメタ情報とシステムコールエラーの関連については、「[10.3 システムコールエラー一覧](#)」を参照してください。

KFRB00866-E

Changed table definition after executed hdeprep, table name = aa...aa, table-id = bb...bb.

(C)

hdeprep コマンド実行後に、抽出対象表の表定義が変更されています。表名 = aa...aa, 表 ID = bb...bb

aa...aa : 表名

bb...bb : 表 ID

(S)処理を中断します。

(O)表示された表に対して表構造の変更を行っていない場合、hdeprep コマンドを実行した後、hdestart を再実行してください。表構造の変更を行った場合、抽出-反映間の DB の整合性を確保した後、抽出側及び反映側 Datareplicator を初期開始してください。

KFRB00868-W

```
Not check effectivity of hde_prpfile, reason = aa...a,bb...bb. (C)
hde_prpfile の有効性チェックを行いません。理由コード = aa...aa, bb...bb
```

aa...aa : 理由コード 1

HiRDB CONNECT の場合、HiRDB に接続できないため、表定義変更をチェックしないで起動します。

bb...bb : 理由コード 2 (CONNECT 発行時の SQLCODE を表示)

(S)処理を続行します。

KFRB00869-E

```
Unable to allocate several server's files into same file system, server name = aa...aa, operand
= bb...bb, definition file = cc...cc. (C)
同じファイルシステム領域に複数サーバのファイルを割り当てることはできません。サーバ名 =
aa...aa, オペランド名 = bb...bb, 定義ファイル名 = cc...cc
```

aa...aa : サーバ名

bb...bb : オペランド名

cc...cc : 定義ファイル

(S)処理を中止します。

(O)抽出システム定義の nodecontrol オペランドが server の場合、同じファイルシステム領域に複数のサーバのファイルを格納できません。

ファイルシステム領域内に同じサーバのファイルだけを格納するように定義ファイルを修正した後、再実行してください。

KFRB00870-E

```
Error occurred by the command execution, command = aa...a, code = bb...b. (C)
コマンド実行エラーが発生しました。コマンド名 = aa...a, リターンコード = bb...b.
```

Datareplicator で実行するコマンドでエラーが発生しました。

aa...a : コマンド名

bb...b : コマンドのリターンコード

(S)処理を中断します。

(O)エラーとなったコマンドが実行可能な環境かどうかを確認してください。エラーの原因を取り除いた後、再実行してください。

KFRB00915-E

```
Unable to open analysis file, analysis file = aa...aa, errno = xx...xx. (S + C)
```

定義情報を解析するために使う解析ファイルがオープンできません。

aa...aa : 解析ファイル名

抽出システム定義ファイル、抽出環境定義ファイル、又は送信環境定義ファイルの場合は、解析時に一時的に作成するワークファイルのファイル名（定義ファイル名_tmp）が出力されます。

xx...xx : エラー番号

エラー番号が0の場合には、パス名に誤りがあります。

(S)処理を中止します。

[対策]システム管理者に連絡して、正しくインストールされているかどうかを確認してください。

KFRB00916-E

```
Incorrect variable, file = aa...aa, line = bb...bb, variable = cc...cc. (S + C)
```

定義ファイルに記述されている変数の指定値に誤りがあります。

aa...aa : 定義ファイル名

抽出システム定義ファイル、抽出環境定義ファイル、又は送信環境定義ファイルの場合は、解析時に一時的に作成するワークファイルのファイル名（定義ファイル名_tmp）が出力されます。

bb...bb : エラーが発生した行

cc...cc : 変数名

(S)処理を中止します。

[対策]定義ファイルに記述されている変数の指定値を確認して、修正してください。

KFRB00917-E

```
Command name invalid, file = aa...aa, line = bb...bb, command = cc...cc. (S + C)
```

定義ファイルに記述されているコマンドが解析できません。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

cc...cc : コマンド名

(S)処理を中止します。

[対策]定義ファイルに記述されているコマンドを確認して, 修正してください。

KFRB00918-E

```
Operand name in definition file invalid, file = aa...aa, line = bb...bb, operand = cc...cc.  
(S + C)
```

定義ファイルに記述されているオペランド名に誤りがあります。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

cc...cc : オペランド名

(S)処理を中止します。

[対策]定義ファイルに記述されているオペランド名を確認して, 修正してください。

KFRB00919-E

```
Command argument invalid, file = aa...aa, line = bb...bb, command = cc...cc. (S + C)
```

コマンドの引数に誤りがあります。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

cc...cc : コマンド名

(S)処理を中止します。

[対策]定義ファイルに記述されているコマンドの引数を確認して, 修正してください。

KFRB00920-E

```
Operand argument invalid, file = aa...aa, line = bb...bb, operand = cc...cc. (S + C)
```

オペランドの引数に誤りがあります。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

cc...cc : オペランド名

(S)処理を中止します。

[対策]定義ファイルに記述されているオペランドの引数を確認して, 修正してください。

KFRB00921-E

```
Unable to analyze definition file due to insufficient memory, memory requirement = aa...aa. (S + C)
```

定義ファイルの解析処理でメモリが不足しました。

aa...aa : エラー発生時の要求メモリ量

(S)処理を中止します。

[対策]エラーの原因を取り除いた後, 再実行してください。

KFRB00922-E

```
Number of nests in definition file exceeds the limit, file = aa...aa, line = bb...bb. (S + C)
```

定義ファイルのネスト数が最大ネスト数を超えました。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

(S)処理を中止します。

[対策]定義ファイルのネスト数を確認して, 修正してください。

KFRB00923-E

```
Operand aa...aa specified twice, file = bb...bb, line = cc...cc. (S + C)
```

オペランドの指定が重複しています。

aa...aa : オペランド名

bb...bb : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

cc...cc : エラーが発生した行

(S)処理を中止します。

[対策]定義ファイルに記述されているオペランドを確認して, 修正してください。

KFRB00940-E

```
Unable to set environment variable, file = aa...aa, line = bb...bb. (S + C)
```

環境変数に設定されている変数の指定値に誤りがあります。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

(S)処理を中止します。

[対策]定義ファイルに記述されている環境変数を確認して, 修正してください。

KFRB00941-E

```
I/O error occurred, file = aa...aa. (S + C)
```

定義を解析するためのファイル読み出しで入出力エラーが発生しました。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

(S)処理を中止します。

[対策]定義ファイルの内容を確認して, 修正してください。

KFRB00942-E

Unable to open definition file, definition file = aa...aa, errno = xx...xx. (S + C)

定義ファイルをオープンできません。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

xx...xx : errno に設定されたエラー番号

エラー番号が 0 の場合には, パス名に誤りがあります。

(S)処理を中止します。

[対策]定義ファイルの内容を確認して, 修正してください。

KFRB00943-E

Record length exceeds the limit, file = aa...aa, line = bb...bb. (S + C)

定義ファイルの記述情報のレコード長が 80 バイトを超えています。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

(S)処理を中止します。

[対策]定義ファイルの内容を確認して, 修正してください。

KFRB00944-E

Variable name specified wrong, file = aa...aa, line = bb...bb. (S + C)

指定した変数に, 次に示す誤りがあります。

- 変数名の記述がありません。
- 変数名が誤っています。

aa...aa : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

bb...bb : エラーが発生した行

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00945-E

```
Unable to open definition file specified with include, file = aa...aa, line = bb...bb, definition  
file = cc...cc, errno = xx...xx. (S + C)
```

include で指定したファイルをオープンできません。

aa...aa : ファイル名

bb...bb : エラーが発生した行

cc...cc : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00946-E

```
Definition aa...aa described wrong, file = bb...bb, line = cc...cc. (S + C)
```

オペランドの引数に誤りがあります。

aa...aa : オペランド名

bb...bb : 定義ファイル名

抽出システム定義ファイル, 抽出環境定義ファイル, 又は送信環境定義ファイルの場合は, 解析時に一時的に作成するワークファイルのファイル名 (定義ファイル名_tmp) が出力されます。

cc...cc : エラーが発生した行

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00947-E

```
Command aa...aa invalid. (S + C)
```

コマンド名に誤りがあります。

aa...aa : コマンド名

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00948-E

Operand name in aa...aa command invalid. (S + C)

コマンド名に誤りがあります。

aa...aa : コマンド名

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00949-E

Command argument invalid, command = aa...aa. (S + C)

コマンドの引数に誤りがあります。

aa...aa : コマンド名

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00950-E

Operand argument invalid, operand = aa...aa. (S + C)

オペランドの引数に誤りがあります。

aa...aa : オペランド名

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00951-E

Operand specified twice, command = aa...aa. (S + C)

指定したオペランドが重複しています。

aa...aa : コマンド名

(S)処理を中止します。

[対策]定義ファイルの内容を確認して、修正してください。

KFRB00955-E

Specified file size exceeds a size of Character special file, file name = aa...aa, file size = bb...bb, specified size = cc...cc. (S)

オペランドに指定したファイルサイズがキャラクタ型スペシャルファイルの実際のサイズを超えています。

aa...aa：ファイル名

bb...bb：キャラクタ型スペシャルファイルのサイズ

cc...cc：オペランドに指定したファイルサイズ

(S)処理を中止します。

[対策]エラーとなったファイルについて、該当するオペランドの指定値を修正するか、キャラクタ型スペシャルファイルのサイズを変更した後、再実行してください。

KFRB00956-E

Operand name invalid, file = aa...a, operand = bb...b. (L)

指定したオペランド名は無効です。

aa...aa：定義ファイル名

bb...bb：オペランド名

(S)処理を中断します。

(O)bb...bb に出力されているオペランドを削除した後、再実行してください。

KFRB00991-Q

Do you really want to terminate all processes (y or n)? (S)

反映側 Datareplicator のプロセスを停止するかどうかを応答してください。

- ・「y」を応答：Datareplicator のプロセス終了処理を開始します。
- ・「y」以外を応答：Datareplicator のプロセス終了処理をキャンセルします。

(S)応答を待ちます。

(O)反映側 Datareplicator のプロセスを終了する場合は「y」を入力してください。反映側 Datareplicator のプロセスの終了をキャンセルする場合は「n」を入力してください。

KFRB00992-I

```
hdsemergencystop command started. (S + L)
```

hdsemergencystop コマンドを開始しました。

(S)処理を続行します。

KFRB00993-E

```
Error occurred by the hdsemergencystop command, addinfo[aa...aa, bb...bb] (S + L)
```

hdsemergencystop コマンドでエラーが発生しました。

aa...aa : 関数名 (内部情報)

bb...bb : エラー番号 (内部情報)

(S)処理を終了します。

(O)hdsemergencystop コマンドを再実行してください。hdsemergencystop コマンドを数回実行してもエラーとなる場合は、OS がコマンドを受け付けられない状態となっていますので、OS を再起動してください。

KFRB00994-I

```
hdsemergencystop command ended. (S + L)
```

hdsemergencystop コマンドが終了しました。

(S)処理を終了します。

KFRB01000-E

```
<extract definition> expansion error or <reflect definition> parsing error, file name = aa...aa. (C)
```

抽出定義の展開又は反映定義の解析に失敗しました。ファイル名 = aa...aa

Information:line no = xx...xx

aa...aa : 反映定義ファイルの絶対パス名

xx...xx : エラーが発生した行番号

エラーが発生した format 文, load 文, または group 文の先頭の行番号, または, エラーが発生した字句の行番号を出力します。

(S)反映処理を中断します。

(O)このメッセージの直前に出力されているエラーメッセージに従って対処し、反映処理を再起動してください。

直前にエラーメッセージが出力されていない場合は、エラーが発生した行番号の定義内容を見直し、反映処理を再起動してください。

注

「Information」はエラー情報ファイルだけ出力します。

KFRB01001-E

Information for update queue file is not <extract definition>. (C)

反映情報キューファイルから読み込んだ情報は抽出定義ではありません。

(S)反映処理を中断します。

(O)内部エラーです。保守員に連絡してください。

KFRB01002-E

Insufficient shared memory for parse, size = aa...aa. (C)

定義解析結果格納用の共用メモリ不足が発生しました。必要領域サイズ = aa...aa

aa...aa : 必要領域サイズ

(S)反映処理を中断します。

(O)反映環境定義の defshmsize オペランドの設定値を大きくした後、反映処理を再起動してください。

KFRB01003-E

<extract definition> could not convert to target database language. (C)

抽出定義を反映側 DB の文字コードに変換できません。

抽出定義をサーバマシン上の文字コードに変換できません。

(S)反映処理を中断します。

(O)文字コード環境の設定が正しいかどうかを確認してください。

KFRB01004-E

<reflect definition> file not specified. (C)

反映定義ファイルが指定されていません。

反映定義ファイルが指定されていません。反映環境定義の defmerge オペランドの指定が false の場合、reffile オペランドの指定は省略できません。

(S)反映処理を中断します。

(O)反映定義ファイルを作成して、reffile オペランドに反映定義ファイルの所在を指定してください。

KFRB01005-E

```
<load statement> not specified in <reflect definition> file. (C)  
反映定義ファイル内に load 文の指定がありません。
```

反映定義ファイル内に load 文の指定がありません。defmerge オペランドの指定が false の場合、少なくとも一つは load 文の指定が必要です。

(S)反映処理を中断します。

(O)反映定義ファイル内に load 文の指定を追加してください。

KFRB01100-E

```
<extracted data name> in <format clause> in <format statement> is invalid, <extracted  
data name> = aa...aa, reason = bb...bb. (C)  
format 文の format 句に指定した更新情報名が不正です。更新情報名 = aa...aa, 理由コード = bb...bb
```

aa...aa：更新情報名

実際に指定されている名称が 8 バイトを超える場合、8 バイト目までを出力します。

bb...bb：理由コード

length over：更新情報名称長が 8 バイトを超えている。

(S)反映処理を中断します。

(O)正しい更新情報名に変更した後、反映処理を再起動してください。

KFRB01101-E

```
<extracted data name> in <format clause> in <format statement> is duplicate, <extracted  
data name> = aa...aa. (C)  
format 文の format 句に指定した更新情報名が重複しています。更新情報名 = aa...aa.
```

aa...aa：更新情報名

(S)反映処理を中断します。

(O)更新情報名の重複を取り除いた後、反映処理を再起動してください。

KFRB01102-E

<extracted data name> aa...aa in <format clause> in <format statement> is not specified in <extract definition>. (C)

format 文の format 句に指定した更新情報名 aa...aa は抽出定義内で定義されていません。

aa...aa：更新情報名

(S)反映処理を中断します。

(O)抽出定義内で定義されている更新情報名に変更した後、反映処理を再起動してください。

KFRB01103-E

<field name> in <name clause> in <format statement> is invalid, <extracted data name> = aa...aa, <field name> = bb...bb, reason = cc...cc. (C)

format 文の name 句に指定したフィールド名が不正です。更新情報名 = aa...aa, フィールド名 = bb...bb, 理由コード = cc...cc

aa...aa：更新情報名

bb...bb：フィールド名

実際に指定されている名称が 30 バイトを超える場合、30 バイト目までを出力します。

cc...cc：理由コード

length over：更新情報名称長が 8 バイトを超えている。

(S)反映処理を中断します。

(O)正しいフィールド名に変更した後、反映処理を再起動してください。

KFRB01104-E

<field name> in <name clause> in <format statement> is duplicate, <extracted data name> = aa...aa, <field name> = bb...bb. (C)

format 文の name 句に指定したフィールド名が重複しています。更新情報名 = aa...aa, フィールド名 = bb...bb.

aa...aa：更新情報名

bb...bb：フィールド名

(S)反映処理を中断します。

(O)フィールド名の重複を取り除いた後、反映処理を再起動してください。

KFRB01105-E

<field name> in <name clause> in <format statement> exceeds maximum value,
<extracted data name> = aa...aa. (C)

format 文の name 句に指定したフィールド名の数が最大数を超えました。更新情報名 = aa...aa

aa...aa : 更新情報名

(S) 反映処理を中断します。

(O) フィールド名の数を減らした後、反映処理を再起動してください。

KFRB01106-E

Number of <extracted data> fields is not equal to number of <name clause>, <extracted data name> = aa...aa. (C)

更新情報のフィールド数と name 句の数が一致しません。更新情報名 = aa...aa

aa...aa : 更新情報名

(S) 反映処理を中断します。

(O) 更新情報のフィールド数と name 句の数 (const 句を指定した name 句は除く) を一致させた後、反映処理を再起動してください。なお、name 句に指定するフィールド名称が不正な場合にも、このメッセージが出力される場合があります。更新情報のフィールド数と name 句の数が一致している場合は、name 句に指定したフィールド名称が正しいかどうか確認してください。

KFRB01107-E

Initial value in <const clause> in <format statement> is invalid, <extracted data name> = aa...aa, <field name> = bb...bb, reason = cc...cc. (C)

format 文の const 句に指定した初期値が不正です。更新情報名 = aa...aa, フィールド名 = bb...bb, 理由コード = cc...cc

aa...aa : 更新情報名

bb...bb : フィールド名

cc...cc : 理由コード

length over : 長さが上限を超えている。

invalid constant : 初期値として指定できる定数以外を指定している。

(S) 反映処理を中断します。

(O) 初期値を正しく修正した後、反映処理を再起動してください。

KFRB01108-E

Number of <format statement> exceeds maximum value. (C)

反映定義ファイル内に指定した format 文の数が最大数を超過しました。

(S)反映処理を中断します。

(O)反映定義ファイル内の format 文の指定数が指定最大数以下になるように削除してください。

KFRB01109-E

<column data UOC identifier> in <by clause> in <format statement> is invalid, <extracted data name> = aa...aa, <field name> = bb...bb. (C)

format 文の by 句に指定した列データ編集 UOC 識別子が不正です。更新情報名 = aa...aa, フィールド名 = bb...bb

aa...aa : 更新情報名

bb...bb : フィールド名

(S)反映処理を中断します。

(O)正しい列データ編集 UOC 識別子に変更した後、反映処理を再起動してください。

KFRB01110-E

<nocodecnv> is specified for field that is mapping key column. (C)

マッピングキーに対応するフィールドに対して、nocodecnv オプションが指定されています。

(S)反映処理を中断します。

(O)nocodecnv オプションの指定対象を確認・変更した後、反映処理を再起動してください。

KFRB01111-E

Column data UOC is specified for field that is aa...aa type extraction column, <extracted data name> = bb...bb, <field name> = cc...cc. (C)

aa...aa 型の抽出列データタイプに対応するフィールドに対して、列データ編集 UOC が指定されています。更新情報名 = bb...bb, フィールド名 = cc...cc

aa...aa : データ型名

BLOB : BLOB 型

BINARY : BINARY 型

bb...bb : 更新情報名

cc...cc : フィールド名

(S)反映処理を中断します。

(O)列データ編集 UOC の指定対象を確認・変更した後、反映処理を再起動してください。

KFRB01112-E

Column data UOC is specified for field that is mapping key column, <extracted data name> = aa...aa, <field name> = bb...bb. (C)

マッピングキーに対応するフィールドに対して、列データ編集 UOC が指定されています。更新情報名 = aa...aa, フィールド名 = bb...bb

aa...aa : 更新情報名

bb...bb : フィールド名

(S)反映処理を中断します。

(O)列データ編集 UOC の指定対象を確認・変更した後、反映処理を再起動してください。

KFRB01113-E

<nocodecnv> is specified for field that is not character column, <extracted data name> = aa...aa, <field name> = bb...bb. (C)

文字列属性でない列に対応するフィールドに対して、nocodecnv オプションが指定されています。更新情報名 = aa...aa, フィールド名 = bb...bb

aa...aa : 更新情報名

bb...bb : フィールド名

(S)反映処理を中断します。

(O)nocodecnv オプションの指定対象を確認・変更した後、反映処理を再起動してください。

KFRB01114-E

Column data UOC is specified for field that is Abstract Data Type extraction column. (C)

抽象データ型抽出列データタイプに対応するフィールドに対して、列データ編集 UOC が指定されています。

(S)反映処理を中断します。

(O)列データ編集 UOC の指定対象を確認・変更した後、反映処理を再起動してください。

KFRB01115-E

```
Unknown Abstract Data Type found, <type name> = aa...aa.    (C)
未サポートの抽象データ型に対する抽出定義を検知しました。抽象データ型名 = aa...aa.
```

aa...aa：抽象データ型名

(S)終了処理を開始します。

(O)Datareplicator でサポートしている抽象データ型かどうかを、「[1.4.2 抽象データ型を使った表のデータ連動](#)」を参照して確認してください。

Datareplicator でサポートしていない抽象データ型の場合：

ディクショナリ表を検索して、未サポートの抽象データ型の列を特定します。列を特定したら、抽出定義から該当する列を削除し、抽出側 Datareplicator 及び反映側 Datareplicator を初期化してください。

ディクショナリ表の検索方法についてはマニュアル「[HiRDB UAP 開発ガイド](#)」を参照してください。

Datareplicator でサポートしている抽象データ型の場合：

HiRDB 及び Datareplicator のバージョンを、該当する抽象データ型がサポートされているバージョンにバージョンアップしてください。バージョンアップの注意事項については、リリースノートを確認してください。

KFRB01116-E

```
<sequence identifier> in <const clause> in <format statement> is invalid, <extracted data name> = aa...aa, <field name> = bb...bb, reason = cc...cc.    (C)
```

format 文の const 句に指定された順序数生成子識別子が不正です。

aa...aa：更新情報名

bb...bb：フィールド名

cc...cc：理由コード

length over：長さが 30 バイトを超えている。

(S)反映処理を中断します。

(O)正しい順序数生成子識別子に変更した後、反映処理を再起動してください。

KFRB01117-E

```
<authorization identifier> in <const clause> in <format statement> is invalid, <extracted data name> = aa...aa, <field name> = bb...bb, reason = cc...cc.    (C)
```

format 文の const 句に指定された認可識別子が不正です。

aa...aa：更新情報名

bb...bb：フィールド名

cc...cc：理由コード

length over：長さが 30 バイトを超えている。

(S)反映処理を中断します。

(O)正しい認可識別子に変更した後、反映処理を再起動してください。

KFRB01200-E

```
<field name> in <load clause> in <load statement> is invalid, <extracted data name> =  
aa...aa, <field name> = bb...bb, reason = cc...cc. (C)
```

load 文の load 句に指定したフィールド名が不正です。更新情報名 = aa...aa, フィールド名 =
bb...bb, 理由コード = cc...cc

aa...aa：更新情報名

bb...bb：フィールド名

実際に指定されている名称が 30 バイトを超える場合、30 バイト目までを出力します。

cc...cc：理由コード

length over：長さが 30 バイトを超えている。

(S)反映処理を中断します。

(O)正しいフィールド名に変更した後、反映処理を再起動してください。

KFRB01201-E

```
<field name> aa...aa in <load clause> in <load statement> is not specified in <format  
statement>, <extracted data name> = bb...bb. (C)
```

load 文の load 句に指定したフィールド名 aa...aa は format 文で定義されていません。更新情報名 =
bb...bb

aa...aa：フィールド名

bb...bb：更新情報名

(S)反映処理を中断します。

(O)format 文で定義されているフィールド名に変更した後、反映処理を再起動してください。

KFRB01202-E

<field name> in <load clause> in <load statement> exceeds maximum value, <extracted data name> = aa...aa. (C)

load 文の load 句に指定したフィールド名の数が最大数を超えました。更新情報名 = aa...aa

aa...aa：更新情報名

(S)反映処理を中断します。

(O)フィールド名の数を減らした後、反映処理を再起動してください。

KFRB01203-E

Number of <field name> in <load clause> in <load statement> is not equal to <reflect table> columns, <extracted data name> = aa...aa. (C)

load 文の load 句に指定したフィールド名の数と反映表の列数が一致しません。更新情報名 = aa...aa

aa...aa：更新情報名

(S)反映処理を中断します。

(O)フィールド名の数と反映表の列数を一致させた後、反映処理を再起動してください。

KFRB01204-E

<extracted data name> in <from clause> in <load statement> is invalid, <extracted data name> = aa...aa, reason = bb...bb. (C)

load 文の from 句に指定した更新情報名が不正です。更新情報名 = aa...aa, 理由コード = bb...bb

aa...aa：更新情報名

実際に指定されている名称が 8 バイトを超える場合、8 バイト目までを出力します。

bb...bb：理由コード

length over：長さが 8 バイトを超えている。

(S)反映処理を中断します。

(O)正しい更新情報名に変更した後、反映処理を再起動してください。

KFRB01205-E

<extracted data name> aa...aa in <from clause> in <load statement> is not specified in <extract definition>. (C)

load 文の from 句に指定した更新情報名 aa...aa は抽出定義内で定義されていません。

aa...aa：更新情報名

(S)反映処理を中断します。

(O)抽出定義内で定義されている更新情報名に変更した後、反映処理を再起動してください。

KFRB01206-E

```
<reflect table> existence check failed, <table name> = aa...aa. (C)
```

反映対象の表の存在チェックに失敗しました。表名 = aa...aa.

詳細情報

aa...aa : 表名

詳細情報 : エラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)反映処理を中断します。

(O)詳細情報に出力している SQL コードを参照して、エラー要因を排除してください。

KFRB01207-E

```
<table name> in <to clause> in <load statement> is not real table, <table name> =  
aa...aa. (C)
```

load 文の to 句に指定した表名は実表ではありません。表名 = aa...aa.

aa...aa : 表名

(S)反映処理を中断します。

(O)実表に変更した後、反映処理を再起動してください。

KFRB01208-E

```
<authorization identifier> in <to clause> in <load statement> is invalid, <extracted data  
name> = aa...aa, reason = bb...bb. (C)
```

load 文の to 句に指定した認可識別子が不正です。更新情報名 = aa...aa, 理由コード = bb...bb

aa...aa : 更新情報名

bb...bb : 理由コード

length over : 長さが 30 バイトを超えている。

(S)反映処理を中断します。

(O)正しい認可識別子に変更した後、反映処理を再起動してください。

KFRB01209-E

<table identifier> in <to clause> in <load statement> is invalid, <extracted data name> = aa...aa, reason = bb...bb. (C)

load 文の to 句に指定した表識別子が不正です。更新情報名 = aa...aa, 理由コード = bb...bb

aa...aa : 更新情報名

bb...bb : 理由コード

length over : 長さが 28 バイトを超えている。

(S) 反映処理を中断します。

(O) 正しい表識別子に変更した後、反映処理を再起動してください。

KFRB01210-E

<uoc name> in <by clause> in <load statement> is invalid, <extracted data name> = aa...aa, reason = bb...bb. (C)

load 文の by 句に指定した uoc 名が不正です。更新情報名 = aa...aa, 理由コード = bb...bb

aa...aa : 更新情報名

bb...bb : 理由コード

length over : 長さが 64 バイトを超えている。

invalid name : アポストロフィ (') で囲まれた絶対パスの形式になっていない。

(S) 反映処理を中断します。

(O) 正しい UOC 名に変更した後、反映処理を再起動してください。

KFRB01211-E

Not specified all <mapping key> field in <load clause> in <load statement>, <extracted data name> = aa...aa. (C)

load 文の load 句にマッピングキーに該当するすべてのフィールドが指定されていません。更新情報名 = aa...aa

aa...aa : 更新情報名

(S) 反映処理を中断します。

(O) マッピングキーに該当するフィールドを追加した後、反映処理を再起動してください。

KFRB01212-E

Duplicate <mapping key> field in <load clause> in <load statement>, <extracted data name> = aa...aa, <field name> = bb...bb. (C)

load 文の load 句にマッピングキーに該当するフィールドが重複して指定されています。更新情報名 = aa...aa, フィールド名 = bb...bb

aa...aa : 更新情報名

bb...bb : フィールド名

(S)反映処理を中断します。

(O)マッピングキーに該当するフィールドの重複を排除した後、反映処理を再起動してください。

KFRB01213-E

Number of <load statement> exceeds maximum value. (C)

反映定義ファイル内に指定した load 文の数が最大数を超過しました。

(S)反映処理を中断します。

(O)反映定義ファイル内の load 文の指定数が暗黙的な反映対象の表 (load 文の明示的な指定がなく、抽出対象の表の名称が仮定された反映対象の表) の数を含めて指定最大数以下になるように削除してください。

KFRB01214-E

Unable to specified <constant field>, <extracted data name> = aa...aa, <field name> = bb...bb. (C)

定数フィールドは指定できません。

UOC を使うときは、load 句に定数フィールドは指定できません。更新情報名 = aa...aa, フィールド名 = bb...bb

aa...aa : 更新情報名

bb...bb : フィールド名

(S)反映処理を中断します。

(O)load 句に指定した定数フィールドを取り除いた後、反映処理を再起動してください。

KFRB01215-E

Unable to reflect multi-value column reflect to timestamp table, extract data name = aa...aa, field no = bb...bb. (C)

繰返し列を時系列情報表に反映することはできません。更新情報名 = aa...aa, フィールド番号 = bb...bb

aa...aa : 更新情報名

bb...bb : load 句に指定したフィールド名称の順序番号 (繰返し列のフィールドが複数指定されていても出力するのは一つだけです)

(S)反映処理を中断します。

(O)load 句に指定した繰返し列に対応するフィールド名称を削除するか, to 句に指定した timestamp を削除した後, 反映処理を再起動してください。

KFRB01216-E

```
Failed to get information of unique check of mapping key, SQLCODE = aa... aa. (C)
マッピングキー列のユニークチェックに使用する情報の取得に失敗しました。
```

aa...aa : SQLCODE

(S)処理を中止します。

(O)詳細情報を参照してエラーの原因を取り除いた後, 反映処理を再起動してください。

KFRB01217-E

```
Mapping key column does not satisfy condition of unique check, table name = aa...aa, code
= bb...bb. (C)
マッピングキー列がユニークチェックの条件を満たしていません。
```

aa...aa : 反映対象表名

bb...bb : 要因コード

UNIQUE : 反映対象表に, ユニークチェックの条件を満たすインデックスが定義されていません。

NOT NULL : ユニークチェックの条件を満たすインデックスの構成列に, NOT NULL 属性ではない列が含まれています。

(S)処理を中止します。

(O)マッピングキーに対応する列に間違いがないか見直しを行い, 間違いがある場合は定義を修正してください。指定に間違いがない場合は, 要因コードに従い処置してください。処置ができない場合は, 反映環境定義の mapping_key_check 又は反映定義のチェック句の指定値を変更した後, 反映処理を再実行してください。

- UNIQUE

反映対象表に, ユニークチェックの条件を満たすインデックスを定義してください。

- NOT NULL

ユニークチェックの条件を満たすインデックスの構成列を, NOT NULL 属性にしてください。

KFRB01218-E

Combination of definition value aa...aa in bb...bb and cc...cc in dd...dd is illegal. (C)

反映側 Datareplicator 稼働中の表単位排他取得機能で、使用できないオペランド又は定義句の組み合わせを検知しました。

aa...aa：オペランド名又は定義句

bb...bb：ファイル名

REFLECT ENVIRONMENT FILE：反映環境定義

REFLECT DEFINITION FILE：反映定義

cc...cc：オペランド名又は定義句

dd...dd：ファイル名

REFLECT DEFINITION FILE：反映定義

(S)処理を中止します。

(O)エラーとなったオペランド又は定義句の値を修正した後、反映処理を再起動してください。

KFRB01300-E

<group name> in <group clause> in <group statement> is invalid, <group name> = aa...aa, reason = bb...bb. (C)

group 文の group 句に指定したグループ名が不正です。グループ名 = aa...aa, 理由コード = bb...bb

aa...aa：グループ名

実際に指定されている名称が 8 バイトを超える場合、8 バイト目までを出力します。

bb...bb：理由コード

length over：長さが 8 バイトを超えている。

invalid name：以下の名称が使われている。

- trngroup
- othergrp
- uocxxx*

注※

xxx は 3 けたの符号なし整数を表します。

(S)反映処理を中断します。

(O)正しいグループ名に変更した後、反映処理を再起動してください。

KFRB01301-E

```
<group name> in <group clause> in <group statement> is duplicate, <group name> =  
aa...aa. (C)
```

group 文の group 句に指定したグループ名が重複しています。グループ名 = aa...aa.

aa...aa : グループ名

(S)反映処理を中断します。

(O)グループ名の重複を取り除いた後、反映処理を再起動してください。

KFRB01302-E

```
<authorization identifier> in <by clause> in <group statement> is invalid, <group name> =  
aa...aa, <authorization identifier> = bb...bb, reason = cc...cc. (C)
```

group 文の by 句に指定した認可識別子が不正です。グループ名 = aa...aa, 認可識別子 = bb...bb, 理由コード = cc...cc

aa...aa : グループ名

bb...bb : 認可識別子

実際に指定されている名称が 30 バイトを超える場合、30 バイト目までを出力します。

cc...cc : 理由コード

length over : 長さが 30 バイトを超えている。

(S)反映処理を中断します。

(O)正しい認可識別子に変更した後、反映処理を再起動してください。

KFRB01303-E

```
<table identifier> in <by clause> in <group statement> is invalid, <group name> =  
aa...aa, <table identifier> = bb...bb, reason = cc...cc. (C)
```

group 文の by 句に指定した表識別子が不正です。グループ名 = aa...aa, 表識別子 = bb...bb, 理由コード = cc...cc

aa...aa : グループ名

bb...bb : 表識別子

実際に指定されている名称が 30 バイトを超える場合、30 バイト目までを出力します。

cc...cc : 理由コード

length over : 長さが 30 バイトを超えている。

multiple table：キーレンジ分割又はハッシュ分割の指定があるにもかかわらず表識別子が複数指定されている。

(S)反映処理を中断します。

(O)正しい表識別子に変更した後、反映処理を再起動してください。

KFRB01304-E

```
<table name> in <by clause> in <group statement> is not <reflect table>, <table name> = aa...aa. (C)
```

group 文の by 句に指定した表名は反映対象の表ではありません。表名 = aa...aa.

aa...aa：表名

(S)反映処理を中断します。

(O)反映対象の表に変更した後、反映処理を再起動してください。

KFRB01305-E

```
<group name> in <group clause> in <group statement> exceeds maximum value. (C)
```

group 文の group 句に指定したグループ名の数が最大数を超過しました。

(S)反映処理を中断します。

(O)グループ名の数を減らした後、反映処理を再起動してください。

KFRB01306-E

```
<table name> in <by clause> in <group statement> is duplicate, <table name> = aa...aa. (C)
```

group 文の by 句に指定した表名が重複しています。表名 = aa...aa.

aa...aa：表名

(S)反映処理を中断します。

(O)表名の重複を取り除いた後、反映処理を再起動してください。

KFRB01307-E

```
Column is not mapping key in <having clause> or <hash-divide key>, <group name> = aa...aa, <column name> = bb...bb. (C)
```

group 文の having 句に指定した列名又はハッシュの分割キー列がマッピングキーではありません。グループ名 = aa...aa, 列名 = bb...bb.

aa...aa : グループ名

bb...bb : 列名

(S)反映処理を中断します。

(O)条件文の列にマッピングキーを指定した後、反映処理を再起動してください。ハッシュ分割のときは、マッピングキーとハッシュ分割キーを一致させてから、反映処理を再起動してください。

KFRB01308-E

Column name is not exist in <having clause>, <group name> = aa...aa, <column name> = bb...bb. (C)

group 文の having 句に存在しない列が指定されました。グループ名 = aa...aa, 列名 = bb...bb.

aa...aa : グループ名

bb...bb : 列名

(S)反映処理を中断します。

(O)条件文に表が持つ列の名称を指定した後、反映処理を再起動してください。

KFRB01309-E

Column type and constant type is not matched in <having clause>, <group name> = aa...aa, <column name> = bb...bb. (C)

group 文の having 句で指定した列と定数のデータ型の対応がとれません。グループ名 = aa...aa, 列名 = bb...bb.

aa...aa : グループ名

bb...bb : 列名

(S)反映処理を中断します。

(O)条件文を変更した後、反映処理を再起動してください。

KFRB01310-E

Number of key range overflow in <group clause>, <group name> = aa...aa. (C)

group 文に指定されたキーレンジが最大数を超えました。グループ名 = aa...aa.

aa...aa : 反映グループ名

(S)反映処理を中断します。

(O)グループ名のキーレンジ数を上限以下とした後、反映処理を再起動してください。

KFRB01311-E

Number of range condition overflow in <having clause>, <group name> = aa...aa. (C)
group 文の having 句に指定された条件文が最大数を超過しました。グループ名 = aa...aa.

aa...aa : 反映グループ名

(S)反映処理を中断します。

(O)キーレンジの条件文数を上限以下とした後、反映処理を再起動してください。

KFRB01312-E

Decimal data overflow in <having clause>, <group name> = aa...aa, <decimal data> = bb...bb. (C)
group 文の having 句に指定した 10 進データがオーバフローしました。グループ名 = aa...aa, 10 進データ値 = bb...bb.

aa...aa : グループ名

bb...bb : 10 進データ値

(S)反映処理を中断します。

(O)条件文に指定した 10 進データを変更した後、反映処理を再起動してください。

KFRB01313-E

Invalid data format in <having clause>, <group name> = aa...aa, <constant data> = bb...bb. (C)
group 文の having 句に指定した定数の値が不正です。グループ名 = aa...aa, 定数値 = bb...bb.

aa...aa : グループ名

bb...bb : 定数値

(S)反映処理を中断します。

(O)条件文に指定した定数を変更した後、反映処理を再起動してください。

KFRB01314-E

Invalid date format in <having clause>, <group name> = aa...aa, <date data> = bb...bb. (C)
group 文の having 句で指定した外部形式の日付データが不正です。グループ名 = aa...aa, 日付データ = bb...bb.

aa...aa : グループ名

bb...bb : 日付データ

(S)反映処理を中断します。

(O)条件文の日付データを変更した後、反映処理を再起動してください。

KFRB01315-E

Invalid time format in <having clause>, <group name> = aa...aa, <time data> = bb...bb. (C)

group 文の having 句で指定した外部形式の時間データが不正です。グループ名 = aa...aa, 時間データ = bb...bb.

aa...aa : グループ名

bb...bb : 時間データ

(S)反映処理を中断します。

(O)条件文の時間データを変更した後、反映処理を再起動してください。

KFRB01316-E

Invalid FES name in <in clause>, <group name> = aa...aa, <FES name> = bb...bb, error info = cc...cc, reason = dd...dd. (C)

group 文の in 句で反映先 FES の指定が不正です。反映グループ名 = aa...aa, 反映先 FES = bb...bb, エラー情報 = cc...cc, 理由コード = dd...dd

aa...aa : 反映グループ名

bb...bb : 反映先 FES のサーバ名

最大 128 バイト目までを出力します。

cc...cc : エラーとなった個所

host name : 反映先 FES のホスト名

server name : 反映先 FES のサーバ識別子

port number : 反映先 FES のポート番号

dd...dd : 理由コード

length over : 長さが最大長を超えている。

invalid name : アポストロフィ (') で囲まれた形式になっていない。

(S)反映処理を中断します。

(O)反映先 FES のサーバ名を変更した後、反映処理を再起動してください。

KFRB01317-E

Invalid FES number in <in clause>, <group name> = aa...aa, <FES number> = bb...bb. (C)

group 文の in 句で指定された反映先 FES 数が不正です。グループ名 = aa...aa, 反映先 FES 数 = bb...bb.

aa...aa : グループ名

bb...bb : 反映先 FES 数

(S)反映処理を中断します。

(O)反映先 FES の指定数を変更した後, 反映処理を再起動してください。

KFRB01318-E

Not exist hash function library, library name = aa...aa. (C)

ハッシュ関数ライブラリが存在しません。ライブラリ名 = aa...aa.

aa...aa : ライブラリ名

(S)反映処理を中断します。

(O)HiRDB のハッシュ関数ライブラリをインストールした後, 反映処理を再起動してください。

KFRB01319-E

Failed to load hash function library, library name = aa...aa, errno = bb...bb. (C)

ハッシュ関数ライブラリのロードに失敗しました。ライブラリ名 = aa...aa, エラー番号 = bb...bb.

aa...aa : ライブラリ名

bb...bb : エラー番号

(S)反映処理を中断します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して, エラーの原因を取り除いた後, 反映処理を再起動してください。

KFRB01320-E

Table is not hash-divided in <group clause>, <table name> = aa...aa. (C)

group 文で指定された表はハッシュ分割されていません。表名 = aa...aa.

aa...aa : 表名

(S)反映処理を中断します。

(O)反映先の表定義を変更した後、反映処理を再起動してください。

KFRB01321-E

Not support hash function in <group clause>, <table name> = aa...aa. (C)
group 文で指定された表のハッシュ分割に対応していません。表名 = aa...aa.

aa...aa : 表名

(S)反映処理を中断します。

(O)反映先の表定義を変更した後、反映処理を再起動してください。

KFRB01322-E

Failed to get hash-divided table information , <table name> = aa...aa. (C)
ハッシュ分割表の情報取得に失敗しました。表名 = aa...aa.

aa...aa : 表名

(S)反映処理を中断します。

(O)エラーファイルに出力した詳細情報を参照してエラーの原因を取り除いた後、反映処理を再起動してください。

KFRB01323-E

Hash key column type mapping error, <group name> = aa...aa, <table name> = bb...bb,
<column name> = cc...cc. (C)
ハッシュキー列のデータ型が抽出側と一致していません。反映グループ名 = aa...aa, 表名 =
bb...bb, 列名 = cc...cc.

aa...aa : 反映グループ名

bb...bb : 表名

cc...cc : 列名

(S)反映処理を中断します。

(O)反映先の表定義を変更した後、反映処理を再起動してください。

KFRB01324-E

Invalid divide number in <divide clause>, <group name> = aa...aa, <divide number> =
bb...bb. (C)

group 文の divide 句で指定された分割数が不正です。反映グループ名 = aa...aa, 分割数 = bb...bb.

aa...aa : 反映グループ名

bb...bb : 分割数

(S) 反映処理を中断します。

(O) ハッシュの分割数を変更した後、反映処理を再起動してください。

KFRB01325-E

Decimal data overflow in <divide clause>, <group name> = aa...aa, <decimal data> = bb...bb. (C)

group 文の divide 句に指定された 10 進データ値がオーバフローしました。反映グループ名 = aa...aa, 10 進データ値 = bb...bb.

aa...aa : 反映グループ名

bb...bb : 10 進データ値

(S) 反映処理を中断します。

(O) ハッシュの分割数を変更した後、反映処理を再起動してください。

KFRB01326-E

<rdarea name> in <rdarea_list clause> in <group statement> is invalid, <group name> = aa...aa, <rdarea name> = bb...bb, reason = cc...cc. (C)

group 文のハッシュ分割定義で指定した RD エリア名が不正です。反映グループ名 = aa...aa, RD エリア名 = bb...bb, 理由コード = cc...cc

aa...aa : 反映グループ名

bb...bb : RD エリア名

実際に指定されている名称が 30 バイトを超える場合、30 バイト目までを出力します。

cc...cc : 理由コード

length over : 長さが 30 バイトを超えている。

(S) 反映処理を中止します。

(O) 正しい RD エリア名に変更した後、反映処理を再起動してください。

KFRB01327-E

<rdarea name> in <rdarea_list clause> in <group statement> is duplicate, <group name> = aa...aa, <rdarea name> = bb...bb. (C)

group 文のハッシュ分割定義で指定した RD エリア名が重複しています。反映グループ名 = aa...aa, RD エリア名 = bb...bb.

aa...aa : 反映グループ名

bb...bb : RD エリア名

(S) 反映処理を中止します。

(O) RD エリア名の重複を取り除いた後、反映処理を再起動してください。

KFRB01328-E

FES, which includes RDareas and not, coexist error in <hash_fes_clause>, <table name> = aa...aa. (C)

group 文のハッシュ分割定義で RD エリア指定のある FES 指定と RD エリア指定のない FES 指定が混在しています。表名 = aa...aa.

aa...aa : 表名

(S) 反映処理を中止します。

(O) RD エリア名の指定有無の混在を修正した後、反映処理を再起動してください。

KFRB01329-E

Invalid timestamp format in <having clause>, <group name> = aa...aa, <timestamp data> = bb...bb. (C)

group 文の having 句で指定された外部形式の時刻印データが不正です。反映グループ名 = aa...aa, 時刻印データ = bb...bb.

aa...aa : 反映グループ名

bb...bb : 時刻印データ

(S) 処理を中止します。

(O) 条件文の時刻印データを変更した後、反映処理を再実行してください。

KFRB01332-E

Invalid "other" in <group statement>, <group name> = aa...aa. (C)

group 文の other の指定が不正です。グループ名 = aa...aa.

aa...aa：反映グループ名

(S)反映処理を中断します。

(O)other を指定する場合、以下の条件に従い定義を修正した後、再実行してください。

[キーレンジ単位分割方式の場合]

group 文内の最後の分割条件に other を 1 つだけ指定する。

[ハッシュ分割方式の場合]

group 文内の最後の反映先 FES 指定に RD エリア名として other を 1 つだけ指定する。

KFRB01333-E

```
sequence existence check failed, <sequence name> = aa...aa. (C)
```

順序数生成子の存在チェックに失敗しました。

aa...aa：順序数生成子名

(S)反映処理を中断します。

(O)詳細情報に出力している SQL コードを参照して、エラー要因を排除してください。

KFRB01401-E

```
Dual file definition has a mistake, file = aa...aa, line = bb...bb, code = cc...cc. (C + S + L)
```

二重化定義の記述に誤りがあります。ファイル = aa...aa, 行番号 = bb...bb, 詳細コード = cc...cc

aa...aa：二重化定義ファイル名

bb...bb：行番号

cc...cc：要因コード

(S)処理を中止します。

(O)二重化定義の誤りを修正した後、システムを再起動してください。要因コードについては「[10.4 要因コード一覧](#)」を参照してください。

KFRB01402-E

```
Unable to use dual file function, code = aa...aa, info = bb...bb. (C + S + L)
```

エラーが発生したため、ファイル二重化機能を使用することができません。詳細コード = aa...aa, 詳細情報 = bb...bb

aa...aa：要因コード

bb...bb : 詳細情報

(S)処理を中止します。

(O)要因コードに基づいてエラーの原因を取り除いた後、システムを再起動してください。要因コードについては「[10.4 要因コード一覧](#)」を参照してください。

KFRB01403-E

```
Operand combination error, definition file = aa...aa, operand = bb...bb, definition file = cc...cc, operand = dd...dd. (C)
```

オペランドの組み合わせが不正です。

aa...aa : 定義ファイル名

bb...bb : オペランド名

cc...cc : 定義ファイル名

dd...dd : オペランド名

(S)処理を中止します。

(O)オペランドの組み合わせを確認し修正した後、コマンドを再実行してください。

KFRB02001-E

```
Invalid data of reflect status file. (C)  
反映ステータスファイルの内容が不正です。
```

(S)処理を中断します。

(O)hdsstart -i コマンドで再実行してください。

KFRB02002-E

```
Socket connection error, errno = xx...xx. (C)  
ソケットの確立に失敗しました。エラー番号 = xx...xx.
```

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

Port check error, code = yy, zz...zz. (C)

ポートチェックでエラーを検知しました。詳細コード = yy, zz...zz.

yy：詳細コード

zz...zz：詳細コードごとの情報 (yy が 3, 4, 5, 7, 11, 12, 13, 14, 15 のときだけ, zz...zz が出力されます)

(S)処理を中断します。

(O)抽出側システムと反映側システムの接続に関する整合性チェックでエラーを検知しました。次に示す表に従って対策してください。

詳細コード	意味	対策
1	反映側 Datareplicator で出力している場合 Datareplicator, 又は XDM/DS 以外のプログラムが反映システム定義の hdsservice オペランドに指定したポート番号に接続しようとしています。	hdsservice オペランドに指定したポート番号を使用しているプログラムがないか確認し、ほかのプログラムと重複しないポート番号にしてください。
	抽出側 Datareplicator で出力している場合 次の原因が考えられます。 (1)反映システム定義の hdsservice オペランドに指定したポート番号と送信環境定義の hdeservice オペランドで指定したポート番号が一致していません。 (2)反映側システムの受信プロセスが生成されていません。	(1)反映システム定義の hdsservice オペランドで指定したポート番号と送信環境定義の hdeservice オペランドで指定したポート番号を一致させてください。 (2)反映側 Datareplicator が出力する KFRB00019-E に表示されるエラー番号を参照して対処してください。
2	抽出側システムと反映側システムの間接続に互換性がありません。次の原因が考えられます。 • 抽出側 DB が XDM/RD E2 の場合に、TIME 型又は INTERVAL HOUR TO SECOND 型で小数部を指定した列がある。	抽出側システムと反映側システムのバージョンをチェックして、正しい組み合わせで接続し直してください。
3	反映側 Datareplicator の反映システム定義に指定されていないデータ連動識別子が指定されています。	抽出側システムに指定したデータ連動識別子を反映システム定義に追加した後、接続し直してください。
4	反映側 Datareplicator の反映システム定義に指定されている反映側 Datareplicator 識別子と抽出側システムで指定されている反映側 Datareplicator 識別子が一致していません。	抽出側システムに指定した反映側 Datareplicator 識別子を修正した後、接続し直してください。
5	前回、送信した更新情報の続きではない更新情報を受信しました。次の原因が考えられます。 (1)抽出側 Datareplicator を初期化し、反映側 Datareplicator を初期化していない。 (2)同じ反映側 Datareplicator 識別子、及びデータ連動識別子を指定した別の抽出側 Datareplicator から接続しようとした。	(1)反映側 Datareplicator を初期化した後、送信処理を起動してください。 (2)抽出側 Datareplicator の接続先が正しいか確認してください。

詳細 コード	意 味	対 策
6	抽出側 DB が、反映側 Datareplicator の対象外の DB です。	抽出側システムの保守員に連絡してください。
7	反映側 Datareplicator での受信バッファが確保できません。 zz...zz に確保しようとした受信バッファのサイズを出力します。	受信バッファは、抽出側 Datareplicator の更新情報編集バッファのサイズで確保します。 抽出システム定義の sendcontrol オペランドを省略、又は nodemst を指定している場合 送信環境定義の editbufsize オペランドを小さくし、抽出側 Datareplicator を再起動してください。 オペランドに sendmst を指定している場合 抽出システム定義の smt_editbufsize オペランドを小さくし、抽出側 Datareplicator を再起動してください。
8	サポートされていない文字コードで送信しようとしています。	抽出側システムの保守員に連絡してください。
9	サポートされていない送信データ圧縮方式が指定されています。	抽出側システムの保守員に連絡してください。
10	サポートされていない更新情報形式で送信しようとしています。	抽出側システムの保守員に連絡してください。
11	抽出側システムの各バックエンドサーバに対応するデータ連動識別子が重複しています。zz...zz に重複したデータ連動識別子を出力します。	抽出環境定義の個別定義部で指定する dsid オペランドについて、同一の反映先に接続する抽出側システムのバックエンドサーバ間で一意なデータ連動識別子を指定した後、抽出側および反映側 Datareplicator を初期開始してください。
12	開始処理中に反映ステータスファイルの I/O エラーが発生しました。	ファイルの障害を取り除いた後、反映側 Datareplicator を再起動してください。
13	開始処理中にセマフォロック、又はアンロックエラーが発生しました。	セマフォの障害を取り除いた後、反映側 Datareplicator を再起動してください。
14	開始処理中に反映情報キューファイルの I/O エラーが発生しました。	ファイルの障害を取り除いた後、反映側 Datareplicator を再起動してください。
15	抽出側システムの更新情報バッファのサイズが前回のサイズと異なるため、処理を続行できません。	更新情報バッファサイズを元に戻した後、接続し直してください。
16	受信したデータがポートチェック情報ではありません。	反映環境定義の hdsservice オペランドに指定したサービスに対して、不正に通信をしているアプリケーションがないか確認してください。
17	DB 種別が前回の DB 種別と異なっているため処理を続行できません。	抽出側システムの保守員に連絡してください。
18	文字コードの組み合わせが不正です。	反映システム定義の dblocale オペランド、及び抽出システム定義の dblocale オペランドの設定が正しいかどうかを確認してください。

詳細コード	意味	対策
19	更新情報入力用のデータ連動識別子に対して、更新情報が送信されました。	反映環境定義ファイルの ujcodekind オペランドの設定を rcv にしてください。
20	Datareplicator Extension 固有の dbkind オペランドに指定した反映対象の DBMS が正しくありません。	Datareplicator Extension 固有の dbkind オペランドを正しい指定に変更し、再実行してください。
	メインフレームの DB と Oracle 又は SQLServer をデータ連動する場合、必要となる関連プログラム*がインストールされていません。	必要となる関連プログラム*をインストールして、再実行してください。SAM ファイルを使って連動している場合は、関連プログラムをインストールし SAM ファイルを再作成した後に、再実行してください。
21	HiRDB Datareplicator Extension をインストールしていません。	HiRDB Datareplicator Extension をインストールした後、再実行してください。
22	他社 DB 接続ライセンスの有効期限を過ぎています。	システム管理者に連絡してください。
23	ライセンスファイルが壊れています。	HiRDB Datareplicator Extension を再インストールしてください。
24	繰返し列に対応していない抽出側システムからの接続要求がありました。	抽出側システムを繰返し列に対応したバージョンの Datareplicator にしてください。 又は、抽出定義から繰返し列を除いてください。
25	抽出システム定義と反映システム定義の dblocale オペランドが一致しません。	抽出システム定義と反映システム定義の dblocale オペランドに同じ文字コード体系を指定した後、再実行してください。
26	反映システム定義に syncgroup001 オペランドが指定されていますが、抽出側システムはトランザクションプランチ情報を送信しない設定になっています。	[3.6.2 反映トランザクション同期機能の準備] を参照して、反映トランザクション同期機能を使用する場合の前提条件を確認してください。
27	前回送信されてきた同期イベントコードと、今回送信されてきた同期イベントコードが異なります。	次のどちらかの対処をしてください。 <ul style="list-style-type: none"> 送信環境定義の eventsync オペランドの指定値を、前回と同じ値に戻してください。 反映側 Datareplicator を初期化してください。

注※

メインフレームの DB と Oracle 又は SQLServer をデータ連動する場合に必要な関連プログラムを次に示します。

抽出側システム	Oracle	SQLServer
XDM/DS	XDM/DS/拡張連携 1	XDM/DS/拡張連携 2
VOS3 Database Datareplicator	XDM/DS/拡張連携 1	データ連動できません。
VOS1 PDMII E2 VOS1 RDB1 E2	VOS1 Database Adaptor 1	VOS1 Database Adaptor 2

KFRB02004-E

Connection was closed. (C)

通信回線が切断されました。

(S)処理を中断します。

(O)抽出側システムと反映側システム間の通信回線切断を検知しました。

以下の要因が考えられます。

- 抽出側 Datareplicator を強制停止した。
この場合、抽出側 Datareplicator を再起動してください。
- ネットワーク障害が発生した。
この場合、ネットワーク障害復旧後、抽出側 Datareplicator の送信プロセスの状態を確認し、停止している場合は再起動してください。

KFRB02005-E

Extracted data is invalid. (C)

更新情報が正しくありません。

(S)処理を中断します。

(O)内部エラーです。反映側 Datareplicagtor の運用ディレクトリ (HDSPATH) 下のダンプファイル 'hdstcpmst_<プロセス ID>.bdmp' を取得し、サポートサービスに送付してください。

KFRB02006-E

Received data sequence is invalid. (C)

受信したデータのシーケンスが正しくありません。

(S)処理を中断します。

(O)内部エラーです。反映側 Datareplicagtor の運用ディレクトリ (HDSPATH) 下のダンプファイル 'hdstcpmst_<プロセス ID>.bdmp' を取得し、サポートサービスに送付してください。

KFRB02007-E

Unable to save extract definition in reflect status file, because reflect status file is insufficient, necessary statssize is xx...xx. (C)

反映ステータスファイルのサイズが小さいため、抽出定義を格納できません。必要な statssize は xx...xx です。

xx...xx : statssize が必要とするサイズ (バイト)

(S)処理を中断します。

(O)反映環境定義の statssize オペランドの値を大きくした後、hdsstart -i コマンドで再起動してください。

KFRB02008-E

Connection count exceed maximum value. (C)

最大接続数を超過してデータ連動しようとしています。

(S)処理を中断します。

(O)接続しようとしているデータ連動識別子が反映システム定義に指定されているか確認してください。必要ならば修正した後、再実行してください。

KFRB02009-E

Data send error, errno = xx...xx. (C)

抽出側システムへのデータ送信でエラーが発生しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB02010-E

Data receive error, errno = xx...xx. (C)

抽出側システムからのデータ受信でエラーが発生しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB02011-E

Receive unknown data. (C)

受信対象外のデータを受信しました。

(S)処理を中断します。

(O)内部エラーです。反映側 Datareplicagtor の運用ディレクトリ (HDSPATH) 下のダンプファイル hdstcpmst_<プロセス ID>.bdmp' を取得し、サポートサービスに送付してください。

KFRB02012-E

Connection accept error, errno = xx...xx. (C)

コネクションの受付中にエラーが発生しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB02013-E

Socket option set error, errno = xx...xx. (C)

ソケットオプションの設定中にエラーが発生しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB02014-E

Select system call error, errno = xx...xx. (C)

select システムコールでエラーが発生しました。エラー番号 = xx...xx.

xx...xx : errno に設定されたエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB02015-E

Receive data exceed maximum receive data length. (C)

受信しようとしているデータが受信最大長より大きくなっています。

(S)処理を中断します。

(O)内部エラーです。保守員に連絡してください。

KFRB02016-E

Receive data extension failed. (C)

受信データの伸長に失敗しました。

(S)処理を中断します。

(O)内部エラーです。反映側 Datareplicagtor の運用ディレクトリ (HDSPATH) 下のダンプファイル 'hdstcpmst_<プロセス ID>.bdmp' を取得し、サポートサービスに送付してください。

KFRB02017-E

Receive data length is 0 byte. (C)

受信しようとしているデータ長が 0 バイトです。

(S)処理を中断します。

(O)内部エラーです。反映側 Datareplicagtor の運用ディレクトリ (HDSPATH) 下のダンプファイル 'hdstcpmst_<プロセス ID>.bdmp' を取得し、サポートサービスに送付してください。

KFRB02018-E

Specified service name not found in services file, service name = aa...aa. (C)

指定したサービス名が services ファイルの中にありません。サービス名 = aa...aa.

aa...aa : サービス名

(S)処理を中断します。

(O)services ファイルに指定したサービス名を追加するか、services ファイルの中にあるサービス名を指定してください。

KFRB02019-I

Communication master process started, protocol = aa...aa. (C)

反映通信マスタプロセスを起動しました。プロトコル種別 = aa...aa.

aa...aa : プロトコル種別

(S)処理を続行します。

KFRB02020-I

Communication master process ended, protocol = aa...aa. (C)

反映通信マスタプロセスを終了しました。プロトコル種別 = aa...aa.

aa...aa : プロトコル種別

(S)処理を続行します。

KFRB02021-I

Extracted data receipt started. (C)
更新情報の受信を開始しました。

(S)処理を続行します。

KFRB02022-I

Extracted data receipt ended. (C)
更新情報の受信を終了しました。

(S)処理を続行します。

KFRB02023-I

Accepted stop request in receiver, request kind = aa...aa. (C)
受信処理において停止要求を受け付けました。要求種別 = aa...aa.

aa...aa : 要求種別を示します。

Normal :

hdsstop コマンドを受け付けたため、送信終了通知の受信を完了した後、処理を終了します。

Event :

hdsstop -t event コマンドを受け付けたため、これ以降最初に出現するイベントを受信した後、処理を終了します。

Immediate :

hdsstop -t immediate コマンドを受け付けたため受信処理を終了します。

Force :

hdsstop -t force コマンドを受け付けたため受信処理を終了します。

(S)処理を続行します。

KFRB02024-E

OSI function call error. function name = aa...aa, errno = xx...xx. (C)
OSI の関数呼び出しでエラーが発生しました。関数名 = aa...aa, エラー番号 = xx...xx.

aa...aa : エラーが発生した OSI の関数

xx...xx : OSI の関数が返したエラー番号

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、tiuser.h 又は OS のマニュアルを参照してください。

KFRB02025-E

OSI communication program is not installed. (C)

OSI 通信に必要なプログラムがインストールされていません。

(S)処理を中断します。

(O)OSI 通信に必要なプログラムをインストールした後、反映側 Datareplicator を再起動してください。

KFRB02026-E

OSI communication program is not active. (C)

OSI 通信に必要なプログラムが稼働していません。

(S)処理を中断します。

(O)OSI 通信に必要なプログラムを稼働した後、反映側 Datareplicator を再起動してください。

KFRB02027-E

Sender process was held. (E)

送信プロセスが閉塞されました。

(S)処理を中断します。

(O)抽出側 DB と反映側 DB の整合性をとった後、送信プロセスを起動してください。

KFRB02028-E

Connect system call error, errno = xx...xx, host name = aa...aa, service name = bb...bb.
(E)

Connect システムコールでエラーが発生しました。エラー番号 = xx...xx, ホスト名 = aa...aa, サービス名 = bb...bb.

xx...xx : errno に設定されたエラー番号

aa...aa : ホスト名

bb...bb : サービス名

(S)処理を中断します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、tiuser.h 又は OS のマニュアルを参照してください。

KFRB02029-E

Missing specified host name in hosts file, host name = aa...aa. (E)
指定したホスト名が hosts ファイルの中にありません。ホスト名 = aa...aa.

aa...aa : ホスト名

(S)処理を中断します。

(O)hosts ファイルにホスト名を正しく設定して、再実行してください。

KFRB02030-E

Error occurred on target system, message number = nnnnn. (C)
反映側システムでエラーが発生しました。メッセージ番号 = nnnnn.

nnnnn : 反映側システムで発生したエラーのメッセージ番号

nnnnn は、Datareplicator が出力するメッセージ ID の KFRBnnnnn-T に対応します。

(S)処理を中断します。

(O)反映側システムで発生したエラーを取り除いた後、再実行してください。

KFRB02031-I

Sender process started. (E)
送信プロセスを起動しました。

(S)処理を続行します。

KFRB02032-I

Sender process ended, status = aa...aa. (E)
送信プロセスを終了しました。終了ステータス = aa...aa.

aa...aa : 終了ステータス

Auto-Normal : 停止ログによる正常停止

Manual-Normal : コマンドによる正常停止

Event : イベント検知停止

Auto-Force : 停止ログによる強制停止

Manual-Force : コマンドによる強制停止

Hold : 縮退による停止

Error : エラー検知による停止

(S)処理を続行します。

KFRB02033-I

```
Sender process sent log data.(transaction (xx...xx, yy...yy) : insert (xx...xx, yy...yy) : update  
(xx...xx, yy...yy) : delete (xx...xx, yy...yy) : purge (xx...xx, yy...yy) ) (E)  
更新情報を送信しました。(transaction (xx...xx, yy...yy) : insert (xx...xx, yy...yy) : update  
(xx...xx, yy...yy) : delete (xx...xx, yy...yy) )
```

操作した SQL 数が次のように出力されます。

transaction : 更新情報の中のトランザクション数

insert : insert 文の発行回数

update : update 文の発行回数

delete : delete 文の発行回数

purge : purge 文の発行回数

xx...xx : 更新情報中に含まれる件数

yy...yy : 送信した更新情報の累計

(S)処理を続行します。

KFRB02034-I

```
Sender process connected to target system, host name = aa...aa, service name = bb...bb.  
(E)
```

反映側システムと接続しました。ホスト名 = aa...aa, サービス名 = bb...bb.

aa...aa : ホスト名

bb...bb : サービス名

(S)処理を続行します。

KFRB02035-W

```
Target system is not active, host name = aa...aa, services name = bb...bb. (E)
```

反映側システムが起動していません。ホスト名 = aa...aa, サービス名 = bb...bb.

aa...aa : ホスト名

bb...bb : サービス名

(S)処理を続行します。

(O)反映側システムを起動してください。

KFRB02036-I

Sender process sent event data, event code = xx...xx. (E)
イベントデータを送信しました。イベントコード = xx...xx.

xx...xx : イベントコード

(S)処理を続行します。

KFRB02037-I

Sender process accepted stop request, request kind = aa...aa. (E)
送信処理において停止要求を受け付けました。要求種別 = aa...aa.

aa...aa : 要求種別を示します。

Normal : 正常停止要求

(S)処理を続行します。

KFRB02038-E

Value of editbufsize operand must be more than aa...aa. (E)
editbufsize オペランドの設定値は aa...aa より大きくなければなりません。

aa...aa : 必要なバッファ長

(S)処理を中断します。

(O)送信環境定義ファイルの editbufsize オペランドの値を大きくした後、再実行してください。

KFRB02039-E

No response from target system, host name = aa...aa, service name = bb...bb. (E)
反映側システムが応答しません。ホスト名 = aa...aa, サービス名 = bb...bb.

aa...aa : ホスト名

bb...bb : サービス名

(S)処理を中断します。

(O)反映側システムで異常が発生していないかを確認してください。

KFRB02040-I

Connection close request from source site system accepted. (C)
抽出側システムからの送信終了要求を受け付けました。

(S)処理を続行します。

KFRB02041-W

Halfway end of multi-byte character code found in data, substitute space code. (E)
コード変換対象データが複数バイト文字の途中で終了しているため、空白に置き換えます。

(S)処理を続行します。

KFRB02042-E

Log data is inconsistency with extract definition. (C)
抽出した更新情報と抽出定義の間に矛盾があります。

抽出情報キューファイル中の更新情報がすべて送信されていない状態で抽出定義が変更された可能性があります。

(S)処理を中断します。

(O)データ連動を再開する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

KFRB02043-E

Insufficient buffer occurred while editing log data. (E)
更新情報の編集集中に更新情報バッファが不足しました。

(S)処理を中断します。

(O)更新情報編集バッファサイズの見積もり式に従って適切なバッファサイズを設定した後、再実行してください。

KFRB02046-I

Sender process ended. (E)
送信プロセスを終了しました。

(S)処理を終了します。

KFRB02047-E

Target system is stopping now. (E)

反映側システムは停止処理中です。

(S)処理を終了します。

(O)反映側 Datareplicator を再起動してください。抽出側 Datareplicator は、送信環境定義で指定した回数だけリトライします。送信プロセス（送信処理）が終了した場合は、送信プロセスを再起動してください。

KFRB02048-I

Sender process connected to target system, NSAP address = aa...aa, Tselector = bb...bb.

(E)

反映側システムと接続しました。NSAP アドレス = aa...aa, T セレクタ = bb...bb.

aa...aa : NSAP アドレス値

bb...bb : T セレクタ値

(S)処理を続行します。

KFRB02049-W

Target system is not active, NSAP address = aa...aa, Tselector = bb...bb. (E)

反映側システムが起動していません。NSAP アドレス = aa...aa, T セレクタ = bb...bb.

aa...aa : NSAP アドレス値

bb...bb : T セレクタ値

(S)処理を続行します。

(O)反映側システムを起動してください。

KFRB02050-E

No response from target system, NSAP address = aa...aa, Tselector = bb...bb. (E)

反映側システムが応答しません。NSAP アドレス = aa...aa, T セレクタ = bb...bb.

aa...aa : NSAP アドレス値

bb...bb : T セレクタ値

(S)処理を中断します。

(O)反映先システムで異常がないかどうかを確認してください。

KFRB02051-E

Fail to convert address for osi. (E)

OSI 通信で使用するアドレスの変換でエラーが発生しました。

(S)処理を中断します。

(O)NSAP アドレス又は T セレクタの指定が間違っていないかどうかを確認してください。

KFRB02052-E

Error occurred in send-uoc, status = xx...xx. (E)

送信 UOC でエラーが発生しました。ステータス = xx...xx.

xx...xx : UOC が設定したステータスコード

(S)処理を中断します。

(O)ステータスコードを参照して、対処を決定してください。

KFRB02053-W

Connection was closed. Sender process retry connect to target system, reason = aa...aa.
(E)

回線が切断されました。回線の再接続を行います。理由 = aa...aa.

aa...aa : 回線切断理由

Close : close パケットを検知

Econnreset : 返信通知待ちで反映側システムのデータ受信プロセス停止を検知

Epipe : データ送信時に反映システムのデータ受信プロセス停止を検知

Que-full : 反映システムのキューファイル満杯を検知

other : 上記以外のエラーを検知

Timeout : 反映側システムからの応答なし※

注※

Timeout は、ネットワークが切断された場合にも出力することがあります。

(S)回線を再接続した後、処理を続行します。

KFRB02054-E

No received data in specified time. (C)

指定時間内にデータが送信されてきませんでした。

(S)処理を中断します。

(O)通信相手システムに異常がないかどうかを確認してください。このメッセージが抽出側システムで出力された場合は、該当する送信先を再起動してください。

KFRB02055-E

No sent data in specified time. (E)

指定時間内にデータを送信できませんでした。

(S)処理を中断します。

(O)通信相手システムに異常がないかどうかを確認してください。このメッセージが抽出システムで出力された場合は、該当する送信先を再起動してください。

KFRB02056-E

In spite of reach maximum retry times, unable to connect to target system, host name = aa...aa, service name = bb...bb, reason = cc...cc. (E)

最大リトライ回数に達したにもかかわらず、反映側システムと接続できませんでした。ホスト名 = aa...aa, サービス名 = bb...bb, 理由 = cc...cc.

aa...aa : ホスト名

bb...bb : サービス名

cc...cc : errno にセットされたエラー番号

(S)処理を中断します。

(O)反映側システムを起動した後、該当する送信先を再起動してください。

KFRB02057-E

In spite of reach maximum retry times, unable to connect to target system, NSAP address = aa...aa, Tselector = bb...bb, reason = cc...cc. (E)

最大リトライ回数に達したにもかかわらず、反映側システムと接続できませんでした。NSAP アドレス = aa...aa, Tセクタ = bb...bb, 理由 = cc...cc.

aa...aa : NSAP アドレス値

bb...bb : Tセクタ値

cc...cc : OSI の関数が返したエラー番号

(S)処理を中断します。

(O)反映側システムを起動した後、該当する送信先を再起動してください。

KFRB02058-W

Log data is PURGE TABLE of partitioned table. This log data is no out. table name = aa...aa. (E)

更新情報は分割表の PURGE TABLE です。この更新情報は無視します。テーブル名称 = aa...aa.

aa...aa : テーブル名

(S)処理を続行します。

(O)反映側 DB との間に不整合が発生します。反映側 DB との同期を取ってください。

KFRB02059-W

Log data is PURGE TABLE of partitioned table. This log data change to event data. event code = aa...aa, table name = bb...bb. (E)

更新情報は分割表の PURGE TABLE です。イベントデータに変更します。イベントコード = aa...aa, テーブル名称 = bb...bb.

aa...aa : イベントコード

bb...bb : テーブル名

(S)処理を続行します。

KFRB02060-W

Undetermined transaction deleted, count = xx...xx. (E)

未解決トランザクションを削除します。未解決トランザクション数 = xx...xx.

xx...xx : 削除した未解決トランザクション数

(S)処理を続行します。

KFRB02061-E

UOC for obtain ADT input data returned error status, UOC name = aa...aa, information = bb...bb. (E)

抽象データ型入力データ取得 UOC からエラーレベルのステータスが返されました。UOC 名 = aa...aa, 付加情報 = bb...bb.

aa...aa : UOC 名

bb...bb : 付加情報

(S)処理を中断します。

(O)付加情報を参照して問題を解決した後、該当する送信先を再起動してください。

KFRB02062-W

UOC for obtain ADT input data returned warning status, UOC name = aa...aa, information = bb...bb. (E)

抽象データ型入力データ取得 UOC から警告レベルのステータスが返されました。UOC 名 = aa...aa, 付加情報 = bb...bb.

aa...aa : UOC 名

bb...bb : 付加情報

(S)処理を続行します。

KFRB02063-W

Cannot accept restart event while processing restart event, rejected event = aa...aa. (C)

再起動イベント処理中のため再起動イベントは受け付けられません。イベントコード = aa...aa.

aa...aa : イベントコード

(S)処理を続行します。

KFRB02064-W

Stop request accepted from other process. (C)

他プロセスからの停止要求を受け付けました。

(S)終了処理を開始します。

KFRB02065-W

Unable to start data transmission for <send identifier : aa...aa>, because reduction status is continued. (E)

縮退状態継続のため送信先識別子 aa...aa の送信処理を開始できません。

aa...aa : 送信先識別子

(S)該当する送信先への送信処理を中断します。

(O)送信処理を再開する場合は、縮退状態の継続を解除した後に該当する送信先を再起動してください。

KFRB02066-W

DSID was not synchronized reflection group in target system. (C)

反映側システムでデータ連動識別子が同期反映グループになっていません。

次のどちらかの理由で、送信環境定義の reflect_mode オペランドを無効とします。

- 反映側システムでデータ連動識別子が同期反映グループになっていません。
- 反映側システムが同期反映グループをサポートしていません。

(S)反映トランザクション同期機能を使用しないで、処理を続行します。

(O)反映トランザクション同期機能を使用する場合は、反映側システムでデータ連動識別子を同期反映グループとして定義してください。反映側システムが同期反映グループをサポートしていない場合、反映側システムを同期反映グループをサポートしているバージョンにバージョンアップした後、同期反映グループを定義してください。

KFRB02072-E

Invalid extract data format for target system, extract data format kind = aa...aa, table name = bb...bb. (C)

反映側システムでサポートしていない不正な更新情報形式を検知しました。更新情報形式 = aa...aa, 表名 = bb...bb

aa...aa：更新情報形式の種別

1：スカラ関数 SUBSTR 使用時の、BLOB 型又は BINARY 型の後方削除更新ログ形式

bb...bb：抽出対象表の表名

付加情報としてマッピングキー値が出力されます。

(S)処理を続行します。

[対策]

付加情報として出力されるマッピングキー値の行で、スカラ関数 SUBSTR（後方削除更新）を使用した BLOB 型又は BINARY 型のデータが、抽出側 DB と反映側 DB で不整合になっています。次の手順で不整合を解消してください。

1. 抽出側 DB の更新業務を停止し、抽出された更新情報を反映側 DB にすべて反映します。
2. 反映側 Datareplicator を正常停止します。
3. 反映側 Datareplicator を、BLOB 型又は BINARY 型の後方削除更新をデータ連動できるバージョン (08-01 以降) にバージョンアップします。

反映側 Datareplicator を 08-01 以降にバージョンアップできない場合は、HiRDB を正常停止した後に、HiRDB のシステム共通定義の pd_rpl_func_control オペランドに指定している BACKWARD_CUTOFF_UPDATE を削除してください。その後、HiRDB を正常開始します。

4. このメッセージの付加情報として出力されるマッピングキー値を参考にして、抽出側 DB と反映側 DB で不整合となっている行を特定します。
5. 手順 4 で特定した抽出側 DB の行を反映側 DB の行に反映し、不整合を解消します。
6. 反映側 Datareplicator を正常開始します。

KFRB02073-W

```
Unable to send event code aa...aa.      (E)
イベントコード aa...aa は送信できません。
```

aa...aa : イベントコード

反映側システムが次のどちらかに該当するため、256 以上のイベントコードを送信できません。

- Datareplicator のバージョンが 08-09 以前である。
- XDM/DS である。

イベントコードの送信を行わず、処理を続行します。

(S)処理を継続します。

(O)このメッセージが出力された場合、次のどちらかの対応を行ってください。

- 反映側 Datareplicator のバージョンを 08-10 以降にバージョンアップしてください。
- 発行するイベントコードを 0 から 255 までとしてください。

KFRB02080-W

```
Detected invalid extract data, extract data kind = aa...aa, SQL kind = bb...bb, table name = cc...cc.      (C)
サポートしていない不正な更新情報を検知しました。更新情報種別 = aa...aa, SQL 種別 = bb...bb, 表名 = cc...cc
```

aa...aa : 更新情報の形式種別

1 : WITHOUT ROLLBACK オプションを指定している表を UPDATE 文以外で更新

bb...bb : SQL 種別

cc...cc : 抽出対象表の表名

付加情報としてマッピングキー値が出力されます。

(S)処理を続行します。

[対策]aa...aa が 1 の場合、抽出側 DB と反映側 DB で不整合が発生しています。次の手順で回復してください。

1. 抽出側 DB の更新業務を停止して、抽出された更新情報をすべて反映します。
2. 反映側システムを正常停止します。
3. このメッセージの付加情報として出力されるマッピングキー値を基に、抽出側 DB と反映側 DB で不整合となっている行を特定します。
4. 手順 3. で特定した抽出側 DB の行を反映側 DB に反映して、抽出側 DB と反映側 DB の同期を取ります。
5. 反映側システムを正常開始します。

KFRB02500-I

```
Send-master process started.      (E)  
送信マスタプロセスを起動しました。
```

(S)処理を続行します。

KFRB02501-I

```
Send-master process ended, status = aa...aa.    (E)  
送信マスタプロセスを終了しました。終了ステータス = aa...aa.
```

aa...aa : 終了ステータス

Normal : 正常停止

No-Process : 使える送信プロセスがないために停止

Signal : シグナル受信による停止

Error : エラー検知による停止

(S)処理を終了します。

KFRB02502-I

```
Data transmission for <send identifier : aa...aa> ended, status = bb...bb.    (E)  
送信先識別子 aa...aa への送信処理を終了しました。終了ステータス = bb...bb.
```

aa...aa : 送信先識別子

bb...bb : 終了ステータス

Auto-Normal : 停止ログ検知による正常停止

Manual-Normal : コマンドによる正常停止

Event : イベント検知停止

Auto-Force：停止ログ検知による強制停止

Manual-Force：コマンドによる強制停止

Hold：縮退による停止

Error：エラー検知による停止

(S)該当する送信処理を中断します。

KFRB02506-E

```
Found abnormal termination of sender process. (E)
```

送信プロセスの異常終了を検知しました。

(S)異常が発生した送信プロセスの処理を中断します。

(O)内部エラーです。このメッセージの直前に出力されたエラーメッセージを参照してください。エラーメッセージが出力されていない場合は、保守員に連絡してください。

KFRB02507-E

```
Unable to begin data transmission for <send identifier : aa...aa> (E)
```

送信先識別子 aa...aa への送信処理を開始できません。

aa...aa：送信先識別子

(S)処理を中断します。

(O)直前に出力されているメッセージの対処に従って、要因を取り除いてください。

KFRB02508-I

```
Transmission started for <send identifier : aa...a>. (E)
```

送信先識別子 aa...aa への送信処理を開始します。

aa...aa：送信先識別子

(S)処理を続行します。

KFRB02601-W

```
Warning from SQL output function, info = "aa...aa". (E)
```

更新 SQL 出力処理で警告を検知しました。詳細情報 = "aa...aa"

更新 SQL 出力処理の過程で警告を検知しました。1 行分の SQL 出力過程で複数の警告を検出した場合には、最後に検知した警告に対するメッセージだけが出力されます。

aa...aa : 詳細情報

メッセージ	意味
"because a data conversion error occurred, the value is printed with hexadecimal, datatype = 0xaa...aa."	パック形式データを文字列に変換する際にエラーが発生しました。 aa...aa : データコード*
"because of unsupported data type, 'aa...aa' was printed instead of the value, datatype = 0xbb...bb."	サポートしていない列データ型が存在しています。 aa...aa : 列データ型 *BLOB* : BLOB 型 *BINARY* : BINARY 型 *ADT* : ADT 型 bb...bb : データコード*
"because multi column was unsupported, 'aa...aa' was printed instead of the value."	繰返し列が存在しています。 aa...aa : *MCOL*

注※

データコードが示す列データ型は次のようになります。

- 0x65 : INTERVAL YEAR TO DAY
- 0x6F : INTERVAL HOUR TO SECOND
- 0x71 : DATE
- 0x79 : TIME
- 0x7d : TIMESTAMP
- 0x83 : ADT
- 0x91 : BINARY
- 0x93 : BLOB
- 0xE5 : DECIMAL

(S) 処理を続行します。

KFRB02602-E

Error occurred in SQL output function, info = "aa...aa". (E)
更新 SQL 出力処理でエラーを検知しました。詳細情報 = "aa...aa"

更新 SQL 出力処理の過程でエラーを検知しました。

aa...aa : 詳細情報

メッセージ	意味
fopen error, file = aa...aa, errno = bb...bb.	ファイル操作に失敗しました。エラー番号に基づいて、エラーの原因を取り除いてください。
fprintf error, file = aa...aa, errno = bb...bb.	

メッセージ	意味
fflush error, file = aa...aa, errno = bb...bb. sprintf error, errno = bb...bb.	エラー番号については、error.h 又は各 OS のマニュアルを参照してください。 aa...aa : ファイル名※ bb...bb : エラー番号
internal error, file = aa...aa, line = bb...bb, detail = cc...cc.	保守員に連絡してください。 aa...aa : ソースファイル名 bb...bb : 行番号 cc...cc : 内部コード

注※

抽出側 Datareplicator の運用ディレクトリからの相対パスです。

(S) 処理を中断します。

(O) 詳細情報に基づいて問題を解決した後、送信先を再起動してください。

KFRB02603-I

```
SQL statement is printed.( transaction(aa...aa, bb...bb) : insert(aa...aa, bb...bb) :
update(aa...aa, bb...bb) : delete(aa...aa, bb...bb) : purge(aa...aa, bb...bb) ) (E)
SQL 文を出力しました。(transaction (aa...aa, bb...bb) : insert (aa...aa, bb...bb) : update
(aa...aa, bb...bb) : delete (aa...aa, bb...bb) : purge (aa...aa, bb...bb))
```

出力した SQL 数が次のように出力されます。

aa...aa : 今回出力した件数

bb...bb : 出力件数の累計

(S) 処理を続行します。

KFRB02604-I

```
Event statement is printed, event code = aa...aa. (E)
イベント文を出力しました。イベントコード = aa...aa
```

イベント文を出力しました。

aa...aa : イベントコード

(S) 処理を続行します。

KFRB02605-W

```
Reduction status of <send identifier : aa...aa> is continued. (E)
```

送信先識別子 aa...aa の縮退状態を継続します。

送信先識別子 aa...aa の縮退状態を継続します。

aa...aa : 送信先識別子

(S) 該当する送信先に対する更新 SQL 出力処理を中断します。

KFRB03001-I

Reflect process initialization started. (C)

反映プロセスの初期化処理を開始します。

(S)処理を続行します。

KFRB03002-I

Reflect process is now starting database update, mode = [nn...nn, xx...x, m]. (C)

反映プロセスは反映側 DB の更新処理を開始します。開始モード = [nn...nn, xx...x, m].

nn...nn : 開始モード種別 1

Initial : 初期開始

Normal : 通常開始

Recover : リラン開始

xx...x : 反映抑止機能使用有無

Skip : 反映抑止機能を使用している。

表示なし : 反映抑止機能を使用していない。

m : 開始モード種別 2

N : 通常状態からの開始

C : クリティカル状態からの開始

(S)処理を続行します。

KFRB03004-W

INSERT operation might occur DB contradiction, table name = aa...aa. (C)

クリティカルセッション中の INSERT 操作により同一行が 2 行反映された可能性があります。表名 = aa...aa.

クリティカル状態での再開の INSERT 操作によって、同一行が二重に挿入された可能性があります。

aa...aa : 表名

(S)処理を続行します。

[対策]未反映情報ファイルを参照して、表の内容が正しいかどうかを確認してください。

KFRB03005-I

Reflect process ended critical session. Changing normal session. (C)

反映プロセスのクリティカルセッションを完了しました。

反映プロセスがクリティカル状態での反映処理を終了しました。通常状態での反映処理に推移します。

(S)処理を続行します。

KFRB03006-I

Reflect process caught event, event id = nn...nn. (C)

反映プロセスはイベントを検知しました。イベントコード = nn...nn.

nn...nn：イベントコード

イベントコードに「-1」が出力される場合は、抽出側システムからの稼働単位の終了要求を検知したことを示します。

(S)処理を続行します。

KFRB03007-W

Reflect process found automatically rollback transaction. Retrying the transaction. (C)

反映プロセスは暗黙的にロールバックされたトランザクションを検知しました。トランザクションを再実行します。

(S)処理を続行します。

[対策]排他が競合しています。参照系プログラムを調査して、排他が不要なときは無排他検索などに変更してください。

KFRB03008-I

Reflect process reached the end point of received data, file information = [mm...mm, nn...nn]. (C)

反映プロセスは受信した更新情報の終端に到達しました。ファイル情報 = [mm...mm, nn...nn].

反映プロセスは受信した更新情報の終端まで読み込みを完了しました。

mm...mm：ファイル ID

nn...nn：ファイルオフセット

(S)処理を続行します。

次のデータを受信するまで、監視状態になります。

KFRB03009-I

Reflect process completed commitment, reason = aa...aa, file information = [mm...mm, nn...nn]. (C)

反映プロセスは同期点処理を完了しました。理由コード = aa...aa, ファイル情報 = [mm...mm, nn...nn]
詳細情報

aa...aa : 同期点処理を実行した理由

COMMIT INTERVAL : cmtintvl 指定値に達した

DISCONNECT INTERVAL : discintvl 指定値に達した

MAX SQL NUMBER : reflect_trn_max_sqlnum 指定値に達した

EVENT INFORMATION: イベントを検知した

END OF CONNECTION : 抽出システムとの 1 回の送信単位終了を検知した

END POINT OF QUEUE : 更新情報の終端を検知した

PURGE TABLE REQUEST : PURGE TABLE を検知した※

AFTER PURGE TABLE : PURGE TABLE を検知した (PURGE TABLE 実行後)

DEFINITION REQUEST : 定義変更を検知した

COMMAND REQUEST : コマンド停止を検知した

SYNC EVENT : 同期イベントを検知した

注※

同期点処理は、契機となった PURGE TABLE を実行する前までの更新を対象とするため、このメッセージで出力される更新件数に PURGE TABLE は含まれません。

mm...mm : ファイル ID

nn...nn : ファイルオフセット

詳細情報 :

Read transaction count = xx...xx(yy...yy)

SQL count = [Insert:xx...xx(yy...yy), Update:xx...xx(yy...yy), Delete:xx...xx(yy...yy),

Purge:xx...xx(yy...yy), timestamp:xx...xx(yy...yy), Commit:xx...xx(yy...yy)]

Additional Transaction Info = zz...zz.

Read transaction count : 入力されたトランザクション数

xx...xx : 前回の同期点処理から入力されたトランザクション数

yy...yy : 反映プロセスの起動時から入力されたトランザクション数

SQL count : SQL 文の発行回数

Insert : Insert 文の発行回数

Update : Update 文の発行回数

Delete : Delete 文の発行回数

Purge : Purge 文の発行回数

timestamp : 時系列情報表への Insert 文の発行回数

Commit : Commit 文の発行回数

xx...xx : 前回の同期点処理からの発行回数

yy...yy : 反映プロセスの起動時からの発行回数

Additional Transaction Info : 抽出トランザクション情報 (zz...zz)

(S)処理を続行します。

同期点処理発行までの SQL 発行数を詳細情報に出力します。

KFRB03010-W

No row satisfying search condition in reflect table. Then the SQL statement skipped, table name = aa...aa, SQLKIND = bb...bb, database status = cc...cc. (C)

条件にあう行が反映表の中にないので、当該 SQL を無視しました。表名 = aa...aa, SQL 種別 = bb...bb, データベース状態 = cc...cc.

条件に合致する行が反映対象の表の中に存在しないため、該当する SQL を無視しました。

aa...aa : 表名

bb...bb : SQL 種別

cc...cc : データベース状態

NORMAL : 通常状態

CRITICAL : クリティカル状態

(S)処理を続行します。

[対策]未反映情報ファイルを参照して、表の内容が正しいかどうかを確認してください。

KFRB03011-I

Reflect process is now terminating database update, mode = nn...nn. (C)

反映プロセスの終了処理を開始します。終了モード = nn...nn.

nn...nn : 終了モードを示します。

Normal : 正常終了

Immediate：即時停止要求による終了

Event：イベント検知終了

Force：強制終了

Defserv：定義検知による終了

(S)処理を続行します。

KFRB03012-I

Reflect process terminated. (C)

反映プロセスを終了します。

(S)処理を終了します。

KFRB03013-I

Reflect process terminated abnormally. (C)

反映プロセスは、エラーを検知したため異常終了します。

(S)処理を終了します。

KFRB03014-I

Reflect process aborted due to unexpected extract data received, additional information = [n, nn...nn, nn...nn]. (C)

反映プロセスは、更新情報の異常を検知したため異常終了します。付加情報 = [n, nn...nn, nn...nn]

n, nn...nn, nn...nn：ファイルオフセット（内部情報）

(S)処理を中止します。

(O)コアダンプなど必要な資料を取得して、保守員に連絡してください。

KFRB03015-E

Unable to continue Reflect process due to abnormal termination of child process. (C)

子プロセスの終了を検知したため反映プロセスを続行できません。

(S)処理を終了します。

(O)SQL 実行プロセス又は UOC 実行プロセスで出力されたメッセージを確認してください。また、コアダンプなどがあれば、取得して保守員に連絡してください。

KFRB03016-E

Can not change number of group on rerun mode. (C)

リランモードではキーグループの変更はできません。

(S)処理を終了します。

[対策]反映定義を前回の実行時と同様に戻して、反映処理を再実行してください。

KFRB03017-I

Reflect process completed transaction recovery, kind = aa...aa, file information = [bb...bb, cc...cc] . (C)

反映プロセスはトランザクション回復を完了しました。種別 = aa...aa, ファイル情報 = [bb...bb, cc...cc]

aa...aa : 回復種別

COMMIT : コミット回復

COMMIT & RESTART : コミット回復とロールバック回復が混在

ROLLBACK : ロールバック回復

bb...bb 及び cc...cc : キューファイル情報

(S)処理を続行します。コミット回復だけのときは次のトランザクションから、そうでないときは前回実行中のトランザクション再実行から始めます。

KFRB03021-E

Bad extracted data received, data kind = n. (C)

更新情報の異常を検知しました。不正内容種別 = n.

n : 更新情報種別 (内部情報)

(S)処理を終了します (引き続き [KFRB03014-I](#) を出力します)。

KFRB03022-W

Not found Table-id in reflect definition, Table-id = nn...nn. (C)

反映定義にない表の更新情報を検知しました。表 ID = nn...nn.

nn...nn : テーブル ID (内部情報)

(表単位反映方式で実行中の場合には、複数の反映プロセスのうち、どれか一つから出力します)

(S)処理を継続します。

[対策]反映環境定義で defmerge=false 指定時は無視してください。defmerge=true 指定時は抽出定義と更新情報の不一致です。反映側 Datareplicator を停止して、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

KFRB03023-E

Row length between extracted data and extract definition unmatched, table name = aa...aa. (C)

抽出定義と行の長さが一致していない更新情報を検知しました。表名 = aa...aa.

aa...aa : 表名

(S)終了処理を開始します。

[対策]抽出定義と更新情報の不一致です。反映側 Datareplicator を停止して、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。また、抽出側システムの運用に誤りがなかったかどうかを確認してください。

KFRB03024-E

Bad column type found in extract definition, table name = aa...aa. (C)

抽出定義の列属性不正を検知しました。表名 = aa...aa.

aa...aa : 表名

(S)終了処理を開始します。

[対策]抽出定義と更新情報の不一致です。反映側 Datareplicator を停止して、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。また、抽出側システムの運用に誤りがなかったかどうかを確認してください。

KFRB03025-E

Could not convert to target database language, table name = aa...aa. (C)

更新情報を反映側 DB の文字コードに変換できません。表名 = aa...aa.

aa...aa : 表名

(S)終了処理を開始します。

[対策]反映処理を再実行してください。続けてこのメッセージが出力される場合は、保守員に連絡してください。

KFRB03026-E

Number of extracted column data is not equal to extract definition, table name = aa...aa. (C)

更新に必要な列が揃っていない更新情報を検知しました。表名 = aa...aa.

INSERT に必要な列の情報がそろっていない更新情報を検知しました。

aa...aa : 表名

(S)終了処理を開始します。

[対策]抽出定義と更新情報の不一致です。反映側 Datareplicator を停止して、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。また、抽出側システムの運用に誤りがなかったかどうかを確認してください。

KFRB03027-E

Multi-value/Array column data found, table name = aa...aa. (C)
繰返し列又は配列列を含む更新情報を検知しました。表名 = aa...aa.

aa...aa : 表名

(S)終了処理を開始します。

[対策]繰返し列 (05-02 よりも前のバージョンのとき) /配列列は反映できません。反映側 Datareplicator を停止して、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

繰返し列 又は配列列は、抽出定義から該当列を削除するか、又は UOC を作成して反映定義を変更してください。

KFRB03028-W

Data convert error occurred. Reflect process set null value and continue, extract data name = aa...aa, reflect table name = bb...bb, reflect column name = cc...cc. (C)
データ変換エラーが発生しました。反映プロセスは、不正な抽出データへ null 値を設定し処理を続行します。更新情報名 = aa...aa, 反映表名 = bb...bb, 反映列名 = cc...cc.

aa...aa : 更新情報名

bb...bb : 反映対象の表名

cc...cc : 反映対象の列名

(S)処理を続行します。

[対策]未反映情報ファイルを参照して、抽出データを特定して確認してください。必要であれば、手動でデータを更新してください。

KFRB03029-E

Data convert error occurred, extract data name = aa...aa, reflect table name = bb...bb, reflect column name = cc...cc. (C)
データ変換エラーが発生しました。更新情報名 = aa...aa, 反映表識別子 = bb...bb, 反映列名 = cc...cc.

aa...aa：更新情報名

bb...bb：反映対象の表名

cc...cc：反映対象の列名

(S)処理を中断します。

[対策]反映側 Datareplicator を停止して、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

このエラーが発生した列を未反映情報ファイルを参照して特定して、反映対象から削除するか、又は UOC で反映してください。

KFRB03030-W

Could not convert to target database language. Reflect process skipped this extract data, table name = aa...aa, column name = bb...bb. (C)

更新情報を反映側 DB の文字コードに変換できません。当該更新情報を無視して反映処理を続けます。表名 = aa...aa, 列名 = bb...bb.

詳細情報

aa...aa：抽出表名

bb...bb：抽出列名

詳細情報：文字コード変換エラーが発生したデータの先頭 50 バイトが、16 進数文字列で出力されます。

(S)処理を続行します。

(O)詳細情報を参照して、コード変換エラーの原因を調査してください。必要であれば、該当する抽出表を基に影響のあった反映表を再作成してください。

KFRB03031-E

HiRDB data base CONNECT error occurred, authorization id = aa...aa, SQLCODE = nn...nn. (C)

HiRDB への CONNECT 処理でエラーが発生しました。認可識別子 = aa...aa, SQLCODE = nn...nn.

詳細情報

HiRDB への CONNECT がエラーになりました。

aa...aa：認可識別子

nn...nn：SQL コード

詳細情報：エラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03032-E

```
HiRDB data base PREPARE error occurred, table name = aa...aa, SQLCODE = nn...nn.  
(C)  
SQL の PREPARE 時にエラーが発生しました。表名 = aa...aa, SQLCODE = nn...nn.  
詳細情報
```

HiRDB への PREPARE がエラーになりました。

aa...aa : 表名

nn...nn : SQL コード

詳細情報：マッピングキーの情報及びエラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03033-E

```
HiRDB data base EXECUTE error occurred, table name = aa...aa, SQLCODE = nn...nn.  
(C)  
SQL の EXECUTE 時にエラーが発生しました。表名 = aa...aa, SQLCODE = nn...nn.  
詳細情報
```

HiRDB への EXECUTE がエラーになりました。

aa...aa : 表名

nn...nn : SQL コード

詳細情報：マッピングキーの情報及びエラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03034-E

HiRDB data base EXECUTE IMMEDIATE error occurred, table name = aa...aa, SQLCODE = nn...nn. (C)

SQL の EXECUTE IMMEDIATE 時にエラーが発生しました。表名 = aa...aa, SQLCODE = nn...nn.

詳細情報

HiRDB への EXECUTE IMMEDIATE がエラーになりました。

aa...aa : 表名

nn...nn : SQL コード

詳細情報：マッピングキーの情報及びエラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03035-E

HiRDB data base COMMIT error occurred, SQLCODE = nn...nn. (C)

HiRDB の COMMIT 処理でエラーが発生しました。SQLCODE = nn...nn.

詳細情報

HiRDB への COMMIT がエラーになりました。

nn...nn : SQL コード

詳細情報：エラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03036-E

HiRDB data base DISCONNECT error occurred, SQLCODE = nn...nn. (C)

HiRDB の DISCONNECT 処理でエラーが発生しました。SQLCODE = nn...nn.

詳細情報

HiRDB への DISCONNECT がエラーになりました。

nn...nn : SQL コード

詳細情報：エラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03037-E

HiRDB data base ROLLBACK error occurred, SQLCODE = nn...nn. (C)

HiRDB の ROLLBACK 処理でエラーが発生しました。SQLCODE = nn...nn.

詳細情報

HiRDB への ROLLBACK がエラーになりました。

nn...nn：SQL コード

詳細情報：エラーが発生した理由を示す HiRDB のメッセージが出力されます。

(S)処理を中止します。

[対策]「10.5 SQL コードの詳細」を参照して、出力された SQL コードに該当する対処方法に従ってください。

KFRB03038-W

Update data for reflect column not exist in update information, extracted name = aa...aa, reflect table name = bb...bb. (C)

更新情報中に反映対象列の更新データが存在しません。更新情報名 = aa...aa, 反映表識別子 = bb...bb.

aa...aa：更新情報名

bb...bb：反映対象の表名

(S)反映列に対してデータを変更しない更新のため、この更新情報の反映をスキップして処理を続行します。

[対策]必要であれば、抽出定義、反映定義に誤りがないかどうかを確認してください。

KFRB03039-E

HiRDB data base SQL error occurred, SQLKIND = aa...aa. (C)

HiRDB への SQL 処理でエラーが発生しました。

aa...aa：SQL 種別

"TRG"：トリガ処理

"AUD"：監査証跡処理

(S)処理を中止します。

[対策]

エラーの原因を取り除いた後、再実行してください。

SQL 種別が AUD の場合

抽出システム定義、又は反映システム定義の hirdb_audit_trail オペランドの値を確認してください。監査証跡を取得する場合は、オペランドの指定値を変更してください

hirdb_audit_trail オペランドの指定に問題がない場合は、HiRDB のメッセージを確認してエラーの原因を取り除いてください。

KFRB03040-W

Reflect user own coding function returned warning code, process name = aa...aa, status = nn...nn. (C)

反映情報編集 UOC から警告レベルのステータスが返されました。プロセス名 = aa...aa, ステータスコード = nn...nn.

aa...aa : 反映 UOC プロセス名

nn...nn : UOC からのステータスコード

(S)処理を続行します。

[対策]反映情報編集 UOC の処理内容を調査して対策してください。

KFRB03041-E

Reflect user own coding function returned error code, process name = aa...aa, status = nn...nn. (C)

反映情報編集 UOC からエラーレベルのステータスが返されました。プロセス名 = aa...aa, ステータスコード = nn...nn.

aa...aa : 反映 UOC プロセス名

nn...nn : UOC からのステータスコード

(S)処理を中止します。

[対策]原因を取り除いた後、反映処理を再実行してください。

KFRB03042-W

HiRDB data base EXECUTE error occurred. Reflect process skip this SQL and continue, table name = nn...nn, SQLCODE = aa...aa. (C)

SQL の EXECUTE 時にエラーが発生しましたが、この SQL をスキップして処理を続行します。表名 = nn...nn, SQLCODE = aa...aa.

nn...nn : 反映表名

aa...aa : SQL コード

(S)処理を続行します。

KFRB03043-W

HiRDB data base EXECUTE IMMEDIATE error occurred. Reflect process skip this SQL and continue, table name = nn...nn, SQLCODE = aa...aa. (C)

SQL の EXECUTE IMMEDIATE 時にエラーが発生しましたが、この SQL をスキップして処理を続行します。表名 = nn...nn, SQLCODE = aa...aa.

nn...nn : 反映表名

aa...aa : SQL コード

(S)処理を続行します。

KFRB03044-E

Column data UOC function returned error code, UOC function name = xx...xx, reflection table name = mm...mm, reflection column name = nn...nn, status = aa...aa. (C)

列データ編集 UOC 関数からエラーレベルのステータスが返されました。UOC 関数名 = xx...xx, 反映表名 = mm...mm, 反映列名 = nn...nn, ステータスコード = aa...aa.

xx...xx : 列データ編集 UOC 関数名

mm...mm : 反映表名

nn...nn : 反映列名

aa...aa : 列データ編集 UOC からのステータス

(S)処理を中止します。

[対策]原因を取り除いた後、反映処理を再実行してください。

KFRB03045-E

Invalid column data type set in column data UOC function, UOC function name = xx...xx, reflection table name = mm...mm, reflection column name = nn...nn. (C)

列データ編集 UOC 関数で不正な列データタイプが設定されました。UOC 関数名 = xx...xx, 反映表名 = mm...mm, 反映列名 = nn...nn.

xx...xx：列データ編集 UOC 関数名

mm...mm：反映表名

nn...nn：反映列名

(S)処理を中止します。

[対策]原因を取り除いた後、反映処理を再実行してください。

KFRB03046-E

Invalid column data length set in column data UOC function, UOC function name = xx...xx, reflection table name = mm...mm, reflection column name = nn...nn. (C)

列データ編集 UOC 関数で不正なデータ長が設定されました。UOC 関数名 = xx...xx, 反映表名 = mm...mm, 反映列名 = nn...nn.

xx...xx：列データ編集 UOC 関数名

mm...mm：反映表名

nn...nn：反映列名

(S)処理を中止します。

[対策]原因を取り除いた後、反映処理を再実行してください。

KFRB03047-W

HiRDB data base PREPARE error occurred. Reflect process skip this SQL and continue, table name = nn...nn, SQLCODE = aa...aa. (C)

SQL の PREPARE 時にエラーが発生しましたが、この SQL をスキップして処理を続行します。表名 = nn...nn, SQLCODE = aa...aa.

nn...nn：反映表名

aa...aa：SQL コード

(S)処理を続行します。

KFRB03048-E

HiRDB data base set 'W' in SQLWARN7, table name = nn...nn. (C)

SQL 実行時に SQLWARN7 に 'W' が設定されました。表名 = nn...nn.

[詳細情報]：マッピングキー情報

繰返し列への反映時に、対象行の繰返し列に指定した要素が存在しないため、反映が無視されました。

nn...nn : 反映表名

(S)処理を中止します。

[対策]原因を取り除いて、反映処理を再実行してください

KFRB03049-W

HiRDB data base set 'W' in SQLWARN7. Reflect process skipped this SQL and continue, table name = nn...nn. (C)

SQL 実行時に SQLWARN7 に 'W' が設定されましたが、この SQL をスキップして処理を続行します。表名 = nn...nn.

[詳細情報] : マッピングキー情報

繰返し列への反映時に、対象行の繰返し列に指定した要素が存在しないため反映が無視されましたが、この SQL をスキップして処理を続けます。

nn...nn : 反映表名

(S)処理を続行します。

KFRB03051-E

Data base CONNECT error occurred, data base = aa...aa, authorization id = nn...nn, error code = bb...bb. (C)

DBMS への CONNECT 処理でエラーが発生しました。DBMS 種別 = aa...aa, 認可識別子 = nn...nn, DBMS エラーコード = bb...bb.

aa...aa : DBMS 種別

nn...nn : 認可識別子

bb...bb : DBMS エラーコード

(S)処理を中止します。

[対策]DBMS のマニュアルを参照してエラーの原因を取り除いてから、反映処理を再実行してください。

KFRB03052-E

Data base PREPARE error occurred, data base = aa...aa, table name = nn...nn, error code = bb...bb. (C)

DBMS で SQL 文解析エラーが発生しました。DBMS 種別 = aa...aa, 表名 = nn...nn, DBMS エラーコード = bb...bb.

aa...aa : DBMS 種別

nn...nn : 認可識別子.表名

bb...bb : DBMS エラーコード

(S)処理を中止します。

[対策]DBMS のマニュアルを参照してエラーの原因を取り除いてから、反映処理を再実行してください。

KFRB03053-E

Data base EXECUTE error occurred, data base = aa...aa, table name = nn...nn, error code = bb...bb. (C)

DBMS への SQL 実行処理でエラーが発生しました。DBMS 種別 = aa...aa, 表名 = nn...nn, DBMS エラーコード = bb...bb.

aa...aa : DBMS 種別

nn...nn : 認可識別子.表名

bb...bb : DBMS エラーコード

(S)処理を中止します。

[対策]DBMS のマニュアルを参照してエラーの原因を取り除いてから、反映処理を再実行してください。

KFRB03054-E

Data base ROLLBACK error occurred, data base = aa...aa, error code = bb...bb. (C)

DBMS への ROLLBACK 処理でエラーが発生しました。DBMS 種別 = aa...aa, DBMS エラーコード = bb...bb.

aa...aa : DBMS 種別

nn...nn : 認可識別子

bb...bb : DBMS エラーコード

(S)処理を中止します。

[対策]DBMS のマニュアルを参照してエラーの原因を取り除いてから、反映処理を再実行してください。

KFRB03055-E

Data base COMMIT error occurred, data base = aa...aa, error code = bb...bb. (C)

DBMS への COMMIT 処理でエラーが発生しました。DBMS 種別 = aa...aa, DBMS エラーコード = bb...bb.

aa...aa : DBMS 種別

bb...bb : DBMS エラーコード

(S)処理を中止します。

[対策]DBMS のマニュアルを参照してエラーの原因を取り除いてから、反映処理を再実行してください。

KFRB03056-E

Data base DISCONNECT error occurred, data base = aa...aa, error code = bb...bb. (C)
DBMS への DISCONNECT 処理でエラーが発生しました。DBMS 種別 = aa...aa, DBMS エラーコード = bb...bb.

aa...aa : DBMS 種別

bb...bb : DBMS エラーコード

(S)処理を中止します。

[対策]DBMS のマニュアルを参照してエラーの原因を取り除いてから、反映処理を再実行してください。

KFRB03057-W

Data base EXECUTE error occurred, Reflect process skip this SQL and continue, data base = aa...aa, table name = bb...bb, error code = cc...cc. (C)
DBMS への SQL 実行処理でエラーが発生しましたが、この SQL をスキップして処理を続行します。
DBMS 種別 = aa...aa, 反映表識別子 = bb...bb, DBMS エラーコード = cc...cc.

aa...aa : DBMS 種別

bb...bb : 表名

cc...cc : DBMS エラーコード

(S)処理を続行します。

KFRB03058-W

Data base CONNECT error occurred. Retry connection to database, data base = aa...aa, authorization id = bb...bb, error code = cc...cc. (C)
DBMS への CONNECT 処理でエラーが発生しました。再度 CONNECT 処理を行ないます。DBMS 種別 = aa...aa, 認可識別子 = bb...bb, DBMS エラーコード = cc...cc.

aa...aa : DBMS 種別

bb...bb : 認可識別子

cc...cc : DBMS エラーコード

(S)処理を続行します。

KFRB03060-W

Reflect process terminated because HiRDB data base set 'W' in SQLWARN6, table name = nn...nn, SQLCODE = aa...aa. (C)

スキップ対象の SQLCODE の SQL エラーが発生しましたが、SQLWARN6（暗黙的ロールバック）が発生したため、処理を停止します。

nn...nn：反映表名

aa...aa：SQL コード

(S)処理を中止します。

(O)SQL エラーの原因を取り除いて反映処理を再実行してください。

KFRB03061-E

Reflect process terminated, reason = during rollback. (C)

直前に実行したトランザクションはロールバック中のおそれがあるため、処理を停止します。

スキップ対象の SQL エラーが発生しましたが、HiRDB が暗黙的ロールバック中のおそれがあります。

ロールバック中に反映処理を再実行すると抽出側 DB と反映側 DB の不整合が発生するおそれがあるため、処理を停止します。

スキップ対象の SQL エラーは次に示すエラーです。

- 反映環境定義の skip_sqlcode オペランドに指定した SQLCODE の SQL エラー

(S)処理を中止します。

(O) pdls -d trn コマンドで PROGRAM が「hdssqle」で始まるトランザクションが存在しなくなったことを確認し、反映処理を起動してください。

KFRB03070-E

Error occurred in XA mode start. (C)

XA モードの開始でエラーが発生しました。

(S)処理を中断します。

(O)反映先 HiRDB のメッセージを確認して、エラーの原因を取り除いた後、再実行してください。

KFRB03071-E

Commitment prepare error occurred in transaction, XID = aa...aa. (C)

トランザクションのコミット準備でエラーが発生しました。XID = aa...aa.

Datareplicator のトランザクションが、HiRDB に「FORGETTING」状態[※]で残っている時に、hdsstart -i を実行し、反映側 Datareplicator を初期化しています。

aa...aa : トランザクション識別子

(S)処理を中断します。

(O)HiRDB に「FORGETTING」状態で残っているトランザクションを HiRDB の pdfgt コマンドで決着した後、反映側 Datareplicator を再起動してください。

注※

反映システム定義で commitment_method=fxa_sqlc を指定している場合、反映処理がエラー停止すると、HiRDB に「FORGETTING」状態のトランザクションが残ります。

KFRB03072-E

Commitment error occurred in transaction, XID = aa...aa. (C)

トランザクションのコミットでエラーが発生しました。XID = aa...aa.

aa...aa : トランザクション識別子

(S)処理を中断します。

(O)反映先 HiRDB のメッセージを確認して、エラーの原因を取り除いた後、再実行してください。

KFRB03073-E

Rollback error occurred in transaction, XID = aa...aa. (C)

トランザクションのロールバックでエラーが発生しました。XID = aa...aa.

aa...aa : トランザクション識別子

(S)処理を中断します。

(O)反映先 HiRDB のメッセージを確認して、エラーの原因を取り除いた後、再実行してください。

KFRB03074-E

Error occurred in HiRDB interface function, function = aa...aa, value = bb...bb. (C)

HiRDB インタフェース関数でエラーが発生しました。関数名 = aa...aa, 戻り値 = bb...bb.

aa...aa : 関数名称

bb...bb : 関数の戻り値

(S)処理を中断します。

(O)保守員に連絡してください。

KFRB03075-E

```
HiRDB don't support 2 phase commitment. (C)  
反映先 HiRDB は 2 相コミット方式に対応していません。
```

反映システム定義で `commit_method` に `fxa_sqlc` 又は `fxa_all` を指定していますが、反映先 HiRDB は 2 相コミットをサポートしていません。環境変数の `PATH`, `SHLIB_PATH` が正しく指定されていなかったり、メモリ不足でライブラリをローディングできなかつたりした可能性もあります。

(S)処理を中断します。

(O)HiRDB のバージョンと `commit_method` の指定、環境変数 `PATH`, `SHLIB_PATH` の指定を確認した後、再実行してください。

KFRB03076-W

```
HiRDB cannot accept the recovery request of the transaction. Retrying the recovery request,  
XID = aa...aa, function name = bb...bb, return = cc...cc. (C)  
HiRDB がトランザクションの回復要求を受け付けられませんでした。回復要求を再実行します。XID  
= aa...aa, 関数名 = bb...b, 戻り値 = cc...cc
```

aa...aa : トランザクション識別子

bb...bb : 関数名称

cc...cc : 関数の戻り値

(S)処理を続行します。

KFRB03077-I

```
HiRDB accepted the recovery request of the transaction, XID = aa...aa, function name =  
bb...bb, return = cc...cc. (C)  
HiRDB がトランザクションの回復要求を受け付けました。XID = aa...aa, 関数名 = bb...bb, 戻り値  
= cc...cc
```

aa...aa : トランザクション識別子

bb...bb : 関数名称

cc...cc : 関数の戻り値

(S)処理を続行します。

KFRB03078-E

In spite of reach maximum retry times, HiRDB cannot accept the recovery request of the transaction, XID = aa...aa, function name = bb...bb, return = cc...cc, retry count = dd...dd.
(C)

最大リトライ回数に達したにもかかわらず、HiRDB がトランザクションの回復要求を受け付けられませんでした。XID = aa...aa, 関数名 = bb...bb, 戻り値 = cc...cc, リトライ回数 = dd...dd

aa...aa : トランザクション識別子

bb...bb : 関数名称

cc...cc : 関数の戻り値

dd...dd : リトライ回数

(S)処理を中断します。

[対策]反映環境定義の xa_recovery_retry_count オペランド, 及び xa_recovery_retry_interval オペランドの指定値を大きくしてください。その後, 反映側 HiRDB が開始してから, 反映処理を再開始してください。

KFRB03080-E

Environment handle allocation error occurred. (C)

ODBC 環境ハンドル設定中にエラーが発生しました。

(S)処理を中止します。

[対策]ODBC のマニュアルを参照してエラーの原因を取り除いた後, 反映処理を再実行してください。

KFRB03081-E

Connection handle allocation error occurred. SQLSTATE = aa...aa. (C)

ODBC 接続ハンドル設定中にエラーが発生しました。SQLSTATE = aa...aa.

aa...aa : SQLSTATE

(S)処理を中止します。

[対策]ODBC のマニュアルを参照してエラーの原因を取り除いた後, 反映処理を再実行してください。

KFRB03082-E

Statement handle allocation error occurred. SQLSTATE = aa...aa. (C)

ODBC ステートメントハンドル設定中にエラーが発生しました。SQLSTATE = aa...aa.

aa...aa : SQLSTATE

(S)処理を中止します。

[対策]ODBC のマニュアルを参照してエラーの原因を取り除いた後、反映処理を再実行してください。

KFRB03083-E

Unable to get information of ODBC driver, SQLSTATE = aa...aa. (C)
ODBC ドライバの情報が読み込めませんでした。SQLSTATE = aa...aa.

aa...aa : SQLSTATE

(S)処理を中止します。

[対策]ODBC のマニュアルを参照してエラーの原因を取り除いた後、反映処理を再実行してください。

KFRB03084-E

Unable to set attributes of connections, Attribute = aa...aa, Value = bb...bb, SQLSTATE = cc...cc. (C)
ODBC 接続属性の変更ができませんでした。属性 = aa...aa, 値 = bb...bb, SQLSTATE = cc...cc.

Connect 属性の設定ができませんでした。

aa...aa : 属性

bb...bb : 値

cc...cc : SQLSTATE

(S)処理を中止します。

[対策]ODBC のマニュアルを参照してエラーの原因を取り除いた後、反映処理を再実行してください。

KFRB03085-W

ODBC function returned warning code, function name = aa...aa, SQLSTATE = bb...bb. (C)
ODBC 関数で警告が発生しました。ODBC 関数名 = aa...aa, SQLSTATE = bb...bb.

ODBC 関数が警告メッセージを返しました。

aa...aa : ODBC 関数名

bb...bb : SQLSTATE

(S)処理を続行します。

KFRB03086-E

Data base CONNECT error occurred, datasource = aa...aa, authorization id = bb...bb, SQLSTATE = cc...cc, error code = dd...dd. (C)

ODBC への CONNECT 処理でエラーが発生しました。データソース名 = aa...aa, 認可識別子 = bb...bb, SQLSTATE = cc...cc, SQLCODE = dd...dd.

aa...aa : データソース名

bb...bb : 認可識別子

cc...cc : SQLSTATE

dd...dd : DBMS エラーコード

(S)処理を中止します。

[対策]ODBC のマニュアルを参照してエラーの原因を取り除いた後、反映処理を再実行してください。

KFRB03087-E

Invalid ODBC version of ODBC driver, version = aa...aa. (C)

ODBC ドライバの ODBC バージョンが不正です。ODBC バージョン = aa...aa.

aa...aa : ODBC のバージョン

(S)処理を中止します。

[対策]使っている ODBC ドライバのサポートバージョンを確認してください。

KFRB03088-E

Invalid ODBC API conformance level of ODBC driver, conformance level = aa...aa. (C)

ODBC ドライバの ODBC 合致レベルが不正です。ODBC 合致レベル = aa...aa.

aa...aa : 合致レベル

(S)処理を中止します。

[対策]使っている ODBC ドライバの合致レベルを確認してください。

KFRB03089-E

ODBC statement handle free error occurred, SQLSTATE = aa...aa, OPTION = bb...bb. (C)

ODBC ステートメントハンドル開放中にエラーが発生しました。SQLSTATE = aa...aa, OPTION = bb...bb.

aa...aa : SQLSTATE

bb...bb : SQLFreeStmt()のオプション

(S)処理を中止します。

[対策]ODBCのマニュアルを参照してエラーの原因を取り除いた後、反映処理を再実行してください。

KFRB03090-E

ODBC function returned error code, function name = aa...aa, SQLSTATE = bb...bb. (C)
ODBC 関数でエラーが発生しました。ODBC 関数名 = aa...aa, SQLSTATE = bb...bb.

aa...aa : ODBC 関数名

bb...bb : SQLSTATE

(S)処理を中止します。

[対策]ODBCのマニュアルを参照してエラーの原因を取り除いた後、反映処理を再実行してください。

KFRB03091-W

Unable to set ODBC timeout, SQLSTATE = aa...aa. (C)
ODBC へのタイムアウト時間設定を失敗しました。SQLSTATE = aa...aa.

ODBC 関数のタイムアウト時間を設定できません。

aa...aa : SQLSTATE

(S)デフォルトの動作 (タイムアウトなし) で処理を続行します。

KFRB03092-W

Data base CONNECT error occurred, Retry connection to database, datasource = aa...aa, authorization id = bb...bb, SQLSTATE = cc...cc, error code = dd...dd. (C)
ODBC への CONNECT 処理でエラーが発生しました。再度 CONNECT 処理を行ないます。データソース名 = aa...aa, 認可識別子 = bb...bb, SQLSTATE = cc...cc, SQLCODE = dd...dd.

aa...aa : データソース名

bb...bb : 認可識別子

cc...cc : SQLSTATE

dd...dd : エラーコード

(S)処理を続行します。

KFRB03094-W

HiRDB data base SQL error occurred, SQLKIND = aa...aa. (C)

HiRDB への SQL 処理でエラーが発生しました。SQLKIND = aa...aa.

HiRDB への SQL 処理でエラーが発生しましたが、このエラーを無視して処理を続行します。

aa...aa : SQL 種別

"AUD" : 監査証跡処理

(S)処理を続行します。

[対策]

エラーの原因を取り除いた後、再実行してください。

SQL 種別が AUD の場合

- 反映システム定義の hirdb_audit_trail オペランドの値を確認してください。監査証跡を取得する場合は、オペランドの指定値を none_cont, 又は none_stop 以外に変更してください
- 反映情報編集 UOC で HiRDB へ SQL を発行しない場合、反映システム定義の hirdb_audit_trail オペランドの値を none_cont, 又は none_stop 以外に変更するか、反映情報編集 UOC の実行ファイル作成時に HiRDB のライブラリをリンクしないでください。
- 反映情報編集 UOC で HiRDB へ SQL を発行する場合、作成規則に反した SQL が発行されていないかどうか確認し、必要に応じて反映情報編集 UOC を修正してください。

上記に問題がない場合は、HiRDB のメッセージを確認してエラーの原因を取り除いてください。

KFRB03095-W

Column data UOC function returned warning code, UOC function name = xx...xx, reflection table name = mm...mm, reflection column name = nn...nn, status = aa...aa. (C)

列データ編集 UOC 関数から警告レベルのステータスが返されました。UOC 関数名 = xx...xx, 反映表名 = mm...mm, 反映列名 = nn...nn, ステータスコード = aa...aa.

xx...xx : 列データ編集 UOC 関数名

mm...mm : 反映表名

nn...nn : 反映列名

aa...aa : 列データ編集 UOC からのステータス

(S)処理を続行します。

[対策]列データ編集 UOC の処理内容を調査して対策してください。

KFRB03101-W

Row not found at UPDATE request and now changing INSERT, extracted name = aa...aa, reflect table name = bb...bb. (C)

UPDATE 対象の行が存在しません。INSERT に変換します。更新情報名 = aa...aa, 反映表名 = bb...bb.

aa...aa : 更新情報名

bb...bb : 反映対象表名

付加情報としてマッピングキー値を出力します。

(S)処理を続行します。

KFRB03102-W

Row already exists at INSERT request and now changing UPDATE, extracted name = aa...aa, reflect table name = bb...bb. (C)

INSERT 対象の行は既に存在します。UPDATE に変換します。更新情報名 = aa...aa, 反映表名 = bb...bb.

aa...aa : 更新情報名

bb...bb : 反映対象表名

付加情報としてマッピングキー値を出力します。

(S)処理を続行します。

KFRB03103-W

Change DELETE request to UPDATE (NULL value) because table merge option, extracted name = aa...aa, reflect table name = bb...bb. (C)

マージ表オプションにより、DELETE を UPDATE (NULL 値) に変換します。更新情報名 = aa...aa, 反映表名 = bb...bb.

aa...aa : 更新情報名

bb...bb : 反映対象表名

付加情報としてマッピングキー値を出力します。

(S)処理を続行します。

KFRB03104-E

Conversion failed from INSERT to UPDATE statement, extracted name = aa...aa, reflect table name = bb...bb. (C)

INSERT 対象行を UPDATE に変換できません。更新情報名 = aa...aa, 反映表名 = bb...bb.

aa...aa : 更新情報名

bb...bb : 反映対象表名

付加情報としてマッピングキー値を出力します。また、HDSPATH ディレクトリ下に簡易ダンプを出力します。

(S)処理を終了します。

(O)簡易ダンプを保存した後、保守員に連絡してください。

KFRB03105-E

Conversion failed from DELETE to UPDATE statement, extracted name = aa...aa, reflect table name = bb...bb. (C)

DELETE 対象行を UPDATE に変換できません。更新情報名 = aa...aa, 反映表名 = bb...bb.

aa...aa : 更新情報名

bb...bb : 反映対象表名

付加情報としてマッピングキー値を出力します。また、HDSPATH ディレクトリ下に簡易ダンプを出力します。

(S)処理を終了します。

(O)簡易ダンプを保存した後、保守員に連絡してください。

KFRB03106-W

Though the DELETE request was changed to UPDATE(NULL value) request, the UPDATE(NULL value) request was skipped because of no row satisfying search condition, extracted name = aa...aa, reflect table name = bb...bb. (C)

DELETE を UPDATE (NULL 値) に変換しましたが、条件に合う行が反映表の中にないので、UPDATE (NULL 値) を無視しました。更新情報名 = aa...aa, 反映表名 = bb...bb

マージ表の sqlconvopt2 オプションが指定されたため、DELETE を NULL 値の UPDATE に変換して SQL を発行しました。しかし、条件に合う行が反映表の中になかったため、この SQL を無視しました。

aa...aa : 更新情報名

bb...bb : 反映対象表名

付加情報としてマッピングキー値を出力します。

(S)処理を続行します。

KFRB03110-W

Log data is ADT attribute data specified INSERT. Reflect process set null value and continue, extract data name = aa...aa, reflect table name = bb...bb, reflect column name = cc...cc. (C)

更新情報は抽象データ型の属性データを指定した INSERT です。反映プロセスは、不正な抽出データへ null 値を設定し処理を続行します。更新情報名 = aa...aa, 反映表名 = bb...bb, 反映列名 = cc...cc.

時系列情報表やマージ表オプションの指定によって、抽象データ型の属性指定による INSERT が実行されました。反映プロセスは不正な抽出データへ null 値を設定した後、反映処理を続行します。

aa...aa : 更新情報名

bb...bb : 反映対象表名

cc...cc : 反映列名

(S)処理を終了します。

(O)反映エラー情報ファイルを参照して抽出データを特定した後、抽出データを確認してください。必要であれば手動でデータを更新してください。

KFRB03111-E

Not exist hash function library, library name = aa...aa. (C)

ハッシュ関数ライブラリが存在しません。ライブラリ名 = aa...aa.

aa...aa : ライブラリ名

(S)処理を終了します。

[対策]エラーファイルなど必要な資料を取得した後、保守員に連絡してください。

KFRB03112-E

Failed to load hash function library, library name = aa...aa, errno = bb...bb. (C)

ハッシュ関数ライブラリのロードに失敗しました。ライブラリ名 = aa...aa, エラー番号 = bb...bb.

aa...aa : ライブラリ名

bb...bb : エラー番号

(S)処理を終了します。

[対策]エラーファイルなど必要な資料を取得した後、保守員に連絡してください。

KFRB03113-E

Failed to find hash function symbol, function name = aa...aa, errno = bb...bb. (C)

ハッシュ関数のシンボル取得に失敗しました。関数名 = aa...aa, エラー番号 = bb...bb.

aa...aa : 関数名

bb...bb : エラー番号

(S)処理を終了します。

[対策]エラーファイルなど必要な資料を取得した後、保守員に連絡してください。

KFRB03114-E

Hash function error occurred, return code = aa...aa. (C)

ハッシュ関数がエラー終了しました。リターンコード = aa...aa.

aa...aa : リターンコード

(S)処理を終了します。

[対策]エラーファイルなど必要な資料を取得した後、保守員に連絡してください。

KFRB03115-E

Invalid hash value returned from hash function, RDAREA number = aa...aa, RDAREA index = bb...bb. (C)

ハッシュ関数が不正な RD エリア指定順序を返しました。RD エリア数 = aa...aa, RD エリア指定順序 = bb...bb.

aa...aa : RD エリア数

bb...bb : RD エリア指定順序

(S)処理を終了します。

[対策]エラーファイルなど必要な資料を取得した後、保守員に連絡してください。

KFRB03116-W

Null value set in hash key column, hash key column name = aa...aa. (C)

ハッシュのキー列に NULL 値が設定されました。ハッシュキー列名称 = aa...aa.

aa...aa : ハッシュキー列名称

(S)該当する更新情報は反映させないで、処理を続行します。

KFRB03117-E

```
Failed to get hash key column data, hash key column name = aa...aa. (C)  
ハッシュのキー列の値の取得に失敗しました。ハッシュキー列名称 = aa...aa.
```

aa...aa：ハッシュキー列名称

(S)処理を終了します。

[対策]エラーファイルなど必要な資料を取得した後、保守員に連絡してください。

KFRB03201-I

```
Definition Server process started. mode = nn...nn. (C)  
反映定義サーバプロセスを開始します。開始モード = nn...nn.
```

nn...nn：開始モードの種別

Initial：初期開始

Normal：通常開始

Recover：リラン開始

(S)処理を続行します。

KFRB03202-I

```
Definition Server process terminated, status = nn...nn, addinfo = xx...xx. (C)  
反映定義サーバプロセスを終了します。ステータス = nn...nn, 詳細情報 = xx...xx.
```

nn...nn：終了ステータス

Initial：初期状態（再起動時には初期化から開始します）

Normal：正常終了状態

Immediate：即時終了状態

Event：イベント終了状態

Error：エラー終了状態（再起動時にはリランを実行します）

xx...xx：詳細情報（内部情報）

(S)処理を終了します。

[対策]終了ステータスが Initial 又は Error の場合は、エラーが発生している可能性があります。先に出力されているエラーメッセージを確認してください。エラーが発生しているときには、原因を取り除いた後、反映処理を再実行してください。

KFRB03203-E

Definition Server process terminated abnormally, module = aa...aa, line = nn...nn, addinfo = xx...xx. (C)

反映定義サーバプロセスの異常検知により終了します。モジュール名 = aa...aa, 行番号 = nn...nn, 詳細情報 = xx...xx.

aa...aa : モジュール名

nn...nn : 行番号

xx...xx : 詳細情報 (内部情報)

(S)処理を中止します。

(O)コアダンプなど必要な資料を取得した後、保守員に連絡してください。

KFRB03204-I

Now, watching reflect process. (C)

反映プロセスを監視します。

(S)処理を続行します。

KFRB03205-I

Found abnormal termination of Reflect process, group name = aa...aa. (C)

反映プロセスの異常終了を検知しました。グループ名 = aa...aa.

aa...aa : グループ名

(S)処理を続行します。

KFRB03206-E

Reflection-env file parsing error occurred. (C)

反映環境定義ファイルを解析中にエラーが発生しました。

(S)終了処理を開始します。

[対策]反映環境定義ファイルに指定しているオペランドを修正してください。

KFRB03207-E

Internal error occurred in definition server, factor code = aa...aa, addinfo = [mm...mm, nn...nn]. (C)

反映定義サーバプロセスで内部エラーが発生しました。要因コード = aa...aa, 詳細情報 = [mm...mm, nn...nn].

aa...aa : 要因コード

mm...mm, nn...nn : 詳細情報 (内部情報)

(S)処理を終了します。

[対策]反映エラー情報ファイルなどの必要な資料を取得した後, 保守員に連絡してください。

KFRB03208-I

Defer start of reflection, deferred time = aa...aa minutes, reflect mode = nn...nn. (C)

反映処理の開始を遅延します。遅延時間 = aa...aa 秒, 遅延開始後反映処理モード = nn...nn.

aa...aa : 遅延時間

(0 分の場合, 反映処理の再起動要求まで遅延します)

nn...nn : 遅延開始後の反映処理の方式

TBL : 表単位反映方式

TRN : トランザクション単位反映方式

SPD : 未決状態 (breaktime, breakmode オペランドが指定されていないため, 反映処理の再起動を要求したときに決定します)

(S)処理を続行します。

KFRB03209-I

Initialization completed in reflection. (C)

反映処理の初期化を完了しました。

(S)処理を続行します。

KFRB03210-I

Accepted stop request in reflector, request kind = aa...aa. (C)

反映処理において停止要求を受け付けました。要求種別 = aa...aa.

aa...aa : 要求種別

Normal :

hdsstop コマンドを受け付けたため, 最終の送信終了通知までの反映を完了した後, 処理を終了します。

Event :

hdsstop -t event コマンドを受け付けたため、これ以降最初に出現するイベントまで反映した後、処理を終了します。

Force :

hdsstop -t force コマンドを受け付けたため、データベースの整合が取れるまで反映した後、処理を終了します。

Spd :

hdsrfctl -m spd コマンドを受け付けたため、これ以降最初に出現するイベントまで反映した後、処理を終了します。

S-Immediate :

hdsstop -t immediate コマンドを受け付けたため、反映プロセスの終了を待ちます。

R-Immediate :

hdsrfctl -d データ連動識別子 -m immediate コマンドを受け付けたため、反映プロセスの終了を待ちます。

G-Immediate :

hdsrfctl -g 同期反映グループ名 -m immediate コマンドを受け付けたため、反映プロセスの終了を待ちます。

(S)処理を続行します。

KFRB03211-I

All reflect process caught event, event-id = aa...aa. (C)

全ての反映プロセスがイベントに到達しました。イベント ID = aa...aa.

aa...aa : イベント ID

(S)処理を続行します。

KFRB03213-W

Reflection skip information, extract id = aa...aa. (C)

指定された更新情報の反映を抑止しました。更新情報識別子 = aa...aa

aa...aa : 反映を抑止した更新情報識別子

(S)処理を続行します。

KFRB03214-E

Reflection skip list file definition error, file name = aa...aa, line no = bb...bb, reason = cc...cc. (C)

反映抑止対象リストファイルの定義エラーを検知しました。ファイル名 = aa...aa, 行番号 = bb...bb

aa...aa : 反映抑止対象リストファイル名

bb...bb : エラー行番号

cc...cc : 理由コード

11 : 解析エラー

12 : 登録重複エラー

13 : 未サポート制御コード検出

(S)処理を終了します。

(O)反映抑止対象リストファイルを修正してください。

KFRB03301-I

Synchronization control process started. (C)

同期管理プロセスを起動しました。

(S)処理を続行します。

KFRB03302-I

Synchronization control process ended. (C)

同期管理プロセスを終了しました。

(S)処理を続行します。

KFRB03303-E

Exceeded limit time or transaction count of processing, detail = aa...aa. (C)

同期監視処理の限界時間, あるいは限界トランザクション数を超えました。詳細 = aa...aa

同期待ち限界時間, 又は同期待ち限界トランザクション数を超えました。

aa...aa : 限界を超えた要因

time : 同期待ち限界時間

transaction count : 同期待ち限界トランザクション数

(S)処理を中断します。

(O)次の対処をしてください。

aa...aa が time の場合 :

次のどちらかの対策をしてください。

- エラー情報ファイルなどを調査して、処理時間が長くなる原因を取り除いた後、反映処理を再起動してください。
- 反映システム定義の syncwait_limit_time オペランドの指定値を大きくした後、反映側 Datareplicator を再起動してください。

aa...aa が transaction count の場合：

反映システム定義の syncwait_limit_tran_count オペランドの指定値を大きくした後、反映側 Datareplicator を再起動してください。

これらの対処によっても改善されない場合には、反映側 Datareplicator の同期反映グループを解除 (hdsstart -c 同期反映グループ名) して再起動してください。

KFRB03304-E

Unable to specified dsid of Synchronization control process, group name = aa...aa, dsid = bb, command = cc...cc. (C)

同期反映グループを構成するデータ連動識別子は指定できません。同期反映グループ名 = aa...aa, データ連動識別子 = bb, コマンド = cc...cc

同期反映グループを構成するデータ連動識別子に対する操作はできません。

aa...aa：同期反映グループ名

bb：データ連動識別子

cc...cc：コマンド名

(S)処理を続行します。ただし、コマンドは受け付けられません。

(O)同期反映グループ名を指定して、再度コマンドを実行してください。

KFRB03311-E

Unable to continue Reflect process due to abnormal termination of Synchronized control process. (C)

同期管理プロセスの停止を検知したため反映プロセスを続行できません。

同期管理プロセスの停止を検知しました。同期反映グループを構成するすべての反映処理を停止します。

(S)処理を中断します。

(O)このメッセージの直前に出力されたメッセージからエラー原因を取り除き、hdsstart コマンド又は hdsrftcl コマンドで同期反映グループを再起動してください。

KFRB03312-E

Detected illegal extract data. (C)

不正な更新情報を検知しました。

反映トランザクション同期機能で使用できない更新情報を検知しました。抽出側 HiRDB の pd_rpl_reflect_mode オペランドの値が変更された可能性があります。

(S)処理を中断します。

(O)hdsstop コマンドで反映側 Datareplicator を停止した後、同期反映グループを解除 (hdsstart -c 同期反映グループ名) して再起動してください。

KFRB03314-W

Executed purge table, table name = aa...a. (C)

Purge Table を実行しました。table name = aa...a

Purge table の実行によって反映トランザクションの一部がコミットされました。これ以降、同期イベントを検知するまで、反映トランザクションの同期は保証されません。

aa...a : Purge table を実行した抽出表名

表名の後ろが 30 バイトまで空白になった名称が出力されます。

(S)処理を続行します。

(O)これ以降、同期イベントを検知するまで、反映トランザクションの同期は保証されません。抽出側システムで、直ちに同期イベントを実行してください。

KFRB03315-E

UOC cannot be executed. (C)

UOC は実行する事ができません。

反映トランザクション同期機能では、反映情報編集 UOC は使用できません。

(S)処理を中断します。

(O)反映定義の load 文から by 'uoc 名'の指定を削除してください。

KFRB03316-W

Detected illegal event code, event code = aa...a. (C)

不正なイベントコードを検知しました。イベントコード = aa...a

反映トランザクション同期機能では、eventsync で指定したイベントコード以外は扱えません。

aa...a : 検知したイベントコード

(S)該当するイベントコードを無視して、処理を続行します。

KFRB03317-E

```
Detected illegal extract definition.      (C)  
不正な抽出定義を検知しました。
```

反映トランザクション同期機能使用中は、抽出定義の変更はできません。

(S)処理を中断します。

(O)hdsstop コマンドで反映側 Datareplicator を停止した後、同期反映グループを解除 (hdsstart -c 同期反映グループ名) して再起動してください。

KFRB04001-E

```
File access error was occurred, operation = nnnnn, file = aa...aa, errno = xx...xx.  (S)
```

ファイル操作でエラーを検知しました。

nnnnn : 操作種別 (システムコール名)

aa...aa : ファイルの絶対パス名

ただし、hdsenv.ini ファイルについては、絶対パスで出力されません。

ファイルの場所は、Datareplicator のインストールディレクトリ下です。

xx...xx : エラー番号

エラー番号が 0 の場合は、次のことを示します。

操作種別が open : 初期化されていないか、初期化処理がエラー

操作種別が read : ファイルの内容不正

操作種別が write : ディスク容量不足

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

ファイル名に hdsenv.ini が出力されている場合は、hdsenv.ini ファイルにアクセスできない状態か、ファイル内容が不正です。Datareplicator を再インストールしてください。

KFRB04002-E

```
Necessary file not found, file = aa...aa.  (S)
```

必要なファイルがありません。

aa...aa：ファイル名

(S)処理を中止します。

(O)実行したコマンド名, 及び aa...aa に出力されているファイル名を基に, 次に示す表を参照して対策してください。

コマンド名	ファイル名	意味	対策
hdestart_n	抽出システム定義ファイル名	抽出システム定義ファイルが見付かりません。原因として, 次に示す可能性が考えられます。 <ol style="list-style-type: none"> 1. 環境変数 HDEPATH に間違っ た運用ディレクトリを指定している。 2. システムマネージャがないノードで抽出マスタプロセスを起動しようとした (-b オプション指定を省略して実行した)。 	<ul style="list-style-type: none"> • 原因が 1 の場合 環境変数 HDEPATH に, 正しい運用ディレクトリを指定してください。 • 原因が 2 の場合 抽出マスタプロセスを起動するために, システムマネージャがあるノードでコマンドを実行してください。
hdestop_n	errfile_1 サーバ名	抽出ノードマスタエラー情報ファイルが見付かりません。原因として, 次に示す可能性が考えられます。 <ol style="list-style-type: none"> 1. 環境変数 HDEPATH に, 間違っ た運用ディレクトリを指定している。 2. 抽出システム定義の nodecontrol オペランドに指定されている値が unit である。 3. -b オプションに指定したサーバが, コマンドを実行したノード上に存在しない。又は-b オプションに不正なサーバ名を指定した。 4. 抽出ノードマスタエラー情報ファイルが不当に削除された。 	<ul style="list-style-type: none"> • 原因が 1 の場合 環境変数 HDEPATH に, 正しい運用ディレクトリを指定してください。 • 原因が 2 の場合 抽出側 Datareplicator を停止する場合, hdestop コマンドを使用してください。hdestop_n コマンドを使用する場合の前提条件などについては, 「7. コマンドの文法」を参照してください。 • 原因が 3 の場合 -b オプションに指定したサーバが存在するノード上でコマンドを実行しているかどうかを見直してください。 • 原因が 4 の場合 hdestop_n コマンドは実行できません。抽出側 Datareplicator を停止する場合, hdestop コマンドを使用してください。
	抽出システム定義ファイル名	抽出システム定義ファイルが見付かりません。原因として, 次に示す可能性が考えられます。 <ol style="list-style-type: none"> 1. 環境変数 HDEPATH に, 間違っ た運用ディレクトリを指定している。 2. システムマネージャがないノードで抽出マスタプロセスを停止しよう 	<ul style="list-style-type: none"> • 原因が 1 の場合 環境変数 HDEPATH に, 正しい運用ディレクトリを指定してください。 • 原因が 2 の場合 システムマネージャがあるノードでコマンドを実行してください。

コマンド名	ファイル名	意味	対策
		とした (-b オプション指定を省略して実行した)。	
その他	-	コマンド実行に必要なファイルが存在しません。	エラー原因を取り除いた後、再実行してください。

KFRB04003-E

Invalid file size, file = nn...nn, size = aa...aa. (S)

ファイルのサイズが不正です。

nn...nn：ファイル名

aa...aa：ファイルのサイズ

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04004-E

Invalid file type, file = aa...aa. (S)

ファイルの種別が不正です。

aa...aa：ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04005-E

Invalid file data, file = aa...aa. (S)

ファイルの内容が不正です。

aa...aa：ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04006-E

Unable to use both UNIX regular file and Character special file. (S)

UNIX の通常ファイルとキャラクタ型スペシャルファイルを混在して使えません。

(S)処理を中断します。

(O)「4.6.2 抽出側 Datareplicator で使うファイルの準備」、又は「4.7.2 反映側 Datareplicator で使うファイルの準備」を参照して、ファイル種別の混在を修正した後、再実行してください。

KFRB04007-E

Necessary directory not found, directory = aa...aa. (S+L)

必要なディレクトリが存在しません。

aa...aa：ディレクトリ名

(S)処理を中止します。

(O)実行したコマンド名、及び aa...aa に出力されているディレクトリ名を基に、次に示す表を参照して対策してください。

コマンド名	ディレクトリ名	意味	対策
hdestart_n	環境変数 HDEPATH に設定されたディレクトリ名	抽出側 Datareplicator の運用ディレクトリが見付かりません。	環境変数 HDEPATH に、正しい運用ディレクトリを指定してください。
hdestop_n			

KFRB04011-E

Semaphore operation failure, sem-id = aa...aa, kind = nn...nn, errno = xx...xx. (S)

排他処理の操作に失敗しました。

aa...aa：セマフォ ID

nn...nn：排他種別

xx...xx：errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04012-E

Unable to allocate semaphore, sem-key = aa...aa, size = bb...bb, errno = cc...cc. (S)

セマフォの割り当てに失敗しました。

aa...aa：セマフォキー

bb...bb : セマフォ割り当てサイズ

cc...cc : errno にセットされたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04021-E

Signal operation failure, process-id = aa...aa, kind = bb...bb, errno = xx...xx. (S)

シグナルの操作に失敗しました。

aa...aa : 相手プロセスの ID

bb...bb : 送信しようとしたシグナル

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

errno=1 の場合、相手プロセスを起動したユーザと同じユーザで再実行してください。

KFRB04031-E

Cannot exec child process, process name = aa...aa, errno = xx...xx. (C)

子プロセスを実行できません。

aa...aa : 子プロセス名

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04032-E

Cannot fork child process, errno = xx...xx. (S)

子プロセスを生成できません。

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04033-E

```
Timeout occurred in Source site Datareplicator start, waiting time = aa...aasecond, process name = bb...bb. (S)
```

抽出側 Datareplicator の起動が aa…aa 秒以内に完了しませんでした。

aa...aa：起動完了待ち時間

bb...bb：起動対象のプロセス名

(S)処理を中止します。

(O)保守員に連絡し、hdestart コマンドを実行したディレクトリ下に出力されているコアダンプを取得してください。

KFRB04034-E

```
Found abnormal termination process, pid = aa...aa, process name = bb...bb. (S)
```

プロセスの異常終了を検知しました。

aa...aa：異常終了した子プロセスのプロセス ID

bb...bb：起動対象のプロセス名

(S)処理を中止します。

(O)syslogfile, 又は msterrfile1, msterrfile2 に出力されているメッセージを参照し、エラー原因を取り除いて hdestart コマンドを再実行してください。メッセージが出力されていない場合は、保守員に連絡してください。

KFRB04041-E

```
Shared memory operation failure, type = mm...mm, shm-id = aa...aa, kind = nn...nn, errno = xx...xx. (S)
```

共用メモリの操作に失敗しました。

mm...mm：共用メモリ種別

aa...aa：共用メモリ ID

nn...nn：操作種別

xx...xx：errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04042-E

```
Unable to allocate shared memory, type = mm...mm, shm-key = aa...aa, size = bb...bb,  
errno = cc...cc. (S)
```

共用メモリの割り当てに失敗しました。

mm...mm : 共用メモリ種別

aa...aa : 共用メモリキー

bb...bb : 共用メモリ割り当てサイズ

cc...cc : errno にセットされたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04051-E

```
Insufficient memory, type = mm...mm, size = nn...nn, errno = xx...xx. (S)
```

メモリが不足しています。

mm...mm : 詳細コード

nn...nn : メモリ割り当てサイズ

xx...xx : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04061-E

```
Necessary environment variable value cannot get, variable name = aa...aa. (S)
```

操作に必要な環境変数が設定されていません。

aa...aa : 環境変数名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04062-E

```
Environment variable value invalid, variable name = aa...aa. (S)
```

環境変数に指定した値が不正です。

aa...aa : 環境変数名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04071-E

```
Standard input not terminal. (S)
```

標準入力端末ではありません。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04101-E

```
Invalid command argument. (S)
```

コマンドの引数の指定に誤りがあります。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04102-E

```
Invalid option specified, option name = aa...aa. (S)
```

指定できないコマンドオプションを検知しました。

aa...aa : コマンドオプション

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04103-E

```
Necessary option not specified, option name = aa...aa. (S)
```

必要なコマンドオプションが指定されていません。

aa...aa : コマンドオプション

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04104-E

Option value is invalid, option name = aa...aa. (S)

コマンドオプションに指定された値が不正です。

aa...aa : コマンドオプション

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04105-E

Options can not specify simultaneously, option name = aa...aa and bb...bb. (S)

コマンドオプションとして、同時に指定できないものがあります。

aa...aa 及び bb...bb : コマンドオプション

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04201-E

Invalid User-id found. (S)

ユーザ ID が不正です。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04202-E

Invalid Password found. (S)

パスワードが不正です。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04203-E

Invalid Time range in definition file, definition file = aa...aa. (S)

定義ファイルで指定した時間が不正です。

aa...aa：反映システム定義ファイル名 又は反映環境定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04204-E

Invalid HiRDB Datareplicator-id found in definition file, definition file = aa...aa. (S)

定義ファイルで指定した反映側 Datareplicator 識別子が不正です。

aa...aa：反映システム定義ファイル名 又は反映環境定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04205-E

Invalid replication node-id found in definition file, definition file = aa...aa. (S)

定義ファイルで指定したデータ連動識別子が不正です。

aa...aa：反映システム定義ファイル名 又は反映環境定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04206-E

Invalid service name found in definition file, definition file = aa...aa. (S)

定義ファイルで指定したサービス名が不正です。

aa...aa：反映システム定義ファイル名 又は反映環境定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04207-E

Invalid file name found in definition file, file name = aa...aa, definition file = bb...bb. (S)

定義ファイルで指定したファイル名が不正です。

aa...aa : 指定したファイル名

bb...bb : 反映システム定義ファイル名 又は反映環境定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04208-E

Invalid T-SELECTOR found in definition file, definition file = aa...aa. (S)

定義ファイルで指定した T セレクタが不正です。

aa...aa : 反映システム定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04211-E

Sector size is larger than queuesize, sector size = aa...aa, queuesize = bb...bb. (S)

反映情報キューファイルサイズよりも、セクタ長の方が大きくなっています。

aa...aa : セクタ長

bb...bb : キューファイルサイズ

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04212-E

Sector size is larger than statssize, sector size = aa...aa, statssize = bb...bb. (S)

反映ステータスファイルのサイズよりも、セクタ長の方が大きいです。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04213-E

Command option value is not equal to dsid number of incomplete initialization, previous value = aa...aa, current value = bb...bb. (S)

指定したデータ連動識別子番号は、未完了となった部分初期化で指定されていたデータ連動識別子番号と異なります。

aa...aa : 前回指定したデータ連動識別子番号

bb...bb : 今回指定したデータ連動識別子番号

(S)処理を中止します。

(O)前回指定したデータ連動識別子番号を指定して部分初期化を再実行するか、反映側 Datareplicator 全体を初期化してください。

KFRB04214-E

The initializing which designated dsid was incomplete, dsid no= aa...aa. (S)

部分初期化が完了していません。

aa...aa : 前回の部分初期化で指定したデータ連動識別子番号

(S)処理を中止します。

(O)前回指定したデータ連動識別子番号を指定して部分初期化を再実行するか、反映側 Datareplicator 全体を初期化してください。

KFRB04215-E

Target site Datareplicator is not initialized. (S)

反映側 Datareplicator が初期化されていません。

(S)処理を中止します。

(O)反映側 Datareplicator 全体を初期化してください。

KFRB04216-I

Target site Datareplicator initialization completed. (C)

反映側 Datareplicator の初期化が完了しました。

(S)システムを終了します。

KFRB04217-E

Unable to initialize target site Datareplicator, because there are data to be reflected. (S)

反映環境を初期化できません。

(S)処理を中断します。

(O)反映側 Datareplicator の状況に応じて対処してください。

反映側 Datareplicator の状況	対処方法
<ul style="list-style-type: none">未反映の更新情報が残っており、反映が必要な場合	未反映の更新情報をすべて反映した後、再度初期化してください。
<ul style="list-style-type: none">未反映の更新情報が残っているが、反映が必要ない場合すべての更新情報を反映したが、hdsstop -t force コマンドで Datareplicator を強制終了した場合反映情報ステータスファイル、又は反映情報キューファイルをキャラクタ型スペシャルファイルで作成し、初めて反映側 Datareplicator を初期化する場合反映環境定義の statsfile オペランドに指定したステータスファイル名を変更する場合に、変更するステータスファイル名と同じ名前のファイルが既に存在しているとき	hdsstart -i -f コマンドで、再度初期化してください。

KFRB04218-I

Command request was ignored because process is running, process = aa...aa, pid = bb...bb. (S+L)

起動しようとしたプロセスは既に稼働中のため、起動要求を無視します。

aa...aa：プロセス種別

hdemaster：抽出マスタプロセス

hdenodemst：抽出ノードマスタプロセス

bb...bb：プロセス ID

(S)処理を中止します。

KFRB04219-E

Command request was ignored because process is not running, process = aa...aa. (S+L)

停止しようとしたプロセスは稼働中ではないため、停止要求を無視します。

aa...aa：プロセス種別

hdemaster：抽出マスタプロセス

hdenodemst：抽出ノードマスタプロセス

注

次に示すすべての条件が重なる環境下で `hdestop_n` コマンドを実行した場合、抽出ノードマスタープロセスが稼働中であってもコマンドの実行結果としてこのメッセージが出力されます。

- 抽出システム定義の `nodecontrol` オペランドの指定値が `unit` である。
- 運用ディレクトリ下に `errfile1_サーバ名` というファイルが存在する。
- `hdestop_n` コマンドを `-b` オプション指定で実行する。

この場合、抽出側 `Datareplicator` を停止するためには `hdestop` コマンドを使用してください。

`hdestop_n` コマンドを使用するための前提条件などについては、「[7. コマンドの文法](#)」を参照してください。

(S)処理を中止します。

KFRB04220-E

```
Unable to execute aa...aa command because of definition, info = "bb...bb". (S+L)
```

コマンドを実行するための前提条件を `Datareplicator` の定義が満たしていないため、コマンドの実行ができません。

`aa...aa` : コマンド名

`bb...bb` : オペランド情報

(S)処理を中止します。

(O)コマンド名及びオペランド情報を基に、次に示す表を参照して対策してください。

コマンド名	ディレクトリ名	意味	対策
<code>hdestop_n</code>	<code>nodecontrol=unit</code>	抽出システム定義の <code>nodecontrol</code> オペランドに指定されている値が <code>unit</code> であるため、コマンドを実行できません。	抽出側 <code>Datareplicator</code> を停止する場合、 <code>hdestop</code> コマンドを使用してください。 <code>hdestop_n</code> コマンドを使用するための前提条件などについては、「 7. コマンドの文法 」を参照してください。

KFRB04221-E

```
Command request acceptance error was occurred, command = aa...aa, reason = bb...bb.  
(S)
```

コマンド要求の受け付けエラーが発生しました。

`aa...aa` : コマンド名

`bb...bb` : 理由コード

(S)処理を中止します。

(O)コマンド名及び理由コードを基に、次に示す表を参照して対策してください。

コマンド名	理由コード	意味	対策
hdsrfctl	1	コマンドの引数が不正です。	コマンドの引数を見直してください。反映エラー情報ファイル、syslog ファイル、又は標準エラー出力に出力されるメッセージに従って対処してください。
	11	反映処理の起動オプションを指定してこのコマンドを実行しましたが、反映処理は既に起動済みです。	反映処理は既に起動済みであるため、このメッセージに対する対策は不要です。
	12	反映処理の起動オプションを指定してこのコマンドを実行しましたが、反映マスタプロセスからの応答がありません。	反映エラー情報ファイルを参照して、反映側 Datareplicator が正しく動作しているかどうか確認してください。 エラーメッセージが出力されていない場合、反映側 Datareplicator が応答できない状態となっている可能性があります。リソース削除コマンドを実行した後、反映側 Datareplicator を再起動してください。
hdechgstatus	11	一部のノード又はサーバの状態更新に失敗しました。	次の対策をしてください。 1. 抽出マスタエラー情報ファイル中の KFRB00724-W メッセージを確認して、更新に失敗したホスト名を特定してください。 2. KFRB00724-W メッセージの直前にエラーメッセージが出力されている場合は、エラーメッセージに従って対処してください。 3. エラーメッセージが出力されていない場合は、抽出ノードマスタエラー情報ファイルに出力されている KFRB00725-W メッセージを確認して、更新に失敗したサーバ名を特定してください。KFRB00725-W メッセージの直前に出力されているエラーメッセージに従って対処してください。

KFRB04301-E

Target site Datareplicator is already started. (S)

反映側 Datareplicator は既に起動されています。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04302-E

Target site Datareplicator is not started. (S)

反映側 Datareplicator は起動されていません。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04303-E

Same type command was already accepted. This command was ignored, accepted command = aa...aa. (S)

既に同系統のコマンドが実行されています。このコマンドは無効にします。

aa...aa : 受け付け済みのコマンド

(S)処理を中止します。

KFRB04304-E

After HiRDB Datareplicator initialized, operand changed in definition file, operand = aa...aa, definition file = bb...bb. (S)

Datareplicator を初期化した後に、定義ファイル内のオペランドが変更されました。

aa...aa : オペランド名

オペランド名が refenv001~refenv128 の場合、そのオペランドで指定した反映処理環境ファイルの中の statsfile オペランドが変更された可能性もあります。

bb...bb : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04305-E

Update queue file must be more than two in definition file, definition file = aa...aa. (S)

反映情報キューファイルは 2 個以上必要です。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04306-E

Replication node-id not specified in definition file, definition file = aa...aa. (S)

データ連動識別子が指定されてません。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04307-E

Protocol not specified in definition file, definition file = aa...aa. (S)

プロトコルが指定されてません。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04308-E

Unreflected data file must be two in definition file, definition file = aa...aa. (S)

未反映情報ファイルは2個必要です。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04309-E

Reflect status file not specified in definition file, definition file = aa...aa. (S)

反映ステータスファイルが指定されてません。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04310-E

Reflect environment file not specified for replication node-id in definition file, definition file = aa...aa. (S)

データ連動識別子に対応する反映環境定義ファイルが反映システム定義ファイルに指定されていません。

aa...aa：反映環境定義ファイルの絶対パス名

(S)処理を中止します。

(O)反映システム定義ファイルの refenvxxx に、反映定義ファイル名を指定してください。反映システム定義ファイルを修正後、再実行してください。

KFRB04311-E

File size not specified in definition file, file kind = aa...aa, definition file = bb...bb (S)

ファイルサイズが指定されてません。

aa...aa：ファイル種別

bb...bb：定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04312-E

Shared memory size not specified in definition file, definition file = aa...aa. (S)

定義情報格納用の共用メモリサイズが指定されてません。

aa...aa：定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04313-E

Cannot accept command while processing command, rejected command = aa...aa. (S)

コマンド処理中のためコマンドは受け付けられません。

aa...aa：コマンド名

(S)処理を中止します。

(O)次のどちらかの可能性があります。

- 他コマンド実行中に該当コマンドを実行しています。
先に実行したコマンドが終了した後、該当コマンドを再度実行してください。
- 他コマンドが強制停止された後、該当コマンドを実行しています。

Unix 版抽出側 Datareplicator の場合は `hdeshmclean -l clean` コマンド、Unix 版反映側 Datareplicator の場合は `hdsshmclean -l clean` コマンドを実行し、残った共用リソースを削除した後、Datareplicator を再起動してください。

Windows 版 Datareplicator の場合は、マシンを再起動してください。

KFRB04314-E

```
File name not unique in definition file, file name = aa...aa. (S)
```

定義ファイル中のファイル名称が一意ではありません。

aa...aa : ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04315-E

```
Replication node-id not unique in definition file, definition file = aa...aa. (S)
```

定義ファイル中のデータ連動識別子が一意ではありません。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04316-E

```
Event code not unique in definition file, definition file = aa...aa. (S)
```

定義ファイル中のイベントコードが一意ではありません。

aa...aa : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04317-E

T-SELECTOR not specified in definition file, definition file = aa...aa. (S)

T セレクタを指定していません。

aa...aa : 反映システム定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04318-E

Invalid SQLCODE found in skip_sqlcode, definition file = nn...nn. (S)
skip_sqlcode に不正な SQLCODE が指定されています。

nn...nn : 定義ファイル

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04319-E

oracleusr not specified in definition file, definition file = aa...aa. (S)

反映システム定義ファイルに oracleusr を指定していません。

aa...aa : 反映システム定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04320-E

odbcusr not specified in definition file, definition file = aa...aa. (S)

反映システム定義ファイルに odbcusr を指定していません。

aa...aa : 反映システム定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04321-E

datasource not specified in definition file, definition file = aa...aa. (S)

反映システム定義ファイルに datasource を指定していません。

aa...aa : 反映システム定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04322-E

Duplicate file kind appointed in definition file, operand = aa...aa, definition file = bb...bb. (S)

Datereplicator ファイルシステム領域への割り当てファイル種別を重複して指定しています。

aa...aa : オペランド名

bb...bb : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04323-E

Invalid file kind appointed in definition file, file kind = aa...aa, definition file = bb...bb. (S)

Datereplicator ファイルシステム領域への割り当てに指定したファイル種別が不正です。

aa...aa : 割り当てファイルの種別

bb...bb : 反映環境定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04324-E

Unable to use file system, file system = aa...aa, error code = bb...bb. (S)

Datereplicator ファイルシステム領域が使いません。

aa...aa : Datereplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中止します。

(O)Datareplicator ファイルシステム領域に指定したファイルが初期化済みかどうかを確認した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB04325-E

```
Failed to initialize file system, file system = aa...aa, error code = bb...bb. (S)
```

Datareplicator ファイルシステム領域の初期化に失敗しました。

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中止します。

(O)Datareplicator ファイルシステム領域に指定したファイルを `hdsfmkfs` コマンドで初期化した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB04326-E

```
Failed to prepare file system, file system = aa...aa, error code = bb...bb. (S)
```

Datareplicator ファイルシステム領域の準備に失敗しました。

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中止します。

(O)Datareplicator ファイルシステム領域に指定したファイルが使えるかどうかを確認した後、再実行してください。

Datareplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB04327-E

```
Unable to add file system, file system = aa...aa, add file = bb...bb, add size = cc...cc, error code = dd...dd. (S)
```


Datereplicator ファイルシステム領域にファイルを追加できません。

aa...aa : Datereplicator ファイルシステム領域名

bb...bb : 追加ファイル名

cc...cc : 追加ファイルサイズ

dd...dd : Datereplicator ファイルシステム領域のアクセスエラー詳細コード

(S)処理を中止します。

(O)Datereplicator ファイルシステム領域に格納するファイルのサイズを確認した後、再実行してください。

Datereplicator ファイルシステム領域のアクセスエラー詳細コードについては、[KFRB00714-E](#)の説明を参照してください。

KFRB04328-E

HiRDB don't support 2 phase commitment. (S)

反映システム定義で `commit_method` に `fxa_sqle` 又は `fxa_all` を指定していますが、反映先 HiRDB は 2 相コミットをサポートしていません。環境変数の `PATH`, `SHLIB_PATH` が正しく指定されていなかったり、メモリ不足でライブラリをローディングできなかつたりした可能性もあります。

(S)処理を中止します。

(O)HiRDB のバージョンと `commit_method` の指定、環境変数 `PATH`, `SHLIB_PATH` の指定を確認した後、再実行してください。

KFRB04329-E

Unable to share the file system with another dsid, file system = aa...aa, definition file = bb...bb. (S)

部分初期化をする場合は、ほかの `dsid` との間で Datereplicator ファイルシステム領域名を共有できません。

aa...aa : Datereplicator ファイルシステム領域名

bb...bb : 定義ファイル名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04330-E

Insufficient status file size. (S)

反映ステータスファイルサイズが不足しています。

(S)処理を中止します。

(O)見積もり式に従って反映環境定義の statssize オペランドの設定値を十分な値に変更した後、再実行してください。なお、Datareplicator 06-03 以降では最低サイズが 165KB なので注意してください。

KFRB04331-Q

Do you really want to clean up all resources (y or n)? (S)

Datareplicator の共用リソースを削除するかどうかを応答してください。

- 「y」を応答：Datareplicator の共用リソースを削除します。
- 「n」を応答：Datareplicator の共用リソースの削除をキャンセルして終了します。

(S)応答を待ちます。

(O)Datareplicator の共用リソースを削除する場合は「y」を入力してください。Datareplicator の共用リソースの削除をキャンセルする場合は「n」を入力してください。

KFRB04332-I

All resources were cleaned up successfully. (S)

Datareplicator の共用リソースを削除しました。

(S)処理を終了します。

KFRB04333-I

There are no resources. (S)

Datareplicator の共用リソースはありません。

(S)処理を終了します。

KFRB04334-E

There is not host-name or server-name resource. (S)

ホスト名またはサーバ名のリソースがありません

hdeshmclean コマンドの-x オプションに指定したホスト名、又はサーバ名のリソースがありません。

(S)処理を中断します。

(O)hdeshmclean コマンドの-x オプションに指定したホスト名, 又はサーバ名を見直してください。

抽出システム定義の nodecontrol オペランドに server を指定している場合:

-x に, 削除対象とする共用リソースが存在するバックエンドサーバ名を指定してください。

抽出システム定義の errfile_unique オペランドに true を指定している場合:

-x に, 削除対象とする共用リソースが存在するサーバのホスト名を指定してください。

KFRB04335-E

Invalid combination of startmode operand in Synchronization reflection group. (S)

同期反映グループ間の startmode オペランドの組合せが不正です。

(S)処理を中断します。

(O)同期反映グループ内での startmode オペランドの組み合わせを以下のどちらかに修正した後, 再度 hdsstart を実行してください。

(a) tm, もしくは tbl だけの組み合わせとする。

(b) すべて spd で統一する。

KFRB04336-E

Dsid specified as Synchronization reflection group does not exist. (S)

同期反映グループに指定したデータ連動識別子は存在しません。

(S)処理を中断します。

(O)dsidxxx オペランドに定義したデータ連動識別子を指定してください。修正後, 初期化を実行してください。

KFRB04337-E

Invalid dsid specified as Synchronization reflection group. (S)

同期反映グループに指定したデータ連動識別子が不正です。

(S)処理を中断します。

(O)データ連動識別子を, 次の点に注意して見直してください。修正後, 初期化を実行してください。

- 1 文字, 又は 3~8 文字を指定している。
- 0~9, A~F 以外の文字を指定している。

KFRB04338-E

Synchronization reflection group name does not exist. (S)

指定した同期反映グループ名は存在しません。

(S)処理を中断します。

(O)同期反映グループ名を、次の点に注意して見直してください。修正後、初期化を実行してください。

- syncgroup001 オペランドで指定した名称と異なる。
- syncgroup001 オペランドの指定がない。

KFRB04339-E

```
Duplicate dsid specified as Synchronization reflection group. (S)  
同期反映グループに指定したデータ連動識別子が重複しています。
```

(S)処理を中断します。

(O)重複したデータ連動識別子を見直してください。修正後、初期化を実行してください。

KFRB04343-W

```
Synchronous reflection group canceled, group name = nn...n. (S)
```

同期反映グループはキャンセルされました。これ以降、該当する同期反映グループでの反映トランザクション同期機能は機能しません。

nn...n：同期反映グループ名

(S)処理を続行します。

KFRB04344-E

```
Invalid Synchronization reflection group name. (S)
```

同期反映グループが不正です。

(S)処理を中断します。

(O)同期反映グループを修正してください。

KFRB04345-E

```
Not specified dsid as Synchronization reflection group. (S)
```

同期反映グループのデータ連動識別子番号は部分初期化できません。

(S)処理を中断します。

(O)同期反映グループ以外のデータ連動識別子番号を指定してください。

KFRB04400-E

HiRDB access error occurred, SQL kind = aa...aa, SQLCODE = nn...nn. (S)

HiRDB へのアクセスエラーが発生しました。

aa...aa : SQL のアクセス種別

nn...nn : SQLCODE

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

SQLCODE については、マニュアル「HiRDB メッセージ」を参照してください。なお、SQLCODE が-449 の場合、抽出対象 HiRDB 上に複数のイベント制御用テーブル (hde_dtbl) が存在している可能性があります。イベント制御用テーブルが抽出対象 HiRDB 上に一つだけ存在するように、残りのイベント制御用テーブルを削除してください。

KFRB04401-E

Invalid send system identifier, identifier = aa...aa. (S)

送信先識別子が不正です。

aa...aa : 送信先識別子

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04402-E

Invalid server name, server name = aa...aa. (S)

サーバ名が不正です。

aa...aa : サーバ名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04410-E

Source site Datareplicator is already started. (S)

抽出側 Datareplicator は既に起動されています。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04411-E

Source site Datareplicator is not started. (S)

抽出側 Datareplicator は起動されていません。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04412-I

Source site Datareplicator initialization is interrupted. (S)

抽出側 Datareplicator の初期化処理を中断します。

(S)処理を終了します。

KFRB04413-Q

Please confirm HiRDB is not running in replication mode and there is no need to use queue file recovery function, do you really want to continue initialization (y or n)? (S)

初期化を実行して問題ないか、次のことを確認してから初期化処理を続行するかどうかを応答してください。

- 抽出側 HiRDB が HiRDB Datareplicator 連携を使っているか
- 抽出情報キューファイル回復機能を使用する必要があるか

「y」を応答：初期化処理を実行します。

「n」を応答：初期化処理をキャンセルして終了します。

抽出側 HiRDB が HiRDB Datareplicator 連携を使っている場合：

「y」を応答すると、抽出側 DB と反映側 DB との間に不整合が発生することがあります。不整合が発生した場合は、抽出側と反映側の両方のデータ連動の環境の初期化、及び反映側 DB の再作成が必要です。

抽出情報キューファイル回復機能を使用する必要がある場合：

「n」を応答してください。「y」を応答すると、抽出情報キューファイル回復機能を使用できなくなります。

(S)応答を待ちます。

(O)初期化処理を実行する場合は「y」を入力してください。初期化処理をキャンセルする場合は「n」を入力してください。

KFRB04414-Q

Extracted data may be lost by initialization, do you really want to continue initialization (y or n) ? (S)

初期化を実行して問題ないかどうかを確認してから、初期化処理を続行するかどうかを応答してください。

- 「y」を応答：初期化処理を実行します。
- 「n」を応答：初期化処理をキャンセルして終了します。

(S)応答を待ちます。

(O)初期化処理を実行する場合は「y」を入力してください。初期化処理をキャンセルする場合は「n」を入力してください。

KFRB04415-E

SQL output processing can not start while Source site Datareplicator is running. (S + E)

抽出側 Datareplicator が起動しているため、更新 SQL 出力処理を開始できません。

(S)処理を中断します。

(O)抽出側 Datareplicator を停止した後、コマンドを再実行してください。

KFRB04416-E

Extraction processing or transmission processing can not start while SQL output processing is working. (S + E)

更新 SQL 出力処理が起動しているため、抽出処理又は送信処理を開始できません。

(S)処理を中断します。

(O)抽出側 Datareplicator を停止した後、コマンドを再実行してください。

KFRB04500-I

hdeprep command terminated normally[, number of extraction tables = aa...aa]. (S)

hdeprep コマンドは正常に終了しました。抽出定義ファイルに extract_all を指定している場合だけ、抽出対象表の数として「number of extraction tables = aa...aa」を出力します。

aa...aa：抽出対象表の数

(S)処理を終了します。

KFRB04501-E

Specified <table name> in <extract clause> in <extract statement> is not defined in HiRDB system, lineno = aa...aa, table name = bb...bb. cc...cc. (S)

extract 文の extract 句に指定された表名は HiRDB システム内で定義されていません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：認可識別子

cc...cc：表識別子

(S)処理を中止します。

(O)認可識別子.表識別子の指定を修正した後、再実行してください。

KFRB04502-E

Specified <column name> in <extract clause> in <extract statement> is not defined in HiRDB system, lineno = aa...aa, column name = bb...bb. (S)

extract 文の extract 句に指定された列名は HiRDB システム内で定義されていません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：列名

(S)処理を中止します。

(O)列名を修正又は削除後、再実行してください。

KFRB04503-E

Unable to specified <where clause> for forward matching log data name, lineno = aa...aa. (S)

前方一致の更新情報名に対して where 句は指定できません。

aa...aa：抽出定義ファイル内の行番号

(S)処理を中止します。

(O)from 句の更新情報名を全名称で指定するか、又は where 句を削除した後で、再実行してください。

KFRB04504-E

Condition constant value is invalid, <constant value> = aa...aa, reason = bb...bb, lineno = cc...cc. (S)

送信行選択条件の比較定数値が不正です。

aa...aa：比較定数値

bb...bb：理由コード

length over：長さが上限を超えている

invalid character：数値以外を指定している

cc...cc：抽出定義ファイル内の行番号

(S)処理を中止します。

(O)定数値を修正した後で、再実行してください。

KFRB04505-E

```
Number of condition in <where clause> exceeds maximum value, lineno = aa...aa, log data  
name = bb...bb. (S)
```

where 句に指定した送信行選択条件の数か上限（256）を超えました。

aa...aa：抽出定義ファイル内の行番号

bb...bb：更新情報名

(S)処理を中止します。

(O)送信行選択条件を見直して修正した後で、再実行してください。

KFRB04506-E

```
Unable to specified <where clause> for one of log data names from same table, lineno =  
aa...aa, log data name = bb...bb. (S)
```

1つの表を複数の更新情報名で抽出対象としている場合、その更新情報名に対して where 句を指定できません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：更新情報名

(S)処理を中止します。

(O)送信行選択の対象とする更新情報名を見直して修正した後で、再実行してください。

KFRB04507-E

```
Condition column type is invalid, lineno = aa...aa, column name = bb...bb. (S)
```

送信行選択条件列の属性が不正です。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信行選択条件列名

(S)処理を中止します。

(O)送信行選択条件列の属性を見直して修正した後、再実行してください。

KFRB04508-E

```
Duplicate <send system identifier> in <extract control statement file>, send system id =  
aa...aa. (S)
```

抽出システム定義内の送信先識別子が重複しています。

aa...aa：送信先識別子

(S)処理を中止します。

(O)送信先識別子の重複を排除した後、再実行してください。

KFRB04509-E

```
Missing value for operand, operand name = aa...aa. (S)
```

オペランドの値の指定がありません。

aa...aa：オペランド名

(S)処理を中止します。

(O)必要なオペランドに値を指定した後、再実行してください。

KFRB04510-E

```
Invalid sendidxx operand in definition file, definition file = aa...aa. (S)
```

sendidxx オペランドが不正です。

aa...aa：抽出システム定義ファイル名

(S)処理を中止します。

(O)抽出システム定義ファイルのオペランドの指定値を修正した後、再実行してください。

KFRB04511-E

<system table name> is specified in <extract clause> in <extract statement>, lineno = aa...aa. (S)

データディクショナリ表は抽出対象にできません。

aa...aa : 行番号

(S)処理を中止します。

(O)extract 文を削除した後、再実行してください。

KFRB04512-E

Number of <extract statement> exceeds value of extinfo num operand. (S)

extract 文の数が抽出システム定義ファイルの extinfo num オペランドで指定された値を超えました。

(S)処理を中止します。

(O)extract 文の数が extinfo num オペランドに指定した値以下になるように修正した後、再実行してください。

KFRB04513-E

Only mapping key column is specified in <extract clause> in <extract statement>, lineno = aa...aa. (S)

マッピングキーだけの抽出指定はできません。

aa...aa : 抽出定義ファイル内の行番号

(S)処理を中止します。

(O)マッピングキーを見直すか又は extract 文を削除した後、再実行してください。

KFRB04514-E

Specified user has no DBA privilege, user-id = aa...aa. (S)

指定されたユーザは DBA 権限を持っていません。

aa...aa : ユーザ ID

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04515-E

Number of <mapping key> exceeds maximum value, lineno = aa...aa. (S)

指定できるマッピングキーの数が上限 (64) を超えています。

aa...aa : 抽出定義ファイル内の行番号

(S)処理を中止します。

(O)マッピングキーを見直して修正した後、再実行してください。

KFRB04516-E

<event table name> is specified in <extract clause> in <extract statement>, lineno = aa...aa. (S)

イベント制御用テーブルは抽出対象にできません。

aa...aa : 抽出定義ファイル内の行番号

(S)処理を中止します。

(O)extract 文を削除した後、再実行してください。

KFRB04517-E

Invalid <log data name> in <to clause> in <extract statement>, lineno = aa...aa, log data name = bb...bb, reason = cc...cc. (S)

extract 文の to 句に指定した更新情報名が不正です。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 更新情報名

cc...cc : 理由コード

length over : 長さが 8 バイトを超えている。

(S)処理を中止します。

(O)更新情報名を修正した後、再実行してください。

KFRB04518-E

Invalid <log data name> in <from clause> in <send statement>, lineno = aa...aa, log data name = bb...bb, reason = cc...cc. (S)

send 文の from 句に指定した更新情報名が不正です。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 更新情報名

cc...cc : 理由コード

length over : 長さが 8 バイトを超えている。

(S)処理を中止します。

(O)更新情報名を修正した後, 再実行してください。

KFRB04519-E

```
Invalid <schema name> in <extract clause> in <extract statement>, lineno = aa...aa,  
schema name = bb...bb, reason = cc...cc. (S)
```

extract 文の extract 句に指定したスキーマ名が不正です。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : スキーマ名

cc...cc : 理由コード

length over : 長さが 30 バイトを超えている。

(S)処理を中止します。

(O)スキーマ名を修正した後, 再実行してください。

KFRB04520-E

```
Invalid <table name> in <extract clause> in <extract statement>, lineno = aa...aa, table  
name = bb...bb, reason = cc...cc. (S)
```

extract 文の extract 句に指定した表名が不正です。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 表名

cc...cc : 理由コード

length over : 長さが 30 バイトを超えている。

(S)処理を中止します。

(O)表名を修正した後, 再実行してください。

KFRB04521-E

```
Invalid <column name> in <extract clause> in <extract statement>, lineno = aa...aa,  
column name = bb...bb, reason = cc...cc. (S)
```

extract 文の extract 句に指定した列名が不正です。

aa...aa：抽出定義ファイル内の行番号

bb...bb：列名

cc...cc：理由コード

length over：長さが 30 バイトを超えている。

(S)処理を中止します。

(O)列名を修正するか又は削除した後、再実行してください。

KFRB04522-E

```
Invalid <send system identifier> in <send clause> in <send statement>, lineno = aa...aa,  
send system identifier = bb...bb, reason = cc...cc. (S)
```

send 文の send 句に指定した送信先識別子が不正です。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信先識別子

cc...cc：理由コード

length over：長さが 8 バイトを超えている。

(S)処理を中止します。

(O)送信先識別子を修正した後、再実行してください。

KFRB04523-E

```
Specified table name is not base table name, lineno = aa...aa, table name = bb...bb. (S)
```

指定された表名は実表ではありません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：表名

(S)処理を中止します。

(O)表名及び列名を実表のものに修正した後、再実行してください。

KFRB04524-E

```
Syntax error in <extract definition file>, lineno = aa...aa. (S)
```

抽出定義ファイル内の文法エラーです。

aa...aa : 抽出定義ファイル内の行番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04525-E

```
Number of <extract statement> exceeds maximum value. (S)
```

extract 文の数が最大値を超えました。

(S)処理を中止します。

(O)extract 文を減らした後、再実行してください。

KFRB04526-E

```
Number of <column name> exceeds maximum value, lineno = aa...aa, table name =  
bb...bb. (S)
```

列名の指定数が最大値を超えました。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 表名

(S)処理を中止します。

(O)抽出対象列を見直して減らした後、再実行してください。

KFRB04527-E

```
Invalid <mapping key> column length in <key clause> in <extract statement>, lineno =  
aa...aa, column name = bb...bb. (S)
```

マッピングキーを構成する列の中に、マッピングキーの最大長(255バイト)を超えているものがあります。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 列名

(S)処理を中止します。

(O)該当する列をマッピングキーから除外した後、再実行してください。

KFRB04528-E

```
Duplicate <mapping key> column in <key clause> in <extract statement>, lineno = aa...aa, column name = bb...bb. (S)
```

extract 文の key 句に指定されたマッピングキー列名が重複しています。

aa...aa：抽出定義ファイル内の行番号

bb...bb：列名

(S)処理を中止します。

(O)重複を排除した後、再実行してください。

KFRB04529-E

```
<mapping key> column is not specified in <extract clause> in <extract statement>, lineno = aa...aa, column name = bb...bb. (S)
```

extract 文の extract 句にマッピングキー列が指定されていません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：列名

(S)処理を中止します。

(O)マッピングキー列（一意になる列）を定義した後、再実行してください。

KFRB04530-E

```
<send system identifier> in <send clause> in <send statement> does not define in <extract control statement file>, lineno = aa...aa, send system identifier = bb...bb. (S)
```

send 文の send 句に指定された送信先識別子が抽出システム定義内で定義されていません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信先識別子

(S)処理を中止します。

(O)送信先識別子を修正した後、再実行してください。

KFRB04531-E

Number of <send statement> exceeds maximum value. (S)

send 文の数が最大値を超えました。

(S)処理を中止します。

(O)send 文の数を減らした後、再実行してください。

KFRB04532-W

Event table is not defined. (S)

イベント制御用テーブルが定義されていません。

(S)処理を続行します。ただし、イベントは発行できません。

(O)イベントを使う場合は、イベント制御用テーブル作成後、hdeprep コマンドを再実行してください。

KFRB04533-E

Specified <log data name> is not defined in <extract statement>, lineno = aa...aa, log data name = bb...bb. (S)

指定された更新情報名は extract 文で定義されていません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：更新情報名

(S)処理を中止します。

(O)extract 文を追加するか、又は更新情報名を対応の取れた名称に修正した後、再実行してください。

KFRB04534-E

Duplicate <log data name> in <to clause> in <extract statement>, lineno = aa...aa, log data name = bb...bb. (S)

extract 文の to 句に指定された更新情報名が重複しています。

aa...aa：抽出定義ファイル内の行番号

bb...bb：更新情報名

(S)処理を中止します。

(O)重複している名称を削除するか又は変更した後、再実行してください。

KFRB04535-E

Event table must be a fix table. (S)

イベント制御用テーブルが FIX 属性の表として定義されていません。

(S)処理を中止します。

(O)イベント制御用テーブルを再作成 (DROP TABLE 及び CREATE TABLE の実行) 後, 再実行してください。

KFRB04536-E

Invalid column numbers of event table. (S)

イベント制御用テーブルの列数が正しくありません。

(S)処理を中止します。

(O)イベント制御用テーブルの構造を見直して, イベント制御用テーブルを再作成 (DROP TABLE 及び CREATE TABLE の実行) 後, 再実行してください。

KFRB04537-E

Invalid column of event table. (S)

イベント制御用テーブルの列属性 (データ型・長さ・名称) が正しくありません。

(S)処理を中止します。

(O)イベント制御用テーブルの構造を見直して, イベント制御用テーブルを再作成 (DROP TABLE 及び CREATE TABLE の実行) 後, 再実行してください。

KFRB04538-E

Unable to decide target system for <log data name>, log data name = aa...aa. (S)

更新情報名に対する送信先システムが特定できません。

aa...aa : 更新情報名

(S)処理を中止します。

(O)送信先システムとの対応を見直して修正した後, 再実行してください。

KFRB04539-E

Unsupported data type was found, lineno = aa...aa, column name = bb...bb. (S)

このデータ型は抽出できません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：列名

(S)処理を中止します。

(O)対象の列を抽出定義から削除した後、再実行してください。

KFRB04540-E

```
Extraction master process not exist. (S)
```

抽出マスタプロセスが存在しません。

(S)処理を中止します。

(O)抽出側 Datareplicator を起動した後、再実行してください。

KFRB04541-E

```
Condition column length is invalid, lineno = aa...aa, column name = bb...bb. (S)
```

送信行選択条件列の長さが不正です。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信行選択条件列名

(S)処理を中止します。

(O)送信行選択条件列の長さを見直して修正した後、再実行してください。

KFRB04542-E

```
Number of constant value in <in condition> exceeds maximum value, lineno = aa...aa.  
(S)
```

in 条件に指定した定数値の数が上限（16）を超えています。

(S)処理を中止します。

(O)定数値の数を見直して修正した後、再実行してください。

KFRB04543-E

```
Constant value length is invalid, lineno = aa...aa. (S)
```

送信行選択条件列の比較定数の長さが不正です。

aa...aa：抽出定義ファイル内の行番号

(S)処理を中止します。

(O)定数値の長さを見直して修正した後、再実行してください。

KFRB04544-E

```
Constant value type is invalid, lineno = aa...aa. (S)
```

送信行選択条件列の比較定数の属性が不正です。

aa...aa：抽出定義ファイル内の行番号

(S)処理を中止します。

(O)定数値の長さを見直して修正した後、再実行してください。

KFRB04545-E

```
Condition column is not mapping key, lineno = aa...aa, column name = bb...bb. (S)
```

送信行選択条件列がマッピングキーではありません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信行選択条件列名

(S)処理を中止します。

(O)送信行選択条件列を見直して修正した後、再実行してください。

KFRB04546-E

```
Not found condition column in log data, lineno = aa...aa, column name = bb...bb. (S)
```

更新情報内に送信行選択条件列がありません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信行選択条件列名

(S)処理を中止します。

(O)送信行選択条件列を見直して修正した後、再実行してください。

KFRB04547-E

```
Constant value is larger than condition column value range, lineno = aa...aa. (S)
```

送信行選択条件の比較定数値が、条件列の値範囲を超えています。

aa...aa：抽出定義ファイル内の行番号

(S)処理を中止します。

(O)定数値を見直して修正した後、再実行してください。

KFRB04548-E

```
Duplicate <send system identifier> / <log data name> in <send statement>, lineno = aa...aa, send system identifier = bb...bb, log data name = cc...cc. (S)
```

send 文に指定した送信先識別子／更新情報名の組み合わせが重複しています。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信先識別名

cc...cc：更新情報名

(S)処理を中止します。

(O)send 文の間での送信先識別子と更新情報名との組み合わせの重複を見直して修正した後、再実行してください。

KFRB04549-E

```
Unable to specified <flike condition> for numerical column, lineno = aa...aa, column name = bb...bb. (S)
```

数値属性列に対して flike 条件は指定できません。

aa...aa：抽出定義ファイル内の行番号

bb...bb：送信行選択条件列名

(S)処理を中止します。

(O)送信行選択条件列を見直して修正した後、再実行してください。

KFRB04550-E

```
Starting offset in <flike condition> is invalid, <column name> = aa...aa, reason = bb...bb, lineno = cc...cc. (S)
```

flike 条件の比較開始オフセット値が不正です。

aa...aa : 列名

bb...bb : エラー要因

double byte : NCHAR または NVARCHAR でオフセット値が奇数

not numeric : 数値以外を指定している

cc...cc : 抽出定義ファイル内の行番号

(S)処理を中止します。

(O)比較開始オフセットを見直して修正した後、再実行してください。

KFRB04551-E

```
Starting offset in <flike condition> is larger than condition column definition length, lineno = aa...aa. (S)
```

flike 条件の比較開始オフセット値が、条件列の定義長を超えています。

aa...aa : 抽出定義ファイル内の行番号

(S)処理を中止します。

(O)比較開始オフセットを見直して修正した後、再実行してください。

KFRB04552-E

```
Recovery type of BLOB column is not 'all', lineno = aa...aa, column name = bb...bb. (S)
```

BLOB 属性列の列回復制約が ALL ではありません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)抽出対象列の列回復制約を ALL に変更した後、再実行してください。

KFRB04553-E

```
BLOB column length is invalid, lineno = aa...aa, column name = bb...bb. (S)
```

BLOB 属性列の定義長が不正です。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)抽出対象列の定義長を見直して修正した後、再実行してください。

KFRB04554-E

Unable to specified BLOB column for mapping key, lineno = aa...aa, column name = bb...bb. (S)

マッピングキーに対して BLOB 属性列は指定できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)マッピングキー列を見直して修正した後、再実行してください。

KFRB04555-E

Unable to specified Abstract Data Type column for mapping key, lineno = aa...aa, column name = bb...bb. (S)

マッピングキーに対しては、抽象データ型列は指定できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)マッピングキー列を見直して修正した後、再実行してください。

KFRB04556-E

Number of <adt statement> exceeds maximum value. (S)

adt 文の数が最大値を超えました。

(S)処理を中止します。

(O)send 文の数を減らした後、再実行してください。

KFRB04557-E

```
Invalid <schema name> in <adt clause> in <adt statement>, lineno = aa...aa, schema  
name = bb...bb. (S)
```

adt 文の adt 句に指定したスキーマ名が間違っています。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : スキーマ名

(S)処理を中止します。

(O)スキーマ名を修正した後、再実行してください。

KFRB04558-E

```
Invalid <data type name> in <adt clause> in <adt statement>, lineno = aa...aa, data type  
name = bb...bb. (S)
```

adt 文の adt 句に指定したデータ型名が間違っています。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : データ型名

(S)処理を中止します。

(O)データ型名を修正した後、再実行してください。

KFRB04559-E

```
Invalid <UOC name> in <by clause> in <adt statement>, lineno = aa...aa, UOC name =  
bb...bb. (S)
```

adt 文の by 句に指定した UOC 名が不正です。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : UOC 名

(S)処理を中止します。

(O)UOC 名を修正した後、再実行してください。

KFRB04560-E

```
Invalid <UOC library name> in <lib clause> in <adt statement>, lineno = aa...aa, UOC  
library name = bb...bb. (S)
```


adt 文の lib 句に指定した UOC ライブラリ名が間違っています。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : UOC ライブラリ名

(S)処理を中止します。

(O)UOC ライブラリ名を修正した後、再実行してください。

KFRB04561-E

```
Not exist UOC library specified by <adt statement>, lineno = aa...aa, UOC library name
= bb...bb. (S)
```

adt 文に指定した UOC ライブラリが存在しません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : UOC ライブラリ名

(S)処理を中止します。

(O)UOC ライブラリを作成した後、再実行してください。

KFRB04562-E

```
Not exist UOC function into UOC library, lineno = aa...aa, UOC function name = bb...bb,
UOC library name = cc...cc. (S)
```

UOC ライブラリ中に UOC 関数が存在しません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : UOC 関数名

cc...cc : UOC ライブラリ名

(S)処理を中止します。

(O)UOC ライブラリを再作成した後、再実行してください。

KFRB04563-E

```
Specified data type is not nest unit Abstract data type of extract target, lineno = aa...aa,
schema name = bb...bb, data type name = cc...cc. (S)
```

指定したデータ型は、抽出対象のネスト単位の抽象データ型ではありません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : スキーマ名

cc...cc : データ型名

(S)処理を中止します。

(O)データ型名を修正した後、再実行してください。

KFRB04564-E

```
Duplicate <adt statement> for same data type, lineno = aa...aa, schema name = bb...bb,  
data type name = cc...cc. (S)
```

同じデータ型に対して、adt 文を重複して指定しています。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : スキーマ名

cc...cc : データ型名

(S)処理を中止します。

(O)adt 文の重複を修正した後、再実行してください。

KFRB04565-E

```
<adt statement> is not defined for this Abstract data type, schema name = aa...aa, data  
type name = bb...bb. (S)
```

この抽象データ型に対して、adt 文が定義されていません。

aa...aa : スキーマ名

bb...bb : データ型名

(S)処理を中止します。

(O)すべての抽象データ型に対して adt 文を定義した後、再実行してください。

KFRB04566-E

```
Unsupported Abstract data type was found, lineno = aa...aa, type name = bb...bb. (S)
```

この抽象データ型は、抽出できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽象データ型名

(S)処理を中止します。

(O)対象の抽象データ型を抽出定義から削除した後、再実行してください。

KFRB04567-E

```
Unable to specify <adt statement> for this Abstract data type, lineno = aa...aa, type name = bb...bb. (S)
```

この抽象データ型に対して、adt 文は指定できません。

aa...aa : スキーマ名

bb...bb : データ型名

(S)処理を中止します。

(O)不要な adt 文を削除した後、再実行してください。

KFRB04568-E

```
Unable to extract multi-value column, lineno = aa...aa, column name = bb...bb. (S)
```

繰返し列を抽出できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)繰返し列を抽出列から削除するか、又は抽出対象 HiRDB を 05-05 以降に変更した後、再実行してください。

KFRB04569-E

```
Unable to specify multi-value column for mapping key, lineno = aa...aa, column name = bb...bb. (S)
```

マッピングキーには繰返し列を指定できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)マッピングキー列を見直して修正した後、再実行してください。

KFRB04570-E

```
Unable to specify each data type for same column, lineno = aa...aa, table name = bb...bb,  
column name = cc...cc. (S)
```

同一列に対して異なるデータ型を指定できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象表名

cc...cc : 抽出対象列名

(S)処理を中止します。

(O)同一列に対する、異なる抽出データ型の指定を取り除いた後、再実行してください。

KFRB04571-E

```
Unable to specify data type of extract for this column in <extract statement>, lineno = aa...aa,  
table name = bb...bb, column name = cc...cc. (S)
```

当該抽出対象列には、抽出定義文で抽出データ型を指定できません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 抽出対象表名

cc...cc : 抽出対象列名

(S)処理を中止します。

(O)抽出データ型の指定を取り除く、または、抽出対象列を抽出データ型を指定可能なデータ型の列に変更した後、再実行してください。

KFRB04572-E

```
Unable to specified BINARY column for mapping key, lineno = aa...aa, column name =  
bb...bb. (S)
```

マッピングキーに対して BINARY 属性列を指定することはできません。

aa... aa : 抽出定義ファイル内行番号

bb...bb : 抽出対象列名

(S)処理を中止します。

(O)マッピングキーに対応する列に間違いがないか見直しを行い、間違いがある場合は修正してから再実行してください。

KFRB04573-E

```
Failed to get information of unique check of mapping key, SQLCODE = aa...aa, SQL = bb...bb. (S)
```

マッピングキー列のユニークチェックに使用する情報の取得に失敗しました。

aa... aa : SQLCODE

bb...bb : SQL 文

(S)処理を中止します。

(O)詳細情報を参照してエラーの原因を取り除いた後、コマンドを再実行してください。

KFRB04574-E

```
Mapping key column does not satisfy condition of unique check, table name = aa...aa, mapping key = bb...bb, code = cc...cc. (S)
```

マッピングキー列がユニークチェックの条件を満たしていません。

aa...aa : 抽出対象表名

bb...bb : マッピングキー構成列名

cc...cc : 要因コード

UNIQUE : 抽出対象表に、ユニークチェックの条件を満たすインデックスが定義されていません。

NOT NULL : ユニークチェックの条件を満たすインデックスの構成列に、NOT NULL 属性ではない列が含まれています。

(S)処理を中止します。

(O)マッピングキーに対応する列に間違いがないか見直しを行い、間違いがある場合は定義を修正してください。指定に間違いがない場合は、要因コードに従い処置してください。処置ができない場合は、抽出定義のチェック句の指定値を変更した後か、コマンド実行時の-k オプションの指定値を変更して、hdeprep コマンドを再実行してください。

- UNIQUE
抽出対象表に、ユニークチェックの条件を満たすインデックスを定義してください。
- NOT NULL
ユニークチェックの条件を満たすインデックスの構成列を、NOT NULL 属性にしてください。

KFRB04575-E

Extract data size exceeds maximum value, lineno = aa...aa, log data name = bb...bb. (S + L)

更新情報サイズが最大値を超えました。

aa...aa : 行番号

bb...bb : 更新情報名

(S)処理を中止します。

(O)更新情報長が最大値 (256MB) になるように抽出定義を修正した後、コマンドを再実行してください。

KFRB04579-E

Invalid extraction environment for extraction table that is specified without rollback option, lineno = aa...aa, reason = bb...bb. (S)

WITHOUT ROLLBACK オプションを指定した表を抽出対象とする場合の条件を満たしていません。

aa...aa : 抽出定義ファイル内の行番号

bb...bb : 理由コード

(S)処理を中止します。

[対策]理由コードごとに次のように対策してください。

理由コード	意味	対策
ukey	WITHOUT ROLLBACK オプションを指定している表では、マッピングキーに指定している列を更新できません。そのため、抽出定義文の ukey 句は指定できません。	抽出定義文の指定を key 句に変更して、再度実行してください。
sendcontrol	WITHOUT ROLLBACK オプションを指定している表を抽出対象とする場合は、抽出システム定義の sendcontrol オペランドに sendmst は指定できません。	抽出システム定義の sendcontrol オペランドの指定を nodemst (省略値) に変更して、hdestart -i コマンドで初期化してから再度実行してください。
divided table	WITHOUT ROLLBACK オプションを指定したサーバ間横分割表は、抽出対象にできません。	WITHOUT ROLLBACK オプションを外して表定義を行った後、再度実行してください。

KFRB04580-E

Without rollback option is unable to specify for event table. (S)

イベント制御用テーブル (hde_dtbl) には、WITHOUT ROLLBACK オプションを指定できません。

(S)処理を中止します。

[対策] イベント制御用テーブルの WITHOUT ROLLBACK オプションを外して表定義を行った後、再度実行してください。

KFRB04582-E

```
Number of extraction tables exceeds value of extinfo operand, extinfo value = aa...aa. (S)
```

抽出対象表の数が抽出システム定義の extinfo オペランドで指定された値を超えました。

aa...aa : extinfo オペランドの指定値

(S)処理を中止します。

(O)抽出システム定義ファイルの extinfo オペランドの指定値を抽出対象表の数以上に変更して、抽出側 Datareplicator を初期化後、再実行してください。又は、抽出定義の extract_all 文の except 句に連動対象外にする表を指定し、再実行してください。

KFRB04583-W

```
The extraction of table aa...aa.bb...bb was skipped, reason = cc...cc. (S)
```

抽出定義の extract_all 文では抽出できない表のため、抽出対象外とします。

aa...aa : 認可識別子

bb...bb : 表識別子

cc...cc : 理由コード

(S)処理を続行します。

(O)理由コードごとに次の表に従って対処した後、再実行してください

理由コード	意味	対策
table size over	表当たりの列定義長の合計が最大値 (256MB) を超えています。	列を選択して表を連動対象とする場合は、extract_all 文ではなく、extract 文で抽出対象の表を定義するよう抽出定義を修正してください。表を連動対象外にする場合、対処は不要です。
column type	抽出対象にできない列が定義されている表です。	列を選択して表を連動対象とする場合は、extract_all 文ではなく、extract 文で抽出対象の表を定義するよう抽出定義を修正してください。表を連動対象外にする場合、対処は不要です。

理由コード	意味	対策
mapping key	マッピングキーに使用できる列を構成列とするインデクスが定義されていません。	表を連動対象とする場合は、表に対して、マッピングキーに使用する列を構成列とするユニークインデクスを定義してください。定義するユニークインデクスは、ユニークチェックの条件を満たす必要があります。ユニークチェックの条件、及びマッピングキーに使用できる列の条件は、「 5.5 抽出定義 」を参照してください。表を連動対象外にする場合、対処は不要です。
schema name length over	スキーマ名の長さが 28 バイトを超えています。	表は連動対象にできません。対処は不要です。
table option	抽出対象にできないオプションを指定した表です。	表のオプションとその場合の前提条件、及び制限事項を確認し、必要に応じて表の定義などを変更してください。オプションと条件は、「 1.3 データ連動で使える表 」を参照してください。表を連動対象外にする場合、対処は不要です。

KFRB04584-E

Specified <table name> in <except clause> in <extract_all statement> is not defined in HiRDB system, lineno = aa...aa, table name = bb...bb.cc...cc. (S)

extract_all 文の except 句に指定された表名は HiRDB システム内の実表として定義されていません。

aa...aa：抽出定義ファイル内行番号

bb...bb：認可識別子

cc...cc：表識別子

(S)処理を中止します。

(O)except 句の認可識別子.表識別子の指定を修正した後、再実行してください。

KFRB04585-E

Number of <except clause> in <extract_all statement> exceeds maximum value, lineno = aa...aa. (S)

extract_all 文の except 句の指定数が最大値 (2048) を超えました。

aa...aa：抽出定義ファイル内行番号

(S)処理を中止します。

(O)except 句の指定内容を確認してください。抽出対象外にする表が最大数を超える場合は、extract_all 文ではなく、extract 文で抽出対象の表を指定するように抽出定義を修正してください。抽出定義を修正した後、再実行してください。

KFRB04586-E

```
There is no table to extract. (S)
```

extract_all 文を指定した場合に抽出対象にできる表がありません。

(S)処理を中止します。

(O)抽出側 DB に抽出対象にできる表があるか確認してください。KFRB04583-W が出力されている場合、KFRB04583-W の対処に従い、表を連動対象にできるようにした後、再実行してください。

KFRB04587-E

```
Cannot specify aa...aa in bb...bb option, reason = cc...cc. (S)
```

aa...aa オプションに bb...bb を指定できません。

aa...aa : オプションの指定値

bb...bb : オプション名

cc...cc : エラー理由

extract_all : 抽出定義に extract_all 文を指定する場合、-k オプションに none を指定できません。

(S)処理を中止します。

(O)エラー理由が extract_all の場合：-k オプションを省略、又は none 以外を指定して再実行してください。

KFRB04588-E

```
Invalid <schema name> in <except clause> in <extract_all statement>, lineno = aa...aa,  
schema name = bb...bb, reason = cc...cc. (S)
```

extract_all 文の except 句に指定したスキーマ名が不正です。

aa...aa : 抽出定義ファイル内行番号

bb...bb : スキーマ名

cc...cc : 理由コード

length over : 長さが 30 バイトを超えています。

(S)処理を中止します。

(O)スキーマ名を修正した後、再実行してください。

KFRB04589-E

```
Invalid <table name> in <except clause> in <extract_all statement>, lineno = aa...aa, table  
name = bb...bb, reason = cc...cc. (S)
```

extract_all 文の except 句に指定した表名が不正です。

aa...aa：抽出定義ファイル内行番号

bb...bb：表名

cc...cc：理由コード

length over：長さが 30 バイトを超えています。

(S)処理を中止します。

(O)表名を修正した後、再実行してください。

KFRB04601-E

```
Duplicate <dataset name> in <restruct clause> in <restruct statement> or in <extract  
clause> in <extract statement>, lineno = nn...nn, dataset name = aa...aa. (S + L)
```

restruct 句 又は extract 句に指定しているデータセット名称が重複しています。

nn...nn：更新情報定義ファイル内の行番号

aa...aa：データセット名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04602-E

```
Duplicate <field name> in <field clause> in <restruct statement> or in <extract statement>  
in <extract statement>, lineno = nn...nn, field name = aa...aa. (S + L)
```

field 句 又は extract 句に指定しているフィールド名称が重複しています。

nn...nn：更新情報定義ファイル内の行番号

aa...aa：フィールド名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04603-E

```
Invalid field data position in <position clause> in <restruct statement>, lineno = nn...nn.  
( S + L )
```

position 句のデータ開始位置に指定した値が間違っています。

nn...nn：更新情報定義ファイル内の行番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04604-E

```
Invalid field type length in <attr clause> in <restruct statement>, lineno = nn...nn. ( S +  
L )
```

attr 句の再定義フィールド属性の長さに指定した値が間違っています。

nn...nn：更新情報定義ファイル内の行番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04605-E

```
Invalid <DBM name> in <restruct clause> in <restruct statement> or in <extract clause> in  
<extract statement>, lineno = nn...nn, DBM name = aa...aa. ( S + L )
```

restruct 句 又は extract 句に指定している DBM 名称が間違っています。

nn...nn：更新情報定義ファイル内の行番号

aa...aa：DBM 名称

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04606-E

Invalid <dataset name> in <restruct clause> in <restruct statement> or in <extract clause> in <extract statement>, lineno = nn...nn, dataset name = aa...aa. (S + L)

restruct 句 又は extract 句に指定しているデータセット名称が間違っています。

nn...nn：更新情報定義ファイル内の行番号

aa...aa：データセット名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04607-E

Invalid <field name> in <field clause> in <restruct statement>, lineno = nn...nn, field name = aa...aa. (S + L)

field 句に指定しているフィールド名称が不正です。

nn...nn：更新情報定義ファイル内の行番号

aa...aa：フィールド名称

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04608-E

Cannot specify asterisk as all columns in <extract clause> in <extract statement>, lineno = nn...nn. (S + L)

extract 句でアスタリスクの指定はできません。

nn...nn：更新情報定義ファイル内の行番号

(S)処理を中止します。

(O)フィールド名称を指定した後、再実行してください。

KFRB04609-E

Invalid uoc file name length in <uocname statement>, lineno = nn...nn. (S + L)

UOC ファイル名が間違っています。

nn...nn : 更新情報定義ファイル内の行番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04610-E

```
Cannot specify not null option in <attr clause> in <restruct statement>, lineno = nn...nn,  
field name = aa...aa. (S + L)
```

attr オペランドの not null オプションは指定できません。

nn...nn : 更新情報定義ファイル内の行番号

aa...aa : フィールド名称

(S)処理を中止します。

(O)エラーになった行の not null オプションを削除した後、再実行してください。

KFRB04611-E

```
Specified dsid is not for hdssamqin command, dsid = xx. (S + L)
```

更新情報入力コマンド (hdssamqin コマンド) で指定したデータ連動識別子の種別が間違っています。

xx : データ連動識別子

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04612-E

```
hdssamqin command is already in use. (S + L)
```

更新情報入力コマンド (hdssamqin コマンド) は既に使用中です。

(S)処理を中止します。

(O)実行中のコマンドが終了するのを待ってから再実行してください。実行中のコマンドが存在しない場合、前回のコマンド処理が異常終了している可能性があります。その場合は、コマンドに -o オプションを付けて実行してください。

KFRB04613-E

Invalid update kind in log data file, read position = aa...aa, update kind = bb...bb. (S + L)

更新情報ファイル中の更新種別が間違っています。

aa...aa：更新データの読み込み位置

bb...bb：更新種別

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04614-E

Failed to convert numeric data, read position = aa...aa, field name = bb...bb, error code = nn...nn. (S + L)

数値データの変換に失敗しました。

aa...aa：更新データの読み込み位置

bb...bb：変換に失敗したフィールド名称

nn...nn：エラーコード

1：データけた数のオーバーフロー

2：データ値不正

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04615-E

Update queue file write error, errno = xx...xx. (S + L)

反映情報キューファイルへの格納でエラーが発生しました。

xx...xx：errno にセットされたエラー番号

(S)処理を中止します。

(O)エラー番号を errno.h 又は OS のマニュアルを参照してエラーの原因を取り除いた後、再実行してください。

KFRB04616-E

Invalid kind of queue data, error code = nn...nn. (S + L)

キュー形式入力ファイルのデータが不正です。

nn...nn : エラーコード

- 1 : 先頭がポート情報でない (キュー形式データでない)
- 2 : データ連動識別子が不一致
- 3 : 抽出元 DBMS 不正
- 4 : 送信シーケンス ID 不正
- 5 : 抽出定義情報が存在しない
- 6 : 更新情報が存在しない
- 7 : 更新情報分割コード不正

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04617-E

Internal error, function name = aa...aa, code = nn. (S + L)

内部エラーが発生しました。

aa...aa : 関数名

nn : 終了コード

(S)処理を中止します。

(O)保守員に連絡してください。

KFRB04618-I

Log data file replication completed successfully. (S + L)

更新情報ファイルの入力が完了しました。

(S)処理を終了します。

KFRB04619-E

Invalid data start position, DBM name = aa...aa, dataset name = bb...bb, field name = cc...cc. (S + L)

データ開始位置が間違っています。

aa...aa : DBM 名称

bb...bb : データセット名

cc...cc : フィールド名称

(S)処理を中止します。

(O)更新情報定義ファイルの `restruct` 文を見直してエラーの原因を取り除いた後、再実行してください。

KFRB04620-E

```
hdssamqin command caught signal nn...nn. (S + L)
```

更新情報入力コマンドがシグナルを受信しました。

nn...nn : 受信したシグナル

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04621-E

```
First log data is not 'PH' log. (S + L)
```

先頭ログがログヘッダ情報 (ログ識別: 'PH') ではありません。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04622-E

```
Invalid runid or file number of log data file, the last time runid = xxxx, file number = mm...mm, the present runid = yyyy, file number = nn...nn, endinfo = aa...aa. (S + L)
```

ラン ID 又はファイルの順序番号が間違っています。

xxxx : 最終ラン ID

mm...mm : 最終ファイル番号

yyyy : 現在のラン ID

nn...nn : 現在のファイル番号

aa...aa : 前回実行時の終了情報

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04623-E

```
Used PDMII user change routine. (S + L)
```

PDMII のユーザ変換ルーチンを使った更新情報データは入力できません。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04624-E

```
Failed to get update data from PDMII log file, read position = aa...aa, field name = bb...bb,  
field position = mm...mm, field length = nn...nn, total data size = xx...xx. (S + L)
```

PDMII E2 のログファイルから更新データを取得できませんでした。

aa...aa : 更新データの読み込み位置

bb...bb : フィールド名称

mm...mm : フィールド位置

nn...nn : フィールドの長さ

xx...xx : 全データサイズ

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04625-E

```
No dataset name in 'ED' log, DBM name = aa...aa, dataset name = bb...bb. (S + L)
```

更新抽出定義情報 (ED ログ) に更新情報定義ファイルで指定したデータセット名称が存在しません。

aa...aa : DBM 名称

bb...bb : データセット名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04626-E

No DBM name in 'ED' log, DBM name = aa...aa. (S + L)

更新抽出定義情報 (ED ログ) に更新情報定義ファイルで指定した DBM 名称が存在しません。

aa...aa : DBM 名称

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04627-E

Invalid log data. (S + L)

ログデータが不正です。更新抽出定義情報 (ED ログ) 取得前に更新ログが出力されています。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04628-E

Duplicate 'ED' log data. (S + L)

更新抽出定義情報 (ED ログ) は既に取得しています。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04629-E

No 'ED' log data. (S + L)

更新抽出定義情報 (ED ログ) が未取得です。

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04630-E

Invalid mapping key field name, field name = aa...aa. (S + L)

更新情報定義ファイルで指定したマッピングキーフィールド名称が不正です。

aa...aa : フィールド名称

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04631-E

```
Update queue file is full. (S + L)
```

反映情報キューファイルが満杯です。

(S)処理を中止します。

(O)反映処理の実行によって反映情報キューファイルが空くのを待った後、再実行してください。

KFRB04632-I

```
Log data file read information, read position = nn...nn. (S + L)
```

更新情報ファイル処理済み位置情報です。

nn...nn：読み込み位置（反映情報キューファイルの反映済み位置）

(S)処理を中止します。

(O)以前に出力されているエラーメッセージを参照してエラーの原因を取り除いた後、再実行してください。

KFRB04633-E

```
Extract field is all mapping key, DBM name = aa...aa, dataset name = bb...bb. (S + L)
```

更新情報定義ファイルで指定した抽出フィールドが、すべてマッピングキーです。

aa...aa：DBM 名称

bb...bb：データセット名

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04634-E

```
Name data convert error occurred, aa...aa name = bb...bb, code = nn...nn. (S + L)
```

DBM 名称、データセット名称、フィールド名称、更新情報名のコード変換でエラーが発生しました。

aa...aa：エラーになった名称の種別

DBM：DBM 名称

dataset : データセット名
field : フィールド名称
update : 更新情報名

bb...bb : エラーになった名称

nn...nn : 変換エラーコード

(S)処理を中止します。

(O)エラーになった名称を修正した後、再実行してください。

KFRB04635-E

```
Mapping key field is not a lowest lank field, DBM name = aa...aa, dataset name = bb...bb,  
mapping key field = cc...cc. (S + L)
```

更新情報定義ファイルの extract 句で指定したマッピングキーフィールドが、最下位フィールドレベルのフィールドではありません。

aa...aa : DBM 名称

bb...bb : データセット名

cc...cc : マッピングキーフィールド名称

(S)処理を中止します。

(O)更新情報定義ファイルを修正した後、再実行してください。

KFRB04636-E

```
No extract field name in 'ED' log, DBM name = aa...aa, dataset name = bb...bb, field name  
= cc...cc. (S + L)
```

更新抽出定義情報 (ED ログ) に、更新情報定義ファイルの extract 句で指定したフィールド名称が存在しません。

aa...aa : DBM 名称

bb...bb : データセット名

cc...cc : フィールド名称

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04637-I

Unextracted data is in output file, file name = aa...aa. (S + L)

更新情報入力ファイルからの未抽出データを未抽出情報ファイルに出力します。

aa...aa : 未抽出情報ファイル名

(S)処理を中止します。

(O)未抽出情報ファイルに出力された更新情報を反映する場合は、更新情報定義ファイルの未抽出情報に該当するデータセット名称だけの定義を作成して、-c オプションを指定して hdssamqin コマンドを再実行してください。

KFRB04638-E

Field length too large for reflect column, DBM name = aa...aa, dataset name = bb...bb, field name = cc...cc. (S + L)

フィールド長が反映列に指定できる最大長を超えています。

aa...aa : DBM 名称

bb...bb : データセット名

cc...cc : フィールド名称

(S)処理を中止します。

(O)更新情報定義ファイルを修正した後、コマンドを再実行してください。

KFRB04639-E

Invalid use of length information field, DBM name = aa...aa, dataset name = bb...bb, field name = cc...cc. (S + L)

可変長データのレコード長格納フィールドは、抽出文に指定できません。

aa...aa : DBM 名称

bb...bb : データセット名

cc...cc : フィールド名称

(S)処理を中止します。

(O)更新情報定義ファイルを修正した後、コマンドを再実行してください。

KFRB04640-E

Invalid kind of log data file. (S + L)

更新情報ファイルの種別が不正です。

(S)処理を中止します。

(O) -c オプションを付けて、コマンドを再実行してください。

KFRB04641-E

DBMS license check error, status = aa...aa. (S + L)

DBMS ライセンスチェックエラーが発生しました。

aa...aa : 詳細コード

(S)処理を中止します。

(O)次に示す表に従って対策した後、コマンドを再実行してください。

詳細コード	意味	対策
100	HiRDB Datareplicator Extension をインストールしていません。	HiRDB Datareplicator Extension をインストールした後、再実行してください。
101 102	ライセンスファイルが壊れています。	HiRDB Datareplicator Extension を再インストールしてください。
103	他社 DB 接続時のライセンスがありません。	ライセンスプロダクトをインストールした後、再実行してください。
104	他社 DB 接続ライセンスの有効期限を過ぎています。	システム管理者に連絡してください。

KFRB04642-E

Invalid log data file, file name = aa...aa. (S + L)

更新情報ファイルが不正です。

aa...aa : 更新情報ファイル名

(S)処理を中止します。

(O)更新情報ファイルを再作成した後、コマンドを再実行してください。

KFRB04643-W

Insufficient buffer occurred while editing log data, record size = aa...aa. (S + L)

更新情報編集バッファが不足です。更新レコード長 = = aa...aa.

aa...aa : 更新レコード長

(S)処理を中止します。

(O)hdssamqin コマンドの -l オプションで更新レコード長以上の更新情報編集バッファサイズを指定して、コマンドを再実行してください。

KFRB04701-E

Unable to get directory path list, function = aa...aa, errno = bb...bb. (S + L)

運用ディレクトリファイルから監視対象一覧を取得する処理でエラーが発生しました。

aa...aa : エラーが発生した関数名

bb...bb : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB04702-E

Invalid operation in MIB file. operation = aa...aa. (S + L)

MIB ファイルで指定したコマンドが間違っています。

aa...aa : MIB ファイルで指定したオペレーション

(S)処理を中止します。

(O)エラーが発生したマシンの MIB ファイルの内容が更新された可能性があります。MIB ファイルを Datareplicator をインストールした時点の内容に戻した後、再実行してください。

KFRB04703-E

Memory allocate error, errno = aa...aa, size = bb...bb. (L)

監視対象の情報取得中にメモリ領域を確保できませんでした。

aa...aa : errno に設定されたエラー番号

bb...bb : 領域取得サイズ

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB04704-E

```
Shared memory access error, path = aa...aa, function = bb...bb, errno = cc...cc. (L)
```

監視対象の情報取得中に共用メモリへのアクセスに失敗しました。

aa...aa : 運用ディレクトリ名

bb...bb : エラーが発生した関数名

cc...cc : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

エラー番号については、errno.h 又は OS のマニュアルを参照してください。

KFRB04705-W

```
Unable to access shared memory for exclusion, path = aa...aa. (L)
```

ほかのコマンドからの排他制御によって、Datareplicator の情報を取得できませんでした。

aa...aa : 運用ディレクトリ名

(S)情報を取得できない Datareplicator をスキップして、次の Datareplicator 情報を取得する処理に移ります。

KFRB04706-E

```
Remote control operation format error, operation = aa...aa. (L)
```

MIB ブラウザから指定された遠隔操作処理のオペレーションのフォーマットが間違っています。

aa...aa : MIB ブラウザで指定された遠隔操作のオペレーション

(S)処理を中止します。

(O)MIB ブラウザの SNMP 設定値で指定された遠隔操作処理オペレーションのフォーマットを確認した後、再実行してください。

KFRB04707-E

```
Remote control operation index error, index = aa...aa. (L)
```


MIB ブラウザから指定された遠隔操作処理のオペレーションのインデクス値が間違っています。

aa...aa : MIB ブラウザで指定された遠隔操作の対象になる Datareplicator のインデクス

(S)処理を中止します。

(O)MIB ブラウザの SNMP 設定値で指定された遠隔操作処理オペレーションのインデクスを確認した後、再実行してください。

KFRB04708-I

```
Remote control operation start, operation = aa...aa, bb...b = cc...cc. (L)
```

MIB ブラウザから指定された遠隔操作処理を開始します。

aa...aa : MIB ブラウザで指定された遠隔操作の対象になる Datareplicator のオペレーション

bb...bb : Datareplicator の種別

HDEPATH : 抽出側 Datareplicator

HDSPATH : 反映側 Datareplicator

cc...cc : 運用ディレクトリ名

(S)処理を続行します。

KFRB04709-E

```
Unable to execute remote control operation, command name = aa...aa, errno = bb...bb.  
(L)
```

MIB ブラウザから指定された遠隔操作処理を実行できません。

aa...aa : MIB ブラウザで指定された遠隔操作処理のオペレーションのコマンド名

bb...bb : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04751-E

```
Duplicate <directory path> in <directory file>, directory path = aa...aa. (S)
```

監視対象に追加しようとした運用ディレクトリは、登録済みです。

aa...aa : 指定した運用ディレクトリ名

(S)処理を中止します。

(O)登録済みの運用ディレクトリを確認した後、再実行してください。

KFRB04752-E

```
Not exist <directory path> in <directory file>, directory path = aa...aa. (S)
```

監視対象から削除しようとした運用ディレクトリは、登録されていません。

aa...aa：指定した運用ディレクトリ名

(S)処理を中止します。

(O)登録済みの運用ディレクトリを確認した後、再実行してください。

KFRB04753-E

```
Not exist authorization to access <directory file>. (S)
```

運用ディレクトリファイルへのアクセス権がありません。

(S)処理を中止します。

(O)root ユーザで再実行してください。

KFRB04801-E

```
command format error. (S)
```

Datareplicator ファイルシステム領域の初期化コマンド 又は表示コマンドの形式が間違っています。

(S)処理を中止します。

(O)正しいコマンド形式で再実行してください。

KFRB04802-E

```
Invalid -l operand, file number must be 1 - 255. (S)
```

オペランド (-l) の指定が間違っています。ファイル数は 1 個から 255 個までです。

(S)処理を中止します。

(O)オペランド (-l) の指定を確認した後、再実行してください。

KFRB04803-E

```
Invalid -f operand, max file path length must be under 127 bytes. (S)
```

オペランド (-f) の指定が間違っています。Datareplicator ファイルシステム領域名は 127 文字までです。

(S)処理を中止します。

(O)オペランド (-f) の指定内容を確認した後、再実行してください。

KFRB04804-Q

```
aa...aa = bb...bb is active.
```

```
Force initialization for file system area, do you want to continue initialization (y or n) ?.
```

```
(S)
```

Datareplicator ファイルシステム領域を強制初期化します。初期化するときは「y」を応答してください。初期化しないときは「n」を応答してください。

このメッセージは、hdsfmkfs コマンドに -r force を指定したときに出力されます。

aa...aa : 抽出側システムのときは「HDEPATH」、反映側システムのときは「HDSPATH」

bb...bb : Datareplicator 運用ディレクトリ名

(S)応答を待ちます。

(O)初期化するときは「y」を、初期化しないときは「n」を応答してください。

KFRB04805-E

```
Specified system file area is not initialized, please initialize by hdsfmkfs command, file  
system area = aa...aa. (S)
```

指定した Datareplicator ファイルシステム領域は初期化されていません。hdsfmkfs コマンドで初期化してください。

aa...aa : Datareplicator ファイルシステム領域名

(S)処理を中止します。

(O)hdsfmkfs コマンドでファイルシステム領域を初期化した後、再実行してください。

KFRB04806-E

```
Specified file is not character device, file system area = aa...aa. (S)
```

指定した Datareplicator ファイルシステム領域はキャラクタ型スペシャルファイルではありません。

aa...aa : Datareplicator ファイルシステム領域名

(S)処理を中止します。

(O)正しいキャラクタ型スペシャルファイル名を指定して再実行してください。

KFRB04807-E

```
Specified file is already used by another user, aa...aa = bb...bb, file system area =  
cc...cc. (S)
```

指定した Datareplicator ファイルシステム領域は、ほかの Datareplicator が使っています。

aa...aa : 抽出側システムの場合は「HDEPATH」、反映側システムの場合は「HDSPATH」

bb...bb : 使用中の Datareplicator 運用ディレクトリ名

cc...cc : Datareplicator ファイルシステム領域名

(S)処理を中止します。

(O)領域を使っている Datareplicator が停止した後に再実行してください。

KFRB04808-E

```
Invalid -l operand, shortage number of sectors, device sector = aa...aa, minimum need  
sector = bb...bb. (S)
```

オペランド (-l) の指定が間違っています。指定したファイル数ではセクタ数が不足します。

aa...aa : Datareplicator ファイルシステム領域のセクタ数

bb...bb : 指定したファイル数で必要な最小のセクタ数

(S)処理を中止します。

(O)オペランド (-l) で指定するシステムファイル数を正しく計算した後、再実行してください。

KFRB04809-Q

```
This area is already used by aa...aa = bb...bb.  
Initialization for file system area, do you want to continue initialization (y or n) ?. (S)
```

Datareplicator ファイルシステム領域を初期化します。初期化するときは「y」を応答してください。初期化しないときは「n」を応答してください。

このメッセージは、Datareplicator が使った Datareplicator ファイルシステム領域を初期化するときに出力します。

aa...aa : 抽出側システムの場合は「HDEPATH」、反映側システムの場合は「HDSPATH」

bb...bb : Datareplicator 運用ディレクトリ名

(S)応答を待ちます。

(O)初期化するときは「y」を、初期化しないときは「n」を応答してください。

KFRB04810-E

```
Current path get error occurred , errno = aa...aa. (S)
```

カレントパスの取得 (getcwd) がエラーになりました。

aa...aa : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。又は、Datareplicator ファイルシステム領域名を絶対パスで指定してください。

KFRB04811-E

```
I/O error occurred , type = aa...aa, file system area = bb...bb, errno = cc...cc. (S)
```

Datareplicator ファイルシステム領域の入出力がエラーになりました。

aa...aa : 入出力の種類

bb...bb : エラーになった Datareplicator ファイルシステム領域名

cc...cc : errno に設定されたエラー番号

(S)処理を中止します。

(O)入出力エラーの原因を取り除いた後、再実行してください。

KFRB04812-E

```
Link I/O error occurred , type = aa...aa, file system area = bb...bb, link name = cc...cc,  
errno = dd...dd. (S)
```

Datareplicator ファイルシステム領域のリンクで入出力エラーになりました。

aa...aa : リンクの種類

bb...bb : Datareplicator ファイルシステム領域名

cc...cc : リンク名

dd...dd : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラーの原因を取り除いた後、再実行してください。

KFRB04813-E

```
Insufficient memory , size = aa...aa, errno = bb...bb, table = cc...cc. (S)
```

メモリが不足しました。

aa...aa : 確保しようとしたメモリサイズ

bb...bb : errno に設定されたエラー番号

cc...cc : メモリ上に作成しようとしたテーブル名

(S)処理を中止します。

(O)メモリ不足を解消した後、再実行してください。

KFRB04814-E

```
Duplicate file name, system file name = aa...aa. (S)
```

システムファイル名が重複しています。

aa...aa : システムファイル名

(S)処理を中止します。

(O)正しいシステムファイル名に変更した後、再実行してください。

KFRB04815-E

```
File number over flow, file system area = aa...aa, max file number = bb...bb. (S)
```

hdsfmkfs コマンドで指定したファイル数の上限値を超えました。

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : ファイル数の上限値

(S)処理を中止します。

(O)hdsfstatfs コマンドでファイルシステム領域の状態を確認した後、hdsfmkfs コマンドで指定するシステムファイル数を増加するか、別のファイルシステム領域にシステムファイルを割り当てた後、再実行してください。

KFRB04816-E

```
Shortage of file size, file system area = aa...aa, area size = bb...bb. (S)
```

Datareplicator ファイルシステム領域のサイズが不足しました。

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : Datareplicator ファイルシステム領域サイズ

(S)処理を中止します。

(O)hdsfstatfs コマンドでファイルシステム領域の状態を確認した後、別のファイルシステム領域にシステムファイルを割り当ててから、再実行してください。

KFRB04817-E

```
Too small sector size, file system area = aa...aa, sector size = bb...bb. (S)
```

キャラクタ型スペシャルファイルのセクタ長が小さ過ぎます。セクタ長は 256 バイト以上必要です。

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : セクタ長

(S)処理を中止します。

(O)セクタ長を大きくした後、再実行してください。

KFRB04818-E

```
Exist specified file as character special file, file system area = aa...aa, storage file = bb...bb. (S)
```

Datareplicator ファイルシステム領域に格納しようとするファイルが既にキャラクタ型スペシャルファイルとして存在しています。

aa...aa : Datareplicator ファイルシステム領域名

bb...bb : 格納しようとするファイル名

(S)処理を中止します。

(O)格納しようとするファイルを削除するか、又は格納しようとするファイル名を変更した後、再初期化してください。

KFRB04819-E

```
Invalid -q operand. (S)
```

オペランド (-q) の指定が間違っています。

(S)処理を中止します。

(O)オペランド (-q) の指定を確認した後、再実行してください。

KFRB04820-E

Invalid -n operand. (S)

オペランド (-n) の指定が間違っています。

(S)処理を中止します。

(O)オペランド (-n) の指定を確認した後、再実行してください。

KFRB04821-W

Ignore aa...a request, reason = bb...b, kind = cc...c, inf = dd...d. (C)

aa...a 要求を無視します。理由 = bb...b, 種別 = cc...c, 対象 = dd...d

bb...b のため、aa...a 要求を無視します。

aa...a : リセット要求

send_counter_reset : 送信データ件数のリセット

reflect_counter_reset : 反映処理数のリセット

bb...b : リセット要求を無視した理由

process_starting : プロセス起動中

cc...c : 識別子の種別

send_id : 送信先識別子

ds_id : データ連動識別子

dd...d : 識別子

(S)処理を続行します。

(O)送信データ件数をリセットする場合、次のどちらかの方法でリセットしてください。

- 抽出側の送信プロセスを停止後、-r オプションを指定して、再度送信プロセスを起動する。
- 送信環境定義に eventcntreset パラメタを指定している場合は、該当するイベントを発行する。

また、反映処理数をリセットする場合、次のどちらかの方法でリセットしてください。

- 反映側 Datareplicator を停止後、-r オプションを指定して、反映側 Datareplicator を再起動する。
- 反映環境定義に eventcntreset パラメタを指定している場合は、該当するイベントを発行する。

KFRB04822-I

It reset the number of the aa...a, kind = bb...b, inf = cc...c. (C)

aa...a のリセットを行いました。種別 = bb...b, 対象 = cc...c

aa...a をリセットしました。

aa...a : リセットした項目

send_counter : 送信データ件数

reflect_counter : 反映処理数

bb...b : 識別子の種別

send_id : 送信先識別子

ds_id : データ連動識別子

cc...c : 識別子

(S)処理を続行します。

KFRB04901-E

Definition format error, file = aa...aa, line = bb...bb. (S)

変換定義ファイルの定義フォーマットが不正です。

aa...aa : 変換定義ファイル名

bb...bb : 行番号

(S)処理を中止します。

(O)変換定義ファイル内の行番号が示す行の定義内容を修正した後、再実行してください。

KFRB04902-E

Invalid character code, file = aa...aa, line = bb...bb, kind = cc...cc, reason = dd...dd. (S)

文字コードが不正です。

aa...aa : 変換定義ファイル名

bb...bb : 行番号

cc...cc : 種別

before : 変換元文字コード

after : 変換先文字コード

dd...dd : 不正理由

(S)処理を中止します。

(O)変換定義ファイル内の行番号が示す行に、定義した文字コードを不正理由に従って修正した後、再実行してください。

KFRB04951-E

```
Command(aa...aa) execution error, code = bb...bb, info = cc...cc. (S)
```

現用ファイル複写コマンドでエラーが発生しました。

aa...aa：コマンド名

抽出側：hdefcopy

反映側：hdsfcopy

bb...bb：エラーコード

cc...cc：エラーコードごとの詳細情報（出力情報がない場合、**を表示します）

(S)処理を中止します。

(O)エラーコードごとに、次のように対処してください。

エラーコード	詳細内容	対処方法
1	二重化制御ファイルがありません。	ファイル二重化で運用されていません。ファイル二重化で運用されているノード上で、コマンドを再実行してください。
2	コピー元の物理ファイルは二重化されていません。	二重化に使用している物理ファイル名を指定して、コマンドを再実行してください。
3	二つの物理ファイルは、同一の論理ファイルを構成していません。	同じ論理ファイルを構成している物理ファイル名を指定して、コマンドを再実行してください。
4	二つの物理ファイルのファイル種別が異なります。	ファイル種別が同じ物理ファイルを指定して、コマンドを再実行してください。
5	複写元ファイルのステータスが不正です。	状態表示コマンドで、複写元ファイルのステータスがACTIVE状態であることを確認してください。現用ファイルの複写が必要な場合、複写元にACTIVE状態のファイル名を指定して、コマンドを再実行してください。
6	複写先ファイルのステータスが不正です。	状態表示コマンドで、複写先ファイルのステータスがHOLDであることを確認してください。現用ファイルの複写が必要な場合、複写先にHOLD状態のファイル名を指定して、コマンドを再実行してください。
7	マップトファイルが不正です。	Datareplicatorを再起動した後で、コマンドを再実行してください。
11	二つの物理ファイルのセクタ長が異なります。 詳細情報：複写元ファイルのセクタ長	セクタ長が同じ物理ファイルを指定して、コマンドを再実行してください。
12	複写元ファイルが不正です。	Datareplicatorを初期化するか、正しいファイルを指定して、コマンドを再実行してください。

エラーコード	詳細内容	対処方法
13	複写元ファイルのサイズが不正です。	Datareplicator を初期化するか、正しいファイルを指定して、コマンドを再実行してください。
14	複写先ファイルの容量が不足しています。 詳細情報：複写元のファイルサイズ。	詳細情報で示すサイズのデータを格納できるファイルを複写先の物理ファイルにして、コマンドを再実行してください。
15	指定したファイルのサイズは、ファイルサイズの制限値を超えました。 詳細情報：ファイル名、詳細コード	KFRB00069-E の対処を参照してください。

KFRB04953-E

Command(aa...aa) execution error, code = bb...bb. (S)

二重化ファイル状態表示コマンドでエラーが発生しました。

aa...aa：コマンド名

抽出側：hdefstate

反映側：hdsfstate

bb...bb：エラーコード

(S)処理を中止します。

(O)エラーコードごとに、次のように対処してください。

エラーコード	詳細内容	対処方法
1	二重化制御ファイルがありません。	ファイル二重化で運用されていません。ファイル二重化で運用されているノード上で、コマンドを再実行してください。
2	指定されたファイルは二重化されていません。	二重化に使用している物理ファイル名を指定して、コマンドを再実行してください。
3	ファイルが未使用状態です。	指定したファイルは、Datareplicator で1度も使用されていません。正しいファイル名を指定して、コマンドを再実行してください。

KFRB04954-I

Command(aa...aa) execution success. (S)

コマンドが完了しました。

aa...aa：コマンド名

(S)処理を続行します。

KFRB05001-I

Capture process started. (E)

抽出プロセスを開始します。

抽出プロセスを開始します。

(S)処理を続行します。

KFRB05002-I

Capture process terminated. (E)

抽出プロセスを終了します。

抽出プロセスを終了します。

(S)処理を終了します。

KFRB05003-E

Capture process terminated abnormally, module = aa...aa, line = bb...bb. (E)

抽出プロセスを異常終了します。モジュール名 = aa...aa, 行番号 = bb...bb.

aa...aa : モジュール名 (内部情報)

bb...bb : 行番号 (内部情報)

(S)処理を中止します。

(O)コアダンプ・共用メモリダンプを採取した後、保守員に連絡してください。

KFRB05004-E

Internal error occurred in capture process, addinfo = [aa...aa], runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)

抽出プロセスにおいて内部矛盾を検知しました。詳細情報 = [aa...aa], ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

aa...aa : エラー番号

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を中止します。

(O) 「10.4 要因コード一覧」を参照して、出力されたエラー番号に該当するユーザ対処に従ってください。

KFRB05005-E

```
This data-type is unable to be extracted, data-type = aa...aa.    (E)  
抽出不可能なデータ型です。データ型 = aa...aa.
```

aa...aa：データ型

(S)処理を中止します。

(O)共用メモリダンプ・HiRDB システムログファイル・システムログファイル・データ連動用連絡ファイル
を採取した後、保守員に連絡してください。

KFRB05006-E

```
Error occurred in analyzing HiRDB log file, addinfo = [aa...aa, bb...bb], runid = xx...xx,  
blkno = yy...yy, recno = zz...zz.    (E)  
HiRDB システムログファイルを解析中にエラーが発生しました。詳細情報 = [aa...aa, bb...bb], ラ  
ン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.
```

aa...aa：詳細情報 1 (内部情報)

bb...bb：詳細情報 2 (内部情報)

xx...xx：ラン ID

yy...yy：システムログブロック番号

zz...zz：システムログレコード番号

(S)処理を中止します。詳細情報に内部情報が出力されます。

(O)共用メモリダンプ・HiRDB システムログファイル・システムログファイル・データ連動用連絡ファイ
ルを採取した後、保守員に連絡してください。

KFRB05007-E

```
Error occurred in reading HiRDB log file, addinfo = [aa...aa], runid = xx...xx, blkno =  
yy...yy, recno = zz...zz.    (E)  
HiRDB システムログファイルの入力処理中にエラーが発生しました。詳細情報 = [aa...aa], ラン ID  
= xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.
```

aa...aa：エラー番号

xx...xx：ラン ID

yy...yy：システムログブロック番号

zz...zz：システムログレコード番号

(S)処理を中止します。

(O)「10.4 要因コード一覧」を参照して、出力されたエラー番号に該当するユーザ対処に従ってください。

KFRB05008-I

Accepted stop request, request kind = aa...aa. (E)

抽出プロセスにおいて停止要求を受け付けました。要求種別 = aa...aa.

aa...aa：要求種別を示します。

Normal：正常停止要求

(S)終了処理を開始します。

KFRB05009-E

Invalid version, HiRDB Datareplicator not compatible with HiRDB. (E)

HiRDB と HiRDB Datareplicator のバージョンがありません。

(S)処理を中止します。

(O)

データ連動回復機能を使用していない場合：

各システムのバージョンを合わせてください。

データ連動回復機能を使用している場合：

データ連動回復機能固有のエラーは、すべて KFRB05009-E メッセージのエラー情報ファイルに出力されます。メッセージ本文の内容は無視して「function：」の部分に出力される情報でエラー内容を特定してください。詳細については、「9.6.5(2) アンロードログファイルによるデータ連動回復の実行でエラーとなった場合の対処方法」を参照してください。

KFRB05010-E

Unable to access communication file due to permanent lock error. (E)

データ連動用連絡ファイルへの排他が解除されないため、アクセスできません。

(S)処理を中止します。

(O)共用メモリダンプ・システムログファイル・データ連動用連絡ファイルを採取した後、保守員に連絡してください。

KFRB05011-E

HiRDB overwrites unextracted system-log. (E)

抽出対象の HiRDB システムログへの上書きが発生しました。

(S)処理を中止します。

(O)データ連動を再開する場合は、HiRDB Datareplicator 連携を再開する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

KFRB05012-I

Replication-start-log of HiRDB captured, runid = xx...xx, blkno = yy...yy, recno = zz...zz.
(E)

HiRDB のデータ連動開始ログを検知しました。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

HiRDB のデータ連動開始ログ (HiRDB Datareplicator 連携の開始ログ) を検知しました。

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を続行します。

KFRB05013-I

Replication-end-log of HiRDB captured, runid = xx...xx, blkno = yy...yy, recno = zz...zz.
(E)

HiRDB のデータ連動終了ログを検知しました。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

HiRDB のデータ連動終了ログ (HiRDB Datareplicator 連携の終了ログ) を検知しました。

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を中止します。

(O)データ連動を再開する場合は、HiRDB Datareplicator 連携を再開する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

KFRB05014-E

Unable to read HiRDB system-log, because the value of logiosize operand is smaller than pd_log_max_data_size operand of HiRDB, logiosize = aa...aa. (E)

抽出環境定義ファイルの logiosize オペランドの値が HiRDB の pd_log_max_data_size の値より小さいため、HiRDB システムログを入力することができません。logiosize = aa...aa.

aa...aa : logiosize の値

(S)処理を中止します。

(O)抽出環境定義ファイルの logiosize オペランドの値を修正した後、再起動してください。

KFRB05015-I

Normal termination process started, found normal termination log of HiRDB, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)

HiRDB の正常停止ログ (EOF) を検知したため、抽出プロセスは正常停止処理を行います。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を終了します。

KFRB05016-E

HiRDB replication function is terminated. (E)

HiRDB の HiRDB Datareplicator 連携機能が停止もしくは中断しています。

(S)処理を中止します。

(O)データ連動を再開する場合は、HiRDB Datareplicator 連携を再開する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

KFRB05017-I

Table to be extracted is not defined in this server, server name = aa...aa. (E)

本サーバでは抽出対象の表が定義されていません。サーバ名 = aa...aa.

aa...aa : サーバ名

(S)処理を終了します。

KFRB05018-I

Found the end of HiRDB system-log, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)
HiRDB システムログの終端を検知しました。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を続行します。

KFRB05019-E

Read point in HiRDB system-log skipped, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)
HiRDB システムログのリードポイントがスキップしました。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

HiRDB の抽出プロセスが抽出を完了する前に pdrplstop と pdrplstart が実行された可能性があります。

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を中止します。

(O)データ連動を再開する場合は、HiRDB Datareplicator 連携を再開する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

HiRDB のシステム共通定義に pd_rpl_init_start = N を指定している場合、抽出側 Datareplicator を起動する前に pdrplstart コマンドを実行してください。

KFRB05020-I

The value of extsuppress operand is true, extraction suppressed in this server, server name = aa...aa. (E)
本サーバは抽出除外指定されています。サーバ名 = aa...aa.

aa...aa : サーバ名

(S)処理を続行します。

KFRB05021-W

Ignored event-command because Event-table is not FIX-TABLE, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)

イベント制御用テーブルが FIX 表として作成されていないため、イベントは無視します。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を続行します。ただし、イベント情報は取得しないため、イベントは使えません。

(O)イベントを使わない運用に切り替えてください。

[対策]抽出処理の完了後、抽出側 Datareplicator を停止して、イベント制御用テーブルを再作成した後、hdeprep コマンドを再実行してください。その後、抽出処理を再起動してください。

KFRB05022-W

EOF block is not the last block of read blocks, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)

EOF ブロックが読み込みブロックの最終ブロックではありません。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

このメッセージは、HiRDB のシステムログにデータ連動で抽出できないログを検知したときに出力します。

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz : システムログレコード番号

(S)処理を続行します。

KFRB05023-W

Ignored column-log, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)

列単位ログのため無視します。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

このメッセージは、HiRDB のシステムログにデータ連動用ではない列単位のログを検知したときに出力します。

xx...xx : ラン ID

yy...yy : システムログブロック番号

zz...zz：システムログレコード番号

(S)処理を続行します。

KFRB05025-E

Definition inconsistent. the value of extsuppress operand is true, but table of this server is defined in extract definition, server name = aa...aa. (E)

本サーバは抽出除外指定されているにもかかわらず、本サーバの表が抽出定義に抽出対象表として定義されています。サーバ名 = aa...aa.

aa...aa：サーバ名

(S)処理を中止します。

(O)データ連動を再開する場合、HiRDB Datareplicator 連動を再開する前に、抽出側と反映側の両方のデータ連動の環境を同期を取って初期化して、抽出側 DB を基に反映側 DB を再作成してください。

KFRB05026-I

Normal termination process started, found planned termination log of HiRDB, runid = xx...xx, blkno = yy...yy, recno = zz...zz. (E)

HiRDB の計画停止ログ (EOF) を検知したため、抽出プロセスは正常停止処理を行います。ラン ID = xx...xx, ブロック番号 = yy...yy, レコード番号 = zz...zz.

xx...xx：ラン ID

yy...yy：システムログブロック番号

zz...zz：システムログレコード番号

(S)処理を終了します。

KFRB05027-W

Ignored event-command because specified send identifier is wrong, event code = aa...aa, send identifier = bb...bb. (E)

指定された送信先識別子が不正なため、イベントは無視します。イベントコード = aa...aa, 送信先識別子 = bb...bb.

aa...aa：イベントコード

bb...bb：送信先識別子

(S)該当するイベントを無視して処理を続行します。

KFRB05028-E

Exception data were detected, exception data = aa...aa, message-id = bb...bb. (E)
例外データを検出しました。例外データ = aa...aa, メッセージ ID = bb...bb.

aa...aa : 例外データ

bb...bb : アドバンストキュー内のメッセージ ID

(S)処理を終了します。

(O)データ連動できるデータへの置換を指定した後、抽出処理を再起動してください。

KFRB05029-W

Exception data were detected, exception data = aa...aa, extraction data type = bb...bb, substitution type = cc...cc, message-id = dd...dd. (E)

例外データを検出しました。例外データ = aa...aa, 抽出データ型 = bb...bb, 置換型 = cc...cc, メッセージ ID = dd...dd.

aa...aa : 例外データ

bb...bb : 抽出データ型

cc...cc : 置換パターン

 null : ナル値置換

 high : 最大値置換

 low : 最小値置換

dd...dd : アドバンストキュー内のメッセージ ID

(S)処理を続行します。

KFRB05030-E

Extract data does not belong to extraction target were detected. table-id = aa...aa. (E)
抽出対象表以外の更新情報を検出しました。テーブル ID = aa...aa.

aa...aa : 表 ID

(S)処理を終了します。

(O)再度データ連動する場合は、データ連動のリセットが必要です。

KFRB05031-W

The number of vacant transmission queue files decreased to dd..dd. (C)

空いている抽出情報キューファイルの数が dd..dd になりました。

dd..dd : 抽出情報キューファイルの数

(S)処理を続行します。

(O)送信処理が開始されているかどうかを確認し、開始されていないときは開始してください。

KFRB05034-I

Queue file recovery started,first file = aa...aa,last file = bb...bb (C)

抽出情報キューファイルの回復を開始します。先頭ファイル = aa...aa,最終ファイル = bb...bb

aa...aa : 回復をする抽出情報キューファイルの先頭のファイル名 (qufile001~qufile016)。

bb...bb : 回復をする抽出情報キューファイルの最終のファイル名 (qufile001~qufile016)。

(S)処理を続行します。

KFRB05035-I

Queue file recovery completed,first file = aa...aa,last file = bb...bb (C)

抽出情報キューファイルの回復を完了しました。先頭ファイル = aa...aa,最終ファイル = bb...bb

aa...aa : 回復をした抽出情報キューファイルの先頭のファイル名 (qufile001~qufile016)。

bb...bb : 回復をした抽出情報キューファイルの最終のファイル名 (qufile001~qufile016)。

(S)処理を続行します。

(O)回復を開始したときに出力される KFRB05034-I メッセージと、回復対象ファイルの範囲が同じことを確認してください。

KFRB05037-W

Queue file that was formatted in an old version exists. (C)

古いバージョンの HiRDB Datareplicator で初期化された抽出情報キューファイルがあります。

古いバージョンの HiRDB Datareplicator で初期化された抽出情報キューファイルがあるため、抽出情報キューファイル回復機能は使用できません。

(S)処理を続行します。

(O)抽出情報キューファイル回復機能復を使用するシステムの場合は、抽出側 Datareplicator を初期開始してください。抽出情報キューファイル回復機能を使用しないシステムの場合は、初期開始は必要ありません。

KFRB05038-W

All of queue files are in an initial state. (C)

すべての抽出情報キューファイルが初期状態のため、抽出情報キューファイル回復機能を実行する必要はありません。

(S)処理を中止します。

KFRB05039-E

Unable to execute queue file recovery function, reason = aa...aa. (C)

抽出情報キューファイル回復機能による回復を行うことができません。理由コード = aa...aa

抽出情報キューファイル回復機能の使用条件を満たしていません。

aa...aa : 理由コード

syncterm :

抽出システム定義の syncterm オペランドに true を指定している場合は、抽出情報キューファイル回復機能を使用できません。

(S)処理を中止します。

(O)データ連動回復機能を使用して回復してください。

KFRB05040-I

Datareplicator recovery function started. (C)

データ連動回復機能 (システムログ方式) による回復を開始します。

(S)処理を続行します。

KFRB05041-I

Skip of update information for Datareplicator recovery function completed, additional transaction info = aa...aa. (C)

データ連動回復機能 (システムログ方式) での反映済み更新情報のスキップ処理が完了しました。抽出トランザクション情報 = aa...aa

データ連動回復機能による回復が完了しました。以降は通常のデータ連動として処理を続行し、更新情報の送信を行います。

aa...aa : 最後にスキップした更新情報の抽出トランザクション情報

(S)処理を続行します。

KFRB05042-E

Recover information file is invalid, file name = aa...aa, reason = bb...bb. (C)

回復情報ファイルが不正です。ファイル名 = aa...aa, 理由コード = bb...bb

aa...aa : ファイル名

bb...bb : 理由コード

file not found :

環境変数 HDEPATH で指定したディレクトリの下に回復情報ファイルがありません。

invalid file format :

回復情報ファイルのフォーマットが不正です。

invalid recover information condition :

回復情報ファイル内の回復情報の同期が取れていないため、この回復情報を使用した回復を実行できません。

(S)処理を中断します。

(O)次に示す表に従って対処してください。

理由コード	意味	対策
file not found	環境変数 HDEPATH で指定したディレクトリの下に回復情報ファイルがありません。	次の点を確認してください。 <ul style="list-style-type: none">環境変数 HDEPATH で指定したディレクトリの下に回復情報ファイルが転送されている。回復情報ファイルの名称に誤りがない。
invalid file format	回復情報ファイルのフォーマットが不正です。	次の点を確認してください。 <ul style="list-style-type: none">回復情報ファイルの転送方法がバイナリモードである。回復情報ファイルがキャラクタ型スペシャルファイルの場合、通常ファイルにコピーするときに dd コマンドに指定した count の指定数が正しい。 どちらにも該当しない場合、回復情報ファイルの内容が壊れています。アンロードログファイルによるデータ連動回復を実行してください。
invalid recover information condition	回復情報ファイル内の回復情報の同期が取れていないため、この回復情報を使用した回復を実行できません。	アンロードログファイルによるデータ連動回復を実行してください。

KFRB05043-I

Now, watching capture process, interval = nn...nn, server name = aa...aa. (L)

抽出処理の監視を開始しました。

nn...nn : 監視間隔時間 (分)

aa...aa : 監視対象のサーバ名

(S)処理を続行します。

KFRB05044-W

```
Extraction process stopped, server name = aa...aa. (L)
```

抽出処理の停止を検知しました。

aa...aa : 監視対象のサーバ名

(S)処理を続行します。

KFRB05045-I

```
Stop watching capture process, server name = aa...aa. (L)
```

抽出処理の監視を停止しました。

aa...aa : 監視対象のサーバ名

(S)処理を続行します。

KFRB06001-I

```
Agent process started. (L)
```

エージェントを開始しました。

(S)処理を続行します。

KFRB06002-I

```
Agent process terminated. (L)
```

エージェントを終了しました。

(S)処理を終了します。

KFRB06003-I

```
Variable of agent option is changed. (L)
```

エージェントの設定が変更されました。

(S)処理を続行します。

KFRB06004-E

Agent process is already started. (S + L)

エージェントは起動済みです。

(S)起動処理を中止します。

KFRB06005-E

Agent process is not started. (S + L)

エージェントは起動されていません。

(S)停止処理を中止します。

KFRB06006-E

Agent process start failed. (S)

エージェントの開始処理でエラーが発生しました。

(S)起動処理を中止します。

(O)直前のエラーメッセージを参照してエラーの原因を取り除いた後、再実行してください。

KFRB06007-W

Collector process's output is invalid, process name = aa...aa. (L)

情報収集プロセスの出力が不正です。

aa...aa：プロセス名

(S)監視時間間隔で5回まで処理を続行して、エラーがなくなる場合は処理を終了します。

(O)インストールの不正が原因と考えられます。その場合は再インストールした後、再起動してください。そのほかの原因のときは、保守員に連絡してください。

KFRB06008-W

Collector process error occurred, process name = aa...aa, exit code = bb...bb. (L)

情報収集プロセスでエラーが発生しました。

aa...aa：プロセス名

bb...bb：プロセスの終了コード

(S)監視時間間隔で5回まで処理を続行して、エラーがなくなる場合は処理を終了します。

(O)エラーの原因を取り除いてください。

KFRB06010-E

Other process is changing agent option. (S + L)

ほかのプロセスがエージェントの設定を更新中です。

(S)設定更新処理を中止します。

(O)しばらく待った後、再実行してください。

KFRB06011-I

Command request was accepted, command = aa...aa. (L)

コマンド要求を受け付けました。

aa...aa：コマンドライン

(S)処理を続行します。

KFRB06012-E

File access error was occurred, operation = aa...aa, file = bb...bb, errno = cc...cc. (S + L)

ファイル操作でエラーを検知しました。

aa...aa：操作種別

bb...bb：ファイル名

cc...cc：errno に設定されたエラー番号

エラー番号が0の場合は次のことを示します。

read：ファイル内容不正

write：ディスク容量不足

(S)処理を中止します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06013-E

Failed to connect service control manager, code = aa...a, errno = bb...bb. (S + L)

サービス制御マネージャへの接続に失敗しました。

aa...aa : コード

bb...bb : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06014-E

```
Failed to set state of event object to signaled, function name = aa...aa, event name = bb...bb, errno = cc...cc. (S + L)
```

イベントの設定に失敗しました。

aa...aa : 関数名

bb...bb : イベント名

cc...cc : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06015-E

```
Invalid setup information. (S + L)
```

セットアップ情報が不正です。

(S)処理を中止します。

(O)Datareplicator を再インストールしてください。

KFRB06016-E

```
Failed to regist service handler function, errno = aa...aa. (L)
```

サービスコントロールハンドラの登録に失敗しました。

aa...aa : GetLastError() で取得したエラー番号

(S)処理を中止します。

(O)エラー番号を `errno.h` 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06017-E

```
Failed to create event object, function name = aa...aa, event name = bb...bb, errno = cc...cc. (L)
```

イベントオブジェクトの作成に失敗しました。

aa...aa : 関数名

bb...bb : イベント名

cc...cc : `GetLastError()` で取得したエラー番号

(S)処理を中止します。

(O)エラー番号を `errno.h` 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06018-E

```
Agent process terminated abnormally, error code = aa...aa. (L)
```

エージェントが異常終了しました。

aa...aa : エラーコード

(S)処理を中止します。

(O)マニュアルでエラーコードに対応したメッセージを参照して、エラーの原因を取り除いてください。

KFRB06101-E

```
Signal operation failure, process-id = aa...aa, kind = bb...bb, errno = cc...cc. (S + L)
```

シグナル操作に失敗しました。

aa...aa : 相手プロセスの ID

bb...bb : 送ろうとしたシグナル

cc...cc : `errno` に設定されたエラー番号

(S)処理を中止します。

(O)エラー番号を `errno.h` 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06102-E

Cannot exec child process, process name = aa...aa, errno = bb...bb. (S + L)

子プロセスを実行できません。

aa...aa : 子プロセス名

bb...bb : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06103-E

Cannot fork child process, errno = aa...aa. (S + L)

子プロセスを生成できません。

aa...aa : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06104-E

Invalid command argument. (S + L)

コマンドの引数指定に誤りがあります。

(S)処理を中止します。

(O)コマンドの引数をマニュアルで確認して、再実行してください。

KFRB06105-E

Option value is invalid, option name = aa...aa. (S + L)

コマンドオプションに指定された値が不正です。

aa...aa : コマンドオプション

(S)処理を中止します。

(O)コマンドオプションの値の範囲をマニュアルで確認して、再実行してください。

KFRB06106-E

Insufficient memory, type = aa...aa, size = bb...bb, errno = cc...cc. (L)

メモリが確保できません。

aa...aa : 詳細コード

bb...cc : メモリ割り当てサイズ

cc...cc : errno に設定されたエラー番号

(S)処理を中止します。

(O)エラー番号を errno.h 又は使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB06201-E

Install directory not found. (L)

Datareplicator のインストールディレクトリが取得できません。レジストリ情報が間違っています。

(S)処理を中止します。

(O)Datareplicator を再インストールしてください。

KFRB06202-E

Insufficient memory for SNMP request, size = aa...aa, errno = bb...bb. (L)

SNMP エージェントで、監視側から要求された情報取得 又は遠隔操作を実行するためのメモリが確保できません。

aa...aa : メモリ割り当てサイズ

bb...bb : errno に設定されたエラー番号

(S)要求された情報取得又は遠隔操作を中止します。

(O)使用している OS のマニュアルを参照して、エラー番号からエラーの原因を取り除いてください。その後、監視側で情報取得 又は遠隔操作を再実行してください。

KFRB06203-E

Insufficient memory for MIB analyze, size = aa...aa, errno = bb...bb. (L)

SNMP エージェントで、MIB ファイルを解析するためのメモリが確保できません。

aa...aa : メモリ割り当てサイズ

bb...bb : errno に設定されたエラー番号

(S)SNMP エージェントを停止します。

(O)使用している OS のマニュアルを参照して、エラー番号からエラーの原因を取り除いてください。その後、SNMP サービスを再起動してください。

KFRB06204-E

Invalid command in MIB file, object id = aa...aa. (L)

SNMP エージェントで、MIB ファイルの解析中に間違ったコマンドを発見しました。MIB ファイルが破壊されているおそれがあります。

aa...aa : 間違ったコマンドを指定しているオブジェクト ID

(S)SNMP エージェントを停止します。

(O)Datareplicator をインストールし直して MIB ファイルを回復した後、SNMP サービスを再起動してください。

KFRB06205-E

Collector process time over for 10 sec, process = aa...aa. (L)

SNMP エージェントで、情報収集プロセスが 10 秒経過しても、情報を取得できません。

aa...aa : 情報収集プロセスの詳細情報

(S)情報収集プロセスを打ち切ります。

(O)情報収集プロセスが出力しているメッセージに従って対処した後、監視側から情報取得を再実行してください。

KFRB06206-E

Collector process error occurred, process = aa...aa. (L)

SNMP エージェントで、情報収集プロセスでエラーが起きました。

aa...aa : 情報収集プロセスの詳細情報

(S)情報収集プロセスを終了します。

(O)情報収集プロセスが出力しているメッセージに従って対処した後、監視側から情報取得を再実行してください。

KFRB06207-E

Invalid SNMP request type, request type = aa...aa. (L)

SNMP エージェントで、情報取得 又は遠隔操作で要求したオブジェクト ID の属性が間違っています。SNMP エージェントが処理できる属性は、INTEGER, Counter, DisplayString です。

aa...aa : オブジェクト ID の属性コード

(S)要求した情報取得 又は遠隔操作を終了します。

(O)監視側で要求したオブジェクト ID の属性を確認した後、監視側から情報取得又は遠隔操作を再実行してください。

KFRB06208-E

Invalid SNMP request code, request code = aa...aa. (L)

SNMP エージェントで、情報取得 又は遠隔操作の要求コードが間違っています。SNMP エージェントが処理できる要求コードは、GET, GETNEXT, SET です。

aa...aa : 要求コード

(S)要求した情報取得 又は遠隔操作を終了します。

(O)監視側の要求コードを確認した後、監視側から情報取得 又は遠隔操作を再実行してください。

KFRB06209-E

Invalid SNMP specified value size, value size = aa...aa. (L)

SNMP エージェントで、遠隔操作で指定したパラメタの文字列長が間違っています。

aa...aa : 指定されたパラメタの文字列長

(S)要求した遠隔操作を終了します。

(O)遠隔操作で指定したパラメタの文字列長を確認した後、監視側から遠隔操作を再実行してください。

KFRB06210-E

Invalid SNMP specified value, value = aa...aa. (L)

SNMP エージェントで、遠隔操作で指定した数値パラメタが間違っています。

aa...aa : 指定した数値

(S)要求した遠隔操作を終了します。

(O)遠隔操作で指定した数値の最大値と最小値を確認した後、監視側から遠隔操作を再実行してください。

KFRB06211-E

```
Process create error occurred, process = aa...aa, error code = bb...bb. (L)
```

SNMP エージェントで、情報取得 又は遠隔操作のための情報収集プロセスを作成できません。

aa...aa : 作成しようとしたプロセス情報

bb...bb : CreateProcess 関数で発生したエラー番号

(S)情報取得 又は遠隔操作を中止します。

(O)使用している OS のマニュアルを参照して、エラー番号からエラーの原因を取り除いた後、監視側から情報取得 又は遠隔操作を再実行してください。

KFRB06212-E

```
Wait for process failed, process = aa...aa, error code = bb...bb. (L)
```

SNMP エージェントで、情報取得のための情報収集プロセスの完了を待てません。

aa...aa : 完了待ちしようとしたプロセス情報

bb...bb : WaitForSingleObject 関数で返ったエラー番号

(S)情報取得を中止します。

(O)使用している OS のマニュアルを参照して、エラー番号からエラーの原因を取り除いた後、監視側から情報取得 又は遠隔操作を再実行してください。

KFRB06213-E

```
Unable to get a process exit code, process = aa...aa, error code = bb...bb. (L)
```

SNMP エージェントで、情報収集プロセスの終了コードを取得できません。

aa...aa : 終了コードを得ようとしたプロセス情報

bb...bb : GetExitCodeProcess 関数で返ったエラー番号

(S)情報取得を中止します。

(O)使用している OS のマニュアルを参照して、エラー番号からエラーの原因を取り除いた後、監視側から情報取得 又は遠隔操作を再実行してください。

KFRB06501-E

Specified DSN name is not for SQL Server. (C + S)
SQL Server への DNS が指定されていません。

不正な DSN が指定されました。DSN 名は SQL Server 用ではありません。

(S)処理を中止します。

(O)次のどれかに不正な DSN 名を指定しています。

- 抽出システム定義
- hdeprepS -D
- hdeeventS -D
- hderesstateS -D

正しい DSN 名を指定して、再実行してください。

KFRB06503-E

Error occurred in object creation, object type = aa...aa, object name = bb...bb, errcode = cc...cc, info = dd...dd (C + S)

SQL Server オブジェクトの作成が失敗しました。object type = aa...aa, object name = bb...bb, errcode = cc...cc, info = dd...dd

aa...aa : オブジェクト種別

Index : インデックス

Table : テーブル

Trigger : トリガ

bb...bb : オブジェクト名

cc...cc : SQLSTATE

dd...dd : SQL Server エラーメッセージ

(S)処理を中止します。

[対策]SQL Server のマニュアルを参照し、SQLSTATE 及び SQL Server エラーメッセージを基にエラーの原因を取り除き、再実行してください。

KFRB06504-E

Object does not exists, object type = aa...aa, object name = bb...bb, errcode = cc...cc, info = dd...dd (C + S)

SQL Server オブジェクトが存在しません。object type = aa...aa, object name = bb...bb, errcode = cc...cc, info = dd...dd

aa...aa : オブジェクト種別

Table : 表

bb...bb : オブジェクト名

cc...cc : SQLSTATE

dd...dd : SQL Server エラーメッセージ

(S)処理を中止します。

[対策]SQL Server のマニュアルを参照し、SQLSTATE 及び SQL Server エラーメッセージを基にエラーの原因を取り除き、再実行してください。

KFRB06505-E

SQL Server error, func = aa...aa, return = bb...bb, errcode = cc...cc, info = dd...dd (C + S)

SQL Server でエラーが発生しました。関数名 = aa...aa, 戻り値 = bb...bb, errcode = cc...cc, info = dd...dd

aa...aa : 関数名

bb...bb : 戻り値

cc...cc : SQLSTATE

dd...dd : SQL Server エラーメッセージ

(S)処理を中止します。

[対策]SQL Server のマニュアルを参照し、SQLSTATE 及び SQL Server エラーメッセージを基にエラーの原因を取り除き、再実行してください。

KFRB06506-E

SQL Server version error. This SQL Server version is not supported. (C + S)

SQL Server がサポート対象外のバージョンです。

(S)処理を中止します。

[対策]Datareplicator Extension のマニュアルを参照し、サポート対象バージョンの SQL Server を使用してください。

KFRB06507-E

QUEUE_ID value in the queue table has exceeded aa...aa (99% of its maximum value).
(C)

RPL_EXT_QUEUE_TBL の QUEUE_ID の値が aa...aa(最大値の 99%)を超えました。

aa...aa : QUEUE_ID の最大値の 99%の値

(S)処理を中止します。

[対策]抽出側 Datareplicator を初期開始してください。

KFRB06508-W

"hderesstateS" command may show previous/default value as resource operand table could not be updated with aa...aa, func = bb...bb, return = cc...cc, errcode = dd...dd, info = ee...ee (C)

RPL_EXT_RESOPDTBL の値を aa...aa に更新できませんでした。hderesstateS コマンドは前回、もしくはデフォルト値を表示します。関数名 = bb...bb, 戻り値 = cc...cc, errcode = dd...dd, info = ee...ee

aa...aa : 抽出システム定義の sqls_msgkeepime オペランドで指定した時間を秒単位に変換した値

bb...bb : 関数名

cc...cc : 戻り値

dd...dd : SQLSTATE

ee...ee : SQL Server エラーメッセージ

(S)処理を続行します。

[対策]SQL Server のマニュアルを参照し、SQLSTATE 及び SQL Server エラーメッセージを基にエラーの原因を取り除き、一度抽出側 Datareplicator を停止した後、再実行してください。

KFRB09084-E

Specified group name not found, aa...a. (S + L)

指定した反映グループ名が見つかりません。

aa...a : 反映グループ名

(S)処理を中断します。

[対策]コマンドオプションに正しいグループ名を指定してください。

KFRB09087-E

Output information exceeds the limits aa...a[KB]. (S + L)

解析結果出力ファイルのサイズが上限 (-k オプション指定値) を超えるため、情報の出力を中止します。

aa...a : 解析結果出力ファイルサイズの上限 (-k オプション指定値)

(S)処理を中断します。

[対策]hdsrefinfmt コマンドの-k オプションに指定するサイズを増やすか、又は-t オプションに指定する件数を減らしてください。

KFRB09088-W

The information at requested number can't be output, kind = aa...a, req-num = bb...b, out-num = cc...c, reason = dd...d. (S + L)

要求された件数 (-t オプション指定値) のトランザクション情報を出力することができません。コマンドは、cc...c に示す出力可能な件数分の情報だけを出力します。

aa...a : トランザクション情報種別

un-reflect : 未反映トランザクション情報

reflected : 反映済みトランザクション情報

bb...b : 要求件数 (-t オプション指定値)

cc...c : 出力可能な件数

dd...d : 理由コード

no more data exist in queue file :

反映情報キューファイル中には cc...c 件を超える情報が格納されていないか、又は情報が格納されていた反映情報キューファイルが既に上書きされています。

definition change was detected :

定義変更を検知したため、これより古い反映済みトランザクション情報を出力できません。

maximum number :

出力可能な最大件数 (2,000,000,000 件) に達しました。

(S)処理を続行します。

KFRB09089-E

Option value is invalid, Option = aa. (S + L)

コマンドオプションに指定された値が不正です。

aa : コマンドオプション

(S)処理を中断します。

[対策]コマンドオプションに正しい値を指定してください。

KFRB09090-W

Length of the character string exceed 45 bytes, resource = aa...a, id = bb...b. (S + L)

認可識別子／表識別子／列名の長さが 45 バイトを超えました。hdsrefinfmt コマンドの出力結果には、これらの名称は出力されません (表 ID／列 ID だけが出力されます)。

aa..a : リソース種別

TableName : 認可識別子又は表識別子

ColumnName : 列名

bb...b : 表 ID 又は列 ID

(S)処理を続行します。

KFRB09091-E

Invalid environment variable value, variable name = aa...a. (S + L)

環境変数 aa...a に、不正な値が指定されました。

aa...a : 環境変数名

HDS_RFI_ELANG : 抽出側 DB の文字コード

HDS_RFI_PLANG : 反映側システムの文字コード

(S)処理を中止します。

[対策]環境変数に正しい値を設定してください。

KFRB09092-E

The option requires an argument, Option = aa. (S + L)

コマンドオプション aa には、オプション引数が必要です。

aa : コマンドオプション

(S)処理を中断します。

[対策]正しいコマンドオプションを指定してください。

KFRB09093-E

System call error(function=fstat), Filename=aa...a, errno=bb...b. (S + L)

fstat システムコールでエラーが返されました。

aa...a : ファイル名

bb...b : エラー番号

(S)処理を中断します。

[対策]エラー番号を基に、使用している OS のマニュアルを参照して、エラーの原因を取り除いてください。

KFRB09094-E

Internal error, aa...a. (S + L)

内部矛盾が発生しました。

aa...a : 保守情報

(S)処理を終了します。

[対策]保守員に連絡してください。

KFRB09200-E

Insufficient memory [aa...aa], require size = bb...bb. (S + L)

メモリが確保できません。

aa...aa : 確保しようとしたメモリの種別

bb...bb : 要求サイズ

(S)処理を中止します。

[対策]不要なプロセス又はコマンドを終了させた後、再実行してください。

KFRB09201-E

Duplicate option specified, Option name = aa...aa. (S)

パラメタ (オプション) を重複して指定しています。

aa...aa : 重複したパラメタ (オプション) 名称

(S)処理を中止します。

[対策]重複しているパラメタ（オプション）の一方を削除した後、再実行してください。

KFRB09202-E

```
Exceeds max number of option, Option name = aa...aa. (S)
```

パラメタ（オプション）の指定数が上限を超えています。

aa...aa：上限を超えたパラメタ（オプション）名称

(S)処理を中止します。

[対策]指定するパラメタ（オプション）数を減らした後、再実行してください。

KFRB09203-E

```
Unrecognized option specified. (S)
```

指定できないパラメタ（オプション）を指定しています。

(S)処理を中止します。

[対策]不要なパラメタ（オプション）を削除した後、再実行してください。

KFRB09204-E

```
Necessary option not specified, Option name = aa...aa. (S)
```

必要なパラメタ（オプション）を指定していません。

aa...aa：パラメタ（オプション）名

(S)処理を中止します。

[対策]必要なパラメタ（オプション）を指定した後、再実行してください。

KFRB09205-E

```
Specified value is out of range, Option name = aa...aa. (S)
```

パラメタ（オプション）の指定値は、指定できる範囲（値）ではありません。

aa...aa：パラメタ（オプション）名

(S)処理を中止します。

[対策]パラメタ（オプション）の値を修正した後、再実行してください。

KFRB09206-E

Conflict options are specified, Option name = aa...aa and bb...bb. (S)

排他的関係にあるパラメタ (オプション) を同時に指定しています。

aa...aa, bb...bb : パラメタ (オプション) 名

(S)処理を中止します。

[対策]パラメタ (オプション) の指定をどちらか一方だけにした後、再実行してください。

KFRB09207-W

Unrecognized value specified, ignore this. (S)

パラメタ (オプション) の指定値は、指定できる範囲 (値) ではありませんが、これを無視します。

(S)処理を続行します。結果は想定していないものとなることがあります。

[対策]次回の実行までにパラメタ (オプション) の指定内容を見直してください。

KFRB09208-E

Unanalyzable option specified, argc count = aa...aa,analyze parameter count = bb...bb.

パラメタ (オプション) をすべて解析できません。

aa...aa : 指定したパラメタ数

bb...bb : 解析したパラメタ数

(S)処理を中止します。

[対策]パラメタ (オプション) の指定形式を見直してください。

KFRB09209-E

Too long path name, Option name = aa...aa. (S)

パラメタ (オプション) に指定したファイルのパス長が 255 バイトを超えています。

aa...aa : パラメタ (オプション) 名

(S)処理を中止します。

[対策]パラメタ (オプション) の指定内容を見直してください。255 バイトを超えるときは、カレントディレクトリを移動してください。

KFRB09210-E

Specified value format error, Option name = aa...aa. (S)

パラメタ (オプション) に指定した形式が間違っています。

aa...aa : パラメタ (オプション) 名

(S)処理を中止します。

[対策]指定形式を修正した後, 再実行してください。

KFRB09211-E

End-date is smaller than start-date, Option name = aa...aa. (S)

終了日時が開始日時よりも前になっています。

aa...aa : パラメタ (オプション) 名

(S)処理を中止します。

[対策]日付の指定を修正した後, 再実行してください。

KFRB09212-I

Usage : hdstrcredit -f Input-file [, Input-file] [-o Output-file] [-l Level] [-p pid [, -p pid ...] ...] [-t YYYYMMDDHHMMSS [, YYYYMMDDHHMMSS] [-O Options] (S)

コマンドパラメタ (オプション) 一覧を表示します。

(S)処理を中止します (ここで表示されたパラメタ (オプション) のどれかのエラーに引き続いて出力されるメッセージです)。

KFRB09213-E

System call error (function = aa...aa), errno = bb...bb. (S + L)

システムコールがエラーになりました。

aa...aa : エラーになったシステムコール名

bb...bb : システムコールから返されたエラー番号

(S)処理を中止します。

[対策]エラーの原因を調査, 除去した後, 再実行してください。

KFRB09214-E

Specified file not found, file name = aa...aa. (S)

ファイルが存在しません。

aa...aa：ファイル名

(S)処理を中止します。

[対策]パラメタ（オプション）を修正した後、再実行してください。

KFRB09215-E

No access permission, file name = aa...aa. (S)

ファイルにアクセスできません（権限がありません）。

aa...aa：ファイル名

(S)処理を中止します。

[対策]ファイルに権限を与えるか、権限を持ったユーザで実行してください。

KFRB09216-E

No record in file, file name = aa...aa. (S)

ファイルは空です。

aa...aa：ファイル名

(S)処理を中止します。

[対策]正しいファイル名を指定した後、再実行してください。

KFRB09217-E

Invalid contents in file, file name = aa...aa. (S)

ファイルの形式が正しくありません。

aa...aa：ファイル名

(S)処理を中止します。

[対策]正しいファイル名を指定した後、再実行してください。

KFRB09250-E

```
Internal error, function name = aa...aa, code = bb...bb, information = cc...cc. (S)
```

内部矛盾が発生しました。関数名 = aa...aa, 詳細コード = bb...bb, 保守情報 = cc...cc.

aa...aa : 関数名

bb...bb : 詳細コード

0401 : 未定義メッセージ ID

0402 : 未定義出力先区分

0801 : 未定義プラットフォーム

cc...cc : 保守情報

(S) 詳細コードが 0401, 又は 0402 の場合は, 処理を続行します。

詳細コードが 0801 の場合は, 処理を終了します。

[対策] 保守員に連絡してください。

KFRB09253-I

```
hdeinfoget started. (T)
```

hdeinfoget コマンドを開始しました。

(S) 処理を続行します。

KFRB09254-I

```
hdeinfoget terminated, return code = aa...aa. (T)
```

hdeinfoget コマンドが終了しました。終了コード = aa...aa.

aa...aa : 終了コード

0 : 正常終了

8 : 異常終了

12 : 割り込み (シグナル発生) によって終了

(S) 処理を終了します。

(O) 終了コードが 0 以外の場合は, このメッセージの前に出力されたメッセージを参照して, 対策してください。

KFRB09256-E

```
Invalid option, option = aa...aa, reason code = bb...bb. (S)
```

オプションの指定に誤りがあります。オプション名 = aa...aa, 理由コード = bb...bb.

aa...aa : オプション名

- e : 初期情報出力先ディレクトリ指定オプション
- k : 取得情報種別指定オプション
- n : 取得対象 Datareplicator 識別子指定オプション
- その他 : 引数指定された未定義オプション

bb...bb : 理由コード

- required option : 必須オプション未指定
- duplicate option : オプション重複指定
- not specified value : 指定値の不足
- undefined option : 未定義オプション

(S)処理を終了します。

(O)usage を確認し, 正しい引数を指定して実行してください。

KFRB09257-I

```
hdsinfoget started. (T)
```

hdsinfoget コマンドを開始しました。

(S)処理を続行します。

KFRB09258-I

```
hdsinfoget terminated, return code = aa...aa. (T)
```

hdsinfoget コマンドが終了しました。終了コード = aa...aa.

aa...aa : 終了コード

- 0 : 正常終了
- 8 : 異常終了
- 12 : 割り込み (シグナル発生) によって終了

(S)処理を終了します。

(O)終了コードが0以外の場合は, このメッセージの前に出力されたメッセージを参照して, 対策してください。

KFRB09259-E

```
Invalid directory, option = -e, reason code = aa...aa, directory = bb...bb. (S)
```

指定されたディレクトリが不正です。オプション名 = -e, 理由コード = aa...aa, オプション指定ディレクトリパス = bb...bb.

aa...aa : 理由コード

not exist : 存在しない

write access denied : 書き込み権限なし

not absolute path : 絶対パスで指定されていない

bb...bb : オプション指定ディレクトリパス

(S)処理を終了します。

(O)コマンドの実行ユーザが書き込み可能な既存ディレクトリを, 絶対パスで指定してください。

KFRB09263-E

```
Failed to create directory, reason code = aa...aa, pathname = bb...bb. (S)
```

ディレクトリ作成に失敗しました。理由コード = aa...aa, ディレクトリパス = bb...bb.

aa...aa : 理由コード

already exists : pathname が既に存在する

other : その他

bb...bb : ディレクトリパス

(S)処理を終了します。

(O)

理由コードが already exists の場合

同一サーバマシンにおいて, 同じ初期情報出力先ディレクトリを指定し, 同時 (秒単位) に 2 回以上コマンドを実行できません。初期情報出力先ディレクトリ, 又は実行タイミングを変更して再度実行してください。

理由コードが other の場合

エラー原因を取り除いた後, 再度実行してください。想定されるエラー原因を次に示します。

- ディスク容量不足
- 権限不足

KFRB09264-E

```
Failed to create file, reason code = aa...aa, pathname = bb...bb. (S)
```

ファイル作成に失敗しました。理由コード = aa...aa, ファイルパス = bb...bb.

aa...aa : 理由コード

already exists : pathname が既に存在する

other : その他

bb...bb : ファイルパス

(S)処理を終了します。

(O)

理由コードが already exists の場合

同一サーバマシンにおいて、同じ初期情報出力先ディレクトリを指定し、同時（秒単位）に2回以上コマンドを実行できません。初期情報出力先ディレクトリ、又は実行タイミングを変更して再度実行してください。

理由コードが other の場合

エラー原因を取り除いた後、再度実行してください。想定されるエラー原因を次に示します。

- ディスク容量不足
- 権限不足

KFRB09265-E

```
Not found hdeid in system, hdeid = aa...aa. (S)
```

サーバマシン上に情報取得対象となる抽出側 Datareplicator 識別子が見つかりません。情報取得対象抽出側 Datareplicator 識別子 = aa...aa.

aa...aa : 情報取得対象抽出側 Datareplicator 識別子

-n オプション指定時は、その指定値。

-n オプション省略時は、"any"。

(S)処理を終了します。

(O)情報取得対象となる抽出側 Datareplicator 識別子で HiRDB Datareplicator の動作実績がない場合、該当する抽出側 Datareplicator 識別子を検出できません。HiRDB Datareplicator を起動した後、コマンドを再度実行してください。

KFRB09266-E

```
Not found hdsid in system, hdsid = aa...aa. (S)
```

サーバマシン上に情報取得対象となる反映側 Datareplicator 識別子が見つかりません。情報取得対象反映側 Datareplicator 識別子 = aa...aa.

aa...aa : 情報取得対象反映側 Datareplicator 識別子

-n オプション指定時は、その指定値。

-n オプション省略時は, "any".

(S)処理を終了します。

(O)情報取得対象となる反映側 Datareplicator 識別子で HiRDB Datareplicator の動作実績がない場合, 該当する反映側 Datareplicator 識別子を検出できません。HiRDB Datareplicator を起動した後, コマンドを再度実行してください。

KFRB09267-I

```
Collection of Datareplicator information is started, hdeid = aa...aa. (T)
```

抽出側 Datareplicator 情報の取得を開始しました。抽出側 Datareplicator 識別子 = aa...aa.

aa...aa : 抽出側 Datareplicator 識別子

(S)処理を続行します。

KFRB09268-I

```
Collection of Datareplicator information is terminated, hdeid = aa...aa. (T)
```

抽出側 Datareplicator 情報の取得が終了しました。抽出側 Datareplicator 識別子 = aa...aa.

aa...aa : 抽出側 Datareplicator 識別子

(S)処理を続行します。

KFRB09269-W

```
Character special file is not collected, file = aa...aa. (S)
```

キャラクタ型スペシャルファイルは収集されません。ファイルパス = aa...aa.

aa...aa : ファイルパス

(S)処理を続行します。

[対策]手作業で aa...aa ファイルを取得してください。

KFRB09270-I

```
Collection of OS information is started. (T)
```

OS 情報の取得を開始しました。

(S)処理を続行します。

KFRB09271-I

Collection of OS information is terminated. (T)

OS 情報の取得が終了しました。

(S)処理を続行します。

KFRB09272-I

Collection of Datareplicator information is started, hdsid = aa...aa. (T)

反映側 Datareplicator 情報の取得を開始しました。反映側 Datareplicator 識別子 = aa...aa.

aa...aa：反映側 Datareplicator 識別子

(S)処理を続行します。

KFRB09273-I

Collection of Datareplicator information is terminated, hdsid = aa...aa. (T)

反映側 Datareplicator 情報の取得が終了しました。反映側 Datareplicator 識別子 = aa...aa.

aa...aa：反映側 Datareplicator 識別子

(S)処理を続行します。

KFRB09274-W

Failed to read Troubleshooting file, file = aa...aa. (S)

トラブルシューティング情報ファイルの読み込みに失敗しました。トラブルシューティング情報ファイルパス = aa...aa.

aa...aa：トラブルシューティング情報ファイルパス

(S)処理を続行します。

[対策]このコマンドが取得対象とする情報をマニュアルで確認し、手動で取得してください。

KFRB09275-W

Failed to copy file, source = aa...aa. (S)

ファイルのコピー取得に失敗しました。取得対象ファイルパス = aa...aa.

aa...aa：取得対象ファイルパス

(S)処理を続行します。

(O)手作業でファイルを取得してください。

KFRB09276-W

```
Failed to execute command, command = aa...aa. (S)
```

コマンドの実行結果ファイル出力に失敗しました。コマンド = aa...aa.

aa...aa : コマンド

(S)処理を続行します。

(O)手作業でコマンドを実行し、実行結果を取得してください。

KFRB09277-W

```
Unable to complete archive, directory = aa...aa. (S)
```

取得情報格納ディレクトリのアーカイブ、又は圧縮処理が完了しませんでした。取得情報格納ディレクトリパス = aa...aa.

aa...aa : 取得情報格納ディレクトリパス

(S)処理を続行します。

(O)情報取得処理は完了しています。aa...aa ディレクトリを手作業で取得してください。なお、aa...aa ディレクトリは自動的に削除されないため、取得後手動で削除してください。

KFRB09278-I

```
Complete archive, file = aa...aa. (T)
```

取得情報のアーカイブが完了しました。取得情報アーカイブファイルパス = aa...aa.

aa...aa : 取得情報アーカイブファイルパス

(S)処理を続行します。

KFRB09301-E

```
Specified file not found, file-type = aa...aa, file name = bb...bb. (S)
```

指定されたファイルが見つかりません。

aa...aa : ファイル種別

NMTSTATFILE : 環境変数 HDEPATH 又は -b オプションに指定したサーバ名に誤りがあります。

bb...bb : ファイル名

(S)処理を中止します。

(O)正しい値を指定して、再実行してください。

KFRB09302-E

Unable to execute, because other command is not stopped. (S)

ほかのコマンドが実行中なので、このコマンドを実行できません。

(S)処理を中止します。

(O)実行中のほかのコマンドが終了してから、このコマンドを再実行してください。

KFRB09303-E

Source site Datareplicator is not stopped. (S)

抽出側 Datareplicator が動作中です。

(S)処理を中止します。

(O)抽出側 Datareplicator を停止した後、再実行してください。

KFRB09304-E

Specified queue file is in use, file name = aa...aa. (S)

指定されたキューファイルは使用中状態であるため、登録解除できません。

aa...aa : ファイル名

(S)処理を中止します。

(O)抽出情報キューファイルが未使用状態になってから、再実行してください。

KFRB09305-E

Transmission queue file cannot be added any more, because nn...nn queue files already exist. (S)

既に最大数の抽出情報キューファイルが登録されているため、これ以上追加登録できません。

nn...nn : 登録されている抽出情報キューファイル数

(S)処理を中止します。

KFRB09306-E

Specified file is already in use, file-type = aa...aa, file name = bb...bb. (S)

指定されたファイルは、既に抽出情報キューファイルとして登録されています。

aa...aa：ファイル種別

QUEFILE：抽出情報キューファイル

bb...bb：ファイル名

(S)処理を中止します。

(O)別のファイル名を指定してください。

KFRB09307-E

No more transmission queue files can be removed, because at least nn...nn queue files are required. (S)

抽出情報キューファイルをこれ以上登録解除できません。

nn...nn：登録されている抽出情報キューファイル数

(S)処理を中止します。

KFRB09308-I

Transmission queue file was added successfully. (S)

抽出情報キューファイルの追加登録を完了しました。

(S)コマンドを終了します。

KFRB09309-I

Transmission queue file was removed successfully. (S)

抽出情報キューファイルの登録解除を完了しました。

(S)コマンドを終了します。

KFRB09310-E

Specified file not found in source site system, file-type = aa...aa, file name = bb...bb. (S)

指定されたファイルは、抽出側システムに登録されていません。

aa...aa : ファイル種別

EXTQUEUEFILE : 環境変数 HDEPATH 又は -d オプションに指定した抽出情報キューファイル名に誤りがあります。

bb...bb : ファイル名

(S)処理を中止します。

(O)指定誤りを見直して、再実行してください。

KFRB09311-E

```
Specified file was not found, the file must exist as a character special file, filename =  
aa...aa.      (S)
```

-a オプションに指定したファイルは存在しません。割り当て済みのキューファイルはキャラクタ型スペシャルファイルであるか、又は Datareplicator ファイルシステム領域に格納されているため、-a オプションに指定するファイルはキャラクタ型スペシャルファイルとして作成済みである必要があります。

aa...aa : -a オプションに指定したファイル名

(S)処理を中止します。

(O)追加登録するキューファイル用のキャラクタ型スペシャルファイルを作成してからコマンドを再実行してください。

KFRB09312-E

```
Dualing information for queue file was not found in the specified dual file definition, filename  
= aa...aa.      (S)
```

-n オプションで指定された二重化定義ファイル中に、-a オプションで指定されたキューファイルの二重化情報がありません。

aa...aa : -a オプションに指定したファイル名

(S)処理を中止します。

(O)追加登録するキューファイル用の二重化定義ファイルに、キューファイルの情報を記述してからコマンドを再実行してください。

KFRB09313-E

```
Datareplicator is not use file duplicate function.      (S)
```

Datareplicator はファイル二重化機能を使用していません。キューファイルを二重化で追加することはできません。

(S)処理を中止します。

(O)追加登録するキューファイルを二重化しないで (-n オプションを指定しないで) コマンドを再実行してください。

KFRB10001-I

```
The copy command of active-file was executed, file(aa...aa). (L)
```

現用ファイル複写コマンドが実行されました。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10002-E

```
The mapped-file is broken, file(aa...aa).Please restart Datareplicator. (L)
```

マップトファイルが壊れました。

aa...aa : 論理ファイル名

(S)起動処理を中止します。

(O)HiRDB Datareplicator を再起動してください。

KFRB10003-E

```
An active file does not exist.logicalfile = aa...aa(A:bb...bb,B:cc...cc). (L)
```

現用可能なファイルが存在しません。

aa...aa : 論理ファイル名

bb...bb : A 系状態

cc...cc : B 系状態

(S)起動処理を中止します。

(O)データ連動回復機能を使用して回復した後、初期化して再実行してください。

KFRB10004- I

```
A state of a file was changed. aa...aa->bb...bb(cc...cc). (L)
```

ファイルの状態が変更されました。

aa...aa : 変更前状態

bb...bb : 変更後状態

cc...cc : ファイル名

(S)処理を続行します。

KFRB10005- I

```
A file was opened. logicalfile = aa...aa(A:bb...bb, B:cc...cc).      (L)
```

ファイルがオープンされました。

aa...aa : 論理ファイル名

bb...bb : A 系状態

cc...cc : B 系状態

(S)処理を続行します。

KFRB10006-W

```
A process is downing in writing file. logicalfile = aa...aa(A:bb...bb, B:cc...cc).      (L)
```

ファイル書き込み中にプロセスが終了しました。

aa...aa : 論理ファイル名

bb...bb : A 系状態

cc...cc : B 系状態

(S)処理を続行します。

KFRB10007- I

```
File copy from aa...aa to bb...bb completed.      (S + L)
```

ファイルコピーが完了しました。

aa...aa : コピー元 (正常なファイル)

bb...bb : コピー先 (異常なファイル)

(S)処理を続行します。

KFRB10008-W

The sector length of a file is different, file(aa...aa, bb...bb), sector-size:cc...cc. (L)

ファイルのセクタ長が異なります。

aa...aa : コピー元 (正常なファイル)

bb...bb : コピー先 (異常なファイル)

cc...cc : セクタ長

(S)処理を続行します。

(O)コピー先の RAW デバイスのセクタ長をコピー元と同じにしてください。

KFRB10009-W

The source copying file is invalid, file(aa...aa). (L)

コピー元のファイルが不正です。

aa...aa : コピー元 (正常なファイル)

(S)処理を続行します。

(O)コピー元のファイルが不正です。現用可能なファイルが存在していない場合に出力されます。データ連動回復機能を使用して回復した後、初期化して再実行してください。

KFRB10010-W

The file size of a copying file is invalid, file(aa...aa). (L)

コピー元のファイルサイズが不正です。

aa...aa : コピー元 (正常なファイル)

(S)処理を続行します。

(O)コピー元のファイルが不正です。現用可能なファイルが存在していない場合に出力されます。データ連動回復機能を使用して回復した後、初期化して再実行してください。

KFRB10011-W

The disk capacity of a copy place is insufficient, file(aa...aa), copy-size:bb...bb. (L)

コピー先のディスク容量が不足しています。

aa...aa : コピー先 (異常なファイル)

bb...bb : コピー元のファイルサイズ

(S)処理を続行します。

(O)コピー先のディスク容量を増やしてください。

KFRB10012- I

The copy of file started, file(aa...aa). (L)

ファイルコピーを行います。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10013-W

The copy of file failed, file(aa...aa). (L)

ファイルコピーに失敗しました。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10014-W

The kind of file is invalid, file(aa...aa). (L)

ファイル種別が無効です。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10015-W

A block of file was broken, file(aa...aa). (L)

ファイルのブロックが壊れています。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10016-W

A size of file was zero, file(aa...aa). (L)

ファイルサイズが 0 バイトです。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10017-W

A copy of file was failed, file(aa...aa). (L)

ファイルのコピーに失敗しました。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10018-E

The program unable to continue. Please restart this program, file(aa...aa). (L)

プログラムは、続行できません。プログラムを再起動してください。

aa...aa : ファイル名

(S)処理を停止します。

(O)HiRDB Datareplicator を再起動してください。障害ファイルを回復しておくことを推奨します。

KFRB10019-W

The status of file is hold at last time, file(aa...aa). (L)

現用チェックで、ファイルヘッダのステータスが前回 HOLD でした。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10020-W

The last write time of file is different, file(aa...aa). (L)

最終更新時刻が異なります。

aa...aa : ファイル名

(S)処理を続行します。

KFRB10021-I

A file problem has occurred.Stopping usage of the file. (L)

ファイル障害が発生しました。切り替え処理を行います。

(S)処理を続行します。

KFRB10022-I

Stopping usage of file completed. (L)

切り替え処理が完了しました。

(S)処理を続行します。

KFRB10023-W

Stopping usage of file failed. (L)

切り替え処理が失敗しました。

(S)処理を続行します。

KFRB10024-E

The logical file name is different, A(aa...aa, bb...bb), B(cc...cc, dd...dd). (L)

A系ファイルのヘッダ内に設定されている論理ファイル名と、B系ファイルのヘッダ内に設定されている論理ファイル名が不一致です。

aa...aa : A系ファイルの物理ファイル名

bb...bb : A系ファイルのヘッダ内に設定されている論理ファイル名

cc...cc : B系ファイルの物理ファイル名

dd...dd : B系ファイルのヘッダ内に設定されている論理ファイル名

(S)処理を中止します。

(O)メッセージに出力されているA系、B系ファイルの物理ファイル名と論理ファイル名との対応が、二重化定義ファイルの設定内容と一致しているかどうか確認してください。

一致していない場合は、物理ファイル名として作成したシンボリックリンクのリンク先が不正となっている可能性があります。リンク先が不正であった場合は、正しいリンク先となるようにシンボリックリンクを再作成した後、Datareplicatorを再起動してください。

KFRB11001-E

Invalid command argument.

Usage : aa...aa (S)

コマンドラインが不正です。

aa...aa : コマンド指定方法

(S)処理を中止します。

(O)コマンドラインの指定を修正した後、再実行してください。

KFRB11002-E

Environment variable value invalid, variable name = aa...aa. (S)

環境変数の指定が不正です。

aa...aa : 環境変数名

(S)処理を中止します。

(O)環境変数の指定を修正した後、再実行してください。

KFRB11003-E

Error occurred in aa...aa command. (S)

Datareplicator が起動するコマンドの処理中にエラーが発生しました。

aa...aa : 起動するコマンド名

hdestart : 抽出環境の初期化処理に失敗しました。

hdeprep : 抽出定義プリプロセスファイルの作成処理に失敗しました。

pddef : HiRDB データベース定義ユティリティでエラーが発生しました。

pload : HiRDB データベース作成ユティリティでエラーが発生しました。

(S)処理を中止します。

(O)起動するコマンドが出力するエラー情報を基にエラーの原因を取り除いた後、再実行してください。

KFRB11004-E

Time out error occurred. (S)

タイムアウトが発生したため抽出環境の初期化に失敗しました。

(S)処理を中止します。

(O)hdestart コマンドが出力するエラー情報を基にタイムアウトエラーの原因を調査した後、再実行してください。

KFRB11005-E

```
MSTERRFILE1 exclusion error occurred. (S)
```

抽出マスタエラー情報ファイル (msterrfile1) の入出力処理中に排他エラー、又はタイムアウトが発生したため抽出環境の初期化に失敗しました。

(S)処理を中止します。

(O)次の見直しをした後、再実行してください。

- 排他エラーに関する見直し
コマンド実行者権限、及び抽出マスタプロセスの実行者権限を見直してください。
- タイムアウトに関する見直し
hdestart コマンドが出力するエラー情報を基にタイムアウトエラーの原因を調査してください。

10.3 システムコールエラー一覧

システムコールでエラーになった場合に出力される KFRB00864-W メッセージ及び KFRB00865-E メッセージ中のシステムコール名とパラメタ情報の関係を、次の表に示します。なお、パラメタ情報として出力する必要がある情報がない場合、「**」が出力されます。

表 10-2 システムコールエラー一覧

カテゴリ	システムコール	パラメタ情報	システムコール内容
ファイル関連エラー	close	ファイル名	ファイルをクローズする
	CloseHandle	ファイル名	ファイルをクローズする
	CreateFile	ファイル名	ファイルをオープンする
	DeleteFile	ファイル名	ファイルの削除
	fclose	ファイル名	ストリームを閉じる
	fcntl	なし	ファイルをロックする
	fcntl	なし	ファイルのロックを解除する
	fgets	ファイル名	ストリームから文字列の入力
	fopen	ファイル名	ファイルを開き、ストリームに連結
	fstat	なし	ファイルの情報を取得する
	ftruncate	切り詰める長さ	ファイルを指定した長さに切り詰める
	GetFileSize	なし	ファイルの情報を取得する
	ioctl	なし	セクタ長を取得する
	LockFileEx	なし	ファイルをロックする
	lseek	ファイル名	ファイルポインタを移動する
	open	ファイル名	ファイルをオープンする
	read	ファイル名	ファイルからデータを読み込む
	ReadFile	ファイル名	ファイルからデータを読み込む
	SetEndOfFile	切り詰める長さ	ファイルを指定した長さに切り詰める
	SetFilePointer	ファイル名	ファイルポインタを移動する
	stat	なし	ファイルの情報を取得する
	unlink	ファイル名	ファイルの削除
	UnLockFileEx	なし	ファイルのロックを解除する
	write	ファイル名	ファイルにデータを書き込む
	WriteFile	ファイル名	ファイルにデータを書き込む

カテゴリ	システムコール	パラメタ情報	システムコール内容
メモリ	calloc	メモリの割り当てサイズ	メモリの割り当て
	free	なし	メモリの解放
	GetProcessHeap	なし	メモリの割り当て
	HeapAlloc	メモリの割り当てサイズ	メモリの割り当て
	HeapFree	なし	メモリの解放
	malloc	メモリの割り当てサイズ	メモリの割り当て
	realloc	メモリの割り当てサイズ	メモリの割り当て
セキュリティ	InitializeSecurityDescriptor	なし	セキュリティ識別子の初期化
	SetSecurityDescriptorDacl	なし	ACLにセキュリティ識別子の追加
ファイルマッピング	CreateFileMapping	なし	ファイルマッピングオブジェクト生成
	MapViewOfFile	なし	ファイルマッピングを行う
	mmap	なし	ファイルマッピングを行う
	munmap	なし	ファイルマッピング解除
	UnmapViewOfFile	なし	ファイルマッピング解除
時間	time	なし	経過時間を秒単位で取得
シグナル	sigaction	なし	シグナル受信時のプロセス動作変更
	sigaddset	なし	シグナル集合に追加
	sigdelset	なし	シグナル集合から削除
	sigemptyset	なし	シグナル集合を空に初期化
	sigprocmask	なし	シグナルリストの変更
	SetConsoleCtrlHandler	なし	シグナル受信時のプロセス動作変更
プロセス	atexit	なし	プロセス終了時関数登録

10.4 要因コード一覧

ファイルの二重化に関するメッセージで表示される要因コード一覧を表 10-3 に、ログ入力エラーによるメッセージで表示される要因コード一覧を表 10-4 に、そのほかの要因コード一覧を表 10-5 に示します。

表 10-3 ファイルの二重化に関するメッセージの要因コード一覧

カテゴリ	エラー番号	内容	ユーザの対処
二重化定義解析	-1001	構文に誤りがあります。	二重化定義の構文を見直し、Datareplicator を初期化してください。
	-1002	論理ファイル名が重複しています。	
	-1003	論理ファイル名の指定に誤りがあります。	
	-1004	物理ファイル名が重複しています。	
	-1005	物理ファイル名の指定に誤りがあります。	
	-1007	二重化定義ファイル中に、エントリがありません。	
	-1008	行長が 1,024 バイトを超えました。	
	-1009	オペランド名が不正です。	
二重化制御ファイル操作	-1101	データ連動用連絡ファイルに構成エラーがあります。	データ連動用連絡ファイルを二重化する場合には、論理ファイル名と物理ファイル名 (A 系, B 系のどちらか) の名称を一致させてください。
	-1102	二重化制御ファイルが壊れています。	Datareplicator を初期化してください。
	-1104	現バージョンと互換性のないバージョンの二重化制御ファイルにアクセスしました。	
	-1105	初期化時にシステム定義に file_dupenv オペランドが指定されていましたが、二重化制御ファイルが存在しません。	
	-1106	初期化時にシステム定義に file_dupenv オペランドが指定されていませんでしたが、二重化制御ファイルが存在します。	ファイル二重化機能を使用しない場合には、二重化制御ファイルを削除した後、Datareplicator を初期化してください。
	-1107	物理ファイルに UNIX ファイルとキャラクタスペシャルファイルを混在させることはできません。	二重化定義に指定する物理ファイルは、A 系と B 系のファイル種別を同じにしてください。
	-1108	追加対象の論理ファイルは、既に二重化制御ファイルに登録されています。	hdemodq コマンドに指定した追加キュー二重化定義ファイルの論理ファイル名を見直し、既存の二重化ファイルと重複しないような論理ファイル名を指定してください。

カテゴリ	エラー番号	内容	ユーザの対処
	-1109	追加対象の物理ファイルは、既に二重化制御ファイルに登録されています。	hdemodq コマンドに指定した追加キュー二重化定義ファイルの物理ファイル名を見直し、既存の二重化ファイルと重複しないような物理ファイル名を指定してください。
	-1110	二重化制御ファイルへのアクセスに失敗しました。	syslog ファイル (Windows の場合はイベントログ) に出力されている、エラーメッセージを参照してください。
現用ファイル複写コマンド	-1301	RAW デバイス環境での、複写元と回復対象のセクタ長が違います。	複写元と同じセクタ長に合わせてください。
	-1302	コピー元のファイルが HiRDB Datareplicator のものでないか、又は壊れています。	コピー元のファイルを確認してください。
	-1303	コピー元のファイルサイズが不正、又は壊れています。	
	-1304	複写先のディスク容量が不足しています。	複写先のディスク容量を確認してください。
	-1305	複写先のディスクが、ラージファイルに対応していません。	複写先システムが使用できるファイルサイズの最大値を見直してください。
ファイル二重化機能	-1401	ファイルが壊れています。	現用ファイル複写コマンドを実行してください。
	-1402	ブロックが破壊されています。	このエラーは、次の場合に発生することがあります。この場合、強制初期化オプション (hdsstart -i -f) を指定することで、正常に初期開始できます。 <ul style="list-style-type: none"> バージョンアップ直後に反映側 Datareplicator を初期化する場合 反映情報キューファイルと反映ステータスファイルをキャラクタ型スペシャルファイルで作成した後に、初めて初期化する場合 反映環境定義の statsfile オペランドに指定したステータスファイルの名称を変更する場合に、変更するステータスファイルと同じ名称のファイルが既に存在する場合 上記以外の通常運用時にこのエラーが発生した場合には、現用ファイル複写コマンドを実行してください。
	-1403	空ファイルです。	Datareplicator の初期化に失敗しています。初期化の原因を取り除いて、初期化し直してください。

カテゴリ	エラー番号	内容	ユーザの対処
	-1404	前回、現用ファイル複写コマンドが正常に終了していません。	現用ファイル複写コマンドを実行し直してください。
	-1405	片系運転でも続行できません。	Datareplicator を再起動してください。
	-1406	前回、A系で障害が発生しました。	現用ファイル複写コマンドを実行してください。
	-1407	前回、B系で障害が発生しました。	
	-1408	A系とB系を比較してA系の最終更新時刻が古いため、A系をHOLD状態にしました。	
	-1409	A系とB系を比較してB系の最終更新時刻が古いため、B系をHOLD状態にしました。	
	-1410	現用可能なファイルが存在しません。	Datareplicator のデータ連動回復機能を実行してください。
	-1411	A系とB系の論理ファイル名が異なります。	論理ファイルを構成する物理ファイル名を指定してください。
内部矛盾	-2003	引数が不正です。	保守員に連絡してください
	-2004	発行シーケンス不良です。	
	-2102	FDS 管理テーブルに不在です。	
	-2201	バッファが足りません。	
	-2202	ファイル記述子が不正です。	
その他	-2101	マップトファイルが破壊されています。	OS 又はディスクが不安定な可能性があります。OS 又はディスクの障害を見直してください。
	-2103	Datareplicator の定義値に指定されたファイル数が多過ぎます。	定義値を見直してください。
全ファイル共通	-3001	ファイルの互換性がありません。	抽出側 Datareplicator の場合は hdestart -i コマンドを、反映側 Datareplicator の場合は hdsstart -i -f コマンドを実行してください。

表 10-4 ログ入力エラーによるメッセージで表示される要因コード一覧

カテゴリ	エラー番号	内容	ユーザの対処
ログ入力エラー	-1101	パラメタが不正です。	保守員に連絡してください。
	-1105	プロトコルが不正です。	保守員に連絡してください。
	-1108	定義が不正です。	抽出対象 HiRDB の定義不正を修正した後、抽出側 Datareplicator を再起動してください。

カテゴリ	エラー番号	内容	ユーザの対処
	-1109	環境変数が不正です。	環境変数の設定不正を修正後、抽出側 Datareplicator を再起動してください。
	-1113	メモリ不足です。	環境のメモリ不足を解消した後、抽出側 Datareplicator を再起動してください。
	-1117	ログファイルのオープンに失敗しました。	保守員に連絡してください。
	-1118	ログ入力開始位置が不正です	抽出側 HiRDB の連動状態を打ち切っていない状態で抽出側 Datareplicator を初期化した場合は、抽出側 HiRDB の連動状態を打ち切った後に抽出側 Datareplicator の初期化を行ってください。 抽出情報キューファイル回復機能を実行中の場合、回復に必要なログファイルが上書きされているため、この機能による回復はできません。データ連動回復機能などを使用して回復してください。 上記以外の場合は保守員に連絡してください。
	-1121	バッファサイズが不正です。	抽出環境定義の logiosize オペランド指定値を十分なサイズに変更した後、抽出側 Datareplicator を再起動してください。
	-1143	I/O エラーです。	保守員に連絡してください。
	-1144	ログブロック及びログブロック内部のデータが不正です。	

表 10-5 そのほかの要因コード一覧

カテゴリ	エラー番号	内容	ユーザの対処
ファイルアクセス	-3001	ファイルの互換性がありません。	抽出側 Datareplicator の場合は hdestart -i コマンドを、反映側 Datareplicator の場合は hdsstart -i -f コマンドを実行してください。

10.5 SQL コードの詳細

SQL コードごとの発生要因と対処方法を表 10-6 に示します。

そのほかの SQL コード、もしくは表 10-6 に示した要因に該当しない場合についてはマニュアル「HiRDB メッセージ」を参照してください。

表 10-6 SQL コードごとの発生要因一覧

SQL コード	要因	対処方法
100	<ol style="list-style-type: none"> 1. 反映対象の表を直接操作し、対象行を削除した。 2. データ連動開始前から抽出側 DB と反映側 DB の行の対応が不一致となっている。 3. 抽出定義の extract 文の key 句で指定した列を更新した。 4. 抽出定義の extract 文の key 句で指定した列に重複した値がある。 	<ol style="list-style-type: none"> 1. エラー情報ファイル、又は未反映情報ファイルに出力している情報を基に、エラーの原因となっている行を特定し、削除した行を追加した後、反映処理を再起動してください。 2. 抽出側 DB と反映側 DB の整合性を確保した後、抽出側及び反映側 Datareplicator を初期化してください。 3. 抽出側 DB と反映側 DB の整合性を確保した後、抽出側及び反映側 Datareplicator を初期化してください。マッピングキーを更新する場合、ukey 句を使用してください。 4. 抽出側 DB のインデクスの構成列はすべて extract 文の key 句で指定してください。その後、抽出側 DB と反映側 DB の DB の整合性を確保した後、抽出側及び反映側 Datareplicator を初期化してください。
-210	<p>次の表に対して、非ナル値制約、又は FIX 属性を指定しているおそれがあります。</p> <ul style="list-style-type: none"> • マージ表の sqlconvopt2 オプションを指定している場合 • 時系列情報表の場合 	非ナル値制約を取り除く、もしくは反映対象の表を非 FIX 属性に変更した後、反映処理を再起動してください。
-404	<ol style="list-style-type: none"> 1. 抽出列のデータ長が反映対象列の定義長を超えている。 2. 文字コード変換後のデータ長が反映対象列の定義長を超えている。 	反映対象列の定義長を見直し、反映処理を再起動してください。
-803	<ol style="list-style-type: none"> 1. 反映対象の表を直接操作し、行を追加又は更新した。 2. データ連動開始前から抽出側 DB と反映側 DB の行の対応が不一致となっている。 3. 反映側 DB のユニークチェックによって反映側 DB に行を追加できない。 	<ol style="list-style-type: none"> 1. エラー情報ファイル、又は未反映情報ファイルに出力している情報を基に、エラーの原因となっている行を特定し、追加又は更新した行を元に戻した後、反映処理を再起動してください。 2. 抽出側 DB と反映側 DB の整合性を確保した後、抽出側及び反映側 Datareplicator を初期化してください。 3. 反映側 DB のユニークチェックの対象となるインデクスの構成列を見直してください。その後、抽出側 DB と反映側 DB の整合性を確保した後、抽出側及び反映側 Datareplicator を初期化してください。
-912	<ol style="list-style-type: none"> 1. COMMIT 発行間隔が長い（反映環境定義の cmtintvl の値が大きい）。 	「4.7.3 反映方法の設計」を参照し、次の対処をしてください。

SQL コード	要因	対処方法
	2. 抽出側 HiRDB で実行する pdload をデータ連携している。	<ul style="list-style-type: none">• 反映環境定義の cmtintvl の指定値を小さくし、反映処理を再起動してください。• 反映側 DB の排他制御資源を pdload による挿入件数以上となるようにし、反映処理を再起動してください。

付録

付録 A HiRDB との対応状況

HiRDB の適用 OS と使用できる Datareplicator のバージョン, 及び HiRDB の機能と Datareplicator でのサポート状況について説明します。

付録 A.1 HiRDB の適用 OS と使用できる Datareplicator のバージョン

HiRDB の適用 OS と使用できる Datareplicator のバージョンを次の表に示します。

表 A-1 HiRDB の適用 OS と使用できる Datareplicator のバージョン

HiRDB の適用 OS	32 ビット版 HiRDB ^{*1} で使用できる Datareplicator のバージョン	64 ビット版 HiRDB で使用できる Datareplicator のバージョン
AIX 5L V5.1	06-00 以降	06-00 以降 ^{*2}
AIX 5L V5.2	07-00 以降	07-00 以降 ^{*2}
AIX 5L V5.3	07-03 以降	07-03 以降 ^{*2}
AIX V6.1	08-03 以降	08-03 以降
AIX V7.1	08-06 以降	08-06 以降
HP-UX 11.0	03-00 以降	05-01 以降 ^{*2}
HP-UX 11i	06-00 以降	06-00 以降 ^{*2}
HP-UX 11i V2(PA-RISC)	07-04 以降	07-04 以降 ^{*2}
HP-UX 11i V2(IPF)	—	07-01 以降
HP-UX 11i V3(IPF)	—	08-02 以降
Solaris 8	05-03 以降	05-03 以降 ^{*2}
Solaris 9	07-00 以降	07-00 以降 ^{*2}
Solaris 10	08-00 以降	08-00 以降 ^{*2}
Red Hat Linux 7.1	06-03 以降	—
Red Hat Linux 7.2	06-03 以降	—
Red Hat Enterprise Linux AS 2.1	07-00 以降	—
Red Hat Enterprise Linux AS 3(x86)	07-01 以降	—
Red Hat Enterprise Linux ES 3(x86)	07-01 以降	—
Red Hat Enterprise Linux AS 4(x86)	08-00 以降	—
Red Hat Enterprise Linux ES 4(x86)	08-00 以降	—

HiRDB の適用 OS	32 ビット版 HiRDB※1 で使用できる Datareplicator のバージョン	64 ビット版 HiRDB で使用できる Datareplicator のバージョン
Red Hat Enterprise Linux AS 3(AMD64 & Intel EM64T)	07-03 以降	×
Red Hat Enterprise Linux AS 4(AMD64 & Intel EM64T)	08-00 以降	×
Red Hat Enterprise Linux ES 4(AMD64 & Intel EM64T)	08-00 以降	×
Red Hat Enterprise Linux AS 3(IPF)	—	07-01 以降
Red Hat Enterprise Linux AS 4(IPF)	—	08-00 以降
Red Hat Enterprise Linux 5.1 Advanced Platform (x86)	08-03 以降	—
Red Hat Enterprise Linux 5.1 (x86)	08-03 以降	—
Red Hat Enterprise Linux 5.1 Advanced Platform (AMD/Intel 64)	08-03 以降	—
Red Hat Enterprise Linux 5.1 (AMD/ Intel 64)	08-03 以降	—
Red Hat Enterprise Linux 5.1 Advanced Platform (Intel Itanium)	—	08-03 以降
Red Hat Enterprise Linux 5.1 (Intel Itanium)	—	08-03 以降
Windows 2000	05-02 以降※2	—
Windows XP Professional	07-01 以降※2	—
Windows XP x64 Edition	08-00 以降※2	×
Windows Server 2003	07-00 以降※2	—
Windows Server 2003 x64 Editions	07-04 以降※2	×
Windows Server 2003 R2	08-00 以降※2	—
Windows Server 2003 R2 x64 Editions	08-00 以降※2	×
Windows Server 2003(IPF)	—	07-01 以降
Windows Vista Ultimate	08-01 以降	×
Windows Vista Business	08-01 以降	×
Windows Vista Enterprise	08-01 以降	×
Windows Server 2008 Standard	08-03 以降	08-03 以降
Windows Server 2008 Enterprise	08-03 以降	08-03 以降

HiRDB の適用 OS	32 ビット版 HiRDB※1 で使用できる Datareplicator のバージョン	64 ビット版 HiRDB で使用できる Datareplicator のバージョン
Windows Server 2008 R2	08-04 以降	08-04 以降
Windows 7 Professional	08-04 以降	08-04 以降
Windows 7 Enterprise	08-04 以降	08-04 以降
Windows 7 Ultimate	08-04 以降	08-04 以降
Windows Server 2012	08-06 以降	08-04 以降
Windows 8 Pro	08-06 以降	08-06 以降
Windows 8 Enterprise	08-06 以降	08-06 以降
Windows Server 2012 R2	08-06 以降	08-06 以降
Windows 8.1 Pro	08-06 以降	08-06 以降
Windows 8.1 Enterprise	08-06 以降	08-06 以降

(凡例)

－：HiRDB の適用 OS ではありません。

×：Datareplicator は使用できません。

注

HiRDB/シングルサーバ、及び HiRDB/パラレルサーバ共通の内容です。

注※1

32 ビット版 HiRDB は、POSIX ライブラリ版を含みます。

注※2

HiRDB のバージョンが 08-02 以降の場合、使用できる Datareplicator のバージョンは 08-01 以降になります。

付録 A.2 HiRDB の機能と Datareplicator でのサポート状況

HiRDB がサポートしている機能のうち、Datareplicator の動作に影響がある機能と Datareplicator でのサポート状況を、次の表に示します。

なお、次の表は HiRDB 08-03 及び Datareplicator 08-02 時点の状況です。最新の情報については、次のサイトで公開しているオンラインマニュアルを参照してください。

<http://www.hitachi.co.jp/Prod/comp/soft1/manual/common/hirdb.htm>

表 A-2 HiRDB の機能と Datareplicator でのサポート状況

HiRDB の機能			サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作
連動対象とする表の構成	CREATE TABLE	FIX	抽出側：02-00 以降 (HiRDB は 03-03 以降)	—
			反映側：01-00 以降	
		SHARE	抽出側：02-00 以降	—
			反映側：07-01 以降	
		SUPPRESS DECIMAL	抽出側：03-00 以降	抽出側 DB とは異なる値が、反映側 DB に反映されます。
			反映側：01-00 以降	
		WITHOUT ROLLBACK	抽出側：08-02 以降	抽出側 DB と反映側 DB の内容が不整合となる場合があります。また、反映処理で SQL 実行エラーとなる場合があります。
			反映側：08-02 以降	
		REFERENCES	抽出側：02-00 以降	—
			反映側：01-00 以降 (条件あり※2)	
		ARRAY	抽出側：05-02 以降 (HiRDB は 05-05 以降)	抽出側 DB とは異なる値が、反映側 DB に反映されます。
			反映側：05-02 以降	
		NO SPLIT	抽出側：05-02 以降	抽出側 DB とは異なる値が、反映側 DB に反映されます。
			反映側：01-00 以降	
		CHARACTER SET	抽出側：08-03 以降	抽出側 DB とは異なる値が、反映側 DB に反映されます。
			反映側：08-03 以降	

HiRDB の機能			サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作	
	SUPPRESS		抽出側：03-00 以降	抽出側 DB とは異なる値が、反映側 DB に反映されます。	
			反映側：01-00 以降	—	
	COMPRESSED		抽出側：08-06 以降	抽出側 DB と反映側 DB の内容が不整合となる場合があります。また、反映処理で SQL 実行エラーとなる場合があります。	
			反映側：08-06 以降		
	データ型	INTEGER		抽出側：02-00 以降	—
		SMALLINT		反映側：01-00 以降	
		DECIMAL	精度の最大値が 29 の場合	抽出側：02-00 以降 反映側：01-00 以降	—
			精度の最大値が 38 の場合	抽出側：08-03 以降 反映側：08-03 以降	—
		FLOAT		抽出側：02-00 以降 反映側：01-00 以降	—
		SMALLFLT			
		CHARACTER			
		VARCHAR			
		NCHAR			
		NVARCHAR			
	MCHAR				
MVARCHAR					
DATE					
TIME	(うるう秒を使用しない場合)	抽出側：02-00 以降 反映側：01-00 以降	—		
	(うるう秒を使用する場合)	抽出側：02-00 以降 反映側：01-00 以降 (HiRDB は 08-02 以降)	反映側 HiRDB が 08-02 より前の場合は、反映処理で SQL 実行エラーとなります。		

HiRDB の機能			サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作		
			TIMESTAMP	抽出側：07-00 以降 反映側：07-00 以降	—	
			INTERVAL YEAR TO DAY	抽出側：02-00 以降 反映側：01-00 以降	—	
			INTERVAL HOUR TO SECOND			
			BLOB	抽出側：02-01 以降 反映側：02-01 以降	hdeprep コマンドがエラーとなり、環境を構築できません。	
			BINARY	抽出側：07-00 以降 反映側：07-00 以降	—	
			抽象データ型	SGMLTEXT	抽出側：05-00 以降 反映側：05-00 以降	—
				FREWORD	抽出側：08-02 以降 反映側：08-02 以降	—
				XML	抽出側：08-02 以降 反映側：08-02 以降	—
				ユーザが CREATE TYPE で定義した抽象データ型	抽出側：未サポート 反映側：未サポート	hdeprep コマンドがエラーとなり、環境を構築できません。
			定義長が 256 バイト以上の列を一つ以上持つインデクス			抽出側：未サポート
反映側：01-00 以降	—					
トリガ			抽出側：07-00 以降 反映側：07-00 以降	—		
連動対象とする表への操作	PURGE TABLE	分割されていない表、又は分割方法がサーバ内の横分割の表の場合	抽出側：04-00 以降 (HiRDB は 04-04 以降)	抽出側 HiRDB のバージョンが 04-04 より前であった場合、反映側 DB に対して Purge table が連動されません。		
			反映側：04-00 以降			

HiRDB の機能		サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作
	分割方法がサーバ間の横分割の表の場合	抽出側：未サポート 反映側：04-00 以降	反映側 DB に対して Purge table が連動されません。
	INSERT	抽出側：02-00 以降 反映側：01-00 以降	—
	UPDATE	BLOB 型, 又は 32,001 バイト以上の BINARY 型を連結演算する場合	抽出側 DB とは異なる値が, 反映側 DB に反映されます。
		BLOB 型, 又は 32,001 バイト以上の BINARY 型を後方削除更新する場合	抽出側 DB とは異なる値が, 反映側 DB に反映されます。
		上記以外の場合	—
	DELETE	抽出側：02-00 以降 反映側：01-00 以降	—
	SQL 文全般	USER (抽出側 DB への更新値として USER を使用する。) (反映側 DB への更新値として USER を使用する。)	抽出側 DB の更新値と同じ値が反映されません。
		CURRENT_DATE 値関数 CURRENT_TIME 値関数 (抽出側 DB への更新値として CURRENT_DATE 値関数, 又は CURRENT_TIME 値関数を使用する。) (反映側 DB への更新値として CURRENT_DATE 値関数, 又は CURRENT_TIME 値関数を使用する。)	—
		CURRENT_TIMESTAMP 値関数 (抽出側 DB への更新値として CURRENT_TIMESTAMP 値関数を使用する。) (反映側 DB への更新値として CURRENT_TIMESTAMP 値関数を使用する。)	—
データ連動に影響する	空白変換機能	抽出側：02-00 以降 反映側：05-03 以降	— ハッシュ分割した表に対して, 意図していない

HiRDB の機能				サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作	
HiRDB の 機能					い反映処理が実行され ます。	
	文字 コード 種別	sjis		抽出側：02-00 以降 反映側：01-00 以降	—	
		euc		抽出側：02-00 以降	—	
				反映側：01-02 以降	反映環境を構築できま せん。	
		korea		抽出側：未サポート 反映側：未サポート	環境を構築できませ ん。	
		lang-C		抽出側：02-00 以降 反映側：01-00 以降	—	
		utf-8	UCS-2 の範囲		抽出側：07-01 以降 反映側：07-01 以降	環境を構築できませ ん。
			UCS-4 の範囲		抽出側：07-01 以降	抽出環境を構築できま せん。
					反映側：07-01 以降 (抽出側 DB と反映側 DB の文字コード種別が異なる 場合は未サポートです。)	4 バイト以上の文字 は、未定義文字に変換 して反映されます。
	chinese		抽出側：未サポート 反映側：未サポート	環境を構築できませ ん。		
	系切り 替え	HiRDB と Datareplicat or が同じサー バマシンで稼 働する場合	モニタモード/ サーバモード/ ユーザサーバ ホットスタン バイ	IP アドレス 引き継ぎ あり	抽出側：02-00 以降 反映側：01-00 以降	—
				IP アドレス 引き継ぎ なし	抽出側：06-02 以降	系切り替え発生時に、 抽出側システムが動作 しません。
					反映側：01-00 以降	—
			高速系切り替え機能/ 1：1 スタンバイレス型系切り 替え機能		抽出側：06-02 以降	系切り替え発生時に、 抽出側システムが動作 しません。
反映側：01-00 以降					—	
影響分散スタンバイレス型系 切り替え機能			抽出側：07-01 以降 (Windows 版は未サポー ト)	系切り替え発生時に、 抽出側システムが動作 しません。		

HiRDB の機能				サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作
				反映側：01-00 以降	—
			NVMe over Fabrics による複製ディスクを用いた系切り替え機能	抽出側：未サポート	系切り替え発生時に、抽出側システムが動作しません。
				反映側：未サポート	系切り替え発生時に、反映側システムが動作しません。
		HiRDB と Datareplicator が異なるサーバマシンで稼働する場合		抽出側：未サポート	抽出環境を構築できません。
				反映側：05-01 以降	系切り替え発生後、HiRDB との接続が切れ、反映処理が停止します。
リアルタイム SAN レプリケーション	全同期方式/ 全非同期方式/ ハイブリッド方式/ ログ同期方式	メインサイト		抽出側：02-00 以降 反映側：01-00 以降	—
		リモートサイト		抽出側：未サポート	更新情報が抽出されません。
				反映側：01-00 以降	—
DNS 対応				抽出側：02-00 以降 反映側：01-00 以降	—
連携用機能	Datareplicator の反映トランザクション同期機能を使用する場合			抽出側：07-04 以降 反映側：07-04 以降	特に影響はありません。
	Datareplicator のデータ連動回復機能を使用する場合			[表 9-10 アンロードログファイルによるデータ連動回復をサポートするバージョンと製品の組み合わせ]を参照してください。	データ連動回復機能を使用できません。
インナレプリカ機能	カレント RD エリアへの更新だけを連動対象とする場合			抽出側：06-01 以降	オリジナル RD エリア、及びすべてのレプリカ RD エリアへの更新が連動対象となります。
				反映側：01-00 以降	—
	オリジナル RD エリア、及びすべてのレプリカ RD エリアへの更新を連動対象とする場合			抽出側：02-00 以降 反映側：01-00 以降	—
ユティリティ	pdload -d			抽出側：未サポート 反映側：未サポート	反映処理で SQL 実行エラーとなります。又

HiRDB の機能		サポートしている Datareplicator のバージョン※1	サポートしていない バージョンの Datareplicator を使用した場合の動作
			は、既存の行と pdload で挿入した行が混在します。
	pdload -lp pdload -ln	抽出側：未サポート 反映側：01-00 以降	更新情報が抽出されません。 —
	pdrbal	抽出側：未サポート 反映側：01-00 以降	追加となったバックエンドサーバからは抽出されません。 —
	pdrorg -la	抽出側：未サポート 反映側：01-00 以降	反映処理で SQL 実行エラーとなります。又は、既存の行が二重に反映されます。 —
UAP 実行環境	ログレスモード	抽出側：未サポート	更新情報が抽出されません。
		反映側：01-00 以降	—
	PDCWAITTIME PDSWAITTIME	抽出側：—	特に影響はありません。
		反映側：未サポート	PDCWAITTIME, PDSWAITTIME に 0 を指定した場合の動作で、反映側 DB へ SQL が実行されます。

(凡例)

—：未対応のバージョンはありません。

注※1

サポートバージョンが記載されている場合は、記載されているバージョン以降のすべてのバージョンで抽出、又は反映できます。ただし、条件が記載されている場合は、条件が優先されます。

注※2

反映方式がトランザクション単位反映方式であり、抽出側 DB への更新が、参照表、被参照表の順序で更新されている必要があります。

付録 B Datareplicator 定義の予約語

Datareplicator 定義の予約語を次の表に示します。ここに示す予約語は、大文字／小文字、又は大文字と小文字の混在のすべてが予約語になります。さらに、Datareplicator 抽出定義、Datareplicator 反映定義、更新情報定義のオペランド名も予約語となります。

これらの文字列を、抽出側 Datareplicator の定義、反映側 Datareplicator の定義、及び更新情報定義で使うときは、文字列を「"」（ダブルクォーテーション）で囲んで指定してください。

表 B-1 Datareplicator の予約語の一覧

予約語	抽出側 Datareplicator の定義の予約語	反映側 Datareplicator の定義の予約語	更新情報定義の予約語
<A>	adt	○	○
	and	○	—
	attr	—	○
	binary	—	○
	by	○	—
<C>	char	—	○
	check	○	—
	comp	—	○
	const	—	—
	construct	○	○
<D>	divide	—	○
<E>	except	○	—
	extract	○	○
	extract_all	○	—
	extract_date	—	○
	extract_time	—	○
<F>	field	—	○
	filetype	—	○
	flike	○	—
	float	—	○
	format	—	○
	from	○	○

予約語		抽出側 Datareplicator の定義 の予約語	反映側 Datareplicator の定義 の予約語	更新情報定義の予約語
<G>	group	—	○	—
<H>	hash	—	○	—
	having	—	○	—
	hdsdefserv* ¹	—	○	—
	hdsmain* ¹	—	○	—
	hdsreflect* ¹	—	○	—
	hdssql* ¹	—	○	—
	hdstcpmst* ¹	—	○	—
<I>	in	○	○	—
	ins	—	—	○
	into	—	○	—
	is	—	○	—
	ivl	—	—	○
<K>	key	○	—	○
<L>	lib	○	○	○
	load	—	○	—
<N>	nchar	—	—	○
	name	—	○	—
	nocodecnv	—	○	—
	none	○	○	—
	not	—	○	○
	not_null_unique	○	○	—
	null	—	○	○
<O>	or	○	—	—
	other	—	○	—
	othergrp	—	○	—
<P>	pack	—	—	○
	packns	—	—	○
	pdm	—	—	○
	position	—	—	○

予約語		抽出側 Datareplicator の定義 の予約語	反映側 Datareplicator の定義 の予約語	更新情報定義の予約語
<Q>	que	—	—	○
<R>	reflect_date	—	○	—
	reflect_kind	—	○	—
	reflect_time	—	○	—
	reptype	○	○	○
	restruct	—	—	○
<S>	send	○	—	—
	seq_no	—	○	—
	sqlconvopt1	—	○	—
	sqlconvopt2	—	○	—
<T>	through	○	○	○
	timestamp	—	○	—
	to	○	○	○
	trngroup	—	○	—
<U>	ukey	○	—	○
	unique	○	○	—
	unpack	—	—	○
	unpackns	—	—	○
	uocname	—	—	○
	uocxxx ^{※2}	—	○	—
	upd	—	—	○
	update	—	—	○
<W>	where	○	—	—

(凡例)

- ：予約語であることを示します。
- ：予約語に該当しないことを示します。

注

注※1 で説明している予約語以外は、大文字、小文字と、それらの組み合わせがすべて予約語です。例えば、by の場合は by, bY, By, 及び BY が予約語です。

注※1

すべてが小文字の場合だけ予約語です。

注※2

xxx は、3けたの符号なし整数を表します。

付録 C UNIX 版 Datareplicator と Windows 版 Datareplicator の機能差

UNIX 版 Datareplicator と Windows 版 Datareplicator で異なる機能を次の表に示します。

表 C-1 UNIX 版 Datareplicator と Windows 版 Datareplicator で異なる機能

内容	UNIX 版 Datareplicator	Windows 版 Datareplicator
ファイルシステム	<ul style="list-style-type: none"> UNIX のファイルシステム 通常ファイル又はキャラクタ型スペシャルファイルを使えます。 	<ul style="list-style-type: none"> FAT, NTFS キャラクタ型スペシャルファイルに該当するファイルはありません。
通信プロトコル	<ul style="list-style-type: none"> TCP/IP OSI (メインフレーム側 DB とデータ連動する場合) 	<ul style="list-style-type: none"> TCP/IP
一つの抽出側 Datareplicator に対する反映側 Datareplicator の数の範囲	<p>抽出システム定義 sendcontrol オペランドの指定値で、範囲が決まります。</p> <p>nodemst を指定したとき： 1～64 個</p> <p>sendmst を指定したとき： 1～4096 個</p>	<p>抽出システム定義 sendcontrol オペランドの指定値で、範囲が決まります。</p> <p>nodemst を指定したとき： 1～63 個</p> <p>sendmst を指定したとき： 1～4096 個</p>
送信先識別子の範囲 (抽出側 Datareplicator の指定)	<p>抽出システム定義 sendidxx オペランドの数 (xx は 01～64)。</p> <p>又は抽出システム定義 sendidxxxx オペランドの数 (xxxx は 0001～4096)。</p>	<p>抽出システム定義 sendidxx オペランドの数 (xx は 01～63)。</p> <p>又は抽出システム定義 sendidxxxx オペランドの数 (xxxx は 0001～4096)。</p>
送信プロセス最大起動数 (抽出側 Datareplicator の指定)	1～1024	1～63
データ連動識別子の範囲 (反映側 Datareplicator の指定)	反映システム定義 dsidxxx オペランドの数 (xxx は 001～128)。	反映システム定義 dsidxxx オペランドの数 (xxx は 001～063)。
UOC の作成方法	実行形式ファイル	DLL ファイル
メッセージの出力先	<ul style="list-style-type: none"> 標準エラー出力 エラー情報ファイル 稼働トレースファイル syslog ファイル 	<ul style="list-style-type: none"> 標準エラー出力 エラー情報ファイル 稼働トレースファイル イベントログ
syslog ファイル, 又はイベントログへの抽出ノード マスタエラー情報ファイル内の情報の出力制御	node_syslogout オペランドで制御。	syslogout オペランドで制御。

付録 D Datareplicator のバージョンダウン

Datareplicator をバージョンダウンする方法について説明します。

付録 D.1 製品形名によるバージョンダウン手順の違い

Datareplicator はバージョン（例：Version 10 と Version 8）によって製品形名が異なります。HiRDB/パラレルサーバで Datareplicator を使うときは、システムマネージャ及びすべてのバックエンドサーバの Datareplicator でバージョンを合わせてください。

付録 D.2 バージョンダウンの手順

Datareplicator をバージョンダウンする手順について説明します。

1. 抽出側 Datareplicator, 及び反映側 Datareplicator の抽出, 反映が完了していること（未抽出及び未反映のデータが残っていないこと）を確認します。
2. HiRDB と Datareplicator を正常終了します。HiRDB/パラレルサーバのときは、すべてのサーバを正常終了してください。
3. Datareplicator をアンインストールします。
4. バージョンダウンする Datareplicator をインストールします。
5. 次に示す Datareplicator の項目をバージョンアップ時に変更していた場合、バージョンアップ前の状態に戻します。
 - 各種定義ファイル
 - 外字マッピングテーブル
 - UOC
6. Datareplicator を初期化します。
7. HiRDB 及び Datareplicator を起動します。

なお、この方法でバージョンダウンできない場合は、「6.5.6(1) 反映側 DB の再作成時の運用手順」で示す手順に従って初期化した上でバージョンダウンしてください。

付録 D.3 バージョンダウン後の注意

バージョンダウン後に注意することを次に示します。

- バージョンダウンする前と後の製品形名が異なる場合、又は「[2.11.3 バージョンアップ時の注意](#)」に示すファイル互換性がない場合は、バージョンダウン後に初めて Datareplicator を開始するときに必ず初期開始を実行してください。
- 64 ビット版 Datareplicator から 32 ビット版 Datareplicator にバージョンダウンする場合、UOC も 32 ビット版に戻してください。
- 反映情報編集 UOC を使用している場合には、バージョンダウン後、反映情報編集 UOC を再リンクしてください。リンク手順の詳細は、「[8.1.2 反映情報編集 UOC の作成手順 \(UNIX\)](#)」を参照してください。

(英字)

ADT (Abstract Data Type)

→ 「[抽象データ型](#)」を参照してください。

BES (Back End Server)

→ 「[バックエンドサーバ](#)」を参照してください。

BLOB (Binary Large Object)

文書、画像、音声などの長大なデータのことで。

DS (Dictionary Server)

→ 「[ディクショナリサーバ](#)」を参照してください。

FES (Front End Server)

→ 「[フロントエンドサーバ](#)」を参照してください。

FIX 属性の表

行の長さが固定長の表のことです。

FREEWORD 型

→ 「[抽象データ型](#)」を参照してください。

LOB データ

文書、画像、音声などの長大な可変長データのことです。LOB データは、ユーザ LOB 用 RD エリアに格納します。LOB 列構成基表を格納したユーザ用 RD エリアとは別にデータロード、及びデータベースの再編成ができます。

MGR (System Manager)

→ 「[システムマネージャ](#)」を参照してください。

RD ノード

分散データベース機能で、ネットワーク上の位置で区別する各 DBMS のことです。分散ネットワーク上の接続先サーバ DBMS を指定するためのノード名のことには RD ノード名称といいます。RD ノード名称は、OSI-RDA プロトコルではリソース名称のことです。

SGMLTEXT 型

→ 「[抽象データ型](#)」を参照してください。

TCP/IP

米国国防総省高等研究計画局（DARPA）のプロジェクトである ARPANET が開発したプロトコルのことです。TCP/IP プロトコルは、主に LAN で使われます。

UAP (User Application Program)

業務をプログラムとして作成したものです。アプリケーションプログラムともいいます。

XML 型

→ 「[抽象データ型](#)」を参照してください。

(ア行)

アソシエーション

OSI プロトコルを使って通信するための、論理的な通信路のことです。

アドバンストキュー

Oracle が提供するメッセージキューイング機能です。詳細については、Oracle のマニュアルを参照してください。

アプリケーション

HiRDB の業務処理の総称です。アプリケーションとして作成するプログラムをアプリケーションプログラム（又はユーザアプリケーションプログラム UAP）といいます。

アンロード統計ログファイル

統計ログファイルの内容をアンロードして作成したファイルのことです。

アンロードログファイル

システムログファイルの内容をアンロードして作成したファイルのことです。

インタフェース領域

HiRDB と UAP との間で情報をやり取りするための領域のことです。

インデクス

データ検索の処理速度を上げるため、表中の特定の列の値を基に作成した索引のことです。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、キーとキー値から構成されます。キーとは列の内容を示した列名のことで、キー値とは列の値のことです。インデクスには、単一列インデクスと複数列インデクスがあります。単一列インデクスとは、表の一つの列に作成した一つのインデクスのことです。また、複数列インデクスとは、表の複数の列で作成した一つのインデクスのことです。

埋込み型 UAP

高級言語（C 言語又は COBOL 言語）で記述されたソースプログラムに、直接 SQL を記述する UAP のことです。

(カ行)

キーレンジ分割

表を横分割する方法のことで、表を構成する列のうち、特定の列が持つ値の範囲を条件として表を横分割することです。表を横分割するときの条件にした特定の列を分割キーといいます。

クライアント／サーバ

プログラムとプログラムで通信する場合の関係を示す用語です。業務処理を依頼する方をクライアント、要求を受けて業務を実行する方をサーバといいます。クライアント／サーバとは、プログラム間の相対的な関係を示します。クライアント／サーバはプログラムを示すこともあれば、ワークステーションやパーソナルコンピュータなどのマシンを示すこともあります。

クラスタキー

特定の列の値の昇順又は降順に行を格納するためのキーとして指定した列のことです。

繰返し列

リレーショナルデータベースで扱う表で、一つのセルに複数の要素を持つ列のことです。複数の要素を一つのセルに格納できるため、従来は複数の表に分けていた情報を一つの表で表現できます。

Datareplicator を使ったデータ連動では、繰返し列を一つの列として扱います。繰返し列の要素単位でのデータ連動はできません。

グローバルバッファ

HiRDB 本体と HiRDB ファイルとの、データの入出力に使うバッファです。

グローバルバッファプール

同じ大きさのグローバルバッファの集まりです。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 エリア又はインデクスには、必ずグローバルバッファプールを割り当てます。

更新情報

抽出側 DB の行に対して行われた更新内容を示す情報のことです。

サーバ

HiRDB のマニュアルでは、HiRDB システムを構成する機能の単位のことを示します。

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

次に示す HiRDB のファイルを総称してシステムファイルといいます。

- システムログファイル
- シンクポイントダンプファイル
- ステータスファイル

システムマネージャ (System Manager)

HiRDB/パラレルサーバの構成要素の一つです。主にコマンドやユーティリティの実行を制御します。

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ファイルのことです。

シンクポイントダンプファイル

HiRDB を再開始するとき、又はデータベースを回復するときに必要な HiRDB 管理情報を格納するファイルのことです。一定の時間間隔でシステムログをシンクポイントで区切り、区切った時点での HiRDB 管理情報を保存します。障害が起こったときは最新のシンクポイントダンプの情報とそれ以降のシステムログ情報を参照してシステムを回復します。取得したシステムログを最初から読み込まなくて済むので、システムの回復時間を短縮できます。

スーパーユーザ (superuser)

UNIX OS の最高権限を持つユーザのことです。UNIX ファイルシステムのすべてのファイルに対してアクセス権を持ちます。スーパーユーザの利用者名称は root です。

ステータスファイル

HiRDB を再開始するときに必要な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。

(タ行)

抽象データ型 (Abstract Data Type)

マルチメディアデータや空間データなどの複雑で大容量のデータを、任意で定義したデータ型のことです。SQL3 をオブジェクト指向に拡張した規格に従ったデータ型です。抽象データ型として、複雑な構造を持つデータや操作をユーザが独自に定義して使えます。

Datareplicator がデータ連動の対象にできる HiRDB の抽象データ型は、SGMLTEXT 型、FREEWORD 型、及び XML 型です。SGMLTEXT 型及び FREEWORD 型を使うときは、HiRDB Text Search Plug-in 又は HiRDB XML Extension が必要です。XML 型を使うときは、HiRDB XML Extension が必要です。

通常ファイル

UNIX の単純な動的ファイルのことです。生成や削除が UNIX のコマンドのできるので、テスト用にデータベースを作成する際に利用するのに適しています。

ディクショナリサーバ (Dictionary Server)

HiRDB/パラレルサーバの構成要素の一つで、データディクショナリを管理するサーバです。

データウェアハウス (Data Warehouse)

データウェアハウスとは、基幹系システムのデータを、情報系システムで、エンドユーザが利用しやすい形にしていつでも利用できるように管理する仕組みのことをいいます。

基幹系業務のデータベースは、データの更新処理を効率的かつ高速に実行するために作成されています。しかし、情報系業務のデータベースでは、処理の効率性や高速性よりも、エンドユーザの目的に応じたデータを、使いやすい形で管理することが求められます。データウェアハウスは、このような、情報系業務でのデータの有効活用のための仕組みです。データウェアハウスを構築すると、基幹系業務で蓄積されたデータを、エンドユーザの意思決定支援や業務分析などの情報系業務で有効に活用できます。また、更新されたデータを時系列順に蓄積するデータベース (時系列データベース) を作成すると、情報の変化の動向や傾向を分析することもできます。

Datareplicator と HiRDB Dataextractor のレプリケーション機能を使って、HiRDB サーバをデータウェアハウスのデータベースとして活用できます。

データ操作言語

適用業務プログラムがデータベースを操作するときの、データベース操作を規定する言語のことです。

データ定義言語

データベースの構成や内容を定義する言語です。

データディクショナリ

データベース・テーブル構造、列定義、インデクス定義などを含むデータベース設計仕様を格納したディクショナリのことです。一つのデータベースは実際には複数のサーバに分割格納されていて (HiRDB/パラレルサーバの場合)、これらを統括管理します。

データディクショナリ LOB 用 RD エリア

ストアドプロシジャを格納する RD エリアのことです。ストアドプロシジャの定義ソース格納用と、オブジェクト格納用の二つがあります。

デッドロック

複数のトランザクションが複数の資源を確保しようとして、互いのトランザクションが資源を解放するのを待ち続けて処理が止まってしまう状態のことです。

同期点

トランザクション処理の区切りのことです。トランザクション処理が正常終了したことを示す同期点処理をコミット、トランザクション処理がうまくいかなかったため無効にする同期点処理をロールバックといいます。

統計ログファイル

HiRDB が出力する統計情報 (統計ログ) を格納するファイルです。

トランザクション

ファイルからデータを読み出して、変更したデータを書き込む処理 (更新処理) では、データの一貫性を保持するため、途中で分けられません。このような処理の単位をトランザクションといいます。トランザクションの処理結果は、有効にするか無効にするかのどちらかに必ず決められます。

トリガ

ある表への操作 (更新, 挿入, 削除) を契機に, 自動的に SQL が実行される動作のことです。トリガを使用すれば, ある表が更新されたときに, その更新を契機に関連するほかの表も自動的に更新するなどの運用ができます。

(ナ行)

ナル値

値が設定されていないことを示す値のことです。

ノード

分散システムでネットワークを構成するそれぞれの通信制御本体、及びそれにつながるシステムのことです。ホスト又は HiRDB システムを意味します。

(ハ行)

バックエンドサーバ (Back End Server)

HiRDB/パラレルサーバの構成要素の一つです。データベースへのアクセスや排他制御を実行する機能と、ソートマージの仕事のほかのバックエンドサーバから依頼され実行する機能を持ちます。

ビュー表

実際にある表（実表）から特定の行や列を選択して、新たに定義した仮想の表のことです。

プラグイン

HiRDB のデータ管理を拡張する機能のことです。HiRDB の抽象データ型を使って、多様なデータを検索できます。HiRDB のプラグインには、次に示す種類があります。

- 日本語全文検索プラグイン (HiRDB Text Search Plug-in)
- 画像検索プラグイン (HiRDB Image Search Plug-in)
- 空間検索プラグイン (HiRDB Spatial Search Plug-in)

フレキシブルハッシュ分割

表を横分割する方法のことで、表を構成する列が持つ値をハッシュ関数を使って、均等に RD エリアに格納して、表を横分割することです。表を横分割するときの条件にした特定の列を分割キーといいます。

プロセス

UAP 又は HiRDB などのアプリケーションソフトが OS で稼働すると生成される作業領域で、プログラムの実行単位のことです。各プロセス単位に仮想空間や時分割された CPU 資源が割り当てられます。

フロントエンドサーバ (Front End Server)

HiRDB/パラレルサーバの構成要素の一つです。クライアントとの接続や、バックエンドサーバへの処理の分配と処理結果の収集をします。

分割キーインデクス

表を横分割するとき格納条件を指定した列（分割キー）を第 1 構成列とするインデクスのことで、次のインデクスのことです。

- 分割キーに作成した単一系列インデクス

- 分割キーを先頭とする複数の列に作成した複数列インデクス

分割キーインデクス以外のインデクスを非分割キーインデクスといいます。

ポート番号

ネットワーク上で利用できるサービスに割り当てられる番号です。

ホスト

ネットワークにつながれた、HiRDB が稼働する一つの計算機（マシン）のことです。HiRDB/パラレルサーバの場合は、複数のホストにサーバを分散して配置します。マルチ HiRDB の場合は、複数の HiRDB から構成される一つのホストになります。

ボリューム

セグメントを格納するための割り当て単位です。

(マ行)

マルチ HiRDB

一つのホストに複数の HiRDB がある形態です。それぞれの HiRDB は個別に運用できます。それぞれの HiRDB ごとに、別々の HiRDB 管理者を登録してください。

(ヤ行)

ユーザ LOB 用 RD エリア

BLOB データ（文書、画像、又は音声などの長大なデータ）を格納するための RD エリアです。

横分割

一つの表、インデクス、又は LOB 列を、複数のユーザ用 RD エリア、又はユーザ LOB 用 RD エリアに分割して格納することを横分割といいます。表を横分割する場合、この表に作成するインデクスも、横分割する表に対応させて横分割できます。また、表に LOB 列が含まれる場合、横分割する表に対応して複数のユーザ LOB 用 RD エリアに分割して格納できます。

(ラ行)

レプリケーション機能

メインフレームと HiRDB、又は HiRDB と HiRDB を連携して、あるデータベースの情報をほかのシステムのデータベースに反映する機能です。HiRDB のレプリケーション機能には、データ連動機能（HiRDB Datareplicator）とデータ抽出・反映サービス機能（HiRDB Dataextractor）があります。

連動回復用バックアップファイル

アンロードログ情報が格納されている HiRDB ファイルシステム領域内の HiRDB ファイルを、HiRDB ファイルシステムのバックアップコマンド (pdfbkup) で HiRDB ファイル単位に退避したものです。データ連動回復機能で使用するファイルです。

索引

記号

/etc/hirdbds 75
/etc/hosts 91
/etc/inetd.conf 91
/etc/services 91, 94
/opt/hirdbds/lib/sample/ 504, 506

数字

1 : 1 系切り替え構成 607
1 表から n 表にデータ連動するパターン 122, 212
2 : 1 系切り替え構成 607
2 相コミット使用時のトランザクション状態の確認 850
2 相コミット使用時のトランザクションの決着 850

A

ADM 43, 65, 265
ADM の場合のマッピングキーの指定方法 226
ADT [用語解説] 1248
attr 再定義フィールド属性 (長さ) [更新情報定義] 496

B

besdef (サーバ名) 370
BES [用語解説] 1248
BLOB [用語解説] 1248
breakmode [反映環境定義] 456
breaktime [反映環境定義] 456
by 句 列データ編集 UOC 関数識別子 [反映定義] 480
by 'uoc 名' [反映定義] 486

C

check_pending [反映環境定義] 469
check [抽出定義] 419
Cluster Server 608
cm_errno_check [抽出システム定義] 393
cmtintvl [反映環境定義] 458

cmwaittime [抽出システム定義] 381
commit_wait_time [反映環境定義] 468
commit_wait_time [反映システム定義] 445
commitment_method [反映システム定義] 439
COMMIT 発行間隔 [反映環境定義] 468
COMMIT 発行間隔 [反映システム定義] 445
commondef 369, 370
COMPRESSED オプションを指定している表についての注意 591
connection_accept_hostname [抽出システム定義] 390
connection_accept_service [抽出システム定義] 391
connection_accept_waittime [抽出システム定義] 391
connection_retry_time [抽出システム定義] 391
const 初期値 [反映定義] 478
control_reference_trigger [反映環境定義] 469
control_trigger [反映環境定義] 469

D

Datareplicator 定義の予約語 1241
Datareplicator で使えるデータ型 55
Datareplicator のアンインストール [UNIX] 75
Datareplicator のアンインストール [Windows] 99
Datareplicator のインストール (UNIX) 74
Datareplicator のインストール (Windows) 95
Datareplicator の定義の概要 364
Datareplicator の定義の構成 364
Datareplicator の定義例 501
Datareplicator のバージョンダウン 1246
Datareplicator の目的 37
Datareplicator ファイルシステム領域 161
Datareplicator ファイルシステム領域の状態の表示 [コマンドの文法] 713
Datareplicator ファイルシステム領域の初期化 [コマンドの文法] 709

Datareplicator ファイルシステム領域名〔抽出環境定義〕 400
 Datareplicator ファイルシステム領域名〔反映環境定義〕 465
 db_connect_retry_interval〔反映環境定義〕 465
 db_connect_retry_number〔反映環境定義〕 464
 dblocale〔抽出システム定義〕 380
 dblocale〔反映システム定義〕 438
 DBM 52
 DBM 名称 496
 DBM 名称〔更新情報定義〕 498
 DB コネクトリトライ回数〔反映環境定義〕 464
 DB コネクトリトライ間隔〔反映環境定義〕 465
 defmerge〔反映環境定義〕 458
 defshmsize〔反映環境定義〕 460
 devicexx〔抽出環境定義〕 400
 devicexx〔反映環境定義〕 466
 discintvl〔反映環境定義〕 468
 discintvl〔反映システム定義〕 439
 disconnect 発行間隔〔反映環境定義〕 468
 disconnect 発行間隔〔反映システム定義〕 439
 divide into SQL プロセス分割数〔反映定義〕 490
 DLL ファイル 778, 811, 826
 dsidxxx〔反映システム定義〕 435
 dsid〔抽出環境定義〕 397
 DS〔用語解説〕 1248

E

ebcdic_type〔反映環境定義〕 460
 EBCDIC/KEIS 235
 EBCDIK 235
 EBCDIK/KEIS 235
 EBCDIK/KEIS から EUC への文字コードの変換規則 238
 EBCDIK/KEIS から JIS8/シフト JIS への文字コードの変換規則 237
 EBCDIK から JIS8 への文字コードの変換規則 241
 editbufsize〔送信環境定義〕 412
 errfile_unique〔抽出システム定義〕 379
 errfilesz〔抽出システム定義〕 379

errfilesz〔反映システム定義〕 437
 EUC 236
 EUC から JIS8/シフト JIS への変換規則 244
 EUC コード体系 380, 438
 event_maxentry〔反映環境定義〕 457
 eventcntreset〔送信環境定義〕 415
 eventcntreset〔反映環境定義〕 458
 eventretbl〔反映環境定義〕 457
 eventretrn〔反映環境定義〕 457
 eventspd〔反映環境定義〕 458
 eventsync〔送信環境定義〕 415
 eventtbl〔反映環境定義〕 457
 eventtrn〔反映環境定義〕 457
 except_suppress〔抽出システム定義〕 383
 except_suppress〔反映システム定義〕 439
 except 認可識別子.表識別子〔抽出定義〕 425
 ext_wait_interval〔抽出環境定義〕 398
 extdef〔抽出システム定義〕 377
 extinfunum〔抽出システム定義〕 381
 extract_all〔抽出定義〕 424
 extract_delay_limit_time〔抽出環境定義〕 399
 extract_init〔反映環境定義〕 464
 extract_level〔抽出環境定義〕 399
 extract_tselector〔送信環境定義〕 408
 extract DBM 名称.データセット名〔更新情報定義〕 498
 extract 認可識別子.表識別子〔抽出定義〕 418
 EXTSHM 84, 88
 extsuppress〔抽出環境定義〕 398
 extwatch_intvl〔抽出システム定義〕 393

F

FES〔用語解説〕 1248
 field 再定義フィールド名称〔更新情報定義〕 496
 file_dupenv〔抽出システム定義〕 389
 file_dupenv〔反映システム定義〕 445
 filetype〔更新情報定義〕 495
 FIX 属性の表〔用語解説〕 1248
 format 更新情報名〔反映定義〕 478

FREEWORD 型 (用語解説) 1248

from 更新情報名 (抽出定義) 421

from 更新情報名 (反映定義) 481

G

group 反映グループ名 (反映定義) 487

H

hash (反映定義) 489

having other (反映定義) 489

having キーレンジ分割条件文 (反映定義) 488

HA モニタ 607

HDE_BIN_COL_MAXLEN 84, 108

HDE_BIN_COL_MAXLEN についての注意 546

hde_usend.h 834

hde_usendcheck()からの戻り値 834

hde_usendcheck (送信データの判定) 827

hdechgstatus [コマンドの文法] 653

hdeevent [コマンドの文法] 654

hdefcopy [コマンドの文法] 656

hdefstate [コマンドの文法] 657

hdehost (送信環境定義) 408

hdeid (抽出システム定義) 377

hdeinfoget [コマンドの文法] 659

hdemaster 93

hdemodq [コマンドの文法] 662

hdenmserv 98

HDEPATH 83, 107

hdeprep [コマンドの文法] 665

hdeservice (送信環境定義) 407

hdeshmclean [コマンドの文法] 667

hdestart 529

hdestart_n [コマンドの文法] 682

hdestart [コマンドの文法] 672

hdestate [コマンドの文法] 687

hdestop 531

hdestop_n [コマンドの文法] 698

hdestop [コマンドの文法] 695

HDS_MST_STDCLOSE 87, 109

HDS_RFI_ELANG 87, 109

HDS_RFI_PLANG 87, 109

HDS_SHM_MODE 88

hds_ubegin() (編集開始指示関数) 780

hds_ubegin()からの戻り値 781

hds_ucoleditX()からの戻り値 818

hds_ucoleditX (列データの編集関数) 812

hds_ucommon.h 791, 819

hds_uedit() (編集・加工指示関数) 782

hds_uedit()からの戻り値 789

hds_uend() (編集終了指示関数) 790

hds_uend()関数からの戻り値 790

hds_ureflect.h 791, 819

hdscnvedt [コマンドの文法] 701

hdschgstatus [コマンドの文法] 706

HDSCLTWAITTIME 87, 109

hdsemergencystop [コマンドの文法] 707

hdsfcopy [コマンドの文法] 708

hdsfmkfs [コマンドの文法] 709

hdsfstate [コマンドの文法] 711

hdsfstatfs [コマンドの文法] 713

hdsid (反映システム定義) 434

hdsinfoget [コマンドの文法] 716

HDSPATH 87, 109

hdsrefinfm [コマンドの文法] 719

hdsrftcl [コマンドの文法] 724

hdssamqin [コマンドの文法] 727

hdsservice (反映システム定義) 436

hdsshmclean [コマンドの文法] 729

hdsstart 572

hdsstart [コマンドの文法] 732

hdsstate [コマンドの文法] 738

hdsstop 574

hdsstop [コマンドの文法] 747

hdstrcredit [コマンドの文法] 751

HiRDB 43, 62

hirdb_audit_trail (抽出システム定義) 392

hirdb_audit_trail (反映システム定義) 436

HiRDB/Parallel Server 71

HiRDB/Single Server 71
 HiRDB Dataextractor 38
 HiRDB Datareplicator 37
 HiRDB Datareplicator 連携機能 361
 HiRDB Datareplicator 連携の開始 [コマンドの文法] 769
 HiRDB Datareplicator 連携の開始指定 361
 HiRDB Datareplicator 連携の開始・終了・中止 548
 HiRDB Datareplicator 連携の実行状況の確認 556
 HiRDB Datareplicator 連携の中止 548
 HiRDB Datareplicator 連携の中止 [コマンドの文法] 770
 hirdbds/ 76
 HiRDB Text Search Plug-in 71
 hirdbusr [反映システム定義] 434
 HiRDB XML Extension 71
 HiRDB から HiRDB ヘデータ連動する場合の設計 256
 HiRDB から HiRDB ヘデータ連動する場合のソフトウェア構成 62
 HiRDB から HiRDB へのデータ連動 43
 HiRDB からメインフレーム側 DB ヘデータ連動する場合の設計 261
 HiRDB からメインフレーム側 DB ヘデータ連動する場合のソフトウェア構成 64
 HiRDB からメインフレーム側 DB へのデータ連動 45
 HiRDB システムの状態の表示 [コマンドの文法] 767
 HiRDB システムログファイル 126
 HiRDB 接続認可識別子 [反映システム定義] 434
 HiRDB との対応状況 1231
 HiRDB 認可識別子 260
 HiRDB の機能と Datareplicator でのサポート状況 1233
 HiRDB の適用 OS と使用できる Datareplicator のバージョン 1231

I

info_message_out [抽出システム定義] 383
 info_message_out [反映システム定義] 439
 int_trc_filesz [抽出システム定義] 385
 int_trc_filesz [反映システム定義] 442

int_trc_getl [反映環境定義] 465
 int_trc_getv [送信環境定義] 413
 int_trc_getv [抽出環境定義] 399
 int_trc_lvl [抽出システム定義] 383
 int_trc_lvl [反映システム定義] 440
 int_trc_output_path [抽出システム定義] 387
 int_trc_output_path [反映システム定義] 444
 int_trc_rintvl [抽出システム定義] 385
 int_trc_rintvl [反映システム定義] 442
 int_trc_unit [抽出システム定義] 385
 int_trc_unit [反映システム定義] 442
 IP アドレス 91

J

JIS8/シフト JIS 235
 JIS8/シフト JIS コード体系 380, 438
 JIS8/シフト JIS 又は EUC から、EBCDIK/KEIS への文字コードの変換規則 242
 JIS8 から EBCDIK への文字コードの変換規則 241

K

keepalive [送信環境定義] 410
 keepalive [反映システム定義] 437
 key フィールド名称 | 再定義フィールド名称 [更新情報定義] 498
 key 列名 [抽出定義] 419

L

LANG 83, 86
 LDR_CNTRL 84, 88
 load [反映定義] 481
 LOB データ [用語解説] 1248
 logiosize [抽出環境定義] 398
 logmrg コマンド [データ連動回復機能のコマンド] 888

M

mapping_key_check [反映環境定義] 468
 maxtrandata [送信環境定義] 412
 maxtran [送信環境定義] 412

MGR [用語解説] 1248
msglocale [抽出システム定義] 381
msglocale [反映システム定義] 438
mstservice [抽出システム定義] 381

N

name フィールド名称 [反映定義] 478
nocodecnv [反映定義] 480
node_connection_accept [抽出システム定義] 389
node_host [抽出システム定義] 394
node_pdconfpath [抽出システム定義] 394
node_pddir [抽出システム定義] 393
node_shlibpath [抽出システム定義] 394
node_syslogout [抽出システム定義] 391
nodecontrol [抽出システム定義] 389
nodedef (ホスト名) 369
nodemst 方式 131
NODISCLAIM 84, 88
none [抽出定義] 420
none [反映定義] 483
not_null_unique [抽出定義] 419
not_null_unique [反映定義] 483
nsap_address [送信環境定義] 409
NSAP アドレス [送信環境定義] 409
nsnidxxx [送信環境定義] 410
n表から1表にデータ連動するパターン 122, 215

O

OSI プロトコル 94, 267
overwrite_continue [送信環境定義] 411
overwrite [送信環境定義] 411

P

PATH 83, 86, 107, 108
pd_log_rpl_no_standby_file_opr [HiRDB の定義] 428
pd_rpl_func_control 60
pd_rpl_hdepath [HiRDB の定義] 428
pd_rpl_init_start [HiRDB の定義] 427

PDCONFPATH 83, 107
pddef 222
PDDIR 83, 86, 107, 109
PDHOST 83, 86, 107, 109
PDLANG 83, 86
pdload 222
pdlogchg [コマンドの文法] 766
pdls [コマンドの文法] 767
PDMII E2 43, 65, 67, 265
PDMII E2 の場合のマッピングキー 226
PDMII E2 の場合のマッピングキーの指定方法 226
PDMII E2 ログ形式 SAM ファイル 494
pdmod 223
PDNAMEPORT 83, 86, 107, 109
pdrbal 223
pdrorg 223
pdrplstart [コマンドの文法] 769
pdrplstop [コマンドの文法] 770
pdrstr 223
PDUSER 83, 107
position データ開始位置 [更新情報定義] 496
prg_eventno [送信環境定義] 413
protocol1 [反映システム定義] 435
protocol2 [反映システム定義] 435
protocol [送信環境定義] 408
PSALLOC 84, 88

Q

queue_read_wait_interval [送信環境定義] 413
queuesize [抽出環境定義] 397
queuesize [反映環境定義] 454
qufilexxx [抽出環境定義] 397
qufilexxx [反映環境定義] 454
qufullwarn [抽出環境定義] 399
quiosize [抽出環境定義] 398

R

RDB1 E2 43, 67
RDB1 E2 の場合のマッピングキー 227

RD ノード [用語解説] 1248
readbufnum [送信環境定義] 412
recover_info_send_interval [抽出システム定義]
392
recover_info_send [抽出システム定義] 392
recvwatchtime [送信環境定義] 413
ref_data_backspace [反映環境定義] 461
ref_wait_interval [反映環境定義] 468
ref_wait_interval [反映システム定義] 444
refenvxxx [反映システム定義] 436
reffile [反映環境定義] 454
reflect_counter_reset [反映システム定義] 447
reflect_delay_limit_time [反映環境定義] 470
reflect_mode [送信環境定義] 414
reflect_trn_max_sqlnum [反映環境定義] 472
reflect_tselector [反映システム定義] 436
restartmode [反映環境定義] 456
restruct DBM 名称.データセット名 [更新情報定
義] 496
retry_interval [送信環境定義] 411
retrynum [送信環境定義] 411

S

SAM ファイル 47
SAM ファイルからの更新情報の抽出 [コマンドの文
法] 727
SAM ファイルの運用方法 588
SAM ファイル [反映側 Datareplicator で使うファイ
ルの作成] 323
SAM ファイル [反映側 Datareplicator で使うファイ
ルの設定] 326
SAM ファイル [反映側 Datareplicator のディスク資
源の一覧] 345
SAM ファイル [反映処理で使うファイルとプロセス]
144
SAM ファイルを使った, メインフレーム側 DB から
HiRDB へのデータ連動 47
SAM ファイルを使って, メインフレーム側 DB から
HiRDB ヘデータ連動する場合の運用手順 522
SAM ファイルを使って, メインフレーム側 DB から
HiRDB ヘデータ連動する場合の設計 267

SAM ファイルを使って, メインフレーム側 DB から
HiRDB ヘデータ連動する場合のソフトウェア構成 67
send_counter_reset [抽出システム定義] 392
send_delay_limit_time [送信環境定義] 414
sendcontrol [抽出システム定義] 377
senddefxxxx [抽出システム定義] 379
senddefxx [抽出システム定義] 378
sendhdsid [送信環境定義] 407
sendidxxxx [抽出システム定義] 378
sendidxx [抽出システム定義] 378
sendintvl_scale [送信環境定義] 409
sendintvl [送信環境定義] 409
sendmst 方式 131
sendprocnum [抽出システム定義] 387
senduoc [送信環境定義] 409
send 送信先識別子 [抽出定義] 421
services ファイル 98, 112
services ファイルに指定する内容 112
set_tool1 コマンド [データ連動回復機能のコマン
ド] 890
set_tool2 コマンド [データ連動回復機能のコマン
ド] 891
setup_tool1 コマンド [データ連動回復機能のコマン
ド] 890
setup_tool2 コマンド [データ連動回復機能のコマン
ド] 891
set 形式の定義の規則 368
SGMLTEXT 型 [用語解説] 1248
shiftspace_cnv [反映環境定義] 461
SHLIB_PATH 83, 87
skip_codecnv_error [反映環境定義] 465
skip_mvcelmwarn [反映環境定義] 463
skip_sqlcode [反映環境定義] 462
SMALLFLT, FLOAT 型の抽出列 227
smt_editbufsize [抽出システム定義] 388
smt_queue_read_wait_interval [抽出システム定
義] 388
smt_readbufnum [抽出システム定義] 388
smt_sendintvl_scale [抽出システム定義] 388
smt_sendintvl [抽出システム定義] 388

SQL 61
sql_lockerr_retrinum [反映環境定義] 470
sqlconvopt1 [反映定義] 482
sqlconvopt2 [反映定義] 482
sqlerr_during_retry [反映環境定義] 463
sqlerr_skip_info [反映環境定義] 464
SQL エラーのスキップ 155
SQL 通番を取得する場合の注意事項 254
startmode [反映環境定義] 455
statsfile [反映環境定義] 455
statssize [反映環境定義] 455
syncgroup001 [反映システム定義] 446
syncgrp_discintvl [反映システム定義] 446
syncterm [抽出システム定義] 381
syncwait_limit_time [反映システム定義] 447
syncwait_limit_tran_count [反映システム定義] 446
syslog_message_suppress [抽出システム定義] 380
syslog_message_suppress [反映システム定義] 437
syslogout [抽出システム定義] 380
syslogout [反映システム定義] 437

T

tblcheck [反映環境定義] 460
tblcmtintvl [反映環境定義] 459
TCP/IP [用語解説] 1249
Terminal Service の使用 113
termlevel [抽出システム定義] 382
timestamp [反映定義] 482
TMS-4V/SP 43, 65, 265
TMS-4V/SP の場合のマッピングキーの指定方法 227
to 更新情報名 [更新情報定義] 498
to 更新情報名 [抽出定義] 419
trncmtintvl [反映環境定義] 459
TZ 83, 86, 107, 109
T セレクタ 267
T セレクタ [送信環境定義] 408
T セレクタ値 94

T セレクタ [反映システム定義] 436

U

UAC 115
UAP 実行時の注意 221
UAP [用語解説] 1249
UCE 名 94
UCS2 236
ujcodekind [反映環境定義] 467
ukey 列名 [抽出定義] 419
undefcode_cnv [反映環境定義] 461
unique [抽出定義] 420
unique [反映定義] 483
UNIX 版 Datareplicator と Windows 版 Datareplicator の機能差 1245
unreffile1 [反映環境定義] 455
unreffile2 [反映環境定義] 455
unreffilesz [反映環境定義] 455
unset_tool コマンド [データ連動回復機能のコマンド] 891
unsetup_tool コマンド [データ連動回復機能のコマンド] 891
UOC 134, 152
use_convertlib [反映システム定義] 444
user1 93
User Account Control 115
UTF-8 236
utf-8 コード体系 380
UTF-8 文字コード体系 438

W

watchintvl [抽出システム定義] 381
where 句 [抽出定義] 421
Windows Vista, 及び Windows Server 2008 の場合の動作環境 115
WITHOUT ROLLBACK オプションを指定している表についての注意 589

X

xa_recovery_retry_count [反映環境定義] 471

xa_recovery_retry_interval [反映環境定義] 471
XDM/DS 46
XDM/DS からデータ連動機能を使う場合のループバック抑止について 277
XDM/DS と反映側 Datareplicator との対応 266
XDM/DS と反映側 DB との対応 265
XDM/RD E2 64, 65, 261, 265
XDM/RD E2 [データ連動システムの組み合わせ] 43
XDM/RD E2 の場合のマッピングキーの指定方法 226
XDM/SD E2 43, 65, 265
XDM/SD E2 の場合のマッピングキーの指定方法 226
XDM/XT 38
XID 850
XML 型 [用語解説] 1249
XNF/H/BASE 71
XNF のネットワーク定義 94

あ

アイコン [Windows] 98
相手システム識別子 [送信環境定義] 407
アソシエーション [用語解説] 1249
アドバンストキュー [用語解説] 1249
アプリケーション [用語解説] 1249
アンインストール [UNIX] 75
アンインストール [Windows] 99
アンロード統計ログファイル [用語解説] 1249
アンロードログファイルによるデータ連動回復 869
アンロードログファイル [用語解説] 1249

い

イベント開始方式 338, 578
イベント機能 189
イベントコード 190
イベントコード [hdeevent コマンドの文法] 654
イベントコードの指定範囲 295
イベントコードの定義 190
イベント終了 575
イベント制御用テーブルの構成 296
イベント制御用テーブルの条件 296

イベント制御用テーブルの設計 295
イベント制御用テーブルの例 296
イベント同時管理数 [反映環境定義] 457
イベントの発行 189
イベント番号 [送信環境定義] 413
インストール時に登録されるサービス [Windows] 97
インストールするサーバマシン [UNIX] 74
インストールするサーバマシン [Windows] 95
インストールの実行 [UNIX] 75
インストールの実行 [Windows] 96
インストール前の準備 [UNIX] 74
インストール前の準備 [Windows] 95
インタフェースブロック名 780, 782, 790
インタフェース領域 [用語解説] 1249
インデクス [用語解説] 1249
インナレプリカ機能を使った RD エリアのデータ連動 137
インナレプリカのデータ連動の定義 137

う

埋込み型 UAP [用語解説] 1250
運用の概要 520

え

影響分散スタンバイレス型系切り替え機能での運用 622
英文メッセージテキストと和文メッセージテキストの切り替え 914
エラー監視間隔 [抽出システム定義] 381
エラー情報ファイル 127, 143
エラー情報ファイルの運用方法 [抽出側 Datareplicator] 542
エラー情報ファイルの運用方法 [反映側 Datareplicator] 583
エラー対処後の処置 [抽出側 Datareplicator] 843
エラー対処後の処置 [反映側 Datareplicator] 851
エラーの対処方法 [抽出側 Datareplicator] 840
エラーの対処方法 [反映側 Datareplicator] 848
エントリ 91

か

- 開始モード [抽出側 Datareplicator] 530
- 開始モード [反映側 Datareplicator] 573
- 階層型システムへの応用形態の設計 271
- 回復情報ファイル 323, 326, 345
- 稼働環境定義文 [更新情報定義] 495
- 稼働統計解析ユティリティ (PDMJANL) 144
- 稼働トレース収集プロセス 131, 146
- 稼働トレース情報取得間隔 [抽出システム定義] 385
- 稼働トレース情報取得間隔 [反映システム定義] 442
- 稼働トレースの取得 [抽出側 Datareplicator] 133
- 稼働トレースの取得 [反映側 Datareplicator] 151
- 稼働トレースファイル 127, 143, 323, 325
- 稼働トレースファイルの運用方法 [抽出側 Datareplicator] 544
- 稼働トレースファイルの運用方法 [反映側 Datareplicator] 585
- 稼働トレースファイルの編集 [コマンドの文法] 751
- 環境構築時の初期化手順 525
- 環境の設定 70
- 環境変数の設定 (UNIX) 83
- 環境変数の設定 (Windows) 107
- 関連する製品 71

き

- キー情報ブロック名 784
- キーレンジ単位分割方式 148, 149, 327
- キーレンジ分割条件文 [反映定義] 488
- キーレンジ分割 [用語解説] 1250
- キーレンジ分割を指定するときの規則 491
- 起動方法 [抽出側 Datareplicator] 529
- 起動方法 [反映側 Datareplicator] 572
- 行 52
- 強制終了 532, 575
- 共通定義部 369, 370

<

- 空白文字の変換規則 238, 239
- クライアント／サーバ [用語解説] 1250

- クラスタキー [用語解説] 1250
- 繰返し列 57
- 繰返し列の設計 250
- 繰返し列 [用語解説] 1250
- グローバルバッファプール [用語解説] 1250
- グローバルバッファ [用語解説] 1250

け

- 計画系切り替え 607
- 系切り替え機能を使うときの運用 605
- 系切り替え機能を使うときの準備 [Amazon EC2 環境の HA モニタの場合] 631
- 現用ファイルの複写 [コマンドの文法] 656, 708

こ

- 更新 SQL 出力機能 901
- 更新 SQL ファイル 902
- 更新情報 43, 49
- 更新情報定義 [定義の文法] 494
- 更新情報定義ファイル 144, 322, 325, 345
- 更新情報定義ファイルの運用方法 587
- 更新情報定義ファイル名 [hdssamqin コマンドの文法] 727
- 更新情報データセット 45, 46, 48
- 更新情報内容の確認 [コマンドの文法] 719
- 更新情報入力プロセス 146
- 更新情報の履歴を時系列順に取得するパターン 124, 219
- 更新情報フィールド定義 477
- 更新情報編集バッファ 289
- 更新情報編集バッファサイズ [送信環境定義] 412
- 更新情報編集バッファサイズの指定 290
- 更新情報編集バッファ長 [抽出システム定義] 388
- 更新情報名 [更新情報定義] 498
- 更新情報名 [反映定義] 481
- 更新情報 [用語解説] 1250
- 更新情報読み込み用 I/O バッファ数 [抽出システム定義] 388
- 構成変更 592
- 構成要素 52

構造型 50
 構造型データベース 50
 高速系切り替え機能を使う場合の Datareplicator の
 定義方法 615
 構文形式の定義の規則 368
 コネクションリトライ回数 290
 コネクションリトライ回数〔送信環境定義〕 411
 コネクションリトライ時のリトライ間隔〔送信環境定
 義〕 411
 個別定義部 369, 370
 コマンド開始方式 338, 578
 コマンドの一覧 648
 コマンドログファイル 323, 326, 345
 コマンドログファイル〔抽出側 Datareplicator で使
 うファイルの作成〕 281
 コマンドログファイル〔抽出側 Datareplicator で使
 うファイルの設定〕 285
 コマンドログファイル〔抽出側 Datareplicator の
 ディスク資源の一覧〕 300
 コマンドログファイル〔抽出処理で使うファイル〕
 128
 コマンドログファイルの運用方法〔抽出側
 Datareplicator〕 545
 コマンドログファイルの運用方法〔反映側
 Datareplicator〕 587
 コマンドログファイル〔反映処理で使うファイル〕
 143

さ

サーバマシンごとの定義 369
 サーバ〔用語解説〕 1251
 サービス〔Windows〕 97
 サービス名 91, 94, 264, 267
 サービス名〔HiRDB から HiRDB へデータ連動する場
 合の設計〕 260
 サービス名〔送信環境定義〕 407
 サービス名の登録〔Windows〕 112
 サービス名〔反映システム定義〕 436
 再開始 531, 574
 最大更新情報名数〔抽出システム定義〕 381
 再定義フィールド属性〔更新情報定義〕 496

再定義フィールドに指定できるフィールド属性と長さ
 496
 再定義フィールド名称 496
 作業表用ファイル〔用語解説〕 1251

し

時間指定開始方式 338, 577
 識別子の定義例 261
 時系列情報の取得 151
 時系列情報表の作成 252
 時系列情報を取得する表の定義 253
 システム共通定義〔HiRDB の定義〕 427
 システムコールエラー一覧 1222
 システム設計 200
 システムの対応とファイルの構成例 501
 システムファイル〔用語解説〕 1251
 システムマネージャ〔用語解説〕 1251
 システム用 RD エリア〔用語解説〕 1251
 システムログ I/O バッファ 285, 286
 システムログ I/O バッファサイズ〔抽出環境定義〕
 398
 システムログファイル 43
 システムログファイルによるデータ連動回復 860
 システムログファイルの運用 549
 システムログファイルの操作 553
 システムログファイル〔用語解説〕 1251
 自動系切り替え 607
 終了方法〔抽出側 Datareplicator〕 531
 終了方法〔反映側 Datareplicator〕 574
 終了モード〔抽出側 Datareplicator〕 532
 終了モード〔反映側 Datareplicator〕 575
 受信 141
 受信プロセス 145
 取得できる時系列情報 252
 順序数生成子 254
 障害回復時の初期化手順 857
 障害時の対処 837
 障害の対処の手順〔抽出側 Datareplicator〕 838
 障害の対処の手順〔反映側 Datareplicator〕 847

状態監視サービス 97
初期化 525, 592
初期開始 531, 573
シンクポイントダンプファイル [用語解説] 1251

す

スーパユーザ [用語解説] 1251
スキーマ 52
ステータス情報の取得 [抽出側 Datareplicator] 132
ステータス情報の取得 [反映側 Datareplicator] 151
ステータスファイル 127, 143
ステータスファイルの運用方法 [抽出側 Datareplicator] 540
ステータスファイルの運用方法 [反映側 Datareplicator] 583
ステータスファイル [用語解説] 1252
スワップ 142

せ

正常開始 531, 574
正常終了 532, 575
セグメント 52
セグメントオカレンス 52
接続単位終了イベント 190
選択条件に指定できる定数 422
選択条件列属性と定数との関連 422
全表抽出定義文 424

そ

相互系切り替え 607, 619
相互系切り替え構成 607
送信 126, 132
送信間隔 290
送信環境定義 [定義の文法] 404
送信環境定義の単位の設計 292
送信環境定義の例 505
送信環境定義ファイル 128, 280, 284, 299
送信環境定義ファイルの運用方法 537
送信環境定義ファイル名 [抽出システム定義] 378
送信行を選択してデータ連動するパターン 123, 217

送信先識別子 259
送信先識別子 [hdeevent コマンドの文法] 654
送信先識別子 [hdestart コマンドの文法] 673
送信先識別子 [hdestate コマンドの文法] 687
送信先識別子 [hdestop コマンドの文法] 695
送信先識別子 [抽出システム定義] 377
送信先識別子 [抽出定義] 421
送信先ホスト名 [送信環境定義] 408
送信先を追加する 537
送信処理の縮退の設計 290
送信処理の送信間隔の設計 290
送信滞留時間 168
送信遅延開始 293, 534
送信データ UOC 134, 822
送信データ UOC の作成手順 (UNIX) 823
送信データ UOC の作成手順 (Windows) 826
送信データ UOC のサンプル 836
送信データ UOC を作成するときの注意 834
送信データ件数リセットイベント 190
送信プロセス 131
送信プロセス最大起動数 [抽出システム定義] 387
送信プロセス数の制御 133
送信プロセスの抽出情報キューファイル読み込み間隔 [送信環境定義] 413
送信プロセスの抽出情報キューファイル読み込み間隔 [抽出システム定義] 388
送信文の指定と送信される更新情報 420
送信方法の設計 287
送信マスタプロセス 131, 133
送信マスタプロセス送信間隔 [抽出システム定義] 388
送信抑止受信元識別子 [送信環境定義] 410
双方向更新システム 270
双方向更新システムの階層型システムへの応用形態 272
即時終了 575
ソフトウェア構成 62

た

第3水準・第4水準の文字 116

滞留監視機能 168
滞留時間 168
単一方向更新システム 270
単一方向更新システムの階層型システムへの応用形態
271
単独停止 342

ち

遅延開始 338
チャンネル接続 267
抽出 125, 132
抽出側 Datareplicator 39
抽出側 Datareplicator 運用ディレクトリ名〔HiRDB
の定義〕 428
抽出側 Datareplicator が使うディレクトリ名の指定
361
抽出側 Datareplicator 識別子 259
抽出側 Datareplicator 識別子〔抽出システム定義〕
377
抽出側 Datareplicator で使うファイルの運用 537
抽出側 Datareplicator でのイベント発行〔コマンド
の文法〕 654
抽出側 Datareplicator と XDM/DS との対応 263
抽出側 Datareplicator と反映側 Datareplicator と
の対応 258
抽出側 Datareplicator の運用 533
抽出側 Datareplicator の環境変数〔UNIX〕 83
抽出側 Datareplicator の環境変数〔Windows〕
107
抽出側 Datareplicator の起動〔コマンドの文法〕
672
抽出側 Datareplicator の起動と終了 529
抽出側 Datareplicator の共用リソースの削除〔コマ
ンドの文法〕 667
抽出側 Datareplicator の終了〔コマンドの文法〕
695
抽出側 Datareplicator の障害時の対処 838
抽出側 Datareplicator の状態変更〔コマンドの文法〕
653
抽出側 Datareplicator のステータス情報の取得〔コ
マンドの文法〕 687

抽出側 Datareplicator の設計 278
抽出側 Datareplicator の抽出処理 125
抽出側 Datareplicator の定義の一覧 365
抽出側 Datareplicator の定義のテンプレート 503
抽出側 Datareplicator の定義の例 503
抽出側 Datareplicator のディレクトリ構成〔UNIX〕
77
抽出側 Datareplicator のディレクトリ構成
〔Windows〕 101
抽出側 Datareplicator を使うために必要な HiRDB
の定義 427
抽出側 DB 39
抽出側 DB が ADM の場合のデータ型の対応 232
抽出側 DB が ADM の場合のマッピングキー 226
抽出側 DB が HiRDB の場合のデータ型の対応 228
抽出側 DB が HiRDB の場合のマッピングキー 224
抽出側 DB が PDMII E2 の場合のデータ型の対応 233
抽出側 DB が PDMII E2 の場合のマッピングキー 226
抽出側 DB が RDB1 E2 の場合のデータ型の対応 234
抽出側 DB が RDB1 E2 の場合のマッピングキー 227
抽出側 DB が TMS-4V/SP の場合のデータ型の対応
234
抽出側 DB が TMS-4V/SP の場合のマッピングキー
227
抽出側 DB が XDM/RD E2 の場合のデータ型の対応
231
抽出側 DB が XDM/RD E2 の場合のマッピングキー
226
抽出側 DB が XDM/SD E2 の場合のデータ型の対応
230
抽出側 DB が XDM/SD E2 の場合のマッピングキー
226
抽出側 DB と XDM/DS との対応 266
抽出側 DB と反映側 DB との対応 49
抽出側 DB と反映側 DB との対応の設計 221
抽出側 DB と反映側 DB の文字コードの対応 236
抽出側 DB の条件 221
抽出側 HiRDB 39
抽出側 HiRDB と抽出側 Datareplicator との対応 257
抽出側 HiRDB の運用 548

抽出側 HiRDB の運用手順 557

抽出側 HiRDB の設計 361

抽出側 HiRDB の定義 427

抽出側システム 39

抽出側システムでエラーが発生したときの対処 852

抽出側システムとの通信で使うサービス名 113

抽出側システムと反映側システムとの対応の設計 256

抽出側システムと反映側システムの形態 63, 65, 66, 68

抽出側システムと反映側システムの対応 62, 64, 66, 67

抽出側障害情報の取得〔コマンドの文法〕 659

抽出環境定義〔定義の文法〕 395

抽出環境定義の単位の設計 287

抽出環境定義の例 504

抽出環境定義ファイル 127, 280, 283, 299

抽出環境定義ファイルの運用方法 537

抽出環境定義ファイル名〔抽出システム定義〕 377

抽出完了状態 550

抽出コマンドプロセス 130

抽出サーバステータスファイル 127, 281, 284, 300

抽出サービス 97

抽出再定義文〔更新情報定義〕 495

抽出システム定義〔定義の文法〕 372

抽出システム定義の例 504

抽出システム定義ファイル 127, 280, 283, 299

抽出システム定義ファイルの運用方法 537

抽出終端検知後の抽出再開開始間隔〔抽出環境定義〕 398

抽出状態 550

抽出情報キュー I/O バッファサイズ〔抽出環境定義〕 398

抽出情報キュー I/O バッファ〔送信用〕 289

抽出情報キュー I/O バッファ〔抽出用〕 286

抽出情報キューファイル 126, 281, 284, 299

抽出情報キューファイル回復機能 895

抽出情報キューファイル回復機能の回復手順 896

抽出情報キューファイル回復機能の前提条件 896

抽出情報キューファイルサイズ〔抽出環境定義〕 397

抽出情報キューファイル〔データ連動の仕組み〕 43, 45

抽出情報キューファイルの運用方法 538

抽出情報キューファイルの構成変更〔コマンドの文法〕 662

抽出情報キューファイルへのデータの格納方法 126

抽出情報キューファイル名〔抽出環境定義〕 397

抽出処理 43, 125

抽出処理監視プロセス 131

抽出処理で使うファイル 126

抽出処理の運用 533

抽出処理のエラー監視間隔の設計 287

抽出処理の開始方式の設計 292

抽出処理の単位 131

抽出処理の停止方式の設計 294

抽出処理のプロセス構成 128

抽出対象外データ格納ファイル 144, 323, 326, 345

抽出対象外データ格納ファイルの運用方法 588

抽出対象の SQL 221

抽出対象の UAP 221

抽出対象のデータ型 221

抽出対象の表 221

抽出対象のユティリティ 222

抽出滞留時間 168

抽出遅延開始 293, 535

抽出定義 49

抽出定義〔定義の文法〕 416

抽出定義と反映定義の例 508

抽出定義ファイル 128, 280, 284, 299

抽出定義ファイルの運用方法 537

抽出定義ファイルの再変換 538

抽出定義ファイル名〔hdeprep コマンドの文法〕 665

抽出定義プリプロセスファイル 128, 281, 284, 299

抽出定義プリプロセスファイルの運用方法 537

抽出定義プリプロセスファイルの作成〔コマンドの文法〕 665

抽出定義プリプロセスファイルの有効性チェック 128

抽出ノードマスタエラー情報ファイル 127, 281, 284, 300

抽出ノードマスタトレースファイル 127, 281, 284, 300
抽出ノードマスタプロセス 131
抽出プロセス 131
抽出文〔更新情報定義〕 497
抽出方法の設計 285
抽出マスタエラー情報ファイル 127, 281, 284, 300
抽出マスタステータスファイル 127, 281, 284, 300
抽出マスタトレースファイル 127, 281, 284, 300
抽出マスタプロセス 131
抽出マスタプロセスと抽出ノードマスタプロセスとの通信で使うサービス名 112
抽出未完了状態 551
抽象データ型 56
抽象データ型〔用語解説〕 1252
チューニング 643

つ

通常ファイル〔用語解説〕 1252
通信環境の設定 (UNIX) 91
通信環境の設定 (Windows) 112
通信プロトコル 267
通信待ち時間〔抽出システム定義〕 381

て

定義内容の変更〔送信環境定義〕 405
定義内容の変更〔抽出環境定義〕 396
定義内容の変更〔抽出システム定義〕 374
定義内容の変更〔抽出定義〕 417
定義内容の変更〔反映環境定義〕 451
定義内容の変更〔反映システム定義〕 432
定義内容の変更〔反映定義〕 477
定義の規則 367
定義の体系 364
定義例 501
ディクショナリサーバ〔用語解説〕 1252
停止イベント 190
ディスク資源の設計 299, 344
ディレクトリ構成 (UNIX) 76

ディレクトリ構成 (Windows) 100
データウェアハウス〔用語解説〕 1252
データ型 55
データセット 52
データセット名 496
データセット名〔更新情報定義〕 498
データ操作言語〔用語解説〕 1252
データ定義言語〔用語解説〕 1253
データディクショナリ LOB 用 RD エリア〔用語解説〕 1253
データディクショナリ〔用語解説〕 1253
データベース 52
データベース回復ユティリティ 223
データベース構成変更ユティリティ 223
データベース再編成ユティリティ 223
データベース作成ユティリティ 222
データベース抽出・反映サービス機能 37
データベース定義ユティリティ 222
データベースの種類 50
データベースの種類とデータ連動の関係 50
データベース用語の対応 52
データ連動 37
データ連動回復機能 860
データ連動回復機能 1 (トランザクション検索フェーズ) 878
データ連動回復機能 2 (抽出キュー作成フェーズ) 878
データ連動回復機能のコマンド 887
データ連動回復機能を実行した後の作業 892
データ連動機能 37, 119
データ連動識別子 259
データ連動識別子〔hdsrfctl コマンドの文法〕 724
データ連動識別子〔hdssamqin コマンドの文法〕 727
データ連動識別子〔hdsstate コマンドの文法〕 738
データ連動識別子〔抽出環境定義〕 397
データ連動識別子〔反映システム定義〕 435
データ連動システム 37, 39
データ連動システムの運用手順 520
データ連動システムの組み合わせ 43

データ連動システムの形態の種類 270
データ連動システムの形態の設計 270
データ連動システムの構築手順 69
データ連動システムのソフトウェア構成 62
データ連動システムの用語 38
データ連動システムの利点 38
データ連動できるシステム 39
データ連動できるデータベース 49
データ連動で使える表 53
データ連動について 43
データ連動の仕組み 43
データ連動用連絡ファイル 127, 281, 285, 300, 549
データ連動用連絡ファイルの運用方法 544
データ連動を適用できる業務システム 39
デッドロック [用語解説] 1253
テンプレート 503, 506

と

同一形式の表にデータ連動するパターン 120, 204
同期イベント 190
同期管理プロセス 146
同期点処理 158
同期点 [用語解説] 1253
同期反映グループ 180
同期反映グループ名 [hdsstate コマンドの文法] 738
統計ログファイル [用語解説] 1253
同時開始 293, 337, 533, 576
同時停止 294, 340
トランザクション管理情報バッファ 288
トランザクション単位反映イベント 190
トランザクション単位反映イベントコード [反映環境定義] 457
トランザクション単位反映再起動イベント 190
トランザクション単位反映再起動イベントコード [反映環境定義] 457
トランザクション単位反映方式 146, 327
トランザクションの決着 850
トランザクションブランチ情報 179
トランザクション [用語解説] 1253

トリガが定義されている表のデータ連動 150
トリガ [用語解説] 1253

な

ナル値 [用語解説] 1253

に

二重化制御ファイル 281, 285, 323, 326
二重化定義 [定義の文法] 430, 500
二重化定義ファイル [抽出側 Datareplicator で使うファイルの作成] 280
二重化定義ファイル [抽出側 Datareplicator で使うファイルの設定] 284
二重化定義ファイル [抽出側 Datareplicator のディスク資源の一覧] 299
二重化定義ファイル [反映側 Datareplicator で使うファイルの作成] 322
二重化定義ファイル [反映側 Datareplicator で使うファイルの設定] 325
二重化定義ファイル [反映側 Datareplicator のディスク資源の一覧] 344
二重化ファイルの状態表示 [コマンドの文法] 657, 711
日本語漢字コード 236
認可識別子 [抽出定義] 418
認可識別子 [反映定義] 483, 487

の

ノードマスタプロセス起動サービス 97
ノード [用語解説] 1254

は

バージョンアップ 117
パスワード 260
パスワード [反映システム定義] 434
バックエンドサーバごとの定義 370
バックエンドサーバ [用語解説] 1254
ハッシュ分割方式 148, 328
バッファの設計 285, 287
反映 141

- 反映 SQL プロセス 146
- 反映 UOC プロセス 146
- 反映エラー情報ファイル 323, 325, 345
- 反映エラー情報ファイルサイズ [反映システム定義] 437
- 反映側 Datareplicator 39
- 反映側 Datareplicator 識別子 260
- 反映側 Datareplicator 識別子 [反映システム定義] 434
- 反映側 Datareplicator で使うファイルの運用 581
- 反映側 Datareplicator での/etc/services の設定例 94
- 反映側 Datareplicator での文字コードの変換 236
- 反映側 Datareplicator と反映側 HiRDB との対応 260
- 反映側 Datareplicator の環境変数 [UNIX] 86
- 反映側 Datareplicator の環境変数 [Windows] 108
- 反映側 Datareplicator の起動 [コマンドの文法] 732
- 反映側 Datareplicator の起動と終了 572
- 反映側 Datareplicator の強制終了 [コマンドの文法] 707
- 反映側 Datareplicator の共用リソースの削除 [コマンドの文法] 729
- 反映側 Datareplicator の終了 [コマンドの文法] 747
- 反映側 Datareplicator の障害時の対処 847
- 反映側 Datareplicator の状態変更 [コマンドの文法] 706
- 反映側 Datareplicator のステータス情報の取得 [コマンドの文法] 738
- 反映側 Datareplicator の設計 321
- 反映側 Datareplicator の通信環境の設定例 94
- 反映側 Datareplicator の定義の一覧 365
- 反映側 Datareplicator の定義のテンプレート 506
- 反映側 Datareplicator の定義の例 506
- 反映側 Datareplicator のディレクトリ構成 [UNIX] 80
- 反映側 Datareplicator のディレクトリ構成 [Windows] 104
- 反映側 Datareplicator の反映処理 141
- 反映側 DB 39
- 反映側 HiRDB 39
- 反映側システム 39
- 反映側システムでエラーが発生したときの対処 844
- 反映側システムで表単位反映方式で反映処理する場合の反映抑止方法 892
- 反映側システムとの通信で使うサービス名 112
- 反映側システムとの通信で使うホスト名 113
- 反映側障害情報の取得 [コマンドの文法] 716
- 反映環境定義 [定義の文法] 449
- 反映環境定義の例 507, 508
- 反映環境定義ファイル 143, 322, 325, 344
- 反映環境定義ファイルの運用方法 582
- 反映環境定義ファイル名 [反映システム定義] 436
- 反映グループ 147
- 反映グループ定義 486
- 反映グループ定義による反映グループ分け 491
- 反映グループ名 [反映定義] 487
- 反映コマンドプロセス 145
- 反映サービス 97
- 反映時エラーのスキップ 152
- 反映システム定義 [定義の文法] 431
- 反映システム定義の例 507, 508
- 反映システム定義ファイル 143, 322, 325, 344
- 反映システム定義ファイルの運用方法 582
- 反映障害による未送信情報の取得 901
- 反映情報キューファイル 322, 325, 344
- 反映情報キューファイルサイズ [反映環境定義] 454
- 反映情報キューファイル [データ連動の仕組み] 44, 46, 48
- 反映情報キューファイルの運用方法 582
- 反映情報キューファイル [反映処理で使うファイル] 142
- 反映情報キューファイル [反映処理の概要] 141
- 反映情報キューファイル名 [反映環境定義] 454
- 反映情報編集 UOC 152, 772
- 反映情報編集 UOC で使う関数の文法 780
- 反映情報編集 UOC の作成手順 (UNIX) 774
- 反映情報編集 UOC の作成手順 (Windows) 778

- 反映情報編集 UOC のサンプル 803
- 反映情報編集 UOC のヘッダファイルの一覧 791
- 反映情報編集 UOC を作成するときの注意 800
- 反映処理 44, 141
- 反映処理コミット間隔〔反映環境定義〕 458
- 反映処理数リセットイベント 190
- 反映処理数リセットイベントコード〔反映環境定義〕 458
- 反映処理停止イベントコード〔反映環境定義〕 458
- 反映処理で使うファイル 142
- 反映処理に関する監査証跡を取得しない場合の注意 801
- 反映処理の COMMIT 発行間隔の設計 333
- 反映処理の DISCONNECT 発行間隔の設計 332
- 反映処理の運用 576
- 反映処理の開始方式の設計 336
- 反映処理のコミット間隔の変更〔hdsrctl コマンドの文法〕 724
- 反映処理の再開始〔hdsrctl コマンドの文法〕 724
- 反映処理の制御〔コマンドの文法〕 724
- 反映処理の単位 149
- 反映処理の停止〔hdsrctl コマンドの文法〕 724
- 反映処理の停止方式の設計 339
- 反映処理の同期点処理 158
- 反映処理のプロセス構成 144
- 反映処理の方式 146
- 反映処理の方式の設計 326
- 反映処理方式の切り替えの設計 343
- 反映ステータスファイル 141, 323, 325, 344
- 反映ステータスファイルサイズ〔反映環境定義〕 455
- 反映ステータスファイル名〔反映環境定義〕 455
- 反映制御ブロック名 782
- 反映対象表の作成 223
- 反映対象表の作成方法 223
- 反映対象表の条件 224
- 反映滞留時間 169
- 反映通信マスタプロセス 145
- 反映定義 49
- 反映定義サーバプロセス 145
- 反映定義〔定義の文法〕 474
- 反映定義ファイル 143, 322, 325, 344
- 反映定義ファイルの運用方法 582
- 反映定義ファイル名〔反映環境定義〕 454
- 反映定義を省略したときに仮定される値 474
- 反映データブロック名 784
- 反映トランザクション 178
- 反映トランザクション同期機能 173
- 反映トランザクション同期機能を使用している場合の構成変更 600
- 反映トレースファイル 143, 323, 325, 345
- 反映表定義 480
- 反映プロセス 145
- 反映プロセスの反映情報キューファイル読み込み間隔〔反映環境定義〕 468
- 反映プロセスの反映情報キューファイル読み込み間隔〔反映システム定義〕 444
- 反映方式の変更〔hdsrctl コマンドの文法〕 724
- 反映方法の設計 326
- 反映マスタステータスファイル 143, 323, 325, 344
- 反映マスタステータスファイルの運用方法〔反映側 Datareplicator〕 583
- 反映マスタプロセス 145
- 反映抑止対象リストファイル 144
- 反映抑止によるスキップ 153

ひ

- ビュー表〔用語解説〕 1254
- 表 52
- 表識別子〔抽出定義〕 418
- 表識別子〔反映定義〕 483, 487
- 表単位反映イベント 190
- 表単位反映イベントコード〔反映環境定義〕 457
- 表単位反映再起動イベント 190
- 表単位反映再起動イベントコード〔反映環境定義〕 457
- 表単位反映方式 147, 327
- 表単位分割方式 148, 327

ふ

- ファイル構成〔抽出側 Datareplicator〕 278
- ファイル構成〔反映側 Datareplicator〕 321
- ファイル転送プログラム 48
- ファイル独自の形式の定義の規則 369
- ファイルの二重化 196
- ファイルの二重化運用時のファイル障害への対処〔抽出側 Datareplicator〕 845
- ファイルの二重化を使うときの運用 633
- フィールド 52
- フィールド名称〔更新情報定義〕 498
- フィールド名称〔反映定義〕 478, 481
- 複数システム間でのデータ連動システムの注意 272
- 複数の抽出対象表を反映側で一つにまとめ、一つの表に反映 215
- 不整合が発生した場合の初期化手順 857
- 部分初期開始 531, 574
- プラグイン〔用語解説〕 1254
- フレキシブルハッシュ分割〔用語解説〕 1254
- プロセス〔用語解説〕 1254
- フロントエンドサーバ〔用語解説〕 1254
- 分割キーインデクス〔用語解説〕 1254

へ

- 別形式の表にデータ連動するパターン 120, 207
- ヘッダファイルの一覧 791, 819, 834

ほ

- ポート番号 91, 94
- ポート番号〔用語解説〕 1255
- ホスト名 91, 260, 265
- ホスト名の登録〔Windows〕 113
- ホスト〔用語解説〕 1255
- ボリューム〔用語解説〕 1255

ま

- マージ表の作成 156
- マッピングキー 49, 224
- マッピングキーの対応の設計 224

- マルチ FES 機能 149
- マルチ FES 機能に対応した反映方式 149
- マルチ FES 機能を使う場合の反映グループ定義の例 514
- マルチ FES 機能を使う場合の反映処理の方式の設計 328
- マルチ HiRDB〔用語解説〕 1255

み

- 三つ以上のシステムの間で双方向更新をするデータ連動システム 276
- 未反映情報ファイル 143, 323, 325, 344
- 未反映情報ファイルサイズ〔反映環境定義〕 455
- 未反映情報ファイルの運用方法 585
- 未反映情報ファイル名（セカンダリ）〔反映環境定義〕 455
- 未反映情報ファイル名（プライマリ）〔反映環境定義〕 455

め

- メインフレーム側 DB 39, 265
- メインフレーム側 DB から HiRDB へデータ連動する場合の設計 265
- メインフレーム側 DB から HiRDB へデータ連動する場合のソフトウェア構成 65
- メインフレーム側 DB から HiRDB へのデータ連動 46
- メインフレーム側 DB と反映側 Datareplicator との対応 268
- メッセージ出力の抑止〔抽出側 Datareplicator〕 135
- メッセージ出力の抑止〔反映側 Datareplicator〕 159
- メッセージの一覧 909
- メッセージの概要 910
- メッセージの記述形式 912
- メッセージの出力形式 911
- メッセージの出力先 910
- メモリ資源の設計 307, 349

も

- 文字コードの種類 235
- 文字コードの対応の設計 235
- 文字コードの変換 236

文字コード変換用マッピングテーブルの編集 [コマンドの文法] 701

戻り値 781

ゆ

ユーザ LOB 用 RD エリア [用語解説] 1255

ユーザOWNコーディング 134, 152, 771

ユーザOWNコーディングの障害時の対処 [抽出側 Datareplicator] 845

ユーザOWNコーディングの障害時の対処 [反映側 Datareplicator] 853

ユーザOWNコーディングを使った更新情報の加工 152

ユーザOWNコーディングを使ってデータ連動するパターン 123, 218

ユーザ権限の確認 [UNIX] 74

ユーザ権限の確認 [Windows] 95

ユティリティ実行時の注意 222

ユニット識別子 [pdls コマンドの文法] 767

ユニット制御情報定義 [HiRDB の定義] 428

よ

要因コード一覧 1224

用語解説 1248

横分割 [用語解説] 1255

予約語 1241

ら

ラージファイルの運用 637

り

リソースの設計 [抽出側 Datareplicator] 299

リソースの設計 [反映側 Datareplicator] 344

リソースの設計 [反映側 HiRDB] 335

リバランスユティリティ 223

リレーショナル型 50

リレーショナル型データベース 50

る

ループバック 273

ループバックの抑止 273

れ

レコード 52

レコード型 52

レコード実現値 52

列 52

列データ編集 UOC 152, 807

列データ編集 UOC 関数識別子 [反映定義] 480

列データ編集 UOC で使う関数の文法 812

列データ編集 UOC で使うヘッダファイルの一覧 818

列データ編集 UOC の作成手順 (UNIX) 808

列データ編集 UOC の作成手順 (Windows) 811

列データ編集 UOC のサンプル 820

列データ編集 UOC を作成するときの注意 819

列名 [抽出定義] 418

レプリケーション機能 37

レプリケーション機能 [用語解説] 1255

連結演算による更新データの連動 136

連動回復用バックアップファイル 869

連動回復用バックアップファイル [用語解説] 1256

連動系切り替え 606

連動パターン 120

連動パターンの設計 203

ろ

ログ関係のファイルのステータスの変更 [コマンドの文法] 766

わ

割り当てファイル種別 [抽出環境定義] 400

割り当てファイル種別 [反映環境定義] 465