

IBM® DB2 Universal Database™



レプリケーションのガイドおよびリファレンス

バージョン 8

IBM® DB2 Universal Database™



レプリケーションのガイドおよびリファレンス

バージョン 8

ご注意!

本書および本書で紹介する製品をご使用になる前に、**特記事項**に記載されている情報をお読みください。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

| | |
|-----|--|
| 原典： | SC27-1121-00 IBM® DB2 Universal Database™ Replication Guide and Reference Version 8 |
| 発行： | 日本アイ・ビー・エム株式会社 |
| 担当： | ナショナル・ランゲージ・サポート |

第1刷 2002.10

この文書では、平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1994, 2002. All rights reserved.

© Copyright IBM Japan 2002

目次

| | |
|-------------------|------|
| 本書について | xi |
| 本書の対象読者 | xi |
| 本書の使用方法 | xi |
| 規則 | xiii |
| 用語 | xiv |
| 構文図の見方 | xiv |
| ロード・マップ | xv |

| | |
|--|-------------|
| DB2 レプリケーション バージョン 8 での 変更点 | xvii |
| ユーザビリティの改善 | xvii |
| パフォーマンスの向上 | xviii |
| 新規のユーザー・インターフェース | xix |
| 新機能 | xx |
| 保守容易性の向上 | xxiv |
| レプリケーション・システム・コマンドの変 更 | xxiv |
| コントロール表の変更 | xxvi |
| サポートされなくなった機能 | xxvii |

第 1 部 レプリケーションのガイド . 1

| | |
|--|----------|
| 第 1 章 レプリケーションの計画 | 3 |
| メモリーの計画 | 3 |
| キャプチャー・プログラムによって使用され るメモリー | 3 |
| アプライ・プログラムによって使用されるメ モリー | 5 |
| レプリケーション・アラート・モニターによ って使用されるメモリー | 5 |
| ストレージの計画 | 5 |
| ログの影響の計画 | 6 |
| ターゲット表およびコントロール表のストレ ージ要件の計画 | 8 |
| 一時ファイルのストレージ要件の計画 | 10 |
| 競合検出の計画 | 12 |
| 非 DB2 リレーショナル・ソースの計画 | 12 |
| キャプチャー・トリガーの場合のトランザ クション・スループット率の計画 | 13 |
| 非 DB2 リレーショナル・ソース・サーバ ーの場合のログの影響の計画 | 13 |

| | |
|---|----|
| Oracle ソース・サーバーの場合のロックの 計画 | 13 |
| 既存のトリガーとキャプチャー・トリガー との共存の計画 | 13 |
| コード・ページ・トランザクションの計画 | 14 |
| 互換コード・ページを持つデータベース間 でのデータのレプリケーション | 14 |
| レプリケーション用の各国語サポート (NLS) の構成 | 15 |

第 2 章 レプリケーションの設定 17

| | |
|---|----|
| レプリケーション・サーバーに対するアクセ スのコントロール | 17 |
| レプリケーションのための接続要件 | 17 |
| レプリケーション用のユーザー ID の許可 | 20 |
| 管理の許可要件 | 20 |
| キャプチャー・プログラムの許可要件 | 22 |
| 非 DB2 リレーショナル・データベースで のキャプチャー・トリガーの場合の許可要 件 | 23 |
| アプライ・プログラムの許可要件 | 24 |
| レプリケーション・アラート・モニターの 場合の許可要件 | 26 |
| レプリケーション用のユーザー ID およびパ スワードの保管 (UNIX、Windows) | 26 |
| レプリケーション・コントロール表のセット アップ | 27 |
| コントロール表の作成 (UNIX、Windows) | 27 |
| コントロール表の作成 (z/OS) | 27 |
| コントロール表の作成 (OS/400) | 28 |
| 非 DB2 リレーショナル・ソース用のコン trol表の作成 | 28 |
| キャプチャー・コントロール表のセットを 複数作成する | 29 |
| レプリケーション・プログラムのセットアッ プ | 30 |
| レプリケーション・プログラムのセットア ップ (UNIX、Windows) | 30 |
| キャプチャー・プログラムおよびアプラ イ・プログラムのセットアップ (OS/400) | 35 |

| | |
|--|----|
| レプリケーション・プログラムのセットアップ (z/OS) | 36 |
| ジャーナルのセットアップ (OS/400) | 37 |
| ソース表用ジャーナルの作成 (OS/400) | 37 |
| ジャーナルおよびジャーナル・レシーバーの管理 (OS/400). | 39 |

第 3 章 表およびビューをレプリケーション・ソースとして登録する 43

| | |
|--|----|
| DB2 表をソースとして登録する | 43 |
| 非 DB2 リレーショナル表をソースとして登録する. | 46 |
| ソース表用の登録オプション | 47 |
| 列のサブセットの登録 (垂直方向のサブセット化). | 48 |
| フル・リフレッシュ・コピーと変更キャプチャー・レプリケーション | 48 |
| 変更後イメージ列と変更前イメージ列 | 50 |
| 変更前イメージ接頭部. | 53 |
| エラー発生時におけるキャプチャー・プログラムの停止. | 54 |
| キャプチャー・プログラムが更新を保存する方法. | 54 |
| 変更のリキャプチャーの防止 (update-anywhere レプリケーション) | 55 |
| 競合検出の設定 (update-anywhere レプリケーション) | 60 |
| リモート・ジャーナリングを使用する表の登録 (OS/400) | 62 |
| 主キーの代わりに相対レコード番号 (RRN) を使用する (OS/400) | 63 |
| レプリケーション・ソースとしてのビューの動作. | 64 |
| 単一の表に対するビュー. | 64 |
| 複数の表の結合に対するビュー. | 64 |
| 表のビューをソースとして登録する | 66 |
| CCD 表をソースとして保守する (IMS) | 67 |

第 4 章 ソースのサブスクライブ. 69

| | |
|---------------------------------------|----|
| ソースおよびターゲットの分類方法の計画 | 69 |
| サブスクリプション・セット・メンバー数の計画 | 70 |
| アプライ修飾子ごとのサブスクリプション・セット数の計画 | 71 |
| サブスクリプション・セットの作成 | 72 |

| | |
|--|-----|
| サブスクリプション・セットのオプション処理 | 74 |
| セットがアクティブかどうかの指定 | 75 |
| アプライ・プログラムが取り出すデータに相当する分数の指定 | 75 |
| アプライ・プログラムが参照保全性のあるターゲット表をロードする方法の決定 | 77 |
| アプライ・プログラムがセットのメンバーへの変更を複製する方法の指定. | 78 |
| サブスクリプション・セット用の SQL ステートメントまたはストアド・プロシージャの定義. | 80 |
| サブスクリプション・セットのレプリケーションのスケジューリング | 80 |
| サブスクリプション・セット内におけるソース表およびビューのターゲット表およびビューへのマッピング | 82 |
| ターゲット・タイプの選択 | 85 |
| 読み取り専用ターゲット表の定義 | 87 |
| multi-tier 構成内の中間層の定義 | 92 |
| 読み取り / 書き込みターゲットの定義 (update-anywhere) | 94 |
| 既存の表をターゲット表として使用する. | 97 |
| すべてのターゲット表タイプに共通のプロパティ. | 98 |
| ターゲットに適用するソース列. | 98 |
| ターゲットに適用するソース行. | 98 |
| ソース列からターゲット列にマップする方法 | 100 |
| ターゲット・キー | 100 |
| アプライ・プログラムがターゲット・キー変更オプションを使用してターゲット・キー列を更新する方法 | 102 |

第 5 章 特殊なデータ・タイプのレプリケーション 105

| | |
|--|-----|
| レプリケーションにおける一般的なデータの制約事項. | 105 |
| ラージ・オブジェクトのレプリケーション | 106 |
| DATALINK 値のレプリケーション | 107 |
| ASNDCOPY 出口ルーチンの設定と使用方法 | 109 |
| DLFM_ASNCOPYD の設定と使用方法 (UNIX、Windows). | 111 |
| ASNDCOPYD の設定と使用方法 (OS/400) | 113 |

| | | | |
|--|------------|---|------------|
| 第 6 章 レプリケーション環境におけるデータのサブセット化 | 117 | trace_limit (UNIX, Windows, z/OS) | 144 |
| 登録時におけるデータのサブセット化 | 117 | キャプチャー・プログラムの始動 (OS/400) | 145 |
| ビューを使用したソース・データのサブセット化 | 118 | 実行中のキャプチャー・プログラムの動作の変更 | 146 |
| 特定の行のキャプチャーを防止するために CD 表にトリガーを定義する (UNIX, Windows, z/OS) | 118 | キャプチャー・パラメーター表の稼働パラメーターの変更 | 147 |
| サブスクリプションの際のデータのサブセット化 | 119 | キャプチャー・プログラムの停止 | 148 |
| 第 7 章 レプリケーション環境におけるデータ操作 | 123 | キャプチャーの延期 (UNIX, Windows, z/OS) | 149 |
| ストアド・プロシージャまたは SQL ステートメントを使用したデータ拡張 | 124 | キャプチャーの再開 (UNIX, Windows, z/OS) | 150 |
| 名前が異なるソース列とターゲット列のマッピング | 125 | キャプチャーの再初期化 | 150 |
| 算出列の作成 | 125 | 第 10 章 アプライ・プログラムの操作 | 153 |
| 第 8 章 レプリケーション SQL スクリプトのカスタマイズおよび実行 | 127 | アプライ・プログラムの始動 (UNIX, Windows, z/OS) | 153 |
| 第 9 章 キャプチャー・プログラムの操作 129 | | apply_path (UNIX, Windows, z/OS) | 155 |
| キャプチャー・プログラムのデフォルトの稼働パラメーター | 129 | apply_qual (UNIX, Windows, z/OS) | 156 |
| キャプチャー・プログラムの稼働パラメーターの変更 | 131 | control_server (UNIX, Windows, z/OS) | 156 |
| キャプチャー・プログラムの始動 (UNIX, Windows, z/OS) | 133 | copyonce (UNIX, Windows, z/OS) | 156 |
| autoprune (UNIX, Windows, z/OS) | 135 | db2_subsystem (z/OS) | 157 |
| autostop (UNIX, Windows, z/OS) | 136 | delay (UNIX, Windows, z/OS) | 157 |
| capture_path (UNIX, Windows, z/OS) | 136 | errwait (UNIX, Windows, z/OS) | 158 |
| capture_schema (UNIX, Windows, z/OS) | 137 | inamsg (UNIX, Windows, z/OS) | 158 |
| capture_server (UNIX, Windows, z/OS) | 138 | loadxit (UNIX, Windows, z/OS) | 158 |
| commit_interval (UNIX, Windows, z/OS) | 138 | logreuse (UNIX, Windows, z/OS) | 159 |
| lag_limit (UNIX, Windows, z/OS) | 138 | logstdout (UNIX, Windows, z/OS) | 159 |
| logreuse (UNIX, Windows, z/OS) | 139 | notify (UNIX, Windows, z/OS) | 159 |
| logstdout (UNIX, Windows, z/OS) | 139 | opt4one (UNIX, Windows, z/OS) | 160 |
| memory_limit (UNIX, Windows, z/OS) | 140 | pwdfile (UNIX, Windows) | 160 |
| monitor_interval (UNIX, Windows, z/OS) | 140 | sleep (UNIX, Windows, z/OS) | 160 |
| monitor_limit (UNIX, Windows, z/OS) | 140 | spillfile (UNIX, Windows, z/OS) | 161 |
| prune_interval (UNIX, Windows, z/OS) | 141 | sqlerrorcontinue (UNIX, Windows) | 162 |
| retention_limit (UNIX, Windows, z/OS) | 142 | term (UNIX, Windows, z/OS) | 163 |
| sleep_interval (UNIX, Windows, z/OS) | 142 | trlreuse (UNIX, Windows, z/OS) | 163 |
| startmode (UNIX, Windows, z/OS) | 143 | アプライ・プログラムの始動 (OS/400) | 164 |
| term (UNIX, Windows, z/OS) | 144 | アプライ・プログラムの停止 | 165 |
| | | ASNDONE 出口ルーチンの変更 (UNIX, Windows, z/OS) | 166 |
| | | ASNDONE 出口ルーチンの変更 (OS/400) | 168 |
| | | ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ | 169 |
| | | ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (UNIX, Windows) | 169 |
| | | ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (z/OS) | 171 |

| | | | |
|--|-----|---|-----|
| ASNLOAD 出口の動作のカスタマイズ (UNIX、Windows、z/OS) | 173 | 登録の再活性化 | 205 |
| ASNLOAD 出口ルーチンを使ったターゲット ット表のリフレッシュ (OS/400) | 175 | 登録の除去 | 207 |
| 第 11 章 レプリケーションのモニター . . . 177 | | キャプチャー・スキーマの変更 | 207 |
| レプリケーション・プログラムの現在の状況 のチェック (UNIX、Windows、z/OS) | 177 | 新規サブスクリプション・セットの作成 | 211 |
| キャプチャー・プログラムおよびアプライ・ プログラムのジャーナル・ジョブの状況のチ ェック (OS/400) | 179 | 既存のサブスクリプション・セットへの新規 サブスクリプション・セット・メンバーの追 加 (UNIX、Windows、z/OS) | 212 |
| 傾向のヒストリー・データの検討 | 179 | サブスクリプション・セットの属性の変更 | 215 |
| キャプチャー・プログラムのメッセージの 検討 | 181 | サブスクリプション・セット名の変更 | 215 |
| キャプチャー・プログラムのスループット の検査 | 181 | サブスクリプション・セットの分割 | 218 |
| キャプチャー・プログラムによって処理さ れるデータの待ち時間の表示 | 182 | サブスクリプション・セットのマージ | 223 |
| アプライ・プログラムのメッセージの検討 | 182 | サブスクリプション・セットのアプライ修飾 子の変更 | 226 |
| アプライ・プログラムのスループットの検 査 | 183 | サブスクリプション・セットの非活動化 | 229 |
| トランザクションのレプリケーションに要 した平均時間の表示 | 183 | サブスクリプション・セットの除去 | 231 |
| レプリケーション環境の自動モニターの設定 | 184 | データベース・アプリケーション・イベント とレプリケーション・イベントの調整 | 232 |
| モニター・コントロール表の作成 | 185 | USER タイプ・シグナルを使用したイベン ト END_SYNCHPOINT の設定 | 232 |
| レプリケーション・アラート・モニターの 連絡先情報の定義 | 186 | キャプチャー CMD STOP シグナルの使 用方法 | 233 |
| レプリケーション・アラート・モニターの アラート条件の選択 | 187 | アプライ・プログラム外部の CAPSTART ハンドシェイク・シグナルの実行 | 236 |
| レプリケーション・アラート・モニターの 開始 | 189 | CAPSTOP シグナルの実行 | 238 |
| レプリケーション・アラート・モニターの 開始のスケジューリング | 195 | 別のシステムへのレプリケーション構成のプ ロモート | 239 |
| レプリケーション・アラート・モニターの 再初期化 | 195 | 第 13 章 レプリケーション環境の保守 . . . 241 | |
| レプリケーション・アラート・モニターの 停止 | 196 | ソース・システムの保守 | 241 |
| キャプチャー・プログラムの進行のモニター (OS/400) | 196 | ソース・オブジェクトの保守 | 241 |
| 第 12 章 レプリケーション環境の変更 . . . 199 | | ソース・ログおよびジャーナル・レシーバ ーの保守および保存 | 242 |
| 新規オブジェクトの登録 | 199 | コントロール表の保守 | 247 |
| 登録済みオブジェクトの登録属性の変更 | 200 | RUNSTATS ユーティリティの使用 方法 (UNIX、Windows、z/OS) | 248 |
| ソース表への列の追加 | 201 | パッケージおよびプランの再バインド (UNIX、Windows、z/OS) | 248 |
| 登録済みオブジェクトの変更のキャプチャー の停止 | 204 | コントロール表の再編成 | 248 |
| | | コントロール表のプルーニング | 250 |
| | | レプリケーションの失敗の防止およびエラ ーからのリカバー | 254 |
| | | ターゲット表の保守 | 256 |
| | | 第 2 部 レプリケーション・センタ ー 257 | |

| | |
|--|------------|
| 第 14 章 DB2 レプリケーション・センターの使用 | 259 |
| DB2 レプリケーション・センターの前提条件 | 261 |
| DB2 レプリケーション・センターの始動 | 261 |
| レプリケーション・センター・ランチパッドの使用 | 263 |
| レプリケーション・センターのユーザー ID およびパスワードの管理 | 264 |
| レプリケーション・プロファイルの作成 | 265 |
| コントロール表プロファイルの作成 | 265 |
| ソース・オブジェクト・プロファイルの作成 | 266 |
| ターゲット・オブジェクト・プロファイルの作成 | 267 |
| レプリケーション・コントロール表の作成 | 268 |
| キャプチャー・コントロール表の作成 | 268 |
| アプライ・コントロール表の作成 | 270 |
| モニター・コントロール表の作成 | 270 |
| レプリケーション・センターへのサーバーの追加 | 271 |
| データベースの変更キャプチャーの使用可能化 (UNIX および Windows) | 272 |
| ソースの登録 | 273 |
| サブスクリプション・セットの作成 | 275 |
| サブスクリプション・セットの情報の定義 | 276 |
| ソースからターゲットへのマッピング | 276 |
| サブスクリプション・セットのスケジューリング | 278 |
| サブスクリプション・セットへの SQL ステートメントまたはストアド・プロシージャの追加 | 278 |
| サブスクリプション・セットの活動化または非活動化 | 279 |
| レプリケーション・オブジェクトのプロモート | 279 |
| 登録済みの表またはビューのプロモート | 280 |
| サブスクリプション・セットのプロモート | 280 |
| ターゲット表の強制フル・リフレッシュ | 281 |
| レプリケーション定義の除去または削除 | 283 |
| キャプチャー・プログラムの操作 | 283 |
| アプライ・プログラムの操作 | 284 |
| レプリケーション・アラート・モニターの操作 | 285 |
| 第 15 章 基本的なデータ・レプリケーションのシナリオ: DB2 for Windows | 287 |

| | |
|---|-----|
| 開始する前に | 287 |
| このシナリオの計画 | 288 |
| レプリケーション・ソース | 288 |
| レプリケーション・ターゲット | 289 |
| レプリケーション・オプション | 289 |
| このシナリオ用のレプリケーション環境のセットアップ | 290 |
| ステップ 1: キャプチャー・プログラムのレプリケーション・コントロール表を作成する | 290 |
| ステップ 2: ソース・データベースのレプリケーションを使用可能にする | 291 |
| ステップ 3: レプリケーション・ソースを登録する | 292 |
| ステップ 4: アプライ・プログラムのレプリケーション・コントロール表を作成する | 295 |
| ステップ 5: サブスクリプション・セットとサブスクリプション・セット・メンバーを作成する | 295 |
| ステップ 6: アプライ・パスワード・ファイルを作成する | 302 |
| ステップ 7: シナリオのデータを複製する | 303 |
| レプリケーション環境における操作 | 305 |
| ステップ 1: ソース表を更新する | 305 |
| ステップ 2: キャプチャー・プログラムの状況を表示する | 306 |
| ステップ 3: アプライ・プログラムの状況を表示する | 307 |
| ステップ 4: キャプチャー・プログラムとアプライ・プログラムを停止する | 309 |
| レプリケーションのモニター | 310 |
| ステップ 1: モニター・プログラムのレプリケーション・コントロール表を作成する | 310 |
| ステップ 2: レプリケーション・アラートの連絡先を作成する | 312 |
| ステップ 3: キャプチャー・プログラムのアラート条件を選択する | 312 |
| ステップ 4: アプライ・プログラムのアラート条件を選択する | 313 |
| ステップ 5: モニター修飾子のレプリケーション・アラート・モニターを開始する | 315 |

第 3 部 レプリケーションのリファレンス **319**

**第 16 章 レプリケーション・オブジェクト
の命名規則 321**

**第 17 章 レプリケーション用のシステム・
コマンド (UNIX、Windows、z/OS) . . . 323**

asnacmd: アプライの操作
(UNIX、Windows、z/OS) 324

asnanalyze: アナライザーの操作 (UNIX およ
び Windows) 325

asnapply: アプライの開始
(UNIX、Windows、z/OS) 328

asnccap: キャプチャーの開始
(UNIX、Windows、z/OS) 334

asnccmd: キャプチャーの操作
(UNIX、Windows、z/OS) 340

asnmcmd: レプリケーション・アラート・モ
ニターの操作 (UNIX、Windows、z/OS). . . 346

asnmon: レプリケーション・アラート・モニ
ターの開始 (UNIX、Windows、z/OS) . . . 348

asnpwd: パスワード・ファイルの保守 (UNIX
および Windows) 352

asnscrt: キャプチャー、アプライ、またはレ
プリケーション・アラート・モニターを開始
する DB2 レプリケーション・サービスの作
成 (Windows のみ) 355

asnsdrop: DB2 レプリケーション・サービ
スのドロップ (Windows のみ) 357

asntrc: レプリケーション・トレース機能の操
作 (UNIX、Windows、z/OS) 358

**第 18 章 レプリケーション用のシステム・
コマンド (OS/400) 365**

ADDDPRREG: DPR 登録の追加 (OS/400) . . . 366

ADDDPRSUB: DPR サブスクリプション・セ
ットの追加 (OS/400) 374

ADDDPRSUBM: DPR サブスクリプション・
セットのメンバーの追加 (OS/400) 390

ANZDPR: アナライザーの操作 (OS/400) . . 400

CHGDPRCAPA: DPR キャプチャー属性の変
更 (OS/400) 404

CRTDPRtbl: レプリケーション・コントロ
ール表の作成 (OS/400) 408

ENDDPRAPY: アプライ・プログラムの停止
(OS/400) 409

ENDDPRCAP: キャプチャー・プログラムの
停止 (OS/400) 413

GRTDPRAUT: ユーザーの許可 (OS/400) . . . 415

INZDPRCAP: DPR キャプチャー・プログラ
ムの再初期化 (OS/400) 424

OVDRPRCAPA: DPR キャプチャー属性のオ
ーバライド (OS/400) 426

RMVDPRREG: DPR 登録の除去 (OS/400) . . 431

RMVDPRSUB: DPR サブスクリプション・セ
ットの除去 (OS/400) 432

RMVDPRSUBM: DPR サブスクリプション・
セット・メンバーの除去 (OS/400) 434

RVKDPRAUT: 権限の取り消し (OS/400) . . 436

STRDPRAPY: アプライ・プログラムの始動
(OS/400) 438

STRDPRCAP: キャプチャー・プログラムの
始動 (OS/400) 446

WRKDPTRRC: DPR トレース機能の使用方
法 (OS/400) 454

**第 19 章 レプリケーション・プログラムの
操作 (z/OS) 461**

レプリケーション・プログラムの操作に JCL
またはシステム開始タスクを使用する方法
(z/OS) 461

JCL を使用したレプリケーション・プロ
グラムの操作 461

システム開始タスクを使用してレプリケ
ーション・プログラムを操作する方法 . . . 463

レプリケーション・プログラムを自動的に再
始動するために MVS 自動リスタート・マネ
ージャー (ARM) を使用する方法 (z/OS) . . 463

データ共有モードへのレプリケーション環
境のマイグレーション (z/OS) 464

**第 20 章 Windows Service Control
Manager を使用してシステム・コマンドを
発行する方法 (Windows) 467**

レプリケーション・サービスの作成 467

レプリケーション・サービスの操作 469

レプリケーション・サービスのドロップ . . 469

**第 21 章 各種オペレーティング・システム
でのレプリケーション・プログラムのスケジ
ューリング 471**

UNIX オペレーティング・システムでのプロ
グラムのスケジューリング 471

| | |
|---|-----|
| Windows オペレーティング・システムでのプログラム のスケジューリング | 471 |
| z/OS オペレーティング・システムでのプログラム のスケジューリング | 472 |
| OS/400 オペレーティング・システムでのプログラム のスケジューリング | 472 |

**第 22 章 DB2 レプリケーション・コンポー
ネントの通信方法 473**

| | |
|---|-----|
| レプリケーション・センター、キャプチャー・プログラム またはトリガー、およびアプライ・プログラム | 473 |
| キャプチャー・プログラムおよびアプライ・プログラム | 475 |
| キャプチャー・トリガーおよびアプライ・プログラム | 476 |
| レプリケーション・センターおよびレプリケーション・アラート・モニター | 477 |
| レプリケーション・アラート・モニター、キャプチャー・プログラム、およびアプライ・プログラム | 478 |

第 23 章 表構造 481

| | |
|--|-----|
| 表の概観 | 482 |
| キャプチャー・コントロール・サーバーで使用される表のリスト | 486 |
| アプライ・コントロール・サーバーで使用される表のリスト | 489 |
| モニター・コントロール・サーバーで使用される表のリスト | 490 |
| ターゲット・サーバーで使用される表のリスト | 492 |
| キャプチャー・コントロール・サーバー上の表、およびその列に関する記述 | 493 |
| ASN.IBMSNAP_CAPSCHEMAS | 493 |
| schema.IBMSNAP_AUTHTKN (OS/400) | 493 |
| schema.IBMSNAP_CAPENQ | 494 |
| (UNIX、Windows、z/OS) | 494 |
| schema.IBMSNAP_CAPMON | 495 |
| schema.IBMSNAP_CAPPARMS | 496 |
| schema.IBMSNAP_CAPTRACE (DB2 のみ) | 498 |
| schema.CCD_table (DB2 以外) | 499 |
| schema.CD_table | 500 |
| schema.IBMSNAP_PRUNCNTL | 501 |
| schema.IBMSNAP_PRUNE_LOCK | 503 |
| schema.IBMSNAP_PRUNE_SET | 503 |

| | |
|---|-----|
| schema.IBMSNAP_REG_EXT (OS/400) | 504 |
| schema.IBMSNAP_REGISTER | 505 |
| schema.IBMSNAP_REG_SYNCH (DB2 以外のリレーショナル) | 510 |
| schema.IBMSNAP_RESTART | 511 |
| schema.IBMSNAP_SEQTABLE (Informix) | 512 |
| schema.IBMSNAP_SIGNAL | 513 |
| schema.IBMSNAP_UOW | 515 |
| アプライ・コントロール・サーバー上の表、およびその列に関する記述 | 517 |
| ASN.IBMSNAP_APPENQ | 517 |
| ASN.IBMSNAP_APPLY_JOB (OS/400) | 518 |
| ASN.IBMSNAP_APPLYTRACE | 518 |
| ASN.IBMSNAP_APPLYTRAIL | 519 |
| ASN.IBMSNAP_SUBS_COLS | 523 |
| ASN.IBMSNAP_SUBS_EVENT | 524 |
| ASN.IBMSNAP_SUBS_MEMBR | 525 |
| ASN.IBMSNAP_SUBS_SET | 528 |
| ASN.IBMSNAP_SUBS_STMTS | 532 |
| モニター・コントロール・サーバー上の表、およびその列に関する記述 | 534 |
| ASN.IBMSNAP_ALERTS | 534 |
| ASN.IBMSNAP_CONDITIONS | 535 |
| ASN.IBMSNAP_CONTACTGRP | 538 |
| ASN.IBMSNAP_CONTACTS | 538 |
| ASN.IBMSNAP_GROUPS | 539 |
| ASN.IBMSNAP_MONENQ | 539 |
| ASN.IBMSNAP_MONSERVERS | 539 |
| ASN.IBMSNAP_MONTRACE | 540 |
| ASN.IBMSNAP_MONTRAIL | 541 |
| ターゲット・サーバー上の表、およびその列に関する記述 | 542 |
| 基礎集約表 | 542 |
| 変更集約表 | 542 |
| 整合変更データ (CCD) 表 | 543 |
| ポイント・イン・タイム表 | 545 |
| レプリカ表 | 546 |
| ユーザー・コピー表 | 547 |

第 24 章 レプリケーション・メッセージ 549

| | |
|---|-----|
| 付録 A. z/OS におけるユニコードおよび ASCII のコード化スキーム | 659 |
| コード化スキームの選択 | 659 |
| コード化スキームの設定 | 660 |

| | |
|--|-----|
| 付録 B. キャプチャー・プログラムがジャーナル・エントリー・タイプを処理する方法 (iSeries) | 661 |
| 付録 C. アプリケーション (UNIX、Windows) 内部からのレプリケーション・プログラムの始動 | 663 |
| 用語集 | 665 |

| | |
|----------------------|-----|
| 索引 | 679 |
| 特記事項 | 701 |
| 商標 | 704 |
| IBM と連絡をとる | 707 |
| 製品情報 | 707 |

本書について

本書は、DB2® データ・レプリケーション環境を計画、セットアップ、および保守する方法について説明します。

DB2 DataPropagator™ 製品に焦点を当てて述べられています。レプリケーション環境をユーザーのビジネス・ニーズに合うように調整するため、IBM® レプリケーション・ソリューションでは別の製品と一緒に使用することもできます。

本書の対象読者

本書は、データベース管理者、LAN 管理者、およびデータ・レプリケーション環境のセットアップおよび保守担当者を対象にしています。読者は、標準データベース用語を知り、オペレーティング・システム (レプリケーションも含むようになる) の作業知識を持ち、データベース設計、データベース管理、データベース・セキュリティー、サーバー接続、およびネットワークングでの経験を持つ必要があります。また、ユーザー環境のアプリケーションについて理解し、そのアプリケーションが複製したいデータをどのように操作するのかを知っている必要があります。さらに、レプリケーションの概念とコンポーネントについても熟知している必要があります。

本書の使用方法

本書のほとんどのセクションは、すべてのプラットフォームでのレプリケーション関数に関するものです。中には、プラットフォーム固有の情報を含むセクションもあります。

本書の編成と内容は、最終リリース以降、変更されています。以下の 3 部構成になっています。

- 第 1 部: *レプリケーションのガイド* では、ユーザーのレプリケーション環境をプランニング、セットアップ、実行、および保守する方法について説明します。以下の章があります。
 - 3 ページの『第 1 章 レプリケーションの計画』では、レプリケーション環境を計画および設計する方法について説明します。
 - 17 ページの『第 2 章 レプリケーションの設定』では、レプリケーション用に環境を設定する方法について説明します。
 - 43 ページの『第 3 章 表およびビューをレプリケーション・ソースとして登録する』では、レプリケーション・ソースを登録するために必要な情報について説明します。

- 69 ページの『第 4 章 ソースのサブスクライブ』では、サブスクリプション・セットの作成方法と、メンバーをサブスクリプション・セットへ追加する方法について説明します。
- 105 ページの『第 5 章 特殊なデータ・タイプのレプリケーション』では、ソース表の LOB 値および DATALINK 値のためのレプリケーション・オプションについて説明します。
- 117 ページの『第 6 章 レプリケーション環境におけるデータのサブセット化』では、どのようにデータがターゲットへアプライされるのかを説明する他、ターゲットへキャプチャーおよびアプライされるデータをカスタマイズする方法についても説明します。
- 123 ページの『第 7 章 レプリケーション環境におけるデータ操作』では、ソース・データを操作するための、キャプチャー・プログラムまたはアプライ・プログラムの使用方法について説明します。
- 127 ページの『第 8 章 レプリケーション SQL スクリプトのカスタマイズおよび実行』では、レプリケーション環境で SQL を実行する方法について説明します。
- 129 ページの『第 9 章 キャプチャー・プログラムの操作』では、すべてのオペレーティング・システム環境について、キャプチャー・プログラムを操作する方法を説明します。
- 153 ページの『第 10 章 アプライ・プログラムの操作』では、すべてのオペレーティング・システム環境について、アプライ・プログラムを操作する方法を説明します。
- 177 ページの『第 11 章 レプリケーションのモニター』では、レプリケーション環境をモニターする方法について説明します。
- 199 ページの『第 12 章 レプリケーション環境の変更』では、レプリケーション環境で日常の変更を行なう方法について説明します。
- 241 ページの『第 13 章 レプリケーション環境の保守』では、ソース・システム、コントロール表、およびターゲット表の保守方法について説明します。
- 第 2 部: レプリケーション・センターでは、レプリケーションのためのグラフィカル・ユーザー・インターフェースについて説明します。以下の章があります。
 - 259 ページの『第 14 章 DB2 レプリケーション・センターの使用』では、レプリケーション・センターについて説明します。
 - 287 ページの『第 15 章 基本的なデータ・レプリケーションのシナリオ: DB2 for Windows』では、レプリケーション・センターを使用して、サンプル・データに簡単なレプリケーション・シナリオを実行する方法について説明します。
- 第 3 部: レプリケーションのリファレンスでは、レプリケーション・コマンド、レプリケーション表構造、およびレプリケーション・メッセージについて説明します。以下の章があります。

- 321 ページの『第 16 章 レプリケーション・オブジェクトの命名規則』では、レプリケーション・オブジェクトに有効な名前の指定方法について説明します。
 - 323 ページの『第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)』では、DB2 レプリケーションの経験のあるユーザーが、UNIX、Windows、および z/OS オペレーティング・システムでレプリケーション操作をするために、レプリケーション・センターの代わりに使用することができるコマンドについて説明します。
 - 365 ページの『第 18 章 レプリケーション用のシステム・コマンド (OS/400)』では、OS/400 オペレーティング・システムで、ローカルに、レプリケーションをセットアップ、管理、および保守するためのコマンドについて説明します。
 - 461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』では、z/OS で JCL やシステム起動タスクを使用して、レプリケーション・プログラムを開始および操作する方法について説明します。
 - 467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』では、Windows NT オペレーティング・システムで、レプリケーション・プログラムを操作するためのサービスを作成する方法について説明します。
 - 471 ページの『第 21 章 各種オペレーティング・システムでのレプリケーション・プログラムのスケジューリング』では、さまざまなオペレーティング・システムで、キャプチャー、アプライ、およびレプリケーション・アラート・モニター・プログラムをスケジューリングする方法について説明します。
 - 473 ページの『第 22 章 DB2 レプリケーション・コンポーネントの通信方法』では、レプリケーション・コンポーネントが相互通信のためにコントロール表を使用する方法について説明します。
 - 481 ページの『第 23 章 表構造』では、さまざまなレプリケーション・サーバー上にあるレプリケーション表の構造について説明します。
 - 549 ページの『第 24 章 レプリケーション・メッセージ』では、UNIX、Windows、および z/OS でのレプリケーションで出されるメッセージについて説明します。
- 付録には、役に立つ補足情報があります。

規則

本書では、以下の強調表示規則を使用しています。

- **太字体** は、フィールド名、フォルダー名、アイコン、またはメニュー選択などの、ユーザー・インターフェース・コントロールやコマンドを表します。
- **モノスペース** は、示された通りに正確に入力するテキストの例を表します。
- **イタリック** は、値に置き換える変数を表します。また、本の表題や語の強調にも使用されます。

用語

本書では、データベース、接続、コピー、SQL、および LAN の概念における標準用語を使用しています。本書で使用するレプリケーション概念はすべて、用語集に定義されています。

特に指定しない限り、以下を意味するものとします。

UNIX UNIX とは、DB2 Universal Database for all UNIX platforms (UNIX、HP UX、AIX など)、および DB2 Universal Database for Linux を指します。

Windows

Windows とは、DB2 Universal Database for Windows を指します。

OS/400 OS/400 とは、DB2 DataPropagator for iSeries を指します。

z/OS z/OS とは、DB2 Universal Database for z/OS and OS/390 を指します。z/OS は、OS/390 の次世代オペレーティング・システムで、z/OS 上の UNIX System Services (USS) も含まれます。

iSeries iSeries とは、iSeries と iSeries サーバーの両方を指します。iSeries は、AS/400 の次世代サーバーです。OS/400 オペレーティング・システムは、iSeries および iSeries サーバーの両方で稼働します。

たとえば、*アプライ・プログラムの始動 (UNIX、Windows、z/OS)* というセクションでは、DB2 Universal Database for Linux and for all UNIX platforms、DB2 Universal Database for Windows、または DB2 Universal Database for z/OS and OS/390 からアプライ・プログラムを始動する方法について説明しています。また、*アプライ・プログラムの始動 (OS/400)* というセクションでは、DB2 DataPropagator for iSeries を使用する際のアプライ・プログラムの始動方法について説明しています。

構文図の見方

本書で使用される構文図には、以下の規則が適用されます。

- 構文図は、左から右へ、上から下へ、線のパスに沿って読んでいきます。
 - 記号 `>>---` は、ステートメントの先頭を示します。
 - 記号 `--->` は、ステートメント構文が次の行へ続くことを示します。
 - 記号 `>---` は、ステートメントが前の行から続いていることを示します。
 - 記号 `---<` は、ステートメントの終わりを示します。
- 完全なステートメントでない構文ユニット図は、記号 `>---` で始まり、記号 `--->` で終わります。
- キーワード、その同義語、および予約パラメーターは、オペレーティング・システムによって異なりますが、英大文字または小文字のいずれかで示されます。これらの項

| 目的 | 参照先 |
|---|--|
| 製品の最終変更についての情報を得る | CD-ROM 上のインストール・ノート、または製品と一緒にインストールされたリリース・ノート。 DB2 DataPropagator ライブラリーについては、 www.ibm.com/software/data/dpropr/library.html |
| 技術支援リソースおよび顧客支援オプションについて知る | www.ibm.com/software/data/dpropr/support.html |
| IBM 研修サービスのクラスの空き状況を調べる | www.ibm.com/services/learning/ |
| DB2 データ・レプリケーションの現リリースに関する新規情報を得る | xvii ページの『DB2 レプリケーション バージョン 8 での変更点』 |
| DB2 DataPropagator を以前のバージョンからバージョン 8 へマイグレーションする | DB2 DataPropagator ライブラリーについては、 www.ibm.com/software/data/dpropr/library.html |
| レプリケーション・アナライザーのコマンド行ツールの使用方法について学び、ユーザーのレプリケーション環境についての HTML レポートを作成する | DB2 DataPropagator ライブラリーについては、 www.ibm.com/software/data/dpropr/library.html |
| お客様の事例を読む | DataPropagator の事例紹介ページは、Web サイト www.ibm.com/software/data/dpropr/casestudy.html |
| レプリケーション・センターのウィンドウについて理解する | レプリケーション・センターのヘルプ |
| エラー・メッセージをデバッグする | OS/400 では、エラー・メッセージを受け取ったら <F1> をタイプする。 UNIX、Windows、および z/OS では、549 ページの『第 24 章 レプリケーション・メッセージ』を参照。 |
| その他の DB2 情報を得る | 製品の Web ページを www.ibm.com/software/data/ で探す。 |

DB2 レプリケーション バージョン 8 での変更点

このセクションでは、DB2 レプリケーション バージョン 7 以降の主な変更を要約しています。これらの変更には、ユーザビリティの改善、パフォーマンスの向上、新機能、保守容易性の向上、レプリケーション・システム・コマンドの変更、コントロール表の変更、サポートされなくなった機能などがあります。それらについて、これから詳しく述べていきます。

ユーザビリティの改善

キャプチャー・プログラムとアプライ・プログラム間でのハンドシェーク・メカニズムが拡張された: ハンドシェークとは、レプリケーション・ソース用データのキャプチャーを開始することを、キャプチャー・プログラムへ知らせるために、アプライ・プログラムが使用するメカニズムです。バージョン 8 では、このメカニズムが変更、拡張されました。アプライ・プログラムは、新規のシグナル (IBMSNAP_SIGNAL) 表にシグナルを挿入し、キャプチャー・プログラムがソース用データのキャプチャーを開始する必要がある場合をコントロールします。

キャプチャー・プログラムとアプライ・プログラムは、どの順にでも開始できる: バージョン 8 では、キャプチャー・プログラムをアプライ・プログラムの開始後に開始することも、逆にアプライ・プログラムをキャプチャー・プログラム開始後に開始することもできます。バージョン 7 では、キャプチャー・プログラムはアプライ・プログラムの開始前に開始する必要がありました。

キャプチャー・プログラムの実行中に、登録やサブスクリプション・セットを追加できる: キャプチャー・プログラムを再初期設定、停止、または再始動しなくても、新規のレプリケーション・ソースの登録、既存の登録の更新、新規のサブスクリプション・セットの追加、または既存のサブスクリプション・セットの更新が可能です。

各登録ごとにキャプチャーするものをコントロールできる: レプリケーションのために表を登録する際、キャプチャー・プログラムに行の変更をキャプチャーさせるのは、表のどの列が変更されても必ず行なうのか、または登録された列が変更される時のみ行なうかを指定することができます。前バージョンでは、キャプチャー・プログラムで始動パラメーターを使用してキャプチャーしたものがコントロール対象で、つまり、すべての表は同一に扱われていました。バージョン 8 では、各登録ごとにキャプチャーするものをコントロールできるため、始動パラメーターは使用できません。

レプリカからのデータの再キャプチャーをコントロールできる: ソースを登録する際、いくつかの表から再キャプチャーされた変更のみを必要とし、それ以外は必要としないかどうかを指定することができます。デフォルトは以下のとおりです。

- 変更はレプリカ表から再キャプチャーされず、別のレプリカ表へ転送される。
- `update-anywhere` レプリケーションでのマスター表の変更は再キャプチャーされ、レプリカ表へ送られる。

プログラムごとに 1 つの Windows サービスが作成できる: バージョン 7 では、キャプチャー・プログラムやアプライ・プログラムをすべて操作するのに、1 つの Windows サービスしか作成できませんでした。バージョン 8 では、レプリケーション・アラート・モニターの他、キャプチャー・プログラムやアプライ・プログラムごとに、別々のサービスを作成することができます。各サービスを使用して、レプリケーションを開始または停止することができます。レプリケーション・センターを使用するか、レプリケーション・プログラム用のサービスを作成 (`asnsrct` コマンド) またはドロップ (`asnsdrop` コマンド) する新規のコマンドを使用することができます。

キャプチャー・プログラムおよびアプライ・プログラムに ARM がサポートされる: z/OS 環境では、キャプチャー・プログラム、アプライ・プログラム、およびレプリケーション・アラート・モニターは、MVS 自動リスタート・マネージャー (ARM) を使用できます。ARM とは MVS のリカバー機能で、特定のバッチ・ジョブや開始済みタスクの可用性を向上させることができます。ジョブまたはタスクが失敗するか、ジョブやタスクを実行しているシステムに障害が発生した場合、ARM はオペレーターの介入なしに、ジョブまたはタスクを再始動できます。ARM は、処理対象のアプリケーションを識別するためにエレメント名を使用し、ARM を使用できるアプリケーションはそれぞれ、ARM との通信に使用する固有のエレメント名を使用します。レプリケーション用のエレメント名として、キャプチャー・プログラムには `ASNTCxxxxyyyy`、アプライ・プログラムには `ASNTAxxxxyyyy`、レプリケーション・アラート・モニターには `ASNAMxxxxyyyy` を使用します。

メッセージが改善された: 既存のメッセージが改善され、新規のメッセージが追加されました。解説およびユーザー応答のセクションは更新されました。

パフォーマンスの向上

レプリケーション表間の結合が減少した: バージョン 8 の場合、状態によっては、結合が削除されます。多くの場合、アプライ・プログラムは、ユーザーがコピーしたターゲット表を転送するのに、CD と UOW 表を結合する必要はありません。また、ブルーニングでも、CD と UOW 表を結合する必要はありません。

キャプチャー・ブルーニングと、DB2 ログの読み取りを並行して稼働できる (UNIX、Windows、z/OS): キャプチャー・プログラムは、表のブルーニング中に DB2 ログを読み取ります。したがって、ブルーニングはキャプチャー待ち時間に影響を与えません。バージョン 7 では、キャプチャー・プログラムはこれらのタスクを、並行してではなく、順に実行していました。また、バージョン 8 では、キャプチャー・プログラムは、新規シグナル (IBMSNAP_SIGNAL) 表やモニター (IBMSNAP_CAPMON) 表の他に、UOW 表、CD 表、トレース表もブルーニングします。

ターゲット表のフル・リフレッシュがより高速に行なえる (UNIX, Windows, z/OS): DB2 レプリケーションでは、ターゲット表をより高速にフル・リフレッシュできるよう、以下の DB2 製品でロードが改善されました。

- DB2 Universal Database for Windows and UNIX (バージョン 8)
- DB2 Universal Database for z/OS and OS/390 (バージョン 7 以降)

アプライ・プログラムがサブスクリプション・セットを 1 つだけ持つ場合、処理を最適化する: バージョン 8 では、単一サブスクリプション・セットに関する情報をキャッシュして再利用するように、アプライ・プログラムを開始できます。新規のキーワード **opt4one** を使用すると、CPU 使用率やスループットを改善できます。

複数メンバーを持つサブスクリプション・セットの更新回数が減少した: DB2 レプリケーションの以前のバージョンと比較すると、バージョン 8 では、複数メンバーを持つサブスクリプション・セットのコントロール表を更新する回数が少なくなりました。

新規のユーザー・インターフェース

バージョン 8 では、レプリケーション環境のセットアップや保守、管理ツールを使用したキャプチャー・プログラム、アプライ・プログラム、およびレプリケーション・アラート・モニターの操作が可能です。新しい DB2 レプリケーション・センターはグラフィカル・ツールで、DB2 - DB2 間レプリケーション環境の管理や、DB2 - 非 DB2 リレーショナル・データベース間のレプリケーション の管理をサポートします。

DB2 レプリケーション・センターは、DB2 コントロール・センターのツールのセットの一部で、その他の DB2 センターと似たところがあります。レプリケーション・センターには、DB2 コントロール・センターや DB2 DataJoiner レプリケーション管理 (DJRA) ツールの、以前から使用可能なレプリケーション関数のすべてが含まれます。また、レプリケーション・センターにはランチパッドがあり、これを使用して DB2 レプリケーション環境のセットアップに必要な基本関数を実行することができます。ランチパッドは、さまざまなステップがお互いにどのように関係しているかをグラフィカルに表示します。

レプリケーション・センターを使用して、以下を行なうことができます。

- コントロール表、ソース・オブジェクト、およびターゲット・オブジェクト用プロファイルのデフォルトの定義
- レプリケーション・コントロール表の作成
- レプリケーション・ソースの登録
- サブスクリプション・セットの作成と、サブスクリプション・セットへのサブスクリプション・セット・メンバーの追加
- キャプチャー・プログラムの操作
- アプライ・プログラムの操作

- レプリケーション処理のモニター
- レプリケーション用基本トラブルシューティングの実行

さらに、レプリケーション・センターを使用して、その他多くのレプリケーション管理タスクを実行することができます。

新機能

複数のキャプチャー・プログラムが、同じ DB2 ログまたはジャーナルを並行して読み取ることができる: 複数のキャプチャー・プログラムを単一 DB2 ログ (DB2 カタログ) またはジャーナルに対して稼働することができます。z/OS データ共用グループの場合、複数のキャプチャー・プログラムがデータ共用グループのログを読み取ることができます。各キャプチャー・プログラムは互いに独立しています。必要な場合、単一ソース表を複数のキャプチャー・プログラムへ登録することができます。したがって、低い待ち時間表の場合、その各表に専用のキャプチャー・プログラムを持たせ、それぞれ異なるランタイム優先順位と異なるキャプチャー特性 (プルーニング・インターバルなど) を設定することができます。また、別の編成が、それぞれ独自のレプリケーション環境を、同じソース・データと異なるキャプチャー・プログラムを使用して保守することも可能です。z/OS プラットフォームでは、複数のキャプチャー・プログラムを使用して、単一 DB2 サブシステム内に ASCII、EBCDIC、および UNICODE ソース表を混在させることもできます。

連合データベースごとに複数の非 DB2 リレーショナル・ソースを定義できる: 非 DB2 ソースを含むレプリケーション環境では、単一連合データベース内に複数の非 DB2 リレーショナル・ソースを定義することができます。

自動モニター: 新しいレプリケーション・アラート・モニターは、連続的に実行し、キャプチャーおよびアプライ・プログラムをモニターします。ユーザーは、モニターする基準にしきい値を定義し、そのしきい値に達するか、またはそれを超えた場合に、自動的に電子メールで知らせる先を指定します。レプリケーション・センターまたは 2 つの新規コマンド (**asnmon** と **asnmcmd**) を使用して、レプリケーション・アラート・アラートを構成、操作することができます。

オンデマンド・モニター: キャプチャー、アプライ、およびモニター・プログラムの状況を **asnccmd**、**asnacmd**、**asnmcmd** 状況コマンドを使用して照会することができます。

暗号化されたパスワード・ファイル (UNIX、Windows): バージョン 7 では、アプライ・プログラムやレプリケーション・アナライザーが使用するパスワード・ファイルには、暗号化された情報ではなく、プレーン・テキストが含まれていました。バージョン 8 では、パスワード・ファイル内のパスワードは暗号化されています。プレーン・テキストにはパスワードは保管されません。新規のコマンド (**asnpwd**) を使用して、パスワード・ファイルを作成、保守することができます。

改善された ASNLOAD 出口ルーチン (UNIX, Windows, z/OS): ASNLOAD 出口ルーチンは、ソース・フォーマット (C) とコンパイル済みフォーマットの両方で、サンプルの出口ルーチンとして提供されます。サンプルの出口ルーチンは DB2 プラットフォームによって異なり、そのプラットフォームで提供されるユーティリティ・オプションを利用します。サンプルのコンパイル済みプログラム出口ルーチンを提供されたまま使用したり、レプリケーション構成をカスタマイズして動作を変更したり、出口ルーチンのコード自体をカスタマイズすることができます。

コールド・スタートをより幅広くコントロールできる (UNIX, Windows, および z/OS) : **warm** 始動パラメーターは以下の 2 つのパラメーターに置き換えられ、コールド・スタートをより幅広くコントロールすることができるようになりました。

warmsi

ウォーム・スタート情報が使用できる場合は、キャプチャー・プログラムは前回実行時に終了した場所から処理を再開します。しかし、これが初回で、キャプチャー・プログラムの開始中、または新規の再始動 (IBMSNAP_RESTART) 表が空の場合は、キャプチャー・プログラムはコールド・スタートに切り替えます。これは、バージョン 8 のデフォルトの始動パラメーターです。

warmsa

ウォーム・スタート情報が使用できる場合は、キャプチャー・プログラムは前回実行時に終了した場所から処理を再開します。しかし、キャプチャー・プログラムがウォーム・スタートできない場合は、コールド・スタートに切り替えます。

アプライ・プログラムによるコミットがより頻繁になる: ユーザー・コピー、ポイント・イン・タイム、またはレプリカ・ターゲット表がサブスクリプション・セットにある場合は、アプライ・プログラムが、指定したトランザクション数を処理した後にその作業をコミットするよう指定することができます。これを行なうには、アプライ・プログラムをトランザクション・モードで実行する必要があります。

参照保全性をより多くのタイプのターゲット表に: 多くの場合、アプライ・プログラムがトランザクション・モードで作業をコミットするよう開始すると、ユーザー・コピーやポイント・イン・タイムのターゲット表に参照保全性を持たせることができます。

キャプチャー・プログラムの操作パラメーターを設定する方法が増えた: キャプチャー・プログラムを操作するために、提供されたデフォルトを使用することができます。また、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) を使用して、ユーザーのレプリケーション環境に合うよう新しいデフォルトを作成することもできます。さらに、あるセッションではデフォルトを使用したくない場合、プログラムの開始時に、キャプチャー・プログラムの操作パラメーターを提供することもできます。キャプチャー・プログラムの実行中に、レプリケーション・センター、**asnccmd** コマンドの **chgparms** キーワード (UNIX, Windows, z/OS)、または **OVRDPRCAPA** コマンド (iSeries) を使用して、操作パラメーターを変更することができます。これらの変更は、セッションを終了するか、または別の変更コマンドを出すまで、有効です。

ターゲット・キー列への変更を複製するための新規オプション: バージョン 7 では、キー列への変更がターゲット表へ正しく複製されたことを、ソース表を登録して、更新を削除 / 挿入ペアとしてキャプチャーすることで確かめることができました。バージョン 8 では、サブスクリプション・セット・メンバーの定義時に、アプライ・プログラムが WHERE 節をその述部に主キー列を使用して組み立てる際に、変更前イメージ値か、変更後イメージ値のいずれを使用するかを指定することができます。変更前イメージ値を使用すると、更新を挿入へ変換されることを回避できます。登録が更新に削除 / 挿入ペアを使用するか、サブスクリプション・セット・メンバーが Apply WHERE 節の述部に変更前イメージ値を使用するかを指定することができます。

コントロール表に追加のヒストリー・データが提供される: DB2 レプリケーションでは、レプリケーション活動を記述する追加のヒストリー・データを、コントロール表に提供します。そのようなデータは 2 つの新しい表、アプライ・トレース (IBMSNAP_APPLYTRACE) 表とキャプチャー・モニター (IBMSNAP_CAPMON) 表に入れます。それらのデータを照会するには、レプリケーション・センターを使用できます。

キャプチャー・プログラムでブルーニングできる表が多くなる: キャプチャー・プログラムでは、新規のシグナル (IBMSNAP_SIGNAL) 表やモニター (IBMSNAP_CAPMON) 表の他に、CD 表、UOW 表、トレース (IBMSNAP_CAPTRACE) 表もブルーニングします。さらに、OS/400 環境では、キャプチャー・プログラムは、アプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表をブルーニングします。

より長い表名や列名をサポートする: 現在 DB2 レプリケーションは、ソース表やターゲット表の名前は最長 128 文字まで、列名は 30 文字まで (ロング・ネームをサポートするデータベースのため) サポートします。

キャプチャー・プログラムの実行中に、ソース表や CD 表へ列を追加できる: キャプチャー・プログラムを再初期設定したり、停止および再始動しなくても、レプリケーション・ソース表へ列を追加することができます。UNIX、Windows、および z/OS では、キャプチャー・プログラムの実行中に、CD 表を変更することもできます。

新規のシグナルでキャプチャー・プログラムをコントロールする: シグナル (IBMSNAP_SIGNAL) 表に書き込まれるシグナルによって、キャプチャー・プログラムをコントロールすることができるようになりました。シグナル表は、ログ・レコードを介してキャプチャー・プログラムと通信する方法を提供します。キャプチャー・プログラムは、以下の状態の場合に、シグナルを使用します。

- 特定の表に対する変更のキャプチャーを開始する時を決定するため。
- 特定の表に対する変更のキャプチャーを停止する時を決定するため。¹
- 終了する時を決定するため。

1. このシグナルは OS/400 環境ではサポートされません。

- update-anywhere レプリケーションを実行する必要があるかどうか。
- アプライ・イベントに正確なエンドポイントを設定するためのログ・シーケンス番号を提供するため。

シグナル表は、アプライ・プログラムがキャプチャー・プログラムヘデータのキャプチャー開始時を知らせるようにするだけでなく、ログ・レコードの読み取りを正確に終了したり、ユーザー定義シグナルをログ・レコードに入れることもできます。

データ・リンク値の複製 (AIX、Solaris オペレーティング環境、Windows、iSeries):

- 外部ファイルをポイントする DATALINK 値がある場合、列が RECOVERY YES で定義されているならば、整合したバージョンのファイルを検索することができます。過去のリリースでは、DB2 がファイルの最新コピーを複製しても、その複製ファイルが、複製されたデータベース・データ値と矛盾しないことを保証するものではありませんでした。
- ソース・データベース内の複数の変更について、同じターゲット・ファイルを保守することができます。
- AIX、Windows、および Solaris オペレーティング環境では、DB2 Data Links Manager レプリケーション・デーモン (DLFM_ASNCOPYD) へ接続し、レプリケーション用のデータ・リンク・ファイルを検索したり、保管することができます。以前のリリースのように、別の ASNDLCOPYD デーモンを開始および保守する必要はありません。OS/400 では、まだ、別の ASNDLCOPYD デーモンを開始および保守する必要があります。

ユニコード・エンコード方式が追加された (z/OS): DB2 DataPropagator for z/OS バージョン 8 は、UNICODE および ASCII エンコード方式をサポートします。この機能は DB2 DataPropagator for OS/390 バージョン 7 で登場しました。

64 ビット・サポートが追加された (Windows、UNIX、z/OS): バージョン 8 では、DB2 が 64 ビット・サポートを提供するプラットフォームで複製することができます。アプリケーションを 64 ビットの実オペレーティング・システムで稼働すれば、これらのシステムが提供するメモリー・アドレス・スペースが増大したことによる利益を得ることができます。

移行ユーティリティー: 新規のレプリケーション移行ユーティリティー (**asnmig8**) は、バージョン 5、バージョン 6、またはバージョン 7 のレプリケーション表をすべて、バージョン 8 のフォーマットに変換するために使用できる、移行スクリプトのセットで構成されています。

保守容易性の向上

新規のトレース機能 (UNIX、Windows、z/OS): 新規のレプリケーション・トレース機能 (**asntrc**) は、DB2 トレース機能と似ています。キャプチャーおよびアプライ・プログラムを停止、再始動しなくても、トレース機能を開始または停止することができます。さらに、トレース出力は簡潔で、通常、以前のリリースで生成されていたトレース・ファイルよりも小さくなり、DB2 トレース・フォーマットと整合しています。

トレースの改善 (iSeries): 新規のトレース・ポイントは、より多くのデバッグ情報を提供するために、バージョン 7 のトレース機能へ追加されました。

レプリケーション・アナライザー・プログラムが更新された: 新規の V8 機能を分析するために、レプリケーション・アナライザー・プログラムは変更されました。アナライザーは、特定のシステムにあるレプリケーション・コントロール表の状態について、レポートを生成します。これらのレポートは、ユーザーのレプリケーション環境の検査および調整、または問題の診断に使用することができます。アナライザーおよびそのドキュメンテーションは、Web からダウンロードすることができます。

レプリケーション・システム・コマンドの変更

新規および変更されたレプリケーション・システム・コマンド

(UNIX、Windows、z/OS): Windows、UNIX、および z/OS の既存のシステム・コマンド構文が変更されました。以下の変更も行なわれました。

- キャプチャー・コマンド行 (**asnccmd**) は、**asnccmd** へ名前が変更になりました。これは、アプライ・プログラムを操作するために使用する、新規のアプライ・コマンド行 (**asnacmd**) や、モニター・プログラムを操作するために使用する、新規のモニター・コマンド行 (**asnmcmd**) と整合した名前になりました。
- キャプチャー・プログラムを開始する **asnccp** コマンドの名前が、**asnccp** へ変更されました。

UNIX、Windows、および z/OS オペレーティング・システムで稼働する、以下の新規システム・コマンドが追加されました。

- **asnacmd** (アプライ・コマンド行) は、アプライ・プログラムを操作および停止します。
- **asnmon** (モニター・コマンド) は、レプリケーション・アラート・モニターを開始します。
- **asnmcmd** (モニター・コマンド行) は、レプリケーション・アラート・モニターを操作および停止します。
- **asnanalyze** (アナライザー・コマンド) は、レプリケーション・コントロール表の状態についてのレポートを生成します。
- **asnpwd** (パスワード・コマンド) は、分散レプリケーション環境に必要なパスワード・ファイルを作成および保守します。

- **asntrc** (トレース機能) は、キャプチャーおよびアプライ・プログラム用トレースを生成するために、始動オプションを置き換えます。

新規および変更されたレプリケーション・システム・コマンド (**OS/400 オペレーティング・システム (iSeries)**): OS/400 システムで稼働する、以下の新規システム・コマンドが追加されました。

- **ADDDPRREG** (DPR 登録の追加) は、ユーザー表をレプリケーション用に登録します。
- **RMVDPRREG** (DPR 登録の除去) は、レプリケーションで使用できるソース表のリストから、ユーザー表を除去します。
- **ADDDPRSUB** (DPR サブスクリプション・セットの追加) は、空のサブスクリプション・セット、またはメンバーが 1 つだけのサブスクリプション・セットを作成します。
- **RMVDPRSUB** (DPR サブスクリプション・セットの除去) は、空のセットか、またはセットの全メンバーを除去します。
- **ADDDPRSUBM** (DPR サブスクリプション・セット・メンバーの追加) は、メンバーを既存のサブスクリプション・セットへ追加します。
- **RMVDPRSUBM** (DPR サブスクリプション・セット・メンバーの除去) は、サブスクリプション・セットからサブスクリプション・セット・メンバーを 1 つ除去します。
- **OVRDPRCAPA** (DPR キャプチャー属性のオーバーライド) は、現在実行中のキャプチャー・プログラムの属性を変更します。
- **ANZDPR** (アナライザー) は、特定のシステムにあるレプリケーション・コントロール表の状態についてのレポートを生成します。これらのレポートは、ユーザーのレプリケーション環境の検査および調整、または問題の診断に使用することができます。
- **WRKDPTRC** (トレース・オプション) は、Dump などの各種のトレース・オプションを操作します。

変更された既存のシステム・コマンド (OS/400 システム)

- **DPRVSN** (DataPropagator バージョン) パラメーターは、すべてのシステム・コマンドから取り除かれました。
- **CAPCTLLIB** (キャプチャー・コントロール・ライブラリー) パラメーターが、Capture コマンドに追加されました。
- 新規のトレース関数やモニター機能を利用するために、新規のパラメーターが、**CHGDPRCAPA** (DPR キャプチャー属性の変更) および **STRDPRCAP** (DPR キャプチャーの開始) コマンドに追加されました。
- スペースの再利用のために CD 表や UOW 表を自動的に再編成するよう、新規のパラメーターが **ENDDPRCAP** (DPR キャプチャーの終了) コマンドに追加されました。

- アプライ・プログラムを 1 度だけ実行し、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表をクリーンアップし、単一サブスクリプション・セットの処理を最適化するための、新規のパラメーターが **STRDPRAPY** (DPR アプライの開始) コマンドに追加されました。

コントロール表の変更

新機能のサポートとユーザビリティの改善のため、バージョン 8 では、コントロール表へ重要な変更が行なわれました。新規の表が追加され、それによって、必要でなくなった表もあります。また、変更された既存の表もいくつかあります。

以下の新規の表が追加された:

- IBMSNAP_APPENQ は、単一アプライ修飾子に対して 1 つのアプライ・プログラムだけが実行されることを保証するものです。
- IBMSNAP_APPLYTRACE は、アプライ・プログラムからの重要なメッセージを保持します。
- IBMSNAP_CAPENQ は、単一キャプチャー・スキーマに対して 1 つのキャプチャー・プログラムだけが実行されることを保証するものです。
- IBMSNAP_CAPMON は、キャプチャー・プログラムの進行状況をモニターするための操作統計を保持します。
- IBMSNAP_CAPSCHEMAS は、すべてのキャプチャー・スキーマの名前を保持します。
- IBMSNAP_PRUNE_SET は、CD 表のプルーニングを調整します。
- IBMSNAP_RESTART は、キャプチャー・プログラムがログやジャーナル内の正しい時点からキャプチャーを再開できるようにします。
- IBMSNAP_SIGNAL は、キャプチャー・プログラムをコントロールするために使用するシグナルを保持します。

レプリケーション・アラート・モニター用に、以下の新規の表が追加された:

- IBMSNAP_ALERTS は、レプリケーション・アラート・モニターから発行されるすべてのアラートの履歴を保持します。
- IBMSNAP_CONDITIONS は、モニターされる各サーバーのアラート条件を保持します。
- IBMSNAP_CONTACTGRP はグループとの連絡先をマップします。
- IBMSNAP_CONTACTS は、連絡先名とアドレスを保持します。
- IBMSNAP_GROUPS は、連絡先グループを保持します。

2. この表は OS/400 では使用されません。

- IBMSNAP_MONENQ は、単一モニター修飾子に対して 1 つのモニター処理だけが実行されることを保証するものです。
- IBMSNAP_MONSERVERS は、レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーを最後にモニターした時刻を保持します。
- IBMSNAP_MONTRACE は、レプリケーション・アラート・モニターの活動をトレースします。
- IBMSNAP_MONTRAIL は、モニター・サイクルごとにモニターの活動の履歴を保持します。

以下の表が変更された:

- IBMSNAP_APPLYTRAIL
- IBMSNAP_AUTHTKN (OS/400 のみ)
- IBMSNAP_CAPPARMS (以前は IBMSNAP_CCPPARMS)
- IBMSNAP_CAPTRACE (以前は IBMSNAP_TRACE)
- IBMSNAP_PRUNCNTL
- IBMSNAP_REG_EXT (OS/400 のみ)
- IBMSNAP_REGISTER
- IBMSNAP_SUBS_COLS
- IBMSNAP_SUBS_EVENT
- IBMSNAP_SUBS_MEMBR
- IBMSNAP_SUBS_SET
- IBMSNAP_UOW

CD 表も変更されました。

DB2 レプリケーションの以前のバージョンの表で、以下のものは現在は使用されません:

- IBMSNAP_CRITSEC は IBMSNAP_SIGNAL に置き換えられました。
- IBMSNAP_WARMSTART は IBMSNAP_RESTART に置き換えられました。

サポートされなくなった機能

DB2 DataJoiner レプリケーション管理 (DJRA) ツールは、バージョン 8 ではサポートされません。したがって、DJRA を使用して、バージョン 8 のレプリケーション・コントロール表を作成したり、バージョン 8 のコントロール表を使用する、ソースの登録や、サブスクリプション・セットの定義はできません。バージョン 7 のレプリケーション環境では、DJRA は引き続きサポートされます。バージョン 8 のレプリケーション環境用レプリケーション・センターを使用してください。

DB2 コントロール・センターは、バージョン 8 のレプリケーション・コントロール表をサポートしません。したがって、コントロール・センターを使用して、バージョン 8 のコントロール表を使用する、ソースの登録や、サブスクリプション・セットの定義はできません。バージョン 7 のレプリケーション環境用コントロール・センターは使用できます。バージョン 8 のレプリケーション環境用レプリケーション・センターを使用してください。

ASNSAT コマンドは、もう使用できません。また、レプリケーション・サブスクリプションの汎用化や、DB2 サテライト・レプリケーション環境のセットアップは、サテライト管理センターから行なえなくなりました。モバイル作業フォースのためにデータをレプリケーションする必要がある場合は、サテライト DB2 データベースを、DB2 Everyplace バージョン 8 へ移行することをお勧めします。詳細は、IBM 担当者へお尋ねください。

第 1 部 レプリケーションのガイド

本書の第 1 部は、以下の章で構成されています。

3 ページの『第 1 章 レプリケーションの計画』では、レプリケーション環境を計画する方法について説明します。

17 ページの『第 2 章 レプリケーションの設定』では、レプリケーション用に環境を設定する方法について説明します。

43 ページの『第 3 章 表およびビューをレプリケーション・ソースとして登録する』では、レプリケーション・ソースを登録するために必要な情報について説明します。

69 ページの『第 4 章 ソースのサブスクリプション』では、サブスクリプション・セットの作成方法と、メンバーをサブスクリプション・セットへ追加する方法について説明します。

105 ページの『第 5 章 特殊なデータ・タイプのレプリケーション』では、ソース表の LOB 値および DATALINK 値のためのレプリケーション・オプションについて説明します。

117 ページの『第 6 章 レプリケーション環境におけるデータのサブセット化』では、どのようにデータがターゲットへアプライされるのかを説明する他、ターゲットへキャプチャーおよびアプライされるデータをカスタマイズする方法についても説明します。

123 ページの『第 7 章 レプリケーション環境におけるデータ操作』では、ソース・データを操作するための、キャプチャー・プログラムまたはアプライ・プログラムの使用方法について説明します。

127 ページの『第 8 章 レプリケーション SQL スクリプトのカスタマイズおよび実行』では、レプリケーション環境で SQL を実行する方法について説明します。

129 ページの『第 9 章 キャプチャー・プログラムの操作』では、すべてのオペレーティング・システム環境について、キャプチャー・プログラムを操作する方法を説明します。

153 ページの『第 10 章 アプライ・プログラムの操作』では、すべてのオペレーティング・システム環境について、アプライ・プログラムを操作する方法を説明します。

177 ページの『第 11 章 レプリケーションのモニター』では、レプリケーション環境をモニターする方法について説明します。

199 ページの『第 12 章 レプリケーション環境の変更』では、レプリケーション環境を変更する方法について説明します。

241 ページの『第 13 章 レプリケーション環境の保守』では、ソース表、コントロール表、およびターゲット表を保守する方法について説明します。

第 1 章 レプリケーションの計画

この章では、レプリケーション環境を計画する方法を説明しています。この章は、以下のセクションを含んでいます。

- 『メモリーの計画』
- 5 ページの『ストレージの計画』
- 12 ページの『競合検出の計画』
- 12 ページの『非 DB2 リレーショナル・ソースの計画』
- 14 ページの『コード・ページ・トランザクションの計画』

メモリーの計画

DB2 レプリケーションに必要なメモリーの量を計画する必要があります。DB2 レプリケーションでは、必要なメモリーだけしか使用されません。必要なメモリー量は、ソースから複製されているデータの量とトランザクションの並行性に直接比例します。基本的に、レプリケーションされているデータの量が多ければ多いほど、また、同時実行されているトランザクションの数が多ければ多いほど、より多くのメモリーがレプリケーションに必要となります。

キャプチャー・プログラムとアプライ・プログラムを実行すると、かなり大量のメモリー・リソースが消費される可能性があります。

キャプチャー・プログラムによって使用されるメモリー

キャプチャー・プログラムが DB2 ログを読み取る際には、関連したコミット・レコードまたはアポート・レコードを読み取るまで、キャプチャー・プログラムは個々のトランザクション・レコードをメモリーに保管します。アポートされたトランザクションに関連したデータはメモリーから消去され、コミット・レコードに関連したデータは CD 表および UOW 表に書き込まれます。コミットされたトランザクションは、キャプチャー・プログラムがコミット・インターバルに達して作業をコミットするまでメモリー内に留まります。

キャプチャー・プログラムが使用しているメモリー量をモニターするには、キャプチャー・モニター (IBMSNAP_CAPMON) 表の CURRENT_MEMORY 列の値を調べてください。

キャプチャー・プログラムの始動時に **memory_limit** パラメーターを設定すると、トランザクションと関連付けられているストレージとして、キャプチャー・プログラムが確実に指定された量のメモリーを使用するようにできます。このパラメーターによってその他のストレージ使用が限定されることはありません。また、**memory_limit** パラメー

ターは、キャプチャー・プログラムの実行中でも変更できます。メモリー限度に達すると、キャプチャーは一部のトランザクションを予備ファイルに書き込みます。予備ファイルのストレージ要件については、10 ページの『キャプチャー・プログラム用予備ファイルのスペース所要量の計画』を参照してください。キャプチャー・プログラムによって使用されるメモリー・リソースについては、このプログラムのストレージ・スペース要件と関連付けて検討する必要があります。

また、キャプチャー・プログラムのメモリー要件について計画する際には、ユーザー・トランザクションのサイズとコミット・インターバルも検討する必要があります。キャプチャー・プログラムの実行時にバッチ・ジョブを一時コミットなしで長時間実行すると、大量のメモリーが消費されます。一般に、コミット・インターバルの値が小さければ小さいほど、キャプチャー・プログラムが必要とするメモリーは少なくなります。

登録に関する情報の読み取り: アクティブ登録に関する情報は、キャプチャー・プログラムがインスタンス生成される時、およびキャプチャー・プログラムの実行中に登録が動的に追加されるときに、読み取られてメモリーに保管されます。

ログ・レコードの読み取り (UNIX、Windows、z/OS): DB2 レプリケーションは、ログ・レコードを読み取る際にメモリー・バッファーを使用します。UNIX および Windows オペレーティング・システムの場合は、バッファーのデフォルトのサイズは 4 KB (K バイト) ページ x 50 です。z/OS オペレーティング・システムの場合は、デフォルトのサイズは 1 KB ページ x 66 で、これは ECSA (拡張共通サービス域) ストレージです。レプリケーションが ECSA を使用するのはこの場合だけです。

OS/400 上で使用されるメモリー: CURRENT_MEMORY は、アクティブ CD 表のために標準入出力バッファーによって使用されるメモリーに入らないトランザクション・レコードを保留するために割り振られた追加のメモリーの最新のアカウントです。これは、多数のトランザクションを保留するために使用されている追加のメモリーの量を示す指標です。これは、特定のジャーナル・ジョブによって使用されているメモリーすべての正確な合計ではありません。

キャプチャー・モニター (IBMSNAP_CAPMON) 表に保管されている情報により、メモリーの使用を調整するのに役立つ稼動統計が提供されます。この表内の値は特定のキャプチャー・モニター・インターバルに関するものであり、各モニター・インターバルにわたった累積的な値ではないことに注意してください。CURRENT_MEMORY 列のデータには、加算されるカウントは含まれません。レコードの作成時に、この値にはモニター・インターバルが終了した時点で使用中であったメモリーが反映されます。キャプチャー・モニター・インターバルは、キャプチャー・プログラムがこの表にデータを挿入する頻度を決定します。以下の方法のいずれかを使用して、キャプチャー・プログラムによって使用されているメモリーの量を調整してください。

予備の分も考慮したメモリー限度のチューニング:

1. キャプチャー・プログラムの始動時に、デフォルトのメモリー限度を使用する。

2. キャプチャー・モニター (IBMSNAP_CAPMON) 表の TRANS_SPILLED 列を調べて、データがメモリーから一時ファイルにあふれ出たかどうかをチェックする。この列は、メモリー制限のために特定のキャプチャー・モニター・インターバルの間にディスクにあふれたソース・システム・トランザクションの数を示しています。
3. データがメモリーからあふれた場合は、引き上げたメモリー限度または値を低くしたコミット・インターバルのいずれかを使用する。

あふれるのを防ぐためのメモリー限度のチューニング:

1. キャプチャー・プログラムの始動時に、メモリー限度を高く設定する (どのくらい高くするのは、システム・リソースに応じて決まります)。
2. キャプチャー・モニター (IBMSNAP_CAPMON) 表の CURRENT_MEMORY 列を調べて、使用中のメモリー量をチェックする。この列は、キャプチャー・プログラムが特定のキャプチャー・モニター・インターバルの間に使用したメモリーの量を (MB 単位で) 示しています。
3. メモリー限度として指定した量よりもかなり少ないメモリーしか使用されていない場合は、メモリー限度としてもう少し下げた値を設定する。

アプライ・プログラムによって使用されるメモリー

アプライ・プログラムがデータを取り出す場合、通常個々の行を取り出すために使用されるメモリーは少量です。使用されるメモリー量は、表の列のサイズおよび一度に取り出される行の数に比例します。たとえば、アプライ・プログラムが LOB 列を取り出す場合は、このプログラムは 2 G バイトのメモリーを潜在的に使用する可能性があります。

アクティブ・サブスクリプション・セットに関する情報は、アプライ・プログラムの実行中に読み取られてメモリーに保管されます。通常、アプライ・プログラムによって一度に使用されるメモリー量は、ほとんどのメンバーが含まれているサブスクリプション・セットを処理するのに必要なメモリー量に比例します。

レプリケーション・アラート・モニターによって使用されるメモリー

メモリーは、定義を保管するため、および通知として送信する前にアラートをメモリー内に保持するために使用されます。定義用に必要なメモリーの量は、定義の数に直接比例します。レプリケーション・アラート・モニターは、アラート通知の保管用として 32 KB のメモリーを予約します。それより多いメモリーが、必要に応じて要求されたり、不要になった場合にはリリースされたりします。

ストレージの計画

DB2 が必要とするストレージに加えて、以下の項目のレプリケーションのためにストレージが確実に使用可能であるようにしておく必要があります。

データベース・ログおよびジャーナル・データ

データのレプリケーションをサポートするために記録される追加データ。詳細については、『ログの影響の計画』を参照してください。

ターゲット表およびコントロール表

複製されるデータおよびコントロール表 (CD 表を含む)。詳細については、8 ページの『ターゲット表およびコントロール表のストレージ要件の計画』を参照してください。

一時ファイル

レプリケーション・プログラムによって予備ファイルに保管されるデータ、および診断ログ・ファイル (たとえば、*CAP.log および *APP.log)。詳細については、10 ページの『一時ファイルのストレージ要件の計画』を参照してください。

OS/400: キャプチャー用の現行レシーバー・サイズ

これから取り込まれることになっている登録済みソース表の場合は、ジャーナル・エントリはまだレシーバーの現行チェーン内に残っていなければなりません。詳しくは、40 ページの『ジャーナル・レシーバー削除出口ルーチンの使用』を参照してください。

以下の部分で指定されているサイズはすべて目安にすぎません。実動可能システムを準備して設計するには、障害防止などの要因も考慮に入れなければなりません。たとえば、ネットワークが停止する可能性を考慮に入れて、データを保留する期間 (8 ページの『ターゲット表およびコントロール表のストレージ要件の計画』で説明されています) を延長しなければならない場合があります。

ヒント: ストレージ見積もりが不当に高いと思われる場合は、アプライ・プログラムがサブスクリプション・セットを実行する頻度とレプリケーション表の整理の頻度を再検討してください。ストレージの使用量、最大限の障害許容度、および CPU オーバーヘッドの間のトレードオフについて検討する必要があります。

ログの影響の計画

レプリケーション・サーバーに対するログの影響を計画する必要があります。DB2 レプリケーションでは、ソース表とターゲット表の両方をログに記録する (ジャーナリングする) 必要があります。

DB2 ソース・サーバーの場合のログの影響の計画

一般に、レプリケーションに関係するすべての表のために、現行ログの 3 倍の量のログが追加が必要となります。基本的に、ソース表用、CD 表用、およびレプリケーション・コントロール表用のログ・スペースが必要です。このセクションでは、使用しているレプリケーション環境で予測可能なログの影響をより正確に見積もるのに役立つ可能性のあるその他の要因を示します。

アプリケーションによってソース・データベースに加えられる更新と、レプリケーション要件を検討してください。たとえば、更新アプリケーションが一般に表の列の 60% を更新する場合、レプリケーション要件によっては、ログ・レコードの大きさが、類似の表がレプリケーションされない場合に比べて半分以上も増加する可能性があります。

UNIX、Windows、および z/OS:

- DB2 は UPDATE ステートメントごとに完全な行イメージをログに記録します。これが発生するのは、DATA CAPTURE CHANGES キーワードを使用して表を作成 (または変更) してからでなければ表を複製できないからです。
- ログへの追加量が最大になるレプリケーション要件の 1 つとして、変更前イメージと変更後イメージのキャプチャー (update-anywhere レプリケーション・シナリオにおけるレプリケーション・ターゲット表の例が当てはまる) が挙げられます。ログの量を少なくする方法の 1 つとして、レプリケーション・ソースに対して定義されている列数を少なくすることができます。たとえば、必要でない場合は変更前イメージを取り込まないでください。

OS/400:

- DB2 は UPDATE ステートメントごとに完全な行イメージをログに記録します。ログの量を少なくする方法の 1 つは、レプリケーション・ソースに対して定義されている列数を少なくすることであり、たとえば、必要でない場合は変更前イメージを取り込まないことです。
- CD 表用および UOW 表用として使用されるストレージの量を最小化するには、これらの表を頻繁に再編成する。なぜなら、プルーニングではユーザーのために DASD をリカバーするようなことはないからです。 **ENDDPRCAP** コマンドでキーワード **RGZCTLBTL** (コントロール表の再編成) を使用すれば、コントロール表を再編成できます。通常操作条件で DASD の使用パターンを監視して、DASD 使用の予測と管理に役立ててください。ジャーナリングがオンになっている場合は、UOW 表および CD 表に対する DB2 ログの挿入および削除が行われるにつれてログまたはジャーナルの量が増えることも考慮に入れてください。
- 現行レシーバーがいっぱいになると、システムは新しいレシーバーに切り替える。レプリケーションする必要がなくなった古いレシーバーについては、保管するか削除するかをオプションで指定できます。たぐさんのトランザクションを扱うシステムの場合、キャプチャー・プログラムは場合によって遅れることがあります。キャプチャーが頻繁に遅れる場合は、ソース表を複数のジャーナルに分離して、ワークロードをキャプチャー・プログラムの複数インスタンスに分散できます。

ターゲット・サーバーの場合のログの影響の計画

ソース・データベースのロギングに加えて、行が適用されるターゲット・データベースでもロギングされます。ログへの影響は、アプライ・プログラム用に選択したコミット・モードによって異なります。

表モード

表モード処理では、アプライ・プログラムはフェッチしたデータがすべて適用

された後で単一コミットを発行します。アプライ・プログラムは一時チェックポイントを発行しません。このケースでは、アプライ・プログラムが 1 回のインターバルで処理する最大データ量を見積もり、そのデータ量を格納できるようにログ・スペースを調整する必要があります。

トランザクション・モード

トランザクション・モード処理では、アプライ・プログラムはそれぞれの更新をソース・トランザクションの順序どおりにすべてターゲット表にコピーし、それらの変更を 1 回のインターバルでトランザクション境界に達したときにまとめてコミットします。一時コミットのインターバルを設定するには、サブスクリプション・セット・オプション (**commit_count(x)**) の値 x を設定します。すべての応答セットを取り出すと、アプライ・プログラムは予備ファイルの内容をコミット・シーケンスの順序に従って適用します。このタイプの処理では、すべての予備ファイルを同時に開いて処理できます。たとえば、コミット・カウントを 1 に設定と、アプライ・プログラムは 1 つのトランザクションが終わるたびにコミットします。コミット・カウントを 2 に設定した場合は、アプライ・プログラムは 2 つのトランザクションのセットが 1 つ終わるたびにコミットします。

OS/400: ターゲット・オペレーティング・システムが OS/400 である場合は、ターゲット表のログ・スペース (ジャーナル・レシーバー・スペース) も検討する必要があります。OS/400 では、ターゲット表用のジャーナル・レシーバーを MNGRCV(*SYSTEM) および DLTRCV(*YES) パラメーターで作成でき、さらにジャーナリングする必要があるのは変更後イメージ列だけであるため、以下の公式を使用して、ターゲット表用ジャーナル・レシーバーの大きさを見積もってください。

`journal_receiver_volume=target_table_row_length X journal_receiver_threshold`

ターゲット表およびコントロール表のストレージ要件の計画

新しいターゲット表の大きさを見積もる必要があります。ターゲット表に必要なスペースはソース表のスペース以下であるのが普通ですが、ターゲット表が非正規化されたり、その中に変更前イメージ (変更後イメージに加えて) またはヒストリー・データが含まれていると、ソース表のスペースよりもずっと大きくなる場合があります。ターゲット表のサイズは、レプリケーションする際に何を選択するかによって異なります。選択するものとしては、たとえば、レプリケーションするソース表のパーセント、レプリケーションする列のデータ・タイプ、変更前および変更後のイメージをレプリケーションするかどうか、算出列を追加するかどうか、行をサブセット化するかどうか、レプリケーション時になんらかのトランスフォーメーションを行うかどうかなどが挙げられます。

CD 表および一部のレプリケーション・コントロール表 (IBMSNAP_UOW、IBMSNAP_CAPTRACE、IBMSNAP_APPLYTRACE、IBMSNAP_APPLYTRAIL、IBMSNAP_CAPMON、IBMSNAP_ALERTS) も、DB2 ソース・データベース用として必要なディスク・スペースに影響を与えます。レプリケーション環境のセットアップの仕

方によっては、これらの表が非常に大きくなる可能性があります。上記以外のレプリケーション・コントロール表の場合は、必要なスペースは通常少なく、かつ静的です。

CD 表は、複製されるデータの量に応じてサイズが大きくなり、やがてキャプチャー・プログラムによって整理されます。スペースを節約するために、CD 表を圧縮された表スペースに定義する場合があります。CD 表に必要なスペースを見積もるには、データをどのくらい保持してからプルーニングするのかについて、その期間を最初に決定した後、キャプチャー・プログラムがこれらの表を自動的にプルーニングする頻度、またはユーザー自身がコマンドを使用してこれらの表をプルーニングする頻度を指定します。

複製済みデータのバイト数を計算する場合は、キャプチャー・プログラムが CD 表に加えたオーバーヘッド・データとして 21 バイトを含める必要があります。複製済みデータのバイト数を計算する場合は、キャプチャー・プログラムが CD 表に追加したオーバーヘッド・データの分として 21 バイトをそれぞれの行ごとに組み込む必要があります。キャプチャー・プログラムが CD 表にデータをキャプチャーし続けられる期間 (たとえばネットワークが停止するなどしてデータが適用不能なときも含む) を決定してください。上記の偶発的な障害が発生した期間内にソース表で通常取り込まれる可能性のある挿入、更新、および削除の数を見積もってください。

CD 表の推奨サイズを判別するために、次のガイドラインを使用してください。

```
recommended_CD_size =  
  ( (21 bytes) + sum(length of all registered columns) ) X  
  (number of inserts, updates, and deletes to source table  
  during the contingency period)
```

例: CD 表の行の長さが 100 バイト (これにオーバーヘッドの分の 21 バイトを加える) で、24 時間にわたる偶発的障害の期間内に 100,000 個の更新が取り込まれるとすると、CD 表に必要なストレージは約 12 MB となります。

この式では、登録済み列には変更前イメージと変更後イメージの両方が含まれるものとしています。更新が INSERT 操作と DELETE 操作の対に変換されるようになっている場合は、挿入、更新、および削除の合計数を判別するときにそれらを勘定に入れてください。たとえば、ソース表に対する 1 つの更新は、それぞれ CD 表の 2 つの行としてカウントしてください。

UOW 表は、特定のコミット・インターバルの間にキャプチャー・プログラムによって挿入される行数、および整理される行数に基づいて、拡大および縮小します。アプリケーション・トランザクションが COMMIT を発行するたび、およびトランザクションが登録済みのレプリケーション・ソース表に対して INSERT、DELETE、または UPDATE 操作を実行するたびに、行が 1 つ UOW 表に挿入されます。当初はこの表に必要なスペースを多めに見積もっておき、実際に使用されるスペースをモニターして、回収できるスペースがあるかどうかを判別してください。

一時ファイルのストレージ要件の計画

予備ファイルおよび診断ログ・ファイルのストレージ要件を計画する必要があります。

診断ログ・ファイルのスペース所要量の計画 (UNIX、Windows、z/OS)

診断ログ・ファイルには、レプリケーション・プログラムのアクティビティに関する情報 (プログラムの始動時刻と停止時刻、およびプログラムが発行したその他の通知メッセージまたはエラー・メッセージなど) が保管されます。デフォルトでは、プログラムは再始動後もログ・ファイルにメッセージを付加します。これらのログ・ファイルを含むディレクトリーにはファイルを保管するのに十分なスペースが確実にあるようにしてください。これらのファイルのロケーションは、キャプチャー・プログラム、アプライ・プログラム、およびレプリケーション・アラート・モニター・プログラムのそれぞれの始動時に始動パラメーター `capture_path`、`apply_path`、および `monitor_path` に設定された値によって決まります。

ストレージに関して不安がある場合、プログラム・ログを再利用するオプションがあります。これを使用すると、プログラムは始動するたびにログを削除して再作成します。プログラムの始動時に、ログを再利用するかどうかを指定できます。

キャプチャー・プログラム用予備ファイルのスペース所要量の計画

十分なメモリーがない場合、キャプチャー・プログラムはトランザクションを予備ファイルに書き込みます (つまり、メモリーに入りきらないトランザクションをここに入れます)。キャプチャー・プログラムは、最大のトランザクションを書き込みます。しかし、最大のトランザクションがメモリー限度を超える原因となったトランザクションであるとは限りません。

- **UNIX、Windows:** UNIX および Windows では、予備ファイルは常にディスク上にある。1 トランザクション当たり 1 つのファイルが `capture_path` ディレクトリーに作成されます。
- **OS/400:** OS/400 では、予備ファイルはライブラリー `QTEMP` の中に作成される。予備ファイルが必要な各登録ごとに 1 つの予備ファイルが作成されます。
- **z/OS:** z/OS では、予備ファイルは VIO (仮想入出力) に送られる。

キャプチャー予備ファイルのサイズは、以下の要因によって決まります。

メモリー限度

memory_limit 稼働パラメーターを使用して、キャプチャー・プログラムが使用できるメモリー量を指定します。許可するメモリーが多ければ多いほど、キャプチャー・プログラムが入りきらない分をファイルに入れる可能性が少なくなります。

トランザクションのサイズ

トランザクションのサイズが大きいと、入りきらないためにファイルに入れる必要性が高まります。

同時実行されるトランザクションの数

キャプチャー・プログラムが同時に処理するトランザクションの数が増えた場合、またはキャプチャー・プログラムがインターリーブド・トランザクションを処理する場合は、キャプチャー・プログラムはより多くの情報をメモリーまたはディスクに保管する必要があります。

コミット・インターバル

通常、コミット・インターバルの値が小さければ小さいほど、必要とされるストレージは少なくなります。なぜなら、キャプチャーが情報をコミットする前にその情報をメモリーに保管しておく期間が短くなるからです。

アプライ・プログラム用予備ファイルのスペース所要量の計画

アプライ・プログラムには、データを保管するための一時的なスペースが必要です。ASNLOAD ユーティリティーを使用する場合は、ロード予備ファイルではなくロード入力ファイルになります。アプライ・プログラムは、更新をターゲット表に適用するまでの間それらの更新を保留するために予備ファイルを使用します。一般に、予備ファイルはディスク・ファイルです。ただし、z/OS プラットフォームでは、入りきらないデータをメモリーに入れるように指定できます。仮想メモリーの制約がない限り、予備ファイルはディスク上ではなく仮想メモリーに保管してください。

予備ファイルのサイズは、各レプリケーション・インターバル内でレプリケーションするよう選択されたデータのサイズに比例します。予備ファイルのサイズは、通常はこのデータのサイズのおよそ 2 倍です。予備ファイルのサイズを見積もるには、アプライ・プログラム用に計画した頻度インターバル (またはデータ・ブロック値) と、同じ期間内 (または変更のピーク時) になされた変更の量を比較します。

OS/400 では、予備ファイルの行サイズは定数で 32 KB です。

UNIX、Windows、および z/OS では、予備ファイルの行サイズはターゲット行のサイズであり、これにはあらゆるレプリケーション・オーバーヘッド列が含まれます。この行サイズは、DB2 パック内部フォーマットでのサイズではなく、拡張された解釈済みの文字フォーマット (SELECT から取り出された状態) でのサイズです。行には、行の長さ、個々の列ストリング上の NULL 終了文字も含まれます。以下の例では、レプリケーションするよう選択されたデータに必要な予備ファイルのサイズを見積もります。この例では、その予備ファイルに保管される他のデータに必要な追加のスペースは勘定に入っていません。

例: 変更量のピークが時間あたり 12,000 件の場合に、アプライ・プログラムの頻度が 1 時間インターバルで計画されていると、予備ファイルには 1 時間分の更新件数、つまり 12,000 件を保留しなければなりません。1 件の更新が 100 バイトのデータを表すとすれば、予備ファイルは最小限でも約 1.2 MB になります。これに加え、この予備ファイルに保管される他のデータ用のスペースも必要です。

競合検出の計画

標準または拡張競合検出を使用している場合は、レプリカ・ターゲット表用の CD (または CCD) 表に変更前イメージを保管する必要があります。また、参照保全規則に制限があります。対等シナリオおよび update-anywhere シナリオの場合、またはアプライ・プログラムがトランザクション・モード処理を使用する場合は、ソース規則に合わせた参照保全規則を定義する必要があります。

対等シナリオまたは update-anywhere シナリオを使用し、かつ競合検出をオンにしない場合は、更新競合が防止されるようなアプリケーション環境を設計する必要があります。使用するアプリケーション環境では競合が起こり得ないのであれば、競合検出を使用しないことで処理サイクルを節約できます。

以下の方法のいずれかを使用して、対等レプリケーションおよび update-anywhere レプリケーションでの競合を防止してください。

キー別のフラグメント化

レプリケーション・ソースが、特定のサイトのキー範囲別にレプリカによって更新されるようにアプリケーションを設計します。たとえば、ニューヨークにあるサイトでは、米国東部のセールス・レコード (キー範囲として 49999 以下の ZIP コード³を使用) だけを更新できます。ただし、すべてのセールス・レコードの読み取りは行うことができます。

時間別のフラグメント化

特定のサイトで特定の期間においてだけ表を更新できるようにアプリケーションを設計します。それらの期間は、互いに十分隔たっていない限りなりません。なぜなら、これからマスター・バージョンになるサイトに対して加えられた、すべての保留中の変更をレプリケーションできるようにするためです。夏時間 (またはサマータイム) などの時間の変更や、時間帯の相違にも対応します。

非 DB2 リレーショナル・ソースの計画

非 DB2 リレーショナル・データベースからのレプリケーションの場合は、キャプチャー・プログラムの代わりにキャプチャー・トリガーが使用されます。これらのトリガーは、変更済みのデータを非 DB2 リレーショナル・ソース表からキャプチャー、その変更済みのデータを CCD 表にコミットします。キャプチャー・トリガーは、トランザクションのスループット率とログ・スペース所要量に影響します。また、使用している環境に既存のトリガーがある場合は、それらを新しいキャプチャー・トリガーとマージしなければならない場合があります。詳細については、以下のセクションを参照してください。

3. 米国の郵便番号

- 『キャプチャー・トリガーの場合のトランザクション・スループット率の計画』
- 『非 DB2 リレーショナル・ソース・サーバーの場合のログの影響の計画』
- 『Oracle ソース・サーバーの場合のロックの計画』
- 『既存のトリガーとキャプチャー・トリガーとの共存の計画』

キャプチャー・トリガーの場合のトランザクション・スループット率の計画

トリガー・ベースの変更キャプチャーはトランザクション・スループット率に影響を与えるため、ソース・システムのトランザクション・ワークロードが増加します。また、キャプチャー・トリガーは更新トランザクションの場合の応答時間も増加させます。この影響が最大となるのは、複製されるアプリケーション・ソース表をきわめて頻繁に更新するトランザクションの場合です。

非 DB2 リレーショナル・ソース・サーバーの場合のログの影響の計画

非 DB2 リレーショナル・ソース・サーバーの場合、ソース・アプリケーションはより多くのアクティブ・ログ・スペースを必要とします。なぜなら、ログの量が複製されるソース表のおよそ 3 倍になるからです。変更は、トリガーによってソース表から取り込まれて、CCD 表に保管されます。変更済みのデータは変更中のソース表と同じコミット有効範囲内で書き込まれ、データは後でトリガー・ベースのブルーニング機構によって削除されます。したがって、1 回のソース INSERT、UPDATE、または DELETE 操作は、それぞれ 1 回の INSERT、UPDATE、または DELETE 操作に INSERT 操作と DELETE 操作をさらに 1 回ずつ加えたものとなります。更新を DELETE 操作と INSERT 操作の対に変更する場合は、ログの量はさらに増えます。

ログ・スペースを使い尽くしてしまい、キャプチャー・トリガーがレコードを CCD 表に挿入できなくなると、ユーザーまたはアプリケーション・プログラムによって試行されたトランザクションは正常に完了しません。

Oracle ソース・サーバーの場合のロックの計画

現在 Oracle ソースを更新中のアプリケーションがすべて終了してからでなければ、アプライ・プログラムはデータの適用を開始できません。データを処理して同期点を設定できるように、アプライ・プログラムは CCD 表をロックする必要があります。CCD 表に対するロックが保留されるのは、アプライ・サイクル全体を通してではなく、アプライ・プログラムが同期点を設定するまでの間だけです。ソース表を更新する必要のあるアプリケーションは、アプライ・プログラムが CCD 表をアンロックするまで待たなければなりません。

既存のトリガーとキャプチャー・トリガーとの共存の計画

キャプチャー・トリガー・ロジックは、ソースの登録時にレプリケーション・センターによって生成される SQL スクリプトに含まれています。デフォルトでは、INSERT ト

リガー、UPDATE トリガー、および DELETE トリガーは、これらのタイプの変更（挿入、更新、および削除）をソース表から複製できるように作成されます。トリガー名は、CCD 表の名前と、トリガーのタイプを記述する先行文字（I は INSERT、U は UPDATE、D は DELETE）から構成されます。たとえば、CCD 表の名前が undjr02.ccd001 である場合は、生成される DELETE トリガーの名前は undjr02.dccd001 です。スクリプト内に生成されたトリガー名を変更してはなりません。

レプリケーション用に登録する表にトリガーがすでに存在しており、かつそのトリガーの名前が生成されたスクリプト内の名前と同じである場合は、そのスクリプトの生成時に警告を受け取ります。その生成されたスクリプトを実行しないでください。なぜなら、既存のトリガーが RDBMS によって上書きされる可能性があるからです。既存のトリガーと新しいトリガーとをどのようにマージするのかを決定し、既存のロジックとレプリケーション・センターによって生成されたトリガー・ロジックとをマージするスクリプトを作成してください。

作成するトリガー・タイプがレプリケーション用に登録する表にすでに存在しており、かつ RDBMS がそのタイプのトリガーを 1 つの表当たり 1 つだけしか許可しない場合は、生成されたスクリプトを実行する前にロジックをマージする必要があります。

コード・ページ・トランザクションの計画

レプリケーション・コンポーネントは、各種のプラットフォームでデータのコード・ページ・トランザクションの処理を DB2 データベースに依頼するデータベース・アプリケーションです。これらは SQL SELECT、INSERT、UPDATE、および DELETE ステートメントを使用してデータを処理します。

互換コード・ページを持つデータベース間でのデータのレプリケーション

使用するレプリケーション構成では SQL ステートメントが必要で、さらに異なるコード・ページを使用するシステム間でデータをやり取りする必要がある場合は、DRDA などの下層の DB2 プロトコルがコード・ページ・トランザクションを処理します。また、データが DB2 リレーショナル・データベースと非 DB2 リレーショナル・データベース間で渡される場合は、DB2 レプリケーションは必要なコード・ページ・トランザクションの処理をすべて下層のデータベース製品に依頼します。

異なるコード・ページを使用するデータベース間でデータを複製する予定である場合は、「DB2 管理ガイド」をチェックして、持っているコード・ページが互換性のあるものかどうかを判別してください。たとえば、DB2 for UNIX または DB2 for Windows を使用している場合は、文字データの変換に関するセクションを参照してください。

使用しているデータベースが互換コード・ページを持っていることを検査したら、次にそれらのデータベース間でコード・ページの使用方法に違いがあるかどうかを判別してください。たとえば、あるデータベース製品では 1 つの表内のそれぞれの列ごとに異なる

るコード・ページを指定できるとします。一方、別のデータベース製品では、列単位での異なるコード・ページの指定は許可されず、コード・ページはデータベース・レベルでのみ指定する必要があるとします。最初の製品の複数のコード・ページが指定されている表は、2 番目の製品の単一データベースには複製できません。したがって、各データベースによるコード・ページの処理方法は、使用している環境内の各種データベース間で必ず正常にデータがレプリケーションされるようにするためにはどのようにレプリケーションをセットアップする必要があるのかということに影響します。

レプリケーション用の各国語サポート (NLS) の構成

レプリケーション用の NLS 構成は、システム間のデータベース接続をセットアップする際に定義されます。ただし、キャプチャー・プログラムを UNIX または Windows プラットフォームで実行している場合は、キャプチャー・プログラムはデータをそこから取り込んでいるデータベースと同じコード・ページを使用する必要があります。キャプチャー・プログラムがその同じコード・ページを使用していない場合は、DB2CODEPAGE と呼ばれる DB2 環境変数または登録変数を設定する必要があります。

DB2 は、アプリケーションのコード・ページをそのアプリケーションが実行されているアクティブ環境から導き出します。通常、DB2CODEPAGE が設定されていない場合は、コード・ページはオペレーティング・システムによって指定される言語 ID から導き出されます。ほとんどの場合、データベースの作成時にデフォルトのコード・ページを使用したのであれば、この値はキャプチャー・プログラムにとって正しい値です。しかし、データベースの作成時にデフォルトのコード・ページ以外のコード・ページを明示的に指定した場合は、キャプチャー・プログラムのために DB2CODEPAGE 変数を設定する必要があります。そうしないと、キャプチャー・プログラムがデータを CD 表に挿入する際にデータが正しく変換されない可能性があります。DB2CODEPAGE 変数に対して使用する値は、CREATE DATABASE ステートメントで指定する値と同じでなければなりません。DB2CODEPAGE 変数の設定については、「DB2 管理ガイド」を参照してください。

第 2 章 レプリケーションの設定

データを複製するためには、その前に環境をセットアップする必要があります。

この章は、以下のセクションを含んでいます。

- 『レプリケーション・サーバーに対するアクセスのコントロール』
- 20 ページの『レプリケーション用のユーザー ID の許可』
- 26 ページの『レプリケーション用のユーザー ID およびパスワードの保管 (UNIX、Windows)』
- 27 ページの『レプリケーション・コントロール表のセットアップ』
- 30 ページの『レプリケーション・プログラムのセットアップ』
- 37 ページの『ジャーナルのセットアップ (OS/400)』

レプリケーション・サーバーに対するアクセスのコントロール

ほとんどのレプリケーション環境では、データは複数のサーバーに分散されています。このような環境を使用している場合は、レプリケーション・プログラムがすべてのサーバーに確実に接続できるようにしておく必要があります。サーバー間の接続を実現するには正しいソフトウェアがインストールされていなければならない、さらにサーバー間の接続を構成する必要があります。非 DB2 リレーショナル・データベースへのレプリケーションを行う場合は、連合サーバーおよび関連付けられた接続も構成する必要があります。

レプリケーションのための接続要件

アプライ・プログラム、レプリケーション・センター、またはレプリケーション・コマンドを実行するワークステーションは、必ずソース・サーバー、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、およびターゲット・サーバーのデータベースに接続できなければなりません。

レプリケーション・アラート・モニターを使用する場合、これを実行するワークステーションは、モニター・コントロール・サーバーとこのモニターがモニターするあらゆるサーバーに接続できなければなりません。レプリケーション・センターを使用してモニターをセットアップする場合は、レプリケーション・センターがモニター・コントロール・サーバーに接続できることを確認してください。

レプリケーションの設計で、ソース・データベースとは異なるサーバーでデータのステージングが関係しているなら、さまざまなサーバー間の通信について慎重に考慮する必

必要があります。エミュレーション層の数、LAN ブリッジの数、およびルーター・リンクの数については、どれもレプリケーションのパフォーマンスに影響を与えるものなので、必ず制限してください。

データベースがネットワークに接続される場合のコネクティビティーは、接続されるプラットフォームによって異なります。

非 DB2 リレーショナル・サーバーへの接続

非 DB2 リレーショナル・サーバーとの間で相互にデータのレプリケーションを行う場合は、その非 DB2 リレーショナル・サーバーにアクセスして接続できなければなりません。

非 DB2 リレーショナル・ソース・サーバーからのレプリケーションを試行する前に、使用する連合サーバーと連合データベースをセットアップする必要があります。セットアップには以下の 3 つのメイン・ステップがあります。

1. ラッパーを定義して、DB2 データベースが他の非 DB2 リレーショナル・データベースにアクセスできるようにする。
2. サーバー・マッピングを使用して非 DB2 リレーショナル・データベースを定義する。
3. DB2 データベースへの接続に使用されるユーザー ID とパスワードの組み合わせが非 DB2 リレーショナル・データベースへのアクセスに使用されるものとは異なる場合は、ユーザー・マッピングを作成する必要がある。

「*DB2 Federated Systems Guide*」(GC27-1224) の説明に従って、使用する環境が正しく構成されていることを確認してください。

UNIX または Windows サーバーからの z/OS または iSeries サーバーへの接続

すべてのリモート・サーバーに接続できることを確認してください。z/OS または OS/400 システムと Windows または UNIX システムとの間の接続を構成する方法は、「*DB2 Connect Quick Beginnings*」を参照してください。

プロシージャ:

DB2 for Windows ワークステーションから iSeries サーバーに接続する場合

1. DB2 for Windows クライアントから iSeries サーバーに接続する前に、使用するワークステーションが正しくセットアップされていることを確認する。
 - DB2 Universal Database または DB2 クライアント・アプリケーション・イネーブラー (CAE) クライアントがワークステーションにインストールされていなければならない。
 - TCP/IP がワークステーションにセットアップされている必要があること。
2. iSeries サーバーにログオンして、リレーショナル・データベースを探します。

- a. 接続先となる iSeries サーバーにログオンします。
- b. **dsprdbdire** コマンドをサブミットしてから、***LOCAL** に **local** と指定します。
- c. 出力の中から、リレーショナル・データベースの名前を探します。たとえば、以下の出力では、データベースの名前は **DB2400E** です。

```

MYDBOS2          9.112.14.67
RCHASDPD         RCHASDPD
DB2400E         *LOCAL
RCHASLJN         RCHASLJN

```

3. OS/400 データベースを DB2 for Windows のカタログに入れる。
 - a. Windows ワークステーションで、「スタート」->「プログラム」->「**IBM DB2**」->「**コマンド・ウィンドウ (Command Window)**」をクリックします。DB2 CLP コマンド・ウィンドウが開きます。
 - b. コマンド・ウィンドウで、以下の 3 つのコマンドを正確な順序で入力します。

```
db2 catalog tcpip node server_name remote server_name server 446 system
server_name ostype OS400
```

```
db2 catalog dcs database rdb_name AS rdb_name
```

```
db2 catalog database rdb_name AS rdb_name at node server_name
authentication dcs
```

ここで、*server_name* は iSeries システムの TCP/IP ホスト名で、*rdb_name* は 2 (18 ページ) で説明されている iSeries リレーショナル・データベースの名前です。

4. コマンド・ウィンドウで、以下のコマンドを実行します。

```
db2 terminate
```

5. iSeries システムにログオンするために使用する iSeries ユーザー・プロファイルが CCSID37 を使用していることを確認します。

- a. iSeries システムにログオンします。
- b. 以下のコマンドを入力します。ここで、*user* はユーザー・プロファイルです。

```
CHGUSRPRF USRPRF (user) CCSID(37)
```

- c. 次のように入力して、DDM サーバーが iSeries システムで始動していることを確認します。

```
STRTCPSVR SERVER(*DDM)
```

6. DB2 for Windows NT および DB2 for iSeries が接続されていることを確認する。

```
db2 connect to rdb_name user user_name using password
```

レプリケーション用のユーザー ID の許可

DB2 および非 DB2 のリレーショナル・サーバーに置かれているデータにアクセスする必要がある場合は、以下の許可要件が満たされていることを確認してください。

- 『管理の許可要件』
- 22 ページの『キャプチャー・プログラムの許可要件』
- 23 ページの『非 DB2 リレーショナル・データベースでのキャプチャー・トリガーの場合の許可要件』
- 24 ページの『アプライ・プログラムの許可要件』
- 26 ページの『レプリケーション・アラート・モニターの場合の許可要件』

管理の許可要件

レプリケーション・センターを使用してレプリケーションを管理します。(詳しくは、259 ページの『第 14 章 DB2 レプリケーション・センターの使用』を参照してください。) レプリケーション環境が OS/400 オペレーティング・システム上のみある場合は、OS/400 システム・コマンドを使用してレプリケーションを管理できます。(詳しくは、365 ページの『第 18 章 レプリケーション用のシステム・コマンド (OS/400)』を参照してください。) レプリケーションを管理するためには、レプリケーション構成に関与するすべてのデータベースに対するユーザー ID を少なくとも 1 つは持っている必要があります。さらにそのユーザー ID にはレプリケーションをセットアップする権限がなければなりません。使用するユーザー ID は、すべてのシステムで同じである必要はありません。ただし、同じであった方が楽です。レプリケーションのセットアップには、オブジェクト (コントロール表や表スペースなど) の作成、プランのバインディング (UNIX、Windows、および z/OS の場合)、SQL パッケージの作成 (OS/400 の場合)、ならびに表、登録、およびサブスクリプション・セットを作成するための生成済み SQL の実行が含まれます。使用するレプリケーション環境内のすべてのサーバーに対して同一の許可ユーザー ID を使用するか、あるいはそれぞれのサーバーごとに別々の許可ユーザー ID を使用するかは任意です。

UNIX、Windows、および z/OS の場合の要件

レプリケーションのセットアップに使用するユーザー ID で以下のタスクを実行できることを確認してください。

- すべてのサーバー (ソース・サーバー、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、モニター・コントロール・サーバー、およびターゲット・サーバー) への接続
- ソース・サーバー、キャプチャー・コントロール・サーバー、モニター・コントロール・サーバー、およびターゲット・サーバーに置かれているカタログ表からの選択

- ソース・サーバー、モニター・コントロール・サーバー、キャプチャー・コントロール・サーバー、およびアプライ・コントロール・サーバーでの表 (レプリケーション・コントロール表も含む)、表スペース、およびビューの作成
- ターゲット・サーバーでの表および表スペースの作成 (新しいターゲット表の作成に DB2 レプリケーション・プログラムを使用する場合) (既存の表をターゲットとして使用する場合は必須ではありません)
- レプリケーションに関与するそれぞれの DB2 データベース (ソース・サーバー、ターゲット・サーバー、モニター・コントロール・サーバー、およびアプライ・コントロール・サーバーも含む) でのプランのバインドまたはパッケージの作成
- 共有ライブラリーを使用したストアド・プロシージャの作成およびストアド・プロシージャの呼び出し (UNIX と Windows のみ)

非 DB2 リレーショナル・データベースの場合は、使用するユーザー ID で以下のアクションを実行できなければなりません。

- 表を作成します。
- ソース表およびコントロール表に対するキャプチャー・トリガーの作成
- プロシージャの作成
- DB2 連合データベースでのニックネームの作成
- シーケンスの作成 (Oracle データベースの場合のみ)
- カタログ表からの選択

ほとんどのレプリケーション管理者は DBADM 特権または SYSADM 特権を持っています。DB2 for z/OS の場合、レプリケーション管理者は、少なくともカタログからの選択を許可されている必要があり、さらに索引作成特権も含め、ASN スキーマを使用した表の作成、およびソース表の特性を持つ CD 表とターゲット表の作成を行うために必要なすべての特権を持っている必要があります。

OS/400 の場合の要件

レプリケーションのセットアップに使用するユーザー ID で以下のタスクを実行できることを確認してください。

- すべてのサーバー (ソース・サーバー、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、モニター・コントロール・サーバー、およびターゲット・サーバー) への接続
- ソース・サーバー、キャプチャー・コントロール・サーバー、モニター・コントロール・サーバー、およびターゲット・サーバーに置かれているカタログ表からの選択

- ・ソース・サーバー、モニター・コントロール・サーバー、キャプチャー・コントロール・サーバー、およびアプライ・コントロール・サーバーでの表 (レプリケーション・コントロール表も含む) とビューの作成
- ・ターゲット・サーバーでの表の作成 (新しいターゲット表の作成に DB2 レプリケーション・プログラムを使用する場合) (既存の表をターゲットとして使用する場合は必須ではありません)
- ・レプリケーションに関与するそれぞれの DB2 データベース (ソース・サーバー、ターゲット・サーバー、モニター・コントロール・サーバー、およびアプライ・コントロール・サーバーも含む) でのプランのバインドまたはパッケージの作成

ほとんどのレプリケーション管理者は DBADM 特権または SYSADM 特権を持っています。

DPR 権限付与 (**GRTDPRAUT**) コマンドは、ユーザーがソースの登録、それらのソースへのサブスクライブ、およびコントロール表の作成を行うのを許可するために使用します。OS/400 システム間でのみレプリケーションを行う場合は、すべてのサーバーに対して同じユーザー ID を使用する必要があります。コマンド構文およびパラメーター記述の詳細については、415 ページの『GRTDPRAUT: ユーザーの許可 (OS/400)』を参照してください。

DPR 権限付与 (**GRTDPRAUT**) コマンドがマシンにインストールされていない場合は、オブジェクト権限付与 (**GRTOBJAUT**) コマンドを使用する必要があります。

キャプチャー・プログラムの許可要件

キャプチャー・プログラムを実行するユーザー ID は、DB2 システム・カタログにアクセスできなければなりません。また、キャプチャー・コントロール・サーバー上のすべてのレプリケーション・コントロール表へのアクセスと更新が可能で、さらにキャプチャー・プログラム・パッケージを実行できなければなりません。レプリケーション管理者のユーザー ID を使用してキャプチャー・プログラムを実行できますが、これは要件ではありません。

UNIX および Windows の場合の要件

キャプチャー・プログラムを実行するユーザー ID が以下の権限と特権を持っていることを確認してください。

- ・DBADM または SYSADM 権限。
- ・キャプチャー・パス・ディレクトリーに対する WRITE 特権。これは、キャプチャー・プログラムはキャプチャー・プログラムの始動時に指定された **capture_path** ディレクトリーに診断ファイルを作成するためです。

z/OS の場合の要件

キャプチャー・プログラムを実行するために使用するユーザー ID は、USS にアクセスできるものとして登録する必要があります。

また、キャプチャー・ロード・ライブラリーに APF 許可が与えられていること、およびキャプチャー・プログラムを実行するユーザー ID が以下の権限と特権を持っていることも確認してください。

- キャプチャー CONTROL サーバー上のすべてのレプリケーション表に対する SELECT、UPDATE、INSERT、および DELETE 特権。(これらの表のリストについては、486 ページの『キャプチャー・コントロール・サーバーで使用される表のリスト』を参照。)
- DB2 カタログ (SYSIBM.SYSTABLES および SYSIBM.SYSCOLUMNS) に対する SELECT 特権。
- TRACE 特権。
- MONITOR1 および MONITOR2 特権。
- キャプチャー・プログラム・パッケージに対する EXECUTE 特権。

また、そのユーザー ID にキャプチャー・パス・ディレクトリーに対する WRITE アクセス権があること (USS)、または高位修飾子が付けられていること (z/OS) を確認してください。キャプチャー・プログラムを USS シェルで実行するためには、STEPLIB システム変数が設定されていて、さらにこの変数にキャプチャー・ロード・ライブラリーが組み込まれている必要があります。使用する PATH には HFS パス (/usr/lpp/db2repl_8_1/bin) が含まれていなければなりません。

OS/400 の場合の要件

DPR 権限付与 (GRTDPRAUT) コマンドは、ユーザーにローカル・システムでのキャプチャー・プログラムの実行を許可するために使用します。コマンド構文およびパラメーター記述の詳細については、415 ページの『GRTDPRAUT: ユーザーの許可 (OS/400)』を参照してください。OS/400 システム間でのみレプリケーションを行う場合は、すべてのサーバーに対して同じユーザー ID を使用する必要があります。GRTDPRAUT コマンドがマシンにインストールされていない場合は、オブジェクト権限付与 (GRTOBJAUT) コマンドを使用する必要があります。

非 DB2 リレーショナル・データベースでのキャプチャー・トリガーの場合の許可要件

非 DB2 RDBMS から複製する場合は、ソースからの変更のキャプチャーにはキャプチャー・トリガーが使用されます。リモート・ソース表を変更するリモート・ユーザー ID (たとえば、ユーザー・アプリケーションからの) には、CCD 表への挿入を行う権限が必要です。ほとんどの場合、INSERT、UPDATE、または DELETE トリガーを実行するのに明示的な権限は必要ありません。なぜなら、表に対してトリガーが定義されると、

それ以降トリガーは INSERT、UPDATE、または DELETE を実行中のアプリケーションにとって透過的に実行されるからです。Informix データベースの場合は、登録済みのソース表に対して INSERT、UPDATE、および DELETE アクションを実行するリモート・ユーザー ID には EXECUTE PROCEDURE 特権が必要です。

アプライ・プログラムの許可要件

アプライ・プログラムを始動および操作するユーザー ID でプログラムの実行とパスワード・ファイルの読み取りができ、さらにそのユーザー ID がアプライ・パス・ディレクトリーに対する WRITE 特権を持っていることを確認してください。また、パスワード・ファイル内のユーザー ID は他のサーバーへの接続に使用されるため、それらのユーザー ID には以下のタスクを実行できるだけの十分な特権が与えられていなければなりません。

z/OS の場合の要件

アプライ・プログラムを実行するユーザー ID が以下の権限と特権を持っていることを確認してください。

- アプライ・プログラムを実行するために使用するユーザー ID は、USS にアクセスできるものとして登録する必要があります。ロード・ライブラリーに APF 許可が必要となるのは、アプライ・プログラムが ARM の指定付きで登録される場合だけです。アプライ・プログラムを USS シェルで実行するためには、STEPLIB システム変数が設定されていて、さらにこの変数にアプライ・ロード・ライブラリーが組み込まれている必要があります。使用する PATH には HFS パス (/usr/lpp/db2repl_8_1/bin) が含まれていなければなりません。
- キャプチャー・コントロール・サーバー、ターゲット・サーバー、およびアプライ・コントロール・サーバーでのアプライ・プログラム・プランの実行

UNIX および Windows の場合の要件

アプライ・プログラムを実行するユーザー ID が以下の権限と特権を持っていることを確認してください。

- レプリケーション・ソース表 (関連した CD 表および CCD 表も含む) へのアクセス

使用するソース表が非 DB2 リレーショナル・データベース管理システム上にある場合は、ユーザー ID は DB2 連合データベースおよび非 DB2 リレーショナル・データベースの両方において、連合データベースで定義されているニックネームを使用してソース表にアクセスできるだけの十分な特権を持っている必要があります。

- レプリケーション・ターゲット表に対するアクセスおよび更新

アプライ・ユーザー ID は、ターゲット表に対する ALTER 特権を持っている必要があります。

使用するターゲット表が非 DB2 RDBMS 上にある場合は、ユーザー ID は DB2 連合データベースおよび 非 DB2 リレーショナル・データベースにおいて、連合データベースで定義されているニックネームを使用してターゲット表にアクセスし、それらを ALTER できるだけの十分な特権を持っている必要があります。

- DB2 レプリケーション・プログラムによって生成されて、キャプチャー・コントロール・サーバーおよびアプライ・コントロール・サーバーでビルドされたすべてのコントロール表に対するアクセスと更新
- アプライ・プログラムによって使用される任意のパスワード・ファイルの読み取り

非 DB2 リレーショナル・データベース管理システムの場合の要件

使用するソース表が非 DB2 リレーショナル・データベース管理システム上にある場合は、ALTER されたデータを非 DB2 リレーショナル・ターゲットにプッシュしたりそこからデータをプルするユーザー ID は、DB2 連合データベースおよび 非 DB2 リレーショナル・データベースにおいて十分な特権を持っている必要があります。

非 DB2 リレーショナル・ターゲットの場合、アプライ・プログラムを実行中のユーザー ID には、DB2 連合データベース上のニックネームに対して WRITE を実行できる特権と、ユーザー・マッピングを介して実際の非 DB2 ターゲットへの WRITE を実行できる特権が必要です。

非 DB2 リレーショナル・ソースの場合、アプライ・プログラムを実行中の ID には以下の特権が必要です。

- DB2 連合データベース上のニックネームに対して READ と WRITE を実行できる特権、およびユーザー・マッピングを介してキャプチャー CONTROL 表に対し READ と WRITE を実行できる特権
- DB2 連合データベース上のニックネームからの READ を実行できる特権、およびユーザー・マッピングを介して非 DB2 サーバー上の実際の CCD 表からの READ を実行できる特権
- DB2 連合データベース上のニックネームからの READ を実行できる特権、およびユーザー・マッピングを介して非 DB2 サーバー上の実際のソース表からの READ を実行できる特権

OS/400 の場合の要件

DPR 権限付与 (GRTPRAUT) コマンドは、ユーザーにローカル・システムでのアプライ・プログラムの実行を許可するために使用します。OS/400 システム間でのみレプリケーションを行う場合は、すべてのサーバーに対して同じユーザー ID を使用する必要があります。GRTPRAUT コマンドがマシンにインストールされていない場合は、オブジェクト権限付与 (GRTOBJAUT) コマ

ンドを使用する必要があります。 コマンド構文およびパラメーター記述の詳細については、415 ページの『GRTDPRAUT: ユーザーの許可 (OS/400)』を参照してください。

使用するレプリケーション環境内のそれぞれのサーバーごとに異なるユーザー ID を使用できます。

レプリケーション・アラート・モニターの場合の許可要件

UNIX および Windows の場合の要件

レプリケーション・アラート・モニターを始動するユーザー ID が、モニター・コントロール表が置かれているモニター・コントロール・サーバー、およびモニターしているコントロール表を含んでいるサーバーでの有効なログオン ID であることを確認してください。また、レプリケーション・アラート・モニターを実行するユーザー ID が以下の権限を持っていることも確認してください。

- モニター CONTROL サーバー上のモニター CONTROL 表に対する SELECT、UPDATE、INSERT、および DELETE 特権 (これらの表のリストについては、490 ページの『モニター・コントロール・サーバーで使用される表のリスト』を参照。)
- モニターするサーバー上にあるキャプチャー・コントロール表およびアプライ・コントロール表に対する SELECT 権限
- BINDADD 権限 (モニター・パッケージに対して自動バインド機能を使用する場合にのみ必要)

UNIX および Windows では、レプリケーション・アラート・モニターが診断ファイルを保管する **monitor_path** ディレクトリーに対する WRITE 特権と、レプリケーション・アラート・モニターによって使用されるパスワード・ファイルに対する読み取りアクセスが必要です。

レプリケーション用のユーザー ID およびパスワードの保管 (UNIX、Windows)

レプリケーション環境が複数のサーバーに分散されていない場合は、ユーザー ID とパスワードを保管する必要はありません。しかし、ほとんどのレプリケーション環境では、データは複数のサーバーに分散されています。そのような環境がある場合は、データベースへの接続の試行時に有効なユーザー ID とパスワードを入力して、DB2 がユーザーの一致を検査できるようにする必要があります。レプリケーション・センターの場合とそれ以外のレプリケーション・プログラムの場合とでは、パスワード情報の保管方法が異なります。

asnpwd コマンドは、アプライ・プログラム、レプリケーション・アラート・モニター、およびレプリケーション・アナライザーがリモート・サーバー上のデータにアクセスできるように、パスワード・ファイルを作成して保守するために使用します (キャプ

チャー・プログラムはパスワード・ファイルを必要としません)。パスワード・ファイル内の情報は、機密性を確保するために暗号化されます。コマンド構文およびパラメーター記述の詳細については、352 ページの『asnpwd: パスワード・ファイルの保守 (UNIX および Windows)』を参照してください。

レプリケーション・センターの場合のパスワード要件は、レプリケーション・センターのヘルプおよび 264 ページの『レプリケーション・センターのユーザー ID およびパスワードの管理』を参照してください。

レプリケーション・コントロール表のセットアップ

レプリケーションに使用されるコントロール表を作成できます。

- 『コントロール表の作成 (UNIX、Windows)』
- 『コントロール表の作成 (z/OS)』
- 28 ページの『コントロール表の作成 (OS/400)』
- 28 ページの『非 DB2 リレーショナル・ソース用のコントロール表の作成』
- 29 ページの『キャプチャー・コントロール表のセットを複数作成する』

コントロール表の作成 (UNIX、Windows)

レプリケーション・センターを使用して、UNIX および Windows でのキャプチャー・プログラム用とアプライ・プログラム用のレプリケーション・コントロール表を作成します。レプリケーション・コントロール表の作成時にコントロール表の作成方法をカスタマイズしない場合は、2 つの表スペース (1 つは UOW 表用、もう 1 つはその他のコントロール表用) が作成されます。それらのデフォルトのレプリケーション表スペースを使用したくない場合は、既存の表スペースを指定するか、新しい表スペースを作成するか、または現行の DB2 デフォルト表スペースを使用できます。

UNIX オペレーティング・システム用のプロファイルと Windows オペレーティング・システム用のプロファイルを別々に作成して、このタイプのシステム用のコントロール表を作成するときに、当該のタイプで使用すべきデフォルトが識別されるようにできます。それらのコントロール表に対してプロファイルを設定してしまえば、それ以降はコントロール表のセットを作成するたびにプロファイルを設定する必要がなくなります。ただし、それらのデフォルトはコントロール表の作成時にオーバーライドできます。また、プロファイルもいつでも変更できます。ただし、その変更の影響を受けるのは、プロファイルの変更後に作成したコントロール表だけです。

コントロール表の作成 (z/OS)

レプリケーション・センターを使用して、z/OS でのレプリケーション・コントロール表を作成します。z/OS プラットフォーム用のプロファイルを作成して、このタイプのシステム用のコントロール表を作成するときに使用すべきデフォルトが識別されるようにできます。それらのコントロール表に対してプロファイルを設定してしまえば、それ以

降はコントロール表のセットを作成するたびにプロファイルを設定する必要がなくなります。ただし、それらのデフォルトはコントロール表の作成時にオーバーライドできません。また、プロファイルもいつでも変更できます。ただし、その変更の影響を受けるのは、プロファイルの変更後に作成したコントロール表だけです。

コントロール表の作成 (OS/400)

レプリケーション・コントロール表は、DB2 DataPropagator for iSeries のインストール時に自動的に作成されます。それらの表は、DataPropagator (DPROPR) のデフォルト・スキーマ (ASN と呼ばれる) 内に作成されます (まだ存在していない場合)。

新しいキャプチャー・スキーマを使用してキャプチャー・コントロール表の新しいセットを作成できます。スキーマは、最大 25 個まで作成できます。29 ページの『キャプチャー・コントロール表のセットを複数作成する』の説明に従って、DPR 表の作成 (CRTDPRTBL) コマンドを使用してください。また、CRTDPRTBL コマンドは、レプリケーション・コントロール表が誤って削除または破壊された場合にも使用できます。このコマンドの詳細については、408 ページの『CRTDPRTBL: レプリケーション・コントロール表の作成 (OS/400)』を参照してください。

重要: OS/400 では、コントロール表は必ず CRTDPRTBL コマンドを使用して作成するようにしてください。レプリケーション・センターは OS/400 用のコントロール表の作成をサポートしていません。

非 DB2 リレーショナル・ソース用のコントロール表の作成

Informix のような非 DB2 RDBMS から のレプリケーションを行う場合は、DB2 からのレプリケーションの場合とまったく同じように、レプリケーション・センターを使用してコントロール表を作成する必要があります。このようなタイプのソースの場合、レプリケーション・センターは非 DB2 リレーショナル・データベースに以下のキャプチャー・コントロール表を作成します。

- プルーニング・コントロール表 (IBMSNAP_PRUNCNTL)
- プルーニング・セット表 (IBMSNAP_PRUNE_SET)
- 登録同期表 (IBMSNAP_REG_SYNCH)
- 登録表 (IBMSNAP_REGISTER)
- 順序付け表 (IBMSNAP_SEQTABLE) (Informix の場合のみ)
- シグナル表 (IBMSNAP_SIGNAL)

連合データベース内に順序付け表 (IBMSNAP_SEQTABLE) 以外のすべての表のニックネームが作成されます。(この順序付け表は、Informix トリガーだけが使用します。アプライ・プログラムはこれを使用しません。) シグナル表 (IBMSNAP_SIGNAL) および登録同期表 (IBMSNAP_REG_SYNCH) にトリガーが自動的に作成されます。

重要: IBMSNAP_SIGNAL 表および IBMSNAP_REG_SYNCH 表に作成されたトリガーは、除去または変更をしないでください。

キャプチャー・コントロール表のセットを複数作成する

1 台のサーバー上で複数のキャプチャー・プログラムを使用する場合は、キャプチャー・コントロール表のセットを複数作成して、それぞれの表セットが必ず固有のキャプチャー・スキーマを持つようにする必要があります。このスキーマにより、ある表のセットを使用するキャプチャー・プログラムが識別されます。複数のキャプチャー・スキーマを使用すると、複数のキャプチャー・プログラムを並行実行できます。

次の場合には、複数のキャプチャー・プログラムを実行しなければならない可能性があります。

- 待ち時間が短い表は他の表とは別に処理するという方法でパフォーマンスを最適化する場合。待ち時間が短い表がある場合は、それらの表の複製はそれら専用のキャプチャー・プログラムを使用して行うのが望ましいことがあります。この方法を使用すると、それらの表には別のランタイム優先順位が与えられます。また、キャプチャー・プログラムのパラメーター（ブルーニング・インターバルやモニター・インターバルなど）を、それらの表の短い待ち時間に合うように設定することも可能です。
- 可能なときにはキャプチャー・スループットを上げる場合。これは、複数の CPU を持つソース環境では大いに益になる可能性があります。スループット増加のトレードオフは、ログ・リーダーが複数存在することに関連した CPU オーバーヘッドの増加です。

同一の連合データベース内で複数の非 DB2 ソース・データベースからのレプリケーションを実行する場合は、キャプチャー・コントロール表のセットを複数作成して、各セットに独自のスキーマを持たせる必要があります。あるいはその代わりに、複数の連合データベースを別個に使用しても構いません。そのようにすると、各サーバーのキャプチャー・コントロール表でデフォルトの ASN スキーマを使用できます。

z/OS システムでは、UNICODE コード化スキームと EBCDIC コード化スキームを別々に使用して作業したい場合、または 1 つのサブシステムでキャプチャー・プログラムのインスタンスを複数実行したい場合は、複数のキャプチャー・スキーマを使用できます。コントロール表の作成については、27 ページの『コントロール表の作成 (z/OS)』を参照してください。

OS/400 システムでは、DPR 表の作成 (**CRTDPRTBL**) コマンドを使用してキャプチャー・コントロール表の追加のセットを作成します。その際には、**CAPCTLLIB** パラメーターを使用してスキーマ名を指定します。このコマンドの詳細については、408 ページの『CRTDPRTBL: レプリケーション・コントロール表の作成 (OS/400)』を参照してください。

レプリケーション・プログラムのセットアップ

以下のセクションでは、環境内のサーバーに対してレプリケーション・プログラムをセットアップするステップについて説明しています。

- 『レプリケーション・プログラムのセットアップ (UNIX、Windows)』
- 35 ページの『キャプチャー・プログラムおよびアプライ・プログラムのセットアップ (OS/400)』
- 36 ページの『レプリケーション・プログラムのセットアップ (z/OS)』

レプリケーション・プログラムのセットアップ (UNIX、Windows)

以下の説明を読んで、レプリケーション・プログラムをセットアップしてください。

- 『レプリケーション・プログラムのための環境変数の設定 (UNIX、Windows)』
- 31 ページの『キャプチャー・プログラムを実行するための DB2 データベースの準備 (UNIX、Windows)』
- 32 ページの『オプション：キャプチャー・プログラム・パッケージのバインディング (UNIX、Windows)』
- 32 ページの『オプション：アプライ・プログラム・パッケージのバインディング (UNIX、Windows)』
- 33 ページの『オプション：レプリケーション・アラート・モニター・プログラム・パッケージのバインディング (UNIX、Windows)』

レプリケーション・プログラムのための環境変数の設定 (UNIX、Windows)

キャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニター・プログラムを始動する前と停止する前、およびレプリケーション・センターまたはレプリケーション・システム・コマンドを使用する前に、環境変数を設定する必要があります。

プロシージャー:

環境変数を設定するには、以下のようにします。

1. 次のように入力して、DB2 インスタンス名 (DB2INSTANCE) の環境変数を設定します。

Windows の場合

```
SET DB2INSTANCE=db2_instance_name
```

UNIX の場合

```
export DB2INSTANCE=db2_instance_name
```

2. デフォルトのコード・ページ値以外のコード・ページを使用してソース・データベースを作成した場合は、DB2CODEPAGE 環境変数をそのコード・ページに設定する。15 ページの『レプリケーション用の各国語サポート (NLS) の構成』を参照してください。⁴
3. オプション: 環境変数 DB2DBDFT をソース・サーバーに設定します。
4. **UNIX の場合:** 使用しているシステムに固有のライブラリー・パス・システム変数および実行可能モジュール・パス・システム変数に、レプリケーション用のライブラリーと実行可能モジュールがインストールされているディレクトリーが組み込まれていることを確認する。

キャプチャー・プログラムを実行するための DB2 データベースの準備 (UNIX、Windows)

プロシージャー:

キャプチャー・プログラムを実行するために DB2 データベースを準備するには、次のようにします。

1. 次のように入力して、キャプチャー・コントロール・サーバーのデータベースに接続する。

```
db2 connect to database
```

ここで、*database* はキャプチャー・コントロール・サーバーのデータベースです。

2. キャプチャー・コントロール・サーバーのデータベースを、ロールフォワード・リカバーのために準備する。これを行うには、**update database configuration** (データベース構成の更新) コマンド (ログ保存リカバー) と **backup database** (データベースのバックアップ) コマンドを発行します。インストール要件に基づいて構成値を大きくする必要がある場合があります。多数の行または非常に大きな行を使用するトランザクションの場合は、CAPPARMS メモリー限度パラメーターの値を大きくすることが推奨されます。多くの大規模ワークステーション・シナリオの場合、適切なデータベース構成値は次のとおりです: APPLHEAPSZ 1000、LOGFILSIZ 4000、LOGPRIMARY 8、LOGSECOND 40、DBHEAP 1000、LOGBUFSZ 16、MAXAPPLS 200。

4. キャプチャーは、データを取り込んでいるデータベースのコード・ページと同じコード・ページで実行されている必要があります。DB2 は、キャプチャーが実行されているアクティブ環境からキャプチャー・コード・ページを導き出します。DB2CODEPAGE が設定されていない場合は、DB2 はコード・ページ値をオペレーティング・システムから取得します。データベースの作成時にデフォルトのコード・ページを使用したのであれば、オペレーティング・システムから取得した値はキャプチャーにとって正しいものです。

オプション：キャプチャー・プログラム・パッケージのバインディング (UNIX、Windows)

以下のステップはオプションです。なぜなら、UNIX および Windows では、キャプチャー・プログラムは実行時に自動的に BIND されるからです。

プロシージャー:

キャプチャー・プログラム・パッケージをバインドするには、次のようにします。

1. 次のように入力して、キャプチャー・コントロール・サーバーのデータベースに接続する。

```
db2 connect to database
```

ここで、*database* はキャプチャー・コントロール・サーバーのデータベースです。

2. キャプチャー・プログラムのバインド・ファイルがあるディレクトリーに変更します。

Windows:

```
drive:¥sqllib¥bnd
```

UNIX: *db2homedir/sqllib/bnd*

ここで、*db2homedir* は DB2 インスタンスのホーム・ディレクトリーです。

3. 次のようなコマンドを入力して、キャプチャー・プログラム・パッケージを作成してソース・サーバー・データベースにバインドします。

```
db2 bind @capture.lst isolation ur blocking all
```

ただし *ur* は、パフォーマンスを向上するための非コミット読み取り形式のリストを指定します。

これらのコマンドは、パッケージを作成します。その名前は、*capture.lst* ファイル内で見つけることができます。

オプション：アプライ・プログラム・パッケージのバインディング (UNIX、Windows)

UNIX および Windows では、アプライ・プログラムは実行時に自動的にバインドされます。したがって、これらのオペレーティング・システムでは、以下のステップはオプションです。

プロシージャー:

アプライ・プログラム・パッケージをバインドするには、次のようにします。

1. アプライ・プログラムのバインド・ファイルがあるディレクトリーに変更します。

Windows:

```
drive:¥sqllib¥bnd
```

UNIX: `db2homedir/sqllib/bnd`

ここで、`db2homedir` は DB2 インスタンスのホーム・ディレクトリーです。

2. アプライ・プログラムが接続するそれぞれのソース・サーバー、ターゲット・サーバー、キャプチャー・コントロール・サーバー、およびアプライ・コントロール・サーバーごとに、以下のステップを実行する。
 - a. 次のように入力して、データベースに接続します。

```
db2 connect to database
```

ここで、`database` はソース・サーバー、ターゲット・サーバー、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーです。データベースがリモート・データベースとしてカタログされている場合は、**db2 connect to** コマンドでユーザー ID とパスワードを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。

```
db2 connect to database user userid using password
```

- b. 次の 2 つのコマンドを両方入力して、Applyプログラム・パッケージを作成してデータベースにバインドします。

```
db2 bind @applycs.lst isolation cs blocking all grant public
```

```
db2 bind @applyur.lst isolation ur blocking all grant public
```

ただし、`cs` はカーソル固定形式でリストを指定するのに対し、`ur` は非コミット読み取り形式でリストを指定します。

これらのコマンドは、パッケージを作成します。その名前は、`applycs.lst` および `applyur.lst` ファイル内で見つけることができます。

オプション：レプリケーション・アラート・モニター・プログラム・パッケージのバインディング (UNIX、Windows)

パッケージを BIND するための以下のステップはオプションにすぎません。レプリケーション・アラート・モニター・パッケージは、実行時に自動的にバインドされます。オプションを指定したい場合、またはすべての BIND 処理が正常に完了したことをチェックしたい場合は、以下のステップを実行してください。

プロシージャ:

レプリケーション・アラート・モニター・プログラム・パッケージをバインドするには、次のようにします。

1. レプリケーション・アラート・モニター・プログラムのバインド・ファイルが配置されているディレクトリーに変更する。

Windows:

```
drive:¥sqllib¥bnd
```

UNIX: *db2homedir/sqllib/bnd*

ここで、*db2homedir* は DB2 インスタンスのホーム・ディレクトリーです。

2. それぞれのモニター・コントロール・サーバーごとに以下のステップを実行する。
 - a. 次のように入力して、モニター・コントロール・サーバーのデータベースに接続する。

```
db2 connect to database
```

ここで、*database* はモニター・コントロール・サーバーです。データベースがリモート・データベースとしてカタログされている場合は、**db2 connect to** コマンドでユーザー ID とパスワードを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。

```
db2 connect to database user userid using password
```

- b. 以下のコマンドを入力し、レプリケーション・アラート・モニター・プログラム・パッケージを作成してデータベースにバインドする。

```
db2 bind @asnmoncs.lst isolation cs blocking all grant public
```

```
db2 bind @asnmonur.lst isolation ur blocking all grant public
```

ただし、*cs* はカーソル固定形式でリストを指定するのに対し、*ur* は非コミット読み取り形式でリストを指定します。

これらのコマンドは、パッケージを作成します。その名前は、*asnmoncs.lst* および *asnmonur.lst* ファイル内で見つけることができます。

3. 現在モニターの対象としており、レプリケーション・アラート・モニター・プログラムの接続先となっているそれぞれのサーバーごとに、以下のステップを実行する。
 - a. 次のように入力して、データベースに接続します。

```
db2 connect to database
```

ここで、*database* はモニターされているサーバーです。データベースがリモート・データベースとしてカタログされている場合は、**db2 connect to** コマンドでユーザー ID とパスワードを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。

```
db2 connect to database user userid using password
```

- b. 以下のコマンドを入力し、レプリケーション・アラート・モニター・プログラム・パッケージを作成してデータベースにバインドする。

```
db2 bind @asnmonit.lst isolation ur blocking all grant public
```

ただし *ur* は、非コミット読み取り形式のリストを指定します。

これらのコマンドは、パッケージを作成します。その名前は、*asnmonit.lst* ファイル内で見つけることができます。

キャプチャー・プログラムおよびアプライ・プログラムのセットアップ (OS/400)

アプライ・プログラムを他の非 OS/400 オペレーティング・システムのリモート・システムで使用する場合は、環境をセットアップする必要があります。以下のセクションでは、レプリケーション環境をセットアップするステップについて説明しています。

- 『リモート・システムで使用するための SQL パッケージの作成 (OS/400)』
- 36 ページの『SQL パッケージへの特権の付与』

リモート・システムで使用するための SQL パッケージの作成 (OS/400)

以下のケースでは、**CRTSQPKG** コマンドを使用してパッケージを作成する必要があります。

- リモート・ジャーナリングを使用する場合。キャプチャー・プログラムを実行中のシステムで **CRTSQPKG** コマンドを実行し、ソース表が配置されているシステムを指示してください。
- **ADDDPRSUB** または **ADDDPRSUBM** コマンドを使用してサブスクリプション・セットまたはサブスクリプション・セットのメンバーを追加する前。次のようにして、ターゲット・サーバーで **CRTSQPKG** コマンドを実行してください。
 - ソース表が別のマシン上にある場合は、そのソース表が配置されているシステムを指示する。
 - アプライ・コントロール・サーバーが別のマシン上にある場合は、そのアプライ・コントロール・サーバーを指示する。

SQL パッケージを使用すると、レプリケーション・プログラムが分散レプリケーション環境で稼働できるようになります。その場合、レプリケーションが OS/400 システム間で行われる環境であっても、OS/400 システムと他のいずれかのオペレーティング・システム (UNIX や Windows など) との間で行われる環境であっても構いません。

CRTSQPKG コマンドの使用方法は、「*DB2 Universal Database for iSeries SQL Programming*」を参照してください。

パッケージは ASN 修飾子を使って作成されます。OS/400 では、パッケージは ASN ライブラリーに作成されます。他のプラットフォームでは、パッケージは ASN スキーマに作成されます。

アプライ・プログラム用の SQL パッケージの作成: SQL パッケージを作成して、接続する必要のあるすべてのリモート・サーバーとアプライ・プログラムが相互作用できるようにする必要があります。たとえば、アプライ・プログラムがリモート・システムに接続できるようにするためには、アプライ・プログラムが実行されているシステムで以下のコマンドを実行してください。

```
CRTSQPKG PGM(QDP4/QZSNAPV2) RDB(remote_system)
```

remote_system は、アプライ・プログラムが接続する必要のあるリモート・システムでのリレーショナル・データベース入り口名です。

レプリケーション・アナライザー用の SQL パッケージの作成: SQL パッケージを作成して、キャプチャー・コントロール・サーバーやターゲット・サーバーなどといった、分析の対象としているサーバーとレプリケーション・アナライザーが相互作用できるようにする必要があります。レプリケーション・アナライザーが実行されているシステムで以下のコマンドを実行してください。

```
CRTSQLPKG PGM(QDP4/QZSNANZR) RDB(remote_system)
```

ここで、*remote_system* は分析の対象としているシステムの名前です。

レプリケーション管理コマンド用の SQL パッケージの作成: OS/400 システム間のレプリケーションでは、リモート・ジャーナルを使用する場合、このコマンドを使用してキャプチャー・プログラム用およびレプリケーション管理コマンド用のパッケージを作成する必要があります。キャプチャーが実行されているシステムで以下のコマンドを実行してください。

```
CRTSQLPKG PGM(QDP4/QZSNSQLF) RDB(source_system) OBJTYPE(*SRVPGM)
```

ここで、*source_system* はソース表が実際に存在するシステムの名前です。

SQL パッケージへの特権の付与

パッケージの作成後には、ソース・データベースに登録されているファイルにサブスクライブするすべてのユーザーに対して *EXECUTE 特権を付与する必要があります。ソース・データベースが常駐している OS/400 システムにログオンし、以下の方法のいずれかを使用してください。

- 以下のオブジェクト権限付与 (**GRTOBJAUT**) コマンドを使用する。

```
GRTOBJAUT OBJ(ASN/package_name) OBJTYPE(*SQLPKG)  
USER(subscriber_name) AUT(*OBJOPR *EXECUTE)
```

- SQL を使用してソース・データベースに接続し、以下の GRANT SQL ステートメントを実行する。

```
CONNECT TO data_server_RDB_name  
GRANT EXECUTE ON PACKAGE ASN/package_name TO subscriber_name
```

- **GRTDPRAUT** コマンドを使用する (このコマンドがローカル・システムにインストールされている場合)。コマンド構文およびパラメーターの詳細については、415 ページの『GRTDPRAUT: ユーザーの許可 (OS/400)』を参照してください。

レプリケーション・プログラムのセットアップ (z/OS)

IBM DB2 DataPropagator for z/OS のインストール時にレプリケーション・プログラムをセットアップしてカスタマイズする必要があります。「*Program Directory for IBM DB2 DataPropagator for z/OS*」に記載されている説明を参照してください。

ジャーナルのセットアップ (OS/400)

DB2 DataPropagator for iSeries は、データに対する変更についてジャーナルから受け取る情報を使用することによって、レプリケーションのための CD および UOW 表にデータを入れます。

DB2 DataPropagator for iSeries は、大部分の操作のコミットメント・コントロール下で実行するため、コントロール表のジャーナリングを必要とします。(QSQRN ジャーナルは、**CRTDPRIBL** コマンドが集合を作成するとき作成されます。)

管理者は、ソース表、CD 表、およびターゲット表が含まれているライブラリーにジャーナルが含まれていることを確認する必要があります。すべてのソース表のジャーナル処理が正しく行われるのを見届ける責任もあります。

OS/400 でレプリケーション用の表を登録する前に、その表が変更前イメージと変更後イメージの両方についてジャーナリングされている必要があります。

以下のセクションでは、レプリケーションに必要なジャーナル・セットアップについて説明しています。

- 『ソース表用ジャーナルの作成 (OS/400)』
- 39 ページの『ジャーナルおよびジャーナル・レシーバーの管理 (OS/400)』

ソース表用ジャーナルの作成 (OS/400)

ソース表のジャーナルをセットアップするには、ソース表のジャーナルおよびジャーナル・レシーバーを定義するために作成する権限を所有している必要があります。(使用するソース表がすでにジャーナリングされている場合は、このセクションをスキップしてください。)

重要: ソース表のために使用するジャーナルは、DB2 DataPropagator for iSeries によって ASN (またはこれ以外のキャプチャー・スキーマ) ライブラリーに作成されたジャーナル以外のものにしてください。

プロシージャ:

ソース表のジャーナルを作成するには:

1. ジャーナル・レシーバーの作成 (**CRTJRNRCV**) コマンドを使用して、選択したライブラリーにジャーナル・レシーバーを作成します。定期的に保管されるライブラリーにジャーナル・レシーバーを置く。将来ジャーナル・レシーバーの命名規則を作成するのに使用できるようなジャーナル・レシーバー名 (RCV0001 など) を選択する。*GEN オプションを使用して、ジャーナル・レシーバーを変更するときに命名規則を継続することができます。このタイプの命名規則は、ジャーナル・レシーバーの変更をシステム管理に許可する場合にも役立ちます。以下の例では、ジャーナル・レシーバー用に JRNLIB という名前のライブラリーを使用します。

```
CRTJRNRCV  JRNRCV(JRNLIB/RCV0001)
            THRESHOLD(100000)
            TEXT('DataPropagator Journal Receiver')
```

2. ジャーナルの作成 (CRTJRN) コマンドを使用して、ジャーナルを作成します。

```
CRTJRN  JRN(JRNLIB/DJRN1)
        JRNRCV(JRNLIB/RCV0001)
        MNGRCV(*SYSTEM) DLTRCV(*YES)
        TEXT('DataPropagator Journal')
```

- ステップ 1 (37 ページ) で作成したジャーナル・レシーバーの名前を指定する。
- アタッチされたレシーバーが大きくなりすぎる場合、システムがジャーナル・レシーバーを変更して、新しいレシーバーをアタッチするために、レシーバーの管理 (MNGRCV) パラメーターを使用します。このオプションを選択すると、**CRTJRN** コマンドを使って、手でレシーバーを切り離したり、新しいレシーバーを作成およびアタッチしたりする必要がなくなります。
- デフォルトの属性 MINENTDTA(*NONE) を使用する。このキーワードの場合は、他の値は無効です。
- オーバーライドする理由がある場合 (たとえば、リカバーさせる目的でこれらのジャーナル・レシーバーを保管する必要がある場合) にのみ、DLTRCV(*NO) を指定します。DLTRCV(*YES) を指定すると、これらのレシーバーは、保管する機会もなく削除されてしまう可能性があります。

CRTJRN コマンドの RCVSIZOPT パラメーターに 2 つの値 (*RMVINTENT および *MINFIXLEN) を使用して、ストレージの可用性やシステム・パフォーマンスを最適化することができます。詳細については、「*OS/400 Programming: Performance Tools Guide*」を参照してください。

3. 以下の例で示す方法で、物理ファイルのジャーナル開始 (**STRJRNPF**) コマンドを使って、ソース表のジャーナリングを開始します。

```
STRJRNPF FILE(library/file)
        JRN(JRNLIB/DJRN1)
        OMTJRNE(*OPNCLO)
        IMAGES(*BOTH)
```

ステップ 2 で作成したジャーナルの名前を指定します。キャプチャー・プログラムには、IMAGES パラメーターの *BOTH という値が必要です。

4. ソース表のジャーナリングのセットアップを変更する。
 - a. IMAGES(*BOTH) を使用して、ソース表に対するジャーナリングが確実に変更前イメージと変更後イメージの両方について行われるようにする。
 - b. そのジャーナルに属性 MNGRCV(*YES) および DLTRCV(*YES) が指定されていることを確認する。
 - c. そのジャーナルに MINENTDTA(*NONE) が指定されていることを確認する。

ジャーナルおよびジャーナル・レシーバーの管理 (OS/400)

キャプチャー・プログラムは、ジャーナル・エントリーの受信 (RCVJRNE) コマンドを使って、ジャーナルを受信します。以下のセクションでは、レプリケーション環境でジャーナルおよびジャーナル・レシーバーを管理する方法が説明されています。

- ・ 『ジャーナル・レシーバーのシステム管理の指定 (OS/400)』
- ・ 『実行管理機能オブジェクトの定義の変更 (OS/400)』
- ・ 40 ページの『ジャーナル・レシーバーのユーザー管理の指定』
- ・ 40 ページの『ジャーナル・レシーバー削除出口ルーチンの使用』

ジャーナル・レシーバーのシステム管理の指定 (OS/400)

推奨: OS/400 システムにジャーナル・レシーバーの変更を管理させてください。これは、システム変更ジャーナル管理と呼ばれています。ジャーナルを作成するとき、またはジャーナルをその値に変更するとき、MNGRCV(*SYSTEM) を指定してください。システム変更ジャーナル管理を使用する場合、システムに変更させるジャーナル・レシーバーの限界値を指定するジャーナル・レシーバーを作成する必要があります。このしきい値は、最低でも 5,000 KB にし、システムでのトランザクションの数に基づいて決定する必要があります。システムは、レシーバーが限界値のサイズに達して、新しいジャーナル・レシーバーが作成およびアタッチされると (可能な場合)、そのレシーバーを自動的に切り離します。

実行管理機能オブジェクトの定義の変更 (OS/400)

DB2 DataPropagator for iSeries のインストール時に、このライブラリー用の SQL ジャーナルと SQL ジャーナル・レシーバー、および実行管理機能オブジェクトがインストール・プログラムによって作成されます。表 1 で、作成される実行管理機能オブジェクトをリストします。

表 1. 実行管理機能オブジェクト

| 記述 | オブジェクト・タイプ | 名前 |
|----------|------------|--------------|
| サブシステム記述 | *SBSD | QDP4/QZSNDPR |
| ジョブ・キュー | *JOBQ | QDP4/QZSNDPR |
| ジョブ記述 | *JOBDD | QDP4/QZSNDPR |

上記の 3 つのタイプの実行管理機能オブジェクトについて、そのデフォルトの定義を変更しても構いませんが、独自の定義も提供できます。独自のサブシステム記述を作成する場合は、サブシステムの名前として QZSNDPR を指定して、そのサブシステムを QDP4 以外のライブラリーに作成する必要があります。これらの定義の変更についての詳細は、「iSeries Work Management」(SC41-5306) を参照してください。

ジャーナル・レシーバーのユーザー管理の指定

ジャーナルの作成時に MNGRCV(*USER) を指定すると (独自にジャーナル・レシーバーの変更管理を行いたい場合)、ジャーナル・レシーバーがストレージの限界値 (レシーバーに指定した場合) に達すると、ジャーナルのメッセージ・キューにメッセージが送信されます。

CHGJRN コマンドを使用して、古いジャーナル・レシーバーを切り離し、新しいレシーバーをアタッチします。このコマンドを使用すると、ジャーナルされないエントリエラー状態を避け、ジャーナルが使用するストレージの量を制限することができます。パフォーマンスに影響がでないようにするため、システムが最大使用回数に達していないときにこのことを行ってください。

CHGJRN MNGRCV(*SYSTEM) を指定すると、ジャーナル・レシーバーの管理をシステムが行うよう切り替えることができます。

次の 2 つの理由で、現行のジャーナル・レシーバーを定期的に切り離し、新しいレシーバーをアタッチする必要があります。

- 各ジャーナル・レシーバーに特定の管理可能な期間が含まれていると、ジャーナル・エントリのアナライズがより簡単になるから。
- 大きなジャーナル・レシーバーは、システム・パフォーマンスに影響を及ぼし、補助記憶装置の貴重なスペースをふさいでしまう場合があるから。

ジャーナルのデフォルトのメッセージ・キューは QSYSOPR です。QSYSOPR メッセージ・キューに大きなボリュームのメッセージがあると、別のメッセージ・キュー (DPRUSRMSG など) とジャーナルを関連付けることができます。メッセージ処理プログラムを使って、DPRUSRMSG メッセージ・キューをモニターすることができます。ジャーナル・メッセージ・キューに送信できるメッセージに関する説明は、OS/400 *Backup and Recovery* を参照してください。

ジャーナル・レシーバー削除出口ルーチンの使用

DB2 DataPropagator for iSeries をインストールすると、ジャーナル・レシーバー削除出口ルーチン (DLTJRNRCV) が自動的に登録されます。この出口ルーチンは、ジャーナル・レシーバーが削除されると、そのレシーバーがソース表のジャーナリングに使用されているかどうかにかかわらず、常に呼び出されます。この出口ルーチンは、ジャーナル・レシーバーを削除できるかどうかを決定します。

ジャーナル・レシーバー削除出口ルーチンを使用して、ジャーナル管理をシステムに任せるには、**CHGJRN** または **CRTJRN** コマンドに対して DLTRCV(*YES) および MNGRCV(*SYSTEM) を指定します。

重要: ジャーナル・レシーバー削除出口ルーチン用の登録を除去する場合は、ソース表用に使用されるすべてのジャーナルを変更して、それらが DLTRCV(*NO) 属性を持つようにする必要があります。

レシーバーが関連付けられているジャーナルがどのソース表にも関連付けられていない場合、この出口ルーチンはそのレシーバーの削除を承認します。

ジャーナル・レシーバーが1つかそれ以上のソース表で使われている場合、この出口ルーチンは、キャプチャー・プログラムによって処理されなかったエントリーが、削除されているレシーバーに含まれていないことを確認します。キャプチャー・プログラムがそのレシーバーのエントリーをさらに処理する必要がある場合には、出口ルーチンはレシーバーの削除を承認しません。

ジャーナル・レシーバーを削除する必要があるのに、ジャーナル・レシーバー出口ルーチンが削除を承認しない場合には、DLTJRNCV DLT OPT(*IGNEXITPGM) を指定して、その出口ルーチンをオーバーライドします。

ジャーナル・レシーバー削除出口ルーチンの削除: ジャーナル・レシーバーの削除を手動で処理したい場合、以下のコマンドを使用して、ジャーナル・レシーバー削除出口ルーチンを削除することができます。

```
RMVEXITPGM EXITPNT (QIBM_QJO_DLT_JRNCV)
                FORMAT(DRCV0100)
                PGMNBR(value)
```

プロシージャ:

RMVEXITPGM コマンドの PGMNBR 値を決定するには:

1. **WRKREGINF** コマンドを実行します。
2. 「登録情報の処理」ウィンドウで、出口点のエントリー QIBM_QJO_DLT_JRNCV を検索します。「Opt」フィールドに 8 と入力します。
3. 「出口プログラムの処理」ウィンドウで、ライブラリー QDP4 にある出口プログラムのエントリー QZSNDREP を検索します。必要な数値は、出口プログラム番号の見出しの下にあります。

ジャーナル・レシーバー削除出口ルーチンの登録: 出口点を除去した後でそれを元に戻す必要が生じた場合は **ADDEXITPGM** コマンドを使用してください。その出口ルーチンは以下のコマンドを使用して登録する必要があります。

```
ADDEXITPGM EXITPNT(QIBM_QJO_DLT_JRNCV)
                FORMAT(DRCV0100)
                PGM(QDP4/QZSNDREP)
                PGMNBR(*LOW)
                CRTEXTIPNT(*NO)
                PGMFTA(65535 10 QSYS)
```

第 3 章 表およびビューをレプリケーション・ソースとして登録する

DB2 レプリケーションの場合、レプリケーション・ソースとして使用する表とビューは、それらを登録するという方法で示します。レプリケーション用として特定の表またはビューを登録する際には、後でさまざまな目的のために異なるターゲットに対して使用できる使用可能なデータのソースを作成します。この章で説明されている管理タスクは、レプリケーション・ゴールに基づいてデータが各ソースからどのように取り込まれるのかを定義するコントロール情報をセットアップするのに役立ちます。

ソースの登録時には、レプリケーション・ソースとして使用する表またはビュー、表内でレプリケーションのために使用可能にする列、および DB2 レプリケーションがソースからデータと変更をキャプチャーする方法に関するプロパティを示してください。

DB2 レプリケーションでは、以下のオブジェクトをソースとして登録できます。

- DB2 表
- 非 DB2 リレーショナル表 (ニックネームを使用)
- 表内のデータのサブセット (DB2 または非 DB2 リレーショナル)
- 単一表のビュー (DB2)
- 複数の表の内部結合を表すビュー (DB2)

この章は、以下のセクションを含んでいます。

- 『DB2 表をソースとして登録する』
- 46 ページの『非 DB2 リレーショナル表をソースとして登録する』
- 47 ページの『ソース表用の登録オプション』
- 64 ページの『レプリケーション・ソースとしてのビューの動作』
- 66 ページの『表のビューをソースとして登録する』
- 67 ページの『CCD 表をソースとして保守する (IMS)』

DB2 表をソースとして登録する

このセクションでは、DB2 表をレプリケーション・ソースとして登録する方法を説明します。DB2 レプリケーションでは、次のタイプの DB2 表がソースとしてサポートされています。

UNIX および Windows の場合

- ユーザー・アプリケーションが保守する DB2 表

- カタログ表 (フル・リフレッシュのみのレプリケーション)
- 自動サマリー表
- 外部 CCD 表

z/OS の場合

- ユーザー・アプリケーションが保守する DB2 表
- カタログ表
- 外部 CCD 表

OS/400 の場合

- ユーザー・アプリケーションが保守する DB2 表 (ローカルまたはリモート側でジャーナリングされる)
- 外部 CCD 表

OS/400 の場合を除くすべての DB2 ソースについては、ソース表 DDL は DATA CAPTURE CHANGES オプションを必要とします。このオプションをソースから除去しないでください。

表をソースとして登録すると、CD (変更データ) 表が作成されます。登録された表に関連付けられているキャプチャー・プログラムは、そのソース用のログを読み取って、登録された列に対して発生した未完了の変更を、トランザクションがコミットまたはロールバックされるまでメモリーに保管します。ロールバックの場合は、それらの変更はメモリーから削除されます。コミットの場合は、それらの変更はキャプチャー・プログラムがコミット・ログ・レコードを読み取ると同時に CD 表に挿入されます。これらの変更は、キャプチャー・プログラムが各キャプチャー・サイクルの後でコミットするまでメモリー内に残されます。キャプチャー・プログラムは、CAPSTART シグナルがユーザーまたはアプライ・プログラムによって発行されるまで、DB2 ソース表のデータのキャプチャーを開始しません。

非リレーショナル・ソース表の場合の注: IMS などの非リレーショナル・データベース管理システムからのデータを含む DB2 表を登録できます。これを行うには、IMS DataPropagator や Data Refresher などといった、非リレーショナル・データベースからのデータを含む CCD 表を移植するためのアプリケーションが必要です。このアプリケーションは、IMS データベース内の非リレーショナル・セグメントへの変更をキャプチャーし、CCD 表を移植します。この CCD 表はコンプリートでなければなりません。コンデンスされていてもコンデンスされていなくても構いません。他の CCD ソースと同様に、CCD ソース表には関連付けられているキャプチャー・プログラムがありません。なぜなら、この表には非リレーショナル・ソース表からの変更済みデータがすでに保管されているからです。IMS DataPropagator および Data Refresher 製品は、登録 (IBMSNAP_REGISTER) 表内の値を保守して、アプライ・プログラムがこのソース表からの読み取りを正しく行えるようにします。上記の製品のいずれかを使用してこれらのタイプの CCD 表を保守するのではなく、自分自身でそれらの表を保守する場合は、67 ページの『CCD 表をソースとして保守する (IMS)』を参照してください。

前提条件:

ソースとして登録する表を処理させるキャプチャー・コントロール・サーバー上に、キャプチャー・コントロール表がすでに存在していなければなりません。キャプチャー・コントロール表を作成する必要がある場合は、27 ページの『レプリケーション・コントロール表のセットアップ』を参照してください。

制約事項 (OS/400):

- SQL ステートメントは長さが 32,000 文字までに制限されているため、1 つの表当たりおよそ 2000 列までしか登録できない。正確な列数は、列名の長さによって決まります。
- 単一キャプチャー・スキーマに対し、同じジャーナルを使用するソース表を 300 個より多く登録してはならない。

プロシージャ:

以下の方法のいずれかを使用して、DB2 表を登録してください。

レプリケーション・センター

「表の登録」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

ヒント: 登録の際の時間を節約するために、キャプチャー・コントロール・サーバーに対してあらかじめソース・オブジェクト・プロファイルをセットアップしておけます。そのようにした場合は、表の登録時に、レプリケーション・センターはレプリケーション・センターのデフォルトではなく、そのソース・オブジェクト・プロファイルに定義されたデフォルトを使用します。これによって登録の際の時間を節約できます。なぜなら、それぞれの表を一度に 1 つずつ選択してデフォルト設定を手動で変更する代わりに、デフォルトを一度に上書きできるからです。

レプリケーション用のシステム・コマンド (OS/400)

ADDDPRREG システム・コマンドを使用します。このコマンドの構文およびパラメーターについての記述は、366 ページの『ADDDPRREG: DPR 登録の追加 (OS/400)』を参照してください。

DB2 表の登録時には、ソース・サーバー、ソース表の名前、およびキャプチャー・スキーマを指定して、どの表を登録するのかを示してください。別々のキャプチャー・スキーマを使用して、同じ表を複数回登録できます。登録用のデフォルト設定を使用するか、あるいはレプリケーションに関するニーズに合わせて登録オプションを変更するかは任意です。登録オプションとそれらのデフォルトの完全なリスト、およびこれらのデフォルトを使用した方がよい場合と変更した方がよい場合についての説明は、47 ページの『ソース表用の登録オプション』を参照してください。

非 DB2 リレーショナル表をソースとして登録する

このセクションでは、非 DB2 リレーショナル表をレプリケーション・ソースとして登録する方法を説明します。DB2 レプリケーションは、ニックネームを使用して非 DB2 リレーショナル表にアクセスします。

非 DB2 リレーショナル表をソースとして登録すると、CCD (整合変更データ) 表が作成されます。登録済みの非 DB2 リレーショナル表に対する変更が発生すると、キャプチャー・トリガーがキャプチャー・プログラムをシミュレートして、その変更を CCD 表に挿入します。キャプチャー・トリガーは、非 DB2 リレーショナル・ソース表への変更のキャプチャーを、そのソースが登録された時点から開始します。

デフォルトでは、CCD 所有者はソース表のスキーマ名から導き出されます。CCD 所有者をスキーマ名と一致しないように変更する場合は、ソース表の所有者が CCD 表への書き込みを許可されていることを確認してください。ソース表の所有者が CCD 表を更新できない場合は、ソース表のトリガーは変更を CCD 表に書き込めません。

前提条件:

このソースを処理させるキャプチャー・コントロール・サーバー上に、キャプチャー・コントロール表がすでに存在していなければなりません。キャプチャー・コントロール表を作成する必要がある場合は、28 ページの『非 DB2 リレーショナル・ソース用のコントロール表の作成』を参照してください。

制約事項:

- 複数の非 DB2 リレーショナル・ソース・サーバーにアクセスするのに単一の連合 DB2 データベースを使用している場合は、その単一連合データベースではそれぞれの非 DB2 リレーショナル・ソース・サーバーごとに異なるキャプチャー・スキーマを使用する必要があります。どれも同じであってはなりません。各非 DB2 リレーショナル表は、それぞれ 1 つのキャプチャー・スキーマの下にしか登録できません。
- データ・タイプ LOB または DATALINK を持つ非 DB2 リレーショナル表には列を登録できない。これらのデータ・タイプを含む表を登録する場合は、列のサブセットを登録する必要があります。列のサブセットだけを登録する方法についての詳細は、48 ページの『列のサブセットの登録 (垂直方向のサブセット化)』を参照してください。

プロシージャ:

以下に、非 DB2 リレーショナル表を登録する方法を示します。

レプリケーション・センター

「ニックネームの登録」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

ヒント: 登録の際の時間を節約するために、キャプチャー・コントロール・サーバーに対してあらかじめソース・オブジェクト・プロファイルをセットアッ

プしておけます。そのようにした場合は、表の登録時に、レプリケーション・センターはレプリケーション・センターのデフォルトではなく、CCD 表用のソース・オブジェクト・プロファイルに定義されたデフォルトと CCD 表のニックネームを使用します。これによって登録の際の時間を節約できます。なぜなら、それぞれの表を一度に 1 つずつ選択してデフォルト設定を手動で変更する代わりに、デフォルトを一度に上書きできるからです。

非 DB2 リレーショナル表の登録時には、ソース表のニックネームを指定して、どの表を登録するのかを示してください。登録用のデフォルト設定を使用するか、あるいはレプリケーションに関するニーズに合わせて登録オプションを変更するかは任意です。登録オプションとそれらのデフォルトの完全なリスト、およびこれらのデフォルトを使用した方がよい場合と変更した方がよい場合についての説明は、『ソース表用の登録オプション』を参照してください。

ソース表用の登録オプション

このセクションでは、表をレプリケーション・ソースとして登録する際に使用できるさまざまなオプションについて説明します。これらのオプションは、表の登録に関するタスクのうち、より大きな方のタスクの一部です。各表の登録方法の参照先を以下に示します。

- DB2 表、43 ページの『DB2 表をソースとして登録する』を参照してください。
- 非 DB2 リレーショナル表。46 ページの『非 DB2 リレーショナル表をソースとして登録する』を参照してください。

表のビューを作成してそれらのビューをソースとして登録するときには、ビューの登録オプションは基礎表の登録定義によって決定されます。ビューが基礎表からどの特性を継承するのか、およびビューが基礎登録定義に従ってどのように動作するのかについての詳細は、64 ページの『レプリケーション・ソースとしてのビューの動作』を参照してください。

登録する表の選択を済ませたら、レプリケーションのためにどの列を使用可能にするのかを示して、このソースから登録されたデータの処理方法と保管方法を決定するプロパティを定義できます。また、キャプチャー・プログラムによる CD 表へのソース・データの保管方法 (またはキャプチャー・トリガーによる CCD 表へのデータの保管方法) などといった、他の登録オプションも指定できます。このセクションでは、表をソースとして登録する際に指定可能な以下のオプションについて説明します。

- 48 ページの『列のサブセットの登録 (垂直方向のサブセット化)』
- 48 ページの『フル・リフレッシュ・コピーと変更キャプチャー・レプリケーション』
- 50 ページの『変更後イメージ列と変更前イメージ列』
- 53 ページの『変更前イメージ接頭部』
- 54 ページの『エラー発生時におけるキャプチャー・プログラムの停止』

- 54 ページの『キャプチャー・プログラムが更新を保存する方法』
- 55 ページの『変更のリキャプチャーの防止 (update-anywhere レプリケーション)』
- 60 ページの『競合検出の設定 (update-anywhere レプリケーション)』
- 63 ページの『主キーの代わりに相対レコード番号 (RRN) を使用する (OS/400)』

列のサブセットの登録 (垂直方向のサブセット化)

デフォルト: すべての列がレプリケーションのために登録されます

レプリケーション用のソース表を定義するときには、その表内のすべての列をレプリケーションのために登録しなければならないわけではなく、ソース表の列のサブセットを登録できます。この垂直サブセットは、サブスクライブするターゲットからすべての列を使用可能にするのは望ましくない場合に有用です。また、このオプションは、このソースのターゲット表ではソース表に定義されているデータ・タイプのすべてはサポートされていない場合にも選択するとよいでしょう。

列のサブセットを登録するには、ターゲット表へのレプリケーションのために使用可能にしたい列だけを選択します。選択しない列は、どのターゲット表へのレプリケーションにも使用できません。CD (および CCD) 表には一部のタイプのターゲット表にとって十分なキー・データ (ポイント・イン・タイムなど) が含まれていないため、使用するサブセットにターゲットに対してキー列 (主キーまたは固有索引) として機能する列が含まれていることを確認してください。

ヒント: ソース表内の列のサブセットを登録するのは、未登録の列は決して複製しないことを確信している場合だけにしてください。ソースの列のサブセットを登録し、後になって登録しなかった列を複製する必要が生じた場合は、登録を変更して未登録の列を追加しなければなりません (非 DB2 リレーショナル・ソースの場合は、1 つの登録に新しい列を追加するのにすべての登録をまとめて再定義する必要があります)。このソースに関連した内部 CCD を持つ予定である場合は、後で列を追加するのはさらに困難になる可能性があります。なぜなら、新しい列を登録すると、それらは内部 CCD ではなく、CD 表に追加されるからです。これらの問題を回避するためには、ソースのすべての列を登録し、代わりにアプライ・プログラムを使用して、ターゲットに複製する列をサブセット化するという方法もあります。ソースでなくターゲットでサブセット化を行う方法についての詳細は、98 ページの『ターゲットに適用するソース列』を参照してください。

フル・リフレッシュ・コピーと変更キャプチャー・レプリケーション

デフォルト: 変更キャプチャー・レプリケーション

レプリケーション・サイクルで毎回ソース表内のすべてのデータをターゲットにレプリケーションする (フル・リフレッシュのみのレプリケーション) か、ターゲットが最後にリフレッシュされた時点以降に発生した変更のみをレプリケーションする (変更キャプチャー・レプリケーション) かを選択できます。

制約事項: 複数ノード DB2 Enterprise Server Edition (ESE) 構成で変更を取り込めるのは、ソース表がパーティションに分割されていない状態でカタログ・ノードに常駐している場合だけです。

フル・リフレッシュのみのレプリケーション

ターゲットがフル・リフレッシュのみのレプリケーション用に登録されているソースに対してサブスクライブしている場合、アプライ・プログラムは毎回のレプリケーション・サイクルで、ターゲット表からすべてのデータを削除し、ソースの登録済み列に含まれているデータをコピーして、ソース・データをターゲットに移植します。キャプチャー・プログラムは関与せず、CD 表もありません。アプライ・プログラムは直接ソース表からデータを読み取ります。

表が小さい場合のヒント: コピーするのに時間もリソースもあまり要しない非常に小さなソース表を使用している場合は、フル・リフレッシュのみのレプリケーションを選択するとよいでしょう。

表が大きい場合のヒント: 表が比較的大きく、かつフル・リフレッシュのみのレプリケーションを使用したい場合は、表をより迅速にロードするために ASNLOAD 出口ルーチンを使用するとよいでしょう。詳細については、169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』を参照してください。

制約事項: このソースにサブスクライブするのをコンデンスされたターゲット表にする予定であり、さらにそのターゲット表の固有索引は用意できない場合は、ソースをフル・リフレッシュのみのレプリケーション用として登録する必要があります。

変更キャプチャー・レプリケーション

デフォルト: すべての行に対する変更が取り込まれます

変更キャプチャー・レプリケーション時には、変更されたデータだけがターゲット表にレプリケーションされます。このソース用に選択したターゲット表のタイプによっては、この表の初期ロードを実行する必要があります。ほとんどの場合、アプライ・プログラムは初期フル・リフレッシュを実行してから、継続して変更キャプチャー・レプリケーションを実行します。

ターゲット表に対するフル・リフレッシュを許可しないことを選択すると、ソース表とターゲット表の再同期化が必要になった場合にターゲット表を手動で再ロードする必要があります。ターゲットが初期ソース・データを含んでいる状態でロードされた後、キャプチャー・プログラムはソースで発生した変更をキャプチャー CD 表に保管します。非 DB2 リレーショナル・ソースの場合の変更キャプチャー・レプリケーションでは、ソースで発生した変更をキャプチャー・トリガーがキャプチャー CCD 表に保管します。アプライ・プログラムは、CD 表または CCD 表から変更を読み取って、その登録されているソースにサブスクライブしているターゲットにそれらの変更を適用します。

DB2 ソース表を変更キャプチャー・レプリケーション用として定義する場合、ソースで発生した変更をすべて CD 表に保管するのは望ましくないことがあります。行 (水平方向) サブセットを登録して変更をフィルター操作し、実際にソースで発生している変更より少ない変更が CD 表に取り込まれるようにできます。以下の 2 つの行キャプチャー規則のいずれかを選択して、キャプチャー・プログラムがソース表からのどの変更済み行を CD 表に記録するのかを決定できます。

- すべての行に対する変更をキャプチャーする
- 変更が登録済みの列に発生した場合にのみ変更をキャプチャーする (DB2 のみ)

デフォルトでは、ソース表で行のいずれかの列 (登録済みでも未登録でも) が更新されるたびに、必ず変更が取り込まれます。列のサブセットだけを登録した場合は、ソース表に対して変更が発生するたびに、キャプチャー・プログラムは登録済み列の値を行単位で CD 表に記録します。これは、変更された列が登録済みの列とは異なる列であった場合でも実行されます。このデフォルト・オプションは、ソース表へのすべての変更の履歴を保持したい場合に使用してください。非 DB2 リレーショナル・ソースの場合に使用可能なオプションはこれだけです。キャプチャー・トリガーは、変更が未登録の列に対して発生した場合であっても、ソースで変更のあったすべての行をキャプチャーします。

例: 表には 100 個の列があり、それらの列うち 50 個をレプリケーション用に登録しているとします。デフォルトでは、表内の 100 個の列のどれか 1 つにでも変更が加えられると、いつでもキャプチャー・プログラムによって行が CD 表に書き込まれます (あるいは、キャプチャー・トリガーによって行が CCD 表に書き込まれます)。

DB2 ソースがある場合は、登録済みの列への変更だけをキャプチャー・プログラムに取り込ませたいことがあります。このケースでは、キャプチャー・プログラムは登録済みの列に対する変更が発生したときだけ行を CD 表に書き込みます。

提案: 監査を目的とする情報が必要な場合、または表ではほとんどいつも登録済みの列に対してのみ変更が発生している場合は、すべての行への変更のキャプチャーを選択してください。未登録の列にのみ影響する変更が頻繁に発生している場合は、登録済みの列だけに対する変更のキャプチャーを選択してください。このオプションは、ソース表へのすべての変更の履歴を保持するのは望ましくない場合に使用してください。

変更後イメージ列と変更前イメージ列

デフォルト: 変更後イメージ列のみ

ソースを変更キャプチャー・レプリケーション用に登録する際には、キャプチャー・プログラムに変更後イメージ値 (変更後の列内の値) だけを取り込ませるのか、あるいは変更後イメージ値と変更前イメージ値 (変更前に列内に入っていた値) の両方を取り込ませるのかを選択できます。UNIX、Windows、および z/OS の場合は、変更前イメージ値を取り組むかどうかを表内の列ごとに選択できます。OS/400 の場合は、変更前イメージのキャプチャーを表内のすべての列について行うか、あるいはどの列に対しても

行わないかのどちらかを選択できます。このオプションは、個々の列ごとには選択できません。以下の各セクションでは、どのような場合に各オプションを選択する必要があるのかについて説明します。

以下に示すように、いくつかの非 DB2 リレーショナル・ソース表の場合、CCD 表には変更後イメージ値だけを含めるようにする必要があります。

- Sybase または Microsoft SQL Server の表には、タイプが **TIMESTAMP** の列は 1 つしか含められない。データ・ソースが Sybase または Microsoft SQL Server であり、かつソース表にタイプが **TIMESTAMP** の列があるときは、この列をレプリケーション・ソースの一部として定義する場合、この列については変更後イメージのみを選択してください。
- Oracle 表には、タイプが **LONG** である列を 1 つしか持つことができません。データ・ソースが Oracle であり、かつソース表にタイプが **LONG** の列があるときは、この列をレプリケーション・ソースの一部として定義する場合、この列については変更後イメージのみを選択してください。

以下の特定のデータ・タイプを指定された列の場合は、CD 表に変更前イメージ値を含められません。

- LOB データ・タイプの列
- DATALINK データ・タイプの列

変更後イメージ値のみのキャプチャー

変更キャプチャー・レプリケーション用に登録するそれぞれの列ごとに、変更が行われるたびにキャプチャー・プログラムまたはキャプチャー・トリガーに対して、変更後イメージ値だけを記録させることを選択することができます。変更後イメージ値だけを組み合わせることを選択した場合は、CD (または CCD) 表にはそれぞれの変更された値ごとに 1 つの列が含まれ、その列には変更発生後のソース列の値が保管されます。

このソースに対しては基礎集約ターゲット表タイプおよび変更集約ターゲット表タイプだけを使用する予定である場合は、変更前イメージは必要ありません。ターゲット表を計算済みの値のために使用する予定である場合は、変更前イメージ列は無意味です。なぜなら、計算された列には変更前イメージがないからです。他のすべてのターゲット表タイプでは、変更前イメージ列を利用することができます。集約ターゲット表についての詳細は、88 ページの『ソースにおけるデータまたは変更の算出サマリー』を参照してください。

変更前イメージ値と変更後イメージ値のキャプチャー

変更キャプチャー・レプリケーション用に登録するそれぞれの列ごとに、変更が行われるたびにキャプチャー・プログラムまたはキャプチャー・トリガーに対して、変更前イメージ値と変更後イメージ値の両方を記録させることを選択することができます。変更前イメージ値と変更後イメージ値をキャプチャーすることを選択した場合は、CD (また

は CCD) 表にはそれぞれの変更された値ごとに 2 つの列が含まれます。一方の列は変更が発生する前にソース列に入っていた値用で、もう一方の列は変更発生後の値用です。

変更前イメージと変更後イメージの両方を CD (または CCD) 表に保管することを選択した場合は、以下に示すように、変更前イメージ列と変更後イメージ列には、ソース表に対して実行されたそれぞれ異なるアクションの種類に応じて異なる値が含まれません。

| アクション | 列値 |
|-------|--|
| 挿入 | 変更前イメージ列には NULL 値が含まれます。変更後イメージ列には挿入された値が含まれます。 |
| 更新 | 変更前イメージ列には、変更が発生する前の列値が含まれます。変更後イメージ列には、変更が発生した後の列値が含まれます。 更新を削除と挿入の対としてキャプチャーすることを選択した場合は、削除行ではその行の変更前イメージ列と変更後イメージ列の両方に更新の変更前イメージが含まれ、挿入行では変更前イメージ列には NULL 値が、変更後イメージ列には変更後イメージが含まれます。このオプションの詳細については、54 ページの『キャプチャー・プログラムが更新を保存する方法』を参照してください。 |
| 削除 | 変更前イメージ列と変更後イメージ列に、変更が発生する前の列値が含まれます。 |

UNIX、Windows、および OS/400 の場合の重要事項: 定義によって変更前イメージが含まれている列の場合、DB2 レプリケーションでは列名が 29 文字までに制限されます。なぜなら、列名全体で許容される文字数が 30 文字だけだからです。列名がそれより長いと、左方から切り捨てるようにプロファイルを設定していないかぎり、デフォルトでは余分な文字が DB2 レプリケーションによって右方から切り捨てられます。DB2 レプリケーションによってターゲット列に変更前イメージ列 ID (通常は X) が追加される上に、各列名は必ず固有でなければならないため、29 文字より長い列名は使用できません。複製するつもりがない表にはさらに長い列名を使用できますが、その列を将来複製する可能性がある場合は 29 文字の名前を使用することを検討してください。

z/OS の場合の重要事項: DB2 for z/OS の表であれば 18 文字の列名を使用できますが、DB2 DataPropagator は 18 番目の文字をターゲット表では変更前イメージ列 ID に置き換えるため、列名の最初の 17 文字が必ず固有になるようにする必要があります。

以下のセクションでは、変更前イメージ値のキャプチャーが必要となる可能性のあるケースについて説明しています。

- 53 ページの『ソース・データのヒストリーを保持する場合』
- 53 ページの『競合検出を使用する update-anywhere 構成の場合』
- 53 ページの『ターゲットでキー列が更新の対象となっている場合』

ソース・データのヒストリーを保持する場合: 監査目的でデータを保持する必要がある場合は、ある期間内にデータがどのように変更されたかについてのレコードを持つように、変更前イメージと変更後イメージの両方を選択します。監査やアプリケーション・ロールバック機能を必要とする業界では、変更前イメージと変更後イメージのコピーのセットが役立ちます。

競合検出を使用する update-anywhere 構成の場合: レプリカ表 (競合検出が None 以外に設定されているもの) の間で競合が起こり得る update-anywhere 構成では、レプリカの CD 表に変更後イメージ列と変更前イメージ列の両方を登録して、競合が発生した場合には変更をロールバックできるようにする必要があります。

ターゲットでキー列が更新の対象となっている場合: ソースを登録する際には、この表をソースとして使用して定義する可能性のあるターゲット表があるかどうかを検討してください。通常ターゲット表はコンデンスされており、そのターゲット表内の各行を固有にする列または列のセットを必要とします。それらの固有列によっていわゆるターゲット・キーが構成されます。これらのターゲット・キー列のいずれかがソースで更新される可能性がある場合は、DB2 レプリケーションではターゲット表で必ず正しい行が更新されるようにするための特殊な処理が必要となります。DB2 レプリケーションがターゲット表で必ず正しい行を新しいキー値で更新するようにするために、ターゲット・キーを構成する列については変更後イメージと変更前イメージの両方をキャプチャーすることを選択できます。これらの登録済み列の変更前イメージ値は、アプライ・プログラムが非キー・ソース列の変更をターゲット表のターゲット・キー列に適用するときに必要となります。変更を適用するときには、アプライ・プログラムはこの行をターゲット表でソースの CD (または CCD) 表内の変更前イメージ値と一致するターゲット・キー値を探すという方法で検索し、次にそのターゲット行をソースの CD (または CCD) 表内の変更後イメージ値で更新します。

これらの変更前イメージ値をソース表またはビューの登録時に登録しても、DB2 レプリケーションはユーザー・アプリケーションがターゲット・キーに対して更新を行うことを知りません。後で (サブスクリプション・セットを作成して) どのターゲットをこのソースにサブスクライブするのかを定義するときに、アプライ・プログラムに対し、変更をソースの非キー列からターゲットのキー列に適用する際に特殊な更新を実行するよう指定できます。詳細については、102 ページの『アプライ・プログラムがターゲット・キー変更オプションを使用してターゲット・キー列を更新する方法』を参照してください。

変更前イメージ接頭部

デフォルト (レプリケーション・センター): X

デフォルト (OS/400 システム・コマンド): @

CD (または CCD) 表の変更後イメージ列と変更前イメージ列の両方をキャプチャーすることを選択した場合は、変更後イメージ列名はソース表での列名となり、変更前イメ

ージ列名は先頭に 1 文字の接頭部を追加されたソース表での列名となります。変更前イメージ列名用のデフォルトの 1 文字接頭部は変更可能です。変更前イメージ接頭部と CD (または CCD) 列名の組み合わせは、あいまいさのないものでなければなりません。これは、接頭部列名は CD (または CCD) 表内の現行列名または潜在的な列名と同じであってはならないという意味です。

例: 変更前イメージ接頭部として X を使用していて、さらに COL という名前のソース列を登録している場合は、XCOL という名前の列は登録できません。なぜなら、XCOL が別のソース列の実際の列名なのか、それとも COL という列名と変更前イメージ接頭部 X を持つ変更前イメージ列の名前なのかがはっきりしないからです。

表の変更前イメージ列を一切複製していない場合は、変更前イメージ接頭部を持たないことを選択して、このプロパティを NULL に設定できます。

エラー発生時におけるキャプチャー・プログラムの停止

デフォルト: キャプチャー・プログラムは特定のエラーを検出したときに停止します

登録の処理中に特定の問題を検出すると、キャプチャー・プログラムは停止するか、または実行を続けます。以下のオプションのいずれかを選択して、登録されたソースの処理中に特定のエラーを検出した場合のキャプチャー・プログラムの反応を決定できます。

エラー発生時にキャプチャーを停止

キャプチャー・プログラムは、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表にエラー・メッセージを書き込んで終了します。

エラー発生時にキャプチャーを停止しない

特定のエラーが発生した場合、キャプチャー・プログラムは実行を続けます。ソース処理の初回の試行中にエラーを検出した場合は、キャプチャー・プログラムは登録を活動化しません。登録されたソースがすでに活動化されていた場合は、登録の処理を停止します。

このオプションにより、以下の致命的でないエラーが発生した場合にキャプチャー・プログラムが稼働し続けるかどうか決定されます。

- 登録が正しく定義されていません。
- キャプチャー・プログラムが変更済みデータの行の挿入を試行したときに、CD 表が検出されなかった。
- キャプチャー・プログラムが始動または再初期化されたときに、(非 OS/400) ソース表の DATA CAPTURE CHANGES オプションがオフにされた状態で検出された。

致命的エラーの場合は、キャプチャー・プログラムは実行を続けません。

キャプチャー・プログラムが更新を保存する方法

デフォルト: 更新は CD 表の単一行に保管されます

ソースの更新を CD (または CCD) 表に保管する方法を選択できます。ソース表に対する更新を取り込んだ場合、キャプチャー・トリガーまたはキャプチャー・プログラムは、更新された値を CD 表の単一行に保管するか、あるいは CD (または CCD) 表の 2 つの行を使用して、削除を一方の行に、挿入をもう一方の行に保管するかのいずれかを実行できますデフォルトでは、更新は単一行に保管されます。このデフォルトを使用すると、ストレージが削減され、パフォーマンスが向上します。なぜなら、CD (または CCD) 表には 1 行だけが保管され、さらにその行は変更が行われるたびにアプライ・プログラムによって読み取られるからです。ただし、キャプチャー・プログラムまたはキャプチャー・トリガーに対してソース表への更新を DELETE と INSERT の対としてキャプチャーするよう指示する必要があるシナリオがいくつかあります。

ソース・アプリケーションがサブスクリプション・セット・メンバーの述部で参照される列を 1 つ以上更新する場合は、更新を DELETE ステートメントと INSERT ステートメントとしてキャプチャーする必要があります。ソース・データだけにサブスクライブするターゲットを、特定の列値を基にした述部を使用して定義する (たとえば、WHERE DEPT = 'J35') 予定であるとしします。その列を変更すると (たとえば、DEPT='FFK' に変更)、取り込まれた変更は、ターゲットにレプリケーションする対象として選択されません。なぜなら、その列は述部の基準に合致していないからです。つまり、新しい FFK 部門は、サブスクリプション・セットが部門 J35 に基づいているため複製されません。更新を DELETE および INSERT の対に変換すると、そのターゲット表列は確実に削除されます。

取り込まれたそれぞれの更新は、CD (または CCD) 表の 2 つの行に変換されます。この変換は、すべての列について実行されます。この取り込んだデータの増加に合わせて CD (または CCD) 表のスペース割り振りの調節が必要になる場合もあります。

DATALINK 値の場合の重要事項: DATALINK 列が ON UNLINK DELETE として定義されている場合、DELETE と INSERT の対が同じトランザクション内で処理されるため、このリンク解除は無視されます。外部ファイルは削除されませんが、更新は行われます。

変更のリキャプチャーの防止 (update-anywhere レプリケーション)

新しいソース表の場合のデフォルト: 変更がリキャプチャーされます

新しいレプリカ表の場合のデフォルト: 変更はリキャプチャーされません

制約事項: 非 DB2 リレーショナル・データベースからの表は、update-anywhere にかかわれません。したがって、このオプションは DB2 ソース専用です。

update-anywhere レプリケーションの場合は、変更はマスター表または関連したレプリカ表で発生する可能性があります。update-anywhere レプリケーションで使用する予定の表を登録する場合、DB2 レプリケーションはその表がユーザーの構成の中でマスター表となることを前提とします。リキャプチャー・オプションを使用すると、あるサイトで発生して 2 番目のサイトに複製された変更を、その 2 番目のサイトでリキャプチャー

し、それによってその変更を追加の各サイトへのレプリケーションに使用できるようにするかどうかをコントロールできます。登録時に、このオプションをマスター表に対して設定してください。その後、そのマスター・ソース表をレプリカ・ターゲットにマップする際に、レプリカの変更をリキャプチャーして他の表に転送するかどうかを設定できます。(マスターをレプリカにマップする方法についての詳細は、94 ページの『読み取り / 書き込みターゲットの定義 (update-anywhere)』を参照してください。)

update-anywhere 構成でマスターとして機能するソース表を登録する場合は、以下の 2 つのオプションのうち、どちらかを選択できます。

マスターで変更をリキャプチャーする

あるレプリカで発生したマスターに対する更新は、マスターでリキャプチャーされて、他のレプリカに転送されます。

マスターで変更をリキャプチャーしない

あるレプリカで発生してマスターに加えられた更新は、マスターでリキャプチャーされず、他のレプリカに転送されません。

update-anywhere 構成のレプリカ表を登録する場合は、以下の 2 つのオプションのうち、どちらかを選択できます。

レプリカで変更をリキャプチャーする

マスターで発生してレプリカに加えられた更新は、そのレプリカでリキャプチャーされて、このレプリカにサブスクライブしている他のレプリカに転送されます。

レプリカで変更をリキャプチャーしない

マスターで発生してレプリカに加えられた更新は、そのレプリカでリキャプチャーされず、このレプリカにサブスクライブしている他のレプリカに転送されません。

変更のリキャプチャーを防止すると、パフォーマンスを向上させ、さらにストレージ・コストを削減できます。なぜなら、キャプチャー・プログラムが同じ変更をそれぞれのレプリカごとにリキャプチャーしないからです。

以下のセクションでは、**update-anywhere** 構成に基づいて変更をリキャプチャーするかどうかを決定する方法について説明しています。

- 57 ページの『レプリカを 1 つだけ持つマスターの場合』
- 57 ページの『マスターの相互に排他的なパーティションである複数のレプリカの場合』
- 58 ページの『変更を複数のレプリカに複製するマスターの場合』
- 58 ページの『他のレプリカに変更を複製するレプリカの場合 (multi-tier)』

レプリカを 1 つだけ持つマスターの場合

マスター: 変更をマスターでリキャプチャーしません

単一レプリカ: 変更をレプリカでリキャプチャーしません

update-anywhere 構成内にレプリカを 1 つしか持たない予定である場合は、マスター表でもレプリカ表でも変更がリキャプチャーされないようにしたいことがあります。マスター表が他のレプリカ表にとってのソースではなく、レプリカが他のレプリカにとってのソースではない場合 (multi-tier 構成において) は、これが最良の設定です。関与するのがこの 2 つの表だけである場合は、レプリカで発生した変更をマスターでリキャプチャーする必要は無く、またマスターで発生したいかなる変更も単一レプリカでリキャプチャーする必要がありません。

マスターの相互に排他的なパーティションである複数のレプリカの場合

マスター: 変更をマスターでリキャプチャーしません

レプリカ: 変更をレプリカでリキャプチャーしません

マスター表のパーティションである複数のレプリカを持つ予定である場合は、マスター表でも各レプリカ表でも変更がリキャプチャーされないようにしたいことがあります。どのレプリカも他のレプリカ表にとってのソースではない場合は、これが最良の設定です。レプリカがマスターのパーティションである場合は、複数のレプリカがマスターの同じデータにサブスクライブすることがあってはなりません。したがって、どのレプリカで発生したどの変更もマスターでリキャプチャーする必要は無く、他のレプリカに転送する必要もありません。なぜなら、そのソース・データにサブスクライブしているのはその変更が発生したレプリカだけだからです。

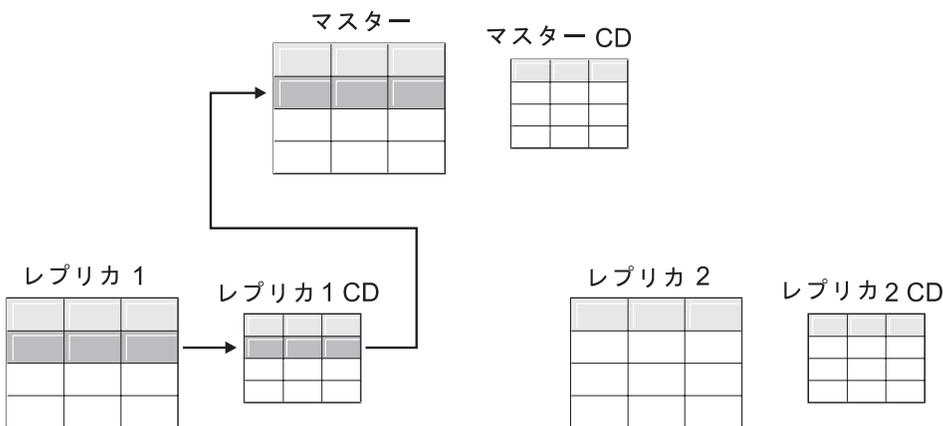


図 1. マスターの相互に排他的なパーティションであるレプリカ用のリキャプチャー・オプション: マスターの同じデータにサブスクライブしていない複数のレプリカを持っている場合は、どの表に対してもこのリキャプチャー・オプションを使用する必要はありません。

変更を複数のレプリカに複製するマスターの場合

マスター: 変更をマスターでリキャプチャーします

レプリカ: 変更をレプリカでリキャプチャーしません

マスター表の同じデータにサブスクライブする複数のレプリカを持つ予定である場合は、キャプチャー・プログラムに変更をマスターでリキャプチャーさせる必要がある場合があります。そのようにすると、レプリカで発生した変更はマスターでリキャプチャーされて、その更新されたマスター・データにサブスクライブしている他のレプリカへと複製されます。

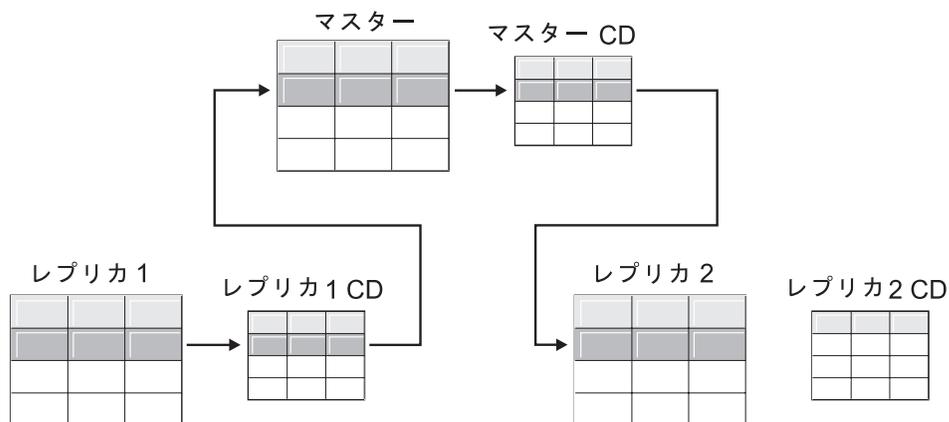


図2. 変更を複数のレプリカに複製するマスター用のリキャプチャー・オプション: マスターの同じデータにサブスクライブしている複数のレプリカを持っている場合は、マスターでリキャプチャー・オプションを使用して、あるレプリカで発生した変更がマスターでリキャプチャーされて他のレプリカ表に転送されるようにできます。

他のレプリカに変更を複製するレプリカの場合 (multi-tier)

マスター: 変更をマスターでリキャプチャーしません

レプリカ: 変更をレプリカでリキャプチャーします

マスター (層 1) があるレプリカ (層 2) に対してソースとして機能し、次にそのレプリカが同様に別のレプリカ (層 3) に対してソースとして機能する multi-tier 構成を持てます。このタイプの構成を持つ予定である場合は、キャプチャー・プログラムに中間のレプリカ (層 2) で変更をリキャプチャーさせて、マスターで発生した変更がその次のレプリカ (層 3) に転送されるようにしたい場合があります。

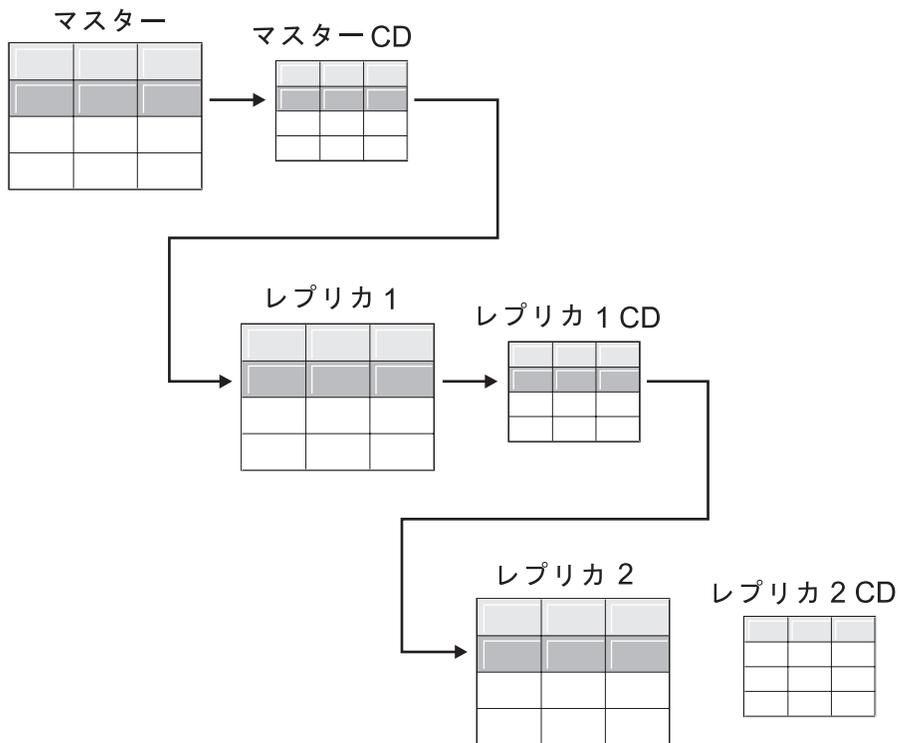


図3. 層 2 でリキャプチャー・オプションを使用すると、層 1 での変更が層 3 にまで複製されるようにできます： multi-tier 構成で中間層として機能するレプリカ表がある場合は、そのレプリカでリキャプチャー・オプションを使用して、マスターで発生した変更が中間層のレプリカでリキャプチャーされて下層のレプリカに転送されるようにできます。

また、中間のレプリカ (層 2) にリキャプチャーを設定してある場合は、最終のレプリカ (層 3) で発生した変更が中間のレプリカ (層 2) でリキャプチャーされてマスター (層 1) に転送されます。

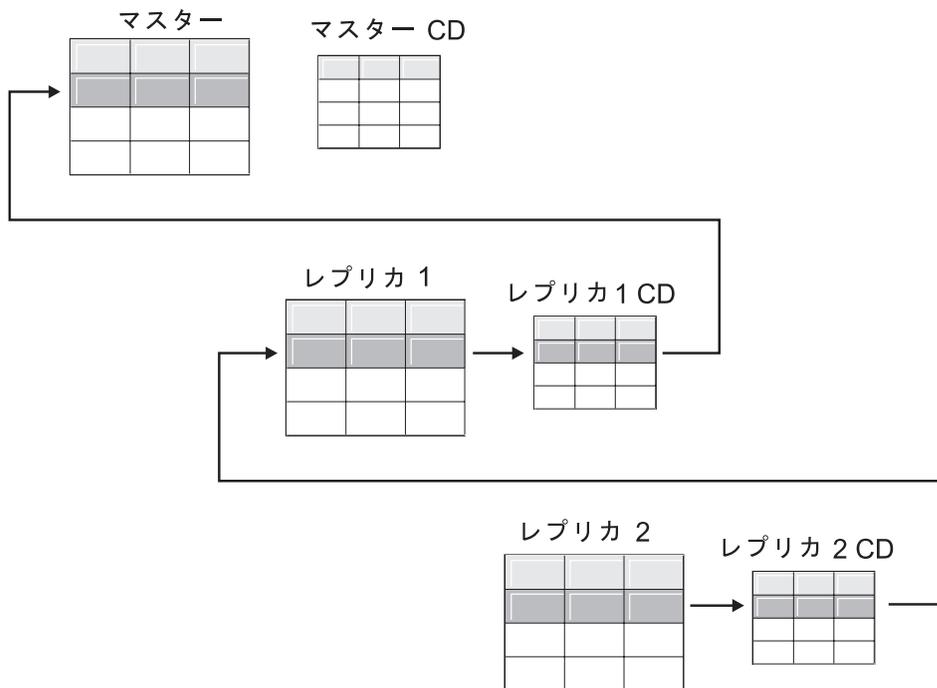


図4. 層 2 でリキャプチャー・オプションを使用すると、層 3 での変更が層 1 にまで複製されるようにできます： multi-tier 構成で中間層として機能するレプリカ表がある場合は、そのレプリカでリキャプチャー・オプションを使用して、下層のレプリカで発生した変更が中間層のレプリカでリキャプチャーされてマスターに転送されるようにできます。

競合検出の設定 (update-anywhere レプリケーション)

デフォルト: 競合検出なし

制約事項:

- 非 DB2 リレーショナル・データベースからの表は update-anywhere にかかわれない。したがって、非 DB2 リレーショナル・ソースでは競合検出は行われません。
- DATALINK 列を含む update-anywhere 構成がある場合は、競合検出レベルに None を指定する必要がある。DB2 は DATALINK 列が参照する外部ファイルの更新競合はチェックしません。
- LOB 列を含む update-anywhere 構成がある場合は、競合検出レベルに None を指定する必要がある。LOB データ・タイプの列は、update-anywhere レプリケーションの対象になりません。

update-anywhere 構成では、マスターとそのレプリカの間で時々競合が発生する可能性があります。以下に、競合が起こる可能性のある場合を示します。

- マスター表である行が更新されたが、1 つ以上のレプリカ表で同じ行にそれとは異なる更新が加えられ、さらにアプライ・プログラムがそれらの競合する変更を同一サイクルの間に処理した。
- 制約に違反した。

競合検出レベルを個々のレプリケーション・ソースに設定した場合でも、アプライ・プログラムはサブスクリプション・セットのすべてのメンバーのレベルとして、サブスクリプション・セット・メンバーの競合検出レベルのうち最も高いものを使用します。

DB2 レプリケーションでは、3 つのレベルの競合検出 (検出なし、標準検出、および拡張検出) が提供されます。逸失またはリジェクトされたトランザクションの許容度とパフォーマンス要件に基づいて、どのタイプの検出を使用するかを決定できます。

update-anywhere レプリケーションに使用する予定のソースの登録時に、以下の競合検出レベルの中からいずれかを選択できます。

なし 競合検出なし。マスター表とレプリカ表の間で競合している更新は検出されません。このオプションは、**update-anywhere** レプリケーションではお勧めしません。

標準 適度な競合検出。

各アプライ・サイクルの間に、アプライ・プログラムはマスターの CD 表のキー値とレプリカの CD 表のキー値を比較します。両方の CD 表に同じキー値が存在している場合、それは競合です。競合が発生すると、アプライ・プログラムは直前にレプリカでコミットされたトランザクションを取り消します。これは、レプリカの CD 表からの読み取りを実行し、マスターで発生した変更だけを保持するという方法で行われます。

拡張 マスターとそのレプリカとの間での最良のデータ保全性を実現する競合検出。

標準検出の場合と同様に、アプライ・プログラムは各アプライ・サイクルの間にマスターの CD 表のキー値とレプリカの CD 表のキー値を比較します。両方の CD 表に同じキー値が存在している場合、それは競合です。ただし拡張検出では、競合をチェックする前に、アプライ・プログラムはすべての未完了トランザクションがコミットされるまで待ちます。すべての未完了トランザクションを確実に捕らえられるように、アプライ・プログラムはこれから処理されるトランザクションに対してサブスクリプション・セット内のすべてのターゲット表をロックし、CD 表内のすべての変更が取り込まれた後で競合検出を開始します。競合が発生すると、アプライ・プログラムは直前にレプリカでコミットされたトランザクションを取り消します。これは、レプリカの CD からの読み取りを実行し、マスターで発生した変更だけを保持するという方法で行われます。

制約事項: 拡張競合検出を指定しても、アプライ・プログラムが不定期接続環境で実行されている (**COPYONCE** キーワードを使用して始動された) 場合は、アプライ・プログラムは標準競合検出を使用します。

アプライ・プログラムは、読み取り従属関係を検出することはできません。たとえば、後で (DELETE ステートメントによって、またはロールバック・トランザクションによって) 除去されることになる情報をアプリケーションが読み取る場合、アプライ・プログラムは従属関係を検出することはできません。

競合が発生し得るレプリケーション構成を (検出なしまたは標準検出のいずれかを選択して) セットアップする場合は、発生したあらゆる競合を識別して処理するための方法を組み込む必要があります。競合のあったトランザクション更新はレプリケーションのインフラストラクチャーによってすでに検出されてバックアウトされているとしても、アプリケーション・デザイナーは、いったんコミットされたが現在はバックアウト済みのトランザクションをどのように処理するのかを決定する必要があります。ASNDONE 出口ルーチンは各サブスクリプション・サイクルの最後で実行されるため、アプリケーション・デザイナーはこのルーチンを上記のようなアプリケーション固有のロジックのためのランチ点として使用できます。バックアウト済みの競合する更新に関する情報は、保持制限ブルーニングの対象として適格となるまで CD 表および UOW 表に残されます。このアプリケーション固有のロジックを追加する方法についての詳細は、使用しているプラットフォームの種類に応じて 166 ページの『ASNDONE 出口ルーチンの変更 (UNIX、Windows、z/OS)』または 167 ページの『ASNDONE 出口ルーチンの変更 (OS/400)』を参照してください。

リモート・ジャーナリングを使用する表の登録 (OS/400)

デフォルト: リモート・ジャーナルはソースとして使用されません

リモート・ジャーナリングを使用する OS/400 表を登録する場合、DB2 レプリケーションではレプリケーション・ソースとしてローカル・ジャーナルの代わりにリモート・ジャーナルを使用することを定義できます。レプリケーション用のリモート・ジャーナリング・オプションを選択して、CD 表、キャプチャー・プログラム、およびキャプチャー・コントロール表を、ソース表が置かれている OS/400 サーバーとは別個の OS/400 データベース・サーバーに移動してください。

OS/400 で表をソースとして登録する場合、デフォルトではリモート・ジャーナリングを使用しないことが前提とされます。

推奨: ある OS/400 表から別の OS/400 表にデータを複製する場合、セットアップ済みのリモート・ジャーナルがあるときは常に、登録時にリモート・ジャーナリング関数を使用することを強く推奨します。レプリケーションにリモート・ジャーナリングを使用すると、パフォーマンスが大幅に向上します。リモート・ジャーナル関数を使用すると、登録、キャプチャー・プログラム、およびキャプチャー・コントロール表をソース表が置かれているシステムから遠く離れたところに移動できるため、そのシステムにはより多くのリソースが使用可能な状態で残されます。これにより、プロセッサ使用量が削減され、ディスク・スペースが節約されます。また、ターゲット・サーバーにあるリモート・ジャーナルを使用する場合は CD 表がターゲット表と同じシステム上に存在するため、アプライ・プログラムは予備ファイルを使用せずに直接 CD 表からターゲット

ト表に変更を適用できます。予備ファイルを使用しないことで、アプライ・プログラムによって使用されるリソースの量が削減されます。

提案: リモート・ジャーナルを使用する表をソースとして登録するのは、登録がレプリケーション・ターゲットと同じ OS/400 システム上にある場合だけにしてください。DB2 レプリケーションを使用すると、登録がターゲットと同じ OS/400 システム上になくてもリモート・ジャーナルをソースとして登録できますが、その場合にはジャーナルをターゲット・システムに置くことで得られるパフォーマンス上の利点を得られません。

リモート・ジャーナリングを使用する OS/400 表を登録する前に、使用するリモート・ジャーナルがアクティブ状態になっていることを確認してください。

リモート・ジャーナル関数についての詳細は、「*Backup and Recovery*」(SC41-5304) および「*OS/400 Remote Journal Function for High Availability and Data Replication*」(SG24-5189) を参照してください。

主キーの代わりに相対レコード番号 (RRN) を使用する (OS/400)

通常、あるソース用のターゲット表はそのソース内の主キー列と同じキー列を使用します。アプライ・プログラムはこのキー値を使用して、自分がどのデータをソースの CD 表からターゲットに複製したのかをトラッキングします。主キー、固有索引、または固有索引として使用できる列の組み合わせを含んでいない OS/400 表を登録する場合は、その表を相対レコード番号 (RRN) を使用して登録する必要があります。RRN を使用して複製することを選択すると、CD 表とターゲット表の両方に、それぞれの行ごと固有の値を含む INTEGER 型の列 (IBMQSQ_RRN) が追加されます。この列に含まれているのは、ソース表の各行に対応する RRN です。

ソース表が再編成されないかぎり、RRN はソース表の行に対する主キーとして使用されます。ソース表が再編成されると、ソース表の各行の RRN が変更されます。したがって、CD 表およびターゲット表の行に含まれている RRN の値は、当該の行のソース表内での新しい位置を反映した正しい値ではなくなります。ソース表を (たとえば、削除された行を圧縮するために) 再編成すると、必ず DB2 DataPropagator for iSeries はそのソース表のセットになっているすべてのターゲット表に対してフル・リフレッシュを実行します。この理由から、RRN を主キーとして使用するターゲット表をやはり RRN を使用する他のターゲットと一緒にサブスクリプション・セットに入れて、RRN 以外で一意性を表すなんらかの因子を使用する表とはセットにしないでください。

レプリケーション・ソースとしてのビューの動作

レプリケーション用のビューを登録すると、それらは登録オプションを基礎表の登録定義から継承します。最も重要なことは、ビューが変更キャプチャー・レプリケーション用として登録されるか、あるいはフル・リフレッシュのみとして登録されるかが、ビューの基礎表によって決定されることです。以下のセクションでは、登録されたビューがさまざまなシナリオのレプリケーションにおいてどのように動作するのが説明されています。

- 『単一の表に対するビュー』
- 『複数の表の結合に対するビュー』

単一の表に対するビュー

基礎表がレプリケーション用に登録されている場合は、単一表に対するビューを登録できます。単一の登録済みの表に対するビューを登録すると、ビューはその基礎表が持っているレプリケーション・タイプを継承します。基礎表がフル・リフレッシュのみのレプリケーション用に登録されている場合は、ビューはフル・リフレッシュのみのレプリケーションを持ちます。そのビューを変更キャプチャー・レプリケーション用に登録することはできません。なぜなら、その基礎表は変更をトラッキングするための関連した CD 表を持っていないからです。基礎表が変更キャプチャー・レプリケーション用に登録されている場合は、ビューは変更キャプチャー・レプリケーションを持つため、フル・リフレッシュのみのレプリケーション用に登録することはできません。

変更キャプチャー・レプリケーション用に登録されている表に対するビューを登録すると、基礎表の CD 表に対するビューが作成されます。この CD ビューには、登録したビューから参照される列だけが含まれています。

ビュー内の列のサブセットは登録できません。なぜなら、ビュー内のすべての列が自動的に登録されるからです。

複数の表の結合に対するビュー

複数の表の結合に対するビューを登録する場合は、その結合内の表が 1 つでも登録されているかぎり、基礎表は登録済みの表であっても未登録の表であっても構いません。また、ソースとして登録されている CCD 表の内部結合も持てます。

結合をレプリケーション・ソースとして登録すると、DB2 レプリケーションは同一の SOURCE_OWNER 値と SOURCE_TABLE 値が含まれている複数の行を登録 (IBMSNAP_REGISTER) 表に追加します。これらの行は、各自の SOURCE_VIEW_QUAL 値によって区別されます。これらの各エントリーによって結合のコンポーネントが識別されます。

制約事項: CCD 表が組み込まれた結合を定義する場合は、その結合内の他のすべての表が CCD 表である必要があります。

ある結合ビューがレプリケーション・ソースとして存続できるようにするためには、その結合ビューを相関 ID を使用して作成する必要があります (単一表に対するビューの場合には相関 ID は不要です)。

例:

```
create view REGRES1.VW000 (c000,c1001,c2001,c2002,c1003) as
  select a.c000,a.c001,b.c001,b.c002,a.c003
  from REGRES1.SRC001 a, REGRES1.SRC005 b
  where a.c000=b.c000;
```

ここで、VW000 はビューの名前です。SRC001 および SRC005 は、ビューの一部である表です。C000、C001、C002、および C003 は、両方の表 (SRC001 と SRC005) の C000 列が等しいという条件の下でビューの一部となる列です。

ビューが継承するレプリケーションのタイプは、そのビューの基礎表の組み合わせによって決まります。各基礎表は次のいずれかです。

- 変更キャプチャー・レプリケーション用に登録されているもの
- フル・リフレッシュのみのレプリケーション用に登録されているもの
- 未登録のもの

表 2 には、基礎表のさまざまな組み合わせと、それぞれの組み合わせからソース・ビューと CD ビューが結果的にどのタイプになるのかが示されています。

表 2. ビューの場合の基礎表の組み合わせ

| 表 1 | 表 2 | 結合ビューおよび CD ビューの記述 |
|------------------------|------------------------|--|
| 変更キャプチャー用に登録されているもの | 変更キャプチャー用に登録されているもの | このビューは、変更キャプチャー・レプリケーション用に登録されたものである。この CD ビューには、表 1 の CD 表と表 2 の CD 表から参照される列が含まれている。 |
| 変更キャプチャー用に登録されているもの | フル・リフレッシュのみ用に登録されているもの | このビューは、変更キャプチャー・レプリケーション用に登録されたものである。この CD ビューには、表 1 の CD 表から参照される列と、表 2 から参照される列が含まれている。各レプリケーション・サイクルでは、表 1 に入っている列に対する変更だけが登録済みビューのターゲットにレプリケーションされる。 |
| フル・リフレッシュのみ用に登録されているもの | フル・リフレッシュのみ用に登録されているもの | このビューは、フル・リフレッシュのみのレプリケーション用に登録されたものである。CD ビューはない。 |
| フル・リフレッシュのみ用に登録されているもの | 未登録のもの | このビューは、フル・リフレッシュのみのレプリケーション用に登録されたものである。CD ビューはない。 |
| 変更キャプチャー用に登録されているもの | 未登録のもの | このビューは、変更キャプチャー・レプリケーション用に登録されたものである。この CD ビューには、表 1 の CD 表から参照される列と、表 2 から参照される列が含まれている。各レプリケーション・サイクルでは、表 1 に入っている列に対する変更だけが登録済みビューのターゲットにレプリケーションされる。 |
| 未登録のもの | 未登録のもの | このビューは有効なレプリケーション・ソースではないため登録できない。 |

複数のソース表がレプリケーション・ソースとして組み込まれているビューを定義する場合は、二重削除を防ぐための配慮が必要です。二重削除は、同一レプリケーション・

サイクルの間に両方とも 1 つのビューの一部である表から行を削除した場合に発生します。たとえば、CUSTOMERS 表と CONTRACTS 表を含むビューを作成したとします。二重削除は、同一のレプリケーション・サイクルで CUSTOMERS 表から 1 つの行を削除し、同様に (ビューの結合点から) それに対応する行を CONTRACTS 表からも削除した場合に発生します。ここで問題となるのは、その行が結合の 2 つのソース表から削除されているために、その行はビューに (基本ビューにも CD 表ビューにも) 表示されず、したがってこの二重削除がターゲットに複製できないことです。

二重削除を防ぐためには、結合内のソース表のいずれか 1 つに CCD 表を定義する必要があります。この CCD 表は、コンデンスされた非コンプリートの表でなければならず、さらにターゲット・サーバー上になければなりません。結合内のいずれかのソース表に対し、コンデンスされた非コンプリートの CCD 表を定義すれば、ほとんどの場合の二重削除問題は解決されます。なぜなら、この CCD 表の IBMSNAP_OPERATION 列を使用すれば削除を検出できるからです。サブスクリプション・サイクルの後 に実行するサブスクリプション・セットの定義に、単に SQL ステートメントを追加してくださいこの SQL ステートメントは、CCD 表内の IBMSNAP_OPERATION が “D” と等しくなるターゲット表からすべての行を除去します。

同一アプライ・サイクルで、CCD を持つソース表のある行が更新されたが、結合内のもう一方の表ではそれに対応する行が削除されたという場合には、更新と削除に関する問題がまだ発生する可能性があります。その結果、アプライ・プログラムは結合された表でその対応する行を検出できず、更新された値を複製できません。

表のビューをソースとして登録する

このセクションでは、DB2 表のビューをレプリケーション・ソースとして登録する方法を説明します。

前提条件:

- ソースとして登録するビューを処理させるキャプチャー・コントロール・サーバー上に、キャプチャー・コントロール表がすでに存在していなければならない。キャプチャー・コントロール表を作成する必要がある場合は、27 ページの『レプリケーション・コントロール表のセットアップ』を参照してください。
- ソース・ビューの名前は DB2 表の命名規則に従う必要がある。

制約事項:

- 非 DB2 リレーショナル表のビューは登録できない。
- 別のビューの上層のビューは登録できない。
- OS/400 では、SQL ステートメントは長さが 32,000 文字までに制限されているため、1 つのビュー当たりおよそ 2000 列までしか登録できない。正確な列数は、列名の長さによって決まります。
- ビューを定義されているすべての CCD 表は、レプリケーション・ソースとして登録するためにはコンプリートでコンデンスされたものでなければならない。

推奨: ビューをソースとして登録する前に、基礎表のうち少なくとも 1 つをソースとして登録してください。表の登録方法は、43 ページの『DB2 表をソースとして登録する』を参照してください。

プロシージャ:

以下の方法のいずれかを使用して、ビューを登録してください。

レプリケーション・センター

「ビューの登録」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

レプリケーション用のシステム・コマンド (OS/400)

ADDDPRREG システム・コマンドを使用します。このコマンドの構文およびパラメーターについての記述は、366 ページの『ADDDPRREG: DPR 登録の追加 (OS/400)』を参照してください。

ビューの登録時には、ビューの名前、ソース・サーバー、およびキャプチャー・スキーマを指定して、どのビューを登録するのかを示してください。ビュー用の登録オプションは、それらのビューが定義されているソース表の登録定義から導き出されます。登録オプションとそれらのデフォルトの完全なリスト、およびこれらのデフォルトを使用した方がよい場合と変更した方がよい場合についての説明は、47 ページの『ソース表用の登録オプション』を参照してください。基礎表に基づいてビューがどのレプリケーション・タイプ (変更キャプチャーまたはフル・リフレッシュのみ) を継承するのかについては、64 ページの『レプリケーション・ソースとしてのビューの動作』を参照してください。

CCD 表をソースとして保守する (IMS)

アプライ・プログラムによって移植されたのではなく外部的に移植された CCD 表、または IMS DataPropagator や DataRefresher などのプログラムによって保守されている CCD 表がある場合は、アプライ・プログラムがそれらの CCD 表をソースとして読み取れるように、また正しく機能できるように、それらの表を保守する必要があります。このセクションでは、CCD 表をレプリケーション・ソースとして保守する方法を説明します。

外部ツールによって移植された CCD 表を保守するためには、登録 (IBMSNAP_REGISTER) 表の 3 つの列 (CCD_OLD_SYNCHPOINT、SYNCHPOINT、および SYNCHTIME) を更新する必要があります。(登録表のこれらの列についての詳細は、505 ページの『*schema*.IBMSNAP_REGISTER』を参照してください。) これらの 3 つの列は、以下のタイプのイベントが発生するたびに更新する必要があります。

- はじめてフル・リフレッシュまたは CCD 表のロードが行われたとき
 - CCD_OLD_SYNCHPOINT を CCD 表の IBMSNAP_COMMITSEQ の最小値を表す値に設定する。

- SYNCHPOINT を CCD 表の IBMSNAP_COMMITSEQ の最大値を表す値に設定する。SYNCHPOINT を 0 に設定しないでください。順序付け用に独自の値を作成する場合は、初回用の SYNCHPOINT 値は 1 にしてください。
- SYNCHTIME を CCD 表の IBMSNAP_LOGMARKER の最大タイム・スタンプ値を表す値に設定する。
- フル・リフレッシュ後またはロード後に CCD 表に対してリフレッシュが行われたとき
 - CCD_OLD_SYNCHPOINT 値を変更してはならない。
 - SYNCHPOINT を CCD 表の IBMSNAP_COMMITSEQ の新しい最大値を表す値に設定する。
 - SYNCHTIME を CCD 表の IBMSNAP_LOGMARKER の新しい最大タイム・スタンプ値を表す値に設定する。
- 2 回目以降のフル・リフレッシュまたは CCD 表のロードが行われたとき
 - CCD_OLD_SYNCHPOINT を CCD 表の IBMSNAP_COMMITSEQ の最小値を表す値に設定する。
 - SYNCHPOINT を CCD 表の IBMSNAP_COMMITSEQ の最大値を表す値に設定する。
 - SYNCHTIME を CCD 表の IBMSNAP_LOGMARKER の最大タイム・スタンプ値を表す値に設定する。

重要: 上記では、CCD 表で IBMSNAP_COMMITSEQ および IBMSNAP_LOGMARKER に使用されている値は常に増えてゆく値であることを前提としています。アプライ・プログラムは、CCD_OLD_SYNCHPOINT 値が最新に適用された SYNCHPOINT 値より大きくならないかぎり、ソース CCD 表でフル・リフレッシュが実行されたことを検出しません。

関連概念:

- 259 ページの 『第 14 章 DB2 レプリケーション・センターの使用』

関連作業:

- 69 ページの 『第 4 章 ソースのサブスクライブ』

関連資料:

- 366 ページの 『ADDDPRREG: DPR 登録の追加 (OS/400)』

第 4 章 ソースのサブスクライブ

レプリケーション・ソースとして使用する表とビューを登録後、ターゲット表またはビューがソースからのソース・データと変更内容を受け取れるように、ターゲット表またはビューのサブスクリプションを定義できます。本章で説明する管理タスクは、キャプチャー・プログラムとアプライ・プログラムが、ソース・データをコピーするため、あるいは変更データを取り込んでそれをターゲット表に適切なインターバルで複製するために使用するコントロール情報のセットアップを支援します。

本章は、以下のセクションで構成されています。

- 『ソースおよびターゲットの分類方法の計画』
- 72 ページの『サブスクリプション・セットの作成』
- 74 ページの『サブスクリプション・セットのオプション処理』
- 82 ページの『サブスクリプション・セット内におけるソース表およびビューのターゲット表およびビューへのマッピング』
- 85 ページの『ターゲット・タイプの選択』
- 98 ページの『すべてのターゲット表タイプに共通のプロパティ』

ソースおよびターゲットの分類方法の計画

どのターゲットがどのソースをサブスクライブするかを定義する前に、ソースとターゲットの分類方法を計画する必要があります。DB2 レプリケーションは、グループ単位でソースからターゲットへのマッピングを処理します。これらのグループは、同一のキャプチャー・プログラムによって処理される 1 つ以上のソースと、ソース・データのすべてまたは一部をサブスクライブし、同一のアプライ・プログラムによって処理される 1 つ以上のターゲットで構成されています。これらのグループを、サブスクリプション・セットと呼び、ソースからターゲットへのマッピングをサブスクリプション・セット・メンバーと呼びます。

サブスクリプション・セットを計画するときは、以下の規則と制約に注意してください。

- サブスクリプション・セットは、ソース・サーバーをターゲット・サーバーにマップする。サブスクリプション・セット・メンバーは、ソース表またはビューを、ターゲット表またはビューにマップします。サブスクリプション・セットとそのメンバーは、アプライ・コントロール・サーバーに保管されます。

- アプライ・プログラムは、サブスクリプション・セットのすべてのメンバーを単一グループとして処理する。このため、サブスクリプション・セットのいずれかのメンバーで、何らかの理由でフル・リフレッシュ・コピーが必要な場合、セット全体のすべてのメンバーがリフレッシュされます。
- 1つのセットのメンバーにおいて、すべてのソース表のキャプチャー・スキーマは同じでなければならない。
- OS/400 システムでは、1つのサブスクリプション・セットのメンバーにおいて、すべてのソース表は同じジャーナルに記録されなければならない。
- IMS DataPropagator で作成され、サブスクリプション・セットのメンバーである外部 CCD 表すべてのキャプチャー・スキーマは同じでなければならない。

固有のアプライ修飾子を持つ単一のアプライ・プログラムは、サブスクリプション・セットを1つでもあるいは多数でも処理できます。単一のサブスクリプション・セットには、サブスクリプション・セット・メンバーを1つでもあるいは多数でも入れることができます。以下のセクションでは、アプライ・プログラムごとに少数または多数のセットを保有する場合と、サブスクリプション・セットごとに少数または多数のサブスクリプション・セット・メンバーを保有する場合のトレードオフについて説明します。

サブスクリプション・セット・メンバー数の計画

サブスクリプション・セットにメンバーを追加するとき、ソースとターゲットのすべてのペア (サブスクリプション・セット・メンバー) を1つのサブスクリプション・セットにまとめるか、それぞれのペアごとに別々のサブスクリプション・セットを作成するか、あるいはそれぞれにかなりの数のペアが含まれた、少数のサブスクリプション・セットを作成するかを決定しなければなりません。

アプライ・プログラムは、サブスクリプション・セットのメンバーを1つの (論理) トランザクションで複製するため、以下のどちらの状態においても、複数のメンバーを1つのサブスクリプション・セットにまとめる必要があります。

- ソース表に相互に論理的な関連がある場合。
- ターゲット表に参照保全制約がある場合。

複数のメンバーを1つのサブスクリプション・セットにまとめると、すべてのメンバーのレプリケーションを必ず同時に開始することができます。さらに、サブスクリプション・セットを処理するのに必要なデータベース接続の数と、レプリケーション環境を保守するための管理オーバーヘッドが削減されます。サブスクリプション・セットに SQL ステートメントまたはストアード・プロシージャが含まれている場合、これらのステートメントまたはプロシージャを使用して、そのサブスクリプション・セットのすべてのメンバーを処理できます。

サブスクリプション・セットの表の間に、論理的または参照保全のリレーションシップがない場合、その表を1つまたは幾つかのサブスクリプション・セットにまとめることができます。サブスクリプション・セットの数を制限する主な理由は、レプリケーシ

ン環境の管理を簡単にすることです。ただし、サブスクリプション・セットの数を増やすと、レプリケーションの失敗による影響は最小になります。

アプライ・プログラムが失敗する原因となるエラーをより簡単に突き止められるようにするには、少数のメンバーしかサブスクリプション・セットに追加しないようにします。メンバーが少数なら、多数のメンバーがセットに含まれている場合より迅速に問題のソースを見付けることができます。サブスクリプション・セットの 1 つのメンバーで障害が起こった場合、そのセットの他のメンバーに適用されたデータはすべてロールバックされます。そのため、すべてのメンバーがサイクルを完了しない限り、どのメンバーも正常にサイクルを完了できません。アプライ・プログラムは、失敗したサブスクリプション・セットをその最後の正常なコミット・ポイントまでロールバックします。このコミット・ポイントは、アプライ・プログラムを始動したときに **commit_count** キーワードを指定していれば、現行のアプライ・サイクル内にある可能性があります。

アプライ修飾子ごとのサブスクリプション・セット数の計画

サブスクリプション・セットを定義するとき、そのサブスクリプション・セット用のアプライ修飾子を指定します。アプライ修飾子は、アプライ・プログラムのインスタンスを 1 つ以上のサブスクリプション・セットに関連付けます。それぞれのサブスクリプション・セットを処理するのは、ただ 1 つのアプライ・プログラムですが、アプライ・プログラムはそれぞれ 1 つ以上のサブスクリプション・セットを、1 つ 1 つのアプライ・サイクル内で処理できます。

アプライ・プログラムのインスタンス (それぞれに独自のアプライ修飾子があります) は必要な数だけ実行でき、アプライ・プログラムはそれぞれサブスクリプション・セットに必要な数だけ処理できます。基本オプションが 2 つあります。

- アプライ修飾子をそれぞれ 1 つのサブスクリプション・セットに関連付ける (個々のアプライ・プログラムが、厳密に 1 つのサブスクリプション・セットを処理する)。

速度が重要な場合は、幾つかのアプライ修飾子間にセットを分散でき、こうすることで、アプライ・プログラムの幾つかのインスタンスを同時に実行できます。1 つのアプライ・プログラムで 1 つのサブスクリプション・セットを処理するように決めていると、このアプライ・プログラムの **OPT4ONE** 始動オプションを使用できます。このオプションは、サブスクリプション・セットのコントロール表情報をメモリーにロードするものです。このオプションを使用すると、アプライ・プログラムは、サブスクリプション・セット情報のコントロール表をアプライ・サイクルごとには読み取りません。したがって、アプライ・プログラムのパフォーマンスが向上します。ただし、実行するアプライ・プログラムが多くなれば、それらが使用するシステム・リソースも多くなり、全体としてのパフォーマンスは低下する可能性があります。

- アプライ修飾子をそれぞれ複数のサブスクリプション・セットに関連付ける (個々のアプライ・プログラムが、多数のサブスクリプション・セットを処理する)。

複数のアプライ修飾子を使用すれば、単一ユーザー ID からアプライ・プログラムの複数のインスタンスを実行できます。

アプライ・プログラムでは、指定されたアプライ修飾子に対応するセットすべてを、可能な限り現状のまま保持しようとしています。アプライ・サイクルが開始すると、アプライ・プログラムは含まれる現行データが最も少ないサブスクリプション・セットを判別し、まずそのセットの処理を開始します。

速度が主要なゴールでない場合は、1つのアプライ修飾子が指定された膨大な数のサブスクリプション・セットを複製することがあります。たとえば、営業時間後まで待ってから複製する場合に、これは大変適切なオプションということもあります。

1つのアプライ・プログラムで複数のサブスクリプション・セットを処理する欠点は、そのアプライ・プログラムがサブスクリプション・セットを順次処理することです。したがって、全体のレプリケーション待ち時間は増加する可能性があります。

一部のサブスクリプション・セットに対して特定の要件がある場合、この2つのオプションを組み合わせることができます。たとえば、1つのアプライ・プログラムにほとんどのサブスクリプション・セットを処理させることができるので、関連するサブスクリプション・セットをまとめて処理すれば、アプライ・プログラムを有効に利用できるでしょう。そして、別のアプライ・プログラムに単一のサブスクリプション・セットを処理させることができるので、そのサブスクリプション・セットのレプリケーション待ち時間を確実に最小にできます。さらに、アプライ・プログラムの2つのインスタンスを使用すれば、サブスクリプション・セットの並行処理全体が強化されます。

サブスクリプション・セットの作成

登録済みのソースからデータを複製する前に、アプライ・プログラムが1つの集合として処理するサブスクリプション・セット・メンバー（ソースからターゲットへのマッピング）のコレクションである、サブスクリプション・セットを作成しなければなりません。このセクションでは、それぞれのサブスクリプション・セットごとに定義するプロパティーについて説明します。セットに追加するメンバーごとに適用されるこれらのプロパティーは、データのレプリケーション先とレプリケーション元のサーバーを定義し、使用するキャプチャー・プログラム（登録済みソースのキャプチャー・スキーマを基にしています）とアプライ・プログラム、およびアプライ・プログラムがセットを処理する時期と方法も定義します。

サブスクリプション・セット・メンバーをサブスクリプション・セットに追加する必要はありません。代わりに、ソースからターゲットへのマッピングを含まない空のセットを作成できます。以下のような理由から、空のセットの作成が必要になることがあります。

- 後でメンバーをセットに追加する計画があり、メンバーを追加するまではサブスクリプション・セットをアクティブにする計画はない。
- 空のサブスクリプション・セットが処理に適格であるとき、いつでも SQL ステートメントまたはストアド・プロシージャを呼び出すために、アプライ・プログラムでそのセットを処理する。

前提条件:

1. サブスクリプション・セット用に、アプライ・コントロール表をアプライ・コントロール・サーバーで作成しなければならない。
2. サブスクリプション・セットにサブスクリプション・セット・メンバーを追加する前に、ソースとして使用する表またはビューを登録しなければならない。レプリケーションのためにソースを登録する必要がある場合は、43 ページの『第 3 章 表およびビューをレプリケーション・ソースとして登録する』の説明を読んでください。セットの分類方法について検討する必要もあります。セットを計画する必要がある場合、詳細については、69 ページの『ソースおよびターゲットの分類方法の計画』を参照してください。

プロシージャー:

サブスクリプション・セットを作成するには、以下の 2 つの方法のどちらかを使用できます。

レプリケーション・センター

「サブスクリプション・セットの作成 (Create subscription sets)」ノートブックを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

レプリケーション用のシステム・コマンド (OS/400)

ADDDPRSUB システム・コマンドを使用します。このコマンドの構文およびパラメーターについての記述は、374 ページの『ADDDPRSUB: DPR サブスクリプション・セットの追加 (OS/400)』を参照してください。

サブスクリプション・セットを作成するには、以下の基本的な特性を指定します。

アプライ・コントロール・サーバー別名

サブスクリプション・セットを処理するアプライ・プログラム用のコントロール表を含むサーバーのローカルの別名。どのデータベースにあるアプライ・コントロール・サーバーにも、同一の別名を定義します。このサーバーからレプリケーション・センターまたはアプライ・プログラムを実行して、レプリケーション・センターがアプライ・コントロール表を適切に移植するようにし、さらにどのアプライ・プログラムも標準の別名を使用して適切なサーバーに接続するようにします。

サブスクリプション・セット名

サブスクリプション・セットの名前。サブスクリプション・セットを処理するアプライ・コントロール・サーバーにおいて、セット名は指定されたアプライ修飾子に対して固有でなければなりません。名前のは長さは 18 文字まで可能です。

アプライ修飾子

新規または既存のアプライ修飾子の名前。サブスクリプション・セットを処理するアプライ・プログラムを示します。同じアプライ修飾子を使用して、複数のサブスクリプション・セットを処理できます。同じアプライ修飾子を持つサ

ブスクリプション・セットは、同じアプライ・コントロール・サーバーで定義しなければなりません。新規のアプライ修飾子を作成する場合は、アプライ修飾子の命名方法の規則について、321ページの『第16章 レプリケーション・オブジェクトの命名規則』を参照してください。

キャプチャー・コントロール・サーバー別名

サブスクリプション・セットに登録済みのソースを処理するキャプチャー・プログラム用のコントロール表を含むサーバーの別名。どのデータベースにあるキャプチャー・コントロール・サーバーにも、同一の別名を定義します。このサーバーからレプリケーション・センターまたはアプライ・プログラムを実行して、レプリケーション・センターがキャプチャー・コントロール表とアプライ・コントロール表を適切に移植するようにし、さらにどのアプライ・プログラムも標準の別名を使用して適切なサーバーに接続するようにします。

キャプチャー・スキーマ

キャプチャー・スキーマの名前。サブスクリプション・セットに登録済みのソースを定義するキャプチャー・コントロール表のセットを示します。サブスクリプション・セットのすべてのソース表は、同じサーバーに常駐していなければならない。1つのキャプチャー・プログラムだけがソース表への変更をキャプチャーすることができます。

ターゲット・サーバー別名

アプライ・プログラムがソースからの変更を複製する先の表またはビューが含まれるターゲット・サーバーの名前。どのデータベースにあるターゲット・サーバーにも、同一の別名を定義します。このサーバーからレプリケーション・センターまたはアプライ・プログラムを実行して、レプリケーション・センターがアプライ・コントロール表を適切に移植するようにし、さらにどのアプライ・プログラムも標準の別名を使用して適切なサーバーに接続するようにします。

サブスクリプション・セットを作成するときは、アプライ・プログラムがセットを処理する方法について、デフォルトの設定値を使用できます。あるいは、サブスクリプションのプロパティを、レプリケーションの要求を満たすように変更できます。サブスクリプション・セット用の処理オプションとそのデフォルト、およびデフォルトの使用または変更が必要な場合についての説明の完全なリストは、『サブスクリプション・セットのオプション処理』を参照してください。

サブスクリプション・セットのオプション処理

このセクションでは、アプライ・プログラムがサブスクリプション・セットを処理する方法を指定するために定義可能なプロパティについて説明します。さらに、レプリケーションの要求に基づいて選択すべき設定値を決定する際にも、このセクションは役立ちます。

- 『セットがアクティブかどうかの指定』
- 『アプライ・プログラムが取り出すデータに相当する分数の指定』
- 78 ページの『アプライ・プログラムがセットのメンバーへの変更を複製する方法の指定』
- 80 ページの『サブスクリプション・セット用の SQL ステートメントまたはストアド・プロシージャの定義』
- 80 ページの『サブスクリプション・セットのレプリケーションのスケジューリング』

セットがアクティブかどうかの指定

デフォルト: 非アクティブ

アプライ・プログラムがサブスクリプション・セットの処理を開始するかどうかを指定できます。サブスクリプション・セットを活動化すると、アプライ・プログラムは、そのセットのフル・リフレッシュを開始します。活動化レベルは以下の 3 つから選択しませんが。

アクティブ

アプライ・プログラムは、その次のサイクル内でセットを処理します。アプライ・プログラムに次の実行時でセットを処理させる場合は、セットを活動化します。後になっても、メンバーをセットに追加できます。セットを活動化すると、ユーザーがセットを非活動化するまで、セットはアクティブのままです。アプライ・プログラムはその処理を続けます。

非アクティブ

アプライ・プログラムはセットを処理しません。アプライ・プログラムでセットを処理する準備ができていない場合は、セットを非アクティブにしておきます。

1 回のみアクティブ

アプライ・プログラムは次のサイクル内でセットを処理し、その後セットを非活動化します。セットを 1 回だけ実行したい場合は、このオプションを指定します。このオプションを選択する前に、必ずすべてのサブスクリプション・セット・メンバーを追加してください。その理由は、サブスクリプション・セットを再活動化しない限り、アプライ・プログラムは後で追加したメンバーを処理しないためです。

アプライ・プログラムが取り出すデータに相当する分数の指定

デフォルト: 20 分

アプライ・プログラムが各アプライ・サイクル内でレプリケーション・ソースから取り出すデータを、それに相当するおおよその分数で指定できます。この指定が役に立つ状態が幾つかあります。

- 1 回のサブスクリプション・セット・サイクル内で処理されるデータ量が大量の場合。
1 回のアプライ・サイクルで大量の変更ブロックを複製するサブスクリプション・セットは、予備ファイルまたはログ (ターゲット・データベースの場合) でのオーバーフローの原因となります。たとえば、アプライ・プログラムをバッチ処理するシナリオでは、レプリケーションを必要とするエンキューされたトランザクションのバックログが大量に生成される可能性があります。
- ネットワークの停止が長引くと、大量のデータ・ブロックが CD 表に累積され、アプライ・プログラムの予備ファイルとターゲットのログがオーバーフローする可能性がある。

指定する分数を、データ・ブロックと呼びます。指定するデータ・ブロッキングの値は、サブスクリプション・セット (IBMSNAP_SUBS_SET) の表で MAX_SYNCH_MINUTES 列に保管されます。データの累積がデータ・ブロックのサイズより大きいと、アプライ・プログラムは 1 つのアプライ・サイクルを幾つかのミニサイクルに変換します。リソースが、指定されたブロッキング因数をハンドルするにはまだ不十分な場合、アプライ・プログラムはデータ・ブロックのサイズを削減し、使用可能なシステム・リソースに一致させます。より小さいデータ集合を取り出すことにより、アプライ・プログラムはネットワーク負荷と取り出されたデータ用に一時的に必要なスペースの両方を減少させます。

例: アプライ・プログラムがミニサイクル当たりせいぜい 10 分相当のデータしか取り出さないように指定すると、最後のミニサイクルの約 10 分間以内で、アプライ・プログラムはかなりの量のコミット済みデータをソースの CD 表から取り出すようになります。

ログ・ファイルと予備ファイルをオーバーフローさせない点に加えて、ミニサイクルには、他にも幾つかの利点があります。レプリケーション・サイクルでエラーが発生している場合、アプライ・プログラムは、失敗したミニサイクルで行った変更のみをロールバックしなければなりません。ミニサイクルでレプリケーションが失敗した場合、アプライ・プログラムは、最後に正常に実行されたミニサイクルからサブスクリプション・セットを処理しようとはしますが、これで大量の変更データを処理できる場合は、かなりの時間を節約できます。77 ページの図 5 で、変更データが変更のサブセットに分割される仕組みを示します。

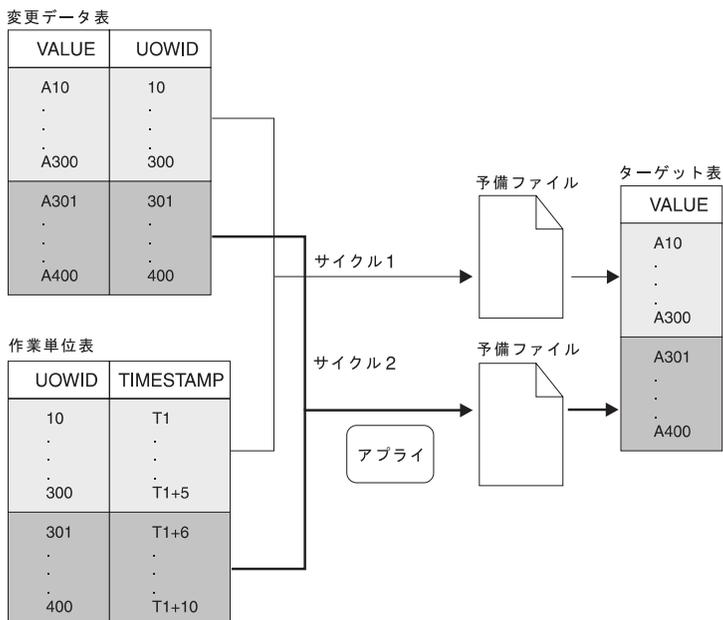


図5. データ・ブロッキング：ネットワークのトラフィック量を少なくするには、データ・ブロック値を指定します。

設定する分数が小さければ、インターバル内に生じるサブスクリプション・セットのトランザクションをすべてコピーし、ミニサイクル内での予備ファイルやログのオーバーフローを防ぐことができます。

重要 (OS/400 の場合): 各アプライ・サイクル内で複製されるデータの総量が 400 万行を超えないようにしてください。

データを処理するとき、アプライ・プログラムは以下のどのアクションも行いません。

- 作業単位を分割する (すなわち、コミットされないで長時間実行中のバッチ・ジョブは、データ・ブロッキング因数で分割できない)。
- 以前コミットされたミニサブスクリプション・サイクルをロールバックする。
- フル・リフレッシュ中にデータ・ブロッキング因数を使用する。

アプライ・プログラムが参照保全性のあるターゲット表をロードする方法の決定

セットのターゲット表の間で参照保全が要求される場合、ターゲット表の初期ロード時にアプライ・プログラムがソース・データを処理する方法を選択しなければなりません。デフォルトでは、アプライ・プログラムはソースのすべての行を読み取り、それをメモリーに保管し、その後それらの行をターゲット表に挿入することによって、ターゲットのフル・リフレッシュを実行します。しかし、ターゲット表を初期ロードする方法

には他のオプションがあります。サブスクリプション・セットを作成するとき、あるいはソースからターゲットへのマッピング (メンバー) をセットごとに定義するとき、これらを決定することはありません。すなわち、ターゲットのロード方法は、アプライ・プログラムの始動パラメーターを設定するとき決定します。サブスクリプション・セット・メンバーを定義するとき、それぞれのメンバーごとにレプリケーション要件を考慮する必要があり、それぞれのメンバーを処理するアプライ・プログラムで使用する始動オプションを決定できます。

これらの参照保全リレーションシップを作成する時期について、以下の 2 つの選択肢のどちらかを検討してください。

- ターゲット表が移植される前。

この場合は、ターゲット表の抽出およびロードの全段階を通じて、ソース表への変更が行われないことが必要です。また、この初期の移植中にロード処理の速度を確保し、参照制約チェックをバイパスするため、LOADX 始動オプションを使用してアプライ・プログラムを始動しなければなりません。アプライ始動オプション LOADX を使用しない場合、ターゲット表への挿入が失敗する可能性があります。

- アプライ・プログラムがターゲット表を完全に移植し、それらの表の該当するセットに変更を適用する 1 つのサイクルを正常に完了した後。

参照保全制約をこれらの表に追加する時期を待つ場合、ターゲット表のロード中に、ソース表でさらに変更を行うことができるという利点があります。バイパスする必要があるという制約がないため、始動オプション LOADX を使用してもしなくても、アプライ・プログラムを始動できます。始動オプション LOADX を使用するよりも、一般的にフル・リフレッシュはかなり速度があります。ターゲット表の初期の移植中、ターゲットは参照保全リレーションシップの点で相互に同期がとれていないことがあります。しかし、ロード中に、セットの変更はすべて取り込まれます。アプライ・プログラムが変更の入った最初のセットを複製した後、すべてのターゲット表には同じトランザクションが入り、参照保全性を持つようになります。この時点でセットを非活動化して参照保全制約を追加し、その後セットを再び活動化できます。

ターゲット表の初期ロード方法のために提供されている始動オプションについての詳細は、169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』を参照してください。

アプライ・プログラムがセットのメンバーへの変更を複製する方法の指定

サブスクリプション・セットに変更キャプチャー・レプリケーションが含まれるとき、アプライ・プログラムがセットのソースからターゲットへのすべてのマッピングへの変更をレプリケーションする方法を決定できます。ターゲット表の初期ロード後、アプライ・プログラムは CD (または CCD) 表の読み取りを開始し、変更を予備ファイルに収集します。それぞれの CD (または CCD) 表ごとに、アプライ・プログラムは別々の予備ファイルを作成します。そこでアプライ・プログラムは予備ファイルから変更を読み取り、変更をターゲット表にアプライします。以下の 3 つの方法のいずれかで、これを実行します。

- 表モード処理の使用。
- トランザクション・モード処理の使用。
- 表モードとトランザクション・モードが混合した処理の使用。サブスクリプション・セットのターゲット表タイプに依存します。

サブスクリプション・セットの処理タイプを指定すると、ターゲット表またはビューへの変更をアプライ・プログラムがコミットする頻度をコントロールできます。アプライ・プログラムは、それぞれのサブスクリプション・セット・メンバーごとに 1 回、あるいは相当数のトランザクションをアプライした後にコミットできます。コミットが 1 回の場合、サブスクリプション・セットの待ち時間を削減できますが、コミットが複数の場合は、アプライ・プログラムがオリジナルのコミット・シーケンスのデータをアプライできます。

表モード

アプライ・プログラムは CD (または CCD) 表の予備ファイルからすべての変更を読み取り、対応するターゲット表に変更をアプライしてから、次の CD (または CCD) 表の処理を開始します。セット内のすべての CD (または CCD) 表からの変更の読み取りとアプライが完了したら、アプライ・プログラムは DB2 コミットを実行して、サブスクリプション・セット内のすべてのターゲット表への変更をすべてコミットします。

トランザクション・モード

アプライ・プログラムは直ちにすべての予備ファイルを開き、予備ファイルからの変更を同時に処理します。変更がターゲット表に適用される順序は、トランザクションがソース表で実行された順序です。アプライ・プログラムは、始動時に指定されたインターバルで DB2 コミットを実行します。サブスクリプション・セットのターゲット表に参照保全制約がある場合は、このタイプの処理を使用してください。

どのサブスクリプション・セットの場合でもアプライ・プログラムがトランザクション・モード処理を使用するように指定できます。しかし、その指定で変更されるのは、セットにユーザー・コピー・ターゲット表とポイント・イン・タイム・ターゲット表が含まれる場合のアプライ・プログラムの動作のみで、以下のタイプのターゲット表がセットに含まれる場合の動作は変更されません。

- CCD ターゲット表。CCD 表がソースとして含まれるセットは、常に表モードで処理されます。
- ソース表が CCD 表である場合、それに対応するターゲット表。 CCD 表が含まれるセットは、常に表モードで処理されます。
- レプリカ・ターゲット表。レプリカ表が含まれるセットは、常にトランザクション・モードで処理されます。

サブスクリプション・セット用の SQL ステートメントまたはストアド・プロシージャの定義

アプライ・プログラムがサブスクリプション・セットを処理するたびに実行される SQL ステートメントまたはストアド・プロシージャを定義できます。これらのステートメントは、CCD 表のプルーニング、またはターゲットにアプライされる前のソース・データの取り扱いに役立ちます。SQL ステートメントまたはストアド・プロシージャの実行時期と実行場所を指定できます。

- アプライ・プログラムがデータをアプライする前にキャプチャー・コントロール・サーバーで。
- アプライ・プログラムがデータをアプライする前にターゲット・サーバーで。
- アプライ・プログラムがデータをアプライした後にターゲット・サーバーで。

アプライ・プログラムは、ステートメントまたはプロシージャを上記にリストした順序で処理します。

レプリケーション・センターを使用して SQL ステートメントをサブスクリプション・セットに追加するときは、「SQL ステートメントの追加」ウィンドウまたは「プロシージャ呼び出し」ウィンドウで **PREPARE ステートメント** をクリックし、ステートメントの構文を検査します。

サブスクリプション・セットのレプリケーションのスケジューリング

アプライ・プログラムがサブスクリプション・セットを処理する頻度をコントロールできるので、その結果、ターゲット表のデータをどのくらいで最新にするかをコントロールできるようになります。時間に基づくスケジューリングまたはイベントに基づくスケジューリングを使用して、どのくらいの頻度でサブスクリプション・セットを処理対象として適格とするかをコントロールでき、あるいはこれらのスケジューリング・オプションを一緒に使用することもできます。アプライ・プログラムは、サブスクリプション・セットが処理対象として適格になると、その処理を開始します。たとえば、ある 1 日のインターバルを設定する一方で、あるイベントによってサブスクリプション・サイクルを起動するよう指定することもできます。これらのスケジューリング・オプションの両方を使用する場合、サブスクリプション・セットはスケジュールされた時刻とイベント発生時刻の両方で処理対象として適格になります。

アプライ・プログラムが処理するサブスクリプション・セットのインターバル内またはイベントとイベントの間で、複製されるデータ量が膨大な場合、特定のサブスクリプション・セットを処理対象として適格にすることは可能です。しかし、アプライ・プログラムがそれを処理できるのは、先行するインターバルまたはイベントで、すべてのサブスクリプション・セットへのデータのアプライが終了してからです。サブスクリプション・セットの処理が終了すると、アプライ・プログラムは直ちに次の処理に適格なサブスクリプション・セットの処理を開始します。このケースでは、レプリケーション待ち時間が予想とは異なるかもしれませんが、失われるデータはありません。

時間に基づくスケジューリング

セットを処理する時期をコントロールする最も単純な方法は、時間に基づくスケジューリング (相対タイミングまたはインターバル・タイミングとしても知られています) を使用することです。特定の開始日付、時間、およびインターバルを決定します。インターバルは、特定の値 (1 分～1 年) または連続した値にすることができますが、時間インターバルはおおよその値になります。アプライ・プログラムは、作業負荷とリソースの可用性に基づいて、可能な限りすみやかにサブスクリプション・セットの処理を開始します。ある時間インターバルを選択しても、レプリケーションの頻度が正確にそのインターバルになるとは限りません。連続したタイミングを指定すると、アプライ・プログラムは可能な限り頻繁にデータを複製します。

イベントに基づくスケジューリング

イベントに基づくスケジューリング (イベント・タイミングとしても知られています) を使用してデータを複製するために、サブスクリプション・セットを定義するときにイベント名を指定できます。イベントの発生時に、アプライ・プログラムにそのイベントを認識させるには、イベント名に対応するタイム・スタンプを持つサブスクリプション・イベント (IBMSNAP_SUBS_EVENT) 表も移植しなければなりません。アプライ・プログラムはイベントを検出するとレプリケーションを開始します。

サブスクリプション・イベント表には、表 3 で示されているように、4 つの列がありません。

表 3. サブスクリプション・イベント表

| EVENT_NAME | EVENT_TIME | END_OF_PERIOD | END_SYNCHPOINT |
|------------|----------------------------|----------------------------|----------------|
| END_OF_DAY | 2002-05-01-17.00.00.000000 | 2002-05-01-15.00.00.000000 | |

EVENT_NAME は、サブスクリプション・セットの定義時に指定するイベントです。EVENT_TIME は、アプライ・プログラムがセットの処理を開始する時刻を示すタイム・スタンプです。END_OF_PERIOD は、該当時刻より後の更新が後のイベントまたは時刻まで据え置かれることを指定するオプション値です。END_SYNCHPOINT は、該当するログ・シーケンス番号より後の更新が後のイベントまたは時刻まで据え置かれることを指定するオプション値です。END_OF_PERIOD と END_SYNCHPOINT の両方の値を指定すると、END_SYNCHPOINT の値が優先されます。EVENT_TIME の値はアプライ・コントロール・サーバーのクロックにより設定されますが、END_OF_PERIOD の値は ソース・サーバーのクロックにより設定されます。2 つのサーバーが別の時間帯にある場合、この区別は重要です。

表 3 によると、END_OF_DAY というイベントの場合、EVENT_TIME のタイム・スタンプ値 (2002-05-01-17.00.00.000000) は、アプライ・プログラムがサブスクリプション・セットの処理を開始する時刻です。END_OF_PERIOD のタイム・スタンプ値 (2000-05-01-17.00.00.000000) は、更新が複製されなかった後、次の日のサイクルで複製される時刻です。つまり、イベントは 3 時前に作成されたすべての未確定の更新を複製して、それに続くすべての更新を延期します。

ユーザーまたはアプリケーションは、行を表に INSERT する SQL の INSERT ステートメントを使用してイベントをサブスクリプション・イベント (IBMSNAP_SUBS_EVENT) 表に通知して、イベントを活動化しなければなりません。たとえば、現行のタイム・スタンプに 1 分加算した値を使用して、EVENT_NAME に指定したイベントを起動します。このイベントに結び付けられたサブスクリプション・セットはすべて、1 分内で実行するのに適したものとなります。フル・リフレッシュと変更キャプチャー・レプリケーション両方の目的で、イベントを手動で通知しなければなりません。

イベントは、翌週、翌年、または毎週土曜日のように、前もって通知することができません。アプライ・プログラムが実行されている場合、アプライ・プログラムは指定されたおおよその時刻に処理を開始します。アプライ・プログラムは、指定された時刻に停止しており、後で再始動されると、サブスクリプション・イベント表をチェックして、通知されたイベントのサブスクリプション・セットの処理を開始します。

アプライ・プログラムはこの表を整理しません。ユーザーはこの表を移植して保守しなければなりません。さらに、レプリケーション・センターを使用してサブスクリプション・イベント表を更新することはできません。この表にイベントを追加するには、SQL ステートメントを実行するか自動手順を定義しなければなりません。

例:

```
INSERT INTO ASN.IBMSNAP_SUBS_EVENT
      (EVENT_NAME, EVENT_TIME)
      VALUES ('EVENT01', CURRENT_TIMESTAMP + 1 MINUTES)
```

アプライ・プログラムがサブスクリプション・セットを処理した最新の時点 (サブスクリプション・セットのコントロール表の LASTRUN 列にある値で指定) より前に発生するイベントは、期限切れのイベントと見なされて無視されます。そのため、アプライ・プログラムが実行中である場合は、期限切れのイベントを通知することを避けるため、時間的にわずかに先のイベントを通知しなければなりません。

サブスクリプション・セット内におけるソース表およびビューのターゲット表およびビューへのマッピング

サブスクリプション・セット内で、アプライ・プログラムがそのセットの処理中にグループとして処理するソースからターゲットへのマッピングを追加できます。ソースからターゲットへのこれらのマッピングを、サブスクリプション・セット・メンバーと呼びます。サブスクリプション・セット・メンバーを定義するとき、ソース・データをサブスクライブするターゲット表またはビューを指定し、複製されたデータのターゲットでの表示方法を定義できます。

前提条件:

ソースでの変更をサブスクライブするターゲットをセットアップする前に、ソースとして使用する表またはビューを登録しなければなりません。レプリケーション用のソースをまだ登録していない場合は、43ページの『第3章 表およびビューをレプリケーション・ソースとして登録する』の説明を読んでそれに従ってください。さらに、サブスクリプション・セットの作成や、セットに追加するメンバー数の計画も必要です。サブスクリプション・セットの作成が必要な場合は、72ページの『サブスクリプション・セットの作成』を参照してください。サブスクリプション・セット・メンバーの計画が必要な場合は、70ページの『サブスクリプション・セット・メンバー数の計画』を参照してください。

制約事項:

- **DB2** レプリケーションは非 DB2 リレーショナル表のビューをソースとしてサポートする。
- ターゲット・ビューを定義する場合、そのビューは挿入可能なビューでなければならない。すなわち、ビューのすべての列は更新可能でなければならない。ビューの全選択にキーワード **UNION ALL** を組み込むことができません。
- レプリケーション・センターを使用する場合、該当する列がターゲット表にまだ存在していなければ、サブスクリプション・セット・メンバーにその列を追加できない。
- **Windows、UNIX、z/OS** の場合: それぞれのサブスクリプション・セットごとに、最大 200 のメンバーを定義できます。
- **OS/400** の場合: それぞれのサブスクリプション・セットごとに、最大 78 のメンバーを定義できます。

プロシージャー:

サブスクリプション・セット・メンバーを追加するのに、以下の 2 つの方法のどちらかを使用できます。

レプリケーション・センター

以下のノートブックのいずれかを使用します。

- サブスクリプション・セットの作成。サブスクリプション・セットを作成するときは、このノートブックを使用します。
- サブスクリプション・セット・プロパティ。すでにサブスクリプション・セットを作成済みで、このセットに 1 つ以上のサブスクリプション・セット・メンバーを追加する場合は、このノートブックを使用します。
- サブスクリプション・セットにメンバーを追加する。1 つのメンバーを複数のサブスクリプション・セットに追加するには、このノートブックを使用します。たとえば、このノートブックを開いて 4 つのサブスクリプション・セットを選択した場合、1 つのメンバーをそれぞれのセットに追加できます。各メンバーは同じソースを使用する必要があります。

詳細については、レプリケーション・センターのオンライン・ヘルプを参照してください。

レプリケーション用のシステム・コマンド (OS/400)

ADDDPRSUBM システム・コマンドを使用します。このコマンドの構文およびパラメーターについての記述は、390 ページの『ADDDPRSUBM: DPR サブスクリプション・セットのメンバーの追加 (OS/400)』を参照してください。

ソースをターゲットにマップするには、ソースとして使用する登録済みの表またはビューについて、以下の情報を指定します。

- ソース表またはビュー、およびターゲット表またはビュー (ターゲット表の表スペースと索引も含まれる)。
- ターゲット表のタイプ。
- ターゲット表に複製するソース表の登録済みの列。

レプリケーション・センターを使用してソースをターゲットにマップするとき、LOB 列と DATALINK 列は列マッピングに自動的に組み込まれません。これらの列は明示的に選択しなければなりません。

- ターゲット表に複製するソース表の行 (行を指定するには WHERE 文節を組み込む)。

選択したソースを DB2 ターゲットにマップするには、ターゲット表またはビューについて、以下の情報を指定します。

- ターゲット表またはビューのスキーマ。
- ターゲットとして使用する表またはビューの名前。

デフォルト: デフォルト名は、ターゲット・サーバーのターゲット・オブジェクト・プロファイルがあれば、それからもたらされます。このプロファイルを設定していないと、デフォルトは TG にソース表またはビューの名前が続いたものになります。(たとえば、ソース表の名前が EMPLOYEE なら、ターゲット表のデフォルト名は TGEMPLOYEE になります)

- ターゲット表のタイプ

デフォルト: ユーザー・コピー

指定したターゲット表が存在しない場合、レプリケーション・センターまたは **ADDDPRSUBM** システム・コマンドを使用して作成します。

選択したソースを非 DB2 リレーショナル・ターゲットにマップするには、ターゲット表について、以下の情報を指定します。

- ターゲット表のニックネームのスキーマ
- ターゲット表のニックネーム
- リモート・スキーマ
- リモート表の名前

デフォルト: デフォルト名は、ターゲット・サーバーのターゲット・オブジェクト・プロファイルがあれば、それからもたらされます。このプロファイルを設定していな

いと、デフォルトは TG にソース表またはビューの名前が続いたものになります。(たとえば、ソース表の名前が EMPLOYEE なら、ターゲット表のデフォルト名は TGEMPLOYEE になります)

- ターゲット表のタイプ

デフォルト: ユーザー・コピー

サブスクリプション・セット・メンバーを追加するとき、ターゲット表のデフォルト・タイプであるユーザー・コピーが使用できます。あるいはレプリケーションの要求を満たす別のターゲット表タイプを選択することができます。

まだ存在しないターゲット表のサブスクリプション・セット・メンバーを追加するときは、デフォルトの設定値を使用することができます。あるいは、メンバーのプロパティを、レプリケーションの要求を満たすように変更できます。まず、使用するターゲット表のタイプを選出してから、アプライ・プログラムがそのターゲットにデータを複製する方法に合うようにプロパティを設定することができます。多様なレプリケーションのシナリオの記述と、それぞれのケースで使用するターゲット表のタイプについては、『ターゲット・タイプの選択』を参照してください。このセクションは、レプリケーションのゴールに基づく設定値選択全般を通じてのガイドとしても役立ちます。選択するターゲット・タイプにかかわらず、すべてのメンバーが共有するプロパティの共通セットを変更できます。サブスクリプション・セット・メンバー用のオプションとそのデフォルト、およびデフォルトの使用または変更が必要な場合についての説明の完全なリストは、98 ページの『すべてのターゲット表タイプに共通のプロパティ』を参照してください。

ターゲット・タイプの選択

このセクションでは、選択可能なターゲット表の各タイプについて記述し、選択するターゲット表のタイプと、レプリケーションのゴールに基づいたターゲット表のプロパティの定義方法の決定を支援します。また、既存の表をターゲットとして使用する場合に必要なことについても説明します。必要なターゲット表のタイプは、ターゲットでのデータ表示方法と、使用しているレプリケーション構成に左右されます。既存の表をターゲットとして使用できますが、新規の表も作成できます。

すべての非 DB2 のリレーショナル・ターゲット表と索引の名前は、DB2 の表と索引の命名規則に従わなければなりません。

制約事項:

- 変更後イメージ・ターゲットの列の NULL 属性は、ソース表またはビューのその列の NULL 属性と互換性がなければならない。既存の列に互換性を持たせるには、SQL の COALESCE 式を使用します。
- 非 DB2 リレーショナル・データベースのソース表の場合、ターゲット表は以下のタイプのみ定義できる。

- ユーザー・コピー表
- ポイント・イン・タイム表
- 外部 CCD 表
- RRN 列をキー列として使用する OS/400 システムのソース表の場合、ターゲット表は以下のタイプのみ定義できる。
 - ポイント・イン・タイム表
 - 外部 CCD 表
- z/OS サブシステムのソース表の場合、ユーザー・コピー表用のサブスクリプション・セットの where 文節を満たすために、アプライ・プログラムが CD 表および UOW 表を結合するならば、CD 表および UOW 表のコード化スキームは同じでなければなりません。コード化スキームについての詳細は、659 ページの『付録 A. z/OS におけるユニコードおよび ASCII のコード化スキーム』を参照してください。

ターゲット表のタイプは以下から選択できます。

ユーザー・コピー

サブスクリプション・セット・メンバーで定義された列のみが組み込まれる読み取り専用ターゲット表です。ユーザー・コピー表の構造はソース表と同じにすることができ、そうでなければ、ソース列のサブセットを組み込むことができます。変更前イメージまたは算出列が含まれる場合と含まれない場合があります。

ポイント・イン・タイム

サブスクリプション・セット・メンバーで定義された列とタイム・スタンプ列が組み込まれる読み取り専用ターゲット表です。ポイント・イン・タイム表の構造はソース表と同じにすることができ、そうでなければ、ソース列のサブセットを組み込むことができます。変更前イメージまたは算出列が含まれる場合と含まれない場合があります。

基礎集約

SQL 列関数 (SUM、AVG など) を使用して、ソース表の内容全体のサマリーを計算する読み取り専用ターゲット表です。

基礎集約表は、ソース表の内容を要約します。基礎集約表には、アプライ・プログラムが集約を実行したタイム・スタンプも組み込まれます。基礎集約表は、ソース表の状態を定期的にトラッキングする場合に使用します。

変更集約

SQL 列関数 (SUM、AVG など) を使用して、ソース表に加えられた最近の変更の内容 (CD 表または内部 CCD 表に保管されています) 全体に関するサマリーを計算する読み取り専用ターゲット表です。

変更集約表は、ソース表ではなく、CD 表または内部 CCD 表の内容を要約します。変更集約表には、変更が取り込まれた (CD 表または CCD 表に書き込まれた) 時間インターバルにマークを付けるための 2 つのタイム・スタンプも

組み込まれます。レプリケーション・サイクルの合間に行われた変更 (UPDATE、INSERT、および DELETE 操作) をトラッキングする場合は、変更集約表を使用します。

CCD (整合変更データ)

レプリケーション・コントロール情報用の列が追加された読み取り専用ターゲット表です。これらの列に組み込まれるのは、ログ・レコード番号 (またはジャーナル・レコード番号)、ソース表が SQL の INSERT、DELETE、または UPDATE ステートメントを使用して変更されたかどうかの標識、および挿入、削除、更新に関連したコミット・ステートメントのログ・レコード番号とタイム・スタンプです。オプションで、変更前イメージ列と UOW 表からの列を組み込むこともできます。

レプリカ

update-anywhere レプリケーション用の読み取り / 書き込みターゲット表です。ユーザー・アプリケーション・プログラムまたはユーザーが直接更新できるタイプのターゲット表は、レプリカ表のみです。したがって、レプリカ表は、マスター表から、そしてローカル・アプリケーション・プログラムまたはユーザーから変更を受け取ります。レプリカ表の構造はソース表と同じにすることができ、そうでなければ、ソース列のサブセットを組み込むことができます。しかし、追加のレプリケーション・コントロール列 (タイム・スタンプなど) は組み込みません。レプリカ表は DB2 データベースでのみサポートされています。

以下のセクションで、各ターゲット・タイプごとに可能な使用法について説明します。各セクションでは、使用できるターゲット表のタイプを通じてのガイドと、レプリケーションの要求を満たすターゲット表のプロパティの設定方法のガイドを行います。

- 『読み取り専用ターゲット表の定義』
- 90 ページの『行の最終的な変更をターゲット表に複製する』
- 92 ページの『multi-tier 構成内の中間層の定義』
- 94 ページの『読み取り / 書き込みターゲットの定義 (update-anywhere)』
- 97 ページの『既存の表をターゲット表として使用する』

ターゲット表のタイプを選択したら、そのターゲット表のデフォルトの設定値を使用できます。あるいは、ターゲット表のプロパティを、レプリケーションの要求を満たすように変更できます。共通のターゲット表オプションとそのデフォルト、およびデフォルトの使用または変更が必要な場合についての説明の完全なリストは、98 ページの『すべてのターゲット表タイプに共通のプロパティ』を参照してください。

読み取り専用ターゲット表の定義

ターゲット表タイプ: ユーザー・コピー、ポイント・イン・タイム、基礎集約、変更集約、CCD

ソース・データをターゲットでどのように表示するかによって、読み取り専用ターゲット表に以下のものが含まれるように定義できます。

- 『ソース表またはビューのコピー』
- 89 ページの『変更の履歴または監査情報』
- 『ソースにおけるデータまたは変更の算出サマリー』

ソース表またはビューのコピー

ターゲット表タイプ: ユーザー・コピー、ポイント・イン・タイム

ソース表のコピー: デフォルトでは、サブスクリプション・セット・メンバーを定義するときに、ユーザー・コピー表がターゲット・タイプとして作成されます。コピー時にターゲット表をソース表に突き合わせる場合は、このデフォルト・タイプを使用します。ユーザー・コピー表に追加のレプリケーション・コントロール列は含まれませんが、ソース表の行または列のサブセット、あるいはレプリケーションされない追加の列を入れることができます。

タイム・スタンプが含まれたソース表のコピー: ターゲットに変更が適用されたポイント・イン・タイムをトラッキングする場合は、ポイント・イン・タイムをターゲット・タイプとして選択します。ポイント・イン・タイム・ターゲットにはソース表と同じデータが含まれ、アプライ・プログラムがそれぞれの行をターゲットにコミットしたポイント・イン・タイムを通知するために追加されたタイム・スタンプ列も含まれます。タイム・スタンプ列は、当初は NULL です。ポイント・イン・タイム表には、ソース表の行または列のサブセット、あるいは複製されない追加の列を入れることができます。

ソースにおけるデータまたは変更の算出サマリー

ターゲット表タイプ: 基礎集約、変更集約

制約事項: 非 DB2 リレーショナル・ターゲットを、集約ターゲット表タイプにすることはできません。非 DB2 リレーショナル・ソースのタイプを、集約ターゲット表にすることはできません。

ソース表の内容全体またはソース表データに加えられた最新の変更のサマリーを含むターゲット表を作成することができます。集約ターゲット表タイプでは、COUNT、SUM、MIN、MAX、AVG などの 集約 SQL 列関数を使用して、ターゲットの列を定義できます。これらの列には、オリジナルのソース・データは含まれません。含まれるのは、定義した SQL 関数の算出値です。アプライ・プログラムはフル・リフレッシュの間、集約を作成しません。行は、アプライ・プログラムがセットを処理する経過において付加されます。集約表を使用すると、DB2 レプリケーションが 1 行 1 行ではなく要約情報だけをレプリケーションできるため、ネットワーク帯域幅とターゲット表内のスペース両方を節約できるという利点があります。

ソース表の内容の要約:

基礎集約ターゲット表を使用して、各レプリケーション・サイクル内でソース表の状態をトラッキングします。基礎集約ターゲット表では、アプライ・プログラムがソース表から集約します（読み取って計算を実行します）。基礎集約表には、アプライ・プログラムが集約を実行したタイム・スタンプも組み込まれます。

基礎集約表のみが登録済みソース表のターゲットである場合、ソース表への変更をキャプチャーする必要はありません。

例: 週ごとの平均カスタマー数を知りたいとします。ソース表にカスタマーごとの行があれば、アプライ・プログラムが週ごとにソース表の行数を合計し、結果を基礎集約表に保管します。集約を毎週実行すると、ターゲット表にはその年の週ごとのカスタマー数を示す 52 のエントリーが含まれるようになります。

CD 表または CCD 表の内容の要約:

レプリケーション・サイクルの合間にソース表で行われた変更 (UPDATE、INSERT、および DELETE 操作) をトラッキングする場合は、変更集約表を使用します。変更集約ターゲット表では、アプライ・プログラムが CD 表または内部 CCD 表から集約します（読み取って計算を実行します）。変更集約表には、キャプチャー・プログラムが CD 表または CCD 表に変更を挿入した時間インターバルにマークを付けるための 2 つのタイム・スタンプも組み込まれます。

例: 毎週新たに獲得したカスタマー数 (INSERT) と、失った既存のカスタマー数 (DELETE) を知りたいとします。CD 表の挿入された行と削除された行の数を週ごとに数え、その数を変更集約表に保管します。

重要: サブスクリプション・セット・メンバーのソース表がフル・リフレッシュのみのレプリケーション用に登録済みの場合、ソースで CD 表または CCD 表が必須である変更集約ターゲット表を持つことはできません。

変更のヒストリーまたは監査情報

ターゲット表タイプ: CCD

ソース・データの監査、あるいはデータの使用状況のヒストリーを保持することが必要な場合があります。ターゲット・タイプとして CCD 表を使用すると、CCD 表をどのように定義するかによって決まる多様な方法で、ソースの変更履歴をトラッキングできます。たとえば、データの変更が発生したときの変更前後の比較や、ソース表への更新を行ったユーザー ID をトラッキングできます。

ソース表のヒストリーを保持する読み取り専用ターゲット表を定義するには、以下の属性を持つようにターゲット CCD 表を定義します。

非コンデンス

ソースの変更すべてに関するレコードを保持するには、CCD 表を非コンデンスに定義し、発生した変更ごとに 1 行が保管されるようにします。非コンデンス

スの表には同じキー値を持つ複数の行が含まれるため、固有索引は定義しないでください。非コンデンス CCD 表は、UPDATE、INSERT、または DELETE 操作のたびに 1 つの行を保留するので、ソース表で実行された操作の履歴を保守することができます。UPDATE 操作を INSERT および DELETE 操作 (パーティション化キー列用) としてキャプチャーする場合、CCD 表は、それぞれの更新ごとに 2 つの行、すなわち DELETE に 1 行、INSERT に 1 行を指定します。

コンプリートまたは非コンプリート

CCD 表をコンプリートにするか、あるいは非コンプリートにするかを選択できます。非コンプリートな CCD 表にはソース行のコンプリート・セットが最初含まれていないため、ソース表への更新 (アプライ・プログラムが CCD 表の移植を開始してからの更新) ヒストリーを保持する非コンプリートな CCD 表を作成します。

UOW (作業単位) 列の組み込み

監査機能を向上させるには、UOW 表からの付加的な列を含めます。ユーザー指向の識別がさらに必要であれば、UOW 表で、DB2 for z/OS の関連 ID に関する列、1 次許可 ID、または OS/400 のジョブ名およびユーザー・プロファイルを使用することができます。CCD 表に組み込み可能な UOW 列についての詳細は、543 ページの『整合変更データ (CCD) 表』を参照してください。

行の最終的な変更をターゲット表に複製する

ターゲット表タイプ: 内部 CCD

ソース表で変更が頻繁に発生する場合、最後のアプライ・サイクル以降にソースで発生したコミット済みの変更を要約するために、内部 CCD 表を作成できます。キャプチャー・プログラムがログからの変更を追加するとき、CD 表は常に流動的であるため、CCD のソース変更のローカル・キャッシュが、ターゲット用のより継続的なソースとして機能します。

オリジナルのソース表が更新されるとき、キャプチャー・プログラムはソースのログでの頻繁な変更を読み取り、その変更をソースの CD 表に追加します。その CD 表から、アプライ・プログラムは CD 表での変更を読み取り、内部 CCD 表を移植します。内部 CCD 表を、最終のサイクル内で発生した CD 表のそれぞれの行ごとの最新の変更のみが含まれるように定義できます。したがって、CCD 表はアプライ・サイクルの間では静的であるため (CD 表から CCD 表に複製するアプライ・プログラムの場合)、ターゲット用のより継続的なソースになります。ソースからのコンデンス変更を行うことで、同じ行の多数の変更をターゲット表にレプリケーションしなくなるため、レプリケーションのパフォーマンス全体が改善されます。

キャプチャー・プログラムは新しい変更を常に CD 表に追加しているため、2 番目のアプライ・プログラムは CD 表の代わりに内部 CCD 表から変更を読み取ります。そのため、様々な変更を様々なターゲットに複製することはなく、ターゲット相互の同期が

とれた状態を保つことができます。2番目のアプライ・プログラムは、オリジナルのソース表をフル・リフレッシュのために使用し、内部 CCD 表を変更キャプチャー・レプリケーションのために使用します。

推奨事項:

- ソース表と内部 CCD 表との間でサブスクリプション・セット・メンバーを定義するのは、ソース表と他のターゲット表との間で他のサブスクリプション・セット・メンバーを定義する前に行う。そうするとアプライ・プログラムは、ソース表から変更を複製するために CD 表ではなく 内部 CCD 表を使用するようになります。ソース表に内部 CCD 表を定義する前に他のサブスクリプション・セット・メンバーを定義し、それらのメンバーを使用してレプリケーションを開始すると、ソース表のすべてのターゲットに対してフル・リフレッシュの実行が必要になる可能性があります。
- すべての内部 CCD 表を 1 つのサブスクリプション・セットに結合して、ソース・データベースのすべてのターゲット表相互の同期が必ずとれるようにする。
- 頻繁に変化するソース列のサブセットを他のターゲットに適用するだけの場合でも、すべての登録済みソース列が内部 CCD に複製されるというデフォルトを使用する。そうすると、オリジナルのソース表の他の登録済み列のデータを今後必要とする可能性のあるターゲット表のために、内部 CCD 表をソースとして使用できます。今後のどのようなターゲットへの変更キャプチャー・レプリケーションにおいても使用可能なのは、内部 CCD 表の列のみです。

内部 CCD 表は、レプリケーションにおける暗黙的なソースとして使用します。レプリケーション・ソースとして明示的に定義することはできません。サブスクリプション・セット・メンバーを追加する場合、オリジナルのソース表 (内部 CCD 表ではありません) をターゲット表にマップします。内部 CCD 表には以下の属性があります。

内部 CCD 表はソースの CD 表の代わりとして機能します。内部 CCD 表についての情報は、登録 (IBMSNAP_REGISTER) 表で、この内部 CCD 表のソース表と同じ列に保管されます。内部 CCD 表独自の行は、登録表内にはありません。アプライ・プログラムは、CD 表からではなく、内部 CCD 表 (存在する場合) から変更を自動的に複製します。それぞれのレプリケーション・ソースごとに、内部 CCD 表は 1 つだけ存在できます。

ローカル

CCD 表はソース表と同じデータベース内にあります。

非コンプリート

アプライ・プログラムは、フル・リフレッシュのために内部 CCD ではなくオリジナルのソース表を使用するため、後続ターゲットにすべてのソース行の初期コピーがすでに含まれるという理由から、CCD は非コンプリートです。

コンデンス

内部 CCD はコンデンスされています。すなわち、その表にはキー値ごとに 1

行が含まれ、アプライ・プログラムは変更ごとに 1 行をアプライする代わりに、CCD 表のそれぞれの行ごとに最新の変更をアプライするようになります。

UOW 列なし

内部 CCD 表では、UOW 表の列の追加はサポートされません。UOW 列を組み込んでいるターゲット CCD 表をすでに定義している場合は、内部 CCD 表を使用できません。

重要 (update-anywhere の場合): 内部 CCD 表を定義すると、アプライ・プログラムはレプリカをターゲットとしてサブスクリプション・セットを処理するときにそれを無視します。マスター・ソースの CCD 表からレプリカへの変更をアプライします。

multi-tier 構成内の中間層の定義

ターゲット表タイプ: CCD

基本のレプリケーション・モデルは、ソースが 1 つでターゲットが 1 つ以上の 2 層モデルですが、3 (またはそれより多い) 層の構成もセットアップできます。multi-tier 構成にはソース表とターゲット表が含まれ、そのターゲット表は他のターゲット表にはソースとして機能します。

multi-tier レプリケーション環境をセットアップする 1 つの理由は、第 3 層のターゲットに継続的なソースを提供することです。変更を層 1 から層 2 の CCD 表に収集できるため、各層への変更を複製する頻度をコントロールでき、ターゲット (層 3) に複製される変更の数を削減できます。しかも、ソース・システムへのデータベース接続の多くを回避できるので、接続コストを第 2 層に移動できます。

たとえば、3 層モデルで、第 1 層 (層 1) はソース・データベース、第 2 層 (層 2) は層 1 のターゲットです。しかも層 2 はターゲットを含む第 3 層 (層 3) のソースであり、変更を 1 つまたは多数の層 3 のデータベースに配布できます。レプリケーション構成内に 2 つより多い層が含まれる場合、ソースとターゲットの両方として機能する中間層は CCD 表です。

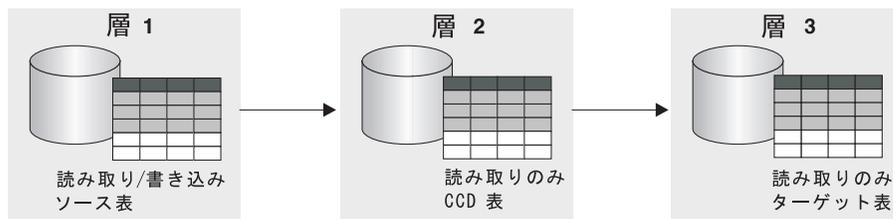


図 6. 3 層レプリケーションのモデル: データを、ソース表からターゲット表に、さらにその表から別のターゲット表に複製できます。

制約事項:

非 DB2 リレーショナル表または非 DB2 リレーショナル・データベースの CCD 表は、 multi-tier 構成の中間層としては使用できません。

プロシージャー:

multi-tier レプリケーションをセットアップするには、ターゲット表が後続のターゲットへのソースとして機能するようにします。

1. ソース表 (層 1) をレプリケーション用に登録する。表をレプリケーションのために登録する方法については、43 ページの『DB2 表をソースとして登録する』を参照してください。

このソースにおいて、キャプチャー・プログラムは層 1 で発生する変更をキャプチャーし、それを層 1 の CCD 表に保管します。

2. ソース・サーバーと (層 2 の) ターゲット・サーバーとの間でサブスクリプション・セットを作成する。サブスクリプション・セットの作成方法については、72 ページの『サブスクリプション・セットの作成』を参照してください。

このサブスクリプション・セットにおいてアプライ・プログラムは、層 1 から層 2 の CCD 表に変更を適用します。

3. ソース表 (層 1) と CCD ターゲット表 (層 2) をマッピングするサブスクリプション・セット・メンバーを定義する。サブスクリプション・セット・メンバーの定義方法については、82 ページの『サブスクリプション・セット内におけるソース表およびビューのターゲット表およびビューへのマッピング』を参照してください。

このメンバーにターゲット表を定義するとき、以下の属性を持つ CCD 表になるようなターゲット表を選択します。

外部登録済みソース

外部ターゲット表として定義しなければならず、この表が後続の層のためのソースとして機能できるように登録しなければなりません。他の登録済みソースと同様に、外部 CCD 表にも独自の行が登録 (IBMSNAP_REGISTER) 表内にあります。ソースとしても機能可能な外部 CCD 表は、ただ 1 つのソース表によってのみ移植されます。

すべての外部 CCD 表は、同じキャプチャー・スキーマを使用してサブスクリプション・セットに登録しなければなりません。

コンプリート

後続の層においてアプライ・プログラムがフル・リフレッシュと変更キャプチャー・レプリケーションの両方の実行に使用するため、コンプリートな CCD 表を使用しなければなりません。

コンデンス

コンデンスされた CCD を使用します。すなわち、この表にはキー値ごとに 1 行が含まれ、必ず最新の変更のみが後続の層に複製されます。アプライ・プログラムは変更ごとに 1 行を適用する代わりに、CCD 表のそれぞれ

れの行ごとに最新の変更を適用します。コンデンスされた表はそれぞれの行ごとにユニーク・キー値が必須であるため、固有索引を定義しなければなりません。

4. CCD 表が登録されるため、中間層のデータベースにキャプチャー・コントロール表がまだない場合は作成する。
5. 登録済み CCD 表が含まれる層 2 のサーバーと (層 3 の) 後続のターゲット・サーバーとの間で、サブスクリプション・セットを作成する。サブスクリプション・セットの作成方法については、72 ページの『サブスクリプション・セットの作成』を参照してください。

このセットにおいてアプライ・プログラムは、CCD 表から後続の層のターゲット表に変更をアプライします。アプライ・プログラムは、CCD 表をフル・リフレッシュと変更キャプチャー・レプリケーションの両方に使用します。アプライ修飾子は通常、CCD にデータを入れるのに使った修飾子とは別のものを使用しますが、同一のものを使用することも可能です。

6. CCD ソース表 (層 2) と後続のターゲット表 (層 3) をマッピングするサブスクリプション・セット・メンバーを定義する。サブスクリプション・セット・メンバーの定義方法については、82 ページの『サブスクリプション・セット内におけるソース表およびビューのターゲット表およびビューへのマッピング』を参照してください。

この CCD ソース表をサブスクライブするターゲット表を組み込んで、複数のメンバーをセットアップできます。これが multi-tier 構成の最終層であれば、ターゲット表のタイプは何でも可能です。ただし、計画している層が 3 つより多い場合は、層 3 のターゲット表をステップ 3 で指定したように定義し、ステップ 4 から 5 を繰り返して後続の層を追加します。

重要: フル・リフレッシュが外部 CCD (中間層) で発生した場合、ソースとしてその外部 CCD を使用する、すべての後続の層におけるアプライ・プログラムはフル・リフレッシュを実行します。これをカスケード・フル・リフレッシュと呼びます。

読み取り / 書き込みターゲットの定義 (update-anywhere)

ターゲット表タイプ: レプリカ

update-anywhere レプリケーションでは、マスター・ソース表における変更はレプリカ・タイプの従属ターゲット表にレプリケーションされ、さらにそのレプリカ表における変更はマスター・ソース表に逆にレプリケーションされることが可能です。

update-anywhere レプリケーションで、マスター表とそのレプリカのすべては、ソースとターゲットの両方として機能する読み取り / 書き込み表です。

前提条件:

update-anywhere レプリケーションに必要な条件は以下のとおりです。

- 単一アプリケーション・プログラムはマスター表とレプリカ表の両方を更新しないため、参照保全制約を宣言して使用しなければならない。参照保全違反はアプリケーションのロジックでは検出できません。
- 参照保全違反を防止するため、複数のマスター表を対象に存在するすべての参照制約をレプリカ表に含めなければならない。いくつかの参照制約を省略すると、レプリカ表に加えられた更新がマスター表に複製されるとき、参照保全違反を引き起こす可能性があります。管理ツールは、ソース表からターゲット表への参照制約定義のコピーを行わず、新しい制約を生成することもできません。
- フル・リフレッシュ中の参照保全チェックをバイパスするには、ASNLOAD 出口ルーチンを使用しなければならない。

制約事項:

- update-anywhere レプリケーションで、CCD 表をソースまたはターゲットとして使用することはできない。
- LOB データ・タイプの列は、update-anywhere レプリケーションの対象にならない。
- DATALINK データ・タイプの列は、ソース表を競合検出しないように登録する場合を除いて、update-anywhere レプリケーションの対象にならない。
- 非 DB2 データベースのタイプをレプリカ・ターゲット表にすることはできない。そのため、update-anywhere レプリケーションの対象にできません。

プロシージャ:

マスター表と 1 つ以上のレプリカ表との間で (それぞれのレプリカ表が別々のデータベースにある場合) update-anywhere 構成をセットアップするには、以下のようにします。

1. キャプチャー・プログラムはそれぞれのレプリカ表ごとに変更をキャプチャーするため、レプリカ表を入れるデータベースそれぞれにキャプチャー・コントロール表がまだない場合は作成する。
2. ソース表 (マスター表) をレプリケーションのために登録する。表をレプリケーションのために登録する方法については、43 ページの『DB2 表をソースとして登録する』を参照してください。

このソースにおいて、キャプチャー・プログラムはマスター表での変更をキャプチャーし、それをマスターの CD 表に保管します。

3. 1 つ以上のレプリカ表を入れるマスター・データベースとターゲット・データベースとの間に、サブスクリプション・セットを作成する。サブスクリプション・セットの作成方法については、72 ページの『サブスクリプション・セットの作成』を参照してください。

すべてのレプリカ表が同じデータベースに、そしてすべてのマスター表が別のデータベースに含まれている場合、サブスクリプション・セットは 1 つだけ必要です。レプリカ表が複数のデータベースに含まれている場合、保有するレプリカ・データベースと同数のサブスクリプション・セットが必要です。

4. マスター表とそのおのおのに関連したレプリカ表との間のマッピングごとに、サブスクリプション・セット・メンバーを定義する。サブスクリプション・セット・メンバ

一の定義方法については、82 ページの『サブスクリプション・セット内におけるソース表およびビューのターゲット表およびビューへのマッピング』を参照してください。

この構成では、レプリカ表が含まれたサーバーで通常実行されるアプライ・プログラムは 1 つしかありません。このセットにおけるアプライ・プログラムは、マスターの CD 表から変更をプルして、それをレプリカ表に適用します。また、アプライ・プログラムはレプリカ表の CD 表から変更をプッシュして、それをマスター表に適用します。

重要: update-anywhere 構成において、マスター表とレプリカ表はデータを往復させて互いに複製するため、レプリカ・ターゲット表にはソース表と同じ列が含まれている必要があります。脱落した列がマスター・サイトで NULL 可能、または NOT NULL WITH DEFAULT として定義されている場合に限り、マスター表の列のサブセットが含まれるレプリカ・ターゲットを作成できますが、レプリカで新しい列の追加または列の名前変更は行わないでください。

5. レプリカ表にソースのプロパティを定義する。

レプリカ表を組み込んでサブスクリプション・セット・メンバーを作成するとき、DB2 レプリケーションはそのレプリカ表をレプリケーション・ソースとして自動的に登録します。レプリカ・ターゲット表はソースとして機能するため、ターゲット表に共通するプロパティとは別に設定可能なプロパティがあります。これらのプロパティにより、キャプチャー・プログラムがレプリカへの変更をハンドルする方法が決定されます。ただし、マスター表から継承されても、レプリカ表に合うように変更できないプロパティが 2 つあります。競合検出レベルとフル・リフレッシュの可否のプロパティです。このソースにおいて、キャプチャー・プログラムはレプリカ表での変更をキャプチャーし、それをレプリカの CD 表に保管します。登録オプションとそれらのデフォルトの完全なリスト、およびこれらのデフォルトを使用した方がよい場合と変更した方がよい場合についての説明は、47 ページの『ソース表用の登録オプション』を参照してください。

重要: マスターとレプリカがソースとターゲットの両方として機能するとしても、フル・リフレッシュ・コピーは、マスターからレプリカへのみ行われ、レプリカからマスターへは行われません。

競合を防止するため、レプリカ表のターゲット・キーを、マスター・ソース表の主キーまたは固有索引と同じにしなければなりません。マスター表はレプリカを更新でき、レプリカはマスターを更新できるため、アプライ・サイクルの合間にマスター表のある行が更新され、1 つ以上のレプリカ表の同じ行にそれとは異なる更新が加えられた場合 (変更がマスター CD 表とレプリカ CD 表で行われます)、競合が発生する可能性があります。レプリカ表は競合検出レベルをマスター・ソース表またはビューから継承します。最善なのは、マスターからすべてのレプリカ表へデータを複製するときに、競合が決して発生しないようにアプリケーションを設計することです。マスター・ソースを登録したとき、競合検出のレベルを 3 つの選択肢から選びました。競合検出レベルの選択と、(標準または拡張競合検出を選択した場合に) 発生す

る競合への対応方法についての詳細は、60 ページの『競合検出の設定 (update-anywhere レプリケーション)』を参照してください。

ソース表における参照保全制約を定義した場合、保全違反を防止するために、同じ参照保全制約をレプリカ表にも定義しなければなりません。参照保全違反が発生した場合、サブスクリプション・サイクルは自動的に再試行されます。

既存の表をターゲット表として使用する

事前定義した DB2 表を、サブスクリプション・セットでのターゲット表として使用できます。すなわち、DB2 レプリケーションの外部で定義されているターゲット表を組み込むサブスクリプション・セット・メンバーを定義することができます。このようなユーザー定義のターゲット表は、表の構造が有効である限り、レプリケーションで有効なターゲット表タイプ (ユーザー・コピー、ポイント・イン・タイム、基礎または変更集約、CCD、あるいはレプリカ) のいずれかになります。たとえば、ユーザー定義のポイント・イン・タイム表には、IBMSNAP_LOGMARKER という TIMESTAMP タイプの列が組み込まれなければなりません。

要件:

- サブスクリプション・セット・メンバーの定義に含まれる列が既存のターゲット表より少ない場合、レプリケーションに関連するターゲット表の列は NULL が許可されているか、あるいは NOT NULL WITH DEFAULT として定義されていなければならない。
- ポイント・イン・タイム CCD 表、ユーザー・コピー CCD 表、レプリカ CCD 表、およびコンデンス CCD 表には固有索引が必要である。既存のターゲット表を使用してサブスクリプション・セット・メンバーを定義するときは、既存の固有索引を使用するか、または新規の固有索引を指定することができます。

制約事項:

- サブスクリプション・セット・メンバーの定義に、既存のターゲット表より多くの列を含めることはできない。
- レプリケーション・センターを使用する場合、該当する列がターゲット表にまだ存在していなければ、サブスクリプション・セット・メンバーにその列を追加できない。

DB2 レプリケーションは、既存のターゲット表とサブスクリプション・セット・メンバー定義との間の矛盾をチェックします。

重要 (multi-tier の場合): ソース表が層 1、CCD 表が層 2、そして既存の表が層 3 であるような multi-tier 構成をセットアップする場合、層 1 と層 2 との間でサブスクリプション・セット・メンバーを定義するときに、CCD 表を既存のターゲット表に指定されている属性に一致するように定義します。そして、この CCD 表をソース表とする既存のターゲット表のサブスクリプション・セット・メンバーを定義します。

すべてのターゲット表タイプに共通のプロパティ

このセクションでは、ターゲット表を作成するときにタイプに関係なく設定できる共通のプロパティについて説明します。必要なレプリケーションのタイプを基にして、使用するターゲット表またはビューに合うようにプロパティを変更することができます。以下のセクションでは、ソース・データをターゲット表にマップする方法に応じて定義できる共通な特性について説明します。

- 『ターゲットに適用するソース列』
- 『ターゲットに適用するソース行』
- 100 ページの『ソース列からターゲット列にマップする方法』
- 100 ページの『ターゲット・キー』
- 102 ページの『アプライ・プログラムがターゲット・キー変更オプションを使用してターゲット・キー列を更新する方法』

ターゲットに適用するソース列

デフォルト: すべての登録済みソース列はターゲットに複製されます。

レプリケーションのシナリオによっては、すべての列をターゲット表にレプリケーションしたいわけではない場合があります。また、ソース表に定義したすべてのデータ・タイプでターゲット表がサポートできないものがある場合もあります。ソース表より列数が少ない列 (垂直) のサブセットを定義することができます。

デフォルトでは、ターゲット表にはソース表の **LOB** 列と **DATALINK** 列を除くすべての登録済みの列が含まれます。ソース表に存在する列で、ターゲット表に入れたくないものがある場合は、ターゲット表に複製したいソース列のみを選択します。選択しないソース表の登録済みの列は、他のサブスクリプション・セット・メンバーに引き続き使用できますが、現行のソースからターゲットへのマッピングには組み込まれません。

算出列をターゲット表に追加することもできます。これらの列は、**SUBSTR** のような SQL スカラー関数で定義するか、あるいは列 **A** の値を列 **B** の値で除算する (**colA/colB**) ような派生列にすることができます。これらの算出列は、ソース表のすべての列を参照します。

ターゲットに適用するソース行

デフォルト: すべてのソース行はターゲットに複製されます。

デフォルトでは、ターゲット表にはソース表のすべての行が含まれます。レプリケーションのシナリオによっては、すべての行をソース表からターゲット表にレプリケーションしたいわけではなかったり、あるいは異なる種類のデータを含むソース行を異なるターゲット表にレプリケーションしたいという場合があります。一定の条件 (SQL **WHERE** 文節) と一致する行が含まれる行 (水平方向) サブセットを定義できます。

SQL の述部には、通常 ID または区切り ID を入れることができます。WHERE 文節についての詳細は、「DB2 SQL リファレンス」を参照してください。

例:

- ターゲット表が会社の運営上の 1 部門のオペレーショナル・データ・ストアであると想定する。WHERE 文節をサブスクリプション・セット・メンバーで定義すると、その部門 (またはその部門内のすべての部) のすべての行を、ソース表からターゲット表に複製できます。
- 同じデータベース内に幾つかのターゲット表があると想定する。WHERE 文節を 1 つのサブスクリプション・セット・メンバーで定義すると、すべての LOB 列 (主キー列も加えて) を 1 つのターゲット表に複製でき、WHERE 文節を別のサブスクリプション・セット・メンバーで定義すると、他の列すべてを別々のターゲット表に複製できます。したがって、ターゲット・データベースはソース表からのデータすべてを保有できますが、ターゲット・データベースのソース表を非正規化して、データウェアハウスへの照会パフォーマンスを調整します。

行の述部の制約:

- この文節に WHERE を入力しない。これは暗黙で指定されます。副選択ステートメントの文節にのみ WHERE を入力してください。
- 文節をセミコロン (;) で終わらせない。
- 変更前イメージ列、算出列、または IBMSNAP 列を使用して、データをサブセット化またはフィルター操作する場合は、119 ページの『サブスクリプションの際のデータのサブセット化』を参照してください。
- WHERE 文節にブール式 OR を含める場合、たとえば、(COL1=X OR COL2=Y) のように、述部を括弧で囲む。
- ターゲット表が変更集約表であって、変更前イメージ列を含んでいる場合、変更前イメージ列を GROUP BY 文節に含めなければならない。

次に示す例には、ターゲット表の行をフィルターに掛けるために使用できる WHERE 文節が含まれます。これらの例は一般的であり、モデルとして使用するためのものです。

- 特定の値を持つ行を指定する WHERE 文節
特定の値 (たとえば、管理職にある社員を表す MGR) をもつ行だけをコピーするには、次のような WHERE 文節を使用します。

```
EMPLOYEE = 'MGR'
```
- ある範囲の値を持つ行を指定する WHERE 文節
ある範囲の値 (たとえば従業員番号 5000~7000) の行をターゲット表にコピーするには、次のような WHERE 文節を使用します。

```
EMPID BETWEEN 5000 AND 7000
```

ソース列からターゲット列にマップする方法

デフォルト: ソース列名は同じターゲット列名にマップします (ターゲット表がまだ存在しない場合)。

デフォルトでは、ターゲット表 (存在していない場合) の列名はソース表の列名と一致し、ソース列のデータ値は同じ名前のターゲット列に複製されます。レプリケーション・コントロール列 (IBMSNAP または IBMQSQ で始まる) を除くターゲット表のすべての列の名前を変更できます。ターゲット表が存在していれば、それぞれの列を明示的にマップしなければなりません。

DB2 表を、既存のニックネームを持つ非 DB2 リレーショナル表にマップする場合、一部の列のデータ・タイプに互換性がないことがあります。ソース列のデータ・タイプにターゲット列のデータ・タイプとの互換性がない場合、以下のようにターゲットでデータ・タイプを変更して、ソースとの互換性を持たせることができます。

- ソースのデータ・タイプがターゲットで必要とするデータ・タイプに一致するように調整するための算出列を追加できる。
- 非 DB2 リレーショナル・ターゲット表のニックネームを変更して、データ・タイプ変換を変更できる。

例: データ・タイプが DATE の DB2 列を含む DB2 ソース表から、データ・タイプが DATE の Oracle 列を含む Oracle ターゲット表にデータを複製するとします。

表 4. DB2 DATE 列の Oracle DATE 列へのマッピング

| DB2 列 | ニックネーム・データのマッピング | Oracle 列 |
|-------------|---------------------------------|-------------|
| A_DATE DATE | A_DATE TIMESTAMP A_DATE DATE | A_DATE DATE |

Oracle データ・タイプ DATE を含む Oracle ターゲット表が作成されます (日付データとタイム・スタンプ・データの両方が入ります)。連合データベースにある Oracle の DATE データ・タイプの初期のニックネームは、DB2 データ・タイプを TIMESTAMP としてマップします。DB2 レプリケーション・センターとレプリケーション用の OS/400 システム・コマンドは、ニックネームのデータ・タイプを DATE に変更するので、DATE は Oracle にレプリケーションされますが、TIMESTAMP はされません。

ターゲット・キー

デフォルト索引名: デフォルト名は、ターゲット・サーバーのターゲット・オブジェクト・プロファイルがあれば、それからもたらされます。このプロファイルを設定していないと、デフォルトは「IX + ターゲット表の名前」となります。たとえば、ターゲット表の名前が TGMEMPLOYEE なら、ターゲット表の索引のデフォルト名は IXTGMEMPLOYEE になります。

コンデンス・ターゲット表が変更キャプチャー・レプリケーションに関係している場合、アプライ・プログラムは、そのターゲット表にターゲット・キー と呼ばれる主キー

または固有索引があることを必要とします。ターゲット表の固有索引として使用する列は選択できます。以下のタイプのターゲット表はコンデンスされており、ターゲット・キーが必須です。

- ユーザー・コピー
- ポイント・イン・タイム
- レプリカ
- コンデンス CCD

ターゲット表を新規に作成する場合は、デフォルトの索引名とスキーマを使用できますが、デフォルトをユーザーの命名規則に一致するように変更することもできます。

新しいターゲット表の固有索引を作成するためのオプションが 2 つあります。

- ターゲット表の固有索引として使用する列を指定する。
- DB2 レプリケーションでユーザー向けの固有索引を選択する。

固有索引用の列を選択しないと、DB2 レプリケーションではソース表に以下の定義のいずれかがあるかどうかをこの順序でチェックします。

1. 主キー
2. ユニーク制約
3. 固有索引

DB2 レプリケーションで、ソース表のこれらの定義のいずれかが検出され、関連した列が登録されていてターゲット表の一部であることが検出された場合、DB2 レプリケーションはソース表の主キー（または固有索引か RRN）をターゲット・キーとして使用します。

主キーまたは固有索引を持たない OS/400 ソース表の場合は、相対レコード番号 (RRN) を一意性の因子として使用するよう表の登録を変更します。サブスクリプション・セット・メンバーを定義するときに、RRN 列をターゲット表の固有索引として指定します。OS/400 ソース表における RRN の定義についての詳細は、63 ページの『主キーの代わりに相対レコード番号 (RRN) を使用する (OS/400)』を参照してください。

ターゲット・キーとして RRN を使用する OS/400 システムのターゲット表の場合、これらのターゲット表を複製するには、アプライ・プログラムを OS/400 システムで実行します。

既存のターゲット表の場合、固有索引を選択する必要があります。以下のオプションのいずれかを選択することができます。

- ターゲット表の既存の索引を使用する。

既存の索引を使用するには、レプリケーション・センターで、索引を表す列を選択します。レプリケーション・センターが厳密に一致するものを検出すれば、アプライ・

プログラムが使用するようにターゲット・キーを設定するだけでですが、検出しなかった場合は、固有索引を作成してから、ターゲット・キーをアプライ・プログラムが使用するように設定します。

- ターゲット表の別の索引を作成する。

固有索引がまだない場合は作成され、アプライ・プログラムが使用するようにターゲット・キーが設定されます。

重要: ソース表で更新される可能性のある列が組み込まれたターゲット表のキーを選択する場合、ターゲット・キー列に特殊な更新を加えるようアプライ・プログラムに指示しなければなりません。詳細については、『アプライ・プログラムがターゲット・キー変更オプションを使用してターゲット・キー列を更新する方法』を参照してください。

アプライ・プログラムがターゲット・キー変更オプションを使用してターゲット・キー列を更新する方法

制約事項: 更新を削除および挿入のペアとしてキャプチャーするために登録されたソース表において、ターゲット・キー変更オプションを使用することはできません。

サブスクリプション・セット・メンバーを定義するときにターゲット・キー変更オプションを選択すると、ターゲット・キーの変更時、アプライ・プログラムはターゲット・キー列に特殊な更新を加えます。アプライ・プログラムでこれらの特殊な更新を行うためには、ターゲット表のターゲット・キー列の一部になっているソース表の列を、CD (または CCD) 表の変更前イメージ列に登録しなければなりません。ターゲット・キーを構成する列の変更前イメージ値をキャプチャーするためのソース登録を定義していない場合は、異なるキーを持つターゲット表をサブスクライブする前に、登録を変更して変更前イメージ値を組み込まなければなりません。

ターゲット・キー列の変更前イメージ値が CD (または CCD) 表にあることを確認後、ターゲット・キー列の更新時にアプライ・プログラムが変更前イメージ値を使用するように、サブスクリプション・セット・メンバーのオプションを選択します。

ターゲット・キー列の更新時にアプライ・プログラムが変更前イメージ値を使用するように指定しないと、ターゲット・キーの一部となっているソース表の列を更新しても、DB2 レプリケーションはデータを正しくレプリケーションしません。アプライ・プログラムは、新しい値を使用してターゲット表の行の更新を試みますが、更新のための新しいキー値がターゲット表で検出されません。そこでアプライ・プログラムはこの更新を INSERT に変換して、新しいキー値をターゲット表に挿入します。この場合、古いキー値をもつ古い行はターゲット表内に残ります。これは不要な行です。ターゲット・キー列が変更前イメージ値を使用して処理されるような変更を指定すると、アプライ・プログラムは古いキー値を検出でき、それを削除して新しい値を挿入します。

関連概念:

- 259 ページの 『第 14 章 DB2 レプリケーション・センターの使用』

関連作業:

- 43 ページの 『第 3 章 表およびビューをレプリケーション・ソースとして登録する』
- 117 ページの 『第 6 章 レプリケーション環境におけるデータのサブセット化』
- 659 ページの 『付録 A. z/OS におけるユニコードおよび ASCII のコード化スキーム』

関連資料:

- 390 ページの 『ADDDPRSUBM: DPR サブスクリプション・セットのメンバーの追加 (OS/400)』
- 374 ページの 『ADDDPRSUB: DPR サブスクリプション・セットの追加 (OS/400)』
- 543 ページの 『整合変更データ (CCD) 表』

第 5 章 特殊なデータ・タイプのレプリケーション

LOB、DATALINK、ROWID、または DB2 以外のデータ・タイプなど、特殊なデータ・タイプを複製するときには、一定の条件および制約事項がありますので注意してください。場合によっては、DB2 レプリケーションでこれらのデータ・タイプを処理できるように追加のセットアップ・ステップを実行する必要があります。この章では、これらの条件および制約事項について説明します。この章は次のセクションから構成されています。

- 『レプリケーションにおける一般的なデータの制約事項』
- 106 ページの『ラージ・オブジェクトのレプリケーション』
- 107 ページの『DATALINK 値のレプリケーション』

レプリケーションにおける一般的なデータの制約事項

DB2 レプリケーションでは現在のところ、特定のデータ・タイプに関して制約事項があります。

- 一般的なデータ圧縮の制約事項

DB2 レプリケーションでは、EDITPROC または FIELDPROC 文節を使用して変更されたデータをレプリケーションすることはできません。

- データ暗号化の制約事項

DB2 レプリケーションでは、暗号化されたデータをレプリケーションすることはできません。

- データ・タイプの制約事項

DB2 レプリケーションでは、いかなる状況でも次のデータ・タイプをレプリケーションすることはできません。

- DB2 以外のリレーショナル・ソースからの LOB 列
- VALIDPROC が定義されているすべての列

DB2 レプリケーションは、ソース表およびターゲット表が DB2 for z/OS にある場合は、長い可変グラフィック (LONG VARGRAPHIC) データをレプリケーションできません。

DB2 レプリケーションでは、要約データ・タイプを含む表をレプリケーションすることはできません。

DB2 では、地理情報データ・タイプ列を含む表をレプリケーションすることはできませんが、実際の地理情報データ・タイプ列をレプリケーションすることはできません。

ユーザー定義のデータ・タイプ (DB2 Universal Database の固有のデータ・タイプ) は、レプリケーションされる前に変更データ (CD) 表で基本データ・タイプに変換されます。

ラージ・オブジェクトのレプリケーション

DB2 Universal Database はラージ・オブジェクト (LOB) のデータ・タイプをサポートします。このデータ・タイプには、バイナリー LOB (BLOB)、文字 LOB (CLOB)、2 バイト文字 LOB (DBCLOB) が含まれます。ここでは、ここに示したタイプの LOB データをすべて取り上げます。

キャプチャー・プログラムはログ・レコード内の LOB 記述子を読み、変更されているために複製する必要があるデータが LOB 列にあるかどうかを判別しますが、変更データ (CD) 表には LOB データをコピーしません。LOB 列が変更されると、キャプチャー・プログラムは CD 表に標識を設定します。アプライ・プログラムはこの標識を読み取ると、次に LOB 列全体 (LOB 列の変更部分だけではない) をソース表からターゲット表に直接コピーします。

LOB 列には最大 2 GB のデータを入れることができるので、アプライ・プログラムには十分なネットワーク帯域幅を指定できるようにしてください。同じように、ターゲット表にも LOB データが入るだけのディスク・スペースを指定しなければなりません。

制約事項:

- アプライ・プログラムは、常に最新バージョンの LOB 列をソース表 (CD 表ではない) から直接コピーします。これはその列が CD 表の他の列よりも新しい場合にも当てはまります。このため、ターゲット行の LOB 列が変更された場合この LOB 列は、そのターゲット行の残りのデータと矛盾するものになる可能性があります。ターゲット行でデータの矛盾が発生する可能性を最小化するために、アプライ・サイクルのインターバルは、アプリケーションで実用上の問題が起きない範囲で、できるだけ短くしてください。
- 1 つの表で複製できる LOB 列は 10 以下です。10 よりも多い LOB 列を含む表を登録すると、アプライ・プログラムからエラー・メッセージが戻されます。1 つの表に 10 よりも多くの LOB 列を登録しようとする、レプリケーション・センターからエラー・メッセージが戻されます。
- LOB データは読み取り専用表にのみコピーできます。したがって、LOB データをレプリカ表に複製することはできません。
- DB2 for OS/390 バージョン 6 (またはそれ以降) と DB2 Universal Database (その他のオペレーティング・システム用) の間で LOB データをコピーするには、DB2 Connect 7 以降が必要です。
- ニックネームを使って LOB データを参照することはできません。
- LOB、DATALINK、または ROWID 列の変更前イメージ値はサポートされていません。

- エクステンダーの LOB 列データに関連付けられた追加のコントロール・ファイルがデータベースの外部で保守されている、テキスト、音声、ビデオ、イメージ、その他のエクステンダーの場合、DB2 Extenders™ におけるレプリケーションはサポートしていません。
- DB2 は、LOB 全体のレプリケーションしかできません。LOB の部分的なレプリケーションはできません。
- OS/400 上のレプリケーション環境でリモート・ジャーナル・セットアップを使用する場合は、LOB 列はレプリケーションできません。

DATALINK 値のレプリケーション

リモート・ネットワークを介して大きなファイル (マルチメディア・データなど) にアクセスするのは、非効率的で費用がかかります。外部ファイル・システムに保管されるデータを表す DATALINK データ・タイプを使用すると、構造化されていないファイルに、より迅速にアクセスし、ファイルを複製できます。

DB2 Universal Database は、アクセス・コントロール、参照保全、および構造化されていない大きなファイルのリカバーをデータベースで管理できるようにする、DATALINK データ・タイプをサポートしています。DB2 Universal Database は、次のオペレーティング・システムで DATALINK の値をサポートします。

- AIX
- Solaris™ 実行環境版
- Windows NT
- Windows 2000
- OS/400

DATALINK 列の値には、外部ファイルのロケーションを示す URL が入っています。DB2 レプリケーションは、DATALINK 列の値、およびここから参照されるファイルをレプリケーションするときに、次のコンポーネントを使用します。

ASNDLCOPY 出口ルーチン

ソース・ファイル・システム上の URL をターゲット・ファイル・システム上の URL にマップした後、適切なファイル・コピー・デーモンに接続して、参照されたファイルを複製します。

Data Links Manager レプリケーション・デーモン (DLFM_ASNCOPYD)

ASNDLCOPY 出口ルーチンと共同で、DATALINK 列の値から参照されるファイルをコピーします。DLFM_ASNCOPYD デーモンは DB2 Data Links Manager バージョン 8 の一部です。このデーモンは、AIX、Solaris™ 実行環境版、および Windows オペレーティング・システムで使用できます。

ASNDLCOPYD デーモン

ASNDLCOPY 出口ルーチンと共同で、DATALINK 列の値から参照されるファ

イルをコピーします。このデーモンは DB2 for iSeries で提供されます。
ASNLDLCPYD は OS/400 で、またオプションとして、他のオペレーティング・システムで使用してください。

アプライ・プログラムはデータ・タイプが DATALINK のデータを読み取ると、参照データを予備ファイルに入れるとともに、更新されたファイルの URL を入力ファイルに入れます。

アプライ・プログラムは続いて ASNLDLCPY 出口ルーチン呼び出しします。この ASNLDLCPY 出口ルーチンは、ソース・ファイル・システム上に物理ファイルが存在することを確認し、ターゲット・ファイル・システム上の対応するファイルに URL をマップし、このターゲット・ファイルのロケーションを結果ファイルに保管し、適切なファイル・コピー・デーモン (DLFM_ASNLCPYD、ASNLDLCPYD、または FTP) に接続し、ソース・ファイル・システムからターゲット・ファイル・システムへ外部ファイルをコピーします。

推奨: アプライ・プログラムは ASNLDLCPY ルーチンの処理が完了するのを待ってからサブスクリプション・セットのレプリケーションを完了させるため、DATALINK 列に対しては、別個のサブスクリプション・セットを使用してください。外部ファイルのコピー時にエラーが生じると、サブスクリプション・セット全体のレプリケーションが失敗します。サブスクリプション・セットが失敗した場合、アプライ・プログラムはサブスクリプション・セットを非活動化せずに、次のアプライ・サイクルでサブスクリプション・セットを再度処理します。

UNIX および Windows の場合: loadxit パラメーターを y に設定して ASNLOAD 出口ルーチン呼び出しして、アプライ・プログラムを始動してください。ASNLOAD 出口ルーチンはフル・リフレッシュ時に外部ファイル (DATALINK の値が指し示す) をコピーします。詳細については、169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』を参照してください。

OS/400 の場合: アプライ・プログラムがフル・リフレッシュ時に外部ファイルをコピーできるように、ASNLDLCP 出口ルーチン呼び出すように ASNLOAD 出口ルーチンを変更してください。詳細については、169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』を参照してください。

重要: 外部ファイルは非常に大きい場合があるため、アプライ・プログラムとファイル転送メカニズムの両方について、これらのファイルをコピーするために使用する十分なネットワーク帯域幅を確保してください。同じように、ターゲット・システムにもこれらのファイルが入るだけのディスク・スペースを確保しなければなりません。

制約事項:

- OS/400 上の DB2 データベースと他のオペレーティング・システム上の DB2 データベースの間で DATALINK 列を複製することはできません。

- OS/400 オペレーティング・システムの場合、DATALINK の値の "comment" 属性に対するサポートはありません。
- DATALINK 列で update-anywhere レプリケーションを使用する場合は、競合検出レベルに **None** を指定して、DATALINK 列、および同じサブスクリプション・セット内のその他の列の両方について、競合検出をオフにする必要があります。DB2 レプリケーションは、DATALINK 列が参照する外部ファイルの更新競合はチェックしません。
- DATALINK 列における変更前イメージ値はサポートしていません。
- ターゲット表が基礎集約表または変更集約表である場合は、DATALINK 列がサポートできません。
- 整合変更データ (CCD) 表の中のデータを複製するときには、次の制約事項が適用されます。
 - 内部 CCD 表は、DATALINK 標識 (関連付けられた URL に関する情報を含む VARCHAR タイプの文字ストリング) を含むことはできますが、DATALINK の値を含むことはできません。アプライ・プログラムは、これらの表タイプでデータを複製するときには、ASNDLCOPY 出口ルーチン呼び出しません。
 - コンデンスされた外部 CCD 表は DATALINK 列を含むことができます。
 - 非コンデンス CCD ターゲット表は DATALINK 列を含むことができません。

次のセクションでは、DATALINK の値および参照されるファイルの両方をターゲット・システムに複製するためにアプライ・プログラムから使用される、ユーザー出口ルーチンおよびファイル・コピー・デーモン (オペレーティング・システムごとに異なる) について説明します。

- 『ASNDLCOPY 出口ルーチンの設定と使用方法』
- 111 ページの『DLFM_ASNCOPYD の設定と使用方法 (UNIX、Windows)』
- 113 ページの『ASNDLCOPYD の設定と使用方法 (OS/400)』

ASNDLCOPY 出口ルーチンの設定と使用方法

サブスクリプション・セットのレプリケーションの準備ができると、アプライ・プログラムは変更データ (CD) 表の中の対象の行を識別します。DATALINK 列の値が検出された場合、アプライ・プログラムは更新されたファイルの URL を入力ファイルに入れます。アプライ・プログラムは続いて ASNDLCOPY 出口ルーチン呼び出します。出口ルーチンはこの入力ファイルを読み取り、DATALINK ソース・ファイルの各ロケーションを対応するターゲット・ファイル・ロケーションにマップします。次に、ASNDLCOPY 出口ルーチンはファイル・コピー・デーモンに接続し、参照されたファイルを、ソース・ファイル・システムから新しくマップされたターゲット・ファイル・システム・ロケーションにマップします。

ASNDLCOPY ルーチンが完了した時点で、このルーチンはアプライ・プログラムに戻りコードを渡します。ゼロ以外の戻りコードは、1 つまたは複数のファイルでレプリケー

ションが失敗したことをアプライ・プログラムに知らせます。このような場合、アプライ・プログラムはメッセージを出し、そのサブスクリプションをスキップして次のサブスクリプションを処理します。ゼロの戻りコードは、レプリケーションが正常に行われたことをアプライ・プログラムに知らせます。

ASNDLCOPY 出口ルーチンのソース・コードを使用して、システムの要件に合わせてサンプル・プログラム (ASNDLCOPY.smp と呼ばれ、`¥sqllib¥samples¥repl` ディレクトリに置かれている) を変更できます。サンプル・プログラムには、次の構成ファイルが含まれます。

ASNDLSRVMAP

ソース URL をターゲット URL にマップします。

例: `http://source.com/file` to `http://target.com/file`

ASNDLUSER

ソースおよびターゲットのファイル・システムに接続するときに使用される、ログオンおよびアドレス・ロケーション情報が入っています。

ASNDLPARM

ASNDLCOPY 出口ルーチンの関数をコントロールするために使用される稼働パラメーターが入っています。これらのパラメーターには、ソース・ファイルを異なるターゲット・ファイル・ロケーションに複製するために使用される

REPLACE_FILE パラメーター、そして複製するファイルの最後の変更時刻を保存するために使用される **PRESERVE_MODTIME** パラメーターが含まれます。ASNDLPARM は、UNIX および Windows オペレーティング・システムでのみ使用される、オプションの構成ファイルです。

外部ファイルを複製するために、ユーザー自身の出口ルーチンを構成することができませんが、プログラムの名前は ASNDLCOPY にする必要があります。構成ファイルは、アプライ・プログラムの現行実行パスに入れてください。

構成ファイルのセットアップ方法、およびこの出口ルーチンの変更方法については、`¥sqllib¥samples¥repl` ディレクトリにあるサンプル・プログラムの PROLOG セクションを参照してください。

プロシージャー:

ASNDLCOPY 出口ルーチンを使用するには:

1. サイトの要件に合わせて、必要であれば ASNDLCOPY ルーチンを変更します。

アプライ・プログラムのトレース・オプションをオンにすると、ASNDLCOPY ルーチンはログ・ファイルとトレース・ファイルという 2 つのファイルを作成します。このログ・ファイルの名前は以下のとおりです。

`ASNDLApplyQualSetNameSrcSrvrTgtSrvr.LOG`

ApplyQual はアプライ修飾子、*SetName* はサブスクリプション・セットの名前、*SrcSrvr* はソース・サーバーの名前、そして *TgtSrvr* はターゲット・サーバーの名前です。ログ・ファイルには、ASNDLCOPY ルーチンが生成するすべてのメッセージが入れられます。トレース・ファイルの名前は以下のとおりです。

ASNDLApplyQualSetNameSrcSrvrTgtSrvr.TRC

トレース・ファイルには、ASNDLCOPY ルーチンが生成するトレース情報がすべて入れられます。

- 必要に応じて ASNDLUSER、ASNDLSRVMAP、および ASNDLPARM 構成ファイルを構成します。

UNIX および Windows の場合: ASNDLPARM ファイルの中で **REPLACE_FILE** パラメーターを YES (デフォルト) に設定した場合、ターゲット・ディレクトリーにターゲット・ファイルがすでに存在するときは、ASNDLCOPY 出口ルーチンはソース・ファイルの内容を別のターゲット・システム・ファイルに複製します。ASNDLCOPY 出口ルーチンは、ソース・ファイルの内容を一時ファイルに直接コピーします。この一時ファイルの名前は、ソース・ファイルの名前と同じですが、“new ” という接尾部が付けられます。(この接尾部は ASNDLPARM ファイルの中で変更できます。)アプライ・プログラムは次に、結果ファイルからオリジナルのターゲット・ファイルの URL と一時ファイルの URL を受け取ります。アプライ・プログラムが変更をターゲット表に伝搬すると、DB2 は、レプリケーション・トランザクションをコミットするときに、一時ファイルの名前を、オリジナルのターゲットの URL 中のファイル名に変更します。

- ASNDLCOPY 出口ルーチンを変更した場合には、プログラムをコンパイルし、実行可能プログラムを適切なディレクトリーに入れてください。

アプライ・プログラムは ASNDONE 出口ルーチンを、正常に処理されたかエラーが発生したかに関係なく、サブスクリプションの処理が完了した後に呼び出します。そのため、ASNDLCOPY ルーチンが外部ファイルのレプリケーションに失敗した場合は、ASNDONE 出口ルーチンを使って必要なクリーンアップを任意に実行できます。

DLFM_ASNCOPYD の設定と使用方法 (UNIX、Windows)

DB2 Data Links Manager バージョン 8 をインストールしてある場合は、Data Links Manager レプリケーション・デーモン (DLFM_ASNCOPYD) を使用して、DATALINK データ・タイプから参照されるファイルをコピーできます。

ASNDLCOPY 出口ルーチンはソースおよびターゲットの URL をマップした後、ファイルをコピーするためのデーモンに接続します。使用するファイル・コピー・デーモンに接続するようにアドレスとポート番号を指定して、ASNDLUSER 構成ファイルを構成できます。任意の FTP デーモン、または DLFM_ASNCOPYD ファイル・コピー・デーモンを使用できます。

FTP デーモンも DLFM_ASNCOPYD デーモンもソース・ファイル・システムからターゲット・ファイル・システムへ外部ファイルをコピーします。しかし、DLFM_ASNCOPYD ファイル・コピー・デーモンには追加の関数があります。

- RECOVERY YES と定義された DATALINK 列から参照される特定のバージョンのファイルを検索する機能。
- ユーザーのアクセス権に基づいて、READ PERMISSION DB と定義された DATALINK 列から参照されるファイルを検索する機能。
- 複製されたファイルの最終変更時刻を保存する機能。

DLFM_ASNCOPYD の制約事項:

複製したファイルを DLFM_ASNCOPYD を使用してコピーするには、DB2 Universal Database バージョン 8 で、DB2 Data Links Manager バージョン 8 を使用する必要があります。

DLFM_ASNCOPYD ファイル・コピー・デーモンは、次のオペレーティング・システムでしか使用できません。AIX、Solaris™ 実行環境版、Windows NT、および Windows 2000。

FTP の場合の Solaris™ 実行環境版の制約事項:

Solaris™ 実行環境版で DATALINK 列の値を複製する場合、FTP デーモンを使用してファイルをコピーするには、MDTM (modtime) コマンドをサポートする FTP デーモンを使用する必要があります。ソースおよびターゲットのファイル・システムで実行される FTP デーモンは、特定のファイルの最終変更時刻を表示する、MDTM をサポートする必要があります。Solaris™ 実行環境版のバージョン 2.6、または FTP による MDTM サポートを含まないその他のバージョンを使用する場合は、WU-FTPD などの追加ソフトウェアが必要です。

プロシージャ:

DLFM_ASNCOPYD ファイル・コピー・デーモンを設定するには:

1. このファイル・コピー・デーモンへの接続を必要とするユーザーを識別します。
2. これらのファイルが置かれているディレクトリーに基づいて、ファイルへのアクセス権をユーザーに付与します。
3. DLFM_ASNCOPYD デーモンが使用可能であり、正しいポート番号が指定されていることを確認します。

このポート番号は、ASNDLUSER 構成ファイルで指定されているポート番号と一致する必要があります。

詳細については、*DB2 Data Links Manager Quick Beginnings* および *DB2 Data Links Manager Administration Guide and Reference* を参照してください。

データ・リンク・ファイル・マネージャーは、標準の SQL 操作によってアプリケーションがファイルにリンクするたびに、RECOVERY YES として定義された DATALINK 列で、ソース・ファイルの新しいバージョンをアーカイブします。キャプチャー・プログラムは、RECOVERY YES として定義された DATALINK 列で行の変更をキャプチャーすると、ファイルのバージョン番号を記録し、このバージョン番号を CD 表に入れます。アプライ・プログラムは、CD 表でデータの変更とバージョン番号を読み取り、新しい DATALINK 列の値の URL とバージョン番号を ASNDLCOPY 出口ルーチンに渡します。ASNDLCOPY 出口ルーチンが DLFM_ASNCOPYD デーモンに接続すると、このファイル・コピー・デーモンは外部ファイルの一貫性のあるバージョンをリトリートします。

ソース・システム上にファイルのより新しいバージョンが存在している場合も、データ・リンク・ファイル・マネージャーは、CD 表に取り込まれているバージョンと整合するバージョンのファイルを提供します。このため、ターゲット・サーバーは、キャプチャー・プログラムがまだログ内に取り込んでいないバージョンを受け取ることはできません。

ASNDLCOPYD の設定と使用方法 (OS/400)

ASNDLCOPYD は、ASNDLCOPY 出口ルーチンがソースとターゲットの URL をマップした後で、許可ユーザーが OS/400 ソース・サーバーから OS/400 ターゲット・サーバーにファイルをリトリートできるようにするデーモンです。ASNDLCOPY 出口ルーチンはソースおよびターゲットの URL をマップした後、ファイルをリトリートするために ASNDLCOPYD デーモンに接続します。ASNDLCOPYD ファイル・コピー・デーモンは FTP デーモンと似ていますが、DATALINK の値を複製するために次の関数を提供します。

- ファイル情報 (ファイル・サイズや最終変更時刻など) を抽出するためのコマンド
- 特定ファイルの中身を取り出すコマンド

READ PERMISSION DB として定義されている DATALINK 列を複製するために、ASNDLCOPYD ファイル・コピー・デーモンに接続するように ASNDLCOPY 出口ルーチンを構成できます。

ASNDLCOPYD のサンプル・ファイルはライブラリー QDP4、ソース・ファイル QCSRC、メンバー ASNDLCPD に入っています。このサンプル・ファイルは 3 つのプログラムを構築します。

ASNDLCOPYD

メインの親プログラムとファイル・コピー・デーモン。

ASNCHILD

クライアントから ASNDLCOPYD デーモンへの接続を調整するプログラム。ASNCHILD は ASNDLCOPYD デーモンの一部です。クライアントからの要求のたびに、新しい ASNCHILD プロセスが作成されます。

ASNDLCFG

ユーザー ID の追加と除去、そしてユーザー ID のパスワードの変更のための構成プログラム。

注: OS/400 またはその他のオペレーティング・システムで、現在 DB2 バージョン 7 上で ASNDLCOPYD ファイル・コピー・デーモンを使用している場合は、DB2 バージョン 8 で引き続きこのデーモンを使用できます。

前提条件:

ASNDLCOPYD デーモンを実行するにはルート (管理者) 権限が必要です。

プロシージャ:

ASNDLCOPYD ファイル・コピー・デーモンを使用するには:

1. ライブラリー QDP4、ソース・ファイル QCSRC、メンバー ASNDLCPD の ASNDLCOPYD サンプル・プログラムにアクセスします。
2. サイトの要件に合わせてサンプル・プログラムを変更します。
3. プログラム・デーモンを構築します。

- a. 基本モジュールを構築します。

```
CRTCMOD MODULE(libraryname/ASNDLCPD) SRCFILE(QDP4/QCSRC)
          DBGVIEW(*SOURCE) SYSIFCOPT(*ALL)
```

- b. 子プログラム (ASNCHILD) を構築します。

```
CRTPGM PGM(libraryname/ASNCHILD) MODULE(libraryname/ASNDLCPD)
```

- c. 親プログラム (ASNDLCOPYD) を構築します。

```
CRTPGM PGM(libraryname/ASNDLCOPYD) MODULE(libraryname/ASNDLCPD)
```

- d. 構成プログラム (ASNDLCFG) を構築します。

```
CRTPGM PGM(libraryname/ASNDLCFG) MODULE(libraryname/ASNDLCPD)
```

この *libraryname* は既存のライブラリー名です。詳細については、サンプル・プログラムの PROLOG セクションを参照してください。

4. 実行可能プログラムを QDP4 ライブラリーに入れます。
5. サイトの要件に合わせて構成ファイルを変更します。
6. 管理者権限およびスーパーユーザー・アクセスを使用して ASNDLCOPYD デーモンを始動します。ポート番号、および構成ファイルを含むディレクトリーの両方を指定してください。

ASNDLCOPYD ファイル・コピー・デーモンは、ASNDLCOPYD プログラムが生成するすべてのメッセージのためのログ・ファイルを作成します。このログ・ファイルの名前は ASNDLCOPYDYMMDDHHMMSS.LOG (YYYYMMDDHHMMSS は、デーモンの実行開始時刻) になります。

OS/400 の場合、DB2 レプリケーションは必ず、DATALINK 列の値から参照される、最新のバージョンの外部ファイルをレプリケーションします。

関連作業:

- 153 ページの 『第 10 章 アプライ・プログラムの操作』

第 6 章 レプリケーション環境におけるデータのサブセット化

通常、レプリケーションにはなんらかのサブセット化が関与します。レプリケーション・ソースの登録時には、ソース表からレプリケーションする列と行を選択します。サブスクリプション・セットの作成時には、各ターゲット表に複製する登録済み列を選択します。

基本的なサブセット化の方式は、43 ページの『第 3 章 表およびビューをレプリケーション・ソースとして登録する』および 69 ページの『第 4 章 ソースのサブスクリプト』で説明されています。この章では、データをサブセット化するのに使用できるいくつかの高度な技法を説明します。レプリケーション要件に応じて、ソースの場合は登録時に、ターゲットの場合はサブスクリプションの際に、以下の技法を使用してデータをサブセット化できます。

- ソースに対するターゲットが 1 つだけの場合、またはすべてのターゲットがまったく同じデータを必要とする場合は、ニーズがターゲットごとに異なる可能性について考慮する必要がないため、登録時のサブセット化または操作が可能である。
- 1 つのソースに対して複数のターゲットが存在し、アプライされるデータに関する要件がその複数のターゲット間でそれぞれ異なるときは、登録時のサブセット化は不可能な場合がある。このケースでは、データのサブセット化はサブスクリプションの際に行うことになります。

レプリカ・ターゲット表に対するレプリケーションの場合は、これらの技法はどれも使用しないでください。update-anywhere 構成では、マスター表と各レプリカ表は互いにデータを複製し合います。ソース表の使用されていない列が NULL 可能であるかぎり、レプリカ表はソース表の列のサブセットを持てます。そうでない場合は、レプリカ表はソース表と同じ列を含んでいなければならない、そのために列のサブセット化、新しい列の追加、または列の名前変更が不可能 となります。

この章は、以下のセクションを含んでいます。

- 『登録時におけるデータのサブセット化』
- 119 ページの『サブスクリプションの際のデータのサブセット化』

登録時におけるデータのサブセット化

高度な技法を使用して、登録時にデータをサブセット化できます。これらの技法は、データの同じサブセットを 1 度キャプチャーし、そのサブセットを多数のターゲット表に複製する場合に特に便利です。データのサブセット化を、登録されているソースからそのデータが取り込まれる前と後のどちらに行うのかを選択できます。このセクションで

示されている技法は、 update-anywhere レプリケーションまたは対等レプリケーション以外のすべてのレプリケーション構成で使用できます。

登録時にデータをサブセット化すると、レプリケーションのパフォーマンスが全体的に向上する可能性があります。なぜなら、キャプチャー・プログラムが CD 表に追加するデータの量が削減されて、アプライ・プログラムが読み取るデータ量が削減されるからです。また、CD 表内の行数が少なくなるため、ストレージも削減されます。

このセクションでは、登録時にデータをサブセット化できる以下の方法について説明します。

- 『ビューを使用したソース・データのサブセット化』
- 『特定の行のキャプチャーを防止するために CD 表にトリガーを定義する (UNIX、Windows、z/OS)』

ビューを使用したソース・データのサブセット化

ソースの登録時には、レプリケーションに使用できるようにする列を選択します。選択した列はレプリケーションのために取り込まれます。いくつかのケースでは、変更レプリケーション用のソースを登録した後で、そのソースのビューを登録する必要が生じることがあります。

たとえば、人材部門では給与情報が組み込まれた人事データを含む表が保守されているとします。バックアップ・データベースを保守するために、この人事表全体がバックアップ・サイトに登録され、このサイトからサブスクライブされています。しかし、別のターゲット・サイトがこの人事表にサブスクライブする必要がある場合は、給与情報はこの 2 番目のサブスクライバーから隠蔽しなければならないことがあります。解決策は、この人事表に対するビューを登録し、2 番目のサブスクライバー用に登録されたビューだけにアクセス権を許可して、給与情報がアクセスから保護されるようにすることです。サブスクリプションは、この登録されたビューに対して作成できます。

また、複数のソース表が組み込まれたビューも登録できます。たとえば、カスタマー表と支店表がある場合、カスタマーを適切にサブセット化してターゲットに正しく複製する唯一の方法は、この 2 つの表を結合して、特定の支店のカスタマーだけが特定のターゲットに複製されるようにすることです。このケースでは、二重削除を防止するための配慮が必要です。

特定の行のキャプチャーを防止するために CD 表にトリガーを定義する (UNIX、Windows、z/OS)

ソースの登録時にレプリケーション・センターを使用すると、キャプチャーする列を選択できますが、それらの行における特定の変更がレプリケーションされるのを防止する機能は提供されません。レプリケーションのシナリオによっては、行内の特定の変更が取り込まれてターゲット表にレプリケーションされるのを防止したい場合があります。

たとえば、ターゲット表にはすべての行が含まれているようにし、それらの表からはどの行も削除されないようにしたい場合は、ソースから削除が複製されないようにする必要があります。

特定の変更が取り込まれるのを抑制するには、CD 表にトリガーを定義してください。これらのトリガーにより、キャプチャー・プログラムが無視する必要のある変更と、CD 表に行を追加してはならないことが指定されます。これらのトリガーは、レプリケーション・センターでは作成できませんが、既存の CD 表に対して（つまり、ソースを登録した後で）手動で作成できます。値が 99999 の SQLSTATE が表示されるトリガーの失敗はすべてキャプチャー・プログラムによって無視され、その行は CD 表に挿入されません。

例: 表 SAMPLE.TABLE (CD 表は SAMPLE.CD_TABLE) からのレプリケーション中はソース表でのすべての DELETE 操作を抑制したいとします。以下のトリガーは、DELETE 操作であるすべての行が CD 表に挿入されるのを抑制します。

```
CREATE TRIGGER SAMPLE.CD_TABLE_TRIGGER
NO CASCADE BEFORE INSERT ON SAMPLE.CD_TABLE
REFERENCING NEW AS CD
FOR EACH ROW MODE DB2SQL
WHEN (CD.IBMSNAP_OPERATION = 'D')
SIGNAL SQLSTATE '99999' ('CD INSERT FILTER')
```

登録時に生成された SQL にこの CREATE TRIGGER ステートメントを追加する場合は、登録を完了して CD 表にトリガーを作成するために、その変更した SQL を実行する必要があります。

これらのトリガーは、キャプチャー・プログラムが CD 表に対する行の挿入を試行するたびに実行されます。そのため、ここでのトリガーの使用がレプリケーション環境での最高のパフォーマンスをもたらすものなのかどうかについて検討する必要があります。CD 表にトリガーを追加すると、データ・スループットが上がる場合と下がる場合があります。CD 表のトリガーは、ソースで著しく多発する変更を抑制する場合に使用するようにしてください。変更の大部分をキャプチャーする予定だが、そのうちの一部の変更については複製されるのを抑制したいという場合は、不要な行をサブスクリプションの際に抑制します。

サブスクリプションの際のデータのサブセット化

このセクションでは、サブスクリプションの際に述部を使用して行をサブセット化する方法を説明します。サブスクリプションの際にデータをサブセット化すると、レプリケーションのパフォーマンスが全体的に向上する可能性があります。なぜなら、アプライ・プログラムが取り出すデータの量が削減されるからです。また、ターゲット表内の行数が少なくなるため、ストレージも削減されます。

アプライ・プログラムは、フル・リフレッシュ・レプリケーション時および変更キャプチャー・レプリケーション時にどのデータをコピーするのかを述部を使用して判別しま

す。レプリケーション・センターでは、フル・リフレッシュ・レプリケーション用および変更キャプチャー・レプリケーション用の述部値を指定できます。変更キャプチャー・レプリケーションの場合にのみ使用する補足的な述部情報を追加したいときがあります。なぜなら、この情報はフル・リフレッシュ時には利用不能であるからです。この補足的な述部情報は、ユーザーが提供する SQL を使用して、UOW_CD_PREDICATES 列のサブスクリプション・セット・メンバー (IBMSNAP_SUBS_MEMBR) 表に追加する必要があります。

たとえば、ALL.CUSTOMERS という名前の登録済みの表があり、それに関連した CD 表の名前が ALL.CD_CUSTOMERS であるとします。サブスクリプション・ターゲットには ALL.CUSTOMERS のサブセットだけを含めたいとします (ALL.CUSTOMERS の ACCT_BALANCE 列は 50000 を超えます)。さらに、ターゲット表のヒストリー・データを保守する必要がある (つまり、ターゲット表内のデータはどれも削除されたくない) とします。レプリケーション・センターを使用すれば、PREDICATES 値が 'ACCT_BALANCE > 50000' のサブスクリプション・セット・メンバーを作成できます。

レプリケーション・センターを使用する場合は、ターゲット表での削除は防止できません。なぜなら、操作のタイプに関する情報は CD 表に保管されるため、ソース表またはビューではこの情報が利用不能だからです。したがって、以下の情報が組み込まれた SQL ステートメントを使用して、追加の変更キャプチャー述部を生成する必要があります。⁵

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR SET UOW_CD_PREDICATES = 'IBMSNAP_OPERATION <>"D"'
WHERE APPLY_QUAL = 'apply_qual' AND SET_NAME = 'set_name' AND
SOURCE_OWNER = 'ALL' AND SOURCE_TABLE = 'CUSTOMERS'
```

CD 表内の変更前イメージ列、CD 表からのあらゆるオーバーヘッド列、または UOW 表からのあらゆる列などといった、フル・リフレッシュ時に利用不能な列を参照するサブスクリプション・セット・メンバー述部の場合は、必ず UOW_CD_PREDICATES 列を手動でセットアップする必要があります。

デフォルトでは、ターゲット表がユーザー・コピーである場合でも、アプライ・プログラムは UOW 表と CD 表を結合しません。このプログラムは CD 表から直接データをフェッチして適用します。述部が UOW 表を参照する必要があり、かつターゲット表がユーザー・コピーである場合は、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表の JOIN_UOW_CD 列の値を Y に設定する必要があります。このフラグを設定すれば、UOW 表と CD 表がアプライ・プログラムによって確実に結合されます。

5. シナリオによっては、この更新ステートメントに列を追加して、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表の単一行が確実に更新されるようにしなければならない場合があります。

行のサブセットに対して 1024 バイト (サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表の PREDICATES 列の容量) を超える述部を指定する場合は、ソース・ビューを使用する必要があります。

第 7 章 レプリケーション環境におけるデータ操作

ターゲット表のデータは、ソース表における表示ほど正確に表示される必要はありません。ソース・データは、ターゲット表に複製される前にトランスフォームまたは拡張することができます。たとえば、データ整理の実行、データ集約の実行、あるいはソースにないターゲット表の列の移植のようなデータ操作を行う場合があります。

本章では、データをトランスフォームするために使用できる幾つかの先進的な技法について説明します。

データ操作は、登録済みソースからのキャプチャー前、キャプチャー後のいずれかで行うことができます。データを 1 回操作して、トランスフォームされたデータを多数のターゲット表に複製する場合は、サブスクリプションではなく登録で操作します。すべてのソース・データをキャプチャーし、トランスフォームされたデータを選択して個々のターゲットに適用する場合は、登録ではなくサブスクリプション中にデータを操作します。

レプリケーションのシナリオによっては、CD 表に保管されたソース・データの内容を操作することになります。トリガー、サブスクリプション中の式、あるいはソース・ビューのどれを使用しても、同じジョブを実行できます。それぞれの方式に、良い点と悪い点があります。トリガーは、使用される CPU の点からコストがかかりすぎる可能性があります。関数をセットアップするのに、サブスクリプションは複数回必要ですが、ビューを使用すると 1 回でセットアップできます。

たとえば、ソース表で特定の値が脱落している場合、キャプチャー・プログラムで NULL 値をキャプチャーする必要のない場合があります。

CD 表に対してトリガーを使用して、データを CD 表に挿入するときにキャプチャー・プログラムがデータを拡張する条件を指定できます。この場合、キャプチャー・プログラムがソースで NULL 値を発見したら、デフォルト値を CD 表に挿入するように指定できます。データがソース表の更新から脱落している場合、以下のコードを使用して、確定したデフォルトを提供するトリガーを作成できます。

```
CREATE TRIGGER ENHANCECD
NO CASCADE BEFORE INSERT ON CD_TABLE
REFERENCING NEW AS CD
FOR EACH ROW MODE DB2SQL
WHEN (CD.COL1 IS NULL)
SET CD.COL1 = 'MISSING DATA'
END
```

トリガーの代わりに、登録済みソース・ビューまたはサブスクリプションの式で DB2 の COALESCE スカラー関数を使用できます。登録済みビューで、この合体関数が最初の非 NULL 値を戻します。

ソース・ビューの使用例の一部:

```
CREATE VIEW SAMPLE.SRCVIEW (columns) AS SELECT
    ... COALESCE(A.COL1, 'MISSING DATA') ...
FROM SAMPLE.TABLE A
```

式の使用例の一部:

```
COALESCE(CD.COL1, 'MISSING DATA')
```

アプライ・プログラムは、データをターゲットに適用する前または後どちらかに、以下の方法でデータを操作できます。

- 『ストアド・プロシージャーまたは SQL ステートメントを使用したデータ拡張』
- 125 ページの『名前が異なるソース列とターゲット列のマッピング』
- 125 ページの『算出列の作成』

ストアド・プロシージャーまたは SQL ステートメントを使用したデータ拡張

サブスクリプション・セット情報を定義するとき、特定のセットを処理するたびにアプライ・プログラムに実行させる実行時処理ステートメントを、SQL ステートメントまたはストアド・プロシージャーを使って定義できます。これらの実行時処理により、レプリケーション中にデータを操作できるようになります。このようなステートメントは、CCD 表のプルーニングやサブスクリプション・セット処理でのシーケンス・コントロールにも役立ちます。実行時処理ステートメントは、サブスクリプション・セットの処理前ならキャプチャー・コントロール・サーバーで、あるいはサブスクリプション・セットの処理前または後であればターゲット・サーバーで実行できます。たとえば、データを取り出す前、またはターゲット表にデータを複製した後、あるいはその両方の場合に SQL ステートメントを実行することができます。

ストアド・プロシージャーは、パラメーターなしの SQL CALL ステートメントを使用します。プロシージャー名の長さは、18 文字以下でなければなりません (OS/400 の場合は、最大 128 文字です)。ソースまたはターゲット表が非 DB2 リレーショナル・データベースにある場合、SQL ステートメントは連合 DB2 データベースを対象として実行されます。SQL ステートメントが非 DB2 データベースを対象として実行されることは決してありません。各タイプの実行時プロシージャーは単一トランザクションとして一緒に実行されます。さらに、それぞれのステートメントごとに受け入れ可能 SQLSTATE を定義することもできます。

それぞれのセットの完了後に (特定のセットの完了後ではなく) データを操作する場合は、ASNDONE 出口ルーチンを使用します。

名前が異なるソース列とターゲット列のマッピング

レプリケーション・センターを使用してターゲット表を作成しているとき、ターゲット表のタイプに関係なく、ターゲット表で列を名前変更できます。さらに、列の属性（データ・タイプ、長さ、位取り、精度、NULL 可能）が互換性のある場合はその属性の変更もできます。レプリケーション・センターを使用して、既存のターゲット表の列を名前変更することはできません。ソース列とターゲット列が一致しない場合は、レプリケーション・センターを使用してソースからターゲットに列をマップするか、あるいはソースの列名に一致する名前を含むターゲット表のビューを作成するかのどちらかが可能です。

算出列の作成

既存のターゲット表の列名を変更することはできませんが、ソース列の式を変更して、既存のターゲット表の列に正しくマップするようしたり、あるいはそれとの互換性を持たせることができます。SQL 式を使用すれば、既存のソース列から新しい列を派生させることもできます。集約ターゲット表タイプでは、COUNT や SUM といった集約関数を使って新しい列を定義できます。ターゲット表の他のタイプでは、式でスカラー関数を使用して新しい列を定義できます。ソース表とターゲット表の列の名前だけが違って、他は互換性がある場合、レプリケーション・センターを使用すれば列をもう一方の表の列にマップできます。

たとえば、既存のソース表 (SRC.TABLE) とターゲット表 (TGT.TABLE) があるとします。

```
CREATE TABLE SRC.TABLE (SRC_COL1 CHAR(12) NOT NULL, SRC_COL2 INTEGER,
    SRC_COL3 DATE, SRC_COL4 TIME, SRC_COL5 VARCHAR(25))
CREATE TABLE TGT.TABLE (TGT_COL1 CHAR(12) NOT NULL,
    TGT_COL2 INTEGER NOT NULL, TGT_COL3 TIMESTAMP, TGT_COL4 CHAR(5))
```

以下のステップで、サブスクリプション中に算出列を使用して必要なターゲット表をマップします。

1. レプリケーション・センターを使用して、ソース表の SRC_COL1 をターゲット表の TGT_COL1 にマップする。これらの列には互換性があるので、一方から他方にマップするのに式を使う必要はありません。
2. 算出列の式 COALESCE(SRC_COL2, 0) を使用して TGT_COL2 にマップし、データを提供する。SRC_COL2 は NULL 可能で TGT_COL2 は非 NULL のため、非 NULL の値が必ず TGT_COL2 に入るようにこのステップを実行しなければなりません。
3. 算出列の式 TIMESTAMP(CHAR(SRC_COL3) CONCAT CHAR(SRC_COL4)) を使用して TGT_COL3 にマップし、データを提供する。この列式はターゲット・データベースのタイム・スタンプ列にマップするデータを提供します。

4. 算出列の式 `SUBSTR(SRC_COL5, 1,5)` を使用して `TGT_COL4` にマップし、データを提供する。

第 8 章 レプリケーション SQL スクリプトのカスタマイズおよび実行

コントロール表の作成、ソース表の登録、およびサブスクリプション・セットの作成のためには、レプリケーション・センターで生成される SQL スクリプトを実行しなければなりません。SQL スクリプトは、レプリケーション・センターまたはタスク・センターを使用して、あるいは DB2 コマンド行から実行できます。SQL スクリプトは、必要に応じて、ユーザーの要求を満たすように変更できます。

レプリケーション・センターには、生成された SQL スクリプトをすぐに実行するためのオプション、あるいはそれをタスクとして保管するかまたはファイルに保管し、後でスクリプトを実行するためのオプションがあります。SQL をレプリケーション・センターから実行するように選択する場合でも、今後の参照用に、SQL をタスクとして保管するかまたはファイルに保管することが必要な場合もあります。たとえば、大規模なレプリケーション・サブスクリプション・セットの定義を SQL ファイルに保存しておけば、必要なときにその定義を再実行できます。

生成済みの SQL スクリプトを編集するときは、終了文字を変更しないように注意してください。また、ファイルに保管されるスクリプトが複数ある場合は、スクリプト区切り文字を変更しないでください。

以下のタスクを行うために、SQL スクリプトをユーザーの環境に合わせてカスタマイズすることがあります。

- 同一のレプリケーション・アクション (複数のサーバー用にカスタマイズされたもの) の複数のコピーを作成する。
- CD 表の表スペースまたはデータベースをサイズ変更する。
- サイト別の標準を定義する。
- 定義を結合してバッチ・ジョブとして実行する。
- 指定時刻までレプリケーション・アクションの実行を延期する。
- バックアップまたはサイト専用のカスタマイズしたい SQL スクリプトのライブラリーを作成したり、不定期接続の環境におけると同様に、分散サイトでスタンドアロン実行したりする。
- 表や索引の作成ステートメントを編集し、データベース・オブジェクトを表すようにする。
- Informix および非 DB2 リレーショナル・データベースの場合、目標の表スペースに表が確実に作成されるようにする。
- Microsoft SQL Server の場合、既存のセグメントにコントロール表を作成する。

- 複数のサブスクリプション・セットを同時に定義する方法として、サブスクリプション・セット・メンバーの述部を表示また編集する。述部に置換変数を使用して、その変数をプログラミング論理によって解決することができます。

SQL スクリプトを DB2 コマンド行から実行する場合は、SQL スクリプトの実行時にサーバーに手動で接続しなければなりません。生成されたスクリプトには、CONNECT ステートメントが含まれています。SQL スクリプトを実行する前に、ユーザー ID とパスワードをサーバーに指定する SQL ステートメントを編集しなければなりません。たとえば、以下の例に似た行を探し、プレースホルダー (XXXX) を上書きして情報を追加します。

```
CONNECT TO srcdb USER XXXX USING XXXX ;
```

プロシージャ:

次の方法のどちらかを使用して、SQL スクリプトが含まれたファイルを DB2 コマンド行から実行します。

- SQL スクリプトにセミコロン (;) が終了文字として含まれている場合は、このコマンドを使用する。

```
db2 -tvf filename
```

- SQL スクリプトに区切り文字として他の文字が含まれている場合は (この例では、異機種のレプリケーションなどでポンド記号 (#) が終了文字)、このコマンドを使用する。

```
db2 -td# -vf filename
```

推奨事項: スクリプトの実行前には、常に管理ログ・ファイルを読んでください。

第 9 章 キャプチャー・プログラムの操作

この章は、DB2 データベースに対するログ・ベースのキャプチャーに関するものです。トリガー・ベースのキャプチャーを使用している場合は、登録時にトリガーが作成されるため、ユーザーはこの章で説明されている操作を実行しません。

この章は、以下のセクションを含んでいます。

- 『キャプチャー・プログラムのデフォルトの稼働パラメーター』
- 131 ページの『キャプチャー・プログラムの稼働パラメーターの変更』
- 133 ページの『キャプチャー・プログラムの始動 (UNIX、Windows、z/OS)』
- 145 ページの『キャプチャー・プログラムの始動 (OS/400)』
- 146 ページの『実行中のキャプチャー・プログラムの動作の変更』
- 147 ページの『キャプチャー・パラメーター表の稼働パラメーターの変更』
- 148 ページの『キャプチャー・プログラムの停止』
- 149 ページの『キャプチャーの延期 (UNIX、Windows、z/OS)』
- 150 ページの『キャプチャーの再開 (UNIX、Windows、z/OS)』
- 150 ページの『キャプチャーの再初期化』

重要: DB2 ユーティリティは、変更をキャプチャー・プログラムから見える方法ではログに記録しません。そのため、キャプチャー・プログラムは DB2 ユーティリティによって行われた変更を一切キャプチャーしません。

キャプチャー・プログラムのデフォルトの稼働パラメーター

キャプチャーには、デフォルト値のあるパラメーターがいくつかあります。製品出荷時のデフォルト値は、表 5 および 130 ページの表 6 に示されています。ほとんどの稼働パラメーターのデフォルト値は、出荷時にはキャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表に保管されています。これらのデフォルトをご使用のレプリケーション環境で使用し、必要に応じて 131 ページの『キャプチャー・プログラムの稼働パラメーターの変更』で説明されている方法のいずれかを使用して変更してください。

表 5. キャプチャー稼働パラメーターのデフォルト設定 (UNIX、Windows、z/OS)

| 稼働パラメーター | デフォルト値 | IBMSNAP_CAPPARMS 表の列名 |
|-----------------|-----------------------|-----------------------|
| capture_server | DB2DBDFT ¹ | 該当せず |
| capture_schema | ASN ² | 該当せず |
| retention_limit | 10080 分 | RETENTION_LIMIT |
| lag_limit | 10080 分 | LAG_LIMIT |

表 5. キャプチャー稼働パラメーターのデフォルト設定 (UNIX、Windows、z/OS) (続き)

| 稼働パラメーター | デフォルト値 | IBMSNAP_CAPPARMS 表の列名 |
|------------------|----------------------------------|-----------------------|
| commit_interval | 30 秒 | COMMIT_INTERVAL |
| prune_interval | 300 秒 | PRUNE_INTERVAL |
| trace_limit | 10080 分 | TRACE_LIMIT |
| monitor_limit | 10080 分 | MONITOR_LIMIT |
| monitor_interval | 300 秒 | MONITOR_INTERVAL |
| memory_limit | 32 MB | MEMORY_LIMIT |
| autoprune | y ³ | AUTOPRUNE |
| term | y ³ | TERM |
| autostop | n ⁴ | AUTOSTOP |
| logreuse | n ⁴ | LOGREUSE |
| logstdout | n ⁴ | LOGSTDOUT |
| sleep_interval | 5 秒 | SLEEP |
| capture_path | キャプチャーが始動されたディレクトリー ⁵ | CAPTURE_PATH |
| startmode | warmst ⁶ | STARTMODE |

注:

1. キャプチャー・コントロール・サーバーは、DB2DBDFT 環境変数の値である (指定されている場合)。
2. キャプチャー・スキーマのデフォルトは変更できない。別のキャプチャー・スキーマを使用するには、**capture_schema** 始動パラメーターを使用する。
3. はい
4. いいえ
5. キャプチャーが Windows サービスとして始動する場合、そのキャプチャー・パスは `sqlib\%bin` である。
6. キャプチャー・プログラムはウォーム・スタートする。これがこのプログラムの初回の始動である場合にのみ、コールド・スタートに切り替わる。

これらの稼働パラメーターとそのデフォルトについての詳細は、133 ページの『キャプチャー・プログラムの始動 (UNIX、Windows、z/OS)』を参照してください。

表 6. キャプチャー稼働パラメーターのデフォルト設定 (OS/400)

| 稼働パラメーター | デフォルト値 | IBMSNAP_CAPPARMS 表の列名 |
|-----------|----------------------|-----------------------|
| CAPCTLLIB | ASN ¹ | 該当せず |
| JOBID | *LIBL/QZSNDPR | 該当せず |
| JRN | *ALL | 該当せず |
| RETAIN | 10080 分 | RETENTION_LIMIT |
| LAG | 10080 分 | LAG_LIMIT |
| FRCFRQ | 30 秒 | COMMIT_INTERVAL |
| CLNUPITV | *IMMED ² | 該当せず |
| CLNUPITV | 86400 秒 ² | PRUNE_INTERVAL |
| CLNUPITV | *IMMED ² | 該当せず |
| TRCLMT | 10080 分 | TRACE_LIMIT |
| MONLMT | 10080 分 | MONITOR_LIMIT |

表 6. キャプチャー稼働パラメーターのデフォルト設定 (OS/400) (続き)

| 稼働パラメーター | デフォルト値 | IBMSNAP_CAPPARMS 表の列名 |
|----------|-------------------|-----------------------|
| MONITV | 300 秒 | MONITOR_INTERVAL |
| MEMLMT | 32 MB | MEMORY_LIMIT |
| WAIT | 120 秒 | 該当せず |
| RESTART | *YES ³ | 該当せず |

注:

1. キャプチャー・スキーマのデフォルトは変更できない。別のキャプチャー・スキーマを使用するには、キャプチャー・プログラムの始動時に CAPCTLLIB パラメーターを指定する。他のほとんどの稼働パラメーターのデフォルト値は、出荷時にはキャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表に保管されている。
2. CLNUPITV には 2 つのサブパラメーターがある。デフォルトでは、キャプチャー・プログラムは実行開始直後にブルーニングを実行し、ブルーニング・インターバルに達するたび (デフォルトでは 24 時間ごと) に再びブルーニングを実行する。
3. デフォルトでは、キャプチャー・プログラムはウォーム・スタートする。

これらの稼働パラメーターとそのデフォルトについての詳細は、365 ページの『第 18 章 レプリケーション用のシステム・コマンド (OS/400)』を参照してください。

キャプチャー・プログラムの稼働パラメーターの変更

稼働パラメーターのデフォルト値を、ご使用の環境で通常使用している値に変更できます。これらのデフォルト値は、キャプチャー・プログラムの始動時にオーバーライドできます。あるいは、キャプチャー・プログラムの実行中に変更できます。

IBMSNAP_CAPPARMS 表での新しいデフォルト値の設定

キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表には、キャプチャー・プログラムの動作をコントロールするために変更できるパラメーターが含まれています。この表のスキーマ名はキャプチャー・スキーマです。この表が作成されると、その中にはキャプチャー・プログラム用の出荷時のデフォルト値が入れられます。CAPPARMS 表の列値が設定されていない場合は、129 ページの表 5 および 130 ページの表 6 に示されているハードコーディングされたデフォルト値が使用されます。表内の値の変更方法についての詳細は、147 ページの『キャプチャー・パラメーター表の稼働パラメーターの変更』を参照してください。

キャプチャー・プログラムの始動時に行うパラメーター値の指定

キャプチャー・プログラムの始動時に、このプログラム用の値を指定できます。始動時に設定された値により、現行セッションの間のキャプチャーの動作がコントロールされ、デフォルトの稼働パラメーター値およびキャプチャー・パラメーター表内に存在する可能性のあるあらゆる値がオーバーライドされます。キャプチャー・パラメーター表内の値については、これらの値による更新は行われません。キャプチャー・プログラムを始動する前にキャプチャー・パ

ラメーター表を変更せず、さらにキャプチャー・プログラムの始動時にパラメーターをまったく指定しなかった場合は、稼働パラメーターに対してデフォルト値が使用されます。

キャプチャー・プログラムの実行中に行うパラメーター値の変更

キャプチャーの実行中に、その稼働パラメーターを一時的に変更できます。キャプチャー・プログラムは、ユーザーが値を再び変更するまで、またはキャプチャー・プログラムを停止して再始動するまで、その新しい値を使用します。キャプチャー・パラメーターは、セッションの間必要なだけ何度でも変更できます。詳細については、146 ページの『実行中のキャプチャー・プログラムの動作の変更』を参照してください。

例 (UNIX、Windows): キャプチャー・スキーマ ASNPROD のキャプチャー・コミット・インターバルのデフォルト設定を使用したくないとします。

1. ASNPROD キャプチャー・スキーマ用のキャプチャー・パラメーター表を更新する。コミット・インターバルを 60 秒に設定してください。これにより、今後のキャプチャー・プログラムの始動時にはデフォルト・コミット・インターバルが 60 秒に設定されます。

```
update asnprod.ibmnap_capparms set commit_interval=60;
```

2. 場合によっては、なんらかのパフォーマンス・チューニングが必要となることもあり、そのために、比較的小さな値を指定したコミット・インターバルを使用してキャプチャーを始動してことがあります。キャプチャー・パラメーター表内の値を変更する代わりに、単に 20 秒に設定したコミット・インターバルのパラメーターを使用してキャプチャー・プログラムを始動してください。キャプチャー・プログラムが 20 秒のコミット・インターバルを使用して稼働している間、このプログラムのパフォーマンスをモニターします。

```
asncap capture_server=srcdb1 capture_schema=asnprod commit_interval=20
```

3. 値をさらに小さくしたコミット・インターバルを試してみる。キャプチャー・プログラムを停止する代わりに、コミット・インターバルを 15 秒に設定するパラメーターの変更要求をサブミットしてください。キャプチャー・プログラムは稼働を継続しますが、ただし 15 秒ごとにデータをコミットするようになります。

```
asnccmd capture_server=srcdb1 capture_schema=asnprod chgparms  
commit_interval=15
```

重要: 変更するパラメーターは、**chgparms** コマンドの直後に指定する必要があります。

4. 引き続きパフォーマンスのモニターとコミット・インターバルのパラメーターの変更を行う。キャプチャー・プログラムを停止する必要はありません。最終的に、ニーズに合ったコミット・インターバルを見つけたときは、キャプチャー・パラメーター表を (ステップ 1 で説明されている方法で) 更新して、次の始動時にキャプチャー・プログラムがその新しい値をデフォルト・コミット・インターバルとして使用するようになります。

例 (OS/400): キャプチャー・スキーマ ASNPROD のキャプチャー・コミット・インターバルのデフォルト設定を使用したくないとします。

1. ASNPROD キャプチャー・スキーマ用のキャプチャー・パラメーター表を更新する。コミット・インターバルを 90 秒に設定してください。これにより、今後のキャプチャー・プログラムの始動時にはデフォルト・コミット・インターバルが 90 秒に設定されます。

```
CHGDPRCAP CAPCTLLIB(ASNPROD) FRCFRQ(90)
```

2. 場合によっては、なんらかのパフォーマンス・チューニングが必要となることもあり、そのために、比較的小さな値を指定したコミット・インターバルを使用してキャプチャーを始動してることがあります。キャプチャー・パラメーター表内の値を変更する代わりに、45 秒に設定したコミット・インターバルのパラメーターを使用してキャプチャー・プログラムを始動してください。キャプチャー・プログラムが 45 秒のコミット・インターバルを使用して稼働している間、このプログラムのパフォーマンスをモニターしてください。

```
STRDPRCAP CAPCTLLIB(ASNPROD) FRCFRQ(45)
```

3. 値をさらに小さくしたコミット・インターバルを試してみる。キャプチャー・プログラムを停止する代わりに、コミット・インターバルを 30 秒に設定するパラメーターの変更要求をサブミットしてください。キャプチャー・プログラムは稼働を継続しますが、ただし 30 秒ごとにデータをコミットするようになります (注: OS/400 では、30 秒より短いコミット・インターバルは設定できません)。

```
OVRDPRCAP CAPCTLLIB(ASNPROD) FRCFRQ(30)
```

4. 最終的に、ニーズに合ったコミット・インターバルを見つけたときは、キャプチャー・パラメーター表を (ステップ 1 で説明されている方法で) 更新して、次の始動時にキャプチャー・プログラムがその新しい値をデフォルト・コミット・インターバルとして使用するようになります。

キャプチャー・プログラムの始動 (UNIX、Windows、z/OS)

キャプチャー・プログラムを始動して、ログからのデータのキャプチャーを開始してください。キャプチャー・プログラムがデータをキャプチャーするのは、DB2 データベースからだけです。非 DB2 リレーショナル・ソースから変更をキャプチャーするためにトリガー・ベースのキャプチャーを使用している場合は、登録時にトリガーが作成されるため、キャプチャー・プログラムを始動する必要はありません。

重要: DB2 ユーティリティは、変更をキャプチャー・プログラムから見える方法ではログに記録しません。そのため、キャプチャー・プログラムは DB2 ユーティリティによって行われた変更を一切キャプチャーしません。

キャプチャー・プログラムの始動後、キャプチャー・プログラムがデータのキャプチャーをすぐには開始しないことがあります。このプログラムがデータのキャプチャーを開始するのは、ターゲット表を完全にリフレッシュしたことをアプライ・プログラムがキ

ャプチャー・プログラムにシグナルで通知した後だけです。このシグナルを受け取ると、キャプチャー・プログラムは所定のソース表用のログからの変更のキャプチャーを開始します。

ヒント: キャプチャー・ログ・ファイル (UNIX および Windows の場合は `db2instance.capture_server.capture_schema.CAP.log`、z/OS の場合は `capture_server.capture_schema.CAP.log`) の中に、変更のキャプチャーが開始されたことを示すメッセージがないかどうか調べます。以下に例を示します。

```
ASN0104I Change capture has been started for the source
table "REGRESS.TABLE1" for changes found in the log beginning
with log sequence number "0000:0275:6048".
```

前提条件:

キャプチャー・プログラムを始動する前に、以下の前提条件が満たされていることを必ず確認してください。

- ソース・サーバーおよびキャプチャー・コントロール・サーバーへの接続が構成済みである。
- 正しい許可を受けている。
- 適切なキャプチャー・スキーマ用のコントロール表が作成されており、登録が定義済みである。
- レプリケーション・プログラムが構成済みである。

プロシージャ:

以下の方法のいずれかを使用して、DB2 for UNIX、DB2 for Windows、および DB2 for z/OS でキャプチャー・プログラムを始動してください。

レプリケーション・センター

「キャプチャーの開始」ウィンドウを使用して、レプリケーション・センターのオブジェクト・ツリー内において選択されているキャプチャー・コントロール・サーバーのキャプチャー・スキーマによって識別されるキャプチャー・プログラムを実行します。詳細については、レプリケーション・センターのヘルプを参照してください。

asncap システム・コマンド

コマンド構文およびパラメーター記述の詳細については、334 ページの『`asncap`: キャプチャーの開始 (UNIX、Windows、z/OS)』を参照してください。

MVS コンソールまたは TSO (z/OS)

詳細については、461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』を参照してください。

Windows サービス (Windows)

詳細については、467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』を参照してください。

キャプチャー・プログラムの始動にどの手順を使用するかにかかわらず、始動パラメーターを選択できます。次のセクションでは、始動パラメーターについて説明し、各パラメーターで選択するべき値を推奨します。

- 『autoprune (UNIX、Windows、z/OS)』
- 136 ページの 『autostop (UNIX、Windows、z/OS)』
- 136 ページの 『capture_path (UNIX、Windows、z/OS)』
- 137 ページの 『capture_schema (UNIX、Windows、z/OS)』
- 138 ページの 『capture_server (UNIX、Windows、z/OS)』
- 138 ページの 『commit_interval (UNIX、Windows、z/OS)』
- 138 ページの 『lag_limit (UNIX、Windows、z/OS)』
- 139 ページの 『logreuse (UNIX、Windows、z/OS)』
- 139 ページの 『logstdout (UNIX、Windows、z/OS)』
- 140 ページの 『memory_limit (UNIX、Windows、z/OS)』
- 140 ページの 『monitor_interval (UNIX、Windows、z/OS)』
- 140 ページの 『monitor_limit (UNIX、Windows、z/OS)』
- 141 ページの 『prune_interval (UNIX、Windows、z/OS)』
- 142 ページの 『retention_limit (UNIX、Windows、z/OS)』
- 142 ページの 『sleep_interval (UNIX、Windows、z/OS)』
- 143 ページの 『startmode (UNIX、Windows、z/OS)』
- 144 ページの 『term (UNIX、Windows、z/OS)』
- 144 ページの 『trace_limit (UNIX、Windows、z/OS)』

autoprune (UNIX、Windows、z/OS)

デフォルト: `autoprune=y`

autoprune パラメーターは、キャプチャー・プログラムがそのコントロール表のいくつかを自動的に整理するかどうかを指定します。デフォルト (`autoprune=y` を使用) では、キャプチャー・プログラムは CD 表と UOW 表の行、ならびにキャプチャー・トレース (IBMSNAP_CAPTRACE) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、および シグナル (IBMSNAP_SIGNAL) 表の行を自動的に整理します。

autoprune=n を設定した場合は、プルーニング・コマンドを使用してこれらの表を整理する必要があります。

自動ブルーニングをオンに設定してキャプチャーを始動する場合は、ブルーニング・インターバルを設定して、使用しているレプリケーション環境でのブルーニング頻度を最適化してください。詳細については、141 ページの『`prune_interval` (UNIX、Windows、z/OS)』を参照してください。

キャプチャー・プログラムは以下のパラメーターを使用して、どの行が整理されてもよいほど古いのかを判別します。

- 142 ページの『`retention_limit` (UNIX、Windows、z/OS)』(CD 表、UOW 表、およびシグナル表の場合)
- 140 ページの『`monitor_limit` (UNIX、Windows、z/OS)』(モニター表の場合)
- 144 ページの『`trace_limit` (UNIX、Windows、z/OS)』(キャプチャー・トレース表の場合)

表のブルーニングについての詳細は、250 ページの『コントロール表のブルーニング』を参照してください。

autostop (UNIX、Windows、z/OS)

デフォルト: `autostop=n`

autostop パラメーターは、ログの終わりに達した後にキャプチャー・プログラムが稼働し続けるか終了するかをコントロールします。

デフォルト (`autostop=n`) では、キャプチャー・プログラムはトランザクションの検索後に終了しません。

モバイル環境または時々しか接続しない環境でレプリケーションを行っている場合は **autostop=y** オプションを使用してください。 `autostop` を使用すると、キャプチャー・プログラムは確実にすべての適格トランザクションを検索し、ログの終わりに達した時点で必ず停止します。それ以外のトランザクションも検索する場合は、キャプチャーを再始動する必要があります。 **autostop=y** オプションは、テスト環境でも使用されません。

推奨: ほとんどのケースでは、`autostop=y` を使用すべきではありません。なぜなら、これを使用すると、レプリケーションの管理に多大なオーバーヘッドが加えられるからです (たとえば、キャプチャー・プログラムを始終再始動する必要があります)。

capture_path (UNIX、Windows、z/OS)

キャプチャー・パスは、キャプチャー・プログラムが自分用の作業ファイルとログ・ファイルを保管するディレクトリーです。デフォルトでは、キャプチャー・パスはキャプチャー・プログラムが始動されるディレクトリーです。キャプチャー・プログラムを Windows サービスとして始動する場合、デフォルトではキャプチャー・プログラムは `¥sqllib¥bin` ディレクトリーで始動されます。z/OS プラットフォームでは、キャプチャー

ー・プログラムは POSIX アプリケーションであるため、デフォルトのキャプチャー・パスは、次に示す方法のどちらでこのプログラムを始動するかによって決まります。

- キャプチャー・プログラムを USS コマンド行プロンプトから始動する場合。キャプチャー・パスは、このプログラムが始動されたディレクトリーです。
- キャプチャー・プログラムを開始済みのタスクまたは JCL を使用して始動する場合。デフォルト・キャプチャー・パスは、その開始済みタスクまたはジョブと関連したユーザー ID のホーム・ディレクトリーです。

キャプチャー・パスを変更して、キャプチャー・プログラムがファイルを保管する場所を指定できます。パス名 (たとえば、/home/db2inst/capture_files) を指定できます。z/OS プラットフォームでは、パス名または //CAPV8 などの高位修飾子 (HLQ) のいずれかを指定できます。HLQ を使用すると、z/OS 順次データ・セット・ファイル名のファイル命名規則に準拠する順次ファイルが作成されます。順次データ・セットは、プログラムを実行中のユーザー ID と関連付けられています。それ以外の場合は、これらのファイル名は、明示的に名前を指定されたディレクトリー・パスに保管されるファイル名 (HLQ がファイル名の最初の部分として連結されているもの) に類似します。たとえば、sysadm.CAPV8.*filename* のようになります。

capture_schema (UNIX、Windows、z/OS)

デフォルト: `capture_schema=ASN`

capture_schema パラメーターにより、どのキャプチャー・プログラムを始動するのかが識別されます。デフォルトでは、キャプチャー・スキーマは ASN です。

別のスキーマをすでにセットアップしてある場合は、**capture_schema** パラメーターを使用してそのスキーマを指定すれば、当該のキャプチャー・プログラムを始動できます。詳細については、29 ページの『キャプチャー・コントロール表のセットを複数作成する』を参照してください。

次のような場合には、複数のキャプチャー・スキーマを使用することがあります。

アプリケーションの独立性を保ってアーカイブする場合

アプリケーション A 用のキャプチャー・プログラムとアプリケーション B 用のキャプチャー・プログラムを別個に持てるように、複数のキャプチャー・スキーマを作成してください。各キャプチャー・プログラムは、それぞれ独自のコントロール表を使用します。キャプチャー・プログラムのいずれかがダウンしても、1 つのアプリケーションだけしか影響を受けません。それ以外のアプリケーションは、別のキャプチャー・プログラムからサービスを受けているために影響されません。

各アプリケーションのそれぞれ異なる要件を満たす場合

同じソース表を使用するが、データ要件がそれぞれ異なる複数のアプリケーションがある場合は、複数のキャプチャー・スキーマを作成してください。たとえば、給与計算アプリケーションは機密従業員データを必要としますが、内部

従業員登録にはこのようなデータは必要ありません。その機密情報を 1 つのキャプチャー・スキーマだけに登録して、それ以外のキャプチャー・スキーマには登録しないようにできます。同様に、一部のアプリケーションではキャプチャー・プログラムに異なる動作をさせる必要がある場合は、1 つの表を複数回登録できます。たとえば、一部のアプリケーションではキャプチャー・プログラムが更新の保管を削除と挿入の対として実行する必要があるというような場合です。

登録に関する問題を分離する場合

1 つの登録に問題がある場合は、別のキャプチャー・スキーマを作成して、作業登録をそのスキーマに移動できます。この方法により、その問題のある登録をオリジナルのスキーマ内でデバッグし、影響を受けていない登録を別のスキーマを使用して実行できます。

capture_server (UNIX、Windows、z/OS)

デフォルト (UNIX、Windows): `capture_server= DB2DBDFT` 環境変数の値 (設定されている場合)

デフォルト (z/OS): `capture_server= None`

capture_server パラメーターは、キャプチャー・コントロール・サーバーを指定します。キャプチャー・コントロール表 (登録表など) はキャプチャー・コントロール・サーバー上に配置され、それらの中にはソース表用の登録情報が含まれます。キャプチャー・プログラムは DB2 ログを読み取るため、キャプチャー・プログラムがソース・データベースと同じサーバーで実行されている必要があります。

commit_interval (UNIX、Windows、z/OS)

デフォルト: `commit_interval=30`

commit_interval パラメーターは、キャプチャー・プログラムがキャプチャー・コントロール表 (UOW 表および CD 表も含む) に対してデータをコミットする頻度を秒単位で指定します。デフォルトでは、キャプチャー・プログラムは UOW 表および CD 表にデータをコミットする前に 30 秒待ちます。コミット・インターバル内に更新された表に対して、ロックが保留されます。**commit_interval** パラメーターに比較的大きな値を指定すると、キャプチャー・プログラムの CPU 使用量が削減されますが、それと同時に、頻繁に実行されているサブスクリプション・セットの待ち時間が増える可能性があります。なぜなら、アプライ・プログラムが取り出せるのはコミット済みのデータだけだからです。

lag_limit (UNIX、Windows、z/OS)

デフォルト: `lag_limit=10,080`

lag_limit パラメーターは、キャプチャー・プログラムが DB2 ログからのレコードを処理する際に許される遅れの分数を表します。

デフォルトでは、ログ・レコードが 10,080 分 (7 日) より古い場合、キャプチャー・プログラムがコールド・スタートに切り替えるのを許可する値が **startmode** パラメーターに指定されないかぎり、キャプチャー・プログラムは始動しません。

遅れが限度に達したためにキャプチャー・プログラムが始動しない場合は、キャプチャー・プログラムがログの読み取りで遅れを出している理由を判別する必要があります。この遅れの限度に関するパラメーターを実際に使用することのないテスト環境では、遅れの限度をより高く設定して、キャプチャー・プログラムの再始動を試行する場合があります。別の方法として、使用するテスト環境のソース表にはほんの少しのデータしか入っていない場合にはキャプチャー・プログラムをコールド・スタートして、すべてのターゲット表内のデータを完全にリフレッシュすることもあります。

logreuse (UNIX、Windows、z/OS)

デフォルト: **logreuse=n**

キャプチャー・プログラムは、操作情報をログ・ファイルに保管します。

UNIX および Windows プラットフォームでは、ログ・ファイルの名前は *db2instance.capture_server.capture_schema.CAP.log* です。たとえば、DB2INST.SRCDB1.ASN.CAP.log。

z/OS プラットフォームの場合は、ファイル名はこれと類似のものですが、そのファイル名には DB2 インスタンス名が含まれていないという点だけが異なります。たとえば、SRCDB1.ASN.CAP.log。このファイルは、**capture_path** パラメーターで指定されているディレクトリーに保管されます。**capture_path** パラメーターが高位修飾子 (HLQ) として指定されている場合は、z/OS 順次データ・セット・ファイルのファイル命名規則が適用されます。したがって、ログ・ファイル名を作成するのに使用される **capture_schema** 名は、その名前の最初の 8 文字までの長さに切り捨てられます。

デフォルト (**logreuse=n**) では、キャプチャー・プログラムはメッセージをログ・ファイルに付加します。これは、キャプチャー・プログラムが再始動された後であっても同様に行われます。メッセージのヒストリーが必要な場合は、デフォルトのままにしておいてください。次のような場合には、キャプチャー・プログラムが再始動時にログを削除して再作成するようにさせる (**logreuse=y**) 必要があります。

- ログが大きくなったため、ログを消去したい場合。
- ログに保管されているヒストリーが不要な場合。
- スペースを節約したい場合。

logstdout (UNIX、Windows、z/OS)

デフォルト: **logstdout=n**

logstdout パラメーターが使用可能なのは、**asncap** コマンドを使用する場合だけです。このパラメーターは、レプリケーション・センターでは利用不能です。

デフォルトでは、キャプチャー・プログラムは一部の警告メッセージと通知メッセージをログ・ファイルにのみ送信します。トラブルシューティング中や、テスト環境でキャプチャー・プログラムの動作をモニターしているときには、このようなメッセージは標準出力 (STDOUT) に送信されるようにする (**logstdout=y**) 場合があります。

memory_limit (UNIX、Windows、z/OS)

デフォルト: **memory_limit=32**

memory_limit パラメーターは、キャプチャー・プログラムが使用できるメモリーの量を MB (M バイト) 単位で指定します。

デフォルトでは、キャプチャー・プログラムは 32 MB のメモリーを使用してトランザクション情報を保管します。その後こぼれ出た分の情報は、**capture_path** ディレクトリーに配置されているファイルに入れられます。このメモリーの限度は、パフォーマンスに関するニーズに基づいて変更できます。メモリーの限度を高めを設定すると、キャプチャーのパフォーマンスは向上しますが、システム上のその他のユーザーが使用可能なメモリーが少なくなります。メモリーの限度を低めに設定すると、その他のユーザーのためにメモリーが解放されます。メモリーの限度をあまりに低く設定して、キャプチャー・プログラムが入りきらない分をファイルに入れるようになると、システム上で使用するスペースが増え、さらに入出力のためにシステムがスローダウンすることになります。

メモリーの限度は、レプリケーション・アラート・モニターを使用してモニターできます。また、CAPMON 表内のデータを使用すると、メモリー制限のためにディスクにこぼれ出たソース・システム・トランザクションの数も判別できます。CAPMON 表の TRANS_SPILLED 列の値を合計してください。

monitor_interval (UNIX、Windows、z/OS)

デフォルト: **monitor_interval=300**

monitor_interval パラメーターは、キャプチャー・プログラムが情報をキャプチャー・モニター (IBMSNAP_CAPMON) 表に書き込む頻度を指定します。

デフォルトでは、キャプチャー・プログラムはキャプチャー・モニター表に 300 秒 (5 分) ごとに行を挿入します。この稼働パラメーターは、コミット・インターバルと関連して機能します。データを細分化されたレベルでモニターしてみたい場合は、コミット・インターバルに近いモニター・インターバルを使用してください。

monitor_limit (UNIX、Windows、z/OS)

デフォルト: **monitor_limit=10080**

monitor_limit パラメーターは、整理の対象にする前に、行がどのくらい古くなるまでモニター表に残しておく必要があるのかを指定します。

デフォルトでは、キャプチャー・モニター (IBMSNAP_CAPMON) 表内の 10,080 分 (7 日) より古い行が整理されます。IBMSNAP_CAPMON 表には、キャプチャー・プログラムに関する稼働統計が含まれています。1 週間分より少ない統計しか必要でない場合は、デフォルトのモニター限度を使用してください。統計を頻繁にモニターする場合は、1 週間分の統計を保持する必要性はないと考えられるので、モニター限度を低めに設定し、キャプチャー・モニター表がより頻繁に整理されるようにして、古い統計が除去されるようにできます。この統計をヒストリーの分析に使用する場合、および 1 週間より長い期間の統計が必要である場合は、モニター限度の値を大きくしてください。

prune_interval (UNIX、Windows、z/OS)

デフォルト: `prune_interval=300`

prune_interval パラメーターは、キャプチャー・プログラムが一部のコントロール表から古い行を削除しようとする、その試行頻度を指定します。このパラメーターが有効なのは、`autoprune=y` の場合だけです。

デフォルトでは、キャプチャー・プログラムは CD 表および UOW 表を 300 秒 (5 分) ごとに整理します。表の整理頻度が不十分であると、それらの表が含まれている表スペース用のスペースが使い尽くされる可能性があります。そうなった場合は、キャプチャー・プログラムは強制停止されます。表のプルーニングがあまりに頻繁にまたはピーク時に実行される場合は、そのようなプルーニングによって同じシステム上で実行されているアプリケーション・プログラムに支障が生じる可能性があります。オプションとしてのプルーニング頻度をレプリケーション環境で設定する場合は、実際のレプリケーション環境に最適な値を使用してください。一般に、パフォーマンスは表のサイズが小さく保たれている場合に最適となります。

プルーニング・インターバルの値を低くする前に、プルーニングが発生する可能性があるほどデータが頻繁にアプライされていることを確認してください。アプライ・プログラムがデータを頻繁にアプライしていない場合は、プルーニング・インターバルの値を低く設定しても無益です。なぜなら、CD 表と UOW 表を整理するためには、その前にアプライ・プログラムがすべてのターゲットに対してデータを複製する必要があるからです。

プルーニング・インターバルにより、キャプチャー・プログラムがどのくらいの頻度で表の整理を試行するのかが決定されます。プルーニング間隔は、データが整理の対象となるのに十分なだけ古くなったと見なす時点 を決定するパラメーター (**trace_limit**、**monitor_limit**、**retention_limit**) と関連して機能します。たとえば、**prune_interval** が 300 秒、**trace_limit** が 10080 秒である場合は、キャプチャー・プログラムは 300 秒ごとに整理を試行します。トレース表で 10080 分 (7 日) より古い行を検出すると、キャプチャー・プログラムはそれらをすべて整理します。

表のブルーニングについての詳細は、250 ページの『コントロール表のブルーニング』を参照してください。

retention_limit (UNIX、Windows、z/OS)

デフォルト: `retention_limit=10,080`

retention_limit パラメーターは、保持制限ブルーニングの対象として適格と見なすまでに、古いデータをどれだけの期間 CD 表、UOW 表、およびシグナル (IBMSNAP_SIGNAL) 表に残しておくのかを決定します。

サブスクリプション・セットが非活動にされているか、またはときたましか実行されないために、通常のブルーニング処理が使用禁止になっている場合は、データは長い間 CD 表および UOW 表に残されます。このデータが現行 DB2 タイム・スタンプから保持制限の値を引いたものより古くなると、このデータは保持制限ブルーニング処理によって表から削除されます。サブスクリプション・セットを非常にまれにしか実行しないか、またはアプライ・プログラムを停止すると、使用している CD 表と UOW 表が非常に大きくなって、保持制限ブルーニングの対象として適格となる可能性があります。

整理される行のいずれか 1 つでも レプリケーションの候補になっており、それにもかかわらずなんらかの理由でそれらがターゲット表にまだアプライされていない場合は、ターゲット表をリフレッシュして、ソースと同期化させる必要があります。比較的高い保持制限を使用すれば、フル・リフレッシュが発生するのを回避できます。ただし、使用している CD 表および UOW 表が大きくなり、システム上のスペースを使用しません。

`update-anywhere` レプリケーションを実行している場合は、保持制限ブルーニングにより、リジェクトされたトランザクションが確実に削除されます。レプリカ・ターゲット表に対して競合検出を使用している場合、競合しているトランザクションが検出されると、結果的にトランザクションが拒否されます。それらのリジェクトされたトランザクションと関係のある CD 表および UOW 表の行は複製されず、保持制限に達した時点で整理されます。リジェクトされたトランザクションと関係のあるすべての古い行が削除された場合は、フル・リフレッシュは不要です。

また、保持ブルーニングにより、もはや不要となったシグナル情報もシグナル (IBMSNAP_SIGNAL) 表から確実に削除されます。

コントロール表のブルーニングについての詳細は、250 ページの『コントロール表のブルーニング』を参照してください。

sleep_interval (UNIX、Windows、z/OS)

デフォルト: `sleep_interval=5`

スリープ・インターバルは、キャプチャー・プログラムがログの終わりに到達し、バッファが空になった後、キャプチャー・プログラムは何秒待ってから再びログを読み取

るかという、その秒数を表します。 z/OS プラットフォームでのデータ共有の場合は、スリープ・インターバルは、バッファがその全容量の半分より少ない領域しか使用されていない状態に戻った後、キャプチャー・プログラムがスリープする秒数を表します。

デフォルトでは、キャプチャー・プログラムは 5 秒間スリープします。キャプチャー・プログラムがログを読み取るために生じるオーバーヘッドを削減したい場合は、スリープ・インターバルを変更してください。スリープ・インターバルの値を小さくすることは、その分遅延が生じる機会が少なくなるということを意味します。スリープ・インターバルの値を大きくすると、まばらに更新が行われるシステムでは潜在的な CPU 使用量が節約されます。

startmode (UNIX、Windows、z/OS)

デフォルト: `startmode=warmsi`

キャプチャーは、以下の始動モードのいずれかを使用して始動できます。

warmsi (ウォーム・スタート、初回はコールド・スタートに切り替える)

キャプチャー・プログラムはウォーム・スタートします。ただし、今回はじめてキャプチャー・プログラムを始動する場合は、コールド・スタートに切り替えられます。この始動モードは、コールド・スタートになるのは必ずキャプチャー・プログラムの最初の始動時だけにしたい場合に使用してください。

warmns (ウォーム・スタート、決してコールド・スタートに切り替えない)

キャプチャー・プログラムはウォーム・スタートする。ウォーム・スタートできない場合、コールド・スタートに切り替わりません。 **warmns** を日常のレプリケーション環境で使用する場合は、ウォーム・スタートの実行を妨げているあらゆる問題 (データベースや表スペースが使用不能など) を修復する機会があります。この始動モードを使用して、予期せぬコールド・スタートが実行されるのを防いでください。ウォーム・スタートした場合、キャプチャー・プログラムは終了したところから処理を再開します。キャプチャー・プログラムの始動後にエラーが発生した場合は、キャプチャー・プログラムは終了し、すべての表はそのまま残されます。

ヒント: **warmns** は、キャプチャー・プログラムをはじめて始動するときには使用できません。なぜなら、キャプチャー・プログラムの最初の始動時にはウォーム・スタート情報がないからです。キャプチャー・プログラムの最初の始動時に **cold** 始動モードを使用してから、**warmns** 始動モードを使用してください。始動モードを切り替えたくない場合は、代わりに **warmsi** を使用できます。

warmsa (ウォーム・スタート、常に必要に応じてコールド・スタートに切り替える)

ウォーム・スタート情報が使用可能な場合、キャプチャー・プログラムは前回の実行で終了したところから処理を再開します。キャプチャー・プログラムは、ウォーム・スタートできない場合、コールド・スタートに切り替わりま

す。通常は、コールド・スタートへの切り替えは望ましくありません。なぜなら、すべてのターゲット表をリフレッシュする必要があるからです。

cold コールド・スタート時には、キャプチャー・プログラムは初期化の間に CD 表および UOW 表のすべての行を削除します。これらのレプリケーション・ソースへのすべてのサブスクリプション・セットは、次のアプライ処理サイクルの間に完全にリフレッシュされます (つまり、すべてのデータがソース表からターゲット表にコピーされます)。キャプチャー・プログラムがコールド・スタートを試行したときにフル・リフレッシュが使用不可にされていた場合、キャプチャー・プログラムは始動しますが、アプライ・プログラムは失敗して、エラー・メッセージを発行します。

キャプチャー・プログラムがコールド・スタートするよう明示的に要求する必要がある場合はめったにありません。コールド・スタートが必要となるのは、キャプチャー・プログラムの最初の始動時、および **warmsi** が推奨始動モードである場合だけです。

重要: 変更データのヒストリーを正確なものにしておきたい場合は、キャプチャー・プログラムのコールド・スタートは行わないでください。キャプチャー・プログラムをシャットダウンされた後でアプライ・プログラムが変更を複製できないと、ギャップが生じることがあります。また、コールド・スタートを避けた以上、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表で STARTMODE のデフォルトとしてコールド・スタート (cold) を指定しないでください。

term (UNIX、Windows、z/OS)

デフォルト: **term=y**

term パラメーターにより、DB2 の状況がキャプチャー・プログラムの動作にどのように影響するかが決定されます。

デフォルトでは、キャプチャー・プログラムは DB2 が終了すると終了します。

term=n は、DB2 がアクティブでない場合に DB2 が始動するのをキャプチャー・プログラムが待つようにする場合に使用してください。DB2 が静止している場合は、キャプチャーは終了しません。これは、キャプチャーはアクティブなままであるが、データベースは使用しないということです。

trace_limit (UNIX、Windows、z/OS)

デフォルト: **trace_limit=10,080**

trace_limit は、整理の対象にする前に、行がどのくらい古くなるまでキャプチャー・トレース (IBMSNAP_CAPTRACE) 表に残しておく必要があるのかを指定します。

キャプチャーが整理を実行する場合、デフォルトでは、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表内の行は 10,080 分 (7 日) ごとに整理の対象として適格となります。CAPTRACE 表には、キャプチャー・プログラム用の監査証跡情報が含まれています。キャプチャーが実行することはすべてこの表に記録されます。そのため、キャプチャー・プログラムが非常にアクティブであると、この表が急速に大きくなる可能性があります。監査情報に関するニーズに合わせてトレース限度を変更してください。

キャプチャー・プログラムの始動 (OS/400)

キャプチャー・プログラムを始動して、ジャーナルからのデータのキャプチャーを開始してください。

キャプチャー・プログラムの始動後、キャプチャー・プログラムがデータのキャプチャーをすぐには開始しないことがあります。このプログラムがデータのキャプチャーを開始するのは、アプライ・プログラムがキャプチャー・プログラムに対し、所定のソース表用のログからの変更のキャプチャーを開始するよう求めるシグナルを送った後だけです。

前提条件:

キャプチャー・プログラムを始動する前に、17 ページの『第 2 章 レプリケーションの設定』に示されている指示に従って、以下の前提条件が満たされていることを必ず確認してください。

- 正しい許可を受けている。
- 適切なキャプチャー・スキーマ用のコントロール表が作成されており、登録が定義済みである。
- レプリケーション・プログラムが構成済みである (キャプチャー・プログラムがリモート・ジャーナルを読み取っている場合)。

プロシージャ:

以下の方法のいずれかを使用して、OS/400 上でキャプチャー・プログラムを始動してください。

レプリケーション・センター

「キャプチャーの開始」ウィンドウを使用して、レプリケーション・センターのオブジェクト・ツリー内において選択されているキャプチャー・コントロール・サーバーのキャプチャー・スキーマによって識別されるキャプチャー・プログラムを実行します。詳細については、レプリケーション・センターのヘルプを参照してください。

STRDPRCAP システム・コマンド (OS/400)

コマンド構文およびパラメーター記述の詳細については、446 ページの『STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)』を参照してください。

実行中のキャプチャー・プログラムの動作の変更

キャプチャー・プログラムの実行中に、1 つまたは複数の稼働パラメーターの値をオーバーライドすると、このプログラムの動作を変更できます。この変更は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表には書き込まれません。キャプチャー・プログラムは、ユーザーがキャプチャー・プログラムを停止するまで、またはさらに新しい値を指定するまで、その新しい値を使用します。

UNIX、Windows、および z/OS では、キャプチャー・プログラムの実行中に以下のキャプチャー・パラメーターを変更できます。

- Autoprune
- Autostop
- Commit_interval
- Lag_limit
- Logreuse
- Logstdout
- Memory_limit
- Monitor_interval
- Monitor_limit
- Prune_interval
- Retention_limit
- Sleep_interval
- Term
- Trace_limit

OS/400 では、所定のキャプチャー・スキーマについて、以下の稼働パラメーターの値をオーバーライドできます。

- CLNUPITV
- FRCFRQ
- MEMLMT
- MONLMT
- MONITV
- PRUNE
- RETAIN
- TRCLMT

値を変更した場合、その影響がすべてのパラメーターに即時に及ぼされるとはかぎらない場合があります。

前提条件:

特定のキャプチャー・スキーマを指定されたキャプチャー・プログラムが始動されている必要があります。

プロシージャー:

以下の方法のいずれかを使用して、パラメーターの現行値を調べ、それらの値を現行セッション用に変更してください。

レプリケーション・センター

レプリケーション・センターで、キャプチャー・プログラムの実行中に「稼働パラメーターの変更」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

chgparms システム・コマンド (UNIX、Windows、z/OS)

340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照。

OVRDPRCAPA システム・コマンド (OS/400)

426 ページの『OVRDPRCAPA: DPR キャプチャー属性のオーバーライド (OS/400)』を参照。

キャプチャー・パラメーター表の稼働パラメーターの変更

キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表には、キャプチャー・プログラム用の稼働パラメーターが含まれています。始動時に、キャプチャー・プログラムはこの表から、デフォルトに設定されている稼働中の動作に関する値を使用します。ただし、始動パラメーターを使用して新しい値が指定された場合を除きます。

キャプチャー・パラメーター表には 1 つの行しか入れられません。デフォルト値のいずれかを変更する場合は、行を挿入する代わりに列を更新することができます。この行を削除しても、出荷時のデフォルトが始動パラメーターによってオーバーライドされないかぎり、キャプチャー・プログラムはそれらのデフォルトを使用してやはり始動します。

キャプチャー・プログラムはこの表を始動時にしか読み取らないため、新しい設定でキャプチャー・プログラムを実行したい場合は、キャプチャー・プログラムの停止と始動を行う必要があります。キャプチャー・プログラムの実行中にキャプチャー・パラメーター表を変更し、さらにキャプチャー・プログラムを再初期化しても、キャプチャー・プログラムの動作は変更されません。この表の列についての記述は、481 ページの『第 23 章 表構造』を参照してください。

プロシージャー:

以下の方法のいずれかを使用して、グローバル稼働パラメーターを変更してください。これらのパラメーターは、キャプチャー・プログラムによって使用されるもので、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表に保管されています。

レプリケーション・センター

レプリケーション・センターで「キャプチャー・パラメーターの管理」ウィンドウを使用して、キャプチャー・パラメーター表内の任意の値を表示または変更します。詳細については、レプリケーション・センターのヘルプを参照してください。

CHGDPRCAPA システム・コマンド (OS/400)

404 ページの『CHGDPRCAPA: DPR キャプチャー属性の変更 (OS/400)』を参照。

キャプチャー・プログラムの停止と始動を行った後でしか、パラメーターの変更は有効になりません。

キャプチャー・プログラムの停止

特定のキャプチャー・スキーマ用のキャプチャー・プログラムを停止できます。キャプチャー・プログラムを停止すると、それ以降当該のソースからはデータが取り込まれなくなります。

OS/400: キャプチャー・プログラムが停止した時点において開かれた状態にあった UOW 表とすべての CD 表を再編成することにした場合、キャプチャー・プログラムがシャットダウンするのに時間が必要です (このプログラムは即時にはシャットダウンしません)。

前提条件:

特定のキャプチャー・スキーマを指定されたキャプチャー・プログラムが始動されている必要があります。

プロシージャー:

以下の方法のいずれかを使用して、特定のキャプチャー・スキーマ用のキャプチャー・プログラムを停止してください。

レプリケーション・センター

レプリケーション・センターで「キャプチャーの停止」ウィンドウを使用して、選択したキャプチャー・スキーマ用の実行中のキャプチャー・プログラムを停止します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnccmd stop システム・コマンド (UNIX、Windows、z/OS)

340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照。

ENDDPRCAP システム・コマンド (OS/400)

413 ページの『ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』を参照。

プルーニング中にキャプチャー・プログラムを停止または中断した場合は、プルーニングも中断されます。キャプチャー・プログラムを再開または再始動すると、プルーニングは **autoprune** パラメーターに基づいて再開されます。

登録をドロップするためにキャプチャー・プログラムを停止する必要はありません。登録をドロップするときには、必ずその前にその登録を非活動化してください。詳細については、204 ページの『登録済みオブジェクトの変更のキャプチャーの停止』を参照してください。

キャプチャーの延期 (UNIX、Windows、z/OS)

キャプチャー・プログラムを中断すると、キャプチャー・プログラム環境を損なうことなく、ピーク時にオペレーティング・システムのリソースを操作可能なトランザクションのために解放できます。進行中の作業の終了後にキャプチャー・プログラムがシャットダウンしないようにする場合は、キャプチャー・プログラムを停止するのではなく、中断してください。キャプチャーを再開する場合には、キャプチャーを再始動する場合に必然的に生じるオーバーヘッドが生じません。

重要: レプリケーション・ソースを除去する前に、キャプチャー・プログラムを中断しないでください。その代わりに、レプリケーション・ソースを非活動化してから除去してください。

前提条件:

特定のキャプチャー・スキーマを指定されたキャプチャー・プログラムが始動されている必要があります。

プロシージャ:

以下の方法のいずれかを使用して、キャプチャー・プログラムを実行中に中断してください。

レプリケーション・センター

レプリケーション・センターで「キャプチャー操作」ウィンドウを使用して、キャプチャー・プログラムを中断します。詳細については、レプリケーション・センターのヘルプを参照してください。

asncmd suspend システム・コマンド

323 ページの『第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)』を参照。

プルーニング中にキャプチャー・プログラムを停止または中断した場合は、プルーニングも中断されます。キャプチャー・プログラムを再開または再始動すると、プルーニングは **autoprun** パラメーターに基づいて再開されます。

キャプチャーの再開 (UNIX、Windows、z/OS)

中断されたキャプチャー・プログラムに再びデータのキャプチャーを開始させる場合は、このプログラムを再始動する必要があります。

前提条件:

特定のキャプチャー・スキーマを指定されたキャプチャー・プログラムが中断されている必要があります。

プロシージャー:

以下の方法のいずれかを使用して、キャプチャー・プログラムを再開してください (中断されている場合)。

レプリケーション・センター

レプリケーション・センターで「キャプチャー操作」ウィンドウを使用して、中断されているキャプチャー・プログラムを再開します。詳細については、レプリケーション・センターのヘルプを参照してください。

asncmd resume システム・コマンド

323 ページの『第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)』を参照。

プルーニング中にキャプチャー・プログラムを停止または中断した場合は、プルーニングも中断されます。キャプチャー・プログラムを再開または再始動すると、プルーニングは **autoprun** パラメーターに基づいて再開されます。

キャプチャーの再初期化

キャプチャー・プログラムの実行中に既存の登録済みオブジェクトの属性を変更する場合は、必ずキャプチャー・プログラムを再初期化してください。たとえば、登録 (IBMSNAP_REGISTER) 表内の値 CONFLICT_LEVEL、CHGONLY、RECAPTURE、および CHG_UPD_TO_DEL_INS などを変更する場合は。

OS/400 でのキャプチャーの場合は、以前取り込まれていなかったジャーナルを対象としてデータのキャプチャーを開始する場合にも再初期化が必要です。

前提条件:

特定のキャプチャー・スキーマを指定されたキャプチャー・プログラムが始動されている必要があります。

プロシージャー:

以下の方法のいずれかを使用して、キャプチャー・プログラムを実行中に再初期化してください。

レプリケーション・センター

レプリケーション・センターで「キャプチャー操作」ウィンドウを使用して、キャプチャー・プログラムを再初期化します。詳細については、レプリケーション・センターのヘルプを参照してください。

asncmd reinit システム・コマンド

323 ページの『第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)』を参照。

INZDPRCAP システム・コマンド

424 ページの『INZDPRCAP: DPR キャプチャー・プログラムの再初期化 (OS/400)』を参照。

関連作業:

- 461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』
- 467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』

関連資料:

- 355 ページの『asnscrt: キャプチャー、アプライ、またはレプリケーション・アラート・モニターを開始する DB2 レプリケーション・サービスの作成 (Windows のみ)』
- 340 ページの『asncmd: キャプチャーの操作 (UNIX、Windows、z/OS)』
- 334 ページの『asncap: キャプチャーの開始 (UNIX、Windows、z/OS)』
- 413 ページの『ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』
- 446 ページの『STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)』

第 10 章 アプライ・プログラムの操作

この章では、アプライ・プログラムの始動および停止の方法を説明します。また、ASNDONE および ASNLOAD 出口ルーチンの使用方法も説明します。

この章は、以下のセクションを含んでいます。

- 『アプライ・プログラムの始動 (UNIX、Windows、z/OS)』
- 164 ページの『アプライ・プログラムの始動 (OS/400)』
- 165 ページの『アプライ・プログラムの停止』
- 166 ページの『ASNDONE 出口ルーチンの変更 (UNIX、Windows、z/OS)』
- 167 ページの『ASNDONE 出口ルーチンの変更 (OS/400)』
- 169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』

アプライ・プログラムの始動 (UNIX、Windows、z/OS)

アプライ・プログラムのインスタンスを始動して、ターゲットへのデータのアプライを開始できます。

アプライ・プログラムは始動された後、次のイベントのいずれかが発生するまで、連続的に実行されます (**copyonce** 始動パラメーターが使用されない場合)。

- ユーザーがレプリケーション・センターまたはコマンドを使用してアプライ・プログラムを停止する。
- アプライ・プログラムがアプライ・コントロール・サーバーに接続できない。
- アプライ・プログラムは処理用のメモリーを割り振ることができない。

アプライ・プログラムの状況を照会する方法については、177 ページの『レプリケーション・プログラムの現在の状況のチェック (UNIX、Windows、z/OS)』を参照してください。

前提条件:

アプライ・プログラムを始動する前に、17 ページの『第 2 章 レプリケーションの設定』の説明に従って、システムが正しくセットアップされていることを確認してください。

- 必要なレプリケーション・サーバーのすべてへの接続が構成されている。
- 正しい許可を受けている。
- 必要なアプライ修飾子に対してソースおよびコントロール・データを含むコントロール表が作成されている。

- レプリケーション・プログラムが構成済みである。
- z/OS の場合、必要なすべてのサーバーにアプライ・プログラムが手動でバインドされている。
- UNIX および Windows 上で実行されるリモート・サーバーに対するエンド・ユーザー認証用のパスワード・ファイルが存在している。

また、次の条件が満たされていることを確認してください。

- アプライ修飾子に対して少なくとも 1 つのアクティブなサブスクリプション・セットが存在し、サブスクリプション・セットが、1 つ、または複数の次の項目を含んでいる必要があります。
 - サブスクリプション・セット・メンバー
 - SQL ステートメント
 - プロシージャラー
- コンデンスされているターゲット表はすべてターゲット・キーを必要とします。ターゲット・キーは、アプライ・プログラムが各アプライ・サイクル中に複製する変更のトラッキングに使用する主キーまたは固有索引のいずれかである、固有列のセットです。(非コンデンス CCD 表には主キーまたは固有索引はありません。)

プロシージャラー:

次の方法のいずれかを使用して、アプライ・プログラムを始動します。

レプリケーション・センター

「アプライの開始」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnapply システム・コマンド (UNIX、Windows、z/OS)

詳細については、328 ページの『asnapply: アプライの開始 (UNIX、Windows、z/OS)』を参照してください。

Windows サービス (Windows)

詳細については、467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』を参照してください。

MVS コンソールまたは TSO (z/OS)

詳細については、461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』を参照してください。

どのプロシージャラーを使用してアプライ・プログラムを始動した場合にも始動パラメーターを設定する必要があります。次のセクションでは、始動パラメーターについて説明し、各パラメーターで選択するべき値を推奨します。

- 155 ページの『apply_path (UNIX、Windows、z/OS)』
- 156 ページの『apply_qual (UNIX、Windows、z/OS)』

- 156 ページの『control_server (UNIX、Windows、z/OS)』
- 156 ページの『copyonce (UNIX、Windows、z/OS)』
- 157 ページの『db2_subsystem (z/OS)』
- 157 ページの『delay (UNIX、Windows、z/OS)』
- 158 ページの『errwait (UNIX、Windows、z/OS)』
- 158 ページの『inamsg (UNIX、Windows、z/OS)』
- 158 ページの『loadxit (UNIX、Windows、z/OS)』
- 159 ページの『logreuse (UNIX、Windows、z/OS)』
- 159 ページの『logstdout (UNIX、Windows、z/OS)』
- 159 ページの『notify (UNIX、Windows、z/OS)』
- 160 ページの『opt4one (UNIX、Windows、z/OS)』
- 160 ページの『pwdfile (UNIX、Windows)』
- 160 ページの『sleep (UNIX、Windows、z/OS)』
- 161 ページの『spillfile (UNIX、Windows、z/OS)』
- 162 ページの『sqlerrorcontinue (UNIX、Windows)』
- 163 ページの『term (UNIX、Windows、z/OS)』
- 163 ページの『trlreuse (UNIX、Windows、z/OS)』

apply_path (UNIX、Windows、z/OS)

デフォルト (UNIX、Windows、z/OS): `apply_path=current_directory`

デフォルト (Windows NT 上のサービス): `apply_path=sqllib¥bin`

アプライ・パスは、アプライ・プログラムがログおよび作業ファイルを保管するディレクトリです。デフォルトのアプライ・パスはプログラムを始動するディレクトリです。ログおよび作業ファイルを他の場所に保管するには、アプライ・パスを変更します(たとえば、AIX システムの場合は `/home/db2inst/apply_files`)。アプライ・ログ・ファイルにアクセスするには、このディレクトリに移動する必要があるため、どのディレクトリを選択したか覚えておいてください。

重要: 選択したディレクトリに、アプライ・プログラムから使用される一時ファイルに十分なスペースがあることを確認してください。詳細については、11 ページの『アプライ・プログラム用予備ファイルのスペース所要量の計画』を参照してください。

Windows NT システムでのアプライ・プログラムのインスタンスの始動: レプリケーション・センターまたは `asnapply` コマンドを使用してアプライ・プログラムを始動する場合、もし大文字と小文字の区別以外は同じ複数のアプライ修飾子が使用されるのであれば、アプライ・パスを指定する必要があります。Windows NT システムのファイル名は大文字小文字が区別されません。たとえば、`APPLYQUAL1`、`ApplyQual1`、`applyqual1`

という 3 つのアプライ修飾子があるとして、アプライ・プログラムの各インスタンスのログ・ファイルのファイル名が競合しないように、これらのアプライ・インスタンスはそれぞれ異なる **apply_path** を使用して開始する必要があります。

apply_qual (UNIX、Windows、z/OS)

処理対象のサブスクリプション・セットのアプライ修飾子を指定する必要があります。(アプライ修飾子は、サブスクリプション・セットを作成するときに定義します。) 1 つの始動コマンドで指定できるアプライ修飾子は 1 つだけです。

重要: アプライ修飾子の大文字小文字は区別されます。入力された値が、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の APPLY_QUAL 列の値と一致する必要があります。

複数のアプライ修飾子を定義してあれば、アプライ・プログラムの別のインスタンスを始動できます。始動されたアプライ・プログラムの各インスタンスは、同じアプライ・コントロール・サーバーで表される異なるサブスクリプション・セットを処理します。たとえば、2 つのサブスクリプション・セットが定義され、それぞれのセットに固有の修飾子、APPLY1 および APPLY2 があります。アプライ・プログラムの 2 つのインスタンス (各修飾子に対して 1 つずつ) を始動できます。各インスタンスは、CNTRLSSVR と呼ばれるアプライ・コントロール・サーバーにあるコントロール表を使用します。アプライ・プログラムの各インスタンスは、各自のサブスクリプション・セットを別個に処理するため、アプライ・プログラムの 1 つのインスタンスですべてのセットを処理する場合よりもパフォーマンスが向上します。

control_server (UNIX、Windows、z/OS)

デフォルト (UNIX、Windows): DB2DBDFT 環境変数の値 (使用可能な場合)

デフォルト (z/OS): なし

アプライ・コントロール・サーバーは、サブスクリプション定義およびアプライ・コントロール表が置かれているサーバーです。1 つのアプライ修飾子に対して、コントロール・サーバーを 1 つだけ指定してください。値が指定されない場合、アプライ・プログラムはデフォルト・サーバー上で始動されます。デフォルトはオペレーティング・システムによって異なります。

アプライ・プログラムがコントロール・サーバーに接続できない場合、アプライ・プログラムは終了します。アプライ・プログラムは他のサーバーに接続できなくても終了しません。この場合には、エラー・メッセージを発行して処理を続けます。

copyonce (UNIX、Windows、z/OS)

デフォルト: **copyonce=n**

copyonce パラメーターは、アプライ・プログラムのコピー・サイクルを決定します。

copyonce=y を指定して開始されたアプライ・プログラムは、適格なサブスクリプション・セットをそれぞれ一度だけ処理した後で終了します。この場合、次の条件のいずれかが満たされた場合、サブスクリプション・セットは処理に適格となります。

- サブスクリプション・セットは相対タイミングを使用し、時間が経過し、サブスクリプション・セットはアクティブである。
- サブスクリプション・セットはイベント・ベースのタイミングを使用し、アクティブであり、イベントが発生したが、アプライ・プログラムはまだサブスクリプション・セットを処理していない。

copyonce=n を使用してアプライ・プログラムを始動する典型的な状況は、アプライ・プログラムの実行を継続し、適格なサブスクリプションの処理を続ける必要があるときです。

時々ネットワークに接続するようなダイヤルイン環境でアプライ・プログラムを実行するときには、**copyonce=n** ではなく、**copyonce=y** を使用してください。また、テスト環境でアプライ・プログラムを実行するときにも、**copyonce=y** を使用することが考えられます。

ヒント: サブスクリプション・セットが適格であり、レプリケーションできるデータがあるかぎり、アプライ・プログラムがサブスクリプション・セットの処理を何回も行うようにしたい場合は、**copyonce=y** ではなく、**sleep=n** を使用してください。

Copyonce=y は、複製すべきデータが残っている場合も、各セットを一度しか処理しません。

db2_subsystem (z/OS)

db2_subsystem パラメーターは、**control_server** が z/OS 上にある場合に、DB2 サブシステムの名前を指定します。入力する DB2 サブシステム名は最大 4 文字です。このパラメーターにはデフォルトはありません。このパラメーターは必須です。

delay (UNIX、Windows、z/OS)

デフォルト: **delay=6** 秒

delay パラメーターは、アプライ・プログラムが各アプライ・サイクルの最後に待機する秒数を設定します。

デフォルトでは、連続してレプリケーションを行う場合 (サブスクリプション・セットで **sleep=0** 分を使用する場合)、アプライ・プログラムはサブスクリプション・セットが正常に処理されてから、6 秒間待った後にサブスクリプション・セットを再試行します。複製すべきデータベース・アクティビティーがない場合は、ゼロ以外の値を使用して CPU サイクルを節約してください。待ち時間を少なくするには遅延値を小さくします。

errwait (UNIX、Windows、z/OS)

デフォルト: **errwait=300** 秒 (5 分)

errwait パラメーターは、サブスクリプション・サイクルが失敗した後に、アプライ・プログラムがサブスクリプション・セットを再試行するまでに待つ秒数を指定します。

デフォルトでは、アプライ・プログラムはサブスクリプション・サイクルが失敗した後 300 秒待機してから、サブスクリプション・セットを再試行します。テスト環境ではより小さい値を使用することが考えられます。最小値は 1 秒です。実稼働環境では、このパラメーターのデフォルトを変更する前に、トレードオフを考慮してください。

- 小さい値を使用すると、アプライ・プログラムがハード・エラーを繰り返し再試行することにより、CPU サイクルが無駄になります。たとえば、ターゲット表に問題があるときにアプライ・プログラムがサブスクリプション・セットの処理を繰り返し再試行した場合は、CPU サイクルを不必要に消費することになります。ログ・ファイルの中に、あるいはアプライ・プログラムが z/OS 上で実行されている場合はオペレーター・コンソールに、大量のメッセージが送られることもあります。
- 大きな値を使用すると、アプライ・プログラムが一時的エラー条件を再試行するまで待つ必要がある場合の待ち時間が増えることになります。たとえば、迅速に解決されるネットワーク・エラーを検出したときにもアプライ・プログラムは不必要に待つことになるため、**errwait** パラメーターに大きな値を使用すると、待ち時間が増加します。

inamsg (UNIX、Windows、z/OS)

デフォルト: **inamsg=y**

inamsg パラメーターは、アプライ・プログラムが非アクティブになったときにメッセージを発行するかどうかを指定します。

デフォルトでは、アプライ・プログラムは非アクティブになるとメッセージを発行しません。サブスクリプション・セットの処理間のアプライ・プログラムの待ち時間が長くない場合は特にそうですが、メッセージはアプライ・ログ・ファイルのスペースを多量に消費します。このため、アプライ・プログラムが非アクティブになった場合のメッセージを発行させたくないことも考えられます。これらのメッセージをオフにするには、**inamsg=n** を使用します。

loadxit (UNIX、Windows、z/OS)

デフォルト: **loadxit=n**

loadxit パラメーターは、アプライ・プログラムが ASNLOAD 出口ルーチンを使用してターゲット表をリフレッシュするかどうかを指定します。

デフォルトでは、アプライ・プログラムはターゲット表をリフレッシュするために ASNLOAD 出口ルーチンを使用しません (**loadxit=n**)。アプライ・プログラムが

ASNLOAD 出口ルーチン呼び出してターゲット表をリフレッシュするようにしたい場合は、**loadxit=y** を使用します。フル・リフレッシュ時にターゲット表に大量のデータをコピーするときには、ASNLOAD 出口を使用することを考えてみてください。アプリ・プログラムで ASNLOAD を使用するべき状況と、ASNLOAD の使用方法については、169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』を参照してください。

logreuse (UNIX、Windows、z/OS)

デフォルト: **logreuse=n**

アプリ・プログラムは、操作情報をログ・ファイルに保管します。UNIX および Windows の場合、ログ・ファイルの名前は *db2instance.control_server.apply_qualifier.APP.log* になります。z/OS オペレーティング・システムの場合のファイル名は DB2 インスタンス名が含まれないことを除いては同じです。

このパラメーターは、ログ・ファイルに追加するか、上書きするかを指定します。

デフォルトでは、アプリ・プログラムは開始されるたびに、ログ・ファイルにメッセージを追加します (**logreuse=n**)。アプリ・プログラムから発行されたメッセージのヒストリーを保存する場合は、デフォルトのままにします。次のような状況では、**logreuse=y** を使用して、アプリ・プログラムが開始時にログを削除し、再作成することができます。

- ログが大きくなったため、ログを消去してスペースを節約したい場合。
- ログに保管されているヒストリーが不要な場合。

logstdout (UNIX、Windows、z/OS)

デフォルト: **logstdout=n**

logstdout パラメーターは、**asnapply** コマンドを使用した場合にしか使用できません。このパラメーターはレプリケーション・センターからは使用できません。

logstdout パラメーターは、アプリ・プログラムが、ログ・ファイルにのみメッセージを送信するか、ログ・ファイルと標準出力にメッセージを送信するかを指定します。

デフォルトでは、アプリ・プログラムはログ・ファイルにのみメッセージを送信します。トラブルシューティングの場合や、アプリ・プログラムの稼働状態をモニターする場合は、メッセージを標準出力に送信するように選択できます (**logstdout=y**)。

notify (UNIX、Windows、z/OS)

デフォルト: **notify=n**

notify パラメーターは、アプライ・プログラムがサブスクリプションを処理した後に、ASNDONE 出口ルーチンに通知するかどうかを指定します。

デフォルトでは、アプライ・プログラムはサブスクリプションの処理が完了した後で ASNDONE 出口ルーチンに通知しません。**notify=y** が指定されると、アプライ・プログラムはサブスクリプション・サイクルが完了した後、アプライ・コントロール表の検査や、E メール・メッセージの送信など、追加処理を実行するために、ASNDONE を呼び出します。ASNDONE の詳細については、166 ページの『ASNDONE 出口ルーチンの変更 (UNIX、Windows、z/OS)』を参照してください。

opt4one (UNIX、Windows、z/OS)

デフォルト: **opt4one=n**

opt4one パラメーターは、アプライ・プログラムの処理が、1 つのサブスクリプション・セット用に最適化されるかどうかを指定します。

デフォルトでは、アプライ・プログラムは複数のサブスクリプション・セット用に最適化されています。アプライ・プログラムは各コピー・サイクルの最初にレプリケーション・コントロール表から情報を読み取ります。アプライ修飾子に対してサブスクリプション・セットが 1 つである場合は、アプライ・プログラムがメモリー・キャッシュ内にサブスクリプション・セットのメンバーおよび列の情報を入れてそれを再利用するように、**opt4one=y** を使用してアプライ・プログラムを始動してください。アプライ・プログラムを 1 つのサブスクリプション・セット用に最適化すると、アプライ・プログラムから使用される CPU が少なくなるため、スループット率が向上します。

重要: **opt4one=y** を使用した場合に、セットにメンバーを追加するか、その他の方法でセットを変更したときには、アプライ・プログラムがコントロール表の中の変更を入手できるように、アプライ・プログラムを停止してから再度開始する必要があります。

pwdfile (UNIX、Windows)

デフォルト: **pwdfile=asnpwd.aut**

データを複数のサーバーに分散する場合は、アプライ・プログラムがリモート・サーバー上のデータにアクセスできるように、暗号化されたパスワード・ファイルにユーザー ID とパスワードを保管できます。詳細については、26 ページの『レプリケーション用のユーザー ID およびパスワードの保管 (UNIX、Windows)』を参照してください。

sleep (UNIX、Windows、z/OS)

デフォルト: **sleep=y**

sleep パラメーターは、アプライ・プログラムが適格なサブスクリプション・セットを処理した後で、スリープ・モードで実行を続けるか、終了するかを指定します。

デフォルトでは、アプライ・プログラムは **sleep=y** で始動されます。アプライ・プログラムは適格なサブスクリプション・セットがあるかどうかをチェックします。適格なサブスクリプション・セットが検出されると、アプライ・プログラムはセットを処理し、別の適格なセットがないか探します。アプライ・プログラムは適格なセットが検出されると、その処理を続けます。適格なセットが検出されない場合、アプライ・プログラムはスリープ・モードで実行を続け、定期的に「ウェイクアップ」して、適格なサブスクリプション・セットがあるかどうかをチェックします。通常の場合は、長期間にわたって更新を適用するために、アプライ・プログラムをアクティブな状態で実行しておきたいため、この方法でアプライ・プログラムを始動します。

sleep=n を指定して始動されたアプライ・プログラムは、適格なサブスクリプション・セットがあるかどうかをチェックし、それを処理します。アプライ・プログラムは適格なサブスクリプション・セットが見つからなくなるまでセットの処理を続け、複製すべきデータがなくなるまで適格なセットの処理を繰り返した後、終了します。**sleep=n** を使用する典型的な状況は、適格なサブスクリプション・セットが検出されたときのみアプライ・プログラムが実行され、その後プログラムが終了される、モバイル環境またはテスト環境です。この場合は、アプライ・プログラムをスリープ・モードで待機させ、適格なセットがないかどうかをチェックするために定期的にウェイクアップさせることはしません。このような環境では、アプライ・プログラムを無期限に実行するのではなく、アプライ・プログラムをいつ実行するかをユーザーがコントロールできるようにします。

ヒント: 各サブスクリプション・セットを一度だけ処理する場合は、**sleep=n** ではなく、**copyonce=y** を使用してください。

spillfile (UNIX、Windows、z/OS)

デフォルト (UNIX、Windows): **spillfile=disk**

デフォルト (z/OS): **spillfile=MEM**

アプライ・プログラムはソース表からデータをリトリートし、アプライ・プログラムが実行されているシステム上の予備ファイルに入れます。

UNIX および Windows オペレーティング・システムの場合、**spillfile** の唯一有効な設定は **disk** です。予備ファイルは、必ず **apply_path** で指定されたロケーションのディスクに置かれるからです。

USS を含む z/OS オペレーティング・システムの場合、予備ファイルはデフォルトでメモリー内に保管されます。アプライ・プログラムが **ASNAPLDD** カード上の指定を使用して予備ファイルの割り振りを行うように、予備ファイルをディスクに保管するように指定できます。**ASNAPLDD** カードが指定されていない場合は、**VIO** が使用されます。レプリケーション・サイクルが短い場合は、レプリケーションするデータが少量で

あれば、メモリー内に予備ファイルを作成するのが妥当です。詳細については、11 ページの『アプライ・プログラム用予備ファイルのスペース所要量の計画』を参照してください。

sqlerrorcontinue (UNIX、Windows)

デフォルト: `sqlerrorcontinue=n`

sqlerrorcontinue パラメーターは、アプライ・プログラムが特定の SQL エラーに対処する方法を指定します。

デフォルトでは、アプライ・プログラムは SQL エラーを検出すると、そのサブスクリプション・セットの処理を停止し、エラー・メッセージを生成します。通常の場合、実稼働環境ではデフォルトを使用します。

テスト環境の場合は、ターゲット表にデータを挿入するときに、特定の SQL エラーが発生することが予想されます。これらのエラーはユーザーにとっては許容できる場合もありますが、エラーにより現行のサブスクリプション・サイクルは停止します。このような状況では、アプライ・プログラムがエラーを無視し、そのサイクルで複製したデータをロールバックしないように、**sqlerrorcontinue=y** を使用してアプライ・プログラムを始動できます。アプライ・プログラムはターゲット表へのデータ挿入時に SQL エラーを受け取ると、`apply_qualifier.sqs` ファイルの中の値をチェックします。一致するものがあると、エラーに関する詳細をエラー・ファイル、`apply_qualifier.err` に書き込み、処理を続行します。アプライ・プログラムは `apply_qualifier.sqs` ファイルにリストされていない SQL エラーを検出すると、セットの処理を停止し、次のセットに進みます。

sqlerrorcontinue=y オプションを使用してアプライ・プログラムを始動する前に、`apply_qualifier.sqs` ファイルを作成し、アプライ・プログラムの呼び出しに使用したディレクトリーにこのファイルを保管する必要があります。このファイルには、最大 20 の 5 バイトの値を続けてリストしてください。アプライ・プログラムの実行中にこのファイルの内容を変更したときには、アプライ・プログラムが新しい値を認識できるように、アプライ・プログラムを停止してから再度開始してください。

例: ターゲット表で次のエラーを受け取ったときにアプライ・プログラムがサブスクリプション・セットの処理を続けるようにするとします (`sqlstate/code`)。

42704/-803

重複索引違反

次の SQL 状態を含む SQL 状態ファイルを作成します。

42704

ターゲット表の更新時にこの SQL 状態が戻された場合、アプライ・プログラムはセット内のその他のターゲット表に変更を適用し、エラーおよび拒否された行の両方を示すエラー・ファイルを作成します。

ヒント: アプライ・トレール (IBMSNAP_APPLYTRAIL) 表の STATUS 列をチェックします。16 という値は、アプライ・プログラムがサブスクリプション・セットを正常に処理したが、*apply_qualifier.sqs* ファイルで定義された、許容されるなんらかのエラーが発生したことを意味します。

term (UNIX、Windows、z/OS)

デフォルト: **term=y**

term パラメーターは、DB2 の状況が、アプライ・プログラムの稼働にどのような影響を与えるかを決定します。

デフォルトでは、アプライ・プログラムは DB2 が終了すると終了します。

DB2 がアクティブでないときに、アプライ・プログラムが DB2 の始動を待つようにするには、**term=n** を使用します。z/OS プラットフォームの場合、DB2 が静止したときにアプライ・プログラムがアクティブであると、アプライ・プログラムはアクティブなまま残り、DB2 が開始されるまで再接続しません。UNIX および Windows プラットフォームの場合、DB2 が静止したときにアプライ・プログラムがアクティブであると、アプライ・プログラムはアクティブなまま残り、DB2 が静止モードを終了するまで再接続しません。

trlreuse (UNIX、Windows、z/OS)

デフォルト: **trlreuse=n**

trlreuse パラメーターは、アプライ・プログラムの始動時に、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表を再利用する (追加する) か、上書きするかを指定します。

デフォルトでは、アプライ・プログラムは開始されると、アプライ・トレール表にエントリーを追加します。この表は、アプライ・コントロール・サーバーにあるすべてのアプライ・インスタンスの操作の履歴を保持します。この表は、診断およびパフォーマンス統計のリポジトリです。更新の履歴を保持する場合は、デフォルトを使用します。次のような状況では、アプライ・プログラムの始動時に、アプライ・トレール表に追加するのではなく、表を空にすることができます (**trlreuse=y**)。

- アプライ・トレール表が大きくなったため、消去してスペースを節約したい場合。
- 表に保管されている履歴が不必要な場合。

ヒント: **trlreuse=y** を使用する代わりに、アプライ・プログラムがサブスクリプション・セットを正常に処理した後で (**status=0**)、SQL 処理を使用してアプライ・トレール表から行を削除できます。

アプライ・プログラムの始動 (OS/400)

アプライ・プログラムのインスタンスを始動して、ターゲットへのデータのアプライを開始できます。

アプライ・プログラムは始動された後、次の条件のいずれか発生するまで、連続的に実行されます。

- ユーザーが COPYONCE(*YES) 始動パラメーターを使用してプログラムを始動する。
- ユーザーが ALWINACT(*NO) を指定し、処理すべきデータがない。
- ユーザーがレプリケーション・センターまたはコマンドを使用してアプライ・プログラムを停止する。
- アプライ・プログラムがアプライ・コントロール・サーバーに接続できない。
- アプライ・プログラムは処理用のメモリーを割り振ることができない。

前提条件:

アプライ・プログラムを始動する前に、システムが正しくセットアップされていることを確認してください。

- すべてのレプリケーション・サーバーへの接続が構成されている。
- 正しい許可を受けている。
- コントロール表が作成されている。
- レプリケーション・プログラムが構成済みである。

また、次の条件が満たされていることを確認してください。

- アプライ修飾子に対して少なくとも 1 つのアクティブなサブスクリプション・セットが存在し、サブスクリプション・セットが、1 つ、または複数の次の項目を含んでいる必要があります。
 - サブスクリプション・セット・メンバー
 - SQL ステートメント
 - プロシージャ
- コンデンスされているターゲット表はすべてターゲット・キーを必要とします。ターゲット・キーは、アプライ・プログラムが各アプライ・サイクル中に複製する変更のトラッキングに使用する主キーまたは固有索引のいずれかである、固有列のセットです。(非コンデンス CCD 表には主キーまたは固有索引はありません。)

プロシージャ:

次の方法のいずれかを使用して、アプライ・プログラムを始動します。

レプリケーション・センター

「アプライの開始」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

STRDPRAPY システム・コマンド

詳細については、438 ページの『STRDPRAPY: アプライ・プログラムの始動 (OS/400)』を参照してください。

アプライ・プログラムを始動するときには、稼働パラメーターにこれらのデフォルト設定を使用できます。

表 7. アプライ・プログラムのデフォルト設定 (OS/400)

| 稼働パラメーター | (*value) の記述 |
|-----------------------|--|
| USER (*CURRENT) | システムにサインオンしたユーザー。 |
| JOBID (*LIBL/QZSNDPR) | 製品ライブラリー名 / ジョブ記述。 |
| APYQUAL (*USER) | 現行ユーザー名 (上記から)。 |
| CTLSVR (*LOCAL) | ローカル RDB サーバー名。 |
| TRACE (*NONE) | トレースを生成しない。 |
| FULLREFPGM (*NONE) | ASNLOAD 出口ルーチンを実行しない。 |
| SUBNFYPGM (*NONE) | ASNDONE 出口ルーチンを実行しない。 |
| INACTMSG (*YES) | アプライ・プログラムは非アクティブ期間を開始すると、プログラムが非アクティブになる期間を示すメッセージ ASN1044 を生成する。 |
| ALWINACT (*YES) | 処理するものがない場合はスリープする。 |
| DELAY (6) | 再度処理する前に、アプライ・サイクルの後に 6 秒待機する。 |
| RTYWAIT (300) | 失敗した操作を再始動する前に 300 秒待機する。 |
| COPYONCE (*NO) | 1 つのコピー・サイクルを完了した後で終了せずに処理を続行する。 |
| TRLREUSE (*NO) | アプライ・プログラムの始動時にアプライ・トレール (IBMSNAP_APPLYTRAIL) 表を空にしない。 |
| OPTSNGSET (*NO) | アプライ・プログラムのパフォーマンスを 1 つのサブスクリプション・セットの処理用に最適化しない。 |

構文図を含めて、これらの稼働パラメーターの詳細は、438 ページの『STRDPRAPY: アプライ・プログラムの始動 (OS/400)』を参照してください。

アプライ・プログラムの停止

ユーザーはアプライ・プログラムのインスタンスを停止できます。ユーザーがアプライ・プログラムを停止すると、アプライ・プログラムはターゲット表にデータをコピーしなくなり、次にプログラムが開始されたときに正しく開始されるように、コントロール表の中の情報を更新します。

前提条件:

アプライ・プログラムのインスタンスが開始されている必要があります。

プロシージャ:

アプライ・プログラムのインスタンスを停止するには、次の方法のいずれかを使用します。

レプリケーション・センター

「アプライの停止」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnacmd stop システム・コマンド (UNIX、Windows、z/OS)

詳細については、324 ページの『asnacmd: アプライの操作 (UNIX、Windows、z/OS)』を参照してください。

ENDDPRAPY システム・コマンド (OS/400)

詳細については、409 ページの『ENDDPRAPY: アプライ・プログラムの停止 (OS/400)』を参照してください。

ASNDONE 出口ルーチンの変更 (UNIX、Windows、z/OS)

このセクションでは、UNIX、Windows、および z/OS オペレーティング・システムでの ASNDONE 出口ルーチンのカスタマイズ方法を説明します。

notify=y パラメーターを使用してアプライ・プログラムを始動すると、アプライ・プログラムはサブスクリプションの処理が成功したかどうかに関係なく、サブスクリプションの処理が終了した後、ASNDONE 出口ルーチン呼び出します。次のリストは、レプリケーション環境で使用するために ASNDONE 出口ルーチンをどのように変更できるかの例を示しています。

- トランザクションが拒否されたことが判明した場合は、この出口ルーチンを使用して、リジェクトされたトランザクションを UOW 表で確認し、その後のアクション (たとえば、E メールをレプリケーション・オペレーターに自動的に送信する、メッセージを発行する、またはアラートを生成する) を開始します。
- この出口ルーチンを使用して、失敗したサブスクリプション・セットが訂正されるまでアプライ・プログラムが再試行を続けないように、失敗したサブスクリプション・セットを非活動化します。失敗したサブスクリプション・セットを検出するには、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表で STATUS= -1 を探すように出口ルーチンを変更します。サブスクリプション・セットを非活動化するには、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表に ACTIVATE=0 を設定するように出口ルーチンを構成します。
- データが各 サブスクリプション・セットに適用された後で出口ルーチンを使用してデータを操作します。(この代わりに、SQL ステートメントまたはストアード・プロシージャを使用して、アプライ・プログラムによる特定の サブスクリプション・セットの処理の前または後で実行される実行時処理ステートメントを定義することもできます)

プロシージャ:

サンプルの ASNDONE 出口ルーチンの変更したバージョンを使用するには、次のようにします。

1. 要件に合うように、ASNDONE ルーチンを調整します。

UNIX、Windows の場合: この出口ルーチンの変更方法については、サンプル・プログラム (¥sqllib¥samples¥repl¥asndone.smp) の PROLOG セクションを参照してください。

z/OS の場合: サンプル・プログラム SASNAJCL(ASNDONE) の PROLOG セクションを参照してください。

2. プログラムをコンパイル、リンク、バインドし、実行可能ファイルを適切なディレクトリに入れます。
3. **notify=y** パラメーターを使用してアプライ・プログラムを始動し、ASNDONE 出口ルーチン呼び出します。

ASNDONE 出口ルーチンの変更 (OS/400)

このセクションでは、OS/400 環境での ASNDONE 出口ルーチンのカスタマイズ方法を説明します。

SUBNFYPGM パラメーターを ASNDONE 出口ルーチンの名前に設定してアプライ・プログラムを始動すると、アプライ・プログラムは、サブスクリプションの処理が成功したかどうかに関係なく、サブスクリプションの処理が終了した後、ASNDONE 出口ルーチン呼び出します。次のリストは、レプリケーション環境で使用するために ASNDONE 出口ルーチンをどのように変更できるかの例を示しています。

- トランザクションが拒否されたことが判明した場合は、この出口ルーチンを使用して、リジェクトされたトランザクションを UOW 表で確認し、その後のアクション (たとえば、E メールをレプリケーション・オペレーターに自動的に送信する、メッセージを発行する、またはアラートを生成する) を開始します。
- この出口ルーチンを使用して、失敗したサブスクリプション・セットが訂正されるまでアプライ・プログラムが再試行を続けないように、失敗したサブスクリプション・セットを非活性化します。失敗したサブスクリプション・セットを検出するには、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表で STATUS= -1 を探すように出口ルーチンを変更します。サブスクリプション・セットを非活性化するには、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表に ACTIVATE=0 を設定するように出口ルーチンを構成します。
- データが各 サブスクリプション・セットに適用された後で出口ルーチンを使用してデータを操作します。(この代わりに、SQL ステートメントまたはストアード・プロシージャを使用して、アプライ・プログラムによる特定の サブスクリプション・セットの処理の前または後で実行される実行時処理ステートメントを定義することもできます。詳細については、124 ページの『ストアード・プロシージャまたは SQL ステートメントを使用したデータ拡張』を参照してください)

プロシージャ:

サンプルの ASNDONE 出口ルーチンの変更したバージョンを使用するには、次のようにします。

1. サイトの要件に合うように、ASNDONE 出口ルーチンを調整します。

次の表は、C、COBOL、および RPG 言語のこのルーチンのソース・コードがどこにあるかを示しています。

| コンパイラ言語 | ライブラリー名 | ソース・ファイル名 | メンバー名 |
|---------|---------|-----------|---------|
| C | QDP4 | QCSRC | ASNDONE |
| COBOL | QDP4 | QCBLESRC | ASNDONE |
| RPG | QDP4 | QRPGLESRC | ASNDONE |

プログラムの変更時に、以下の活動化グループに関連することを考慮してください。

新しい活動化グループを使用して実行するようにプログラムが作成されている場合:

アプライ・プログラムおよび ASNLOAD プログラムは、リレーショナル・データベースの接続およびオープン・カーソルなどの SQL リソースを共有しません。

OS/400 オペレーティング・システムにある活動化処理コードは、コントロールがアプライ・プログラムに戻される前に、ASNLOAD プログラムによって割り振られたリソースをすべて解放します。アプライ・プログラムが ASNLOAD プログラムを呼び出すたびに、追加のリソースが使用されます。

このプログラムが呼び出し側の活動化グループで実行するために作成される場合: アプライ・プログラムと SQL リソースを共有します。アプライ・プログラムへの影響が最小限で済むようプログラムを設計します。たとえば、現行のリレーショナル・データベース接続を変更すると、予期しないアプライ・プログラムの処理を引き起こす可能性があります。

このプログラムが名前付き活動化グループで実行するために作成される場合: アプライ・プログラムとの間でリソースを共有しません。名前付き活動化グループを使用すると、ASNLOAD プログラムが呼び出されるたびに、活動化グループのオーバーヘッドは回避されます。実行時のデータ構造および SQL リソースは、呼び出し間で共有されます。アプライ・プログラムが終了するまで、アプリケーションの終結処置は実行されません。したがって、サブスクリプション通知プログラムは、コントロールがアプライ・プログラムに戻される時、ソース表、ターゲット表、またはコントロール表をロックすることにより、アプライ・プログラムでロックの競合が生じることがないように設計します。

2. プログラムをコンパイル、リンク、バインドし、実行可能ファイルを適切なディレクトリに入れます。
3. アプライ・プログラムを始動し、**STRDPRAPY** コマンドのパラメーター **SUBNFYPGM** を使用して、ASNDONE プログラムの名前を指定します。たとえば、プログラムの名前が **ASNDONE_1** でライブラリー **APPLIB** にある場合には、次のコマンドを使用します。

```
SUBNFYPGM(APPLIB/ASNDONE_1)
```

ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ

デフォルトでは、アプライ・プログラムはサブスクリプション・セット内の各ターゲット表のリフレッシュを実行するときに ASNLOAD 出口ルーチンを使用しません。アプライ・プログラムはソース表に対して全選択を行い、アプライ・プログラムが実行されているサーバー上の予備ファイルにデータを持ち込み、INSERT ステートメントを使用してターゲット表にデータを取り込みます。ソース表が大きい場合には、フル・リフレッシュ時にターゲットにデータを効率的にコピーするために、代わりに ASNLOAD 出口ルーチンを使用できます。

ASNLOAD 出口ルーチンは、ソース・フォーマットとコンパイル済みフォーマットの両方で、サンプルの出口プログラムとして提供されます。サンプルの出口プログラムは DB2 プラットフォームによってそれぞれ異なり、そのプラットフォームで提供されるユーティリティー・オプションを利用します。

アプライ・プログラムが ASNLOAD 出口ルーチンを呼び出したときにエラーが発生するか、出口ルーチンがゼロ以外の戻りコードを戻した場合、アプライ・プログラムはメッセージを発行し、現行のサブスクリプション・セットの処理を停止し、次のサブスクリプション・セットを処理します。

前提条件:

ASNLOAD 出口ルーチンを使用する前に、次の前提条件が満たされていることを確認してください。

- ターゲット表の列は、ソース表の順序とデータ・タイプと一致する。
- ターゲット表は、レプリケーション・マッピングの一部である表のみを含む。

次のセクションでは、さまざまなプラットフォーム上で ASNLOAD 出口ルーチンを使用する方法を説明します。

- 『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (UNIX、Windows)』
- 171 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (z/OS)』
- 174 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (OS/400)』

ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (UNIX、Windows)

ASNLOAD 出口ルーチンは、DB2 の IMPORT ユーティリティーまたは DB2 の LOAD ユーティリティーと一緒に DB2 の EXPORT ユーティリティーを使用する、または新しい LOAD FROM CURSOR ユーティリティーを使用するなど、多数のユーティリティー・オプションを提供します。ユーザーがサンプルの出口ルーチンを呼び出す

と、デフォルトでは、ソース・サーバー、ターゲット・サーバー、およびランタイム環境に基づいて、使用されるユーティリティが選択されます。

コンパイル済みの出口ルーチンを使用したり、レプリケーション構成をカスタマイズして動作を構成したり、出口コード自体を変更することができます。レプリケーション構成は、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBER) 表の列を更新するか、サンプルの構成ファイル (asnload.ini) を更新することによりカスタマイズできます。

プロシージャ:

提供されたままの ASNLOAD ルーチンを使用する場合は、**loadxit=y** パラメーターを使用してアプライ・プログラムを始動します。

サンプルの ASNLOAD 出口ルーチンの変更したバージョンを使用するには、次のようにします。

1. サイトの要件に合うように、ASNLOAD ルーチンを調整します。この出口ルーチンの変更方法については、サンプル・プログラム (¥sqllib¥samples¥repl¥asnload.smp) の PROLOG セクションを参照してください。

重要: サンプルのソースは、asnload.ini ファイルからユーザー ID とパスワードの組み合わせを使用します。asnload.ini ファイルに特定のサーバーのユーザー ID とパスワードがない場合、または asnload.ini ファイルを使用できない場合は、`user/using` 句を使用しないで接続が行われます。

2. プログラムをコンパイル、リンク、バインドし、実行可能ファイルを適切なディレクトリに入れます。
3. ユーザーが指定したコードを使用して取り込みが行われるメンバーについては、LOADX_TYPE を 2 に設定してください。詳細については、173 ページの『ASNLOAD 出口の動作のカスタマイズ (UNIX、Windows、z/OS)』を参照してください。
4. **loadxit=y** パラメーターを使用してアプライ・プログラムを始動し、ASNLOAD 出口ルーチンを呼び出します。

ASNLOAD 出口ルーチンへの入力を構成するには、174 ページの『ASNLOAD に構成ファイルを使用』を参照してください。

ASNLOAD 出口ルーチンから生成されるファイル:

これらのファイルは、ASNLOAD 出口ルーチンを呼び出したアプライ・インスタンスの **apply_path** ディレクトリに保管されます。

- `asnload||apply_qualifier.trc`

トレースがオンの場合、このファイルはトレース情報を保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

- `asnload||apply_qualifier.msg`

このファイルは、ロード統計を含めて、一般出口障害、警告メッセージおよび情報メッセージを保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

- `asnaEXPT||apply_qualifier.msg`

このファイルは、DB2 の EXPORT ユーティリティが発行したエラー・メッセージ、警告メッセージ、または通知メッセージを保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

- `asnaIMPT||apply_qualifier.msg`

このファイルは、DB2 の IMPORT ユーティリティが発行したエラー・メッセージ、警告メッセージ、または通知メッセージを保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

- `asnaLOAD||apply_qualifier.msg`

このファイルは、DB2 の LOAD ユーティリティが発行したエラー・メッセージ、警告メッセージ、または通知メッセージを保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (z/OS)

ASNLOAD 出口ルーチンは、カーソルに基づくフェッチを行い、ソースからデータを入力し、ターゲットにデータをロードする、LOAD ユーティリティを呼び出します。ASNLOAD 出口ルーチンは LOG NO を指定して LOAD を使用し、表スペースの COPYPEND 状況をリセットします。サンプルの ASNLOAD ソース・コードを CHANGE して、ロード・オプションを CHANGE できます。ソースは、2 つのヘッダー・ファイルと 3 つの C++ プログラムから構成されています。

プロシージャ:

提供されたままの ASNLOAD ルーチンを使用する場合は、`loadxit=y` パラメーターを使用してアプライ・プログラムを始動します。

サンプルの ASNLOAD ルーチンの変更したバージョンを使用するには、次のようになります。

1. サイトの要件に合うように、ASNLOAD ルーチンを調整します。この出口ルーチンの変更方法については、サンプル・プログラム SASNAJCL(ASNLOAD) の PROLOG セクションを参照してください。
2. プログラムをコンパイル、リンク、バインドし、実行可能ファイルを適切なディレクトリに入れます。アプライ・プランに ASNLOAD パッケージを追加します。
 - a. 次の条件が満たされていることを確認してください。

- ユーティリティー・サポートを含む DB2 Universal Database for z/OS および DB2 Universal Database for OS/390 バージョン 7 以上がインストールされている。
 - DSNUTILS ストアード・プロシージャが実行されている。DSNUTILS は WLM 環境で実行する必要があります。DSNUTILS の使用方法の詳細は、『DB2 (OS/390 & z/OS) ユーティリティー・ガイド』(SC88-8773) を参照してください。
- b. connect(1) プリコンパイラー・オプションを指定して、2 つのヘッダー・ファイルと 3 つのプログラムをプリコンパイルします。プリコンパイル、コンパイル、リンク、およびバインド用のサンプルの JCL がサンプルの ASNLOAD プログラムから提供されています。
 - c. ASNLOAD 出口ルーチンをリンク・エディットします。アプライ・プログラムが APF 許可ライブラリーから実行される場合に備えて、SETCODE AC(1) を指定してコンパイルおよびリンクする必要があります。
 - d. ASNLOAD 出口ルーチンを DSNUTILS およびアプライ・パッケージとバインドします。
- サンプルの ASNLOAD は LOG NO を指定してロードを実行した後、表スペースを修理して nocopypend を設定します。表スペースのバックアップは行いません。ASNLOAD は、アプライ・プログラムのインスタンスを実行しているユーザー ID の元に一時ファイルを作成します。また、ロードに関するすべての情報を含むファイルも作成します。
3. ユーザーが指定したコードを使用して取り込みが行われるメンバーについては、loadx_type = 2 を設定します。
 4. loadxit=y パラメーターを使用してアプライ・プログラムを始動し、ASNLOAD 出口ルーチンを呼び出します。

ASNLOAD 出口ルーチンへの入力を構成するには、174 ページの『ASNLOAD に構成ファイルを使用』を参照してください。

ASNLOAD 出口ルーチンから生成されるファイル: これらのファイルは、ASNLOAD 出口ルーチンを呼び出したアプライ・インスタンスの **apply_path** ディレクトリーまたは HLQ に保管されます。

- *userid.apply_qual.LOADMSG*

このファイルは、ロード統計を含めて、障害、警告メッセージおよび情報メッセージを保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

- *userid.apply_qual.LOADTRC*

トレースがオンの場合、このファイルはトレース情報を保持します。ASNLOAD 出口ルーチンがこのファイルを作成します。ファイルが存在する場合は、情報がファイルに追加されます。

ASNLOAD 出口の動作のカスタマイズ (UNIX、Windows、z/OS)

レプリケーション構成をカスタマイズして ASNLOAD 出口ルーチンの動作を構成したり、出口コード自体をカスタマイズすることができます。レプリケーション構成は、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表の列を更新するか、構成ファイルを更新することによりカスタマイズできます。

サブスクリプション・メンバー表の使用

サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表の列を使用して、ASNLOAD 出口ルーチンの動作をカスタマイズできます。LOADX_TYPE を使用してロード・オプションを選択します。LOADX_TYPE の有効な値は次のとおりです。

NULL (デフォルト)

z/OS の場合: クロスローダー・ユーティリティーを使用します。

UNIX、Windows の場合: ASNLOAD 出口ルーチンが適切なユーティリティーを決定します (オプション 3、4、または 5)。

- 1 このメンバーについては ASNLOAD 出口ルーチンを呼び出しません。
このメンバーについて ASNLOAD 出口ルーチンを呼び出たくない場合は、LOADX_TYPE を 1 に設定します。
- 2 ユーザー自身の出口ロジックを提供します。
ユーザー自身のロジックを ASNLOAD 出口ルーチンで提供する場合は、ASNLOAD 出口ルーチンにより取り込みを行うサブスクリプション・セットのメンバーについて LOADX_TYPE を 2 に設定してください。たとえば、サンプルの出口ルーチンの中で、DB2 のエクスポート・ユーティリティーで bcp オプションを使用して bcp 可能なフォーマットで DB2 データベースからデータをエクスポートした後、Sybase のバルク・コピー・プログラム (bcp) を呼び出して Sybase のターゲット表にロードを行うためのメカニズムを提供できます。LOADX_TYPE を 2 に設定しているのに出口ロジックを提供しないと、出口は失敗します。
- 3 クロスローダー・ユーティリティーを使用します。
UNIX および Windows プラットフォームの場合、クロスローダー・ユーティリティーはターゲットにロードするデータをフェッチするために SELECT ステートメントを必要とします。このステートメントは、ロード対象のターゲット表に対してローカルである DB2 表を参照できます。また、ニックネームを使用して、リモートの表を参照できます。レプリケーション・ソースがターゲット表から見てリモートの DB2 表である場合は、ターゲット・サーバー・データベースに DB2 ソース表のニックネームを作成する必要があります。また、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表の LOADX_SRC_N_OWNER および LOADX_SRC_N_TABLE 列にニックネーム所有者および表を指定する必要があります。DB2 以外のレプリケーション・ソースを使用する場合は、レプリケーション・コントロール表にユーティリティー

ーで使用できるニックネームがすでに含まれているため、
LOADX_SRC_N_OWNER および LOADX_SRC_N_TABLE に情報を追加する
必要はありません。

4 (UNIX および Windows のみ)

EXPORT ユーティリティと LOAD ユーティリティの組み合わせを使用し
ます。

5 (UNIX および Windows のみ)

EXPORT ユーティリティと IMPORT ユーティリティの組み合わせを使用
します。

ASNLOAD に構成ファイルを使用

ASNLOAD 出口ルーチンへの入力を構成するために、オプションの構成ファイルを使用
できます。このファイルは ASNLOAD の実行に必須ではありません。

UNIX および Windows プラットフォームの場合、構成ファイルのファイル名は
asnload.ini にする必要があります。ASNLOAD 出口ルーチンは、**apply_path** ディレク
トリーでこのオプションの構成ファイルを探します。サンプル・ファイル
sqllib/samples/repl/asnload.ini を編集し、ASNLOAD 出口ルーチンを呼び出したアプラ
イ・プログラムのインスタンスの **apply_path** ディレクトリーに保管してください。

z/OS プラットフォームの場合、構成ファイルは *apply_qualifer.ASNLOAD.INI* という名
前のカタログ順次ファイルにする必要があります。ASNLOAD 出口ルーチンへの入力を
構成するには、このオプション・ファイルを作成するために提供されているサンプルの
JCL を使用してください。

ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ (OS/400)

アプライ・プログラムの代わりに出口ルーチンを使用して、もっと効率よくフル・リフ
レッシュを行います。たとえば、ソース表の各列および各行をターゲット表にコピーし
ている場合、分散データ管理 (DDM) ファイルおよびファイルのコピー (CPYF) CL コ
マンドを使用する出口ルーチンのフル・リフレッシュを、ソース表からターゲット表へ
とファイル全体をコピーするように設計することができます。

プロシージャー:

ASNLOAD 出口ルーチンを使用してターゲット表をリフレッシュするには、
FULLREFPGM パラメーターを使用してアプライ・プログラムを始動します。

サンプルの ASNLOAD ルーチンの変更したバージョンを使用するには、次のようにし
ます。

1. サイトの要件に合うように、ASNLOAD 出口ルーチンを調整します。この出口ルー
チンの変更方法については、サンプル・プログラムの PROLOG セクションを参照し
てください。C、COBOL、および RPG 言語のソースを使用することができます。

| コンパイラ言語 | ライブラリー名 | ソース・ファイル名 | メンバー名 |
|---------|---------|-----------|---------|
| C | QDP4 | QCSRC | ASNLOAD |
| COBOL | QDP4 | QCBLLSRC | ASNLOAD |
| RPG | QDP4 | QRPGLESRC | ASNLOAD |

2. プログラムをコンパイル、リンク、バインドし、実行可能ファイルを適切なディレクトリに入れます。

アプライ・プログラムを妨害しないようにするため、出口ルーチンが（呼び出し側の活動化グループではなく）新しい活動化グループを使うようにコンパイルします。

名前付き活動化グループまたは新しい活動化グループを使って出口ルーチンをコンパイルすることができます。パフォーマンスを向上させるには、名前付き活動化グループを使用します。名前付き活動化グループを使用すると、出口ルーチンは必要に応じて変更をコミットまたはロールバックする必要があります。アプライ・プログラムは、（終了するまで）変更をコミットまたはロールバックすることはありません。出口ルーチンは、完了時に変更を明示的にコミットするか、または変更を暗黙的にコミットするためにコンパイルされる必要があります。出口ルーチンの完了時にコミットされていない変更はすべて、以下のいずれかが行われるまでコミットされることはありません。

- アプライ・プログラムが、同じ活動化グループを使って別の出口ルーチンを呼び出す。
- アプライ・プログラムに応じて開始したジョブが終了する。

3. FULLREFPGM パラメーターを ASNLOAD プログラムの名前に設定してアプライ・プログラムを始動します。

始動されたアプライ・プログラムは、ユーザーから指定された ASNLOAD 出口ルーチンを使用します。別の ASNLOAD 出口ルーチンを使用させたい場合は、アプライ・プログラムを終了してから再始動します。

ASNLOAD 出口ルーチンを実行すると、すべてのターゲット表が 1 つずつリフレッシュされます。

関連作業:

- 461 ページの 『第 19 章 レプリケーション・プログラムの操作 (z/OS)』
- 467 ページの 『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』

関連資料:

- 355 ページの 『asnscrt: キャプチャー、アプライ、またはレプリケーション・アラート・モニターを開始する DB2 レプリケーション・サービスの作成 (Windows のみ)』
- 324 ページの 『asnacmd: アプライの操作 (UNIX、Windows、z/OS)』
- 328 ページの 『asnapply: アプライの開始 (UNIX、Windows、z/OS)』
- 409 ページの 『ENDDPRAPY: アプライ・プログラムの停止 (OS/400)』

- 438 ページの 『STRDPRAPY: アプライ・プログラムの始動 (OS/400)』

第 11 章 レプリケーションのモニター

この章では、レプリケーション環境をモニターするために使用できる方式について説明します。この章の情報を使用して、レプリケーション・プログラムの現在の状況をチェックしたり、履歴・データを検討して最近のメッセージやスループット、または待ち時間の統計を確認できます。また、キャプチャー・プログラムおよびアプライ・プログラムとは別個に実行されるレプリケーション・アラート・モニターを使用して、レプリケーション環境をモニターすることもできます。

この章は、以下のセクションを含んでいます。

- 『レプリケーション・プログラムの現在の状況のチェック (UNIX、Windows、z/OS)』
- 179 ページの『キャプチャー・プログラムおよびアプライ・プログラムのジャーナル・ジョブの状況のチェック (OS/400)』
- 179 ページの『傾向の履歴・データの検討』
- 184 ページの『レプリケーション環境の自動モニターの設定』
- 196 ページの『キャプチャー・プログラムの進行のモニター (OS/400)』

レプリケーション・プログラムの現在の状況のチェック (UNIX、Windows、z/OS)

キャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニターの現在の状況は簡単に評価できます。

レプリケーション・プログラムの現在の状況をチェックするには、次の方法のいずれかを使用します。

レプリケーション・センター (UNIX、Windows、z/OS)

「状況の照会」ウィンドウを使用して、キャプチャー・プログラムまたはアプライ・プログラムの現在の状況をチェックします。(レプリケーション・センターを使用してレプリケーション・アラート・モニターの状況を照会することはできません。) 詳細については、レプリケーション・センターのヘルプを参照してください。

コマンド行 (UNIX、Windows、z/OS)

- キャプチャー・プログラムの **asnccmd** システム・コマンド、**status** パラメーター。詳細については、340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

- アプライ・プログラムの **asnacmd** システム・コマンド、 **status** パラメーター。詳細については、324 ページの『asnacmd: アプライの操作 (UNIX、Windows、z/OS)』を参照してください。
- レプリケーション・アラート・モニターの **asnmcmd** システム・コマンド、 **status** パラメーター。詳細については、346 ページの『asnmcmd: レプリケーション・アラート・モニターの操作 (UNIX、Windows、z/OS)』を参照してください。

プログラムの状況を照会すると、そのプログラムに関連する各スレッドの状態を記述するメッセージが戻されます。

- キャプチャー・プログラムには、管理スレッド、プルーニング・スレッド、ワーカー・スレッド、シリアライゼーション・スレッドという 4 つのスレッドがあります。
- アプライ・プログラムには、管理スレッドとワーカー・スレッドという 2 つのスレッドがあります。
- レプリケーション・アラート・モニター・プログラムには、管理スレッド、ワーカー・スレッド、およびシリアライゼーション・スレッドという 3 つのスレッドがあります。

ユーザーは受け取ったメッセージから、プログラムが正しく機能しているかどうかを判断できます。一般的にワーカー・スレッド、管理スレッド、およびプルーニング・スレッドは作動状態にあり、実行が期待されるタスクを実行しています。シリアライゼーション・スレッドは通常は待機状態にあります。このスレッドはグローバル・シグナル・ハンドラーであり、シグナルを待っています。プルーニング・スレッドは、CD 表、および次のレプリケーション・コントロール表をプルーニングします。

- 作業単位 (IBMSNAP_UOW) 表
- キャプチャー・トレース (IBMSNAP_CAPTRACE) 表
- キャプチャー・モニター (IBMSNAP_CAPMON) 表
- シグナル (IBMSNAP_SIGNAL) 表

ユーザーが受け取ったメッセージが、プログラムは実行中であることを示している場合、環境がその逆の証拠を示している場合は、さらに調査が必要です。たとえば、アプライ・プログラムの状況を照会した結果、ワーカー・スレッドが作業中であることが判明したが、データが期待どおりにターゲット表にアプライされていない場合は、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表の中で、データがアプライされない理由を EXPLAIN するメッセージがないか探することができます。プログラムの作業を阻害するシステム・リソースの問題が発生したことも考えられます。

ユーザーが受け取ったメッセージが典型的な状態を示していない場合は、179 ページの表 8 に示すような追加アクションを実行できます。

表 8. 処理スレッドの状況に関連する問題に対する推奨アクション

| 処理スレッドの状況 | 記述および推奨アクション |
|-----------|---|
| 存在 | スレッドは存在していますが開始できません。IBM ソフトウェア・サポートに連絡してください。 |
| 開始済み | CPU の不足など、システム・リソースの問題の可能性がありますので調査してください。 |
| 初期化済み | スレッドは初期化されていますが作業できません。IBM ソフトウェア・サポートに連絡してください。 |
| 休止 | この状態は、キャプチャー・プログラムのスレッドにのみ当てはまります。スレッドがこの状態にある場合は、キャプチャー・プログラムはユーザーにより中断され、ユーザーによる操作の再開を待っています。 |
| 停止 | スレッドは実行されていません。アプライ・トレール (IBMSNAP_APPLYTRAIL) 表またはキャプチャー・トレース (IBMSNAP_CAPTRACE) 表で、スレッドが停止された理由を EXPLAIN するメッセージを調べてください。たとえば、ブルーニング・スレッドが停止されたことを示すメッセージを受け取った場合は、IBMSNAP_CAPTRACE 表で理由を確認します。表が非常に大きいため、即時にブルーニングする場合は、キャプチャー・プログラムを停止後に再始動すると、ブルーニング・スレッドを開始できます。 |

キャプチャー・プログラムおよびアプライ・プログラムのジャーナル・ジョブの状況のチェック (OS/400)

DB2 for iSeries は、サブシステム・ジョブの処理 (WRKSBSJOB) システム・コマンドを使用して、キャプチャー・プログラムおよびアプライ・プログラムのジャーナル・ジョブの状況をチェックします。

1. 次のコマンドを入力します。

```
WRKSBSJOB subsystem
```

この *subsystem* はサブシステム名です。ユーザーが自身のサブシステム記述を作成していないかぎり、多くの場合、サブシステムは QZSNDPR です。

2. 実行中のジョブのリストで、目的のジョブを探します。ジャーナル・ジョブは、割り当てられているジャーナルに従って命名されています。ここでジョブが見つからない場合には、ジョブの処理 (WRKJOB) システム・コマンドまたはサブミットしたジョブの処理 (WRKSBMJOB) システム・コマンドを使用します。ジョブのジョブ・ログを探して、ジョブが正常完了していること、またはジョブが失敗した理由を確認します。

傾向のヒストリー・データの検討

最近のレプリケーション操作からのヒストリー・データを検討して、データの傾向を評価できます。一定期間にわたり認識される傾向から、安定した量のデータが複製されていることを確認したり、パフォーマンスを向上させるために調整の余地があることを認識できます。

ヒストリー・データは、アプライ・トレール (IBMSNAP_APPLYTRAIL)、アプライ・トレース (IBMSNAP_APPLYTRACE)、キャプチャー・モニター (IBMSNAP_CAPMON)、およびキャプチャー・トレース (IBMSNAP_CAPTRACE) の各コントロール表から生成できます。これらの表の整理の頻度により、生成できるレポートが異なります。トラブルシューティングやパフォーマンス評価のためにデータを検査できるように、これらの表には少なくとも 1 週間分のデータを保存しておくことをお勧めします。

表 9 は、ユーザーが表示できるヒストリー・データを示しています。

表 9. ヒストリー情報の保管場所

| | |
|---|------------------------------|
| 質問: | レプリケーション・センターで 使用するウィンドウ: |
| キャプチャー・プログラムからの最新のメッセージはどのようなものか？ | キャプチャー・メッセージ |
| 平均で、 | キャプチャー・スループット分析 |
| <ul style="list-style-type: none"> 一定期間中に CD 表で処理された行数は？ 整理される行数は？ コミットされる行数は？ キャプチャー・プログラムが使用するメモリー量は？ | |
| ソースでデータが更新されてから、キャプチャー・プログラムによって取り込まれるまでの平均時間はどれくらいか？ | キャプチャー待ち時間 |
| アプライ・プログラムからの最新のメッセージはどのようなものか？ | アプライ・レポート |
| 平均で、 | アプライ・スループット分析 |
| <ul style="list-style-type: none"> 一定期間中にターゲット表で処理された行数は？ サブスクリプション・セットの処理の経過時間は？ | |
| ソース表が更新されてから、対応するターゲット表が更新されるまでの平均の経過時間はどれくらいか？ | エンドツーエンド待ち時間 |

分析対象とするデータ量を識別するために、時間の範囲を選択できます。時間範囲の開始と終了の両方の日時を指定してから、算出結果の平均値として結果を表示するように指定します。時間インターバル (1 秒、1 分、1 時間、1 日、または 1 週間) を選択して、結果をグループ化します。たとえば、9:00 p.m. から 9:59 p.m. までのアプライ・プログラムのスループットを分析するように選択した場合、データを 1 分インターバルで表示すると、結果は 60 行で表示され、それぞれの行に、60 分の範囲の間の各 1 分間のアクティビティーの要約が表示されます。また、1 時間というインターバルを選択した場合は、結果は 1 行に表示され、指定された 1 時間という期間の平均スループットが示されます。ユーザーがインターバルを指定しない場合は、APPLYTRAIL 表のロー・データが表示されます。

レプリケーション・センターのウィンドウは、さまざまなコントロール表およびログ・ファイルに含まれた情報から結果を表示します。次のセクションでは、レプリケーション・センターで、履歴・データを使用してレプリケーション操作を評価するための方法をさらに詳しく説明します。

- 『キャプチャー・プログラムのメッセージの検討』
- 『キャプチャー・プログラムのスループットの検査』
- 182 ページの『キャプチャー・プログラムによって処理されるデータの待ち時間の表示』
- 182 ページの『アプライ・プログラムのメッセージの検討』
- 183 ページの『アプライ・プログラムのスループットの検査』
- 183 ページの『トランザクションのレプリケーションに要した平均時間の表示』

キャプチャー・プログラムのメッセージの検討

指定された一定期間内にキャプチャー・トレース (IBMSNAP_CAPTRACE) 表に挿入されたメッセージを検討するには、「キャプチャー・メッセージ」ウィンドウを使用します。IBMSNAP_CAPTRACE 表には、キャプチャー・プログラムから発行された、初期化、プルーニング、警告、およびエラーなどの重要なイベントの行が入ります。

たとえば、「キャプチャー・メッセージ」ウィンドウでは、1 週間の間にキャプチャー・プログラムから記録されたすべてのエラー・メッセージと警告メッセージを検討できます。

キャプチャー・プログラムのスループットの検査

指定された時間範囲のキャプチャー・プログラムのパフォーマンス結果を表示するには、「キャプチャー・スループット分析」ウィンドウを使用します。キャプチャー・プログラムは統計情報を定期的にキャプチャー・モニター (IBMSNAP_CAPMON) 表に記録し、プルーニング時には、プルーニング統計をキャプチャー・トレース (IBMSNAP_CAPTRACE) 表に記録します。「キャプチャー・スループット分析」ウィンドウはこれらの表の情報を使用して、4 つの異なるタスクのパフォーマンス率の計算結果を表示します。

4 つのタイプの情報の結果をすべて使用して、キャプチャー・プログラムのスループット・パフォーマンスを検査できます。結果を絶対値で表示するか平均値で表示するかを指定できます。

- ログで挿入された、またはスキップされた行数
- CD 表で整理された行数
- コミットされたトランザクション数
- メモリー使用

たとえば、「キャプチャー・スループット分析」ウィンドウでは、キャプチャー・プログラムのスループットの 1 週間の平均パフォーマンスを検討できます。これを行うには、時間範囲の開始と終了の両方の日時を指定してから、算出結果の平均値として結果を表示するように指定します。

キャプチャー・プログラムによって処理されるデータの待ち時間の表示

ソースでデータが更新されてから、キャプチャー・プログラムによって取り込まれるまでのおおよその時間を表示するには、「キャプチャー待ち時間」ウィンドウを使用します。経過時間は、一定期間にわたり、CD 表の中のデータの新鮮さを示します。この平均待ち時間は、キャプチャー・モニター (CAPMON) 表の情報から生成できます。この情報の派生元は、登録 (IBMSNAP_REGISTER) 表です。

現行のキャプチャー待ち時間は、登録 (IBMSNAP_REGISTER) 表の中のグローバル・レコードからの SYNCHTIME 列の中の CURRENT_TIMESTAMP 値を使用して計算されます。

(CURRENT_TIMESTAMP) - (SYNCHTIME)

表 10. 現行キャプチャー待ち時間を計算するための値の例

| パラメーター | 列値 |
|-------------------|---------------------|
| CURRENT_TIMESTAMP | 2001-10-20-10:30:25 |
| SYNCHTIME | 2001-10-20-10:30:00 |

たとえば、表 10 の値を使用すると、現行待ち時間は 25 秒になります。

10:30:25 - 10:30:00 = 25

キャプチャー待ち時間は時間とともに変化し、これらの変更のヒストリーはキャプチャー・モニター (IBMSNAP_CAPMON) 表に保管されます。レプリケーション・センターはキャプチャー・モニター表の中の情報を使用して、平均待ち時間、またはヒストリー待ち時間を計算します。平均待ち時間の場合の公式は、現行待ち時間のものと同じですが、CURRENT_TIMESTAMP 値ではなく、MONITOR_TIME 値が使用されます。

MONITOR_TIME 値は、キャプチャー・プログラムによってキャプチャー・モニター表に行が挿入された時間を示すタイム・スタンプです。平均待ち時間は、秒、分、時間、日、または週で表示できます。たとえば、「キャプチャー待ち時間」ウィンドウで、キャプチャー・プログラムの過去 1 週間の平均待ち時間を 1 時間単位で表示できます。

アプライ・プログラムのメッセージの検討

「アプライ・レポート」ウィンドウを使用して、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表に挿入されたデータを検討して、一定期間について、特定のアプライ・プログラムが成功しているかどうかをチェックします。

IBMSNAP_APPLYTRAIL 表には、サブスクリプション・セットの実行に関するデータとして、サブスクリプション・セットの状況、エラー・メッセージ、および処理された行数が含まれています。

「アプライ・レポート」ウィンドウでは次のデータを表示できます。

- すべてのサブスクリプション・セット
- 失敗したサブスクリプション・セット
- 成功したサブスクリプション・セット
- 失敗したサブスクリプション・セットごとのエラー・サマリー

たとえば、「アプライ・レポート」ウィンドウでは、アプライ・プログラムが先週サブスクリプション・セットを正常に処理したかどうかを確認できます。複製できなかったサブスクリプション・セットがある場合には、これらのセットに対してアプライ・プログラムから発行されたエラー・メッセージを表示できます。また、「アプライ・レポート」ウィンドウと一緒に「アプライ・スループット分析」ウィンドウを使用できます。「アプライ・レポート」ウィンドウを使用してどのセットが正常にレプリケーションされたかを確認した後、「アプライ・スループット分析」ウィンドウを使用してレプリケーションされた行数と、レプリケーションに要した時間を確認できます。

アプライ・プログラムのスループットの検査

特定のアプライ修飾子のパフォーマンス統計を検査するには「アプライ・スループット分析」ウィンドウを使用します。SQL ステートメントを作成しなくても、データをフィルターに掛け、グループ化できます。たとえば、特定のアプライ修飾子により処理されたサブスクリプション・セット内のターゲット表で挿入、更新、削除、および再処理された行数を表示できます。また、特定のアプライ修飾子について、アプライ・プログラムがサブスクリプション・セットの処理に費やした時間も表示できます。

トランザクションのレプリケーションに要した平均時間の表示

特定のサブスクリプション・セット内のトランザクションのレプリケーションに使用された平均時間のおおよその値を表示するには、「エンドツーエンド待ち時間」ウィンドウを使用します。変更キャプチャー・レプリケーションで実行されるイベントの順序の記述は、レプリケーション・センターのヘルプを参照してください。

「エンドツーエンド待ち時間」ウィンドウでは、たとえば、一定期間内のアプライ・サイクルごとに、サブスクリプション・セットのおおよその待ち時間を表示できます。また、時間をインターバルで分けて、インターバルごとの平均待ち時間を表示することもできます。

レプリケーション・センターでは次の式を使用してエンドツーエンド待ち時間を計算します。

$$(\text{ENDTIME} - \text{LASTRUN}) + (\text{SOURCE_CONN_TIME} - \text{SYNCHTIME})$$

ここで、

- ENDTIME は、アプライ・プログラムがサブスクリプション・セットの処理を終了した時間です。
- LASTRUN は、アプライ・プログラムがサブスクリプション・セットの処理を開始した時間です。
- SOURCE_CONN_TIME は、アプライ・プログラムがデータをフェッチするためにキャプチャー・コントロール・サーバーに接続した時間です。
- SYNCHTIME は、キャプチャー・プログラムによる、CD 表へのデータのコミットの最新の時刻です。

表 11. エンドツーエンド待ち時間の値の計算の例

| パラメーター | 列値 |
|------------------|---------------------|
| ENDTIME | 2001-10-20-10:01:00 |
| LASTRUN | 2001-10-20-10:00:30 |
| SOURCE_CONN_TIME | 2001-10-20-10:00:32 |
| SYNCHTIME | 2001-10-20-10:00:00 |

たとえば、あるサブスクリプション・セットに、表 11 に示すような値があるとします。前述の式を使用すると、このサブスクリプション・セットのエンドツーエンドの平均待ち時間は 62 秒になります。

$$(10:01:00 - 10:00:30) + (10:00:32 - 10:00:00) = 62$$

レプリケーション環境の自動モニターの設定

レプリケーション環境を設定し、キャプチャー・プログラムおよびアプライ・プログラムを始動したら、指定されたとおりにキャプチャー・プログラムがデータをキャプチャーし、アプライ・プログラムがデータを適用していることを確認する必要があります。いずれかのプログラムが作動していない場合には、プログラムが停止されるような状況がいつ発生したかを知る必要があります。自動化されたレプリケーション・アラート・モニターを使用して、レプリケーション環境の稼働およびパフォーマンスをチェックします。

レプリケーション・アラート・モニターは DB2 for UNIX、DB2 for Windows、または DB2 for z/OS で実行され、これらのプラットフォーム上で、また DB2 for iSeries で、データベース・サーバーをモニターできます。たとえば、ユーザー環境内の専用の NT サーバー上に自動モニターをセットアップし、この NT サーバーからレプリケーション・アラート・モニターを実行できます。モニター・プログラムはすべてのレプリケーション・サーバーに接続でき、エンタープライズ内のすべてのプラットフォーム上のすべてのキャプチャー・プログラムおよびアプライ・プログラムのアクティビティをモニターできます。情報は、モニター・コントロール・サーバー上のモニター・コントロール表に保管されます。モニター・コントロール・サーバーは、DB2 for UNIX、DB2 for Windows、または DB2 for z/OS で使用できます。

ユーザーがモニター対象のアラート条件を選択すると、レプリケーション・アラート・モニターはキャプチャー・プログラムおよびアプライ・プログラムのアクティビティをモニターできるようになります。レプリケーション・アラート・モニターは稼働中、これらのアラート条件をチェックします。検出されたアラート条件はすべてログに記録されます。レプリケーション・アラート・モニターがアラート条件を検出したときに E メールを使用して自動的に連絡先に通知するようにしたい場合は、連絡先（名前と E メール・アドレスから構成される）または連絡先のグループをセットアップできます。また、操作エラーが発生したときに E メールを送信するようにモニターをセットアップすることもできます。

レプリケーション・アラート・モニターは、連合データベース・システムでソースとして使用される、DB2 以外のリレーショナル・データベースに関連するトリガーはモニターしません。

次のセクションでは、レプリケーション・アラート・モニターを使用してレプリケーション環境の自動モニターを行う方法について説明します。

- 『モニター・コントロール表の作成』
- 186 ページの『レプリケーション・アラート・モニターの連絡先情報の定義』
- 187 ページの『レプリケーション・アラート・モニターのアラート条件の選択』
- 195 ページの『レプリケーション・アラート・モニターの開始のスケジューリング』
- 189 ページの『レプリケーション・アラート・モニターの開始』
- 195 ページの『レプリケーション・アラート・モニターの再初期化』
- 196 ページの『レプリケーション・アラート・モニターの停止』

モニター・コントロール表の作成

レプリケーション・アラート・モニターを使用してレプリケーション環境をモニターするには、最初にサーバー上にモニター・コントロール表を作成する必要があります。モニター・コントロール表が作成されたサーバーは、モニター・コントロール・サーバーと呼ばれます。DB2 for UNIX、DB2 for Windows、DB2 for z/OS、または DB2 for iSeries のサーバーはモニター・コントロール・サーバーとして使用できます。レプリケーション・アラート・モニターがモニター・コントロール・サーバー上で実行されていることを確認してください。

モニター・コントロール表をセットアップする前に、レプリケーション構成に対するモニター・ストラテジーを決定する必要があります。レプリケーション・プログラムが実行されているすべてのサーバー（DB2 for iSeries のサーバーを除く）上でレプリケーション・アラート・モニターを実行することもできますし、中央にモニター・コントロール・サーバーを定義することもできます。集中モニター・コントロール・サーバーを使用する場合は、サーバーはすべてのレプリケーション・プログラムからリモートで実行されるため、リモート接続を実行して情報を入手し、中央で情報を統合します。企業の要件を考慮して、適切なモニター構成を決定してください。たとえば、集中モニター方

式の利点は、データを統合する単純なモニター構成です。集中モニター方式には欠点もあります。アラート条件の検出および報告に時間がかかりますし、リモート・サーバーとの接続が失われた場合、モニターは問題を検出できません。

プロシージャー:

モニター・コントロール表は次の方法で作成できます。

レプリケーション・センター

「モニター・コントロール表の作成」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

モニター・コントロール表は次の方法でドロップできます。

レプリケーション・センター

「モニター・コントロール表をドロップ」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

レプリケーション・アラート・モニターの連絡先情報の定義

レプリケーション・アラート・モニターを使用してレプリケーション環境をモニターするには、アラート条件の通知先の個人またはグループに関する連絡先情報を定義する必要があります。アラート条件が発生すると、レプリケーション・アラート・モニターは E メール・アドレスを使用して、個人またはグループの連絡先に通知を送信します。ユーザーが定義した連絡先は、連絡先が定義されているモニター・コントロール・サーバーを使用する、レプリケーション・アラート・モニターのすべてのインスタンスから使用できます。このため、複数のモニター・コントロール・サーバーがある場合には、それぞれのサーバーに連絡先を定義する必要があります。

(連絡先の名前と E メール・アドレスを使用して) 連絡先を定義したら、必要に応じて連絡先をグループ化できます。連絡先のグループを作成するには、グループの名前を指定し (たとえば DB2 管理者)、そのグループに含める連絡先を選択します。

プロシージャー:

次の方法を使用して、連絡先情報を定義します。

レプリケーション・センター

「連絡先の作成」ウィンドウまたは「連絡先グループの作成」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

定義済みの連絡先または連絡先グループの情報を変更するには、次の方法を使用します。

レプリケーション・センター

「連絡先のプロパティ」ウィンドウまたは「連絡先グループのプロパティ」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

連絡先情報を別のモニター・コントロール・サーバーにコピーするには、次の方法を使用します。

レプリケーション・センター

「連絡先のコピー」および「連絡先グループ」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

これらの連絡先はレプリケーション・センターに固有です。レプリケーション・センターは、タスク・センターまたはヘルス・センターで作成された連絡先を認識しません。

レプリケーション・アラート・モニターのアラート条件の選択

レプリケーション・アラート・モニターを使用して、次の情報またはアラート条件をトラッキングできます。

キャプチャー・プログラムまたはアプライ・プログラムの状況のアラート条件

レプリケーション・アラート・モニターは、キャプチャー・プログラムまたはアプライ・プログラムが終了したときにアラート通知を送信できます。

キャプチャー・プログラムまたはアプライ・プログラムから発行されたエラー・メッセージまたは警告メッセージのアラート条件

レプリケーション・アラート・モニターは、キャプチャー・プログラムによってキャプチャー・トレース (IBMSNAP_CAPTRACE) 表に、またはアプライ・プログラムによってアプライ・トレース (IBMSNAP_APPLYTRACE) またはアプライ・トレール (IBMSNAP_APPLYTRAIL) 表にエラー・メッセージまたは警告メッセージが挿入されたときに、アラート通知を送信できます。

しきい値の超過のアラート条件

レプリケーション・アラート・モニターは、次のしきい値が超過したときに、アラート通知を送信できます。

- キャプチャー・プログラムが許された以上のメモリーを使用している場合。
- キャプチャー・プログラムの待ち時間が指定された限度を超えた場合。
- キャプチャー・プログラムのヒストリー待ち時間が指定された限度を超えた場合。
- トランザクションのエンドツーエンドの待ち時間が指定された限度を超えた場合。
- アプライ・プログラムが、許された以上の数のサブスクリプション・セット内の行を再処理した場合。

各種イベントのアラート条件

レプリケーション・アラート・モニターは、次のイベントが発生したときにアラート通知を送信できます。

- update-anywhere シナリオで競合が検出されたため、アプライ・プログラムがトランザクションを拒否した場合。
- アプライ・プログラムがサブスクリプション・セットのフル・リフレッシュを実行しようとした場合。
- サブスクリプション・セットの処理が失敗した場合。
- サブスクリプション・セットの処理が遅れた場合。
- アプライ・プログラムがサブスクリプション・セットを非アクティブにした場合。

レプリケーション・アラート・モニターは、コントロール表が DB2 レプリケーションのバージョン 8 以上のアーキテクチャー・レベルになっているキャプチャー・プログラムおよびアプライ・プログラムをモニターするためにしか使用できません。アラート条件を選択するときには、モニター修飾子を指定する必要があります。

レプリケーション・アラート・モニターは、ユーザーがアプリケーション・アラート・モニターを開始すると、ユーザーが指定したすべてのアラート条件について、ユーザーが指定した時間インターバルで、キャプチャー・プログラムおよびアプライ・プログラムのアクティビティをモニターします。レプリケーション・アラート・モニターを開始した後でアラート条件を変更することもできます。

プロシージャー:

キャプチャー・プログラムのアラート条件を選択するには、次の方法のいずれかを使用します。

レプリケーション・センター

「キャプチャー・スキーマ用のアラート条件を選択」ウィンドウまたは「プロパティ」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

不必要なアラート条件が検出されないように、ユーザーの環境と互換性のあるしきい値を選択してください。たとえば、キャプチャー・プログラムが 30 秒のコミット・インターバルで実行されている場合、キャプチャー・プログラムの待ち時間として 10 秒のしきい値は指定しないでください。これでは、アラートが頻繁に発行されてしまいます。

アプライ・プログラムまたはサブスクリプション・セットのアラート条件を選択するには、次の方法のいずれかを使用します。

レプリケーション・センター

「アプライ修飾子またはサブスクリプション・セット用のアラート条件を選択」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

不必要なアラート条件が検出されないように、ユーザーの環境と互換性のあるしきい値を選択してください。たとえば、時間ベースのスケジューリングを使用してアプライ・プログラムを実行しているときに、サブスクリプション・セットが 10 分ごとに実行されるようにスケジュールした場合は、サブスクリプション遅延のアラート条件として 8 分のしきい値は指定しないでください。これではアラートが頻繁に発行されてしまいます。10 分よりも大きな値になるようにしきい値を設定してください。

レプリケーション・アラート・モニターの開始

前提条件:

レプリケーション・アラート・モニターを開始する前に:

- モニター・コントロール表が作成されていることを確認してください。詳細については、185 ページの『モニター・コントロール表の作成』を参照してください。
- パスワード・ファイルが作成済みであり、モニター用のコントロール表にアクセスするための許可が設定されていることを確認してください。また、レプリケーション・アラート・モニターがバインドされていることも確認してください。詳細については、17 ページの『第 2 章 レプリケーションの設定』を参照してください。
- モニター対象のアラート条件が選択されていることを確認してください。詳細については、187 ページの『レプリケーション・アラート・モニターのアラート条件の選択』を参照してください。

プロシージャ:

レプリケーション・アラート・モニターを開始するには、次の方法のいずれかを使用します。

レプリケーション・センター (UNIX、Windows、z/OS)

「モニターの開始」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnmon システム・コマンド (UNIX、Windows、z/OS)

詳細については、348 ページの『asnmon: レプリケーション・アラート・モニターの開始 (UNIX、Windows、z/OS)』を参照してください。

Windows サービス (Windows)

詳細については、467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』を参照してください。

MVS コンソールまたは TSO (z/OS)

詳細については、461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』を参照してください。

レプリケーション・アラート・モニターを開始して、同時に複数のキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーをモニターできます。たとえば、2 つのレプリケーション・アラート・モニター・プログラムを始動してモニター・コントロール・サーバー間でワークロードを分散したり、主幹業務のアプリケーション用にレプリケーション・アラート・モニターの専用インスタンスを確保することができます。レプリケーション・アラート・モニターのインスタンスごとに、異なるモニター修飾子を指定する必要があります。レプリケーション・アラート・モニターは、キャプチャー・プログラムおよびアプライ・プログラムとは別個の、自身の処理スレッドで実行されます。

レプリケーション・アラート・モニターを開始するには、モニター・コントロール・サーバーおよびモニター修飾子が必要です。次のセクションでは、レプリケーション・アラート・モニターを開始するときに、その動作をコントロールする方法について説明します。

- 『レプリケーション・アラート・モニターの実行の指定』
- 191 ページの『レプリケーション・アラート・モニターからのログ・メッセージの出力の選択』
- 192 ページの『レプリケーション・アラート・モニターからのデータのプルーニング・インターバルの指定』
- 192 ページの『選択されたアラート条件の通知基準の指定』

レプリケーション・アラート・モニターの実行の指定

デフォルト:

monitor_interval=300 秒 (5 分)

runonce= n

レプリケーション・アラート・モニターは開始されると、デフォルトとして、ユーザーから選択されたアラート条件をモニターするインターバルで実行されます。レプリケーション・アラート・モニターを 1 時間おきに実行したり、その他の時間インターバルで実行したり、一度だけ実行するようにスケジュールすることさえできます。**runonce** パラメーター、または **monitor_interval** パラメーターを使用して、**asnmon** コマンドの動作を変更します。また、レプリケーション・センターを使用して、レプリケーション・アラート・モニターを開始したときに実行時間を指定することもできます。

runonce=y が指定されると、レプリケーション・アラート・モニターはユーザーから選択されたすべてのアラート条件を一度チェックします。**monitor_interval** パラメーターは無視されます。バッチ処理の中にレプリケーション・アラート・モニターの操作を含

めるときに **runonce** を使用できます。たとえば、アプライ・プログラムが完了した後で、**runonce=y** を使用して、失敗したサブスクリプション・セットがあるかどうかを確認できます。失敗したサブスクリプション・セットがある場合には、レプリケーション・アラート・モニターは連絡先の個人またはグループに通知を送信します。

デフォルトでは **monitor_interval** は 300 秒 (5 分) です。レプリケーション・アラート・モニターは 300 秒ごとに、特定のモニター修飾子について、すべてのアラート条件をチェックします。レプリケーション・アラート・モニターはアラート条件を検出すると、通知を送信します。

レプリケーション・アラート・モニターからのログ・メッセージの出力の選択

デフォルト:

logreuse= n

monitor_path= asnmon コマンドが呼び出されたディレクトリー

レプリケーション・アラート・モニターは開始されると、作業ファイルを保管するディレクトリー内にログ・ファイルを作成し、このログ・ファイルにメッセージを追加します。デフォルトでは、**monitor_path** ディレクトリーはプログラムを始動したディレクトリーです。**monitor_path** を変更できるのは、レプリケーション・アラート・モニターを開始するときだけです。稼働中に変更することはできません。レプリケーション・アラート・モニターがログ・ファイルを保管する場所として、別の **monitor_path** を指定できます。

ログ・ファイルの名前は **db2instance.monitor_cntl_server.monitor_qual.MON.log** です。たとえば、**DB2INST.MONDB1.MONQUAL.MON.log**。

デフォルトでは、レプリケーション・アラート・モニターは、レプリケーション・アラート・モニターの再始動後であっても、ログ・ファイルにメッセージを追加します。メッセージの履歴をプログラム・ログ・ファイル内に保持したい場合は、このデフォルト (**logreuse= n**) のままにします。プログラム・ログ・ファイルを拡大できるだけ十分なスペースが **monitor_path** ディレクトリーにあることを確認してください。

次のような場合には、レプリケーション・アラート・モニターが再始動時にログを削除して再作成するように指定することもできます。この場合には、パラメーター変数

logreuse=y: を使用します。

- ログが大きくなったため、ログを消去したい場合。
- ログ内に保管されている履歴が必要でない場合。
- バッチ・ファイルからレプリケーション・アラート・モニターを開始する場合に、スペースの制約があるため、ログ・ファイルを別のロケーションに移したい場合。

logstdout パラメーターは **asnmon** コマンドを使用した場合にしか使用できません。このパラメーターはレプリケーション・センターからは使用できません。 デフォルトで

は、レプリケーション・アラート・モニターはログ・ファイルにのみメッセージを送信
しません (**logstdout = n**)。トラブルシューティングを行う場合、またはキャプチャー・
プログラムまたはアプライ・プログラムの操作を緊密にモニターする場合には、レプリ
ケーション・アラート・モニターから標準出力にメッセージが送信されるように指定で
きます (**logstdout=y**)。

レプリケーション・アラート・モニターからのデータのプルーニング・イン ターバルの指定

デフォルト:

autopruney

alert_prune_limit=10080 分 (7 日)

trace_limit=10080 分 (7 日)

自動プルーニングは **autopruney** パラメーターによってコントロールされます。デフォ
ルトは **autopruney** であるため、レプリケーション・アラート・モニターは、モニタ
ー・コントロール表にコピー済みの行は、IBMSNAP_ALERTS 表から自動的に整理しま
す。

新規のモニター・サイクルの開始時にプルーニングの対象となる行がある場合、レプリ
ケーション・アラート・モニターはモニター・アラート (IBMSNAP_ALERTS) 表をプ
ルーニングします。デフォルトでは、レプリケーション・アラート・モニターは、10080
分 (7 日) よりも古い行を削除します。**alert_prune_limit** パラメーターの値を変更し
て、表から整理するデータの古さを指定すれば、表内にどのくらい古いデータを保管す
るかをコントロールできます。

レプリケーション・アラート・モニターが多数のアラート条件を記録しているか、
IBMSNAP_ALERTS 表のストレージ・スペースが小さいときには、**alert_prune_limit**
パラメーターの値を小さくできます。また、すべてのアラート・アクティビティのヒ
ストリーを保持しておくことも考えられます。この場合には、**alert_prune_limit** パ
ラメーターの値を大きくします。

モニター・トレース (IBMSNAP_MONTRACE) 表、およびモニター・トレール
(IBMSNAP_MONTRAIL) 表の行も、デフォルトで、10080 分 (7 日) 保管されます。こ
れは、**trace_limit** パラメーターのデフォルトです。**trace_limit** パラメーターで指定さ
れた値よりも古い行は整理されます。

選択されたアラート条件の通知基準の指定

デフォルト:

max_notifications_per_alert=3 特定のアラートからの通知

max_notifications_minutes=60 分

レプリケーション・アラート・モニターはユーザーから選択されたアラート条件を検出すると、それを保管します。担当者が E メールによって自動的にアラート条件の通知を受けると、通知パラメーターを設定できます。

ユーザーが受け取る通知の数は、 **max_notifications_minutes** および **max_notifications_per_alert** パラメーターに設定された値によっても異なります。

デフォルトでは、アラート条件が複数回検出された場合、レプリケーション・アラート・モニターは、60 分の間に、そのアラート条件に関して最大 3 つの通知を送信します。 **max_notifications_per_alert** パラメーターを使用して、 **max_notifications_minutes** パラメーターで指定された時間内に特定のアラート条件が発生した場合に受け取りたい通知の最大数を指定します。これらのパラメーターを使用することにより、問題が解決されるまでにアラート条件が長期間続いた場合に、繰り返し発行される通知の数を制限することができます。

通知を使用可能にするには、 **email_server** パラメーターも設定する必要があります。このパラメーターの値は、SMTP プロトコルを使用する E メール・サーバーのアドレスに設定してください。

E メール通知の内容は、指定された E メール・アドレスがポケットベル用のものかどうかによって異なります。次の例は、1 セットのアラートについて、それぞれの場合に予想される情報のタイプを示しています。ポケットベル以外の装置宛てに送信される E メールは、各アラートが発生した時刻 (特定のサーバーの)、発生した回数、および関連するメッセージを示しています。ポケットベル宛てに送信される E メールは、完全なメッセージの代わりに、アラートを起動したパラメーターの要約を含むことを除いては、同じになります。アラートが何回も発生した場合、タイム・スタンプはアラートが最後に発生した時刻を反映しています。

次の例は、送信される通知の例を示しています。

ポケットベル以外の装置宛ての E メール通知の例:

```
-----  
To:      repladmin@company.com  
From:    replmon@server.com  
Subject: Monitor: "MONQUAL" Alerts issued  
  
ASN5129I MONITOR "MONQUAL". The Replication Alert Monitor on  
server "WSDB" reports an e-mail alert  
  
2002-01-20-10.00.00    1 ASN0552E Capture : "ASN" The program  
encountered a SQL error. The server name is "CORP". The SQL  
request is "PREPARE". The table name "PROD1.INVOICESCD".  
The SQLCODE is "-204". The SQLSTATE is "42704". The SQLERRMC  
is "PROD1.INVOICESCD". The SQLERRP is "readCD"  
  
2002-01-20-10.05.00    2 ASN5152W Monitor "MONQUAL". The current  
Capture latency exceeds the threshold value. The Capture control
```

server is "CORP". The schema is "ASN". The Capture latency is "90" seconds. The threshold is "60" seconds

2002-01-20-10.05.00 4 ASN5154W Monitor "MONQUAL". The memory used by the Capture program exceeds the threshold value. The Capture control server is "CORP". The schema is "ASN". The amount of memory used is "34" megabytes. The threshold is "30" megabytes.

ポケットベル宛ての E メール通知の例:

To: repladmin@company.com
From: replmon@server.com
Subject: Monitor: "MONQUAL" Alerts issued

MONQUAL - MONDB

2002-01-20-10.00.00 ASN0552E 1 CAPTURE-ERRORS - CORP - ASN
2002-01-20-10.05.00 ASN5152W 2 CAPTURE_CLATENCY - CORP - ASN - 90 - 60
2002-01-20-10.05.00 ASN5154W 4 CAPTURE_MEMORY - CORP - ASN - 34 - 30

E メール通知のサイズがその E メールタイプの制限を超えた場合は、複数の E メール通知に分けて通知が送信されます。通常の E メール通知のサイズは 1024 文字に制限されています。ポケットベルの E メール・アドレスの場合の制限は 250 文字です。

通知は ASNMAIL 出口ルーチンによって処理されます。この出口ルーチンは次の入力を受け取ります。

```
asnmail email_server to-address subject alert-message alert-message
```

ここで、

- *email_server* は、SMTP プロトコルを使用する E メール・サーバーのアドレスです。このサーバー・アドレスは、`asnmon` コマンドの始動時に指定された `EMAIL_SERVER` パラメーターから渡されます。
- *to-address* は、通知を受ける連絡先の E メール・アドレスです。
- *subject* は通知の件名です。これは変換されたメッセージです。
- *alert-message* はアラート・メッセージを含むストリングです。

アラートを E メール通知で送信する代わりに、ユーザーのコードを提供して、問題管理システムにアラートを入力することもできます。入力パラメーターを含む `asnmail` 出口のサンプルが提供されています。このサンプル、`asnmail.c` は、`sqllib¥samples¥repl¥ディレクトリー`にあります。サンプル・プログラムを使用するための説明は、サンプル・プログラムの `PROLOG` を参照してください。

操作エラーの通知基準の指定

レプリケーション・アラート・モニターは、レプリケーション・アラート・モニターがモニター・コントロール・サーバーに接続できないなど、モニター処理で発生するすべ

でのエラーを保管します。レプリケーション・アラート・モニターの操作に関連するエラーが発生したときに通知を送信するには、**monitor_errors** パラメーターを使用します。このパラメーターの値は、通知の送信先の E メール・アドレスに設定してください。このパラメーターで E メール・アドレスが指定されていない場合は、エラーはログに記録されますが、送信はされません。

email_server パラメーターが指定されていない場合、このパラメーターは無視されません。

レプリケーション・アラート・モニターの開始のスケジュールリング

レプリケーション・アラート・モニターは、キャプチャー・プログラムまたはアプリ・プログラムの始動をスケジュールするのと同じ方法で、実行開始をスケジュールできます。詳細については、471 ページの『第 21 章 各種オペレーティング・システムでのレプリケーション・プログラムのスケジュールリング』を参照してください。

レプリケーション・アラート・モニターを連続的に実行するか、特定のサイクルで実行するかを開始時に指定するための情報は、190 ページの『レプリケーション・アラート・モニターの実行の指定』を参照してください。

レプリケーション・アラート・モニターの再初期化

前提条件:

特定のモニター修飾子を持つレプリケーション・アラート・モニターが開始されている必要があります。

レプリケーション・アラート・モニターの稼働中にユーザーによって変更された連絡先、アラート条件、およびレプリケーション・アラート・モニターのパラメーターの値をレプリケーション・アラート・モニターに入手させるかどうかを指定できます。たとえば、レプリケーション・アラート・モニターの実行中に、連絡先の新しい E メール・アドレスが追加された場合は、モニターを再初期化します。

プロシージャ:

レプリケーション・アラート・モニターを再初期化するには、次の方法のいずれかを使用します。

レプリケーション・センター (UNIX、Windows、z/OS)

レプリケーション・センターで、モニター修飾子を右マウス・ボタンでクリックし、「**モニターの再初期化**」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnmcmd システム・コマンド (Windows、UNIX、z/OS)

asnmcmd システム・コマンド、**reinit** パラメーターの詳細については、346 ページの

346 ページの『asnmcmd: レプリケーション・アラート・モニターの操作 (UNIX、Windows、z/OS)』を参照してください。

レプリケーション・アラート・モニターの停止

前提条件:

特定のモニター修飾子を持つレプリケーション・アラート・モニターが開始されている必要があります。

プロシージャー:

レプリケーション・アラート・モニターを停止するには、次の方法のいずれかを使用します。

レプリケーション・センター (UNIX、Windows、z/OS)

操作ツリーで、モニター修飾子を右マウス・ボタンでクリックし、「**モニターの停止**」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnmcmd システム・コマンド (Windows、UNIX、z/OS)

asnmcmd システム・コマンド、**stop** パラメーターの詳細については、346 ページの『asnmcmd: レプリケーション・アラート・モニターの操作 (UNIX、Windows、z/OS)』を参照してください。

Windows サービス (Windows)

詳細については、467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』を参照してください。

MVS コンソールまたは TSO (z/OS)

詳細については、461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』を参照してください。

キャプチャー・プログラムの進行のモニター (OS/400)

キャプチャー・プログラムが終了した場合は、再始動 (IBMSNAP_RESTART) 表を調べて、キャプチャー・プログラムがどこまで進行したかを確認できます。ソース表によって使用される各ジャーナルごとに 1 つの行があります。LOGMARKER 列には、正常に処理された最後のジャーナル・エントリーのタイム・スタンプが記されています。SEQNBR 列には、そのエントリーのジャーナルを入力した順番を表す数値が記されています。

キャプチャー・プログラムがまだ実行中の場合には、次のステップを使用して、進行状況を確認できます。

1. 取り込まれる各ソース表ごとに、その CD 表を開きます。

2. CD 表の最後の行の COMMITSEQ 列にある 16 進値をメモします。
3. 同じ COMMITSEQ 値を使用した列の作業単位 (IBMSNAP_UOW) 表を見ます。一致する COMMITSEQ が IBMSNAP_UOW 表に存在しない場合には、CD 表の 2 番目の行から最後の行で同じ処理を繰り返してください。一致するものが見つかるまで、CD 表全体を逆方向にも検索してみます。
4. 一致する COMMITSEQ が見つかったら、UOW 表の LOGMARKER 列にある値をメモします。これが、処理されたジャーナル・エントリーのタイム・スタンプです。その時間までソース表に加えられているすべての変更は、適用される準備ができています。
5. ジャーナルの表示 (**DSPJRN**) システム・コマンドを使用して、キャプチャー・プログラムによる処理待ちのジャーナル・エントリーの数を確認します。以下の例で示すようにして、出力ファイル (または、印刷レポート用のプリンター) に直接出力します。

```
DSPJRN FILE(JRNLIB/DJRN1)
        RCVRNG(*CURCHAIN)
        FROMTIME(timestamp)
        TOTIME(*LAST)
        JRNCDE(J F R C)
        OUTPUT(*OUTFILE)
        ENTDTALEN(1) OUTFILE(library/outfile)
```

この *timestamp* は、4 で識別されたタイム・スタンプです。

出力ファイルにあるレコード数が、キャプチャー・プログラムが処理しなければならぬ残っているジャーナル・エントリーの大体の数です。

第 12 章 レプリケーション環境の変更

この章では、レプリケーション環境の日常の変更を行うときに考慮する必要のある問題について説明します。

この章は、以下のセクションを含んでいます。

- 『新規オブジェクトの登録』
- 200 ページの『登録済みオブジェクトの登録属性の変更』
- 201 ページの『ソース表への列の追加』
- 204 ページの『登録済みオブジェクトの変更のキャプチャーの停止』
- 205 ページの『登録の再活性化』
- 207 ページの『登録の除去』
- 207 ページの『キャプチャー・スキーマの変更』
- 211 ページの『新規サブスクリプション・セットの作成』
- 212 ページの『既存のサブスクリプション・セットへの新規サブスクリプション・セット・メンバーの追加 (UNIX、Windows、z/OS)』
- 215 ページの『サブスクリプション・セットの属性の変更』
- 215 ページの『サブスクリプション・セット名の変更』
- 218 ページの『サブスクリプション・セットの分割』
- 223 ページの『サブスクリプション・セットのマージ』
- 226 ページの『サブスクリプション・セットのアプライ修飾子の変更』
- 229 ページの『サブスクリプション・セットの非活性化』
- 231 ページの『サブスクリプション・セットの除去』
- 232 ページの『データベース・アプリケーション・イベントとレプリケーション・イベントの調整』
- 239 ページの『別のシステムへのレプリケーション構成のプロモート』

新規オブジェクトの登録

レプリケーション環境では、いつでも新規の表、ビュー、またはニックネームを登録できます。キャプチャー・プログラムを再初期化する必要はありません。

プロシージャ:

オブジェクトを登録するには、次の方法のいずれかを使用します。

レプリケーション・センター

「表の登録」、「ビューの登録」、または「ニックネームの登録」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

ADDDPRREG システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、366 ページの『ADDDPRREG: DPR 登録の追加 (OS/400)』を参照してください。

新しく登録されたオブジェクトは、そのオブジェクトを参照するサブスクリプション・セットがアプライ・プログラムによって処理されたときに、キャプチャー・プログラムによって自動的に初期化されます。アプライ・プログラムは、この新規オブジェクトの変更のキャプチャーを開始するようにキャプチャー・プログラムにシグナルを送ります。オブジェクトの登録の詳細は、43 ページの『第 3 章 表およびビューをレプリケーション・ソースとして登録する』を参照してください。

登録済みオブジェクトの登録属性の変更

既存の登録済みオブジェクトの登録属性はいつでも変更できます。登録属性には次の属性が含まれます。

- CHGONLY
- CONFLICT_LEVEL
- RECAPTURE
- DISABLE_REFRESH
- CHG_UPD_TO_DEL_INS
- STOP_ON_ERROR
- BEFORE_IMG_PREFIX

注: 変更前イメージ接頭部の値は、この値が NULL のときにしか更新できません。

プロシージャ:

1. 次の方法を使用して属性を変更します。

レプリケーション・センター

「登録済み表」フォルダーで内容ペインの中の登録済み表を右マウス・ボタンでクリックし、「プロパティ」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

2. 属性を変更したら、変更が認識されるように、キャプチャー・プログラムを再初期化する必要があります。キャプチャー・プログラムを再初期化するには、次の方法のいずれかを使用します。

レプリケーション・センター

「キャプチャー・コントロール・サーバー」フォルダーで、内容ペインの中

のキャプチャー・コントロール・サーバーを右マウス・ボタンでクリックし、「キャプチャーの再初期化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnccmd システム・コマンド (Windows、UNIX、z/OS)

reinit パラメーターを使用します。パラメーター記述およびコマンド構文の詳細については、340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

INZDPRCAP システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、424 ページの『INZDPRCAP: DPR キャプチャー・プログラムの再初期化 (OS/400)』を参照してください。

ソース表への列の追加

登録済みソース表に列を追加する必要がある場合は、最初に DB2 レプリケーションがこの表をどのように使用しているかを考えてください。このソース表の中の新しい列を複製する必要がある場合には、既存のキャプチャー・プログラムおよびアプライ・プログラムが新規列を認識し、中断なしに処理を続行できるようにしてください。新規列の中のデータを複製するかどうかによって、特別の処理ステップを実行する必要があることがあります。

複製しない

新規列の中のデータを複製する必要がない場合には、特別な処理ステップを実行する必要はありません。キャプチャー・プログラムは即時に変更を認識し、実行を続けます。

複製する

これらの新規列の中のデータを複製する必要がある場合には、新規列のデータが取り込まれ、キャプチャー・プログラムおよびアプライ・プログラムがエラーなしに実行を継続できるように、次のステップを実行します。

前提条件:

このプロシージャを使用する前に、ソース表、変更データ (CD) 表、およびターゲット表の構造、そしてシステムで定義されている登録およびサブスクリプション・セットについてよく調べておいてください。

制約事項:

主キーとして相対レコード番号 (RRN) を使用する iSeries の表に列を追加する場合は、これらのステップを使用しないでください。RRN は CD 表の最後の列である必要があります。RRN を含む iSeries の表に列を追加するときには、登録を除去し、ソース表に列を追加してから、RRN が取り込まれることを指定して、新しい登録として再度この表を追加してください。iSeries の登録の

除去および追加の詳細は、431 ページの『RMVDPREG: DPR 登録の除去 (OS/400)』および 366 ページの『ADDDPRREG: DPR 登録の追加 (OS/400)』を参照してください。

DB2 以外のリレーショナル・データベース上の登録済みソースに列を追加するためにこれらのステップを使用することはできません。DB2 以外のリレーショナル・ソースの登録には、変更のキャプチャーに使用されるトリガーのセットが含まれます。これらのトリガーを変更することはできません。このため、このソース表に新規列を追加し、これらの列の中のデータを複製する必要があるときには、既存の登録済みソースをドロップしてから、再作成する必要があります。

プロシージャ:

1. 変更するソース表に対するすべてのアクティビティを静止します。
2. 次の方法のいずれかを使用して、キャプチャー・プログラムを停止します。

レプリケーション・センター

「キャプチャーの停止」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asncmd システム・コマンド (Windows、UNIX、z/OS)

stop パラメーターを使用します。パラメーター記述およびコマンド構文の詳細については、340 ページの『asncmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

ENDDPRCAP システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、413 ページの『ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』を参照してください。

ヒント: このプロシージャの間キャプチャー・プログラムをアクティブにしておく必要がある場合には、ソース表に対するアクティビティを停止した後、シグナル (IBMSNAP_SIGNAL) 表に USER シグナルを挿入します。キャプチャー・プログラムが USER シグナルを処理するまで待ちます。

キャプチャー・プログラムは USER シグナルを処理すると、関連する CD 表に対して処理を必要とするアクティビティがなくなるため、この CD 表へのアクセスを必要としなくなります。

3. 次の方法を使用して、このソース表をサブスクライブするすべてのサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の

アクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

注: この処理中にサブスクリプション・セットを非活動化したくない場合は、新規列を追加しているときに、このソース表に対して、これらのサブスクリプション・セットに関連付けられたアプライ・プログラムが実行されていないことを確認してください。この代わりに、これらのアプライ・プログラムが、前の USER シグナルに関連付けられたシグナルのログ・シーケンス番号 (LSN) までデータの処理を終了していることを確認することもできます。

このステップの方法では、表を変更できるように、CD 表に対して排他的アクセスを確保しています。

4. SQL を使用して ALTER TABLE ADD ステートメントをサブミットし、ソース表に新規列を追加します。
5. CD 表に新規列を追加するには、次の方法を使用します。

レプリケーション・センター

「登録済み表」フォルダーで内容ペインの中の登録済み表を右マウス・ボタンでクリックし、「プロパティ」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

キャプチャー・プログラムは自動的に登録を再初期化し、新規列で初めてログ・データを読み取ったときに、これらの新規列の変更をキャプチャーします。

6. SQL を使用して ALTER TABLE ADD ステートメントをサブミットし、ターゲット表に新規列を追加します。
7. 次の方法を使用して、ステップ 3 でまだ非活動化していなかった、関連するサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

絶対に必要である場合は、ここでこのソース表に対するアクティビティを再開できます。しかし、関連するサブスクリプション・セットがまだ変更されていないため、これらの新規列に対して行われた変更が失われないように、これらのサブスクリプション・セットは非活動化しておく必要があります。

8. 次の方法を使用して、関連するサブスクリプション・セット・メンバーに新規列を追加します。

レプリケーション・センター

「ターゲット表への列の追加」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

9. **UNIX、Windows、z/OS の場合: opt4one** を *y* に設定してアプライ・プログラムを実行している場合には、アプライ・プログラムを停止してから再始動します。
10. 次の方法を使用してサブスクリプション・セットを再活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の非活動化されたサブスクリプション・セットを右マウス・ボタンでクリックし、「活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

登録済みオブジェクトの変更のキャプチャーの停止

キャプチャー・プログラムがオブジェクトに必要なすべての処理を終了できるように、登録済みオブジェクトを削除する前に、オブジェクトを非活動化する必要があります。また、このオブジェクトで一時的に変更のキャプチャーを停止しても、他の登録済みオブジェクトに対してはキャプチャー・プログラムを実行し続けておきたい場合も、登録済みオブジェクトを非活動化できます。

キャプチャー・プログラムは、非活動化されたソース・オブジェクトについては変更のキャプチャーを停止します。しかし、これらのソース・オブジェクトに関連する変更データ (CD) 表、登録属性、およびサブスクリプション・セットは、システム上に残ります。

登録済みオブジェクトを非活動化する前に、この登録済みオブジェクトに関連付けられたすべてのサブスクリプション・セットを非活動化する必要があります。これにより、ユーザーがオブジェクトを削除する、または再度活動化する準備が整う前に、アプライ・プログラムがオブジェクトを自動的に再活動化し、非活動化処理に介入してくることを防止できます。

オブジェクトが非活動化され、DB2 レプリケーションがそのオブジェクトに対する変更のキャプチャーを停止すると、登録済みオブジェクトに関連付けられたすべてのサブスクリプション・セットが影響を受けます。これらのサブスクリプション・セットの実行を続けたい場合は、この登録済みオブジェクトをソースとして使用するサブスクリプション・セット・メンバーを、非活動化されたサブスクリプション・セットから除去する必要があります。

制約事項:

非活動化できる DB2 登録済みオブジェクトは、キャプチャー・プログラム・ソースとして定義されているものだけです。

キャプチャー・トリガーから使用される DB2 以外のリレーショナル・データベース・オブジェクトを非活動化することはできません。

プロシージャ:

登録済みオブジェクトを非活動化するには、次のようにします。

1. 次の方法を使用して、関連するすべてのサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

詳細については、229 ページの『サブスクリプション・セットの非活動化』を参照してください。

2. 次の方法のいずれかを使用して、登録済みオブジェクトを非活動化します。

レプリケーション・センター

「登録済み表」フォルダーで内容ペインの中の登録済み表を右マウス・ボタンでクリックし、「変更の取り込みを停止する」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

CAPSTOP シグナル

シグナル (IBMSNAP_SIGNAL) 表に手動で CAPSTOP シグナルを挿入します。詳細については、513 ページの『*schema.IBMSNAP_SIGNAL*』を参照してください。

登録の再活動化

登録および関連するサブスクリプション・セットを一時的に非活動化し、その後に登録を再活動化してデータのキャプチャーを再開するには、レプリケーション・センターを使用して単純にこれらのサブスクリプション・セットを再活動化します。キャプチャー・プログラムは、アプライ・プログラムが CAPSTART シグナルを送信した後に登録を再活動化します。

しかし、予期しないエラーのためにキャプチャー・プログラムが登録を非活動化した場合は、登録を再活動化するために特別な処理を行う必要があります。予期しないエラーが発生すると、この登録の STOP_ON_ERROR 列の値が N に設定されている場合、キ

ャプチャー・プログラムは、登録 (IBMSNAP_REGISTER) 表の中の STATE 列の値を S (Stopped) に設定します。この STATE 列の値は、キャプチャー・プログラムがこの登録の処理を停止したこと、そして登録の修復が必要であることを意味します。アプライ・プログラムは停止状態の登録に対して CAPSTART シグナルを発行することはありません。

これらの予期しないエラーを訂正し、登録を再活動化に適格なものにするには、次のプロシージャーを使用します。

前提条件:

この非活動化された登録に関してキャプチャー・プログラムから生成されたエラー・メッセージを読みます。

DB2 レプリケーションのキャプチャー・コントロール表の構成、およびシステム上で実行中のキャプチャー・プログラムについて調べておいてください。

プロシージャー:

1. エラー・メッセージに含まれた情報を使用して、登録を変更します。
2. キャプチャー・コントロール・サーバーから、次の SQL スクリプトを実行して、IBMSNAP_REGISTER 表の中の STATE 列をリセットします。

```
UPDATE Schema.IBMSNAP_REGISTER
   SET STATE          = 'I'
 WHERE
   SOURCE_OWNER      = 'SrcSchema' AND
   SOURCE_TABLE      = 'SrcTbl'    AND
   SOURCE_VIEW_QUAL  = SrcVwQual  AND
   STATE             = 'S';
```

この *Schema* はキャプチャー・スキーマの名前であり、*SrcSchema* は登録済みソース表スキーマであり、*SrcTbl* は登録済みソース表の名前であり、*SrcVwQual* はこのソース表のソース・ビュー修飾子です。

STATE 列が I (Inactive) に設定されると、キャプチャー・プログラムは、通常はアプライ・プログラムから出される CAPSTART シグナルを受け取りしだい、データのキャプチャーを開始することができます。

例: アクティブな登録のソース表が誤って DATA CAPTURE NONE に変更されたとします (本来は DATA CAPTURE CHANGES である必要があります)。また、エラーを検出してもキャプチャー・プログラムが停止しないことを指定する STOP_ON_ERROR = 'N' を使用してこの登録が定義されていたとします。キャプチャー・プログラムは次の再始動、または再初期化時に、ソース表のこの誤った条件を認識し、この登録について、登録 (IBMSNAP_REGISTER) 表の中の STATE 列を S (Stopped) に設定します。登録は停止状態になるため、アプライ・プログラムが対応するサブスクリプション・セットを処理しようとする、エラー・メッセージが発行されます。以下のことを行う必要があります。

- 表のオプションを DATA CAPTURE CHANGES にリセットする ALTER TABLE ステートメントをサブミットして、SQL によりソース表の設定を訂正します。
- 上記の SQL スクリプトを使用して、手で登録を停止状態から非活動状態にリセットします。

アプライ・プログラムはサブスクリプション・セット全体に対してフル・リフレッシュを実行します。

登録の除去

ユーザーが登録をドロップすると、DB2 レプリケーションはオブジェクトの登録をドロップし、関連する変更データ (CD) 表または整合変更データ (CCD) 表をドロップし、DB2 以外のリレーショナル・データベース・ソースの CCD オブジェクト・ニックネームおよびキャプチャー・トリガーをドロップします。実際のソース表またはビューはデータベース内に残ります。

前提条件:

キャプチャー・プログラムがこのオブジェクトの現行の処理を終了できるように、最初にソース・オブジェクトを非活動化します。

重要: 非活動化は非同期処理です。オブジェクトの除去前に必ず非活動化処理が終了するようにしてください。

プロシージャ:

ソース表またはビューの登録を除去するには、次の方法のいずれかを使用します。

レプリケーション・センター

「登録済み表の削除」または「登録済みビューの削除」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

RMVDPREG システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、431 ページの『RMVDPREG: DPR 登録の除去 (OS/400)』を参照してください。

キャプチャー・スキーマの変更

既存のキャプチャー・スキーマを変更するには、次のプロシージャを使用します。

前提条件:

次の SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表、およびシステムに定義されているサブスクリプション・セットについてよく調べておいてください。

UNIX、Windows、z/OS の場合: モニター定義が設定されているか、あるいは、変更しようとしているキャプチャー・スキーマの下でレプリケーション・アラート・モニター・プログラムを始動している場合には、これらのモニター定義をドロップしてください。キャプチャー・スキーマを変更した後、レプリケーション・センターを使用して、新しいキャプチャー・スキーマ名を使用してモニター定義を再作成します。次に、**asnmcmd** システム・コマンドで **reinit** パラメーターを使用して、関連するレプリケーション・アラート・モニター・プログラムを再初期化できます。この代わりに、**asnmcmd** システム・コマンドで **stop** パラメーターを使用してレプリケーション・アラート・モニター・プログラムを停止してから、**asnmon** システム・コマンドを使用してプログラムを再始動することもできます。

新しいキャプチャー・スキーマ名を決定します。詳細については、321 ページの『第 16 章 レプリケーション・オブジェクトの命名規則』を参照してください。

このプロシージャを使用する前に、キャプチャー・コントロール・サーバー、およびこのキャプチャー・コントロール・サーバーに関連するすべてのアプライ・コントロール・サーバーが、バージョン 8 にマイグレーション済みであることを確認してください。

制約事項:

ソース・サーバーが DB2 以外のリレーショナル・データベースである場合は、このプロシージャは使用しないでください。

プロシージャ:

1. 次の方法のいずれかを使用して、新しいキャプチャー・スキーマ用のコントロール表を作成します。

レプリケーション・センター (UNIX、Windows、z/OS)

「レプリケーション・コントロール表の作成」ノートブックを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

CRTDPRTBL システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、408 ページの『CRTDPRTBL: レプリケーション・コントロール表の作成 (OS/400)』を参照してください。

2. 次の方法のいずれかを使用して、既存のキャプチャー・スキーマを使用しているキャプチャー・プログラムを停止します。(キャプチャー・プログラムが実行されていない場合は、このステップはスキップしてください)

レプリケーション・センター

「キャプチャーの停止」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnccmd システム・コマンド (UNIX、Windows、z/OS)

stop パラメーターを使用します。パラメーター記述およびコマンド構文の詳細については、340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

ENDDPRCAP システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、413 ページの『ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』を参照してください。

3. 次の方法を使用して、関連するすべてのサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

4. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、このキャプチャー・スキーマに属するソース表を持つ関連するサブスクリプション・セットのキャプチャー・スキーマ名を変更します。

```
UPDATE ASN.IBMSNAP_SUBS_SET
  SET CAPTURE_SCHEMA = 'NewSchema'
  WHERE
    CAPTURE_SCHEMA = 'ExistingSchema';
```

この *NewSchema* は新しいキャプチャー・スキーマ名であり、*ExistingSchema* は、変更しようとしているキャプチャー・スキーマの名前です。

5. このキャプチャー・スキーマに登録されたターゲット表 (たとえば CCD 表またはレプリカ・タイプ表) を持つサブスクリプション・セットを作成してある場合には、アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、これらのサブスクリプション・セットのターゲット・スキーマ名を変更します。

```
UPDATE ASN.IBMSNAP_SUBS_SET
  SET TGT_CAPTURE_SCHEMA = 'NewSchema'
  WHERE
    TGT_CAPTURE_SCHEMA = 'ExistingSchema';
```

この *NewSchema* は新しいキャプチャー・スキーマ名であり、*ExistingSchema* は、変更しようとしているキャプチャー・スキーマの名前です。

6. キャプチャー・コントロール・サーバーから SQL ステートメントを実行して、既存のキャプチャー・コントロール表のそれぞれから、ステップ 1 で作成した対応する新しいキャプチャー・コントロール表のそれぞれにアクティブ情報をコピーします。たとえば、新しい登録 (IBMSNAP_REGISTER) 表にアクティブ情報をコピーするには、次のようにします。

```
INSERT INTO NewSchema.IBMSNAP_REGISTER
      SELECT * FROM
      ExistingSchema.IBMSNAP_REGISTER;
```

この *NewSchema* は新しいキャプチャー・スキーマ名であり、*ExistingSchema* は、変更しようとしているキャプチャー・スキーマの名前です。

次の表を含めて、既存のキャプチャー・コントロール表のそれぞれに対してこのステップを繰り返します。

- IBMSNAP_CAPMON
- IBMSNAP_CAPPARMS
- IBMSNAP_CAPTRACE
- IBMSNAP_PRUNCNTL
- IBMSNAP_PRUNE_SET
- IBMSNAP_REG_EXT (OS/400 のみ)
- IBMSNAP_REGISTER
- IBMSNAP_RESTART
- IBMSNAP_SIGNAL
- IBMSNAP_UOW

(IBMSNAP_CAPENQ [UNIX、Windows、z/OS の場合] または IBMSNAP_PRUNE_LOCK コントロール表には行が含まれていないため、これらの表についてはこのステップを繰り返す必要はありません。)

CD 表は変更しないでください。。

7. 既存のスキーマ、および関連するキャプチャー・コントロール表をドロップするには、次の方法を使用します。

レプリケーション・センター

「キャプチャー・コントロール・サーバー」フォルダーで、キャプチャー・コントロール表をドロップするデータベースを右マウス・ボタンでクリックし、「キャプチャー・コントロール表のドロップ」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

8. 新しいスキーマ名を使用してキャプチャー・プログラムを再始動するには、次の方法のいずれかを使用します。

レプリケーション・センター

「キャプチャーの開始」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asncap システム・コマンド (UNIX、Windows、z/OS)

capture_schema=*NewSchema* および **startmode**=warmsi または warmns
パラメーター・オプションを使用してください。パラメーター記述およびコ

マンド構文の詳細については、334 ページの『asncap: キャプチャーの開始 (UNIX、Windows、z/OS)』を参照してください。

STRDPRCAP システム・コマンド (OS/400)

RESTART(*YES) パラメーターを使用します。パラメーター記述およびコマンド構文の詳細については、446 ページの『STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)』を参照してください。

9. 次の方法を使用して、関連するサブスクリプション・セットを再活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の非活動化されたサブスクリプション・セットを右マウス・ボタンでクリックし、「活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

新規サブスクリプション・セットの作成

既存の登録済みオブジェクトについては、いつでも新しいサブスクリプション・セットを作成し、新しいサブスクリプション・セット・メンバーをセットに追加できます。

このプロシージャは、サブスクリプション・セット・メンバーを含む、または含まない、新しいサブスクリプション・セットの追加に使用されます。

前提条件:

新しいサブスクリプション・セットを作成する前に、ソースとして使用する表またはビューを登録します。

プロシージャ:

次の方法のいずれかを使用して、新しいサブスクリプション・セットを作成します。

レプリケーション・センター

「サブスクリプション・セットの作成」ノートブックを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

ADDDPRSUB システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、374 ページの『ADDDPRSUB: DPR サブスクリプション・セットの追加 (OS/400)』を参照してください。

追加情報については、69 ページの『第 4 章 ソースのサブスクライブ』を参照してください。

重要: 対応するアプライ・プログラムがアクティブである場合は、サブスクリプション・セットが完全に定義されるまで、新しいサブスクリプション・セットを活性化しないでください。

既存のサブスクリプション・セットへの新規サブスクリプション・セット・メンバーの追加 (UNIX、Windows、z/OS)

サブスクリプション・セットにサブスクリプション・セット・メンバーが追加されると、DB2 レプリケーションはサブスクリプション・セット内の全メンバーのフル・リフレッシュを強行します。セット全体にフル・リフレッシュを強制せずに既存のサブスクリプション・セットに新しいサブスクリプション・セット・メンバーを追加するには、次のプロシージャを使用します。

前提条件:

これらの SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表の構造、およびシステムで定義されているサブスクリプション・セットについてよく調べておいてください。

プロシージャ:

1. 次の方法を使用して、サブスクリプション・セットを非活性化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活性化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

2. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、プルーニング・コントロール (IBMSNAP_PRUNCNTL) 表の中の同期点および同期時刻の値を、プルーニング・セット (IBMSNAP_PRUNE_SET) 表の中の対応する値と同期化してください。

```
UPDATE Schema.IBMSNAP_PRUNCNTL
SET SYNCHPOINT =
(SELECT SYNCHPOINT FROM Schema.IBMSNAP_PRUNE_SET
WHERE APPLY_QUAL = 'ApplyQual' AND
SET_NAME = 'Name' AND
TARGET_SERVER = 'Target_Server'),
SYNCHTIME =
(SELECT SYNCHTIME FROM Schema.IBMSNAP_PRUNE_SET
WHERE APPLY_QUAL = 'ApplyQual' AND
SET_NAME = 'Name' AND
TARGET_SERVER = 'Target_Server')
WHERE
APPLY_QUAL = 'ApplyQual' AND
SET_NAME = 'Name' AND
TARGET_SERVER = 'Target_Server';
```

この *Schema* はキャプチャー・スキーマの名前であり、*ApplyQual* はアプライ修飾子であり、*Name* はメンバーを追加するサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

この SQL を実行することにより、各サブスクリプション・セット・メンバーについて、アプライ・プログラムを変更データ (CD) 表の正しい位置から開始できるようになります。

3. 次の方法を使用して、このサブスクリプション・セットに新規のサブスクリプション・セット・メンバーを追加します。

レプリケーション・センター

「サブスクリプション・セットにメンバーを追加する」ノートブックを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

このサブスクリプション・セットは非活動化されたままにしておいてください。

4. キャプチャー・コントロール・サーバーから、この新しいサブスクリプション・セット・メンバーについて CAPSTART シグナルを挿入します。
 - a. 次の SQL SELECT ステートメントを実行して、この新規のサブスクリプション・セット・メンバーの MAP_ID の値を検索します。

```
SELECT MAP_ID FROM Schema.IBMSNAP_PRUNCNTL
WHERE
    SOURCE_OWNER      = 'SrcSchema'      AND
    SOURCE_TABLE      = 'SrcTbl'        AND
    SOURCE_VIEW_QUAL  = 'SrcVwQual'     AND
    APPLY_QUAL        = 'ApplyQual'     AND
    SET_NAME          = 'Name'          AND
    TARGET_SERVER     = 'Target_Server' AND
    TARGET_OWNER      = 'TgtSchema'     AND
    TARGET_TABLE      = 'TgtTbl'
WITH UR;
```

この *Schema* はキャプチャー・スキーマの名前であり、*SrcSchema* はソース表スキーマであり、*SrcTbl* はソース表名であり、*SrcVwQual* はこのソース表のソース・ビュー修飾子であり、*ApplyQual* はアプライ修飾子であり、*Name* はメンバーを追加するサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションであり、*TgtSchema* はターゲット表のスキーマであり、*TgtTbl* はターゲット表の名前です。

- b. MAP_ID 値を使用し、次の SQL ステートメントを実行して、シグナル (IBMSNAP_SIGNAL) 表に行を挿入します。

```
INSERT INTO Schema.IBMSNAP_SIGNAL
(SIGNAL_TIME,
 SIGNAL_TYPE,
 SIGNAL_SUBTYPE,
 SIGNAL_INPUT_IN,
 SIGNAL_STATE,
```

```

        SIGNAL_LSN)
VALUES (CURRENT_TIMESTAMP,
        'CMD',
        'CAPSTART',
        'Mapid',
        'P',
        NULL);

```

この *Schema* はキャプチャー・スキーマの名前であり、*Mapid* は前述の SQL SELECT ステートメントから入手した MAP_ID の値です。

5. キャプチャー・プログラムが CAPSTART シグナルを認識したことを確認します。
 - a. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行し、IBMSNAP_PRUNCNTL 表の中の同期点の値を表示します。

```

SELECT SYNCHPOINT FROM Schema.IBMSNAP_PRUNCNTL
WHERE
    MAP_ID                = 'Mapid'
WITH UR;

```

この *Schema* はキャプチャー・スキーマの名前であり、*Mapid* はステップ 4a の SQL SELECT ステートメントから入手した MAP_ID の値です。

- b. 同期点の値を確認します。

値が NULL である場合は、キャプチャー・プログラムがシグナルを認識し、同期点値が NULL でなくなるまで、ステップ 5a および 5b を繰り返します。
6. 任意のユーティリティー・プログラムを使用して、ソース表のデータをアンロードします。次に、このデータをターゲット表にロードします。
7. **opt4one** を *y* に設定してアプライ・プログラムを実行している場合には、アプライ・プログラムを停止してから再始動します。
8. アプライ・コントロール・サーバーから次のスクリプトを実行してサブスクリプション・セット (IBMSNAP_SUBS_SET) 表をリフレッシュすることにより、アプライ・プログラムがこのサブスクリプション・セットに対して即時に実行されるようにするとともに、フル・リフレッシュが行われることを防止します。

```

UPDATE ASN.IBMSNAP_SUBS_SET
SET SYNCHPOINT      = NULL,
    SYNCHTIME       = CURRENT_TIMESTAMP,
    LASTSUCCESS     = CURRENT_TIMESTAMP,
    LASTRUN         = CURRENT_TIMESTAMP - X MINUTES,
    STATUS          = 0,
    ACTIVATE        = 1

WHERE
    APPLY_QUAL      = 'ApplyQual'    AND
    SET_NAME        = 'Name'         AND
    WHOS_ON_FIRST   = 'Val';

```

この *X* は `sleep_minutes` の設定値よりも大きい値であり、`ApplyQual` はアプライ修飾子であり、`Name` はメンバーを追加するサブスクリプション・セットの名前であり、`Val` は F または S のいずれかです。

このサブスクリプション・セットはアクティブになり、アプライ・プログラムはサブスクリプション・セット全体の処理を再開します。

サブスクリプション・セットの属性の変更

既存のサブスクリプション・セットの属性を変更する必要があることもあります。ユーザーによる変更が考えられる属性には次のものがあります。

- 更新を適用するスケジュール (時間ベースのレプリケーションまたはイベント・ベースのレプリケーション)
- サブスクリプション・ステートメント
- サブスクリプション・セット・メンバーの `WHERE` 文節述部
- コミット・カウント
- データ・ブロック値 (`MAX_SYNCH_MINUTES`)

プロシージャ:

サブスクリプション・セットの属性を変更するには、レプリケーション・センターを使用して、次のステップを実行します。

1. サブスクリプション・セットを非活動化します。
2. サブスクリプション・セットおよびサブスクリプション・セット・メンバーを変更します。
3. サブスクリプション・セットを再活動化します。

最初にサブスクリプション・セットを非活動化することにより、アプライ・プログラムをアクティブ状態で実行したまま、ユーザーが変更を入力している間は、アプライ・プログラムがこのサブスクリプション・セットを処理することを防止できます。アプライ・プログラムは、サブスクリプション・セットが再活動化された後、次のアプライ・サイクル中に、サブスクリプション・セットの変更を認識します。

注: アプライ・プログラムの `opt4one` パラメーターを `y` に設定すると、アプライ・プログラムを停止して再始動しないかぎり、変更は認識されません (UNIX、Windows、z/OS)。

サブスクリプション・セット名の変更

サブスクリプション・セットおよびそのすべてのメンバーをドロップしてから再作成する必要なしに、サブスクリプション・セットの名前を変更するには、次のプロシージャを使用します。

前提条件:

これらの SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表の構造、およびシステムで定義されているサブスクリプション・セットについてよく調べておいてください。

UNIX、Windows、z/OS の場合: モニター定義が設定されているか、変更しようとしているサブスクリプション・セットでアラート条件を検出するようにレプリケーション・アラート・モニター・プログラムを始動してある場合には、これらのモニター定義をドロップしてください。サブスクリプション・セット名を変更した後、レプリケーション・センターを使用してモニター定義を再作成します。次に、**asnmcmd** システム・コマンドで **reinit** パラメーターを使用して、関連するレプリケーション・アラート・モニター・プログラムを再初期化できます。この代わりに、**asnmcmd** システム・コマンドで **stop** パラメーターを使用してレプリケーション・アラート・モニター・プログラムを停止してから、**asnmon** システム・コマンドを使用してプログラムを再始動することもできます。

使用する新しいサブスクリプション・セット名を決定します。

プロシージャー:

1. 次の方法を使用して、変更するサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

2. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表、およびサブスクリプション列 (IBMSNAP_SUBS_COLS) 表の中のサブスクリプション・セットの名前を変更します。

```
UPDATE ASN.IBMSNAP_SUBS_SET
  SET SET_NAME = 'NewSetName'
  WHERE
    APPLY_QUAL = 'ApplyQual' AND
    SET_NAME = 'ExistSetName' AND
    WHOS_ON_FIRST = 'Val';

UPDATE ASN.IBMSNAP_SUBS_MEMBR
  SET SET_NAME = 'NewSetName'
  WHERE
    APPLY_QUAL = 'ApplyQual' AND
    SET_NAME = 'ExistSetName' AND
    WHOS_ON_FIRST = 'Val';
```

```

UPDATE ASN.IBMSNAP_SUBS_COLS
  SET SET_NAME      = 'NewSetName'
  WHERE
        APPLY_QUAL   = 'ApplyQual'      AND
        SET_NAME      = 'ExistSetName'   AND
        WHOS_ON_FIRST = 'Val';

```

この *NewSetName* は新しいサブスクリプション・セット名であり、*ApplyQual* はアプライ修飾子であり、*ExistSetName* はサブスクリプション・セットの既存の名前であり、*Val* は F または S のいずれかです。

- このサブスクリプション・セットが事前または事後に実行される SQL ステートメントまたはプロシージャ呼び出しを使用する場合は、アプライ・コントロール・サーバーから次の SQL スクリプトを実行して、サブスクリプション・ステートメント (IBMSNAP_SUBS_STMTS) 表の中のサブスクリプション・セット名を変更します。

```

UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET SET_NAME      = 'NewSetName'
  WHERE
        APPLY_QUAL   = 'ApplyQual'      AND
        SET_NAME      = 'ExistSetName'   AND
        WHOS_ON_FIRST = 'Val';

```

この *NewSetName* は新しいサブスクリプション・セット名であり、*ApplyQual* はアプライ修飾子であり、*ExistSetName* はサブスクリプション・セットの既存の名前であり、*Val* は F または S のいずれかです。

- キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、プルーニング・セット (IBMSNAP_PRUNE_SET) 表およびプルーニング・コントロール (IBMSNAP_PRUNCNTL) 表の中のサブスクリプション・セット名を変更します。

```

UPDATE Schema.IBMSNAP_PRUNE_SET
  SET SET_NAME      = 'NewSetName'
  WHERE
        APPLY_QUAL   = 'ApplyQual'      AND
        SET_NAME      = 'ExistSetName'   AND
        TARGET_SERVER = 'Target_Server';

```

```

UPDATE Schema.IBMSNAP_PRUNCNTL
  SET SET_NAME      = 'NewSetName'
  WHERE
        APPLY_QUAL   = 'ApplyQual'      AND
        SET_NAME      = 'ExistSetName'   AND
        TARGET_SERVER = 'Target_Server';

```

この *Schema* はキャプチャー・スキーマの新しい名前であり、*NewSetName* は新しいサブスクリプション・セットの名前であり、*ApplyQual* はアプライ修飾子であり、*ExistSetName* はサブスクリプション・セットの既存の名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

- UNIX、Windows、z/OS の場合: opt4one** を y に設定してアプライ・プログラムを実行している場合には、アプライ・プログラムを停止してから再始動します。

6. 次の方法を使用して、サブスクリプション・セットを再活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の非活動化されたサブスクリプション・セットを右マウス・ボタンでクリックし、「活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

サブスクリプション・セットの分割

次のプロシージャーを使用すると、サブスクリプション・セット情報を除去して再作成する必要なしに、サブスクリプション・セットを複数のサブスクリプション・セットに分割できます。

前提条件:

これらの SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表の構造、およびシステムで定義されているサブスクリプション・セットについてよく調べておいてください。

分割するサブスクリプション・セットのサブスクリプション・セット・メンバーを識別し、これらのサブスクリプション・セット・メンバーに関連するソース表およびターゲット表を判別します。

分割するサブスクリプション・セットのキャプチャー・コントロール・サーバー、ターゲット・サーバー、およびアプライ・コントロール・サーバーを識別します。このプロシージャーを使用して作成する新しいサブスクリプション・セットでは、これらのキャプチャー・コントロール・サーバー、ターゲット・サーバー、およびアプライ・コントロール・サーバーのロケーションを使用する必要があります。

UNIX、Windows、z/OS の場合: モニター定義が設定されているか、分割しようとしているサブスクリプション・セットでアラート条件を検出するようにレプリケーション・アラート・モニター・プログラムを始動してある場合には、これらのモニター定義をドロップしてください。サブスクリプション・セットを分割した後、レプリケーション・センターを使用して、モニター定義を再作成します。次に、**asnmcmd** システム・コマンドで **reinit** パラメーターを使用して、関連するレプリケーション・アラート・モニター・プログラムを再初期化できます。この代わりに、**asnmcmd** システム・コマンドで **stop** パラメーターを使用してレプリケーション・アラート・モニター・プログラムを停止してから、**asnmcmd** システム・コマンドを使用してプログラムを再始動することもできます。

プロシージャー:

1. 次の方法を使用して、分割するサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活性化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

2. 次の方法のいずれかを使用して、新しいサブスクリプション・セットを作成します。

レプリケーション・センター

「サブスクリプション・セットの作成」ノートブックを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

ADDDPRSUB システム・コマンド (OS/400)

SRCTBL(*NONE)、**TGTTBL(*NONE)**、および **ACTIVATE(*NO)** パラメーター・オプションを使用します。パラメーター記述およびコマンド構文の詳細については、374 ページの『ADDDPRSUB: DPR サブスクリプション・セットの追加 (OS/400)』を参照してください。

これらの方法は両方とも、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表内に新しい行を作成します。

この新しいサブスクリプション・セットは非アクティブなままおいておきます。

3. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、既存のサブスクリプション・セットから IBMSNAP_SUBS_SET 表の中の新しいサブスクリプション・セットの行に情報をコピーします。

```
UPDATE ASN.IBMSNAP_SUBS_SET
SET STATUS =
    (SELECT STATUS FROM ASN.IBMSNAP_SUBS_SET B
     WHERE APPLY_QUAL = 'ApplyQual' AND
           SET_NAME = 'ExistName' AND
           WHOS_ON_FIRST = 'Val'),
    LASTRUN =
    (SELECT LASTRUN FROM ASN.IBMSNAP_SUBS_SET B
     WHERE APPLY_QUAL = 'ApplyQual' AND
           SET_NAME = 'ExistName' AND
           WHOS_ON_FIRST = 'Val'),
    SYNCHPOINT =
    (SELECT SYNCHPOINT FROM ASN.IBMSNAP_SUBS_SET B
     WHERE APPLY_QUAL = 'ApplyQual' AND
           SET_NAME = 'ExistName' AND
           WHOS_ON_FIRST = 'Val'),
    SYNCHTIME =
    (SELECT SYNCHTIME FROM ASN.IBMSNAP_SUBS_SET B
     WHERE APPLY_QUAL = 'ApplyQual' AND
           SET_NAME = 'ExistName' AND
           WHOS_ON_FIRST = 'Val'),
    LASTSUCCESS =
    (SELECT LASTSUCCESS FROM ASN.IBMSNAP_SUBS_SET B
     WHERE APPLY_QUAL = 'ApplyQual' AND
```

```

                                SET_NAME      = 'ExistName' AND
                                WHOS_ON_FIRST = 'Val')
WHERE
    APPLY_QUAL      = 'ApplyQual' AND
    SET_NAME        = 'NewName'   AND
    WHOS_ON_FIRST  = 'Val';

```

この *ApplyQual* はアプライ修飾子であり、*ExistName* は分割する既存のサブスクリプション・セットの名前であり、*Val* は F または S のいずれかであり、*NewName* は、ユーザーが作成している新しいサブスクリプション・セットの名前です。

4. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、プルーンング・セット (*IBMSNAP_PRUNE_SET*) 表に新しいサブスクリプション・セット用の新しい行を挿入します。

```

INSERT INTO Schema.IBMSNAP_PRUNE_SET
    (APPLY_QUALIFIER,
     SET_NAME,
     TARGET_SERVER,
     SYNCHTIME,
     SYNCHPOINT
VALUES ('ApplyQual',
       'NewName',
       'Target_Server',
       NULL,
       x'00000000000000000000');

```

この *Schema* はキャプチャー・スキーマの名前であり、*ApplyQual* はアプライ修飾子であり、*NewName* は作成している新しいサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

5. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、*IBMSNAP_PRUNE_SET* 表の中の既存のサブスクリプション・セットの行から新しいサブスクリプション・セットの行に情報をコピーします。

```

UPDATE Schema.IBMSNAP_PRUNE_SET
    SET SYNCHPOINT =
        (SELECT SYNCHPOINT FROM Schema.IBMSNAP_PRUNE_SET B
         WHERE APPLY_QUAL = 'ApplyQual' AND
               SET_NAME   = 'ExistName' AND
               TARGET_SERVER = 'Target_Server'),
    SYNCHTIME =
        (SELECT SYNCHTIME FROM Schema.IBMSNAP_PRUNE_SET B
         WHERE APPLY_QUAL = 'ApplyQual' AND
               SET_NAME   = 'ExistName' AND
               TARGET_SERVER = 'Target_Server')
WHERE
    APPLY_QUAL = 'ApplyQual' AND
    SET_NAME   = 'NewName'   AND
    TARGET_SERVER = 'Target_Server';

```

この *Schema* はキャプチャー・スキーマの名前であり、*ApplyQual* はアプライ修飾子であり、*ExistName* は分割する既存のサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションであり、*NewName* はユーザーが作成しようとしている新しいサブスクリプション・セットの名前です。

6. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBER) 表およびサブスクリプション列 (IBMSNAP_SUBS_COLS) 表で、新しいサブスクリプション・セットに移動する各 サブスクリプション・セット・メンバーのサブスクリプション・セット名を変更します。

```
UPDATE ASN.IBMSNAP_SUBS_MEMBER
  SET SET_NAME      = 'NewName'
  WHERE
    APPLY_QUAL      = 'ApplyQual' AND
    SET_NAME        = 'ExistName' AND
    WHOS_ON_FIRST   = 'Val'       AND
    SOURCE_OWNER    = 'SrcSchema' AND
    SOURCE_TABLE    = 'SrcTbl'    AND
    SOURCE_VIEW_QUAL = SrcVwQual  AND
    TARGET_OWNER    = 'TgtSchema' AND
    TARGET_TABLE    = 'TgtTbl';
```

```
UPDATE ASN.IBMSNAP_SUBS_COLS
  SET SET_NAME      = 'NewName'
  WHERE
    APPLY_QUAL      = 'ApplyQual' AND
    SET_NAME        = 'ExistName' AND
    WHOS_ON_FIRST   = 'Val'       AND
    TARGET_OWNER    = 'TgtSchema' AND
    TARGET_TABLE    = 'TgtTbl';
```

この *NewName* はユーザーが作成している新しいサブスクリプション・セットであり、*ApplyQual* はアプライ修飾子であり、*ExistName* は分割する既存のサブスクリプション・セットであり、*Val* は F または S のいずれかであり、*SrcSchema* はソース表スキーマであり、*SrcTbl* はソース表名であり、*SrcVwQual* はこのソース表のソース・ビュー修飾子であり、*TgtSchema* はターゲット表のスキーマであり、*TgtTbl* はターゲット表名です。

新しいサブスクリプション・セットに移動するサブスクリプション・セット・メンバーごとにこのステップを繰り返します。

7. 分割するサブスクリプション・セットが事前または事後に実行される SQL ステートメントまたはプロシージャー呼び出しを使用する場合は、サブスクリプション・ステートメント (IBMSNAP_SUBS_STMTS) 表の中で適切なステートメントを新しいサブスクリプション・セットに移動します。
 - a. アプライ・コントロール・サーバーから次の SQL スクリプトを実行して、ステートメントを移動します。

```

UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET SET_NAME      = 'NewName'
  WHERE
    APPLY_QUAL      = 'ApplyQual' AND
    SET_NAME        = 'ExistName' AND
    WHOS_ON_FIRST   = 'Val'       AND
    STMT_NUMBER     in (Stmt1,Stmt2,..Stmnt);

```

この *NewName* はユーザーが作成している新しいサブスクリプション・セットの名前であり、*ApplyQual* はアプライ修飾子であり、*ExistName* は分割する既存のサブスクリプション・セットの名前であり、*Val* は F または S のいずれかであり、*Stmt1*、*Stmt2*、および *Stmnt* は、新しいサブスクリプション・セットに移動するステートメントの番号に対応します。

- b. 両方のサブスクリプション・セットについて、ステートメントの新しいカウントを反映するように、IBMSNAP_SUBS_SET 表の中の AUX_STMTS 列の値を調整します。必要であれば、重複しないようにステートメントの番号を変更します。
8. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、移動した各 サブスクリプション・セット・メンバーについて、プルーニング・コントロール (IBMSNAP_PRUNCNTL) 表の中のサブスクリプション・セットの名前を変更します。

```

UPDATE Schema.IBMSNAP_PRUNCNTL
  SET SET_NAME      = 'NewName'
  WHERE
    APPLY_QUAL      = 'ApplyQual'      AND
    SET_NAME        = 'ExistName'      AND
    TARGET_SERVER   = 'Target_Server'  AND
    SOURCE_OWNER    = 'SrcSchema'      AND
    SOURCE_TABLE    = 'SrcTbl'         AND
    SOURCE_VIEW_QUAL = SrcVwQual       AND
    TARGET_OWNER    = 'TgtSchema'      AND
    TARGET_TABLE    = 'TgtTbl';

```

この *Schema* はキャプチャー・スキーマの名前であり、*NewName* はステップ 2 で作成した新しいサブスクリプション・セットの名前であり、*ApplyQual* はアプライ修飾子であり、*ExistName* は分割された既存のサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションであり、*SrcSchema* はソース表スキーマであり、*SrcTbl* はソース表名であり、*SrcVwQual* はこのレプリケーション・ソース表のソース・ビュー修飾子であり、*TgtSchema* はターゲット表スキーマであり、*TgtTbl* はターゲット表名です。

新しいサブスクリプション・セットに移動したサブスクリプション・セット・メンバーごとにこのステップを繰り返します。

9. **UNIX、Windows、z/OS の場合: opt4one** を *y* に設定してアプライ・プログラムを実行している場合には、アプライ・プログラムを停止してから再始動します。
10. 次の方法を使用して、両方のサブスクリプション・セットを再活性化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の非活動化された両方のサブスクリプション・セットを右マウス・ボタンでクリックし、「活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

サブスクリプション・セットのマージ

2 つのサブスクリプション・セットを 1 つにマージする場合は次のプロシージャーを使用します。2 つのサブスクリプション・セットの中のターゲット表が同じトランザクション整合性を持つようにしたいが、サブスクリプション・セット情報を削除して再作成したくない場合は、サブスクリプション・セットをマージできます。

前提条件:

これらの SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表の構造、およびシステムで定義されているサブスクリプション・セットについてよく調べておいてください。

マージする各サブスクリプション・セットのキャプチャー・コントロール・サーバー、ターゲット・サーバー、およびアプライ・コントロール・サーバーを識別します。マージするすべてのサブスクリプション・セットが、同じキャプチャー・コントロール・サーバー、ターゲット・サーバー、およびアプライ・コントロール・サーバーを使用して作成されていることを確認します。

UNIX、Windows、z/OS の場合: モニター定義が設定されているか、マージしようとしているサブスクリプション・セットでアラート条件を検出するようにレプリケーション・アラート・モニター・プログラムを始動してある場合には、これらのモニター定義をドロップしてください。サブスクリプション・セットをマージした後、レプリケーション・センターを使用して、モニター定義を再作成します。次に、**asnmcmd** システム・コマンドで **reinit** パラメーターを使用して、関連するレプリケーション・アラート・モニター・プログラムを再初期化できます。この代わりに、**asnmcmd** システム・コマンドで **stop** パラメーターを使用してレプリケーション・アラート・モニター・プログラムを停止してから、**asnmon** システム・コマンドを使用してプログラムを再始動することもできます。

制約事項:

マージされる 2 つのサブスクリプション・セットは、同じキャプチャー・サーバーから、そして同じキャプチャー・スキーマを通じてデータを得ている必要があります。

プロシージャー:

1. 次の方法のいずれかを使用して、関連するキャプチャー・プログラムを停止します。

レプリケーション・センター

「キャプチャーの停止」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

asnccmd システム・コマンド (Windows、UNIX、z/OS)

stop パラメーターを使用します。パラメーター記述およびコマンド構文の詳細については、340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

ENDDDPRCAP システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、413 ページの『ENDDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』を参照してください。

両方のサブスクリプション・セットが、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表に示されたものと同じ同期点および同期時刻に達するまで待ちます。

重要: サブスクリプション・セットがマージされたときにデータが失われないように、2 つのサブスクリプション・セットでは、同じ同期点値までのソース・データが処理済みである必要があります。

ヒント: キャプチャー・プログラムを停止したくない場合は、シグナル (IBMSNAP_SIGNAL) 表の中に USER シグナルを挿入し、END_SYNCHPOINT (サブスクリプション・イベント [IBMSNAP_SUBS_EVENT] 表の中にある) を IBMSNAP_SIGNAL 表の中の SIGNAL_LSN 列の値に設定してイベントを生成し、このエンドポイントまでのデータのみが適用されるようにします。

2. 次の方法を使用して、2 つのサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の 2 つのアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

3. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、別のサブスクリプション・セット内に移動するサブスクリプション・セットに対応する行を、IBMSNAP_SUBS_SET 表から削除します。

```
DELETE FROM ASN.IBMSNAP_SUBS_SET
WHERE
    APPLY_QUAL = 'ApplyQual' AND
    SET_NAME = 'Subset_To_Move' AND
    WHOS_ON_FIRST = 'Val';
```

この *ApplyQual* はアプライ修飾子であり、*Subset_To_Move* は別の既存のサブスクリプション・セット内に移動するサブスクリプション・セットの名前であり、*Val* は F または S のいずれかです。

4. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、別のサブスクリプション・セット内に移動するサブスクリプション・セットに対応する行を、プルーニング・セット (IBMSNAP_PRUNE_SET) 表から削除します。

```
DELETE FROM Schema.IBMSNAP_PRUNE_SET
WHERE
    APPLY_QUAL    = 'ApplyQual'      AND
    SET_NAME      = 'Subset_To_Move' AND
    TARGET_SERVER = 'Target_Server' ;
```

この *Schema* はキャプチャー・スキーマの名前であり、*ApplyQual* はアプライ修飾子であり、*Subset_To_Move* は別の既存のサブスクリプション・セット内に移動するサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

5. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表およびサブスクリプション列 (IBMSNAP_SUBS_COLS) 表の中で、移動するサブスクリプション・セットの名前を、別のサブスクリプション・セットの名前に変更します。

```
UPDATE ASN.IBMSNAP_SUBS_MEMBR
SET SET_NAME = 'Existing_Merged_Subset'
WHERE
    APPLY_QUAL    = 'ApplyQual'      AND
    SET_NAME      = 'Subset_To_Move' AND
    WHOS_ON_FIRST = 'Val';
```

```
UPDATE ASN.IBMSNAP_SUBS_COLS
SET SET_NAME = 'Existing_Merged_Subset'
WHERE
    APPLY_QUAL    = 'ApplyQual'      AND
    SET_NAME      = 'Subset_To_Move' AND
    WHOS_ON_FIRST = 'Val';
```

この *Existing_Merged_Subset* は、移動するサブスクリプション・セットとマージされる既存のサブスクリプション・セットの名前であり、*ApplyQual* はアプライ修飾子であり、*Subset_To_Move* は既存のサブスクリプション・セット内に移動するサブスクリプション・セットの名前であり、*Val* は F または S のいずれかです。

6. 移動するサブスクリプション・セットが事前または事後に実行される SQL ステートメントまたはプロシージャー呼び出しを使用する場合は、サブスクリプション・ステートメント (IBMSNAP_SUBS_STMTS) 表の中でサブスクリプション・セットの名前を変更します。

- a. アプライ・コントロール・サーバーから次の SQL スクリプトを実行して、サブスクリプション・セットの名前を変更します。

```
UPDATE ASN.IBMSNAP_SUBS_STMTS
SET SET_NAME = 'Existing_Merged_Subset'
WHERE
    APPLY_QUAL    = 'ApplyQual'      AND
    SET_NAME      = 'Subset_To_Move' AND
    WHOS_ON_FIRST = 'Val';
```

この *Existing_Merged_Subset* は、移動するサブスクリプション・セットとマージされる既存のサブスクリプション・セットの名前であり、*ApplyQual* はアプライ修飾子であり、*Subset_To_Move* は既存のサブスクリプション・セット内に移動するサブスクリプション・セットの名前であり、*Val* は F または S のいずれかです。

- b. 既存のマージされたサブスクリプション・セット内のステートメントの新しいカウントを反映するように、IBMSNAP_SUBS_SET 表の中の AUX_STMTS 列の値を調整します。必要であれば、重複しないようにステートメントの番号を変更します。
7. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、プルーニング・コントロール (IBMSNAP_PRUNCNTL) 表の中で、移動したサブスクリプション・セットの名前を、マージされたサブスクリプション・セットの名前に変更します。

```
UPDATE Schema.IBMSNAP_PRUNCNTL
   SET SET_NAME      = 'Existing_Merged_Subset'
   WHERE
     APPLY_QUAL      = 'ApplyQual'      AND
     SET_NAME        = 'Subset_To_Move' AND
     TARGET_SERVER   = 'Target_Server' ;
```

この *Schema* はキャプチャー・スキーマの名前であり、*Existing_Merged_Subset* は移動するサブスクリプション・セットとマージされる既存のサブスクリプション・セットの名前であり、*ApplyQual* はアプライ修飾子であり、*Subset_To_Move* は別の既存のサブスクリプション・セット内に移動するサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

8. **UNIX、Windows、z/OS の場合: opt4one** を *y* に設定してアプライ・プログラムを実行している場合には、アプライ・プログラムを停止してから再始動します。
9. 次の方法を使用して、マージしたサブスクリプション・セットを再活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の非活動化されたサブスクリプション・セットを右マウス・ボタンでクリックし、「活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

サブスクリプション・セットのアプライ修飾子の変更

サブスクリプション・セットのアプライ修飾子を変更する必要がある場合には、サブスクリプション・セットを削除および再作成することなく、SQL を使用して変更を行うことができます。

同じアプライ修飾子を使用する複数のサブスクリプション・セットがある場合には、アプライ・プログラムの作業負荷のバランスを取るために、いくつかのサブスクリプション・セットを新しいアプライ修飾子に移動することも考えられます。

移動するサブスクリプション・セットごとに このプロシーチャーの SQL ステートメントを実行する必要があります。

前提条件:

これらの SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表の構造、およびシステムで定義されているサブスクリプション・セットについてよく調べておいてください。

また、次の情報も確認してください。

- 新しいアプライ修飾子の名前。(詳細については、321 ページの『第 16 章 レプリケーション・オブジェクトの命名規則』を参照。)
- 既存のアプライ修飾子から新しいアプライ修飾子に移動するサブスクリプション・セット。
- これらのサブスクリプション・セットに定義されている、事前または事後に実行される SQL ステートメントまたはプロシーチャー呼び出し。

UNIX、Windows、z/OS の場合: モニター定義が設定されているか、変更しようとしているアプライ修飾子の下でレプリケーション・アラート・モニター・プログラムを始動してある場合には、これらのモニター定義をドロップしてください。アプライ修飾子を変更した後、レプリケーション・センターを使用して、新しいアプライ修飾子名を使用してモニター定義を再作成します。次に、**asnmcmd** システム・コマンドで **reinit** パラメーターを使用して、関連するレプリケーション・アラート・モニター・プログラムを再初期化できます。この代わりに、**asnmcmd** システム・コマンドで **stop** パラメーターを使用してレプリケーション・アラート・モニター・プログラムを停止してから、**asnmmon** システム・コマンドを使用してプログラムを再始動することもできます。

プロシーチャー:

1. 次の方法を使用して、変更するサブスクリプション・セットを非活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

2. アプライ・コントロール・サーバーから次の SQL ステートメントを実行して、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表、サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表、およびサブスクリプション列 (IBMSNAP_SUBS_COLS) 表の中のサブスクリプション・セットのアプライ修飾子を変更します。

```

UPDATE ASN.IBMSNAP_SUBS_SET
  SET APPLY_QUAL = 'NewApplyQual'
  WHERE
    APPLY_QUAL = 'ExistApplyQual' AND
    SET_NAME = 'Name' AND
    WHOS_ON_FIRST = 'Val';

UPDATE ASN.IBMSNAP_SUBS_MEMBR
  SET APPLY_QUAL = 'NewApplyQual'
  WHERE
    APPLY_QUAL = 'ExistApplyQual' AND
    SET_NAME = 'Name' AND
    WHOS_ON_FIRST = 'Val';

UPDATE ASN.IBMSNAP_SUBS_COLS
  SET APPLY_QUAL = 'NewApplyQual'
  WHERE
    APPLY_QUAL = 'ExistApplyQual' AND
    SET_NAME = 'Name' AND
    WHOS_ON_FIRST = 'Val';

```

この *NewApplyQual* は新しいアプライ修飾子であり、*ExistApplyQual* は既存のアプライ修飾子であり、*Name* はサブスクリプション・セットの名前であり、*Val* は F または S のいずれかです。

- このサブスクリプション・セットが事前または事後に実行される SQL ステートメントまたはプロシージャ呼び出しを使用する場合は、アプライ・コントロール・サーバーから次の SQL スクリプトを実行して、サブスクリプション・ステートメント (IBMSNAP_SUBS_STMTS) 表の中のサブスクリプション・セットのアプライ修飾子を変更します。

```

UPDATE ASN.IBMSNAP_SUBS_STMTS
  SET APPLY_QUAL = 'NewApplyQual'
  WHERE
    APPLY_QUAL = 'ExistApplyQual' AND
    SET_NAME = 'Name' AND
    WHOS_ON_FIRST = 'Val';

```

この *NewApplyQual* は新しいアプライ修飾子であり、*ExistApplyQual* は既存のアプライ修飾子であり、*Name* はサブスクリプション・セットの名前であり、*Val* は F または S のいずれかです。

- キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、プルーニング・セット (IBMSNAP_PRUNE_SET) 表およびプルーニング・コントロール (IBMSNAP_PRUNCNTL) 表の中のサブスクリプション・セットのアプライ修飾子を変更します。

```

UPDATE Schema.IBMSNAP_PRUNE_SET
  SET APPLY_QUAL = 'NewApplyQual'
  WHERE
    APPLY_QUAL = 'ExistApplyQual' AND
    SET_NAME = 'Name' AND
    TARGET_SERVER = 'Target_Server';

```

```

UPDATE Schema.IBMSNAP_PRUNCNTL
SET APPLY_QUAL = 'NewApplyQual'
WHERE
  APPLY_QUAL = 'ExistApplyQual' AND
  SET_NAME = 'Name' AND
  TARGET_SERVER = 'Target_Server';

```

この *Schema* はキャプチャー・スキーマの名前であり、*NewApplyQual* は新しいアプライ修飾子であり、*ExistApplyQual* は既存のアプライ修飾子であり、*Name* はサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

5. 移動する残りのサブスクリプション・セットのそれぞれについて、ステップ 2 から 4 を繰り返します。
6. **UNIX、Windows、z/OS の場合: opt4one** を *y* に設定してアプライ・プログラムを実行している場合には、アプライ・プログラムを停止してから再始動します。
7. 次の方法を使用してサブスクリプション・セットを再活動化します。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中の非活動化されたサブスクリプション・セットを右マウス・ボタンでクリックし、「活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

サブスクリプション・セットの非活動化

サブスクリプション・セットは、除去することなく、非活動化できます。サブスクリプション・セットを非活動化すると、アプライ・プログラムは、現在の処理サイクルを完了させてから、サブスクリプション・セットの処理を停止します。サブスクリプション・セットを非活動化させる時間の長さによっては、これらの非活動化したサブスクリプション・セットに関して特別な保守が必要になります。

短期間 一時的に非活動化したサブスクリプション・セットについては、特別な処理要件はありません。サブスクリプション・セットの属性を変更するとき、またはターゲット表の障害を修復するときには、サブスクリプション・セットを一時的に非活動化する必要があります。

サブスクリプション・セットを非活動化、変更、および再活動化するには、レプリケーション・センターを使用します。

より長い期間

現在は必要ないが、将来使用する可能性のあるサブスクリプション・セットは非活動化しておくことができます。しかし、このサブスクリプション・セットを長期間にわたって非活動化しておく必要がある場合には、累積した変更データによってキャプチャー・プログラムおよびアプライ・プログラムのパフォーマンスが影響を受ける可能性があるため、追加の処理が必要になります。

キャプチャー・プログラムは、ブルーニング・プロセス時にはアプライ・プログラムからの情報を使用します。長い時間にわたりアプライ・プログラムが非アクティブになるか、サブスクリプション・セットが非活動化されると、ブルーニング情報が不整合になり、非活動化されたサブスクリプション・セットに関連してアクティブな登録が残っている場合には、作業単位 (UOW) 表および変更データ (CD) 表で迅速かつ効果的なブルーニングを行うことができなくなります。この不整合な情報により、残りのアクティブなアプライ・プログラムのパフォーマンスが大幅に低下したり、ブルーニング処理が高価な CPU を不必要に消費してしまう可能性があります。UOW 表および CD 表は、最終的にはキャプチャー・プログラムの保持制限 (デフォルト値は 7 日) に基づいて整理されます。しかし、レプリケーション環境の規模によっては、この期間に大量のデータが累積される可能性があります。

このようなブルーニングの問題を防止するために、長期間にわたり非活動化しておく必要のあるサブスクリプション・セットについては、SQL を使用してブルーニング情報をリセットすることができます。

前提条件:

これらの SQL ステートメントを実行する前に、DB2 レプリケーション・コントロール表の構造、およびシステムで定義されているサブスクリプション・セットについてよく調べておいてください。

プロシージャ:

1. レプリケーション・センターから、サブスクリプション・セットがアクティブでないことを確認します。
2. キャプチャー・コントロール・サーバーから次の SQL ステートメントを実行して、ブルーニング・セット (IBMSNAP_PRUNE_SET) 表およびブルーニング・コントロール (IBMSNAP_PRUNCNTL) 表で、非活動化したサブスクリプション・セットのブルーニング情報をリセットします。

```
UPDATE Schema.IBMSNAP_PRUNE_SET
  SET SYNCHPOINT = x'00000000000000000000' AND
      SYNCHTIME   = NULL
WHERE
  APPLY_QUAL    = 'ApplyQual' AND
  SET_NAME      = 'Name'       AND
  TARGET_SERVER = 'Target_Server';
```

```
UPDATE Schema.IBMSNAP_PRUNCNTL
  SET SYNCHPOINT = NULL AND
      SYNCHTIME   = NULL
WHERE
  APPLY_QUAL    = 'ApplyQual' AND
  SET_NAME      = 'Name'       AND
  TARGET_SERVER = 'Target_Server';
```

この *Schema* はキャプチャー・スキーマの名前であり、*ApplyQual* はアプライ修飾子であり、*Name* はサブスクリプション・セットの名前であり、*Target_Server* はターゲット表のデータベース・ロケーションです。

登録済みオブジェクトに関連付けられたすべてのサブスクリプション・セットを非活動化したときには、登録済みオブジェクトも非活動化して、キャプチャー・プログラムが不必要にデータを取り込まないようにする必要があります。

サブスクリプション・セットの除去

特定のサブスクリプション・セットでデータのレプリケーションが必要なくなったときには、サブスクリプション・セットを除去できます。しかし、除去するサブスクリプション・セットに対してアプライ・プログラムの処理が行われている場合には、アプライ・プログラムのジョブはアベンドし、このジョブの中の他のサブスクリプション・セットは、ユーザーがジョブを再始動するまで処理されません。

プロシージャ:

1. サブスクリプション・セットに対するアプライ・プログラムの現在の処理がすべて完了していることを確認するために、次の方法を使用して、サブスクリプション・セットを除去する前にサブスクリプション・セットを非活動化してください。

レプリケーション・センター

「サブスクリプション・セット」フォルダーで内容ペインの中のアクティブなサブスクリプション・セットを右マウス・ボタンでクリックし、「非活動化」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。

2. 次の方法のいずれかを使用して、非活動化したサブスクリプション・セットを除去します。

レプリケーション・センター

「サブスクリプション・セットの削除」ウィンドウを使用します。詳細については、レプリケーション・センターのヘルプを参照してください。

RMVDPRSUB システム・コマンド (OS/400)

パラメーター記述およびコマンド構文の詳細については、432 ページの『RMVDPRSUB: DPR サブスクリプション・セットの除去 (OS/400)』を参照してください。

重要: キャプチャー・プログラムは、登録済みオブジェクトのすべてのサブスクリプション・セットが除去されても、変更データ (CD) 表でのデータのキャプチャーおよび行の書き込みを続けます。このキャプチャー・プログラムによる処理の続行を防止するために、サブスクリプション・セットを除去した後で、登録済みオブジェクトを非活動化または除去してください。

データベース・アプリケーション・イベントとレプリケーション・イベントの調整

シグナル (IBMSNAP_SIGNAL) 表に手動でシグナルを挿入すれば、データベース・イベントとレプリケーション・イベントを調整できます。シグナルと呼ばれる、手動で挿入された IBMSNAP_SIGNAL 行は、実行中のキャプチャー・プログラムに特定のアクションを実行するように指示します。

USER タイプ・シグナルを使用したイベント END_SYNCHPOINT の設定

SIGNAL_TYPE 列の値を USER に設定することにより、DB2 リカバー・ログ内の正確な時点を確立し、データベース・アプリケーション・イベントとレプリケーション・イベントを調整できます。

たとえば、オンライン・トランザクション処理 (OLTP) データを、別個に保守されるデータウェアハウスに複製する場合、ウェアハウスのデータは、随時の照会のために比較的安定したものにしておく必要があります。このため、ウェアハウスのデータでは、OLTP アプリケーションの労働日の特定の時点までに発生した変更のみを更新することにします。この場合のデータベース・アプリケーション・イベントは、労働日の論理的な終了時です。レプリケーション・イベントは、特定の日のビジネスがクローズしてから、次の日のビジネスがクローズするまでの変更の適用ということになります。サブスクリプション・セットはイベント処理に対してのみ構成されていると仮定します。

プロシージャー:

USER タイプのシグナルを作成するには、次のようにします。

1. IBMSNAP_SIGNAL 表に次の行を挿入して、キャプチャー USER タイプのシグナルを作成します。

```
INSERT INTO Schema.IBMSNAP_SIGNAL
      (signal_type,
       signal_subtype,
       signal_state)
VALUES('USER',
       'USER APPLY EVENT SIGNAL',
       'P');
```

データベース・アプリケーション・イベントが発生したとき (この場合はアプリケーションの労働日の終わり) に、この SQL INSERT ステートメントを実行します。

キャプチャー・プログラムは、データベース・リカバー・ログでこのレコードを検出した後で、このシグナル表のログ・レコードを処理します。キャプチャー・プログラムが反応するのは、この挿入に対応するコミット・レコードが検出された場合、つまりこのイベントがコミットされたことが検証された場合だけです。

USER タイプのシグナルがコミットされると、キャプチャー・プログラムは、処理される挿入ログ・レコードに対応する、次の IBMSNAP_SIGNAL 列値を更新します。

- SIGNAL_STATE = 'R' (キャプチャー・プログラムが受け取り済み)

- `SIGNAL_LSN` = このシグナル行の挿入を含む DB2 の作業単位のコミット・ログ・レコードからのログ・シーケンス番号
2. 挿入されたシグナル行の `SIGNAL_LSN` 列にある値を、サブスクリプション・イベント (`IBMSNAP_SUBS_EVENT`) コントロール表の `END_SYNCPOINT` 値として使用します。この新しい値は、新しい労働日のすべてのデータがキャプチャー・プログラムによって収集されたこと、そしてアプライ・プログラムが `SIGNAL_LSN` 列の値までのデータのみをフェッチしてアプライする必要があることをアプライ・プログラムに知らせます。

`IBMSNAP_SIGNAL` 表に対する更新トリガーを作成すれば、`IBMSNAP_SUBS_EVENT` への挿入を自動化できます。

```
CREATE TRIGGER EVENT_TRIG
NO CASCADE AFTER UPDATE ON Schema.IBMSNAP_SIGNAL
REFERENCING NEW AS N
FOR EACH ROW MODE DB2SQL
WHEN (N.SIGNAL_SUBTYPE = 'USER APPLY EVENT SIGNAL')
INSERT INTO ASN.IBMSNAP_SUBS_EVENT VALUES
('WH_APPLY_EVENT',
(CURRENT_TIMESTAMP + 2 MINUTES),
N.SIGNAL_LSN,
null);
```

このトリガーは、キャプチャー・プログラムによって `IBMSNAP_SIGNAL` 表が更新されるたびに起動されます。`SIGNAL_SUBTYPE` 列が 'USER APPLY EVENT SIGNAL' に更新されると、トリガーは `IBMSNAP_SUBS_EVENT` 表に行を挿入します。この行は、2 分間が経過した後、最後の労働日 (キャプチャー・プログラムにより算出された `SIGNAL_LSN` 値の以前にコミットされている) 以降のフェッチおよび適用を処理する必要があることをアプライ・プログラムに指示します。

キャプチャー CMD STOP シグナルの使用法

`SIGNAL_TYPE` 列の値を `CMD` に設定し、`SIGNAL_SUBTYPE` 列の値を `STOP` に設定することにより、キャプチャー・プログラムの処理を、DB2 リカバー・ログの正確な時点で停止できます。この機能は主に 2 つの目的で使用されます。

- 以前のログ・レコードを読み取れない状態にした、ソース表の変更に対してキャプチャー・プログラムを調整するため。このような状態は、ユーザーが表をドロップしてから再作成した場合、またはユーザーが `KEEPDICTIONARY` オプションを `YES` に設定しないで表を再編成したときに発生する可能性があります。
- 複製された分散データベース・システム間で共通のリカバー点を調整するため。

ソース表の変更とキャプチャー・プログラムの調整

キャプチャー `CMD` タイプ `STOP` サブタイプのシグナルを使用して、キャプチャー・プログラムをシャットダウンして、ソース表の変更を調整できます。

プロシージャ:

ソース表の変更を調整するには次のようにします。

1. 次の SQL ステートメントを使用して、シグナル (IBMSNAP_SIGNAL) 表に行を挿入してキャプチャー CMD タイプ STOP サブタイプのシグナルを作成します。

```
INSERT INTO Schema.IBMSNAP_SIGNAL
    (signal_type,
     signal_subtype,
     signal_state)
VALUES('CMD',
       'STOP',
       'P');
```

この行は、データベース・アプリケーション・イベントが発生したときに、ソース表のアクティビティが静止した後、そしてログ・レコードに問題を発生させたアクティビティが変更される前に、挿入する必要があります。

キャプチャー・プログラムは、データベース・リカバー・ログでこのレコードを検出した後で、このシグナル表のログ・レコードを処理します。キャプチャー・プログラムが反応するのは、この挿入に対応するコミット・レコードが検出された場合、つまりこのイベントがコミットされたことが検証された場合だけです。

キャプチャー・プログラムは、この挿入された IBMSNAP_SIGNAL 行を含む DB2 作業単位のコミット・ログ・レコードより以前の、ログ上のトランザクションからのすべての取り込まれたデータをコミットした後で、すべてのキャプチャー・スレッドを順番にシャットダウンします。キャプチャー・プログラムは終了する前に、処理される挿入ログ・レコードに対応する IBMSNAP_SIGNAL 表の行の中の次の値も更新します。

- SIGNAL_STATE = 'R' (キャプチャー・プログラムが受け取り済み)
- SIGNAL_LSN = このシグナル行の挿入を含む DB2 の作業単位のコミット・ログ・レコードからのログ・シーケンス番号

変更のあるソース表のすべてのログ・レコードは、終了時にキャプチャー・プログラムにより処理されます。

2. ユーザーのシナリオに従って、ソース表をドロップして再作成するか、KEEPDICTIONARY オプションを YES に設定せずにソース表を再編成および圧縮します。
3. 複製された列をドロップまたは変更した場合は、このソース表に対して作成された、対応する登録およびサブスクリプション・セットをここで変更する必要があります。必要であれば、影響を受けたサブスクリプション・セットが、現在停止中のキャプチャー・プログラムに追いつくのを待つことにより、アプライ・プログラムとの間でこのような変更をさらに調整することができます。サブスクリプション・セットは、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の中の SYNCHPOINT 列の値が Schema.IBMSNAP_RESTART 表の中の MAX_COMMITSEQ 列の値と等しくなると、キャプチャー・プログラムと同期します。

分散リカバリーの設定

キャプチャー CMD タイプ STOP サブタイプのシグナルを使用して、ソース・データベースおよびターゲット・データベースを同じリカバリー点に設定し、共通整合点でデータベースをリカバリーできます。

前提条件:

このプロシージャーを使用する前に、ターゲット・データベースにアプライ・コントロール表が作成されていることを確認してください。

また、IBMSNAP_SIGNAL 表に行を挿入する前に、ソース・データベースに対するすべてのアクティビティーが静止していることを確認します。しかし、IBMSNAP_SIGNAL 表に行を挿入するまでは、データベース表のバックアップまたはイメージ・コピーを作成しないでください。

サブスクリプション・セットがイベント処理用の典型的な構成になっていない場合には、サブスクリプション・セットを一時的にイベント・ベースのタイミング用に設定する必要があります。次の SQL ステートメントを使用して、サブスクリプション・イベント (IBMSNAP_SUBS_EVENT) 表に行を挿入します。

```
INSERT INTO ASN.IBMSNAP_SUBS_EVENT
VALUES('RECOVERY_EVENT',
CURRENT_TIMESTAMP + 2 MINUTES,
SIGNAL_LSN_value,
NULL);
```

この *SIGNAL_LSN_value* は、キャプチャー・プログラムによって設定され、IBMSNAP_SIGNAL 表の中に保管されるログ・シーケンス番号です。

プロシージャー:

分散リカバリー点を設定するには、次のようにします。

1. 次の SQL ステートメントを使用して、IBMSNAP_SIGNAL 表に行を挿入してキャプチャー CMD タイプ STOP サブタイプのシグナルを作成します。

```
INSERT INTO Schema.IBMSNAP_SIGNAL
(signal_type,
signal_subtype,
signal_state)
VALUES('CMD',
'STOP',
'P');
```

キャプチャー・プログラムは、データベース・リカバリー・ログでこのレコードを検出した後で、このシグナル表のログ・レコードを処理します。キャプチャー・プログラムが反応するのは、この挿入に対応するコミット・レコードが検出された場合、つまりこのイベントがコミットされたことが検証された場合だけです。

キャプチャー・プログラムは、この挿入された `IBMSNAP_SIGNAL` 行を含む `DB2` 作業単位のコミット・ログ・レコードより以前の、ログ上のトランザクションからのすべての取り込まれたデータをコミットした後で、すべてのキャプチャー・スレッドを順番にシャットダウンします。キャプチャー・プログラムは終了する前に、処理される挿入ログ・レコードに対応する `IBMSNAP_SIGNAL` 表の行の中の次の値も更新します。

- `SIGNAL_STATE = 'R'` (キャプチャー・プログラムが受け取り済み)
- `SIGNAL_LSN` = このシグナル行の挿入を含む `DB2` の作業単位のコミット・ログ・レコードからのログ・シーケンス番号

ソース・データベースのすべてのログ・レコードは、終了時にキャプチャー・プログラムにより処理されます。

2. ソース・データベースのバックアップまたはイメージ・コピー・ユーティリティを実行します。
3. 挿入した `IBMSNAP_SIGNAL` 表の行の `SIGNAL_LSN` 列の値を、`IBMSNAP_SUBS_EVENT` 表の中の `END_SYNCPOINT` 値として使用します。この値は、バックアップ時点以前にコミットされたすべてのデータがキャプチャー・プログラムによって収集されたこと、そしてアプライ・プログラムが `SIGNAL_LSN` 列の値までのデータのみをフェッチしてアプライする必要があることをアプライ・プログラムに知らせます。

サブスクリプション・セットは、`SIGNAL_LSN` 値までのすべてのデータを処理します。

4. ターゲット・データベースのバックアップまたはイメージ・コピー・ユーティリティを実行します。これでソース・データベースとターゲット・データベースは同じリカバリー点を持つことになるため、共通整合点で両方のデータベースをリカバーできます。

アプライ・イベントが設定され、ソース・データベースのバックアップまたはイメージ・コピー・ユーティリティのアクティビティが完了したらすぐに、ソース・データベースのすべてのアクティビティを再開できます。キャプチャー・プログラムも開始することができます。ターゲット・データベースのバックアップまたはイメージ・コピー・ユーティリティのアクティビティが完了したら、サブスクリプション・セットのスケジューリング・オプションを元の設定 (時間ベース、イベント・ベースまたはその両方) に戻すことができます。

アプライ・プログラム外部の CAPSTART ハンドシェーク・シグナルの実行

アプライ・プログラムがサブスクリプション・セットを使用して `CD` 表の変更をフェッチおよびアプライできるようにするためには、そのサブスクリプション・セット内の各サブスクリプション・セット・メンバーのキャプチャー・プログラムとアプライ・プログラム間でハンドシェーク (同期化された通信) が必要になります。

アプライ・プログラムは、CMD タイプ CAPSTART サブタイプのシグナルをシグナル (IBMSNAP_SIGNAL) 表に挿入することにより、ハンドシェークを開始します。アプライ・プログラムは、コンプリートとして定義されているターゲット表でサブスクリプション・メンバーのフル・リフレッシュを実行する前に、このシグナルを挿入します。

プロシージャー:

CAPSTART ハンドシェーク・シグナルを実行するには、次のようにします。

- 次の SQL ステートメントを使用して、IBMSNAP_SIGNAL 表に行を挿入してキャプチャー CMD タイプ CAPSTART サブタイプのシグナルを作成します。

```
INSERT INTO Schema.IBMSNAP_SIGNAL
    (signal_type,
     signal_subtype,
     signal_input_in,
     signal_state)
VALUES('CMD',
       'CAPSTART',
       mapid,
       'P');
```

この *mapid* は *Schema.IBMSNAP_PRUNCNTL* 表の *MAP_ID* 列の値であり、ハンドシェークを必要とするサブスクリプション・セット・メンバーの行に対応します。

注: 必要であれば、サブスクリプション・セット・メンバーのフル・リフレッシュを実行する前に、この SQL INSERT ステートメントを実行してください。

キャプチャー・プログラムは、データベース・リカバー・ログでこのレコードを検出した後で、このシグナル表のログ・レコードを処理します。キャプチャー・プログラムが反応するのは、この挿入に対応するコミット・レコードが検出された場合、つまりこのイベントがコミットされたことが検証された場合だけです。

キャプチャー・プログラムは、登録済み表の以前の使用に基づいて、関連する登録がすでにメモリー内に入れているかどうかをチェックします。登録済み表が使用されていない場合、キャプチャー・プログラムは関連する登録情報をメモリー内に読み込み、この登録済み表がアクティブであり、使用中であることを示す値を登録 (IBMSNAP_REGISTER) 表の中に設定します。

キャプチャー・プログラムは登録済み表が使用中であるかどうかに関係なく、*Schema.IBMSNAP_PRUNCNTL* 表の中の関連する行の *SYNCHPOINT* および *SYNCHTIME* の値を、この挿入されたシグナル行を含む DB2 作業単位のコミット・ログ・レコードからのログ・シーケンス番号、およびこの同じコミット・ログ・レコードからのタイム・スタンプにそれぞれ設定します。

キャプチャー・プログラムは、処理される挿入ログ・レコードに対応する IBMSNAP_SIGNAL 表の行の中の次の値を更新します。

- SIGNAL_STATE = 'C' (キャプチャー・プログラムにより受け取り済みで完全)

- SIGNAL_LSN = このシグナル行の挿入を含む DB2 の作業単位のコミット・ログ・レコードからのログ・シーケンス番号

CAPSTOP シグナルの実行

登録の変更のキャプチャーを手動で停止したい場合は、CAPSTOP シグナルを開始できます。このシグナルは、登録を非活動化する場合、または登録を除去する前に使用できます。

プロシージャー:

CAPSTOP シグナルを実行するには、次のようにします。

1. 次の SQL ステートメントを使用して、IBMSNAP_SIGNAL 表に行を挿入してキャプチャー CMD タイプ CAPSTOP サブタイプのシグナルを作成します。

```
INSERT INTO Schema.IBMSNAP_SIGNAL
    (signal_type,
     signal_subtype,
     signal_input_in,
     signal_state)
VALUES('CMD',
       'CAPSTOP',
       source_owner.source_table,
       'P');
```

この *Schema* はキャプチャー・スキーマの名前であり、*source_owner.source_table* は取り込まれた変更を必要としなくなった表の完全修飾名です。

キャプチャー・プログラムは、データベース・リカバー・ログでこのレコードを検出した後で、このシグナル表のログ・レコードを処理します。キャプチャー・プログラムが反応するのは、この挿入に対応するコミット・レコードが検出された場合、つまりこのイベントがコミットされたことが検証された場合だけです。

キャプチャー・プログラムは、登録済み表の以前の使用に基づいて、関連する登録がすでにメモリー内に入れられているかどうかをチェックします。登録済み表が現在使用中でない場合には、キャプチャー・プログラムは CAPSTOP シグナルを無視します。

登録済み表が使用されている場合は、キャプチャー・プログラムはこの登録に関連するメモリーをクリアし、登録を非活動化します (IBMSNAP_REGISTER 表の STATE 列を 'I' に設定することにより)。その後キャプチャー・プログラムはこの登録済み表について変更のキャプチャーを停止します。

キャプチャー・プログラムは、処理される挿入ログ・レコードに対応する IBMSNAP_SIGNAL 表の行の中の次の列値を更新します。

- SIGNAL_STATE = 'C' (キャプチャー・プログラムにより受け取り済みで完全)

- SIGNAL_LSN = このシグナル行の挿入を含む DB2 の作業単位のコミット・ログ・レコードからのログ・シーケンス番号

2. オプション: 登録を除去します。

別のシステムへのレプリケーション構成のプロモート

あるシステム (たとえばテスト・システム) で登録済みオブジェクトまたはサブスクリプション・セットを定義し、レプリケーション環境を別のシステム (たとえば実動システム) にコピーする必要がある場合には、レプリケーション・センターのプロモート関数を使用できます。これらの関数は、登録済みオブジェクトまたはサブスクリプション・セットをリバース・エンジニアリングして、適切なデータ定義言語 (DDL) およびデータ操作言語 (DML) 付きのスクリプト・ファイルを作成します。ソースの再登録またはサブスクリプション・セットの再作成を行う必要なしに、レプリケーション定義を別のデータベースにコピーできます。

たとえば、プロモート関数を使用して、リモートのターゲット・データベース用のサブスクリプション・セットを定義します。テスト環境でモデルになるターゲット・システムを定義した後、リモート・ターゲット・システム用のサブスクリプション・セットのスクリプトを作成できます (そして使用するアプライ修飾子の変更などを行うことができます)。こうしないと、中央のコントロール・ポイントからはサポートされません。

重要: プロモート関数はターゲット・システムとは接続せず、そのシステムのレプリケーション構成パラメーターの検証を行いません。

3 つのプロモート関数があります。

登録済み表のプロモート

この関数は、指定された表の登録情報をプロモートします。この関数は、オプションで、基本表、索引および表スペース定義のプロモートも行います。プロモートする複数の表に対して異なるキャプチャー・スキーマおよび異なるサーバー名を指定できます。また、プロモートされたソース表に関連する変更データ (CD) 表のスキーマ名も変更できます。

複数の登録済み表を同時にプロモートできます。ユーザーが指定した新しいスキーマ名は、プロモートされるすべての表に適用されます。

この関数は、DB2 Universal Database バージョン 8 で登録された表しかプロモートしません。

登録済みビューのプロモート

この関数は、指定されたビューの登録情報をプロモートします。この関数は、オプションで、基本ビュー、登録抹消された基本表 (ビューのベースである)、索引および表スペース定義のプロモートも行います。プロモートする複数のビューに対して異なるキャプチャー・スキーマおよび異なるサーバー名を指

定できます。また、プロモートされたソース・ビューに関連する CD ビュー、およびこれらの CD ビューのベースである CD 表のスキーマ名も変更できません。

複数の登録済みビューを同時にプロモートできます。ユーザーが指定した新しいスキーマ名は、プロモートされるすべてのビューに適用されます。

重要: プロモートするビューが登録済みソース表に基づくものである場合は、登録済み表のプロモート関数を使用して、登録済みのソース表を別個にプロモートする必要があります。これらの登録済みソース表が、登録済みビューのプロモート関数によって自動的にプロモートされることはありません。しかし、このビューのベースである、登録抹消された基本表は、必要であればこの関数によってプロモートされます。

サブスクリプション・セットのプロモート

この関数はサブスクリプション・セットをプロモートします。この関数により、データベース間でサブスクリプション・セット (すべてのサブスクリプション・セット・メンバーを含めて) をコピーできます。

サブスクリプション・セットのプロモート関数は、登録済み表のプロモート関数と一緒に使用する必要があります。

重要: プロモート関数を使用して、OS/400、UNIX、Windows、および z/OS オペレーティング・システム上の登録済みオブジェクトおよびサブスクリプション・セットをプロモートできます。プロモート関数は、同種のシステム間でのみレプリケーション定義をコピーします。たとえば、1 つの DB2 Universal Database for z/OS システムから別の DB2 Universal Database for z/OS システムへコピーします。

プロモート関数を使用して、DB2 以外のリレーショナル・データベースとの間でレプリケーション定義をコピーすることはできません。また、プロモート関数を使用して、OS/400 リモート・ジャーナルを含むレプリケーション定義をコピーすることもできません。

関連概念:

- 259 ページの 『第 14 章 DB2 レプリケーション・センターの使用』

関連作業:

- 43 ページの 『第 3 章 表およびビューをレプリケーション・ソースとして登録する』
- 69 ページの 『第 4 章 ソースのサブスクライブ』

関連資料:

- 513 ページの 『*schema.IBMSNAP_SIGNAL*』

第 13 章 レプリケーション環境の保守

この章では、データベース内にあり、DB2 レプリケーションに使用される、ソース・システム、コントロール表、およびターゲット表の保守の方法を説明します。

DB2 レプリケーションはデータベース・システムと共同で処理を行うため、既存のデータベース・アクティビティーの変更は最小限ですみます。しかし、システム全体の円滑な実行を保証し、潜在的な問題を回避するためには、レプリケーション環境の処理要件を判別し、これらの要件がデータベース・システムに影響を与える可能性を判別する必要があります。この章では、DB2 レプリケーションの 3 つの関数コンポーネントの保守の要件について説明します。

- 『ソース・システムの保守』
- 247 ページの『コントロール表の保守』
- 256 ページの『ターゲット表の保守』

ソース・システムの保守

レプリケーション・ソース・システムは、変更キャプチャー・メカニズム、レプリケーションするソース表 (OS/400 システムで使用されるリモート・ジャーナルを含む)、キャプチャー・プログラムから使用されるログ・データ、および DB2 以外のリレーショナル・データベース・ソースで使用されるキャプチャー・トリガーで構成されます。このセクションでは、ソース表とログ・ファイルを正しく保守する方法、そしてこれらの表およびファイルが常に DB2 レプリケーションからアクセス可能であるようにする方法を説明します。

ソース・オブジェクトの保守

レプリケーション・ソース・オブジェクトは、システム上の他のデータベース表およびビューと同じ保守を必要とする、データベース表およびビューです。これらのオブジェクトに関して、既存のユーティリティーおよび保守ルーチンを引き続き実行してください。

キャプチャー・プログラムおよびアプライ・プログラムが常に処理を進められるように、DB2 レプリケーションでのこれらのソース表の可用性を考慮する必要があります。DB2 レプリケーションは、ほとんどのレプリケーション処理においては、ソース表に直接アクセスする必要はありません。しかし、DB2 レプリケーションは、次の 2 つのアクションのいずれかが発生したときには、ソース表または表スペースにアクセスする必要があります。

- アプライ・プログラムがフル・リフレッシュを実行したとき。

- ロギング・マネージャーが圧縮されたログ・レコードを読み取ろうとしたとき (z/OS のみ)。

フル・リフレッシュ時にレプリケーション・アプライ・プログラムの処理が阻害されないように、ソース表に対して読み取りアクセスが可能であることを確認してください。また、z/OS の場合は、ソース表が圧縮されている場合に、DB2 が圧縮されたログ・レコードの表スペースに対してラッチを入手できるように、ユーティリティーがオンライン・モードで実行されることを確認してください。ユーティリティーおよび保守ルーチンが、データベース (または z/OS の場合は圧縮された表) をオフラインにする必要のある排他モードで実行される場合は、レプリケーションでソース・オブジェクトを使用することはできません。

ソース・ログおよびジャーナル・レシーバーの保守および保存

DB2 リカバー・ログには、DB2 リカバー機能の提供と、実行中のキャプチャー・プログラムへの情報の提供という 2 つの目的があります。DB2 リカバー、および DB2 レプリケーションの両方についてログ・データを保存する必要があります。また、このデータを削除する前に、キャプチャー・プログラムおよび DB2 が、ログまたはジャーナル・レシーバーのセットの処理を完全に終了していることを確実に確認する必要があります。

注: DB2 レプリケーションは、DB2 以外のリレーショナル・データベースからのログ・データは使用しません。

ログ・データの保存 (UNIX、Windows、z/OS)

ログ・データはログ・バッファー、アクティブ・ログ、またはアーカイブ・ログに入っています。キャプチャー・プログラムはウォーム・スタートのたびに、プログラムの停止後に作成されたすべての DB2 ログと、処理が完全に終了していないすべての DB2 ログを要求します。

UNIX および Windows の場合: キャプチャー・プログラムがアーカイブ・ログからデータをリトリブできるように、ユーザー出口アーカイブを使用するようにデータベースを構成する必要があります。

DB2 の実行時は常にキャプチャー・プログラムを実行するようしておけば、キャプチャー・プログラムは一般的に DB2 のリカバー・ログに合わせた最新の状態になります。DB2 がアクティブになっている間は常にキャプチャー・プログラムを実行するか、ログ・レコードを 1 週間以上保存するのであれば、既存のログ保存手順を使用し続けることができます。しかし、次の場合には、DB2 レプリケーションに合わせて、ログ保存手順を変更する必要があります。

- DB2 がバックアップを完了したら即時にログ・レコードを削除する場合、そしてこれらのログ・レコードが順方向リカバーに必要とされない場合。
- ストレージの制約があるため、アーカイブしたりリカバー・ログを頻繁に削除する必要がある場合。

プロシージャー:

キャプチャー・プログラムで使用するために保存する必要のあるログ・レコードと、削除できるログ・レコードを判別するには、次のようにします。

UNIX および Windows の場合:

1. 次の SQL ステートメントを実行し、再始動 (IBMSNAP_RESTART) 表から MIN_INFLIGHTSEQ 値を入手します。

```
SELECT MIN_INFLIGHTSEQ
FROM ASN.IBMSNAP_RESTART
WITH UR;
```

MIN_INFLIGHTSEQ 値が表示されます。(IBMSNAP_RESTART 表には行が 1 つしかありません。) MIN_INFLIGHTSEQ 値は、ビット・データ列の場合は char(10) であり、16 進の 20 文字のように見えます。たとえば、次のようになります。

```
00000000123456123456
```

MIN_INFLIGHTSEQ 値の最後の 12 文字を見てください。この例では次のようになっています。

```
123456123456
```

2. コマンド行から **db2 get db cfg** コマンドを入力し、アクティブ・ログ・ファイルのパスを入手します。たとえば、次のようになります。

```
db2 get db cfg for yourdbname
```

この *yourdbname* はデータベース名です。画面に表示された出力からアクティブ・ログ・ファイルのパスを確認します。たとえば、次のようになります。

```
Path to log files =C:¥DB2¥NODE0000¥SQL00001¥SQLLOGDIR¥
```

3. DB2 コマンド行から **db2flsn** コマンドを入力し、MIN_INFLIGHTSEQ 値の最後の 12 文字を入力します。たとえば、次のようになります。

```
C:¥DB2¥NODE0000¥SQL00001¥>db2flsn 123456123456
```

db2flsn コマンドを実行するには、SQLLOGCTL.LFH ファイルにアクセスする必要があります。このファイルは、アクティブ・ログ・ファイルのパスの 1 つ上のディレクトリー (C:¥DB2¥NODE0000¥SQL00001¥) にあります。

システムは、ログ・シーケンス番号により識別されるログ・レコードを含むファイルの名前をリトリブして表示します。たとえば、次のようになります。

Given LSN is contained in the log file S000123.LOG

4. リトリブされたログ・ファイルの経過日数を確認してください。

キャプチャー・プログラムは、任意の時点から再始動を実行するために、このログ・ファイル、およびこれより新しいログ・ファイルを必要とします。キャプチャー・プログラムの連続稼働を保証するために、このログ・ファイル、およびこれより新しいログ・ファイルは保存する必要がありますが、これより古いログは削除できます。

z/OS の場合、

1. 次の SQL ステートメントを実行し、再始動 (IBMSNAP_RESTART) 表から MIN_INFLIGHTSEQ 値を入手します。

```
SELECT MIN_INFLIGHTSEQ
FROM ASN.IBMSNAP_RESTART
WITH UR;
```

MIN_INFLIGHTSEQ 値が表示されます。(IBMSNAP_RESTART 表には行が 1 つしかありません。) たとえば、次のようになります。

```
0000555551F031230000
```

常に 0000 である、最初の 4 文字は無視してください。次の 12 文字は、アクティブ・ログ・シーケンス番号に相当します。(この 12 文字の値は、非データ共有環境では相対バイト・アドレス [RBA] であり、データ共有環境ではログ・レコード・シーケンス番号 [LRSN] です。)最後の 4 文字は、非データ共有環境では 0000 になります。これら最後の 4 文字は、データ共有環境ではメンバー ID に相当します。

2. DSNJU004 ユーティリティを使用してログ・マップ印刷ユーティリティを呼び出します。このユーティリティは、ブートストラップ・データ・セット (BSDS) に関する情報を表示します。

たとえば、次のようになります。

```
# ACTIVE LOG COPY 1 DATA SETS
# START RBA/TIME      END RBA/TIME          DATE  LTIME  DATA SET INFORMATION
#-----
# 555551F03000        555551F05FFF          1998.321 12:48  DSN=DSNC710.LOGCOPY1.DS02
#2001.57 15:46:32.2  2001.057 15:47:03.9  PASSWORD=(NULL)STATUS=TRUNCATED,REUSABLE
# 555551F06000        555551F09FFF          1998.321 12:49  DSN=DSNC710.LOGCOPY1.DS03
#2001.57 15:47:32.2  2001.057 15:48:12.9  PASSWORD=(NULL)STATUS=TRUNCATED,REUSABLE
```

3. MIN_INFLIGHTSEQ 値の 12 文字のアクティブ・ログ番号を、表示された各行の開始 RBA および対応する終了 RBA の範囲と比較します。
4. 自分の 12 文字のアクティブ・ログ番号の値が当てはまる行を探します。この例では次のようになっています。

```
# 555551F03000      000001F05FFF      1998.321 12:48 DSN=DSNC710.LOGCOPY1.DS02
#2001.57 15:46:32.2 2001.057 15:47:03.9 PASSWORD=(NULL)STATUS=TRUNCATED,REUSABLE
```

5. このアクティブ・ログ番号に対応するデータ・セット情報を確認します。この例では次のようになっています。

```
DSNC710.LOGCOPY1.DS02
```

6. このデータ・セットの日時を確認します。

キャプチャー・プログラムは、任意の時点から再始動を実行するために、このデータ・セット、およびこれより新しいデータ・セットを必要とします。キャプチャー・プログラムの連続稼働を保証するために、このデータ・セット、およびこれより新しいデータ・セットは保存する必要がありますが、これより古いデータ・セットは削除できます。

このログ・ファイルまたはデータ・セットの経過日数はベンチマークとを考えてください。キャプチャー・プログラムの連続稼働を保証するために、このファイル、およびこれより新しいログ・ファイルは保存する必要がありますが、これより古いログは削除できます。

推奨: キャプチャー・プログラムはログ・バッファーから直接ログ・レコードを読み取るため、最適なパフォーマンスを実現するために、DB2 がアクティブになっている間は常にキャプチャー・プログラムを実行してください。

ジャーナル・レシーバーの保存 (OS/400)

キャプチャー・プログラムから必要とされるすべてのジャーナル・レシーバーを保存することが重要です。**RESTART(*YES)** パラメーターを指定してキャプチャー・プログラムを再始動すると、キャプチャー・プログラムは以前に終了した場所から処理を続行し、1 つまたは複数のソース表により使用されるすべてのジャーナル・レシーバーを必要とします。

キャプチャー・プログラムが必要なジャーナル・レシーバーのすべてにアクセスできるように、DB2 DataPropagator for iSeries のインストール時に自動的に登録された、ジャーナル・レシーバー削除出口プログラムを使用してください。この出口プログラムは、ユーザー、またはユーザーのアプリケーション・プログラムの 1 つがジャーナル・レシーバーの削除を試みるたびに呼び出されます。この出口プログラムは、ジャーナル・レシーバーを削除できるかどうかを決定します。

推奨: **CHGJRN** または **CRTJRN** コマンドで **DLTRCV(*YES)** および **MNGRCV(*SYSTEM)** を指定し、ジャーナル・レシーバー削除出口ルーチンを使用し、ジャーナル管理をシステムに任せるようにします。

ジャーナル・レシーバーが 1 つまたは複数のソース表で使用されている場合、ジャーナル・レシーバー削除出口ルーチンは、キャプチャー・プログラムによってまだ処理され

ていないエントリーが、削除対象のレシーバーに含まれていないことを確認します。キャプチャー・プログラムがそのレシーバーのエントリーをさらに処理する必要がある場合には、出口プログラムはレシーバーの削除を承認しません。詳細については、39ページの『ジャーナルおよびジャーナル・レシーバーの管理 (OS/400)』を参照してください。

圧縮辞書の使用法 (z/OS)

DB2 の圧縮辞書ユーティリティを使用する場合は、キャプチャー・プログラムとの間でこれらのユーティリティの使用を調整する必要があります。

DB2 圧縮辞書の更新 (z/OS)

キャプチャー・プログラムがログ・レコードを要求した場合、DB2 は圧縮された表スペースに保管されている表のログ・レコードを圧縮解除する必要があります。DB2 は現行の圧縮辞書を使用して圧縮を解除します。圧縮辞書が一時的に使用できない場合、DB2 はキャプチャー・プログラムにエラーを戻します。キャプチャー・プログラムは何度か処理の続行を試みます。しかし、辞書が引き続き使用できない状態である場合は、キャプチャー・プログラムは ASN0011E メッセージを発行して終了します。また、圧縮辞書がもはや使用できない場合には、キャプチャー・プログラムは登録を非活動化します。これらの状況を防止するために、表の圧縮辞書に影響を与えるアクティビティを実行する前に、キャプチャー・プログラムが表のすべてのログ・レコードを処理するようにしてください。これらのアクティビティには次のものが含まれます。

- 圧縮設定を変更する表スペースの変更
- データ共有環境から非データ共有環境へのコピーを含めて、DSN1COPY を使用したサブシステム間での圧縮した表スペースのコピー
- 表スペースに対する REORG ユーティリティの実行

推奨: 再編成時に圧縮辞書の現行バージョンを維持するには、KEEPDICTIONARY=YES オプションを使用します。KEEPDICTIONARY=YES オプションを使用すると、既存のログ・レコードに対して辞書の互換性を維持できます。

しかし、新しい圧縮辞書を生成する場合は、現在実行中のアプリケーション、およびキャプチャー・プログラムに対して、REORG ユーティリティを次のように同期させてください。

1. 表を更新するすべてのアプリケーション・プログラムを静止させる。
2. キャプチャー・プログラムで、ログに保管されている表の更新をすべてキャプチャーする。
3. 圧縮表に対して REORG ユーティリティを使用し、新しい圧縮辞書を作成する。
4. アプリケーション・プログラムを再始動する。

DB2 圧縮辞書のラッチ (z/OS)

圧縮辞書の可用性についても考慮する必要があります。キャプチャー・プログラムが圧縮されたログ・レコードを読み取る際には、DB2 は辞書にアクセスするために、ソースの圧縮された表のラッチを取ります。DB2 のログ読み取りインターフェースがこのラッチを必要としたときに、ソース・システム上の圧縮された表スペースが STOPPED 状態にある場合は、キャプチャー・プログラムは停止します。その逆に、ソースの表スペースへの完全アクセスを必要とする、または表スペースが STOPPED 状態であることを必要とするユーティリティーは、キャプチャー・プログラムが辞書の読み取り中にラッチを保持しているために締め出されることがあります。

ラッチが使用できないために一時的なロックアウトが発生しないように、ソースの圧縮された表スペースを DB2 (またはベンダー) のユーティリティーで排他的に使用する必要があるときには、キャプチャー・プログラムを中断してください。

コントロール表の保守

DB2 レプリケーションは、ソース定義、サブスクリプション・セット定義、およびその他のレプリケーション固有のコントロール情報を保管するためにコントロール表を使用します。コントロール表のサイズは静的なものもあれば、データベースおよびレプリケーションのサイズの要件に従って動的に拡大 (または縮小) するものもあります。

次のコントロール表のサイズは、通常処理時に頻繁に変更されます。

- アプライ・ジョブ (IBMSNAP_APPLY_JOB) (OS/400 のみ)
- アプライ修飾子相互参照 (IBMSNAP_AUTHTKN) (OS/400 のみ)
- アプライ・トレース (IBMSNAP_APPLYTRACE)
- アプライ・トレール (IBMSNAP_APPLYTRAIL)
- キャプチャー・モニター (IBMSNAP_CAPMON)
- キャプチャー・トレース (IBMSNAP_CAPTRACE)
- 変更データ (*schema.CD_table*)
- 整合変更データ (*schema.target_table*)
- レプリケーション・アラート・モニター・アラート (IBMSNAP_ALERTS)
- レプリケーション・アラート・モニター・トレース (IBMSNAP_MONTRACE)
- レプリケーション・アラート・モニター・トレール (IBMSNAP_MONTRAIL)
- シグナル (IBMSNAP_SIGNAL)
- サブスクリプション・イベント (IBMSNAP_SUBS_EVENT)
- 作業単位 (IBMSNAP_UOW)

これらの動的なコントロール表のサイズおよび拡大により、システムのパフォーマンスが影響を受ける可能性があります。

このセクションでは、コントロール表に対して実行する必要がある保守のアクティビティについて説明します。

RUNSTATS ユーティリティーの使用方法 (UNIX、Windows、z/OS)

RUNSTATS ユーティリティーは、表および関連する索引の物理的特性に関する統計を更新します。以前に DB2 レプリケーションで使用していたのと同じ頻度で、既存の表に対して引き続き RUNSTATS ユーティリティーを実行する必要があります。しかし、表に含まれるデータの量が大量である場合には、変更データ (CD)、作業単位 (IBMSNAP_UOW)、およびその他の動的なコントロール表に対して RUNSTATS ユーティリティーを一度だけ実行してください。RUNSTATS が動的な表に関して有用な情報を報告するのは、これらの表が実動レベルで最大のサイズにある場合です。最適化プログラムはデータへのアクセスの最良のストラテジーを判断します。

パッケージおよびプランの再バインド (UNIX、Windows、z/OS)

DB2 レプリケーションのパッケージおよびプランの多くは、分離 UR (非コミット読み取り) を使用してバインドされます。パッケージおよびプランの自動再 BIND に使用される内部の保守プログラムは、キャプチャー・プログラムおよびアプライ・プログラムがカーソル固定などの標準オプションを使用してレプリケーション・パッケージを再 BIND した場合に、プログラム間で競合の問題を発生させる可能性がありますので、パッケージおよびプランの再 BIND が必要な場合は注意してください。DB2 レプリケーション・パッケージは、最適なシステム・パフォーマンスを維持するために、分離 UR にバインドしておく必要があります。

詳細については、30 ページの『レプリケーション・プログラムのセットアップ』を参照してください。

コントロール表の再編成

頻繁に更新される動的なコントロール表は定期的に再編成する必要があります。変更データ (CD) 表および作業単位 (IBMSNAP_UOW) 表は、変更キャプチャー時には多数の INSERTS を受け取り、プルーニング時には多数の DELETES を受け取ります。キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびアプライ・トレール (IBMSNAP_APPLYTRAIL) 表のサイズは、レプリケーションのソース表の更新率によって、大きく変化する可能性があります。

プロシージャー:

データのフラグメント化を防止し、スペースをレクラメーション処理するために、次の表の再編成方式のいずれかを使用してください。

REORG コマンド (UNIX、Windows)

PREFORMAT オプションを指定した REORG ユーティリティ (z/OS)

このユーティリティの PREFORMAT オプションは、キャプチャー・プログラムの挿入処理を迅速化します。

RGZPFM (物理ファイル・メンバーの再編成) コマンド (OS/400)

ENDDPRCAP コマンドで **RGZCTLB(*YES)** パラメーターを指定すると、キャプチャー・プログラムが終了したときに UOW 表およびアクティブな CD 表を再編成できます。(コマンド構文およびパラメーター記述の詳細については、413 ページの『ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』を参照。)

推奨: 次の動的なコントロール表は週に一度は再編成してください。

- CD 表
- IBMSNAP_ALERTS
- IBMSNAP_APPLYTRACE
- IBMSNAP_APPLYTRAIL
- IBMSNAP_CAPMON
- IBMSNAP_CAPTRACE
- IBMSNAP_MONTRAIL
- IBMSNAP_MONTRACE
- IBMSNAP_UOW

静的なコントロール表に対しては、未使用のスペースをレクラメーション処理するためのユーティリティを実行したり、頻繁に更新される最適化プログラムの統計を生成する必要はありません。

- アプライ・エンキュー (IBMSNAP_APPENQ)
- キャプチャー・エンキュー (IBMSNAP_CAPENQ) (UNIX、Windows、z/OS)
- キャプチャー・パラメーター (IBMSNAP_CAPPARMS)
- キャプチャー・スキーマ (IBMSNAP_CAPSCHEMAS)
- プルーニング・ロック (IBMSNAP_PRUNE_LOCK)
- プルーニング・セット (IBMSNAP_PRUNE_SET)
- プルーニング・コントロール (IBMSNAP_PRUNCNTL)
- レジスター (IBMSNAP_REGISTER)
- 登録拡張 (IBMSNAP_REG_EXT) (OS/400 のみ)
- 登録同期 (IBMSNAP_REG_SYNCH)
- レプリケーション・アラート・モニター条件 (IBMSNAP_CONDITIONS)
- レプリケーション・アラート・モニター連絡先 (IBMSNAP_CONTACTS)
- レプリケーション・アラート・モニター連絡先グループ (IBMSNAP_CONTACTGRP)

- レプリケーション・アラート・モニター・エンキュー (IBMSNAP_MONENQ)
- レプリケーション・アラート・モニター・グループ (IBMSNAP_GROUPS)
- レプリケーション・アラート・モニター・サーバー (IBMSNAP_MONSERVERS)
- 再始動 (IBMSNAP_RESTART)
- 順序付け (IBMSNAP_SEQTABLE)
- サブスクリプション列 (IBMSNAP_SUBS_COLS)
- サブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR)
- サブスクリプション・セット (IBMSNAP_SUBS_SET)
- サブスクリプション・ステートメント (IBMSNAP_SUBS_STMTS)

コントロール表のプルーニング

古いデータを除去し、システム・パフォーマンスを向上させるために、レプリケーション・コントロール表の定期的な整理が必要です。このセクションでは、コントロール表の整理のさまざまな方式と、これらの方式がシステムのパフォーマンスにどのような影響を与えるかについて説明します。

キャプチャー・プログラムによって保守される動的なコントロール表のプルーニング

次の動的なコントロール表の拡張をモニターし、使用可能な各種のプルーニング方式を考慮する必要があります。

- CD 表
- IBMSNAP_UOW (UOW)
- IBMSNAP_CAPMON
- IBMSNAP_CAPTRACE
- IBMSNAP_SIGNAL
- IBMSNAP_AUTHTKN (OS/400 のみ)

これらの表でプルーニングが一定インターバルで自動的に行われるように、キャプチャー・プログラムを設定できます。また、プルーニング・プロセスを立ち上げることにより、要求時に一度だけプルーニングを行うこともできます。ユーザーが次にプルーニング・コマンドを入力するまで、キャプチャー・プログラムはプルーニングを行いません。

推奨: これらのコントロール表の拡張を管理するために、自動プルーニングを使用することを考えてみてください。自動プルーニングにより、ストレージ・コストは最小化され、アプライ・プログラムの効率は向上し、また、これらの表の古いデータが定期的に除去されるため、ストレージのオーバーフローによるシステム障害のリスクが一般的に減少します。自動プルーニングを呼び出すには次のようにします。

- キャプチャー・プログラムの **autoprun** パラメーターを *y* に設定します (UNIX、Windows、z/OS)。
- キャプチャー・プログラムのパラメーター設定、**CLNUPITV(*IMMED)** または **CLNUPITV(*DELAYED)** を使用します (OS/400)。

自動ブルーニングでは、**prune_interval** 稼働パラメーター (UNIX、Windows、および z/OS の場合)、または **RETAIN** パラメーター (OS/400 の場合) を設定して、自動ブルーニング処理を発生させる頻度を指定します。

プロシージャ:

次の方法のいずれかを使用してブルーニングを開始します。

レプリケーション・センター

「キャプチャー・コントロール表の整理」ウィンドウを使用して、表の整理を一度行います。詳細については、レプリケーション・センターのヘルプを参照してください。

asncap システム・コマンドで **autoprun=y** を指定 (UNIX、Windows、z/OS)

自動ブルーニングを指定してキャプチャー・プログラムを始動するには、このコマンドを使用します。コマンド構文およびパラメーター記述の詳細については、334 ページの『asncap: キャプチャーの開始 (UNIX、Windows、z/OS)』を参照してください。

asnccmd システム・コマンドで **chgparms autoprun=y** を指定 (UNIX、Windows、z/OS)

実行中のキャプチャー・プログラムで自動ブルーニングを使用可能にするには、このコマンドを使用します。コマンド構文およびパラメーター記述の詳細については、340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

asnccmd システム・コマンドでブルーニング・パラメーターを使用 (UNIX、Windows、z/OS)

実行中のキャプチャー・プログラムでブルーニングを一度開始するには、このコマンドを使用します。コマンド構文およびパラメーター記述の詳細については、340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』を参照してください。

STRDPRCAP CLNUPITV(*IMMED) または STRDPRCAP CLNUPITV(*DELAYED) システム・コマンド (OS/400)

キャプチャー・プログラムを始動した後で、指定したインターバルで古い行を削除するには、これらのコマンドを使用します。パラメーター記述およびコマンド構文の詳細については、446 ページの『STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)』を参照してください。

OVRDPRCAPA PRUNE(*IMMED) または OVRDPRCAPA PRUNE(*DELAYED) システム・コマンド (OS/400)

実行中のキャプチャー・プログラムによるコントロール表の整理の方式を変更

するには、このコマンドを使用します。コマンド構文およびパラメーター記述の詳細については、426 ページの『OVRDPRCAPA: DPR キャプチャー属性のオーバーライド (OS/400)』を参照してください。

CD 表および UOW 表のブルーニング: キャプチャー・プログラムは、自動的に呼び出された場合も、要求時に実行する場合も、アプライ・プログラムから報告された進行に基づいて、各ブルーニング・サイクルで CD 表および UOW 表のブルーニングを行います。進行は、ブルーニング・セット (IBMSNAP_PRUNE_SET) 表の SYNCHPOINT 列で示されます。この通常 のブルーニングは、各 CD 表をサブスクライプするすべてのアプライ・プログラムを通じて最小の同期点値に基づいて、また UOW 表の場合は全体を通じて最小の同期点値に基づいて行われます。

しかし通常のブルーニングでは、関連するサブスクリプション・セットが非常にまれにしか実行されない場合は、CD 表および UOW 表の効果的なブルーニングを行うことはできません。関連するアプライ・プログラムの実行頻度を決定するとき、これらのアプライ・プログラムを停止するとき、そしてサブスクリプション・セットを比較的長い間非活動化するときには、ブルーニングの効率性を考慮してください。

サブスクリプション・セットを非常にまれにしか実行しないか、またはアプライ・プログラムを停止すると、使用している CD 表と UOW 表が非常に大きくなって、保持制限ブルーニングの対象として適格となる可能性があります。保持制限は、キャプチャー・プログラムの稼働パラメーターであり、このデフォルト値は 1 週間です。この値は、表の中の古いデータがどれくらいたつと保持制限ブルーニングに適格になるかを決定します。

サブスクリプション・セットが非アクティブになるか、まれにしか実行されないために、通常のブルーニング処理が使用禁止になると、データが非常に長い時間表の中に留まる可能性があります。このデータが DB2 の現行タイム・スタンプから保持制限値を引いた値よりも古くなると、保持制限ブルーニング処理は、このデータを表からブルーニングします。

保持制限ブルーニングを必要とするような条件の発生は回避してください。古いデータが累積されることにより、ストレージのオーバーフローが発生し、性能低下を招く可能性があります。詳細については、229 ページの『サブスクリプション・セットの非活動化』を参照してください。

推奨: すべてのサブスクリプション・セットに対して、アプライ・プログラムを少なくとも週に一度実行してください。

ソース・サーバーが多様なターゲット・システムに変更データを提供している場合、そして、各ターゲットの要件が大きく異なるものであり、また、ターゲットによっては、少数の登録済みソースに対してアプライ・プログラムがまれにしか実行されない場合は、複数のキャプチャー・プログラムを使用することを考えてみてください。複数のキャプチャー・プログラムを実行することにより、異なるキャプチャー・スキーマを使用

して、多様な処理要件を管理できます。1 つのキャプチャー・スキーマを使用して、サブスクリプション・セットの固有のタイミング要件によりまれにしか整理されない表を分離し、残りのソース表に対しては、別のキャプチャー・スキーマを使用できます。

キャプチャー・モニター表とキャプチャー・トレース表のブルーニング: キャプチャー・プログラムは、キャプチャー・プログラムの次の稼働パラメーターの値に基づいて、各ブルーニング・サイクルでキャプチャー・モニター (IBMSNAP_CAPMON) 表およびキャプチャー・トレース (IBMSNAP_CAPTRACE) 表のブルーニングを行います。

- 行が IBMSNAP_CAPMON 表の中に維持される時間の長さを示す **monitor_limit** パラメーター (UNIX、Windows、z/OS の場合) および **MONLMT** パラメーター (OS/400 の場合)
- 行が IBMSNAP_CAPTRACE 表の中に維持される時間の長さを示す **trace_limit** パラメーター (UNIX、Windows、z/OS の場合) および **TRCLMT** パラメーター (OS/400 の場合)

モニター限度パラメーターおよびトレース限度パラメーターのデフォルト値は両方ともに 1 週間です。IBMSNAP_CAPMON 表の中にキャプチャー・プログラムの待ち時間およびスループットのヒストリー情報をどのくらい長く保持したいか、そして IBMSNAP_CAPTRACE 表の中に監査およびトラブルシューティングの情報をどのくらい長く保持したいかによって、これらの値は変更できます。

シグナル表のブルーニング: シグナル (IBMSNAP_SIGNAL) 表もまた、各ブルーニング・サイクル中にブルーニングされます。シグナル行は、**SIGNAL_STATE** 列の値が C になるとブルーニングに適格になります。C という値は、シグナル情報が完成しており、キャプチャー・プログラム、またはその他のユーザー処理から必要とされることはないため、ブルーニングに適格であることを意味します。**SIGNAL_TIME** 列の値が、DB2 の現行タイム・スタンプから保持制限パラメーターの値を引いたものよりも古いシグナル行は、保持制限ブルーニングに適格です。

その他の動的コントロール表のブルーニング

キャプチャー・プログラムは、自分が保守している表に対してのみブルーニング操作を実行します。整合変更データ (CCD) 表はアプライ・プログラムによって保守されているため、キャプチャー・プログラムはこれらの表を自動的に整理しません。CCD 表のタイプによっては、ブルーニングの必要のないものもあります。コンプリートなコンデンス CCD 表はその場で更新されます。

コンプリートなコンデンス CCD 表からユーザーが除去することが考えられる唯一のレコードは、従属するターゲット表内にすでに複製済みであり、IBMSNAP_OPERATION 列の値が D (削除) であるものです。非コンデンス CCD 表は、ヒストリー・データを

含むものであり、非常に大きくなる可能性があります。このデータは監査のために保存しておく必要があるため、非コンデンス CCD 表に対してはブルーニング操作を実行しないでください。

しかし、内部 CCD 表のブルーニングは考慮に値します。これらの表は、システムの更新アクティビティが多い場合は、すぐに大きくなる可能性があります。内部 CCD 表からは最新の変更だけがフェッチされるため、古い行を保存しておく必要はありません。

内部 CCD 表のブルーニングを使用可能にするために、従属するすべてのターゲットへのアプライがすでに完了している変更データをブルーニングする、事後に実行される SQL ステートメントを、関連するサブスクリプション・セットに追加することを考えてみてください。また、自動スケジューリング機能に必要な SQL DELETE ステートメントを追加して、これらの表から行を削除することもできます。

アプライ・トレール (IBMSNAP_APPLYTRAIL) 表およびアプライ・トレース (IBMSNAP_APPLYTRACE) 表も手動で整理する必要があります。頻繁に実行されるアプライ・プログラムで複数のサブスクリプション・セットを定義および使用する場合、IBMSNAP_APPLYTRAIL 表は急速に拡張するため、頻繁なブルーニングが必要です。これらの表の拡張を管理する最良の方法は、サブスクリプション・セットの 1 つに、事後に実行される SQL ステートメントまたはプロシージャーを追加することです。また、自動スケジューリング機能に SQL DELETE ステートメントを追加することもできます。

レプリケーションの失敗の防止およびエラーからのリカバー

このセクションでは、コントロール表およびレプリケーション・データに影響を与える可能性のある、レプリケーションの失敗を防止し、失敗からリカバーする方法について説明します。

- キャプチャー・プログラムのコールド・スタートの防止
- コントロール表の入出力エラーおよび接続障害からのリカバー
- 脱落したソース・データのリトリート

キャプチャー・プログラムのコールド・スタートの防止

キャプチャー・プログラムのコールド・スタートは、プログラムを初めて開始するとき、またはコントロール表およびターゲット表のリフレッシュが必要な場合にだけ実行してください。キャプチャー・プログラムをコールド・スタートすると、レプリケーション環境内のすべてのターゲット表がリフレッシュされます。

UNIX、Windows または z/OS 上で、warmns、warmnsa、または warmsi オプションを指定してキャプチャー・プログラムを始動した場合、プログラムは再始動 (IBMSNAP_RESTART) 表の中の再始動点に基づいてログ・レコードのリトリートを試みます。キャプチャー・プログラムがログを検出できないと、キャプチャー・プログラムのウォーム・スタートは失敗します。ユーザーが warmns または warmsi オプションを

指定してキャプチャー・プログラムを始動すると、再始動プロセスは終了し、エラー・メッセージが発行されます。ユーザーが `warmrsa` オプションを指定してキャプチャー・プログラムを始動すると、再始動プロセスは停止し、キャプチャー・プログラムはコールド・スタートを実行し、CD 表および UOW 表の中のすべてのレコードを削除します。

キャプチャー・プログラムのコールド・スタートを防止するために、次の推奨事項を考慮してください。

- UNIX、Windows および z/OS オペレーティング・システムの場合は `warmrsa` を使用せずに、できるだけ `warmns` または `warmnsi` 始動モードを使用してキャプチャー・プログラムを再始動してください。`warmns` および `warmnsi` オプションを使用すると、再始動プロセスが失敗した場合の、キャプチャー・プログラムの自動コールド・スタートを防止できます。詳細については、334 ページの『`asncap`: キャプチャーの開始 (UNIX、Windows、z/OS)』を参照してください。
- OS/400 オペレーティング・システムの場合は、**RESTART(*YES)** パラメーターを指定してキャプチャー・プログラムを始動してください。キャプチャー・プログラムは、以前に終了したときに停止したポイントから処理を続けます。詳細については、446 ページの『`STRDPRCAP`: キャプチャー・プログラムの始動 (OS/400)』を参照してください。
- レプリケーション・アラート・モニター、またはその他のメカニズムを使用して、キャプチャー・プログラムからのヒストリー・データの状況を確認してください。次にこの情報を使用して、DB2 がアクティブのときには必ずキャプチャー・プログラムが実行されていることを検証できます。詳細については、177 ページの『第 11 章 レプリケーションのモニター』を参照してください。
- システム上に十分な DB2 ログ・データおよびジャーナル・レシーバーが保存され、このデータが DB2 から使用可能であることを確認してください。ログ保存の詳細は、242 ページの『ソース・ログおよびジャーナル・レシーバーの保守および保存』を参照してください。

コントロール表の入出力エラーおよび接続障害からのリカバー

コントロール表で入出力エラーまたは接続障害が発生した場合には、DB2 の標準リカバー手順を使用して、表の順方向リカバーを行います。表のデータが失われることはありません。

キャプチャー・プログラムは入出力エラーまたは接続障害を検出すると、適切なエラー・メッセージを発行してシャットダウンします。ユーザーはエラーを訂正してから、障害発生時点からキャプチャー・プログラムを再開できます。

アプライ・プログラムは、コントロール表に重大なエラーを検出するとシャットダウンします。アプライ・プログラムはターゲット表のエラー、またはネットワーク接続のエラーを検出すると、アプライ・トレール (`IBMSNAP_APPLYTRAIL`) 表にエラーを書き込み、処理を続行します。

脱落したソース・データのリトリブ

ソース表で障害時点まで順方向リカバーが行われると、DB2 レプリケーションは正常に進行します。表がリカバーされると、キャプチャー・プログラムは表のデータ変更の収集を続行します。

しかし、キャプチャー・プログラムおよびアプライ・プログラムは、読み取り専用ターゲット表のポイント・イン・タイム・リカバーは検出しません。ユーザーがソース表をリカバーした場合、アプライ・プログラムが、ソース表にもはや存在していない変更をターゲット表に複製した可能性があるため、ユーザーがターゲット表を同じ論理ポイント・イン・タイムまで戻せない場合は、ソース表とターゲット表の間で不整合が生じる可能性があります。

複数レベルでレプリケーションが行われる場合は、この状況がより複雑なものになります。各レベル間のリカバー点を照合するためのメカニズムを提供するか、リカバー方式としてフル・リフレッシュを使用する必要があります。

分散リカバー点の詳細は、232 ページの『データベース・アプリケーション・イベントとレプリケーション・イベントの調整』を参照してください。

ターゲット表の保守

ターゲット・サーバー上の表は、データベース・システムの他の表を保守するのと同じ方法で保守してください。ターゲット表が既存のデータベース表である場合も、DB2 レプリケーションにより自動的に生成されるように指定された表である場合も、これらの表に対して現在のバックアップおよび保守のルーチンを使用してください。

重要: ユーティリティーを実行するためにターゲット表をオフラインにする前に、アプライ・プログラムを非活動化してください。

関連概念:

- 473 ページの 『第 22 章 DB2 レプリケーション・コンポーネントの通信方法』

関連作業:

- 3 ページの 『第 1 章 レプリケーションの計画』
- 17 ページの 『第 2 章 レプリケーションの設定』

第 2 部 レプリケーション・センター

本書の第 2 部は、以下の章で構成されています。

259 ページの『第 14 章 DB2 レプリケーション・センターの使用』では、レプリケーション・センターについて説明します。

287 ページの『第 15 章 基本的なデータ・レプリケーションのシナリオ: DB2 for Windows』では、レプリケーション・センターを使用して、サンプル・データに簡単なレプリケーション・シナリオを実行する方法について説明します。

第 14 章 DB2 レプリケーション・センターの使用

DB2® レプリケーション・センターは、レプリケーション環境をセットアップおよび管理するため、また、キャプチャー・プログラム、アプライ・プログラム、およびレプリケーション・アラート・モニター・プログラムを実行するために使用できるユーザー・インターフェース・ツールです。レプリケーション・センターを使用して、次のような管理タスクを実行することができます。

- レプリケーション・コントロール表の作成
- レプリケーション・ソースの登録
- サブスクリプション・セットを作成し、そのセットにサブスクリプション・セット・メンバーを追加
- キャプチャー・プログラムの操作
- アプライ・プログラムの操作
- レプリケーション処理のモニター

レプリケーション・センターにはランチパッドもあり、これを使用して DB2 レプリケーション環境のセットアップに必要な基本関数を実行することができます。ランチパッドは、さまざまなステップがお互いにどのように関係しているかをグラフィカルに表示します。

レプリケーション・センターを使用して、DB2 から DB2 へのレプリケーション環境や、DB2 と DB2 以外のリレーショナル・データベース間のレプリケーションをセットアップすることができます。DB2 レプリケーション・センターは、DB2 コントロール・センターのツールのセットの一部です。レプリケーション・センターの詳細なタスク情報については、オンライン・ヘルプを参照してください。

本章では、以下のタスクについて説明します。

レプリケーション・センターの使用

- 261 ページの『DB2 レプリケーション・センターの前提条件』を参照。レプリケーション・センターの前提条件は、DB2 コントロール・センターの前提条件と非常に似ています。
- 261 ページの『DB2 レプリケーション・センターの始動』を参照。レプリケーション・センターはいくつかの方法で始動することができます。
- 263 ページの『レプリケーション・センター・ランチパッドの使用』を参照。ランチパッドの使用はオプションですが、初めてのユーザーには非常に役に立ちます。

レプリケーション・センターのセットアップ

- 264 ページの『レプリケーション・センターのユーザー ID およびパスワードの管理』を参照。レプリケーション・センターがデータベースに接続し、システムにログオンするために使用するパスワードを保守することができます。
- 265 ページの『レプリケーション・プロファイルの作成』を参照。プロファイルの作成はオプションですが、大規模なレプリケーション環境を管理する場合はとても便利です。
- 268 ページの『レプリケーション・コントロール表の作成』を参照。レプリケーション・コントロール・サーバーとして働く各データベース内に、コントロール表を作成する必要があります。
- 271 ページの『レプリケーション・センターへのサーバーの追加』を参照。ユーザーがレプリケーション・コントロール表を作成すると、サーバーは自動的にレプリケーション・センターに追加されます。管理したいサーバーだけを追加することにより、レプリケーション環境のビューをカスタマイズすることができます。
- 272 ページの『データベースの変更キャプチャーの使用可能化 (UNIX および Windows)』を参照。UNIX[®] および Windows[®] システム上のキャプチャー・コントロール・サーバーが、変更をキャプチャーし、データベース・バックアップを開始できるようにする必要があります。

レプリケーション環境の定義

- 273 ページの『ソースの登録』を参照。表またはビューをレプリケーション・ソースとして登録できます。
- 275 ページの『サブスクリプション・セットの作成』を参照。空のセットを作成し、いつでもここにサブスクリプション・セット・メンバーを追加するか、またはサブスクリプション・セットの作成時にサブスクリプション・セット・メンバーを作成することができます。

レプリケーション環境の保守

- 279 ページの『サブスクリプション・セットの活動化または非活動化』を参照。どのサブスクリプション・セットでも、一時的に、または永続的に、非活動化または活動化することができます。
- 279 ページの『レプリケーション・オブジェクトのプロモート』を参照。テスト環境から実稼働環境へ、表の登録およびサブスクリプション・セットをプロモートすることができます。
- 281 ページの『ターゲット表の強制フル・リフレッシュ』を参照。アプリ・プログラムがサブスクリプション・セットのフル・リフレッシュを実行する時期をコントロールすることができます。

- 283 ページの『レプリケーション定義の除去または削除』を参照。レプリケーション・センターからレプリケーション・オブジェクトを除去し、レプリケーション・コントロール・サーバーからレプリケーション定義を削除することができます。

レプリケーション環境の操作

- 283 ページの『キャプチャー・プログラムの操作』を参照。ネットワーク内のどのサーバーからでも、キャプチャー・プログラムを始動および停止することができます。また、キャプチャー・プログラム用のその他の多くの操作も実行することができます。
- 284 ページの『アプライ・プログラムの操作』を参照。ネットワーク内のどのサーバーからでも、アプライ・プログラムを始動および停止することができます。また、アプライ・プログラム用のその他の多くの操作も実行することができます。
- 285 ページの『レプリケーション・アラート・モニターの操作』を参照。レプリケーション・アクティビティをモニターするためのアラート条件を定義することができます。

DB2 レプリケーション・センターの前提条件

レプリケーション・センターを実行するには、システムに正しい Java™ Runtime Environment (JRE) がインストールされていなければなりません。DB2 のインストール時に、JRE をインストールするオプションがあります。JRE をインストールしないことを選択した場合は、システムに Java 2 Runtime Environment または Java 2 Software Development Kit のどちらかのバージョン 1.3 があることを確認してください。

レプリケーション・センターを使用して、リモート・システム上のキャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニター・プログラムを操作する場合は、レプリケーション・センターを実行するローカル・システムおよび、キャプチャー・プログラムまたはアプライ・プログラムを実行するそれぞれのリモート DB2 システム上で、DB2 Administration Server (DAS) が実行されていることを確認してください。

DB2 レプリケーション・センターの始動

レプリケーション・センターは、UNIX または Windows 操作環境用の通常の DB2 インストールの一部としてインストールされます。カスタム・インストールを実行する場合は、汎用管理ツール (General Administration Tools) コンポーネントを選択してレプリケーション・センターをインストールする必要があります。

レプリケーション・センターを始動するには、コマンド・ウィンドウに **db2rc** コマンドを入力します。

Windows システムでは、以下に示すように、Windows の **Start** メニューを使用してレプリケーション・センターを始動することもできます。

1. 「スタート」をクリックします。
2. 「プログラム」を選択します。
3. 「IBM DB2」を選択します。
4. 「汎用管理ツール」を選択します。
5. 「レプリケーション・センター」をクリックします。

すでに DB2 コントロール・センターを実行中の場合は、「ツール」メニューから「レプリケーション・センター」を選択するか、またはレプリケーション・センターのアイコンをクリックして、レプリケーション・センターを始動することができます。

図 7はレプリケーション・センターを示しています。

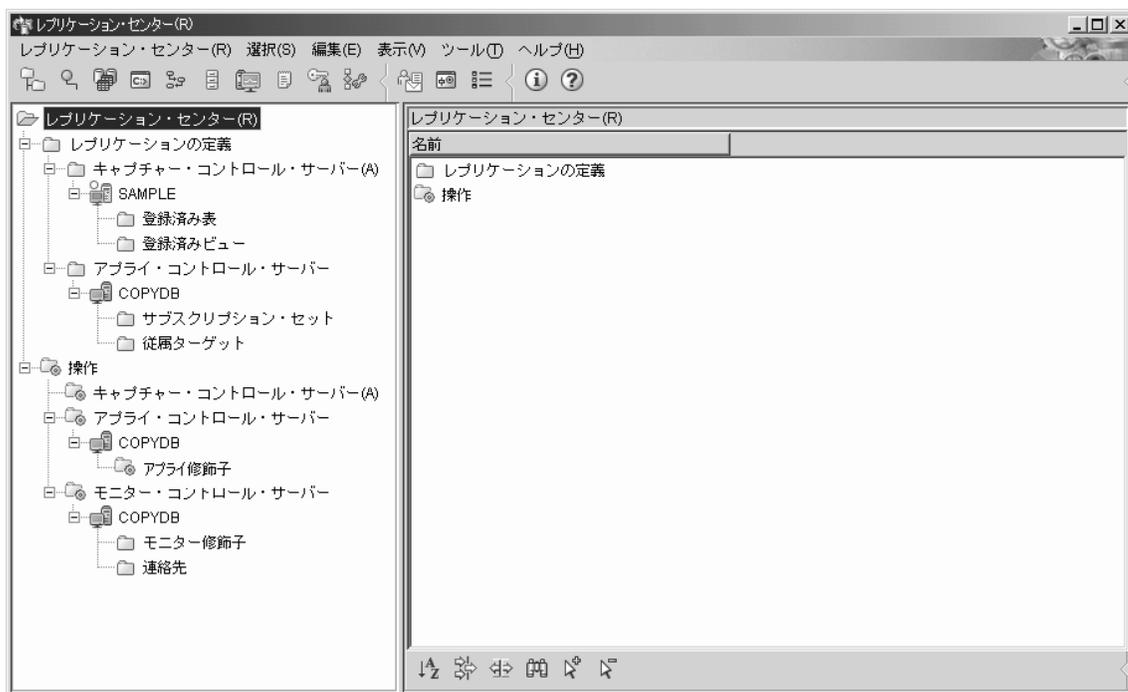


図 7. DB2 レプリケーション・センター

レプリケーション・センター・ランチパッドの使用

最初にレプリケーション・センターを始動すると、レプリケーション・センター・ランチパッドが表示されます。これを使用して、DB2 レプリケーション環境のセットアップに必要な基本関数を実行することができます。またランチパッドは、さまざまなステップがお互いにどのように関係しているかをグラフィカルに表示します。

ランチパッドは、レプリケーション・センターのよく使用される関数のほとんどにアクセスできますが、これらの関数にはレプリケーション・センターのナビゲーターからもアクセスできます。レプリケーション・センター・ナビゲーターを使用して、ランチパッドを使用して作成したものは、何でも表示または操作することができます。レプリケーション・センターには、ランチパッドからはアクセスできない、他の多くの、より高度な関数があります。

ランチパッドから、以下のタスクを実行することができます。

- キャプチャー・コントロール表の作成
このオプションは「キャプチャー・コントロール表の作成」ウィンドウをオープンし、ここから、キャプチャー・プログラムのために必要なレプリケーション・コントロール表を、特定のデータベースに作成することができます。
- ソース表の登録
このオプションは「表の登録」ウィンドウをオープンし、ここから、登録したいソース表の登録情報（ソース列、CD 表の情報など）を定義することができます。
- アプライ・コントロール表の作成
このオプションは「アプライ・コントロール表の作成」ウィンドウをオープンし、ここから、アプライ・プログラムのために必要なレプリケーション・コントロール表を、特定のデータベース内に作成することができます。
- サブスクリプション・セットの作成
このオプションは「サブスクリプション・セットの作成」ウィンドウをオープンし、ここからサブスクリプション・セット情報を定義することができます。その情報とは、キャプチャー、ターゲットおよびアプライ・コントロール・サーバー、ソースからターゲットへのマッピング、各サブスクリプション・セット・メンバーのプロパティ、セットのスケジュール、およびセット用の SQL ステートメントなどです。
- キャプチャー・プログラムの始動
このオプションは「キャプチャーの開始」ウィンドウをオープンし、ここからキャプチャー・プログラムを始動し、始動パラメーターを指定することができます。
- アプライ・プログラムの始動
このオプションは「アプライの開始」ウィンドウをオープンし、ここからアプライ・プログラムを始動し、始動パラメーターを指定することができます。

推奨事項: ランチパッドではこれらのタスクを順番に実行する必要はありませんが、DB2 レプリケーションに慣れていないユーザーは、ランチパッドに示されている順序に

従うことをお勧めします。必要なレプリケーションやデータベース・オブジェクトがすでに存在する場合は、ランチパッド内のステップを飛ばしたり、繰り返すこともできます。ランチパッドまたはレプリケーション・センター・ナビゲーターを使用して、必要なレプリケーションとデータベース・オブジェクトを作成してください。

「レプリケーション・センター」メニューから「ランチパッド」を選択するか、またはナビゲーター内の「レプリケーション・センター」フォルダーを右クリックし、「開始ランチパッド」を選択すれば、いつでもランチパッドを使用することができます。

レプリケーション・センターのユーザー ID およびパスワードの管理

レプリケーション・センターは、多くのデータベース・サーバー（ソース・サーバー、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、モニター・コントロール・サーバー、およびターゲット・サーバー）に接続できる必要があります。またレプリケーション・センターは、キャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニターを実行するシステムにも接続できる必要があります。リモートのデータベースおよびシステムはすべて、データベースに接続し、各システムにログオンするための有効なユーザー ID とパスワードが必要です。レプリケーション・センターは、各ユーザー ID とパスワードの指定が一度ですむようにしているので、レプリケーション・センターがリモート・データベースに接続するたびに、または各リモート・システムにログオンするたびに、プロンプトを求めることはありません。

デフォルトでは、レプリケーション・センターはユーザー ID とパスワード情報をメタデータ・ファイルに保管します。パスワードはこのファイル内で暗号化されないため、レプリケーション・センターがパスワードをメモリー内にのみ保持し、このファイルにはパスワードを保管しないように指定することができます。DB2 でパスワードが変更された場合は、レプリケーション・センター内のパスワードも変更し、両方で同じパスワードを使用する必要があります。レプリケーション・センターは、アプライ・プログラム、レプリケーション・アナライザー、またはレプリケーション・アラート・モニターとパスワード情報を共有しません。

レプリケーション・センターのユーザー ID およびパスワードを管理するには、次のようになります。

1. 「レプリケーション・センター」フォルダーを右クリックします。
2. 「レプリケーション・センターのパスワード管理」を選択します。

「レプリケーション・センターのパスワード管理」ウィンドウで、次のどのタスクでも実行することができます。

- データベース別名またはシステム名のための、新規ユーザー ID とパスワードの対を追加する。
- 既存のユーザー ID のパスワードを変更する。
- ユーザー ID を除去する。

- ユーザー ID とパスワードを使用して、選択したデータベース別名への接続または、選択したシステムへのログオンをテストする。

レプリケーション・プロファイルの作成

レプリケーションのセットアップの一部として、多くの場合複数のデータベース内に、レプリケーション・コントロール表を作成します。また、多くのソース表およびビューを登録し、そのすべてが CD 表を持ちます。さらにサブスクリプション・セットを作成する一部として、多くのターゲット表を定義します。これらのレプリケーション・オブジェクトのそれぞれについて、特定の命名規則を使用し、共通の属性を持たせることとなります (たとえば、CD 表用のすべての表スペースが同じページ・サイズを使用する可能性があります)。レプリケーション・センターでは、レプリケーション・オブジェクトを作成するたびにこれらの共通の定義を指定しなくてもすむように、これらの命名規則および共通の定義を反映するプロファイルがユーザーが作成できるようにしています。以下のレプリケーション・オブジェクト用のプロファイルを作成することができます。

- レプリケーション・コントロール表
- レプリケーション・ソース・オブジェクト (CD 表)
- レプリケーション・ターゲット・オブジェクト

各オブジェクト・プロファイルに、CD 表、索引、および表スペースといったデータベース・オブジェクト用の命名規則を指定し、これらのオブジェクト用の共通属性 (ページ・サイズやバッファ・プールなど) を指定します。各プロファイルに指定する値は、「コントロール表の作成」ウィンドウ、「表の登録」ウィンドウ、「ビューの登録」ウィンドウ、または「サブスクリプション・セットの作成」ウィンドウに表示されるデフォルト値になります。これらのデフォルト値は、特定のレプリケーション・オブジェクトの作成時にオーバーライドするか、または「OK」をクリックして、プロファイルに定義された値を受け入れることができます。

コントロール表プロファイルの作成

各レプリケーション・コントロール表 (たとえば、登録表 `IBMSNAP_REGISTER`) について、プロファイルに表スペース情報および索引情報を定義することができます。デフォルトにより、レプリケーション・センターはパフォーマンスを最適化するため、レプリケーション・コントロール表と一緒に表スペース内にまとめます。これらの表スペース用に、レプリケーション・センターが表スペースの作成時に使用する命名規則を定義するか、またはすでに存在する表スペースを指定することができます。また、コントロール表のための、その他のオペレーティング・システム特有の表スペース情報を定義することもできます。コントロール表の多くは、1 つまたは複数の索引も必要とします。レプリケーション・センターがこれらの索引を作成する時に使用する命名規則を定義するか、またはすでに存在する索引を指定することができます。

DB2 がサポートする各タイプのオペレーティング・システム用の、固有のコントロール表プロファイルを定義することができます。また、DB2 レプリケーションがサポートする DB2 以外のデータベースの各タイプ用のプロファイルを定義することもできます。レプリケーション・センターは OS/400® システム用のコントロール表プロファイルを提供しませんが、その理由は、DB2 DataPropagator™ for iSeries のインストール時にレプリケーション・コントロール表が作成されるためです。

コントロール表プロファイルを作成するには、次のようにします。

1. 「レプリケーション定義」フォルダーを右クリックします。
2. 「コントロール表プロファイルの管理」を選択します。
「コントロール表プロファイルの管理」ウィンドウで、次のようにします。
 - a. プロファイルを作成する、オペレーティング・システムまたは DB2 以外のデータベース環境を選択します。
 - b. リストからレプリケーション・コントロール表を選択します。
 - c. 選択したコントロール表の表スペースと索引の特性を定義します。
別の表スペースを作成するコントロール表を必ずすべて選択し、その特性を定義してください。
 - d. 特定のオペレーティング・システム・プラットフォーム用の、または DB2 以外のリレーショナル・データベース・システム用の、コントロール表をすべて定義したら、「適用」をクリックします。「クローズ」をクリックし、「コントロール表プロファイルの管理」ウィンドウをクローズします。

ソース・オブジェクト・プロファイルの作成

それぞれの DB2 レプリケーション・ソース・オブジェクト (表またはビュー) ごとに、キャプチャー・プログラムは CD 表を必要とします。ユーザーはソース・オブジェクトを登録する時に、CD 表と CD 表用の索引の両方について、名前と特性を指定します。ソース・オブジェクト用のプロファイルを作成すれば、特定のソース・データベースから登録するすべてのソースについて、共通の特性を定義することができます。これらの共通の特性を使用して、多くの表またはビューを 1 つのアクションの一部として登録することができます。

レプリケーション・センターが CD 表の作成時に使用する命名規則、CD 表用の表スペース、および CD 表用の索引を定義することができます。また、ユーザーが指定した接頭部と接尾部をレプリケーション・センターが反映した後、これらのオブジェクトの名前がオペレーティング・システム特有の長さ制限 (たとえば、UNIX および Windows データベースの場合は 128 文字) を超えた場合に、オブジェクト名を切り捨てるための規則を定義することもできます。たとえば、CD 表の名前を “CD_sourcetablename” (sourcetablename は、登録されるソース表ごとに異なる) とし、これを “CD_repltablespace” という名前の表スペースに入れるプロファイルを作成することができます。

キャプチャー・コントロール・サーバー用のソース・オブジェクト・プロファイルを作成する前に、そのサーバーをレプリケーション・センターに追加する必要があります。

ソース・オブジェクト・プロファイルを作成するには、「ソース・オブジェクト・プロファイルの管理」ウィンドウを使用します。このウィンドウは次の 2 つの方法でオープンすることができます。

- 「**レプリケーション定義**」フォルダーからウィンドウをオープンする場合は、次のようにします。
 1. 「**レプリケーション定義**」フォルダーを右クリックします。
 2. 「**ソース・オブジェクト・プロファイルの管理**」を選択します。
「ソース・オブジェクト・プロファイルの管理」ウィンドウで、プロファイルを作成するソース・サーバーを選択します。
- ソース・サーバーからウィンドウをオープンする場合は、次のようにします。
 1. 「**レプリケーション定義**」フォルダーを展開します。
 2. 「**キャプチャー・コントロール・サーバー**」フォルダーを展開します。
 3. ソース・サーバーを右クリックして、「**ソース・オブジェクト・プロファイルの管理**」を選択します。
この場合、「ソース・オブジェクト・プロファイルの管理」ウィンドウで、プロファイルを作成するソース・サーバーを選択する必要はありません。

「ソース・オブジェクト・プロファイルの管理」ウィンドウで、CD 表の特性、CD 表用の表スペース、および CD 表用の索引を定義します。これらのオブジェクトごとに切り捨て規則を選択することもできます。

ターゲット・オブジェクト・プロファイルの作成

サブスクリプション・セット・メンバーの作成時に、ソース・オブジェクト (表、ビュー、またはニックネーム) とターゲット表間のレプリケーションのマッピングを定義します。ターゲット表がまだ存在しない場合は、ターゲット表およびターゲット表用の索引の両方について、名前と特性を指定します。ターゲット・オブジェクト用のプロファイルを作成すれば、特定のターゲット・データベース内のすべてのターゲット表について、共通の特性を定義することができます。

レプリケーション・センターがターゲット表の作成時に使用する命名規則、ターゲット表用の表スペース、およびターゲット表用の索引を定義することができます。また、ユーザーが指定した接頭部と接尾部をレプリケーション・センターが反映した後、これらのオブジェクトの名前がオペレーティング・システム特有の長さ制限 (たとえば、UNIX および Windows データベースの場合は 128 文字) を超えた場合に、オブジェクト名を切り捨てるための規則を定義することもできます。たとえば、ターゲット表の名前を “*TGsourcetablename*” (*sourcetablename* は、登録されるソース表ごとに異なる) とし、これを “*TS_targettablename*” (*targettablename* は、ターゲット表ごとに異なる) という名前の表スペースに入れるプロファイルを作成することができます。

ターゲット・サーバー用のターゲット・オブジェクト・プロファイルを作成するには、その前にターゲット・サーバーをローカルの DB2 データベースにカタログしておく必要があります。ただし、ターゲット・サーバーをレプリケーション・センターに、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、またはモニター・コントロール・サーバーとして追加する必要はありません。

ターゲット・オブジェクト・プロファイルを作成するには、次のようにします。

1. 「**レプリケーション定義**」フォルダーを右クリックします。
2. 「**ターゲット・オブジェクト・プロファイルの管理**」を選択します。
3. 「サーバーの選択」ウィンドウで、ターゲット表プロファイルを作成するデータベース・サーバーを選択し、「**OK**」をクリックします。

「ターゲット・オブジェクト・プロファイルの管理」ウィンドウで、プロファイルを作成するターゲット・サーバーを選択します。ターゲット表の特性、ターゲット表用の表スペース、およびターゲット表用の索引を定義します。これらのオブジェクトごとに切り捨て規則を選択することもできます。

レプリケーション・コントロール表の作成

レプリケーション・コントロール表には、レプリケーション環境のセットアップに関するすべての情報が保管され、キャプチャー・プログラムおよびアプライ・プログラムがレプリケーション中に使用する操作情報が保管されます。データベース・サーバーをレプリケーション・センターに追加する前に、レプリケーション・コントロール表を作成する必要があります。レプリケーション・コントロール表を特定のデータベース内に作成すると、レプリケーション・センターはそのデータベース・サーバーをレプリケーション・センターのナビゲーターに自動的に追加します。

レプリケーション・センターを使用して OS/400 システム用のレプリケーション・コントロール表を作成することはできませんが、その理由は、DB2 DataPropagator for iSeries のインストール時にレプリケーション・コントロール表が作成されるためです。コントロール表を再作成する場合、または代替のキャプチャー・スキーマを使用してコントロール表を作成する場合は、OS/400 **CRTDPRTBL** コマンドを使用してください。

同じデータベース内にすべてのレプリケーション・コントロール表を作成すれば、キャプチャー・コントロール・サーバーはアプライ・コントロール・サーバーとしての役割も果たします。同様に、キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーは、モニター・コントロール・サーバーとしての役割も果たします。

キャプチャー・コントロール表の作成

キャプチャー・コントロール・サーバー用のコントロール表は、次の 2 つの方法で作成することができます。

- 「コントロール表の作成 - クイック」ウィンドウをオープンします。
 1. 「レプリケーション定義」フォルダーを展開します。
 2. 「キャプチャー・コントロール・サーバー」フォルダーを右クリックし、「キャプチャー・コントロール表の作成」→「クイック」を選択します。
 3. 「サーバーの選択」ウィンドウで、キャプチャー・コントロール表を作成するサーバーを選択し、「OK」をクリックします。

「コントロール表の作成 - クイック」ウィンドウでは、ユーザーのレプリケーション環境についていくつか簡単な質問をします。レプリケーション・センターはユーザーの答えに基づいて、該当のサイズの特定の表スペース内にレプリケーション・コントロール表を作成し、さらにパフォーマンスを最適にするため、これらの表スペース内のコントロール表をグループにまとめます。
- 「キャプチャー・コントロール表の作成」ウィンドウをオープンします。
 1. 「レプリケーション定義」フォルダーを展開します。
 2. 「キャプチャー・コントロール・サーバー」フォルダーを右クリックし、「キャプチャー・コントロール表の作成」→「カスタム」を選択します。
 3. 「サーバーの選択」ウィンドウで、キャプチャー・コントロール表を作成するサーバーを選択し、「OK」をクリックします。このサーバーは、キャプチャー・プログラムを実行するサーバーです。データベースが、DB2 以外のリレーショナル・ソース用のゲートウェイとして働く連合データベースである場合は、その連合データベースを選択し、そのデータベースからサーバー・マッピングを検索し、その連合データベース用に定義された DB2 以外のリレーショナル・サーバーのリストを表示します。

「キャプチャー・コントロール表の作成」ウィンドウで、各コントロール表の特性を定義します。

 - a. リストからレプリケーション・コントロール表を選択します。
 - b. 選択したコントロール表の表スペースと索引の特性を定義します。

別の表スペースを作成するコントロール表を必ずすべて選択し、そのコントロール表の表スペースと索引の特性を定義してください。

ヒント: 選択したデータベースのオペレーティング・システム・プラットフォーム用のコントロール表プロファイルを作成した場合は、そのプロファイルから設定値を受け入れるか、あるいはオーバーライドすることができます。

キャプチャー・コントロール表用に固有のスキーマ名を指定することができます。デフォルトは ASN です。選択したデータベースに複数のキャプチャー・プログラムを実行する場合は、別のスキーマ名が必要です。
 - c. すべてのコントロール表を定義したら、「OK」をクリックします。

アプライ・コントロール表の作成

アプライ・コントロール・サーバー用のコントロール表は、次の 2 つの方法で作成することができます。

- 「コントロール表の作成 - クイック」ウィンドウをオープンします。
 1. 「レプリケーション定義」フォルダーを展開します。
 2. 「アプライ・コントロール・サーバー」フォルダーを右クリックし、「アプライ・コントロール表の作成」→「クイック」を選択します。
 3. 「サーバーの選択」ウィンドウで、アプライ・コントロール表を作成するサーバーを選択し、「OK」をクリックします。

「コントロール表の作成 - クイック」ウィンドウでは、ユーザーのレプリケーション環境についていくつか簡単な質問をします。レプリケーション・センターはユーザーの答えに基づいて、該当のサイズの特定の表スペース内にレプリケーション・コントロール表を作成し、さらにパフォーマンスを最適にするため、これらの表スペース内のコントロール表をグループにまとめます。

- 「アプライ・コントロール表の作成」ウィンドウをオープンします。
 1. 「レプリケーション定義」フォルダーを展開します。
 2. 「アプライ・コントロール・サーバー」フォルダーを右クリックし、「キャプチャー・コントロール表の作成」→「カスタム」を選択します。
 3. 「サーバーの選択」ウィンドウで、アプライ・コントロール表を作成するサーバーを選択し、「OK」をクリックします。

「アプライ・コントロール表の作成」ウィンドウで、各コントロール表の特性を定義します。

- a. リストからレプリケーション・コントロール表を選択します。
- b. 選択したコントロール表の表スペースと索引の特性を定義します。

別の表スペースを作成するコントロール表を必ずすべて選択し、そのコントロール表の表スペースと索引の特性を定義してください。

ヒント: 選択したデータベースのオペレーティング・システム・プラットフォーム用のコントロール表プロファイルを作成した場合は、そのプロファイルから設定値を受け入れるか、あるいはオーバーライドすることができます。

- c. すべてのコントロール表を定義したら、「OK」をクリックします。

モニター・コントロール表の作成

モニター・コントロール表は、UNIX、Windows、または z/OS™ データベースに作成することができます。OS/400 データベースまたは DB2 以外のリレーショナル・データベースにモニター・コントロール表を作成することはできません。モニター・コントロ

ール・サーバーを使用すれば、OS/400 データベースを含め、レプリケーション・ネットワーク内の任意の DB2 データベースのレプリケーション・アクティビティをモニターすることができます。

モニター・コントロール・サーバー用のコントロール表を作成するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・コントロール・サーバー」フォルダーを右クリックし、「モニター・コントロール表の作成」を選択します。
3. 「サーバーの選択」ウィンドウで、モニター・コントロール表を作成するサーバーを選択し、「OK」をクリックします。

「モニター・コントロール表の作成」ウィンドウで、各コントロール表の特性を定義します。

- a. リストからレプリケーション・コントロール表を選択します。
- b. 選択したコントロール表の表スペースと索引の特性を定義します。
別の表スペースを作成するコントロール表を必ずすべて選択し、そのコントロール表の表スペースと索引の特性を定義してください。
- c. すべてのコントロール表を定義したら、「OK」をクリックします。

レプリケーション・センターへのサーバーの追加

レプリケーション・コントロール表を特定のデータベース内に作成すると、レプリケーション・センターはそのデータベース・サーバーをレプリケーション・センターのナビゲーターに自動的に追加します。また、レプリケーション・センターのナビゲーターにデータベース・サーバーを追加または除去する場合、そのデータベース内に作成した他のレプリケーション・オブジェクトに影響を与えずに行うことができ、また、そのサーバー上で実行されている可能性のあるキャプチャー・プログラム、キャプチャー・トリガー、アプライ・プログラム、またはレプリケーション・アラート・モニターに影響を与えずに行うことができます。レプリケーション・センターは以下の理由から、ローカルでカタログされているデータベースすべてを自動的にリストしません。

- レプリケーション・センターは有効なレプリケーション・オブジェクトのみを表示します。ローカルでカタログされたデータベースにレプリケーション・コントロール表が含まれない場合、レプリケーション・センターはそのデータベースをナビゲーターに表示しません。
- ユーザーのレプリケーション環境がレプリケーション・コントロール表の作成権限を管理者に限定している場合でも、他の人々が自分に必要なデータベース内でレプリケーション・オブジェクト（たとえば、登録されたソースやサブスクリプション・セット）を管理することを許可できます。
- レプリケーション・チームの全員が同じ権限を持っている場合であっても、それぞれの人にとってみれば、特定のレプリケーション・サーバーにのみ焦点をあてたい場合

があります。レプリケーション・センターが表示する以上のものがレプリケーション環境に含まれている場合でも、それぞれの人は、自分が管理したいデータベース・サーバーだけを追加することができます。

重要: レプリケーション・センターにデータベース・サーバーを追加する前に、まず、そのサーバーをローカル DB2 データベースにカタログし、データベース内に必ずレプリケーション・コントロール表が存在するようにしなければなりません。

レプリケーション・センターに以下のサーバーを追加することができます。

- キャプチャー・コントロール・サーバー

キャプチャー・コントロール・サーバーをレプリケーション・センターに追加するには、「**キャプチャー・コントロール・サーバー**」フォルダーを右クリックし、「**追加**」を選択します。「キャプチャー・コントロール・サーバーの追加」ウィンドウで、リストから 1 つまたは複数のデータベース・サーバーを選択します。

DB2 以外のリレーショナル・サーバーをレプリケーション・センターにキャプチャー・サーバーとして追加することもでき、これを行うには、「キャプチャー・コントロール・サーバーの追加」ウィンドウに表示された特定の連合データベースを右クリックし、「**非 DB2 サーバーの検索**」を選択します。レプリケーション・センターは、その連合データベースに定義された、DB2 以外のリレーショナル・サーバーを表に追加します。

- アプライ・コントロール・サーバー

アプライ・コントロール・サーバーをレプリケーション・センターに追加するには、「**アプライ・コントロール・サーバー**」フォルダーを右クリックし、「**追加**」を選択します。「アプライ・コントロール・サーバーの追加」ウィンドウで、リストから 1 つまたは複数のデータベース・サーバーを選択します。

- モニター・コントロール・サーバー

モニター・コントロール・サーバーをレプリケーション・センターに追加するには、「**モニター・コントロール・サーバー**」フォルダーを右クリックし、「**追加**」を選択します。「モニター・コントロール・サーバーの追加」ウィンドウで、リストから 1 つまたは複数のデータベース・サーバーを選択します。

追加するサーバーごとに、選択されたデータベースにレプリケーション・センターが接続する時に使用する、有効なユーザー ID とパスワードを入力する必要があります。レプリケーション・センターは、各サーバー用に保管されたパスワード情報があれば、それを使用します。

データベースの変更キャプチャーの使用可能化 (UNIX および Windows)

UNIX または Windows システム上の DB2 データベースのデフォルトのロギングは循環ログであり、これは固定サイズのファイルを使用し、ログがいっぱいになると再利用されます。レプリケーションはアーカイブ・ロギングを必要とし、これは無限に増える 1 つまたは複数のログ・ファイルを使用し、決して再利用されることはありません (も

もちろん、DB2 ユーティリティを使用してアーカイブ・ログを管理し、ディスク・スペースがすべて一杯にならないようにできます。

DB2 データベースのアーカイブ・ロギングを使用可能にするには、次のようにします。

1. 「レプリケーション定義」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを展開します。
3. アーカイブ・ロギングを可能にしたいデータベースを右クリックし、「データベースのレプリケーションを使用可能にする」を選択します。
4. 「データベースのレプリケーションを使用可能にする」ウィンドウで「OK」をクリックし、データベース構成を変更 (LOGRETAIN を RECOVERY に) し、データベース・バックアップを開始します。

アーカイブ・ログを管理する出口ルーチンも使用したい場合は、USEREXIT データベース構成パラメーターを手作業で設定する必要があります。

「キャプチャー・コントロール・サーバー」フォルダーの内容ペインで複数のデータベースを選択することもでき、選択したデータベースすべてのアーカイブ・ロギングを同時に使用可能にすることができます。

その他のオペレーティング・システム上のデータベースにアーカイブ・ロギングを使用可能にする必要はありません。その理由は、それらの環境のデフォルトのロギングはアーカイブであるためです。また、DB2 以外のリレーショナル・データベース用にアーカイブ・ロギングを使用可能にする必要もありません。その理由はキャプチャー・トリガーがデータベース・ログに依存してしていないからです。

ソースの登録

レプリケーション用に 1 つまたは複数の表を登録するには、次のようにします。

1. 「キャプチャー・コントロール・サーバー」フォルダーを展開します。
2. 登録したいソース表を含むデータベース・サーバーを展開します。
3. 「登録済み表」フォルダーを右クリックし、「表の登録」を選択します。「登録可能な表の追加」ウィンドウがオープンします。

データベースには数百の表が含まれることがあるので、表のリストをあらかじめフィルターに掛けておけば、「表の登録」ウィンドウには必要な表だけを表示することができます。
4. 「登録可能な表の追加」ウィンドウから、検索基準があれば指定し、「検索」をクリックします。すべての表を含めたい場合は、「すべて検索」をクリックします。
5. フィルターに掛けられたリストから、レプリケーション・ソースとして登録したい 1 つまたは複数の表を選択し、「OK」をクリックします。「表の登録」ウィンドウはオープンされたままです。
6. 「選択済み表」リストから、レプリケーション・ソースとして登録したい最初の表を選択します。レプリケーション・ソースには、次の情報を定義することができます。

- いつキャプチャー・プログラムが CD 表に行を書き込むか (または、いつキャプチャー・トリガーが整合変更データ (CCD) 表に行を書き込むか) を指定する行キャプチャー規則。
- レプリケーションに使用したい特定の列 (変更前イメージの列および、変更後イメージの列を含む)。

登録しない列は、サブスクリプション・セットで使用できません。
- 登録された変更前イメージの列を変更後イメージの列と関連付けるために使用する接頭部。
- このソース表に基づいて、アプライ・プログラムにターゲット表をリフレッシュさせるかどうか。
- 変更を、削除と挿入の対としてキャプチャーするかどうか (パーティション化キーへの変更に便利)。
- update-anywhere シナリオにおいて、従属するレプリカ内で変更を再度取り込むかどうか。
- update-anywhere シナリオ用の、競合検出のレベル。

対等 (peer-to-peer) のシナリオの場合は、「**検出せず**」を選択する必要があります。

これらのオプションの詳細については、レプリケーション・センターのオンライン・ヘルプを参照してください。

登録されるソース表ごとに、CD 表と CD 表用の索引についての情報も指定します。このデータベース・サーバー用のソース・オブジェクト・プロファイルを作成した場合は、そのプロファイルに定義されたデフォルトを受け入れることも、デフォルトをオーバーライドすることもできます。

ビューを登録する場合は、「**登録済みビュー**」フォルダーを右クリックし、「**ビューの登録**」を選択します。ビューをレプリケーション・ソースとして登録するには、ビューは既存のものでなければなりません。ビューが存在しない場合は、「ビューの登録」ウィンドウ内の「**ビューの作成**」をクリックします。「ビューの作成」ウィンドウで、ビュー名およびビューを定義する SQL ステートメントを指定します。ビューを定義する SQL ステートメントを作成するには、「**SQL アシスタント**」をクリックし、「**SQL アシスタント**」ウィンドウを使用することができます。

リモートでジャーナルに記録されている OS/400 の表を、他の表を登録するのと同様に登録することができます。他の表の登録と異なる点は、ソース表の名前に加えて、ジャーナル・ライブラリーおよびジャーナル・レシーバー名を指定する必要があることです。ただし、ジャーナル・ライブラリーとジャーナル名が、ソース表またはソース・ファイルで使用されるジャーナル・ライブラリーおよびジャーナル名と同じ場合は、指定する必要はありません。

表の登録と同様に、ニックネームを登録できます。ただし、実際の表名 (DB2 以外のデータベースに保管されている名前) ではなく、表のニックネーム (DB2 データベースに保管されている名前) を指定する必要があります。

サブスクリプション・セットの作成

1 つまたは複数の表、ニックネーム、またはビューを登録した後、これらのソースにサブスクライブする必要があります。つまり、サブスクリプション・セットを作成し、このセットにメンバーを追加します。空のサブスクリプション・セットを作成し、後からこれにメンバーを追加することも、サブスクリプション・セットの作成中にすべてのメンバーを追加することもできます。

サブスクリプション・セットを作成するには、次のようにします。

1. 「**キャプチャー・コントロール・サーバー**」フォルダーを展開します。
2. サブスクリプション・セットを作成したいソース表を含むデータベース・サーバーを展開します。
3. 「**登録済み表**」フォルダーをクリックします。
4. 「**登録済み表**」フォルダーの内容ペインで、ソース表を右クリックし、「**サブスクリプション・セットの作成**」を選択します。「サブスクリプション・セットの作成」ウィンドウがオープンします。

別の方法として、次のステップを使用してサブスクリプション・セットを作成することもできます。

1. 「**アプライ・コントロール・サーバー**」フォルダーを展開します。
2. 特定のアプライ・コントロール・サーバーを展開します。
3. 「**サブスクリプション・セット**」フォルダーを右クリックし、「**作成**」を選択します。「サブスクリプション・セットの作成」ウィンドウがオープンします。

サブスクリプション・セットの作成は、主に次の 4 つのサブタスクからなります。

- 276 ページの『サブスクリプション・セットの情報の定義』
- 276 ページの『ソースからターゲットへのマッピング』
- 278 ページの『サブスクリプション・セットのスケジューリング』
- 278 ページの『サブスクリプション・セットへの SQL ステートメントまたはストアド・プロシージャの追加』

サブスクリプション・セットを作成した後、サブスクリプション・セットの編集、サブスクリプション・セットのメンバーの追加や除去、ステートメントやプロシージャの追加や除去、サブスクリプション・セットの活動化、そのメンバーの強制フル・リフレッシュ、あるいは、それを別のデータベースにプロモートする、といった作業を行うことができます。

サブスクリプション・セット作成の詳細については、レプリケーション・センターのオンライン・ヘルプを参照してください。

サブスクリプション・セットの情報の定義

「サブスクリプション・セットの作成」ウィンドウで、サブスクリプション・セットに次の情報を定義することができます。

- アプライ・コントロール・サーバーの別名
- サブスクリプション・セット名
- アプライ修飾子
- キャプチャー・コントロール・サーバーの別名
- キャプチャー・コントロール表 (サブスクリプション・セットに登録されたソースを定義する表) のセットを識別する、キャプチャー・スキーマ
- ターゲット・サーバーの別名
- サブスクリプション・セットを作成したらすぐにアクティブにするかどうか
デフォルトでは、新しいサブスクリプション・セットが作成されると、すぐに非アクティブになります。サブスクリプション・セットを即時にアプライ・プログラムの処理対象にするか、あるいは 1 回のアプライ・サイクルだけアクティブにするかを選択できます。
- サブスクリプション・セットの処理プロパティ

ソースからターゲットへのマッピング

サブスクリプション・セット情報を定義した後、ソース表およびビューをターゲット表にマッピングすることができます。「サブスクリプション・セットの作成」ウィンドウの「ソースからターゲットへのマッピング」ページで、次のようにします。

1. 「追加」をクリックして、「登録済みソースの追加」ウィンドウを表示します。このウィンドウから、選択されたソース・データベースの登録済みソースのリストをフィルターに掛けることができます。
2. フィルターに掛けられたリストから、サブスクリプション・セット・メンバーのソースとして追加したい 1 つまたは複数の表を選択し、「OK」をクリックします。「サブスクリプション・セットの作成」ウィンドウは「ソースからターゲットへのマッピング」ページでオープンされたままです。
3. 「ソースからターゲットへのマッピング」ページの表で、ターゲットのスキーマ、名前、またはターゲットのタイプを選択し、これらの値のどれでも、まだ存在しないターゲット表用の値に変更します。ソースを選択した後、ターゲット・スキーマおよびターゲット名は、選択されたターゲット・サーバーのターゲット・オブジェクト・プロファイル (ある場合) に基づいて、自動的に生成されます。
4. 「ソースからターゲットへのマッピング」ページの表で、表内のソースとターゲットの対を選択し、「詳細」をクリックして「メンバー・プロパティ」ウィンドウを表

示します。このウィンドウから、以下のような情報を含む、ソース表とターゲット表の正確なマッピングを指定することができます。

- ターゲットがサブスクライブするソース列の選択
- ソース列からターゲット列へのマッピング (算出列の作成を含む)
- ターゲット表の索引の指定
- オプションとして、WHERE 文節を使用してソースの行をフィルターに掛け、ターゲット表にはソース・データのサブセットだけを含めるようにする
- UNIX、Windows、および z/OS システムの場合は、ターゲット表用の表スペースを指定

レプリカのターゲット・タイプの場合、レプリカ定義 (行のキャプチャー規則、変更をリキャプチャーするかどうか、および更新をどのように扱うか)、レプリカ表の CD 表、および CD 表の索引も指定します。

CCD 表の場合は CCD 表のプロパティも指定します。このプロパティには、CCD 表が完全であるか不完全であるか、圧縮されているか、圧縮されていないか、およびこれをレプリケーション・ソースとして登録したいかどうか、が含まれます。

空のサブスクリプション・セットを作成する場合は、「ソースからターゲットへのマッピング」ページを空のままにしておきます。後から、サブスクリプション・セット・メンバーをサブスクリプション・セットに追加することができます。次のノートブックの 1 つを使用して、既存のサブスクリプション・セットにメンバーを追加することができます。

- サブスクリプション・セット・プロパティ。すでにサブスクリプション・セットを作成済みで、このセットに 1 つ以上のサブスクリプション・セット・メンバーを追加する場合は、このノートブックを使用します。

「サブスクリプション・セット」フォルダーの内容ペインからサブスクリプション・セットを右クリックし、「プロパティ」を選択します。

- サブスクリプション・セットにメンバーを追加する。1 つのメンバーを複数のサブスクリプション・セットに追加するには、このノートブックを使用します。たとえば、このノートブックを開いて 4 つのサブスクリプション・セットを選択した場合、1 つのメンバーをそれぞれのセットに追加できます。各メンバーは同じソースを使用する必要があります。

「登録済み表」フォルダーの内容ペインで、ソース表を右クリックし、「メンバーの追加」を選択します。

「サブスクリプション・セットにメンバーを追加する」ノートブックを使用すると、複数のサブスクリプション・セットに登録済みソースを追加することができます。これを使用すれば、いくつかの空のサブスクリプション・セットを作成し、これらのセットに同じ「ソースからターゲットへのマッピング」を移植することができます。

「サブスクリプション・セットにメンバーを追加する」ノートブックで選択されたす

すべてのサブスクリプション・セットは、同じキャプチャー・サーバーおよびキャプチャー・スキーマを使用する必要があります。

サブスクリプション・セットのスケジューリング

ソースからターゲットへのマッピングを行った後 (または空のサブスクリプション・セットを作成した後)、サブスクリプション・セットのタイミング情報を定義します。「サブスクリプション・セットの作成」ウィンドウの「スケジュール」ページで、サブスクリプション・セットを最初に処理の対象とするのはいつかを指定します。デフォルトは、ローカル・マシンの現在の日付と時刻です。また、サブスクリプション・セットをどのような頻度で処理の対象にするかも指定します。次のような頻度を指定できます。

- 時間に基づくレプリケーション

アプライ・プログラムは、定期的な時間インターバルを使用してこのサブスクリプション・セットを処理します。

- イベント・ベースのレプリケーション

アプライ・プログラムは、あるイベントが起こるたびにこのサブスクリプション・セットを処理します。

- 時間に基づくレプリケーションとイベント・ベースのレプリケーションの両方

アプライ・プログラムは、定期的な時間インターバルおよびイベントが起こる都度の両方を使用して、このサブスクリプション・セットを処理します。この場合、サブスクリプション・セットは、スケジュールされた時刻および、イベントが起こった時の両方で処理の対象になります。

サブスクリプション・セットへの SQL ステートメントまたはストアド・プロシージャの追加

サブスクリプション・セットのタイミング情報を定義した後、オプションとして、サブスクリプション・セットに SQL ステートメントまたはストアド・プロシージャを追加することができます。「サブスクリプション・セットの作成」ウィンドウの「ステートメント」ページで、アプライ・プログラムがサブスクリプション・セットを処理する時に実行する、SQL ステートメントまたはストアド・プロシージャを追加することができます。「追加」をクリックし、サブスクリプション・セットにステートメントまたはプロシージャを追加します。

「SQL ステートメントまたはプロシージャ呼び出しの追加」ウィンドウで、SQL ステートメントを入力するか、または SQL アシスタントを使用してステートメントを定義することができます。これらのステートメントまたはプロシージャを、ターゲット・サーバーでサブスクリプション・セットを処理する前または後に実行すべきこと、またはキャプチャー・コントロール・サーバーでサブスクリプション・セットを処理する前に実行すべきことを指定できます。また、アプライ・プログラムが正常終了として受け入れる SQLSTATE 値 (たとえば、存在しない行を削除しようとした場合の 02000 など) を追加することもできます。これらの SQLSTATE 値は正常として受け入れられ

るので、アプライ・トレール表 (IBMSNAP_APPLYTRAIL) にエラー条件は示されず、したがってレプリケーション・アラート・モニターはこれらについてアラートを生成しません。

サブスクリプション・セットの活動化または非活動化

通常、サブスクリプション・セットはアクティブにして、アプライ・プログラムで処理できるようにしておきます。しかし、一時的に、または無期限に、サブスクリプション・セットを非アクティブにしておきたい場合があります。サブスクリプション・セットの作成時に非アクティブにした場合は、いずれアクティブにすることが必要になります。

アクティブなサブスクリプション・セットを非アクティブにするには、次のようにします。

1. 「サブスクリプション・セット」フォルダーをクリックし、内容ペインにサブスクリプション・セットを表示します。
2. アクティブなサブスクリプション・セットを右クリックし、「非活動化」を選択します。レプリケーション・センターは、サブスクリプション・セットを即時に非活動化します。

非アクティブなサブスクリプション・セットをアクティブにするには、次のようにします。

1. 「サブスクリプション・セット」フォルダーをクリックし、内容ペインにサブスクリプション・セットを表示します。
2. アクティブなサブスクリプション・セットを右クリックし、次のオプションの 1 つを選択します。
 - サブスクリプション・セットをアクティブにする場合は、「活動化」→「無期限」
 - サブスクリプション・セットを 1 回のアプライ・サイクルのみアクティブにする場合は、「活動化」→「1 回のみ」

レプリケーション・センターは、サブスクリプション・セットを即時にアクティブにします。

レプリケーション・オブジェクトのプロモート

ソースを登録し、データベース・サーバーにサブスクリプション・セットを作成した後、ソースを再度登録したり、サブスクリプション・セットを再度作成したりせずに、レプリケーション定義を別のデータベースにコピーしたい場合があります (たとえば、テスト・システムから実動システムに)。レプリケーション・センターはプロモート関数を提供し、レプリケーション定義のあるデータベースから別のデータベースにコピーする作業を支援します。

制約事項:

- プロモート関数を使用してレプリケーション定義をコピーできるのは、類似のシステム間のみです。たとえば、ある DB2 (UNIX および Windows) システムから、別の DB2 (UNIX および Windows) システムへはコピー可能ですが、DB2 (UNIX および Windows) システムから DB2 for z/OS システムへはコピーできません。システムのオペレーティング・システム・プラットフォームがすべて同じであれば、OS/400、UNIX、Windows、または z/OS システム用にプロモート関数を使用できます。
- プロモート関数を使用して、DB2 以外のデータベースまたは連合データベース・オブジェクト用のレプリケーション定義をコピーすることはできません。
- プロモート関数を使用して、OS/400 リモート・ジャーナルを含むレプリケーション定義をコピーすることはできません。

登録済みの表またはビューのプロモート

登録済みの表をプロモートするには、次のようにします。

1. 「登録済みの表」フォルダーをクリックし、内容ペインに登録済みのソース表を表示します。
2. ソース表を右クリックし、「プロモート」を選択します。「登録済みの表のプロモート」ウィンドウがオープンします。
3. 「登録済みの表のプロモート」ウィンドウで、登録情報のコピー先のデータベース・サーバーについて、次の情報を指定します。
 - キャプチャー・コントロール・サーバーの別名
登録済みソース表の新しいキャプチャー・コントロール・サーバーを選択します。
 - キャプチャー・スキーマ
登録済みソース表の新しいキャプチャー・スキーマを指定します。
 - CD 表のスキーマ
ソース表に関連する CD 表の新しいスキーマ名を指定します。
 - 表スキーマ
表の新しいスキーマ名を指定します。プロモート関数を使用して、新しいデータベース内にソース表を作成することができます。

登録済みのビューをプロモートするには、「登録済みのビュー」フォルダーをクリックし、内容ペインに登録済みのソース・ビューを表示して、ソース・ビューを右クリックし、「プロモート」を選択します。

サブスクリプション・セットのプロモート

サブスクリプション・セットをプロモートするには、次のようにします。

1. 「サブスクリプション・セット」フォルダーをクリックし、内容ペインにサブスクリプション・セットを表示します。

2. サブスクリプション・セットを右クリックし、「**プロモート**」を選択します。「サブスクリプション・セットのプロモート」ウィンドウがオープンします。
3. 「サブスクリプション・セットのプロモート」ウィンドウで、サブスクリプション・セット情報のコピー先のデータベース・サーバーについて、次の情報を指定します。
 - アプライ・コントロール・サーバーの別名
サブスクリプション・セットの新しいアプライ・コントロール・サーバーを選択します。サブスクリプション・セットにすでに定義されているアプライ・コントロール・サーバーを選択することができます。
 - キャプチャー・コントロール・サーバーの別名
サブスクリプション・セットの新しいキャプチャー・サーバーを選択します。サブスクリプション・セットにすでに定義されているキャプチャー・コントロール・サーバーを選択することができます。
 - ターゲット・サーバーの別名
サブスクリプション・セットの新しいターゲット・サーバーを選択します。サブスクリプション・セットにすでに定義されているターゲット・サーバーを選択することができます。
 - アプライ修飾子
サブスクリプション・セットの新しいアプライ修飾子を入力します。
 - サブスクリプション・セット名
サブスクリプション・セットの新しい名前を入力します。
 - キャプチャー・スキーマ
サブスクリプション・セット内のソース表の新しいキャプチャー・スキーマを入力します。
 - ソース表またはビューのスキーマ名
サブスクリプション・セット内のソース表の新しいスキーマ名を入力します。
 - ターゲット表またはビューのスキーマ名
サブスクリプション・セット内のターゲット表の新しいスキーマ名を入力します。

現行のサブスクリプション・セット定義の値を使用したい場合は、すべてのフィールドをブランクのままにできます。

ターゲット表の強制フル・リフレッシュ

ターゲット表の再ロードが必要になる場合があります。たとえば、ソース・データベースのログまたはジャーナルに不一致があると、キャプチャー・プログラムは停止し、コールド・スタートが要求され、その結果、すべてのターゲット表をそのソース・データベースに基づいてすべてリフレッシュ（フル・リフレッシュ）することが必要になりま

す。小さな表の場合は、アプライ・プログラムに自動的にフル・リフレッシュを行わせてもかまわないでしょう。しかし大きな表の場合は、ASNLOAD 出口ルーチンを使用する必要があります。

レプリケーション・センターを使用して、通常はアプライ・プログラムが実行するフル・リフレッシュをバイパスし、ユーザーが、ソース表からのアンロードまたは抽出および、ターゲット表へのロードを実行することができます。レプリケーション・センターは、レプリケーション・コントロール表に必要な変更を行い、ロードが完了した後、レプリケーションが確実に継続できるようにします。

手動フル・リフレッシュを実行するには、次のようにします。

1. 「サブスクリプション・セット」フォルダーをクリックし、内容ペインにサブスクリプション・セットを表示します。
2. サブスクリプション・セットを右クリックし、「フル・リフレッシュ」→「手動」を選択します。
3. 「フル・リフレッシュ - 手動の概要」ウィンドウにあるテキストを読み、「次へ」をクリックします。
4. 続いて表示されるウィンドウで「次へ」をクリックし、タスクを完了します。

「フル・リフレッシュ - 手動」ウィンドウでは、以下のステップを実行することができます。

1. 選択したサブスクリプション・セットの現行のサブスクリプションを使用不可にする。

サブスクリプション・セットを使用不可にした後、ソース表をアンロードし、ターゲット表にロードすることができます。

2. 選択したサブスクリプション・セットのサブスクリプションを、再び使用可能にする。

ステップ 1 とステップ 2 用に生成された SQL スクリプトは、即時に、または後で実行することができます。これらのステップは、必ず「フル・リフレッシュ - 手動」ウィンドウで指定された順序で実行してください。そうしないと、レプリケーション環境は予測不能な結果になる可能性があります。

自動フル・リフレッシュを行うため、アプライ・プログラムが次のアプライ・サイクルでフル・リフレッシュを開始するには、次のようにします。

1. 「サブスクリプション・セット」フォルダーをクリックし、内容ペインにサブスクリプション・セットを表示します。
2. サブスクリプション・セットを右クリックし、「フル・リフレッシュ」→「自動」を選択します。

レプリケーション定義の除去または削除

レプリケーション・センターを使用して、作成済みのレプリケーション定義をどれでも除去または削除することができます。以下のタスクのいずれかを実行することができます。

- レプリケーション・センターからユーザー ID を除去する。
- キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、またはモニター・コントロール・サーバーから、レプリケーション・コントロール表をドロップする (OS/400 システムには該当しない)。
- キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、またはモニター・コントロール・サーバーを、レプリケーション・センターから除去する。
- ソース表またはビューの登録を削除する。
- サブスクリプション・セットを削除する。
- サブスクリプション・セットからメンバーを削除する。
- サブスクリプション・セットからステートメントを削除する。
- サブスクリプション・セットからストアド・プロシージャを除去する。

各タスクの詳細については、レプリケーション・センターのオンライン・ヘルプを参照してください。

キャプチャー・プログラムの操作

レプリケーション・センターから、レプリケーションのための日常の操作タスクを実行することができます。たとえば、キャプチャー・プログラムを始動または停止することができます。キャプチャー・プログラムを操作するには、「操作」フォルダーを展開して、「キャプチャー・コントロール・サーバー」フォルダーをクリックし、現在定義されているキャプチャー・コントロール・サーバーを内容ペインに表示します。キャプチャー・コントロール・サーバーの 1 つを右クリックし、次の操作の 1 つを選択します。

- キャプチャー・プログラムを始動する。
- キャプチャー・プログラムを停止する。
- キャプチャー・プログラムを中断する。
- キャプチャー・プログラムを再開する (延期の後)
- キャプチャーのプルーニング処理を開始し、CD 表、UOW 表、キャプチャー・モニター表、キャプチャー・トレース表、およびシグナル表のプルーニングを行う。
- キャプチャー・プログラムを再初期化し、登録表を再度読み取る。
- キャプチャー・パラメーター表に保管された値を表示または変更する。
- キャプチャー・プログラムが使用している現行のパラメーターを表示または変更する。

- キャプチャー・プログラムが出したメッセージを表示する。
- キャプチャー・プログラムが収集した統計を表示する。
 - キャプチャー・プログラムが CD 表に挿入またはスキップした行数
 - キャプチャー・プログラムが CD 表から除去した行数
 - キャプチャー・プログラムがコミットしたトランザクションの数
 - キャプチャー・プログラムが使用しているメモリーの量
- キャプチャー・プログラムの平均待ち時間を表示する。
- キャプチャー・プログラムの状況を照会する。

レプリケーション・ネットワーク内で実行中のどのキャプチャー・プログラムに対しても、これらのタスクのどれでも実行することができます。

アプライ・プログラムの操作

レプリケーション・センターを使用して、アプライ・プログラムを操作することもできます。アプライ・プログラムを操作するには、「操作」フォルダーを展開して、「アプライ・コントロール・サーバー」フォルダーをクリックし、アプライ・コントロール・サーバーの 1 つを展開してから、「アプライ修飾子」フォルダーをクリックして、現在定義されているアプライ修飾子を内容ペインに表示します。アプライ修飾子の 1 つを右クリックし、次の操作の 1 つを選択します。

- アプライ・プログラムを始動する。
- アプライ・プログラムを停止する。
- サブスクリプション・セット・アクティビティのレポートを表示する。
 - すべてのサブスクリプション・セットを表示する。
 - 失敗したサブスクリプション・セットを表示する。
 - 成功したサブスクリプション・セットを表示する。
 - 失敗したサブスクリプション・セットのエラー・サマリー・レポートを表示する。
- アプライ・プログラムのパフォーマンス情報を表示する。
 - CD 表からフェッチした行数を表示する。
 - サブスクリプション・セットごとの経過時間を表示する。
- サブスクリプション・セットごとのエンドツーエンドの待ち時間のレポートを表示する。
- アプライ・プログラムの状況を照会する。

レプリケーション・ネットワーク内で実行中のどのアプライ・プログラムに対しても、これらのタスクのどれでも実行することができます。

レプリケーション・アラート・モニターの操作

レプリケーション・センターを使用して、レプリケーション・アラート・モニター用の連絡先およびアラート条件を定義することができます。

レプリケーション・アラート・モニターが、指定されたアラート条件のいずれかを検出した時に知らせる連絡先を作成するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・コントロール・サーバー」フォルダーを展開します。
3. 「モニター・コントロール・サーバー」を展開します。
4. 「連絡先」フォルダーを右クリックし、「連絡先の作成」→「担当者」または、「連絡先の作成」→「グループ」を選択します。
5. 「連絡先の作成」ウィンドウで、担当者の名前および E メールまたはポケットベルのアドレスを指定します。「連絡先グループの作成」ウィンドウで、グループの名前とグループのメンバーを指定します。

キャプチャー・プログラムのアラート条件を選択するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・コントロール・サーバー」フォルダーを展開します。
3. 「モニター・コントロール・サーバー」を展開します。
4. 「モニター修飾子」フォルダーを右クリックし、「キャプチャー・スキーマ用のアラート条件を選択」を選択します。
5. 「キャプチャー・スキーマ用のアラート条件を選択」ウィンドウで、次の情報を指定します。
 - レプリケーション・アラート・モニターのそのインスタンスに関連付けられたモニター修飾子
 - モニターしたいキャプチャー・コントロール・サーバー
 - モニターしたいキャプチャー・スキーマ
 - 任意のアラート条件

アプライ・プログラムのアラート条件を選択するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・コントロール・サーバー」フォルダーを展開します。
3. 「モニター・コントロール・サーバー」を展開します。
4. 「モニター修飾子」フォルダーを右クリックし、「アプライ修飾子またはサブスクリプション・セットのアラート条件を選択」を選択します。
5. 「アプライ修飾子またはサブスクリプション・セットのアラート条件を選択」ウィンドウで、次の情報を指定します。

- レプリケーション・アラート・モニターのそのインスタンスに関連付けられたモニター修飾子
- モニターしたいアプライ・コントロール・サーバー
- モニターしたい任意の特定のサブスクリプション・セット
- モニターしたい任意の特定のアプライ修飾子
- 任意のアラート条件

モニター修飾子用のレプリケーション・アラート・モニターを開始するには、次のようにします。

1. 「**操作**」フォルダーを展開します。
2. 「**モニター・コントロール・サーバー**」フォルダーを展開します。
3. 「**モニター・コントロール・サーバー**」を展開します。
4. 「**モニター修飾子**」フォルダーを展開します。
5. モニター修飾子を右クリックし、「**モニターの開始**」を選択します。

レプリケーション・ネットワーク内で実行中のいずれのレプリケーション・アラート・モニターに対しても、これらのタスクのどれでも実行することができます。

第 15 章 基本的なデータ・レプリケーションのシナリオ: DB2 for Windows

この章ではシナリオを使って、DB2® レプリケーション・センターとキャプチャー・プログラムおよびアプライ・プログラムを実際に操作してみます。ここに示す簡単なシナリオでは、DB2 レプリケーション・ソースの変更を、DB2 for Windows® Enterprise Server Edition (ESE) または Workgroup Server Edition (WSE) のデータベース内のターゲット表にコピーしているのを、そのステップをたどってみてください。

このシナリオは、以下の部分からなります。

1. 『開始する前に』
2. 288 ページの『このシナリオの計画』
3. 290 ページの『このシナリオ用のレプリケーション環境のセットアップ』
4. 305 ページの『レプリケーション環境における操作』

開始する前に

このシナリオをご使用のコンピューター上で実行する場合は、以下のステップでシステムをセットアップしてください。

1. コンピューターに DB2 for Windows がインストールされていることを確認します。
2. デフォルトの DB2 インスタンスを作成したことを確認します。このシナリオは、すべてのデータベースが同じインスタンス内にあると想定しています。
3. SAMPLE データベースへのアクセス権があることを確認します。このシナリオでは、このデータベースはソース・サーバーとキャプチャー・コントロール・サーバーの両方の役割を果たします。

SAMPLE データベースを作成するには、「ファースト・ステップ」アプリケーションを使用します（「スタート」→「プログラム」→「IBM DB2」→「セットアップ・ツール」→「ファースト・ステップ」を選択します）。データベースを作成したら、「ファースト・ステップ」ウィンドウをクローズします。

DB2 のインストール時にファースト・ステップをインストールしなかった場合は、DB2 コマンド・ウィンドウを開き、**db2sampl** コマンドを実行して、SAMPLE データベースを作成してください。

4. DB2 コントロール・センターを使って COPYDB という新しいデータベースを作成します。このデータベースは、ターゲット・サーバーとして、またアプライ・コントロール・サーバーとして使用します。データベースを作成するには、「データベース」フォルダーを右クリックし、「作成」→「ウィザードを使用するデータベース」

を選択し、指示にしたがって、デフォルト・オプションを使用して新しいデータベースを作成します。データベースの名前と別名は両方とも COPYDB にしてください。

この章のステップでは、SAMPLE データベースにある DEPARTMENT 表のデータを使用します。完全修飾名は `schema.DEPARTMENT` であり、`schema` は、表を作成したユーザー ID です。表 12 に DEPARTMENT 表を示します。

表 12. DEPARTMENT 表

| DEPTNO | DEPTNAME | MGRNO | ADMRDEPT | LOCATION |
|--------|---------------------------------|--------|----------|----------|
| A00 | SPIFFY COMPUTER SERVICE DIV. | 000010 | A00 | - |
| B01 | PLANNING | 000020 | A00 | - |
| C01 | INFORMATION CENTER | 000030 | A00 | - |
| D01 | DEVELOPMENT CENTER | - | A00 | - |
| D11 | MANUFACTURING SYSTEMS | 000060 | D01 | - |
| D21 | ADMINISTRATION SYSTEMS | 000070 | D01 | - |
| E01 | SUPPORT SERVICES | 000050 | A00 | - |
| E11 | OPERATIONS | 000090 | E01 | - |
| E21 | SOFTWARE SUPPORT | 000100 | E01 | - |

このシナリオの残りの部分では、SAMPLE および COPYDB データベースの作成に使用したユーザー ID を使用してください。このユーザー ID はデータベースの作成者なので、レプリケーション・タスクを実行するために必要な権限 (DBADM または SYSADM) があります。

このシナリオの計画

ここでは、自分のグループがレポートを生成するアプリケーションを使用するとします。このアプリケーションでは、SAMPLE データベースの DEPARTMENT 表にある情報が必要になります。そこで、ソース表のデータを直接は使用せずに、変更をターゲット表にコピーし、レポート生成アプリケーションだけがターゲット表を読み取れるようにします。管理を簡単にするため、ターゲット表をソース表と同じマシンに保管します。

必要なのは簡単なデータ分散構成であり、変更は、1 つのレプリケーション・ソースから 1 つの読み取り専用コピーにレプリケーションされます。ここでは、レプリケーション・タスクを実行する前に考える必要がある、設計と計画の問題を扱います。

レプリケーション・ソース

レプリケーション・ソースは、SAMPLE データベースの `schema.DEPARTMENT` 表です。環境のセットアップに先立って、その表から何を複製したいかを定める必要があります。このシナリオでは、すべての列を登録し、すべての列をサブスクライブすることになります。

レプリケーション・ターゲット

レプリケーション・ターゲットを COPYDB データベースにすることを決め、これは 287 ページの『開始する前に』で作成しました。現時点では、このデータベースにはターゲット表がありません。ターゲット表は、ユーザーの指定にしたがってレプリケーション・センターに作成させることにします。このように、ターゲット表を自動的に生成する方法は、レプリケーション・ソースとの正確なマッピングができるので、望ましい方法です。既存のターゲット表を使用することもできますが、このシナリオではターゲット表は存在していないと想定しています。

COPYDB 内のターゲット表に、表 13 に示された情報の列を含めることにします。

表 13. COPYDB 表の列

| DEPTNO | レプリケーション・ソース表の DEPTNO 列にある情報。この列は、ターゲット表の主キーになります。 |
|----------|--|
| DEPTNAME | レプリケーション・ソース表の DEPTNAME 列にある情報。 |
| MGRNO | レプリケーション・ソース表の MGRNO 列にある情報。 |
| ADMRDEPT | レプリケーション・ソース表の ADMRDEPT 列にある情報。 |
| LOCATION | レプリケーション・ソース表の LOCATION 列にある情報。 |

ターゲット表の列はソース表のデータを単純に反映したものであり、ソース表の行ごとにターゲット表には 1 行だけが存在するので、ターゲット表のタイプとして、ユーザー・コピーを使用できます。

レプリケーション・オプション

このシナリオの目的から、ここでは、CD 表、ターゲット表、およびレプリケーション・コントロール表を、それぞれの該当のデフォルト表スペースに保管することを決めます (表 14 参照)。SAMPLE と COPYDB のデータベースは同じマシンにあります。それぞれ表スペースは別のコンテナにあります。

表 14. このシナリオで使用される表および表スペース

| データベース | 表 | 表スペース | 内容 |
|--------|---------------------|---------------------------------------|-------------------------------------|
| SAMPLE | schema.DEPARTMENT | USERSPACE1 | ソース表 |
| | schema.CDDEPARTMENT | TSCDDEPARTMENT | DEPARTMENT 表の CD 表 |
| | キャプチャー・コントロール表 | TSASNCA および TSASNUOW | キャプチャー・プログラムのレプリケーション・コントロール表 |
| COPYDB | schema.TGDEPTCOPY | TSTGDEPTCOPY | ターゲット表 |
| | アプライ・コントロール表 | TSASNAA | アプライ・プログラムのレプリケーション・コントロール表 |
| | モニター・コントロール表 | REPLMONT1、REPLMONT2、 および REPLMONT3 | レプリケーション・アラート・モニターのレプリケーション・コントロール表 |

表スペース・レベルで起こりうる競合を減らすため、通常、CD 表はソース表とは別の表スペースに作成する必要があります。レプリケーション・コントロール表の表スペースについては、デフォルトを受け入れるか、またはレプリケーション・センター内にプ

ロファイルを定義してください。実稼働環境の場合、起こりうる競合を避けるには、それぞれの表スペースを別の装置に作成するのがベストです。

レプリケーションのスケジューリングについては、DB2 レプリケーションを使用して、ソース表での変更を 1 分ごとに調べ、その変更をターゲット表にレプリケーションすることになります。レポート生成アプリケーションの場合、1 分ごとに変更を反映するというような応答時間は必要ないでしょうが、レプリケーション環境をテストして、すべての機能が正確に働いているか確認することになります。

また、レプリケーション・サイクルが終わるごとに、1 週間 (7 日) 以上経過したアプライ・トレール表のレコードを削除することになります。このようなプルーニング操作により、表が大きくなりすぎるのを防ぐことができます。

このシナリオ用のレプリケーション環境のセットアップ

レプリケーション・モデルの計画を立てれば、レプリケーション環境をセットアップする準備ができたこととなります。以下のステップのほとんどはレプリケーション・センターを使用するので、レプリケーション・センターが実行されていることを確認してください (Windows の「スタート」メニューから、「プログラム」→「IBM DB2」→「汎用管理ツール」→「レプリケーション・センター」を選択します)。

ステップ 1: キャプチャー・プログラムのレプリケーション・コントロール表を作成する

キャプチャー・プログラムは、レプリケーション・コントロール表から現行の登録情報を読み取り、またキャプチャー・プログラムの状況をこの表に保管します。キャプチャー・コントロール・サーバーとして働くデータベースには、キャプチャー・コントロール表を含める必要があります。

キャプチャー・コントロール表を作成するには、次のようにします。

1. 「レプリケーション定義」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを右クリックし、「キャプチャー・コントロール表の作成」→「クイック」を選択します。あるいは別の方法として、「キャプチャー・コントロール表の作成」→「カスタム」を選択し、キャプチャー・コントロール表をカスタマイズすることもできます。
3. 「サーバーの選択」ウィンドウで、SAMPLE データベースを選択します。このデータベースはキャプチャー・コントロール・サーバーになります。「OK」をクリックします。
4. 「コントロール表の作成」- 「クイック」- 「サーバー情報」ウィンドウで、「レプリケーション用のソースの受け入れとそれらのソースに対する変更のキャプチャー」を選択します。その後、「次へ」をクリックします。

5. 「コントロール表の作成」 - 「クイック」 - 「レプリケーションの詳細」 ウィンドウで、「次へ」をクリックします。このウィンドウの情報は何も変更する必要はありません。
6. 「コントロール表の作成」 - 「クイック」 - 「表スペース」 ウィンドウで、TSASNCA 表スペースの表スペース仕様を入力します。たとえば、バッファークラスタを IBMDEFAULTBP に設定します。このシナリオでは、デフォルトのキャプチャー・スキーマ ASN を受け入れます。
7. 「コントロール表の作成」 - 「クイック」 - 「表スペース」 ウィンドウで、TSASNUOW 表スペースの表スペース仕様を入力します。
8. 「コントロール表の作成」 - 「クイック」 - 「表スペース」 ウィンドウで、両方の表スペースの情報を入力した後、「OK」をクリックします。
9. 「メッセージ・ダイアログ」 ウィンドウで「クローズ」をクリックします。このウィンドウは、キャプチャー・コントロール表を作成する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
10. 「今実行するか、SQL を保管する」 ウィンドウで有効なユーザー ID とパスワードを入力し、「OK」をクリックすれば、SQL スクリプトは即時に実行されます。
11. 「DB2 メッセージ」 ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
12. 「キャプチャー・コントロール・サーバー」 フォルダを展開します。フォルダの下に SAMPLE データベースが表示されます。

ステップ 2: ソース・データベースのレプリケーションを使用可能にする

キャプチャー・プログラムは DB2 ログを読み取り、登録された表への変更を含むログ・レコードを探します。キャプチャー・プログラムがログを読み取る前に DB2 がログ・ファイルを再利用することがないように、ログはアーカイブ・ログでなければなりません。UNIX[®] および Windows 環境の場合、DB2 のデフォルトは循環ログなので、この設定をアーカイブ・ログに変更する必要があります。

ソース・データベースのレプリケーションを使用可能にするには、次のようにします。

1. 「キャプチャー・コントロール・サーバー」 フォルダを展開します。
2. SAMPLE データベースを右クリックし、「レプリケーション用データベースを使用可能にする」を選択します。
3. 「レプリケーション用データベースを使用可能にする」 ウィンドウで「OK」をクリックし、SAMPLE データベースにアーカイブ・ログを使用して、データベースのバックアップを開始します。オプションとして、「データベースのバックアップ」ウィザードを使用して、バックアップ処理をガイドさせることもできます。
4. 「データベースのバックアップ」 ウィンドウで、データベース・バックアップ用の情報を指定し、「今バックアップする」をクリックします。

データベースをバックアップした後、キャプチャー・プログラムを始動することもできますが、ここではまだ始動しません。キャプチャー・プログラムを始動したい場合は、303 ページの『ステップ 7: シナリオのデータを複製する』を参照してください。

ステップ 3: レプリケーション・ソースを登録する

キャプチャー・コントロール表を作成し、データベースをレプリケーションの対象として使用可能にした後、DEPARTMENT 表をレプリケーション・ソースとして登録します。

表をレプリケーション・ソースとして登録するには、次のようにします。

1. 「レプリケーション定義」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを展開します。
3. 「SAMPLE データベース」フォルダーを展開します。
4. 「登録済み表」フォルダーを右クリックし、「表の登録」を選択します。
5. 「登録可能な表の追加」ウィンドウで、「すべて検索」をクリックし、レプリケーション・ソースとして登録できる SAMPLE データベース内のすべての表をリストします。DEPARTMENT 表を選択し、「OK」をクリックします。「表の登録」ウィンドウが、293 ページの図 8 に示すように表示されます。



図 8. 「表の登録」ウィンドウ

6. 「表の登録」ウィンドウで、「**CD 表**」ノートブック・タブをクリックします。CD 表スペースについて、以下の情報を指定します。
 - 「表スペースの仕様」エリアで、「**コンテナ名**」フィールドをクリックし、TSCDDEPARTMENT 表スペースのコンテナ名を指定します。
 - 「表スペースの仕様」エリアで、「**サイズ**」フィールドを 1 に変更します。
 - 「表スペースの仕様」エリアで、「**単位**」フィールドを MB に変更します。
 - この新しい表スペースについてのその他の情報 (バッファ・プールを IBMDEFAULTBP に設定するなど) を指定します。

表スペース情報を入力した後、「**OK**」をクリックします。

7. 「メッセージ・ダイアログ」ウィンドウで「**クローズ**」をクリックします。このウィンドウは、キャプチャー・コントロール表を作成する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。

8. 「今実行するか、SQL を保管する」ウィンドウで「OK」をクリックすれば、SQL スクリプトは即時に実行されます。
9. 「DB2 メッセージ」ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
10. SAMPLE データベース・フォルダーの内容ペインには、登録済みの表として DEPARTMENT 表が表示されます。図9に、SAMPLE データベース・フォルダーの内容ペインに、DEPARTMENT 表が登録済みの表として表示されている例が示されています。

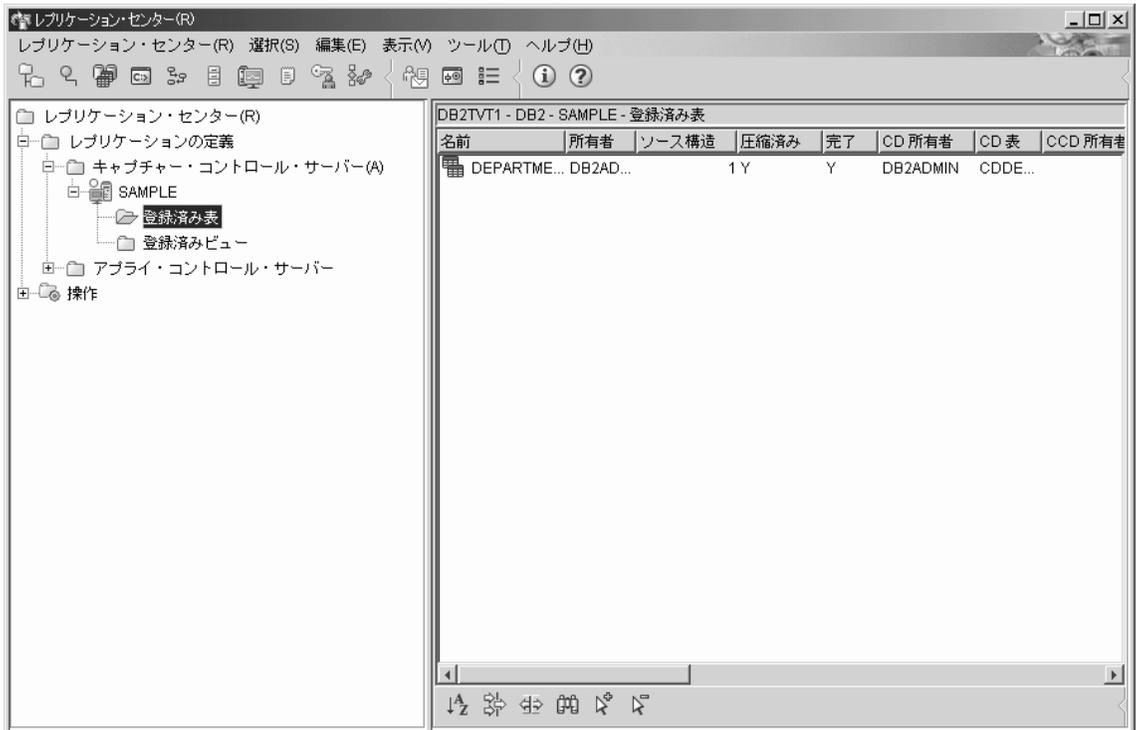


図9. DEPARTMENT 表が SAMPLE データベースの登録済みの表としてリストされています。

これで、DEPARTMENT 表はレプリケーション・ソースとして定義されました。SQL スクリプトを実行した時に、レプリケーション・センターがこのレプリケーション・ソースの CD 表および CD 表の索引を作成し、キャプチャー・コントロール表を更新しました。

ステップ 4: アプライ・プログラムのレプリケーション・コントロール表を作成する

アプライ・プログラムは、レプリケーション・コントロール表から現行のサブスクリプション・セット情報を読み取り、またアプライ・プログラムの状況をこの表に保管します。アプライ・コントロール・サーバーとして働くデータベースには、アプライ・コントロール表を含める必要があります。

アプライ・コントロール表を作成するには、次のようにします。

1. 「レプリケーション定義」フォルダーを展開します。
2. 「アプライ・コントロール・サーバー」フォルダーを右クリックし、「アプライ・コントロール表の作成」→「クイック」を選択します。あるいは別の方法として、「アプライ・コントロール表の作成」→「カスタム」を選択し、アプライ・コントロール表をカスタマイズすることもできます。
3. 「サーバーの選択」ウィンドウで、COPYDB データベースを選択します。このデータベースはアプライ・コントロール・サーバーになります。「OK」をクリックします。
4. 「コントロール表の作成」-「クイック」-「サーバー情報」ウィンドウで、「キャプチャー済み変更をターゲット表に適用」を選択します。その後、「次へ」をクリックします。
5. 「コントロール表の作成」-「クイック」-「レプリケーションの詳細」ウィンドウで、「次へ」をクリックします。このウィンドウの情報は何も変更する必要はありません。
6. 「コントロール表の作成」-「クイック」-「表スペース (Table Space)」ウィンドウで、TSASNAA 表スペースの表スペース仕様を入力します。たとえば、バッファークールを IBMDEFAULTBP に設定します。「OK」をクリックします。
7. 「メッセージ・ダイアログ」ウィンドウで「クローズ」をクリックします。このウィンドウは、アプライ・コントロール表を作成する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
8. 「今実行するか、SQL を保管する」ウィンドウで有効なユーザー ID とパスワードを入力し、「OK」をクリックすれば、SQL スクリプトは即時に実行されます。
9. 「DB2 メッセージ」ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
10. 「アプライ・コントロール・サーバー」フォルダーを展開します。フォルダーの下に COPYDB データベースが表示されます。

ステップ 5: サブスクリプション・セットとサブスクリプション・セット・メンバーを作成する

ソース表を登録したら、サブスクリプション・セットを作成する必要があります。サブスクリプション・セットは、レプリケーション・ソース・データベース (このシナリオでは SAMPLE) とターゲット・データベース (このシナリオでは COPYDB) のリレーシ

ョンシップを定義します。サブスクリプション・セット・メンバーは、レプリケーション・ソース表 (このシナリオでは DEPARTMENT) と、1 つまたは複数のターゲット表 (このシナリオでは 1 つしかなく、名前は DEPTCOPY) のリレーションシップを定義します。

サブスクリプション・セットとサブスクリプション・セット・メンバーを作成するには、次のようにします。

1. 「レプリケーション定義」フォルダーを展開します。
2. 「アプライ・コントロール・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「サブスクリプション・セット」フォルダーを右クリックし、「作成」を選択します。

別の方法として、SAMPLE データベースの「登録済み表」フォルダーを選択して、内容ペインの DEPARTMENT 表を右クリックし、「サブスクリプション・セットの作成」を選択しても、サブスクリプション・セットを作成することができます。

5. 「サブスクリプション・セットの作成」ウィンドウの「セット情報」ページに、以下の情報を入力します。
 - a. 「セット名」フィールドに、DEPTSUB と入力します。このストリングはサブスクリプション・セットを識別するものであり、特定のアプライ修飾子に固有でなければなりません。
 - b. 「アプライ修飾子」フィールドに、DEPTQUAL と入力します。このストリングは、このレプリケーション定義を識別するものであり、このサブスクリプション・セットを実行するアプライ・プログラムのインスタンスに固有です。

ヒント: アプライ修飾子は大文字小文字が区別されます。アプライ修飾子を小文字で指定する場合は、入力時にアプライ修飾子を区切る必要があります (たとえば "deptqual")。単に deptqual と入力すると、デフォルトにより、レプリケーション・センターはその値を大文字に変換します。

- c. 「キャプチャー・コントロール・サーバーの別名」フィールドのブラウズ・ボタンをクリックします。「キャプチャー・コントロール・サーバーの選択」ウィンドウで SAMPLE データベースを選択し、「OK」をクリックします。
- d. 「ターゲット・サーバーの別名」フィールドのブラウズ・ボタンをクリックします。「ターゲット・サーバーの選択」ウィンドウで COPYDB データベースを選択し、「OK」をクリックします。COPYDB データベースは、ターゲット・サーバーであり、またアプライ・コントロール・サーバーでもあり、その両方です。
- e. 「サブスクリプション・セットの活動化」チェック・ボックスを選択します。

「セット情報」ページのその他のフィールドの設定値を変更する必要はありません。「サブスクリプション・セットの作成」ウィンドウは、図 10 に似たウィンドウになります。

The screenshot shows a dialog box titled "サブスクリプション・セットの作成 - COPYDB" (Subscription Set Creation - COPYDB). The main title bar also includes "DB2TVT1 - DB2 - COPYDB". The dialog has several tabs: "設定情報" (Configuration Information), "ソースからターゲットへのマッピング" (Mapping from Source to Target), "スケジュール" (Schedule), and "ステートメント" (Statement). The "設定情報" tab is active. It contains the following fields and options:

- アプライ・コントロール・サーバー別名(A)**: COPYDB (COPYDB)
- セット名(S)**: DEPTSUB
- アプライ修飾子(Q)**: DEPTQUAL (dropdown menu)
- キャプチャー・コントロール・サーバー別名(C)**: SAMPLE (SAMPLE)
- キャプチャー・スキーマ(P)**: ASN (dropdown menu)
- ターゲット・サーバー別名(T)**: COPYDB (COPYDB)
- サブスクリプション・セットの活動化(V)**
 - 無期限にアクティブにする(D)
 - 1回のアプライ・サイクルの間だけアクティブにする(N)
- 処理プロパティの設定**
 - データ・ブロック化係数(E)**: 20 (spin box)
 - セット・メンバーに対してトランザクション処理を使用する許可をアプライに与える(W)**
 - この数までトランザクションがターゲット表に適用されるとアプライがコミットされる数(U): 0 (spin box)

At the bottom, there are three buttons: "OK(O)", "キャンセル" (Cancel), and "ヘルプ" (Help).

図 10. 「サブスクリプション・セットの作成」ウィンドウ

6. 「サブスクリプション・セットの作成」ウィンドウの「ソースからターゲットへのマッピング」ページに、以下の情報を入力します。
 - a. 「追加」をクリックし、登録済みのソースをサブスクリプション・セット・メンバーに追加します。
 - b. 「登録済みソースの追加」ウィンドウで、「すべて検索」をクリックし、SAMPLE データベース内の登録済みソースをすべて表示します。

- c. 「登録済みソースの追加」ウィンドウで、DEPARTMENT 表を選択し、「OK」をクリックします。
- d. 「サブスクリプション・セットの作成」ウィンドウの「ソースからターゲットへのマッピング」ページで、ターゲット表の名前を TGDEPARTMENT から TGDEPTCOPY に変更します。サブスクリプション・セット・メンバー表の「ターゲット名」列にある TGDEPARTMENT を選択し、デフォルト名を TGDEPTCOPY で上書きします。

ここではユーザー・コピー・ターゲット表を作成するので、ターゲット・タイプは変更しないでください。
- e. 「詳細」をクリックし、「メンバー・プロパティ」ウィンドウをオープンします。このウィンドウから、サブスクリプション・セット・メンバーのプロパティを定義することができます。

「メンバー・プロパティ」ウィンドウの「列選択」または「列のマッピング」ページは何も変更する必要はありません。その理由は、すべての列を複製し、ソース表と同じ列をターゲット表に作成するからです。「メンバー・プロパティ」ウィンドウの例が 299 ページの図 11 に示されています。

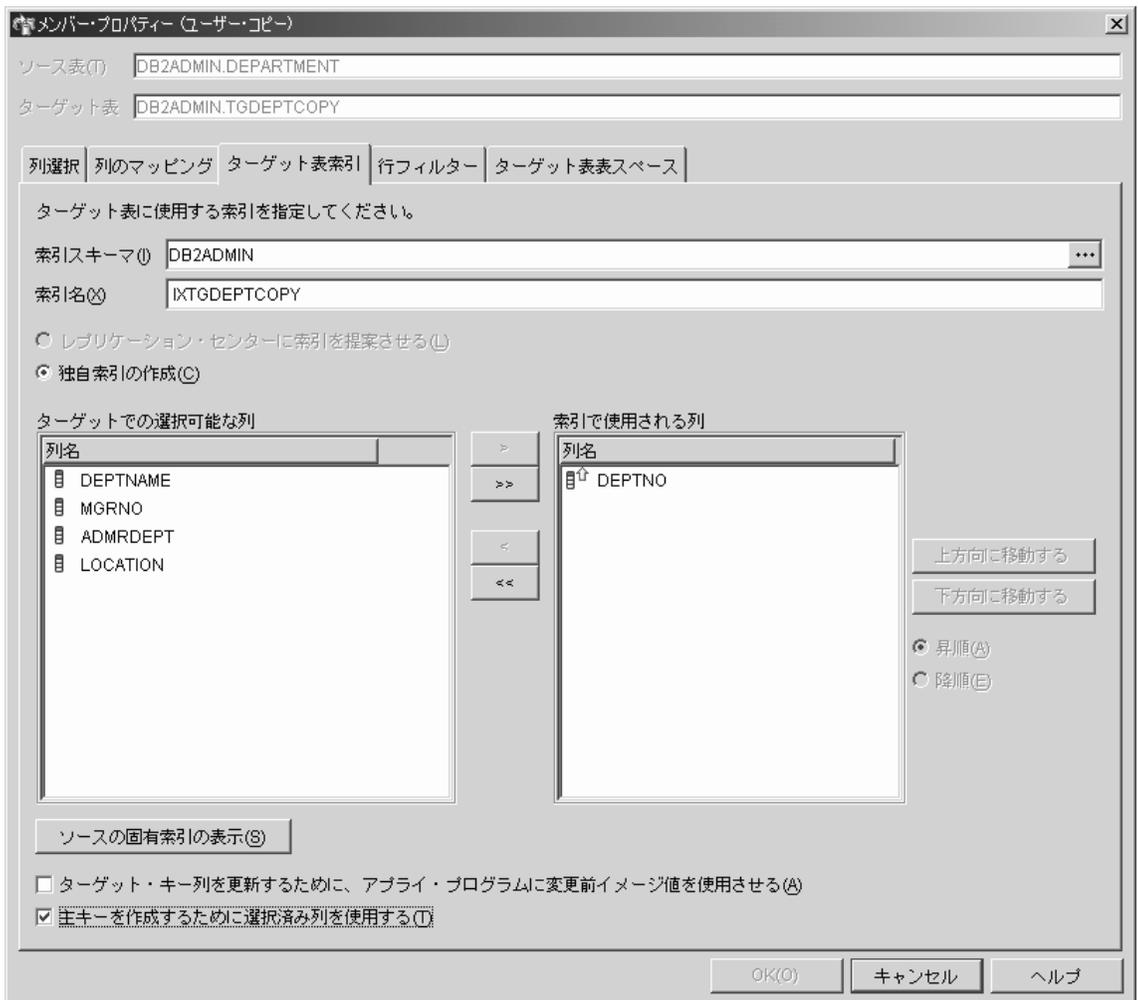


図 11. 「メンバー・プロパティ」ウィンドウ

7. 「メンバー・プロパティ」ウィンドウの「ターゲット表索引」ページで、次のようになります。
 - a. 「ターゲット内の選択可能な列」のリストから、DEPTNO 列を選択します。
 - b. 「移動」ボタン (>) をクリックし、DEPTNO 列を「索引で使用される列」リストに移動します。
 - c. 「主キーを作成するために選択済み列を使用する」を選択し、DEPTNO 列をターゲット表の主キーとして使用します。
8. 「メンバー・プロパティ」ウィンドウの「行フィルター」ページで、「WHERE ステートメント」フィールドに以下の文節を入力します。


```
DEPTNO >='E00'
```

この WHERE 文節は、特定の基準を満たす行だけを複製することを示します。ここでは、部門番号が“E00”より大きいか等しいという基準を満たす行が複製されます。この WHERE 文節により、ターゲット表には 9 行すべてではなく、3 行が含まれることになります。

9. 「メンバー・プロパティ」ウィンドウの「ターゲット表の表スペース」ページで、新しい TSTGDEPTCOPY 表スペースについて以下の情報を指定します。
 - 「表スペースの仕様」エリアで、「コンテナ名」フィールドをクリックし、TSTGDEPTCOPY 表スペースのコンテナ名を指定します。
 - 「表スペースの仕様」エリアで、「サイズ」フィールドを 1 に変更します。
 - 「表スペースの仕様」エリアで、「単位」フィールドを MB に変更します。
 - この新しい表スペースについてのその他の情報 (バッファ・プールを IBMDEFAULTBP に設定するなど) を指定します。

この新しい表スペースについてのその他の情報 (バッファ・プールを IBMDEFAULTBP に設定するなど) を指定することもできます。

10. 「OK」をクリックし、「メンバー・プロパティ」ウィンドウをクローズします。
11. 「サブスクリプション・セットの作成」ウィンドウの「スケジュール」ページで、分を 1 に変更し、アプライ・プログラムがこのサブスクリプション・セットを 1 分おきに処理するようにします。「レプリケーションの頻度」エリア内の「分」フィールドのスピン・ボタンを使って、1 分のインターバルを選択します (またはフィールドに 1 と入力します)。

「開始日付」、「開始時刻」、「時間に基づく」、および「相対タイミングの使用」のデフォルト値を保管します。
12. 「サブスクリプション・セットの作成」ウィンドウの「ステートメント」ページで「追加」をクリックし、「SQL ステートメントまたはプロシージャ呼び出しの追加」ウィンドウをオープンします。このウィンドウを使用して、サブスクリプション・セットの実行時に処理される SQL ステートメントを定義します。「SQL ステートメントまたはプロシージャ呼び出しの追加」ウィンドウで、以下の情報を入力します。
 - a. 「SQL ステートメント」フィールドに、次のように入力します。

```
DELETE FROM ASN.IBMSNAP_APPLYTRAIL WHERE LASTRUN  
< (CURRENT TIMESTAMP - 7 DAYS)
```

このステートメントは、アプライ・トレール表にある 7 日より前のレコードをすべて削除します。

アプライ・プログラムは、追加された SQL ステートメントを、サブスクリプション・セットを処理した後にターゲット・サーバーで実行します。SQL ステートメントはターゲット・サーバーで実行する必要があります。その理由は、アプ

ライ・コントロール・サーバーとターゲット・サーバーは一緒の場所に置かれており、アプライ・トレール表がアプライ・コントロール・サーバー内にあるためです。

ヒント: アプライ・プログラムは、サブスクリプション・セットに追加された SQL ステートメントまたはプロシーチャーを、サブスクリプション・サイクルごとに実行します。この例では、アプライ・プログラムがこのステートメントを 1 分おきに実行するので非効率です。実際に APPLYTRAIL 表からデータが削除されるのは、24 時間ごとになります (7 日より前のデータを削除)。

- b. 「**SQLSTATE**」フィールドに 02000 を入力し、「追加」をクリックします。この SQL 状態は、行が見つからない (row not found) というエラーが許容できるものであり、アプライ・プログラムがこのようなエラーを無視することを示します。

ヒント: このサブスクリプション・セットに関して、アプライ・プログラムに無視させたい SQL 状態を 10 個まで定義することができます。

- c. 「OK」をクリックして、「SQL ステートメントまたはプロシーチャー呼び出しの追加」ウィンドウをクローズします。
13. 「OK」をクリックし、「サブスクリプション・セットの作成」ウィンドウをクローズします。
14. 「メッセージ・ダイアログ」ウィンドウで「クローズ」をクリックします。このウィンドウは、アプライ・コントロール表を更新し、ターゲット表を作成する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
15. 「今実行するか、SQL を保管する」ウィンドウで「OK」をクリックすれば、SQL スクリプトは即時に実行されます。

SQL スクリプトをファイルに保管して後から使用することも、即時に実行することもできます。

- a. 「ファイルに保管」を選択します。
- b. 「仕様の保管」エリアに、ファイル名などの情報を記入します。
- c. 「適用」をクリックしてファイルに保管します。スクリプトが複数の部分からなる場合は、それぞれの部分が別のファイルに保管され、指定した名前に番号を追加した名前が使用されます。「今実行するか、SQL を保管する」ウィンドウはオープンされたままです。
- d. 「今実行する」を選択します。
- e. 「OK」をクリックしてスクリプトを実行し、「今実行するか、SQL を保管する」ウィンドウをクローズします。

SQL スクリプトをファイルに保管して後から使用することも、SQL スクリプトを保管して実行することもできます。

16. 「DB2 メッセージ」 ウィンドウに、スクリプトが SAMPLE サーバーと COPYDB サーバーの両方で正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
17. 「アプライ・コントロール・サーバー」フォルダーと COPYDB フォルダーを展開し、次に「サブスクリプション・セット」フォルダーをクリックします。「サブスクリプション・セット」フォルダーの内容ペインに、 DEPTSUB サブスクリプション・セットが表示されます。図 12 に、「サブスクリプション・セット」フォルダーの内容ペインに、 DEPTSUB サブスクリプション・セットが表示された例が示されています。

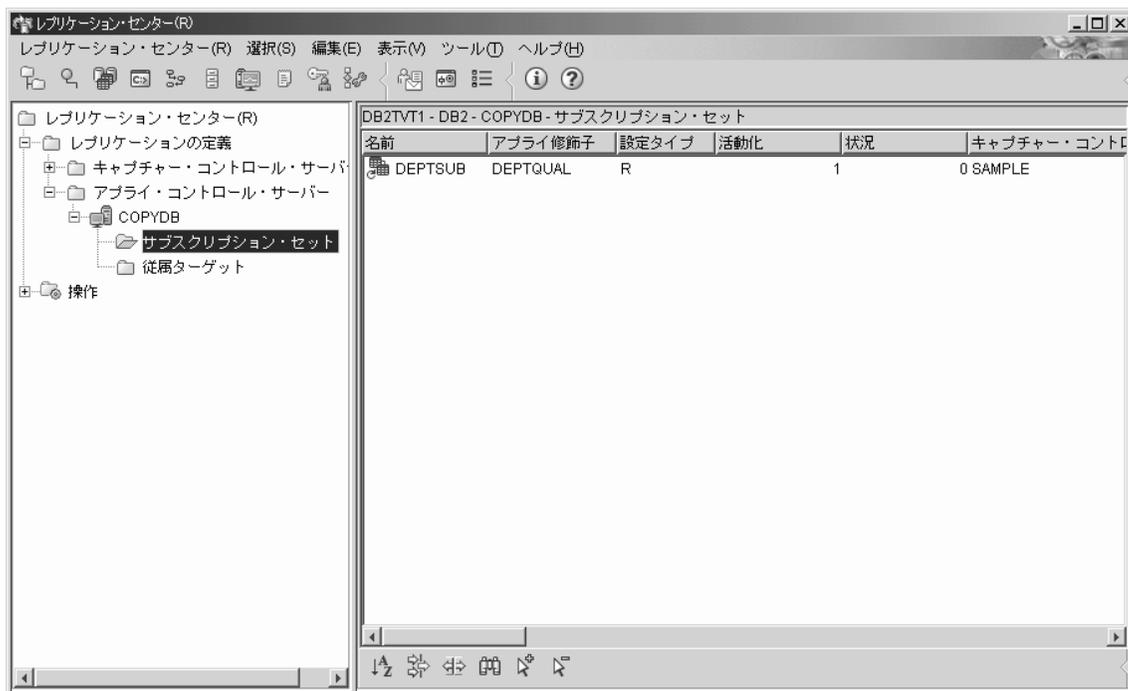


図 12. COPYDB データベースの DEPTSUB サブスクリプション・セットがリストされます。

ステップ 6: アプライ・パスワード・ファイルを作成する

アプライ・プログラムは、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、およびターゲット・サーバーと接続する必要があるため、ユーザー認証のためのパスワード・ファイルを作成する必要があります。パスワード・ファイルの内容は暗号化されているため、ユーザーはファイルの変更はできませんが、ファイルの読み取りができるのはアプライ・プログラムだけです。

パスワード・ファイルを作成するには、次のようにします。

1. Windows の「コマンド・プロンプト」ウィンドウをオープンし、`C:\%sqllib%\bin` ディレクトリーに変更します。
2. 次のコマンドを入力し、デフォルトのパスワード・ファイルを作成します。

```
asnpwd init using "path"
```

path は、パスワード・ファイルの作成に使用する、完全修飾されたディレクトリー・パスおよびファイル名です。コマンドが正常に完了したことを示すメッセージ ASN198II が表示されます。

たとえば、パスワード・ファイルを `c:\%sqllib%\repl` ディレクトリーに保管し、ファイル名を `asnpwd.aut` にする場合は、次のコマンドを入力します。

```
asnpwd init using "c:\%sqllib%\repl\asnpwd.aut"
```

ヒント: パスワード・ファイルは、アプライ・プログラムを始動するディレクトリーに作成してください。アプライ・プログラムの始動時に、パスワード・ファイルのファイル名 (PWDFILE キーワードを使用して) およびアプライ・プログラムが自分のログと作業ファイルを保管するディレクトリーの値を (APPLY_PATH キーワードを使用して) 指定します。アプライ・プログラムの作業ファイルの 1 つはパスワード・ファイルです。

3. 以下のコマンドを入力し、アプライ・プログラムが接続する必要があるデータベースごとに、ユーザー ID とパスワード情報を追加します。

```
asnpwd add alias SAMPLE id userid password password using "path"
```

userid は、キャプチャー・コントロール表とアプライ・コントロール表を更新するための権限を持つ、有効な DB2 ユーザー ID です。コマンドが正常に完了したことを示すメッセージ ASN198II が表示されます。

ステップ 7: シナリオのデータを複製する

レプリケーション・ソースを登録し、サブスクリプション・セットを作成した後、キャプチャー・プログラムとアプライ・プログラムを始動し、ターゲット表の初期フル・リフレッシュを実行し、変更キャプチャー・レプリケーションを開始します。

キャプチャー・プログラムを始動するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを選択します。キャプチャー・コントロール・サーバーの内容ペインに SAMPLE データベースが表示されます。
3. SAMPLE データベースを右クリックし、「キャプチャーの開始」を選択します。
4. 「キャプチャー開始」ウィンドウで、CAPTURE_PATH キーワードを選択します。「ログ・ファイルを保管するディレクトリー」フィールドに、キャプチャー・プログ

ラムが自分の出力 (作業ファイルとログ・ファイルを含む) を書き込むディレクトリーを入力します。キャプチャー・プログラムのその他のキーワードは何も変更する必要はありません。

5. 「キャプチャーの開始」ウィンドウで「**OK**」をクリックします。
6. 「今実行するか、コマンドを保管する」ウィンドウで「**OK**」をクリックすれば、コマンドは即時に実行されます。
7. 「DB2 メッセージ」ウィンドウに、コマンドが正常に実行されたというメッセージが表示されます。「**クローズ**」をクリックします。これでキャプチャー・プログラムが実行されますが、アプライ・プログラムがすべての登録済みの表のフル・リフレッシュを完了するまでは、登録済みの表の変更キャプチャーを開始しません。

アプライ・プログラムを始動するには、次のようにします。

1. 「**操作**」フォルダーを展開します。
2. 「**アプライ・コントロール・サーバー**」フォルダーを展開します。
3. 「**COPYDB**」フォルダーを展開します。
4. 「**アプライ修飾子**」フォルダーを選択します。アプライ修飾子の内容ペインに、サブスクリプション・セット **DEPTSUB** のアプライ修飾子 **DEPTQUAL** が表示されます。
5. **DEPTQUAL** アプライ修飾子を右クリックし、「**アプライの開始**」を選択します。
6. 「アプライの開始」ウィンドウに、次の情報を入力します。
 - a. 「**アプライ・プログラムを実行する場所**」エリアの「**システム**」リストから、アプライ・プログラムをそこで実行するシステムを選択します。
 - b. **APPLY_PATH** キーワードを選択します。アプライ・プログラムが自分のログと作業ファイルを保管するディレクトリーの値を入力します。
 - c. **PWDFILE** キーワードを選択します。アプライ・パスワード・ファイルのファイル名を入力します (たとえば、**asnpwd.aut**)。パスワード・ファイルのパスは、**APPLY_PATH** キーワードの値です。
 - d. アプライ・プログラムのその他のキーワードは何も変更しないでください。

ヒント: **LOADXIT** キーワードを指定して、**ASNLOAD** プログラムを呼び出すことができます。デフォルトの **ASNLOAD** プログラムは、**DB2** エクスポート (**EXPORT**) ユーティリティーを使ってソース表からデータをエクスポートし、**DB2** ロード (**LOAD**) ユーティリティーを使ってターゲット表のフル・リフレッシュを実行します。**ASNLOAD** を変更し、**IBM** またはベンダーのユーティリティーを呼び出すことができます。

7. 「アプライの開始」ウィンドウで「**OK**」をクリックします。
8. 必要ならば、アプライ・プログラムをそこで実行するシステムの、有効なユーザー **ID** とパスワードを、「今実行するか、コマンドを保管する」ウィンドウに入力します。

9. 「今実行するか、コマンドを保管する」ウィンドウで「OK」をクリックすれば、コマンドは即時に実行されます。
10. 「DB2 メッセージ」ウィンドウに、コマンドが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。これでアプライ・プログラムが実行されます。

1 回のレプリケーション・サイクル後に TGDEPTCOPY ターゲット表を表示すると、表 15 に示すデータと一致する結果が表示されます。表の内容を表示するには、次のいずれかの方法を使用します。

- DB2 コントロール・センターを使用します。
 1. DB2 インスタンスの「データベース」フォルダーを展開します。
 2. 「COPYDB」フォルダーを展開します。
 3. 「表」フォルダーを選択します。
 4. 内容ペインの TGDEPTCOPY 表を右クリックし、「サンプル内容」を選択します。
- DB2 コマンド・センターまたは DB2 コマンド・ウィンドウを使用して、次の SQL ステートメントを実行します。

```
SELECT * FROM schema.TGDEPTCOPY
```

表 15. TGDEPTCOPY 表

| DEPTNO | DEPTNAME | MGRNO | ADMRDEPT | LOCATION |
|--------|------------------|--------|----------|----------|
| E01 | SUPPORT SERVICES | 000050 | A00 | - |
| E11 | OPERATIONS | 000090 | E01 | - |
| E21 | SOFTWARE SUPPORT | 000100 | E01 | - |

レプリケーション環境における操作

レプリケーション環境の起動後、レプリケーション・ソース表に加えられる変更はターゲット表にレプリケーションされます。キャプチャー・プログラムとアプライ・プログラムの両方の状況を表示すれば、レプリケーションの待ち時間やレプリケーション環境についてのその他の情報を見ることができます。キャプチャー・プログラムとアプライ・プログラムは連続して実行できますが、停止したい場合もあります (たとえば、コントロール表を含む表スペースを使用するユーティリティを実行する場合など)。

ステップ 1: ソース表を更新する

Spiffy Computer Service 社で新しい 2 つの部門が発足したとします。1 つは技術情報部門 (technical writing)、もう 1 つは広報部門 (public relations) です。ターゲット表にはこの両方の部門が含まれます。

ソース表を次のように更新します。

1. 「スタート」→「プログラム」→「IBM DB2」→「コマンド・ウィンドウ」を選択し、DB2 コマンド・ウィンドウをオープンします。
2. 次のように入力して、ソース・サーバーに接続します。

```
DB2 CONNECT TO SAMPLE
```

3. 次のコマンドを入力し、それぞれ Enter キーを押して、2 つの新しい行 (各部門ごとに 1 行) を追加します。

```
DB2 INSERT INTO DEPARTMENT
VALUES ('F01','TECHNICAL WRITING','000110','F01',NULL)
DB2 INSERT INTO DEPARTMENT
VALUES ('G01','PUBLIC RELATIONS','000120','G01',NULL)
DB2 COMMIT
```

4. 次のように入力して、ターゲット・サーバーに接続します。

```
DB2 CONNECT TO COPYDB
```

5. 少なくとも 1 分ほど待ってから、次のコマンドを入力し、ターゲット・データベースに新しい行が複製されていることを確認します。

```
DB2 SELECT * FROM TGDEPTCOPY
```

サブスクリプション・セットは 1 分おきにレプリケーションの対象として適格になるので、1 分待つ必要があります。データが大量にある場合は、アプライ・プログラムがそのデータをターゲット表に適用するのにもう少し時間がかかることがあります。

表 16 にレプリケーションの結果を示します。2 つの新しい行が表に追加されています。

表 16. 変更を複製した後の TGDEPTCOPY 表

| DEPTNO | DEPTNAME | MGRNO | ADMRDEPT | LOCATION |
|--------|-------------------|--------|----------|----------|
| E01 | SUPPORT SERVICES | 000050 | A00 | - |
| E11 | OPERATIONS | 000090 | E01 | - |
| E21 | SOFTWARE SUPPORT | 000100 | E01 | - |
| F01 | TECHNICAL WRITING | 000110 | F01 | - |
| G01 | PUBLIC RELATIONS | 000120 | G01 | - |

ステップ 2: キャプチャー・プログラムの状況を表示する

レプリケーション・センターを使用して、キャプチャー・プログラムについて次のような状況情報を表示します。

- キャプチャー・プログラムが出したエラー・メッセージ
- キャプチャー・プログラムのスループットの分析
- キャプチャー・プログラムの現在の待ち時間のサマリー
- キャプチャー・プログラムの現在の操作状況

このような状況から、キャプチャー・プログラムがどのように実行されているかをスナップショットとして見ることができます。

キャプチャー・プログラムの状況を照会するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを選択します。
3. 内容ペインの SAMPLE データベースを右クリックし、「状況の照会」を選択します。
4. 「検索」をクリックし、現行情報を表示します。

キャプチャー・プログラムのスループット分析を表示するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを選択します。
3. 内容ペインの SAMPLE データベースを右クリックし、「キャプチャー・スループット分析の表示」を選択します。
4. 「キャプチャー・スループット分析」ウィンドウで、次の情報を見ることができます。
 - DB2 ログから CD 表に挿入された行数、または CHGONLY キーワードの設定などの理由からスキップされた行数
 - CD 表から除去された行数
 - キャプチャー・プログラムによりコミットされたトランザクションの数
 - 特定の時間インターバル内にキャプチャー・プログラムが使用したメモリー
5. 「検索」をクリックし、現行情報を表示します。

キャプチャー・プログラムの現在の待ち時間のサマリーを表示するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを選択します。
3. 内容ペインの SAMPLE データベースを右クリックし、「キャプチャー待ち時間の表示」を選択します。
4. 「キャプチャー待ち時間」ウィンドウに、特定の時間インターバル内のキャプチャー・プログラムの待ち時間の平均、最小、および最大が表示されます。
5. 「検索」をクリックし、現行情報を表示します。

ステップ 3: アプライ・プログラムの状況を表示する

レプリケーション・センターを使用して、アプライ・プログラムについて次のような状況情報を表示します。

- サブスクリプション・セット情報のサマリー (正常終了したサブスクリプション・セット、と失敗したサブスクリプション・セットを含む)

- アプライ・プログラムのパフォーマンスのサマリー
- レプリケーションのエンドツーエンド待ち時間のサマリー
- アプライ・プログラムの現在の操作状況

このような状況から、アプライ・プログラムがどのように実行されているかをスナップショットとして見ることができます。

アプライ・プログラムの状況を照会するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「アプライ・コントロール・サーバー」フォルダーを選択します。
3. 「COPYDB」フォルダーを展開します。
4. 「アプライ修飾子」フォルダーを選択します。
5. 内容ペインの DEPTQUAL アプライ修飾子を右クリックし、「状況の照会」を選択します。
6. 「検索」をクリックし、現行情報を表示します。

アプライ・プログラムのパフォーマンスのサマリーを表示するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「アプライ・コントロール・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「アプライ修飾子」フォルダーを選択します。
5. 内容ペインの DEPTQUAL アプライ修飾子を右クリックし、「アプライ・スループット分析」を選択します。
6. 「アプライ・スループット分析」ウィンドウに、次の情報が表示されます。
 - アプライ・プログラムが CD 表からフェッチした行数
 - それぞれのサブスクリプション・セットごとの経過時間
7. 「検索」をクリックし、現行情報を表示します。

エンドツーエンドのレプリケーション待ち時間のサマリーを表示するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「アプライ・コントロール・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「アプライ修飾子」フォルダーを選択します。
5. 内容ペインの DEPTQUAL アプライ修飾子を右クリックし、「エンドツーエンド待ち時間の表示」を選択します。
6. 「エンドツーエンド待ち時間の表示」ウィンドウに、特定の時間インターバル内のサブスクリプション・セットごとの待ち時間の平均が表示されます。

7. 「検索」をクリックし、現行情報を表示します。

ステップ 4: キャプチャー・プログラムとアプライ・プログラムを停止する

レプリケーション環境の保守における 1 つの重要な部分は、定期的なデータベース保守です。そのような保守のために、キャプチャー・プログラムおよびアプライ・プログラムを停止しなければならない場合があります。たとえば、キャプチャー・プログラムおよびアプライ・プログラムで使用される表スペースを直接使用するユーティリティを実行するには、まずそれらのプログラムを停止しなければなりません。

キャプチャー・プログラムを停止するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「キャプチャー・コントロール・サーバー」フォルダーを選択します。
3. 内容ペインの SAMPLE データベースを右クリックし、「キャプチャーの停止」を選択します。
4. 「キャプチャーの停止」ウィンドウで「OK」をクリックします。
5. 「今実行するか、コマンドを保管する」ウィンドウで「OK」をクリックすれば、コマンドは即時に実行されます。
6. 「DB2 メッセージ」ウィンドウに、コマンドが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。これでキャプチャー・プログラムは停止しました。

アプライ・プログラムを停止するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「アプライ・コントロール・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「アプライ修飾子」フォルダーを選択します。
5. 内容ペインの DEPTQUAL アプライ修飾子を右クリックし、「アプライの停止」を選択します。
6. 「アプライの停止」ウィンドウで、「OK」をクリックします。
7. 「今実行するか、コマンドを保管する」ウィンドウで、アプライ・プログラムを始動したディレクトリー（「ディレクトリー」フィールドの APPLY_PATH キーワードに指定した値）を入力するか、またはブラウズ・ボタンでパスを選択します。「OK」をクリックします。
8. 「今実行するか、コマンドを保管する」ウィンドウで「OK」をクリックすれば、コマンドは即時に実行されます。
9. 「DB2 メッセージ」ウィンドウに、コマンドが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。これでアプライ・プログラムは停止しました。

キャプチャー・プログラムとアプライ・プログラムが停止したら、データベースに対して DB2 ユーティリティを実行することができます。ユーティリティの実行はこのシナリオの有効範囲外です。

レプリケーションのモニター

レプリケーション環境の起動後、キャプチャー・プログラムとアプライ・プログラムが正常に実行されているかどうかを知りたい場合があります。また、ある種のレプリケーション・エラーについては、エラーの発生時に自動的に通知するようにセットアップしておくことも必要です。

レプリケーション・センターを使用して、キャプチャー・プログラムとアプライ・プログラムの状況を照会し、ある種の統計を表示すれば、この両方のプログラムが正常に実行されているかどうかを知ることができます。また、キャプチャー・プログラムまたはアプライ・プログラムがある種のレプリケーション・エラーを検出した場合に、レプリケーション・アラート・モニターがユーザーに通知するようにセットアップすることもできます。

ステップ 1: モニター・プログラムのレプリケーション・コントロール表を作成する

レプリケーション・アラート・モニター・プログラムは、レプリケーション・モニター・コントロール表から現行のモニター情報を読み取り、またモニター・プログラムの状況をこの表に保管します。モニター・サーバーとして機能するデータベースには、モニター・コントロール表を含める必要があります。

モニター・コントロール表を作成するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・コントロール・サーバー」フォルダーを右クリックし、「モニター・コントロール表の作成」を選択します。
3. 「サーバーの選択」ウィンドウで、COPYDB データベースを選択します。このデータベースはモニター・コントロール・サーバーになります。
4. 「モニター・コントロール表の作成」ウィンドウで、IBMSNAP_CONTACTS コントロール表を選択し、RELPMONTS1 表スペース・プロパティの情報を記入します。コンテナ名の隣にあるブラウザ・ボタンをクリックし、この表スペースのロケーションをカスタマイズします。この表スペースについてのその他の情報 (バッファー・プールを IBMDEFAULTBP に設定するなど) を指定することもできます。「モニター・コントロール表の作成」ウィンドウは、311 ページの図 13 に似たウィンドウになります。

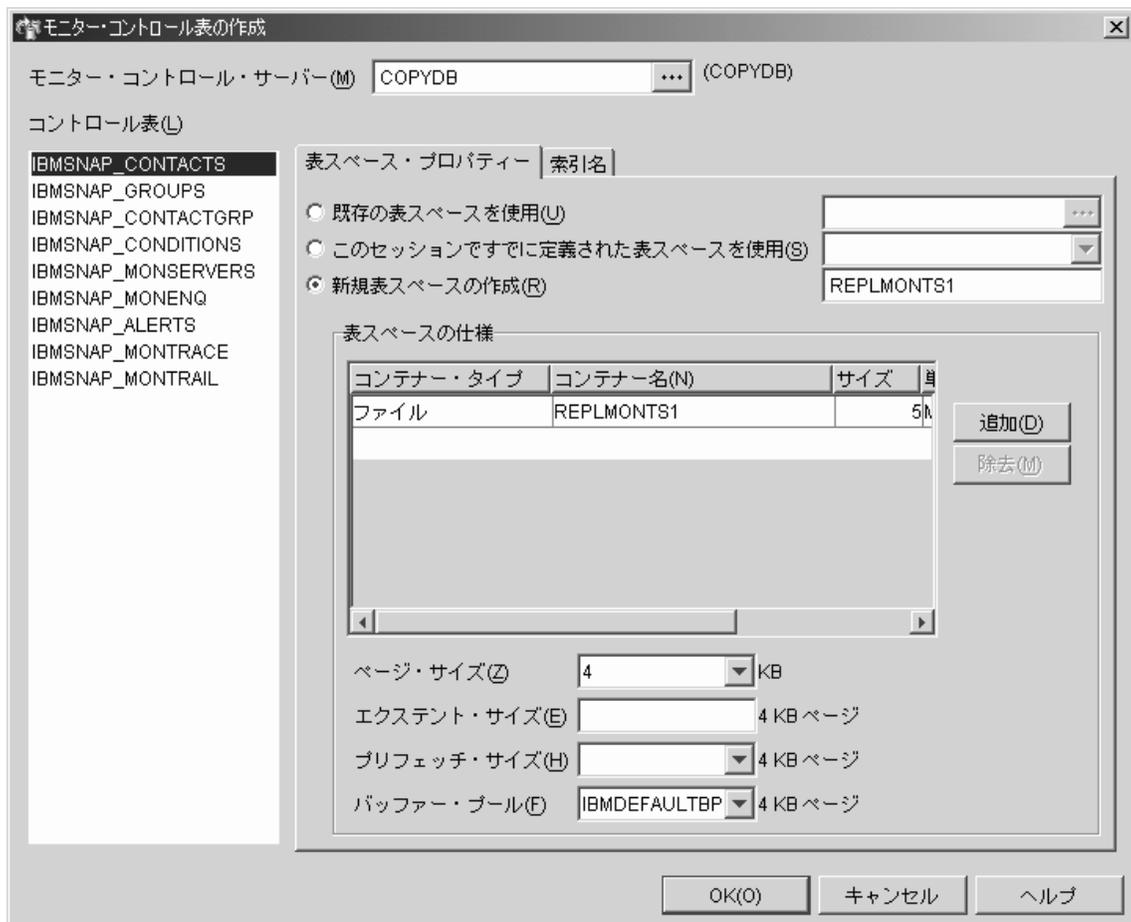


図 13. 「モニター・コントロール表の作成」ウィンドウ

- 「モニター・コントロール表の作成」ウィンドウで、IBMSNAP_ALERTS コントロール表を選択し、RELPMONTS2 表スペース・プロパティの情報を記入します。
- 「モニター・コントロール表の作成」ウィンドウで、IBMSNAP_MONTRACE コントロール表を選択し、RELPMONTS3 表スペース・プロパティの情報を記入します。
- 「モニター・コントロール表の作成」ウィンドウで「OK」をクリックし、その他のコントロール表情報 (索引名など) についてデフォルト値を受け入れます。
- 「メッセージ・ダイアログ」ウィンドウで「クローズ」をクリックします。このウィンドウは、モニター・コントロール表を作成する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
- 「今実行するか、SQL を保管する」ウィンドウで「OK」をクリックすれば、SQL スクリプトは即時に実行されます。

10. 「DB2 メッセージ」ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
11. 「モニター・コントロール・サーバー」フォルダーを展開します。フォルダーの下に COPYDB データベースが表示されます。

ステップ 2: レプリケーション・アラートの連絡先を作成する

レプリケーション・アラート・モニター・プログラムは、キャプチャー・プログラムおよびアプライ・プログラムの特定のアクティビティを検出した時に、アラートを出すことができます。特定のアラート条件について、個別の連絡先を作成するか、またはレプリケーション・アラート・モニターが複数の人にアラートを出す必要がある場合には、連絡先のグループを作成することができます。

連絡先を作成するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・コントロール・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「連絡先」フォルダーを右クリックし、「連絡先」→「担当者」を選択します。
5. 「連絡先の作成」ウィンドウで、ここでは担当者として自分の名前と E メールアドレスを指定してください。「OK」をクリックして、ノートブックをクローズします。
6. 「メッセージと SQL スクリプト」ウィンドウで「クローズ」をクリックします。このウィンドウは、モニター・コントロール表を更新する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
7. 「今実行するか、SQL を保管する」ウィンドウで「OK」をクリックすれば、SQL スクリプトは即時に実行されます。
8. 「DB2 メッセージ」ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
9. 「連絡先」フォルダーをクリックします。定義した連絡先が「連絡先」の内容ペインに表示されます。

ステップ 3: キャプチャー・プログラムのアラート条件を選択する

レプリケーション・アラート・モニター・プログラムは、キャプチャー・プログラムの特定のアクティビティをモニターすることができます。どのアクティビティをモニターするかを選択する必要があります。これらのアクティビティごとに、アラート条件を選択します。キャプチャー・プログラムがこの条件を検出すると、レプリケーション・アラート・モニターは、アラート条件に定義された連絡先にアラートを送信します。

キャプチャー・プログラムのモニター定義を作成するには、次のようにします。

1. 「操作」フォルダーを展開します。

2. 「モニター・コントロール・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「モニター修飾子」フォルダーを右クリックし、「キャプチャー・スキーマ用のアラート条件を選択」を選択します。
5. 「キャプチャー・スキーマ用のアラート条件を選択」ウィンドウで、次の情報を指定します。
 - a. 「モニター修飾子」フィールドに、MON1 と入力します。
 - b. 「キャプチャー・コントロール・サーバー」フィールドのブラウズ・ボタンをクリックし、モニターしたいキャプチャー・コントロール・サーバーを選択します。「キャプチャー・コントロール・サーバーの選択」ウィンドウで SAMPLE データベースを選択し、「OK」をクリックします。
 - c. 「追加」をクリックして「選択済みキャプチャー・スキーマ」リストに ASN を追加します。
 - d. 「キャプチャー・スキーマの選択」ウィンドウで、「すべて検索」をクリックします。リストから ASN を選択し、「OK」をクリックします。
 - e. 「アラート条件」リストで、「エラー」を選択します。
 - f. 「値」エリアで、「連絡先」フィールドのブラウズ・ボタンをクリックし、このアラート条件の連絡先を選択します。
 - g. 「連絡先または連絡先グループの選択」ウィンドウで、ステップ 2 で作成した連絡先を選択し、「OK」をクリックしてウィンドウをクローズします。
 - h. 「OK」をクリックし、「キャプチャー・スキーマ用のアラート条件を選択」ウィンドウをクローズします。
6. 「メッセージと SQL スクリプト」ウィンドウで「クローズ」をクリックします。このウィンドウは、モニター・コントロール表を更新する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
7. 「今実行するか、SQL を保管する」ウィンドウで「OK」をクリックすれば、SQL スクリプトは即時に実行されます。
8. 「DB2 メッセージ」ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
9. 「COPYDB」フォルダーを展開して、「モニター修飾子」フォルダーを展開し、「MON1」フォルダーを選択します。定義したアラート条件が、モニター修飾子の内容ペインに表示されます。

ステップ 4: アプライ・プログラムのアラート条件を選択する

レプリケーション・アラート・モニター・プログラムは、アプライ・プログラムの特定のアクティビティーをモニターすることができます。どのアクティビティーをモニターするかを選択する必要があります。これらのアクティビティーごとに、アラート条件を選択します。アプライ・プログラムがこの条件を検出すると、レプリケーション・アラート・モニターは、アラート条件に定義された連絡先にアラートを送信します。

アプライ・プログラムのモニター定義を作成するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「モニター修飾子」フォルダーを右クリックし、「アプライ修飾子またはサブスクリプション・セット用のアラート条件を選択」を選択します。
5. 「アプライ修飾子またはサブスクリプション・セット用のアラート条件を選択」ウィンドウで、次の情報を指定します。
 - a. 「モニター修飾子」フィールドに MON1 と入力します (キャプチャー・プログラムのアラート条件を作成しなかった場合)。
 - b. 「アプライ・コントロール・サーバー」フィールドのブラウズ・ボタンをクリックし、モニターしたいアプライ・コントロール・サーバーを選択します。「アプライ・コントロール・サーバーの選択」ウィンドウで COPYDB データベースを選択し、「OK」をクリックします。
 - c. 「追加」をクリックして、DEPTSUB を「サブスクリプション・セットの選択」リストに追加します。
 - d. 「サブスクリプション・セットを追加」ウィンドウで、「すべて検索」をクリックします。リストから DEPTSUB を選択し、「OK」をクリックします。
 - e. 「アラート条件」リストで、「フル・リフレッシュが行われました」を選択し、す。
 - f. 「値」エリアで、「連絡先」フィールドのブラウズ・ボタンをクリックし、このアラート条件の連絡先を選択します。
 - g. 「連絡先または連絡先グループの選択」ウィンドウで、ステップ 2 で作成した連絡先を選択し、「OK」をクリックしてウィンドウをクローズします。
 - h. 「アプライ修飾子またはサブスクリプション・セットのアラート条件を選択」ウィンドウで、「OK」をクリックします。
6. 「メッセージと SQL スクリプト」ウィンドウで「クローズ」をクリックします。このウィンドウは、モニター・コントロール表を更新する SQL スクリプトの生成結果を表示します。エラーがある場合は、このウィンドウに表示されます。
7. 「今実行するか、SQL を保管する」ウィンドウで「OK」をクリックすれば、SQL スクリプトは即時に実行されます。
8. 「DB2 メッセージ」ウィンドウに、スクリプトが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。
9. 「COPYDB」フォルダーを展開して、「モニター修飾子」フォルダーを展開し、「MON1」フォルダーを選択します。定義したアラート条件が、モニター修飾子の内容ペインに表示されます。315 ページの図 14 に、「モニター修飾子」フォルダーに MON1 モニター修飾子が表示された内容ペインの例を示します。

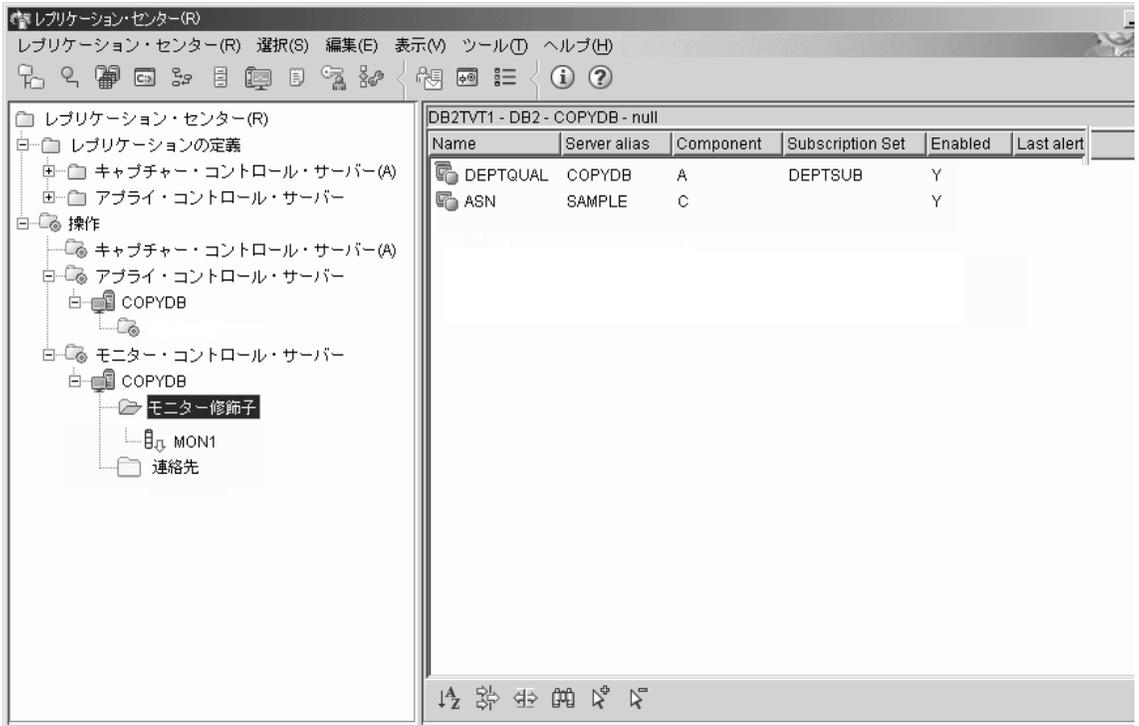


図 14. モニター修飾子 MON1 は COPYDB データベースにリストされます。

ステップ 5: モニター修飾子のレプリケーション・アラート・モニターを開始する

キャプチャーまたはアプライ・プログラムのアラート条件を選択した後、レプリケーション・アラート・モニター・プログラムを始動して、キャプチャーおよびアプライ・プログラムのアクティビティーを、モニター修飾子に関連付けられた特定の条件についてモニターすることができます。キャプチャーまたはアプライ・プログラムが指定された条件の 1 つを検出すると、レプリケーション・アラート・モニターは、アラート条件に定義された連絡先にアラートを送信します。

レプリケーション・アラート・モニターを開始するには、次のようにします。

1. 「操作」フォルダーを展開します。
2. 「モニター・サーバー」フォルダーを展開します。
3. 「COPYDB」フォルダーを展開します。
4. 「モニター修飾子」フォルダーを展開します。
5. モニター修飾子 MON1 を右クリックし、「モニターの開始」を選択します。

6. 「モニターの開始」ウィンドウに、次の情報を入力します。
 - a. **MONITOR_PATH** キーワードを選択します。レプリケーション・アラート・モニターが自分のログと作業ファイルを保管するディレクトリーの値を入力します。

ヒント: **MONITOR_PATH** キーワードの値は、**APPLY_PATH** キーワードの値と同じにし、レプリケーション・アラート・モニターとアプライ・プログラムが同じパスワード・ファイルを使用できるようにします。
 - b. **EMAIL_SERVER** キーワードを選択します。自分の E メール・サーバー名を入力してください。
 - c. **MONITOR_ERRORS** キーワードを選択します。レプリケーション・アラート・モニターに問題が起こった時に知らせる相手として、自分の E メール・アドレスを入力するか、またはブラウズ・ボタンをクリックして、「連絡先または連絡先グループの選択」ウィンドウをオープンします。
 - d. 「**OK**」をクリックして、「モニターの開始」ウィンドウをクローズします。
7. 「今実行するか、コマンドを保管する」ウィンドウで「**OK**」をクリックすれば、コマンドは即時に実行されます。
8. 「DB2 メッセージ」ウィンドウに、コマンドが正常に実行されたというメッセージが表示されます。「**クローズ**」をクリックします。

レプリケーション・アラート・モニターがモニターしたアラートを表示するには、次のようにします。

1. 「**操作**」フォルダーを展開します。
2. 「**モニター・サーバー**」フォルダーを展開します。
3. 「**COPYDB**」フォルダーを展開します。
4. 「**モニター修飾子**」フォルダーを展開します。
5. モニター修飾子 **MON1** を選択します。
6. モニター修飾子の内容ペインでアラート条件の 1 つを右クリックし、「**アラート条件**」を選択します。
7. 「アラートの表示」ウィンドウで、時刻範囲を指定し、「**検索**」をクリックします。

レプリケーション・アラート・モニターを停止するには、次のようにします。

1. 「**操作**」フォルダーを展開します。
2. 「**モニター・サーバー**」フォルダーを展開します。
3. 「**COPYDB**」フォルダーを展開します。
4. 「**モニター修飾子**」フォルダーを展開します。
5. モニター修飾子 **MON1** を右クリックし、「**モニターの停止**」を選択します。
6. 「今実行するか、コマンドを保管する」ウィンドウで「**OK**」をクリックすれば、コマンドは即時に実行されます。

7. 「DB2 メッセージ」ウィンドウに、コマンドが正常に実行されたというメッセージが表示されます。「クローズ」をクリックします。

第 3 部 レプリケーションのリファレンス

本書の第 3 部は、以下の章で構成されています。

321 ページの『第 16 章 レプリケーション・オブジェクトの命名規則』では、レプリケーション・オブジェクトに有効な名前の指定方法について説明します。

323 ページの『第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)』では、DB2 レプリケーションの経験のあるユーザーが、UNIX、Windows、および z/OS オペレーティング・システムでレプリケーション操作をするために、レプリケーション・センターの代わりに使用することができるコマンドについて説明します。

365 ページの『第 18 章 レプリケーション用のシステム・コマンド (OS/400)』では、OS/400 オペレーティング・システムで、ローカルに、レプリケーションをセットアップ、管理、および保守するためのコマンドについて説明します。

461 ページの『第 19 章 レプリケーション・プログラムの操作 (z/OS)』では、z/OS で JCL やシステム起動タスクを使用して、レプリケーション・プログラムを開始および操作する方法について説明します。

467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』では、Windows オペレーティング・システムで、レプリケーション・プログラムをサービスとして開始する方法について説明します。

471 ページの『第 21 章 各種オペレーティング・システムでのレプリケーション・プログラムのスケジューリング』では、さまざまなオペレーティング・システムで、レプリケーション・プログラムをスケジューリングする方法について説明します。

473 ページの『第 22 章 DB2 レプリケーション・コンポーネントの通信方法』では、レプリケーション・コンポーネントが相互通信のためにコントロール表を使用する方法について説明します。

481 ページの『第 23 章 表構造』では、さまざまなレプリケーション・サーバー上にあるレプリケーション表の構造について説明します。

549 ページの『第 24 章 レプリケーション・メッセージ』には、UNIX、Windows、および z/OS プラットフォーム用の、すべてのレプリケーション・メッセージがリストされています。

第 16 章 レプリケーション・オブジェクトの命名規則

次の表は、レプリケーション・オブジェクトの名前の制限を示しています。

表 17. レプリケーション・オブジェクトの名前の制限

| オブジェクト | 名前の制限 |
|-------------|--|
| ソース表とターゲット表 | データベース管理システムの命名規則に従ってください。 |
| ソース列とターゲット列 | データベース管理システムの命名規則に従ってください。(すべての変更前イメージ列には、1 文字の接頭部が追加されます。変更前イメージ列の名前が未確定にならないように、ソース列名が 29 文字の固有なものになるようにしてください。) |
| キャプチャー・スキーマ | UNIX、Windows、z/OS の場合: キャプチャー・スキーマには、30 文字以下のストリングを使用できます (ただし、DB2 for z/OS および DB2 for OS/390 の場合の制限は 18 文字)。 OS/400 の場合: キャプチャー・スキーマ (CAPCNTLIB) には 10 文字以下のストリングを使用できます。 |
| アプライ修飾子 | UNIX、Windows、z/OS の場合: アプライ修飾子は、18 文字以下のストリングにできます。 OS/400 の場合: アプライ修飾子には 18 文字以下のストリングを使用できますが、アプライ・ジョブの長さは 10 文字までにしかできないため、特定のアプライ修飾子について、最初の 10 文字は固有でなければなりません。 |
| モニター修飾子 | UNIX、Windows、z/OS の場合: モニター修飾子には 18 文字以下のストリングを使用できます。 |

また、レプリケーション・オブジェクトの名前には、次の有効な文字のみを使用するようにしてください。

- A から Z (上段シフト文字)
- a から z (小文字)
- 数表示 (0 から 9)
- 下線文字 "_"

ブランクは使用できません。また、コロン ":" および正符号 "+" などの特殊文字も使用できません。

レプリケーション・システム・コマンドおよびレプリケーション・センターは、デフォルトとして、ユーザーから指定されたすべての名前を大文字に変換します。入力された名前のおりに正確に大文字と小文字を維持するには、大文字小文字混合文字の名前を二重引用符 (またはターゲット・システムで使用できるように構成されている他の文字) で囲んでください。たとえば、myqual または MyQual または MYQUAL と入力すると、名前は MYQUAL として保管されます。これらの同じ名前を二重引用符で囲んで入力する

と、myqual または MyQual または MYQUAL としてそれぞれ保管されます。オペレーティング・システムによっては二重引用符が認識されないことがあります。その場合は、エスケープ文字としてバックスラッシュまたは円記号 (¥) を使用する必要があります。

Windows オペレーティング・システムの場合は、同じ名前を区別するために固有パスを使用する必要があります。たとえば、3 つのアプライ修飾子、myqual、MyQual、および MYQUAL を使用するとします。3 つの名前は同じ文字を使用していますが、大文字と小文字が異なります。これら 3 つの修飾子が同じアプライ・パスにあると、名前の競合が発生します。

重要: キャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニター用に Windows サービスをセットアップするときには、キャプチャー・スキーマ、アプライ修飾子、およびモニター修飾子に固有の名前を使用する必要があります。大文字と小文字を使用してこれらの名前を区別することはできません。

第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)

この章では、以下の 1 つ以上のオペレーティング・システムの下で実行する、レプリケーション・コマンドについて説明します。

- UNIX
- Windows
- z/OS

これらのコマンドはすべて `asn` という接頭部を持ち、オペレーティング・システムのコマンド・プロンプトか、シェル・スクリプトに入力されます。コマンドの 1 つ、**asnanalyze** は、OS/400 オペレーティング・システムにあるリモート・データにも使用できます。

この章には、コマンド別のセクションがあります。各セクションには、コマンドの簡単な記述、構文図、およびパラメーターとそれに対応する定義の表があります。各セクションの最後には、コマンドの使用例と、関連情報への相互参照があります。

コマンドには、以下のものがあります。

- 324 ページの『`asnacmd`: アプライの操作 (UNIX、Windows、z/OS)』
- 325 ページの『`asnanalyze`: アナライザーの操作 (UNIX および Windows)』
- 328 ページの『`asnapply`: アプライの開始 (UNIX、Windows、z/OS)』
- 334 ページの『`asnccap`: キャプチャーの開始 (UNIX、Windows、z/OS)』
- 340 ページの『`asnccmd`: キャプチャーの操作 (UNIX、Windows、z/OS)』
- 346 ページの『`asnmcmd`: レプリケーション・アラート・モニターの操作 (UNIX、Windows、z/OS)』
- 348 ページの『`asnmon`: レプリケーション・アラート・モニターの開始 (UNIX、Windows、z/OS)』
- 352 ページの『`asnpwd`: パスワード・ファイルの保守 (UNIX および Windows)』
- 355 ページの『`asnscrt`: キャプチャー、アプライ、またはレプリケーション・アラート・モニターを開始する DB2 レプリケーション・サービスの作成 (Windows のみ)』
- 357 ページの『`asnsdrop`: DB2 レプリケーション・サービスのドロップ (Windows のみ)』
- 358 ページの『`asntrc`: レプリケーション・トレース機能の操作 (UNIX、Windows、z/OS)』

asnacmd: アプライの操作 (UNIX、Windows、z/OS)

UNIX、Windows、および z/OS 上の UNIX System Services (USS) でアプライ・プログラムを操作するには、**asnacmd** コマンドを使用します。このコマンドは、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

asnacmd コマンドを使用してアプライ・プログラムを操作する構文

```
▶▶▶asnacmd—apply_qual=apply_qualifier—[control_server=db_name]—[status—stop]▶▶▶
```

表 18 は、呼び出しパラメーターを定義します。

表 18. UNIX、Windows、および z/OS オペレーティング・システム用 *asnacmd* 呼び出しパラメーター定義

| パラメーター | 定義 |
|-----------------------------------|--|
| apply_qual=apply_qualifier | <p>アプライ・プログラムが、処理されるサブスクリプション・セットの識別に使用するアプライ修飾子を指定します。</p> <p>アプライ修飾子を指定してください。入力する値は、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の APPLY_QUAL 列の値と一致する必要があります。アプライ修飾子名には大文字小文字の区別があり、最大 18 文字です。</p> |
| control_server=db_name | <p>サブスクリプション定義とアプライ・コントロール表が存在するアプライ・コントロール・サーバーの名前を指定します。</p> <p>UNIX および Windows の場合: アプライ・コントロール・サーバーを指定しないと、このパラメーターはデフォルトで DB2DBDFT 環境変数の値になります。</p> <p>z/OS の場合: コントロール・サーバー・パラメーターは、コントロール・サーバーに接続するデータベース・サーバーの名前です。</p> |
| status | <p>アプライ内の各スレッド (管理およびワーカー) の状態を示すメッセージを受け取ることを指定します。</p> |
| stop | <p>アプライ・プログラムを通常の方法で停止することを指定します。</p> |

asnacmd の例

次の例は、**asnacmd** コマンドの使用法を示しています。

例 1

それぞれのアプライ・スレッドの状態についてメッセージを受け取る場合のコマンド例です。

```
asnacmd apply_qual=AQ1 control_server=dbx status
```

例 2

アプライ・プログラムを停止するには、次のようにします。

```
asnacmd apply_qual=AQ1 control_server=dbx stop
```

関連作業:

- 461 ページの 『第 19 章 レプリケーション・プログラムの操作 (z/OS)』

関連資料:

- 409 ページの 『ENDDPRAPY: アプライ・プログラムの停止 (OS/400)』
- 438 ページの 『STRDPRAPY: アプライ・プログラムの始動 (OS/400)』

asnanalyze: アナライザーの操作 (UNIX および Windows)

レプリケーション・コントロール表の状態についてのレポートを生成するには、**asnanalyze** コマンドを使用します。このコマンドは、任意のオペレーティング・システムにあるレプリケーション・コントロール表を分析します。OS/400 オペレーティング・システムでも可能ですが、コマンドは UNIX または Windows から呼び出す必要があります。

コマンドを呼び出す場合、**asnanalyze** コマンドと最初のパラメーターの間にはスペースを 1 つ入れる必要があります。パラメーターなしでコマンドを実行すると、画面にコマンド・ヘルプが表示されます。

asnanalyze コマンドを使用してアナライザーを操作するための構文

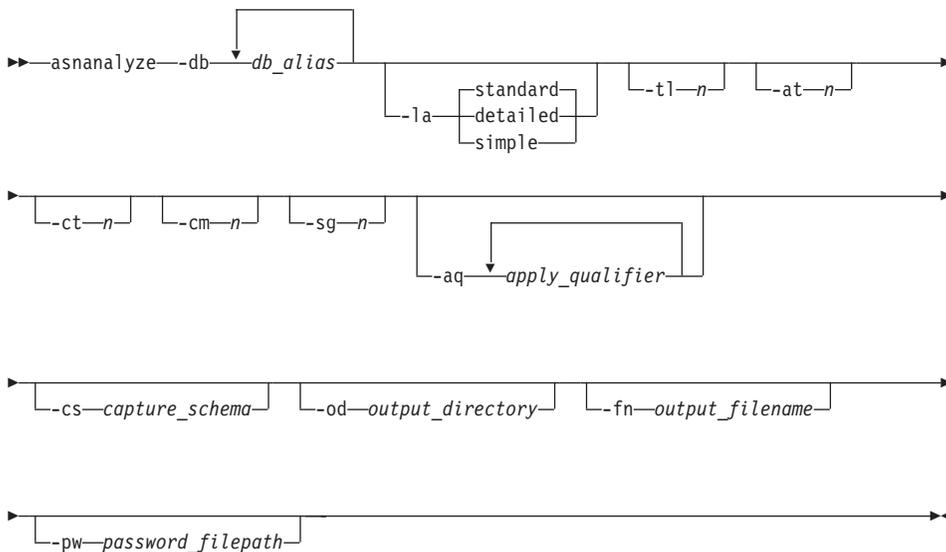


表 19 は、呼び出しパラメーターを定義します。

表 19. UNIX および Windows オペレーティング・システム用 asnanalyze 呼び出しパラメーター定義

| パラメーター | 定義 |
|-------------------------------------|---|
| -db <i>db_alias</i> | <p>キャプチャー・コントロール・サーバー、ターゲット・サーバー、およびアプライ・コントロール・サーバーを指定します。</p> <p>データベース別名を少なくとも 1 つ指定する必要があります。複数のデータベース別名がある場合は、ブランク・スペースを使用して値を区切ります。</p> |
| -la <i>level_of_analysis</i> | <p>報告される分析のレベルを指定します。</p> <p>standard (デフォルト) コントロール表の内容および、キャプチャー・プログラムおよびアプライ・からの状況情報を含むレポートを生成します。</p> <p>detailed 標準レポートの情報に加えて次の情報を生成します。</p> <ul style="list-style-type: none"> 変更データ (CD) 表および作業単位 (UOW) 表のブルーニング情報。 DB2 for z/OS 表スペースのパーティション化および圧縮情報 サブスクリプション・キーのためのターゲット索引の分析 <p>simple 標準レポートで情報を生成しますが、サブカラムの詳細は含まれません。</p> |
| -tl <i>n</i> | アプライ・トレール (IBMSNAP_APPLYTRAIL) 表から検索するエントリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 日です。 |
| -at <i>n</i> | アプライ・トレース (IBMSNAP_APPLYTRACE) 表から検索するエントリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 日です。 |
| -ct <i>n</i> | キャプチャー・トレース (IBMSNAP_CAPTRACE) 表から検索するエントリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 日です。 |
| -cm <i>n</i> | キャプチャー・モニター (IBMSNAP_CAPMON) 表から検索するエントリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 日です。 |
| -sg <i>n</i> | シグナル (IBMSNAP_SIGNAL) 表から検索するエントリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 日です。 |
| -aq <i>apply_qualifier</i> | <p>分析する特定のサブスクリプション・セットを識別するアプライ修飾子を指定します。</p> <p>複数のアプライ修飾子を指定することができます。複数のアプライ修飾子がある場合は、ブランク・スペースを使用して値を区切ります。アプライ修飾子を指定しないと、指定されたデータベース別名のすべてのサブスクリプション・セットが分析されます。</p> |
| -cs <i>capture_schema</i> | <p>分析するキャプチャー・スキーマの名前を指定します。</p> <p>このパラメーターを使用する場合、指定できるキャプチャー・スキーマは 1 つだけです。</p> |
| -od <i>output_directory</i> | アナライザー・レポートを保管するディレクトリーを指定します。デフォルトは、現行ディレクトリーです。 |

表 19. UNIX および Windows オペレーティング・システム用 *asnanalyze* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|-------------------------------------|---|
| -fn <i>output_filename</i> | アナライザー・レポート出力を含むファイルの名前を指定します。 アナライザーを実行するオペレーティング・システムのファイル名規則を使用します。ファイル名がすでに存在する場合、ファイルは上書きされます。デフォルトのファイル名は、 <i>asnanalyze.htm</i> です。 |
| -pw <i>password_filepath</i> | パスワード・ファイルの名前とパスを指定します。このパラメーターを指定しないと、アナライザーは現行ディレクトリーで <i>asnpwd.aut</i> ファイルを探します。 |

asnanalyze の例

次の例は、**asnanalyze** コマンドの使用法を示しています。

例 1

proddb1 という名前のデータベース上のレプリケーション・コントロール表を分析する例です。

```
asnanalyze -db proddb1
```

例 2

proddb1 と proddb2 のデータベース上のレプリケーション・コントロール表について、詳細レベルの分析を入手する例です。

```
asnanalyze -db proddb1 proddb2 -la detailed
```

例 3

proddb1 と proddb2 のデータベース上の、IBMSNAP_APPLYTRAIL、IBMSNAP_APPLYTRACE、IBMSNAP_CAPTRACE、IBMSNAP_CAPMON、および IBMSNAP_SIGNAL 表から、最新の 2 日の情報を分析する例です。

```
asnanalyze -db proddb1 proddb2 -tl 2 -at 2 -ct 2 -cm 2 -sg 2
```

例 4

proddb1 および proddb2 データベース上の、IBMSNAP_APPLYTRAIL、IBMSNAP_APPLYTRACE、IBMSNAP_CAPTRACE、IBMSNAP_CAPMON、および IBMSNAP_SIGNAL 表から、qual1 および qual2 のアブライ修飾子についてのみ、最新の 2 日の情報を simple レベルで分析するには、次のように指定します。

```
asnanalyze -db proddb1 proddb2 -la simple -tl 2 -at 2 -ct 2 -cm 2 -sg 2  
-aq qual1 qual2 -od c:%mydir -fn anzout -pw c:%SQLLIB
```

このコマンド例は、アナライザーの出力を *c:%mydir* ディレクトリーの下に *anzout* ファイルに書き込み、*c:%SQLLIB* ディレクトリーからパスワード情報を使用します。

例 5

特定のキャプチャー・スキーマを分析する例です。

```
asnanalyze -db proddb1 proddb2 -cs BSN
```

例 6

コマンド・ヘルプを表示するには、次のように入力します。

```
asnanalyze
```

関連資料:

- 400 ページの 『ANZDPR: アナライザーの操作 (OS/400)』

asnapply: アプライの開始 (UNIX、Windows、z/OS)

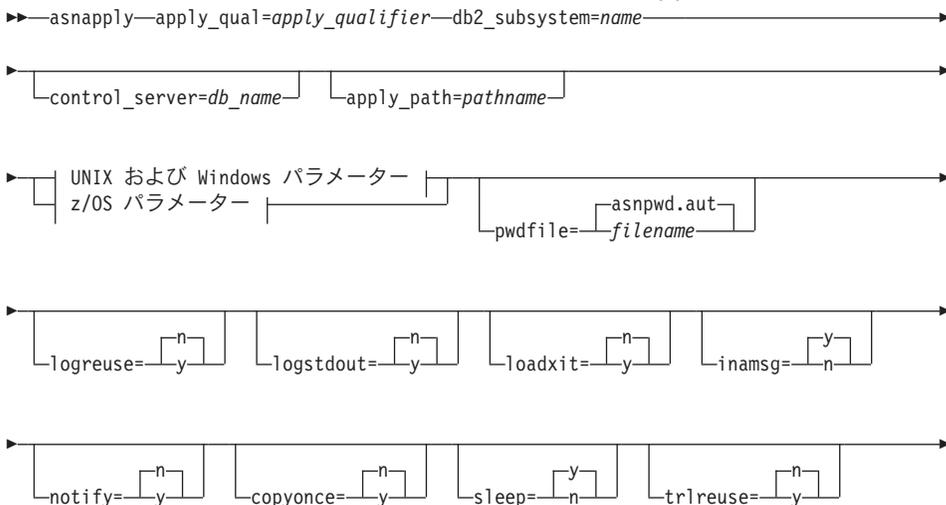
UNIX、Windows、および z/OS 上の UNIX System Services (USS) でアプライ・プログラムを始動するには、 **asnapply** コマンドを使用します。このコマンドは、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

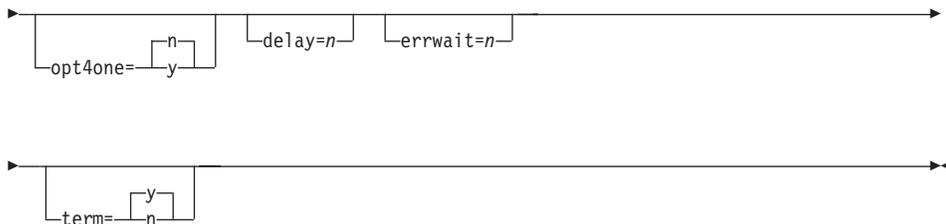
アプライ・プログラムを始動すると、次のいずれかが生じるまで連続して稼働します。

- 通常の方法で停止する。
- オペレーターが取り消す。
- 予期しないエラーまたは障害が発生する。

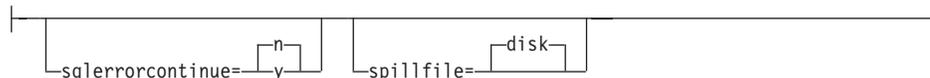
asnapply コマンドを使用してアプライ・プログラムを始動するための構文

(1)





UNIX および Windows パラメーター:



z/OS パラメーター:



注:

- 1 `db2_subsystem` パラメーターは z/OS オペレーティング・システムでのみ使用しません。

表 20 は、呼び出しパラメーターを定義します。

表 20. UNIX、Windows、および z/OS オペレーティング・システム用 *asnapply* 呼び出しパラメーター定義

| パラメーター | 定義 |
|---|---|
| <code>apply_qual=apply_qualifier</code> | <p>アプライ・プログラムが、処理されるサブスクリプション・セットの識別に使用するアプライ修飾子を指定します。このパラメーターは必須です。</p> <p>入力する値は、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の APPLY_QUAL 列の値と一致する必要があります。アプライ修飾子名には大文字小文字の区別があり、最大 18 文字です。</p> |
| <code>db2_subsystem=name</code> | <p>z/OS の場合のみ: DB2 サブシステムの名前を指定します。入力する DB2 サブシステム名は最大 4 文字です。このパラメーターにはデフォルトはありません。このパラメーターは必須です。</p> |
| <code>control_server=db_name</code> | <p>サブスクリプション定義とアプライ・プログラム・コントロール表が存在する、アプライ・コントロール・サーバーの名前を指定します。</p> <p>UNIX および Windows の場合: アプライ・コントロール・サーバーを指定しないと、このパラメーターはデフォルトで DB2DBDFT 環境変数の値になります。</p> <p>z/OS の場合: コントロール・サーバー・パラメーターは、コントロール・サーバーに接続するデータベース・サーバーの名前です。</p> |

表 20. UNIX、Windows、および z/OS オペレーティング・システム用 *asnapply* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|-----------------------------|--|
| apply_path =pathname | <p>アプライ・プログラムが使用する作業ファイルのロケーションを指定します。デフォルトは、asnapply コマンドが呼び出されたディレクトリーです。</p> |
| pwdfile =filename | <p>パスワード・ファイルの名前を指定します。パスワード・ファイルを指定しない場合、デフォルトは <code>asnpwd.aut</code> です。</p> <p>このコマンドは、apply_path パラメーターで指定されたディレクトリー内でパスワード・ファイルを探します。apply_path パラメーターを指定しないと、このコマンドは、コマンドを呼び出したディレクトリー内でパスワード・ファイルを探します。</p> |
| logreuse =y/n | <p>アプライ・プログラムが、ログ・ファイル (<code>db2instance.control_server.apply_qualifier.APP.log</code>) を再利用するか、またはメッセージを付加するかを指定します。</p> <p>n (デフォルト) アプライ・プログラムは、アプライ・プログラムの再始動後であっても、ログ・ファイルにメッセージを付加します。</p> <p>y アプライ・プログラムは、ログ・ファイルを削除し、アプライ・プログラムの再始動時にそれを再作成することにより、ログ・ファイルを再利用します。</p> <p>z/OS の場合: ログ・ファイルには DB2 インスタンス名は含まれません (<code>control_server.apply_qualifier.APP.log</code>)。</p> |
| logstdout =y/n | <p>アプライ・プログラムがメッセージをどこに送信するかを指定します。</p> <p>n (デフォルト) アプライ・プログラムはログ・ファイルにのみメッセージを送信します。</p> <p>y アプライ・プログラムは、メッセージをログ・ファイルと標準出力 (stdout) の両方に送信します。</p> |
| loadxit =y/n | <p>アプライ・プログラムが ASNLOAD を呼び出すかどうかを指定します。ASNLOAD は IBM 提供の出口ルーチンであり、エクスポートおよびロード・ユーティリティーを使用して、ターゲット表をリフレッシュします。</p> <p>n (デフォルト) アプライ・プログラムは ASNLOAD を呼び出しません。</p> <p>y アプライ・プログラムは ASNLOAD を呼び出します。</p> |

表 20. UNIX、Windows、および z/OS オペレーティング・システム用 *asnapply* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|----------------------|--|
| inamsg =y/n | <p>アプライ・プログラムを非アクティブにしたとき、アプライ・プログラムからメッセージを出すかどうかを指定します。</p> <p>y (デフォルト) アプライ・プログラムは非アクティブ時にメッセージを出します。</p> <p>n アプライ・プログラムは非アクティブ時にメッセージを出しません。</p> |
| notify =y/n | <p>アプライ・プログラムが ASNDONE を呼び出すかどうかを指定します。ASNDONE は、アプライ・プログラムがサブスクリプション・セットのコピーを終了した時に、ユーザーにコントロールを戻すための出口ルーチンです。</p> <p>n (デフォルト) アプライ・プログラムは ASNDONE を呼び出しません。</p> <p>y アプライ・プログラムは ASNDONE を呼び出します。</p> |
| copyonce =y/n | <p>アプライ・プログラムが呼びだされた時点で適格と見なされたサブスクリプション・セットごとに、アプライ・プログラムがコピー・サイクルを 1 回実行するかどうかを指定します。その後、アプライ・プログラムは終了します。適格と見なされるサブスクリプション・セットとは、以下の基準を満たすものです。</p> <ul style="list-style-type: none"> サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の中で (ACTIVATE > 0)。ACTIVATE 列の値がゼロより大きい場合、そのサブスクリプション・セットは無期限にアクティブであるか、または 1 回だけのサブスクリプション処理に使用されています。 (REFRESH_TYPE = R または B) または (REFRESH_TYPE = E であり、指定されたイベントが発生)。REFRESH_TYPE 列の値は IBMSNAP_SUBS_SET 表に保管されます。 <p>サブスクリプション・セット表の MAX_SYNCH_MINUTES 限度および、サブスクリプション・イベント (IBMSNAP_SUBS_EVENT) 表の END_OF_PERIOD タイム・スタンプが指定されている場合は、これに従います。</p> <p>n (デフォルト) アプライ・プログラムは、適格なサブスクリプション・セットごとにコピー・サイクルを 1 回実行しません。</p> <p>y アプライ・プログラムは、適格なサブスクリプション・セットごとにコピー・サイクルを 1 回実行します。</p> |

表 20. UNIX、Windows、および z/OS オペレーティング・システム用 *asnapply* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|----------------------|---|
| sleep =y/n | <p>処理の対象として適格となる新しいサブスクリプションがない場合に、アプライ・プログラムがどうするかを指定します。</p> <p>y (デフォルト) アプライ・プログラムはスリープ状態に入ります。</p> <p>n アプライ・プログラムは停止します。</p> |
| trlreuse =y/n | <p>アプライ・プログラムの始動時に、アプライ・プログラムがアプライ・トレール (IBMSNAP_APPLYTRAIL) 表を空にするかどうかを指定します。</p> <p>n (デフォルト) アプライ・プログラムは IBMSNAP_APPLYTRAIL 表にエントリを付加します。アプライ・プログラムは表を空にしません。</p> <p>y アプライ・プログラムはプログラム始動時に IBMSNAP_APPLYTRAIL 表を空にします。</p> |
| opt4one =y/n | <p>アプライ・プログラムに定義されているサブスクリプション・セットが 1 つだけの場合、アプライ・プログラムのパフォーマンスを最適化するかどうかを指定します。</p> <p>n (デフォルト) サブスクリプション・セットが 1 つの場合、アプライ・プログラムのパフォーマンスを最適化しません。</p> <p>y サブスクリプション・セットが 1 つの場合、アプライ・プログラムのパフォーマンスを最適化します。</p> <p>最適化を y に設定すると、アプライ・プログラムはサブスクリプション・セット・メンバーの情報をキャッシュに入れて再利用します。このようにサブスクリプション・セット・メンバーの情報を再利用すると、CPU 使用率が減り、スループットが向上します。</p> |
| delay =n | <p>連続レプリケーションを使用する場合に、それぞれのアプライ・サイクルが終了した後、何秒待つかを示す遅延時間 (秒単位) を指定します。 <i>n</i> は、0、1、2、3、4、5、または 6 です。デフォルトは 6 です。</p> |
| errwait =n | <p>アプライ・プログラムがエラー状態になった後、何秒待ってから再試行するかを示す秒数 (1 から 300) を指定します。デフォルト値は 300 秒 (5 分) です。</p> <p>重要: アプライ・プログラムはほとんど切れ目なく稼動しており、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表に多くの行を生成するので、ここにあまり小さい数を指定しないでください。</p> |

表 20. UNIX、Windows、および z/OS オペレーティング・システム用 *asnapply* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|------------------------------|--|
| term =y/n | <p>DB2 の状況がアプライ・プログラムの動作にどのように影響するかを指定します。</p> <p>y (デフォルト) DB2 が終了すると、アプライ・プログラムは終了します。</p> <p>n DB2 がアクティブでない場合、アプライ・プログラムは DB2 の始動を待ちます。</p> <p>UNIX および Windows の場合: DB2 が静止し、アプライ・プログラムがアクティブの場合、アプライ・プログラムはアクティブのままであり、DB2 が静止モードでなくなるまで再接続しません。</p> <p>z/OS の場合: DB2 が静止し、アプライ・プログラムがアクティブの場合、アプライ・プログラムはアクティブのままであり、DB2 が再度開始されるまで再接続しません。</p> |
| sqlerrorcontinue =y/n | <p>UNIX および Windows の場合のみ: アプライ・プログラムがある種の SQL エラーを検出した場合、アプライ・プログラムが処理を継続するかどうかを指定します。</p> <p>アプライ・プログラムは、失敗した SQLSTATE を SQLSTATE ファイルに指定された値に照らしてチェックします。この SQLSTATE ファイルは、アプライ・プログラムの実行前にユーザーが作成します。ファイルの内容と一致すれば、アプライ・プログラムは失敗した行についての情報をエラー・ファイル (<i>apply_qualifier.ERR</i>) に書き込み、処理を継続します。SQLSTATE ファイルには 5 バイト値を 20 個まで含めることができます。</p> <p>n (デフォルト) アプライ・プログラムは SQLSTATE ファイルをチェックしません。</p> <p>y アプライ・プログラムは処理中に SQLSTATE ファイルをチェックします。</p> |
| spillfile =filetype | <p>フェッチした応答セットをどこに保管するかを指定します。</p> <p>UNIX および Windows の場合、有効な値は次のとおりです。</p> <p>disk (デフォルト) ディスク・ファイル。</p> <p>z/OS の場合、有効な値は次のとおりです。</p> <p>mem (デフォルト) メモリー・ファイル。応答セット用の十分なメモリーがない場合、アプライ・プログラムは失敗します。</p> <p>disk ディスク・ファイル。</p> |

asnapply の例

次の例は、**asnapply** コマンドの使用法を示しています。

例 1

アプライ修飾子 **AQ1** を使用し、コントロール・サーバー名は **dbx** で、
/home/files/apply/ ディレクトリーにある作業ファイルを使用して、アプライ・プログラムを始動する例です。

```
asnapply apply_qual=AQ1 control_server=dbx apply_path=/home/files/apply/  
pwdfile=pass1.txt
```

アプライ・プログラムは /home/files/apply/ ディレクトリーで **pass1.txt** という名前のパスワード・ファイルを探します。

例 2

ASNLOAD 出口ルーチン呼び出すアプライ・プログラムを始動します。

```
asnapply apply_qual=AQ1 control_server=dbx pwdfile=pass1.txt loadxit=y
```

この例では、アプライ・プログラムは **pass1.txt** という名前のパスワード・ファイルを現行ディレクトリーで探します。

例 3

適格なサブスクリプション・セットごとにコピー・サイクルを 1 回実行するアプライ・プログラムを始動します。

```
asnapply apply_qual=AQ1 control_server=dbx apply_path=/home/files/apply/  
copyonce=y
```

この例では、アプライ・プログラムは /home/files/apply/ ディレクトリーで、デフォルトのパスワード・ファイル (**asnpwd.aut**) を探します。

関連作業:

- 461 ページの 『第 19 章 レプリケーション・プログラムの操作 (z/OS)』

関連資料:

- 438 ページの 『STRDPRAPY: アプライ・プログラムの始動 (OS/400)』

asnccap: キャプチャーの開始 (UNIX、Windows、z/OS)

UNIX、Windows、および z/OS 上の UNIX System Services (USS) でキャプチャー・プログラムを始動するには、**asnccap** コマンドを使用します。このコマンドは、レプリケーション・センターではなく、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

キャプチャー・プログラムを始動すると、停止されるかまたはリカバリー不能エラーが検出されるまで実行を続けます。

asncap コマンドを使用してキャプチャー・プログラムを始動するための構文

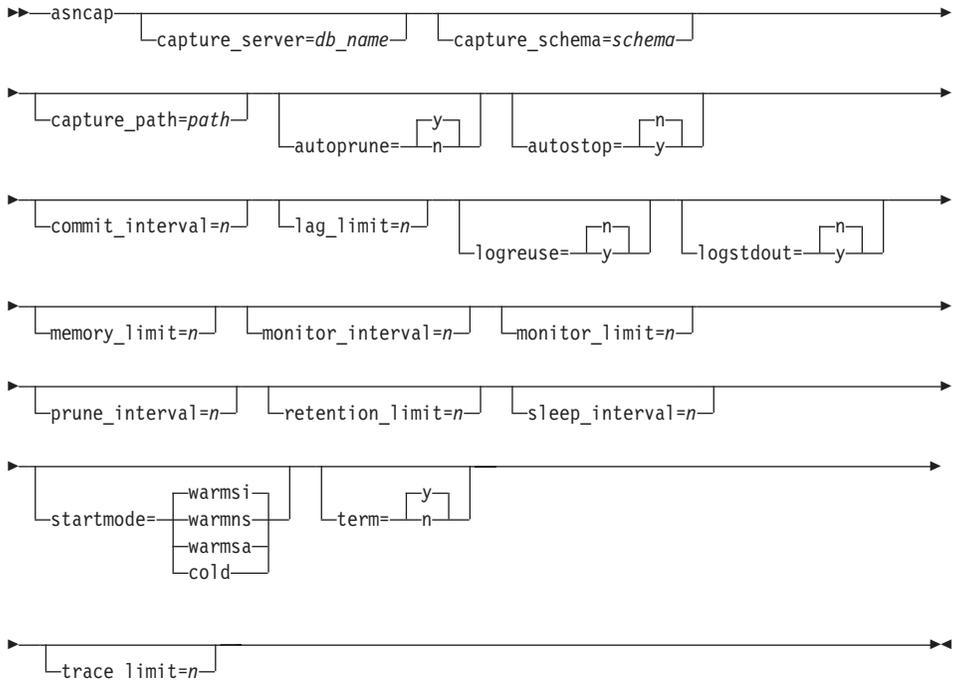


表 21 は、呼び出しパラメーターを定義します。

表 21. UNIX、Windows、および z/OS オペレーティング・システム用 asncap 呼び出しパラメーター定義

| パラメーター | 定義 |
|-------------------------------------|---|
| <code>capture_server=db_name</code> | <p>キャプチャー・コントロール・サーバーの名前を指定します。</p> <p>UNIX および Windows の場合: キャプチャー・コントロール・サーバーを指定しないと、このパラメーターはデフォルトで DB2DBDFT 環境変数の値になります。</p> <p>z/OS の場合: <code>capture_server</code> パラメーターは、コントロール・サーバーに接続するデータベース・サーバーの名前です。データ共有の場合、グループ・アタッチ名を使用しないでください。その代わりに、メンバー・サブシステム名を指定してください。</p> |
| <code>capture_schema=schema</code> | <p>特定のキャプチャー・プログラムを識別するために使用するキャプチャー・スキーマの名前を指定します。入力するスキーマ名の長さは 1 から 30 文字でなければなりません。デフォルトは ASN です。</p> |
| <code>capture_path=path</code> | <p>キャプチャー・プログラムが使用する作業ファイルのロケーションを指定します。デフォルトは、<code>asncap</code> コマンドが呼び出されたディレクトリーです。</p> |

表 21. UNIX、Windows、および z/OS オペレーティング・システム用 asncap 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|--------------------------|---|
| autoprune=y/n | <p>変更データ (CD) 表、作業単位 (UOW) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびシグナル (IBMSNAP_SIGNAL) 表の中の行の自動ブルーニングを可能にするかどうかを指定します。</p> <p>y (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表で指定されたインターバルで、適切な行の自動ブルーニングを行います。キャプチャー・プログラムは、行が複製されたものか否かに関係なく、保持制限より古い CD、UOW、および IBMSNAP_SIGNAL の行を削除します。</p> <p>n 自動ブルーニングは使用不可になります。</p> |
| autostop=y/n | <p>キャプチャー・プログラムの始動前にログに記録されたすべてのトランザクションを検索した後、キャプチャー・プログラムを終了するかどうかを指定します。</p> <p>n (デフォルト) キャプチャー・プログラムは、トランザクションを検索した後、終了しません。</p> <p>y キャプチャー・プログラムは、トランザクションを検索した後、終了します。</p> |
| commit_interval=n | <p>キャプチャー・プログラムが、何秒待ってから、作業単位 (UOW) 表および変更データ (CD) 表に行をコミットするかを示す秒数を指定します。デフォルトは 30 秒です。</p> |
| lag_limit=n | <p>キャプチャー・プログラムがログ・レコードを処理するときに、何分の遅れが出たらシャットダウンするかを示す許容分数を指定します。デフォルトは 10,080 分 (7 日) です。</p> |
| logreuse=y/n | <p>キャプチャー・プログラムがログ・ファイルを再利用するか、またはログ・ファイルにメッセージを付加するかを指定します (ログ・ファイル名は <code>db2instance.capture_server.capture_schema.CAP.log</code>)。</p> <p>n (デフォルト) キャプチャー・プログラムは、キャプチャー・プログラムの再始動後であってもログ・ファイルにメッセージを付加します。</p> <p>y キャプチャー・プログラムは、まず現行のログ・ファイルを切り捨て、次にキャプチャー・プログラムの再始動時に新しいログを開始して、ログ・ファイルを再利用します。</p> <p>z/OS の場合: ログ・ファイル名には DB2 インスタンス名は含まれません (<code>capture_server.capture_schema.CAP.log</code>)。</p> |

表 21. UNIX、Windows、および z/OS オペレーティング・システム用 asncap 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|---------------------------|---|
| logstdout=y/n | <p>キャプチャー・プログラムがメッセージをどこに送信するかを指定します。</p> <p>n (デフォルト) キャプチャー・プログラムはログ・ファイルにのみメッセージを送信します。</p> <p>y キャプチャー・プログラムは、ログ・ファイルと標準出力 (stdout) の両方にメッセージを送信します。</p> |
| memory_limit=n | <p>トランザクションを作成するためにキャプチャー・プログラムが使用できるメモリーの最大サイズ (MB 単位) を指定します。このメモリー限度に達すると、キャプチャー・プログラムはトランザクションをファイルに書き出します。デフォルトは 32 MB です。</p> |
| monitor_interval=n | <p>キャプチャー・プログラムが、キャプチャー・モニター (IBMSNAP_CAPMON) 表に行を挿入する頻度 (秒単位) を指定します。デフォルトは 300 秒 (5 分) です。</p> |
| monitor_limit=n | <p>キャプチャー (IBMSNAP_CAPMON) 表内の行は、何分たったら、ブルーニングの対象として適格になるかを示す分数を指定します。 monitor_limit パラメーターの値よりも古い、すべての IBMSNAP_CAPMON 行は、次のブルーニング・サイクルで削除されます。デフォルトは 10,080 分 (7 日) です。</p> |
| prune_interval=n | <p>変更データ (CD) 表、作業単位 (UOW) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびシグナル (IBMSNAP_SIGNAL) 表の整理の頻度 (秒数) を指定します。 autoprun パラメーターを n に設定した場合は、このパラメーターは無視されます。デフォルトは 300 秒 (5 分) です。</p> |
| retention_limit=n | <p>変更データ (CD) 表、作業単位 (UOW) 表、またはシグナル (IBMSNAP_SIGNAL) 表の行は、何分たったら、ブルーニングの対象として適格になるかを示す分数を指定します。 retention_limit パラメーターの値よりも古い行は、次のブルーニング・サイクルで削除されます。デフォルトは 10,080 分 (7 日) です。</p> |
| sleep_interval=n | <p>キャプチャー・プログラムが、アクティブ・ログの処理を終了し、バッファが空であると判断するまで、何秒スリープするかを示す秒数を指定します。デフォルトは 5 秒です。</p> <p>z/OS の場合: バッファが半分以上空いていることを戻した後、キャプチャー・プログラムが何秒スリープするかを指定します。</p> |

表 21. UNIX、Windows、および z/OS オペレーティング・システム用 asncap 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|-----------------------|--|
| startmode=mode | <p>キャプチャー始動時にキャプチャー・プログラムが使用する、処理プロシージャを指定します。</p> <p>warmsi (デフォルト)</p> <p>ウォーム・スタート情報を入手できる場合、キャプチャー・プログラムは、直前の実行の終了時点から処理を再開します。これがキャプチャー・プログラムの最初の始動である場合は、自動的にコールド・スタートに切り替えます。</p> <p>ウォーム・スタート中は、キャプチャー・プログラムは、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、変更データ (CD) 表、作業単位 (UOW) 表、および再始動 (IBMSNAP_RESTART) 表に手を付けずそのままにしておきます。キャプチャー・プログラムの始動後にエラーが起こった場合、キャプチャー・プログラムは終了します。</p> <p>warmns</p> <p>ウォーム・スタート情報を入手できる場合、キャプチャー・プログラムは、直前の実行の終了時点から処理を再開します。キャプチャー・プログラムの始動後にエラーが起こった場合、キャプチャー・プログラムは終了します。キャプチャー・プログラムは、ウォーム・スタートできない場合、コールド・スタートに切り替わりません。</p> <p>warmsa</p> <p>ウォーム・スタート情報を入手できる場合、キャプチャー・プログラムは、直前の実行の終了時点から処理を再開します。キャプチャー・プログラムは、ウォーム・スタートできない場合、コールド・スタートに切り替わります。</p> <p>cold</p> <p>キャプチャー・プログラムを始動すると、初期化中に CD、UOW、および IBMSNAP_CAPTRACE 表内のすべての列を削除します。レプリケーション・ソースに対するすべてのサブスクリプションは、次のアプライ処理のサイクルですべてリフレッシュされます。ターゲットが非コンプリート整合変更データ (CCD) 表の場合、フル・リフレッシュは行われません。</p> |

表 21. UNIX、Windows、および z/OS オペレーティング・システム用 asncap 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|----------------------|--|
| term=y/n | <p>DB2 が終了した場合にキャプチャー・プログラムは終了するかどうかを指定します。</p> <p>y (デフォルト) DB2 が終了した場合、キャプチャー・プログラムは終了します。</p> <p>n DB2 が MODE(QUIESCE) で終了した場合、キャプチャー・プログラムは実行を継続します。DB2 が始動されると、キャプチャー・プログラムは warm モードで始動し、DB2 が終了した時点からのキャプチャーを開始します。</p> <p>DB2 が FORCE または異常終了で終了した場合は、このパラメーターを n にしてもキャプチャー・プログラムは終了します。</p> <p>このパラメーターを n にし、制限付きアクセス (ACCESS MAINT) を使用して DB2 を始動すると、キャプチャー・プログラムは接続できないので、結果として終了します。</p> |
| trace_limit=n | <p>キャプチャー・トレース (IBMSNAP_CAPTRACE) 表の行は、何分たったら、ブルーニングの対象として適格になるかを示す分数を指定します。</p> <p>trace_limit パラメーターの値よりも古い、すべての IBMSNAP_CAPTRACE 行が、次のブルーニング・サイクルで削除されます。デフォルトは 10,080 分 (7 日) です。</p> |

asncap の例

次の例は、**asncap** コマンドの使用法を示しています。

例 1

db という名前のキャプチャー・コントロール・サーバーおよび、キャプチャー・スキーマ ASN を使用し、/home/files/capture/logs/ ディレクトリーにある作業ファイルを使用して、初めてキャプチャー・プログラムを始動する例です。

```
asncap capture_server=db capture_schema=ASN
      capture_path=/home/files/capture/logs/ startmode=cold
```

例 2

キャプチャー・プログラムが停止した後、ブルーニングを行わずにキャプチャー・プログラムを再始動する例です。

```
asncap capture_server=db autoprune=n sleep_interval=10 startmode=warmsa
```

この例では、キャプチャー・プログラムは、対応するコントロール表内のすべての行を保存し、アクティブ・ログの処理を終了し、バッファーが空であると判断した後、10 秒スリープします。ウォーム・スタート情報を入手できない場合、キャプチャー・プログラムは、直前の実行の終了時点から処理を再開し、コールド・スタートに切り替えます。

例 3

warmns startmode で、変更されたパラメーター設定値を使用して、キャプチャー・プログラムを再始動します。

```
asncap capture_server=db autoprune=y prune_interval=60 retention_limit=1440
startmode=warmns
```

このコマンドは、キャプチャー・プログラムを再始動し、新しいパラメーター設定値を使用するようにします。新しいパラメーター設定値は、CD、UOW、および IBMSNAP_SIGNAL 表がプルーニングの対象として適格になるまでの時間を減らし、プルーニングの頻度をデフォルトのパラメーター設定値よりも増やしています。ウォーム・スタート情報入手できない場合、キャプチャー・プログラムは、直前の実行の終了時点から処理を再開しますが、自動的にコールド・スタートに切り替えることはしません。

例 4

その作業ファイルをすべて、capture_files という新しいサブディレクトリーに送信するキャプチャー・プログラムを始動します。

1. 該当のディレクトリーに行き、capture_files という新しいサブディレクトリーを作成します。

```
cd /home/db2inst
mkdir capture_files
```

2. キャプチャー・プログラムを始動し、たった今作成した新しいサブディレクトリーにあるキャプチャー・パスを指定します。

```
asncap capture_server=db capture_schema=ASN
capture_path=/home/db2inst/capture_files startmode=warmnsi
```

関連作業:

- 461 ページの 『第 19 章 レプリケーション・プログラムの操作 (z/OS)』

関連資料:

- 446 ページの 『STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)』

asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)

UNIX、Windows、および z/OS 上の UNIX System Services (USS) でキャプチャー・プログラムを操作するには、**asnccmd** コマンドを使用します。このコマンドは、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

asnccmd コマンドを使用してキャプチャー・プログラムを操作するための構文

```
▶▶ asnccmd [capture_server=db_name] [capture_schema=schema]
```



パラメーター:

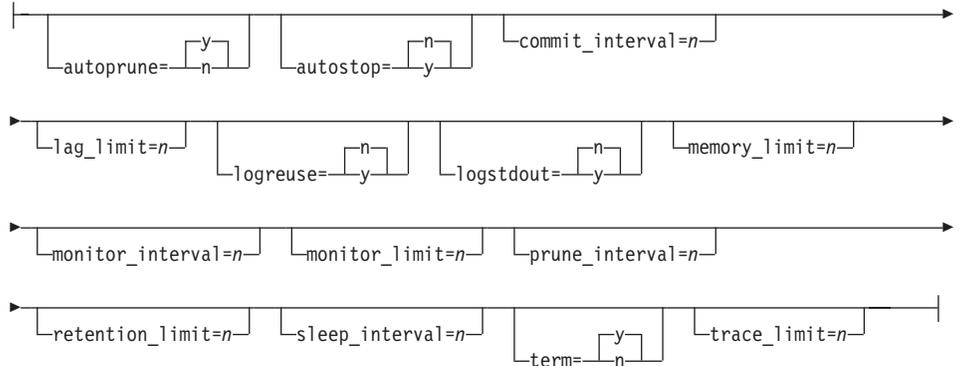


表 22 は、呼び出しパラメーターを定義します。

表 22. UNIX、Windows、および z/OS オペレーティング・システム用 *asnccmd* 呼び出しパラメーター定義

| パラメーター | 定義 |
|-------------------------------------|---|
| <code>capture_server=db_name</code> | <p>キャプチャー・コントロール・サーバーの名前を指定します。</p> <p>UNIX および Windows の場合: キャプチャー・コントロール・サーバーを指定しないと、このパラメーターはデフォルトで DB2DBDFT 環境変数の値になります。</p> <p>z/OS の場合: コントロール・サーバーに接続するデータベース・サーバーの名前です。データ共有の場合、グループ・アタッチ名を使用しないでください。その代わりに、メンバー・サブシステム名を指定してください。</p> |
| <code>capture_schema=schema</code> | <p>特定のキャプチャー・プログラムを識別するために使用するキャプチャー・スキーマの名前を指定します。入力するスキーマ名の長さは 1 から 30 文字でなければなりません。デフォルトは ASN です。</p> |
| <code>chgparms</code> | <p>キャプチャー・プログラムの実行中にそのパラメーター値を変更することを指定します。新しいパラメーター値を指定するかまたは、キャプチャー・プログラムの始動時に渡された値をオーバーライドすることができます。どのパラメーターをオーバーライドできるかについては、343 ページの表 23 を参照してください。</p> |

表 22. UNIX、Windows、および z/OS オペレーティング・システム用 `asnccmd` 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|-----------------|---|
| prune | 変更データ (CD) 表、作業単位 (UOW) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびシグナル (IBMSNAP_SIGNAL) 表の整理を一度行う場合にこのパラメーターを指定します。このコマンドが正常にキューに入ると、キャプチャー・プログラムはメッセージを出します。 |
| qryparms | 現行の稼働パラメーター値を標準出力 (stdout) に書き込む場合に指定します。 |
| reinit | キャプチャー・プログラムに、新しく追加されたレプリケーション・ソースを登録 (IBMSNAP_REGISTER) 表から、または新しいチューニング・パラメーターをキャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表から、入手させる場合に指定します。たとえば、新しいレプリケーション・ソースを追加した場合、またはキャプチャー・プログラムの実行中に、ALTER ADD を使用してレプリケーション・ソースと変更データ (CD) 表に列を追加した場合に、このパラメーターを使用します。 重要: キャプチャー・プログラムの実行中に、レプリケーション・ソースの取り消しまたはレプリケーション・ソース表のドロップを行った後で、キャプチャー・プログラムを再初期化するために、reinit パラメーターを使用しないでください。この場合は、キャプチャー・プログラムを停止してから、startmode パラメーターに warmsa、warmns、または warmsi を指定した asnccap コマンドを使用して、再始動してください。 |
| resume | 中断したキャプチャー・プログラムにデータのキャプチャーを再開させることを指定します。 |
| status | それぞれのキャプチャー・スレッド (管理、ブルーニング、シリアライゼーション、およびワーカー) の状態を示すメッセージを受け取ることを指定します。 |
| stop | キャプチャー・プログラムを通常どおりに停止して、その時点までに処理したログ・レコードをコミットすることを指定します。 |
| suspend | ピーク時に、キャプチャー・プログラムの環境を損なわずにすむ範囲内で、運用トランザクションにオペレーティング・システムのリソースを解放することを指定します。 重要: レプリケーション・ソースを取り消すためにキャプチャーを中断しないでください。中断するのではなく、キャプチャー・プログラムを停止してください。 |

表 23 は、chgpargs 呼び出しパラメーターを定義しています。

表 23. UNIX、Windows、および z/OS オペレーティング・システム用 *asnccmd chgpargs* パラメーター定義

| パラメーター | 定義 |
|-----------------------------------|---|
| autoprun = <i>y/n</i> | <p>変更データ (CD) 表、作業単位 (UOW) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびシグナル (IBMSNAP_SIGNAL) 表のこの行の自動ブルーニングを可能にするかどうかを指定します。</p> <p>y (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表で指定されたインターバルで、適切な行の自動ブルーニングを行います。キャプチャー・プログラムは、行が複製されたものか否かに関係なく、保持制限より古い CD、UOW、および IBMSNAP_SIGNAL の行を削除します。</p> <p>n 自動ブルーニングは使用不可になります。</p> |
| autostop = <i>y/n</i> | <p>キャプチャー・プログラムの始動前にログに記録されたすべてのトランザクションを検索した後、キャプチャー・プログラムを終了させるかどうかを指定します。</p> <p>n (デフォルト) キャプチャー・プログラムは、トランザクションを検索した後、終了しません。</p> <p>y キャプチャー・プログラムは、トランザクションを検索した後、終了します。</p> |
| commit_interval = <i>n</i> | <p>キャプチャー・プログラムが、何秒待ってから、作業単位 (UOW) 表および変更データ (CD) 表に行をコミットするかを示す秒数を指定します。デフォルトは 30 秒です。</p> |
| lag_limit = <i>n</i> | <p>キャプチャー・プログラムがログ・レコードを処理するときに、何分の遅れが出たらシャットダウンするかを示す許容分数を指定します。デフォルトは 10,080 分 (7 日) です。</p> |

表 23. UNIX、Windows、および z/OS オペレーティング・システム用 `asnccmd chgparms` パラメーター定義 (続き)

| パラメーター | 定義 |
|---------------------------------|--|
| <code>logreuse=y/n</code> | <p>キャプチャー・プログラムがログ・ファイルを再利用するか、またはログ・ファイルにメッセージを付加するかを指定します (ログ・ファイル名は <code>db2instance.capture_server.capture_schema.CAP.log</code>)。</p> <p>n (デフォルト) キャプチャー・プログラムは、キャプチャー・プログラムの再始動後であってもログ・ファイルにメッセージを付加します。</p> <p>y キャプチャー・プログラムは、まず現行のログ・ファイルを切り捨て、次にキャプチャー・プログラムの再始動時に新しいログを開始して、ログ・ファイルを再利用します。</p> <p>chgparms パラメーターを使用してこのパラメーターを <code>y</code> に変更すると、ログは即時に切り捨てられ、再利用されます。この場合の logreuse パラメーターの変更は、キャプチャー・プログラムの次の始動には影響しません。</p> <p>z/OS の場合: ログ・ファイル名には DB2 インスタンス名は含まれません (<code>capture_server.capture_schema.CAP.log</code>)。</p> |
| <code>logstdout=y/n</code> | <p>キャプチャー・プログラムがメッセージをどこに送信するかを指定します。</p> <p>n (デフォルト) キャプチャー・プログラムはログ・ファイルにのみメッセージを送信します。</p> <p>y キャプチャー・プログラムは、ログ・ファイルと標準出力 (<code>stdout</code>) の両方にメッセージを送信します。</p> |
| <code>memory_limit=n</code> | <p>トランザクションを作成するためにキャプチャー・プログラムが利用できるメモリーの最大サイズ (MB 単位) を指定します。このメモリー限度に達すると、キャプチャー・プログラムはトランザクションをファイルに書き出します。デフォルトは 32 MB です。</p> |
| <code>monitor_interval=n</code> | <p>キャプチャー・プログラムが、キャプチャー・モニター (IBMSNAP_CAPMON) 表に行を挿入する頻度 (秒単位) を指定します。デフォルトは 300 秒 (5 分) です。</p> |
| <code>monitor_limit=n</code> | <p>キャプチャー (IBMSNAP_CAPMON) 表内の行は、何分たったら、ブルーニングの対象として適格になるかを示す分数を指定します。 monitor_limit パラメーターの値よりも古い、すべての IBMSNAP_CAPMON 行は、次のブルーニング・サイクルで削除されます。デフォルトは 10,080 分 (7 日) です。</p> |
| <code>prune_interval=n</code> | <p>変更データ (CD) 表、作業単位 (UOW) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびシグナル (IBMSNAP_SIGNAL) 表の整理の頻度 (秒数) を指定します。 <code>autoprun</code> パラメーターを <code>n</code> に設定した場合は、このパラメーターは無視されます。デフォルトは 300 秒 (5 分) です。</p> |

表 23. UNIX、Windows、および z/OS オペレーティング・システム用 *asnccmd chgparms* パラメーター定義 (続き)

| パラメーター | 定義 |
|-----------------------------------|---|
| retention_limit = <i>n</i> | 変更データ (CD) 表、作業単位 (UOW) 表、またはシグナル (IBMSNAP_SIGNAL) 表の行は、何分たったら、ブルーニングの対象として適格になるかを示す分数を指定します。 retention_limit パラメーターの値よりも古い行は、次のブルーニング・サイクルで削除されます。デフォルトは 10,080 分 (7 日) です。 |
| sleep_interval = <i>n</i> | キャプチャー・プログラムが、アクティブ・ログの処理を終了し、バッファーが空であると判断するまで、何秒スリープするかを示す秒数を指定します。デフォルトは 5 秒です。 z/OS の場合: バッファーが半分以上空いていることを戻した後、キャプチャー・プログラムが何秒スリープするかを指定します。 |
| term = <i>y/n</i> | DB2 が終了した場合にキャプチャー・プログラムは終了するかどうかを指定します。 y (デフォルト) DB2 が終了した場合、キャプチャー・プログラムは終了します。 n DB2 が MODE(QUIESCE) で終了した場合、キャプチャー・プログラムは実行を継続します。 DB2 が始動されると、キャプチャー・プログラムは warm モードで始動し、 DB2 が終了した時点からのキャプチャーを開始します。 DB2 が FORCE または異常終了で終了した場合は、このパラメーターを n にしてもキャプチャー・プログラムは終了します。 このパラメーターを n にし、制限付きアクセス (ACCESS MAINT) を使用して DB2 を始動すると、キャプチャー・プログラムは接続できないので、結果として終了します。 |
| trace_limit = <i>n</i> | キャプチャー・トレース (IBMSNAP_CAPTRACE) 表の行は、何分たったら、ブルーニングの対象として適格になるかを示す分数を指定します。 trace_limit パラメーターの値よりも古い、すべての IBMSNAP_CAPTRACE 行が、次のブルーニング・サイクルで削除されません。デフォルトは 10,080 分 (7 日) です。 |

asnccmd の例

次の例は、**asnccmd** コマンドの使用法を示しています。

例 1

新しく追加したレプリケーション・ソースを、実行中のキャプチャー・プログラムに認識させるには、次のように入力します。

```
asnccmd capture_server=db capture_schema=ASN reinit
```

例 2

CD、UOW、IBMSNAP_CAPMON、IBMSNAP_CAPTRACE、および IBMSNAP_SIGNAL 表の整理を 1 回行う例です。

asnccmd

```
asnccmd capture_server=db capture_schema=ASN prune
```

例 3

それぞれのキャプチャー・スレッドの状態についてメッセージを受け取る場合の例です。

```
asnccmd capture_server=db capture_schema=ASN status
```

例 4

キャプチャー・プログラムの現行操作値を標準出力に送信する例です。

```
asnccmd capture_server=db capture_schema=ASN qryparms
```

例 5

キャプチャー・プログラムの実行中に自動プルーニングを行わないようにするには、次のように入力します。

```
asnccmd capture_server=db capture_schema=ASN chgparms autoprun=n
```

例 6

キャプチャー・プログラムの実行を停止するには、次のように入力します。

```
asnccmd capture_server=db capture_schema=ASN stop
```

関連作業:

- 461 ページの 『第 19 章 レプリケーション・プログラムの操作 (z/OS)』

関連資料:

- 426 ページの 『OVRDPRCAPA: DPR キャプチャー属性のオーバーライド (OS/400)』

asnmcmd: レプリケーション・アラート・モニターの操作 (UNIX、Windows、z/OS)

UNIX、Windows、および z/OS 上の UNIX System Services (USS) でレプリケーション・アラート・モニターを操作するには、**asnmcmd** コマンドを使用します。このコマンドは、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

asnmcmd コマンドを使用してレプリケーション・アラート・モニターを操作するための構文

```
▶ asnmcmd [monitor_server=server] monitor_qual=monitor_qualifier [reinit | status | stop] ▶
```

表 24 は、呼び出しパラメーターを定義します。

表 24. UNIX、Windows、および z/OS オペレーティング・システム用 *asnmcmd* 呼び出しパラメーター定義

| パラメーター | 定義 |
|---------------------------------------|---|
| monitor_server=server | レプリケーション・アラート・モニター・プログラムを実行し、モニター・コントロール表が存在する、モニター・コントロール・サーバーの名前を指定します。これを入力する場合は、最初のパラメーターでなければなりません。 UNIX および Windows の場合: モニター・コントロール・サーバーを指定しないと、このパラメーターはデフォルトで DB2DBDFT 環境変数の値になります。 z/OS の場合: デフォルトは DSN です。 |
| monitor_qual=monitor_qualifier | レプリケーション・アラート・モニター・プログラムが使用するモニター修飾子を指定します。モニター修飾子は、モニター対象のサーバーおよび関連するモニター条件を識別します。 モニター修飾子は指定しなければなりません。モニター修飾子名には大文字小文字の区別があり、最大 18 文字です。 |
| reinit | レプリケーション・アラート・モニター・プログラムが新しいパラメーターをモニター・コントロール表から取得するように指定します。 |
| status | レプリケーション・アラート・モニター内の各スレッド (管理、シリアライゼーション、およびワーカー) の状態を示すメッセージを受け取ることを指定します。 |
| stop | レプリケーション・アラート・モニターを通常の方法で停止することを指定します。 |

asnmcmd の例

次の例は、**asnmcmd** コマンドの使用法を示します。

例 1

指定したモニター修飾子について、レプリケーション・アラート・モニターを停止します。

```
asnmcmd monitor_server=wsdb monitor_qual=monqual stop
```

例 2

レプリケーション・アラート・モニターのスレッドの状態を示すメッセージを受け取ります。

```
asnmcmd monitor_server=wsdb monitor_qual=monqual status
```

例 3

モニター・コントロール表からの最新の値を使用して、レプリケーション・アラート・モニターをリフレッシュします。

```
asnmcmd monitor_server=wsdb monitor_qual=monqual reinit
```

関連作業:

- 177 ページの 『第 11 章 レプリケーションのモニター』

関連資料:

- 348 ページの 『asnmmon: レプリケーション・アラート・モニターの開始 (UNIX、Windows、z/OS)』

asnmmon: レプリケーション・アラート・モニターの開始 (UNIX、Windows、z/OS)

UNIX、Windows、および z/OS 上の UNIX System Services (USS) でレプリケーション・アラート・モニターを操作するには、 **asnmmon** コマンドを使用します。このコマンドは、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

レプリケーション・アラート・モニターは次の情報を記録します。

- キャプチャー・プログラムとアプライ・プログラムの状況
- コントロール表に書き込まれたキャプチャーおよびアプライのエラー・メッセージ
- しきい値

asnmmon コマンドを使用してレプリケーション・アラート・モニターを開始するための構文

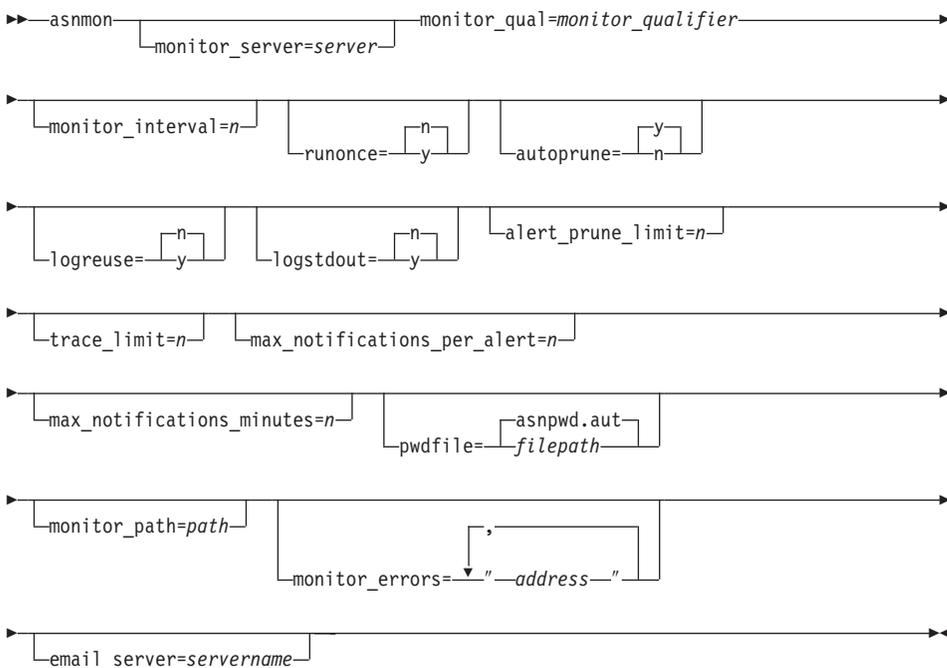


表 25 は、呼び出しパラメーターを定義します。

表 25. UNIX、Windows、および z/OS オペレーティング・システム用 *asnmon* 呼び出しパラメーター定義

| パラメーター | 定義 |
|---------------------------------------|--|
| monitor_server=server | <p>レプリケーション・アラート・モニター・プログラムを実行し、モニター・コントロール表が存在する、モニター・コントロール・サーバーの名前を指定します。これを入力する場合は、最初のパラメーターでなければなりません。</p> <p>UNIX および Windows の場合: モニター・コントロール・サーバーを指定しないと、このパラメーターはデフォルトで DB2DBDFT 環境変数の値になります。</p> <p>z/OS の場合: デフォルトは DSN です。</p> |
| monitor_qual=monitor_qualifier | <p>レプリケーション・アラート・モニター・プログラムが使用するモニター修飾子を指定します。モニター修飾子は、モニター対象のサーバーおよび関連するモニター条件を識別します。</p> <p>モニター修飾子は指定しなければなりません。モニター修飾子名には大文字小文字の区別があり、最大 18 文字です。</p> |
| monitor_interval=n | <p>このモニター修飾子について、どのような頻度 (秒数) で、レプリケーション・アラート・モニター・プログラムを実行するかを指定します。デフォルトは 300 秒 (5 分) です。</p> <p>runonce パラメーターを y に設定すると、レプリケーション・アラート・モニターはこのパラメーターを無視します。</p> <p>重要: この monitor_interval パラメーターは、レプリケーション・アラート・モニター・プログラムにのみ影響を与えます。このパラメーターはキャプチャー・プログラムには影響しません。</p> |
| runonce=y/n | <p>このモニター修飾子について、レプリケーション・アラート・モニター・プログラムを 1 回だけ実行するかどうかを指定します。</p> <p>n (デフォルト)</p> <p>レプリケーション・アラート・モニター・プログラムは、monitor_interval パラメーターに指定された頻度で実行されます。</p> <p>y</p> <p>レプリケーション・アラート・モニター・プログラムはモニター・サイクルを 1 回だけ実行します。</p> <p>runonce パラメーターを y に設定すると、レプリケーション・アラート・モニターは monitor_interval パラメーターを無視します。</p> |

表 25. UNIX、Windows、および z/OS オペレーティング・システム用 *asnmon* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|--|---|
| autoprun _{e=y/n} | レプリケーション・アラート・モニターのアラート表 (IBMSNAP_ALERTS) 内の行の自動ブルーニングを使用可能にするかどうかを指定します。 y (デフォルト) レプリケーション・アラート・モニター・プログラムは、 alert_prune_limit パラメーターの値よりも古い、IBMSNAP_ALERTS 表内の行を自動的に除去します。 n 自動ブルーニングは使用不可になります。 |
| logreuse _{=y/n} | レプリケーション・アラート・モニター・プログラムが、ログ・ファイル (<i>db2instance.monitor_server.monitor_qualifier.MON.log</i>) を再利用するか、またはメッセージを付加するかを指定します。 n (デフォルト) レプリケーション・アラート・モニター・プログラムはログ・ファイルにメッセージを付加します。 y レプリケーション・アラート・モニター・プログラムは、ログ・ファイルを削除し、レプリケーション・アラート・モニター・プログラムの再始動時にそれを再作成することにより、ログ・ファイルを再利用します。 |
| logstdout _{=y/n} | レプリケーション・アラート・モニター・プログラムがメッセージをどこに送信するかを指定します。 n (デフォルト) レプリケーション・アラート・モニター・プログラムはログ・ファイルにのみメッセージを送信します。 y レプリケーション・アラート・モニター・プログラムは、メッセージをログ・ファイルと標準出力 (stdout) の両方に送信します。 |
| alert_prune_limit _{=n} | レプリケーション・アラート・モニターのアラート表 (IBMSNAP_ALERTS) に、行を保存しておく期間 (分) を指定します。この値よりも古い行はすべて除去されます。デフォルトは 10,080 分 (7 日) です。 |
| trace_limit _{=n} | レプリケーション・アラート・モニターのトレース表 (IBMSNAP_MONTRACE) 表内に何分留まれば、その行がブルーニングの対象として適格になるかを示す分数を指定します。 trace_limit パラメーターの値よりも古い、すべての IBMSNAP_MONTRACE 行が、次のブルーニング・サイクルで削除されます。デフォルトは 10,080 分 (7 日) です。 |
| max_notifications_per_alert _{=n} | max_notifications_minutes パラメーター値で指定された期間中にアラートが起こった場合、同じアラートをユーザーに送信する最大回数を指定します。このパラメーターは、同じアラートをユーザーに何回も再送するのを避けるために使用します。デフォルトは 3 です。 |
| max_notifications_minutes _{=n} | このパラメーターは max_notifications_per_alert パラメーターと一緒に働き、アラート条件が起こる期間を示します。デフォルトは 60 分です。 |

表 25. UNIX、Windows、および z/OS オペレーティング・システム用 *asnmon* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|--------------------------------------|---|
| <code>pwdfile=filepath</code> | パスワード・ファイルの完全修飾名を指定します。このファイルは asnpwd コマンドを使用して定義します。デフォルトのファイル名は、 <code>asnpwd.aut</code> です。 |
| <code>monitor_path=path</code> | レプリケーション・アラート・モニター・プログラムが使用するログ・ファイルのロケーションを指定します。デフォルトは、 asnmon コマンドが呼び出されたディレクトリーです。 |
| <code>monitor_errors=address</code> | アラート・モニターがモニター・コントロール・サーバーに接続する前に致命的エラーが検出された場合、その通知を送信する先の E メール・アドレスを指定します。このパラメーターを使用して、無効な開始パラメーター、誤りのモニター修飾子、ダウンしているデータベース、またはその他のエラーにより、モニター・コントロール・サーバー接続が失敗したという通知を送信します。 E メール・アドレスのテキストは二重引用符で囲みます。 複数の E メール・アドレスを入力することができます。E メール・アドレスをコンマで区切ってください。コンマの前後にはスペースを入力できます。 |
| <code>email_server=servername</code> | Eメールのサーバー・アドレスを指定します。このパラメーターは、SMTP (Simple Mail Transfer Protocol) を指定して ASNMAIL 出口ルーチンを使用する場合にのみ入力します。 |

asnmon の例

次の例は、**asnmon** コマンドの使用法を示しています。

例 1

デフォルトのパラメーターを使用して、レプリケーション・アラート・モニターを開始します。

```
asnmon monitor_server=wsdb monitor_qual=monqual
```

例 2

指定したモニター修飾子について、120 秒 (2 分) おきに実行するレプリケーション・アラート・モニターを開始します。

```
asnmon monitor_server=wsdb monitor_qual=monqual monitor_interval=120
```

例 3

レプリケーション・アラート・モニターを開始し、指定したモニター修飾子について、1 回だけ実行することを指定します。

```
asnmon monitor_server=wsdb monitor_qual=monqual runonce=y
```


asnpwd の例

次の例は、**asnpwd** コマンドの使用法を示しています。

例 1

現行ディレクトリーに、デフォルト名 `asnpwd.aut` を使用してパスワード・ファイルを作成します。

```
asnpwd INIT
```

例 2

`c:¥myfiles` ディレクトリーに `pass1.aut` という名前のパスワード・ファイルを作成します。

```
asnpwd INIT Using c:¥myfiles¥pass1.aut
```

例 3

`oneuser` というユーザー ID とそのパスワードを、`c:¥myfiles` ディレクトリー内の `pass1.aut` という名前のパスワード・ファイルに追加し、このユーザー ID に `db1` データベースへのアクセス権を付与します。

```
asnpwd ADD ALIAS db1 ID oneuser PASSWORD mypwd using c:¥myfiles¥pass1.aut
```

例 4

`c:¥myfiles` ディレクトリー内の `pass1.aut` という名前のパスワード・ファイル内のエントリーのユーザー ID またはパスワードを変更します。

```
asnpwd MODIFY Alias sample ID chglocalid PASSWORD chgmajorpwd  
USING c:¥myfiles¥pass1.aut
```

例 5

`c:¥myfiles` ディレクトリー内の `pass1.aut` という名前のパスワード・ファイルから、`sample` というデータベース別名を削除します。

```
asnpwd delete alias sample USING c:¥myfiles¥pass1.aut
```

例 6

コマンド・ヘルプを表示します。

```
asnpwd
```

関連資料:

- 415 ページの 『GRTDPRAUT: ユーザーの許可 (OS/400)』

表 27. Windows オペレーティング・システム用の *asnscri* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|-------------------------|--|
| <i>asnapply_command</i> | <p>アプライ・プログラムを始動するための完全な asnapply コマンドを指定します。文書化されている asnapply コマンド構文と該当の asnapply パラメーターを使用します。</p> <p>重要: DB2PATH 環境変数が定義されていない場合、asnapply コマンドに apply_path パラメーターを含めて、作業ファイルのロケーションを指定する必要があります。DB2PATH 変数が定義されていて、apply_path を指定すると、apply_path パラメーターは DB2PATH 変数をオーバーライドします。</p> <p>asnscri コマンドは、ユーザーが入力した asnapply パラメーターの構文の妥当性を検査しません。</p> |
| <i>asnmon_command</i> | <p>レプリケーション・アラート・モニター・プログラムを始動するための完全な asnmon コマンドを指定します。文書化されている asnmon コマンド構文と該当の asnmon パラメーターを使用します。</p> <p>重要: DB2PATH 環境変数が定義されていない場合、asnmon コマンドに monitor_path パラメーターを含めて、ログ・ファイルのロケーションを指定する必要があります。DB2PATH 変数が定義されていて、monitor_path を指定すると、monitor_path パラメーターは DB2PATH 変数をオーバーライドします。</p> <p>asnscri コマンドは、ユーザーが入力した asnmon パラメーターの構文の妥当性を検査しません。</p> |

asnscri の例

次の例は、**asnscri** コマンドの使用法を示しています。

例 1

inst1 という名前の DB2 インスタンスの下でキャプチャー・プログラムを呼び出す DB2 レプリケーション・サービスを作成するには、次のように指定します。

```
asnscri -C inst1 .%joesmith password asncap capture_server=sampled
capture_schema=ASN capture_path=X:%logfiles
```

例 2

inst2 という名前の DB2 インスタンスの下で、ログオン・アカウント .%joesmith およびパスワード my\$pwd を使用して、アプライ・プログラムを呼び出す DB2 レプリケーション・サービスを作成するには、次のように指定します。

```
asnscri -A inst2 .%joesmith my$pwd asnapply control_server=db2 apply_qual=aq2
apply_path=X:%sqllib
```

例 3

inst3 という名前の DB2 インスタンスの下でレプリケーション・アラート・モニター・プログラムを呼び出す DB2 レプリケーション・サービスを作成するには、次のように指定します。

```
asnsrct -M inst3 .%joesmith password asnmon monitor_server=db3 monitor_qual=mq3
monitor_path=X:%logfiles
```

例 4

inst4 という名前の DB2 インスタンスの下でキャプチャー・プログラムを呼び出す DB2 レプリケーション・サービスを作成し、デフォルトの作業ファイル・ディレクトリを完全修飾された **capture_path** でオーバーライドするには、次のように指定します。

```
asnsrct -C inst4 .%joesmith password X:%sqllib%bin%asncap capture_server=scdb
capture_schema=ASN capture_path=X:%logfiles
```

関連作業:

- 467 ページの 『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』

関連資料:

- 357 ページの 『asnsdrop: DB2 レプリケーション・サービスのドロップ (Windows のみ)』
- 328 ページの 『asnapply: アプライの開始 (UNIX、Windows、z/OS)』
- 334 ページの 『asncap: キャプチャーの開始 (UNIX、Windows、z/OS)』
- 348 ページの 『asnmon: レプリケーション・アラート・モニターの開始 (UNIX、Windows、z/OS)』

asnsdrop: DB2 レプリケーション・サービスのドロップ (Windows のみ)

Windows NT または Windows 2000 オペレーティング・システムの Windows Service Control Manager (SCM) から、DB2 レプリケーション・サービスをドロップするには、**asnsdrop** コマンドを使用します。(DB2 レプリケーション・サービスの作成には **asnsrct** コマンドを使用します。)

asnsdrop コマンドを使用して **DB2** レプリケーション・サービスをドロップするための構文

```
▶▶—asnsdrop—service_name—————▶▶
```

表 28 は、呼び出しパラメーターを定義します。

表 28. Windows オペレーティング・システム用の *asnsdrop* 呼び出しパラメーター定義

| パラメーター | 定義 |
|---------------------|---|
| <i>service_name</i> | DB2 レプリケーション・サービスの完全修飾名を指定します。DB2 レプリケーション・サービス名を知るには、Windows SCM に入ります。Windows 2000 オペレーティング・システムでは、DB2 レプリケーション・サービスの「プロパティ」ウィンドウをオープンするとサービス名がわかります。 DB2 レプリケーション・サービス名にスペースが含まれる場合は、サービス名全体を二重引用符で囲んでください。 |

asnsdrop の例

次の例は、**asnsdrop** コマンドの使用法を示しています。

例 1

DB2 レプリケーション・サービスをドロップする例です。

```
asnsdrop DB2.SAMPLEDB.SAMPLEDB.CAP.ASN
```

例 2

A S N というスキーマ名の DB2 レプリケーション・サービスをドロップする例です。

```
asnsdrop "DB2.SAMPLEDB.SAMPLEDB.CAP.A S N"
```

関連作業:

- 467 ページの『第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)』

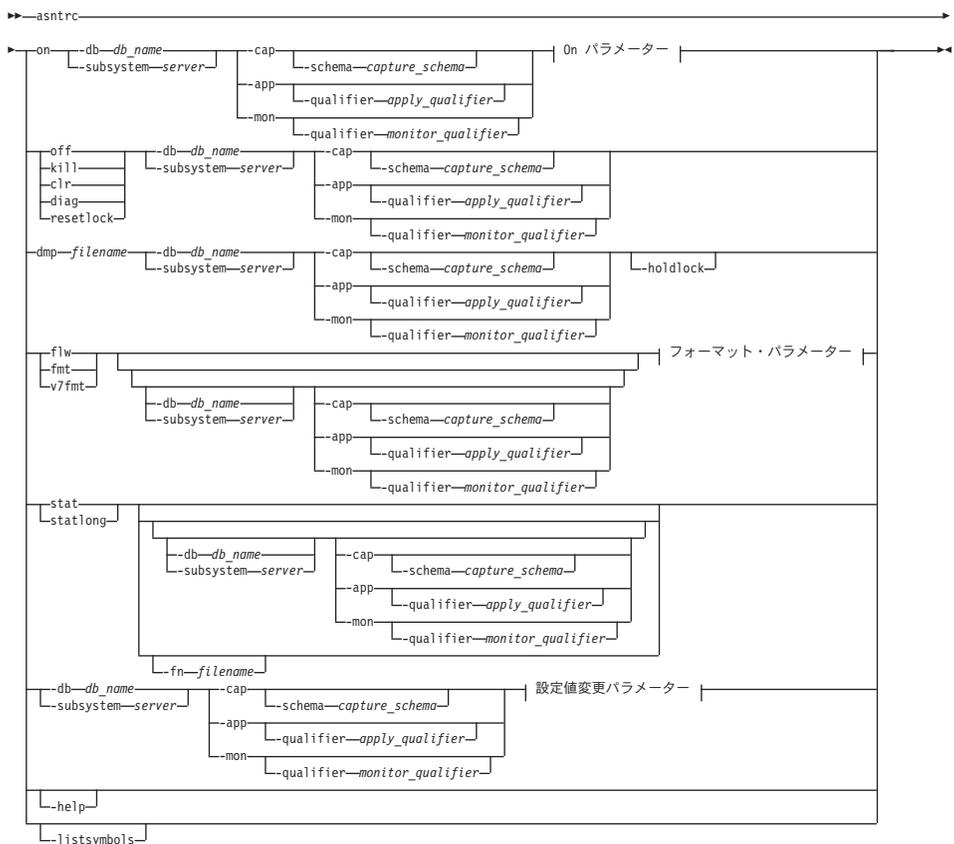
関連資料:

- 355 ページの『asnsrct: キャプチャー、アプライ、またはレプリケーション・アラート・モニターを開始する DB2 レプリケーション・サービスの作成 (Windows のみ)』

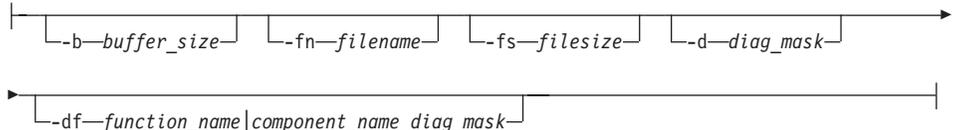
asntrc: レプリケーション・トレース機能の操作 (UNIX、Windows、z/OS)

UNIX、Windows、および z/OS 上の UNIX System Services (USS) でトレース機能を実行するには、**asntrc** コマンドを使用します。トレース機能は、キャプチャー、アプライおよびレプリケーション・アラート・モニター・プログラムから、プログラム・フロー情報をログに記録します。このトレース情報を IBM ソフトウェア・サポートに提供して、トラブルシューティングに役立てることができます。このコマンドは、オペレーティング・システムのプロンプトまたはシェル・スクリプト内で実行します。

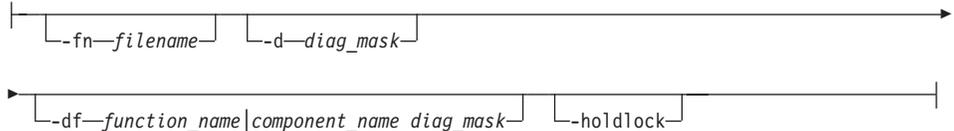
asntrc コマンドを使用してトレース機能を実行するための構文



On パラメーター:



フォーマット・パラメーター:



設定値変更パラメーター:



表 29 は、呼び出しパラメーターを定義します。

表 29. UNIX、Windows、および z/OS オペレーティング・システム用 asnlrc 呼び出しパラメーター定義

| パラメーター | 定義 |
|-------------------------------------|---|
| on | 特定のキャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニター・プログラムについて、トレース機能をオンにすることを指定します。トレース機能は、トレース処理中に使用する共有メモリー・セグメントを作成します。 |
| -db db_name | UNIX および Windows の場合のみ: トレースの対象となるアプライ・プログラムまたはキャプチャー・プログラムが存在する、データベース名を指定します。 |
| -subsystem server | z/OS の場合のみ: トレースの対象となる、キャプチャー、アプライ、またはレプリケーション・アラート・モニター・プログラムが存在する、データベース・サーバーの名前を指定します。 |
| -cap | キャプチャー・プログラムをトレースすることを指定します。キャプチャー・プログラムは -schema パラメーターで識別されます。 |
| -schema capture_schema | トレースするキャプチャー・プログラムの名前を指定します。キャプチャー・プログラムは指定したキャプチャー・スキーマにより識別されます。このパラメーターは -cap パラメーターと一緒に使用します。 |
| -app | アプライ・プログラムをトレースすることを指定します。アプライ・プログラムは -qualifier パラメーターで識別されます。 |
| -qualifier apply_qualifier | トレースするアプライ・プログラムの名前を指定します。このアプライ・プログラムは、指定したアプライ修飾子により識別されます。このパラメーターは -app パラメーターと一緒に使用します。 |
| -mon | レプリケーション・アラート・モニター・プログラムをトレースすることを指定します。レプリケーション・アラート・モニター・プログラムは -qualifier パラメーターで識別されます。 |
| -qualifier monitor_qualifier | トレースするレプリケーション・アラート・モニター・プログラムの名前を指定します。このレプリケーション・アラート・モニター・プログラムは、指定したモニター修飾子により識別されます。このパラメーターは -mon パラメーターと一緒に使用します。 |
| off | 特定のキャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニター・プログラムについて、トレース機能をオフにし、使用中の共有メモリー・セグメントを解放することを指定します。 |
| kill | トレース機能を強制的に異常終了させることを指定します。 このパラメーターは、何らかの問題により、トレース機能を off パラメーターでオフにできない場合のみ使用してください。 |
| clr | トレース・バッファーをクリアすることを指定します。このパラメーターは、トレース・バッファーの内容を消去しますが、バッファーはアクティブのままにします。 |
| diag | トレース機能の実行中に、フィルター設定を表示することを指定します。 |
| resetlock | トレース機能のバッファー・ラッチを解放することを指定します。このパラメーターは、エラー状態が起こり、トレース・プログラムがバッファー・ラッチを保留したまま終了した場合に、バッファー・ラッチをエラー状態からリカバーできるようにします。 |

表 29. UNIX、Windows、および z/OS オペレーティング・システム用 *asnlrc* 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 |
|------------------------------|---|
| dmp <i>filename</i> | トレース・バッファの現在の内容をファイルに書き込むことを指定します。 |
| -holdlock | トレース機能がバッファをコピーするためのメモリーが不足している場合でも、ロックを保留している間に、トレース機能がファイルのダンプまたはコマンドの出力を完了できることを指定します。 |
| flw | トレース機能が作成し、共有メモリーまたはファイルに保管した要約情報を表示することを指定します。この情報には、プログラム・フローが含まれ、それぞれの処理およびスレッドごとに、関数と呼び出しのスタック構造がわかるように字下げして表示されます。 |
| fmt | トレース機能が作成し、共有メモリーまたはファイルに保管した詳細情報を表示することを指定します。このパラメーターは、トレースしたデータ構造の内容全体を発生順に表示します。 |
| v7fmt | トレース機能が作成し、共有メモリーまたはファイルに保管した情報を表示することを指定します。このトレース情報はバージョン 7 のフォーマットで表示されます。 |
| stat | トレース機能の状況を表示することを指定します。この状況情報には、トレース・バージョン、アプリケーション・バージョン、エンタリー数、バッファ・サイズ、使用中のバッファ量、状況コード、およびプログラム・タイム・スタンプが含まれます。 |
| statlong | トレース機能の状況に z/OS バージョン・レベル情報を追加して表示することを指定します。この追加情報には、アプリケーション内の各モジュールのサービス・レベルが含まれ、長ストリングのテキストとして表示されます。 |
| -fn <i>filename</i> | ミラーリングされたトレース情報を含むファイル名を指定します。ここには、トレース機能からのすべての出力が含まれます。 |
| -help | 有効なコマンド・パラメーターを記述と一緒に表示します。 |
| -listsymbols | -df パラメーターで使用できる有効な関数およびコンポーネント ID を表示します。 |
| -b <i>buffer_size</i> | トレース・バッファのサイズをバイト単位で指定します。数値の後に、キロバイトなら K を、メガバイトなら M を指定できます。これらの文字には大文字小文字の区別はありません。 |
| -fs <i>filesize</i> | ミラーリングされたトレース情報ファイルのサイズ制限をバイト単位で指定します。 |

表 29. UNIX、Windows、および z/OS オペレーティング・システム用 asnlrc 呼び出しパラメーター定義 (続き)

| パラメーター | 定義 | | | | | | | | |
|--|--|---|---------------------------|---|-------------------------------------|---|---------------------------|---|--------------|
| -d <i>diag_mask</i> | <p>トレース機能により記録されるトレース・レコードのタイプを指定します。トレース・レコードは、以下の診断マスク番号により分類されます。</p> <table border="0"> <tr> <td style="vertical-align: top;">1</td> <td>フロー・データ。関数の入力点と出口点が含まれます。</td> </tr> <tr> <td style="vertical-align: top;">2</td> <td>基本データ。トレース機能が検出したすべての主要なイベントが含まれます。</td> </tr> <tr> <td style="vertical-align: top;">3</td> <td>詳細データ。主要なイベントとその記述が含まれます。</td> </tr> <tr> <td style="vertical-align: top;">4</td> <td>パフォーマンス・データ。</td> </tr> </table> <p>重要: 診断マスク番号の大きいものは、診断マスク番号の小さいものを包含していません。</p> <p>これらの番号を 1 つまたは複数入力し、必要なトレース・レコードだけを含む診断マスクを作成することができます。たとえば、-d 4 を指定すると、パフォーマンス・データだけが記録されます。フローとパフォーマンスのデータだけを記録するには -d 1,4 と指定し、すべてのトレース・レコードを記録するには -d 1,2,3,4 (デフォルト) と指定します。番号はコンマで区切ります。</p> <p>トレース機能がグローバル・トレース・レコードを記録しないようにするには、診断マスク番号 0 (ゼロ) を入力します。トレース機能に新しい診断マスク番号を指定する前に、診断レベルをリセットするには、-d 0 を入力します。</p> | 1 | フロー・データ。関数の入力点と出口点が含まれます。 | 2 | 基本データ。トレース機能が検出したすべての主要なイベントが含まれます。 | 3 | 詳細データ。主要なイベントとその記述が含まれます。 | 4 | パフォーマンス・データ。 |
| 1 | フロー・データ。関数の入力点と出口点が含まれます。 | | | | | | | | |
| 2 | 基本データ。トレース機能が検出したすべての主要なイベントが含まれます。 | | | | | | | | |
| 3 | 詳細データ。主要なイベントとその記述が含まれます。 | | | | | | | | |
| 4 | パフォーマンス・データ。 | | | | | | | | |
| -df <i>function_name/component_name</i> <i>diag_mask</i> | <p>特定の関数またはコンポーネント ID をトレースすることを指定します。</p> <p>関数またはコンポーネント ID 名の後に診断マスク番号 (1、2、3、4) を入力します。1 つまたは複数の番号を入力できます。番号はコンマで区切ります。</p> | | | | | | | | |

asnlrc の例

次の例は、**asnlrc** コマンドの使用法を示しています。

例 1

UNIX または Windows の下で実行中のキャプチャー・プログラムをトレースするには、次のようにします。

1. 最大のバッファー・サイズとファイル・サイズを持つトレース・ファイル名を指定して、トレース機能を開始します。

```
asnlrc on -db mydb -cap -schema myschema -b 256k -fn myfile.trc -fs 500m
```

2. キャプチャー・プログラムを始動し、適切な長さの時間実行させます。
3. トレース機能がオンになっている間に、共有メモリーからデータを直接表示します。

トレース機能から処理とスレッドのサマリー情報を表示するには、次のように指定します。

```
asnlrc flw -db mydb -cap -schema myschema
```

キャプチャー・ログ・リーダーからのみ、フロー、基本、詳細、およびパフォーマンスのデータ・レコードを表示するには、次のように指定します。

```
asnlrc fmt -db mydb -cap -schema myschema -d 0
      -df "Capture Log Read" 1,2,3,4
```

4. トレース機能を停止します。

```
asnlrc off -db mydb -cap -schema myschema
```

トレース・ファイルには、キャプチャー・プログラムの始動時点から、トレース機能をオフにした時点までに生成された、すべてのキャプチャー・プログラム・トレース・データが含まれます。

5. トレース機能を停止した後、生成されたバイナリー・ファイルからのデータをフォーマットします。

```
asnlrc flw -fn myfile.trc
```

および

```
asnlrc fmt -fn myfile.trc -d 0 -df "Capture Log Read" 1,2,3,4
```

例 2

UNIX または Windows の下で実行中のレプリケーション・アラート・モニター・プログラムのトレース機能を開始するには、次のように指定します。

```
asnlrc on -db mydb -mon -qualifier monq
```

例 3

UNIX または Windows で実行中のアプライ・プログラムのパフォーマンス・データだけをトレースするには、次のように指定します。

```
asnlrc on -db mydb -app -qualifier aq1 -b 256k -fn myfile.trc -d 4
```

例 4

z/OS の下で実行中のキャプチャー・プログラムのすべてのフローおよびパフォーマンス・データをトレースするには、次のように指定します。

```
asnlrc on -subsystem dbserv1 -cap -schema myschema -b 256k
      -fn myfile.trc -d 1,4
```

例 5

すべてのグローバルなパフォーマンス・データおよび、UNIX または Windows の下で実行中のキャプチャー・プログラムの、特定のキャプチャー・ログ・リーダーのフロー・データをトレースするには、次のように指定します。

asnlrc

```
asnlrc on -db mydb -cap -schema myschema -b 256k -fn myfile.trc -d 4  
-df "Capture Log Read" 1
```

例 6

UNIX または Windows の下で実行中のキャプチャー・プログラムをトレースし、トレース機能の特定の時点のイメージを表示して保管するには、次のようにします。

1. 最新のレコードを保留するために十分なバッファ・サイズを指定して、トレース・コマンドを始動します。

```
asnlrc on -db mydb -cap -schema myschema -b 4m
```

2. キャプチャー・プログラムを始動し、適切な長さの時間実行させます。
3. 共有メモリーに保管された、特定の時点の詳細なトレース情報を表示します。

```
asnlrc fmt -db mydb -cap -schema myschema
```

4. 特定の時点のトレース情報をファイルに保管します。

```
asnlrc dmp myfile.trc -db mydb -cap -schema myschema
```

5. トレース機能を停止します。

```
asnlrc off -db mydb -cap -schema myschema
```

関連資料:

- 454 ページの 『WRKDPRTTC: DPR トレース機能の使用方方法 (OS/400)』

第 18 章 レプリケーション用のシステム・コマンド (OS/400)

この章では、iSeries サーバーの OS/400 オペレーティング・システムの下で実行するレプリケーション・コマンドについて説明します。これらのコマンドは、オペレーティング・システムのコマンド・プロンプトか、またはコマンド行プログラムを使用して入力することができます。

この章には、コマンド別のセクションがあります。各セクションには、コマンドの簡単な記述、構文図、およびパラメーターとそれに対応する定義の表があります。各セクションの最後には、コマンドの使用例と、関連情報への相互参照があります。

コマンドには、以下のものがあります。

- 366 ページの『ADDDPRREG: DPR 登録の追加 (OS/400)』
- 374 ページの『ADDDPRSUB: DPR サブスクリプション・セットの追加 (OS/400)』
- 390 ページの『ADDDPRSUBM: DPR サブスクリプション・セットのメンバーの追加 (OS/400)』
- 400 ページの『ANZDPR: アナライザーの操作 (OS/400)』
- 404 ページの『CHGDPRCAPA: DPR キャプチャー属性の変更 (OS/400)』
- 408 ページの『CRTDPRTBL: レプリケーション・コントロール表の作成 (OS/400)』
- 409 ページの『ENDDPRAPY: アプライ・プログラムの停止 (OS/400)』
- 413 ページの『ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)』
- 415 ページの『GRTPRAUT: ユーザーの許可 (OS/400)』
- 424 ページの『INZDPRCAP: DPR キャプチャー・プログラムの再初期化 (OS/400)』
- 426 ページの『OVRDPRCAPA: DPR キャプチャー属性のオーバーライド (OS/400)』
- 431 ページの『RMVDPRREG: DPR 登録の除去 (OS/400)』
- 432 ページの『RMVDPRSUB: DPR サブスクリプション・セットの除去 (OS/400)』
- 434 ページの『RMVDPRSUBM: DPR サブスクリプション・セット・メンバーの除去 (OS/400)』
- 436 ページの『RVKDPRAUT: 権限の取り消し (OS/400)』
- 438 ページの『STRDPAPY: アプライ・プログラムの始動 (OS/400)』
- 446 ページの『STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)』
- 454 ページの『WRKDPTRC: DPR トレース機能の使用方法 (OS/400)』

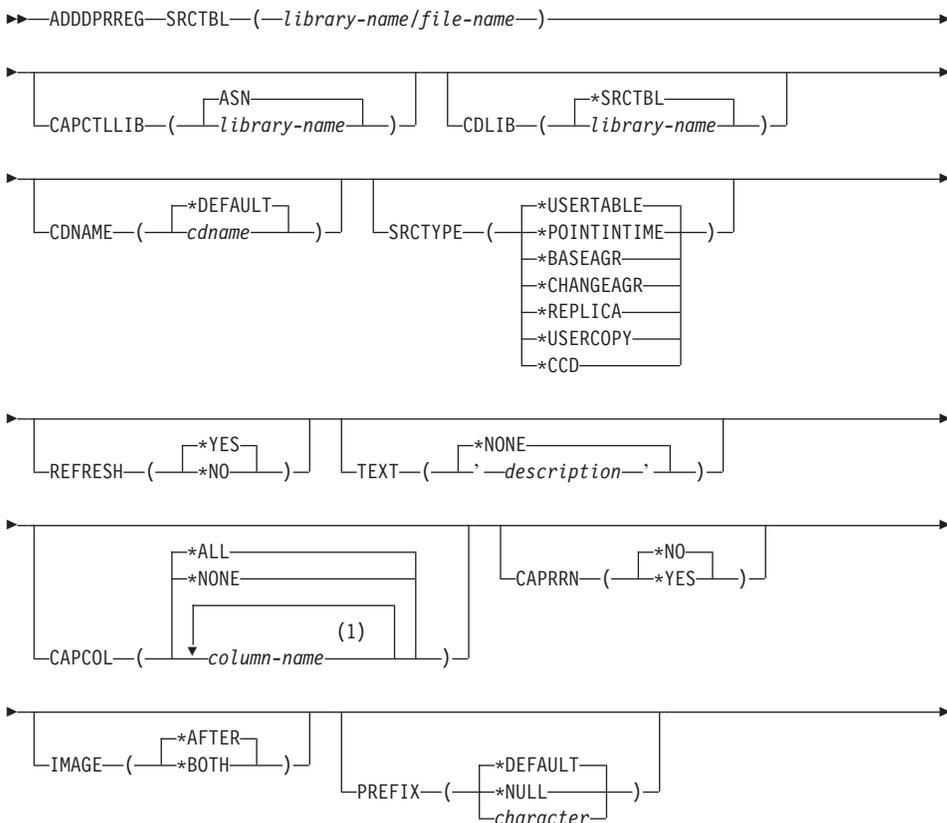
ADDDPRREG: DPR 登録の追加 (OS/400)

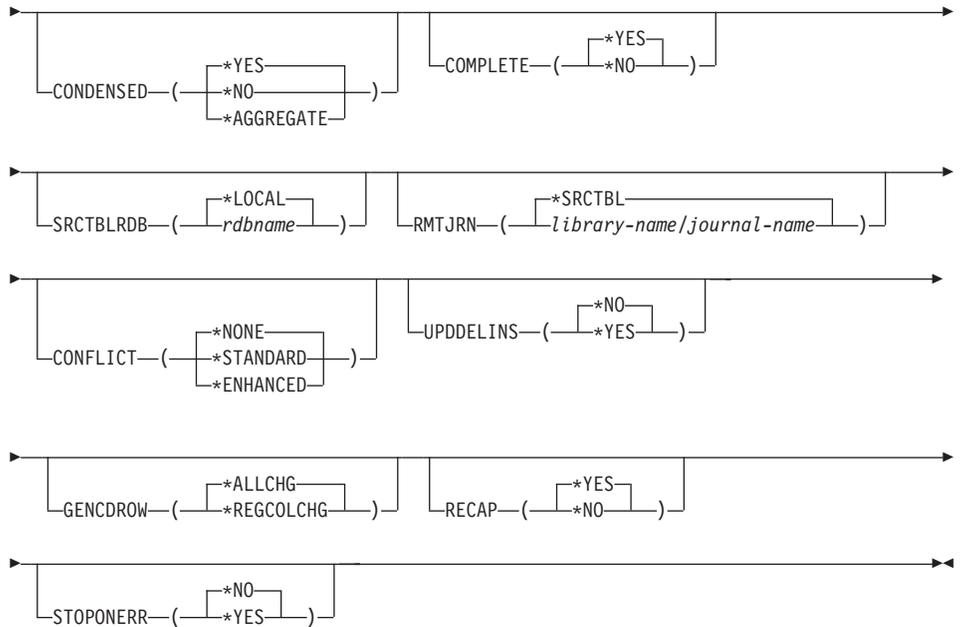
表を DB2 DataPropagator for iSeries のソース表として登録するには、DPR 登録の追加 (ADDDPRREG) コマンドを使用します。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

ADDDPRREG コマンドを使用して表を登録するための構文





注:

- 1 列名は 300 個まで指定できます。

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| SRCTBL | <p>ソース表として登録する表を指定します。キャプチャー・プログラムは、OS/400 ライブラリー内の任意の物理ファイルまたは、外部で定義され、1 つのフォーマットを持つ集合をサポートします。このパラメーターは必須です。</p> <p><i>library-name/file-name</i> 登録する表の修飾名を表します。</p> |
| CAPCTLLIB | <p>キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。</p> <p><i>library-name</i> キャプチャー・コントロール表を含むライブラリーの名前。 CRDTPRTBL コマンドに CAPCTLLIB パラメーターを指定して、このライブラリーを作成することができます。</p> |
| CDLIB | <p>この登録されたソースの変更データ (CD) 表を作成するライブラリーを指定します。</p> <p>*SRCTBL (デフォルト) ソース表が存在するライブラリー内に CD 表を作成します。</p> <p><i>library-name</i> 指定したライブラリー名に CD 表を作成します。</p> |

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| CDNAME | <p>変更データ (CD) 表の名前を指定します。</p> <p>*DEFAULT (デフォルト)</p> <p>デフォルト名 (現行タイム・スタンプを基にしたもの) を使用して CD 表を作成します。たとえば、現行のタイム・スタンプが 2002 年 1 月 23 日 9 時 58 分 26 秒の場合、デフォルト名は ASN020123095826CD となります。</p> <p><i>cdname</i></p> <p>ここに指定した名前を CD 表を作成します。</p> |
| SRCTYPE | <p>登録するソース表のタイプを指定します。ご使用のレプリケーション構成に基づいてソース・タイプを選択します。</p> <ul style="list-style-type: none"> • 基本データ配分またはデータ統合構成の場合は、デフォルトの USERTABLE を使用します。 • update-anywhere 構成の場合は REPLICA を使用します。 • 複数の階層からなる構成を持ち、ターゲット表を、レプリケーション構成内の下位の層のためのソースにしたい場合は、 POINTINTIME、BASEAGR、CHANGEAGR、USERCOPY、または CCD を使用します。 <p>既存のターゲット表をソースとして登録する場合、指定されたソース・タイプに示された IBMSNAP 表の列がターゲット表に含まれていないと、登録は失敗します。</p> |

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|--------------|---|
| SRCTYPE (続き) | <p>*USERTABLE (デフォルト)</p> <p>ユーザー・データベース表。登録された表の最もよくあるタイプです。表には、DB2 DataPropagator for iSeries の列 ID である、IBMSNAP または IBMQSQ で始まる列を含めることはできません。</p> <p>*POINTINTIME</p> <p>ポイント・イン・タイム・コピー表。これは、ソース表の内容の一部またはすべてと一致する内容を含み、さらに、DB2 DataPropagator for iSeries システム列を保持します。このシステム列は、特定の行がソース・システムで最後に挿入または更新されたポイント・イン・タイムを示します。表には IBMSNAP_LOGMARKER タイム・スタンプ列を含める必要があり、オプションとして IBMQSQ_RRN と呼ばれる INTEGER 列を含めることができます。</p> <p>*BASEAGR</p> <p>基本集約コピー。ユーザー表またはポイント・イン・タイム表からインターバルごとに集約されたデータが入ります。基礎集約表には、IBMSNAP_HLOGMARKER および IBMSNAP_LLOGMARKER タイム・スタンプ列を含める必要があります。</p> <p>*CHANGEAGR</p> <p>変更を集約したコピー表。ソース表に記録された変更に基づくデータの集約が入ります。この表には、IBMSNAP_HLOGMARKER および IBMSNAP_LLOGMARKER タイム・スタンプ列を含める必要があります。</p> <p>*REPLICA</p> <p>レプリカ・サブスクリプションのターゲット表。ターゲット表からの変更を元のソース表に戻せるようにするには、このタイプの表を登録します。この表には、DB2 DataPropagator for iSeries のシステム列または、DataPropagator for iSeries の列 ID である、IBMSNAP または IBMQSQ で始まる列を含めることはできません。この表は、元のソース表からのすべての列を保持します。</p> <p>*USERCOPY</p> <p>ソース表の内容のすべてまたは一部と一致する内容を持つ、ターゲット表です。ユーザー・コピー表はユーザー・データ列のみ保持します。</p> |
| SRCTYPE (続き) | <p>*CCD</p> <p>整合変更データ (CCD) 表。ソース表からのトランザクション整合性のあるデータが入ります。表には、次のように定義された列を含める必要があります。</p> <ul style="list-style-type: none"> • IBMSNAP_INTENTSEQ CHAR(10) FOR BIT DATA NOT NULL • IBMSNAP_OPERATION CHAR(1) NOT NULL • IBMSNAP_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL • IBMSNAP_LOGMARKER TIMESTAMP NOT NULL |

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| REFRESH | <p>フル・リフレッシュ機能を使用可能にするかどうかを指定します。この値を使用して、ソース・データベースからのフル・リフレッシュを実行するアプライ・プログラムの機能をオフにすることができます。</p> <p>*YES (デフォルト) フル・リフレッシュを使用可能にします。</p> <p>*NO フル・リフレッシュは使用不可です。</p> <p>ターゲット表が基礎集約表または変更集約表の場合、このパラメーターは *NO にする必要があります。</p> |
| TEXT | <p>この登録に関する記述テキストを指定します。</p> <p>*NONE (デフォルト) エントリーには記述がありません。</p> <p><i>description</i> この登録を記述するテキスト。最大 50 文字を入力でき、テキストは単一引用符で囲む必要があります。</p> |
| CAPCOL | <p>この登録済み表について、どの列の変更をキャプチャーするかを指定します。</p> <p>*ALL (デフォルト) すべての列の変更をキャプチャーします。</p> <p>*NONE この表の変更をキャプチャーしません。この値は、この表をフル・リフレッシュ用としてのみ登録する場合に指定します。登録されたこの表については変更データ (CD) 表は作成されず、キャプチャー・プログラムは表の変更をキャプチャーしません。</p> <p><i>column-name</i> その列の変更をキャプチャーする列名。列名は 300 個まで指定できます。列名はスペースで区切ります。</p> |
| CAPRRN | <p>変更されたレコードの相対レコード番号 (RRN) をキャプチャーするかどうかを指定します。</p> <p>*NO (デフォルト) 相対レコード番号をキャプチャーしません。</p> <p>*YES 相対レコード番号をキャプチャーします。変更データ (CD) 表に IBMQSQ_RRN と呼ばれる追加の列が作成されます。</p> <p>ソース表にユニーク・キーがない場合のみ、このパラメーターを *YES にします。</p> |

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| IMAGE | <p>変更データ (CD) 表に、ソース表の変更前イメージと変更後イメージの両方を含めるかどうかを指定します。この値は、キャプチャー列パラメーター (CAPCOL) に指定したすべての列にグローバルに適用されます。</p> <p>この IMAGE パラメーターは、CAPCOL パラメーターが *NONE の場合は無効です。</p> <p>このパラメーターに *AFTER を指定した場合でも、ソース表は *BOTH イメージでジャーナルに記録する必要があります。</p> <p>*AFTER (デフォルト) キャプチャー・プログラムはソース表の変更後イメージのみ CD 表に記録します。</p> <p>*BOTH キャプチャー・プログラムはソース表の変更前イメージと変更後イメージの両方を CD 表に記録します。</p> |
| PREFIX | <p>変更データ (CD) 表の変更前イメージの列を識別する接頭部文字を指定します。ソース表の登録された列名に、この接頭部文字で始まる列名がないことを確認してください。</p> <p>*DEFAULT (デフォルト) デフォルトの接頭部 (@) が使用されます。</p> <p>*NULL 変更前イメージは取り込まれません。この値は、IMAGE パラメーターを *BOTH にした場合は無効です。</p> <p><i>character</i> オブジェクト名に使用できる任意の 1 つの英字。</p> |
| CONDENSED | <p>ソース表が圧縮されているかどうかを指定します。コンデンス表には、現行データが含まれ、1 つの主キーの値に対して複数の行が含まれることはありません。</p> <p>*YES (デフォルト) ソース表は圧縮されています。</p> <p>*NO ソース表は圧縮されていません。</p> <p>*AGGREGATE ソース表のタイプは *BASEAGR (基本集約) または *CHANGEAGR (変更集約) のいずれかです。この値を使用する場合、COMPLETE パラメーターを *NO にする必要があります。</p> |
| COMPLETE | <p>ソース表が完全であるかどうかを指定します。これは、表に、対象となる主キー値ごとに 1 行が含まれていることを意味します。</p> <p>*YES (デフォルト) ソース表は完全です。</p> <p>*NO ソース表は完全ではありません。</p> |

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| SRCTBLRDB | <p>リモート・ジャーナリングを使用するかどうかを指定します。リモート・ジャーナリングでは、ソース表とリモート・ジャーナルは別のシステムにあります。このパラメーターを使用して、ソース表のロケーションを指定します。</p> <p>*LOCAL (デフォルト) ソース表はローカルにあります (ADDDPRREG コマンドを実行するマシン上に)。</p> <p><i>rdbname</i> ソース表が存在するリレーショナル・データベースの名前。 RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、このリレーショナル・データベース名を見つけることができます。</p> |
| RMTJRN | <p>このジャーナルの名前とソース・システム上のジャーナルの名前が異なる場合に、リモート・ジャーナルの名前を指定します。このコマンドは、リモート・ジャーナルが存在するシステムから実行する必要があります。</p> <p>*SRCTBL (デフォルト) リモート・ジャーナル名は、ソース表のジャーナル名と同じです。</p> <p><i>library-name/journal-name</i> このシステムにあり、リモート・ソース表のジャーナリングに使用される、修飾されたライブラリー名とジャーナル名。</p> <p>リモート・ジャーナル名は、SRCTBLRDB パラメーターでリモート・ソース表のロケーションを指定した場合のみ指定できます。</p> |
| CONFLICT | <p>レプリカ・サブスクリプション内の競合を検出するときにアプライ・プログラムが使用する、競合レベルを指定します。</p> <p>*NONE (デフォルト) 競合検出を行いません。</p> <p>*STANDARD 適度な競合検出。アプライ・プログラムは、レプリカ変更データ (CD) 表内のすでに取り込まれた行について、競合を探します。</p> <p>*ENHANCED 拡張競合検出を行います。このオプションは、すべてのレプリカとソース表の間に最良のデータ保全性を提供します。</p> |
| UPDDELINS | <p>キャプチャー・プログラムが、更新されたソース・データを、変更データ (CD) 表にどのように保管するかを決めます。</p> <p>*NO (デフォルト) キャプチャー・プログラムは、それぞれのソース変更を CD 表内の 1 つの行に保管します。</p> <p>*YES キャプチャー・プログラムは、それぞれのソース変更を CD 表内に 2 行を使用して保管し、その 1 行は削除用、もう 1 行は挿入用です。アプライ・プログラムは最初に削除用の行を処理し、2 番目に挿入用の行を処理します。</p> |

表 30. ADDDPRREG コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| GENCDROW | <p>キャプチャー・プログラムがソース表のすべての行から変更をキャプチャーするかどうかを指定します。</p> <p>*ALLCHG (デフォルト) キャプチャー・プログラムはソース表のすべての行から変更をキャプチャー (登録されていない列の変更も含む)、これらの変更を変更データ (CD) 表に追加します。</p> <p>*REGCOLCHG キャプチャー・プログラムは、登録された列に変更があった場合のみ変更をキャプチャーします。その後、キャプチャー・プログラムはこれらの行を CD 表に追加します。</p> <p>*REGCOLCHG は、CAPCOL パラメーターが *ALL または *NONE の場合には指定できません。</p> |
| RECAP | <p>アプライ・プログラムが行った変更を、キャプチャー・プログラムが再度キャプチャーするかどうかを指定します。</p> <p>*YES (デフォルト) アプライ・プログラムがソース表を変更した場合に、その変更をキャプチャーし、変更データ (CD) 表に入れます。</p> <p>*NO アプライ・プログラムがソース表を変更した場合に、その変更は取り込まず、したがって、変更データ (CD) 表には入れません。このオプションは、REPLICA タイプの表を登録する場合に使用してください。</p> |
| STOPONERR | <p>キャプチャー・プログラムがエラーを検出した場合に、キャプチャー・プログラムを停止するかどうかを指定します。</p> <p>*NO (デフォルト) キャプチャー・プログラムはエラーを検出しても停止しません。キャプチャー・プログラムはメッセージを出し、エラーの原因となった登録を非活動化してから、処理を継続します。</p> <p>*YES キャプチャー・プログラムはエラーを検出した場合に、メッセージを出してから停止します。</p> |

ADDDPRREG の例

次の例は、**ADDDPRREG** コマンドの使用法を示しています。

例 1

デフォルトのキャプチャー・スキーマの下に、HR ライブラリーから EMPLOYEE という名前のソース表を登録します。

```
ADDDPRREG SRCTBL(HR/EMPLOYEE)
```

例 2

BSN キャプチャー・スキーマの下に、HR ライブラリーから EMPLOYEE という名前のソース表を登録し、HRCDLIB ライブラリーの下に CDEMPLOYEE という名前の CD 表を作成します。

ADDDPRREG

```
ADDDPRREG SRCTBL(HR/EMPLOYEE) CAPCTLLIB(BSN) CDLIB(HRCDLIB) CDNAME(CDEMPLOYEE)
```

例 3

BSN キャプチャー・スキーマの下に、DEPT ライブラリーから SALES という名前のポイント・イン・タイム・コピーのソース表を登録します。

```
ADDDPRREG SRCTBL(DEPT/SALES) CAPCTLLIB(BSN) SRCTYPE(*POINTINTIME)
```

例 4

DEPT ライブラリーから SALES という名前のソース表を登録し、CD 表には、ソース表の変更について変更前イメージと変更後イメージの両方を含めます。

```
ADDDPRREG SRCTBL(DEPT/SALES) IMAGE(*BOTH)
```

例 5

DEPT ライブラリーから SALES という名前のソース表 (RMTRDB1 という名前のリレーショナル・データベースで、リモート・ジャーナルを使用する) を登録します。

```
ADDDPRREG SRCTBL(DEPT/SALES) SRCTBLRDB(RMTRDB1) RMTJRN(RMTJRNLIB/RMTJRN)
```

例 6

HR ライブラリーから EMPLOYEE ソース表を登録し、EMPNO、NAME、DEPT、および NETPAY の列についてのみ、変更をキャプチャーします。

```
ADDDPRREG SRCTBL(HR/EMPLOYEE) CAPCOL(EMPNO NAME DEPT NETPAY)
```

関連作業:

- 43 ページの 『第 3 章 表およびビューをレプリケーション・ソースとして登録する』

ADDDPRSUB: DPR サブスクリプション・セットの追加 (OS/400)

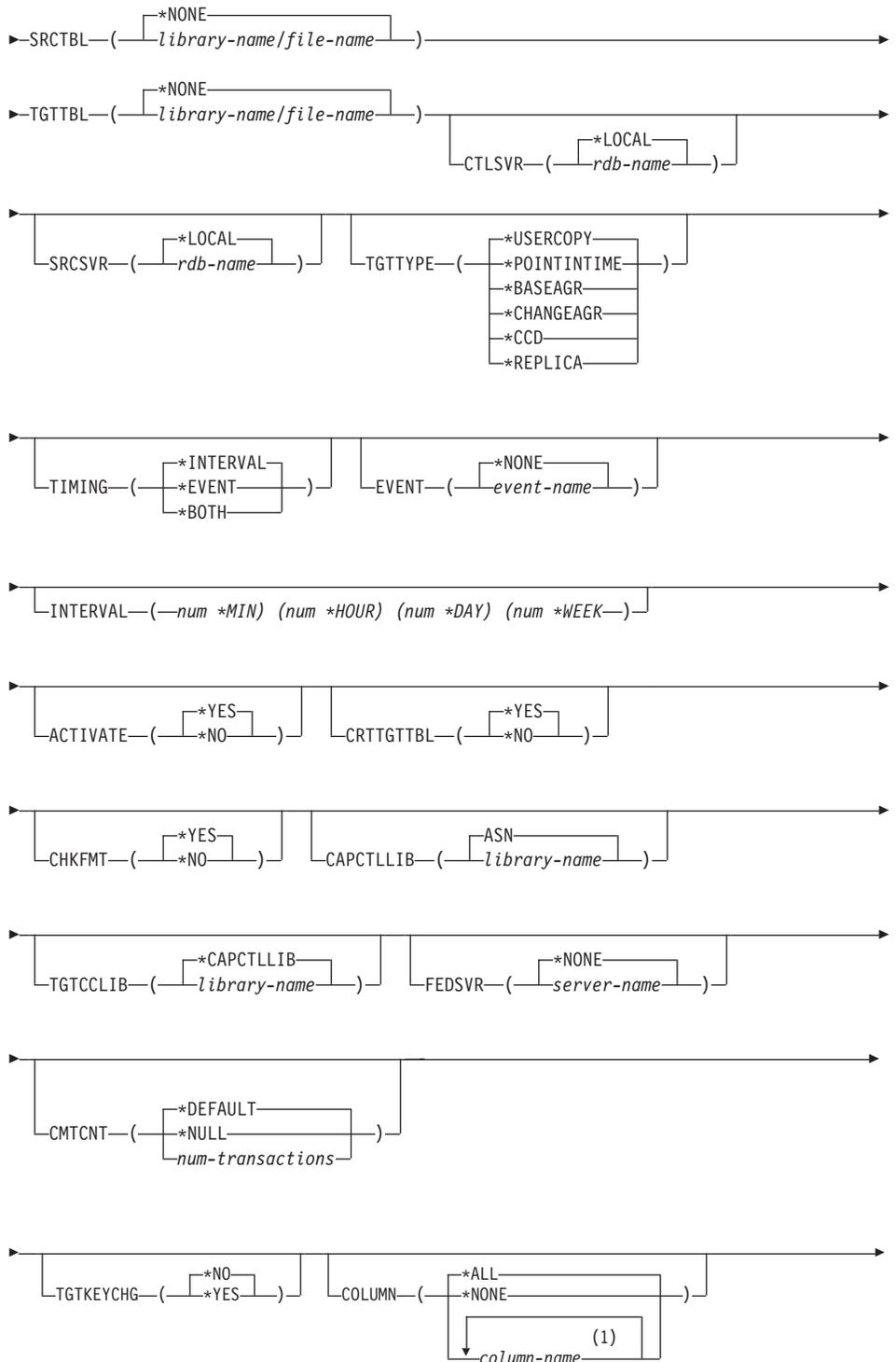
1 つのメンバーを持つ、またはメンバーを持たないサブスクリプション・セットを作成するには、DPR サブスクリプション・セットの追加 (**ADDDPRSUB**) コマンドを使用します。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

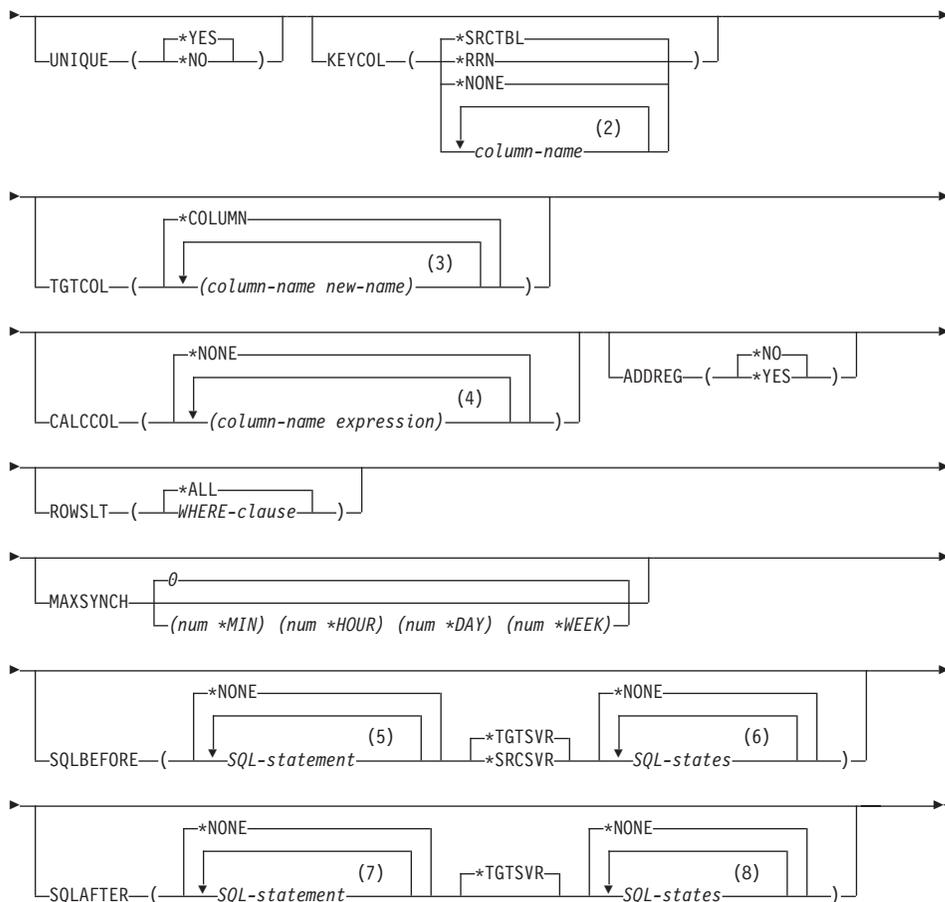
このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

ADDDPRSUB コマンドを使用してサブスクリプション・セットを作成するための構文

▶—ADDDPRSUB—APYQUAL—(—*apply-qualifier*—)—SETNAME—(—*set-name*—)—▶



ADDDPRSUB



注:

- 1 列名は 300 個まで指定できます。
- 2 列名は 120 個まで指定できます。
- 3 列名は 300 個まで指定できます。
- 4 100 個までの列名および式を指定できます。
- 5 SQL ステートメントは 3 個まで指定できます。
- 6 10 個までの SQLSTATES を指定できます。
- 7 SQL ステートメントは 3 個まで指定できます。
- 8 10 個までの SQLSTATES を指定できます。

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|----------------|--|
| APYQUAL | <p>どのアプライ・プログラムがこのサブスクリプション・セットを処理するかを示すアプライ修飾子を指定します。アプライ修飾子の下のサブスクリプション・セットは別のジョブで実行されます。このパラメーターは必須です。</p> <p><i>apply-qualifier</i> アプライ修飾子の名前。</p> |
| SETNAME | <p>サブスクリプション・セットの名前を指定します。このパラメーターは必須です。</p> <p><i>set-name</i> サブスクリプション・セットの名前。入力するサブスクリプション・セット名は、指定されたアプライ修飾子について固有でなければなりません。固有でない場合、ADDDPRSUB コマンドはエラーになります。アプライ・プログラムはターゲット表のセットをグループとして扱うので、何らかの理由で 1 つのターゲット表が失敗すると、サブスクリプション・セット全体が失敗します。</p> |
| SRCTBL | <p>情報をサブスクリプション・セットにコピーするために使用する、ソース表の名前を指定します。この表をサブスクリプション・セットのメンバーにするには、この表をキャプチャー・コントロール・サーバーにあらかじめ登録しておく必要があります。このパラメーターは必須です。</p> <p>*NONE (デフォルト) このサブスクリプション・セットはソース・メンバーを持ちません。メンバーのないサブスクリプション・セットを作成する場合に使用します。</p> <p><i>library-name/file-name</i> ソース表の修飾名。1 つのメンバーを持つサブスクリプション・セットを作成する場合に使用します。</p> |
| TGTTBL | <p>ターゲット表の名前を指定します。 CRTTGTTBL パラメーターを *YES に設定し、ターゲット表が存在しないと、ターゲット表は自動的に作成されます。このパラメーターは必須です。</p> <p>*NONE (デフォルト) このサブスクリプション・セットはターゲット・メンバーを持ちません。メンバーのないサブスクリプション・セットを作成する場合に使用します。</p> <p><i>library-name/file-name</i> ターゲット表の修飾名。1 つのメンバーを持つサブスクリプション・セットを作成する場合に使用します。</p> |
| CTLSVR | <p>アプライ・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) アプライ・コントロール表はローカルに存在します (ADDDPRSUB コマンドを実行するマシン上)。</p> <p><i>rdb-name</i> アプライ・コントロール表が置かれているリレーショナル・データベースの名前。RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| SRCSVR | <p>キャプチャー・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) ソース表はローカル・マシンに登録されています (ADDDPRSUB コマンドを実行するマシン)。</p> <p><i>rdb-name</i> キャプチャー・コントロール表が存在するリレーショナル・データベースの名前。 RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> |
| TGTTYPE | <p>ターゲット表のタイプを指定します。これらのいずれかのタイプとしてターゲット表を作成した後、DPR 登録の追加 (ADDDPRREG) コマンドの SRCTBL パラメーターにこのパラメーター値を使用し、このターゲット表を multi-tier レプリケーションのソース表として登録することができます。</p> <p>*USERCOPY (デフォルト) ターゲット表はユーザー・コピーであり、これはソース表の内容のすべてまたは一部と一致する内容を持つ、ターゲット表です。ユーザー・コピーはポイント・イン・タイムのコピーのように扱われますが、ポイント・イン・タイムのターゲット表に存在する、DB2 DataPropagator for iSeries システム列は 1 つも含まれていません。</p> <p>この値は、KEYCOL パラメーターに *RRN の値が指定されている場合は無効です。</p> <p>SRCTBL パラメーターで指定した表は、ユーザー・データベース、ポイント・イン・タイム・コピー、または整合変更データ (CCD) のいずれかでなければなりません。</p> <p>重要: ターゲット表がすでに存在する場合、DB2 DataPropagator for iSeries は、これに対する変更を自動的にジャーナルに記録しません。ジャーナリングは、DB2 DataPropagator for iSeries の外側で開始する必要があります。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|--------------|---|
| TGTTYPE (続き) | <p>*POINTINTIME</p> <p>ターゲット表はポイント・イン・タイム・コピーです。ポイント・イン・タイム・コピーは、ソース表の内容の一部またはすべてと一致する内容を持つターゲット表であり、DB2 DataPropagator for iSeries システム列 (IBMSNAP_LOGMARKER) を保持します。この列は、特定の行がキャプチャー・コントロール・サーバーでいつ挿入または変更されたかを示します。</p> <p>*BASEAGR</p> <p>ターゲット表は基本集約コピーであり、これはソース表から集約された (算出された) データを含む、ターゲット表です。基本集約ターゲットのソース表は、ユーザーまたはポイント・イン・タイム表のどちらかでなければなりません。このターゲット表には、システム・タイム・スタンプ列の IBMSNAP_HLOGMARKER および IBMSNAP_LLOGMARKER が含まれます。</p> <p>*CHANGEAGR</p> <p>表は変更集約コピーであり、これは、変更データ (CD) 表の内容を基に集約された (算出された) データを含むターゲット表です。このターゲット表は、システム・タイム・スタンプ列 IBMSNAP_HLOGMARKER および IBMSNAP_LLOGMARKER を使用して作成されます。</p> <p>*CCD</p> <p>表は整合変更データ (CCD) 表であり、これは、変更データ (CD) 表と作業単位 (UOW) 表内のデータを結合したものから作成されたターゲット表です。CCD 表は、アプライ・プログラムにトランザクション整合性のあるデータを提供し、次の列を含む必要があります。</p> <ul style="list-style-type: none"> • IBMSNAP_INTENTSEQ • IBMSNAP_OPERATION • IBMSNAP_COMMITSEQ • IBMSNAP_LOGMARKER <p>*REPLICA</p> <p>ターゲット表はレプリカ表であり、これは update-anywhere レプリケーションにのみ使用されます。レプリカ・ターゲット表はマスター・ソース表から変更を受信し、またレプリカ・ターゲット表への変更は、マスター・ソース表に戻して伝搬されます。レプリカ表はソース表として自動的に登録されます。</p> |
| TIMING | <p>アプライ・プログラムがサブスクリプション・セットの処理に使用するタイミング (スケジューリング) のタイプを指定します。</p> <p>*INTERVAL (デフォルト)</p> <p>アプライ・プログラムは、サブスクリプション・セットを特定の時間インターバル (たとえば、1 日に 1 度) で処理します。</p> <p>*EVENT</p> <p>アプライ・プログラムは、特定のイベントが起こった時にサブスクリプション・セットを処理します。</p> <p>*BOTH</p> <p>アプライ・プログラムは、特定の時間インターバル、またはイベントが起こった時、のどちらでも最初に起こった時に処理します。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|---|
| EVENT | <p>イベントを指定します。入力するイベントは、サブスクリプション・イベント表 (IBMSNAP_SUBS_EVENT) のイベント名と一致する必要があります。</p> <p>*NONE (デフォルト) イベントは使用しません。</p> <p><i>event-name</i> IBMSNAP_SUBS_EVENT 表に記述されているイベントを表す、固有の文字ストリング。</p> |
| INTERVAL | <p>ターゲット・コピーのリフレッシュを行う時間インターバルを指定します。これは開始時刻から開始時刻までの時間インターバルであり、週、日、時間、および分で指定します。これは 2 つの部分からなる値です。最初の部分は数値、2 番目の部分は時間の単位です。</p> <p>*MIN 分</p> <p>*HOUR 時間</p> <p>*DAY 日</p> <p>*WEEK 週</p> <p>時間の単位と一緒に、数値の組み合わせを指定することができます。たとえば、((2 *WEEK) (3 *DAY) (35 *MIN)) は、2 週、3 日、および 35 分の時間インターバルを指定します。同じ時間単位で複数を指定すると、最後の指定が使用されます。</p> |
| ACTIVATE | <p>サブスクリプション・セットがアクティブかどうかを指定します。アプライ・プログラムは、このパラメーターが *YES でない場合、このサブスクリプション・セットを処理しません。</p> <p>*YES (デフォルト) サブスクリプション・セットはアクティブです。</p> <p>*NO サブスクリプション・セットはアクティブではありません。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| CRTTGTTBL | <p>ターゲット表 (またはビュー) を作成するかどうかを指定します。</p> <p>*YES (デフォルト) ターゲット表 (またはビュー) が存在しなければ、作成します。存在する場合は、既存の表またはビューがターゲットになり、この既存の表またはビューのフォーマットの CHKFMT パラメーターが *YES に設定されているかどうかをチェックします。 UNIQUE および KEYCOL パラメーターに指定された値を使用して、(そのような索引がまだ存在しなければ) ターゲット表に追加の索引が作成されます。既存のターゲット表に、追加索引の条件に違反するような行が含まれていると、コマンドは失敗します。</p> <p>*NO ターゲット表またはビューを作成しません。アプライ・プログラムを始動する前に、正しい属性を使用して表またはビューを作成する必要があります。</p> <p>表またはビューが存在する場合に CHKFMT を *YES にすると、 ADDDPRSUB コマンドは、既存の表のフォーマットが、設定されたサブスクリプション・セット定義と一致することを確認します。 CHKFMT を *NO にする場合は、ユーザーは既存の表のフォーマットがサブスクリプション・セット定義と一致することを確認しておく必要があります。</p> <p>重要: 表またはビューがすでに存在する場合、 DB2 DataPropagator for iSeries は、既存のオブジェクトへの変更を自動的にジャーナルに記録しません。ジャーナリングは、DB2 DataPropagator for iSeries の外側で開始する必要があります。</p> |
| CHKFMT | <p>DB2 DataPropagator for iSeries が、サブスクリプション・セットとターゲット表をチェックし、列が一致することを確認するかどうかを指定します。このパラメーターは、 CRTTGTTBL パラメーターが *YES の場合は無視され、また CRTTGTTBL パラメーターが *NO でターゲット表が存在しない場合も無視されます。</p> <p>*YES (デフォルト) DB2 DataPropagator for iSeries は、このサブスクリプション・セットに定義された列がターゲット表内の列と一致することを確認します。両者が一致しないと、このコマンドは失敗します。</p> <p>*NO DB2 DataPropagator for iSeries は、サブスクリプション・セットと既存のターゲット表間の相違を無視します。ユーザーはターゲット表がサブスクリプション・セットと互換性があることを確認する必要があります。</p> |
| CAPCTLLIB | <p>キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。これらのキャプチャー・コントロール表は、このサブスクリプション・セットのソースを処理します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。</p> <p><i>library-name</i> キャプチャー・コントロール表を含むライブラリーの名前。これは、ソース表が登録されたライブラリーです。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|---|
| TGTCCLIB | <p>ターゲット・コントロール・ライブラリーを指定します。</p> <p>*CAPCTLLIB (デフォルト) ターゲット・コントロール・ライブラリーは、キャプチャー・コントロール表が存在するライブラリーと同じです。</p> <p><i>library-name</i> ターゲット・コントロール表を含むライブラリーの名前。</p> <p>ターゲット表を別のサブスクリプション・セット (たとえば外部 CCD 表など) のソースとして使用する場合、このパラメーター値は、この表がソースとして使用される時のキャプチャー・スキーマです。</p> |
| FEDSVR | <p>このサブスクリプション・セットのソースが連合データベース・システムであるかどうかを指定します。</p> <p>*NONE (デフォルト) ソース・サーバーは連合データベース・システムではありません。</p> <p><i>server-name</i> このサブスクリプション・セットの連合データベース・システムの名前 (DB2 以外のリレーショナル・ソースの場合)。</p> |
| CMTCNT | <p>コミットメント・カウントを指定します。これは、アプライ・プログラムがトランザクションをいくつ処理したらコミットするかを示す数です。</p> <p>*DEFAULT (デフォルト) 使用する値をコマンドが決めます。TGTTYPE が *REPLICA の場合、CMTCNT はゼロ (0) です。TGTTYPE が *REPLICA 以外の場合、CMTCNT は NULL です。</p> <p>*NULL サブスクリプション・セットは読み取り専用です。アプライ・プログラムは、サブスクリプション・セット・メンバーの応答セットを 1 度に 1 メンバーずつフェッチし、すべてのデータの処理を終了した後、サブスクリプション・セット全体について 1 つのコミットを出します。</p> <p><i>num-transactions</i> いくつのトランザクション処理を処理したらアプライ・プログラムが変更をコミットするかを示す数を指定します。このパラメーターは、TGTTYPE パラメーターが *REPLICA の場合のみ有効です。</p> <p>重要: 値 1 は値 0 と同じです。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| TGTKEYCHG | <p>ターゲット表のターゲット・キー列の一部であるソース列に変更があった場合、アプライ・プログラムがその更新をどのように扱うかを指定します。このパラメーターは、ADDDPRREG コマンドの USEDELINS パラメーターと組み合わせて働きます。</p> <ul style="list-style-type: none"> • USEDELINS が YES で TGTKEYCHG が YES の場合、更新はできません。 • USEDELINS が YES で TGTKEYCHG が NO の場合、更新は削除と挿入の対になります。 • USEDELINS が NO で TGTKEYCHG が YES の場合、アプライ・プログラムは特別な論理を使用してこの条件を扱います。 • USEDELINS が NO で TGTKEYCHG が NO の場合、アプライ・プログラムは変更を通常の更新として処理します。 <p>*NO (デフォルト) ソース表に対する更新は、キャプチャー・プログラムによりステージ化され、アプライ・プログラムによりターゲット表に処理が行われます。</p> <p>*YES アプライ・プログラムは、ターゲット・キー列の変更前イメージに基づいてターゲット表を更新します。つまり、アプライ・プログラムは述部を新しい値ではなく、古い値に変更します。</p> |
| COLUMN | <p>ターゲット表に含める列を指定します。列名は修飾できません。列名は、ソース表の登録時に CAPCOL パラメーターに指定した列名のリストから選択してください。</p> <p>この表の登録時に IMAGE パラメーターを *BOTH に設定した場合は、変更前イメージ列名を指定することができます。変更前イメージ列名は、接頭部を持つオリジナルの列名です。この接頭部は、ADDDPRREG コマンドの PREFIX パラメーターに指定した文字です。</p> <p>*ALL (デフォルト) ソースに登録した列のすべてがターゲット表に含まれます。</p> <p>*NONE ソース表からの列は 1 つもターゲット表に含まれません。*NONE は、算出列だけをターゲット表に含めたい場合に使用します。この値は、CALCCOL パラメーターに合計関数が含まれているが、GROUP BY が実行されない、という場合に必要です。</p> <p><i>column-name</i> ターゲット表に含めたいソース列の名前を 300 個まで指定できます。列名はスペースで区切ります。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|--|
| UNIQUE | <p>ターゲット表が KEYCOL パラメーターで示されたユニーク・キーを持つかどうかを指定します。</p> <p>*YES (デフォルト)</p> <p>ターゲット表はキーごとに正味 1 つの変更をサポートします。つまり、キーに対していかに多くの変更がなされたとしても、そのキーについてはターゲット表に 1 つの行しか存在しないということです。</p> <p>この値は、表がデータの変更のヒストリーではなく、現行のデータを含むことを指定します。コンデンス表には、1 つの主キー値に対して複数の行が含まれることはなく、リフレッシュ用の最新情報を提供するために使用できます。</p> <p>*NO</p> <p>ターゲット表はキーごとに複数の変更をサポートします。変更はターゲット表に付加されます。</p> <p>この値は、表が現行のデータではなく、変更のヒストリーを含むことを指定します。非コンデンス表には、それぞれのキー値に対して複数の行が含まれ、データの変更ヒストリーを提供するために使用できます。ただし、非コンデンス表はリフレッシュ用の最新データは提供できません。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| KEYCOL | <p>ターゲット表のキーを記述する列を指定します。列名は修飾できません。</p> <p>*POINTINTIME、*REPLICA、および *USERCOPY ターゲット表 (TGTTYPE パラメーターで指定されている) の場合、ターゲット表に 1 つまたは複数の列をターゲット・キーとして指定する必要があります。アプライ・プログラムはこのターゲット・キーを使用して、変更キャプチャー・レプリケーション中に、変更された個々の固有な行を識別します。</p> <p>*SRCTBL (デフォルト)</p> <p>ターゲット表のキー列は、ソース表のキー列と同じです。 ADDDPRREG コマンドは、ソース表がキー付きの場合、ソース表に指定されたキーを使用します。次のキー列が使用されます。</p> <ul style="list-style-type: none"> 物理ファイル作成コマンド (CRTPF) を使用して表を作成した時に、DDS を使用して定義したキー列 CREATE TABLE および ALTER TABLE SQL ステートメントを使用して定義した、主キーおよびユニーク・キー CREATE INDEX SQL ステートメントを使用して定義したユニーク・キー <p>1 つの列を、キーとして、異なる順序付けで複数回使用すると、ターゲット表のキーは昇順で定義されます。</p> <p>*RRN</p> <p>ターゲット表のキー列は IBMQSQ_RRN 列です。ターゲット表は IBMQSQ_RRN 列を使用して作成され、この列がキーとして使用されます。アプライ・プログラムの実行時に、ソース表がユーザー表であり、ターゲット表がポイント・イン・タイムまたはユーザー・コピーの場合、ターゲット表の IBMQSQ_RRN 列が、ソース表内の関連するレコードの RRN (相対レコード番号) で更新されます。それ以外では、ターゲット表の IBMQSQ_RRN 列は、ソース表内の IBMQSQ_RRN 列の値で更新されます。</p> <p>*NONE</p> <p>ターゲット・コピーはターゲット・キーを含みません。ターゲット表のタイプが *POINTINTIME、*REPLICA、または *USERCOPY の場合、*NONE は指定できません。</p> <p><i>column-name</i></p> <p>ターゲット・キー列として使用したいターゲット列の名前。列名は 120 個まで指定できます。列名はスペースで区切ります。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|--|
| TGTCOL | <p>アプリ・プログラムがターゲット表内で更新するすべての列の新しい名前を指定します。これらの名前は、ソース表から取られた列名をオーバーライドします。列名は修飾できません。 COLUMN パラメーターに *NONE を指定した場合はこのパラメーターを使用しないでください。</p> <p>このパラメーターを使用して、ターゲット表の列により分かりやすい名前を付けることができます。それぞれのソース列の名前と、ターゲット表の対応する列の名前を指定します。</p> <p>*COLUMN (デフォルト) ターゲット列は、COLUMN パラメーターに指定した列と同じです。</p> <p><i>column-name</i> ターゲットではその名前を変更したい、ソース表からの列名。列名は 300 個まで指定できます。</p> <p><i>new-name</i> ターゲット列の新しい名前。新しい列名を 300 個まで指定できます。このパラメーターを使用しないと、ターゲット表の列名はソースの列名と同じになります。</p> |
| CALCCOL | <p>ターゲット表のユーザー定義の列または算出された列のリストを指定します。列名は修飾できません。それぞれの列名と式の対を括弧で囲みます。</p> <p>各 SQL 式には列名を指定する必要があります。 GROUP BY ステートメントのない SQL 式として列を定義する場合は、 COLUMN パラメーターを *NONE にする必要があります。</p> <p>*NONE (デフォルト) ターゲット表はユーザー定義の列または算出された列を含みません。</p> <p><i>column-name</i> ターゲット表のユーザー定義の列または算出された列の列名。列名は 100 個まで指定できます。</p> <p><i>expression</i> ターゲット表のユーザー定義の列または算出された列の式。 SQL 列の式は 100 個まで指定できます。</p> |
| ADDREG | <p>ターゲット表をソース表として自動的に登録するかどうかを指定します。このパラメーターは CCD ターゲット・タイプの表を登録する場合に使用します。</p> <p>*NO (デフォルト) ターゲット表はソース表として登録されません。 DB2 DataPropagator for iSeries は、ターゲット・タイプが *REPLICA の場合、このパラメーター値を無視します。レプリカ・ターゲット表は必ず、ソース表として自動的に登録されます。</p> <p>*YES ターゲット表はソース表として登録されます。ターゲット表をすでにユーザーが登録していると、このコマンドは失敗します。</p> <p>ターゲット表のタイプが *USERCOPY、*POINTINTIME、*BASEAGR、または *CHANGEAGR の場合、このパラメーターを *YES にしないでください。</p> <p>CRITGTTBL パラメーターを *NO にした場合、これをソースとして登録する前に、ターゲット表を作成する必要があります。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| ROWSLT | <p>SQL の WHERE 文節に入れる述部を指定します。アプライ・プログラムはこれらの述部を使用して、ソースの変更データ (CD) 表のどの行をターゲット表に適用するかを決めます。ソースの変更のサブセットだけをターゲット表に複製したい場合に、このパラメーターを使用します。</p> <p>*ALL (デフォルト) アプライ・プログラムは、CD 表内のすべての変更をターゲット表に適用します。</p> <p><i>WHERE-clause</i> アプライ・プログラムが CD 表からどの行をターゲット表に適用するかを指定する SQL の WHERE 文節です。WHERE キーワードはこのパラメーターで暗黙に想定されているので、WHERE キーワードを含めないでください。この WHERE 文節は、この文節を実行するデータ・サーバー上で有効なものでなければなりません。</p> <p>注: このパラメーターの WHERE 文節は、SQLBEFORE または SQLAFTER パラメーターに指定された WHERE 文節とはまったく関係ありません。</p> |
| MAXSYNCH | <p>最大の同期化の分数を指定します。このパラメーターは、時間しきい値の制限を指定し、サブスクリプションのサイクル中に、キャプチャー・プログラムおよびアプライ・プログラムが処理する変更データの量を規制するために使用されます。時間しきい値の制限は 2 つの部分からなる値で指定します。最初の部分は数値、2 番目の部分は時間の単位です。</p> <p>*MIN 分</p> <p>*HOUR 時間</p> <p>*DAY 日</p> <p>*WEEK 週</p> <p>時間の単位と一緒に、数値の組み合わせを指定することができます。たとえば、((1 *WEEK) (2 *DAY) (35 *MIN)) は、1 週、2 日、および 35 分の時間インターバルを指定します。同じ時間単位で複数指定すると、最後の指定が使用されます。</p> <p>デフォルトはゼロ (0) であり、変更データのすべてを適用することを示します。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| SQLBEFORE | <p>アプリ・プログラムがターゲット表をリフレッシュする前に実行する、SQL ステートメントを指定します。このパラメーターは以下の 3 つの要素からなります。</p> <p>要素 1: SQL コード</p> <p>*NONE (デフォルト) SQL ステートメントを指定しません。</p> <p><i>SQL-statement</i> 実行したい SQL ステートメント。SQL ステートメントの構文が正しいことを確認してください。DB2 DataPropagator for iSeries は構文の妥当性を検査しません。また、適切な SQL 命名規則を使用する必要があります。SQL ファイル参照は、システムの命名規則 (LIBRARY/FILE) ではなく、LIBRARY.FILE の形式でなければなりません。3 つまでの SQL ステートメントを指定できます。</p> <p>要素 2: 実行するサーバー</p> <p>*TGTSVR (デフォルト) SQL ステートメントは、ターゲット表が存在するターゲット・サーバーで実行されます。</p> <p>*SRCSVR SQL ステートメントは、ソース表が存在するキャプチャー・コントロール・サーバーで実行されます。</p> <p>要素 3: 許される SQLSTATE 値</p> <p>*NONE (デフォルト) SQLSTATE 値 00000 のみが正常と見なされます。</p> <p><i>SQL-states</i> 1 から 10 個までの許容される SQLSTATE 値のリスト。SQLSTATE 値はスペースで区切ります。SQLSTATE 値は、00000 から FFFFF の範囲の 5 桁の 16 進数です。</p> <p>SQL ステートメントは、SQLSTATE 値 00000 または、リストされた許容可能な SQLSTATE 値の 1 つで完了した場合に、正常終了となります。</p> |

表 31. ADDDPRSUB コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| SQLAFTER | <p>アプライ・プログラムがターゲット表をリフレッシュした後に実行する、SQL ステートメントを指定します。このパラメーターは以下の 3 つの要素からなります。</p> <p>要素 1: SQL コード</p> <p>*NONE (デフォルト) SQL ステートメントを指定しません。</p> <p><i>SQL-statement</i> 実行したい SQL ステートメント。SQL ステートメントの構文が正しいことを確認してください。DB2 DataPropagator for iSeries は構文の妥当性を検査しません。また、適切な SQL 命名規則を使用する必要があります。SQL ファイル参照は、システムの命名規則 (LIBRARY/FILE) ではなく、LIBRARY.FILE の形式でなければなりません。3 つまでの SQL ステートメントを指定できます。</p> <p>要素 2: 実行するサーバー</p> <p>*TGTSVR (デフォルト) SQL ステートメントは、ターゲット表が存在するターゲット・サーバーで実行されます。</p> <p>要素 3: 許される SQLSTATE 値</p> <p>*NONE (デフォルト) SQLSTATE 値 00000 のみが正常と見なされます。</p> <p><i>SQL-states</i> 1 から 10 個までの許容される SQLSTATE 値のリスト。SQLSTATE 値はスペースで区切ります。SQLSTATE 値は、00000 から FFFFF の範囲の 5 桁の 16 進数です。</p> <p>SQL ステートメントは、SQLSTATE 値 00000 または、リストされた許容可能な SQLSTATE 値の 1 つで完了した場合に、正常終了となります。</p> |

ADDDPRSUB の例

次の例は、ADDDPRSUB コマンドの使用法を示しています。

例 1

AQHR アプライ修飾子の下に **SETHR** という名前のサブスクリプション・セットを作成します。

```
ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE)
TGTTBL(TGTLIB/TGTEMPL)
```

このサブスクリプション・セットには 1 つのサブスクリプション・セット・メンバーが含まれ、HR ライブラリー下の **EMPLOYEE** という名前の登録済みソース表から、TGTLIB ライブラリー下の **TGTEMPL** という名前のターゲット表にデータを複製します。

ADDDPRSUB

例 2

2 つだけの列 (EMPNO (キー) および NAME) を持つ、SETHR という名前のサブスクリプション・セットを、EMPLOYEE という名前の登録済みソース表から作成し、これらの列を TGTEMPL という名前の既存のターゲット表に複製します。

```
ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE)
  TGTTBL(TGTLIB/TGTEMPL) CRTTGTTBL(*NO) COLUMN(EMPNO NAME) KEYCOL(EMPNO)
```

例 3

SETHR という名前のサブスクリプション・セットを、EMPLOYEE という名前の登録済みソース表からのデータを使用して作成し、このデータを TGTREPL という名前のレプリカ・タイプのターゲット表に複製します。

```
ADDDPRSUB APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/EMPLOYEE)
  TGTTBL(TGTLIB/TGTREPL) TGTTYPE(*REPLICA)
```

例 4

NOMEM という名前のサブスクリプション・セットを、サブスクリプション・セット・メンバーなしで作成します。

```
ADDDPRSUB APYQUAL(AQHR) SETNAME(NOMEM) SRCTBL(*NONE) TGTTBL(*NONE)
```

関連作業:

- 69 ページの 『第 4 章 ソースのサブスクライブ』

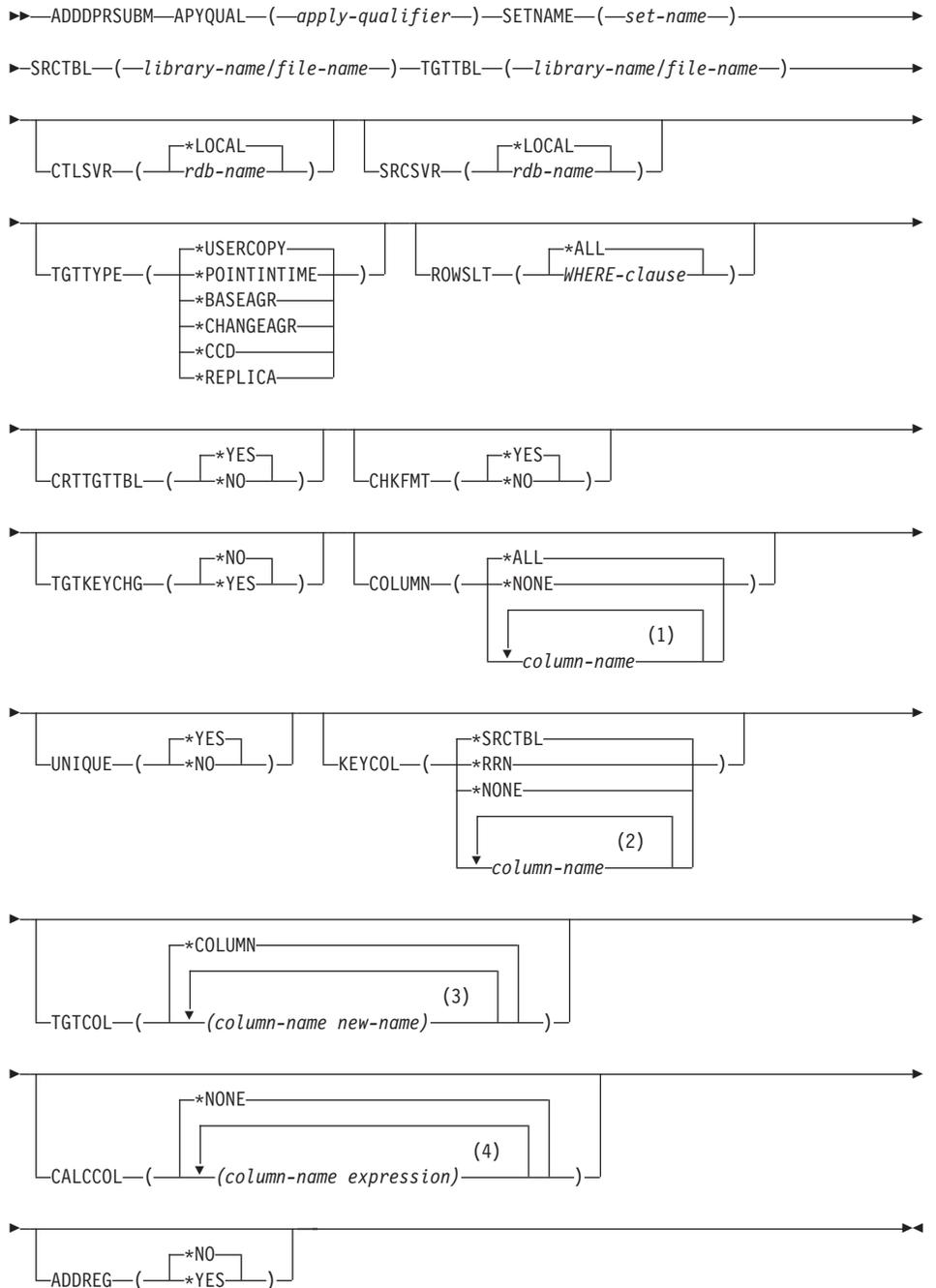
ADDDPRSUBM: DPR サブスクリプション・セットのメンバーの追加 (OS/400)

既存のサブスクリプション・セットにメンバーを追加するには、DPR サブスクリプション・セット・メンバーの追加 (ADDDPRSUBM) コマンドを使用します。サブスクリプション・セットは、ADDDPRSUB コマンドを使用して、UNIX、Windows、または z/OS 上のシステム・コマンドを使用して、またはレプリケーション・センターから作成することができます。サブスクリプション・セット内のすべてのソース表は、すでにジャーナルに記録済みであり、登録済みでなければならず、その後でないこのコマンドは使用できません。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

ADDDPRSUBM コマンドを使用してサブスクリプション・セットにメンバーを追加するための構文



注:

- 1 列名は 300 個まで指定できます。

ADDDPRSUBM

- 2 列名は 120 個まで指定できます。
- 3 列名は 300 個まで指定できます。
- 4 100 個までの列名および式を指定できます。

表 32. *ADDDPRSUBM* コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| APYQUAL | どのアプライ・プログラムがこのサブスクリプション・セットを処理するかを示すアプライ修飾子を指定します。アプライ修飾子の下のサブスクリプション・セットは別のジョブで実行されます。このパラメーターは必須です。 <i>apply-qualifier</i> アプライ修飾子の名前。 |
| SETNAME | サブスクリプション・セットの名前を指定します。このパラメーターは必須です。 <i>set-name</i> サブスクリプション・セットの名前。入力するサブスクリプション・セット名は、指定されたアプライ修飾子について固有でなければなりません。固有でない場合、 ADDDPRSUBM コマンドはエラーになります。アプライ・プログラムはターゲット表のセットをグループとして扱うので、何らかの理由で 1 つのターゲット表が失敗すると、そのセット全体が失敗します。 |
| SRCTBL | このサブスクリプション・セット・メンバーのソースである表の名前を指定します。この表をサブスクリプション・セットのメンバーにするには、この表をキャプチャー・コントロール・サーバーにあらかじめ登録しておく必要があります。このパラメーターは必須です。 <i>library-name/file-name</i> ソース表の修飾名。 |
| TGTTBL | このサブスクリプション・セット・メンバーのターゲット表の名前を指定します。 CRTTGTTBL パラメーターを *YES に設定し、ターゲット表が存在しないと、ターゲット表は自動的に作成されます。このパラメーターは必須です。 <i>library-name/file-name</i> ターゲット表の修飾名。 |
| CTLSVR | アプライ・コントロール表を含むシステムのリレーショナル・データベース名を指定します。 *LOCAL (デフォルト) アプライ・コントロール表はローカルに存在します (ADDDPRSUBM コマンドを実行するマシン上)。 <i>rdb-name</i> アプライ・コントロール表が置かれているリレーショナル・データベースの名前。RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。 |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| SRCSV | <p>キャプチャー・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) ソース表はローカル・マシンに登録されています (ADDDPRSUBM コマンドを実行するマシン)。</p> <p><i>rdb-name</i> キャプチャー・コントロール表が存在するリレーショナル・データベースの名前。 RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> |
| TGTYPE | <p>ターゲット表のタイプを指定します。これらは、ターゲット表の内容を記述する DB2 レプリケーション用語です。これらのいずれかのタイプとしてターゲット表を作成した後、DPR 登録の追加 (ADDDPRREG) コマンドの SRCTBL パラメーターにこのパラメーター値を使用し、このターゲット表をソース表として登録することができます。</p> <p>*USERCOPY (デフォルト) ターゲット表はユーザー・コピーであり、これはソース表の内容のすべてまたは一部と一致する内容を持つ、ターゲット表です。ユーザー・コピーはポイント・イン・タイムのコピーのように扱われますが、ポイント・イン・タイムのターゲット表に存在する、DB2 DataPropagator for iSeries システム列は 1 つも含まれていません。</p> <p>この値は、KEYCOL パラメーターに *RRN の値が指定されている場合は無効です。</p> <p>SRCTBL パラメーターで指定した表は、ユーザー・データベース、ポイント・イン・タイム・コピー、または整合変更データ (CCD) のいずれかでなければなりません。</p> <p>重要: ターゲット表がすでに存在する場合、DB2 DataPropagator for iSeries は、これに対する変更を自動的にジャーナルに記録しません。ジャーナリングは、DB2 DataPropagator for iSeries の外側で開始する必要があります。</p> |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-------------|--|
| TGTYPE (続き) | <p>*POINTINTIME</p> <p>ターゲット表はポイント・イン・タイム・コピーです。ポイント・イン・タイム・コピーは、ソース表の内容の一部またはすべてと一致する内容を持つターゲット表であり、DB2 DataPropagator for iSeries システム列 (IBMSNAP_LOGMARKER) を保持します。この列は、特定の行がキャプチャー・コントロール・サーバーでいつ挿入または変更されたかを示します。</p> <p>*BASEAGR</p> <p>ターゲット表は基本集約コピーであり、これはソース表から集約された (算出された) データを含む、ターゲット表です。基本集約ターゲットのソース表は、ユーザー表またはポイント・イン・タイム表のどちらかでなければなりません。このターゲット表には、システム・タイム・スタンプ列の IBMSNAP_HLOGMARKER および IBMSNAP_LLOGMARKER が含まれます。</p> <p>*CHANGEAGR</p> <p>表は変更集約コピーであり、これは、変更データ (CD) 表の内容を基に集約された (算出された) データを含むターゲット表です。このターゲット表は、システム・タイム・スタンプ列 IBMSNAP_HLOGMARKER および IBMSNAP_LLOGMARKER を使用して作成されます。</p> <p>*CCD</p> <p>表は整合変更データ (CCD) 表であり、これは、変更データ (CD) 表と作業単位 (UOW) 表内のデータを結合したものから作成されたターゲット表です。CCD 表は、アプライ・プログラムにトランザクション整合性のあるデータを提供し、次の列を含む必要があります。</p> <ul style="list-style-type: none"> • IBMSNAP_INTENTSEQ • IBMSNAP_OPERATION • IBMSNAP_COMMITSEQ • IBMSNAP_LOGMARKER <p>*REPLICA</p> <p>ターゲット表はレプリカ表であり、これは update-anywhere レプリケーションにのみ使用されます。レプリカ・ターゲット表はマスター・ソース表から変更を受信し、またレプリカ・ターゲット表への変更は、マスター・ソース表に戻して伝搬されます。レプリカ表はソース表として自動的に登録されます。</p> |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|---|
| ROWSLT | <p>SQL の WHERE 文節に入れる述部を指定します。アプライ・プログラムはこれらの述部を使用して、ソースの変更データ (CD) 表のどの行をターゲット表に適用するかを決めます。ソースの変更のサブセットだけをターゲット表に複製したい場合に、このパラメーターを使用します。</p> <p>*ALL (デフォルト) アプライ・プログラムは、CD 表内のすべての変更をターゲット表に適用します。</p> <p><i>WHERE-clause</i> アプライ・プログラムが CD 表からどの行をターゲット表に適用するかを指定する SQL の WHERE 文節です。WHERE キーワードはこのパラメーターで暗黙に想定されているので、WHERE キーワードを含めないでください。この WHERE 文節は、この文節を実行するデータ・サーバー上で有効なものでなければなりません。</p> <p>注: このパラメーターの WHERE 文節は、SQLBEFORE または SQLAFTER パラメーターに指定された WHERE 文節とはまったく関係ありません。</p> |
| CRTGTGTL | <p>ターゲット表 (またはビュー) を作成するかどうかを指定します。</p> <p>*YES (デフォルト) ターゲット表 (またはビュー) が存在しないければ、作成します。存在する場合は、既存の表またはビューがターゲットになり、この既存の表またはビューのフォーマットの CHKFMT パラメーターが *YES に設定されているかどうかをチェックします。UNIQUE および KEYCOL パラメーターに指定された値を使用して、(そのような索引がまだ存在しなければ) ターゲット表に追加の索引が作成されます。既存のターゲット表に、追加索引の条件に違反するような行が含まれていると、コマンドは失敗します。</p> <p>*NO ターゲット表またはビューを作成しません。アプライ・プログラムを始動する前に、正しい属性を使用して表またはビューを作成する必要があります。</p> <p>表またはビューが存在する場合に CHKFMT を *YES に設定すると、ADDDPRSUBM コマンドは、既存の表のフォーマットが、設定されたサブスクリプション・セット定義と一致することを確認します。CHKFMT を *NO にする場合は、ユーザーは既存の表のフォーマットがサブスクリプション・セット定義と一致することを確認しておく必要があります。</p> <p>重要: 表またはビューがすでに存在する場合、DB2 DataPropagator for iSeries は、既存のオブジェクトへの変更を自動的にジャーナルに記録しません。ジャーナリングは、DB2 DataPropagator for iSeries の外側で開始する必要があります。</p> |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| CHKFMT | <p>DB2 DataPropagator for iSeries が、サブスクリプション・セット・メンバーの定義を既存のターゲット表に照らし合わせ、列が一致することをチェックするかどうかを指定します。このパラメーターは、 CRTTGTTBL パラメーターが *YES の場合は無視され、また CRTTGTTBL パラメーターが *NO でターゲット表が存在しない場合も無視されません。</p> <p>*YES (デフォルト)</p> <p>DB2 DataPropagator for iSeries は、このサブスクリプション・セット・メンバーに定義された列がターゲット表内の列と一致することを検査します。両者が一致しないと、このコマンドは失敗します。</p> <p>*NO</p> <p>DB2 DataPropagator for iSeries は、サブスクリプション・セット・メンバーと既存のターゲット表間の相違を無視します。ユーザーはターゲット表がサブスクリプション・セット・メンバーと互換性があることを確認する必要があります。</p> |
| TGTKEYCHG | <p>ターゲット表のターゲット・キー列の一部であるソース列に変更があった場合、アプライ・プログラムがその更新をどのように扱うかを指定します。このパラメーターは、ADDDPRREG コマンドの USEDELINS パラメーターと組み合わせて働きます。</p> <ul style="list-style-type: none"> • USEDELINS が YES で TGTKEYCHG が YES の場合、更新はできません。 • USEDELINS が YES で TGTKEYCHG が NO の場合、更新は削除と挿入の対になります。 • USEDELINS が NO で TGTKEYCHG が YES の場合、アプライ・プログラムは特別な論理を使用してこの条件を扱います。 • USEDELINS が NO で TGTKEYCHG が NO の場合、アプライ・プログラムは変更を通常の更新として処理します。 <p>*NO (デフォルト)</p> <p>ソース表に対する更新は、キャプチャー・プログラムによりステージ化され、アプライ・プログラムによりターゲット表に処理が行われます。</p> <p>*YES</p> <p>アプライ・プログラムは、ターゲット・キー列の変更前イメージに基づいてターゲット表を更新します。つまり、アプライ・プログラムは述部を新しい値ではなく、古い値に変更します。</p> |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| COLUMN | <p>ターゲット表に含める列を指定します。列名は修飾できません。列名は、ソース表の登録時に CAPCOL パラメーターに指定した列名のリストから選択してください。</p> <p>この表の登録時に IMAGE パラメーターを *BOTH に設定した場合は、変更前イメージ列名を指定することができます。変更前イメージ列名は、接頭部を持つオリジナルの列名です。この接頭部は、ADDDPRREG コマンドの PREFIX パラメーターに指定した文字です。</p> <p>*ALL (デフォルト) ソースに登録した列のすべてがターゲット表に含まれます。</p> <p>*NONE ソース表からの列は 1 つもターゲット表に含まれません。 *NONE は、算出列だけをターゲット表に含めたい場合に使用します。この値は、CALCCOL パラメーターに合計関数が含まれているが、 GROUP BY が実行されない、という場合に必要です。</p> <p><i>column-name</i> ターゲット表に含めたいソース列の名前を 300 個まで指定できます。列名はスペースで区切ります。</p> |
| UNIQUE | <p>ターゲット表が KEYCOL パラメーターで示されたユニーク・キーを持つかどうかを指定します。</p> <p>*YES (デフォルト) ターゲット表はキーごとに正味 1 つの変更をサポートします。つまり、キーに対していかに多くの変更がなされたとしても、そのキーについてはターゲット表に 1 つの行しか存在しないということです。</p> <p>この値は、表がデータの変更の履歴ではなく、現行のデータを含むことを指定します。コンデンス表には、1 つの主キー値に対して複数の行が含まれることはなく、リフレッシュ用の最新情報を提供するために使用できます。</p> <p>*NO ターゲット表はキーごとに複数の変更をサポートします。変更はターゲット表に付加されます。</p> <p>この値は、表が現行のデータではなく、変更の履歴を含むことを指定します。非コンデンス表には、それぞれのキー値に対して複数の行が含まれ、データの変更履歴を提供するために使用できます。ただし、非コンデンス表はリフレッシュ用の最新データは提供できません。</p> |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| KEYCOL | <p>ターゲット表のキーを記述する列を指定します。列名は修飾できません。</p> <p>*POINTINTIME、*REPLICA、および *USERCOPY ターゲット表 (TGTTYPE パラメーターで指定されている) の場合、ターゲット表に 1 つまたは複数の列をターゲット・キーとして指定する必要があります。アプライン・プログラムはこのターゲット・キーを使用して、変更キャプチャー・レプリケーション中に、変更された個々の固有な行を識別します。</p> <p>*SRCTBL (デフォルト)</p> <p>ターゲット表のキー列は、ソース表のキー列と同じです。 ADDDPRREG コマンドは、ソース表がキー付きの場合、ソース表に指定されたキーを使用します。次のキー列が使用されます。</p> <ul style="list-style-type: none"> 物理ファイル作成コマンド (CRTPF) を使用して表を作成した時に、DDS を使用して定義したキー列 CREATE TABLE および ALTER TABLE SQL ステートメントを使用して定義した、主キーおよびユニーク・キー CREATE INDEX SQL ステートメントを使用して定義したユニーク・キー <p>1 つの列を、キーとして、異なる順序付けで複数回使用すると、ターゲット表のキーは昇順で定義されます。</p> <p>*RRN</p> <p>ターゲット表のキー列は IBMQSQ_RRN 列です。ターゲット表は IBMQSQ_RRN 列を使用して作成され、この列がキーとして使用されます。アプライン・プログラムの実行時に、ソース表がユーザー表であり、ターゲット表がポイント・イン・タイムまたはユーザー・コピーの場合、ターゲット表の IBMQSQ_RRN 列が、ソース表内の関連するレコードの RRN (相対レコード番号) で更新されます。それ以外では、ターゲット表の IBMQSQ_RRN 列は、ソース表内の IBMQSQ_RRN 列の値で更新されます。</p> <p>*NONE</p> <p>ターゲット・コピーはターゲット・キーを含みません。ターゲット表のタイプが *POINTINTIME、*REPLICA、または *USERCOPY の場合、*NONE は指定できません。</p> <p><i>column-name</i></p> <p>ターゲット・キー列として使用したいターゲット列の名前。列名は 120 個まで指定できます。列名はスペースで区切ります。</p> |

表 32. ADDDPRSUBM コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| TGTCOL | <p>アプライ・プログラムがターゲット表内で更新するすべての列の新しい名前を指定します。これらの名前は、ソース表から取られた列名をオーバーライドします。列名は修飾できません。COLUMN パラメーターに *NONE を指定した場合はこのパラメーターを使用しないでください。</p> <p>このパラメーターを使用して、ターゲット表の列により分かりやすい名前を付けることができます。それぞれのソース列の名前と、ターゲット表の対応する列の名前を指定します。</p> <p>*COLUMN (デフォルト) ターゲット列は、COLUMN パラメーターに指定した列と同じです。</p> <p><i>column-name</i> ターゲットではその名前を変更したい、ソース表からの列名。列名は 300 個まで指定できます。</p> <p><i>new-name</i> ターゲット列の新しい名前。新しい列名を 300 個まで指定できます。このパラメーターを使用しないと、ターゲット表の列名はソースの列名と同じになります。</p> |
| CALCCOL | <p>ターゲット表のユーザー定義の列または算出された列のリストを指定します。列名は修飾できません。それぞれの列名と式の対を括弧で囲みます。</p> <p>各 SQL 式には列名を指定する必要があります。GROUP BY ステートメントのない SQL 式として列を定義する場合は、COLUMN パラメーターを *NONE にする必要があります。</p> <p>*NONE (デフォルト) ターゲット表はユーザー定義の列または算出された列を含みません。</p> <p><i>column-name</i> ターゲット表のユーザー定義の列または算出された列の列名。列名は 100 個まで指定できます。</p> <p><i>expression</i> ターゲット表のユーザー定義の列または算出された列の式。SQL 列の式は 100 個まで指定できます。</p> |
| ADDREG | <p>ターゲット表をソース表として自動的に登録するかどうかを指定します。このパラメーターは CCD ターゲット・タイプの表を登録する場合に使用します。</p> <p>*NO (デフォルト) ターゲット表はソース表として登録されません。DB2 DataPropagator for iSeries は、ターゲット・タイプが *REPLICA の場合、このパラメーター値を無視します。レプリカ・ターゲット表は必ず、ソース表として自動的に登録されます。</p> <p>*YES ターゲット表はソース表として登録されます。ターゲット表をすでにユーザーが登録していると、このコマンドは失敗します。</p> <p>ターゲット表のタイプが *USERCOPY、*POINTINTIME、*BASEAGR、または *CHANGEAGR の場合、このパラメーターを *YES にしないでください。</p> <p>CRRTGTTBL パラメーターを *NO にした場合、これをソースとして登録する前に、ターゲット表を作成する必要があります。</p> |

ADDDPRSUBM の例

次の例は、**ADDDPRSUBM** コマンドの使用法を示しています。

例 1

サブスクリプション・セット・メンバーを AQHR アプライ修飾子の下の SETHR という名前のサブスクリプション・セットに追加します。

```
ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTDTAX) TGTTBL(TGTHR/TGTTAX)
```

例 2

2 つだけの列 (AMOUNT および NAME) を持つサブスクリプション・セット・メンバーを、YTDTAX という名前の登録済みソース表から追加し、これらの列を TGTTAX という名前の既存のターゲット表に複製します。

```
ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTDTAX) TGTTBL(TGTLIB/TGTTAX)  
CRTTGTTBL(*NO) COLUMN(AMOUNT NAME) CHKfmt(*YES)
```

このコマンドは、このサブスクリプション・セット・メンバーに定義された AMOUNT 列と NAME 列がターゲット表内の列と一致することを検査します。

例 3

サブスクリプション・セット・メンバーを SETHR という名前のサブスクリプション・セットに追加し、このデータを TGTYPD という名前の整合変更データ・ターゲット表に複製します。

```
ADDDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) SRCTBL(HR/YTDTAX) TGTTBL(TGTLIB/TGTYPD)  
TGTTYPE(*CCD) ADDREG (*YES)
```

このコマンドは、ターゲット表を DB2 DataPropagator for iSeries のソース表として登録します。

関連作業:

- 69 ページの 『第 4 章 ソースのサブスクライブ』

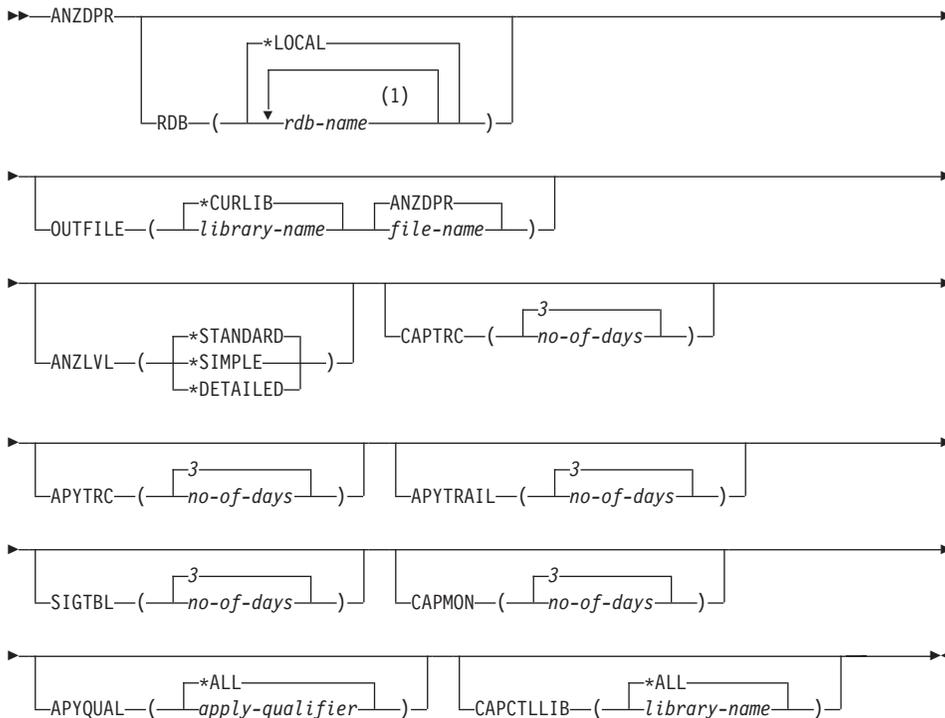
ANZDPR: アナライザーの操作 (OS/400)

キャプチャー・プログラムまたはアプライ・プログラムの障害の分析、レプリケーション構成のセットアップの検証、または問題診断およびパフォーマンス・チューニングの情報を入手するには、DPR の分析 (**ANZDPR**) コマンドを使用します。このコマンドは、レプリケーション構成をセットアップしてから実行してください。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

ANZDPR コマンドを使用してアナライザーを操作する方法



注:

- 1 最大 10 のデータベースを指定できます。

表 33. ANZDPR コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------|--|
| RDB | 分析対象のデータベースを指定します。 *LOCAL (デフォルト) ローカル・システム上のデータベース。 <i>rdb-name</i> データベースを示す、RDB ディレクトリー・エントリー名。 最大 10 のデータベースを入力できます。ローカル・システム上のデータベースを含めて複数のデータベースを分析したい場合は、リストの最初のエントリーを *LOCAL にしてください。また、現行システムからこれらすべてのデータベースに接続できることを確認してください。 |

表 33. ANZDPR コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| OUTFILE | <p>アナライザー出力の保管に使用されるライブラリーとファイル名を指定します。このコマンドは、出力を HTML ファイルに書き込みます。</p> <p>*CURLIB (デフォルト) 現行ライブラリー。</p> <p><i>library-name</i> ライブラリーの名前。</p> <p>ANZDPR (デフォルト) 出力は、ANZDPR という名前の HTML ファイルに書き込まれます。</p> <p><i>file-name</i> HTML 出力ファイルの名前。</p> <p>ファイル名がすでに存在する場合、ファイルは上書きされます。ファイル名が存在しない場合は、RCDLEN(512) および SIZE(*NOMAX) という属性のファイルがコマンドにより作成されます。</p> |
| ANZLVL | <p>報告される分析のレベルを指定します。分析のレベルは次のとおりです。</p> <p>*STANDARD (デフォルト) コントロール表の内容と、キャプチャー・プログラムおよびアプライ・プログラムの状況情報を含むレポートが生成されます。</p> <p>*SIMPLE 標準レポートで情報を生成しますが、サブカラムの詳細は含まれません。システム・リソースの使用量を抑えて小さなレポートを生成する場合は、このオプションを使用してください。</p> <p>*DETAILED 最大限に詳細な分析を含むレポートを生成します。詳細レポートには、サブスクリプション・セット情報に加えて、標準レポートの情報が含まれます。</p> |
| CAPTRC | <p>キャプチャー・トレース (IBMSNAP_CAPTRACE) 表から報告されるエンタリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 です。</p> <p><i>no-of-days</i> 報告される日数。</p> |
| APYTRC | <p>アプライ・トレース (IBMSNAP_APPLYTRACE) 表から報告されるエンタリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 です。</p> <p><i>no-of-days</i> 報告される日数。</p> |
| APYTRAIL | <p>アプライ・トレール (IBMSNAP_APPLYTRAIL) 表から報告されるエンタリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 です。</p> <p><i>no-of-days</i> 報告される日数。</p> |
| SIGTBL | <p>シグナル (IBMSNAP_SIGNAL) 表から報告されるエンタリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 です。</p> <p><i>no-of-days</i> 報告される日数。</p> |

表 33. ANZDPR コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| CAPMON | <p>キャプチャー・モニター (IBMSNAP_CAPMON) 表から報告されるエントリーの日付範囲 (0 から 30 日) を指定します。デフォルトは 3 です。</p> <p><i>no-of-days</i> 報告される日数。</p> |
| APYQUAL | <p>分析対象のアプライ修飾子を指定します。</p> <p>*ALL (デフォルト) すべてのアプライ修飾子が分析されます。</p> <p><i>apply-qualifier</i> 分析対象のアプライ修飾子の名前。最大 10 のアプライ修飾子を入力できます。</p> |
| CAPCTLLIB | <p>分析対象のキャプチャー・コントロール・ライブラリーの名前である、キャプチャー・スキーマを指定します。特定のキャプチャー・コントロール・ライブラリーを分析するか、デフォルトの *ALL を選択してすべてのキャプチャー・コントロール・ライブラリーを分析できます。</p> <p>*ALL (デフォルト) すべてのキャプチャー・コントロール・ライブラリーが分析されます。</p> <p><i>library-name</i> 分析対象の、特定のキャプチャー・コントロール・ライブラリーの名前。</p> |

ANZDPR の例

次の例は、**ANZDPR** コマンドの使用方法を示しています。

例 1

標準レベルの分析を使用して、ローカル・データベースと、**RMTRDB1** という名前のリモート・データベースの両方に対してアナライザーを実行するには、次のようにします。

```
ANZDPR RDB(*LOCAL RMTRDB1) OUTFILE(MYLIB/ANZDPR) ANZLVL(*STANDARD) CAPTRC(1)
  APYTRC(1) APYTRAIL(1) SIGTBL(1) CAPMON(1) APYQUAL(*ALL)
```

この例では、すべてのアプライ修飾子に関して、**IBMSNAP_CAPTRACE**、**IBMSNAP_APPLYTRACE**、**IBMSNAP_APPLYTRAIL**、**IBMSNAP_SIGNAL**、および **IBMSNAP_CAPMON** 表から 1 日分のエントリーが生成され、**MYLIB** という名前のライブラリー内の **ANZDPR** という名前の **HTML** ファイルに出力が書き込まれます。

例 2

すべてのデフォルト値を使用してアナライザーを実行するには、次のようにします。

```
ANZDPR
```

関連資料:

- 325 ページの 『asnanalyze: アナライザーの操作 (UNIX および Windows)』

CHGDPRCAPA: DPR キャプチャー属性の変更 (OS/400)

DPR キャプチャー属性の変更 (**CHGDPRCAPA**) コマンドは、キャプチャー・プログラムにより使用され、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表に保管されるグローバル操作パラメーターを変更するために使用されます。これらのパラメーターに対して行われた変更は、次のアクションのいずれかが実行されるまでは有効になりません。

- **INZDPRCAP** コマンドの実行。
- キャプチャー・プログラムの終了と再始動。

実行中のキャプチャー・プログラムの動作を変更するには、426 ページの『OVRDPRCAPA: DPR キャプチャー属性のオーバーライド (OS/400)』を参照してください。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

CHGDPRCAPA コマンドを使用して **DPR** キャプチャー属性を変更する方法

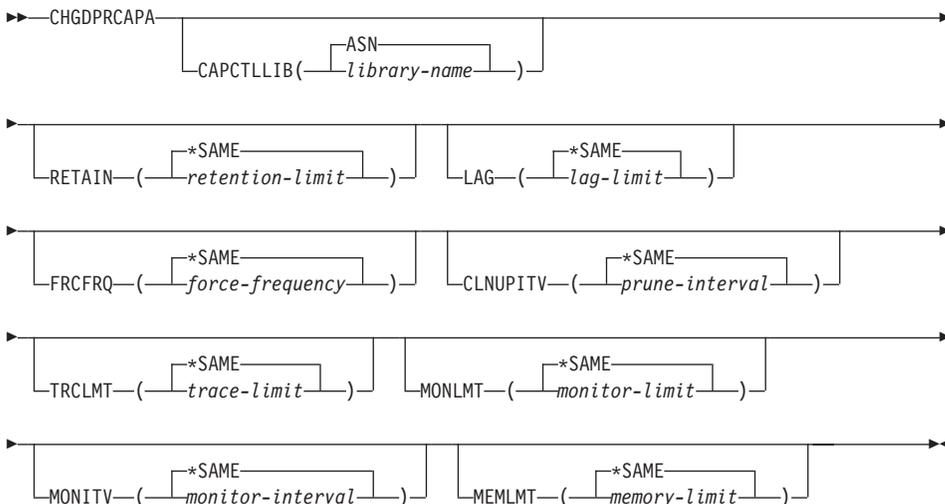


表 34. CHGDPRCAPA コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| CAPCTLLIB | <p>キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。</p> <p><i>library-name</i> キャプチャー・コントロール表を含むライブラリーの名前。</p> |
| RETAIN | <p>データが除去されずに、変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表に保存される分数である、新規の保持制限を指定します。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の RETENTION_LIMIT 列に保管されます。</p> <p>この値は、CLNUPITV パラメーターの値と共同で処理を行います。CLNUPITV の値に達すると、CD、UOW、IBMSNAP_SIGNAL、および IBMSNAP_AUTHTKN データのうちで、保持制限よりも古いデータが削除されます。</p> <p>表内のデータに矛盾が生じないように、データがこの RETAIN パラメーター値に達する前に変更情報がコピーされるようにアプライ・インターバルを設定してください。データの矛盾が発生した場合、アプライ・プログラムはフル・リフレッシュを実行します。</p> <p>デフォルトは 10,080 分 (7 日) です。最大値は、35000000 分です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>retention-limit</i> 新しい保持制限値。</p> |
| LAG | <p>キャプチャー・プログラムの処理が遅れても、その間は再始動が行われない分数である、新規の遅延限度を指定します。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の LAG_LIMIT 列に保管されます。</p> <p>遅延限度に達すると (つまり、ジャーナル・エントリーのタイム・スタンプが現在の時間から遅延限度を引いたものより古くなると)、キャプチャー・プログラムは、そのジャーナルで処理中の表のコールド・スタートを開始します。アプライ・プログラムはその後フル・リフレッシュを実行し、キャプチャー・プログラムに新しい開始点を提供します。</p> <p>デフォルトは 10,080 分 (7 日) です。最大値は、35000000 分です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>lag-limit</i> 新しい遅延限度値。</p> |

表 34. CHGDPRCAPA コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| FRCFRQ | <p>キャプチャー・プログラムが変更データ (CD) 表および作業単位 (UOW) 表に変更を書き込む頻度 (30 秒から 600 秒) を指定します。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の COMMIT_INTERVAL 列に保管されます。</p> <p>キャプチャー・プログラムは、バッファーがフルになるか、FRCFRQ 時間制限が満了するか、いずれか先に発生した時点で、これらの変更をアプライ・プログラムから使用可能にします。</p> <p>このパラメーターは、ソース表の変更率が少ないサーバー上で、アプライ・プログラムが変更をより早期に使用できるようにするために使用してください。FRCFRQ パラメーター値はグローバル値であり、すべての定義済みソース表で使用されます。FRCFRQ 値を低い数値に設定すると、システム・パフォーマンスが影響を受ける可能性があります。</p> <p>デフォルトは 30 秒です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>force-frequency</i> キャプチャー・プログラムが、アプライ・プログラムで変更を使用できるようにする前に、CD 表および UOW 表の変更をバッファー・スペース内に保持する秒数である、新規のコミット・インターバル値です。</p> |
| CLNUPITV | <p>キャプチャー・プログラムが変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表から古いレコードを除去するまでの最大時間 (時間単位) を指定します。</p> <p>このパラメーターは、RETAIN パラメーターと組み合わせて、CD、UOW、IBMSNAP_SIGNAL、および IBMSNAP_AUTHTKN 表のブルーニング、MONLMT パラメーターと組み合わせて、IBMSNAP_CAPMON 表のブルーニング、TRCLMT パラメーターと組み合わせて、IBMSNAP_CAPTRACE 表のブルーニングをコントロールします。(キャプチャー・プログラムの RETAIN、MONLMT、および TRCLMT パラメーターを設定するには、STRDPRCAP コマンドを使用します。)</p> <p>このパラメーターの値は、自動的に時間から秒に変換され、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の PRUNE_INTERVAL 列に保管されます。PRUNE_INTERVAL 列が (CHGDPRCAPA コマンドを使用せずに) 手動で変更された場合、F4 キーを使用してプロンプトを出すと、丸めによる変更が行われている場合があります。</p> <p>*SAME (デフォルト) このキャプチャー属性値は変更されません。</p> <p><i>prune-interval</i> 時間数 (1 から 100) で指定された、ブルーニング・インターバル。</p> |

表 34. CHGDPRCAPA コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|--|
| TRCLMT | <p>トレース限度を指定します (分単位)。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の TRACE_LIMIT 列に保管されます。</p> <p>キャプチャー・プログラムは、トレース限度よりも古い IBMSNAP_CAPTRACE 行を削除します。デフォルトは 10,080 分 (トレース・エントリーが 7 日間) です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>trace-limit</i> ブルーニング後に、トレース・データが IBMSNAP_CAPTRACE 表に保持される分数。</p> |
| MONLMT | <p>モニター限度を指定します (分単位)。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の MONITOR_LIMIT 列に保管されます。</p> <p>キャプチャー・プログラムは、モニター限度よりも古い IBMSNAP_CAPMON 行を削除します。</p> <p>デフォルトは 10,080 分 (モニター・エントリーが 7 日間) です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>monitor-limit</i> ブルーニング後に、モニター・データが IBMSNAP_CAPMON 表に保持される分数。</p> |
| MONITV | <p>キャプチャー・プログラムが、キャプチャー・モニター (IBMSNAP_CAPMON) 表に行を挿入する頻度 (秒単位) を指定します。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の MONITOR_INTERVAL 列に保管されます。</p> <p>デフォルトは 300 秒 (5 分) です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>monitor-interval</i> IBMSNAP_CAPMON 表への行挿入の間隔の秒数。モニター・インターバルは、少なくとも 120 秒 (2 分) あげる必要があります。120 よりも小さい数値がユーザーから指定された場合、このコマンドは自動的にパラメーター値を 120 に設定します。</p> |
| MEMLMT | <p>キャプチャー・ジャーナル・ジョブが利用できるメモリーの最大サイズ (MB 単位) を指定します。この値は、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の MEMORY_LIMIT 列に保管されます。</p> <p>デフォルトは 32 MB です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>memory-limit</i> メモリーの最大値の MB 数。</p> |

CHGDPRCAPA の例

次の例は、**CHGDPRCAPA** コマンドの使用方法を示しています。

例 1

キャプチャー・プログラムが **IBMSNAP_CAPMON** 表に行う行挿入の頻度を 6,000 秒 (100 分) 間隔に変更するには、次のようにします。

```
CHGDPRCAPA CAPCTLLIB(ASN) MONITV(6000)
```

この頻度の値は、デフォルト ASN ライブラリー内の **IBMSNAP_CAPPARMS** 表に保管されます。

例 2

LIB1 と呼ばれるキャプチャー・コントロール・ライブラリー内の **IBMSNAP_CAPPARMS** 表で保持制限、遅延限度、トレース限度、およびモニター限度を変更するには、次のようにします。

```
CHGDPRCAPA CAPCTLLIB(LIB1) RETAIN(6000) LAG(3000) TRCLMT(3000) MONLMT(6000)
```

例 3

キャプチャー・プログラムが **CD** 表および **UOW** 表に変更を書き込む頻度を示すコミット・インターバルを変更するには、次のようにします。

```
CHGDPRCAPA CAPCTLLIB(ASN) FRCFRQ(360)
```

関連作業:

- 129 ページの 『第 9 章 キャプチャー・プログラムの操作』

CRTDPRTBL: レプリケーション・コントロール表の作成 (OS/400)

レプリケーション・コントロール表が誤って削除または破壊された場合は、**DPR** 表の作成 (**CRTDPRTBL**) コマンドを使用して手動で表を作成します。

重要: **CRTDPRTBL** コマンドは、OS/400 のコントロール表を作成するために使用できる唯一のコマンドです。レプリケーション・センターを使用して、コントロール表を作成しないでください。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

CRTDPRTBL コマンドを使用してレプリケーション・コントロール表を作成する方法

ENDDPRAPY

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

ENDDPRAPY コマンドを使用してアプライ・プログラムを停止する方法

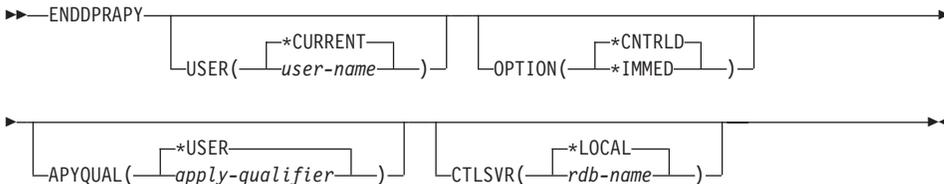


表 36. ENDDPRAPY コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|-------------------------|--|
| USER | このパラメーターは、 APYQUAL パラメーターの値が *USER である場合はアプライ・プログラムに関連付けられたアプライ修飾子を指定しますが、それ以外の場合は無視されます。 |
| *CURRENT (デフォルト) | 現行ジョブに関連付けられたユーザーのアプライ・プログラムです。 |
| <i>user-name</i> | 指定したユーザーのアプライ・プログラム。 |
| ENDDPRAPY | コマンドでプロンプトを出すと、F4 キーを押して、サブスクリプションを定義したユーザーのリストを表示することができます。 |

表 36. ENDDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|--|
| OPTION | <p>アプライ・プログラムを停止する方法を指定します。</p> <p>*CNTRLD (デフォルト)</p> <p>アプライ・プログラムは、停止前にすべてのタスクを完了します。アプライ・プログラムがサブスクリプション・セットを完了している場合、これらのタスクを終わらせるまでに、かなりの時間がかかる場合があります。</p> <p>*IMMED</p> <p>アプライ・プログラムは、ENDJOB OPTION(*IMMED) コマンドですべてのタスクを完了します。終結処理を行うことなく、このタスクはすぐに終了します。望ましくない結果を引き起こすことがあるため、コントロールされた終了が正常に行われなかった場合にのみ、このオプションを使用してください。(ENDDPRAPY コマンドの発行時に、アプライ・プログラムがスリープ状態でなければ、ターゲット表の目次を検査する必要があります。)</p> <p>アプライ・プログラムがターゲット表へのフル・リフレッシュを実行していた場合、その表がソース表の目次を使ってリフレッシュされる前にアプライ・プログラムが終了しているため、ターゲット表が空である可能性があります。ターゲット表が空の場合、このレプリケーション・ターゲットにフル・リフレッシュを行う必要があります。</p> <p>サブスクリプション・セットが使用中 (サブスクリプション・セット (IBMSNAP_SUBS_SET) にある STATUS 列の値が 1 である) と見なされている場合があります。この場合、この値を 0 または -1 に設定し直してください。こうすると、アプライ・プログラムはサブスクリプション・セットを再度実行することができます。</p> |
| APYQUAL | <p>アプライ・プログラムによって使われるアプライ修飾子を指定します。</p> <p>*USER (デフォルト)</p> <p>USER パラメーターに対するユーザー名をアプライ修飾子。</p> <p><i>apply-qualifier</i></p> <p>このアプライ・プログラムが実行するサブスクリプション・セットをグループ化するのに使われる名前。アプライ修飾子名として最大 18 文字を指定できます。この名前は、リレーショナル・データベース名と同じ命名規則に準拠します。実行中のサブスクリプションは、APPLY_QUAL 列にあるこの値を使用して、サブスクリプション・セット (IBMSNAP_SUBS_SET) 表にあるレコードが識別します。</p> <p>ENDDPRAPY コマンドでプロンプトを出すと、F4 キーを押して、既存のサブスクリプションを使用したアプライ修飾子のリストを表示することができます。</p> |

表 36. ENDDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| CTLSVR | <p>アプライ・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) アプライ・コントロール表はローカルに (ENDDPRAPY コマンドが実行されたマシンに) 存在します。</p> <p><i>rdb-name</i> アプライ・コントロール表が置かれているリレーショナル・データベースの名前。RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> <p>ENDDPRAPY コマンドでプロンプトを出すと、F4 キーを押して、RDB ディレクトリーにあるデータベースのリストから選択することができます。</p> |

使用上の注意

ENDDPRAPY コマンドは、**APYQUAL** および **CTLSVR** パラメーターの値を使用して、参照済みのアプライ・プログラムのジョブ名、ジョブ番号、およびジョブ・ユーザーに応じたアプライ・ジョブ (**IBMSNAP_APPLY_JOB**) 表を検索し、そのジョブを終了します。

以下のいずれかの状態が生じると、**ENDDPRAPY** コマンドはエラー・メッセージを発行します。

- **IBMSNAP_APPLY_JOB** 表が存在していないか、または破壊された場合。
- アプライ修飾子およびコントロール・サーバー名の **IBMSNAP_APPLY_JOB** 表にレコードがない場合。
- アプライ・ジョブがすでに終了している場合。
- コマンドを実行しているユーザー ID が、アプライ・ジョブを終了する権限を所有していない場合。

ENDDPRAPY の例

次の例は、**ENDDPRAPY** コマンドの使用方法を示しています。

例 1

AQHR アプライ修飾子を使用するアプライ・プログラムを終了するには、次のようにします。

```
ENDDPRAPY OPTION(*CNTRLD) APYQUAL(AQHR)
```

アプライ・プログラムは、すべてのタスクが完了すると終了します。

例 2

アプライ・プログラムを即時に終了するには、次のようにします。

ENDDPRAPY OPTION(*IMMED) APYQUAL(AQHR)

終結処理を行うことなく、アプライ・プログラムのタスクはすぐに終了します。

例 3

DB1X という名前のリレーショナル・データベース上にあるアプライ・コントロール表を使用するアプライ・プログラムを終了するには、次のようにします。

ENDDPRAPY OPTION(*CNTRLD) APYQUAL(AQHR) CTLSVR(DB1X)

関連作業:

- 153 ページの 『第 10 章 アプライ・プログラムの操作』

ENDDPRCAP: キャプチャー・プログラムの停止 (OS/400)

DPR キャプチャー・プログラムの終了 (**ENDDPRCAP**) コマンドを使用して、キャプチャー・プログラムを停止します。

このコマンドを使用して、システムをシャットダウンする前にキャプチャー・プログラムを停止します。また、システムで実行している他のプログラムのパフォーマンスを向上させるために、システムの使用がピークになる間、プログラムを停止することもできます。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

ENDDPRCAP コマンドを使用してキャプチャー・プログラムを停止する方法

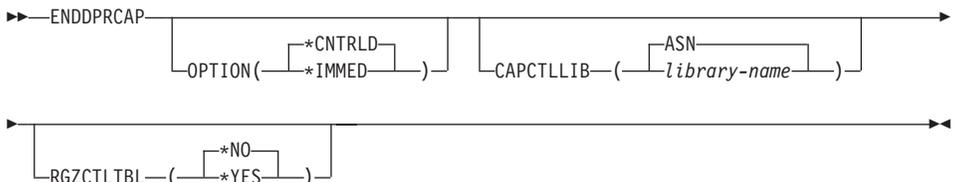


表 37. ENDDPRCAP コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| OPTION | <p>キャプチャー・プログラムを停止する方法を指定します。</p> <p>*CNTRLD (デフォルト)</p> <p>キャプチャー・プログラムは、すべてのタスクを完了した後、通常終了します。</p> <p>ENDDPRCAP コマンドは、*CNTRLD オプションを指定すると、処理が終了するまで長く時間がかかる場合があります。これは、停止する前に、キャプチャー・プログラムはそれに従属する処理をすべて完了させるからです。</p> <p>*IMMED</p> <p>キャプチャー・プログラムは、ENDJOB OPTION(*IMMED) コマンドですべてのタスクを完了してから通常終了します。</p> |
| CAPCTLLIB | <p>キャプチャー・コントロール表が置かれるライブラリーの名前である、キャプチャー・スキーマを指定します。このライブラリーには、ソース表の登録情報を保管する登録 (IBMSNAP_REGISTER) 表が入っています。</p> <p>ASN (デフォルト)</p> <p>キャプチャー・コントロール表は ASN ライブラリー内にあります。ASN ライブラリーはデフォルト・ライブラリーです。</p> <p><i>library-name</i></p> <p>キャプチャー・コントロール表を含むライブラリーの名前。</p> |
| RGZCTLBL | <p>キャプチャー・プログラムの終了時に、コントロール表 (変更データ (CD) および作業単位 (UOW) 表を含む) に対して物理ファイル・メンバーの再編成 (RGZPFM) コマンドが実行されるかどうかを指定します。表に対して RGZPFM コマンド処理が実行されないかぎり、システムはディスク・スペースをリカバーしません。コントロール表がアプライ・プログラム、またはその他のアプリケーション・プログラムからアクセスされているときには、RGZPFM コマンドは実行されません。</p> <p>*NO (デフォルト)</p> <p>RGZPFM コマンドは実行されません。</p> <p>*YES</p> <p>RGZPFM コマンドは実行されます。</p> |

使用上の注意

ENDJOB コマンドを使用すると、一時オブジェクトが QDP4 ライブラリーに残される場合があります。これらのオブジェクトのタイプは ***DTAQ** および ***USRSPC** で、QDP4 *nnnnnn* という名前です。ここで、*nnnnnn* は、オブジェクトを使用したジョブのジョブ番号です。これらのオブジェクトを使用したジョブ (オブジェクト名にあるジョブ番号で識別) がアクティブでないとき、オブジェクトを削除することができます。

このコマンドを発行してもキャプチャー・コントロール・ライブラリー下のジョブが終了しない場合は、***IMMED** オプションを指定して **ENDJOB** コマンドを使用して、このジョブ、および DB2 DataPropagator for iSeries サブシステムで実行中のすべてのジャーナル・ジョブを終了させます。キャプチャー・プログラムだけを終了させたい場合には、同じサブシステムで実行しているアプライ・ジョブを終了させないでください。

まれなことです。が、キャプチャー・コントロール・ジョブが異常終了した場合、キャプチャー・コントロール・ジョブにより作成された (**CAPCTLLIB** パラメーターに従って命名される) ジャーナル・ジョブが実行中のまま残されることがあります。こうしたジョブを終了させる唯一の方法は、*IMMED または *CNTRLD オプションのいずれかを指定した **ENDJOB** コマンドを使用することです。

ENDDPRCAP の例

次の例は、**ENDDPRCAP** コマンドの使用方法を示しています。

例 1

すべての処理タスクが完了した後、ASN ライブラリー内のキャプチャー・コントロール表を使用するキャプチャー・プログラムを終了するには、次のようにします。

```
ENDDPRCAP OPTION(*CNTRLD) CAPCTLLIB(ASN) RGZCTLTBL(*NO)
```

例 2

キャプチャー・スキーマ BSN のキャプチャー・プログラムを即時に終了するには、次のようにします。

```
ENDDPRCAP OPTION(*IMMED) CAPCTLLIB(BSN) RGZCTLTBL(*NO)
```

例 3

すべての処理タスクが完了した後、キャプチャー・プログラムを終了し、キャプチャー・コントロール表を再編成するには、次のようにします。

```
ENDDPRCAP OPTION(*CNTRLD) CAPCTLLIB(ASN) RGZCTLTBL(*YES)
```

関連作業:

- 129 ページの 『第 9 章 キャプチャー・プログラムの操作』

GRTDPRAUT: ユーザーの許可 (OS/400)

DPR 権限の付与 (**GRTDPRAUT**) コマンドは、ユーザーがキャプチャー・プログラムおよびアプライ・プログラムを実行できるように、レプリケーション・コントロール表に対するユーザーのリストを許可します。たとえば、キャプチャー・プログラムおよびアプライ・プログラムを実行しているユーザーの許可要件が、レプリケーション・ソースおよびターゲットを定義するユーザーの許可要件とは異なる場合があります。

権限を付与するための *ALLOBJ 権限を所有している必要があります。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

GRTDPRAUT

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

GRTDPRAUT コマンドを使用して、レプリケーション・コントロール表に対してユーザーを許可する方法

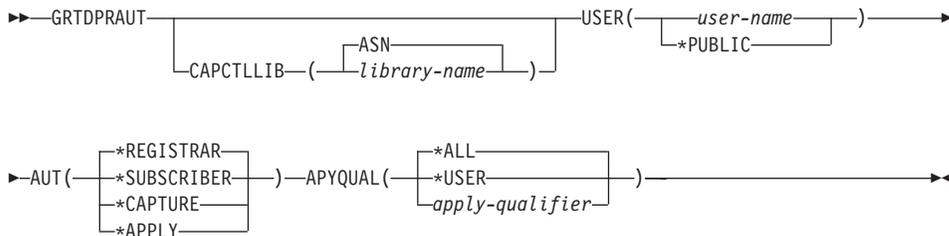


表 38. GRTDPRAUT コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| CAPCTLLIB | <p>ユーザーが権限を付与されているレプリケーション・コントロール表を含むライブラリーである、キャプチャー・スキーマを指定します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。</p> <p><i>library-name</i> レプリケーション・コントロール表を含むライブラリーの名前。</p> |
| USER | <p>権限を所有するユーザーを指定します。</p> <p><i>user-name</i> 権限を所有するユーザーの名前を最大 50 まで指定します。</p> <p>*PUBLIC ファイルに対して *PUBLIC 許可を認可するよう指定しますが、(その許可がタスクを行う上で不十分な場合には) 特定の許可を所有していないユーザー、ファイルに関連付けられた許可リストにないユーザー、およびグループ・プロファイルに許可がないユーザーに関してのみ使用されます。</p> |

表 38. GRTDPRAUT コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------|---|
| AUT | <p>付与されている権限のタイプを指定します。</p> <p>*REGISTRAR (デフォルト) ユーザーは、登録を定義、変更、および消去するための権限を付与されます。</p> <p>AUT(*REGISTRAR) 付き権限の完全なリストについては、419 ページの表 39 を参照してください。</p> <p>*SUBSCRIBER ユーザーは、サブスクリプション・セットを定義、変更、および消去するための権限を付与されます。</p> <p>AUT(*SUBSCRIBER) 付き権限の完全なリストについては、419 ページの表 40 を参照してください。</p> <p>*CAPTURE ユーザーは、キャプチャー・プログラムを実行するための権限を付与されます。</p> <p>AUT(*CAPTURE) を付与された権限の完全なリストについては、420 ページの表 41 を参照してください。</p> <p>*APPLY ユーザーは、アプライ・プログラムを実行するための権限を付与されます。</p> <p>このコマンドは、アプライ・プログラムがアクセスする他のデータベースにあるオブジェクトに権限を付与することはありません。</p> <p>アプライ・プログラムが起動されるとき、DRDA アプリケーション・サーバーのジョブに関連するユーザーは、*APPLY 権限も付与されている必要があります。ソースが iSeries サーバーの場合は、USER パラメーターで指定されたアプリケーション・サーバー・ジョブ・ユーザーと、APYQUAL パラメーターで指定されたアプライ修飾子を指定して、ソース・サーバー・システム上で GRTDPRAUT コマンドを実行する必要があります。</p> <p>ターゲット・サーバーがコントロール・サーバーと同じで、その両方がコマンドが実行されるシステムに常駐していない場合、権限がターゲット表に付与されることはありません。</p> <p>AUT(*APPLY) を付与された権限の完全なリストについては、422 ページの表 42 を参照してください。</p> |

表 38. GRTDPRAUT コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| APYQUAL | <p>USER パラメーターで指定された、ユーザーにより使用されるアプライ修飾子を指定します。このパラメーターは、AUT(*APPLY) または AUT(*SUBSCRIBER) が指定されるときだけ使われます。</p> <p>*ALL (デフォルト) ユーザーは、すべてのアプライ修飾子に関して、アプライ・プログラムを実行する、またはサブスクリプション・セットを定義または除去する権限を付与されます。</p> <p>*USER USER パラメーターで指定されたユーザーは、ユーザー名と同じアプライ修飾子を持つサブスクリプション・セットに対する権限を付与されます。</p> <p><i>apply-qualifier</i> ユーザーは、アプライ・プログラムの実行、または、このアプライ修飾子に関連するアプライ修飾子のサブスクリプション・セットの定義および除去の権限を付与されます。</p> <ul style="list-style-type: none"> ユーザーは、APYQUAL パラメーターを指定して入力された値と一致する APPLY_QUAL 列の中に値があるブルーニング・コントロール (IBMSNAP_PRUNCNTL) 表のレコードに関連付けられているすべてのレプリケーション・ソース、変更データ (CD) 表、および整合変更データ (CCD) 表に対する権限を付与されます。 ユーザーは、このシステム上にあるサブスクリプション・メンバー (IBMSNAP_SUBS_MEMBR) 表に記載されたサブスクリプション・セットに対する権限を付与されます。 |

使用上の注意

使用中のファイルでは許可を変更できないため、キャプチャー・プログラムまたはアプライ・プログラムの実行中や、ソース表を使用するアプリケーションがアクティブのときは、**GRTDPRAUT** コマンドを使用することはできません。

以下の表では、

- AUT(*REGISTRAR)
- AUT(*SUBSCRIBER)
- AUT(*CAPTURE)
- AUT(*APPLY)

を、**GRTDPRAUT** コマンドに対して指定すると付与される権限をリストします。

以下の表では、**GRTDPRAUT** コマンドに対して AUT(*REGISTRAR) を指定すると付与される権限をリストします。

表 39. GRTDPRAUT AUT(*REGISTRAR) を指定して付与される権限

| ライブラリー | オブジェクト | 種類 | 許可 |
|------------------------|--------------------|---------|-----------------------------------|
| QSYS | capctllib | *LIB | *USE、*ADD |
| capctllib ¹ | QSQRN | *JRN | *OBJOPR、*OBJMGT |
| capctllib ¹ | QZS8CTLBLK | *USRSPC | *CHANGE |
| capctllib ¹ | IBMSNAP_REGISTER | *FILE | *OBJOPR、*READ、 *ADD、*UPDT、*DLT |
| capctllib ¹ | IBMSNAP_REGISTERX | *FILE | *OBJOPR、*READ、 *ADD、*UPDT、*DLT |
| capctllib ¹ | IBMSNAP_REGISTERX1 | *FILE | *OBJOPR、*READ、 *ADD、*UPDT、*DLT |
| capctllib ¹ | IBMSNAP_REGISTERX2 | *FILE | *OBJOPR、*READ、 *ADD、*UPDT、*DLT |
| capctllib ¹ | IBMSNAP_REG_EXT | *FILE | *OBJOPR、*READ、 *ADD、*UPDT、*DLT |
| capctllib ¹ | IBMSNAP_REG_EXTX | *FILE | *OBJOPR、*READ、 *ADD、*UPDT、*DLT |
| capctllib ¹ | IBMSNAP_PRUNCNTL | *FILE | *OBJOPR、*READ |
| capctllib ¹ | IBMSNAP_PRUNCNTLX | *FILE | *OBJOPR、*READ |
| capctllib ¹ | IBMSNAP_PRUNCNTLX1 | *FILE | *OBJOPR、*READ |
| capctllib ¹ | IBMSNAP_PRUNCNTLX2 | *FILE | *OBJOPR、*READ |
| capctllib ¹ | IBMSNAP_PRUNCNTLX3 | *FILE | *OBJOPR、*READ |
| ASN | ASN4B* | *SQLPKG | *USE |
| ASN | ASN4C* | *SQLPKG | *USE |

注:

1. ライブラリー列の *capctllib* エントリーは、GRTDPRAUT コマンドの **CAPCTLLIB** パラメーターに渡される値を表します。このコマンドは、同時には 1 つのキャプチャー・コントロール・ライブラリーに対する権限しか更新できません。

以下の表では、GRTDPRAUT コマンドに対して AUT(*SUBSCRIBER) を指定すると付与される権限をリストします。

表 40. GRTDPRAUT AUT(*SUBSCRIBER) を指定して付与される権限

| ライブラリー | オブジェクト | 種類 | 許可 |
|--------|--------------------|-------|---------------------------------|
| QSYS | ASN | *LIB | *OBJOPR、*READ、 *ADD、*EXECUTE |
| QSYS | capctllib | *LIB | *OBJOPR、*READ、 *ADD、*EXECUTE |
| ASN | IBMSNAP_SUBS_SET | *FILE | *CHANGE |
| ASN | IBMSNAP_SUBS_COLS | *FILE | *CHANGE |
| ASN | IBMSNAP_SUBS_EVENT | *FILE | *CHANGE |
| ASN | IBMSNAP_SUBS_STMTS | *FILE | *CHANGE |
| ASN | IBMSNAP_SUBS_MEMBR | *FILE | *CHANGE |

表 40. GRTDPRAUT AUT(*SUBSCRIBER) を指定して付与される権限 (続き)

| ライブラリー | オブジェクト | 種類 | 許可 |
|------------------------|-------------------|---------|--|
| capctllib ¹ | IBMSNAP_REGISTER | *FILE | *OBJOPR、*READ、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REG_EXT | *FILE | *OBJOPR、*READ、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTL | *FILE | *OBJOPR、*READ、 *DLT、*ADD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTLX | *FILE | *USE |
| ASN | ASN4A* | *SQLPKG | *USE |
| ASN | ASN4U* | *SQLPKG | *USE |

注:

1. ライブラリー列の *capctllib* エントリーは、**GRTDPRAUT** コマンドの **CAPCTLLIB** パラメーターに渡される値を表します。このコマンドは、同時には 1 つのキャプチャー・コントロール・ライブラリーに対する権限しか更新できません。

以下の表では、**GRTDPRAUT** コマンドに対して AUT(*CAPTURE) を指定すると付与される権限をリストします。

表 41. GRTDPRAUT AUT(*CAPTURE) を指定して付与される権限

| ライブラリー | オブジェクト | 種類 | 許可 |
|------------------------|--------------------|-------|--|
| QSYS | capctllib | *LIB | *OBJOPR、*OBJMGT、 *READ、*EXECUTE |
| QSYS | QDP4 | *LIB | *OBJOPR、*ADD、 *READ、*EXECUTE |
| capctllib ¹ | QZSN | *MSGQ | *CHANGE |
| capctllib ¹ | IBMSNAP_REGISTER | *FILE | *OBJOPR、*OBJMGT、 *READ、*ADD、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTERX | *FILE | *OBJOPR、*OBJMGT、 *READ、*ADD、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTERX1 | *FILE | *OBJOPR、*OBJMGT、 *READ、*ADD、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTERX2 | *FILE | *OBJOPR、*OBJMGT、 *READ、*ADD、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REG_EXT | *FILE | *OBJOPR、*OBJMGT、 *READ、*ADD、 *UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REG_EXTX | *FILE | *OBJOPR、*OBJMGT、 *READ、*ADD、 *UPD、*EXECUTE |

表 41. GRTDPRAUT AUT(*CAPTURE) を指定して付与される権限 (続き)

| ライブラリー | オブジェクト | 種類 | 許可 |
|------------------------|--------------------|---------|---|
| capctllib ¹ | IBMSNAP_PRUNCNTL | *FILE | *OBJOPR、*OBJMGT、 *READ、*UPD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTLX | *FILE | *OBJOPR、*OBJMGT、 *READ、*UPD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTLX1 | *FILE | *OBJOPR、*OBJMGT、 *READ、*UPD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTLX2 | *FILE | *OBJOPR、*OBJMGT、 *READ、*UPD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTLX3 | *FILE | *OBJOPR、*OBJMGT、 *READ、*UPD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_CAPTRACE | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_CAPTRACEX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_RESTART | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_RESTARTX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_AUTHTKN | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_AUTHTKNX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_UOW | *FILE | *OBJOPR、*OBJMGT、 *READ、*UPD、 *DLT、*ADD、 *EXECUTE |
| capctllib ¹ | IBMSNAP_UOW_IDX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_PRUNE_SET | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_PRUNE_SETX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_CAPPARMS | *FILE | *READ、*EXECUTE |
| capctllib ¹ | IBMSNAP_SIGNAL | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_SIGNALX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_CAPMON | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_CAPMONX | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_PRUNE_LOCK | *FILE | *CHANGE |
| ASN | ASN4B* | *SQLPKG | *USE |
| ASN | ASN4C* | *SQLPKG | *USE |
| ASN | QZS8CTLBLK | *USRSPC | *CHANGE |

注:

1. ライブラリー列の *capctllib* エントリーは、**GRTDPRAUT** コマンドの **CAPCTLLIB** パラメーターに渡される値を表します。このコマンドは、同時には 1 つのキャプチャー・コントロール・ライブラリーに対する権限しか更新できません。

GRTDPRAUT

以下の表では、**GRTDPRAUT** コマンドに対して **AUT(*APPLY)** を指定すると付与される権限をリストします。

表 42. *GRTDPRAUT AUT(*APPLY)* を指定して付与される権限

| ライブラリー | オブジェクト | 種類 | 許可 |
|------------------------|-----------------------|-------|---|
| QSYS | ASN | *LIB | *OBJOPR、*READ、*EXECUTE |
| QSYS | capctllib | *LIB | *OBJOPR、*READ、*EXECUTE |
| QDP4 | QZSNAPV2 | *PGM | *OBJOPR、*READ、*OBMGT、*OBJALTER、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTER | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTERX | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTERX1 | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTERX2 | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTER_EXT | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_REGISTER_EXTX | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| capctllib ¹ | IBMSNAP_SIGNAL | *FILE | *OBJOPR、*READ、*UPD、*ADD、*EXECUTE |
| capctllib ¹ | IBMSNAP_SIGNALX | *FILE | *OBJOPR、*READ、*UPD、*ADD、*EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNE_LOCK | *FILE | *CHANGE |
| capctllib ¹ | IBMSNAP_UOW | *FILE | *OBJOPR、*READ、*UPD、*ADD、*EXECUTE |
| capctllib ¹ | IBMSNAP_PRUNCNTL | *FILE | *OBJOPR、*READ、*UPD、*ADD、*EXECUTE |
| capctllib ¹ | IBMSNAP_AUTHTKN | *FILE | *OBJOPR、*READ、*UPD、*ADD、*EXECUTE |
| capctllib ¹ | IBMSNAP_AUTHTKNX | *FILE | *OBJOPR、*READ、*UPD、*ADD、*EXECUTE |
| ASN | IBMSNAP_SUBS_SET | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |
| ASN | IBMSNAP_SUBS_SETX | *FILE | *OBJOPR、*READ、*UPD、*EXECUTE |

表 42. GRTDPRAUT AUT(*APPLY) を指定して付与される権限 (続き)

| ライブラリー | オブジェクト | 種類 | 許可 |
|--------|--------------------|---------|--|
| ASN | IBMSNAP_APPLYTRAIL | *FILE | *OBJOPR、*READ、 *UPD、*ADD、 *EXECUTE |
| ASN | IBMSNAP_APPLYTRACE | *FILE | *OBJOPR、*READ、 *UPD、*EXECUTE |
| ASN | IBMSNAP_APPLYTRACX | *FILE | *OBJOPR、*READ、 *UPD、*EXECUTE |
| ASN | IBMSNAP_SUBS_COLS | *FILE | *USE |
| ASN | IBMSNAP_SUBS_EVENT | *FILE | *USE |
| ASN | IBMSNAP_SUBS_STMTS | *FILE | *USE |
| ASN | IBMSNAP_SUBS_MEMBR | *FILE | *USE |
| ASN | ASN4A* | *SQLPKG | *USE |
| ASN | ASN4U* | *SQLPKG | *USE |
| ASN | IBMSNAP_APPLY_JOB | *FILE | *OBJOPR、*READ、 *UPD、*ADD、 *EXECUTE |

注:

1. ライブラリー列の *capctllib* エントリーは、GRTDPRAUT コマンドの CAPCTLLIB パラメーターに渡される値を表します。このコマンドは、同時には 1 つのキャプチャー・コントロール・ライブラリーに対する権限しか更新できません。

GRTDPRAUT の例

次の例は、GRTDPRAUT コマンドの使用方法を示しています。

例 1

登録の定義および変更を行う権限を USER1 という名前のユーザーに許可するには、次のようにします。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*REGISTRAR)
```

例 2

サブスクリプション・セットの定義および変更を行う権限を USER1 という名前のユーザーに許可するには、次のようにします。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*SUBSCRIBER)
```

例 3

キャプチャー・プログラムを実行する権限を USER1 という名前のユーザーに許可するには、次のようにします。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*CAPTURE)
```

例 4

アプライ修飾子 A1 に関連付けられた既存のサブスクリプション・セットの定義および変更を行う権限を USER1 という名前のユーザーに許可するには、次のようにします。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*SUBSCRIBER) APYQUAL(A1)
```

例 5

ターゲット・サーバーとコントロール・サーバーが同じであるという条件で、アプライ修飾子 A1 に関連付けられたすべてのサブスクリプション・セットのコントロール・サーバー・システム上にあるアプライ・プログラムを実行するための権限をユーザーに与えるには、以下の操作を行ってください。

1. アプライ・プログラムが稼働するシステムで、以下のコマンドを実行します。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*APPLY) APYQUAL(A1)
```

2. ソース・サーバー・システム上で適切な **GRTDPRAUT** コマンドを実行します。

- アプライ・プログラムによって使われるソース・サーバーのアプリケーション・サーバー・ジョブが、ユーザー・プロファイル USER1 の下で実行される場合、ソース・サーバー・システムで以下のコマンドを実行します。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(USER1) AUT(*APPLY) APYQUAL(A1)
```

- アプライ・プログラムによって使われるソース・サーバーのアプリケーション・サーバー・ジョブが、別のユーザー・プロファイル (たとえば、QUSER) の下で実行される場合には、ソース・サーバー・システムで以下のコマンドを実行します。

```
GRTDPRAUT CAPCTLLIB(ASN) USER(QUSER) AUT(*APPLY) APYQUAL(A1)
```

関連作業:

- 17 ページの 『第 2 章 レプリケーションの設定』

関連資料:

- 352 ページの 『asnpwd: パスワード・ファイルの保守 (UNIX および Windows)』
- 436 ページの 『RVKDPRAUT: 権限の取り消し (OS/400)』

INZDPRCAP: DPR キャプチャー・プログラムの再初期化 (OS/400)

DPR キャプチャー・プログラムの初期化 (**INZDPRCAP**) コマンドを使用して、ソース表の更新済みリストを使用するようにキャプチャー・プログラムに指示し、キャプチャー・プログラムを初期化します。

キャプチャー・プログラムのコントロール下にあるソース表は、キャプチャー・プログラムの実行中に変更することができます。**INZDPRCAP** コマンドを使用して、キャプチャー・プログラムが必ず最新のレプリケーション・ソースを処理するようにします。

このコマンドを実行する前に、キャプチャー・プログラムを実行しておく必要があります。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

INZDPRCAP コマンドを使用してキャプチャー・プログラムを初期化する方法

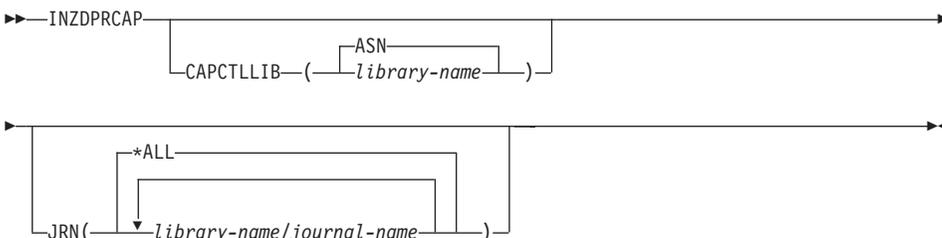


表 43. INZDPRCAP コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| CAPCTLLIB | <p>キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。ASN ライブラリーはデフォルト・ライブラリーです。</p> <p><i>library-name</i> キャプチャー・コントロール表を含むライブラリーの名前。</p> |
| JRN | <p>最大 50 のジャーナルのサブセットをキャプチャー・プログラムが使用して作動するように指定します。キャプチャー・プログラムは、現在このジャーナルに記録されているすべてのソース表の処理を開始します。</p> <p>*ALL (デフォルト) キャプチャー・プログラムは、すべてのジャーナルを使って動作します。</p> <p><i>library-name/journal-name</i> キャプチャー・プログラムが使用して作動するジャーナルの修飾名。</p> |

INZDPRCAP の例

次の例は、INZDPRCAP コマンドの使用方法を示しています。

例 1

TRAINING という名前のライブラリーにある QSQJRN ジャーナルを使用して、キャプチャー・プログラムの初期化を行うには、次のようにします。

```
INZDPRCAP CAPCTLLIB(ASN) JRN(TRAINING/QSQJRN)
```

キャプチャー・コントロール表はデフォルト ASN スキーマ内にあります。

例 2

すべてのジャーナルを扱うキャプチャー・プログラムの初期化を行うには、次のようにします。

```
INZDPRCAP CAPCTLLIB(BSN) JRN(*ALL)
```

キャプチャー・コントロール表は BSN という名前のスキーマ内にあります。

関連作業:

- 129 ページの 『第 9 章 キャプチャー・プログラムの操作』

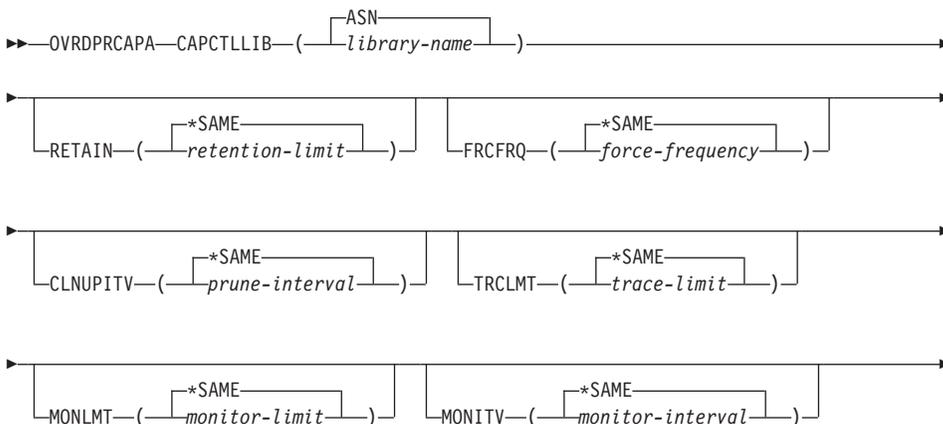
OVRDPRCAPA: DPR キャプチャー属性のオーバーライド (OS/400)

実行中のキャプチャー・プログラムの動作を変更するには、DPR キャプチャー属性のオーバーライド (**OVRDPRCAPA**) コマンドを使用します。このコマンドは、キャプチャー・プログラムの始動時にキャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表、または **STRDPRCAP** コマンドからキャプチャー・プログラムに渡された値をオーバーライドすることにより、プログラムの動作を変更します。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

OVRDPRCAPA コマンドを使用してキャプチャー・プログラムの属性をオーバーライドする方法



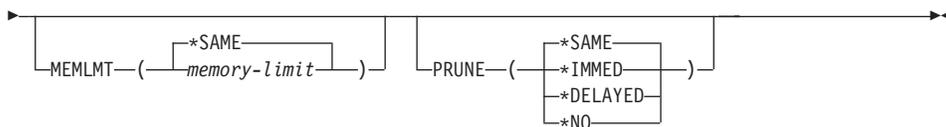


表 44. OVRDPRCAPA コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| CAPCTLLIB | <p>キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。このライブラリーには、ソース表の登録情報を保管する登録 (IBMSNAP_REGISTER) 表が入っています。このパラメーターは必須です。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。</p> <p><i>library-name</i> キャプチャー・コントロール表を含むライブラリーの名前。 CRTPRTBL コマンドに CAPCTLLIB パラメーターを指定して、このライブラリーを作成することができます。</p> |
| RETAIN | <p>データが除去されずに、変更データ (CD)、作業単位 (UOW)、シグナル (IBMSNAP_SIGNAL)、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表に保存される分数を指定します。</p> <p>この値は、DPR キャプチャー・プログラムの始動 (STRDPRCAP) コマンドの CLNUPITV パラメーターと共同で処理を行います。最初にキャプチャー・プログラムは、現在実行中のアプライ・プログラムの最も古いものよりも古い、CD、UOW、IBMSNAP_SIGNAL、または IBMSNAP_AUTHTKN 行を削除します。その後、CD、UOW、IBMSNAP_SIGNAL、または IBMSNAP_AUTHTKN 表の新規の行、または残りの行は、経過時間が RETAIN パラメーターの値に達したときに削除されます。</p> <p>表内のデータに矛盾が生じないように、データがこの RETAIN パラメーター値に達する前に変更情報がコピーされるようにアプライ・インターバルを設定してください。データの矛盾が発生した場合、アプライ・プログラムはフル・リフレッシュを実行します。</p> <p>デフォルトは 10,080 分 (7 日) です。最大値は、35000000 分です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>retention-limit</i> 新しい保持制限値。</p> |

表 44. OVRDPRCAPA コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| FRCFRQ | <p>キャプチャー・プログラムが変更データ (CD) 表および作業単位 (UOW) 表に変更を書き込む頻度 (30 秒から 600 秒) を指定します。</p> <p>キャプチャー・プログラムは、バッファーがフルになるか、FRCFRQ 時間制限が満了するか、いずれか先に発生した時点で、これらの変更をアプライ・プログラムから使用可能にします。このパラメーター値は、キャプチャー・プログラムが DPR キャプチャー・プログラムの初期化 (INZDPRCAP) コマンドからの変更に応答するためにかかる時間の長さに影響します。</p> <p>このパラメーターは、ソース表の変更率が少ないサーバー上で、アプライ・プログラムが変更をより早期に使用できるようにするために使用してください。FRCFRQ パラメーター値はグローバル値であり、すべての登録済みソース表で使用されます。FRCFRQ 値を低い数値に設定すると、システム・パフォーマンスが影響を受ける可能性があります。</p> <p>デフォルトは 30 秒です。</p> <p>*SAME (デフォルト) この値は変更されません。</p> <p><i>force-frequency</i> キャプチャー・プログラムが、アプライ・プログラムで変更を使用できるようにする前に、CD 表および UOW 表の変更をバッファー・スペース内に保持する秒数の新規の値。</p> |
| CLNUPITV | <p>キャプチャー・プログラムが変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表から古いレコードを除去するまでの最大時間 (時間単位) を指定します。</p> <p>このパラメーターは、RETAIN パラメーターと組み合わせて、CD、UOW、IBMSNAP_SIGNAL、および IBMSNAP_AUTHTKN 表のブルーニング、MONLMT パラメーターと組み合わせて、IBMSNAP_CAPMON 表のブルーニング、TRCLMT パラメーターと組み合わせて、IBMSNAP_CAPTRACE 表のブルーニングをコントロールします。</p> <p>(キャプチャー・プログラムの RETAIN、MONLMT、および TRCLMT パラメーターを設定するには、STRDPRCAP コマンドを使用します。)</p> <p>CLNUPITV パラメーターの値は、自動的に時間から秒に変換され、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の PRUNE_INTERVAL 列に保管されます。</p> <p>*SAME (デフォルト) このキャプチャー属性値は変更されません。</p> <p><i>prune-interval</i> 時間数 (1 から 100) で指定された、ブルーニング・インターバル。</p> |

表 44. OVRDPRCAPA コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| TRCLMT | <p>キャプチャー・トレース (IBMSNAP_CAPTRACE) 表の整理の頻度を示す、トレース限度を指定します。</p> <p>*SAME (デフォルト) キャプチャー・プログラムは、現行のトレース限度値を継続して使用します。</p> <p><i>trace-limit</i> IBMSNAP_CAPTRACE 表の各ブルーニング操作の間隔の分数。</p> |
| MONLMT | <p>キャプチャー・モニター (IBMSNAP_CAPMON) 表の整理の頻度を示す、モニター限度を指定します。</p> <p>*SAME (デフォルト) キャプチャー・プログラムは、現行のモニター限度値を継続して使用します。</p> <p><i>monitor-limit</i> IBMSNAP_CAPMON 表の各ブルーニング操作の間隔の分数。</p> |
| MONITV | <p>キャプチャー・プログラムがキャプチャー・モニター (IBMSNAP_CAPMON) 表に行を挿入する頻度を示すモニター・インターバル (秒単位) を指定します。</p> <p>*SAME (デフォルト) キャプチャー・プログラムは、現行のモニター・インターバル値の使用を継続します。</p> <p><i>monitor-interval</i> IBMSNAP_CAPMON 表への行挿入の間隔の秒数。モニター・インターバルは、少なくとも 120 秒 (2 分) あげる必要があります。120 よりも小さい数値がユーザーから入力された場合、このコマンドは自動的にパラメーター値を 120 に設定します。</p> |
| MEMLMT | <p>キャプチャー・ジャーナル・ジョブが使用できるメモリーの最大サイズ (MB 単位) を指定します。</p> <p>*SAME (デフォルト) キャプチャー・プログラムは、現行のメモリー限度値を継続して使用します。</p> <p><i>memory-limit</i> メモリーの最大値の MB 数。</p> |

表 44. OVRDPRCAPA コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|--------------|--|
| PRUNE | <p>キャプチャー・プログラムが変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表から行を削除する方法を変更するには、このパラメーターを使用します。</p> <p>*SAME (デフォルト) キャプチャー・プログラムは、STRDPRCAP コマンドの始動時にユーザーから指定されたブルーニング・パラメーターを継続して使用します。</p> <p>*IMMED キャプチャー・プログラムは、STRDPRCAP コマンドの始動時にユーザーから指定された CLNUPITV パラメーターの値に関係なく、表のブルーニングを即時に開始します。</p> <p>*DELAYED キャプチャー・プログラムは、指定されたブルーニング・インターバルの最後に古い行を削除します。</p> <p>PRUNE(*DELAYED) は、STRDPRCAP コマンドの CLNUPITV パラメーターの 2 番目の部分が *IMMED または *DELAYED に設定された場合は、ブルーニングの頻度に影響を与えることはありません。しかし PRUNE(*DELAYED) は、STRDPRCAP コマンドの始動時に CLNUPITV パラメーターの 2 番目の部分が *NO に設定された場合は、ブルーニングを開始します。</p> <p>*NO キャプチャー・プログラムはブルーニングを開始しません。この値は、STRDPRCAP コマンドの CLNUPITV パラメーターの設定値をオーバーライドします。</p> |

OVRDPRCAPA の例

次の例は、**OVRDPRCAPA** コマンドの使用方法を示しています。

例 1

CD、UOW、IBMSNAP_SIGNAL、IBMSNAP_CAPMON、IBMSNAP_CAPTRACE、および IBMSNAP_AUTHTKN 表 (デフォルト ASN ライブラリー内にある) のブルーニング・パラメーターを変更し、実行中のキャプチャー・プログラムのキャプチャー・ジャーナル・ジョブの IBMSNAP_CAPMON モニター・インターバルおよびメモリー限度を変更するには、次のようにします。

```
OVRDPRCAPA CAPCTLLIB(ASN) CLNUPITV(12) MONITV(600) MEMLMT(64)
```

例 2

BSN ライブラリー内にある CD、UOW、IBMSNAP_SIGNAL、IBMSNAP_CAPMON、IBMSNAP_CAPTRACE、および IBMSNAP_AUTHTKN 表のブルーニングを開始するには、次のようにします。

```
OVRDPRCAPA CAPCTLLIB(BSN) PRUNE(*IMMED)
```

関連作業:

- 129 ページの『第 9 章 キャプチャー・プログラムの操作』

関連資料:

- 340 ページの『asnccmd: キャプチャーの操作 (UNIX、Windows、z/OS)』

RMVDPRREG: DPR 登録の除去 (OS/400)

ソース表がデータ伝搬の目的で使用されないようにするために、登録 (IBMSNAP_REGISTER) 表から 1 つのソース表を除去するには、DPR 登録の除去 (RMVDPRREG) コマンドを使用します。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

RMVDPRREG コマンドを使用して DPR 登録を除去する方法

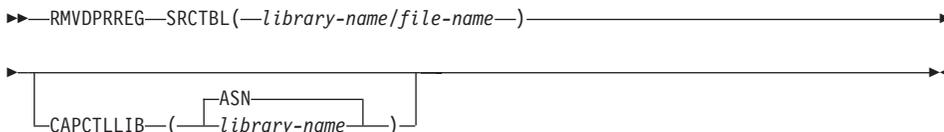


表 45. RMVDPRREG コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|-----------|---|
| SRCTBL | 除去する登録を識別します。このパラメーターは必須です。 <i>library-name/file-name</i> 登録済みファイルの修飾名。 |
| CAPCTLLIB | キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。 ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。 <i>library-name</i> キャプチャー・コントロール表を含むライブラリーの名前。 |

RMVDPRREG の例

次の例は、RMVDPRREG コマンドの使用方法を示しています。

表 46. RMVDPRSUB コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|-----------------|---|
| APYQUAL | <p>アプライ・プログラムでサブスクリプション・セットの識別に使用されるアプライ修飾子を指定します。このパラメーターは必須です。</p> <p><i>apply-qualifier</i> アプライ修飾子の名前。</p> |
| SETNAME | <p>サブスクリプション・セットの名前を指定します。このパラメーターは必須です。</p> <p><i>set-name</i> サブスクリプション・セットの名前。指定されたアプライ修飾子に対して存在していないサブスクリプション・セット名が指定されると、エラー・メッセージが戻されます。</p> |
| CTLSVR | <p>アプライ・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) アプライ・コントロール表はローカルに (RMVDPRSUB コマンドが実行されたマシンに) 存在します。</p> <p><i>rdb-name</i> アプライ・コントロール表が置かれているリレーショナル・データベースの名前。 RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> |
| RMVREG | <p>このコマンドが、サブスクリプション・セット内のすべてのサブスクリプション・セット・メンバーのターゲット表に関連付けられた登録を除去するかどうかを指定します。このパラメーターは、RMVMBRS パラメーターが *YES に設定されている場合にのみ使用してください。</p> <p>*NO (デフォルト) 登録を除去しません。</p> <p>*YES 登録を除去します。</p> |
| DLTTGTBL | <p>このコマンドが、サブスクリプション・セットのドロップ後に、サブスクリプション・セット・メンバーのターゲット表をドロップするかどうかを指定します。このパラメーターは、RMVMBRS パラメーターが *YES に設定されている場合にのみ使用してください。</p> <p>*NO (デフォルト) ターゲット表はドロップされません。</p> <p>*YES ターゲット表はドロップされます。</p> |
| RMVMBRS | <p>このコマンドが、サブスクリプション・セットと、そのサブスクリプション・セット内のすべてのメンバーを除去するかどうかを指定します。</p> <p>*NO (デフォルト) サブスクリプション・セット内に既存のメンバーがある場合には、サブスクリプション・セットは除去されません。</p> <p>*YES サブスクリプション・セットと、そのサブスクリプション・セット・メンバーがすべて除去されます。</p> |

RMVDPRSUB の例

次の例は、**RMVDPRSUB** コマンドの使用方法を示しています。

例 1

サブスクリプション・セット・メンバーを含まない、**SETHR** という名前のサブスクリプション・セットを除去するには、次のようにします。

```
RMVDPRSUB APYQUAL(AQHR) SETNAME(SETHR)
```

例 2

SETHR という名前のサブスクリプション・セットと、そのサブスクリプション・セット・メンバーをすべて除去するには、次のようにします。

```
RMVDPRSUB APYQUAL(AQHR) SETNAME(SETHR) RMVMBRS(*YES)
```

例 3

SETHR という名前のサブスクリプション・セットと、そのサブスクリプション・セット・メンバーのすべて、および関連する登録を除去するには、次のようにします。

```
RMVDPRSUB APYQUAL(AQHR) SETNAME(SETHR) RMVREG(*YES) RMVMBRS(*YES)
```

関連作業:

- 199 ページの 『第 12 章 レプリケーション環境の変更』
- 69 ページの 『第 4 章 ソースのサブスクライブ』

RMVDPRSUBM: DPR サブスクリプション・セット・メンバーの除去 (OS/400)

サブスクリプション・セットから 1 つのサブスクリプション・セット・メンバーを除去するには、**DPR サブスクリプション・セット・メンバーの除去 (RMVDPRSUBM)** コマンドを使用します。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

RMVDPRSUBM コマンドを使用してサブスクリプション・セットから 1 つのサブスクリプション・セット・メンバーを除去する方法

```
▶▶—RMVDPRSUBM—APYQUAL—(—apply-qualifier—)—SETNAME—(—set-name—)————▶▶
```

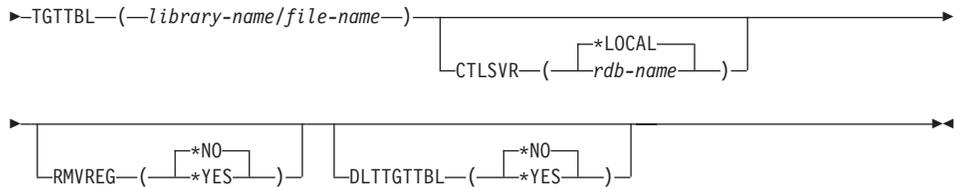


表 47. RMVDPRSUBM コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| APYQUAL | <p>アプライ・プログラムでサブスクリプション・セットの識別に使用されるアプライ修飾子を指定します。このパラメーターは必須です。</p> <p><i>apply-qualifier</i> アプライ修飾子の名前。</p> |
| SETNAME | <p>サブスクリプション・セットの名前を指定します。このパラメーターは必須です。</p> <p><i>set-name</i> サブスクリプション・セットの名前。指定されたアプライ修飾子に対して存在していないサブスクリプション・セット名が指定されると、エラー・メッセージが戻されます。</p> |
| TGTTBL | <p>サブスクリプション・セット・メンバーに関して登録されているターゲット表を指定します。このパラメーターは必須です。</p> <p><i>library-name/file-name</i> ターゲット表の修飾名。</p> |
| CTLSVR | <p>アプライ・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) アプライ・コントロール表はローカルに (RMVDPRSUBM コマンドが実行されたマシンに) 存在します。</p> <p><i>rdb-name</i> アプライ・コントロール表が置かれているリレーショナル・データベースの名前。RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> |
| RMVREG | <p>このコマンドが、サブスクリプション・セット・メンバーのターゲット表に関連付けられた登録を除去するかどうかを指定します。</p> <p>*NO (デフォルト) 登録を除去しません。</p> <p>*YES 登録を除去します。</p> |
| DLTTGTTBL | <p>このコマンドが、サブスクリプション・セット・メンバーのドロップ後に、サブスクリプション・セット・メンバーのターゲット表をドロップするかどうかを指定します。</p> <p>*NO (デフォルト) ターゲット表はドロップされません。</p> <p>*YES ターゲット表はドロップされます。</p> |

RMVDPRSUBM の例

次の例は、**RMVDPRSUBM** コマンドの使用方を示しています。

例 1

EMP という名前のターゲット表を使用するサブスクリプション・セット・メンバーを、RMTRDB1 という名前のリレーショナル・データベース上の SETEMP サブスクリプション・セットから除去するには、次のようにします。

```
RMVDPRSUBM APYQUAL(AQHR) SETNAME(SETEMP) TGTTBL(TGTEMP/EMP) CTLSVR(RMTRDB1)
```

例 2

SETHR サブスクリプション・セットからサブスクリプション・セット・メンバーをドロップし、登録をドロップしてから表をドロップするには、次のようにします。

```
RMVDPRSUBM APYQUAL(AQHR) SETNAME(SETHR) TGTTBL(TGTHR/YTD TAX) RMVREG(*YES)
DLTTGTTBL(*YES)
```

関連作業:

- 69 ページの『第 4 章 ソースのサブスクライブ』

RVKDPRAUT: 権限の取り消し (OS/400)

DPR 権限の取り消し (**RVKDPRAUT**) コマンドは、ユーザーがレプリケーション・ソースおよびサブスクリプション・セットを定義または変更を行うことができないようにするため、レプリケーション・コントロール表に対する権限を取り消します。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

RVKDPRAUT コマンドを使用してレプリケーション・コントロール表の権限を取り消す方法

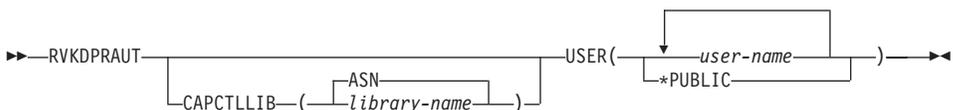


表 48. RVKDPRAUT コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| CAPCTLLIB | <p>ユーザー権限が取り消されるライブラリーの名前である、キャプチャー・スキーマを指定します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表は ASN ライブラリー内にあります。</p> <p><i>library-name</i> レプリケーション・コントロール表を含むライブラリーの名前。</p> |
| USER | <p>権限が取り消されるユーザーを指定します。このパラメーターは必須です。</p> <p><i>user-name</i> 権限が取り消されるユーザーの名前を最大 50 まで指定します。</p> <p>*PUBLIC 特定の許可を所有していないすべてのユーザー、許可リストにないすべてのユーザー、およびグループ・プロファイルが許可を所有していないすべてのユーザーから、許可を取り消すように指定します。</p> |

使用上の注意

以下のいずれかの状態が生じると、このコマンドはエラー・メッセージを戻します。

- 指定したユーザーが存在しない。
- コマンドを実行しているユーザーが、指定したユーザー・プロファイルへの権限を所有していない。
- このコマンドを実行しているユーザーには、DB2 DataPropagator for iSeries のコントロール表に対する権限を取り消す許可がない。
- DB2 DataPropagator for iSeries のコントロール表が存在しない。
- キャプチャー・プログラムまたはアプライ・プログラムが実行中。

RVKDPRAUT の例

次の例は、RVKDPRAUT コマンドの使用方法を示しています。

例 1

ASN ライブラリー下のコントロール表に対する権限を、HJONES という名前のユーザーから取り消すには、次のようにします。

```
RVKDPRAUT CAPCTLLIB(ASN) USER(HJONES)
```

例 2

ASN ライブラリー下のコントロール表に対する権限を、無許可の全ユーザーから取り消すには、次のようにします。

```
RVKDPRAUT CAPCTLLIB(ASN) USER(*PUBLIC)
```

関連作業:

- 17 ページの『第 2 章 レプリケーションの設定』

関連資料:

- 415 ページの『GRTDPRAUT: ユーザーの許可 (OS/400)』

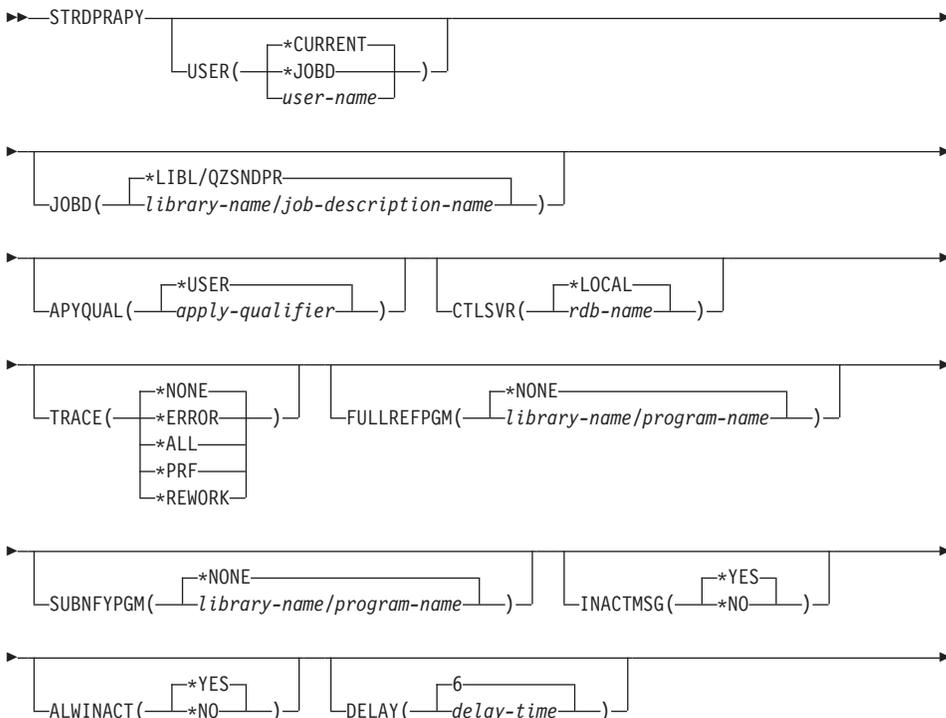
STRDPRAPY: アプライ・プログラムの始動 (OS/400)

DPR アプライ・プログラムの始動 (**STRDPRAPY**) コマンドは、ローカル・システム上でアプライ・プログラムを始動するために使用されます。アプライ・プログラムの実行は、ユーザーがプログラムを停止するまで、またはリカバリー不能エラーが検出されるまで続けられます。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

STRDPRAPY コマンドを使用して DPR アプライ・プログラムを始動する方法



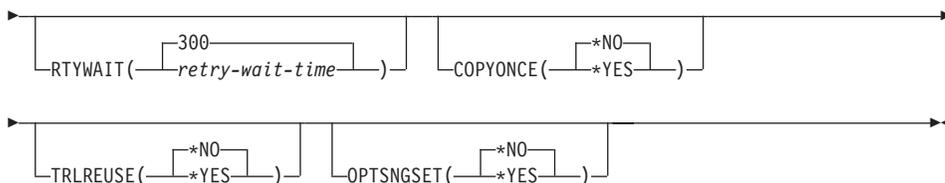


表 49. STRDPRAPY コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|--------------|--|
| USER | <p>アプライ・プログラムが開始するユーザー ID の名前を指定します。このコマンドを実行するには、指定されたユーザー・プロファイルに対する許可が必要です (*USE 権限を持つ)。アプライ・プログラムは、この指定されたユーザー・プロファイルの下で実行されます。</p> <p>コントロール表は、CTLSVR パラメーターで指定されたりレージョナル・データベース上にあります。同じコントロール表は、USER パラメーターに指定した値を無視して使われます。</p> <p>*CURRENT (デフォルト) 現行ジョブに関連付けられたユーザー ID は、このアプライ・プログラムと関連付けられたユーザー ID と同じです。</p> <p>*JOBID このアプライ・プログラムに関連付けられたジョブ記述の中で指定されたユーザー ID。ジョブ記述は USER(*RQD) を指定することはできません。</p> <p><i>user-name</i> このアプライ・プログラムに関連付けられたユーザー ID。以下の IBM 提供のオブジェクトは、このパラメーターに対しては有効ではありません： QDBSHR、QDFTOWN、QDOC、QLPAUTO、QLPINSTALL、QRJE、QSECOFR、QSPL、QSYS、または QTSTRQS。</p> <p>STRDPRAPY コマンドでプロンプトを出すと、F4 キーを押して、サブスクリプション・セットを定義したユーザーのリストを表示することができます。</p> |
| JOBID | <p>アプライ・プログラムをサブミットするときに使用するジョブ記述の名前を指定します。</p> <p>*LIBL/QZSNDPR (デフォルト) DB2 DataPropagator for iSeries から提供されるデフォルトのジョブ記述。</p> <p><i>library-name/job-description-name</i> アプライ・プログラムで使用するジョブ記述の名前。</p> |

表 49. STRDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|----------------|--|
| APYQUAL | <p>アプライ・プログラムによって使われるアプライ修飾子を指定します。このアプライ修飾子によりグループ化されるすべてのサブスクリプション・セットが、アプライ・プログラムで実行されます。</p> <p>*USER (デフォルト) ユーザーが入力する USER パラメーターの値が、アプライ修飾子の名前として使用されます。</p> <p><i>apply-qualifier</i> このアプライ・プログラムによって実行されるサブスクリプション・セットをグループ化するために使用される名前。アプライ修飾子名として最大 18 文字を指定できます。この名前は、リレーショナル・データベース名と同じ命名規則に準拠します。</p> <p>STRDPRAPY コマンドでプロンプトを出すと、F4 キーを押して、既存のサブスクリプション・セットを使用したアプライ修飾子のリストを表示することができます。</p> |
| CTLSVR | <p>アプライ・コントロール表を含むシステムのリレーショナル・データベース名を指定します。</p> <p>*LOCAL (デフォルト) アプライ・コントロール表はローカルに (STRDPRAPY コマンドが実行されたマシンに) 存在します。</p> <p><i>rdb-name</i> アプライ・コントロール表が置かれているリレーショナル・データベースの名前。RDB ディレクトリー・エントリーの作業 (WRKRDBDIRE) コマンドを使用して、この名前を検出することができます。</p> <p>STRDPRAPY コマンドでプロンプトを出すと、F4 キーを押して、使用可能な RDB 名を表示することができます。</p> |

表 49. STRDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-------------------|---|
| TRACE | <p>アプライ・プログラムがトレースを生成するかどうかを指定します。アプライ・プログラムは、QPZSNATRC と呼ばれるスプール・ファイルにトレース・データを書き込みます。</p> <p>*NONE (デフォルト) トレースが生成されません。</p> <p>*ERROR トレースはエラー情報だけを保持します。</p> <p>*ALL トレースはエラーおよび実行フロー情報を保持します。</p> <p>*PRF トレースは、アプライ・プログラムの実行のさまざまな段階でパフォーマンスの分析に使用できる情報を保持します。</p> <p>*REWORK トレースは、アプライ・プログラムによって再処理された行についての情報を保持します。</p> |
| FULLREFPGM | <p>アプライ・プログラムが、出口ルーチン呼び出して、ターゲット表を初期化するかどうかを指定します。アプライ・プログラムはターゲット表のフル・リフレッシュの実行準備が整うと、自分でフル・リフレッシュを実行せずに、指定された出口ルーチン呼び出します。</p> <p>フル・リフレッシュ出口ルーチンがアプライ・プログラムで使用される場合、アプライ・トレール (IBMSNAP_APPLYTRAIL) 表にある ASNLOAD 列の値は Y です。</p> <p>詳しい情報と使用例については、169 ページの『ASNLOAD 出口ルーチンを使ったターゲット表のリフレッシュ』を参照してください。</p> <p>*NONE (デフォルト) フル・リフレッシュ出口ルーチンが使用されません。</p> <p><i>library-name/program-name</i> ターゲット表のフル・リフレッシュを実行するアプライ・プログラムから呼び出されるプログラムの修飾名。たとえば、ライブラリー DATAPROP にあるプログラム ASNLOAD を呼び出すための修飾名は DATAPROP/ASNLOAD です。</p> |

表 49. STRDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| SUBNFYPM | <p>サブスクリプション・セットの処理が終了したときに、アプライ・プログラムから出口ルーチン呼び出すかどうかを指定します。出口ルーチンへの入力には、サブスクリプション・セット名、アプライ修飾子、完了状況、および拒否回数を含む統計が含まれます。</p> <p>通知プログラムを使用すると、作業単位 (UOW) 表を検査して、リジェクトされたトランザクションを判別したり、メッセージの発行やイベントの生成などのアクションをさらに続けるかを判別できます。</p> <p>詳細については、167 ページの『ASNDONE 出口ルーチンの変更 (OS/400)』を参照してください。</p> <p>*NONE (デフォルト) 出口ルーチンが使用されません。</p> <p><i>library-name/program-name</i> サブスクリプション・セットの処理時に、アプライ・プログラムから呼び出される出口ルーチン・プログラムの修飾名。たとえば、ライブラリー DATAPROP にあるプログラム APPLYDONE を呼び出すための修飾名は DATAPROP/APPLYDONE です。</p> |
| INACTMSG | <p>アプライ・プログラムが作業を完了し、一定の期間非アクティブになったときに、メッセージを生成するかどうかを指定します。</p> <p>*YES (デフォルト) アプライ・プログラムがアクティブではなくなる前に、メッセージ ASN1044 を生成します。メッセージ ASN1044 は、アプライ・プログラムが非アクティブであった時間の長さを示します。</p> <p>*NO メッセージが生成されません。</p> |
| ALWINACT | <p>アプライ・プログラムを非アクティブ (スリープ) 状態で実行できるかどうかを指定します。</p> <p>*YES (デフォルト) 処理するものが何もない場合、アプライ・プログラムはスリープします。</p> <p>*NO アプライ・プログラムで処理するものが何もない場合、アプライ・プログラムをサブミットして開始したジョブは終了します。</p> |
| DELAY | <p>連続レプリケーションが使用される場合に、それぞれのアプライ・プログラムのサイクルの終わりにおける遅延時間 (秒単位) を指定します。</p> <p>6 (デフォルト) 遅延時間は 6 秒です。</p> <p><i>delay-time</i> 0 から 6 までの数値で入力された遅延時間。</p> |

表 49. STRDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|-----------------|--|
| RTYWAIT | <p>エラーが生じた場合に、失敗した操作を再試行するまでアプライ・プログラムが待機する時間 (秒単位) を指定します。</p> <p>300 (デフォルト) 再試行待ち時間は 300 秒 (5 分) です。</p> <p><i>retry-wait-time</i> 0 から 35000000 の数値で入力された待ち時間。この時間が過ぎると、アプライ・プログラムは失敗した操作を再試行します</p> |
| COPYONCE | <p>アプライ・プログラムが呼びだされた時点で適格と見なされたサブスクリプション・セットごとに、アプライ・プログラムがコピー・サイクルを 1 回実行するかどうかを指定します。その後、アプライ・プログラムは終了します。適格と見なされるサブスクリプション・セットとは、以下の基準を満たすものです。</p> <ul style="list-style-type: none"> サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の中で (ACTIVATE > 0)。ACTIVATE 列の値がゼロより大きい場合、そのサブスクリプション・セットは無期限にアクティブであるか、または 1 回のみのサブスクリプション処理に使用されています。 (REFRESH_TYPE = R または B) または (REFRESH_TYPE = E であり、指定されたイベントが発生)。REFRESH_TYPE 列の値は IBMSNAP_SUBS_SET 表に保管されます。 <p>IBMSNAP_SUBS_SET 表で MAX_SYNCH_MINUTES 限度と、サブスクリプション・イベント (IBMSNAP_SUBS_EVENT) 表で END_OF_PERIOD タイム・スタンプが指定されている場合、これらの設定が使用されます。</p> <p>*NO (デフォルト) アプライ・プログラムは、適格なサブスクリプション・セットごとにコピー・サイクルを 1 回実行しません。</p> <p>*YES アプライ・プログラムは、適格なサブスクリプション・セットごとにコピー・サイクルを 1 回実行した後、終了します。</p> |
| TRLREUSE | <p>アプライ・プログラムの始動時に、アプライ・プログラムがアプライ・トレール (IBMSNAP_APPLYTRAIL) 表を空にするかどうかを指定します。</p> <p>*NO (デフォルト) アプライ・プログラムはプログラム始動時に IBMSNAP_APPLYTRAIL 表を空にしません。</p> <p>*YES アプライ・プログラムはプログラム始動時に IBMSNAP_APPLYTRAIL 表を空にします。</p> |

表 49. STRDPRAPY コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|--|
| OPTSNGSET | <p>アプライ・プログラムのパフォーマンスを、処理するサブスクリプション・セットが 1 つだけである状況に対して最適化するかどうかを指定します。このパラメーターは、レプリカ・ターゲット表には影響しません。</p> <p>このパラメーターが *YES に設定された場合、アプライ・プログラムはサブスクリプション・セットのメンバーおよび列を一度しかフェッチせず、複数の連続する処理サイクルで同じサブスクリプション・セットを処理する場合は、このフェッチした情報を再利用します。</p> <p>*NO (デフォルト) 処理するサブスクリプション・セットが 1 つだけである場合、アプライ・プログラムのパフォーマンスは最適化されません。</p> <p>*YES 処理するサブスクリプション・セットが 1 つだけである場合、アプライ・プログラムのパフォーマンスは最適化されます。アプライ・プログラムはその後の処理サイクルでサブスクリプション・セット情報を再利用するため、CPU リソースの所要量が減少し、スループット率が向上します。</p> |

使用上の注意

システムの QSTRUPPGM 値で参照されるコマンドを追加することにより、サブシステムを自動的に開始するようシステムを設定することができます。QDP4/QZSNDPR サブシステムを使用する場合、それは **STRDPRAPY** コマンド処理の一部として開始されません。

CTLSVR パラメーターによって指定されたリレーショナル・データベース (RDB) が DB2 Universal Database for iSeries データベースである場合、サーバーの表は ASN ライブラリーにあります。RDB が DB2 Universal Database for iSeries データベースでない場合、ASN を修飾子として使って、表にアクセスすることができます。

アプライ・プログラムを始動するときのエラー条件

以下のいずれかの状態が生じると、**STRDPRAPY** コマンドはエラー・メッセージを戻します。

- ユーザーが存在していない場合。
- コマンドを実行しているユーザーが、そのコマンドまたはジョブ記述で指定した、ユーザー・プロファイルへの権限を所有していない場合。
- アプライ・プログラムのインスタンスが、このアプライ修飾子とコントロール・サーバーの組み合わせのローカル・システムですでにアクティブの場合。
- **CTLSVR** パラメーターによって指定された RDB 名がリレーショナル・データベース・ディレクトリーの中にある場合。
- **CTLSVR** パラメーターによって指定された RDB 上にコントロール表が存在しない場合。

- **APYQUAL** パラメーターによって指定されたアプライ修飾子に対して定義されたサブスクリプション・セットがない場合。

すべてのサブスクリプション・セット (IBMSNAP_SUBS_SET) 表の中の固有のアプライ修飾子ごとに、1 つのアプライ・プログラムを始動する必要があります。

STRDPRAPY コマンドを発行するたびに異なるアプライ修飾子を指定することにより、複数のアプライ・プログラムを始動できます。これらのアプライ・プログラムは、同じユーザー・プロファイルの下で実行します。

アプライ・プログラムのジョブの識別

各アプライ・プログラムは、アプライ修飾子とコントロール・サーバー名の両方を使って識別されます。実行時には、アプライ・プログラムに応じて始まるジョブに十分な外部属性がないため、特定のアプライ修飾子とコントロール・サーバーとの組み合わせに関連付けられているアプライ・プログラムを正確に識別することはできません。そのため、以下に示す方法でジョブが識別されます。

- **USER** パラメーターと関連付けられたユーザー・プロファイルの下で、ジョブが開始します。
- アプライ修飾子の最初の 10 文字は切り捨てられて、ジョブ名になります。
- DB2 DataPropagator for iSeries は、ローカル・システム上の ASN ライブラリー内に名前のあるアプライ・ジョブ (IBMSNAP_APPLY_JOB) 表を保持しています。この表が、アプライ修飾子およびコントロール・サーバーの値を、正しいアプライ・プログラムのジョブにマップします。
- ジョブ・ログを表示することができます。アプライ修飾子とコントロール・サーバー名が、アプライ・プログラムに対する呼び出しで使用されています。

通常、以下に示す 2 つの条件が当てはまれば、QZSNDPR で実行中のジョブのリストを表示することにより、正しいアプライ・プログラムを識別することができます。

- アプライ修飾子の最初の 10 文字が固有である。
- アプライ・プログラムが、ローカル・コントロール・サーバーでのみ開始される。

STRDPRAPY の例

次の例は、**STRDPRAPY** コマンドの使用方を示しています。

例 1

AQHR アプライ修飾子、およびローカルに常駐しているコントロール表を使用して、エラーおよび実行フローの情報を含むトレース・ファイルを生成するアプライ・プログラムを始動するには、次のようにします。

```
STRDPRAPY APYQUAL(AQHR) CTLSVR(*LOCAL) TRACE(*ALL)
```

例 2

ローカルに常駐するアプライ・コントロール表を使用してアプライ・プログラムを始動し、アプライ・プログラムで処理するものが何もない場合は、このアプライ・プログラムを始動したジョブが自動的に終了するように指定するには、次のようにします。

```
STRDPRAPY APYQUAL(AQHR) CTLSVR(*LOCAL) ALWINACT(*NO)
```

例 3

プログラム始動時に IBMSNAP_APPLYTRAIL 表を空にするアプライ・プログラムを始動するには、次のようにします。

```
STRDPRAPY APYQUAL(AQHR) CTLSVR(*LOCAL) TRLREUSE(*YES)
```

例 4

すべてのデフォルト値を使用してアプライ・プログラムを始動するには、次のようにします。

```
STRDPRAPY
```

関連作業:

- 153 ページの 『第 10 章 アプライ・プログラムの操作』

関連資料:

- 328 ページの 『asnapply: アプライの開始 (UNIX、Windows、z/OS)』

STRDPRCAP: キャプチャー・プログラムの始動 (OS/400)

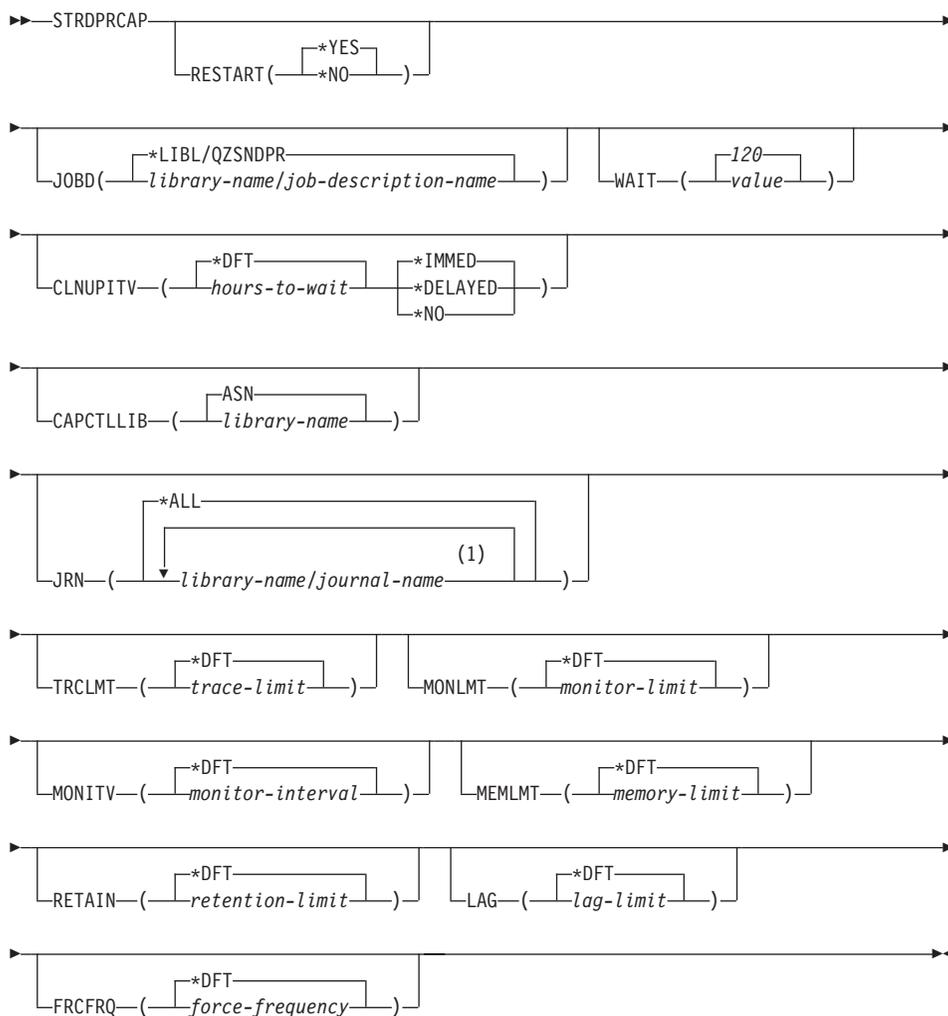
iSeries サーバー上の OS/400 データベース表への変更のキャプチャーを開始するには、DPR キャプチャー・プログラムの始動 (**STRDPRCAP**) コマンドを使用します。このコマンドは登録 (IBMSNAP_REGISTER) 表内のすべてのレプリケーション・ソースを処理するため、適切な権限に基づいてこのコマンドを実行していることを確認してください。

キャプチャー・プログラムを始動すると、停止されるかまたはリカバリー不能エラーが検出されるまで実行を続けます。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

STRDPRCAP コマンドを使用して **DPR** キャプチャー・プログラムを始動する方法



注:

- 1 最大 50 のジャーナルを指定できます。

表 50. STRDPRCAP コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義およびプロンプト |
|----------------|---|
| RESTART | <p>キャプチャー・プログラムがウォーム・スタートとコールド・スタートを処理する方法を指定します。</p> <p>*YES (デフォルト) キャプチャー・プログラムは、以前に終了したときに停止したポイントから変更の処理を継続します。これは、ウォーム・スタートとしても知られており、オペレーションの通常モードです。</p> <p>*NO キャプチャー・プログラムは、変更データ (CD) 表からすべての情報を消去します。JRN(*ALL) を指定すると、キャプチャー・プログラムは作業単位 (UOW) 表からすべての情報を消去します。</p> <p>変更の取り込みが再開される前に、影響を受けたソース表のすべてのサブスクリプションはフル・リフレッシュされます。この処理は、コールド・スタートとしても知られています。</p> <p>RESTART(*NO) および JRN(<i>library-name/journal-name</i>) を指定すると、指定したジャーナルのキャプチャー・プログラムのコールド・スタートを行うことができます。</p> |
| JOB | <p>キャプチャー・プログラムをサブミットするときに使用するジョブ記述の名前を指定します。</p> <p>*LIBL/QZSNDPR (デフォルト) DB2 DataPropagator for iSeries から提供されるデフォルトのジョブ記述を指定します。</p> <p><i>library-name/job-description-name</i> キャプチャー・プログラムで使用するジョブ記述の名前。</p> |
| WAIT | <p>キャプチャー・プログラムが状況をチェックするまで待つ最大時間を秒単位 (60 から 6,000) で指定します。この値を使用して、キャプチャー・プログラムの応答時間を調整することができます。</p> <p>この値を低くすると、キャプチャー・プログラムが終了または初期化するまでにかかる時間は減りますが、システム・パフォーマンスによくない影響を及ぼす場合があります。この値を高くすると、キャプチャー・プログラムが終了または初期化するまでにかかる時間は増えますが、システム・パフォーマンスを向上させることができます。この値をあまり高くしすぎると、キャプチャー・プログラムは定期的に処理を行っていても、応答時間が悪くなることがあります。応答時間がどの程度下がるかは、ソース表に加える変更の量やシステムで発生する他の作業の量によって異なります。</p> <p>120 (デフォルト) キャプチャー・プログラムは 120 秒待機します。</p> <p><i>value</i> キャプチャー・プログラムが待機する最大時間数 (秒)。</p> |

表 50. STRDPRCAP コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|------------------|---|
| CLNUPITV | <p>キャプチャー・プログラムが変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表から古いレコードを除去するまでの最大時間 (時間単位) を指定します。</p> <p>このパラメーターは、RETAIN パラメーターと組み合わせて、CD、UOW、IBMSNAP_SIGNAL、および IBMSNAP_AUTHTKN 表のブルーニング、MONLMT パラメーターと組み合わせて、IBMSNAP_CAPMON 表のブルーニング、TRCLMT パラメーターと組み合わせて、IBMSNAP_CAPTRACE 表のブルーニングをコントロールします。</p> <p>(キャプチャー・プログラムの RETAIN、MONLMT、および TRCLMT パラメーターを設定するには、STRDPRCAP コマンドを使用します。これらのパラメーターの設定値を変更するには、CHGDPRCAPA または OVRDPRCAPA コマンドを使用します。)</p> <p>CLNUPITV パラメーターには 2 つの部分があります。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の PRUNE_INTERVAL 列の値を使用します。</p> <p><i>hours-to-wait</i> 時間数 (1 から 100) で指定された、ブルーニング・インターバル。</p> <p>*IMMED (デフォルト) キャプチャー・プログラムは、指定されたインターバルの開始時に (つまり即時に)、そしてその後は各インターバルで、古いレコードを除去します。</p> <p>*DELAYED キャプチャー・プログラムは、指定されたインターバルの終了時に、そしてその後は各インターバルで、古いレコードを除去します。</p> <p>*NO キャプチャー・プログラムはレコードを除去しません。</p> |
| CAPCTLLIB | <p>キャプチャー・スキーマ (キャプチャー・コントロール表が存在するライブラリーの名前) を指定します。</p> <p>ASN (デフォルト) キャプチャー・コントロール表が常駐するデフォルト・ライブラリーです。</p> <p><i>library-name</i> キャプチャー・コントロール表が常駐するライブラリーの名前。</p> |

表 50. STRDPRCAP コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| JRN | <p>最大 50 のジャーナルのサブセットをキャプチャー・プログラムが使用して作動するように指定します。キャプチャー・プログラムは、現在このジャーナルに記録されているすべてのソース表の処理を開始します。</p> <p>*ALL (デフォルト) キャプチャー・プログラムは、ソース表が記録されているジャーナルをすべて使用して作業を開始します。</p> <p><i>library-name/journal-name</i> キャプチャー・プログラムが使用して作動するジャーナルの修飾名。複数のジャーナルを入力するときには、スペースを使用してジャーナルを区切ってください。</p> |
| TRCLMT | <p>トレース限度を指定します (分単位)。キャプチャー・プログラムは、トレース限度よりも古いキャプチャー・トレース (IBMSNAP_CAPTRACE) 表の行をすべて除去します。デフォルトは 10,080 分 (トレース・エントリーが 7 日間) です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の TRACE_LIMIT 列の値を使用します。</p> <p><i>trace-limit</i> ブルーニング後に、トレース・データが IBMSNAP_CAPTRACE 表に保持される分数。</p> |
| MONLMT | <p>モニター限度を指定します (分単位)。キャプチャー・プログラムは、モニター限度よりも古いキャプチャー・モニター (IBMSNAP_CAPMON) 表の行をすべて除去します。デフォルトは 10,080 分 (モニター・エントリーが 7 日間) です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の MONITOR_LIMIT 列の値を使用します。</p> <p><i>monitor-limit</i> ブルーニング後に、モニター・データが IBMSNAP_CAPMON 表に保持される分数。</p> |
| MONITV | <p>キャプチャー・プログラムが、キャプチャー・モニター (IBMSNAP_CAPMON) 表に行を挿入する頻度 (秒単位) を指定します。デフォルトは 300 秒 (5 分) です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の MONITOR_INTERVAL 列の値を使用します。</p> <p><i>monitor-interval</i> IBMSNAP_CAPMON 表への行挿入の間隔の秒数。モニター・インターバルは、少なくとも 120 秒 (2 分) あげる必要があります。120 よりも小さい数値がユーザーから入力された場合、パラメーター値は 120 に設定されます。</p> |

表 50. STRDPRCAP コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|---|
| MEMLMT | <p>キャプチャー・ジャーナル・ジョブが使用できるメモリーの最大サイズ (MB 単位) を指定します。デフォルトは 32 MB です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の MEMORY_LIMIT 列の値を使用します。</p> <p><i>memory-limit</i> メモリーの最大値の MB 数。</p> |
| RETAIN | <p>データが除去されずに、変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表に保存される分数である、新規の保持制限を指定します。この値は、CLNUPITV パラメーターの値と共同で処理を行います。CLNUPITV の値に達すると、CD、UOW、IBMSNAP_SIGNAL、および IBMSNAP_AUTHTKN データのうちで、保持制限よりも古いデータが削除されます。</p> <p>表内のデータに矛盾が生じないように、データがこの RETAIN パラメーター値に達する前に変更情報がコピーされるようにアプライ・インターバルを設定してください。データの矛盾が発生した場合、アプライ・プログラムはフル・リフレッシュを実行します。</p> <p>デフォルトは 10,080 分 (7 日) です。最大値は、35000000 分です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の RETENTION_LIMIT 列の値を使用します。</p> <p><i>retention-limit</i> CD、UOW、IBMSNAP_SIGNAL、および IBMSNAP_AUTHTKN データが保存される分数。</p> |
| LAG | <p>キャプチャー・プログラムの処理が遅れても、その間は再始動が行われない分数である、新規の遅延限度を指定します。</p> <p>遅延限度に達すると (つまり、ジャーナル・エントリーのタイム・スタンプが現在の時間から遅延限度を引いたものより古くなると)、キャプチャー・プログラムは、そのジャーナルで処理中の表のコールド・スタートを開始します。アプライ・プログラムはその後フル・リフレッシュを実行し、キャプチャー・プログラムに新しい開始点を提供します。</p> <p>デフォルトは 10,080 分 (7 日) です。最大値は、35000000 分です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の LAG_LIMIT 列の値を使用します。</p> <p><i>lag-limit</i> キャプチャー・プログラムの遅れとして許容される分数。</p> |

表 50. STRDPRCAP コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義およびプロンプト |
|---------------|--|
| FRCFRQ | <p>キャプチャー・プログラムが変更データ (CD) 表および作業単位 (UOW) 表に変更を書き込む頻度 (30 秒から 600 秒) を指定します。キャプチャー・プログラムは、バッファがフルになるか、FRCFRQ 時間制限が満了するか、いずれか先に発生した時点で、これらの変更をアプライ・プログラムから使用可能にします。</p> <p>このパラメーターは、ソース表の変更率が少ないサーバー上で、アプライ・プログラムが変更をより早期に使用できるようにするために使用してください。FRCFRQ パラメーター値はグローバル値であり、すべての定義済みソース表で使用されます。FRCFRQ 値を低い数値に設定すると、システム・パフォーマンスが影響を受ける可能性があります。</p> <p>デフォルトは 30 秒です。</p> <p>*DFT (デフォルト) キャプチャー・プログラムは、キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表の COMMIT_INTERVAL 列の値を使用します。</p> <p><i>force-frequency</i> キャプチャー・プログラムが、アプライ・プログラムで変更を使用できるようにする前に、CD 表および UOW 表の変更をバッファ・スペース内に保持する秒数。</p> |

使用上の注意

STRDPRCAP コマンドの **CLNUPITV** パラメーターは、キャプチャー・プログラムが変更データ (CD) 表、作業単位 (UOW) 表、シグナル (IBMSNAP_SIGNAL) 表、キャプチャー・モニター (IBMSNAP_CAPMON) 表、キャプチャー・トレース (IBMSNAP_CAPTRACE) 表、およびアプライ修飾子相互参照 (IBMSNAP_AUTHTKN) 表から古いレコードを除去するまでに待つ最大時間数を指定します。

STRDPRCAP コマンドは手動で実行することができます。また、このコマンドは、初期プログラム・ロード (IPL 始動プログラム) の一部として、自動的に実行することができます。

JOB パラメーターで指定されたジョブ記述がジョブ・キュー QDP4/QZSNDPR を使用する場合に、DB2 DataPropagator for iSeries サブシステムがアクティブでない場合は、**STRDPRCAP** コマンドはサブシステムを始動します。ジョブ記述が別のジョブ・キューおよびサブシステムを使用するよう定義されている場合には、**STRDPRCAP** コマンドの実行前または後のいずれかに、サブシステムの始動 (**STRSBS**) コマンドを使用して、このサブシステムを手動で始動する必要があります。

STRSBS QDP4/QZSNDPR

システムの QSTRUPPGM システム値で参照されるプログラムに **STRSBS** コマンドを追加することにより、サブシステムを自動的に開始するようシステムを設定することができます。

ウォーム・スタートまたはコールド・スタートを使用したキャプチャー・プログラムの再始動

STRDPRCAP コマンドに対する **RESTART** パラメーターの値が、キャプチャー・プログラムがウォーム・スタートとコールド・スタートを処理する方法をコントロールします。

ウォーム・スタート処理: ほとんどの場合、ウォーム・スタート情報は保管されます。時々、ウォーム・スタート情報が保管されないことがあります。そのような場合、キャプチャー・プログラムは、CD 表、UOW 表、またはブルーニング・コントロール (IBMSNAP_PRUNCNTL) 表を使用して、それが停止した時刻に再同期します。

自動コールド・スタート: 場合によっては、ウォーム・スタートが指定されていても、キャプチャー・プログラムは自動的にコールド・スタートに切り替えます。OS/400 システムでは、コールド・スタートはジャーナルによるジャーナル方式で動作します。たとえば、ジャーナルがラグの制限を超える場合、そのジャーナルを使用するすべてのレプリケーション・ソースは、別のジャーナルを使用するレプリケーション・ソースがコールド・スタートでなくても、コールド・スタートで始動します。

キャプチャー・プログラムが様々なタイプのジャーナル・エントリーを処理する方法について詳しくは、661 ページの表 100 を参照してください。

STRDPRCAP の例

次の例は、**STRDPRCAP** コマンドの使用方法を示しています。

例 1

2 つの異なるジャーナルに関して、キャプチャー・プログラムのウォーム・スタートを開始するには、次のようにします。

```
STRDPRCAP RESTART(*YES) JRN(HR/QSQJRN ACCTS/QSQJRN)
```

例 2

1 つの指定されたジャーナルに関してキャプチャー・プログラムを始動するには、次のようにします。

```
STRDPRCAP CAPCTLLIB(BSN) JRN(MARKETING/QSQJRN)
```

キャプチャー・コントロール表は BSN という名前のライブラリーにあります。

例 3

2 つのジャーナルのブルーニングを行わないでキャプチャー・プログラムを始動するには、次のようにします。

```
STRDPRCAP RESTART(*YES) CLNUPITV(*DFT *NO) JRN(HR/QSQJRN ACCTS/QSQJRN)
```

例 4

デフォルトのキャプチャー・コントロール・ライブラリーにある、1 つの指定されたジャーナルについてキャプチャー・プログラムを始動し、トレース限度プルーニング、モニター限度プルーニング、IBMSNAP_CAPMON 表挿入、およびメモリー限度のデフォルト・パラメーターを変更するには、次のようにします。

```
STRDPRCAP CAPCTLLIB(ASN) JRN(SALES/QSQJRN) TRCLMT(1440) MONLMT(1440)
MONITV(3600) MEMLMT(64)
```

例 5

キャプチャー・プログラムのコールド・スタートを開始するには、次のようにします。

```
STRDPRCAP RESTART(*NO)
```

例 6

すべてのデフォルト値を使用してキャプチャー・プログラムを始動するには、次のようにします。

```
STRDPRCAP
```

関連作業:

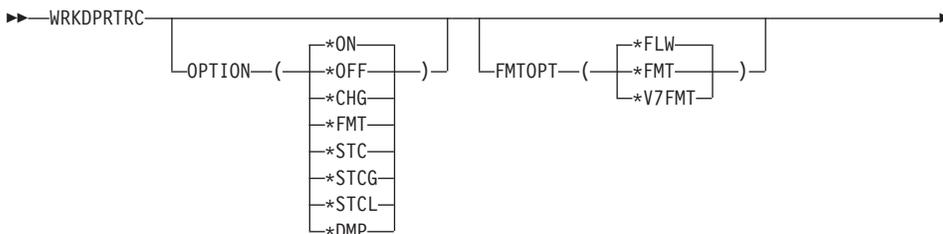
- 129 ページの 『第 9 章 キャプチャー・プログラムの操作』

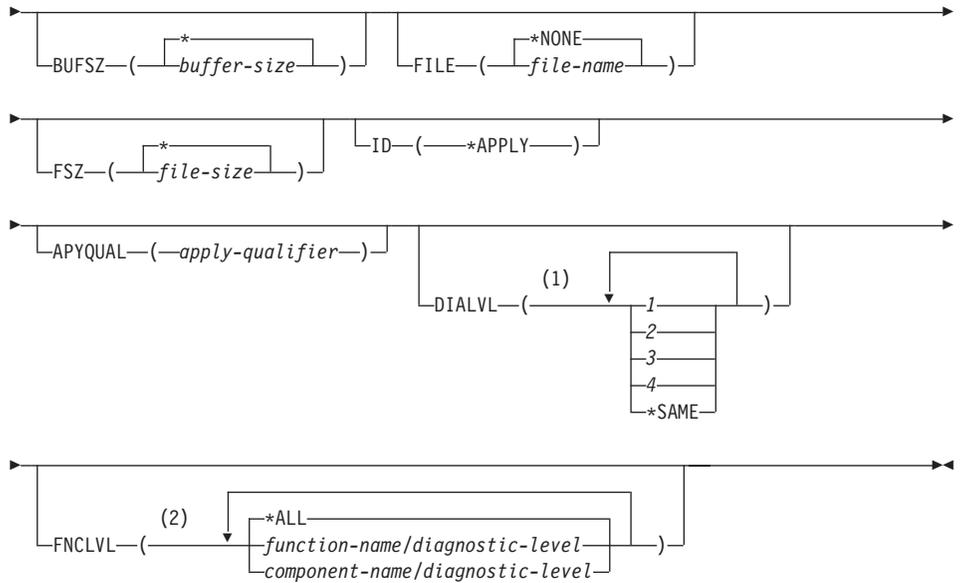
WRKDPRTRC: DPR トレース機能の使用法 (OS/400)

トレース機能を実行するには、DPR トレース (**WRKDPRTRC**) コマンドを実行します。トレース機能は、指定されたアプライ・プログラムのプログラム・フロー情報を記録します。このトレース情報を IBM ソフトウェア・サポートに提供して、トラブルシューティングに役立てることができます。

コマンド行にコマンド名を入力したら、F4 キーを押してコマンド構文を表示することができます。

このコマンド、およびこのコマンドのすべてのパラメーターの詳しい記述を表示するには、画面の一番上のコマンドにカーソルを移動し、F1 キーを押します。特定のパラメーターの記述を表示するには、そのパラメーター上にカーソルを移動し、F1 キーを押します。

WRKDPRTRC コマンドを使用して DPR トレース機能を実行する方法

**注:**

- 1 複数の値を指定できます。
- 2 最大 20 の関数またはコンポーネントを指定できます。

表 51. WRKDPRTTC コマンド・パラメーター定義 (OS/400 用)

| パラメーター | 定義 |
|---------------|---|
| OPTION | <p>トレース関数を 1 つ指定します。</p> <p>*ON (デフォルト) トレース機能をオンにします。このオプションは、トレース用の共有メモリー・セグメントを自動的に作成します。</p> <p>*OFF トレース機能をオフにします。</p> <p>*CHG トレース機能パラメーターの値を変更します。</p> <p>*FMT 共有メモリーからのトレース機能の出力をフォーマットします。</p> <p>*STC トレース機能の状況を表示します。この状況情報には、トレース・バージョン、アプリケーション・バージョン、エントリー数、バッファー・サイズ、使用中のバッファー量、状況コード、およびプログラム・タイム・スタンプが含まれます。</p> <p>このパラメーター・オプションは、UNIX、Windows および z/OS オペレーティング・システムで使用される asnlrc コマンドの stat オプションと同等です。</p> <p>*STCG レプリケーション・センターで読み取れるフォーマットでトレース機能の状況を表示します。</p> <p>*STCL 追加のバージョン・レベル情報を含めてトレース機能の状況を表示します。この追加情報には、アプリケーション内の各モジュールのサービス・レベルが含まれ、長ストリングのテキストとして表示されます。</p> <p>このパラメーター・オプションは、UNIX、Windows および z/OS オペレーティング・システムで使用される asnlrc コマンドの statlong オプションと同等です。</p> <p>*DMP トレース・バッファーの現在の内容をファイルに書き込みます。</p> <p>WRKDPRTTC コマンドのプロンプトから F4 キーを押すと、トレース・オプションのリストを表示できます。</p> |

表 51. WRKDPRTTC コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義 |
|---------------|---|
| FMTOPT | <p>フォーマット ID のオプションを指定します。 OPTION(*FMT) パラメーターと一緒に使用されます。</p> <p>*FLW (デフォルト) 関数呼び出しのフローを表示します。</p> <p>*FMT トレース・バッファーまたはトレース・ファイルのフォーマットを表示します。すべての詳細データが表示されます。</p> <p>*V7FMT トレース・バッファーまたはトレース・ファイル情報をバージョン 7 の形式にフォーマットします。</p> <p>WRKDPRTTC コマンドのプロンプトから F4 キーを押すと、フォーマット・オプションのリストを表示できます。</p> |
| BUFSZ | <p>トレース・バッファーのサイズを指定します (バイト単位)。数字の後ろに M、K、または G を入力すれば、メガバイト、キロバイト、またはギガバイトをそれぞれ指定できます。</p> <p>デフォルトは 2 メガバイトです。</p> <p>* (デフォルト) 2 MB のデフォルト・サイズが使用されます。</p> <p><i>buffer-size</i> バッファー・サイズ (バイト単位)。</p> |
| FILE | <p>トレース出力をファイルに書き込むかどうかを指定します。</p> <p>*NONE (デフォルト) トレース出力は共有メモリーにのみ入れられます。</p> <p><i>file-name</i> 出力ファイルの名前。 OPTION(*DMP) パラメーターが使用された場合、このファイル名はダンプ・ファイルの名前を表します。</p> |
| FSZ | <p>トレース・データが保管されるファイルのサイズを指定します (バイト単位)。数字の後ろに M、K、または G を入力すれば、メガバイト、キロバイト、またはギガバイトをそれぞれ指定できます。</p> <p>デフォルトは 2 GB です。</p> <p>* (デフォルト) 2 GB のデフォルト・サイズが使用されます。</p> <p><i>file-size</i> ファイル・サイズ (バイト単位)。</p> |
| ID | <p>トレース対象のプログラムのタイプを指定します。</p> <p>*APPLY (デフォルト) アプライ・プログラムのトレース。</p> |

表 51. WRKDPTRTC コマンド・パラメーター定義 (OS/400 用) (続き)

| パラメーター | 定義 |
|----------------|--|
| APYQUAL | <p>トレースするアプライ・プログラムの名前を指定します。</p> <p><i>apply-qualifier</i> アプライ修飾子の名前。</p> |
| DIALVL | <p>トレース機能により記録されるトレース・レコードのタイプを指定します。トレース・レコードは、以下の診断マスク番号により分類されます。</p> <p>1 フロー・データ。関数の入力点と出口点が含まれます。</p> <p>2 基本データ。トレース機能が検出したすべての主要なイベントが含まれます。</p> <p>3 詳細データ。主要なイベントとその記述が含まれます。</p> <p>4 パフォーマンス・データ。</p> <p>*SAME このコマンドは、直前のトレース機能で使用された診断レベル設定値を使用します。</p> <p>1 つ、または複数の診断マスク番号を入力できます。番号は昇順に入力する必要があります。数字間にスペースを入れないでください。</p> <p>重要: 番号レベルは包括的ではありません。それぞれが独立したレベルを表しています。</p> <p>トレース機能の開始時のデフォルトは DIALVL(1234) です。その後、にトレース機能呼び出した場合のデフォルトは *SAME です。</p> <p>WRKDPTRTC コマンドのプロンプトから F4 キーを押すと、使用可能な診断レベルのリストを表示できます。</p> |
| FNCLVL | <p>特定の関数またはコンポーネント ID をトレースするかどうかを指定します。</p> <p>*ALL (デフォルト) すべての関数およびコンポーネントがトレース関数に含まれます。</p> <p><i>function-name/diagnostic-level</i> トレースする関数の名前と、対応する診断マスク番号。</p> <p><i>component-name/diagnostic-level</i> トレースするコンポーネントの名前と、対応する診断マスク番号。</p> <p>最大 20 の関数またはコンポーネント名を入力できます。</p> |

WRKDPTRTC の例

次の例は、**WRKDPTRTC** コマンドの使用方法を示しています。

例 1

すべての関数およびコンポーネントについて、アプライ修飾子 AQ1 のアプライ・トレースを開始し、出力を TRCFILE という名前のファイルに書き込むには、次のようにします。

```
WRKDPRTTRC OPTION(*ON) FILE(TRCFILE) ID(*APPLY) APYQUAL(AQ1)
```

例 2

アプライ修飾子 AQ1 のアプライ・トレースを終了するには、次のようにします。

```
WRKDPRTTRC OPTION(*OFF) ID(*APPLY) APYQUAL(AQ1)
```

例 3

すべての関数およびコンポーネントについて、アプライ修飾子 AQ1 のアプライ・トレースを診断レベル 3 および 4 (詳細データおよびパフォーマンス・データ) に変更するには、次のようにします。

```
WRKDPRTTRC OPTION(*CHG) ID(*APPLY) APYQUAL(AQ1) DIALVL(34)
```

例 4

アプライ修飾子 AQ1 のアプライ・トレースの状況を表示するには、次のようにします。

```
WRKDPRTTRC OPTION(*STC) ID(*APPLY) APYQUAL(AQ1)
```

例 5

アプライ修飾子 AQ1 の関数呼び出しを診断レベル 3 および 4 で表示するには、次のようにします。

```
WRKDPRTTRC OPTION(*FMT) FMTOPT(*FLW) ID(*APPLY) APYQUAL(AQ1) DIALVL (34)
```

例 6

アプライ修飾子 AQ1 のアプライ・トレース情報を DMPFILE という名前のダンプ・ファイルに書き込むには、次のようにします。

```
WRKDPRTTRC OPTION(*DMP) FILE(DMPFILE) ID(*APPLY) APYQUAL(AQ1)
```

関連資料:

- 358 ページの 『asnlrc: レプリケーション・トレース機能の操作 (UNIX、Windows、z/OS)』

第 19 章 レプリケーション・プログラムの操作 (z/OS)

この章は、以下のセクションで構成されています。

- 『レプリケーション・プログラムの操作に JCL またはシステム開始タスクを使用する方法 (z/OS)』
- 463 ページの『レプリケーション・プログラムを自動的に再始動するために MVS 自動リスタート・マネージャー (ARM) を使用する方法 (z/OS)』
- 464 ページの『データ共有モードへのレプリケーション環境のマイグレーション (z/OS)』

レプリケーション・プログラムの操作に JCL またはシステム開始タスクを使用する方法 (z/OS)

z/OS では、JCL を使用するか、システム開始タスクとして、キャプチャー・プログラム、アプライ・プログラムおよびレプリケーション・アラート・モニターを操作できます。

JCL を使用したレプリケーション・プログラムの操作

このセクションでは、JCL を使用して、キャプチャー・プログラム、アプライ・プログラムおよびレプリケーション・アラート・モニターを操作する方法を説明します。

サンプルの JCL はプログラム・ディレクトリーにあります。

JCL を使用して z/OS でキャプチャー・プログラムを始動する方法

1. ライブラリー SASNLJCL(ASNL2RN#) にある JCL をカスタマイズします。この # は、DB2 for z/OS のレベルです (たとえば、バージョン 7 の場合は、# を 7 という数表示で置き換えます)。キャプチャー・ジョブの PARM フィールドに、適切なオプションの呼び出しパラメーターを指定することにより、z/OS 用の JCL を準備します。バッチ・モードで実行する場合は、JCL で時間帯および言語環境変数も設定できます。

次に示す呼び出し JCL の行の例では、TZ および LANG 変数を設定しています。

```
//CAPJFA EXEC PGM=ASNCAP, PARM='ENVAR('TZ=PST8PDT', 'LANG=en_US')/  
DSN6 cold capture_schema=JFA autostop'
```

2. TSO または MVS コンソールから JCL をサブミットします。

JCL を使用して z/OS 上でアプライ・プログラムを始動する方法

アプライ・ジョブの PARM フィールドに適切な呼び出しパラメーターを指定することにより、z/OS 用の JCL を準備します。サイトの要件に合うように、JCL を調整しま

す。ライブラリー SASNAJCL(ASNA2RN#) にある呼び出し JCL は、Apply for z/OS 製品に含まれています。この # は、DB2 for z/OS のレベルです (たとえば、バージョン 7 の場合は、# を 7 という数表示で置き換えます)。

呼び出し JCL でのこの行の例を次に示します。

```
//apyasn EXEC PGM=ASNAPPLY,PARM='control_server=CTLDB1
                                apply_qual=myqual spillfile=disk'
```

JCL を使用して z/OS 上でレプリケーション・アラート・モニターを開始する方法

レプリケーション・アラート・モニター・ジョブの PARM フィールドで適切な呼び出しパラメーターを指定して、z/OS 用の JCL を準備します。サイトの要件に合うように、JCL を調整します。ライブラリー SASNMJCL(ASNMON#) にある呼び出し JCL は、レプリケーション・アラート・モニター (z/OS 版) 製品に含まれています。

呼び出し JCL でのこの行の例を次に示します。

```
//monasn EXEC PGM=ASNMON,PARM='monitor_server=MONDB1
                                monitor_qualifier=monqual'
```

JCL を使用して z/OS 上の開始済みプログラムを変更する方法

キャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニターを開始した後で、MODIFY コマンドを使用して、プログラムを停止したり、関連するタスクを実行することができます。MODIFY コマンドは MVS コンソールから実行する必要があります。次の構文例に示すように、F という省略形を使用できます。

(1)

▶—F—*jobname*—, —| パラメーター |————▶

注:

- 1 パラメーターの記述は、323 ページの『第 17 章 レプリケーション用のシステム・コマンド (UNIX、Windows、z/OS)』を参照してください。

基本的に、F *jobname* , は、**asnacmd**、**asnccmd**、または **asnmcmd** など、実際のコマンド名を置き換えます。たとえば、キャプチャー・プログラムを停止するには、次のコマンドを使用します。

```
F capjfa,stop
```

MODIFY の情報については、「z/OS MVS System Commands」(SA22-7627) を参照してください。

システム開始タスクを使用してレプリケーション・プログラムを操作する方法

このセクションでは、システム開始タスクを使用して、キャプチャー・プログラム、アプライ・プログラムおよびレプリケーション・アラート・モニターを操作する方法を説明します。

Capture for z/OS プログラムをシステム開始タスクとして開始するためのセットアップ

1. PROCLIB にプロシージャー (*procname*) を作成します。このプロシージャーには、キャプチャー・プログラムを実行するのに必要な JCL (たとえば、SASNLJCL ライブラリーの ASNL2RN6、ASNL2RN7、または ASNL2RN8) が入っています。
2. *procname* について、RACF STARTED クラス内にエントリーを作成します。このエントリーは、*procname* を、キャプチャー・プログラムを始動するために使用される RACF ユーザー ID と関連付けます。キャプチャー・プログラムを始動する前に、このユーザー ID に必要な DB2 許可が付与されるようにしてください。
3. MVS システム・コンソールから、**start *procname*** を実行します。

レプリケーション・プログラムを自動的に再始動するために MVS 自動リスタート・マネージャー (ARM) を使用する方法 (z/OS)

キャプチャー・プログラム、アプライ・プログラムおよびレプリケーション・アラート・モニターと一緒に、自動リスタート・マネージャー (ARM) を使用できます。ARM は、特定のバッチ・ジョブまたは開始タスクの可用性を改善するための MVS リカバリー関数です。ジョブまたはタスクが失敗するか、ジョブやタスクを実行しているシステムに障害が発生した場合、ARM はオペレーターの介入なしに、ジョブまたはタスクを再始動できます。ARM は、処理対象のアプリケーションを識別するためにエレメント名を使用します。MVS ARM が使用可能な各アプリケーションは、自分自身について固有なエレメント名を生成し、ARM とのすべての連絡にこの名前を使用します。ARM はエレメント名をトラッキングし、エレメント名に対して再始動ポリシーを定義します。ARM の設定についての詳細は、「z/OS MVS Programming Sysplex Services Guide」(SA22-7617) を参照してください。

前提条件:

ARM がインストール済みであり、レプリケーション・プログラムが正しく設定されていることを確認してください。レプリケーション・プログラムと一緒に ARM を使用する場合は、レプリケーション・プログラムが APF 許可であることを確認してください。たとえば、アプライ・プログラムまたはレプリケーション・アラート・モニターで ARM を使用する場合は、適切なロード・モジュールを APF 許可ライブラリーにコピーする必要があります。(キャプチャー・プログラムは、ARM を使用するかしないかに関係なく、必ず APF 許可にする必要があります。)

ARM を構成するときには、レプリケーション・プログラムに次のエレメント名を使用してください。

キャプチャー・プログラム

ASNTCxxxxyyyy

アプライ・プログラム

ASNTAxxxxyyyy

レプリケーション・アラート・モニター

ASNAMxxxxyyyy

この xxxx は DB2 サブシステム名であり、yyyy はデータ共有メンバー名です (後者はデータ共有構成の場合にのみ必要)。エレメント名の長さは常に 16 文字であり、ブランクが埋め込まれます。エレメント名は、シスプレックス全体を通じて固有のものにする必要があります。このため、ARM を使用するには、1 つのサブシステムにつき、特定のプログラムのインスタンスは 1 つしか実行できません。

レプリケーション・プログラムは初期化時にエレメント名を使用して ARM に登録されます。登録時にイベント出口を ARM に提供することはありません。(レプリケーション・プログラムは MVS サブシステムとして実行されるのではないため、イベント出口は必要ありません。) 登録済みプログラムが異常終了した場合 (たとえばセグメント違反の発生)、ARM はユーザーの代わりに登録済みプログラムを再始動します。登録されたレプリケーション・プログラムは、通常終了した場合 (たとえば、STOP コマンドによる終了)、または無効な登録を検出した場合は、登録解除されます。

ヒント: パラメーター NOTERM=Y を指定してキャプチャー・プログラムまたはアプライ・プログラムを始動すると、DB2 が静止してもプログラムは停止しません。この場合、プログラムは ARM から登録解除されません。プログラムは実行を続けますが、DB2 が再始動されるまでは、データをキャプチャーしません。

データ共有モードへのレプリケーション環境のマイグレーション (z/OS)

キャプチャー・プログラムが非データ共有モードで実行されているときに、システムをデータ共有モードにマイグレーションするときには、plexify ユーティリティ (ASNPLXFY) を一度実行する必要があります。キャプチャー・プログラムが正しい LRSN から開始するように、キャプチャー・プログラムをウォーム・スタートする前に、このユーティリティを実行してください。このユーティリティは、再始動 (IBMSNAP_RESTART) 表の中のデータをマイグレーションします。非データ共有のログ・シーケンス番号 (RBA) は、データ共有環境での同等のシーケンス番号 (LRSN) に変換されます。

前提条件:

キャプチャー・プログラムを実行するために使用したものと同一ユーザー ID を使用するか、同じ特権を持つものを使用します。ASNPLXFY ユーティリティが APF 許可

であることを確認してください。ASNPLXFY プランは、サブシステムにバインドする必要があります。また、サブシステムはデータ共有モードで実行されている必要があります。このユーティリティーのバインディングの詳細は、プログラム・ディレクトリーを参照してください。

プロシージャー:

USS データ共有環境で **ASNPLXFY** ユーティリティーを実行するには、次のようにします。

1. キャプチャー・プログラムを停止してください。
2. コマンド行から次のコマンドを入力します。

```
ASNPLXFY yoursubsystem captureschema
```

サブシステムの名前が必要です。キャプチャー・スキーマはオプションです。デフォルトのキャプチャー・スキーマは **ASN** です。

3. キャプチャー・プログラムをウォーム・スタートします。

第 20 章 Windows Service Control Manager を使用してシステム・コマンドを発行する方法 (Windows)

このセクションでは、Windows オペレーティング・システムの場合に、レプリケーション・プログラムを始動するサービスを作成する方法を説明します。キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、およびモニター・コントロール・サーバーごとにサービスを作成できます。サービスは他の DB2 サービスとグループ化されます。サービスを開始した後で、プログラムのパラメーターを変更する場合は、サービスをドロップして、新しいサービスを作成する必要があります。

- 『レプリケーション・サービスの作成』
- 469 ページの『レプリケーション・サービスの操作』
- 469 ページの『レプリケーション・サービスのドロップ』

レプリケーション・サービスの作成

レプリケーション・サービスを作成する前に、DB2 インスタンス・サービスが実行されていることを確認してください。レプリケーション・サービスを作成するときに DB2 インスタンス・サービスが実行されていないと、レプリケーション・サービスは作成されませんが、自動的に開始されなくなります。

次の方法のいずれかを使用して、レプリケーション・サービスを作成します。

- レプリケーション・センターで始動するプログラムの始動パラメーターを指定し、「**Windows サービスとして *program_name* プログラムを開始**」を選択します。詳細については、レプリケーション・センターのヘルプを参照してください。
- **asnsrct** コマンドを使用します。コマンド構文およびパラメーター記述の詳細については、355 ページの『asnsrct: キャプチャー、アプライ、またはレプリケーション・アラート・モニターを開始する DB2 レプリケーション・サービスの作成 (Windows のみ)』を参照してください。

ヒント: レプリケーション・サービスが正しくセットアップされた場合は、サービスが正常に開始された後、サービス名が **STDOUT** に送信されます。サービスが開始されない場合は、開始しようとしているプログラムのログ・ファイルを調べてください。デフォルトでは、ログ・ファイルは **DB2PATH** 環境変数で指定されたディレクトリー内にあります。サービスとして開始するプログラムのパス・パラメーターを指定して (**capture_path**、**apply_path**、**monitor_path**)、このデフォルトをオーバーライドできます。また、Windows Service Control Manager (SCM) を使用して、サービスの状況を表示できます。

サービスを作成するときには、Windows にログオンするために使用するアカウント名と、そのアカウント名のパスワードを指定する必要があります。

システムには複数のレプリケーション・サービスを追加できます。すべてのキャプチャー・サーバーのスキーマごとにサービスを追加し、すべてのアプライ・コントロール・サーバーおよびモニター・コントロール・サーバーの修飾子ごとにサービスを追加できます。たとえば、5つのデータベースがあり、各データベースがアプライ・コントロール・サーバー、キャプチャー・コントロール・サーバー、およびモニター・コントロール・サーバーである場合は、15のレプリケーション・サービスを作成できます。各サーバーに複数のスキーマまたは修飾子がある場合は、さらに多くのサービスを作成できません。

ユーザーが作成したレプリケーション・サービスは、自動モードで SCM に追加され、サービスが開始されます。Windows は固有のサービス名および表示名を使用してサービスを登録します。

レプリケーション・サービス名

レプリケーション・サービス名は、各サービスを一意的に識別します。サービスを停止または開始するときにはこれを使用します。名前は次のようなフォーマットになります。

`DB2.instance.alias.program.qualifier_or_schema`

ここで、

- *instance* は DB2 インスタンスの名前です。
- *alias* は、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、またはモニター・コントロール・サーバーのデータベース別名です。
- *program* は次の値のいずれかです。CAP (キャプチャー・プログラム)、APP (アプライ・プログラム)、または MON (レプリケーション・アラート・モニター・プログラム)
- *qualifier_or_schema* は、アプライ修飾子、モニター修飾子、またはキャプチャー・スキーマという ID のいずれかです。

インスタンス: 次のサービス名は、スキーマ ASN を持ち、INST1 という名前のインスタンスの下でデータベース DB1 を処理しているキャプチャー・プログラムのものです。

`DB2.INST1.DB1.CAP.ASN`

レプリケーション・サービスの表示名

表示名は、「サービス」ウィンドウで表示されるテキスト・ストリングで、読みやすい形式のサービス名です。たとえば、次のようになります。

`DB2 - INST1 DB1 CAPTURE ASN`

サービスの記述を追加する場合は、レプリケーション・サービスを作成した後で、Service Control Manager (SCM) を使用してください。また、SCM を使用し、サービスのユーザー名およびパスワードを指定することもできます。

レプリケーション・サービスの操作

レプリケーション・サービスを作成したら、サービスを停止してから、再度開始できます。

次の方法のいずれかを使用して、サービスを停止します。

- SCM
- **net stop** コマンド

重要: レプリケーション・サービスを停止すると、サービスに関連するプログラムは自動的に停止します。しかし、レプリケーション・システム・コマンド (**asnacmd**、**asnccmd**、または **asnmcmd**) を使用してプログラムを停止すると、プログラムを始動するために使用されたサービスは、ユーザーが明示的に停止するまで、実行を続けます。

レプリケーション・コマンドのサービスを開始するには、次の方法のいずれかを使用します。

- SCM
- **net start** コマンド
- レプリケーション・センター

重要: サービスからレプリケーション・プログラムを始動した場合、同じスキーマまたは修飾子を使用してプログラムを始動しようとすると、エラーが発生します。

使用している Windows オペレーティング・システムによっては、レプリケーション・サービスの操作をユーザーが変更できます。

- Windows 2000 の場合は、SCM を使用して、レプリケーション・サービスを休止、再開、および再始動できます。
- Windows NT の場合は、SCM を使用して、レプリケーション・サービスの休止、または続行ができます。

レプリケーション・サービスのドロップ

レプリケーション・サービスが必要なくなった場合は、SCM からドロップされるように、サービスをドロップできます。また、サービスによって開始されるプログラムの始動パラメーターを変更する場合は、サービスをドロップして、新しい始動パラメーターを使用する新しいサービスを作成する必要があります。

レプリケーション・コマンドのサービスをドロップするには、**asnsdrop** コマンドを使用します。

関連資料:

- 355 ページの 『asnsrct: キャプチャー、アプライ、またはレプリケーション・アラート・モニターを開始する DB2 レプリケーション・サービスの作成 (Windows のみ)』
- 357 ページの 『asnsdrop: DB2 レプリケーション・サービスのドロップ (Windows のみ)』

第 21 章 各種オペレーティング・システムでのレプリケーション・プログラムのスケジューリング

オペレーティング・システム (OS) のコマンドを使用してあらかじめ指定しておいた時刻にキャプチャー・プログラム、アプライ・プログラム、またはレプリケーション・アラート・モニター・プログラムが始動するようにスケジューリングできます。

UNIX オペレーティング・システムでのプログラムのスケジューリング

プログラムを特定の時刻に開始するには **at** コマンドを使用します。たとえば、次のコマンドを使用すると、金曜日の午後 3 時にプログラムが開始されます。

キャプチャー・プログラムのスケジューリング:

```
at 3pm Friday asncap autoprune=n
```

アプライ・プログラムのスケジューリング:

```
at 3pm Friday asnapply applyqual=myqual
```

レプリケーション・アラート・モニター・プログラムのスケジューリング:

```
at 3pm Friday asnmon monitor_server=db2srv1 monitor_qualifier=mymon
```

Windows オペレーティング・システムでのプログラムのスケジューリング

Windows Service Control Manager を使用していない場合は、**AT** コマンドを使用してこのプログラムを特定の時刻に開始できます。**AT** コマンドを入力する前に、Windows スケジュール・サービスを始動してください。

以下の例では、各プログラムは 15 時 (午後 3 時) に始動されます。

キャプチャー・プログラムのスケジューリング:

```
c:¥>AT 15:00 /interactive "c:¥SQLLIB¥BIN¥db2cmd.exe c:¥CAPTURE¥asncap.exe"
```

アプライ・プログラムのスケジューリング:

```
c:¥>AT 15:00 /interactive "c:¥SQLLIB¥BIN¥db2cmd.exe  
c:¥SQLLIB¥BIN¥asnapply.exe control_server=cntldb apply_qual=qualid1"
```

レプリケーション・アラート・モニター・プログラムのスケジューリング:

```
c:¥>AT 15:00 /interactive "c:¥SQLLIB¥BIN¥db2cmd.exe  
c:¥CAPTURE¥asnmon.exe monitor_server=db2srv1 monitor_qualifier=mymon"
```

z/OS オペレーティング・システムでのプログラムのスケジューリング

指定した時点で Capture for z/OS を始動するには、**\$TA JES2** コマンドまたは **AT NetView** コマンドのいずれかを使用します。

z/OS 上でプログラムをスケジューリングする場合

1. z/OS 用のプログラムを呼び出すプロシージャを PROCLIB に作成する。
2. ICHRIN03 RACF モジュール (または、MVS セキュリティー・パッケージの該当する定義) を修正し、プロシージャをユーザー ID に関連付けます。
3. SYS1.LPALIB でモジュールをリンク・エディットします。

\$TA JES2 コマンドの使用については、「MVS/ESA JES2 コマンド」を参照してください。**AT NetView** コマンドの使用については、「NetView (MVS) コマンド・リファレンス」を参照してください。

OS/400 オペレーティング・システムでのプログラムのスケジューリング

特定の時刻にアプライ・プログラムを始動するには、**ADDJOBSCDE** コマンドを使用します。

SBMJOB コマンドを使用して、OS/400 でのキャプチャー・プログラムの始動をスケジューリングします。

```
SBMJOB CMD('STRDPRCAP...')SCDDATE(...)SCDTIME(...)
```

第 22 章 DB2 レプリケーション・コンポーネントの通信方法

レプリケーション・コンポーネントはお互いに独立して実行され、各コンポーネントがコントロール表に保管する情報を頼りにお互いに通信します。DB2[®] レプリケーションには次の 4 つのコンポーネントがあります。

- レプリケーション・センター
- キャプチャー・プログラムまたはトリガー
- アプライ・プログラム
- レプリケーション・アラート・モニター

レプリケーション・センターは、登録されたソース、サブスクリプション・セット、およびアラート条件についての初期情報をコントロール表に保管します。キャプチャー・プログラム、アプライ・プログラム、およびキャプチャー・トリガーは、コントロール表を更新することにより、レプリケーションの進行状況を示し、変更の処理を調整します。レプリケーション・アラート・モニターは、キャプチャー・プログラム、アプライ・プログラム、およびキャプチャー・トリガーにより更新されたコントロール表を読み取り、サーバーでの問題や進行状況を理解します。

レプリケーション・センター、キャプチャー・プログラムまたはトリガー、およびアプライ・プログラム

レプリケーション・ソースとして、表、ビュー、またはニックネームを登録すると、レプリケーション・センターは SQL スクリプトを作成します。この SQL スクリプトは、このソースの情報を、すべての登録情報を含む登録表 (IBMSNAP_REGISTER) が入ったレプリケーション・コントロール表に保管します。レプリケーション・センターにより生成される SQL スクリプトは、登録されたソース用の CD 表も作成します。

IBMSNAP_REGISTER 表には、登録されたソースごとに 1 つの行があり、また、登録されたビュー内の基礎表ごとに 1 つの行があります。この表は、登録されたそれぞれのソースについて、次の種類の情報を含んでいます。

- ソース表のスキーマ名および名前
- 登録されたそれぞれのソース表の構造タイプ
- CD 表のスキーマ名および名前
- 登録されたビューの場合、このビュー内の基礎表の CD 表の名前 (基礎表が登録されている場合)
- 内部 CCD 表がある場合は、そのスキーマ名および名前
- update-anywhere ソースの競合検出レベル

キャプチャー・プログラムとアプライ・プログラムは、IBMSNAP_REGISTER 表の情報を使用して、自分の該当する状況をお互いに知らせます。この表は関連する情報用にいくつかの列を持っています。この表についての詳細は、505 ページの『schema.IBMSNAP_REGISTER』を参照してください。

OS/400® ソースの場合、リモート側でジャーナルに記録された表を含め、IBMSNAP_REGISTER 表への拡張があり、IBMSNAP_REG_EXT には、iSeries™ システムに固有の追加の情報 (ジャーナル・ライブラリーやジャーナル名など) が含まれません。

サブスクリプション・セットを作成し、これにメンバーを追加すると、レプリケーション・センターは SQL スクリプトを作成します。このスクリプトは、このサブスクリプション・セット用の情報を複数のレプリケーション・コントロール表に保管します。これらの表にはすべてのサブスクリプション・セット情報が含まれ、その内訳は、サブスクリプション・セット表 (IBMSNAP_SUBS_SET)、サブスクリプション・セット・メンバー表 (IBMSNAP_SUBS_MEMBR)、サブスクリプション・セット列の表 (IBMSNAP_SUBS_COLS)、およびサブスクリプション・セット・ステートメント表 (IBMSNAP_SUBS_STMTS) です。レプリケーション・センターにより生成される SQL スクリプトは、ターゲット表も作成します (まだ存在しない場合)。

メインのサブスクリプション・セット表である IBMSNAP_SUBS_SET には、それぞれのサブスクリプション・セットごとに 1 つの行があります。この表は、それぞれのサブスクリプション・セットについて、次の種類の情報を含んでいます。

- アプライ修飾子
- サブスクリプション・セットの名前
- サブスクリプション・セットのタイプ: 読み取り専用か、または読み取り / 書き込み (update-anywhere)
- ソースおよびターゲット・データベースの名前と別名
- サブスクリプション・セットを処理するタイミング
- サブスクリプション・セットの現在の状況

この表は関連する情報用にいくつかの列を持っています。この表についての詳細は、528 ページの『ASN.IBMSNAP_SUBS_SET』を参照してください。

その他のサブスクリプション・セット表には、サブスクリプション・セットのメンバー、列、およびセットを使用して処理される SQL ステートメント (またはストアド・プロシージャ) についての情報が含まれます。

キャプチャー・プログラムおよびアプライ・プログラム

キャプチャー・プログラムは、レプリケーション・コントロール表のいくつかを使用して、ソース・データベースにどのような変更がなされたかを示し、アプライ・プログラムはそれらのコントロール表の値を使用して、ターゲット・データベースに何をコピーする必要があるかを検出します。キャプチャー・プログラムは、アプライ・プログラムから指示を受けない限り何も情報を取り込まず、また、アプライ・プログラムは、レプリケーション・ソースおよびそれに関連するサブスクリプション・セットが定義されるまで、キャプチャー・プログラムに変更のキャプチャーを開始する指示を出しません。

以下に示す処理は、データ保全性を保つために、通常のレプリケーション・シナリオでは、アプライ・プログラムとキャプチャー・プログラムがどのように通信するかを説明しています。

ソース・データベースからデータを取り込む

1. キャプチャー・プログラムは始動時に `IBMSNAP_REGISTER` 表を読み取り、登録されたレプリケーション・ソースの中のどれについて変更をキャプチャーする必要があるかを判断し、登録情報をメモリーに保留します。
2. キャプチャー・プログラムは `DB2` ログまたはジャーナルを継続して読み取り、登録されたソース表またはビューの変更レコード (`INSERT`、`UPDATE`、および `DELETE`) を検出します。また、アプライ・プログラムまたはユーザーによって開始されたシグナル・アクションを拾い出すため、シグナル表 (`IBMSNAP_SIGNAL`) への挿入も検出します。アプライ・プログラムが `CAPSTART` シグナルを `IBMSNAP_SIGNAL` 表に挿入し、コミットされたシグナルをキャプチャー・プログラムが検出すると、キャプチャー・プログラムは登録を開始し、関連するソースの変更のキャプチャーを開始します。
3. 登録されたソースの変更のキャプチャーをキャプチャー・プログラムが開始すると、キャプチャー・プログラムは、`DB2` ログまたはジャーナルで検出したコミット済みの変更ごとに、`CD` 表に 1 行を書き込みます (更新 `ID` を `DELETE` または `INSERT` ステートメントとして保管するように指定した場合は 2 行)。キャプチャー・プログラムは、コミットされていない変更については、変更がコミットされるかまたは打ち切られるまで、メモリーに保持します。外部 `CCD` 表でない、登録済みのレプリケーション・ソースは、それぞれ 1 つの関連する `CD` 表を持ちます。
4. コミット・インターバルごとに、キャプチャー・プログラムは `CD` および `UOW` 表に書き込んだデータをコミットし、さらに `IBMSNAP_REGISTER` 表を更新して、どの `CD` 表に新しくコミットされた変更があるかを示すフラグを付けます。

データをターゲット・データベースに適用する

5. 新しく定義されたサブスクリプション・セットについてはすべて、アプライ・プログラムは最初に、キャプチャー・プログラムに変更をキャプチャーするようにシグナルを出します。その後、セットのそれぞれのメンバーについてフル・リフレッシュを実行します (完全なターゲット表は除く)。

- レプリケーションの対象として適切なサブスクリプション・セットがある場合、アプライ・プログラムは `IBMSNAP_REGISTER` 表を調べて、レプリケーションしなければならない変更があるかどうかを判別します。
- アプライ・プログラムは `CD` 表からターゲット表に変更をコピーします。
- アプライ・プログラムは `IBMSNAP_SUBS_SET` 表を更新し、アプライ・プログラムがそれぞれのサブスクリプション・セット用にコピーしたデータの量を記録します。
- アプライ・プログラムは、`CD` 表から変更を読み取った時点を示す値で、プルーン・セット表 (`IBMSNAP_PRUNE_SET`) を更新します。

CD 表のプルーン

- キャプチャー・プログラムが `CD` 表の整理を行う場合、`IBMSNAP_PRUNE_SET` 表にある情報を使用して、どの変更が適用されたかを判別し、これらの既にレプリケーション済みの変更を `CD` 表から削除します。

キャプチャー・トリガーおよびアプライ・プログラム

キャプチャー・トリガーは、レプリケーション・コントロール表のいくつかを使用して、ソース・データベースにどのような変更がなされたかを示し、アプライ・プログラムはそれらのコントロール表の値を使用して、ターゲット・データベースに何をコピーする必要があるかを検出します。

キャプチャー・トリガーは情報のキャプチャーを即時に開始します。キャプチャー・プログラムと異なり、アプライ・プログラムからのシグナルを待つことはしません。

以下に示す処理は、データ保全性を保つために、通常のレプリケーション・シナリオでは、キャプチャー・トリガーとアプライ・プログラムがどのように通信するかを説明しています。

ソースからデータを取り込む

- 登録されたレプリケーション・ソース表で `DELETE`、`UPDATE`、または `INSERT` 操作が行われた場合はいつでも、キャプチャー・トリガーはその変更を、そのソース表用の `CCD` 表に記録します。

データをターゲットに適用する

- 新しく定義されたサブスクリプション・セットについてはすべて、アプライ・プログラムは最初に、キャプチャー・トリガーにシグナルを出し、どの時点から変更データのフェッチを開始するかを示す、有効な開始点を `CCD` 表に記録するように指示します。その後、セットのそれぞれのメンバーについてフル・リフレッシュを実行します (完全なターゲット表は除く)。
- アプライ・プログラムは、`DB2` 以外のリレーショナル・ソース用のサブスクリプション・セットを処理する時には、登録同期化表 (`IBMSNAP_REG_SYNC`) を更新します。この更新により、この表の `UPDATE` トリガーが着火されます。このトリガーは、`IBMSNAP_REGISTER` 表の `SYNCHPOINT` 値を更新し、ターゲットにコピーし

た CCD 表内の最高の SYNCHPOINT 値を記録します。次のサイクルで、アプライ・プログラムは、この SYNCHPOINT より小さいか等しい SYNCHPOINT 値を持つ、CCD 表内の新しいデータを処理します。IBMSNAP_REG_SYNCH 表は DB2 データベースにはないので、アプライ・プログラムはレプリケーション・センターが作成したニックネームを使用して表に書き込みます。

4. アプライ・プログラムは IBMSNAP_REGISTER 表を調べて、複製しなければならない変更があるかどうかを判別します。
5. アプライ・プログラムは CCD 表から得られる変更をターゲット表にコピーします。
6. アプライ・プログラムはサブスクリプション・セット表 (IBMSNAP_SUBS_SET) を更新し、アプライ・プログラムがそれぞれのサブスクリプション・セット用にコピーしたデータの量を記録します。
7. アプライ・プログラムは、CCD 表から変更を読み取った時点を示す値で、登録されたそれぞれのソースのプルーニング・コントロール表 (IBMSNAP_PRUNCNTL) を更新します。

CCD 表のプルーニング

8. IBMSNAP_PRUNCNTL 表の UPDATE トリガーは、ソース・データベースにあるすべての CCD 表を調べて、すでに複製された変更を CCD 表から削除します。

レプリケーション・センターおよびレプリケーション・アラート・モニター

アラート条件を定義する際に、この条件が起こった時に誰に通知するかを定義します。レプリケーション・センターは、このアラート条件とその連絡先情報をレプリケーション・コントロール表に保管する SQL スクリプトを作成します。これらのコントロール表には、すべてのアラート条件および通知情報が含まれ、その内訳は、モニター条件表 (IBMSNAP_CONDITIONS)、モニター連絡先表 (IBMSNAP_CONTACTS)、モニター・グループ表 (IBMSNAP_GROUPS)、およびモニター・グループ連絡先表 (IBMSNAP_CONTACTGRP) です。

メインのモニター・アラート表である、モニター条件表には、モニターの対象となるそれぞれの条件ごとに 1 行登録されています。この表は、それぞれのアラート条件について、次の種類の情報を保持します。

- モニター修飾子
- モニターしたいキャプチャー・サーバーまたはアプライ・サーバーの名前および別名
- モニターしたいコンポーネント (キャプチャー・プログラムまたはアプライ・プログラム)
- キャプチャー・スキーマまたはアプライ修飾子
- サブスクリプション・セットの名前 (セットをモニターする場合)
- モニターしたいアラート条件
- 条件が起こった時に知らせる連絡先

この表は関連する情報用にいくつかの列を持っています。この表についての詳細は、535 ページの『ASN.IBMSNAP_CONDITIONS』を参照してください。

レプリケーション・アラート・モニターのその他の表には、アラート条件が起こった場合に誰に知らせるか (個別の連絡先または連絡先のグループ)、その連絡先にどのような方法で知らせるか (E メールやポケットベル)、およびその条件が起こり続ける場合、どの程度の頻度で連絡先に知らせるか、といった情報が含まれます。

レプリケーション・アラート・モニター、キャプチャー・プログラム、およびアプライ・プログラム

レプリケーション・アラート・モニターは、キャプチャー・コントロール表のいくつかを使用してキャプチャー・プログラムをモニターし、アプライ・コントロール表のいくつかを使用してアプライ・プログラムをモニターします。モニターは、それぞれのキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーで、何をモニターするかによって、異なるレプリケーション・コントロール表から読み取ります。レプリケーション・アラート・モニターは、キャプチャー・プログラムまたはアプライ・プログラムへの介入や通信は行いません。

次の処理は、レプリケーション・アラート・モニターがキャプチャー・プログラムやアプライ・プログラムの条件をどのようにモニターし、アラート条件が起こった場合にどのように連絡先に知らせるかを説明しています。

1. レプリケーション・アラート・モニターは、モニター条件表 (IBMSNAP_CONDITIONS) 内のそれぞれの条件 (モニター修飾子について) のアラート条件および連絡先を読み取ります。
2. アラート条件が定義されているキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーごとに、レプリケーション・アラート・モニターは次のタスクを行います。
 - a. レプリケーション・アラート・モニターはサーバーに接続し、そのサーバーの各アラート条件に関連付けられたレプリケーション・コントロール表を読み、条件のいずれかを満たすかどうかを調べます。
 - b. いずれかの条件を満たした場合、レプリケーション・アラート・モニターはその条件に関するデータをメモリーに保管し、そのサーバーの残りのアラート条件の処理を続けます。
 - c. そのサーバーのすべてのアラート条件の処理が終了したら、レプリケーション・アラート・モニターはキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーを切断し、アラートをモニター・アラート表 (IBMSNAP_ALERTS) に挿入し、その条件について連絡先に知らせます。

関連概念:

- 259 ページの 『第 14 章 DB2 レプリケーション・センターの使用』

関連資料:

- 489 ページの 『アプライ・コントロール・サーバーで使用される表のリスト』
- 486 ページの 『キャプチャー・コントロール・サーバーで使用される表のリスト』
- 490 ページの 『モニター・コントロール・サーバーで使用される表のリスト』

第 23 章 表構造

この章では、各サーバー (キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、モニター・コントロール・サーバー、およびターゲット・サーバー) のレプリケーションに使用される、リレーショナル・データベース表について説明します。また、以下のような、表を参照する 3 つの方法を紹介します。

- 482 ページの『表の概観』のセクションには早見表があります。これには、キャプチャー・コントロール・サーバー、アプライ・コントロール・サーバー、およびモニター・コントロール・サーバー用の表のリスト、各表の列、そして各表の索引が含まれます。
- 各サーバー上の表の概観については、以下を参照してください。
 - 486 ページの『キャプチャー・コントロール・サーバーで使用される表のリスト』
 - 489 ページの『アプライ・コントロール・サーバーで使用される表のリスト』
 - 490 ページの『モニター・コントロール・サーバーで使用される表のリスト』
 - 492 ページの『ターゲット・サーバーで使用される表のリスト』
- 各サーバー上の表に関する詳細な記述、および各表の列の記述は、以下を参照してください。
 - 493 ページの『キャプチャー・コントロール・サーバー上の表、およびその列に関する記述』
 - 517 ページの『アプライ・コントロール・サーバー上の表、およびその列に関する記述』
 - 534 ページの『モニター・コントロール・サーバー上の表、およびその列に関する記述』
 - 542 ページの『ターゲット・サーバー上の表、およびその列に関する記述』

各セクションでは、コントロール表は実表名 (たとえば、ASN.IBMSNAP_APPLYTRACE) のアルファベット順にリストされ、ターゲット表は、英語の表名 (たとえば、replica table) のアルファベット順にリストされています。各表の列は、表内に現れる順にリストされています。

コントロール表の中には、更新に SQL を使用しないことを必要とするものがあります (詳細については、特定の表の記述を参照のこと)。コントロール表を正しく変更しないと、予期しない結果や、データの損失、レプリケーション・パフォーマンスの低下などの問題を引き起こす可能性があります。

表の概観

図 15 および 483 ページの図 16 は、キャプチャー・コントロール・サーバー上の表、各表の列、そして各表の索引を示しています。485 ページの図 18 および 484 ページの図 17 は、アプライ・コントロール・サーバー上の表、各表の列、そして各表の索引を示しています。486 ページの図 19 は、モニター・コントロール・サーバー上の表、各表の列、そして各表の索引を示しています。

キャプチャー・コントロール・サーバーで使用される表 (1/2)

| | |
|---|---|
| <p><i>OS/400 only</i></p> <p>schema.IBMSNAP_AUHTHKN (no unique index)</p> <p>APPLY_QUAL CHAR(18) NOT NULL IBMSNAP_AUHTHKN CHAR(26) NOT NULL JRN_LIB CHAR(10) NOT NULL JRN_NAME CHAR(10) NOT NULL IBMSNAP_LOGMARKER TIMESTAMP NOT NULL</p> | <p>ASN.IBMSNAP_CAPSCHEMAS (CAP_SCHEMA_NAME)</p> <p>CAP_SCHEMA_NAME VARCHAR(30)</p> <p><i>OS/400</i></p> <p>ASN.IBMSNAP_CAPSCHEMAS (CAP_SCHEMA_NAME)</p> <p>CAP_SCHEMA_NAME VARCHAR(30) STATUS CHAR(1)</p> |
| <p><i>UNIX, Windows, and z/OS only</i></p> <p>schema.IBMSNAP_CAPENQ (no unique index)</p> <p>LOCKNAME CHAR(9)</p> | <p>schema.IBMSNAP_CAPTRACE (TRACE_TIME)</p> <p>OPERATION CHAR(8) NOT NULL TRACE_TIME TIMESTAMP NOT NULL DESCRIPTION VARCHAR(1024) NOT NULL</p> <p><i>OS/400</i></p> <p>schema.IBMSNAP_CAPTRACE (TRACE_TIME)</p> <p>OPERATION CHAR(8) NOT NULL TRACE_TIME TIMESTAMP NOT NULL JOB_NAME CHAR(26) NOT NULL JOB_STR_TIME TIMESTAMP NOT NULL DESCRIPTION VARCHAR(298) NOT NULL</p> |
| <p>schema.IBMSNAP_CAPMON (MONITOR_TIME)</p> <p>MONITOR_TIME TIMESTAMP NOT NULL RESTART_TIME TIMESTAMP NOT NULL CURRENT_MEMORY INT NOT NULL CD_ROWS_INSERTED INT NOT NULL RECAP_ROWS_SKIPPED INT NOT NULL TRIGR_ROWS_SKIPPED INT NOT NULL CHG_ROWS_SKIPPED INT NOT NULL TRANS_PROCESSED INT NOT NULL TRANS_SPILLED INT NOT NULL MAX_TRAN_SIZE INT NOT NULL LOCKING_RETRIES INT NOT NULL JRN_LIB CHAR(10) JRN_NAME CHAR(10) LOGREADLIMIT INT NOT NULL CAPTURE_IDLE INT NOT NULL SYNCHTIME TIMESTAMP NOT NULL</p> | <p>schema.IBMSNAP_PRUNCNTL (SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL, APPLY_QUAL, SET_NAME, TARGET_SERVER, TARGET_TABLE, TARGET_OWNER, MAP_ID)</p> <p>TARGET_SERVER CHAR(18) NOT NULL TARGET_OWNER VARCHAR(30) NOT NULL TARGET_TABLE VARCHAR(128) NOT NULL SYNCHTIME TIMESTAMP SYNCHPOINT CHAR(10) FOR BIT DATA SOURCE_OWNER VARCHAR(30) NOT NULL SOURCE_TABLE VARCHAR(128) NOT NULL SOURCE_VIEW_QUAL SMALLINT NOT NULL APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL CNTL_SERVER CHAR(18) NOT NULL TARGET_STRUCTURE SMALLINT NOT NULL CNTL_ALIAS CHAR(8) PHYS_CHANGE_OWNER VARCHAR(30) PHYS_CHANGE_TABLE VARCHAR(128) MAP_ID VARCHAR(10) NOT NULL</p> |
| <p>schema.IBMSNAP_CAPPARMS (no unique index)</p> <p>RETENTION_LIMIT INT LAG_LIMIT INT COMMIT_INTERVAL INT PRUNE_INTERVAL INT TRACE_LIMIT INT MONITOR_LIMIT INT MONITOR_INTERVAL INT MEMORY_LIMIT SMALLINT REMOTE_SRC_SERVER CHAR(18) AUTOPRUNE CHAR(1) TERM CHAR(1) AUTOSTOP CHAR(1) LOGREUSE CHAR(1) LOGSTDOUT CHAR(1) SLEEPINTERVAL SMALLINT CAPTURE_PATH VARCHAR(1040) STARTMODE VARCHAR(10)</p> | <p>schema.IBMSNAP_PRUNE_LOCK (no unique index)</p> <p>DUMMY CHAR(1)</p> |

図 15. キャプチャー・コントロール・サーバーで使用される表：キャプチャー・コントロール・サーバーで、キャプチャー・プログラム、アプライ・プログラム、およびキャプチャー・トリガーから使用される表。各表の固有索引を構成する列は、表名の下に括弧内に示されています。

キャプチャー・コントロール・サーバーで使用される表 (2/2)

| | |
|---|--|
| <p>schema.IBMSNAP_PRUNE_SET (TARGET_SERVER, APPLY_QUAL, SET_NAME)</p> <p>TARGET_SERVER CHAR(18) NOT NULL APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL SYNCHTIME TIMESTAMP SYNCHPOINT CHAR(10) FOR BIT DATA NOT NULL</p> | <p>schema.IBMSNAP_REG_SYNCH (TRIGGER_ME)</p> <p>TRIGGER_ME CHAR(1) NOT NULL</p> |
| <p>schema.IBMSNAP_REGISTER (SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL)</p> <p>SOURCE_OWNER VARCHAR(30) NOT NULL SOURCE_TABLE VARCHAR(128) NOT NULL SOURCE_VIEW_QUAL SMALLINT NOT NULL GLOBAL_RECORD CHAR(1) NOT NULL SOURCE_STRUCTURE SMALLINT NOT NULL SOURCE_CONDENSED CHAR(1) NOT NULL SOURCE_COMPLETE CHAR(1) NOT NULL CD_OWNER VARCHAR(30) CD_TABLE VARCHAR(128) PHYS_CHANGE_OWNER VARCHAR(30) PHYS_CHANGE_TABLE VARCHAR(128) CD_OLD_SYNCHPOINT CHAR(10) FOR BIT DATA CD_NEW_SYNCHPOINT CHAR(10) FOR BIT DATA DISABLE_REFRESH SMALLINT NOT NULL CCD_OWNER VARCHAR(30) CCD_TABLE VARCHAR(128) CCD_OLD_SYNCHPOINT CHAR(10) FOR BIT DATA SYNCHPOINT CHAR(10) FOR BIT DATA SYNCHTIME TIMESTAMP CCD_CONDENSED CHAR(1) CCD_COMPLETE CHAR(1) ARCH_LEVEL CHAR(4) NOT NULL DESCRIPTION CHAR(254) BEFORE_IMG_PREFIX VARCHAR(4) CONFLICT_LEVEL CHAR(1) CHG_UPD_TO_DEL_INS CHAR(1) CHGONLY CHAR(1) RECAPTURE CHAR(1) OPTION_FLAGS CHAR(4) NOT NULL STOP_ON_ERROR CHAR(1) STATE CHAR(1) STATE_INFO CHAR(8)</p> | <p>schema.IBMSNAP_RESTART (no unique index)</p> <p>MAX_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL MAX_COMMIT_TIME TIMESTAMP NOT NULL MIN_INFLIGHTSEQ CHAR(10) FOR BIT DATA NOT NULL CURR_COMMIT_TIME TIMESTAMP NOT NULL CAPTURE_FIRST_SEQ CHAR(10) FOR BIT DATA NOT NULL</p> <p>OS/400</p> <p>schema.IBMSNAP_RESTART (JRN_LIB, JRN_NAME)</p> <p>MAX_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL MAX_COMMIT_TIME TIMESTAMP NOT NULL MIN_INFLIGHTSEQ CHAR(10) FOR BIT DATA NOT NULL CURR_COMMIT_TIME TIMESTAMP NOT NULL CAPTURE_FIRST_SEQ CHAR(10) FOR BIT DATA NOT NULL</p> <p>UID INTEGER NOT NULL SEQNBR BIGINT NOT NULL JRN_LIB CHAR(10) NOT NULL JRN_NAME CHAR(10) NOT NULL STATUS CHAR(1)</p> |
| <p>OS/400 only</p> <p>schema.IBMSNAP_REG_EXT (VERSION, SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL)</p> <p>VERSION INT NOT NULL SOURCE_OWNER VARCHAR(30) NOT NULL SOURCE_TABLE VARCHAR(128) NOT NULL SOURCE_NAME CHAR(10) SOURCE_MBR CHAR(10) SOURCE_TABLE_RDB CHAR(18) JRN_LIB CHAR(10) JRN_NAME CHAR(10) FR_START_TIME TIMESTAMP SOURCE_VIEW_QUAL SMALLINT NOT NULL CMT_BEHAVIOR_CASE SMALLINT NOT NULL WITH DEFAULT MAX_ROWS_BTWN_CMTS SMALLINT NOT NULL WITH DEFAULT</p> | <p>schema.IBMSNAP_SEQTABLE (SEQ)</p> <p>SEQ INTEGER NOT NULL</p> |
| | <p>schema.IBMSNAP_SIGNAL (SIGNAL_TIME)</p> <p>SIGNAL_TIME TIMESTAMP NOT NULL WITH DEFAULT SIGNAL_TYPE VARCHAR(30) NOT NULL SIGNAL_SUBTYPE VARCHAR(30) SIGNAL_INPUT_IN VARCHAR(500) SIGNAL_STATE CHAR(1) NOT NULL SIGNAL_LSN CHAR(10) FOR BIT DATA</p> |
| | <p>schema.IBMSNAP_UOW (IBMSNAP_COMMITSEQ, IBMSNAP_LOGMARKER)</p> <p>IBMSNAP_UOWID CHAR(10) FOR BIT DATA NOT NULL IBMSNAP_COMMITSEQ CHAR(10) FOR BIT DATA NOT NULL IBMSNAP_LOGMARKER TIMESTAMP NOT NULL IBMSNAP_AUTHTKN VARCHAR(30) NOT NULL IBMSNAP_AUTHID VARCHAR(30) NOT NULL IBMSNAP_REJ_CODE CHAR(1) NOT NULL WITH DEFAULT IBMSNAP_APPLY_QUAL CHAR(18) NOT NULL WITH DEFAULT</p> |

図 16. キャプチャー・コントロール・サーバーで使用される表 (続き): キャプチャー・コントロール・サーバーで、キャプチャー・プログラム、アプライ・プログラム、およびキャプチャー・トリガーから使用される表。各表の固有索引を構成する列は、表名の下の括弧内に示されています。

アプライ・コントロール・サーバーで使用される表 (1/2)

| ASNJBMSNAP_APPLYTRAIL (no unique index) | |
|---|------------------------------------|
| APPLY_QUAL | CHAR(18) NOT NULL |
| SET_NAME | CHAR(18) NOT NULL |
| SET_TYPE | CHAR(1) NOT NULL |
| WHOS_ON_FIRST | CHAR(1) NOT NULL |
| ASNLOAD | CHAR(1) |
| FULL_REFRESH | CHAR(1) |
| EFFECTIVE_MEMBERS | INT |
| SET_INSERTED | INT NOT NULL |
| SET_DELETED | INT NOT NULL |
| SET_UPDATED | INT NOT NULL |
| SET_REWORKED | INT NOT NULL |
| SET_REJECTED_TRXS | INT NOT NULL |
| STATUS | SMALLINT NOT NULL |
| LASTRUN | TIMESTAMP NOT NULL |
| LASTSUCCESS | TIMESTAMP |
| SYNCHPOINT | CHAR(10) FOR BIT DATA |
| SYNCHTIME | TIMESTAMP |
| SOURCE_SERVER | CHAR(18) NOT NULL |
| SOURCE_ALIAS | CHAR(8) |
| SOURCE_OWNER | VARCHAR(30) |
| SOURCE_TABLE | VARCHAR(128) |
| SOURCE_VIEW_QUAL | SMALLINT |
| TARGET_SERVER | CHAR(18) NOT NULL |
| TARGET_ALIAS | CHAR(8) |
| TARGET_OWNER | VARCHAR(30) NOT NULL |
| TARGET_TABLE | VARCHAR(128) NOT NULL |
| CAPTURE_SCHEMA | VARCHAR(30) NOT NULL |
| TGT_CAPTURE_SCHEMA | VARCHAR(30) |
| FEDERATED_SRC_SRVR | VARCHAR(18) |
| FEDERATED_TGT_SRVR | VARCHAR(18) |
| JRN_LIB | CHAR(10) |
| JRN_NAME | CHAR(10) |
| COMMIT_COUNT | SMALLINT |
| OPTION_FLAGS | CHAR(4) NOT NULL |
| EVENT_NAME | CHAR(18) |
| ENDTIME | TIMESTAMP NOT NULL WITH DEFAULT |
| SOURCE_CONN_TIME | TIMESTAMP |
| SQLSTATE | CHAR(5) |
| SQLCODE | INT |
| SQLERRP | CHAR(8) |
| SQLERRM | VARCHAR(70) |
| APPERRM | VARCHAR(760) |

| ASNJBMSNAP_APPENQ (APPLY_QUAL) | |
|--|----------|
| APPLY_QUAL | CHAR(18) |

| <i>OS/400 only</i> ASNJBMSNAP_APPLY_JOB | |
|---|-------------------|
| APPLY_QUAL | CHAR(18) NOT NULL |
| CONTROL_SERVER | CHAR(18) NOT NULL |
| JOB_NAME | CHAR(10) NOT NULL |
| USER_NAME | CHAR(10) NOT NULL |
| JOB_NUMBER | CHAR(6) NOT NULL |

| ASNJBMSNAP_APPLYTRACE (APPLY_QUAL, TRACE_TIME) | |
|--|------------------------|
| APPLY_QUAL | CHAR(18) NOT NULL |
| TRACE_TIME | TIMESTAMP NOT NULL |
| OPERATION | CHAR(8) NOT NULL |
| DESCRIPTION | VARCHAR(1024) NOT NULL |

図 17. アプライ・コントロール・サーバーで使用される表：アプライ・コントロール・サーバーでアプライ・プログラムから使用される表。各表の固有索引を構成する列は、表名の下に括弧内に示されています。

アプライ・コントロール・サーバーで使用される表 (2/2)

| | |
|--|--|
| <p>ASN.IBMSNAP_SUBS_COLS (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST, TARGET_OWNER, TARGET_TABLE, TARGET_NAME)</p> <p>APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL WHOS_ON_FIRST CHAR(1) NOT NULL TARGET_OWNER VARCHAR(30) NOT NULL TARGET_TABLE VARCHAR(128) NOT NULL COL_TYPE CHAR(1) NOT NULL TARGET_NAME VARCHAR(30) NOT NULL IS_KEY CHAR(1) NOT NULL COLNO SMALLINT NOT NULL EXPRESSION VARCHAR(254) NOT NULL</p> | <p>ASN.IBMSNAP_SUBS_STMTS (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST, BEFORE_OR_AFTER, STMT_NUMBER)</p> <p>APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL WHOS_ON_FIRST CHAR(1) NOT NULL BEFORE_OR_AFTER CHAR(1) NOT NULL STMT_NUMBER SMALLINT NOT NULL EI_OR_CALL CHAR(1) NOT NULL SQL_STMT VARCHAR(1024) ACCEPT_SQLSTATES VARCHAR(50)</p> |
| <p>ASN.IBMSNAP_SUBS_EVENT (EVENT_NAME, EVENT_TIME)</p> <p>EVENT_NAME CHAR(18) NOT NULL EVENT_TIME TIMESTAMP NOT NULL END_SYNCHPOINT CHAR(10) FOR BIT DATA END_OF_PERIOD TIMESTAMP</p> | <p>ASN.IBMSNAP_SUBS_MEMBR (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST, SOURCE_OWNER, SOURCE_TABLE, SOURCE_VIEW_QUAL, TARGET_OWNER, TARGET_TABLE)</p> <p>APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL WHOS_ON_FIRST CHAR(1) NOT NULL SOURCE_OWNER VARCHAR(30) NOT NULL SOURCE_TABLE VARCHAR(128) NOT NULL SOURCE_VIEW_QUAL SMALLINT NOT NULL TARGET_OWNER VARCHAR(30) NOT NULL TARGET_TABLE VARCHAR(128) NOT NULL TARGET_CONDENSED CHAR(1) NOT NULL TARGET_COMPLETE CHAR(1) NOT NULL TARGET_STRUCTURE SMALLINT NOT NULL PREDICATES VARCHAR(1024) MEMBER_STATE CHAR(1) TARGET_KEY_CHG CHAR(1) NOT NULL UOW_CD_PREDICATES VARCHAR(1024) JOIN_UOW_CD CHAR(1) LOADX_TYPE SMALLINT LOADX_SRC_N_OWNER VARCHAR(30) LOADX_SRC_N_TABLE VARCHAR(128)</p> |
| <p>ASN.IBMSNAP_SUBS_SET (APPLY_QUAL, SET_NAME, WHOS_ON_FIRST)</p> <p>APPLY_QUAL CHAR(18) NOT NULL SET_NAME CHAR(18) NOT NULL SET_TYPE CHAR(1) NOT NULL WHOS_ON_FIRST CHAR(1) NOT NULL ACTIVATE SMALLINT NOT NULL SOURCE_SERVER CHAR(18) NOT NULL SOURCE_ALIAS CHAR(8) TARGET_SERVER CHAR(18) NOT NULL TARGET_ALIAS CHAR(8) STATUS SMALLINT NOT NULL LASTRUN TIMESTAMP NOT NULL REFRESH_TYPE CHAR(1) NOT NULL SLEEP_MINUTES INT EVENT_NAME CHAR(18) LASTSUCCESS TIMESTAMP SYNCHPOINT CHAR(10) FOR BIT DATA SYNCHTIME TIMESTAMP CAPTURE_SCHEMA VARCHAR(30) NOT NULL TGT_CAPTURE_SCHEMA VARCHAR(30) FEDERATED_SRC_SRVR VARCHAR(18) FEDERATED_TGT_SRVR VARCHAR(18) JRN_LIB CHAR(10) JRN_NAME CHAR(10) OPTION_FLAGS CHAR(4) NOT NULL COMMIT_COUNT SMALLINT MAX_SYNCH_MINUTES SMALLINT AUX_STMTS SMALLINT NOT NULL ARCH_LEVEL CHAR(4) NOT NULL</p> | |

図 18. アプライ・コントロール・サーバーで使用される表 (続き): アプライ・コントロール・サーバーでアプライ・プログラムから使用される表。各表の固有索引を構成する列は、表名の下の括弧内に示されています。

モニター・コントロール・サーバーで使用される表

| | |
|---|---|
| <p>ASN.IBMSNAP_ALERTS (MONITOR_QUAL, COMPONENT, SERVER_NAME, SCHEMA_OR_QUAL, CONDITION_NAME, ALERT_CODE)</p> <p>MONITOR_QUAL CHAR(18) NOT NULL ALERT_TIME TIMESTAMP NOT NULL COMPONENT CHAR(1) NOT NULL SERVER_NAME CHAR(18) NOT NULL SERVER_ALIAS CHAR(8) SCHEMA_OR_QUAL VARCHAR(30) NOT NULL SET_NAME CHAR(18) NOT NULL WITH DEFAULT CONDITION_NAME CHAR(18) NOT NULL OCCURRED_TIME TIMESTAMP NOT NULL ALERT_COUNTER SMALLINT NOT NULL ALERT_CODE CHAR(10) NOT NULL RETURN_CODE INT NOT NULL NOTIFICATION_SENT CHAR(1) NOT NULL ALERT_MESSAGE VARCHAR(1024) NOT NULL</p> | <p>ASN.IBMSNAP_GROUPS (GROUP_NAME)</p> <p>GROUP_NAME VARCHAR(127) NOT NULL DESCRIPTION VARCHAR(1024)</p> |
| <p>ASN.IBMSNAP_CONDITIONS (MONITOR_QUAL, SERVER_NAME, COMPONENT, SCHEMA_OR_QUAL, SET_NAME, CONDITION_NAME)</p> <p>SERVER_NAME CHAR(18) NOT NULL COMPONENT CHAR(1) NOT NULL SCHEMA_OR_QUAL VARCHAR(30) NOT NULL SET_NAME CHAR(18) NOT NULL WITH DEFAULT MONITOR_QUAL CHAR(18) NOT NULL SERVER_ALIAS CHAR(8) ENABLED CHAR(1) NOT NULL CONDITION_NAME CHAR(18) NOT NULL PARM_INT INT PARM_CHAR VARCHAR(128) CONTACT_TYPE CHAR(1) NOT NULL CONTACT VARCHAR(127) NOT NULL</p> | <p>ASN.IBMSNAP_MONENQ (MONITOR_QUAL)</p> <p>MONITOR_QUAL CHAR(18) NOT NULL</p> |
| <p>ASN.IBMSNAP_CONTACTGRP (GROUP_NAME, CONTACT_NAME)</p> <p>GROUP_NAME VARCHAR(127) NOT NULL CONTACT_NAME VARCHAR(127) NOT NULL</p> | <p>ASN.IBMSNAP_MONSERVERS (MONITOR_QUAL, SERVER_NAME)</p> <p>MONITOR_QUAL CHAR(18) NOT NULL SERVER_NAME CHAR(18) NOT NULL SERVER_ALIAS CHAR(8) LAST_MONITOR_TIME TIMESTAMP NOT NULL START_MONITOR_TIME TIMESTAMP END_MONITOR_TIME TIMESTAMP LASTRUN TIMESTAMP NOT NULL LASTSUCCESS TIMESTAMP STATUS SMALLINT NOT NULL</p> |
| <p>ASN.IBMSNAP_CONTACTS (CONTACT_NAME)</p> <p>CONTACT_NAME VARCHAR(127) NOT NULL EMAIL_ADDRESS VARCHAR(128) NOT NULL ADDRESS_TYPE CHAR(1) NOT NULL DELEGATE VARCHAR(127) DELEGATE_START DATE DELEGATE_END DATE DESCRIPTION VARCHAR(1024)</p> | <p>ASN.IBMSNAP_MONTRACE (MONITOR_QUAL, TRACE_TIME)</p> <p>MONITOR_QUAL CHAR(18) NOT NULL TRACE_TIME TIMESTAMP NOT NULL OPERATION CHAR(8) NOT NULL DESCRIPTION VARCHAR(1024) NOT NULL</p> |
| | <p>ASN.IBMSNAP_MONTRAIL (no unique index)</p> <p>MONITOR_QUAL CHAR(18) NOT NULL SERVER_NAME CHAR(18) NOT NULL SEVER_ALIAS CHAR(8) STATUS SMALLINT NOT NULL LASTRUN TIMESTAMP NOT NULL LASTSUCCESS TIMESTAMP ENDTIME TIMESTAMP NOT NULL WITH DEFAULT LAST_MONITOR_TIME TIMESTAMP NOT NULL START_MONITOR_TIME TIMESTAMP END_MONITOR_TIME TIMESTAMP SQLCODE INT SQLSTATE CHAR(5) NUM_ALERTS INT NOT NULL NUM_NOTIFICATIONS INT NOT NULL</p> |

図 19. モニター・コントロール・サーバーで使用される表： モニター・コントロール・サーバーで、レプリケーション・アラート・モニター・プログラムから使用される表。各表の固有索引を構成する列は、表名の下に括弧内に示されています。

キャプチャー・コントロール・サーバーで使用される表のリスト

キャプチャー・コントロール・サーバーに保管される表には、ユーザーの登録済みソースに関する情報と、キャプチャー・プログラムまたはトリガーがソースを処理する方法に関する情報が入っています。UNIX、Windows、および z/OS の場合は、レプリケーション・センターを使用して、ユーザーの指定に合わせてこれらのコントロール表を作成します。OS/400 の場合は、DataPropagator for iSeries のインストール時に、ASN ライ

ブラリの中これらのコントロール表が自動的に作成されます。OS/400 上でレプリケーションのシステム・コマンドを使用して、代替のキャプチャー・スキーマ内にキャプチャー・コントロール表を作成できます。

表 52. キャプチャー・コントロール・サーバーで使用される表のクイック・リファレンス

| 表名 | 記述 | 参照トピック |
|---|--|--------|
| ASN.IBMSNAP_CAPSCHEMAS | キャプチャー・スキーマ表 | 493 |
| | すべてのキャプチャー・スキーマの名前を保持します。 | |
| schema.IBMSNAP_AUTHTKN (OS/400) | アプライ修飾子相互参照表 | 493 |
| | update-anywhere をサポートするための情報が入っています。 | |
| schema.IBMSNAP_CAPENQ (UNIX、Windows、z/OS) | キャプチャー・エンキュー表 | 494 |
| | この表は、各キャプチャー・スキーマについて、次のことを保証するために使用されます。 <ul style="list-style-type: none"> DB2 for UNIX および DB2 for Windows の場合は、1 つのデータベースに対して 1 つのキャプチャー・プログラムのみが実行されていること。 データを共有しない DB2 for z/OS の場合は、1 つのサブシステムに対して 1 つのキャプチャー・プログラムのみが実行されていること。 データを共有する DB2 for z/OS の場合は、1 つのデータ共有グループに対して 1 つのキャプチャー・プログラムのみが実行されていること。 | |
| schema.CD_table | 変更データ (CD) 表 | 500 |
| | ソースに発生する変更に関する情報を保持します。この表は、レプリケーション・ソースが登録されるまでは作成されません。 | |
| schema.CCD_table | 整合変更データ (CCD) 表 | 499 |
| | ソースに発生する変更に関する情報と、これらの変更の順序を識別するための追加の列を保持します。 | |
| schema.IBMSNAP_CAPMON | キャプチャー・モニター表 | 495 |
| | キャプチャー・プログラムの進行状況をモニターする助けとなる操作統計を保持します。 | |
| schema.IBMSNAP_CAPPARMS | キャプチャー・パラメーター表 | 496 |
| | キャプチャー・プログラムの操作をコントロールするためにユーザーが指定できるパラメーターを保持します。 | |
| schema.IBMSNAP_CAPTRACE | キャプチャー・トレース表 | 498 |
| | キャプチャー・プログラムからの重要なメッセージを保持します。 | |
| schema.IBMSNAP_PRUNE_LOCK | ブルーニング・ロック表 | 503 |
| | コールド・スタート、または保持制限ブルーニング時に、キャプチャー・プログラムの CD 表へのアクセスを直列化するために使用されます。 | |

表 52. キャプチャー・コントロール・サーバーで使用される表のクイック・リファレンス (続き)

| 表名 | 記述 | 参照 トピック |
|---|---|------------|
| <i>schema</i> .IBMSNAP_PRUNE_SET | ブルーニング・セット表 CD 表のブルーニングを調整します。 | 503 |
| <i>schema</i> .IBMSNAP_PRUNCNTL | ブルーニング・コントロール表 キャプチャー・プログラムとアプライ・プログラムの間で同期点更新を調整します。 | 501 |
| <i>schema</i> .IBMSNAP_REG_EXT (OS/400) | 登録拡張表 登録表を拡張したものです。ジャーナル名やリモート・ソース表のデータベース・エントリー名など、レプリケーション・ソースに関する追加情報が含まれます。 | 504 |
| <i>schema</i> .IBMSNAP_REGISTER | 登録表 レプリケーション・ソース表の名前、その属性、および対応する CD 表および CCD 表の名前など、レプリケーション・ソースに関する情報が入ります。 | 505 |
| <i>schema</i> .IBMSNAP_REG_SYNCH (DB2 以外のリレーショナル) | 登録同期表 DB2 以外のリレーショナル・データ・ソースから複製するときに使用されます。この表の更新トリガーは、アプライ・プログラムが登録表から情報を読み込む前に、登録表のすべての行で SYNCHPOINT 値の更新を始めることにより、キャプチャー・プログラムをシミュレートします。 | 510 |
| <i>schema</i> .IBMSNAP_RESTART | 再始動表 キャプチャー・プログラムがログまたはジャーナル内の正しい時点からキャプチャーを再開できるようにするための情報が入っています。OS/400 環境では、この表は RCVJRNE (ジャーナル・エントリーの受信) コマンドの開始時刻を判別するためにも使用されます。 | 511 |
| <i>schema</i> .IBMSNAP_SEQTABLE (DB2 以外のリレーショナル) | 順序付け表 Informix 表のログ・シーケンス番号と同等のものとして DB2 レプリケーションから使用される一連の固有番号を保持します。 | 512 |
| <i>schema</i> .IBMSNAP_SIGNAL | シグナル表 キャプチャー・プログラムにプロンプトを出すために使用されるすべてのシグナルを保持します。これらのシグナルは手動で、またはアプライ・プログラムから送信できます。 | 513 |
| <i>schema</i> .IBMSNAP_UOW | 作業単位 (UOW) 表 ソース表にコミットされたトランザクションに関する追加情報を提供します。 | 515 |

関連資料:

- 493 ページの 『ASN.IBMSNAP_CAPSCHEMAS』
- 493 ページの 『*schema*.IBMSNAP_AUTHTKN (OS/400)』
- 494 ページの 『*schema*.IBMSNAP_CAPENQ (UNIX、Windows、z/OS)』

- 495 ページの 『*schema.IBMSNAP_CAPMON*』
- 496 ページの 『*schema.IBMSNAP_CAPPARMS*』
- 498 ページの 『*schema.IBMSNAP_CAPTRACE* (DB2 のみ)』
- 499 ページの 『*schema.CCD_table* (DB2 以外)』
- 500 ページの 『*schema.CD_table*』
- 501 ページの 『*schema.IBMSNAP_PRUNCNTL*』
- 503 ページの 『*schema.IBMSNAP_PRUNE_LOCK*』
- 503 ページの 『*schema.IBMSNAP_PRUNE_SET*』
- 504 ページの 『*schema.IBMSNAP_REG_EXT* (OS/400)』
- 505 ページの 『*schema.IBMSNAP_REGISTER*』
- 510 ページの 『*schema.IBMSNAP_REG_SYNCH* (DB2 以外のリレーショナル)』
- 511 ページの 『*schema.IBMSNAP_RESTART*』
- 512 ページの 『*schema.IBMSNAP_SEQTABLE* (Informix)』
- 513 ページの 『*schema.IBMSNAP_SIGNAL*』
- 515 ページの 『*schema.IBMSNAP_UOW*』

アプライ・コントロール・サーバーで使用される表のリスト

アプライ・コントロール・サーバーに保管される表には、ユーザーのサブスクリプション定義に関する情報が入っています。UNIX、Windows、および z/OS の場合は、レプリケーション・センターを使用して、ユーザーの指定に合わせてこれらのコントロール表を作成します。OS/400 の場合は、DataPropagator for iSeries のインストール時に、これらのコントロール表が自動的に作成されます。

表 53. アプライ・コントロール・サーバーで使用される表のクイック・リファレンス

| 表名 | 記述 | 参照 トピック |
|--------------------------------|--|------------|
| ASN.IBMSNAP_APPENQ | アプライ・エンキュー表 | 517 |
| | 1 つのアプライ修飾子に対して 1 つのアプライ・プログラムだけが確実に実行されるようにするために使用されます。 | |
| ASN.IBMSNAP_APPLY_JOB (OS/400) | アプライ・ジョブ表 | 518 |
| | アプライ・コントロール・サーバーで実行中のアプライ・プログラムのインスタンスごとに固有のアプライ修飾子が確実に存在するようにするために使用されます。 | |
| ASN.IBMSNAP_APPLYTRACE | アプライ・トレース表 | 518 |
| | アプライ・プログラムからの重要なメッセージを保持します。 | |
| ASN.IBMSNAP_APPLYTRAIL | アプライ・トレール表 | 519 |
| | アプライ・プログラムに関する監査証跡情報を保持します。 | |

表 53. アプライ・コントロール・サーバーで使用される表のクイック・リファレンス (続き)

| 表名 | 記述 | 参照 トピック |
|------------------------|--|------------|
| ASN.IBMSNAP_SUBS_COLS | サブスクリプション表 ターゲット表またはビュー内の列を、ソース表またはビュー内の対応する列にマップします。 | 523 |
| ASN.IBMSNAP_SUBS_EVENT | サブスクリプション・イベント表 アプライ・プログラムによるサブスクリプション・セットの処理をコントロールするためにユーザーが定義するイベントを保持します。 | 524 |
| ASN.IBMSNAP_SUBS_MEMBR | サブスクリプション・メンバー表 ソースとターゲット表の対を識別し、その対の処理情報を指定します。 | 525 |
| ASN.IBMSNAP_SUBS_SET | サブスクリプション・セット表 アプライ・プログラムによりグループとして処理されるサブスクリプション・セット・メンバーの各セットの処理情報を保持します。 | 528 |
| ASN.IBMSNAP_SUBS_STMTS | サブスクリプション・ステートメント表 ユーザーがサブスクリプション・セットに対して定義する、SQL ステートメント、またはストアード・プロシージャ呼び出しを保持します。これらは、アプライ・プログラムによるセットの処理前、または処理後に呼び出されます。 | 532 |

関連資料:

- 517 ページの 『ASN.IBMSNAP_APPENQ』
- 518 ページの 『ASN.IBMSNAP_APPLY_JOB (OS/400)』
- 518 ページの 『ASN.IBMSNAP_APPLYTRACE』
- 519 ページの 『ASN.IBMSNAP_APPLYTRAIL』
- 523 ページの 『ASN.IBMSNAP_SUBS_COLS』
- 524 ページの 『ASN.IBMSNAP_SUBS_EVENT』
- 525 ページの 『ASN.IBMSNAP_SUBS_MEMBR』
- 528 ページの 『ASN.IBMSNAP_SUBS_SET』
- 532 ページの 『ASN.IBMSNAP_SUBS_STMTS』

モニター・コントロール・サーバーで使用される表のリスト

モニター・コントロール・サーバー上の表には、アラート条件が発生したときに、レプリケーション・アラート・モニターから、いつ、どのように、そしてだれに連絡するかに関する情報が含まれます。UNIX、Windows、および z/OS の場合は、レプリケーション・センターを使用して、ユーザーの指定に合わせてこれらのコントロール表を作成します。DataPropagator for iSeries にはモニター・コントロール表はありません。

表 54. モニター・コントロール・サーバーで使用される表のクイック・リファレンス

| 表名 | 記述 | 参照 トピック |
|------------------------|---|------------|
| ASN.IBMSNAP_ALERTS | <p>モニター・アラート表</p> <p>レプリケーション・アラート・モニターから発行されるすべてのアラートのレコードを保持します。</p> | 534 |
| ASN.IBMSNAP_CONDITIONS | <p>モニター条件表</p> <p>レプリケーション・アラート・モニターから担当者への連絡が必要なアラート条件と、特定の条件が発生したときの連絡先のグループまたは個人の名前が入っています。</p> | 535 |
| ASN.IBMSNAP_CONTACTGRP | <p>モニター・グループ連絡先表</p> <p>連絡先グループを構成する個人の連絡先が入っています。</p> | 538 |
| ASN.IBMSNAP_CONTACTS | <p>モニター連絡先表</p> <p>連絡先名に関連付けられたアラート条件が発生したときに、レプリケーション・アラート・モニターから各個人またはグループに通知する方法に関する情報が入っています。</p> | 538 |
| ASN.IBMSNAP_GROUPS | <p>モニター・グループ表</p> <p>各連絡先グループの名前と記述が入っています。</p> | 539 |
| ASN.IBMSNAP_MONENQ | <p>モニター・エンキュー表</p> <p>1 つのモニター修飾子に対して 1 つのレプリケーション・アラート・モニター・プログラムだけが確実に実行されるようにするために使用されます。</p> | 539 |
| ASN.IBMSNAP_MONSERVERS | <p>モニター・サーバー表</p> <p>サーバーがレプリケーション・アラート・モニター・プログラム (モニター修飾子によって識別される) からモニターされた最後の時刻を保持します。</p> | 539 |
| ASN.IBMSNAP_MONTRACE | <p>モニター・トレース表</p> <p>モニター・プログラムからの重要なメッセージを保持します。</p> | 540 |
| ASN.IBMSNAP_MONTRAIL | <p>モニター・トレール表</p> <p>各モニター・サイクルに関する重要な情報を保持します。</p> | 541 |

関連資料:

- 534 ページの 『ASN.IBMSNAP_ALERTS』
- 535 ページの 『ASN.IBMSNAP_CONDITIONS』
- 538 ページの 『ASN.IBMSNAP_CONTACTGRP』
- 541 ページの 『ASN.IBMSNAP_MONTRAIL』
- 538 ページの 『ASN.IBMSNAP_CONTACTS』
- 539 ページの 『ASN.IBMSNAP_GROUPS』
- 539 ページの 『ASN.IBMSNAP_MONENQ』
- 539 ページの 『ASN.IBMSNAP_MONSERVERS』

ターゲット・サーバーで使用される表のリスト

ターゲット・サーバーには、さまざまなタイプのターゲット表が保管されます。ユーザーがターゲット表として既存の表を使用しない場合、レプリケーション・センターは、ユーザーがサブスクリプション・セット・メンバーをどのように定義しているかに従って、ユーザーの指定に合わせてターゲット表を構築します。

表 55. ターゲット表のクイック・リファレンス

| 表名 | 記述 | 参照トピック |
|--------------------------------|--|--------|
| <i>schema.base_aggregate</i> | 基礎集約表 ソース表から集約されたデータを保持します。 | 542 |
| <i>schema.change_aggregate</i> | 変更集約表 CD 表から集約されたデータを保持します。 | 542 |
| <i>schema.CCD</i> | 整合変更データ (CCD) 表 ソースに発生する変更に関する情報と、これらの変更の順序を識別するための追加の列を保持します。 | 543 |
| <i>schema.point_in_time</i> | ポイント・イン・タイム表 ソース・ログの中でデータがコミットされた特定の時刻を記録する追加の列を持つ、ソース・データのコピー。 | 545 |
| <i>schema.replica</i> | レプリカ表 update-anywhere レプリケーションで使用されるターゲット表のタイプ。 | 546 |
| <i>schema.user_copy</i> | ユーザー・コピー表 ソース表のコピー。 | 547 |

関連資料:

- 542 ページの 『基礎集約表』
- 542 ページの 『変更集約表』
- 543 ページの 『整合変更データ (CCD) 表』
- 545 ページの 『ポイント・イン・タイム表』
- 546 ページの 『レプリカ表』
- 547 ページの 『ユーザー・コピー表』

キャプチャー・コントロール・サーバー上の表、およびその列に関する記述

このセクションでは、キャプチャー・コントロール・サーバー上に保管される各表について、かなり詳しく説明します。また、各表の列のリストやその簡単な説明もあります。コントロール表はアルファベット順にリストされ、列は各表内に現れる順に左から右へリストされています。

ASN.IBMSNAP_CAPSCHEMAS

サーバー: キャプチャー・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。この表を不適切に変更すると、管理ツールの使用中に予期しない結果が生じることがあります。

キャプチャー・スキーマ表は、すべてのキャプチャー・スキーマの名前を保留します。レプリケーション・センターやその他のユーティリティーはこの表を使用して、特定のキャプチャー・コントロール・サーバーのすべての表を迅速に検出できます。ユーザーが新しいキャプチャー・スキーマを作成するたびに、行が 1 つ自動的に挿入されます。

次の 2 つの表は、オペレーティング・システム別のキャプチャー・スキーマ表のレイアウトを示しています。

表 56. OS/400 以外のオペレーティング・システムの場合のキャプチャー・スキーマ表の列

| 列名 | 記述 |
|-----------------|---|
| CAP_SCHEMA_NAME | キャプチャー・スキーマの名前。キャプチャー・スキーマごとに 1 つの行があります。 |

表 57. OS/400 の場合のキャプチャー・スキーマ表の列

| 列名 | 記述 |
|-----------------|---|
| CAP_SCHEMA_NAME | キャプチャー・スキーマの名前。キャプチャー・スキーマごとに 1 つの行があります。 |
| STATUS | このキャプチャー・スキーマによって識別されるキャプチャー・プログラムが実行中かどうかを示すフラグ。 Y キャプチャー・プログラムは実行中です。 N キャプチャー・プログラムは実行されていません。 |

schema.IBMSNAP_AUTHTKN (OS/400)

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

schema.IBMSNAP_AUTHTKN (OS/400)

アプライ修飾子相互参照表は、OS/400 環境でのみ使用されます。この表は、update-anywhere レプリケーション時に、アプライ修飾子により識別される、特定のアプライ・プログラムによって実行されたトランザクションをトラッキングするために使用されます。キャプチャー・プログラムは、ユーザーが設定した保持制限に基づいて、この表の整理を行います。

表 58 では、アプライ修飾子相互参照表の列の要旨を示します。

表 58. アプライ修飾子相互参照表の列

| 列名 | 記述 |
|-------------------|--|
| APPLY_QUAL | トランザクションを処理したアプライ・プログラムを識別するためのアプライ修飾子。この修飾子は、update-anywhere レプリケーションにおいて、アプライ・プログラムが同じ変更を繰り返してレプリケーションしないようにするために使用されます。 |
| IBMSNAP_AUTHTKN | トランザクションに関連付けられたジョブ名。Capture for iSeries はこの列の名前と、トランザクションを発行したジョブの名前を突き合わせて、トランザクションがアプライ・プログラムから発行されたものか、ユーザー・アプリケーションから発行されたものかを判別します。ジョブ名が一致すると、Capture for iSeries は、この表の APPLY_QUAL 列にあるアプライ修飾子を、UOW 表の対応する行の APPLY_QUAL 列にコピーします。名前が一致しない場合、Capture for iSeries は UOW 行の APPLY_QUAL 列を NULL に設定します。この列は自動的に他の表にコピーされません。ユーザー・データ列として選択してコピーする必要があります。 |
| JRN_LIB | トランザクションの発行元のジャーナルのライブラリー名。 |
| JRN_NAME | トランザクションの発行元のジャーナルの名前。 |
| IBMSNAP_LOGMARKER | トランザクションがキャプチャー・コントロール・サーバーでコミットされたおおよその時間。 |

schema.IBMSNAP_CAPENQ (UNIX、Windows、z/OS)

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

キャプチャー・エンキュー表は、DB2 以外のリレーショナル・サーバー、または OS/400 サーバーでは使用されません。

1 つのキャプチャー・スキーマについて、キャプチャー・エンキュー表は次のことを保証します。

- ワークステーション用の DB2 の場合は、1 つのデータベースに対して 1 つのキャプチャー・プログラムのみが実行されていること。
- データを共有しない DB2 for z/OS の場合は、1 つのサブシステムに対して 1 つのキャプチャー・プログラムのみが実行されていること。
- データを共有する DB2 for z/OS の場合は、1 つのデータ共有グループに対して 1 つのキャプチャー・プログラムのみが実行されていること。

キャプチャー・プログラムは実行中、この表を排他的にロックします。

表 59 では、キャプチャー・エンキュー表の要旨を示します。

表 59. キャプチャー・エンキュー表の列

| 列名 | 記述 |
|----------|------------------|
| LOCKNAME | この列にはデータは含まれません。 |

schema.IBMSNAP_CAPMON

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

キャプチャー・プログラムは、インターバルが終了するたびにキャプチャー・モニター表に行を挿入して、操作統計を提供します。レプリケーション・センターはこの表 (およびその他の表) の情報を使用するため、ユーザーはキャプチャー・プログラムの状況をモニターできます。キャプチャー・パラメーター (IBMSNAP_CAPPARMS) 表で、ユーザーが MONITOR_INTERVAL に対して指定した値は、キャプチャー・プログラムがキャプチャー・モニター表に挿入を行う頻度を示し、ユーザーが MONITOR_LIMIT に対して指定した値は、表の行がプルーニングの対象となるまでに、表内に留まる分数を示します。

表 60 では、キャプチャー・モニター表の列の要旨を示します。

表 60. キャプチャー・モニター表の列

| 列名 | 記述 |
|--------------------|--|
| MONITOR_TIME | この表に行が挿入されたときのタイム・スタンプ (キャプチャー・コントロール・サーバーの)。 |
| RESTART_TIME | キャプチャー・プログラムの現在の呼び出しが再始動されたときのタイム・スタンプ。 |
| CURRENT_MEMORY | キャプチャー・プログラムが使用したメモリーの量 (MB 単位)。 |
| CD_ROWS_INSERTED | すべてのソース表について、キャプチャー・プログラムが CD 表に挿入した行数。 |
| RECAP_ROWS_SKIPPED | update-anywhere レプリケーションの場合、キャプチャー・プログラムが処理したが、CD 表に挿入していない行数。キャプチャー・プログラムの登録では、この表に複製された変更でも、このソース・サーバーから発生したものではない変更は、リキャプチャーしないように定義されているため、これらの行はスキップされます。 |
| TRIGR_ROWS_SKIPPED | キャプチャー・プログラムが処理したが、CD 表に挿入していない行数。キャプチャー・プログラムの登録で、特定の行を抑制するようにトリガーが定義されているため、これらの行はスキップされます。 |
| CHG_ROWS_SKIPPED | キャプチャー・プログラムが処理したが、CD 表に挿入していない行数。キャプチャー・プログラムの登録では、登録済みの行で発生した変更のみをキャプチャーするように定義されているため、これらの行はスキップされます。 |
| TRANS_PROCESSED | キャプチャー・プログラムが処理した、ソース・システム上のトランザクションの数。 |
| TRANS_SPILLED | メモリー制限のためにキャプチャー・プログラムがディスクに書き出した、ソース・システム上のトランザクションの数。 |

表 60. キャプチャー・モニター表の列 (続き)

| 列名 | 記述 |
|-------------------|--|
| MAX_TRAN_SIZE | ソース・システムで発生した最大のトランザクション。トランザクション・サイズを知ることによって、メモリー・パラメーターの変更を検討することになります。 |
| LOCKING_RETRIES | デッドロックにより再処理が必要となった回数。 |
| JRN_LIB (OS/400) | キャプチャー・プログラムが処理していたジャーナルのライブラリー名。 |
| JRN_NAME (OS/400) | キャプチャー・プログラムが処理していたジャーナルの名前。 |
| LOGREADLIMIT | 読み取るログ・レコードの数についての内部制限に達したため、トランザクションの構築の完了前にログ読み取りが割り込みを受けた回数。 |
| CAPTURE_IDLE | 処理するものがないため、キャプチャー・プログラムがスリープした回数。 |
| SYNCHTIME | この表にモニター・レコードが挿入されたときに、登録表のグローバル行から読み取られた SYNCHTIME の現行値。 |

schema.IBMSNAP_CAPPARMS

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

この表の情報は、SQL を使って更新できます。

キャプチャー・パラメーター表は、キャプチャー・プログラムの操作をコントロールするためにユーザーが変更できるパラメーターを保持します。これらのパラメーターを定義して、キャプチャー・プログラムがブルーニングを行う前にデータを CD 表および UOW 表の中に保持する時間の長さや、ログ・レコードの処理においてキャプチャー・プログラムに許される遅延時間などを設定できます。ユーザーがこの表のパラメーターを変更しても、キャプチャー・プログラムは始動時にしか変更を読み取りません。

表 61 では、キャプチャー・パラメーター表の列の要旨を示します。

表 61. キャプチャー・パラメーター表の列

| 列名 | 記述 |
|-----------------|---|
| RETENTION_LIMIT | 通常基準に基づいてブルーニングが行われていないときに、CD 表、UOW 表、およびシグナル表の中の行がブルーニングの対象となるまで、表の中に留まる時間の長さ。一般的に CD 行および UOW 行は、すべてのターゲットに適用された後で除去され、シグナル行は、サイクルが完了 (SIGNAL_STATE = C) したときに除去されます。 |
| LAG_LIMIT | ログ・レコードの処理時にキャプチャー・プログラムがシャットダウンせずに処理を遅らせることができる分数。リフレッシュ頻度の高い期間では、リフレッシュよりもフル・リフレッシュの方が経済的です。 |
| COMMIT_INTERVAL | UOW 表および CD 表を含めて、キャプチャー・プログラムがキャプチャー・コントロール表にデータをコミットする頻度 (秒単位)。キャプチャー・スレッドとブルーニング・スレッドの競合を避けるために、この値は DB2 ロックアウトの値よりも小さい値にする必要があります。 |

表 61. キャプチャー・パラメーター表の列 (続き)

| 列名 | 記述 |
|-------------------|---|
| PRUNE_INTERVAL | キャプチャー・プログラムが、不必要になった CD 表、UOW 表、シグナル表、トレース表、およびキャプチャー・モニター表の行を自動的に除去する (AUTOPRUNE = Y) 頻度 (秒単位)。除去するインターバルを短くするとスペースは節約できますが、処理コストが増加します。除去するインターバルを長くすると CD 表および UOW 表のスペースはより多く必要になりますが、処理コストは減少します。 |
| TRACE_LIMIT | ブルーニングの対象となるまでに、行がキャプチャー・トレース (IBMSNAP_CAPTRACE) 表の中に留まる分数。ブルーニング・プロセス時に、分数 (現在のタイム・スタンプからキャプチャー・トレース表に行が挿入された時刻を引いたもの) が TRACE_LIMIT の値を超えると、キャプチャー・トレース表の中の行を削除します。 |
| MONITOR_LIMIT | ブルーニングの対象となるまでに、行がキャプチャー・モニター (IBMSNAP_CAPMON) 表の中に留まる分数。ブルーニング・プロセス時に、分数の値 (現在のタイム・スタンプから MONITOR_TIME を引いたもの) が MONITOR_LIMIT の値を超えると、キャプチャー・モニター表の中の行を削除します。 |
| MONITOR_INTERVAL | モニター・スレッドがキャプチャー・モニター (IBMSNAP_CAPMON) 表に行を追加する頻度 (秒単位)。Capture for iSeries の場合は、120 よりも長いインターバルを入力してください。 |
| MEMORY_LIMIT | キャプチャー・プログラムが使用することを許されているメモリーの量 (MB 単位)。この割り振りがすべて使用されてしまうと、メモリー・トランザクションはファイルに書き出されません。 |
| REMOTE_SRC_SERVER | DB2 レプリケーションの将来のオプション用に予約済み。この列には現在、デフォルト値の NULL が入っています。 |
| AUTOPRUNE | キャプチャー・プログラムが、不必要になった行を CD 表、UOW 表、シグナル表、トレース表、およびキャプチャー・モニター表から自動的に除去するかどうかを示すフラグ。 Y 自動ブルーニングはオン。 N 自動ブルーニングはオフ。 |
| TERM | DB2 が静止したときにキャプチャー・プログラムが終了するかどうかを示すフラグ。 Y DB2 が終了するとキャプチャー・プログラムは終了します。 N キャプチャー・プログラムはアクティブのまま、DB2 が再始動されるのを待ちます。 |
| AUTOSTOP | アクティブ・ログの最後に達したときにキャプチャー・プログラムが変更のキャプチャーを終了するかどうかを示すフラグ。 Y アクティブ・ログの最後に達するとキャプチャー・プログラムは変更のキャプチャーをすぐに終了します。 N アクティブ・ログの最後に達してもキャプチャー・プログラムは実行を続けます。 |
| LOGREUSE | キャプチャー・プログラムがキャプチャー・ログ・ファイルに上書きするか、ファイルに追加するかを示すフラグ。 Y キャプチャー・プログラムは再始動時に、最初にログ・ファイルを削除してから再作成してログ・ファイルを再利用します。 N キャプチャー・プログラムは新しい情報をキャプチャー・ログ・ファイルに追加します。 |

表 61. キャプチャー・パラメーター表の列 (続き)

| 列名 | 記述 |
|---------------------------------------|--|
| LOGSTDOU | <p>キャプチャー・プログラムがログ・ファイル・メッセージを誘導するかどうかを示すフラグ。</p> <p>Y キャプチャー・プログラムは、標準出力 (STDOUT) とログ・ファイルの両方にログ・ファイル・メッセージを誘導します。</p> <p>N キャプチャー・プログラムは、ほとんどのログ・ファイル・メッセージをログ・ファイルにのみ送ります。初期化メッセージは、標準出力 (STDOUT) とログ・ファイルの両方に送られます。</p> |
| SLEEP_INTERVAL (UNIX、Windows、z/OS) | <p>アクティブ・ログの最後に達したとき (UNIX および Windows、または z/OS のデータを共有しない環境の場合)、または戻されたデータの量が不十分なとき (z/OS のデータ共有環境の場合) にキャプチャー・プログラムがスリープする秒数。</p> |
| CAPTURE_PATH | <p>キャプチャー・プログラムからの出力が送信されるパス。</p> |
| STARTMODE | <p>キャプチャー・プログラムの始動時に使用される処理プロシージャ。</p> <p>cold キャプチャー・プログラムは、初期化時に CD 表と UOW 表の中のすべての行を削除します。これらのレプリケーション・ソースに対するすべてのサブスクリプションは、次のアプライ処理サイクル時にフル・リフレッシュされます (つまり、すべてのデータがソース表からターゲット表にコピーされます)。キャプチャー・プログラムがコールド・スタートを試行したときに、フル・リフレッシュが使用不可な場合には、キャプチャー・プログラムは開始されますが、アプライ・プログラムは失敗し、エラー・メッセージが発行されます。</p> <p>warmsi キャプチャー・プログラムはウォーム・スタートします。ただし、キャプチャー・プログラムを初めて開始する場合には、コールド・スタートに切り替えられます。warmsi 開始モードは、キャプチャー・プログラムを初めて開始する場合にのみコールド・スタートが行われるようにします。</p> <p>warmns キャプチャー・プログラムはウォーム・スタートします。ウォーム・スタートできない場合、コールド・スタートに切り替わりません。warmns 開始モードは、予期せずにコールド・スタートが発生することを防止します。これは、修復が必要な、ウォーム・スタートの進行を阻害している問題 (データベースまたは表スペースが使用できないなど) が発生した場合に便利です。ウォーム・スタートした場合、キャプチャー・プログラムは終了したところから処理を再開します。キャプチャー・プログラムの始動後にエラーが発生した場合は、キャプチャー・プログラムは終了し、すべての表はそのまま残されます。</p> <p>warmsa ウォーム・スタート情報が使用可能な場合、キャプチャー・プログラムは前回の実行で終了したところから処理を再開します。キャプチャー・プログラムは、ウォーム・スタートできない場合、すべてのターゲット表をリフレッシュするコールド・スタートに切り替わります。</p> |

schema.IBMSNAP_CAPTRACE (DB2 のみ)

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

キャプチャー・トレース表は、キャプチャー・プログラムから重要なメッセージを保持します。

次の 2 つの表は、オペレーティング・システム別のキャプチャー・トレース表のレイアウトを示しています。

表 62. UNIX、Windows、および z/OS のキャプチャー・トレース表の列

| 列名 | 記述 |
|-------------|---|
| OPERATION | キャプチャー・プログラムの操作のタイプ (たとえば、初期化、キャプチャー、またはエラー条件)。 |
| TRACE_TIME | キャプチャー・トレース表に行が挿入されたときの、キャプチャー・コントロール・サーバーにおける時刻。 |
| DESCRIPTION | メッセージ ID とメッセージ・テキスト。エラー・メッセージ、警告メッセージ、または情報メッセージです。この列に入れられるテキストは英語のみです。 |

表 63. OS/400 のキャプチャー・トレース表の列

| 列名 | 記述 | | | | | | | | |
|--------------|--|----|----|------|------------------------|-------|--------------|-------|-------|
| OPERATION | 初期化、キャプチャー、またはエラー条件など、キャプチャー・プログラムで実行された操作のタイプ。 | | | | | | | | |
| TRACE_TIME | キャプチャー・トレース表に行が挿入された時刻。トレース限度ブルーニングの対象となる TRACE_TIME 行は、キャプチャー・プログラムが CD 表および UOW 表のブルーニングを行うときに削除されます。 | | | | | | | | |
| JOB_NAME | このトレース・エントリーを書き込んだジョブの完全修飾名。 | | | | | | | | |
| | <table border="1"> <thead> <tr> <th>位置</th> <th>記述</th> </tr> </thead> <tbody> <tr> <td>1-10</td> <td>QDPRCTL5 またはジャーナル・ジョブ名</td> </tr> <tr> <td>11-20</td> <td>始動したユーザーの ID</td> </tr> <tr> <td>21-26</td> <td>ジョブ番号</td> </tr> </tbody> </table> | 位置 | 記述 | 1-10 | QDPRCTL5 またはジャーナル・ジョブ名 | 11-20 | 始動したユーザーの ID | 21-26 | ジョブ番号 |
| 位置 | 記述 | | | | | | | | |
| 1-10 | QDPRCTL5 またはジャーナル・ジョブ名 | | | | | | | | |
| 11-20 | 始動したユーザーの ID | | | | | | | | |
| 21-26 | ジョブ番号 | | | | | | | | |
| JOB_STR_TIME | JOB_NAME 列で指定されたジョブの開始時刻。 | | | | | | | | |
| DESCRIPTION | メッセージ ID とメッセージ・テキスト。メッセージ ID は DESCRIPTION 列の最初の 7 文字です。メッセージ・テキストは、DESCRIPTION 列の位置 9 から始まります。 | | | | | | | | |

schema.CCD_table (DB2 以外)

サーバー: キャプチャー・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。この表を不適切に変更すると、データの損失が生じることがあります。

キャプチャー・コントロール・サーバー上の整合変更データ (CCD) 表は、DB2 以外のソースで発生した変更に関する情報と、これらの変更の順序を識別するための追加の列を含む表です。キャプチャー・コントロール・サーバー上の CCD 表は、アプライ・プログラム以外のプログラムによってデータが取り込まれる表です。次のいずれかになります。

- DB2 以外のリレーショナル・ソースの内部 CCD 表。

変更キャプチャー・レプリケーションの場合、DB2 以外のリレーショナル・ソースで更新が行われると、キャプチャー・プログラムは挿入変更を起動します。このタイプの CCD 表の名前は、登録 (IBMSNAP_REGISTER) 表の中で、変更元のレプリケーシ

ョン・ソースと同じ行に保管されます。この表は、DB2 以外のリレーショナル・ソースを登録したときに作成されるブルーニング・トリガーによって自動的にブルーニングが行われます。

- 非リレーショナルおよびマルチベンダーのデータ用の外部 CCD 表。

外部プログラムは、DB2 レプリケーションにより、レプリケーション・ソースとして使用される CCD 表を作成できます。これらの外部プログラムは、IMS データのコピーをリレーショナル・データベース内に再作成できるように、CCD 表の中の IMS 変更をキャプチャーします。外部プログラムは、コントロール列の正しい値を初期化、保守および提供する必要があります。IMS DataPropagator または DataRefresher などのプログラムによって保守されていない、外部で取り込まれる CCD 表がある場合には、アプライ・プログラムがソースとして CCD 表を読み取り、正しく機能することができるように、ユーザー自身でこれらの表を保守する必要があります。外部で取り込まれる CCD の保守の詳細は、67 ページの『CCD 表をソースとして保守する (IMS)』を参照してください。

サブスクリプション・セット・メンバー内のターゲットとしての CCD 表の詳細は、543 ページの『整合変更データ (CCD) 表』を参照してください。

表 64 では、CCD 表の列の要旨を示します。

表 64. CCD 表の列

| 列名 | 記述 |
|-----------------------|---|
| IBMSNAP_INTENTSEQ | 変更を一意的に識別するシーケンス番号。この値はグローバルに昇順です。 |
| IBMSNAP_OPERATION | レコードの操作のタイプを示すフラグ。 I 挿入 U 更新 D 削除 |
| IBMSNAP_COMMITSEQ | トランザクションの順序を指定するシーケンス番号。 |
| IBMSNAP_LOGMARKER | データがコミットされた時刻。 |
| user key columns | CCD 表が圧縮されている場合、この列には、ターゲット・キーを構成する列が含まれます。 |
| user non-key columns | ソース表からの非キー・データ列。ソース表の中の列名とこれらの列名が一致している必要はありませんが、データ・タイプは一致している必要があります。 |
| user computed columns | SQL 式から派生したユーザー定義の列。ソース・データ・タイプを別のターゲット・データ・タイプに変換するために、SQL 関数で算出列を使用することができます。 |

関連資料:

- 543 ページの 『整合変更データ (CCD) 表』

schema.CD_table

サーバー: キャプチャー・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。この表を不適切に変更すると、データの損失が生じることがあります。

変更データ (CD) 表は、レプリケーション・ソースに対して行われたすべてのコミット済み変更を記録します。CD 表のプルーニングは、プルーニング・セット (IBMSNAP_PRUNE_SET) 表によって調整されます。(CD 表の整理についての詳細は、503 ページの『*schema.IBMSNAP_PRUNE_SET*』を参照してください。) CD 表はキャプチャー・コントロール表とは異なり、ユーザーがレプリケーション・ソースを定義したときに作成されます。キャプチャー・コントロール・サーバー用のコントロール表を作成するときに自動的に作成されるものではありません。

表 65 では、CD 表の列の要旨を示します。

表 65. CD 表の列

| 列名 | 記述 |
|---------------------------------|---|
| IBMSNAP_COMMITSEQ | 取り込まれたコミット・ステートメントのログ・シーケンス番号。UOW 表にもあるこの列は、アプライ・プログラムが、UOW 表と CD 表を結合する必要なしに、ユーザー・コピー・ターゲット表を処理できるようにするために、CD 表に含められているものです。CD 表と UOW 表の結合が必要な場合は、IBMSNAP_COMMITSEQ 列を使用して結合が行われず。 |
| IBMSNAP_INTENTSEQ | 変更 (挿入、更新、または削除) のログ・レコードのログ・シーケンス番号。この値はグローバルに昇順です。更新を削除と挿入のペアとして処理するようにユーザーが選択した場合には、削除行の IBMSNAP_INTENTSEQ 値は、挿入行の対応する値よりも、わずかに小さくなるように構成されます。 |
| IBMSNAP_OPERATION | レコードの操作のタイプを示すフラグ。 I 挿入 U 更新 D 削除 |
| <i>user column after-image</i> | 多くの場合、変更後イメージ列には、変更発生後のソース列にある値が含まれます。この列は、ソース列と同じ名前、データ・タイプ、および NULL 属性になります。更新の場合、この列は、更新されたデータの新しい値を反映します。削除の場合、この列は、削除されたデータの値を反映します。挿入の場合、この列は、挿入されたデータの値を反映します。 |
| <i>user column before-image</i> | この列は、ソースが変更前イメージ列の値を含むものとして登録されている場合のみ CD 表の中に存在します。多くの場合、変更前イメージ列には、変更発生前のソース列にあった値が含まれます。この列は、ソース列と同じ名前になり、登録 (IBMSNAP_REGISTER) 表の中の BEFORE_IMG_PREFIX 列にある値が接頭部として使用されます。また、データ・タイプもソース列と同じになります。しかし、ソース列が NULL 属性であるかどうかに関係なく、挿入操作ではいつでも NULL 値が許されます。更新の場合、この列は、更新されたデータを反映します。削除の場合、この列は、削除されたデータを反映します。挿入の場合、この列は NULL になります。 |

schema.IBMSNAP_PRUNCNTL

schema.IBMSNAP_PRUNCNTL

サーバー: キャプチャー・コントロール・サーバー

プルーニング・コントロール表

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

プルーニング・コントロール表には、このキャプチャー・スキーマに対して定義されている、すべてのサブスクリプション・セット・メンバーに関する詳細情報が入っています。この表は、プルーニング時にプルーニング・セット (IBMSNAP_PRUNE_SET) 表と一緒に使用されます。また、アプライ・プログラムとキャプチャー・プログラムの間で初期化ハンドシェイク・プロセス時にも使用されます。

DB2 ソースの場合は、**prune** コマンドを発行してプルーニングを呼び出すこともできますし、自動的に起動することもできます。AUTOPRUNE を設定するためにキャプチャー・パラメーターを使用する方法についての詳細は、496 ページの

『*schema.IBMSNAP_CAPPARMS*』を参照してください。DB2 以外のリレーショナル・ソースの場合のプルーニングは、ユーザーがソースを登録したときに作成されたプルーニング・トリガーを使用して行われます。

表 66 では、プルーニング・コントロール表の列の要旨を示します。

表 66. プルーニング・コントロール表の列

| 列名 | 記述 |
|------------------|---|
| TARGET_SERVER | このメンバーのターゲット表またはビューが置かれているサーバー名。 |
| TARGET_OWNER | このメンバーのターゲット表またはビューの高位修飾子。 |
| TARGET_TABLE | このメンバーのターゲット表またはビューの名前。 |
| SYNCHTIME | キャプチャー・プログラムは、アプライ・プログラムとの初期化ハンドシェイク・プロセス時にこのタイム・スタンプを設定します。この値は、CAPSTART シグナル挿入のトランザクションと関連付けられたコミット・ログ・レコードのタイム・スタンプによって決まります。この値は、その後初期化処理が行われないかぎり、再度更新されることはありません。 |
| SYNCHPOINT | キャプチャー・プログラムは、アプライ・プログラムとの初期化ハンドシェイク・プロセス時にこの値を設定します。この値は、CAPSTART シグナル挿入のトランザクションと関連付けられたコミット・ログ・レコードのログ・シーケンス番号によって決まります。この値は、その後初期化処理が行われないかぎり、再度更新されることはありません。 |
| SOURCE_OWNER | このメンバーのソース表またはビューの高位修飾子。 |
| SOURCE_TABLE | このメンバーのソース表またはビューの名前。 |
| SOURCE_VIEW_QUAL | この列は、SOURCE_OWNER および SOURCE_TABLE 列に同じ値を持つ、複数の異なるソース・ビューがある場合の複数登録をサポートするために使用されます。この値は、ソースとして定義されている物理表の場合は 0 に、ソースとして定義されているビューの場合は 0 より大きい値に設定されます。 |
| APPLY_QUAL | このメンバーを処理しているアプライ・プログラムを識別するためのアプライ修飾子。 |
| SET_NAME | このサブスクリプション・セット・メンバーが所属するサブスクリプション・セットの名前。 |
| CNTL_SERVER | APPLY_QUAL により識別される、このアプライ・プログラムのアプライ・コントロール表が置かれているサーバーの名前。 |

表 66. ブルーニング・コントロール表の列 (続き)

| 列名 | 記述 |
|-------------------|--|
| TARGET_STRUCTURE | ターゲット表またはビューのタイプを示す値。 <ol style="list-style-type: none"> 1 ソース表 3 CCD 表 4 ポイント・イン・タイム表 5 基礎集約表 6 変更集約表 7 レプリカ表 8 ユーザー・コピー表 |
| CNTL_ALIAS | CNTL_SERVER 列で指定されているアプライ・コントロール・サーバーに対応する DB2 Universal Database 別名。 |
| PHYS_CHANGE_OWNER | この特定のサブスクリプション・セット・メンバーのソースに関連付けられた登録 (IBMSNAP_REGISTER) 表の PHYS_CHANGE_OWNER 列の中の値。 |
| PHYS_CHANGE_TABLE | この特定のサブスクリプション・セット・メンバーのソースに関連付けられた登録 (IBMSNAP_REGISTER) 表の PHYS_CHANGE_TABLE 列の中の値。 |
| MAP_ID | この表で、より短い、簡単に使用できる索引を提供する一意性の因子。シグナル表への CAPSTART 挿入を、ブルーニング・コントロール表の中の適切な行に関連付けるためにも使用されます。 |

schema.IBMSNAP_PRUNE_LOCK

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

ブルーニング・ロック表は、コールド・スタート、または保持制限ブルーニング時に、CD 表へのアクセスを直列化するために使用されます。この表は、これらの重要なフェーズでアプライ・プログラムが CD 表にアクセスしないようにするものです。この表には行がありません。

schema.IBMSNAP_PRUNE_SET

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

schema.IBMSNAP_PRUNE_SET

ブルーニング・セット表は、各サブスクリプション・セットについてキャプチャー・プログラムおよびアプライ・プログラムの進行をトラッキングすることにより、CD 表および UOW 表のブルーニングの調整を行います。1 つのソースからターゲットへのマッピングに対して 1 つの行があるブルーニング・コントロール (IBMSNAP_PRUNCNTL) 表とは異なり、ブルーニング・セット表には、1 つのサブスクリプション・セットに対して 1 つの行があります。

表 67 では、ブルーニング・セット表の列の要旨を示します。

表 67. ブルーニング・セット表の列

| 列名 | 記述 |
|---------------|--|
| TARGET_SERVER | このセットのターゲット表またはビューが置かれているサーバー名。 |
| APPLY_QUAL | このセットを処理しているアプライ・プログラムを識別するためのアプライ修飾子。 |
| SET_NAME | サブスクリプション・セットの名前。 |
| SYNCHTIME | アプライ・プログラムはこの列を使用して進行状況を記録します。サブスクリプション・セットのデータの処理が、このタイム・スタンプまで終了していることを示します。 |
| SYNCHPOINT | アプライ・プログラムはこの列を使用して進行状況を記録します。サブスクリプション・セットのデータの処理が、この同期点の値まで終了していることを示します。 |

schema.IBMSNAP_REG_EXT (OS/400)

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

登録拡張表は、登録 (IBMSNAP_REGISTER) 表の補足情報を提供する OS/400 固有の表です。登録拡張表には、登録表のすべての行に対して、OS/400 固有の列をいくつか追加で含む、対応する行があります。

この表は、登録表のトリガー・プログラム (ライブラリー QDP4 の QZSNJLV8) によって保守されます。トリガーは、登録表が作成されるときに定義されます。

この表の情報は、OS/400 サーバー上でレプリケーション・ソースがどこでどのように定義されているかをトラッキングするために使用されます。

表 68 では、登録拡張表の列の要旨を示します。

表 68. 登録拡張表の列

| 列名 | 記述 |
|--------------|---|
| VERSION | ソースの登録に使用された DB2 DataPropagator for iSeries のバージョン。 |
| SOURCE_OWNER | ユーザーにより登録されたソース表またはビューの高位修飾子。 |
| SOURCE_TABLE | ユーザーにより登録されたソース表またはビューの名前。 |

表 68. 登録拡張表の列 (続き)

| 列名 | 記述 |
|--------------------|--|
| SOURCE_NAME | コマンドの発行に使用されたソース表またはビューの 10 文字のシステム名。 |
| SOURCE_MBR | ジャーナル・エントリーの受信 (RCVJRNE) コマンドの発行および ALIAS サポートに使用されるソース表メンバーの名前。 |
| SOURCE_TABLE_RDB | リモート・ジャーナルを使用する場合、この列には、ソース表が実際に置かれているシステムのデータベース名が含まれます。ローカル・ジャーナルの場合、この列は NULL になります。 |
| JRN_LIB | ソース表が使用するジャーナルのライブラリー名。 |
| JRN_NAME | ソース表が使用するジャーナルの名前。この列で、アスタリスクの後ろに 9 つのブランクが続くときには、ソース表が現在ジャーナルの中にないことを意味します。この場合キャプチャー・プログラムは、このソースのデータをキャプチャーすることはできません。 |
| FR_START_TIME | アプライ・プログラムがフル・リフレッシュの実行を開始した時刻。 |
| SOURCE_VIEW_QUAL | 登録表内の類似した列を突き合わせることによって、サブスクリプションのビューをサポートします。この値は、ソースとして定義されている物理表の場合は 0 に、ソースとして定義されているビューの場合は 0 より大きい値に設定されます。SOURCE_OWNER および SOURCE_TABLE 列の値が同じである、複数の異なるソース・ビューについて複数のサブスクリプションをサポートするには、この列が必要です。 |
| CMT_BEHAVIOR_CASE | ソース表を更新するアプリケーション・プログラムがコミットメント・コントロールを使用する方法を表す整数。キャプチャー・プログラムはこの値を使用して、構成済みであるが、CD 表にまだ書き込んでいない CD 行に使用するメモリーを管理します。 <ul style="list-style-type: none"> -1 アプリケーションのコミットメント・コントロール・パターンはまだ設定されていません。これは、この列の初期値です。 0 ソースを更新するアプリケーションではコミットメント・コントロールが使用されていません。 1 ソースを更新するアプリケーションはすべて、コミットメント・コントロールを使用します。このため、コミットメント・コントロール下の同じソース表が、2 つの異なるアプリケーションから同時に更新されることはありません。 2 ソースを更新する同時アプリケーションの中にはコミットメント・コントロールを使用するものも使用しないものもあります。2 つのアプリケーションが同時にコミットメント・コントロールを使用してソース表を更新する可能性があります。 |
| MAX_ROWS_BTWN_CMTS | キャプチャー・プログラムがデータを CD 表にコミットする前に処理できる行の最大数。 |

schema.IBMSNAP_REGISTER

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

登録表は、レプリケーション・ソース表の名前、属性、および、これらに関連付けられた CD 表および CCD 表の名前など、レプリケーション・ソースに関する情報を保持します。キャプチャー・プログラムで処理されるように、新しいレプリケーション・ソース表またはビューが定義されるたびに、この表に行が 1 つ自動的に挿入されます。

レプリケーション・ソースの定義を調べる必要があるときには、登録表を使用します。

表 69 では、登録表の列の要旨を示します。

表 69. 登録表の列

| 列名 | 記述 |
|------------------|---|
| SOURCE_OWNER | ユーザーにより登録されたソース表またはビューの高位修飾子。 |
| SOURCE_TABLE | ユーザーにより登録されたソース表またはビューの名前。 |
| SOURCE_VIEW_QUAL | この列は、SOURCE_OWNER および SOURCE_TABLE 列に同じ値を持つ、複数の異なるソース・ビューがある場合の複数登録をサポートするために使用されます。この値は、ソースとして定義されている物理表の場合は 0 に、ソースとして定義されているビューの場合は 0 より大きい値に設定されます。 |
| GLOBAL_RECORD | この行がグローバル・レコードかどうかを示すフラグ。グローバル・レコードの場合は、進行を示すために、SYNCHPOINT および SYNCHTIME 列がキャプチャー・プログラムにより設定されます。キャプチャー・プログラムをまだ実行していないか、キャプチャー・プログラムを実行するつもりがない場合は (フル・リフレッシュのみのレプリケーションとしてソースが定義されているか、DB2 以外のリレーショナル・ソースであるため)、グローバル・レコードはありません。 Y この行はグローバル・レコードです。 N この行はグローバル・レコードではありません。 |
| SOURCE_STRUCTURE | ソース表またはビューの構造を示す値 1 ユーザー表 3 CCD 表 4 ポイント・イン・タイム表 5 基礎集約表 6 変更集約表 7 レプリカ表 8 ユーザー・コピー表 |
| SOURCE_CONDENSED | ソース表がコンデンス表であるかどうかを示すフラグ。圧縮されている場合は、同じキーを持つすべての行が、1 つの行に圧縮されます。 Y ソースは圧縮されています。 N ソースは圧縮されていません。 A ソースは基礎集約表または変更集約表です。 |
| SOURCE_COMPLETE | 主キー値を持つ行がソース表にどのように保管されるかを示すフラグ。 Y ソース表は、関係するそれぞれの主キー値につき 1 行を保持します。 N ソース表は、主キー値の行のサブセットを保持します。 |

表 69. 登録表の列 (続き)

| 列名 | 記述 |
|-------------------|---|
| CD_OWNER | <p>ソースの CD 表の高位修飾子。</p> <p>ソースが表の場合 外部 CCD 表ではないすべての登録済みソース表の場合、この列は、このソース表に関連付けられた CD 表の高位修飾子を保持します。</p> <p>ソースがビューの場合 この列は、CD ビューの高位修飾子を保持します。</p> <p>ソースが外部 CCD 表の場合 この列は NULL です。</p> |
| CD_TABLE | <p>ソースの CD 表の名前。</p> <p>ソースが表の場合 外部 CCD 表ではないすべての登録済みソース表の場合、この列には、このソース表の取り込まれた更新を保留する CD 表の名前が入ります。</p> <p>ソースがビューの場合 この列は、CD ビューの名前を保持します。</p> <p>ソースが外部 CCD 表の場合 この列は NULL です。</p> |
| PHYS_CHANGE_OWNER | <p>アプライ・プログラムが変更キャプチャー・レプリケーションに使用する、表またはビューの高位修飾子。</p> <p>ソースが表の場合 外部 CCD 表ではないすべての登録済みソース表の場合、この列は、このソース表に関連付けられた物理 CD 表の高位修飾子を保持します。</p> <p>ソースがビューの場合 この列は、このソース・ビューに関連付けられた物理 CD 表の高位修飾子を保持します。</p> <p>ソースが外部 CCD 表の場合 この列は、外部 CCD 表の高位修飾子を保持します。</p> |
| PHYS_CHANGE_TABLE | <p>アプライ・プログラムが変更キャプチャー・レプリケーションに使用する、表またはビューの名前。</p> <p>ソースが表の場合 外部 CCD 表ではないすべての登録済みソース表の場合、この列は、このソース表に関連付けられた物理 CD 表の名前を保持します。</p> <p>ソースがビューの場合 この列は、このソース・ビューに関連付けられた物理 CD 表の名前を保持します。</p> <p>ソースが外部 CCD の場合 この列は、外部 CCD 表の名前を保持します。</p> |
| CD_OLD_SYNCHPOINT | <p>この列は、アプライ・プログラムとキャプチャー・プログラムの間の初期ハンドシェイクに使用されます。キャプチャー・プログラムはその後、ソース・ログのこのログ・シーケンス番号からデータのキャプチャーを開始します。この列は、CD 表で保持制限ブルーニングが行われたことを示すためにも使用されます。この値が NULL の場合、登録は非アクティブです。</p> |

表 69. 登録表の列 (続き)

| 列名 | 記述 |
|--------------------|---|
| CD_NEW_SYNCHPOINT | この列は、キャプチャー・プログラムが CD 表に新しい行を挿入するにつれて進められます。アプライ・プログラムはこの列を使用して、複製する新しい変更があるかどうかを確認します。 |
| DISABLE_REFRESH | フル・リフレッシュが使用可能かどうかを示すフラグ。 0 フル・リフレッシュを使用可能にします。 1 フル・リフレッシュを使用不可にします。 |
| CCD_OWNER | ソースに内部 CCD 表が関連付けられている場合、この列は、内部 CCD の高位修飾子を保持します。外部 CCD 表の場合、この列は NULL になります。 |
| CCD_TABLE | ソースに内部 CCD 表が関連付けられている場合、この列は、内部 CCD の名前を保持します。外部 CCD 表の場合、この列は NULL になります。 |
| CCD_OLD_SYNCHPOINT | CCD 表が再初期化されたときのログ・シーケンス番号。この列は、CCD のフル・リフレッシュ処理に関連しています。この列の値を変更する必要があるのは、CCD 表が初期に、またはその後、フル・リフレッシュされた場合だけです。この値は、CCD 表に残っているどの行よりも古いものである可能性があります。この列が保守されないと、CCD 表をレプリケーション・ソースとして使用するアプライ・プログラムは、CCD 表が再初期化されたかどうか分からないので、CCD ソースの完全なコピーを再初期化できません。 |
| SYNCHPOINT | グローバル行の場合 (GLOBAL_RECORD = Y)、同期点は、キャプチャー・プログラムによって処理された最後のログまたはジャーナル・レコードのログ・シーケンス番号を表します。CCD 表 (内部または外部) に関する登録情報を含む IBMSNAP_REGISTER 表の中の行の場合、CCD 表の中に使用可能な新しいデータがあることを示すために、同期点の値は、CCD 表を保守するプログラムによって進められます。 |
| SYNCHTIME | グローバル行の場合 (GLOBAL_RECORD = Y)、同期時刻は、キャプチャー・プログラムによって処理された最後のログまたはジャーナル・レコードのタイム・スタンプを表します。キャプチャー・プログラムが DB2 ログの最後に達すると、同期時刻は現在の DB2 タイム・スタンプまで進められます。CCD 表 (内部または外部) に関する登録情報を含む IBMSNAP_REGISTER 表の中の行の場合、CCD 表の中の使用可能データの現行性 (currency) を示すために、同期時刻の値は、CCD 表を保守するプログラムによって進められます。 |
| CCD_CONDENSED | このソースに関連付けられた内部 CCD が圧縮されているかどうかを示すフラグ。圧縮されている場合は、同じキーを持つすべての行が、1 つの行に圧縮されます。 Y 内部 CCD は圧縮されています。 N 内部 CCD は圧縮されていません。 NULL このソースに対して内部 CCD 表が定義されていません。 |
| CCD_COMPLETE | このソースに関連付けられた内部 CCD 表が完成しているかどうかを示すフラグ。完成しているとは、ソース表からのすべての行が初期に含まれていることを意味します。 N 内部 CCD は完成していません。 NULL このソースに対して内部 CCD 表が定義されていません。 |
| ARCH_LEVEL | 行での定義の構造レベル。このレベルは、IBM で定義されていて、バージョン 8 では 0801 です。 |
| DESCRIPTION | レプリケーション・ソースの記述。 |

表 69. 登録表の列 (続き)

| 列名 | 記述 |
|--------------------|--|
| BEFORE_IMG_PREFIX | <p>CD 表の中の変更前イメージ列の名前を識別する 1 文字の接頭部。変更前イメージの接頭部と CD 列名の組み合わせは判別可能なものにする必要があります。つまり、接頭部の付いた CD 列名は、現在の列名、そして想定される変更後イメージ列の名前と同じであってはなりません。BEFORE_IMG_PREFIX の長さ (バイト単位) は次のとおりです。</p> <p>1 ASCII または EBCDIC の 1 バイトの接頭文字の場合。</p> <p>2 ASCII の 2 バイトの接頭文字の場合。</p> <p>4 EBCDIC DBCS 接頭文字の場合。この長さは、シフトインおよびシフトアウト文字を見込んでいます。</p> |
| CONFLICT_LEVEL | <p>このソースの競合検出のレベルを示すフラグ。</p> <p>0 アプライ・プログラムは競合をチェックしません。更新の競合を防止するために、アプリケーション側でデータ整合性を適用する必要があります。</p> <p>1 カスケード・トランザクション拒否の標準検出。アプライ・プログラムは、これまでに取り込まれた変更に基づいて競合をチェックします。アプライ・プログラムは、レプリカに競合するトランザクションがあればそれを元に戻し、競合するトランザクションに対して従属関係を持つトランザクションがあれば、それも元に戻します。アプライ・プログラムが競合検出を開始した後で取り込まれた変更は、このアプライ・サイクルではチェックされません。</p> <p>2 カスケード・トランザクション拒否の拡張検出。アプライ・プログラムは、キャプチャー・プログラムがログまたはジャーナルからすべての変更をキャプチャーするまで待ち (SYNCHTIME 列の記述を参照)、1 に設定された場合と同じように、標準競合検出を行います。アプライ・プログラムはエンキュー中、競合検出プロセス中に変更が行われないように、ソース表にロックをかけます。</p> |
| CHG_UPD_TO_DEL_INS | <p>キャプチャー・プログラムが CD 表に更新を保管する方法を示すフラグ。</p> <p>Y キャプチャー・プログラムは、1 つは削除用、1 つは挿入用の 2 つの行を CD 表で使用して更新を保管します。アプライ・プログラムは最初に削除を処理し、次に挿入を処理します。このフラグを Y に設定すると、レプリケーション・ソースに対するすべての更新は、2 つの行を使用して CD 表に保管されます。このフラグは、パーティション化された列、またはサブスクリプション・セット述部から参照される列の更新が正しく処理されるようにします。</p> <p>N ソース表に対する更新はそれぞれ、CD 表の中の 1 つの行に保管されます。</p> |
| CHGONLY | <p>キャプチャー・プログラムがソースで発生したすべての変更をキャプチャーするか、登録済み列で発生した変更のみをキャプチャーするかを示すフラグ。一般的には、キャプチャー・プログラムによって CD 表に挿入される行数を最小化するために、このオプションは Y に設定されます。しかし、ソース表の中のどの行が更新されたかを正確にトラッキングするために、このオプションを N に設定することも考えられます。たとえば、ソース表の中のどの行が更新されたかを監査するために、主キー列の値だけをキャプチャーできます。</p> <p>Y キャプチャー・プログラムは、ソース表の登録済み列で発生した変更のみをキャプチャーします。</p> <p>N キャプチャー・プログラムは、ソース表すべての列で発生した変更をキャプチャーします。</p> |

表 69. 登録表の列 (続き)

| 列名 | 記述 |
|---------------|---|
| RECAPTURE | <p>この列は update-anywhere レプリケーションに使用されます。表またはビューで行われた変更のリキャプチャーを行い、他の表またはビューに転送するかどうかを示すフラグを保持します。</p> <p>マスター側の表の場合:</p> <p>N レプリカから適用されたマスターに対する更新のリキャプチャーは行われず、他のレプリカに複製されません。</p> <p>Y レプリカから適用されたマスターに対する更新は他のレプリカに複製されます。</p> <p>レプリカ側の表の場合:</p> <p>Y マスターから適用されたレプリカに対する更新のリキャプチャーが行われ、このレプリカをソースとして使用する他の表に複製するために使用できます。</p> <p>N マスターから適用されたレプリカに対する更新のリキャプチャーは行われません。</p> |
| OPTION_FLAGS | DB2 レプリケーションの将来のオプション用に予約済み。この列には現在、デフォルト値のNNNNが入っています。 |
| STOP_ON_ERROR | <p>キャプチャー・プログラムが、始動、開始、再開、または CD 表への行の挿入時にエラーを検出したときに、終了するか、登録の処理を停止するだけかを示すフラグ。</p> <p>Y キャプチャー・プログラムは、始動、開始、再開、または CD 表への行の挿入時にエラーを検出すると終了します。</p> <p>N キャプチャー・プログラムは、始動、開始、再開、または CD 表への行の挿入時にエラーを検出すると、登録を停止しますが、終了はしません。プログラムはその他の登録処理を続けます。</p> |
| STATE | <p>登録の状態を示すフラグ。</p> <p>S キャプチャー・プログラムはこの登録の処理を停止しました。アプライ・プログラムは、登録が修復され、I (非アクティブ) 状態になるまで、この登録を処理しません。</p> <p>A 登録はアクティブです。</p> <p>I 登録は非アクティブです。</p> |
| STATE_INFO | キャプチャー・プログラムが登録の処理を停止した場合、この列は、この障害に関して発行されたエラー・メッセージを保持します。 |

schema.IBMSNAP_REG_SYNCH (DB2 以外のリレーショナル)

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

登録同期表は、アプライ・プログラムが DB2 以外のリレーショナル・データ・ソースからデータをフェッチする準備をするときに、登録 (IBMSNAP_REGISTER) 表の中のすべての行の SYNCHPOINT 値の更新を開始するために、更新トリガーを使用します。

表 70 では、登録同期表の列の要旨を示します。

表 70. 登録同期表の列

| 列名 | 記述 |
|------------|--|
| TRIGGER_ME | 登録表の中のすべての行の SYNCHPOINT 値を更新するためにトリガーが開始されたかどうかを示す Y というフラグ。 |
| TIMESTAMP | Microsoft SQL Server および Sybase のソースの場合、この列は、表のタイム・スタンプ列で更新が発生したときにシステムから生成される固有の番号を保持します。この値は、登録 (IBMSNAP_REGISTER) 表に記録される SYNCHPOINT 値を生成するために使用されます。 |

schema.IBMSNAP_RESTART

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。ユーザーがこの表から行を削除した場合、キャプチャー・プログラムはコールド・スタートせざるをえません。

再始動表には、必要とされる中で一番古いログまたはジャーナル・レコードからキャプチャー・プログラムを再始動できるようにするための情報が含まれます。この表は、DB2 レプリケーションのバージョン 7 およびそれ以前のバージョンのウォーム・スタート表を置き換えるものです。この表は、コミット・ポイントのたびに更新される行を保持します。このため、キャプチャー・プログラムは、すでに処理済みで、CD 表および UOW 表に挿入済みの情報のリキャプチャーを行う必要がなく、常に正しい場所から正確に再始動できます。

キャプチャー・プログラムをまだ開始したことがない場合は、この表は空であるため、キャプチャー・プログラムはコールド・スタートを実行する必要があります。

次の 2 つの表は、オペレーティング・システム別の再始動表のレイアウトを示していません。

表 71. UNIX、Windows、および z/OS の再始動表の列

| 列名 | 記述 |
|------------------|---|
| MAX_COMMITSEQ | キャプチャー・プログラムが CD 表および UOW 表にコミット済みのログ・シーケンス番号 (IBMSNAP_COMMITSEQ) の最大値。 |
| MAX_COMMIT_TIME | MAX_COMMITSEQ 列のログ・シーケンス番号に関連付けられたタイム・スタンプ。 |
| MIN_INFLIGHTSEQ | キャプチャー・プログラムがウォーム・リスタート時に開始する時点のログ・シーケンス番号。この値は、キャプチャー・プログラムが見つけた、コミットまたはアポート・レコードがまだ検出されていない、一番若いログ・シーケンス番号です。 |
| CURR_COMMIT_TIME | この表がキャプチャー・プログラムによって更新されたときの、ローカルの現行タイム・スタンプ。 |

表 71. UNIX、Windows、および z/OS の再始動表の列 (続き)

| 列名 | 記述 |
|-------------------|---|
| CAPTURE_FIRST_SEQ | 最後にコールド・スタートを実行したときにキャプチャー・プログラムの開始時点であったリカバー・ログに関連付けられたログ・シーケンス番号。この値は、キャプチャー・プログラムでコールド・スタートを実行せざるをえなくなるようなデータベース RESTORE が発生したかどうかを検出してユーザーに警告するために使用されます。このような事態は、データベース・ログ・マネージャーが特定の RESTORE 操作時にログ・シーケンス番号を再利用することがあるために発生します。 |

OS/400 の場合、再始動表は **RCVJRNE** (ジャーナル・エントリーの受信) コマンドの開始時刻を判別するために使用されます。1 つのレプリケーション・ソース、またはレプリケーション・ソースのグループで使用されるジャーナルごとに、再始動表に行が 1 つ挿入されます。

表 72. OS/400 の再始動表の列

| 列名 | 記述 |
|-------------------|--|
| MAX_COMMITSEQ | UOW 表からの最新のコミットのジャーナル・レコード番号。 |
| MAX_COMMIT_TIME | MAX_COMMITSEQ 列のジャーナル・レコード番号に関連付けられたタイム・スタンプ、または、キャプチャー・プログラムがログの処理を終了し、実行する処理がない場合は、現在のタイム・スタンプ。 |
| MIN_INFLIGHTSEQ | キャプチャー・プログラムがウォーム・リスタート時に開始する時点のログ・シーケンス番号。 |
| CURR_COMMIT_TIME | この表が更新された時点の現行タイム・スタンプ。 |
| CAPTURE_FIRST_SEQ | キャプチャー・プログラムがコールド・スタート後に開始する時点のジャーナル・レコード番号。 |
| UID | UOW 表の IBMSNAP_UOWID 列の内容で接頭部として使用される固有番号。 |
| SEQNBR | キャプチャー・プログラムが処理した最後のジャーナル・エントリーのシーケンス番号。 |
| JRN_LIB | キャプチャー・プログラムが処理しているジャーナルのライブラリー名。 |
| JRN_NAME | キャプチャー・プログラムが処理しているジャーナルの名前。 |
| STATUS | キャプチャー・プログラムが特定のジャーナル・ジョブを処理しているかどうかを示すフラグ。 Y キャプチャー・プログラムはジャーナル・ジョブを処理しています。 N キャプチャー・プログラムはジャーナル・ジョブを処理していません。 |

schema.IBMSNAP_SEQTABLE (Informix)

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

順序表は、Informix 表のログ・シーケンス番号と同等のものとして DB2 レプリケーションから使用される一連の固有番号を保持します。これらの固有 ID は、キャプチャー

ー・プログラム、アプライ・プログラム、およびレプリケーション・アラート・モニターが最後のサイクル時にどこまで処理を進めたかを連絡し合えるように、登録 (IBMSNAP_REGISTER) 表で同期点値の代わりに使用されます。

表 73 では、順序表の列の要旨を示します。

表 73. 順序表の列

| 列名 | 記述 |
|-----|---|
| SEQ | Informix 表のログまたはジャーナル ID (同期点) として使用される固有の番号。 |

schema.IBMSNAP_SIGNAL

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

この表の情報は、SQL を使って更新できます。

シグナル表には、キャプチャー・プログラムに特定のアクションを実行するように促すシグナルが保管されます。シグナルは、ユーザーまたはアプライ・プログラムから入力されます。

シグナル表は DATA CAPTURE CHANGES 属性を指定して作成されます。つまり、この表に対して実行されたすべての挿入、更新、および削除操作は、DB2 リカバリー・ログから読み取られたログ・レコードのように、キャプチャー・プログラムから見る事ができます。キャプチャー・プログラムは、シグナル表の更新および削除ログ・レコードはすべて無視しますが、シグナル挿入の、有効に作成およびコミットされたログ・レコードはすべて、注意が必要な「シグナル」として認識します。シグナル挿入によるログ・レコードに対してキャプチャー・プログラムが実行するアクションは、その挿入に関してシグナル表がどのように指定されているかによって異なります。シグナル表の中の値は、取るべきアクションをキャプチャー・プログラムに指示します。

この表の中で、完全を表す、SIGNAL_STATE 値が C のレコード、またはまたは保持制限ブルーニングの対象となるタイム・スタンプを持つレコードは、キャプチャー・プログラムによるブルーニングが行われると削除されます。

表 74 では、シグナル表の列の要旨を示します。

表 74. シグナル表の列

| 列名 | 記述 |
|-------------|--|
| SIGNAL_TIME | 行を一意的に識別するために使用されるタイム・スタンプ。キャプチャー・プログラムはこの固有値を使用して、キャプチャー・シグナルの処理が終了した時刻を示す、シグナル表の中の正しい行を検出します。タイム・スタンプ列は NOT NULL WITH DEFAULT として作成されるため、キャプチャー・シグナルは一般的に、DB2 で現行タイム・スタンプが SIGNAL_TIME 値として提供されるのと同じ方法で挿入できます。 |

表 74. シグナル表の列 (続き)

| 列名 | 記述 |
|----------------|--|
| SIGNAL_TYPE | 通知されたシグナルのタイプを示すフラグ。 |
| | <p>CMD ユーザー、アプライ・プログラム、またはよく知られたシステム・コマンドまたはシグナルである、別のアプリケーションから通知されたシグナル。使用可能なシグナルのサブタイプのリストについては、この表の SIGNAL_SUBTYPE 列を参照してください。</p> <p>USER ユーザーから通知されたシグナル。キャプチャー・プログラムは、SIGNAL_LSN 列の値を、シグナルが挿入されたときのログの LSN で更新し、SIGNAL_STATE 列の値を、P (保留) から R (受信) に更新します。</p> |
| SIGNAL_SUBTYPE | システム・コマンドからのシグナルが発生したときに (SIGNAL_TYPE = CMD)、キャプチャー・プログラムが実行するアクション。 |
| | <p>CAPSTART キャプチャー・プログラムは、SIGNAL_INPUT_IN 列の中のアプライ修飾子で識別される、特定のサブスクリプション・セット・メンバーの登録済みソースで変更のキャプチャーを開始します。たとえば、アプライ・プログラムはセット内のすべてのターゲット表に対してフル・リフレッシュを実行する前にこのシグナルを発行して、このセットで変更キャプチャー・レプリケーションを行う準備ができていることをキャプチャー・プログラムに知らせます。アプライ・プログラムがこのシグナルを通知します。</p> <p>STOP キャプチャー・プログラムは変更のキャプチャーを停止して終了します。このコマンドは、ユーザーからのみ発行できます。アプライ・プログラムからは発行できません。</p> <p>CAPSTOP キャプチャー・プログラムは、SIGNAL_INPUT_IN 列の中の <i>source_owner.source_table</i> で識別される特定の登録済みソースの変更のキャプチャーを停止します。このコマンドは、ユーザーからのみ発行できます。アプライ・プログラムからは発行できません。</p> <p>UPDANY アプライ・プログラム (SIGNAL_INPUT_IN 列の中のアプライ修飾子で識別される) は、update-anywhere 構成で、2 つのキャプチャー・プログラムを使用していることを、キャプチャー・プログラムに知らせます。アプライ・プログラムがこのシグナルを通知します。</p> <p>シグナル・タイプが USER の場合、シグナル・サブタイプは使用されないか、キャプチャー・プログラムから認識されないため、これは必要フィールドではありません。どのような値に設定してもかまいません。</p> |

表 74. シグナル表の列 (続き)

| 列名 | 記述 |
|-----------------|---|
| SIGNAL_INPUT_IN | <p>SIGNAL_TYPE = USER の場合、この列は、ユーザー定義の入力を保持します。 SIGNAL_TYPE = CMD の場合は、この値の意味は、このシグナルの SIGNAL_SUBTYPE によって異なります。</p> <p>CMD + CAPSTART マッピング ID。DB2 以外のリレーショナル・ソースは、キャプチャー・プログラムではなく、キャプチャー・トリガーによって処理されるため、シーケンス内の次の値でブルーニング・コントロール (IBMSNAP_PRUNCNTL) 表を更新するところの、挿入後 SIGNAL_TRIGGER が使用されます。</p> <p>CMD + UPDANY update-anywhere 構成でアプライ・プログラムを識別するためのアプライ修飾子。</p> <p>CMD + CAPSTOP キャプチャー・プログラムによる変更キャプチャーを停止する必要がある、ソース所有者およびソース表の名前。(source_owner.source_table)</p> |
| SIGNAL_STATE | <p>シグナルの状況を示すフラグ。</p> <p>P シグナルは保留。キャプチャー・プログラムはまだシグナルを受け取っていません。ユーザーがシグナルを通知するときには、SIGNAL_STATE を P に設定してください。</p> <p>R キャプチャー・プログラムはシグナルを受け取りました。キャプチャー・プログラムは、SIGNAL_TYPE = USER であるか、SIGNAL_TYPE = CMD および SIGNAL_SUBTYPE = STOP であるシグナルを受け取ると、SIGNAL_STATE を R に設定します (完全を示す C に変更するのではなく)。</p> <p>C キャプチャー・プログラムはシグナルの処理を完了しました。キャプチャー・プログラムは、SIGNAL_TYPE = CMD の場合、STOP を除くすべての SIGNAL_SUBTYPE について、この値を C に設定します。</p> |
| SIGNAL_LSN | <p>コミット・レコードのログ・シーケンス番号。この値は、キャプチャー・プログラムからのみ設定されます。</p> |

schema.IBMSNAP_UOW

サーバー: キャプチャー・コントロール・サーバー

デフォルト・スキーマ: ASN

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

作業単位 (UOW) 表は、ソース表にコミットされたトランザクションに関する追加情報を提供します。ユーザー・コピー以外のすべてのターゲット表タイプの場合、アプライ・プログラムはターゲット表に変更を適用するときに、IBMSNAP_COMMITSEQ 値を突き合わせることで、UOW 表と変更データ (CD) 表を結合します。キャプチャー・プログラムをコールド・スタートすると、この表のすべてのエントリは削除されます。

OS/400 の場合: Capture for iSeries は、レプリケーション・ソースのサブセットのデータのキャプチャーを開始することがあるため、部分的コールド・スタートを行った場合は、UOW 表の中のすべての行が削除されることはありません。

キャプチャー・プログラムでは、キャプチャー・スキーマごとに 1 つの UOW 表が必要になります。キャプチャー・プログラムは、レプリケーション・ソースでコミットされたログまたはジャーナル・レコードごとに、この表に新しい行を 1 つ挿入します。

OS/400 の場合: 一部のユーザー・プログラムはコミットメント・コントロールを使用しません。このような場合、Capture for iSeries は、CD 表に複数の行が書き込まれた後で、任意に新しい UOW 行を挿入します。このように見せかけのコミットメント境界を設けることにより、UOW 表のサイズを削減できます。

キャプチャー・プログラムは、アプライ・プログラムがブルーニング・セット (IBMSNAP_PRUNE_SET) 表に挿入した情報に基づいて、UOW 表の整理も行います。

OS/400 の場合: UOW 表は、ブルーニング・セット (IBMSNAP_PRUNE_SET) 表の情報によってではなく、保持制限に従って整理されます。

表 75 では、UOW 表の列の要旨を示します。

表 75. UOW 表の列

| 列名 | 記述 |
|-------------------|---|
| IBMSNAP_UOWID | この作業単位についての、ログ・レコード・ヘッダーからの作業単位 ID。この列を、非コンプリート CCD ターゲット表の一部とすることができます。 |
| IBMSNAP_COMMITSEQ | 取り込まれたコミット・ステートメントのログ・レコード・シーケンス番号。ユーザー・コピー以外のすべてのターゲット表タイプの場合、アプライ・プログラムはターゲット表に変更を適用するときに、この列の値に基づいて、UOW 表と CD 表を結合します。 |
| IBMSNAP_LOGMARKER | データがコミットされたときの時刻 (キャプチャー・コントロール・サーバー)。 |
| IBMSNAP_AUTHTKN | トランザクションに関連付けられた許可トークン。この ID は、データベースの監査に役立ちます。DB2 Universal Database for z/OS の場合、この列は相関 ID です。DB2 Universal Database for iSeries の場合、この列は、トランザクションを発生させたジョブのジョブ名です。この列は自動的に他の表にコピーされません。ユーザー・データ列として選択してコピーする必要があります。この列を、非コンプリート CCD ターゲット表の一部とすることができます。 |
| IBMSNAP_AUTHID | トランザクションに関連付けられた許可 ID。これはデータベースの監査に役立ちます。長さは 18 文字です。それを超えると切り捨てられます。DB2 Universal Database for z/OS の場合、この列は 1 次許可 ID です。DB2 Universal Database for iSeries の場合、この列は、トランザクションを発生させたアプリケーションを実行しているユーザー・プロファイル ID の名前になります。この列にはブランクが埋め込まれた 10 文字の ID が入ります。この列は自動的に他の表にコピーされません。ユーザー・データ列として選択してコピーする必要があります。この列を、非コンプリート CCD ターゲット表の一部とすることができます。 |

表 75. UOW 表の列 (続き)

| 列名 | 記述 |
|--------------------|---|
| IBMSNAP_REJ_CODE | <p>拒否またはロールバックされた行があるかどうかを示すフラグ。この値は、レプリケーション・ソースの定義時に競合検出が標準または拡張と指定された場合に <code>update-anywhere</code> レプリケーションでのみ使用されます。この列を、非コンプリート CCD ターゲット表の一部とすることができます。</p> <p>0 トランザクションで競合の発生は報告されていません。</p> <p>1 マスターとレプリカの同じ行が更新されたため、競合が発生しています。レプリカ側のトランザクションは拒否され、ロールバックされます。</p> <p>2 このトランザクションは、以前にリジェクトされたトランザクションに従属するため、拒否されてロールバックされました。前のトランザクションは、マスターとレプリカの中で同じ行が更新されたために拒否されたものであり、レプリカ側のトランザクションが、拒否されてロールバックされています。</p> <p>3 参照保全制約違反が少なくとも 1 つ含まれているため、このトランザクションは拒否され、ロールバックされました。このトランザクションはソース表で定義された参照制約に違反しているため、アプライ・プログラムはこのサブスクリプション・セットに失敗というマークを付けます。参照保全定義が訂正されるまで、更新はコピーできません。</p> <p>4 このトランザクションは、以前にリジェクトされたトランザクションに従属するため、拒否されてロールバックされました。前のトランザクションは、参照保全制約違反が少なくとも 1 つ含まれているために拒否されました。</p> |
| IBMSNAP_APPLY_QUAL | <p>変更を適用したアプライ・プログラムを識別するためのアプライ修飾子。この列を、非コンプリート CCD ターゲット表の一部とすることができます。</p> |

アプライ・コントロール・サーバー上の表、およびその列に関する記述

このセクションでは、アプライ・コントロール・サーバー上に保管される各表について、かなり詳しく説明します。また、各表の列のリストやその簡単な説明もあります。コントロール表はアルファベット順にリストされ、列は各表内に現れる順に左から右へリストされています。

ASN.IBMSNAP_APPENQ

サーバー: アプライ・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

アプライ・エンキュー表は、1 つのアプライ修飾子に対して 1 つのアプライ・プログラムだけが確実に実行されるようにするために使用されます。アプライ・プログラムは、アプライ・プログラムがシャットダウンするまで、この表の中の行を排他的にロックします。この表は OS/400 では使用されません。

518 ページの表 76 では、アプライ・エンキュー表の列の要旨を示します。

ASN.IBMSNAP_APPENQ

表 76. アプライ・エンキュー表の列

| 列名 | 記述 |
|------------|---|
| APPLY_QUAL | 同じアプライ・プログラムにより処理されるサブスクリプション・セットのグループを固有に指定します。この値は大文字小文字が区別されます。サブスクリプション・セットの定義時には、この値を指定しなければなりません。 |

ASN.IBMSNAP_APPLY_JOB (OS/400)

サーバー: アプライ・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

OS/400 特有のものであるアプライ・ジョブ表は、アプライ・コントロール・サーバー上で実行中のアプライ・プログラムのすべてのインスタンスの APPLY_QUAL の値が固有のものであることを保証するために使用されます。アプライ・プログラムの 1 つのインスタンスが開始されるたびに、この表に行が追加されます。アプライ・プログラムの新しいインスタンスを始動するとき、その APPLY_QUAL 値がすでに存在していると、始動コマンドは失敗します。

表 77 では、アプライ・ジョブ表の列の要旨を示します。

表 77. アプライ・ジョブ表の列

| 列名 | 記述 |
|----------------|---|
| APPLY_QUAL | サブスクリプション・セットのグループの固有 ID。この値は、サブスクリプション・セットの定義時にユーザーによって指定されます。アプライ・プログラムの各インスタンスは、APPLY_QUAL 値によって開始されます。この値は、update-anywhere レプリケーションで、アプライ・プログラムによる変更の循環レプリケーションを避けるために使用されます。 |
| CONTROL_SERVER | アプライ・コントロール表およびビューが定義されているデータベースの名前。 |
| JOB_NAME | このトレース・エントリーを書き込んだジョブの完全修飾名。 位置 1-10 APPLY_QUAL 位置 11-20 アプライ・プログラムを始動したユーザーの ID 位置 21-26 ジョブ番号 |
| USER_NAME | アプライ・プログラムの新しいインスタンスを始動したユーザーの名前。 |
| JOB_NUMBER | 特定のジャーナルに対する現行ジョブのジョブ番号。ジャーナルがアクティブでない場合は、この列には、最後に処理されたジョブのジョブ番号が入っています。 |

ASN.IBMSNAP_APPLYTRACE

サーバー: アプライ・コントロール・サーバー

アプライ・トレース表は、アプライ・プログラムからの重要なメッセージを保持します。アプライ・プログラムはこの表のプルーニングを自動的に行うことはありませんが、1つのサブスクリプション・セットの後で実行される SQL ステートメントを追加することにより、簡単にプルーニングを自動化できます。

表 78 では、アプライ・トレース表の列の要旨を示します。

表 78. アプライ・トレース表の列

| 列名 | 記述 |
|-------------|--|
| APPLY_QUAL | メッセージを挿入したアプライ・プログラムを一意的に識別します。 |
| TRACE_TIME | この表に行が挿入されたときのアプライ・コントロール・サーバーの時刻。 |
| OPERATION | アプライ・プログラムの操作のタイプ、たとえば、初期化、アプライ、またはエラー条件。 |
| DESCRIPTION | メッセージ ID とメッセージ・テキスト。メッセージ ID は DESCRIPTION 列の最初の 7 文字です。メッセージ・テキストは、DESCRIPTION 列の位置 9 から始まります。 |

ASN.IBMSNAP_APPLYTRAIL

サーバー: アプライ・コントロール・サーバー

アプライ・トレール表には、アプライ・プログラムによって実行される、すべてのサブスクリプション・セット・サイクルの監査証跡情報が含まれます。アプライ・トレール表には、サブスクリプションに対して実行された更新の履歴が入ります。この表は、診断およびパフォーマンス統計のリポジトリです。アプライ・プログラムで問題が生じたときにアプライ・トレール表を参照するのは非常に効果的です。アプライ・プログラムはこの表のプルーニングを自動的に行うことはありませんが、1つのサブスクリプション・セットの後で実行される SQL ステートメントを追加することにより、簡単にプルーニングを自動化できます。

表 79 では、アプライ・トレール表の列の要旨を示します。

表 79. アプライ・トレール表の列

| 列名 | 記述 |
|------------|--|
| APPLY_QUAL | サブスクリプション・セットを処理したアプライ・プログラムを一意的に識別します。 |
| SET_NAME | アプライ・プログラムが処理したサブスクリプション・セットの名前。 |
| SET_TYPE | 最後のアプライ・サイクルの後でサブスクリプション・セット (IBMSNAP_SUBS_SET) 表の SET_TYPE 列に表示された値。各値の意味については、528 ページの『ASN.IBMSNAP_SUBS_SET』を参照してください。 |

ASN.IBMSNAP_APPLYTRAIL

表 79. アプライ・トレール表の列 (続き)

| 列名 | 記述 |
|-------------------|--|
| WHOS_ON_FIRST | <p>update-anywhere レプリケーション・シナリオでは、処理順序をコントロールするために以下の値を使用します。</p> <p>F (first の略) ソース表がレプリカであり、ターゲット表がマスターです。レプリカ表とマスター表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。F は読み取り専用のサブスクリプションでは使用されません。update-anywhere で使用されるものです。</p> <p>S (second の略) ソース表はマスター表またはその他のソースであり、ターゲット表はレプリカまたはその他のコピーです。マスター表とレプリカ表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。S は、すべての読み取り専用サブスクリプションについて使用されます。</p> |
| ASNLOAD | <p>アプライ・プログラムを始動するために使用された値。</p> <p>Y パラメーター loadxit=y を指定してアプライ・プログラムを始動したため、サブスクリプション・セットのフル・リフレッシュを実行するために ASNLOAD ユーザー出口ルーチンが呼び出されることを意味します。</p> <p>N フル・リフレッシュが必要ないか、アプライ・プログラムの始動時に loadxit パラメーターが指定されていないため、ASNLOAD 出口ルーチンが呼び出されないことを意味します。</p> <p>NULL ASNLOAD 出口ルーチンを呼び出すかどうかをアプライ・プログラムが判断する前に、アプライ・プログラムエラーが生じたことを示します。</p> |
| FULL_REFRESH | <p>フル・リフレッシュが発生したかどうかを示すフラグ。</p> <p>Y サブスクリプション・セットに対してフル・リフレッシュが実行されたことを示します。</p> <p>N サブスクリプション・セットに対してフル・リフレッシュが実行されなかったことを示します。</p> <p>NULL フル・リフレッシュが必要かどうかをアプライ・プログラムが判断する前に、エラーが生じたことを示します。</p> |
| EFFECTIVE_MEMBERS | <p>フル・リフレッシュか、挿入、リフレッシュ、および削除のレプリケーションのどちらかにより、1 回の Apply サイクルで変更されたサブスクリプション・セットの数。値の範囲は、0 以上、定義済みのサブスクリプション・セット・メンバーの数以下です。</p> |
| SET_INSERTED | <p>サブスクリプション・サイクルにおいてサブスクリプション・セット・メンバーに挿入された行の合計数。</p> |
| SET_DELETED | <p>サブスクリプション・サイクルにおいてサブスクリプション・セット・メンバーから削除された行の合計数。</p> |
| SET_UPDATED | <p>サブスクリプション・サイクルにおいてサブスクリプション・セット・メンバーで更新された行の合計数。</p> |
| SET_REWORKED | <p>最後のサイクルでアプライ・プログラムが再処理した合計行数。アプライ・プログラムは、以下の条件下で変更を再実行します。</p> <ul style="list-style-type: none"> 行がターゲット表にすでに存在しているため挿入が失敗した場合、アプライ・プログラムは、挿入操作を既存行の更新操作に変換します。 行がターゲット表に存在していないため更新が失敗した場合、アプライ・プログラムは、更新操作を挿入操作に変換します。 |

表 79. アプライ・トレール表の列 (続き)

| 列名 | 記述 |
|-------------------|---|
| SET_REJECTED_TRXS | update-anywhere 競合のためにリジェクトされたトランザクションの合計数。この列は、競合検出が「標準」または「詳細」と定義されている update-anywhere サブスクリプション・セットに対してのみ使用されます。 |
| STATUS | <p>特定のサイクル後のアプライ・プログラムの作業状況を表す値。</p> <p>-1 レプリケーションは失敗しました。アプライ・プログラムは適用済みの行のセット全体をバックアウトし、データはコミットされません。始動パラメーターが SQLERRCONTINUE = Y の場合、最後のサイクル中にアプライ・プログラムに戻される SQLSTATE は、SQLERRCONTINUE (apply_qualifier.SQS) の入力ファイルでユーザーが指定した許容エラーの 1 つではありません。</p> <p>0 アプライ・プログラムはサブスクリプション・セットを正常に処理しました。始動パラメーターが SQLERRCONTINUE = Y の場合、アプライ・プログラムは、ユーザーから SQLERRCONTINUE 始動パラメーター (apply_qualifier.SQS で) で指定された SQL エラーを検出してせず、行を拒否していません。</p> <p>2 アプライ・プログラムはサブスクリプション・セットを複数のサイクルで処理しています。アプライ・プログラムは、MAX_SYNCH_MINUTES コントロール列に従って分割された 1 つの論理サブスクリプションを正常に処理しました。</p> <p>16 アプライ・プログラムはサブスクリプション・セットを正常に処理し、0 という状況に戻しました。しかしアプライ・プログラムは、ユーザーが SQLERRCONTINUE 始動パラメーター (apply_qualifier.SQS で) で指定したいくつかの SQL エラーを検出したため、いくつかの行を拒否しました。失敗した行の詳細は、apply_qualifier.ERR ファイルで確認してください。</p> <p>例: ユーザーは SQLERRCONTINUE = Y と設定し、SQL の許容される状態を 23502 (SQL コード -407) と指定します。23502 エラーが発生しますが、他のエラーは発生していません。アプライ・プログラムはサブスクリプション・セットの処理を終了し、状態を 16 に設定します。次の実行時に、23502 エラーが発生した後、07006 (SQL コード -301) が発生します。アプライ・プログラムは今回は、サブスクリプション・セットの処理を停止し、適用済みの行のセット全体をバックアウトし、状況を -1 に設定します (データはコミットされていないため)。</p> <p>18 アプライ・プログラムは複数のサイクルでサブスクリプション・セットを処理し、2 という状況に戻しています。これは、MAX_SYNCH_MINUTES コントロール列に従って分割された 1 つの論理サブスクリプションが正常に処理されたことを意味します。しかし、ユーザーが SQLERRCONTINUE 始動パラメーター (apply_qualifier.SQS で) で指定した SQL エラーのいくつかを検出されたため、いくつかの行が拒否されています。失敗した行の詳細は、apply_qualifier.ERR ファイルで確認してください。</p> |
| LASTRUN | 最後のサブスクリプションが開始された概算の時刻。アプライ・プログラムは、サブスクリプション・セットを処理するたびに LASTRUN 値を設定します。これは、アプライ・プログラムがサブスクリプション・セットの処理を開始する、アプライ・コントロール・サーバーにおけるおおよその時刻です。 |
| LASTSUCCESS | サブスクリプション・セットが最後に正常に処理されたときの、処理開始時点のアプライ・コントロール・サーバーのタイム・スタンプ。 |
| SYNCHPOINT | アプライ・プログラムはこの列を使用して進行状況を記録します。サブスクリプション・セットのデータの処理が、この同期点の値まで終了していることを示します。 |

表 79. アプライ・トレール表の列 (続き)

| 列名 | 記述 |
|--------------------|---|
| SYNCHTIME | アプライ・プログラムはこの列を使用して進行状況を記録します。サブスクリプション・セットのデータの処理が、このタイム・スタンプまで終了していることを示します。 |
| SOURCE_SERVER | ソース表およびビューが定義されている DB2 Universal Database データベース名。 |
| SOURCE_ALIAS | SOURCE_SERVER 列で指定されているソース・サーバーの DB2 Universal Database別名。 |
| SOURCE_OWNER | アプライ・プログラムが処理中であったソース表またはビューの高位修飾子。この値は、アプライ・サイクルが失敗したときにのみ設定されます。 |
| SOURCE_TABLE | アプライ・プログラムが処理中であったソース表またはビューの名前。この値は、アプライ・サイクルが失敗したときにのみ設定されます。 |
| SOURCE_VIEW_QUAL | アプライ・プログラムが処理中であったソース表またはビューのソース・ビュー修飾子の値。この値は、アプライ・サイクルが失敗したときにのみ設定されます。 |
| TARGET_SERVER | ターゲットの表またはビューが保管されているサーバーのデータベース名。 |
| TARGET_ALIAS | TARGET_SERVER 列で指定されているターゲット・サーバーの DB2 Universal Database 別名。 |
| TARGET_OWNER | アプライ・プログラムが処理中であったターゲット表の高位修飾子。この値は、アプライ・サイクルが失敗したときにのみ設定されます。 |
| TARGET_TABLE | アプライ・プログラムが処理中であったターゲット表の名前。この値は、アプライ・サイクルが失敗したときにのみ設定されます。 |
| CAPTURE_SCHEMA | このサブスクリプション・セットのキャプチャー・サーバー表のスキーマ名。 |
| TGT_CAPTURE_SCHEMA | ターゲット表が別のサブスクリプション・セットのソースでもある場合 (multi-tier 構成中の外部の CCD 表、または update-anywhere 構成中のレプリカ表など) は、この列には、表がソースとして機能するときに使用されるキャプチャー・スキーマが含まれます。 |
| FEDERATED_SRC_SRVR | DB2 以外のリレーショナル・ソースの場合にのみアプライされる、サブスクリプション・セットのソースである、連合リモート・サーバーの名前。 |
| FEDERATED_TGT_SRVR | DB2 以外のリレーショナル・ターゲット・サーバーの場合にのみアプライされる、サブスクリプション・セットのターゲットである、連合リモート・サーバーの名前。 |
| JRN_LIB | OS/400 キャプチャー・サーバーにのみアプライされるこの列は、ソース表が使用するジャーナルのライブラリー名です。 |
| JRN_NAME | OS/400 キャプチャー・サーバーにのみアプライされるこの列は、ソース表が使用するジャーナルの名前です。この列で、アスタリスクの後ろに 9 つのブランクが続くときには、ソース表が現在ジャーナルの中にあることを意味します。この場合は、このソース表のデータをキャプチャーすることはできません。 |
| COMMIT_COUNT | サブスクリプション・セット (IBMSNAP_SUBS_SET) 表の中に記録される、最後のアプライ・サイクルからの COMMIT_COUNT の値。 |
| OPTION_FLAGS | DB2 レプリケーションの将来のオプション用に予約済み。この列には現在、デフォルト値の NNNN が入っています。 |
| EVENT_NAME | セットの処理を起動したイベントを表すために使用される固有の文字ストリング。 |
| ENDTIME | アプライ・プログラムがサブスクリプション・セットの処理を終了したときの、アプライ・コントロール・サーバーにおけるタイム・スタンプ。セットの処理に要した時間を知るには、LASTRUN を ENDTIME から減算します。 |
| SOURCE_CONN_TIME | アプライ・プログラムが初めてフェッチ・ソース・データに接続したときの、キャプチャー・コントロール・サーバーにおけるタイム・スタンプ。 |
| SQLSTATE | 失敗した実行の SQL 状態コード。それ以外の場合は、NULL になります。 |
| SQLCODE | 失敗した実行の SQL エラー・コード。それ以外の場合は、NULL になります。 |
| SQLERRP | 実行の失敗原因となった SQL エラーが生じたサーバーのデータベース製品 ID。それ以外の場合は、NULL になります。 |

表 79. アプライ・トレール表の列 (続き)

| 列名 | 記述 |
|---------|--|
| SQLERRM | 失敗した実行の SQL エラー情報。 |
| APPERRM | アプライ・プログラムの実行に失敗したときのエラー・メッセージ ID およびテキスト。 |

ASN.IBMSNAP_SUBS_COLS

サーバー: アプライ・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

サブスクリプション列表には、サブスクリプション・セット内にコピーされるサブスクリプション・セット・メンバーの列に関する情報が含まれます。一对のソース表とターゲット表で、1 つまたは複数の列の情報が変更されると、この表で自動的に行が挿入または削除されます。

この表は、サブスクリプション・セット・メンバーの特定の列に関する情報が必要な場合に使用してください。

表 80 では、サブスクリプション列表の列の要旨を示します。

表 80. サブスクリプション列表の列

| 列名 | 記述 |
|---------------|---|
| APPLY_QUAL | このサブスクリプション・セット・メンバーを処理するアプライ・プログラムを一意的に識別します。 |
| SET_NAME | このメンバーが所属するサブスクリプション・セットの名前。 |
| WHOS_ON_FIRST | update-anywhere レプリケーション・シナリオでは、処理順序をコントロールするために以下の値を使用します。 F (first の略) ソース表がレプリカであり、ターゲット表がマスターです。レプリカ表とマスター表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。F は読み取り専用のサブスクリプションでは使用されません。update-anywhere で使用されるものです。 S (second の略) ソース表はマスター表またはその他のソースであり、ターゲット表はレプリカまたはその他のコピーです。マスター表とレプリカ表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。S は、すべての読み取り専用サブスクリプションについて使用されます。 |
| TARGET_OWNER | ターゲット表またはビューの高位修飾子。 |
| TARGET_TABLE | データがアプライされる表またはビュー。 |

ASN.IBMSNAP_SUBS_COLS

表 80. サブスクリプション列表の列 (続き)

| 列名 | 記述 |
|-------------|--|
| COL_TYPE | 列のタイプを示すフラグ。 A 変更後イメージ列。 B 変更前イメージ列。 C スカラー関数を使用する算出列または SQL 式。 D DATALINK 列。 F 列関数を使用する算出列。 L LOB 標識値。 P 変更前イメージ述部列。 R システムから提供され、主キー列として使用される相対レコード番号列。DB2 DataPropagator for iSeries でのみ使用されます。 |
| TARGET_NAME | ターゲット表またはビューの列の名前。ソース列名と一致する必要はありません。 内部の CCD 列名は変更できません。それらの名前は表の列名と一致している必要があります。 |
| IS_KEY | 列がターゲット・キーの一部であるかどうかを示すフラグ。ターゲット・キーは、コンデンス・ターゲット表の固有索引、または主キーのいずれかです。 Y 列はターゲット・キー全体であるか、ターゲット・キーの一部です。 N 列は、ターゲット・キーの一部ではありません。 |
| COLNO | オリジナル・ソースにおける列の数値ロケーション。これは、表示およびサブスクリプションで他のユーザー列に対して相対的に保たれます。 |
| EXPRESSION | ターゲット列の内容を作成するために使用された SQL 式、またはソース列名。 |

ASN.IBMSNAP_SUBS_EVENT

サーバー: アプライ・コントロール・サーバー

この表の情報は、SQL を使って更新できます。

サブスクリプション・イベント表には、サブスクリプション・セットに関連付けられたイベント・トリガーに関する情報が含まれます。また、イベント名と関連付けられた名前とタイム・スタンプも含まれます。アプライ・プログラムを始動するために新規イベントを作成するときに、この表に行を挿入してください。81 ページの『イベントに基づくスケジューリング』を参照してください。

表 81 では、サブスクリプション・イベント表の列の要旨を示します。

表 81. サブスクリプション・イベント表の列

| 列名 | 記述 |
|------------|--|
| EVENT_NAME | イベントの固有 ID。この ID は、サブスクリプション・セットのレプリケーションを起動するために使用されます。 |

表 81. サブスクリプション・イベント表の列 (続き)

| 列名 | 記述 |
|----------------|---|
| EVENT_TIME | 現在または将来の通知時刻の、アプライ・コントロール・サーバーのタイム・スタンプ。レプリケーション・イベントをシグナルするユーザー・アプリケーションがこの列に値を提供します。 |
| END_SYNCHPOINT | この時点までに取り込まれたデータのみをアプライするようにアプライ・プログラムに指示するログ・シーケンス番号。シグナル表を参照し、タイム・スタンプに関連付けられた正確なログ・シーケンス番号を検索すれば、使用する正確な END_SYNCHPOINT を知ることができません。ログのこの時点以降にコミットされたトランザクションは、その後のイベントが通知されるまでは複製されません。ユーザーが END_SYNCHPOINT および END_OF_PERIOD の値を指定した場合、アプライ・プログラムは END_SYNCHPOINT 値を使用するので、複製する最大ログ・シーケンス番号を知るためにコントロール表から計算を実行する必要はなくなりません。 |
| END_OF_PERIOD | この時点までにログに記録されたデータのみをアプライするためにアプライ・プログラムによって使用されるタイム・スタンプ。ログのこの時点以降にコミットされたトランザクションは、その後のイベントが通知されるまでは複製されません。 |

ASN.IBMSNAP_SUBS_MEMBR

サーバー: アプライ・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

サブスクリプション・メンバー表には、サブスクリプション・セットに対して定義された、ソース表とターゲット表の個々のペアに関する情報が含まれています。ユーザーがサブスクリプション・セット・メンバーを追加すると、この表に 1 つの行が自動的に挿入されます。

この表は、サブスクリプション・セット内で特定のソース表とターゲット表の対を指定するために使用します。

表 82 では、サブスクリプション・メンバー表の列の要旨を示します。

表 82. サブスクリプション・メンバー表の列

| 列名 | 記述 |
|------------|--|
| APPLY_QUAL | このサブスクリプション・セット・メンバーを処理するアプライ・プログラムを一意的に識別します。 |
| SET_NAME | このメンバーが所属するサブスクリプション・セットの名前。 |

ASN.IBMSNAP_SUBS_MEMBR

表 82. サブスクリプション・メンバー表の列 (続き)

| 列名 | 記述 |
|------------------|--|
| WHOS_ON_FIRST | <p>update-anywhere レプリケーション・シナリオでは、処理順序をコントロールするために以下の値を使用します。</p> <p>F (first の略) ソース表がレプリカであり、ターゲット表がマスターです。レプリカ表とマスター表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。F は読み取り専用のサブスクリプションでは使用されません。update-anywhere で使用されるものです。</p> <p>S (second の略) ソース表はマスター表またはその他のソースであり、ターゲット表はレプリカまたはその他のコピーです。マスター表とレプリカ表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。S は、すべての読み取り専用サブスクリプションについて使用されます。</p> |
| SOURCE_OWNER | このメンバーのソース表またはビューの高位修飾子。 |
| SOURCE_TABLE | このメンバーのソース表またはビューの名前。 |
| SOURCE_VIEW_QUAL | 登録表内の類似した例を突き合わせることによって、物理表のビューをサポートします。この値は、ソースとして定義されている物理表の場合は 0 に、ソースとして定義されているビューの場合は 0 より大きい値に設定されます。同じ SOURCE_OWNER および SOURCE_TABLE 列値を持つ別々のソース・ビューについて複数のサブスクリプションをサポートするためにこの列を使用します。 |
| TARGET_OWNER | このメンバーのターゲット表またはビューの高位修飾子。 |
| TARGET_TABLE | このメンバーのターゲット表またはビューの名前。 |
| TARGET_CONDENSED | <p>以下のことを示すフラグ。</p> <p>Y 特定の主キー値に関して、ターゲット表に表示される行は 1 つだけです。</p> <p>N 完全な更新履歴を保持して、すべての変更を残す必要があります。</p> <p>A ターゲット表は、基礎集約表または変更集約表です。</p> |
| TARGET_COMPLETE | <p>以下のことを示すフラグ。</p> <p>Y ターゲット表は、関係するそれぞれの主キー値につき 1 つの行を保持します。</p> <p>N ターゲット表は、主キー値の行のサブセットを保持します。</p> |
| TARGET_STRUCTURE | <p>ターゲット表の構造。</p> <p>1 ユーザー表</p> <p>3 CCD 表</p> <p>4 ポイント・イン・タイム表</p> <p>5 基礎集約表</p> <p>6 変更集約表</p> <p>7 レプリカ</p> <p>8 ユーザー・コピー</p> |

表 82. サブスクリプション・メンバー表の列 (続き)

| 列名 | 記述 |
|-------------------|--|
| PREDICATES | <p>TARGET_TABLE 列内の表用の WHERE 文節に入れられる述部をリストします。この WHERE 文節は、ソース表の行サブセットを作成します。述部が認識されるのは、WHOS_ON_FIRST が S に設定されている場合だけです。アプライ・プログラムは ORDER BY 文節を生成できないため、述部に ORDER BY 文節を含めることはできません。集約表は、ダミー述部とその後に続く GROUP BY 文節を必要としています。</p> <p>アプライ・プログラムはフル・リフレッシュおよび変更キャプチャー・レプリケーションの両方でこれらの述部を使用するため、この列に、CD 表または UOW 表の列に関係する述部を含めることはできません。CD 表または UOW 表の参照を含む述部は、UOW_CD_PREDICATES 列に保管されます。</p> |
| MEMBER_STATE | <p>メンバーの状態を示すフラグ。</p> <p>N (New の略) メンバーはこのサブスクリプション・セットの新規メンバーです。</p> <p>L (Loaded の略) このサブスクリプション・セットのメンバーがロードされましたが、変更キャプチャー・サイクルはまだ発生していません。</p> <p>S (Synchronized の略) このサブスクリプション・セットのメンバーは、整合した同期点値に同期化されています。</p> |
| TARGET_KEY_CHG | <p>ユーザーがソース表で、ターゲット表のターゲット・キー列のソース列を変更したときに、アプライ・プログラムが更新をどのように処理するかを示すフラグ。</p> <p>Y アプライ・プログラムは、ターゲット・キー列の変更前イメージに基づいてターゲット表を更新します。つまり、アプライ・プログラムは述部を新しい値ではなく、古い値に変更します。ターゲット・キーの変更前イメージが CD 表の中に存在するように、変更前イメージの各列を登録してください。登録表の中の対応する登録エントリーについて、CHG_UPD_TO_DEL_INS 列の値が N に設定されていることを確認してください。</p> <p>N アプライ・プログラムは、ターゲット・キーを構成する列が更新されていないという仮定に基づくロジックを使用して、更新および削除を処理します。</p> |
| JOIN_UOW_CD | <p>アプライ・プログラムがユーザー・コピー・ターゲット表を処理するときに、CD 表および UOW 表の結合を行うかどうかを示すフラグ。このフラグは、CD 表の中にない、UOW 表の列を使用する述部を持つサブスクリプション・セット・メンバーをユーザーが定義したときに必要になります。ターゲット表のタイプがユーザー・コピー以外のものである場合には、アプライ・プログラムはメンバーの処理時に CD 表および UOW 表の結合を使用するため、メンバーの処理時にはこの列は無視されます。</p> <p>Y アプライ・プログラムはメンバーの処理時に CD 表および UOW 表の結合を使用します。</p> <p>N アプライ・プログラムはメンバーの処理時に CD 表および UOW 表の結合を使用しません。CD 表からのみ変更が読み取られます。</p> <p>NULL アプライ・プログラムはメンバーの処理時にこの列を無視します。ターゲット表がユーザー・コピーであり、この列の値が NULL である場合には、アプライ・プログラムはメンバーの処理時に CD 表および UOW 表を結合しません。</p> |
| UOW_CD_PREDICATES | <p>アプライ・プログラムが変更キャプチャー・レプリケーションにのみ必要とし、フル・リフレッシュには必要としない、CD 表または UOW 表からの列を含む述部が入っています。アプライ・プログラムは変更キャプチャー・レプリケーション時には、この列の述部と、PREDICATES 列の述部を処理します。アプライ・プログラムはフル・リフレッシュ時には、PREDICATES 列の述部のみを処理します。</p> |

ASN.IBMSNAP_SUBS_MEMBR

表 82. サブスクリプション・メンバー表の列 (続き)

| 列名 | 記述 |
|------------------|---|
| LOADX_TYPE | <p>このメンバーのロードのタイプ。この列の値は、デフォルトのオーバーライドに使用されません。</p> <p>NULL</p> <p>z/OS の場合: このメンバーにはクロスローダー・ユーティリティが使用されません。</p> <p>UNIX および Windows の場合: ASNLOAD 出口が、このメンバーに最適なユーティリティを決定します (オプション 3、4、または 5)。</p> <p>1 このメンバーには ASNLOAD は使用されません。これにより、ユーザーが始動時に LOADX を指定した場合にでも、特定のサブスクリプション・セット・メンバーについて ASNLOAD オプションがオフになります。</p> <p>2 ユーザー定義の、またはユーザーが変更した ASNLOAD 終了コードが使用されます。</p> <p>3 このメンバーにはクロスローダーが使用されます。</p> <p>4 UNIX および Windows の場合のみ: このメンバーには EXPORT/LOAD が使用されます。</p> <p>5 UNIX および Windows の場合のみ: このメンバーには EXPORT/INPORT が使用されます。</p> |
| LOAD_SRC_N_OWNER | <p>ユーザーが作成したニックネームの所有者。次の場合はこの値が必須です。</p> <ul style="list-style-type: none">このメンバーに対してクロスローダーが使用される (LOADX_TYPE は 3)。ターゲット・サーバーは UNIX または Windows である。ソースにまだニックネームが反映されていない。 |
| LOAD_SRC_N_TABLE | <p>ユーザーが作成したニックネーム表。次の場合はこの値が必須です。</p> <ul style="list-style-type: none">このメンバーに対してクロスローダーが使用される (LOADX_TYPE は 3)。ターゲット・サーバーは UNIX または Windows である。ソースにまだニックネームが反映されていない。 |

ASN.IBMSNAP_SUBS_SET

サーバー: アプライ・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。

サブスクリプション・セット表には、アプライ・コントロール・サーバーで定義されたすべてのサブスクリプション・セットが記載され、これらのセットのレプリケーションの進行状況が文書化されます。行は、サブスクリプション・セット定義を作成したときに自動的にこの表に挿入されます。

529 ページの表 83 では、サブスクリプション・セット表の列の要旨を示します。

表 83. サブスクリプション・セット表の列

| 列名 | 記述 |
|---------------|--|
| APPLY_QUAL | このサブスクリプション・セットを処理するアプライ・プログラムを一意的に識別します。 |
| SET_NAME | サブスクリプション・セットの名前。 |
| SET_TYPE | セットが読み取り専用か読み取り / 書き込みかを示すフラグ。 R セットは読み取り専用です。 U セットは update-anywhere 構成であるため、読み取り / 書き込みです。 |
| WHOS_ON_FIRST | update-anywhere レプリケーション・シナリオでは、処理順序をコントロールするために以下の値を使用します。 F (first の略) ソース表がレプリカであり、ターゲット表がマスターです。レプリカ表とマスター表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。F は読み取り専用のサブスクリプションでは使用されません。 update-anywhere で使用されるものです。 S (second の略) ソース表はマスター表またはその他のソースであり、ターゲット表はレプリカまたはその他のコピーです。マスター表とレプリカ表の間で更新の矛盾が生じた場合、レプリカの側の矛盾するトランザクションは拒否されます。S は、すべての読み取り専用サブスクリプションについて使用されます。 |
| ACTIVATE | アプライ・プログラムが次のサイクルでこのセットを処理するかどうかを示すフラグ。 0 サブスクリプション・セットは非活動化されています。アプライ・プログラムはこのセットを処理しません。 1 サブスクリプション・セットは無期限にアクティブです。アプライ・プログラムは、ユーザーがセットを非活動化するか、アプライ・プログラムが処理を行えなくなるまで、各アプライ・サイクルでセットを処理します。 2 サブスクリプション・セットは、1 つのアプライ・サイクルでのみアクティブです。アプライ・プログラムはセットを一度処理した後、セットを非活動化します。 |
| SOURCE_SERVER | ソース表およびビューが定義されている、キャプチャー・コントロール・サーバーのデータベース名。 |
| SOURCE_ALIAS | SOURCE_SERVER 列で指定されているキャプチャー・コントロール・サーバーに対応する DB2 Universal Database 別名。 |
| TARGET_SERVER | ターゲットの表またはビューが保管されているサーバーのデータベース名。 |
| TARGET_ALIAS | TARGET_SERVER 列で指定されているターゲット・サーバーの DB2 Universal Database 別名。 |

表 83. サブスクリプション・セット表の列 (続き)

| 列名 | 記述 |
|---------|---|
| STATUS | <p>特定のサイクル後のアプライ・プログラムの作業状況を表す値。</p> <p>-1 レプリケーションは失敗しました。アプライ・プログラムは適用済みの行のセット全体をバックアウトし、データはコミットされません。始動パラメーターが <code>SQLERRCONTINUE = Y</code> の場合、最後のサイクル中にアプライ・プログラムに戻される <code>SQLSTATE</code> は、<code>SQLERRCONTINUE (apply_qualifier.SQS)</code> の入力ファイルでユーザーが指定した許容エラーの 1 つではありません。</p> <p>0 アプライ・プログラムはサブスクリプション・セットを正常に処理しました。始動パラメーターが <code>SQLERRCONTINUE = Y</code> の場合、アプライ・プログラムは、ユーザーから <code>SQLERRCONTINUE</code> 始動パラメーター (<code>apply_qualifier.SQS</code> で) で指定された <code>SQL</code> エラーを検出していず、行を拒否していません。</p> <p>2 アプライ・プログラムはサブスクリプション・セットを複数のサイクルで処理しています。アプライ・プログラムは、<code>MAX_SYNC_MINUTES</code> コントロール列に従って分割された 1 つの論理サブスクリプションを正常に処理しました。</p> <p>16 アプライ・プログラムはサブスクリプション・セットを正常に処理し、0 という状況に戻しました。しかしアプライ・プログラムは、ユーザーが <code>SQLERRCONTINUE</code> 始動パラメーター (<code>apply_qualifier.SQS</code> で) で指定したいくつかの <code>SQL</code> エラーを検出したため、いくつかの行を拒否しました。失敗した行の詳細は、<code>apply_qualifier.ERR</code> ファイルで確認してください。</p> <p>例: ユーザーは <code>SQLERRCONTINUE = Y</code> と設定し、<code>SQL</code> の許容される状態を 23502 (<code>SQL</code> コード -407) と指定します。23502 エラーが発生しますが、他のエラーは発生していません。アプライ・プログラムはサブスクリプション・セットの処理を終了し、状態を 16 に設定します。次の実行時に、23502 エラーが発生した後、07006 (<code>SQL</code> コード -301) が発生します。アプライ・プログラムは今回は、サブスクリプション・セットの処理を停止し、適用済みの行のセット全体をバックアウトし、状況を -1 に設定します (データはコミットされていないため)。</p> <p>18 アプライ・プログラムは複数のサイクルでサブスクリプション・セットを処理し、2 という状況に戻しています。これは、<code>MAX_SYNC_MINUTES</code> コントロール列に従って分割された 1 つの論理サブスクリプションが正常に処理されたことを意味します。しかし、ユーザーが <code>SQLERRCONTINUE</code> 始動パラメーター (<code>apply_qualifier.SQS</code> で) で指定した <code>SQL</code> エラーのいくつかを検出されたため、いくつかの行が拒否されています。失敗した行の詳細は、<code>apply_qualifier.ERR</code> ファイルで確認してください。</p> |
| LASTRUN | <p>サブスクリプション・セットが最後に開始された概算の時刻。アプライ・プログラムは、サブスクリプション・セットを処理するたびに <code>LASTRUN</code> 値を設定します。これは、アプライ・プログラムがサブスクリプション・セットの処理を開始する、アプライ・コントロール・サーバーにおけるおおよその時刻です。</p> |

表 83. サブスクリプション・セット表の列 (続き)

| 列名 | 記述 |
|--------------------|--|
| REFRESH_TYPE | <p>アプライ・プログラムにこのサブスクリプション・セットの処理を促すために使用されるスケジューリングのタイプ。</p> <p>R アプライ・プログラムは時間ベースのスケジューリングを使用します。アプライ・プログラムは、 SLEEP_MINUTES の中の値を使用して、サブスクリプション・セットの処理を開始する時間を判断します。</p> <p>E アプライ・プログラムはイベント・ベースのスケジューリングを使用します。アプライ・プログラムはサブスクリプション・イベント (IBMSNAP_SUBS_EVENT) 表の中の時刻値を確認して、サブスクリプション・セットの処理を開始する時間を判断します。イベントが生じないと、レプリケーション (変更のキャプチャーまたはフル・リフレッシュ) を何も開始できません。</p> <p>B アプライ・プログラムは時間ベースのスケジューリングとイベント・ベースのスケジューリングの両方を使用します。このため、アプライ・プログラムは時間またはイベントの基準に基づいてサブスクリプション・セットを処理します。</p> |
| SLEEP_MINUTES | サブスクリプション・セット処理の間の非活動時間 (分単位) を指定します。処理時間は、REFRESH_TYPE が R または B の場合にのみ使用されます。 |
| EVENT_NAME | イベントの名前を表す固有の文字ストリング。この ID は、サブスクリプション・セットに対してレプリケーションを起動したい場合にサブスクリプション・イベント表を更新するのに使用します。イベント名は、 REFRESH_TYPE が E または B の場合にのみ使用されます。 |
| LASTSUCCESS | サブスクリプション・セットが最後に正常に処理されたときの、処理開始時点のアプライ・コントロール・サーバーのタイム・スタンプ。 |
| SYNCHPOINT | アプライ・プログラムはこの列を使用して進行状況を記録します。サブスクリプション・セットのデータの処理が、この同期点の値まで終了していることを示します。 |
| SYNCHTIME | アプライ・プログラムはこの列を使用して進行状況を記録します。サブスクリプション・セットのデータの処理が、このタイム・スタンプまで終了していることを示します。 |
| CAPTURE_SCHEMA | このサブスクリプション・セットのソースを処理するキャプチャー・コントロール表のスキーマ名。 |
| TGT_CAPTURE_SCHEMA | ターゲット表が別のサブスクリプション・セットのソースでもある場合 (multi-tier 構成中の外部の CCD 表、または update-anywhere 構成中のレプリカ表など) は、この列には、表がソースとして機能するときに使用されるキャプチャー・スキーマが含まれます。 |
| FEDERATED_SRC_SRVR | DB2 以外のリレーショナル・ソースの場合にのみアプライされる、サブスクリプション・セットのソースである、連合リモート・サーバーの名前。 |
| FEDERATED_TGT_SRVR | DB2 以外のリレーショナル・ターゲットの場合にのみアプライされる、サブスクリプション・セットのターゲットである、連合リモート・サーバーの名前。 |
| JRN_LIB | OS/400 キャプチャー・サーバーにのみアプライされるこの列は、ソース表が使用するジャーナルのライブラリ名です。 |
| JRN_NAME | OS/400 キャプチャー・サーバーにのみアプライされるこの列は、ソース表が使用するジャーナルの名前です。この列で、アスタリスクの後ろに 9 つのブランクが続くときには、ソース表が現在ジャーナルの中になことを意味します。この場合は、このソース表のデータをキャプチャーすることはできません。 |
| OPTION_FLAGS | DB2 レプリケーションの将来のオプション用に予約済み。この列には現在、デフォルト値の NNNN が入っています。 |

ASN.IBMSNAP_SUBS_SET

表 83. サブスクリプション・セット表の列 (続き)

| 列名 | 記述 |
|------------------|--|
| COMMIT_COUNT | アプライ・プログラムがサブスクリプション・セットに対して実行する処理のタイプを示すフラグ。 NULL これは、読み取り専用のサブスクリプション・セットの場合のデフォルト設定です。アプライ・プログラムは、すべてのデータが処理されるまで、 <i>n</i> のサブスクリプション・セット・メンバーのメンバー 1 つずつについて、フェッチした応答セットを処理していき、セット全体のデータ処理が終了すると最後に、1 つのコミットを発行します。この COMMIT_COUNT 設定値を使用する利点は、処理が早く終了する可能性があることです。 非 NULL の整数 アプライ・プログラムはトランザクションのモードでサブスクリプション・セットを処理します。すべての応答セットがフェッチされると、各トランザクションが IBMSNAP_INTENTSEQ 値の順に並べられ、コミット・シーケンスの順番で予備ファイルの内容が適用されます。このタイプの処理では、すべての予備ファイルを同時に開いて処理できます。この列で指定された数のトランザクションの後に、コミットが発行されます。たとえば、1 は各トランザクションの後でコミットすることを意味し、2 は、トランザクション 2 つごとにコミットすることを意味します。0 という整数は、すべてのフェッチ・データがアプライされた後で 1 つのコミットを発行することを意味します。トランザクションのモードの処理を使用する利点は、この処理ではターゲットでの参照保全定義が可能であり、暫定のコミットを発行できることです。 |
| MAX_SYNC_MINUTES | サブスクリプション・サイクルでフェッチおよびアプライする変更データの量を規制するための時間しきい値限度。アプライ・プログラムはキャプチャー・サーバーの UOW 表または CCD 表の中の IBMSNAP_LOGMARKER に基づいてサブスクリプション・セットの処理をミニサイクルに分割し、ミニサイクルが正常に終了するたびに、ターゲット・サーバーで COMMIT を発行します。設定された限度が不適切になるようなりソース制約をアプライ・プログラムが検出した場合、この限度は自動的に再計算されます。1 より小さい MAX_SYNC_MINUTES 値は、NULL の MAX_SYNC_MINUTES 値と同じものとして処理されます。 |
| AUX_STMTS | サブスクリプション・ステートメント (IBMSNAP_SUBS_STMTS) 表で定義する SQL ステートメントの数。これらのステートメントは、アプライ・プログラムがサブスクリプション・セットを処理する前または後に実行可能です。 |
| ARCH_LEVEL | 行に含まれる定義の構造レベル。この列は、行を作成する基礎となった規則を識別します。このレベルは、IBM で定義されていて、バージョン 8 では 0801 です。 |

ASN.IBMSNAP_SUBS_STMTS

サーバー: アプライ・コントロール・サーバー

重要: SQL を使用してこの表を更新するときには、注意してください。変更の方法が不適切だと、予期せぬ結果が生じたり、データが失われたりします。サブスクリプションのエントリー数は、ASN.IBMSNAP_SUBS_SET.AUX_STMTS 列に反映されていなければなりません。あるサブスクリプション・セットの AUX_STMTS がゼロの場合、アプライ・プログラムはサブスクリプション・ステートメント表の対応するエントリーを無視します。

サブスクリプション・ステートメント表には、各サブスクリプション・セット処理サイクルの前または後に実行される、ユーザー定義の SQL ステートメントまたはストアード・プロシージャ呼び出しが入っています。即時実行 (EI) ステートメントまたはストアード・プロシージャは、ソースまたはターゲット・サーバーでのみ実行できません。

この表には、SQL ステートメントまたはストアード・プロシージャ呼び出しを使うサブスクリプション・セットを定義するときに、値が挿入されます。

表 84 では、サブスクリプション・ステートメント表の列の要旨を示します。

表 84. サブスクリプション・ステートメント表の列

| 列名 | 記述 |
|-----------------|---|
| APPLY_QUAL | SQL ステートメントまたはストアード・プロシージャを処理するアプライ・プログラムを一意的に識別します。 |
| SET_NAME | SQL ステートメントまたはストアード・プロシージャが関連付けられているサブスクリプション・セットの名前。 |
| WHOS_ON_FIRST | update-anywhere レプリケーション・シナリオでは、処理順序をコントロールするために以下の値を使用します。 F (first の略) ターゲット表はユーザー表または親レプリカです。ソース表は従属のレプリカであり、ソース表とターゲット表との間で更新の競合が生じた場合、ソース表の競合するトランザクションは拒否されます。F は、読み取り専用サブスクリプションについては使用されません。 S (second の略) ソース表はユーザー表、親レプリカ、またはその他のソースです。ターゲット表は従属のレプリカまたは他のコピーであり、ソース表とターゲット表との間で更新の競合が生じた場合、ターゲット表の競合するトランザクションは拒否されます。S は、すべての読み取り専用サブスクリプションについて使用されます。 |
| BEFORE_OR_AFTER | ステートメントの発行時間と場所を示す値。 A ステートメントは、すべての応答セット行が適用された後、ターゲット・サーバーで実行されます。 B ステートメントは、応答セット行が適用される前に、ターゲット・サーバーで実行されます。 S ステートメントは、応答セット・カーソルを開く前に、キャプチャー・コントロール・サーバー上で実行されます。 G DB2 レプリケーションの使用のために予約済み。 X DB2 レプリケーションの使用のために予約済み。 |
| STMT_NUMBER | BEFORE_OR_AFTER 列値の有効範囲で、実行の相対順序を定義します。 |
| EI_OR_CALL | 次のどちらであるかを示す値。 E SQL ステートメントは、EXEC SQL EXECUTE IMMEDIATE として実行されません。 C SQL ステートメントには、EXEC SQL CALL として実行されるストアード・プロシージャ名が入っています。 |

ASN.IBMSNAP_SUBS_STMTS

表 84. サブスクリプション・ステートメント表の列 (続き)

| 列名 | 記述 |
|------------------|--|
| SQL_STMT | 以下のいずれかの値が入ります。 ステートメント EI_OR_CALL = E の場合に EXEC SQL EXECUTE IMMEDIATE ステートメントとして実行される SQL ステートメント。 プロシージャ EI_OR_CALL が C の場合に、EXEC SQL CALL ステートメントとして実行される、パラメーター、または CALL キーワードなしの SQL ストアド・プロシージャの 8 バイトの名前。 |
| ACCEPT_SQLSTATES | サブスクリプション・セットの定義時に指定した 1~10 個の 5 バイト SQLSTATE 値。これらの非ゼロ値は、正常実行としてアプライ・プログラムで受け入れられます。それ以外の値は実行が失敗する原因となります。 |

モニター・コントロール・サーバー上の表、およびその列に関する記述

このセクションでは、モニター・コントロール・サーバー上に保管される各表について、かなり詳しく説明します。また、各表の列のリストやその簡単な説明もあります。コントロール表はアルファベット順にリストされ、列は各表内に現れる順に左から右へリストされています。

ASN.IBMSNAP_ALERTS

サーバー: モニター・コントロール・サーバー

モニター・アラート表は、レプリケーション・アラート・モニターから発行されるすべてのアラートのレコードを保持します。この表では、どのようなアラート条件が発生したか、どのサーバーで発生したか、そしていつアラートが検出されたかがトラッキングされます。

表 85 では、モニター・アラート表の列の要旨を示します。

表 85. モニター・アラート表の列

| 列名 | 記述 |
|--------------|--|
| MONITOR_QUAL | アラートを発行したレプリケーション・アラート・モニター・プログラムを識別するモニター修飾子。 |
| ALERT_TIME | この表にアラートが挿入されたときのモニター・コントロール・サーバーの時刻。 |
| COMPONENT | モニター対象のレプリケーション・コンポーネント。 C キャプチャー・プログラム A アプライ・プログラム |
| SERVER_NAME | アラート条件が発生した、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの名前。 |
| SERVER_ALIAS | アラート条件が発生した、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの DB2 Universal Database 別名。 |

表 85. モニター・アラート表の列 (続き)

| 列名 | 記述 |
|-------------------|---|
| SCHEMA_OR_QUAL | モニター対象のキャプチャー・スキーマまたはアプライ修飾子。 |
| SET_NAME | アプライ・プログラムでアラート条件が設定されている場合、この列は、モニター対象のサブスクリプション・セットの名前を指定します。セット名が指定されない場合は、アプライ修飾子レベルでモニターが行われます。つまり、特定のアプライ修飾子を持つすべてのセットがモニターされます。 |
| CONDITION_NAME | アラートが起動されたときにテストされた条件コード。 |
| OCCURRED_TIME | キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーでアラート条件が発生した時刻。 |
| ALERT_COUNTER | 連続するモニター・サイクルで、このアラートが以前に検出された回数。 |
| ALERT_CODE | アラートの発生時に発行されたメッセージ・コード。 |
| RETURN_CODE | ユーザー条件から戻された整数値。 |
| NOTIFICATION_SENT | 通知メッセージが送信されたかどうかを示すフラグ。 Y 通知メッセージが送信されました。 N MAX_NOTIFICATIONS_PER_ALERT および MAX_NOTIFICATIONS_MINUTES パラメーターのため、通知メッセージは送信されていません。 |
| ALERT_MESSAGE | メッセージ・コードを含めて、送信されたメッセージのテキスト。 |

ASN.IBMSNAP_CONDITIONS

サーバー: モニター・コントロール・サーバー

モニター条件表には、レプリケーション・アラート・モニターから担当者への連絡が必要なアラート条件と、特定の条件が発生したときの連絡先のグループまたは個人の名前が含まれます。レプリケーション・アラート・モニターは、キャプチャー・コントロール・サーバーおよびアプライ・コントロール・サーバー上の複数の条件の組み合わせをモニターできます。

表 86 では、モニター条件表の列の要旨を示します。

表 86. モニター条件表の列

| 列名 | 記述 |
|----------------|--|
| SERVER_NAME | この条件のモニター対象のキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの名前。 |
| COMPONENT | モニター対象のレプリケーション・コンポーネント。 C キャプチャー・プログラム A アプライ・プログラム |
| SCHEMA_OR_QUAL | モニター対象のキャプチャー・スキーマまたはアプライ修飾子。 |
| SET_NAME | アプライ・プログラムでアラート条件が設定されている場合は、モニター対象のサブスクリプション・セットの名前。セット名が指定されない場合は、アプライ修飾子レベルでモニターが行われます。つまり、特定のアプライ修飾子を持つすべてのセットがモニターされます。 |
| MONITOR_QUAL | この条件について、キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーをモニターしているレプリケーション・アラート・モニター・プログラムを識別するモニター修飾子。 |

ASN.IBMSNAP_CONDITIONS

表 86. モニター条件表の列 (続き)

| 列名 | 記述 |
|----------------|--|
| SERVER_ALIAS | この条件のモニター対象のキャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの DB2 Universal Database 別名。 |
| ENABLED | レプリケーション・アラート・モニターが次のモニター・サイクルでこの条件を処理するかどうかを示すフラグ。 Y レプリケーション・アラート・モニターは次のモニター・サイクルでこの定義を処理します。 N レプリケーション・アラート・モニターは次のモニター・サイクルでこの定義を無視します。 |
| CONDITION_NAME | 特定のキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーで、レプリケーション・アラート・モニターがモニターしている条件の名前。キャプチャー・プログラムの条件は CAPTURE で始まり、アプライ・プログラムの条件は APPLY で始まり ます。 CAPTURE_STATUS キャプチャー・プログラムの状況。 CAPTURE_ERRORS キャプチャー・プログラムがエラー・メッセージを通知するかどうか。 CAPTURE_WARNINGS キャプチャー・プログラムが警告メッセージを通知するかどうか。 |

表 86. モニター条件表の列 (続き)

| 列名 | 記述 |
|---------------------|--|
| CONDITION_NAME (続き) | <p>CAPTURE_LASTCOMMIT 最後のモニター・サイクルでキャプチャー・プログラムが最後にデータをコミットした時刻。</p> <p>CAPTURE_CLATENCY キャプチャー・プログラムの現在の待ち時間。</p> <p>CAPTURE_HLATENCY キャプチャー・プログラムの待ち時間が特定の秒数を超えたかどうか。</p> <p>CAPTURE_MEMORY キャプチャー・プログラムが使用しているメモリーの量 (MB)。</p> <p>APPLY_STATUS アプライ・プログラムの状況。</p> <p>APPLY_SUBSFALING 失敗したサブスクリプション・セットがあるかどうか。</p> <p>APPLY_SUBSINACT 失敗した、または非アクティブのサブスクリプション・セットがあるかどうか。</p> <p>APPLY_ERRORS アプライ・プログラムがエラー・メッセージを通知するかどうか。</p> <p>APPLY_WARNINGS アプライ・プログラムが警告メッセージを通知するかどうか。</p> <p>APPLY_FULLREFRESH フル・リフレッシュが発生したかどうか。</p> <p>APPLY_REJTRANS (update-anywhere) アプライ・プログラムがサブスクリプション・セットでトランザクションを拒否するかどうか。</p> <p>APPLY_SUBSDELAY アプライ・プログラムが、ユーザーが COND_INT パラメーターで指定した時間よりも遅れるかどうか。</p> <p>APPLY_REWORKED アプライ・プログラムが、ターゲット表の行の再処理を行ったかどうか。</p> <p>APPLY_LATENCY アプライ・プログラムのエンドツーエンドの待ち時間がしきい値を超えるかどうか。</p> |
| PARAM_INT | 条件の整数パラメーター。このパラメーターの値は、条件のロジックに基づきます。ストアード・プロシージャから使用されるにせよ、使用されないにせよ、入力パラメーターとして渡されます。 |
| PARAM_CHAR | 条件の文字パラメーター。このパラメーターの値は、条件のロジックに基づきます。ストアード・プロシージャから使用されるにせよ、使用されないにせよ、入力パラメーターとして渡されます。 |

ASN.IBMSNAP_CONDITIONS

表 86. モニター条件表の列 (続き)

| 列名 | 記述 |
|--------------|--|
| CONTACT_TYPE | この条件が発生したときに個人またはグループに連絡するかどうかを示すフラグ。 C 個人連絡先 G 連絡先のグループ |
| CONTACT | この条件が発生したときに通知する個人連絡先、または連絡先のグループ。 |

ASN.IBMSNAP_CONTACTGRP

サーバー: モニター・コントロール・サーバー

モニター・グループ連絡先表には、連絡先グループを構成する個人の連絡先が入っています。アラート条件が発生したときに、これらの個人のグループにレプリケーション・アラート・モニターから連絡がいくように指定できます。1 個人は、複数の連絡先グループに所属できます (列は固有ではありません)。

表 87 では、モニター・グループ連絡先表の列の要旨を示します。

表 87. モニター・グループ連絡先表の列

| 列名 | 記述 |
|--------------|---|
| GROUP_NAME | 連絡先グループの名前。 |
| CONTACT_NAME | グループの一部である連絡先の名前。これらの個人は、モニター連絡先 (IBMSNAP_CONTACTS) 表で指定されます。 |

ASN.IBMSNAP_CONTACTS

サーバー: モニター・コントロール・サーバー

モニター連絡先表は、個人 (またはそのグループ) に関連付けられたアラート条件が発生したときに、レプリケーション・アラート・モニターから個人に通知を行うために必要となる情報が入っています。1 行に 1 人指定します。

表 88 では、モニター連絡先表の列の要旨を示します。

表 88. モニター連絡先表の列

| 列名 | 記述 |
|----------------|---|
| CONTACT_NAME | 連絡先の名前。 |
| EMAIL_ADDRESS | この連絡先の主な E メールまたはポケットベルのアドレス。 |
| ADDRESS_TYPE | この連絡先の E メール・アドレスが、E メール・アカウントであるか、ポケットベル・アドレスであるかを示すフラグ。 E E メール・アドレスは E メール・アカウント用です。 P E メール・アドレスはポケットベル用です。 |
| DELEGATE | 代行期間中に通知を受ける連絡先の名前。 |
| DELEGATE_START | DELEGATE 列で指定された個人に通知を送信する場合の、代行期間の開始日。 |

表 88. モニター連絡先表の列 (続き)

| 列名 | 記述 |
|--------------|-----------|
| DELEGATE_END | 代行期間の終了日。 |
| DESCRIPTION | 連絡先の記述。 |

ASN.IBMSNAP_GROUPS

サーバー: モニター・コントロール・サーバー

モニター・グループ表には、各連絡先グループの名前と記述が入っています。1 つの行に 1 つのグループが指定されます。

表 89 では、モニター・グループ表の列の要旨を示します。

表 89. モニター・グループ表の列

| 列名 | 記述 |
|-------------|-------------|
| GROUP_NAME | 連絡先グループの名前。 |
| DESCRIPTION | 連絡先グループの記述。 |

ASN.IBMSNAP_MONENQ

サーバー: モニター・コントロール・サーバー

モニター・エンキュー表は、DB2 レプリケーションの将来のオプション用に予約済みです。

表 90 では、モニター・エンキュー表の列の要旨を示します。

表 90. モニター・エンキュー表の列

| 列名 | 記述 |
|--------------|------------------------------|
| MONITOR_QUAL | DB2 レプリケーションの将来のオプション用に予約済み。 |

ASN.IBMSNAP_MONSERVERS

サーバー: モニター・コントロール・サーバー

モニター・サーバー表は、レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーを最後にモニターしたときの情報を保持します。

540 ページの表 91 では、モニター・サーバー表の列の要旨を示します。

ASN.IBMSNAP_MONSERVERS

表 91. モニター・サーバー表の列

| 列名 | 記述 |
|--------------------|---|
| MONITOR_QUAL | キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーをモニターしているレプリケーション・アラート・モニターを識別するモニター修飾子。 |
| SERVER_NAME | レプリケーション・アラート・モニターによるモニター対象のキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの名前。 |
| SERVER_ALIAS | レプリケーション・アラート・モニターによるモニター対象のキャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの DB2 UDB 別名。 |
| LAST_MONITOR_TIME | レプリケーション・アラート・モニター・プログラムがこのサーバーに最後に接続したときの時刻 (キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの)。この値は、コントロール表からメッセージをフェッチする下限値として使用され、最後に成功したモニター・サイクルの START_MONITOR_TIME の値と同じ値になります。 |
| START_MONITOR_TIME | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーに接続したときの時刻 (キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの)。この値は、コントロール表からアラート・メッセージをフェッチする上限値として使用されます。 |
| END_MONITOR_TIME | レプリケーション・アラート・モニターがこのサーバーのモニターを終了したときの時刻 (キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの)。 |
| LASTRUN | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの処理を最後に開始したときの時刻 (モニター・コントロール・サーバーの)。 |
| LASTSUCCESS | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの処理を最後に正常に完了した時刻の (モニター・コントロール・サーバーの) LASTRUN 列の値。このサーバーのモニターが繰り返し失敗するときには、この値が同じである可能性があります (この列の履歴は IBMSNAP_MONTRAIL 表にあります)。 |
| STATUS | モニター・サイクルの状況を示すフラグ。 -1 レプリケーション・アラート・モニターは、このサーバーを正常に処理できませんでした。 0 レプリケーション・アラート・モニターは、このサーバーを正常に処理しました。 1 レプリケーション・アラート・モニターは、このサーバーを現在処理中です。 |

ASN.IBMSNAP_MONTRACE

サーバー: モニター・コントロール・サーバー

モニター・トレース表には、レプリケーション・アラート・モニターの監査証跡情報が含まれます。レプリケーション・アラート・モニターによる処理はすべてこの表に記録されるため、レプリケーション・アラート・モニター・プログラムの問題が発生した場合は、この表を参照すると便利です。

表 92 では、モニター・トレース表の列の要旨を示します。

表 92. モニター・トレース表の列

| 列名 | 記述 |
|--------------|---|
| MONITOR_QUAL | メッセージを発行したレプリケーション・アラート・モニターを示すモニター修飾子。 |

表 92. モニター・トレース表の列 (続き)

| 列名 | 記述 |
|-------------|---|
| TRACE_TIME | メッセージがこの表に挿入されたときのタイム・スタンプ。 |
| OPERATION | メッセージを分類するために使用される値。 ERROR エラー・メッセージ WARNING 警告メッセージ INFO 情報メッセージ |
| DESCRIPTION | メッセージ・コードおよびテキスト。 |

ASN.IBMSNAP_MONTRAIL

サーバー: モニター・コントロール・サーバー

モニター・トレール表には、各モニター・サイクルに関する情報が入ります。レプリケーション・アラート・モニターは、モニター対象のキャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーごとに行を 1 つ挿入します。

表 93 では、モニター・トレール表の列の要旨を示します。

表 93. モニター・トレール表の列

| 列名 | 記述 |
|-------------------|---|
| MONITOR_QUAL | キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーをモニターしているレプリケーション・アラート・モニターを識別するモニター修飾子。 |
| SERVER_NAME | レプリケーション・アラート・モニターによるモニター対象のキャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの名前。 |
| SERVER_ALIAS | レプリケーション・アラート・モニターによるモニター対象のキャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの DB2 Universal Database 別名。 |
| STATUS | モニター・サイクルの状況を示すフラグ。 -1 レプリケーション・アラート・モニターは、このサーバーを正常に処理できませんでした。 0 レプリケーション・アラート・モニターは、このサーバーを正常に処理しました。 1 レプリケーション・アラート・モニターは、このサーバーを現在処理中です。 |
| LASTRUN | レプリケーション・アラート・モニター・プログラムが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの処理を最後に開始したときの時刻 (モニター・コントロール・サーバーの)。 |
| LASTSUCCESS | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーの処理を最後に正常に完了したときの時刻 (モニター・コントロール・サーバーの)。 |
| ENDTIME | この表にこの行が挿入された時刻。 |
| LAST_MONITOR_TIME | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーに最後に接続したときの時刻 (キャプチャー・コントロール・サーバーまたはアプライ・コントロール・サーバーの)。この値は、コントロール表からメッセージをフェッチする下限値として使用され、直前の成功したモニター・サイクルの START_MONITOR_TIME の値と同じ値になります。 |

表 93. モニター・トレール表の列 (続き)

| 列名 | 記述 |
|--------------------|--|
| START_MONITOR_TIME | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーのモニターを最後に開始した時刻。 |
| END_MONITOR_TIME | レプリケーション・アラート・モニターが、キャプチャー・コントロール・サーバー、またはアプライ・コントロール・サーバーのモニターを最後に終了した時刻。 |
| SQLCODE | このモニター・サイクル中に発生したエラーの SQLCODE。 |
| SQLSTATE | このモニター・サイクル中に発生したエラーの SQLSTATE。 |
| NUM_ALERTS | このモニター・サイクル中に発生したアラート条件の数。 |
| NUM_NOTIFICATIONS | このモニター・サイクル中に送信された通知の数。 |

ターゲット・サーバー上の表、およびその列に関する記述

このセクションでは、ターゲット・サーバーで使用される各表について、かなり詳しく説明します。また、各表の列のリストやその簡単な説明もあります。表名はアルファベット順にリストされ、列名は各表内に現れる順に左から右へリストされています。

基礎集約表

schema.base_aggregate

サーバー: ターゲット・サーバー

重要: SQL を使用してこの表をリフレッシュした場合は、アプライ・プログラムによってフル・リフレッシュが実行されたときにユーザーのリフレッシュが失われる危険性があります。

基礎集約表は、ソース表上にあるデータに対して実行された集約関数の結果を含むターゲット表です。

表 94 では、基礎集約表の列の要旨を示します。

表 94. 基礎集約表の列

| 列名 | 記述 |
|---------------------|--|
| <i>user columns</i> | ソース表から算出された集約データ。 |
| IBMSNAP_LLOGMARKER | ソース表のデータの集約が開始されたときのソース・サーバーの現行タイム・スタンプ。 |
| IBMSNAP_HLOGMARKER | ソース表のデータの集約が完了したときのソース・サーバーの現行タイム・スタンプ。 |

変更集約表

schema.change_aggregate

サーバー: ターゲット・サーバー

重要: SQL を使用してこの表をリフレッシュした場合は、アプライ・プログラムによってフル・リフレッシュが実行されたときにユーザーのリフレッシュが失われる危険性があります。

変更集約表は、変更データ (CD) 表にあるデータに対して実行された集約関数の結果を含むターゲット表です。この表は基礎集約表と似ていますが、CD 表で実行される関数は、特定の時間インターバルで発生する変更に対してのみ使用される点が異なります。

表 95 では、変更集約表の列の要旨を示します。

表 95. 変更集約表の列

| 列名 | 記述 |
|------------------------------|---|
| <i>user key columns</i> | ターゲット・キーを構成する列。 |
| <i>user nonkey columns</i> | ソース表からの非キー・データ列。このターゲット表の中の列名はソース表の中の列名と一致している必要はありませんが、データ・タイプは一致している必要があります。 |
| <i>user computed columns</i> | SQL 式から派生したユーザー定義の列。ソース・データ・タイプを別のターゲット・データ・タイプに変換するために、SQL 関数で算出列を使用することができます。 |
| IBMSNAP_LLOGMARKER | 集約されている (CD+UOW) または CCD 表内の最も古い IBMSNAP_LOGMARKER または IBMSNAP_LLOGMARKER 値。 |
| IBMSNAP_HLOGMARKER | 集約されている (CD+UOW) または CCD 表内の最も新しい IBMSNAP_LOGMARKER または IBMSNAP_HLOGMARKER 値。 |

整合変更データ (CCD) 表

schema.CCD_table

この表の情報は、SQL を使って更新できます。

サーバー: ターゲット・サーバー

重要: SQL を使用してこの表をリフレッシュした場合は、アプライ・プログラムによってフル・リフレッシュが実行されたときにユーザーのリフレッシュが失われる危険性があります。

整合変更データ (CCD) 表は、ソースで発生した変更に関する情報と、これらの変更の順序を識別するための追加の列を含む、サブスクリプション・セット・メンバー内のターゲットです。列の値は、CD 表と UOW 表の結合により得られます。ターゲット・サーバーである CCD 表としては次のものが考えられます。

- CD 表の代替として機能する内部 CCD 表。
変更キャプチャー・レプリケーションの場合、アプライ・プログラムはこの表からターゲットへ直接変更を適用します。このタイプの CCD 表の名前は、登録 (IBMSNAP_REGISTER) 表の中で、変更元のレプリケーション・ソースと同じ行に保管されます。
- 読み取り専用のターゲット表である外部 CCD。

整合変更データ表

このタイプの CCD は、ターゲット・サーバーのソース・データの監査証跡を保持します。

- multi-tier のレプリケーション構成の中間層である外部 CCD。

このタイプの CCD は、層 1 に対してはターゲット表であり、層 3 に対してはソース表です。このタイプの CCD 表の名前は、登録 (IBMSNAP_REGISTER) 表の中のそれ自身の行に保管されます。

CCD 表およびターゲットの使用方法的詳細は、85 ページの『ターゲット・タイプの選択』を参照してください。

キャプチャー・プログラムは、CCD 表にデータを挿入せず、この表の整理も行いません。アプリケーションの要件に応じて、CCD 表の履歴の保存期間を決定してください。したがって、デフォルトでは CCD 表のプルーニングは自動ではありませんが、サブスクリプション・サイクルの後に処理される SQL ステートメントを使えば簡単に自動化することができます。

外部 CCD の場合、UOW 表から次のいくつかの列を含めるように選択できます。APPLY_QUAL、IBMSNAP_AUTHID、IBMSNAP_AUTHTKN、IBMSNAP_REJ_CODE、および IBMSNAP_UOWID。

IBMSNAP_OPERATION 列で最初に組み込まれた操作コードと、シーケンス番号 IBMSNAP_INTENTSEQ および IBMSNAP_COMMITSEQ は、CCD 表に含まれます。圧縮 CCD 表の場合、それぞれの列ごとに最新の値のみが保持されます。

キャプチャー・トリガーにより取り込まれる、または非リレーショナル・データを含む CCD 表の詳細は、499 ページの『*schema.CCD_table* (DB2 以外)』を参照してください。

表 96 では、CCD 表の列の要旨を示します。

表 96. CCD 表の列

| 列名 | 記述 |
|----------------------------|---|
| IBMSNAP_INTENTSEQ | 1 つの変更を固有に識別するログまたはジャーナル・レコードのシーケンス番号この値はグローバルに昇順です。 |
| IBMSNAP_OPERATION | レコードの操作のタイプを示すフラグ。 I 挿入 U 更新 D 削除 |
| IBMSNAP_COMMITSEQ | 取り込まれたコミット・ステートメントのログ・レコード・シーケンス番号。この値は、ソース表の元のトランザクションによる挿入、更新、および削除をグループ化します。 |
| IBMSNAP_LOGMARKER | キャプチャー・コントロール・サーバーにおけるコミット時刻。 |
| <i>user key columns</i> | CCD 表が圧縮されている場合、この列には、ターゲット・キーを構成する列が含まれます。 |
| <i>user nonkey columns</i> | ソース表からの非キー・データ列。このターゲット表の中の列名はソース表の中の列名と一致している必要はありませんが、データ・タイプの互換性は必要です。 |

表 96. CCD 表の列 (続き)

| 列名 | 記述 |
|------------------------------|---|
| <i>user computed columns</i> | SQL 式から派生したユーザー定義の列。ソース・データ・タイプを別のターゲット・データ・タイプに変換するために、SQL 関数で算出列を使用することができます。 |
| APPLY_QUAL (オプション) | この CCD 表を処理するアプライ・プログラムを一意的に識別します。 |
| IBMSNAP_AUTHID (オプション) | このトランザクションに関連する許可 ID。これはデータベースの監査に役立ちます。AUTHID の長さは 18 文字です。それを超えると切り捨てられます。DB2 Universal Database for z/OS の場合、この列は 1 次許可 ID です。DB2 Universal Database for iSeries の場合、この列は、トランザクションを発生させたアプリケーションを実行しているユーザー・プロファイル ID の名前になります。この列は、ブランクを埋め込んだ 10 文字の ID を保留します。この列は自動的に他の表にコピーされません。ユーザー・データ列として選択してコピーする必要があります。この列は、非コンプリート CCD ターゲット表のユーザー・データ列として選択できます。 |
| IBMSNAP_AUTHTKN (オプション) | このトランザクションに関連する許可トークン。この ID は、データベースの監査に役立ちます。DB2 Universal Database for z/OS の場合、この列は関連 ID です。DB2 Universal Database for iSeries の場合、この列は、トランザクションを発生させたジョブのジョブ名です。この列は自動的に他の表にコピーされません。ユーザー・データ列として選択してコピーする必要があります。この列は、非コンプリート CCD ターゲット表のユーザー・データ列として選択できます。 |
| IBMSNAP_REJ_CODE (オプション) | レプリケーション・ソースの定義時に競合検出が標準または詳細と指定されている場合、この値は update-anywhere レプリケーションでのみ設定されます。update-anywhere 構成に参加できないため、DB2 以外のリレーショナル・ターゲットの場合は使用できません。 0 認識された競合のないトランザクション。 1 ソース表とレプリカ表の同じ行に、複製されなかった変更による競合があるトランザクション。競合が生じると、トランザクションはレプリカ表でロールバックされます。 2 最低でも 1 つの同一行競合がある、以前のトランザクションに従属するトランザクションのカスケード拒否。競合が生じると、トランザクションはレプリカ表でロールバックされます。 3 最低でも 1 つの参照保全制約違反があるトランザクション。このトランザクションはソース表で定義された参照制約に違反しているため、アプライ・プログラムはこのサブスクリプション・セットに失敗というマークを付けます。参照保全定義が訂正されるまで、更新はコピーできません。 4 最低でも 1 つの制約競合がある、以前のトランザクションに従属するトランザクションのカスケード拒否。 |
| IBMSNAP_UOWID (オプション) | この作業単位についての、ログ・レコード・ヘッダーからの作業単位 ID。 |

関連資料:

- 499 ページの 『*schema.CCD_table* (DB2 以外)』

ポイント・イン・タイム表

schema.point_in_time

サーバー: ターゲット・サーバー

ポイント・イン・タイム表

重要: SQL を使用してこの表をリフレッシュした場合は、アプライ・プログラムによってフル・リフレッシュが実行されたときにユーザーのリフレッシュが失われる危険性があります。

ポイント・イン・タイム表は、ソース・データのコピーと、特定の行がソース・サーバーで挿入または更新されたおおよそのポイント・イン・タイムのタイム・スタンプを含む追加のシステム列 (IBMSNAP_LOGMARKER) を保持します。

表 97 では、ポイント・イン・タイム表の列の要旨を示します。

表 97. ポイント・イン・タイム表の列

| 列名 | 記述 |
|------------------------------|--|
| <i>user key columns</i> | ターゲット・キーを構成する列。 |
| <i>user nonkey columns</i> | ソース表またはビューからの非キー・データ列。このターゲット表の中の列名はソース表の中の列名と一致している必要はありませんが、データ・タイプは一致している必要があります。 |
| <i>user computed columns</i> | SQL 式から派生したユーザー定義の列。ソース・データ・タイプを別のターゲット・データ・タイプに変換するために、SQL 関数で算出列を使用することができます。 |
| IBMSNAP_LOGMARKER | キャプチャー・コントロール・サーバーにおけるおおよそのコミット時刻。この列は、フル・リフレッシュ後は NULL になります。 |

レプリカ表

schema.replica

サーバー: ターゲット・サーバー

この表の情報は、SQL を使って更新できます。

レプリカ表には、ソース表と同じ主キー列が必要です。この類似性のために、レプリカ表は、その後のサブスクリプション・セットでソース表として使用できます。ターゲット表からソース表への変換は、ユーザーがレプリカ・ターゲット・タイプを定義し、CHANGE DATA CAPTURE 属性を指定すると、自動的に行われます。詳細については、94 ページの『読み取り / 書き込みターゲットの定義 (update-anywhere)』を参照してください。

表 98 では、レプリカ表の列の要旨を示します。

表 98. レプリカ表の列

| 列名 | 記述 |
|----------------------------|--|
| <i>user key columns</i> | ターゲット・キーを構成する列。マスター表と同じ主キーであることが必要です。 |
| <i>user nonkey columns</i> | ソース表からの非キー・データ列。このターゲット表の中の列名はソース表の中の列名と一致している必要はありませんが、データ・タイプは一致している必要があります。 |

ユーザー・コピー表

schema.user_copy

サーバー: ターゲット・サーバー

重要: SQL を使用してこの表をリフレッシュした場合は、アプライ・プログラムによってフル・リフレッシュが実行されたときにユーザーのリフレッシュが失われる危険性があります。

ユーザー・コピー表は、ソース表の列のコピーを含むターゲット表です。このターゲット表では、ソース表の行または列のサブセットを使用できますが、追加列を含めることはできません。

サブセット化とデータ拡張の場合を除き、ユーザー・コピー表はソース表の正しい状態を反映しますが、それが最新の状態であるとはかぎりません。ユーザー・コピー表 (または他の任意のターゲット表タイプ) への参照により、ソース表への直接アクセスが多すぎるときに生じる競合を減らすことができます。ローカル・ユーザー・コピー表にアクセスすると、照会ごとにネットワークを使用してリモート・ソース表へアクセスするよりも迅速です。

表 99 では、ユーザー・コピー表の列の要旨を示します。

表 99. ユーザー・コピー表の列

| 列名 | 記述 |
|------------------------------|--|
| <i>user key columns</i> | ターゲット・キーを構成する列。 |
| <i>user nonkey columns</i> | ソース表またはビューからの非キー・データ列。このターゲット表の中の列名はソース表の中の列名と一致している必要はありませんが、データ・タイプは一致している必要があります。 |
| <i>user computed columns</i> | SQL 式から派生したユーザー定義の列。ソース・データ・タイプを別のターゲット・データ・タイプに変換するために、SQL 関数で算出列を使用することができます。 |

第 24 章 レプリケーション・メッセージ

本章には、DB2 (iSeries 版) 以外のすべてのデータベース管理システムのレプリケーション・プログラムについて、DB2 レプリケーションが出すメッセージのリスト含まれています。本章の各メッセージの下に、メッセージの説明および、操作を再開するためのアクションの提案が記述されています。DB2 コマンド・プロンプトで次のコマンドを入力し、メッセージの説明を表示することもできます。

`db2 message_number`

iSeries 版の注: DB2 (iSeries 版) で表示されるレプリケーション・メッセージは、レプリケーション・メッセージ・ファイル (MSGF) から、オンラインでのみ使用できます。コマンド行からメッセージを表示するには、`[DSPMSGD RANGE(message_number) MSGF(library_name/filename)]` と入力します。

本章のレプリケーション・メッセージの範囲は、以下のとおりです。

ASN0001 から ASN0499

キャプチャー・プログラムのメッセージ

ASN0500 から ASN0999

キャプチャー・プログラムとアプライ・プログラムに共通のメッセージ

ASN1000 から ASN1499

アプライ・プログラムのメッセージ

ASN1500 から ASN1999

レプリケーション・センターのメッセージ

ASN5000 から ASN5099

マイグレーション関連のメッセージ

ASN5100 から ASN5199

レプリケーション・アラート・モニターのメッセージ

ASN5200 から ASN5299

NT サービスのメッセージ

特に記述のないかぎり、本章のエラー・コードはすべて、特定のメッセージが出された場所を見つけるために、IBM ソフトウェア・サポートが使用する内部エラー・コードです。また、特に記述のないかぎり、エラー・メッセージにはゼロ以外の戻りコードが使用されます。

キャプチャー・エラーのヒント: キャプチャー・プログラム・エラーが出された場合は、DB2 保守が最新レベルであることを検査してください。キャプチャー・プログラム

は、DB2 API を使用するアプリケーション・プログラムです。キャプチャー・プログラムのエラーの多くは、DB2 保守が最新レベルでないことが原因です。

ASN0004E CAPTURE *capture_schema*. The Capture program could not start the trace. The return code is *return_code*. The reason code is *reason_code*.

説明: START TRACE DB2 コマンドの実行時または、キャプチャー・プログラムが DB2 ログを読み取った時に、エラーが起きました。

ユーザーの処置: ご使用のオペレーティング・システムの DB2 データベース・マネージャーの、「メッセージおよびコード」資料の DB2 コードのセクションを参照し、該当する理由コードを見つけてください。詳細は、次の管理資料のいずれかを参照してください。START TRACE DB2 エラーの場合は Call Attachment Facility (CAF)、DB2 ログ読み取りエラーの場合は Instrumentation Facility Interface (IFI)、あるいは DBA にご連絡ください。CAF または IFI が戻すメッセージは、システム表示コンソールにも印刷されます。

ASN0005E CAPTURE *capture_schema*. The Capture program encountered an error when reading the DB2 log. The log sequence number is *lsn*, the SQLCODE is *sql_return_code*, and the reason code is *reason_code*.

説明: キャプチャー・プログラムが DB2 ログを読み取った時にエラーが起きました。これは SQL エラーの可能性がります。

- DB2 レプリケーションの場合、*sqlcode* 値は非同期読み取りログ API に対するものです。
- VSE 用キャプチャーの場合、*sqlcode* は VSE/VSAM GET マクロに対するものです。
- VM 用キャプチャーの場合、*sqlcode* は診断 X'A4' に対するものです。

ユーザーの処置: ご使用のオペレーティング・シ

ステムの DB2 データベース・マネージャーの、「メッセージおよびコード」資料の DB2 コードのセクションを参照し、次のように該当する理由コードを見つけてください。

- z/OS のキャプチャー・プログラムの場合、使用オペレーティング・システムの DB2 データベース・マネージャーの管理資料の、Instrumentation Facility Interface (IFI) セクションを参照するか、または DBA に連絡してください。
- VSE のキャプチャーの場合、「VSE/VSAM コマンドおよびマクロ」、「VSE/ESA システム・マクロリファレンス」、および「VSE/ESA V2R3 メッセージおよびコード」の資料で詳細を参照してください。
- VM/ESA の場合、「VM/ESA プログラミング・サービス」で詳細を参照してください。
- Linux、Windows、および UNIX のキャプチャーの場合、DB2 Universal Database のアクティブまたはアーカイブ・データベース・ログ管理資料を参照するか、または IBM ソフトウェア・サポートに連絡してください。

ASN0006E CAPTURE *capture_schema*. The Capture program encountered an unexpected log error of unknown log variation.

説明: キャプチャー・プログラムが DB2 ログ・レコードを処理中に、予期しないログ・エラーが起り、次のいずれかのインターフェースにより報告されませんでした。

- z/OS 用キャプチャー・プログラムの Instrumentation Facility Interface (IFI)
- キャプチャー・プログラムの非同期読み取りログ API

キャプチャー・プログラムは、ログ・レコードにどのタイプの SQL 更新が関連しているかを判別できませんでした。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN0008I CAPTURE *capture_schema*. The Capture program was stopped.

説明: キャプチャー・プログラムが停止しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0009E CAPTURE *capture_schema*. The registered source table *src_owner.src_table* does not have the DATA CAPTURE CHANGES attribute.

説明: 登録を初期化しようとした時に、キャプチャー・プログラムはソース表が誤って定義されていることを検出しました。キャプチャー・プログラムは、ソース表に DATA CAPTURE CHANGES 属性が設定されていないと、そのソース表に関連するログ・レコードを処理できません。この登録は無効であり、これは通常、提供された管理ツールを使用せずにこの登録を生成した場合にのみ起こります。このメッセージは、キャプチャー・プログラムのウォーム・スタート時または、この登録に対してサブスクリプション・セットに最初の CAPSTART シグナルを受け取った時に、出されます。CAPSTART シグナルの処理中にこのエラーが起こった場合、キャプチャー・プログラムは登録を活動化しません。このエラーが (ウォーム・スタートまたは `reinit` キャプチャー・コマンドからの) 再初期化中に起こった場合、キャプチャー・プログラムはその登録を「停止」状態にします。この場合、登録を修復してからでないと、アプライ・プログラムは関連するサブスクリプション・セットを再同期化できないことを示します。

ユーザーの処置:

1. ソース表の「データ・キャプチャー変更」をオンに変更します。たとえば、次のように指定します。
 - `alter table regress.table3 data capture changes`
2. キャプチャー・プログラムが登録を非活動化した場合 (`state = stopped`)、登録を更新し、状態を `inactive` に設定します。
3. レプリケーション・センターを使用して、アプライ・プログラムに強制的に、このソース表からレプリケーションするすべてのサブスクリプション・セットのフル・リフレッシュを実行させます。

ASN0011E CAPTURE *capture_schema*. The Capture program log read failed because the DB2 compression dictionary that was used to create the compressed log record no longer exists. The log record that could not be read was for the registered source table *src_owner.src_table*. The reason code is *reason_code*.

説明: キャプチャー・プログラムは、DB2 ログ読み取り IFI からゼロ以外の応答コードを受け取りました。応答コードは、対応する DB2 表スペースの圧縮辞書が使用できないために、ログ・レコード上のデータを処理できないことを示します。

このソース表を含む圧縮された表スペースはおそらく、`KEEPDICTIONARY` オプションを使用しないで `REORG` コーティリティーを実行して、再編成されたものです。残りの圧縮ログ・レコードを読み取れないので、キャプチャー・プログラムはこの登録を非活動化する必要があります。この登録を非活動化するかまたは除去しないと、キャプチャー・プログラムは続行できません。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: 圧縮された表スペースについての制約および、非活動化された登録と、これに対応するアプライ・プログラムのフル・リフレッシュについての詳細は、「レプリケーション環境の保守」の章を参照してください。

ASN0013E CAPTURE *capture_schema*. The Capture program required a column that was not defined in the change data (CD) table. The table name is *table_name*.

説明: 変更データ表に必須の列が定義されていません。

ユーザーの処置: 変更データ表の定義が正しいことを確認してください。詳細は、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照してください。

ASN0019E CAPTURE *capture_schema*. The Capture program libraries are not authorized for the Authorized Program Facility (APF).

説明: キャプチャー・プログラムを開始できません。

ユーザーの処置: キャプチャー・リンク・ライブラリーを APF 用に許可し、プログラムを再始動します。

ASN0020I CAPTURE *capture_schema*. Netview Generic Alerts Interface failure. The Netview return code is *return_code*.

説明: プログラム・インターフェースが失敗したため、プログラムにより NMVT (ネットワーク管理ベクトル・トランスポート) を Netview に送信できません。これは 2 次の通知メッセージです。

ユーザーの処置: インターフェース・エラーを判別するには、Netview プログラミング資料の戻りコードの記述を参照してください。エラーが訂正されるまで、SSCP (システム・サービス・コントロール・ポイント) はキャプチャー・プログラムのアラートを受け取りません。

ASN0021I CAPTURE *capture_schema*. Netview Program to Program Interface unavailable. The Netview return code is *return_code*.

説明: Netview が使用できません。これは 2 次の通知メッセージです。

ユーザーの処置: Netview の問題を判別するには、Netview プログラミング資料の戻りコードの記述を参照してください。たとえば、サブシステムが開始されていない場合があります。

ASN0023I CAPTURE *capture_schema*. The Capture program has been reinitialized and is capturing changes for *number* registrations. Stopped *number* registrations are in a stopped state. Inactive *number* registrations are in an inactive state.

説明: キャプチャー・プログラムに REINIT コマンドが実行されました。これにより、キャプチャー・プログラムは、すべての登録の内部コントロール情報をリフレッシュしようとしてしました。

ユーザーの処置: キャプチャー・プログラムがすべての登録の変更をキャプチャーしている場合は、アクションは必要ありません。そうでない場合は、先行するエラー・メッセージを見て失敗の原因を調べ、提案されているユーザー処置にしたがって、失敗した登録定義を修復してください。登録を修復した後、再度キャプチャー・プログラムに REINIT コマンドを実行します。

ASN0028I CAPTURE *capture_schema*. The Capture program is suspended by an operator command.

説明: オペレーター・コマンドによりキャプチャー・プログラムが中断され、プログラムは待ち状態になりました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0029I CAPTURE *capture_schema*. The Capture program is resumed by an operator command.

説明: オペレーター・コマンドにより、キャプチャー・プログラムは中断状態から再開され、キャプチャー・プログラムは実行を継続しています。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0031E CAPTURE *capture_schema*. The program parameter table IBMSNAP_CAPPARMS can have only one row.

説明: キャプチャー・プログラムのパラメーター表が正しく定義されていない、または有効でない行を使用して更新されています。

ユーザーの処置: IBMSNAP_CAPPARMS 表にはただ 1 つの行しかないことを確認してください。追加情報については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照してください。

ASN0035E CAPTURE *capture_schema*. A row with an unsupported architecture level was found in the table IBMSNAP_REGISTER. The row is not valid and specifies CD table *cd_owner.cd_table*, and the architecture level is *arch_level*.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録定義に無効なアーキテクチャー・レベルが含まれています。バージョン 8 のキャプチャー・プログラムは、アーキテクチャー・レベル 0801 の登録だけを使用することができます。この登録は無効であり、これは通常、レプリケーション・センターを使用せずにこの登録を生成した場合に起こります。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に出されます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: 「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照し、登録表の ARCH_LEVEL 列に必要な値をチェックしてください。ソース・サーバー側の登録表内の値が正しいことを検査します。値が正しくない場合、登録のアーキテクチャー・レベルの値を、レプリケーション・センターおよびキャプチャー・プログラムの互換性のあるバージョンを使用するように更新してください。

ASN0049I CAPTURE *capture_schema*. A row for the SIGNAL_SUBTYPE CAPSTOP was inserted into the table IBMSNAP_SIGNAL.

説明: キャプチャー・プログラムは、データのキャプチャーを停止するシグナルを受け取りました。キャプチャー・プログラムは進行中の現行作業をコミットして、終了します。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0055E **CAPTURE** *capture_schema*. The Capture program encountered a column *column_name* with an unsupported **SQLTYPE** in table *table_name*.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録定義にサポートされない SQL タイプ (抽象タイプなど) が含まれています。この登録は無効であり、これは通常、提供された管理ツールを使用せずにこの登録を生成した場合に起こります。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に出されます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: この登録を除去します。DB2 レプリケーションはこれをサポートできません。

ASN0057E **CAPTURE** *capture_schema*. The Capture program encountered error *errno* on operation for file *filename*.

説明: キャプチャー・プログラムがファイルの処理中にエラーが起きました。キャプチャー・プログラムは終了します。

ユーザーの処置: 必要なすべてのパスおよびファイルに対して、キャプチャー・プログラムが正しいアクセスおよびセキュリティ許可を持っていることを確認してください。また、システムに適切なスペースが使用可能であることを確認してください。このメッセージが製品の障害により出されたと思われる場合は、IBM ソフトウェア・サポートに連絡してください。

ASN0058W **CAPTURE** *capture_schema*. The **MAP_ID** *mapid* in a **CAPSTART** row in the **IBMSNAP_SIGNAL** table does not correspond to any entry in the **IBMSNAP_PRUNCNTL** table.

説明: CAPSTART シグナルに指定された MAP_ID の値が、IBMSNAP_PRUNCNTL 表の MAP_ID 列の現行値のどれとも一致しません。サブスクリプション・セットが削除されたか、またはユーザーが CAPSTART シグナルを誤って挿入した可能性があります。

ユーザーの処置: ユーザーがこの CAPSTART がユーザーを実行した場合は、シグナル表の MAP_ID が正しく挿入されていることをチェックし、再度試みます。アプライ・プログラムがこの CAPSTART シグナルを実行した場合は、サブスクリプション・セットがまだ存在することを検査します。

ASN0059W **CAPTURE** *capture_schema*. The **SYNCHPOINT** field in the **IBMSNAP_PRUNCNTL** table is not zeros for the **CAPSTART** of subscription with **MAP_ID** *map_id*.

説明: アプライ・プログラムはキャプチャー・プログラムにフル・リフレッシュのシグナルを出す時、IBMSNAP_SIGNAL 表に CAPSTART シグナルの行を挿入します。同時に、IBMSNAP_PRUNCNTL 表の SYNCHPOINT 列は 16 進のゼロにセットされます。するとキャプチャー・プログラムは、キャプチャー・プログラムが CAPSTART シグナルを受け取ったことをアプライ・プログラムに確認するため、次のようにして応答します。キャプチャー・プログラムは IBMSNAP_PRUNCNTL 表の SYNCHPOINT 列に、CAPSTART ログ・レコードに対応するログ・シーケンスの番号をセットします。アプライ・プログラムは SYNCHPOINT 列の値を 16 進のゼロにセットしたので、キャプチャー・プログ

ラムがゼロ以外の値を挿入したことをチェックします。キャプチャー・プログラムは、値が 16 進のゼロでなかった場合にも、SYNCHPOINT 値を更新します。ただし、SYNCHPOINT 値が 16 進ゼロでない場合には、キャプチャー・プログラムは検出した値が期待したものでなかったという、この警告を出します。

この警告は、APPLY CAPSTART シグナルを自分で出し、アプライ・プログラムのアクションを完全にシミュレートしていない場合に起こります。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0060E CAPTURE *capture_schema*. The Capture program encountered an internal error *error_code*.

説明: キャプチャー・プログラムで予期しないエラーが起きました。キャプチャー・プログラムは終了します。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN0061E CAPTURE *capture_schema*. A registration that is not valid has been found. Source table *source_owner.source_table* does not exist in the system catalog tables.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録に指定されたソース表がソース・システム・カタログにありません。このメッセージは、キャプチャー・プログラムのウォーム・スタート時または、無効な登録を含むサブスクリプション・セットにアプライ・プログラムが最初の CAPSTART シグナルを出した時に、出されます。このエラーによる、キャプチャー・プログラムの終了は起こりません。キャプチャー・コントロール表 IBMSNAP_REGISTER 内の列 *source_owner* および *source_table* の値が誤って指定されているか、またはソース表がドロップ

されており、もう存在しません。

ユーザーの処置: 登録がエラーの場合は、列 *source_owner* および *source_table* の値を訂正します。ソース表がもう存在しない場合は、その登録はもう無効なので、除去することができます。

ASN0062E CAPTURE *capture_schema*. A registration that is not valid has been found. CD table *phys_change_owner.phys_change_table* does not exist in the system catalog tables.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録に指定された CD 表がソース・システム・カタログ表にありません。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に出されます。このエラーによる、キャプチャー・プログラムの終了は起こりません。キャプチャー・コントロール表

IBMSNAP_REGISTER 内の列 *phys_change_owner* および *phys_change_table* の値が誤って指定されているか、または CD 表がドロップされ、すでに存在していません。

ユーザーの処置: 登録がエラーの場合は、列 *phys_change_owner* および *phys_change_table* の値を訂正します。CD 表がもう存在しない場合は、その登録はもう無効であり、除去することができます。

ASN0063E CAPTURE *capture_schema*. The source table *source_owner.source_table* associated with the subscription having MAP_ID *mapid* does not exist in the system catalog tables. The Capture program could not start capturing changes for this subscription.

説明: キャプチャー・プログラムは、サブスクリプションの CAPSTART シグナルに応答しようとしたが、サブスクリプションに対応するソース表がソース・システム・カタログ表にありません。このエラー・メッセージは、無効なサブスクリプションの最初の CAPSTART シグナルを受け取った時に発生します。このエラーによる、キャプチャー・プログラムの終了は起こりません。キャプチャー・コントロール表

IBMSNAP_PRUNCNTL 内の列 *source_owner* および *source_table* の値が誤って指定されているか、またはソース表がドロップされており、もう存在しません。

ユーザーの処置: サブスクリプションがエラーの場合は、列 *source_owner* および *source_table* の値を訂正します。ソース表がもう存在しない場合は、そのサブスクリプションはもう無効なので、除去することができます。

ASN0064E CAPTURE *capture_schema*. The registration is not valid for an associated subscription having MAP_ID *mapid*. The Capture program cannot start capturing change data for this subscription.

説明: キャプチャー・プログラムは、特定のサブスクリプションに関連する登録を初期化しようとしたが、登録に 1 つまたは複数の無効な列が含まれています。このエラー・メッセージは、この登録に対してサブスクリプションの最初の

CAPSTART シグナルを受け取った時に発生します。このエラーによる、キャプチャー・プログラムの終了は起こりません。キャプチャー・コントロール表 IBMSNAP_REGISTER 内の列 *phys_change_owner* および *phys_change_table* の値が誤って指定されているか、または CD 表がドロップされ、すでに存在していません。

ユーザーの処置: 登録がエラーの場合は、列 *phys_change_owner* および *phys_change_table* の値を訂正します。その登録をもう必要としない場合は、それを除去することができます。

ASN0065E CAPTURE *capture_schema*. A registration that is not valid has been found. The source table *source_owner.source_table* is not a local physical table.

説明: キャプチャー・プログラムは登録を初期化しようとしたが、登録のソース表はローカルの物理表ではなく、ソースとして使用され、ニックネームで呼ばれる、DB2 以外のリレーショナル・サーバー上にあります。DB2 以外のリレーショナル・サーバーをソースとして使用する場合、各サーバーからのデータは、トリガー・プログラムによりキャプチャーされます。各ソース表は、DB2 以外のリレーショナル・サーバー上に作成された、それ自身の登録表に存在する必要があります。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に発生します。このエラーによる、キャプチャー・プログラムの終了は起こりません。登録が、DB2 データベースの IBMSNAP_REGISTER 表に誤って作成されています。

ユーザーの処置: DB2 以外のリレーショナル・サーバー側で、この登録を再作成し、有効にする必要があります。

ASN0066E CAPTURE *capture_schema*. A registration that is not valid has been found. The CD table *phys_change_owner.phys_change_table* is not a local physical table.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、サブスクリプションに対応する登録の CD 表は、ソースとして使用される、DB2 以外のリレーショナル・データベースのニックネームです。このエラー・メッセージは、この登録に対してサブスクリプションの最初の CAPSTART シグナルを受け取った時に発生します。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: DB2 以外のリレーショナル・ソースは、トリガー・プログラムを使用してキャプチャーされ、DB2 以外のリレーショナル・ソース・システム内の、それ自身の登録表内に存在する必要があります。このようなソースの CCD 表も、DB2 以外のリレーショナル・ソース・システム内に作成されます。DB2 以外のソース表の登録が、DB2 データベース内の登録表に誤って登録されています。DB2 以外の正しいリレーショナル・サーバー側で、この登録を再作成する必要があります。

ASN0067E CAPTURE *capture_schema*. The view registration associated with the subscription having MAP_ID *map_id* was not found in the table IBMSNAP_REGISTER. The Capture program could not start capturing change data for this subscription.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、サブスクリプションに対応するビュー登録が存在しません。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った

時に発生します。このエラーによる、キャプチャー・プログラムの終了は起こりません。

IBMSNAP_REGISTER または IBMSNAP_PRUNCNTL キャプチャー・コントロール表内の列 *source_owner*、*source_table*、および *source_view_qual* の値が、誤って指定されている可能性があります。つまり、一致するものがないか、または登録がドロップされており、もう存在していません。

ユーザーの処置: サブスクリプションまたは登録がエラーの場合は、列 *source_owner*、*source_table*、および *source_view_qual* の値を訂正します。登録がもう存在しない場合は、サブスクリプションはもう無効なので、除去することができます。

ASN0068E CAPTURE *capture_schema*. The insert statement is too long for CD table *phys_chg_owner.phys_chg_tbl*.

説明: CD 表内の列の数が多すぎます。SQL INSERT ステートメントが、32K のキャプチャー・コーディング制限を超えます。

ユーザーの処置: 登録に表のすべての列が定義されているが、ターゲットではそれらの列のサブセットだけが必要な場合には、その登録の列の数を減らします。あるいは、表を 2 つの登録に分割し、各登録が表の列の異なるサブセットを持つようにします。

ASN0069E CAPTURE *capture_schema*. SQLCODE *sqlcode* was returned during an insert into the CD table *phys_chg_owner.phys_chg_tbl*. The CD table appears to have been dropped.

説明: キャプチャー・プログラムが CD 表に行を挿入しようとしたところ、CD 表はもう存在しないことを示す SQLCODE を DB2 が戻しました。CD 表が不注意でドロップされたか、または

登録全体がドロップされた可能性があります。この CD 表を参照する行が IBMSNAP_REGISTER 表にまだある場合、キャプチャー・プログラムは、CD_OLD_SYNCHPOINT 列の値を NULL にしてこの登録を非活動化し、この CD 表の変更をもうキャプチャーしないようにします。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: CD 表がもう存在せず、もう必要ない場合は、その登録はもう無効なので除去してください。登録を除去する前に、登録を非活動化することをお勧めします。この CD 表を使用する登録に関連するサブスクリプション・セットもすべて、非活動化すべきです。さらに、関連するサブスクリプション・セット・メンバーも除去し、これらのサブスクリプション・セットを活動化し、正常に実行できるようにします。

ASN0070E CAPTURE *capture_schema*. **The combination of column name *column_name* in the CD table *phys_chg_owner.phys_chg_tbl*, and the value of BEFORE_IMG_PREFIX *before_img_prefix* in the table IBMSNAP_REGISTER for this registration matches multiple column names in the source table. The ambiguity in the registration definition cannot be resolved by the Capture program.**

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録用の CD 表内の列が未確定です。列は、1 つのソース列の変更前イメージまたは、別のソース列の変更後イメージ、のどちらかを参照することができます。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に送られます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ャー・プログラムの終了は起こりません。

ユーザーの処置: IBMSNAP_REGISTER 表の、BEFORE_IMG_PREFIX 列の現行の値を、このような未確定さをもたらさない文字値に変更します。

ASN0071E CAPTURE *capture_schema*. **The data type attribute of the column *column_name* in the CD table *phys_chg_owner.phys_chg_tbl* is not compatible with the data type attribute of the corresponding source column.**

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録の CD 表の列が対応するソース列と互換性がありません。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に送られます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: この登録の CD 表を訂正します。

ASN0072E CAPTURE *capture_schema*. **The before-image column *column_name* in the CD table *phys_chg_owner.phys_chg_tbl* must allow NULL values.**

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録の CD 表の変更前イメージ列が、NULL 値を受け付けるように定義されていません。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に送られます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: この登録の CD 表を訂正します。

ASN0073E CAPTURE *capture_schema*. The specification *input_in* describing the CD table on a CAPSTOP signal is not valid.

説明: キャプチャー・プログラムは、CAPSTOP シグナルに指定された INPUT_IN 値が、*phys_change_owner.phys_change_table* の有効な形式ではないことを検出しました。このエラーによる、キャプチャー・プログラムの終了は起こらず、このシグナルにはアクションは取られません。

ユーザーの処置: INPUT_IN の値が、非活動化したい登録に関連付けられた CD 表の名前と一致することを確認してください。IBMSNAP_SIGNAL 表に新しい行を挿入します。

ASN0074E CAPTURE *capture_schema*. There is no row in the IBMSNAP_REGISTER table that corresponds to *src_owner.src_table* that is specified on a CAPSTOP signal.

説明: キャプチャー・プログラムは、CAPSTOP シグナルに指定された INPUT_IN 値が有効な形式ではなく、登録表の *source_owner.source_table* の値と一致しないことを検出しました。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: INPUT_IN の値を訂正し、再度シグナルを挿入します。

ASN0075W CAPTURE *capture_schema*. The registration corresponding to the INPUT_IN, *src_owner.src_table* on a CAPSTOP signal was not capturing changes. No action is taken.

説明: キャプチャー・プログラムは、CAPSTOP シグナルに指定された INPUT_IN 値が有効な形

式であり、登録表の *source_owner.source_table* の値と一致するが、この登録がすでに活動状態にないことを検出しました。このエラーによる、キャプチャー・プログラムの終了は起こらず、キャプチャー・プログラムはこのシグナルに何もアクションをしません。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0076I CAPTURE *capture_schema*. Capture has stopped capturing changes for source table *source_owner.source_table* in response to a CAPSTOP signal.

説明: キャプチャー・プログラムは、CAPSTOP シグナルに指定された登録を正常に非活動化しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0077E CAPTURE *capture_schema*. The values specified for the PHYS_CHANGE_OWNER and PHYS_CHANGE_TABLE columns in the IBMSNAP_PRUNCNTL table where MAP_ID = *mapid* are NULL or do not match a valid row in the IBMSNAP_REGISTER table.

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、サブスクリプションの IBMSNAP_PRUNCNTL 表の列 PHYS_CHANGE_OWNER および PHYS_CHANGE_TABLE の値が、IBMSNAP_REGISTER 表の登録行と一致しません。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションに最初の CAPSTART シグナルを受け取った時に出されます。このエラー

による、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: このサブスクリプションの IBMSNAP_PRUNCNTL 表の値を訂正します。レプリケーション・センターを使用してこのサブスクリプションを作成した場合は、潜在する管理上の問題について、IBM ソフトウェア・サポートに連絡します。

ASN0078E CAPTURE *capture_schema*. **The before-image column** *column_name* **in the CD table** *phys_owner.phys_table* **has no corresponding after-image column in the CD table for this registration. The registration is not valid.**

説明: キャプチャー・プログラムは登録を初期化しようとしたのですが、登録の CD 表の変更前イメージ列と対応する変更後イメージ列がありません。この登録は無効であり、これは通常、提供された管理ツールを使用せずにこの登録を生成した場合にのみ起こります。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションの最初の CAPSTART シグナルを受け取った時に、出されます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: この登録の CD 表について、CD 表に変更前イメージ列を含める場合には、対応する変更後イメージ列も必ず含まれていることを確認し、訂正します。

ASN0079E CAPTURE *capture_schema*. **SQLCODE** *sqlcode* **was returned during an update to the IBMSNAP_REGISTER table for the registrations associated with the CD table** *phys_chg_owner.phys_chg_tbl*. **The rows might have been deleted.**

説明: キャプチャー・プログラムが、指定された CD 表のデータをキャプチャーしたことを示すために IBMSNAP_REGISTER 表を更新しようとしたところ、その行はもう存在しないことを示す SQLCODE を DB2 が戻しました。登録がドロップされた可能性があります。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: 登録がドロップされた場合、その登録にはこれ以上のアクションは必要ありません。登録をドロップする場合は、まずその登録を非活動化することをお勧めします。

IBMSNAP_REGISTER 表の行を不注意でドロップした場合には、関連する CD 表をドロップし、登録を再作成します。登録に関連するすべてのサブスクリプション・セットを非活動化します。登録をドロップする必要がある場合は、関連するサブスクリプション・セット・メンバーをドロップし、これらのサブスクリプション・セットを活動化し、正常に実行できるようにします。登録が再作成されると、アプライ・プログラムにシグナルが送信され、関連するサブスクリプション・セットにフル・リフレッシュを実行すべきことを知らせます。

ASN0080E CAPTURE *capture_schema*. **A table space full condition has been encountered for CD table** *phys_chg_owner.phys_chg_tbl*, **which is associated with the registration for source table** *source_owner.source_table*.

説明: キャプチャー・プログラムは指定された CD 表への挿入を処理しようとしたのですが、表スペースがいっぱいという条件が起こり、挿入を処理できませんでした。通常、この条件は、CD 表スペースのスペース割り振りが不十分であるか、プルーニングの頻度が少ないか、またはプルーニングが効果的でないことが原因です。このエラーにより、キャプチャー・プログラムは終了します。

ユーザーの処置: 次のステップで、表スペースがいっぱいという条件の原因を調べます。

1. 通常の処理条件を満たすだけの、十分なスペースがこの CD 表の表スペースに割り振られていることを確認します。
2. キャプチャー・コントロール表のストレージ必要量を減らすため、プルーニングを十分な頻度で実行していることを確認します。
3. 通常のプルーニング処理を行なうのに十分な頻度でアプライ・プログラムを実行していることを確認します。
4. サブスクリプション・セットを長期間、非活性化した結果、通常のプルーニングに追加のステップが必要になる、といったことがないことを検査します。

追加情報については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」を参照してください。

ASN0082W CAPTURE *capture_schema*. **The Capture program encountered a registration with a column** *column_name* **in the CD table** *phys_chg_owner.phys_chg_tbl* **with a column length** *CD_column_length* **that is shorter than the length of the corresponding column in the source table** *source_owner.source_table*, **with a length of** *src_column_length*.

説明: 登録の初期化中に、キャプチャー・プログラムは、登録の定義に、CD 表の列が対応するソース表の列の長さより短いものが含まれていることを検出しました。この登録定義は避けたいものであり、通常、提供された管理ツールを使用せずにこの登録を生成、または変更した場合にのみ起こります。この登録定義は許されますが、警告メッセージが表示され、キャプチャーされたソース表のデータは定義された CD 表の列に適合しない可能性があることが通知されます。このメッセージは、キャプチャー・プログラムのウォーム・スタート中または、この登録に対するサブスクリプションの最初の CAPSTART シグナルを受け取った時に、出されます。登録は正常に初期化されます。

ユーザーの処置: このような方法で登録を定義する特別な理由がある場合 (たとえば、変更されたデータの長さが CD 表の列の長さ以下であるという確信がある場合) を除き、登録はソース表と CD 表のデータ定義が正確に一致するように定義すべきです。

ASN0083E CAPTURE *capture_schema*.
SQLCODE *sqlcode* was returned
when trying to process an
insert into the CD table
phys_chg_owner.phys_chg_tbl. The
CD table column *column_name* is
too short and cannot contain
the captured data from the
corresponding column in the
source table,
source_owner.source_table. The
registration has been stopped
by the Capture program.

説明: キャプチャーは、CD 表への挿入を処理しようとしたが、ソース表の対応する列の長さより短い列が CD 表に含まれることを示す SQLCODE が DB2 から戻されました。この登録定義は避けたいものであり、通常、提供された管理ツールを使用せずにこの登録を生成、または変更した場合にのみ起こります。このエラーによる、キャプチャー・プログラムの終了は起こりませんが、登録は停止状態になります。

ユーザーの処置: この登録定義を再度、評価します。登録を変更して、ソース表の列の長さと CD 表の列の長さが一致するようにするか、またはデータを切り捨てるトリガーを CD 表に追加します。

ASN0084E CAPTURE *capture_schema*. The
registration with the source
table *source_owner.source_table*
and the CD table
phys_chg_owner.phys_chg_tbl has
been stopped by the Capture
program.

説明: このエラー・メッセージは、キャプチャー・プログラムにより登録が停止状態 (IBMSNAP_REGISTER 表の STATE 列に値 'S' がセットされる) に置かれた場合には常に出されます。このアクションの理由は、1 つまたは複数の先行するメッセージに記述されています。

ユーザーの処置: 先行するエラー・メッセージを見て失敗の原因を調べ、提案されているユーザー処置にしたがって、失敗した登録定義を修正してください。登録定義を修正した後、IBMSNAP_REGISTER 表の STATE 列の値を手作業で 'I' にセットし、アプライ・プログラムが登録を再度使用できることを示す必要があります。

ASN0100I CAPTURE *capture_schema*. The
Capture program initialization is
successful.

説明: このメッセージは単なる通知です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0101W CAPTURE *capture_schema*. The
Capture program warm start
failed because existing data is
too old. A cold start will be
attempted.

説明: 変更データ表のデータは古すぎます。コールド・スタートが実行されます。

ユーザーの処置: 「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料のキャプチャー・プログラム操作に関する記述を参照してください。

ASN0102W CAPTURE *capture_schema*. The
Capture program switches to
cold start because the warm
start information is insufficient.

説明: 再始動情報の検索中に問題が起こりました。再始動表のデータが無効です。コールド・スタートが実行されます。

- DB2 Universal Database の場合、ウォーム・スタート中に DB2 がログを読み取っている時に、非同期読み取りログ API エラーが起こりました。

- z/OS の場合、ウォーム・スタート中に DB2 がログを読み取っている時に、Instrumentation Facility Information (IFI) エラーが起きました。

ユーザーの処置: 「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料のキャプチャー・プログラム操作に関する記述を参照してください。

ASN0104I CAPTURE *capture_schema*. In response to a CAPSTART signal with MAP_ID *mapid*, change capture has started for the source table *source_owner.source_table* for changes found on the log beginning with log sequence number *log_sequence_number*.

説明: キャプチャー・プログラムは CAPSTART シグナルを正常に処理しました。これが、特定のソース表に関連する最初の CAPSTART シグナルの場合、このメッセージは、キャプチャー・プログラムが現在、ソース表への更新をキャプチャーしている最中であることを示します。これが、変更をすでにキャプチャーした表に対する CAPSTART シグナルである場合、このメッセージは、キャプチャー・プログラムがシグナルを受け取り、入力 MAP_ID 値に関連付けられたサブスクリプション・セットの変更の受け取りを、アプライ・プログラムが開始できるようにするための処理を実行したことを示します。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0105I CAPTURE *capture_schema*. n rows have been pruned from the table *table_owner.table_name* at *timestamp*.

説明: キャプチャー・プログラムは、CD 表、UOW 表、TRACE 表、MONITOR 表、または

SIGNAL 表からレコードのブルーニングをしました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0109I CAPTURE *capture_schema*. The Capture program has successfully initialized and is capturing data changes for *number* registrations. *Stopped_number* registrations are in a stopped state. *Inactive_number* registrations are in an inactive state.

説明: このメッセージは、キャプチャー・プログラムが登録エントリーの再初期化を完了した時に出されます。再初期化は、ウォーム・スタート中、CAPSTART シグナルの処理中、またはキャプチャー REINIT コマンドへの応答内で行なわれます。

ユーザーの処置: キャプチャー・プログラムがすべての登録の変更をキャプチャーしている場合は、アクションは必要ありません。そうでない場合は、先行するエラー・メッセージを見て失敗の原因を調べ、提案されているユーザー処置にしたがって、失敗した登録定義を修復してください。登録定義を修復した後、`reinit` パラメーターを指定して `asnccmd` コマンドを実行します。

ASN0111I CAPTURE *capture_schema*. The pruning cycle started at *timestamp*.

説明: このメッセージは、各ブルーニング・サイクルの始めに出されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0112I CAPTURE *capture_schema*. The pruning cycle ended at *timestamp*.

説明: このメッセージは、各プルーフリング・サイクルの終わりに出されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0113W CAPTURE *capture_schema*. The pruning retention limit will be reached in the next 24 hours.

説明: この警告は、IBMSNAP_UOW 表の値が、翌日に保存期限のプルーフリングが行なわれることを示す場合に出されます。

ユーザーの処置: 通常のプルーフリングが行なわれない理由を調査してください。通常、この理由は、1 つまたは複数のアプライ・プログラムが長期間実行されておらず、したがって CD 表および UOW 表が効果的にプルーフリングできていないためです。考えられる別の要因は、IBMSNAP_PRUNE_SET 表の対応する同期点値を除去またはリセットせずに、サブスクリプション・セットを除去または非活動化した場合です。レプリケーション・アナライザー・ツールを使用すれば、この状況の詳細な分析が提供されます。

ASN0114E CAPTURE *capture_schema*. Pruning has failed with SQL code *sqlcode* when pruning the table *table_owner.table_name*.

説明: このエラー・メッセージは、予期しない SQL コードによりプルーフリングが失敗した場合に出されます。プルーフリングは終了し、次のインターバルまたはコマンド呼び出しで、再度試みられます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: この SQL コードが一時的エラーの場合は、アクションは必要ありません。そうでない場合は、「DB2 メッセージおよびコード」

資料の、SQL エラーについて示されたアクションを取ります。

ASN0121E CAPTURE *capture_schema*. The Capture program warm start failed because existing data is too old. The Capture program will terminate.

説明: ウォーム・スタート情報の時刻が、遅れの限界を超えました。

ユーザーの処置: 応答は必要ありません。キャプチャー・プログラムは終了します。

ASN0122E CAPTURE *capture_schema*. An error occurred while reading the restart information or DB2 log. The Capture program will terminate.

説明: 再始動情報の検索中に問題が起きました。再始動表のデータが無効であるか、または z/OS の場合、再始動中にログを読み取っている時に、Instrumentation Facility Interface (IFI) エラーが起きました。エラーが解決した後、ウォーム・スタート・オプションを使用して再始動することができます。

ユーザーの処置: 応答は必要ありません。キャプチャー・プログラムは終了します。

ASN0123I CAPTURE *capture_schema*. At program termination, the highest log sequence number of a successfully captured log record is *max_commitseq* and the lowest log sequence number of a record still to be committed is *min_inflightseq*.

説明: キャプチャー・プログラムは終了し、監査目的用に、その時の再始動表の値を記録します。

ユーザーの処置: このメッセージは通知のみであ

り、アクションは必要ありません。

ASN0133I CAPTURE *capture_schema*. The Capture program has reached the end of the active log and will terminate because the AUTOSTOP feature is specified.

説明: キャプチャー・プログラムは、ユーザー・オプション AUTOSTOP で指定されたように、アクティブ・ログの終わりに達したので終了しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0142E CAPTURE *capture_schema*. The Capture program is unable to perform an insert operation on the monitor table IBMSNAP_CAPMON. The SQL code is *sqlcode*. The monitoring information for this interval will be skipped.

説明: このエラー・メッセージは、予期しない SQL コード によりモニター・スレッドが失敗した場合に出されます。このインターバルのモニター機能はスキップされ、プログラムは次のインターバルに再度試みます。このエラーによる、キャプチャー・プログラムの終了は起こりません。

ユーザーの処置: この SQL コードが一時的エラーの場合は、アクションは必要ありません。そうでない場合は、「DB2 メッセージおよびコード」資料の、SQL エラーについて示されたアクションを取ります。

ASN0143W CAPTURE *capture_schema*. The program detected that the source database *src_db_name* has been restored or rolled forward. The Capture program has switched from a warm start to a cold start.

説明: キャプチャー・プログラムは、startmode として warmsa または warmsi を使用して開始しました。キャプチャー・プログラムがウォーム・スタートを試みた時、DB2 ログ読み取り API から、ソース・データベースがリストアまたはロール・フォワードされたこと、およびログ・シーケンス番号が再利用されたことを示す戻りコードを受け取りました。ソース・データベースの状態とキャプチャーされたデータの状態は、もはや整合性が取れていません。キャプチャー・プログラムはコールド・スタートに切り替わりました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0144E CAPTURE *capture_schema*. The program detected that the source database *src_db_name* has been restored or rolled forward. A cold start is recommended to restore consistency.

説明: キャプチャー・プログラムは、開始モード warmns または warmsi を使用して開始しました。キャプチャー・プログラムがウォーム・スタートを試みた時、DB2 ログ読み取り API から、ソース・データベースがリストアまたはロール・フォワードされたこと、およびログ・シーケンス番号が再利用されたことを示す戻りコードを受け取りました。ソース・データベースの状態とキャプチャーされたデータの状態は、もはや整合性が取れていません。キャプチャー・プログラムは終了し、自動的にコールド・スタートに切り替えることはしません。

ユーザーの処置: キャプチャー・プログラムのウォーム・スタートを実行しても安全だという確信がある場合は、キャプチャー・プログラムを再始動します。これは 2 回目の試みでは終了しません。キャプチャー・プログラムのウォーム・スタートの後で、キャプチャーしたデータが整合性のある状態かどうか確かでない場合は、キャプチャー・プログラムのコールド・スタートを実行することをお勧めします。

ASN0180W CAPTURE *capture_schema*. The table **IBMSNAP_SIGNAL** is not an EBCDIC table as required by capture. The signal has been processed.

説明: キャプチャー・プログラムは、IBMSNAP_SIGNAL 表が EBCDIC 表として定義されていないことを検出しました。適切に処理するには、シグナルを EBCDIC に変換する追加の処理が必要です。この追加処理は、わずかなパフォーマンス低下をもたらします。

ユーザーの処置: すぐにできる簡単な方法として、次のステップを実行します。

1. キャプチャー・プログラムを停止してください。
2. IBMSNAP_SIGNAL 表をドロップし、EBCDIC エンコード方式を使用して再作成します。
3. キャプチャー・プログラムを再始動します。

ASN0181W CAPTURE *capture_schema*. The row for the signal with timestamp *signal_time* no longer exists in the **IBMSNAP_SIGNAL** table. The signal has been processed.

説明: キャプチャー・プログラムはシグナルからの要求を処理しましたが、**SIGNAL_STATE** および **SIGNAL_LSN** を更新できませんでした。したがって、シグナルの発行者は、キャプチャー・プ

ログラムがシグナルを受け取ったことを判断できません。

ユーザーの処置: キャプチャー・プログラムからのシグナルの更新を、別の処理が待っているかどうかを調べ、必要ならばシグナルを再送します。

ASN0182W CAPTURE *capture_schema*. The row for signal with timestamp *signal_time* no longer exists in **IBMSNAP_SIGNAL** table and the table is not EBCDIC. The signal will be ignored by capture.

説明: キャプチャー・プログラムが EBCDIC でエンコードされていないシグナルを受け取ったため、初期化障害が起きました。

IBMSNAP_SIGNAL 表の行はもはや存在しないため、キャプチャー・プログラムはシグナルを EBCDIC に変換できませんでした。キャプチャー・プログラムは何のシグナルが送られたのかわからないので、このシグナルを無視します。

ユーザーの処置: 何のシグナルが送られたのかを判別し、シグナルを再送します。

すぐにできる簡単な方法として、次のステップを実行します。

1. キャプチャー・プログラムを停止してください。
2. IBMSNAP_SIGNAL 表をドロップし、EBCDIC エンコード方式を使用して再作成します。
3. キャプチャー・プログラムを再始動します。

ASN0500E *pgmname* : *program_qualifier* : The parameter **input** *input_value* supplied for parameter name *parameter_name* is not valid.

説明: 無効な入力パラメーターを指定して、プログラムまたはコマンド・プログラムが呼び出されました。メッセージに、エラーとなったプログラムの名前が、パラメーター名およびパラメーター値と一緒に示されています。

ユーザーの処置: 有効な呼び出しパラメーターを資料で調べ、入力を訂正し、タスクまたはコマンドを再サブミットします。

ASN0501E *pgmname : program_qualifier : The value input_value supplied for the parameter parameter_name is not the correct data type.*

説明: 関連するデータ・タイプが無効な入力値を使用して、プログラムまたはコマンド・プログラムが呼び出されました。メッセージに、エラーとなったプログラムの名前、誤りの入力値、およびこの入力値を指定したパラメーターの名前が示されています。

ユーザーの処置: パラメーター入力として正しいデータ・タイプを含めるように呼び出しを訂正し、再サブミットします。

ASN0502E *pgmname : program_qualifier : The value input_value of length invalid_string_length, supplied for parameter parameter_name, is greater than the maximum allowed string length of allowed_string_length.*

説明: 無効なストリング長を持つ入力値を使用して、プログラムまたはコマンド・プログラムが呼び出されました。メッセージに、エラーとなったプログラムの名前、誤りの入力値、およびこの入力値を指定したパラメーターが示されています。

ユーザーの処置: パラメーター入力に正しいストリング長を含めるように呼び出しを訂正し、再サブミットします。

ASN0503E *pgmname : program_qualifier : The integer value input_value, supplied for parameter parameter_name, is outside the supported range for this parameter.*

説明: サポート範囲外の入力値を使用して、プログラムまたはコマンド・プログラムが呼び出されました。メッセージに、エラーとなったプログラムの名前、誤りの入力値、およびこの入力値を指定したパラメーターが示されています。

ユーザーの処置: パラメーター入力として正しい範囲の値を含めるように呼び出しを訂正し、再サブミットします。

ASN0504E *pgmname : program_qualifier : The program did not recognize the invocation parameter incorrect_input.*

説明: 無効なパラメーターまたはコマンドを指定して、プログラムまたはコマンド・プログラムが呼び出されました。メッセージに、このメッセージを出したプログラム、および認識できなかった呼び出し入力が見示されています。

ユーザーの処置: 有効な入力パラメーターを資料で調べ、入力を訂正し、タスクまたはコマンドを再サブミットします。

ASN0505E *pgmname : program_qualifier : The program was unable to get or set an IPC key.*

説明: プログラムまたはコマンド・プログラムは、コマンドの処理に必要な処理間通信を初期化できませんでした。このエラーにより、失敗したプログラムは終了します。

ユーザーの処置: 失敗したプログラムまたはコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0506E *pgmname : program_qualifier : The program could not attach to the replication communications message queue.*

説明: プログラムまたはコマンド・プログラムが、ユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0507E *pgmname : program_qualifier : The program could not create the replication communications message queue.*

説明: プログラムまたはコマンド・プログラムが、ユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0508E *pgmname : program_qualifier : The program could not send a message to the replication communications message queue.*

説明: プログラムまたはコマンド・プログラムが、ユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0509E *pgmname : program_qualifier : The program could not process a received message because of an incorrect message version.*

説明: プログラムまたはコマンド・プログラムが、ユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0510E *pgmname : program_qualifier : The program encountered a timeout while waiting for reply messages.*

説明: コマンド・プログラムがユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0511E *pgmname : program_qualifier : The program was unable to process the received message because of an unknown message function.*

説明: プログラムがユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再試行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0512E *pgmname : program_qualifier : The program could not read from its replication communications message queue.*

説明: プログラムがユーザー・コマンドの処理を試みている時に内部エラーが起きました。プログラムはこの障害により終了しませんが、コマンドは実行されません。

ユーザーの処置: 失敗したコマンドを再実行します。問題が持続するようならば、IBM ソフトウェア・サポートに連絡してください。

ASN0513E *pgmname : program_qualifier : The program could not open the message file named msg_file.*

説明: プログラムで使用されたこのメッセージ・ファイルが誤ってインストールされているか、または言語環境変数が正しくセットされていません。

ユーザーの処置: インストールおよび構成についての情報は、資料を参照してください。

ASN0514E *pgmname : program_qualifier : The program could not open the log file log_file.*

説明: プログラムが自分のプログラム・メッセージ・ログのファイルをオープンしようとした時に内部エラーが起り、この障害のために異常終了しました。この問題は、ファイルが不注意で削除されたため、またはこの処理に関連するユーザー ID が、ファイルをオープンするための十分な権限を持っていないために起こった可能性があります。

ユーザーの処置: 処理のユーザー ID に十分な権限が与えられていることを検査します。ファイルが不注意で削除された場合は、新しいログ・ファイルを作成するプログラムを再始動します。

ASN0515E *pgmname : program_qualifier : The program could not close the log file.*

説明: プログラムが自分のプログラム・メッセージ・ログに使用されたファイルをクローズしようとした時に、内部エラーが起きました。プログラムが終了する前に、ファイルが不注意で削除された可能性があります。最終の終了メッセージは出されない場合があります。

ユーザーの処置: ファイルが不注意で削除された場合は、新しいログ・ファイルを作成するプログラムを再始動します。

ASN0516E *pgmname : program_qualifier : The program could not close the message catalog.*

説明: メッセージ・カタログ・ファイルのクローズを試みている時に、プログラムで内部エラーが起きました。プログラムが終了する前に、ファイルが不注意で削除された可能性があります。最終の終了メッセージは出されない場合があります。

ユーザーの処置: メッセージ・ファイルが削除された場合は、再インストールする必要があります。

ASN0517E *pgmname : program_qualifier : The program has recovered the ability to read from its replication communications message queue.*

説明: プログラムは、前の障害の後、コマンドを処理するために必要な、メッセージ・キューからの読み取り機能を再初期化することができました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0518E *pgmname* : *program_qualifier* : The program does not accept multiple commands.

説明: 複数のコマンドを指定してコマンド・プログラムが呼び出されました。各コマンド呼び出しは、1つの入力コマンドを、他の必要なコマンド入力と一緒に指定して実行する必要があります。
注: CHGPARMS コマンドの場合、1つの CHGPARMS コマンド呼び出しで複数のパラメーターを変更することができます。

ユーザーの処置: コマンド入力を訂正し、コマンドを再サブミットしてください。

ASN0519E *pgmname* : *program_qualifier* : The parameter input *parameter_value* supplied for CHGPARMS parameter *parameter_name* is not valid.

説明: 誤りのパラメーター入力を使用して CHGPARMS コマンドが呼び出されました。

ユーザーの処置: コマンド入力を訂正し、コマンドを再サブミットしてください。

ASN0520I *pgmname* : *program_qualifier* : The STATUS command response: *thread_type* thread is in the *status_condition* state.

説明: **status** コマンドへの応答として、コマンドを受け取ったプログラムに関連するスレッドごとに、これらのメッセージの1つが出され、そのスレッドの現行状態を示します。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0521I *pgmname* : *program_qualifier* : The QRYPARMS command response: *parameter_name* was set to *parameter_value* by the following method: *method*.

説明: QRYPARMS への応答として、プログラム・パラメーターごとにメッセージが出されません。各パラメーターごとに、パラメーターの名前、パラメーターの現行の設定値、およびパラメーター値をセットするためにユーザーが使用した方法 (デフォルトにより、IBMSNAP_CAPPARMS 表を CHANGE して、始動オプションにより、または CHGPARMS コマンドを使用して) がメッセージに示されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0522I *pgmname* : *program_qualifier* : The program received the *command_type* command.

説明: プログラムは処理するコマンドを受け取りました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0523I *pgmname* : *program_qualifier* : The CHGPARMS command response: *parameter_name* has been set to *parameter_value*.

説明: CHGPARMS コマンドへの応答として、変更されたプログラム・パラメーターごとにこのメッセージの1つが出されます。各パラメーターごとに、パラメーターの新しい値がメッセージに示されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0524E *pgmname* : *program_qualifier* : **The program required parameter *parameter_name* was not specified.**

説明: 必要パラメーターの 1 つを指定せずに、プログラムまたはコマンド・プログラムが呼び出されました。欠如しているパラメーターが *capture_server* または *control_server* の場合、プログラムまたはコマンドは、(該当する場合) DB2DBDFT 環境変数の設定値を使用してデータベース名を暗黙にアクセスしてみましたが、これも成功しませんでした。

ユーザーの処置: 該当のパラメーターおよびその入力値を含めるように呼び出しを訂正してください。

ASN0525E *pgmname* : *program_qualifier* : **The program could not read from its external communications message queue.**

説明: プログラムまたはコマンド・プログラムは、コマンドの処理に必要な、外部通信メッセージ・キューからの読み取り機能を初期化できませんでした。

ユーザーの処置: 失敗したコマンドを再実行し、問題が持続する場合は、IBM ソフトウェア・サポートに連絡してください。

ASN0526E *pgmname* : *program_qualifier* : **The program was invoked without any command input.**

説明: 処理するコマンドを指定せずに、コマンド・プログラムが呼び出されました。コマンド処理は行なわれません。

ユーザーの処置: 必要な入力をすべて指定して、コマンドを再サブミットしてください。

ASN0527E *pgmname* : *program_qualifier* : **The program was invoked without any CHGPARMS command input.**

説明: CHGPARMS コマンドを使用してコマンド・プログラムが呼び出されましたが、処理するコマンド入力が指定されていません。コマンド処理は行なわれません。

ユーザーの処置: 必要な入力をすべて指定して、コマンドを再サブミットしてください。

ASN0528E *pgmname* : *program_qualifier* : **The program will terminate because the required control table *control_table_name* does not exist.**

説明: キャプチャー・プログラムまたはアプライ・プログラムは、必要なキャプチャー・コントロール表に SQL 操作を実行しようとした。プログラムは DB2 から *not found* の戻りコードを受け取りました。この戻りコードは、マイグレーションが完了していないか、または不注意により、必要なキャプチャー・コントロール表が環境からドロップされた場合に起こります。

ユーザーの処置: 欠落しているコントロール表の名前をメッセージ・テキストで見つけます。この問題を訂正する方法は、どの表が欠落しているかにより異なります。たとえば、表が

IBMSNAP_PRUNE_LOCK の場合、表は簡単に再作成でき、キャプチャー・プログラムを再始動することができます。しかし、表が

IBMSNAP_RESTART であり、表の正しい内容をリストアできない場合、表の再作成と、キャプチャー・プログラムのコールド・スタートが必要になります。

ASN0529I *pgmname* : *program_qualifier* : **The value of *parameter_name* was set to *parameter_value* at startup by the following method: *method*.**

説明: プログラムが開始され、プログラム・パラメーターは、指定された始動オプションおよび IBMSNAP_CAPPARMS 表の既存の内容を組み合わせ、初期化されました。このメッセージは監査目的で書き込まれます。パラメーターは、次の方法の 1 つを使用してセットされました。その方法とは、デフォルトにより、IBMSNAP_CAPPARMS 表の CHANGE により、または始動オプションにより、のいずれかです。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0530E *pgmname* : *program_qualifier* : **The program could not connect to database *database_name* with USERID *user_id* . The SQLCODE is *sql_code*.**

説明: プログラムが次の関数の 1 つを実行した時にエラーが起きました。

- DB2 (VSE および VM 版) への CONNECT 関数
- DB2 Call Attachment Facility (CAF) への CONNECT 関数
- DB2 UDB への暗黙接続

ユーザーの処置: ご使用のオペレーティング・システムの DB2 データベース・マネージャーの、「メッセージおよびコード」資料の DB2 コードを参照し、該当する理由コードを見つけてください。

DB2 for z/OS エラーの場合は、Call Attachment Facility を記述している管理ガイドのセクションを参照してください。疑問および診断については、DBA にご相談ください。

プログラムを DB2 UDB (UNIX 版) の下で実行している場合は、LIBPATH 環境変数が、プログ

ラムが開始された環境と同じになっていることを確認してください。追加情報については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の「レプリケーションのセットアップ」の説明を参照してください。

ASN0531E *pgmname* : *program_qualifier* : **The program could not open the plan. The SQL return code is *return_code*, the reason code is *reason_code*, the subsystem name is *DB2_subsystem*, and the plan name is *plan_name*.**

説明: プログラムがプラン ASNLPLAN をオープンしようとした時にエラーが起きました。

ユーザーの処置: ご使用のオペレーティング・システムの DB2 データベース・マネージャーの、「メッセージおよびコード」資料の DB2 コードのセクションを参照し、該当する理由コードを見つけてください。Call Attachment Facility を記述している管理ガイドのセクションを参照してください。

ASN0532E *pgmname* : *program_qualifier* : **DB2 release *release_number* is not supported.**

説明: プログラムはこのリリースの DB2 をサポートしません。

ユーザーの処置: 適切なリリースの DB2 を使用して、プログラムを実行してください。

ASN0533E *pgmname* : *program_qualifier* : **DB2 was terminated abnormally.**

説明: プログラムがまだアクティブな時に DB2 が終了しました。

z/OS、VSE/ESA、または VM/ESA の場合、プログラムがアクティブな最中に DB2 が終了し、ユーザーは NOTERM 呼び出しパラメーターを指定しませんでした。

ユーザーの処置: DB2 を開始し、プログラムを開始します。

ASN0534E *pgmname : program_qualifier : DB2 database cannot be used, because it is in the state state.*

説明: プログラムがまだアクティブな時に DB2 が終了しました。データベースは次の状態の 1 つになっています。それは、UNDETERMINED、TERMINATED、QUIESCED、ROLLWARD、または ACTIVE です。

ユーザーの処置: DB2 を開始してからプログラムを開始します。

ASN0535E *pgmname : program_qualifier : The program could not disconnect from the database db_server. The return code is return_code, and the reason code is reason_code.*

説明: DB2 へのアタッチメントを終了中に、プログラムは Call Attachment Facility (CAF) からエラー・コードを受け取りました。

ユーザーの処置: プログラムを再始動します。

ASN0536E *pgmname : program_qualifier : An error was returned while getting the instance name. The SQLCODE is sqlcode.*

説明: DB2 Universal Database の SQLEGENS API がエラーを戻しました。

ユーザーの処置: 「DB2 Universal Database API リファレンス」資料で SQLEGENS API についての情報を参照してエラーを判別するか、または IBM ソフトウェア・サポートに連絡してください。

ASN0537E *pgmname : program_qualifier : The program could not connect to database database_name, the return code is return_code, and the reason code is reason_code.*

説明: プログラムが次の関数の 1 つを実行した時にエラーが起きました。

- DB2 (VSE および VM 版) への CONNECT 関数
- DB2 Call Attachment Facility (CAF) への CONNECT 関数
- DB2 UDB への暗黙接続

ユーザーの処置: ご使用のオペレーティング・システムの DB2 データベース・マネージャーの、「メッセージおよびコード」資料の DB2 コードを参照し、該当する理由コードを見つけてください。

DB2 for z/OS エラーの場合は、Call Attachment Facility を記述している管理ガイドのセクションを参照してください。疑問および診断については、DBA にご相談ください。

プログラムを DB2 UDB (UNIX 版) の下で実行している場合は、LIBPATH 環境変数が、プログラムが開始された環境と同じになっていることを確認してください。追加情報については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の「レプリケーションのセットアップ」の説明を参照してください。

ASN0538I *pgmname : program_qualifier : The program is waiting for DB2 to come up.*

説明: プログラムを最初に立ち上げた時に、DB2 が立ち上がっていない場合、プログラムは DB2 が立ち上がるまで待ちます。DB2 が立ち上がった後、キャプチャー・プログラムは接続を行い、変更のキャプチャーを開始します。

キャプチャーの呼び出しパラメーターに NOTERM オプションが指定されており、DB2

が円滑にダウンした場合、キャプチャー・プログラムは DB2 が立ち上がるのを待ちます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0539E *pgmname* : *program_qualifier* :
Database or subsystem named
database_server_name **needs to be**
configured with
LOGRETAIN=RECOVERY.

説明: キャプチャー・プログラムはコールド・スタートまたはウォーム・スタートを実行しようとしたが、ソース・データベースが、キャプチャー・プログラムがログ読み取りインターフェースを使用できるように適切に定義されていません。データベース構成パラメーター LOGRETAIN の受け入れ可能な設定値は、RECOVERY (ON とも呼ばれる) です。

ユーザーの処置: **update database configuration for** コマンドを使用して、LOGRETAIN パラメーターの設定値を変更してください。このパラメーターを RECOVERY (または ON) にする場合は、このデータベースをアプリケーションで使用する前に、**backup** コマンドを使用してデータベースをバックアップする必要がありますことに注意してください。

ASN0540E *pgmname* : *program_qualifier* : **The program was not able to execute the autobind operation successfully on package**
pkg_name **from the file**
path_filename. **The SQLSTATE**
sqlstate **was returned.**

説明: プログラムを実行するには、バインドまたは再バインドが必要です。プログラムは自動バインドを試みましたが、自動バインドは失敗しました。プログラムは初期化できませんでした。

ユーザーの処置: 対応するデータベース・メッセージを見て、自動バインドが失敗した原因につい

ての追加の詳細があるかどうか調べてください。可能性のある問題としては、許可の失敗、コントロール表の欠落または誤り、またはプログラムと一致しないバインド・ファイルなどがあります。状態を訂正し、プログラムを再始動します。

ASN0541E *pgmname* : *program_qualifier* : **An incorrect value** *column_value* **was supplied for column** *column_name* **of the program parameter table**
parms_table.

説明: このメッセージは、キャプチャー・プログラムの初期化中に、IBMSNAP_CAPPARMS 表の妥当性検査でパラメーター設定値の 1 つが無効であることが検出された場合に出されます。キャプチャー・プログラムはこのエラーで終了します。

ユーザーの処置: 資料を参照し、この表に可能な許容パラメーター値を調べてください。値を訂正し、キャプチャー・プログラムを再始動します。

ASN0542E *pgmname* : *program_qualifier* : **The maximum number of lock time-out or deadlock retries has been reached, SQLSTATE**
sqlstate **was returned.**

説明: キャプチャー・プログラムは内部的に、タイムアウトまたはデッドロック条件を複数回、再試行しました。持続するロック条件を受けているプログラム・タスクが重大なもの (たとえばキャプチャーのワーカー・スレッドなど) である場合、キャプチャー・プログラム全体が終了します。プログラム・タスクが重大でない場合 (ブルーニングやモニターなど)、タスクは後で再試行され、キャプチャー・プログラムはアクティブなままです。

ユーザーの処置: 対応するデータベース・メッセージを見て、ロック競合の元となる原因についての追加の詳細があるかどうか調べてください。エラーがユーザー・エラー (たとえばユーザー保持

ロック) の場合は、状態を訂正します。条件が持続する場合は、IBM ソフトウェア・サポートに連絡してください。

ASN0543E *pgmname* : *program_qualifier* : **The program cannot obtain *bytes_number* bytes of storage for a object.**

説明: プログラムは、必要なメモリー内ストレージ構造用のメモリーを入手できません。プログラムは終了します。

ユーザーの処置: プログラムにより大きなメモリーを割り当てることを考慮し、プログラムを再始動します。

ASN0544E *pgmname* : *program_qualifier* : **The program is already active.**

説明: DB2 サブシステムまたはデータベースごとに複数のキャプチャー・プログラムを開始しようとした。

- VSE/ESA の場合、VSE 版キャプチャーは各データベースごとに固有のロック名を生成します。このロック名が使用中であれば、VSE 版キャプチャーがそのデータベースにすでにアクティブであることを示します。
- VSE/ESA の場合、VM 版キャプチャーは、ロックとして使用されるリソース ID がすでに使用中であると判断しました。リソース ID は、CAPTURE ASNPARMs ファイルの ENQ_NAME パラメーターに指定されます。

ユーザーの処置:

- DB2 for z/OS サブシステムの場合、データ共有グループのメンバーであるすべてのサブシステムについて、キャプチャー・プログラムのインスタンスを 1 つだけ実行する、またはスタンドアロン・システムのどこでも、キャプチャー・プログラムのインスタンスを 1 つだけ実行します。ENQ リソースを表示し、固有リソース名に違反したものを調べます。

- その他の DB2 データベース・オペレーティング・システムの場合、データベースごとに 1 つだけのキャプチャー・プログラムを実行します。

- VM 版キャプチャーの場合、1 つのシステム上の複数の DB2 データベースに VM 版キャプチャーを実行したい場合、CAPTURE ASNPARMs 内の ENQ_NAME パラメーターを変更し、各キャプチャー・プログラムに固有の値を確実に与えることができます。

ASN0545E *pgmname* : *program_qualifier* : **The program started with the startup parameter PWDFILE, but the password file *password_file* was not found.**

説明: プログラムはパスワード・ファイルを見つけることができません。ユーザーは PWDFILE パラメーターを使用して、パスワード・ファイル名を指定します。ユーザーがパス始動パラメーターを指定する場合、パスワード・ファイルは指定されたディレクトリーに存在する必要があります。ユーザーがパス始動パラメーターを指定しない場合、パスワード・ファイルは、プログラムが実行されている現行ディレクトリーに存在する必要があります。

ユーザーの処置: パスワード・ファイル名が正しく指定されており、適切なディレクトリーにあることを確認します。

ASN0546W *pgmname* : *program_qualifier* : **The program call issued to the Automatic Restart Manager failed. The invoked IXCARM macro is *arm_call*, the return code is *return_code*, and the reason code is *reason_code*.**

説明: キャプチャー・プログラムまたはアプライ・プログラムは、自動リスタート・マネージャー (ARM) に対して、接続できない、切断できない、または作動可能状況を受け取れません。メッ

セージには、失敗した呼び出しおよび、ARM からプログラムに戻された戻りコードまたは理由コードが表示されます。プログラムは終了しませんが、ARM 処理を継続することはできません。

ユーザーの処置: この障害の原因について、自動リスタート・マネージャーの資料を調べてください。

ASN0547I *pgmname : program_qualifier : The number of substitution variables passed, nbr_vars, does not match the number of tokens, nbr_tokens, in the text of message number msg_nbr.*

説明: プログラム・コードとプログラム・メッセージ・ファイルが一致しません。プログラムのリリース・レベルとメッセージ・ファイル・カタログが一致していない可能性があります。

ユーザーの処置: プログラム・メッセージ・ファイルが、該当の許可設定値を使用して正しくインストールされていることを検査してください。

ASN0548I *pgmname : program_qualifier : The program received an operator stop command.*

説明: この通知メッセージは、プログラムに停止コマンドが実行されたことを示します。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0552E *pgmname : program_qualifier : The program encountered an SQL error. The server name is server_name. The SQL request is sql_request. The table name is table_name. The SQLCODE is sqlcode. The SQLSTATE is sqlstate. The SQLERRMC is sql_tokens. The SQLERRP is error_module.*

説明: キャプチャー、アプライ、またはモニター・プログラムが EXEC SQL ステートメントまたは CLI 呼び出しを実行した時に、ゼロ以外の SQLCODE が戻されました。この SQLCODE は、調査すべき DB2 の問題 (たとえば、スペース不足条件、またはアプリケーションが DB2 を使用できないなど) が原因の場合があります。このメッセージには 2 番目のメッセージが続いている場合があります、ここに、この SQLCODE が出された時に行なわれていたレプリケーションについての情報があります。

ユーザーの処置: ご使用のオペレーティング・システムの DB2 データベース・マネージャーの、「メッセージおよびコード」資料を参照し、SQLCODE の説明および、DB2 で取るべき可能性のある訂正アクションの情報を調べてください。レプリケーションが、このメッセージの直後に別のメッセージを出している場合は、そのメッセージの説明およびユーザー処置を見てください。

ASN0553E *pgmname : program_qualifier : Internal error error_number occurred for message number msg_number containing num_tokens substitution fields: sub_tokens.*

説明: *error number* は 10 進の内部エラー番号であり、次のように定義されています。

- 1 インスタンスが無効
- 2 アクセス否認

- 3 ファイルがない
- 4 メッセージがない
- 5 ロケールが無効
- 6 システム・エラー
- 7 メモリーが不足

msg_number は、プログラムが出そうとしていたメッセージです。 *num_tokens* は、メッセージに与えられた置換トークンの番号です (*pgmname* および *program_qualifier* トークンを含まない)。
sub_tokens は、エラーのメッセージの置換トークンがコンマで区切られています。

ユーザーの処置: 示されたエラー・コードに基づき、可能な訂正アクションをします。たとえば、メッセージ・ファイルが見つからない、またはアクセスできない場合は、メッセージ ASN0513 も見る必要があります、ここにファイル名があります。正しい許可を持つメッセージ・ファイルが存在することを検査します。エラー・コード 4 が出された場合は、古いメッセージ・ファイルが存在する可能性があります。

ASN0554E *pgmname : program_qualifier : The program encountered a DB2 log full condition on server server_name.*

説明: プログラムは挿入または更新を処理しようとしたのですが、DB2 トランザクション・ログがいっぱいのため、DB2 に拒否されました。プログラムは処理を停止します。

ユーザーの処置: データベース・ファイルを含むファイル・システム上に残っているスペース量をチェックします。データベース構成ファイル内の最大ログ・サイズを増やすことを検討してください。

ASN0555W *pgmname : program_qualifier : The program cannot register with Automatic Resource Manager (ARM) because it is not APF authorized.*

説明: キャプチャー、アプライ、またはモニター・プログラムは、プログラム・ライブラリーが APF 許可されていないため、自動リソース・マネージャー・サービスの使用を登録できません。

ユーザーの処置: キャプチャー、アプライ、またはモニター・プログラムを自動リソース・マネージャーに登録したい場合は、プログラム・ライブラリーを APF 用に許可し、プログラムを再始動します。

ASN0777I *pgmname : program_qualifier : Additional information message_text, reason code(s): rc1, rc2, rc3.*

説明: このメッセージ内に示された *Additional information* は、情報テキスト・メッセージを参照します。理由コードは、このメッセージ・テキストに関連する追加の戻りコード情報を提供します。情報コード・フィールドが該当しない場合、ここには "*" (アスタリスク) が含まれます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN0888E *pgmname : program_qualifier : EEE error condition message_text, error code(s): rc1, rc2, rc3.*

説明: このメッセージに示されている *EEE error condition* は、指定された修飾子 (表示されている場合) を持つ、指定されたプログラム内で起こった EEE 特有のエラーの記述です。エラー・コードは、このメッセージ・テキストに関連する追加の情報を提供します。エラー・コード・フィールドが該当しない場合、ここには "*" (アスタリスク) が含まれます。

ユーザーの処置: *EEE error condition* の情報と、示されたエラー・コードの情報を使用して、エラーの原因を判別してください。エラーを解決できない場合は、IBM ソフトウェア・サポートに連絡してください。

ASN0999E *pgmname : program_qualifier :*
Error condition *message_text,*
error code(s): *rc1, rc2, rc3.*

説明: このメッセージに示されている *Error condition* は、指定された修飾子 (表示されている場合) を持つ、指定されたプログラムで起こったエラーの記述です。エラー・コードは、このメッセージ・テキストに関連する追加の情報を提供します。エラー・コード・フィールドが該当しない場合、ここには "*" (アスタリスク) が含まれません。

ユーザーの処置: *Error condition* の情報と、示されたエラー・コードの情報を使用して、エラーの原因を判別してください。エラーを解決できない場合は、IBM ソフトウェア・サポートに連絡してください。

ASN1001E **APPLY** *apply_qualifier.* **The Apply program encountered an SQL error. The** *ERRCODE is error_code. The SQLSTATE is sqlstate. The SQLCODE is sqlcode. The SQLERRM is sqlerrm. The SQLERRP is sqlerrp. The server name is server_name. The table name is table_name.*

説明: SQL ステートメントの実行中にエラーが起きました。

ユーザーの処置: SQL エラー・コードの説明は、データベースのメッセージ・リファレンスを参照してください。

ASN1002E **APPLY** *apply_qualifier.* **The** *table_name* **could not be locked. ERRCODE is** *error_code,* **SQLSTATE is** *sqlstate,* **SQLCODE is** *sqlcode,* **SQLERRM is** *sqlerrm,* **SQLERRP is** *sqlerrp,* **server name is** *server_name,* **table name is** *table_name*

説明: アプライ・プログラムは表をロックできませんでした。

ユーザーの処置: データベースのメッセージ・リファレンスを参照してください。

ASN1003E **APPLY** *apply_qualifier.* **The Apply program could not connect to the server** *server.* **The error code is** *error_code.* **The SQLSTATE is** *sqlstate.* **The SQLCODE is** *sqlcode.* **The SQLERRM is** *sqlerrm.* **The SQLERRP is** *sqlerrp.*

説明: アプライ・プログラムはデータベースに接続を試み、失敗を示す戻りコードを受け取りました。アプライ・プログラムがデータベースに接続できない理由は、多数考えられます。たとえば、データベースがダウンしているとか、多すぎるユーザーがデータベースをアクセスしていると、失敗の戻りコードを受け取ります。

ユーザーの処置: DB2 の「メッセージおよびコード」の資料で SQLCODE を探し、接続が失敗した理由を調べます。ユーザー ID とパスワードを保管する情報については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の「レプリケーションのセットアップ」の説明を参照してください。

SQL エラー・コードの説明は、データベースのメッセージ・リファレンスを参照してください。

ASN1006E APPLY *apply_qualifier*. The product registration module has unexpected content.

説明: DB2 レプリケーションの登録モジュール (ASNAPR61) の内容が、このバージョンの DB2 用ではありません。正しい登録モジュールを提供するまで、製品を使用することはできません。

ユーザーの処置: DB2 がエラーなしで正常にインストールされたことを検査してください。エラーが起きている場合は、これを訂正して再度試みてください。

DB2 が正常にインストールされており、フィーチャ登録モジュール (ASNAPR61) を正しくアクセスしている場合は、IBM ソフトウェア・サポートに連絡してください。

ASN1008E APPLY *apply_qualifier*. The subscription set with Apply qualifier *qualifier* and set name *set_name* is not defined correctly. ERRCODE is *error_code*.

説明: サブスクリプション・セットが正しく定義されていません。

ユーザーの処置: サブスクリプション・セットの WHOS_ON_FIRST 列が正しく指定されていることを確認してください。

ASN1009E APPLY *apply_qualifier*. There is no subscription set defined for Apply qualifier *qualifier*.

説明: アプライ修飾子 *qualifier* にサブスクリプション・セットが定義されていません。

ユーザーの処置: アプライ修飾子 *qualifier* に少なくとも 1 つのサブスクリプション・セットを定義してください。

ASN1010E APPLY *apply_qualifier*. The Apply program could not insert row *row* into the audit trail table due to the following error: *error_code*.

説明: これは、監査証跡表が IBMSNAP_APPLYTRAIL 表と同じ構造を使用してセットアップされていないことを示す SQL 戻りコードです。

ユーザーの処置: 「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明および、データベース SQL 資料を参照してください。

ASN1011E APPLY *apply_qualifier*. The copy request has incompatible source and target attributes. The SQL code is *error_code*.

説明: これは、ターゲット表の属性はソース表の属性と互換性がなければならぬことを示す SQL コードです。

ユーザーの処置: ソースとターゲットの属性の互換性については、登録表の SOURCE_STRUCTURE 列を参照してください。

ASN1012E APPLY *apply_qualifier*. The source table structure is not valid. The error code is *error_code*.

説明: これは、登録票のソース表構造が、登録表の SOURCE_STRUCTURE 列にしたがってセットアップされていないことを示す SQL 戻りコードです。

ユーザーの処置: IBMSNAP_REGISTER 表で使用される有効な SOURCE_STRUCTURE 列の値については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照してください。

ASN1013E APPLY *apply_qualifier*. The target table structure is not valid. The error code is *error_code*.

説明: subscription-targets-member 表のターゲット表の構造が無効です。

ユーザーの処置: 有効なターゲット表の構造については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照してください。

ASN1014E APPLY *apply_qualifier*. The Apply program could not find a source for the copy request because it could not find the change data table. The error code is *error_code*.

説明: 登録表に変更データ表が定義されていません。その理由は、アプライ・プログラムが変更データ表の名前を登録表に見つけられなかったか、またはソース表が正しく登録されていません。

ユーザーの処置: 「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照し、登録表に変更データ表が正しく定義されていることを検査してください。

ASN1015I APPLY *apply_qualifier*. The Apply program is waiting for the Capture program at server *server_name* to advance the global SYNCHTIME. Verify that the Capture program is running.

説明: このメッセージは単なる通知です。

ユーザーの処置: キャプチャー・プログラムが実行されていることを検査してください。

ASN1016I APPLY *apply_qualifier*. Refresh copying has been disabled. The error code is *error_code*.

説明: フル・リフレッシュを実行しようとし、アプライ・プログラムは登録表の DISABLE_REFRESH 列がオンにセットされていることを検出しました。

ユーザーの処置: DISABLE_REFRESH 列をオフにするか、またはアプライ・プログラムをう回して手動リフレッシュを実行します。

ASN1017E APPLY *apply_qualifier*. The Apply program could not find any target column names. The error code is *error_code*.

説明: アプライ・プログラムはサブスクリプション列表に列を見つけられません。

ユーザーの処置: サブスクリプション・セットとサブスクリプション・セット・メンバーを再定義します。詳細については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の「レプリケーションのセットアップ」の説明を参照してください。

ASN1018I APPLY *apply_qualifier*. The Apply program is processing subscription set *set_name(whos_on_first).(set_number of total_sets)*.

説明: このメッセージは単なる通知です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1019E APPLY *apply_qualifier*. The target table does not have any key columns. The error code is *error_code*.

説明: アプライ・プログラムで、固有索引または 1 次キーを必要とする列の 1 つにキー列の名前がありません。

ユーザーの処置: サブスクリプション・セットとサブスクリプション・セット・メンバーを再定義します。詳細については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の「レプリケーションのセットアップ」の説明を参照してください。

ASN1020E APPLY *apply_qualifier*. The Apply program could not reserve a storage block. The error code is *error_code*.

説明: アプライ・プログラムが、必要な (メモリ) ストレージを入手できません。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1021E APPLY *apply_qualifier*. The Apply program cannot read the work file *filename*. The error code is *error_code*.

説明: システム・エラーにより、アプライ・プログラムは作業ファイルを読み取ることができません。

ユーザーの処置: 問題がスペース不足により起こったのかどうかを判別し、システム管理者に連絡して必要なものを入手します。

ASN1022E APPLY *apply_qualifier*. The Apply program cannot write to the work file *filename*. The error code is *error_code*.

説明: ユーザーが、ファイルの 1 つまたはすべてに適切なアクセス権限を持っていない、またはターゲット・ファイルに書き込んだ後、不十分なスペースしか残っていません。

ユーザーの処置: 問題がアクセス権限の不足またはスペース不足により起こったのかどうかを判別し、システム管理者に連絡して必要なものを入手します。

ASN1023E APPLY *apply_qualifier*. The Apply program cannot open the work file *filename*. The error code is *error_code*.

説明: アプライ・プログラムが作業ファイルをオープンできません。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1024E APPLY *apply_qualifier*. The Apply program cannot close the work file *filename*. The error code is *error_code*.

説明: アプライ・プログラムが作業ファイルをクローズできません。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1025I APPLY *apply_qualifier*. The Apply program completed processing for subscription set *set_name(whos_on_first)*. The return code is *return_code*.

説明: このメッセージは単なる通知です。

ユーザーの処置: このメッセージは通知のみであ

り、アクションは必要ありません。

ASN1026I APPLY *apply_qualifier*. The Apply program encountered an error while trying to bind. SQLSTATE is *sqlstate*, SQLCODE is *sqlcode*.

説明: バインドの実行中にエラーが起きました。

ユーザーの処置: データベースのメッセージ・リファレンスを参照してください。

ASN1027E APPLY *apply_qualifier*. There are too many large object (LOB) columns specified. The error code is *error_code*.

説明: サブスクリプション・セット・メンバーに指定されたラージ・オブジェクト (BLOB、CLOB、または DBCLOB) 列が多すぎます。可能な列の数の最大は 10 です。

ユーザーの処置: サブスクリプション・セット・メンバーから、超過したラージ・オブジェクト列を除去してください。

ASN1028I APPLY *apply_qualifier*. The before-image column for a key column is not found. The error code is *error_code*.

説明: TARGET_KEY_CHG=Y を持つメンバーのサブスクリプションのセットアップが誤りです。

ユーザーの処置: キー列 (IS_KEY=Y) の場合、IBMSNAP_SUBS_COLS 表に変更前イメージ列を含める必要があります。これは col_type=B (ユーザーが指定)、または col_type=P (レプリケーションが提供) にすることができます。サブスクリプションを手作業でセットアップした場合は、問題を自分で訂正する必要があります。レプリケーション・センターまたはレプリケーション・コマンドを使用してサブスクリプションをセットアップ

した場合は、IBM ソフトウェア・サポートに連絡してください。

ASN1029E APPLY *apply_qualifier*. The SQL statement of the subscription set named *set_name* with a whos_on_first value of *whos_on_first* did not execute successfully. The statement failed with SQLCODE *sqlcode* and SQLSTATE *sqlstate*. The apply program internal error code is *error_code*.

説明: ユーザー指定 SQL ステートメントが正常に実行されませんでした。

ユーザーの処置: 詳細情報は、IBMSNAP_APPLYTRAIL 表の対応する情報および、データベースの SQL 資料を参照してください。

ASN1031E APPLY *apply_qualifier*. The SQL statement is empty. The error code is *error_code*.

説明: SQL ステートメントが空のストリングです。

ユーザーの処置: 実行する SQL ステートメントを指定してください。

ASN1032E APPLY *apply_qualifier*. The Apply program log file could not be opened. The error code is *error_code*, and the return code is *return_code*.

説明: アプリケーション・プログラムがログ・ファイルをオープンできません。

ユーザーの処置: 戻りコードの詳細については、ご使用の特定のオペレーティング・システムの問題判別資料を参照してください。

ASN1033E **APPLY** *apply_qualifier*. The Apply program could not write to the Apply log file. The error code is *error_code*, and the return code is *return_code*.

説明: アプライ・プログラムがログ・ファイルに書き込めません。

ユーザーの処置: 戻りコードの詳細については、ご使用の特定のオペレーティング・システムの問題判別資料を参照してください。

ASN1034I **APPLY** *apply_qualifier*. The Apply program initialization is successful.

説明: このメッセージは、アプライ処理を正常に初期化すると出されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1035E **APPLY** *apply_qualifier*. The Apply program could not access the subscription columns table. The error code is *error_code*. The SQLSTATE is *sqlstate*. The SQLCODE is *sqlcode*. The SQLERRM is *sqlerrm*. The SQLERRP is *sqlerrp*. The server name is *server_name*. The table name is *table_name*.

説明: SQL ステートメントの実行中にエラーが起きました。

ユーザーの処置: データベースの SQL メッセージ・リファレンスを参照してください。

ASN1036E **APPLY** *apply_qualifier*. The column type *col_type* for expression *expression* is not valid. The error code is *error_code*.

説明: サブスクリプション列表の COL_TYPE 列の値が無効です。

ユーザーの処置: 値を A、B、C、D、F、L、または R に変更してください。

ASN1038E **APPLY** *apply_qualifier*. No column names or expressions were specified in the subscription columns table.

説明: COPY ステートメントに対し列名または式を指定する必要があります。

ユーザーの処置: サブスクリプション定義の要件の詳細については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の「レプリケーションのセットアップ」の説明を参照してください。

ASN1039E **APPLY** *apply_qualifier*. The Apply program plan, *plan_name*, could not be opened. The error code is *error_code*. The return code is *return_code*. The reason code is *reason_code*.

説明: アプライ・プログラムのプランをオープンできません。

ユーザーの処置: 「Apply for z/OS Program Directory」を参照してください。

ASN1040E **APPLY** *apply_qualifier*. **The Apply program encountered an z/OS error. The error code is *error_code*, and the return code is *return_code*.**

説明: z/OS システム操作の実行が失敗しました。

ユーザーの処置: z/OS システム・ライブラリー情報を参照してください。

ASN1041I **APPLY** *apply_qualifier*. **The Apply program was started using subsystem name: *subsystem*.**

説明: このメッセージは、指定されたサブシステム名を使用して、アプライ・プログラムが開始されたことを知らせています。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1042W **APPLY** *apply_qualifier*. **There are too many invocation parameters.**

説明: アプライ・プログラムの呼び出し時に指定したパラメーターの数が、可能な最大を超えています。

ユーザーの処置: 呼び出しパラメーターの該当する数については、該当オペレーティング・システムの「キャプチャーおよびアプライ」の章を参照してください。

ASN1043E **APPLY** *apply_qualifier*. **There is already one Apply instance running with this Apply program qualifier *qualifier*. The error code is *error_code*, and the reason code is *reason_code*.**

説明: 検査の試みは失敗しました。

ユーザーの処置: 指定されたアプライ修飾子を持

つアプライ・プログラムの 1 つだけのインスタンスが、このサブシステムまたはデータベースの、このユーザー ID の下で実行されていることを確認してください。

ASN1044I **APPLY** *apply_qualifier*. **The Apply program will become inactive for *number* minutes and *number* seconds.**

説明: アプライ・プログラムは非アクティブです。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1045I **APPLY** *apply_qualifier*. **The Apply program was started using database *database*.**

説明: このメッセージは、どのデータベースからアプライ・プログラムが実行されているかを知らせています。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1047I **APPLY** *apply_qualifier*. **There are too many columns specified. The error code is *error_code*.**

説明: サブスクリプションのメンバーに指定された列が多すぎます。

ユーザーの処置: ユーザーは、サブスクリプションのメンバーに指定された列の数を減らす必要があります。サポートされる列の最大数は、列名をすべて合計した長さにより決まります。列名の長さを短くすれば、より多くの列を指定することができます。

ASN1048E APPLY *apply_qualifier*. The execution of an Apply cycle failed. See the Apply trail table for full details: *text*

説明: アプライ・サイクルが失敗しました。このメッセージで、*text* は、*target_server*、*target_owner*、*target_table*、*stmt_number*、および *cntl_server* を示しています。

ユーザーの処置: 監査証跡表の APPERRM フィールドをチェックし、アプライ・サイクルが失敗した理由を調べてください。

ASN1049E APPLY *apply_qualifier*. The Apply program encountered a system error. The error code is *error_code*. The return code is *return_code*.

説明: システム操作の実行が失敗しました。

ユーザーの処置: 該当のオペレーティング・システムのシステム・ライブラリー情報を参照してください。

ASN1050E APPLY *apply_qualifier*. The Apply program encountered an operation that is not valid while updating the target table. The error code is *error_code*. The operation to be applied is *operation*.

説明: ソース表から取り出した行のオペレーション・フィールドが無効です。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1051E APPLY *apply_qualifier*. The Apply program detected a gap between the source *source table* and the target table. The error code is *error_code*.

説明: アプライ・プログラムは、アプライ・プログラムが変更データをコピーする前に、キャプチャー・プログラムが変更データを失ったことを検出しました。たとえば、キャプチャー・プログラムがコールド・スタートされた、または保存期限によるプルーニングが発生した可能性があります。

ユーザーの処置: コントロール表をチェックして、ギャップが存在する理由を調べてください。コントロール表の情報をリセットして定義を再度実行する前に、データの保全性を維持するための適切なアクションを取ってください。

ASN1052E APPLY *apply_qualifier*. The Apply program could not find the ASNLOAD program.

説明: アプライ・プログラムは現行ディレクトリーに ASNLOAD プログラムを見つけれません。

ユーザーの処置: ASNLOAD が、アプライ・プログラムを呼び出したディレクトリーに存在することを確認してください。

ASN1053E APPLY *apply_qualifier*. The execution of the ASNLOAD exit routine failed. The return code is *return_code*.

説明: ASNLOAD 出口ルーチンがエラーを検出し、エラー情報をアプライ・プログラムに戻しました。次の値が有効な戻りコードです。

98

予期しないエラーが起きました (ASNLOAD 出口ルーチンは予期しないエラーにより、失敗しました。処理は行なわれません。)

DB2 pwdfile キーワードが提供されましたが、パスワード・ファイルが見つかりません (pwdfile パラメーターが渡されましたが、パスワード・ファイルが見つかりません。これはエラーであり、接続およびその他の処理は行なわれません。)

100

user/using 句を持つ DB2 Connect が失敗しました - pwdfile は見つかりました (暗号化されたアプライ・パスワード・ファイルにある値を使用して、user/using 句を使用して接続が行なわれました。パスワード・ファイルおよび DB2 サーバーのユーザー ID/ パスワードの組み合わせは見つかりましたが、接続は失敗しました。)

101

user/using 句を持たない DB2 接続が失敗しました - pwdfile が見つかりません (user/using 句なしで接続が行なわれ、パスワード・ファイルが提供されませんでした。接続は失敗しました。)

102

user/using 句を持たない DB2 接続が失敗しました - pwdfile は見つかりましたが、エントリーがありません (DB2 サーバーのサーバー・エントリーが pwdfile に見つからないので、user/using 句を使用せずに接続が行なわれました。接続は失敗しました。)

103

user/using 句を持つ DB2 Connect が失敗しました - asnload.ini から uid/pwd が使用されました (user/using 句を使用して接続が行なわれ、asnload.ini ファイルにある値が使用されました。このファイルおよび DB2 サーバーのユーザー ID/ パスワードの組み合わせは見つかりましたが、接続は失敗しました。)

104

user/using 句を持たない DB2 接続が失敗しました - asnload.ini が見つかりません (user/using 句なしで接続が行なわれ、asnload.ini ファイルが見

つかりませんでした。接続は失敗しました。)

105

user/using 句を持たない DB2 接続が失敗しました - サーバーの uid/pwd が見つかりません (user/using 句なしで接続が行なわれました。asnload.ini ファイルは見つりましたが、uid/pwd の組み合わせが提供されていません。接続は失敗しました。)

106

ユーザーが LOADX_TYPE = 2 を指定し、ユーザー・コードが提供されません (ユーザーは、ASN.IBMSNAP_SUBS_MEMBR 表の LOADX_TYPE の値を 2 にセットしました。これは、ユーザーが ASNLOAD 出口ルーチンにカスタム・コードを提供することを示します。しかし、このコードが見つからず、ASNLOAD 出口ルーチンはアプライ・プログラムが LOADX_TYPE 値 2 を渡した時に失敗しました。)

107

DB2 インポート・ユーティリティーが失敗しました (インポート・ユーティリティーの実行が失敗しました。ユーティリティーから戻された SQL コードが理由コードとして渡されます。)

108

DB2 エクスポート・ユーティリティーが失敗しました (エクスポート・ユーティリティーの実行が失敗しました。ユーティリティーから戻された SQL コードが理由コードとして渡されます。)

109

DB2 ロード・ユーティリティーが失敗しました (ロード・ユーティリティーの実行が失敗しました。ユーティリティーから戻された SQL コードが理由コードとして渡されます。)

110

DB2 ロード・ユーティリティーが失敗しました - クロスロードとして呼び出されました (ロード・ユーティリティーの実行が失敗しました。ロード・ユーティリティーは、カーソル・オプション

からロードを使用して呼び出されました。ユーティリティから戻された SQL コードが理由コードとして渡されます。)

111

ユーザーは LOADX_TYPE に無効な値をセットしました (ASNLOAD 出口ルーチンが、LOADX_TYPE 値 (ユーザーがセットした値) を使用して呼び出されました。LOADX_TYPE 値はこの環境には無効であり、ASNLOAD 出口ルーチンは失敗しました。)

112

LOADX_TYPE 3 は select 用ニックネームが必要です (ASNLOAD 出口ルーチンは失敗しました。ASNLOAD 出口ルーチンが、ユーザーがセットした LOADX_TYPE 値を使用して呼び出されました。リモートの DB2 ソース表にニックネームを作成し、ASN.IBMSNAP_SUBS_MEMBR 表に保管してある場合を除き、LOADX_TYPE 値はこの環境には無効です。)

113

LOADX_TYPE 4 はターゲット表と互換性がありません (ASNLOAD 出口ルーチンは失敗しました。ASNLOAD 出口ルーチンが、ユーザーがセットした LOADX_TYPE を使用して呼び出されました。DB2 Linux/UNIX/Windows ロード・ユーティリティではターゲット表をサービスできないため、LOADX_TYPE 値はこの環境には無効です。)

114

LOADX_TYPE 5 はターゲット表と互換性がありません (ASNLOAD 出口ルーチンは失敗しました。ASNLOAD 出口ルーチンが、ユーザーがセットした LOADX_TYPE を使用して呼び出されました。DB2 インポート・ユーティリティではターゲット表をサービスできないため、LOADX_TYPE 値はこの環境には無効です。)

115

ASNDLCOPY 出口ルーチンが失敗しました (ASNLOAD 出口ルーチンは、サブスクリプション・セット・メンバー用の DATALINK 列があるため、ASNLCOPY 出口ルーチンを呼び出しました。ASNLCOPY 出口ルーチンが失敗しました。したがって、このサブスクリプション・セット・メンバーをロードする処理も失敗しました。)

ASNLCOPY 出口ルーチンが失敗しました。したがって、このサブスクリプション・セット・メンバーをロードする処理も失敗しました。)

ユーザーの処置: 戻りコードおよび対応する説明 (上記) を調べます。ASNLOAD メッセージ・ファイルの追加情報および、該当する場合は DB2 ユーティリティが生成したメッセージ・ファイルの追加情報をチェックします。

ASN1054E **APPLY** *apply_qualifier*. The Apply program could not find a row in the IBMSNAP_REGISTER or IBMSNAP_PRUNCNTL table that corresponds to the subscription set member with a set name *set_name*, for source owner *src_ownr*, source table *src_tbl*, and source view qualifier *src_view_qual*.

説明: ソース表の登録が誤り、または不完全です。

ユーザーの処置: 登録をドロップし、再定義してください。

ASN1055E **APPLY** *apply_qualifier*. The Apply program could not find the prune control information for source owner *src_ownr*, source table *src_tbl*, source view qualifier *src_view_qual*, target owner *tgt_ownr*, and target table *tgt_tbl*.

説明: ソース表の登録が誤りです。

ユーザーの処置: サブスクリプションをドロップし、再度行ってください。

ASN1056E APPLY *apply_qualifier*. The Apply program could not connect to the server due to lack of user ID/password. The error code is *error_code*.

説明: アプライ・プログラムは、サーバーに接続するためのパスワードとユーザー ID を見つけれませんでした。

ユーザーの処置: アプライ・プログラムのパスワード・ファイルが存在することを確認してください。アプライ・プログラムのパスワード・ファイルは、アプライ・プログラムを開始したディレクトリーと同じディレクトリーにあります。DB2 Universal Database Satellite Edition を使用している場合は、パスワードとユーザー ID がクライアント・システムに定義されていることを確認してください。

ASN1057E APPLY *apply_qualifier*. The Apply program could not read the password in the Apply password file. The error code is *error_code*.

説明: アプライ・プログラムはパスワードを見つけれませんでした。

ユーザーの処置: AUTHENTICATION=SERVER 体系を使用したい場合は、該当のオペレーティング・システムの「キャプチャーおよびアプライ」の章の、「アプライ・プログラム」セクションに記述されているように、パスワードを提供する必要があります。

ASN1058E APPLY *apply_qualifier*. The Apply program could not close the password file. The error code is *error_code*.

説明: アプライ・プログラムはパスワード・ファイルをクローズできませんでした。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1059E APPLY *apply_qualifier*. The Apply program detects syntax that is not valid for line *line* in the password file. The error code is *error_code*.

説明: アプライ・プログラムはパスワード・ファイル内の行を認識できませんでした。

ユーザーの処置: パスワード・ファイルの構文エラーを訂正してください。該当のオペレーティング・システムの「キャプチャーおよびアプライ」の章の、「アプライ・プログラム」セクションを参照してください。

ASN1060E APPLY *apply_qualifier*. The dynamic allocation for the temporary work file failed. The error code is *error_code*.

説明: 動的割り振り中にシステム・エラーが起きました。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1061E APPLY *apply_qualifier*. The specified keyword parameter is not valid. The error code is *error_code*.

説明: 無効な呼び出しパラメーターが指定され、アプライ・プログラムはこれを無視しました。

ユーザーの処置: 呼び出しパラメーターを訂正します。該当のオペレーティング・システムの「キャプチャーおよびアプライ」の章の、「アプライ・プログラム」セクションを参照してください。

ASN1062W APPLY *apply_qualifier*. The Apply program must use SELECT and INSERT statements to perform a full refresh of this subscription-set member. The following information pertains to this subscription-set member: the set name is *set_name*, the source owner is *source_owner*, the source table is *source_table*, the source view qualifier is *source_view_qual*, the target owner is *target_owner*, and the target table is *target_table*.

説明: ASNLOAD 出口ルーチンは、ユーザー指定 LOADX_TYPE 値を検出できず、このサブスクリプション・セット・メンバーを処理するユーティリティを使用できません。したがって、ASNLOAD 出口ルーチンはフル・リフレッシュのコントロールをアプライ・プログラムに戻します。ASNLOAD 出口ルーチンは現在、ある種のターゲット表タイプ (たとえば Sybase および MS SQL Server ターゲット表) をサポートせず、これを処理できません。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。しかし、これらのサブスクリプション・セット・メンバーの LOADX_TYPE の値を 1 にセットすれば、ASNLOAD 出口ルーチンによる不必要な処理を避けることができます。

ASN1063E APPLY *apply_qualifier*. A subscription set cannot have more than 200 members. The error code is *error_code*.

説明: サブスクリプションの数が、可能な最大数 200 を超えました。

ユーザーの処置: サブスクリプション・セットから超過のメンバーを除去します。

ASN1064W APPLY *apply_qualifier*. The Apply program cannot perform a full refresh for the subscription set named *set_name*, because the Capture program for this source has not yet been cold started.

説明: アプライ・プログラムはサブスクリプション・セットのフル・リフレッシュを試みることができません。その理由は、このソースのキャプチャー・プログラムが一度もコールド・スタートをしたことがなく、アプライ・プログラムが挿入する CAPSTART シグナルを処理するための準備ができていないためです。

ユーザーの処置: このソースのキャプチャー・プログラムを開始してください。

ASN1065E APPLY *apply_qualifier*. The Apply program cannot perform a full refresh for the subscription set named *set_name*, because the registered source table *source_owner.source_table* is in a stopped state.

説明: アプライ・プログラムはこのサブスクリプション・セットにフル・リフレッシュを実行することができません。その理由は、ソース登録の 1 つが停止状態である (IBMSNAP_REGISTER 表の STATE 列の値が 'S' になっている) ためです。キャプチャー・プログラムは、ユーザーの介入を必要とする登録上の問題がある場合、登録を停止状態にします。登録用にキャプチャーされたデータの安全性が疑わしい可能性があるため、アプライ・プログラムはフル・リフレッシュを実行する必要があります。(これは、登録されたソース表がデータ・キャプチャーなしで変更された場合に起こる可能性があります。)

ユーザーの処置: 生成されたキャプチャー・エラー・メッセージの情報を使用して、停止された登録の問題を修正します。手作業で、

IBMSNAP_REGISTER 表の STATE 列の値を 'I' (Inactive - 非アクティブ) にします。これでアプライ・プログラムはフル・リフレッシュを実行することができます。

ASN1066E **APPLY** *apply_qualifier*. An internal Apply program error occurred. The error code is *error_code*.

説明: アプライ・プログラムの内部エラーが起りました。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1067E **APPLY** *apply_qualifier*. The Apply program has detected update conflicts and compensated rejected transactions. See the unit-of-work table for details. The error code is *error_code*.

説明: 別のロケーションから、複数のアプリケーションが表の同じ行を更新しました。いくつかのトランザクションは、拒否され、補整されています。

ユーザーの処置: 詳細は、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」資料の表構造の説明を参照してください。

ASN1068E **APPLY** *apply_qualifier*. The Apply program has deactivated the subscription due to an RI violation. The error code is *error_code*.

説明: ソース表からレプリカにデータをコピーしている時に、参照保全性への違反が検出されました。アプライ・プログラムは終了し、サブスクリプションは非活動化されました。

ユーザーの処置: 参照保全性のエラーを訂正し、

サブスクリプションを再度活動化します。

ASN1070E **APPLY** *apply_qualifier*. The Apply program could not lock the target table. The ERRCODE is *error_code*. The SQLSTATE is *sqlstate*. The SQLCODE is *sqlcode*. The SQLERRM is *sqlerrm*. The SQLERRP is *sqlerrp*. The server name is *server_name*. The table name is *table_name*.

説明: アプライ・プログラムは、更新競合をチェックする前に、ターゲット表をロックできませんでした。

ユーザーの処置: アプライを再始動する前に、すべてのターゲット表が使用可能であることを検査してください。

ASN1071E **APPLY** *apply_qualifier*. The Apply program could not reposition the work file. The error code is *error_code*.

説明: アプライ・プログラムは、一時作業ファイルの読み取り中にエラーを検出しました。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1072E **APPLY** *apply_qualifier*. The Apply program could not find the ASNDONE program.

説明: アプライ・プログラムは、ユーザー出口プログラム ASNDONE を見つけられませんでした。

ユーザーの処置: ASNDONE プログラムが正しいディレクトリーにあることを検査してください。

ASN1073E APPLY *apply_qualifier*. The execution of the ASNDONE program failed. The return code is *return_code*.

説明: ユーザー出口プログラム ASNDONE の呼び出し中にエラーが起きました。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1074E APPLY *apply_qualifier*. The Apply program could not find the ASNDLCOPY program.

説明: アプライ・プログラムは、現行の探索パスに ASNDLCOPY プログラムを見つけられませんでした。

ユーザーの処置: ASNDLCOPY プログラムを探索パスに追加し、アプライ・プログラムを再度実行します。

ASN1075E APPLY *apply_qualifier*. The ASNDLCOPY program failed. The return code is *return_code*. Additional information can be found in the ASNDL *appl_qualset_namecapture_server target_server*.LOG file.

説明: ASNDLCOPY プログラムがエラーを検出し、エラー情報をアプライ・プログラムに戻しました。次の値が有効な戻りコードです。

98

予期しないエラーが起きました。

99

ASNDLCOPY プログラムに渡された引き数が無効です。

100

メモリーの割り振りができません。

101

ASNDLSRVMAP 構成ファイルをオープンできません。

102

ASNDLSRVMAP 構成ファイルのエントリー数が最大数を超過しています。

103

ASNDLSRVMAP 構成ファイルに無効なエントリーが見つかりました。

104

指定されたファイル・サーバーの ASNDLUSER 構成ファイルにユーザー・ログイン情報が見つかりません。

105

ASNDLPARM 構成ファイルに無効なエントリーが見つかりました。

106

ASNDLUSER 構成ファイルをオープンできません。

107

ASNDLUSER 構成ファイルに無効なエントリーが見つかりました。

108

入力ファイルからの読み取り中に入出力エラーが起きました。

109

入力ファイルに無効なエントリーが見つかりました。

110

入力ファイルをオープンできません。

111

結果ファイルをオープンできません。

112

結果ファイルへの書き込み中に入出力エラーが起きました。

113
FTP プロトコルのコントロール・チャンネルを初期化中にエラーが起きました。

114
コントロール・チャンネルを使用してデータを送信中にエラーが起きました。

115
指定されたユーザーおよびパスワードを使用して、ファイル・サーバーにログオンできません。

116
コピー・デーモンがまだ開始されていません。

117
FTP プロトコルのデータ・チャンネルを初期化中にエラーが起きました。

118
ソース・ファイル・サーバーからファイルを検索できません。

119
ターゲット・ファイル・サーバーにファイルを保管できません。

120
受動モードでファイルを転送中にエラーが起きました。

121
指定されたファイル参照のパス・マッピングを見つけれません。

122
FTP BINARY コマンドの実行時にエラーが起きました。

123
FTP SIZE コマンドの実行時にエラーが起きました。

124

FTP MODTIME コマンドの実行時にエラーが起きました。

125
FTP SITE UMASK コマンドの実行時にエラーが起きました。

126
FTP SITE TOUCH コマンドの実行時にエラーが起きました。

127
FTP SITE CHMOD コマンドの実行時にエラーが起きました。

ユーザーの処置: 戻りコードおよび対応する説明(上記)を調べます。戻りコードは、製品と一緒に配布されるサンプル ASNDLCOPY プログラムに基づきます。ログ・ファイルに追加情報が提供されます。

ASN1076E The Apply program cannot read the format of the result file that was generated by the ASNDLCOPY program.

説明: ASNDLCOPY プログラムにより生成された結果ファイルが、予想したフォーマットではありません。

ユーザーの処置: ASNDLCOPY プログラムを変更した場合は、その変更により無効なフォーマットがもたらされていないかチェックしてください。その変更が問題の原因ではない場合、使用するマシンに結果ファイル用の十分なスペースがあることをチェックしてください。

ASN1077E **APPLY** *apply_qualifier*. The Apply program encountered an **DATALINK** column value that is not valid while updating the target table. The error code is *error_code*.

説明: ソース表から取り出した行の DATALINK 列フィールドが無効です。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1078E **APPLY** *apply_qualifier*. The **ASNDLCOPY** program was terminated by the signal *signal_number*. Additional information can be found in the *filename* file.

説明: ASNDLCOPY プログラムは、示されているシグナルで異常終了しました。

ユーザーの処置: 指定されたログ・ファイルを見て、エラーの原因を調べてください。

ASNDLCOPY プログラムを変更し、変更されたコードによりシグナルが生成された場合は、コードを修正して再実行してください。それ以外の場合は、IBM ソフトウェア・サポートに連絡してください。

ASN1097I **APPLY** *apply_qualifier*. The Apply program stopped.

説明: 前に報告されたエラーにより、アプライ・プログラムが停止しました。

ユーザーの処置: このメッセージの前に報告されたエラーを修正します。

ASN1207E **APPLY** *apply_qualifier*. The subscription for *subscription* was not activated.

説明: 選択されたサブスクリプションは非アクティブです。

ユーザーの処置: サブスクリプションをアクティブにするか、または別のものを選択します。

ASN1210E **APPLY** *apply_qualifier*. An Apply qualifier must be specified following the keyword **-q**.

説明: アプライ修飾子を、キーワード **-q** の後に指定する必要があります。

ユーザーの処置: キーワード **-q** の後にアプライ修飾子を指定してください。

ASN1212E **APPLY** *apply_qualifier*. A read-only set name *set_name* is found following the keyword *keyword*.

説明: キーワード **U** または **D** の後に読み取り専用のセット名が指定されました。

ユーザーの処置: キーワード **U** および **D** には、レプリカのみを指定してください。

ASN1221I **APPLY** *apply_qualifier*. Set *set_name* has been successfully refreshed with *number* rows at *time*.

説明: このメッセージは単なる通知です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1242E **APPLY** *apply_qualifier*. An SQL error occurred. **ERRCODE** is *error_code*, **SQLSTATE** is *sqlstate*, **SQLCODE** is *sqlcode*, **SQLERRM** is *sqlerrm*, **SQLERRP** is *sqlerrp*, **table name** is *table_name*.

説明: このメッセージは単なる通知です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1243E **APPLY** *apply_qualifier*. There is no eligible subscription in the **ASN.IBMSNAP_SUBS_SET** table.

説明: サブスクリプション・セットが選択されていないか、またはアプライ修飾子が無効です。

ユーザーの処置: サブスクリプション名とアプライ修飾子を検査します。

ASN1304E **APPLY** *apply_qualifier*. The **ASNSAT** program terminated due to a Capture error.

説明: キャプチャー・プログラムがエラーを戻しました。

ユーザーの処置: キャプチャー・ログ・ファイルからエラーを判別します。

ASN1305E **APPLY** *apply_qualifier*. The **ASNSAT** program terminated due to an Apply error.

説明: アプライ・プログラムがエラーを戻しました。

ユーザーの処置: アプライ・ログ・ファイルからエラーを判別します。

ASN1310E **APPLY** *apply_qualifier*. The **ASNSAT** program encountered a system error while attempting to invoke the Capture program. **Return code** is *return_code*.

説明: ASNCAP の呼び出し中に、オペレーティング・システム・エラーが起きました。

ユーザーの処置: キャプチャー・プログラムが実行パス内にあることを確認してください。

ASN1311E **APPLY** *apply_qualifier*. The **ASNSAT** program encountered a system error while attempting to invoke the Apply program. **Return code** is *return_code*.

説明: ASNAPPLY の呼び出し中に、オペレーティング・システム・エラーが起きました。

ユーザーの処置: アプライ・プログラムが実行パス内にあることを確認してください。

ASN1312E **APPLY** *apply_qualifier*. The **default target server, DB2DBDFT, is not set.**

説明: ユーザーはターゲット・サーバー名を指定せず、ASNSAT プログラムは DB2DBDFT からデフォルトのデータベース名を判別できませんでした。

ユーザーの処置: ターゲット・サーバー名を `-t` キーワードの後に指定します。

ASN1314E **APPLY** *apply_qualifier*. An SQL error occurred while **ASNSAT** was getting the default Apply qualifier. **SQLSTATE** is *sqlstate*, **SQLCODE** is *sqlcode*.

説明: ユーザーはアプライ修飾子を指定しませんでした。ASNSAT プログラムは USER 特殊レジスターを検索中にエラーを検出しました。

ユーザーの処置: キーワード `-q` の後にアブライ修飾子を指定してください。

ASN1315E **APPLY** *apply_qualifier*. **Cannot connect to database server. SQLSTATE is** *sqlstate*, **SQLCODE is** *sqlcode*.

説明: ターゲット・データベースへの接続を試みている時に、エラーが起きました。

ユーザーの処置: データベースのメッセージ・リファレンスを参照してください。

ASN1316E **APPLY** *apply_qualifier*. **ASNSAT encountered an error while trying to bind. The SQLSTATE is** *sqlstate*, **SQLCODE is** *sqlcode*.

説明: 自動バインドを試みている時に、エラーが起きました。

ユーザーの処置: `sqllib\bnd` ディレクトリーにバインド・ファイルが存在することを確認してください。

ASN1317E **APPLY** *apply_qualifier*. **An SQL error occurred while ASNSAT was getting the CD_TABLE value from ASN.IBMSNAP_REGISTER table. SQLSTATE is** *sqlstate*, **SQLCODE is** *sqlcode*.

説明: 登録表から選択している時に、SQL エラーが起きました。

ユーザーの処置: データベースのメッセージ・リファレンスを参照してください。

ASN1318E **APPLY** *apply_qualifier*. **An SQL error occurred while ASNSAT attempted to get the DB2 node type. SQLSTATE is** *sqlstate*, **SQLCODE is** *sqlcode*.

説明: ノード・タイプ構成パラメーターを検索中に、エラーが起きました。

ユーザーの処置: データベースのメッセージ・リファレンスを参照してください。

ASN1500I **The Replication action** *action_name* **started at** *timestamp* **with architecture level** *architecture_level*. **The Capture server is** *capture_serveralias* **and the Capture schema is** *capture_schema*.

説明: *action name* に有効な値は、*Create Capture server control tables* および *Drop Capture server control tables* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1501I **The Replication action** *action_name* **started at** *timestamp* **with architecture level** *architecture_level*. **The Capture server is** *capture_serveralias*, **the remote server is** *remote_servername*, **and the Capture schema is** *capture_schema*.

説明: *action name* に有効な値は、*Create Capture server control tables* および *Drop Capture server control tables* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1502I The Replication action
action_name **started at** *timestamp*
with architecture level
architecture_level. **The Apply**
control server is
apply_serveralias.

説明: *action name* に有効な値は、*Create Apply server control tables* および *Drop Apply server control tables* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1503I The Replication action
action_name **started at** *timestamp*.
The Capture server is
capture_serveralias, **the Capture**
schema is *capture_schema*, **the**
source owner is *source_owner*,
and the source table, view, or
nickname is *source_table*.

説明: *action name* に有効な値は、*Create Registration*、*Drop Registration*、*Alter Registration*、*Add Registration*、および *Promote Registration* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1504I The Replication action
action_name **started at** *timestamp*.
The Capture server is
capture_serveralias, **the remote**
server is *remote_server*, **the**
Capture schema is
capture_schema, **the source owner**
is *source_owner*, **and the source**
table, view, or nickname is
source_table.

説明: *action name* に有効な値は、*Create Registration* および *Drop Registration* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1505I The Replication action
action_name **started**. **The**
subscription set information
follows: the Apply control
server is *control_server*, **the**
Apply qualifier is *apply_qualifier*,
the set name is *set_name*, **the**
target server is *target_server* **for**
remote server *remote_servername*,
the Capture server is
capture_server **for remote server**
remote_servername, **and the**
Capture schema is
capture_schema.

説明: *action name* に有効な値は、*Create Subscription Set*、*Drop Subscription Set*、*Alter Subscription Set*、および *Promote Subscription Set* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1506I The Replication action
action_name **started at** *timestamp*.
The subscription set
information follows: the Apply
control server is *control_server*,
the Apply qualifier is
apply_qualifier, **the set name is**
set_name, **the target server is**
target_server, **the Capture server**
is *capture_server* **for remote**
server *remote_servername*, **and the**
Capture schema is
capture_schema.

説明: *action name* に有効な値は、*Create Subscription Set*、*Drop Subscription Set*、*Alter Subscription Set*、および *Promote Subscription Set* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1507I The Replication action
action_name **started at** *timestamp*.
The subscription set information follows: the Apply control server is *control_server*,
the Apply qualifier is *apply_qualifier*, **the set name is** *set_name*, **the target server is** *target_server* **for remote server** *remote_server*, **the Capture server is** *capture_server*, **and the Capture schema is** *capture_schema*.

説明: *action name* に有効な値は、*Create Subscription Set*、*Drop Subscription Set*、*Alter Subscription Set*、および *Promote Subscription Set* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1508I The Replication action
action_name **started at** *timestamp*.
The subscription set information follows: the Apply control server is *control_server*,
the Apply qualifier is *apply_qualifier*, **the set name is** *set_name*, **the target server is** *target_server*, **the Capture server is** *capture_server*, **and the Capture schema is** *capture_schema*.

説明: *action name* に有効な値は、*Create Subscription Set*、*Drop Subscription Set*、*Alter Subscription Set*、および *Promote Subscription Set* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

り、アクションは必要ありません。

ASN1510I The Replication action
action_name **ended successfully at** *timestamp*.

説明: *action name* に有効な値は、*Create Capture server control tables*、*Drop Capture server control tables*、*Create Apply control server control tables*、および *Drop Apply control server control tables* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1511I The Replication action
action_name **ended successfully for source owner** *source_owner* **and source table, view, or nickname** *source_table*.

説明: *action name* に有効な値は、*Create Registration*、*Drop Registration*、*Alter Registration*、*Add Registration Column*、および *Promote Registration* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1512I The Replication action
action_name **ended successfully for Apply qualifier** *apply_qual*, **set name** *set_name*.

説明: *action name* に有効な値は、*Create Subscription Set*、*Drop Subscription Set*、*Alter Subscription Set*、*Add Statements to Subscription Set*、*Drop Statements from Subscription Set*、および *Promote Subscription Set* です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1513I The Replication action
action_name ended successfully
for Apply qualifier *apply_qual*,
set name *set_name*, who's on
first *whos_on_first*, source owner
source_owner, source table
source_table, source view
qualifier *source_view_qual*, target
owner *target_owner*, and target
table *target_table*.

説明: *action_name* に有効な値は、以下のものです。

- Add Subscription Member
- Add Subscription Member Column
- Drop Subscription Member

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1514I The Replication action ended at
timestamp with successful
successes, errors errors, and
warning warnings.

説明: このメッセージは単なる通知です。

ユーザーの処置: アクションは必要ありません。

ASN1550E The Replication action
action_name ended in error. The
value for the input parameter
input_parameter is missing.

説明: このアクションには入力パラメーターは必須であり、それが欠落しています。

ユーザーの処置: 必要パラメーターを指定し、レプリケーション・アクションを再実行します。

ASN1551E The Replication action
action_name ended in error. The
value *value* for the input
parameter *input_parameter* is
incorrect. The reason code is
reason_code.

説明: 入力パラメーターに提供された値が、有効な値ではありません。次の値が *reason code* に有効な値です。

0 ブロッキングの分の値は 0 から 999 の間にする必要があります。

1 コミット・カウント値は 0 から 999 の間にする必要があります。

2 サーバー・タイプ値は Capture Server にする必要があります。

3 表のタイプ値は次のタイプの 1 つにする必要があります。

- USERTABLE
- CCD TABLE
- POINT IN TIME
- BASE AGGREGATE
- CHANGE AGGREGATE
- REPLICA
- USERCOPY

4 リモート・サーバー名の値は NULL にする必要があります。

5 サーバー・タイプ値は次のタイプの 1 つにする必要があります。

- Capture Server
- Control Server
- Capture and Control Server
- Capture, Control and Target Server

6 内部 CCD 表が不完全です。

7 アプライ修飾子が最大長である 18 文字を超えています。

- 8 セット名が最大長である 18 文字を超えています。
- 9 イベント名の長さは 18 文字以下である必要があります。
- 10 ソースのキャプチャー・スキーマ名が最大長である 30 文字を超えています。
- 11 ターゲットのキャプチャー・スキーマ名が最大長である 30 文字を超えています。
- 12 BEFORE_OR_AFTER ステートメント値は 'A'、'B'、または 'S' である必要があります。
- 13 EI_OR_CALL 値は 'C' または 'E' である必要があります。
- 14 SQLSTATES は 50 桁以下の長さにする必要があります。
- 15 SQLSTATES は数値である必要があります。
- 16 CONFLICT_LEVEL はゼロ (0) または NONE にする必要があります。
- 17 CHGONLY 値は 'N' である必要があります。
- 18 外部 CCD 表はコンデンスされておらず、LOB 列を含んでいます。
- 19 CONFLICT_LEVEL は 0 と 2 の間である必要があります。
- 20 CHGONLY 値は 'Y' または 'N' である必要があります。
- 21 RECAPTURE 値は 'Y' または 'N' である必要があります。
- 22 DISABLE_REFRESH 値は 0 または 1 である必要があります。
- 23 CHG_UPD_TO_DEL_INS 値は 'Y' または 'N' である必要があります。
- 24 STOP_ON_ERROR 値は 'Y' または 'N' である必要があります。

- 25 BEFORE_IMG_PREFIX 値は 1 文字だけにする必要があります。
- 26 前のシナリオのどこでも、対応する表スペースの *New Tablespace* フラグが「真」にセットされていません。
- 27 表名が有効なコントロール表ではありません。コントロール表の有効なリストについては、「レプリケーションのガイド およびリファレンス」を参照してください。

ユーザーの処置: 入力パラメーターに有効な値を指定し、レプリケーション・アクションを再実行します。詳細は、オンライン・ヘルプを参照してください。

ASN1552E The Replication action *action_name* ended in error. The value *value1* for input parameter *input_parameter1* is incompatible with the value *value2* for input parameter *input_parameter2*.

説明: レプリケーション・パラメーターに指定された値が、別のパラメーター指定と矛盾します。

ユーザーの処置: 入力パラメーターに有効な値を指定し、レプリケーション・アクションを再実行します。詳細は、オンライン・ヘルプを参照してください。

ASN1553E The value *value1* for input parameter *input_parameter1* is incompatible with the value *value2* for the existing subscription set *subscription_set*, Apply qualifier *apply_qual*, and who's on first *whoson_first*.

説明: レプリケーション・パラメーターに指定された値が、既存のサブスクリプション・セットの値の 1 つと矛盾します。

ユーザーの処置: 入力パラメーターに有効な値を

指定するか、またはサブスクリプション・セット定義を変更し、レプリケーション・アクションを再実行します。詳細は、オンライン・ヘルプを参照してください。

ASN1560E The replication action ended in error. An SQL error was encountered. SQL Message:
sql_message.

説明: SQL ステートメントの実行中にエラーが起きました。

ユーザーの処置: データベースの SQL メッセージ・リファレンスを参照してください。

ASN1561E Connection to the server *server_alias* cannot be established. An SQL error was encountered. SQL message:
sql_message.

説明: 指定されたサーバーへの接続を確立できませんでした。

ユーザーの処置: データベースの SQL メッセージ・リファレンスを参照してください。ユーザー ID とパスワード情報が正しいことをチェックしてください。

ASN1562E The Replication action ended in error. An unexpected error occurred. Reference Code *reference_code.*

説明: 実行時エラーにより、指定されたアクションを実行できません。

ユーザーの処置: IBM ソフトウェア・サポートに連絡してください。

ASN1563E The Replication action *action_name* ended in error. The Replication architecture level *arch_level* does not support server *server_alias.*

説明: 指定されたレプリケーション・アーキテクチャー・レベルは、指定されたサーバー・オペレーティング・システム、バージョン、またはリリースではサポートされません。この時点で *arch_level* に有効な値は *0801* だけです。

ユーザーの処置: メッセージで指定されたレプリケーション・アーキテクチャーの、サポートされるオペレーティング・システム、バージョン、およびリリースについては、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」のセクションを参照してください。

ASN1564E The Replication action *action_name* ended in error. The Capture server Replication architecture level for Capture schema *capture_schema* does not support this Replication action.

説明: *captureschema.IBMSNAP_REGISTER* にあるレプリケーション・アーキテクチャー・レベルでは、指定されたレプリケーション・アクションはできません。

ユーザーの処置: キャプチャー・サーバーのコントロール表を *0801* にマイグレーションしてから、このアクションを実行してください。

ASN1565E The Replication action *action_name* ended in error. The Apply Control Server Replication architecture level does not support this replication action.

説明: *ASN.IBMSNAP_SUBS_SET* にあるレプリケーション・アーキテクチャー・レベルでは、指

定されたレプリケーション・アクションはできません。

ユーザーの処置: アプライ・コントロール・サーバーのコントロール表を 0801 にマイグレーションしてから、このアクションを実行してください。

ASN1567W The table space container information for table space *tablespace_name* cannot be read, because the DB2 stored procedure *procedure_name* in the library *library_name* cannot be found.

説明: DB2 ストアード・プロシージャー READTSCINFOS が、キャプチャーまたはターゲット DB2 サーバーに見つかりません。このストアード・プロシージャーは、そのサーバーの DB2 表スペース・コンテナ情報を検索するために必要です。

ユーザーの処置: ストアード・プロシージャーがサーバーにあるかどうかをチェックします。ファイル db2rtsc が、sqllib ディレクトリーの関数ディレクトリーにあるかを調べます。サーバーが V8 より前のサーバーの場合、ファイル db2rtsc が存在しない場合があります。ストアード・プロシージャーが存在しない場合は、出力スクリプトに提供された表スペース・コンテナ定義を編集してください。

ASN1568E The name length *length* for the DB2 object, *objectname* exceeds the allowed limit of *allowed_limit*.

説明: 2 番目のパラメーターに指定された DB2 オブジェクト・タイプは、3 番目のパラメーターに指定された実際のオブジェクトの長さよりも短い長さにすることができます。プロパティ・ファイルにあるように、*object* に有効な値は次のものです: *Table*、*Index*、*Tablespace*、*Table owner*。

ユーザーの処置: 「DB2 SQL リファレンス」を参照し、正しい名前の長さを指定してください。

ASN1569E The name of the DB2 object to be created is identical to the existing name *objectowner.objectname* of type *objecttype*.

説明: すでに同じ名前と同じタイプの DB2 オブジェクトがあるため、この DB2 オブジェクトを作成することはできません。プロパティ・ファイルにあるように、*object* に有効な値は次のものです: *Table*、*Index*、*Tablespace*、*Table owner*。

ユーザーの処置: DB2 に存在しないオブジェクト名を指定し、レプリケーション・タスクを再実行してください。

ASN1570E The DB2 object *object*, *objectowner.objectname* does not exist.

説明: DB2 オブジェクトが DB2 カタログにありません。レプリケーション・アクションで、サブスクリプション・セットのソースまたはターゲットとして定義するには、このオブジェクトが存在する必要があります。このオブジェクトは、既存の登録またはサブスクリプション・セット定義の一部で定義されましたが、DB2 カタログに見つかりません。プロパティ・ファイルにあるように、*object* に有効な値は次のものです: *Table*、*Index*、*Tablespace*、*Table owner*。

ユーザーの処置: DB2 に存在する名前を指定し、レプリケーション・タスクを再実行してください。オブジェクトを既存の登録またはサブスクリプション・セット定義の一部で定義した場合は、オブジェクトが DB2 カタログにあることを検査してください。

ASN1571E The DB2 table *tableowner.tablename* cannot be created: the DB2 definition is not valid for data type *datatype* and column *column_name*. The reason code is *reason_code*.

説明: 次の値が *reason code* に有効な値です。

- 0** データ・タイプはこのプラットフォームではサポートされません。
- 1** 列の長さはこのプラットフォームではサポートされません。
- 2** 列のスケールの精度は、このプラットフォームではサポートされません。

ユーザーの処置: 該当の DB2 プラットフォームの「SQL リファレンス」を参照してください。

ASN1572E The row size *row_size* for DB2 object *objectowner.objectname* of type *object_type* exceeds its DB2 buffer pool row size *bufferpool_rowsize*. The DB2 object cannot be created.

説明: 表の行サイズは、その表の表スペースのページ・サイズを超えることはできません。表スペースのページ・サイズは、これが属するバッファ・プールのページ・サイズから導かれます。スクリプトは生成されません。

ユーザーの処置: 別の表スペースに表を作成する必要がある場合があります。該当の DB2 オペレーティング・システムの資料を参照してください。

ASN1573E The number of columns *number_columns* for the DB2 object *objectowner.objectname* of type *object_type* exceeds the DB2 limit *db2_limit*. The DB2 object cannot be created.

説明: DB2 オブジェクト (表または索引) が含むことができる列の数は、DB2 オペレーティング・システムにより異なりますが、事前に定義された数を超えるとはできません。スクリプトは生成されません。オブジェクト・タイプに有効な値は、*table*、*index* です。

ユーザーの処置: DB2 オブジェクトを再設計してください。

ASN1574E The DB2 PageSize *page_size* for Tablespace *tablespace_name* is not valid. Reason code *reason_code*.

説明: 表スペースを正常に作成するには、PageSize が有効である必要があります。有効な理由コードは次の値です。

- 0** PageSize が与えられたバッファ・プールの PageSize と等しくない。
- 1** PageSize が 4K、8K、16K、32K のいずれかでない。

ユーザーの処置: 「DB2 SQL リファレンス」を参照し、該当の PageSize 範囲または値を調べてください。

ASN1575W The DB2 table *tableowner.tablename* will be created in the DB2 default Tablespace.

説明: 指定された表を作成する表スペース名が指定されていないので、表は DB2 デフォルト表スペースに作成されます。デフォルト表スペースの仕様が指定された表にとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。表を、その表の独自の表スペースに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその表にとって適切な場合は、アクションは必要ありません。

ASN1576W The DB2 index *index_name* will be created in the DB2 default Indexspace or Tablespace.

説明: 指定された索引をそこに作成する、表スペース (ワークステーション・オペレーティング・システムの場合) または索引スペース (z/OS オペレーティング・システムの場合) が指定されていません。したがって、索引は DB2 デフォルトを使用して作成されます。デフォルトの仕様が指定された索引にとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。索引を、その索引の独自の表スペースまたは索引スペースに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその索引にとって適切な場合は、アクションは必要ありません。

ASN1577W The DB2 table space *tablespace* will be created in the DB2 default database.

説明: z/OS オペレーティング・システムの場合のみですが、そこに指定された表スペースを作成するデータベースが提供されませんでした。したがって、表スペースは DB2 デフォルトを使用して作成されます。デフォルトの仕様が指定された表スペースにとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。表スペースを、独自のデータベースに入れたい場合は、適切な指定をしてレプリケーション・タスク

を再実行します。デフォルトがその表スペースにとって適切な場合は、アクションは必要ありません。

ASN1578I The DB2 table space *tablespace* will be created in the DB2 default storage group.

説明: ワークステーションおよび z/OS オペレーティング・システムの場合のみですが、そこに指定された表スペースを作成するストレージ・グループが提供されませんでした。したがって、表スペースは DB2 デフォルトを使用して作成されます。デフォルトの仕様が指定された表スペースにとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。表スペースを、独自のストレージ・グループに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその表スペースにとって適切な場合は、アクションは必要ありません。

ASN1579I The DB2 index *index_name* will be created in the DB2 default storage group.

説明: ワークステーションおよび z/OS オペレーティング・システムの場合のみですが、そこに DB2 索引を作成するストレージ・グループが提供されませんでした。したがって、DB2 はデフォルト仕様を使用して索引を作成しました。デフォルトの仕様が指定された索引にとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。索引を、独自のストレージ・グループに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその索引にとって適切な場合は、アクションは必要ありません。

ASN1580I The DB2 table space *tablespace* will be created in the DB2 default buffer pool.

説明: ワークステーションおよび z/OS オペレーティング・システムの場合のみですが、そこに指定された表スペースを作成するバッファ・プールが提供されませんでした。したがって、表スペースは DB2 デフォルトを使用して作成されます。デフォルトの仕様が指定された表スペースにとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。表スペースを、独自のバッファ・プールに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその表スペースにとって適切な場合は、アクションは必要ありません。

ASN1581I The DB2 index *index_name* will be created in the DB2 default buffer pool.

説明: ワークステーションおよび z/OS オペレーティング・システムの場合のみですが、そこに指定された索引を作成するバッファ・プールが提供されませんでした。したがって、索引は DB2 デフォルトを使用して作成されます。デフォルトの仕様が指定された索引にとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。索引を、独自のバッファ・プールに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその索引にとって適切な場合は、アクションは必要ありません。

ASN1582W The Tablespace *tablespace* will be created in Buffer Pool *buffer_pool* but the buffer pool does not exist or is not active.

説明:

- DB2 UDB データベース上のアプリケーションの場合、指定された表スペースを作成するバッファ・プールが存在しません。
- DB2 for z/OS データベース上のアプリケーションの場合、表スペースを作成するバッファ・プールがアクティブではありません。

ユーザーの処置:

- DB2 UDB データベースの場合、スクリプトの実行時にバッファ・プールが存在することを確認してください。
- DB2 for z/OS データベースの場合、スクリプトの実行時にバッファ・プールがアクティブであることを確認してください。

•

ASN1583E The PageSize *page_size* for Tablespace *tablespace* does not match the default buffer pool PageSize.

説明: 与えられた PageSize がデフォルトのバッファ・プール PageSize と一致しません。表スペースを作成できません。

ユーザーの処置: PageSize を変更するか、別のバッファ・プールを選択してください。

ASN1584E The Replication action *action_name* ended in error. The Capture server Replication architecture level *arch_level* for Capture schema *capture_schema* is not a valid architecture level.

説明: captureschema.IBMSNAP_REGISTER にあるレプリケーション・アーキテクチャー・レベル

では、指定されたレプリケーション・アクションはできません。

ユーザーの処置: そのアーキテクチャー・レベルはサポートされていないので、キャプチャー・コントロール・サーバー上のコントロール表を手作業でドロップします。有効なアーキテクチャー・レベルのコントロール表を作成してください。

ASN1585E The Replication action *action_name* ended in error. The Apply control server Replication architecture level *arch_level* is not a valid architecture level.

説明: ASN.IBMSNAP_SUBS_SET にあるレプリケーション・アーキテクチャー・レベルでは、指定されたレプリケーション・アクションはできません。

ユーザーの処置: そのアーキテクチャー・レベルはサポートされていないので、アプライ・コントロール・サーバー上のコントロール表を手作業でドロップします。有効なアーキテクチャー・レベルのコントロール表を作成してください。

ASN1586W The DB2 table *tableowner.tablename* will be created in the DB2 default database.

説明: z/OS オペレーティング・システムの場合は、そこに指定された表を作成するデータベースが提供されませんでした。したがって、表は DB2 デフォルトを使用して作成されます。デフォルトの仕様が指定された表にとって適切でない場合、問題が起こる可能性があります。

ユーザーの処置: DB2 のデフォルトについては、「SQL リファレンス」を参照してください。表スペースを、独自のデータベースに入れたい場合は、適切な指定をしてレプリケーション・タスクを再実行します。デフォルトがその表にとって適切な場合は、アクションは必要ありません。

ASN1587E The value *value* for the parameter *parameter_name* of the DB2 object *db2object_name*, which has a type of *type*, is not valid.

説明: 提供された値が無効である、または別のパラメーター値と矛盾します。

ユーザーの処置: 有効な値については、「SQL リファレンス」を参照してください。

ASN1588E The value *encoding_scheme* provided for the parameter *encoding_scheme* is not valid for the DB2 server *server_name*.

説明: コード化スキームに提供された値が、サーバーの DB2 バージョンに有効なものではありません。スクリプトは生成されません。

ユーザーの処置: DB2 バージョンのコード化スキームとして有効な値は、「SQL リファレンス」を参照してください。

ASN1589W The calculation of the size of the table space container *container* of the table space *tspc* resulted in an incorrect container size. Therefore the container size has been changed to size *size* megabytes.

説明: 表スペース・コンテナのサイズの計算が、有効な表スペース・コンテナ定義に使用するには小さすぎる値になりました。DB2 が定義を受け入れることができるようにするため、レプリケーション固有の最小コンテナ・サイズを、表スペース・コンテナ定義に使用しました。

ユーザーの処置: 計算が現行のソース表サイズの % に基づくものである場合は、ソース表にデータが含まれているか、およびソース表の統計が最新のものかどうかを (RUNSTATS ユーティリティを使用して) チェックしてください。計算が行

数に基づくものである場合は、行の数が現実的なものかどうかをチェックしてください。

ASN1590E The DB2 table space *table_sp_name* is partitioned and in the DB2 *object_type* group. It should not be partitioned and it should be in the *object_type* IBMCATGROUP.

説明: 提供された表スペースはパーティション表スペースであり、DB2 カタログ・ノードまたはパーティション・グループに存在していません。パーティション表スペースにレプリケーション・コントロール表を作成することは、サポートされていません。スクリプトは生成されません。

ユーザーの処置: パーティションでない表スペースを指定してください。

ASN1600E The REMOTE SERVER *remote_servername* cannot be found.

説明: 指定されたリモート・サーバー名が、連合カタログ表 SYSIBM.SYSSERVERS の列 REMOTE_SERVER に見つかりません。この DB2 以外のリレーショナル・サーバーはアクセスできません。

ユーザーの処置: リモート・サーバー名に指定された入力进行检查して、アクションを再度呼び出すか、または連合カタログ表に適切に入れます。

ASN1601E The REMOTE AUTHID information for the REMOTE SERVER *remote_servername* cannot be found.

説明: 連合カタログ表 SYSIBM.SYSXXXXX に、指定された REMOTE_SERVER 値のリモート認証情報が見つかりません。この DB2 以外のリレーショナル・サーバーはアクセスできません。

ユーザーの処置: リモート・サーバー名に指定さ

れた入力进行检查して、アクションを再度呼び出すか、または連合カタログ表に適切に入れます。

ASN1602E The server *server_alias* does not support access to Federated servers.

説明: 連合レプリケーション関数は、DB2 UDB ワークステーション V8 以上でのみサポートされます。

ユーザーの処置: 指定されたデータベース・サーバーが上記にリストされたものの 1 つであることを確認します。またはサポートしないサーバーに対してレプリケーション・タスクを実行しないようにします。

ASN1603E The Replication Apply control server cannot reside on a non-IBM server.

説明: DB2 以外のリレーショナル・サーバーは、レプリケーション・キャプチャー・コントロール・サーバーまたはターゲット・サーバーにはできませんが、アプライ・コントロール・サーバーにはできません。

ユーザーの処置: DB2 サーバーをレプリケーション・アプライ・コントロール・サーバーとして指定します。

ASN1604E The remote table *remoteowner.tablename* exists in the non-IBM server, but the provided nickname *nicknameowner.nickname* cannot be found in the Federated server.

説明: 指定されたリモート表はリモート・データベースに存在しますが、対応するニックネームが連合データベースに見つかりません。

ユーザーの処置:

1. ニックネームの作成方法は、DB2 の連合の資料を参照してください。

2. 連合データベースにニックネームを作成します。
3. 再度、レプリケーション・タスクを実行します。

ASN1605E The nickname

nicknameowner.nickname exists in the Federated server but the remote table *remoteowner.remotetable* cannot be found in the non-IBM server.

説明: 指定されたりモート表のニックネームは存在しますが、対応するリモート表がリモート・データベースに存在しません。これは、レプリケーション定義の作成時の矛盾する定義の状態です。

ユーザーの処置:

1. ニックネームをドロップします。
2. 表のタイプにより、次のアクションを実行します。
 - 表がユーザー表の場合、リモート・サーバー内にリモート表を作成します。
 - 表がキャプチャー・コントロール・サーバー上のレプリケーション・コントロール表の場合、次のアクションを実行します。
 - a. キャプチャー・コントロール・サーバー上の既存のコントロール表からデータをコピーします。
 - b. キャプチャー・コントロール・サーバー上のコントロール表をドロップします。
 - c. キャプチャー・コントロール・サーバー上にコントロール表を作成します。
3. 連合サーバーにニックネームを作成します。
4. 再度、レプリケーション・タスクを実行します。

ASN1606W The nickname

nickname_owner.nickname_name exists in the Federated server but the remote table *table_owner.table_name* cannot be found in the non-IBM server.

説明: 指定されたりモート表のニックネームは存在しますが、対応するリモート表がリモート・データベースに存在しません。これは孤立したニックネームですが、この矛盾した状態は、レプリケーション定義をドロップする時は許容されます。スクリプトは生成されます。

ユーザーの処置: レプリケーション定義をドロップする時には、ソース・ニックネームはドロップされません。整合性のあるカタログにするには、ニックネームをドロップしてください。

ASN1607W It is strongly recommended to alter the nickname

nickname_owner.nickname_name defined for the Replication subscription target: to alter the local data type of column *column_name* from *existing_local_datatype* to *recommended_local_datatype* and ensure the proper source to target column data type mapping.

説明: ソース列のデータ・タイプと、これに対応するニックネーム・ターゲット列のデータ・タイプに不一致があります。これは DB2 の互換性規則に違反しませんが、ネイティブの IBM 以外のエンド・ユーザー・アプリケーションにとって問題の原因になる可能性があります。列データのレプリケーション中には問題は起こりません。問題は、エンド・ユーザー・アプリケーションがデータを検索した場合に起こります。たとえば、DB2 以外のリレーショナル・データ・タイプから DB2 データ・タイプへのデフォルト・マッピングを使用して、ニックネーム・データ・タイプを作成し

た場合、その列は最大範囲のデータ・タイプ値を持つことになり、より限定されたデータ・タイプの要件を持つエンド・ユーザー・アプリケーションとは合わない可能性があります。スクリプトは生成されます。

ユーザーの処置: ターゲットをチェックし、ターゲットで必要とするニックネームのデータ・タイプが、本当にソース列のデータ・タイプかどうかを確認します。そうであれば、'ALTER NICKNAME' ステートメントを実行し、ニックネーム列のローカル・データ・タイプを変更します。ニックネームのローカル・データ・タイプをソース列のデータ・タイプと同じものに変更すると、DB2 以外のリレーショナル・サーバー上のエンド・ユーザー・アプリケーションは、ソース列のデータ・タイプと同じデータ・タイプを強制的に見ることになります。

ASN1608I **The nickname *source_nickname* for the source and the nickname *ccd_nickname* for the Consistent Change Data table have a column data type that is altered: the local data type column *local_datatype* is set to *changed_datatype* because the remote data type is *remote_datatype*. Reason Code *reason_code*.**

説明: このメッセージは、CCD のニックネームの作成時に出されます。ニックネームは、適切なデータ・タイプの設定を保証するため、DB2 以外のリレーショナル・サーバーで作成された CCD のデータ・タイプに基づいて変更されます。これに失敗すると、レプリケーションの動作は不適切なものになります。スクリプトは生成され、ユーザー提供の定義を更新しました。

ユーザーの処置: レプリケーションの更新を受け入れられる場合は、アクションは必要ありません。

ASN1609E **The nickname *nicknameowner.nickname* exists in the Federated server but the remote table *remoteowner.remotetable* does not contain all the necessary columns.**

説明: ターゲット表のニックネームが存在し、サブスクリプションで要求された列のサブセットだけが含まれています。

ユーザーの処置: ターゲット表として別のニックネームを使用するか、または存在するニックネームの列と一致するようにサブスクリプションを変更します。

ASN1620E **Both Capture server control tables and Apply control server control tables already exist. Capture server control tables exist with architecture level *capturearch_level* and Capture Schema *capture_schema*. Apply control server control tables exist with architecture level *applyarch_level*.**

説明: 表 `captureschema.IBMSNAP_REGISTER` および `captureschema.IBMSNAP_SUBS_SET` が、指定されたサーバーにすでに存在します。

ユーザーの処置:

- 存在する `captureschema.IBMSNAP_REGISTER` コントロール表のアーキテクチャー・レベルが `0201` の場合:
 - 存在する `captureschema.IBMSNAP_REGISTER` に、すでに有効なレプリケーション定義が入っている場合は、キャプチャー・サーバー・コントロール表を `0801` アーキテクチャーにマイグレーションします。

- 表が空の場合は、V8 より前のキャプチャー・サーバー・コントロール表を単純にドロップし、レプリケーション・タスクを再度実行します。
- 存在するコントロール表のアーキテクチャー・レベルが 0201 でない場合は、別のキャプチャー・スキーマ名を持つキャプチャー・サーバー・コントロール表を作成することを考慮してください。
- 存在する IBMSNAP_SUBS_SET コントロール表のアーキテクチャー・レベルが 0201 の場合:
 - 存在する ASN.IBMSNAP_SUBS_SET 表に、すでに有効なレプリケーション定義が入っている場合は、アプライ・コントロール・サーバー・コントロール表を 0801 アーキテクチャーにマイグレーションします。
 - 表が空の場合は、V8 より前のアプライ・コントロール・サーバー・コントロール表を単純にドロップし、レプリケーション・タスクを再度実行します。
- 存在するコントロール表のアーキテクチャー・レベルが 0201 でない場合は、別のサーバーにアプライ・コントロール・サーバーのコントロール表を作成することを考慮してください。

ASN1621W At least one row was found in the control table
controlowner.controltable. Dropping of this control table will result in dropping of all Replication definitions stored in the table.

説明: ドロップ対象のコントロール表が空ではありません。生成されたスクリプトを実行すると、レプリケーション・コントロール情報は削除されます。

ユーザーの処置: 次の依存関係について問題がないことを確認した上で、生成されたスクリプトを実行してください。

- キャプチャー・コントロール・サーバーからコントロール表をドロップすることにより、これ

に従属する既存のサブスクリプション・セットに影響がないことを確認します。

- アプライ・コントロール・サーバーからコントロール表をドロップすることにより、これに従属する既存のサブスクリプション・セット (multi-tier シナリオの場合) に影響がないことを確認します。
- これらの定義に対して、レプリケーションのキャプチャー処理またはアプライ処理をもう実行する必要がないことを確認します。

アーキテクチャー・レベルが 0201 の場合は、コントロール表をドロップする前に、キャプチャーまたはアプライ・コントロール・サーバーのコントロール表を 0801 アーキテクチャーにマイグレーションします。

ASN1622E The Replication action
action_name ended in error. The required control table
controlowner.controltable could not be found.

説明: レプリケーション定義はレプリケーション・コントロール表に保管されます。これらの表は、登録の前に、またはサブスクリプション定義を作成する前に、存在している必要があります。IBMSNAP_REGISTER 表の存在は、キャプチャー・コントロール・サーバーのコントロール表が特定のキャプチャー・スキーマにすでに存在しているかどうかをチェックするために使用されます。IBMSNAP_SUBS_SET 表の存在は、アプライ・コントロール・サーバーのコントロール表がすでに存在しているかどうかをチェックするために使用されます。IBMSNAP_SUBS_MEMBR の存在は、サブスクリプション・メンバーの存在をチェックする時にチェックされます。

ユーザーの処置: コントロール表 IBMSNAP_SUBS_MEMBR 表が存在しない場合、その環境は整合性のとれていない状態にあります。アプライ・コントロール・サーバーからすべてのコントロール表をドロップし、その後、これ

らを作成してから、アクションを実行する必要があります。

あるいは、コントロール表 `IBMSNAP_REGISTER` または `IBMSNAP_SUBS_SET` が存在しない場合、これらを作成してから、登録またはサブスクリプション定義をコントロール・サーバーに追加します。これ以外の場合、以下のことができます。

1. 登録関係のアクションを行なう場合、該当のキャプチャー・スキーマが提供されているかどうか、または該当のキャプチャー・コントロール・サーバーが入力として提供されているかどうか、をチェックします。
2. サブスクリプション関係のアクションを行なう場合、該当のアプライ・コントロール・サーバーが入力として提供されているかどうかをチェックします。
3. ターゲット・サーバー側で自動登録する必要があるターゲット表 (CCD またはレプリカ) を含むサブスクリプション・セットを作成する場合は、サブスクリプション・ターゲット・サーバー側に、キャプチャー・コントロール・サーバーの該当のコントロール表があることをチェックします。

ASN1623W The Replication control table, *controlowner.controltable* could not be found and is not dropped.

説明: `Drop Capture control tables` または `Drop Apply control server control tables` アクションが実行され、コントロール表が欠落しています。スクリプトは、そのコントロール表の該当の `DROP` ステートメントを生成しません。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1624I The server *server_alias* is not a known Replication Capture server for *capture_schema*.

説明: `captureschema.IBMSNAP_REGISTER` 表が見つかりません。該当のキャプチャー・サーバー・コントロール表 (`IBMSNAP_REGISTER` 表を含む) がサーバー上に存在している場合は、サーバーはレプリケーション・キャプチャー・サーバーとして定義されています。

ユーザーの処置: 必要ならば、該当のキャプチャー・サーバー・コントロール表を作成します。

ASN1625I The server *server_alias* is not a known Replication Apply control server.

説明: `ASN.IBMSNAP_SUBS_SET` 表が見つかりません。該当のアプライ・コントロール・サーバー・コントロール表 (`IBMSNAP_SUBS_SET` 表を含む) がサーバー上に存在している場合は、サーバーはレプリケーション・アプライ・コントロール・サーバーとして定義されています。

ユーザーの処置: 必要ならば、アプライ・コントロール・サーバー上に該当のコントロール表を作成します。

ASN1626E Capture server control tables already exist for architecture level *arch_level* with the same Capture schema.

説明: 表 `captureschema.IBMSNAP_REGISTER` が指定されたサーバーにすでに存在します。

ユーザーの処置:

- 存在する `captureschema.IBMSNAP_REGISTER` 表のアーキテクチャー・レベルが `0801` の場合、次のオプションを考慮してください。
 - 表は同じキャプチャー・スキーマですすでに存在しているので、コマンドを実行する必要はありません。

- 別のキャプチャー・スキーマの下でコマンドを実行します。
- 存在する `captureschema.IBMSNAP_REGISTER` コントロール表のアーキテクチャー・レベルが `0201` の場合:
 - 存在する `captureschema.IBMSNAP_REGISTER` に、有効なレプリケーション定義がすでに入っている場合は、キャプチャー・コントロール・サーバーのコントロール表を `0801` アーキテクチャーにマイグレーションします。
 - コントロール表が空の場合は、V8 より前のキャプチャー・サーバー・コントロール表を単純にドロップし、レプリケーション・タスクを再度実行します。

これ以外のアーキテクチャー・レベルは無効です。表を作成する前に、表を手作業でドロップする必要があります。

ASN1627E Some Capture server control tables already exist with the same Capture Schema but for which an architecture level cannot be determined.

説明: 指定されたサーバーに、他のキャプチャー・サーバー・コントロール表は見つかりましたが、`captureschema.ASN.IBMSNAP_REGISTER` 表が存在しません。キャプチャー・サーバー・コントロール表は、表をドロップしないと作成できません。キャプチャー・サーバー側のレプリケーション定義は整合性のとれていない状態にあります。

ユーザーの処置: 残っているキャプチャー・サーバー・コントロール表をドロップしてキャプチャー・コントロール・サーバーの定義をきれいにし、コントロール表作成 (Create control table) タスクを再度実行します。ドロップ・タスクを実行するとデータが失われるので、タスクを実行する前に残っているコントロール表の内容を確認してください。

ASN1628E The Capture server control tables are not at the architecture level requested.

説明: 指定されたアーキテクチャー・レベルを持つ `captureschema.IBMSNAP_REGISTER` 表が存在しません。スクリプトは生成されません。

ユーザーの処置: 該当のキャプチャー・コントロール・サーバーとキャプチャー・スキーマ用の、適切なアーキテクチャー・レベルで、レプリケーション・タスクを再度実行してください。

ASN1629E No Capture server control tables were found for the provided Capture schema.

説明: キャプチャー・コントロール・サーバー上にコントロール表がありません。コントロール表はドロップされず、スクリプトは生成されません。

ユーザーの処置: 該当のキャプチャー・コントロール・サーバーとキャプチャー・スキーマ用の、適切なアーキテクチャー・レベルで、レプリケーション・タスクを再度実行してください。

ASN1630W Some Capture server control tables already exist with Capture Schema *capture_schema* but their architecture level cannot be determined. The Replication action *action_name* for the provided architecture level *arch_level* and Capture Schema will drop control tables that might not belong to the architecture level provided.

説明: キャプチャー・サーバー上に `captureschema.IBMSNAP_REGISTER` 表が存在しません。レプリケーション・アーキテクチャー・レベルは不明であり、誤ったアーキテクチャー・レベルを指定すると、重要なデータを失う可能性が

あります。特定のキャプチャー・サーバー・コントロール表のアーキテクチャー・レベルを類推して判断するようなチェックは行なわれません。コントロール表が存在する場合はドロップされず。スクリプトは生成されます。

ユーザーの処置: DB2 レプリケーションの該当のアーキテクチャー・レベルを使用して、再度タスクを実行してください。

ASN1631E Apply control server control tables already exist for architecture level *arch_level*.

説明: 表 ASN.IBMSNAP_SUBS_SET が指定されたサーバーにすでに存在します。スクリプトは生成されません。

ユーザーの処置: 存在する

ASN.IBMSNAP_SUBS_SET コントロール表のアーキテクチャー・レベルが 0201 の場合:

- 存在する ASN.IBMSNAP_SUBS_SET 表に、有効なレプリケーション定義がすでに入っている場合は、アプライ・コントロール・サーバー・コントロール表を 0801 アーキテクチャーにマイグレーションします。
- 表が空の場合は、V8 より前のアプライ・コントロール・サーバー・コントロール表を単純にドロップし、レプリケーション・タスクを再度実行します。

これ以外のアーキテクチャー・レベルは無効です。表を作成する前に、表を手作業でドロップする必要があります。

ASN1632E Some Apply control server control tables already exist but for which an architecture level cannot be determined.

説明: 指定されたサーバーに、他のアプライ・コントロール・サーバー・コントロール表は見つかりましたが、ASN.IBMSNAP_SUBS_SET 表が存在しません。アプライ・コントロール・サーバー・コントロール表は、表をドロップしないと作

成できません。アプライ・コントロール・サーバー側のレプリケーション定義は整合性のとれていない状態にあります。スクリプトは生成されません。

ユーザーの処置: アプライ・コントロール・サーバーに残っているコントロール表をドロップし、アプライ・コントロール・サーバーのレプリケーション定義をきれいにします。 *Create control table* タスクを再度実行します。 *Drop* タスクを実行するとデータが失われるので、タスクを実行する前に残っているコントロール表の内容を確認してください。

ASN1633E The Apply control server control tables are not at the architecture level requested.

説明: 指定されたアーキテクチャー・レベルを持つ ASN.IBMSNAP_SUBS_SET 表が存在しません。スクリプトは生成されません。

ユーザーの処置: 該当のアプライ・コントロール・サーバー用の適切なアーキテクチャー・レベルで、レプリケーション・タスクを再度実行してください。

ASN1634E No Apply control server control tables were found.

説明: アプライ・コントロール・サーバーからドロップするコントロール表がありません。スクリプトは生成されません。

ユーザーの処置: 該当のアプライ・コントロール・サーバー用の適切なアーキテクチャー・レベルで、レプリケーション・タスクを再度実行してください。

ASN1635W Some Apply control server control tables already exist but their architecture level cannot be determined. The Replication action *action_name* for the provided architecture level *arch_level* will drop control tables that might not belong to the architecture level provided.

説明: アプライ・コントロール・サーバー上に、ASN.IBMSNAP_SUBS_SET 表が存在しません。レプリケーション・アーキテクチャー・レベルは不明であり、誤ったアーキテクチャー・レベルを指定すると、重要なデータを失う可能性があります。特定のアプライ・コントロール・サーバー・コントロール表のアーキテクチャー・レベルを類推して判断するようなチェックは行なわれません。コントロール表が存在する場合はドロップされます。スクリプトは生成されます。

ユーザーの処置: DB2 レプリケーションの該当のアーキテクチャー・レベルを使用して、再度タスクを実行してください。

ASN1636E The Replication Action of Manual Full Refresh ended with an error for the Apply qualifier *apply_qual* and set name *set_name*. The synchpoint in the *capschema.IBMSNAP_PRUNCNTL* table for the source member *sourceowner.sourcetable* and the target member *targetowner.target_table* is not translated by the Capture program.

説明: 同期点が 0 より小さいか、または 16 進のゼロです。

ユーザーの処置: 16 進 ゼロを変換するロード前スクリプトを実行し、サーバーでキャプチャーが実行されていることを確認してください。

ASN1637E The replication action 'Manual Full Refresh' ended in error for the Apply qualifier *apply_qualifier* and the set name *set_name*. The target structure of at least one of the target subscription-set members in the given subscription set is greater than eight. None of the subscription-set members is eligible for a manual full refresh.

説明: 指定されたサブスクリプション・セットのターゲット・サブスクリプション・セット・メンバーの、少なくとも 1 つが 8 より多いターゲット構造を持っています。手動フル・リフレッシュは、8 より多いターゲット構造をサポートしません。

ユーザーの処置: サブスクリプション・セット・メンバーのターゲット構造が 8 と等しいかこれ以下であることを確認し、レプリケーション・タスクを再度実行してください。

ASN1638W The subscription-set member with a target of *targetowner.targetname* and a source of *sourceowner.sourcename* is not complete. This subscription-set member is not included in the manual full refresh.

説明: 手動フル・リフレッシュは完全なターゲットだけをサポートします。指定されたサブスクリプション・セット・メンバーは完全でないので、含めることができません。

ユーザーの処置: アクションは必要ありません。

ASN1639E The replication action 'Manual Full Refresh' ended in error for the Apply qualifier *apply_qualifier* and the set name *set_name*.
None of the target subscription-set members in the given subscription set is complete or eligible for a manual full refresh.

説明: 手動フル・リフレッシュは完全なターゲットだけをサポートしますが、ターゲットは完全ではありません。

ユーザーの処置: サブスクリプション・セットのサブスクリプション・セット・メンバーの少なくとも 1 つが完全であることを確認し、レプリケーション・タスクを再度実行します。

ASN1640E The replication action ended in error for the Apply qualifier *apply_qualifier* and the set name *set_name*. There are no subscription-set members in the subscription set.

説明: サブスクリプション・セットにサブスクリプション・セット・メンバーが含まれていません。

ユーザーの処置: サブスクリプション・セットに少なくとも 1 つのサブスクリプション・セット・メンバーを追加し、レプリケーション・タスクを再度実行します。

ASN1641E The replication action *action_name* ended in error. This action on an OS/400 system is supported only through OS/400 commands.

説明: レプリケーション・センターも、コマンド行も、OS/400 システム上のレプリケーション・アクションをサポートしません。可能なアクション

は、creating capture server control tables (キャプチャー・サーバー・コントロール表の作成)、creating apply server control tables (アプライ・サーバー・コントロール表の作成)、dropping capture server control tables (キャプチャー・サーバー・コントロール表のドロップ)、またはdropping apply server control tables (アプライ・サーバー・コントロール表のドロップ) です。

ユーザーの処置: レプリケーション・アクションを実行するには、OS/400 コマンドを実行します。

ASN1650I The replication action *action_name* started at *timestamp*.
The monitor server is *server_name* and the *Group_or_Contact* name is *group_name_or_contact_name*.

説明: 指定されたモニター・サーバーでレプリケーション・アクションが開始しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1653I The replication action *action_name* for *group_contact_or_condition_name* ended successfully at *timestamp*.
The monitor server is *server_name*.

説明: 指定されたモニター・サーバーでレプリケーション・アクションが正常に終了しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1654E The replication action
action_name ended in error. The
length of the input parameter
parameter-name, parameter_length
exceeds the limit *maximum-limit*.

説明: 指定された入力パラメーターの長さが、可能な最大長より長い値です。スクリプトは生成されません。

ユーザーの処置: 入力パラメーター値を検査し、パラメーター値を再入力してください。

ASN1655E The replication action
action_name ended in error. The
value *input_value* of the input
parameter *input_parameter* is
incorrect.

説明: 指定された入力パラメーターの値が正しくありません。

ユーザーの処置: 有効なパラメーター値を資料で参照してください。

ASN1656E The replication action
action_name ended in error. The
value of the input parameter
input_parameter is missing.

説明: 指定された入力パラメーターの値は、このアクションに必須です。その値が欠落しています。スクリプトは生成されません。

ユーザーの処置: この必須入力パラメーターに値を指定し、レプリケーション・アクションを再実行します。

ASN1657E The replication action
action_name ended in error. At
least one optional parameter
value must be specified.

説明: 各パラメーター値がオプションであるコマンドを実行する場合、少なくとも1つのオプション

・パラメーター値を指定する必要があります。スクリプトは生成されません。

ユーザーの処置: 正しいパラメーターを使用して、コマンドを再度実行します。

ASN1658E The replication action
action_name ended in error. The
value *value1* of the input
parameter *input_parameter1* must
be different than the value
value2 of the input parameter
input_parameter2.

説明: ある入力パラメーターの値が、別の入力パラメーターの値と同じであり、矛盾する定義を作成する結果になります。スクリプトは生成されません。

ユーザーの処置: 正しいパラメーター値を使用して、コマンドを再度実行します。

ASN1659E The replication action
action_name ended in error. The
contact *contact-name* already
exists.

説明: 指定された連絡先の名前は、ASN.IBMSNAP_CONTACTS 表の行の1つにすでに存在します。連絡先名は固有である必要があります。スクリプトは生成されません。

ユーザーの処置: 別の連絡先名を使用して、コマンドを再度実行します。

ASN1660E The replication action
action_name ended in error. The
contact *contact-name* does not
exist.

説明: 指定された連絡先の名前が、ASN.IBMSNAP_CONTACTS 表の行にありません。連絡先名を変更、置換、委任、またはドロップするには、ASN.IBMSNAP_CONTACTS 表に

その名前が存在する必要があります。スクリプトは生成されません。

ユーザーの処置: 別の連絡先名を使用して、コマンドを再度実行します。

ASN1661E The replication action *action_name* ended in error. The contact *contact-name* cannot be dropped, because dropping the contact empties each associated group.

説明: グループは、少なくとも 1 つの連絡先を持つ必要があります。指定された連絡先は、それぞれの関連グループ内の最後の連絡先であり、最後の連絡先はドロップできません。スクリプトは生成されません。

ユーザーの処置: 各関連グループをドロップしてから、連絡先をドロップします。

ASN1662E The replication action *action_name* ended in error. The contact *contact-name* cannot be dropped, because the contact is associated with one or more conditions.

説明: ドロップ対象の連絡先名は、キャプチャーまたはアプライ・コンポーネントの条件に関連付けられた、唯一の連絡先です。スクリプトは生成されません。

ユーザーの処置: DROP CONTACT コマンドの SUBSTITUTE オプションを使用するか、または SUBSTITUTE コマンドを使用して、条件の連絡先名を CHANGE します。条件が必要ない場合は、条件をドロップしてから連絡先をドロップします。

ASN1663E The replication action *action_name* ended in error. The value *startdate_value* that is specified for the start date is greater than the value *enddate_value*, which is specified for the end date.

説明: 終了日より後の開始日を入力することはできません。スクリプトは生成されません。

ユーザーの処置: 有効な日付の組み合わせを使用して、コマンドを再度実行します。

ASN1664E The replication action *action_name* ended in error. The group *group-name* already exists.

説明: 指定されたグループ名は、ASN.IBMSNAP_GROUPS 表の行の 1 つにすでに存在します。グループ名は固有である必要があります。

ユーザーの処置: グループ名を変更し、コマンドを再度実行します。

ASN1665E The replication action *action_name* ended in error. The group *group_name* does not exist.

説明: 指定されたグループ名が、ASN.IBMSNAP_GROUPS 表の行にありません。グループ名を変更またはドロップするには、ASN.IBMSNAP_GROUPS 表にその名前が存在する必要があります。スクリプトは生成されません。

ユーザーの処置: グループ名を検査し、コマンドを再度実行します。

ASN1666E The replication action *action_name* ended in error. The group *group_name* cannot be dropped because it is associated with one or more conditions.

説明: ドロップ対象のグループは、キャプチャーまたはアプライ・コンポーネントの条件に関連付けられた、唯一のグループです。スクリプトは生成されません。

ユーザーの処置: グループをドロップするには、関連する条件の連絡先を変更し、コマンドを再実行します。

ASN1667E The replication action *action_name* ended in error. The contact *contact-name* is not associated with the specified group *group_name*.

説明: ドロップ対象の連絡先名は、指定されたグループと関連付けられていません。

ユーザーの処置: 指定した連絡先名を検査し、コマンドを再度実行します。

ASN1668E The replication action *action_name* ended in error. The contact *contact-name* is already associated with the specified group *group_name*.

説明: 指定された連絡先名は、指定されたグループとすでに関連付けられています。

ユーザーの処置: アクションは必要ありません。

ASN1671E The replication action *action_name* ended in error. The alert condition *condition-name* already exists for the monitor qualifier *mon-qual*, the server *server-name*, the schema or qualifier *schema-or-qualifier*, and the subscription-set name *set-name*.

説明: 作成するアラート条件は、同一のパラメーター指定で、モニター・コントロール・サーバー上にすでに存在します。

ユーザーの処置: このアラート条件を検査し、コマンドを再度実行します。

ASN1672E The replication action *action_name* ended in error. The alert condition *condition-name* does not exist for the monitor qualifier *mon-qual*, the server *server-name*, the schema or qualifier *schema-or-qualifier*, and the subscription-set name *set-name*.

説明: ドロップまたは変更するアラート条件は、モニター・コントロール・サーバー上に存在しません。

ユーザーの処置: アラート名を検査し、コマンドを再度実行します。

ASN1673W The condition *condition_name* is valid only at the apply qualifier level.

説明: 条件名をサブスクリプション・セット名の値と一緒に指定することは無効です。サブスクリプション・セットの名前は無視されます。

ユーザーの処置: サブスクリプション・セット名の値を指定しないでください。

ASN1674W The condition *condition_name* is valid only with update-anywhere subscription sets.

説明: 条件名は、「update-anywhere」サブスクリプション・セットにのみ有効です。

ユーザーの処置: この条件をセットしないでください。この条件は無視されます。

ASN1675I This is a test message from the Replication Center.

説明: このメッセージは、連絡先に入力された Eメール・アドレスを検査するためのテスト用 Eメール送信に使用されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1677E The replication action *action_name* ended in error. The apply qualifier *apply-qual* and the subscription-set name *set-name* do not exist on the server *server-name*.

説明: アプライ修飾子およびサブスクリプション・セット名が、指定されたアプライ・コントロール・サーバーの ASN.IBMSNAP_SUBS_SET 表にありません。

ユーザーの処置: 有効なアプライ修飾子と有効なサブスクリプション・セット名を指定します。

ASN1678E The replication action *action_name* ended in error. The capture schema *cap-schema* does not exist on the server *server-name*.

説明: 指定されたキャプチャー・コントロール・サーバーの ASN.IBMSNAP_CAPSCHEMAS 表にキャプチャー・スキーマがありません。

ユーザーの処置: 有効なキャプチャー・スキーマを指定してください。

ASN1679E The replication action *action_name* ended in error. The contact *contact_name* that you attempted to substitute is not associated with a condition.

説明: 連絡先名が ASN.IBMSNAP_CONDITIONS 表にありません。ASN.IBMSNAP_CONDITIONS 表に連絡先がある場合のみ、これを置き換えることができます。スクリプトは生成されません。

ユーザーの処置: 有効な連絡先名を指定してください。

ASN1680I The Replication action *action_name* started at time. The Monitor server is *server_name*.

説明: このメッセージは単なる通知です。

ユーザーの処置: アクションは必要ありません。

ASN1681E The Replication action ended in error. Monitor control tables already exist for Architecture *Levelarch_level*.

説明: モニター・サーバーにモニター・コントロール表がすでに存在します。

ユーザーの処置: 存在するモニター表のアーキテクチャー・レベルが 0801 の場合、表はすでに存在するので、コマンドを実行する必要はありません。

ASN1682E The Replication action ended in error. No Monitor control tables were found.

説明: ドロップするモニター・コントロール表がありません。スクリプトは生成されません。

ユーザーの処置: モニター・コントロール表を含

む該当のサーバーに、レプリケーション・タスクを再度実行します。

ASN1700E The column *tableowner.tablename.columnname* of data type *data_type* cannot be included in the registration.
Reason Code *reason_code*.

説明: この列は、定義されているように、レプリケーション・キャプチャーのメカニズムではサポートできません。指定された列の登録用スクリプトは生成されません。有効な理由コードは次の値です。

- 0 データ・タイプがサポートされていません。
- 1 列がすでに登録されています。
- 2 z/OS fieldproc 列。
- 3 この列は変更前イメージ列として適格ではありません。
- 4 このデータ・タイプは 連合 DB2 ではサポートされません。
- 5 列がソース・オブジェクトにありません。
- 6 その表に登録された LOB 列の最大数を超えます。
- 7 列名は変更前イメージ接頭部で始まりません。
- 8 この列は変更前イメージ列または変更後イメージ列として適格ではありません。

ユーザーの処置: 理由コードをチェックし、列を登録できない理由を調べてください。追加情報または制約事項については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」を参照してください。

ASN1701E The provided locksize value *lock_size* for the given table space *tablespace_name* is not valid.

説明: z/OS オペレーティング・システムの場合、Locksize は P(PAGE)、R(ROW) または A(ANY) にする必要があります。

ユーザーの処置: 正しい locksize を指定し、アクションを再度サブミットします。

ASN1702W Replication definitions for the registered column *objectowner.objectname.columnname* has been changed to support null values.

説明: NULL 値をサポートするには、変更前イメージ列が必要です。変更前イメージ列の値がない場合、INSERT ステートメントは失敗します。ユーザー提供の定義を更新するスクリプトが生成されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1703E The table *tableowner.tablename* cannot be registered for change-capture replication.
Reason code *reason_code*.

説明: この表は、定義されているように、レプリケーション・キャプチャーのメカニズムではサポートできません。スクリプトは生成されません。有効な理由コードは次の値です。

- 0 z/OS validproc を持つ表。
- 1 既存の内部 CCD 表。
- 2 既存の CD 表。
- 3 DB2 カタログ表 (Windows、UNIX、iSeries)
- 4 表はすでに登録済みです。

- 5 内部 CCD 表のソースが登録されたソースではありません。
- 6 ソースは CD 表であり、登録できません。
- 7 このソース名はこのセッションで重複しています。
- 8 ソースはレプリケーション・コントロール表です。
- 9 ソース列に登録に適切なものはありません。
- 10 この表に登録された LOB 列の最大数を超えます。
- 11 構造化データ・タイプはサポートされません。
- 12 変更前イメージ接頭部は 1 文字だけです。
- 13 内部エラーが起きました。

ユーザーの処置: 理由コードをチェックし、表を変更キャプチャー・レプリケーション用に登録できない理由を調べてください。追加情報および制約事項については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」を参照してください。

ASN1704E The view *viewowner.viewname* cannot be registered. Reason code *reason_code*.

説明: このビューは、定義されているように、レプリケーション・キャプチャーのメカニズムではサポートできません。スクリプトは生成されません。有効な理由コードは次の値です。

- 0 ビューの従属表が 1 つも定義されていません。
- 1 ビューが従属するソース表の列が登録されていません。
- 2 ビューが内部 ccd 上にあります。
- 3 ビューがすでに登録されています。

- 4 ビューが 'OUTER JOIN' 構文を持っています。
- 5 ビューに、関数を持つ複数の表の列またはビューの列が含まれ、ビュー定義に各表のための相関が提供されていません。
- 6 ビューに集合関数への参照が含まれています。
- 7 ビューに副選択 / 副照会が含まれていません。
- 8 ビューに別のビューへの参照が含まれています。
- 9 ビューに UNION があります。
- 10 列に相関が提供されていません。
- 11 基本表にスキーマ名がありません。
- 12 基本表がありません。
- 13 ビューに表として、表の式が含まれています。
- 14 従属表が存在しません。
- 15 ビューのビューは登録できません。
- 16 指定されたソース・オブジェクトはビューではありません。
- 17 このソース・ビューはこのセッションで重複しています。
- 18 このビュー定義はサポートできません。
- 19 このビューは、ビュー定義に特定の列名ではなくアスタリスク (*) を持っています。
- 20 ビューに CCD と CCD 以外の表の結合が含まれています。

ユーザーの処置: 理由コードをチェックし、ビューを登録できない理由を調べてください。追加情報および制約事項については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」を参照してください。

ASN1705E The change data object, objectowner.objectname already exists in the server.

説明: 変更データ表またはビューを、登録する現行ソースとして使用できません。その理由は、キャプチャー・サーバーにすでに存在するためです。スクリプトは生成されません。

ユーザーの処置: 変更データ・オブジェクトに別の名前を指定してください。

ASN1706W A column column_name is added to a registered source sourceowner.sourcename. The registered source maintains an internal CCD table. The new column has to be first added to the CCD table subscription member before adding to any existing or not-yet existing subscription member.

説明: 従属サブスクリプション・セットに新しい列が必要な場合、必要なサブスクリプション・セット・メンバーにその列を追加する前に、まず、内部 CCD サブスクリプション・メンバーにその列を追加する必要があります。

ユーザーの処置: 変更データ・オブジェクトに別の名前を指定してください。

ASN1707W The Replication action Alter Registration for sourceowner.sourcename is not in effect until a Capture REINIT command is issued at the Capture server.

説明: 登録されたソースは正常に更新されました。ただし、キャプチャー・プログラムは、REINIT コマンドでそれを強制されるまで、対応する captureschema.IBMSNAP_REGISTER 表の更新を認識しません。スクリプトは生成されます。

スクリプトの効果を実行するには、後でキャプチャー・コマンドが必要です。

ユーザーの処置: 変更を即時に有効にするには、

1. 生成されたスクリプトを実行します。
2. 該当のキャプチャー・スキーマの該当のキャプチャー・プログラムの REINIT を実行します。

ASN1708E The table, view or nickname objectowner.objectname is not a Replication registered source.

説明: 上に指定されたレプリケーション・オブジェクトは、レプリケーション・コントロール表に定義されていません。スクリプトは生成されません。

ユーザーの処置: コマンドにオブジェクトが正しく指定されていること、およびそれが存在することを確認します。

ASN1709W Associated subscription sets will not be valid after the registered source sourceowner.sourcename is dropped.

説明: サブスクリプション・メンバーは、ソース・メンバーを定義している、基礎にあるソース登録に依存しています。登録されたソース表をドロップした場合、サブスクリプション・セットの従属ソース・メンバーはもはや無効です。指定された登録ソースに関連付けられたサブスクリプション・セットが、キャプチャー・コントロール・サーバーの captureserver.IBMSNAP_PRUNCNTL 表にあります。ここで、SOURCE_OWNER および SOURCE_TABLE は、ドロップされる登録済みソースに対応します。該当のアプライ・コントロール・サーバーおよびサブスクリプション・セット名は、IBMSNAP_PRUNCNTL 表内の列です。アプライが実行されている場合は、関連するサブスクリプション・セットは失敗します。スクリプトは生成されます。

ユーザーの処置: 登録されたソースが従属サブスクリプション・セットを持つ場合は、スクリプトを実行する前に、従属サブスクリプション・セットを非活性化またはドロップします。

ASN1710W Dependent view registered sources will not be valid after the registered source
sourceowner.sourcetable is dropped.

説明: ビュー登録は、ビュー定義を構成する表の、基礎になる登録に依存しています。登録されたソース表をドロップした場合、その表に基づくビュー登録はすべて無効になります。影響を受ける可能性のあるビューは、キャプチャー・サーバーの `captureserver.IBMSNAP_REGISTER` 表にあります。ここで、`PHYS_CHANGE_OWNER` および `PHYS_CHANGE_TABLE` は、ドロップされる登録済みソースの `CD_OWNER` および `CD_TABLE` と同じです。アプライが実行されている場合、ビュー登録に依存する、関連するサブスクリプション・セットは失敗します。スクリプトは生成されません。

ユーザーの処置: 登録されたソースが従属ビュー登録を持つ場合は、スクリプトを実行する前に、該当のサブスクリプション・セットまたはビュー登録を、非活性化またはドロップします。

ASN1711W The source
sourceowner.sourcename is still active so dropping it will result in a Capture failure.

説明: アクティブな登録が、NULL でない `SYNCHPOINT` 値を `captureschema.IBMSNAP_REGISTER` 表に持っています。キャプチャー・プログラムが開始すると、すべてのアクティブな登録は必ず存在し、かつ有効であると想定します。したがって、登録されたソースをドロップした場合は、キャプチャー・プログラムにその旨を知らせる必要があります。なぜなら、ドロップ・アクションは登録情報を無効

にするからです。この情報をキャプチャー・プログラムに提供できないと、キャプチャー・プログラムが失敗する原因になります。スクリプトは生成されますが、実行する準備はできていません。

ユーザーの処置:

1. 該当の登録を非活性化します (レプリケーション・センター GUI を使用するか、または `STOP` シグナルおよび `CMD` タイプのコマンドを実行)。
2. `captureschema.IBMSNAP_SIGNAL` 表の `SIGNAL_STATE` の完了を待ちます。
3. 登録をドロップするスクリプトを実行します。

ASN1712E The table, view, or nickname
objectowner.objectname is not a valid Replication registered source. Reason code
reason_code.

説明: キャプチャー・サーバー・コントロール表のこの登録ソースに、矛盾する情報が見つかりました。スクリプトは生成されません。

ユーザーの処置: 登録されたソースをドロップし、登録を再度作成してください。

ASN1713E The registered source
sourceowner.sourcename cannot be deactivated. Reason code
reason_code.

説明: 有効な理由コードは次の値です。

- 0 ソースは `FULL REFRESH` として登録されているので、非活性化することはできません。
- 1 ソースは `CCD` であり、`CCD` 登録を非活性化することはできません。
- 2 ソースはビューであり、ビュー登録を非活性化することはできません。

ユーザーの処置: このメッセージは通知のみであ

り、アクションは必要ありません。

ASN1714E The registered source
sourceowner.sourceName **cannot be**
altered. Reason code
reason_code.

説明: 有効な理由コードは次の値です。

- 0** このソースの CD 表は RRN 列を持っています (iSeries のみ)。RRN は表の最後の列である必要があるため、ソースは変更できません。
- 1** ソースはビューであり、ビュー登録を変更することはできません。
- 2** ソースはフル・リフレッシュ用に登録されているので、変更できません。
- 3** ソース表の列が変更される列と一致しません。
- 4** 列は、LOB、DATALINK、または ROWID データ・タイプであり、変更前イメージ値には適格ではありません。
- 5** 変更前イメージ列の値は NULL にすることはできません。
- 6** 指定された列の変更後イメージの値が登録されていません。
- 7** 変更前イメージの接頭部は、登録された既存のソースで使用されている場合は更新できません。
- 8** 現行の変更前イメージ接頭部を使用すると、この登録されたソース内の列の 1 つが未確定になります。
- 9** 変更前イメージ接頭部は 1 文字だけです。
- 10** 内部エラーが起きました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1715E The Replication action ended in error. The native OS/400 message is
as400native_message.

説明: OS/400 オペレーティング・システムまたは iSeries サーバーに該当コマンドを実行中にエラーが起きました。スクリプトは生成されません。

ユーザーの処置: 詳細なエラー情報は OS/400 コンソール・ログを参照してください。

ASN1716W The Replication action ended with a warning. The native OS/400 message is
as400native_message.

説明: OS/400 オペレーティング・システムまたは iSeries サーバーに該当コマンドを実行中に警告が出されました。スクリプトは生成されます。

ユーザーの処置: 詳細な警告情報は iSeries コンソール・ログを参照してください。

ASN1717I The Replication action ended with an Informational Clause. The native OS/400 message is
as400native_message.

説明: OS/400 オペレーティング・システムまたは iSeries サーバーに該当コマンドを実行中に、通知メッセージが出されました。スクリプトは生成されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1718E The nickname
nicknameowner.nickname **cannot be**
registered. Reason code
reasoncode.

説明: このニックネームは、定義されているように、レプリケーション・キャプチャーのメカニズムではサポートされません。スクリプトは生成さ

れません。有効な理由コードは次の値です。

- 0 内部 CCD 表 (ユーザーの cd 表) がすでに存在します。
- 1 ニックネームがネイティブのカタログ表にあります。
- 2 ニックネームがすでに登録されています。
- 3 連合登録は、ニックネームを、登録されるソースであると予想します。
- 4 キャプチャー・プログラム用に適格となる列がありません。
- 5 提供されたニックネームは前の登録と重複していますが、対応するスクリプトが実行されていません。
- 6 連合登録は、ユーザー表のみをサポートします。
- 7 連合登録は、コンデンスされていない、完全でない CCD 表のみをサポートしません。
- 8 提供された CCD ニックネームは、前の登録の CCD ニックネームと重複していますが、その登録のスクリプトはまだ実行されていません。

ユーザーの処置: 理由コードをチェックし、ニックネームを登録できない理由を調べてください。追加の説明または制約事項については、オンライン・ヘルプを参照してください。

ASN1719W The non-IBM triggers that were defined for registered source *nicknameowner.nickname* will be dropped. Any additional logic later provided by users in these triggers will be lost.

説明: 登録されたソースをドロップすることは、ソースの登録中に作成されたすべてのオブジェクトを、後から更新したかどうかに関係なく、すべ

てドロップすることを意味します。スクリプトは生成されます。

ユーザーの処置: 必要ならば、登録されたソースをドロップする前に、トリガー・ロジックをコピーしてください。

ASN1720E Change Data table information for the source nickname *nicknameowner.nickname* is not found in the *capschema.IBMSNAP_REGISTER* table.

説明: 指定されたソース・ニックネームの `captureschema.IBMSNAP_REGISTER` 表に行が見つかりましたが、そのソースの CCD 表情報がありません。レプリケーション定義をドロップするには、変更データ表の情報が必要です。スクリプトは生成されません。

ユーザーの処置: ソース名が正しいことを確認し、アクションを再度呼び出してください。

ASN1722W The view *view_owner.viewname* will be registered as full refresh, because all the base tables of this view are registered as full refresh.

説明: このビューはフル・リフレッシュとして登録する必要があります。その理由は、このビューの基礎表がフル・リフレッシュのみとして登録されている、または登録されたレプリケーション・ソースではないからです。

ユーザーの処置: アクションは必要ありません。

ASN1723W The view *viewowner.viewname* will be registered for change-capture replication, because one or more base tables from this view are registered for change-capture replication.

説明: このビューは変更キャプチャー・レプリケーションとして登録する必要があります。その理由は、このビューの基礎表が変更キャプチャー・レプリケーションとして登録されているからです。

ユーザーの処置: アクションは必要ありません。

ASN1724E The name of the object that you are creating on the non-DB2 relational server is identical to the *objectowner.objectname* of type *objecttype*.

説明: 指定されたオブジェクトは作成できません。その理由は、DB2 以外のリレーショナル・サーバー上に、同じタイプで同じ名前を持つ既存のオブジェクトがあるためです。

ユーザーの処置: オブジェクトに固有の名前を指定し、レプリケーション・タスクを再実行してください。

ASN1725W The trigger named *triggerowner.trigger_name* already exists on the remote table *remoteowner.remotetablename*. You must not run the generated script until you have determined how to merge the contents of the existing trigger with the generated trigger definition.

説明: この名前を持つトリガーが、DB2 以外のリレーショナル・データベースのリモート表にすでに存在します。生成されたスクリプトの CREATE TRIGGER ステートメントを実行すると、RDBMS はこの重複を指摘せずに、既存のトリガーを上書きする可能性があります。あるいは、RDBMS は、オブジェクトがすでに存在することを示す SQL エラーを戻します。生成されたトリガー名はカスタマイズできませんが、その理由は、カスタマイズされたトリガーは登録のドロップ時にドロップできないからです。

ユーザーの処置: まず、既存のトリガーを生成されたトリガーと組み合わせる方法を決めます。次に、既存のロジックとレプリケーション・ツールが生成したトリガー・ロジックを組み合わせるための、ユーザー独自のスクリプトを作成するか、またはレプリケーション・ツールが生成したスクリプトを更新し、ユーザーの既存のトリガー定義を組み込むようにします。

ASN1726W The trigger named *triggerowner.trigname* does not exist in the remote table *owner.tablename* on the remote server *rmtservername*.

説明: リモート・データベースにトリガーが存在しません。トリガーはドロップされた可能性があります。

ユーザーの処置: アクションは必要ありません。

**ASN1727I The registered source
registered_source is deactivated.**

説明: 指定された登録されたソースは、すでに非活動化されています。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

**ASN1728W The CCSID
Unicode_ASCII_EBCDIC of the
change data (CD) table
cdowner.cdname for the source
table sourceowner.sourcetable does
not match the CCSID
Unicode_ASCII_EBCDIC of the
IBMSNAP_UOW table for the
capture schema capture_schema.**

説明: 与えられたキャプチャー・スキーマについて、アプライ・プログラムは、ASN.IBMNSNAP_MEMBR 表の JOIN_UOW_CD 列が Y であれば、IBMSNAP_UOW 表と指定されたソースの CD 表を結合します。この列に Y が含まれるのは、関連するサブスクリプション・セット・メンバーのターゲット・タイプがユーザー・コピーではないか、または IBMSNAP_UOW 表のいずれかの列をサブスクリプション・セット・メンバーの WHERE 文節で使用している場合です。アプライ・プログラムが異なるコード化スキームを使用して表を結合すると、エラーが起ります。コード化スキームの詳細については、『DB2 Universal Database レプリケーションのガイドおよびリファレンス資料の付録 B』を参照してください。

ユーザーの処置: この登録を使用するサブスクリプション・メンバーについては、ターゲット表はタイプ「ユーザー・コピー」を使用して定義し、WHERE 文節では IBMSNAP_UOW の列を一切、使用しないようにします。

**ASN1729E The registration for the
nickname nicknameowner.nickname
cannot be dropped. The reason
code is reasoncode.**

説明: このニックネームの登録はドロップできません。スクリプトは生成されません。有効な理由コードは次の値です。

0 指定されたニックネームは、前の登録のドロップに含まれていたニックネームと重複します。しかし、その登録ドロップのスクリプトはまだ実行されていません。

ユーザーの処置: 理由コードの説明を検討し、追加情報および制約事項については、「DB2 Universal Database レプリケーションのガイドおよびリファレンス」を参照してください。

**ASN1800E The subscription set set_name
already exists for the Apply
qualifier apply_qual, Who's On
First whos_onfirst, at the Apply
control server server_alias.**

説明: 特定のアプライ修飾子およびアプライ・コントロール・サーバーについて、同じ名前を持つサブスクリプション・セットは 1 つだけです。スクリプトは生成されません。

ユーザーの処置: 新しいセット名を作成するか、または既存のセットに新しいメンバーを追加します。

ASN1801E The statement number *statement_number* is associated with a statement string length *statement_stringlength* that exceeds the maximum statement length for the Apply qualifier *apply_qual*, the set name *set_name*, who's on first value *whos_onfirst*, at the Apply control server *server_alias*.

説明: ステートメントの長さが、許される限界 (V8 では 1024) を超えます。スクリプトは生成されません。

ユーザーの処置: ステートメントの長さが制限値以下になるように、ステートメントのストリングを調整してください。

ASN1802W The Replication subscription source member is defined with **RECAPTURE='N'**. All changes to the replica target will not be propagated to the other replica targets.

説明: 「update-anywhere」シナリオでは、あるターゲット・レプリカでなされた変更は、RECAPTURE='N' の場合、ソース側で再キャプチャーされません。同じソースにサブスクリプションしている複数のターゲット・レプリカがある場合、1 つのターゲット・レプリカでなされた変更は、その他のレプリカ・ターゲットには反映されません。

ユーザーの処置: その他のレプリカ・ターゲットにも変更を伝搬したい場合は、RECAPTURE='Y' をセットしてください。

ASN1803I There exist *orphan_statements* Replication subscription set statements from a previously defined subscription set that was later dropped. These orphan statements are not dropped for the subscription set, for the provided Apply qualifier, at the provided Apply control server.

説明: 前のサブスクリプション・セットが、その該当ステートメントのすべてをドロップせずに、ドロップされました。ドロップされた前のサブスクリプション・セットと同じ名前を持つ、新しいサブスクリプション・セット用のスクリプトが生成されます。前のサブスクリプションの「ステートメント」はドロップされません。

ユーザーの処置: 孤立しているステートメントをドロップするには、「Drop Subscription」ステートメントを実行します。

ASN1804I The Replication subscription set **MAX_SYNCH_MINUTES** *maxsynch_minutes* is not within the allowed range for the provided subscription set and Apply qualifier, at the provided Apply control server. The Replication default value is used instead.

説明: この列に有効な範囲は、0 から 999 です。

ユーザーの処置: デフォルト値の 30 分を受け入れられる場合は、アクションは必要ありません。

ASN1805I The Replication subscription set COMMIT_COUNT

commitcount_value is not within the allowed range for the provided subscription set and Apply qualifier, at the provided Apply control server. The Replication default value is used instead.

説明: この列に有効な範囲は、0 から 999 です。

ユーザーの処置: デフォルト値の 0 分を受け入れられる場合は、アクションは必要ありません。

ASN1806E The replication action ended in error for the Apply qualifier *apply_qualifier*, subscription-set name *set_name*, who's on first value *whos_on_first*, source member *sourceowner.sourcetable*, source view qualifier *source_view_qual*, target member *targetowner.targettable*. The subscription-set member cannot be added to the provided subscription set. Reason code *reason_code*.

説明: メンバーを追加するとサブスクリプション・セットは無効になります。スクリプトは生成されません。有効な理由コードは次の値です。

- 0 そのサブスクリプション・セットは、メンバーの最大数に達しました。
- 1 キャプチャー・スキーマのソース・メンバーは、キャプチャー・スキーマのサブスクリプション・セットと同じではありません。
- 2 iSeries のソース・メンバーは、サブスクリプション・セット・ジャーナルと同じではありません。

- 3 コンデンスされた表のメンバー構造が、他のメンバー構造と互換性がありません。
- 4 ソース・メンバーは変更キャプチャー・レプリケーションをサポートしませんが、ターゲット・メンバーは変更キャプチャーに依存しています。ターゲット構造は CCD またはレプリカ表のどちらかですが、ソースには CD 表がありません。
- 5 ソース・メンバーは完全な表ではありません。
- 6 ターゲット・メンバーの定義は、ターゲット表が存在すると想定していますが、ターゲット表が存在しません。
- 7 ターゲット・メンバーの定義は、ターゲット表の作成を求めています、ターゲット表がすでに存在します。
- 8 そのセットには、フル・リフレッシュをサポートされるターゲット表だけが含まれていますが、新しいメンバーは変更キャプチャー・レプリケーションをサポートします。
- 10 そのセットには、変更キャプチャー・レプリケーションによりサポートされるターゲット表だけが含まれますが、新しいメンバーはフル・リフレッシュだけをサポートします。
- 11 レプリカ規則: ターゲット・メンバーがレプリカの場合、ソース・メンバーはレプリカまたはユーザー表のどちらかにできます。
- 12 ターゲット構造は、このオペレーティング・システムではサポートされません。
- 13 ターゲット構造は CCD であり、登録ソースとしてセットされていますが (自動登録)、構造が完全ではありません。
- 14 ソース・メンバーが登録されていません。

- 15 ソース・メンバー列は列定義を持っていますが、ターゲット・タイプが集合ではありません。
- 16 サブスクリプション・セットから除外されたターゲット列の、少なくとも 1 つが、NULL 可能でも、デフォルト使用の NOT NULL でもありません。
- 17 ターゲット・メンバーは、更新できないビューです。
- 18 サブスクリプション・セット・メンバーがすでに存在します。
- 19 登録されたソースに有効なマッピングを持つ、ターゲット列または式が見つかりません。
- 20 複数の有効なソースが見つかりましたが、整合性のある定義がされていません。
- 21 外部 CCD 表はコンデンスされておらず、DataLink または LOB 列を含んでいます。
- 22 ソース・メンバーのジャーナル・ライブラリーまたはジャーナル名が一致しません。
- 23 リモート・ジャーナル名が無効です。
- 24 ジャーナル名またはライブラリーが無効です。
- 25 リモート・ジャーナリングを持つレプリカ表は許されません。
- 26 指定された登録されたソース表の内部 CCD 表がすでに存在します。
- 27 内部 CCD 表のソースとターゲットのサーバーは同じにする必要があります。
- 28 内部 CCD 表は不完全である必要があります。
- 29 ソース表はリモート側でジャーナルされ、LOB または DATALINK 列を含みます。

- 30 IBMSNAP_PRUNCNTL 表に関連情報がありません。
- 31 IBMSNAP_PRUNE_SET 表に関連情報がありません。
- 32 ビューをソースとして持つ内部 CCD 表は許されません。

ユーザーの処置: 理由コードに基づき、別のサブスクリプション・セットに新しいメンバーを作成するか、または新しいメンバー用に新しいサブスクリプション・セットを作成します。

ASN1807I The replication subscription member is added to the provided subscription set and Apply qualifier, at the provided Apply control server with an informational clause. Reason code *reason_code*.

説明: このメッセージは通知のみであり、アクションは必要ありません。スクリプトは生成されません。有効な理由コードは次の値です。

- 0 新しいセットは、レプリカと読み取り専用ターゲット・メンバーが混在することになります。
- 1 サブスクリプション・セットはトランザクション・コミット・カウントをサポートしますが、ターゲット・メンバーはトランザクション処理に適格ではありません。
- 2 少なくとも 1 つのメンバーが CCD 表のターゲット・メンバーを持っていますが、すべてのメンバーがターゲット・メンバー CCD 表を持っているわけではありません。同じセットに、表の異なる生成が含まれます。
- 3 ターゲット・メンバーは、コンデンスされていない、完全でない CCD 表であ

り、IBMSNAP からの追加の列がありません。このターゲット表は実質的には CD 表と同じです。

ユーザーの処置: 理由コードの説明を検討し、次のオプションを使用して対応してください。

- 0 すべてのレプリカ表を 1 つのセットに保持し、読み取り専用表は別のセットに保持することを考慮します。
- 1 トランザクション処理をサポートするターゲット・タイプは、すべて同じセットに保持し、他の表は別のセットに保持することを考慮します。
- 2 複数のセットにまたがって、整合性のある経過日数を維持するため、同じ生成の一部である CCD ターゲット表をすべて同じセット内に保持することを考慮します。また、CCD 以外のターゲット表はすべて、別のセットに保持することを考慮します。2 番目のセットは、データが最初のセットにあることを前提とし、これに依存しています (たとえば、middle-tier ステージのように)。
- 3 CCD ターゲット表が必要かどうかを考慮してください。

ASN1808E The Replication action ended in error for Apply qualifier
apply_qualifier, **set name** *set_name*,
Who's On First *whos_on_first*,
source member
sourceowner.sourcetable, **source**
view qualifier *source_view_qual*,
target member
targetowner.targettable. **The subscription target member expects its index key columns to be updated but at least one index key does not have its before-image column registered in the subscription source member.**

説明: ターゲット表は、その索引キーの更新を許しています (PRIMARY_KEY_CHG = 'Y')。この要求をサポートするため、アプライ・プログラムは索引キーの変更前イメージ列をアクセスする必要があります。したがって、これらの変更前イメージ列がソース・メンバーの変更データ表に存在する必要があります。これがない場合、アプライ・プログラムは失敗します。スクリプトは生成されません。

ユーザーの処置: サブスクリプション・ターゲット索引のそれぞれの列について、

- その列の変更前イメージ列が、キャプチャー・サーバー側で、ソース・メンバーにすでに登録されているかどうかをチェックします。
- 登録されていない場合は、該当の変更前イメージ列を登録します。

ASN1809W The Replication action expects a subscription index key columns to be updated but the subscription member will be added to the subscription set without allowing updates to the target index key columns.
Reason code *reason_code*.

説明: 上のメッセージにリストされているケースでは、PRIMARY_KEY_CHG の設定は意味がありません。ユーザー提供の定義をオーバーライドするレプリケーション定義を使用して更新されたスクリプトが生成されます。有効な理由コードは次の値です。

- 0 ターゲット表タイプ CCD:
PRIMARY_KEY_CHG は無効です。
- 1 IBMSNAP_REGISTER 表の
CHG_UPD_TO_DEL_INS の値が 'Y' に
セットされています。
- 2 ターゲット表がコンデンスされていま
せん。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1810W The Replication subscription source member is defined on the source server with DB2 Referential Integrity constraints, but the subscription target member is a Replica that does not preserve these constraints.

説明: ターゲット表側の参照保全制約が、レプリカ・サイト側で DB2 により強制されません。レプリカ・サイト側ではこのことは意図していない可能性があります。スクリプトは生成されますが、スクリプトを実行する準備ができていない可能性があります。

ユーザーの処置: 必要ならば、生成されたスクリ

プトを更新し、ターゲット側の該当の参照保全制約を組み込みます。

ASN1811W The index definition for the target subscription member cannot guarantee proper uniqueness at the target.
Reason code *reason_code*.

説明: ある種のターゲット・タイプの場合、アプライ・プログラムは、ターゲット表内の行を正しく更新または削除するために、固有索引の定義に依存しています。提供された索引が固有性を保証しない場合、アプライ・プログラムは何らかの再作業をすることになります。使用するアプリケーションがこのようなケースに該当しないことを確認してください。有効な理由コードは次の値です。

- 0 SQL 関数により少なくとも 1 つの列が生成されましたが、これは索引の固有性を保証しません。
- 1 固有索引には、NULL 可能列は SQL 関数により生成されません。

ユーザーの処置: サブスクリプション・ターゲット索引のそれぞれの列について、

- アプライ・コントロール・サーバーの列タイプ (ASN.IBMSNAP_SUBS_COLS の COL_TYPE) が 'F' かどうかをチェックします。
- 'F' ならば、SQL 式を含めないように索引列の式を再定義するか、または、索引キーからその列を除去します (ASN.IBMSNAP_SUBS_COLS、IS_KEY 列を 'N' にする)。

ASN1812E The Replication action ended in error for Apply qualifier
apply_qualifier, **set name** *set_name*,
who's on first value
whos_on_first, **source member**
sourceowner.sourcetable, **source**
view qualifier *source_view_qual*,
target member
targetowner.targettable. **The**
subscription target member
cannot be added because the
required target key is not valid.
Reason code *reason_code*.

説明: 固有索引を必要とするターゲット・メンバーは、ターゲット・タイプがポイント・イン・タイム、ユーザー・コピー、およびレプリカの表、およびコンデンス CCD です。これらのターゲットが固有索引を持たない場合、アプライ・プログラムは失敗します。スクリプトは生成されません。有効な理由コードは次の値です。

- 0 ターゲット表はまだ存在しませんが、ターゲットのキー情報をソース表から導き出すことができません。
- 1 ターゲットのキー情報が見つからず、導き出すこともできず、CD 表に RRN が定義されていません (iSeries のみ)。
- 2 ターゲット表またはビューがすでに存在し、必要なターゲット・キー情報がありません。
- 3 ターゲット表またはビューがすでに存在しますが、ターゲット・キー情報が既存のパーティション化キー情報と互換性がありません。

ユーザーの処置: 有効なターゲット・キーを定義します。以下の理由コード別に、それぞれのアクションをします。

- 0 ソース表に該当の固有索引を作成し、レプリケーションで提案された索引を導き出すために使用できるようにします。

- 2 必要なターゲット・キー情報を提供しません。
- 3 パーティション化索引の DB2 規則については、「SQL リファレンス」を参照してください。たとえば、提供したキーに、必要なパーティション化キーが含まれていない可能性があります。

ASN1813I The Replication subscription source member is defined on the source server with some DB2 constraints, but the subscription target member does not preserve these constraints. Reason code
reason_code.

説明: ソース表の制約は、ターゲット・メンバーの定義中に指定されなければ、DB2 により強制されません。レプリカ表のサーバー側では、このことは意図していない可能性があります。スクリプトは生成されますが、実行する準備ができていない可能性があります。制約については、以下の有効な理由コードに記述されています。

- 0 ターゲット・メンバー内に、少なくとも 1 つの NOT NULL WITH DEFAULT 文節。
- 1 パーティション表スペース。

ユーザーの処置: 必要ならば、生成されたスクリプトを更新し、ターゲット側の該当の DB2 制約を組み込みます。

ASN1814E The target column *column_name* of data type *datatype* cannot be added to subscription target member *tableowner.tablename*.
Reason Code *reason_code*.

説明: サブスクリプション・メンバーは、サブスクリプション列のチェックに失敗します。スクリプトは生成されません。有効な理由コードは次の値です。

- 0 列のデータ・タイプはレプリケーションによりサポートされません。DB2 によりサポートされないデータ・タイプの場合、メッセージ ASN1648E が出されます。
- 1 ターゲットのデータ・タイプは、対応するソースのデータ・タイプと互換性がありません。
- 2 ソース表の登録に列が見つかりません。
- 3 連合ターゲットにはサポートされない列のタイプです。
- 4 ターゲット列は LOB です。ターゲット・メンバーの LOB 列の最大数を超過しています。
- 5 ソース列は SQL 列関数を含んでいますが、ターゲット・メンバーの構造は基本集合でも、変更集合でもありません。
- 6 ターゲット表のタイプはレプリカであり、ソース列は LOB 列です。
- 7 ターゲット表のタイプはレプリカであり、ソース列は DATALINK 値です。しかし、CONFLICT_LEVEL > 0 です。
- 8 LOB 列を持つ、コンデンスでない CCD ターゲット表はサポートされません。
- 9 列が既存のターゲット表にありません。

ユーザーの処置: 理由コードの説明を検討し、次のように対処します。

- 0 サポートされているデータ・タイプに変更します。
- 1 ターゲットのデータ・タイプがソースのデータ・タイプと一致していることを確認します。
- 2 ソース表の列を登録します。
- 3 連合ターゲットにサポートされている有効なデータ・タイプを選択します。

- 4 ターゲット・メンバー側の LOB 列の数が、許される制限を超えないことを確認します。
- 5 ソース列の式またはターゲット表構造のどちらかを変更します。
- 6 サブスクリプション・メンバーから、レプリカ・ターゲットの LOB 列を除去します。
- 7 レプリカが 0 より大きい競合レベルを必要とする場合は、サブスクリプション・メンバーから DATALINK 列を除去します。そうでない場合は、レプリカの競合レベルを変更します。
- 8 LOB 列を除去します。
- 9 列名を検査します。

ASN1815E The Replication action ended in error. The subscription set *set_name* for Apply qualifier *apply_qual*, Who's On First *whos_on_first* is to be dropped if empty, but at least one member exists for this set. The subscription set cannot be dropped.

説明: 特定のサブスクリプション・セットに指定されたアプライ修飾子について、指定されたアプライ・コントロール・サーバーの ASN.IBMSNAP_SUBS_MEMBR に少なくとも 1 つのメンバーが存在するため、サブスクリプション・セットはドロップされません。スクリプトは生成されません。

ユーザーの処置: まだ存在するサブスクリプション・メンバーをドロップしてから、サブスクリプション・セットをドロップします。あるいは、サブスクリプション・セットが空であるという要件を指定せずに、*Drop Subscription Set* タスクを実行します。

ASN1816W The Replication subscription set contains at least one member that will be dropped once the subscription set is dropped.

説明: サブスクリプション・セットが正常にドロップされると、すべてのセット・メンバーも自動的にドロップされます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1818W The Replication subscription member is updated with new subscription Where Clause predicates. Previous predicates already exist for the subscription member. They will be overwritten by the new predicate information.

説明: 指定されたメンバーは、すでに述部を含んでいます。新しい述部により、古い述部は上書きされます。スクリプトは生成されます。

ユーザーの処置: レプリケーション・タスク用の完全な述部の文節を指定します。述部文節が既存の述部文節のすべてを含んでいない場合は、これを更新する必要がある場合があります。

ASN1819W The Replication subscription set is disabled successfully. Note however that disabling a subscription set has a direct impact to the Capture pruning logic of all source members for that subscription set.

説明: キャプチャーのプルーニングのロジックは、アプライ・プログラムが従属サブスクリプション・メンバーを入れるまで、CD 表のプルーニングはしません。スクリプトは生成されます。サブスクリプション・セットを使用不可にするよりはむしろ、サブスクリプション・セットをドロ

ップした方がよい場合は、スクリプトを更新します。以下を参照してください。

ユーザーの処置: サブスクリプション・セットをかなり長期間、使用不可にし、そのために CD 表のプルーニング処理が影響を受ける場合、または従属する登録されたソース用の CD 表への影響が、キャプチャー・プログラムおよびキャプチャー・サーバーの CD 表に劇的に影響を与える場合は、単に使用不可にするよりも、サブスクリプション・セットをドロップして、後でこれを再作成した方がよいでしょう。あるいは、該当の登録を非活動化します。

ASN1820E The Replication string for subscription set *set_name*, apply qualifier *apply_qual*, Who's On First *whos_on_first* contains DB2 syntax that is not valid. The string type is *string_type*, string text is *string_text* and SQL Message is *sql_message*.

説明: 指定されたストリングが無効です。スクリプトは生成されません。

ユーザーの処置: 該当のオブジェクト構文を訂正し、再度、レプリケーション・タスクを実行します。

ASN1821W Dependent subscription sets will no longer be valid after the existing subscription set is dropped, if this subscription set contains target members that are registered sources at its target server.

説明: 従属サブスクリプションは、ソース・メンバー表が存在することを前提とし、これに依存しています。これらのソース・メンバーがレプリケーション・ターゲットとして保守されており、これらのターゲットがドロップされると、アプライ・プログラムは従属サブスクリプション・セッ

トの処理時に失敗します。従属サブスクリプション・セットは、ターゲット・サーバーの `captureserver.IBMSNAP_PRUNCNTL` 表に `SOURCE_OWNER` または `SOURCE_TABLE` 行が含まれており、その値がドロップされるターゲット表のものである場合は、影響を受けない可能性があります。スクリプトは生成されます。

ユーザーの処置: 必要な場合、スクリプトを実行する前に、従属サブスクリプション・セットを非活動化またはドロップします。

ASN1822E The Replication action ended in error for Apply qualifier
apply_qual, set name *set_name*,
source member
sourceowner.sourcename, **target member** *targetowner.targetname*.
The provided subscription member does not exist for the provided subscription set.

説明: 指定されたアプライ・コントロール・サーバーの、指定されたアプライ修飾子の `ASN.IBMSNAP_SUBS_MEMBR` に、指定されたメンバーが見つかりません。

ユーザーの処置: 指定したアプライ修飾子、セット名、メンバー名、およびコントロール・サーバーが正しいことを確認してください。

ASN1823E The subscription set *set_name* does not exist for the Apply qualifier *apply_qual*, Who's On First *whos_onfirst*, at the Apply control server *server_alias*.

説明: 指定されたアプライ・コントロール・サーバーの、指定されたアプライ修飾子の `ASN.IBMSNAP_SUBS_SET` に、指定されたサブスクリプション・セットが見つかりません。

ユーザーの処置: 指定したアプライ修飾子、セット名、メンバー名、およびコントロール・サーバーが正しいことを確認してください。

ASN1824W The Replication subscription set was updated with a COMMIT_COUNT of 0.

説明: ソースは複数の表のビューであり、セットのコミット・カウントが `NULL` です。そのセットにはコミット・カウント `0` が強制されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1825W The Replication action to drop a member did not drop the view.

説明: ビューをドロップする要求が出されましたが、アクションは要求されたようには完了しませんでした。

ユーザーの処置: 手作業でビューをドロップする必要があります。

ASN1826W The Capture schema *capture_schema* does not exist at the provided source server.

説明: サブスクリプション・セットにメンバーを追加する前に、キャプチャー・スキーマが存在することを確認してください。

ユーザーの処置: 指定されたキャプチャー・スキーマを持つソース・サーバーに、キャプチャー・サーバー・コントロール表を作成します。

ASN1827W The column *target_columnname* of the target member *target_member* does not preserve a DB2 column attribute of the corresponding column *source_columnname* of the source member *source_member*. Reason Code *reason_code*.

説明: ソース列の DB2 列属性が、対応するターゲット列のものと異なります。有効な理由コードは次の値です。

1

ソース列は NULL 可能であり、ターゲット列は NULL 不可です。

2

ソース列は NULL 不可であり、ターゲット列は NULL 可能です。

3

ソース列はデフォルト値を持ち、ターゲット列は持ちません。

4

ターゲット列はデフォルト値を持ち、ソース列は持ちません。

ユーザーの処置: 理由コードが 1 の場合は、ターゲット列にアプライされる NULL 値が、ソース列にあるかどうかをチェックします。必要ならば、ターゲット列を NULLABLE に変更します。あるいは、必要ならば、生成されたスクリプトを変更し、ターゲット側の該当の DB2 属性を含めます。

ASN1828E The Replication action ended in error for Apply qualifier *apply_qual*, set name *set_name*, source member *sourceowner.sourcename*, target member *targetowner.targetname*. When the subscription target server is a non-IBM target server, either the action is not supported or is supported with restrictions. Reason Code is *reason_code*.

説明: これらは、現在の制約事項です。スクリプトは生成されません。有効な理由コードは次の値です。

0 サポートされていません。

1 サポートされるターゲット表構造：ポイント・イン・タイム、CCD、ユーザー・コピー。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1829I A valid nickname *nicknameowner.nickname* is found for the subscription target table. Column data type mapping rules are enforced. The nickname is used as is.

説明: このサブスクリプションの有効な連合データベース (列のデータ・タイプ・マッピング・チェックが有効) に、既存のターゲット・ニックネームが見つかりました。しかし、DB2 以外のリレーショナル・データベース・サーバー上にターゲット表が存在するかどうかのチェックはありません。

ユーザーの処置: 指定されたニックネームにリモート表が存在するかを確認してください。存在しない場合、アプライ・プログラムは失敗します。

ASN1830E The Replication action ended in error for the Apply qualifier *apply_qualifier*, subscription set *set_name*, whos on first value of *whos_on_first*, source member *sourceowner.sourcetable*, source view qualifier *source_view_qual*, target member *targetowner.targettable*, and predicate of *predicate*. The Subscription-set member cannot be added to this subscription set. The reason code is *reason_code*.

説明: サブスクリプション・セット・メンバーは無効であり、スクリプトは作成されません。可能な理由コードは以下のとおりです。

0 述部が、存在しない CD または UOW 表からの列を参照している。

ユーザーの処置: 指定された述部が正確なものであることを検査し、高度な変更述部機能については資料を参照してください。

ASN1831E The Replication action ended in error. No subscription statements exist for the subscription set *set_name* for the Apply qualifier *apply_qual*, with a whos on first value of *whos_onfirst*, at the Apply control server *control_server*.

説明: このアプライ修飾子を持つ、指定されたサブスクリプション・セット名に、サブスクリプション・ステートメントがありません。

ユーザーの処置: このアプライ修飾子の下の指定されたサブスクリプション・セット名に、サブスクリプション・セット・ステートメントが含まれていることを検査してください。

ASN1832W A column named *column_name* already exists in the ASN.IBMSNAP_SUBS_COLS control table.

説明: 指定された列が

ASN.IBMSNAP_SUBS_COLS 表にすでに存在しません。

ユーザーの処置: アクションは必要ありません。

ASN1833E The CCSID *Unicode_ASCII_EBCDIC* of the change data (CD) table *cdowner.cdname* for the source table *sourceowner.sourcetable* does not match the CCSID *Unicode_ASCII_EBCDIC* of the IBMSNAP_UOW table for the capture schema *capture_schema*. The provided subscription member definition would require a join of these two tables.

説明: 与えられたキャプチャー・スキーマについて、アプライ・プログラムは、関連するサブスクリプション・セット・メンバーのターゲット・タイプがユーザー・コピーでないか、またはサブスクリプション・セット・メンバーの WHERE 文節で IBMSNAP_UOW 表の列を 1 つも使用していない場合、IBMSNAP_UOW 表と与えられたソースの CD 表を結合します。サブスクリプション・セット・メンバーが、与えられたソース表とキャプチャー・スキーマを使用し、ソース表の CD 表を IBMSNAP_UOW 表と結合することにより定義されている場合、アプライ・プログラムがこのメンバーを処理すると、表のコード化スキーマが異なるためにエラーが起きます。コード化スキーマの詳細については、『DB2 Universal Database レプリケーションのガイドおよびリファレンス資料の付録 B』を参照してください。

ユーザーの処置: 以下のいずれかを行ないます。

- ユーザー・コピーのターゲット・タイプを選択し、サブスクリプション・メンバーの WHERE 文節で IBMSNAP_UOW 表の列を使用しないようにします。または、
- 別のキャプチャー・スキーマを使用してソースを登録し、新しいキャプチャー・スキーマの IBMSNAP_UOW 表と同じコード化スキームを持つ表スペースに、CD 表を作成します。

ASN1834W The default target capture schema of 'ASN' will be used for the subscription set.

説明: サブスクリプション・セットはターゲット・キャプチャー・スキーマを必要とし、デフォルト値 'ASN' が使用されます。

ユーザーの処置: デフォルトが、このサブスクリプション・セットのターゲット・キャプチャー・スキーマ列にとって適切な場合は、アクションは必要ありません。

ASN1835W The target column *column_name* of data type *datatype* has been added to the subscription-set member target *tableowner.tablename*, but the corresponding source column *column_name* of data type *datatype* can contain data that is not applicable to the target column. Reason code *reason_code*.

説明: ソース列の定義が、ターゲット列の定義と正確に一致しません。したがって、アプライ・プログラムがソースから選択したデータがターゲット列にとって適切でない場合、アプライ・プログラムは失敗するか、またはソース・データを変更する (切り捨てる) 場合があります。注: アプリケーションが、アプライ・プログラムを失敗させるようなデータを生成しない場合は、定義が不一致であっても問題はありません。

有効な理由コードは次の値です。

1
ターゲット列の長さが、ソース列の式を解決した結果より短い。

2
ターゲット列のスケールが、ソース列の式を解決した結果より短い。

3
ターゲット列の精度が、ソース列の式を解決した結果より短い。

4
ターゲット列とソース列のデータ・タイプは、ある種のソース値についてのみ互換性がある。

ユーザーの処置: 可能ならば、ターゲット・サイトの定義を変更して、ソース定義と互換性があるようにします。(これは通常、ターゲット・サイトで実行されるアプリケーションにより導かれます。)

定義の不一致をそのままにしておく必要がある場合は (その理由は、ターゲットの定義がソース定義と異なるべき、特別の理由があるため)、アプリケーションを見直し、定義の不一致により実行時に問題が起こらないことを確認してください。

ASN1836W The target table *owner.name* will not be dropped because it is registered as a source under the Capture schema *schemaname*.

説明: ターゲット表は指定されたキャプチャー・スキーマの下にソースとして登録されています。表をドロップすると、登録は無効になります。

ユーザーの処置: 表の登録をドロップし、次に表をドロップします。

**ASN1837W The DB2 target
tableowner.tablename is not
dropped.**

説明: ターゲット表はレプリカまたは外部整合変更データ表 (CCD) であり、ターゲット・サーバーでも登録されているので、従属ターゲットのソースである可能性があります。表を自動的にドロップすることはできません。

ユーザーの処置: レプリカまたは外部 CCD 表に登録されたソースをドロップします。次に、手作業で DB2 ターゲット表をドロップするかまたは、レプリカまたは外部 CCD サブスクリプション・メンバーをドロップします。メンバーをドロップすると DB2 ターゲット表がドロップされます。

**ASN1900E The table or view
objectowner.objectname cannot be
promoted to the new server.
Reason code *reason_code*.**

説明: 有効な理由コードは次の値です。

- 0 このオペレーティング・システムの表タイプには、プロモート要求はサポートされません。
- 1 プロモートに必要なソース・サーバーのオペレーティング・システムは、ターゲット・サーバーのオペレーティング・システムと一致する必要があります。
- 2 表またはビューがありません。

ユーザーの処置: 理由コードの説明を検討し、次のように対処します。

- 0 このメッセージは通知のみであり、アクションは必要ありません。
- 1 現在の制約です。
- 2 表またはビューがソース・サーバーのオペレーティング・システムに存在することを検査します。

**ASN1901E The registered source
sourceowner.sourcename cannot be
promoted for Capture schema
captureschema at Capture server
capture_server. Reason code
reason_code.**

説明: 有効な理由コードは次の値です。

- 0 表またはビューは登録されたソースではありません。
- 1 登録されたソースはレプリカ表です。
- 2 登録されたソースは DB2 (iSeries 版) にありますが、リモート・ジャーナルを持っています。
- 3 表またはビューはすでにプロモートされています。
- 4 ビューのビューは、レプリケーション・プロモート登録関数ではサポートされません。

ユーザーの処置: 理由コードの説明を検討し、次のように対処します。

- 0 IBMSNAP_REGISTER 表に指定した表またはビュー名には、指定されたキャプチャー・スキーマのエントリが含まれていません。表またはビュー登録を、この特定のキャプチャー・スキーマにプロモートすることはできません。
- 1 指定された表は、`captureschema.IBMSNAP_REGISTER` 表でタイプがレプリカ (`SOURCE_STRUCTURE` 列の値が 7) になっています。表は登録されたソースとしてプロモートすることはできません。レプリカをプロモートできるのは、ソース・ユーザー表とレプリカ・ターゲット間で適切な定義を維持できるサブスクリプション・セットのコンテキスト内のみです。
- 2 登録されたソースはリモート・ジャーナ

ルを持つ DB2 (iSeries 版) で保守されており、これは SQL スクリプトでのみプロモートすることができます。

ASN1902W Make sure that the schemas exist on the promoted Capture server before running the script. The Replication definitions will be incomplete if the object does not exist at the promoted Capture server.

説明: プロモート・タスクでは、新しいキャプチャー・サーバーおよび新しいキャプチャー・サーバー・スキーマを提供できるようにしています。ただし、プロモート・タスクは、キャプチャー・サーバーとスキーマの名前および存在するかどうかを確認するために、キャプチャー・サーバーに接続することはしません。スクリプトを正常に実行するには、スクリプトを実行する前にこの情報を確認する必要があります。

ユーザーの処置: スクリプトを実行する前に、該当の SQL を実行してから、必要なオブジェクトを生成します。

ASN1903W The object *objectowner.objectname* does not exist on the promoted Apply control server, yet some promoted objects depend on its existence. Failure to create this object will result in incomplete Replication definitions at promoted Apply control server. Reason code *reason_code*.

説明: プロモート・タスクでは、新しいアプライ・コントロール・サーバー名を指定できるので、生成されたスクリプトが必ず適切に実行されるように、このタスクはいくつかの必須オブジェクトが存在するかどうかを検出します。スクリプトは生成されますが、実行する準備はできていません。有効な理由コードは次の値です。

- 0 アプライ・コントロール・サーバーのコントロール表がありません。
- 1 プロモートされたセットのすべてのソース・メンバーの登録情報。

ユーザーの処置: スクリプトを実行する前に、該当の SQL を実行してから、必要なオブジェクトを生成します。

ASN1904I The Replication subscription member is promoted successfully for the provided Apply qualifier, at the provided Apply control server. Reason code *reason_code*.

説明: このメッセージは通知のみであり、アクションは必要ありません。スクリプトは生成されますが、実行する前に一部を更新する必要がある場合があります。有効な理由コードは次の値です。

- 0 ソース・メンバーの構造が、ターゲット・メンバーの構造と互換性がありません。
- 1 ターゲット・メンバーはレプリカ (replica1) であり、これは別のレプリカ (replica2) のソース・メンバーでもありません。 replica2 の登録行の RECAPTURE 値は、ユーザー表からの更新を replica2 で複製することを許しません。
- 2 ソース・メンバーはユーザー表であり、この表は複数のレプリカ (replica1 および replica2) のソース・メンバーでもあります。ユーザー表の登録行の RECAPTURE 値は、 replica1 での更新を replica2 で複製することも、その逆も許しません。

ユーザーの処置: 理由コードの説明を検討し、次のように対処します。

- 0 ASN.IBMSNAP_SUBS_MEMBR 表の TARGET_STRUCTURE 列をチェックします。この列の値は、対応するソース・

メンバーの
captureschema.IBMSNAP_REGISTER 表、
SOURCE_STRUCTURE 列と互換性がある
必要があります。

1, 2 必要な場合、値を更新します。

ASN1905W The Capture server alias and the Capture schema name on both the host system and new system are the same. The generated replication definitions cannot work if run on the host system.

説明: プロモート・タスクは、キャプチャー・サーバーの別名とキャプチャー・スキーマ名が、ホストおよび新しいシステムの両方で同じであること検出しました。生成された SQL スクリプトを変更しないと、実行時に失敗します。

ユーザーの処置: 次のアクションの 1 つを取ります。1) ホストまたは新しいシステムに、別のキャプチャー・サーバー別名およびキャプチャー・スキーマ名の値を使用して、同じタスクを実行します。2) または、生成されたスクリプト内のキャプチャー・サーバー別名またはキャプチャー・スキーマ名を変更します。

ASN1950E ASNCLP : An unexpected token *token_name* was found. Valid tokens include *list_of_tokens*.

説明: 誤りの構文を使用してコマンドが入力されました。

ユーザーの処置: 資料でコマンド構文を確認してください。

ASN1951E ASNCLP : The command was entered with profile *profile_name* that is not valid.

説明: コマンドでプロファイルを使用するには、プロファイルが存在する必要があります。

ユーザーの処置: 該当する SET PROFILE コマンドを実行してから、失敗したコマンドを再入力してください。 that failed.

ASN1952E ASNCLP : The program encountered an internal error.

説明: レプリケーション・コマンド行プロセッサは、リカバリー不能エラー条件を検出しました。

ユーザーの処置: エラーに関するログ・ファイルを入手し、IBM ソフトウェア・サポートに連絡してください。

ASN1953I ASNCLP : Command completed.

説明: この ASNCLP セッションのすべてのコマンドが正常に終了しました。ただし、このセッション内のいくつかの個々のコマンドは、エラー、警告、または通知メッセージを出している可能性があることに注意してください。

ユーザーの処置: このセッション内のコマンドにより、何らかのエラー、警告、または通知メッセージが出されているかどうかを、ASNCLP ログ・ファイルを見てチェックしてください。

ASN1954E ASNCLP : Command failed.

説明: ASNCLP セッション内の少なくとも 1 つのコマンドが失敗し、処理が停止しました。

ユーザーの処置: ASNCLP ログ・ファイルを見てエラーを診断します。それから、エラーを修正し、コマンドを再度試行します。

ASN1955I **ASNCLP : The program will use the following files:**
capture_script_file_name for the **Capture SQL script,**
control_script_file_name for the **control SQL script,**
target_script_file_name for the **target SQL script, and**
log_file_name for the **log file.**

説明: ASNCLP セッションは、指定されたファイルに情報を生成しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1956I **ASNCLP : The program now generates the script for action:**
action_name.

説明: このコマンドのすべての入力は正常に解析され、スクリプトを生成するコマンドを呼び出し中です。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1957E **ASNCLP : The value *value* for the input parameter *input_parameter* is incorrect. The reason code is *reason_code*.**

説明: 入力パラメーターの値が誤りです。有効な理由コードは次の値です。

- 1 入力パラメーターは文字値ですが、数値にする必要があります。
- 2 入力パラメーターは数値ですが、文字値にする必要があります。
- 3 コマンド行プロセッサは指定されたファイルをアクセスできません。

ユーザーの処置: 理由コードをチェックし、有効な入力パラメーター値を指定します。

ASN1976E *pgmname* : *program_qualifier*. **The specified database alias *db_alias_name* already exists in the password file *password_file_name*.**

説明: 指定されたキーは、パスワード・ファイルにすでに存在します。

ユーザーの処置: ADD パラメーターでなく MODIFY パラメーターを使用して、このコマンドを再度入力してください。

ASN1977E *pgmname* : *program_qualifier*. **The value of the input parameter *parameter_name* is missing.**

説明: 上のメッセージに示されている入力パラメーターを指定する必要があります。

ユーザーの処置: 有効な入力パラメーター値を使用して、ユーティリティを再度呼び出します。

ASN1978E *pgmname* : *program_qualifier*. **The value of the input parameter *parameter_name* is too long.**

説明: asnpwd コマンドは、データベース別名の長さは最大 8 文字、ユーザー ID とパスワードの長さは両方とも最大 128 文字をサポートします。ユーザー ID とパスワードの特定の長さは、使用するオペレーティング・システムにより異なります。

ユーザーの処置: 有効な長さを持つ入力パラメーターを使用して API を呼び出します。

ASN1979E *pgmname* : *program_qualifier*. **The program encountered an unexpected token *token_name*. Expected tokens include *list_of_tokens*.**

説明: 誤りの構文を使用してコマンドが入力されました。

ユーザーの処置: 資料でコマンド構文を確認してください。

ASN1980E *pgmname* : *program_qualifier*. **The program did not complete successfully because** *reason*.

説明: asnpwd コマンドで、メッセージに示されているシステム問題が起きました。

ユーザーの処置: メッセージの情報に基づいてアクションをします。エラーを修正した後、コマンドを再入力します。

ASN1981I *pgmname* : *program_qualifier*. **The program completed successfully using password file** *password_file_name*.

説明: asnpwd コマンドが正常に終了しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN1982E *pgmname* : *program_qualifier*. **The specified database alias** *db_alias_name* **does not exist in the password file** *password_file_name*.

説明: MODIFY または DELETE パラメーターで指定されたキーは、パスワード・ファイルにありません。

ユーザーの処置: ADD パラメーターを使用して、コマンドを再入力します。

ASN1983E *pgmname* : *program_qualifier*. **The program cannot find the password file** *password_file_name*.

説明: パスワード・ファイルがありません。

ユーザーの処置: 指定されたパスにパスワード・ファイルがあることを検査してください。

Password Management ユーティリティーを初めて

使用する場合は、INIT パラメーターを使用します。

ASN1984E *pgmname* : *program_qualifier*. **The program cannot be initialized because the password file** *password_file_name* **already exists.**

説明: 指定されたパスにパスワード・ファイルがすでに存在します。

ユーザーの処置: パスワード・ファイルが削除されたことを検査してください。その後、コマンドを再試行します。

ASN1985E *pgmname* : *program_qualifier*. **The program encountered an internal error when using the password file** *password_file_name*.

説明: パスワード・ファイルのアクセスを試みている時に、オペレーティング・システムで予期しないエラーが起きました。このエラーについて使用できる情報はありません。ただし、このエラーは、パスワード・ファイルを手作業で編集した結果、ファイルのフォーマットが変わったために起こった可能性があります。

ユーザーの処置: コマンドを再試行します。問題が持続する場合は、INIT パラメーターを使用して新しいパスワード・ファイルを作成します。

ASN5101I **MONITOR** *monitor_qualifier*. **The Replication Alert Monitor program started successfully.**

説明: このメッセージは、レプリケーション・アラート・モニター・プログラムが正常に開始された後で表示されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5102I **MONITOR** *monitor_qualifier*. The Replication Alert Monitor program initialized successfully and is monitoring *number-of-alert-conditions* alert conditions.

説明: レプリケーション・アラート・モニター・プログラムが正常に開始されました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5103I **MONITOR** *monitor_qualifier*. The Replication Alert Monitor program re-initialized successfully and is monitoring *number-of-alert-conditions* alert conditions.

説明: レプリケーション・アラート・モニター・プログラムが正常に再初期化されました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5104W **MONITOR** *monitor_qualifier*. *number-of-alert-conditions* alert conditions were ignored.

説明: レプリケーション・アラート・モニター・プログラムが初期化、または再初期化されました。前に出されたメッセージで示されている、無効ないくつかのアラート条件が除外されている可能性があります。

ユーザーの処置: IBMSNAP_CAPTRACE 表を見て、除外されたアラート条件についてのメッセージをチェックしてください。

ASN5107I **MONITOR** *monitor_qualifier*. The Replication Alert Monitor program stopped.

説明: レプリケーション・アラート・モニター・プログラムが終了しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5110I **MONITOR** *monitor_qualifier*. The Alert Monitor program encountered an SQL error. The **ERRCODE** is *error_code*. The **SQLSTATE** is *sqlstate*. The **SQLCODE** is *sqlcode*. The **SQLERRM** is *sqlerrm*. The **SQLERRP** is *sqlerrp*. The server name is *server_name*. The table name is *table_name*.

説明: SQL ステートメントの実行中にエラーが起きました。

ユーザーの処置: SQL エラー・コードの説明は、データベースのメッセージ・リファレンスを参照してください。

ASN5111I **MONITOR** *monitor_qualifier*. *number-of-rows* rows were pruned from the table *schema.table-name*.

説明: レプリケーション・アラート・モニター・プログラムは、IBMSNAP_ALERTS または IBMSNAP_MONTRACE 表から行のプルーニングを行ないました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5112E **MONITOR** *monitor_qualifier*. The pruning program failed with a SQLCODE of *sqlcode* when the program pruned the table *schema.table-name*.

説明: プルーニング操作の実行中に、プルーニング・プログラムが失敗しました。

ユーザーの処置: この SQLCODE に関するアクションを読み、エラーを訂正してください。

ASN5116E **MONITOR** *monitor_qualifier*. The monitor qualifier does not exist on the monitor control server *monitor-server*.

説明: レプリケーション・アラート・モニター・プログラムは、このモニター・コントロール・サーバーのモニター修飾子を見つけることができません。

ユーザーの処置: *monitor_qual* パラメーターに使用されているモニター修飾子名が正しいことを検査してください。

ASN5117E **MONITOR** *monitor_qualifier*. There are no valid Alert Conditions for this monitor qualifier on Monitor Server *monitor_server*.

説明: レプリケーション・アラート・モニター・プログラムは、このモニター修飾子のアラート条件を見つけることができません。

ユーザーの処置: *monitor_qual* パラメーターに使用されているモニター修飾子名が正しいことをチェックし、アラート条件が使用可能になっていることをチェックしてください。また、これ以前に出されているメッセージがないかチェックしてください。

ASN5118E **MONITOR** *monitor_qualifier*. The program cannot connect to the monitor control server *server_name*. The SQLCODE is *sqlcode*, and the SQLSTATE is *sqlstate*.

説明: モニター・プログラムはモニター・コントロール・サーバーに接続しようとし、示された SQLCODE で失敗しました。

ユーザーの処置: この SQLCODE に関するアクションを読み、エラーを訂正してください。

ASN5119E **MONITOR** *monitor_qualifier*. The program cannot connect to the server *server_name*. The SQLCODE is *sqlcode*, and the SQLSTATE is *sqlstate*.

説明: モニター・プログラムがモニター対象のキャプチャー・コントロール・サーバーまたはアプリ・コントロール・サーバーに接続しようとした時に、SQL CONNECT ステートメントが失敗しました。

ユーザーの処置: この SQLCODE に関するアクションを読み、エラーを訂正してください。

ASN5121E **MONITOR** *monitor_qualifier*. A contact does not exist for the alert condition with a component of *component*, a server of *server*, a schema or qualifier of *schema_or_qualifier*, and a condition name of *condition*.

説明: このアラート条件のための、指定された連絡先が IBMSNAP_CONTACTS 表にありません。

ユーザーの処置: 連絡先情報を検査し、アラート条件を訂正します。

ASN5122E MONITOR *monitor_qualifier*. The contact group *group-name* does not exist or is empty. The component is *component*, the server is *server*, the schema or qualifier is *schema_or_qualifier*, and the condition name is *condition*.

説明: アラート条件に指定された連絡先グループは、IBMSNAP_CONTACTGRP 表に対応する連絡先を持たないか、または IBMSNAP_CONTACTGRP 表にありません。連絡先グループを空にすることはできません。

ユーザーの処置: このグループの連絡先を検査し、アラート条件を訂正します。

ASN5123E MONITOR *monitor_qualifier*. The table *table-name* is not found. The Capture control server is *capture-server*. The schema is *schema*. The condition name is *condition-name*.

説明: レプリケーション・アラート・モニター・プログラムは、キャプチャー・コントロール・サーバーの条件のモニターを試みましたが、表が見つかりません。

ユーザーの処置: キャプチャー・コントロール・サーバーに表があることを検査するか、またはアラート条件を訂正します。

ASN5124E MONITOR *monitor_qualifier*. The table *table-name* is not found. The Apply control server is *apply-control-server*. The Apply qualifier is *apply-qualifier*. The subscription-set name is *set-name*. The condition name is *condition-name*.

説明: レプリケーション・アラート・モニター・プログラムは、アプライ・コントロール・サーバーの条件のモニターを試みましたが、表が見つかりません。

ユーザーの処置: アプライ・コントロール・サーバーに表があることを検査するか、またはアラート条件を訂正します。

ASN5125E MONITOR *monitor_qualifier*. The Apply qualifier *apply-qualifier* or the subscription set *set-name* is not found.

説明: レプリケーション・アラート・モニター・プログラムは、アプライ・コントロール・サーバーの条件のモニターを試みましたが、アプライ修飾子またはサブスクリプション・セットが見つかりません。

ユーザーの処置: アプライ・コントロール・サーバーにアプライ修飾子とサブスクリプション・セットがあることを検査するか、またはアラート条件を訂正します。

ASN5126E MONITOR *monitor_qualifier*. There is an error sending a notification. The return code is *rc*.

説明: アラート通知を試みた時、ASNMAIL 出口がエラーを戻しました。エラーの理由には、次のものがあります。

1 SMTP プロトコルが失敗しました。ご使

用の E メール・サーバーのアドレスを管理者に確認してください。

- 2 SMPT ソケットが失敗しました。ご使用の E メール・サーバーまたはクライアントの構成を検査してください。
- 3 E メール・アドレスが無効です。E メール・アドレスを検査します。
- 4 ソフトウェア・エラーです。
- 99 ASNMAIL 出口がありません。

ユーザーの処置: 戻りコード 1 および 2 の場合、ご使用の E メール・サーバーまたはクライアントの構成を検査してください。戻りコード 3 の場合、E メール・アドレスが正しいことをチェックしてください。戻りコード 99 の場合、ASNMAIL 出口が適切にインストールされていることを検査してください。

ASN5127E MONITOR *monitor_qualifier*. An invalid value *value* exists in the column *column-name* of the table *table-name*.

説明: このメッセージは、レプリケーション・アラート・モニター・プログラムが、プログラム初期化中に無効な値を持つ列を検出したことを示します。

ユーザーの処置: 指定された表の列定義の値を検査します。

ASN5129I MONITOR *monitor_qualifier*. The Replication Alert Monitor on server *server-name* reports an e-mail alert.

説明: レプリケーション・アラート・モニター・プログラムが E メール・アラートを送信しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5130I MONITOR *monitor_qualifier*. *capture_message*. The Capture control server is *capture-server*. The schema is *schema*. The monitor control server is *monitor-server*.

説明: レプリケーション・アラート・モニター・プログラムは、CAPTURE_ERRORS または CAPTURE_WARNINGS 条件の処理時に、IBMSNAP_CAPTRACE 表からキャプチャー・プログラム・メッセージを検索しました。

ユーザーの処置: キャプチャー・プログラム・メッセージを読み、該当のアクションをします。また、キャプチャー・コントロール・サーバーからのエラーまたは警告があるかどうかもチェックしてください。

ASN5131I MONITOR *monitor_qualifier*. *apply_message*. The Apply control server is *apply-server*. The Apply qualifier is *apply-qualifier*. The monitor control server is *monitor-server*.

説明: レプリケーション・アラート・モニター・プログラムは、APPLY_SUBSFAILING、APPLY_ERRORS、または APPLY_WARNINGS アラート条件の処理時に、IBMSNAP_APPLYTRAIL または IBMSNAP_APPLYTRACE 表からアプライ・プログラム・メッセージを検索しました。

ユーザーの処置: アプライ・プログラム・メッセージを読み、該当のアクションをします。また、アプライ・コントロール・サーバーからのエラーまたは警告があるかどうかもチェックしてください。

ASN5133I **MONITOR** *monitor_qualifier*. **The following alert** *message_number* **has occurred** *number_of_times* **times in the last** *number_of_minutes* **minutes. The notification for this alert will be suspended.**

説明: このメッセージは、MAX_NOTIFICATIONS_MINUTES パラメーター (デフォルトは 60 分) に指定された分数の、MAX_NOTIFICATIONS_PER_ALERT パラメーター (デフォルトは 3) に指定された回数の、アラートが検出された後に出されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5134I **MONITOR** *monitor_qualifier*. **Alerts issued.**

説明: このメッセージのテキストは、レプリケーション・アラート・モニター・プログラムが送信する E メール の件名行に表示されます。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5135W **MONITOR** *monitor_qualifier*. **Too many alerts** *number_of_alerts* **for server** *server_name* **between** *lower_bound_time* **and** *upper_bound_time*, **schema or qualifier** *schema_qual_name*, **condition name** *condition_name*.

説明: アラート・モニターは、モニター・サイクルに許された最大アラート (1024) に達したか、またはメモリーの割り振りができません。アラート・モニターは通知を送信し、モニター・コントロール・サーバーを更新し、サーバーに再接続して次の条件から開始します。指定されたアラート条件のいくつかのアラートは、モニター・コン

ロール・サーバーに送信および挿入されていない可能性があります。

ユーザーの処置: 指定されたアラート条件について、アラートが失われているかを、サーバーで直接チェックする必要があります。

ASN5136W **MONITOR** *monitor_qualifier*. **There has been an error calling DAS component. The return code is** *rc* **for server** *server_name* **for schema or qualifier** *schema_qual_name* **and condition name** *condition_name*.

説明: 示されているサーバーの条件を処理中に、DAS コンポーネントがエラーを戻しました。

ユーザーの処置: クライアントとリモート・サーバーの両方で DAS が適切に実行されていることを検査します。

ASN5150W **MONITOR** *monitor_qualifier*. **The Capture program is not running. The Capture control server is** *capture_server*, **and the schema is** *schema*.

説明: アラート条件 CAPTURE_STATUS は、キャプチャー・プログラムが実行されていないことを示しています。

ユーザーの処置: 指定されたキャプチャー・コントロール・サーバー上の、キャプチャー・プログラムの状況を検査してください。

ASN5151W MONITOR *monitor_qualifier*. The elapsed time since the last commit of Capture program exceeds the threshold value. The Capture control server is *capture_server*. The schema is *schema*. The last commit time is *time*. The threshold is *minutes* minutes.

説明: CAPTURE_LASTCOMMIT アラート条件は、現行タイム・スタンプ値と IBMSNAP_RESTART 表の MAX_COMMIT_TIME 列の値の差が、このアラート条件のしきい値 (IBMSNAP_CONDITIONS 表の PARM_INT 列に指定された値) より大きいことを検出しました。

ユーザーの処置: キャプチャー・コントロール・サーバーをチェックし、コミット遅延の理由を調べてください。

ASN5152W MONITOR *monitor_qualifier*. The current Capture latency exceeds the threshold value. The Capture control server is *capture_server*. The schema is *schema*. The Capture latency is *latency* seconds. The threshold is *threshold* seconds.

説明: CAPTURE_CLATENCY アラート条件は、IBMSNAP_RESTART 表の CURR_COMMIT_TIME 列と MAX_COMMIT_TIME 列の値の差が、このアラート条件のしきい値 (IBMSNAP_CONDITIONS 表の PARM_INT 列に指定された値) より大きいことを検出しました。

ユーザーの処置: キャプチャー・コントロール・サーバーをチェックし、キャプチャー待ち時間の理由を調べてください。

ASN5153W MONITOR *monitor_qualifier*. The historic Capture latency exceeds the threshold value. The Capture control server is *capture_server*. The schema is *schema*. The Capture latency is *latency* seconds. The threshold is *threshold* seconds.

説明: CAPTURE_HLATENCY アラート条件は、IBMSNAP_CAPMON 表の MONITOR_TIME 列と SYNCHTIME 列の値の差が、このアラート条件のしきい値 (IBMSNAP_CONDITIONS 表の PARM_INT 列の値) より大きいことを検出しました。この待ち時間の値は、過去の待ち時間のものである可能性があります、過去の待ち時間の傾向を追跡するのに役立つ場合があります。

ユーザーの処置: キャプチャー・コントロール・サーバーをチェックし、キャプチャー待ち時間の理由を調べてください。

ASN5154W MONITOR *monitor_qualifier*. The memory used by the Capture program exceeds the threshold value. The Capture control server is *capture_server*. The schema is *schema*. The amount of memory used is *memory* megabytes. The threshold is *threshold* megabytes.

説明: CAPTURE_MEMORY アラート条件は、IBMSNAP_CAPMON 表の CURRENT_MEMORY 列の値が、このアラート条件のしきい値 (IBMSNAP_CONDITIONS 表の PARM_INT 列の値) より大きいことを検出しました。

ユーザーの処置: キャプチャー・コントロール・サーバーをチェックし、過度のメモリー使用の理由を調べてください。必要ならば、キャプチャー・プログラムの `memory_limit` パラメーターを変更します。

ASN5160W MONITOR *monitor_qualifier*. **The Apply program is not running. The Apply control server is *apply_server*, and the Apply qualifier is *apply_qualifier*.**

説明: APPLY_STATUS アラート条件は、アプライ・プログラムが実行されていないことを検出しました。

ユーザーの処置: 指定されたアプライ・コントロール・サーバーで、アプライ・プログラムの状況を検査してください。

ASN5161W MONITOR *monitor_qualifier*. **The subscription set is inactive and is in an error state. The Apply control server is *apply-control-server*. The Apply qualifier is *apply_qualifier*. The name of the subscription set is *set-name*. The who's on first value is *wof*, and the lastsucces value is *timestamp*.**

説明: APPLY_SUBSINACT アラート条件は、サブスクリプション・セットが非アクティブであり、状況がゼロ (0) でないことを検出しました。

ユーザーの処置: サブスクリプション・セットをアクティブにする必要がある場合は、アプライ・コントロール・サーバーでこのサブスクリプション・セットをチェックします。

ASN5162W MONITOR *monitor_qualifier*. **A full refresh occurred. The Apply control server is *apply-control-server*. The Apply qualifier is *apply_qualifier*. The name of the subscription set is *set-name*. The who's on first value is *wof*, and the target table is *table-name*.**

説明: APPLY_FULLREFRESH アラート条件は、ターゲット表が過去のモニター・サイクル中にリフレッシュされたことを検出しました。

ユーザーの処置: フル・リフレッシュがエラーの場合は、この指定されたターゲット表のフル・リフレッシュの原因を調べてください。

ASN5163W MONITOR *monitor_qualifier*. **The subscription is delayed beyond the threshold. The Apply control server is *apply-control-server*. The Apply qualifier is *apply_qualifier*. The name of the subscription set is *set-name*. The who's on first value is *wof*. The time delayed is *time*, and the threshold is *threshold seconds*.**

説明: APPLY_SUBSDELAYED アラート条件は、「CURRENT TIMESTAMP 引く LASTRUN が、しきい値より大きい」という条件を満たすサブスクリプション・セットを検出しました。

ユーザーの処置: 前に出されたメッセージを見て、このサブスクリプション・セットにエラーがないか、およびアプライ・プログラムが実行されていることをチェックしてください。

ASN5164W MONITOR *monitor_qualifier*. The rows reworked in a subscription exceeds the threshold. The Apply control server is *apply-control-server*. The Apply qualifier is *apply-qualifier*. The name of the subscription set is *set-name*. The who's on first value is *wof*. The number of reworked rows is *rows*, and the threshold is *threshold* rows.

説明: APPLY_REWORKED アラート条件は、指定されたしきい値を超える SET_REWORKED 列の値 (IBMSNAP_APPLYTRAIL 表内) を持つサブスクリプション・セットを検出しました。

ユーザーの処置: この行数が再作業された理由を調べてください。

ASN5165W MONITOR *monitor_qualifier*. Transactions have been rejected in the subscription set. The Apply control server is *apply-control-server*. The Apply qualifier is *apply-qualifier*. The name of the subscription set is *set-name*. The who's on first value is *wof*. The number of rejected transactions is *transactions*.

説明: APPLY_TRANSREJECT アラート条件は、このサブスクリプション・セットにリジェクトされたトランザクションを検出しました。

ユーザーの処置: これらのトランザクションが拒否された理由を調べてください。

ASN5166W MONITOR *monitor_qualifier*. A manual full refresh is required. The Apply control server is *apply-control-server*. The Apply qualifier is *apply-qualifier*. The name of the subscription set is *set-name*.

説明: 指定されたサブスクリプション・セットにフル・リフレッシュが必要です。

ユーザーの処置: フル・リフレッシュが必要な理由を調べてください。

ASN5167W MONITOR *monitor_qualifier*. End-to-end latency exceeded the threshold. The Apply control server is *apply-control-server*. The Apply qualifier is *apply-qualifier*. The name of the subscription set is *set-name*. End-to-End latency is *latency* seconds, and the threshold is *threshold* seconds.

説明: APPLY_LATENCY アラート条件は、このサブスクリプション・セットのエンドツーエンドの所要時間が、このアラート条件のしきい値 (IBMSNAP_CONDITIONS 表の PARM_INT 列に指定された値) より大きいことを検出しました。

ユーザーの処置: このような過度のエンドツーエンドの所要時間値の理由を、アプライ・コントロール・サーバーで調べてください。

ASN5190W MONITOR *monitor_qualifier*. A stored procedure failed with an SQLCODE of *sqlcode* and with an SQL message of *sql_message*.

説明: ユーザー定義のアラート条件 (DB2 ストアド・プロシージャーとして作成されたもの) がゼロ以外の SQLCODE 値を戻しました。

ユーザーの処置: アクションは、ストアド・プ

ロシージャーがどのように作成されているかにより異なります。メッセージが異常条件を示している場合は、DBA に連絡してください。

ASN5191W MONITOR *monitor_qualifier*.
message

説明: このメッセージは、ユーザー定義のアラート条件が警告メッセージを出した場合に表示されます。

ユーザーの処置: 表示されたメッセージを読み、該当のアクションをします。

ASN5192E MONITOR *monitor_qualifier*.
message

説明: このメッセージは、ユーザー定義のアラート条件がエラー・メッセージを出した場合に表示されます。

ユーザーの処置: 表示されたメッセージを読み、該当のアクションをします。

ASN5200E ASNSCRT: The replication process type is a required parameter and must be specified when invoking the asnsCRT command.

説明: レプリケーション処理のタイプを指定せずに、asnsCRT コマンドが呼び出されました。

ユーザーの処置: レプリケーション処理のタイプとして -C、-A、または -M を指定し、コマンドを再度入力します。

ASN5201E ASNSCRT: The database instance is a required parameter and must be specified when invoking the asnsCRT command.

説明: データベース・インスタンスを指定せずに、asnsCRT コマンドが呼び出されました。

ユーザーの処置: データベース・インスタンス名を指定して、コマンドを再度入力します。

ASN5202E ASNSCRT: A replication process path is a required parameter and must be specified when invoking the asnsCRT command.

説明: レプリケーション処理のパスを指定せずに、asnsCRT コマンドが呼び出されました。

ユーザーの処置: asncap、asnapply、または asnmon コマンドへのパスを指定して、コマンドを再度入力します。

ASN5203E ASNSCRT: The Capture server is a required parameter and must be specified when invoking this asnsCRT command.

説明: キャプチャー・コントロール・サーバーを指定せずに、asnsCRT コマンドが呼び出されました。

ユーザーの処置: キャプチャー・コントロール・サーバー名を指定して、コマンドを再度入力します。

ASN5204E ASNSCRT: The Apply control server is a required parameter and must be specified when invoking this asnsCRT command.

説明: アプライ・コントロール・サーバーを指定せずに、asnsCRT コマンドが呼び出されました。

ユーザーの処置: アプライ・コントロール・サーバー名を指定して、コマンドを再度入力します。

ASN5205E ASNSCRT: The Apply qualifier is a required parameter and must be specified when invoking this asnsrct command.

説明: アプライ修飾子を指定せずに、asnsrct コマンドが呼び出されました。

ユーザーの処置: アプライ修飾子を指定して、コマンドを再度入力します。

ASN5206E ASNSCRT: The monitor control server is a required parameter and must be specified when invoking this asnsrct command.

説明: モニター・コントロール・サーバーを指定せずに、asnsrct コマンドが呼び出されました。

ユーザーの処置: モニター・コントロール・サーバー名を指定して、コマンドを再度入力します。

ASN5207E ASNSCRT: The monitor qualifier is a required parameter and must be specified when invoking this asnsrct command.

説明: モニター修飾子を指定せずに、asnsrct コマンドが呼び出されました。

ユーザーの処置: モニター修飾子を指定して、コマンドを再度入力します。

ASN5208I ASNSCRT: The replication service *service_name* was created successfully.

説明: asnsrct コマンドは指定されたサービスを正常に作成しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5209I ASNSCRT: The replication service *service_name* started successfully.

説明: asnsrct コマンドは指定されたサービスを正常に開始しました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5210E ASNSCRT: The replication service *service_name* was not created, because the display name already exists (either as a service name or as another display name) in the service control manager database.

説明: asnsrct コマンドは指定されたサービスを作成できません。その理由は、表示名が別のサービス名または表示名として、サービス・コントロール・マネージャーのデータベースにすでに存在するためです。

ユーザーの処置: サービス・コントロール・マネージャーのデータベースで、重複するサービス名または表示名を持つサービスを除去します。それから、コマンドを再入力します。

ASN5211E ASNSCRT: The replication service *service_name* was not created, because the specified service name is not valid.

説明: asnsrct コマンドは指定されたサービスを作成できません。その理由は、システム API が、サービス名が誤りであることを示すエラー・コードを戻したためです。指定されたサービスには、インスタンス名、データベース名、またはスキーマ名に特殊文字が含まれている可能性があります。サービス名には特殊文字は使用できません。

ユーザーの処置: 可能ならば、インスタンス名、データベース名、またはスキーマ名を変更してく

ださい。それから、コマンドを再入力します。

ASN5212E ASNSCRT: The replication service *service_name* was not created, because the specified service name already exists.

説明: asnsCRT コマンドは指定されたサービスを作成できません。その理由は、同じサービス名を持つ別のサービスが、サービス・コントロール・マネージャーにすでに存在するためです。

ユーザーの処置: 同じサービス名を持つ既存のサービスを除去します。それから、コマンドを再入力します。

ASN5213E ASNSCRT: The replication service *service_name* was not started, because the service binary file could not be found.

説明: asnsCRT コマンドは指定されたサービスを開始できません。その理由は、PATH 環境変数に指定されたシステム・パスを使用して、対応する asncap、asnapply、または asnmon コマンドを呼び出すことができないためです。完全修飾されたパスを指定した場合は、asnsCRT コマンドはそのパス内に asncap、asnapply または asnmon コマンドを見つけることができませんでした。

ユーザーの処置: 指定されたパスが正しいことを確認してください。それから、コマンドを再入力します。

ASN5214E ASNSCRT: The replication service *service_name* did not start, because an instance of the service is already running.

説明: asnsCRT コマンドは指定されたサービスを開始できません。その理由は、サービスがすでに実行中であるためです。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5215E ASNSCRT: The replication service *service_name* did not start, because the service depends on a DB2 instance service that does not exist or has been marked for deletion.

説明: asnsCRT コマンドは指定されたサービスを開始できません。その理由は、対応する DB2 インスタンス・サービスが存在しない、または削除されたためです。

ユーザーの処置: 対応する DB2 インスタンス・サービスがサービス・コントロール・マネージャーに存在することを検査してください。それから、コマンドを再入力します。

ASN5216E ASNSCRT: The replication service *service_name* did not start, because this service depends on another service that failed to start.

説明: asnsCRT コマンドは指定されたサービスを開始できません。その理由は、対応する DB2 インスタンス・サービスが開始できなかったためです。

ユーザーの処置: 対応する DB2 インスタンス・サービスがサービス・コントロール・マネージャーで開始されていることを検査してください。それから、コマンドを再入力します。

ASN5217E ASNSCRT: The replication service *service_name* did not start, because the service is disabled.

説明: asnsCRT コマンドは指定されたサービスを開始できません。その理由は、サービス・コントロール・マネージャー内でそのサービスが使用不可になっているためです。

ユーザーの処置: サービス・コントロール・マネージャー内で、サービスの始動タイプが自動また

は手動にセットされていることを検査してください。それから、コマンドを再入力します。

ASN5218E ASNSCRT: The replication service *service_name* did not start, because the service cannot log on. This error occurs if the service starts from an account that does not have the proper "Log on as a service" access right.

説明: asnsrct コマンドは指定されたサービスを開始できません。その理由は、対応する DB2 インスタンス・サービスがログオンできないためです。

ユーザーの処置: サービス・コントロール・マネージャーで、指定されたサービスを探します。そこで提供されたアカウント名とパスワードが正しいことを確認します。それから、コマンドを再入力します。

ASN5219E ASNSCRT: The replication service *service_name* did not started, because the service is marked for deletion.

説明: asnsrct コマンドは指定されたサービスを開始できません。その理由は、サービスが削除されたためです。

ユーザーの処置: サービス・コントロール・マネージャーのウィンドウをクローズします。それから、コマンドを再入力します。

ASN5220E ASNSDROP: The service name is a required parameter and must be specified when invoking the asnsdrop command.

説明: サービス名を指定せずに、asnsdrop コマンドが呼び出されました。

ユーザーの処置: サービス名を指定して、コマンドを再入力します。

ASN5221I ASNSDROP: The service *service_name* has been successfully removed.

説明: サービス名を指定して、asnsdrop コマンドが呼び出されました。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5222E ASNSDROP: The replication service *service_name* cannot be removed, because the requested access is denied.

説明: asnsdrop コマンドは指定されたサービス名を除去できません。その理由は、ユーザーがそれを除去するための適切な許可を持っていないためです。

ユーザーの処置: 現行ユーザーが、対応する DB2 インスタンスにログオンするための許可を持っていることを検査してください。それから、コマンドを再入力します。

ASN5223E ASNSDROP: The replication service *service_name* cannot be removed, because the specified service name is not valid.

説明: asnsdrop コマンドは指定されたサービス名を除去できません。その理由は、サービス名に使用禁止の特殊文字が含まれるためです。

ユーザーの処置: サービス・コントロール・マネージャーで、指定されたサービスを探します。サービス名が正しいことを検査し、コマンドを再入力します。

ASN5224E ASNSDROP: The replication service *service_name* cannot be removed, because the specified service does not exist.

説明: `asnsdrop` コマンドは指定されたサービス名を除去できません。その理由は、サービス名がサービス・コントロール・マネージャーにないためです。

ユーザーの処置: サービス・コントロール・マネージャーで、指定されたサービスを探します。サービス名が正しいことを検査し、コマンドを再入力します。

ASN5225E ASNSDROP: The replication service *service_name* cannot be stopped, because other running services are dependent on it. The *service_name* is not removed.

説明: `asnsdrop` コマンドは指定されたサービスを除去できません。その理由は、この指定されたサービスに從属する、他のサービスが現在実行されているためです。

ユーザーの処置: サービス・コントロール・マネージャーで、この指定されたサービスに從属するすべてのサービスを停止します。それから、コマンドを再入力します。

ASN5226E ASNSDROP: The replication service *service_name* cannot be removed, because the system is shutting down.

説明: `asnsdrop` コマンドは指定されたサービスを除去できません。その理由は、オペレーティング・システムがシャットダウンしているためです。

ユーザーの処置: システムが再始動された後、コマンドを再度入力します。

ASN5227I ASNSDROP: The replication service *service_name* cannot be removed, because it has already been marked for deletion.

説明: `asnsdrop` コマンドは指定されたサービスを除去できません。その理由は、指定されたサービスがすでに削除されているためです。

ユーザーの処置: このメッセージは通知のみであり、アクションは必要ありません。

ASN5228E *pgmname* : The command cannot command *action* the replication service *service_name*, because the system call *API_func_name* returned an unexpected error code *error_code*.

説明: `asnsrct` および `asndrop` コマンドは、サービスを処理するため、システム呼び出しを行いません。指定されたシステム呼び出しは予期しないエラー・コードを戻したため、指定されたコマンドは要求されたアクションを完了できませんでした。

ユーザーの処置: コマンドを再度入力します。このエラー・コードは、単に一時的なシステム条件を示している場合があります。エラー・コードについての詳細は、使用しているオペレーティング・システムの資料を調べてください。

ASN5229E ASNSCRT: The account is a required parameter and must be specified when invoking the `asnsrct` command.

説明: 対応する DB2 インスタンスのアカウント名を指定せずに、`asnsrct` コマンドが呼び出されました。

ユーザーの処置: 対応する DB2 インスタンスのアカウント名を指定して、コマンドを再入力します。

ASN5230E ASNSCRT: The password is a required parameter and must be specified when invoking the asnsCRT command.

説明: 対応する DB2 インスタンスのパスワードを指定せずに、asnsCRT コマンドが呼び出されました。

ユーザーの処置: 対応する DB2 インスタンスのパスワードを指定して、コマンドを再入力します。

ASN5231E ASNSCRT: The replication service *service_name* did not start, because the account name specified on the account parameter does not exist.

説明: 対応する DB2 インスタンスの不明のアカウント名を指定して、asnsCRT コマンドが呼び出されました。

ユーザーの処置: サービス・コントロール・マネージャで、指定されたサービスを探します。そこで提供されたアカウント名とパスワードが正しいことを確認します。それから、コマンドを再入力します。

ASN5232E ASNSCRT: The required parameter *path* was not specified.

説明: サービスとして開始する時に、レプリケーション・コマンドには、コマンドに特有のパス・キーワードでパスを指定する必要があります (asncap ならば *capture_path*、asnapply ならば *apply_path*、asnmon ならば *monitor_path*)。パス・キーワードを指定すると、エラーが起これなければ、サービスは登録されます。

パス・キーワードを指定しないと、asnsCRT コマンドは DB2 グローバル登録プロファイル変数 DB2PATH を検索します。この変数に NULL 以外の値が含まれる場合、asnsCRT コマンドは

DB2PATH の値を使用して、コマンドに該当のパス・キーワードを追加します。この変数がセットされていない場合、asnsCRT コマンドはサービスを登録できません。

ユーザーの処置: 該当のパス・キーワードを指定するかまたは、DB2 グローバル登録プロファイル変数 DB2PATH を定義した後で、コマンドを再度入力します。

付録 A. z/OS におけるユニコードおよび ASCII のコード化スキーム

DB2 DataPropagator for OS/390 および DB2 DataPropagator for z/OS バージョン 7 以降では、ユニコードと ASCII のコード化スキームをサポートします。ユニコードのコード化スキームを活用するには、少なくとも DB2 for OS/390 および DB2 for z/OS バージョン 7 が必要で、以下のセクションで説明するように、DB2 DataPropagator のソース、ターゲット、およびコントロールの各表を手動で作成しなければなりません。ただし、既存のレプリケーション環境は、コード化スキームが変更されていなければ、DB2 DataPropagator for OS/390 および DB2 DataPropagator for z/OS バージョン 7 以降で機能します。使用しているシステムがユニコード・システムの場合は、キャプチャー、アプライ、およびレプリケーション・アラート・モニターの各プログラムの BIND PLAN コマンドと PACKAGE コマンドに、ENCODING(EBCDIC) を付け足さなければなりません。

コード化スキームの選択

ソース表、CD 表、およびターゲット表で同じコード化スキームを使用していると、レプリケーション環境でのデータ変換の必要性を最小にすることができます。これらの表のコード化スキームを選択するときは、1 つの CCSID 規則に従ってください。

表スペース・データは ASCII、EBCDIC、またはユニコードの CCSID を使ってエンコードされます。1 つの表スペース内のすべての表は、同じコード化スキームを使用しなければなりません。ある SQL ステートメントで参照されるすべての表のコード化スキームは同じでなければなりません。また、表示または結合して使用する表もすべて、同じコード化スキームを使用しなければなりません。

単一の CCSID 規則に従わないと、DB2 はバインドまたは実行中に違反を検出して SQLCODE -873 を戻します。

どの表を ASCII またはユニコードにする必要があるかを決定するのは、クライアント / サーバー構成です。具体的には、表のコード化スキームを選択するとき、以下の規則に従ってください。

- DB2 for OS/390 のソースまたはターゲット表は EBCDIC、ASCII、またはユニコードにする。これらの表は、サポートされているどの DBMS (DB2 ファミリー、または DataJoiner を使用した非 DB2) の同一または異なるコード化スキームを持つ表とも、相互にコピーを行うことができます。
- DB2 for OS/390 ソース・サーバーで、サブスクリプション・セットが作成されるときにターゲット・タイプが USERCOPY で JOIN_UOW_CD=Y の場合は、同じサーバ

一上の CD 表と UOW 表が同じコード化スキームを使用する必要はない。そうでない場合は、CD 表と UOW 表は同じコード化スキームを使用しなければなりません。

- シグナル (IBMSNAP_SIGNAL) 表は、キャプチャー・プログラムがシグナルをその表から選択するときに EBCDIC に変換する必要がないように、EBCDIC になっている必要がある。
- 同じコントロール・サーバー上のすべてのコントロール表 (ASN.IBMSNAP_SUBS_xxxx) は、同じコード化スキームを使用しなければならない。
- その他のコントロール表は、任意のコード化スキームを使用できる。

コード化スキームの設定

表に適切なコード化スキームを指定するには、表の生成に使用される SQL を変更します。

1. 適切なコード化スキームを使用して新しいソース表とターゲット表を作成するか、あるいは既存のターゲット表とソース表のコード化スキームを変更する。既存の表のコード化スキームを変更するときは、その前にキャプチャー・プログラムとアプライ・プログラムを停止しておき、その後でキャプチャー・プログラムをコールド・スタートし、アプライ・プログラムを再始動するようお勧めします。既存の表のコード化スキーム (1 つの表スペース内では同一のコード化スキームを持っていないければなりません) のコード化スキームを変更するには、以下のようにします。
 - a. Reorg Tablespace ユーティリティを使用して、既存の表スペースをアンロードする。
 - b. 既存の表スペースをドロップする。
 - c. 新しいコード化スキームを指定する表スペースを再作成する。
 - d. Load ユーティリティを使用して、古いデータを新しい表スペースにロードする。

Load と Reorg の各ユーティリティについての詳細は、「DB2 (OS/390 & z/OS) ユーティリティ・ガイド」(SC88-8773) を参照してください。

2. レプリケーション・センターを使用して、適切なコード化スキームを持つ新しいコントロール表を作成する。
3. Reorg ユーティリティと Load ユーティリティを使用して、既存のコントロール表と CD 表のコード化スキームを変更する。
4. レプリケーション・センターを使用して新しいレプリケーション・ソースまたはサブスクリプション・セットを作成するときに、適切なコード化スキームを指定する。

CCSID についての詳細は、「DB2 UDB サーバー (OS/390) SQL 解説書」(SD88-7293) にあります。

付録 B. キャプチャー・プログラムがジャーナル・エントリー・タイプを処理する方法 (iSeries)

以下の表では、キャプチャー・プログラムが様々なジャーナル・エントリー・タイプを処理する方法について説明します。

表 100. ジャーナル・エントリーによるキャプチャー・プログラムの処理

| ジャーナル・ コード ³ | エントリー・ タイプ | 記述 | 処理 |
|----------------------------|---------------|--|--|
| C | CM | コミット済みのレコード変更のセット | UOW 表にレコードを挿入します。 |
| C | RB | ロールバック | 挿入された UOW 行はありません。 |
| F | AY | 物理ファイル・メンバーに適用されたジャーナル済みの変更 | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | CE | 物理ファイルのデータの終わりの変更 | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | CR | 消去された物理ファイル・メンバー | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | EJ | 終了した物理ファイル・メンバーのジャーナリング | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | IZ | 初期化された物理ファイル・メンバー | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | MD | 物理ファイルから削除されたメンバー (DLTLIB、DLTF、または RMVM) | ASN2004 メッセージを発行し、フル・リフレッシュを試行します。 |
| F | MF | 解放された物理ファイル・メンバーのストレージ | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | MM | 移動されたメンバーを含む物理ファイル (ライブラリーのオブジェクトの名前変更 (RNMOBJ)、ファイルのオブジェクトの移動 (MOVOBJ)) | ASN200A メッセージを発行し、フル・リフレッシュを試行します。 |
| F | MN | 名前変更されたメンバー (ファイルの RNMOBJ、メンバーの名前変更 (RNMM)) を含む物理ファイル | ASN200A メッセージを発行し、フル・リフレッシュを試行します。 |
| F | MR | リストアされた物理ファイル・メンバー | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | RC | 物理ファイル・メンバーから削除されたジャーナル済みの変更 | ASN2004 メッセージを発行し、ファイルのフル・リフレッシュを行います。 |
| F | RG | 再編成された物理ファイル・メンバー | レプリケーション・キーとしてソース表の RRN が使用されている場合には、ASN2004 メッセージを発行して、ファイルのフル・リフレッシュを行います。 |
| J | NR | 次のジャーナル・レシーバーの ID | キャプチャー・プログラムをリセットします。 |

表 100. ジャーナル・エントリーによるキャプチャー・プログラムの処理 (続き)

| ジャーナル・ コード ³ | エントリー・ タイプ | 記述 | 処理 |
|----------------------------|---------------|-----------------------------------|----------------------------|
| J | PR | 直前のジャーナル・レシーバーの ID | 固有のシーケンス番号カウンターを増分 します。 |
| R | DL | 物理ファイル・メンバーから削除され たレコード | DLT レコードを CD 表に挿入します。 |
| R | DR | ロールバックで削除されたレコード | DLT レコードを CD 表に挿入します。 |
| R | PT | 物理ファイル・メンバーに追加された レコード | ADD レコードを CD 表に挿入します。 |
| R | PX | 物理ファイル・メンバーに直接追加さ れたレコード | ADD レコードを CD 表に挿入します。 |
| R | UB | 物理ファイル・メンバーで更新された レコードの変更前イメージ | 注 1 を参照してください。 |
| R | UP | 物理ファイル・メンバーで更新された レコードの変更後イメージ | 注 1 を参照してください。 |
| R | BR | ロールバックで更新されたレコードの 変更前イメージ | 注 2 を参照してください。 |
| R | UR | ロールバックで更新されたレコードの 変更後イメージ | 注 2 を参照してください。 |

注:

- 登録表の PARTITION_KEYS_CHG 列が N の場合、R-UP イメージおよび R-UB イメージが単一の UPD レコードを CD 表に形成します。そうでない場合には、R-UB イメージが DLT レコードを CD 表に挿入し、R-UP イメージが ADD 表を CD 表に挿入します。
- 登録表の PARTITION_KEYS_CHG 列が N の場合、R-UR イメージおよび R-BR イメージが単一の UPD レコードを CD 表に形成します。そうでない場合には、R-BR イメージが DLT レコードを CD 表に挿入し、R-UR イメージが ADD 表を CD 表に挿入します。
- 以下の値が、ジャーナル・コードに使用されます。
 - C** コミットメント・コントロール操作
 - F** データベース・ファイル操作
 - J** ジャーナルまたはジャーナル・レシーバー操作
 - R** 特定のレコードに対する操作

他のジャーナル・エントリー・タイプはすべて、キャプチャー・プログラムによって無視されます。

付録 C. アプリケーション (UNIX、Windows) 内部からのレプリケーション・プログラムの始動

レプリケーションの 1 サイクルにおけるレプリケーション・プログラム (キャプチャー・プログラム、アプライ・プログラム、およびレプリケーション・アラート・モニター) はどれも、アプリケーションの内部から呼び出しルーチンで始動できます。API が同期実行のみをサポートしているため、これらのルーチンを使用するには、キャプチャー・プログラムの場合は AUTOSTOP オプション、アプライ・プログラムの場合は COPYONCE オプションを指定しなければなりません。

API のサンプルとそのおのおの MAKE ファイルは以下のディレクトリーにあります。

NT の場合

sqllib¥samples¥repl

UNIX の場合

sqllib/samples/repl

上記のディレクトリーには、キャプチャー・プログラムを始動するための以下のファイルが含まれています。

capture_api.c

Windows または UNIX でキャプチャー・プログラムを始動するためのサンプル・コード。

capture_api_nt.mak

Windows でのサンプル・コードの MAKE ファイル。

capture_api_unix.mak

UNIX でのサンプル・コードの MAKE ファイル。

上記のディレクトリーには、アプライ・プログラムを始動するための以下のファイルが含まれています。

apply_api.c

Windows または UNIX でアプライ・プログラムを始動するためのサンプル・コード。

apply_api_nt.mak

Windows でのサンプル・コードの MAKE ファイル。

apply_api_unix.mak

UNIX でのサンプル・コードの MAKE ファイル。

上記のディレクトリーには、レプリケーション・アラート・モニターを始動するための以下のファイルが含まれています。

monitor_api.c

Windows または UNIX でレプリケーション・アラート・モニターを始動するためのサンプル・コード。

monitor_api_nt.mak

Windows でのサンプル・コードの MAKE ファイル。

monitor_api_unix.mak

UNIX でのサンプル・コードの MAKE ファイル。

用語集

[ア行]

アーカイブ・ログ (archive log). (1) クローズされ、通常の処理には必要なくなったログ・ファイルの集まり。 ロールフォワード・リカバリーで使うために保存される。(2) アクティブ・ログからコピーされたログ・レコードが入っている DB2 Universal Database for z/OS and OS/390 ログの部分。古くなってアクティブ・ログには適さなくなったレコードが保留されている。

アプライ修飾子 (Apply qualifier). アプライ・プログラムの各インスタンスごとに固有のレプリケーション・サブスクリプション・セットを識別する大文字小文字の区別のある文字ストリング。

アプライ待ち時間 (Apply latency). レプリケーションが 1 つのサイクルを完了するために必要なおおよその時間。「キャプチャー待ち時間 (Capture latency)」も参照。

アプライ・コントロール・サーバー (Apply control server). アプライ・コントロール表が含まれるデータベースで、サブスクリプション・セットおよびそのメンバーに関する情報を保管する。「アプライ・サーバー (Apply server)」と対比。

アプライ・サーバー (Apply server). アプライ・プログラムが実行中のシステム。「アプライ・コントロール・サーバー (Apply control server)」と対比。

アプライ・サイクル (Apply cycle). データがソース表からターゲット表へ複製される間の時間間隔。

アプライ・プログラム (Apply program). ソース対ターゲットの適用規則に応じてレプリケーション・ターゲット表を最新表示したり更新したり

するのに使われるプログラム。「キャプチャー・プログラム (Capture program)」および「キャプチャー・トリガー (Capture trigger)」と対比。

アラート条件 (alert condition). レプリケーションをモニターするユーザー指定条件で、この条件に合致した場合、レプリケーション・アラート・モニターは、エラーが発生したか、あるいは操作可能しきい値に到達したことを、個別またはグループ連絡先に通知を送信することになる。

異機種のレプリケーション (heterogeneous replication). DB2 と非 DB2 リレーショナル・データベース間のレプリケーション。「連合データベース・システム (federated database system)」も参照。

一時表 (temporary table). 一時データを保留する表。たとえば、大量の行が含まれる照会からの中間結果を保留またはソートするのに有効である。別の SQL ステートメントによって作成されるものと、作成済みの一時表と宣言済みの一時表から作成される 2 種類の一時表がある。

イベント・タイミング (event timing). レプリケーション・サブスクリプション・サイクルを開始するタイミングを制御する最も精密なメソッド。イベント・タイミングを使用するには、処理を行うイベント名と時間を指定する必要がある。「インターバル・タイミング (interval timing)」と対比。

インターバル・タイミング (interval timing). サブスクリプション・サイクルを開始するタイミングを制御する最も簡単な方法。インターバル・タイミングを使用するには、ユーザーは、サブスクリプション・サイクルを開始する日付と時刻を指定し、サブスクリプション・サイクルを実行す

る頻度を示す時間間隔を設定する必要がある。
「イベント・タイミング (*event timing*)」と対比。

ウォーム・スタート (warm start). (1) 以前に初期化された入出力作業キューの再使用を可能にする再始動。(2) DB2 レプリケーションでは、キャプチャー・コントロール表から既存のデータを使用して、キャプチャー・プログラムを開始するプロセス。「コールド・スタート (*cold start*)」と対比。

エンドツーエンド待ち時間

(end-to-end latency). レプリケーションが変更をキャプチャーし、それをターゲット・データベースにアプライするために必要なおおよその時間。「アプライ待ち時間 (*Apply latency*)」および「キャプチャー待ち時間 (*Capture latency*)」も参照。

オブジェクト (object). (1) SQL で作成または操作できるもの。たとえば、表、ビュー、索引、パッケージなど。(2) オブジェクト指向設計またはオブジェクト指向プログラミングでは、データとそのデータに関連付けられた操作からなる抽象的なもの。(3) NetWare では、ネットワーク上で定義されたエンティティーで、ファイル・サーバーへのアクセスを持つ。(4) インフォメーション・カタログ・センターでは、単位または特殊な情報のグループを示す項目。インフォメーション・カタログ・センターの各オブジェクトは情報を識別して記述するが、実際の情報は含まれない。たとえば、オブジェクトはレポートの名前を提供し、その作成日をリストし、さらにその目的を記述する。

【力行】

外部 CCD 表 (external CCD table). 登録されたレプリケーション・ソースであるために、直接サブスクライブできる CCD 表。その独自の行がレジスター表の中に存在し、レジスター表では SOURCE_OWNER および SOURCE_TABLE 列として識別されている。「整合変更データ表

(*consistent-change data table*)」も参照。「内部 CCD 表 (*internal CCD table*)」と対比。

カスケード・リジェクト (cascade rejection). 競合が検出され、しかもそれ自身がリジェクトされたトランザクションと関連しているためにレプリケーション・トランザクションをリジェクトするプロセス。

監査証跡 (audit trail). イベントのシーケンスをリンクする論理パスの書式データで、レコードの内容を反映したトランザクションをトレースするために使用される。

キー (key). 表、索引、または参照制約の記述内で識別される列または順番に並べた列の集合。同じ列が複数のキーの一部となることができる。

基礎集約表 (base aggregate table). レプリケーション・ソース表から集約されるデータが含まれるレプリケーション・ターゲット表のタイプ。アプライ・プログラムが集約を実行する時刻にマークを付けるタイム・スタンプが含まれる。「変更集約表 (*change aggregate table*)」と対比。

ギャップ (gap). キャプチャー・プログラムが一定範囲のログまたはジャーナル・レコードを読み取ることができずに、変更データを失う可能性のある状態。

キャプチャー待ち時間 (Capture latency). キャプチャー・プログラムが、CD 表にどのくらい前にデータをコミットしたかを示すおおよその測定。「アプライ待ち時間 (*Apply latency*)」も参照。

キャプチャー・コントロール・サーバー (Capture control server). (1) キャプチャー・コントロール表が含まれるデータベースで、登録済みレプリケーション・ソース表に関する情報を保管する。(2) キャプチャー・プログラムが実行中のシステム。

キャプチャー・スキーマ (Capture schema).

キャプチャー・プログラムの特別なインスタンスで使用されるキャプチャー・コントロール表のスキーマ。

キャプチャー・トリガー (Capture trigger).

DB2 以外のリレーショナル・ソース表に対して実行される削除、挿入、および更新操作を収集するためのメカニズム。「キャプチャー・プログラム (Capture program)」および「アプライ・プログラム (Apply program)」と対比。

キャプチャー・プログラム (Capture program).

DB2 ソース表に加えられた変更を取り込むために、データベース・ログまたはジャーナルの記録を読み取るプログラム。「アプライ・プログラム (Apply program)」および「キャプチャー・トリガー (Capture trigger)」と対比。

行キャプチャー規則 (row-capture rules).

キャプチャー・プログラムが行を CD 表にいつ書き込むか、書き込むかどうか、またはキャプチャー・トリガーが行を CCD 表にいつ書き込むか、書き込むかどうかを定義した登録済み列への変更を基にした規則。

競合検出 (conflict detection). update-anywhere レプリケーション構成では、次のプロセスのいずれかを競合検出が参照する。

- 制約エラーの検出プロセス。
- ソース表とターゲット表の両方の同じ行が、同じレプリケーション・サイクルの中でユーザーまたはアプリケーションによって更新されたかどうかを検出するプロセス。競合が検出されると、その競合の原因になったトランザクションはリジェクトされる。

許可トークン (authorization token). (1) トランザクションに関連したトークン。(2) DB2 Universal Database for z/OS では相関 ID を指す。(3) DB2 Universal Database for iSeries では、トランザクションの原因となるジョブの名前を指す。

区切り ID (delimited identifier). 引用符 (") で囲まれたストリング。このストリングは、1 文字とその後につづくゼロ個以上の文字 (そのおのおのが文字、数字、または下線文字) で構成されていなければならない。「通常識別子 (ordinary identifier)」も参照。

クライアント (client). データベース・サーバーと通信したりアクセスしたりするプログラム (またはプログラムが稼動するワークステーション)。

グローバル・レコード (global record). キャプチャー・プログラムの特別なインスタンスに対してグローバル・レプリケーション特性を定義する登録表の行。

結合 (join). 列値を突き合わせて複数の表からデータを取り出すための SQL 関係操作。

コールド・スタート (cold start). (1) DB2 レプリケーションでは、キャプチャー・プログラムの前の操作からの再始動情報を使用しないで、キャプチャー・プログラムを開始するプロセス。「ウォーム・スタート (warm start)」と対比。(2) 初期プログラム・ロード・プロシージャを使用して、システムまたはプログラムを始動するプロセス。(3) DB2 Universal Database for z/OS and OS/390 がどのログ・レコードも処理しないで再始動するときのプロセス。

固有索引 (unique index). 表に同一のキー値がないことを保証する索引。

コンデンス CCD 表 (condensed CCD table). 行の最新の値のみを含み、各キー値ごとに 1 つの行のみを持つ CCD 表。「非コンデンス CCD 表 (noncondensed CCD table)」と対比。「整合変更データ表 (consistent-change data (CCD) table)」も参照。

コンデンス (condensed). データに加えられた変更のヒストリーではなく、現行データが表に入っていることを示す表属性。コンデンス表には、表内の各主キー値につき 1 つ以上の行が含

まれることはない。そのため、コンデンス表は、最新表示のための現行情報を提供するのに使うことができる。

コントロール表 (control table). 「レプリケーション・コントロール表 (replication control table)」を参照。

コントロール・サーバー (control server). キャプチャー、アプライ、またはモニター・プログラムに対するレプリケーション・コントロール表を含むデータベース・サーバー。「アプライ・コントロール・サーバー (Apply control server)」、「キャプチャー・コントロール・サーバー (Capture control server)」、および「モニター・コントロール・サーバー (Monitor control server)」も参照。

コントロール・センター (Control Center). データベース・オブジェクト (データベースや表など) とその相互関係を示すグラフィカル・インターフェース。コントロール・センターから、DBA ユーティリティ、Visual Explain、およびパフォーマンス・モニター・ツールで提供されるタスクを実行することができる。

コンプリート CCD 表 (complete CCD table). CCD 表の初期状態では、レプリケーション・ソース表またはビューからのすべての行と、ソース表またはビューからの任意の述部が含まれる。「非コンプリート CCD 表 (noncomplete CCD table)」と対比。「整合変更データ表 (consistent-change data (CCD) table)」も参照。

[サ行]

サーバー (server). 「データベース・サーバー (database server)」を参照。

再作業 (rework). (1) レプリケーション・ターゲット表への挿入を、ターゲット表にすでに行が存在するために挿入が失敗した場合、更新へ変換すること。(2) レプリケーション・ターゲット表へ

の更新を、ターゲット表に行が存在しないために更新が失敗した場合、挿入へ変換すること。

作業単位 (unit of work). (1) アプリケーション・プロセスにおけるリカバリー可能な一連の操作。アプリケーション・プロセスは、常に単一の作業単位であるが、コミットまたはロールバック操作のため、アプリケーション・プロセス全体には複数の作業単位が関与する。DB2 Universal Database for z/OS and OS/390 マルチサイト更新操作では、単一の作業単位に複数のリカバリー単位 (unit of recovery) を組み込むことができる。「トランザクション (transaction)」の同義語。(2) インフォメーション・カタログ・センターでは、アプリケーション・プロセスにおけるリカバリー可能な一連の操作。アプリケーション・プロセスは、常に単一の作業単位であるが、コミットまたはロールバック操作のため、アプリケーション・プロセス全体には複数の作業単位が関与する。

作業単位 (UOW) 表 (unit-of-work (UOW) table). データベース・ログまたはジャーナルから読み取ったコミット・レコードを含むキャプチャー・コントロール・サーバーに保管されたレプリケーション・コントロール表。レコードには、リカバリー単位 ID が組み込まれている。このリカバリー単位 ID は、作業単位表と CD 表を結合させて、トランザクション間で整合性のある変更データを生成するために使用される。

作業ファイル (work file). サブスクリプション・セットの処理中にアプライ・プログラムが使用する一時ファイル。

サブスクリプション (subscription). (1) サブスクリプション・セットおよびサブスクリプション・セット・メンバーを作成するプロセス。「登録 (registration)」と対比。(2) 「サブスクリプション・セット (subscription set)」を参照。

サブスクリプション・サイクル (subscription cycle). DB2 レプリケーションにおける 1 つのプロセスで、アプライ・プログラムが所定のサブ

スクリプション・セットの変更済みデータを取り出し、変更内容をターゲット表に複製し、その状況と現在の進行を反映するように、該当するレプリケーション・コントロール表を更新する。

サブスクリプション・セット (subscription set). サブスクリプション・サイクル中に変更されたデータのレプリケーションをコントロールするレプリケーション定義。サブスクリプション・セットにはサブスクリプション・セット・メンバーが含まれる場合がある。

サブスクリプション・セット・メンバー (subscription-setmember). 登録済みレプリケーション・ソースをレプリケーション・ターゲットでマップするレプリケーション定義。各メンバーは、ターゲット表の構造と、ソース表から複製される行と列を定義する。

サブセット (subset). 表全体からターゲット表へ複製するのではなく、ソース表の一部からデータを複製すること。行単位または列単位のサブセットが可能である。

差分リフレッシュ (differential refresh). 「変更キャプチャー・レプリケーション (change-capture replication)」を参照。

参照制約 (referential constraints). 外部キーの非 NULL 値が有効なのは親キーの値として表示されるときだけである、という参照保全規則。

参照保全 (referential integrity). 外部キーの値がすべて有効であるデータベースの状態。参照保全の保守には、参照制約が定義されている表のデータを変更するすべての操作に、参照制約 (referential constraints) の制約が必要である。

シグナル (signal). キャプチャー、アプライ、およびモニター・プログラムが相互に非同期で通信することができるレプリケーションの通信メカニズム。

ジャーナリング (journaling). iSeries システムでは、物理ファイル・メンバーやアクセス・パス

などのオブジェクトへの変更をジャーナルへ記録する処理、またはシステム関数やユーザー関数を使用してジャーナル・エントリーをデポジットする処理。

ジャーナル ID (journal identifier (JID)).

iSeries システムでは、あるオブジェクトに対してジャーナリングを開始すると、そのオブジェクトに割り当てられる固有の ID。この JID 値を使用して、オブジェクトとジャーナル・エントリーが関連付けられる。

ジャーナル (journal). iSeries システムでは、記録 (ジャーナル) されるオブジェクトを識別するシステム・オブジェクト、現行のジャーナル・レシーバー、およびジャーナル用システム上のすべてのジャーナル・レシーバーを指す。「ジャーナル・レシーバー (journal receiver)」も参照。

ジャーナル・エントリー (journal entry).

iSeries システムでは、ジャーナルされた変更や、ジャーナルされたその他の活動についての情報を含む、ジャーナル・レシーバー内の 1 つのレコード。「ジャーナル・コード (journal code)」および「ジャーナル・エントリー・タイプ (journal entry type)」も参照。

ジャーナル・エントリー・タイプ (journal entry type). iSeries システムでは、システム生成ジャーナル・エントリーの操作タイプ、またはユーザー生成ジャーナル・エントリーのタイプを識別する、ジャーナル・エントリー内の 2 文字フィールド。たとえば、PT は書き込み操作のエントリー・タイプである。「ジャーナル・コード (journal code)」も参照。

ジャーナル・コード (journal code). iSeries システムでは、ジャーナル・エントリーのカテゴリーを識別する、ジャーナル・エントリー内の 1 文字コード。たとえば、F は、ファイルへの操作を識別し、R は、レコードへの操作を識別する、など。「ジャーナル・エントリー (journal entry)」も参照。

ジャーナル・レシーバー (journal receiver).

iSeries システムでは、データベース・ファイルへの変更、他のジャーナル・オブジェクト、あるいはセキュリティ関連イベントへの変更などのイベントが発生した時に追加されるジャーナル・エントリが含まれるシステム・オブジェクト。

「ジャーナル (journal)」も参照。

集約表 (aggregate table). ソース表からのデータの集約が含まれる読み取り専用のレプリケーション・ターゲット表。このデータは MIN、MAX、SUM、AVG といった SQL 列関数を基にしている。

主キー (primary key). 表の定義の一部であるユニーク・キー。主キーは、参照制約定義の親キーのデフォルト。表の行を固有に識別する列または複数の列の組み合わせ。

述部 (predicate). 比較演算を明示または暗黙指定する検索条件の要素。

ステージング表 (staging table). データがターゲット・データベースへ複製される前に、データを保管するために使用される CCD 表。ステージング・データに使用される CCD 表は、1 つ以上のターゲット表へデータを更新するための中間ソースとして機能する。「整合変更データ表 (consistent-change data table)」も参照。

整合変更データ表 (consistent-change data (CCD) table). ヒストリーの保管、監査、またはステージング・データに使用されるレプリケーション・ターゲット表のタイプ。CCD 表はレプリケーション・ソースでもある。「コンプリート CCD 表 (complete CCD table)」、「コンデンス CCD 表 (condensed CCD table)」、「外部 CCD 表 (external CCD table)」、「内部 CCD 表 (internal CCD table)」、「非コンプリート CCD 表 (noncomplete CCD table)」、および「非コンデンス CCD 表 (noncondensed CCD table)」も参照。

ソース表 (source table). ターゲット表に複製されるデータを含む表。「ターゲット表 (target table)」と対比。

ソース・サーバー (source server). 登録済みレプリケーション・ソースを含むデータベース。

[タ行]

ターゲット表 (target table). 登録済みレプリケーション・ソースから複製された変更に対するターゲットである表。ターゲット表には、ユーザー・コピー表、ポイント・イン・タイム表、基礎集約表、変更集約表、CCD 表またはレプリカ表がある。

ターゲット・サーバー (target server). レプリケーション・ターゲット表が含まれるデータベース。

対等表 (peer table). 対等レプリケーション構成の一部として定義されるレプリケーション・ソースまたはターゲット表。

対等レプリケーション (peer-to-peer replication). すべての対等表が、登録済みソースおよび読み取り-書き込みターゲットの両方であり、フル・リフレッシュの基本ソース表がない場合のレプリケーション構成。この構成では、対等表の間にはレプリケーション階層はない。

「update-anywhere レプリケーション (update-anywhere replication)」と対比。

「multi-tier レプリケーション (multi-tier replication)」も参照。

タイム・スタンプ (timestamp). 年、月、日、時、分、秒、およびマイクロ秒で表される日時で構成される、7 つの部分から成る値。

逐次化 (serialization). (1) 項目の連続配列。(2) リソースの保全性を保護するリソースへのアクセスを制御する処理。

通常識別子 (ordinary identifier). (1) SQL では、名前を作成するために使用される文字。英字

(a～z および A～Z) の後に、ゼロ個以上の英字、記号、数字、または下線文字が続く。(2) DB2 Universal Database for z/OS and OS/390 では、英大文字の後にゼロ個以上の文字が続く。それぞれの文字は英大文字、数値、数字、または下線文字である。

データ統合 (data consolidation). 読み取り専用のターゲット・データベースを 1 つ含むレプリケーション構成。ターゲット表には、1 つ以上のソース表からのデータの行が含まれる。

データ分布 (data distribution). シングル・ソース表が含まれるレプリケーション構成。この構成から、変更が 1 つ以上の読み取り専用ターゲット表に複製される。ターゲット表へのレプリケーションが発生する前に、ソース表からのデータの完全セットが表に含まれている必要がある。

データベース管理システム (database management system (DBMS)). 「データベース・マネージャー (database manager)」の同義語。

データベース・サーバー (database server). ローカル・アプリケーションまたは中間データベース・サーバーからの要求のターゲット。DB2 環境では、データベース・サーバーの機能は、分散データ機能によって提供され、ローカル・アプリケーションまたは中間データベース・サーバーとして動作するリモート・データベース・サーバーから DB2 データにアクセスするために使用される。

データベース・マネージャー (database manager). 効率のよいアクセス、整合性、リカバリー、データの並行性制御、プライバシー、およびセキュリティのための中央制御、データ独立性、および複合物理構造の各種サービスを提供することによってデータを管理するコンピューター・プログラム。

データベース・ログ (database log). データベースに対するすべての変更内容を記録するログ・

レコードで構成される一連の 1 次および 2 次ログ・ファイル。データベース・ログを使って、コミットされないトランザクション (作業単位) に加えられた変更をロールバックしたり、データベースを整合性のある状態にリカバーしたりする。

データ・ブロック (data blocking). アプライ・サイクル中にデータを変更する時間に相当する特定の分数 (時間) を複製する処理。

同期点 (synchpoint). 最新のアプライ・サイクル中に適用された最新の変更の DB2 ログまたはジャーナル・レコードのシーケンス番号に対するレプリケーション・コントロール表の値。この値は、CD 表の整理を調整するためにも使用される。

同期レプリケーション (synchronous replication). リアルタイム・レプリケーションとしても認識されている。更新をソース・トランザクションの有効範囲内で連続して送達するレプリケーションのタイプ。

登録 (registration). (1) DB2 の表、ビュー、またはニックネームをレプリケーション・ソースとして登録する処理。「サブスクリプション (subscription)」と対比。(2) 「レプリケーション・ソース (replication source)」を参照。

特殊タイプ (distinct type). 内部的には既存のタイプ (そのソース・タイプ) として表示されるが、セマンティクスとしての用途においては別個で非互換のタイプと見なされるユーザー定義のデータ・タイプ。「ユーザー定義タイプ (user-defined type)(UDT)」も参照。

トランザクション (transaction). (1) ワークステーションとプログラム、2 つのワークステーション、または 2 つのプログラム間での交換で、特定のアクションまたは結果が伴うもの。たとえば、カスタマーの預金の入力、カスタマーの貸借の更新など。「作業単位 (unit of work)」の同義語。(2) Net.Data の 1 回の呼び出し。永続

Net.Data を使用する場合は、トランザクションは複数回 Net.Data を呼び出すことができる。

トランザクション整合レプリケーション

(transaction-consistent replication). すべてのトランザクションの更新が、ターゲット表に複製されるレプリケーション処理のタイプ。「トランザクション・ベース・レプリケーション (*transaction-based replication*)」と対比。

トランザクション・ベース・レプリケーション

(transaction-based replication). すべてのトランザクションがソース表でコミットされている時に、ターゲット表に複製されるレプリケーション処理のタイプ。「トランザクション整合レプリケーション (*transaction-consistent replication*)」と対比。

トランザクション・モード処理

(transaction-mode processing). レプリケーション・サブスクリプション・セット処理のタイプで、アプライ・プログラムがソース CD 表からデータを検索し、次にデータをソースで使用したシーケンスと同じコミット・シーケンスでターゲット表に適用する。アプライ・プログラムは、順次ではなく、すべてのサブスクリプション・セット・メンバーをまとめてトランザクションを処理する。「表モード処理 (*table-mode processing*)」と対比。

トリガー (trigger). (1) 特定の SQL が実行されているときに、データベース・マネージャーが間接的に呼び出すデータベースのオブジェクト。「キャプチャー・トリガー (*Capture trigger*)」も参照。(2) DB2 データベースに保管され、DB2 表にあるイベントが発生したときに実行される一連の SQL ステートメント。

トレース (trace). (1) モニター、監査、パフォーマンス、アカウントिंग、統計、および保守容易性 (グローバル) 関連のデータをモニターし、収集する能力を提供する DB2 Universal Database for z/OS and OS/390 の機能。(2) DB2 レプリケーションでは、キャプチャー・プログラ

ム、アプライ・プログラム、またはレプリケーション・アラート・モニターのモニター、監査、およびパフォーマンス・データを収集する能力を提供する機能を指す。

[ナ行]

内部 CCD 表 (internal CCD table). 登録されたレプリケーション・ソースでないために、直接サブスクリプションできない CCD 表。レジスター表にはその独自の行が存在しない。関連した登録済みレプリケーション・ソースの行に対する CCD_OWNER および CCD_TABLE 列によって識別される。「外部 CCD 表 (*external CCD table*)」と対比。「整合変更データ表 (*consistent-change data (CCD) table*)」も参照。

ニックネーム (nickname). (1) データ・ソース・オブジェクト (表やビューなど) を参照するために連合サーバーが使用する ID。(2) DB2 以外のデータベースの物理データベース・オブジェクト (表やストアード・プロシージャなど) を表すために、DB2 DataJoiner データベースで定義された名前。

[ハ行]

バイナリー・ラージ・オブジェクト (binary large object (BLOB)). バイト・シーケンス。そのシーケンスのサイズは、0 バイトから 2 ギガバイト未満の範囲である。このストリングには、関連したコード・ページおよび文字セットがない。BLOB にはイメージ、音声、およびビデオ・データが含まれる。「文字ラージ・オブジェクト (*character large object*)」および「2 バイト文字ラージ・オブジェクト (*double-byte character large object*)」も参照。

パッケージ (package). SQL ステートメントを実行するために使用されるプログラム準備中に作成される制御構造。

非コミット読み取り (UR) (uncommitted read (UR)). アプリケーションが他のトランザクションの非コミット変更にアクセスするのを可能にする分離レベル。他のアプリケーションが表を消去または変更しようとしないう限り、アプリケーションは自分が読み取っている行以外のアプリケーションをロックしない。

非コンデンス CCD 表 (noncondensed CCD table). キー値ごとに、複数の行が含まれた CCD 表。これらの複写行は、表の行にある値に対する変更履歴を表す。「コンデンス CCD 表 (condensed CCD table)」と対比。「整合変更データ表 (consistent-change data (CCD) table)」も参照。

非コンプリート CCD 表 (noncomplete CCD table). 初期状態は空で、レプリケーション・ソースに変更が行われるにつれて行が追加される CCD 表。「コンプリート CCD 表 (complete CCD table)」と対比。「整合変更データ表 (consistent-change data (CCD) table)」も参照。

非常時接続 (occasionally connected). 常にネットワークに接続している訳ではないターゲット・サーバーが含まれるレプリケーション構成。この構成により、ユーザーはローカル・データベースをソースのデータと短時間同期化するために、基本データ・ソースに接続することができる。

非同期レプリケーション (asynchronous replication). ソース表を更新した元のトランザクションの有効範囲外にあるターゲット表へ、ソース表からデータをコピーする処理。「同期レプリケーション (synchronous replication)」と対比。

ビュー (view). (1) 照会によって生成されたデータからなる論理表。基本表の基礎設定に基づいており、ビューのデータは基本表で実行されている SELECT タイプの照会によって判別される。(2) オブジェクトに関する情報とオブジェクトに

含まれる情報を参照する方法。それぞれのビューは、オブジェクトに関する異なった情報を明らかにする場合がある。

表モード処理 (table-mode processing). レプリケーション・サブスクリプション・セット処理のタイプで、アプライ・プログラムがソース CD 表からすべてのデータを検索し、次にデータをターゲット表ごとに適用 (1 回に 1 メンバー) し、最終的にこの作業をコミットする。「トランザクション・モード処理 (transaction-mode processing)」と対比。

プッシュ構成 (push configuration). アプライ・プログラムが、ターゲット・サーバー以外のソース・サーバーまたはレプリケーション・サーバーで実行するレプリケーション構成。アプライ・プログラムはソース・サーバーからの更新をターゲットにアプライするためにプッシュする。「プル構成 (pull configuration)」と対比。

プルーニング、整理 (pruning). レプリケーション・コントロール表、CD 表、CCD 表、またはキャプチャーあるいはアプライのログ・ファイルから、古いデータを除去するタスク。

プル構成 (pull configuration). アプライ・プログラムがターゲット・サーバーで実行するレプリケーション構成。アプライ・プログラムはソース・サーバーからの更新をターゲットにアプライするためにプルする。「プッシュ構成 (push configuration)」と対比。

フル・リフレッシュ (full refresh). すべてのデータのプロセスが登録に一致し、レプリケーション・ソース表に対するサブスクリプション・セット述部がターゲット表にコピーされる。フル・リフレッシュは、ターゲット表の既存のデータをすべて置換する。データ分布構成で、フル・リフレッシュは他のデータが複製される前に完了しておく必要がある。「変更キャプチャー・レプリケーション (change-capture replication)」と対比。

ブロッキング (blocking). アプリケーションをバインドするときに指定するオプション。このオプションを指定すると、通信サブシステムが複数行の情報をキャッシュできるようになるため、FETCH ステートメントで各要求を 1 行ずつネットワークに伝送する必要がなくなる。「ブロック・フェッチ (*block fetch*)」も参照。

ブロック・フェッチ (block fetch). 大量の行を一緒に検索 (またはフェッチ) する DB2 の機能。ブロック・フェッチを使用すると、ネットワークにまたがって送信されるメッセージの数をかなり減らすことができる。データを更新しないカーソルのみに適用する。

プロモート (promote). ソースを再度登録したりサブスクリプション・セットを再作成したりせずに、あるデータベースから別のデータベースへ、サブスクリプション・セットまたは登録済みソースのレプリケーション定義をコピーすること。

変更キャプチャー・レプリケーション (change-capture replication). レプリケーション・ソース表に対して行われた変更をキャプチャーし、これをレプリケーション・ターゲット表にアプライする処理。「フル・リフレッシュ (*full refresh*)」と対比。

変更後イメージ (after-image). 変更データ (CD) 表、またはデータベース・ログかジャーナルに記録されたソース表の列の更新後の内容。「変更前イメージ (*before-image*)」と対比。

変更集約表 (change aggregate table). CD 表の内容を基に集約されるデータが含まれるレプリケーション・ターゲット表のタイプ。変更が CD 表へ書き込まれた時間間隔にマークを付けるための 2 つのタイム・スタンプも含まれる。「基礎集約表 (*base aggregate table*)」と対比。

変更前イメージ (before-image). トランザクションによる更新の前のレプリケーション・ソース表の内容で、変更データ (CD) 表、またはデータ

ベース・ログかジャーナルに記録されている。「変更後イメージ (*after-image*)」と対比。

変更データ表 (change data (CD) table). レプリケーション・ソース表の変更済みデータの入った、キャプチャー・コントロール・サーバーのレプリケーション表。

ポイント・イン・タイム表 (point-in-time table). すべてまたは一部のソース表と内容が一致するレプリケーション・ターゲット表のタイプ。ソース・システムで特定の行が挿入または更新された大体の時刻を識別する追加列を含む。

保持制限整理 (retention-limit pruning). ユーザー指定制限より先のキャプチャー・プログラムによる CD または UOW 表の整理。

ホット・スポット更新 (hot-spot update). 短い期間で、同じ行に対して行われる一連の繰り返し更新。

[マ行]

マスター表 (master table). update-anywhere レプリケーションでは、レプリカ表のデータに対するオリジナルのソース表を指す。レプリケーション競合検出が使用可能な場合、マスター表への変更は保存されるが、レプリカ表への変更はリジェクトされる。「update-anywhere レプリケーション (*update-anywhere replication*)」、「レプリカ表 (*replica table*)」、および「競合検出 (*conflict detection*)」も参照。

待ち時間 (latency). レプリケーション・ターゲットで現れるレプリケーション・ソースに行われる更新に必要な時間。

メンバー (member). 「サブスクリプション・セット・メンバー (*subscription-set member*)」を参照。

文字ラージ・オブジェクト (character large object (CLOB)). 長さが 2 ギガバイト以下の文字シーケンス (1 バイトまたはマルチバイト、あ

るいはその両方)。一般には、CLOB の値は、文字ストリングが VARCHAR タイプの制限を超過した場合にはいつでも使用される。文字ラージ・オブジェクト・ストリングともいう。「バイナリー・ラージ・オブジェクト (*binary large object*)」および「2 バイト文字ラージ・オブジェクト (*double-byte character large object*)」も参照。

モニター修飾子 (Monitor qualifier). レプリケーション・モニター処理のインスタンスを識別する大文字小文字の区別のある文字ストリング。

モニター・コントロール・サーバー (Monitor control server). レプリケーション・コントロール表が含まれるデータベースで、レプリケーションがモニターするアラート条件に関する情報を保管する。

[ヤ行]

ユーザー定義タイプ (user-defined type)(UDT). データベース・マネージャーにもともとあったものではなく、ユーザーにより作成されたデータ・タイプ。DB2 Universal Database では、ユーザー定義タイプの代わりに「特殊タイプ (*distinct type*)」という用語を使用する。

ユーザー・コピー表 (user copy table). 登録済みソース表の全部または一部と一致する内容を持つレプリケーション・ターゲット表。ユーザー・データ列のみを含む。

ユニーク・キー (unique key). 値が同じものがないように制約されているキー。

ユニコード (Unicode). ISO 10646 標準のサブセットである国際文字エンコード・スキーム。各文字は固有の 2 バイト・コードで定義される。

予備ファイル (spill file). アプライ・プログラムが作成する一時ファイルで、ターゲット表を更新するためのデータを保持するために使用される。

[ラ行]

ラージ・オブジェクト (large object (LOB)). バイト・シーケンス。そのシーケンスのサイズは、0 バイトから 2 ギガバイト未満である。バイナリー・ラージ・オブジェクト (*binary large object*) (バイナリー)、文字ラージ・オブジェクト (*character large object*) (1 バイト文字または混合文字)、または 2 バイト文字ラージ・オブジェクト (*double-byte character large object*) (2 バイト文字) の 3 つのタイプのどれでもよい。

リアルタイム・レプリケーション (real-time replication). 「同期レプリケーション (*synchronous replication*)」を参照。

リキャプチャー (recapture). update-anywhere レプリケーションで、レプリカ表で変更をキャプチャーし、これらの変更をマスター表か他のレプリカ表に転送することを指す。

リジェクトされたトランザクション (rejected transaction). マスター表と競合するレプリカ表からの 1 つ以上の更新データを含むトランザクション。

リモート・データベース (remote database). 現在使用しているワークステーション以外のワークステーションに物理的に置かれているデータベース。「ローカル・データベース (*local database*)」と対比。

リレーショナル・データベース管理システム (RDBMS) (relational database management system (RDBMS)). リレーショナル・データベースへのアクセスを編成および提供するハードウェアとソフトウェアの集まり。

レプリカ表 (replica table). update-anywhere (任意の場所で更新) レプリケーションでは、ローカルに更新でき、サブスクリプション・セット定義を通して、マスター表から更新情報も受け取ることができるタイプのターゲット表。レプリケーション競合検出が使用可能な場合、レプリカ表への

変更はリジェクトされるが、マスター表への変更は保存される。「*update-anywhere* レプリケーション (*update-anywhere replication*)」、「マスター表 (*master table*)」、および「競合検出 (*conflict detection*)」も参照。

レプリケーション (replication). 複数の場所にある定義済みデータ集合を保守するための処理。これには、特定の変更内容がある場所 (ソース) から別の場所 (ターゲット) にコピーしたり、2つの場所にあるデータを同期化することが関係する。

レプリケーション管理者 (replication administrator). レプリケーション・ソースの登録およびサブスクリプション・セットの作成に責任を負うユーザー。このユーザーはキャプチャーおよびアプライ・プログラムも実行できる。

レプリケーション・アナライザー (Replication Analyzer). セットアップの問題、構成エラー、およびパフォーマンスの問題に応じたレプリケーション環境を分析できるプログラム。

レプリケーション・アラート・モニター (Replication Alert Monitor). キャプチャーおよびアプライ・プログラムのアクティビティをモニターすることが可能で、ユーザー定義アラート条件により、特定のユーザーへ通知を送信することができるプログラムの集合。

レプリケーション・コントロール表 (replication control table). レプリケーション定義または制御情報が保管されている表。

レプリケーション・センター (Replication Center). キャプチャーおよびアプライ・コントロール・サーバー、登録済みソース、サブスクリプション・セット、およびモニター・コントロール・サーバーを表示する DB2 レプリケーションのグラフィカル・ユーザー・インターフェース。レプリケーション・センターから、レプリケーション管理者はキャプチャーおよびアプライ・プログラムの操作可能タスクの実行もできる。

レプリケーション・ソース (replication source). レプリケーションのソースとして登録されている、データベース表、ビュー、またはニックネーム。このタイプの表に行った変更はサブスクリプション・セット・メンバーで定義されたターゲット表にキャプチャーされ、コピーされる。「サブスクリプション・セット (*subscription set*)」および「サブスクリプション・セット・メンバー (*subscription-set member*)」も参照。

連合データベース・システム (federated database system). 分散データベース管理システム (DBMS) の特殊タイプ。他のサーバーにあるデータの照会と操作が可能になる。データは、Oracle、Sybase、Microsoft SQL Server などのデータベース・マネージャーにあるか、あるいはスプレッドシート、Web サイト、データマートなどのリストまたは格納先にある。SQL ステートメントは、1つの文内で、複数のデータベース・マネージャーや個々のデータベース (複数) を参照することができる。たとえば、DB2 Universal Database 表、Oracle の表、および Sybase のビューを結合することができる。

ローカル・データベース (local database). 使用中のワークステーションに物理的に存在するデータベース。「リモート・データベース (*remote database*)」と対比。

ログ (log). (1) システムで行われた変更を記録するファイル。(2) DB2 Universal Database for OS/390 実行中に発生するイベントおよびその順序を記述する、レコードの集合。記録された情報は、DB2 Universal Database for OS/390 実行中に障害が起きた場合のリカバリーに使用される。(3) 「データベース・ログ (*database log*)」を参照。

ロッキング (locking). データの整合性を確保するためにデータベース・マネージャーで使われる機構。ロッキングにより、複数の並行ユーザーが非整合データにアクセスできないようにする。

ロック (lock). (1) イベントまたはデータへのアクセスの逐次化のための手段。(2) あるアプリケーション・プロセスが加えた非コミットの変更が、別のアプリケーション・プロセスで認識されないようにしたり、アプリケーション・プロセスが、別のプロセスからアクセスされているデータを変更しないようにしたりするための手段。

[数字]

2 バイト文字ラージ・オブジェクト

(double-byte character large object (DBCLOB)). 2 バイト文字シーケンス。そのシーケンスのサイズは、0 バイトから 2 ギガバイトまでの範囲である。2 バイト・テキストのラージ・オブジェクトの保管に使われるデータ・タイプ。「2 バイト文字ラージ・オブジェクト・ストリング (*double-byte character large object string*)」ともいう。このようなストリングには常に関連したコード・ページがある。

B

BLOB. 「バイナリー・ラージ・オブジェクト (*binary large object*)」を参照。

C

CCD 表 (CCD table). 「整合変更データ表 (*consistent-change data (CCD) table*)」を参照。

CD 表 (CD table). 「変更データ表 (*change data table*)」を参照。

CLOB. 文字ラージ・オブジェクト (Character large object)。

D

DB2 以外のリレーショナル・データベース・サーバー (non-DB2 relational database server).

IBM 以外のベンダーからの Informix データベース・サーバーまたはリレーショナル・データベース・サーバー。

DBCLOB. 2 バイト文字ラージ・オブジェクト (Double-byte character large object)。

DBMS. データベース管理システム (Database management system)。

L

LOB. 「ラージ・オブジェクト (*large object*)」を参照。

M

multi-tier レプリケーション (multi-tier replication). 変更があるデータベースのレプリケーション・ソースから、別のデータベースのレプリケーション・ターゲットに複製され、別のデータベースのレプリケーション・ターゲットへ再度複製されるレプリケーション構成。

N

NULL 値 (null value). 値が指定されていないパラメーターの位置。

NULL 可能 (nullable). 列、関数パラメーター、または結果に値がなくてもよい状態。

O

ODBC. 「*Open Database Connectivity (ODBC)*」を参照。

ODBC ドライバー (ODBC driver). ODBC 関数呼び出しを実行し、データ・ソースと対話するドライバー。

Open Database Connectivity (ODBC). 呼び出し可能 SQL (SQL プリプロセッサを必要としない) を使用して、データベース管理システムに

アクセスできるようにする API。ODBC アーキテクチャーを使用すると、データベース・ドライバ (*database driver*) と呼ばれるモジュールを追加することができる。これらのモジュールはアプリケーションを、選択したデータベース管理システムに実行時にリンクする。アプリケーション・プログラムがサポートされるすべてのデータベース管理システムのモジュールに直接的にリンクされる必要はない。

R

RDBMS. 「リレーショナル・データベース管理システム (*relational database management system*)」を参照。

U

UDT. 「ユーザー定義タイプ (*user-defined type*)」を参照。

update-anywhere レプリケーション

(update-anywhere replication). すべての表が、登録済みソースおよび読み取り-書き込みターゲットの両方であるレプリケーション構成。1つの表は他の表のすべてをフル・リフレッシュするための 1 次ソース表である。この構成では、ソースおよびターゲット表の間には、暗黙的なレプリケーション階層はない。「対等レプリケーション (*peer-to-peer replication*)」と対比。

「*multi-tier* レプリケーション (*multi-tier replication*)」、 「マスター表 (*master table*)」、 および 「レプリカ表 (*replica table*)」も参照。

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

圧縮辞書 (z/OS) 246

アナライザー

OS/400 版

作成, SQL パッケージの 36
呼び出しパラメーター 401

UNIX 版, 呼び出しパラメーター
326

Windows 版, 呼び出しパラメーター
326

アナライザー・レポート

ANZDPR コマンド 400

asnanalyze コマンド 325

アプライ修飾子

アプライ・プログラムの始動時に
使用 153, 164

数, 関連したサブスクリプション・
セットの 71

サブスクリプション・セット内の
変更 226

状況のモニター 183

命名規則 321

アプライ修飾子相互参照

(AUGHTKN) 表 493

アプライ・エンキュー (APPENQ) 表
517

アプライ・コントロール表

のリスト 517

APPENQ (アプライ・エンキュー)
1) 517

APPLYTRACE (アプライ・トレー
ス) 518

APPLYTRAIL (アプライ・トレー
ル) 519

アプライ・コントロール表 (続き)

APPLY_JOB (アプライ・ジョ
ブ) 518

SUBS_COLS (サブスクリプショ
ン列) 523

SUBS_EVENT (サブスクリプショ
ン・イベント) 524

SUBS_MEMBR (サブスクリプシ
ョン・メンバー) 525

SUBS_SET (サブスクリプショ
ン・セット) 528

SUBS_STMTS (サブスクリプショ
ン・ステートメント) 532

アプライ・コントロール・サーバ
ー上のコントロール表 517

レプリケーション・センターへの
追加 271

アプライ・ジョブ (APPLY_JOB) 表
518

アプライ・トレース (APPLYTRACE)
表

構造 518

プルーニング 253

アプライ・トレール (APPLYTRAIL)
表

構造 519

プルーニング 253

アプライ・プログラム

お互いの通信

キャプチャー・トリガー 473,
476

キャプチャー・プログラム
473, 475

レプリケーション・アラ
ー

ト・モニター 478

レプリケーション・センター
473

許可要件 24

コネクティビティ 17

コマンド 323

実行時処理ステートメント 124

アプライ・プログラム (続き)

スケジューリング 80, 471

スループット分析 183

操作 284

データ・ブロッキング 75

トランザクション・モード処理
78

パフォーマンス・データ 179

表モード処理 78

待ち時間分析 183

ミニサイクル 75

メッセージ 182

ユーザー ID 24

予備ファイル, 記憶要件 10

OS/400 版

作成, SQL パッケージの 35
始動 164, 438

状況のチェック 179

セットアップ 35, 37

停止 165, 409

ALWINACT パラメーター
442

APYQUAL パラメーター 440

COPYONCE パラメーター
443

CTLSVR パラメーター 440

DELAY パラメーター 442

FULLREFPGM パラメーター
441

INACTMSG パラメーター
442

JOBID パラメーター 439

OPTSNGSET パラメーター
444

RTYWAIT パラメーター 443

SUBNFYPGM パラメーター
442

TRACE パラメーター 441

TRLREUSE パラメーター
443

USER パラメーター 439

アプライ・プログラム (続き)

OS/400 用
スケジューリング 472

UNIX 版
開始 328
構成 32
始動 153, 663
状況のチェック 177
セットアップ 30
操作 324
停止 165, 324
バインド 33
パスワード・ファイル 27
apply_path パラメーター 155, 330
apply_qual パラメーター 156, 324, 329
control_server パラメーター 156, 324, 329
copyonce パラメーター 156, 331
delay パラメーター 157, 332
errwait パラメーター 158, 332
inamsg パラメーター 158, 331
loadxit パラメーター 158, 330
logreuse パラメーター 159, 330
logstdout パラメーター 159, 330
notify パラメーター 159, 331
opt4one パラメーター 160, 332
pwdfile パラメーター 160, 330
sleep パラメーター 160, 332
spillfile パラメーター 161, 333
sqlerrorcontinue パラメーター 162, 333
status 324
term パラメーター 163, 333
trlreuse パラメーター 163, 332

アプライ・プログラム (続き)

Windows 版
開始 328
構成 32
始動 153, 663
状況のチェック 177
セットアップ 30
操作 153, 324
停止 165, 324
バインド 33
パスワード・ファイル 27
apply_path パラメーター 155, 330
apply_qual パラメーター 156, 324, 329
control_server パラメーター 156, 324, 329
copyonce パラメーター 156, 331
delay パラメーター 157, 332
errwait パラメーター 158, 332
inamsg パラメーター 158, 331
loadxit パラメーター 158, 330
logreuse パラメーター 159, 330
logstdout パラメーター 159, 330
notify パラメーター 159, 331
opt4one パラメーター 160, 332
pwdfile パラメーター 160, 330
sleep パラメーター 160, 332
spillfile パラメーター 161, 333
sqlerrorcontinue パラメーター 162, 333
status 324
term パラメーター 163, 333
trlreuse パラメーター 163, 332

z/OS 版
開始 328

アプライ・プログラム (続き)

z/OS 版 (続き)
始動 153
状況のチェック 177
セットアップ 36
操作 324
停止 165, 324
apply_path パラメーター 155, 330
apply_qual パラメーター 156, 324, 329
control_server パラメーター 156, 324, 329
copyonce パラメーター 156, 331
db2_subsystem パラメーター 157, 329
delay パラメーター 157, 332
errwait パラメーター 158, 332
inamsg パラメーター 158, 331
loadxit パラメーター 158, 330
logreuse パラメーター 159, 330
logstdout パラメーター 159, 330
notify パラメーター 159, 331
opt4one パラメーター 160, 332
pwdfile パラメーター 160, 330
sleep パラメーター 160, 332
spillfile パラメーター 161, 333
status 324
term パラメーター 163, 333
trlreuse パラメーター 163, 332

z/OS 用
開始 461

アプリケーション
始動、レプリケーション・プログラム 663

アラート条件
レプリケーション・アラート・モニター用 187

アラート・モニター
参照：レプリケーション・アラート・モニター

異機種レプリケーション
制約事項
集約表 88
CCD 表 51
multi-tier レプリケーション 92
update-anywhere 55, 95
ソースの登録 46

イベント、調整 232
イベントに基づくスケジューリング 81

インターバル・タイミング 81
ウォーム・スタート、キャプチャー・プログラム
OS/400 版 448, 453
UNIX 版 143, 338
Windows 版 143, 338
z/OS 版 338
z/OS 用 143

「エラー発生時にキャプチャーを停止」オプション 54

延期
キャプチャー・プログラム
UNIX 版 149, 340
Windows 版 149, 340
z/OS 版 149, 340

オブジェクト
再活動化 205
属性の変更 200
登録 199
非活動化 204
変更のキャプチャーの停止 204

[力行]

開始
アプライ・プログラム
UNIX 版 328
Windows 版 328
z/OS 版 328

開始 (続き)
アプライ・プログラム (続き)
z/OS 用 461

キャプチャー・プログラム
Windows サービスの使用 467
z/OS 用 461

レプリケーション・アラート・モニター
UNIX 版 189, 348
Windows 版 189, 348
z/OS 版 189, 348
z/OS 用 461

外部 CCD 表
multi-tier レプリケーション 92

カスタマイズ、SQL スクリプトの 127

カタログ表、登録 43

各国語サポート (NLS) 15

活動化、サブスクリプション・セットの 75, 279

環境変数
キャプチャー・プログラム 30
DB2CODEPAGE 15, 31
DB2DBDFT 31
DB2INSTANCE 30
LIBPATH 31

監査
ギャップ、データの 89
コールド・スタート 89
ソース・データ 53

管理
許可要件 20

基礎集約表
構造 542
使用法 88
定義 85

既存の表、ターゲットとして 97

ギャップの検出 89

キャプチャー・エンキュー (CAPENQ) 表 494

キャプチャー・コントロール表
のリスト 493
AUTHTKN (アプライ修飾子相互参照) 493
CAPENQ (キャプチャー・エンキュー) 494

キャプチャー・コントロール表 (続き)
CAPMON (キャプチャー・モニター) 495
CAPPARMS (キャプチャー・パラメーター)
構造 496
使用 131
変更 147

CAPSCHEMAS (キャプチャー・スキーマ) 493

CAPTRACE (キャプチャー・トレース) 498

CCD (整合変更データ) 499

CD (変更データ) 500

PRUNCNTL (プルーニング・コントロール) 501

PRUNE_LOCK (プルーニング・ロック) 503

PRUNE_SET (プルーニング・セット) 503

REGISTER (登録) 505

REG_EXT (登録拡張) 504

REG_SYNCH (登録同期) 510

RESTART (再始動) 511

SEQTABLE (順序付け) 512

SIGNAL (シグナル) 513

UOW (作業単位) 515

キャプチャー・コントロール・サーバー
上のコントロール表 493
複数のキャプチャー・スキーマ 29
レプリケーション・センターへの追加 271

キャプチャー・シグナル 232

キャプチャー・スキーマ
使用、複数の 29
変更 207
命名規則 321

キャプチャー・スキーマ (CAPSCHEMAS) 表 493

キャプチャー・トリガー
お互いの通信
アプライ・プログラム 473, 476

- キャプチャー・トリガー (続き)
 - お互いの通信 (続き)
 - レプリケーション・センター 473
 - 競合、既存のトリガーとの 13
 - 許可要件 23
 - 計画 12
 - 名前 13
- キャプチャー・トレース (CAPTRACE) 表
 - 構造 498
 - ブルーニング 253
- キャプチャー・パラメーター (CAPPARMS) 表
 - 構造 496
 - 使用 131
 - 変更 147
- キャプチャー・パラメーターの変更
 - OS/400 版 404
 - UNIX 版 340
 - Windows 版 340
 - z/OS 版 340
- キャプチャー・プログラム
 - お互いの通信
 - アプライ・プログラム 473, 475
 - レプリケーション・アラート・モニター 478
 - レプリケーション・センター 473
 - 許可要件 22
 - コールド・スタートの防止 254
 - コネクティビティ 17
 - コマンド 323
 - シグナル 232
 - 実行、複数の 29
 - スキーマの変更 207
 - スケジューリング 471
 - スループット分析 181
 - 設定、環境変数 30
 - 設定、デフォルトの、パラメーターの場合 131
 - 操作 283
 - パフォーマンス・データ 179
 - 変更、パラメーター値の 131
- キャプチャー・プログラム (続き)
 - 変更する、動作を、実行中における 146
 - 待ち時間分析 182
 - メッセージ 181
 - メモリー、使用する 3
 - ユーザー ID 22
 - を始動する場所 138
 - OS/400 版
 - ウォーム・スタート・パラメーター 448
 - 許可要件 22
 - コールド・スタート、自動 453
 - コールド・スタート・パラメーター 448
 - 再初期化 424
 - 作成、SQL パッケージの 35, 36
 - 始動 145, 446
 - ジャーナルおよびジャーナル・レシーバーの管理 39
 - 状況のチェック 179
 - 進行 196
 - セットアップ 35, 37
 - 操作 129
 - 属性のオーバーライド 426
 - 属性の変更 404
 - 停止 148, 413
 - デフォルト・パラメーター 130, 131
 - CAPCTLLIB パラメーター 449
 - CLNUPITV パラメーター 449
 - FRCFRQ パラメーター 452
 - JOBID パラメーター 448
 - JRN パラメーター 450
 - LAG パラメーター 451
 - MEMLMT パラメーター 451
 - MONITV パラメーター 450
 - MONLMT パラメーター 450
 - RESTART パラメーター 448
 - RETAIN パラメーター 451
 - TRCLMT パラメーター 450
 - WAIT パラメーター 448
- キャプチャー・プログラム (続き)
 - OS/400 用
 - ジャーナル・エントリー・タイプ 661
 - スケジューリング 472
 - UNIX 版
 - ウォーム・スタート・パラメーター 143, 338
 - 延期 149, 340
 - コールド・スタート・パラメーター 143, 338
 - 構成 31
 - 再開 150, 340
 - 再初期化 150, 340
 - 始動 133, 334, 663
 - 状況 340
 - 状況のチェック 177
 - セットアップ 30
 - 操作 129, 340
 - 停止 148, 340
 - デフォルト・パラメーター 129
 - バインド 32
 - パラメーターの変更 340
 - ブルーニング 340
 - autoprune パラメーター 135, 336, 343
 - autostop パラメーター 136, 336, 343
 - capture_path パラメーター 136, 335
 - capture_schema パラメーター 137, 335, 341
 - capture_server パラメーター 138, 335, 341
 - commit_interval パラメーター 138, 336, 343
 - lag_limit パラメーター 138, 336, 343
 - logreuse パラメーター 139, 336, 344
 - logstdout パラメーター 139, 337, 344
 - memory_limit パラメーター 140, 337, 344

キャプチャー・プログラム (続き)

UNIX 版 (続き)

monitor_interval パラメーター
140, 337, 344
monitor_limit パラメーター
140, 337, 344
prune_interval パラメーター
141, 337, 344
retention_limit パラメーター
142, 337, 345
sleep_interval パラメーター
142, 337, 345
startmode パラメーター 143,
338
term パラメーター 144, 339,
345
trace_limit パラメーター 144,
339, 345

Windows 版
ウォーム・スタート・パラメ
ーター 143, 338
延期 149, 340
コールド・スタート・パラメ
ーター 143, 338
構成 31
再開 150, 340
再初期化 150, 340
始動 133, 334, 663
状況 340
状況のチェック 177
セットアップ 30
操作 129, 340
停止 148, 340
デフォルト・パラメーター
129
バインド 32
パラメーターの変更 340
ブルーニング 340
autoprune パラメーター 135,
336, 343
autostop パラメーター 136,
336, 343
capture_path パラメーター
136, 335
capture_schema パラメーター
137, 335, 341

キャプチャー・プログラム (続き)

Windows 版 (続き)

capture_server パラメーター
138, 335, 341
commit_interval パラメーター
138, 336, 343
lag_limit パラメーター 138,
336, 343
logreuse パラメーター 139,
336, 344
logstdout パラメーター 139,
337, 344
memory_limit パラメーター
140, 337, 344
monitor_interval パラメーター
140, 337, 344
monitor_limit パラメーター
140, 337, 344
prune_interval パラメーター
141, 337, 344
retention_limit パラメーター
142, 337, 345
sleep_interval パラメーター
142, 337, 345
startmode パラメーター 143,
338
term パラメーター 144, 339,
345
trace_limit パラメーター 144,
339, 345

z/OS 版

ウォーム・スタート・パラメ
ーター 143, 338
延期 149, 340
開始 334
コールド・スタート・パラメ
ーター 143, 338
再開 150, 340
再初期化 150, 340
始動 133
状況 340
状況のチェック 177
セットアップ 36
操作 129, 340
停止 148, 340

キャプチャー・プログラム (続き)

z/OS 版 (続き)

デフォルト・パラメーター
129
パラメーターの変更 340
ブルーニング 340
autoprune パラメーター 135,
336, 343
autostop パラメーター 136,
336, 343
capture_path パラメーター
136, 335
capture_schema パラメーター
137, 335, 341
capture_server パラメーター
138, 335, 341
commit_interval パラメーター
138, 336, 343
lag_limit パラメーター 138,
336, 343
logreuse パラメーター 139,
336, 344
logstdout パラメーター 139
memory_limit パラメーター
140, 337, 344
monitor_interval パラメーター
140, 337, 344
monitor_limit パラメーター
140, 337, 344
prune_interval パラメーター
141, 337, 344
retention_limit パラメーター
142, 337, 345
sleep_interval パラメーター
142, 337, 345
startmode パラメーター 143,
338
term パラメーター 144, 339,
345
trace_limit パラメーター 144,
339, 345

z/OS 用
開始 461

キャプチャー・モニター (CAPMON)
表
構造 495

- キャプチャー・モニター (CAPMON)
 - 表 (続き)
 - ブルーニング 253
- キャプチャー・ログ・ファイル 139
- 行
 - サブセット化
 - ソースでの 49
 - ターゲットでの 98
 - 使用可能、レプリケーションに 49
 - 定義、ターゲット表での 98
 - 登録、ソース表内の 49
- 行 (水平方向) のサブセット化
 - ソースでの 49
 - ターゲットでの 98
- 行キャプチャー規則 49
- 競合
 - 回避 12
- 競合検出
 - 概要 60
 - 計画 12
 - 対等レプリケーション 12
 - 要件 53
 - レベル 61
 - update-anywhere レプリケーション 12
- 許可
 - アプライ・プログラム 24
 - 管理のための 20, 22
 - キャプチャー・トリガーの場合 23
 - キャプチャー・プログラムの場合 22
 - レプリケーション・アラート・モニター用 26
- 区切り文字、生成済み SQL スクリプト内の 128
- グローバル・レコード 506
- クロスローダー・ユーティリティ 173
- 計画
 - 競合検出 12, 60
 - 共存、トリガーの 13
 - ストレージ要件 5
 - トランザクション・スループット率 13
- 計画 (続き)
 - メモリー 3
 - ログの影響 6, 13
 - ロック、CCD 表に対する 13
 - 結合、ソースとしての 64
 - 現行レシーバーのサイズ 7, 39
 - コード・ページ
 - 互換性のある 14
 - トランザクション 14
 - DB2CODEPAGE 環境変数 15
 - コールド・スタート、キャプチャー・プログラム
 - 回避 254
 - OS/400 版 448, 453
 - UNIX 版 143, 338
 - Windows 版 143, 338
 - z/OS 版 338
 - z/OS 用 143
- 更新
 - 競合 60
 - 削除および挿入として 54
- 更新済み主キー列 54
- 構成
 - アプライ・プログラム
 - UNIX 版 32
 - Windows 版 32
 - キャプチャー・プログラム
 - UNIX 版 31
 - Windows 版 31
 - コネクティビティー 17
 - レプリケーション・アラート・モニター
 - UNIX 版 33
 - Windows 版 33
- 構成パラメーター、DB2 用
 - APPLHEAPSZ 31
 - DBHEAP 31
 - LOGBUFSZ 31
 - LOGFILSIZ 31
 - LOGPRIMARY 31
 - LOGSECOND 31
 - MAXAPPLS 31
- コネクティビティー
 - コントロール表の障害のリカバリー 255
 - DB2 プラットフォーム間 17, 18
- コマンド
 - 参照：レプリケーション・コマンド
- 固有のデータ・タイプ 106
- コントロール表
 - アプライ
 - 作成 270
 - アプライ・コントロール・サーバー 489
 - アプライ・コントロール・サーバー上の 517
 - キャプチャー
 - 作成 268
 - キャプチャー・コントロール・サーバー上の 493
 - キャプチャー・サーバー 487
 - 許可要件、OS/400 の場合 37
 - クイック・リファレンス
 - アプライ・コントロール・サーバー 489
 - 概観 482
 - キャプチャー・サーバー 487
 - ターゲット・サーバー 492
 - 再バインド、パッケージおよびプラン 248
 - 再編成 248
 - 作成
 - アプライ用 270
 - キャプチャー用 268
 - 非 DB2 リレーショナル・ソースの場合 28
 - 複数のセット 29
 - レプリケーション・アラート・モニター用 185, 270
 - OS/400 での 28, 408
 - UNIX および Windows での 27
 - z/OS での 27
 - ストレージ要件 8
 - 静的 249
 - 接続障害のリカバリー 255
 - ターゲット・サーバー 492
 - 動的 247
 - ドロップ
 - レプリケーション・アラート・モニター用 186

コントロール表 (続き)

入出力エラーのリカバー 255
付与、権限の、OS/400 の場合
22
ブルーニング 250
プロファイル 265
保守 247
モニター
作成 270
モニター・コントロール・サーバ
ー上の 534
ALERTS (モニター・アラ
ート) 534
APPENQ (アプライ・エンキュー)
517
APPLYTRACE (アプライ・トレー
ス) 518
APPLYTRAIL (アプライ・トレー
ル) 519
APPLY_JOB (アプライ・ジョ
ブ) 518
AUTHTKN (アプライ修飾子相互
参照) 493
CAPENQ (キャプチャー・エンク
ュー) 494
CAPMON (キャプチャー・モニタ
ー)
構造 495
ブルーニング 253
CAPPARMS (キャプチャー・パラ
メーター)
構造 496
CAPSCHEMAS (キャプチャー・
スキーマ) 493
CAPTRACE (キャプチャー・トレ
ース)
構造 498
ブルーニング 253
CCD (整合変更データ)
キャプチャー・コントロー
ル・サーバー 499
ターゲット・サーバー 543
CD (変更データ) 500
CONDITIONS (モニター条
件) 535

コントロール表 (続き)

CONTACTGRP (モニター・グル
ープ連絡先) 538
CONTACTS (モニター連絡
先) 538
GROUPS (モニター・グル
ープ) 539
MONENQ (モニター・エンク
ュー) 539
MONSERVERS (モニター・サー
バー) 539
MONTRACE (モニター・トレー
ス) 540
MONTRAIL (モニター・トレ
ール) 541
OS/400 の権限の取り消し 436
OS/400 の権限の付与 416
PRUNCNTL (ブルーニング・コン
トロール) 501
PRUNE_LOCK (ブルーニング・ロ
ック) 503
PRUNE_SET (ブルーニング・セ
ット) 503
REGISTER (登録) 505
REG_EXT (登録拡張) 504
REG_SYNCH (登録同期) 510
RESTART (再始動) 511
RUNSTATS ユーティリティ
248
SEQTABLE (順序付け) 512
SIGNAL (シグナル) 513
SUBS_COLS (サブスクリプシ
ョン列) 523
SUBS_EVENT (サブスクリプシ
ョン・イベント) 524
SUBS_MEMBR (サブスクリプシ
ョン・メンバー) 525
SUBS_SET (サブスクリプシ
ョン・セット) 528
SUBS_STMTS (サブスクリプシ
ョン・ステートメント) 532
UOW (作業単位) 515
コントロール・サーバー、レプリケ
ーション・センターへの追加 271

[サ行]

サーバー
レプリケーション・センターへの
追加 271
サービス
Windows SCM 467
再開
キャプチャー・プログラム
UNIX 版 150, 340
Windows 版 150, 340
z/OS 版 150, 340
再活性化
オブジェクト 205
登録 205
表 205
再始動 (RESTART) 表 511
再初期化、キャプチャー・プログラ
ムの
UNIX 版 150
Windows 版 150
z/OS 版 150
再バインド、パッケージおよびプラ
ン 248
再編成
コントロール表 248
作業単位 (UOW) 表
構造 515
ストレージ要件 9
ブルーニング 252, 515
列、CCD 表の 90
索引
ターゲット表 100
作成、コントロール表の 27
サブスクリプト、ソースの 72
サブスクリプション列 (SUBS_COLS)
表 523
サブスクリプション・イベント
(SUBS_EVENT) 表
構造 524
通知、イベントの 81
サブスクリプション・サイクル 75
サブスクリプション・ステートメン
ト (SUBS_STMTS) 表 532
サブスクリプション・セット
数、アプライ修飾子の 71

- サブスクリプション・セット (続き)
 - 活動化 279
 - 活動化レベル 75
 - 行 98
 - 削除 231, 432
 - 作成 72, 211, 275
 - 参照保全 97
 - 実行時処理ステートメント 124
 - 処理モード 78
 - スケジューリング
 - イベントに基づく 81
 - 時間に基づく 81
 - ストアド・プロシージャ 80
 - 追加 374
 - データ整合性 97
 - 非活動化 229, 279
 - プロモート 280
 - 分割 218
 - 変更
 - アプライ修飾子 226
 - 属性 215
 - 名前 215
 - マージ 223
 - ミニサイクル 75
 - メンバーの追加 82, 212
 - 列 98
 - multi-tier レプリケーション 92
 - SQL ステートメント 80
 - update-anywhere レプリケーション 94
 - サブスクリプション・セット
 - (SUBS_SET) 表 528
 - サブスクリプション・セットの作成 275
 - サブスクリプション・セット・メンバー
 - 数、サブスクリプション・セットごとの 70
 - 削除 434
 - 選択、ターゲット・タイプの 85
 - 追加 82, 212, 390
 - 定義、ターゲット・キーの 100
 - 適用、行のサブセット 98
 - 適用、列のサブセット 98
 - マッピング、データ・タイプの 100
 - サブスクリプション・セット・メンバー (続き)
 - マッピング、列間の 100
 - multi-tier レプリケーション 92
 - update-anywhere レプリケーション 94
 - サブスクリプション・メンバー (SUBS_MEMBER) 表 173, 525
 - サブセット化
 - 行、ターゲットでの変更の 98
 - 高度な技法
 - 使用、述部の 119
 - 使用、ビューの 118
 - 使用する、トリガーを、CD 表で 118
 - 登録時の 117
 - 登録済みの行、変更を含む 49
 - 登録済み列の 48
 - 列、ターゲットでの 98
 - 差分リフレッシュ・レプリケーション
 - 参照：変更キャプチャー・レプリケーション
 - 算出列 98
 - 作成 125
 - ソース表 88
 - CD 表 89
 - 参照保全 97
 - 時間に基づくスケジューリング 81
 - シグナル
 - 分散リカバー点の設定 235
 - CAPSTART 236
 - CAPSTOP 238
 - STOP 233, 235
 - USER 232
 - シグナル (SIGNAL) 表
 - 構造 513
 - ブルーニング 253
 - システム開始タスク 463
 - システム変更ジャーナル管理 39
 - 実行、SQL スクリプトの 127
 - 実行管理機能オブジェクト 39
 - 実行時処理 80, 124
 - 始動
 - アプライ・プログラム OS/400 版 164, 438
- 始動 (続き)
 - アプライ・プログラム (続き)
 - UNIX 版 153, 663
 - Windows 版 153, 663
 - z/OS 版 153
 - キャプチャー・プログラム
 - OS/400 版 145, 446
 - UNIX 版 133, 334, 663
 - Windows 版 133, 334, 663
 - z/OS 版 133, 334
 - レプリケーション・アラート・モニター
 - UNIX 版 663
 - Windows 版 663
 - 自動ブルーニング 250
 - 自動リスタート・マネージャ (ARM) 463
 - シナリオ
 - アプライ・コントロール表の作成 295
 - アプライ・パスワード・ファイルの作成 302
 - アプライ・プログラムのアラート条件の選択 313
 - アプライ・プログラムの状況 307
 - キャプチャー・コントロール表の作成 290
 - キャプチャー・プログラムとアプライ・プログラムの停止 309
 - キャプチャー・プログラムのアラート条件の選択 312
 - キャプチャー・プログラムの状況 306
 - 計画 288
 - サブスクリプション・セットの作成 295
 - セットアップ 290
 - 前提条件 287
 - ソースの登録 292
 - ソース表の更新 305
 - ソース・データベースのレプリケーションを使用可能にする 291
 - 操作 305
 - データを複製する 303

| | | |
|-----------------------|-----------------------|-----------------------|
| シナリオ (続き) | 終了文字、生成済み SQL スクリプト内の | ストレージ (続き) |
| モニター・コントロール表の作成 | 128 | データベース・ログおよびジャーナル・データ |
| 310 | 主キー | 6 |
| レプリケーションのモニター | 使用される、ターゲットとして | 要件 |
| 310 | 101 | CD 表 |
| レプリケーション・アラート・モニターの開始 | 相対レコード番号、OS/400 の場合 | UOW 表 |
| 315 | 63 | スループット |
| 連絡先の作成 | 論理パーティション化 | アプライ・プログラム |
| 312 | 54 | キャプチャー・プログラム |
| ジャーナル | 述部 | 183 |
| エントリー・タイプ | サブセット化 | 181 |
| 661 | 119 | スループット率 |
| 開始 | 定義、ターゲット表の | キャプチャー・トリガー |
| 38 | 98 | 13 |
| 管理 | 順序付け (SEQTABLE) 表 | 整合変更データ (CCD) 表 |
| 39 | 512 | 外部 |
| 作成 | 診断ファイル | multi-tier レプリケーション |
| 38 | ストレージ | 92 |
| 作成、ソース表の | 10, 11 | 構造 |
| 37 | 垂直方向 (列) のサブセット化 | キャプチャー・コントロール・サーバー |
| 使用 | ソースでの | 499 |
| 37 | 48 | ターゲット・サーバー |
| 使用、リモート・ジャーナル関数 | ターゲットでの | 543 |
| 62 | 水平方向 (行) のサブセット化 | 使用法 |
| セットアップ | ソースでの | ヒストリーまたは監査 |
| 37 | 49 | 89 |
| デフォルトのメッセージ・キュー | ターゲットでの | multi-tier レプリケーション |
| 40 | 98 | 92 |
| 登録、ソースとして | スキーマ | 追加、UOW 列の |
| 43 | 変更 | 90 |
| QSQRN ジャーナル | 命名規則 | 内部 |
| 37 | 321 | 複数のターゲット |
| ジャーナル・ジョブ | スケジューリング | 90 |
| 状況のチェック | サブスクリプション・セット | 非 DB2 リレーショナル・データ・ソース |
| 179 | 80, | 使用、CCD 表の |
| ジャーナル・メッセージ・キュー | 81 | 46 |
| 40 | レプリケーション・プログラム | 非リレーショナル・データ・ソース |
| ジャーナル・レシーバー | 471 | 使用、CCD 表の |
| 管理 | ステージング、データの | 43 |
| 39 | 92 | CCD 表の保守 |
| 限界値 | ステージングされたレプリケーション | 67 |
| 39 | 93 | レプリケーション・ソース |
| 現行、サイズ | ストアド・プロシージャ | 92 |
| 7 | 操作、データの | ロック |
| 作成、ソース表の | 124 | 13 |
| 37 | 定義、サブスクリプション・セットの | 生成済み SQL スクリプト |
| システム管理 | 80 | 127 |
| 39 | ストレージ | 静的なコントロール表 |
| ジャーナル・レシーバー削除出口ルーチン | アプライ診断ファイル | 249 |
| 40 | 11 | 制約事項 |
| 保守 | 一時ファイル | 異機種のレプリケーション |
| 242 | 10 | 51, |
| 保存 | キャプチャー診断ファイル | 92, 95 |
| 245 | 10 | 既存のターゲット表 |
| ユーザー管理 | キャプチャー予備ファイル | 97 |
| 40 | 10 | 固有のデータ・タイプ |
| ジャーナル・レシーバー削除出口ルーチン | コントロール表 | 106 |
| 削除 | 8 | ストアド・プロシージャ |
| 41 | 診断ファイル | 124 |
| 説明 | 10 | 地理情報データ・タイプ |
| 40 | ターゲット表 | 105 |
| 登録 | 8 | データ暗号化 |
| 41 | | 105 |
| 集約表 | | |
| 基礎集約 | | |
| 88, 542 | | |
| 変更集約 | | |
| 89, 542 | | |

制約事項 (続き)

- データ・タイプ 105
 - 非 DB2 リレーショナル・データ・ソース 55, 60
 - ビュー 66
 - ユーザー定義のデータ・タイプ 106
 - ユニコード 表 659
 - 要約データ・タイプ 105
 - 列名の制限 52
 - ASCII 表 659
 - CCD 表 95
 - DATALINK 値 60, 95
 - DB2 Enterprise Server Edition 49
 - DB2 エクステンダー・ラージ・オブジェクト 107
 - EDITPROC 文節 105
 - FIELDPROC 文節 105
 - LOB データ・タイプ 95
 - LONG VARGRAPHIC データ・タイプ 105
 - Microsoft SQL Server 51
 - Oracle ソース 51
 - Sybase 51
 - VALIDPROC 文節 105
 - WHERE 文節 99
- ## 接続
- iSeries サーバーへの 18
 - z/OS サーバーへの 18
- ## 設定、環境変数
- キャプチャー・プログラム 30
- ## セットアップ
- アプライ・プログラム
 - OS/400 版 35
 - UNIX 版 30
 - Windows 版 30
 - キャプチャー・プログラム
 - OS/400 版 35
 - UNIX 版 30
 - Windows 版 30
 - ジャーナル 37
 - レプリケーション・アラート・モニター 33
- ## 接頭部、変更前イメージ
- ソース
 - サブスクライブ 72

ソース (続き)

- 登録
 - 非 DB2 リレーショナル 46
 - ビュー 64, 66
 - レプリケーション・センター 273
 - DB2 表 43
 - IMS データ・ソース 43
 - 登録、行の 49
 - 登録、列の 48
 - 登録オプション
 - エラー発生時にキャプチャーを停止 54
 - 行 (水平方向) のサブセット化 49
 - 競合検出 60
 - 更新、削除および挿入として 54
 - 使用、リモート・ジャーナル 62
 - 相対レコード番号 63
 - フル・リフレッシュ・コピー 48
 - 変更キャプチャー・レプリケーション 48
 - 変更後イメージ列 50
 - 変更前イメージ列 50
 - リキャプチャー、変更の (update-anywhere) 55
 - 列 (垂直方向) のサブセット化 48
 - プロファイル 266
 - プロモート 280
 - マッピング、ターゲットへの 82
 - CCD (整合変更データ) 表 92
 - CCD 表の保守 67
- ## ソースの登録
- 273
- ## ソース表
- 作成、ジャーナルの 37
 - 脱したデータのリトリート 256
 - 保守 241
 - 列の追加 201

ソース・サーバー

- 非 DB2 リレーショナル
 - ログの影響 13
 - DB2
 - ログの影響 6
 - ソース・システム、保守 241
 - ソース・ログ、保守 242
 - 相関 ID 64
 - 操作
 - アプライ・プログラム 284, 324
 - キャプチャー・プログラム 283, 340
 - レプリケーション・アラート・モニター 285, 346
 - 操作、データの
 - 作成、算出列の 125
 - サブスクリプションでの 124
 - 登録での 123
 - 名前変更、列の 125
 - 相対タイミング 81
 - 相対レコード番号
 - サポート、OS/400 に対する 63
 - 主キーとしての、OS/400 の場合 63
 - 使用される、ターゲットとして 101
 - 属性
 - サブスクリプション・セットの変更 215
 - 登録済みオブジェクトの変更 200
 - 属性のオーバーライド (OS/400)
 - キャプチャー・プログラム 426
- ## [夕行]
- ターゲット
 - フル・リフレッシュの強制 281
 - プロファイル 267
 - ターゲット索引 100
 - ターゲット表
 - 新しい列 125
 - 基礎集約
 - 構造 542
 - 使用法 88
 - 定義 85

ターゲット表 (続き)

ストレージ要件 8
定義、行の 98
定義、ターゲット・キーの 100
定義、列の 98
適用、行のサブセット 98
適用、列のサブセット 98
のリスト 542
表の構造、クイック・リファレンス 492
フラグメント化 98
変更集約
構造 542
使用法 89
定義 85
ポイント・イン・タイム
構造 545
使用法 88
定義 85
保守 256
マッピング、ソースへの 82
ユーザー定義 87, 97
ユーザー・コピー
構造 547
使用法 88
定義 85
レプリカ
競合検出 12
構造 546
使用法 94
定義 85
CCD (整合変更データ)
概要 85
構造 543
ターゲット・キー 100
ターゲット・キー列
更新 102
ターゲット・サーバー
上の表 542
ログの影響 7
対等レプリケーション、競合検出 12
大量のレプリケーション・ジョブ 75
チューニング
commit_interval パラメーター 4

チューニング (続き)

memory_limit パラメーター 4
地理情報データ・タイプ 105
データ
高度なサブセット化技法 117
サブセット化
使用、述部の 119
使用、ビューの 118
使用する、トリガーを、CD表で 118
使用する、ビューを、述部を指定するするために 119
登録時の 117
ソース表からのリトリート 256
操作 123
トランスフォーム
作成、算出列の 125
サブスクリプションでの 124
登録での 123
名前変更、列の 125
ヒストリーの表示 179
防止、二重削除の 65
データ暗号化の制約事項 105
データ共有モード 464
データ整合性 97
データベース、変更キャプチャーの使用可能化 272
データ・タイプ
制約事項 105
マッピング、列間の 100
レプリケーション
ラージ・オブジェクト (LOB) 106
DATALINK 値 107
データ・ブロッキング 75
データ・リンク
レプリケーション 107
停止
アプライ・プログラム
OS/400 版 165, 409
UNIX 版 165, 324
Windows 版 165, 324
z/OS 版 165, 324
キャプチャー・プログラム
OS/400 版 148, 413
UNIX 版 148, 340

停止 (続き)

キャプチャー・プログラム (続き)
Windows 版 148, 340
z/OS 版 148, 340
レプリケーション・アラート・モニター
UNIX 版 196, 346
Windows 版 196, 346
z/OS 版 196, 346
ディスク・スペース
一時ファイル 10
要件 5
出口ルーチン
ジャーナル・レシーバー削除 (OS/400) 40
ASNDCOPY 109
ASNDONE
使用 166, 167
ASNLOAD
カスタマイズ 173
使用 169
OS/400 用 174
UNIX 版 169
Windows 版 169
z/OS 版 171
デフォルト
アプライ・パラメーターの (UNIX、Windows、z/OS) 154
アプライ・プログラムのパラメーターの (OS/400) 165
キャプチャー・パラメーターの (OS/400) 130, 131
キャプチャー・パラメーターの (UNIX、Windows、z/OS) 129, 135
動的なコントロール表 247
登録
オブジェクト 199
オプション、ソース用の
エラー発生時にキャプチャーを停止 54
行 (水平方向) のサブセット化 49
競合検出 60
更新、削除および挿入として 54

登録 (続き)

オプション、ソース用の (続き)
使用、リモート・ジャーナル
62
相対レコード番号 63
フル・リフレッシュ・コピー
48
変更キャプチャー・レプリケ
ーション 48
変更後イメージ列 50
変更前イメージ接頭部 53
変更前イメージ列 50
リキャプチャー、変更の
(update-anywhere) 55
列 (垂直方向) のサブセット化
48
再活動化 205
削除 207, 431
属性の変更 200
追加 366
非 DB2 リレーショナル・デー
タ・ソース 46
非活動化 204
ビュー
概要 64, 66
プロシージャー 199
表 199
変更のキャプチャーの停止 204
列の追加 201
DB2 表 43
IMS データ・ソース 43
登録 (REGISTER) 表 505
登録拡張 (REG_EXT) 表 504
登録同期 (REG_SYNCH) 表 510
登録変数
DB2CODEPAGE 15, 31
DB2DBDFT 31
DB2INSTANCE 30
特殊なデータ・タイプ
レプリケーション
ラージ・オブジェクト
(LOB) 106
DATALINK 値 107
トラブルシューティング・コマンド
asntrc 358
WRKDPTRC 454

トランザクション

メモリー、使用する 3
トランザクション・スループット率
キャプチャー・トリガー 13
トランザクション・モード処理 7,
78
トランスフォーム、データの
作成、算出列の 125
サブスクリプションでの 124
登録での 123
名前変更、列の 125
トリガー
キャプチャー、データの 12
マージ 13
抑制、データのキャプチャーの
118
CD 表の 118
トレース機能
OS/400 版 454
UNIX 版 358
Windows 版 358
z/OS 版 358

[ナ行]

内部 CCD 表
複数のターゲット 90
内部結合、ソースとしての 64
名前
アプライ修飾子の規則 321
キャプチャー・スキーマの規則
321
キャプチャー・トリガーの 13
サブスクリプション・セット
215
表示名 468
モニター修飾子の規則 321
レプリケーション・サービスの
468
Windows サービス用 322
名前変更、列の 125
二重削除 65
ニックネーム
クロスローダー・ユーティリテー
ーの 173

ニックネーム (続き)

制約事項
集約表 88
CCD 表の場合 51
multi-tier レプリケーション
92
update-anywhere 55, 95
登録 46
入出力エラーのリカバー、コントロ
ール表 255
認証、エンド・ユーザー
UNIX 版 17, 26
Windows 版 17, 26
ネットワーク・コネクティビティー
17

[ハ行]

バイナリー・ラージ・オブジェクト
(BLOB)
レプリケーションの考慮事項
106
バインド
アプライ・プログラム
UNIX 版 33
Windows 版 33
z/OS 版 36
キャプチャー・プログラム
UNIX 版 32
Windows 版 32
z/OS 版 36
レプリケーション・アラート・モ
ニター
UNIX 版 33
Windows 版 33
パスワード・ファイル
保管 26
asnpwd コマンド 352
パッケージ、再バインド 248
バッチ・ジョブ
実行 461
メモリー、使用する 3
パラメーター、呼び出し
アナライザー
OS/400 版 401
UNIX 版 326

パラメーター、呼び出し (続き)

- アナライザー (続き)
 - Windows 版 326
- アプライ・プログラム
 - OS/400 版 164, 439
 - UNIX 版 154, 329
 - Windows 版 154, 329
 - z/OS 版 154, 329
- キャプチャー・プログラム
 - OS/400 版 448
 - UNIX 版 135, 335
 - Windows 版 135, 335
 - z/OS 版 135, 335
- レプリケーション・アラート・モニター
 - UNIX 版 349
 - Windows 版 349
 - z/OS 版 349
- 非 DB2 リレーショナル・データ・ソース
 - 使用、CCD 表の 46
 - 制約事項
 - 集約表 88
 - multi-tier レプリケーション 92
 - update-anywhere 55, 60, 95
 - ソース・サーバー 13
 - 登録 46
 - ロック 13
- 非アクティブなサブスクリプション・セット 75
- 非活動化
 - サブスクリプション・セット 75, 229
 - 登録済みオブジェクト 204
- 非活動化、サブスクリプション・セットの 279
- ヒストリー・データ
 - ソース・データ 53
 - CCD 表 89
- ビュー
 - 使用、関連 ID の 64
 - 制約事項 64, 66
 - 属性の変更 200
 - 登録
 - 概要 64

ビュー (続き)

- 登録 (続き)
 - ソースとして 66
 - プロシージャ 199
- 表
 - アプライ・コントロール・サーバー上の 517
 - 基礎集約 542
 - キャプチャー・コントロール・サーバー上の 493
 - 競合検出 12
 - 構造 481
 - コントロール表
 - 再編成 248
 - 作成 27
 - 静的 249
 - 接続障害のリカバー 255
 - 動的 247
 - 入出力エラーのリカバー 255
 - プルーニング 250
 - 保守 247
 - RUNSTATS ユーティリティ 248
 - 再活動化 205
 - 属性の変更 200
 - ターゲット表
 - 保守 256
 - 参照：ターゲット表
 - ターゲット・サーバー上の 542
- 登録
 - 非 DB2 リレーショナル 46
 - プロシージャ 199
 - DB2 43
 - 登録の除去 207
 - 非活動化 204
 - 変更集約 542
 - 変更のキャプチャーの停止 204
 - ポイント・イン・タイム 545
 - モニター・コントロール・サーバー上の 534
 - ユーザー・コピー 547
 - 列の追加 201
 - レプリカ 12, 546
- ALERTS (モニター・アラート) 534

表 (続き)

- APPENQ (アプライ・エンキュー) 517
- APPLYTRACE (アプライ・トレース) 518
- APPLYTRAIL (アプライ・トレール) 519
- APPLY_JOB (アプライ・ジョブ) 518
- AUTHTKN (アプライ修飾子相互参照) 493
- CAPENQ (キャプチャー・エンキュー) 494
- CAPMON (キャプチャー・モニター) 253, 495
- CAPPARMS (キャプチャー・パラメーター) 496
- CAPSCHEMAS (キャプチャー・スキーマ) 493
- CAPTRACE (キャプチャー・トレース) 253, 498
- CCD (整合変更データ)
 - キャプチャー・コントロール・サーバー 499
 - ターゲット・サーバー 543
- CCD 表の保守 67
- CD (変更データ) 500
- CONDITIONS (モニター条件) 535
- CONTACTGRP (モニター・グループ連絡先) 538
- CONTACTS (モニター連絡先) 538
- GROUPS (モニター・グループ) 539
- MONENQ (モニター・エンキュー) 539
- MONSERVERS (モニター・サーバー) 539
- MONTRACE (モニター・トレース) 540
- PRUNCNTL (プルーニング・コントロール) 501
- PRUNE_LOCK (プルーニング・ロック) 503

表 (続き)

PRUNE_SET (プルーニング・セット) 503
 REGISTER (登録) 505
 REG_EXT (登録拡張) 504
 REG_SYNCH (登録同期) 510
 RESTART (再始動) 511
 SEQTABLE (順序付け) 512
 SIGNAL (シグナル) 513
 SUBS_COLS (サブスクリプション列) 523
 SUBS_EVENT (サブスクリプション・イベント) 524
 SUBS_MEMBR (サブスクリプション・メンバー) 173, 525
 SUBS_SET (サブスクリプション・セット) 528
 SUBS_STMTS (サブスクリプション・ステートメント) 532
 UOW (作業単位) 515

表構造 481
 表示名 468
 表モード処理 7, 78
 非リレーショナル・データ・ソース
 使用、CCD 表の 43
 CCD 表の保守 67

ヒント
 アプライ・トレール表からの行の
 削除 163
 アプライ・プログラムがセットを
 正常に処理したかのチェック
 163
 検査する、変更のキャプチャーが
 開始されたことを 134
 サービスが正しくセットアップさ
 れたことの検証 467
 スリープを使用するか copyonce
 パラメーターを使用するか
 157, 161
 セットの追加処理にストアード・
 プロシージャを使用 166
 見積もる、使用を、スペースの
 6
 ASNDONE でストアード・プロシ
 ージャを使用 167

ファイル

予備 10
 asndone.smp 167
 asnload.ini 174
 *.APP.log 159
 *.CAP.log 139
 *.err 162
 *.sqz 162, 163

ファイル・コピー・デーモン
 ASNDLCOPYD 113
 DLFM_ASNCOPYD 111

複数のターゲット表 90
 フラグメント化
 垂直
 ソースでの 48
 ターゲットでの 98

水平
 ソースでの 49
 ターゲットでの 98
 対等レプリケーション 12
 update-anywhere レプリケーシ
 ョン 12

プラン、再バインド 248

プルーニング
 アプライ・トレース
 (APPLYTRACE) 表 253
 アプライ・トレール
 (APPLYTRAIL) 表 253
 キャプチャー・トレース
 (CAPTRACE) 表 253
 キャプチャー・プログラム
 UNIX 版 340
 Windows 版 340
 z/OS 版 340

キャプチャー・モニター
 (CAPMON) 表 253

コントロール表 250
 シグナル (SIGNAL) 表 253
 CD (変更データ) 表 252
 UOW (作業単位) 表 252, 515

プルーニング・コントロール
 (PRUNCNTL) 表 501
 プルーニング・セット (PRUNE_SET)
 表 503
 プルーニング・ロック
 (PRUNE_LOCK) 表 503

フル・リフレッシュ・コピー

強制 281
 登録オプション 48
 Apply for iSeries 63, 441

ブロッキング因数 75

プロファイル
 記述 265
 コントロール表 265
 ソース・オブジェクト 266
 ターゲット・オブジェクト 267

プロモート
 サブスクリプション・セット
 280
 登録済みの表またはビュー 280
 レプリケーション構成 239

分割
 サブスクリプション・セット
 218

分散リカバー点 235
 変換、データの 15
 変更キャプチャー
 使用可能化 272

変更キャプチャー・レプリケーシ
 ョン
 記述 49
 登録オプション 48
 変更後イメージ列 50
 変更集約表
 構造 542
 使用法 89
 定義 85

変更前イメージ接続部 53
 変更前イメージ列
 制約事項 52
 登録 50
 変更集約表 99

変更データ (CD) 表
 構造 500
 ストレージ要件 9
 プルーニング 252
 要約、内容の 89

変更のキャプチャーの停止 204
 編集、SQL スクリプトの 127

ポイント・イン・タイム表
 構造 545
 使用法 88

[マ行]

マージ
サブスクリプション・セット 223
トリガー 13
マスター表 (update-anywhere)
概要 94
リキャプチャー、変更の 55
待ち時間
アプライ・プログラム 183
キャプチャー・プログラム 182
マッピング
ソースからターゲット 82
ソース列からターゲット列への 100
データ・タイプ、表間の 100
ミニサイクル 75
メッセージ 181, 182, 549
メッセージ・キュー、ジャーナル用 40
メモリー
アプライ・プログラム 5
キャプチャー・プログラム 3
計画 3
サブスクリプション・セット 5
使用、CAPMON 表、調整のために 4
登録 4
トランザクション 3
バッチ・ジョブ 3
読み取り、ログ・レコードの 4
レプリケーション・アラート・モニター 5
文字ラージ・オブジェクト (CLOB)
レプリケーションの考慮事項 106
モニター
自動化 184
履歴の傾向 179
プログラムの状況 179
OS/400 版 196
参照：レプリケーション・アラート・モニター
モニター修飾子、命名規則 321

モニター条件 (CONDITIONS) 表 535
モニター連絡先 (CONTACTS) 表 538
モニター・アラート (ALERTS) 表 534
モニター・エンキュー (MONENQ) 表 539
モニター・グループ (GROUPS) 表 539
モニター・グループ連絡先 (CONTACTGRP) 表 538
モニター・コントロール表のリスト 534
ALERTS (モニター・アラート) 534
CONDITIONS (モニター条件) 535
CONTACTGRP (モニター・グループ連絡先) 538
CONTACTS (モニター連絡先) 538
GROUPS (モニター・グループ) 539
MONENQ (モニター・エンキュー) 539
MONSERVERS (モニター・サーバー) 539
MONTRACE (モニター・トレース) 540
MONTRAIL (モニター・トレール) 541
モニター・コントロール・サーバー上のコントロール表 534
作成、コントロール表の 185
レプリケーション・センターへの追加 271
モニター・サーバー (MONSERVERS) 表 539
モニター・トレース (MONTRACE) 表 540
モニター・トレール (MONTRAIL) 表 541

[ヤ行]

ユーザー ID
アプライ・プログラム 24
キャプチャー・トリガーの場合 23
キャプチャー・プログラムの場合 22
許可 22
パスワード・ファイル 26
レプリケーション・アラート・モニター用 26
ユーザー定義のデータ・タイプ 106
ユーザー定義表 87, 97
ユーザー・コピー表
構造 547
使用法 88
定義 85
ユニコード表 659
要約データ・タイプ 105
呼び出しパラメーター
アナライザー
OS/400 版 401
UNIX 版 326
Windows 版 326
アプライ・プログラム
OS/400 版 164, 439
UNIX 版 154, 329
Windows 版 154, 329
z/OS 版 329
z/OS 用 154
キャプチャー・プログラム
OS/400 版 129, 145, 448
UNIX 版 135, 335
Windows 版 135, 335
z/OS 版 335
z/OS 用 135
レプリケーション・アラート・モニター
UNIX 版 349
Windows 版 349
z/OS 版 349
予備ファイル
ストレージ、アプライ用 11
ストレージ、キャプチャー用 10
ストレージ、診断ファイル用 10

[ラ行]

ラージ・オブジェクト (LOB)

レプリケーションの考慮事項

106

ランチパッド 263

リカバー点、分散 235

リキャプチャー、変更の

(update-anywhere) 55

リモート・ジャーナル、ソースとして 62

リモート・ソース表 62

レシーバーのサイズ、現行 7

列

計算 98

サブセット化

ソースでの 48

ターゲットでの 98

算出 125

使用可能、レプリケーションに 48

相対レコード番号、OS/400 での 63

定義、ターゲット表での 98

登録、ソース表内の 48

登録済みソース表への追加 201

名前変更 125

変更後イメージ 50

変更前イメージ 50

マッピング、ソースからターゲットへの 100

列 (垂直方向) のサブセット化

ソースでの 48

ターゲットでの 98

レプリカ表

構造 546

定義 85

定義、読み取り / 書き込みターゲットの 94

リキャプチャー、変更の 55

レプリケーション環境

コピー 239

レプリケーション構成のコピー 239

レプリケーション・アナライザー

OS/400 版

作成、SQL パッケージの 36

呼び出しパラメーター 401

UNIX 版、呼び出しパラメーター 326

Windows 版、呼び出しパラメーター 326

レプリケーション・アラート・モニター

アラート条件の選択 187

お互いの通信

アプライ・プログラム 478

キャプチャー 478

レプリケーション・センター 477

許可要件 26

コントロール表 185

再初期化 195

実行時間の指定 190

出力の保管 191

スケジューリング 195, 471, 472

説明 184

操作 285

通知の設定 192, 194

停止 196

トレース 192

プルーニング 192

メモリーの使用量 5

連絡先のコピー 187

連絡先の定義 186

連絡先の変更 186

ASNMAIL 出口 194

UNIX 版

開始 189, 348

始動 663

状況のチェック 177

操作 346

バインド 33

Windows 版

開始 189, 348

始動 663

状況のチェック 177

操作 346

バインド 33

レプリケーション・アラート・モニター (続き)

z/OS 版

開始 189, 348

状況のチェック 177

操作 346

z/OS 用

開始 461

レプリケーション・イベントの調整 232

レプリケーション・コマンド

バックアップ・データベース 31

ADDJOBSCDE 472

ASNL2RNx 461

AT 471, 472

AT NetView

Apply for z/OS 472

Capture for z/OS 472

CRTJRNRVCV 37

db2rc 261

DSPJRN 196

OS/400 版

ADDDPRREG 366

ADDDPRSUB 374

ADDDPRSUBM 390

ADDEXITPGM 41

ANZDPR 400

ANZDPRJRN 40

CHGDPRCAPA 404

CHGJRN 40

CRTDPRTBL 408

CRTJRN 38

ENDDPRAPY 409

ENDDPRCAP 148, 413

ENDJOB 414

GRTDPRAUT 36, 416

GRTOBJAUT 36

INZDPRCAP 424

OVRDPRCAPA 426

RCVJRNE 39

RMVDPRREG 431

RMVDPRSUB 432

RMVDPRSUBM 434

RMVEXITPGM 41

RVKDPRAUT 436

STRDPRAPY 165, 438

| | | |
|--|---------------------------|----------------------------------|
| レプリケーション・コマンド (続き) | レプリケーション・コマンド (続き) | レプリケーション・センター (続き) |
| OS/400 版 (続き) | \$TA JES2 | 非活動化、サブスクリプション・セットの 279 |
| STRDPRCAP 446 | Apply for z/OS 472 | フル・リフレッシュの強制 281 |
| STRJRNP 38 | Capture for z/OS 472 | プロファイル 265 |
| WRKDPTRC 454 | レプリケーション・サービス | プロモート関数 239 |
| WRKJOB 179 | 作成 355, 467 | ユーザー ID およびパスワード 264 |
| WRKREGINF 41 | 操作 469 | ランチパッド 263 |
| WRKSBJOB 179 | ドロップ 357, 469 | レプリケーション・アラート・モニターの操作 285 |
| WRKSBSJOB 179 | 名前 468 | レプリケーション・センターの始動 261 |
| OS/400 用 | レプリケーション・センター | レプリケーション・センターのパスワード 264 |
| SBMJOB 472 | アプライ・プログラムの操作 284 | レプリケーション・センターのユーザー ID 264 |
| UNIX 版 | お互いの通信 | レプリケーション・ソース結合 64 |
| asnacmd 324 | アプライ・プログラム 473 | サブスクライブ 72 |
| asnanalyze 325 | キャプチャー・トリガー 473 | 登録 |
| asnapply 328 | キャプチャー・プログラム 473 | 行 49 |
| asncap 334 | レプリケーション・アラート・モニター 477 | 非 DB2 リレーショナル・データ・ソース 46 |
| asnccmd 340 | 活動化、サブスクリプション・セットの 279 | ビュー 66 |
| asnmcmd 346 | 記述 259 | 列 48 |
| asnmon 348 | キャプチャー・プログラムの操作 283 | DB2 表 43 |
| asnpwd 352 | コネクティビティ 17 | IMS データ・ソース 43 |
| asntrc 358 | コントロール表 268 | マッピング、ターゲットへの CCD (整合変更データ) 表 92 |
| update database configuration (データベース構成の更新) 31 | コントロール表プロファイル 265 | CCD 表の保守 67 |
| Windows 版 | サーバーの追加 271 | レプリケーション・メッセージ 549 |
| asnacmd 324 | サブスクリプション・セットの作成 275 | 連絡先 |
| asnanalyze 325 | サブスクリプション・セットのプロモート 280 | レプリケーション・アラート・モニター用 186 |
| asnapply 328 | 始動 261 | ロールフォワード・リカバリー ログ |
| asncap 334 | ソースの登録 273 | 計画、影響の 13 |
| asnccmd 340 | ソース・オブジェクト・プロファイル 266 | ログ要件 |
| asnmcmd 346 | ターゲット・オブジェクト・プロファイル 267 | ターゲット・サーバー 7 |
| asnmon 348 | データベースの変更キャプチャーの使用可能化 272 | 非 DB2 リレーショナル・ソース・サーバー 13 |
| asnpwd 352 | 定義の削除 283 | DB2 ソース・サーバー 6 |
| asnsert 355 | 定義の除去 283 | ログ・レコード |
| asnsdrop 357 | 登録済みの表またはビューのプロモート 280 | アーカイブされた、キャプチャーの前に 7 |
| asntrc 358 | | |
| z/OS 版 | | |
| asnacmd 324 | | |
| asnapply 328 | | |
| asncap 334 | | |
| asnccmd 340 | | |
| asnmcmd 346 | | |
| asnmon 348 | | |
| asnpwd 352 | | |
| asnsert 355 | | |
| asnsdrop 357 | | |
| asntrc 358 | | |
| z/OS 用 | | |
| MODIFY 462 | | |

ログ・レコード (続き)
 圧縮辞書 (z/OS) 246
 保守 242
 保存 242
 ロック
 CCD 表に対する 13
 論理パーティション化キー
 記述 54

[数字]

2 バイト文字ラージ・オブジェクト
 (DBCLOB)
 レプリケーションの考慮事項
 106
3 層レプリケーション構成 93

A

ADDDPRREG コマンド 366
ADDDPRSUB コマンド 374
ADDDPRSUBM コマンド 390
ADDEXITPGM コマンド 41
ADDJOBSCDE コマンド 472
ALERTS (モニター・アラート) 表
 534
alert_prune_limit パラメーター 192
ALWINACT パラメーター 442
ANZDPR コマンド 400
ANZDPRJRN コマンド 40
APPENQ (アプライ・エンキュー) 表
 517
APPLHEAPSZ 構成パラメーター 31
APPLYTRACE (アプライ・トレース)
 表
 構造 518
 プルーニング 253
APPLYTRAIL (アプライ・トレール)
 表
 構造 519
 プルーニング 253
APPLY_JOB (アプライ・ジョブ) 表
 518
apply_path パラメーター 155, 330
apply_qual パラメーター 156, 324,
 329
APYQUAL パラメーター 440
ARM (自動リスタート・マネージャ
 ー) 463
ASCII 表 659
ASN メッセージ 549
asnacmd コマンド 324
asnanalyze コマンド 325
asnapply コマンド 328
asnacp コマンド 334
asnccmd コマンド 340
ASNDLCOPY 出口ルーチン 109
ASNDLCOPYD ファイル・コピー・
 デーモン 113
ASNDONE 出口ルーチン
 使用 166, 167
 リジェクトされたトランザクシ
 ョン 62
asndone.smp ファイル 167
ASNL2RNx コマンド 461
ASNLOAD 出口ルーチン
 エラー処理 169
 記述 169
 クロスローダー・ユーティリテ
 ーの使用 173
 前提条件 169
 動作のカスタマイズ 173
 ファイル、生成される 170
asnload.ini ファイルの使用 174
ATALINK レプリケーションの
 108
OS/400 版 174
UNIX 版 169
Windows 版 169
z/OS 版 171
asnload.ini ファイル 174
ASNMAIL 出口 194
asnmcmd コマンド 346
asnmon コマンド 348
ASNPLXFY ユーティリティー 464
asnpwd コマンド 352
asnscrt コマンド 355
asnscdrop コマンド 357
asntrc コマンド 358
AT NetView コマンド
 Apply for z/OS 472
 Capture for z/OS 472

AT コマンド
 アプライ・プログラム 471, 472
 キャプチャー・プログラム 471,
 472
 レプリケーション・アラート・モ
 ニター 471, 472
AUTHTKN (アプライ修飾子相互参
 照) 表 493
autoprune パラメーター
 概要 135
 レプリケーション・アラート・モ
 ニターからのプルーニング 192
 asncap コマンドで使用 336
 asnccmd コマンドで使用 343
autostop パラメーター 136, 336,
 343

B

backup database (データベースのバック
 アップ) コマンド 31
BLOB (バイナリー・ラージ・オブジ
 ェクト)
 レプリケーションの考慮事項
 106

C

CALL プロシージャー
 定義、サブスクリプション・セッ
 トの 80
 レプリケーション前後の実行時処
 理 124
CAPCTLLIB パラメーター 449
CAPENQ (キャプチャー・エンキュー
 ー) 表 494
CAPMON (キャプチャー・モニター)
 表
 構造 495
 プルーニング 253
CAPPARMS (キャプチャー・パラメ
 ーター) 表
 構造 496
 使用 131
 変更 147

CAPSCHEMAS (キャプチャー・スキーマ) 表 493
CAPSTART シグナル 236
CAPSTOP シグナル 238
CAPTRACE (キャプチャー・トレース) 表
構造 498
ブルーニング 253
capture_path パラメーター 136, 335
capture_schema パラメーター 137, 335, 341
capture_server パラメーター 138, 335, 341
CCD (整合変更データ) 表
外部
multi-tier レプリケーション 92
構造
キャプチャー・コントロール・サーバー 499
ターゲット・サーバー 543
使用方法
ヒストリーまたは監査 89
multi-tier レプリケーション 92
追加、UOW 列の 90
内部
複数のターゲット 90
非 DB2 リレーショナル・データ・ソース
使用、CCD 表の 46
非リレーショナル・データ・ソース
使用、CCD 表の 43
CCD 表の保守 67
レプリケーション・ソース 92
ロック 13
CD (変更データ) ビュー 64
CD (変更データ) 表
結合用 64
構造 500
ストレージ要件 9
トリガー 118
ビュー用の 64
ブルーニング 252
要約、内容の 89

CHGDPRCAPA コマンド 404
CHGJRN コマンド 40
CLNUPITV パラメーター 449
CLOB (文字ラージ・オブジェクト)
レプリケーションの考慮事項 106
cold 始動モード 144
commit_interval パラメーター
概要 138
チューニング 4
asncap コマンドで使用 336
asnccmd コマンドで使用 343
CONDITIONS (モニター条件) 表 535
CONTACTGRP (モニター・グループ
連絡先) 表 538
CONTACTS (モニター連絡先) 表 538
control_server パラメーター 156, 324, 329
COPYONCE パラメーター 443
copyonce パラメーター 156, 331
CRTDPRTBL コマンド 408
CRTJRN コマンド 38
CRTJRNRCV コマンド 37
CTLSVR パラメーター 440

D

Data Links Manager レプリケーション・デーモン 111
DATALINK 値
制約事項 60, 95
保管、更新の 55
レプリケーション 107
ASNDLCOPY 出口ルーチン 109
ASNDLCOPYD ファイル・コピー・デーモン 113
DLFM_ASNCOPYD ファイル・コピー・デーモン 111
DB2 Enterprise Server Edition
制約事項 49
DB2 Extenders
制約事項 107
DB2 ビュー
登録 66

DB2 表
登録 43
DB2 レプリケーション
許可要件 20
DB2CODEPAGE 環境変数 15, 31
DB2DBDFT 環境変数 31
DB2INSTANCE 環境変数 30
db2rc コマンド 261
db2_subsystem パラメーター 157, 329
DBADM 21, 22
DBCLOB (2 バイト文字ラージ・オブジェクト)
レプリケーションの考慮事項 106
DBHEAP 構成パラメーター 31
DELAY パラメーター 442
delay パラメーター 157, 332
DLFM_ASNCOPYD ファイル・コピー・デーモン 111
DPR 登録 (OS/400)
削除 431
追加 366
DSPJRN コマンド 196

E

EDITPROC 文節
制約事項、圧縮 105
email_server パラメーター 192
ENDDPRAPY コマンド 409
ENDDPRCAP コマンド 148, 413
ENDJOB コマンド 414
errwait パラメーター 158, 332

F

FIELDPROC 文節
制約事項、圧縮 105
FRCFRQ パラメーター 452
FULLREFPGM パラメーター 441

G

GROUPS (モニター・グループ) 表 539

GRTPRAUT コマンド 36
 構文 416
GRTOBJAUT コマンド 36

I

IMS DataPropagator 43
IMS データ・ソース
 使用、CCD 表の 43
 登録 43
 CCD 表の保守 67
INACTMSG パラメーター 442
inamsg パラメーター 158, 331
INZDPRCAP コマンド 424
iSeries サーバー
 接続 18

J

JCL
 アプライ・プログラムの始動
 461
 キャプチャー・プログラムの始動
 461
 レプリケーション・アラート・モ
 ニターの開始 461
JOBDD パラメーター 439, 448
JOIN_UOW_CD 列 119
JRN パラメーター 450

L

LAG パラメーター 451
lag_limit パラメーター 138, 336,
 343
LIBPATH 31
loadxit パラメーター 158, 330
LOB (ラージ・オブジェクト)
 レプリケーションの考慮事項
 106
 update-anywhere の制約事項 95
LOGBUFFSZ 構成パラメーター 31
LOGFILSIZ 構成パラメーター 31
LOGPRIMARY 構成パラメーター
 31

logreuse パラメーター (アプライ
 用) 159, 330
logreuse パラメーター (キャプチャー
 用) 139, 336, 344
logreuse パラメーター (レプリケーシ
 ョン・アラート・モニターの) 191
LOGSECOND 構成パラメーター 31
logstdout パラメーター (アプライ
 用) 159, 330
logstdout パラメーター (キャプチャ
 ー用) 139, 337, 344
logstdout パラメーター (レプリケー
 ション・アラート・モニター
 の) 191
LONG VARGRAPHIC データ・タイ
 プ 105

M

MAXAPPLS 構成パラメーター 31
max_notifications_minutes パラメータ
 ー 192
max_notifications_per_alert パラメータ
 ー 192
MAX_SYNCH_MINUTES、データ・
 ブロッキング 75
MEMLMT パラメーター 451
memory_limit パラメーター
 概要 140
 チューニング 4
 asncap コマンドで使用 337
 asnccmd コマンドで使用 344
Microsoft SQL Server
 レプリケーションの制約事項 51
MODIFY コマンド 462
MONENQ (モニター・エンキュー)
 表 539
monitor_errors パラメーター 194
monitor_interval パラメーター (キャ
 プチャー用) 140, 337, 344
monitor_interval パラメーター (レプ
 リケーション・アラート・モニター
 の) 190
monitor_limit パラメーター 140,
 337, 344
monitor_path パラメーター 191

MONITV パラメーター 450
MONLMT パラメーター 450
MONSERVERS (モニター・サーバ
 ー) 表 539
MONTRACE (モニター・トレース)
 表 540
MONTRAIL (モニター・トレール)
 表 541
multi-tier レプリケーション
 定義、サブスクリプション・セッ
 トの 92
MVS コンソール 461

N

NLS (各国語サポート) 15
notify パラメーター 159, 331

O

opt4one パラメーター 160, 332
OPTSNGSET パラメーター 444
Oracle ソース
 制約事項 51
OS/400 データ・ソース
 リモート・ジャーナリングを使用
 する 62
OVRDPRCAPA コマンド 426

P

PREDICATES 列 119
PRUNCNTL (プルーニング・コント
 ロール) 表 501
prune_interval パラメーター 141,
 337, 344
PRUNE_LOCK (プルーニング・ロッ
 ク) 表 503
PRUNE_SET (プルーニング・セット)
 表 503
pwdfile パラメーター 160, 330

R

RCVJRNE コマンド 39
REGISTER (登録) 表 505
REG_EXT (登録拡張) 表 504
REG_SYNCH (登録同期) 表 510
RESTART (再始動) 表 511
RESTART パラメーター 448
RETAIN パラメーター 451
retention_limit パラメーター 142, 337, 345
RMVDPRREG コマンド 431
RMVDPRSUB コマンド 432
RMVDPRSUBM コマンド 434
RMVEXITPGM コマンド 41
ROWID 106
RRN 63
RTYWAIT パラメーター 443
runonce パラメーター 190
RUNSTATS ユーティリティ 248
RVKDPRAUT コマンド 436

S

SBMJOB コマンド 472
SCM (Service Control Manager)
レプリケーション・サービスの作成 355, 467
レプリケーション・サービスの操作 469
レプリケーション・サービスのドロップ 357, 469
レプリケーション・サービスの命名 468
SEQTABLE (順序付け) 表 512
Service Control Manager (SCM)
レプリケーション・サービスの作成 355, 467
レプリケーション・サービスの操作 469
レプリケーション・サービスのドロップ 357, 469
レプリケーション・サービスの命名 468
SIGNAL (シグナル) 表
構造 513

SIGNAL (シグナル) 表 (続き)
ブルーニング 253
sleep パラメーター 160, 332
sleep_interval パラメーター 142, 337, 345
spillfile パラメーター 161, 333
SQL スクリプト 127
SQL ステートメント
実行時処理 124
定義、サブスクリプション・セットの 80
SQL パッケージ
作成、アプライ・プログラム用の 35
作成、キャプチャー・プログラム用の 35, 36
作成、レプリケーション・アナライザー用 36
SQL ファイル、編集 127
sqlerrorcontinue パラメーター 162, 333
startmode パラメーター 143, 338
status
アプライ・プログラム 177, 179
キャプチャー・プログラム 177, 179
ジャーナル・ジョブ 179
レプリケーション・アラート・モニター 177
STOP シグナル 233, 235
STRDPRAPY コマンド 165, 438
STRDPRCAP コマンド 446
STRJRNPf コマンド 38
SUBNFYPGM パラメーター 442
SUBS_COLS (サブスクリプション列) 表 523
SUBS_EVENT (サブスクリプション・イベント) 表
構造 524
通知、イベントの 81
SUBS_MEMBR (サブスクリプション・メンバー) 表 173, 525
SUBS_SET (サブスクリプション・セット) 表 528
SUBS_STMTS (サブスクリプション・ステートメント) 表 532

Sybase
レプリケーションの制約事項 51
SYSADM 21, 22

T

term パラメーター (アプライ用) 163, 333
term パラメーター (キャプチャー用) 144, 339, 345
TRACE パラメーター 441
trace_limit パラメーター
概要 144
レプリケーション・アラート・モニターからのブルーニング 192
asncap コマンドで使用 339
asnccmd コマンドで使用 345
asnmon コマンドで使用 350
TRCLMT パラメーター 450
TRLREUSE パラメーター 443
trlreuse パラメーター 163, 332
TSO 461

U

UOW (作業単位) 表
構造 515
ストレージ要件 9
ブルーニング 252, 515
列、CCD 表の 90
UOW_CD_PREDICATES 列 119
update database configuration (データベース構成の更新) コマンド 31
update-anywhere レプリケーション競合検出
概要 60
計画 12
要件 53, 60
定義、サブスクリプション・セットの 94
フラグメント化 12
リキャプチャー、変更の 55
USER シグナル 232
USER パラメーター 439

V

VALIDPROC 文節 105

W

WAIT パラメーター 448

warmns 始動モード 143

warmsa 始動モード 143

warmsi 始動モード 143

WHERE 文節

行サブセット 98

PREDICATES 列の制約事項 119

Windows Service Control Manager
(SCM) 467

Windows サービス名 322

WRKDPTRTC コマンド 454

WRKJOB コマンド 179

WRKREGINF コマンド 41

WRKSBJOB コマンド 179

WRKSBSJOB コマンド 179

Z

z/OS サーバー

接続 18

[特殊文字]

区切り文字 128

\$TA JES2 コマンド 472

*.APP.log ファイル 159

*.CAP.log ファイル 139

*.err ファイル 162

*.sqz ファイル 162, 163

; 区切り文字 128

特記事項

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品、プログラムまたはサービスの操作性の評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権の許諾については、下記の宛先に書面にてご照会ください。

〒106-0032 東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

本書に含まれる情報には、技術的に不正確なもの、または誤植が含まれる場合があります。これらに対する変更は、定期的に行われます。これらの変更は、資料の改訂版に含まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム（本プログラムを含む）との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常のビジネス・プロセスで用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームの

アプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。したがって IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのすべての部分、またはすべての派生した創作物には、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。 © Copyright IBM Corp. _西暦年_. All rights reserved.

商標

以下は、IBM Corporation の商標です。

| | |
|---|------------------|
| ACF/VTAM | LAN Distance |
| AISPO | MVS |
| AIX | MVS/ESA |
| AIXwindows | MVS/XA |
| AnyNet | Net.Data |
| APPN | NetView |
| AS/400 | OS/390 |
| BookManager | OS/400 |
| C Set++ | PowerPC |
| C/370 | pSeries |
| CICS | QBIC |
| Database 2 | QMF |
| DataHub | RACF |
| DataJoiner | RISC System/6000 |
| DataPropagator | RS/6000 |
| DataRefresher | S/370 |
| DB2 | SP |
| DB2 Connect | SQL/400 |
| DB2 Extenders | SQL/DS |
| DB2 OLAP Server | System/370 |
| DB2 Universal Database | System/390 |
| Distributed Relational Database Architecture | SystemView |
| DRDA | Tivoli |
| eServer | VisualAge |
| Extended Services | VM/ESA |
| FFST | VSE/ESA |
| First Failure Support Technology | VTAM |
| IBM | WebExplorer |
| IMS | WebSphere |
| IMS/ESA | WIN-OS/2 |
| iSeries | z/OS |
| | zSeries |

以下は、他社の商標または登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Action Media、LANDesk、MMX、Pentium および ProShare は Intel Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

UNIX は、The Open Group がライセンスしている米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名などはそれぞれ各社の商標または登録商標です。

IBM と連絡をとる

技術上の問題がある場合は、お客様サポートに連絡をとってください。

製品情報

以下の情報は英語で提供されます。内容は英語版製品に関する情報です。

DB2 Universal Database 製品に関する情報は、 www.ibm.com/software/data/db2/udb から入手できます。

このサイトには、技術ライブラリー、資料の注文方法、クライアント・ダウンロード、ニュースグループ、フィックスパック、ニュース、および Web リソースへのリンクに関する最新情報が掲載されています。

米国以外の国で IBM に連絡する方法については、 [IBM Worldwide](http://www.ibm.com/planetwide) ページ (www.ibm.com/planetwide) にアクセスしてください。



部品番号: CT191JA

Printed in Japan

SC88-9163-00



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12

(1P) P/N: CT191JA



Spine information:



IBM® DB2™ Universal
Database

DB2 レプリケーションのガイドおよびリ
ファレンス

バージョン 8