

Rational™ ClearQuest® MultiSite

管理ガイド

バージョン: 2003.06.10 およびそれ以降

G126-5408-00

UNIX/WINDOWS 版

法的通知

Copyright © 1997-2003, Rational Software Corporation. All Rights Reserved.

バージョン番号: 2003.06.10 およびそれ以降

本マニュアル (「本著作物」) は、アメリカ合衆国その他の国々の著作権法及び種々の条約により保護されています。Rational Software Corporation の文書による事前の同意を得ることなく本著作物を複製し又は頒布することは、禁じられています。

本著作物はライセンスに基づいて提供されるもので、ライセンス規定に従う場合にのみ、使用または複製できます。ライセンスで明示的に許可されている場合を除き、本マニュアルまたはその複製を第三者に提供することは禁じられています。本著作物の権利または所有権を譲渡することはできません。ライセンス条項の全文については、ライセンス契約書をお読みください。

Rational Software Corporation, Rational, Rational Suite, Rational Suite ContentStudio, Rational Apex, Rational Process Workbench, Rational Rose, Rational Summit, Rational Unified process, Rational Visual Test, AnalystStudio, ClearCase, ClearCase Attache, ClearCase MultiSite, ClearDDTS, ClearGuide, ClearQuest, PerformanceStudio, PureCoverage, Purify, Quantify, Requisite, RequisitePro, RUP, SiteCheck, SiteLoad, SoDa, TestFactory, TestFoundation, TestMate, TestStudio は、Rational Software Corporation の米国およびその他の国における登録商標です。Rational のロゴ、Connexis, ObjecTime, Rational Developer Network, RDN, ScriptAssure, XDE は、Rational Software Corporation の米国およびその他の国における商標です。その他すべての名前は、識別の目的でのみ使用されているものであり、それぞれの会社の商標または登録商標です。

米国特許番号 5,193,180、5,335,344、5,535,329、5,574,898、5,649,200、5,675,802、5,754,760、5,835,701、6,049,666、6,126,329、6,167,534、6,206,584 の請求の範囲内の部分。このほかにも米国特許及び国際特許申請中。

米国政府の権利

このソフトウェアおよび文書は、「商業的コンピュータソフトウェア」、「商業的ソフトウェア」または「使用が制限されたコンピュータソフトウェア」として提供され、規約は該当する DFARS 252.227、DFARS 252.211、FAR 2.101、FAR 52.227 (またそれ以前に定められた条項) に規定されています。本ソフトウェア製品およびドキュメントの使用、複製、または開示は、DFARS 227.7202、FAR 52.227-19 の下位条項 (c)、または FAR 52.227-14 (またはその改訂された規定) に定められるように、該当する Rational Software Corporation ライセンス契約書の条項の制約を受けます。

免責事項

本書および関連ソフトウェアは、ライセンス契約に基づいて使用することができます。そのような使用許諾契約書に別段の明示的な規定がある場合を除き、また、それぞれの国の法律により禁止または制限されている場合を除き、Rational Software Corporation は、本メディア、ソフトウェア製品、およびその関連文書について、明示的にも暗黙的にも、商品性に関する保証、非権利侵害性に関する保証、特定目的への適合性に関する保証、取り扱い、使用、または取引行為に伴う保証、およびライセンシーによる静穏無事な製品使用に対する妨害がないことの保証について一切の責任を負いません。

第三者の通知、コード、使用許諾および確認

Portions Copyright ©1992-1999, Summit Software Company. All rights reserved.

Microsoft、Microsoft のロゴ、Active Accessibility、Active Client、Active Desktop、Active Directory、ActiveMovie、Active Platform、ActiveStore、ActiveSync、ActiveX、Ask Maxwell、Authenticode、AutoSum、BackOffice、BackOffice のロゴ、bCentral、BizTalk、Bookshelf、ClearType、CodeView、DataTips、Developer Studio、Direct3D、DirectAnimation、DirectDraw、DirectInput、DirectX、DirectXJ、DoubleSpace、DriveSpace、FrontPage、Funstone、Genuine Microsoft Products のロゴ、IntelliEye、IntelliEye のロゴ、IntelliMirror、IntelliSense、J/Direct、JScript、LineShare、Liquid Motion、Mapbase、MapManager、MapPoint、MapVision、Microsoft Agent のロゴ、Microsoft eMbedded Visual Tools のロゴ、Microsoft Internet Explorer のロゴ、Microsoft Office Compatible のロゴ、Microsoft Press、Microsoft Press のロゴ、Microsoft QuickBasic、MS-DOS、MSDN、NetMeeting、NetShow、Office のロゴ、Outlook、PhotoDraw、PivotChart、PivotTable、PowerPoint、QuickAssembler、QuickShelf、RelayOne、Rushmore、SharePoint、SourceSafe、TipWizard、V-Chat、VideoFlash、Visual Basic、Visual Basic のロゴ、Visual C++、Visual C#、Visual FoxPro、Visual InterDev、Visual J++、Visual SourceSafe、Visual Studio、Visual Studio のロゴ、Vizact、WebBot、WebPIP、Win32、Win32s、Win64、Windows、Windows CE のロゴ、Windows のロゴ、Windows NT、Windows Start のロゴ、XENIX は、Microsoft Corporation の米国およびその他の国における商標または登録商標です。

Sun、Sun Microsystems、Sun のロゴ、Ultra、AnswerBook 2、medialib、OpenBoot、Solaris、Java、Java 3D、ShowMe TV、SunForum、SunVTS、SunFDDI、StarOffice、および SunPCi は、Sun Microsystems の米国および他の国における商標または登録商標です。

Purify は、Sun Microsystems, Inc. の米国特許番号 5,404,499 の下にライセンス供与されています。

Globetrotter ソフトウェア (FLEXIm ライブラリおよびユーティリティ) の本来の用途は、ソフトウェアライセンス管理であり、他の製品またはアプリケーションにこれらのソフトウェアを組み込むことは、ライセンスに含まれません。

BasicScript は、Summit Software Company の登録商標です。

デザインパターン : Erich Gamma、Richard Helm、Ralph Johnson および John Vlissides による再使用可能なオブジェクト指向のソフトウェアのエLEMENT。Copyright © 1995 by Addison-Wesley Publishing Company, Inc. All rights reserved.

Copyright © 1997 OpenLink Software, Inc. All rights reserved.

本ソフトウェアおよびドキュメントは部分的に、カリフォルニア大学理事会により使用許諾されている BSD Networking Software Release 2 に基づいています。当社はその開発におけるカリフォルニア大学バークレー校の Computer Systems Research Group および Electrical Engineering and Computer Sciences Department 並びに「その他の協力者」の役割を認めます。

本製品は、Apache (http://www.webdav.org/mod_dav/) における使用のために Greg Stein により開発されたソフトウェアを含んでいます。

追加の法的通知は、お客様の Rational ソフトウェア インストールに含まれています。

目次

まえがき	xvii
本書について	xvii
ClearCase マニュアル ロードマップ	xviii
ClearQuest とほかの Rational 製品との統合	xix
表記規則	xx
オンライン マニュアル	xxi
カスタマ サポート	xxi

MultiSite の概要

MultiSite の紹介	3
MultiSite のアーキテクチャの理解	3
データベース セットのレプリカ	3
族、ファミリー、サイト	4
スキーマ リポジトリの種類	5
MultiSite の用語	5
データベースとデータベースのレプリカ	6
ファミリー内のレプリカの同期	6
独立した開発の実現: マスターシップ	8
族内での単一のコード ページの適用	8
MultiSite の操作	11
マスターシップ	11
競合の解決	12
操作ログ	12
各レプリカの操作の追跡	13
oplog ID とエポック数	15
間接的な同期	16
MultiSite の実装の計画	19
MultiSite のインストール	19
MultiSite のライセンス	20
ClearCase と ClearQuest での SHIPPING サーバーの使用	21

マスターシップ ポリシーの定義	22
互換性に関する問題	23
mkreplica -export を実行する前の cqintsrv プロセスの強制終了	23
ClearQuest UCM 統合におけるマスターシップ ポリシー	23
ClearQuest MultiSite とほかの Rational 製品との統合	23
MultiSite ユース モデル	23
言語とコード ページのサポート	24
複数コンピュータでの multiutil コマンドの実行	24
マスターシップ作成方針	25
転送方法の同期	25
同期パターン	25
交換の方向	27
1 対 1 型同期とリング型同期	28
1 対多型同期	28
多対多型同期	30
同期スケジュール	30
データベースのバックアップ方針	32
レプリカのパラメータのスクラブ	33
oplog のスクラブ	33
export_sync のスクラブ	34
スペースを含むパス名の処理	34
MultiSite 管理者の管理責任	35
配置タスクの概要	36

MultiSite のコマンド セット 39

multiutil の使用	39
サブコマンドの説明	40
レプリカの作成、同期、管理に関するコマンド	40
オブジェクトのマスターシップに関するコマンド	40
障害回復に関するコマンド	41
multiutil ユーティリティのコマンド	41
MultiSite のほかのコマンド	42
MultiSite の API 関数	42
コマンドでのレプリカの指定	43

転送方法の選択	45
ファイルベースの方法	45
電子メールの使用方法	45
FTP の使用方法	46
物理メディアの使用方法	47
保存および転送	47
パケットのディレクトリ	47
パケットの送付	48
保存および転送の問題	48
レプリカ ホスト間通信	48
パケット サイズの制限	48
保存および転送機能の設定	49
保存および転送機能へのパケットの引き渡し	49
記憶クラスを使用してのパケットの区別	49
間接発送ルートの設定	50
再試行、有効期限、戻りデータ	51
ホストにアクセスできない場合のタイムアウト時間の設定	52
混在環境でのエラーの通知	52
パケット以外のファイルの送信	52
ファイアウォールを通過する保存および転送の使用方法	53
ファイアウォールに関する考慮事項	54
ファイアウォールの設定によるアクセス制限	55
公開ホストへの SHIPPING サーバーのインストール	55
albd_server と shipping_server で使用するポートの制御	55
ポート値の指定	56
ファイアウォールを通過する保存および転送を使用するためのチェックリスト	57

レプリカ作成と同期

データベース レプリカの作成	61
レプリカ作成の概要	61
データベースのアクティブ化	61
レプリカ作成パケットのエクスポート	62
空の製造元データベースの作成	63

レプリカ作成パケットのインポート	63
新しいレプリカの追加	64
インポートが失敗した場合の回復	64
レプリカ作成のシナリオ	64
前提条件	65
データベース セットのアクティブ化	65
エクスポート フェーズ	65
送付フェーズ	66
インポート フェーズ	66
レプリカの同期	69
同期の成功に関する想定	69
スキーマの更新情報を含むパケットの適用	69
手動による同期	70
エクスポート フェーズ	70
送付フェーズ	70
インポート フェーズ	71
同期の自動化	71

MultiSite 管理

レプリカの管理	75
レプリカのプロパティの表示	75
同期サーバーの移動または名前変更	75
レプリカの移動または製造元データベース ソフトウェアの変更	76
レプリカに対する ID ブロック割り当ての変更	76
レプリカのマスターシップの変更	77
レプリカの削除	77
族からの動作レプリカの削除	77
族からの動作不能サイトの削除	79
族の最後のレプリカを削除した後の MultiSite の使用法	80
マスターシップの管理	81
ユーザー データベース オブジェクトのマスターシップのコマンド	81
レコードのマスターシップ情報の表示	81

データベース オブジェクトのマスターシップの変更	81
GUI によるレコードのマスターシップの転送	82
chmaster によるレコードのマスターシップの転送	83
GUI によるワークスペース項目のマスターシップの転送	84
chmaster によるワークスペース項目のマスターシップの転送	84
ユーザーまたはグループのマスターシップの転送	85
GUI によるユーザーまたはグループのマスターシップの変更	85
chmaster によるユーザーまたはグループのマスターシップの変更	86
作業スキーマ リポジトリのマスターシップの転送	87
誤ってマスターシップを変更した場合の修正	88

トラブルシューティング

MultiSite の操作のトラブルシューティング 91

レプリカのエクスポートの問題	91
mkreplica -export の失敗からの回復	91
スキーマ リポジトリとユーザー データベースのアンロック	92
以後の multiutil コマンドが失敗する	92
レプリカのインポートの問題	93
同期エクスポートの問題	94
oplog 項目が見つからない	94
出力記憶ベ이에パケットが累積する	95
レプリカ自体を更新することはできない	95
転送の問題	95
エラー メッセージ	95
無効な宛先	97
デリバリーに失敗する	98
シッピング サーバーが起動できない、または接続を拒否された	98
発送オーダーの有効期限切れ	98
同期インポートの問題	99
入力記憶ベ이에パケットが累積する	99
パケットを適用できるローカル レプリカがない	100
入カストリームからの読み取りが失敗する	100
その他の問題	100
失われたパケットの回復	101

循環重複リンクの削除	102
名前の競合の解決	102
ワークスペース名の競合と ClearQuest Web	103
ワークスペース項目の名前の変更	103
あいまいなワークスペース項目の操作	103
状態なしレコード タイプ名の競合の解消	104
レコード名の変更	104
レコードの一意性の確認	104
名前の競合がある状態なしレコードの検索	104
ユーザー名やユーザー グループ名の競合の特定	105
ユーザー名の変更	105
あいまいなユーザーとユーザー グループに対する multiutil の使用法	106
データベースのレプリカを作成した後のデータベース登録の更新	106
データベース レプリカの復元	107
バックアップからのレプリカの復元	108

MultiSite リファレンス ページ

MultiSite リファレンス ページ	111
activate	112
chepoch	115
chmaster	118
chreplica	123
control_panel	126
describe	128
dumpoplog	131
lsepoch	136
lspacket	139
lsreplica	142
mkorder	148
mkreplica	153
MultiSite コントロール パネル	167
multiutil	173
recoverpacket	174
restorereplica	178

rmreplica	182
scruboplog	185
shipping.conf	189
shipping_server	196
syncreplica	200
索引	211

図目次

図 1	MultiSite の族	4
図 2	レプリカの同期プロセス	7
図 3	データベースに対する変更履歴	13
図 4	ファミリの状態	13
図 5	古いレプリカ	14
図 6	2 つのレプリカ間での更新	14
図 7	ピア ツー ピア型同期パターン	26
図 8	階層型同期パターン	26
図 9	一方向と双方向の更新	27
図 10	1 対 1 型同期パターン	28
図 11	リング型同期パターン	28
図 12	シングルハブ型同期パターン	28
図 13	マルチハブ型同期パターン	29
図 14	ツリー型同期パターン	29
図 15	多対多型同期パターン	30
図 16	同期スケジュール	32
図 17	保存および転送の構成	54

表目次

表 1	boston_hub レプリカにおける 2 行のエポック数マトリックス	15
表 2	boston_hub レプリカにおける 3 行のエポック数マトリックス	17
表 3	記憶ベイに必要なディスク容量	20
表 4	ファミリの情報.....	36
表 5	レプリカの作成、同期、管理に関するコマンド	40
表 6	オブジェクトのマスターシップに関するコマンド	40
表 7	障害回復に関するコマンド.....	41
表 8	multiutil ユーティリティのコマンド	41
表 9	MultiSite のほかのコマンド	42
表 10	MultiSite の API 関数	42
表 11	パケット送付方法の選択肢.....	45
表 12	発送エラー メッセージ.....	96

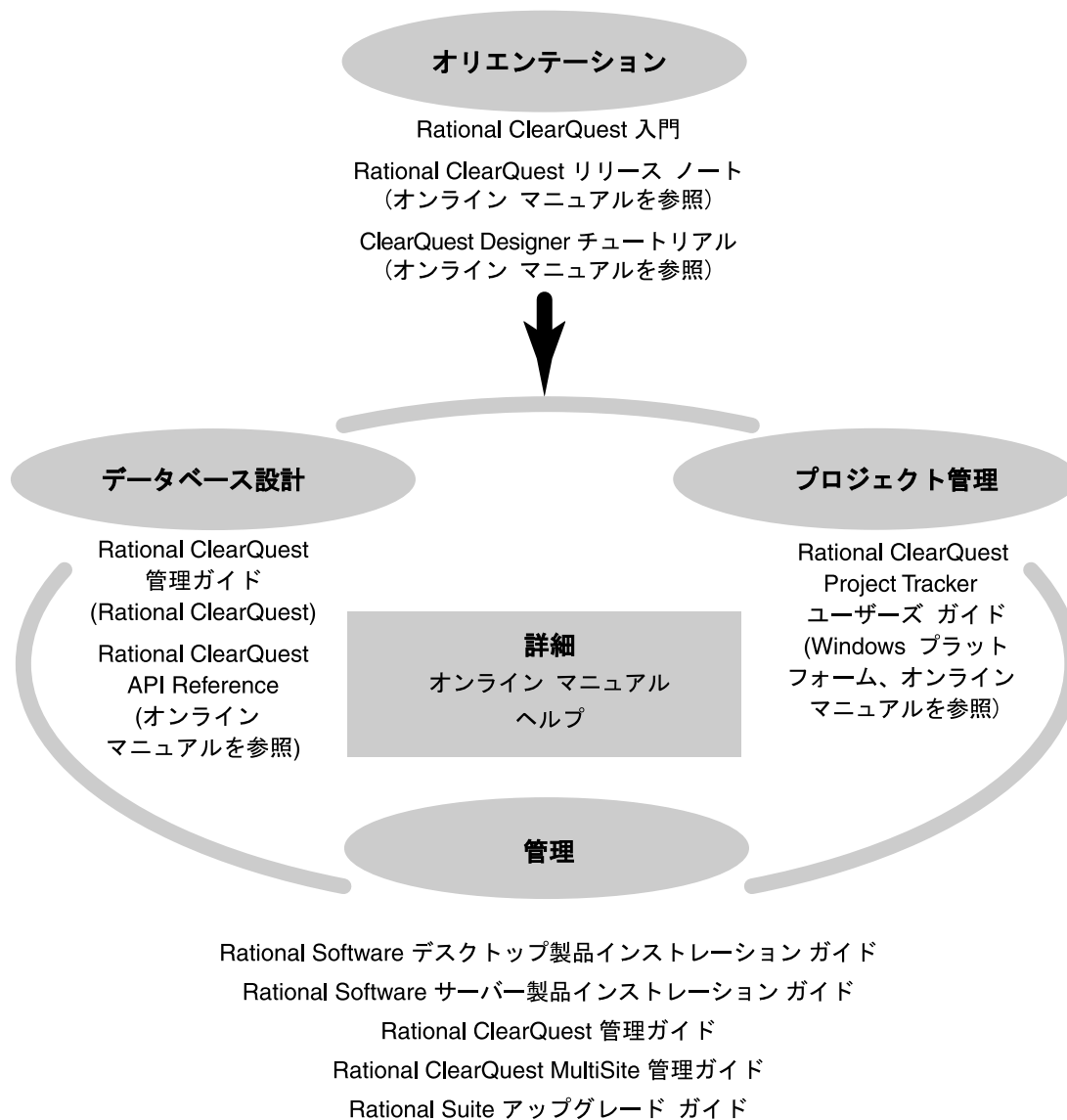
まえがき

Rational™ ClearQuest® MultiSite (以下 MultiSite) は Rational ClearQuest と共に使用するオプション製品です。MultiSite を使用すると、離れた場所にいるプロジェクト チームによるソフトウェアの平行開発が可能になると共に、ClearQuest データベースを自動的にエラーなく複製することができます。

本書について

本書はすべての MultiSite 管理者を対象とし、ClearQuest の使用経験があることを前提として説明しています。本書では、MultiSite の概要を紹介し、MultiSite の設定方法と使用方法を説明し、トラブルシューティングを提示します。

ClearCase マニュアル ロードマップ



ClearQuest とほかの Rational 製品との統合

統合	説明	説明が記載されている場所
ClearQuest - ベース ClearCase®	変更依頼を ClearCase エLEMENT のバージョンに関連付けます。	『Rational ClearCase ソフトウェア 開発ガイド』 『Rational ClearCase プロジェクト 管理ガイド』 『Rational ClearQuest 管理ガイド』
ClearQuest - PurifyPlus	開発者が、PurifyPlus から ClearQuest を呼び出せるように します。	PurifyPlus のヘルプ 『Rational ClearQuest 管理ガイド』
ClearQuest - RequisitePro	開発者が、RequisitePro から ClearQuest を呼び出して、要求を ClearQuest 変更依頼に関連付ける ことができますようにします。	『Rational Suite® 管理ガイド』 『Rational ClearQuest 管理ガイド』
ClearQuest - SoDA	ClearQuest から情報を収集して、 その情報をさまざまなレポート 形式で表示します。	『Using Rational SoDA for Word』 『Using Rational SoDA for Frame』 SoDA のヘルプ
ClearQuest - Test Manager	開発者が、TestManager から ClearQuest を呼び出せるように します。	『Using Rational TestManager』 『Rational ClearQuest 管理ガイド』
ClearQuest - Robot	開発者が、Robot から ClearQuest を 呼び出せるようにします。	『Rational Robot User's Guide』 Rational Robot のヘルプ
ClearQuest - UCM	UCM アクティビティを ClearQuest レコードにリンクします。	『Rational ClearCase ソフトウェア 開発ガイド』 『Rational ClearCase プロジェクト 管理ガイド』 『Rational ClearQuest 管理ガイド』

表記規則

本書の表記規則は次のとおりです。

- `ccase-home-dir` は、ClearCase 製品ファミリーがインストールされているディレクトリを表します。デフォルトのインストール ディレクトリは、UNIX では `/opt/rational/clearcase`、Windows では `C:\Program Files\Rational\ClearCase` です。
- `cquest-home-dir` は、Rational ClearQuest がインストールされているディレクトリを表します。デフォルトのインストール ディレクトリは、UNIX では `/opt/rational/clearquest`、Windows では `C:\Program Files\Rational\ClearQuest` です。
- **太字**は、コマンド名やブランチ名など、ユーザーが入力可能な名前に使用します。
- **sans serif** フォントは、ファイル名、ディレクトリ名、ファイル拡張子に使用します。
- メニュー名やチェック ボックス名のような、GUI 要素は、`[]` で囲んで表記します。
- **等幅**フォントは、例に使用します。ユーザー入力とプログラム出力を区別する必要がある場合、ユーザー入力には**太字**を使用します。
- 出力されない文字は、`<EOF>`、`<NL>` のように表示します。
- キー名やキーの組み合わせは、大文字で `SHIFT`、`CTRL+G` のように表示します。
- `[]` 大カッコは、書式や構文の記述でオプション項目を囲むために使用します。
- `{ }` 中カッコは、書式や構文の記述で選択項目のリストを囲むために使用します。
- `|` 縦棒は、選択項目のリストを区切るために使用します。
- ... 構文記述内の省略記号は、省略記号に先行する項目や行を 1 回以上繰り返せることを示します。それ以外の省略記号は、情報の省略を示します。

メモ: あるコンテキストでは、`"*` や `"?"` と同様に、`"..."` をパス名内でワイルドカードとして使用することができます。詳細については、`wildcards_ccase` のリファレンス ページを参照してください。
- コマンド名またはオプション名に省略形がある場合、`"中点" (·)` は最短の有効な省略形を示します。たとえば次のような機能があります。

`lsc:checkout`

オンライン マニュアル

Rational ClearQuest には、次のオンライン マニュアルが含まれています。

ヘルプ システム: [ヘルプ] メニュー、[ヘルプ] ボタン、[F1] キーを使用します。

リファレンス ページ: Windows 上で、Rational ClearQuest MultiSite のリファレンス ページを表示するには、コマンドプロンプトで「multiutil man command-name」と入力します。

API リファレンス: [スタート]、[プログラム]、[Rational Software]、[Rational ClearQuest] をポイントし、[ClearQuest API リファレンス] をクリックします。

チュートリアル: [スタート]、[プログラム]、[Rational Software]、[Rational ClearQuest] をポイントし、[ClearQuest Designer チュートリアル] をクリックします。

PDF マニュアル: 次のディレクトリに移動します。

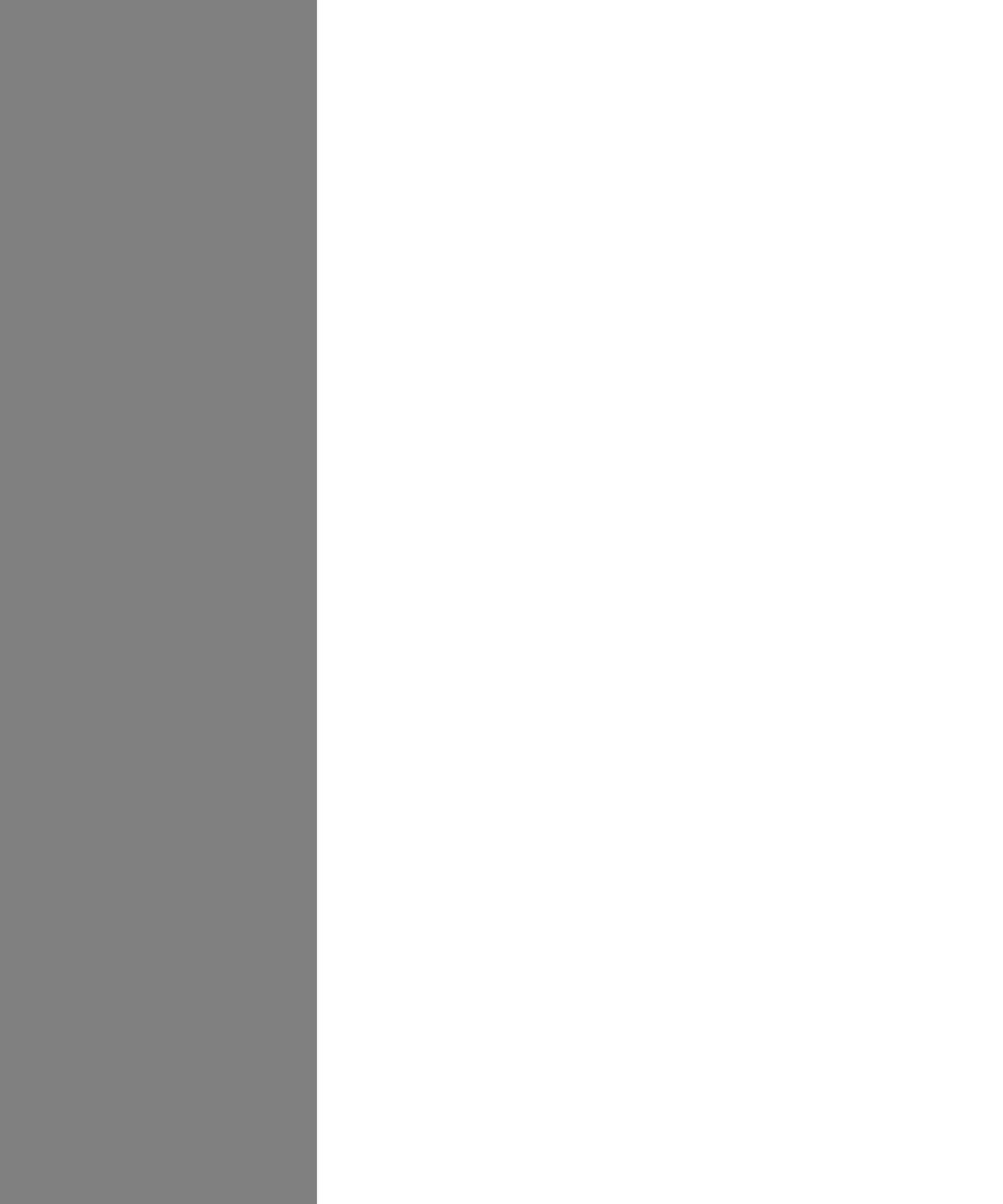
- UNIX の場合は、cquest-home-dir/doc/books
- Windows の場合は、cquest-home-dir¥doc¥books

カスタマ サポート

ソフトウェアやマニュアルの問題については、電話、ファックス、または電子メールで以下の Rational カスタマ サポートまでお問い合わせください。サポートの時間帯、対応言語、その他のサポート情報については、日本ラショナル ソフトウェア社の Web サイトのサポート情報 (<http://www.rational.co.jp/supports/>) を参照してください。

地域	電話	Fax	電子メール
アジア太平洋 (日本を含む)	+61-2-9419-0111	+61-2-9419-0123	support@apac.rational.com (英語のみ対応) support@japan.rational.com (日本語対応可)

MultiSite の概要



Rational ClearQuest MultiSite は、Rational ClearQuest への追加機能です。MultiSite を使用すると、異なる場所で作業する開発者が同じデータベース セット (スキーマ リポジトリと関連ユーザー データベース) を使用できます。それぞれの場所にはスキーマ リポジトリとユーザー データベースの別個のコピーまたはレプリカが置かれます。1 つのレプリカに加えられた変更は、いつでも更新パッケージに格納してほかのレプリカに送信できます。更新プロセスは自動的に行うことも、手作業でコマンドを実行して行うこともできます。

MultiSite を使用することにより、組織は独立しているが相互に関連のある開発作業結果を、複数の都市、国、大陸に分散することができます。たとえば、米国のある会社が、インド、アルゼンチン、日本、オーストラリアに開発とテストを行うサイトを所有しているとします。すべてのエンジニアが米国のデータベースにアクセスするのは非現実的です。そのため、この会社では MultiSite を使用して、開発を分散します。

地理的に 1 つの場所で MultiSite を使用することもできます。具体的には、独立した複数のグループによる同じ開発データの使用、またはバックアップ メカニズムが挙げられます。たとえば、バックアップの信頼性を高める必要がある場合は、データベース セットのレプリカをローカルに作成できます。

この章では、MultiSite の主要機能の概要について説明します。「第 2 章 MultiSite の操作」では、これらの機能について詳しく説明します。

MultiSite のアーキテクチャの理解

次の項で、MultiSite のアーキテクチャについて説明します。

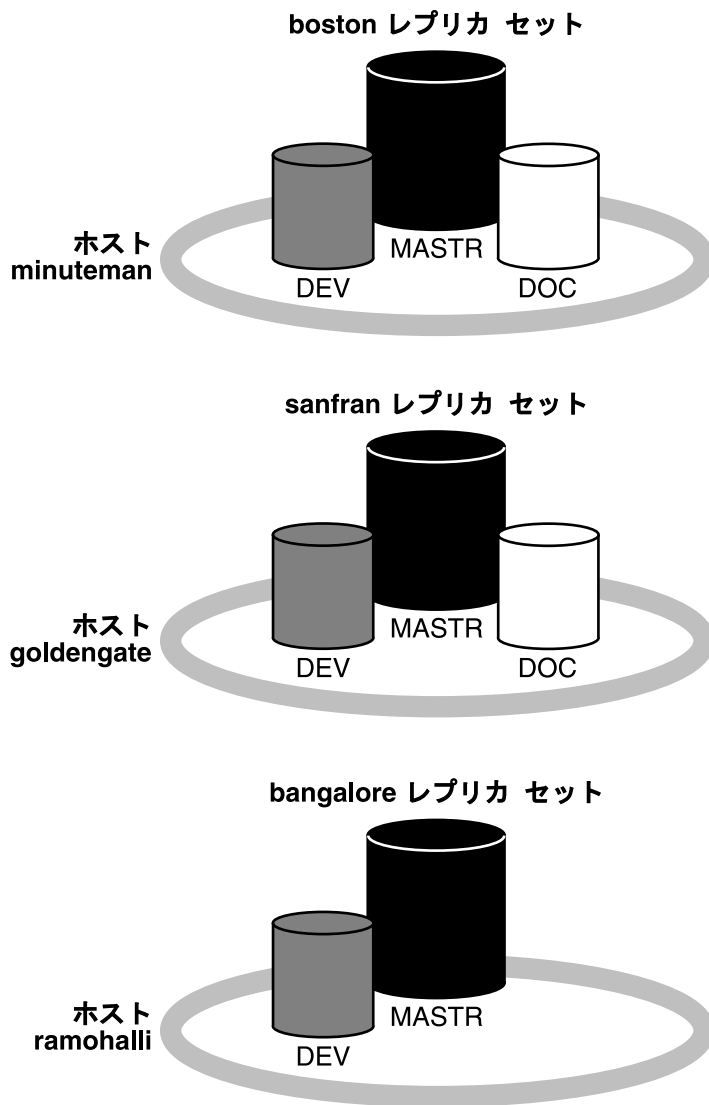
データベース セットのレプリカ

データベース セットは、スキーマ リポジトリと関連ユーザー データベースで構成されます。ユーザー データベースは、対応するスキーマ リポジトリなしでは存在できないので、データベース セットをレプリカを作成する場合には、スキーマ リポジトリとユーザー データベースの両方のレプリカを作成する必要があります。データベースのレプリカを操作する場合、実際にはスキーマ リポジトリとユーザー データベースのレプリカの 2 つのデータベースを操作することになります。

族、ファミリー、サイト

族 (図 1) は、データベース セットのすべてのレプリカで構成されます。族内で、レプリカはレプリカ ファミリーに分類されます。レプリカ ファミリーは、特定のデータベースのすべてのレプリカです。

図 1 MultiSite の族



スキーマ リポジトリとユーザー データベースは別のレプリカ ファミリに属します。ユーザー データベースのファミリー名はデータベース名と同じです。スキーマ リポジトリのファミリー名は常に MASTR です。

サイトは、同じ場所にある同じ族内のレプリカの名前付きコレクションです。各サイトには、スキーマ リポジトリのレプリカが 1 つと各ユーザー データベース ファミリのレプリカが最大 1 つあります。各サイトは同期サーバーからサービスを受けます。同期サーバーは、サイトのファミリー内のレプリカとの間で更新パケットを送受信します。サイトの更新時には、ユーザー データベースのレプリカかスキーマ リポジトリのレプリカ、またはその両方への更新を実施できます。

スキーマ リポジトリの種類

族内では、スキーマ リポジトリのレプリカの 1 つが稼働中のスキーマ リポジトリです。稼働中のスキーマ リポジトリでは、スキーマを変更したり、追加のユーザー データベースを作成してからレプリカを作成することができます。ほかのスキーマ リポジトリでは、スキーマを変更したり、ユーザー データベースを変更することはできません。各族には、稼働中のスキーマ リポジトリは 1 つしかありません。

どちらの種類スキーマ リポジトリでも、次のタスクを実行できます。

- 管理者ワークステーションからの **multiutil** コマンドの実行
- 保存および転送機能のインストールと設定
- 新しいレコードのサブミット
- 現在のレプリカによってマスター登録されたレコードの変更
- 現在のレプリカによってマスター登録されたレコードの管理

MultiSite の用語

MultiSite のマニュアルでは以下の用語が使用されます。

用語	定義
レプリカ	ユーザー データベースまたはスキーマ リポジトリのコピー。レプリカを参照するには、サイト名とファミリー名を使用します。
ファミリー	特定のユーザー データベースのすべてのレプリカ、または特定のスキーマ リポジトリのすべてのレプリカ。ユーザー データベースのレプリカのファミリー名は、元になるデータベースのデータベース名です。スキーマ リポジトリのファミリー名は常に MASTR です。
サイト	スキーマ リポジトリとそのユーザー データベースのレプリカ。

用語	定義
族	スキーマ リポジトリのすべてのレプリカと関連付けられたユーザー データベースのすべてのレプリカ。 同じデータベース セットから作成したレプリカは、同じ族名を使用します。族名は、データベース セットをアクティブ化すると指定されます。
ホストまたは同期サーバー	サイトのパケットを処理するネットワーク ノードの LAN 名または IP アドレス。このホストには、 Rational Shipping Server をインストールしておく必要があります。

データベースとデータベースのレプリカ

各レプリカは、スキーマ リポジトリ データベース テーブルに記録されます。このテーブルには、関連付けられた同期サーバーを含め、レプリカに関する情報が含まれています。各スキーマ リポジトリ データベースには、ほかに、データベース セット内の各データベースへの接続方法に関する情報が含まれています。この情報のレプリカは作成されません。

ClearQuest データベースに格納されているほとんどの情報についてレプリカが作成されますが、以下の情報のレプリカは作成されません。

- スキーマのチェックアウトされたコピー
- ユーザー データベースのスキーマ (ローカル管理者は、データベースのアップグレード時期を選択する必要があります)

ファミリー内のレプリカの同期

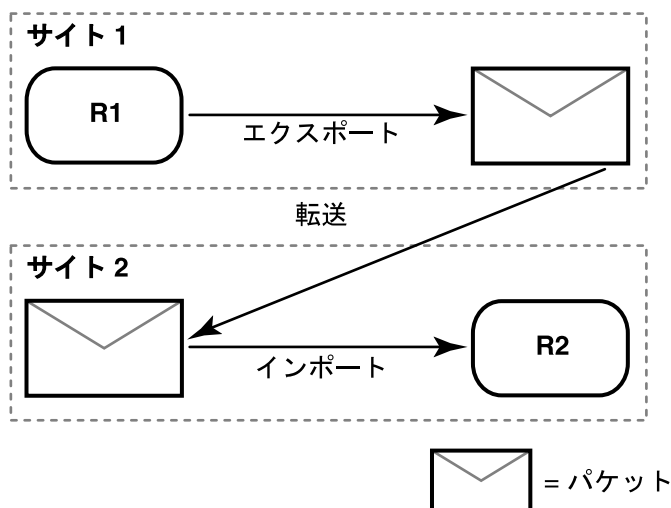
ClearQuest データベースのレプリカ内の情報は、複数のレプリカで変更されます。そのため、ファミリー内の各レプリカの内容は異なるのが普通です。実際、あるレプリカの内容がほかのレプリカの内容と同じになることはほとんどありません。レプリカ間の内容が大幅に異なることがないように、各レプリカからほかのレプリカに更新情報が送られます。その更新情報を受け取ったユーザー データベースのレプリカでは、ほかのいくつかのレプリカで行われた開発アクティビティを反映させて、レプリカのデータベースとスキーマ リポジトリの両方を更新します。

情報は、レプリカからパケット単位でエクスポートされます。論理パケットには、新しいレプリカを作成するため (レプリカ作成パケット)、または 1 つ以上の既存のパケットを更新するため (更新パケット) に必要なすべての情報が収められています。柔軟性を持たせると共に、データ送付装置の制約に対応するために、各論理パケットを一組の物理パケットとして作成できます。

mkreplica コマンドまたは syncreplica コマンドに **-export** オプションを指定して論理パケットを作成し、レプリカに送信すると、そのレプリカでは **mkreplica** コマンドまたは **syncreplica** コマンドに **-import** オプションを指定して、パケットのインポート処理を行います。送信側のレプリカで行われた変更 (ほかのレプリカの分が含まれる場合もあります) が、インポートするレプリカのユーザー データベースとスキーマ リポジトリに反映されます。論理パケットが複数の物理パケットから構成される場合、各物理パケットはインポート コマンドにより正しい順序で処理されます。同時に複数のインポートが行われない限り、同一レプリカで同じパケットが複数回インポートされてもエラーにはなりません。

図 2 は、エクスポート、送付、インポートの 3 つの同期処理フェーズを示しています。サイト 1 で **syncreplica -export** コマンドを実行すると、R1 の操作の記録がパケットに書き込まれます。そのパケットをサイト 2 に送付します。サイト 2 で **syncreplica -import** コマンドを実行すると、パケットの内容が R2 にインポートされます。各同期プロセスは一方方向です。2 つのレプリカ間で相互に更新を行うためには、同期プロセスを 2 回実行する必要があります。

図 2 レプリカの同期プロセス



各ファミリの使用パターン、組織の要望、ホスト コンピュータ間の接続のレベルに合わせて、同期方式を選ぶことができます。たとえば、あるファミリに関しては高速ネットワークを介して 1 時間ごとにレプリカを更新し、別のファミリに関しては電子メールやディスク ファイルを使用して月に 1 回または 2 回だけ更新することができます。同期の計画については、23 ページの「MultiSite ユース モデル」を参照してください。レプリカの作成と同期については、第 6 章と第 7 章を参照してください。12 ページの「操作ログ」では、レプリカと同期をサポートする仕組みについて説明します。

独立した開発の実現：マスターシップ

複数のレプリカに対して個別に変更が行われるので、それらの変更が競合することがあります。MultiSite 環境では、変更の追跡とデータの破壊防止のために、マスターシップと呼ばれる排他的変更権限方式を使用しています。マスターシップでは、レプリカのユーザーがデータを変更できる時期を決定します。

異なるレプリカで完全に独立して作業を進めると、結果として混乱を招きます。レコード SAMPL00001 が 3 つのレプリカで同時に作成された場合にも、どれが本当のレコード SAMPL00001 なのか、ほかの 2 つのバージョンをどう扱うべきなのか、わからなくなります。

このような場合、特定のオブジェクトにマスター レプリカ (またはマスター) を割り当てます。オブジェクトの最初のマスターは、作成されたオブジェクトが属するレプリカです。マスターシップは後で変更することができます (「第 9 章 マスターシップの管理」を参照)。一般的に、マスター レプリカ内でのみ、オブジェクトを修正または削除することができます。

ClearQuest データベースのほとんどのオブジェクトには、マスター レプリカがあります。マスターシップが、どのようにして競合する変更を防ぐかについては、11 ページの「マスターシップ」を参照してください。

一部の競合は避けられません。たとえば、jsmith という名前の新しいユーザーを、同期処理の間の同時期に 2 つ以上のユーザー データベースのレプリカで作成できます。このような競合は、オブジェクトの命名規則を設定することによって最小限に抑えることができますが、競合が発生した場合には、更新パケットのインポート中に処理されます。詳細については、12 ページの「競合の解決」を参照してください。

族内での単一のコード ページの適用

ClearQuest を使用する場合、ClearQuest データベースにアクセスする各クライアントは独自のコード ページを持ちます。コード ページは、該当する環境で有効な文字セットを指定します。ClearQuest を使用している場合、コード ページは、特定のクライアント上で正しく操作される文字セットを定義します。

ClearQuest と ClearQuest MultiSite のインターナショナル化サポートには制限があるので、異なるコード ページを使用するクライアントが同じデータを変更しようすると、データが破壊されることがあります。MultiSite 環境でこのような問題が発生する可能性があるのは、あるコード ページのデータを、別コード ページを実行しているレプリカにエクスポートした場合です。この場合、データの破壊や相違 (レプリカ ファミリのメンバが同じレコードに対して異なる値を持つ) が発生します。そのため、MultiSite 環境内のすべてのデータベース、データベース ホスト、サーバー、クライアントは、同じコード ページを使用し、データを ASCII 文字に制限する必要があります (UNIX クライアントの場合は必須)。

単一のコード ページの使用を強制し、データの一貫性を維持するために、ClearQuest 管理者はデータベース セットのデータ コード値を設定してから、データベース セットのレプリカを作成できるようにする必要があります。また、各データベースのコード ページを、製造元データベースのツールを使用して設定する必要もあります。バージョン 2003.06.00 より前の ClearQuest を実行しているクライアントがデータベースにアクセスする場合は、CharacterSetValidation パッケージを適用して、ユーザーが、複数のコード ページのデータをデータベース レコードに入力できないようにする必要があります。

コード ページとデータベース セット用のデータ コード ページ設定の詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

この章では、第 1 章で紹介した内容についてさらに詳しく説明します。

マスターシップ

次のオブジェクトには、マスター レプリカがあります。

- レコード
- ユーザーとグループ
- ワークスペース項目 (クエリー、レポート、グラフ、フォルダ)
- スキーマ リポジトリ

ユーザー データベース レコードの場合は、マスターシップ情報がフィールド値としてレコードに保管されます。ユーザーはマスターシップ フィールドの値を変更して、マスターシップを別のレプリカに転送することができます。レコードのマスターシップは、次の同期中に新しいマスター レプリカに送信されます。ほかのすべてのデータベース オブジェクトの場合は、管理者がマスターシップを変更する必要があります。

MultiSite 環境で作業する場合は、ワークフローを調節して、レコードや障害に合わせてマスターシップの変更が要求されるソフトウェア ライフサイクルのすべての段階で責任を担えるようにする必要があります。

たとえば、**Paris** のユーザーは、**Boston** の開発者が作業する必要がある障害を登録することができます。しかし、マスターシップを変更しないと、**Boston** の開発者は **Paris** のユーザーが入力した障害を修正できません。

マスターシップとワークフローの関係の詳細については、22 ページの「マスターシップ ポリシーの定義」を参照してください。

競合の解決

マスターシップの制限によって、複数のレプリカで発生する可能性のある変更の不整合の大部分が防止されますが、中には回避できないものもあります。名前の競合を避けるために、ファミリーの管理者は、オブジェクトの命名規則を作成して適用する必要があります。ユースモデルをサイト全体で一貫して適用すれば、競合の可能性が低下します。たとえば、ファミリーの管理者は次の規則に従います。

- すべての場所固有のオブジェクトに、場所の識別子を設定する必要があります。
- 複数のレプリカで使用されるすべてのオブジェクトを、同じレプリカで作成します。

名前の競合が発生すると、MultiSite では、名前の中に競合の発生元のレプリカ (キーサイト) の名前が表示されます。競合する名前が表示された場合は、直ちに競合するオブジェクトの名前を変更します。詳細については、102 ページの「名前の競合の解決」を参照してください。

操作ログ

ここでは、同期をサポートする仕組みについて説明します。この情報は MultiSite を使用する上で必須ではありません。しかし、「第 10 章 MultiSite の操作のトラブルシューティング」に記載されているエラー回復機能についての理解を深めるために役立ちます。

レプリカ データベースに対する変更の大部分は、項目として操作ログ (oplog) に記録されます。これらの項目には、変更を別のレプリカに反映するために必要な、以下のすべての情報が収められています。

- 変更が最初に行われたレプリカの識別情報
- 1 回のチェックアウト時に行われた、データベース レコードまたはスキーマ リポジトリのスキーマに対する変更。たとえば、新しいレコードの登録、スキーマの更新など。
- 整数のシーケンス番号: あるレプリカで最初に行われた変更分が 1、次の変更分が 2 という順で設定されます。これは、oplog 項目の oplog ID と呼ばれます。

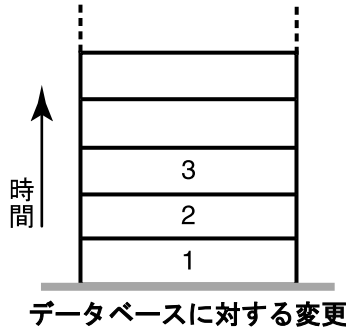
情報の正確な種類と量は個々の操作によって異なります。たとえば、新しいレコードの登録に関する oplog 項目は、既存のレコードの修正に関する oplog 項目とは内容が異なり、量も多くなります。

メモ: レプリカの oplog 項目は、ほかのレプリカを更新するために使用した後で削除することができます。詳細については、33 ページの「レプリカのパラメータのスクラブ」を参照してください。

各レプリカの操作の追跡

非レプリカ データベースに対する操作の履歴は、進展します (図 3 を参照してください)。

図 3 データベースに対する変更履歴

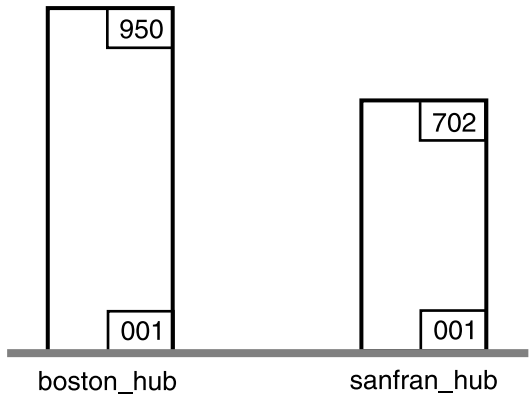


レプリカ ファミリ内部では、レプリカごとに変更が追跡されます。**oplog** 項目に操作対象の元のレプリカの識別情報が含まれるのはこのためです。したがって、レプリカ ファミリの履歴は **oplog** 項目のいくつかのスタックと考えることができます。各スタックは、対象レプリカで開始された操作の **oplog ID** の時系列順で表されます。

図 4 にファミリ内の 2 つのレプリカの状態を示します。

- レプリカ **boston_hub** では、**oplog ID** が 1 ～ 950 の操作が行われました。
- レプリカ **sanfran_hub** では、1 ～ 702 の操作が行われました。

図 4 ファミリの状態



あるレプリカはローカル操作に関してのみ、正確なデータを保持しています。更新パッケージを受け取るまで、あるレプリカでのほかのレプリカに関する情報は最新ではありません。たとえば、レプリカ **boston_hub** は 950 までのローカル操作を記録していますが、レプリカ **sanfran_hub** での操作については 504 までしか更新パッケージを受け取っていません。同様に、レプリカ **sanfran_hub** は 702 までのローカル操作を記録していますが、レプリカ **boston_hub** での操作については 791 までしか更新パッケージを受け取っていません。

図 5 に、この状況を示します。両方のレプリカとも、ほかのレプリカで行われた操作に関しては、最新の状況が反映されていません。

図 5 古いレプリカ

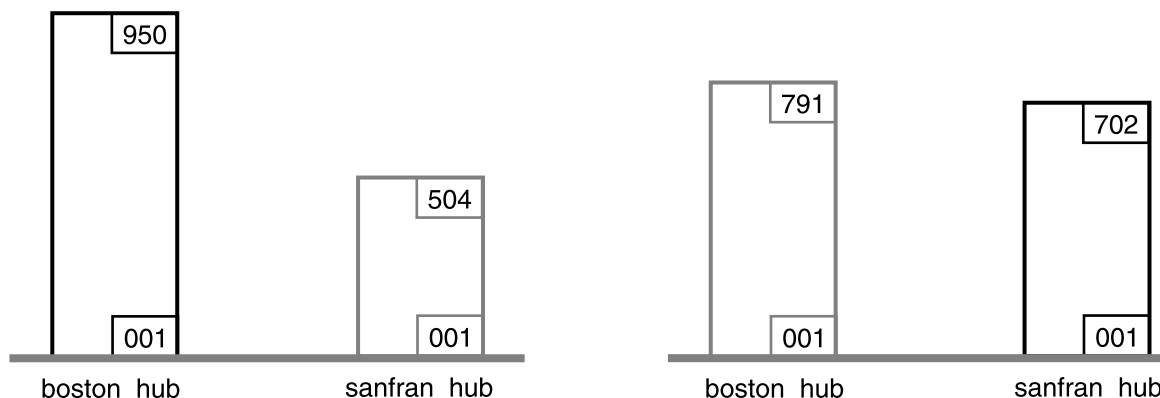


図 5 に示したように、レプリカ ファミリを一連の **oplog** のスタックとして描くと、同期プロセスを理解しやすくなります。たとえば、レプリカ **boston_hub** からレプリカ **sanfran_hub** に送信される更新パッケージは、レプリカ **boston_hub** に関するスタックの増分 (792 ~ 950) で構成されます。図 6 に 2 つの増分を示します。**sanfran_hub** では自身の状態を認識しているので、更新する必要があるのはほかのレプリカで開始された操作のみです (エラーから回復中の特定の状況では、レプリカのローカル操作に関するデータをリセットする必要があります。「第 10 章 MultiSite の操作のトラブルシューティング」を参照してください)。

図 6 2 つのレプリカ間での更新



メモ: 更新パッケージが **sanfran_hub** にインポートされるまでの間に、**boston_hub** ではさらに変更が行われている可能性があります。それらの変更は更新パッケージに含まれていません。

oplog ID とエポック数

エポック数は、特定のレプリカで開始された操作の総数です。図 4 では、`boston_hub` のエポック数が 950 になっています。

MultiSite の同期処理では、レプリカ間で転送されるデータ量を最小化しようとします。各レプリカでは、以下のエポック数を記録しています。

- **現在のレプリカに加えられた変更**: 現在のレプリカで開始された操作の数です。
- **現在のレプリカにインポートされたシブリング レプリカに加えられた変更**: `syncreplica` によって更新パケットから現在のレプリカに操作が書き込まれると、現在のレプリカにインポートされたシブリング レプリカで開始された操作の数を記録するエポック数が増分されます。
- **ほかのレプリカの状態の見積もり**: ほかの各レプリカに関する、そのレプリカとそのほかのレプリカ部の変更数の見積もりです。現在のレプリカは、ほかのレプリカに送信した操作を追跡し、これらの操作が正常にインポートされると想定します。

表 1 に、これらのエポック数がエポック数マトリックスにどのように保持されるかを示します。各レプリカにこのようなマトリックスが保持されます。各レプリカで操作するか、ほかのレプリカと更新パケットを交換すると、その内容が更新されます。

- `boston_hub` レプリカで作業が行われると、そのレプリカ自体のエポック数が増分されます。
- `boston_hub` レプリカが `sanfran_hub` レプリカから更新パケットを受け取ると、そのエポック数マトリックス内の自身の行 (`boston_hub`) と `sanfran_hub` の行が更新されます。
- `boston_hub` レプリカが `sanfran_hub` レプリカに送信する更新パケットを生成すると、そのエポック数マトリックス内の `sanfran_hub` の行が更新されます。

`syncreplica -export` コマンドは、エポック数をすぐに更新します。インポートするレプリカから更新パケットを受け取り正しく適用したという確認応答を待機しません。MultiSite の処理が正常に行われている間は、各レプリカのエポック数マトリックスの正確性を保持するために手動で処理する必要はありません。しかし、パケットを正常に適用できなかった場合には、手動による介入が必要になります。詳細については、101 ページの「失われたパケットの回復」を参照してください。

表 1 `boston_hub` レプリカにおける 2 行のエポック数マトリックス

	<code>boston_hub</code> で開始された操作	<code>sanfran_hub</code> で開始された操作
<code>boston_hub</code> 自体の状態のレコード	950	504
<code>sanfran_hub</code> の状態に対する <code>boston_hub</code> の見積もり	912	504

このマトリックスの内容は、**boston_hub** レプリカで **lsepoch** コマンドを使用して表示することができます。例を次に示します。

```
multiutil lsepoch -clan telecom -site boston_hub -family PRODA -user bostonadmin  
-password secret
```

```
Multiutil: サイト 'boston_hub' で再現された各サイトのエポック見積もり  
(@minuteman):
```

```
boston_hub: 950
```

```
sanfran_hub: 504
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり  
(@goldengate):
```

```
boston_hub: 912
```

```
sanfran_hub: 504
```

boston_hub で **syncreplica -export** コマンドを入力すると、このマトリックスが次のように使用されて、**sanfran_hub** 向けの更新パケットが生成されます。

- 1 **boston_hub** レプリカでは、ローカル操作の数が 950 (マトリックスの左上隅の数値) で、**sanfran_hub** レプリカが oplog ID 912 (マトリックスの左下隅の数値) までのすべての操作をインポートしたと見積もっています。
- 2 **boston_hub** レプリカが **sanfran_hub** レプリカに送信する更新パケットには、913 ~ 950 の **boston_hub** oplog 項目が含まれています。Boston の管理者が **syncreplica -export** を起動すると、**sanfran_hub** の行が更新されます。

```
multiutil lsepoch -clan telecom -site boston_hub -family PRODA -user lexadmin  
-password secret
```

```
Multiutil: サイト 'boston_hub' で再現された各サイトのエポック見積もり  
(@minuteman):
```

```
boston_hub: 950
```

```
sanfran_hub: 504
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり  
(@goldengate):
```

```
boston_hub: 950
```

```
sanfran_hub: 504
```

間接的な同期

1 つのファミリにレプリカが 3 つ以上ある場合、間接的に同期することがあります。あるレプリカから送信される更新パケットにそのレプリカ以外での変更が含まれることがあります。たとえば、**boston_hub** レプリカが **sanfran_hub** レプリカと **bangalore** レプリカとの間で更新パケットを交換する場合、以前に **sanfran_hub** から受け取った oplog 項目を **bangalore** に送信します。それらの項目によって、**sanfran_hub** での変更に関して、**bangalore** レプリカが最新の情報に更新されることもされないこともあります (**sanfran_hub** から **bangalore** に更新パケットを直接送信した場合、**bangalore** は最新状態に更新されます)。

メモ: あるレプリカが同じファミリのほかのレプリカから更新パッケージを直接受け取らない場合、それらのレプリカに関するエポック数マトリックスの値は0になる可能性があります。これはエラーではありません。

表 2 に、レプリカ **boston_hub** のエポック数マトリックスを示します。

表 2 boston_hub レプリカにおける 3 行のエポック数マトリックス

	boston_hub で 開始された操作	bangalore で開始 された操作	sanfran_hub で 開始された操作
boston_hub 自体の 状態のレコード	950	653	504
sanfran_hub の状態に 対する boston_hub の 見積もり	912	653	504
bangalore の状態に対する boston_hub の見積もり	709	653	221

このマトリックスの内容は、**lsepoch** コマンドを使用して表示することができます。例を次に示します。

```
multiutil lsepoch -clan telecomm -site boston_hub -family PRODA -user susan -password  
passwd
```

```
Multiutil: サイト 'boston_hub' で再現された各サイトのエポック見積もり  
(@minuteman):
```

```
boston_hub: 950  
sanfran_hub: 504  
bangalore: 653
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり  
(@goldengate):
```

```
boston_hub: 912  
sanfran_hub: 504  
bangalore: 653
```

```
Multiutil: サイト 'bangalore' で再現された各サイトのエポック見積もり  
(@ramohalli):
```

```
boston_hub: 709  
sanfran_hub: 221  
bangalore: 653
```

Boston サイトでは **sync replica -export** コマンドが、次のようにこのマトリックスを使用して、**bangalore** レプリカの更新をエクスポートします。

- 1 **boston_hub** レプリカでは、ローカル操作の数が 950 (マトリックスの左上隅の数値) で、**bangalore** レプリカが **oplog ID 709** (マトリックスの左下隅の数値) までのすべての操作をインポートしたと見積もっています。

- 2 **sanfran_hub** レプリカで開始された操作に関しては、**boston_hub** は oplog ID 504 までのすべての操作をインポートしており、**bangalore** が oplog ID 221 までのすべての操作をインポートしたと見積もっています。
- 3 **boston_hub** が **bangalore** に送信する更新パケットには、710 ～ 950 の **boston_hub** での操作と、222 ～ 504 の **sanfran_hub** での操作が含まれています。この時点で、**boston_hub** レプリカで実行される **lsepoch** コマンドの出力は次のようになります。

```
multiutil lsepoch -clan telecom -site boston_hub -family PRODA -user susan -password passwd
```

```
Multiutil: サイト 'boston_hub' で再現された各サイトのエポック見積もり (@minuteman):
```

```
boston_hub: 950  
sanfran_hub: 504  
bangalore: 653
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり (@goldengate):
```

```
boston_hub: 912  
sanfran_hub: 504  
bangalore: 653
```

```
Multiutil: サイト 'bangalore' で再現された各サイトのエポック見積もり (@ramohalli):
```

```
boston_hub: 950  
sanfran_hub: 504  
bangalore: 653
```

Rational ClearQuest MultiSite をインストールして使用する前に、実装計画を立てる必要があります。実装計画には次のものを含めます。

- MultiSite のインストール
- MultiSite のライセンス
- MultiSite ユース モデル
- MultiSite 管理者の管理責任

この章では、これらのことについて詳しく説明します。実装計画のドキュメントを作成し、開発環境を変更する前に、一連のテスト レプリカで試してみることをお勧めします。

サンプルの開発チェックリストについては、36 ページの「配置タスクの概要」を参照してください。

MultiSite のインストール

MultiSite のインストール手順の詳細については、インストレーション ガイドを参照してください。

各サイトには、パケット転送を処理する同期サーバーが必要です。このホストには、Rational Shipping Server をインストールしておく必要があります。各サイトには必ず、ClearQuest MultiSite 管理ツール (multiutil) をインストールした MultiSite 管理ホストが必要です。このホストを使用して multiutil コマンドを実行し、レプリカの同期と管理を行います。

ClearQuest クライアントとデータベースをホストする同期サーバー ホスト、管理ホスト、コンピュータはすべて、同じコード ページを使用する必要があります。データベース ファミリでのデータの破損を防止するため、データベース セットのデータ コード ページ値は、これらのコンピュータで使用されるコード ページに合わせて設定する必要があります。コード ページとデータ コード ページ値の設定の詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

SHIPPING サーバーである各ホストには、MultiSite の記憶ベイ ディレクトリ用に十分なディスク容量が必要です。記憶ベイには、MultiSite パケットと共に、パケットに対応する発送オーダー ファイルが格納されます。表 3 に、記憶ベイが配置されるディスク パーティションに必要なディスク容量を示します。

表 3 記憶ベ이에必要なディスク容量

パケットのタイプ	必要なディスク容量
レプリカの作成	最低でも、データベースとスキーマ リポジトリのサイズの 2 倍。パケットは、エクスポート元データベースの 4 倍になる可能性があります。
更新	Windows の場合、記憶ベ이에格納される最大パケットのサイズの 2 倍。2 倍のディスク容量が必要な理由は、記憶ベ이에同時に同じパケットのインスタンスが 2 つ存在する可能性があるからです。その 1 つはほかの宛先へ送信中のもので、もう 1 つは現在のホスト上のレプリカへの適用待ちのものです。
	UNIX の場合、記憶ベ이에格納される最大パケットのサイズ。

更新パケットのサイズを算出するための公式はありません。一般的に、同期を頻繁に行うほど、パケットのサイズも小さくなります。とはいえ、1 時間ごとに同期を実行したとしても、1 時間以内に大容量の開発アクティビティやリリース アクティビティが実行される可能性があります。そうなれば大きなサイズのパケットが生成されることになります。使用可能なディスク容量で、サイズの大きな不測のパケットに対処できるかどうかが不明な場合は、MultiSite を設定して、更新パケットのサイズに制限を設けることができます。詳細については、syncreplica のリファレンス ページを参照してください。

記憶ベいの指定方法の詳細については、shipping.conf (UNIX) と MultiSite コントロール パネル (Windows) のリファレンス ページを参照してください。

MultiSite のライセンス

レプリカのオブジェクトにアクセスする場合は、必ず MultiSite ライセンスが必要です。アクセスには、MultiSite のコマンドか GUI、ClearQuest のコマンドか GUI、または標準オペレーティング システムのコマンドを使用できます。MultiSite ライセンスの必要数は、レプリカ ClearQuest データベースにアクセスする開発者の数によって計算できます。すべての開発者がこれらの ClearQuest データベースにアクセスする場合は、ClearQuest ライセンスと同数の MultiSite ライセンスが必要です。レプリカ ClearQuest データベースにアクセスしない開発者がいる場合は、MultiSite ライセンスの数を減らすことができます。

たとえば、2 か所のサイトを持つ企業で、A サイトには 20 名の開発者、B サイトには 5 名の開発者がいるとします。A サイトには ClearQuest データベースが 3 つあり、そのうち 2 つは B サイトで複製し、残りの 1 つは複製しません。A サイトの開発者のうち 5 名は非レプリカ ClearQuest データベースにのみアクセスし、残りの 15 名はすべての ClearQuest データベース

を使用して作業します。このケースでは、この企業は次のライセンス数を購入する必要があります。

サイト	ClearQuest のライセンス数	MultiSite のライセンス数
A	20	15
B	5	5

メモ: この例では、開発者ごとに 1 つの ClearQuest ライセンスを購入することを前提としています。開発者数より少ない ClearQuest ライセンスを購入した場合は、同じ比率で MultiSite ライセンスも購入します。たとえば、企業が B サイト用に ClearQuest ライセンスを 3 つ購入した場合は、B サイト用の MultiSite ライセンスも 3 つ購入します。

ライセンスの取得とセットアップの詳細については、Rational ClearQuest のインストール情報を参照してください。

ClearCase と ClearQuest での SHIPPING サーバーの使用

Rational ClearCase MultiSite と Rational ClearQuest MultiSite の両方を使用する場合は、両方の製品を同じ SHIPPING サーバーで使用します。SHIPPING サーバーは、ClearCase MultiSite をインストールするときにインストールされます。

メモ: ClearQuest MultiSite と ClearCase、または ClearQuest MultiSite のみを使用する場合は、SHIPPING サーバーをインストールする必要があります。詳細については、Rational ClearQuest のインストール情報を参照してください。

ClearCase MultiSite と ClearQuest MultiSite の両方を使用する場合は、次の制限が適用されます。

- VOB レプリカ パッケージと ClearQuest データベースのレプリカ パッケージには、異なる記憶クラスを使用する必要があります。複数の記憶クラスを作成し、**-sclass** オプションを使用して特定クラスを指定できます。**-sclass** オプションを使用しない場合は、次のデフォルトクラスが使用されます。
 - ClearCase MultiSite の場合のデフォルト記憶クラスは、**-default** です。ClearCase MultiSite をインストールするときに作成されます。
 - ClearQuest MultiSite の場合、**-sclass** オプションを使用する **multiutil** コマンドのデフォルト記憶クラスは、**cq_default** です。**shipping_server** コマンドと **mkorder** コマンドでは、デフォルトクラスとして **-default** を使用します。

`cq_default` クラスはインストール時に作成されません。`cq_default` クラスを使用する場合は、このクラスと、このクラスの発送ベイと戻りベイを作成する必要があります。詳細については、`shipping.conf` (UNIX) と `MultiSite コントロール パネル` (Windows) のリファレンス ページを参照してください。

`cq_default` 記憶クラスを作成しない場合は、`ClearQuest MultiSite` で使用するための別のクラスを作成し、`multiutil` コマンドで `-sclass` オプションを使用して、作成した記憶クラスを指定する必要があります。`cq_default` 記憶クラスが存在せず、`multiutil` コマンドで `-sclass` オプションを指定しないと、`-default` クラスに関連付けられた記憶ベイにパケットが格納されることになり、インポート側のサイトで問題が発生する可能性があります。

- `ClearQuest MultiSite` 記憶クラスと `ClearCase MultiSite` 記憶クラスには別のベイを使用する必要があります。
- 一方の製品をアンインストールすると、もう一方の製品も動作を停止する可能性があります。先に両製品をアンインストールし、その後、使用する製品をインストールしてください。

`ClearCase MultiSite` と `ClearQuest MultiSite` の両方を使用する場合は、このガイドラインに従うことをお勧めします。

- `ClearQuest` レプリカのパケットをエクスポートする場合は、`-sclass` オプションを使用して記憶クラスを指定します。
- シッピング サーバーを操作するための電子メール通知機能を有効にし、発信元が `ClearQuest MultiSite` の操作になっているメッセージの場合にのみ使用するアドレスを指定します。詳細については、`control_panel` リファレンス ページを参照してください。

マスターシップ ポリシーの定義

`ClearQuest MultiSite` では、管理プロセスの変更を計画する場合にマスターシップ ポリシーについて考慮する必要があります。マスターシップにより、プロセスにおける別の制御の層が追加されます。

たとえば、レコードの状態が変わるときに、別のレプリカにレコードのマスターシップを割り当てることができます。また、レコードの状態とは関係なく、特定のタイプのすべてのレコードを特定のレプリカのマスターにすることも可能です。この場合は、それらのレコードの修正を、すべてマスター レプリカで実行する必要があります。

マスターシップは、プロセスに対していろいろな影響を与えます。たとえば、次のような影響があります。

- レコードまたはフィールドの値を修正するフックは、現在のレプリカがそのレコードのマスターである場合にのみ実行可能です。

- ユーザーとグループの修正は、そのユーザーまたはグループのマスターであるレプリカで実行する必要があります。
 - ワークスペース項目 (クエリー、レポート、レポート書式) の編集は、マスター レプリカで実行する必要があります。
 - スキーマの修正またはカスタマイズは、作業スキーマ リポジトリでのみ実行可能です。
- マスターシップの詳細については、「第 9 章 マスターシップの管理」を参照してください。

互換性に関する問題

ClearCase/ClearQuest UCM 統合を使用する場合は、この統合を必要としないコンピュータから **multiutil** を実行する必要があります。これは、**multiutil** には、UCM 統合ではサポートされない特別なデータベース セット名が必要なためです。

mkreplica -export を実行する前の cqintsrv プロセスの強制終了

ClearQuest Integration Server (cqintsrv) は、現在のセッションに関する情報をキャッシュします。これらのプロセスは、作業スキーマ リポジトリで最初の **mkreplica -export** コマンドを実行する前に終了させる必要があります。このプロセスが終了していないと、ClearCase の操作時にエラーメッセージが表示され、セッションが無効になったことが示されます。

ClearQuest UCM 統合におけるマスターシップ ポリシー

ClearQuest UCM を MultiSite と統合して使用する場合は、デフォルトの処理として、デリバー前にマスターシップのチェックが行われます。

ClearQuest MultiSite とほかの Rational 製品との統合

MultiSite を使用する環境で、RequisitePro、TestManager、Rational Administrator を統合して使用する場合は、制限があります。現在のレプリカが ClearQuest レコードのマスターであって、関連付けられた Rational Project レコードが同じレプリカのマスターでない場合、ClearQuest レコードに保存された統合情報を変更する (たとえば、[要求] タブで新しい要求を追加するなど) ことはできません。

MultiSite ユース モデル

次に、MultiSite ユース モデルのさまざまな面について説明します。

言語とコード ページのサポート

ClearQuest ユーザー データベースでは、入力するすべてのデータが、同じコード ページに属している必要があります。MultiSite 環境では、すべての言語が同じコード ページを使用するわけではないので、単一のコード ページに統一するのが困難になる場合もあります。たとえば、英語をはじめとする多くのヨーロッパ言語では 1252 コード ページを使用しますが、日本語で使用するのは 932 コード ページです。MultiSite の設定を行う前に、大多数のユーザーによって使用される言語を決定し、データベース セットのデータ コード ページ値を、使用言語のコード ページに設定する必要があります。

コード ページとデータ コード ページ値の設定の詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

複数コンピュータでの multiutil コマンドの実行

デフォルトでは、サイトあたり 1 台のコンピュータのみがスキーマ リポジトリとユーザー データベースを管理し、multiutil コマンドを使用するように設定されています。このコンピュータは、次の 2 つの方法で指定されます。

- multiutil activate を実行します。multiutil activate が実行されるコンピュータは、以降の multiutil コマンドを実行するように設定されます。
- mkreplica -import を実行します。multiutil mkreplica -import が実行されるコンピュータは、以降の multiutil コマンドを実行するように設定されます。

activate または mkreplica -import が実行されたコンピュータ以外のコンピュータから multiutil を実行する場合は、ユーザー サイトにあるスキーマ リポジトリ (データベース セット) にアクセスするようにコンピュータを設定する必要があります。

UNIX コンピュータを設定するには、cqreg add_dbset サブコマンドを使用します。このコマンドの詳細を表示するには、UNIX プロンプトで「man cqreg」と入力してください。

Windows コンピュータを設定するには、次のように installutil adddbset コマンドを使用します。

```
installutil adddbset dbset-name db-vendor server-hostname  
    { db-path lename.suffix | database-name }  
    ro-login-name ro-login-password connection-options
```

dbset-name は、スキーマ リポジトリの名前で、常に次の形式で指定されます。

CQMS.clan-name.site-name

接続オプションは次のとおりです。

```
Oracle データベース    HOST=host;SID=sid; server_ver=server-version-number;  
                        client_ver=client-version-number
```

```
その他すべて          ""
```

次の例に、`installutil adddbset` コマンドを使用して、telecomm 族の boston サイトにあるスキーマ リポジトリに接続する方法を示します。dbset-name は CQMS.TELECOMM.BOSTON です。

```
E:\Program Files\Rational\ClearQuest> installutil adddbset  
CQMS.TELECOMM.BOSTON ORACLE bar_host cquser cquser password client_ver=7.0
```

installutil とスキーマ リポジトリへの接続の詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

マスターシップ作成方針

作成する計画では、レコードと、ほかのオブジェクトのマスターであるレプリカを明確にする必要があります。オブジェクトのマスターシップは、ファミリにレプリカを作成した後で変更することができます。詳細については、8 ページの「独立した開発の実現：マスターシップ」と 81 ページの「データベース オブジェクトのマスターシップの変更」を参照してください。

マスターシップの変更は、標準の同期メカニズムにより、レプリカ間に伝達されます。ワークフローに応じ、オブジェクトによっては、マスターシップをより頻繁に変更することが必要になるものもあります。たとえば、レコードのマスターシップは、そのライフサイクル中に、レプリカ同士の間で数回転送される必要がある場合があります。

このようなマスターシップの変更を容易に実行するため、次の方法のいずれかを使用して、レコードに対するマスターシップの要求プロセスを合理化します。

- マスターシップの変更が必要になり、マスター レプリカの管理者にメッセージを送信する場合の、電子メール規則を記述します。
- ほかの管理者が ClearQuest Web を使用してユーザーのレプリカに対するアクセスを許可し、それらの管理者が必要に応じてログオンし、マスターシップフィールドを変更することができるようにします。
- マスター レプリカで作業する管理者に連絡し、マスターシップの変更を依頼します。

転送方法の同期

更新パケットとレプリカ作成パケットの転送は、いくつかの方法があります。選択する方法は、サイト間の接続形態、パケットを送付する速度、セキュリティの重要性に応じて異なります。詳細については、「第 5 章 転送方法の選択」を参照してください。

同期パターン

ファミリの同期パターンを使用して、更新パケットを交換するレプリカと、交換の方向を定義します。7 ページの図 2 に、一組のポイントツーポイント更新が関係した単純な同期パターンを示します。ただし、すべての更新をポイントツーポイントで行う必要はありません。それは、更新情報は蓄積されるためです。たとえば、3 つのレプリカの間で、次の更新が行われるとします。

更新 1: レプリカ 1 からレプリカ 2 に変更情報を送る。

更新 2: レプリカ 2 からレプリカ 3 に変更情報を送る。

レプリカ 1 からレプリカ 3 を直接更新する必要はありません。更新 2 には更新 1 からの変更情報も含まれているからです。この機能により、更新方針とパターンを柔軟に工夫することができます。効率を上げるために、単一の更新で複数のレプリカをターゲットにすることができます。たとえば、ファミリー内のほかのすべてのレプリカをターゲットにすることができます。

一般に、組織構造、通信/送付コストなどに応じて、任意のトポロジを構成することができます。図 7 に単純なピア ツー ピア型同期パターンを、図 8 に二重ハブ階層型パターンを示します。

図 7 ピア ツー ピア型同期パターン



図 8 階層型同期パターン



同期パターンは、次の要素を考慮して選択します。

- サイト間の帯域幅
- ネットワーク トポロジー
- 変更の待ち時間: 1 つのレプリカで行われた変更が、ファミリー内の別のレプリカに伝達されるまでの所要時間
- 障害の許容値

次に、一方向交換と両方向交換、および一般的な同期パターンについて説明します。

交換の方向

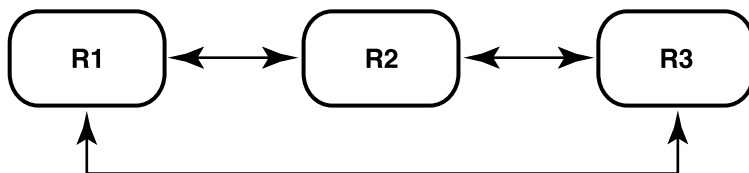
図 9 に示すように、同期処理は一方向と双方向で実行することができます。

図 9 一方向と双方向の更新

一方向



双方向



ほとんどの場合、双方向同期を使用します。一方向同期は、次のような場合に使用します。

- バックアップとしてレプリカを使用する。
- 読み取り専用で情報をほかのサイト (または企業) に提供する。
- セキュリティの高い開発プロジェクトで、オープン性の高いプロジェクトと同じデータを使用する。この場合、オープンなプロジェクトからセキュリティの高いプロジェクトには更新情報を送りますが、逆方向には更新情報を送しません。

一方向の更新には多少のリスクが伴います。たとえば、マスタースhipを間違えて変更した場合修正できません。内容が破損したレプリカと更新情報を直接交換していないレプリカから復元するには、余分な工数がかかります。読み取り専用のレプリカでは、どんな作業も誤って実行されることのないようにする必要があります。そのためには、フックの作成を行います。

1 対 1 型同期とリング型同期

図 10 1 対 1 型同期パターン

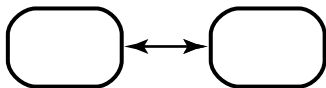


図 11 リング型同期パターン

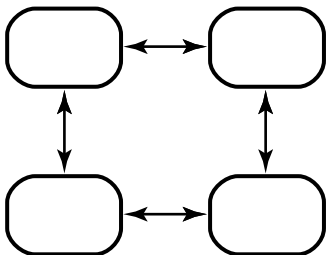


図 10 と図 11 に示す 1 対 1 型とリング型 (またはラウンドロビン型) の同期パターンは、レプリカが少数の場合に最適です。レプリカの数が増加すると、1 つのレプリカで生じた変更を、リングの反対側のレプリカが受け取るまでに要する時間も長くなります。

1 対多型同期

図 12 シングルハブ型同期パターン

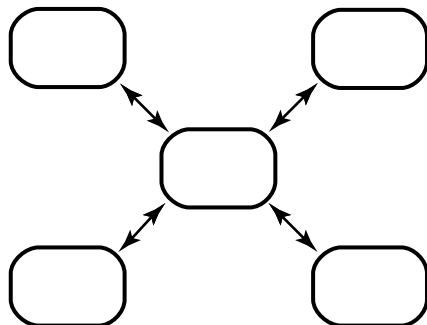


図 13 マルチハブ型同期パターン

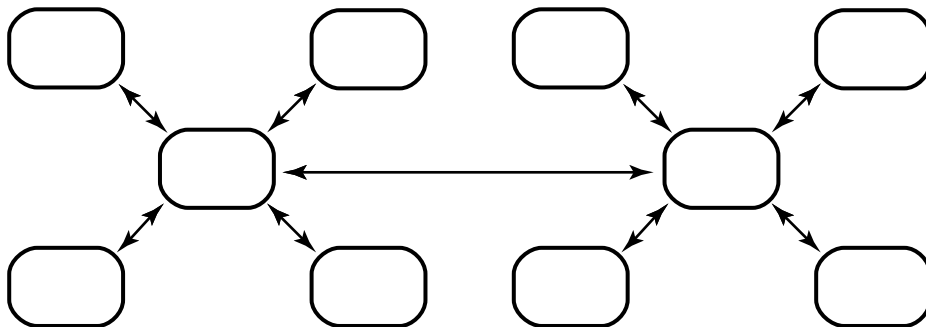
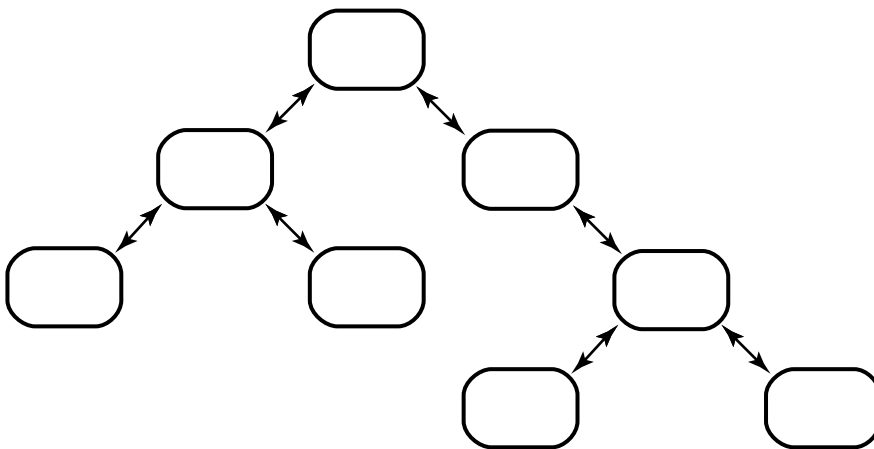


図 14 ツリー型同期パターン



ハブ型パターン (図 12 と 図 13) では、ハブ レプリカがすべてのスポーク レプリカとパケットを交換します。ツリー型パターン (図 14) では、ルート レプリカがブランチ レプリカとパケットを交換します。

利点:

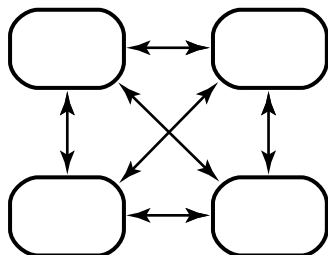
- ほかの 1 つのレプリカとのみパケットを送受信するスポーク レプリカとブランチ レプリカにとってより効率的です。

欠点:

- ハブまたはルート サイトが停止すると、すべてのスポーク / ブランチ サイトは、通信を継続するためにパターンを再設定する必要があります。
- 同期パターンを変更して、直接同期をとっていなかったレプリカがパケットを交換できるようにすると、生成される最初のパケットがシステムにとって大きすぎてしまうことがあります。

多対多型同期

図 15 多対多型同期パターン



多対多型同期パターン (図 15) では、各レプリカがほかのすべてのレプリカとパケットを交換します。

利点:

- サイトが少ない企業でこのパターンを使用すると、すべてのシブリングに対するレプリカのエポック テーブルが高精度で維持されます。
- 1 つのサイトが使用不能になっても、同期を継続するためにほかのサイトがパターンを変更する必要はありません。

欠点:

- 各管理者は、より多くの同期ジョブを管理し、パケットの追跡により多くの時間を費やす必要があります。

同期スケジュール

ファミリの同期スケジュールを使用して、ファミリ内のレプリカが更新パケットを送受信するタイミングを定義します。同期スケジュールは、各サイトでの開発の割合、サイト間の接続状況、MultiSite をバックアップ方針として使用するかどうかなど、多数の要因の影響を受けます。

同期方針を計画するときには、次の点を考慮してください。

- 開発の割合

同期のスケジュールを頻繁に作成すると、誤ってレプリカが削除された場合でも失われる作業が少なく済み、バックアップから復元することができます。変更箇所も少なくなるので、マージ処理も簡単になります。

同期処理がバックアップと重ならないようにする必要があります。

- 時差

更新情報を送るときまたは自動更新を設定するときには、時差を考慮に入れます。

図 16 に、複数のタイムゾーンにまたがるレプリカ同士の同期方法を示します。

- スキーマ リポジトリとユーザー データベースの両方に影響する変更

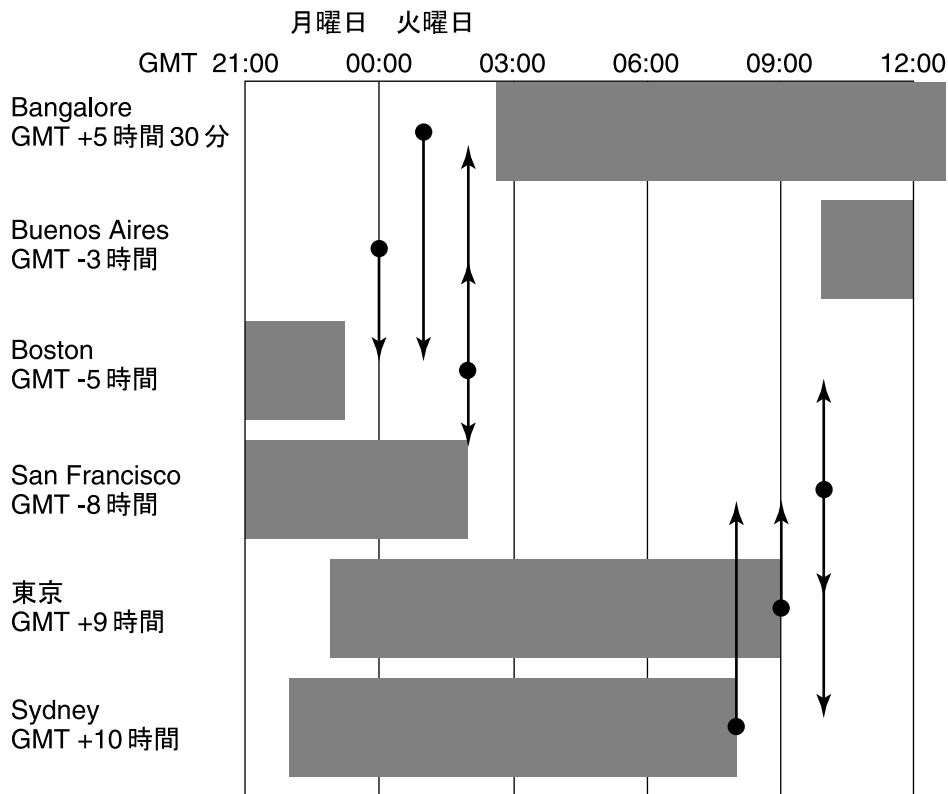
多くの変更がスキーマ リポジトリとユーザー データベースに記録され、両方の操作ログに **oplog** エントリが作成されます。最初にスキーマ リポジトリの同期をとり、次にユーザー データベースの同期をとることをお勧めします。

たとえば、図 8 に示すファミリの管理者は、次の決定を行います。

- 開発速度の速いハブ レプリカは、1 時間ごとに同期をとります。
- 各ハブ レプリカは、対応するスポーク レプリカと毎日同期をとります。各スポーク レプリカがハブ レプリカに更新パケットを送信し、その後、ハブ レプリカがスポーク レプリカに更新パケットを送り返します。これらの更新パケットのサイズは大きく、インポートに長い時間がかかると考えられるので、業務時間内やバックアップ処理の間は同期処理の実行を禁止します。
- すべてのレプリカ ホストは、受信されるとすぐに、受信ハンドラを使用してパケットをインポートします。

図 16 に、(1 時間ごとのハブ間更新ではなく) ハブ スポーク更新の同期スケジュールを示します。このスケジュールは時差を考慮して作成されており、各同期フェーズが次のフェーズの開始前に必ず終了するように、時間的な余裕も考慮されています。

図 16 同期スケジュール



キー

- = 作業時間
- バックアップ時間
- = エクスポート
- ▼ = インポート

データベースのバックアップ方針

すべてのサイトの製造元データベースに対して、定期的なバックアップの実行をお勧めします。データベース サーバー コンピュータが故障したり、データベースの記憶領域が少なくなった場合は、バックアップ コピーからデータベースを復元し、レプリカ復元手順を使用して失われた操作を元に戻すことができます。詳細については、107 ページの「データベース レプリカの復元」を参照してください。

レプリカのパラメータのスクラブ

コマンドを使用してレプリカを変更すると、レプリカの操作ログにエントリが記録されます。このメカニズムの詳細については、12 ページの「操作ログ」を参照してください。また、更新パッケージをエクスポートするときに、宛先の各レプリカごとに **export_sync** レコードが記録されます。**export_sync** レコードは、レプリカのエポック数マトリックスをリセットするために、**recoverpacket** コマンドによって使用されます。

ディスク容量とデータベース レコードを再利用するために、**oplog** エントリと **export_sync** レコードをスクラブすることができます。しかし、スクラブしたそれらのエントリとレコードを長期間にわたって保存しておく必要があります。レプリカの障害と更新パッケージの損失から確実に回復できるようにするためです。スクラブの頻度を設定するための指針を次に説明します。

oplog のスクラブ

oplog エントリは、長時間保持する必要があります。これらのエントリは、レプリカで更新パッケージを生成するときに必要になります。また **oplog** エントリは、災害などの障害からほかのレプリカを回復するために必要になる可能性があります。そのための **oplog** エントリを提供できるレプリカが存在しない場合は、回復対象のレプリカを作成し直す必要があります。108 ページの「バックアップからのレプリカの復元」を参照してください。同期をとるときにも **oplog** エントリを使用する必要があります。したがって、同期方針の中で **oplog** をスクラブする頻度を決定します。

デフォルトの設定では、**oplog** エントリがスクラブされることはありません。**VOB** ファミリ内での同期パターンを確立し、パッケージのエクスポートとインポートを正常に行えることを検証するまで、その設定を変更してはなりません。

レプリカの **oplog** を削除しても安全な場合は、次の処理を行います。

- 1 ほかの管理者と相談して、**oplog** エントリを保持する期間を決定します。

各レプリカでは、**restorereplica** 操作が正常に完了するために必要とされる期間、**oplog** エントリを保持する必要があります。**oplog** エントリをスクラブする頻度は、次の要因によって決まります。

- ファミリでレプリカ間の同期をとるパターン
- レプリカ間の同期をとる頻度

同期の頻度とは、更新パッケージをエクスポートする頻度と、ほかのレプリカで更新パッケージがインポートされる頻度の両方を指します。また、検証方法を設定して、ほかのサイトで更新パッケージを正常に処理できたことを確認してから、**oplog** をスクラブするようにします。

- レプリカをバックアップする頻度

たとえば、すべてのサイトで毎週レプリカ をバックアップし、2 週間前のバックアップの状態に戻せるようにする場合、各レプリカに対して 3 週間分の **oplog** エントリを保持する必要があります。レプリカの同期を週に 1 回とる場合、その週の更新パケットはまだほかのレプリカに送られていないと想定し、もう 1 週間分の **oplog** エントリを保持する必要があります。最後に、安全性に余裕を持たせるために、1 か月を追加します。結果として、スクラブに備えて **oplog** エントリを保持する期間は 2 か月になります。

- 2 レプリカの同期をとります。

- 3 サイズを縮小する **oplog** を持つレプリカで、**scruboplog** コマンドを実行します。次の例では、**sanfran_hub** サイトの (PRODA ファミリで指定した) ユーザー データベース レプリカの **oplog** をスクラブします。

```
multiutil scruboplog -clan telecommunications -site sanfran_hub -family PRODA -user  
sfadmin -password secret -before 31-Oct-2001
```

注意: 更新パケットに入れる前にレプリカの **oplog** エントリをスクラブすると、そのレプリカから更新パケットをエクスポートすることができません。これは重大なエラーで、ファミリー全体の整合性が損なわれます。

export_sync のスクラブ

通常の同期操作には **export_sync** レコードは必要ありません。エクスポート イベント レコードも同期に関するエクスポート情報を記録しますが、**export_sync** レコードとは異なります。

export_sync レコードは、レプリカのエポック数マトリックスをリセットするために、**recoverpacket** コマンドによって使用される日付ベースのレコードです。 (**-actual** または **lsepoch/chepoch** を使用するために) このパケット回復方法を使用しない場合は、それらのレコードを積極的にスクラブすることができます。 **recoverpacket** コマンドを使用する場合は、次のバックアップまでの間、**export_sync** レコードを保持する必要があります。 101 ページの「失われたパケットの回復」を参照してください。

export_sync レコードは、**oplog** エントリと同じ頻度でスクラブされます。

スペースを含むパス名の処理

Windows では、受信ハンドラまたは発送オーダーのパス名にスペースが含まれている場合、受信ハンドラまたは発送オーダーが格納されるファイル システムで DOS の「8.3 形式の短い名前」の解決機能を有効にする必要があります。このプロパティはデフォルトで有効です。このプロパティが有効でない場合、 SHIPPING サーバーは受信ハンドラを呼び出したり、発送オーダーを処理したりすることができません。

MultiSite 管理者の管理責任

MultiSite 管理者は、次の処理を行う必要があります。

- MultiSite のユース モデルの決定と実装を支援します。

新しいプロジェクトが設定されたときに、MultiSite 管理者はプロジェクト マネージャーと協力して、各種のオブジェクトのマスターとするレプリカを決定します。また管理者は、必要に応じたマスターシップの変更、データベース セットに対する適切なデータコード ページ値の決定を行う必要があります。

- MultiSite レプリカ作成と同期をモニターします。

MultiSite 管理者は記憶ベ이를チェックして、パケットが累積していないことを確認します。管理者の電子メール アドレスを、**shipping.conf** ファイル (UNIX) または MultiSite コントロール パネル (Windows) の **ADMINISTRATOR** エントリに記録します。

- システムのログ ファイルをモニターします。

エラー メッセージとステータス メッセージは、UNIX の場合は **shipping_server_log** ファイルに、Windows の場合はイベント ビューアに書き込まれます。

- 新しいバージョンの MultiSite と、新しいパッチをインストールします。

パッチと新しいバージョンに関する情報は、Rational の Web サイトから入手することができます。使用しているアーキテクチャに合わせて、必須のパッチと推奨されるパッチをインストールしてください。

MultiSite のバージョンの互換性については、Rational ClearQuest のインストールガイドを参照してください。

- ほかのすべての MultiSite 管理者と問題を調整します。

レプリカの初期の設定と同期を終えた後で、MultiSite 管理者は回復作業に関する調整も行う必要があります。そのために、更新パケット交換やマスターシップの変更が必要になることがあります。マスターシップを変更する場合、マスター レプリカを担当する MultiSite 管理者はオブジェクトのマスターになるレプリカにマスターシップを転送する必要があります。

MultiSite の配置図を作成し、ファミリーについての情報を記録することをお勧めします。表 4 に、情報の記録の例を示します。ファミリーの同期パターン図を作成することも有効です。

表 4 ファミリの情報

レプリカ名	レプリカ ホスト	管理者	電子メール、 電話番号	地域	時差
sanfran_hub	goldengate	John Cole	jcole、 x1462	San Francisco, CA, USA	GMT-8
boston_hub	minuteman	Susan Goechs	susan、 x3742	Boston, MA, USA	GMT-5
tokyo	shinjuku	Masako Ito	masako、 x7761	Tokyo, Japan	GMT+9
sydney	taronga	Bruce Fife	bfife、 x5080	Sydney, Australia	GMT+10
bangalore	ramohalli	Sonia Kumar	kumar、 x2347	Bangalore, India	GMT+5:30
buenosaires	mardelplata	Juan Fangio	fangio、 x4300	Buenos Aires, Argentina	GMT-3

- 各レプリカは、必要とされる特別な方法で確実に処理されるようにします。

バックアップからレプリカを復元する処理は重要なイベントです。108 ページの「バックアップからのレプリカの復元」に記載されている手順に従わないと、ファミリのレプリカ間に修復不能な不整合が発生する可能性があります。

レプリカのバックアップに対する特別な要件はありません。Rational ClearQuest の管理ガイドに記載されているバックアップの指示に従って操作してください。

配置タスクの概要

MultiSite を配置する前に、計画をチェックしてテストを行い、同期処理と障害追跡処理が有効に機能することを確認します。

- 配置を計画します。
 - MultiSite のマニュアルを確認します。
 - 適用される変更とポリシーについて記述した、MultiSite ワークフロー ドキュメントを作成します。

このドキュメントには、マスターシップポリシー、レプリカの作成方針、同期の方法、パターン、スケジュール、レプリカ環境でタスクを実行するユーザーのワークフローについて記述します。また、MultiSiteを実装するために必要な変更についても記述します。たとえば、スキーマへのマスターシップフィールドの追加、既存のフックの修正、マスターシップの転送を自動化するフックの作成などがあります。ユーザーコミュニティの代表者は、このドキュメントの内容を確認してください。

- c レプリカに対する適切なデータ コード ページ値を決定します。詳細については、『Rational ClearQuest 管理ガイド』を参照してください。
- d 実働データベースの検証とクリーンアップを行う必要があるかどうかを決定します。

2 配置をテストします。

- a テスト データベース インスタンスを使用して、テスト ベッド環境を設定します。詳細については、『Rational ClearQuest 管理ガイド』を参照してください。
- b テスト環境に MultiSite をインストールします。

ClearQuest MultiSite をインストールする場合は、ClearQuest を必ずアップグレードしてください。詳細については、ClearQuest のインストール情報を参照してください。

メモ : ClearQuest MultiSite の使用を開始する前に、スキーマ リポジトリに関連付けられたすべてのユーザー データベースを、ClearQuest と同じバージョンにアップグレードする必要があります。

- c バージョン 2003.06.00 より前の ClearQuest MultiSite を実行しているクライアントからデータベースにアクセスする場合は、CharacterSetValidation パッケージを適用します。詳細については、『Rational ClearQuest 管理ガイド』を参照してください。
- d スキーマ リポジトリのデータ コード ページ値を設定し、CharacterSetValidation パッケージを適用する場合は、ユーザー データベースのデータ コード ページ値を設定します。詳細については、『Rational ClearQuest 管理ガイド』を参照してください。
- e テスト データベースのレプリカの作成、同期の設定、スキーマへの必要な変更、プロセスのテストを行います。レプリカに対するバックアップと復元プロセスは必ずテストしてください。
- f テスト結果を検討し、必要があればワークフローを変更します。

3 実働データベースのレプリカを作成します。

- a レプリカの作成元のサイトで、MultiSite のライセンスをセットアップします。
- b 実働環境に MultiSite をインストールします。
- c データベースをバックアップします。
- d 以前のバージョンの ClearQuest MultiSite を実行しているクライアントからデータベースにアクセスする場合は、CharacterSetValidation パッケージを適用します。

- e スキーマの変更を実働データベースに適用します。
 - f 実働データベースをアップグレードします。
 - g スキーマ リポジトリのデータ コード ページ値を設定し、**CharacterSetValidation** パッケージを適用する場合は、ユーザー データベースのデータ コード ページ値を設定します。
 - h クライアント コンピュータをアップグレードします。
 - i 実働データベース セットをアクティブ化します。
 - j 実働データベース セットのレプリカを作成します。
 - k レプリカ データベースを使用して、**MultiSite** ユース ケースを検証します。
 - l 必要に応じてスキーマを変更し、データベースをアップグレードして同期をとります。
 - m レプリカの作成元レプリカから新しいレプリカへの一方向同期を設定します。同期スクリプトをテストし、同期が正常に実行されることを確認します。ユーザーが新しいレプリカにアクセスしないようにします。この時点で問題が発生しても、データを失わずに新しいレプリカを削除できます。
 - n 新しいサイトで、**MultiSite** のライセンスをセットアップします。新しいサイトの適切なコンピュータに **MultiSite** をインストールします。
 - o 新しいサイトのユース ケースを検証します。
 - p 実働レプリカ間での双方向同期を設定します。
 - q サイトの実働データベースを使用して、ユース ケースを検証します。
- 4 新しいレプリカを使用する前に、ユーザーに通知します。両方のサイトのユーザーにワークフロー ドキュメントを発行します。

これで、新しいワークフロー規則が有効になります。新しいデータベース レプリカの **Web** サーバーをセットアップして、ほかのサイトの使用者からもリモートアクセスできるようにすることをお勧めします。

この章では、MultiSite に関する情報を表示または変更するための MultiSite コマンドと API 関数の概要を示します。MultiSite のコマンドのリファレンス ページは「第 11 章 MultiSite リファレンス ページ」にあります。

multiutil の使用

multiutil コマンドは、レプリカに対して操作を実行するときに使用します。このコマンドは、次の機能を備えています。

- multitool コマンドには一連のサブコマンドがあります。それらのサブコマンドを使用して、レプリカの作成、同期、管理、オブジェクトのマスタースhip変更、障害からの回復などの機能を実行することができます。
- いくつかのサブコマンドとコマンド オプションは短縮可能です。詳細については、リファレンス ページを参照してください。
- multiutil はシングル コマンド モードで使用することができます。たとえば、次のように指定します。

multiutil lspacket

また、対話型モードで使用することもできます。

multiutil

```
multiutil> lspacket
```

```
multiutil> quit
```

- コマンド、オプションでは、大文字と小文字が区別されます。これらは必ず小文字で入力してください。
- help コマンド、-help オプションを使用すると、構文の要約が表示されます。

```
multiutil help chreplica
```

```
用法: chreplica [-cl.an name] [-site name]
```

```
...
```

```
multiutil chreplica -help
```

```
用法: chreplica [-cl.an name] [-site name]
```

```
...
```

- Windows 上では、man コマンドを使用すると、リファレンス ページが表示されます。

サブコマンドの説明

以下の項では、multiutil のさまざまな種類のサブコマンドについて説明します。

レプリカの作成、同期、管理に関するコマンド

表 5 の各コマンドで、レプリカの新規作成、特性変更、同期を行います。

表 5 レプリカの作成、同期、管理に関するコマンド

コマンド	説明
activate	データベース セットのレプリカの作成を準備します。
chreplica	レプリカのプロパティを変更します。
dumpoplog	レプリカの oplog の内容を表示します。
lspacket	mkreplica または syncreplica が作成した 1 つ以上のパケット ファイルを一覧表示します。
lsreplica	1 つ以上のレプリカを一覧表示します。
mkreplica	新規レプリカを作成します。
rmreplica	レプリカを削除します。
scruboplog	oplog 項目を削除します。
syncreplica	レプリカをそのファミリー内の 1 つ以上のレプリカと同期します。

オブジェクトのマスターシップに関するコマンド

複数のレプリカで競合する変更を取り込むことを避けるため、特定のオブジェクトにはマスター レプリカ (マスター) が割り当てられます。オブジェクトの最初のマスターは、オブジェクトが作成されたレプリカです。マスターシップの詳細については、8 ページの「独立した開発の実現: マスターシップ」を参照してください。マスターシップの管理に使用できるコマンドのリストを表 6 に示します。

表 6 オブジェクトのマスターシップに関するコマンド

コマンド	説明
chmaster	オブジェクトのマスターシップを転送します。
describe	オブジェクトのマスター レプリカを表示します。

障害回復に関するコマンド

各レプリカは、エポック数マトリックスを使用してそのレプリカ自身の状態とその他すべてのレプリカの状態を追跡します(レプリカは常に変化しているため、そのレプリカ自体に対する変更は認識していますが、ほかのレプリカの状態については推定にすぎません)。レプリカは、更新パケットを送信するたびに、そのレプリカ自体のエポック数マトリックスを更新します。ただし、この更新では、パケットが宛先にデリバリーされ該当するレプリカに適用されることが前提になっています。詳細については、12 ページの「操作ログ」を参照してください。

パケットが正常にデリバリーされなかった場合は、表 7 の障害回復コマンドを使用します。

表 7 障害回復に関するコマンド

コマンド	説明
chepoch	レプリカのエポック数マトリックスを変更します。
lsepoch	レプリカのエポック数マトリックスを一覧表示します。
recoverpacket	レプリカのエポック数マトリックスをリセットし、送信されなかったパケットを再送します(パケットが失われた場合や使用不能になった場合に必要です)。
restorereplica	バックアップからレプリカを復元します。このコマンドはレプリカを特別な状態に設定します。この状態では、レプリカはエポック数マトリックスの訂正内容をほかのレプリカに送信します。レプリカが通常の開発作業で利用できるのは、ほかのレプリカの現在の状態を通知する特別な更新内容を受け取ってからです。

multiutil ユーティリティのコマンド

multiutil の次のコマンドは、その他の作業を実行します。

表 8 multiutil ユーティリティのコマンド

コマンド	説明
cd	現在の作業ディレクトリを変更します。
exit	multiutil の対話型セッションを終了します。
help	multiutil のコマンドの構文を表示します。
man	Windows 上で、リファレンス ページを表示します。UNIX 上では、コマンドの構文を表示します。
quit	multiutil の対話型セッションを終了します。

MultiSite のほかのコマンド

multiutil のサブコマンドではない、MultiSite のコマンドのリストを表 9 に示します。これらのコマンドは、Rational ClearCase のインストールディレクトリの下にあります。

表 9 MultiSite のほかのコマンド

コマンド	ccase-home-dir 以下の場所	説明
mkorder	etc (UNIX) bin (Windows)	保存および転送の際の発送オーダーを設定します。
notify	bin	保存および転送のための電子メール プログラム。
shipping_server	etc (UNIX) bin (Windows)	保存および転送パケット転送サーバー。

MultiSite の API 関数

フックまたは外部アプリケーションで API 関数を使用すると、作業しているデータベースがレプリカであるかどうか、また、現在のレプリカが、変更するレコードまたはオブジェクトのマスターであるかどうかを知ることができます。

表 10 に、MultiSite で使用できる 3 種類の API メソッドを示します。すべての API コマンドが網羅されたリストが必要な場合は、Rational ClearQuest の API リファレンスを参照してください。

表 10 MultiSite の API 関数

API メソッド	関連するオブジェクト	機能
SiteHasMastership	エンティティ ワークスペース ユーザー	レコード、ワークスペース項目、ユーザーのマスターがどのレプリカであるかを示す値を返します。
GetSiteExtendedName	エンティティ ワークスペース ユーザー	ratl_keysite 名の値を返します。これにより、どのレコード、ユーザー、グループの名前が重複し、どの名前を変更しなければならないかを知ることができます。

表 10 MultiSite の API 関数

API メソッド	関連するオブジェクト	機能
GetLocalReplica	セッション オブジェクト	レプリカに関する情報を一覧表示します。このメソッドを使用すると、作業しているデータベースがレプリカであるかどうかを知ることができます。

コマンドでのレプリカの指定

multiutil のコマンドでレプリカを指定するとき、場合によってはサイト、ファミリー、族の指定が必要になることがあります。サイト内に族が 1 つしかない場合、**-clan** 引数は省略可能です。レプリカを作成する場合を除き、**-site** 引数も省略可能です。

たとえば、次のコマンドは telecomm 族の一部である PRODA ファミリの boston_hub レプリカを指定しています。

```
multiutil lsreplica -clan telecomm -site boston_hub -family PRODA -user susan -password passwd
```


この章では、レプリカ間でパケットを転送する方法について説明します。転送方法は、レプリカ間の接続に応じて選ぶ必要があります。レプリカが IP 接続できない場合は、ファイルベースの方法を選択する必要があります。IP 接続が可能な場合は、Rational ClearQuest MultiSite の保存および転送機能を使用できます。

表 11 に、さまざまな状況での推奨方法を示します。

表 11 パケット送付方法の選択肢

サイトの状況	お勧めする方法
サイト間が高速回線で接続されている。	保存および転送
1 つ以上のサイトにファイアウォールが設置されている。	ファイルベースの方法 (電子メール、ftp、物理メディア)、保存および転送
パケットを迅速に転送する必要がある。	ファイルベースの方法 (電子メール、ftp)、保存および転送
サイト間は電子的に接続されていない。	ファイルベースの方法 (物理メディア)

ファイルベースの方法

この転送方法では、電子メール、ftp、物理メディア (CD、磁気テープ、ディスクなど) を使用します。

電子メールの使用法

パケットの送付手段として既存の電子メール メカニズムを使用することができます。送信側でパケットを圧縮してエンコードし、そのデータを受信側の特定のメール アドレスに送信します。受信側では、受け取った情報をデコードして圧縮を解除するスクリプトに、メールを転送します。メッセージが大きくなりすぎて電子メールで送信できなくなることがないように、最大サイズを指定してパケットを生成することができます。これには `-maxsize` オプション、`shipping.conf` ファイル (UNIX)、MultiSite コントロール パネル (Windows) のいずれかを使用します。

利点:

- 送付の仕組みがよく理解され、広く利用されています。
- システム管理者の労力をほとんど必要としません。

欠点:

- データの転送経路を制御できません。
- 何の知らせもなくメッセージが傍受されたり失われる可能性があります。
- **ftp** または保存および転送よりも効率が低くなります。

メモ:

- 電子メールでの送付を自動化するスクリプトを作成することができます。送信用のスクリプトで、パケットを生成し、圧縮してエンコードし、電子メールで処理できる大きさのパケットに分割し、順序付けします。次に、パケットを宛先の場所またはレプリカのアドレスに送信します。

受信側では、パケットを受け取ったアカウントは後処理を行うプロセスにパケットをリダイレクトまたは送信します。後処理のプロセスでは、分割されたパケットを連結してデコードし、圧縮を解除した後で、レプリカの記憶ベイに格納します。

MultiSite のインポート コマンドはパケットの順序の乱れや欠落の問題に対処しています。したがって、スクリプトでその問題を取り扱う必要はありません。

- **ssh** と **scp** (安全なシェルおよび安全なコピー) を使用すると、高いセキュリティを保った状態でファイアウォールを通過してファイルを移送することができます。
- セキュリティ上の配慮から、パケットは暗号化する必要があります。

FTP の使用法

ftp ユーティリティを使用してレプリカ間でパケットを転送できます。送信側で、**MultiSite** 管理者またはスクリプトがパケットを作成して圧縮し、ほかのサイトの **MultiSite** 管理者がアクセスできる場所へ **ftp** を使用してファイルを転送します。受信側のスクリプトは、この転送先サイトをポーリングして、新しいファイルを探します。新しいファイルを受信したら、スクリプトは **ftp** を使用してファイルを取得し、展開して、処理します。

利点:

- 送付の仕組みがよく理解され、広く利用されています。
- 電子メールよりも信頼性が高く効率的です。

欠点:

- 転送先サイトを使用する必要があります。
- 転送先サイトをポーリングする必要があります。
- **ftp** ユーティリティでの対話が必要なため、実装方法が複雑になります。
- 第 3 のシステム (転送先サイト) を使用するため、管理工数が増加します。

物理メディアの使用法

パケットをファイルにして CD、磁気テープ、またはディスクに書き込み、それを別のサイトに送付します。mkreplica コマンドと syncreplica コマンドには -out オプションが用意されており、これを使用してパケットを物理ファイルにすることができます。

送付にファイルベースの方法を使用する場合は、-maxsize オプションを指定して、ファイルのサイズが所定の上限を超えないようにしてください。

保存および転送

MultiSite の保存および転送機能 (SHIPPING サーバー) は、レプリカ作成と同期の送付フェーズを自動化するためのファイル転送サービスです。この機能により、各種サイズのパケットの処理と一連の MultiSite ホストを経由したファイル転送 (一度に 1 つのホップ) が実現し、データ通信エラーに対処することができます。保存および転送プロセスの動作原理を次に示します。

- 1 エクスポート フェーズ中、パケット ファイルと発送オーダー ファイルが作成されます。発送オーダー ファイルには送信に関する指示が記録されています。
- 2 パケットと発送オーダーは、ClearQuest データベース レプリカに関連付けられている同期サーバー上にある記憶ベイ ディレクトリの 1 つに格納されます。

パケットが記憶クラスに関連付けられている場合は、そのパケットは記憶クラスによって指定されている記憶ベイに格納されます。記憶クラスは、UNIX では shipping.conf ファイルで、Windows では MultiSite コントロール パネルで、それぞれ定義できます。

- 3 SHIPPING サーバーは、発送オーダーの指示に従って、ローカル サイトの記憶ベイから別のサイトのホスト上の対応するベイへ、パケット ファイルを転送します。
- 4 必要であれば、受信ホストの SHIPPING サーバーはそのパケットを次の宛先に送信します。

パケットのディレクトリ

各記憶クラスには、記憶ベイと戻りベイがあります。これは、パケットを保持するためのディレクトリです。記憶ベイは通常の発送操作に、戻りベイはパケットを正常にデリバリーできなかった場合に、それぞれ使用されます。

記憶ベイと戻りベイには、それぞれに incoming と outgoing の 2 つのサブディレクトリがあり、パケットとそれに対応する発送オーダー ファイルが保持されます。発送操作では、これらのディレクトリでパケットを検索します。

メモ: Windows の場合、ベイがあるディスク パーティションには、格納しうる最大パケットサイズの少なくとも 2 倍のディスク空き容量が必要です。その理由は、発送ベイ内に同時に同じパケットのインスタンスが 2 つ存在する可能性があるからです。1 つはほかの宛先へ送信中のもので、もう 1 つは現在のホスト上のレプリカへの適用待ちのものです。

Rational Shipping Server をホストにインストールすると、記憶ベイと戻りベイと共に **-default** 記憶クラスが作成されます。記憶ベイの名前は **ms_ship** に、戻りベイの名前は **ms_rtn** になります。各ベイには、**incoming** と **outgoing** の両ディレクトリが作成されます。**MultiSite** コントロール パネル (**Windows**) を使用して新しい記憶ベイまたは戻りベイを作成すると、ベイと共にそのサブディレクトリが作成されます。**UNIX** では、目的のベイとその **incoming** と **outgoing** の両サブディレクトリを作成し、次に作成したベイを **shipping.conf** ファイルに指定する必要があります。

パケットの送付

手作業または自動で明示的にコマンドを実行すると、送信側のホスト上で SHIPPING サーバーが起動します。SHIPPING サーバー プロセスは受信側のホスト上の **albd_server** プロセスと通信します。すると、**albd_server** プロセスは受信側のホスト上で SHIPPING サーバーを受信モードで起動します。送信側と受信側でそれぞれ呼び出された SHIPPING サーバー間で **TCP/IP** 接続が確立された後で、ファイルが転送されます。

保存および転送の問題

以下の項では、保存および転送機能を使用する場合に検討すべき問題点について説明します。

レプリカ ホスト間通信

ホスト同士が相互に通信できる必要があります。ネットワークでホスト名を使用する場合、送信側のホストが宛先のホスト名から **IP** アドレスを解決できる必要があります。そのためには、**hosts** ファイル、**hosts NIS** マップ、またはドメイン ネーム サービスを更新する必要があります。**TCP/IP** アクセスを確認するには、送信側の各ホストで **rcp** を使用してファイルを受信側のホストにコピーするか、保存および転送を使用してパケットを送信します (49 ページの「保存および転送機能へのパケットの引き渡し」を参照してください)。

メモ: ネットワーク内でホストが **IP** アドレスだけで認識される場合、ホスト名の代わりに **IP** アドレスを使用します。ホスト名から **IP** アドレスを解決する必要はありません。

パケット サイズの制限

mkreplica コマンド と **syncreplica** コマンドは、システムでサポートされるサイズよりも大きいパケットを作成しようとする場合と失敗します。この問題を回避して信頼性を高めるには、**-maxsize** オプションを使用して、パケットを複数のパケットに分割します。

```
multiutil mkreplica -export -maxsize 1g ...
```

```
multiutil syncreplica -export -maxsize 500m ...
```

shipping.conf ファイル (**UNIX**) か **MultiSite** コントロール パネル (**Windows**) でも、最大パケット サイズを指定できます。

デフォルトのパケット サイズの制限については、mkreplica リファレンス ページを参照してください。

保存および転送機能の設定

保存および転送機能の設定はホストに固有です。記憶ベイと戻りベイの場所、マルチホップのパケット転送をサポートするためのルーティング情報、転送エラーが発生した場合の処理、受信ハンドラなどを指定することができます。

保存および転送機能を使用する前に、ディスクに十分な空き容量があることを確認し、shipping.conf ファイルまたは MultiSite コントロール パネルを設定し、パケットの記憶クラスを作成してください。

設定方法の詳細については、UNIX では shipping.conf のリファレンス ページ、Windows では MultiSite コントロール パネルのリファレンス ページを参照してください。

保存および転送機能へのパケットの引き渡し

レプリカ作成パケットまたは更新パケットを生成するときに、保存および転送機能を使用してパケットを転送するように指定することができます。syncreplica と mkreplica では、共に以下のオプションを使用できます。

- **-fship** オプションを指定すると、パケット ファイルと発送オーダー ファイルがホストの記憶ベイの 1 つに置かれ、 SHIPPING サーバーが起動して、パケット ファイルが宛先のホストまたはルート上の中間のホストに送信されます。
- **-ship** オプションを指定すると、パケット ファイルと発送オーダー ファイルがホストの記憶ベイに置かれますが、SHIPPING サーバーは起動しません。パケット ファイルは、SHIPPING サーバーによりベイが次回にポーリングされたときに送信されます。

記憶クラスを使用してのパケットの区別

保存および転送機能では、さまざまな方法でパケットを処理するよう設定できます。各パケットを記憶クラスに割り当て、各記憶クラスに独自の記憶ベイ、戻りベイ、有効期限を持たせることができます。

メモ: UNIX では 1 つの記憶クラスに複数の記憶ベイと戻りベイを割り当てることができます。この場合、SHIPPING サーバーはパケットのサイズを使用して、記憶ベイを選択します。逆に、複数の記憶クラスで 1 つ以上のベイを共有することもできます。

スキーマ リポジトリやユーザー データベース レプリカからのパケットのデフォルトの記憶クラスは、使用するコマンドによって異なります。mkorder コマンドと shipping_server コマンドでは、**-default** 記憶クラスを使用します。このクラスは、Rational Shipping Server のインストール時に作成されます。**-sclass** 引数を使用するすべての multiutil コマンドは **cq_default** 記憶クラスを使用します。このクラスは、インストール時には作成されません。

複数の記憶クラスを使用して、さまざまな族に属するレプリカに関するパケットを区分することができます。記憶ベイ ディレクトリに対するオペレーティング システムの権限を調整することにより、無許可で使用されないようにパケットを保護することができます。また、保存および転送機能を使用してサイト間で非 **MultiSite** ファイルを転送する場合、別の記憶クラスを使用することができます。

保存および転送機能を使用してパケットを **VOB** レプリカや **ClearQuest** データベース レプリカから転送する場合は、別の記憶クラスを使用する必要があります。**mkorder** コマンドと **shipping_server** コマンドは、**ClearCase MultiSite** と **ClearQuest MultiSite** の両方で使用するので、**ClearQuest** レプリカからのパケットに対してこれらのコマンドを使用する場合は記憶クラスを指定する必要があります。また、**cq_default** 記憶クラスを作成しない場合は、**multiutil** コマンドで **-sclass** オプションを使用して **ClearQuest MultiSite** 記憶クラスを指定する必要があります。

記憶クラスを作成するときには次のガイドラインに従ってください。

- 記憶ベイは一意である必要があります。**VOB** レプリカからのパケットに対して使用する名前またはディレクトリと同じものを使用しないでください。
- 指定するディレクトリが存在するパーティションには、パケットを格納するために十分な空き容量が必要です。
- 記憶クラス名では、大文字と小文字が区別されます。記憶クラス名には小文字のみを使用するか、すべての記憶クラスに適用する大文字小文字の規則を規定しておくことをお勧めします。

間接発送ルートの設定

パケットの発送オーダーには、パケットの最終宛先のホスト名または経由するいくつかのホスト名を指定します。デフォルトでは、保存および転送機能により、パケットは宛先のホストに直接送信されます。必ず中間のホストを経由してパケットを送信するように指定するには、パケットを **shipping.conf** ファイル (UNIX) または **MultiSite** コントロール パネル (Windows) でルーティング ホップと関連付けます。

例を次に示します。

- UNIX ホスト上で、**shipping.conf** ファイルに次の行を含めます。
`ROUTE sydney_fw sanfran_hub boston_hub tokyo`
- Windows ホスト上で、**MultiSite** コントロール パネルの [ルーティング情報] セクションの [次のルーティング ホップ] ボックスに **sydney_fw** を、[宛先ホスト名] ボックスに **sanfran_hub**、**boston_hub**、**tokyo** をそれぞれ指定します。

最終宛先ホストが **sanfran_hub**、**boston_hub**、**tokyo** のいずれかであるパケットは、ホスト **sydney_fw** に送られます。この時点で、ローカル ホストの役目は終了し、パケットを中継する責任は **sydney_fw** に移ります。ホスト **sydney_fw** は、**shipping.conf** ファイルまたは **MultiSite** コントロール パネルの設定に応じて、パケットを最終宛先まで直接送信するか、別の中間ホストを経由して送信します。

メモ: マルチホップ転送の場合、送信元ホストで **-fship** オプションを指定すると、最初のホップへの送信は直ちに実行されます。以降のホップへの送信は、中間のホストで SHIPPING サーバーが起動したときに行われますが、中間のホスト上でパケットが受信された直後とはかぎりません。

再試行、有効期限、戻りデータ

SHIPPING サーバーは別ホストへのパケットの転送を 1 回試行します。パケットが転送されなかった場合 (たとえば、受信側のホストが使用不可能な場合)、SHIPPING サーバーはエラーメッセージとログ ファイル エントリを生成し、処理を終了します。管理者は再試行する頻度を設定することができます。

- パケットが正常に転送された場合は、SHIPPING サーバーはパケットとその発送オーダーを削除します。異常終了した場合は、パケットと発送オーダーは記憶ベイに残されます。
- **shipping_server -poll** を実行すると、1 つまたは複数の記憶ベイ内にあるすべてのパケットが転送されます。したがって、転送が正常に終了せずに残っているパケットがあると、(転送可能であれば) **shipping_server -poll** が次回に起動したときに送信されます。

shipping_server コマンドを繰り返し起動することで、正常に送信されなかったパケットの再送信をいつまでも継続することができます。しかし、通常は再試行を続けるのではなく、管理者が問題を解決します。そのために、各発送オーダーに有効期限を指定することができます。具体的には、以下の方法があります。

- コマンド オプション **-pexpire**
- (UNIX) 送信側のホストの **shipping.conf** ファイル内の **EXPIRATION** エントリ
- (Windows) 送信側のホストの **MultiSite** コントロール パネルで指定されている [パケットの有効期限] の値

デフォルトでは、発送オーダーは作成後 14 日で有効期限が切れます。

SHIPPING サーバーは、有効期限が切れた発送オーダーを検出すると、対応するパケットを宛先に転送せず、代わりに以下の処理を行います。

- 発送オーダーを変更して、パケットを送信元のホストに戻します。戻されたパケットは戻りベイに置かれます。
- 送信元のホストの 1 つ以上のアドレスに電子メールでメッセージを送信します (戻されたパケットが送信元のホストに届くと、さらに別のメッセージが送信されます)。

パケットが戻される過程でも複数のホップを経由する可能性があります。50 ページの「間接発送ルートの設定」で説明した場合と同様です。戻される途中のパケットは中間の各ホスト上の戻りベイに置かれます。各ホップでは、**shipping_server -poll** により、記憶ベイの処理に加えて戻りベイの処理が行われます。14 日の有効期限内に戻されないパケットは削除されます。

ホストにアクセスできない場合のタイムアウト時間の設定

SHIPPING サーバーは、宛先ホストにパケットの送信を試み、ホストにアクセスできないと判断すると、`/var/adm/rational/clearcase/shipping/ms_downhost` ディレクトリ (UNIX) または `ccase-home-dir¥var¥shipping¥ms_downhost` ディレクトリ (Windows) にファイルを作成します。ファイル名は、アクセスできなかったホストの名前になります。

次のいずれかのパラメータが指定された場合は、SHIPPING サーバーはそれ以降の発送操作時に宛先ホストのディレクトリをチェックします。

- (Windows) MultiSite コントロールパネルの [接続不可能なホストに対する時間制限 (分)] の値
- (UNIX) `shipping.conf` ファイルの `DOWNHOST-TIMEOUT` 設定、または `SHP_DOWNHOST_TIMEOUT_RETRY` 環境変数 (両方のパラメータが指定されている場合、SHIPPING サーバーは `DOWNHOST-TIMEOUT` を使用します)。

宛先ホストが `ms_downhost` ディレクトリに見つかり、現在の時刻とファイルが最後に変更された時刻との差がSHIPPING サーバー ホストのタイムアウト設定より短い場合は、SHIPPING サーバーは宛先ホストへのパケットの送信を試行しません。この差がタイムアウト設定以上だった場合は、SHIPPING サーバーは宛先ホストへのパケットの送信を試行します。タイムアウトが設定されていない場合は、SHIPPING サーバーは宛先ホストへのパケットの送信を試行します (アクセスできなかったホストへのパケットの送信試行には 30 秒ほどかかります)。

混在環境でのエラーの通知

電子メールによる通知を有効にしていない Windows ホストを経由してパケットが送信された場合、そこでエラーが発生すると、電子メールによる通知メッセージは送られません。この場合、イベント ログにメッセージが書き込まれます。メッセージには、適切なユーザーに障害を知らせるように指示が書かれています。電子メールでの通知を有効にする方法については、MultiSite コントロール パネルのリファレンス ページを参照してください。

パケット以外のファイルの送信

`mkorder` ユーティリティを使用して発送オーダーを作成したファイルは、保存および転送機能を使用して送信することができます。ファイルを即座に送信することも、SHIPPING サーバーによる送信待ちにすることも可能です。

- ファイルを即座に送信するには、`mkorder` コマンドに `-fship` オプションを指定します。
`/opt/rational/clearcase/etc/mkorder -data /usr/rptgen/brdcst.0702 -fship
-copy boston_hub tokyo`

- ファイルを送送ペイに保管して、 SHIPPING サーバーの次回の実行で送信するには、
-ship オプションを指定します。

```
/opt/rational/clearcase/etc/mkorder -data /usr/rptgen/brdcst.0702 -ship  
-copy boston_hub tokyo
```

メモ: 送送オーダー ファイルは送送対象のファイルと同じディレクトリに置く必要があります。

mkorder コマンドを実行した後は、元のファイルを削除してもかまいません。

同じ名前のファイルが受信ホストに既にある場合、送送したファイルは **filename_1** のように名前が変更されます。さらに別のファイルを同じ名前で送送した場合は **filename_2** になり、以後も同じように変更されます。

ファイアウォールを通過する保存および転送の使用法

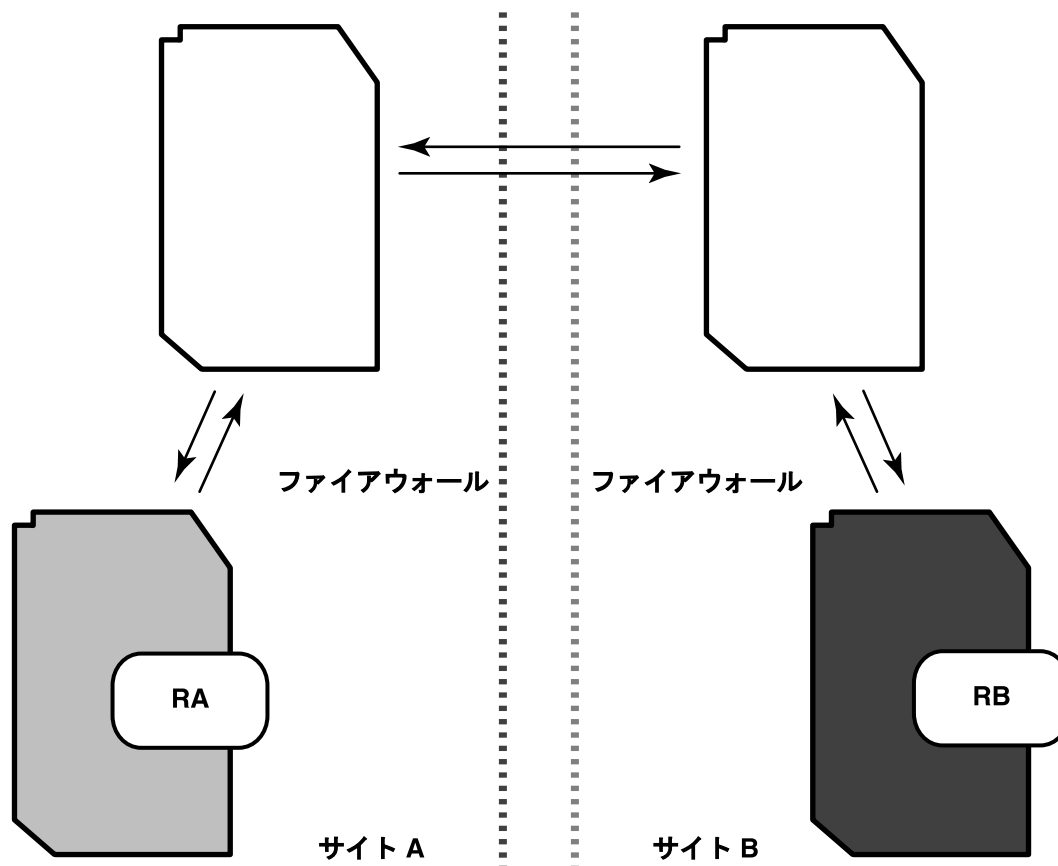
デフォルトでは、保存および転送機能 (SHIPPING サーバー) を使用してファイアウォールを通過することはできません。ファイアウォールを通過できるようにするには、通常は、特定の IP アドレスに特定のポートへのアクセスを許可する必要があります。SHIPPING サーバーは送送と受信のレプリカ ホストで利用可能な任意のポート番号を選択して接続に使用することができるので、特別の許可が与えられるポート番号を 1 つ (または少数) に限定することはできません。

ファイアウォールを使用しているサイトでは、「公開ホスト」を指定できます。公開ホストとは、ファイアウォールを通過して通信を行うよう設定し、SHIPPING サーバー ソフトウェアをインストールするホストです。SHIPPING サーバーをサイトの同期サーバーに設定し、パケットを公開ホストに送送すると、公開ホスト上の SHIPPING サーバーがそのパケットをファイアウォールの反対側のホストに転送します。公開ホストで高いセキュリティを維持するためには、SHIPPING サーバーが使用できるポート番号の範囲を指定する必要があります。

メモ: サイトのセキュリティ強化のために、SHIPPING サーバーを公開ホストにインストールするのはほかに適した転送方法がない場合のみにしてください。これ以外の方法については、45 ページの「ファイルベースの方法」を参照してください。

図 17 に、公開ホスト構成の例を示します。公開ホスト間では、ファイアウォールを通過して通信が行われます。保存および転送ソフトウェアは両方にインストールされていますが、ClearQuest ソフトウェアはどちらにもインストールされていません。

図 17 保存および転送の構成



ファイアウォールに関する考慮事項

公開ホストに SHIPPING サーバーをインストールするときは、以下のことを考慮してください。

- 送信ベイがデータでいっぱいになる可能性があります。

公開ホスト上で SHIPPING サーバーを使用すると、ローカル ネットワーク上の SHIPPING サーバーを利用できる任意のコンピュータの送信ベイを、誰でもネットワーク経由でいっぱいにすることができます。ディスクがいっぱいになることや、それに関連する問題を避けるためには、以下の措置を取ってください。

 - ローカル ネットワーク内のすべての送信ベイを独自のパーティション上に配置します。送受信ベイがいっぱいになっても、システムのパフォーマンスを低下させないためです。
- 必要なコンピュータ上にのみ SHIPPING サーバーをインストールします。つまり、同期サーバーと管理者が使用するコンピュータにのみインストールします。パケットが盗み見される危険があります。

通常の更新パケットでは、情報はエンコードされていません。したがって、セキュリティの低いネットワークを通じて発送するパケットは、暗号化する必要があります。また、更新パケットのフォーマットはあまり複雑ではありません。専門のプログラマであれば、フォーマットを解明して、スキーマリポジトリまたはユーザーデータベースを破壊する処理を組み入れたパケットを作成することも可能です。データを暗号化すれば、その種の攻撃を受けにくくなります。

ファイアウォールの設定によるアクセス制限

プログラムが接続できるポートと、ファイアウォールへのアクセスを許可する IP アドレスを指定することをお勧めします。許可されるポート番号と IP アドレスを制限することにより、権限のないコンピュータによりファイアウォールが破られる可能性が低減します。

公開ホストでアクセスを許可する必要があるポートは次のとおりです。

- TCP ポート 371 (albd_server ポート)
- 環境変数 CLEARCASE_MIN_PORT と CLEARCASE_MAX_PORT で指定したポートの範囲 (55 ページの「albd_server と shipping_server で使用するポートの制御」を参照)

自分のサイトの公開ホストにファイアウォール経由でパケットを転送するホストの IP アドレスには、ファイアウォール経由のアクセスを許可する必要があります。

ファイアウォールの設定の詳細については、ファイアウォールのマニュアルを参照してください。

公開ホストへの SHIPPING サーバーのインストール

UNIX の場合、ClearCase 製品ファミリのインストーラで、SHIPPING サーバー ソフトウェアだけをインストールするオプションを選択することができます。『ClearCase 製品ファミリ インストレーション ガイド』に記載されている指示に従って、[ClearCase MultiSite Shipping Server-only Installation] オプションを選択してください。公開ホストには ClearCase をインストールしないでください。

Windows の場合は、Rational Shipping Server インストール オプションを使用してください。

albd_server と shipping_server で使用するポートの制御

環境変数の CLEARCASE_MIN_PORT と CLEARCASE_MAX_PORT を使用して、albd_server と SHIPPING サーバーが通信の目的で割り当てることのできるポート番号の範囲を指定します。SHIPPING サーバーがポート番号を割り当てる際には、CLEARCASE_MIN_PORT の値から CLEARCASE_MAX_PORT の値までの番号を使用します。この範囲内のポートを割り当てられない場合、SHIPPING サーバーは一時スリープし、その後で再び割り当てを試みます。

ポート環境変数が設定されていることを検知すると、送信側のホストの SHIPPING サーバーは TCP を使用して、受信側のホスト上の `albd_server` と接続を試みます。この接続に失敗すると、SHIPPING サーバーは UDP を試みます。したがって、TCP による接続が可能な場合、公開ホスト上で UDP を使用可能にしたり、UDP ポートを開設する必要はありません。

個別の SHIPPING サーバーを実行するのに、同時に 3 つ以上のポートが必要になることはありません。複数の要求を送るとき、SHIPPING サーバーは子プロセスを生成します。それぞれの子プロセスが個々の要求を処理します。SHIPPING サーバーが起動する子プロセスの数は最大で 10 です (10 個起動するのは同時に処理する要求が 10 個ある場合のみ)。したがって、必要なポートの範囲は最大で 20 です。それよりも範囲が小さい場合、アクセスがエラーになることがあります、後で再試行できます。

ポート値の指定

`CLEARCASE_MIN_PORT` の値の範囲は 1024 から 65534 ままで、`CLEARCASE_MAX_PORT` の値の範囲は 1025 から 65535 までです。`CLEARCASE_MAX_PORT` の値は `CLEARCASE_MIN_PORT` の値よりも大きい必要があります。

メモ: ダイナミック/プライベート ポート範囲である 49152 から 65535 の範囲を使用することをお勧めします。

UNIX 上でポート値を指定するには、以下の場所に環境変数の `CLEARCASE_MIN_PORT` と `CLEARCASE_MAX_PORT` を設定します。

- 公開ホスト上の `shipping.conf` ファイル。詳細については、`shipping.conf` リファレンス ページを参照してください。
- 公開ホスト上の `clearcase` スクリプトについて、次の操作を行います。
 - a ファイル `ccase-home-dir/etc/clearcase` を編集します。
 - b 以下の行を追加します。その際、`min-port` と `max-port` を、ポート番号の最小値と最大値で置き換えます。`albd_server` で始まるセクションの前にこれらの行を置く必要があります。

```
#
# ポート番号の最小値と最大値を設定する
#
CLEARCASE_MIN_PORT=min-port
CLEARCASE_MAX_PORT=max-port
export CLEARCASE_MIN_PORT
export CLEARCASE_MAX_PORT
```

Windows でポートの最小値と最大値を指定するには

- 1 公開ホスト上で [コントロール パネル] を開き、[システム] アイコンをクリックします。
- 2 CLEARCASE_MIN_PORT と CLEARCASE_MAX_PORT の 2 つの環境変数を作成し、その値を指定します。

ファイアウォールを通過する保存および転送を使用するためのチェックリスト

このチェックリストは、ファイアウォールを通過して保存および転送を使用する場合の手順をまとめたものです。

- 1 シッピング サーバーが使用できるポート番号の範囲と、サイトの公開ホストにパケットを送信するホストの IP アドレスを決定します。
- 2 ファイアウォールを設定して、使用を許可するポート番号と IP アドレスを制限します。決定したポート番号の範囲のほかに、TCP ポート 371 へのアクセスを許可する必要があることに注意してください。
- 3 シッピング サーバー ソフトウェアを公開ホストにインストールします。
- 4 環境変数 CLEARCASE_MIN_PORT と CLEARCASE_MAX_PORT を設定します。
- 5 サイトの各レプリカ サーバーで、別のサイトにパケットを送信する場合の次のホップのホストとして、公開ホストを指定します。たとえば、企業内に 3 つのサイト (SiteA、SiteB、SiteC) があり、それぞれに 1 台ずつある公開サーバーではシッピング サーバー (SSA、SSB、SSC) が実行されており、レプリカ サーバー ホストが 3 つあるとします。

UNIX の場合は、`shipping.conf` ファイルを編集して `ROUTE` オプションを追加します。たとえば、SiteA の各レプリカ サーバー ホストでは、次のように編集します。

```
ROUTE SSA SiteB_host1 SiteB_host2 SiteB_host3 SiteC_host1
SiteC_host2 SiteC_host3
```

Windows の場合は、MultiSite コントロール パネルを開いて、[ルーティング情報] セクションに適切な値を設定します。たとえば、SiteA の各レプリカ サーバーでは、[次のルーティング ホップ] を SSA に、[宛先ホスト名] を SiteB_host1、SiteB_host2、SiteB_host3、SiteC_host1、SiteC_host2、SiteC_host3 に設定します。

- 6 公開ホストでは、`shipping.conf` ファイルを編集して、`ROUTE` オプションを追加してパケットの次の宛先を指定します。

SiteA の公開ホストでは、手順 5 の例と同様に、次の `ROUTE` オプションを `shipping.conf` ファイルに追加します。

```
ROUTE SSB SiteB_host1 SiteB_host2 SiteB_host3
ROUTE SSC SiteC_host1 SiteC_host2 SiteC_host3
```

SiteB の公開ホストでは、次の **ROUTE** オプションを **shipping.conf** ファイルに追加します。

```
ROUTE SSA SiteA_host1 SiteA_host2 SiteA_host3
```

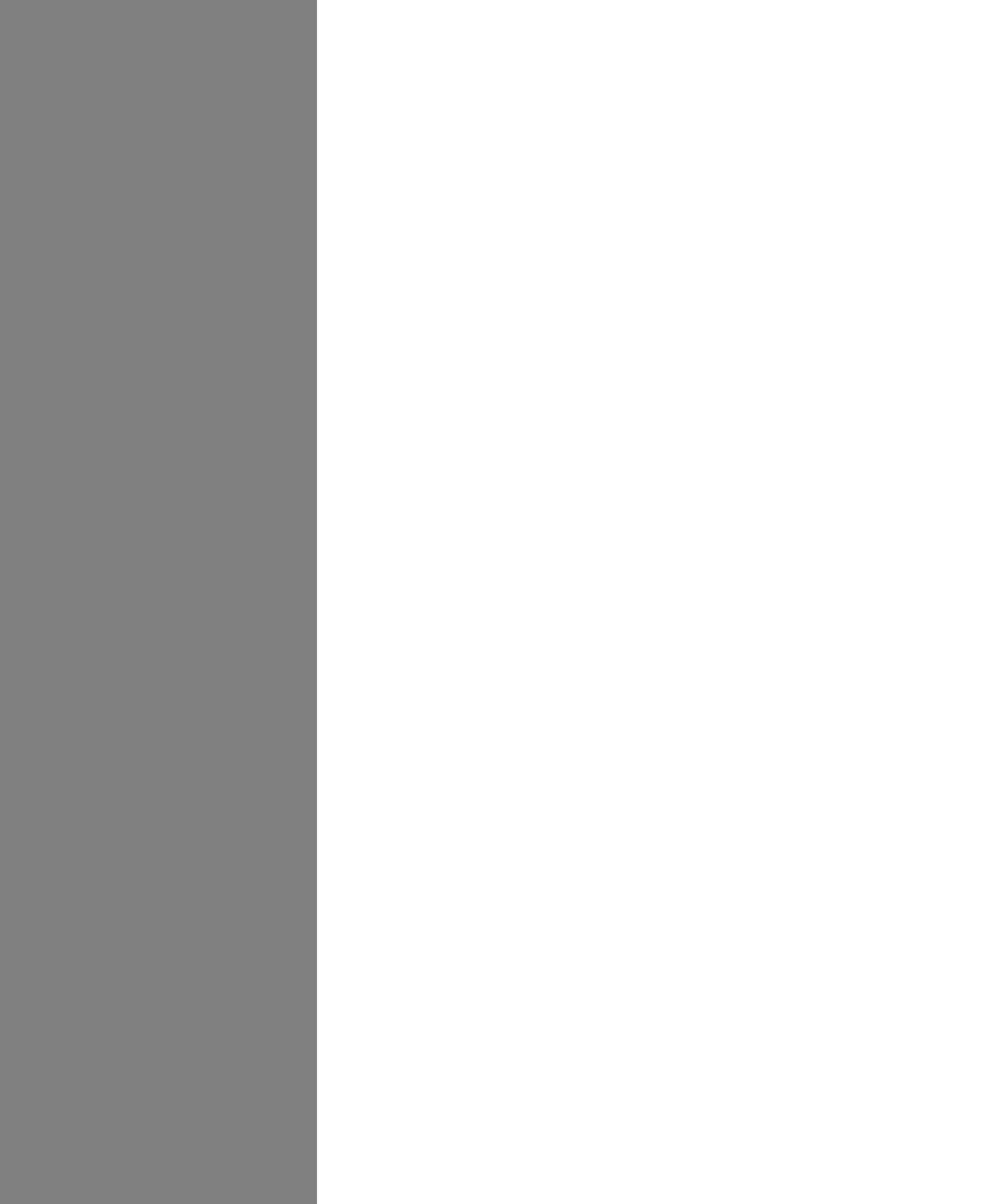
```
ROUTE SSC SiteC_host1 SiteC_host2 SiteC_host3
```

SiteC の公開ホストでは、次の **ROUTE** オプションを **shipping.conf** ファイルに追加します。

```
ROUTE SSA SiteA_host1 SiteA_host2 SiteA_host3
```

```
ROUTE SSB SiteB_host1 SiteB_host2 SiteB_host3
```

レプリカ作成と同期



データベース レプリカの作成

6

この章では、データベース レプリカの計画方法と作成方法について説明します。レプリカを作成する前に、マスターシップと、パケットのデリバー方法について決定する必要があります。「第3章 MultiSite の実装の計画」を参照してください。

レプリカ作成の概要

レプリカ作成作業は、以下のフェーズで構成されます。

- 1 保存および転送の設定: 保存および転送を使用する場合は、エクスポートする側のレプリカとインポートする側のレプリカで **shipping.conf** ファイル (UNIX) または [MultiSite Configuration 情報] (Windows) を設定する必要があります。49 ページの「保存および転送機能の設定」を参照してください。
- 2 アクティブ化: 初めてデータベース セットのレプリカを作成する場合は、そのデータベース セットをアクティブにする必要があります。
- 3 エクスポート: **mkreplica -export** コマンドを入力します。このコマンドは、新しいレプリカオブジェクトとレプリカ作成パケットを作成します。
- 4 転送: レプリカ作成パケットを 1 つ以上のほかのサイトに送信します。
- 5 データベース作成: 新しいレプリカを作成する場所に、スキーマ リポジトリと各ユーザーデータベース レプリカに使用する空の製造元データベースを作成します。
- 6 インポート: 新しいレプリカを作成する場所に、**mkreplica -import** を使用してレプリカ作成パケットをインポートします。

この手順は、すべてのパケット デリバー方法とすべてのプラットフォームで同じです。

以下の項では、各フェーズについて詳しく説明します。

データベースのアクティブ化

データベースの最初のレプリカを作成する前に、データベース セット (スキーマ リポジトリと、それに関連付けられたユーザー データベース) をアクティブにする必要があります。データベース セットをアクティブにした後は、何度でもレプリカを作成できます。データベース セットをアクティブにする必要があるのは一度だけです。

データベース セットをアクティブにする際は、そのデータベース セットに族名とサイト名を割り当て、その同期サーバ (パケットを処理するホスト) を指定します。

データベース セットのアクティブ化には **activate** コマンドを使用します。詳細については、**activate** のリファレンス ページを参照してください。

レプリカ作成パケットのエクスポート

レプリカの作成は、レプリカ作成パケットを作成し、データベース レプリカをホストするサイトにそれを送信することによって行います。これには、**mkreplica -export** コマンドを使用します。**mkreplica -export** の使用法の詳細については、**mkreplica** のリファレンス ページを参照してください。

レプリカ作成のエクスポート フェーズにおいて、レプリカ作成コマンドはデータベースのコピー中、そのデータベースをロックします。データベースはコマンド実行が終了するまでロックされます。ログインすることはできません。

データベースに対して **mkreplica -export** コマンドを使用する前に行うことは、次のとおりです。

- データベースにログインしているユーザーがいないことを確認する。ユーザーがデータベースにログインしたままの状態 で **mkreplica -export** を実行すると、そのユーザーが加えた変更は、データベースがアンロックされた後の変更もすべて失われます。
- データベースをアップグレードしてスキーマの最新バージョンを使用する。
- データ コード ページ値が、レプリカ作成パケットをインポートするサイトで使用されるコード ページに設定されていることを確認する。コード ページに互換性がない場合、パケットのインポートは失敗します。データ コード ページ値の詳細については、『**Rational ClearQuest 管理ガイド**』を参照してください。

パケットの作成に要する時間はデータベースのサイズによって変わり、データベースのコピーまたはバックアップ手順の実行に要する時間の倍かかることもあります。したがって、レプリカ作成のエクスポート フェーズは業務時間外にスケジュールする必要があります。また、エクスポート フェーズの時間中にスケジュールされているバックアップがあれば、そのバックアップをキャンセルする必要があります。

この例では、**PRODA** ユーザー データベースとそのスキーマ リポジトリのレプリカを作成します。新しいサイトは **sanfran_hub** という名前で、**goldengate** という同期サーバを使用します。この例では、レプリカ作成パケットの発送に保存および転送を使用します。

```
multiutil mkreplica -export -clan telecom -site boston_hub -family PRODA -user susan  
-password passwd -maxsize 50m -fship -workdir c:\temp\packets -sclass cq_default  
goldengate:sanfran_hub
```

空の製造元データベースの作成

レプリカ作成パッケージをインポートする前に、レプリカ データを格納するための空の製造元データベースを作成する必要があります。製造元データベースの作成方法と MultiSite でサポートされているデータベースのリストについては、Rational ClearQuest のインストールに関する情報を参照してください。

メモ: データの破損を防止するために、製造元データベースのコード ページ (文字セット) は、使用しているデータベース セットのデータ コード ページ値と同じでなければなりません。コードページとデータ コード ページ値の詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

注意: データベース レプリカを受信する前に ClearQuest データベースを作成しないでください。ClearQuest データベースは、空の製造元データベースにレプリカ パッケージをインポートすると作成されます。ClearQuest データベースをメンテナンス ツールまたは ClearQuest Designer で作成した場合、レプリカのインポートは失敗します。

レプリカ作成パッケージのインポート

レプリカ作成パッケージをインポートする際は、既存の製造元データベースに未加工データをインポートします。レプリカ作成パッケージには、データベースが格納されるのではなく、データベースを形成するメタデータとレコード データが格納されます。パッケージは、特定のデータベース製造元に限って使用できるというわけではありません。サポートされているすべての製造元データベースからスキーマ リポジトリまたはユーザー データベースの作成に使用できます。

レプリカをインポートする際は、次の点に留意してください。

- サイト名はエクスポート コマンドに使用した名前を使用しなければなりません。レプリカ作成パッケージのインポート時にサイト名を変更することはできません。
- インポートする側のサイトで使用しているコード ページと、インポートされるレプリカのデータ コード ページ値とが一致しない場合、インポートは失敗します。
- 初めの時点では、multiutil コマンドによりレプリカにアクセスすることができるのは、mkreplica -import コマンドを最初に実行したコンピュータのみです。以後、別のコンピュータから multiutil コマンドを実行するには、multiutil によってレプリカにアクセスするようにそのコンピュータを構成する必要があります。24 ページの「複数コンピュータでの multiutil コマンドの実行」を参照してください。

レプリカ作成パッケージをインポートするには、mkreplica -import コマンドを実行して、作成した空の製造元データベースにレプリカ データをインポートします。スキーマ リポジトリとユーザー データベースの両方に、データベース パラメータとログイン情報を入力します。

新しいレプリカの追加

既存のサイトに新しいユーザー データベース レプリカを追加する場合は、そのスキーマ リポジトリに対して製造元データベースを作成する必要はありません。新しいレプリカが同じ族になる場合、`mkreplica -import` コマンドは新しいレプリカをそのサイト内の既存スキーマ リポジトリに関連付けます。`mkreplica` のリファレンス ページを参照してください。

インポートが失敗した場合の回復

`mkreplica -import` プロセスが何らかの理由で中断するか失敗した場合は、次の手順に従います。

- 1 いつインポートが失敗したかを確認します。これは、`mkreplica -import` によって生成されるエラー メッセージで確認することができます。
- 2 インポートが失敗した製造元データベースを削除し、新しい製造元データベースを作成します。
 - インポートが失敗したのがスキーマ リポジトリのインポート中である場合は、そのスキーマ リポジトリの製造元データベースを削除し、新しい製造元データベースを作成します。
 - インポートが失敗したのが、スキーマ リポジトリのインポートが正常に終了した後である場合は、そのユーザー データベース レプリカに対応する製造元データベースを削除し、新しい製造元データベースを作成します。
- 3 `mkreplica -import` を再実行します。

レプリカ作成のシナリオ

この項の例では、ソフトウェア開発が **Boston** で行われ、新しい開発オフィスが **San Francisco** にあるものとします。現在、新しいリリースについての作業が始まろうとしています。

2 つのサイトの関連特性

地域	同期サーバ	レプリカ名 (サイト名)
Boston	minuteman	boston_hub
San Francisco	goldengate	sanfran_hub

前提条件

新しいレプリカを作成する前に、元のサイトで次の手順を実行する必要があります。

- 1 MultiSite のライセンスがインストールされていることを確認します。

データベース セットに対して **activate** コマンドを入力すると、元のデータベースのユーザーは、(ClearQuest ライセンスと) MultiSite ライセンスがない限り、データベースにアクセスできなくなります。

- 2 特定のデータベースのレプリカを初めて作成する場合は、すべてのユーザーがデータベースからログオフする必要があります。

レプリカのエクスポート開始後、**mkreplica -export** コマンドによってデータベースがロックされます。すべてのユーザーは、レプリカ作成の手順が開始される前にログオフし、終了後に再ログオンする必要があります。レプリカ作成プロセスの実行中にセッションが開いたままになっている場合、そのセッションのデータは失われます。

- 3 ClearCase/ClearQuest UCM 統合を使用している場合は、スキーマ リポジトリに対して最初の **mkreplica -export** を実行する前に、すべての **cqintsrv** プロセスを終了させる必要があります。

- 4 ユーザー データベースとスキーマ リポジトリのサイズを決定します。

レプリカ作成パッケージは、各データベースの 4 倍のサイズになることがあります。使用する作業ディレクトリに十分な空き容量があることを確認してください。作業ディレクトリに対しては書き込み権限が必要です。また、指定するディレクトリは存在していないものでなければなりません。

データベース セットのアクティブ化

次のコマンドにより、**Boston** のデータベース セットをアクティブ化します。このコマンドは族名を **telecomm**、サイト名を **boston_hub** とし、同期サーバとして **minuteman** を指定します。

```
multiutil activate -user susan -password passwd -clan telecomm -site boston_hub -host minuteman
```

エクスポート フェーズ

Boston では、次の手順を実行します。

- 1 **mkreplica -export** コマンドを使用して **San Francisco** 用のレプリカを作成します。

次のコマンドにより、**telecomm** 族の **PRODA** ユーザー データベースのレプリカを **sanfran_hub** という名前で作成します。また、**telecomm** 族のスキーマ リポジトリのレプリカも、**sanfran_hub** という名前で作成されます。新しいサイトの同期サーバは **goldengate** です。管理者は **Rational Shipping Server** により、**-fship** オプションを使用して直ちにパッケージを送信します。

```
multiutil mkreplica -export -clan telecomm -site boston_hub -family PRODA -user susan  
-password passwd -maxsize 50m -fship -workdir c:¥temp¥packets -sclass cq_default  
goldengate:sanfran_hub
```

- 2 新しくレプリカが作成されたデータベースをバックアップします。

このバックアップには、データベース レプリカが作成されたという事実が記録されます。データベース レプリカを作成する前にとったバックアップ コピーからデータベースを復元しようとしても、レプリカ復元プロシージャは失敗します (**restorereplica** コマンドは正常に終了する可能性があります、ほかのレプリカから更新パケットをインポートすることはできません。元のデータベースは非レプリカとしてマークされているからです)。

送付フェーズ

- 3 レプリカ作成パケットを新しいサイトに送信します。この処理は、手順 1 で使用したオプションによって異なります。

- **-fship** オプションを使用した場合、パケットは直ちに新しいサイトに送信されます。
- **-ship** オプションを使用した場合は、パケットを新しいサイトに送信するために **shipping_server** を実行する必要があります。たとえば、次のように指定します。

shipping_server -sclass cq_default -poll
- **-out** を使用してパケットをファイルに書き込んだ場合は、そのファイルを新しいサイトに送る必要があります。

インポート フェーズ

次の手順は San Francisco で行います。San Francisco の族内にはレプリカはありません。

- 4 保存および転送を使用した場合は、同期サーバ上で **lspacket** コマンドを入力してパケットの到着を確認します。

```
multiutil lspacket -short  
Multiutil:Pack  
¥incoming¥mk_sanfran_hub_21-May-01_19-28-01.xml'...
```

- 5 新しいスキーマ リポジトリとユーザー データベース用に空の製造元データベースを作成します。

- 6 インポート形式のレプリカ作成コマンドを入力します。

mkreplica -import コマンドでは、着信したパケットのパス名は、lspacket コマンドのリストで表示されるとおりに指定する必要があります。たとえば、次のように指定します。

```
multiutil mkreplica -import -site sanfran_hub -repository ORC1 -vendor ORACLE
-dbologin orcadmin password -connectopts
host=sanfran_dbserver;SID=ORC1;server_ver=8.1;client_ver=8.0;log_type=long
-database ORC1 -vendor ORACLE ORC1 -dbologin orcuser password -connectopts
host=sanfran_dbserver;SID=ORC1;server_ver=8.1;client_ver=8.0;log_type=long
-comments "telecommunication 族の San Francisco に使用する PRODA データベースと
そのスキーマ リポジトリの最初のレプリカをインポートします"
d:¥temp¥ms_ship¥incoming¥mk_sanfran_hub_21-May-01_19-28-01.xml
```

- 7 インポートが正常に終了したことを確認した上で、レプリカ作成パケットを削除します (更新パケットは自動的に削除されます)。

- 8 開発を開始します。

San Francisco のユーザーは、非レプリカ データベースの場合と同じ方法で新しいレプリカにアクセスすることができます。

この章では、同期を取るプロセスについて説明します。同期を取る際には、レプリカの作成時と同様に、エクスポート、送付、インポートの手順に従います。

- 1 エクスポート: 一方のレプリカで、**-export** オプションを指定して、**syncreplica** (レプリカの同期) コマンドを実行します。これにより、データの packets が作成されます。
- 2 送付: 作成された packets を 1 つ以上のほかのレプリカに送付します。
- 3 インポート: ほかのレプリカで、**-import** オプションを指定して、**syncreplica** コマンドを実行します。それにより、packets に記録されている変更が既存のレプリカに適用されます。

syncreplica コマンドはパフォーマンスを高めるように最適化されています。コマンド行に指定したターゲットのレプリカを更新するために必要な情報のみが含まれる packets が作成されます。

同期の成功に関する想定

同期処理のエクスポート フェーズとインポート フェーズは必ず別の時点で行われます。送信側のレプリカでは、packets が正常に受け取られ処理されたことをシブリング レプリカから通知してもらう必要はありません。送信側のレプリカでは、受信側での packets の処理が正常に行われたものと想定します。この想定により、パフォーマンスが最適化されます。次回に送付される packets には、前回の更新情報は含まれません。

障害が発生した場合は (たとえば、転送中の packets 損失や、シブリング レプリカでの CD 読み取り不能など)、送信側のレプリカのエポック数を調整して、損失データを再送信できるようにする必要があります。詳細については、「第 10 章 MultiSite の操作のトラブルシューティング」を参照してください。

スキーマの更新情報を含む packets の適用

作業スキーマ リポジトリからエクスポートされる packets には、ほかのサイトでデータベースをアップグレードすることが必要になる、新しいスキーマ リビジョンが格納されている場合があります。この場合は、ユーザー データベース レプリカが新しいスキーマ バージョンにアップグレードされるまで、**syncreplica -import** コマンドを実行してこのプロセスを終了することはできません。

パケットにスキーマ リポジトリとユーザー データベース レプリカの両方の更新情報が含まれている場合は、**sync replica -import** を実行するとプロセスが停止し、次のメッセージが出力されます。

packet_name は、**revision_number** ではなく **revision_number** リビジョン向けです ; サイト管理者がデータベースをアップグレードしてから **sync replica** を再実行してください。

この場合は、関係するユーザー データベース レプリカをアップグレードしてから、**sync replica -import** を再実行する必要があります。

同期プロセスを自動化している場合は、自動化スクリプトが失敗し、スキーマの変更に依存するそのほかのパケットは適用できなくなります。

手動による同期

ここでは、手動で **sync replica** コマンドを入力して、レプリカを同期する方法について説明します。

エクスポート フェーズ

- 1 送信側のホストで更新パケットを作成します。適切な転送オプションを指定して、**sync replica -export** コマンドを使用します。

サイト間が電子的に接続されている場合は、保存および転送機能を使用して、パケットを直ちに送信 (**-fship**) または記憶ベ이에保存 (**-ship**) することができます。

次の例では、**-fship** オプションを使用してパケットを直ちに送信します。

```
multiutil sync replica -export -clan telecom -site sanfran_hub -family PRODA -user jcole  
-password passwd -maxsize 50m -workdir c:\temp\packets -fship -sclass cq_default  
bangalore
```

送付フェーズ

-fship オプションを使用しなかった場合は、次の方法でパケットを送信します。

- **sync replica -export -ship** を使用した場合は、次のいずれかの方法で **shipping_server** を起動します。

```
shipping_server -sclass cq_default -poll  
shipping_server shipping-order-pathname
```

- **-fship** または **-ship** を使用しなかった場合は、電子メール、通常の郵便、都合の良い方法のいずれかでパケットを送付します。

インポート フェーズ

- 2 (フロッピーディスク、CD、テープ、電子メールを使用した場合) パケット ファイルを、受信側のレプリカの同期サーバーにあるディレクトリにコピーします。
- 3 `lspacket` コマンドを使用して、パケットが受信されたことを確認します。
- 4 受信側のレプリカでパケットを適用します。 `syncreplica -import` コマンドを使用して、パケット内の変更をレプリカに適用します。

次の例では、`-receive` オプションを指定して `syncreplica` を実行し、`shipping` の `incoming` ディレクトリにあるすべてのパケットをインポートします。

```
multiutil syncreplica -import -family PRODA -user kumar -password secret -receive  
-sclass cq_default
```

次の例では、引数としてディレクトリ パス名を指定します。 `syncreplica -import` により、そのディレクトリから更新パケットが検出され、ホスト上のレプリカに適用されます。

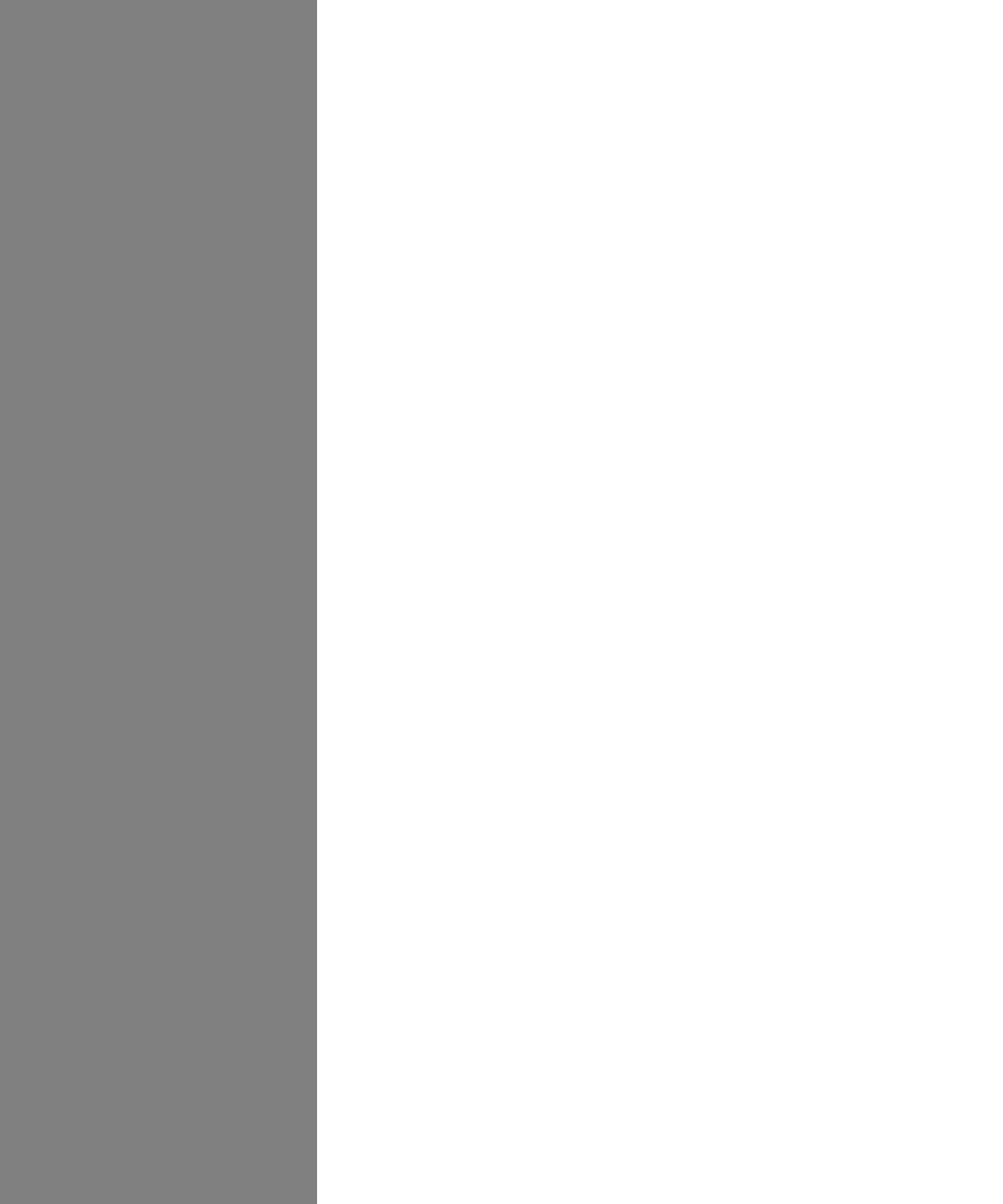
```
multiutil syncreplica -import -family PRODA -user kumar -password secret  
¥¥ramohalli¥¥shipping¥¥incoming¥¥
```

同期の自動化

同期の各フェーズを自動化するには、`cron` (UNIX) ジョブ、`at` (Windows) ジョブ、サードパーティのスケジューリング ツールを使用できます。 `shipping.conf` ファイル (UNIX) または `MultiSite` コントロール パネル (Windows) で受信ハンドラを指定して、インポート フェーズを自動化することもできます。

Windows では、`cquest-home-dir¥¥msimportauto.bat` スクリプトをテンプレートとして使用できます。

MultiSite 管理



この章では、既存のレプリカを管理する方法について説明します。また、レプリカを削除する方法についても説明します。レプリカを作成する方法については、「第 6 章 データベース レプリカの作成」を参照してください。

レプリカのプロパティの表示

`lsreplica` コマンドを実行すると、レプリカに関する情報が表示されます。

たとえば、DOC ファミリー内のすべてのレプリカの名前を表示するには、次のように指定します。

```
multiutil lsreplica -clan telecomm -site boston_hub -family DOC -user susan -password  
passwd -short  
BANGALORE  
BOSTON_HUB
```

同期サーバーの移動または名前変更

レプリカに関連付けられた同期サーバーは変更することができます。たとえば、使用しているコンピュータでハードウェア障害が発生した場合には、既存の同期サーバーの名前を変更する必要があります。この場合は、そのホストに関連付けられたレプリカのプロパティを更新する必要があります。これは、保存および転送機能が更新パッケージをレプリカに送付するルートを判定できるようにするためです。

同期サーバーを移動するには

- 1 新しいコンピュータに Rational Shipping Server をインストールします
(Rational ClearQuest のインストール情報を参照してください)。
- 2 `chreplica` コマンドを使用して、新しい同期サーバーをレプリカに関連付けます。

```
multiutil chreplica -clan telecomm -site bangalore -family PRODA -user kumar -password  
secret -host server3 bangalore
```
- 3 古い同期サーバーで同期プロセスを自動化していた場合、新しいサーバーで同期処理用のエクスポート スクリプトとインポート スクリプトを設定する必要があります。

- 4 ルーティング ホップを使用する場合は、**shipping.conf** ファイル (UNIX) または MultiSite コントロール パネル (Windows) でホスト名を更新します。
- 5 更新パッケージをすべてのシブリング レプリカにエクスポートします。

レプリカの移動または製造元データベース ソフトウェアの変更

レプリカの別のホストへの移動、またはレプリカの製造元データベース ソフトウェアの変更を行うには

- 1 『Rational ClearQuest 管理ガイド』の指示に従って操作します。
- 2 同期サーバーでメンテナンス ツールを使用して、データベースの接続情報を変更します。
- 3 更新パッケージをすべてのシブリング レプリカにエクスポートします。

レプリカに対する ID ブロック割り当ての変更

MultiSite では、各レプリカに割り当てられるレコード ID 番号の数を管理します。この割り当て処理は、ID ブロック (ID のグループ) を使用して実行されます。

デフォルトでは、各レプリカに対して、作成時に 4096 個の ID から成る ID ブロックが割り当てられます。レプリカの残り ID 数がしきい値の 1024 に達すると、4096 個の ID から成る別の ID ブロックが割り当てられます。これにより、すべての ID が一意に保たれます。ID ブロックの割り当ては、同期中に作業スキーマ リポジトリによって内部処理されます。

レプリカ ファミリのアクティビティ レベルに応じて、レプリカに割り当てられる ID ブロックのサイズを大きくして、同期処理がスムーズに実行されるようにすることが可能です。たとえば、デフォルト設定の場合は、同期パッケージに多くの新しいレコードが含まれて、受信側のレプリカの数が増えるにつれて ID ブロックに残っている ID の数を超過すると、同期インポートは失敗します。

レプリカに割り当てられる ID 数を制御するには、次のいずれかの操作を行います。

- **mkreplica -export** コマンドを使用して新しいレプリカを作成するときに、**-size** オプションと **-threshold** オプションを使用します。たとえば、実際の ID ブロック サイズを 25000、しきい値を 50% にして新しいレプリカを作成するには、次のように指定します。

```
multiutil mkreplica -export -clan telecom -site boston_hub -family DEV -u susan -p  
passwd -size 250 -threshold 50 -out c:\cqms¥boston_hub.xml goldengate:sanfran_hub  
Multiutil: パッケージ ファイル 'c:\cqms¥boston_hub.xml' が生成されました
```

- `chreplica` コマンドを使用して、サイズとしきい値の設定を変更します。たとえば、サイトにあるすべてのレプリカの実際の ID ブロック サイズを 50000、しきい値を 30% にするには、次のように指定します。

```
multiutil chreplica -clan testclan -site boston_hub -user susan -p passwd -host  
minuteman -size 500 -threshold 30 boston_hub
```

レプリカのマスターシップの変更

作業スキーマ リポジトリの変更については、87 ページの「作業スキーマ リポジトリのマスターシップの転送」を参照してください。

レプリカの削除

ここでは、レプリカを削除する方法について説明します。手順については、すべてを完了する必要があります。そうしないと、ファミリー内のほかのレプリカで、同期に関する問題やマスターシップに関する問題が発生する可能性があります。

レプリカを削除すると、同じファミリーに属するほかのレプリカは、削除するレプリカのエポック数の追跡を停止します。レプリカを削除しても、データベースは削除されません。

レプリカを削除する場合に必要な同期サイクルは 2 つです。まず、レプリカのすべてのオブジェクトのマスターシップを別のレプリカに転送します。次に、ほかのすべてのレプリカに対し、削除したレプリカが更新プロセスに関係しなくなったことを通知します。この情報を伝達できるのは同期プロセスによってのみなので、レプリカをレプリカ自体のサイトで削除することはできません。削除すると、レプリカで更新パケットを作成できなくなります。

ファミリーから削除したレプリカは、同期アクティビティに関与しなくなり、MultiSite 情報は追跡されなくなります。削除したレプリカは自身の `oplog` を更新しなくなります。そのため、ユーザーはそのレプリカ内のオブジェクトのマスターシップを転送できなくなります。

族からの動作レプリカの削除

まだアクセス可能で動作しているレプリカを削除するには、次の手順に従って操作します。この構文例では、(作業スキーマ リポジトリが存在する) `sanfran_hub` サイトと `boston_hub` サイトも含む族の場合に、`tokyo` サイトの `DOC` ファミリのレプリカを削除し、`tokyo` という名前のサイトの使用を中止する方法を示します。各コマンドは、1 行で入力してください。

- 1 削除対象のレプリカでのすべての作業を停止します。すべての更新パケットをインポートします。

- 2 すべてのオブジェクトのマスターシップを別のレプリカに転送します。

tokyo サイトで次のコマンドを実行します。

```
multiutil chmaster -clan telecomm -site tokyo -family DOC -user masako -password  
secret boston_hub -all -long
```

chmaster コマンドによってエラーが返された場合は、問題を解決してからコマンドを再実行します。

- 3 サイト全体の使用を中止する場合は、サイトのスキーマ リポジトリにあるユーザーとグループのマスターシップも転送する必要があります。

tokyo サイトで次のコマンドを実行します。

```
multiutil chmaster -clan telecomm -site tokyo -family MASTR -user masako -password  
secret boston_hub -all -long
```

chmaster コマンドによってエラーが返された場合は、問題を解決してからコマンドを再実行します。

- 4 更新パッケージをマスターシップの転送を受けるサイトに送信します。

tokyo サイトで次のコマンドを実行します。

```
multiutil syncreplica -export -clan telecomm -site tokyo -family DOC -user masako  
-password secret -workdir c:\work -fship boston_hub
```

boston_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -import -clan telecomm -site boston_hub -family DOC -user susan  
-password passwd -receive
```

- 5 更新パッケージを、マスターシップの転送を受けるサイトから残りのすべてのサイトに送信します。

boston_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -export -clan telecomm -site boston_hub -family DOC -user susan  
-password passwd -workdir c:\work -fship sanfran_hub
```

sanfran_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -import -clan telecomm -site sanfran_hub -family DOC -user jcole  
-password secret -receive
```

- 6 作業スキーマ リポジトリ サイトで rmreplica コマンドを実行します。最後の引数を必ず指定してください。この引数は削除するレプリカを示すものです。

boston_hub サイトで次のコマンドを実行します。

```
multiutil rmreplica -clan telecomm -site boston_hub -family DOC -user susan -password  
passwd tokyo
```

- 7 サイトの使用を中止する場合は、サイトのスキーマ リポジトリで **rmreplica** を実行する必要があります。

boston_hub サイトで次のコマンドを実行します。

```
multiutil rmreplica -clan telecomm -site boston_hub -family MASTR -user susan  
-password passwd tokyo
```

- 8 更新パッケージを、作業スキーマ リポジトリ サイトからほかのすべてのサイトに送信します。

boston_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -export -clan telecomm -site boston_hub -family DOC -user susan  
-password passwd -workdir c:\work -fship sanfran_hub
```

sanfran_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -import -clan telecomm -site sanfran_hub -family DOC -user jcole  
-password secret -receive
```

- 9 削除したレプリカの製造元データベースを削除します。

メモ: Rational では、**rmreplica** を使用して族から削除したデータベースの使用をサポートしていません。削除したデータベースを使用しようとすると、データが破壊される可能性があります。

族からの動作不能サイトの削除

サイトのデータベースが修復できないほどの損傷を受けた場合や、バックアップせずに削除されてしまった場合に、そのようなサイトを族から削除するには、次の手順に従って操作します。この例では、(作業スキーマ リポジトリが存在する) **sanfran_hub** サイトと **boston_hub** サイトも含む族から、**tokyo** サイトの **DOC** ファミリのレプリカを削除し、**tokyo** サイトの使用を中止する方法を示します。各コマンドは、1 行で入力してください。

- 1 すべてのオブジェクトのマスターシップを、修復不能なレプリカから別のレプリカに強制的に転送します。

boston_hub サイトで次のコマンドを実行します。

```
multiutil chmaster -clan telecomm -site boston_hub -family DOC -user admin -password  
secret boston_hub -all -force tokyo
```

- 2 サイトの使用を中止する場合は、すべてのユーザーとグループのマスターシップを強制的に転送します。

boston_hub サイトで次のコマンドを実行します。

```
multiutil chmaster -clan telecomm -site boston_hub -family MASTR -user admin  
-password secret boston_hub -all -force tokyo
```

- 3 作業スキーマ リポジトリ サイトで **rmreplica** を実行し、修復不能なレプリカを削除します。最後の引数を必ず指定してください。この引数は削除するレプリカを示すものです。

boston_hub サイトで次のコマンドを実行します。

```
multiutil rmreplica -clan telecom -site boston_hub -family DOC -user admin -password secret tokyo
```

- 4 サイト全体の使用を中止する場合は、**rmreplica** コマンドを実行してサイトのスキーマ リポジトリを削除します。

boston_hub サイトで次のコマンドを実行します。

```
multiutil rmreplica -clan telecom -site boston_hub -family MASTR -user admin -password secret tokyo
```

- 5 更新パケットを、作業スキーマ リポジトリが存在するサイトから残りのすべてのサイトに送信します。

boston_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -export -clan telecom -site boston_hub -family DOC -user admin -password secret -workdir c:\work -fship sanfran_hub
```

sanfran_hub サイトで次のコマンドを実行します。

```
multiutil syncreplica -import -clan telecom -site sanfran_hub -family DOC -user admin -password secret -receive
```

- 6 削除したレプリカとスキーマ リポジトリの製造元データベースを削除します。

族の最後のレプリカを削除した後の MultiSite の使用法

族の最後のレプリカで **rmreplica -dbset** コマンドを使用すると (作業スキーマ リポジトリが存在するサイトのデータベースのみを残します)、データベースは **MultiSite** 環境の構成要素ではなくなります。コマンドの実行後は非レプリカ データベースになり、この状態の間は、新しいレプリカを作成するために使用することができなくなります。詳細については、**rmreplica** リファレンス ページを参照してください。

データベースの状態を変更して再びレプリカを作成できるようにするために、**activate** コマンドを実行する必要はありません。**Rational ClearQuest** メンテナンス ツールを使用して、データベース セット名を **multiutil** によって予期される形式 (たとえば **CQMS.clan-name.site**) に戻す必要があります。

この章では、レプリカ内のオブジェクトのマスターシップを管理する方法について説明します。
この章を読む前に、8 ページの「独立した開発の実現：マスターシップ」をお読みください。

ユーザー データベース オブジェクトのマスターシップのコマンド

次の `multiutil` コマンドを使用して、ユーザー データベースのマスターシップを管理します。

- `chmaster`
- `describe`

これらのコマンドの詳細については、このマニュアルのリファレンス ページを参照してください。

レコードのマスターシップ情報の表示

クエリー結果またはレコードフォーム上で、南京錠型のアイコンは、結果セットにおいてほかのサイトでマスター登録されているレコードを示します。

データベース オブジェクトのマスターシップの変更

`chmaster` コマンドまたは GUI を使用してオブジェクトのマスターシップを転送できます。次のような場合にマスターシップを変更します。

- レプリカをマスターとするレコードまたは共用クエリーを、別のレプリカのユーザーが変更できるようにする場合
- 別のレプリカをマスターとするユーザー情報を変更する場合
- レプリカの使用を停止している間、そのレプリカをマスターとするすべてのオブジェクトのマスターシップを、残りのレプリカのいずれかに転送する必要がある場合

マスターシップの変更は、標準の同期メカニズムにより、レプリカ間に伝達されます。マスターシップを変更する一般的な手順は以下のとおりです。

- 1 マスター レプリカで、1 つ以上のオブジェクトのマスターシップをほかのレプリカに転送します。
- 2 古いマスター レプリカ側で、古いマスター レプリカから新しいマスター レプリカに更新パッケージをエクスポートします。
- 3 新しいマスター レプリカ側で、更新パッケージをインポートします。

マスターシップの変更情報が収められている更新パッケージが新しいマスター レプリカにインポートされるまで、マスターシップの変更内容は「更新パッケージ内」にあり、データベースファミリ内のレプリカの間で、どのレプリカがオブジェクトのマスターであるかに関する情報に相違が生じます。

たとえば、`sanfran_hub` レプリカの管理者が `QA_ENGINEERING` ユーザー グループのマスターシップを `bangalore` レプリカに転送し、その後、更新パッケージをエクスポートしたとします。この時点では次のようになります。

- `sanfran_hub` レプリカでは、`bangalore` がユーザー グループのマスターであると解釈します。
- `bangalore` レプリカでは、`sanfran_hub` がユーザー グループのマスターであると解釈します。
- どのレプリカでも `QA_ENGINEERING` ユーザー グループを変更できません。

`bangalore` レプリカで更新パッケージをインポートしてマスターシップの転送を完了すると、`bangalore` のユーザーが `QA_ENGINEERING` ユーザー グループを変更できるようになります。

マスターシップを変更する際は、次の点に注意してください。

- ファミリに読み取り専用または一方向（更新パッケージをインポートするがエクスポートはしない）のレプリカがある場合、それらのレプリカにマスターシップを転送するには注意が必要です。読み取り専用または一方向のレプリカにマスターシップを転送した後は、そのままではオブジェクトのマスターシップを変更できません。マスターシップを変更するには、ファミリの同期パターンを変更する必要があります。
- サイトで行ったマスターシップの変更を、逆に操作して戻すことはできません。88 ページの「誤ってマスターシップを変更した場合の修正」を参照してください。

GUI によるレコードのマスターシップの転送

ClearQuest MultiSite には、`ratl_mastership` というシステム フィールドが含まれています。このフィールドの値は、各レコードのマスターであるレプリカです。ユーザーにレコードマスターシップの変更許可を与えるには、このフィールドをレコードタイプのフォームに追加する必要があります。レコード タイプのフォームには、`ratl_mastership` フィールドをいつでも追加できます。

メモ: レコードのマスター レプリカのユーザーのみがそのマスターシップを変更できます。

ユーザーにレコード マスターシップの変更許可を与えるレコード タイプごとに、ClearQuest Designer を作業スキーマ リポジトリで使用して、レプリカ ファミリのスキーマを変更します。レコード フォームに `ratl_mastership` フィールドを追加するには、次の手順に従います。

- 1 ワークスペースで、[レコード タイプ] フォルダを展開し、次に目的のレコード タイプを展開します。
- 2 フォームをダブルクリックします。
- 3 フィールドを追加するタブをクリックするか [編集] メニューの [タブの追加] をクリックして、フィールドを含む新規タブを追加します。
- 4 [フィールド リスト] で、`ratl_mastership` フィールドをタブまでドラッグします。
- 5 変更したスキーマをチェックインします。
- 6 新しいスキーマを使用するように、該当するユーザー データベースをアップグレードします。
- 7 更新パケットを生成し、送信します。更新パケットは自動的にスキーマの更新を含みます。ほかのサイトの管理者は、サイトのユーザー データベースのレプリカを、新しいスキーマ リビジョンを使用するようにアップグレードする必要があります。詳細については、69 ページの「スキーマの更新情報を含むパケットの適用」を参照してください。

スキーマの変更については、『Rational ClearQuest 管理ガイド』を参照してください。

chmaster によるレコードのマスターシップの転送

レコードのマスターシップを転送するには

- 1 マスター レプリカ (この例では `boston_hub`) を対象にして `chmaster` コマンドを実行します。

```
multiutil chmaster -clan telecom -site boston_hub -family DOC -user susan -password  
passwd bangalore entity:DOC00013  
multiutil: The mastership of entity:DOC00013 has been changed to  
site 'bangalore'
```
- 2 古いマスター レプリカから新しいマスター レプリカに更新パケットをエクスポートします。

```
multiutil sync replica -export -clan telecom -site boston_hub -family DOC -user susan  
-password passwd -workdir d:\shipping\temp -fship -sclass cq_default bangalore
```
- 3 新しいレプリカ側で、更新パケットをインポートします。

```
multiutil sync replica -import -clan telecom -site bangalore -family DOC -user kumar  
-password passwd -receive -sclass cq_default
```

- 4 新しいレプリカで、マスターシップが転送されたことを確認します。

```
multiutil describe -clan telecomm -site bangalore -family DOC -user kumar -password  
passwd entity:DOC00013  
multiutil: The mastership of entity:DOC00013 is 'bangalore'
```

GUI によるワークスペース項目のマスターシップの転送

ClearQuest クライアントを使用してワークスペース項目のマスターシップを変更するには

メモ: [共有クエリー] フォルダ内のワークスペース項目を変更するには、共有フォルダ管理者権限が必要です。

- 1 ワークスペース内で、変更する項目を右クリックし、[マスターシップ] をクリックします。
- 2 [マスターシップの変更] ダイアログ ボックスで、[新規のマスター サイト] リストから新しいマスター レプリカを選択します。
- 3 [OK] をクリックします。
- 4 古いマスター レプリカから新しいマスター レプリカに更新パッケージをエクスポートし、新しいマスター レプリカ側でパッケージをインポートします。

chmaster によるワークスペース項目のマスターシップの転送

以下のガイドラインに従ってワークスペース項目を指定します。

- ワークスペース名を引用符で囲みます。
- 英文字の大小は、ClearQuest ワークスペースの項目に表示されるものに従います。
- ワークスペース項目のワークスペースのフルパス名 (フォルダとサブフォルダ) を含めます。

例を次に示します。

"Workspace:Personal Queries(susan)¥My Projects¥My Query"

ワークスペース項目のマスターシップを変更するには

- 1 マスター レプリカ (この例では boston_hub) を対象にして chmaster コマンドを実行します。

```
multiutil chmaster -clan telecomm -site boston_hub -family DOC -user susan -password  
passwd bangalore "Workspace:Public Queries¥Triage¥project report"
```
- 2 古いマスター レプリカから新しいマスター レプリカに更新パッケージをエクスポートします。

```
multiutil syncreplica -export -clan telecomm -site boston_hub -family DOC -user susan  
-password passwd -workdir d:¥shipping¥temp -fship -sclass cq_default bangalore
```
- 3 新しいレプリカ側で、更新パッケージをインポートします。

```
multiutil syncreplica -import -clan telecomm -site bangalore -family DOC -user kumar  
-password passwd -receive -sclass cq_default
```

- 4 新しいレプリカで、マスターシップが転送されたことを確認します。

```
multiutil describe -clan telecomm -site bangalore -family DOC -user kumar -password  
passwd "Workspace:Public Queries¥Triage¥project report"  
multiutil: The mastership of Workspace:Public  
Queries¥Triage¥project report is 'bangalore'
```

ユーザーまたはグループのマスターシップの転送

新しいユーザーまたは新しいグループなど、ユーザー管理上の変更を含む同期パケットを受信する場合、その変更を使用するようにユーザー データベースをアップグレードする必要があります。MultiSite 環境でのユーザーの管理の詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

ユーザーまたはグループのマスターシップを変更するには

- 1 古いマスター レプリカ側で次のように操作します。
 - a **chmaster** コマンドまたは **ClearQuest Designer** を使用して、ユーザーまたはグループのマスターシップを新しいレプリカに変更します。
 - b 更新パケットを新しいマスター レプリカにエクスポートします。
- 2 着信したパケットにはユーザー データベースの変更が必要であることを、新しいマスター レプリカの管理者に通知します。
- 3 新しいマスター レプリカ側で次のように操作します。
 - a 更新パケットをインポートします。
 - b 変更に関連するユーザー データベースをアップグレードします。
 - c 同期パケットをエクスポートして送信し、ファミリー内のほかのレプリカが変更内容を基に更新されるようにします。

GUI によるユーザーまたはグループのマスターシップの変更

ユーザーのマスターシップを変更するには

- 1 **ClearQuest Designer** で [ツール] メニューの [ユーザー管理] をクリックします。
- 2 [ユーザー管理] ダイアログ ボックス内で、変更するユーザーをダブルクリックします。
- 3 [ユーザー プロパティ] ダイアログ ボックス内で、[マスターシップ] リストを開き、新しいマスター レプリカを選択します。
- 4 [OK] をクリックします。
- 5 [OK] をクリックします。

- 6 古いマスター レプリカから新しいマスター レプリカに更新パッケージをエクスポートして送信し、新しいマスター レプリカ側でパッケージをインポートします。

ユーザー グループのマスターシップを変更するには

- 1 ClearQuest Designer で [ツール] メニューの [ユーザー管理] をクリックします。
- 2 [ユーザー管理] ダイアログ ボックス内で、変更するユーザー グループをダブルクリックします。
- 3 [グループ アクション] メニューの [グループの編集] をクリックします。
- 4 [グループ プロパティ] ダイアログ ボックス内で、[マスターシップ] リストを開き、新しいマスター レプリカを選択します。
- 5 [OK] をクリックします。
- 6 [OK] をクリックします。
- 7 古いマスター レプリカから新しいマスター レプリカに更新パッケージをエクスポートして送信し、新しいマスター レプリカ側でパッケージをインポートします。

chmaster によるユーザーまたはグループのマスターシップの変更

以下の規則に基づいて、ユーザーとグループを指定します。

- ユーザー名またはグループ名にスペースが含まれている場合は、それらを引用符で囲みます。例を次に示します。

user:"John Smith"

- ユーザー名またはグループ名は作成されたときに割り当てられたものと同じ形式を使用します。

次の例では、ユーザー John Smith と Jane Doe のマスターシップを boston_hub レプリカから bangalore レプリカに変更します。

- 1 マスター レプリカ (この例では boston_hub) を対象にして chmaster コマンドを実行します。

```
multiutil chmaster -clan telecom -site boston_hub -family DOC -user susan -password  
passwd bangalore user:"John Smith" user:"Jane Doe"
```

```
multiutil: The mastership of records "Jane Doe" and "John Smith" of  
type "user" has been changed to site 'bangalore'
```

multiutil:一部のユーザーまたはグループのマスターシップはこのサイトから転送されています。
ローカルユーザー管理者は、新規のマスター サイト 'BANGALORE' にあるユーザー データ
ベースを、変更がほかのユーザー データベースに対して参照可能になる前に、更新する必要があります。

- 2 古いマスター レプリカから新しいマスター レプリカに更新パッケージをエクスポートします。

```
multiutil sync replica -export -clan telecom -site boston_hub -family DOC -user susan  
-password passwd -workdir d:\shipping\temp -fship -sclass cq_default bangalore
```

- 3 新しいレプリカ側で、更新パッケージをインポートします。
`multiutil syncreplica -import -clan telecomm -site bangalore -family DOC -user kumar -password passwd -receive -sclass cq_default`
- 4 新しいマスター レプリカ側で、ユーザー データベースを新しいユーザー情報でアップグレードします。ユーザー データベースのアップグレードの詳細については、『Rational ClearQuest 管理ガイド』を参照してください。
- 5 新しいレプリカで、マスターシップが転送されたことを確認します。
`multiutil describe -clan telecomm -site bangalore -family DOC -user kumar -password passwd user:"John Smith" user:"Jane Doe"`
`multiutil: The mastership of user:John Smith is 'bangalore'`
`multiutil: The mastership of user:Jane Doe is 'bangalore'`

作業スキーマ リポジトリのマスターシップの転送

作業スキーマ リポジトリの管理者は、スキーマの変更と、族への新しいファミリの追加を担当します。詳細については、5 ページの「スキーマ リポジトリの種類」を参照してください。これらのタスクの担当を別のサイトに移動する場合、作業スキーマ リポジトリのマスターシップを変更する必要があります。

作業スキーマ リポジトリのマスターシップを転送するには

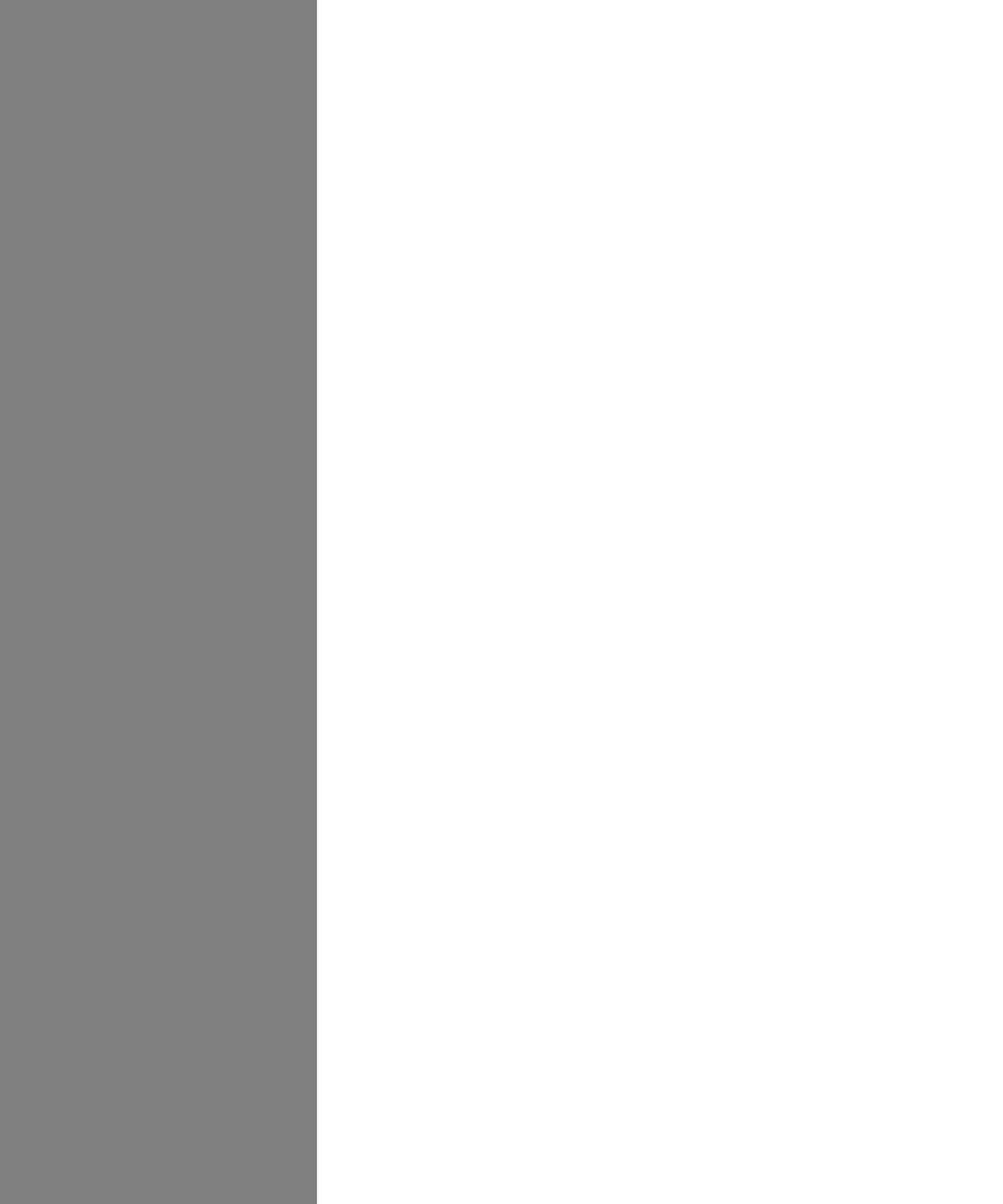
- 1 作業スキーマ リポジトリのレプリカを対象にして、`chmaster` コマンドを実行します。
`multiutil chmaster -clan telecomm -site boston_hub -family MASTR -user susan -password passwd sanfran_hub -workingmaster`
- 2 古い作業スキーマ リポジトリから新しい作業スキーマ リポジトリに更新パッケージをエクスポートします。
`multiutil syncreplica -export -family MASTR -user susan -password passwd -workdir c:\temp\shipping -fship -sc cq_default sanfran_hub`
- 3 新しい作業スキーマ リポジトリ側で、更新パッケージをインポートします。
`multiutil syncreplica -import -family MASTR -user jcole -password passwd -receive -sc cq_default`
- 4 新しい作業スキーマ リポジトリ側で、マスターシップが転送されたことを確認します。
`multiutil describe -clan telecomm -site sanfran_hub -family MASTR -workingmaster`

誤ってマスターシップを変更した場合の修正

誤ってレプリカのマスターシップを変更した場合、次の手順に従います。

- 1 該当するレプリカで、更新パッケージを新しいマスター レプリカに送信します。
- 2 新しいマスター レプリカ側で次のように操作します。
 - a パッケージをインポートします。
 - b マスターシップを元のレプリカに戻します。
 - c 更新パッケージを元のレプリカにエクスポートし、送信します。
- 3 元のレプリカで、更新パッケージをインポートします。

トラブルシューティング



MultiSite の操作のトラブルシューティング

10

この章では、Rational ClearQuest MultiSite のコマンドを実行して予期しない結果が発生した場合によくある状況について説明します。警告やエラー メッセージが表示される場合もあります。次のような状況に分類することができます。

- **予測可能な状態**は、異なるレプリカで何らかの矛盾した変更が避けられないために発生します。多くの場合、これらの矛盾は MultiSite の操作で解決できるので、特に対応する必要はありません。
- **回復可能なエラー**は、回復手順を実行すれば解決できるユーザー エラーやハードウェアのトラブルなどの問題です。
- **重大なエラー**は、Rational カスタマ サポートに問い合わせることが必要な場合がある問題です。

以下の説明は、一般的な MultiSite のデータの流れに沿って構成されています。レプリカの作成に続き、レプリカ同期のフェーズ、つまりエクスポート、転送、インポートの順になります。

レプリカのエクスポートの問題

`mkreplica -export` コマンドにより、指定した名前のレプリカがファミリーに存在することがわかった場合 (`Replica replica-name already exists`)、新しいレプリカに別の名前を選択し、`mkreplica -export` コマンドを再入力します。

パケットの転送中に `mkreplica -export -fship` が失敗した場合、作成レプリカでは新しいレプリカのレプリカ オブジェクトが削除されません。レプリカの作成を完了するには、`shipping_server` を使用してレプリカ作成パケットを転送します。

`mkreplica -export` の失敗からの回復

`mkreplica -export` が失敗した場合、データベースがロックされたままの状態である可能性があります。この問題を解決するには、この項で説明している手順に従うか、Rational カスタマ サポートにお問い合わせください。

スキーマ リポジトリとユーザー データベースのアンロック

- 1 `installutil unlockschemarepo` コマンドを使用して、スキーマ リポジトリをアンロックします。このコマンドの構文は次のとおりです。

```
installutil unlockschemarepo db-vendor server database dbo-login dbo-password  
connection-options
```

接続オプションは次のとおりです。

Oracle データベース `HOST=host;SID=sid`

その他すべて `""`

たとえば、サーバー `QE_TEST1` 上の `SQL_SERVER` スキーマ リポジトリ データベース `test_master_sitea` をアンロックするには、次のように入力します。

```
installutil unlockschemarepo SQL_SERVER QE_TEST1 test_master_sitea multisite  
multisite ""
```

ここで、`multisite` は、`test_master_sitea` データベースの `dbo-login` と `dbo-password` です。

- 2 `installutil unlockuserdb` コマンドを使用して、ユーザー データベースをアンロックします。このコマンドの構文は次のとおりです。

```
installutil unlockuserdb db-vendor server database dbo-login dbo-password  
connection-options
```

接続オプションは次のとおりです。

Oracle データベース `HOST=host;SID=sid`

その他すべて `""`

たとえば、サーバー `QE_TEST1` 上の `SQL_SERVER` ユーザー データベース `test_master_sitea` をアンロックするには、次のように入力します。

```
installutil unlockuserdb SQL_SERVER QE_TEST1 test_user_sitea multisite multisite ""
```

ここで、`multisite` は、`test_user_sitea` データベースの `dbo-login` と `dbo-password` です。

以後の multiutil コマンドが失敗する

以後 `mkreplica -export` を実行して「レプリカが既に存在している」または「別の `multiutil` 処理が実行中」という意味のメッセージが表示される場合は、Rational カスタマ サポートにお問い合わせください。

レプリカのインポートの問題

`mkreplica -import` コマンドがコード ページの不一致以外の理由 (たとえば、ネットワーク障害やディスクの空き容量不足) で失敗した場合は、コマンドを再実行してください。`mkreplica -import` コマンドが失敗した理由が、エクスポートする側のレプリカのデータ コード ページ値と、インポートする側のサイトにある既存のスキーマ リポジトリのコード ページ値が一致していないためである場合は、いずれかのコード ページ値を設定し直してから `mkreplica -import` を再実行してください。

`mkreplica` は、スキーマ リポジトリのインポート中またはユーザー データベースのインポート中に失敗する場合があります。

`mkreplica -import` がスキーマ リポジトリのインポート中に失敗した場合は、次の手順に従います。

- 1 スキーマ リポジトリとユーザー データベース レプリカの製造元データベースを削除します。
- 2 新しく作成されたデータベース セット名を削除します。データベース セット名の形式は、`CQMS.clan-name.site-name` です。次のコマンドを使用します。
`installutil dropdbset CQMS.clan-name.site-name`
- 3 製造元データベースを再作成します。
- 4 `mkreplica -import` コマンドを入力します。

`mkreplica -import` を実行すると、スキーマ リポジトリはインポートされますが、ユーザー データベースのインポート中に失敗する場合は、次の手順に従います。

- 1 ユーザー データベース レプリカ用に作成した製造元データベースを削除します。
- 2 製造元データベースを再作成します。
- 3 `mkreplica -import` コマンドを、リポジトリ データベース オプションなしでもう一度実行します。例を次に示します。

```
multiutil mkreplica -import -clan telecomm -site tokyo -user masako -p secret -database  
cq_userdb -vendor SQL_SERVER -dbologin juseradmin secret -rwlogin juseradmin  
secret
```

同期エクスポートの問題

ここでは、同期のエクスポート フェーズで発生する問題について説明します。

oplog 項目が見つからない

`syncreplica -export` は、次の警告メッセージを表示して失敗することがあります。

Multiutil: エラー: 'family' データベースは oplog-ID oplog-ID によって、元は 'site' にあった oplog エントリをエクスポートしました。次にエクスポートされる oplog-ID は oplog-ID ですが、本来は oplog-ID である必要があります。oplog エントリがとぎれている場合は、oplog エントリが不足している可能性があります。

(oplog 項目の詳細については、12 ページの「操作ログ」を参照してください。)

このエラーは、送信中のレプリカのエポック数マトリックスが **oplog** 項目と一致しない場合に発生します。例を次に示します。

- **sydney** から **buenosaires** への更新を送信する前に、**syncreplica** は **sydney** のエポック数マトリックスをチェックします。これで、**sydney** が **buenosaires** に送信した最終の操作が 3620 であったことを特定します。
- **sydney** のデータベースでスクラブする **oplog** が、3620 に続く操作の一部を削除していることがわかります。それ以降で **oplog**に残っている **sydney** の最初の操作は 5755 です。

この不一致は、予測できることがあります。たとえば、ファミリーの同期パターンを変更すると、それまで互いにやり取りしていなかったレプリカが更新パケットの交換を開始します。2 つのレプリカ (**syncreplica -export** と後続の **syncreplica -import**) を同期すると、送受信しているレプリカに対するエポック数マトリックスの行が更新されますが、それ以外のレプリカ行は更新されません。2 つのレプリカが互いのディレクトリに更新をほとんどまたはまったく送信しない場合、エポック数マトリックスの関連行の期限が切れているか、すべてが 0 で構成されています。これは、レプリカが操作を間接的に (たとえば、ハブ レプリカを通して) 受け取っている場合は問題ありません。

この場合、**sydney** に、**buenosaires** に関する本当の状況、つまり **sydney** は通常の同期メカニズムを通して受信していないことを通知する必要があります。この情報が通知されると、**sydney** は **buenosaires** に送る **oplog** 項目を決定できます。

この状況は、**oplog** 項目を **scruboplog** コマンドを使用して削除してからほかのレプリカに送信した場合にも発生します。任意のレプリカで **oplog** をスクラブする前に、ファミリー内のレプリカを同期してください。33 ページの「レプリカのパラメータのスクラブ」を参照してください。

出力記憶ベ이에パケットが累積する

パケット デリバーに関する問題は回復可能なエラーです。ほとんどの場合、MultiSite の自動再試行機能でエラーから回復できます。

1 つ以上のホストに転送するために、保存および転送機能に送信されたレプリカ作成パケットまたは更新パケットには、発送オーダー ファイルがあります。(論理パケットには、それぞれ独自の発送オーダーを持つ複数の物理パケットを含めることができます。)通常、発送オーダーには有効期限があります。有効期限は次のいずれかで指定されます。

- パケットを作成した **syncreplica** または **mkreplica** コマンド (または任意のファイルを保存および転送機能に送信する **mkorder** コマンド) の **-pexpire** オプションで指定した日時。
- UNIX では、送信元ホストの保存および転送構成ファイル (**shipping.conf**) にある **EXPIRATION** の値。
- Windows では、送信側のホストの [MultiSite Configuration 情報] で指定されている [パケットの有効期限] の値。

発送オーダーの期限が切れるまで、繰り返しデリバーしようとします。

レプリカ自体を更新することはできない

送信するレプリカを宛先に指定すると、エクスポート中に次のメッセージを受信します。

レプリカ自体を更新することはできません

送信側のレプリカが指定した唯一のレプリカである場合、**syncreplica -export** コマンドは失敗します。ほかのレプリカを指定した場合、このメッセージは警告として表示され **syncreplica -export** コマンドは処理を継続します。

転送の問題

ここでは、同期の転送フェーズで発生する問題について説明します。

エラー メッセージ

表 12 のメッセージは、**mkorder** コマンド、**mkreplica** コマンド、**shipping_server** コマンド、**syncreplica** コマンドから生成されます。

表 12 発送エラー メッセージ(1 / 2)

エラー メッセージ	意味
クラス class-name: の記憶ベイが見つかりません。 構成ファイルにそのようなベイは指定されていません	shipping.conf ファイルまたは [MultiSite Configuration 情報] で記憶クラス class-name に記憶ベイが割り当てられていません。
クラス class-name の記憶ベイが見つかりません。 該当するベイはすべてアクセス不可になっているか、 byte-count バイトの空き容量がありません	権限がないかディスクの空き容量が不足しているために、クラス class-name の記憶ベイが使用できません。
オーダー ファイルの可能性のある shipping-order-pname を処理できません。ユーザー username (UID uid) は所有者ではありません	(UNIX) シッピング サーバーは root として実行 されていません。username は発送オーダー ファイルを所有していません。
ホスト hostname (next-hop-hostname 経由) へ の循環デリバー ルートがオーダー shipping-order-pname について検出されました	発送オーダーが、パケットのデリバー ルート における前のホップとして next-hop-hostname をリストに表示しています。このパケットが next-hop-hostname (shipping.conf ファイルの ROUTE エントリまたは [MultiSite Configuration 情報] の [ルーティング情報] セク ションに指定されます) に送信された場合、最 終的に現在のホストに戻ることにになります。デ リバー パスに存在するホストのルーティング情 報を確認し、循環ルートを修正してください。
ファイル file-pname には有効な発送オーダーが ありません	シッピング サーバーは、発送オーダーでない ファイル进行处理しようとしてしました。
セキュリティ上の理由により、発送オーダー shipping-order-pname を処理できませんでした。 発送オーダーと同じディレクトリにデータ ファイル file-pname が必要です	発送オーダーとそれに関連付けられたパケット ファイルは、同じディレクトリにある必要があ ります。
オーダー shipping-order-pname をホスト hostname に返す処理を中止します (元のデータ ファイルは file-pname)	シッピング サーバーは、パケットやほかのフア イルを送信元のホストに戻せないため (たと えば発送オーダーの有効期限が切れた場合)、発送 オーダーとデータ ファイルを削除しました。
発送ベイ storage-bay-pname を無視します : reason	shipping.conf ファイルまたは [MultiSite Configuration 情報] で指定された記憶ベイ ディレクトリが存在しないか、アクセスできま せん。

表 12 発送エラー メッセージ(2 / 2)

shipping order shipping-order-pname not found (perhaps previously sent?)	受信ハンドラの処理中に、 SHIPPING サーバーは別のホストに転送されるパケットの発送オーダーを見つけることができません。 shipping_server -poll が呼び出され、既にパケットが発送された可能性があります。(パケットがホスト上のレプリカに適用されると、パケットが転送される前にインポートが発生します。このため、ポーリング操作が予定されているとパケットを送信する可能性があります。)
--------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

無効な宛先

ローカル ホストの **hosts** ファイル、**hosts NIS** マップ、ドメイン名サービスは、以下のホストのいずれか 1 つをリストに表示する必要があります。

- 宛先ホスト
- 宛先ホストに対応する **Next-hop** ホスト (UNIX ではホストの **shipping.conf** ファイルにある **ROUTE** エントリで定義し、Windows ではホストの [MultiSite Configuration 情報] にある [ルーティング情報] セクションで定義します。)

メモ: ネットワークにあるホストの IP アドレスしかわからない場合、ホスト名の代わりに IP アドレスを使用することもできます。

エントリがないとパケットのデリバリー先がわからないので SHIPPING サーバーは失敗します。この場合、ログ ファイル (UNIX) または Windows イベントビューアにエラー メッセージが記録されます。

宛先ホスト名のスペルが間違っている場合、**mkorder** コマンドを使用して正しいホスト名の新規発送オーダーを作成します。**mkreplica -export** コマンドでホスト名のスペルを間違えて入力すると、間違ったホスト名が記録されます。**lsreplica -long** でエラーを確認し、**chreplica** でスペルを訂正します。

そのほかの場合は、リモート ホストにあるホストのデータベースを訂正する必要があります。送信側のホストは、**TCP/IP** チャネル経由で受信側のホストと通信する必要があります。送信ホストで **rcp** コマンドを使用して、受信側のホストにファイルをコピーします。コマンドが失敗した場合、ホストのセットアップかネットワークに問題があります。コマンドが正常に実行された場合は、**Rational** カスタマ サポートにお問い合わせください。

デリバーに失敗する

SHIPPING サーバーが有効な宛先にパケットをデリバーできない場合、必ずエラー メッセージが次の場所に記録されます。

- (UNIX) ファイル `/var/adm/rational/clearcase/log/shipping_server_log`。また、ターミナル デバイスがある場合、ターミナル デバイスに表示されます。
- (Windows) Windows イベント ビューア。

リモート ホストのダウンやネットワーク接続のダウンなど一時的な問題の場合、引き続き `shipping_server -poll` を呼び出せばパケットは正常に転送されます。問題が一時的なものでない場合、最終的には発送オーダーの有効期限が切れます。

SHIPPING サーバーが起動できない、または接続を拒否された

受信側のサイトの SHIPPING サーバーが起動しなかったり接続が拒否された場合は、受信ホストの `albd_server` ログを調べて障害の内容を確認します。

UNIX の `shipping.conf` ファイルに構文エラーがあると、接続が拒否されます。たとえば、ファイルの電子メール アドレスが間違っていると、`albd_server` ログに次のようなエラーが表示されます。

エラー : `shipping_server(9951)` : エラー : 構成ファイル (60 行目) に構文エラーがあります

発送オーダーの有効期限切れ

SHIPPING サーバーは、発送オーダーの有効期限切れを見つけると、元のホストにパケットを戻そうとします。また、初めの送信元ホストの 1 人以上の管理者にメール メッセージを送信し、パケットが送信元ホストに戻ったときには別のメール メッセージを送信します。Windows では、電子メールによる通知が有効になっていない場合、SHIPPING サーバーは Windows イベント ビューアにメッセージを書き込みます。

`lspacket` コマンドを使用して、ホストの戻りベイを確認します。保存および転送が既にパケット ファイルに戻している場合があります。その場合、パケットを再デリバーしてみてください。

- ネットワーク接続など、保存および転送パケットデリバー メカニズムを修正します。次に、`mkorder` を使用して戻りベイの各物理パケット ファイルに対する新規発送オーダーを作成します。
- 保存および転送メカニズムを修正できない場合、別の方法でパケットをデリバーします。たとえば、パケット ファイルを CD にコピーし、その CD をリモート サイトに郵送します。

パケット ファイルがホストの戻りベイにない場合、転送中の可能性があります。14 日以内に元のホストに戻せないパケットは削除されるので、すぐにファイルを検索してください。

同期インポートの問題

ここでは、同期のインポート フェーズで発生する問題について説明します。

入力記憶ベイにパケットが累積する

更新パケットが行方不明になり、レプリカに適用できない場合、回復可能なエラーが発生します。次のような現象が発生します。

- 自分のサイトの 1 つ以上のレプリカが通常のスケジュールで更新されない。
- `lspacket` コマンドで、記憶ベイに累積した未処理パケットが表示される。これらのパケットは、行方不明のパケットに依存しているので、処理できません。

パケットが行方不明であることを確認し、どのような操作が必要かを決定するには

- 1 `syncreplica -import -receive` コマンドを入力して、記憶ベイのすべての入力パケットを正しい順序で処理します。`syncreplica` でいずれのパケットの処理も失敗する場合は、パケットは行方不明になっています。
- 2 `syncreplica -import` コマンドを入力し、記憶ベイで最も古いパケットを指定します。

```
multiutil syncreplica -import -clan telecomm -site sanfran_hub -family DEV -user jcole -p  
passwd packet-pathname
```

Multiutil: パケット packet-pathname は処理されていません...

Multiutil: UPDATE_PACKET パケットは、BOSTON_HUB から 2002-03-25

17:42:41 に送信されました。'DEV' の は再現できません: この複製は、2 を介して再現しただけで、複製 BOSTON_HUB のエポック 6 を再現してはいません。

Multiutil: UPDATE_PACKET パケットは、BOSTON_HUB から 2002-03-25

17:42:41 に送信されました。'MASTR' の は再現できません: この複製は、6 を介して再現しただけで、複製 BOSTON_HUB のエポック 8 を再現してはいません。

この例では 1 つ以上のパケットが行方不明になっています。行方不明になっているパケットには、`boston_hub` サイトの DEV ファミリ内のユーザー データベースで最初に発生した操作 3～6 と、`boston_hub` サイトのスキーマ リポジトリで発生した操作 7～8 が含まれます。通常、パケットには複数のレプリカからの操作があり、そのいずれかのレプリカからの操作が行方不明になると `syncreplica -import` コマンドは失敗します。

行方不明のパケットを探します。それらのパケットは、処理し忘れたメディアにあったり、保存および転送の設定 (UNIX では `shipping.conf` ファイル、Windows では [MultiSite Configuration 情報] にあります) が間違った記憶ベイを指定しているために処理されなかったパケット ファイルにある場合もあります。行方不明のパケットが見つかったら、以下のいずれかのことを行います。

- **sync replica -import** コマンドで名前を指定し、行方不明のパケットを処理します。(複数のパケット ファイルが正しい順序でインポートされますが、コマンドライン引数の順序とは関係ありません。)
- 単一の **sync replica -import -receive** コマンドを入力し、記憶バイに累積したすべての更新パケットを処理します。

行方不明のパケットが見つからない場合は、101 ページの「失われたパケットの回復」を参照してください。

パケットを適用できるローカル レプリカがない

インポートが次のメッセージを表示して失敗することがあります。

```
multiutil: エラー: Sync. packet pathname is not applicable to any local replicas.
```

このエラーは、同期サーバーが移動したのに、**host-name** プロパティが **chreplica** コマンドを使用して更新されていない場合に発生します。

host-name プロパティが不正であることを確認するには、**lsreplica** コマンドを使用します。たとえば、このエラーが **bangalore** レプリカで発生した場合は、次のように入力します。

```
multiutil lsreplica -site bangalore -user kumar -p secret -long bangalore
```

```
名前:bangalore; 所属:TELECOMM; ファミリー:PRODA; ホスト:shiphost1; ステータス: NORMAL, NOT CONNECTED; 説明:Production database
```

ホスト名が不正の場合は、**chreplica** コマンドを使用して変更します。次に、ファミリー内のほかのレプリカに更新パケットを送信します。

入カストリームからの読み取りが失敗する

sync replica -import コマンドが失敗して次のようなメッセージが表示された場合、パケットは破損しています。

```
エラー: 入カストリームからの読み取りに失敗しました: No such file or directory
```

パケットを削除し、送信側のレプリカの管理者に連絡してパケットを再作成して再送信してもらいます (101 ページの「失われたパケットの回復」を参照してください)。次にインポートします。

その他の問題

受け取ったレプリカ作成パケットや更新パケットを処理すると、次のような理由で失敗する可能性があります。

- ディスク パーティションがいっぱいになった。
- 受信側のレプリカがロックされている。

- ライセンスのエラーが発生した。
- 複数のインポートが同時に発生した。

1 つのレプリカで複数の **sync replica -import** コマンドを同時に実行することはできません。これらに該当する場合は、問題を解決し、**sync replica -import** コマンドを再入力します。

失われたパケットの回復

パケットまたは更新パケットが生成されたにもかかわらず、その 1 つ以上の宛先にまったく適用されないという状況がいくつかあります。

- 破損したメディアにパケットが格納されている場合や宛先ホストで読み取れない場合。
- ハードディスク障害時にパケット ファイルが失われた場合。
- パケットは破損していないが、ほかのパケットが失われたために適用できない場合。詳細については、99 ページの「入力記憶ベイにパケットが累積する」を参照してください。

sync replica -export コマンドは、生成する更新パケットが正常にデリバリーされることを前提にしています。たとえば、レプリカ **boston_hub** がレプリカ **sanfran_hub** に更新を送信すると、**sync replica** コマンドは **boston_hub** で発生する操作が **sanfran_hub** レプリカにインポートされると見なします。わかりやすくするため、この例では更新パケットにはファミリー内のほかのレプリカで生成された操作は反映されません。

パケットが失われた場合、**boston_hub** はレプリカ **sanfran_hub** の状態に関する見積もりをリセットする必要があります。この訂正の後には、**boston_hub** から **sanfran_hub** に送信される次の更新パケットに **sanfran_hub** が必要とする操作が含まれます。

エポック行をリセットするには、次の手順に従います。

- 1 受信側のレプリカ **sanfran_hub** で、次のコマンドを入力してレプリカのエポック数マトリックスを表示します。

```
multiutil lsePOCH -clan telecom -site sanfran_hub -family PRODA -user jcole -p secret
sanfran_hub
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり
(@goldengate):
```

```
BANGALORE: 950
```

```
BOSTON_HUB: 1300
```

```
SANFRAN_HUB: 2000
```

- 2 送信側のレプリカ **boston_hub** で、この出力結果を **chepoch** コマンドに使用します。これにより、**boston_hub** の **sanfran_hub** に対するエポック数の見積もりが、**sanfran_hub** のエポック数マトリックスでの実際の値に設定されます。

```
multiutil chepoch -clan telecomm -site boston_hub -family PRODA -user bostonadmin
-password secret sanfran_hub bangalore=950 boston_hub=1300 sanfran_hub=2000
Multiutil: サイト 'sanfran_hub' で再現されたサイト 'bangalore' のエポック見
積もりを 950 に変更します [はい | いいえ | 終了]yes

Multiutil: サイト 'sanfran_hub' で再現されたサイト 'boston_hub' のエポック
見積もりを 1300 に変更します [はい | いいえ | 終了]yes

Multiutil: サイト 'sanfran_hub' で再現されたサイト 'sanfran_hub' のエポック
見積もりを 2000 に変更します [はい | いいえ | 終了]yes

Multiutil: サイト 'sanfran_hub' 用の 3 エポック見積もりの変更に成功しました ; 0
個失敗があります。

Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり
(@goldengate):
BANGALORE: 950
BOSTON_HUB: 1300
SANFRAN_HUB: 2000
```

循環重複リンクの削除

レプリカが作成される ClearQuest 環境では、循環重複リンクが作成される可能性があります。その場合、障害とその重複が両方とも重複状態に設定されます。循環重複リンクは、たとえば次の手順で作成できます。

- 1 boston_hub で、Defect1 を Defect2 の重複にします。
- 2 sanfran_hub で、Defect2 を Defect1 の重複にします。
- 3 boston_hub と sanfran_hub を同期します。これで、両方の障害が重複状態になります。

循環重複リンクを削除するには、障害のいずれかをそのマスターであるサイトで重複解除します。

名前の競合の解決

ワークスペース項目 (クエリー、レポート、チャートなど)、ユーザー、グループ、そのほかの状態なしレコードについてはサイト固有の命名規則を適用しない場合、異なるオブジェクトに同じ名前が付けられる可能性があります。

たとえば、1 つの族にある 2 つのサイトのユーザー管理者が、1 つの同期サイクルのうちに同じユーザー名を追加する可能性があります。この場合、レプリカが同期されると、2 人のユーザーが同じ名前になります。

ClearQuest は、内部的に、レコードとワークスペースの名前が一意であることを保証しています。

- 状態を使用するレコードタイプについては、ClearQuest ではデータベース ID 番号を使用して一意性を保証しています。
- 状態なしレコードタイプ (ユーザーとグループを含む) の場合、ClearQuest については、一意のキーを使用し、元のサイト (キーサイト) の名前を保存します。
- ワークスペース項目については、ClearQuest では元のサイト (キーサイト) の名前とワークスペース項目の名前を保存します。

ワークスペース名の競合と ClearQuest Web

2 つのワークスペース項目 (クエリー、レポートなど) が同じ名前である場合、Windows クライアントでも UNIX クライアントでも予期したとおりに、両方の項目がマスターシップの制限とデータベースの権限に従って機能します。ただし、ClearQuest Web ではいずれか一方の項目のみが機能します。混乱を防ぐため、片方または両方の項目の名前を変更してください。

ワークスペース項目の名前の変更

ワークスペース項目を変更するには、現在のレプリカがそのワークスペース項目のマスターである必要があります。ワークスペース項目のマスターであるレプリカを特定するには、目的の項目を右クリックし、[マスターシップ] をクリックします。

ワークスペース項目の名前を変更するには

- 1 ワークスペース項目を右クリックし、[名前の変更] をクリックします。
- 2 反転表示された領域に新しい名前を入力し、[入力] をクリックします。

あいまいなワークスペース項目の操作

multiutil コマンドを使用して名前の衝突があるワークスペース項目を操作する必要がある場合は、そのキーサイト名を参照する必要があります。キーサイト名とは、ワークスペース項目が作成されたサイトの名前です。例を次に示します。

```
"Workspace:¥Public Queries¥Project Report<keysite-name>"
```

次の例では、オブジェクトセレクタでキーサイト名を使用しています。

```
multiutil describe -clan telecomm -site tokyo -family PRODA -user tokyoadmin -password  
secret "workspace:Public Folder¥Project Report<boston_hub>"
```

Multiutil: 'workspace:Public Queries¥Project report<boston_hub>' のマスターシップは 'boston_hub' です。

状態なしレコード タイプ名の競合の解消

状態なしレコード名の競合を解消するには、いずれかのレコード名を変更する必要があります。

レコード名の変更

名前の競合がある状態なしレコードの名前を変更するには

- 1 問題が発生しているレコードを検索します。104 ページの「名前の競合がある状態なしレコードの検索」を参照してください。
- 2 目的のレコード名を変更します。レコードを変更するには、そのレコードのマスターシップが必要です。

状態なしレコード名を変更するには、そのレコードの一意のキーとして使用されているフィールドを変更する必要があります。このためには、状態を変更せずにレコードを変更できるアクションをスキーマに使用します。

- 3 ファミリを同期します。

レコードの一意性の確認

レコードが一意であることを保証するために、状態なしレコード タイプは **ratl_keysite** フィールドを使用します。**ratl_keysite** フィールドは、オブジェクトが作成されたサイトの名前を格納する内部システム フィールドです。

たとえば、**NetInc** という名前の新しい顧客が、同期までの間に 2 つのレプリカで作成されたとします。各レプリカが同期されると、これらは同じ名前を持つ 2 つの顧客レコードとして表示されます。一意であることを保証するために、**ClearQuest** は **ratl_keysite** フィールドを参照します。

describe コマンドまたは **chmaster** コマンドを使用してあいまいなレコードを操作するには、キーサイト名 (元のサイト名) を参照する必要があります。たとえば、次のように指定します。

```
customer:NetInc<keysite-name>
```

次の例では、オブジェクト セレクタでキーサイト名を使用しています。

```
multiutil describe -clan telecom -site tokyo -family PRODA -user masako -password secret  
customer:NetInc<boston_hub>
```

Multiutil: 'customer:NetInc<boston_hub>' のマスターシップは 'boston_hub' です。

名前の競合がある状態なしレコードの検索

同じ名前を持つ状態なしレコードを検索するためのクエリーに、**ratl_keysite** フィールドを使用できます。名前の競合がある状態なしレコードのクエリーを実行する場合は、次のガイドラインに従います。

- 各状態なしレコード タイプのクエリーを作成する際、**ratl_keysite** フィールドを表示フィールドとフィルタの両方として使用します。
- クエリーで重複するレコード名が見つかった場合は、承認済みの、サイト固有の命名規則に従ってそれらの名前を変更します。レコードを変更するには、現在のレプリカがそのレコードのマスターである必要があることに注意してください。

レコードの表示や変更を容易にするため、名前の競合が発生しそうな任意の状態なしレコードタイプのフォームに **ratl_keysite** フィールドを追加できます。詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

ユーザー名やユーザー グループ名の競合の特定

あいまいなユーザー名でログオンするには、ユーザー ログオン名の一部にキーサイト名を使用します (たとえば、**username<keysite-name>** のように入力します。ここで、**keysite-name** は、そのユーザーが作成されたサイトです)。キーサイト名を使用せずにあいまいなユーザー名を使用してログオンすると、無効なログオンによるエラーが発生します。[詳細] をクリックすると、次のエラーが表示されます。

ユーザー名 'xxx' は曖昧です ; 名前変更するか '<' SITE '>' で修飾してから、作業を進めてください。

ユーザー名の変更

ClearQuest Designer では、名前の競合があるユーザーのユーザー情報を変更しようとする、名前に < と > という文字が含まれていることが原因で次のようなエラー メッセージが表示されます。

エラー ! 文字列の値 ("DupUser<SITE1>") は無効です : 名前に次の文字を含めないでください : ! " # \$ % & ' () * + , . / : ; < = > ? @ [\] ^ _ { | } ~

そのユーザーの名前を変更する必要があります。このとき、ユーザー情報は **Name** フィールドを除いて変更できません。ユーザー名を、文字 < > を含まない一意の名前にすれば変更できるようになります。

ユーザー グループの名前の変更や削除を行うことはできません。

ユーザー名を変更するには

- 1 [ツール] メニューの [ユーザー管理] をクリックします。
- 2 [ユーザー管理] ダイアログ ボックス内で、変更するユーザーをダブルクリックします。
- 3 [ユーザー プロパティ] ダイアログ ボックスで、ユーザー名を変更します。
- 4 [OK] をクリックします。

- 5 [DB アクション] メニューの [アップグレード] をクリックして、関連付けられたユーザー データベースをアップグレードします。
- 6 [アップグレード] ダイアログ ボックスで、アップグレードするユーザー データベースを選択します。
- 7 [OK] をクリックします。
- 8 [OK] をクリックします。
- 9 新しいマスター レプリカの管理者は、変更を含む同期パケットを受信したらデータベースをアップグレードする必要があります。詳細については、『Rational ClearQuest 管理ガイド』を参照してください。

あいまいなユーザーとユーザー グループに対する multiutil の使用法

`describe` コマンドまたは `chmaster` コマンドを使用して別のユーザーまたはグループと同じ名前を持つユーザーまたはグループを操作する必要がある場合は、それぞれのキーサイト名 (元のサイト名) を参照する必要があります。

目的のユーザーがマスターになっている場所を検索するには、`describe` コマンドを使用します。キーサイトが `boston_hub` レプリカである例を次に示します。

```
multiutil describe -clan telecom -site tokyo -family PRODA -user masako -password secret  
user:jsmith<boston_hub>
```

Multiutil: 'user:jsmith<boston_hub>' のマスターシップは 'boston_hub' です。

データベースのレプリカを作成した後のデータベース登録の更新

作業スキーマ リポジトリ サイトに新しいユーザー データベースを追加する場合は、新しいユーザー データベースのレプリカを作成してからユーザーをそこに登録することをお勧めします。

メモ: ただし、ユーザーがすべてのデータベースに登録されている場合は例外です。すべてのデータベースに登録されているユーザーは、新しいデータベースに問題なくアクセスできます。

新しいデータベースのレプリカが作成される前に登録されたユーザーは、データベース登録が作業スキーマ リポジトリ サイトで更新されない限り新しいデータベース レプリカにログインできません。

- 1 ClearQuest Designer を使用して、作業スキーマ リポジトリにログインします。この操作は作業スキーマ リポジトリで実行する必要があります。また、少なくともユーザー管理者権限が必要です。
- 2 ClearQuest Designer で [ツール] メニューの [ユーザー管理] をクリックします。

- 3 目的のレプリカにログオンできないユーザーを選択します。
- 4 [DB Subscriptions] をクリックします。
- 5 [Database Subscriptions] ダイアログ ボックスで、[OK] をクリックします。
- 6 変更確認のダイアログ ボックスで、[はい] をクリックします。
- 7 登録の問題がある各ユーザーについて、手順 3 から手順 6 までを繰り返します。
- 8 変更確認のダイアログ ボックスで、[はい] をクリックします。
- 9 [ユーザー管理] ダイアログ ボックスで、[Upgrade the user DB] をクリックします。
- 10 [Select Site] ダイアログ ボックスで、アップグレードするユーザー データベースを選択します (作成したばかりのレプリカを選択します)。
- 11 登録の問題があるすべてのサイトに対して変更を同期します。
- 12 ユーザーが、レプリカが作成されたデータベースにのみ登録されている場合は、そのユーザーのデータベース登録は新しいユーザー データベースのレプリカがあるすべてのサイトで表示できます。このレプリカがないサイトでは、そのユーザーのデータベース登録を表示できません。

データベース レプリカの復元

レプリカは失われることがあります。失われる原因は通常、ハードウェア障害 (ディスク クラッシュなど)、ソフトウェア障害 (OS レベルでのファイル システムの破損など)、または人為的なエラーです。非レプリカ データベースが失われた場合、バックアップから最新のコピーを復元して開発作業を再開できます。バックアップをとってから障害が発生するまでに行われた変更は、回復できません。

同様に、レプリカを失った場合は、バックアップから最新のコピーを復元できます。しかし、その方法はより複雑になります。

- バックアップから障害が発生するまでの間に行われた作業の一部は回復できる場合があります。更新パケットで操作の一部が別のレプリカに送信された場合、これらの操作を検索してインポートする必要があります。
- 復元されたレプリカのコピーは、有効期限が過ぎています。復元されたレプリカでの開発を進める前に、このレプリカをファミリー内のほかのレプリカと整合性を持たせる必要があります。整合性を持たせることができない場合、修復不可能な損傷を受けることがあります。

この手順は非常に工数がかかるので、重大な損傷が発生した場合に使用します (たとえば、レプリカの格納されているディスクが使用できない場合など)。

バックアップからのレプリカの復元

バックアップからレプリカを復元するには

- 1 製造元データベースのツールを使用して、レプリカ データベースのコピーをバックアップから復元します。
- 2 **restorereplica** コマンドを使用して復元プロシーダを開始します。

このコマンドは、レプリカに特別なロックを設定します。この時点から手順 6 が完了するまでの間に、**syncreplica -import** コマンドはロックを一時的に調整して更新の適用を許可し、次にフル ロックを復元します。この間、**syncreplica -import** のみがレプリカを修正できます。

- 3 すべての更新パケットが宛先レプリカで処理されていることを確認します。
- 4 復元されたレプリカで、ファミリー内のその他すべてのレプリカに対する更新パケットを生成し、シブリング レプリカにパケットを送信します。

標準の同期メソッドを使用してパケットを送信できます。より迅速にレプリカを回復するには、**syncreplica -export -fship** を使用してパケットを作成します。

レプリカは特別な復元状態にあるので、送信される各更新パケットには確認応答の返信について特別な要求が含まれています。また、レプリカの以前のエポック数も含まれますが、これは手順 1 の復元によって現在のエポック数に変更されています。各宛先レプリカは、これらの数を使用してレプリカの行をロールバックします。

- 5 ファミリー内の各レプリカが更新パケットを復元されたレプリカに送信するのを待ちます。手順 4 に従って、更新パケットの作成とデリバリーを迅速に行うことができます。

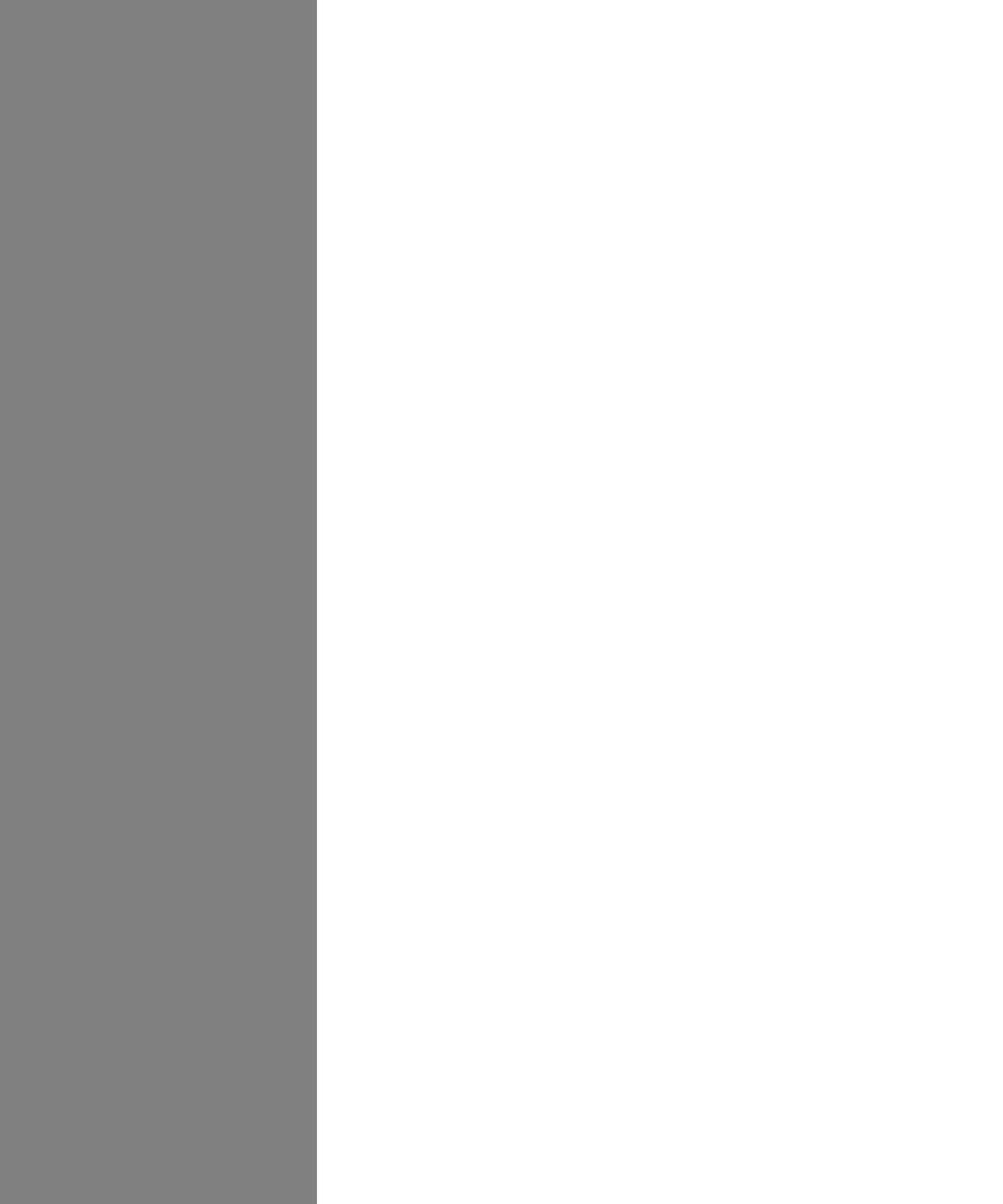
全体として、これらの更新パケットには、バックアップの時点からレプリカがその記憶を失う前に送信した最後の更新までの間に発生したすべての操作が、レプリカで開始された操作と共に含まれています。(パケットには、ほかのレプリカで発生したより新しい操作も含まれています。)さらに、着信した各パケットには送信ホストから要求されている確認応答の返信が含まれています。

- 6 **syncreplica -import** を使用して、着信した更新パケットを処理します。レプリカがファミリー内のほかのレプリカから確認応答の返信を受け取ると、**syncreplica -import** はレプリカの復元を完了したことを次のように報告します。

```
Database <name> is unlocked after restoration.
```

これで、レプリカでの開発作業を再開できるようになりました。

MultiSite リファレンス ページ



MultiSite リファレンス ページ

11

管理ガイドのこの章は、MultiSite リファレンス ページです。

activate

データベース セットのレプリカの作成を準備します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
activate [ -dbset dbset-name ] -u ser user-name [ -p assword ] password  
        -cl an clan-name -site site-name -host hostname
```

説明

activate コマンドは、レプリカを作成するデータベース セット (スキーマ リポジトリとそのユーザー データベース) を準備します。データベース セットをアクティブにする前に、セット内のすべてのユーザー データベースをアップグレードして、**ClearQuest** と同じバージョンにする必要があります。

データベース セットをアクティブにするときに、族名とサイト名を指定する必要があります。データベース セットの論理名は **CQMS.clan-name.site-name** に変更されます。アクティブになったデータベース セットは、ほかの **multiutil** コマンドでもアクセスできるようになり、**mkreplica** コマンドで、データベース セット内の 1 つ以上のユーザー データベースのレプリカを作成できます。

メモ: データベース セットをアクティブにする必要があるのは、最初のレプリカを作成する前の一度だけです。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

-dbset dbset-name

アクティブにするデータベース セットの名前です。ClearQuest のインストール環境にデータベース セットが 1 つしかない場合は、この引数を省略できます。データベース セット名は、メンテナンス ツールで [既存の接続] の下に表示されます。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

族とサイトの指定

デフォルト

なし

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

ホストの指定

デフォルト

なし

-host hostname

同期サーバーの名前です。同期サーバーとは、レプリカの記憶領域と戻りベイがあるホストのことです。同期サーバーに Rational Shipping Server をインストールしている必要があります。

例

次の例では、読みやすくするために長い行を途中で改行していますが、実際には、各コマンドを 1 行で入力してください。

- デフォルトのデータベース セットをアクティブにします。族名 **telecomm** とサイト名 **boston_hub** を指定します。このサイトは、**minuteman** を同期サーバーとして使用します。
multiutil activate -dbset CLSIC -user susan -p passwd -clan telecomm -site boston_hub -host minuteman

関連項目

mkreplica

chepoch

エポック数の見積もりを変更します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
chepoch [ -cl-an clan-name ] [ -site site-name ] -family family-name  
-u-ser user-name [ -p-assword ] password  
{ [ -f-orce ] replica [ replica=value ... ] }
```

説明

このコマンドは、ほかのレプリカに対するエポック数の見積もりを変更します。レプリカ自体のエポック数はそのレプリカの実状態を記録しているので変更できません。エポック数の詳細については、12 ページの「操作ログ」を参照してください。chepoch を使用したシナリオについては、94 ページの「oplog 項目が見つからない」と 101 ページの「失われたパケットの回復」を参照してください。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリの指定

デフォルト

族 : このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト : 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリー：デフォルトはありません。ファミリーを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ：ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ：ファミリー名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

対話形式のプロンプトの非表示

デフォルト

各変更を確認する必要があります。

-f-orce

確認の手順を省略します。

変更の指定

デフォルト

エポック数の見積もりを変更するレプリカを指定する必要があります。**chepoch** は **replica=value** の組を標準入力から 1 行につき 1 つずつ読み取ります。**lsepoch** の出力をコピー アンド ペーストするか、以下に示す形式でデータを入力できます。余分な空白も使用できます。入力を終了するには、行頭でピリオド (.) を入力して改行 (<CR>) します。

replica

エポック数の見積もりを変更するレプリカのサイト名。エポック数の見積もりの変更とは、現在のレプリカによる **replica** の状態の見積もりを変更することです。

replica=value

以下に示す引数を 1 つ以上取ります。

replica エポック数マトリックスの列です。この引数は、先行する **replica** 引数と共に、マトリックス内の特定の場所を指定します。

value 指定されたマトリックスの場所に入力される新規エポック数です。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **sanfran_hub** レプリカに対する **boston_hub** レプリカの行で、2 列のエポック見積もりを変更します。

```
multiutil chepoch -clan telecomm -site boston_hub -family SAMPL -user susan -p passwd  
sanfran_hub sanfran_hub=100 boston_hub=350
```

```
Multiutil: サイト 'sanfran_hub' で再現されたサイト 'sanfran_hub' の  
エポック見積もりを 100 に変更します [はい | いいえ | 終了] はい
```

```
Multiutil: サイト 'boston_hub' で再現されたサイト 'sanfran_hub' の  
エポック見積もりを 350 に変更します [はい | いいえ | 終了] はい
```

```
Multiutil: サイト 'sanfran_hub' 用の 2 エポック見積もりの変更に成功しました ;  
0 個失敗があります。
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり  
(@goldengate):  
BOSTON_HUB: 350  
SANFRAN_HUB: 100
```

- 前の例と同様ですが、**-force** を使用して確認の手順を表示しないようにします。

```
multiutil chepoch -clan telecomm -site boston_hub -family SAMPL -user susan -p passwd  
sanfran_hub sanfran_hub=100 boston_hub=350 -force
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり  
(@goldengate):
```

```
BOSTON_HUB: 350  
SANFRAN_HUB: 100
```

関連項目

lsepoch、recoverpacket、restorereplica

chmaster

オブジェクトのマスターシップを転送します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
chmaster [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
        -u-ser username [ -p-assword ] password new-master-replica  
        { { entity-selector... | { -all [ -l-ong ] | -workingmaster }  
        [ -force obsolete-replica | -forceall ] }  
        }
```

説明

このコマンドは、レプリカ間で 1 つ以上のオブジェクトのマスターシップを転送します。この転送がすぐに反映されるのは現在のレプリカだけです。ほかのレプリカに対しては、通常の更新パケットの交換を通じて、マスターシップの転送が通知されます。

制約事項

権限チェック: スーパー ユーザー権限が必要です。

マスターシップ: 現在のレプリカがオブジェクトのマスターである必要があります。**-force** または **-forceall** を使用してこの制約を無効にできますが、特別な場合を除いて、これらのオプションは使用しないでください。

オプションと引数

族、サイト、ファミリーの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリー: デフォルトはありません。ファミリーを指定する必要があります。

-clan clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-family family-name

ユーザー データベース ファミリ: ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ: ファミリ名は **MASTR** です。**-workingmaster** オプションを使用する場合は、**MASTR** を指定する必要があります。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

新しいマスター レプリカの指定

デフォルト

なし

new-master-replica

マスタースリップの転送先のレプリカの名前です。

オブジェクトの指定

デフォルト

なし特定のオブジェクト (**entity-selector**)、レプリカ内のすべてのオブジェクト (**-all**)、作業スキーマ リポジトリのみ、のいずれかを指定する必要があります。

entity-selector

マスターシップを変更するオブジェクトを指定します。次のオブジェクトのマスターシップを変更できます。

オブジェクト	構文
レコード	record-type:record-id
ユーザーまたはグループ	user:username group:group-name
共用ワークスペースの アイテム	"workspace:Public Queries¥folder-name¥query-name"
個人用ワークスペースの アイテム	"workspace:Personal Queries(username)¥folder-name¥query-name"
名前が一意ではない状態なし レコード	record-type:record-id<keysite-name>
名前が一意ではないワークス ペースのアイテム	"workspace:query-name<keysite-name>"
名前が一意ではないユーザー またはグループ	user:username<keysite-name> group:group-name<keysite-name>

名前を一意にする方法については、102 ページの「名前の競合の解決」を参照してください。

-all [-l-long]

-clan、**-site**、**-family** で指定したレプリカ内にあり、そのレプリカがマスターであるすべてのオブジェクトのマスターシップを、**new-master-replica** に転送します。エラーが発生してもコマンドは引き続き実行され、実行終了後に一部のマスターシップの変更 に失敗したことが報告されます。

-long を指定すると、**chmaster** はマスターシップが変更されるオブジェクトを表示します。

メモ : 作業スキーマ リポジトリのマスターシップを変更するには、**-workingmaster** を使用します。

-workingmaster

指定したサイトに作業スキーマ リポジトリのマスターシップを転送します。このオプションは、作業スキーマ リポジトリのサイトでのみ使用できます。

このオプションを使用する場合は、**-family MASTR** を指定する必要があります。

-f-orc obsolete-replica

警告 : **-force** オプションの使い方を誤ると、ファミリー内のレプリカ間で相違が発生する可能性があります。

-force を指定すると、**chmaster** は **obsolete-replica** で指定されたレプリカ内のすべてのオブジェクトのマスターシップを転送します。この形式の **chmaster** は、レプリカ **obsolete-replica** が (たとえば、間違って削除されて) 使用できなくなった場合にのみ使用してください。

-f-orc all

警告 : **-forceall** オプションの使い方を誤ると、ファミリー内のレプリカ間で相違が発生する可能性があります。

-forceall を指定すると、**chmaster** は、現在のレプリカがそのオブジェクトのマスターでない場合でも、指定されたレプリカにオブジェクトのマスターシップを転送します。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- ユーザー **admin** のマスターシップを **boston_hub** レプリカから **sanfran_hub** に転送します。

```
multiutil chmaster -clan telecom -site boston_hub -family DEV -user susan -p passwd  
sanfran_hub user:admin
```

```
Multiutil: The mastership of record 'admin' of type 'user' has  
been changed from 'BOSTON_HUB' to 'SANFRAN_HUB'.
```

Multiutil: 一部のユーザーまたはグループのマスターシップはこのサイトから転送されています。ローカルのユーザー管理者は、新規のマスター サイト '**sanfran_hub**' にあるユーザー データベースを、変更がほかのユーザー データベースに対して参照可能になる前に、更新する必要があります。

- 作業スキーマ リポジトリがある **tokyo** レプリカで、すべてのスキーマと作業スキーマ リポジトリ タスクのマスターシップを **sydney** レプリカに転送します。

```
multiutil chmaster -clan testclan -site tokyo -family MASTR -user masako -p passwd  
sydney -workingmaster
```

```
Multiutil: 動作中のマスターが 'TOKYO' から 'SYDNEY' に変更されました。
```

- DEV データベース内にあり、**sanfran_hub** レプリカがマスターであるすべてのオブジェクトのマスターシップを **boston_hub** レプリカに転送します。

```
multiutil chmaster -clan telecom -site sanfran_hub -family DEV -user jcole -p passwd
boston_hub -all
```

Multiutil: 変更されたオブジェクトの総数: 5。

- 前の例と同様ですが、**-long** オプションを使用します。

```
multiutil chmaster -clan telecom -site sanfran_hub -family DEV -user jcole -p passwd
boston_hub -all -long
```

Multiutil: データベース 'DEV' 内の以下のオブジェクトのマスターシップが、
'SANFRAN_HUB' から 'BOSTON_HUB' に変更されました。

Multiutil: タイプ: customer、表示名: John Smith。

Multiutil: タイプ: customer、表示名: Ethan Hunt。

Multiutil: タイプ: customer、表示名: Jane Smith。

Multiutil: タイプ: customer、表示名: Anne Johnson。

Multiutil: タイプ: customer、表示名: Joe Lee。

- **boston_hub** レプリカで、**bangalore** レプリカがマスターであるすべてのアイテムのマスターシップを **boston_hub** レプリカに転送します。**bangalore** は使用できなくなったものとします。

```
multiutil chmaster -clan telecom -site boston_hub -family DOC -user susan -password
passwd boston_hub -all -force bangalore
```

ユーザーとグループのマスターシップを、**bangalore** スキーマ リポジトリから
boston_hub に変更します。

```
multiutil chmaster -clan telecom -site boston_hub -family MASTR -user susan
-password passwd boston_hub -all -force bangalore
```

作業スキーマ リポジトリのマスターシップを、**bangalore** から **boston_hub** に変更します。

```
multiutil chmaster -clan telecom -site boston_hub -family MASTR -user susan
-password passwd boston_hub -workingmaster -force bangalore
```

- **boston_hub** で **-forceall** を使用して、ユーザー **admin** のマスターシップを **sanfran_hub** から **tokyo** に変更します。

```
multiutil chmaster -clan telecom -site boston_hub -family DEV -user susan -p passwd
tokyo user:admin -forceall
```

Multiutil: The mastership of record 'admin' of type 'user' has
been changed from 'SANFRAN_HUB' to 'TOKYO'.

関連項目

describe、syncreplica

第9章 マスターシップの管理

chreplica

サイト内のレプリカのプロパティを変更します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
chreplica [ -cl-an clan-name ] [ -site site-name ]  
          -u-ser username [ -p-assword ] password  
          [ -host hostname | -size id-block-size | -thres-hold id-block-threshold ] replica-selector
```

説明

このコマンドを使用して、サイト内のすべてのレプリカについて、同期サーバー ホスト情報または ID ブロック割り当てを変更することができます。詳細については、75 ページの「同期サーバーの移動または名前変更」と 76 ページの「レプリカに対する ID ブロック割り当ての変更」を参照してください。

制約事項

なし

オプションと引数

族とサイトの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

新しい値の指定

デフォルト

変更するサイトを指定する必要があります。デフォルトの ID ブロック サイズは 4096、デフォルトのしきい値は 25 パーセントです。

-host hostname

新しい同期サーバー (Rational Shipping Server がインストールされているサーバー) の名前です。

-size id-block-size

ID ブロックのサイズです。1 ～ 1023 までの任意の数値を入力できます。
id-block-size の値を 100 倍した値が、実際の ID ブロックのサイズです。たとえば、サイズが 30,000 の ID ブロックを指定するには「300」と入力し、サイズが 25,000 の ID ブロックを指定するには「250」と入力します。

-thres-hold id-block-threshold

レプリカに割り当てるレコード ID 番号の数です。**id-block-threshold** には整数のパーセント値を指定します。1 ～ 63 までの任意の数値を入力できます。使用可能なレコード ID 数の残りが、現在の ID ブロック サイズに指定されているパーセント値を下回ると、追加のブロックが割り当てられます。

replica-selector

変更するサイトです。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **sanfran_hub** レプリカを同期サーバー **goldengate** に関連付けます。
`multiutil chreplica -clan telecom -site sanfran_hub -user jcole -p passwd -host goldengate sanfran_hub`
- **sanfran_hub** レプリカを同期サーバー **goldengate** に関連付け、10,000 という ID ブロックサイズを指定します。
`multiutil chreplica -clan telecom -site sanfran_hub -user jcole -p passwd -host goldengate -size 100 sanfran_hub`
- **sydney** レプリカを同期サーバー **taronga** に関連付け、55 パーセントという割り当てしきい値を指定します。
`multiutil chreplica -clan testclan -site sydney -user bfife -p passwd -host taronga -threshold 55 sydney`

関連項目

chmaster、syncreplica

control_panel

Windows での Rational Shipping Server 電子メール通知用のパラメータを設定します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
Windows

構文

```
control_panel -admin admin-email -smtp smtp-server-host  
[ -enable_shipping_server_email_notification ]
```

説明

このコマンドを使用すると、Windows プラットフォームで Rational Shipping Server と ClearQuest MultiSite を使用している場合に、電子メール通知を有効にすることができます。

ClearCase MultiSite を使用せずに ClearQuest MultiSite を使用している場合は、このコマンドで、Rational Shipping Server の操作に関する電子メール通知を受け取るための電子メール アドレスとサーバーを指定します。

ClearQuest MultiSite と ClearCase MultiSite を使用している場合は、通知電子メール用に両方の製品で同じ電子メール アドレスが使用されます。

- ClearQuest MultiSite と ClearCase MultiSite の両方で、Rational Shipping Server の操作に関する通知を同じ電子メール アドレスで受け取る場合は、このコマンドは使用しないでください。この場合は、MultiSite コントロール パネルのオプションを使用して電子メール通知を設定します。手順については、『Rational ClearCase MultiSite 管理ガイド』を参照してください。
- Rational Shipping Server の操作に関する通知を ClearQuest MultiSite から送信するときに、異なる電子メール アドレスを使用するには、使用する電子メール アドレスをこのコマンドで指定します。

制約事項

なし

オプションと引数

-admin admin-email

ここで指定した電子メール アドレスを使用して、Rational Shipping Server から送信されるエラーと情報に関する電子メール通知を送受信します。電子メール アドレスは、次の形式で指定します。tld は最上位のドメインです。

username@domain.tld

-smtp smtp-server-host

-admin で指定した電子メール アドレスで使用する SMTP ホストの名前です。

ClearCase MultiSite を使用していて、ClearQuest MultiSite で別の電子メール アドレスを使用する場合は、ClearCase での通知用に使用しているものと同じ SMTP サーバーを使用する必要があります。

-enable_shipping_server_email_notification

この引数を使用して、Rational Shipping Server を使用する ClearQuest MultiSite 操作に関する電子メール通知を有効にします。

デフォルト: このオプションが指定されていない場合、ClearQuest MultiSite では ClearCase MultiSite 用の電子メール設定を使用します。ClearCase MultiSite をインストールしていない場合は、このオプションを使用して Rational Shipping Server の電子メールによる通知を有効にする必要があります。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- ClearQuest MultiSite での Rational Shipping Server の電子メール通知を有効にします (ClearCase MultiSite はインストールされていません)。

```
multiutil control_panel -admin susan@purpledoc.com -smtp mailsrv0.purpledoc.com  
-enable_shipping_server_email_notification
```

- ClearCase MultiSite も使用している場合に ClearQuest MultiSite で使用する別の電子メール アドレスを設定します。

```
multiutil control_panel -admin susan@purpledoc.com -smtp mailsrv0.purpledoc.com
```

関連項目

MultiSite コントロール パネル

describe

オブジェクトのマスター レプリカを表示します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
describe [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
        -u-ser username [ -p-assword ] password  
        [ -all | -local | object-selector ... ]
```

説明

このコマンドは、レプリカ内の 1 つまたは複数のオブジェクトのマスター レプリカを表示します。レコードのマスターであるレプリカを判別するには、**ratl_mastership** フィールドの値を調べてください。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリの指定

デフォルト

族：このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト：現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ：デフォルトはありません。ファミリを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ : ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ : ファミリ名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

情報を表示するオブジェクトの指定

デフォルト

指定したレプリカ内のすべてのオブジェクトのマスター レプリカを表示します。

-all

指定したファミリ内の各アイテムのマスター レプリカを表示します。

-local

指定したファミリの現在のサイトがマスターであるオブジェクトのみを表示します。

object-selector ...

情報を表示するオブジェクトです。object-selector を、次のいずれかの形式で指定します。

オブジェクト	構文
レコード	record-type:record-id
ユーザーまたはグループ	user:username group:group-name
共用ワークスペースのアイテム	"workspace:Public Queries¥folder-name¥query-name"

オブジェクト	構文
個人用ワークスペースのアイテム	"workspace:Personal Queries(username)¥folder-name¥query-name"
名前が一意ではない状態なしレコード	record-type:record-id<keysite-name>
名前が一意ではないワークスペースのアイテム	"workspace:query-name<keysite-name>"
名前が一意ではないユーザーまたはグループ	user:username<keysite-name> group:group-name<keysite-name>

名前を一意にする方法については、102 ページの「名前の競合の解決」を参照してください。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **sanfran_hub** レプリカで、**Customer** オブジェクト「Jane Smith」のマスター レプリカを表示します。

```
multiutil describe -clan telecomm -site sanfran_hub -family DEV -user jcole -p passwd
Customer:"Jane Smith"
```

Multiutil: 'Customer:Jane Smith' のマスターシップは 'BOSTON_HUB' です。

- **sydney** レプリカ内のすべてのオブジェクトのマスター レプリカを表示します。

```
multiutil describe -clan testing -site sydney -family TEST -user bfife -p passwd
```

Multiutil: 'Defect:TEST000000001' のマスターシップは 'TOKYO' です。

Multiutil: 'Defect:TEST000000002' のマスターシップは 'TOKYO' です。

Multiutil: 'Defect:TEST000000004' のマスターシップは 'TOKYO' です。

Multiutil: 'Email_Rule:New Submissions' のマスターシップは 'TOKYO' です。

Multiutil: 'bucket:Personal Queries' のマスターシップは 'TOKYO' です。

関連項目

chmaster、syncreplica

dumpoplog

レプリカの操作ログの内容を表示します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
dumpoplog [ -cl-an clan-name ] [ -site site-name ] -family family-name  
          -u-ser username [ -p-assword ] password  
          [ -l-ong | -s-hort ] [ -at replica ]  
          [ oplog-ID... | [ -from oplog-ID ] [ -to oplog-ID ] ]  
          [ -since date-time ] [ -reverse ]
```

説明

dumpoplog を使用して、レプリカの操作ログ (oplog) の内容を表示します。oplog は、レコード変更やスキーマ修正などを含め、すべてのデータベース トランザクションを追跡します。各 oplog 項目には oplog ID があります。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリー：デフォルトはありません。ファミリーを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ：ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ：ファミリー名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

表示する情報の指定

デフォルト

形式を指定していない場合、**-short** 形式が使用されます。すべての **oplog** 項目を表示します。

-l-ong | -s-hort

-long を指定すると、パケット データに適用されるスキーマ リビジョンのバージョン情報を含め、**oplog** のすべての列が表示されます。**-short** を指定すると、実行された各データベース操作が表示されます。

-at replica

指定したサイトで発生した **oplog** 項目を表示します。

oplog-ID...

指定した **oplog** 項目を表示します。

-from oplog-ID

oplog-ID から最後の項目または **-to** で指定した項目までの **oplog** 項目を表示します。**oplog ID** は整数で指定します。

-to oplog-ID

1 または **-from** で指定した項目から **oplog-ID** までの **oplog** 項目を表示します。**oplog ID** は整数で指定します。

-since date-time

date-time 以降のすべての **oplog** 項目を表示します。**date-time** 引数は、次のいずれかの形式で指定できます。

date.time | date | time | now

各パラメータの内容は次のとおりです。

date := day-of-week | long-date

time := h[h]:m[m]:s[s] [UTC [[+ | -]h[h]:m[m]]]]

day-of-week := today |yesterday |Sunday | ... |Saturday |Sun | ... |Sat

long-date := d[d]-month[-[yy]yy]

month := January |... |December |Jan |... |Dec

time は、地域のタイムゾーンに合わせた 24 時間形式で時刻を指定します。**time** を省略した場合、デフォルト値は 00:00:00 です。**date** を省略した場合、デフォルト値は **today** です。世紀、年、または特定の日付を省略すると、最新のものが使用されます。タイムゾーンにかかわらず同じ時刻を適用するには、**UTC** を指定します。**UTC** 時間に対して正のオフセットを指定するにはプラス (+) 演算子を、負のオフセットを指定するにはマイナス (-) 演算子を使用します。時または分のオフセットを付けずに **UTC** を指定すると、デフォルトの設定はグリニッジ標準時 (GMT) になります (協定世界時 (UTC) 1970 年 1 月 1 日より前の日付は無効です)。

例を次に示します。

```
22-November-2002
sunday
yesterday.16:00
8-jun
13:00
today
9-Aug.10:00UTC
```

-reverse

oplog 項目の表示順序を逆にします。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **boston_hub** レプリカの **DOC** ファミリに関連付けられた操作の **oplog** を表示します。

```
multiutil dumpoplog -clan telecom -site boston_hub -family DOC -user susan -p passwd
```

```
★ Item    1, 22-Jan-2002.10:50:21 TABLE_PACKLET      (epoch
BOSTON_HUB.1    initiated 22-Jan-2002.10:50:21)
Item    2, 22-Jan-2002.10:50:21 TABLE_PACKLET      (epoch
BOSTON_HUB.2    initiated 22-Jan-2002.10:50:21)
Item    3, 22-Jan-2002.10:50:21 TABLE_PACKLET      (epoch
BOSTON_HUB.3    initiated 22-Jan-2002.10:50:21)
Item    4, 22-Jan-2002.10:50:22 TABLE_PACKLET      (epoch
BOSTON_HUB.4    initiated 22-Jan-2002.10:50:22)
Item    5, 22-Jan-2002.10:50:22 TABLE_PACKLET      (epoch
BOSTON_HUB.5    initiated 22-Jan-2002.10:50:22)
Item    6, 22-Jan-2002.11:06:21 ACTION_PACKLET      (epoch
BOSTON_HUB.6    initiated 22-Jan-2002.11:06:21)
Item    7, 22-Jan-2002.11:10:35 TABLE_PACKLET      (epoch
BOSTON_HUB.7    initiated 22-Jan-2002.11:10:35)
Item    8, 22-Jan-2002.11:10:36 EXPORT_PACKLET      to SANFRAN_HUB
```

- **sanfran_hub** レプリカで行われた **sanfran_hub** の操作の **oplog** を表示します。

```
multiutil dumpoplog -clan telecom -site sanfran_hub -family DEV -user jcole -p passwd
-short -at sanfran_hub
```

```
Item    3, 07-Feb-2002.11:20:29 ACTION_PACKLET      (epoch
SANFRAN_HUB.1    initiated 07-Feb-2002.11:20:29)
Item    4, 07-Feb-2002.11:23:24 TABLE_PACKLET      (epoch
SANFRAN_HUB.2    initiated 07-Feb-2002.11:23:24)
Item    6, 07-Feb-2002.11:24:27 ACTION_PACKLET      (epoch
SANFRAN_HUB.3    initiated 07-Feb-2002.11:24:26)
Item    7, 07-Feb-2002.11:24:49 TABLE_PACKLET      (epoch
SANFRAN_HUB.4    initiated 07-Feb-2002.11:24:49)
```

- **bangalore** レプリカで 2002 年 1 月 28 日に行われたすべての操作の **oplog** を表示します。

```
multiutil dumpoplog -clan telecom -site bangalore -family DOC -user masako -p passwd
-short -since 28-Jan-2002
```

```
Item    1, 28-Jan-2002.10:11:37 TABLE_PACKLET      (epoch
BOSTON_HUB.1    initiated 28-Jan-2002.10:11:37)
Item    2, 28-Jan-2002.10:11:37 TABLE_PACKLET      (epoch
BOSTON_HUB.2    initiated 28-Jan-2002.10:11:37)
Item    3, 28-Jan-2002.10:11:37 TABLE_PACKLET      (epoch
BOSTON_HUB.3    initiated 28-Jan-2002.10:11:37)
Item    4, 28-Jan-2002.10:11:37 TABLE_PACKLET      (epoch
BOSTON_HUB.4    initiated 28-Jan-2002.10:11:37)
Item    5, 28-Jan-2002.10:11:37 TABLE_PACKLET      (epoch
BOSTON_HUB.5    initiated 28-Jan-2002.10:11:37)
Item    6, 28-Jan-2002.10:11:37 TABLE_PACKLET      (epoch
BOSTON_HUB.6    initiated 28-Jan-2002.10:11:37)
```

- **boston_hub** レプリカの 3 ～ 6 までの **oplog** 項目を表示します。

```
multiutil dumpoplog -clan telecom -site boston_hub -family DEV -user susan passwd
-long -from 3 -to 6
Item      3, 28-Jan-2002.10:11:37 TABLE_PACKLET      334 (epoch
BOSTON_HUB.3    initiated 28-Jan-2002.10:11:37) Schema: 14, Schema
Rev: 1, SchemaRev Ver:1
Item      4, 28-Jan-2002.10:11:37 TABLE_PACKLET      336 (epoch
BOSTON_HUB.4    initiated 28-Jan-2002.10:11:37) Schema: 14, Schema
Rev: 1, SchemaRev Ver:1
Item      5, 28-Jan-2002.10:11:37 TABLE_PACKLET      340 (epoch
BOSTON_HUB.5    initiated 28-Jan-2002.10:11:37) Schema: 14, Schema
Rev: 1, SchemaRev Ver:1
Item      6, 28-Jan-2002.10:11:37 TABLE_PACKLET      355 (epoch
BOSTON_HUB.6    initiated 28-Jan-2002.10:11:37) Schema: 14, Schema
Rev: 1, SchemaRev Ver:1
```

- **tokyo** レプリカの最初の **oplog** 項目を表示します。

```
multiutil dumpoplog -clan testclan -site tokyo -family TEST -user masako -p passwd -long
-from 1
Item      1, 29-Jan-2002.13:22:44 SPECIAL_PACKLET      88 (epoch
TOKYO.1    initiated 29-Jan-2002.13:22:44) Schema: 10, Schema Rev:
1, SchemaRev Ver:0
```

関連項目

lsepoch、scruboplog

Isepoch

エポック情報を表示します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
Isepoch [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
-u-ser username [ -p-assword ] password  
[ replica ...]
```

説明

このコマンドは、レプリカのエポック数マトリックスを表示します。マトリックス内のレプリカ自体のエポック行は、そのレプリカの実状態を表します。そのほかの行は、別のレプリカの状態についてのレプリカの見積もりを示します。

メモ: Isepoch の出力には、現在使用中のレプリカの行に加え、削除されたレプリカの行が含まれます。削除されたレプリカの **oplog** レコードは、復元中のレプリカが削除されたレプリカから操作を受け取る必要がある場合に備えて保存されます (たとえば、レプリカが削除される前に作成されたバックアップからレプリカを復元できます)。

制約事項

スーパー ユーザー権限が必要です。族またはサイトを指定しない場合は、ユーザー名とパスワードが族とサイトのすべてのローカル インスタンスに対して有効である必要があります。

オプションと引数

族、サイト、ファミリの指定

デフォルト

ローカルのすべての族とサイトのエポック見積もりを表示します。

-cl-an clan-name

エポック情報を表示する族です。

-site site-name

エポック情報を表示するサイトです。

-fam-ily family-name

エポック情報を表示するファミリです。

ユーザー データベース ファミリ : ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ : ファミリ名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

レプリカの指定

デフォルト

ファミリ内の各レプリカのエポック数を表示します。族もサイトも指定されていない場合は、族とサイトのすべてのローカル インスタンスのエポック見積もりを別々に表示します。

replica ...

エポック情報を表示するレプリカのサイトです。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- DEV ファミリ全体に対する **boston_hub** レプリカのエポック数を表示します。次の例では、ファミリ内のレプリカは **boston_hub** と **sanfran_hub** のみです。

```
multiutil lsepoch -clan telecom -site boston_hub -family DEV -user susan -p passwd
```

```
Multiutil: サイト 'BOSTON_HUB' で再現された各サイトのエポック見積もり (@minuteman):
```

```
BOSTON_HUB: 4
```

```
SANFRAN_HUB: 4
```

```
Multiutil: サイト 'SANFRAN_HUB' で再現された各サイトのエポック見積もり (@goldengate):
```

```
BOSTON_HUB: 4
```

```
SANFRAN_HUB: 4
```

- DEV ファミリの **sanfran_hub** レプリカの状態に対する **boston_hub** レプリカの見積もりを表示します。

```
multiutil lsepoch -clan telecom -site boston_hub -family DEV -user susan -p passwd  
sanfran_hub
```

```
Multiutil: サイト 'sanfran_hub' で再現された各サイトのエポック見積もり (@goldengate):
```

```
BOSTON_HUB: 5
```

```
SANFRAN_HUB: 3
```

- MASTR ファミリ全体に対する **tokyo** レプリカのエポック数を表示します。次の例では、**sydney** レプリカが **tokyo** レプリカからの更新を必要としていることがわかります。

```
multiutil lsepoch -clan testclan -site tokyo -family MASTR -user masako -p passwd
```

```
Multiutil: サイト 'TOKYO' で再現された各サイトのエポック見積もり (@shinjuku):  
TOKYO: 1
```

```
SYDNEY: 0
```

```
Multiutil: サイト 'SYDNEY' で再現された各サイトのエポック見積もり (@taronga):  
TOKYO: 0
```

```
SYDNEY: 0
```

関連項目

chepoch、recoverpacket、restorerereplica

lspacket

パケットの内容の説明を表示します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

lspacket [-l·ong | -s·hort] [pname ...]

説明

このコマンドは、レプリカ作成パケットまたは更新パケットを含む 1 つ以上のファイルの内容についての要約を表示します。デフォルトでは、**lspacket** の出力には次の情報が含まれます。

- 各パケットのパス名
- 各パケットのタイプ (レプリカの作成または更新)
- 生成の日付
- 元になるレプリカ
- パケットを適用する族とファミリー
- パケットの対象となるレプリカ
- パケットのシーケンス番号 (複数の物理パケットに分割された論理パケットの各部分を格納しているファイルを表します)

制約事項

なし

オプションと引数

表示形式

デフォルト

「説明」の項に記載した情報を表示します。

-l-long

デフォルト情報に加え、パケットを作成したレプリカの名前と、パケットの内容を示す oplog ID を表示します。

-s-short

パケットのパス名のみを表示します。

パケットの指定

デフォルト

現在のホストのすべての記憶ベイにあるすべてのパケットを表示します。

pname ...

ファイルかディレクトリ、またはその両方の 1 つ以上のパス名です。

指定した各ファイルのうち、物理パケットを含んでいるものが表示されます。指定した各ディレクトリについて、**lspacket** はこのディレクトリに格納されたパケットを表示します。

例

- 更新パケット lab.xml の内容を表示します。

```
multiutil lspacket -long c:¥cqms¥lab.xml
Multiutil: パケット 'c:¥cqms¥lab.xml' ...
Multiutil:      タイプ: 'CREATE_PACKET'
Multiutil:      Sent: 2002-01-22 10:58:11
Multiutil:      差出人: BOSTON_HUB
                (B6A316BE-CCB4-11D5-AFB5-00B0D0682333)
Multiutil:      所属: 'TELECOMM'
Multiutil:      Recipients: SANFRAN_HUB
Multiutil:      ファミリー: 'DEV'
```

- ローカルホストの記憶ベイにあるすべてのパケットを表示します。

multiutil lspacket

Multiutil: パケット

```
'C:\temp\cqms\ms_ship\incoming\mk_TOKYO_29-January-02_09-47-27.xml'...
```

Multiutil: タイプ: 'CREATE_PACKET'

Multiutil: Sent: 2002-01-29 09:47:28

Multiutil: 差出人: TOKYO (B6A316BE-CCB4-11D5-AFB5-00B0D0682333)

Multiutil: 所属: 'TESTING'

Multiutil: 受取人: SYDNEY

Multiutil: ファミリー: 'TEST'

Multiutil: パケット

```
'C:\temp\cqms\ms_ship\incoming\sync_SANFRAN_HUB_07-February-02_11-24-49.xml'...
```

Multiutil: タイプ: 'UPDATE_PACKET'

Multiutil: Sent: 2002-02-07 11:24:49

Multiutil: 差出人: SANFRAN_HUB
(8AB1A196-BE48-47F1-9255-71FD18D7309D)

Multiutil: 所属: 'TELECOMM'

Multiutil: 受取人: BOSTON_HUB

Multiutil: ファミリー: 'DEV'

- ローカルホストの記憶ベイにあるパケットを **-short** 形式で表示します。

multiutil lspacket -short

Multiutil: パケット

```
'C:\temp\cqms\ms_ship\incoming\mk_TOKYO_29-January-02_09-47-27.xml'...
```

Multiutil: パケット

```
'C:\temp\cqms\ms_ship\incoming\sync_SANFRAN_HUB_07-February-02_11-24-49.xml'...
```

関連項目

mkreplica、MultiSite コントロール パネル、sync replica、shipping.conf

Isreplica

データベースのレプリカを表示します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

- データベースのレプリカを表示します。

```
lsreplica [ -clan clan-name ] [ -site site-name ] -family family-name  
-user username [ -password ] password  
[ -long | -short | -fmt format ]  
[ -siblings  
| [ -siblings ] -family family-name  
| replica ... ]
```
- ファミリの作業スキーマ リポジトリを表示します。

```
lsreplica [ -clan clan-name ] [ -site site-name ] -family family-name  
-user username [ -password ] password -working-master
```

説明

このコマンドは、現在のレプリカによって認識されているすべてのアクティブなレプリカについての情報を表示します。族内または族のファミリ内のすべてのレプリカを表示できます。そのほかのレプリカが存在していても、その作成情報を含んだパッケージが現在のレプリカにまだインポートされていない場合もあります。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ: デフォルトはありません。ファミリを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ: ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ: ファミリ名は **MASTR** です。

-siblings オプションまたは **-siblings -infamily** オプションを使用すると、スキーマ リポジトリによって認識されている、族内の特定のファミリにあるレプリカを表示できます。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

表示形式

デフォルト

各レプリカの作成イベント情報が含まれます。

-l-ong

各レプリカの作成情報と同期サーバーを表示します。現在のレプリカが復元中の場合、このオプションは復元更新を必要とするほかのレプリカのリストに注釈を付けます (**restorereplica** リファレンス ページを参照してください。)

-s-hort

レプリカ名のみを表示します。

-fmt format-string

指定したフォーマット文字列を使用して情報を表示します。フォーマット文字列では、表示する項目を特定し、項目の表示形式を指定する変換指定を使用します。変換指定の形式は C 言語の **printf()** 関数の形式と同様で、パーセント記号 (%) と、表示するデータの種類を示すキーボード上の文字 (小文字) から成ります。

printf() 指定子とは異なり、変換指定はコマンド行で指定した引数では置き換えられません。変換指定は、**multiutil** によって、レプリカから抽出されたフィールド値に自動的に置き換えられます。

format-string は、英数字、変換指定、エスケープ シーケンスで構成される文字列です。二重引用符 (") で囲む必要があります。

変換指定 :

%h	ホスト名
%n	レプリカの名前
%c	族名
%f	ファミリー名
%d	レプリカの説明 (ある場合)
%s	レプリカの状態
%%	% 文字
%z	ID ブロック サイズ
%t	ID ブロックのしきい値

エスケープ シーケンス :

¥n	<NL>
¥t	<TAB>
¥'	一重引用符
¥¥	¥ マーク自体
¥nnn	8 進コードで指定される文字

レプリカの指定

デフォルト

現在のレプリカのファミリー内のすべての既知のレプリカ (現在のレプリカも含む) を表示します。

-working-master

指定した族の作業スキーマ リポジトリを表示します。

-sib-lings

ユーザー データベースの場合は、現在のレプリカのファミリー メンバを表示します。現在のレプリカ自体は表示しません。スキーマ リポジトリの場合は、現在のサイトのレプリカを除く、サイトのすべてのレプリカのファミリー メンバを表示します。このオプションは、シブリング レプリカだけで動作するスクリプトを記述するときに役立ちます。

-infa-mily family

指定したレプリカのファミリー内のレプリカを表示します。レプリカはサイト名を使用して指定してください。このオプションは、**-family** オプションで **MASTR** を指定した場合のみ使用できます。

replica ...

-site オプションは、情報をクエリーしているレプリカを指定します。この引数は、情報を表示する対象のレプリカのサイトを指定します。表示できるのは、同じファミリーのメンバであるレプリカのみです。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **telecomm** 族の **DEV** ファミリー内にあるレプリカを表示します。

```
multiutil lsreplica -clan telecomm -site boston_hub -family DEV -user susan -p passwd -long
```

```
名前: BOSTON_HUB; 所属: TELECOMM; ファミリー: DEV; ホスト: minuteman;  
ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロック  
しきい値: 1024
```

```
名前: SANFRAN_HUB; 所属: TELECOMM; ファミリー: DEV; ホスト: goldengate;  
ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロック  
しきい値: 1024
```

```
名前: BANGALORE; 所属: TELECOMM; ファミリー: DEV; ホスト: ramohalli;  
ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロック  
しきい値: 1024
```

- telecomm 族の DEV ファミリ内のシブリングを表示します。boston_hub サイトのユーザー データベースのレプリカは表示しません。

```
multiutil lsreplica -clan telecomm -site boston_hub -family DEV -user susan -p passwd -long -siblings
```

名前: SANFRAN_HUB; 所属: TELECOMM; ファミリー: DEV; ホスト: goldengate; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

名前: BANGALORE; 所属: TELECOMM; ファミリー: DEV; ホスト: ramohalli; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

- telecomm 族の族メンバを表示します。boston_hub サイトのレプリカは表示しません。

```
multiutil lsreplica -clan telecomm -site boston_hub -family MASTR -user susan -p passwd -long -siblings
```

名前: SANFRAN_HUB; 所属: TELECOMM; ファミリー: MASTR; ホスト: goldengate; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

名前: SANFRAN_HUB; 所属: TELECOMM; ファミリー: DEV; ホスト: goldengate; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

名前: BANGALORE; 所属: TELECOMM; ファミリー: MASTR; ホスト: ramohalli; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

名前: BANGALORE; 所属: TELECOMM; ファミリー: DOC; ホスト: ramohalli; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

- 作業スキーマ リポジトリによって認識されている、testclan 族の TEST ファミリ内のユーザー データベースを表示します。

```
multiutil lsreplica -clan testclan -site sydney -family MASTR -user bfife -p passwd -long -infamily TEST
```

名前: TOKYO; 所属: TESTING; ファミリー: TEST; ホスト: shinjuku; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

名前: SYDNEY; 所属: TESTING; ファミリー: TEST; ホスト: taronga; ステータス: NORMAL, NOT CONNECTED; 説明: ; ブロック サイズ: 4096; ブロックしきい値: 1024

- testclan 族の DEV ファミリの作業スキーマ リポジトリを表示します。

```
multiutil lsreplica -clan testclan -site sydney -family DEV -user bfife -workingmaster
```

- `lsreplica -long` の出力と似たものを出力します。リテラルの二重引用符を含むフォーマット文字列を使用する場合は、一重引用符で囲みます。

```
multiutil lsreplica -clan testing -site tokyo -family TEST -user masako -password passwd  
-fmt "Name:%n; Clan:%c; Family:%f; Host:%h; Status:%s;"
```

```
名前 :TOKYO; 所属 :TESTING; ファミリー :TEST; ホスト :shinjuku; ステータス  
:NORMAL, NOT CONNECTED;
```

```
名前 :SYDNEY; 所属 :TESTING; ファミリー :TEST; ホスト :taronga; ステータス  
:NORMAL, NOT CONNECTED;
```

関連項目

`mkreplica`

mkorder

保存および転送機能が使用する発送オーダーを作成します。

適用

製品	コマンドタイプ
MultiSite	MultiSite コマンド

プラットフォーム
UNIX
Windows

構文

```
mkorder -dat a packet-pname [ -scl-ass storage-class-name ]  
        [ -pex-pire date-time ] [ -not-ify e-mail-address ]  
        [ -c comment | -cq | -cqe | -nc ]  
        [ -shi-p -cop-y | -fsh-ip [ -cop-y ] | -out order-pname ] destination ...
```

説明

このコマンドは、既存のパケット ファイルまたはその他のファイルに対する発送オーダー ファイルを作成します。発送オーダーは、Shipping Server がファイルを 1 つ以上の宛先に送信するために使用します。

mkorder は、mkreplica -out または syncreplica -out を使用して作成したパケットを SHIPPING サーバーに送信します。また、mkorder を使用して、発送オーダーの有効期限が切れたパケットを再送信したり、複数のサイト間で任意のファイルを送信することもできます。発送オーダーは、関連付けられたパケットまたはファイルと同じディレクトリにある必要があります。

メモ: SHIPPING サーバーは、正常にデリバーできたパケットを削除します (宛先がローカルホストの場合を除く)。ほかのサイトにデリバーされた後も自分のサイトに保存する必要があるファイルをこのコマンドを使用して処理する場合は、-copy オプションを指定してください。

制約事項

なし

オプションと引数

パケット ファイルの指定

デフォルト

なし

-dat-a packet-pname

パケットまたはファイルのパス名。

メモ : packet-pname にコロン (:) が含まれている場合、mkorder のプロセスではピリオド (.) に変換されます。Windows マシンではファイル名にコロンが使用できませんが、変換することによりパケットを Windows マシンにデリバリーできるようになります。

発送オーダーの配置場所の指定

デフォルト

packet-pname ファイルが保存されているディレクトリに発送オーダーを作成します。

-scl-ass class-name

パケットと発送オーダーの記憶クラスを指定します。**-ship** または **-fship** も使用すると、mkorder は、UNIX では **shipping.conf** ファイルの記憶クラスを、Windows では MultiSite のコントロール パネルの記憶クラスをそれぞれ検索し、使用する記憶ベイの場所を決定します。

このオプションを省略して **-ship** または **-fship** を使用すると、mkorder は MultiSite のコントロール パネルまたは **shipping.conf** ファイルで **-default** クラスに指定された記憶ベイの場所に発送オーダーを配置します。

-shi-p -cop-y

-fsh-ip [-cop-y]

packet-pname の発送オーダーを作成します。**-fship** を使用すると、**shipping_server** を呼び出してパケットを送信します。**-ship** を使用すると、発送オーダーを記憶ベイに配置します。パケットを送信するには、**shipping_server** を実行します。**-copy** は **-ship** と共に使用します。また、オプションで **-fship** とも使用します。

- **-copy** を指定した場合、mkorder は保存および転送機能の記憶ベイのいずれか 1 つに packet-pname ファイルをコピーし、発送オーダーをそのベイに配置します。コピーは、発送オーダーに指定されているすべての宛先にデリバリーされた後、削除されます。

- `-copy` を指定しない場合、`mkorder` は `packet-pname` ファイルをコピーしません。`mkorder` は、そのファイルがあるディレクトリに発送オーダーを配置します。`packet-pname` は、発送オーダーに指定されているすべての宛先にデリバーされた後、削除されます。

`-out order-pname`

発送オーダーを、記憶ベイではなく指定されたファイルに配置します。ファイルが既に存在する場合はエラーになります。

パケットをデリバーできなかった場合の処理

デフォルト

デリバーできなかったパケットは、保存および転送機能を介して、元になるレプリカのサイトの管理者に戻されます。メール メッセージが保存および転送の管理者に送られます。これは、繰り返し試みたパケットのデリバーが失敗し、割り当てられた時間を使い果たした場合に発生します。また、宛先のホストが不明な場合や、データファイルが存在しない場合にも発生します。有効期限、管理者の電子メール アドレス、通知プログラムは、保存および転送の構成設定により指定されます。

`-pex-pire date-time`

保存および転送機能がパケットのデリバーを中止し、障害を通知するメールを生成する時刻を指定します。このオプションは、`shipping.conf` ファイル (UNIX) または MultiSite のコントロールパネル (Windows) の記憶クラスに指定した有効期限を上書きします。

`date-time` 引数は、次のいずれかの形式で指定できます。

`date.time | date | time | now`

各パラメータの内容は次のとおりです。

`date` := `day-of-week | long-date`

`time` := `h[h]:m[m]:s[s] [UTC [[+ | -]h[h]:m[m]]]]`

`day-of-week` := `today |yesterday |Sunday | ... |Saturday |Sun | ... |Sat`

`long-date` := `d[d]-month[-[yy]yy]`

`month` := `January |... |December |Jan |... |Dec`

`time` は、地域のタイム ゾーンに合わせた 24 時間形式で時刻を指定します。`time` を省略した場合、デフォルト値は `00:00:00` です。`date` を省略した場合、デフォルト値は `today` です。世紀、年、または特定の日付を省略すると、最新のものが使用されます。タイムゾーンにかかわらず同じ時刻を適用するには、`UTC` を指定します。`UTC` 時間に対して正のオフセットを指定するにはプラス (+) 演算子を、負のオフセットを指定するにはマイナス (-) 演算子を使用します。時または分のオフセットを付けずに `UTC` を指定する

と、デフォルトの設定はグリニッジ標準時 (GMT) になります (協定世界時 (UTC) 1970 年 1 月 1 日より前の日付は無効です)。

例を次に示します。

```
22-November-2002
sunday
yesterday.16:00
8-jun
13:00
today
9-Aug.10:00UTC
```

-not-ify e-mail-address

指定された電子メール アドレスに、デリバリー失敗を知らせるメッセージを送信します。

電子メール通知が使用できない Windows ホスト上で障害が発生した場合は、Windows イベント ビューアにメッセージが表示されます。メッセージには、このオプションで指定された **e-mail-address** 値と、このユーザーに操作の状態を通知するよう求めるメモが含まれています。電子メールでの通知を有効にする方法については、**MultiSite コントロール パネル**のリファレンス ページを参照してください。

イベント レコードとコメント

デフォルト

-nc (コメントなし)

-c comment | -cq | -cqe | -nc

送信オーダーに記載するコメントを指定します。**-c** を指定する場合、コメント文字列は 1 行のコマンド行トークンである必要があります。通常は二重引用符で囲みます。**-cq** と **-cqe** を指定した場合は、コマンドによりコメントが要求されます。**-nc** を指定した場合は、送信オーダーにコメントは記載されません。

宛先の指定

デフォルト

なし

destination ...

1 つ以上のホスト名 (別のドメインのホストも使用できる必要があります) または IP アドレスを指定します **MultiSite** パケットを送信するときは、その複製に対する同期サーバーを指定する必要があります。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- デフォルトの記憶ベイに配置されているファイル **p1** について発送オーダーを作成します。発送オーダーを **p1** と同じ記憶ベイに格納し、ファイルがホスト **goldengate** に送信されるように指定します。

```
mkorder -data "c:¥Program
Files¥Rational¥ClearCase¥var¥shipping¥cq_default¥outgoing¥p1" -sclass cq_default
-out "c:¥Program
Files¥Rational¥ClearCase¥var¥shipping¥cq_default¥outgoing¥p1_order" goldengate
発送オーダー "c:¥Program Files¥Rational¥ClearCase¥var
¥shipping¥cq_default¥outgoing¥p1_order" が生成されました。
```

- ホスト **goldengate** にデリバーされるよう指定されたファイルについて、発送オーダーをデフォルトの記憶ベイに作成します。ファイルが正しくデリバーされない場合、**admin** に通知されるように指定します。

```
/opt/rational/clearcase/etc/mkorder -data /usr/tmp/to_goldengate -sclass cq_default
-ship -copy -notify admin goldengate
発送オーダー
"/var/adm/rational/clearcase/shipping/cq_default/outgoing/sh_o_to
_goldengate" が生成されました。
```

- 同じファイルについて発送オーダーを作成し、ここでは特定の記憶クラスに対する記憶ベイに配置します。直ちにデリバーを開始し (**-fship**)、5 月 18 日まで引き続きデリバーを試行できるようにします。

```
mkorder -data c:¥tmp¥to_goldengate -fship -copy -sclass ClassA -pexpire 18-May
goldengate
発送オーダー "c:¥tmp¥sclass¥ClassA¥sh_o_to_goldengate" が生成されました。
生成されたパケットを転送 / デリバーしようとしています ...
-- パケット c:¥tmp¥sclass¥ClassA¥sh_o_to_goldengate を転送 / デリバーしました
```

ファイル

ccase-home-dir/config/services/shipping.conf

関連項目

mkreplca、MultiSite コントロール パネル、shipping.conf、shipping_server、syncreplca
第 10 章 MultiSite の操作のトラブルシューティング

mkreplica

レプリカを作成します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

- 既存のデータベースを複製して、新しいレプリカ オブジェクトとレプリカ作成パッケージを生成します。

```
mkreplica -exp-ort [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name
    -u-ser username [ -p-assword ] password [ -max-size size ] [ -c-omments comments ]
    [ -size id-block-size ] [ -thres-hold id-block-threshold ]
    {
        { -sh-ip | -fsh-ip } -wor-kdir temp-dir-pname    [ -sc-lass storage-class ]
        [ -pex-pire date-time ] [ -not-ify e-mail-addr ]
    | -out packet-file-pname
    } hostname:site-name ...
```

- レプリカ作成パッケージをインポートして、新しいユーザー データベースとスキーマ リポジトリのレプリカを作成します。

```
mkreplica -imp-ort
    { -site site-name -repo-sitory db-info [ -vendor vendor-type ] db-params
    }
    [ [ -data-base db-info [ -vendor vendor-type ] db-params
    [ -c-omments comments ] { packet-file-pname | packet-dir-path }...
```

- レプリカ作成パッケージをインポートして、現在のサイトの既存のスキーマ リポジトリと同じ族に新しいレプリカを作成します。

```
mkrepl-ica -imp-ort { [ -cl-an clan-name ] [ -site site-name ]
-u-ser username [ -p-assword ] password
{ -data-base db-info [ -vendor vendor-type ] db-params
[ -c-omments comments ] { packet-file-pname| packet-dir-path }...
```

説明

メモ: 族の最初のデータベースのレプリカを作成する前に、まず、データベースが属するデータベース セットをアクティブにする必要があります。また、レプリカを作成するデータベースをアップグレードして、スキーマの最新のバージョンを使用する必要があります。

mkreplica -export コマンドの実行には時間がかかる場合があります。データベースとスキーマ リポジトリは、エクスポートの進行中はロックされます。**mkreplica -export** を実行する前に、すべてのユーザーがログアウトしていることを確認してください。詳細については、第 6 章「データベース レプリカの作成」を参照してください。

新しいレプリカの作成プロセスには 3 つのフェーズがあります。

- 1 **mkreplica -export** コマンドは、指定されたユーザー データベースと、それに関連付けられたスキーマ リポジトリの内容を複製します。これは、1 つ以上のサイトに転送する 1 つの論理レプリカ作成パッケージを生成します。論理パッケージは、複数の物理パッケージに分割できます (**-fship** または **-ship** を使用すると、**mkreplica** はそれぞれの物理パッケージに対して発送オーダー ファイルも作成します)。

メモ: 1 つの **mkreplica -export** コマンドで複数のレプリカを作成したほうが、複数の **mkreplica -export** コマンドを使用するよりも効率的です。

- 2 生成されたパッケージを 1 つ以上のほかのサイトに送信します。
- 3 各受信サイトでは、**mkreplica -import** コマンドでレプリカ作成パッケージを使用して新しいレプリカを作成します。新しいレプリカは、複製された 2 つのデータベース (スキーマ リポジトリとユーザー データベース) で構成されます。ユーザー データベースのレプリカを既存のスキーマ リポジトリと同じ族のファミリーに追加する場合は、このコマンドの使い方が異なります。

空の製造元データベースの作成

管理者は、新しい各サイトにレプリカ データ用の空の製造元データベースを作成する必要があります。新しいサイトに初めてレプリカを作成する場合は、空の製造元データベースが少なくとも 2 つ必要です。1 つはスキーマ リポジトリのレプリカ用、もう 1 つはユーザー データベースのレプリカ用です。

メモ: 新しいユーザー データベース レプリカを既存のサイトに追加する場合は、スキーマ リポジトリ用の製造元データベースを作成する必要はありません。新しいユーザー データベース レプリカをサイトの既存のスキーマ リポジトリに関連付けることができます。

oplog の情報

データベースのレプリカを初めて作成したときに、データベースの操作ログ (oplog) が有効になります。レプリカを作成するすべての操作は oplog に記録されます。操作のログは、元のデータベース セット以外すべてのレプリカが削除されるまで、継続して記録されます。レプリカを追加作成すると、oplog 項目に記録されます。既存のレプリカは、標準的な同期メカニズムを使用して新しいレプリカの内容を取得します (sync replica リファレンス ページを参照してください)。

メモ: mkreplica -export コマンドを入力する前に、元のサイトに MultiSite ライセンスがインストールされていることを確認してください。元のデータベース セットをアクティブにすると、ClearQuest ライセンスに加えて MultiSite ライセンスを持っている開発者しかデータベース セットにアクセスできなくなります。mkreplica -export を実行するには、MultiSite のライセンスも必要です。

レプリカへの ID ブロックの割り当て

MultiSite では、各レプリカに割り当てられるレコード ID 番号の数を管理します。この割り当て処理は、ID ブロック (ID のグループ) を使用して実行されます。

デフォルトでは、各レプリカに対して、作成時に 4096 個の ID から成る ID ブロックが割り当てられます。レプリカで利用できる残り ID 数がしきい値の 1024 に達すると、すべての ID が一意になるように、4096 個の ID から成る別の ID ブロックが割り当てられます。ID ブロックの割り当ては、同期中に作業スキーマ リポジトリによって内部処理されます。

レプリカ ファミリのアクティビティ レベルによっては、レプリカに割り当てられる ID ブロックのサイズを大きくしたほうがよい場合があります。たとえばデフォルトの設定では、多数の障害を登録しようとした場合、最初の 4096 個が正常に登録され、それ以降の登録は失敗します。

レプリカに割り当てられる ID 数を制御するには、mkreplica -export コマンドでレプリカを作成するときに、-size オプションと -threshold オプションを組み合わせで使用してください。これらの設定は chreplica コマンドで変更できます。

レプリカ作成パケット

mkreplica -export を呼び出すたびに、1 つの論理レプリカ作成パケットが作成されます (1 つの mkreplica コマンドで複数の新規レプリカを作成した場合も同じです)。各パケットにはレプリカの指定が 1 つ以上含まれ、それぞれに新しいレプリカの名前とそのレプリカに関連付けられた同期サーバーが示されます。

ユーザー データベースとスキーマ リポジトリは、エクスポート フェーズ中はロックされます。

-maxsize オプションは、転送メディアの制限に適合するように、1 つの論理パケットを複数の物理パケットに分割します。

失敗したインポートの回復

レプリカのインポートが何らかの原因 (停電など) で中断したり失敗したりした場合は、製造元データベースを削除し、失敗したインポート操作用に新しい製造元データベースを作成し、`mkreplica -import` を再度実行する必要があります。

スキーマ リポジトリのインポートには成功したが、ユーザー データベースのレプリカのインポートには失敗したという場合もあります。この場合は、ユーザー データベースのレプリカ用に準備した製造元データベースを、一度削除してから再作成する必要があります。詳細については、64 ページの「インポートが失敗した場合の回復」を参照してください。

使用済みパケットのクリーンアップ

レプリカ作成パケットは、インポート後も削除されません。レプリカ作成パケットを `mkreplica -import` でインポートした後、そのパケットを自分で削除する必要があります。

パケットをデリバリーできなかった場合のエラー処理

デリバリーできなかったパケットは、保存および転送機能を使用して、元になるレプリカのサイトの管理者に戻されます。メール メッセージが保存および転送の管理者に送られます。これは、繰り返し試みたパケットのデリバリーが失敗し、割り当てられた時間を使い果たした場合に発生します。また、宛先のホストが不明な場合や、データ ファイルが存在しない場合にも発生します。有効期限、管理者の電子メール アドレス、通知プログラムは、保存および転送の構成設定により指定されます。

制約事項

ロック: データベースがロックされている場合 (アップグレード プロセス中など)、または別の ClearQuest MultiSite 操作が実行中の場合、このコマンドは失敗します。

その他の制約: 異なるバージョンの MultiSite を実行しているホストにデータベースのレプリカを作成することはできません。`mkreplica -export` はどのサイトでも実行できますが、作業スキーマ リポジトリのサイトで実行することをお勧めします。これは、同じ名前の複数のサイトが作成されるのを防ぐためです。

オプションと引数: エクスポート フェーズ

族、サイト、ファミリの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ: デフォルトはありません。ファミリを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ: ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ: 指定する必要はありません。**mkreplica** を実行すると、指定したユーザー データベース ファミリの関連付けられたスキーマ リポジトリが、レプリカ作成パッケージに含まれます。

デフォルト: なし

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

レプリカ作成パッケージのサイズの指定

デフォルト

-maxsize を指定していない場合、使用する発送方法によってデフォルトのパッケージサイズが異なります。

- **-ship** または **-fship** を指定して作成したパッケージは、MultiSite のコントロールパネルで指定した最大パッケージサイズを超えることはありません。
- **-out** を指定して作成したパッケージが 2 GB を超えることはありません。

システムでサポートされるサイズよりも大きいパケットを作成しようとすると、**mkreplica** コマンドでエラーが発生します。

-max-size size

物理パケットの最大サイズで、数字に 1 文字を付けて表します。次のようになります。

500k 500 キロバイト

20m 20 メガバイト

1.5g 1.5 ギガバイト

コメントの指定

デフォルト

なし

-c omments comments

このレプリカの情報と共に保存するコメントです。

ID ブロック割り当ての指定

デフォルト

ID ブロック サイズは 4096、ID ブロックしきい値は 25 パーセントです。

-size id-block-size

ID ブロックのサイズです。1 ～ 1023 までの任意の数値を入力できます。

id-block-size の値を 100 倍した値が、実際の ID ブロックのサイズです。たとえば、サイズが 30,000 の ID ブロックを指定するには「300」と入力し、サイズが 25,000 の ID ブロックを指定するには「250」と入力します。

-thres-hold id-block-threshold

レプリカに割り当ててるレコード ID 番号の数です。**id-block-threshold** には整数のパーセント値を指定します。1 ～ 63 までの任意の数値を入力できます。使用可能なレコード ID 数の残りが、現在の ID ブロック サイズに指定されているパーセント値を下回ると、追加のブロックが割り当てられます。

レプリカ作成パケットの配置

デフォルト

なし **mkreplica -export** で作成したレプリカ作成パケットの保存方法か、ほかのサイトへの転送方法、またはその両方を指定する必要があります。

-shi-p

-fsh-ip

保存および転送記憶ベいの 1 つ以上のファイルにレプリカ作成パケットを保存します。個々の発送オーダー ファイルは、個々の物理パケットに添付され、デリバリー方法とデリバリー先を示します。

-fship (強制発送) は、shipping_server を呼び出してレプリカ作成パケットを送信します。-ship は記憶ベイにパケットを配置します。パケットを送信するには、shipping_server を呼び出します。

送信ホストまたは受信ホスト上の記憶ベイが配置されているディスク パーティションには、レプリカ作成パケットのサイズ以上の空き容量が必要です。

--workdir temp-dir-name

mkreplica が一時的なワークスペースとして使用するディレクトリです。mkreplica 終了時に削除されます。既存のディレクトリを指定することはできません。

-storage-class storage-class

パケットと発送オーダーの記憶クラスを指定します。mkreplica は、MultiSite のコントロール パネル (Windows) または shipping.conf ファイル (UNIX) にある記憶クラスを検索し、使用する記憶ベイの場所を特定します。

デフォルト : mkreplica は、cq_default クラスに指定した場所の記憶ベイにパケットを配置します。

-out packet-file-pname

最初の物理レプリカ作成パケットの名前です。ほかのパケットは、packet-file-pname_2、packet-file-pname_3 のように順に名前を付けたファイルに配置されます。

レプリカ作成パケットは自動的にデリバーされるわけではありません。適切な方法を使用してデリバーしてください。-out でパケットを作成し、その後、保存および転送機能を使用してデリバーできます。mkorder リファレンス ページを参照してください。

パケットをデリバーできなかった場合の処理

デフォルト

デリバーできなかったパケットは、保存および転送機能を使用して、元になるレプリカのサイトの管理者に戻されます。メール メッセージが保存および転送の管理者に送られます。これは、繰り返し試みたパケットの配信が失敗し、割り当てられた時間を使い果たした場合に発生します。また、宛先のホストが不明な場合や、データ ファイルが存在しない場合にも発生します。有効期限、管理者の電子メール アドレス、通知プログラムは、保存および転送の構成設定により指定されます。

-expire date-time

保存および転送機能がパケットのデリバーを中止し、障害を通知するメールを生成する時刻を指定します。このオプションは、shipping.conf ファイル (UNIX) または MultiSite のコントロール パネル (Windows) の記憶クラスに指定した有効期限を上書きします。

date-time 引数は、次のいずれかの形式で指定できます。

date.time | date | time | now

各パラメータの内容は次のとおりです。

date := day-of-week | long-date

time := h[h]:m[m]:s[s] [UTC [[+ | -]h[h]:m[m]]]]

day-of-week := today |yesterday |Sunday | ... |Saturday |Sun | ... |Sat

long-date := d[d]-month[-[yy]yy]

month := January |... |December |Jan |... |Dec

time は、地域のタイムゾーンに合わせた 24 時間形式で時刻を指定します。**time** を省略した場合、デフォルト値は **00:00:00** です。**date** を省略した場合、デフォルト値は **today** です。世紀、年、または特定の日付を省略すると、最新のものが使用されます。タイムゾーンにかかわらず同じ時刻を適用するには、**UTC** を指定します。**UTC** 時間に対して正のオフセットを指定するにはプラス (+) 演算子を、負のオフセットを指定するにはマイナス (-) 演算子を使用します。時または分のオフセットを付けずに **UTC** を指定すると、デフォルトの設定はグリニッジ標準時 (**GMT**) になります (協定世界時 (**UTC**) 1970 年 1 月 1 日より前の日付は無効です)。

例を次に示します。

```
22-November-2002
sunday
yesterday.16:00
8-jun
13:00
today
9-Aug.10:00UTC
```

-not-ify e-mail-address

指定された電子メール アドレスに、デリバリーの失敗を知らせるメッセージを送信します。

電子メール通知が使用できない Windows ホスト上で障害が発生した場合は、Windows イベント ビューアにメッセージが表示されます。メッセージには、このオプションで指定された **e-mail-address** 値と、このユーザーに操作の状態を通知するよう求めるメモが含まれています。電子メールでの通知を有効にする方法については、**MultiSite コントロール パネル**のリファレンス ページを参照してください。

レプリカ仕様

デフォルト

なし

hostname:site-name...

1 つ以上の引数を指定します。いずれの引数もこのパッケージから別のサイトで作成される新しいレプリカを示します。

hostname

新しいレプリカの同期サーバーです。hostname はほかのドメインにあるホストからも使用できる必要があります。保存および転送メカニズムはこれを使用して、更新パッケージをレプリカに送信する方法を決定します。ただし、サイトで保存および転送を使用していない場合でも、この情報は正確な状態にしておく必要があります (chreplica リファレンス ページを参照してください)。

hostname は、ホストの IP アドレスでも、minuteman のようなコンピュータ名でもかまいません。また、minuteman.purpledoc.com のように、IP ドメイン名の追加が必要なこともあります。

UNIX では、uname -n コマンドを使用してコンピュータ名を表示します。Windows NT では、コンピュータ名は [ネットワーク] ダイアログ ボックスに表示されます。このダイアログ ボックスは、コントロール パネルの [ネットワーク] アイコンをクリックして開きます。Windows 2000 では、コンピュータ名は [システムのプロパティ] ダイアログ ボックスの [ネットワーク ID] タブに表示されます。このダイアログ ボックスは、コントロール パネルの [システム] アイコンをクリックして開きます。

site-name

multiutil コマンドでレプリカを識別するために使用する名前です。サイト名は、最大 50 文字までの識別子でなければなりません。この名前は、族内で一意である必要があります。同じ族に属する 2 つのサイトを同じ名前にすることはできません。

オプションと引数: スキーマ リポジトリとユーザー データベースをインポートする場合のインポート フェーズ

サイトとデータベース情報の指定

デフォルト

なし

-site site-name

レプリカをインポートするサイトの名前です。サイト名は、レプリカがエクスポートされたときに指定されています。サイト名が不明の場合は、エクスポート側の管理者に問い合わせてください。

-repository db-info

使用する製造元データベースのデータベース情報です。

製造元データベース	dbinfo の値
DB2	データベース エイリアス
Oracle	SQL*Net エイリアス
SQL Server	物理データベース名

-vendor vendor-type

使用するデータベース製造元です。サポートされている製造元の種類は、DB2、ORACLE、SQL_SERVER です。

db-params

必須のデータベース パラメータは、ClearQuest データベースへの接続に必要なパラメータと同じです。レプリカのインポート先の製造元データベースを作成するときは、このことに注意してください。空の製造元データベースの作成方法と必要なパラメータの詳細については、Rational ClearQuest の **インストール ガイド**を参照してください。

レプリカをインポートするときは、スキーマ リポジトリ レプリカ用の製造元データベースとユーザー データベース レプリカ用の製造元データベースのデータベース パラメータを指定する必要があります。これらのデータベースは、レプリカ パケットをインポートする前に作成する必要があります。

製造元データベース	db-params の値
DB2	-dbologin dbo-name [dbo-pwd]
Oracle	-dbologin dbo-name dbo-pwd [-connectopts connect-options]
SQL Server	-server server-name -dbologin dbo-name [dbo-pwd] -rwlogin rw-login [rw-pwd] [-rologin ro-login [ro-pwd]]

メモ : -rologin は、スキーマ リポジトリのレプリカを作成する場合にのみ使用します。

-data-base db-info

使用する製造元データベースのユーザー データベース情報です。

製造元データベース	dbinfo の値
DB2	データベース エイリアス
Oracle	SQL*Net エイリアス
SQL Server	物理データベース名

-c-omments comments

レプリカの情報と共に保存するコメントです。

レプリカ作成パケットの場所指定

デフォルト

なし

packet-file-pname | packet-dir-path ...

レプリカ作成パケットのパス名を指定します。複数のディスク ファイルにまたがる論理パケットについて、mkreplica は関連する物理パケットの packet-file-pname を含むディレクトリをスキャンします。

packet-dir-path 引数を 1 つまたは複数指定した場合も、mkreplica はこれらのディレクトリで追加パケットを検索します。

オプションと引数：ユーザー データベースのみをインポートする場合のインポート フェーズ

ユーザー データベース ファミリを既存の族に追加する場合、作成する必要があるのは、ユーザー データベースのレプリカ用の製造元データベースのみです。

族とサイトの指定

デフォルト

族：このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、-clan を指定する必要があります。

サイト：現在のサイトです。このホストに複数のサイトがある場合は、-site を指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

データベース情報の指定

-data-base db-info

使用する製造元データベースのユーザー データベース情報です。

製造元データベース	dbinfo の値
DB2	データベース エイリアス
Oracle	SQL*Net エイリアス
SQL Server	物理データベース名

-vendor vendor-type

使用するデータベース製造元を入力します。サポートされている製造元の種類は、DB2、ORACLE、SQL_SERVER です。

db-params

必須のデータベース パラメータは、ClearQuest データベースへの接続に必要なパラメータと同じです。レプリカのインポート先の製造元データベースを作成するときは、このことに注意してください。空の製造元データベースの作成方法と必要なパラメータの詳細については、Rational ClearQuest のインストレーション ガイドを参照してください。

レプリカをインポートするときは、スキーマ リポジトリ レプリカ用の製造元データベースとユーザー データベース レプリカ用の製造元データベースのデータベース パラメータを指定する必要があります。これらのデータベースは、レプリカ パケットをインポートする前に作成する必要があります。

製造元データベース	db-params の値
DB2	-dbologin dbo-name [dbo-pwd]
Oracle	-dbologin dbo-name [dbo-pwd] [-connectopts connect-options]

製造元データベース	db-params の値
SQL Server	-server server-name -dbologin dbo-name dbo-pwd -rwlogin rw-login [rw-pwd] [-rologin ro-login [ro-pwd]]

メモ : -rologin は、スキーマ リポジトリのレプリカを作成する場合にのみ使用します。

-c-omments comments

このレプリカの情報と共に保存するコメントです。これらのコメントは、インポート側のサイトのスキーマ リポジトリ データベースに保存され、ClearQuest Designer の [データベースのプロパティ] ダイアログボックスに表示されます。

レプリカ作成パケットの場所指定

packet-file-pname | packet-dir-path ..

レプリカ作成パケットのパス名を指定します。複数のディスク ファイルにまたがる論理パケットについて、mkreplica は関連する物理パケットの packet-file-pname を含むディレクトリをスキャンします。

packet-dir-path 引数を 1 つまたは複数指定した場合も、mkreplica はこれらのディレクトリで追加パケットを検索します。

デフォルト : なし

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

エクスポート

- boston_hub レプリカで、DEV ファミリのレプリカ作成パケットを生成し、sanfran_hub という名前の新しいレプリカを作成します。新しいレプリカの同期サーバーは goldengate です。

```
multiutil mkreplica -export -clan telecom -site boston_hub -family DEV -u susan -p
passwd -out c:\cqms\lab\boston_hub.xml goldengate:sanfran_hub
Multiutil: パケット ファイル 'c:\cqms\lab\boston_hub.xml' が生成されました
```

- boston_hub レプリカで、sanfran_hub レプリカでインポートしたときに LAB ファミリデータベースのレプリカを作成するパケットを生成します。

```
multiutil mkreplica -export -clan telecom -site boston_hub -family LAB -user susan -p
passwd -out c:\cqms\lab\lab.xml goldengate:sanfran_hub
Multiutil: パケット ファイル 'c:\cqms\lab\lab.xml' が生成されました
```

- **tokyo** レプリカで、**sydney** レプリカのレプリカ作成パッケージを生成し、**-fship** を使用して直ちにパッケージを転送します。

```
multiutil mkreplica -export -clan testing -site tokyo -family TEST -user masako -p passwd
-fship -workdir c:\cqlms\working -sclass cq_default taronga:sydney
```

Multiutil: パッケージ ファイル

'c:\cqlms\working\mk_TOKYO_29-January-02_09-47-27.xml' が生成されました

multiutil: 送信オーダー

"C:\temp\cqlms\ms_ship\outgoing\sh_o_mk_TOKYO_29-January-02_09-47-27.xml" が生成されました。

multiutil: 生成されたパッケージを転送 / デリバーしようとしています ...

multiutil: -- パッケージ

C:\temp\cqlms\ms_ship\outgoing\mk_TOKYO_29-January-02_09-4 を転送 /
デリバーしました

- 前の例と同様ですが、後で保存および転送機能を使用して送信するために、パッケージファイルを記憶バイに配置します。

```
multiutil mkreplica -export -clan telecomm -site boston_hub -family DEV -user susan
-password passwd -c "make a new replica for sanfran_hub" -ship -workdir
c:\temp\working -sclass cq_default -pexpire 22-November-2003
goldengate:sanfran_hub
```

インポート

- 新しいデータベース レプリカ **sanfran_hub** と関連付けられたスキーマ リポジトリ レプリカを、SQL Server データベースにインポートします。

```
multiutil mkreplica -import -site sanfran_hub -repository sanfran_schemarepo -vendor
SQL_SERVER -server sb_server -dblogin jcole passwd -rwlogin jcole passwd -rologin
jcole passwd -database sanfran_userdb -vendor SQL_SERVER -dblogin jcole passwd
-rwlogin jcole passwd
```

- **testing** 族の **sydney** サイトにある新しいユーザー データベース レプリカをインポートします。新しいユーザー データベース レプリカは SQL Server データベースにインポートされます。

```
multiutil mkreplica -import -clan testing -site sydney -user bfife -p passwd
-database syd_userdb -vendor SQL_SERVER -dblogin bfife passwd -rwlogin bfife passwd
```

関連項目

activate

MultiSite コントロール パネル

保存および転送機能を設定します。

適用

製品	コマンドタイプ
MultiSite	管理ツール

プラットフォーム
Windows

構文

%SystemRoot%\System32\ms.cpl

MultiSite コントロール パネルを開くには、コントロール パネルで [MultiSite] アイコンをダブルクリックします。

説明

MultiSite コントロール パネルでは、各ホストの保存および転送の操作を制御します。制御対象となるのは、次に説明する構成パラメータの設定です。パラメータが定義されていないと、対応する操作に失敗することがあります。また、デフォルトがハードコードされていることもあります。

最大パケット サイズ

デフォルト: 2097151 KB

論理パケットを複数の物理パケットに分割する方法を制御します。この値で物理パケットファイルの最大サイズを指定します。物理パケットのサイズを制限すると、ネットワークによってはパケットのデリバリーの信頼性を高めることができます。無制限にするには、0 (ゼロ) を指定します。

この値は、次のコマンドで使用されます (ただし **-maxsize** を指定していない場合に限りです)。

- mkreplica -fship
- mkreplica -ship
- syncreplica -fship
- syncreplica -ship

-out を指定して **mkreplica** または **syncreplica** を呼び出す場合、この値は使用されません。
-maxsize を使用してパケット サイズを制限する必要があります。

管理者の電子メール

デフォルト: なし

次のいずれかのイベント発生時に通知する、ユーザーの電子メール アドレスを指定します。

- ローカル ホストにある有効期限切れのパケットが発信元ホストに戻された場合。
- 次のホップにデリバリーされなかったパケットが発信元ホストに戻された場合。
- **syncreplica -import** によって、レプリカ作成パケットが発見された場合。

メモ: ClearQuest MultiSite を使用しており、ClearCase MultiSite を使用していない場合は、**control_panel** コマンドを使用して、SMTP ホストと管理者の電子メールを指定します。

電子メール通知を有効にするには

- 1 使用する SMTP ホストを指定します。
 - ClearCase MultiSite を使用する場合は、ClearCase コントロール パネルにある [SMTP ホスト] ボックスに、有効なホストが指定されていることを確認します (このボックスは、[オプション] タブにあります)。
 - ClearCase MultiSite を使用しない場合は、**control_panel** コマンドを使用して SMTP ホストを指定します。
- 2 MultiSite コントロール パネルの [管理者の電子メール] ボックスに電子メール アドレスを入力します。指定できるアドレスは 1 つのみです。
- 3 (オプション) [電子メール通知に使用するプログラムのパス] フィールドに異なる値を入力します。

電子メール通知に使用するプログラムのパス

デフォルト: `ccase-home-dir¥bin¥notify.exe`

「管理者の電子メール」のリストにある状況で呼び出される電子メールプログラムを指定します。

接続不可能なホストに対する時間制限 (分)

デフォルト: なし

前回アクセス不可として認識された宛先ホストに SHIPPING サーバーが再び接続を試みるまでの待ち時間を、分単位で指定します。

SHIPPING サーバーは、宛先ホストにパケットの送信を試み、ホストにアクセスできないと判断すると、`ccase-home-dir¥var¥shipping¥ms_downhost` ディレクトリにファイルを作成します。ファイル名は、アクセスできなかったホストの名前になります。[接続不可能なホストに対する時間制限 (分)] が設定されている場合、SHIPPING サーバーは、それ以降の発送操作時に宛先ホストのディレクトリをチェックします。

宛先ホストが `ms_downhost` ディレクトリに見つかり、現在の時刻とファイルが最後に変更された時刻との差がSHIPPING サーバー ホストのタイムアウト設定より短い場合は、SHIPPING サーバーは宛先ホストへのパケットの送信を試行しません。この差がタイムアウト設定以上だった場合は、SHIPPING サーバーは宛先ホストへのパケットの送信を試行します。[接続不可能なホストに対する時間制限 (分)] が設定されていない場合、SHIPPING サーバーは、宛先ホストへのパケット送信を試みます (アクセスできなかったホストへのパケットの送信試行には 30 秒ほどかかります)。

記憶クラス

記憶クラス名

デフォルト: `-sclass` オプションを使用する `multiutil` コマンドでは、いずれの記憶クラスにも割り当てられていないパケットの場合や、記憶クラスが構成されていないパケットの場合に、`cq_default` 記憶クラスを使用します。`cq_default` 記憶クラスは、MultiSite がインストールされている場合には作成されません。`mkorder` コマンドと `shipping_server` コマンドでは、いずれの記憶クラスにも割り当てられていないパケットの場合や、記憶クラスが構成されていないパケットの場合に、`-default` 記憶クラスを使用します。`ClearQuest MultiSite` パケット用に追加記憶クラスを作成できますが、`ClearQuest MultiSite` パケットと `ClearCase MultiSite` パケットでは別の記憶クラスを使用する必要があります。

記憶クラス名を指定します。各記憶クラスごとに、パケットの有効期限、記憶ベイ、戻りベイ、受信ハンドラの値を指定することができます。

メモ: 記憶クラス名では、大文字と小文字が区別されます。

パケットの有効期限

デフォルト: [デフォルトの有効期限を使用する] チェック ボックスがオンになっている場合、記憶クラスは `-default` クラスに関連付けられたパケットの有効期限を使用します (この値は、[パケットの有効期限] ボックスには表示されないの、値を調べるには `-default` クラスを表示する必要があります)。初めて MultiSite をインストールしたときは、`-default` クラスのパケットの有効期限は 14 日に設定されます。

指定された記憶クラスに関連付けられた発送オーダーの有効期限を日単位で指定します。期日の開始は、発送オーダー生成時点です。指定した日数内にすべての宛先にデリバリーできない場合、パケットは発信元ホストに戻され、[管理者の電子メール] ボックスで指定したアドレスにメッセージが送信されます。電子メールによる通知が有効になっていない場合は、Windows イベント ビューアにメッセージが表示されます。

値を 0 (ゼロ) にすると有効期限が指定されず、デリバーが無期限に繰り返されます。

-pexpire オプションを使用して、sync replica または mkreplica に設定を変更できます。

shipping_server プログラムは、パケットの再デリバーを行いません。[パケットの有効期限] は、デリバーされていないパケットを定期的にデリバーするようにホストを設定する場合にのみ指定します。

記憶ベイのパス

デフォルト: -sclass オプションを使用する multiutil コマンドでは、cq_default 記憶クラスを使用します。mkorder コマンドと shipping_server コマンドでは、-default 記憶クラスを使用します。ユーザーは、cq_default 記憶クラスを作成する必要があります。ClearCase MultiSite と ClearQuest MultiSite の両方を使用する場合は、VOB レプリカ パケットとデータベース パケットでは別の記憶ベイを使用する必要があります。

入力と出力の更新パケットと、特定の記憶クラスの発送オーダーを格納するディレクトリの場所を定義します。

NTFS ファイル システムの記憶ベイに置かれたパケットは、記憶ベイの Windows ACL を継承します。記憶ベイの ACL を定義して、MultiSite コマンドがパケットを正しく処理し、無許可アクセスから保護できるようにします。FAT ファイル システムに格納されているパケットは保護されません。

保存および転送機能を使用する前に、ccase-home-dir¥var¥shipping ディレクトリが作成されたディスク パーティションに、予想されるレプリカ作成パケットと更新パケット用の十分な空き容量があることを確認してください。詳細については、19 ページの「MultiSite のインストール」を参照してください。

メモ: 新しい記憶クラスを作成すると、指定した記憶ベイと戻りベイが作成されます。それらのベイの incoming ディレクトリと outgoing ディレクトリも作成されます。

戻りベイのパス

デフォルト: -sclass オプションを使用する multiutil コマンドでは、cq_default 記憶クラスを使用します。mkorder コマンドと shipping_server コマンドでは、-default 記憶クラスを使用します。ユーザーは、cq_default 記憶クラスを作成する必要があります。ClearCase MultiSite と ClearQuest MultiSite の両方を使用する場合は、VOB レプリカ パケットと データベース パケットでは別の戻りベイを使用する必要があります。

指定されたすべての宛先にデリバーできなかったために、パケットが発信元に戻される過程において、入力パケットと出力パケットを保持するディレクトリの場所を定義します。

NTFS ファイル システムの戻りベイに置かれたパケットは、戻りベイの Windows ACL を継承します。戻りベイの ACL を定義して、MultiSite コマンドがパケットを正しく処理し、無許可アクセスから保護できるようにします。FAT ファイル システムに格納されているパケットは保護されません。

受信ハンドラのパス

デフォルト: なし

記憶クラスの packets を受け取ると実行される SHIPPING サーバーのバッチ ファイルまたはプログラムを指定します。デフォルトでは、指定されているファイルはありません。

受け取った packets ごとに、`shipping_server` は以下のことを行います。

- 1 MultiSite コントロール パネルのエントリを読み取り、packets の適切な Receipt Handler 値を見つけます。
 - packets が記憶クラスと関連付けられ、その記憶クラスに Receipt Handler 値がある場合、`shipping_server` は指定されたバッチ ファイルまたはプログラムを使用します。
 - packets が記憶クラスと関連付けられておらず、`-default` 記憶クラスに Receipt Handler 値がある場合、`shipping_server` は Receipt Handler 値を使用します。
- 2 次の方法で受信ハンドラを呼び出します。

```
script-pname [ -d-ata packet-file-pname ] [ -a-ctual shipping-order-pname ]  
[ -s-class storage-class ] -o-rigin hostname
```

各パラメータの内容は次のとおりです。

<code>script-pname</code>	RECEIPT-HANDLER エントリで指定したスクリプトです。
<code>-d-ata packet-file-pname</code>	packets の場所です。このパラメータは、packets の宛先がこのホストである場合にのみ使用します。
<code>-a-ctual shipping-order-pname</code>	発送オーダーの場所です。このパラメータは、packets の宛先が別のホストである場合にのみ使用します。
<code>-s-class storage-class</code>	packets に関連付けられた記憶クラスです。このパラメータは、packets が記憶クラス作成時に関連付けられた場合にだけ使用します。
<code>-o-rigin hostname</code>	packets の最初の送信元となったマシンのホスト名です。

メモ: packets の宛先がローカル ホストと別のホストの両方である場合、`-data` パラメータと `-actual` パラメータの両方を使用します。packets はホスト上のレプリカでインポートされ、次の宛先に転送されます。

ルーティング情報

[ルーティング情報] の設定によって、パケットのネットワーク ルーティングを制御します。

次のルーティング ホップ

デフォルト: なし

[宛先ホスト名] リストで指定されているホスト名のいずれかが最終送信先になっているパケットの、次の宛先を指定します。このホストの役割は、パケットをその宛先にデリバリーすることです。ホストは、そのホスト名 (別のドメインのホストからも使用できる必要があります) または数字による IP アドレスのいずれかで指定できます。

宛先ホスト名

デフォルト: なし

このフィールドに表示されているホストが宛先となっているパケットは、[次のルーティング ホップ] ボックスで指定されているホストに送信されます。ホストは、そのホスト名 (別のドメインのホストからも使用できる必要があります) または数字による IP アドレスのいずれかで指定できます。[宛先ホスト名] としての **-default** 値は、ルーティング ホップと関連付けられていないすべてのホストを受け入れます。

multiutil

MultiSite のユーザーレベルのコマンドです。

適用

製品	コマンドタイプ
MultiSite	MultiSite コマンド

プラットフォーム
UNIX
Windows

構文

- シングルコマンド モード
`multiutil subcommand [options/args]`
- 対話型モード
`multiutil`
`multiutil> subcommand [options/args]`
`. . .`
`multiutil> quit`

説明

`multiutil` は MultiSite の主要プログラムです。ほかの `multiutil` のサブコマンドについては、40 ページの「サブコマンドの説明」を参照してください。

オプションを指定しないで入力すると、`multiutil` は対話型モードになります。コマンドによってエラーが返されると終了します。

ClearCase/ClearQuest UCM 統合を使用する場合は、この統合が必要でないコンピュータから `multiutil` を実行する必要があります。`multiutil` には、UCM 統合ではサポートされない特別なデータベース セット名が必要です。

recoverpacket

エポック数マトリックスをリセットして、損失したパケットの変更内容が再送信されるようにします。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
recoverpacket [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
              -u-ser username [ -p-assword ] password  
              [ -sin-ce date-time ] replica ...
```

説明

recoverpacket コマンドは、送信側のレプリカのエポック行をリセットし、レプリカに送った最新の同期内容を一定の時刻までに反映します。エポック行の各エクスポート時に保存されたリストを走査し、指定した時刻以前のエントリを検索します。エントリが見つかったら、関連付けられた行を使用して、指定された受信側のレプリカのエポック行をリセットします。次にエクスポートされるパケットには、損失したパケットでの変更内容も含まれます。

エポック数の自動リセット

更新パケットを別のレプリカに送信する場合、転送フェーズとインポート フェーズが正常に行われることが前提になります。そのため、送信側のレプリカのエポック数マトリックスが更新され、受信側レプリカでの変更内容が反映されます。ただし、受信側レプリカに届く前にパケットが損失すると、受信側レプリカが最新の状態であるという送信側のレプリカの前提が崩れます。

送信側のレプリカのエポック数を、パケットが送信される前の値に戻す必要があります。送信側のレプリカのエポック数マトリックスを訂正すると、受信側のレプリカに送信する次の更新パケットも同じように変更されます。

受信側のレプリカの管理者は **dumpoplog** コマンドを実行し、最後に正常にインポートされた時刻を確認する必要があります。送信レプリカ側の管理者は、この時刻を **recoverpacket** コマンドで使します。

メモ: レプリカの送信側と受信側のタイムゾーンが異なる場合や、パケットを生成と同時に送信するのではない場合 (パケットを深夜に生成して午前 6:00 に送信する場合など) は、時差を調整する必要があります。

エポック数の手動リセット

指定された時刻のエポック行が保存されていない場合、**recoverpacket** コマンドは失敗します。この場合、受信側のレプリカの管理者は **lsepoch** コマンドを使用して正しいエポック数を判別し、送信側のレプリカの管理者は送信側のレプリカ上で **chepoch** を実行してエポック行をリセットする必要があります。**chepoch** のリファレンスページを参照してください。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ: デフォルトはありません。ファミリを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ: ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ: 作業スキーマ リポジトリについての情報を復元する場合は、**MASTR** ファミリを指定します。損失したパケットがある場合は、**MASTR** とユーザー データベース ファミリの両方で **recoverpacket** を実行してから、**syncreplica** を再度実行します。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

時刻の指定

デフォルト

時刻が指定されていない場合、**recoverpacket** は現在の時刻を使用します (その結果、エポック行をリセットして最新の更新パケットの変更内容を再送信できるようにします)。

-since date-time

受信側のレプリカでパケットが最後に正常に処理された時刻を指定します。

date-time 引数は、次のいずれかの形式で指定できます。

date.time | date | time

各パラメータの内容は次のとおりです。

date := day-of-week | long-date

time := h[h]:m[m]:s[s] [UTC [[+ | -]h[h]:m[m]]]]

day-of-week := today |yesterday |Sunday | ... |Saturday |Sun | ... |Sat

long-date := d[d]-month[-[yy]yy]

month := January |... |December |Jan |... |Dec

time は、地域のタイムゾーンに合わせた 24 時間形式で時刻を指定します。**time** を省略した場合、デフォルト値は **00:00:00** です。**date** を省略した場合、デフォルト値は **today** です。世紀、年、または特定の日付を省略すると、最新のものが使用されます。タイムゾーンにかかわらず同じ時刻を適用するには、**UTC** を指定します。**UTC** 時間に対して正のオフセットを指定するにはプラス (+) 演算子を、負のオフセットを指定するにはマイナス (-) 演算子を使用します。時または分のオフセットを付けずに **UTC** を指定すると、デフォルトの設定はグリニッジ標準時 (**GMT**) になります (協定世界時 (**UTC**) 1970 年 1 月 1 日より前の日付は無効です)。

例を次に示します。

```
22-November-2002
sunday
yesterday.16:00
```

```
8-jun
13:00
today
9-Aug.10:00UTC
```

replica ...

エポック行をリセットするレプリカのサイト名です。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **boston_hub** レプリカで、**sanfran_hub** レプリカのエポック行をリセットし、2002 年 1 月 22 日以降に送信された変更内容が次の更新パケットに含まれるようにします。

```
multiutil recoverpacket -clan telecomm -site boston_hub -family DEV -user susan -p
passwd -since 22-January-2002 sanfran_hub
```

Multiutil: 22-Jan-2002.10:06:52 のエポック インフォメーションを使用しています。

Multiutil: 複製 'sanfran_hub' のエポック見積もりのリセットに成功しました。

SANFRAN_HUB: 3

関連項目

chepoch、lsepoch、restorereplica

restorereplica

バックアップから復元されたレプリカの、損失した操作を置き換えます。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
restorereplica [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
-u-ser username [ -p-assword ] password [ -force ] [ -completed ]  
[ -replace ] [ replica... ]
```

説明

警告: バックアップからレプリカを復元したら直ちにこのコマンドを実行してください。このコマンドを実行する前に、復元したレプリカで通常の開発を進めると、ファミリー内のレプリカ間に不整合が発生し、修復できなくなることがあります。

restorereplica は、バックアップから復元したレプリカでの失われた変更を次のように置き換えます。

- 1 現在のレプリカで、ほかのレプリカへの更新要求を含む特別な更新パッケージを作成します。
- 2 現在のレプリカをロックし、レプリカを復元中としてマークします。
- 3 **lsreplica -long** で、現在のレプリカに復元更新を送信する必要があるレプリカを示します。

現在のレプリカは、レプリカをファミリーの最新の状態にするために必要なすべての復元更新を受け取って適用する (**syncreplica -import** を使用) まで復元状態のままです。全体として、これらの更新内容には、バックアップ以降にファミリーに加えられたすべての変更と、失敗するまでに現在のレプリカで行われたすべての変更が含まれます。

現在のレプリカからの最後の同期エクスポート以降に行われた変更は復元できません。たとえば、レプリカを水曜日の午後 12:30 にバックアップし、最後の同期エクスポートを木曜日の午後 3:00 に実行した場合は、木曜日の午後 3:00 までに行われたすべての変更を復元できます。木曜日の午後 3:00 以降に行われた変更はすべて失われます。

レプリカ復元の手順の詳細については、107 ページの「データベース レプリカの復元」を参照してください。

レプリカのロック

`restorereplica` は現在のレプリカをロックします。現在のレプリカをロックすると、`syncreplica -export` コマンドと `syncreplica -import` コマンドの実行により復元を行っている間、現在のレプリカにこれ以外の変更が加えられることはありません。

`syncreplica` は、必要な最後の更新を適用すると、復元プロセスが完了してレプリカをアンロックしたことを示すメッセージを表示します。

復元プロセスの最適化

デフォルトでは、`restorereplica` は、レプリカがそのファミリー内のほかのすべてのレプリカから (直接的または間接的に) 復元更新を受け取ることを要求します。`syncreplica` コマンドが復元完了のメッセージを表示するのは、すべての更新内容がインポートされた後です。

場合によっては、復元プロセスの正確さを損なうことなく、この要件を緩和できます。レプリカは、1 つのレプリカ (レプリカがバックアップから復元される前に更新内容を送った最後のレプリカ) から復元更新を受け取るだけで、最新の状態に更新されます。最後に更新されたレプリカの名前またはレプリカのリスト (最後に更新されたレプリカが含まれている必要がある) を `restorereplica` に指定できます。`syncreplica` は、指定されたすべてのレプリカから復元更新を受け取った後で、復元完了メッセージを表示します。

警告: 最適化の操作を誤ると、復元したレプリカとほかのレプリカ間に不整合が発生し、修復できなくなることがあります。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリーの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、`-clan` を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、`-site` を指定する必要があります。

ファミリー: デフォルトはありません。ファミリーを指定する必要があります。

-cl an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-family family-name

ユーザー データベース ファミリ : ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ : 指定する必要はありません。ユーザー データベースファミリのメンバを復元すると、必要に応じて、関連付けられたスキーマ リポジトリ レプリカの更新も要求されます。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p assword password

指定したユーザーに関連付けられたパスワードです。

対話形式のプロンプトの非表示

デフォルト

restore replica は確認を要求します。

-f orce

確認のメッセージを表示しません。

必要な更新数の削減

デフォルト

レプリカは、そのレプリカのファミリ内のその他すべてのメンバーに復元更新を要求します。**sync replica** コマンドは、すべての更新が処理されるまでは、レプリカが完全に復元されたことを宣言しません。

警告 : このオプションの使い方を誤ると、失われたすべての変更をほかのレプリカから受け取る前に、レプリカに新しい変更が加えられる可能性があります。その場合、不整合が発生し、VOB ファミリ全体が修復できなくなることがあります。

-completed

レプリカを復元済みとしてマークして、データベースをアンロックします。これは通常の復元処理に優先します。このオプションを使用すると、それ以上復元パケット要求が送信されなくなるので、このレプリカでは復元パケットが再生されなくなります。

-rep-lace replica...

復元更新が必要なレプリカのサブセットを変更します。**replica** はサイト名で指定します。

例

レプリカの復元の例については、107 ページの「データベース レプリカの復元」を参照してください。

関連項目

chepoch、lsepoch、lsreplica、syncreplica

rmreplica

レプリカを削除します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
rmreplica [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
          -u-ser username [ -p-assword ] password  
          [ -dbset new-name ] replica
```

説明

警告: レプリカを削除するには、77 ページの「レプリカの削除」で説明しているすべての操作を行う必要があります。すべての操作を正しい順序で行わないと、データベースファミリ内のほかのレプリカで、同期とマスタースhipに関する問題が発生する可能性があります。

このコマンドは、ほかのレプリカの存在と識別情報を記録するデータベース レプリカ レコードを、現在のレプリカのデータベースから削除します。通常このコマンドは、ほかのレプリカが使用中止になったり削除されたりしたことを記録するために使用します。

制約事項

権限チェック: スーパー ユーザー権限が必要です。

その他の制約: **rmreplica** コマンドは作業スキーマ リポジトリ サイトで実行する必要があります。

オプションと引数

族、サイト、ファミリの指定

デフォルト

族: このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト: 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ: デフォルトはありません。ファミリを指定する必要があります。

-clan clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-family family-name

ユーザー データベース ファミリ: ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ: 指定する必要はありません。指定したサイトにユーザー データベース ファミリが 1 つしかない場合、そのスキーマ リポジトリも削除されます。ユーザー データベース ファミリが複数ある場合は、スキーマ リポジトリは削除されません。

デフォルト: なし

データベース セット名の変更

デフォルト

なし

-dbset new-name

このオプションは、族の最後のレプリカを削除する場合にのみ使用します。族の最後のレプリカを削除する場合は、データベース セットの名前を、**ClearQuest MultiSite** フラグを含まないように変更する必要があります。

レプリカの指定

デフォルト

削除するレプリカがある場所でコマンドを実行する場合、デフォルトは現在のレプリカです。それ以外の場所でコマンドを実行する場合は、レプリカを指定する必要があります。

replica

現在のレプリカのデータベースから削除するレプリカのサイト名です。

例

rmreplica コマンドの使用例については、77 ページの「レプリカの削除」を参照してください。

関連項目

chmaster、mkreplica

scruboplog

レプリカの **oplog** 項目を削除します。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

```
scruboplog [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name  
           [ -u-ser username ] [ -p-assword ] password  
           -before { date-time | oplog-ID }
```

説明

操作ログ (**oplog**) 項目は長期間にわたってレプリカ内に保持する必要があります。**oplog** 項目は、レプリカがほかのすべてのレプリカに送信する更新パケットを生成するときに使用されます。ほかのレプリカを障害から回復させるために、**oplog** 項目が必要になる場合もあります。

ただし、必要に応じて **oplog** 項目を削除 (スクラブ) し、レプリカがあるハードディスクを最適化してもかまいません。**scruboplog** コマンドを使用すると、不要になったレプリカの **oplog** を削除することができます。

oplog 項目は現在のレプリカで行われた変更のみを記録しますが、時間が経過するにつれて、この情報の保持にデータ自体と同じくらいの容量が必要になることがあります。

レプリカの **oplog** 項目をスクラブする前に、その項目がもう必要ないことと、レプリカの **oplog** から削除する情報がファミリー内のほかのレプリカに含まれていることを確認する必要があります。また、**oplog** をスクラブする前に、レプリカを同期する必要もあります。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数

族、サイト、ファミリーの指定

デフォルト

族：このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト：現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリー：デフォルトはありません。ファミリーを指定する必要があります。

-clan clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-family family-name

ユーザー データベース ファミリ：ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ：ファミリー名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-user user

スーパー ユーザー権限を持つユーザーの名前です。

-password password

指定したユーザーに関連付けられたパスワードです。

削除する項目の指定

デフォルト

削除する項目を指定する必要があります。

-before { date-time | oplog-ID }

指定した **date-time** より前、または **oplog-ID** より前のすべての **oplog** 項目を削除します。引数の値自体は含まれません。つまり、指定した日付または時刻に作成された **oplog** 項目は削除されません。指定した **ID** の **oplog** 項目も削除されません。

oplog-ID 引数は整数で指定します。レプリカの操作ログを表示するには、`dumpoplog` を使用します。`date-time` 引数は、次のいずれかの形式で指定できます。

`date.time | date | time`

各パラメータの内容は次のとおりです。

`date` := `day-of-week | long-date`

`time` := `h[h]:m[m]:s[s] [UTC [[+ | -]h[h]:m[m]]]]`

`day-of-week` := `today | yesterday | Sunday | ... | Saturday | Sun | ... | Sat`

`long-date` := `d[d]-month[-[yy]yy]`

`month` := `January |... |December |Jan |... |Dec`

`time` は、地域のタイムゾーンに合わせた 24 時間形式で時刻を指定します。`time` を省略した場合、デフォルト値は `00:00:00` です。`date` を省略した場合、デフォルト値は `today` です。世紀、年、または特定の日付を省略すると、最新のものが使用されます。タイムゾーンにかかわらず同じ時刻を適用するには、`UTC` を指定します。`UTC` 時間に対して正のオフセットを指定するにはプラス (+) 演算子を、負のオフセットを指定するにはマイナス (-) 演算子を使用します。時または分のオフセットを付けずに `UTC` を指定すると、デフォルトの設定はグリニッジ標準時 (GMT) になります (協定世界時 (UTC) 1970 年 1 月 1 日より前の日付は無効です)。

例を次に示します。

```
22-November-2002
sunday
yesterday.16:00
8-jun
13:00
today
9-Aug.10:00UTC
```

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- **boston_hub** レプリカの DEV ファミリにある 2002 年 1 月 21 日より前の **oplog** 項目を削除します。

```
multiutil scruboplog -clan telecomm -site boston_hub -family DEV -user susan -p passwd  
-before 21-January-2002
```

- **tokyo** レプリカの TEST ファミリにある 1 ～ 300 までの **oplog** 項目を削除します。

```
multiutil scruboplog -clan testing -site tokyo -family TEST -user masako -p passwd  
-before 301
```

関連項目

dumpoplog、sync replica

shipping.conf

保存および転送構成ファイルです。

適用

製品	コマンドタイプ
MultiSite	MultiSite データ構造

プラットフォーム
UNIX

構文

`/var/adm/rational/clearcase/config/shipping.conf`

説明

このファイルは、各ホストの保存および転送機能の操作を制御します。コメント行(`#`で始まる)と1つ以上の設定エントリで構成され、次に説明する設定エントリを含めることができます。エントリがないと、対応する保存および転送の操作が失敗することがあり、デフォルトでハードコードされていることもあります。

MultiSite をインストールすると `ccase-home-dir/config/services/shipping.conf.template` ファイルが作成され、その中にこれらすべてのエントリが定義されます。

`/var/adm/rational/clearcase/config/shipping.conf` が存在しない場合は、インストール時にテンプレートファイルをコピーすることで作成されます。

`/var/adm/rational/clearcase/config/shipping.conf` が存在する場合は、インストール時に既存のファイルとテンプレートを比較して必要な変更を行います。

メモ: MultiSite または Rational Shipping Server をデフォルトのインストールディレクトリ(`/opt/rational/clearcase`)にインストールしない場合は、`shipping.conf` ファイルを編集して、`/opt/rational/clearcase` の部分をインストールディレクトリのパス名に変更する必要があります。

パケットサイズ

MAX-DATA-SIZE size [k | m | g]

デフォルト: 2097151 KB

個別の論理パケットを複数の物理パケットに分割する方法を制御します。物理パケットのサイズを制限すると、ネットワークによってはパケットのデリバーの信頼性を高めることができます。size の整数 (オプションで **k**、**m**、**g** のサフィックスを付けます) で、物理パケットファイルの最大サイズを指定します。**k** は KB (キロバイト)、**m** は MB (メガバイト)、**g** は GB (ギガバイト) を表します。サフィックスがない場合は KB を表します。無制限にするには、**0** (ゼロ) を指定します。

この値は、次のコマンドで使用されます (ただし **-maxsize** を指定していない場合にかぎります)。

- mkreplica -fship
- mkreplica -ship
- syncreplica -fship
- syncreplica -ship

-out を指定して **mkreplica** または **syncreplica** を呼び出す場合、この値は使用されません。

-maxsize を使用してパケットサイズを制限する必要があります。

通知

NOTIFICATION-PROGRAM e-mail-program-pathname

デフォルト: /opt/rational/clearcase/bin/notify。NOTIFICATION-PROGRAM エントリが存在しない場合、このプログラムも使用されます。

次のような場合に呼び出される電子メール プログラムです。

- 処理しようとする発送オーダーの有効期限が切れかけていることを **shipping_server** が見つけた場合
- デリバーできなかったパケットが、別のホストの **shipping_server** から発信元のホストに戻された場合 (EXPIRATION の説明を参照してください)
- **syncreplica -import** が、**mkreplica** で処理されるはずのレプリカ作成パケットを検出した場合

メール プログラムは次の方法で呼び出します。

e-mail-program-pathname -s subject -f message-file addr ...

管理者アドレス

ADMINISTRATOR e-mail-address

デフォルト: root

ローカル ホストで保存および転送機能を管理する管理者の電子メール アドレスです。

メール メッセージは、「通知」のリストにある状況で、指定されたアドレスに送信されます。構成ファイルには複数の **ADMINISTRATOR** エントリがあり、メッセージは指定したすべてのメールアドレスに送信されます。

記憶ベイと戻りベイ

STORAGE-BAY storage-class directory-pathname

RETURN-BAY storage-class directory-pathname

デフォルト: **-sclass** オプションを使用する **multiutil** コマンドでは、いずれの記憶クラスにも割り当てられていないパケットの場合や、記憶クラスが構成されていないパケットの場合に、**cq_default** 記憶クラスを使用します。**mkorder** コマンドと **shipping_server** コマンドでは、いずれの記憶クラスにも割り当てられていないパケットの場合や、記憶クラスが構成されていないパケットの場合に、**-default** 記憶クラスを使用します。

これらの行で記憶ベイと戻りベイのディレクトリを定義します。記憶ベイは、入力と出力の更新パケットと、特定の記憶クラスの発送オーダーを保持します。戻りベイは、指定されたすべての宛先にデリバリーできなかったために、パケットが発信元に戻される過程において、入力パケットまたは出力パケットを保持します。

複数の **STORAGE-BAY** エントリと **RETURN-BAY** エントリを使用して、記憶クラス用に複数のベイを定義できます。**shipping_server** は、ベイのディスク容量に基づいて、パケットごとにベイを 1 つ選択します。ベイを指定した順序は関係ありません。

メモ: 記憶クラス名では、大文字と小文字が区別されます。

MultiSite をインストールすると、**-default** という名前のデフォルトの記憶クラスが作成されます。このクラス用の記憶クラスと戻りクラスは、ローカル ホストの **/var/adm/rational/clearcase/shipping** ディレクトリに作成されます。各ベイには **incoming** と **outgoing** というサブディレクトリがあり、それぞれ入力パケットと出力パケットを保持します。発送操作では、これらのサブディレクトリでパケットを検索します。保存および転送機能を使用する前に、**shipping** ディレクトリが作成されたディスク パーティションに、予想されるレプリカ作成パケットと更新パケット用の十分な空き容量があることを確認してください。

-sclass オプションを使用する multiutil コマンドでは、いずれの記憶クラスにも割り当てられていないパケットの場合や、記憶クラスが構成されていないパケットの場合に、cq_default 記憶クラスを使用します。cq_default 記憶クラスは、MultiSite がインストールされている場合には作成されません。mkorder コマンドと shipping_server コマンドでは、いずれの記憶クラスにも割り当てられていないパケットの場合や、記憶クラスが構成されていないパケットの場合に、-default 記憶クラスを使用します。ClearQuest MultiSite パケット用に追加記憶クラスを作成できますが、ClearQuest MultiSite パケットと ClearCase MultiSite パケットでは別の記憶クラスを使用する必要があります。

標準の UNIX コマンド mkdir を使用して、directory-pathname を作成する必要があります。また、新しいベイに incoming ディレクトリと outgoing ディレクトリを作成する必要があります。1 つのベイに配置したパケットには、ベイと同じ所有者、グループ、読み書き権限が割り当てられます (ベイでの実行権限とその他の特別な権限は無視されます)。必要に応じてこれらの権限を調整して、MultiSite コマンドがパケットを正しく処理し、無許可アクセスから保護できるようにします。

メモ: incoming ディレクトリと outgoing ディレクトリは、同じファイル システム上に存在している必要があります。

有効期限

EXPIRATION storage-class number-of-days

EXPIRATION -default number-of-days

デフォルト: -default の場合は 14 日間、cq_default の場合はなし (ユーザーが有効期限を指定する必要があります)。

指定された記憶クラスに関連付けられた発送オーダーの有効期限を日単位で指定します。期日の開始は、発送オーダー生成時点です。指定した日数内にすべての宛先にデリバーできない場合、パケットは発信元ホストに戻され、1 つ以上の電子メール メッセージが送信されます (「管理者 アドレス」と「通知」の説明を参照してください)。

cq_default を記憶クラスとして指定すると、いずれの記憶クラスにも割り当てられていない発送オーダーの場合や、記憶クラスが構成されていない発送オーダーの場合に、有効期限が指定されます。ただし、mkorder コマンドを使用して発送オーダーを作成し、記憶クラスを指定しないと、発送オーダーには -default 記憶クラスに関連付けられた有効期限が指定されます。

値を 0 (ゼロ) にすると有効期限が指定されず、デリバーが無期限に繰り返されます。

-pexpire オプションを使用して、sync replica または mkreplica に設定を変更できます。

shipping_server プログラムは、パケットの再デリバーを行いません。EXPIRATION の指定は、 SHIPPING サーバーを定期的呼び出してデリバーできなかったパケットをデリバーしようとする場合に便利です。

パケットルーティング

ROUTE next-hop host ...
ROUTE next-hop -default

デフォルト: なし

パケットのネットワーク ルーティングを制御します。最終的な宛先が **host** 引数のいずれかであるパケットは、**next-hop** というホストに送信されます。このホストの役割は、パケットを最終的な宛先 (または別の転送先) にデリバリーすることです。**next-hop** や **host** はホスト名 (別のドメインのホストも使用できるもの) または数字による IP アドレスになります。

構成ファイルには複数の **ROUTE** エントリを含めることができます。**-default** は特別なキーワードで、別の **ROUTE** エントリで指定されていないすべてのホストに対応します。

受信ハンドラ

RECEIPT-HANDLER storage-class script-pathname

デフォルト: なし

SHIPPING サーバーのスクリプトを指定して、発送ペイで受信したパケットごとに実行します。

受信した各パケットは、**shipping_server** により次のように処理されます。

1 **shipping.conf** ファイルを読み取り、パケットに該当する **RECEIPT-HANDLER** エントリを検索します。

- パケットが記憶クラスに関連付けられ、その記憶クラスに **RECEIPT-HANDLER** エントリがある場合、**shipping_server** はそのエントリで指定した **script-pathname** を使用します。
- パケットが記憶クラスに関連付けられておらず、**-default** 記憶クラスに **RECEIPT-HANDLER** 値がある場合、**shipping_server** はその値を使用します。

2 次の方法で受信ハンドラを呼び出します。

script-pname [-d-ata packet-file-pname] [-a-ctual shipping-order-pname]
[-s-class storage-class] -o-rigin hostname

各パラメータの内容は次のとおりです。

script-pname	RECEIPT-HANDLER エントリで指定したスクリプトです。
-d-ata packet-file-pname	パケットの場所です。このオプションは、パケットの宛先がこのホストである場合にのみ使用します。
-a-ctual shipping-order-pname	発送オーダーの場所です。このオプションは、パケットの宛先が別のホストである場合にのみ使用します。

-s-class storage-class

パケットに関連付けられた記憶クラスです。このオプションは、パケットが記憶クラス作成時に関連付けられた場合にのみ使用します。

-o-rigin hostname

パケットの最初の送信元となったマシンのホスト名です。

メモ: パケットの宛先がローカル ホストと別のホストの両方である場合、**-data** パラメータと **-actual** パラメータの両方を使用します。パケットはホスト上のレプリカでインポートされ、次の宛先に転送されます。

ポート番号

CLEARCASE_MIN_PORT port-number

CLEARCASE_MAX_PORT port-number

デフォルト: なし

注意: これらのエントリは、ファイアウォールを介して通信可能で、**MultiSite** の SHIPPING サーバー専用オプションを指定してインストールされたホスト上でのみ設定します。SHIPPING サーバーをファイアウォール システム上で使用するには、**clearcase** スクリプトで **CLEARCASE_MIN_PORT** と **CLEARCASE_MAX_PORT** 環境変数も設定する必要があります。詳細については、56 ページの「ポート値の指定」を参照してください。

これらのエントリは、SHIPPING サーバーがファイアウォール システム上で使用するポート値の範囲を指定します。また、SHIPPING サーバー環境での環境変数として設定されます。

値を設定する場合のガイドラインは、次のとおりです。

- **CLEARCASE_MIN_PORT** の値の範囲は 1024 ～ 65534 までです。
- **CLEARCASE_MAX_PORT** の値の範囲は 1025 ～ 65535 までです。
- **CLEARCASE_MAX_PORT** の値は **CLEARCASE_MIN_PORT** の値よりも大きい必要があります。
- ダイナミック/プライベート ポート範囲である 49152 ～ 65535 の範囲を使用することをお勧めします。

ホストにアクセスできない場合のタイムアウト時間

DOWNHOST-TIMEOUT minutes

デフォルト: なし

前回アクセス不可として認識された宛先ホストに SHIPPING サーバーが再び接続を試みるまでの待ち時間を、分単位で指定します。

SHIPPING サーバーは、宛先ホストにパケットの送信を試み、ホストにアクセスできないと判断すると、`/var/adm/rational/clearcase/shipping/ms_downhost` ディレクトリにファイルを作成します。ファイル名は、アクセスできなかったホストの名前になります。次のいずれかのパラメータが指定された場合は、 SHIPPING サーバーはそれ以降の発送操作時に宛先ホストのディレクトリをチェックします。

- `shipping.conf` ファイルの `DOWNHOST-TIMEOUT` 設定
- `SHP_DOWNHOST_TIMEOUT_RETRY` 環境変数

両方のパラメータが設定されている場合、 SHIPPING サーバーは `DOWNHOST-TIMEOUT` を使用します。

宛先ホストが `ms_downhost` ディレクトリに見つかり、現在の時刻とファイルが最後に変更された時刻との差が SHIPPING サーバー ホストのタイムアウト設定より短い場合は、 SHIPPING サーバーは宛先ホストへのパケットの送信を試行しません。この差がタイムアウト設定以上だった場合は、 SHIPPING サーバーは宛先ホストへのパケットの送信を試行します。この設定と `SHP_DOWNHOST_TIMEOUT_RETRY` 環境変数のどちらも設定されていない場合は、 SHIPPING サーバーは宛先ホストへのパケットの送信を試行します (アクセスできなかったホストへのパケットの送信試行には 30 秒ほどかかります)。

shipping_server

保存および転送パケット転送サーバー。

適用

製品	コマンドタイプ
MultiSite	MultiSite コマンド

プラットフォーム
UNIX
Windows

構文

```
shipping_server [ -scl:ass storage-class-name ] { -pol:l | sources ... }
```

このコマンドは、UNIX の場合は `ccase-home-dir/etc` に、Windows の場合は `ccase-home-dir\bin` にあります。

説明

このコマンドは、ローカル ホストで 1 つ以上の発送オーダーを処理し、関連付けられたパケットまたはファイルをリモート サイトに送信します。**shipping_server** は、ファイルをすべての宛先にデリバリーし、その宛先にローカル ホストでないものがある場合はファイルを削除します。

メモ: **shipping_server** は、発送オーダーの処理を開始するときにオーダーをロックします。ロックすることにより、引き続き **shipping_server** を呼び出してオーダーが処理されるのを防ぎます。

TCP/IP 接続

ファイルを転送するため、**shipping_server** は UDP を使用して受信側のホスト上の **albd_server** プロセスに接続し、**albd_server** は受信側のホスト上で **shipping_server** を受信モードで呼び出します。

ファイアウォールを介してパケットを送信する場合 (`CLEARCASE_MIN_PORT` 環境変数と `CLEARCASE_MAX_PORT` 環境変数が設定されている場合)、**shipping_server** は TCP を使用してリモートの **albd_server** に接続しようとします。接続に失敗すると、**shipping_server** は UDP を使用します。詳細については、53 ページの「ファイアウォールを通過する保存および転送の使用法」を参照してください。

UNIX では、shipping_server は送信するパケットごとに 1 つのサブプロセスをフォークします。10 の shipping_server サブプロセスがあり、それぞれが 1 つのパケットを送信しようとするすると、shipping_server の呼び出しを開始することがあります。受信側でも同じ数のサブプロセスがフォークされます。1 つのサブプロセスが終了すると別のサブプロセスが開始されますが、同時に実行できるのは 10 までです。

2 つの shipping_server プロセスの間で TCP 接続が行われると、ファイルが転送されます。受信側の shipping_server は、shipping.conf ファイル (UNIX) または MultiSite コントロールパネル (Windows) の構成設定を使用して記憶ベイを選択します。記憶クラスが複数の記憶ベイに割り当てられている場合、利用可能なディスク容量によりベイが選択されます。

UNIX の場合、パケット ファイルは、記憶ベイ ディレクトリと同じ所有者やグループが作成し、アクセス モードはディレクトリの読み書き権限で決まります (記憶ベイでの実行権限とそのほかの特別な権限があっても無視されます)。

Windows の場合、パケット ファイルは、記憶ベイ ディレクトリの Windows ACL から権限を継承します。

パケット名でのコロン文字

パケット名にコロン (:) がある場合、shipping_server は処理の際にコロンをピリオド (.) に変換します。Windows マシンではファイル名にコロンが使用できませんが、変換することによりパケットを Windows マシンにデリバリーできるようになります。

ファイル名の競合の処理

関連付けられた発送オーダーと同じディレクトリにあるパケットのないファイルであれば、mkorder コマンドと shipping_server コマンドを使用して転送できます。同じ名前のファイルが受信ホストに既にある場合、新しいファイルは filename_1 のように名前が変更されます (さらに別のファイルを同じ名前を送信した場合は filename_2 になり、以後も同じように変更されます)。

ホストにアクセスできない場合のタイムアウト時間の設定

タイムアウト時間を設定して、その間は前回アクセス不可として認識されたホストに SHIPPING サーバーがパケットを送信しないようにすることができます。詳細については、shipping.conf (UNIX) または MultiSite コントロール パネル (Windows) のリファレンス ページを参照してください。

ログ

UNIX の場合、shipping_server は、送受信したすべてのパケットの記録とすべてのエラーをファイル /var/adm/rational/clearcase/log/shipping_server_log に書き込みます。

Windows の場合、shipping_server は、送受信したすべてのパケットの記録、通知メッセージ、ログ メッセージ、すべてのエラーを Windows のイベント ビューアに書き込みます。

制約事項

権限チェック: 発送オーダーが格納されるディレクトリに対して、書き込み権限と実行権限を持っている必要があります。UNIX では、データ ファイルを所有しているか **root** ユーザーである必要があります。

ロック: ロックは適用されません。

マスターシップ: マスターシップの制限はありません。

その他の制約: 発送オーダーと指定するデータ ファイルは同じディレクトリに設定する必要があります。

オプションと引数

記憶クラスへの処理の制限

デフォルト

-poll を指定すると、このホスト上のすべての送信記憶ベイと戻りベイにあるすべての発送オーダーが処理されます。sources を指定すると、指定されたすべての発送オーダーが処理されます。

-sclass storage-class-name

指定した記憶クラスの発送オーダーだけを処理します。

発送オーダーの指定

デフォルト

なし

-poll

shipping.conf ファイル (UNIX) または MultiSite コントロール パネル (Windows) で定義した一部 (-sclass を使用する場合) またはすべての記憶ベイと戻りベイに格納されている発送オーダーを処理します。

メモ: shipping_server は、ファイル名が sh_o_ で始まる発送オーダーだけを処理します。発送オーダーを作成する場合は、この命名規則に従うか、-poll オプションを省略して発送オーダー パス名を指定します。

UNIX では、所有している発送オーダー ファイルだけが処理されます。ただし、ルートがこのプログラムを実行する場合、発送オーダー ファイルは所有権にかかわらず処理されます。

sources ...

ファイルかディレクトリ、またはその両方の 1 つ以上のパス名。有効な発送オーダーがあれば、指定したファイルごとに処理されます。指定するディレクトリごとに、**shipping_server** はそのディレクトリに格納されている発送オーダーの一部 (**-sclass** を使用する場合) またはすべてを処理します。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

- すべての MultiSite 記憶ベイにあるすべての発送オーダーを処理します。

```
shipping_server -poll
```

< 出力がないことは、コマンドが正常に終了したか、発送オーダーが見つからなかったことを意味します >

- 特定の発送オーダーを処理します。パス名引数が指定するのは発送オーダー ファイルです。転送するデータ ファイルではありません。

```
/opt/rational/clearcase/etc/shipping_server
```

```
/var/adm/rational/clearcase/shipping/ms_ship/sh_o_sync_sydney_19-May-02.09:48:45_7660_1
```

< 出力がないことはコマンドが正常に終了したことを意味します >

- 指定されたディレクトリにあるすべての発送オーダー ファイルを処理します。

```
shipping_server "c:¥Program
```

```
Files¥Rational¥ClearCase¥var¥shipping¥ms_ship¥outgoing"
```

< 出力がないことは、コマンドが正常に終了したか、発送オーダーが見つからなかったことを意味します >

- 指定された記憶クラスの記憶ベイにあるすべての発送オーダーを処理します。

```
/opt/rational/clearcase/etc/shipping_server -poll -sclass daily
```

< 出力がないことは、コマンドが正常に終了したか、発送オーダーが見つからなかったことを意味します >

関連項目

mkorder、MultiSite コントロール パネル、shipping.conf、sync replica、
第 10 章 MultiSite の操作のトラブルシューティング

sync replica

更新パッケージをエクスポートまたはインポートします。

適用

製品	コマンドタイプ
MultiSite	multiutil のサブコマンド

プラットフォーム
UNIX
Windows

構文

- 更新パッケージをエクスポートします。

```
sync-replica -exp-ort [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name
-u-ser username [ -p-assword ] password
[ -max-size max-packet-size [ -lim-it num-packets ] ]
[ -nblocking ] [ -compress ]
{ { -sh-ip | -fsh-ip } -wor-kdir directory
[ -sc-lass storage-class ] [ -pex-pire date ] [ -not-ify email ]
| -out { packet-file-pname | staging-area-pname } }
replica ...
```

- 更新パッケージをインポートします。

```
sync-replica -imp-ort [ -cl-an clan-name ] [ -site site-name ] -fam-ily family-name
-u-ser username [ -p-assword ] password
[ -nblocking ]
{ -rec-eive [ -sc-lass storage-class ]
| { packet-file-pname | staging-area-pname } ... }
```

説明

レプリカと 1 つ以上のシブリング レプリカを同期するプロセスは、3 つのフェーズで構成されています。

- 1 一方のサイトで、**syncreplica -export** コマンドを使用して更新パッケージを作成します。このパッケージには、そのサイトと場合によってはほかのサイトのレプリカで発生した変更が含まれています。
- 2 生成されたパッケージを 1 つ以上のほかのサイトに送信します。
- 3 もう一方のサイトでは、**syncreplica -import** コマンドを使用して更新パッケージの変更内容を同じデータベースのレプリカに適用します。この手順は、パッケージを受け取ったすべてのサイトで実行します。

更新パッケージの内容

- 宛先レプリカに対して最後の更新が生成されてから、現在のレプリカで発生したすべての変更 (すでに宛先レプリカに送信した変更内容は、パッケージから除外されています)。
- 別のレプリカで発生した変更内容。現在のレプリカでは該当するレプリカからすでに更新パッケージで受け取っていますが、宛先レプリカにまだ渡していない変更です。

いずれの場合でも、**syncreplica -export** は指定されたすべての宛先で使用する論理更新パッケージを 1 つ作成します。このパッケージを使用して、特定のレプリカのみを更新できます。

エクスポート フェーズでの注意点

MultiSite は、レプリカを効率的に更新するように設計されています。**syncreplica -export** は、既に対応している操作を除外しようとします (ただし、同じレプリカに操作を何度も送信しても問題はありません。最初の操作がインポートされ、残りの同一操作は無視されます)。

syncreplica -export は、**-workdir** オプションで指定したディレクトリにテンポラリ ファイルを保存します。このディレクトリが事前に存在していることはありません。エクスポート パッケージが作成された後で削除されます。

インポート フェーズでの注意点

更新パッケージが適用されるのは、パッケージを受信した同期サーバーに関連付けられた該当するレプリカです。特定のレプリカや格納場所を指定する必要はありません。

インポート プロセスでは、更新パッケージが適切な順序で適用されます。そのため、コマンド行ではどのような順序でパッケージを指定してもかまいません。

インポート フェーズ中にデータベース レプリカが通常のデータベース操作に対してロックされることはありませんが、ほかのすべての **MultiSite** 操作に対してはロックされます。

パケットのスキップ

`syncreplica -import` は、次の場合、更新パケットを処理しません。

- 更新パケットに、このレプリカにまだインポートされていないほかの変更に依存する変更が含まれている場合。一般的に、このレプリカ宛の更新パケットがまだ送信されていないか、転送中に行方不明になった場合を指します。
- マルチパート論理パケットで、初期の物理パケットの処理に問題が発生した場合。

このような場合、`syncreplica -import` は説明メッセージを表示します。

障害パケットと再生パケットの更新

`syncreplica -import` が、レプリカに対して操作を開始した後にエラー メッセージを表示して終了することがあります。たとえば、別のプロセスがデータベースをロックしていると、インポートが失敗します。データベースがロック解除された後で、`syncreplica -import` を実行して再び更新パケット全体を処理できます。

既に正しく処理されている更新パケットをインポートしても問題ありません。同じ変更が 2 度行われることはありません。

更新時の障害についての詳細は、101 ページの「失われたパケットの回復」を参照してください。

更新パケットの削除

1 回の `syncreplica -import` 呼び出しで、同期サーバーに関連付けられたすべてのターゲットレプリカにパケットを正しく適用できた場合、更新パケットはコマンド終了時に削除されます。ただし、パケットが複数の `syncreplica -import` コマンドで処理される場合、パケットは削除されません。

フックの起動

ClearQuest フックは、パケットのインポート中に行われた変更に対しては起動しません。

競合する名前の対処

`syncreplica` は、異なるレプリカで作成されたオブジェクト間の名前の競合を解決します。詳細については、102 ページの「名前の競合の解決」を参照してください。

更新の遅延

`syncreplica` は、ClearQuest ユーザーにレプリカの更新内容について通知しません。すべてのアクティブなユーザーは、ClearQuest の標準のデータベース ポーリング ルーチンを介して数秒間で更新内容を表示できます。

パケットをデリバーできなかった場合のエラー処理

パケットをデリバーできなかった場合は、保存および転送機能を使用して、そのパケットが元になるレプリカの同期サーバーに戻されます。メール メッセージが保存および転送の管理者に送られます。これは、繰り返し試みたパケットのデリバーがすべて失敗し、割り当てられた時間を使い果たした場合に発生します。また、宛先のホストが不明な場合や、データ ファイルが存在しない場合にも発生します。有効期限、管理者の電子メールアドレス、通知プログラムは、保存および転送の構成設定により指定されます。

制約事項

スーパー ユーザー権限が必要です。

オプションと引数 : エクスポート フェーズ

族、サイト、ファミリの指定

デフォルト

族 : このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト : 現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ : デフォルトはありません。ファミリを指定する必要があります。

-cl-an clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-fam-ily family-name

ユーザー データベース ファミリ : ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ : ファミリ名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-u-ser user

スーパー ユーザー権限を持つユーザーの名前です。

-p-assword password

指定したユーザーに関連付けられたパスワードです。

更新パケットサイズの指定

デフォルト

-maxsize を指定していない場合、使用する発送方法によってデフォルトのパケットサイズが異なります。

- -ship または -fship を指定して作成したパケットは、shipping.conf ファイル (UNIX) や MultiSite のコントロールパネル (Windows) で指定した最大パケットサイズを超えることはありません。
- -out を指定して作成したパケットが 2 GB を超えることはありません。

-max-size max-packet-size [-limit num-packets]

物理パケットの最大サイズで、数字に 1 文字を付けて表します。例を次に示します。

500k 500 キロバイト

20m 20 メガバイト

1.5g 1.5 ギガバイト

-limit オプションは syncreplica が生成するパケット数を制限します。各パケットは max-packet-size を超えることはありません。このオプションは、発送ベイやステージング領域のディスク容量が制限されている場合に使用します。

更新パケットの配置

デフォルト

なし。syncreplica -export で作成した更新パケットの保存方法か、ほかのサイトへの転送方法、またはその両方を指定する必要があります。-sclass オプションを省略して -ship または -fship を使用すると、syncreplica は、shipping.conf ファイル (UNIX) または MultiSite のコントロールパネル (Windows) の cq_default クラスに指定した記憶ベイの場所にパケットを配置します。

-ship

-fship

保存および転送記憶ベイの 1 つ以上のファイルに更新パケットを保存します。syncreplica は各物理パケットに個別の発送オーダーを作成し、デリバリー方法と宛先を示します。宛先は、レプリカ データベースで replica-name 引数に関連付けられた同期サーバーになります (同期サーバーの関連付けは、mkreplica -export を使用して作成し、chreplica を使用して変更できます)。

-fship (強制発送) を使用すると、 SHIPPING サーバーが呼び出され、更新パケットが直ちに送信されます。-ship を指定してもこのサーバーは呼び出されません。

-wor-kdir directory

sync replica が使用する一時的な作業ディレクトリです。このディレクトリが事前に存在していることはありません。**sync replica** エクスポート プロセスが完了すると削除されます。

-scl-lass class-name

パケットと発送オーダーの記憶クラスを指定します。**sync replica** は、UNIX の **shipping.conf** ファイルまたは Windows の **MultiSite** のコントロール パネルにある記憶クラスを検索し、使用する記憶ベイの場所を決定します。

-out packet-file-pname

最初の更新パケットの名前です。ほかにも物理パケットがある場合、**packet-file-pname_2**、**packet-file-pname_3** のように順に名前を付けたファイルに配置されます。

更新パケットは自動的にデリバリーされるわけではありません。適切な方法を使用してデリバリーしてください。**-out** でパケットを作成し、その後、保存および転送機能を使用してデリバリーできます。**mkorder** リファレンス ページを参照してください。

staging-area-pname

パケット ファイルを保存するディレクトリです。

パケットをデリバリーできなかった場合の処理

デフォルト

パケットをデリバリーできなかった場合は、保存および転送機能を使用して元になるレプリカの同期サーバーに戻されます。メール メッセージが保存および転送の管理者に送られます。これは、繰り返し試みたパケットのデリバリーが失敗し、割り当てられた時間を使い果たした場合に発生します。また、宛先のホストが不明な場合や、データ ファイルが存在しない場合にも発生します。有効期限、管理者の電子メールアドレス、通知プログラムは、保存および転送の構成設定により指定されます。

-pex-pire date-time

保存および転送機能がパケットのデリバリーを中止して、障害を通知するメールを生成する時刻を指定します。このオプションは、**shipping.conf** ファイル (UNIX) または **MultiSite** のコントロール パネル (Windows) の記憶クラスに指定した有効期限を上書きします。

date-time 引数は、次のいずれかの形式で指定できます。

date.time | date | time

各パラメータの内容は次のとおりです。

date **:=** day-of-week | long-date

time **:=** h[h]:m[m]:s[s] [UTC [[+ | -]h[h]:m[m]]]]

day-of-week := today |yesterday |Sunday | ... |Saturday |Sun | ... |Sat
long-date := d[d]-month[-[yy]yy]
month := January |... |December |Jan |... |Dec

time は、地域のタイムゾーンに合わせた 24 時間形式で時刻を指定します。**time** を省略した場合、デフォルト値は 00:00:00 です。**date** を省略した場合、デフォルト値は **today** です。世紀、年、または特定の日付を省略すると、最新のものが使用されます。タイムゾーンにかかわらず同じ時刻を適用するには、**UTC** を指定します。**UTC** 時間に対して正のオフセットを指定するにはプラス (+) 演算子を、負のオフセットを指定するにはマイナス (-) 演算子を使用します。時または分のオフセットを付けずに **UTC** を指定すると、デフォルトの設定はグリニッジ標準時 (GMT) になります (協定世界時 (UTC) 1970 年 1 月 1 日より前の日付は無効です)。

例を次に示します。

```
22-November-2002  
sunday  
yesterday.16:00  
8-jun  
13:00  
today  
9-Aug.10:00UTC
```

-not-ify e-mail-address

指定された電子メール アドレスに、デリバリーの失敗を知らせるメッセージを送信します。

電子メール通知が使用できない Windows ホスト上で障害が発生した場合は、Windows イベント ビューアにメッセージが表示されます。メッセージには、このオプションで指定された **e-mail-address** 値と、このユーザーに操作の状態を通知するよう求めるメモが含まれています。電子メールでの通知を有効にする方法については、**MultiSite コントロール パネル**のリファレンス ページを参照してください。

宛先レプリカの指定

デフォルト

なし。

replica ...

宛先レプリカのサイト名です。1 つ以上の宛先レプリカを指定できます。たとえば、**boston_hub** は、**boston_hub** が更新パケットを受け取ることを示し、**boston_hub bangalore** は、**boston_hub** と **bangalore** の両方が更新パケットを受け取ることを示します。

オプションと引数：インポート フェーズ

族、サイト、ファミリの指定

デフォルト

族：このサイトでレプリカされた最初の族です。サイトに複数の族がある場合は、**-clan** を指定する必要があります。

サイト：現在のサイトです。このホストに複数のサイトがある場合は、**-site** を指定する必要があります。

ファミリ：デフォルトはありません。ファミリを指定する必要があります。

-clan clan-name

レプリカの族の名前です。

-site site-name

レプリカのサイトの名前です。

-family family-name

ユーザー データベース ファミリ：ユーザー データベースが作成されたときに付けられたデータベース名です。

スキーマ リポジトリ ファミリ：ファミリ名は **MASTR** です。

ユーザー名とパスワードの指定

デフォルト

ユーザー名とパスワードを指定する必要があります。

-user user

スーパー ユーザー権限を持つユーザーの名前です。

-password password

指定したユーザーに関連付けられたパスワードです。

更新パケットの場所指定

デフォルト

なし。

-receive [**-storage-class** storage-class]

メモ：このオプションは、同期サーバー上で **syncreplica** を実行する場合にのみ有効です。

現在のホストの記憶ベイをスキャンします。このホストに関連付けられたレプリカ用の未処理の更新パケットは、ホスト上の該当するレプリカに適用されます。

-storage-class を指定すると、**syncreplica** は指定した記憶クラスの記憶ベイのみをスキャンします。

syncreplica は、レプリカ作成パケットを見つけると、保存および転送の管理者にメールを送信します (現在のホストが **Windows** ホストで、電子メールによる通知が有効になっていない場合、メッセージは **Windows** イベント ビューワに表示されます)。
mkreplica を使用してこれらのレプリカ作成パケットをインポートします。

packet-file-pname | staging-area-pname ...

各 **packet-file-pname** を更新パケットとして処理します。指定された各 **staging-area-pname** に対して、以前未処理だったすべての更新パケットをディレクトリで検索し、該当するレプリカに適用します。

例

この例では、読みやすくするために行を途中で改行しています。実際には、各コマンドを一行で入力してください。

エクスポート

- **boston_hub** レプリカで、**sanfran_hub** レプリカの更新パケットを生成します。パケットは **c:¥cqms¥sanfran_hub_sync.xml** に保存します。

```
multiutil syncreplica -export -clan telecom -site boston_hub -family SAMPL -user susan  
-p passwd -out c:¥cqms¥sanfran_hub_sync.xml sanfran_hub
```

Multiutil: パケット ファイル 'c:¥cqms¥sanfran_hub_sync.xml' が生成されました

- 後で発送するために、パケット ファイルを記憶ベイに配置します。

```
multiutil syncreplica -export -clan telecom -site boston_hub -family DEV -user susan -p  
passwd -maxsize 500mb -workdir c:¥work -ship -sclass cq_default sanfran_hub
```

Multiutil: パケット ファイル

'C:¥work¥sync_BOSTON_HUB_26-March-02_10-55-16.xml' が生成されました

multiutil: 発送オーダー

"C:¥temp¥cqms¥ms_ship¥outgoing¥sh_o_sync_BOSTON_HUB_26-March-02_10-55-16.xml" が生成されました。

- 前の例と同様ですが、パケットを直ちに発送します。

```
multiutil syncreplica -export -clan telecom -site boston_hub -family DEV -user susan  
-password p -maxsize 500mb -workdir c:¥work -fship -sclass cq_default sanfran_hub
```

Multiutil: パケット ファイル

'C:¥work¥sync_BOSTON_HUB_26-March-02_10-56-43.xml' が生成されました

multiutil: 発送オーダー

"C:¥cqms¥ms_ship¥outgoing¥sh_o_sync_BOSTON_HUB_26-March-02_10-56-43.xml" が生成されました。

multiutil: 生成されたパケットを転送 / デリバーしようとしています ...

multiutil: -- パケット

C:¥cqms¥ms_ship¥outgoing¥sync_BOSTON_HUB_26-March-02_10 を転送 /
デリバーしました -

---- メモ : NT イベント ログでエラーを確認してください。

インポート

- **cq_storage** 記憶クラスにあるすべての入力更新パッケージをインポートします。

```
multiutil syncreplica -import -clan telecom -site sanfran_hub -family DEV -user jcole -p  
passwd -receive -sclass cq_storage
```

Multiutil: boston_hub からの 4 トランザクションが MASTR データベースに再現されました。

Multiutil: boston_hub からの 2 トランザクションが DEV データベースに再現されました。

Multiutil: パッケージ

C:\temp\cqms\ms_ship\incoming\sync_boston_hub_22-January-02_11-10-34.xml を削除しています

- **sanfran_hub** レプリカで、更新パッケージ **sanfran_hub_sync.xml** を処理します。

```
multiutil syncreplica -import -clan telecom -site sanfran_hub -family DEV -user jcole -p  
passwd c:\cqms\sanfran_hub_sync.xml
```

Multiutil: boston_hub からの 1 トランザクションが MASTR データベースに再現されました。

Multiutil: boston_hub からの 2 トランザクションが DEV データベースに再現されました。

Multiutil: パッケージ c:\cqms\sanfran_hub_sync.xml を削除しています

- **sanfran_hub** レプリカを最新のスキーマバージョンにアップグレードする前に、**sanfran_hub** レプリカで更新パッケージ **sanfran_hub_sync.xml** を処理しようとしています。

```
multiutil syncreplica -import -clan telecom -site sanfran_hub -family DEV -user jcole -p  
passwd c:\cqms\sanfran_hub_sync.xml
```

Multiutil: The UPDATE_PACKET packet sent from boston_hub at 2002-01-22 15:15:50 is destined for schema revision 2, not 1; re-execute syncreplica after site admin has upgraded database.

Multiutil: boston_hub からの 2 トランザクションが MASTR データベースに再現されました。

Multiutil: パッケージ c:\cqms\sanfran_hub_sync.xml を保存しています ...

- 入力記憶ベイ内のすべての更新パッケージを処理します。

```
multiutil syncreplica -import -clan telecom -site boston_hub -family DEV -user susan -p  
passwd -receive
```

Multiutil: SANFRAN_HUB からの 1 トランザクションが MASTR データベースに再現されました。

Multiutil: SANFRAN_HUB からの 2 トランザクションが DEV データベースに再現されました。

Multiutil: パッケージ

C:\temp\cqms\ms_ship\incoming\sync_SANFRAN_HUB_07-February-02_11-24-49.xml を削除しています

関連項目

mkorder、mkreplica、MultiSite Control Panel、shipping.conf

索引

A

ACL
 記憶ベイ 170
activate 112
albd_server、使用するポートの制御 55

C

ccase-home-dir ディレクトリ xx
chepoch 115
chmaster 118
chreplica 123
CLEARCASE_MAX_PORT 環境変数 55
CLEARCASE_MIN_PORT 環境変数 55
control_panel 126
cquest-home-dir ディレクトリ xx

D

describe 128
dumpoplog 131

E

export_sync records、スクラブ 34

F

FTP とファイアウォール 46

L

lsepoch 136
lspacket 139
lsreplica 142

M

mkorder 148
mkreplica 153
MultiSite コントロール パネル 49, 167
MultiSite のコマンド
 multiutil 39
 multiutil 以外 42
 概要 39
multiutil 173
multiutil コマンド
 概要 39
 構文 173
 要約 40

O

oplog (操作ログ)
 エポック数のギャップ 94
 スクラブ 33

R

recoverpacket 174
restorereplica 178
rmreplica 182

S

scruboplog 185
shipping.conf 189
shipping.conf ファイル
 概要 49
shipping_server 196
 エラー処理の仕組み 51
 トラブルシューティングのシナリオ 95
 ファイアウォール上へのインストール 53
 ログ 197
syncreplica 200
syncreplica コマンド
 例 70

い

一方向の同期
 概要 27
 リスク 27
インポート操作
 障害の原因 100
 正常に終了と想定 69
 損失したパケット 99, 101
 同期手順、手動 71
 同期に関する共通の問題 99
 パケットが破損した際の症状 100

え

エクスポート操作
 記憶ベイでのパケットの累積 95
 更新パケットのデリバリー パターン 25
 損失パケットの再送信 174
 同期手順、手動 70
 同期の問題 94
 レプリカの作成 48

エポック数
 ギャップ 94
 更新での役割 14
 変更のコマンド 115
 変更、メソッド 101
エポック数マトリックス
 値が 0 17
 概要 15
 情報の表示 16, 136
エラー メッセージ
 「トラブルシューティング」も参照
 oplog 項目のギャップ 94
 replica はすでに存在する 91
 転送エラーのリスト 95

か

カスタマ サポート xxi
稼働中のスキーマ リポジトリ
 マスターシップの変更 121
環境変数 55
管理
 記憶ベイ用のディスク容量 19
 スクラブ 33
 責任事項のリスト 35
 バックアップの要件 36

き

記憶クラス
 同期での使用 49
 名前の指定 169
記憶ベイ
 「戻りベイ」も参照
 ACL 170
 概要 47
 ディスク容量の要件 19
 パケット 95, 99
 パス 170, 191
規則、表記 xx

け

計画上の課題

- 概要 19
- 設計のドキュメント化 19
- タイムゾーンと同期方針 31
- 同期方針 30
- ファイアウォール 54

権限

- 「マスタースhip」を参照

こ

更新パケット

- 暗号化 55
- 記憶クラス 49
- 混在環境でのエラーの通知 52
- 削除 202
- 手動での作成 70
- 内容 201

更新パケットの暗号化 55

更新パケットのトポロジー 26

コマンド

- activate 112
- chepoch 115
- chmaster 118
- chreplica 123
- control_panel 126
- describe 128
- dumpoplog 131
- lsepoch 136
- lspacket 139
- lsreplica 142
- mkorder 148
- mkreplica 153
- multiutil 173
- recoverpacket 174
- restorereplica 178

- rmreplica 182
- scruboplog 185
- shipping_server 196
- syncreplica 200

さ

サイト

- 概要 3
- 設計のドキュメント化 19

し

受信ハンドラ、パス 171, 193

す

スクラブ 33

そ

双方向の同期

- 概要 27

た

タイムゾーン 31

て

ディスク容量

- 記憶ペイ 19

データベース レプリカの作成 153

電子メールとファイアウォール 45

転送処理

- shipping_server 196
- お勧めする方法 25
- 間接ルート 50
- 共通の問題 95
- デリバリー障害 98
- デリバリーの仕組み 7
- 同期手順、手動 70
- 発送オーダーの有効期限切れ 98
- ファイアウォール 53
- 保存および転送機能 47
- 無効な宛先 97

と

ドキュメント

- ヘルプの説明 xxi

トラブルシューティング

- oplog 項目のギャップ 94
- shipping_server の問題 95
- shipping_server ログ 197
- 誤ったマスターシップの転送 88
- インポート障害 100
- インポートの問題 99
- 概要 91
- 既存レプリカ 91
- 更新パケットのエクスポート 94
- 更新パケットの作成 94
- 受信パケットの累積 99
- 損失したパケット 101
- デリバリー、再試行 51
- デリバリー障害 98
- 転送の問題 95
- 同期処理の正常終了 69
- 同期と oplog のスクラブ 34
- 保存および転送機能のためのパケット サイズ 48
- 無効な宛先 97
- 有効期限の切れた発送オーダー 98

は

パケット

- 「レプリカ作成パケット」、「更新パケット」も参照
- インポート操作 7
- 概要 6
- 再デリバリー 169, 192
- 情報の表示 139
- 保存および転送機能への引き渡し 49
- ルーティング 172, 193
- 論理と物理 6
- 論理パケットを物理パケットに分割 167, 190

バックアップ

- 要件 36

発送オーダー

- 作成 148
- 処理 196
- 有効期限 51, 98
- 有効期限の設定 169, 192

ひ

- 表記規則 xx

ふ

ファイアウォール

- shipping_server 53
- 同期 53

フック

- 同期中の起動 202

へ

ベイ

- 「戻りベイ」、「記憶ベイ」を参照
- ヘルプ、アクセス xxi

ほ

保存および転送機能

- 概要 47
- カスタマイズ 167
- 間接発送ルート 50
- 記憶クラス 49
- 信頼性、パケット サイズ 48
- 設定 189
- 通知メカニズム 191
- デリバリーの試行 51
- パケットの引き渡し 49
- 発送オーダーの作成 148
- ファイアウォール 53
- ファイアウォールとの併用 53
- ファイルの送信 52
- ポート、サーバーに関する制御 55

ま

マスターシップ

- chmaster コマンドの説明 118
- 誤って変更した場合の修正 88
- 概要 8
- 管理 81
- 削除したレプリカ内のオブジェクト 77

も

戻りベイ

- 「記憶ベイ」も参照
- ACL 170
- 概要 47
- パス 170, 191

ゆ

ユーザー データベース

- レプリカの変更による更新 202

れ

レプリカ

- 「レプリカの作成」、「レプリカの同期」も参照
- lsreplica コマンドの説明 142
- oplog のスクラップ 33
- 削除手順 77
- 名前の競合の解決 12
- バックアップ 36
- 変更履歴、追跡方法 13
- ホストまたはホスト名の変更 123

レプリカ オブジェクト

- 削除 182

レプリカ作成パケット

- 内容とクリーンアップ 155
- 分割方法 48

レプリカの同期

- oplog をスクラップするリスク 34
- sync replica 200
- 一方向方式のリスク 27
- 一方向または両方向 27
- エクスポートに関する共通の問題 94
- エボック数の役割 14
- 概要 6, 69
- 間接ルート 16
- 計画上の課題 30
- 手動での手順 70
- 正常に終了と想定 69
- デリバリーの試行 51
- デリバリー パターン 25
- ファイアウォールの取り扱い方法 53

レプリカ ファミリ

- メンバーの表示 142

