



Informatica® PowerExchange
10.5.2

Navigator ユーザーガイド

本ソフトウェアおよびマニュアルは、使用および開示の制限を定めた個別の使用許諾契約のもとでのみ提供されています。本マニュアルのいかなる部分も、いかなる手段（電子的複製、写真複製、録音など）によっても、Informatica LLC の事前の承諾なしに複製または転載することは禁じられています。

米政府の権利プログラム、ソフトウェア、データベース、および関連文書や技術データは、米国政府の顧客に配信され、「商用コンピュータソフトウェア」または「商業技術データ」は、該当する連邦政府の取得規制と代理店固有の補足規定に基づきます。このように、使用、複製、開示、変更、および適応は、適用される政府の契約に規定されている制限およびライセンス条項に従うものとし、政府契約の条項によって適当な範囲において、FAR 52.227-19、商用コンピュータソフトウェアライセンスの追加権利を規定します。

Informatica、Informatica ロゴ、PowerCenter、および PowerExchange は、米国およびその他の国における Informatica LLC の商標または登録商標です。Informatica の商標の最新のリストは、次の Web サイトにあります <https://www.informatica.com/trademarks.html>。その他の企業名および製品名は、それぞれの企業の商標または登録商標です。

オプトアウトの権利の制限の下、本ソフトウェアによって、本ソフトウェアがデプロイされているコンピューティングおよびネットワーク環境に関する情報、デプロイメントのデータ使用状況およびシステム統計情報が米国の Informatica に自動的に送信されます。この送信は Informatica のプライバシーポリシーの下にサービスの一部と見なされ、Informatica は、<https://www.informatica.com/in/privacy-policy.html> のプライバシーポリシーに従って、この情報を使用または処理します。使用状況の収集は、Administrator ツールで無効にすることができます。

本ソフトウェアまたはドキュメンテーション（あるいはその両方）の一部は、第三者が保有する著作権の対象となります。必要な第三者の通知は、製品に含まれています。

特許については、<https://www.informatica.com/legal/patents.html> を参照してください。

本マニュアルの情報は、予告なしに変更されることがあります。このドキュメントで問題が見つかった場合は、infa_documentation@informatica.com までご報告ください。

Informatica 製品は、それらが提供される契約の条件に従って保証されます。Informatica は、商品性、特定目的への適合性、非侵害性の保証等を含めて、明示的または黙示的ないかなる種類の保証をせず、本マニュアルの情報を「現状のまま」提供するものとします。

発行日: 2022-07-07

目次

序文	13
Informatica のリソース.....	13
Informatica Network.....	13
Informatica ナレッジベース.....	13
Informatica マニュアル.....	13
Informatica 製品可用性マトリックス.....	14
Informatica Velocity.....	14
Informatica Marketplace.....	14
Informatica グローバルカスタマサポート.....	14
 第 1 章 : PowerExchange ナビゲータの概要	15
PowerExchange Navigator の概要.....	15
PowerExchange と PowerCenter との環境変数の非互換性.....	16
PowerExchange Navigator の開始.....	17
PowerExchange Navigator インタフェース.....	17
リソースエクスプローラ、リソースウィンドウ、リソースインスペクタ、およびその他のウ ィンドウ.....	19
タイトルバー.....	21
メニューおよびコマンド.....	21
ツールバー.....	28
ショートカットキー.....	29
ステータスバー.....	30
PowerExchange リソース.....	30
PowerExchange リソースを開く.....	31
リソースの検索.....	31
PowerExchange リソースを閉じる.....	33
PowerExchange リソースの削除.....	33
リソース構成.....	34
リソース構成の追加、編集、インポート.....	35
共有リソース構成の作成.....	37
現在の構成としてのリソース構成の選択.....	37
リソース構成の削除.....	38
DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルの場所.....	38
データマップ設定.....	44
データマップ設定の定義.....	44
IDMS データマップ設定の定義.....	48
パスワードの暗号化ユーティリティ.....	48
暗号化パスワードの生成.....	49

第2章: データマップ	51
データマップの概要	52
データマップ名	52
データマップ用にサポートされるデータソース	53
データファイルおよびファイルリスト処理	54
コピーブック	55
Adabas フィールド定義テーブルおよび Adabas キー	55
データマップのレコードとフィールド	55
データマップテーブルおよび列	57
配列	58
コードページ	61
データチェック	63
エクスポートされたデータマップ	64
Datacom データマップの項目	64
IDMS データマップの項目	65
IMS データマップの項目	65
ユーザーアクセス方法プログラム	65
データマップの追加	66
データマップレコードの追加	69
レコードへのフィールドの追加	70
レコードへのユーザー定義フィールドの追加	70
Datacom レコードへの項目の追加	71
Datacom レコードへのエレメントの追加	71
Datacom レコードへのキーの追加	72
Datacom レコードへのキーセグメントの追加	74
IDMS レコードへの項目の追加	75
IDMS レコードへ CALC エレメントの追加	75
IDMS レコードへのオーナーレコードとセットの追加	75
IMS レコードへの項目の追加	77
IMS レコードへの CCK フィールドの追加	78
IMS レコードへの検索フィールドの追加	79
データマップテーブルの追加	80
テーブルへの列の追加	80
レコードおよびフィールドの表示	81
テーブルとカラムの表示	83
データファイルの表示	83
データマップレコードおよびテーブルのソート	85
データマップまたはデータマップ内の項目のプロパティの表示または編集	86
データマップのプロパティの表示または編集	86
データマップレコードのプロパティの表示または編集	86
レコード内のフィールドのプロパティの表示または編集	87

Datcom レコード内のエレメントのプロパティの表示または編集.	88
Datcom レコード内のキーのプロパティの表示または編集.	88
Datcom レコード内のキーセグメントのプロパティの表示または編集.	88
IDMS レコード内の CALC エレメント用プロパティの表示または編集.	89
IDMS レコード内のオーナーレコードとセット用プロパティの表示または編集.	89
IMS レコードでの CCK フィールドのプロパティの表示または編集.	90
IMS レコードでの検索フィールドのプロパティの表示または編集.	91
データマップテーブル用プロパティの表示または編集.	91
テーブル内の列のプロパティの表示または編集.	92
データマップ内の項目のコピー、切り取り、貼り付け.	92
データマップおよびデータにアクセスするためのログオン情報の編集.	93
リモートシステムへのデータマップの送信.	95
リモートシステムに送信されるデータマップのインポート.	97
データマップと複数のデータファイルの関連付け.	99
PowerExchange ファイルリスト処理.	99
ファイルリストファイル構文規則.	99
ファイルリストファイルを作成してデータマップと関連付け.	100
ODBC およびファイルリスト処理.	100
Linux、UNIX、Windows のファイルリストファイルの例.	100
z/OS のファイルリストファイルの例.	100
データマップ、またはデータマップ内の項目の印刷.	101
データマップの印刷.	101
データマップレコードの印刷.	102
レコード内のフィールドの印刷.	103
データマップテーブルの印刷.	103
テーブル内の列の印刷.	103
データマップまたはデータマップ内の項目の削除.	104
データマップの削除.	104
リモートシステムにエクスポートされたデータマップの削除.	104
データマップレコードの削除.	105
レコードからのフィールドの削除.	105
データマップテーブルの削除.	105
テーブルからの列の削除.	106
Datcom レコードからのエレメントの削除.	106
Datcom レコードからのキーの削除.	106
Datcom レコードからのキーセグメントの削除.	107
IDMS レコードからの CALC エレメントの削除.	107
IDMS レコードからのオーナーレコードとセットの削除.	107
IMS レコードからの CCK フィールドの削除.	107
IMS レコードからの検索フィールドの削除.	108

第 3 章 : 特定のデータソースのためのデータマップ. 109

特定のデータソースのためのデータマップの概要.	109
---------------------------------	-----

Adabas データマップ.	110
Adabas データマップの追加.	111
Adabas データマップ内のテーブルへのキーの追加.	112
Datacom データマップ.	113
Datacom データマップの追加.	113
DB2 カタログデータマップ.	114
DB2 データマップの使用例.	115
DB2 カタログデータマップの追加.	115
DB2 データマップでレコードレイアウトを編集.	116
DB2 アンロードファイルデータマップ.	117
DB2 アンロードファイルデータマップの追加.	117
複数のテーブルから DB2 アンロードデータへアクセス.	118
DB2 アンロードファイルデータマップの追加 - 例.	119
IDMS データマップ.	119
IDMS データマップの追加.	120
IDMS レコードプロパティの編集.	121
IDMS レコードへの式の追加.	122
IMS データマップ.	123
IMS データマップを追加するための前提条件.	124
DL/I バッチまたは BMP を使用した IMS データへのアクセスの構成.	124
IMS ODBA を使用した IMS データへのアクセスの構成.	125
PowerExchange dbmover.cfg ファイルを Windows 上で構成.	126
IMS データマップの追加.	127
IMS データの表示.	127
IMS 階層の表示.	128
IMS レコードへの式の追加.	128
IMS レコードの RBA の取得.	128
IMS データマップ内のプロパティの編集.	129
IMS データマップをリモートシステムに送信.	129
ルックアップトランスフォーメーションを使用して、IMS データベースのデータを検索.	129
IMS データベースへのデータの書き込み.	130
テーブルの IMS オプションの表示または編集.	130
IMS データマップのテスト.	131
IMS データマップの例.	131
SEQ データマップ.	134
SEQ データマップの追加.	135
SEQ または VSAM のレコード階層の表示.	136
VSAM データマップ.	136
VSAM データマップの追加.	136
SEQ または VSAM のレコード階層の表示.	138
VSAM データセットの一括読み込みのパフォーマンス向上.	138
VSAM データセットのレコードに対する RRN または RBA を取得.	140

第 4 章 : コピーブック	142
コピーブックの概要	142
データマップでサポートされるコピーブックタイプ	143
データマップへのコピーブックのインポート	144
レコードまたはセグメントへのコピーブックのインポート	146
Adabas データマップへの FDT のインポート	148
Adabas データマップ用 FDT の表示とリフレッシュ	149
[コピーブックのインポート - ソースの詳細] ダイアログボックス	149
[Import Copybook - Field Property Details (コピーブックのインポート-フィールドプロパティの詳細)] ダイアログボックス	153
[コピーブックのインポート - ローカル詳細] ダイアログボックス	153
[コピーブックのインポート - ローカル DB2 カタログの詳細] ダイアログボックス	154
[コピーブックのインポート - リモートの詳細] ダイアログボックス	155
[コピーブックのインポート - リモート Datacom の詳細] ダイアログボックス	156
[コピーブックのインポート - リモート DB2 カタログの詳細] ダイアログボックス	158
[DB2 カラムの選択] ダイアログボックス	159
[IDMS 詳細プロパティ] ダイアログボックス	160
[コピーブックのインポート - 構成の詳細] ダイアログボックス	161
[レコード定義] ダイアログボックスと [重複レコード定義] ダイアログボックス	162
第 5 章 : 登録グループおよびキャプチャ登録	164
登録グループおよびキャプチャ登録の概要	164
登録タグ名	165
登録グループ	169
登録グループの追加	169
キャプチャ登録の追加	175
ステータスオプション	181
圧縮オプション	182
キャプチャ登録の表示	183
登録グループのプロパティの表示または編集	184
キャプチャ登録のプロパティの表示または編集	186
Datacom キャプチャ登録のエLEMENTの表示	191
IDMS キャプチャ登録のパスの表示	192
キャプチャ登録での DB2 カラムの変更	192
キャプチャ登録の削除	192
登録グループの削除	193
登録グループのすべてのキャプチャ登録の削除	194
第 6 章 : 抽出グループおよび抽出マップ	195
抽出グループおよび抽出マップの概要	195
抽出マップ名	196
抽出グループの追加	198

抽出マップの追加.	200
抽出マップの表示.	202
データ列用の変更インジケータおよび操作前の画像の追加および削除.	203
抽出処理への列の追加または抽出処理からの列の削除.	204
抽出マップと関連付けられたキャプチャ登録の変更.	205
データマップと抽出マップとの統合.	206
抽出グループのプロパティの表示または編集.	207
抽出マップのプロパティの表示.	209
抽出マップからのデータマップの作成.	210
抽出マップの削除.	211
抽出グループの削除.	212
抽出グループ内のすべての抽出マップの削除.	212
抽出定義ウィンドウ - マップタブ.	213
抽出定義ウィンドウ - 拡張タブ.	213
PowerExchange で生成された抽出マップ内のカラム.	214

第 7 章 : パーソナルメタデータ..... 222

パーソナルメタデータの概要.	222
パーソナルメタデータプロファイルの追加.	223
パーソナルメタデータプロファイル中のテーブルのメタデータの表示.	226
パーソナルメタデータプロファイル内のテーブルのプロパティの表示.	227
パーソナルメタデータプロファイル用プロパティの表示または編集.	227
パーソナルメタデータのリフレッシュ.	228
パーソナルメタデータプロファイルの削除.	228

第 8 章 : データベース行のテスト..... 229

データベース行テストの概要.	229
データベース行のテストによる出力ウィンドウの表示または非表示.	230
データベース行のテストおよび非標準コードページデータ.	230
データマップのテスト.	231
抽出マップのテスト.	232
パーソナルメタデータプロファイルのテスト.	233
データベース行のテスト内の PowerExchange リスナコマンドの発行.	234
変更データキャプチャのリスタートトークンの生成.	235
[データベース行のテスト] ダイアログボックス.	237
CAPX の [詳細パラメータ] ダイアログボックス - [全般] タブ.	246
CAPX の [詳細パラメータ] ダイアログボックス - [AS400] タブ.	248
CAPXRT の [詳細パラメータ] ダイアログボックス - [全般] タブ.	249
CAPXRT の [詳細パラメータ] ダイアログボックス - [Adabas] タブ.	253
CAPXRT の [詳細パラメータ] ダイアログボックス - [AS400] タブ.	253
CAPXRT の [詳細パラメータ] ダイアログボックス - [DB2UDB] タブ.	253
CAPXRT の [詳細パラメータ] ダイアログボックス - [MSSQL] タブ.	254
[CAPXRT 詳細パラメータ] ダイアログボックス - [Oracle] タブ.	254

DB2390IMG の [詳細パラメータ] ダイアログボックス.....	255
[詳細パラメータ] ダイアログボックス (IMS)	255
第 9 章 : PowerExchange Navigator の例.....	258
PowerExchange Navigator の例の概要.....	258
単一レコードデータマップの追加 - 例.....	259
手順 1. データマップの追加.....	259
手順 2. データマップへのレコードの追加.....	260
手順 3. データマップ内のレコードへのフィールドの追加.....	260
手順 4. データマップへのテーブルの追加.....	262
手順 5. 単一レコードデータマップのテスト.....	262
配列を含む単一レコードデータマップの追加 - 例.....	263
手順 1. データマップとデータファイルの関連付け.....	263
手順 2. データマップ内のデータのフィルタリング.....	263
手順 3. 配列内の各エレメント用の出力行の作成.....	264
手順 4. 配列内のすべてのエレメント用の単一出力行の作成.....	265
複雑なテーブルを含む複数レコードデータマップの追加 - 例.....	265
手順 1. 複数レコードデータマップと 1 つのデータファイルの関連付け.....	266
手順 2. 複数レコードデータマップ内のレコードの表示.....	266
手順 3. 複数レコードデータマップ内のレコードの階層の定義.....	266
手順 4. 複数レコードデータマップのテスト.....	267
データマップへの COBOL コピーブックのインポート - 例.....	268
手順 1. データマップの追加とデータマップへの COBOL コピーブックのインポート.....	268
手順 2. データマップレコード内のフィールドへの日付マスクの適用.....	271
手順 3. WHERE 句を使用した出力からの空のレコードのフィルタリング.....	271
手順 4. OCCURS DEPENDING ON 句内の項目に対する複数の出力行の表示.....	272
REDEFINES 文を含む COBOL コピーブックのインポート - 例.....	272
手順 1. データマップの追加とデータマップへの REDEFINES 文を含む COBOL コピーブック のインポート.....	273
手順 2. データマップ内の GROUP フィールドへの日付マスクの適用.....	274
PL/I コピーブックのインポート - 例.....	276
手順 1. データマップの追加とデータマップへの PL/I コピーブックのインポート.....	276
手順 2. データマップのテスト.....	278
配列を正規化するためのデータマップでのテーブルプロパティの定義 - 例.....	278
手順 1. 複数回出現が定義された配列を含むデータマップの追加.....	279
手順 2. ユーザー定義フィールドの追加配列の正規化の明示.....	280
手順 3. 配列を正規化するためのテーブルプロパティの定義.....	281
手順 4. データマップでの配列の正規化のテスト.....	282
PowerExchange CallProg 関数を使用したユーザーイグジットプログラムの呼び出し - 例.....	282
手順 1. データマップの追加.....	283
手順 2. ユーザーイグジットプログラムの作成およびコンパイル.....	287
手順 3. ユーザー定義フィールドの追加.....	297
手順 4. MASTER_REC テーブルのカラムのリフレッシュ.....	301

手順 5. ユーザー出口プログラムの結果のテスト.	302
-----------------------------------	-----

付録 A : ユーザー定義フィールド用 PowerExchange 関数. 303

ユーザー定義フィールド用 PowerExchange 関数の概要.	303
PowerExchange 関数.	304
CallProg.	304
Check.	305
CheckNumData.	306
Concat.	307
CopyData.	308
Fragment.	308
GenVRowKey.	310
GetCurrentFileName.	310
GetDatabaseKey.	312
GetDataFlowType.	313
GetDbKey.	314
GetDbKeyOfFirstMember.	314
GetDbKeyOfLastMember.	315
GetDbKeyOfNextMember.	315
GetDbKeyOfOwner.	315
GetDbKeyOfPriorMember.	316
GetFullDbKey.	316
GetFullDbKeyOfFirstMember.	317
GetFullDbKeyOfLastMember.	317
GetFullDbKeyOfNextMember.	318
GetFullDbKeyOfOwner.	318
GetFullDbKeyOfPriorMember.	319
GetIMSRBByLevel.	319
GetPageGroup.	320
GetPageGroupOfOwner.	320
GetPgGrpAndRdx.	321
GetPgGrpAndRdxOfOwner.	321
GetPgGrpOfOwner.	321
GetRadix.	322
GetRadixOfOwner.	322
GetSeqWithinLevel.	322
GetSeqWithinParent.	323
LengthOf.	324
LTrim.	324
RTrim.	325
SetBitA.	325
SetBitN.	326
Split.	327

Strip.	327
ToLower.	328
ToUpper.	329
CallProg Function 関数によるユーザー定義プログラムの呼び出し.	329
ユーザー定義プログラムの場所.	329
CallProg 関数で呼び出されたユーザー定義プログラムのロードエラー.	329
ユーザー定義プログラムへ渡されたパラメータリスト.	330
ユーザー定義プログラムの戻りコード.	332
ユーザー定義プログラム - 例.	333
COBOL プログラム - 例.	334
Assembler プログラム - 例.	335
RPG プログラム - 例.	336
PL/I プログラム - 例.	337
データマップのユーザー定義フィールドのテスト.	338
付録 B: ユーザーアクセス方法プログラム.	339
ユーザーアクセス方式プログラムの概要.	339
ユーザーアクセス方法プログラムへ渡されたパラメータ構造体.	341
ユーザーアクセス方式プログラムからの戻りコード.	343
ユーザーアクセス方式プログラム - 例.	344
付録 C: アプリケーショングループおよびアプリケーション.	345
アプリケーショングループおよびアプリケーションの概要.	345
アプリケーショングループの追加.	346
アプリケーショングループのプロパティの表示または編集.	348
アプリケーションの抽出開始ポイントのリセット.	350
アプリケーショングループの削除.	350
付録 D: データマップのプロパティ.	352
データマッププロパティ - ADABAS アクセス方式.	352
データマッププロパティ - DATACOM アクセス方式.	354
データマッププロパティ - DB2 アクセス方式.	355
データマッププロパティ - DB2UNLD アクセス方式.	355
[DB2 テーブルフィルタ] ダイアログボックス.	356
[DB2 テーブルフィルタ] - [使用可能なテーブル] ダイアログボックス.	358
データマップのプロパティ - DL/I バッチまたは、IMS ODBA アクセス方式.	358
データマッププロパティ - ESDS アクセス方式.	360
データマッププロパティ - IDMS アクセス方式.	361
データマッププロパティ - KSDS アクセス方式.	362
データマッププロパティ - MQSeries アクセス方式.	362
データマッププロパティ - RRDS アクセス方式.	364
データマップのプロパティ - SEQ アクセス方式.	365
データマッププロパティ - TAPE アクセス方式.	367

データマッププロパティ - USER アクセス方式.	368
データマッププロパティ - データのデフォルト値.	370
データマップのプロパティ - データチェック.	371
データマッププロパティ - レコードレベルの出口.	373
データマッププロパティ - キー.	374

付録 E: レコード、フィールドおよびテーブルのプロパティ. 375

レコードプロパティ - 名前.	375
レコードプロパティ - 名前 (Datacom)	376
レコードプロパティ - 名前 (IDMS)	377
[並べ替えエレメントの詳細] ダイアログボックス (IDMS)	379
レコードのプロパティ - インポートの詳細.	380
フィールドプロパティ - 名前.	380
フィールドプロパティ - 配列.	387
非 Adabas フィールド.	387
Adabas フィールド.	387
フィールドプロパティ - 追加のプロパティ.	388
フィールドプロパティ - フィールドチェック.	389
フィールドプロパティ - コードページ.	389
フィールドプロパティ - 可変長.	390
テーブルプロパティ - 定義.	391
テーブルプロパティ - フィルタ.	394
テーブルのプロパティ - IMS オプション.	394
テーブルプロパティ - キー.	395
検索ダイアログボックス.	396

付録 F: DTL__CAPXTIMESTAMP のタイムスタンプ. 397

データソースによって DTL__CAPXTIMESTAMP フィールドに報告されるタイムスタンプ.	397
---	-----

付録 G: COBOL コピーブックのインポート時におけるメモリマップ 作成のトレース. 399

COBOL メモリマップのトレース出力例.	400
-------------------------------	-----

索引. 402

序文

『Informatica® PowerExchange® Navigator ユーザーガイド』では、PowerExchange Navigator ユーザーインタフェースを使用してキャプチャ登録、抽出マップ、データマップ、パーソナルメタデータプロファイルを定義および管理する方法について説明します。

Informatica のリソース

Informatica は、Informatica Network やその他のオンラインポータルを通じてさまざまな製品リソースを提供しています。リソースを使用して Informatica 製品とソリューションを最大限に活用し、その他の Informatica ユーザーや各分野の専門家から知見を得ることができます。

Informatica Network

Informatica Network は、Informatica ナレッジベースや Informatica グローバルカスタマサポートなど、多くのリソースへの入口です。Informatica Network を利用するには、<https://network.informatica.com> にアクセスしてください。

Informatica Network メンバーは、次のオプションを利用できます。

- ナレッジベースで製品リソースを検索できます。
- 製品の提供情報を表示できます。
- サポートケースを作成して確認できます。
- 最寄りの Informatica ユーザーグループネットワークを検索して、他のユーザーと共同作業を行えます。

Informatica ナレッジベース

Informatica ナレッジベースを使用して、ハウツー記事、ベストプラクティス、よくある質問に対する回答など、製品リソースを見つけることができます。

ナレッジベースを検索するには、<https://search.informatica.com> にアクセスしてください。ナレッジベースに関する質問、コメント、ご意見の連絡先は、Informatica ナレッジベースチーム (KB_Feedback@informatica.com) です。

Informatica マニュアル

Informatica マニュアルポータルでは、最新および最近の製品リリースに関するドキュメントの膨大なライブラリを参照できます。マニュアルポータルを利用するには、<https://docs.informatica.com> にアクセスしてください。

製品マニュアルに関する質問、コメント、ご意見については、Informatica マニュアルチーム (infa_documentation@informatica.com) までご連絡ください。

Informatica 製品可用性マトリックス

製品可用性マトリックス (PAM) には、製品リリースでサポートされるオペレーティングシステム、データベースなどのデータソースおよびターゲットが示されています。Informatica PAM は、<https://network.informatica.com/community/informatica-network/product-availability-matrices> で参照できます。

Informatica Velocity

Informatica Velocity は、Informatica プロフェッショナルサービスが開発したヒントとベストプラクティスのコレクションで、多数のデータ管理プロジェクトから得た実体験に基づいています。Informatica Velocity には、世界中の組織と連携してデータ管理ソリューションを計画、開発、デプロイ、管理する Informatica コンサルタントによる集合知を表しています。

Informatica Velocity リソースには、<http://velocity.informatica.com> からアクセスしてください。Informatica Velocity についての質問、コメント、またはアイデアがある場合は、ips@informatica.com から Informatica プロフェッショナルサービスにお問い合わせください。

Informatica Marketplace

Informatica Marketplace は、お使いの Informatica 製品を拡張したり強化したりするソリューションを検索できるフォーラムです。Marketplace で、Informatica デベロッパーやパートナーからの多数のソリューションを活用すれば、生産性を向上したり、プロジェクトでの実装時間を短縮したりできます。Informatica Marketplace は、<https://marketplace.informatica.com> からアクセスしてください。

Informatica グローバルカスタマサポート

電話または Informatica Network を介してグローバルカスタマサポートに連絡できます。

各地域の Informatica グローバルカスタマサポートの電話番号は、Informatica Web サイト (<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>) を参照してください。

Informatica Network のオンラインサポートリソースを見つけるには、<https://network.informatica.com> にアクセスして eSupport オプションを選択します。

第 1 章

PowerExchange ナビゲータの概要

この章では、以下の項目について説明します。

- [PowerExchange Navigator の概要, 15 ページ](#)
- [PowerExchange と PowerCenter との環境変数の非互換性, 16 ページ](#)
- [PowerExchange Navigator の開始, 17 ページ](#)
- [PowerExchange Navigator インタフェース, 17 ページ](#)
- [PowerExchange リソース, 30 ページ](#)
- [リソース構成, 34 ページ](#)
- [データマップ設定, 44 ページ](#)
- [パスワードの暗号化ユーティリティ, 48 ページ](#)

PowerExchange Navigator の概要

PowerExchange Navigator は、Windows 上で稼働するグラフィカルユーザーインターフェース（GUI）です。

PowerExchange Navigator を使用して、PowerExchange オブジェクトの定義と管理を行います。これには、バルクデータ移動操作と変更データキャプチャ（Change Data Capture：CDC）処理の実行に必要な PowerExchange リソースも含まれます。

データベース行のテストを実行して、バルクデータ移動操作や変更データ抽出の実行時に、PowerExchange でデータにアクセスできることを確認します。また、**【データベース行のテスト】**ダイアログボックスを使用して、PowerExchange Listener の LISTTASK と STOPTASK コマンドの発行、およびデータソースのリスタートトークンの生成を行うことができます。リスタートトークンを使用して、PowerCenter®変更データキャプチャ（Change Data Capture: CDC）セッションの PowerExchange Client for PowerCenter（PWXPC）リスタートトークンファイルにデータを入力します。

以下の PowerExchange リソースを使用して、バルクデータ移動操作または変更データキャプチャ（Change Data Capture：CDC）処理を完了します。

- 登録グループおよびキャプチャ登録。PowerExchange で変更をキャプチャするソースデータを定義します。CDC 処理を完了するために必要です。
- 抽出グループおよび抽出マップ。キャプチャ対象として登録されたデータソースの抽出情報を定義します。CDC 処理を完了するために必要です。

- データマップ。IMS データベース、VSAM ファイル、およびフラットファイルなどの非リレーショナルデータソースにアクセスし、DB2 ソースに対してカラムレベルの処理を実行するために、データマップを定義します。非リレーショナルソースの場合、バルクデータ移動操作の完了と、キャプチャ登録用のメタデータの提供にデータマップが必要となります。

データマップでは、データのレイアウトを手動で定義できます。また、コピーブックや Adabas FDT などを入力して、データのレイアウトを定義できます。

定義できるその他の PowerExchange オブジェクトは次のとおりです。

- アプリケーショングループおよびアプリケーション。ODBC 接続を使用して変更データを抽出する場合、キャプチャ対象として登録されているデータソースの抽出情報を定義します。登録グループを定義した場合に、PowerExchange によりアプリケーショングループが生成されます。

注: PowerExchange Client for PowerCenter (PWXPC) を使用して変更データを抽出する場合、アプリケーショングループおよびアプリケーションは使用されません。この方法を推奨します。

- パーソナルメタデータプロファイル。データソースへの接続をテストするプロファイルの定義、メタデータの表示、およびリレーショナルと非リレーショナルのデータソースとターゲットのプレビューを行います。

PowerExchange と PowerCenter との環境変数の非互換性

PowerCenter® と PowerExchange が同じ Linux、UNIX、または Windows マシンにインストールされている場合、場合によっては、PATH や LD_LIBRARY_PATH 環境変数の要件が競合することがあります。これらの場合に正しく実行するには、PowerExchange と PowerCenter は別々の環境で実行する必要があります。

この要件は、PowerCenter 統合サービスまたは PowerCenter リポジトリサービスが、以下の PowerExchange コンポーネントのいずれかと同じマシンで実行されている場合に適用されます。

- PowerExchange リスナ
- PowerExchange ロgger (Linux、UNIX、Windows 用)
- PowerExchange Navigator
- createdatamaps ユーティリティ以外の任意の PowerExchange ユーティリティ

以下の表に、PowerExchange 環境および PowerCenter 環境において PATH 変数および LD_LIBRARY_PATH 変数に適用される制限を示します。

環境	PATH	LD_LIBRARY_PATH
PowerExchange	\$INFA_HOME が \$PWX_HOME より先行してはいけません。それ以外の場合は、PowerExchange リスナまたはロggerをコマンドラインから開始することはできません。	LD_LIBRARY_PATH に PowerCenter のエントリが含まれてはいけません。この要件は、PowerExchange ユーティリティはライブラリを必ず \$PWX_HOME からのみピックアップすることを保証します。
PowerCenter	\$PWX_HOME エントリが \$INFA_HOME エントリより先行してはいけません。	\$LD_LIBRARY_PATH 変数定義は必ず \$INFA_HOME と \$PWX_HOME の両方を含み、\$INFA_HOME が \$PWX_HOME より先行である必要があります。以下に例を示します。 \$INFA_HOME/server/bin:\$PWX_HOME:\$LD_LIBRARY_PATH

PowerExchange インスタンスまたは PowerCenter インスタンスの正しい環境を同じマシンに設定するには、以下のいずれかのストラテジを使用します。

- PowerExchange と PowerCenter は常に別のアカウントで起動し、各アカウントに適切な環境変数を設定します。
- PowerExchange コンポーネントを開始するたびに、`pwxsettask.sh` または `pwxsettask.bat` スクリプトを実行します。

PowerExchange Navigator の開始

[スタート] メニューで PowerExchange Navigator を開始します。PowerExchange Navigator を開始する前に PowerExchange 環境の設定が必要な場合があります。

1. PowerExchange 環境を設定する必要がある場合は、次のコマンドを実行します。

```
pwxsettask setpwxenv
```

2. [スタート] メニューから、PowerExchange Navigator を開始します。

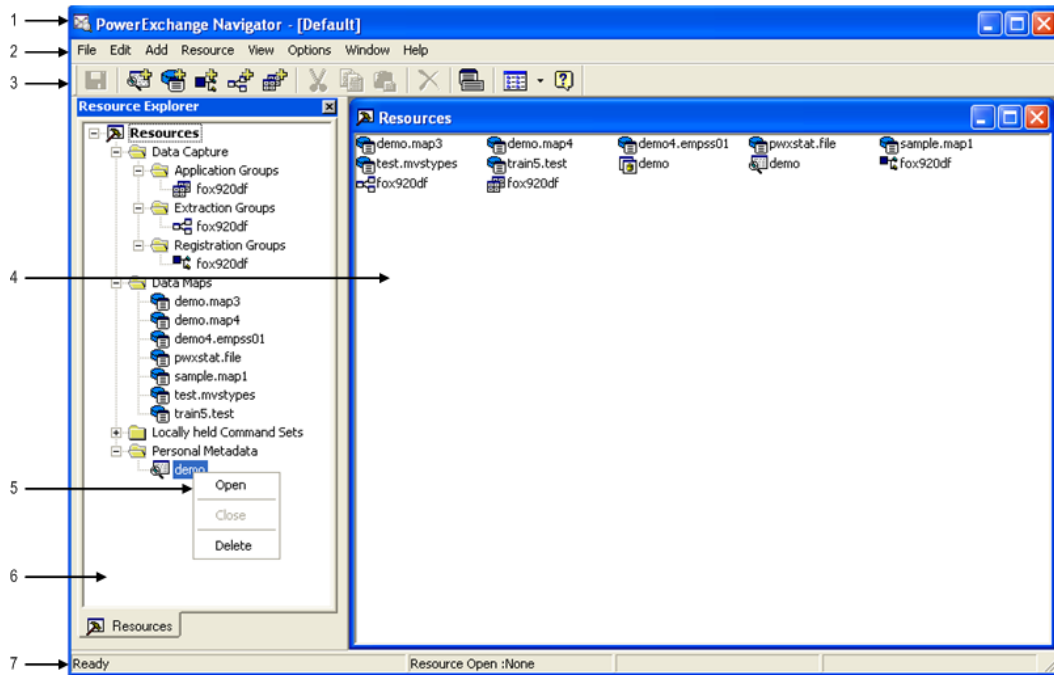
詳細については、[「PowerExchange と PowerCenter との環境変数の非互換性」 \(ページ 16\)](#)を参照してください。

PowerExchange Navigator インタフェース

PowerExchange Navigator では、PowerExchange のリソースとリソース情報が、**リソースエクスプローラ**、**[リソース]** ウィンドウ、**リソースインスペクタ**に表示されます。特定のリソースについては、他のウィンドウに表示されます。

PowerExchange Navigator のメニューバー、ツールバー、およびショートカットメニューを使用して、これらのリソースを操作します。

以下の図に、リソースエクスプローラの [リソース] タブ、[リソース] ウィンドウ、およびリソースを定義および管理するために使用するメニューとツールバーを示します。



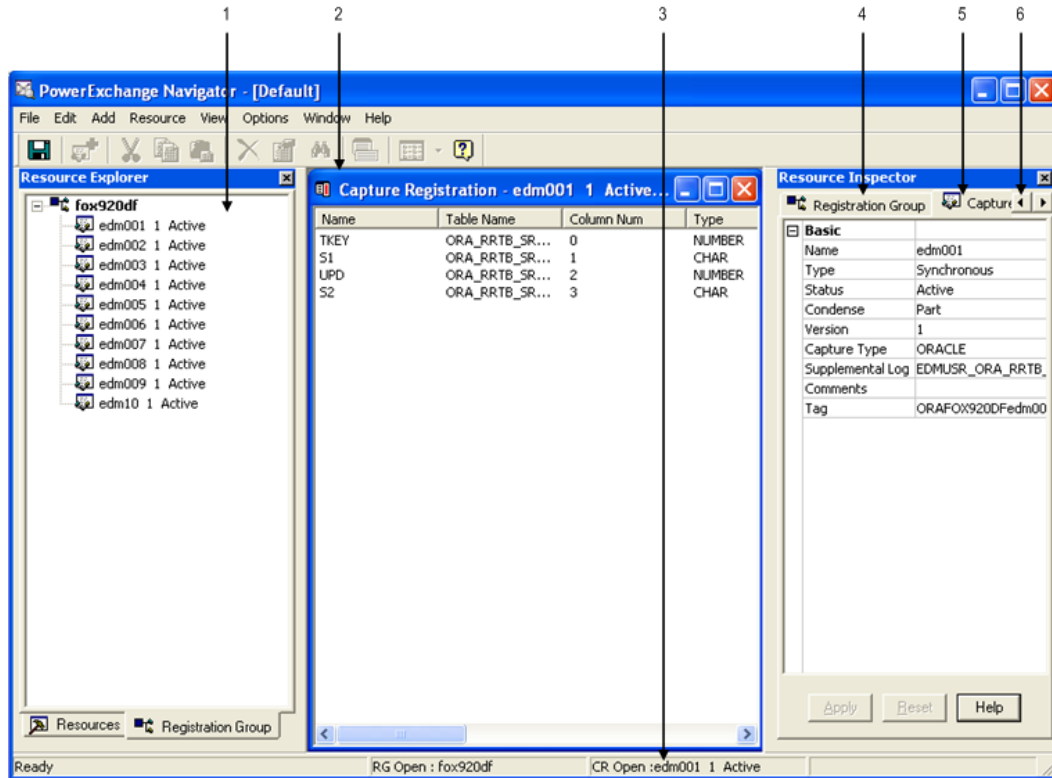
1. タイトルバー
2. メニューバー
3. ツールバー
4. [リソース] ウィンドウ
5. ショートカットメニュー
6. リソースエクスプローラの [リソース] タブ
7. ステータスバー

PowerExchange Navigator GUI には、以下のリソース固有の要素が含まれています。

- リソースエクスプローラの [登録グループ] タブ。
- [キャプチャ登録] ウィンドウ。
- オープン登録グループとキャプチャ登録に関する情報が表示されるステータスバー。

- オープン登録グループとキャプチャ登録の情報が表示される、リソースインスペクタの【登録グループ】タブおよび【キャプチャ登録】タブ。リソースインスペクタの【キャプチャ登録】および【登録グループ】タブ間を切り替えるには、切り替え矢印を使用します。

以下の画像にこれらのエレメントを示します。



- リソースエクスプローラのリソース固有のタブ
- 【キャプチャ登録】ウィンドウ
- ステータスバー
- リソースインスペクタの【登録グループ】タブ
- リソースインスペクタの【キャプチャ登録】タブ
- 切り替え矢印

リソースエクスプローラ、リソースウィンドウ、リソースインスペクタ、およびその他のウィンドウ

PowerExchange Navigator では、PowerExchange のリソースとリソース情報が、**リソースエクスプローラ**、**【リソース】** ウィンドウ、**リソースインスペクタ**に表示されます。特定のリソースについては、他のウィンドウに表示されます。

リソースエクスプローラに表示される情報は、PowerExchange リソースが開いているかどうかと、開いているリソースのタイプにより異なります。

PowerExchange Navigator を起動した場合、PowerExchange リソースを開く前に、以下のウィンドウが表示されます。

- **リソースエクスプローラ**。左側に表示されます。【リソース】タブでは、**リソースエクスプローラ**で展開と圧縮が可能なツリー構造にあるフォルダ内のオブジェクトがグループ化されることにより、PowerExchange のリソース定義をカテゴリ別に表示できます。フォルダは次のとおりです。
 - データキャプチャとそのサブフォルダ、**アプリケーショングループ**、**抽出グループ**、および**登録グループ**
 - **データマップ**
 - **パーソナルメタデータ**
- **【リソース】ウィンドウ**。右側に表示されます。以下の定義されたリソースをリストまたはアイコンビューに表示します。
 - アプリケーショングループ、登録グループ、抽出グループ
 - データマップ
 - パーソナルメタデータプロファイル

リソースエクスプローラ、または **【リソース】ウィンドウ**のいずれかから PowerExchange リソースを操作することができます。

登録グループ、抽出グループ、データマップ、またはパーソナルメタデータプロファイルを開くとき、**リソースエクスプローラ**には、グループ、データマップ、またはプロファイル内のオブジェクトを一覧表示する新しいタブが表示されます。

さらに、以下の PowerExchange リソースのいずれかを開いた場合、**リソースインスペクタ**がインタフェースの右側に表示されて、そのリソースのプロパティが表示されます。

- アプリケーショングループおよびアプリケーション
- 登録グループおよびキャプチャ登録
- 抽出グループおよび抽出マップ

登録グループのキャプチャ登録など、これらのグループの項目を開いた場合、登録とグループは同時に開かれたままの状態になります。次のように、開いている項目に固有のウィンドウが表示されます。

- **アプリケーションウィンドウ**。アプリケーションを表示します。
- **【キャプチャ登録】ウィンドウ**。キャプチャ登録により登録された列を表示します。
- **抽出定義ウィンドウ**。抽出マップを表示します。

さらに、**リソースインスペクタ**には、開いているグループおよびグループ内で開いている項目の両方のタブが表示されます。切り替え矢印を使用して、**リソースインスペクタ**のタブ間を切り替えます。ステータスバーには、オープングループと登録に関する情報が表示されます。

データマップのレコードを開いた場合、**レコードウィンドウ**にレコード内のフィールドが表示され、レコード内のフィールドを追加、変更、または削除することができます。

データマップのテーブルを開いた場合、**テーブルウィンドウ**にテーブル内の列が表示され、テーブル内の列を追加、編集、または削除することができます。

リソースエクスプローラまたはリソースインスペクタの表示または非表示

【表示】 メニューから**リソースエクスプローラ**または**リソースインスペクタ**を表示または非表示にすることができます。

デフォルトでは、**リソースエクスプローラ**と**リソースインスペクタ**は表示されます。

リソースウィンドウのビュータイプの変更

定義されたリソースを大きいアイコンや小さいアイコンとして **【リソース】** ウィンドウに表示したり、**【表示】** ツールバーボタンを使用して、変更内容の有無にかかわらずリストに表示することができます。

ウィンドウの整列

【リソース】 ウィンドウおよび **【キャプチャ登録】** ウィンドウなどのウィンドウを、**【ウィンドウ】** メニューから、重ねて、または左右に並べて整列することができます。

また、開いているウィンドウに切り替えることもできます。ウィンドウを整列することにより、**リソースエクスプローラ**または**リソースインスペクタ**の配置が影響を受けることはありません。

タイトルバー

PowerExchange Navigator のタイトルバーは、GUI の上部にあります。

タイトルバーにはアプリケーション名の PowerExchange Navigator が表示されます。また、次の内容が続けて表示される場合があります。

- リソース構成を表示または編集した場合は、角かっこに囲まれたリソース構成名
- **【リソース】** ウィンドウなどのウィンドウを最大化した場合は、角かっこに囲まれたウィンドウ名

メニューおよびコマンド

PowerExchange Navigator では、PowerExchange リソースを定義および管理し、その他のタスクを完了するためのコマンドを選択できるメニューを使用できます。

次のタイプのメニューが使用可能です。

- **メニューバーのメニュー。** **【PowerExchange Navigator】** ウィンドウの上部にあるメニューバーのメニュー名をクリックして、コマンドのリストを表示します。
- **ショートカットメニュー。** PowerExchange リソースを右クリックして、そのリソースに対して適切なコマンドを含むショートカットメニューを表示します。

メニューバーのメニュー

メニューバーの各メニューにはコマンドのリストが表示されます。

メニューバーには以下のメニューが表示されます。

- ファイル
- 編集
- 追加
- リソース
- モード
- 表示
- オプション
- ウィンドウ
- ヘルプ

これらの各メニューのコマンドについて以下で説明します。

[ファイル] メニュー

以下の表に、[ファイル] メニューのコマンドと、それらのコマンドに使用可能なショートカットキーを示します。

コマンド	説明	ショート カットキ ー
リソースを開く	PowerExchange リソースを開きます。	CTRL+O
リソースを閉じる	PowerExchange リソースを閉じます。	-
リモートノードへの送信	リモートシステムにデータマップを送信します。	-
データベース行のテスト	[データベース行のテスト] ダイアログボックスを開きます。	-
コピーブックのインポート	データマップにコピーブックをインポートします。	-
エクスポートされたデータマップ	エクスポートしたデータマップのリストを表示します。データマップはインポートすることができます。	-
データファイルの表示	データマップに関連するデータファイルを表示します。	-
保存	データマップ、抽出グループ、抽出マップ、登録グループ、またはキャプチャ登録を保存します。	CTRL+S
名前を付けて保存	データマップを別の名前で保存します。	-
印刷	データマップ、レコード、フィールド、テーブル、または列を印刷します。	-
パスワードの暗号化	クリアテキストのパスワードを暗号化します。	-
更新	データマップ、ブラウザプロファイル、アプリケーショングループ、抽出グループ、登録グループなど、ローカルに保持されたリソースはすべてリフレッシュされます。	-
終了	PowerExchange Navigator を終了します。	-

[編集] メニュー

以下の表に、[編集] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
切り取り	リソースを切り取り、クリップボードに移動します	CTRL+X
コピー	リソースをクリップボードにコピーします。	CTRL+C

コマンド	説明	ショートカットキー
貼り付け	クリップボードコンテンツのコピーを挿入ポイントに挿入します。	CTRL+V
検索	次のうち 1 つの項目を検索します。 <ul style="list-style-type: none"> - データマップのレコード内のフィールド - データマップのテーブル内の列 - パーソナルメタデータプロファイル内のテーブル - パーソナルメタデータテーブル内の列 	CTRL+F

[追加] メニュー

以下の表に、[追加] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
パーソナルメタデータ	パーソナルメタデータプロファイルを追加します。	CTRL+T
データマップ	データマップを追加します。	CTRL+D
登録グループ	登録グループを追加します。	CTRL+G
抽出グループ	抽出グループを追加します。	CTRL+E
アプリケーショングループ	アプリケーショングループを追加します。	CTRL+L
レコード	レコードを選択されたデータマップに追加します。	-
テーブル	テーブルを選択されたデータマップに追加します。	-
フィールド	フィールドを選択されたレコードに追加します。	-
カラム	カラムを選択されたテーブルに追加します。	-
キャプチャ登録	キャプチャ登録を選択された登録グループに追加します。	-
抽出定義	抽出マップを選択された抽出グループに追加します。	-

[リソース] メニュー

以下の表に、[リソース] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
プロパティ	選択されたデータマップ、レコード、フィールド、テーブル、またはカラムのプロパティを表示します。	CTRL+P
削除	PowerExchange リソースを削除します。	DELETE

コマンド	説明	ショートカットキー
ログオン	ログオン情報を入力してデータマップとデータへアクセスできるようになります。	-
FDT	Adabas データマップの場合、インポートされたフィールド定義テーブル (FDT) が [Adabas FDT] ダイアログボックスに表示されます。 注: [Adabas FDT] ダイアログボックスで、[FDT のリフレッシュ] をクリックして FDT をリフレッシュします。	-

[モード] メニュー

以下の表に、[モード] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
Char	レコードをデータファイルビューに文字形式で表示します。	-
文字/16 進	レコードがデータファイルビューに文字形式と 16 進形式で表示されます。	-
縦	レコードがデータファイルビューに文字形式と 16 進形式で縦方向に表示されます。	-
ASCII	データファイルビューでエンコードする ASCII (American Standard Code for Information Interchange) 文字を使用したレコードを表示します。	-
EBCDIC	データファイルビューでエンコードする EBCDIC (Extended Binary Coded Decimal Interchange Code) 文字を使用したレコードを表示します。	-

注: これらのコマンドに使用可能なショートカットキーはありません。

[表示] メニュー

以下の表に、[表示] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
ツールバー	ツールバーを表示または非表示にします。	-
ステータスバー	ステータスバーを表示または非表示にします。	-
リソースエクスプローラ	[リソースエクスプローラ] を表示または非表示にします。	-
アウトプット	[データベース行のテストによる出力] ウィンドウを表示または非表示にします。	-
リソースインスペクタ	[リソースエクスプローラ] を表示または非表示にします。	-

注: これらのコマンドに使用可能なショートカットキーはありません。

[オプション] メニュー

以下の表に、[オプション] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
リソース構成	PowerExchange リソースのディレクトリを構成することができます。	-
設定	データマップの設定を構成することができます。	-

注: これらのコマンドに使用可能なショートカットキーはありません。

[ウィンドウ] メニュー

以下の表に、[ウィンドウ] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
カスケード	ウィンドウを重ねて整列します。	-
上下に並べて表示	ウィンドウを上下に並べて表示します。	-
左右に並べて表示	ウィンドウを左右に並べて整列します。	-
アイコンの整列	閉じたウィンドウのアイコンを整列します。	-
ウィンドウを開く	開いているウィンドウのタイトルを表示します。ウィンドウのタイトルをクリックして、そのウィンドウに切り替えます。	-

注: これらのコマンドに使用可能なショートカットキーはありません。

[ヘルプ] メニュー

以下の表に、[ヘルプ] メニューのコマンドと、使用可能なショートカットキーを示します。

コマンド	説明	ショートカットキー
ドキュメントのコンテンツ	PowerExchange ドキュメントを表示します。	-
トピックの検索	PowerExchange Navigator のヘルプを表示します。	F1
バージョン情報	PowerExchange Navigator のバージョン情報を表示します。	-

ショートカットメニュー

ショートカットメニューは、**リソースエクスプローラ**のフォルダーまたはリソースを右クリックした場合や、**レコードウィンドウ**または**テーブルウィンドウ**内の項目を右クリックした場合に表示されます。

ショートカットメニューには、選択された項目に対して適切なコマンドが含まれます。

ショートカットメニューは次の項目に対して使用可能です。

- **リソースエクスプローラ**の**【リソース】** タブのフォルダー
 - データキャプチャと、アプリケーショングループ、抽出グループ、および登録グループなどのサブフォルダー
 - データマップ
 - パーソナルメタデータ
- **【リソースエクスプローラ】** タブ、あるいは**レコードウィンドウ**または**テーブルウィンドウ**のリソース
 - アプリケーショングループおよびアプリケーション
 - 抽出グループ、抽出マップ、および抽出マップ内の列
 - 登録グループおよびキャプチャ登録
 - データマップと、データマップ内のレコード、フィールド、テーブル、および列
 - パーソナルメタデータプロファイル、およびパーソナルメタデータプロファイルのテーブル

コマンドは、メニューバーかショートカットメニュー、またはその両方から使用可能です。以下の表に、ショートカットメニューからのみ使用可能なコマンドを示します。ほかのコマンドについては「[メニューバーのメニュー](#)」(ページ 21)を参照してください。

アプリケーショングループおよびアプリケーション

以下の表に、ショートカットメニューからのみ使用可能なアプリケーショングループとアプリケーションのコマンドを示します。

右クリック	コマンド
開いているアプリケーショングループ	新しい開始ポイントへのリセット。 アプリケーションにより追跡された抽出の抽出開始ポイントをリセットします。

キャプチャ登録

以下の表に、ショートカットメニューからのみ使用可能なキャプチャ登録のコマンドを示します。

右クリック	コマンド
開いているキャプチャ登録	カラムの変更。 選択されるカラムのキャプチャをサポートするソースのキャプチャ登録に対して、カラムを追加または削除します。 エレメントの表示。 Datacom ソースの場合は、オフセットと長さなどの Datacom レコードのエレメントを表示します。 パスの表示。 IDMS ソースの場合は、IDMS ベースレコードの上の親階層を表示します。

データマップ

以下の表に、ショートカットメニューからのみ使用可能なデータマップのコマンドを示します。

右クリック	コマンド
開いているデータマップ	項目の並べ替え。 データマップ内のレコードとテーブルをアルファベット順にソートします。
開いている IMS データマップ	IMS 階層の表示。 IMS データマップの IMS 階層を表示します。

右クリック	コマンド
【レコード】 ウィンドウ内のフィールド名	<p>前にフィールドを追加。 選択されたフィールドの前のレコードにフィールドを追加します。</p> <p>後にフィールドを追加。 選択されたフィールドの後のレコードにフィールドを追加します。</p> <p>フィールドを子として追加。 選択されたフィールドの GROUP タイプの子として、レコードにフィールドを追加します。</p> <p>最後にフィールドを追加。 レコードの最後にフィールドを追加します。</p>
【テーブル】 ウィンドウ内のカラム名	<p>前にカラムを追加。 選択されたカラムの前のテーブルにカラムを追加します。</p> <p>後にカラムを追加。 選択されたカラムの後のテーブルにカラムを追加します。</p> <p>最後にカラムを追加。 テーブルの最後にカラムを追加します。</p>
【レコード】 ウィンドウの【Expr(<i>n</i>)】タブの行	<p>前にフィールドを追加。 選択されたフィールドの前に、フィールドとそのフィールドの式を追加します。</p> <p>後にフィールドを追加。 選択されたフィールドの後に、フィールドとそのフィールドの式を追加します。</p> <p>最後にフィールドを追加。 選択されたフィールドの後に、フィールドとそのフィールドの式を追加します。</p> <p>フィールドの削除。 フィールドを削除します。</p> <p>フィールドを上に移動。 【レコード】 ウィンドウの【Expr(<i>n</i>)】タブでフィールドを上方に移動します。</p> <p>フィールドを下に移動。 【レコード】 ウィンドウの【Expr(<i>n</i>)】タブでフィールドを下方に移動します。</p> <p>フィールドの整理。 【レコード】 ウィンドウの【Expr(<i>n</i>)】タブで、完了したフィールドをリストの上部に移動します。</p> <p>種類カラムの非表示。 【レコード】 ウィンドウの【Expr(<i>n</i>)】タブで種類カラムを非表示にします。</p> <p>種類カラムの表示。 【レコード】 ウィンドウの【Expr(<i>n</i>)】タブでフィールドを表示します。</p> <p>カラムの復元。 【レコード】 ウィンドウの【Expr(<i>n</i>)】タブでフィールドをリストアします。</p>
【レコード】 ウィンドウの【Expr(<i>n</i>)】タブの【式】カラムのセルをダブルクリックします。	<p>Unicode 制御文字の挿入。 Unicode 制御文字を式に挿入します。</p> <p>右から左への読み込み順序。 式の右から左への読み込み順序を有効または無効にします。</p> <p>Unicode 制御文字を表示する。 式の Unicode 制御文字を表示または非表示にします。</p>

データマップに関連付けられたデータファイル

以下の表に、ショートカットメニューからのみ使用可能なデータファイルのコマンドを示します。

右クリック	コマンド
データファイルビュー内の選択されたレコード 注: データファイルビューを表示するには、ファイルソースのデータマップを開き、そのデータマップを右クリックして【プロパティ】を選択した後、【アクセス方式】タブで【ファイルの表示】をクリックします。	<p>スケールの表示。 データファイルのレコード内の文字数を示すルーラーを表示します。</p> <p>スケールの非表示。 ルーラーを非表示にします。</p>

抽出グループおよび抽出マップ

以下の表に、ショートカットメニューからのみ使用可能な抽出ファイルと抽出マップのコマンドを示します。

右クリック	コマンド
開いている抽出マップ	キャプチャ登録の変更。 抽出マップに関連付けられたキャプチャ登録を変更します。 変更インジケータ/操作前の画像の拡張の変更。 変更インジケータおよび抽出マップの操作前の画像カラムを追加または削除します。 データマップの作成。 抽出マップからのデータマップを作成します。
抽出マップのカラム	変更インジケータ/操作前の画像の拡張の変更。 変更インジケータ (CI) および抽出マップの操作前の画像 (BI) カラムを追加または削除します。 自動生成されたカラムの表示。 抽出マップに PowerExchange で生成されたカラムを表示します。

パーソナルメタデータ

以下の表に、ショートカットメニューからのみ使用可能なパーソナルメタデータのプロファイルとテーブルのコマンドを示します。

右クリック	コマンド
開いているパーソナルメタデータプロファイル	更新。 パーソナルメタデータプロファイルに表示されるデータを更新します。
パーソナルメタデータプロファイル内のテーブル	エクスプローラ。 テーブルのメタデータを [テーブル] ウィンドウに表示します。

ツールバー

メニューバーまたはショートカットメニューからコマンドを選択する代わりに、PowerExchange Navigator GUI の上部にあるツールバーを使用して、共有タスクを素早く開始します。

PowerExchange リソースを開いた後、ツールバーは切り替わり、リソースタイプに対して適切なツールバーボタンが表示されます。

ツールバーボタンを指定した場合、ボタンの名前がポップアップに表示され、そのボタンの説明がステータスバーに表示されます。

ツールバーの表示または非表示

ツールバーを表示または非表示にすることができます。

デフォルトでは、ツールバーは表示されます。

ツールバーを表示または非表示にする手順

- ▶ **[表示] > [ツールバー]** をクリックします。

ショートカットキー

ショートカットキーを使用して、タスクを開始します。

以下の表では、ショートカットキーについて説明します。

ショートカットキー	用途
CTRL+L	アプリケーショングループを追加します。
CTRL+D	データマップを追加します。
CTRL+E	抽出グループを追加します。
CTRL+G	登録グループを追加します。
CTRL+T	パーソナルメタデータプロファイルを追加します。
CTRL+F6	レコード または テーブル ウィンドウを閉じます。最初にウィンドウでフィールドまたは列を選択します。
ALT+F4	アクティブなダイアログボックスを閉じるか、PowerExchange Navigator を終了します。
CTRL+C	リソースをクリップボードにコピーします。
CTRL+X	リソースを切り取り、クリップボードに移動します。
DELETE	次のうち 1 つの項目を削除します。 <ul style="list-style-type: none">- PowerExchange リソース- レコードウィンドウ内の選択されたフィールド、またはテーブルウィンドウ内の列、およびその相互参照すべて
SHIFT+F1	項目に対するオンラインヘルプを表示します。
CTRL+P	選択されたデータマップ、レコード、フィールド、テーブル、または列のプロパティを表示します。
F1	PowerExchange Navigator のヘルプを表示します。
ESC	レコード内のユーザー定義フィールドのセルへの情報入力などの編集操作を、変更せずに終了します。
CTRL+F	次のうち 1 つの項目を検索します。 <ul style="list-style-type: none">- データマップのレコード内のフィールド- データマップのテーブル内の列- パーソナルメタデータプロファイル内のテーブル- パーソナルメタデータテーブル内の列
CTRL+V	クリップボードコンテンツのコピーを挿入ポイントに挿入します。
INSERT	レコード または テーブル ウィンドウの選択されたフィールドまたは列より上に、フィールドまたは列を挿入します。
CTRL+上矢印	レコード ウィンドウの [Expr(n)] タブで、ユーザー定義フィールドを上方に移動します。

ショートカットキー	用途
CTRL+下矢印	レコードウィンドウの [Expr(<i>n</i>)] タブで、ユーザー定義フィールドを上方に移動します。
矢印キー	レコードウィンドウ内のフィールドおよびユーザー定義フィールド、またはテーブルウィンドウ内の列を上下に移動します。
CTRL+O	PowerExchange リソースを開きます。
CTRL+S	データマップ、抽出グループ、抽出マップ、登録グループ、またはキャプチャ登録を保存します。

ステータスバー

PowerExchange Navigator GUI の下部にあるステータスバーでは、開いている PowerExchange リソースに関する処理ステータスと情報が表示されます。

ステータスバーには以下の情報が表示されます。

- ユーザーが指定しているメニューコマンドまたはツールバーボタンの、処理ステータスまたは説明
- 開いている PowerExchange リソースのタイプと名前
開いているリソースがない場合、次のテキストが表示されます。
Resource Open : None
- キャプチャ登録が存在するリモート場所のノードなど、開いているリソースのノード場所
- パーソナルメタデータの場合、パーソナルメタデータプロファイルのデータソースタイプ

ステータスバーの表示または非表示

ステータスバーを表示または非表示にすることができます。

デフォルトでは、ステータスバーが表示されます。

ステータスバーを表示または非表示にする手順

- ▶ **【表示】** > **【ステータスバー】** をクリックします。

PowerExchange リソース

PowerExchange Navigator を使用して、一括データ移動操作と変更データキャプチャ（Change Data Capture：CDC）処理の完了に必要な PowerExchange リソースの定義と管理を行います。

これらのリソースには以下の内容が含まれます。

- 登録グループおよびキャプチャ登録
- 抽出グループおよび抽出マップ
- データマップ、および、レコード、テーブル、その他の項目を含むデータマップの項目
- アプリケーショングループおよびアプリケーション

注: PowerExchange Client for PowerCenter (PWXPC) を使用して変更データを抽出する場合、アプリケーショングループおよびアプリケーションは使用されません。この方法を推奨します。

- パーソナルメタデータプロファイル

PowerExchange リソースを開く

PowerExchange リソースのプロパティを表示または編集するには、最初に**リソースエクスプローラ**または**リソースウィンドウ**からリソースを開く必要があります。

注: 1 つのタイプのリソースを、グループの例外およびグループの項目と共に一度に開くことができます。登録グループなどのグループを開き、そのグループのキャプチャ登録を開いた場合、登録とグループは同時に開かれます。PowerExchange リソースの他のすべてのタイプでは、リソースを開いた場合、PowerExchange では開いているその他のリソースが閉じられます。

リソースエクスプローラまたは**リソースウィンドウ**のいずれかから、データマップ、パーソナルメタデータプロファイル、アプリケーション、抽出、および登録グループを開くことができます。**リソースエクスプローラ**からアプリケーション、抽出マップ、およびキャプチャ登録を開きます。

注: PowerExchange リソースが開いている場合、リソース構成は編集できません。

PowerCenter リソースを開く手順

- ▶ **リソースエクスプローラ**で、リソースを右クリックし、**【開く】**をクリックします。

リソースエクスプローラには、リソースのタブが表示されます。ステータスバーには、開いているリソースのタイプと名前が表示されます。

リソースがデータキャプチャリソースの場合、**リソースインスペクタ**には、リソースのプロパティが表示されます。

リソースの検索

検索機能を使用して検索条件を入力し、特定のリソースを検索することができます。

パーソナルメタデータプロファイルのテーブル、データマップレコード内のフィールド、あるいはデータマップテーブル、パーソナルメタデータプロファイルテーブル、キャプチャ登録、または抽出マップの列を検索できます。

検索にはワイルドカードを含めることができます。次のデフォルトのワイルドカードがあります。

- アスタリスク (*) は、複数の一致する文字を表します。
- クエスチョンマーク (?) は 1 個の一致する文字を表します。

ヒント: 検索フィールドの名前にワイルドカード文字が含まれている場合、エスケープ文字を持つ名前のワイルドカード文字が先行されます。デフォルトでは、エスケープ文字は波形符号 (~) です。

検索条件ボックスの右にある矢印をクリックし、ワイルドカードまたはエスケープ文字を検索条件に挿入します。

【詳細】 をクリックして、ワイルドカード文字またはエスケープ文字を変更します。

データマップレコード内のフィールドの検索

データマップのレコード内のフィールドを検索することができます。

1. データマップとレコードを開きます。
2. **レコードウィンドウ**の任意の場所をクリックします。

3. **【編集】 > 【検索】** をクリックします。
4. **【レコードからのフィールドの検索】** ダイアログボックスに検索条件を入力します。
5. **【次を検索】** をクリックします。

PowerExchange Navigator により、検索条件に一致するレコードの最初のフィールドが検索されます。

データマップテーブル内の列の検索

データマップのテーブル内の列を検索することができます。

1. データマップとテーブルを開きます。
2. **テーブルウィンドウの任意の場所** をクリックします。
3. **【編集】 > 【検索】** をクリックします。
4. **【テーブルからのカラムの検索】** ダイアログボックスに検索条件を入力します。
5. **【次を検索】** をクリックします。

PowerExchange Navigator により、検索条件に一致するテーブルの最初の列が検索されます。

キャプチャ登録内の列の検索

キャプチャ登録内の列を検索することができます。

1. 登録グループとキャプチャ登録を開きます。
2. **テーブルウィンドウの任意の場所** をクリックします。
3. **【編集】 > 【検索】** をクリックします。
4. **【キャプチャ登録でのカラムの検索】** ダイアログボックスに検索条件を入力します。
5. **【次を検索】** をクリックします。

PowerExchange Navigator により、検索条件に一致するキャプチャ登録内の最初の列が検索されます。

抽出マップ内の列の検索

抽出マップ内の列を検索することができます。

1. 抽出グループと抽出マップを開きます。
2. **テーブルウィンドウの任意の場所** をクリックします。
3. **【編集】 > 【検索】** をクリックします。
4. **【抽出定義での名前検索】** ダイアログボックスに検索条件を入力します。
5. **【次を検索】** をクリックします。

PowerExchange Navigator により、検索条件に一致する抽出マップ内の最初の列が検索されます。

パーソナルメタデータプロファイル内のテーブルの検索

パーソナルメタデータプロファイル内のテーブルを検索することができます。

1. パーソナルメタデータプロファイルを開きます。
2. **【編集】 > 【検索】** をクリックします。
3. **【テーブルの検索】** ダイアログボックスに検索条件を入力します。

4. **【次を検索】** をクリックします。

PowerExchange Navigator により、検索条件に一致するプロファイル内の最初のテーブルが検索されます。

パーソナルメタデータプロファイル内のテーブル内のカラムの検索

パーソナルメタデータプロファイルのテーブル内の列を検索することができます。

1. パーソナルメタデータプロファイルを開きます。
2. 列を検索するテーブルを開きます。
3. **テーブル**ウィンドウの任意の場所をクリックします。
4. **【編集】** > **【検索】** をクリックします。
5. **【テーブルからのカラムの検索】** ダイアログボックスに検索条件を入力します。
6. **【次を検索】** をクリックします。

PowerExchange Navigator により、検索条件に一致するテーブルの最初の列が検索されます。

PowerExchange リソースを閉じる

リソースの削除またはリソース構成の表示や編集を行うには、開いている PowerExchange リソースを閉じる必要があります。

注: PowerExchange リソースが開いている状態で別のリソースを開いた場合、最初のリソースがデータキャプチャグループで、2 番目のリソースがそのグループの項目ではない限り、PowerExchange では最初のリソースが閉じられます。例えば、登録グループを開き、キャプチャ登録を開いた場合、両方のリソースが同時に開かれた状態になります。

PowerCenter リソースを閉じる手順

- ▶ **リソースエクスプローラ**で、リソースを右クリックし、**【閉じる】** をクリックします。

PowerExchange リソースの削除

データマップから PowerExchange のリソースと項目を削除することができます。

以下の PowerExchange リソースを削除することができます。

- キャプチャ登録
- 抽出マップ
- アプリケーショングループ、登録グループ、抽出グループ
- データマップ
- エクスポートされたデータマップ
- パーソナルメタデータプロファイル

データマップから以下の項目を削除することもできます。

- レコード
- レコードからのフィールド
- テーブル
- テーブルからの列

以下のデータソースのデータマップの場合、レコードから以下の項目を削除することもできます。

- Datacom。エレメント、キー、キーセグメントです。
- IDMS。CALC エレメントおよびオーナーレコードとセットです。
- IMS。CCK 検索および検索フィールドです。

データマップ、パーソナルメタデータプロファイル、キャプチャ登録、抽出マップ、またはグループを削除する前に、開いているリソースを閉じる必要があります。

あるタイプのリソースを削除することにより、他のリソースが影響を受ける可能性があります。例えば、フィールドまたはレコードを削除した場合は、PowerExchange により、そのフィールドまたはレコードへの参照がすべて削除されます。キャプチャ登録を削除した場合は、キャプチャ登録に関連付けられた抽出マップを使用して変更を抽出することができなくなります。

リソース構成

PowerExchange Navigator のリソース構成では、データマップの場所、パーソナルメタデータプロファイル、およびローカルに保存されたキャプチャ登録と抽出マップが定義されます。リソース構成では、必要に応じて DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルの場所も定義します。

PowerExchange は、マシンにインストールされた PowerExchange バージョンごとに、一連のユーザー定義リソース構成を Windows レジストリで保持します。

PowerExchange のインストールまたはアップグレード後に初めて PowerExchange Navigator を起動すると、**[リソース構成]** ダイアログボックスが表示されます。**[OK]** をクリックして**デフォルト**という名前のリソース構成を作成し、以下のデフォルトディレクトリに PowerExchange リソースを格納します。

installation_directory/Examples

リソース構成を追加または編集することができます。複数のリソース構成を定義する場合、現在の構成として特定のリソース構成を選択します。現在の構成以外のすべての構成を削除することができます。また、以前の PowerExchange バージョンからリソース構成をインポートすることもできます。

注意: PowerExchange の新しいバージョンから以前のバージョンにリソース構成をインポートしないでください。新しいバージョンで作成または更新されたリソースファイルは、以前のバージョンでは機能しないことがあります。

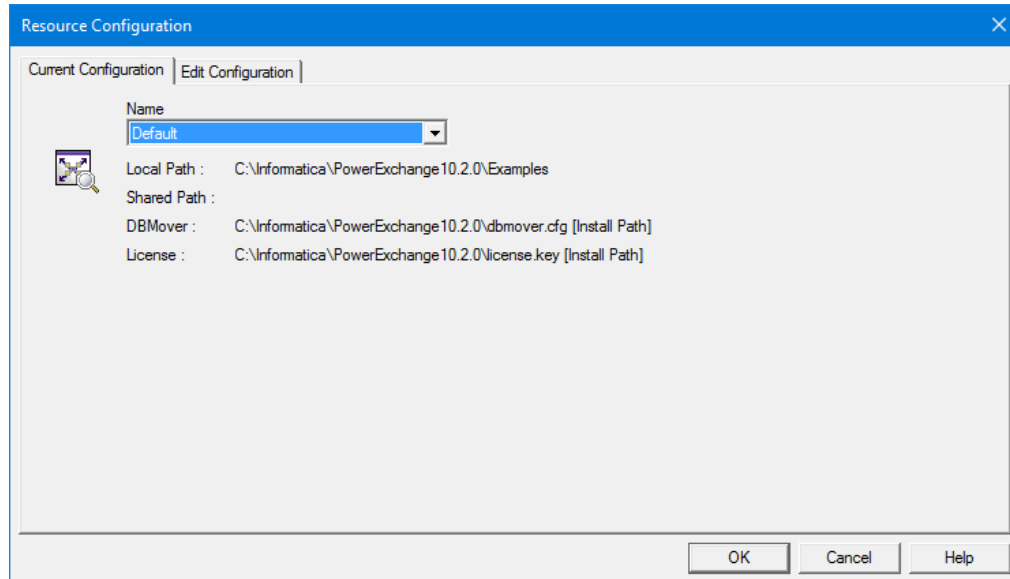
リソース構成の追加、編集、インポート

PowerExchange Navigator 内のリソース構成を追加または編集することができます。また、以前の PowerExchange バージョンからリソース構成をインポートすることもできます。

以前に PowerExchange Navigator をインストールし、そのインストールのレジストリエントリが存在する場合、以下の手順を実行してリソース構成を追加、編集、またはインポートします。

1. **【オプション】 > 【リソース構成】** をクリックします。

【リソース構成】 ダイアログボックスが表示されます。



2. **【構成の編集】** タブをクリックします。

3. 次のうち 1 つのアクションを実行します。

- リソース構成を追加するには、**【追加】** をクリックします。

【構成の追加】 ダイアログボックスが表示されます。

- リソース構成を編集するには、**【使用可能な構成】** リストから構成を選択し、**【編集】** をクリックします。

【構成の編集】 ダイアログボックスが表示されます。

- 別の PowerExchange バージョンからリソース構成をインポートするには、**【インポート】** をクリックします。

【構成のインポート】 ダイアログボックスが表示されます。

4. リソース構成の追加または編集を選択した場合、リソース構成に関する情報を **【構成の編集】** ダイアログボックスまたは **【構成の追加】** ダイアログボックスに入力します。

以下の表に、入力フィールドを示します。

フィールド	説明
名前	リソース構成のユーザー定義名です。
ローカルパス	ローカルのリソース構成では、データマップと他の PowerExchange リソースのディレクトリ場所を入力するか、[参照] ボタンをクリックしてディレクトリを参照します。
共有パス	共有のリソース構成では、共有データマップのディレクトリ場所を入力するか、[参照] ボタンをクリックしてディレクトリを参照します。 [共有パス] の値は [ローカルパス] の値と異なる必要があります。二重バックスラッシュ (\\) は PowerExchange のパス名の有効な文字列ではないため、最初にネットワークドライブを共有パスにマッピングする必要があります。
DBMOVER	必要に応じて、DBMOVER 構成ファイルのパスとファイル名を入力します。 詳細については、 「DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルの場所」 (ページ 38) を参照してください。
ライセンス	必要に応じて、PowerExchange ライセンスキーファイルのパスとファイル名を入力します。 詳細については、 「DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルの場所」 (ページ 38) を参照してください。
現在の構成として設定する	[構成の追加] ダイアログボックスで、リソース構成を現在の構成として設定します。 デフォルトでは、このオプションは選択されています。

- リソース構成をインポートするように選択した場合は、**[構成のインポート]** ダイアログボックスで、インポートするリソース構成または設定を選択します。

以下の表に、入力フィールドを示します。

フィールド	説明
バージョン	現在の PowerExchange バージョンにコピーするリソース構成の PowerExchange バージョンを選択します。
構成	コピーする構成を選択します。
選択した項目で使用可能な構成リストを置換する	このオプションを選択して、使用可能なリソース構成の一覧を選択した項目でインポートするように置換します。

- [OK]** をクリックします。
- [OK]** をクリックします。

GUI のタイトルバーにはアクティブなリソース構成が表示されます。例えば、デフォルトのリソース構成を編集する場合、次のように、タイトルバーにアプリケーション名とリソース構成が表示されます。

Informatica PowerExchange Navigator - [Default]

共有リソース構成の作成

共有ディレクトリにデータマップを保存することができます。複数のユーザーが共有データマップにアクセスできますが、データマップは同時更新から保護されます。

共有データマップにアクセスする最初のユーザーには、フルアクセスがあります。後続の同時ユーザーは、以下の操作を完了できません。

- データマップの名前の変更、編集、または削除。
- エクスポートされたデータマップのインポート、リモートのノードへのデータマップの送信。
- データマップのテスト。

すでに別のユーザーが開いている共有データマップを開こうとする場合は、警告メッセージが表示されます。

1. **【この PC】** を右クリックし、**【ネットワークドライブのマッピング】** をクリックします。
2. **【ネットワークドライブのマッピング】** ダイアログボックスに、以下の情報を入力します。
 - **【ドライブ】** リストから、接続用に使用可能なドライブ文字を選択します。
 - **【フォルダー】** ボックスで、共有データマップを含むディレクトリを入力するか参照します。
 - **【サインイン時に再接続】** を選択します。
3. **【完了】** をクリックします。
4. PowerExchange Navigator で、**【オプション】** > **【リソース構成】** をクリックします。
5. **【リソース構成】** ダイアログボックスで、**【構成の編集】** タブをクリックします。
6. 次のうち 1 つのアクションを実行します。
 - リソース構成を追加するには、**【追加】** をクリックします。
 - リソース構成を編集するには、**【使用可能な構成】** リストから構成を選択し、**【編集】** をクリックします。
7. **【構成の追加】** または **【構成の編集】** ダイアログボックスで、以下の情報を入力します。
 - **【共有パス】** フィールドで、共有リソース用にマッピングしたネットワークドライブを参照します。共有パスとローカスパスの両方を指定した場合は、これらのパスが異なることを確認します。同一の場合は、警告メッセージが表示されます。
 - 新しいリソース構成で、**【現在の構成として設定する】** オプションを選択します。
8. **【OK】** をクリックします。

PowerExchange Navigator のタイトルバーに、リソース構成の名前が表示され、その構成が共有されることが示されます。以下に例を示します。

Informatica PowerExchange Navigator - [Default (Shared)]

現在の構成としてのリソース構成の選択

複数のリソース構成を定義する場合、現在の構成として使用する 1 つを選択します。

現在の構成は別の構成が選択されるまで有効となります。

デフォルトでは、リソース構成を追加した場合、その構成が現在の構成になります。

1. **【オプション】** > **【リソース構成】** をクリックします。
2. **【リソース構成】** ダイアログボックスで、**【現在の構成】** タブをクリックします。
3. **【名前】** リストで、現在のリソース構成として使用するリソース構成を選択します。
4. **【OK】** をクリックします。

PowerExchange Navigator のタイトルバーに、アクティブなリソース構成が表示されます。

リソース構成の削除

現在のリソース構成以外のリソース構成を削除することができます。

1. **【オプション】** > **【リソース構成】** をクリックします。
2. **【リソース構成】** ダイアログボックスで、**【構成の編集】** タブをクリックします。
3. **【使用可能な構成】** リストで、構成を選択して**【削除】** をクリックします。

注: **【削除】** ボタンは現在のリソース構成には使用できません。

DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルの場所

デフォルトでは、PowerExchange Navigator のインストールディレクトリで DBMOVER 構成ファイルと PowerExchange ライセンスキーファイルを使用します。PowerExchange Navigator の環境変数、コマンドライン、または **【リソース構成】** ダイアログボックスを使用して、異なるパスとファイル名を定義できます。

【リソース構成】 ダイアログボックスの **【現在の構成】** タブに、DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルのパスとファイル名が表示されます。各パスとファイル名の後に表示されるカッコ内の文字列は、Navigator でパスとファイル名を決定するために使用するソースを示します。

以下の表に、Navigator が DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルを検索するために使用するソースを優先順位の高い順に示します。

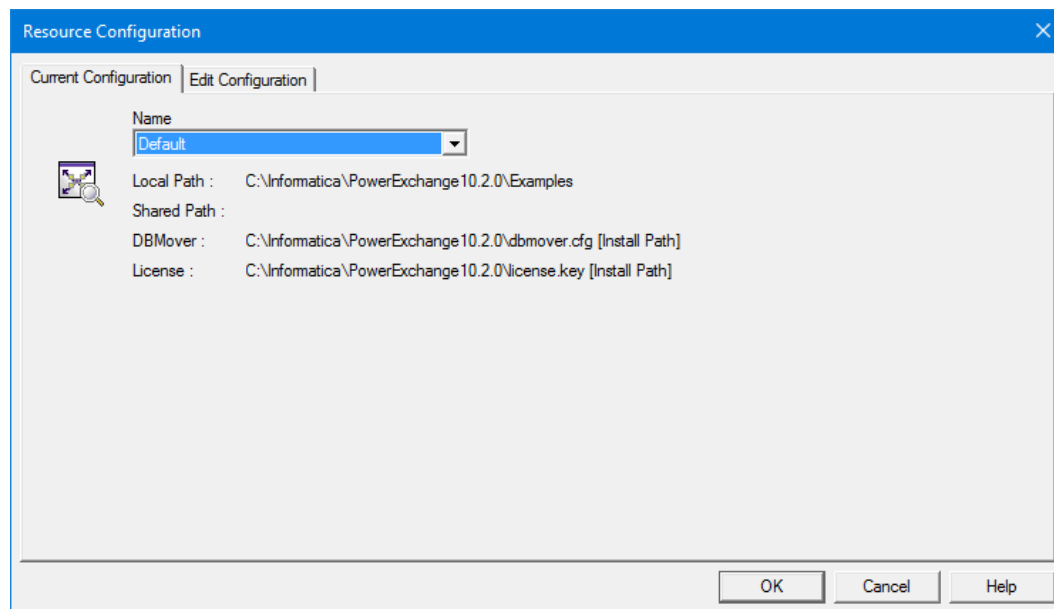
ソース	DBMOVER ソースでの GUI の指定	ライセンスキーソースでの GUI の指定	説明
システムデフォルト	[インストール先のパス]	[インストール先のパス]	オーバーライドなしを指定した場合、Navigator は PowerExchange ルートインストールディレクトリにある.cfg ファイルおよび license.key ファイルをロードします。これは、Navigator の実行可能ファイル dtlui.exe を含むパスです。
環境変数	[\$PWX_CONFIG]	[\$PWX_LICENSE]	Navigator を開始する前に、以下の環境変数を定義できます。 <ul style="list-style-type: none">- PWX_CONFIG。dbmover.cfg へのフルパスを入力します。- PWX_LICENSE。license.key へのフルパスを入力します。
コマンドライン引数	[コマンドライン]	[コマンドライン]	コマンドラインまたは Windows ショートカットから Navigator を起動する場合、次の引数を指定できます。 <ul style="list-style-type: none">- config。DBMOVER 構成ファイルへのフルパスを入力します。- license。ライセンスキーファイルへのフルパスを入力します。
PowerExchange Navigator のオーバーライド	[NAVIGATOR]	[NAVIGATOR]	特定のリソース構成名の 【リソース構成】 ダイアログボックスで、Navigator のオーバーライドを指定できます。

デフォルトの場所

DBMOVER 構成ファイルと PowerExchange ライセンスキーファイルの場所にオーバーライドを指定しない場合は、PowerExchange Navigator の PowerExchange ルートインストールディレクトリにある DBMOVER ファイルおよび license.key ファイルがロードされます。

【リソース構成】 ダイアログボックスの **【現在の構成】** タブに、DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルのパスとファイル名が表示されます。Navigator がインストールディレクトリからこれらのファイルをロードすると、各パスとファイル名の後に **[Install Path]** という文字列が表示されます。

以下の図に、Navigator がデフォルトの場所からファイルをロードするときの **【リソース構成】** ダイアログボックスを示します。



環境変数のオーバーライド

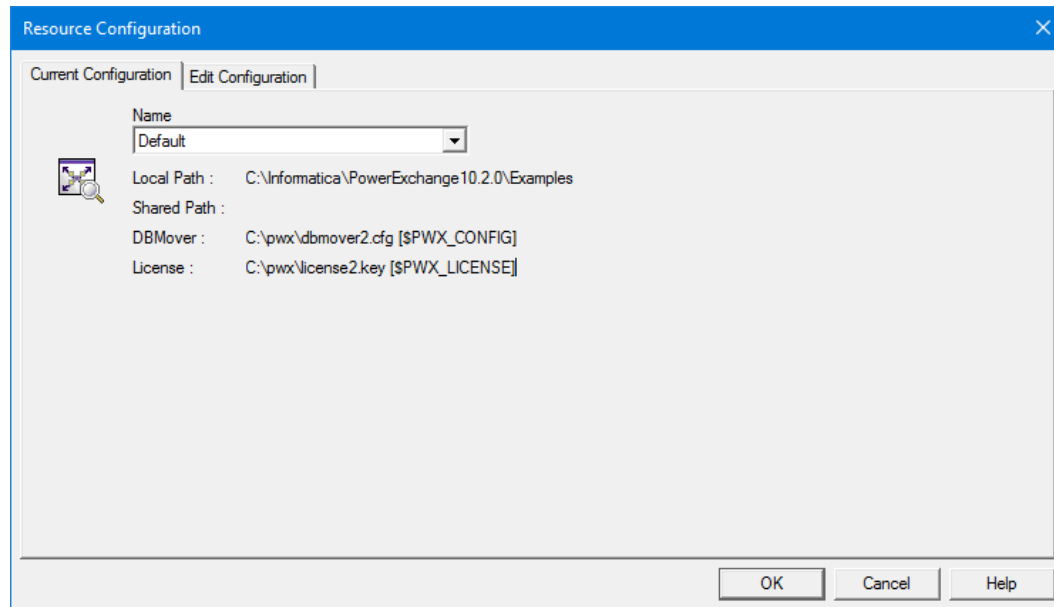
PowerExchange Navigator を起動する前に、環境変数を定義して DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルの場所を指定できます。

以下の環境変数を定義できます。

- **PWX_CONFIG**. DBMOVER 構成ファイルへのフルパスを入力します。以下に例を示します。
PWX_CONFIG=C:\pwx\dbmover2.cfg
- **PWX_LICENSE**. PowerExchange ライセンスキーファイルへのフルパスを入力します。以下に例を示します。
PWX_LICENSE=C:\pwx\license2.key

【リソース構成】 ダイアログボックスの **【現在の構成】** タブに、DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルのパスとファイル名が表示されます。Navigator が環境変数のオーバーライドを使用すると、各パスの後に次の文字列が表示されます。 **[\$PWX_CONFIG]**、**[\$PWX_LICENSE]**。

以下の図に、PowerExchange Navigator で環境変数のオーバーライドを使用する場合の、[リソース構成] ダイアログボックスの [現在の設定] タブを示します。



コマンドラインのオーバーライド

コマンドラインから DBMOVER 構成ファイルと PowerExchange ライセンスキーファイルの場所を指定するには、config および license コマンドライン引数を使用します。

以下のように、コマンドラインで config および license 引数を入力します。

- config。DBMOVER 構成ファイルへのフルパスを入力します。以下に例を示します。
`dtlui.exe config=C:\pwx\dbmover1.cfg`
- license。ライセンスキーファイルへのフルパスを入力します。以下に例を示します。
`dtlui.exe license=C:\pwx\license1.key`

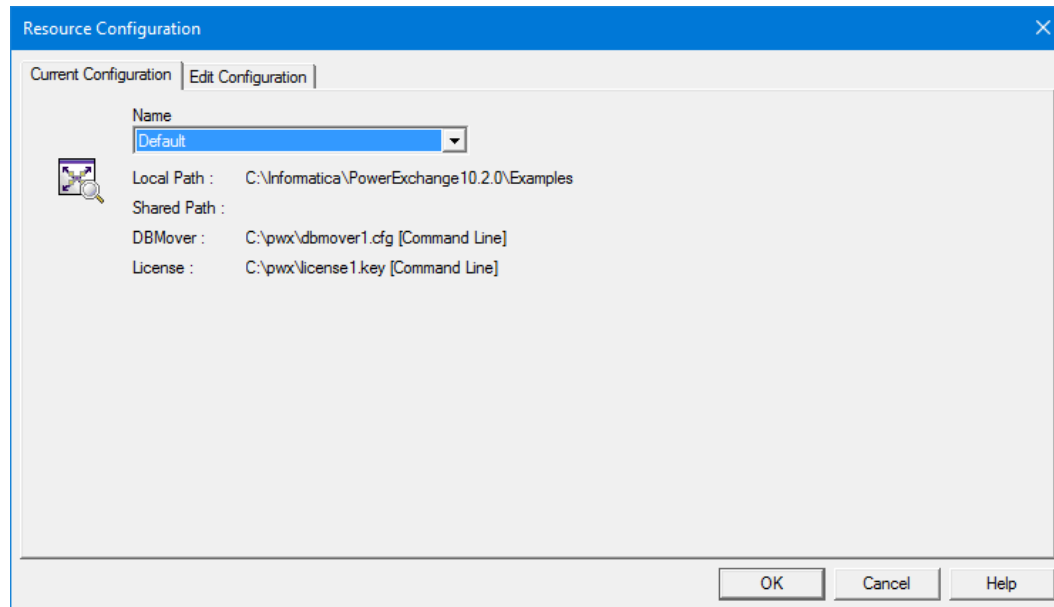
両方の引数を一緒に指定できます。以下に例を示します。

```
dtlui.exe config=C:\pwx\dbmover1.cfg license=C:\pwx\license1.key
```

また、PowerExchange Navigator を起動するための Windows ショートカットタブの [ターゲット] プロパティに、config および license 引数を含めることもできます。詳細については、[「ターゲットプロパティでのコマンドラインのオーバーライドを定義」 \(ページ 41\)](#)を参照してください。

[リソース構成] ダイアログボックスの [現在の構成] タブに、DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルのパスとファイル名が表示されます。Navigator でコマンドラインのオーバーライドを使用すると、各パスとファイル名の後に **[Command Line]** という文字列が表示されます。

Navigator でコマンドラインのオーバーライドを使用すると、[リソース構成] ダイアログボックスの [現在の構成] タブに以下の図が表示されます。

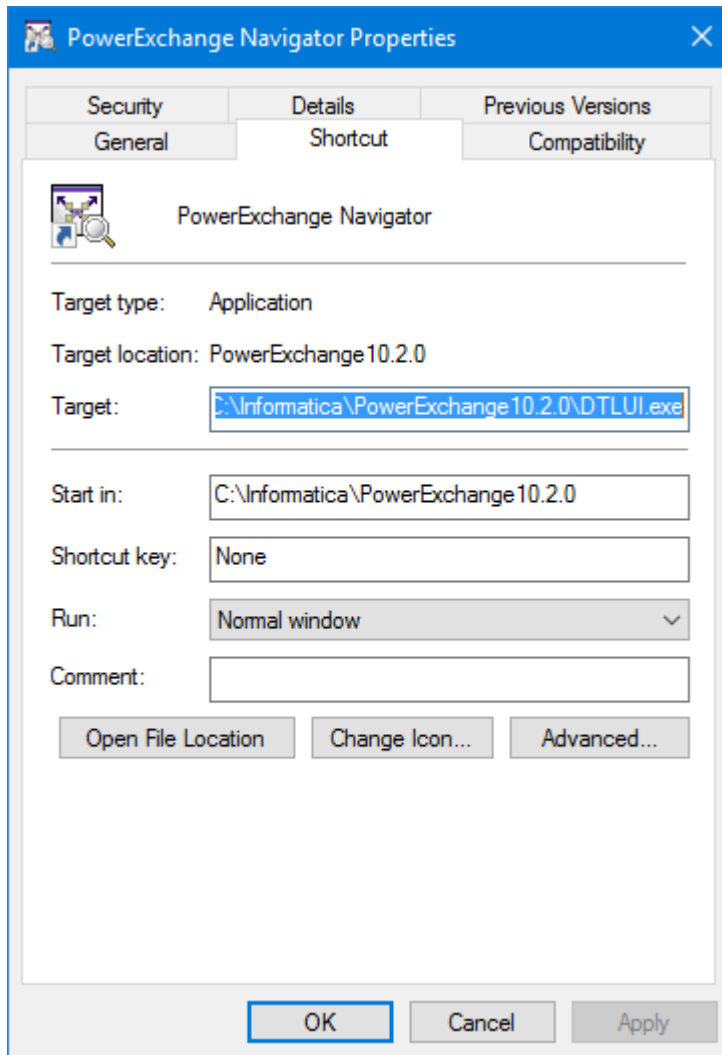


ターゲットプロパティでのコマンドラインのオーバーライドを定義

PowerExchange Navigator の起動時に、Windows ショートカットタブの [ターゲット] プロパティに config 引数と license 引数を追加することができます。

1. **[PowerExchange Navigator のプロパティ]** ダイアログボックスを開くには、次のいずれかの操作を行います。
 - Windows10 では、[スタート] メニューで **PowerExchange Navigator** ショートカットを右クリックし、[詳細] > [ファイルの場所を開く] を選択します。次に、[PowerExchange Navigator] を右クリックし、[プロパティ] を選択します。
 - Windows10 以前のバージョンでは、[スタート] メニューで **[PowerExchange Navigator]** ショートカットを右クリックし、[プロパティ] を選択します。

[PowerExchange Navigatorのプロパティ] ダイアログボックスが表示されます。



2. ダイアログボックスの【ターゲット】フィールドに表示されるコマンドラインの末尾に config または license 引数を追加します。
3. 【OK】をクリックします。

PowerExchange Navigator のオーバーライド

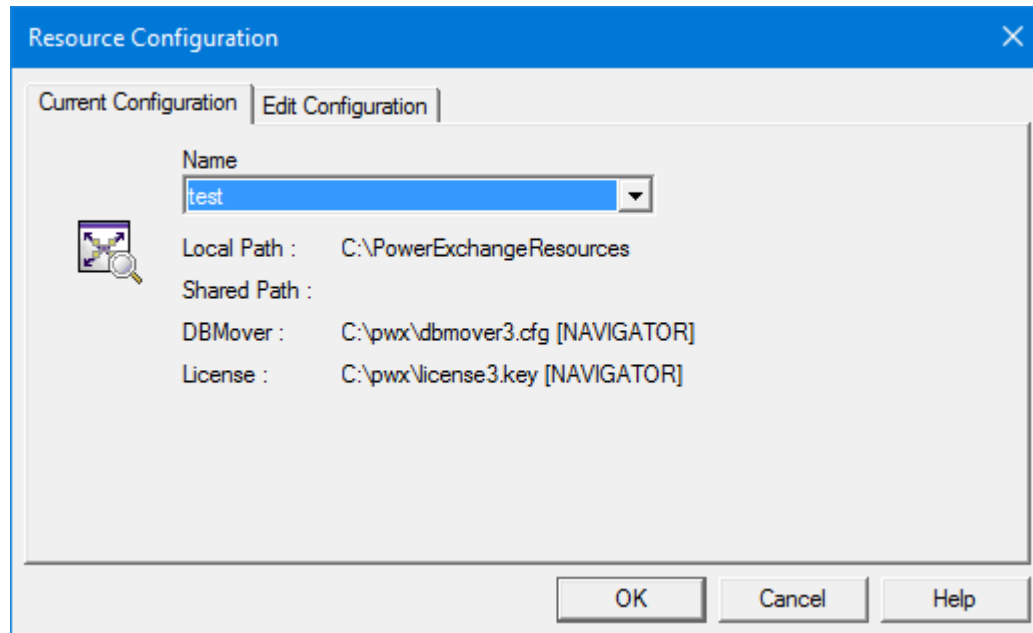
リソース構成を追加または編集する場合は、DBMOVER 構成ファイルと PowerExchange ライセンスキーファイルの場所を指定できます。

【構成の追加】 または 【構成の編集】 ダイアログボックスで、以下の情報を入力します。

- 【DBMover】 で、DBMOVER 構成ファイルへのパスを入力します。
- 【ライセンス】 で、PowerExchange ライセンスキーファイルへのパスを入力します。

【リソース構成】 ダイアログボックスの【現在の構成】タブに、DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルのパスとファイル名が表示されます。Navigator で Navigator のオーバーライドを使用すると、各パスとファイル名の後に [NAVIGATOR] という文字列が表示されます。

次の図に、Navigator で Navigator のオーバーライドを使用する場合の、[リソース構成] ダイアログボックスの [現在の構成] タブを示します。



関連項目：

- [「リソース構成の追加、編集、インポート」 \(ページ 35\)](#)

DBMOVER 構成ファイルおよび PowerExchange ライセンスキーファイルのエラー処理

PowerExchange Navigator では、無効な DBMOVER 構成ファイルまたは PowerExchange ライセンスキーファイルによって起動時にエラーが発生する可能性があります。

この状況が発生すると、Navigator にエラーメッセージが表示され、[リソース構成] ダイアログボックスが表示されます。このダイアログボックスに Navigator のオーバーライドを入力して、DBMOVER 構成ファイルまたは PowerExchange ライセンスキーファイルのパスをオーバーライドできます。

例: DBMOVER 構成ファイルが存在しない

予期した場所に DBMOVER 構成ファイルがない場合、PowerExchange Navigator には次のようなエラーメッセージが表示されます。

```
PWX-00268 DBAPI Error. DB_OPEN failed for file C:\pwx\dbmover.cfg.  
PWX-07406 Permanent error set by Open call on file "C:\pwx\dbmover.cfg" because API  
w fopen() failed. Mode rb.
```

例: DBMOVER 構成ファイルにエラーが含まれている

DBMOVER 構成ファイルにエラーが含まれている場合、PowerExchange Navigator に次のようなエラーメッセージが表示されることがあります。

```
PWX-00961 Input error. File "##DTLCFG". Statement 1. Line 19.  
PWX-00968 Error "NODEa=" sublist limit exceeded. Position 1.
```

例: PowerExchange ライセンスキーファイルが存在しない

予期した場所にライセンスキーファイルがない場合、PowerExchange Navigator には次のようなエラーメッセージが表示されます。

```
PWX-00113 KEY Error - failed to open license key file.  
PWX-00268 DBAPI Error. DB_OPEN failed for file C:\pwx\license.key.  
PWX-23532File name was converted from ANSI to Unicode.  
PWX-00150DBAPI SEQ Filename "C:\pwx\license.key" Open error reason 2 <No such file or directory>.  
PWX-07406 Permanent error set by Open call on file "C:\pwx\license.key" because API  
wfopen() failed. Mode rb.
```

例: PowerExchange ライセンスキーの有効期限が切れている

PowerExchange ライセンスキーの有効期限が切れている場合、Navigator には次のようなエラーメッセージが表示されます。

```
PWX-00115 KEY Error - license has expired. Key=2016-01-31, current=2017-03-02.  
PWX-00115 KEY Error - license has expired. Key=2016-01-31, current=2017-03-02.
```

例: PowerExchange ライセンスキーファイルが不正である

PowerExchange ライセンスキーが不正な場合、Navigator には次のようなエラーメッセージが表示されます。

```
PWX-00121 KEY Error - Not licensed for navigator feature.  
PWX-00111KEY Error - key length must be 44 or 64 license_key, len=63
```

データマップ設定

データマップの作成とテストの設定を定義することができます。

特に指定のない限り、すべてのタイプのデータマップに対して次の設定を定義することができます。

- データマップを追加する場合に、**コピーブックのインポートウィザード**を呼び出すかどうか
- Adabas データマップを追加する場合に、**Adabas FDT インポートウィザード**を呼び出すかどうか
- データマップ上でデータベース行のテストを実行する場合に、テーブル用に生成される SQL で使用される命名規則
- レコードへのフィールドの追加、またはテーブルへの列の追加でのデフォルトの挿入ポイント
- **レコードウィンドウ**、**テーブルウィンドウ**、および**データベース行のテストによる出力ウィンドウ**でのデータの設定の表示

IDMS データマップでは、Distributed Database System (DDS) のディクショナリ情報を定義することができます。

データマップ設定の定義

すべてのタイプのデータマップの追加とテストの設定を定義することができます。

1. **【オプション】** > **【設定】** をクリックします。
2. **【設定】** ダイアログボックスで **【データマップ】** タブをクリックします。

以下の表に、このタブのオプションを示します。

オプション	説明
新しいデータマップのレコード定義をインポートする	<p>データマップを追加するときにコピーブックのインポートウィザードを実行するには、このオプションを選択します。</p> <p>コピーブックのインポートウィザードを実行しない場合は、このオプションの選択を取り消します。または、データマップの追加ウィザードの【名前】ダイアログボックスで、【レコード定義のインポート】オプションの選択を取り消します。</p> <p>デフォルトは選択されています。</p>
キーフィールド/FDT のインポート (Adabas のみ)	<p>データマップを追加するときに Adabas FDT インポートウィザードを実行するには、このオプションを選択します。</p> <p>Adabas FDT インポートウィザードを実行しない場合は、このオプションの選択を取り消します。または、データマップの追加ウィザードの【名前】ダイアログボックスで、【キーフィールド/FDT のインポート】オプションの選択を取り消します。</p> <p>デフォルトは選択されています。</p>
2 層から成る名前を使用する	<p>データマップのデータベース行のテストを実行する場合に、PowerExchange により生成される SQL でテーブルを特定するために使用する命名規則を示します。</p> <p>2 階層式の命名規則を使用するには、このオプションを選択します。2 階層式の名前には、テーブル名が付加されたスキーマ名とマップ名が次の形式で含まれます:</p> <pre>select * from schema.map_table</pre> <p>3 階層式の命名規則を使用するには、このオプションの選択を取り消します。3 階層式の名前には、スキーマ名、マップ名、テーブル名が次の形式で含まれます:</p> <pre>select * from schema.map.table</pre> <p>注: このオプションにより、NRDB および NRDB2 データソースタイプで使用される命名規則が影響を受けることはありません。</p> <ul style="list-style-type: none"> - NRDB では、3 階層式の命名規則を使用します。 - NRDB2 では、2 階層式の命名規則を使用します。
ネットワーク操作のタイムアウト	<p>PowerExchange により接続が切断されてタイムアウトのエラーメッセージが発行される前に、ネットワークの送受信操作中に経過する秒数を入力します。</p> <p>デフォルトは 180 です。</p>
接続プールサイズ/保持	<p>接続プールを使用する場合、最初のテキストボックスに、開いた状態にする PowerExchange Listener への最大接続数を入力します。PowerExchange Navigator は、処理を迅速化するために、この数まで開いた状態の接続を使用します。ゼロより大きい値を入力する場合は、PowerExchange リスナマシン上にある DBMOVER ファイルの MAXTASKS の値が、リスナタスクのプール内の最大接続数に対応できる十分な値であることを確認します。</p> <p>隣接する 2 番目のテキストボックスに、接続プールを閉じるまでに非アクティブな接続を保持する秒数を入力します。</p>

オプション	説明
挿入ポイント	<p>次のうち 1 つのオプションを選択して、レコードへのフィールドの追加、またはテーブルへの列の追加での挿入ポイントを指定します。</p> <ul style="list-style-type: none"> - 現在の項目の前 - 現在の項目の後 - 最後 <p>指定された挿入ポイントは、次のうち 1 つのアクションを完了する場合に使用されます。</p> <ul style="list-style-type: none"> - レコードにフィールドを追加するには、[追加] > [フィールド] をクリックします。 - テーブルに列を追加するには、[追加] > [列] をクリックします。
フィールドオフセットの表示	<p>レコードウィンドウ内のレコードに対するオフセットの表示形式を示すには、次のうち 1 つまたは両方のオプションを選択します。</p> <ul style="list-style-type: none"> - Decimal。10 進数形式です。 - 16 進。16 進数形式です。
カラムの SQL の種類を表示する	<p>テーブルウィンドウ内のテーブルで表示されるカラムの SQL データ型を表示するには、このオプションを選択します。</p> <p>カラムのデータ型を非表示にするには、このオプションの選択を取り消します。</p>
カラム名のフィルタ	<p>列名に基づくデータマップ内の列をフィルタリングするにはこのオプションを選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> - インクルード。指定する名前のパターンと一致する列を含みます。 - 除外。名前のパターンと一致する列を除外します。 <p>デフォルトでは、[除外] は選択されています。</p>
カラム名	<p>フィルタ条件として使用するカラム名を入力します。以下のワイルドカード文字を使用することができます。</p> <ul style="list-style-type: none"> - 1 つ以上の一致文字を表すアスタリスク (*)。 - 疑問符 (?) は、疑問符 (?) <p>デフォルトでは、PowerExchange により、FILLER で始まる名前を持つカラムはすべて除外されます。</p>
レコード数	<p>データマップに関連付けられているデータファイルを表示する際に、[データファイル] ウィンドウに一度に表示できる最大レコード数を入力します。</p> <p>有効な値は 1~999999 です。デフォルトは 10 です。</p> <p>非常に大きい値を指定すると、PowerExchange Navigator の応答時間と CPU 使用率が増加する可能性があります。</p> <p>注: [データファイル] ウィンドウでデータファイルが開かれている場合、このオプションを設定する前に、データファイルと関連付けられているデータマップを閉じてください。そうしないと、次回ウィンドウを開くまで、新しいオプション設定は有効になりません。</p>

オプション	説明
表示形式	<p>データファイルビューでのレコードの表示形式を選択します。次のオプションがあります。</p> <ul style="list-style-type: none"> - 文字/16 進。レコードが文字と 16 進形式で表示されます。 - 文字。レコードが文字形式で表示されます。 - 垂直方向。レコードが文字と 16 進数形式で縦方向に表示されます。 <p>デフォルトは垂直方向です。</p> <p>注: 【データファイル】 ウィンドウでデータファイルが開かれている場合、このオプションを設定する前に、データファイルと関連付けられているデータマップを閉じてください。そうしないと、次回ウィンドウを開くまで、新しいオプション設定は有効になりません。</p>
開発者モード (ノードに送信)	<p>開発者ユーザーがデータベース行テストを開始したときにプロンプトを表示するには、このチェックボックスを選択します。プロンプトは、行テストを実行する前に、実行可能データマップファイルをリモートロケーションに送信するかどうかを尋ねます。この設定を使用して、データマップを管理する開発者のみがデータマップをリモートノードに送信できるように設定できます。そうすると、リモートノードに後のデータマップをオーバーレイするリスクを最小限に抑えることができます。オフロードを使用する場合は、データマップをリモートノードに送信する必要があります。このオプションが選択されていない場合、どのユーザーもデータマップをリモートノードに送信できません。また、データマップは、行テストのリスナの場所に存在している必要があります。</p> <p>デフォルトでは、このチェックボックスは選択されています。</p>
行のテストの 接続タイムアウト (分)	<p>動作可能なデータベース行のテストのためのデータソースへの接続がタイムアウトするまでの分数を入力します。動作可能なデータベース行のテストは、【データベース行のテスト】 ダイアログボックスで開かれ、EOF (ファイルの終わり) に達していないテストです。行のテストがアクティブな場合は、PowerExchange Navigator ウィンドウの右下の隅に、タイムアウトに達するまでの残り時間が表示されます。タイムアウトに達すると、PowerExchange Navigator はデータソースから切断します。</p> <p>有効な値は 0~1440 (24 時間) です。0 を指定すると、行のテストのタイムアウトは実施されません。デフォルトは 15 分です。</p>
行テストのフォントの高さ	<p>行テスト結果のフォントの高さを入力します。有効な値は 8~72 ポイントです。デフォルトは 16 です。</p> <p>行テストのフォントの高さを変更する前行テストを実行した場合は、PowerExchange Navigator を再起動して、変更を有効にします。行テストのフォントの高さを変更する前行テストを実行しなかった場合、変更は PowerExchange Navigator ですぐに、または次に PowerExchange Navigator を起動したときに有効になります。</p>

3. **[OK]** をクリックします。

関連項目：

- [「データファイルの表示」 \(ページ 83\)](#)
- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)
- [「Adabas データマップへの FDT のインポート」 \(ページ 148\)](#)

IDMS データマップ設定の定義

IDMS データマップでは、Distributed Database System (DDS) のディクショナリ情報を定義することができます。

この情報はすべての IDMS データマップに使用されますが、これらの値を個々の IDMS データマップにオーバーライドすることができます。

1. **【オプション】** > **【設定】** をクリックします。
2. **【IDMS のプロパティ】** タブをクリックします。

以下の表に、入力する情報を示します。

オプション	説明
ディクショナリ名	サブスキーマのメタデータが取得される DDS ディクショナリ名です。
DB 名	ディクショナリが含まれる IDMS データベースです。
ディクショナリのユーザー ID	指定したディクショナリにアクセスする権限を持つユーザー ID です (必要な場合)。
ディクショナリのパスワード	ディクショナリのユーザー ID を指定した場合はそのユーザー ID のパスワードを入力します。
ノード名	DDS ノード名です。
ディクショナリノード	DDS ディクショナリノード名です。

3. **【OK】** をクリックします。

関連項目：

- [「IDMS データマップ」 \(ページ 119\)](#)

パスワードの暗号化ユーティリティ

セキュリティを強化するには、パスワードの暗号化ユーティリティを使用してクリアテキストのパスワードまたは有効な PowerExchange パスフレーズを暗号化します。PowerExchange は、i5/OS および z/OS にアクセスする際にのみパスフレーズをサポートします。

暗号化したパスワードまたはパスフレーズを使用して、ソースへのアクセスの制御、pwxcmd コマンドの実行、一部の PowerExchange ユーティリティの実行、または PowerExchange ロgger (Linux、UNIX、Windows 用) の構成を行うことができます。暗号化されたパスワードまたはパスフレーズをコピーして、関連するコマンドパラメータ、ユーティリティ制御文、または構成ファイルに貼り付けます。

以下の表に、暗号化されたパスワードまたはパスフレーズを指定できるパラメータを示します。

場所	パラメータまたはフィールド	参照マニュアル
pwxcmd コマンド	-epassword	<i>PowerExchange コマンドリファレンス</i>
DBMOVER 構成ファイル	UDB CAPI_CONNECTION 文の EPWD LDAP_BIND_EPWD 文の <i>search_user_password</i> LDAP_TLS 文の EPASS SSL 文の EPASS	『 <i>PowerExchange</i> リファレンスマニュアル』
PowerExchange ロggerの pwxccl.cfg 構成ファイル	CAPTURE_NODE_EPWD	『 <i>PowerExchange CDC ガイド (Linux、UNIX、Windows 用)</i> 』
DTLREXE ユーティリティ [DTLREXE ゆていりてい]	-epwd	<i>PowerExchange ユーティリティガイド</i>
DTLUAPPL ユーティリティ	EPWD	<i>PowerExchange ユーティリティガイド</i>
DTLUCBRG ユーティリティ	EPWD	<i>PowerExchange ユーティリティガイド</i>
DTLUCUDB ユーティリティ	EPWD	<i>PowerExchange ユーティリティガイド</i>
DTLURDMO ユーティリティ	EPWD	<i>PowerExchange ユーティリティガイド</i>
DTLUTSK ユーティリティ	EPWD	<i>PowerExchange ユーティリティガイド</i>

暗号化パスワードの生成

一部の PowerExchange 構成ファイル、pwxcmd コマンド、およびユーティリティ文で暗号化パスワードを指定できます。z/OS にアクセスする場合は、有効な PowerExchange パスフレーズを暗号化できます。

1. **【ファイル】 > 【パスワードの暗号化】** をクリックします。

【暗号化されたパスワードの生成】 ダイアログボックスが表示されます。

2. **【パスワード】** ボックスで、クリアテキストのパスワードを入力します。あるいは、i5/OS または z/OS にアクセスする場合は有効な PowerExchange パスフレーズを入力します。

i5/OS アクセス用のパスフレーズは、9～31 文字の長さで指定できます。z/OS アクセス用のパスフレーズは 9～128 文字（非 ODBC）または 9～79 文字（ODBC）の長さで指定できます。パスフレーズには次の文字を含めることができます。

- 大文字および小文字
- 0～9 の数字
- スペース

- 次に示す特殊文字。

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

最初の文字はアポストロフィです。

パスフレーズに一重引用符 (')、二重引用符 (")、通貨記号を含めることはできません。

3. **【暗号化されたパスワードの生成】** をクリックします。

暗号化パスワードまたはパスフレーズが **【暗号化パスワード】** ボックスに表示されます。

4. 暗号化されたパスワードまたはパスフレーズをコピーして、構成ファイル、ユーティリティ文またはコマンドオプションの該当パラメータに貼り付けます。

例えば、以下の暗号化されたパスワードを `pwxccl.cfg` 構成ファイルの `CAPTURE_NODE_EPWD` パラメータに指定します。

`CAPTURE_NODE_EPWD=F40B3B2742FB3A1F`

第 2 章

データマップ

この章では、以下の項目について説明します。

- [データマップの概要, 52 ページ](#)
- [データマップの追加, 66 ページ](#)
- [データマップレコードの追加, 69 ページ](#)
- [レコードへのフィールドの追加, 70 ページ](#)
- [レコードへのユーザー定義フィールドの追加, 70 ページ](#)
- [Datacom レコードへの項目の追加, 71 ページ](#)
- [IDMS レコードへの項目の追加, 75 ページ](#)
- [IMS レコードへの項目の追加, 77 ページ](#)
- [データマップテーブルの追加, 80 ページ](#)
- [テーブルへの列の追加, 80 ページ](#)
- [レコードおよびフィールドの表示, 81 ページ](#)
- [テーブルとカラムの表示, 83 ページ](#)
- [データファイルの表示, 83 ページ](#)
- [データマップレコードおよびテーブルのソート, 85 ページ](#)
- [データマップまたはデータマップ内の項目のプロパティの表示または編集, 86 ページ](#)
- [データマップ内の項目のコピー、切り取り、貼り付け, 92 ページ](#)
- [データマップおよびデータにアクセスするためのログオン情報の編集, 93 ページ](#)
- [リモートシステムへのデータマップの送信, 95 ページ](#)
- [リモートシステムに送信されるデータマップのインポート, 97 ページ](#)
- [データマップと複数のデータファイルの関連付け, 99 ページ](#)
- [データマップ、またはデータマップ内の項目の印刷, 101 ページ](#)
- [データマップまたはデータマップ内の項目の削除, 104 ページ](#)

データマップの概要

非リレーショナルレコードをリレーショナルテーブルにマッピングし、PowerExchange が SQL 言語を使用してデータにアクセスできるようにデータマップを定義することができます。

IMS データベース、VSAM ファイル、シーケンシャルデータセット、フラットファイルのような非リレーショナルソースでは、バルクデータ移動操作の実行や、キャプチャ登録および CDC 用抽出マップの定義に使用されるメタデータの提供に、データマップが必要となります。

DB2 ソースでは、複数フィールドを含む列のデータを別々のフィールドに分割するようにデータマップを定義することもできます。

フラットファイルまたはシーケンシャルデータセットのデータマップは、レコードタイプおよびデータマップにあるデータを定義するデータファイルと関連付けられている必要があります。

データマップを追加する場合、オプションとしてデータソースのデータのレイアウトを定義するコピーブックをインポートし、Adabas データマップには、Adabas フィールド定義テーブル (FDT) および Adabas キーをインポートします。あるいは、1 つ以上のレコードやデータマップ内のテーブルを手動で定義することによって、データのレイアウトを定義することもできます。

データマップ内のレコードによって、ソースまたはターゲット内のレコードのフィールドレイアウトが定義されます。データマップ内のテーブルでは、データのリレーショナルビューが提供されます。PowerExchange では、SQL 言語を使用してデータソースにアクセスするために、以下のようにリレーショナルビューが必要となります。

- PowerExchange では、非リレーショナルソースおよびターゲットに対し、SQL 言語の限定サブセットがサポートされます。PowerExchange により、SQL 文が内部で解釈され、必要に応じてデータソースタイプへのアクセス用に変換されます。

詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

- PowerExchange では、非リレーショナルソースおよびターゲットに対し、大量の SQL 言語もサポートされます。PowerExchange によって、処理のために SQL 文が RDBMS に渡されます。

PowerExchange により、データマップ内のフィールドはバイトレベルにマッピングされます。PL/I コピーブックなど、一部のコピーブックには、ビットレベルのマッピングがあります。ビットレベルのマッピングを含むコピーブックをデータマップにインポートした場合、PowerExchange はビットフィールドをバイトフィールドに連結し、最初のビットフィールドのフィールド名を使用してそのフィールドに名付けます。データマップでは、このフィールドをビットに分離して戻すために、PowerExchange 関数を式の中で 1 つ以上使用するユーザー定義フィールドを定義することができます。次に、PowerCenter で、ビット文字列の一体化したビットの数値を計算するために、式トランスフォーメーションを使用します。

データベースの行のテストをデータマップ上で実行して、ソースデータを取得できることを確認することができます。

データマップの開閉、削除、およびレコードやデータマップ内のテーブルの列のフィールドの検索については、[「PowerExchange リソース」 \(ページ 30\)](#)を参照してください。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)

データマップ名

PowerExchange ではデータマップ用の命名規則が用意されています。

データマップを追加する場合、以下の値を指定します。

- デモなどのスキーマ名
- map1 などのマップ名

PowerExchange はこれらの値を使用して、例えば、demo.map1 というようなスキーマ.map の形式のデータマップ名を構築します。

命名規則を利用して、プロジェクト、オペレーティングシステム、またはその他のカテゴリ別にデータマップをグループ化することができます。

テーブル上でデータベース行のテストを実行すると、PowerExchange により、スキーマ名とマップ名のプレフィックスが付けられたテーブル名を含む SQL 文が生成されます。命名形式は、**【データベース行のテスト】** ダイアログボックス、および **【設定】** ダイアログボックスの **【データマップ】** タブの **【2 層から成る名前を使用する】** オプションの設定で選択する **【DB タイプ】** により異なります。

2 階層、または 3 階層式の命名規則を使用することもできます。

- 2 階層式の命名規則には以下の命令文があります。

```
select * from schema.map_table
```

- 3 階層式の命名規則には以下の形式があります。

```
select * from schema.map.table
```

2 階層式命名規則を使用するには、**【オプション】** - **【設定】** をクリックします。**【設定】** ダイアログボックスの **【データマップ】** タブで、**【2 層から成る名前を使用する】** オプションを選択して 2 階層式の名前を使用します。3 階層式の名前を使用するには、このオプションの選択を取り消します。

関連項目：

- [「データマップ設定の定義」 \(ページ 44\)](#)

データマップ用にサポートされるデータソース

i5/OS、z/OS、Linux、UNIX、Windows オペレーティングシステム上のさまざまなデータソースにデータマップを定義することができます。

データマップを定義する場合には、そのデータソースおよびオペレーティングシステムに適した PowerExchange アクセス方式を選択する必要があります。

以下の表に、各データソースおよびオペレーティングシステムで使用するアクセス方式を示します。

データソース	i5/OS	Linux、UNIX、 および Windows	z/OS	PowerExchange アク セス方式
Adabas ファイル	-	X	X	ADABAS
CA Datacom/DB ファイル	-	-	X	DATAKOM
CA IDMS/DB データベース	-	-	X	IDMS
DB2 データベース	X	X	X	DB2
DB2 アンロードファイル	X	X	X	DB2UNLD
フラットファイルおよびシーケンシャルデータ セット	X	X	X	SEQ
IBM MQSeries メッセージキュー	X	X	X	MQSERIES
IMS データベース	-	-	X	DL/1 バッチ

データソース	i5/OS	Linux、UNIX、 および Windows	z/OS	PowerExchange アク セス方式
PowerExchange リスナ経由でアクセスする IMS データベース	-	-	X	IMS ODBA
テープデータセット	-	-	X	TAPE
ユーザー定義のユーザーアクセス方法プログラ ム	X	X	X	USER
VSAM エントリシーケンスのデータセット (ESDS)	-	-	X	ESDS
VSAM キーシーケンスのデータセット (KSDS)	-	-	X	KSDS
VSAM 相対レコードデータセット (RRDS)	-	-	X	RRDS

データファイルおよびファイルリスト処理

フラットファイルまたはシーケンシャルデータセットのデータマップは、データマップ内のレコードタイプおよびデータを定義するデータファイルと関連付ける必要があります。

複数データファイルをデータマップと関連付けるために、データマップをファイルリストファイルと関連付けることができます。複数データファイルをファイルリストファイルで一覧表示します。PowerExchange は、データファイルにあるデータをファイルのリスト順に連結させて処理します。

注: ファイルリスト処理では、ESDS、SEQ、TAPE アクセス方式で定義されたデータマップのみがサポートされます。

以下のデータソース用に、データファイルとデータマップを関連付ける必要があります。

- DB2 アンロードファイル
- フラットファイル
- シーケンシャルデータセット
- テープデータセット
- ユーザーアクセス方式プログラム
- VSAM ESDS、KSDS、および RRDS データセット

データファイルまたはレコードに関連付けられたファイルを決定するには、データマップ内のユーザー定義フィールドにある式の GetCurrentFileName 関数を使用することができます。

注: PowerExchange は、LARGE 形式のシーケンシャルデータセットを読み取ることができません。データセットの作成時に DD 文で DSNTYPE=LARGE を指定すると、PowerExchange はそれを読み取ることができません。代わりに、DD 文で DSNTYPE=EXTREQ を指定できます。

関連項目：

- [「データマップと複数のデータファイルの関連付け」 \(ページ 99\)](#)
- [「手順 1. データマップとデータファイルの関連付け」 \(ページ 263\)](#)
- [「手順 1. 複数レコードデータマップと 1 つのデータファイルの関連付け」 \(ページ 266\)](#)
- [「GetCurrentFileName」 \(ページ 310\)](#)

コピーブック

PowerExchange Navigator では、用語のコピーブックは、通常、データマップにインポートしてデータのレイアウトを定義するデータソースからのレコード、セグメント、フィールド、キー、およびその他の項目に対するデータ定義を意味します。

コピーブックをインポートすることで、データマップのこれらの項目のレイアウトを手動で定義する必要がなくなります。

コピーブックは、データマップを追加するときか追加した後にインポートできます。1 つ以上のコピーブックをデータマップのレコードまたはセグメントにインポートして、データ定義をレコードまたはセグメントに追加することもできます。

コピーブックは、インポートした後にローカルシステムに格納されます。その後ローカルシステムからそのコピーブックを再インポートできます。

コピーブックのタイプを選択して、**[コピーブックのインポート - ソース詳細]** ダイアログボックスの **[タイプ]** リストにインポートします。インポートできるコピーブックのタイプは、データマップのデータソースタイプによって異なります。

関連項目：

- [「コピーブック」 \(ページ 142\)](#)

Adabas フィールド定義テーブルおよび Adabas キー

データマップを追加するとき、あるいは追加した後に、Adabas フィールド定義テーブル (FDT) および Adabas キーを Adabas データマップにインポートすることができます。

それぞれのデータベースファイルに対応する FDT があり、これにより物理ファイルの各フィールドのレコード構造体とコンテンツが定義されます。レコードの各フィールドで、FDT によりレベル、名前、長さ、形式、オプション、特別なフィールドおよび記述子属性が定義されます。

FDT をインポートした場合、データマップ内のデータ定義がローカルシステムに保存されているため、リモートシステムに接続せずに、後で Adabas データマップを開くことができます。

PowerExchange Navigator から FDT を Adabas データマップに表示することができます。FDT を表示する場合に、リモートシステムにある FDT からのデータでローカルで保存した FDT を、オプションとしてリフレッシュすることができます。

関連項目：

- [「Adabas データマップへの FDT のインポート」 \(ページ 148\)](#)
- [「Adabas データマップ用 FDT の表示とリフレッシュ」 \(ページ 149\)](#)

データマップのレコードとフィールド

データマップには、ソースまたはターゲットのレイアウトを記述するために定義またはコピーブックからインポートしたレコードとフィールドが含まれます。フィールドにはユーザー定義のフィールドと、配列として定義されたフィールドを含めることができます。

注: レコードとは、レコードおよびセグメントの両方を示す用語です。

データマップのリレーショナルビューでは、テーブル列はデータマップレコードで定義されたフィールドに基づきます。

関連項目：

- [「\[テーブルプロパティ\] ダイアログボックスでの配列プロパティの定義」 \(ページ 60\)](#)

単一レコードデータマップと複数レコードデータマップ

ソースまたはターゲットデータベース、またはデータファイル用の 1 つ以上のレコードを定義するデータマップを作成できます。最も簡単なケースでのデータマップは、1 つのレコードと、レコードから派生した 1 つのテーブルで構成されます。テーブルはデータのリレーショナルビューを提供します。

複数のレコードを定義するコピーブックをインポートするか、フィールドのレコード ID 値を指定することにより、複数のレコードを持つデータマップを作成できます。複数のレコードを持つデータマップは複数レコードのデータマップとして参照されます。

複数レコードのデータマップは、次の場合に使用します。

- IMS ソースの各セグメントに異なるレコード定義が存在する。
- シーケンシャル、フラットファイル、または VSAM データソースに対し、[フィールドプロパティ] ダイアログボックスのフィールドに入力したレコード ID 値に基づいて異なるレコードを定義した。
- あるときは数値データ、あるときは英文字データというようなレコード内の他の情報に依存するフィールドデータを格納する再定義レコードを区別する必要がある。

シーケンシャルファイルまたは VSAM データソース用のデータマップを作成する場合、レコードまたは再定義レコード間の階層的リレーションシップを識別するレコードプロパティを定義できます。PowerExchange と PowerCenter はこの情報を、ソースデータのシーケンスを保持しながら複数のレコードタイプを VSAM またはシーケンシャルターゲットに書き込むのに使用します。IMS アンロードデータセットソースでは、複数のレコードタイプを IMS アンロードデータセットターゲットに書き込むために、PowerExchange は IMS DBD から複数レコードのデータマップにインポートしたセグメント情報を使用します。

単一レコードデータマップと複数レコードデータマップの作成例は、[第 9 章、「PowerExchange Navigator の例」 \(ページ 258\)](#)を参照してください。

SEQ または VSAM データマップに定義できる階層的リンク情報の詳細については、[「レコードプロパティ - 名前」 \(ページ 375\)](#)を参照してください。

ユーザー定義フィールド

データマップでは、ソースデータの処理に PowerExchange 関数を式の中で 1 つ以上使用するレコードのフィールドを定義することができます。

これらの関数により、フィールドの分割や連結などの操作を行うことができます。また、ユーザー定義プログラムを呼び出す式で C allProg 関数を使用して、ソースデータを処理することもできます。

関連項目：

- [「レコードへのユーザー定義フィールドの追加」 \(ページ 70\)](#)

レコードレベルの出口プログラム

アセンブリ言語、C、COBOL、または PL/I のレコードレベルの出口プログラムを作成してユーザー定義プログラムを呼び出し、データ取得前にデータマップ内のレコードにあるデータを処理します。

例えば、レコードレベルの出口プログラムを設定し、データ取得前のレコードにある圧縮データを解凍するプログラムを呼び出します。

以下のアクセス方式のいずれかを使用するデータマップに、レコードレベルの出口プログラムを定義できます。

- DATACOM

- DL/1 BATCH
- ESDS
- IDMS
- IMS ODBA
- KSDS
- MQSERIES
- SEQ
- TAPE

レコードレベルの出口プログラムを定義するには、**【データマッププロパティ】** ダイアログボックスの **【レコードレベルの出口】** タブで、レコードレベルの出口プログラムの属性を指定します。レコードレベルの出口プログラムのコンパイル済み DLL 名または共有ライブラリ名、プログラムの呼び出し規約、およびプログラムに渡すパラメータを含めます。PowerExchange は、レコードレベルの出口プログラムを繰り返し呼び出して、すべてのデータが処理されるまでレコード内のデータを処理します。

PowerExchange は、代表的なレコードレベルの出口プログラムをヘッダファイルおよび JCL と共に提供しています。urle*. *と名付けられたこれらのファイルは、PowerExchange Navigator インストールディレクトリのサンプルプログラムディレクトリにあります。これらのサンプルプログラムをコンパイルし、使用環境に合わせてカスタマイズします。

関連項目：

- [「データマッププロパティ - レコードレベルの出口」 \(ページ 373\)](#)
- [「ユーザーアクセス方法プログラム」 \(ページ 339\)](#)

データマップテーブルおよび列

データマップでは、テーブルにデータのリレーショナルビューが提供されます。

コピーブックをデータマップにインポートして、ソース内のデータのレイアウトを定義すると、PowerExchange にインポートされたレコードやフィールドに基づいてテーブルや列がデータマップ内に作成されます。また、データマップにテーブルや列を手動で追加することもできます。テーブルの列はレコードにあるフィールドに基づきます。

関連項目：

- [「データマップテーブルの追加」 \(ページ 80\)](#)

単純なテーブルと複雑なテーブル

単純なテーブルは単一のレコードに基づきます。複雑なテーブルは 1 つ以上のレコードに基づきます。

複雑なテーブルでは、**【テーブルプロパティ】** ダイアログボックスのオプションを使用して、そのテーブルが基にしているレコード間の階層関係を定義します。データマップでは、複雑なテーブルは、IMS データベースまたはシーケンシャルデータセットなどの関連データソースにあるレコードやセグメントの階層をミラーリングします。

複雑なテーブルは、2 つ以上のテーブルに存在する情報を、より正規化された形式で 1 つのテーブルに統合します。

IMS データベース内の親と子のセグメントやシーケンシャルデータセット内の親と子のレコードなど、複数の関連レコードに基づいた複雑なテーブルを定義できます。ただし、PowerCenter にデータソースをインポートするときには、IMS アンロードファイル、シーケンシャルファイル、または VSAM ソースやターゲットなどの

階層的リンク情報を保持するには、PowerExchange が生成できる単純な単一レコードのテーブルとシーケンスフィールドを使用します。これにより、親テーブルのカラムデータが反復されなくなるため、ファイル処理とデータ抽出が迅速に行われます。

以下の表に、複雑なテーブルをサポートする関連アクセス方式と共にデータソースを一覧表示します。

データソース	アクセス方式	コメント
CA IDMS/DB データベース	IDMS	CA IDMS/DB では、1 つのレコードが複数の親を持つ場合もあります。
IMS データベース	IMS ODBA - ODBA インターフェースを使用 DL/1 BATCH - BMP および DL/1 バッチジョブを使用	IMS では、[IMS 階層の表示] コマンドをクリックして、IMS データマップの階層を表示します。
シーケンシャルデータセットまたはフラットファイル	SEQ	-
z/OS のテープデータセット	TAPE	-
VSAM ESDS データセット	ESDS	-
VSAM KSDS データセット	KSDS	-

配列

データマップでは、配列として定義されたフィールドをレコードに含めることができます。また、OCCURS 句が含まれる従来のデータソースの一部では、複数のフィールドが配列として定義されています。

レコード内のフィールドを配列として定義するには、[フィールドプロパティ] ダイアログボックスの [配列] タブでオプションを定義します。次に、配列内のエレメントをデータベース行のテストで表示する方法やターゲットテーブル内の行と列にマップする方法を定義するために、[テーブルプロパティ] ダイアログボックスでオプションを定義します。

[フィールドプロパティ] ダイアログボックスでの配列プロパティの定義

レコード内のフィールドを配列として定義するには、[フィールドプロパティ] ダイアログボックスの [配列] タブでオプションを定義します。

以下の表に、配列を定義する際に使用できるオプションを示します。

オプション	説明
配列	フィールドを配列として定義します。
固定長	フィールドを固定配列として定義します。
変数	フィールドを可変配列として定義します。

オプション	説明
配列サイズ	配列内のエレメントの数です。 有効な値は 1～32768 です。 COBOL コピーブックを使用してインポートされたレコード定義の場合、この値は OCCURS x 句で指定されている x の値に対応します。
カウントフィールド	可変配列の場合は、[配列] フィールドのこのインスタンス内の実際のエレメント数を特定するフィールドの名前です。 COBOL コピーブックを使用してインポートされたレコード定義の場合、この値は OCCURS x DEPENDING ON field-y で指定されている field-y の値に対応します。

例えば、COBOL コピーブックの次の OCCURS 句をデータマップにインポートする場合は、次のようになります。

ITEM OCCURS 3 DEPENDING ON ITEMCT PIC X(15).

以下の表に、この例の【フィールドのプロパティ】ダイアログボックスの【配列】タブに表示されるオプションと値を示します。

オプション	値
配列	このオプションは選択されています。
固定長	このオプションは選択されていません。
変数	このオプションは選択されています。
配列サイズ	3
カウントフィールド	ITEMCT

[テーブルプロパティ] ダイアログボックスでの配列プロパティの定義

配列内のエレメントをデータベース行のテストで表示する方法やターゲットテーブル内の行と列にマップする方法を定義するには、[テーブルプロパティ] ダイアログボックスでオプションを定義します。

以下の表に、配列を定義する際に使用できる、オプションを示します。[テーブルプロパティ] ダイアログボックスのオプションを示します。

オプション	説明
カラムの生成	<p>最初にテーブルを定義する場合、PowerExchange では、そのテーブルが基にしているレコードのフィールド名を使用してカラムに名前が付けられます。ただし、配列として定義されたフィールドを含むレコードには特別な命名規則が使用されます。</p> <p>テーブルを定義した後でカラム名を変更する場合や、カラム名を再生成しない場合は、次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none">- 配列フォーマットの変更の適用。 配列エレメントに対して加えた変更のみがテーブルに反映されます。- 不足しているカラムを使用してリフレッシュ。 新しいカラムのみがテーブルに追加されます。- デフォルト値にリセット。 新しい一連のカラムがフィールド名から派生したデフォルト名で生成されます。加えた変更は破棄されます。- 非表示カラムの削除。 DB2UNLD のみ。各フィールドの【テーブルから非表示】オプションのデフォルト値を使用して、新しい一連のカラムが生成されます。
フィールド	<p>配列またはグループフィールド内のエレメントの表示方法を制御します。</p> <ul style="list-style-type: none">- 配列に各エレメントを表示したり、単一の行にグループフィールドを表示するには、フィールドを選択します。- 配列に各エレメントを表示したり、別の行にグループフィールドを表示するには、フィールドの選択を取り消します。 <p>注: グループフィールドや配列として定義したフィールドのみを【フィールド】リストで表示するには、【グループと配列のみ】オプションを選択します。【フィールド】リストのレコードのすべてのフィールドを表示するには、このオプションの選択を取り消します。このオプションでは、【フィールド】リストおよび【テーブルプロパティ】ダイアログボックスに表示されるフィールドは制御されますが、データベース行のテストによる出力ウィンドウに表示されるフィールドは制御されません。</p>
選択したレコードの複数インスタンスをどのように処理しますか？	<p>次のいずれかのオプションを選択します。</p> <ul style="list-style-type: none">- 新しい行。 配列のすべてのエレメントで、新しい行が表示されるか、ターゲットに書き込まれます。- 無視。 配列の 2 番目以降のエレメントは表示されないか、ターゲットに書き込まれます。- 配列。 【配列】リストに指定されエレメントの数が表示されるか、単一出力行のターゲットに書き込まれます。【オーバーフロー時の新しい行】オプションを選択すると、出力行がいっぱいになるまで入力が行われ、以降のエレメントが無視されるか、オーバーフローエレメントが入力された新しい行が表示されます。
単一入力行の複数配列	<p>複数配列、または OCCURS 節を含む単一レコードから、複数の出力行を生成します。レコードのデータの終了時に、出力フィールドが NULL に設定されます。このオプションの詳細については、「単一入力行の複数配列」(ページ 61)を参照してください。</p> <p>このオプションは、複数の OCCURS 節を持つ、インポートされた COPYLIB のあるテーブルでのみ有効になります。</p>

それらのオプションを使用して配列内のエレメントの表示方法やマップ方法を定義する例については、[「配列を含む単一レコードデータマップの追加 - 例」\(ページ 263\)](#)を参照してください。

関連項目：

- [「配列を含む単一レコードデータマップの追加 - 例」 \(ページ 263\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

単一入力行の複数配列

従来のデータソースの一部には、1つのレコード上に複数の配列、すなわち OCCURS 句が含まれています。

PowerExchange では、1つのレコードから複数の出力行を生成し、レコードのデータの終了時に出力フィールドを NULL に設定することができます。

以下の表に、入力例を示します。

入力行	fld1 - OCCURS DEPENDING	fld2 - OCCURS DEPENDING
1	3 つの値 10、20、30	2 つの値 AA、BB
2	1 つの値 55	4 つの値 DD1、DD2、DD3、DD4

以下の表に、出力例を示します。

出力行	Fld1	Fld2
1	10	AA
2	20	BB
3	30	NULL
4	55	DD1
5	NULL	DD2
6	NULL	DD3
7	NULL	DD4

すべての出力行が完了するように、アカウント番号などの他のフィールドタイプでこの単一の入力処理対多数の出力処理を使用することもできます。

コードページ

PowerExchange では、データマップやデータマップの文字フィールドと関連付けられる標準コードページのセットが用意されています。また、ユーザー定義コードページをデータマップやフィールドと関連付けることができます。

特定のコードページをデータマップやフィールドと関連付けたり、オペレーティングシステム用のデフォルトコードページを使用することができます。

コードページをフィールドと関連付けると、フィールドレベルのコードページはデータマップレベルのコードページをオーバーライドします。

PowerExchange コードページ

PowerExchange では、標準コードページのセットが用意されています。

PowerExchange には、拡張シングルバイト、およびマルチバイトコードページサポートを提供する、以下のデータソース用の ICU ソフトウェアが含まれています。

- Datacom
- DB2 for i5/OS
- DB2 for z/OS
- フラットファイル
- シーケンシャルデータセット
- VSAM データセット

i5/OS および z/OS システムでは、IBM-037 コードページがデフォルトのコードページです。Linux、UNIX、および Windows システムでは、ISO-8859 がデフォルトのコードページです。

インストールされた PowerExchange に定義されているコードページとエイリアスをすべて一覧表示するには、ICUCHECK ユーティリティを使用します。このユーティリティの詳細については、『*PowerExchange* リファレンスマニュアル』を参照してください。

以下の表に、PowerExchange がサポートする一部のコードページを示します。

コードページ	説明
IBM-037*	米国、カナダ、ブラジル
IBM-1047*	Latin 1/オープンシステム
IBM-273*	ドイツ、オーストリア
IBM-277*	デンマーク、ノルウェー
IBM-278*	フィンランド、スウェーデン
IBM-280*	イタリア
IBM-284*	スペイン、ラテンアメリカ諸国
IBM-285*	英国
IBM-297*	フランス
IBM-424*	現代ヘブライ語
IBM-500*	インターナショナル
ISO-8859	UNIX および Windows

注: IBM-のプレフィックスが付けられたコードページは、i5/OS および z/OS システムからのデータを処理することができます。

ユーザー定義のコードページ

ユーザー定義コードページを最大 10 ページまで定義します。

ユーザー定義コードページにより、指定した 16 進値の表示や印刷を行うことができます。

ユーザー定義コードページは、USRCP nn と名付ける必要があります。 nn には、00 から 09 までの値が入ります。

PowerExchange のコードページの詳細については、『PowerExchange リファレンスマニュアル』を参照してください。

データチェック

データチェックを定義して、PowerExchange による抽出中のレコードマッピングエラーおよびフィールドレベルデータのエラーの処理方法を指定します。

次のデータチェックオプションがあります。

- レコードマッピングエラー。PowerExchange が抽出を停止するように指定することも、行をスキップしてログファイルにメッセージを書き込んだ後も継続するように指定することもできます。デフォルトでは、抽出を停止するように指定します。

マッピングエラーが発生したときに抽出を停止するデフォルトのアクションを実行しない場合は、**【行をスキップし、ログファイルメッセージを書き込む】**を選択できます。できる限りレコード ID などのツールを使用してデータのマッピングを行い、PowerExchange による行のスキップは最後の手段とすることをお勧めします。

- フィールド内のデータエラー。PowerExchange で抽出を停止するように指定することも、次のいずれかの操作を実行した後も継続するように指定することもできます。

- 行をスキップしてログファイルにメッセージを書き込む。

- フィールドを NULL に設定する。

- **【データマップのプロパティ】** データログボックスの **【データのデフォルト値】** タブに従って不良データを値で置換する。このオプションは、DATE、TIME、CHAR、VARCHAR、CLOB の各フィールドタイプで使用できます。

デフォルトでは、抽出を停止するように指定します。

注意事項:

- 【データマップのプロパティ】** ダイアログボックスの **【データのデフォルト値】** タブでフィールドタイプを選択すると、**【フィールドプロパティ】** ダイアログボックスの **【フィールドチェック】** タブにある **【データマッピングルールを使用】** オプションを選択した場合にデフォルトの設定が表示されます。また、フィールドタイプを選択すると、ユーザー定義関数への呼び出しによって発生するフィールドをチェックするかどうかも定義されます。ユーザー定義フィールドでデータチェックを実行することは推奨されません。代わりに、以下をお勧めします。

- **【データマップのプロパティ】** ダイアログボックスの **【データのデフォルト値】** タブではフィールドタイプを選択しないでください。

- **【フィールドプロパティ】** ダイアログボックスの **【フィールドチェック】** タブでは、**【データマッピングルールを使用】** を選択せずにフィールドごとにルールを定義します。

- データチェックを実行するとマッピングを実行するマシンの CPU コストが増大するため、必要な場合にのみデータチェックを実行することをお勧めします。
- データチェックは、ターゲットにロードできないデータがフィールドに含まれている場合にのみ必要です。
- DATE または TIME のカラムに無効なデータがあると、ターゲットシステムに問題が発生し、チェックが必要になることがあります。

- 最新のリレーショナルデータベースの多くでは、CHAR、VARCHAR、および CLOB の各カラムに対して空白文字の値よりも小さい 16 進値の制御文字を含めることが許容されています。ただし、これらの制御文字を許容できない従来のデータベースもあります。

関連項目：

- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)
- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)

エクスポートされたデータマップ

データマップのデータソースが i5/OS または z/OS システムのようなリモートシステムにあれば、そのデータマップをリモートシステムに送信し、PowerExchange のバルクデータ移動操作に利用できるようにします。

データマップをリモートシステムに送信すると、PowerExchange は、そのデータマップをバルクデータ移動操作中のデータ抽出に使用するプラットフォームに依存しないファイルにリモートシステム上で変換します。元のデータマップは PowerExchange Navigator システムに残りますが、ローカルデータマップを変更した場合、その変更は、リモートシステムに再度データマップを送信しない限り、リモートデータマップには反映されません。

リモートシステムに送信された、またはエクスポートされた 1 つ以上のデータマップを PowerExchange Navigator システムにインポートすることができます。インポートする各データマップについては、データマップはリモートシステムに残り、ローカルコピーは PowerExchange Navigator システムに保存されます。

関連項目：

- [「リモートシステムへのデータマップの送信」 \(ページ 95\)](#)
- [「リモートシステムに送信されるデータマップのインポート」 \(ページ 97\)](#)

Datacom データマップの項目

Datacom データマップでは、CA Datacom/DB データアクセスを最適化するエレメントとキーを定義します。

エレメントとは、CA Datacom/DB と CA Datacom/DB コマンドから発生するアプリケーションの間で使用される転送単位のことです。エレメントは、連続する 1 つ以上の列、または FIELD エンティティの出現で構成されます。エレメントには、ランタイム時にアプリケーションプログラムで使用する列が含まれている必要があります。各テーブルには 1~255 のエレメントがあります。

キーやキーセグメントにより、データ取得用にソート順が設定されます。キーは列で構成され、非連続や任意のシーケンスの可能性があります。キーの各カラムは、値の昇順と降順のどちらも可能です。キーの長さは最大 180 文字です。最大で、1 つのテーブルには 99、データベースには 999 のキーを定義します。テーブルの各行がキーに対して一意の値を指定している場合、一意のキーとして定義します。

Datacom データマップの各テーブルには、マスターキーとネイティブキーを定義する必要があります。マスターキーは、他のキーと同様に動作しますが、マスターキーへの変更を有効化または無効化することができます。ネイティブキーはデータが格納されている物理的シーケンスを決定します。ネイティブキーはマスターキーと同様のキーにもなります。

関連項目：

- [「Datacom レコードへのエレメントの追加」 \(ページ 71\)](#)
- [「Datacom データマップ」 \(ページ 113\)](#)
- [「Datacom レコードへのキーの追加」 \(ページ 72\)](#)
- [「Datacom レコードへのキーセグメントの追加」 \(ページ 74\)](#)

IDMS データマップの項目

IDMS データマップで、オーナーレコードとセットを定義し、CALC エlementにより CA IDMS/DB データベースのアクセスが最適化されるようにします。

関連項目：

- [「IDMS レコードへ CALC エlementの追加」 \(ページ 75\)](#)
- [「IDMS データマップ」 \(ページ 119\)](#)
- [「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)

IMS データマップの項目

IMS DL/1 バッチデータマップレコードまたは IMS ODBA データマップレコードで、連結キー (CCK) フィールドを定義し、IMS データベースアクセスを最適化するフィールドを検索します。

名前、位置、長さ値を含む CCK フィールドおよび検索フィールドの定義は、IMS DBD セグメントにある定義と一致させる必要があります。検索には IMS DBD セグメントで定義した任意のフィールドを含むことができますが、リストの最上位では最も重要なフィールドを持つ定義済みの SEQ フィールドのみを含みます。

関連項目：

- [「IMS レコードへの CCK フィールドの追加」 \(ページ 78\)](#)
- [「IMS データマップ」 \(ページ 123\)](#)
- [「IMS レコードへの検索フィールドの追加」 \(ページ 79\)](#)

ユーザーアクセス方法プログラム

アセンブリ言語、C、または PL/I で記述されたユーザーアクセス方法プログラムを作成して、データマップのデータを処理します。

USER アクセス方式でデータマップを定義し、ユーザーアクセス方式プログラムの属性を指定することによって、ユーザーアクセス方式プログラムを呼び出します。これには、コンパイル済みのユーザーアクセス方式の DLL または共有ライブラリの名前、プログラムに伝達されるパラメータ、および関連データファイルの完全パスと名前が含まれます。PowerExchange は、ユーザーアクセス方式プログラムを繰り返し呼び出して、すべてのデータが処理されるまで関連データファイルを処理します。

PowerExchange は、代表的なユーザーアクセス方式プログラムをヘッダファイルおよび JCL と共に提供しています。uam*. *と名付けられたこれらのファイルは、PowerExchange Navigator インストレーションディレクトリのサンプルディレクトリにあります。これらのサンプルプログラムを編集およびコンパイルし、使用環境に応じてカスタマイズします。

関連項目：

- [「ユーザーアクセス方式プログラム」 \(ページ 339\)](#)
- [「データマップの追加」 \(ページ 66\)](#)

データマップの追加

データマップを追加する際は、スキーマ名、データマップ名、アクセス方式を指定します。

フラットファイルやシーケンシャルデータセット用のデータマップの場合、そのデータマップをデータファイルに関連付けます。

オプションで、コピーブックをインポートしてデータソースのデータレイアウトを定義します。Adabas データマップの場合は、Adabas フィールド定義テーブル（FDT）および Adabas キーをインポートできます。

1. リソースエクスプローラの [リソース] タブで、[追加] > [データマップ] をクリックします。

[名前] ダイアログボックスが表示されます。

2. データマップに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
スキーマ名	ユーザー定義スキーマ名を入力します。 PowerExchange では、この名前がデータマップ名の最初の部分として、以下のように使用されます。 <i>schema_name.data_map_name</i> 有効な値は英数字です。 アルファベット文字で開始する必要があります。 最大長は 10 文字です。
データマップ名	ユーザー定義マップ名を入力します。 PowerExchange では、この名前がデータマップ名の 2 番目の部分として、以下のように使用されます。 <i>schema_name.data_map_name</i> 有効な値は英数字のみで、文字で始めます。 最大長は 10 文字です。

オプション	説明
アクセス方法	<p>次のいずれかのアクセス方式を選択します。</p> <ul style="list-style-type: none"> - ADABAS。 Adabas ファイルです。 - DATACOM。 CA Datacom/DB ファイルです。 - DB2。 i5/OS 用 DB2、Linux、UNIX、および Windows 用 DB2、z/OS データベース用 DB2 です。 - DB2UNLD。 DB2 アンロードファイルです。 - DL/I BATCH。 z/OS の IMS データベースに DL/I バッチまたは BMP を使用してアクセスします。 - ESDS。 z/OS の VSAM エントリシーケンスのデータセット（ESDS）です。 - IDMS。 z/OS の CA IDMS/DB データベースです。 - IMS ODBA。 z/OS の IMS データベースに IMS ODBA を使用してアクセスします。ODBA クエリを z/OS の PowerExchange リスナのサブタスクとして実行します。 - KSDS。 z/OS の VSAM キーシーケンスのデータセット（KSDS）です。 - MQSERIES。 IBM MQSeries メッセージキューです。 - RRDS。 z/OS の VSAM 相対レコードデータセット（RRDS）です。 - SEQ。 フラットファイルです。 - TAPE。 テープのデータセットです。 - USER。 ユーザー定義のユーザーアクセス方式プログラムです。 <p>注: DL/I BATCH または IMS ODBAP のアクセス方式を PowerCenter セッションまたはデータベース行テストの他の使用可能な方式でオーバーライドできます。</p>
レコード定義のインポート	<p>データマップにコピーブックをインポートするには、このオプションを選択します。データマップのみを追加するには、このオプションの選択を取り消します。デフォルトではオンに設定されています。</p>
キーフィールド/FDT のインポート	<p>Adabas。 Adabas FDT および Adabas キーをデータマップにインポートするには、このオプションを選択します。</p> <p>データマップのみを追加するには、このオプションの選択を取り消します。デフォルトではオンに設定されています。</p>

3. 選択するアクセス方式のダイアログボックスで情報を入力します。

以下の表に、各アクセス方式に表示されるダイアログボックスを示します。

アクセス方法	ダイアログボックス	参照
ADABAS	ADABAS アクセス方式	「データマッププロパティ - ADABAS アクセス方式」 (ページ 352)
DATACOM	DATACOM アクセス方式	「データマッププロパティ - DATACOM アクセス方式」 (ページ 354)
DB2	DB2 アクセス方式	「データマッププロパティ - DB2 アクセス方式」 (ページ 355)
DB2UNLD	DB2UNLD アクセス方式	「データマッププロパティ - DB2UNLD アクセス方式」 (ページ 355)

アクセス方法	ダイアログボックス	参照
DL/I バッチ	DL/I バッチアクセス方式	「データマップのプロパティ - DL/I バッチまたは、IMS ODBA アクセス方式」 (ページ 358)
ESDS	ESDS アクセス方式	「データマッププロパティ - ESDS アクセス方式」 (ページ 360)
IDMS	IDMS アクセス方式	「データマッププロパティ - IDMS アクセス方式」 (ページ 361)
IMS ODBA	IMS ODBA アクセス方式	「データマップのプロパティ - DL/I バッチまたは、IMS ODBA アクセス方式」 (ページ 358)
KSDS	KSDS アクセス方式	「データマッププロパティ - KSDS アクセス方式」 (ページ 362)
MQSERIES	MQSeries アクセス方式	「データマッププロパティ - MQSeries アクセス方式」 (ページ 362)
RRDS	RRDS アクセス方式	「データマッププロパティ - RRDS アクセス方式」 (ページ 364)
SEQ	SEQ アクセス方式	「データマップのプロパティ - SEQ アクセス方式」 (ページ 365)
TAPE	TAPE アクセス方式	「データマッププロパティ - TAPE アクセス方式」 (ページ 367)
USER	USER アクセス方式	「データマッププロパティ - USER アクセス方式」 (ページ 368)

4. Adabas データマップで **【キーフィールド/FDT のインポート】** オプションを選択した場合には、**【次へ】** をクリックします。次に、**【Adabas FDT インポート】** ダイアログボックスで情報を入力します。
- 以下の表に、入力する情報を示します。

オプション	説明
ユーザー ID	その場所にアクセスする権限がある Adabas ユーザー ID を入力します。
パスワード	指定したユーザー ID のパスワードを入力します。ユーザー ID を指定してパスワードを省略した場合、PowerExchange Navigator は 【パスワードが入力されていません】 ダイアログボックスにパスワードの入力を求めるメッセージを表示します。
場所	FDT の場所を選択します。次のオプションがあります。 - ローカル 。FDT は PowerExchange Navigator システムにあります。 - リモート 。FDT はリモートシステムにあります。 【場所】 リストに表示された場所は、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文によって定義されます。 デフォルトは 【ローカル】 です。

オプション	説明
ファイルのパスワード	FDT ファイルがパスワードで保護されている場合、ファイルにアクセスするためにパスワードを入力します。
セッションのユーザー ID とパスワードを保存	ユーザー ID およびパスワード（指定されている場合はファイルパスワードも）を残りのセッション用に保存するには、このオプションを選択します。デフォルトではオンに設定されています。

5. **【レコード定義のインポート】** オプションを選択した場合は、**【次へ】** をクリックします。

このオプションを選択しない場合、**【完了】** をクリックします。

データマップは、**リソースエクスプローラ**の**【データマップ】** タブに表示されます。

FDT を Adabas データマップへインポートした場合、データマップのテーブルプロパティを編集して、インポートされたキーをテーブルに追加します。

関連項目：

- [「Adabas データマップ内のテーブルへのキーの追加」](#) (ページ 112)
- [「データマップへのコピーブックのインポート」](#) (ページ 144)

データマップレコードの追加

データマップに 1 つ以上のレコードを追加します。

すべてのデータソースタイプのデータマップに、レコード名を入力してレコードをデータマップに追加します。

Datcom、IDMS、IMS DL/1 バッチ、IMS ODBA データマップには、追加情報も入力する必要があります。

1. データマップを開きます。
2. **【データマップ】** タブで、データマップを右クリックし、**【レコードの追加】** をクリックします。
【テーブルの追加】 ダイアログボックスが表示されます。
3. **【名前】** タブで、レコードプロパティを定義します。定義できるプロパティはデータソースタイプによって異なります。
4. **【OK】** をクリックします。

レコードは、**リソースエクスプローラ**の**【データマップ】** タブに表示されます。

レコードウィンドウにレコードが表示されます。

Datcom データマップレコードには、オプションとして、エレメント、キー、キーセグメントを追加します。

IDMS データマップレコードには、オプションとして CALC エレメント、オーナーレコード、セットを追加します。

IMS DL/1 バッチデータマップレコードまたは IMS ODBA データマップレコードには、オプションとして、CCK フィールドおよび検索フィールドを追加します。

関連項目：

- [「Datcom レコードへのエレメントの追加」](#) (ページ 71)
- [「Datcom レコードへのキーの追加」](#) (ページ 72)
- [「Datcom レコードへのキーセグメントの追加」](#) (ページ 74)

- [「IDMS レコードへ CALC エLEMENTの追加」 \(ページ 75\)](#)
- [「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)
- [「IMS レコードへの CCK フィールドの追加」 \(ページ 78\)](#)
- [「IMS レコードへの検索フィールドの追加」 \(ページ 79\)](#)
- [「レコード、フィールドおよびテーブルのプロパティ」 \(ページ 375\)](#)

レコードへのフィールドの追加

データマップ内のレコードに 1 つ以上のフィールドを追加します。

1. データマップとレコードを開きます。
2. レコードの最後にフィールドを追加するには、**レコードウィンドウ**のレコード名を右クリックし、**【最後にフィールドを追加】** をクリックします。
フィールドの隣にフィールドを追加するには、フィールドを右クリックして、選択したフィールドの前または後のフィールド、または GROUP フィールドの子としてフィールドを選択して追加します。
3. **【フィールドの追加】** ダイアログボックスの **【名前】** タブで、フィールドのプロパティを定義します。
4. **【OK】** をクリックします。
レコードウィンドウにフィールドが表示されます。

関連項目：

- [「フィールドプロパティ - 名前」 \(ページ 380\)](#)

レコードへのユーザー定義フィールドの追加

データマップでは、ソースデータの処理に PowerExchange 関数を式の中で 1 つ以上使用するレコードのフィールドを定義することができます。

これらの関数により、フィールドの分割や連結などの操作を行うことができます。また、ユーザー定義プログラムを実行する式で C allProg 関数を使用して、ソースデータを処理することもできます。

1. データマップとレコードを開きます。
2. **レコードウィンドウ**で、**【Expr(*n*)】** タブをクリックします。
3. レコードの最後にフィールドを追加するには、**【Expr(*n*)】** タブの任意の場所を右クリックし、**【最後にフィールドを追加】** をクリックします。
フィールドの隣にフィールドを追加するには、フィールドを右クリックして、選択したフィールドの前または後のフィールドを選択して追加します。
4. フィールドのプロパティを定義するには、**【名前】**、**【型】**、**【精度】**、**【位取り】**、**【長さ】**、**【段階】** カラムでセルをクリックします。
5. フィールドに式を入力するには、**【式】** カラムのセルをクリックするか、**【参照】** ボタンをクリックして **【式エディタ】** ダイアログボックスを開きます。

以下の表に、**【式エディタ】** ダイアログボックスで入力する情報を示します。

オプション	説明
関数リスト	PowerExchange 関数を 1 つ以上ダブルクリックして、式を構築します。 関数は 【式】 ボックスに表示されます。
式	式を入力して、データをフィールドに入力または処理します。式には、以下の項目を 1 つ以上含めることができます。 <ul style="list-style-type: none">- My text または 1234 などのテキストまたは数値定数- Concat や Split などの PowerExchange 関数- ユーザー定義アセンブラ、C、COBOL、PL/I プログラムおよびそのパラメータを指定する PowerExchange CallProg 関数
検証	【式】 ボックスの式を検証します。 PowerExchange が、エラーを検出すると、エラーメッセージが 【検証】 ボックスに表示されます。
検証	式の検証で検出されたエラーのエラーメッセージを表示します。

6. **【OK】** をクリックします。
7. ユーザー定義フィールドをテストするには、以下のアクションを実行します。
 - リソースエクスプローラの **【データマップ】** タブで、テーブルを右クリックして **【プロパティ】** をクリックします。
 - **【テーブルプロパティ】** ダイアログボックスで、**【カラムの生成】** リストから **【不足しているカラムを使用してリフレッシュ】** を選択します。
 - **【OK】** をクリックします。
 - テーブル上でデータの行テストを実行します。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)
- [「ユーザー定義フィールド用 PowerExchange 関数の概要」 \(ページ 303\)](#)

Datacom レコードへの項目の追加

Datacom データマップのレコードを編集し、Datacom 固有の項目をレコードに追加します。

Datacom レコードには、1 つ以上のエレメント、キー、キーセグメントを追加することができます。

Datacom レコードへのエレメントの追加

Datacom データマップ内のレコードに 1 つ以上のエレメントを追加します。

Datacom レコードにエレメントを追加する手順

1. Datacom データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。

【フィールドのプロパティ】 ダイアログボックスが表示されます。

3. 【エレメント】 タブで、【新規追加】 ツールバーボタンをクリックします。

【新しいエレメントの追加】 ダイアログボックスが表示されます。

4. 【新しいエレメントの追加】 ダイアログボックスで、エレメントのプロパティを定義します。

以下の表に、入力する情報を示します。

オプション	説明
名前	レコードエレメント名を入力します。
位置	レコードの開始からのエレメントの位置オフセットを入力します。 デフォルトは 1 です。
長さ	エレメントの長さを入力します。 デフォルトは 1 です。

5. 【OK】 をクリックします。

【エレメント】 タブの 【エレメント】 リストにエレメントが表示されます。

関連項目：

- [「Datacom データマップの項目」 \(ページ 64\)](#)
- [「Datacom レコード内のエレメントのプロパティの表示または編集」 \(ページ 88\)](#)

Datacom レコードへのキーの追加

Datacom データマップ内のレコードに 1 つ以上のキーを追加します。

Datacom レコードにキーを追加する手順

1. Datacom データマップを開きます。
2. リソースエクスプローラの 【データマップ】 タブで、レコードを右クリックして 【プロパティ】 をクリックします。

【フィールドのプロパティ】 ダイアログボックスが表示されます。

以下の表に、定義したキーについて 【レコードのプロパティ】 ダイアログボックスの 【キー】 タブに表示される情報を示します。

オプション	説明
長いキー名	キーの長い名前です。 最大長は 32 文字です。
短いキー名	キーの短い名前です。 最大長は 5 文字です。

3. 【キー】 タブで、【新規追加】 ツールバーボタンをクリックします。

【キーのプロパティ】 ダイアログボックスが表示されます。

4. 【キー詳細】 タブで、キーを定義します。

以下の表に、入力する情報を示します。

オプション	説明
長いキー名	キーの長い名前を入力します。 最大長は 32 文字です。
短いキー名	キーの短い名前を入力します。 最大長は 5 文字です。
マスターキー	テーブルのマスターキーを選択します。各テーブルには、マスターキーが必要です。 マスターキーはネイティブキーと同様のキーにもなります。
変更を許可する	マスターキーの変更を許可するには、このオプションを選択します。マスターキーの変更を防ぐには、このオプションの選択を取り消します。このオプションを選択しない場合、CA Datacom/DB により、テーブル内の行のマスターキーの値への変更が禁止されます。 デフォルトでは選択されていません。
一意	マスターキーの重複を禁止するには、このオプションを選択します。 デフォルトでは選択されていません。
重複を許可する	マスターキーの重複を許可するには、このオプションを選択します。 デフォルトでは選択されていません。
ネイティブ	キーをこのテーブルのネイティブシーケンスキーとして定義するには、このオプションを選択します。各テーブルには 1 つの、そして唯一のネイティブキーが必要とされます。ネイティブキーにより、データがロード、保持される順が指定されます。ネイティブキーはシーケンシャル処理で最も多く使用される順を表します。 デフォルトでは選択されていません。
有無を言わせないことが可能	インデックスに入れられた [nil] 値をキーに許可するには、このオプションを選択します。Nil とは、小さな値または空白のことです。 デフォルトでは選択されていません。
複合ブール選択を許可する	複合ブール選択を許可するには、このオプションを選択します。 デフォルトでは選択されていません。

5. **[OK]** をクリックします。

[キー] タブの **[キー]** リストにキーが表示されます。

関連項目：

- [「Datacom データマップの項目」 \(ページ 64\)](#)
- [「Datacom レコード内のキーのプロパティの表示または編集」 \(ページ 88\)](#)

Datacom レコードへのキーセグメントの追加

Datacom データマップのレコードに、1 つ以上のキーセグメントを追加します。

Datacom レコードにキーセグメントを追加する手順

1. Datacom データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。
3. **【キー】** リストの **【キー】** タブで、**【プロパティ】** ツールバーボタンをクリックします。
【キーのプロパティ】 ダイアログボックスが表示されます。
4. **【セグメント詳細】** タブで、**【新規追加】** ツールバーボタンをクリックします。
【新しいセグメント】 ダイアログボックスが表示されます。
5. **【新しいセグメント】** ダイアログボックスで、キーセグメントを定義します。

以下の表に、入力する情報を示します。

オプション	説明
名前	キーセグメント名を入力します。 注: このフィールドは読み取り専用で、 【キーのプロパティ】 ダイアログボックスの 【セグメント詳細】 タブにあります。
位置	キーセグメントの位置を入力します。 デフォルトは 1 です。
長さ	キーセグメントの長さを入力します。 デフォルトは 1 です。
順序	次のうち 1 つのオプションを選択し、キーセグメントの順を指定します。 <ul style="list-style-type: none">- ASC。昇順です。- DESC。降順です。
秘匿度	次のうち 1 つのオプションを選択し、キーセグメントの秘匿度を指定します。 <ul style="list-style-type: none">- ON。秘匿度が有効になっています。- OFF。秘匿度が無効になっています。

6. **【OK】** をクリックします。
【セグメント詳細】 タブの **【セグメント詳細】** リストにキーセグメントが表示されます。

関連項目：

- [「Datacom データマップの項目」 \(ページ 64\)](#)
- [「Datacom レコード内のキーセグメントのプロパティの表示または編集」 \(ページ 88\)](#)

IDMS レコードへの項目の追加

IDMS データマップのレコードを編集し、IDMS 固有の項目をレコードに追加します。

IDMS レコードには、1 つ以上の CALC エlementおよびオーナーレコードとセットを追加することができます。

IDMS レコードへ CALC エlementの追加

IDMS データマップのレコードに、1 つ以上の CALC エlementを追加します。

IDMS レコードに CALC エlementを追加する手順

1. IDMS データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。
3. **【CALC エlement名】** タブで、**【新規追加】** ツールバーボタンをクリックします。
【新しい CALC エlement】 ダイアログボックスが表示されます。
4. **【新しい CALC エlement】** ダイアログボックスで、CALC エlementを定義します。
以下の表に、入力する情報を示します。

オプション	説明
名前	CALC エlementの名前を入力します。
位置	レコードの開始からの CALC エlementの位置オフセットを入力します。 デフォルトは 1 です。
長さ	CALC エlementの長さを入力します。 デフォルトは 1 です。

5. **【OK】** をクリックします。
【CALC エlement名】 タブの、**【CALC エlement名】** リストに CALC エlementが表示されます。
CALC エlementに表示されたプロパティの詳細については、[「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)を参照してください。

関連項目：

- [「IDMS レコード内の CALC エlement用プロパティの表示または編集」 \(ページ 89\)](#)

IDMS レコードへのオーナーレコードとセットの追加

IDMS データマップ内のレコードに 1 つ以上のオーナーレコードとセットを追加します。

IDMS レコードにオーナーレコードとセットを追加する手順

1. IDMS データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。

3. **【オーナーレコードとセット名】** タブで、**【新規追加】** ツールバーボタンをクリックします。
【オーナーレコードとセットのプロパティ】 ダイアログボックスが表示されます。
4. **【オーナーレコードとセットの詳細】** タブで、オーナーレコードとセットのプロパティを定義します。
以下の表に、入力する情報を示します。

オプション	説明
オーナー名	親レコードの名前を入力します。「SYSTEM」と入力し、セットをシステム所有インデックスとして特定します。
セット名	レコードとオーナーのリレーションの説明を入力します。
領域名	オーナーレコードとインデックスの構造体が配置される領域の名前を入力します。既存のオーナーレコードとセットを SYSTEME というオーナー名で編集すると有効になります。
順序	<p>ランタイム時に、新しいメンバレコードの出現をセット出現に追加するための論理的順序を選択します。</p> <ul style="list-style-type: none"> - なし - 最初 - 最後 - 次へ - 前へ - 並べ替え済み <p>注: 【並べ替え済み】 を選択した場合、【ソートエレメント名】 タブが 【オーナーレコードとセット名】 ダイアログボックスに表示されます。 【ソートエレメント名】 タブをクリックしてエレメント用のソート順を設定します。</p> <p>デフォルトは「なし」です。</p>
モード	<p>CA IDMS/DB ポインタがランタイム時に保持される方法を決定する特性を選択します。</p> <ul style="list-style-type: none"> - なし - チェーン - 前へリンクされたチェーンインデックス - VSAM インデックス <p>デフォルトは「なし」です。</p>
重複	<p>CA IDMS/DB により、既存の出現のソートキーを重複させるソートキーでレコードの出現が扱われる方法を選択します。</p> <ul style="list-style-type: none"> - なし - DB キー順 - 最初 - 最後 - 許可されていません <p>デフォルトは「なし」です。</p>

オプション	説明
セットオプション	次のうち 1 つのセットオプションを選択します。 <ul style="list-style-type: none"> - なし - 必須の自動 - 必須の手動 - オプションの自動 - オプションの手動 デフォルトは「なし」です。
自然なシーケンス	正の値の前に負の値を使用してキーフィールドをソートし、評価する自然なソートシーケンスを使用するには、このオプションを選択します。 デフォルトのソートシーケンスを使用するには、このオプションの選択を取り消します。デフォルトでは、CA IDMS/DB は、データを 16 進値でソートする標準照合シーケンスを使用してキーフィールドをソートし、評価します。 デフォルトでは選択されていません。
オーナーにリンクする	セットにある指定されたタイプの各メンバレコードをオーナーレコードにリンクするには、このオプションを選択します。 リレーションがデータマップの定義されたテーブルで既に使用されている場合、このオプションを変更することはできません。
オーナーキーの位置	【オーナーにリンクする】オプションが選択されている場合に有効になります。 オーナーキーのレコード内での位置を入力します。この値はオーナーデータベースキー位置、またはインデックス値（1 に対する相対値）であり、スキーマセット文で定義されます。位置の値を入力しない場合は、値が自動的に割り当てられます。 デフォルトは 0 です。
オーナーページグループ	オーナーレコードの IDMS ページグループを入力します。 最大値は 32767 です。デフォルトは 0 です。
オーナー基数	オーナーレコードの IDMS 基数を入力します。 0 から 12 までの有効な値です。デフォルトは 8 です。

5. 【OK】をクリックします。

【オーナーレコードとセットの詳細】 タブの 【オーナーレコードとセット名】 リストにオーナーレコードとセットが表示されます。

関連項目：

- [「IDMS データマップの項目」 \(ページ 65\)](#)
- [「IDMS レコード内のオーナーレコードとセット用プロパティの表示または編集」 \(ページ 89\)](#)

IMS レコードへの項目の追加

IMS DL/1 バッチデータマップまたは IMS ODBA データマップのレコードを編集して、IMS 固有の項目を追加します。

IMS データマップレコードに、1 つ以上の CCK フィールドおよび検索フィールドを追加することができます。

IMS レコードへの CCK フィールドの追加

IMS DL/1 バッチデータマップまたは IMS ODBA データマップのレコードに、1 つ以上の CCK フィールドを追加します。

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。
3. **【CCK フィールド】** タブで、**【新規追加】** ツールバーボタンをクリックします。
【新しい CCK フィールド】 ダイアログボックスが表示されます。
4. **【新しい CCK フィールド】** ダイアログボックスで、CCK フィールドのプロパティを定義します。
以下の表に、入力する情報を示します。

オプション	説明
名前	CCK フィールドの名前を入力します。名前は、IMS DBD のフィールド名である必要があります。通常は SEQUENCE フィールドの名前です。SEGM 内で定義された任意のフィールド名になることもあります。 最大長は 8 文字です。有効な文字は ASCII 文字、ポンドおよび円記号です。
フィールドの型	フィールドのデータタイプを選択します。
精度	フィールドの精度を入力します。
位取り	フィールドの位取りを入力します。
長さ	フィールドの長さを入力します。
データコードページ	CCK フィールドのコンテンツ用のコードページを選択します。コードページでは、データベースの文字データの文字セットが示されます。

5. **【OK】** をクリックします。
【CCK フィールド】 タブの **【CCK フィールド】** リストに CCK フィールドが表示されます。
CCK カラムを表示するには、CCK フィールドが含まれるレコードに基づく IMS ソースデータマップテーブルのプロパティで、**【CCK カラムを生成する】** オプションを選択します。

関連項目：

- [「IMS データマップの項目」 \(ページ 65\)](#)
- [「IMS レコードでの CCK フィールドのプロパティの表示または編集」 \(ページ 90\)](#)
- [「テーブルプロパティ - 定義」 \(ページ 391\)](#)

IMS レコードへの検索フィールドの追加

IMS DL/1 バッチデータマップまたは IMS ODBA データマップのレコードに、1 つ以上の検索フィールドを追加します。

1. データマップを開きます。

注: メッセージ PWX-01574 が表示された場合は、データマップに対してインポートした DBD によってレコード検索フィールドにキータイプ値 "S" が指定されています。シーケンスフィールドのキータイプ "S" は廃止されています。プロパティを編集して、検索フィールドの **【キー】** の値を **U** に変更します。

2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。

【フィールドのプロパティ】 ダイアログボックスが表示されます。

3. **【検索フィールド】** タブで、**【新規追加】** ツールバーボタンをクリックします。
4. **【新しい検索フィールド】** ダイアログボックスで、検索フィールドを定義します。

以下の表に、入力する情報を示します。

オプション	説明
名前	検索フィールドの名前を入力します。名前は、IMS DBD のフィールド名である必要があります。通常は SEQUENCE フィールドの名前です。SEGM 内で定義された任意のフィールド名になることもあります。 最大長は 8 文字です。 有効な文字は ASCII 文字、ポンドおよび円記号です。
位置	検索フィールドの位置を入力します。 1 から 32767 までの有効な値です。デフォルトは 1 です。
長さ	検索フィールドの長さを入力します。 1 から 32767 までの有効な値です。デフォルトは 1 です。
キー	次のうち 1 つのキータイプを選択します。 <ul style="list-style-type: none">- 空白。- U。一意のキータイプです。- M。多数の、一意でないキータイプです。 デフォルトは U です。

5. **【OK】** をクリックします。

【検索フィールド】 タブの **【検索フィールド】** リストに検索フィールドが表示されます。

関連項目：

- [「IMS データマップの項目」 \(ページ 65\)](#)
- [「IMS レコードでの検索フィールドのプロパティの表示または編集」 \(ページ 91\)](#)

データマップテーブルの追加

コピーブックをデータマップにインポートして、ソース内のデータのレイアウトを定義すると、PowerExchange にインポートされたレコードやフィールドに基づいてテーブルや列がデータマップ内に作成されます。

また、データマップにテーブルや列を手動で追加することもできます。テーブルの列はレコードにあるフィールドに基づきます。

複数レコードに基づく複雑なテーブルを定義することができます。データマップでは、複雑なテーブルは、IMS、CA IDMS/DB データベース、データセットまたはシーケンシャルデータセットなどの関連データソースにあるレコードやセグメントの階層をミラーリングします。複雑なテーブルを定義する場合、そのテーブルが基にしているレコード間の階層関係を定義します。また、データベース行のテストで、レコードが複雑なテーブルのどの列にどのように表示されるか、また、ターゲットテーブルの行や列に対して、レコードが複雑なテーブルのどの列にどのようにマップされるかを制御するオプションを設定することもできます。

1. データマップを開きます。
2. **【データマップ】** タブで、データマップを右クリックし、**【テーブルの追加】** をクリックします。
3. データマップ用のデータソースにより、**【テーブルプロパティ】** ダイアログボックスの以下のすべてのタブのうちいくつかでは、以下のプロパティを表示または定義します。
 - **定義。** 複雑なテーブルのレコードのデータベース行のテストでの表示方法、または複雑なテーブルのレコードのターゲットテーブルでのマップ方法、テーブルが基にしているレコードの変更方法の設定を含む、複雑なテーブルのプロパティは、そのテーブルの対応列に反映されます。
 - **フィルタ。** テーブルに含める、またはテーブルから除外するために名前で行をフィルタリングします。
 - **キー。** Adabas の場合は、テーブルのプライマリキーです。
 - **IMS オプション。** IMS DL/1 バッチおよび IMS ODBA の複雑なテーブルのセグメントのプロパティです。
4. **【OK】** をクリックします。

テーブルは、**リソースエクスプローラ**の **【データマップ】** タブに表示されます。

テーブルウィンドウに、テーブルが表示されます。

関連項目：

- [「Adabas データマップ内のテーブルへのキーの追加」 \(ページ 112\)](#)
- [「複雑なテーブルを含む複数レコードデータマップの追加 - 例」 \(ページ 265\)](#)
- [「テーブルプロパティ - 定義」 \(ページ 391\)](#)
- [「テーブルプロパティ - フィルタ」 \(ページ 394\)](#)
- [「テーブルプロパティ - キー」 \(ページ 395\)](#)
- [「テーブルのプロパティ - IMS オプション」 \(ページ 394\)](#)

テーブルへの列の追加

データマップ内のテーブルに 1 つ以上の列を追加します。

1. データマップとテーブルを開きます。

2. テーブルの最後にカラムを追加するには、**テーブル**ウィンドウでテーブル名を右クリックし、**【最後にカラムを追加】** をクリックします。
列の隣に列を追加するには、列を右クリックして、選択した列の前または後の列を選択して追加します。
【列の追加】 ダイアログボックスが表示されます。
3. **【名前】** タブで、カラムのプロパティを入力します。
以下の表に、これらのプロパティを示します。

オプション	説明
名前	列にユーザー定義名を入力します。列名は、テーブル内で一意である必要があります。 有効な文字は ASCII 英数字、ポンドおよび円記号です。最初の文字はアルファベット文字である必要があります。最大長は 32 文字です。
ベースフィールド	列が基にしているレコードでフィールドを選択します。

4. **【OK】** をクリックします。
テーブルウィンドウに、列が表示されます。

関連項目：

- [「テーブル内の列のプロパティの表示または編集」 \(ページ 92\)](#)

レコードおよびフィールドの表示

データマップの 1 つ以上のレコード、および各レコードのフィールドを表示することができます。

各レコードは独自の**レコード**ウィンドウに表示されます。**レコード**ウィンドウには、レコード名とそのレコード内のフィールドが表示されます。

レコードおよびフィールドの表示手順

1. データマップを開きます。
2. **リソースエクスプローラ**の**【データマップ】** タブで、次のうち 1 つのアクションを実行します。
 - 単一レコードを表示するには、そのレコードをダブルクリックします。
 - 複数レコードを表示するには、別々の**レコード**ウィンドウで、各レコードを右クリックし、**エクスプローラ**をクリックします。

レコードウィンドウの**【フィールド】** タブに、レコード名および各フィールドの以下の情報が表示されます。

- フィールド名
- データ型
- かっこ内の精度、位取りおよび長さの値
- フィールドが配列として定義されている場合に、その配列内のエレメント数が後に続く **A RRAY** の値
- レコード内のフィールドの decimal オフセット

- レコード内のフィールドの 16 進オフセット

レコードウィンドウの **[Expr(*n*)]** タブに、任意のユーザー定義フィールドが表示されます。 *n* 変数はユーザー定義フィールド数を示します。

[Expr(*n*)] タブの各行には、フィールド定義または定義フィールドへの値の割り当てが含まれます。

以下の表に、各フィールドについて **[Expr(*n*)]** タブに表示されるカラムを示します。

カラム	説明
フィールドの状態インジケータ	フィールドのステータスを示します。次のオプションがあります。 <ul style="list-style-type: none"> - チェックマーク。フィールドは有効です。 - X。フィールドは無効です。検証列のコメントはそのフィールドに含まれるエラータイプを示します。 - 矢印。新しいフィールドは、[フィールドの追加] メニューコマンドのうち 1 つを選択した後に選択されます。
名前	フィールドのユーザー定義名です。
タイプ	フィールドのデータタイプです。
精度	フィールドの精度です。
位取り	フィールドの位取りです。
長さ	フィールドの長さです。
段階	ユーザー定義フィールドのデータベース通信タイプを示します。例えば、あるフィールドが R 段階で定義されている場合、データベースが書き込まれたときに、そのフィールドの式は再計算されません。 次のオプションがあります。 <ul style="list-style-type: none"> - R。読み取り。 - W。書き込み。 - RW。読み取りと書き込みです。
式	フィールドに定義された任意の式です。
検証	[式エディタ] ダイアログボックスの [検証] をクリックすると、 [検証] 列にメッセージが表示されます。

3. **[Expr(*n*)]** タブでフィールドをソートまたは削除するには、以下のアクションを実行します。

- ウィンドウ内でフィールドを上下に動かすには、フィールドを右クリックし、**[フィールドを上に移動]**、または **[フィールドを下に移動]** をクリックします。
- 完了したフィールドをリストの最上位に移動するには、レコードウィンドウの **[Expr(*n*)]** タブの任意の場所を右クリックし、**[フィールドの整理]** をクリックします。
- フィールドを削除するには、フィールドを右クリックし、**[フィールドの削除]** をクリックします。

4. **[Expr(*n*)]** タブのフィールドのデータタイプ属性を定義するカラムを表示または非表示にするには、レコードウィンドウの **[Expr(*n*)]** タブの任意の場所を右クリックして、以下のコマンドのうち 1 つをクリックします。

- **[型]**、**[精度]**、**[位取り]**、**[長さ]** 列を表示または非表示にするには、**[種類カラムの非表示]** または **[種類カラムの表示]** をクリックします。
- すべての列を表示し、すべての列をデフォルト幅に戻すには、**[カラムの復元]** をクリックします。

関連項目：

- [「レコードへのユーザー定義フィールドの追加」 \(ページ 70\)](#)

テーブルとカラムの表示

データマップの 1 つ以上のテーブル、および各テーブルの列を表示することができます。

各テーブルは独自の**テーブルウィンドウ**に表示されます。**テーブルウィンドウ**には、テーブル名とそのテーブル内の列が表示されます。

1. データマップを開きます。
2. **リソースエクスプローラ**の**【データマップ】** タブで、次のうち 1 つのアクションを実行します。
 - 単一テーブルを表示するには、そのテーブルをダブルクリックします。
 - 複数テーブルを表示するには、別々の**テーブルウィンドウ**で、各テーブルを右クリックし、**エクスプローラ**をクリックします。

テーブルを開くと、**テーブルウィンドウ**に、テーブル名および、そのテーブルの各列の以下の情報が表示されます。

- 列名
- 以下の形式で列が基にしているレコードとフィールド
record: field
- 列が基にしているフィールドのデータタイプ
- かっこ内の、列が基にしているフィールドの長さ

データファイルの表示

データマップに関連データファイルが含まれている場合は、**データファイルウィンドウ**にそのデータファイルを表示できます。

【データマップ設定】 ダイアログボックスに最初に表示するデフォルトの表示形式およびレコード数を定義します。データファイルを表示するときに、データファイルのすべてのレコードが表示されるまでさらにレコードを表示できます。レコード上にルーラーを表示して、レコードの文字数を表示することもできます。

1. データマップを開きます。
2. **リソースエクスプローラ**の**【リソース】** タブで、**【ファイル】 > 【データファイルの表示】** をクリックします。

注: データマップに関連データファイルが含まれていない場合、このコマンドは使用できません。

セッションで最初にデータマップのデータファイルを開いたときに、**【データファイルリモートノード】** ダイアログボックスが表示されます。その後、**データファイルウィンドウ**が表示されます。

3. **【データファイルリモートノード】** ダイアログボックスが表示されたら、データファイルが存在するシステムの接続情報を入力して、**【OK】** をクリックします。

以下の表に、入力する接続情報を示します。

オプション	説明
User ID	<p>ファイルがあるシステムにアクセスするためにユーザー ID が必要な場合はユーザー ID を入力します。</p> <p>サポートされている Linux、UNIX、または Windows システムのファイルで、PowerExchange LDAP ユーザー認証を有効にしている場合、ユーザー名はエンタープライズユーザー名になります。詳細については、『<i>PowerExchange リファレンスマニュアル</i>』を参照してください。</p>
パスワード	<p>ユーザー ID を指定した場合は、そのユーザー ID のパスワードを入力します。</p> <p>i5/OS または z/OS のファイルへのアクセスには、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符 (')、二重引用符 (")、または通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N) 以上にして PowerExchange リスナを実行します。詳細については、『<i>PowerExchange リファレンスマニュアル</i>』の「SECURITY 文」を参照してください。</p>
場所	<p>データファイルがあるシステムのノード名を選択します。</p> <p>このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。</p> <p>デフォルトは【ローカル】です。</p>
ファイルのパスワード	<p>データファイルにアクセスするためにパスワードが必要な場合は、パスワードを入力します。</p>
セッションのユーザー ID とパスワードを保存	<p>セッション中に入力した情報を保存するにはこのオプションを選択します。</p> <p>デフォルトは選択されています。</p>

以下の表に、データファイル内の各レコードについてデータファイルウィンドウに表示されるカラムを示します。

カラム	説明
レコード番号	レコード番号です。
オフセット	16進データのデータオフセットです。 縦方向で表示されます。
データ	レコードデータです。以下の表示形式のうちの1つで表示されます。 <ul style="list-style-type: none">- 文字/16進。レコードが文字と16進形式で表示されます。- 文字。文字形式内のレコードを表示します。- 垂直方向。レコードが文字と16進数形式で縦方向に表示されます。 デフォルトは縦方向です。

デフォルトでは、10個のレコードが表示されます。

- データファイルのその他のレコードを表示する場合は、レコード数を入力すると【**レコードの取得**】ツールバーボタンの隣のボックスに表示されます。【**レコードの取得**】ツールバーボタンをクリックします。すべてのレコードを取得すると、ファイルの最後に到達したことがメッセージボックスに示されます。
- 表示形式を変更したり、レコードの位取りを表示または非表示したりするには、以下のアクションを実行します。
 - 表示形式を変更するには、【**Mode**】をクリックし、表示形式をクリックします。
 - レコードの位取りを表示または非表示にするには、レコードを右クリックし、【**スケールの表示**】または【**スケールの非表示**】をクリックします。

関連項目：

- [「データマップ設定の定義」](#) (ページ 44)

データマップレコードおよびテーブルのソート

データマップ内のレコードとテーブルはアルファベット順にソートすることができます。

デフォルトでは、レコードおよびテーブルは追加された順にデータマップに表示されます。アルファベット順にレコードやテーブルをソートするのは、一時的な操作です。次にデータマップを開くときには、レコードやテーブルは元のソート順で表示されます。

データマップのデータレコードおよびテーブルをソートする手順

- データマップを開きます。
- リソースエクスプローラの、【**データマップ**】タブで、データマップを右クリックし、【**項目の並べ替え**】をクリックします。

データマップまたはデータマップ内の項目のプロパティの表示または編集

ユーザー定義フィールド、配列として定義されたフィールド、テーブル、データマップの列を含むレコードおよびフィールドのプロパティを表示または編集することができます。

注: レコードとは、レコードおよびセグメントの両方を示す用語です。

データソースタイプによって、データマップレコードの以下の項目のプロパティを表示または編集することもできます。

- **Datacom。** エlement、キー、キーセグメントです。
- **IDMS。** CALC Elementおよびオーナーレコードとセットです。
- **IMS DL/1 バッチ。** CCK および検索フィールドです。

データマップのプロパティの表示または編集

データマップ全体に関連するプロパティを表示または編集することができます。

データマップのプロパティを表示または編集する手順

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、データマップを右クリックして **【プロパティ】** をクリックします。
【データマップのプロパティ】 ダイアログボックスが表示されます。
3. データマップのデータソースタイプに基づき、一部、またはすべてのタブのデータマッププロパティを表示または編集することができます。
 - **名前。** スキーマ名、データマップ名、アクセス方式です。
 - **アクセス方式。** アクセス方式の情報です。
 - **データのデフォルト値。** 日付のデフォルト値、およびデータチェックに失敗するフィールドのための置換文字です。
 - **データチェック。** レコードマッピングおよびフィールドデータエラー用に PowerExchange により実行されるデータチェックのタイプです。
 - **レコードレベルの出口プログラム。** レコードレベルの出口プログラムです。
 - **キー。** 読み取り専用。Adabas の場合は、キーに関する情報です。

データマップレコードのプロパティの表示または編集

データマップ内のレコードまたはセグメントのプロパティを表示または編集することができます。

データマップレコードのプロパティを表示または編集する手順

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。
3. データマップのデータソースタイプに基づき、次の一部またはすべてのタブのレコードプロパティを表示または編集します。

- **名前。** Datacom および IDMS 以外のすべてのデータソースタイプの場合は、ユーザー定義レコード名です。
Datacom の場合は、長いテーブル名、短いテーブル名、およびその他の情報です。
IDMS の場合は、レコード名、対応する IDMS レコード名、およびその他の情報です。
- **インポートの詳細。** 読み取り専用。最後にインポートしたコピーブックに関する情報です。
- **キー。** Datacom の場合は、キー用の長いキー名と短いキー名です。
- **エレメント。** Datacom の場合は、エレメント用の名前、位置、長さです。
- **CALC エレメント名。** IDMS の場合は、CALC エレメント用の名前、位置、長さです。
- **オーナーレコードとセット名。** IDMS の場合は、オーナーレコードとセットのプロパティです。
- **CCK フィールド。** IMS の場合は、CCK フィールドのプロパティです。
- **検索フィールド。** IMS の場合は、検索フィールドのプロパティです。

レコード内のフィールドのプロパティの表示または編集

データマップのレコード内のフィールドのプロパティを表示または編集することができます。

フィールドの名前またはデータタイプを変更するためにフィールドのプロパティを編集する場合は、フィールドを配列として定義するか、フィールドのコードページを変更します。

レコード内のフィールドのプロパティを表示または編集する手順

1. データマップとレコードを開きます。
2. **レコード**ウィンドウで、フィールドを右クリックし、**[プロパティ]** をクリックします。
[フィールドのプロパティ] ダイアログボックスが表示されます。
3. データマップのデータソースタイプおよびフィールドのデータタイプに基づき、一部またはすべてのタブのフィールドプロパティを表示または編集します。
 - **名前。** フィールドのビジネス名です。
 - **配列。** フィールドを配列として定義するためのプロパティです。
 - **追加のプロパティ。** 読み取り専用。フィールドの拡張メタデータプロパティです。
 - **フィールドチェック。** CHAR、CLOB、NUMCHAR、PACKED、SPACKED、UPACKED、UZONED、VARCHAR、ZONED データタイプの場合は、PowerExchange によってフィールドレベルで実行されるデータチェックのタイプです。これによってデータマップに定義されたデータチェックの動作がオーバーライドされます。
 - **コードページ。** いくつかのデータソースタイプの CHAR および NUMCHAR データタイプの場合は、フィールド用のコードページです。これによってデータマップに定義されたコードページがオーバーライドされます。
 - **可変長。** VARBIN および VARCHAR データタイプの場合は、可変長フィールドのプロパティです。

関連項目：

- [「フィールドプロパティ - 名前」 \(ページ 380\)](#)
- [「フィールドプロパティ - 配列」 \(ページ 387\)](#)
- [「フィールドプロパティ - 可変長」 \(ページ 390\)](#)
- [「フィールドプロパティ - 追加のプロパティ」 \(ページ 388\)](#)
- [「フィールドプロパティ - フィールドチェック」 \(ページ 389\)](#)
- [「フィールドプロパティ - コードページ」 \(ページ 389\)](#)

Datacom レコード内のエレメントのプロパティの表示または編集

Datacom データマップのレコード内のエレメントのプロパティを表示または編集することができます。

1. Datacom データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【レコードのプロパティ】 ダイアログボックスの **【エレメント】** タブに、定義されたエレメントのプロパティが表示されます。
3. エレメントのプロパティを編集するには、**【エレメント】** リストからエレメントを選択し、**【プロパティ】** ツールバーボタンをクリックします。
4. **【新しいエレメント】** ダイアログボックスで、エレメントのプロパティを表示または編集します。

関連項目：

- [「Datacom レコードへのエレメントの追加」 \(ページ 71\)](#)
- [「Datacom データマップの項目」 \(ページ 64\)](#)

Datacom レコード内のキーのプロパティの表示または編集

Datacom レコード内のキーのプロパティを表示または編集することができます。

Datacom レコード内のキーのプロパティを表示または編集する手順

1. Datacom データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【キー】** タブで **【キー】** リストからキーを選択し、**【プロパティ】** ツールバーボタンをクリックします。
【キーのプロパティ】 ダイアログボックスの **【キー詳細】** タブに、キーのプロパティが表示されます。
4. **【キー詳細】** タブで、キーのプロパティを表示または編集します。

関連項目：

- [「Datacom レコードへのキーの追加」 \(ページ 72\)](#)
- [「Datacom データマップの項目」 \(ページ 64\)](#)

Datacom レコード内のキーセグメントのプロパティの表示または編集

Datacom レコードのキーセグメントのプロパティを表示または編集することができます。

Datacom レコード内のキーセグメントのプロパティを表示または編集する手順

1. Datacom データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【キー】** タブで、**【キー】** リストからキーを選択して **【プロパティ】** ツールバーボタンをクリックします。
【キーのプロパティ】 ダイアログボックスの **【セグメント詳細】** タブに、定義されたキーセグメントのプロパティが表示されます。

4. キーセグメントのプロパティを編集するには、**【セグメント詳細】** リストからキーセグメントを選択し、**【プロパティ】** ツールバーボタンをクリックします。
5. **【新しいセグメント】** ダイアログボックスで、キーセグメントプロパティを表示または編集します。

関連項目：

- [「Datacom レコードへのキーセグメントの追加」 \(ページ 74\)](#)
- [「Datacom データマップの項目」 \(ページ 64\)](#)

IDMS レコード内の CALC エlement用プロパティの表示または編集

IDMS レコード内の CALC エlementのプロパティを表示または編集することができます。

IDMS レコード内の CALC エlementのプロパティを表示または編集する手順

1. IDMS データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【CALC エlement名】** タブに、定義された CALC エlementのプロパティが表示されます。
4. CALC エlementのプロパティを編集するには、**【CALC エlement】** リストから CALC エlementを選択し、**【プロパティ】** ツールバーボタンをクリックします。
5. **【新しい CALC エlement】** ダイアログボックスで、CALC エlementのプロパティを表示または編集します。

関連項目：

- [「IDMS レコードへ CALC エlementの追加」 \(ページ 75\)](#)
- [「IDMS データマップの項目」 \(ページ 65\)](#)

IDMS レコード内のオーナーレコードとセット用プロパティの表示または編集

IDMS レコードのオーナーレコードとセットのプロパティを表示または編集することができます。

IDMS レコード内のオーナーレコードとセットのプロパティを表示または編集する手順

1. IDMS データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【オーナーレコードとセットのプロパティ】 ダイアログボックスが表示されます。

以下の表に、定義済みのオーナーレコードとセットについて【オーナーレコードとセットのプロパティ】ダイアログボックスの【オーナーレコードとセット】タブに表示される情報を示します。

オプション	説明
オーナー名	読み取り専用。親レコードの名前です。 SYSTEM の名前の予約名で、セットをシステム所有インデックスとして特定します。
セット名	レコードとオーナー間のリレーションの説明を指定します。

3. オーナーレコードとセットのプロパティを編集するには、【オーナーレコードとセット名】リストからオーナーレコードとセットを選択し、【プロパティ】ツールバーボタンをクリックします。
【オーナーレコードとセットのプロパティ】ダイアログボックスが表示されます。
4. 【オーナーレコードとセットの詳細】タブで、オーナーレコードとセットのプロパティを表示または編集します。

関連項目：

- [「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)
- [「IDMS データマップの項目」 \(ページ 65\)](#)

IMS レコードでの CCK フィールドのプロパティの表示または編集

IMS DL/1 バッチデータマップまたは ODBA データマップのレコードで、CCK フィールドのプロパティを表示または編集できます。

1. データマップを開きます。
2. リソースエクスプローラの【データマップ】タブで、レコードを右クリックして【プロパティ】をクリックします。
【レコードのプロパティ】ダイアログボックスの【CCK フィールド】タブに、定義された CCK フィールドのプロパティが表示されます。
3. CCK フィールドのプロパティを編集するには、【CCK フィールド】リストから CCK フィールドを選択し、【プロパティ】ツールバーボタンをクリックします。
4. 【新しい CCK フィールド】ダイアログボックスで、CCK フィールドのプロパティを表示または編集します。

関連項目：

- [「IMS レコードへの CCK フィールドの追加」 \(ページ 78\)](#)
- [「IMS データマップの項目」 \(ページ 65\)](#)

IMS レコードでの検索フィールドのプロパティの表示または編集

IMS DL/1 バッチデータマップまたは IMS ODBA データマップのレコードで、検索フィールドのプロパティを表示または編集できます。

1. データマップを開きます。

注: メッセージ PWX-01574 が表示された場合は、データマップに対してインポートした DBD によってレコード検索フィールドにキータイプ値 "S" が指定されています。シーケンスフィールドのキータイプ "S" は廃止されています。プロパティを編集して、検索フィールドの **【キー】** の値を **U** に変更します。

2. **リソースエクスプローラ**の **【データマップ】** タブで、検索フィールドを表示または編集するレコードを右クリックし、**【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスで、**【検索フィールド】** タブをクリックします。
4. 検索フィールドのプロパティを編集するには、編集するカラムセル内をクリックし、別の値を入力します。
または、リスト内のフィールドを選択し、**【プロパティ】** ボタンをクリックします。次に、個別のダイアログボックスで検索フィールドのプロパティを編集します。

関連項目：

- [「IMS レコードへの検索フィールドの追加」 \(ページ 79\)](#)
- [「IMS データマップの項目」 \(ページ 65\)](#)

データマップテーブル用プロパティの表示または編集

データマップ内のテーブルのプロパティを表示または編集することができます。

テーブル名の変更、または複雑なテーブルの階層を定義することができます。また、複雑なテーブルのレコードのデータベース行のテストでの表示方法、または複雑なテーブルのレコードのターゲットテーブルでのマップ方法を定義することもできます。

データマップテーブルのプロパティを表示または編集する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブで、テーブルを右クリックして **【プロパティ】** をクリックします。
【テーブルのプロパティ】 ダイアログボックスが表示されます。
3. データマップのデータソースタイプによって、一部またはすべてのタブのテーブルプロパティを表示または編集します。
 - **定義。** 複雑なテーブルのレコードのデータベース行のテストでの表示方法、または複雑なテーブルのレコードのターゲットテーブルでのマップ方法、テーブルが基にしているレコードの変更方法の設定を含む、複雑なテーブルのプロパティは、そのテーブルの対応列に反映されます。
 - **キー。** Adabas の場合は、テーブルのプライマリキーです。
 - **IMS オプション。** IMS DL/1 バッチおよび IMS ODBA の複雑なテーブルのセグメントのプロパティです。

関連項目：

- [「データマップの追加」 \(ページ 66\)](#)
- [「Datacom レコードへのエレメントの追加」 \(ページ 71\)](#)
- [「Datacom レコードへのキーの追加」 \(ページ 72\)](#)
- [「Datacom レコードへのキーセグメントの追加」 \(ページ 74\)](#)

- [「IDMS レコードへ CALC エLEMENT の追加」 \(ページ 75\)](#)
- [「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)
- [「IMS レコードへの CCK フィールドの追加」 \(ページ 78\)](#)
- [「IMS レコードへの検索フィールドの追加」 \(ページ 79\)](#)
- [「データマップのプロパティ」 \(ページ 352\)](#)
- [「テーブルプロパティ - 定義」 \(ページ 391\)](#)
- [「テーブルプロパティ - キー」 \(ページ 395\)](#)
- [「テーブルのプロパティ - IMS オプション」 \(ページ 394\)](#)

テーブル内の列のプロパティの表示または編集

データマップのテーブル内の列のプロパティを表示または編集することができます。

列名または列が基にしているフィールドを変更することができます。

1. データマップとテーブルを開きます。
2. テーブルウィンドウで、列を右クリックし、**【プロパティ】** をクリックします。
【カラムのプロパティ】 ダイアログボックスが表示されます。
3. **【名前】** タブで、列のプロパティを表示または編集します。

関連項目：

- [「テーブルへの列の追加」 \(ページ 80\)](#)

データマップ内の項目のコピー、切り取り、貼り付け

同じデータマップにあるレコード、フィールド、テーブル、列のコピー、切り取り、貼り付けを行うことができます。

例えば、データマップにテーブルをコピーすると、PowerExchange によりデータマップ内の選択した位置にコピーしたテーブルが挿入され、その名前にはアンダースコアと数値が付加されます。EMPLOYEE という名前のテーブルをコピーした場合、PowerExchange により EMPLOYEE テーブルのコピーは、EMPLOYEE_1 という名でデータマップに挿入されます。EMPLOYEE テーブルをもう 1 つコピーした場合、2 つめのコピーには EMPLOYEE_2 という名前がつけられます。

データマップに項目をコピーする手順

1. データマップを開きます。
2. 次のうち 1 つのアクションを実行します。
 - レコードまたはテーブルのコピーまたは切り取りを行うには、レコードかテーブルを右クリックし、**【コピー】** または **【切り取り】** をクリックします。
 - レコードのフィールドのコピーまたは切り取りを行うには、そのレコードを開きます。レコードウィンドウで、フィールドを右クリックし、**【コピー】** または **【切り取り】** をクリックします。
 - テーブルの列のコピーまたは切り取りを行うには、そのテーブルを開きます。テーブルウィンドウで、列を右クリックし、**【コピー】** または **【切り取り】** をクリックします。

3. コピーまたは切り取りを行った項目を貼り付ける項目の前か後ろの項目で右クリックし、**【貼り付け】** をクリックします。
4. **【貼り付ける位置】** ダイアログボックスで、選択した項目の前か後のいずれかに、コピーまたは切り取りを行った項目を貼り付けるオプションを選択します。
コピーまたは切り取りを行った項目は、選択した項目の前か後に表示されます。

データマップおよびデータにアクセスするためのログオン情報の編集

データマップまたはデータにアクセスするためにリモートシステムにログインする際に使用するユーザー認証情報は、編集できます。

1. データマップを開きます。
2. **【リソース】** - **【ログオン】** をクリックします。
3. **【データマップログオン】** ダイアログボックスに、データマップとデータにアクセスするリモートシステムにログインするための資格情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
場所	<p>データソースの場所を選択します。データソースが PowerExchange Navigator システムにある場合は、【ローカル】 を選択します。</p> <p>このフィールドには、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。</p> <p>デフォルトは 【ローカル】 です。</p>
データマップアップログオン	<p>データマップへのアクセスには以下の資格情報を入力します。</p> <ul style="list-style-type: none"> - ユーザー ID。 必要に応じて、データベースユーザー ID を入力してデータソースにアクセスします。 <p>サポートされている Linux、UNIX、または Windows システムのデータベースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー名はエンタープライズユーザー名になります。詳細については、『<i>PowerExchange リファレンスマニュアル</i>』を参照してください。</p> <ul style="list-style-type: none"> - パスワード。 ユーザー ID を指定した場合はパスワードを入力します。 <p>i5/OS または z/OS システムのデータマップへのアクセスには、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次を示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVE メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『<i>PowerExchange リファレンスマニュアル</i>』の「SECURITY 文」を参照してください。</p> <ul style="list-style-type: none"> - ファイルのパスワード。 Adabas では、FDT ファイルがパスワードで保護されている場合、ファイルにアクセスするためにパスワードを入力します。

オプション	説明
データアクセスログオン	<p>データソースへのアクセスには以下の資格情報を入力します。</p> <ul style="list-style-type: none"> - ユーザー ID。 z/OS の場合は、必要に応じて、データベースユーザー ID を入力してデータソースにアクセスします。 - パスワード。 z/OS には必須です。ユーザー ID のパスワードを入力します。 i5/OS または z/OS システムのデータへのアクセスには、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。 <ul style="list-style-type: none"> - 大文字および小文字 - 0～9 の数字 - スペース - 次を示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 最初の文字はアポストロフィです。</p> <p>パスフレーズに一重引用符 (')、二重引用符 (")、通貨記号を含めることはできません。</p> <p>IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: 有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p> <p>注: ログオンユーザー ID とパスワードが同一の場合でも【データマップログオン】および【データアクセスログオン】の両方に資格情報を入力する必要があります。z/OS では、さまざまなユーザーがデータマップと実際のデータを所有できます。</p>
次回必要になったときにログオンを変更できるようにする	<p>次の要求された関数を続行する前に資格情報を再度求める場合には、このオプションを選択します。</p>

リモートシステムへのデータマップの送信

データマップのデータソースがリモートシステムにある場合は、そのデータマップをリモートシステムに送信し、PowerExchange のバルクデータ移動操作に利用できるようにします。

データマップをリモートシステムに送信すると、PowerExchange は、そのデータマップをプラットフォームに依存しないファイルに変換してバルクデータ抽出に使用します。元のデータマップは PowerExchange Navigator システムに残ります。ローカルの PowerExchange Navigator データマップを変更した場合は、そのデータマップをリモートシステムに再度送信し、システム上のデータマップを更新します。

1. データマップを開きます。
2. 【ファイル】 【リモートノードへの送信】 をクリックします。

3. **【データマップリモートノード】** ダイアログボックスに、データマップを送信するリモートシステムの接続情報を入力します。

以下の表に、入力する接続情報を示します。

オプション	説明
User ID	<p>リモートシステムにアクセスするためにユーザー ID が必要な場合は、ユーザー ID を入力します。</p> <p>サポートされている Linux、UNIX、または Windows システムで PowerExchange LDAP ユーザー認証を有効にしている場合、ユーザー名はエンタープライズユーザー名になります。詳細については、『<i>PowerExchange リファレンスマニュアル</i>』を参照してください。</p>
パスワード	<p>ユーザー ID を指定した場合は、ユーザー ID のパスワードを入力します。</p> <p>i5/OS または z/OS システムへのアクセスには、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次を示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N) 以上にして PowerExchange リスナを実行します。詳細については、『<i>PowerExchange リファレンスマニュアル</i>』の「SECURITY 文」を参照してください。</p>
場所	<p>システムに存在するデータマップ送信先の場所のノード名を選択します。</p> <p>このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。</p> <p>デフォルトは 【ローカル】 です。</p>
セッションのユーザー ID とパスワードを保存	<p>セッション中に入力した情報を保存するにはこのオプションを選択します。</p> <p>デフォルトは選択されています。</p>

4. **【OK】** をクリックします。

メッセージボックスに、送信処理のステータスが表示されます。

5. **【OK】** をクリックします。

PowerExchange により、リモートシステムのデータマップが保存されます。

関連項目：

- [「エクスポートされたデータマップ」 \(ページ 64\)](#)

リモートシステムに送信されるデータマップのインポート

データマップをリモートシステムに送信またはエクスポートすると、1 つ以上のデータマップを PowerExchange Navigator システムにインポートできます。

インポートする各データマップについては、データマップはリモートシステムに残り、ローカルコピーは PowerExchange Navigator システムに保存されます。

1. 開いている PowerExchange リソースをすべて閉じます。
2. **【ファイル】** **【エクスポートされたデータマップ】** をクリックします。
3. **【データマップインポートノード】** ダイアログボックスで、データマップのインポート元リモートシステムの接続情報を入力します。

以下の表に、入力する接続情報を示します。

オプション	説明
User ID	<p>リモートシステムにアクセスするためにユーザー ID が必要な場合は、ユーザー ID を入力します。</p> <p>サポートされているリモートの Linux、UNIX、または Windows システムで PowerExchange LDAP ユーザー認証を有効にしている場合、ユーザー名はエンタープライズユーザー名になります。詳細については、『<i>PowerExchange</i> リファレンスマニュアル』を参照してください。</p>
パスワード	<p>ユーザー ID を指定した場合は、そのユーザー ID のパスワードを入力します。</p> <p>i5/OS または z/OS システムへのアクセスには、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N) 以上にして PowerExchange リスナを実行します。詳細については、『<i>PowerExchange</i> リファレンスマニュアル』の「SECURITY 文」を参照してください。</p>
場所	<p>データマップのインポート元リモートシステムのノード名を選択します。</p> <p>このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。</p> <p>デフォルトは ローカル です。</p>
ファイルのパスワード	<p>Adabas で FDT ファイルがパスワードで保護されている場合は、ファイルにアクセスできるパスワードを入力します。</p>
セッションのユーザー ID とパスワードを保存	<p>セッション中に入力した情報を保存するにはこのオプションを選択します。</p> <p>デフォルトは選択されています。</p>

4. **【エクスポートされたデータマップ】** ダイアログボックスで、データマップを 1 つ以上インポートします。
メッセージボックスに、インポート処理のステータスが表示されます。
5. **【OK】** をクリックします。
データマップが **リソースエクスプローラ** の **【リソース】** タブに表示されます。

関連項目：

- [「エクスポートされたデータマップ」 \(ページ 64\)](#)

データマップと複数のデータファイルの関連付け

PowerExchange のファイルリスト処理を使用して、複数のデータファイルを 1 つのデータマップに関連付けることができます。

単一のデータファイルを 1 つのデータマップに関連付ける代わりに、ファイルリストファイルを作成します。複数データファイルをファイルリストファイルで一覧表示します。次に、ファイルリストファイルをデータマップと関連付けて、データマッププロパティの **【処理するファイルリスト】** オプションを選択します。データファイルのデータは、PowerExchange によってファイルリストファイルに一覧表示されたデータファイルの順で連結され、処理されます。

ファイルリスト処理は、フラットファイル、シーケンシャルデータセット、ESDS、テープデータセットにサポートされています。

ファイルリスト処理を使用したデータマップに関連付けられたデータファイルを表示する場合、一覧表示されたデータファイルではなくファイルリストファイルが表示されます。

関連項目：

- [「データファイルおよびファイルリスト処理」 \(ページ 54\)](#)

PowerExchange ファイルリスト処理

ファイルリスト処理を使用するために設定されたデータマップを使用してデータベース行のテストまたはバルクデータ移動操作を実行した場合、PowerExchange によりファイルリストファイルが以下のように処理されます。

1. PowerExchange により、空白行および行の先頭や末尾にある空白がすべて無視されます。
2. PowerExchange でファイルリストファイルのエラーが検出されると、ファイルリスト処理が終了します。エラーには、無効なファイル名、存在しないファイル、読み取りエラーが含まれます。
3. PowerExchange では、PowerExchange Listener ログで有効ではないすべてのファイル名が記録されます。ファイル名は 16 進形式で記録されます。
4. PowerExchange でエラーが 1 つも検出されない場合、ファイルリストファイルに一覧表示されたデータファイルのデータは、一覧に挙げられた順に連結され、処理されます。

ファイルリストファイル構文規則

ファイルリストファイルを作成する際には、構文規則に従う必要があります。

ファイルリストファイルの作成の際には以下の構文規則を使用します。

- 行ごとに 1 つのデータファイルを一覧表示する。
- コメント行は、以下のようにスラッシュとアスタリスクで開始する。
/*
- 行の先頭と末尾の空白の間に指定された文字はすべて、データファイル名として見なす。z/OS では、シーケンス番号をオンに設定した場合、そのシーケンス番号もデータファイル名の一部として見なされます。

ファイルリストファイルを作成してデータマップと関連付け

複数データファイルをデータマップと関連付けるには、データファイルを一覧表示するファイルリストファイルを作成し、そのファイルリストファイルをデータマップと関連付けます。

注: ODBC でファイルリストファイル処理を使用するには、データマッププロパティを定義する代わりに、SQL エスケープシーケンスを設定します。

1. データファイルがあるシステム上にファイルリストファイルを作成します。 ファイルリストファイルには、各データファイルのパスとファイル名が一覧表示されます。
2. PowerExchange Navigator で、データマップを編集します。
注: ファイルリスト処理では、ESDS、SEQ、TAPE アクセス方式で定義されたデータマップのみがサポートされます。
3. **【データマップのプロパティ】** ダイアログボックスの **【アクセス方式】** タブで、**【ファイル名】** ボックスに、ファイルリストファイルのパスとファイル名を入力するか、そのファイルを参照する **【参照】** ボタンをクリックします。
4. **【処理するファイルリスト】** オプションを選択します。
5. **【OK】** をクリックします。

ODBC およびファイルリスト処理

ODBC では、ファイルリスト処理を使用することができます。

ODBC でファイルリスト処理を使用するには、以下の SQL エスケープシーケンスを使用します。

DTLDSN

ファイルリストファイルを指定します。

DTLFILELIST

ファイルリスト処理が使用されることを示します。

ODBC、ファイルリスト処理、SQL エスケープシーケンスの詳細については、『*PowerExchange リファレンス マニュアル*』を参照してください。

Linux、UNIX、Windows のファイルリストファイルの例

Linux、UNIX、Windows では、データファイルを一覧表示するファイルリストファイルを定義することができます。

filelist.dat と名付けられたファイルリストファイルの以下のデータファイルを一覧表示することができます。

```
c:\pwx\data\filelist1.dat  
c:\pwx\data\filelist2.dat  
c:\pwx\data\filelist3.dat
```

データマップを追加するには、filelist.dat ファイルをデータマップと関連付けて、データマッププロパティの **【処理するファイルリスト】** オプションを選択します。

z/OS のファイルリストファイルの例

z/OS では、データファイルメンバを一覧表示するファイルリストメンバを定義することができます。

DTLUSR.V901.V1.DEMO.ESDS と名付けられたファイルリストメンバの以下のデータファイルメンバを一覧表示することができます。

```
***** Top of Data *****  
000001 /* FILE LIST PROCESSING DEFINITION FILE
```

```

000002 /*
000003 DTLUSR.V901.V1.DEMO.ESDS1
000004 /*
000005 DTLUSR.V901.V1.DEMO.ESDS2
000006 /*
000007 DTLUSR.V901.V1.DEMO.ESDS3
000008 /*
***** Bottom of Data *****

```

データマップを追加するときに、DTLUSR.V900.V1.DEMO.ESDS メンバをデータマップと関連付けて、データマッププロパティの【**処理するファイルリスト**】オプションを選択します。

データマップ、またはデータマップ内の項目の印刷

データマップ、またはデータマップにあるレコード、フィールド、テーブル、列をプリンタまたはファイルに出力することができます。

ファイルへの出力を選択した場合、デフォルトのファイル名は、[マイドキュメント] フォルダの PowerExchange Navigator.xxx となり、xxx はプリンタタイプに適切なファイル拡張子です。

注: ユーザー定義フィールドは印刷できません。

印刷された文書または保存されたファイルでは、以下の情報を選択することができます。

- 項目のサマリ情報に加えて、詳細情報を含みます。
- 番号やインデントを使用して、項目の階層レベルを特定します。
- すべての依存オブジェクトの情報を含みます。

データマップの印刷

データマップは一度に 1 つだけ印刷することができます。

データマップを印刷する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の【**データマップ**】タブで、データマップを右クリックして【**印刷**】をクリックします。
3. 【**印刷**】ダイアログボックスで、印刷オプションを表示または編集します。

以下の表に、印刷オプションを示します。

オプション	説明
名前	プリンタを選択します。【 プロパティ 】をクリックして、プリンタのプロパティを設定します。
ステータス	読み取り専用。選択したプリンタのステータスです。
タイプ	読み取り専用。選択したプリンタのプリンタタイプです。
説明	読み取り専用。選択したプリンタの場所です。

オプション	説明
コメント	読み取り専用。選択したプリンタに定義された任意のコメントです。
ファイルへの出力	テキスト形式の項目をファイルに保存するには、このオプションを選択します。ファイルへの出力オプションに適切なプリンタタイプを選択します。
すべて	文書内のすべてのページを印刷するには、このオプションを選択します。デフォルトは選択されており、これが唯一の有効なオプションです。
ページ	使用不可。
選択	使用不可。
部数	プリンタで印刷する部数を入力します。 この値はファイルに出力された項目には適用されません。 デフォルトは 1 です。
プリント詳細	サマリ情報に加え、選択された項目の詳細情報を印刷するには、このオプションを選択します。 デフォルトは、サマリ情報の印刷です。
依存関係にあるものを印刷	選択した項目の依存オブジェクトのサマリ情報を印刷するには、このオプションを選択します。また選択されている場合は、詳細情報も印刷します。 デフォルトでは、依存オブジェクトの情報は印刷されません。
レベル番号を使用する	選択したアイテムの任意の階層を特定するレベル番号を印刷するには、このオプションを選択します。 デフォルトでは、インデントを使用します。

4. **[OK]** をクリックします。

PowerExchange は、**[ファイルへの出力]** オプションが選択されていない限り、選択したプリンタでデータマップを印刷します。

5. データマップをファイルへ出力した場合は、そのファイルプロパティを **[名前を付けて保存]** ダイアログボックスに入力して、**[保存]** をクリックします。

PowerExchange により、文書が指定した場所の指定したファイルに保存されます。

データマップレコードの印刷

データマップに 1 つ以上のレコードを印刷することができます。

データマップレコードを印刷する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の **[データマップ]** タブで、1 つ以上のレコードを印刷します。
3. **[印刷]** ダイアログボックスで、印刷オプションを定義します。印刷オプションの詳細については、[「データマップの印刷」 \(ページ 101\)](#) を参照してください。

PowerExchange は、**[ファイルへの出力]** オプションが選択されていない限り、選択したプリンタでレコードを印刷します。

4. レコードをファイルへ出力した場合は、そのファイルプロパティを **[名前を付けて保存]** ダイアログボックスに入力して、**[保存]** をクリックします。

PowerExchange により、文書が指定した場所の指定したファイルに保存されます。

レコード内のフィールドの印刷

データマップ内のレコードに 1 つ以上のフィールドを印刷することができます。

注: ユーザー定義フィールドは印刷できません。

レコード内のフィールドの印刷手順

1. データマップを開きます。
2. レコードを開きます。
3. **レコード**ウィンドウの **【フィールド】** タブで、1 つ以上のフィールドを印刷します。
4. **【印刷】** ダイアログボックスで、印刷オプションを定義します。印刷オプションの詳細については、[「データマップの印刷」 \(ページ 101\)](#) を参照してください。

PowerExchange は、**【ファイルへの出力】** オプションが選択されていない限り、選択したプリンタでフィールドを印刷します。

5. フィールドをファイルへ出力した場合は、そのファイルプロパティを **【名前を付けて保存】** ダイアログボックスに入力して、**【保存】** をクリックします。

PowerExchange により、文書が指定した場所の指定したファイルに保存されます。

データマップテーブルの印刷

データマップに 1 つ以上のテーブルを印刷することができます。

データマップテーブルを印刷する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブで、1 つ以上のテーブルを印刷します。
3. **【印刷】** ダイアログボックスで、印刷オプションを定義します。印刷オプションの詳細については、[「データマップの印刷」 \(ページ 101\)](#) を参照してください。

PowerExchange は、**【ファイルへの出力】** オプションが選択されていない限り、選択したプリンタでテーブルを印刷します。

4. テーブルをファイルへ出力した場合は、そのファイルプロパティを **【名前を付けて保存】** ダイアログボックスに入力して、**【保存】** をクリックします。

PowerExchange により、文書が指定した場所の指定したファイルに保存されます。

テーブル内の列の印刷

データマップ内のテーブルに 1 つ以上の列を印刷することができます。

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブでテーブルをクリックします。
3. **テーブル**ウィンドウで、1 つ以上の列を印刷します。
4. **【印刷】** ダイアログボックスで、印刷オプションを定義します。印刷オプションの詳細については、[「データマップの印刷」 \(ページ 101\)](#) を参照してください。

PowerExchange は、**【ファイルへの出力】** オプションが選択されていない限り、選択したプリンタで列を印刷します。

5. 列をファイルへ出力した場合は、そのファイルプロパティを **【名前を付けて保存】** ダイアログボックスに入力して、**【保存】** をクリックします。

PowerExchange により、文書が指定した場所の指定したファイルに保存されます。

データマップまたはデータマップ内の項目の削除

データマップまたはデータマップの項目を削除することができます。

複数のデータマップ、またはデータマップ項目を一度に削除することができます。

以下のデータマップ項目を削除することができます。

- レコード
- レコードからのフィールド
- テーブル
- テーブルからの列
- Datacom データマップでは、エレメント、キー、キーセグメント
- IDMS データマップでは、CALC エレメント、オーナーレコードとセット
- IMS データマップでは、CCK フィールドおよび検索フィールド

データマップまたはデータマップ項目を削除すると、PowerExchange によって、PowerExchange Navigator システム上の現在のリソース構成ディレクトリからデータマップやデータマップ項目が削除されます。

データマップの削除

データマップを削除することができます。

データマップを削除する手順

1. **リソース**ウィンドウで、1 つ以上のデータマップを削除します。
2. 削除を確認するように求めるメッセージボックスで、**【はい】** をクリックします。
PowerExchange により、データマップが PowerExchange Navigator システム上の現在のリソース構成ディレクトリから削除されます。

リモートシステムにエクスポートされたデータマップの削除

リモートシステムに送信、またはエクスポートしたデータマップを削除することができます。

リモートシステムにエクスポートされたデータマップを削除する手順

1. 開いている PowerExchange リソースをすべて閉じます。
2. **【ファイル】 - 【エクスポートされたデータマップ】** をクリックします。
3. **【データマップインポートノード】** ダイアログボックスに、エクスポートしたデータマップを削除するシステムの以下の接続情報を入力します。
4. **【エクスポートされたデータマップ】** ダイアログボックスに、エクスポートしたデータマップを 1 つ以上削除します。
5. 削除を確認するように求めるメッセージボックスで、**【はい】** をクリックします。
PowerExchange により、リモートシステムのデータマップが削除されます。

関連項目：

- [「リモートシステムへのデータマップの送信」 \(ページ 95\)](#)

データマップレコードの削除

データマップからレコードを削除することができます。

データマップレコードを削除する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の**【データマップ】** タブで、1 つ以上のレコードを削除します。
メッセージボックスにより、レコードの削除を確認するように求められます。
3. レコードに対する相互参照が存在する場合、メッセージボックスにより、そのレコードの削除によってレコードに対するすべての相互参照も削除されることが通知されます。レコードに対する相互参照を表示するには、**【詳細】** をクリックします。**【現在の削除により影響を受ける参照】** リストに、PowerExchange でレコードを削除すると削除される項目のアイコンがグレーで表示されます。
4. 削除を確認するように求めるメッセージボックスで、**【はい】** をクリックします。
PowerExchange により、レコードとそのレコードに対する相互参照が、PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップから削除されます。

レコードからのフィールドの削除

レコードからフィールドを削除することができます。

レコードからフィールドを削除する手順

1. データマップとレコードを開きます。
2. **レコードウィンドウ**の**【フィールド】** タブ、または **【Expr(*n*)】** タブのいずれかで、1 つ以上のレコードを削除します。
メッセージボックスにより、フィールドの削除を確認するように求められます。データマップ内のテーブルに、フィールドに基づく列が含まれている場合、メッセージボックスにより、フィールドの削除によって、そのフィールドを相互参照するすべての列も削除されることが通知されます。
3. どのテーブルのどの列がフィールドを相互参照しているかを表示するには、**【詳細】** をクリックします。**【現在の削除により影響を受ける参照】** リストに、PowerExchange でフィールドを削除すると削除される項目のアイコンがグレーで表示されます。
4. 削除を確認するように求めるメッセージボックスで、**【はい】** をクリックします。
PowerExchange により、レコードからフィールドが削除されます。レコードからすべてのフィールドを削除しても、PowerExchange はそのレコードを削除しません。
フィールドに基づく列がテーブル内にある場合、これらの列は PowerExchange によりテーブルから削除されます。テーブル内のすべての列に基づくレコードから、すべてのフィールドを削除した場合、PowerExchange により、テーブルからすべての列が削除されますが、テーブルは削除されません。

データマップテーブルの削除

データマップからテーブルを削除することができます。

データマップテーブルを削除する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の**【データマップ】** タブで、1 つ以上のテーブルを削除します。

3. 削除を確認するように求めるメッセージボックスで、**【はい】** をクリックします。
PowerExchange により、テーブルが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップから削除されます。

テーブルからの列の削除

テーブルから列を削除することができます。

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブでテーブルをクリックします。
3. **テーブル**ウィンドウで、1 つ以上の列を削除します。
メッセージボックスにより、列の削除を確認するように求められます。
4. 削除を確認するように求めるメッセージボックスで、**【はい】** をクリックします。
PowerExchange により、テーブルから列が削除され、また、その列に対するすべての相互参照も、PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップから削除されます。
テーブルからすべての列を削除しても、PowerExchange はそのテーブルを削除しません。

Datacom レコードからのエレメントの削除

Datacom レコードからエレメントを削除することができます。

Datacom レコードからエレメントを削除する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【エレメント】** タブで、1 つ以上のエレメントを削除します。
PowerExchange により、エレメントが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

Datacom レコードからのキーの削除

Datacom レコードからキーを削除することができます。

Datacom レコードからキーを削除する手順

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【キー】** タブで、1 つ以上のキーを削除します。
PowerExchange により、キーが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

Datacom レコードからのキーセグメントの削除

Datacom レコードからセグメントを削除することができます。

Datacom レコードからキーセグメントを削除する手順

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【キー】** タブで、ダイアログボックス上の **【プロパティ】** ボタンをクリックします。
4. **【キーのプロパティ】** ダイアログボックスの **【セグメント詳細】** タブで、1 つ以上のキーセグメントを削除します。

PowerExchange により、キーセグメントが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

IDMS レコードからの CALC エLEMENT の削除

IDMS レコードから CALC ELEMENT を削除することができます。

IDMS レコードから CALC ELEMENT を削除する手順

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【CALC ELEMENT 名】** タブで、1 つ以上の CALC ELEMENT を削除します。

PowerExchange により、CALC ELEMENT が PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

IDMS レコードからのオーナーレコードとセットの削除

IDMS レコードからオーナーレコードとセットを削除することができます。

IDMS レコードからオーナーレコードとセットを削除する手順

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【オーナーレコードとセット名】** タブで、1 つ以上のオーナーレコードとセットを削除します。

PowerExchange により、オーナーレコードとセットが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

IMS レコードからの CCK フィールドの削除

IMS DL/1 バッチデータマップまたは IMS ODBA データマップのレコードから、CCK フィールドを削除できます。

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。

3. **【レコードのプロパティ】** ダイアログボックスの **【CCK フィールド】** タブで、1 つ以上の CCK フィールドを削除します。

PowerExchange により、CCK フィールドが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

IMS レコードからの検索フィールドの削除

IMS DL/1 バッチデータマップまたは IMS ODBA データマップのレコードから、検索フィールドを削除できます。

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
3. **【レコードのプロパティ】** ダイアログボックスの **【検索フィールド】** タブで、1 つ以上の検索フィールドを削除します。

PowerExchange により、検索フィールドが PowerExchange Navigator システム上の現在のリソース構成ディレクトリのデータマップ内のレコードから削除されます。

第 3 章

特定のデータソースのためのデータマップ

この章では、以下の項目について説明します。

- [特定のデータソースのためのデータマップの概要, 109 ページ](#)
- [Adabas データマップ, 110 ページ](#)
- [Datacom データマップ, 113 ページ](#)
- [DB2 カタログデータマップ, 114 ページ](#)
- [DB2 アンロードファイルデータマップ, 117 ページ](#)
- [IDMS データマップ, 119 ページ](#)
- [IMS データマップ, 123 ページ](#)
- [SEQ データマップ, 134 ページ](#)
- [VSAM データマップ, 136 ページ](#)

特定のデータソースのためのデータマップの概要

データマップを追加するときに構成するオプションは、データマップのデータソースタイプによって異なります。

また、データマップにインポートできるコピーブックのタイプも、データマップに対するデータソースタイプによって異なります。

以下の表に、各データソースに対して、データマップの追加およびコピーブックのデータマップへのインポートの詳細を提供するトピックを一覧表示します。

データソース	PowerExchange アクセス方式	参照
Adabas ファイル	ADABAS	「Adabas データマップ」 (ページ 110)
CA Datacom/DB ファイル	DATACOM	「Datacom データマップ」 (ページ 113)
CA IDMS/DB データベース	IDMS	「IDMS データマップ」 (ページ 119)

データソース	PowerExchange アクセス方式	参照
DB2 カタログ	DB2	「DB2 カタログデータマップ」 (ページ 114)
DB2 アンロードファイル	DB2UNLD	「DB2 アンロードファイルデータマップ」 (ページ 117)
IMS データベース	DL/1 バッチ	「IMS データマップ」 (ページ 123)
PowerExchange リスナ経由でアクセスする IMS データベース	IMS ODBA	「IMS データマップ」 (ページ 123)
シーケンシャルデータセットまたはフラットファイル	SEQ	「SEQ データマップ」 (ページ 134)
ユーザー定義のユーザーアクセス方法プログラム	USER	付録 B, 「ユーザーアクセス方法プログラム」 (ページ 339)
VSAM エントリシーケンスのデータセット (ESDS)	ESDS	「VSAM データマップ」 (ページ 136)
VSAM キーシーケンスのデータセット (KSDS)	KSDS	「VSAM データマップ」 (ページ 136)
VSAM 相対レコードデータセット (RRDS)	RRDS	「VSAM データマップ」 (ページ 136)

関連項目：

- [「データマップ」 \(ページ 51\)](#)
- [「コピーブック」 \(ページ 142\)](#)
- [「データベース行のテスト」 \(ページ 229\)](#)

Adabas データマップ

Adabas データマップを追加するときか追加した後に、コピーブック、Adabas フィールド定義テーブル (FDT)、Adabas キーをインポートできます。

FDT では、データソースのデータのレイアウトを定義します。FDT をインポートした場合は、データマップのデータ定義はローカルシステムに保存されます。Adabas キーをデータマップにインポートする場合は、これらのキーをデータマップのテーブルに追加できます。

データマップを追加した後に FDT をリフレッシュできます。

次のコピーブックタイプのいずれも Adabas データマップにインポートできます。

- COBOL コピーブック。COBOL コピーブックをインポートする場合、データマップを編集してインポートされたフィールド定義を Adabas フィールド定義に関連付ける必要があります。
- PL/I コピーブック。PL/I コピーブックをインポートする場合、データマップを編集してインポートされたフィールド定義を Adabas フィールド定義に関連付ける必要があります。
- ADACMP ユーティリティを使用して作成されたファイル。

- 初期設定のデータ定義モジュール（DDM）。DDM をインポートするには、FDIC データベース ID とファイル番号の値を入力する必要があります。

注: ワイド文字データタイプを含む z/OS 上の Adabas データベースの DDM メタデータをインポートする場合は、PowerExchange ではこれらのフィールドに関して【コードページ】タブにある【ワイド文字】オプションの選択を行いません。Predict または FDT メタデータをインポートするか、データマップのフィールドプロパティを編集して、すべてのワイド文字データタイプに関する【ワイド文字】オプションを選択する必要があります。

- データベース定義を含む Adabas フィールド定義テーブル（FDT）。省略名とスケールなしの情報を提供します。FDT では、フィールド名、配列サイズ、オーバーライド長などの Adabas 関連情報の検証が有効になります。FDT をインポートした後に、FDT が後続のセッションに対してキャッシュされます。これにより、ランタイム前にいくつかの検証チェックをすることができます。インポートした FDT をリフレッシュできます。
- Predict データディクショナリ。ロングネームとスケール情報を提供します。Predict データディクショナリをインポートするには、FDIC データベース ID とファイル番号の値を入力する必要があります。
- テキストファイル。区切り文字のある Informatica 内部形式を使用します。

Adabas フィールドにマッピングされたすべての PowerExchange フィールドを細分化または再定義できます。フィールドを細分化するには、グループを再マッピングするサブフィールドのあるグループフィールドとして定義します。

Adabas データマップをテストするには、データベース行のテストを実行します。

Adabas データマップの追加

Adabas データマップを追加する場合に、Adabas FDT とコピーブックをインポートできます。

Adabas データマップを追加する手順

1. リソースエクスプローラの【リソース】タブでデータマップフォルダを右クリックし、【データマップの追加】を選択します。
データマップを追加するウィザードが起動します。
2. 【名前】ページで以下の情報を入力します。
 - スキーマ名を入力します。
 - アクセス方式に【ADABAS】を選択します。
 - データマップの名前を入力します。
 - 【レコード定義のインポート】と【キーフィールド/FDT のインポート】チェックボックスをオンにします。
 【次へ】をクリックします。
3. 【ADABAS アクセス方式】ページで、ソース情報を入力し、【完了】をクリックします。
4. 【Adabas FDT インポート】ダイアログボックスで、Adabas ソースシステムのノード名を選択し、【OK】をクリックします。
ソースシステムが z/OS であり、PowerExchange でセキュリティが設定されている場合は、さらに有効な z/OS ユーザー ID とパスワードを入力する必要があります。Adabas ファイルセキュリティを使用している場合は、ファイルパスワードを入力します。
5. 【コピーブックのインポート-ソースの詳細】ページで、次のいずれかのコピーブックタイプを選択し、選択したタイプに必要な追加情報を入力します。
 - ADACMP

- **COBOL**
- **DDM**。FDIC データベース ID とファイル番号の値も入力します。
- **FDT**
- **PL/1**
- **PREDICT**。FDIC データベース ID とファイル番号の値も入力します。
- **TEXT**。【ローカル】をソースに選択した場合にのみ利用可能です。

デフォルトは **DDM** です。

【次へ】をクリックします。

6. 【コピーブックのインポート-リモートの詳細】 ページで、リモートコピーブックの情報を入力します。
【DDM】 または 【PREDICT】 ソースタイプを選択した場合は、【名前】 ボックスに名前を入力するか、【参照】 ボタンをクリックしてファイルを参照します。
【次へ】 をクリックします。
7. 【コピーブックのインポート - 構成の詳細】 ページで、インポートしたレコード、フィールド、およびテーブルに対する操作を選択し、【次へ】 をクリックします。
詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。
8. **コピーブック情報のインポート**ウィンドウで、インポート処理の情報を確認し 【OK】 をクリックします。
9. 【レコード定義】 ダイアログボックスで、インポートした各レコードに対するアクションを選択し、【OK】 をクリックします。

詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。

インポートウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: コピーブックメッセージログウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

関連項目：

- [「Adabas データマップへの FDT のインポート」 \(ページ 148\)](#)
- [「データマッププロパティ - ADABAS アクセス方式」 \(ページ 352\)](#)
- [「\[コピーブックのインポート - ソースの詳細\] ダイアログボックス」 \(ページ 149\)](#)
- [「\[コピーブックのインポート - リモートの詳細\] ダイアログボックス」 \(ページ 155\)](#)

Adabas データマップ内のテーブルへのキーの追加

Adabas データマップに Adabas キーをインポートした場合は、データマップの各テーブルに対してキーを定義できます。

Adabas データマップ内のテーブルにキーを追加する手順

1. データマップを開きます。
2. リソースエクスプローラの【データマップ】 タブで、テーブルを右クリックして【プロパティ】 をクリックします。
3. 【テーブルプロパティ】 ダイアログボックスで、【キー】 タブをクリックします。
4. 【プライマリキー】 リストで、テーブルに対するキーを選択します。
選択したキーが【キー詳細】 リストに表示されます。

キーの有効性は、記述子が必要とする Adabas フィールド名があるものとしてレコード内で定義されたすべてのフィールドがテーブルに含まれていることを確認することで決定されます。このフィールドのいずれかをグループにマッピングする場合は、グループのすべてのメンバが存在している必要があります。垂直配列内のフィールドは無効です。

5. **[OK]** をクリックします。

Datacom データマップ

CA Datacom データマップを追加するとき追加した後にコピーブックをインポートできます。

COBOL、DATACOM または PL/I コピーブックを Datacom データマップにインポートできます。

Datacom データマップを追加した後に、エレメント、キー、およびキーセグメントをデータマップに追加できます。

Datacom データマップをテストするには、データベース行のテストを実行します。

関連項目：

- [「Datacom データマップの項目」 \(ページ 64\)](#)

Datacom データマップの追加

Datacom データマップを追加する場合、コピーブックをインポートできます。

Datacom データマップを追加する手順

1. リソースエクスプローラの **[リソース]** タブで、**[追加] - [データマップ]** をクリックします。
2. **[名前]** ダイアログボックスに以下の情報を入力します。
 - **[DATACOM]** アクセス方式を選択します。
 - **[レコード定義のインポート]** オプションを選択します。
3. **[次へ]** をクリックします。
4. **[DATACOM アクセス方式]** ダイアログボックスで、アクセス方法に関する情報を入力し、**[完了]** をクリックします。
5. **[コピーブックのインポート-ソースの詳細]** ダイアログボックスで、**[リモート]** を選択します。

次のうち 1 つのコピーブックを選択します。

- **COBOL**
- **DATACOM**
- **PL/1**

デフォルトは **COBOL** です。

6. **[次へ]** をクリックします。
7. コピーブックタイプの **[コピーブックのインポート-リモートの詳細]** ダイアログボックスに、接続情報を入力します。

DATACOM コピーブックタイプに、2 つのユーザー ID とパスワードを入力します。これらは PowerExchange と CA Datacom/DB データベースの構成に必要です。最初のユーザー ID とパスワードは、DBMOVER コンフィギュレーションファイルの SECURITY 文を使用することによってセキュリティが

有効になる場合に必要です。【ディクショナリのユーザ ID】および【ディクショナリのパスワード】の値は、CA Datacom/DB がこれらの値を要求する場合に必要になります。

メタデータのあるコピーブックをインポートするには、【コピーブック】を選択して、【COBOL】または【PL/1】を選択します。

【テーブルの状態】、【キー】、【エレメント】 オプションは、デフォルトで無効になっています。

8. 【次へ】をクリックします。
9. 【コピーブックのインポート - 構成の詳細】 ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するアクションを選択し、【次へ】をクリックします。詳細については、[「Adabas データマップの追加」 \(ページ 111\)](#)を参照してください。
10. コピーブック情報のインポートウィンドウで、インポートの情報を確認し 【OK】 をクリックします。
11. 【レコード定義】 ダイアログボックスで、インポートした各レコードに対するアクションを選択し、【OK】 をクリックします。詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。

インポートウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: コピーブックメッセージログウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

関連項目：

- [「Datacom レコードへのエレメントの追加」 \(ページ 71\)](#)
- [「Datacom レコードへのキーの追加」 \(ページ 72\)](#)
- [「Datacom レコードへのキーセグメントの追加」 \(ページ 74\)](#)
- [「データマッププロパティ - DATACOM アクセス方式」 \(ページ 354\)](#)
- [「\[コピーブックのインポート - リモートの詳細\] ダイアログボックス」 \(ページ 155\)](#)
- [「\[コピーブックのインポート - リモート Datacom の詳細\] ダイアログボックス」 \(ページ 156\)](#)

DB2 カタログデータマップ

DB2 はリレーショナルデータベースのため、DB2 データマップは PowerExchange で DB2 テーブルにアクセスするためには必要ありません。

ただしデータマップを定義すると、定義しない場合では不可能な方法でデータを処理できることがあります。詳細については、[「DB2 データマップの使用例」 \(ページ 115\)](#)を参照してください。

次の DB2 データソースに対してデータマップを追加できます。

- DB2 for i5/OS
- DB2 for Linux, UNIX, and Windows
- DB2 for z/OS

DB2 データマップをテストするには、データベース行のテストを実行します。

注: マージした式カラムを含む DB2 データマップ用の BI カラムと CI カラムでキャプチャ登録を作成しないでください。BI カラムまたは CI カラムで式カラムを登録しようとすると、BI カラムに、操作前ではなく操作後の画像データが入ります。CI カラムにも、正しくない値が含まれる可能性があります。

DB2 データマップの使用例

次の使用例では、DB2 データマップを使用できる方法の一部について説明します。

文字カラムに格納されたパック 10 進数データ

一部の DB2 テーブルでは、カラムタイプと整合性がない可能性のある形式でフィールド配列を格納するために、単一の DB2 カラムが使用されます。例えば、複数のパック 10 進数フィールドが VARCHAR カラムに含まれている場合もあります。

この問題を解決するには、レコード定義のインポートでデータマップを作成し、問題の VARCHAR カラムを GROUP フィールドに変更して、子パック 10 進数フィールドの配列を追加します。次に、数値データをカラム化してターゲットの数値カラムにロードします。

データ内の値に依存するコードページ

企業が複数の海外向けサイトから単一の DB2 サブシステムにデータを統合するとします。DB2 カタログはデータが単一のコードページに格納されていることを示しますが、実際には、データが複数のコードページに格納され、コードページはデータ内の値に依存しています。更新用アプリケーションはクライアントおよびサーバーの CCSID を同じ値に設定するため、DB2 の変換はオフになります。

コードページを正しく管理するには、レコード定義のインポートでデータマップを作成します。次に、必要に応じてレコード ID 条件を追加してコードページを設定します。データマップのプロパティに 1 回または複数回 (文字フィールドごとに 1 回)、コードページを設定できます。レコード ID 条件を追加してコードページのタイプごとにコードページを設定する手順を繰り返します。

ユーザー定義関数で呼び出される展開ルーチン

DB2 の圧縮テーブルスペース機能を使用する代わりに、企業は社内の圧縮および暗号化のルーチンを使用してデータを格納するとします。

データの展開および復号化の問題を解決するには、レコード定義のインポートでデータマップを作成し、ユーザー定義関数の追加で問題の DB2 カラムを展開して、ユーザー定義カラムを PowerCenter マッピングに含めます。

16 進数の'0'値を含む文字データ

PowerCenter ワークフローは、DB2 リレーショナルソースから読み取った文字データを、最初の 16 進数の'0'値で切り詰めます。場合によっては、ワークフローで 16 進数の'0'値を保持し、完全なカラム文字列をターゲットに書き込むとします。

この目標を達成するには、レコード定義のインポートで DB2 データマップを定義します。次に、以下のアクションを実行します。

- Powercenter Workflow Manager で、[設定オブジェクト] タブの [カスタムプロパティ] フィールドに PreserveLowValues=Yes を指定します。
- 統合サービスマシンの DBMOVER 構成ファイルに LOWVALUES=Y を指定します。

DB2 カタログデータマップの追加

DB2 カタログデータマップを追加する場合、コピーブックをインポートできます。

DB2 カタログデータマップを追加する手順

1. リソースエクスプローラの [リソース] タブで、[追加] - [データマップ] をクリックします。
2. [名前] ダイアログボックスに以下の情報を入力します。
 - DB2 アクセス方式を選択します。
 - [レコード定義のインポート] オプションを選択します。
3. [次へ] をクリックします。

4. **【DB2 アクセス方式】** ダイアログボックスで、アクセス方式に関する情報を入力し、**【完了】** をクリックします。
5. **【コピーブックのインポート-ソースの詳細】** ダイアログボックスで、**【リモート】** を選択します。
【型】 リストで、**【DB2 カタログ】** コピーブックタイプを選択します。
6. **【次へ】** をクリックします。
7. **【コピーブックのインポート - リモート DB2 カタログの詳細】** ダイアログボックスに、次の情報を入力します。
 - オプションとして、インポートの詳細を変更することもできます。入力フィールドは、前回の選択から入力されます。
 - **【名前を付けてローカルに保存】** ボックスに、ファイル名を入力します。
 - 特定の列を選択するには、**【カラムのプレビュー/変更】** をクリックします。
8. **【次へ】** をクリックします。
9. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するアクションを選択し、**【次へ】** をクリックします。詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。
10. **コピーブック情報のインポート**ウィンドウで、インポートの情報を確認し **【OK】** をクリックします。
11. **【レコード定義】** ダイアログボックスで、インポートした各レコードに対するアクションを選択し、**【OK】** をクリックします。詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。

DB2 カタログインポートウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: **コピーブックメッセージログ**ウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

関連項目：

- [「データマッププロパティ - DB2 アクセス方式」 \(ページ 355\)](#)
- [「【コピーブックのインポート - リモート DB2 カタログの詳細】 ダイアログボックス」 \(ページ 158\)](#)

DB2 データマップでレコードレイアウトを編集

DB2 データマップで DB2 テーブルのレコードレイアウトを編集できます。

例えば、3 つの項目を含むレコードのフィールドを編集して、フィールドのデータを子フィールドに分割することができます。

この例では、MISC_DATA フィールドが CHAR(15)フィールドとして定義されています。ただし、フィールドは、それぞれ精度が 9 で位取りが 2 の、3 つのパックされた 10 進数フィールドを格納するために使用されません。

DB2 データマップでレコードレイアウトを編集する手順

1. データマップを開きます。
2. レコードを開きます。
3. **レコード**ウィンドウで、MISC_DATA フィールドを右クリックし、**【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。
4. **【フィールドの型】** で、GROUP を入力します。
5. **【OK】** をクリックします。

- レコードウィンドウで、MISC_DATA フィールドを右クリックし、[フィールドを子として追加] をクリックします。
- MISC_1、MISC_2、MISC_3 フィールドを、MISC_DATA フィールドの子として追加します。各フィールドを PACKED フィールドとして精度 9、スケール 2 で定義します。

DB2 アンロードファイルデータマップ

DB2 テーブルのダイレクトデータマッピングだけでなく、DB2 アンロードファイルをマッピングすることもできます。

DB2 アンロードファイルについては、PowerExchange では次の IBM および BMC ソフトウェアの形式がサポートされています。

- DB2 for z/OS オンライン REORG TABLESPACE ユーティリティ (UNLOAD EXTERNAL)
- DB2 for z/OS オンライン UNLOAD ユーティリティ
- DB2 for z/OS サンプルアンロードプログラム DSNTIAUL
- BMC ソフトウェアアンロードプラス
- Linux、UNIX、および Windows 用 DB2 の高性能アンロードユーティリティ

データベース行のテストを実行して DB2 のアンロードファイルデータマップをテストするときは、NRDB データソースタイプを選択します。

注: DB2 アンロードデータマップを追加し、[DSNUTILB UNLOAD] を [アンロードの種類] に対して選択する場合、PowerExchange にはレコードの OBID に関するフィールドが含まれます。DB2 UNLOAD 制御文で HEADER NONE を指定する DB2 アンロードファイルを作成する場合、[UNDEFINED] を [アンロードの種類] に対して選択します。

注: PowerExchange は、LARGE 形式のシーケンシャルデータセットを読み取ることができません。アップロードファイルの作成時に DD 文で DSNTYPE=LARGE を指定すると、PowerExchange はそのファイルを読み取ることができません。代わりに、DD 文で DSNTYPE=EXTREQ を指定できます。

DB2 アンロードファイルデータマップの追加

DB2 アンロードファイルデータマップを追加する場合、コピーブックをインポートできます。

DB2 アンロードファイルデータマップを追加する手順

- リソースエクスプローラの [リソース] タブで、[追加] - [データマップ] をクリックします。
- [名前] ダイアログボックスに以下の情報を入力します。
 - DB2UNLD アクセス方式を選択します。
 - [レコード定義のインポート] オプションを選択します。
- DB2UNLD アクセス方式ダイアログボックスに、以下の情報を入力します。
 - [ファイル名] ボックスに、ターゲットとなる DB2 アンロードファイル名を入力します。
 - [DB インスタンス] ボックスで、DSN9 などの DB インスタンスを入力します。
 - [テーブル名] ボックスで、テーブル名を入力するか、[参照] ボタンをクリックしてファイルを参照します。[DB2 テーブルフィルタ] ダイアログボックスで、クリエータ名かスキーマ名、データベース

名、またはテーブル名に基づくフィルタ条件を入力します。【次へ】をクリックします。テーブルを選択して【完了】をクリックします。

- **【アンロードの種類】** リストで、アンロードタイプを選択します。アンロードファイルへの一括アクセスに対しては、次の IBM および BMC の形式の、.REORG UNLOAD、.DSNTIAUL/BMC UNLOAD +、.DSNUTILB UNLOAD、.UNDEFINED がサポートされています。デフォルトでは、UNDEFINED タイプは NULL インジケータと共に生成され、カウントフィールドは REORG UNLOAD EXTERNAL タイプと同じポジションになります。これらのアンロードタイプを使用する代わりに、これらの文を特定の要件に合うように変更することができます。ユーザー定義のアンロード形式に対応するようにこれらの文を移動または削除できますが、PowerExchange では適用するデータフィールドにこれらの文が先行する場合に文を解釈します。
4. **【完了】** をクリックします。
 5. **【コピーブックのインポート-ソースの詳細】** ダイアログボックスで、**【リモート】** を選択して、**【DB2 カタログ】** のコピーブックタイプを選択します。
 6. **【次へ】** をクリックします。
 7. **【コピーブックのインポート - リモート DB2 カタログの詳細】** ダイアログボックスに、次の情報を入力します。
 - **【名前を付けてローカルに保存】** ボックスに、ファイル名を入力します。
 - **【Null インジケータ】** ボックスで、マッピングするアンロードファイルで使用する Null インジケータの 16 進値を入力します。
 - マッピングするアンロードファイルで変数が埋め込まれる場合は、**【変数のパディング】** オプションを選択します。このオプションは、BMC UNLOAD PLUS FIXEDVARCHAR オプションに対応しています。
 - 特定の列を選択するには、**【カラムのプレビュー/変更】** をクリックします。
 8. **【次へ】** をクリックします。
 9. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するアクションを選択し、**【次へ】** をクリックします。詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#) を参照してください。
 10. **コピーブック情報のインポート** ウィンドウで、インポートの情報を確認し **【OK】** をクリックします。
 11. **【レコード定義】** ダイアログボックスで、インポートした各レコードに対するアクションを選択し、**【OK】** をクリックします。詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#) を参照してください。

DB2 カタログインポート ウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログ ウィンドウに、インポート処理の結果が表示されます。

注: **コピーブックメッセージログ** ウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

関連項目：

- [「データマッププロパティ - DB2UNLD アクセス方式」 \(ページ 355\)](#)
- [「【コピーブックのインポート - リモート DB2 カタログの詳細】 ダイアログボックス」 \(ページ 158\)](#)

複数のテーブルから DB2 アンロードデータへアクセス

複数のテーブルのテーブルスペースは、複数のテーブルからのデータを含む単一のアンロードファイルになります。

単一のアンロードファイルで複数のテーブルにアクセスできるデータマップを作成できます。

テーブルの OBID を使用して、アンロードファイルから個々のテーブルを選択できます。次のアンロードファイルのタイプは、このタイプの複数テーブルアクセスに対してサポートされています。

- タイプ 1REORG UNLOAD

OBID は、各レコードの 4 番目と 5 番目のバイトに保持されます。

- タイプ 3DSNUTILB UNLOAD

OBID は、各レコードの最初の 2 バイトに保持されます。

次のタスクを実行して、同じアンロードファイルの複数のテーブルのアクセスに使用するデータマップを作成します。

1. データマップを追加します。
2. 基本テーブルを追加します。
3. データマップで各レコードタイプに基本テーブルを指定するには、DTL__OBID フィールドに対して適切な OBID を **【レコードのプロパティ】** ダイアログボックスに入力します。

DB2 アンロードファイルデータマップの追加 - 例

この例では、タイプ 3DSNUTILB UNLOAD ファイルを使用する方法について示します。同じ手順をタイプ 1REORG UNLOAD ファイルに使用できます。

アンロードファイルには、PWXTAB3 および PWXTAB4 テーブルからのデータが含まれます。それぞれのテーブルは独自の構造体になっています。タイプ 3 に対しては、基本テーブルは最初の 2 バイトで OBID によって指定されます。テーブルの OBID は、それぞれ x'0004' と x'0005' です。

DB2 アンロードファイルデータマップを追加する手順

1. データマップを追加して、PWXTAB3 テーブルをマッピングします。
2. **リソースエクスプローラ**の **【データマップ】** タブで、DTL_OBID フィールドを右クリックして **【プロパティ】** をクリックします。
3. **【レコード ID 値】** ボックスで、x'0004' の OBID16 進値を入力し、**【OK】** をクリックします。
DTL_OBID フィールドのアイコンが緑で表示されます。**【レコード ID】** フィルタがフィールドに設定されていることを示します。
4. 次の基本テーブル PWXTAB4 をインポートします。
5. **【レコード ID 値】** ボックスで、x'0005' の OBID16 進値を入力し、**【OK】** をクリックします。
DTL_OBID フィールドのアイコンが緑で表示されます。**【レコード ID】** フィルタがフィールドに設定されていることを示します。
6. PWXTAB4 テーブルをテストするには、データベース行のテストを実行します。
【データベース行のテストによる出力】 ウィンドウに、x'0005' の OBID がある 2 つの行が表示されます。

IDMS データマップ

COBOL、IDMS または PL/I コピーブックを、IDMS データマップを追加するときか追加した後にインポートできます。

IDMS データマップを追加した後に、CALC エレメント、オーナーレコード、セットを追加できます。

データベース行のテストを実行して IDMS データマップをテストします。

関連項目：

- [「IDMS データマップの項目」 \(ページ 65\)](#)

IDMS データマップの追加

この例では、IDMS データマップを追加して、コピーブックをデータマップにインポートする方法について示します。

PowerExchange では、z/OS からメタデータをダウンロードして、CA IDMS/DB 用のデータマップを作成します。セキュリティレベル 0 または 1 で実行する場合、PowerExchange リスナで実行しているユーザー ID のプレフィックスが付けられた z/OS 上で、データセットが作成されます。このユーザー ID が許容可能でない場合は、TEMPHLQ 文を DBMOVER コンフィギュレーションファイルに含めます。

IDMS データマップを追加する手順

1. リソースエクスプローラの **[リソース]** タブで、**[追加] > [データマップ]** をクリックします。
2. **[名前]** ダイアログボックスに以下の情報を入力します。
 - **IDMS** アクセス方式を選択します。
 - **[レコード定義のインポート]** オプションを選択します。
3. **[次へ]** をクリックします。
4. **[IDMS アクセス方式]** ダイアログボックスで、アクセス方式に関する情報を入力し、**[完了]** をクリックします。
5. **[コピーブックのインポート-ソースの詳細]** ダイアログボックスで、**[リモート]** を選択して、次のコピーブックタイプを選択します。
 - **COBOL**
 - **IDMS**
 - **PL/1**デフォルトは **COBOL** です。
注: COBOL および PL/I のインポートが IDMS で許可されます。ただし、IDMS レコードナビゲーション情報を手動で追加する必要があります。
6. **[次へ]** をクリックします。
7. コピーブックタイプに関する **[コピーブックのインポート-リモートの詳細]** ダイアログボックスに、情報を入力します。
注: IDMS パラメータは、インストール構成および標準によって決定されます。
8. **[IDMS]** コピーブックタイプを選択した場合は、**[コピーブックのインポート-リモート IDMS の詳細]** ダイアログボックスで **[詳細]** をクリックします。
[IDMS 詳細プロパティ] ダイアログボックスで、IDMS メタデータを取得するための追加情報を入力します。詳細については、[「\[IDMS 詳細プロパティ\] ダイアログボックス」 \(ページ 160\)](#) を参照してください。
[OK] をクリックします。
9. **[完了]** をクリックします。
10. **[OK]** をクリックします。
[OK] をクリックした場合、PowerExchange Navigator で、**[場所]** リストのノード名で指定された z/OS システム上の PowerExchange リスナと通信します。PowerExchange リスナでバッチジョブ IDMSMJCL がサブミットされ、IDMS メタデータがインポートされます。IDMSMJCL は z/OS システム上の RUNLIB ライブラリのメンバです。

11. **【次へ】** をクリックします。

【コピーブックのインポート - 詳細の設定】 ダイアログボックスが表示されます。

12. コピーブックの設定の詳細を入力します。

IDMS ログベースの変更データキャプチャが、単一のレコードレベルで動作します。データマップが CDC 処理に使用される場合は、**【インポートされたレコードごとにテーブルを作成する】** オプションを選択します。

また、バルクデータ移動および CDC 処理の両方について、インポートされたレコードごとにテーブルを作成し、テーブル間のセットのリレーションを保持する式を作成できます。それには、以下のオプションを選択します。

- **インポートされたレコードごとにテーブルを作成する**
- **IDMS セットのリレーションの式を作成する**

または、一部のバルクデータ移動では、階層パスごとに複雑なテーブルを作成することもできます。それには、**【IDMS 階層パスのテーブルを作成する】** を選択します。

注意: このオプションを選択すると、各行について階層内の最下位の子まで親データが繰り返されるため、データの冗長化が過度になることがあります。また、CDC ワークフローで、これらの階層パスを正しく管理できません。データリレーションを保持するには、ほとんどの場合、単一レコードのテーブルを作成し、IDMS セットのリレーションの式を作成することをお勧めします。

詳細については、[「【コピーブックのインポート - 構成の詳細】 ダイアログボックス」 \(ページ 161\)](#)を参照してください。

13. **【完了】** をクリックします。

14. **【OK】** をクリックします。

15. **【レコード定義】** ダイアログボックスで、**【適用】** をクリックします。

16. COVERAGE レコードで、**【OK】** をクリックします。

関連項目：

- [「IDMS レコードへ CALC エLEMENT の追加」 \(ページ 75\)](#)
- [「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)
- [「リモートシステムへのデータマップの送信」 \(ページ 95\)](#)
- [「【コピーブックのインポート - リモートの詳細】 ダイアログボックス」 \(ページ 155\)](#)
- [「【IDMS 詳細プロパティ】 ダイアログボックス」 \(ページ 160\)](#)
- [「データマッププロパティ - IDMS アクセス方式」 \(ページ 361\)](#)

IDMS レコードプロパティの編集

IDMS メタデータをインポートした後に IDMS レコードプロパティを変更できます。

IDMS レコードプロパティを編集する手順

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックスが表示されます。
3. **【名前】** タブで、レコードプロパティを定義します。
4. **【オーナーレコードとセット名】** タブで、オーナーレコードとセットに対してプロパティを定義します。

関連項目：

- [「IDMS レコードへ CALC エLEMENT の追加」 \(ページ 75\)](#)
- [「IDMS レコードへのオーナーレコードとセットの追加」 \(ページ 75\)](#)
- [「IDMS データマップの項目」 \(ページ 65\)](#)
- [「レコードプロパティ - 名前 \(IDMS\)」 \(ページ 377\)](#)

IDMS レコードへの式の追加

IDMS データマップレコードの追加情報を返すには、式で 1 つ以上の PowerExchange 関数を使用してソースデータを処理するユーザー定義フィールドを追加します。

これらの関数により、フィールドの分割や連結などの操作を行うことができます。

次の関数を呼び出して、CA IDMS/DB ソースデータを取得することもできます。

- GetDatabaseKey
- GetDataFlowType
- GetDbKeyOfFirstMember
- GetDbKeyOfLastMember
- GetDbKeyOfNextMember
- GetDbKeyOfOwner
- GetDbKeyOfPriorMember
- GetFullDbKey
- GetFullDbKeyOfFirstMember
- GetFullDbKeyOfLastMember
- GetFullDbKeyOfNextMember
- GetFullDbKeyOfOwner
- GetFullDbKeyOfPriorMember
- GetPageGroup
- GetPageGroupOfOwner
- GetPgGrpAndRdx
- GetPgGrpAndRdxOfOwner
- GetPgGrpOfOwner
- GetRadix
- GetRadixOfOwner
- GetSeqWithinLevel
- GetSeqWithinParent

データウェアハウスなどのリレーショナル構造体を作成するには、CA IDMS/DB ソースデータから式を使用します。データベースキーを返す IDMS 関数は、バルクデータと CDC ソースデータ両方の結果を返します。

また、ユーザー定義プログラムを呼び出す式で C allProg 関数を使用して、ソースデータを処理することもできます。

関連項目：

- [「レコードへのユーザー定義フィールドの追加」 \(ページ 70\)](#)
- [「GetDataFlowType」 \(ページ 313\)](#)
- [「GetDbKeyOfOwner」 \(ページ 315\)](#)
- [「IMS レコードへの式の追加」 \(ページ 128\)](#)
- [「ユーザー定義フィールド用 PowerExchange 関数」 \(ページ 303\)](#)

IMS データマップ

IMS データベースをデータソースまたはターゲットとして使用するには、IMS データマップを定義します。セッションの処理中に IMS データにリレーショナル方式でアクセスする SQL 文を作成するには、データマップが必須です。

IMS のソースまたはターゲットのデータベースにデータマップを作成するときに、1 つ以上の IMS セグメントを追加し、セグメントのフィールドにデータの物理レイアウトを定義します。データマップにはデータのテーブルビューも含まれます。セグメントおよびフィールドの定義では、データの非リレーショナルビューを指定し、そのテーブルおよび列にデータのリレーショナルビューを指定します。

注: IMS データソースに対して、レコードではルートセグメントと子セグメントを参照します。

データマップには 1 つのセグメントを含めることも複数のセグメントを含めることもできます。

- 関連するデータファイルに 1 つのセグメントタイプが含まれている場合は、単一のセグメントをデータマップに定義します。単一セグメントデータマップでは、複数のテーブルを定義できます。
- ソースデータベースに複数のセグメントタイプが含まれている場合は、複数のセグメントをデータマップに定義します。PowerExchange で取得する列が少なければ少ないほど、ファイル処理とデータ抽出が速くなります。

フィールドの配列、日付の変換、データのフィルタリング、データの拡張にデータマップを使用すると、データの精度が向上しデータ容量が減少します。ルックアップトランスフォーメーションにデータマップを使用して、関連する値を検索し、IMS ターゲットに行が存在するかどうかを判断することもできます。

IMS データマップを定義するには、次のタスクを完了します。

1. IMS データマップを追加します。

データマップを定義する場合、**DL/1 BATCH** または **IMS ODBA** のアクセス方式を選択し、レコードタイプとデータを定義するデータファイルにデータマップを関連付けます。アクセス方式に必要な IMS のソースまたはターゲットのプロパティも入力する必要があります。DL/I バッチアクセスには PCB 番号が必須です。ODBA アクセスには、IMS SSID、PSB 名、および PCB 名が必須です。

注: PowerCenter セッションを構成するかまたはデータベース行のテストを実行すると、データマップのアクセス方式をオーバーライドできます。アクセス方式をオーバーロードする場合は、新しいアクセス方式に必要なプロパティも入力する必要があります。

2. DBD ソースが IMS データで使用可能な場合は、DBD ソースをデータマップにインポートします。

DBD ソースでは、セグメントおよびデータマップの階層シーケンスを定義します。インポート処理時に PowerExchange は、IMS セグメントとフィールドをテーブルにマップし、PowerExchange で必要なソースデータのリレーショナルビューを作成します。

3. 各セグメントを COPYLIB でオーバーレイするには、COBOL コピーブックをデータマップの IMS セグメントそれぞれにインポートします。

4. ローカルまたはリモートのノードにある PowerExchange リスナにデータマップを送信して PowerExchange にデータマップを格納します。
このアクションは、PowerExchange が抽出処理のためにアクセスする OS に依存しないファイルにデータマップを変換します。
5. データベース行のテストを実行してデータマップをテストします。
データベース行のテストでは実際のソースデータにアクセスしてテーブル形式で表示し、ソースデータベースからデータにアクセスできることを確認します。

PowerExchange のインストールには IMSDEMOS ファイルのサンプルが含まれており、そのサンプルには IMS データマップのサンプルの作成に使用できる IMS DBD ソースのサンプルが含まれています。

IMS データマップを追加するための前提条件

IMS データマップを追加する前に、IMS データベースに関する情報を収集する必要があります。

- IMS ソースまたはターゲットに関する以下の情報を収集します。
 - ODBA アクセスおよび IMSUNLD アクセスを使用してファイルをアンロードするために必要な IMS SSID
 - IMS データベース名とデータベースのデータセット名
 - DBD データセットの名前と場所
 - PSB 名
 - IMS DL/I バッチまたは BMP を使用して IMS データにアクセスする場合は、PSB の PCB の数を数えて PCB 番号を決定する。PSB が COMPAT=YES と設定されている場合は、その数に 1 を加える。
 - IMS ODBA アクセスを使用する場合は、PCB 名を記録する
 - IMS セグメントの名前および可変長のセグメント（存在する場合）。さらに、DBD ソースで定義されているように階層シーケンスが指定されていることに留意する
- ロックの競合数を減らすため、IMS データベースの PCB の PROCOPT を読み取り専用に変更します。
- PSB を IMS ODBA インタフェースまたは Netport BMP ジョブで使用する場合に、IMS SYSGEN で PSB を定義します。
- RUNLIB ライブラリの DBMOVER 構成メンバを更新します。
- PowerExchange Navigator システム上の dbmover.cfg 構成ファイルを更新します。
- IMS セグメントに INSERT または UPDATE を実行する場合は、データマップが IMS DBD で定義されたセグメント長全体を表す必要があります。そうしないと、このセグメントに INSERT または UPDATE を実行した場合に、FILLER と定義されていないセグメントの最後に非ブランクデータが書き込まれる場合があります。この問題を回避するには、PowerExchange にインポートする前に、COPYLIB に FILLER 定義を追加します。

DL/I バッチまたは BMP を使用した IMS データへのアクセスの構成

DL/I バッチまたは BMP インタフェースを使用して IMS のソースデータおよびターゲットデータにアクセスするには、z/OS システム上でいくつかの PowerExchange 構成タスクを実行する必要があります。

IMS データマップで **DL/1 BATCH** アクセス方式を指定している場合や **IMS ODBA** アクセス方式をデータベース行のテストや PowerCenter バルクデータ移動セッション用の **DL/1 BATCH** にオーバーライドする場合に、PowerExchange は DL/I バッチアクセスまたは BMP アクセスを使用します。

DL/I バッチアクセスまたは BMP アクセスを使用するデータベース行のテストや PowerCenter セッションを実行する前に、以下のタスクを実行します。

- DL/I バッチまたは BMP の Netport ジョブとして動作するように、RUNLIB ライブラリの IMSJCL メンバをカスタマイズします。PowerCenter でバルクデータ移動セッションを実行するときに PSB 名をオーバーライドする場合は、%PSBNAME 置換変数を含めることができます。このセッションのオーバーライドで%PSBNAME 変数を使用すると、同一の Netport JCL メンバを使用して複数の PSB にアクセスできます。
- BMP アクセスを使用する場合は Netport ジョブを APF 許可します。
- **DL/1 BATCH** アクセス方式を指定する IMS データマップを作成するか、または IMS データマップの **IMS ODBA** アクセス方式を **DL/1 BATCH** でオーバーライドします。PowerCenter セッションプロパティのアクセス方式を IMS ソースまたはターゲット用にオーバーライドしたり、データベース行テストの構成時にオーバーライドしたりできます。
- RUNLIB の DBMOVER 構成メンバに、各 PSB の NETPORT 文および対応する LISTENER 文を追加します。DBMOVER メンバには、最大 10 組の文を入力できます。

ヒント: 10 個を超える PSB を処理する必要がある場合にこの制限を超えないようにするには、PowerCenter セッションプロパティの Netport JCL および **IMS PSBNAME Override** の値に%PSBNAME を指定します。NETPORT 文および LISTENER 文の同一ペアを複数の PSB に使用できます。詳細については、『*PowerCenter 用の PowerExchange インタフェース*』を参照してください。

DBMOVER の変更を有効にするには、PowerExchange リスナを再起動する必要があります。

注: PowerCenter セッションまたはデータベース行テストの PSB 名と PCB 番号をオーバーライドすることもできます。

IMS ODBA を使用した IMS データへのアクセスの構成

IMS ODBA を使用して IMS データにアクセスするには、z/OS システムでいくつかの PowerExchange タスクや IMS 構成タスクを完了する必要があります。

IMS データマップでアクセス方式に **IMS ODBA** を指定している場合やデータベースの行テストまたは PowerCenter バルクデータ移動セッションの **DL/1 BATCH** アクセス方式を **IMS ODBA** にオーバーライドする場合、PowerExchange は IMS ODBA アクセスを使用します。

IMS ODBA クエリは PowerExchange リスナのサブタスクとして実行され、Listener ポートを使用します。

ODBA アクセスを使用するには、次の設定タスクを実行します。

- PowerExchange リスナの STEPLIB ライブラリを APF 許可する
- ODBA アクセス用に PowerExchange リスナの STEPLIB ライブラリにある RESLIB ライブラリを含めるか、または z/OS システムで LNKLST 連結を使用して RESLIB にアクセスできることを確認する
- 各 PSB に定義済みの PCBNAME が含まれることを確認する
- アクセス方式に **IMS ODBA** を指定する IMS データマップを作成するか、または IMS データマップの **DL/1 BATCH** アクセス方式を IMS ODBA でオーバーライドする。PowerCenter セッションプロパティのアクセス方式をオーバーライドすることも、データベース行テストの構成時にオーバーライドすることもできる。

注: PowerCenter セッションまたはデータベース行テストの IMS SSID、PSB 名、および PCB 名をオーバーライドすることもできます。

- RUNLIB の DBMOVER 構成メンバに、IMSID=ssid 文および ODBASUPP=YES 文を追加する。次に、PowerExchange リスナを再起動してこれらの DBMOVER の変更を有効にする
- システムの IMS Resource Recovery Services (RRS) がアクティブなことを確認する。IMS PROCLIB の DFSPBxxx メンバに、RRS=Y と指定する

- IMS ロードモジュールのアセンブルとリンク編集を行ってデータに ODBA アクセスを設定する。次の JCL のような JCL を使用する

```
//ULIA JOB (),CLASS=A,MSGCLASS=H,MSGLEVEL=(1,1),REGION=4M
//*
//STEP1 EXEC HLASMCL,
// PARM.C=('RENT')
//C.SYSLIB DD DISP=SHR,DSN=SYS1.MACLIB
// DD DISP=SHR,DSN=IMS710.ADFSMAC
//C.SYSIN DD *
DFSIMS70 CSECT
*
* THE DFSRPP MACRO GENERATES THE DRA STARTUP TABLE
* THERE ARE A FEW MORE POSSIBLE PARMS
DFSRRP DSECT=NO, GENERATE X
FUNCLV=1, FUNCTION LEVEL X
DDNAME=CCTLDD, DDNAME FOR DRA RESLIB X
DSNAME=IMS1210.SDFSRESL, DSNAME FOR DRA RESLIB X
DBCTLID=IMSC, DBCTL IDENTIFIER X
USERID=, USERID X
MINTHRD=2, MIN NUMBER OF THREADS X
MAXTHRD=7, MAX NUMBER OF THREADS X
TIMER=60, IDENTIFY TIMER VALUE DEFAULT X
FPBUF=5, NUMBER OF FP BUFFERS PER THREAD X
FPBOF=5, NUMBER OF FP OVERFLOW BUFFERS X
SOD=A, SNAP DATASET OUTPUT CLASS X
AGN=, APPLICATION GROUP NAME X
TIMEOUT=60, DRATERM TIMEOUT VALUE X
IDRETRY=0, IDENTIFY RETRY COUNT X
CNBA=10, TOTAL FP NBA BUFFERS FOR CCTL
END
//L.SYSLMOD DD DSN=ABC.A.LOAD(DFSIMSC0),
// DISP=SHR
```

L.SYSLMOD DD 文に "DFS" + DBCTLID_value + "0" の形式でモジュール名を指定する必要があります。このモジュール名は制御領域用の SSID と一致し、IMS データマップに指定する必要があります。この例の JCL ではモジュール名は DFSIMSC0 です。

PowerExchange dbmover.cfg ファイルを Windows 上で構成

IMS データマップを追加する前に、dbmover.cfg ファイルを Windows 上で更新して、IMS ソースデータにアクセスできるようにします。

構成の設定は、DL/1 バッチと IMS ODBA のどちらを使用して IMS にアクセスするかにより異なります。

Windows 上で PowerExchange dbmover.cfg ファイルを構成する手順

1. NODE 文を dbmover.cfg ファイルに追加して、リモートの IMS システムに次のようにアクセスします。
 - DL/1 バッチか BMP を使用して IMS にアクセスするには、z/OS 上の PowerExchange Listener に対して Netport ポートを指定する NODE 文を追加します。フォーマットは次のとおりです。

```
NODE=(location_name,TCPIP,[hostname_or_IPaddress],port_number)
```
 - IMS ODBA を使用して IMS にアクセスするには、z/OS 上の PowerExchange Listener ポートを指定する NODE 文を追加します。フォーマットは次のとおりです。

```
NODE=(location_name,TCPIP,[hostname_or_IPaddress],port_number)
```
2. PowerExchange Listener を再起動して、構成の変更を実装します。
注: IMS データにアクセスするために、Windows の PowerExchange Listener を起動する必要はありません。

IMS データマップの追加

IMS データマップを追加するには、スキーマ名、データマップ名、およびアクセス方式を指定する必要があります。データレイアウトを定義する場合は、データマップの作成時またはそれ以降に IMS DBD をインポートします。

1. リソースエクスプローラの [リソース] タブで、[追加] - [データマップ] をクリックします。
2. [名前] ダイアログボックスに、以下の情報を入力します。

フィールド	説明
スキーマ名	スキーマの名前です。
データマップ名	データマップのユーザー定義名です。
アクセス方式	ソースデータへのアクセスに次のいずれかのオプションを指定します。 <ul style="list-style-type: none">- DL/1 BATCH。サブミットした JCL を通して IMS データベースにアクセスします。アクセス方法は DL/1 バッチですが、Netport ジョブに対して DL/1 あるいは BMP を選択できます。- IMS ODBA。オープンソースのデータベース接続を通して IMS データベースに接続します。- SEQ。フラットファイルとして使用する IMS データベースにアクセスします。 注: データベース行テストや PowerCenter バルクデータ移動セッションを実行するときに、後で DL/1 BATCH または IMS ODBA アクセス方式をオーバーライドできます。
レコード定義のインポート	DBD をデータマップにインポートする場合はこのボックスを選択します。DBD のインポートでは、セグメントおよび対応するテーブルビューをキーフィールド、検索フィールド、CCK フィールドの定義とともにデータマップに追加します。セグメントの階層シーケンスも定義します。 DBD をインポートした後、COBOL または PL/I コピーブックをデータマップの各セグメントにインポートし、その COPYLIB でセグメントをオーバーレイできます。

3. [次へ] をクリックします。
4. [アクセス方式] ダイアログボックスに、選択したアクセス方式の情報を入力します。次に、[完了] をクリックします。
5. [レコード定義のインポート] を選択した場合は、IMS DBD およびセグメントのコピーブックをインポートします。

データマップが完成したら、データマップのテーブルでデータベース行のテストを実行し、データマップを使用してデータにアクセスできることを確認します。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)

IMS データの表示

IMS データマップでは、レコードでデータの非リレーショナルビューが表示され、テーブルでデータのリレーショナルビューが表示されます。

単一のセグメントに対してデータをインポートする場合、セグメントに関するレコードとテーブルビューがデータマップに表示されます。複数のセグメントのデータをインポートする場合は、複数のセグメントがデータマップに表示されます。

DBD ソースのインポートで作成された IMS 階層を表示することもできます。

IMS 階層の表示

DBD ソースのインポートで作成された IMS 階層を表示することができます。

IMS 階層を表示する手順

- ▶ リソースエクスプローラの [データマップ] タブでデータマップを右クリックし、[IMS 階層の表示] をクリックします。

IMS データマップの IMS 階層が表示されます。

IMS レコードへの式の追加

IMS データマップレコードの追加情報を返すには、式で 1 つ以上の PowerExchange 関数を使用してソースデータを処理するユーザー定義フィールドを追加します。

これらの関数により、フィールドの分割や連結などの操作を行うことができます。

次の関数を使用して IMS ソースデータを取得することもできます。

- GetDatabaseKey
- GetIMSRBAByLevel
- GetSeqWithinLevel
- GetSeqWithinParent

また、ユーザー定義プログラムを呼び出す式で C allProg 関数を使用して、IMS ソースデータを処理することもできます。

関連項目：

- [「IDMS レコードへの式の追加」 \(ページ 122\)](#)
- [「ユーザー定義フィールド用 PowerExchange 関数」 \(ページ 303\)](#)

IMS レコードの RBA の取得

IMS レコードの IMS RBA を取得するには、IMS データマップ内のレコードにユーザー定義フィールドを追加します。ユーザー定義フィールドでは、GetDatabaseKey 関数または GetIMSRBAByLevel 関数を式で使用します。

IMS データマップのセグメントデータにセグメント RBA を追加する場合、Informatica では、GetDatabaseKey 関数ではなく GetIMSRBAByLevel 関数を使用することをお勧めしています。GetIMSRBAByLevel 関数では、キーのない、または一意のキーが存在しない親セグメントの RBA を取得することができます。

バルクデータ移動操作または IMS 同期 CDC に使用される IMS データマップには GetIMSRBAByLevel 関数を使用します。

次のタイプの IMS データマップ内のレコードには、GetIMSRBAByLevel 関数を使用できません。

- IMS ログベース CDC に使用される IMS データマップ
- 次のタイプのレコードが含まれる IMS アンロードファイルへのアクセスに使用される IMS データマップ
 - HISAM データセットの任意のフォーマットのアンロードレコード
 - HDAM および HIDAM データセットの BMC SHORT または BMC XSHORT フォーマットのアンロードレコード
 - IMS 高速パスデータセットの標準 IBM フォーマットのアンロードレコード

GetIMSRBByLevel 関数が使用される次のタイプのデータマップでデータベース行のテストを実行するには、次のライブラリを APF 許可します。

- Netport ジョブに BMP と共に IMS DL/1 バッチデータマップを使用する場合は、BMP STEPLIB 連結内のすべてのライブラリを APF 許可する必要があります。このように処理しないと、PowerExchange によって戻りコード 12 でメッセージ PWX-02204 が発行されます。
- IMS ODBA データマップを使用する場合は、PowerExchange Listener の STEPLIB 連結内のすべてのライブラリを APF 許可する必要があります。このように処理しないと、PowerExchange が異常終了します。

関連項目：

- [「レコードへのユーザー定義フィールドの追加」 \(ページ 70\)](#)
- [「GetDatabaseKey」 \(ページ 312\)](#)

IMS データマップ内のプロパティの編集

IMS データマップのセグメントとテーブルを更新するには、DBD または COBOL コピーブックをデータマップに再インポートします。

または、IMS データマップでセグメントとテーブルのプロパティを手動で編集することもできます。

警告： データマップでセグメントとテーブルを手動で編集すると、データマップ内のデータが失われることがあります。

関連項目：

- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)
- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)
- [「レコードまたはセグメントへのコピーブックのインポート」 \(ページ 146\)](#)

IMS データマップをリモートシステムに送信

IMS データマップを格納するには、データマップを同じシステム上の PowerExchange Listener に IMS ソースデータとして送信します。

作成する各データマップは、関連するソースデータを PowerExchange で抽出できるように格納します。データマップを格納した場合、PowerExchange では抽出処理にアクセスできるように、OS に依存しないファイルにデータマップを変換します。

関連項目：

- [「リモートシステムへのデータマップの送信」 \(ページ 95\)](#)

ルックアップトランスフォーメーションを使用して、IMS データベースのデータを検索

IMS データベースでデータを検索するには、PowerCenter ルックアップトランスフォーメーションを使用します。

"GOX" PROCOPT で、PCB のルックアップトランスフォーメーションごとに別々の Netport ジョブを使用します。

注： SSA を IMS 内で効率的に実行できるように、セグメントの IMS キー値を指定します。

IMS データベースへのデータの書き込み

データマップのソースデータを IMS データベースに書き込むことができます。

IMS へのデータ書き込みは、PowerExchange Client for PowerCenter (PWXPC) を使用します。

データを IMS に書き込む場合は、次のガイドラインを参考にしてください。

- セグメントに対してデータを書き込むために必要なすべての処理が IMS 内で実行されるように、セグメントのキー値を指定します。
- SDEP セグメントは使用できません。
- キーのないセグメントの下の子セグメントは更新できません。
- OCCURS 句の各フィールドを別々のフィールドとして識別できるように COPYLIB が変更済みの場合は、その OCCURS 句を含むセグメントを使用できます。これにより、COPYLIB 内の OCCURS が不要になります。OCCURS 句のオカレンスごとに別々の行を生成することはできません。
- データマップを PowerCenter にインポートする場合、キーフィールドとして使用するすべての CCK フィールドがキーフィールドとしてマークが付けられていることを確認します。
- PowerCenter で、コミットの頻度を低くして、セグメントをロックする危険性を減らします。
- 別々の Netport ジョブを使用して IMS データを書き込みます。別々の Netport ジョブを使用することにより、書き込み目的で PSB にアクセスし、IEFRDER ログなどの IMS データを更新するために JCL を変更することができます。
- 次のフィールドタイプのうち 1 つに対してデータをデータベースに書き込む場合、データマップを追加するときにフィールドがオプションであるかどうかを示します。

- フィールドへのスペース

- フィールドへのデータなし

このタイプのフィールドが複数ある場合、データマップの最初のフィールドが選択でき、その後に続くすべてのフィールドがオプションであることを示します。フィールドがオプションであることを指定しない場合、PowerCenter セッションログにエラーが表示されます。

1. z/OS システムにアクセスできることを確認するには、ターゲットデータベースがある z/OS システムの IP アドレスに ping します。
2. dbmover.cfg 構成ファイルに、リモート z/OS システムを指定する NODE 文を含めます。
3. リモートの PowerExchange リスナが起動していることを確認するには、次のコマンドを実行します。

```
dtlrex loc=xxxx prog=ping uid=userid pwd={password|passphrase}
```


xxxx は、PowerExchange 構成ファイル dbmover.cfg の NODE 名です。
4. PowerExchange Navigator で、IMS データマップを追加してテストし、格納します。
5. PowerCenter を使用して、データを IMS に移動します。

テーブルの IMS オプションの表示または編集

【テーブルプロパティ】 ダイアログボックスの **【IMS オプション】** タブで定義されているオプションでは、データを IMS データベースに書き込む方法を示し、IMS で動作するときにパフォーマンスが最適になるようにデフォルト値を提供します。

これらのオプションは、PowerExchange に次のことを実行させるデフォルト値に設定されます。

- IMS をリレーショナルターゲットとして扱う
- IMS COBOL アプリケーションプログラムのように動作する

これらのオプションのデフォルト値を編集する場合、データベースに送信されるデータ量やデータの書き込みに必要な時間に影響が出ます。

IMS オプションを表示または変更する手順

1. IMS データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、テーブルを右クリックして **【プロパティ】** を選択します。
3. **【テーブルプロパティ】** ダイアログボックスの **【IMS オプション】** タブで、テーブルに対して IMS オプションを定義します。

警告: IMS 階層について十分な知識がない場合は、**【削除オプション】** オプションのデフォルト値を編集しないでください。その設定値を変更した場合、IMS データが失われることがあります。

関連項目：

- [「テーブルのプロパティ - IMS オプション」 \(ページ 394\)](#)

IMS データマップのテスト

データマップを追加して PowerExchange に格納した後にデータベース行のテストを実行して、データマップをテストしデータソースにアクセスできることを確認します。

データベース行のテストは実際のソースデータにアクセスされ、テーブル形式で表示されます。データベース行のテストでは、IMS アンロードファイルまたは IMS データベースからデータを読み取ることができます。

IMS データマップをテストするには、**【データベース行のテスト】** ダイアログボックスで、次のようにデータソースタイプを選択します。

- **IMS アンロードファイル。** **【DB タイプ】** リストで、**【IMSUNLD】** を選択します。
- **IMS データベース。** **【DB タイプ】** リストで、**NRDB** または **NRDB2** を選択します。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)

IMS データマップの例

次の例では、PowerExchange で提供する DBD ソースの例を使用して、単一セグメントデータマップと複数セグメントデータマップを作成する方法を示します。

PowerExchange では、次の DBD ソースのサンプルを提供します。

```
DBD  NAME=IMSDemo,ACCESS=(HDAM,VSAM),X
      RMNAME=(DFSHDC40,50,80),X
      EXIT=(*,KEY,PATH,(CASCADE,KEY,PATH),LOG)
DATASET DD1=IMSDemo,DEVICE=3380,SIZE=8192,SCAN=3
SEGMENT NAME=IMSDemo,BYTES=80FIELD NAME=(DTLKEY,SEQ),BYTES=9,START=3
FIELD NAME=RECTYPE,BYTES=1,START=12FIELD NAME=AMOUNT,BYTES=4,START=13
FIELD NAME=BINNO,BYTES=4,START=17FIELD NAME=DECNO,BYTES=3,START=21
FIELD NAME=DTYY,BYTES=2,START=24FIELD NAME=DTMM,BYTES=2,START=26
FIELD NAME=DTDD,BYTES=2,START=28FIELD NAME=ACCT1,BYTES=10,START=30
FIELD NAME=ACCT2,BYTES=10,START=40
FIELD NAME=ACCT3,BYTES=10,START=50
DBDGEN
FINISH
```

PowerExchange を z/OS にインストールした場合、PowerExchange では MVS Installation Assistant で指定される高レベル修飾子を使用してライブラリを作成します。

z/OS で次のジョブを実行して、インストール例のための IMS 環境を作成します。

```
hlq.DTLDEMO(IMSDDEF)
```

IMS 単一セグメントデータマップの追加 - 例

この例では、単一の IMS セグメントに対してデータマップを作成する方法を示します。

この例では、IMSDDEMOS のサンプルを使用し、DBD ソースファイルのサンプルをインポートしてセグメントおよびデータマップの階層シーケンスを定義します。COBOL コピーブックをセグメントに対してインポートすることもできます。この例では、DL/1 バッチアクセス方法を使用します。

単一セグメント IMS データマップを追加する手順

1. リソースエクスプローラの **[リソース]** タブで、**[追加] - [データマップ]** をクリックします。
2. **[名前]** ダイアログボックスに以下の情報を入力します。
 - **[スキーマ名]** ボックスに、「IMSS1」と入力します。
 - **[データマップ名]** ボックスに、「IMSM1」と入力します。
 - **[アクセス方式]** リストで、**[DL/1 BATCH]** を選択します。
 - **[レコード定義のインポート]** オプションを選択します。
3. **[次へ]** をクリックします。
4. **[DL/1 バッチアクセス方式]** ダイアログボックスで、データソースに関する情報を指定します。
IMS アンロードデータセットを読み取っている場合は、**[IMS SSID]** および **[DBD 名]** の値が必要です。
DL/1 バッチか BMP を実行する Netport ジョブを使用して IMS データベースを読み取っている場合は、**[PCB 番号]** の値が必要です。
5. **[完了]** をクリックします。
リソースエクスプローラの **[データマップ]** タブで、IMSS1.IMSM1 データマップが表示されます。
6. **[コピーブックのインポート - ソースの詳細]** ダイアログボックスで、以下の情報を入力します。
 - **[リモート]** オプションを選択します。
 - **[型]** リストで、**[DBD]** を選択します。
7. **[次へ]** をクリックします。
8. **[コピーブックのインポート - リモート DBD の詳細]** ダイアログボックスで、次の情報を入力します。
 - **[ファイル名]** ボックスに、インストール時に指定した高レベル修飾子 (HLQ) を使用して、PDS 名と DBD メンバを入力します。以下に例を示します。
hlq.DTLDEMO(IMSDBD)
 - **[場所]** リストから z/OS の場所を選択します。
 - **[場所]** リストに表示された場所は、PowerExchange Navigator システムの dbmover.cfg コンフィギュレーションファイルの NODE 文によって定義されます。
 - z/OS 上の PowerExchange Listener に対してセキュリティが実装されている場合は、z/OS のユーザー ID とパスワードを入力します。
 - **[名前を付けてローカルに保存]** ボックスに、「IMSDBD」と入力します。
9. **[次へ]** をクリックします。
10. **[コピーブックのインポート - 構成の詳細]** ダイアログボックスで、以下のオプションを選択します。
 - **[DL1 階層パス用テーブルを作成する]** オプションを選択します。
 - その他のデフォルト値を受け入れます。

詳細については、[「データマップへのコピーブックのインポート」](#) (ページ 144) を参照してください。
11. **[完了]** をクリックします。

12. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、選択を確認して **【完了】** をクリックします。
13. DBD ソースをインポートするには、**【OK】** をクリックします。
14. **【レコード定義】** ダイアログボックスで、**【インポート】** をクリックします。必要に応じて、レコードタイプをスキップするか、インポートを停止します。詳細については、[「レコードまたはセグメントへのコピーブックのインポート」 \(ページ 146\)](#) を参照してください。
15. インポートを完了するには、**【次の再定義】** か **【前の再定義】** をクリックして青い矢印で行を選択します。
16. 確認を求められた場合、**【インポートの再開】** をクリックして、インポートを再開します。
17. インポート処理のログ情報を表示します。

IMS 単一セグメントデータマップへのコピーブックのインポート - 例

この例では、コピーブックを IMS 単一セグメントデータマップにインポートする方法を示します。

コピーブックを IMS 単一セグメントデータマップにインポートする手順 - 例

1. データマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、IMS セグメントを選択します。
3. **【ファイル】 - 【コピーブックのインポート】** をクリックします。
4. **【コピーブックのインポート - ソースの詳細】** ダイアログボックスで、以下の情報を入力します。
 - **【リモート】** オプションを選択します。
 - **【型】** リストで、**【COBOL】** を選択します。
5. **【次へ】** をクリックします。
6. **【コピーブックのインポート-リモート DBD の詳細】** ダイアログボックスで、次の情報を入力します。
 - **【ファイル名】** ボックスに、コピーブック名と場所を入力します。
 - **【場所】** リストから z/OS の場所を選択します。

【場所】 リストに表示された場所は、PowerExchange Navigator マシンの dbmover.cfg コンフィギュレーションファイルの NODE 文によって定義されます。

 - z/OS 上の PowerExchange Listener に対してセキュリティが実装されている場合は、z/OS のユーザー ID とパスワードを入力します。
 - **【名前を付けてローカルに保存】** ボックスに、ファイル名を入力します。
7. **【次へ】** をクリックします。
8. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、デフォルトを受け入れて **【完了】** をクリックします。詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#) を参照してください。
9. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、選択を確認して **【完了】** をクリックします。
10. インポートプロンプトに応答します。
11. インポートが完了したときに、データを確認します。
12. データマップを格納します。
13. データベース行のテストを実行してデータマップをテストします。

IMS 複数セグメントデータマップの追加 - 例

この例では、複雑なテーブルを定義する IMS 複数セグメントデータマップを追加する方法を示します。

この例では、複数のセグメントを一度に 1 つ追加して、複雑なテーブルのセグメント間に階層リレーションを作成します。

IMS 複数セグメントデータマップの作成手順

1. 複数のセグメントを含むデータマップを開きます。
2. リソースエクスプローラの **【データマップ】** タブで、テーブルを右クリックして **【プロパティ】** をクリックします。
3. **【テーブルプロパティ】** ダイアログボックスに、以下の情報を入力します。
 - **【レコードの依存関係】** リストで、親にするセグメントを選択します。
 - **【使用可能なレコード】** リストで、子として追加するセグメントを右クリックし、**【レコードを子として追加】** をクリックします。
 - **【レコードの依存関係】** リストで、親セグメントを選択します。
 - **【選択したレコードの複数インスタンスをどのように処理しますか？】** リストで、**【新しい行】** を選択します。
 - **【レコードの依存関係】** リストで、それぞれの子セグメントを選択し、複数インスタンスオプションを **【新しい行】** に設定します。
4. **【OK】** をクリックします。

注: 階層の各セグメントに対し、**【選択したレコードの複数インスタンスをどのように処理しますか？】** オプションに異なるオプションを設定できます。例えば、ROOT を **【無視】** に、SEG1 を **【新しい行】** に、SEG2 を **【配列】** に設定します。

親セグメントが **【配列】** に設定されている場合は、すべての子セグメントを **【無視】** に設定します。

セグメントを同じレベルで定義することもできます。例えば、SEG1 と SEG2 を ROOT セグメントに兄弟として追加します。

関連項目：

- [「IMS 階層の表示」 \(ページ 128\)](#)
- [「テーブルプロパティ - 定義」 \(ページ 391\)](#)

SEQ データマップ

Linux、UNIX、Windows のファイル、または z/OS のシーケンシャルデータセットに PowerExchange を介してアクセスするには、SEQ アクセス方式を使用するデータマップを追加します。

ソースファイルの構造を記述するには、SEQ データマップの追加時またはその後で、COBOL または PL/I のコピーブックをインポートできます。

SEQ データマップをテストするには、データベース行のテストを実行します。

SEQ データマップの追加

データマップを追加する場合、オプションでシーケンシャルデータセットまたはフラットファイルの構造を記述するためのコピーブックをインポートできます。以下の手順はコピーブックを使用することを前提としています。

1. リソースエクスプローラで **[リソース]** タブをクリックします。
2. メニューバーで **[追加] > [データマップ]** をクリックします。
3. **[名前]** ページで以下の情報を入力します。
 - **[スキーマ名]** および **[データマップ名]** を入力します。
 - **[アクセス方式]** リストで、**[SEQ]** を選択します。
 - コピーブックからレコード定義をインポートするために、**[レコード定義のインポート]** オプションを選択します。
4. **[次へ]** をクリックします。
5. **[SEQ アクセス方式]** ページで **[ファイル名]** とその他の必要な情報を入力し、**[完了]** をクリックします。

詳細については、[「データマップのプロパティ - SEQ アクセス方式」 \(ページ 365\)](#)を参照してください。

6. **[コピーブックのインポート - ソースの詳細]** ページで、ソースタイプを選択します。そして次のコピーブックのタイプから 1 つを選択します。

- **COBOL**
- **PL/1**

デフォルトは **COBOL** です。

7. **[次へ]** をクリックします。
8. コピーブックタイプの **[コピーブックのインポート-リモートの詳細]** ページにコピーブックの情報を入力し、**[次へ]** をクリックします。
9. **[コピーブックのインポート - 構成の詳細]** ページで、インポートしたレコード、フィールド、およびテーブルに対する操作を選択し、**[次へ]** をクリックします。

詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。

10. **[コピーブック情報のインポート]** ページで、インポート処理の情報を確認し、**[OK]** をクリックします。
11. **[レコード定義]** ページでインポートしたそれぞれのレコードに対する操作を選択し、**[OK]** をクリックします。

詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。

インポートウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: コピーブックメッセージログウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

12. 複数のレコードをインポートした場合、レコード ID 値を定義するために、それぞれのレコードに対して次の操作を行います。
 - a. リソースエクスプローラの **[データマップ]** タブで、レコードをクリックします。
 - b. **[レコード]** ウィンドウでレコードを示すフィールドを右クリックし、**[プロパティ]** をクリックします。
[フィールドのプロパティ] ダイアログボックス内の **[名前]** タブが表示されます。
 - c. **[レコード ID 値]** ボックスにレコードを定義する値と演算子を入力します。

13. レコードタイプ間のリレーションシップを識別するためにオプションのレコードプロパティを入力します。それぞれのレコードで次の操作を実行します。

- a. レコードを右クリックします。
- b. **【プロパティ】** を選択します。
- c. **【親レコード名】**、**【レコード名の再定義】**、または **【ヘッダ/トレーラレコード】** オプションを選択し、レコード名またはレコードのタイプを選択します。

このメタデータを使用して、複数のレコードデータマップを PowerCenter にインポートし、複数レコードの書き込みを実行できます。

複数レコードの書き込みの詳細については、『*PowerExchange バルクデータ移動ガイド*』、および『*PowerCenter 用の PowerExchange インタフェース*』を参照してください。

SEQ または VSAM のレコード階層の表示

SEQ データマップまたは VSAM データマップのレコードの階層を表示できます。階層は、定義されたオプションのレコードプロパティである**親レコード名**、**再定義レコード名**および**ヘッダ/トレーラレコード**に基づいています。

- ▶ リソースエクスプローラの **【データマップ】** タブで、レコードを右クリックして **【階層の表示】** をクリックします。

選択したレコードを含む階層が表示され、選択したレコードの名前は太字で示されます。

階層ダイアグラムでは、他のレコードを再定義するレコードには **REDEF** というラベルが示されます。

VSAM データマップ

非リレーショナル VSAM データに PowerExchange を使用してアクセスするには、VSAM データマップを追加します。

PowerExchange では、データソースがリレーショナルであるかのように、SQL を使用して VSAM データにアクセスします。

以下のタイプの VSAM データセットに対してデータマップを追加できます。

- キーシーケンスのデータセット (KSDS)
- エントリシーケンスのデータセット (ESDS)
- 固定長関連レコードのデータセット (RRDS)
- 可変長関連レコードのデータセット (VRRDS)

これらのデータセットを、基本形式、拡張形式、圧縮にすることができます。

データベース行のテストを実行して VSAM データマップをテストします。

VSAM データマップの追加

データマップを追加する場合、オプションで VSAM データセットの構造を記述するためのコピーブックをインポートできます。以下の手順はコピーブックを使用することを前提としています。

1. リソースエクスプローラで **【リソース】** タブをクリックします。
2. メニューバーで **【追加】 > 【データマップ】** をクリックします。

3. **【名前】** ページで以下の情報を入力します。
 - **【アクセス方式】** リストで、**【ESDS】**、**【KSDS】** または **【RRDS】** を選択します。
 - **【レコード定義のインポート】** オプションを選択します。

注: **【RRDS】** アクセス方法を固定長と可変長の RRDS (VRRDS) データセットに選択します。
4. **【次へ】** をクリックします。
5. **【アクセス方式】** ページで、アクセス方式に関する情報を入力し、**【完了】** をクリックします。

注: PowerExchange はデフォルトで、VSAM 処理にシステムで定義されたバッファリングを使用します。PowerExchange の処理を調整するには、追加の VSAM バッファを指定します。
6. **【コピーブックのインポート - ソースの詳細】** ページで、ソースタイプを選択します。そして次のコピーブックのタイプから 1 つを選択します。
 - **COBOL**
 - **PL/1**

デフォルトは **COBOL** です。
7. **【次へ】** をクリックします。
8. コピーブックタイプの **【コピーブックのインポート-リモートの詳細】** ページにコピーブックの情報を入力し、**【次へ】** をクリックします。
9. **【コピーブックのインポート - 構成の詳細】** ページで、インポートしたそれぞれのレコードに対する操作を選択し、**【次へ】** をクリックします。

詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。
10. **【コピーブック情報のインポート】** ページで、インポート処理の情報を確認し、**【OK】** をクリックします。
11. **【レコード定義】** ページでインポートしたそれぞれのレコードに対する操作を選択し、**【OK】** をクリックします。

詳細については、[「データマップへのコピーブックのインポート」 \(ページ 144\)](#)を参照してください。

インポートウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: コピーブックメッセージログウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。
12. 複数のレコードをインポートした場合、レコード ID 値を定義するために、それぞれのレコードに対して次の操作を行います。
 - a. **リソースエクスプローラの【データマップ】** タブで、レコードをクリックします。
 - b. **【レコード】** ウィンドウでレコードを示すフィールドを右クリックし、**【プロパティ】** をクリックします。

【フィールドのプロパティ】 ダイアログボックス内の **【名前】** タブが表示されます。
 - c. **【レコード ID 値】** ボックスにレコードを定義する値と演算子を入力します。
13. レコードタイプ間のリレーションシップを識別するためにオプションのレコードプロパティを入力します。それぞれのレコードで次の操作を実行します。
 - a. レコードを右クリックします。
 - b. **【プロパティ】** を選択します。
 - c. **【親レコード名】**、**【レコード名の再定義】**、または **【ヘッダ/トレーラレコード】** オプションを選択し、レコード名またはレコードのタイプを選択します。

このメタデータを使用して、複数のレコードデータマップを PowerCenter にインポートし、複数レコードの書き込みを実行できます。

複数レコードの書き込みの詳細については、『PowerExchange バルクデータ移動ガイド』、および『PowerCenter 用の PowerExchange インタフェース』を参照してください。

関連項目：

- [「データマッププロパティ - ESDS アクセス方式」 \(ページ 360\)](#)
- [「データマッププロパティ - KSDS アクセス方式」 \(ページ 362\)](#)
- [「データマッププロパティ - RRDS アクセス方式」 \(ページ 364\)](#)

SEQ または VSAM のレコード階層の表示

SEQ データマップまたは VSAM データマップのレコードの階層を表示できます。階層は、定義されたオプションのレコードプロパティである親レコード名、再定義レコード名およびヘッダ/トレーラレコードに基づいています。

- ▶ リソースエクスプローラの [データマップ] タブで、レコードを右クリックして [階層の表示] をクリックします。

選択したレコードを含む階層が表示され、選択したレコードの名前は太字で示されます。

階層ダイアグラムでは、他のレコードを再定義するレコードには **REDEF** というラベルが示されます。

VSAM データセットの一括読み込みのパフォーマンス向上

VSAM データセットの一括読み込みのパフォーマンスを向上することができます。

PowerExchange には、パフォーマンスを向上するための次のオプションがあります。

- データの読み込み時に使用するデータ数およびインデックスバッファ数を指定する。
- 制御間隔アクセスを指定する。
- WHERE 句にキーを使用する。

VSAM データバッファの数

VSAM が仮想ストレージと補助ストレージ間でデータを転送するために使用する I/O バッファの数を指定できます。

バッファは、データコンポーネント内の制御間隔のサイズです。データバッファの数の有効値 (BUFND) は、0 から 65535 です。最小値は 1 と STRNO に指定した値を足した値です。STRNO を省略した場合は、STRNO のデフォルト値が 1 であるため、BUFND に少なくとも 2 を指定する必要があります。

注: 必要最小限のバッファ指定では、最適なシーケンシャル処理パフォーマンスを得ることができません。通常、指定するデータバッファが大きくなるほど、パフォーマンスが向上します。データバッファの値を大きくするほど、制御領域の分割時のダイレクト挿入または更新、およびスパンされたレコードへのアクセスのパフォーマンスが向上します。

バッファの最大値は 255 で、254 データバッファと 1 つの挿入バッファが含まれます。

VSAM インデックスバッファ数

キー済みアクセスに対して、仮想ストレージと補助ストレージ間でインデックスエントリの内容を VSAM が転送するために使用する I/O バッファの数を指定します。

バッファとは、インデックス内のコントロールインターバルのサイズです。インデックスバッファの数の有効値 (BUFNI) は、0 から 65535 です。最小値は、STRNO に対して指定した数です。STRNO を省略した場合

は、STRNO のデフォルト値が 1 であるため、BUFNI に少なくとも 1 を指定する必要があります。デフォルトでは、必要最小限の値になります。

インデックスバッファを追加することで、一部またはすべての高レベルインデックスが常駐するようになり、キーダイレクト処理で DASD から取得される高レベルインデックスのレコード数が最小限に抑えられるため、パフォーマンスが向上します。

VSAM 文を DBMOVE コンフィギュレーションメンバに含めることにより、PowerExchange システムレベルで使用されるバッファ数を変更します。この文の値は、データマップのすべてのエントリをオーバーライドします。

バッファの最大値は 255 で、254 データバッファと 1 つの挿入バッファが含まれます。デフォルトでは、必要最小限の値になります。

VSAM 制御間隔アクセスの使用

制御間隔 (CI) アクセスは、VSAM アクセス方法で直接制御間隔の読み取りを使用して、データレコードを取得する読み取りオプションです。

VSAM データセットからすべてのデータを抽出する場合、このオプションによって、シーケンシャル読み取りのパフォーマンスを向上させることができます。データの取得時に WHERE 句を指定する場合には、CI アクセスではパフォーマンスを向上できません。

CI アクセスをデータマップレベルで選択するには、**【データマップのプロパティ】** ダイアログボックスの **【アクセス方式】** タブの **【CI アクセス】** オプションを選択します。**【CI アクセス】** オプションは、KSDS、ESDS、および RRDS アクセス方法で有効です。

圧縮された VSAM データセットまたはスパンされたレコードを持つ VSAM データセットでは、CI アクセスはサポートされません。

VSAM キーの使用

PowerExchange では、VSAM データセットが割り当てられ開いているときに、カタログからそのデータセットのキー情報を読み取ります。

キー長とキーの相対的位置の両方が抽出されます。

例えば、次の行では、keypos=10、keylen=4 に対してレコードレイアウトが定義されます。

```
01 REC.
   04 HEADER    PIC X(10).
   04 KEY.
       08 KEY1    PIC 99.
       08 KEY2    PIC 99.
   04 REST      PIC X(200).
```

作業を最適化するために、KEY1 を WHERE 句に含めます。

キー情報を使用して、最適化を可能にする WHERE 句の処理速度を上げます。

WHERE 句形式が異なると、結果も異なります。

以下に例を示します。

```
Line 1   SELECT * FROM TEST.VSAM1.TAB WHERE (KEY1 > 50) AND (KEY1 < 90)
```

Line 1 は次のように変換されます。

```
(50 < KEY1 < HIGH-VALUES) AND (LOW-VALUES < KEY1 < 90)
```

次の例では Line 1 の例と同じ結果になりますが、経過時間が異なります。

```
Line 2   SELECT * FROM TEST.VSAM1.TAB WHERE (KEY1 > 50 AND KEY1 < 90)
```

Line 2 は次のように変換されます。

(KEY1 > LOW-VALUES AND KEY1 < HIGH-VALUES)

この形式は技術的には最適化されていますが、ファイルを処理する場合、読み取りは LOW-VALUES から開始されて HIGH-VALUES まで読み取るので、理想的ではありません。

Line 2 が推奨される構文です。この構文では、ファイルをレコード 50 に置き、KEY1 が 90 以上になるまでデータを読み取ります。

最適化はキー位置決定に使用され、データは WHERE 句で選択され、正しいレコードがフィルタリングされることを確認します。

注: 0 より大きい値を **【KSDS アクセス方式】** ダイアログボックスの **【最初のレコードをスキップする】** ボックスで指定した場合、KEY1 が指定されていても SELECT 文は最適化されません。

VSAM データセットのレコードに対する RRN または RBA を取得

VSAM エントリシーケンスのデータセット (ESDSs) と相対レコードデータセット (RRDSs) に対して、相対レコード番号 (RRN) か相対バイトアドレス (RBA) が取得できます。

次のように RRN または RBA を返します。

- データマップのプロパティでこれらのオプションを選択するときに、RRN または XRBA の値をプレフィックスとしてレコードに付けます。
- GetDatabaseKey 関数を式で使用するときに、レコードの新しいフィールドに RRN または RBA の値を代入します。

注: オフロード処理を要求する場合は、GetDatabaseKey 関数を式で使用できません。

VSAM データセットレコードの RRN または RBA を取得するためのデータマッププロパティの設定

データマップのプロパティを設定して、VSAM データセットレコードの相対レコード番号 (RRN) または相対バイトアドレス (RBA) を取得できます。

- データマップを開きます。
- データマップを右クリックし、**【プロパティ】** をクリックします。
- 【データマップのプロパティ】** ダイアログボックスの **【アクセス方式】** タブで、次のオプションから 1 つを選択します。
 - RBA を ESDS データセットに対して取得するには、**【XRBA のレコードのプリフィクス】** を選択します。
 - RRN を RRDS データセットに対して取得するには、**【RRN のレコードのプリフィクス】** を選択します。
- 注:** PowerExchange で RRN または RBA の値を返すことを要求する場合、レコード定義の先頭にこの値を含めるための 8 バイトのバイナリフィールドを追加する必要があります。
- レコードを開きます。
- レコードウィンドウで、レコードの最初のフィールドを右クリックして **【前にフィールドを追加】** をクリックします。
- 【フィールド名】** の値を入力します。 **【フィールドの型】** に **BIN** を、**【長さ】** フィールドに **8** を選択します。
- 【OK】** をクリックします。
- テーブルを開きます。

9. テーブルウィンドウで、テーブルの最初の列を右クリックして **【前にフィールドを追加】** をクリックします。
10. **【名前】** フィールドに、新しい列の名前を入力します。
11. **【ベースフィールド】** リストで、新しいフィールドを選択します。
12. データマップを VSAM データセットが存在する z/OS システムに送信するには、**【ファイル】** > **【リモートノードへ送信】** をクリックします。

GetDatabaseKey 関数を使用した VSAM データセットレコードの RRN または RBA の取得

GetDatabaseKey 関数を式で使用して、VSAM データセットレコードの相対レコード番号 (RRN) または相対バイトアドレス (RBA) を取得します。

注: オフロード処理を使用する場合は、GetDatabaseKey 関数を式で使用できません。

データマップのレコード定義で式を定義します。ユーザー定義フィールドをレコードに追加し、式の結果を含めます。

1. データマップとレコードを開きます。
2. **【Expr(*n*)】** タブで、ウィンドウを右クリックし、**【最後にフィールドを追加】** をクリックします。
3. **【名前】** 列に名前を入力します。
4. **【型】** 列で、**BIN** を入力します。
5. **【長さ】** 列で、**8** を入力します。
6. **【式】** 列で、**GetDatabasekey()** を入力します。
式フィールドが有効な場合は、緑のチェックマークアイコンが表示されます。それ以外の場合は、赤のバツアイコンとメッセージが表示されます。
7. テーブルを開きます。
8. テーブルウィンドウで、テーブル名を右クリックして、**【最後にカラムを追加】** をクリックします。
9. **【ベースフィールド】** リストで、新しい列を選択します。
10. **【名前】** フィールドに、新しい列の名前を入力します。
11. **【OK】** をクリックします。
12. **【ファイル】** > **【リモートノードへ送信】** をクリックして、VSAM データセットの存在する z/OS システムにデータマップを送信します。

第 4 章

コピーブック

この章では、以下の項目について説明します。

- [コピーブックの概要, 142 ページ](#)
- [データマップへのコピーブックのインポート, 144 ページ](#)
- [レコードまたはセグメントへのコピーブックのインポート, 146 ページ](#)
- [Adabas データマップへの FDT のインポート, 148 ページ](#)
- [Adabas データマップ用 FDT の表示とリフレッシュ, 149 ページ](#)
- [\[コピーブックのインポート - ソースの詳細\] ダイアログボックス, 149 ページ](#)
- [\[Import Copybook - Field Property Details \(コピーブックのインポート-フィールドプロパティの詳細\)\] ダイアログボックス, 153 ページ](#)
- [\[コピーブックのインポート - ローカル詳細\] ダイアログボックス, 153 ページ](#)
- [\[コピーブックのインポート - ローカル DB2 カタログの詳細\] ダイアログボックス, 154 ページ](#)
- [\[コピーブックのインポート - リモートの詳細\] ダイアログボックス, 155 ページ](#)
- [\[コピーブックのインポート - リモート Datacom の詳細\] ダイアログボックス, 156 ページ](#)
- [\[コピーブックのインポート - リモート DB2 カタログの詳細\] ダイアログボックス, 158 ページ](#)
- [\[DB2 カラムの選択\] ダイアログボックス, 159 ページ](#)
- [\[IDMS 詳細プロパティ\] ダイアログボックス, 160 ページ](#)
- [\[コピーブックのインポート - 構成の詳細\] ダイアログボックス, 161 ページ](#)
- [\[レコード定義\] ダイアログボックスと \[重複レコード定義\] ダイアログボックス, 162 ページ](#)

コピーブックの概要

PowerExchange Navigator では、用語のコピーブックは、通常、データマップにインポートしてデータのレイアウトを定義するデータソースからのレコード、セグメント、フィールド、キー、およびその他の項目に対するデータ定義を意味します。

コピーブックをインポートすることで、データマップのこれらの項目のレイアウトを手動で定義する必要がなくなります。

コピーブックは、データマップを追加するときか追加した後にインポートできます。1 つ以上のコピーブックをデータマップのレコードまたはセグメントにインポートして、データ定義をレコードまたはセグメントに追加することもできます。

コピーブックは、インポートした後にローカルシステムに格納されます。その後ローカルシステムからそのコピーブックを再インポートできます。

コピーブックに、データマップに既にあるレコードやフィールドと重複するレコードやフィールドが含まれている場合、重複項目の扱い方法を示すオプションを選択できます。例えば、項目のインポート、上書き、スキップ、またはインポートの停止を選択できます。

PowerExchange インストールディレクトリの examples フォルダにあるサンプルデータマップやコピーブックを使用して、以下のタイプのコピーブックをデータマップにインポートする方法を学習します。

- COBOL コピーブック
- REDEFINES 文を含む COBOL コピーブック
- 1つのフィールドに複数の定義を持つ PL/I コピーブック

関連項目：

- [「データマップへの COBOL コピーブックのインポート - 例」 \(ページ 268\)](#)
- [「REDEFINES 文を含む COBOL コピーブックのインポート - 例」 \(ページ 272\)](#)
- [「PL/I コピーブックのインポート - 例」 \(ページ 276\)](#)

データマップでサポートされるコピーブックタイプ

コピーブックのタイプを選択して、**「コピーブックのインポート-ソース詳細」** ダイアログボックスの **「タイプ」** リストにインポートします。

インポートできるコピーブックのタイプは、次のように、コピーブックをインポートしている先のデータマップのデータソースタイプによって異なります。

- **DB2 以外のすべてのデータソースタイプに対するデータマップ。** ローカルまたはリモートシステムから COBOL または PL/I コピーブックをインポートして、レコードやフィールドのレイアウトを定義します。
- **Adabas データマップ。** 次のように、フィールド定義テーブル (FDT) とコピーブックの両方を Adabas データマップにインポートすることができます。
 - **FDT。** それぞれのデータベースファイルに対応する FDT があり、これにより物理ファイルの各フィールドのレコード構造体とコンテンツが定義されます。レコードの各フィールドで、FDT によりレベル、名前、長さ、形式、オプション、特殊フィールドおよび記述子属性が定義されます。FDT により省略名と位取りなしの情報が提供されます。FDT は、インポートされた後、後続のセッションに対してローカルシステムでキャッシュされます。

FDT を Adabas データマップに表示することができます。FDT を表示する場合に、リモートシステムにある FDT からのデータでローカルで保存した FDT を、オプションとしてリフレッシュすることができます。
 - **ADACMP。** ADACMP ユーティリティを使用して作成されたファイルです。
 - **DDM。** z/OS ソースの Adabas に対する初期設定のデータ定義モジュール (DDM) です。ロングネームおよびスケール情報を提供します。
 - **PREDICT。** Predict データディクショナリ。ロングネームおよびスケール情報を提供します。
 - **TEXT。** Informatica の内部区切り文字形式を使用するテキストファイルです。
 - **COBOL または PL/I。** COBOL または PL/I コピーブックを Adabas データマップにインポートする場合、インポートしたフィールド定義を Adabas フィールド定義と関連付けるためにデータマップのテーブルプロパティを編集する必要があります。
- **Datacom データマップ。** COBOL または PL/I コピーブックをインポートするか、リモート z/OS システムまたはローカル dcm ファイルから DATACOM コピーブックをインポートします。DATACOM コピーブックにより、結合された CA Datacom/DB スキーマおよび COBOL コピーブック情報が提供されます。
- **IDMS データマップ。** COBOL または PL/I コピーブックをインポートするか、リモート z/OS システムまたはローカル idm ファイルから IDMS コピーブックをインポートします。IDMS コピーブックにより、結合された CA IDMS/DB スキーマおよび COBOL コピーブック情報が提供されます。

- **i5/OS 用 DB2、Linux、UNIX、および Windows 用 DB2、DB2 for z/OS および DB2 アンロードファイル データマップ。** DB2 カタログをインポートできます。
- **順次アクセスする i5/OS データマップ用 DB2。** COBOL または PL/I コピーブックをインポートするか、データを処理するアプリケーションプログラムの外にあるファイル説明のデータ属性を記述する、データ記述仕様 (DDS) をインポートします。
- **IMS DL/1 バッチおよび ODBA データマップ。** データベース記述ブロック (DBD) をインポートして、セグメント、キーフィールド、検索フィールド、IMS データベースからの CCK フィールド定義に追加し、データマップにテーブルを追加します。また、DBD のインポートにより、セグメントの階層シーケンスの定義も行います。

DBD をインポートした後、COBOL または PL/I コピーブックをデータマップの各セグメントにインポートし、その COPYLIB でセグメントをオーバーレイします。データベースの階層メタデータを保持している間、このアクションによりデータマップが再定義されます。

データマップへのコピーブックのインポート

コピーブックは、データマップを追加するときか追加した後にインポートできます。

ヒント: COBOL コピーブックをインポートする場合、マッピングエラーのトラブルシューティングやその他の目的で利用できる拡張メモリマップを提供するトレースを、インポート時に実行できます。詳細については、[付録 G, 「COBOL コピーブックのインポート時におけるメモリマップ作成のトレース」 \(ページ 399\)](#)を参照してください。

1. コピーブックを既存のデータマップにインポートするかデータマップを追加するかに応じて、次のいずれかの手順を実行します。
 - 既存のデータマップの場合は、データマップを開いて **[ファイル] > [コピーブックのインポート]** をクリックします。
 - データマップを追加する場合は、**[名前]** ダイアログボックスの **[レコード定義のインポート]** チェックボックスをオンにします。
[次へ] をクリックします。**[アクセス方式]** ダイアログボックスに情報を入力して、**[完了]** をクリックします。

[コピーブックのインポート - ソースの詳細] ダイアログボックスが表示されます。

2. **[コピーブックのインポート - ソースの詳細]** ダイアログボックスで、インポートするコピーブックの場所とタイプについての情報を入力します。

いくつかのコピーブックタイプでは、追加情報の入力が必要です。

詳細については、[「\[コピーブックのインポート - ソースの詳細\] ダイアログボックス」 \(ページ 149\)](#)を参照してください。

注: IMS データマップについては、DBD コピーブックをインポートします。次に、COBOL または PL/I コピーブックをデータマップの各セグメントにインポートします。

3. 最新のインポート処理の情報を表示するには、**[最新インポート]** をクリックします。

コピーブック情報のインポートウィンドウでは、コピーブックの最新のインポート処理についての以下の情報が表示されます。

- コピーブックがローカルまたはリモートシステムからインポートされたかどうか
 - コピーブックの名前とタイプ
 - インポートされたレコード、フィールド、テーブルに対して定義されたデフォルトアクション
- ダイアログボックスを閉じるには、**[キャンセル]** をクリックします。

4. **【次へ】** をクリックします。
5. リモートまたはローカルのコピーブックのどちらの場所を指定したかを基にして、データソースタイプに以下のダイアログボックスのうちの 1 つにあるコピーブックについての情報を入力します。
 - **【コピーブックのインポート-リモートの詳細】** ダイアログボックス。詳細については、[「【コピーブックのインポート-リモートの詳細】ダイアログボックス」 \(ページ 155\)](#)を参照してください。
 - **【コピーブックのインポート-ローカル詳細】** ダイアログボックス。詳細については、[「【コピーブックのインポート-ローカル詳細】ダイアログボックス」 \(ページ 153\)](#)を参照してください。
6. **【次へ】** をクリックします。
7. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するデフォルトアクションを定義するオプションを選択します。

詳細については、[「【コピーブックのインポート - 構成の詳細】ダイアログボックス」 \(ページ 161\)](#)を参照してください。
8. COBOL コピーブックをインポートしている場合は、**【Import Copybook - Field Property Details (コピーブックのインポート-フィールドプロパティの詳細)】** ダイアログボックスにフィールドプロパティの詳細を入力します。

詳細については、[「【Import Copybook - Field Property Details \(コピーブックのインポート-フィールドプロパティの詳細\)】ダイアログボックス」 \(ページ 153\)](#)を参照してください。
9. **【完了】** をクリックします。
10. **コピーブック情報のインポート**ウィンドウで、インポート操作の情報を確認します。

コピーブック情報のインポートウィンドウには、以下の情報が表示されます。

 - PowerExchange によってコピーブックがローカルまたはリモートシステムからインポートされたかどうか
 - コピーブックの名前とタイプ
 - インポートされたレコード、フィールド、テーブルに対して定義されたデフォルトアクション
11. **【OK】** をクリックします。
12. **【レコード定義】** ダイアログボックスで、インポートした各レコードまたはセグメントに対するアクションを選択します。

データマップのレコードあるいはフィールドを重複する、レコードまたはフィールドがコピーブックに含まれる場合、**【重複レコード定義】** ダイアログボックスが表示されます。**【重複レコード定義】** ダイアログボックスで、重複項目の扱い方法を示すオプションを選択できます。例えば、項目のインポート、上書き、スキップ、またはインポートの停止を選択できます。

詳細については、[「【レコード定義】ダイアログボックスと【重複レコード定義】ダイアログボックス」 \(ページ 162\)](#)を参照してください。
13. **【OK】** をクリックします。
14. コピーブックに REDEFINES 文が含まれる場合、**コピーブックの再定義**ウィンドウにより、コピーブックに REDEFINES 文が含まれていることが通知されます。

以下の手順を実行します。

 - a. 正しいフィールドを含む行を指定するには、**【再定義】** > **【次へ】**、または **【再定義】** > **【前へ】** をクリックします。
 - b. **【インポート】** > **【再開】** をクリックします。
15. **インポート**ウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: **コピーブックメッセージログ**ウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

16. インポートウィンドウを閉じます。
17. データマップで重複レコードまたはフィールドの上書きを選択した場合、以下のアクションを実行します。
 - リソースエクスプローラの [データマップ] タブで、更新されたレコードまたはフィールドに基づいた列を含むテーブルを右クリックして、[プロパティ] をクリックします。
 - [テーブルプロパティ] ダイアログボックスの [レコードの依存関係] リストで、テーブルを選択します。
 - [カラムの生成] リストから、[不足しているカラムを使用してリフレッシュ] を選択します。
 - テーブル上でデータの行テストを実行します。

インポートしたデータマップのデータソースタイプとコピーブックのタイプに基づいて、以下のタスクのうち1つを完了することが必要な場合があります。

- Adabas データマップに COBOL または PL/I コピーブックをインポートした場合、データマップを編集してインポートしたフィールド定義を Adabas フィールド定義に関連付けます。詳細については、[「データマップテーブル用プロパティの表示または編集」](#) (ページ 91) を参照してください。
- DBD ソースを IMS データマップにインポートした場合、COBOL または PL/I コピーブックをデータマップの各セグメントにインポートします。詳細については、[「レコードまたはセグメントへのコピーブックのインポート」](#) (ページ 146) を参照してください。
- ワイド文字データタイプで定義されたフィールドを持つ DDM コピーブックを Adabas for z/OS データマップにインポートした場合、PowerExchange では、[フィールドのプロパティ] ダイアログボックスの [コードページ] タブにある [ワイド文字] オプションの選択を行いません。

ワイド文字データタイプで定義されたフィールドに対して [ワイド文字] オプションを設定するには、以下のいずれかのタスクを実行します。

- PREDICT または FDT コピーブックをインポートします。
- データマップのフィールドプロパティを編集します。

関連項目：

- [「\[コピーブックのインポート - ローカル詳細\] ダイアログボックス」](#) (ページ 153)
- [「\[コピーブックのインポート - ローカル DB2 カタログの詳細\] ダイアログボックス」](#) (ページ 154)
- [「\[コピーブックのインポート - リモートの詳細\] ダイアログボックス」](#) (ページ 155)
- [「\[コピーブックのインポート - リモート Datacom の詳細\] ダイアログボックス」](#) (ページ 156)
- [「\[コピーブックのインポート - リモート DB2 カタログの詳細\] ダイアログボックス」](#) (ページ 158)

レコードまたはセグメントへのコピーブックのインポート

1つ以上のコピーブックをデータマップのレコードまたはセグメントにインポートして、データ定義をレコードまたはセグメントに追加します。

注: DBD を IMS データマップにインポートした後、COBOL コピーブックをデータマップの各セグメントにインポートし、その COPYLIB で各セグメントをオーバーレイします。このアクションにより、IMS データベースの階層メタデータ情報を保持したままで、データマップが再定義されます。

1. データマップを開きます。
2. リソースエクスプローラの [データマップ] タブで、レコードまたはセグメントを選択します。

3. **【ファイル】 > 【コピーブックのインポート】** をクリックします。
4. **【コピーブックのインポート - ソースの詳細】** ダイアログボックスで、インポートするコピーブックの場所とタイプについての情報を入力します。

いくつかのコピーブックタイプでは、追加情報の入力が必要です。

詳細については、[「【コピーブックのインポート - ソースの詳細】 ダイアログボックス」 \(ページ 149\)](#)を参照してください。

注: IMS データマップについては、DBD コピーブックをインポートします。次に、COBOL または PL/I コピーブックをデータマップの各セグメントにインポートします。
5. **【次へ】** をクリックします。
6. リモートまたはローカルコピーブックのどちらを選択したかに基づいて、以下のダイアログボックスのうちの 1 つにあるコピーブックの情報を入力します。
 - **【コピーブックのインポート-リモートの詳細】** ダイアログボックス. 詳細については、[「【コピーブックのインポート - リモートの詳細】 ダイアログボックス」 \(ページ 155\)](#)を参照してください。
 - **【コピーブックのインポート-ローカル詳細】** ダイアログボックス. 詳細については、[「【コピーブックのインポート - ローカル詳細】 ダイアログボックス」 \(ページ 153\)](#)を参照してください。

注: セキュリティが z/OS で有効になっている場合、ユーザー ID とパスワードを入力します。
7. **【次へ】** をクリックします。
8. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコードまたはセグメント、フィールド、およびテーブルに対するアクションを選択し、**【完了】** をクリックします。詳細については、[「【コピーブックのインポート - 構成の詳細】 ダイアログボックス」 \(ページ 161\)](#)を参照してください。
9. **コピーブック情報のインポート**ウィンドウで、インポート操作の情報を確認します。

コピーブック情報のインポートウィンドウには、以下の情報が表示されます。

 - PowerExchange によってコピーブックがローカルまたはリモートシステムからインポートされたかどうか
 - コピーブックの名前とタイプ
 - インポートされたレコード、フィールド、テーブルに対して定義されたデフォルトアクション
10. **【OK】** をクリックします。
11. **【レコード定義】** ダイアログボックスで、インポートした各レコードまたはセグメントに対するアクションを選択します。

データマップのレコードあるいはフィールドを重複する、レコードまたはフィールドがコピーブックに含まれる場合、**【重複レコード定義】** ダイアログボックスが表示されます。**【重複レコード定義】** ダイアログボックスで、重複項目の扱い方法を示すオプションを選択できます。例えば、項目のインポート、上書き、スキップ、またはインポートの停止を選択できます。

詳細については、[「【レコード定義】 ダイアログボックスと【重複レコード定義】 ダイアログボックス」 \(ページ 162\)](#)を参照してください。
12. **【OK】** をクリックします。
13. コピーブックに REDEFINES 文が含まれる場合、**コピーブックの再定義**ウィンドウにより、コピーブックに REDEFINES 文が含まれていることが通知されます。

以下の手順を実行します。

 - a. 正しいフィールドを含む行を指定するには、**【再定義】 > 【次へ】**、または **【再定義】 > 【前へ】** をクリックします。
 - b. **【インポート】 > 【再開】** をクリックします。
14. **インポート**ウィンドウに、インポートしたコピーブックが表示されます。

コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。

注: コピーブックメッセージログウィンドウにエラーメッセージが表示されたら、メッセージをダブルクリックしてコピーブックの関連行を検索します。

15. インポートウィンドウを閉じます。
16. データマップで重複レコードまたはフィールドの上書きを選択した場合、以下のアクションを実行します。
 - リソースエクスプローラの **【データマップ】** タブで、更新されたレコードまたはフィールドに基づいた列を含むテーブルを右クリックして、**【プロパティ】** をクリックします。
 - **【テーブルプロパティ】** ダイアログボックスの **【レコードの依存関係】** リストで、テーブルを選択します。
 - **【カラムの生成】** リストから、**【不足しているカラムを使用してリフレッシュ】** を選択します。
 - テーブル上でデータの行テストを実行します。

関連項目：

- [「\[コピーブックのインポート - ソースの詳細\] ダイアログボックス」 \(ページ 149\)](#)
- [「\[コピーブックのインポート - ローカル詳細\] ダイアログボックス」 \(ページ 153\)](#)
- [「\[コピーブックのインポート - リモートの詳細\] ダイアログボックス」 \(ページ 155\)](#)

Adabas データマップへの FDT のインポート

以前に FDT をデータマップにインポートしていなかった場合、FDT を Adabas データマップにインポートすることができます。

FDT をデータマップにインポートする場合、Adabas キーもインポートします。

Adabas データマップに FDT をインポートする手順

1. データマップを開きます。
2. PowerExchange Navigator により FDT をインポートするように求められない場合は、**【リソース】 - 【FDT】** をクリックします。
3. **【Adabas FDT インポート】** ダイアログボックスで、FDT のある場所のシステムについての接続情報を入力します。詳細については、[「データマップの追加」 \(ページ 66\)](#)を参照してください。
4. **【OK】** をクリックします。

データマップのテーブルプロパティを編集してインポートされたキーをテーブルに追加します。

関連項目：

- [「Adabas データマップ内のテーブルへのキーの追加」 \(ページ 112\)](#)

Adabas データマップ用 FDT の表示とリフレッシュ

PowerExchange Navigator の Adabas データマップにローカルで保存されている FDT を表示することができます。

FDT を表示する場合、リモートの Adabas システムにある元の FDT からのデータでオプションとしてリフレッシュすることができます。

Adabas データマップ用 FDT を表示およびリフレッシュする手順

1. データマップを開きます。
2. **[リソース] - [FDT]** をクリックします。
[Adabas FDT] ダイアログボックスに、FDT が表示されます。
3. **[Adabas FDT]** ダイアログボックスで、**[FDT のリフレッシュ]** をクリックします。
4. **[Adabas FDT インポート]** ダイアログボックスで、FDT を読み取っているシステムの接続情報を入力します。詳細については、[「Adabas データマップへの FDT のインポート」 \(ページ 148\)](#)を参照してください。
5. **[OK]** をクリックします。

メッセージボックスに、リフレッシュ処理のステータスが表示されます。

リフレッシュ処理が成功した場合、ローカルシステムにある FDT のコピーがリモート FDT ファイルからのデータでリフレッシュされます。

[コピーブックのインポート - ソースの詳細] ダイアログボックス

データマップまたはレコードにインポートしているコピーブックの情報を入力します。

ソース

コピーブックの場所を選択します。

以下の表で、オプションについて説明します。

オプション	説明
ローカル	コピーブックは PowerExchange Navigator システムにあります。
リモート	コピーブックはリモートシステムにあります。

タイプ

コピーブックのタイプを選択します。インポートできるコピーブックのタイプは、データマップ内にコピーブックをインポートしていますが、そのデータマップのアクセス方法によって異なります。例えば、

ADACMPのコピーブックタイプは、ADABAS アクセス方法でのみ定義されたデータマップにインポートできます。

COBOL または PL/1 コピーブックをインポートするには、**【COBOL】** または **【PL/1】** をそれぞれ選択します。

以下の表に、アクセス方式別および COBOL または PL/1 コピーブックをインポートする対応データソース別にデータマップのタイプを示します。

アクセス方式で定義されたデータマップへのインポート	対応するデータソース
ADABAS	Adabas ファイル
DATACOM	CA Datacom/DB ファイル
DB2UNLD	DB2 アンロードファイル
DL/1 バッチ	IMS データベース
ESDS	VSAM ESDS
IDMS	CA IDMS/DB データベース
IMS ODBA	PowerExchange リスナ経由でアクセスする IMS データベース
KSDS	VSAM KSDS
MQSERIES	IBM MQSeries メッセージキュー
RRDS	VSAM RRDS
SEQ	フラットファイルまたはシーケンシャルデータセット
TAPE	テープデータセット
USER	ユーザーアクセス方式プログラム

以下の表に、**【タイプ】** オプションと、アクセス方式別および COBOL または PL/1 コピーブック以外の各コピーブックをインポートする対応データソース別にデータマップのタイプを示します。

オプション	説明	アクセス方式で定義されたデータマップへのインポート	対応するデータソース
ADACMP	ADACMP ユーティリティを使用して作成されたファイル。	ADABAS	Adabas ファイル
DATACOM	Datacom コピーブック。結合された CA Datacom/DB スキーマおよび COBOL コピーブック情報が提供されます。	DATACOM	CA Datacom/DB ファイル

オプション	説明	アクセス方式で定義されたデータマップへのインポート	対応するデータソース
DB2 カタログ	DB2 カタログ。	DB2	DB2 for i5/OS、DB2 for Linux, UNIX, and Windows、または DB2 for z/OS データベース
DB2 カタログ	DB2 カタログ。	DB2UNLD	DB2 アンロードファイル
DBD	データベース記述ブロック (DBD)。IMS データベースからセグメント、キーフィールド、検索フィールド、CCK フィールド定義を追加して、テーブルをデータマップに追加します。セグメントの階層シーケンスの定義も行います。 ¹	DL/1 バッチ	IMS データベース
DBD	データベース記述ブロック (DBD)。IMS データベースからセグメント、キーフィールド、検索フィールド、CCK フィールド定義を追加して、テーブルをデータマップに追加します。セグメントの階層シーケンスの定義も行います。 ¹	IMS ODBA	PowerExchange リスナ経由でアクセスする IMS データベース
DDM	初期設定のデータ定義モジュール (DDM)。ロングネームおよびスケール情報を提供します。	ADABAS	z/OS の Adabas ファイル
DDS	データ記述仕様 (DDS)。データを処理するアプリケーションプログラムの外にあるファイル説明のデータ属性を記述します。	SEQ	順次アクセスする DB2 for i5/OS データ
FDT	フィールド定義テーブル (FDT)。各データベースファイルには対応する FDT があり、これにより物理ファイルの各フィールドのレコード構造とコンテンツが定義されます。レコードの各フィールドで、FDT によりレベル、名前、長さ、形式、オプション、特殊フィールドおよび記述子属性が定義されます。FDT により省略名と位取りなしの情報が提供されます。	ADABAS	Adabas ファイル
IDMS	IDMS コピーブック。結合された CA IDMS/DB スキーマおよび COBOL コピーブック情報が提供されます。	IMDS	CA IDMS/DB データベース
PREDICT	Predict データディクショナリ。ロングネームおよびスケール情報を提供します。	ADABAS	z/OS の Adabas ファイル

オプション	説明	アクセス方式で定義されたデータマップへのインポート	対応するデータソース
TEXT	Informatica の内部形式で使用する区切り形式ファイルのテキストファイルです。	ADABAS	Adabas ファイル
1. DBD をインポートする場合、PowerExchange は IMS 12 の IMS カタログ導入前の DBD 文を完全にサポートします。PowerExchange は、IMS カタログに関連する DBD 強化機能のすべてを無視します。			

カラムの範囲

以下の表に、コピーブックの開始カラムと終了カラムの範囲を示します。

オプション	説明
開始	開始カラム番号です。 有効な値は、1 – 999 です。 COBOL コピーブックタイプのデフォルトは 7 です。 その他すべてのコピーブックタイプのデフォルトは 1 です。
終了	終了カラム番号です。 有効な値は、1 – 999 です。 COBOL および PL/I コピーブックタイプのデフォルトは 72 です。 その他すべてのコピーブックタイプのデフォルトは 80 です。

注: IMS DL/1 バッチデータマップに、インポートされた DBD の列の範囲を設定することはできません。

FDIC ファイル詳細

以下の表に、DDM コピーブックタイプまたは PREDICT コピーブックタイプの FDIC ファイルの詳細を示します。

オプション	説明
データベース ID	FDIC データベース ID です。 有効な値は、1 – 99999 です。
ファイル番号	FDIC ファイル番号です。 有効な値は、1 – 9999 です。 デフォルトは 1 です。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)

[Import Copybook - Field Property Details (コピーブックのインポート-フィールドプロパティの詳細)] ダイアログボックス

コピーブックの Picture 句 G (PIC G) で定義される各フィールドに対して選択される 2 バイト文字セット (DBCS) のコードページ値を表示または変更します。

データマップのデフォルトのコードページを使用する

コピーブックの PIC G を使用して定義されている各フィールドのデフォルトのオペレーティングシステムコードページを PowerExchange に使用させる場合は、このオプションを選択します。

デフォルトでは、このオプションは選択されています。

選択したコードページを使用する

PIC G を使用して定義されているコピーブックのフィールドのコードページを選択します。

注: マルチバイトコードページが使用できるのは、Datacom、DB2、VSAM KSDS、シーケンシャルデータセット、およびフラットファイルです。

生成されるそれぞれの文字は選択されているコードページの CHAR 型です。

[コピーブックのインポート - ローカル詳細] ダイアログボックス

ローカルコピーブックの情報を入力します。

ファイル名

コピーブックのパスおよびファイル名を入力するか、[参照] ボタンをクリックしてファイルを参照します。

デフォルトはローカルシステムからデータマップにインポートされた最新のコピーブックです。

プレビュー

[プレビュー] をクリックしてコピーブックを表示します。

フィールド名と説明を連結する

DDS で、このオプションを選択して DDS フィールド名と説明を連結させます。

デフォルトでは選択されていません。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)

[コピーブックのインポート - ローカル DB2 カタログの詳細] ダイアログボックス

ローカル DB2 カタログの情報を入力します。

DB インスタンス

DB2 データベースのインスタンス名を入力します。

場所

DB2 カタログの場所に **[ローカル]** を選択します。カタログは PowerExchange Navigator を実行するシステムにあります。

テーブル名

DB2 カタログをインポートしているテーブル名を入力するか、**[参照]** ボタンをクリックしてテーブルのフィルタ条件を入力します。

ユーザー ID

データベースにアクセスする権限を持つユーザー ID を入力します。

パスワード

指定したユーザー ID のパスワードを入力します。

名前を付けてローカルに保存

DB2 カタログのコピーが保存されたローカルシステムのパスとファイル名を入力するか、または **[参照]** ボタンをクリックしてファイルを参照します。

Null インジケータ

DB2 アンロードファイルデータマップに、アンロードファイルに使用されている Null インジケータの 16 進値を入力します。

変数のパディング

DB2 アンロードファイルデータマップに、アンロードファイルの変数が埋め込まれている場合にこのオプションを選択します。

注: このオプションは、BMC UNLOAD PLUS FIXEDVARCHAR に対応しています。

カラムのプレビュー/変更

[DB2 カラムの選択] ダイアログボックスを表示します。それによりアクセスされるテーブル内の列を表示または選択することができます。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)
- [「\[DB2 カラムの選択\] ダイアログボックス」 \(ページ 159\)](#)

[コピーブックのインポート - リモートの詳細] ダイアログボックス

リモートコピーブックの情報を入力します。

ファイル名

リモートシステムのコピーブックの完全修飾ファイル名を入力します。

DBD で、分割データセット (PDS) ファイル名およびメンバを以下の形式で入力します。

HLQ.DTLDEMO(dbdname)

注: インポートする z/OS のデータセットは、データセットプロファイルで PACK オプションを使用してパック形式に作成してはなりません。

場所

コピーブックのインポート元の場所です。次のオプションがあります。

- **ローカル。** コピーブックは PowerExchange Navigator システムにあります。
- **リモート。** コピーブックはリモートシステムにあります。

【リモート】を選択すると、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が【場所】リストに表示されます。コピーブックが PowerExchange Navigator システムにある場合は、【ローカル】を選択します。

ユーザー ID

z/OS または i5/OS へのアクセスで、SECURITY 文の DBMOVER メンバに(1,x)または(2,x)が指定されている場合は、オペレーティングシステムのユーザー ID を入力します。

パスワード

z/OS システムのコピーブックへのアクセスまたは i5/OS のデータ記述仕様 (DDS) へのアクセスでは、指定されたユーザー ID のパスワードまたは有効な PowerExchange パスフレーズを入力します。

i5/OS のパスフレーズは、9~31 文字の長さで指定できます。z/OS のパスフレーズは、9~128 文字の長さで指定できます。パスフレーズには次の文字を含めることができます。

- 大文字および小文字
- 0~9 の数字
- スペース
- 次に示す特殊文字。
' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 最初の文字はアポストロフィです。

パスフレーズに一重引用符 (')、二重引用符 (")、通貨記号を含めることはできません。

z/OS では、IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。

名前を付けてローカルに保存

コピーブックのコピーが保存されたローカルシステムのパスとファイル名を入力するか、[参照] ボタンをクリックしてファイルを参照します。

名前

DDM に、DDM 名を入力します。

名前の参照

DDM で、[名前の参照] をクリックして DDM を参照します。

プレビュー

コピーブックを表示します。

IDMS コピーブックには使用できません。

DB 名

IDMS コピーブックで、IDMS データベース名を入力します。

詳細

IDMS コピーブックで、[詳細] をクリックして [IDMS 詳細プロパティ] ダイアログボックスに情報を入力します。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)
- [「\[IDMS 詳細プロパティ\] ダイアログボックス」 \(ページ 160\)](#)

[コピーブックのインポート - リモート Datacom の詳細] ダイアログボックス

リモート Datacom コピーブックの情報を入力します。

テーブル名 (長い名前)

テーブル名を入力します。

場所

Datacom コピーブックのインポート元の場所です。次のオプションがあります。

- **ローカル**。コピーブックは PowerExchange Navigator システムにあります。
- **リモート**。コピーブックはリモートシステムにあります。

[リモート] を選択すると、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が [場所] リストに表示されます。

ユーザー ID

z/OS で、セキュリティが有効な場合に必要なおペレーティングシステムのユーザー ID を入力します。

パスワード

z/OS の場合は、指定したユーザー ID のパスワードを入力します。

DB2 for z/OS サブシステムにアクセスする場合は、パスワードではなく有効な PowerExchange パスフレーズを入力できます。PowerExchange パスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。

- 大文字および小文字
 - 0～9 の数字
 - スペース
 - 次に示す特殊文字。
' - ; # \ , . / ! % & * () _ + { } : @ | < > ?
- 注: 最初の文字はアポストロフィです。

パスフレーズに一重引用符 (')、二重引用符 (")、通貨記号を含めることはできません。

IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: 有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange* リファレンスマニュアル』の「SECURITY 文」を参照してください。

名前を付けてローカルに保存

コピーブックのコピーが保存されたローカルシステムのパスとファイル名を入力するか、[参照] ボタンをクリックしてファイルを参照します。

ディクショナリのユーザー ID

Datacom ディクショナリのユーザー ID を入力します。

パスワード

ユーザー ID のパスワードを入力します。

テーブルの状態

読み取り専用。[製品] の値を指定します。

コピーブック

このオプションを選択して、次のうち 1 つのコピーブックの選択を有効にすることができます。

- COBOL
- PL/1

デフォルトでは、[コピーブック] オプションは選択されていません。デフォルトのコピーブックタイプは COBOL です。

キー

読み取り専用。キー情報がインポートされます。

デフォルトは選択されています。

エレメント

読み取り専用。エレメント情報がインポートされます。

デフォルトは選択されています。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)

[コピーブックのインポート - リモート DB2 カタログの詳細] ダイアログボックス

リモート DB2 コピーブックの情報を入力します。

DB インスタンス

DB2 データベースインスタンスの名前を入力します。

場所

DB2 カタログのインポート元の場所です。次のオプションがあります。

- **ローカル**。カタログは PowerExchange Navigator システムにあります。
- **リモート**。カタログはリモートシステムにあります。

【リモート】を選択すると、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が【場所】リストに表示されます。コピーブックが PowerExchange Navigator システムにある場合は、リストで【ローカル】を選択します。

テーブル名

テーブル名を入力するか、[参照] ボタンをクリックして参照し、テーブルに対するフィルタ条件を入力します。

ユーザー ID

データベースにアクセスする権限を持つユーザー ID を入力します。

サポートされている Linux、UNIX、または Windows システムのデータベースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

パスワード

指定したユーザー ID のパスワードを入力します。

z/OS の DB2 カタログにアクセスする場合は、パスワードではなく有効な PowerExchange パスフレーズを入力できます。PowerExchange パスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。
' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符 (')、二重引用符 (")、通貨記号を含めることはできません。

IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: 有効な RACF のパスフレーズは最長 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange* リファレンスマニュアル』の「SECURITY 文」を参照してください。

名前を付けてローカルに保存

コピーブックのコピーが保存されたローカルシステムのパスとファイル名を入力するか、[参照] ボタンをクリックしてファイルを参照します。

Null インジケータ

DB2 アンロードファイルデータマップに、アンロードファイルに使用されている Null インジケータの 16 進値を入力します。

変数のパディング

DB2 アンロードファイルデータマップに、アンロードファイルの変数が埋め込まれている場合にこのオプションを選択します。

注: このオプションは、BMC UNLOAD PLUS FIXEDVARCHAR に対応しています。

カラムのプレビュー/変更

[DB2 カラムの選択] ダイアログボックスを表示します。それによりアクセスされるテーブル内の列を表示または選択することができます。

関連項目：

- [「データマップへのコピーブックのインポート」 \(ページ 144\)](#)
- [「\[DB2 カラムの選択\] ダイアログボックス」 \(ページ 159\)](#)

[DB2 カラムの選択] ダイアログボックス

DB2 カタログをインポートするときの DB2 列を選択する情報を入力します。

利用可能なカラム

選択されたテーブルで個々の列を選択または取り消します。

すべて選択

[すべて選択] をクリックしてすべての列を選択します。

すべて選択解除

[すべて選択解除] すべての列を取り消します。

関連項目：

- [「Adabas データマップへの FDT のインポート」 \(ページ 148\)](#)

[IDMS 詳細プロパティ] ダイアログボックス

IDMS コピーブックの情報を入力します。

スキーマ名

IDMS データベースのスキーマ名を入力します。

スキーマバージョン

IDMS データベーススキーマのバージョンを選択します。次のオプションがあります。

- 最高
- 最低

ディクショナリのユーザー ID

必要に応じて、ユーザー ID を入力して IDMS スキーマまたはサブスキーマにアクセスします。

ディクショナリのパスワード

ユーザー ID のパスワードを入力します。

ノード名

以下のディクショナリを含むノードの DDS ノード名を入力します。

//SYSIDMS

ディクショナリノード

IDMSSUBSC ユーティリティまたは IDMSCHEM ユーティリティの以下のディクショナリを含むノード名を入力します。

//SYSIPT

[コピーブックのインポート - 構成の詳細] ダイアログボックス

インポートしたレコード、フィールド、およびテーブルに対するデフォルトアクションを定義するオプションを選択します。

以下の表に、オプションを示します。

オプション	説明
レコードのインポート時に確認する	レコードをインポートするときにレコード名に対して指示を出すにはこのオプションを選択します。デフォルトではオンに設定されています。
フィールドのインポート時に確認する	フィールドをインポートするときにフィールド名に対する指示を出すにはこのオプションを選択します。デフォルトではオフに設定されています。
テーブルの作成時に確認する	テーブルを作成するときにテーブル名に対する指示を出すにはこのオプションを選択します。デフォルトではオフに設定されています。
インポートされたレコードごとにテーブルを作成する	インポートされた各レコードに基づいてテーブルを作成するには、このオプションを選択します。デフォルトではオンに設定されています。
DL1 階層パス用テーブルを作成する	DBD について、すべての DL/1 階層パスに対するすべてのテーブルを作成するにはこのオプションを選択します。デフォルトではオフに設定されています。
IDMS 階層パス用テーブルの作成	IDMS コピーブックで、このオプションを選択して、すべての IDMS 階層パスのすべてのテーブルを作成します。デフォルトではオフに設定されています。 このオプションを選択した場合、[IDMS セットのリレーションの式を作成する] オプションは使用できません。
IDMS セットのリレーションの式を作成する	IDMS セットのリレーションを表す式を作成します。アプリケーションではこのメタデータを使用して、テーブル間の論理外部キーのリレーションを決定します。 このオプションを選択すると、データマップ内のレコードごとに次のフィールドが PowerExchange によって生成されます。 - GetDatabaseKey 関数を呼び出すフィールド。 - レコードがオーナーであるセットごとに、GetDbKeyOfFirstMember および GetDbKeyOfLastMember 関数を呼び出すフィールド。 - レコードがメンバであるセットごとに、GetDbKeyOfOwner、GetDbKeyOfNextMember、および GetDbKeyOfPriorMember 関数を呼び出すフィールド。 このオプションを選択した場合、[IDMS 階層パス用テーブルの作成] オプションは使用できません。
インポートされたレコードに対してテーブルのカラムをリフレッシュする	IDMS、IMS DL/1 バッチまたは IMS ODBA データマップにインポートされた COBOL または PL/I コピーブックについて、コピーブックのインポート処理中に上書きされるレコードに基づいたテーブルで列を更新するにはこのオプションを選択します。デフォルトではオフに設定されています。
最初のデータ再定義を選択する	PowerExchange で REDEFINES 句を見つけた場合に、最初のデータ定義を使用するにはこのオプションを選択します。デフォルトではオフに設定されています。

オプション	説明
自動的にインポートを開始する	自動的にインポートを開始するにはこのオプションを選択します。デフォルトではオンに設定されています。
レコードが重複したときのアクション フィールドが重複したときのアクション テーブルが重複したときのアクション	PowerExchange で、インポート処理中に重複レコード、フィールド、またはテーブルを見つけた場合に実行するアクションを指定します。 次のオプションがあります。 <ul style="list-style-type: none"> - プロンプト。実行するアクションを要求します。 - 一意の名前。一意の名前を使用するレコードまたはセグメントをインポートします。 - 上書き。レコードまたはセグメントを上書きします。 - スキップ。レコードまたはセグメントをスキップします。 - 追加。コピーブックからのレコードまたはセグメントをデータマップ内のレコードまたはセグメントに追加します。 デフォルトは PROMPT です。

「レコード定義」ダイアログボックスと「重複レコード定義」ダイアログボックス

インポートした各レコードまたはセグメントに対するアクションを選択します。

コピーブックに、データマップのレコードやフィールドと重複するレコードやフィールドが含まれている場合、重複項目の扱い方法を示すオプションを選択できます。例えば、項目のインポート、上書き、スキップ、またはインポートの停止を選択できます。

以下の表に、「**レコード定義**」ダイアログボックスと「**重複レコード定義**」ダイアログボックスで選択できるオプションを示します。

オプション	説明
インポート	項目をインポートするにはこのオプションを選択します。 デフォルトは選択されています。
名前	インポートした項目の名前を入力します。 デフォルトはコピーブックファイル内の名前です。
スキップ	項目のインポートをスキップするにはこのオプションを選択します。 デフォルトでは選択されていません。
インポートの停止	インポートを停止するにはこのオプションを選択します。 デフォルトでは選択されていません。
一意名の生成	重複項目の一意の名前を生成するにはこのオプションを選択します。 デフォルトでは選択されていません。

オプション	説明
上書き	<p>PowerExchange で重複項目を見つけた場合にその項目を上書きするにはこのオプションを選択します。</p> <p>このオプションを選択した場合、項目タイプを選択して以下のうち 1 つを上書きします。</p> <ul style="list-style-type: none"> - フィールドのみ - レコード <p>デフォルトでは選択されていません。</p>
追加	<p>このオプションを選択してコピーブックからの重複項目をデータマップの項目に追加します。</p> <p>デフォルトでは選択されていません。</p>
レコードの参照	<p>参照するデータマップのレコードを選択します。</p>
残りのインポートセッションへの適用	<p>選択したオプションを残りのインポートセッションに適用するにはこのオプションを選択します。</p> <p>デフォルトでは選択されていません。</p>

第 5 章

登録グループおよびキャプチャ登録

この章では、以下の項目について説明します。

- [登録グループおよびキャプチャ登録の概要, 164 ページ](#)
- [登録タグ名, 165 ページ](#)
- [登録グループ, 169 ページ](#)
- [登録グループの追加, 169 ページ](#)
- [キャプチャ登録の追加, 175 ページ](#)
- [キャプチャ登録の表示, 183 ページ](#)
- [登録グループのプロパティの表示または編集, 184 ページ](#)
- [キャプチャ登録のプロパティの表示または編集, 186 ページ](#)
- [Datacom キャプチャ登録のエLEMENTの表示, 191 ページ](#)
- [IDMS キャプチャ登録のパスの表示, 192 ページ](#)
- [キャプチャ登録での DB2 カラムの変更, 192 ページ](#)
- [キャプチャ登録の削除, 192 ページ](#)
- [登録グループの削除, 193 ページ](#)
- [登録グループのすべてのキャプチャ登録の削除, 194 ページ](#)

登録グループおよびキャプチャ登録の概要

PowerExchange で変更をキャプチャするには、変更データキャプチャ（change data capture : CDC）に、ソースデータセット、データベース、セグメントまたはテーブルを登録する必要があります。

登録グループおよびキャプチャ登録では PowerExchange で変更をキャプチャするソースデータが定義され、登録グループおよびキャプチャ登録は CDC 処理で必要となります。

登録グループでは、データソース場所、データソースタイプ、コレクション ID、またオプションとしてソースデータにアクセスできるユーザー ID およびパスワードが定義されます。登録グループを追加する場合は、PowerExchange により、抽出グループおよびアプリケーショングループが同じ名前で追加されます。

登録グループには、1 つ以上のキャプチャ登録が含まれています。キャプチャ登録では、変更をキャプチャするデータソースが指定されます。また、キャプチャ登録では、使用する圧縮オプション、および登録ステータスも指定されます。キャプチャ登録名とは、登録の作成時に指定する名前です。

キャプチャ登録を追加する場合は、PowerExchange により、その登録に抽出マップが追加されます。また、PowerExchange により、キャプチャ登録と生成された抽出マップの両方に登録タグ名が割り当てられます。キャプチャ登録に追加の抽出マップを手動で追加する場合、抽出マップには関連するキャプチャ登録と同じ登録タグ名が付けられます。登録タグ名形式は、ソースタイプによって異なります。

Adabas、IMS、および VSAM ソースなどの非リレーショナルソースのキャプチャ登録を作成するには、まずソースのデータマップを作成する必要があります。非リレーショナルソースでは、データマップにより、キャプチャ登録と CDC 用抽出マップの定義に使用されるメタデータが提供されます。

注: PowerExchange Navigator は、変更データがキャプチャできない特定のタイプのソースオブジェクトのキャプチャ登録を作成しません。詳細については、データソースタイプについて記載した PowerExchange CDC ガイドを参照してください。

CCT ファイルには、PowerExchange Navigator に定義されたキャプチャ登録情報が含まれています。登録グループでは、PowerExchange Navigator で表示される CCT ファイルからのキャプチャ登録がフィルタリングされます。

PowerExchange は、PowerExchange が実行する機能に応じて、読み取りモードまたは更新モードで CCT ファイルを開きます。

次の状況で登録情報が読み取られると、CCT ファイルは読み取りモードで開かれます。

- PowerExchange Logger (Linux, UNIX, and Windows 用) が起動したとき
- z/OS または IBM i の Condense プログラムが開始したとき
- PowerExchange Navigator で登録情報を表示するとき

次の状況で登録情報が追加、更新、または削除されると、CCT ファイルは更新モードで開かれます。

- DTLURDMO を使用してあるシステムから別のシステムに登録情報を移動する
- DTLUCBRG を使用して登録情報を追加する
- DTLUI を使用して登録情報を追加、更新、または削除する

キャプチャ登録を追加した後に、そのプロパティを変更することができます。

関連項目：

- [「登録グループ」 \(ページ 169\)](#)

登録タグ名

登録タグ名は、PowerExchange により生成され、CDC のために登録されたデータソースに割り当てられる一意の識別子です。

PowerExchange では、これらの登録タグを使用して、CDC 処理の対象となる変更レコードを特定します。z/OS 用 PowerExchange ロgger、PowerExchange Condense、および PowerExchange ロgger (Linux、UNIX、Windows 用) では、ログまたは圧縮ファイル内のそれぞれの変更レコードと共に登録タグ名がファイルに格納されます。

z/OS ECCR および z/OS 用 PowerExchange ロggerによって処理された z/OS データソースでは、登録タグ名は EDMNAME としても参照されます。

キャプチャ登録を作成する場合は、PowerExchange により、キャプチャ登録とそれに対応する PowerExchange で生成された抽出マップに対して同じ登録タグ名が割り当てられます。キャプチャ登録に対して他の抽出マップを手動で追加した場合、この追加の抽出マップには同じ登録タグ名が付けられます。

登録タグ名形式は、ソースタイプによって異なります。以下のリストに、各ソースタイプの登録タグ名の形式を示します。

注: このリストでは、読みやすくなるように、登録タグ名を構成する異なるタイプの値がアンダースコアで区切られています。タグ名にはアンダースコアは表示されません。

Adabas

ADAinstanceDBIDnnnnnFILEIDnnnnn

この登録タグ名形式は次の値によって構成されています。

- *ADAinstance*。プレフィックス「ADA」の後に、登録グループの作成時に指定したコレクション識別子が続きます。
- *DBIDnnnnn*。Adabas データベース ID (DBID) 番号です。
- *FILEIDnnnnn*。Adabas ファイル番号です。

Datcom テーブルベース

DCMMUFnameDBIDdatabaseIDSTNshortTableNameVregVersion

この登録タグ名形式は次の値によって構成されています。

- *DCMMUFname*。プレフィックス「DCM」の後に、登録グループの作成時に指定した Datcom MUF 名が続きます。
- *DBIDdatabaseID*。Datcom データベース ID です。
- *STNshortTableName*。PowerExchange が、Datcom テーブル用の [レコードプロパティ] ダイアログボックス内の **【短いテーブル名】** フィールドに表示する 3 文字の短縮形式のテーブル名です。
- *VregVersion*。PowerExchange が生成する登録バージョン番号です。

例: DCMR11G4DBID00800STNTB1V1

DB2 for Linux, UNIX, and Windows

UDBdatabase_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *UDBdatabase*。プレフィックス「UDB」の後に、登録グループの作成時に指定した DB2 for Linux, UNIX, Windows のデータベース名が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

DB2 for z/OS

DB2instance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *DB2instance*。プレフィックス「DB2」の後に、登録グループの作成時に指定した DB2 for z/OS データベースインスタンスが続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

DB2 for i (i5/OS)

AS4instance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *AS4instance*。プレフィックス「AS4」の後に、登録グループの作成時に指定したコレクション識別子が続きます。

- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

IDMS ログベース

IDL logsid_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *IDL logsid*。プレフィックス「IDL」の後に、登録グループの作成時に指定した Logsid 値が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

IMS ログベース

IML reconid_registrationName100000

この登録タグ名形式は次の値によって構成されています。

- *IML reconid*。プレフィックス「IML」の後に、登録グループの作成時に指定した IMS RECON 識別子が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- 100000。固定のバージョン番号です。

IMS 同期

IMS.databaseName.segmentName

この登録タグ名形式は次の値によって構成されています。

- *IMS.databaseName*。プレフィックス「IMS」の後に、データマップの作成時に指定した IMS データベース名が続きます。
- *segmentName*。IMS セグメント名です。

Microsoft SQL Server

MSS instance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *MSS instance*。プレフィックス「MSS」の後に、登録グループの作成時に指定した Microsoft SQL Server データベース名から生成されたインスタンスの値が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

MySQL

MYS instance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *MYS instance*。プレフィックス「MYS」の後に、登録グループの作成時に指定した MySQL インスタンス名が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

Oracle

ORA instance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *ORAinstance*。プレフィックス「ORA」の後に、登録グループの作成時に指定したコレクション識別子が続きます。
- *registrationName*。キャプチャ登録に割り当てた登録名です。
- *n*。生成されたバージョン番号。

PostgreSQL

PGSinstance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *PGSinstance*。プレフィックス「PGS」の後に、登録グループの作成時に指定した PostgreSQL インスタンス名が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

SAP HANA

HANAinstance_registrationName_n

この登録タグ名形式は次の値によって構成されています。

- *HANAinstance*。プレフィックス「PGS」の後に、登録グループの作成時に指定した PostgreSQL インスタンス名が続きます。
- *registrationName*。キャプチャ登録用に指定した登録名です。
- *n*。生成されたバージョン番号。

VSAM

VSAM データセットに割り当てられる登録タグ名は、データマップの VSAM データセット名の長さによって異なります。

- VSAM データマップセット名が 28 文字以下の長さである場合、登録タグ名は以下の形式になります。

VSAMdatasetName

たとえば、USERID1.VSAM.EDMSRCV.MAKEITA というデータセット名は 28 文字の長さです。このデータセットに対して PowerExchange は次の登録タグ名を生成します。

VSAMUSERID1.VSAM.EDMSRCV.MAKEITA

- VSAM データセット名が 29 文字以上である場合、PowerExchange はデータセット名を 23 文字に切り詰め、*nnnnn* という昇順の数字を付加した登録タグ名を作成します。登録タグ名は次の形式になります。

VSAMdataset_namennnnn

たとえば、USERID1.VSAM.EDMSRCV.MAKEITAB というデータセット名は 29 文字の長さです。このデータセットに対して PowerExchange は次の登録タグ名を生成します。

VSAMUSERID1.VSAM.EDMSRCV.MA00001

- 後続のデータセット名が 28 文字を超えており、同じく 28 文字を超える直前のデータセット名の最初の 23 文字がこれと同一である場合、PowerExchange ナビゲータは追加する *nnnnn* の数字を 1 ずつ増分します。

VSAMUSERID1.VSAM.EDMSRCV.MAnnnnn+1

たとえば、データセット名が USERID1.VSAM.EDMSRCV.MAKEITAV である場合、PowerExchange は次の登録タグ名を生成します。

VSAMUSERID1.VSAM.EDMSRCV.MA00002

登録グループ

キャプチャ登録を追加する前に、登録グループを追加する必要があります。

グループでは、データソース場所、データソースタイプ、コレクション ID、またオプションとしてソースデータにアクセスできるユーザー ID およびパスワードが定義されます。登録グループを追加する場合は、PowerExchange により、抽出グループおよびアプリケーショングループも同じ名前で作成されます。また、抽出グループまたはアプリケーショングループを手動で追加することもできます。

PowerExchange では、登録グループ、抽出グループ、アプリケーショングループ情報は、そのグループが追加された Windows システムに格納されます。グループはそのシステムに対して一意です。

登録グループ、抽出グループ、またはアプリケーショングループを削除する場合は、PowerExchange により、PowerExchange Navigator システムのローカルグループ情報が削除されます。PowerExchange では、キャプチャ登録、抽出マップ、または削除されたグループのアプリケーションは削除されません。また、PowerExchange により生成されたアプリケーショングループまたは抽出グループを削除する場合、PowerExchange では、このアプリケーショングループまたは抽出グループの作成を生成した登録グループは削除されません。

別のグループを、削除したグループとして同じプロパティで追加する場合、新しいグループには、削除されたグループ用に存在するすべてのキャプチャ登録、抽出マップ、またはアプリケーションが含まれます。

関連項目：

- [「登録グループの追加」 \(ページ 169\)](#)
- [「アプリケーショングループの追加」 \(ページ 346\)](#)
- [「抽出グループの追加」 \(ページ 198\)](#)

登録グループの追加

登録グループとは、同じソースタイプとインスタンスを持つ、キャプチャ登録の名前が付いたグループのことです。

登録グループは、キャプチャ登録を追加する前に追加する必要があります。キャプチャ登録は、登録グループの定義直後またはそれ以降に追加できます。

1. リソースエクスプローラの **【リソース】** タブで **【登録グループ】** を右クリックし、**【登録グループの追加】** をクリックします。
2. **【登録グループの追加】** ダイアログボックスに、以下の情報を入力します。

名前

登録グループのユーザー定義名を入力します。この値には、最大 16 文字の英数字を指定できます。スペースや特殊文字を入力することはできません。

場所

変更をキャプチャするソースの場所を選択します。このフィールドは、IDMS_L を除くすべてのソースタイプに適用されます。

場所は、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義されているノード名、またはローカル Windows システム（そこで Navigator が実行）の値「local」です。

デフォルトは【ローカル】です。ソースが PowerExchange Navigator システムにある場合は、【ローカル】を選択します。

注: リモートの Microsoft SQL Server ソースの場合は、実際のサーバー名を指定します。サーバーの指定にローカルのエイリアスは使用しないでください。PowerExchange では、登録作成時に、リモートサーバーに接続し、SQL Server パブリケーションを作成するときのローカルエイリアスの使用はサポートされていません。

タイプ

データソースタイプを選択します。次のオプションがあります。

- **ADABAS** (Adabas ファイルの場合)
- **AS4** (DB2 for i (i5/OS) テーブルの場合)
- **DATACOM** (CA Datacom/DB テーブルの場合)
- **DB2** (DB2 for z/OS テーブルの場合)
- **DB2UDB** (DB2 for Linux, UNIX, and Windows テーブルの場合)
- **HANA** (SAP HANA テーブルの場合)
- **IDMS_L** (IDMS ログベースの CDC に対して登録される IDMS テーブルの場合)
- **IMS**、IMS データベースの場合
- **MSSql** (Microsoft SQL Server テーブルの場合)

注: PowerExchange Navigator が Microsoft SQL Server ソースへのキャプチャ登録を作成するには、PowerExchange Navigator システムの license.key ファイルに、SQL Server CDC の使用を承認するキーが含まれる必要があります。そうでない場合、SQL Server ソースへのキャプチャ登録を追加しようとすると、PowerExchange Navigator がライセンスキーエラーとともに失敗します。この失敗は、SQL Server CDC を含むライセンスキーを使用して PowerExchange Listener を実行している場合でも発生します。

- **MySQL** (MySQL テーブルの場合)
- **ORACLE**、Oracle テーブルの場合
- **PostgreSQL** (PostgreSQL テーブルの場合)
- **VSAM**、VSAM ESDS、KSDS、RRDS、および VRRDS データセットの場合

ユーザー ID

【場所】 フィールドで指定されたソースノードへのアクセスを許可するユーザー ID を入力します。このフィールドは、IDMS_L を除くすべてのソースタイプに適用されます。

ユーザー ID のタイプは、次のソースタイプおよび PowerExchange のセキュリティ設定によって異なります。

- ソースが IBM i システムまたは z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を(1,x)または(2,x)に設定すると、PowerExchange のセキュリティが有効になります。
- DB2 for Linux, UNIX, and Windows、Microsoft SQL Server、MySQL、Oracle、または PostgreSQL ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。

注: Microsoft NTLM および Active Directory 認証を使用して Microsoft SQL Server ソースへのアクセスを制御する場合は、`domain\user_name` の形式でユーザー ID を入力します。次に、**【パスワード】** フィールドに、このユーザー ID の有効なパスワードを入力します。

- サポートされている Linux、UNIX、または Windows システムのソースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

パスワード

ソースノードへのアクセスを許可する指定のユーザー ID のパスワードを入力します。このフィールドは、IDMS_L を除くすべてのソースタイプに適用されます。

ソースが IBM i または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。IBM i のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。
' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。

z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N) 以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

ユーザー ID を指定してパスワードやパスフレーズを省略した場合、登録グループを開いたときに PowerExchange Navigator は **【パスワードが入力されていません】** ダイアログボックスにパスワードの入力を求めるメッセージを表示します。

ソース固有のフィールド

【タイプ】 フィールドに入力したソースタイプに応じて、ソース固有のフィールドが表示されます。次の表は、これらのフィールドをソースタイプ別に示します。

ソースタイプ	ソース固有のフィールド
ADABAS	【コレクション識別子】 フィールドに、インスタンス名を入力します。
AS4（Db2 for i）	【コレクション識別子】 フィールドに、ソースシステムの <code>dtllib/CFG</code> ファイルの DBMOVER メンバで、AS4J CAPI_CONNECTION 文の INST パラメータに指定されたインスタンス名を入力します。
DATACOM	【MUF 名】 フィールドに、Multi-User Facility（MUF）名を入力します。

ソースタイプ	ソース固有のフィールド
DB2 (DB2 for z/OS)	【データベースインスタンス】 ボックスに、DB2 サブシステム ID、または DB2 データ共有グループ名を入力します。
DB2UDB (DB2 for Linux, UNIX, and Windows)	【データベース】 フィールドに DB2 データベースの名前を入力します。
IDMS_L	<ul style="list-style-type: none"> - 【登録場所】 リストで、キャプチャ登録の格納場所を選択します。場所は、PowerExchange Navigator システム上の dbmover.cfg 構成ファイルの NODE 文に指定したノード名です。 - 登録場所で 【ユーザー ID】 および 【パスワード】 フィールドに、その場所へのアクセスを可能にするユーザー名とパスワードを入力します。登録が z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を (1,x) または (2,x) に設定すると、PowerExchange のセキュリティが有効になります。 - 【ソースマップの場所】 リストで、IDMS ソーステーブル用のデータマップのある場所を選択します。 - ソースマップの場所で 【ユーザー ID】 および 【パスワード】 フィールドに、その場所へのアクセスを可能にするユーザー名とパスワードを入力します。マップが z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。 - 【Logsid】 フィールドに、ソースシステムの DBMOVER 構成ファイルの LOGSID 文で指定された値を入力します。
IMS	【RECON 識別子】 フィールドに以下のいずれかの値を入力します。 <ul style="list-style-type: none"> - IMS 同期 CDC の場合は、IMS システム ID を入力します。 - IMS ログベース CDC の場合は、RECON データセットのある z/OS システムの DBMOVER コンフィギュレーションメンバにある IMSID 文で指定された値を入力します。

ソースタイプ	ソース固有のフィールド
MSSQL	<ul style="list-style-type: none"> - 【データベースサーバー】 リストで、SQL Server のソースが存在するサーバを選択します。デフォルトのポート番号 1433 以外のポートを使用する場合は、必要に応じて SQL Server インスタンス名とポート番号を追加できます。次の形式を使用します。 <code>server[\\instance_name][,port_number]</code> 例: MSQL001\\MSQLINST5, 6983 インスタンス名を含める場合は PowerExchange Navigator で SQL Server Browser を使用できます。ただし、ポートを指定していない場合や、ポートを指定したが最初の接続試行に失敗した場合は、SQL Server に接続するポートを検索するため、SQL Server Browser がアクティブになっている必要があります。 注: デフォルトのポートを使用せず、このフィールドでインスタンスまたはポート番号を指定しない場合、このグループのソーステーブルでキャプチャ登録を作成しようとしても失敗します。 - 【データベース名】 リストで、SQL Server パブリケーションデータベース名を選択または入力します。 - 【インスタンス】 フィールドに、指定されたデータベースサーバーとデータベース名の組み合わせの、一意のユーザー定義識別子をオプションで入力します。最大長は 7 文字です。この識別子は、パブリケーションデータベースの一連の登録を識別するために使用します。また、この識別子は、登録グループでキャプチャ登録用に生成される抽出マップの名前にも組み込まれます。 PowerExchange ロgger（Linux、UNIX、Windows 用）を使用する場合は、インスタンス識別子がロgger構成ファイルの DBID パラメータ値と一致していることを確認します。インスタンス値を入力しないと、PowerExchange は一意のインスタンス識別子を生成します。これは、パブリケーションデータベース名のすべてまたは一部と、それに続く 3 桁の数字（値を一意にするために必要な場合）で構成されます。 このユーザー定義のインスタンス識別子は移行シナリオで役立ちます。テスト環境から本番環境などのように、変更キャプチャをある環境から別の環境に移行する必要があり、インスタンス識別子を定義しない場合、PowerExchange は新しい環境で生成済みのインスタンス識別子を使用します。生成済みインスタンス識別子は元の環境のものと異なる場合があります。PowerCenter ワークフローで抽出マップ名を更新し、PowerExchange ロggerの DBID パラメータ値を編集する必要性を回避するには、新しい環境で登録を作成するときに、元の環境のインスタンス識別子と一致するインスタンス識別子を入力します。 ヒント: この移行シナリオでは、dbmover.cfg 構成ファイルの CAPT_PATH 文および CAPT_XTRA 文のパスが元の環境と新しい環境で一意であることを確認します。

ソースタイプ	ソース固有のフィールド
MySQL	<ul style="list-style-type: none"> - 【データベースサーバー】 フィールドで、ソースデータベースが配置されている MySQL サーバーのサーバー名または IP アドレスを入力します。MySQL サーバーが Navigator が実行されているローカルホストに配置されている場合は、localhost と入力します。オプションとして、デフォルトのポート 3306 以外のポートを使用している場合はポート番号を追加できます。次の形式を使用します。 <code>server[,port_number]</code> 以下に例を示します。"localhost,3400" - オプションで、【インスタンス】 フィールドに、【データベースサーバー】 フィールドで指定された MySQL サーバーの一意のユーザー定義識別子を入力します。最大長は 7 文字です。この識別子は、MySQL サーバーのデータベース内のテーブルに対する一連の登録を識別するために使用します。また、この識別子は、登録グループでキャプチャ登録用に生成される抽出マップの名前にも組み込まれます。PowerExchange ロgger（Linux、UNIX、Windows 用）を使用する場合は、インスタンス識別子がロgger構成ファイルの DBID パラメータ値と一致していることを確認します。【インスタンス】 の値を入力しないと、PowerExchange は一意のインスタンス識別子を生成します。これは、データベースサーバー名の最初の 4 文字と、それに続く 3 桁の数字（識別子を一意にするために必要な場合）で構成されます。
ORACLE	【コレクション識別子】 フィールドに、ソースシステムの DBMOVER 構成ファイルの ORACLEID 文で指定された Oracle インスタンス ID を入力します。
PostgreSQL	<ul style="list-style-type: none"> - 【データベースサーバー】 リストで、PostgreSQL ソースが存在するサーバーを選択します。オプションで、デフォルトのポート 5432 以外のポートを使用している場合はポート番号を追加できます。次の形式を使用します。 <code>server[,port_number]</code> 以下に例を示します。PGSL001,6983 注: サーバーがデフォルトのポートを使用せず、このフィールドにポート番号を指定しない場合、この登録グループのキャプチャ登録を作成しようとすると失敗します。 - 【データベース名】 リストで、指定したサーバー上の PostgreSQL ソースデータベースの名前を選択します。 - 【インスタンス】 フィールドに、指定されたデータベースサーバーとデータベース名の組み合わせの、一意のユーザー定義識別子をオプションで入力します。最大長は 7 文字です。この識別子を使用して、データベースの一連の登録を識別します。また、この識別子は、登録グループでキャプチャ登録用に生成される抽出マップの名前にも組み込まれます。PowerExchange ロgger（Linux、UNIX、Windows 用）を使用する場合は、インスタンス識別子がロgger構成ファイルの DBID パラメータ値と一致していることを確認します。この値を入力しないと、PowerExchange は一意のインスタンス識別子を生成します。これは、データベースサーバー名の最初の 4 文字と、それに続く 3 桁の数字（識別子を一意にするために必要な場合）で構成されます。

ソースタイプ	ソース固有のフィールド
SAP HANA	<ul style="list-style-type: none"> - 【データベースサーバー】 リストで、SAP HANA のソースが存在するサーバーを選択します。オプションで、デフォルトのポート番号 39015 以外のポートを使用している場合はポート番号を追加できます。次の形式を使用します。 <code>server[,port_number]</code> 以下に例を示します。PGSL001,6983 注: サーバーがデフォルトのポートを使用せず、このフィールドにポート番号を指定しない場合、この登録グループのキャプチャ登録を作成しようとすると失敗します。 - 【データベース名】 リストで、指定したサーバー上の SAP HANA ソースデータベースの名前を選択します。 - 【インスタンス】 フィールドに、指定されたデータベースサーバーとデータベース名の組み合わせの、一意のユーザー定義識別子をオプションで入力します。最大長は 7 文字です。この識別子を使用して、データベースの一連の登録を識別します。また、この識別子は、登録グループでキャプチャ登録用に生成される抽出マップの名前にも組み込まれます。PowerExchange ロgger (Linux、UNIX、Windows 用) を使用する場合は、インスタンス識別子がロgger構成ファイルの DBID パラメータ値と一致していることを確認します。この値を入力しないと、PowerExchange は一意のインスタンス識別子を生成します。これは、データベースサーバー名の最初の 4 文字と、それに続く 3 桁の数字（識別子を一意にするために必要な場合）で構成されます。
VSAM	【コレクション識別子】 フィールドに、インスタンス名を入力します。

登録の追加

このオプションは、キャプチャ登録をグループに追加するときに選択します。このオプションの選択を取り消すと、登録グループのみが追加されます。

デフォルトでは、このオプションは選択されています。

- グループにキャプチャ登録を追加するか、次のように登録グループの定義を完成させます。
 - **【登録の追加】** オプションを選択した場合は、**【次へ】** をクリックして **「キャプチャ登録の追加」** (ページ 175) に進みます。
 - **【登録の追加】** オプションを選択しない場合は、**【完了】** をクリックします。リソースインスペクタの **【登録グループ】** タブに、登録グループ用のプロパティが表示されます。さらに、ソースシステムの PowerExchange リスナが稼働している場合は、リソースエクスプローラの **【登録グループ】** タブに、登録グループに関連付けられたソースインスタンス用に存在するすべてのキャプチャ登録が表示されます。

関連項目：

- **「登録グループ」** (ページ 169)
- **「キャプチャ登録の追加」** (ページ 175)

キャプチャ登録の追加

キャプチャ登録は、PowerExchange の変更キャプチャ処理のソースオブジェクト（IMS セグメント、DB2 テーブル、VSAM データセットなど）を定義します。

キャプチャ登録は、PowerExchange によりソースシステム上の CCT ファイルに保存されます。

キャプチャ登録を追加する前に、登録グループを定義する必要があります。登録グループ内では、特定のソースに対してキャプチャ登録を一度のみ追加することができます。このキャプチャ登録がソースオブジェクトのメタデータを提供し、変更データをキャプチャする列を特定します。

注: 列名の長さが 128 文字を超えるソースオブジェクトは登録することができません。

1. 登録グループを開きます。
2. **リソースエクスプローラ**の**【登録グループ】**タブで登録グループを右クリックし、**【キャプチャ登録の追加】**をクリックします。
3. **【キャプチャ登録の追加 - 名前およびテーブルフィルタ】**ダイアログボックスで、キャプチャ登録名を入力し、必要に応じて任意の情報を入力します。

以下の表に、このダイアログボックスのフィールドを示します。

フィールド	説明
名前	<p>必須。次の形式でのキャプチャ登録のユーザー定義名:</p> <ul style="list-style-type: none"> - z/OS ソースおよび IBMi ソースの場合、名前は 1 文字から 10 文字の長さで、小文字の英数字で構成され、英字で始まる必要があります。 - Linux, UNIX, and Windows ソースの場合、名前は 1 文字から 13 文字の長さで、小文字の英数字で構成され、英字で始まる必要があります。
テーブルフィルタ	<p>オプション。テーブルをフィルタリングするために、ソースタイプに応じて次のいずれかのフィルタフィールドに値を入力してください。</p> <ul style="list-style-type: none"> - Adabas の場合、【スキーマ】 ボックスと 【テーブル】 ボックスにフィルタ情報を入力します。両フィールドの最大長は 128 文字です。 - Datacom の場合、【スキーマ】 ボックスと 【テーブル】 ボックスにフィルタ情報を入力します。両フィールドの最大長は 128 文字です。 - すべての DB2 タイプに対し、【クリエータ/スキーマ】 ボックス、【DB 名/定義者】 ボックス、および 【テーブル】 ボックスにフィルタ情報を入力します。 DB2 for i5/OS の場合の最大長: 【クリエータ/スキーマ】 フィールドでは 10 文字、【テーブル】 フィールドでは 32 文字 DB2 for z/OS の場合の最大長: 【クリエータ/スキーマ】 フィールドでも 【テーブル】 フィールドでも 128 文字 - IDMS の場合、【スキーマ】 ボックスと 【テーブル】 ボックスにフィルタ情報を入力します。両フィールドの最大長は 128 文字です。 - IMS の場合、【スキーマ】 ボックス、【マップ名】 ボックス、および 【テーブル】 ボックスにフィルタ情報を入力します。【マップ名】 フィールドに、PowerExchange IMS データマップ名を入力します。最大長: 【スキーマ】 値と 【マップ名】 値は 10 文字、【テーブル】 値は 128 文字。 - MSSql の場合、【オーナー】 ボックスと 【テーブル】 ボックスにフィルタ情報を入力します。両フィールドの最大長は 128 文字です。 - MySQL の場合、【スキーマ】 フィールドと 【テーブル】 フィールドにフィルタ情報を入力します。両フィールドの最大長は 64 文字です。MySQL では、スキーマ名はデータベース名と同等です。 - Oracle の場合、【スキーマ】 フィールドと 【テーブル】 フィールドにフィルタ情報を入力します。両フィールドの最大長は 128 文字です。 - PostgreSQL の場合、【スキーマ】 フィールドと 【テーブル】 フィールドにフィルタ情報を入力します。両フィールドの最大長は 63 文字です。 - VSAM の場合、【スキーマ】 ボックス、【マップ名】 ボックス、および 【テーブル】 ボックスにフィルタ情報を入力します。【マップ名】 ボックスに、PowerExchange VSAM データマップの名前を入力します。最大長: 【スキーマ】 フィールドと 【マップ名】 フィールドは 10 文字、【テーブル】 フィールドは 128 文字。 <p>どのフィルターフィールドでも、次のワイルドカード文字を使用してマスクを作成できます。</p> <ul style="list-style-type: none"> - アスタリスク (*) : 1 つまたは複数の文字を表す。 - 疑問符 (?) 単一の文字を表す。

フィールド	説明
大文字と小文字を区別する	デフォルトでは、PowerExchange がフィルタボックスに入力したテキストを、データソースの基準に従って大文字または小文字に変換します。例えば、Oracle では大文字が標準です。値が scott でも SCOTT でも、SCOTT が生成されます。 PowerExchange によってデータソース側で標準の文字種に強制変換されては困るフィルタフィールドには、 【大文字と小文字を区別する】 オプションを選択します。
エスケープ文字	フィルタボックスで使用するエスケープ文字を入力して、アスタリスク (*) または疑問符 (?) を区切ります。ワイルドカードではなくリテラル値として解釈されます。 例えば、~をエスケープ文字に指定すると、フィルタ値が tab*の場合は名前が tab で始まるすべてのテーブルが返され、フィルタ値が tab~*の場合は tab*という名前のテーブルが返されます。 注: エスケープ文字は、マルチバイト言語のデータソースには使用できません。

4. **【次へ】** をクリックします。

【キャプチャ登録の追加 - テーブルとカラム】 ダイアログボックスが表示されます。

フィルタ条件を入力した場合、フィルタ条件と一致したテーブルが **【テーブル】** リストに表示されます。それ以外の場合は、**【テーブル】** リストにすべてのテーブルが表示されます。

【テーブル】 リストには、ソースタイプに応じて次のフィールドが 1 つ以上含まれます。

カラム	説明
クリエータ名	テーブルのクリエータの名前です。
スキーマ	スキーマ名です。
名前またはテーブル	テーブル名です。
タイプ	TABLE などのオブジェクトタイプです。
DataSet 名	データセット名です。

5. **【テーブル】** リストで、テーブルをダブルクリックして選択します。

注: テーブルに 128 文字より長い列名が含まれている場合は、エラーメッセージが表示され、そのテーブルを登録することはできません。

【カラム】 リストには、選択したテーブルのリストが表示されます。リストには、ソースタイプに応じて次のフィールドが含まれます。

フィールド	説明
カラムの選択	各カラムの隣にあるチェックボックスは、登録するカラムを選択するためのものです。 注: インデックスカラムはデフォルトで選択されます。
名前	列の名前です。
CCSID	DB2 ソース用の、列に関連したコード化文字セット ID (CCSID) です。

フィールド	説明
PWXCP	DB2 ソース用の、コードページの PowerExchange 内部表現です。
コードページ	DB2 ソース用の、列に関連したコードページです。

6. **【カラム】** リストで、PowerExchange で変更をキャプチャするカラムを選択します。

カラムを個別に選択するには、**【カラムの選択】** チェックボックスをオンにします。すべてのカラムを選択するには、**【すべてのカラムの選択】** または **【すべて選択し、変更を通知する】** をクリックします。以下の表に、すべてのカラムを選択するためのオプションを示します。

チェックボックス	説明
すべてのカラムの選択	<p>選択される列のキャプチャをサポートするソースタイプで使用可能です。変更キャプチャの 【カラム】 リスト内のすべての列を選択します。</p> <p>登録を追加した後は、ソーステーブルに追加される列はいずれも、キャプチャ登録を編集して追加された列を選択しない限り、無視されます。</p>
すべて選択し、変更を通知する	<p>DB2 ソースと Oracle ソースに対して使用できます。このオプションを選択した場合、PowerExchange により、すべての列の変更がキャプチャされます。テーブルのスキーマへの変更により、PowerExchange CDC は失敗し、エラーメッセージが記録されます。</p> <p>z/OS 用の DB2 ソースの場合、DB2 ECCR は、スキーマ変更後のテーブルの最初の変更レコードを読み取った後に異常終了します。</p> <p>Oracle ソースの場合、Oracle CDC は失敗し、以下の場合にエラーメッセージが記録されます。</p> <ul style="list-style-type: none"> - キャプチャ対象として登録したテーブルの変更レコードに、キャプチャ対象として登録していない列が含まれていた場合 - 変更レコードに、キャプチャ対象として登録した列が含まれていない場合 <p>Oracle CDC の場合、テーブルの定義が PowerExchange のキャプチャ登録と互換性のある方法で変更された場合、そのテーブルに対する変更のキャプチャが続行されます。</p> <p>例えば、文字カラムの長さが減少してもキャプチャ登録でその変更が反映されない場合は、Oracle CDC では、そのテーブルに対する変更のキャプチャが続行されます。</p> <p>反対に、カラムのデータタイプが数値から文字へ変更され、キャプチャ登録でその変更が反映されない場合、Oracle CDC では、そのカラムの数字以外のデータを含む最初の変更レコードを検出するまで、そのテーブルに対する変更のキャプチャが続行されます。Oracle CDC により、カラムの数字以外のデータを含む変更レコードが検出されると、Oracle CDC は失敗し、エラーメッセージが記録されます。</p> <p>注: 【すべて選択し、変更を通知する】 オプションをクリアすると、列の横にあったチェックボックスがクリアされます。列を再び選択するには、チェックボックスを選択する必要があります。</p>

ヒント: 列を選択する際は、次の推奨事項を考慮します。

- 登録できるのは特定のソースに対する登録グループで一度のみであるため、変更データをキャプチャするすべてのカラムを選択するお勧めします。対応する抽出マップを編集してカラムの選択を解除できます。
 - キーのシンボルは、列が要約処理に使用されるインデックス列であることを示します。インデックス列のチェックボックスを選択しない場合、その行の一意性を損なう可能性があります。
7. **【次へ】** をクリックします。
8. **【キャプチャ登録の追加 - 種類】** ダイアログボックスで、キャプチャ登録に関する追加情報を入力します。

以下の表に、このダイアログボックスのフィールドを示します。

フィールド	ソースタイプ	説明
圧縮	すべて	<p>PowerExchange Condense または PowerExchange ロgger (Linux、UNIX、Windows 用) が使用されているかどうか、および i5/OS または z/OS 上の PowerExchange Condense の処理のタイプです。次のオプションがあります。</p> <ul style="list-style-type: none"> - なし。PowerExchange は、登録したソースへの変更の処理に、PowerExchange Condense または PowerExchange ロgger (Linux、UNIX、Windows 用) を使用しません。 - 部分。PowerExchange が、PowerExchange Condense または PowerExchange ロgger を使用して、登録したソースに対する変更を処理します。PowerExchange Condense が部分圧縮処理を使用します。トランザクションが一貫することからこのオプションが推奨されます。 - フル。PowerExchange が、PowerExchange Condense の完全圧縮処理を i5/OS または z/OS のソースに使用しますが、いくつかの制限事項が適用されます。この方法では、トランザクションの一貫性が保持されません。 <p>[圧縮] オプションは後からでも、登録バージョン番号を増やさずに変更することができます。詳細については、「圧縮オプション」(ページ 182)を参照してください。</p>
ステータス	すべて	<p>変更キャプチャへの可用性に関するキャプチャ登録のステータスです。次のオプションがあります。</p> <ul style="list-style-type: none"> - 非アクティブ。PowerExchange は、ソースの変更キャプチャにこの登録を使用しません。このオプションが、初期ステータスの設定です。この登録を変更キャプチャでの使用対象にするには、このオプションを手動で「アクティブ」に設定する必要があります。 - アクティブ。PowerExchange はこの登録を使用してソースの変更をキャプチャすることができます。 <p>デフォルトは「非アクティブ」です。</p> <p>注: ステータスを「非アクティブ」から「アクティブ」に切り替え、変更キャプチャ処理の対象にすることができます。</p> <p>ステータス設定の詳細については、「ステータスオプション」(ページ 181)を参照してください。</p>
コメント	すべて	<p>キャプチャ登録に関するユーザー定義コメントです。</p>
データベース ID	Adabas	<p>変更のキャプチャ元のファイルが入っている Adabas データベースの DBID です。</p>
ファイル番号	Adabas	<p>データベース ID フィールドによって特定されるデータベース内の Adabas ファイルのファイル番号。</p>
プライマリデータセット名	IMS 同期	<p>IMS データベースのプライマリデータセットの名前。この名前は最大 44 文字の長さで指定できます。データベースタイプに応じて、プライマリデータセットは次のいずれかになります。</p> <ul style="list-style-type: none"> - フル機能のデータベース: プライマリデータセットは、データベースのプライマリ (または唯一) のデータセットグループが含まれるデータセットです。 - 高速バスデータベース: プライマリデータセットは、データベースの最初 (または唯一) の領域が含まれるデータセットです。 - HALDB データベース: プライマリデータセットは、データセット名に使用される最初のパーティションのデータセットプレフィックスです。

フィールド	ソースタイプ	説明
補足ロググループ名	Oracle	<p>ソースに対して作成する補足ロググループの名前です。PowerExchange により、キャプチャ登録の完了時に補足ロググループを作成するために DDL が生成されます。</p> <p>PowerExchange により Oracle 変更データがキャプチャされる各テーブルの補足ロググループを定義する必要があります。この補足ロググループには、キャプチャされる変更データのすべてのテーブル列が含まれている必要があります。</p>
DDL を今すぐ実行	Oracle	<p>Oracle ソースでは、キャプチャ登録の完了時に補足ロググループを作成するために、このオプションを選択して DDL を実行します。</p> <p>DDL を後で実行するには、このオプションの選択を取り消します。</p> <p>また、[補足ロググループ名] ボックスに値を入力する必要があります。</p> <p>注: Oracle キャプチャユーザー用 ORACAPT.SQL ファイルの GRANT 文には、DDL を実行するために必要な権限が含まれていません。</p>

9. **[完了]** をクリックします。
10. Oracle ソースの場合、**[補足ロググループ SQL をファイルに保存]** ダイアログボックスで、生成された DDL を含むファイルのファイル名を **[ファイル名]** フィールドに入力し、**[保存]** をクリックします。

PowerExchange は DDL を指定したファイルに保存します。デフォルトでは、このファイルは PowerExchange ルートインストールディレクトリにあります。

PowerExchange リスナにより、キャプチャ登録がソースシステム上の CCT ファイルに書き込まれます。PowerExchange Navigator では、キャプチャ登録は、**リソースエクスプローラ**の **[登録グループ]** タブのキャプチャ登録リストに表示されます。また、登録は次に示すインターフェイスにも表示されます。

- **キャプチャ登録**ウィンドウには、キャプチャ登録の列が表示されます。
- **リソースインスペクタ**の **[キャプチャ登録]** タブには、キャプチャ登録用のプロパティが表示されます。この矢印を使用して、リソースインスペクタの **[キャプチャ登録]** タブと **[登録グループ]** タブを切り替えます。
- ステータスバーには、オープン登録グループとキャプチャ登録に関する情報が表示されます。

関連項目：

- [「キャプチャ登録の表示」 \(ページ 183\)](#)
- [「ステータスオプション」 \(ページ 181\)](#)
- [「圧縮オプション」 \(ページ 182\)](#)

ステータスオプション

キャプチャ登録の**ステータスオプション**は、この登録が変更キャプチャに使用できるのか、一時的に停止中なのか、または履歴のためのみに保持されているのかを示します。

キャプチャ登録を作成するときは、その初期 **[ステータス]** 設定は**非アクティブ**です。

PowerExchange Navigator からは、ステータスの変更制限があります。次に示す 1 方向のトランジションが許可されています。

- **[非アクティブ]** > **[アクティブ]** > **[履歴]**
- **[一時停止中]** > **[履歴]**

次の説明を参考にして、使用するステータスオプションを決定します。

非アクティブ

ターゲットが具体化される前でソースからデータをキャプチャできない初期のステータス。

アクティブ

このステータスでは、キャプチャ登録に基づいて登録されたソースから変更がキャプチャされます。

履歴

変更キャプチャ用途には永久にキャプチャ登録を使用させないものの、監査や履歴の用途のためにキャプチャ登録を保持しているステータス。登録ステータスを「履歴」に設定した後は、次のアクションを実行することができません。

- 登録を削除し別の登録を同じ名前で作成する場合以外は、同じ登録名を再度使用します。
- 登録を設定して「アクティブ」に戻します。このソースに対して変更キャプチャを再開するには、キャプチャ登録をもう 1 つ作成する必要があります。

一時停止中

登録されたソースに対する変更キャプチャを一時的に停止するために PWXUCREG ユーティリティで SUSPEND_REGISTRATION コマンドを実行した後に設定されるステータス。PowerExchange Navigator からこのステータスを選択することはできません。このステータスになることができるのは、Adabas、Datcom テーブルベース、IDMS ログベース、および IMS ログベースのタイプの CDC データソースのみです。ステータスが一時停止中の際、登録されたソースに対して PowerExchange は変更をキャプチャしません。別の PWXUCREG ユーティリティコマンドを発行することによって、この登録を再び変更キャプチャに使用できる状態にすることができます。

注: PWXUCREG ユーティリティで ACTIVATE_REGISTRATION コマンドを発行することにより、ステータスを一時停止中からアクティブに戻することができます。詳細については、『PowerExchange ユーティリティガイド』を参照してください。

圧縮オプション

キャプチャ登録の【圧縮】オプションは、登録されたソースに対し PowerExchange Condense または PowerExchange ロgger（Linux、UNIX、Windows 用）の処理を使用できるかどうかを制御します。i5/OS または z/OS 上の PowerExchange Condense の場合、このオプションは要約処理のタイプも示しています。

PowerExchange Navigator から、登録をバージョンニングせずに既存のキャプチャ登録に関する【圧縮】オプションを変更することができます。例えば、[なし] から [部分] へ、または [フル] から [部分] に切り替えることができます。

次の圧縮オプションがあります。

なし

このキャプチャ登録が、PowerExchange Condense または PowerExchange ロgger（Linux、UNIX、Windows 用）による使用の対象外となります。

部分

このキャプチャ登録が、PowerExchange Condense または PowerExchange ロgger（Linux、UNIX、Windows 用）による使用の対象となります。PowerExchange Condense がこの登録を部分圧縮処理に使用します。

このオプションでは、ソースで正常に完了した UOW 内の変更のみが、PowerExchange Condense の要約ファイルまたは PowerExchange ロgger のログファイルに書き込まれます。PowerExchange は、単に最新の変更だけでなく、CDC に関係する列のすべての変更を、UOW の終了時刻に基づいて時間順に書き込みます。このオプションは、トランザクションの一貫性が保たれます。

Informatica では、PowerExchange Condense や PowerExchange ロgger を使用する予定がなくても【部分】 オプションを選択するよう推奨しています。PowerExchange Condense や PowerExchange ロgger の実装は後でも行うことができ、登録を更新する必要ありません。

フル

このキャプチャ登録が、i5/OS または z/OS 上の PowerExchange Condense による完全圧縮処理の対象となります。

PowerExchange Condense では、古い変更を新しい変更で置き換えることによってキーの付いた要約ファイルに変更を蓄積します。この方法では、トランザクションの一貫性が保持されません。

以下の制限は完全要約処理に適用されます。

- Adabas および IDMS ログベースの CDC ソースでは、完全要約処理は使用できません。
- i5/OS 上の PowerExchange Condense の場合、完全圧縮処理は、プライマリキーが指定されたテーブル、または一意のキーで定義された DDS ファイルでのみ使用可能です。
- z/OS の PowerExchange Condense では、完全要約処理は、キー列を指定するテーブルまたはデータマップで使用可能です。

ソースに一意のキーがある場合に限り、このオプションを使用します。PowerExchange Condense では、この一意のキーを圧縮ファイルのキーの一部として使用することで、操作前と操作後のイメージを 1 つに維持します。

ヒント: 完全圧縮処理を使用中に一意のキーを更新できるようにするには、PowerExchange Condense のコンフィギュレーションファイルで KEY_CHANGE_ALW を Y に設定する必要があります。この設定では、PowerExchange Condense はキーの更新を無視し完全圧縮の処理を継続します。

キャプチャ登録の表示

キャプチャ登録の列を表示することができます。

1. キャプチャ登録を含む登録グループを開きます。
2. 表示するキャプチャ登録を開きます。

キャプチャ登録ウィンドウが表示されます。以下の表に、ソースタイプに応じて登録された各カラムに表示されるフィールドを示します。

フィールド	説明
名前	カラムの名前です。
テーブル名	カラムを含むテーブルの名前です。
カラム番号	カラム番号です。
タイプ	カラムのデータタイプです。
長さ	カラムの長さです。
スケール	データの位取りです。

フィールド	説明
キー	カラムがインデックスキーカラムであるかどうかを示す、YES または NO の値です。
CCSID	DB2 ソース用の、カラムで使用されるコードページのコード化文字セット ID (CCSID) です。
PWXCP	DB2 ソース用の、カラムで使用される PowerExchange のコードページです。
コードページ	DB2 ソース用の、カラムで使用されるコードページです。
NULL 可能	カラムが NULL 可能であるかどうかを示す、YES または NO の値です。

3. カラムをカラム名別にアルファベット順でソートするには、**キャプチャ登録**ウィンドウの【名前】列の見出しをクリックします。

登録グループのプロパティの表示または編集

登録グループのプロパティを表示または編集することができます。

登録グループ名、ソース場所、ソースにアクセスするためのユーザー ID およびパスワードは編集できます。データベースタイプを変更することはできません。

1. 登録グループを開きます。

以下の表に、ソースタイプに応じてリソースインスペクタの【登録グループ】タブに表示されるプロパティを示します。

プロパティ	説明
名前	登録グループのユーザー定義名です。
アクセス方式	読み取り専用。データソースタイプに関連付けられたアクセス方法。
場所	変更をキャプチャするソースの場所です。場所の値は、PowerExchange Navigator システム上の dbmover.cfg 構成ファイルの NODE 文に指定したノード名です。
ユーザー ID	<p>ソースにアクセスできるユーザー ID です。ユーザー ID のタイプは、次のソースタイプおよび PowerExchange のセキュリティ設定によって異なります。</p> <ul style="list-style-type: none"> - ソースが i5/OS システムまたは z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を(1,x)または(2,x)に設定すると、PowerExchange のセキュリティが有効になります。 - DB2 LUW、Microsoft SQL Server、MySQL、Oracle、または PostgreSQL ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力してください。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。

プロパティ	説明
パスワード	<p>指定したユーザー名のパスワードです。</p> <p>データソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次を示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p>
インスタンス	<p>ソースが Adabas、DB2 for i5/OS、Oracle、または VSAM の場合のインスタンス名は、登録グループを定義したときに入力したコレクション識別子になります。</p> <p>DB2 for z/OS のソースの場合、このインスタンス名がデータベースのインスタンス名になります。</p> <p>Microsoft SQL Server ソースの場合、インスタンス名は登録グループを作成したときに入力した【インスタンス】の値です。インスタンス名を入力しなかった場合、この値はデータベース名の最初の 4 文字と 000 から始まる昇順番号を結合した構成で生成される値になります。</p> <p>MySQL または PostgreSQL ソースの場合、インスタンス名は登録グループを作成したときに入力した【インスタンス】の値です。インスタンス名を入力しなかった場合、この値はデータベースサーバー名の最初の 4 文字と 000 から始まる昇順番号を結合した構成で生成される値になります。</p>
MUF 名	CA Datacom/DB ソースの Multi-User Facility (MUF) 名です。
サーバー	Microsoft SQL Server ソースの場合、パブリケーションサーバーの名前。PostgreSQL ソースの場合、データベースサーバーの名前。
MySQL Server	MySQL ソースの場合、データベースサーバーの名前。
データベース	DB2 LUW、Microsoft SQL Server、または PostgreSQL ソースの場合、データベースの名前。
DB 名	IDMS ログベースソースのデータベース名です。

プロパティ	説明
Recon 識別子	IMS ソースの場合はデータベースの RECON 識別子です。
Oracle SID	読み取り専用。Oracle ソースのデータベース用のシステム ID (SID) です。

2. プロパティを編集するには、**リソースインスペクタ**のプロパティをクリックし、値を入力します。
プロパティを編集した後に、変更したプロパティの隣に赤い四角が表示されます。
ヒント: 変更を適用する前に取り消す場合は、**[リセット]** をクリックします。
3. 変更を適用するには、**[適用]** をクリックします。
登録グループ名を編集する場合は、その変更を確認するように求められます。新しい名前が既存の登録グループ名と一致する場合は、現存グループを上書きするように求められます。
場所をインスタンスに対して有効でない場所に変更すると、PowerExchange ではエラーメッセージが表示されます。
インスタンスを存在しないインスタンスに変更すると、PowerExchange ではエラーメッセージが表示されます。
4. **[はい]** をクリックして変更を確定します。
リソースインスペクタにより、編集したフィールドはすべて更新されます。

関連項目：

- [「登録グループの追加」 \(ページ 169\)](#)

キャプチャ登録のプロパティの表示または編集

変更キャプチャのソースを登録した後は、キャプチャ登録プロパティをいくつか編集することができます。

編集できるプロパティは、次のとおりです。

- **ステータス。** すべてのソースタイプで、キャプチャ登録のステータス設定を変更することができます。PowerExchange Navigator では、非アクティブからアクティブ、アクティブから履歴、および一時停止中から履歴へのステータス移行を許可します。登録ステータスをアクティブから一時停止中に変えたり一時停止中からアクティブに戻すには、PWXUCREG ユーティリティを使用する必要があります。
- **圧縮。** すべてのソースタイプで、キャプチャ登録の **[圧縮]** オプションを、**[なし]** から **[部分]**、または **[部分]** から **[なし]** へと変更できます。キーのある i5/OS と z/OS のデータソースの場合、**[圧縮]** オプションを **[なし]** または **[部分]** から **[フル]** に、または **[フル]** または **[なし]** から **[部分]** に変更することができます。
- **コメント。** すべてのソースタイプで、キャプチャ登録のコメントを追加または編集することができます。

DB2 データソースでは、列を追加や削除することができます。

キャプチャ登録に他の任意の変更を加える必要がある場合、登録を再作成する必要があります。

警告: アクティブなキャプチャ登録を編集すると、CDC にさまざまな影響が出ることがあります。

1. 登録グループを開きます。
2. キャプチャ登録を開きます。

以下の表に、キャプチャ登録について、データソースタイプに応じてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

以下の表に、すべてのデータソースタイプについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
名前	読み取り専用です。キャプチャ登録のユーザー定義名です。
タイプ	読み取り専用です。PowerExchange のキャプチャタイプです。次のオプションがあります。 <ul style="list-style-type: none"> - ログベース。PowerExchange がデータベースログから変更をキャプチャします。 - 同期。PowerExchange は、同期方式で変更をキャプチャします。
ステータス	変更キャプチャへの可用性に関するキャプチャ登録のステータスです。次のオプションがあります。 <ul style="list-style-type: none"> - 非アクティブ。PowerExchange は、ソースの変更キャプチャにこの登録を使用することができません。このオプションが、登録作成時の初期ステータスの設定です。この登録を変更キャプチャでの使用対象にするには、このオプションを手動で【アクティブ】に設定する必要があります。 - アクティブ。PowerExchange はこの登録を使用してソースの変更をキャプチャすることができます。 - 履歴。PowerExchange が変更キャプチャへの登録の使用を永久に停止しています。登録が使用されないようになり、削除するか監査目的で保持する必要がある場合に、このオプションを選択します。ステータスを【履歴】に設定した後は、登録を再度アクティブにすることができません。 - 一時停止中。PowerExchange の変更キャプチャ処理が、この登録されたソースに対して一時的に停止中です。このステータスは、PWXUCREG ユーティリティの SUSPEND_REGISTRATION コマンドで設定され、Adabas、Datacom テーブルベース、IDMS ログベース、または IMS ログベースの CDC ソースに対してのみ使用できます。このステータスは、PowerExchange Navigator から【履歴】に変えることができます。ただし、このステータスをアクティブに戻すには、PWXUCREG ユーティリティで ACTIVATE_REGISTRATION コマンドを使用する必要があります。 詳細については、「ステータスオプション」 (ページ 181)を参照してください。
一時停止時間	読み取り専用です。変更キャプチャ用の登録を一時的に停止するために PWXUCREG ユーティリティの SUSPEND_REGISTRATION コマンドが実行された時刻を示す、現在のシステムのタイムスタンプ（現地時間への調整なし）です。タイムスタンプの形式は YYYY/MM/DD HH:MM:SS.ssssss で、ssssss は秒の小数部分です。このプロパティは、登録がまだ一度も一時停止状態になったことがなければ、空白です。 このプロパティは、PWXUCREG ユーティリティが対応しているソース（Adabas、Datacom テーブルベース、IDMS ログベース、および IMS ログベースの CDC ソース）に限り表示されます。他のソースタイプの場合、このプロパティは表示されません。 PWXUCREG の一時停止時間に関する詳細については、『PowerExchange ユーティリティガイド』を参照してください。

プロパティ	説明
動作時刻	<p>読み取り専用です。変更キャプチャ用の登録を再アクティベートするために PWXUCREG ユーティリティの ACTIVATE_REGISTRATION コマンドが実行された時刻を示す、現在のシステムのタイムスタンプ（現地時間への調整なし）です。タイムスタンプの形式は YYYY/MM/DD HH:MM:SS.ssssss で、ssssss は秒の小数部分です。</p> <p>この登録ステータスが PowerExchange Navigator から初めてアクティブに設定されていた場合、またはこの登録が PWXUCREG ユーティリティで一度もアクティブに設定も再設定もされることがない場合、このフィールドは空白です。このフィールドは、PWXUCREG ユーティリティで SUSPEND_REGISTRATION コマンドと ACTIVATE_REGISTRATION コマンドをこの順序で発行するまでは、空白のままです。</p> <p>このプロパティは、PWXUCREG ユーティリティが対応しているソース（Adabas、Datacom テーブルベース、IDMS ログベース、および IMS ログベースの CDC ソース）に関してのみ表示されます。他のソースタイプの場合、このプロパティは表示されません。PWXUCREG のアクティベーション時間に関する詳細については、『<i>PowerExchange ユーティリティガイド</i>』を参照してください。</p>
圧縮	<p>PowerExchange Condense または PowerExchange ロgger（Linux、UNIX、Windows 用）が使用されているかどうか、および i5/OS または z/OS 上の PowerExchange Condense の処理のタイプです。次のオプションがあります。</p> <ul style="list-style-type: none"> - なし。PowerExchange は、登録したソースへの変更の処理に、PowerExchange Condense または PowerExchange ロgger（Linux、UNIX、Windows 用）を使用しません。 - 部分。PowerExchange が、PowerExchange Condense または PowerExchange ロggerを使用して、登録したソースに対する変更を処理します。PowerExchange Condense が部分圧縮処理を使用します。このオプションは、トランザクションの一貫性が保たれるため、Informatica で推奨しています。 - フル。PowerExchange が、PowerExchange Condense の完全圧縮処理を i5/OS または z/OS のソースに使用しますが、いくつかの制限事項が適用されます。この方法では、トランザクションの一貫性が保持されません。 <p>詳細については、『圧縮オプション』（ページ 182）を参照してください。</p>
Version	読み取り専用です。登録バージョン番号です。
キャプチャタイプ	読み取り専用です。データソースタイプ。
コメント	キャプチャ登録に関するユーザー定義コメントです。
Tag	読み取り専用です。登録されたソースに割り当てられ、CDC に意味のある変更レコードの特定に使用される PowerExchange 生成のタグ名です。

以下の表に、Adabas データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
データベース ID	データベースの DBID です。
ファイル番号	データベースのファイル番号です。

以下の表に、CA Datacom/DB データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
データベース ID	データベース ID。
短いテーブル名	テーブルの短い名前です。
テーブル ID	内部テーブル ID。
レコードサイズ	レコードサイズです。
テーブルリカバリ	CA DATACOM/DB のリカバリ機能がこのテーブルに使用されているかを示すオプションです。有効な値は「yes」か「no」です。
エレメント数	エレメントの数です。

以下の表に、i5/OS 用 DB2 データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
ファイル名	ファイルの名前です。

以下の表に、IDMS ログベースデータソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
領域名	データベースレコードを格納する一連のページがある IDMS 領域の名前です。
サブスキーマのタイプ	サブスキーマの名前です。サブスキーマは、データベースのプログラムビューを提供し、実行時に使用されます。これには、データエレメント、レコードタイプ、セットタイプ、領域、論理レコードコンポーネントを含みます。 有効な文字は ASCII 文字、ポンドおよび円記号です。
DB 名	IDMS データベースの名前です。 有効な文字は ASCII 文字、ポンドおよび円記号です。
ベースレコード	IDMS ベースレコードの名前です。
場所モード	レコードを領域の特定のページに割り当てる方法を決める次のモードのうちの 1 つです。 - C。CALC モードを示します。 - V。VIA モードを示します。 - D。DIRECT モードを示します。
ベースレコード ID	IDMS ベースレコードの ID です。
圧縮	レコードの圧縮に IDMS 固有の圧縮方式を使用したかどうかを示します。有効な値は「yes」か「no」です。

プロパティ	説明
Presspack 圧縮	レコードの圧縮に IDMS Presspack 圧縮ツールを使用したかどうかを示します。有効な値は「yes」か「no」です。
Presspack データ特性テーブル	Presspack 圧縮を使用する場合、データに関する情報を格納し圧縮と解凍に必須となるデータ特性テーブル (DCT) の名前。
変数	レコードオカレンスが可変長かどうかを示します。有効な値は「yes」か「no」です。
ページグループ	ページグループ番号です。
基数	基数番号です。基数がデータベースキーの形式です。これにより、データベース内のレコードを特定するデータベースキーの構造が決まります。
最小ルート長	ホームページの可変長レコードの最小ルート部分の長さです。
データ長	レコードオカレンスの Data コンポーネントの長さです。Data コンポーネントは、レコード値を表現する一連のバイトです。
プレフィクス長	レコードオカレンスの Prefix コンポーネントの長さです。プレフィックスは、関連する NEXT、PRIOR、および OWNER レコードを指します。
制御部分長	レコードの制御部分 (シンボリックキーを含むそれ以下の部分) の長さです。

以下の表に、IMS 同期データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
データベース名	読み取り専用。データベース名。
プライマリデータセット名	読み取り専用。データセットファイル名です。

以下の表に、Microsoft SQL Server データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
サーバー	ソースが存在するサーバーです。
データベース名	データベース名。
分散 DB 名	SQL Server の分散データベース名です。
アークティクル ID	SQL Server アークティクル ID です。

以下の表に、Oracle データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
補足ログ	読み取り専用。補足ロググループ名です。

以下の表に、VSAM データソースについてリソースインスペクタの【キャプチャ登録】タブに表示されるプロパティを示します。

プロパティ	説明
データセット名	VSAM ESDS、KSDS、または RRDS データセット名です。

- リソースインスペクタで編集対象のプロパティをクリックしてから、プロパティの値を変更します。
プロパティを編集した後に、その隣に赤い四角が表示されます。
ヒント: 変更を元の状態に戻すには、【リセット】をクリックします。
- 変更を適用するには、【適用】をクリックします。

関連項目：

- 「[キャプチャ登録の追加](#)」 (ページ 175)
- 「[キャプチャ登録での DB2 カラムの変更](#)」 (ページ 192)
- 「[登録タグ名](#)」 (ページ 165)
- 「[ステータスオプション](#)」 (ページ 181)
- 「[圧縮オプション](#)」 (ページ 182)

Datacom キャプチャ登録のエレメントの表示

変更キャプチャ対象として登録された Datacom テーブルのエレメントを表示することができます。

- 登録グループを開き、キャプチャ登録を開きます。
- リソースエクスプローラの【登録グループ】タブでキャプチャ登録を右クリックし、【エレメントの表示】をクリックします。

キャプチャ登録 - エレメント詳細ウィンドウには、登録されたテーブルのエレメントに関する以下の情報が表示されます。

- 名前。** エレメントの名前です。
- 位置。** レコードの開始からのエレメントの位置オフセットです。
- 長さ。** 要素の長さです。

IDMS キャプチャ登録のパスの表示

変更キャプチャ対象として登録された IDMS テーブルのパスまたは IDMS ベースレコードの親階層を表示することができます。

IDMS キャプチャ登録のパスを表示する手順

1. 登録グループを開き、キャプチャ登録を開きます。
2. リソースエクスプローラの **【登録グループ】** タブでキャプチャ登録を右クリックし、**【パスの表示】** をクリックします。

パスの表示ウィンドウには、そのベースレコードの親階層が表示されます。

キャプチャ登録での DB2 カラムの変更

DB2 データソースでは、キャプチャ登録への列の追加およびキャプチャ登録からの列の削除を行うことができます。

注: アクティブなキャプチャ登録の変更はキャプチャプロセスに影響を与えます。詳細については、Informatica グローバルカスタマサポートにお問い合わせください。

1. 登録グループを開きます。
2. リソースエクスプローラの **【登録グループ】** タブでキャプチャ登録を右クリックし、**【カラムの変更】** をクリックします。
3. **【キャプチャ登録 - カラム】** ダイアログボックスが表示されます。
4. **【すべて選択し、変更を通知する】** オプションを設定し、全てのカラムを選択してスキーマのカラム変更を通知するかどうかを指定します。
 - カラム変更を通知しない場合はオプションをクリアします。
 - カラム変更を通知する場合はオプションを設定します。
5. カラムを追加するには、**【利用可能なカラム】** リストの列を選択し、**【追加】** をクリックします。
6. カラムを削除するには、**【現在のカラム】** リストの列を選択し、**【削除】** をクリックします。
7. **【OK】** をクリックします。

キャプチャ登録の削除

使用する必要がなくなったキャプチャ登録を削除することができます。一度に 1 つの登録を削除する必要があります。

登録を削除すると、PowerExchange は、データソースが存在するシステム上にある CCT ファイルから登録を削除します。削除された登録に関連付けられた抽出マップは削除されません。ただし、ユーザーは抽出マップを手動で削除できます。

登録が削除された後、同じソースオブジェクトの変更を再度キャプチャする場合、登録を再作成する必要があります。登録を再作成すると、抽出マップが再生成されます。

1. キャプチャ登録が開いている場合は、そのキャプチャ登録を閉じます。

2. リソースエクスプローラの【登録グループ】タブでキャプチャ登録を選択し、ツールバーの【削除】ボタンをクリックします。
削除の確認を求めるメッセージボックスが表示されます。
3. 削除を確認するには、【はい】をクリックします。
4. 登録がアクティブな場合は、2つ目のメッセージボックスが表示され、登録のステータスを【履歴】に設定するのではなく登録を削除することを確認するよう求められます。【はい】をクリックして削除を確認するか、【いいえ】をクリックして削除をキャンセルします。

注: 【はい】をクリックして Microsoft SQL Server の登録を削除する場合、PowerExchange は、パブリケーションデータベースから、関連付けられた SQL Server アーティクルの削除を試みます。PowerExchange が何らかの理由で SQL Server アーティクルを削除できない場合、エラーが表示され、メッセージボックスが表示されて、登録の削除の再確認が求められます。【はい】をクリックして削除を確認した場合、関連付けられた SQL Server アーティクルを手動で削除する必要があります。

登録が削除されると、PowerExchange は、削除された登録で定義されていたソースオブジェクトの変更データをキャプチャしなくなります。

関連項目：

- [「キャプチャ登録のプロパティの表示または編集」 \(ページ 186\)](#)

登録グループの削除

登録グループを削除する場合は、PowerExchange により、PowerExchange Navigator システムのローカルグループ情報が削除されます。

PowerExchange では、そのグループのキャプチャ登録は削除されません。削除したグループとして同じプロパティで別の登録グループを作成する場合、新しいグループには、削除されたグループ用に存在するすべてのキャプチャ登録が含まれます。

また、PowerExchange では、登録グループの作成時に PowerExchange により生成された抽出グループおよびアプリケーショングループは削除されません。

登録グループを削除する方法

1. 登録グループが開いている場合は、その登録グループを閉じます。
2. リソースウィンドウで、1つ以上の登録グループを削除します。
3. グループの削除を確認するように求めるメッセージボックスが表示されたら、【はい】をクリックします。
PowerExchange により、単数または複数の登録グループが削除されます。

関連項目：

- [「登録グループのすべてのキャプチャ登録の削除」](#) (ページ 194)

登録グループのすべてのキャプチャ登録の削除

登録グループとそのキャプチャ登録を完全に削除するには、キャプチャ登録、関連する抽出マップと抽出グループ、および登録グループを削除します。

削除されたキャプチャ登録に関連付けられた抽出マップを開いたり利用したりすることはできません。代わりに、抽出マップを編集して、抽出マップに関連付けられたキャプチャ登録を変更することはできます。

1. 登録グループの各キャプチャ登録を削除します。
2. 削除した登録と関連付けられている抽出マップを削除して、抽出グループを削除します。
3. 登録グループを削除します。

関連項目：

- [「登録グループの削除」](#) (ページ 193)
- [「キャプチャ登録の削除」](#) (ページ 192)
- [「抽出マップと関連付けられたキャプチャ登録の変更」](#) (ページ 205)
- [「抽出マップの削除」](#) (ページ 211)
- [「抽出グループの削除」](#) (ページ 212)

第 6 章

抽出グループおよび抽出マップ

この章では、以下の項目について説明します。

- [抽出グループおよび抽出マップの概要, 195 ページ](#)
- [抽出マップ名, 196 ページ](#)
- [抽出グループの追加, 198 ページ](#)
- [抽出マップの追加, 200 ページ](#)
- [抽出マップの表示, 202 ページ](#)
- [データ列用の変更インジケータおよび操作前の画像の追加および削除, 203 ページ](#)
- [抽出処理への列の追加または抽出処理からの列の削除, 204 ページ](#)
- [抽出マップと関連付けられたキャプチャ登録の変更, 205 ページ](#)
- [データマップと抽出マップとの統合, 206 ページ](#)
- [抽出グループのプロパティの表示または編集, 207 ページ](#)
- [抽出マップのプロパティの表示, 209 ページ](#)
- [抽出マップからのデータマップの作成, 210 ページ](#)
- [抽出マップの削除, 211 ページ](#)
- [抽出グループの削除, 212 ページ](#)
- [抽出グループ内のすべての抽出マップの削除, 212 ページ](#)
- [抽出定義ウィンドウ - マップタブ, 213 ページ](#)
- [抽出定義ウィンドウ - 拡張タブ, 213 ページ](#)
- [PowerExchange で生成された抽出マップ内のカラム, 214 ページ](#)

抽出グループおよび抽出マップの概要

抽出マップを使用して、PowerExchange でキャプチャされた変更データを抽出します。

抽出グループと抽出マップは、登録済みデータソースの抽出情報を定義し、CDC 処理に必要です。

抽出グループは、データベースインスタンス、ノード場所、およびデータソースタイプを定義します。登録グループを作成すると、PowerExchange では、同じ名前の抽出グループが作成されます。抽出グループを手動で追加することもできます。

抽出グループには、1 つまたはそれ以上の抽出マップが含まれています。抽出マップはデータソースをマッピングします。キャプチャ登録を作成すると、PowerExchange では、キャプチャ登録内と同じ列および登録タグ名を含む抽出マップが作成されます。PowerExchange では一意の名前を抽出マップに割り当てます。

キャプチャ登録に対して、追加の抽出マップを手動で追加することもできます。ユーザー定義の抽出マップには、それに関連付けられているキャプチャ登録内と同じ列および登録タグ名が含まれます。PowerExchange では、ユーザー定義の抽出マップに、抽出マップの追加時に指定したマップ名を含む一意の名前を割り当てます。

z/OS システム上の DTLCAMAP ファイルまたは非 z/OS システム上の CAMAPS ファイルには、PowerExchange Navigator で定義されている抽出マップ情報が含まれます。抽出グループは、PowerExchange Navigator で表示される DTLCAMAP または CAMAPS ファイル内のデータをフィルタリングします。

抽出マップを編集してデータ列の変更インジケータ (CI) および操作前の画像 (BI) の列を追加し、抽出処理の対象となる列を追加または削除することができます。

DB2 ソースでは、データマップを抽出マップと統合できます。抽出マップは、関連付けられたデータマップを使用して変更データをマッピングします。

抽出マップをテストするには、データベース行のテストを実行します。

関連項目：

- [「抽出マップのテスト」 \(ページ 232\)](#)
- [「登録タグ名」 \(ページ 165\)](#)
- [「登録グループ」 \(ページ 169\)](#)

抽出マップ名

抽出マップ名には、ソースタイプに応じて異なる特定の形式があります。

以下の一般的な形式を始めに使用します。

xnninstance.mapname

抽出マップを使用して変更を抽出したら、テーブル名が抽出マップ名に付加されます。

xnninstance.mapname_tablename

抽出マップ名の変数は以下のとおりです。

x

抽出マップが生成されたものかユーザー定義かを示す文字。有効な値は以下のとおりです。

- **d**。PowerExchange で生成されたデフォルトの抽出マップ。
- **u**。ユーザー定義の抽出マップ。

nn

ソースタイプを定義する 1 桁または 2 桁の値。有効な値は以下のとおりです。

- **1** (DB2 for z/OS の場合)。
- **2** (IMS の場合)。
- **3** (DB2 for i5/OS の場合)。
- **4** (Adabas の場合)。
- **6** (VSAM の場合)。
- **7** (CA Datacom/DB の場合)。

- 8 (Oracle の場合)。
- 10 (Microsoft SQL Server の場合)。
- B (DB2 for Linux, UNIX, Windows の場合)。
- D (IDMS ログベースの場合)。
- e (MySQL の場合)。
- f (PostgreSQL の場合)。

instance

登録グループが作成されたときに指定または生成されたインスタンスの名前です。インスタンス値はソースタイプによって異なります。

mapname

以下のいずれかの値になります。

- デフォルトの抽出マップの場合、キャプチャ登録のために指定されたユーザー定義名です。
- ユーザー定義の抽出マップの場合、抽出マップが手動で作成されたときに指定された **【マップ名】** の値です。

tablename

リレーショナルソーステーブルのテーブル名、またはソースデータマップからのテーブル名です。

抽出マップでは、*xnninstance* 変数文字列がスキーマと呼ばれます。抽出グループを開くと、PowerExchange Navigator には、グループ内の抽出マップが *schema.mapname* として表示されます。抽出マップを使用して変更を抽出するときには、*schema.mapname_tablename* の完全抽出マップ名が使用されます。

例えば、データベース行のテストの実行時に、PowerExchange Navigator では変更を抽出する以下の SQL 文を生成します。

```
SELECT * FROM schema.mapname_tablename
```

データベース行のテストでは、すべての列はデフォルトで選択されます。ただし、次のいずれかの方法で、SQL 文を編集して特定の列を選択することができます。

- **【データベース行のテスト】** ダイアログボックスで、**【SQL 文】** ボックスの SQL 文を手動で編集し、特定の列を選択します。
- **抽出定義** ウィンドウで特定の列を選択し、データベース行のテストを実行します。

PowerExchange により、指定された列を選択する SQL 文が **【データベース行のテスト】** ダイアログボックスに生成されます。

注: PowerExchange Client for PowerCenter (PWXPC) の **【スキーマ名オーバーライド】** および **【マップ名オーバーライド】** フィールドを使用すると、スキーマ名とマップ名をオーバーライドできます。

関連項目：

- [「登録グループの追加」 \(ページ 169\)](#)
- [「キャプチャ登録の追加」 \(ページ 175\)](#)
- [「抽出マップの追加」 \(ページ 200\)](#)

抽出グループの追加

抽出グループは、抽出マップの名前が付いたグループです。

抽出グループは、データソースインスタンス、ノード場所、およびデータソースタイプを定義します。登録グループを追加する場合は、PowerExchange により、抽出グループが同じ名前で追加されます。次の手順で説明しているように、抽出グループを手動で追加することもできます。グループの定義時またはそれ以降に、抽出マップを抽出グループに追加します。

1. リソースエクスプローラの [リソース] タブで、[追加] - [抽出グループ] をクリックします。
2. [抽出グループの追加] ダイアログボックスに、以下の情報を入力します。

名前

抽出グループのユーザー定義名です。最大長は 16 文字です。

場所

変更データを抽出するソースの場所です。ソースが PowerExchange Navigator システムにある場合は、[ローカル] を選択します。

このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。

デフォルトは [ローカル] です。

タイプ

データソースのタイプです。

データソースタイプを選択し、次のように関連情報を入力します。関連情報は、対応する登録グループの情報と一致する必要があります。

- **ADABAS**。Adabas ファイル。

[コレクション識別子] ボックスに、インスタンス名を入力します。

- **AS4**。i5/OS 用 DB2 テーブル。

[コレクション識別子] ボックスに、ソースシステムの *dtllib*/CFG ファイルの DBMOVER メンバで、AS4J CAPI_CONNECTION 文の INST パラメータに指定されたインスタンス名を入力します。

- **DATACOM**。CA Datacom/DB テーブルベース CDC に対して登録されている CA Datacom/DB テーブルです。

[MUF 名] ボックスに、ソースの Multi-User Facility (MUF) 名を入力します。

- **DB2**。DB2 for z/OS テーブル。

[データベースインスタンス] ボックスに、DB2 サブシステム ID、または DB2 データ共有グループ名を入力します。

- **DB2UDB**。DB2 for Linux, UNIX, and Windows テーブル。

[データベース] ボックスに、データベース名を入力します。

- **IDMS_L**。IDMS ログベース CDC 対象として登録された IDMS テーブル。

[登録場所] リストで、キャプチャ登録の格納場所を選択します。

[ソースマップの場所] リストで、IDMS テーブル用のデータマップのある場所を選択します。

[Logsid] ボックスに、ソースシステムの DBMOVER 構成ファイルの LOGSID パラメータで指定された値を入力します。

- **IMS**。IMS データベース。

IMS 同期 CDC の場合は、**[Recon 識別子]** ボックスに IMS サブシステム ID を入力します。

IMS ログ-ベースの CDC では、**[Recon 識別子]** ボックスに、RECON データセットのある z/OS システムの DBMOVER コンフィギュレーションメンバにある IMSID パラメータで指定された値を入力します。

- **MSSql**。Microsoft SQL Server テーブル。

[データベースサーバー] リストで、ソースが存在するサーバを選択します。

[データベース名] リストで、データベース名を選択または入力します。

- **ORACLE**。Oracle テーブル。

[コレクション識別子] ボックスに、ソースシステムの DBMOVER 構成ファイルの ORACLEID 文で指定された Oracle インスタンス ID を入力します。

- **VSAM**。VSAM ESDS、KSDS、RRDS、および VRRDS データセット。

[コレクション識別子] ボックスに、インスタンス名を入力します。

ユーザー ID

ソースにアクセスできるユーザー ID を入力します。ユーザー ID のタイプは、次のソースタイプおよび PowerExchange のセキュリティ設定によって異なります。

- ソースが i5/OS システムまたは z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を(1,x)または(2,x)に設定すると、PowerExchange のセキュリティが有効になります。
- DB2 for Linux, UNIX, and Windows、Microsoft SQL Server、または Oracle ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力します。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。
- サポートされている Linux、UNIX、または Windows システムのソースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

パスワード

指定したユーザー名のパスワードです。

ソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の長さで指定できます。パスフレーズには次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。

z/OS で IBM IRRPHREX イグジットに指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

ユーザー ID を指定してパスワードを省略した場合、抽出グループを開いたときに PowerExchange Navigator は【パスワードが入力されていません】ダイアログボックスにパスワードの入力を求めるメッセージを表示します。

抽出定義の追加

抽出マップをグループに追加します。

このオプションの選択を取り消すと、抽出グループのみが追加されます。

デフォルトは選択されています。

3. 以下のいずれかのアクションを実行します。

- **【抽出定義の追加】** オプションを選択した場合、**【次へ】** をクリックし、抽出マップの追加手順の [「抽出マップの追加」 \(ページ 200\)](#) へと進みます。
- **【抽出定義の追加】** オプションを選択しない場合、**【完了】** をクリックします。リソースインスペクタの **【抽出グループ】** タブで、抽出グループ用のプロパティが表示されます。さらに、ソースシステムの PowerExchange リスナが稼働している場合、リソースエクスプローラの **【抽出グループ】** タブで、抽出グループに関連付けられたソースインスタンス用に存在するすべての抽出マップが表示されます。

関連項目：

- [「登録グループ」 \(ページ 169\)](#)
- [「抽出マップの追加」 \(ページ 200\)](#)

抽出マップの追加

必要に応じて、抽出マップを抽出グループに手動で追加できます。

キャプチャ登録を定義すると、PowerExchange が抽出マップを 1 つ生成します。ただし、1 つの登録に対しもう 1 つ抽出マップを追加したほうがよい場合もあります。

1. 抽出グループを開きます。
2. リソースエクスプローラの **【抽出グループ】** タブで、抽出グループを右クリックし、**【抽出定義の追加】** をクリックします。
3. **【抽出定義の追加 - 名前】** ダイアログボックスに、以下の情報を入力します。
 - a. **【マップ名】** フィールドで、その抽出マップの一意のユーザー定義名を入力します。最大長は 8 文字です。
 - b. **【テーブル名】** フィールドに、抽出マップ名のテーブルの部分に入るユーザー定義名を入力します。最大長は 20 文字です。

【スキーマ名】 フィールドには、抽出マップ名のスキーマ部分に対する生成された名前が読み取り専用で表示されます。名前のフォーマットは以下のとおりです。

`unninstance`

説明:

- u は、抽出マップがユーザー定義であることを示します。

- *nn* は、ソースタイプを示す 1 桁または 2 桁の値です。例えば、10 は Microsoft SQL Server を表します。
- *instance* は登録グループが作成されたときに指定または生成されたインスタンスの値です。値はソースタイプによって異なります。

4. **【次へ】** をクリックします。

【抽出定義の追加 - キャプチャ登録】 ダイアログボックスが表示されます。**【利用可能なキャプチャ登録】** リストには、抽出マップとの関連付けに使用できるキャプチャ登録が表示されます。

以下の表に、このリストのカラムを示します。

カラム	説明
名前	キャプチャ登録のユーザー定義名です。
バージョン	登録バージョン番号です。
圧縮	<p>PowerExchange Condense または PowerExchange ロgger (Linux、UNIX、Windows 用) が使用されているかどうか、および i5/OS または z/OS 上の PowerExchange Condense の処理のタイプです。次のオプションがあります。</p> <ul style="list-style-type: none"> - なし。PowerExchange は、登録したソースへの変更の処理に、PowerExchange Condense または PowerExchange ロgger (Linux、UNIX、Windows 用) を使用しません。 - 部分。PowerExchange が、PowerExchange Condense または PowerExchange ロgger を使用して、登録したソースに対する変更を処理します。PowerExchange Condense が部分圧縮処理を使用します。このオプションは、トランザクションの一貫性が保たれます。 - フル。PowerExchange が、PowerExchange Condense の完全圧縮処理を i5/OS または z/OS のソースに使用しますが、いくつかの制限事項が適用されます。この方法では、トランザクションの一貫性が保持されません。 <p>詳細については、「圧縮オプション」 (ページ 182) を参照してください。</p>
ステータス	<p>変更キャプチャへの可用性に関するキャプチャ登録のステータスです。次のオプションがあります。</p> <ul style="list-style-type: none"> - I (非アクティブ)。PowerExchange は、ソースの変更キャプチャにこの登録を使用することができません。このオプションが、新規の登録での初期ステータスの設定です。 - A (アクティブ)。PowerExchange はこの登録を使用してソースの変更をキャプチャすることができます。 - H (履歴)。PowerExchange が変更キャプチャへの登録の使用を永久に停止しています。この登録は使用されなくなりましたが、監査の目的で保持しています。 - S (一時停止中)。PowerExchange の変更キャプチャ処理が、この登録されたソースに対して一時的に停止中です。このステータスは、PWXUCREG ユーティリティの SUSPEND_REGISTRATION コマンドで設定され、Adabas、Datacom テーブルベース、IDMS ログベース、または IMS ログベースの CDC ソースに対してのみ使用できます。 <p>詳細については、「ステータスオプション」 (ページ 181) を参照してください。</p>

5. **【利用可能なキャプチャ登録】** リストでキャプチャ登録を選択し、**【追加】** をクリックします。

選択されたキャプチャ登録が **【現在のキャプチャ登録】** リストに移動します。

ヒント:

- 選択を変更するには、**【利用可能なキャプチャ登録】** リスト内の登録をダブルクリックします。登録が **【現在のキャプチャ登録】** リストに移動し、前の登録が **【利用可能なキャプチャ登録】** リストに戻ります。
- **【現在のキャプチャ登録】** リストからキャプチャ登録を削除するには、登録を選択し、**【削除】** をクリックするか登録をダブルクリックします。登録が **【使用可能なキャプチャ登録】** リストに戻ります。

6. **【完了】** をクリックします。

PowerExchange は、DTLCAMAP ファイル（ソースシステムが z/OS の場合）に、または CAMAPS ファイル（ソースシステムが z/OS 以外の場合）に、抽出マップを作成します。抽出マップ名の形式は次のとおりです。

`unnschema.mapname`

説明:

- 「u」は、抽出マップがユーザー定義であることを示します。
- *nn* は、ソースタイプを示す 1 桁または 2 桁の値です。
- *schema* は PowerExchange で生成されたスキーマ名です。
- *mapname* には、手順 3 で指定したマップ名が入ります。

詳細については、「[抽出マップ名](#)」 ([ページ 196](#)) を参照してください。

新しい抽出マップは、**リソースエクスプローラ** 中の **【抽出グループ】** タブの抽出マップリストに表示されます。また、抽出マップは次に示すインターフェイスにも表示されます。

- **抽出定義** ウィンドウには、抽出マップの列が表示されます。
- **リソースインスペクタ** の **【抽出定義】** タブで、抽出マップ用のプロパティが表示されます。この矢印を使用して、**リソースインスペクタ** の **【抽出定義】** タブと **【抽出グループ】** タブを切り替えます。
- ステータスバーには、オープン抽出グループと抽出マップに関する情報が表示されます。

関連項目：

- 「[抽出マップ名](#)」 ([ページ 196](#))
- 「[登録グループの追加](#)」 ([ページ 169](#))
- 「[ステータスオプション](#)」 ([ページ 181](#))
- 「[圧縮オプション](#)」 ([ページ 182](#))

抽出マップの表示

PowerExchange によって生成されたカラムを含む、抽出マップのカラムを表示できます。

1. 抽出グループを開きます。
2. 抽出マップを開きます。

抽出定義 ウィンドウが表示され、次のタブにカラムの情報が表示されます。

- **【マップ】** タブには、PowerExchange によって変更がキャプチャされたカラムのメタデータと、PowerExchange によって生成されたカラムを表示するよう選択した場合は生成された DTL__カラムのメタデータが表示されます。
- **【拡張】** タブには、PowerExchange によって生成された DTL__CI および DTL__BI カラムを 1 つ以上の抽出マップのカラムに追加している場合はその DTL__CI および DTL__BI カラムが表示されます。

PowerExchange で生成された列を表示するには、**【マップ】** タブ上の任意の場所を右クリックし、**【自動生成されたカラムの表示】** をクリックします。

PowerExchange で生成された列を非表示にするには、**【マップ】** タブ上の任意の場所を右クリックし、**【自動生成されたカラムの非表示】** をクリックします。

DTL__CI および DTL__BI カラムを 1 つ以上のデータカラムに対して追加するには、[マップ] タブ上の任意の場所を右クリックし、[変更インジケータ/前のイメージの拡張の変更] をクリックします。これらのカラムが両方のタブに表示されます。

いずれかのタブのカラムのリストをカラム名またはメタデータフィールドでソートするには、メタデータカラムの見出しをクリックします。

関連項目：

- [「抽出定義ウィンドウ - マップタブ」 \(ページ 213\)](#)
- [「抽出定義ウィンドウ - 拡張タブ」 \(ページ 213\)](#)

データ列用の変更インジケータおよび操作前の画像の追加および削除

ソーステーブルの抽出マップを編集して、1 つ以上のデータカラムに対して変更インジケータ (CI) および操作前の画像 (BI) のカラムを追加または削除することができます。

抽出マップでは、CI のカラム名は DTL__CI_data_column_name で、BI のカラム名は DTL__BI_data_column_name です。

DTL__CI カラムは関連付けられているカラムのデータが UPDATE 操作で変更されたかどうかを示します。CDC セッションの SQL WHERE 句フィルタで CI カラムを使用すると、PWXPC から PowerCenter に渡されて処理される変更レコードをフィルタリングできます。DTL__CI カラムには次の値のいずれかが含まれます。

- **Y**。UPDATE でカラムのデータが変更されました。
- **N**。UPDATE でカラムのデータが変更されませんでした。
- **Null**。カラムに対して INSERT 操作または DELETE 操作が行われました。

DTL__BI カラムには関連付けられているカラムの UPDATE 操作の BI が含まれます。ソースデータベースでプライマリキーへの変更が許可されている場合は、BI フィールドと PowerCenter の Flexible Target Key カスタムトランスフォーメーションを使用して、ソースのキーの変更時にターゲットでプライマリキーへの更新を起動できます。

ヒント: PowerExchange ロgger (Linux、UNIX、Windows 用) を使用し、pwxccf.cfg ファイルの CAPT_IMAGE パラメータに AI を指定する場合は、変更インジケータおよび操作前のイメージのカラムは追加しないでください。AI 設定によって、PowerExchange ロgger は操作後のイメージのみ格納します。したがって、PowerExchange では、PowerCenter で下流処理される BI を PWXPC に提供できません。また、CAPT_IMAGE パラメータに BA を入力すると、PowerExchange ロgger によって UPDATE に対する操作前の画像と操作後のイメージの両方が格納されます。ソースで INSERT または DELETE 操作を行うと、変更インジケータカラムに Null 値が入力されます。

1. 抽出グループを開きます。
2. 抽出マップを右クリックし、[変更インジケータ/前のイメージの拡張の変更] をクリックします。
[抽出定義 - カラム変更] ダイアログボックスが表示されます。
3. [変更インジケータ] タブで、DTL__CI カラムに追加するカラムを選択します。[利用可能なカラム] リストで 1 つ以上の列を選択し、[追加] をクリックします。
選択したカラムが [選択したカラム] リストに表示されます。
4. [操作後のイメージ] タブで、DTL__BI カラムに追加するカラムを選択します。[利用可能なカラム] リストで 1 つ以上の列を選択し、[追加] をクリックします。

選択したカラムが**【選択したカラム】** リストに表示されます。

注: 選択したカラムが PowerExchange 関数を起動する式のユーザー定義フィールドであった場合、その関数が BI バッファをサポートしていることを確認してください。関数の詳細については、[付録 A, 「ユーザー定義フィールド用 PowerExchange 関数」 \(ページ 303\)](#)を参照してください。

5. DTL__CI または DTL__BI カラムを削除するには、**【選択したカラム】** リストでカラムを選択し、**【削除】** をクリックします。

選択したカラムが**【利用可能なカラム】** リストに戻ります。

6. **【OK】** をクリックします。

DTL__CI または DTL__BI カラムを追加した場合、カラムは**抽出定義**ウィンドウの**【マップ】**と**【拡張】**の両方のタブに表示されます。

7. 変更を検証するために、抽出マップでデータベース行のテストを実行します。

関連項目：

- [「抽出マップのテスト」 \(ページ 232\)](#)

抽出処理への列の追加または抽出処理からの列の削除

抽出マップを編集して、抽出処理の対象となる列を追加または削除することができます。

抽出処理の対象となる列を追加または削除しても、関連付けられているキャプチャ登録内の列 (PowerExchange で変更データをキャプチャする列) は影響を受けません。

DTL__CI および DTL__BI カラムを抽出マップのカラムと関連付けることもできます。変更インジケータ (DTL__CI) カラムは変更レコードでカラムの内容が変更されたかどうかを示します。操作前のイメージ (DTL__BI) カラムには最後の UPDATE 操作前のカラムの内容が含まれます。

1. 抽出グループを開きます。
2. 抽出マップを開きます。

抽出定義ウィンドウでは次のタブにカラムのメタデータが表示されます。

- **【マップ】** タブには、PowerExchange が変更データをキャプチャしたカラムとそれらのカラムに追加されたすべての DTL__CI と DTL__BI のカラムのメタデータが表示されます。このタブには、表示すると選択した場合、PowerExchange で生成されたカラムのメタデータも表示されます。

PowerExchange で生成されたカラムを表示するには、**【マップ】** タブ上の任意の場所を右クリックし、**【自動生成されたカラムの表示】** をクリックします。

- CI カラムと BI カラムに 1 つ以上の抽出マップカラムがある場合、**【拡張】** タブには CI カラムと BI カラムが表示されます。

3. 列を抽出処理から削除するには、列名の横のチェックボックスの選択を外します。

PowerExchange では、削除された列の変更データの抽出は停止しますが、選択された列のデータの抽出は続行します。

4. 列を抽出処理に追加するには、列名の横のチェックボックスを選択します。

5. カラムの追加または削除後に、そのカラムに対してデータベース行テストを実行して、抽出マップが引き続き有効であることを確認してください。

関連項目：

- [「データ列用の変更インジケータおよび操作前の画像の追加および削除」 \(ページ 203\)](#)
- [「抽出マップのテスト」 \(ページ 232\)](#)
- [「抽出定義ウィンドウ - マップタブ」 \(ページ 213\)](#)
- [「抽出定義ウィンドウ - 拡張タブ」 \(ページ 213\)](#)

抽出マップと関連付けられたキャプチャ登録の変更

抽出マップを編集して、それに関連付けられているキャプチャ登録を変更することができます。

新しいバージョンのキャプチャ登録が追加された場合、抽出マップに関連付けられているキャプチャ登録を変更することも可能です。

1. 抽出グループを開きます。
2. 抽出マップを開きます。
3. 抽出マップを右クリックし、**【キャプチャ登録の変更】** をクリックします。

【抽出定義 - キャプチャ登録の変更】 ダイアログボックスが表示されます。

【キャプチャ登録の変更】 タブでは、抽出マップに現在関連付けられているキャプチャ登録が**【現在のキャプチャ登録】** リストに表示され、選択できる登録が**【使用可能なキャプチャ登録】** リストに表示されます。

以下の表に、各リストのカラムを示します。

カラム	説明
名前	キャプチャ登録のユーザー定義名です。
バージョン	登録バージョン番号です。

カラム	説明
圧縮	<p>PowerExchange Condense または PowerExchange ロgger (Linux、UNIX、Windows 用) が使用されているかどうか、および i5/OS または z/OS 上の PowerExchange Condense の処理のタイプです。次のオプションがあります。</p> <ul style="list-style-type: none"> - なし。PowerExchange は、登録したソースへの変更の処理に、PowerExchange Condense または PowerExchange ロgger (Linux、UNIX、Windows 用) を使用しません。 - 部分。PowerExchange が、PowerExchange Condense または PowerExchange ロgger を使用して、登録したソースに対する変更を処理します。PowerExchange Condense が部分圧縮処理を使用します。このオプションは、トランザクションの一貫性が保たれます。 - フル。PowerExchange が、PowerExchange Condense の完全圧縮処理を i5/OS または z/OS のソースに使用しますが、いくつかの制限事項が適用されます。この方法では、トランザクションの一貫性が保持されません。 <p>詳細については、「圧縮オプション」 (ページ 182)を参照してください。</p>
ステータス	<p>変更キャプチャへの可用性に関するキャプチャ登録のステータスです。次のオプションがあります。</p> <ul style="list-style-type: none"> - 非アクティブ。PowerExchange は、ソースの変更キャプチャにこの登録を使用することができません。このオプションが、新規の登録での初期ステータスの設定です。 - アクティブ。PowerExchange はこの登録を使用してソースの変更をキャプチャすることができます。 - 履歴。PowerExchange が変更キャプチャへの登録の使用を永久に停止しています。この登録は使用されなくなりましたが、監査の目的で保持しています。 - 一時停止中。PowerExchange の変更キャプチャ処理が、この登録されたソースに対して一時的に停止中です。このステータスは、PWXUCREG ユーティリティの SUSPEND_REGISTRATION コマンドで設定され、Adabas、Datacom テーブルベース、IDMS ログベース、または IMS ログベースの CDC ソースに対してのみ使用できます。 <p>詳細については、「ステータスオプション」 (ページ 181)を参照してください。</p>

4. **【現在のキャプチャ登録】** リストで、現在の登録を選択し、**【削除】** をクリックします。
5. **【使用可能なキャプチャ登録】** リストで、抽出マップに関連付ける登録を選択し、**【追加】** をクリックします。
そのキャプチャ登録が **【現在のキャプチャ登録】** リストに表示されます。
6. **【OK】** をクリックします。
7. 変更を確認するには、抽出マップでデータベース行のテストを実行します。

関連項目：

- [「抽出マップのテスト」 \(ページ 232\)](#)
- [「ステータスオプション」 \(ページ 181\)](#)
- [「圧縮オプション」 \(ページ 182\)](#)

データマップと抽出マップとの統合

DB2 ソースでは、データマップを抽出マップと統合できます。

例えば、ソーステーブルの列に、列のデータタイプと矛盾する形式のデータを格納する場合、データマップを作成し、そのデータを PowerExchange 式で操作できます。バックされたデータを CHAR カラムに格納する場合、データマップを作成してデータを操作し、そのデータをターゲットにロードする準備をすることができま

す。次に、データマップを抽出マップと統合できます。抽出マップは、関連付けられたデータマップを使用して変更データをマッピングします。

1. 抽出グループを開きます。
2. 抽出マップを開きます。
3. リソースエクスプローラの **【抽出グループ】** タブで、抽出マップを右クリックし、**【キャプチャ登録の変更】** を選択します。
4. **【抽出定義 - キャプチャ登録の変更】** ダイアログボックスが表示されます。**【マージ】** を選択します。
5. **【バルクデータマップ名】** ボックスで、データマップ名を入力するか、データマップを次のように選択します。
 - **【参照】** ボタンをクリックしてファイルを参照します。
 - **【DB2 データマップの選択】** ダイアログボックスで、**【DB2 マップのフィルタデータ】** の下でフィルタ条件を指定し、データマップのリストを制限します。ワイルドカードを含めることができます。
 - **【次へ】** をクリックします。
 - **【使用可能なデータマップ】** リストでデータマップを選択し、**【完了】** をクリックします。
6. **【OK】** をクリックします。

抽出マップは、関連付けられたデータマップを使用して変更データをマッピングします。統合されたデータマップは、**リソースインスペクタの【抽出定義】** タブの **【マージ対象】** フィールドに表示されます。

関連項目：

- [「DB2 カタログデータマップ」 \(ページ 114\)](#)

抽出グループのプロパティの表示または編集

抽出グループのプロパティを表示または編集することができます。

抽出グループの名前、場所、またはソースにアクセスする際に使用するユーザー ID とパスワードを変更するために、プロパティを編集することができます。抽出グループのデータベースタイプを変更することはできません。

1. 抽出グループを開きます。
リソースインスペクタの **【抽出グループ】** タブでは、データソースタイプに応じて抽出グループの以下のプロパティが表示されます。

プロパティ	説明
名前	抽出グループのユーザー定義名です。
アクセス方式	読み取り専用。データソースタイプと結び付いたアクセス方法。
場所	変更をキャプチャするソースの場所です。

プロパティ	説明
ユーザー ID	<p>ソースにアクセスできるユーザー ID です。ユーザー ID のタイプは、次のソースタイプおよび PowerExchange のセキュリティ設定によって異なります。</p> <ul style="list-style-type: none"> - ソースが i5/OS システムまたは z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を (1,x) または (2,x) に設定すると、PowerExchange のセキュリティが有効になります。 - DB2 LUW、Microsoft SQL Server、MySQL、Oracle、または PostgreSQL ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力してください。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。 - サポートされている Linux、UNIX、または Windows システムのソースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『PowerExchange リファレンスマニュアル』を参照してください。
パスワード	<p>指定したユーザー名のパスワードです。</p> <p>データソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N) 以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p>
インスタンス	<p>登録グループからのインスタンス名。</p> <p>Adabas、DB2 for i5/OS、Oracle、または VSAM ソースの場合、インスタンス名はコレクション識別子です。</p> <p>DB2 for z/OS のソースの場合、このインスタンス名がデータベースのインスタンス名になります。</p> <p>Microsoft SQL Server ソースの場合、インスタンス名は登録グループを作成するときに入力した【インスタンス】の値です。インスタンス名を入力しなかった場合、この値はデータベース名の最初の 4 文字と 000 から始まる昇順番号を結合した構成で生成される値になります。</p> <p>MySQL または PostgreSQL ソースの場合、インスタンス名は登録グループを作成するときに入力した【インスタンス】の値です。インスタンス名を入力しなかった場合、この値はデータベースサーバー名の最初の 4 文字と 000 から始まる昇順番号を結合した構成で生成される値になります。</p>

プロパティ	説明
MUF 名	CA Datacom/DB ソースの Multi-User Facility (MUF) 名です。
サーバー	Microsoft SQL Server ソースの場合、パブリケーションサーバーの名前。PostgreSQL ソースの場合、データベースサーバーの名前。
MySQL Server	MySQL ソースの場合、データベースサーバーの名前。
データベース	DB2 LUW、Microsoft SQL Server、または PostgreSQL ソースの場合、ソースデータベースの名前。
DB 名	IDMS ログベースソースのデータベース名です。
Recon ID	IMS ソースの場合はデータベースの RECON 識別子です。
Oracle SID	読み取り専用。Oracle ソースのデータベース用のシステム ID (SID) です。

2. プロパティを編集するには、**リソースインスペクタ**のプロパティボックスをクリックして値を入力します。
プロパティを編集した後に、変更したプロパティの隣に赤い四角が表示されます。
ヒント: 変更を適用する前に取り消す場合は、**[リセット]** をクリックします。
3. 変更を適用するには、**[適用]** をクリックします。
抽出グループ名を編集する場合は、その変更を確認するように求められます。新しい名前が既存の抽出グループ名と一致する場合は、現存グループを上書きするように求められます。
場所を編集する場合は、その場所がインスタンスに対して有効でなければ、PowerExchange ではエラーメッセージが表示されます。
インスタンスを編集する場合は、指定したインスタンスが存在しなければ、PowerExchange ではエラーメッセージが表示されます。
4. 変更を確定するには、**[はい]** をクリックします。
リソースインスペクタにより、編集したフィールドがすべて更新されます。

関連項目：

- [「抽出グループの追加」 \(ページ 198\)](#)

抽出マップのプロパティの表示

抽出マップのプロパティは表示のみで変更できません。

1. 抽出グループを開きます。
2. 抽出マップを開きます。

リソースインスペクタの【抽出定義】タブで、抽出マップのプロパティが表示されます。以下の表に、これらのプロパティを示します。

プロパティ	説明
スキーマ	PowerExchange で生成されたスキーマ名です。
マップ	PowerExchange で生成されたマップ名です。
登録	関連するキャプチャ登録の名前です。
バージョン	バージョン番号です。
表	抽出マップが参照するテーブルです。
マージ対象	DB2 ソースでは、抽出マップがデータマップと統合された場合、データマップ名が表示されます。
Tag	PowerExchange によって生成され、キャプチャ用に登録されたデータソースに割り当てられる一意の識別子です。

関連項目：

- [「抽出グループの追加」](#)（ページ 198）
- [「登録タグ名」](#)（ページ 165）

抽出マップからのデータマップの作成

PowerExchange データマップを抽出マップから作成することができます。

例えば、シーケンシャルデータマップを、DB2 テーブルまた Adabas テーブルと同じレイアウトで作成できます。

PowerExchange では、データマップを、ローカルシステム上の現在のリソース構成ディレクトリに格納します。

抽出マップからデータマップを作成する手順

1. 抽出グループを開き、抽出マップを開きます。
2. 抽出マップを右クリックし、【データマップの作成】をクリックします。
3. 【マップの作成】ダイアログボックスに、以下の情報を入力します。

スキーマ名

スキーマ名（PowerExchange でデータマップ名の最初の部分として使用される名前）を次のように入力します。

schema_name.map_name

マップ名

マップ名（PowerExchange でデータマップ名の 2 番目の部分として使用される名前）を次のように入力します。

schema_name.map_name

キャプチャ制御フィールドを含める

PowerExchanged で生成された次の列をデータマップに含めます。

- DTL__CAPXtype
- DTL__columnname_BI
- DTL__columnname_CI

これらの列をデータマップから除外する場合は、このオプションの選択を取り消します。

デフォルトは選択されています。

バイナリカウントフィールドと Null インジケータフィールドを含める

PowerExchange で生成された列である、DTL__columnname_CNT と DTL__columnname_IND をデータマップに含めます。

これらの列をデータマップから除外する場合は、このオプションの選択を取り消します。

デフォルトでは選択されていません。

4. **[OK]** をクリックします。

PowerExchange では、データマップを、ローカルシステム上の現在のリソース構成ディレクトリに保存します。

抽出マップの削除

使用しない抽出マップを削除することができます。

抽出マップを削除すると、そのマップを変更データの抽出に使用できなくなります。削除した抽出マップを参照する PowerCenter マッピングおよびセッションは、更新する必要があります。

抽出マップを削除しても、PowerExchange では、その抽出マップに関連付けられているキャプチャ登録は削除されません。

抽出マップを削除する手順

1. 抽出マップが開いている場合は、その抽出マップを閉じます。
2. **リソースエクスプローラ**の**【抽出グループ】**タブで抽出マップを選択し、ツールバーの**【削除】**ボタンをクリックします。
3. マップの削除を確認するように求めるメッセージボックスが表示されたら、**【はい】**をクリックします。
PowerExchange では、抽出マップは削除されますが、その抽出マップに関連付けられているキャプチャ登録は削除されません。
4. PowerCenter で、削除した抽出マップを参照する PowerCenter マッピングおよびセッションを更新します。

関連項目：

- [「キャプチャ登録の削除」 \(ページ 192\)](#)

抽出グループの削除

使用しない抽出グループを削除することができます。

抽出グループを削除すると、PowerExchange では、PowerExchange Navigator システム上のローカルグループ情報は削除されますが、グループ内の抽出マップは削除されません。削除したグループとして同じプロパティで別の抽出グループを作成する場合、新しいグループには、削除されたグループ用に存在するすべての抽出マップが含まれます。

PowerExchange では、この抽出グループが生成された対象である登録グループは削除されません。

抽出グループを削除する手順

1. 抽出グループが開いている場合は、その抽出グループを閉じます。
 2. **リソース**ウィンドウで、1 つ以上の抽出グループを削除します。
 3. グループの削除を確認するように求めるメッセージボックスが表示されたら、**[はい]** をクリックします。
- PowerExchange により、単数または複数の抽出グループが削除されます。

抽出グループ内のすべての抽出マップの削除

抽出グループとその抽出マップを完全に削除するには、抽出マップ、関連するキャプチャ登録とその登録グループ、および抽出グループを削除する必要があります。

抽出グループ内のすべての抽出マップを削除する手順

1. 抽出グループの各抽出マップを削除します。
2. 関連付けられているキャプチャ登録とその登録グループを削除します。
3. 抽出グループを削除します。

関連項目：

- [「抽出マップの削除」 \(ページ 211\)](#)
- [「抽出グループの削除」 \(ページ 212\)](#)
- [「キャプチャ登録の削除」 \(ページ 192\)](#)
- [「登録グループの削除」 \(ページ 193\)](#)

抽出定義ウィンドウ - マップタブ

抽出定義ウィンドウの【マップ】タブには、抽出マップ内の各カラムのメタデータが表示されます。

以下の表に、メタデータカラムを示します。

カラム	説明
名前	列の名前です。
カラムの種類	列のタイプです。以下のいずれかの値となります。 <ul style="list-style-type: none">- Real。抽出マップの元の列。- BI。【変更インジケータ/前のイメージの拡張の変更】コマンドで追加された操作前の画像の列。- CI。【変更インジケータ/前のイメージの拡張の変更】コマンドで追加された変更インジケータの画像の列です。- 自動。PowerExchange で生成された列。
DataType	データタイプです。
精度	フィールドの精度です。
スケール	データの位取りです。
長さ	列の長さです。 可変長データタイプの列では、データは、列で許容されない長さ場合があります。

抽出定義ウィンドウ - 拡張タブ

抽出定義ウィンドウの【拡張】タブに、抽出マップの 1 つ以上のカラムに対して追加された、操作前のイメージのカラム (DTL__BI) と変更インジケータのカラム (DTL__CI) が表示されます。

以下の表に、【拡張】タブの情報のカラムを示します。

カラム	説明
名前	データカラム名、つまりデータカラムに追加された変更インジケータまたは操作前のイメージのカラムの名前です。
CI カラム	変更インジケータのカラム名。DTL__CI_data_column_name の形式です。
BI カラム	操作前のイメージのカラム名。DTL__BI_data_column_name の形式です。
実カラム	変更インジケータまたは操作前のイメージのカラムが追加されたデータカラムです。

注: CI および BI カラムのチェックボックスは無効でグレー表示になっています。

関連項目：

- [「データ列用の変更インジケータおよび操作前の画像の追加および削除」 \(ページ 203\)](#)

PowerExchange で生成された抽出マップ内のカラム

キャプチャ登録で定義されたテーブルカラムに加え、PowerExchange が生成したカラムも抽出マップに含まれます。

PowerExchange が生成したこれらのカラムには、SQL の変更のタイプとタイムスタンプなどの CDC 関連の情報が含まれます。

Designer で抽出マップをインポートする場合、PWXPC は、ソース定義に PowerExchange が生成したカラムを含みます。

抽出マップでデータベース行のテストを実行する場合、PowerExchange ナビゲータは、PowerExchange が生成したカラムを結果に表示します。デフォルトでは、抽出マップを開くと、PowerExchange ナビゲータはこれらのカラムを非表示にします。これらのカラムを表示するには、抽出マップを開き、**【抽出定義】** ウィンドウの任意の場所を右クリックして、**【自動生成されたカラムの表示】** を選択します。

注: デフォルトでは、DTL__columnname_CNT、DTL__columnname_IND、DTL__CI_columnname を除くすべてのカラムが抽出マップで選択されています。これらのカラムを追加するには、抽出マップを編集する必要があります。

以下の表に、PowerExchange が変更レコードごとに生成するカラムを示します。

カラム	説明	データ型	長さ
DTL__CAPXRESTART1	<p>その変更レコードの UOW の末尾の位置を表すバイナリ値です。この後に、変更レコード自体の位置が来ます。</p> <p>すべてのデータソースタイプのシーケンストークンが同じ長さを持っている z/OS を除き、シーケンストークンの長さはデータソースのタイプによって異なります。</p> <p>DTL__CAPXRESTART1 の値は別名「シーケンストークン」です。これを リスタートトークンと組み合わせて、リスタートトークンペアを構成します。</p> <p>変更レコードのシーケンストークンは、昇順に限定されており、繰り返し可能な値です。</p>	VARBIN	255
DTL__CAPXRESTART2	<p>変更レコードの UOW 状態の再構築に使用できる変更ストリームの位置を表すバイナリ値です。ただし、次の例外があります。</p> <ul style="list-style-type: none"> - Microsoft SQL Server CDC。ディストリビューションデータベースの DBID およびディストリビューションの名前を格納するバイナリ値です。 - z/OS または i5/OS で完全圧縮ファイルから抽出された変更データ。キャプチャ登録の登録グループからのインスタンス名を格納するバイナリ値です。 <p>リスタートトークンの長さは、データソースのタイプによって異なります。z/OS では、完全圧縮ファイルから抽出された変更データを除き、すべてのデータソースタイプのリスタートトークンが同じ長さです。</p> <p>DTL__CAPXRESTART2 の値は別名「リスタートトークン」です。これをシーケンストークンと組み合わせて、リスタートトークンペアを構成します。</p>	VARBIN	255
DTL__CAPXROWID	<p>PowerExchange Express CDC for Oracle の場合は、物理的な行 ID 値を入力します。PowerExchange では、テーブルで行の移動が有効になっていない場合にのみ、行 ID 値を Oracle テーブルの変更レコードに含めることができます。</p> <p>行 ID 値のキャプチャを有効にするには、Express CDC 構成ファイルで OPTIONS ROWID=Y 文を設定する必要があります。</p> <p>行 ID は、CDC の抽出セッションでキーのないテーブルを処理する場合に役立ちます。</p>	CHAR	18
DTL__CAPXRRN	<p>Db2 for i ソースの場合のみ、2,147,483,647 までの相対レコード番号を指定します。</p> <p>DTL__CAPXRNN は廃止されましたが、PowerExchange 10.5 より以前に作成された既存の抽出マップでは引き続きサポートされます。</p>	NUM32	4
DTL__CAPXERRN	<p>Db2 for i ソースの場合のみ、システムの最大値である 4,294,967,288 までの拡張値を含む相対レコード番号を指定します。</p>	DTLNUM64U	8
DTL__CAPXUOW	<p>変更レコードの UOW の先頭の変更ストリームの位置を表すバイナリ値です。</p>	VARBIN	255

カラム	説明	データ型	長さ
DTL__CAPXUSER	<p>データソースに変更を加えたユーザーのユーザー ID です。ただし、次の例外があります。</p> <ul style="list-style-type: none"> - Adabas 8.3 CDC ソースの場合で Adabas ファイル定義に SY=SECUID のシステムフィールドが含まれる場合、この値はユーザーのセキュリティユーザー ID (SECUID) です。 - Datacom テーブルベース CDC ソースの場合、この値は MUF 名です。 - DB2 for i (i5/OS) CDC ソースの場合、この値は、AS4J CAPI_CONNECTION 文の LIBASUSER パラメータによって決まります。LIBASUSER=Y の場合、この値は、変更が加えられたファイルのライブラリ名とファイル名です。LIBASUSER=M の場合、この値は、変更が加えられたファイルのライブラリ名、ファイル名、およびデータメンバ名です。LIBASUSER=N の場合、この値は、変更を加えたユーザーのユーザー ID です。LIBASUSER=P の場合、この値は、変更を加えたプログラムの名前です。 - DB2 for z/OS CDC ソースの場合、この値は、LRAP CAPI_CONNECTION の UIDFMT パラメータによって決まります。パラメータ設定に応じて、この値は DB2 接続識別子、相関識別子、接続タイプ、プラン名、ユーザー ID、またはこれらのすべての値 (形式 UID:PLAN:CORR:CONN:CTYPE) になります。UIDFMT パラメータを指定しない場合、この値は、変更を加えたユーザーのユーザー ID です。 - IDMS CDC ソースの場合、この値は、ユーザープログラムが、アプリケーションサブスキーマ制御ブロックのプログラム名フィールドに挿入する値です。通常、この値はユーザープログラム名になります。 - IMS 同期 CDC ソースの場合、この値は、LRAP CAPI_CONNECTION 文の UIDFMTIMS パラメータによって決まります。パラメータの設定に応じて、この値は、ユーザー ID、PSB 名、または <i>userid.psbname</i> の形式による両方の値になる場合があります。UIDFMTIMS パラメータを指定しない場合、デフォルトではユーザー ID が使用されます。 - Microsoft SQL Server CDC ソースの場合、この値は、MSQL CAPI_CONNECTION 文の UIDFMT パラメータによって決まります。UIDFMT=DBNAME の場合、この値は SQL Server パブリケーションデータベース名です。UIDFMT=NONE の場合、この値は NULL です。 - Oracle CDC ソースの場合、この値は PowerExchange が Oracle から取得するユーザー ID です (存在する場合)。それ以外の場合、この値は NULL です。 	VARCHAR	255

カラム	説明	データ型	長さ
DTL__CAPXTIMESTAMP	<p>ソース DBMS がデータベースの変更レコードに関して記録するタイムスタンプです。</p> <p>この値は、ソース DBMS がデータベースログの変更レコードに書き込むタイムスタンプの場合と、ソースデータベースに対するトランザクションコミットのタイムスタンプ場合があります。</p> <p>タイムスタンプのタイプは、ソースタイプと特定のパラメータによって異なります。</p> <ul style="list-style-type: none"> - Linux、UNIX、または Windows の DB2 ソースの場合、トランザクションコミットのタイムスタンプです。 - Microsoft SQL Server ソースでは、変更が配布データベースに書き込まれたタイムになります。 - MySQL ソースの場合、MySQL が変更イベントをバイナリログに記録した時刻です。 - Oracle ソースの場合、タイムスタンプの形式は、Express CDC 構成ファイルの OPTIONS 文で TIME_STAMP_MODE パラメータによって制御します。 - UOWC CAPI_CONNECTION 文を必要とするソースの場合はすべて、タイムスタンプの形式は、DBMOVER ファイルの中の UOWC CAPI_CONNECTION 文の TIMESTAMP パラメータによって制御されています。 <p>各ソースタイプのタイムスタンプの詳細については、付録 F, 「DTL__CAPXTIMESTAMP のタイムスタンプ」 (ページ 397)を参照してください。</p> <p>タイムスタンプの形式は次のようになります。</p> <p>YYYYMMDDhhmmssnnnnnn</p> <p>説明:</p> <ul style="list-style-type: none"> - YYYY は西暦を表す 4 桁です。 - MM は月を示します。 - DD は日付です。 - hhmmssnnnnnn は、時間、分、秒、マイクロ秒です。 <p>注: DB2 for Linux, UNIX and Windows と Oracle では、タイムスタンプにマイクロ秒はサポートされません。</p>	CHAR	20

カラム	説明	データ型	長さ
DTL__CAPXACTION	<p>抽出処理中に PowerExchange がターゲットに渡した変更レコードのタイプを示します。このインジケータは、ソースデータベースの SQL 変更操作のタイプに対応します。</p> <p>有効な値は以下のとおりです。</p> <ul style="list-style-type: none"> - I。挿入。 - D。削除。 - U: UPDATE 操作後の画像。 - T: UPDATE 操作前の画像。(ODBC 接続のみ) <p>CDC セッションの接続で 【イメージタイプ】 として 【BA】 を指定した場合は、ソースの更新に対して削除レコードと挿入レコードが PowerExchange によって生成されます。削除レコードの DTL__CAPXACTION カラムの値は D、挿入レコードの DTL__CAPXACTION カラムの値は I になります。</p> <p>CDC セッションの接続で 【イメージタイプ】 として 【AI】 を指定した場合は、1 回の更新に対して 1 つのレコードが PowerExchange によって生成されます。このレコードの DTL__CAPXACTION カラムの値は U になります。</p> <p>ODBC 接続を使用してステー징テーブルに変更データを書き込む場合、ODBC ドライバの CAPXIMAGETYPE パラメータを TU に設定するか、PowerCenter で SQL エスケープシーケンス DTLIMTYPE=TU を入力すると、このカラムに T または U の値を含めることができます。ソースの更新ごとに、操作前の画像と操作後の画像にそれぞれ対応する 2 つのレコードが PowerExchange によってステー징テーブルに渡されます。操作前の画像のレコードの DTL__CAPXACTION カラムの値は T、操作後の画像のレコードの DTL__CAPXACTION カラムの値は U になります。</p>	CHAR	1
DTL__CAPXCASDELIND	<p>(DB2 for z/OS のソースのみ) テーブルが ON DELETE CASCADE 句を指定したために DB2 が行を削除したかどうかを示します。有効な値は以下のとおりです。</p> <ul style="list-style-type: none"> - Y。カスケード削除のルールによって DB2 がこの行を削除したことを示します。 - N。カスケード削除のルールによって DB2 がこの行を削除しなかったことを示します。 	CHAR	1
DTL__BI_< i>columnname	UPDATE 操作が変更される前のカラムの画像です。	ソースカラムのデータ型	ソースカラムのデータ長

カラム	説明	データ型	長さ
DTL__CI_ <i>columnname</i>	<p>UPDATE 操作でカラムの値が変更されたかどうかを示します。有効な値は以下のとおりです。</p> <ul style="list-style-type: none"> - Y。UPDATE 操作によってカラムの値が変更されました。 - N。UPDATE 操作によってカラムが変更されました。 - <i>null</i>。INSERT または DELETE 操作によってカラムが変更されました。UPDATE 操作による変更ではありません。 <p>注: デフォルトでは、Change Indicator カラムは抽出マップに含まれません。このカラムを追加するには、抽出マップを編集してこの自動生成カラムを選択する必要があります。</p>	CHAR	1

カラム	説明	データ型	長さ
DTL__ST_lob_columnname	<p>(DB2 for z/OS LOB カラムの場合) カラムにすべての LOB データが含まれているかどうかを示します。ECCR は、データがベーステーブルスペースに完全にインラインで格納されていない場合、またはサイズが 32KB を超える場合に、不完全な LOB データを提供します。有効な値は以下のとおりです。</p> <ul style="list-style-type: none"> - C: このソースカラムには、すべての LOB データが含まれます。ECCR は、データがベーステーブルスペースに完全にインラインで格納され、サイズが 32KB を超えていないため、すべての LOB データをキャプチャすることができました。 - I: このソースカラムには不完全な LOB データが含まれます。データが補助テーブルスペースに格納されているか、完全にインラインで格納されているがサイズが 32KB を超えているため、ECCR はすべての LOB データをキャプチャできませんでした。 - null: このカラムには null 値のみが含まれます。 <p>LOB データがベーステーブルに完全にインラインで格納されていない場合は、DTL__ST_lob_columnname カラムを含めます。このカラムが不完全な LOB データを示すために I を指定している場合、PowerCenter トランスフォーメーションを使用して、関連付けられたソースカラムの現在のすべての LOB データを取得できます。</p> <p>Oracle LOB カラムの場合、DTL__ST_job_columnname は、関連付けられたソースカラムに LOB データが含まれているかどうかを示します。PowerExchange Express CDC for Oracle は、データが行に完全に格納されている場合、完全な LOB データを提供します。有効な値は以下のとおりです。</p> <ul style="list-style-type: none"> - C: このソースカラムには、すべての LOB データが含まれます。データが行に完全にインラインで格納されているため、PowerExchange Express CDC はすべての LOB データをキャプチャしています。 - I: ソースカラムが変更されましたが、LOB データが含まれていません。データが行に完全にインラインで格納されていないため、PowerExchange Express CDC は LOB データキャプチャできませんでした。 - null: このソースカラムには null 値のみが含まれます。 <p>LOB データがベーステーブルに完全にインラインで格納されていない場合は、DTL__ST_lob_columnname カラムを含めます。このカラムが不完全な LOB データを示すために I を指定している場合、PowerCenter トランスフォーメーションを使用して、関連付けられたソースカラムの現在のすべての LOB データを取得できます。</p> <p>注: このカラムは、デフォルトで抽出マップに含まれています。削除するには、PowerExchange Navigator で抽出マップを開き、この自動生成カラムをクリアします。</p>	CHAR	1

カラム	説明	データ型	長さ
DTL__columnname_CNT	PowerExchange が VARCHAR タイプおよび VARBIN タイプの可変長カラムに対して生成するバイナリカウント。カウントは、変更データの抽出処理中にカラムの長さを決定するために使用されます。 注: デフォルトでは、バイナリカウントカラムは抽出マップに含まれません。このカラムを追加するには、抽出マップを編集してこの自動生成カラムを選択する必要があります。	NUM32U	0
DTL__columnname_IND	NULL 可能カラムに NULL が含まれているかどうかを示します。PowerExchange では、NULL 可能カラムに対してのみこのカラムを生成します。 注: デフォルトでは、NULL インジケータカラムは抽出マップに含まれません。このカラムを追加するには、抽出マップを編集してこの自動生成カラムを選択する必要があります。	BIN	1

第 7 章

パーソナルメタデータ

この章では、以下の項目について説明します。

- [パーソナルメタデータの概要, 222 ページ](#)
- [パーソナルメタデータプロファイルの追加, 223 ページ](#)
- [パーソナルメタデータプロファイル中のテーブルのメタデータの表示, 226 ページ](#)
- [パーソナルメタデータプロファイル内のテーブルのプロパティの表示, 227 ページ](#)
- [パーソナルメタデータプロファイル用プロパティの表示または編集, 227 ページ](#)
- [パーソナルメタデータのリフレッシュ, 228 ページ](#)
- [パーソナルメタデータプロファイルの削除, 228 ページ](#)

パーソナルメタデータの概要

リレーショナルと非リレーショナルソースおよびターゲットのパーソナルメタデータプロファイルを定義し、接続のテストや、メタデータおよびデータへのアクセスを行うことができます。

パーソナルメタデータプロファイルは、一括データ移動操作や変更データキャプチャ（Change Data Capture：CDC）の実行には必要ありません。

ソースまたはターゲットのメタデータを表示するには、パーソナルメタデータプロファイルを作成します。プロファイル中のテーブルを表示して、テーブル中の各列のメタデータ（データタイプ、日付形式、CCSID など）を確認します。プロファイル中の複数のテーブルのメタデータを同時に表示できます。ソースまたはターゲットのデータをプレビューするには、プロファイル中のテーブルにデータベース行のテストを実行します。

パーソナルメタデータプロファイルを開くおよび閉じる方法の詳細については、[「PowerExchange リソースを開く」 \(ページ 31\)](#)および[「PowerExchange リソースを閉じる」 \(ページ 33\)](#)を参照してください。

パーソナルメタデータプロファイル中のテーブルおよびテーブル内の列を検索する方法の詳細については、[「リソースの検索」 \(ページ 31\)](#)を参照してください。

PowerExchange には、パーソナルメタデータプロファイルのサンプル、demo_oracle が用意されています。このサンプルプロファイルを使用するには、そのプロパティを編集し、Oracle インスタンスを指すようにします。

パーソナルメタデータプロファイルの追加

ソースまたはターゲットに接続する際に使用するパーソナルメタデータプロファイルを追加します。

フィルタを使用してプロファイルに含めるテーブルを制限できます。

1. リソースエクスプローラの [リソース] タブで、[追加] - [パーソナルメタデータ] をクリックします。
2. [パーソナルメタデータ - 名前] ダイアログボックスに、以下の情報を入力します。

名前

パーソナルメタデータプロファイルのユーザー定義名です。

場所

メタデータを表示するソースまたはターゲットの場所です。ソースまたはターゲットが PowerExchange Navigator システムにある場合は、[ローカル] を選択します。

このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文に定義された場所が表示されます。

デフォルトは [ローカル] です。

タイプ

データソースタイプまたはターゲットタイプ。次のいずれかのタイプを選択し、そのタイプのフィールドに入力します。

- **DB2**。DB2 for i5/OS または DB2 for z/OS ソースまたはターゲット。
DB2 for i5/OS の場合は、必要に応じて、データベース名を [SSID または DB 名] フィールドに入力します。デフォルトは、デフォルトのデータベースです。
DB2 for z/OS の場合は、SSID を [SSID または DB 名] フィールドに入力します。
- **DB2400C**。DB2 Call Level Interface (CLI) for i5/OS ソース。
データベース名を [SSID または DB 名] フィールドに入力します。
- **DB2UDB**。DB2 for Linux、UNIX、および Windows のソースまたはターゲットです。
オプションとして、データベース名を [DB 名] フィールドに入力します。
- **MSSQL**。Microsoft SQL Server のソースまたはターゲットです。
SQL Server インスタンス名を [DSN] フィールドに入力します。オプションとして、接続するデータベースをオーバーライドするデータベース名を [データベース] フィールドに入力します。
- **MySQL**。MySQL ソースです。
MySQL データベースサーバー名を [サーバー] フィールドに入力します。
- **NRDB** または **NRDB2**。Adabas、CA Datacom/DB、CA IDMS/DB、IMS、VSAM、フラットファイルなどの非リレーショナルソース用の PowerExchange データマップ。
オプションとして、オーバーライドファイル名を [ファイル名のオーバーライド] フィールドに入力します。
データベース行のテストで生成された SQL に対し、**NRDB2** データソースタイプは 2 階層式の命名規則を使用し、**NRDB** は 3 階層式の命名規則を使用します。PowerExchange データマップには、**NRDB** または **NRDB2** を使用できます。
- **ORACLE**。Oracle ソースまたはターゲット。
オプションとして、Oracle インスタンスの TNS 名を [SQL*Net 名] フィールドに入力します。
- **PGSQL**。PostgreSQL ソース。

注: 次のソースタイプまたはターゲットタイプは廃止されていますので選択しないでください。

- AS4_META
- CAPX
- CAPXRT
- ODBC
- TASK_CNTL

ユーザー ID

ソースにアクセスできるユーザー ID を入力します。ユーザー ID のタイプは、次のソースタイプおよび PowerExchange のセキュリティ設定によって異なります。

- ソースが i5/OS システムまたは z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を(1,x)または(2,x)に設定すると、PowerExchange のセキュリティが有効になります。
- DB2 for Linux, UNIX, and Windows、Microsoft SQL Server、MySQL、Oracle、または PostgreSQL ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。
- サポートされている Linux、UNIX、または Windows システムのソースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『PowerExchange リファレンスマニュアル』を参照してください。

パスワード

指定したユーザー名のパスワードです。

ソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。
' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。

z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。

ユーザー ID を指定してパスワードやパスフレーズを省略した場合、パーソナルメタデータプロファイルを開いたときに PowerExchange Navigator は **【パスワードが入力されていません】** ダイアログボックスにパスワードの入力を求めるメッセージを表示します。

タイプ固有のフィールド

タイプ	フィールド
DB2、DB2400C、または DB2UDB	SSID または DBNAME。DB2 for z/OS では、DB2 サブシステム ID を入力します。DB2400C または DB2UDB では、DB2 データベース名を入力します。
MSSQL	Server。SQL Server パブリケーションサーバー名。 【データベース】 。SQL Server データベース名。
MYSQL	Server。MySQL データベースサーバー名。
ORACLE	SQL *Net 名 Oracle インスタンスの TNS 名。
PGSQL	Server。PostgreSQL データベースサーバー名。 【データベース】 。PostgreSQL データベース名。

説明

パーソナルメタデータプロファイルのユーザー定義の説明です。

3. **【次へ】** をクリックします。
4. **【パーソナルメタデータ - フィルタ】** ダイアログボックスに以下の情報を入力し、プロファイルに含めるテーブルを制限します。

データのフィルタ

ソースタイプまたはターゲットタイプに応じて、次のフィルタフィールドにオプションのフィルタ条件を入力します。

- DB2、DB2400C、または DB2UDB タイプの場合、**【クリエータ/スキーマ】**、**【DB 名/定義元】**、および **【テーブル】** ボックスにフィルタを入力します。
- MYSQL、ORACLE、または PGSQL タイプの場合、**【スキーマ】** および **【テーブル】** ボックスにフィルタを入力します。
- MSSQL タイプの場合、**【オーナー】** および **【テーブル】** ボックスにフィルタを入力します。
- NRDB または NRDB2 タイプの場合、**【スキーマ】**、**【マップ名】**、および **【テーブル】** ボックスにフィルタを入力します。

どのフィルタフィールドでも、次のワイルドカード文字が使用できます。

- 1 つ以上の一致文字を表すアスタリスク (*)
- 疑問符 (?) 疑問符 (?)

大文字と小文字を区別する

デフォルトでは、いずれかのフィルタボックスに入力されたテキストはデータソースの標準文字に変換されます。例えば、Oracle では、大文字が標準であり、scott または SCOTT の値は同じ結果の SCOTT となります。

PowerExchange によってデータソース側で標準の文字種に強制変換されては困るフィルタには、**【大文字と小文字を区別する】** チェックボックスをオンにします。

エスケープ文字

使用するエスケープ文字を入力して、アスタリスク (*) または疑問符 (?) を区切ります。エスケープ文字は、フィルタフィールドでワイルドカードではなくリテラル値として使用されます。

例えば、~のエスケープ文字を指定する場合、tab*のフィルタ値は、tab で始まるすべてのテーブルを返します。tab~*のフィルタ値は、tab*と名付けられたテーブルを返します。

注: マルチバイトが有効化されたデータソースではエスケープ文字を使用しないでください。

5. **【完了】** をクリックします。

リソースエクスプローラの**【パーソナルメタデータ】** タブには、パーソナルメタデータプロファイルと、指定したフィルタに一致するすべてのテーブルが表示されます。

6. パーソナルメタデータプロファイルでデータのプレビューが可能であることを確認するには、プロファイル中のテーブルにデータベース行のテストを実行します。

関連項目：

- [「パーソナルメタデータプロファイルのテスト」 \(ページ 233\)](#)

パーソナルメタデータプロファイル中のテーブルのメタデータの表示

パーソナルメタデータプロファイル中の 1 つ以上のテーブル内の列のメタデータを表示することができます。

一度に 1 つのテーブルまたは複数のテーブルのメタデータを表示することができ、どちらも個別のテーブルウィンドウで表示されます。

パーソナルメタデータプロファイル内のテーブルのメタデータを表示する手順

1. **リソースエクスプローラ**の**【パーソナルメタデータ】** タブで、次のうち 1 つのアクションを完了します。
 - 単一テーブルを表示するには、そのテーブルをダブルクリックします。
 - 複数テーブルを表示するには、別々の**テーブルウィンドウ**で、各テーブルを右クリックし、**エクスプローラ**をクリックします。

テーブルウィンドウには、各テーブル列の次のメタデータが表示されます。

- **名前。** 列の名前です。
 - **タイプ。** データタイプです。
 - **精度。** データの精度です。
 - **位取り。** データの位取りです。
 - **NULL 可能。** 列が NULL 可能であるかどうかを示す、YES または NO の値です。
 - **PowerExchange の種類。** PowerExchange データ型です。PowerExchange では、一部のデータタイプに対し独自の指示があります。
 - **コメント。** テーブルに対するコメントです。
2. メタデータを名前またはデータタイプでアルファベット順にソートするには、**テーブルウィンドウ**の**【名前】** または**【タイプ】** 見出しをクリックします。メタデータを逆アルファベット順でソートするには、列の見出しを再度クリックします。

次のように、ソート順を示す【名前】または【型】の見出しに小さな矢印が表示されます。

- 下矢印はアルファベット順を示します。
- 上矢印は逆アルファベット順を示します。

パーソナルメタデータプロファイル内のテーブルのプロパティの表示

パーソナルメタデータプロファイル中のテーブルのテーブル名、オブジェクトタイプ、およびコメントを表示できます。

- ▶ リソースエクスプローラの【パーソナルメタデータ】タブで、パーソナルメタデータプロファイル中のテーブルを右クリックし、【プロパティ】をクリックします。

【パーソナルメタデータのテーブルプロパティ】ダイアログボックスには、次の情報が表示されます。

名前

テーブルの名前にはパーソナルメタデータのタイプに応じて以下の形式があります。

- NRDB の場合:
schema.map.table
- NRDB2 の場合:
schema.map_table
- リレーショナルソースの場合:
schema.table

タイプ

オブジェクトのタイプです。

コメント

テーブルに対するコメントです。

パーソナルメタデータプロファイル用プロパティの表示または編集

パーソナルメタデータプロファイルのプロパティを表示または編集することができます。

パーソナルメタデータプロファイルを編集して、テーブルに対するフィルタ条件を変更することができます。

1. パーソナルメタデータプロファイルを開きます。
2. リソースエクスプローラの【パーソナルメタデータ】タブで、パーソナルメタデータプロファイルを右クリックして【プロパティ】をクリックします。
【パーソナルメタデータプロパティ】ダイアログボックスが表示されます。
3. 【名前】タブ、【フィルタ】タブ、またはこの両方で、プロファイルのプロパティを表示または編集します。
4. 【OK】をクリックします。

関連項目：

- [「パーソナルメタデータプロファイルの追加」 \(ページ 223\)](#)

パーソナルメタデータのリフレッシュ

パーソナルメタデータプロファイルで表示されるオブジェクトをリフレッシュできます。

新しいテーブルまたはビューをソースに定義した後、パーソナルメタデータのリフレッシュが必要になる場合があります。あるいは、プロファイルを開いたときに PowerExchange Listener がソースシステムで実行されていなかった場合、PowerExchange Listener を起動してパーソナルメタデータをリフレッシュできます。

パーソナルメタデータのリフレッシュ手順

- ▶ **リソースエクスプローラ**の **「パーソナルメタデータ」** タブで、パーソナルメタデータプロファイルを右クリックして **「リフレッシュ」** をクリックします。

パーソナルメタデータプロファイル中のメタデータが、データソースの現在のメタデータでリフレッシュされます。

パーソナルメタデータプロファイルの削除

パーソナルメタデータプロファイルは削除できます。

パーソナルメタデータプロファイルを削除する手順

1. パーソナルメタデータプロファイルが開いている場合は、閉じます。
2. **リソースエクスプローラ**の **「リソース」** タブで、プロファイルを右クリックし、**「削除」** をクリックします。
3. プロファイルの削除を確認するように求めるメッセージボックスが表示されたら、**「はい」** をクリックします。

PowerExchange によってプロファイルが削除されます。

第 8 章

データベース行のテスト

この章では、以下の項目について説明します。

- [データベース行テストの概要, 229 ページ](#)
- [データマップのテスト, 231 ページ](#)
- [抽出マップのテスト, 232 ページ](#)
- [パーソナルメタデータプロファイルのテスト, 233 ページ](#)
- [データベース行のテスト内の PowerExchange リスナコマンドの発行, 234 ページ](#)
- [変更データキャプチャのリスタートトークンの生成, 235 ページ](#)
- [\[データベース行のテスト\] ダイアログボックス, 237 ページ](#)
- [CAPX の \[詳細パラメータ\] ダイアログボックス - \[全般\] タブ, 246 ページ](#)
- [CAPX の \[詳細パラメータ\] ダイアログボックス - \[AS400\] タブ, 248 ページ](#)
- [CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[全般\] タブ, 249 ページ](#)
- [CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[Adabas\] タブ, 253 ページ](#)
- [CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[AS400\] タブ, 253 ページ](#)
- [CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[DB2UDB\] タブ, 253 ページ](#)
- [CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[MSSQL\] タブ, 254 ページ](#)
- [\[CAPXRT 詳細パラメータ\] ダイアログボックス - \[Oracle\] タブ, 254 ページ](#)
- [DB2390IMG の \[詳細パラメータ\] ダイアログボックス, 255 ページ](#)
- [\[詳細パラメータ\] ダイアログボックス \(IMS\) , 255 ページ](#)

データベース行テストの概要

PowerExchange リソースでデータベース行のテストを実行して、すべてのデータソースのデータをプレビューし、PowerExchange 変更データキャプチャソースのリスタートトークンを生成することができます。

また、データベース行のテストを実行して、PowerExchange Navigator 経由で PowerExchange Listener コマンドを発行することもできます。

データをプレビューするには、以下の PowerExchange リソースでデータベース行のテストを実行します。

- データマップ。データベース行のテストを使用して、データマップを検証し、テーブルのソースデータをデータマップでプレビューすることができます。
- 抽出マップ。データベース行のテストを使用して、キャプチャされた変更データをプレビューすることができます。

- パーソナルメタデータプロファイル。データベース行のテストを使用して、パーソナルメタデータのプロファイルにあるテーブルのソースデータをプレビューすることができます。

データベース行のテストはデータソース内のデータにアクセスし、そのデータをテーブル形式で**データベース行のテストによる出力**ウィンドウに表示します。データマップまたは抽出マップの行のテスト結果により、一括データ移動操作や変更データ抽出の実行時に PowerExchange でデータにアクセスできることが示されます。

データベース行のテストでデータソースのデータをプレビューするほか、SQL DTLDESCRIBE 文を使用してデータマップや抽出マップでメタデータをプレビューすることができます。データソースタイプに応じて、列、外部キーおよびプライマリキー、手順およびプロシージャ列、レコード、スキーマ、テーブルのメタデータをプレビューすることができます。**[データベース行のテスト]** ダイアログボックスでメタデータをプレビューする場合、PowerExchange は **[SQL 文]** ボックスに SQL DTLDESCRIBE 文を生成します。この文は編集することができます。

注: DTLDESCRIBE 構文は、リレーショナルソースと非リレーショナルソースの両方に有効です。

DTLDESCRIBE 構文の詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

データベース行のテストを実行して PowerExchange 変更データキャプチャソース用のリスタートトークンを生成する場合、PowerExchange Navigator により、生成されたリスタートトークンが**データベース行のテストによる出力**ウィンドウに表示されます。このウィンドウの出力をコピーして、PWXPCL リスタートトークンファイルにデータを入力することができます。

PowerExchange Navigator でのデータ行テスト中にメッセージ PWX-01936 が発行された場合、適切な Unicode フォントがありません。この場合、PowerExchange インストールディレクトリにあるファイルをインストールし、Windows フォントフォルダにコピーします。

データベース行のテストによる出力ウィンドウの表示または非表示

データベース行のテストによる出力ウィンドウを、表示または非表示にすることができます。

ウィンドウを表示する場合、PowerExchange Navigator GUI の下部に最新のデータベース行のテスト結果が表示されます。

データベース行のテストによる出力ウィンドウを表示または非表示にする手順

- ▶ **[表示]** - **[出力]** をクリックします。

注: このオプションは、データベース行のテストの実行後に有効になります。

データベース行のテストおよび非標準コードページデータ

z/OS と i5/OS の両方の DB2 から、マルチバイトデータを含む非標準コードページデータを選択することができます。PowerExchange Navigator ではダブルバイト文字を表示することができます。

PowerExchange Navigator では UTF-16LE コードページを使用するため、ワイド文字セットを表示することができます。

データベース行のテストでデータをプレビューする前に、データのアクセス元のシステムの PowerExchange 構成ファイルで次の文を編集し、データが正しい形式で表示されることを確認します。

- z/OS の DB2 用 DB2CODEPAGE
- i5/OS の DB2 用 DB2_BIN_CODEPAGE

コードページの詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

データマップのテスト

データベース行のテストを実行してデータマップを検証し、データマップのテーブルデータをプレビューできます。

データベース行のテストはデータソース内のデータにアクセスし、そのデータをテーブル形式で**データベース行のテスト**による出力ウィンドウに表示します。

1. データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブでテーブルをクリックします。
3. 特定の列のデータを取得するには、**テーブルウィンドウ**でその列を選択します。

PowerExchange は、選択された列を **【データベース行のテスト】** ダイアログボックスの **【SQL 文】** ボックスで生成される SQL SELECT 文に追加します。

注: この設定は、SQL DTLDESCRIBE 要求には影響しません。

4. **【ファイル】** > **【データベース行のテスト】** をクリックします。
【データマップリモートノード】 ダイアログボックスが表示されます。
5. リモートシステムの接続情報を入力して、**【OK】** をクリックします。
【データベース行のテスト】 ダイアログボックスが表示されます。
6. **【DB タイプ】** リストで、データソースタイプをクリックします。次に、データソースタイプに関連付けられている情報を入力します。

注: AS4_META データソースタイプは、Informatica グローバルカスタマサポートからの指示があった場合にのみ選択します。

7. データをプレビューするには、**【フェッチ】** リストで **【データ】** を選択します。テーブルやスキーマなど、項目のメタデータをプレビューするには、別のオプションを選択します。
行のテストでは、選択した **【フェッチ】** オプションに基づいて、**【SQL 文】** ボックスに SQL 文が生成されます。

- **【データ】** オプションの場合、行のテストで SQL SELECT 文が生成されます。
- メタデータオプションの場合、行のテストで SQL DTLDESCRIBE 文が生成されます。

8. オプションとして、**【SQL 文】** ボックスで SQL SELECT 文または DTLDESCRIBE 文を編集します。
9. **CAPX** または **CAPXRT** アクセス方式を選択した場合、**【アプリケーション】** ボックスにアプリケーション名を入力します。
10. **CAPX**、**CAPXRT**、**DB2390IMG**、または **NRDB** アクセス方式を選択した場合、**【詳細】** をクリックして行テストのオーバーライド値を入力します。次に、**【詳細パラメータ】** ダイアログボックスに、ソースタイプに指定可能なオーバーライド値を入力します。入力した値は PowerExchange 定義または構成ファイル（データマップや DBMOVER 構成ファイルなど）の特定の値をオーバーライドします。
11. **【*n*行を取得】** フィールドに、行のテストでフェッチする行数を入力します。一度に最大 99999 行のデータを表示できます。デフォルトでは、PowerExchange に 10 行のデータが表示されます。
12. **【CAPX】**、**【CAPXRT】**、**【NRDB】**、**【NRDB2】**、または **【IMSUNLD】** データソースタイプを選択し、**【フェッチ】** リストで **【データ】** を選択した場合、行のテストで特定数の先頭行をスキップしてから、データを表示することができます。行をスキップするには、**【先頭の *n*行をスキップする】** フィールドで 1 から 99999 までの数値を指定します。

制限: **【可変】** プロパティが **【VS】** に設定され、可変長ストリームデータファイルであることを示す SEQ データマップをテストすると、データベース行のテストで警告メッセージ PWX-03042 が返され、**【先頭の *n*行をスキップする】** オプションは無視されます。PowerExchange Navigator では、可変長ストリームデータファイルの行はスキップできません。

13. **【実行】** をクリックします。

データベース行のテストによる出力ウィンドウに、データベース行のテストの結果が表示されます。

関連項目：

- [「\[データベース行のテスト\] ダイアログボックス」 \(ページ 237\)](#)
- [「CAPX の \[詳細パラメータ\] ダイアログボックス - \[全般\] タブ」 \(ページ 246\)](#)
- [「CAPX の \[詳細パラメータ\] ダイアログボックス - \[AS400\] タブ」 \(ページ 248\)](#)
- [「CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[全般\] タブ」 \(ページ 249\)](#)
- [「CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[Adabas\] タブ」 \(ページ 253\)](#)
- [「CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[AS400\] タブ」 \(ページ 253\)](#)
- [「CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[DB2UDB\] タブ」 \(ページ 253\)](#)
- [「CAPXRT の \[詳細パラメータ\] ダイアログボックス - \[MSSQL\] タブ」 \(ページ 254\)](#)
- [「\[CAPXRT 詳細パラメータ\] ダイアログボックス - \[Oracle\] タブ」 \(ページ 254\)](#)
- [「DB2390IMG の \[詳細パラメータ\] ダイアログボックス」 \(ページ 255\)](#)
- [「\[詳細パラメータ\] ダイアログボックス \(IMS\)」 \(ページ 255\)](#)

抽出マップのテスト

抽出マップでデータベース行のテストを実行してキャプチャした変更データをプレビューし、キャプチャ登録への変更をテストすることができます。

1. 抽出グループと抽出マップを開きます。
2. 特定の列のデータを取得するには、**抽出定義**ウィンドウでその列を選択します。
PowerExchange は、選択された列を **【データベース行のテスト】** ダイアログボックスの **【SQL 文】** ボックスで生成される SQL SELECT 文に追加します。
注: テーブルで特定の列を選択しても、SQL DTLDESCRIBE 要求には影響しません。
3. **【ファイル】** > **【データベース行のテスト】** をクリックします。
【データベース行のテスト】 ダイアログボックスが表示されます。
4. **【DB タイプ】** リストで、データソースタイプをクリックします。次に、データソースタイプに関連付けられている情報を入力します。
注: AS4_META データソースタイプは、Informatica グローバルカスタマサポートからの指示があった場合にのみ選択します。
5. データをプレビューするには、**【フェッチ】** リストで **【データ】** を選択します。テーブルやスキーマなど、項目のメタデータをプレビューするには、別のオプションを選択します。
行のテストでは、選択した **【フェッチ】** オプションに基づいて、**【SQL 文】** ボックスに SQL 文が生成されます。
 - **【データ】** オプションの場合、行のテストで SQL SELECT 文が生成されます。
 - メタデータオプションの場合、行のテストで SQL DTLDESCRIBE 文が生成されます。
6. オプションとして、**【SQL 文】** ボックスで SQL SELECT 文または DTLDESCRIBE 文を編集します。
7. **CAPX** または **CAPXRT** アクセス方式を選択した場合、**【アプリケーション】** ボックスにアプリケーション名を入力します。

8. **CAPX**、**CAPXRT**、**DB2390IMG**、または **NRDB** アクセス方式を選択した場合、**【詳細】** をクリックして行テストのオーバーライド値を入力します。次に、**【詳細パラメータ】** ダイアログボックスに、ソースタイプに指定可能なオーバーライド値を入力します。入力した値は PowerExchange 定義または構成ファイル（データマップや DBMOVER 構成ファイルなど）の特定の値をオーバーライドします。
9. **【*n*行を取得】** フィールドに、行のテストでフェッチする行数を入力します。一度に最大 99999 行のデータを表示できます。デフォルトでは、PowerExchange に 10 行のデータが表示されます。
10. **CAPX**、**CAPXRT**、**NRDB**、または **NRDB2** データソースタイプを選択し、**【フェッチ】** リストで **【データ】** を選択した場合、行のテストで特定数の先頭行をスキップしてから、データを表示することができます。行をスキップするには、**【先頭の *n* 行をスキップする】** フィールドで 1 から 99999 までの数値を指定します。
11. **【実行】** をクリックします。

データベース行のテストによる出力ウィンドウに、データベース行のテストの結果が表示されます。これは、データキャプチャに登録したテーブルで発生した変更を表します。

操作前または操作後の画像列を抽出マップに追加する場合、これらの列もデータベース行のテストによる出力に表示されます。

関連項目：

- [「データ列用の変更インジケータおよび操作前の画像の追加および削除」 \(ページ 203\)](#)

パーソナルメタデータプロファイルのテスト

パーソナルメタデータプロファイルのテーブルにデータベース行のテストを実行し、データをプレビューすることができます。

警告： PowerExchange Navigator では、PowerExchange がサポートしないデータタイプの列を選択することができます。サポートされていないデータタイプの列を含めると、データベース行のテストは失敗します。

1. パーソナルメタデータプロファイルを開きます。
2. **【リソースエクスプローラ】** の **【パーソナルメタデータ】** タブで、テーブルをダブルクリックして開きます。
3. 特定の列のデータを取得するには、**テーブルウィンドウ** でその列を選択します。
PowerExchange は、選択された列を **【データベース行のテスト】** ダイアログボックスの **【SQL 文】** ボックスで生成される SQL SELECT 文に追加します。
4. **【ファイル】** > **【データベース行のテスト】** をクリックします。
【データベース行のテスト】 ダイアログボックスが表示されます。
5. **【DB タイプ】** リストで、データソースタイプをクリックします。次に、データソースタイプに関連付けられている情報を入力します。
注: **AS4_META** データソースタイプは、Informatica グローバルカスタマサポートからの指示があった場合にのみ選択します。
6. **【フェッチ】** リストが無効になっています。パーソナルメタデータプロファイルの場合、**【データ】** がデフォルトオプションです。
7. オプションとして、**【SQL 文】** ボックスで SQL SELECT 文または DTLDESCRIBE 文を編集します。
8. **CAPX** または **CAPXRT** アクセス方式を選択した場合、**【アプリケーション】** ボックスにアプリケーション名を入力します。

9. **CAPX、CAPXRT、DB2390IMG**、または **NRDB** アクセス方式を選択した場合、**[詳細]** をクリックして行テストのオーバーライド値を入力します。次に、**[詳細パラメータ]** ダイアログボックスに、ソースタイプに指定可能なオーバーライド値を入力します。入力した値は PowerExchange 定義または構成ファイル（データマップや DBMOVER 構成ファイルなど）の特定の値をオーバーライドします。
10. **[*n*行を取得]** フィールドに、行のテストでフェッチする行数を入力します。一度に最大 99999 行のデータを表示できます。デフォルトでは、PowerExchange に 10 行のデータが表示されます。
11. **CAPX、CAPXRT、NRDB**、または **NRDB2** データソースタイプを選択し、**[フェッチ]** リストで **[データ]** を選択した場合、行のテストで特定数の先頭行をスキップしてから、データを表示することができます。行をスキップするには、**[先頭の *n*行をスキップする]** フィールドで 1 から 99999 までの数値を指定します。
12. **[実行]** をクリックします。
データベース行のテストによる出力ウィンドウに、データベース行のテストの結果が表示されます。

データベース行のテスト内の PowerExchange リスナコマンドの発行

【データベース行のテスト】ダイアログボックスを使用して、任意のシステムで実行されている PowerExchange リスナに対して LISTTASK または STOPTASK コマンドを発行するか、LISTLOCATIONS コマンドを発行できます。

PowerExchange リスナの LISTTASK コマンドはリスナで実行されているクライアントタスクを表示し、STOPTASK コマンドはタスクを停止します。LISTLOCATIONS コマンドは、DBMOVER 構成ファイルの NODE および SVCNODE 文で定義されている場所についての情報を表示します。

1. **[リソースエクスプローラ]** で、PowerExchange リスナが実行されている場所に対して定義されているデータマップをダブルクリックして、データマップを開きます。
注: または、PowerExchange リスナの場所に定義された抽出マップ、またはパーソナルメタデータプロファイルを開くこともできます。
2. **[データマップ]** タブでテーブルビューを選択し、メニューバーで **[ファイル] > [データベース行のテスト]** の順にクリックします。
データマップをリモートの場所へ送信するよう求めるメッセージが表示されることがあります。
3. **[データマップリモートノード]** ダイアログボックスに PowerExchange リスナが実行されている場所の接続情報を入力し、**[OK]** をクリックします。
[データベース行のテスト] ダイアログボックスが表示されます。
4. **[DB タイプ]** リストで、**[TASK_CNTL]** を選択します。
5. **[フェッチ]** リストで、次のうち 1 つのコマンドを選択します。
 - **場所の一覧表示。** PowerExchange リスナが実行されているシステムで、DBMOVER 構成ファイルの NODE または SVCNODE 文で定義されている場所についての情報を表示します。出力には、ノード名、IP アドレス、ポート番号、送受信バッファサイズおよび長さ、受信タイムアウト、SSL の使用が含まれます。
 - **タスクの一覧表示。** PowerExchange リスナで実行されているアクティブな各タスクについての情報を表示します。出力には、タスク ID、TCP/IP アドレス、ポート番号、アプリケーション名、アクセスタイプ、およびステータスが含まれます。
 - **タスクの停止。** 特定の PowerExchange リスナタスクを停止します。タスクを特定するには、タスク ID またはアプリケーション名を入力する必要があります。

6. STOPTASK コマンドを発行する場合は、**【SQL 文】** ボックスにタスク ID またはアプリケーション名を入力します。以下の構文を使用します。
- ```
stoptask {taskid=task_id|appname=application_name}
```
- 中括弧は使用しないでください。これらの括弧は、taskid または appname の選択が必須であることを示します。
- 注: 【アプリケーション】** フィールドにアプリケーション名を入力しないでください。TASK\_CNTL コマンドでは、**【アプリケーション名】** フィールドは無視されます。
7. LISTLOCATIONS コマンドを発行する場合は、必要に応じて **【SQL 文】** ボックスにノードタイプを入力します。以下の構文を使用します。
- ```
listlocations nodetype={N|A|S}
```
- ノードタイプに以下のいずれかの値を指定します。
- N: デフォルト。DBMOVER ファイル内の NODE 文で定義されている場所を一覧表示します。
 - A: DBMOVER ファイル内の NODE または SVCNODE 文で定義されている場所を一覧表示します。
 - S: DBMOVER ファイル内の SVCNODE 文で定義されている場所を一覧表示します。
8. **【実行】** をクリックします。
- データベース行のテストによる出力ウィンドウにコマンドの出力が表示されます。

関連項目：

- [「リモートシステムへのデータマップの送信」 \(ページ 95\)](#)

変更データキャプチャのリスタートトークンの生成

データベース行のテストを実行して、データソースのリスタートトークンを生成することができます。

生成されたこれらのリスタートトークンを使用して、PowerCenter セッションの PWXPC リスタートトークンファイルにデータを入力します。PowerExchange では、指定した場所とソースタイプを使用してリスタートトークンが生成されます。これらのリスタートトークンは、データベース行のテストを実行した時点での変更ストリームの最後を表します。PowerExchange ナビゲータにより、生成されたリスタートトークンが**データベース行のテストによる出力ウィンドウ**に表示されます。このウィンドウの出力をコピーして、PWXPC リスタートトークンファイルを入力します。

または、次のうち 1 つの方法を使用してリスタートトークンを生成します。

- CURRENT_RESTART を使用して、PWXPC リスタートトークンファイルに、特別なオーバーライド文を含める。PWXPC および PowerExchange では、PowerCenter セッションが実行される時点での変更ストリームの最後を表すリスタートトークンが生成されます。
 - GENERATE RSTTKN パラメータおよび有効なキャプチャ登録と共に DTLUAPPL ユーティリティを使用する。DTLUAPPL は、ユーティリティが実行される時点での変更ストリームの最後を表すリスタートトークンを生成します。
1. リスタートトークンを生成するキャプチャ登録の抽出マップを開きます。
 2. リソースエクスプローラの **【抽出マップ】** タブで、抽出マップをクリックして **【ファイル】** > **【データベース行のテスト】** をクリックします。
【データベース行のテスト】 ダイアログボックスが表示されます。
 3. **【DB タイプ】** リストで **CAPXRT** を選択します。
 4. **【場所】** リストで、データソースの場所を表すノード名を選択します。

【場所】 リストに表示された場所は、PowerExchange ナビゲータシステムの dbmover.cfg コンフィギュレーションファイルの NODE 文によって定義されます。

5. 【アプリケーション】 ボックスに、1 から 20 文字の長さのアプリケーション名を入力します。
6. 最新のリスタートトークンを生成するには、【SQL 文】 ボックスに以下の PowerExchange SQL 文を入力します。

```
SELECT CURRENT_RESTART [WHERE [{CONNAME=capi_connection_name|CONTYPE=connection_type}] [[AND]
INSTANCE=db2_database_name]]
```

この SELECT CURRENT_RESTART 文には、以下のパラメータがあります。

CONNAME=*capi_connection_name*

ソースの場所で、PowerExchange がデータソースタイプの決定に使用する、DBMOVER ファイル内の CAPI_CONNECTION 文の名前を指定します。PowerExchange でこの情報が必要な理由は、リスタートトークンの形式がソースタイプによって異なるためです。ソースの場所で複数の CAPI_CONNECTION 文が定義されている場合は、このパラメータを入力します。さもなければ、このパラメータはオプションであるため、代わりに CONTYPE を指定できます。CONNAME と CONTYPE の両方は指定しないでください。

CONTYPE=*connection_type*

PowerExchange がデータソースタイプを決定するために使用する接続タイプを指定します。PowerExchange でこの情報が必要な理由は、リスタートトークンの形式がソースタイプによって異なるためです。

conn_type 変数に、次のいずれかの値を指定します。

- **ADA** (Adabas ソース (z/OS) の場合)。
- **AS4** (DB2 for i5/OS ソースの場合)。
- **DCM** (CA Datacom/DB ソース (z/OS) の場合)。
- **DB2**。DB2 for z/OS ソース (z/OS) です。
- **IDL** (CA IDMS/DB ログベースソース (z/OS) の場合)。
- **IML** (IMS ログベースソース (z/OS) の場合)。
- **IMS** (IMS 同期ソース (z/OS) の場合)。
- **ORA** (Oracle ソースの場合)。
- **MSS** (Microsoft SQL Server ソースの場合)。
- **MYSQL** (MySQL ソースの場合)。
- **PGS** (PostgreSQL ソースの場合)。
- **UDB** (DB2 for Linux, UNIX, Windows ソースの場合)。
- **VSAM** または **VSM**。VSAM ソース (z/OS) です。

z/OS ソースの場合、PowerExchange が z/OS ソースからの変更データをすべて PowerExchange ロggerの z/OS ログファイルに記録するため、すべての z/OS 接続タイプは同じ形式のリスタートトークンになります。z/OS ソースのリスタートトークンは、PowerExchange ロggerログ内の場所を表します。

INSTANCE=*db2_database_name*

DB2 for Linux、UNIX、および Windows ソースの場合は、DB2 データベース名を指定します。【**ファイル名のオーバーライド**】 フィールドに DB2 データベース名を入力した場合を除き、このパラメータ値は必須です。SELECT CURRENT_RESTART 文の WHERE 句に、このパラメータのみを指定するか、オプションの AND キーワードを使用してこのパラメータを CONNAME または CONTYPE パラメータとともに指定します。

7. **【実行】** をクリックします。

データベース行テストによる出力ウィンドウが表示され、生成された再起動トークンが表示されます。

8. 生成されたリスタートトークンをコピーするには、**データベース行のテストによる出力ウィンドウ**で右クリックして **【出力のコピー】** をクリックします。

生成されたリスタートトークンを、PWXPC リスタートトークンファイルで使用することができます。

[データベース行のテスト] ダイアログボックス

[データベース行のテスト] ダイアログボックスで、データベース行のテストのオプションを設定できます。

ダイアログボックスには以下のオプションがあります。

DB タイプ

データソースのタイプまたはアクセス方式。抽出マップの場合、デフォルトは **CAPXRT** です。

オプションを選択し、オプションのタイプに応じた関連フィールドに情報を入力します。

オプション	説明	関連フィールド
AS4_META - このオプションは、Informatica グローバルカスタマサポートの指示の下でのみ使用します。	-	-
CAPX	PowerExchange Condense 圧縮ファイルまたは PowerExchange ロgger (Linux、UNIX、Windows 用) ログファイルの変更をバッチで読み取って、停止するアクセス方式。。	必要に応じて 【ファイル名のオーバーライド】 フィールドに、デフォルトのファイル名にオーバーライドするファイルの名前を入力します。
CAPXRT	変更ストリームから直接、あるいはリアルタイムで PowerExchange Condense の圧縮ファイルまたは PowerExchange ロgger (Linux、UNIX、Windows 用) のログファイルから変更を読み取るアクセス方式。。	必要に応じて 【ファイル名のオーバーライド】 フィールドに、デフォルトのファイル名にオーバーライドするファイルの名前を入力します。
DB2	DB2 for z/OS、DB2 for i5/OS、DB2 for Linux, UNIX, and Windows データベース	【SSID または DB 名】 フィールドに DB2 データベース名を入力します。DB2 for z/OS の場合は、サブシステム ID またはデータ共有グループ名を入力します。
DB2390IMG	リモートシステムの圧縮されていないテーブルスペースの DB2 for z/OS イメージコピー	必要に応じて 【ファイル名のオーバーライド】 フィールドに、デフォルトのファイル名にオーバーライドするファイルの名前を入力します。 【SSID】 フィールドの場合は、DB2 for z/OS のサブシステム ID を入力します。

オプション	説明	関連フィールド
DB2400C	ODBC 標準呼び出しを使用して DB2 for i5/OS データへアクセスするアクセス方式。	【SSID または DB 名】 フィールドの場合は、DB2 データベース名を入力します。
DB2UDB	DB2 for Linux, UNIX, and Windows データベース	【SSID または DB 名】 フィールドの場合は、DB2 データベース名を入力します。
IMSUNLD	IMS アンロードファイル	【ファイル名】 フィールドに、IMS アンロードファイルの名前を入力します。
MSSQL	Microsoft SQL Server データベース	【DSN】 フィールドに、SQL Server のインスタンス名を入力します。 必要に応じて 【データベース】 フィールドで、接続用に指定されたデータベース名をオーバーライドするデータベース名を入力します。
MySQL	MySQL データベースサーバー	【サーバー】 フィールドで、ソースデータベースが配置されている MySQL サーバーのサーバー名または IP アドレスを入力します。
NRDB	VSAM データセット、IMS データベース、フラットファイルなどの非リレーショナルデータソース。 注: このデータソースタイプは、生成された SQL で 3 階層式の命名規則を使用します。	必要に応じて 【ファイル名のオーバーライド】 フィールドに、データマップからのデフォルトのファイル名をオーバーライドするファイル名を入力します。 必要に応じて、【ファイルのパスワード】 フィールドにファイルのパスワードを入力します。
NRDB2	非リレーショナルデータソース 注: このデータソースタイプは、生成された SQL で 2 階層式の命名規則を使用します。	必要に応じて 【ファイル名のオーバーライド】 フィールドに、データマップからのデフォルトのファイル名をオーバーライドするファイル名を入力します。 必要に応じて、【ファイルのパスワード】 フィールドにファイルのパスワードを入力します。
ODBC	ODBC データソース	【DSN】 フィールドに、ODBC データソース名を入力します。 【データベース】 フィールドに、データベース名を入力します。
ORACLE	Oracle データベース	必要に応じて、【SQL *Net 名】 フィールドに、Oracle インスタンスの TNS 名を入力します。
PGSQL	PostgreSQL データベースサーバー	Server。データベースサーバー名。

オプション	説明	関連フィールド
Simulator	このオプションは主に Informatica 内部で使用されます。選択しないでください。	[DB 修飾子 1] フィールドは無視してください。
TASK_CTRL	PowerExchange Navigator 経由での PowerExchange リスナコマンドの発行を可能にするオプション。 詳細については、「 データベース行のテスト内の PowerExchange リスナコマンドの発行 」(ページ 234)を参照してください。	-

場所

データソースの場所のノード名を選択します。

このフィールドに、PowerExchange Navigator システムの DBMOVER 構成ファイルの NODE 文で定義した場所が表示されます。【ローカル】という値は、PowerExchange Navigator システムにソースがあることを示します。

DL/1 アクセス方式を使用する IMS データマップをテストする場合は、Netport ジョブを実行する場所のノード名を入力します。ODBA アクセス方式を使用する IMS データマップをテストする場合は、PowerExchange リスナを実行する場所のノード名を入力します。

ユーザー ID

ソースにアクセスできるユーザー ID です。ユーザー ID のタイプは、ソースタイプおよび DBMOVER 構成ファイルの SECURITY 文に指定した PowerExchange のセキュリティ設定によって異なります。

サポートされている Linux、UNIX、または Windows システムのソースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

Microsoft NTLM および Active Directory 認証を使用する Microsoft SQL Server ソースの場合は、*domain \user_name* の形式でユーザー ID を入力します。次に、【パスワード】フィールドに、このユーザー ID のパスワードを入力します。

パスワード

ユーザー ID のパスワードを入力します（必要な場合）。

ソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。
' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符（'）、二重引用符（"）、または通貨記号を含めることはできません。

z/OS で IBM IRRPHREX イグジットに指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

フェッチ

データベース行のテストでアクセスおよび表示する情報のタイプ。このオプションは、[DB タイプ] の値に応じて利用可能になります。**TASK_CTRL** タイプの場合、このフィールドには、PowerExchange リスナに発行するコマンドのタイプが表示されます。

注: パーソナルメタデータプロファイルの場合、**フェッチ**のリストは使用できず、デフォルト値は**データ**になります。

以下の表に、[フェッチ] リストのオプションを示します。

オプション	データソースタイプで使用可能	説明
カラム	TASK_CTRL 以外のすべて	カラムのメタデータを表示します。
データ	TASK_CTRL 以外のすべて	データ行を表示します。
外部キー	TASK_CTRL 以外のすべて	外部キーのメタデータを表示します。
プライマリキー	TASK_CTRL 以外のすべて	プライマリキーのメタデータを表示します。
プロシージャ列	<ul style="list-style-type: none">- AS4_META- DB2- DB2400C- ODBC- Simulator	特定のストアードプロシージャのメタデータを表示します。
プロシージャ	<ul style="list-style-type: none">- AS4_META- DB2- DB2400C- ODBC- Simulator	使用可能なストアードプロシージャのメタデータを表示します。
レコード	<ul style="list-style-type: none">- NRDB- NRDB2	NRDB および NRDB2 レコードのメタデータを表示します。

オプション	データソースタイプで使用可能	説明
スキーマ	<ul style="list-style-type: none"> - CAPX - CAPXRT - DB2 - DB2390IMG - DB2400C - DB2UDB - IMSUNLD - MSSQL - MYS - NRDB - NRDB2 - ODBC - ORACLE - PGSQL 	指定されたデータソースタイプのスキーマ名のメタデータを表示します。
テーブル	TASK_CTRL 以外のすべて	テーブルのメタデータを表示します。
場所の一覧表示	TASK_CTRL	PowerExchange リスナが実行しているシステムにある DBMOVER 構成ファイルで定義されている NODE 文についての情報を表示します。
タスクの一覧表示	TASK_CTRL	アクティブな各 PowerExchange リスナタスクに関して、TCP/IP アドレス、ポート番号、アプリケーション名、アクセスタイプ、状態などの情報を表示します。
タスクの停止	TASK_CTRL	指定したアプリケーション名に基づいて個々の PowerExchange リスナタスクを停止します。

デフォルトは **【データ】** です。データをプレビューするには、**【データ】** を使用します。

アクセス方式

【アクセス方式】 のリストはオプションを次の組み合わせで選択すると使用可能になります。

- **【DB タイプ】** の値が **【NRDB】** または **【NRDB2】** で、**【フェッチ】** の値が **【テーブル】** または **【カラム】**
- **【DB タイプ】** の値が **【CAPX】** または **【CAPXRT】** で、**【フェッチ】** の値が **【テーブル】**

特定のアクセス方式の情報を取得するには、アクセス方式を 1 つ以上選択します。PowerExchange は、**【SQL 文】** ボックスの DTLDESCRIBE 文にフィルタ用のキーワードを追加します。

アクセス方式が選択されない場合、DTLDESCRIBE 文はすべてのアクセス方式の情報を返します。

以下の表に、【アクセス方式】リストのオプションを示します。

アクセス方式	説明	DTLDESCRIBE キーワード
DM_ADABAS	Adabas	A
DM_CAPX	- CAPX - CAPXRT	C
DM_DATACOM	Datacom	X
DM_DB2	- DB2 for i5/OS - DB2 for z/OS	Z
DM_DB2390IMG	DB2390IMG	G
DM_DB2UNLD	DB2 アンロードファイル	W
DM_DL1	IMS DL/1 バッチ	D
DM_ESDS	VSAM ESDS	E
DM_IDMS	IDMS	I
DM_KSDS	VSAM KSDS	K
DM_MQ	MQSeries	Q
DM_ODBA	IMS ODBA	O
DM_RRDS	VSAM RRDS	N
DM_SEQ	シーケンシャルデータセットまたはフラットファイル	S
DM_TAPE	テープ	T
DM_USER	USER アクセス方式	U
DM_XDB2	DB2 for i5/OS または DB2 for z/OS 抽出マップ。データソースタイプが CAPX または CAPXRT の場合にのみ表示されます。	B
DM_XDB2UDB	DB2 for Linux, UNIX, and Windows 抽出マップ。DB タイプが CAPX または CAPXRT の場合にのみ表示されます。	V
DM_XMSSQL	Microsoft SQL Server 抽出マップ。DB タイプが CAPX または CAPXRT の場合にのみ表示されます。	L
DM_XORACLE	Oracle 抽出マップ。DB タイプが CAPX または CAPXRT の場合にのみ表示されます。	P

ヒント: 【アクセス方式】リストで選択をすべて取り消すには、【アクセス方式】リストを右クリックして【すべて選択解除】をクリックします。

SQL ファイル

【SQL 文】 ボックスで生成された SQL を使用する代わりに、SQL 文を含むファイルを使用します。このオプションを選択する場合は、**【SQL ファイル】** ボックスに SQL ファイルのパスおよびファイル名を入力するか、**【参照】** ボタンをクリックしてファイルを参照します。

デフォルトでは選択されていません。

編集

【SQL ファイル】 オプションが選択されている場合に有効になります。

【編集】 をクリックし、**【SQL ファイル】** ボックスで指定した SQL ファイルを編集します。

SQL レコード長

【SQL ファイル】 オプションが選択されている場合に有効になります。

SQL ファイル内にある SQL のレコード長です。

SQL 文

指定されたデータソースタイプおよび**【フェッチ】**オプションの生成された SQL、DTLDESCRIBE SQL、PowerExchange リスナコマンド文を表示します。

【SQL 文】 ボックスの文を編集することができます。

注: **【SQL ファイル】** オプションを選択した場合、**【SQL 文】** ボックスは無効になります。

PowerExchange は、以下のデータソースタイプと**【フェッチ】**オプションの組み合わせのため、**【SQL 文】** ボックスに、以下の文を生成します。

- **TASK_CTRL** 以外のすべてのデータソース、および**【フェッチ】**リストの**【データ】**
SQL 文が生成され、**【SQL 文】** ボックスに表示されます。PowerExchange は、2 階層の命名規則を使用するデータマップ用の以下の SELECT 文を生成します。

```
select * from schema.map_table
```


PowerExchange は、3 階層の命名規則を使用するデータマップ用の以下の SELECT 文を生成します。

```
select * from schema.map.table
```
- **TASK_CTRL** 以外のすべてのデータソースタイプ、および**【フェッチ】**リストのメタデータオプション
DTLDESCRIBE 文が生成され、**【SQL 文】** ボックスに表示されます。メタデータのオプションには、**【カラム】**、**【外部キー】**、**【プライマリキー】**、**【プロシージャ列】**、**【手順】**、**【スキーマ】**、**【テーブル】** が含まれます。
- **TASK_CTRL** データソースタイプおよび、**フェッチ**リスト内の PowerExchange リスナコマンド
PowerExchange リスナコマンドが生成され、**【SQL 文】** ボックスに表示されます。
注: **AS4_META** データソースタイプは、Informatica グローバルカスタマサポートからの指示があった場合に使用します。

メタデータのフィルタ

【DB タイプ】 オプションと**【フェッチ】**オプションの選択に応じて、任意に**【メタデータ】** フィールドにフィルタ条件を入力します。

注: **【フェッチ】** リストで**【データ】**を選択した場合、いずれのフィルタフィールドも使用できません。

どのフィルタフィールドでも、次のワイルドカード文字が使用できます。

- アスタリスク (*) は、1 つ以上の一致する文字を表します。
- 疑問符 (?) は、単一の一致する文字を表します。

以下の表に、[DB タイプ] と [フェッチ] オプションのさまざまな組み合わせに対して使用可能になるフィルタフィールドを示します。

DB タイプ	フェッチオプション	フィルタフィールド
AS4_META、シミュレータ	カラム、外部キー、プライマリキー、テーブル	- テーブル
CAPX、CAPXRT、DB2390IMG、IMSUNLD、NRDB2、MYSQL、Oracle、PGSQL	カラム、外部キー、プライマリキー、テーブル	- スキーマ - テーブル
CAPX、CAPXRT、DB2390IMG、IMSUNLD、NRDB2、MYSQL、Oracle、PGSQL	スキーマ	- スキーマ
DB2、DB2400C、DB2UDB	カラム、外部キー、プライマリキー、テーブル	- クリエータ/スキーマ DB2 for i5/OS の場合、最大長は 10 文字です。 DB2 for z/OS の場合、最大長は 128 文字です。 - DB 名/定義元 - テーブル DB2 for i5/OS の場合、最大長は 32 文字です。 DB2 for z/OS の場合、最大長は 128 文字です。
DB2、DB2400C、DB2UDB	スキーマ	- クリエータ/スキーマ
DB2、DB2400C、ODBC	プロシージャ呼び出し、プロシージャ	- カタログ - スキーマ - プロシージャ名
MSSQL、ODBC	カラム、外部キー、プライマリキー、テーブル	- 所有者 - テーブル それぞれのフィールドの最大長は 128 文字です。
MSSQL、ODBC	スキーマ	- 所有者
NRDB	カラム、外部キー、プライマリキー、レコード、テーブル	- スキーマ 最大長は 10 文字です。 - マップ名 PowerExchange データマップの名前。最大長は 10 文字です。 - テーブル 最大長は 128 文字です。
NRDB	スキーマ	- スキーマ

また、**[メタデータ]** グループボックスにおいて、以下のチェックボックスの中の有効なものを任意に選択します。

- **大文字と小文字を区別する。**デフォルトでは、フィルタフィールドに入力されたテキストはデータソースの標準文字に変換されます。PowerExchange によってテキストをデータソースの標準文字に変換しない場合は、該当するフィルタフィールドのこのチェックボックスをオンにします。例えば、Oracle ソースの場合、大文字が標準です。このチェックボックスをオフにした場合、scott と SCOTT の両方が SCOTT に変換されます。一方、このチェックボックスをオンにした場合、scott のエントリは小文字のまま維持されます。
- **コメント。**データベースオブジェクトの行テストによってフェッチされるメタデータのすべてのコメントを取得するには、このチェックボックスをオンにします。
- **拡張。**拡張メタデータが返されるようにするには、このチェックボックスをオンにします（選択されたデータソースタイプで適切な場合）。

詳細

CAPX、**CAPXRT**、**DB2390IMG**、**IMSUNLD**、**IMS NRDB**、および **NRDB2** データソースタイプで使用できます。

[詳細] をクリックして **[詳細パラメータ]** ダイアログボックスを表示します。このダイアログボックスで、PowerExchange 構成ファイル、データマップ、および抽出マップの特定の設定をオーバーライドするパラメータ値を入力できます。

アプリケーション

CAPX および **CAPXRT** のデータソースで必要となります。

アプリケーション名。1～20 文字の文字列を入力します。

エスケープ文字

使用するエスケープ文字を入力して、アスタリスク (*) または疑問符 (?) を区切ります。エスケープ文字は、**[メタデータ]** フィルタボックスでワイルドカードではなくリテラル値として使用されます。

例えば、~のエスケープ文字を指定する場合、tab*のフィルタ値は、tab で始まるすべてのテーブルを返します。tab~*のフィルタ値は、tab*と名付けられたテーブルを返します。

注: マルチバイトが有効化されたデータソースではエスケープ文字を使用しないでください。

n 行を取得

フェッチする行数を指定します。

有効な値は 1～99999 です。デフォルトは 10 です。

注: 大きい値を指定すると、PowerExchange Navigator の応答時間と CPU 使用率が増加する可能性があります。

先頭の n 行をスキップする

行のテストで表示するデータをフェッチする際に、スキップする先頭行数のデータを指定します。大きなファイルで行のテストを実行する場合は、このオプションを使用して、行のテストでファイル内の特定ポイントに進んでから、データを表示することができます。関連する可能性のあるデータのみが分析用に返されるため、このオプションは大きいファイルで問題を迅速に診断するのに役立ちます。

このオプションは、**[DB タイプ]** リストで **[CAPX]**、**[CAPXRT]**、**[NRDB]**、**[NRDB2]**、または **[IMSUNLD]** を選択し、**[フェッチ]** リストで **[データ]** を選択した場合のみ使用できます。

有効な値は 0～99999 です。デフォルトは 0 で、行はスキップされません。

大きい値を指定すると、PowerExchange Navigator の応答時間と CPU 使用率が増加する可能性があります。

制限: **[可変]** プロパティが **[VS]** に設定され、可変長ストリームデータファイルであることを示す SEQ データマップをテストすると、データベース行のテストで警告メッセージ PWX-03042 が返され、**[先頭の *n* 行をスキップする]** オプションは無視されます。PowerExchange Navigator では、可変長データファイルの行はスキップできません。

CAPX の [詳細パラメータ] ダイアログボックス - [全般] タブ

CAPX の [詳細パラメータ] ダイアログボックスの **[全般]** タブでは、変更データへのバッチアクセスのデータベース行のテストに対して、オプションのオーバーライドを定義します。

抽出の種類

抽出タイプです。次のいずれかのオプションを選択します。

- **RS.** 前回の抽出の開始ポイントから抽出を再開します。このオプションにより、同じ抽出を何度も実行することができます。
- **SL.** 前回の抽出の終了ポイントから抽出を開始します。この抽出では、z/OS のリスタートトークンまたは i5/OS のチェックポイントタイムスタンプが使用されます。
- **CC.** このタブのトークン値 [リスタートトークン 1] フィールドおよび [リスタートトークン 2] フィールドによって示されるポイントから抽出を開始します。このオプションを選択した場合は、有効なトークン値を指定する必要があります。

z/OS ソースの場合、LRAP CAPI_CONNECTION 文の FUZZYRSTART パラメータに Y のデフォルト値を使用しているときは、z/OS 用 PowerExchange ロガーのログで抽出処理が開始された開始 UOW 位置とは異なる **【リスタートトークン 2】** の位置を入力できます。次の例では、**【リスタートトークン 2】** の値が開始 UOW レコードと一致しない RBA の位置を指しており、**【リスタートトークン 1】** の値は **【リスタートトークン 2】** の位置より前の位置を指しています。

- **[リスタート 1]** フィールドで、00000000000010000000000000000000000000000000000 を指定します。この値は 48 個のゼロで構成され、12 番目の位置に「1」があります。この値を使用して、リスタートトークン 2 の値は行テストを返すレコードを判別します。

- **[リスタート2]** フィールドで、EBCDIC フォーマット（スペース埋め）の 6 バイトのロッガー ID、6 バイトの RBA 位置、8 個のゼロで構成される値を指定します。RBA 位置は、ログ内の抽出処理を再開したい位置の近くの任意の RBA にすることができます。例: E2C2F2D340400000000AEF4000000000

PowerExchange は、これらのリスタートトークン設定を使用して、リスタート 2 位置に基づいて返された最初の UOW で、リスタート 1 位置以降の最初のレコードから抽出処理を開始します。

デフォルトは **RS** です。

イメージタイプ

キャプチャイメージのタイプです。次のいずれかのオプションを選択します。

- **AI。**操作後の画像データをキャプチャします。
- **BA。**最新の変更の前に存在していた操作前の画像データをキャプチャします。また、最新の変更の後に存在する操作後の画像データをキャプチャします。

操作前の画像データには、レコードの削除を強制するアクション文字が付属しています。操作後の画像データは、変更データを含み、挿入操作として提供されます。

- **TU**. 操作前/操作後の画像データをキャプチャします（発生時）。アプリケーションは、操作前の画像データを使用して、要求以降にレコード内で何も変更されていないことを確認します。操作後の画像データは、更新操作として表示されます。更新はフルイメージですが、BAを使用する場合と同様、この更新をキャプチャするアプリケーションは、ターゲットレコードを削除して新しく挿入する代わりにターゲットレコードを変更することができます。TU オプションは ODBC 接続に対して使用します。

デフォルトは **AI** です。

タイムアウト

EOF（ファイルの終わり）を返すまでの、データを待機する最長時間（秒単位）。

有効な値は次のとおりです。

- 0. 現在のログの終端に達すると EOF を返します。ログの終端は、（実際の終端は絶えず移動するので）抽出処理の開始で判別されます。
- 2～86399. データを指定された秒数待ってから、EOF を返します。
- 86400. EOF を返すことなく、無制限に待機を続けます。

デフォルトは 15 です。

進捗状況の更新はありません

データベース行のテストによる出力をアプリケーション用の PowerExchange CDC 監査証跡に追加するかどうかを示します。

- データベース行のテストによる出力を PowerExchange CDC 監査証跡から省略するには、このチェックボックスをオンにします。
- データベース行のテストによる出力を PowerExchange CDC 監査証跡に含めるには、このチェックボックスをオフにします。

デフォルトでは、このチェックボックスはオンです。

リスタートトークン 1

このフィールドは、**【抽出の種類】** フィールドで **【CC】** を選択した場合にのみ使用できます。データベース行のテストで、変更ストリーム内の特定のポイントからデータの取得を開始する場合は、このフィールドに有効なシーケンストークン値を入力します。**【リスタートトークン 2】** フィールドのリスタートトークン値とシーケンストークン値の組み合わせにより、データベース行のテストでデータの取得を開始する変更ストリーム内の特定の更新レコードを指定します。これらのフィールドは、ディスクからアーカイブ済みの古いレコードを省略する場合や、更新した抽出マップと一貫性のないレコードを省略する場合に使用します。

シーケンストークンはバイナリ値であり、その長さはデータソースタイプによって変わります。シーケンストークンとリスタートトークンは、次の方法で変更ストリームの現在の終了位置に生成できます。

- PowerExchange Navigator で、CAPXRT データベース行のテストを実行し、**【SQL 文】** ボックスに **SELECT CURRENT_RESTART** 文を指定します。**【データベース行のテストによる出力】** ウィンドウに生成されたトークンの値が表示されます。
- DTLUAPPL ユーティリティを使用してシーケンストークンとリスタートトークンの値を生成します。ADD 文または MOD APPL 文に、RSTTKEN GENERATE パラメータを含めます。次に PRINT APPL 文を使用して、生成されたトークン値を出力します。

注: 生成されるトークンは ODBC 抽出に使用されている形式で出力されます。データベース行のテストにトークン値を使用するには、値の末尾に 8 つのゼロを追加します。

また、PWXPC CDC セッションを最近実行した場合は、次のメッセージからトークン値をコピーできます。

- PowerExchange メッセージログで、PWX-04565 および PWX-09959 メッセージの **【シーケンス】** フィールドにシーケンストークンおよび **【PowerExchange Logger】** フィールドにリスタートトークンが表示されます。
- PowerCenter ワークフローセッションログで、PWXPC_12060 および PWXPC_12068 メッセージの **【リスタートトークン 1】** フィールドにシーケンストークンおよび **【リスタートトークン 2】** フィールドにリスタートトークンが表示されます。また、PWXPC_10081、PWXPC_10082、および PWXPC_12128 メッセージの最初のトークン値にシーケンストークンおよび 2 番目のトークン値にリスタートトークンが表示されます。

有効なトークン値の作成についてサポートが必要な場合は、Informatica グローバルカスタマサポートにお問い合わせください。

リスタートトークン 2

このフィールドは、**【抽出の種類】** フィールドで **【CC】** を選択した場合にのみ使用できます。データベース行のテストで、変更ストリームの特定のポイントからデータの取得を開始する場合は、このフィールドに有効なリスタートトークン値を入力します。**【リスタートトークン 1】** フィールドのシーケンストークン値とリスタートトークン値の組み合わせにより、データベース行のテストでデータの取得を開始する変更ストリームの特定の更新レコードを指定します。これらのフィールドは、ディスクからアーカイブ済みの古いレコードを省略する場合や、更新した抽出マップと一貫性のないレコードを省略する場合に使用します。

リスタートトークンはバイナリ値であり、多くの場合その長さはデータソースタイプによって変わります。z/OS では、完全圧縮ファイルから抽出された更新データを除き、すべてのデータソースタイプのリスタートトークンが同じ長さです。変更ストリームの現在の終了位置を示すリスタートトークンとシーケンストークンの値は、PowerExchange Navigator または DTLUAPPL ユーティリティで生成できます。また、PWXPC CDC セッションを最近実行している場合は、セッションログや PowerExchange メッセージログのメッセージからトークン値をコピーできます。有効なトークン値の作成についてサポートが必要な場合は、Informatica グローバルカスタマサポートにお問い合わせください。

抽出スキーマ

抽出マップで指定されたスキーマをオーバーライドします。

【OK】 をクリックして変更を保存します。ソースデータベース接続が閉じます。**【データベース行のテスト】** ダイアログボックスで、**【実行】** をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

CAPX の **【詳細パラメータ】** ダイアログボックス - **【AS400】** タブ

CAPX の **【詳細パラメータ】 ダイアログボックスの **【AS400】** タブ**では、DB2 for i5/OS ソースデータへのバッチアクセスのデータベース行のテストに対して、オプションのオーバーライドパラメータを定義します。

AS400 インスタンス

テストする抽出マップのインスタンス名またはコレクション ID をオーバーライドします。このオーバーライドフィールドを使用するのは、次のような場合です。

- 異なる AS400 オブジェクト（異なるジャーナルやスキーマなど）を使用する同一のテーブルがある場合。

- [OK] をクリックして変更を保存します。ソースデータベース接続が閉じます。[データベース行のテスト] ダイアログボックスで、[実行] をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

CAPXRT の [詳細パラメータ] ダイアログボックス - [全般] タブ 249

操作前の画像データには、レコードの削除を強制するアクション文字が付属しています。操作後の画像データは、変更データを含み、挿入操作として提供されます。

- **TU**。操作前/操作後の画像データをキャプチャします（発生時）。アプリケーションは、操作前の画像データを使用して、要求以降にレコード内で何も変更されていないことを確認します。操作後の画像データは、更新操作として表示されます。更新はフルイメージですが、BAを使用する場合と同様、この更新をキャプチャするアプリケーションは、ターゲットレコードを削除して新しく挿入する代わりにターゲットレコードを変更することができます。TU オプションは ODBC 接続に対して使用します。

デフォルトは **AI** です。

タイムアウト

EOF（ファイルの終わり）を返すまでの、データを待機する最長時間（秒単位）。

有効な値は次のとおりです。

- 0。現在のログの終端に達すると EOF を返します。ログの終端は、（実際の終端は絶えず移動するので）抽出処理の開始で判別されます。
- 2～86399。データを指定された秒数待ってから、EOF を返します。
- 86400。EOF を返すことなく、無制限に待機を続けます。

デフォルトは 15 です。

進捗状況の更新はありません

データベース行のテストによる出力をアプリケーション用の PowerExchange CDC 監査証跡に追加するかどうかを示します。

データベース行のテストによる出力を PowerExchange CDC 監査証跡から省略するには、このチェックボックスをオンにします。

データベース行のテストによる出力を PowerExchange CDC 監査証跡に含めるには、このチェックボックスをオフにします。

デフォルトでは、このチェックボックスはオンです。

リスタートトークン 1

このフィールドは、**【抽出の種類】** フィールドで **【CC】** を選択した場合にのみ使用できます。データベース行のテストで、変更ストリーム内の特定のポイントからデータの取得を開始する場合は、このフィールドに有効なシーケンストークン値を入力します。**【リスタートトークン 2】** フィールドのリスタートトークン値とシーケンストークン値の組み合わせにより、データベース行のテストでデータの取得を開始する変更ストリームの特定の更新レコードを指定します。これらのフィールドは、ディスクからアーカイブ済みの古いレコードを省略する場合や、更新した抽出マップと一貫性のないレコードを省略する場合に使用します。

シーケンストークンはバイナリ値であり、その長さはデータソースタイプによって変わります。シーケンストークンとリスタートトークンは、次の方法で変更ストリームの現在の終了位置に生成できます。

- PowerExchange Navigator で、CAPXRT データベース行のテストを実行し、**【SQL 文】** ボックスに **SELECT CURRENT_RESTART** 文を指定します。**【データベース行のテストによる出力】** ウィンドウに生成されたトークンの値が表示されます。
- DTLUAPPL ユーティリティを使用してシーケンストークンとリスタートトークンの値を生成します。ADD 文または MOD APPL 文に、RSTTKEN GENERATE パラメータを含めます。次に PRINT APPL 文を使用して、生成されたトークン値を出力します。

注: 生成されるトークンは ODBC 抽出に使用されている形式で出力されます。データベース行のテストにトークン値を使用するには、値の末尾に 8 つのゼロを追加します。

また、PWXPC CDC セッションを最近実行した場合は、次のメッセージからトークン値をコピーできます。

- PowerExchange メッセージログで、PWX-04565 および PWX-09959 メッセージの **【シーケンス】** フィールドにシーケンストークンおよび **【PowerExchange Logger】** フィールドにリスタートトークンが表示されます。
- PowerCenter ワークフローセッションログで、PWXPC_12060 および PWXPC_12068 メッセージの **【リスタートトークン 1】** フィールドにシーケンストークンおよび **【リスタートトークン 2】** フィールドにリスタートトークンが表示されます。また、PWXPC_10081、PWXPC_10082、および PWXPC_12128 メッセージの最初のトークン値にシーケンストークンおよび 2 番目のトークン値にリスタートトークンが表示されます。

有効なトークン値の作成についてサポートが必要な場合は、Informatica グローバルカスタマサポートにお問い合わせください。

リスタートトークン 2

このフィールドは、**【抽出の種類】** フィールドで **【CC】** を選択した場合にのみ使用できます。データベース行のテストで、変更ストリームの特定のポイントからデータの取得を開始する場合は、このフィールドに有効なリスタートトークン値を入力します。**【リスタートトークン 1】** フィールドのシーケンストークン値とリスタートトークン値の組み合わせにより、データベース行のテストでデータの取得を開始する変更ストリームの特定の更新レコードを指定します。これらのフィールドは、ディスクからアーカイブ済みの古いレコードを省略する場合や、更新した抽出マップと一貫性のないレコードを省略する場合に使用します。

リスタートトークンはバイナリ値であり、多くの場合その長さはデータソースタイプによって変わります。z/OS では、完全圧縮ファイルから抽出された変更データを除き、すべてのデータソースタイプのリスタートトークンが同じ長さです。変更ストリームの現在の終了位置を示すリスタートトークンとシーケンストークンの値は、PowerExchange Navigator または DTLUAPPL ユーティリティで生成できます。また、PWXPC CDC セッションを最近実行している場合は、セッションログや PowerExchange メッセージログのメッセージからトークン値をコピーできます。有効なトークン値の作成についてサポートが必要な場合は、Informatica グローバルカスタマサポートにお問い合わせください。

抽出スキーマ

抽出マップで指定されたスキーマをオーバーライドします。

CAPI 接続名

DBMOVER 構成ファイルの CAPI_CONNECTION 文で定義したデフォルトのデータベース接続をオーバーライドします。

関連するデータソースを参照するには、DBMOVER 構成ファイルにある CAPI_CONNECTION 文の NAME パラメータで定義した値を使用します。

DBMOVER 構成ファイルに複数の CAPI 接続文が含まれるときに必要です。

抽出マップ - 場所

データソースの抽出マップからのリモートである PowerExchange ロgger（Linux、UNIX、Windows 用）から継続抽出モードで変更データを抽出している場合は、抽出マップが存在するシステムのノード名を入力します。

PowerExchange ロgger（Linux、UNIX、Windows 用）が実行している dbmover.cfg ファイル内の NODE 文で、このノードを定義する必要があります。このノード名は通常、pwxccl.cfg コンフィギュレーションファイル内の CAPTURE_NODE パラメータで定義したノード名です。

例えば、PowerExchange ロgger（Linux、UNIX、Windows 用）のプロセスが z/OS システムのリモート IMS ソースから変更データをキャプチャする場合は、z/OS システムのノード名を指定します。

抽出マップ - ユーザー ID

PowerExchange ロgger (Linux、UNIX、Windows 用) から **【場所】** リストで選択したリモートノードの抽出マップにアクセスする際に、認証に使用するユーザー ID です。

このユーザー ID が必要かどうかは、リモート場所での DBMOVER 構成ファイルの SECURITY 設定によります。このユーザー ID は、データソースタイプによって、オペレーティングシステムユーザー ID またはデータベース ID のいずれかとなります。

このユーザー ID は通常、pwxcl.cfg コンフィギュレーションファイル内の CAPTURE_NODE_UID パラメータで定義したユーザー ID です。

サポートされている Linux、UNIX、または Windows システムのデータソースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効 (該当する場合) にしている場合、ユーザー ID はエンタープライズユーザー名になります。詳細については、『*PowerExchange* リファレンスマニュアル』を参照してください。

抽出マップ - パスワード

【UserID】 ボックスで指定したユーザー ID のクリアテキストパスワードです。この値は通常、pwxcl.cfg 構成ファイルの CAPTURE_NODE_PWD パラメータ値と一致します。

PowerExchange ロgger を使用してリモートの i5/OS または z/OS のノードにある抽出マップにアクセスする場合は、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9~31 文字の範囲で指定できます。z/OS のパスフレーズは、9~128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。

- 英字 (大文字と小文字)
- 数字 (0~9)
- スペース
- 次に示す特殊文字。

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符 (')、二重引用符 (")、または通貨記号を含めることはできません。

z/OS で IBM IRRPHREX イグジットに指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange* リファレンスマニュアル』の「SECURITY 文」を参照してください。

【OK】 をクリックして変更を保存します。ソースデータベース接続が閉じます。**【データベース行のテスト】** ダイアログボックスで、**【実行】** をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

CAPXRT の [詳細パラメータ] ダイアログボックス - [Adabas] タブ

CAPXRT の [詳細パラメータ] ダイアログボックスの **[Adabas]** タブでは、Adabas FDT がパスワードで保護されている場合に、Adabas ソースデータへのリアルタイムアクセスのデータベース行のテストに対して、パラメータを定義する必要があります。

ADABAS ファイルのパスワード

Adabas FDT のパスワードです。

Adabas FDT が Adabas でパスワード保護されている場合に必要です。

[OK] をクリックして変更を保存します。ソースデータベース接続が閉じます。**[データベース行のテスト]** ダイアログボックスで、**[実行]** をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

CAPXRT の [詳細パラメータ] ダイアログボックス - [AS400] タブ

CAPXRT の [詳細パラメータ] ダイアログボックスの **[AS400]** タブでは、DB2 for i5/OS ソースデータへのリアルタイムアクセスのデータベース行のテストに対して、オプションのオーバーライドパラメータを定義します。

AS400 ライブラリ/ジャーナル

i5/OS システムの DBMOVER 構成ファイル内で、AS4J CAPI_CONNECTION 文の JOURNAL パラメータに指定されたライブラリ名とジャーナル名をオーバーライドします。以下に例を示します。

STQA/NEWJOURNAL

[OK] をクリックして変更を保存します。ソースデータベース接続が閉じます。**[データベース行のテスト]** ダイアログボックスで、**[実行]** をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

CAPXRT の [詳細パラメータ] ダイアログボックス - [DB2UDB] タブ

CAPXRT の [詳細パラメータ] ダイアログボックスの **[DB2UDB]** タブでは、DB2 for Linux, UNIX, and Windows ソースデータへのリアルタイムアクセスのデータベース行のテストに対して、オプションのオーバーライドパラメータを定義します。

DB2 UDB データベース

PowerExchange 抽出マップの DB2 for Linux, UNIX, and Windows データベースの名前をオーバーライドします。

[OK] をクリックして変更を保存します。ソースデータベース接続が閉じます。**[データベース行のテスト]** ダイアログボックスで、**[実行]** をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

CAPXRT の [詳細パラメータ] ダイアログボックス - [MSSQL] タブ

CAPXRT の [詳細パラメータ] ダイアログボックスの **[MSSQL]** タブでは、PowerExchange ロgger (Linux、UNIX、Windows 用) を継続モードで使用し、複数のパブリケーションデータベース内の SQL Server 記事のデータをキャプチャする場合に、Microsoft SQL Server ソースデータへのリアルタイムアクセスのデータベース行のテストに対して、パラメータを定義する必要があります。

MSS LUW の DBID

PowerExchange ロgger (Linux、UNIX、Windows 用) を継続抽出モードで使用し、複数のパブリケーションデータベース内のアーティクルのデータを Microsoft SQL Server 分散データベースからキャプチャする場合は、PowerExchange ロgger の構成ファイル `pwxcl.cfg` で指定されている DBID パラメータ値を入力します。この DBID 値により、行テスト処理で使用する登録のコレクションが特定されます。この値を指定しなかった場合、行テストは失敗します。

[OK] をクリックして変更を保存します。ソースデータベース接続が閉じます。**[データベース行のテスト]** ダイアログボックスで、**[実行]** をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

[CAPXRT 詳細パラメータ] ダイアログボックス - [Oracle] タブ

[CAPXRT 詳細パラメータ] ダイアログボックスの **[Oracle]** タブでは、Oracle ソースデータへのリアルタイムアクセスのデータベース行のテストに対して、オプションのオーバーライドを定義します。

Oracle インスタンス

`dbmover.cfg` 構成ファイル内の ORACLEID 文にある 2 番目の位置パラメータ `oracle_sid` の Oracle インスタンスをオーバーライドします。

```
ORACLEID=(coll_id,oracle_sid,connect_string,cap_connect_string)
```

このパラメータと **Oracle 接続文字列** パラメータを組み合わせることで、単一セットのキャプチャ登録を使用して、複数の Oracle インスタンスからデータをキャプチャすることができます。

Oracle インスタンス の値は指定しているが、**Oracle 接続文字列** の値は指定しない場合、PowerExchange では、`dbmover.cfg` ファイルの ORACLEID 文で定義した Oracle 接続文字列値が使用されます。

Oracle 接続文字列

`dbmover.cfg` 構成ファイル内の ORACLEID 文の 3 番目の位置パラメータ `connect_string` の Oracle 接続文字列値をオーバーライドします。

このパラメータと **Oracle インスタンス** パラメータを組み合わせることで、単一セットのキャプチャ登録を使用して、複数の Oracle インスタンスからデータをキャプチャすることができます。

Oracle 接続文字列 の値は指定しているが、**Oracle インスタンス** の値は指定していない場合、PowerExchange は、`dbmover.cfg` ファイルの ORACLEID 文で定義した Oracle 接続文字列値が使用されます。

Oracle スキーマ

登録グループに指定された Oracle スキーマをオーバーライドして、単一セットのキャプチャ登録の使用を有効にし、Oracle インスタンスに存在する可能性のある複数のスキーマからデータをキャプチャします。

[OK] をクリックして変更を保存します。ソースデータベース接続が閉じます。[データベース行のテスト] ダイアログボックスで、[実行] をクリックし、データベース接続を再度確立して、更新した詳細プロパティで、データベース行のテストを実行します。

DB2390IMG の [詳細パラメータ] ダイアログボックス

DB2390IMG アクセス方式では、データマップが処理の一部として動的に作成されます。

データマップのスキーマ名とデータマップ名を入力します。PowerExchange は、このスキーマ名とマップ名からデータマップ名を構築します。

以下に例を示します。

`schema_name.data_map_name.dmp`

DB2390IMG 詳細パラメータを定義する情報をデータベース行のテストに入力します。

スキーマ名

データマップのスキーマ名です。

アルファベット文字で開始する必要があります。

最大長は 10 バイトです。

データマップ名

データマップのマップ名です。

アルファベット文字で開始する必要があります。

最大長は 10 バイトです。

[詳細パラメータ] ダイアログボックス (IMS)

DL/1 BATCH または IMS ODBA のアクセス方式を使用する IMS データマップ、または IMS アンロードファイルソースにアクセスする IMS データマップのデータベース行テストのオーバーライド値を入力します。アクセス方式および関連プロパティはオーバーライドできます。

アクセス方式

データマップに指定したアクセス方式をオーバーライドするには、次のオプションのいずれかを選択します。

- **DL/1 BATCH。** データマップに IMS ODBA アクセス方式を指定する場合はこのオプションを選択し、DL/I バッチまたは BMP を使用して IMS データにアクセスし、データベース行のテストを行います。また、[データベース行のテスト] ダイアログボックスで、Netport ジョブを実行するノードを示す [場所] の値を設定します。

- **IMS ODBA**。データマップに DL/I BATCH アクセス方式を指定する場合はこのオプションを選択し、IMS ODBA を使用して IMS データにアクセスし、データベース行のテストを行います。また、**【データベース行のテスト】** ダイアログボックスで、PowerExchange リスナを実行するノードを示す **【場所】** の値を設定します。

デフォルトでは、データマップのアクセス方式が最初に表示されます。デフォルトの設定をオーバーライドするには、リストに表示されている他の使用可能なアクセス方式を選択します。

リストで空白行を選択すると、データベース行のテストにデータマップのアクセス方式が使用されます。さらに、選択したアクセス方式に有効な他のオーバーライド値をこのダイアログボックスに入力できます。

【場所】 の値を変更せずにアクセス方式を変更すると、PowerExchange は次のいずれかのエラーメッセージを表示します。

- DL/I アクセスの場合:
 PWX-02200 Incorrect environment for DL1 execution.
 Check NETPORT for server job or executable for client.
- ODBA アクセスの場合:
 PWX-02220 ODBA "ALLOCATE" Call Failed RC=00000104 Reason=00001110.

重要: アクセス方式を切り替えるには、PowerExchange で新しいアクセス方式を使用するように設定している必要があります。詳細については、[「DL/I バッチまたは BMP を使用した IMS データへのアクセスの構成」 \(ページ 124\)](#) および [「IMS ODBA を使用した IMS データへのアクセスの構成」 \(ページ 125\)](#) を参照してください。

IMS SSID

IMS ODBA データマップをテストする場合にアクセス方式をオーバーライドしていないと、この値を使用してデータマップの IMS サブシステム ID (SSID) をオーバーライドします。 **【アクセス方式】** フィールドのオーバーライド値に IMS ODBA アクセスを指定した場合は、この値を入力する必要があります。ODBA アクセスには SSID が必須です。

IMS アンロードファイルのソースを指定したデータマップをテストする場合は、このオーバーライド値を使用して、DBD ライブラリを別の DBD ライブラリに変更するために DBMOVER メンバの別の IMSID 文を参照します。例えば、DL/I データマップがある場合に、IMS アンロードファイルに行テストを実行してデータベースに対するクエリを減らすには、**IMSUNLD** のデータベースタイプ、アンロードファイルの場所、アンロードファイルの名前を **【データベース行のテスト】** ダイアログボックスに入力します。次に、IMS アンロードファイルで使う DBD ライブラリを含む IMSID 文に関連付けられた IMS SSID をこのフィールドに入力します。

NRDB データベースタイプのデータマップをテストする場合や BMP Netport ジョブを使用する場合は、このオーバーライドに Netport JCL の %IMSID 置換変数を使用できます。このオーバーライドでは、行テストに使用する IMS SSID を指定する置換変数を置き換えます。オーバーライドに置換変数を使用すると、同一の Netport JCL を使用して複数の IMS 環境（開発環境、テスト環境、プロダクション環境など）にアクセスできます。

注: DL/I バッチを使用して IMS データにアクセスしたり、IMS アンロードファイルにアクセスしたりする場合は、IMS SSID は必要はありません。

PSB 名

IMS PSB 名（オーバーライド値）を入力します。

ODBA データマップをテストする場合にアクセス方式をオーバーライドしていないと、この値でデータマップの PSB 名をオーバーライドします。 **【アクセス方式】** フィールドでオーバーライド値に **IMS ODBA** を指定した場合は、この値を入力する必要があります。ODBA アクセスには PSB 名が必須です。このオーバーライドを使用すると、同一のデータマップを使用して異なる PSB へのアクセスをテストできます。

DL/I バッチまたは BMP データマップをテストする場合に Netport JCL の PSB=%PSBNAME 置換変数を指定していると、この値は JCL の PSB 名をオーバーライドします。Netport JCL の %PSBNAME ではなく PSB=%1 置換変数を指定した場合は、行テストに NETPORT 文の PSB 名を使用します（指定した場合）。Informatica は、NETPORT 文の編集や追加を避けるためにこのオーバーライドに %PSBNAME 置換変数を使用し、PowerExchange リスナを再起動することをお勧めします。

注: IMS アンロードファイル（IMSUNLD データベースタイプ）にアクセスする際に PSB 名は使用しません。

PCB 番号

PCB 番号（オーバーライド値）を入力します。このフィールドは、DL/I または BMP を使用して IMS データにアクセスする場合にのみ指定できます。

DL/I バッチまたは BMP のデータマップをテストする場合にアクセス方式をオーバーライドしていないと、この値でデータマップの PCB 番号をオーバーライドします。【アクセス方式】フィールドでオーバーライド値に **DL/1 BATCH** を指定した場合は、この値を入力する必要があります。DL/I アクセスまたは BMP アクセスには PCB 番号が必須です。

注: ODBA アクセスまたは IMS アンロードファイル（IMSUNLD データベースタイプ）へのアクセスに PCB 番号は使用しません。

PCB 名

IMS PCB 名（オーバーライド値）を入力します。PCB 名は ODBA を使用して IMS データにアクセスする際にのみ使用できます。

ソースの IMS ODBA データマップをテストする場合にアクセス方式をオーバーライドしていないと、この値でデータマップの PCB 名をオーバーライドします。【アクセス方式】フィールドでオーバーライド値に **IMS ODBA** を指定した場合は、この値を入力する必要があります。ODBA アクセスには PCB 名が必須です。

注: DL/I バッチアクセス、BMP アクセス、または IMS アンロードファイル（IMSUNLD データベースタイプ）へのアクセスに PCB 名は使用しません。

第 9 章

PowerExchange Navigator の例

この章では、以下の項目について説明します。

- [PowerExchange Navigator の例の概要, 258 ページ](#)
- [単一レコードデータマップの追加 - 例, 259 ページ](#)
- [配列を含む単一レコードデータマップの追加 - 例, 263 ページ](#)
- [複雑なテーブルを含む複数レコードデータマップの追加 - 例, 265 ページ](#)
- [データマップへの COBOL コピーブックのインポート - 例, 268 ページ](#)
- [REDEFINES 文を含む COBOL コピーブックのインポート - 例, 272 ページ](#)
- [PL/I コピーブックのインポート - 例, 276 ページ](#)
- [配列を正規化するためのデータマップでのテーブルプロパティの定義 - 例, 278 ページ](#)
- [PowerExchange CallProg 関数を使用したユーザーイグジットプログラムの呼び出し - 例, 282 ページ](#)

PowerExchange Navigator の例の概要

PowerExchange には、データマップを作成する方法を学習するために使用できるデータマップ、コピーブック、データファイルの例が用意されています。

さらに、PowerExchange には、アセンブリ言語、C、または PL/I で記述された以下のタイプのサンプルプログラムが用意されています。

- ユーザーアクセス方法プログラム: uam*. *
- レコードレベル出口プログラム: urle*. *
- CallProg 関数を使用して呼び出すプログラム: ucpe*. *

ヘッダファイルと JCL の例は、これらのプログラムのために用意されています。

サンプルファイルは、Windows の PowerExchange インストールディレクトリの examples サブディレクトリにあります。以下に例を示します。

C:\Informatika\PowerExchange\...examples\

この節では、サンプルファイルを使用して次のタイプのデータマップを作成します。

- 単一レコードデータマップ
- 配列を含む単一レコードデータマップ
- 複雑なテーブルがある複数レコードデータマップ
- インポートした COBOL コピーブックがあるデータマップ

- REDEFINES 文を含むインポートした COBOL コピーブックがあるデータマップ
- フィールドに複数の定義があり、インポートした PL/I コピーブックがあるデータマップ
- 複数回出現する配列として定義されたフィールドを正規化するデータマップ
- PowerExchange CallProg 関数を使用して、データマップレコード内のフィールドのクラスタイプを返すユーザー出口プログラムを呼び出すユーザー定義フィールドがあるデータマップ

単一レコードデータマップの追加 - 例

この例では、単一レコードデータマップを作成します。

この例では、以下のタスクを完了する方法について示します。

1. データマップを追加する。
2. データマップにレコードを追加する。
3. データマップ内のレコードにフィールドを追加する。
4. データマップにテーブルを追加する。
5. 単一レコードデータマップをテストする。

この例では、demo1.dat データファイルを使用します。

手順 1。データマップの追加

demo1.dat サンプルファイルを使用して、demo.map1 データマップを作成します。

1. リソースエクスプローラの [リソース] タブで、[追加] > [データマップ] をクリックします。
2. [名前] ダイアログボックスに、必要な情報を入力します。

次の表では、必須情報を示します。

オプション	説明
スキーマ名	「demo」と入力します。
データマップ名	「map1」と入力します。
アクセス方式	【SEQ】を選択します。
レコード定義のインポート	このオプションの選択を取り消します。レコード定義を後で指定します。

3. [次へ] をクリックします。
4. **SEQ アクセス方式**ダイアログボックスに、必要な情報を入力します。

次の表では、必須情報を示します。

オプション	説明
ファイル名	[参照] ボタンをクリックし、examples ディレクトリにある demo1.dat ファイルを参照します。
フィールド区切り記号	このオプションを選択して、区切り文字のカンマ (,) を入力します。キーボードから区切り文字を入力できない場合、16 進数表現「x'mm'」を入力します。demo1.dat データファイルは区切り文字を使用しているファイルのため、区切り文字を入力する必要はありません。

5. **【完了】** をクリックします。

demo.map1 データマップは、リソースエクスプローラの **【データマップ】** タブに表示されます。

関連項目：

- [「データマップの追加」 \(ページ 66\)](#)

手順 2. データマップへのレコードの追加

record1 レコードを demo.map1 データマップに追加します。

1. リソースエクスプローラの **【データマップ】** タブで demo.map1 を右クリックし、**【レコードの追加】** をクリックします。

【テーブルの追加】 ダイアログボックスが表示されます。

2. **【レコード名】** ボックスに、「record1」と入力します。
3. **【OK】** をクリックします。

レコードウィンドウに、record1 レコードが表示されます。

関連項目：

- [「データマップレコードの追加」 \(ページ 69\)](#)

手順 3。 データマップ内のレコードへのフィールドの追加

demo.map1 データマップの record1 レコードに、シーケンス番号、名前、性別のフィールドを追加します。

1. リソースエクスプローラの **【データマップ】** タブで、**【record1】** をクリックします。
2. **【追加】** > **【フィールド】** をクリックします。
3. **【フィールドの追加】** ダイアログボックスで、フィールドに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
フィールド名	「seqno」と入力します。
フィールドの型	[NUMCHAR] を選択します。
精度	「2」と入力します。

オプション	説明
位取り	「0」と入力します。
長さ	「2」と入力します。 【長さ】 フィールドは最大入力文字数を示し、このフィールドには先頭の空白やその他の文字を入力できます。長さの値は、精度の値以上の値を入力する必要があります。

4. **【OK】** をクリックします。
レコードウィンドウにシーケンス番号フィールドが表示されます。
5. レコードウィンドウで、**【record1】** をクリックします。
6. **【追加】** > **【フィールド】** をクリックします。
7. **【フィールドの追加】** ダイアログボックスで、フィールドに関する情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
フィールド名	「name」と入力します。
フィールドの型	【CHAR】 を選択します。
長さ	「20」と入力します。

8. **【OK】** をクリックします。
レコードウィンドウに名前フィールドが表示されます。
9. **【データマップ】** タブで、**【record1】** をクリックします。
10. **【追加】** > **【フィールド】** をクリックします。
11. **【フィールドの追加】** ダイアログボックスで、フィールドに関する情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
フィールド名	「gender」と入力します。
フィールドの型	【CHAR】 を選択します。
長さ	「1」と入力します。

12. **【OK】** をクリックします。
レコードウィンドウに性別フィールドが表示されます。

関連項目：

- [「レコードへのフィールドの追加」](#) (ページ 70)

手順 4. データマップへのテーブルの追加

table1 テーブルを demo.map1 データマップに追加します。table1 テーブルの列は、record1 レコードのフィールドに基づきます。

1. リソースエクスプローラの **【データマップ】** タブで、demo.map1 を右クリックして **【テーブルの追加】** を選択します。

2. **【テーブルの追加】** ダイアログボックスで、テーブルに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
テーブル名	「table1」と入力します。
レコードの依存関係	record1 レコードが表示されていることを確認します。
グループと配列のみ	このオプションを選択します。このオプションにより、グループフィールドとして定義されたフィールドや配列が表示されます。このオプションを選択しない場合、レコード内のすべてのフィールドが表示されます。
選択したレコードの複数インスタンスをどのように処理しますか?	【新しい行】 を選択します。このオプションにより、レコードのすべてのインスタンスに対して新しい行が作成されます。

3. **【OK】** をクリックします。

テーブルウィンドウに、table1 テーブルが表示されます。table1 テーブルには、record1 レコードに追加したフィールドに基づいた列が含まれます。

関連項目：

- [「データマップテーブルの追加」](#) (ページ 80)

手順 5. 単一レコードデータマップのテスト

demo.map1 データマップでデータベース行のテストを実行します。データマップにローカルでアクセスします。

1. テーブルウィンドウで、**【table1】** を選択します。
2. **【ファイル】** > **【データベース行のテスト】** をクリックします。
3. データマップをリモート場所へ送信するように求めるメッセージボックスが表示されたら、**【はい】** をクリックします。
4. **【データマップリモートノード】** ダイアログボックスで、**【場所】** リストから **【ローカル】** をクリックします。
5. **【OK】** をクリックします。
6. **【データベース行のテスト】** ダイアログボックスで、**【実行】** をクリックします。
データベース行のテストによる出力ウィンドウに出力が表示されます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)

配列を含む単一レコードデータマップの追加 - 例

この例では、複数のレコードタイプを含むデータファイルを使用して、単一レコードデータマップを追加します。PowerExchange で配列を含むレコードから出力を表示する方法を制御するテーブルのオプションを設定します。

この例では、以下のタスクを完了する方法について示します。

1. データファイルをデータマップと関連付ける。
2. データマップ内のデータをフィルタリングする。
3. 配列内の各エレメント用の出力行を作成する。
4. 配列内のすべてのエレメント用の単一出力行を作成する。

この例では、demo.map2.dmp と demo2.dat ファイルを使用します。

手順 1. データマップとデータファイルの関連付け

データファイルをデータマップと関連付ける。

1. リソースエクスプローラの [リソース] タブで、[demo.map2] をダブルクリックします。
2. [データマップ] タブで、[demo.map2] を右クリックして、[プロパティ] をクリックします。
[データマップのプロパティ] ダイアログボックスが表示されます。
3. [SEQ アクセス方式] タブで [参照] ボタンをクリックして、examples ディレクトリにある demo2.dat ファイルを参照します。
4. [OK] をクリックします。
5. [OK] をクリックします。

関連項目：

- [「データファイルおよびファイルリスト処理」 \(ページ 54\)](#)

手順 2. データマップ内のデータのフィルタリング

レコード ID により、ランタイム時のフィルタリングメカニズムが提供され、目的のレコードを選択するための SQL 文は必要なくなります。

1. リソースエクスプローラの [データマップ] タブで、[record1] をクリックします。
2. レコードウィンドウで、rectype フィールドを右クリックし、[プロパティ] をクリックします。
[フィールドのプロパティ] ダイアログボックスが表示されます。

3. **【レコード ID 値】** リストで、**【=】** または **【<>】** ボタンをクリックして、データをフィルタリングするための演算子を選択します。レコード ID の 10 と 20、これに演算子の設定を組み合わせたものにより、どのデータを表示するかを決めます。次のうち 1 つのアクションを実行します。
 - **【=】** ボタンをクリックして、レコード ID 値が 10 または 20 のいずれかと等しいレコードを表示します。 **データベース行のテストによる出力** ウィンドウには、レコード ID が 10 または 20 と等しくないレコードは表示されません。
 - **【<>】** ボタンをクリックして、レコード ID 値が 10 または 20 より大きい小さい（等しくない）レコードを表示します。

注: データマップには、データではなくレコード ID が格納されます。 **【レコード ID】** フィールドは、データの列ビューに表示されない場合があります。WRITE ではレコード ID フィルタリングを使用できません。
4. **【OK】** をクリックします。

手順 3。配列内の各エレメント用の出力行の作成

配列内の各エレメント用の出力行を作成する。

1. リソースエクスプローラの **【データマップ】** タブで、**【row_out】** テーブルを右クリックして **【プロパティ】** をクリックします。
【テーブルのプロパティ】 ダイアログボックスが表示されます。
2. **【テーブルプロパティ】** ダイアログボックスで、配列内の各エレメントに出力行を作成するためのオプションを選択します。
以下の表に、これらのオプションを示します。

オプション	説明
フィールド	【グループと配列のみ】 を選択します。このオプションにより、グループフィールドや配列フィールドとして定義したフィールドがリストに表示されます。
フィールド	項目フィールドを選択します。このリストでは、グループまたは配列として定義されたフィールドを選択することができます。
選択したレコードの複数インスタンスをどのように処理しますか?	【新しい行】 を選択します。このオプションにより、項目フィールドに出現するデータは、それぞれ固有の行に表示されます。

3. **【OK】** をクリックします。
4. **【データマップ】** タブで、**【row_out】** テーブルをダブルクリックして **【テーブルプロパティ】** ダイアログボックスを開きます。
【items_L】 列が青緑色で表示されます。これは配列の一部であることを示します。
5. データマップをテストするには、データベース行のテストを実行します。データマップにローカルでアクセスし、**【データベース行のテスト】** ダイアログボックスの **【DB タイプ】** リストから **【NRDB】** をクリックします。
データベース行のテストによる出力 ウィンドウに、配列のエレメント数に応じて複数の出力行が単一入力レコードから表示されます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)
- [「【テーブルプロパティ】 ダイアログボックスでの配列プロパティの定義」 \(ページ 60\)](#)

手順 4. 配列内のすべてのエレメント用の単一出力行の作成

配列を含むレコード 1 つに対して単一出力行を作成します。配列にあるすべてのエレメントが単一行に表示されます。

1. row_out テーブルの **【テーブルプロパティ】** ダイアログボックスで、配列内のすべてのエレメントに 1 個の出力行を作成するオプションを選択します。

以下の表に、これらのオプションを示します。

オプション	説明
フィールド	項目フィールドの選択を取り消します。
選択したレコードの複数インスタンスをどのように処理しますか?	【配列】 を選択します。
配列	[1] を選択します。
オーバーフロー時の新しい行	このオプションを選択します。PowerExchange により、出力行がいっぱいになるまで入力が行われ、オーバーフローレコードのある新しい行が表示されます。

2. **【OK】** をクリックします。
3. データマップをテストするには、データベース行のテストを実行します。データマップにローカルでアクセスし、**【データベース行のテスト】** ダイアログボックスの **【DB タイプ】** リストから **【NRDB】** を選択します。

データベース行のテストによる出力ウィンドウに、レコードごとに 1 行が表示されます。レコード内の配列のすべてのエレメントが単一行に表示されます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)
- [「【テーブルプロパティ】 ダイアログボックスでの配列プロパティの定義」 \(ページ 60\)](#)

複雑なテーブルを含む複数レコードデータマップの追加 - 例

この例では、異なるタイプのレコード間に階層関係が存在する複雑なテーブルを定義します。

この例では、以下の概念について示します。

- データマップテーブルでは、関連する複数のデータマップレコードを含めることができる。
- レコード間の階層関係を定義することができる。

この例では、以下のタスクを完了する方法について示します。

1. データファイルを複数レコードデータマップと関連付ける。
2. 複数レコードデータマップ内のレコードを表示する。
3. 複数レコードデータマップ内のレコードの階層を定義する。

4. 複数レコードデータマップをテストする。

この例では、demo.map3.dmp と demo3.dat ファイルを使用します。

手順 1. 複数レコードデータマップと 1 つのデータファイルの関連付け

データファイルを複数レコードデータマップと関連付ける。

1. リソースエクスプローラの [リソース] タブで、[demo.map3] データマップをダブルクリックします。
2. リソースエクスプローラの [データマップ] タブで、[demo.map3] を右クリックして [プロパティ] をクリックします。
[データマップのプロパティ] ダイアログボックスが表示されます。
3. [SEQ アクセス方式] タブで [参照] ボタンをクリックして、examples ディレクトリにある demo3.dat ファイルを参照します。
4. [OK] をクリックします。
5. [OK] をクリックします。

関連項目：

- [「データファイルおよびファイルリスト処理」 \(ページ 54\)](#)

手順 2. 複数レコードデータマップ内のレコードの表示

レコード ID によって、フラットファイル内で複数のレコードを識別します。

demo3.dat ファイルの各行の最初の文字は、demo.map3 データマップにより rectype フィールドとして定義され、rec1 レコードと rec2 レコードを区別します。

1. リソースエクスプローラの [リソース] タブで、[demo.map3] をダブルクリックします。
2. リソースエクスプローラの [データマップ] タブで、[rec1] レコードをクリックします。
3. レコード - rec1 ウィンドウで、[rectype] フィールドをダブルクリックします。
[レコード ID 値] ボックスには、rectype フィールドのレコード ID が 1 であることが示されます。
4. リソースエクスプローラの [データマップ] タブで、[rec2] レコードをクリックします。
5. レコード - rec2 ウィンドウで、[rectype] フィールドをダブルクリックします。
[レコード ID 値] ボックスには、rectype フィールドのレコード ID が 2 であることが示されます。

手順 3. 複数レコードデータマップ内のレコードの階層の定義

rec1 レコードと rec2 レコード間の階層関係を定義します。この階層関係には、データソースの実際の階層が反映されます。

1. リソースエクスプローラの [データマップ] タブで、[row_out] テーブルを右クリックして [プロパティ] をクリックします。
[レコードの依存関係] リストに、rec1 レコードと rec2 レコード間の階層が表示されます。
2. この階層を削除するには、[レコードの依存関係] リストで [rec2] を右クリックし、[削除] をクリックします。
このアクションにより、rec2 レコードは [使用可能なレコード] リストに移動します。

3. **【レコードの依存関係】** リストで **【rec1】** を右クリックし、**【削除】** をクリックします。
このアクションにより、rec1 レコードは **【使用可能なレコード】** リストに移動します。
4. 階層を再定義するには、**【使用可能なレコード】** リストで **【rec1】** を右クリックし、**【レコードの追加】** をクリックします。
このアクションにより、rec1 レコードは **【レコードの依存関係】** リストに移動します。
このレコードでは、**【選択したレコードの複数インスタンスをどのように処理しますか?】** オプションが **【新しい行】** に設定されています。
5. **【使用可能なレコード】** リストで、**【rec2】** を右クリックし、**【レコードを子として追加】** をクリックします。
このアクションにより、rec2 レコードが rec1 親レコードの子レコードとして **【レコードの依存関係】** リストに移動します。
このレコードでは、**【選択したレコードの複数インスタンスをどのように処理しますか?】** オプションが **【新しい行】** に設定されています。
6. **【OK】** をクリックします。
テーブルウィンドウ では、名前列は rec1 レコードの名前フィールドに、アドレス列は rec2 レコードのアドレスフィールドに基づきます。
テーブルの各列の名前は、その列に基づくレコード名のプレフィックスが付けられます。

手順 4.複数レコードデータマップのテスト

複数レコードデータマップでデータベース行のテストを実行します。データマップにローカルでアクセスします。

1. リソースエクスプローラの **【データマップ】** タブで、**【row_out】** をクリックします。
2. **【ファイル】** > **【データベース行のテスト】** をクリックします。
3. データマップをリモート場所へ送信するように求めるメッセージボックスが表示されたら、**【はい】** をクリックします。
4. **【データマップリモートノード】** ダイアログボックスで、**【場所】** リストから **【ローカル】** を選択します。
5. **【OK】** をクリックします。
6. **【データベース行のテスト】** ダイアログボックスで、**【実行】** をクリックします。

データベース行のテストによる出力ウィンドウ では、名前列は rec1 レコードから、アドレス列は rec2 レコードから入力されます。

【row_out】 テーブルの **【テーブルプロパティ】** ダイアログボックスで階層を変更し、rec2 レコードを rec1 レコードの兄弟にすると、**データベース行のテストによる出力ウィンドウ** の出力が変化します。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)

データマップへの COBOL コピーブックのインポート - 例

この例では、複数レコードデータマップを追加し、その中に COBOL コピーブックをインポートします。

この例では、以下のタスクを完了する方法について示します。

1. データマップを追加し、OCCURS DEPENDING ON 句を含む COBOL コピーブックをそのデータマップにインポートする。
データマップを追加するときに、固定長データファイルを使用します。データマップを追加した後に、フラットファイルの階層ビューを作成します。
2. データマップレコード内のフィールドに日付マスクを適用する。
3. WHERE 句を使用して出力から空のレコードをフィルタリングする。
4. OCCURS DEPENDING ON 句での項目に対する複数行の出力を表示する。

この例では、train3.dat と train3.cob ファイルを使用します。

手順 1。データマップの追加とデータマップへの COBOL コピーブックのインポート

シーケンシャルフラットファイルのためのデータマップを追加し、そのデータマップに COBOL コピーブックをインポートします。

1. リソースエクスプローラの [リソース] タブで、[追加] > [データマップ] をクリックします。
2. [名前] ダイアログボックスに、データマップに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
スキーマ名	「COBOL」と入力します。
データマップ名	「map2」と入力します。
アクセス方式	[SEQ] を選択します。
レコード定義のインポート	このオプションを選択します。

3. [次へ] をクリックします。
4. [SEQ アクセス方式] ダイアログボックスで、データファイルに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
ファイル名	[参照] ボタンをクリックし、examples ディレクトリにある train3.dat ファイルを参照します。
固定	このオプションを選択します。
サイズ	「60」と入力します。

5. **【完了】** をクリックします。
6. **【コピーブックのインポート - ソースの詳細】** ダイアログボックスで、コピーブック情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
ソース	【ローカル】 を選択します。
タイプ	【COBOL】 を選択します。
開始	「7」と入力します。
終了	「72」と入力します。

7. **【次へ】** をクリックします。
8. **【コピーブックのインポート - ローカル COBOL 詳細】** ダイアログボックスで、[参照] ボタンをクリックし、examples ディレクトリにある train3.cob ファイルを参照します。
注: コピーブックを参照するには、**【プレビュー】** をクリックします。
9. **【次へ】** をクリックします。
10. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するアクションを選択し、**【完了】** をクリックします。詳細については、[「【コピーブックのインポート - 構成の詳細】 ダイアログボックス」 \(ページ 161\)](#)を参照してください。
11. **コピーブック情報のインポート**ウィンドウで、インポートの情報を確認し **【OK】** をクリックします。
12. NAME_REC レコードと ACCOUNT_REC レコードの **【レコード定義】** ダイアログボックスで、**【OK】** をクリックします。
詳細については、[「【レコード定義】 ダイアログボックスと【重複レコード定義】 ダイアログボックス」 \(ページ 162\)](#)を参照してください。
COBOL インポートウィンドウに、インポートしたコピーブックが表示されます。
コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。
注: コピーブックメッセージログウィンドウに表示されるエラーに対して、コピーブックで関連行を検索するには、メッセージをダブルクリックします。
13. **COBOL インポート**ウィンドウを閉じます。
14. リソースエクスプローラの **【データマップ】** タブで、**[NAME_REC]** レコードをクリックします。
15. レコードウィンドウで、RECTYPE フィールドを右クリックし、**【プロパティ】** をクリックします。
【フィールドのプロパティ】 ダイアログボックス内の **【名前】** タブが表示されます。
16. **【レコード ID 値】** ボックスに、「01」と入力します。

注: インポートされたコピーブックからフィールドに元のプロパティを表示するには、**[追加のプロパティ]** タブをクリックします。

17. **[OK]** をクリックします。

レコードウィンドウに、**RECTYPE** フィールドアイコンが緑色で表示されます。

18. リソースエクスプローラの **[データマップ]** タブで、**[ACCOUNT_REC]** レコードをクリックします。

19. レコードウィンドウで、**RECTYPE** フィールドを右クリックし、**[プロパティ]** をクリックします。

[フィールドのプロパティ] ダイアログボックス内の **[名前]** タブが表示されます。

20. **[レコード ID 値]** ボックスに、「02」と入力します。

21. **[OK]** をクリックします。

レコードウィンドウに、**RECTYPE** フィールドアイコンが緑色で表示されます。

22. リソースエクスプローラの **[データマップ]** タブで、**[COBOL.map2]** を右クリックし、**[テーブルの追加]** をクリックします。

23. **[テーブルの追加]** ダイアログボックスでテーブルに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
テーブル名	「both」と入力します。
使用可能なレコード	[NAME_REC] レコードを右クリックし、 [レコードの追加] をクリックします。次に、 [ACCOUNT_REC] レコードを右クリックし、 [レコードを子として追加] をクリックします。

24. **[OK]** をクリックします。

25. 日付マスクを **POLICY_DATE** レコードに適用する前にテーブル出力を表示するには、データベース行のテストを実行します。リソースエクスプローラの **[データマップ]** タブで、**[両方]** のテーブルをクリックし、**[ファイル]** **[データベース行のテスト]** > をクリックします。

26. **[データベース行のテスト]** ダイアログボックスで、データベース行のテストに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
DB の種類	[NRDB] を選択します。
場所	[ローカル] を選択します。

27. **[実行]** をクリックします。

データベース行のテストによる出力ウィンドウに出力が表示されます。

28. データベース行のテストによる出力ウィンドウとデータベース行のテストウィンドウを閉じます。

29. データベース行のテストによる出力ウィンドウとデータベース行のテストウィンドウを閉じます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)
- [「\[コピーブックのインポート - ローカル詳細\] ダイアログボックス」 \(ページ 153\)](#)
- [「\[コピーブックのインポート - リモートの詳細\] ダイアログボックス」 \(ページ 155\)](#)

手順 2。データマップレコード内のフィールドへの日付マスクの適用

データマップ内のフィールドに日付マスクを適用し、データベース行のテストを実行する前に無関係なフィールドをテーブルから削除します。

1. **both** テーブルの**テーブル**ウィンドウで、次のカラムを 1 つずつ右クリックし、**【削除】** をクリックします。
 - RECTYPE
 - ACCOUNT_REC_ACCOUNT
 - ACCOUNT_REC_RECTYPE各レコードに対して、削除の確認を求めるウィンドウで **【OK】** をクリックします。
2. **リソースエクスプローラ**の **【データマップ】** タブで、**【ACCOUNT_REC】** レコードをクリックします。
3. **レコード**ウィンドウで、**POLICY_DATE** フィールドを右クリックし、**【プロパティ】** をクリックします。
4. **【フィールドフォーマット】** リストから、**【Y2-MM-D2】** を選択します。
5. **【OK】** をクリックします。
6. **リソースエクスプローラ**の **【データマップ】** タブで、**【both】** テーブルをクリックします。
7. **【ファイル】** > **【データベース行のテスト】** をクリックします。
8. **【データベース行のテスト】** ダイアログボックスで、データベース行のテストに関する情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
DB の種類	【NRDB】 を選択します。
場所	【ローカル】 を選択します。

9. **【実行】** をクリックします。
データベース行のテストによる出力ウィンドウでは、POLICY_DATE 列の日付形式に世紀が含まれていません。

関連項目：

- [「手順 3.WHERE 句を使用した出力からの空のレコードのフィルタリング」](#) (ページ 271)
- [「データマップのテスト」](#) (ページ 231)

手順 3.WHERE 句を使用した出力からの空のレコードのフィルタリング

WHERE 句を使用して、**データベース行のテストによる出力**ウィンドウの出力から空のレコードをフィルタリングします。

1. **【データベース行のテスト】** ダイアログボックスで、**【SQL 文】** ボックスに以下の文を入力します。
`select * from COBOL.map2.both where itemct > 0`
2. **【データベース行のテスト】** ダイアログボックスで、**【実行】** をクリックします。
データベース行のテストによる出力ウィンドウで、項目が含まれない 4 番目のレコードが出力に表示されなくなります。
3. **データベース行のテストによる出力**ウィンドウと**データベース行のテスト**ウィンドウを閉じます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)

手順 4.OCCURS DEPENDING ON 句内の項目に対する複数の出力行の表示

OCCURS DEPENDING ON 句の項目に対して、複数の出力行をデータベース行のテストによる出力ウィンドウに表示します。

1. リソースエクスプローラの [データマップ] タブで、[both] テーブルを右クリックして [プロパティ] をクリックします。
2. [テーブルプロパティ] ダイアログボックスで、[フィールド] リストから [ITEMS] チェックボックスを選択します。
3. [OK] をクリックします。
テーブルウィンドウに、ITEMS_L 列のアイコンが青緑色で表示されます。
4. リソースエクスプローラの [データマップ] タブで、[both] テーブルをクリックします。
5. [ファイル] > [データベース行のテスト] をクリックします。
6. [データベース行のテスト] ダイアログボックスで、[実行] をクリックします。
データベース行のテストによる出力ウィンドウに、レコードの項目ごとに 1 行が表示されます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)

REDEFINES 文を含む COBOL コピーブックのインポート - 例

複数レコードデータマップを追加し、その中に REDEFINES 文を含む複雑な COBOL コピーブックをインポートできます。

この例では、以下のタスクを完了する方法について示します。

1. データマップを追加し、その中に COBOL コピーブックをインポートする。
データマップを追加するときに、58320 のレコードを含む EBCDIC データの固定長バイナリデータファイルを使用します。各レコードの長さは 57 バイトです。この EBCDIC データファイルは、IBM メインフレームシステムからデータの読み込みをシミュレートします。

【データマップの追加】 ダイアログボックスで、データファイルのヘッダレコードをスキップするようにレコードのスキップオプションを設定します。

列 1 から 6 には行番号が、列 73 から 80 にはコメントが含まれる COBOL コピーブックをインポートします。COBOL コピーブックには、REDEFINES 文も含まれます。

注: PowerExchange では、コピーブックのコメント行は無視されます。

2. データマップ内の GROUP フィールドに日付マスクを適用する。

この例では、train5.dat と train5.cob ファイルを使用します。

手順 1. データマップの追加とデータマップへの REDEFINES 文を含む COBOL コピーブックのインポート

複数レコードデータマップを追加し、そのデータマップに REDEFINES 文を含む COBOL コピーブックをインポートします。

1. リソースエクスプローラの [リソース] タブで、[追加] > [データマップ] をクリックします。
2. [名前] ダイアログボックスで、データマップに関する情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
スキーマ名	「COBOL」と入力します。
データマップ名	「map3」と入力します。
アクセス方式	[SEQ] を選択します。
レコード定義のインポート	このオプションを選択します。

3. [次へ] をクリックします。
4. [SEQ アクセス方式] ダイアログボックスで、データファイルに関する情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
ファイル名	[参照] ボタンをクリックし、examples ディレクトリにある train5.dat ファイルを参照します。
固定	このオプションを選択します。
サイズ	「57」と入力します。
デフォルト	このオプションを選択します。
エンコード	[EBCDIC] を選択します。
Codepage	[デフォルト] を選択します。
ファイルの先頭の	「1」と入力します。

エンコードとコードページを設定した後、[固定] オプションを再度選択します。

5. [完了] をクリックします。
6. [コピーブックのインポート-ソースの詳細] ダイアログボックスで、コピーブックの情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
ソース	[ローカル] を選択します。

オプション	説明
タイプ	【COBOL】を選択します。
開始	「7」と入力します。
終了	「72」と入力します。

7. **【次へ】** をクリックします。
8. **【コピーブックのインポート - ローカル COBOL 詳細】** ダイアログボックスで、**【参照】** ボタンをクリックし、examples ディレクトリにある train5.cob ファイルを参照します。
注: コピーブックを参照するには、**【プレビュー】** をクリックします。
9. **【次へ】** をクリックします。
10. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するアクションを選択し、**【完了】** をクリックします。詳細については、[「【コピーブックのインポート - 構成の詳細】 ダイアログボックス」 \(ページ 161\)](#)を参照してください。
11. **コピーブック情報のインポート**ウィンドウで、インポートの情報を確認し **【OK】** をクリックします。
12. MASTER_REC レコードの **【レコード定義】** ダイアログボックスで、**【OK】** をクリックします。
詳細については、[「【レコード定義】 ダイアログボックスと【重複レコード定義】 ダイアログボックス」 \(ページ 162\)](#)を参照してください。
コピーブックの再定義ウィンドウにより、コピーブックに **REDEFINES** 文が含まれていることが通知されます。
COBOL インポートウィンドウが表示されます。
13. PIC S9(8)フィールドを含む行を指定するには、**【再定義】** > **【次へ】**、または **【再定義】** > **【前へ】** をクリックします。
14. **【インポート】** > **【再開】** をクリックします。
COBOL インポートウィンドウに、インポートしたコピーブックが表示されます。
コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。
注: メッセージログの最初の行は、**COBOL インポート**ウィンドウのレコードを参照します。**COBOL インポート**ウィンドウで、メッセージログの最初の行をクリックして、特定のレコードを強調表示します。
15. **COBOL インポート**ウィンドウを閉じます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)
- [「【コピーブックのインポート - ソースの詳細】 ダイアログボックス」 \(ページ 149\)](#)
- [「【コピーブックのインポート - ローカル詳細】 ダイアログボックス」 \(ページ 153\)](#)
- [「【コピーブックのインポート - リモートの詳細】 ダイアログボックス」 \(ページ 155\)](#)

手順 2。 データマップ内の GROUP フィールドへの日付マスクの適用

データマップ内の GROUP フィールドに日付マスクを適用します。

1. リソースエクスプローラの **【データマップ】** タブで、**【MASTER_REC】** レコードをクリックします。
2. レコードウィンドウで、**MASTER_DATE** フィールドをクリックします。

MASTER_DATE フィールドは、UZONED フィールドとして定義されている以下のサブフィールドを含む GROUP フィールドとして定義されます。

- DATE_YY
- DATE_MM
- DATE_DD

次の手順では、これらのサブフィールドを削除し、MASTER_DATE GROUP フィールドからのデータベース行のテストによる出力に単一の日付列を作成するために、MASTER_DATE フィールドに対してプロパティを設定します。

3. レコードウィンドウで、DATE_YY フィールドを選択します。Shift キーを押して [DATE_DD] をクリックします。ツールバーにある [削除] ボタンをクリックします。
4. フィールドとともに削除される相互参照に関する詳細を表示するには、確認のダイアログボックスで [詳細] をクリックします。
5. [はい] をクリックします。
6. レコードウィンドウで、MASTER_DATE フィールドを右クリックし、[プロパティ] をクリックします。
7. [フィールドのプロパティ] ダイアログボックスで、フィールドに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
フィールドの型	[CHAR] を選択します。
長さ	「6」と入力します。
フィールドフォーマット	[Y2-MM-D2] を選択します。ハイフンを削除します。

8. [OK] をクリックします。
9. リソースエクスプローラの [データマップ] タブで、[MASTER_REC] テーブルをクリックします。
10. [追加] > [カラム] をクリックします。
11. [カラムの追加] ダイアログボックスに、カラムに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
名前	「new_date」と入力します。
ベースフィールド	[MASTER_REC:MASTER_DATE] を選択します。

12. [OK] をクリックします。
13. リソースエクスプローラの [データマップ] タブで、[MASTER_REC] テーブルをクリックします。
14. [ファイル] > [データベース行のテスト] をクリックします。
15. [データベース行のテスト] ダイアログボックスで、[実行] をクリックします。

データベース行のテストによる出力ウィンドウで、[new_date] 列に日付が表示されます。

関連項目：

- [「テーブルへの列の追加」 \(ページ 80\)](#)
- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)

- [「データマップのテスト」 \(ページ 231\)](#)

PL/I コピーブックのインポート - 例

PL/I コピーブックを COBOL コピーブックの代わりに使用できます。この例では、複数レコードデータマップを追加する方法とその中に PL/I コピーブックをインポートする方法について示します。

この例では、以下のタスクを完了する方法について示します。

1. データマップを追加し、フィールドに複数の定義を持つ PL/I コピーブックをその中にインポートする。
データマップを追加するときに、単一の ASCII ヘッダレコードに続く 58320 のレコードを含む EBCDIC データレコードの固定長バイナリデータファイルを使用します。各レコードの長さは 57 バイトです。この EBCDIC データファイルは、IBM メインフレームシステムからデータの読み込みをシミュレートします。
2. データマップをテストします。

この例では、train5.dat と train5.pl1 ファイルを使用します。

手順 1. データマップの追加とデータマップへの PL/I コピーブックのインポート

データマップを追加し、そのデータマップに PL/I コピーブックをインポートします。

1. リソースエクスプローラの [リソース] タブで、[追加] > [データマップ] をクリックします。
2. [名前] ダイアログボックスで、データマップに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
スキーマ名	「PLI」と入力します。
データマップ名	「map5」と入力します。
アクセス方式	【SEQ】を選択します。
レコード定義のインポート	このオプションを選択します。

3. [次へ] をクリックします。
4. 【SEQ アクセス方式】ダイアログボックスで、データファイルに関する情報を入力します。

以下の表に、入力する情報を示します。

オプション	説明
ファイル名	[参照] ボタンをクリックし、examples ディレクトリにある train5.dat ファイルを参照します。
固定	このオプションを選択します。
サイズ	「57」と入力します。

オプション	説明
デフォルト	このオプションを選択します。
エンコード	[EBCDIC] を選択します。
ファイルの先頭の	「1」と入力します。

エンコードを設定した後、**【固定】** オプションを再度選択します。

5. **【完了】** をクリックします。
6. **【コピーブックのインポート-ソースの詳細】** ダイアログボックスで、コピーブックの情報を入力します。
以下の表に、入力する情報を示します。

オプション	説明
ソース	【ローカル】 を選択します。
タイプ	[PL/1] を選択します。
開始	「1」と入力します。
終了	「72」と入力します。

7. **【次へ】** をクリックします。
8. **【コピーブックのインポート - ローカル PL/1 詳細】** ダイアログボックスで、**【参照】** ボタンをクリックし、examples ディレクトリにある train5.pl1 ファイルを参照します。
注: コピーブックを参照するには、**【プレビュー】** をクリックします。
9. **【次へ】** をクリックします。
10. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対するアクションを選択し、**【完了】** をクリックします。詳細については、[「【コピーブックのインポート - 構成の詳細】 ダイアログボックス」 \(ページ 161\)](#)を参照してください。
11. **コピーブック情報のインポート**ウィンドウで、インポートの情報を確認し **【OK】** をクリックします。
12. MASTER_REC レコードの **【レコード定義】** ダイアログボックスで、**【OK】** をクリックします。
詳細については、[「【レコード定義】 ダイアログボックスと【重複レコード定義】 ダイアログボックス」 \(ページ 162\)](#)を参照してください。
【コピーブックの再定義】 メッセージボックスにより、コピーブックに BIN_NO フィールドの 2 つの定義が含まれていることが通知されます。
PL/1 インポートウィンドウが表示されます。
13. FIXED BINARY((16))フィールドを含む行 6 を指定するには、**【再定義】** > **【次へ】**、または **【再定義】** > **【前へ】** をクリックします。
14. **【インポート】** > **【再開】** をクリックします。
PL/1 インポートウィンドウに、インポートしたコピーブックが表示されます。
コピーブックメッセージログウィンドウに、インポート処理の結果が表示されます。
15. **PL/1 インポート**ウィンドウを閉じます。

関連項目：

- [「データマップのテスト」 \(ページ 231\)](#)
- [「\[コピーブックのインポート - ソースの詳細\] ダイアログボックス」 \(ページ 149\)](#)
- [「\[コピーブックのインポート - ローカル詳細\] ダイアログボックス」 \(ページ 153\)](#)
- [「\[コピーブックのインポート - リモートの詳細\] ダイアログボックス」 \(ページ 155\)](#)

手順 2. データマップのテスト

PL/I コピーブックをインポートしたデータマップの MASTER_REC テーブルに対してデータベース行のテストを実行します。

1. リソースエクスプローラの **[データマップ]** タブで、**[MASTER_REC]** テーブルをクリックします。
2. **[ファイル]** > **[データベース行のテスト]** をクリックします。
3. メッセージボックスで **[はい]** をクリックして、PL1.map5 データマップをリモート場所に送信します。
[データマップリモートノード] ダイアログボックスが表示されます。
4. **[データマップリモートノード]** ダイアログボックスで、ローカルのデフォルトの場所を受け入れ、**[OK]** をクリックします。
[データベース行のテスト] ダイアログボックスが表示されます。
5. **[データベース行のテスト]** ダイアログボックスで、**[実行]** をクリックします。
データベース行のテストによる出力ウィンドウに出力が表示されます。

配列を正規化するためのデータマップでのテーブルプロパティの定義 - 例

複数のレコードのレイアウトを一度に定義するには、複数回出現する配列を定義します。次に、データマップでテーブルプロパティを定義して配列を正規化し、ソースファイルから正しい出力を生成します。

この例では、ユーザー定義フィールドをデータマップに追加します。これらのフィールドでは、PowerExchange GenVRowKey 関数と GetSeqWithinLevel 関数を呼び出し、データベース行のテストによる出力で配列の正規化が行われていることを示します。

この例では、以下のタスクを完了する方法について示します。

1. シーケンシャルフラットファイルのためのデータマップを追加する。
データマップを追加するときに、複数回出現する配列を定義する COBOL コピーブックをそのデータマップにインポートします。
2. データマップにユーザー定義フィールドを追加します。
これらのユーザー定義フィールドでは、PowerExchange GenVRowKey 関数と GetSeqWithinLevel 関数を呼び出し、データベース行のテストによる出力で配列の正規化が行われていることを示します。
配列を含むソース行に対して、GenVRowKey 関数は一意の行番号で識別される複数のテーブル行を生成します。
GetSeqWithinLevel 関数により、現在の階層レベルにあるレコードのシーケンス番号が取得されます。
3. 配列を正規化するテーブルプロパティをデータベース行のテストによる出力に定義します。

4. データベース行のテストを実行して、正規化の結果を表示します。

データベース行のテストによる出力では、配列での各出現に対して 1 行が表示されます。

この例では、demo2.dat データファイルと demo2.cob コピーブックを使用して、demo.map2array データマップを作成します。

demo2.cob コピーブックでは、以下の OCCURS 句を使用して複数回出現する配列を定義します。

```
04 ITEM OCCURS 3 DEPENDING ON ITEMCT PIC X(15).
```

手順 1. 複数回出現が定義された配列を含むデータマップの追加

この手順では、シーケンシャルフラットファイルにデータマップを追加し、複数回出現する配列を定義する COBOL コピーブックをそのデータマップにインポートします。

1. リソースエクスプローラの [リソース] タブで、[追加] > [データマップ] をクリックします。
2. [名前] ダイアログボックスに、以下の情報を入力します。
 - [スキーマ名] ボックスに、「demo」と入力します。
 - [データマップ名] ボックスに、「map2array」と入力します。
 - [レコード定義のインポート] オプションを選択します。
 - [アクセス方式] リストで、[SEQ] を選択します。
3. [次へ] をクリックします。
4. SEQ アクセス方式ダイアログボックスに、以下の情報を入力します。
 - [ファイル名] ボックスで [参照] ボタンをクリックして、サンプルディレクトリにある demo2.dat ファイルを参照します。
 - [レコード形式] 領域で、[フィールド区切り記号] オプションを選択して、カンマ (,) 文字を [フィールド区切り記号] ボックスに入力します。
demo2.dat データファイルは、カンマ (,) で区切られたファイルです。
5. [完了] をクリックします。
6. [コピーブックのインポート - ソースの詳細] ダイアログボックスで、以下の情報を入力します。
 - [ソース] 領域で、[ローカル] を選択します。
 - [型] リストで、[COBOL] を選択します。
 - [カラムの範囲] 領域で、「7」を [開始] ボックスに、「72」を [終了] ボックスに入力します。
7. [次へ] をクリックします。
8. [コピーブックのインポート - ローカル COBOL 詳細] ダイアログボックスで、以下の情報を入力します。
 - [参照] ボタンをクリックし、サンプルディレクトリにある demo2.cob ファイルを参照します。
 - コピーブックを参照するには、[プレビュー] をクリックします。コピーブックには、複数回出現する配列を定義する以下の OCCURS 句が含まれます。
04 ITEM OCCURS 3 DEPENDING ON ITEMCT PIC X(15).
9. プレビューウィンドウを閉じます。[コピーブックのインポート - ローカル COBOL 詳細] ダイアログボックスで、[次へ] をクリックします。
10. [コピーブックのインポート - 構成の詳細] ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対する選択されたアクションを確認して、[完了] をクリックします。
11. コピーブック情報のインポートウィンドウで、インポートの情報を確認し [OK] をクリックします。
12. RECORD_IN レコードをインポートするには、[レコード定義] ダイアログボックスで [OK] をクリックします。

13. レコードにフィールドをインポートするには、各フィールドの[フィールド定義]ダイアログボックスで [OK] をクリックします。
14. RECORD_IN テーブルをインポートするには、[テーブル定義] ダイアログボックスで [OK] をクリックします。
COBOL インポートウィンドウに、インポートしたコピーブックが表示されます。 **コピーブックメッセージログ**ウィンドウに、インポート処理の結果が表示されます。
15. **Cobol インポート**ウィンドウと**コピーブックメッセージログ**ウィンドウを閉じます。
16. リソースエクスプローラの [リソース] タブで、RECORD_IN テーブルを選択して [ファイル] > [データベース行のテスト] をクリックします。
PowerExchange Navigator は、RECORD_IN レコードの RECTYPE フィールドが区切りデータファイルに対して無効な型であることを示すエラーメッセージを表示します。 さらに、ITEMCT フィールドも無効な型です。
17. 以下のアクションを完了して、RECTYPE フィールドと ITEMCT フィールドのデータタイプを変更します。
 - IRECORD_IN レコードの**レコード**ウィンドウで、RECTYPE フィールドをダブルクリックします。
[フィールドのプロパティ] ダイアログボックスで、NUMCHAR の [フィールドの型] を選択します。
[長さ] ボックスに「2」と入力します。 [OK] をクリックします。
 - **レコード**ウィンドウで、ITEMCT フィールドをダブルクリックします。
[フィールドのプロパティ] ダイアログボックスで、NUMCHAR の [フィールドの型] を選択します。
[長さ] ボックスで1を入力します。 [OK] をクリックします。
18. リソースエクスプローラの [リソース] タブで、RECORD_IN テーブルを選択して [ファイル] > [データベース行のテスト] をクリックします。
19. [データベース行のテスト] ダイアログボックスで、[実行] をクリックします。
データベース行のテストによる出力ウィンドウに、データベース行のテストの結果が表示されます。

手順 2. ユーザー定義フィールドの追加配列の正規化の明示

この手順では、複数回出現する配列の正規化を示す PowerExchange 関数を呼び出すユーザー定義フィールドを追加します。

ユーザー定義フィールドは、PowerExchange GenVRowKey 関数と GetSeqWithinLevel 関数を呼び出します。これらの関数は、ユーザーが定義したテーブルプロパティが配列を正規化する方法を示すデータベース行のテストによる出力にデータを追加します。データベース行のテストを実行すると、PowerExchange では、配列での各出現に対して 1 行が表示されます。これらの関数の詳細については、[「GenVRowKey」 \(ページ 310\)](#)と [「GetSeqWithinLevel」 \(ページ 322\)](#)を参照してください。

1. demo.map2array データマップと RECORD_IN レコードを開きます。
2. **レコード**ウィンドウで、[Expr(0)] タブをクリックします。
3. [Expr(0)] タブの任意の場所で右クリックし、[最後にフィールドを追加] をクリックします。
4. [名前] 列のセルに、「r1genvrow_idx」と入力します。
5. [型] 列のセルに、「NUM32U」と入力します。
6. [段階] 列のセルで、[R] を選択します。
この選択は、読み取り処理を示します。
7. フィールドに式を入力するには、[式] 列のセルをクリックし、[参照] ボタンをクリックします。
[式エディタ] ダイアログボックスが表示されます。
8. [式エディタ] ダイアログボックスの [関数リスト] リストで、GenVRowKey 関数をダブルクリックします。

9. **【式リスト】** リストで、GenVRowKey 関数名の最後に「()」と入力します。**【検証】** をクリックします。
【検証】 ボックスに、[エラーはありません]メッセージが表示されます。
10. **【OK】** をクリックします。
11. **【Expr(0)】** タブの任意の場所で右クリックし、**【最後にフィールドを追加】** をクリックします。
12. **【名前】** 列のセルに、「r1genseq」と入力します。
13. **【型】** 列のセルに、「NUM32」と入力します。
14. **【段階】** 列のセルで、**【R】** を選択します。
この選択は、読み取り処理を示します。
15. フィールドに式を入力するには、**【式】** 列のセルをクリックし、**【参照】** ボタンをクリックします。
【式エディタ】 ダイアログボックスが表示されます。
16. **【式エディタ】** ダイアログボックスの **【関数リスト】** リストで、GetSeqWithinLevel 関数をダブルクリックします。
17. **【式リスト】** リストで、GetSeqWithinLevel 関数名の最後に「()」と入力します。**【検証】** をクリックします。
【検証】 ボックスに、[エラーはありません]メッセージが表示されます。
18. **【OK】** をクリックします。

手順 3.配列を正規化するためのテーブルプロパティの定義

この手順では、配列を正規化するテーブルプロパティをデータマップに定義します。

1. demo.map2array データマップを開きます。
2. **リソースエクスプローラ**の **【データマップ】** タブで、RECORD_IN テーブルを右クリックして **【プロパティ】** をクリックします。
【テーブルのプロパティ - 定義】 ダイアログボックスが表示されます。
3. **【カラムの生成】** リストから、**【不足しているカラムを使用してリフレッシュ】** を選択します。
テーブルが基にしているレコードには新しいフィールドが含まれているため、対応する列がテーブルに追加されます。
4. **【OK】** をクリックします。
5. **リソースエクスプローラ**の **【データマップ】** タブで、RECORD_IN テーブルを右クリックして **【プロパティ】** をクリックします。
【テーブルのプロパティ - 定義】 ダイアログボックスが表示されます。
6. **【テーブルプロパティ - 定義】**ダイアログボックスで、以下のアクションを実行します。
 - **【カラムの生成】** リストから、**【配列フォーマットの変更の適用】** を選択します。
テーブルが基にしているレコードには、配列として定義される変更フィールドが含まれているため、これらの変更はテーブルの対応する列に反映されます。
 - **【選択したレコードの複数インスタンスをどのように処理しますか？】** リストで、**【新しい行】** を選択します。
このアクションにより、PowerExchange では、配列での各出現に対して、新しい行がデータベース行のテストに表示されます。
7. **【OK】** をクリックします。
【テーブルプロパティ - 定義】ダイアログボックスに関する詳細については、[「テーブルプロパティ - 定義」\(ページ 391\)](#)を参照してください。

手順 4. データマップでの配列の正規化のテスト

この手順では、データマップ内の配列の正規化をテストするためのデータベース行を実行します。

1. demo.map2array データマップを開きます。
2. リソースエクスプローラの [データマップ] タブで、RECORD_IN テーブルを選択して [ファイル] > [データベース行のテスト] をクリックします。
[データベース行のテスト] ダイアログボックスが表示されます。
3. [データベース行のテスト] ダイアログボックスで、[実行] をクリックします。

データベース行のテストによる出力ウィンドウが表示されます。

以下の画像に、データベース行のテストによる出力ウィンドウを示します。

Row Number	RE...	NAME	SEX	ITE...	ITEM_L	r1genvrow_idx	r1genseq
1	10	Mark Jones	M	3	apple	1	1
2	10	Mark Jones	M	3	orange	2	1
3	10	Mark Jones	M	3	pear	3	1
4	20	Shirley Wong	F	1	raspberry	1	2
5	20	John Jackson	M	2	pansy	1	3
6	20	John Jackson	M	2	daisy	2	3
7	10	Donald Leary	M	0	<Null>	1	4
8	30	David Wu	M	1	fox	1	5
9	10	Jean Connor	F	3	dog	1	6
10	10	Jean Connor	F	3	cat	2	6
11	10	Jean Connor	F	3	rabbit	3	6
12	20	Ronald Rose	M	2	horse	1	7
13	20	Ronald Rose	M	2	pony	2	7
14	20	Betsy Martin	F	1	wolf	1	8

1. GenVRowKey 関数の結果
2. GetSeqWithinLevel 関数の結果

r1genvrow_idx 列には、GenVRowKey 関数の結果が表示されます。これは配列での出現回数です。
r1genseq 列には、GetSeqWithinLevel 関数の結果が表示されます。これはレコード数です。

例えば、次の結果が出力に表示されます。

- 最初の行は、レコード 1 の配列での出現 1 に対するデータを表示します。
- 3 番目の行は、レコード 1 の配列での出現 3 に対するデータを表示します。
- 4 番目の行は、レコード 2 の配列での出現 1 に対するデータを表示します。

PowerExchange CallProg 関数を使用したユーザーイグジットプログラムの呼び出し - 例

この例では、PowerExchange CallProg 関数を呼び出すためのユーザー定義フィールドをデータマップレコードに追加し、ユーザーイグジットプログラムを呼び出します。

ユーザーイグジットプログラムは、指定されたフィールドのデータのクラスタイプを返します。ユーザーイグジットプログラムでは、最大長が 15 バイトのフィールドがテストされます。

以下の表に、ユーザーイグジットプログラムが返すクラスタイプを示します。

クラスタイプ	説明
A	アルファベット
H	最大値
L	最小値
N	ゾーン 10 進数
S	スペース

この例では、以下のタスクを完了する方法について示します。

1. z/OS 上の PowerExchange に付属しているサンプルのデータファイルおよびコピーブックを使用して、データマップを追加する。
2. ユーザーイグジットプログラムを作成してコンパイルする。z/OS 上の PowerExchange リスナの LOADLIB ライブラリ内へそのプログラムをリンク編集します。
3. 次のユーザー定義フィールドを MASTER_REC データマップレコードに追加する。
 - classtype_bin_no フィールド、classtype_dec_no フィールド、および classtype_rec_type フィールド。
1 バイトの CHAR フィールドとして定義されます。CallProg 関数によって呼び出されたユーザーイグジットプログラムは、これらのフィールドのうち指定されたフィールドのクラスタイプを返します。クラスタイプを確認するフィールドごとに個別の classtype フィールドを定義する必要があります。
 - rc_bin_no フィールド、rc_decimal_no フィールド、および rc_rec_type フィールド。
NUM32 フィールドとして定義されます。これらのフィールドを使用して CallProg 関数を呼び出します。これらのフィールドに、ユーザーイグジットプログラムの呼び出しのリターンコードが示されます。

PowerExchange によるデータマップレコードのデータチェックが完了する前に、データマップレコードのユーザー定義フィールドで定義されている式およびプログラム呼び出しが実行されます。
4. MASTER_REC テーブルのカラムをリフレッシュして、MASTER_REC レコードに追加したユーザー定義フィールドを取得する。
5. データマップレコードに対してデータベース行のテストを実行し、ユーザーイグジットプログラムの結果をテストして、ユーザーイグジットプログラムが正しく実行されることを確認する。

手順 1。データマップの追加

この手順では、シーケンシャルフラットファイルのためのデータマップを追加し、COBOL コピーブックをインポートします。

データマップを追加するには、z/OS システム上の PowerExchange に付属しているデータセットメンバを使用します。

以下の表に、データセットメンバを示します。

データセットメンバ	説明
KSDSDAT	ソースデータを含むデータファイル。
KSDSCOB	データのレイアウトを定義するためにインポートする COBOL コピーブック。

1. リソースエクスプローラの【リソース】タブで、【追加】 > 【データマップ】をクリックします。
2. 【名前】ダイアログボックスで、データマップのプロパティを定義します。

以下の表に、プロパティを示します。

プロパティ	値
スキーマ名	demo
データマップ名	userexit
アクセス方式	SEQ

また、【レコード定義のインポート】オプションを選択します。

3. 【次へ】をクリックします。
4. 【SEQ アクセス方式】ダイアログボックスで、データファイルに関する情報を入力します。

以下の表に、入力する情報を示します。

プロパティ	値	注意事項
ファイル名	<i>PWX_installation_dataset</i> .DTLDEMO(KSDSDAT)	ここで、 <i>PWX_installation_dataset</i> は PowerExchange のインストールデータセットです。 例えば、ファイル名として「 <i>PWX.V901.DTLDEMO(KSDSDAT)</i> 」と入力できます。
レコード形式	デフォルト	-
先頭をスキップ	0	-

また、【処理するファイルリスト】オプションが無効になっていることを確認します。デフォルトで、このオプションは無効になっています。

5. 【完了】をクリックします。
6. 【コピーブックのインポート - ソースの詳細】ダイアログボックスで、コピーブックのプロパティを定義します。

以下の表に、入力する値を示します。

プロパティ	値
ソース	リモート
タイプ	COBOL
[カラムの範囲] の [開始]	7
[カラムの範囲] の [終了]	72

7. [次へ] をクリックします。
8. [コピーブックのインポート - リモート COBOL の詳細] ダイアログボックスで、リモートシステムにあるコピーブックのプロパティを定義します。

以下の表に、入力する情報を示します。

プロパティ	値	注意事項
ファイル名	<i>PWX_installation_dataset</i> .DTLDEMO(KSDSCOB)	ここで、 <i>PWX_installation_dataset</i> は PowerExchange のインストールデータセットです。 例えば、ファイル名として「 <i>PWX.V901.DTLDEMO(KSDSCOB)</i> 」と入力できます。
場所	z/OS システムのノード	-
ユーザー ID	z/OS システムのユーザー ID	-

プロパティ	値	注意事項
パスワード	<p>z/OS システムのユーザー ID のパスワードまたは有効な PowerExchange パスフレーズです。</p> <p>z/OS アクセス用のパスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 大文字および小文字 - 0～9 の数字 - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 最初の文字はアポストロフィです。</p> <p>パスフレーズに一重引用符 (')、二重引用符 (")、通貨記号を含めることはできません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p>	-
名前を付けてローカルに保存	ローカル Windows システムのコピーブックファイルの名前	-

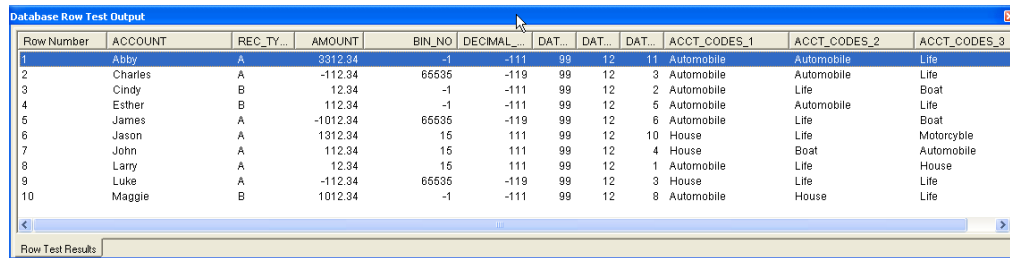
注: コピーブックをプレビューする場合は、**【プレビュー】** をクリックします。コピーブックをプレビューしたら、プレビューウィンドウを閉じます。

9. **【次へ】** をクリックします。
10. **【コピーブックのインポート - 構成の詳細】** ダイアログボックスで、インポートしたレコード、フィールド、およびテーブルに対する選択されたアクションを確認して、**【完了】** をクリックします。
11. **コピーブック情報のインポート** ウィンドウで、インポートの情報を確認し **【OK】** をクリックします。
12. MASTER_REC レコードの **【レコード定義】** ダイアログボックスで、**【OK】** をクリックします。
【コピーブックの再定義】 メッセージボックスが表示され、コピーブックの BIN_NO フィールドに 2 つの定義が存在することが通知されます。この例では、1 つ目の定義が正しい定義です。
13. BIN_NO フィールドの 1 つ目の定義を受け入れるには、**【インポート】** > **【現在行に移動】** をクリックします。次に、**【インポート】** > **【再開】** をクリックしてインポートを再開します。
COBOL インポート ウィンドウにインポートしたコピーブックが表示され、**コピーブックメッセージログ** ウィンドウにインポート処理の結果が表示されます。
14. **COBOL インポート** ウィンドウを閉じます。
15. データマップが正しく追加されたことを確認するには、**リソースエクスプローラ** の **【リソース】** タブで、MASTER_REC テーブルを選択して **【ファイル】** > **【データベース行のテスト】** をクリックします。
16. データマップをリモートの場所に送信するように求められたら、**【はい】** をクリックします。
【データマップリモートノード】 ダイアログボックスが表示されます。
17. **【データマップリモートノード】** ダイアログボックスで、z/OS システムのユーザー ID、パスワード、およびノードを入力します。
【データベース行のテスト】 ダイアログボックスが表示されます。

18. [データベース行のテスト] ダイアログボックスで、デフォルト値を受け入れて【実行】をクリックします。

データベース行のテストによる出力ウィンドウに、データベース行のテストの結果が表示されます。

以下の画像に、データベース行のテストによる出力ウィンドウを示します。



Row Number	ACCOUNT	REC_TY	AMOUNT	BIN_NO	DECIMAL	DAT	DAT	DAT	ACCT_CODES_1	ACCT_CODES_2	ACCT_CODES_3
1	Abby	A	3312.34	-1	-111	99	12	11	Automobile	Automobile	Life
2	Charles	A	-112.34	65535	-119	99	12	3	Automobile	Automobile	Life
3	Cindy	B	12.34	-1	-111	99	12	2	Automobile	Life	Boat
4	Esther	B	112.34	-1	-111	99	12	5	Automobile	Automobile	Life
5	James	A	-1012.34	65535	-119	99	12	6	Automobile	Life	Boat
6	Jason	A	1312.34	15	111	99	12	10	House	Life	Motorcycle
7	John	A	112.34	15	111	99	12	4	House	Boat	Automobile
8	Larry	A	12.34	15	111	99	12	1	Automobile	Life	House
9	Luke	A	-112.34	65535	-119	99	12	3	House	Life	Life
10	Maggie	B	1012.34	-1	-111	99	12	8	Automobile	House	Life

手順 2. ユーザーイグジットプログラムの作成およびコンパイル

この手順では、UCPEC ユーザーイグジットプログラムをコピーして変更します。そのユーザーイグジットプログラムをコンパイルし、PowerExchange リスナ LOADLIB ライブラリ内へとプログラムをリンク編集します。

PowerExchange では、UCPEC プログラムを含むサンプルのユーザーイグジットプログラムが、z/OS 上のインストールデータセットの SRCLIB ライブラリに用意されています。

1. z/OS 上の PowerExchange のインストールデータセットにある SRCLIB ライブラリで、UCPEC ユーザーイグジットプログラムをコピーして、そのプログラムに UCPGCLSC という名前を付けます。
2. UCPGCLSC プログラムを編集し、次のコードで太字で示した変更を加えます。

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. UCPGCLSC.  
*****  
* GLOBAL CUSTOMER SUPPORT SAMPLE CLASS TEST  
* EXAMPLE COBOL PROGRAM CALLED VIA CALLPROG.  
*  
*****          *****  
* USER EXITS ARE NOT SUPPORTED BY INFORMATICA  
* USER EXITS ARE USED AT THE CUSTOMERS OWN RISK  
*  
*  
* USING SYNTAX :-  
* CALLPROG('UCPGCLSC','UCPGCLSC','COBOL','VOID',  
*          TEXT_FIELD,NUMBER1_FIELD)  
*  
* RECEIVES THE FOLLOWING ARGUMENTS :-  
* 1. NUMBER-ARGUMENTS - REQUIRED  
*   THE NUMBER OF ARGUMENTS WHICH FOLLOW.  
*   THE PROGRAM WILL EXIT SETTING A BAD RETURN CODE  
*   IF THE NUMBER IS NOT WHAT IT EXPECTS.  
*  
* 2. FAILURE-CODE. - REQUIRED  
*   AN INTEGER PASSED BACK TO THE CALLER TO INDICATE IF  
*   PROCESSING WAS NOT SUCCESSFUL.  
*   THE FAILURE-CODE IS MONITORED  
*   SO THAT ACTION CAN BE TAKEN TO HANDLE ERRORS.  
*  
*   BECAUSE THE MVS COBOL LINKAGE TYPE ONLY SUPPORTS A  
*   RETURN TYPE OF 'VOID', IT IS NECESSARY TO PASS IT  
*   BACK AS A NORMAL FIELD WITH ITS ACCOMPANYING LENGTH.  
*   (SEE CLLPRGL2 FOR HOW THE RETURN CODE CAN BE PASSED  
*   USING A LINKAGE TYPE OF 'OS' RETURNING 'INT')  
*  
* 3. MESSAGE-BUFFER. - REQUIRED  
*   AN ERROR INTO WHICH THE PROGRAM CAN PUT A MESSAGE  
*   TO ACCOMPANY A NON-ZERO FAILURE CODE, INDICATING
```

```

*      THE REASON.
*
* 4. MESSAGE-BUFFER-LENGTH. - REQUIRED
*      THE LENGTH OF MESSAGE-BUFFER
*
* 5. TEXT-AREA.
*      THIS IS THE 5TH ARGUMENT TO CALLPROG DEFINED IN THE
*      NAVIGATOR EXPRESSIONS SCREEN.
*      IN THIS EXAMPLE, IT IS A FIELD CONTAINING A MAX OF 15 BYTES
*
* 6. TEXT-AREA-LENGTH.
*      THE LENGTH OF TEXT-AREA WHICH VARIES ACCORDING
*      TO THE ACTUAL FIELD LENGTH ON THE FILE.
*
* 7. CLASS-TYPE.
*      THIS IS THE 6TH ARGUMENT TO CALLPROG DEFINED IN THE
*      NAVIGATOR EXPRESSIONS SCREEN.
*      IN THIS EXAMPLE, IT IS A 1 BYTE CHARACTER FIELD WITH VALUES
*      S=SPACES, L=LOW-VALUES, H=HIGH-VALUES, A=ALPHABETIC, N=NUMERIC
*
* 8. CLASS-TYPE-LENGTH.
*      THE LENGTH OF FIELD CLASS-TYPE WHICH WILL ALWAYS
*      BE 1.
*****
*
* ENVIRONMENT DIVISION.
*
* DATA DIVISION.
* WORKING-STORAGE SECTION.
*
01  WS-DATE      PIC X(6).
01  WS-TIME      PIC X(8).
01  WS-DATA.
05  WS-DATA15    PIC X(15).
01  WS-DATAL14  REDEFINES WS-DATA.
05  WS-DATA14    PIC X(14).
05  FILLER      PIC X(01).
01  WS-DATAL13  REDEFINES WS-DATA.
05  WS-DATA13    PIC X(13).
05  FILLER      PIC X(02).
01  WS-DATAL12  REDEFINES WS-DATA.
05  WS-DATA12    PIC X(12).
05  FILLER      PIC X(03).
01  WS-DATAL11  REDEFINES WS-DATA.
05  WS-DATA11    PIC X(11).
05  FILLER      PIC X(04).
01  WS-DATAL10  REDEFINES WS-DATA.
05  WS-DATA10    PIC X(10).
05  FILLER      PIC X(05).
01  WS-DATAL09  REDEFINES WS-DATA.
05  WS-DATA09    PIC X(09).
05  FILLER      PIC X(06).
01  WS-DATAL08  REDEFINES WS-DATA.
05  WS-DATA08    PIC X(08).
05  FILLER      PIC X(07).
01  WS-DATAL07  REDEFINES WS-DATA.
05  WS-DATA07    PIC X(07).
05  FILLER      PIC X(08).
01  WS-DATAL06  REDEFINES WS-DATA.
05  WS-DATA06    PIC X(06).
05  FILLER      PIC X(09).
01  WS-DATAL05  REDEFINES WS-DATA.
05  WS-DATA05    PIC X(05).
05  FILLER      PIC X(10).
01  WS-DATAL04  REDEFINES WS-DATA.
05  WS-DATA04    PIC X(04).
05  FILLER      PIC X(11).
01  WS-DATAL03  REDEFINES WS-DATA.
05  WS-DATA03    PIC X(03).

```



```

05 FILLER      PIC X(12).
01 WS-DATAL02 REDEFINES WS-DATA.
05 WS-DATA02   PIC X(02).
05 FILLER      PIC X(13).
01 WS-DATAL01 REDEFINES WS-DATA.
05 WS-DATA01   PIC X(01).
05 FILLER      PIC X(14).
*
LINKAGE SECTION.

01 LK-NUMBER-ARGUMENTS      PIC S9(9) COMP.

01 LK-FAILURE-CODE         PIC S9(9) COMP.

01 LK-MESSAGE-BUFFER.
05 LK-MESSAGE-BUFFER-BYTE PIC X(1)
    OCCURS 1 TO 255
    DEPENDING ON LK-MESSAGE-BUFFER-LENGTH.
01 LK-MESSAGE-BUFFER-LENGTH PIC S9(9) COMP.

01 LK-TEXT-AREA.
05 LK-TEXT-AREA-BYTE       PIC X(1) OCCURS 15.
01 LK-TEXT-AREA-LENGTH     PIC S9(9) COMP.

01 LK-CLASS                PIC X.
01 LK-CLASS-LENGTH         PIC S9(9) COMP.

PROCEDURE DIVISION USING
    LK-NUMBER-ARGUMENTS
    LK-FAILURE-CODE
    LK-MESSAGE-BUFFER
    LK-MESSAGE-BUFFER-LENGTH
    LK-TEXT-AREA
    LK-TEXT-AREA-LENGTH
    LK-CLASS
    LK-CLASS-LENGTH
.

MAIN SECTION.
0100-MAIN.

    MOVE ZERO TO LK-FAILURE-CODE.
    MOVE ZERO TO LK-MESSAGE-BUFFER-LENGTH.

*****
* EXIT FLAGGING AN ERROR IF THE WRONG NUMBER OF ARGUMENT PAIRS
*****
    IF LK-NUMBER-ARGUMENTS NOT = 2
        DISPLAY 'UCPGCLSC:NUMBER-ARGUMENTS=' LK-NUMBER-ARGUMENTS
            ' (REQUIRED 2)'
            ' EXITTING WITH RC=401'
        MOVE 401 TO LK-FAILURE-CODE
        MOVE 'UCPGCLSC:NOT ENOUGH ARGUMENTS '
            TO LK-MESSAGE-BUFFER
        GO TO 0900-MAIN-EXIT
    END-IF.

*****
* IF DATA PRESENT FIND LENGTH AND TEST CLASS
*****
*
    IF LK-TEXT-AREA-LENGTH = ZERO
        MOVE 'FIELD EMPTY' TO LK-MESSAGE-BUFFER
        MOVE 11 TO LK-MESSAGE-BUFFER-LENGTH
        MOVE 402 TO LK-FAILURE-CODE
        GO TO 0900-MAIN-EXIT
    ELSE
        IF LK-TEXT-AREA-LENGTH > +15
            MOVE 'LENGTH > 15' TO LK-MESSAGE-BUFFER

```

```

        MOVE 11 TO LK-MESSAGE-BUFFER-LENGTH
        MOVE 403 TO LK-FAILURE-CODE
        GO TO 0900-MAIN-EXIT
    ELSE
        MOVE LK-TEXT-AREA TO WS-DATA
    END-IF.
*
    MOVE SPACES TO LK-CLASS.
    MOVE +1 TO LK-CLASS-LENGTH.
*
0100-CLASS15.
*
    IF LK-TEXT-AREA-LENGTH < +15
        GO TO 0100-CLASS14
    END-IF.
*
    IF WS-DATA ALPHABETIC
        MOVE 'A' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA NUMERIC
        MOVE 'N' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA = LOW-VALUES
        MOVE 'L' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA = HIGH-VALUES
        MOVE 'H' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA = SPACES
        MOVE 'S' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
    GO TO 0900-MAIN-EXIT.
*
0100-CLASS14.
*
    IF LK-TEXT-AREA-LENGTH < +14
        GO TO 0100-CLASS13
    END-IF.
*
    IF WS-DATA14 ALPHABETIC
        MOVE 'A' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA14 NUMERIC
        MOVE 'N' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA14 = LOW-VALUES
        MOVE 'L' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA14 = HIGH-VALUES
        MOVE 'H' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA14 = SPACES

```

```

        MOVE 'S' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
    GO TO 0900-MAIN-EXIT.
*
0100-CLASS13.
*
    IF LK-TEXT-AREA-LENGTH < +13
        GO TO 0100-CLASS12
    END-IF.
*
    IF WS-DATA13 ALPHABETIC
        MOVE 'A' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA13 NUMERIC
        MOVE 'N' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA13 = LOW-VALUES
        MOVE 'L' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA13 = HIGH-VALUES
        MOVE 'H' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA13 = SPACES
        MOVE 'S' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
    GO TO 0900-MAIN-EXIT.
*
0100-CLASS12.
*
    IF LK-TEXT-AREA-LENGTH < +12
        GO TO 0100-CLASS11
    END-IF.
*
    IF WS-DATA12 ALPHABETIC
        MOVE 'A' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA12 NUMERIC
        MOVE 'N' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA12 = LOW-VALUES
        MOVE 'L' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA12 = HIGH-VALUES
        MOVE 'H' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
*
    IF WS-DATA12 = SPACES
        MOVE 'S' TO LK-CLASS
        GO TO 0900-MAIN-EXIT
    END-IF.
    GO TO 0900-MAIN-EXIT.
*

```

```

0100-CLASS11.
*
  IF LK-TEXT-AREA-LENGTH < +11
    GO TO 0100-CLASS10
  END-IF.
*
  IF WS-DATA11 ALPHABETIC
    MOVE 'A' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA11 NUMERIC
    MOVE 'N' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA11 = LOW-VALUES
    MOVE 'L' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA11 = HIGH-VALUES
    MOVE 'H' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA11 = SPACES
    MOVE 'S' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
  GO TO 0900-MAIN-EXIT.
*
0100-CLASS10.
*
  IF LK-TEXT-AREA-LENGTH < +10
    GO TO 0100-CLASS09
  END-IF.
*
  IF WS-DATA10 ALPHABETIC
    MOVE 'A' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA10 NUMERIC
    MOVE 'N' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA10 = LOW-VALUES
    MOVE 'L' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA10 = HIGH-VALUES
    MOVE 'H' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA10 = SPACES
    MOVE 'S' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
  GO TO 0900-MAIN-EXIT.
*
0100-CLASS09.
*
  IF LK-TEXT-AREA-LENGTH < +9
    GO TO 0100-CLASS08
  END-IF.

```

```

*
IF WS-DATA09 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA09 NUMERIC
  MOVE 'N' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA09 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA09 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA09 = SPACES
  MOVE 'S' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
GO TO 0900-MAIN-EXIT.
*
0100-CLASS08.
*
IF LK-TEXT-AREA-LENGTH < +8
  GO TO 0100-CLASS07
END-IF.
*
IF WS-DATA08 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA08 NUMERIC
  MOVE 'N' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA08 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA08 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA08 = SPACES
  MOVE 'S' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
GO TO 0900-MAIN-EXIT.
*
0100-CLASS07.
*
IF LK-TEXT-AREA-LENGTH < +7
  GO TO 0100-CLASS06
END-IF.
*
IF WS-DATA07 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.

```

```

*
  IF WS-DATA07 NUMERIC
    MOVE 'N' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA07 = LOW-VALUES
    MOVE 'L' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA07 = HIGH-VALUES
    MOVE 'H' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA07 = SPACES
    MOVE 'S' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
  GO TO 0900-MAIN-EXIT.
*
0100-CLASS06.
*
  IF LK-TEXT-AREA-LENGTH < +6
    GO TO 0100-CLASS05
  END-IF.
*
  IF WS-DATA06 ALPHABETIC
    MOVE 'A' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA06 NUMERIC
    MOVE 'N' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA06 = LOW-VALUES
    MOVE 'L' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA06 = HIGH-VALUES
    MOVE 'H' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA06 = SPACES
    MOVE 'S' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
  GO TO 0900-MAIN-EXIT.
*
0100-CLASS05.
*
  IF LK-TEXT-AREA-LENGTH < +5
    GO TO 0100-CLASS04
  END-IF.
*
  IF WS-DATA05 ALPHABETIC
    MOVE 'A' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
*
  IF WS-DATA05 NUMERIC
    MOVE 'N' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.

```

```

*
IF WS-DATA05 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA05 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA05 = SPACES
  MOVE 'S' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
GO TO 0900-MAIN-EXIT.
*
0100-CLASS04.
*
IF LK-TEXT-AREA-LENGTH < +4
  GO TO 0100-CLASS03
END-IF.
*
IF WS-DATA04 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA04 NUMERIC
  MOVE 'N' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA04 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA04 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA04 = SPACES
  MOVE 'S' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
GO TO 0900-MAIN-EXIT.
*
0100-CLASS03.
*
IF LK-TEXT-AREA-LENGTH < +3
  GO TO 0100-CLASS02
END-IF.
*
IF WS-DATA03 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA03 NUMERIC
  MOVE 'N' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA03 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.

```

```

*
IF WS-DATA03 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA03 = SPACES
  MOVE 'S' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
GO TO 0900-MAIN-EXIT.
*
0100-CLASS02.
*
IF LK-TEXT-AREA-LENGTH < +2
  GO TO 0100-CLASS01
END-IF.
*
IF WS-DATA02 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA02 NUMERIC
  MOVE 'N' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA02 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA02 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA02 = SPACES
  MOVE 'S' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
GO TO 0900-MAIN-EXIT.
*
0100-CLASS01.
*
IF LK-TEXT-AREA-LENGTH < +1
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA01 ALPHABETIC
  MOVE 'A' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA01 NUMERIC
  MOVE 'N' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA01 = LOW-VALUES
  MOVE 'L' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.
*
IF WS-DATA01 = HIGH-VALUES
  MOVE 'H' TO LK-CLASS
  GO TO 0900-MAIN-EXIT
END-IF.

```



```

*
  IF WS-DATA01 = SPACES
    MOVE 'S' TO LK-CLASS
    GO TO 0900-MAIN-EXIT
  END-IF.
  GO TO 0900-MAIN-EXIT.
*
*
0900-MAIN-EXIT.
GOBACK.

```

3. UCPGCLSC プログラムをコンパイルし、PowerExchange リスナ LOADLIB ライブラリ内へとプログラムをリンク編集します。

手順 3。ユーザー定義フィールドの追加

この手順では、ユーザー定義フィールドを追加します。

ユーザー定義フィールドによって PowerExchange CallProg 関数が呼び出され、この関数によってユーザー出口プログラムが呼び出されます。CallProg 関数の詳細については、[「CallProg」 \(ページ 304\)](#)を参照してください。

ユーザー出口プログラムは、データを処理して REC_TYPE フィールド、BIN_NO フィールド、および DECIMAL_NO フィールドのクラスタイプを返します。

1. demo.userexit データマップと MASTER_REC レコードを開きます。
2. **レコード**ウィンドウで、**[Expr(0)]** タブをクリックします。
3. **[Expr(0)]** タブの任意の場所で右クリックし、**[最後にフィールドを追加]** をクリックします。
4. REC_TYPE フィールド用に呼び出されたユーザー出口プログラムからの結果を示す出力フィールドである classtype_rec_type フィールドを追加します。

以下の表に、そのフィールドを定義するプロパティを示します。

プロパティ	値
名前	classtype_rec_type
タイプ	CHAR
精度	0
スケール	0
長さ	1

5. **[Expr(0)]** タブの任意の場所で右クリックし、**[最後にフィールドを追加]** をクリックします。
6. REC_TYPE フィールドを処理するためにユーザー出口プログラムを呼び出す rc_rec_type フィールドを追加します。

以下の表に、そのフィールドを定義するプロパティを示します。

プロパティ	値	注意事項
名前	rc_rec_type	-
タイプ	NUM32	-
精度	0	-
スケール	0	-
長さ	0	-
フェーズ	RW	処理が読み取りまたは書き込みであることを示します。
式	CallProg('UCPGCLSC', 'UCPGCLSC','COBOL', REC_TYPE,classtype_rec_type)	<p>フィールドに式を入力するには、次の手順を実行します。</p> <ol style="list-style-type: none"> 1. 【式】 列のセルをクリックし、【参照】 ボタンをクリックします。【式エディタ】 ダイアログボックスが表示されます。 2. 【式エディタ】 ダイアログボックスの【関数リスト】 リストで、CallProg 関数をダブルクリックします。 3. 【式リスト】 リストで、CallProg 関数名の最後に 「('UCPGCLSC','UCPGCLSC','COBOL',REC_TYPE,classtype_rec_type)」 と入力します。 4. 【検証】 をクリックします。【検証】 ボックスに、[エラーはありません]メッセージが表示されます。 5. 【OK】 をクリックします。

7. レコードウィンドウで、**【Expr(0)】** タブをクリックします。
8. **【Expr(0)】** タブの任意の場所で右クリックし、**【最後にフィールドを追加】** をクリックします。
9. BIN_NO フィールド用に呼び出されたユーザー出口プログラムからの結果を示す出力フィールドである classtype_bin_no フィールドを追加します。

以下の表に、そのフィールドを定義するプロパティを示します。

プロパティ	値
名前	classtype_bin_no
タイプ	CHAR
精度	0
スケール	0
長さ	1

10. **[Expr(0)]** タブの任意の場所で右クリックし、**[最後にフィールドを追加]** をクリックします。
11. BIN_NO フィールドを処理するためにユーザー出口プログラムを呼び出す rc_bin_no フィールドを追加します。

以下の表に、そのフィールドを定義するプロパティを示します。

プロパティ	値	注意事項
名前	rc_bin_no	-
タイプ	NUM32	-
精度	0	-
スケール	0	-
長さ	0	-

プロパティ	値	注意事項
フェーズ	RW	処理が読み取りまたは書き込みであることを示します。
式	CallProg('UCPGCLSC', 'UCPGCLSC','COBOL', BIN_NO,classtype_bin_no)	<p>フィールドに式を入力するには、次の手順を実行します。</p> <ol style="list-style-type: none"> 1. [式] 列のセルをクリックし、[参照] ボタンをクリックします。[式エディタ] ダイアログボックスが表示されます。 2. [式エディタ] ダイアログボックスの[関数リスト] リストで、CallProg 関数をダブルクリックします。 3. [式リスト] リストで、CallProg 関数名の最後に「('UCPGCLSC','UCPGCLSC','COBOL',BIN_NO,classtype_bin_no)」と入力します。 4. [検証] をクリックします。[検証] ボックスに、[エラーはありません]メッセージが表示されます。 5. [OK] をクリックします。

12. **[Expr(0)]** タブの任意の場所で右クリックし、**[最後にフィールドを追加]** をクリックします。
13. DECIMAL_NO フィールド用に呼び出されたユーザー出口プログラムからの結果を示す出力フィールドである classtype_dec_no フィールドを追加します。
以下の表に、そのフィールドを定義するプロパティを示します。

プロパティ	値
名前	classtype_dec_no
タイプ	CHAR
精度	0
スケール	0
長さ	1

14. **[Expr(0)]** タブの任意の場所で右クリックし、**[最後にフィールドを追加]** をクリックします。
15. DECIMAL_NO フィールドのコピーを処理するためにユーザー出口プログラムを呼び出す rc_decimal_no フィールドを追加します。

以下の表に、そのフィールドを定義するプロパティを示します。

プロパティ	値	注意事項
名前	rc_decimal_no	-
タイプ	NUM32	-
精度	0	-
スケール	0	-
長さ	0	-
フェーズ	RW	処理が読み取りまたは書き込みであることを示します。
式	CallProg('UCPGCLSC', 'UCPGCLSC','COBOL', DECIMAL_NO,classtype_dec_no)	<p>フィールドに式を入力するには、次の手順を実行します。</p> <ol style="list-style-type: none"> 1. [式] 列のセルをクリックし、[参照] ボタンをクリックします。[式エディタ] ダイアログボックスが表示されます。 2. [式エディタ] ダイアログボックスの[関数リスト] リストで、CallProg 関数をダブルクリックします。 3. [式リスト] リストで、CallProg 関数名の最後に「('UCPGCLSC','UCPGCLSC','COBOL',DECIMAL_NO,classtype_dec_no)」と入力します。 4. [検証] をクリックします。[検証] ボックスに、[エラーはありません]メッセージが表示されます。 5. [OK] をクリックします。

手順 4.MASTER_REC テーブルのカラムのリフレッシュ

この手順では、MASTER_REC テーブルのカラムをリフレッシュして、MASTER_REC レコードに追加したユーザー定義フィールドを取得します。

1. demo.userexit データマップを開きます。
2. リソースエクスプローラの**[データマップ]** タブで、MASTER_REC テーブルを右クリックして**[プロパティ]** をクリックします。
[テーブルのプロパティ - 定義] ダイアログボックスが表示されます。
3. **[カラムの生成]** リストから、**[不足しているカラムを使用してリフレッシュ]** を選択します。
テーブルの基になっているレコードに新しいフィールドが含まれているため、このアクションによって対応するカラムがテーブルに追加されます。
4. **[OK]** をクリックします。

[テーブルプロパティ - 定義]ダイアログボックスに関する詳細については、「[テーブルプロパティ - 定義](#)」 (ページ 391)を参照してください。

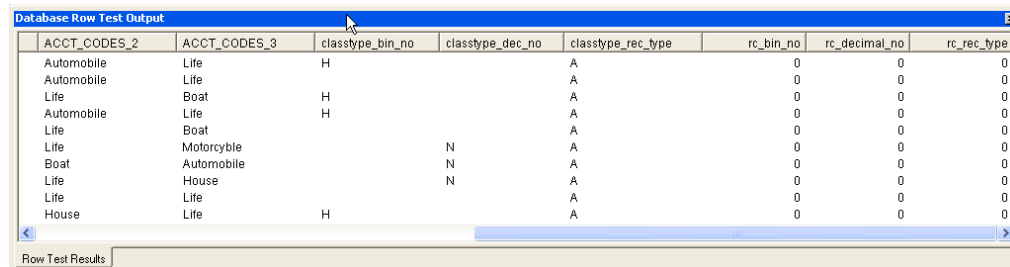
手順 5. ユーザー出口プログラムの結果のテスト

この手順では、データマップレコード内のユーザー出口プログラムの結果をテストするためのデータベース行のテストを実行します。

1. demo.userexit データマップを開きます。
2. リソースエクスプローラの [データマップ] タブで、MASTER_REC テーブルを選択して [ファイル] > [データベース行のテスト] をクリックします。
3. データマップをリモートの場所へ送信するように求められたら、[はい] をクリックします。
[データマップリモートノード] ダイアログボックスが表示されます。
4. [データマップリモートノード] ダイアログボックスで、ユーザーのユーザー ID、パスワード、または有効な PowerExchange パスフレーズおよび z/OS システムのノード名を入力します。
[データベース行のテスト] ダイアログボックスが表示されます。
5. [データベース行のテスト] ダイアログボックスで、デフォルト値を受け入れて [実行] をクリックします。

データベース行のテストによる出力ウィンドウに、ユーザー定義フィールドを含むデータベース行のテストの結果が表示されます。

以下の画像に、データベース行のテストによる出力ウィンドウを示します。



ACCT_CODES_2	ACCT_CODES_3	classtype_bin_no	classtype_dec_no	classtype_rec_type	rc_bin_no	rc_decimal_no	rc_rec_type
Automobile	Life	H		A	0	0	0
Automobile	Life			A	0	0	0
Life	Boat	H		A	0	0	0
Automobile	Life	H		A	0	0	0
Life	Boat			A	0	0	0
Life	Motorcycle		N	A	0	0	0
Boat	Automobile		N	A	0	0	0
Life	House		N	A	0	0	0
Life	Life			A	0	0	0
House	Life	H		A	0	0	0

以下の表に、ユーザー定義フィールドを示します。

ユーザー定義フィールド	値	説明
classtype_bin_no	H	その行の BIN_NO フィールドに最大値が含まれていることを示します。
classtype_dec_no	N	その行の DECIMAL_NO フィールドにゾーン 10 進数値が含まれていることを示します。
classtype_rec_type	A	すべての行の REC_TYPE フィールドにアルファベットの値が含まれていることを示します。
rc_bin_no rc_decimal_no rc_rec_type	0	すべての行に対して UCPGCLSC ユーザー出口プログラムが正常に実行されたことを示します。

付録 A

ユーザー定義フィールド用 PowerExchange 関数

この付録では、以下の項目について説明します。

- [ユーザー定義フィールド用 PowerExchange 関数の概要, 303 ページ](#)
- [PowerExchange 関数, 304 ページ](#)
- [CallProg Function 関数によるユーザー定義プログラムの呼び出し, 329 ページ](#)
- [データマップのユーザー定義フィールドのテスト, 338 ページ](#)

ユーザー定義フィールド用 PowerExchange 関数の概要

データマップでは、ソースデータの処理に PowerExchange 関数を式の中で 1 つ以上使用するレコードのフィールドを定義することができます。

これらの関数により、フィールドの分割や連結などの操作を行うことができます。また、ユーザー定義プログラムを呼び出す式で C allProg 関数を使用して、ソースデータを処理することもできます。

PowerExchange は、代表的なユーザー定義プログラムをヘッダファイルおよび JCL と共に提供しています。ucpe*. *と名付けられたこれらのファイルは、PowerExchange Navigator インストレーションディレクトリのサンプルディレクトリにあります。これらのサンプルプログラムを編集およびコンパイルし、使用環境に応じてカスタマイズします。

ユーザー定義フィールドを定義した後、データベース行のテストを実行して、フィールドに定義されている PowerExchange 関数およびユーザー定義関数をテストします。

ユーザー定義フィールドに対する操作前の画像と変更インジケータのカラムを生成する場合は、そのフィールドに関連付けられている関数が操作前の画像のバッファをサポートしているか確認してください。その関数が操作前の画像のバッファをサポートしていない場合、操作前の画像と変更インジケータは生成されません。

関連項目：

- [「データマップのユーザー定義フィールドのテスト」 \(ページ 338\)](#)

PowerExchange 関数

データマップ内のレコードのソースデータを処理するには、ユーザー定義フィールドの式に次の PowerExchange 関数を含めます。

CallProg

ユーザー定義プログラムまたはサブルーチンを呼び出して、レコードのソースデータを処理します。

構文:

```
[result=]CallProg('program','subroutine','linkage'[,arg1][,arg2][,...])
```

パラメータは次のとおりです。

- **result**。オプションです。NUM32。この引数には、CallProg 関数で呼び出されたプログラムの戻り値が入ります。値は次のうちの 1 つになります。
 - **0**。成功
 - **ゼロ以外**。失敗です。
結果引数を指定せず、外部プログラムからゼロ以外の戻りコードが返された場合は、CallProg により次のデフォルトのマップレベルのエラー応答が実行されます。
 - 抽出を終了します。
 - このサブルーチンをスキップします。
- **program**。サブルーチンが含まれるプログラムの名前です。オペレーティングシステムに応じて、プログラムは次のうちの 1 つになります。
 - **i5/OS**。サービスプログラムです。
 - **Linux または UNIX**。共有オブジェクトです。
 - **Windows**。DLL です。
 - **z/OS**。ロードモジュールです。
プログラム名は、一重引用符で囲みます。
- **subroutine**。プログラムのエントリポイントの名前です。オペレーティングシステムに応じて、サブルーチンは次のうちの 1 つになります。
 - **i5/OS**。サブルーチン名です。
 - **Linux、UNIX、または Windows**。サブルーチン名です。
 - **z/OS (Assembler、C、または COBOL プログラムの場合)**。値を指定する必要がありますが、その値は無視され、ロードモジュールのデフォルトのエントリポイントが使用されます。プログラムと同じ名前を指定します。
 - **z/OS (PL/I プログラムの場合)**。同じロードモジュールに複数のフェッチ可能なサブルーチンが存在する場合に、サブルーチンの名前を指定します。
サブルーチン名は、一重引用符で囲みます。
- **linkage**。リンケージの型です。これによって、引数がプログラムまたはサブルーチンに渡される方法、および戻りコードがプログラムまたはサブルーチンから返される方法が決まります。

以下の表に、各リンケージ型を示します。

リンケージ型	サポートされるオペレーティングシステム	引数	戻り値
C	- i5/OS - Linux、UNIX、および Windows - z/OS	スタックを介して渡される	プログラムの戻りコード
COBOL	z/OS	アドレスのリストとして渡される	エラーコード整数のアドレス
OS	z/OS	アドレスのリストとして渡される	プログラムの戻りコード
OS400	i5/OS	アドレスのリストとして渡される	エラーコード整数のアドレス
PLI	z/OS	アドレスのリストとして渡される	エラーコード整数のアドレス

リンケージ型は、一重引用符で囲みます。

- `[,arg1][,arg2][,...]`。プログラムまたはサブルーチンに渡される 1 つ以上のオプションの引数です。

関連項目：

- [「CallProg Function 関数によるユーザー定義プログラムの呼び出し」 \(ページ 329\)](#)

Check

CallProg 関数で呼び出されたプログラムの戻りコードによって成功か失敗がわかるように指定することができます。

CallProg 関数で呼び出されたプログラムの戻りコードと、指定された 1 つ以上の戻りコードを比較します。一致した場合は、ユーザー定義メッセージが PowerExchange メッセージログに書き込まれます。

構文:

```
Check(CallProgRC, 'message', "comparison", return_code1, return_code2, return_code3)
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
CallProgRC	CallProg 関数で呼び出されたプログラムの戻りコードです。この名前は、CallProg 関数の結果引数に指定されている名前と一致している必要があります。Check 関数では、この戻りコードと指定された戻りコード値を比較します。	NUM32
メッセージ	実行される比較のタイプに応じて、戻りコードがリスト内の指定戻りコードのいずれかに一致した場合、あるいは一致しなかった場合に、PowerExchange メッセージログに書き込まれるユーザー定義メッセージです。メッセージテキストは、一重引用符で囲みます。	-

パラメータ	説明	データ型
比較	<p>実行される比較のタイプです。次のオプションがあります。</p> <ul style="list-style-type: none"> - EQ。戻りコードがリスト内の指定戻りコードのいずれかと一致した場合にエラーが発生します。 - NE。戻りコードがリスト内の指定戻りコードのいずれかと一致しない場合にエラーが発生します。 <p>比較する値を引用符で囲みます。</p>	-
return_code1 return_code2 return_code3	<p>1 つ以上の戻りコード値のリストです。Check 関数では、CallProg 関数で呼び出されたプログラムの戻りコードとこの戻りコード値のリストを比較します。最低 1 つの戻りコード値を指定する必要があります。</p>	NUM32

Check の例

0、3、および 4 の戻りコードが Program1 プログラムの Program2 サブルーチンの許容可能な戻りコードである場合は、次の関数のように指定できます。

```
CallProgRC=CallProg('Program1','Program2','COBOL','field1','field2')
Check(CallProgRC,'Error in Program2','NE',0,3,4)
```

Program2 サブルーチンの戻りコードが 1 である場合は、処理が失敗し、次のメッセージがログに書き込まれます。

Error in Program2

Program2 サブルーチンの戻りコードが 0 である場合は、処理は続行され、指定されたメッセージはログに書き込まれません。

CheckNumData

フィールド型が CHAR、NUMCHAR、ZONED、PACKED のフィールドに有効な数字が含まれているかどうかを呼び出し元プログラムでチェックできるようにします。無効なデータが検出された場合は、対応するアクションが実行されます。

CheckNumData は、データマッププロパティのデータチェックオプションと同様の関数を実行します。CheckNumData では、データマップ全体ではなく単一のフィールドにデータチェックを適用します。CheckNumData を使用することで、CopyData などの関数によって変換処理が実行される前に数値データフィールドを検証できます。

構文:

```
CheckNumData(field_name,field_type,action)
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明
<i>field_name</i>	有効な数値データかをチェックするフィールドの名前。
<i>field_type</i>	CheckNumData でチェックするフィールドのデータ型。次の型が有効です。 <ul style="list-style-type: none">- CHAR- NUMCHAR- ZONED- PACKED
<i>action</i>	フィールドに指定されたデータ型に対して無効なデータが検出された場合に実行するアクション。次のオプションがあります。 <ul style="list-style-type: none">- REPLACE。フィールド値を有効な特定の値で置き換えます。- SETNULL。フィールド値を、ターゲットに対して有効な NULL 文字で置き換えます。- SKIP。ターゲットマッピングのフィールドに無効なデータが含まれている場合は、そのフィールドをスキップします。- TERMINATE。無効なデータが検出された場合は、マッピング処理を終了します。

Concat

2 つ以上のフィールドを連結します。この関数を使用して、複数のフィールドから 1 つのグループフィールドを作成できます。

この関数によって、フィールド内のデータ上の変換が実行されることはありません。コピーされるバイト数はフィールドの長さによって決まります。

注: 同様のデータ型のフィールドのみを連結できます。例えば、文字データ型のフィールドと数値データ型のフィールドを連結することはできません。

構文:

result=Concat(*field1*,*field2*[,...])

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	必須。連結された入力フィールドが入力されるユーザー定義式フィールドです。連結されたフィールドを格納できるだけの長さが必要です。	すべて
<i>field1</i>	1 つ以上のフィールドが連結されるフィールドです。	任意のデータ型の実フィールドまたはユーザー定義フィールド、あるいは一重引用符で囲んだ値です。
<i>field2</i> [,...]	最初のフィールドに連結される 1 つ以上のフィールドです。	任意のデータ型の実フィールドまたはユーザー定義フィールド、あるいは一重引用符で囲んだ値です。

Concat の例

Field1 というフィールドに次の文字が格納されている場合

The quick brown fox

そして、Field2 というフィールドに次の文字が格納されている場合

jumped over the lazy dog

関数呼び出し Concat(Field1,Field2)によって次の文字が返されます。

The quick brown fox jumped over the lazy dog

CopyData

あるフィールドから別のフィールドへデータをコピーします。ソースフィールドをターゲットフィールド、またはこの関数を含む式に指定されたユーザー定義フィールドにコピーできます。

構文:

データを特定のターゲットフィールドにコピーする手順

`CopyData(source_field,target_field)`

データを式に指定されたユーザー定義フィールドにコピーする手順

`CopyData(source_field)`

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>source_field</i>	コピーするフィールドです。このフィールドには、実際のフィールドまたは任意のデータタイプのユーザー定義フィールド、または一重引用符で囲んでコピーするデータ値 ('123456'など) を指定できます。	任意です。 データタイプ
<i>target_field</i>	ソースフィールドのデータをコピーするターゲットフィールドです。ターゲットフィールドの長さは、ソースフィールドの長さ以上である必要があります。そうしないと、変換切り捨てエラーが発生します。 注: また、CopyData 式でターゲットフィールドを指定する場合は、[Expr(<i>n</i>)] タブでフィールド名と属性を定義しないでください。	任意です。 データタイプ

使用上の注意:

- データを特定のターゲットフィールドにコピーする場合は、[レコード] ウィンドウの [Expr(*n*)] タブで、この関数を含む式行の [名前] 値およびフィールド属性を空白のままにします。そうしないと、CopyData は RC=2660 で失敗します。
- 式に関連付けられているユーザー定義フィールドにデータをコピーする場合は、[レコード] ウィンドウの [Expr(*n*)] タブで、この関数を含む式行の [名前] 値およびフィールド属性を入力します。フィールド長が、コピーしたデータを含められるだけ十分に長いことを確認します。任意の適切なデータ型を使用します。関数では、ユーザー定義フィールドに、ソースフィールドのデータ型の代わりにこのデータ型を使用します。複数のデータ型で定義された非リレーショナルソースフィールドの場合は、式を使用して、それぞれが別のデータ型を持つ別個のユーザー定義フィールドにデータを書き込むことができます。

Fragment

ソースフィールドのサブフィールドを指定された 2 つ以上のターゲットフィールドに分割および分散します。

各ターゲットフィールドに分散されるデータの量は、ターゲットフィールドの長さによって決まります。例えば、長さが 2 文字の CHAR フィールドが最初のターゲットフィールドである場合、ソースフィールドの最初の 2 文字がそのターゲットフィールドにコピーされます。そして、長さが 8 文字の CHAR フィールドが 2 番目のターゲットフィールドである場合、ソースフィールドの次の 8 文字がそのターゲットフィールドにコピーされます。

構文:

[result]=Fragment(source_field,target_field1,target_field2[,...])

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
result	任意です。次のうち 1 つの値が返されます。 <ul style="list-style-type: none">- 0. ソースフィールドは、1 つ以上のターゲットフィールドに正常に分割されました。- 1. ソースフィールドの一部のデータはターゲットフィールドに分散されませんでした。- 2. 最後のターゲットフィールドが変数データタイプで、その最大長までデータが入力されませんでした。- 3. 一部のターゲットフィールドにデータが入力されず、NULL が設定されました。	-
source_field	分割されるフィールドです。	任意のデータタイプの実フィールドまたはユーザー定義フィールド、あるいは一重引用符で囲んだ値です。
target_field1	ソースフィールドからの断片のコピー先となるターゲットフィールドです。	変数データタイプを除く、任意のデータタイプのユーザー定義フィールドです。
target_field2 [,...]	ソースフィールドのフラグメントをコピーする 1 つ以上の追加のターゲットフィールドです。	変数データタイプを除く、任意のデータタイプのユーザー定義フィールドです。最後のターゲットフィールドは変数データタイプにもなります。 最後のターゲットフィールドが変数ではなく、その長さがソースフィールドの残りのデータの長さを超えている場合、ターゲットフィールドには NULL が入力され、戻りコードは 0 になります。ただし、最後のターゲットフィールドを変数にした場合は、データが入力されます。変数フィールドの全長にデータが入力されていない場合であっても、戻りコードは 0 になります。

Fragment の例

次のフィールドが定義されている場合

SourceField CHAR (44)
TargetField1 CHAR (16)
TargetField2 CHAR (28)

そして、SourceField というフィールドに次の文字が入力されている場合

The quick brown fox jumped over the lazy dog

関数呼び出し Fragment (SourceField、TargetField1、TargetField2) によって、ソースフィールドのフラグメントが TargetField1 フィールドと TargetField2 フィールドに分散されます。

TargetField1 フィールドの内容は次のとおりです。

The quick brown

TargetField2 フィールドの内容は次のとおりです。

fox jumped over the lazy dog

GenVRowKey

配列が含まれるソース行を基に作成された行から、一意の行番号によって特定される複数のテーブル行を生成します。

この関数によって、単一のソース行から作成された複数のターゲット行を区別することができます。例えば、ソースファイルに配列が含まれていて、その配列から子テーブルが作成されている場合、この関数を使用して、通常は同一である行の一意の複数のテーブル行がターゲットフィールドに入力されます。

注: OCCURS 句があるデータ行では、OCCURS の各インスタンスに対して 1 つの行が返されます。

構文:

result=GenVRowKey()

戻り値:

NUM32U。配列が含まれるソース行を基に作成された行の一意の複数のテーブル行です。

GenVRowKey の例

次のように、COBOL OCCURS 句で定義された配列のあるレコードが含まれたデータマップを定義します。

```
01 RECORD.  
   03 FIELD1 PIC X(4).  
   03 FIELD2 PIC X(4) OCCURS 3 TIMES.
```

このデータマップには、次のデータが格納された 2 つのソース行が含まれます。

```
AAAA111122221111  
BBBB555566667777
```

この例では、2 つの行に次の同一データが含まれます。

```
AAAA 1111
```

GenVRowKey()関数によって次の一意のテーブル行が生成されます。

```
AAAA 1111 1  
AAAA 2222 2  
AAAA 1111 3  
BBBB 5555 1  
BBBB 6666 2  
BBBB 7777 3
```

GetCurrentFileName

非リレーショナルデータソースに定義されているデータマップレコードについて、ソースデータファイルの名前を取得します。

GetCurrentFileName 関数を使用して、レコードのデータが読み取られているデータファイルを決定します。この情報を使用して、次のタスクを実行できます。

- 単一のデータファイルに関連付けられたデータマップの場合は、そのデータファイルのすべてのレコードが処理されたか検証します。
- ファイルリストファイルに関連付けられたデータマップの場合は、そのファイルリストファイルに一覧表示されているすべてのデータファイルの全レコードが処理されたことを検証します。
- 元のデータファイルまで行をトレースします。

この関数は、どのオペレーティングシステムでも実行できます。また、読み取りモードで実行されます。

構文::

```
result=GetCurrentFileName()
```

次の表に、これらのパラメータについて説明します。

結果	説明	データ型
<i>result</i>	レコードのデータが読み取られているデータファイルの名前を返します。 データマップがファイルリストファイルに関連付けられている場合、レコードは複数のデータファイルから読み取られていることがあります。	VARCHAR。最大長は 512 バイト + NULL 終了文字 1 字です。

使用上の注意:

- i5/OS では、ファイルが複数のメンバで構成されていて、ファイル名に*FIRST および*LAST が指定されている場合、GetCurrentFileName 関数によって実際のメンバ名は返されません。GetCurrentFileName 関数により OPEN 関数に渡されたファイル名が返されます。
- Linux、UNIX、および Windows では、.././myfile.dat などの相対ディレクトリパスで修飾されている場合、GetCurrentFileName 関数では完全ファイル名は取得されません。代わりに GetCurrentFileName 関数で次の値が返されます。.././myfile.dat.
- z/OS では次の場合、GetCurrentFileName 関数によりファイルリストファイルに一覧表示されている名前より正確なファイル名が返されることがあります。
 - データマップまたはファイルリストファイルの世代別データグループ (GDG) ファイル名
 - データマップまたはファイルリストファイルの DD 名

関連項目：

- [「データファイルおよびファイルリスト処理」 \(ページ 54\)](#)

GetCurrentFileName の例

次の例では、GetCurrentFileName 関数を使用して、データマップに関連付けられたデータファイルを返す方法を示します。

例 1

c:/Informatica/PowerExchange/examples ディレクトリにある demo3.dat データファイルにデータマップが関連付けられている場合、GetCurrentFileName 関数は、データマップのすべてのレコードに対して次の値を返します。

```
c:/Informatica/PowerExchange/examples/demo3.dat
```

例 2

複数のデータファイルが一覧表示されているファイルリストファイルにデータマップが関連付けられている場合、GetCurrentFileName 関数は、各レコードが読み取られたデータファイルの名前を返します。例えば、10 レコードずつ含まれている次のデータファイルが一覧表示されたファイルリストファイルにデータマップが関連付けられているとします。

```
c:/Informatica/PowerExchange/examples/datafile1.dat  
c:/Informatica/PowerExchange/examples/datafile2.dat
```

GetCurrentFileName 関数により、1~10 のレコードに対して次の値が返されます。

```
c:/Informatica/PowerExchange/examples/datafile1.dat
```

GetCurrentFileName 関数により、11～20 のレコードに対して次の値が返されます。

c:/Informatica/PowerExchange/examples/datafile2.dat

例 3

相対パスが使用されている次のデータファイルがファイルリストファイルに一覧表示されていて、各データファイルに 10 レコード含まれている場合

```
../datafile1.dat  
../datafile2.dat
```

GetCurrentFileName 関数により、1～10 のレコードに対して次の値が返されます。

```
../datafile1.dat
```

GetCurrentFileName 関数により、11～20 のレコードに対して次の値が返されます。

```
../datafile2.dat
```

GetDatabaseKey

レコードまたはセグメントのデータベースキーを取得します。

あるいは、次の方法を使用して、VSAM ESDS データセットの拡張相対バイトアドレス (XRBA)、または VSAM RRDS データセットの相対レコード番号 (RRN) を返すことができます。

- ESDS データセットの XRBA を取得するには、ESDS データマップの【データマップのプロパティ】ダイアログボックスにある【ESDS アクセス方式】タブ上の【XRBA のレコードのプリフィクス】オプションを選択します。
- RRDS データセットの RRN を取得するには、RRDS データマップの【データマップのプロパティ】ダイアログボックスにある【RRDS アクセス方式】タブ上の【RRN のレコードのプリフィクス】オプションを選択します。

構文:

GetDatabaseKey()

または、次のエイリアスを使用します。

GetDbKey()

戻り値:

以下の表に、データマップのソースタイプに基づいて返されるオブジェクトを示します。

アクセス方式	データソース	返されるデータベースキーオブジェクト	データタイプ
ADABAS	Adabas	内部シーケンス番号 (ISN)	NUM32[NUM32]
ESDS[ESDS]	z/OS の VSAM エントリシーケンスのデータセット (ESDS)	拡張相対バイトアドレス (XRBA)	BIN 8[BIN 8]
IDMS	IDMS	データベースキー[データベースキー]	NUM32[NUM32]
DL/1 バッチ IMS ODBA	IMS	相対バイトアドレス (RBA)	BIN 8[BIN 8]

アクセス方式	データソース	返されるデータベースキーオブジェクト	データタイプ
SEQ	順次アクセスする i5/OS ファイル	相対レコード番号 (RRN)	NUM32[NUM32]
RRDS[RRDS]	z/OS 上の VSAM 相対レコードデータセット (RRDS)	相対レコード番号 (RRN)	NUM32[NUM32]

注: IMS RBA の長さは 4 バイトですが、GetDatabaseKey 関数では、要求されたセグメントの RBA 値の 8 バイトバイナリ表現を返します。将来的な拡張を可能にし、FastPath および HALDB データベースの一貫性を保証するために、8 バイトが必要となります。

使用上の注意:

- IMS データマップのセグメントデータにセグメント RBA を追加する場合、Informatica では、GetDatabaseKey 関数ではなく GetIMSRBAByLevel 関数を使用することをお勧めしています。GetIMSRBAByLevel 関数では、キーのない、または一意のキーが存在しない親セグメントの RBA を取得することができます。

バルクデータ移動操作または IMS 同期 CDC に使用される IMS データマップには GetIMSRBAByLevel 関数を使用します。

次のタイプの IMS データマップ内のレコードには、GetIMSRBAByLevel 関数を使用できません。

- IMS ログベース CDC に使用される IMS データマップ
- 次のタイプのレコードが含まれる IMS アンロードファイルへのアクセスに使用される IMS データマップ
 - HISAM データセットの任意のフォーマットのアンロードレコード
 - HDAM および HIDAM データセットの BMC SHORT または BMC XSHORT フォーマットのアンロードレコード
 - IMS 高速パスデータセットの標準 IBM フォーマットのアンロードレコード
- この関数は、ユーザー定義フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

関連項目:

- [「VSAM データセットのレコードに対する RRN または RBA を取得」 \(ページ 140\)](#)
- [「GetIMSRBAByLevel」 \(ページ 319\)](#)

GetDataFlowType

IDMS。データの出所（バルクデータ移動操作、ログキャプチャデータ抽出、またはリアルタイムキャプチャデータ抽出）、およびデータが添付されているかどうかを示す 2 文字のインジケータを返します。

GetDbKeyOfOwner 関数と共に使用され、IDMS データのリレーショナルビューを生成します。これらの関数を使用して、PowerExchange バルクデータ移動操作によってリレーショナルデータベースをマテリアライズすることができます。その後、PowerExchange CDC 処理によってデータを保持することができます。

GetDataFlowType 関数は、次のいずれかと共に動作します。

- 単一レコードのキャプチャ登録
- ベースレコード（つまり最も低いレベル）の複数レコードのキャプチャ登録

GetDbKeyOfOwner 関数では、データベースキーを返し、ページグループおよび基数エレメントをゼロに設定します。

構文:

`result=GetDataFlowType()`

`result` パラメータでは、次の情報を指定する 2 文字の CHAR 値を返します。

- 最初の文字は次のうち 1 つの値となります。
 - **B**。バルクデータ抽出
 - **L**。ログキャプチャデータ抽出
 - **R**。リアルタイムキャプチャデータ抽出です。
- 2 番目の文字は次のうち 1 つの値となります。
 - **D**。通常のデータレコード
 - **C**。データを含まない接続。CDC 処理に関連します。レコードのデータ部分には NULL が含まれます。
 - **U**。データを含まない接続の切断。CDC 処理に関連します。レコードのデータ部分には NULL が含まれます。

オーナーが存在しない場合、`GetDbKeyOfOwner` 関数は大きい値を返します。

GetDbKey

`GetDatabaseKey` のエイリアスです。

関連項目：

- [「GetDatabaseKey」 \(ページ 312\)](#)

GetDbKeyOfFirstMember

IDMS。指定されたセットの最初のメンバのデータベースキーを返します。この関数は、指定されたセットのオーナーであるレコードで使用できます。

この関数は IDMS ページグループまたは基数を返さないため、この関数は、別のページグループを使用してアクセスされる、重複するデータベースが存在しない場合に使用します。重複するデータベースが存在する可能性がある場合は、`GetFullDbKeyOfFirstMember` 関数を使用します。

構文:

`GetDbKeyOfFirstMember('IDMS_set_name')`

この関数は、データ型 NUM32U または BIN4 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<code>IDMS_set_name</code>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetDbKeyOfLastMember

IDMS。指定されたセットの最後のメンバのデータベースキーを返します。この関数は、指定されたセットのオーナーであるレコードで使用できます。

この関数は IDMS ページグループまたは基数を返さないため、この関数は、別のページグループを使用してアクセスされる、重複するデータベースが存在しない場合に使用します。重複するデータベースが存在する可能性がある場合は、GetFullDbKeyOfLastMember 関数を使用します。

構文:

```
GetDbKeyOfLastMember('IDMS_set_name')
```

この関数は、データ型 NUM32U または BIN4 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetDbKeyOfNextMember

IDMS。指定されたセットの次のメンバレコードのデータベースキーを返します。この関数は、指定されたセットのメンバであるレコードで使用できます。

この関数は IDMS ページグループまたは基数を返さないため、この関数は、別のページグループを使用してアクセスされる、重複するデータベースが存在しない場合に使用します。重複するデータベースが存在する可能性がある場合は、GetFullDbKeyOfNextMember 関数を使用します。

構文:

```
GetDbKeyOfNextMember('IDMS_set_name')
```

この関数は、データ型 NUM32U または BIN4 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetDbKeyOfOwner

IDMS。IDMS セット名のオーナーのデータベースキーを取得します。レコードを物理的に読み取ることなくオーナーレコードを特定することができます。

この関数は IDMS ページグループまたは基数を返さないため、この関数は、別のページグループを使用してアクセスされる、重複するデータベースが存在しない場合に使用します。重複するデータベースが存在する可能性がある場合は、GetFullDbKeyOfOwner 関数を使用します。

構文:

```
GetDbKeyOfOwner('IDMS_set_name')
```

この関数は、データ型 NUM32U または BIN4 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

関連項目:

- [「GetFullDbKeyOfOwner」 \(ページ 318\)](#)

GetDbKeyOfPriorMember

IDMS。指定されたセットの前のメンバレコードのデータベースキーを返します。この関数は、指定されたセットのメンバであるレコードで使用できます。

この関数は IDMS ページグループまたは基数を返さないため、この関数は、別のページグループを使用してアクセスされる、重複するデータベースが存在しない場合に使用します。重複するデータベースが存在する可能性がある場合は、GetFullDbKeyOfPriorMember 関数を使用します。

構文:

```
GetDbKeyOfPriorMember('IDMS_set_name')
```

この関数は、データ型 NUM32U または BIN4 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetFullDbKey

IDMS。連結された 4 バイトのページグループおよび基数識別子プレフィックスが含まれる、IDMS レコードの完全修飾データベースキーを返します。

この関数は、重複データベースキーが存在する場合に使用します。重複データベースキーが存在しない場合は、GetDbKeyOfOwner 関数を使用します。

構文:

```
GetFullDbKey()
```

この関数は、データ型 BIN8 の値を返します。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetFullDbKeyOfFirstMember

IDMS。指定されたセットの最初のメンバの完全修飾データベースキーを返します。完全修飾データベースキーには、連結された 4 バイトのページグループおよび基数識別子プレフィックスが含まれます。この関数は、指定されたセットのオーナーであるレコードで使用できます。

この関数は、別のページグループを使用してアクセスされた重複するデータベースが存在する場合に使用します。重複するデータベースが存在しない場合は、GetDbKeyOfFirstMember 関数を使用します。

構文:

```
GetFullDbKeyOfFirstMember('IDMS_set_name')
```

この関数は、データ型 BIN8 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetFullDbKeyOfLastMember

IDMS。指定されたセットの最後のメンバの完全修飾データベースキーを返します。完全修飾データベースキーには、連結された 4 バイトのページグループおよび基数識別子プレフィックスが含まれます。この関数は、指定されたセットのオーナーであるレコードで使用できます。

この関数は、別のページグループを使用してアクセスされた重複するデータベースが存在する場合に使用します。重複するデータベースが存在しない場合は、GetDbKeyOfLastMember 関数を使用します。

構文:

```
GetFullDbKeyOfLastMember('IDMS_set_name')
```

この関数は、データ型 BIN8 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetFullDbKeyOfNextMember

IDMS。指定されたセットの次のメンバの完全修飾データベースキーを返します。完全修飾データベースキーには、連結された 4 バイトのページグループおよび基数識別子プレフィックスが含まれます。この関数は、指定されたセットのメンバであるレコードで使用できます。

この関数は、別のページグループを使用してアクセスされた重複するデータベースが存在する場合に使用します。重複するデータベースが存在しない場合は、GetDbKeyOfNextMember 関数を使用します。

構文:

```
GetFullDbKeyOfNextMember('IDMS_set_name')
```

この関数は、データ型 BIN8 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetFullDbKeyOfOwner

IDMS。連結された 4 バイトのページグループおよび基数識別子プレフィックスが含まれる、IDMS オーナーレコードのオーナーの完全修飾データベースキーを取得します。

この関数は、別のページグループを使用してアクセスされた重複するデータベースが存在する場合に使用します。重複するデータベースが存在しない場合は、GetDbKeyOfOwner 関数を使用します。

構文:

```
GetFullDbKeyOfOwner('IDMS_set_name')
```

この関数は、データ型 BIN8 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスの IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

関連項目：

- [「GetDbKeyOfOwner」 \(ページ 315\)](#)

GetFullDbKeyOfPriorMember

IDMS。指定されたセットの前のメンバの完全修飾データベースキーを返します。完全修飾データベースキーには、連結された 4 バイトのページグループおよび基数識別子プレフィックスが含まれます。この関数は、指定されたセットのメンバであるレコードで使用できます。

この関数は、別のページグループを使用してアクセスされた重複するデータベースが存在する場合に使用します。重複するデータベースが存在しない場合は、GetDbKeyOfPriorMember 関数を使用します。

構文:

GetFullDbKeyOfPriorMember('IDMS_set_name')

この関数は、データ型 BIN8 の値を返します。

次の表に、パラメータを示します。

パラメータ	説明
IDMS_set_name	IDMS セット名です。 この値は、一重引用符で囲みます。 注: システムインデックスを表す IDMS セット名は使用しないでください。

使用上の注意: この関数は、ユーザー定義式フィールドの操作前の画像と変更インジケータのカラムを追加するのに必要な操作前の画像のバッファをサポートします。

GetIMSRBAByLevel

IMS データソースに定義されているデータマップレコードについて、現在のセグメントのパスの指定されたレベルにある IMS セグメントの RBA 値を取得します。

IMS データマップのセグメントデータにセグメント RBA を追加する場合、Informatica では、GetDatabaseKey 関数ではなく GetIMSRBAByLevel 関数を使用することをお勧めしています。GetIMSRBAByLevel 関数では、キーのない、または一意のキーが存在しない親セグメントの RBA を取得することができます。

バルクデータ移動操作または IMS 同期 CDC に使用される IMS データマップには GetIMSRBAByLevel 関数を使用します。

次のタイプの IMS データマップ内のレコードには、GetIMSRBAByLevel 関数を使用できません。

- IMS ログベース CDC に使用される IMS データマップ
- 次のタイプのレコードが含まれる IMS アンロードファイルへのアクセスに使用される IMS データマップ
 - HISAM データセットの任意のフォーマットのアンロードレコード
 - HDAM および HIDAM データセットの BMC SHORT または BMC XSHORT フォーマットのアンロードレコード
 - IMS 高速パスデータセットの標準 IBM フォーマットのアンロードレコード

GetIMSRBByLevel 関数が使用される次のタイプのデータマップでデータベース行のテストを実行するには、次のライブラリを APF 許可します。

- Netport ジョブに BMP と共に IMS DL/1 バッチデータマップを使用する場合は、BMP STEPLIB 連結内のすべてのライブラリを APF 許可する必要があります。このように処理しないと、PowerExchange によって戻りコード 12 でメッセージ PWX-02204 が発行されます。
- IMS ODBA データマップを使用する場合は、PowerExchange Listener の STEPLIB 連結内のすべてのライブラリを APF 許可する必要があります。このように処理しないと、PowerExchange が異常終了します。

構文:

```
result=GetIMSRBByLevel([integer])
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	要求されたセグメントの RBA の 8 バイトバイナリ値を返します。	NUM64、NUM64U、または長さ 8 の BIN
<i>integer</i>	オプション。次のうち 1 つの要求タイプになります。 <ul style="list-style-type: none">- 値なし。現在のセグメントの RBA が返されます。- 正の整数。RBA を取得する IMS 階層にあるセグメントの 4 バイト整数のレベル番号です。レベル番号は、階層内の現在のセグメント、またはセグメントの先祖のレベル番号に制限されます。例えば、5 つのレベルから成る階層のレベル 3 にあるセグメントの場合は、レベル 1、2、および 3 のセグメントの RBA を要求できます。現在のセグメントのレベルより下の階層にあるセグメントの RBA を要求した場合、ランタイムエラーが発生します。- 負の整数。相対要求を示します。例えば、-2 と指定することで、現在のセグメントの 2 レベル上にあるセグメントの RBA を要求できます。絶対値が現在のセグメントのレベル番号以上となる負の整数を指定した場合、ランタイムエラーが発生します。	NUM64

注: IMS RBA の長さは 4 バイトですが、GetIMSRBByLevel 関数では、要求されたセグメントの RBA 値の 8 バイトバイナリ表現を返します。将来的な拡張を可能にし、FastPath および HALDB データベースの一貫性を保証するために、8 バイトが必要となります。

関連項目：

- [「GetDatabaseKey」 \(ページ 312\)](#)

GetPageGroup

IDMS。IDMS レコードが存在するページグループを取得します。

構文:

```
result=GetPageGroup()
```

戻り値:

NUM16 または NUM16U、あるいは長さ 2 の BIN ページグループです。

GetPageGroupOfOwner

IDMS。IDMS オーナーレコードのページグループを取得します。

構文:

```
result=GetPageGroupOfOwner('IDMS_set_name')
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	IDMS オーナーレコードのページグループを返します。	NUM16 または NUM16U、あるいは長さ 2 の BIN
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。	-

GetPgGrpAndRdx

IDMS。IDMS レコードのページグループおよび基数を取得します。

構文:

```
result=GetPgGrpAndRdx()
```

戻り値:

NUM32 または NUM32U、あるいは長さ 4 の BIN です。連結した IDMS レコードのページグループおよび基数です。

GetPgGrpAndRdxOfOwner

IDMS。IDMS オーナーレコードのページグループおよび基数を取得します。

構文:

```
result=GetPgGrpAndRdxOfOwner('IDMS_set_name')
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	連結した IDMS オーナーレコードのページグループおよび基数を返します。	NUM32 または NUM32U、あるいは長さ 4 の BIN
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。	-

GetPgGrpOfOwner

IDMS。IDMS オーナーレコードのページグループを取得します。

構文:

```
result=GetPgGrpOfOwner('IDMS_set_name')
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	IDMS オーナーレコードのページグループです。	NUM16 または NUM16U、あるいは長さ 2 の BIN
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。	-

GetRadix

IDMS。IDMS レコードの基数を取得します。

構文:

```
result=GetRadix()
```

戻り値:

NUM16 または NUM16U、あるいは長さ 2 の BIN IDMS レコードの基数です。

GetRadixOfOwner

IDMS。IDMS レコードの基数を取得します。

構文:

```
result=GetRadixOfOwner(' IDMS_set_name')
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	IDMS オーナーレコードの基数です。	NUM16 または NUM16U、あるいは長さ 2 の BIN
<i>IDMS_set_name</i>	IDMS セット名です。 この値は、一重引用符で囲みます。	-

GetSeqWithinLevel

シーケンシャルフラットファイル、IMS、および IDMS です。現在の階層レベルにあるレコードのシーケンス番号を取得します。

構文:

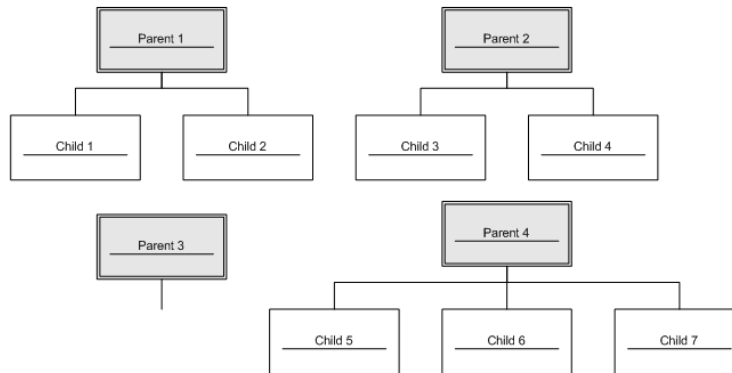
```
result=GetSeqWithinLevel()
```

戻り値:

NUM32。レコードのシーケンス番号です。

GetSeqWithinLevel の例

以下の図に、階層の例を示します。



この階層では、GetSeqWithinLevel 関数によって以下のレコードシーケンス番号が返されます。

1,2,3,4,0,5,6,7

返されたレコードシーケンス番号には、次のレコードが反映されます。

- 親 1 の子 1 および子 2
- 親 2 の子 3 および子 4
- 親 3 の子なし (0)
- 親 4 の子 5、子 6、子 7

GetSeqWithinParent

IMS および IDMS。現在の親レコードインスタンスに対するレコードシーケンス番号を取得します。

構文:

`result=GetSeqWithinParent()`

戻り値:

NUM32。レコードのシーケンス番号です。

GetSeqWithinParent の例

GetSeqWithinLevel で示した階層の場合、GetSeqWithinParent 関数では次のレコードシーケンス番号を返します。

1,2,1,2,0,1,2,3

返されたレコードシーケンス番号には、次のレコードが反映されます。

- 親 1 の子 1 および子 2
- 親 2 の子 1 および子 2
- 親 3 の子なし (0)
- 親 4 の子 1、子 2、子 3

関連項目：

- [「GetSeqWithinLevel」 \(ページ 322\)](#)

LengthOf

フィールドの長さを取得します。

構文:

```
result=LengthOf(field)
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	フィールドの長さを返します。	NUM32
<i>field</i>	長さを返すフィールドです。	任意です。

LTrim

指定された文字をフィールドの左側からトリミングします。

注: トリミングするフィールドは、ターゲットフィールドと同様のデータ型である必要があります。例えば、文字データ型のフィールドを数値データ型のターゲットフィールドにコピーする場合には LTrim 関数を使用できません。

構文:

```
result=LTrim(field, 'character')
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	指定された文字の左側からトリミングされたフィールドのコンテンツが返されます。	CHAR
<i>field</i>	トリミングするフィールドです。	CHAR
<i>character</i>	オプション。フィールドの左側からトリミングする文字です。 文字が指定されなかった場合、デフォルト値はスペースです。 この文字は、一重引用符で囲みます。	-

LTrim の例

Text というフィールドに次の文字が格納されている場合

```
*****The quick brown fox*****
```

関数 LTrim(Text, '*') は、次の文字を返します。

```
The quick brown fox*****
```

RTrim

指定された文字をフィールドの右側からトリミングします。

注: トリミングするフィールドは、ターゲットフィールドと同様のデータ型である必要があります。例えば、文字データ型のフィールドを数値データ型のターゲットフィールドにコピーする場合には RTrim 関数を使用できません。

構文:

RTrim(*field*, 'character')

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	指定された文字の右側からトリミングされたフィールドのコンテンツが返されます。	CHAR
<i>field</i>	トリミングするフィールドです。	CHAR
<i>character</i>	オプション。フィールドの右側からトリミングする文字です。 文字が指定されなかった場合、デフォルト値はスペースです。 この文字は、一重引用符で囲みます。	-

RTrim の例

Text というフィールドに次の文字が格納されている場合

*****The quick brown fox*****

関数 RTrim(Text, '*') は、次の文字を返します。

*****The quick brown fox

SetBitA

フィールドの指定オフセットでのビットのオン設定またはオフ設定を示す文字値を返します。

構文:

result=SetBitA(*field*, *offset*)

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	指定されたビットのオン設定またはオフ設定を示す次のうちのどちらかの値を返します。 - Y。オン。 - N。オフ。	CHAR
<i>field</i>	チェックするフィールドです。	任意です。
<i>offset</i>	フィールド内のビットの位置です。 1 バイトフィールドの場合、オフセットは 0 ビットから始まります。 2 バイトフィールドの場合、オフセットは第 2 バイトの 8 から始まります。	-

SetBitN

フィールドの指定オフセットでのビットのオン設定またはオフ設定を示す文字値を返します。

構文:

```
result=SetBitN(field,offset[,on_value][,off_value])
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	指定されたビットのオン設定またはオフ設定を示す次のうちのどちらかの値を返します。 - 1. オン。 - 0. オフ。	NUM8
<i>field</i>	チェックするフィールドです。	任意です。
<i>offset</i>	フィールド内のビットの位置です。 オフセットは 0 から始まり、フィールドの最後のビットで終わります。例えば、4 バイトの CHAR フィールドがある場合、最後のオフセットは 30 になります。	-
<i>on_value</i>	オプションです。ビット値がオンに設定されている場合、返された値をオーバーライドします。例えば、オン値が 3 に設定され、ビット値がオンに設定されている場合、SetBitN は 3 を返します。 デフォルトは 1 です。	NUM8
<i>off_value</i>	オプションです。ビット値がオフに設定されている場合、返された値をオーバーライドします。例えば、オフ値が 2 に設定され、ビット値がオフに設定されている場合、SetBitN は 2 を返します。 デフォルトは 0 です。	NUM8

Split

指定された開始ポイントから、指定された長さのフィールドコンテンツを取得します。この関数を使用して、グループフィールドから単一フィールドを取得することができます。

注: 分割するフィールドは、ターゲットフィールドと同様のデータ型である必要があります。例えば、文字データ型のフィールドを数値データ型のターゲットフィールドにコピーする場合には Split 関数を使用できません。

CHAR ソースフィールドから CHAR 以外のデータ型のサブフィールドを作成するには、次のように Split 関数と CopyData 関数を組み合わせて使用します。

1. ソースフィールドに対して Split 関数を使用して、新しい CHAR フィールドを作成します。
2. 新しい CHAR フィールドに対して CopyData 関数を使用して、数値データタイプなど、異なるデータタイプのターゲットフィールドを作成します。

構文:

```
result=Split(field,start_byte,length)
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>result</i>	指定された開始バイトから、指定された長さのフィールドコンテンツを返します。	すべての型
<i>field</i>	サブフィールドを取得するフィールドです。	すべての型
<i>start_byte</i>	サブフィールドが始まるバイト番号です。オフセットは 1 から始まります。	NUM32
<i>length</i>	取得するバイト数です。	NUM32

関連項目：

- [「CopyData」 \(ページ 308\)](#)

Split の例

Text というフィールドに次の文字列が格納されている場合

The quick brown fox jumped over the lazy dog

関数 Split(Text,17,10)は次の文字を返します。

fox jumped

関数 Split (Text、1、23) は次の文字を返します。

The quick brown fox

Strip

フィールドの左側、右側、または両側から指定された文字を除去します。

構文:

```
Strip(field,['side'], 'character')
```

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>field</i>	トリミングするフィールドです。	CHAR
<i>side</i>	オプションです。指定された文字が除去される、フィールドのサイドです。次のうち 1 つの値を指定します。 <ul style="list-style-type: none">- L。左側。- R。右側。- B。両側。 サイドが指定されなかった場合、デフォルト値は B です。このサイド値は、一重引用符で囲みます。	-
<i>character</i>	オプションです。指定された側あるいは両側からのフィールドから削除する文字です。文字が指定されなかった場合、デフォルト値はスペースです。この文字は、一重引用符で囲みます。	-

戻り値:

CHAR。 指定された側あるいは両側から指定された文字が除去されたフィールドのコンテンツです。

Strip の例

Text というフィールドに次の文字が格納されている場合

```
*****The quick brown fox*****
```

関数 Strip(Text,'R','*') は、次の文字を返します。

```
*****The quick brown fox
```

関数 Strip(Text,'L','*') は、次の文字を返します。

```
The quick brown fox*****
```

関数 Strip(Text,'B','*') は、次の文字を返します。

```
The quick brown fox
```

ToLower

フィールド内のすべての文字を小文字に変換します。

構文:

ToLower(*field*)

次の表に、これらのパラメータについて説明します。

パラメータ	説明	データ型
<i>field</i>	小文字に変換するフィールドです。	CHAR または VARCHAR

戻り値:

CHAR。 小文字に変換されたフィールドのコンテンツです。

ToUpper

フィールド内のすべての文字を大文字に変換します。

構文:

ToUpper(*field*)

パラメータ	説明	データ型
<i>field</i>	大文字に変換するフィールドです。	CHAR または VARCHAR

戻り値:

CHAR。大文字に変換されたフィールドのコンテンツです。

CallProg Function 関数によるユーザー定義プログラムの呼び出し

CallProg 関数を使用しユーザー定義プログラムを呼び出して、レコードのソースデータを処理することができます。

ユーザー定義プログラムの場所

オペレーティングシステムに応じて、PowerExchange リスナは以下のいずれかの場所からユーザー定義プログラムをロードします。

- Linux、UNIX、または Windows の場合、PowerExchange リスナは、指定された DLL を含む PATH の最初のディレクトリからプログラムをロードします。
- z/OS の場合、PowerExchange リスナは以下のいずれかの場所からプログラムをロードします。
 - DBMOVER 構成ファイルに PC_AUTH=Y 文が含まれている場合、PowerExchange リスナは、DTLLOAD DD 文からユーザー定義プログラムをロードします。
 - DBMOVER 構成ファイルに PC_AUTH=Y 文が含まれていない場合、PowerExchange リスナは、PowerExchange リスナの JCL にある JOBLIB/STEPLIB か、LINKLIST からユーザー定義プログラムをロードします。

注: ユーザー定義プログラムが APF 許可されていないライブラリ内にある場合は、PC_AUTH=Y と DTLLOAD DD を使用してください。

- i5/OS の場合、PowerExchange リスナは、PowerExchange リスナジョブのアクティブなライブラリリストからプログラムをロードします。

CallProg 関数で呼び出されたユーザー定義プログラムのロードエラー

CallProg 関数でユーザー定義プログラムを呼び出した場合、最初の呼び出しでプログラムのエントリポイントの関数アドレスがロードされます。ロードは、次のうち 1 つの理由で失敗することがあります。

- CallProg 関数が、オペレーティングシステムの標準的なプログラム検索順序でプログラムを見つけることができなかった。

- 以下の表に、CallProg 関数で検索するプログラムタイプを示します。

プログラムタイプ	オペレーティングシステム
DLL	Windows
ロードモジュール	z/OS
サービスプログラム	i5/OS
共有オブジェクト	Linux および UNIX

- 一部のオペレーティングシステムでは、プログラム名の太文字と小文字が区別されるが、プログラム名が正確に指定されていなかった。
- CallProg 関数は、モジュールハンドルは見つけたが、モジュールのルーチンを見つけることができなかった。

以下の表に、考えられるエラーの原因を示します。

問題	リンケージ型	オペレーティングシステム
ルーチンの記号がエクスポートされていない。	C OS400	i5/OS
ルーチンが #pragma でフェッチ可能として定義されていない。	C PLI	z/OS
ルーチンはエクスポートされなかった。	C	Windows

ユーザー定義プログラムへ渡されたパラメータリスト

CallProg 関数でユーザー定義プログラムを呼び出した場合、CallProg 関数では、そのユーザー定義プログラムにパラメータリストで情報を渡します。

ユーザー定義プログラムでは、CallProg 関数に情報を返すようにパラメータリストでパラメータを設定できます。

以下の表に、パラメータリストの入出力パラメータを示します。

パラメータ	データタイプ	用途	説明
NbrFlds	整数	入力	<p>CallProg 関数に渡される引数の数を指定します。例えば、CallProg 関数に次の 2 つの引数を渡すとしてします。</p> <pre>CallProg('program', 'subroutine', 'linkage', arg1, arg2)</pre> <p>PowerExchange によって NbrFlds パラメータが 2 に設定されます。</p> <p>注: 内部では、引数は ppData 配列および pDataLen 配列を介して CallProg 関数に渡されます。PowerExchange による NbrFlds 整数、ppData 配列、pDataLen 配列の用法は、C プログラムでの argc 整数、argv 配列の用法に似ています。</p> <pre>int NbrFlds, char *ppData[], int *pDataLen[] main(int argc, char *argv[])</pre>
pDataLen	ポインタの配列	入力と出力	<p>CallProg 関数に渡される引数の長さを指定します。</p> <p>次のように、引数の長さはデータタイプによって決まります。</p> <ul style="list-style-type: none"> - 固定長データタイプ。DOUBLE、FLOAT、NUM8、NUM16、NUM32、NUM64 DATE、TIME TIMESTAMP のデータタイプの場合、値が存在するときは常に同じサイズになり、引数が NULL のときはサイズが 0 になります。 - NULL 終了ではない可変長データタイプ。VARCHAR などのデータタイプの場合、データの長さは pDataLen パラメータを使用して決めることができます。 <p>フィールドが CallProg 関数に返される場合、データの長さが変更されているときはその長さをリセットする必要があります。</p> <p>例えば、VARCHAR データタイプの第 2 引数のデータおよび長さを変更する場合は、次のようにコーディングできます。</p> <pre>char Name[50]; ... strcpy(ppData[1], Name); *pDataLen[1] = strlen(Name);</pre>
pMsgBuffer	文字列に対するポインタ	出力	<p>ユーザー定義プログラムによってエラーを記述する NULL 終了のメッセージ文字列が返されるメッセージバッファを指定します。ユーザー定義プログラムによりゼロ以外の戻りコードが返された場合、PowerExchange は返されたメッセージを発行します。Informatica では、エラーが発生しなかった場合、ユーザー定義プログラムによってメッセージバッファを空にしておくことをお勧めしています。</p> <p>注: エラーメッセージを返す場合は、CallProg ルーチンで割り当てられたメモリバッファサイズ（128 バイト）を超えないようにします。</p>

パラメータ	データタイプ	用途	説明
pMsgBufferSize	整数ポインタ	入力と出力	<p>メッセージバッファサイズを指定します。次の構文によって、ユーザー定義プログラムでは、エラーが発生した場合にメッセージバッファサイズが含まれたエラーメッセージを返すことができます。</p> <pre>strncpy(pMsgBuffer, "Error in program FRED", (*pMsgBufferSize) -1);</pre> <p>返されるメッセージ文字列が NULL 終了ではない場合は、メッセージバッファのサイズを返します。</p>
ppData	ポインタの配列	入力と出力	<p>CallProg 関数に渡される引数のデータに対するポインタの配列です。このデータは、C プログラムの main 関数における argv 配列のデータアクセスと同じ方法でアクセスされます。例えば、STRING タイプのフィールドの第 2 フィールドからデータを取得する場合は、次のようにコーディングできます。</p> <pre>char Name[50]; ... strcpy(Name, ppData[1]);</pre>

ユーザー定義プログラムの戻りコード

ユーザー定義プログラムでは、処理の成功または失敗を示すために CallProg 関数に戻りコードを返します。

次の戻りコードを返すようにユーザー定義プログラムをコーディングします。

- 0。成功
- 正または負のゼロ以外の戻りコード。エラー

CallProg 関数では、ユーザー定義プログラムからゼロ以外の戻りコードを受け取った場合、次のデフォルトのエラー処理を実行します。

1. マップレベルの応答が起動します。
2. マップレベルの応答によって、抽出が終了するか、ユーザー定義プログラムまたはサブルーチンがスキップします。

PowerExchange Navigator の式エディタでは、オプションとして CallProg 関数の呼び出しに戻り引数を指定することができます。戻り引数が省略された場合、CallProg 関数では、ユーザー定義プログラムから返された戻りコードを検出して、デフォルトのエラー処理を行います。

Check 関数では、CallProg 関数で呼び出されたユーザー定義プログラムの戻りコードによって成功または失敗を示すように指定することができます。

関連項目：

- [「Check」 \(ページ 305\)](#)

ユーザー定義プログラム - 例

PowerExchange には、CallProg 関数を介して呼び出せるユーザー定義プログラムを作成する際のテンプレートとして使用できるサンプルプログラムが用意されています。

提供されているサンプルプログラムの言語は次のとおりです。

- Assembler
- C
- COBOL
- PL/I
- RPG

ユーザー定義 C プログラム - 例

PowerExchange には、すべてのオペレーティングシステムに対応したサンプルのユーザー定義 C プログラムが用意されています。

ユーザー定義 C プログラムでは次のプロトタイプを使用します。

```
int CRoutine(int NbrFlds
             ,char *pMsgBuffer, int *pMsgBufferSize
             ,char *ppData[], int * pDataLen[]);
```

以下の表に、サンプルの C プログラムを一覧表示します。

オペレーティングシステム	メンバまたはプログラム	場所	リンケージ型	プログラム要件
i5/OS	-	-	C ¹	サービスプログラムのエクスポートされたルーチンです。
Linux UNIX	-	-	C ¹	実行可能共有オブジェクトのエクスポートされたルーチンです。
Windows	ucpe.c	PowerExchange インストールディレクトリ内の examples ディレクトリ	C ¹	DLL の CDECL 規約のエクスポートされたルーチンです。

オペレーティングシステム	メンバまたはプログラム	場所	リンケージ型	プログラム要件
z/OS	UCPE	hlq.SRCLIB ライブラリ	C ¹	<ul style="list-style-type: none"> - ロードモジュールからフェッチ可能です。 - 呼び出されるモジュールが AMODE (31) にリンクされています。 AMODE (24) ロードモジュールにアクセスする必要がある場合は、16MB ライン下にパラメータを再配置してから AMODE (24) ロードモジュールを呼び出す AMODE (31) プログラムを介してモジュールを呼び出します。
1. CallProg 関数呼び出しの第 3 引数として指定します。				

関連項目：

- [「ユーザー定義プログラムへ渡されたパラメータリスト」 \(ページ 330\)](#)

COBOL プログラム - 例

PowerExchange では、z/OS 上の SRCLIB ライブラリの UCPEC メンバおよび UCPEC01 メンバで、CallProg 関数の COBOL のサンプルコードを提供しています。

COBOL プログラムのオペレーティングシステムへのサポート

CallProg 式の第 3 引数にリンケージ型 COBOL を使用して、COBOL 言語プログラムへリンクします。COBOL プログラムは z/OS で使用できます。

言語環境 (LE) COBOL ランタイムルーチンを使用して COBOL プログラムをコンパイルします。非 LE COBOL ランタイムルーチンでコンパイルされた COBOL プログラムの結果は予測できません。

COBOL プログラムリンケージ

COBOL では、最初の 4 つの引数は常に同じで、必須です。

残りのフィールドは、外部プログラムの必要性によって異なります。ユーザー定義式フィールドの属性は、外部プログラムで要求されるパラメータと一致している必要があります。

COBOL プログラムリンケージ - 例

以下のコードに、サンプルの COBOL プログラムのリンケージを示します。

```

003700 LINKAGE SECTION.
003800
003900 01  NUMBER-FIELDS          PIC S9(9) COMP.
003901
003902 01  FAILURE-CODE          PIC S9(9) COMP.
003903
003904 01  MESSAGE-BUFFER.
003905     05  MESSAGE-BUFFER-BYTE PIC X(1)
003906         OCCURS 1 TO 128
003907         DEPENDING ON MESSAGE-BUFFER-LENGTH.
003908 01  MESSAGE-BUFFER-LENGTH  PIC S9(9) COMP.
003910
004000 01  TEXT-AREA.
```

```

004010      05 TEXT-AREA-BYTE      PIC X(1) OCCURS 15.
004100 01   TEXT-AREA-LENGTH      PIC S9(9) COMP.
004111
004120 01   NUMBER1               PIC S9(9) COMP.
004130 01   NUMBER1-LENGTH        PIC S9(9) COMP.
005730
005800 PROCEDURE DIVISION USING
005801      NUMBER-FIELDS
005802      FAILURE-CODE
005803      MESSAGE-BUFFER
005807      MESSAGE-BUFFER-LENGTH
005809      TEXT-AREA
005810      TEXT-AREA-LENGTH
005811      NUMBER1
005820      NUMBER1-LENGTH

```

サンプルプログラム UPEC では、2 つのフィールドが COBOL プログラムに渡されます。

- 最初のフィールドは TEXT であり、最大サイズは 15 バイトです。COBOL フィールド TEXT-AREA を使用して、最初のフィールドとの間でデータを移動することができます。固定長 CHAR フィールドでない場合は、TEXT-AREA-LENGTH を使用して実際の長さを決定する必要があります。COBOL プログラムで長さを変更する必要がある場合は、フィールド TEXT-AREA-LENGTH に必要な値を格納する必要があります。
- 2 番目のフィールドは数値で、PowerExchange Navigator で NUM32 として定義されています。COBOL フィールド NUMBER1 を使用して、2 番目のフィールドとの間でデータを移動することができます。一般に、このフィールドは使用されません。

フィールドを NULL にできない場合は、COBOL プログラムへのエントリ時にこのフィールドに値 4 を含めます。NUMBER1-LENGTH が NULL の場合は 0 に設定されます。プログラムを呼び出す前にフィールドが NULL の場合、この状態になることがあります。NUMBER1-LENGTH が NULL に設定されている場合は、プログラムにより NUMBER1-LENGTH フィールドが 0 に設定されます。

Assembler プログラム - 例

PowerExchange では、SRCLIB ライブラリの UCPEA メンバで、CallProg プログラムの Assembler のサンプルコードを提供しています。

Assembler プログラムのオペレーティングシステムへのサポート

CallProg 式の第 3 引数にリンケージ型 OS を使用して、Assembler 言語プログラムへリンクします。アセンブリ言語プログラムは z/OS で使用できます。

Assembler プログラムでは、Register 1 が指定するパラメータリストにパラメータアドレスのリストを受信します。Assembler プログラムでは、Register 15 に戻りコードを代入します。

Assembler プログラムリンケージ

Assembler では、最初の 3 つの引数は常に同じで、必須です。引数と戻りコードの用途は、C および COBOL ルーチンの場合とほぼ同じです。

フィールドの定義は、フィールドの数とその型によって異なります。

Assembler プログラムリンケージ - 例

以下のコードに、サンプルのアセンブリ言語プログラムのリンケージを示します。

```

L      R3,0(R1)          get address of argument 1
      L      R4,0(R3)      get NumberFields value
      C      R4,=F'2'      Required value of 2 ?
      BNE    BADARGS
*-----
* Get arguments

```

```

*-----
NBRARGOK DS      0H
          L      R3,4(R1)          get address of argument 2
          ST     R3,AMSGBFF        = address of message buffer
          L      R3,8(R1)          get address of argument 3
          ST     R3,AMSGBFSZ       = size of message buffer
          L      R3,12(R1)         get address of argument 4
          ST     R3,ATEXT          = address of text argument
          L      R3,16(R1)         get address of argument 5
          ST     R3,ATEXTLEN       = length of text argument (15)
          L      R3,20(R1)         get address of argument 6
          ST     R3,ANUMBER        = address of number
          L      R3,24(R1)         get address of argument 7
          ST     R3,ANUMLEN        = length of number (always 4)

...
AMSGBFF DS      F
AMSGBFSZ DS      F
ATEXT   DS      F
ATEXTLEN DS      F
ANUMBER DS      F
ANUMLEN DS      F

```

サンプルプログラムでは、2つのフィールドが Assembler プログラムに渡されます。最初のフィールドは TEXT と呼ばれます。リストの 4 番目のアドレス、つまり 12 (R1) のアドレスを使用して、最初のフィールドとの間でデータを移動することができます。これが固定長 CHAR フィールドでない場合は、データ長を使用して実際の長さ (16 (R1) のアドレス) を決定します。

RPG プログラム - 例

PowerExchange では、*dtllib*/RPGLE の UCPERPGLE メンバで、CallProg プログラムの RPG のサンプルコードを提供しています。*dtllib* は、PowerExchange のインストール時に指定した PowerExchange ソフトウェアライブラリです。

RPG プログラムのオペレーティングシステムへのサポート

PowerExchange では、i5/OS で C および CL プログラムがサポートされます。

CallProg 式の第 3 引数にリンケージ型 OS400 を使用して、C 言語以外のプログラムへリンクします。OS400 リンケージ型は、CL、COBOL、PL/I、RPG など、C および i5/OS 以外の言語で必要です。リンケージ型 OS400 は i5/OS で使用できます。

CL プログラムリンケージ

最初の 4 つの引数は常に同じで、必須です。引数と戻りコードの用途は、他の言語の場合とほぼ同じです。

フィールドの定義は、フィールドの数とその型によって異なります。

CL プログラムリンケージ - 例

以下のコードに、サンプルの CL プログラムのリンケージを示します。

```

PGM PARM(&NBRFLDS &RC &MSGBUFF &MSGBUFFSZ &TEXT &TEXTLEN &NUM &NUMLEN)
/*-----*/
/* ARGUMENTS                                     */
/*-----*/
          DCL      VAR(&NBRFLDS) TYPE(*CHAR) LEN(4)
          DCL      VAR(&RC) TYPE(*CHAR) LEN(4)
          DCL      VAR(&MSGBUFF) TYPE(*CHAR) LEN(128)
          DCL      VAR(&MSGBUFFSZ) TYPE(*CHAR) LEN(4)
          DCL      VAR(&TEXT) TYPE(*CHAR) LEN(15)
          DCL      VAR(&TEXTLEN) TYPE(*CHAR) LEN(4)
          DCL      VAR(&NUM) TYPE(*CHAR) LEN(4)
          DCL      VAR(&NUMLEN) TYPE(*CHAR) LEN(4)

```


サンプルプログラムでは、2つのフィールドがCLプログラムに渡されます。

最初のフィールドはTEXTと呼ばれ、固定長は15です。5番目の引数&TEXTを使用して、最初のフィールドとの間でデータを移動することができます。これが固定長CHARフィールドでない場合は、データ長を使用して実際の長さ（フィールド&TEXTLENなど）を決定する必要があります。

CL言語では整数を直接サポートしていません。数値を取得するには、&TEXTLENのコンテンツは、次のような文を使用してバック10進数に移動する必要があります。

```
CHGVAR      VAR(&DECTEXTLEN) VALUE(%BIN(&TEXTLEN 1 4))
```

プログラムによって長さが変更された場合には、次のような文を使用して、新しい長さをTEXTLENフィールドに移動する必要があります。

```
CHGVAR      VAR(%BINARY(&TEXTLEN)) VALUE(&DECTEXTLEN)
```

PL/I プログラム - 例

PowerExchange では、SRCLIB ライブラリの UCPEP メンバで、CallProg プログラムの PL/I のサンプルコードを提供しています。

PL/I プログラムのオペレーティングシステムへのサポート

CallProg 式の第3引数にリンケージ型 PLI を使用して、PL/I 言語プログラムへリンクします。

PL/I 言語プログラムは z/OS で使用できます。C インターフェース層を使用して Windows および UNIX 上の PL/I プログラムを呼び出します。i5/OS 上の PL/I プログラムを呼び出すには、リンケージ型 OS400 を使用します。

言語環境 (LE) PL/I ランタイムルーチンを使用して PL/I プログラムをコンパイルします。非 LE PL/I ランタイムルーチンでコンパイルされた PL/I プログラムの結果は予測できません。

PL/I プログラムリンケージ

最初の4つの引数は常に同じで、必須です。フィールドの定義は、フィールドの数とその型によって異なります。

PL/I プログラムリンケージ - 例

以下のコードに、サンプルの PL/I プログラムのリンケージを示します。

```
PROC(NUMBER_ARGUMENTS,
     FAILURE_CODE,
     MESSAGE_BUFFER_PTR,
     MESSAGE_BUFFER_LENGTH,
     TEXT_AREA_PTR,
     TEXT_AREA_LENGTH,
     NUMBER1,
     NUMBER1_LENGTH)
  OPTIONS(FETCHABLE)
  REORDER;

/*-----*/
/* LINKAGE FROM CALLER */
/*-----*/
DCL NUMBER_ARGUMENTS      FIXED BIN(31);
DCL FAILURE_CODE          FIXED BIN(31);

DCL MESSAGE_BUFFER_PTR    PTR;
DCL 1 MESSAGE_BUFFER_STR  BASED(ADDR(MESSAGE_BUFFER_PTR)),
  4 MESSAGE_BUFFER        CHAR(255);
DCL MESSAGE_BUFFER_LENGTH FIXED BIN(31);

DCL TEXT_AREA_PTR         PTR;
```

```

DCL 1 TEXT_AREA_STR      BASED(ADDR(TEXT_AREA_PTR)),
    4 TEXT_AREA          CHAR(15);
DCL TEXT_AREA_LENGTH     FIXED BIN(31);

DCL NUMBER1              FIXED BIN(31);
DCL NUMBER1_LENGTH       FIXED BIN(31);

```

サンプルプログラムでは、2つのフィールドがプログラムに渡されます。

- 最初のフィールドは TEXT と呼ばれ、最大サイズは 15 バイトです。PL/I フィールド TEXT-AREA を使用して、最初のフィールドとの間でデータを移動することができます。固定長フィールドでない場合は、TEXT-AREA-LENGTH を使用して実際の長さを決定する必要があります。プログラムで長さを変更する必要がある場合は、フィールド TEXT-AREA-LENGTH に必要な値を格納する必要があります。
- 2 番目のフィールドは数値で、ナビゲータで NUM32 として定義されています。フィールド NUMBER1 を使用して、2 番目のフィールドとの間でデータを移動することができます。フィールド NUMBER1-LENGTH を利用する必要はほとんどありません。

フィールドを NULL にできない場合は、PL/I プログラムへのエントリ時にこのフィールドに値 4 を含めます。

NUMBER1-LENGTH フィールドが NULL の場合は 0 に設定されます。プログラムを呼び出す前にフィールドが NULL の場合、この状態になることがあります。プログラムによってフィールドが NULL に設定された場合は、NUMBER1-LENGTH も 0 に設定されます。

データマップのユーザー定義フィールドのテスト

レコードにユーザー定義フィールドを定義した後、データベース行のテストを実行して、フィールドに定義されている PowerExchange 関数またはユーザー定義関数の動作をテストします。

ユーザー定義フィールドに基づいた列が含まれるデータマップのテーブルに対してデータベース行のテストを実行します。

データマップのユーザー定義フィールドをテストする手順

1. データマップで、ベースフィールドとしてのテストを行うユーザー定義フィールドが使用されているテーブルに列を追加します。
2. データマップを保存します。
3. テーブル上でデータの行テストを実行します。

注: ユーザー定義フィールドが含まれているデータマップに対してデータベース行のテストを実行する場合は、SQL 文に DISTINCT キーワードを含めないようにします。クエリで DISTINCT キーワードが必要な場合は、そのクエリに必要なフィールドが含まれるデータマップに別のテーブルを作成します。

4. データベース行のテストによる出力ウィンドウで出力を確認します。

非リレーショナルソースに SQL を使用方法の詳細については、『PowerExchange リファレンスマニュアル』を参照してください。

関連項目：

- [「テーブルへの列の追加」 \(ページ 80\)](#)
- [「データマップのテスト」 \(ページ 231\)](#)

付録 B

ユーザーアクセス方法プログラム

この付録では、以下の項目について説明します。

- [ユーザーアクセス方式プログラムの概要, 339 ページ](#)
- [ユーザーアクセス方法プログラムへ渡されたパラメータ構造体, 341 ページ](#)
- [ユーザーアクセス方式プログラムからの戻りコード, 343 ページ](#)
- [ユーザーアクセス方式プログラム - 例, 344 ページ](#)

ユーザーアクセス方式プログラムの概要

アセンブリ言語、C、または PL/I で記述されたユーザーアクセス方法プログラムを作成して、データマップのデータを処理します。

USER アクセス方法でデータマップを定義し、ユーザーアクセス方法プログラムの属性を指定することによって、ユーザーアクセス方法プログラムを呼び出します。これには、コンパイル済みのユーザーアクセス方法の DLL または共有ライブラリの名前、プログラムに伝達されるパラメータ、および関連データファイルの完全パスと名前が含まれます。

ユーザーアクセス方法プログラムには単一エントリポイントがあります。PowerExchange は、ユーザーアクセス方法プログラムを繰り返し呼び出して、すべてのデータが処理されるまで関連データファイルを処理します。

以下の表に、ユーザーアクセス方式プログラムに対して PowerExchange が行う呼び出しと、その呼び出しに対してプログラムが取るアクションを示します。

呼び出し	実行可能なユーザーアクセス方法プログラムアクション
OPEN	<p>データファイルを開き、次のように初期化を実行する。</p> <ul style="list-style-type: none"> - ストレージを割り当てる。 - 渡されたパラメータを検証して保存する。 - カウンタを初期化する。 - データファイルが存在することを確認する。 - データファイルを開く。 - ターゲットが交換される場合に、出力ファイルを削除する。 - PowerExchange Navigator に成功またはエラーコードを返す。
READ	<p>次のように、シーケンシャル読み取りを開始または続行します。</p> <ul style="list-style-type: none"> - PowerExchange Navigator から渡された任意の SQL 文を使用して適切なレコードを取得し、次のレコードを取得するためのデータファイルを読み取る。 - レコードポインタがデータを指すように設定し、返されるデータの長さに record_len フィールドの値を設定して、データを PowerExchange Navigator に渡す。 - PowerExchange に、成功、ファイル末尾、エラーコードを返す。
WRITE	<p>次のように、シーケンシャル書き込みを開始または続行する。</p> <ul style="list-style-type: none"> - レコードポインタと record_len フィールドで設定された長さを使用して、書き込むデータを取得する。 - 指定されたデータを書き込む。 - PowerExchange Navigator に成功またはエラーコードを返す。
CLOSE	<p>次のように、クリーンアップ操作を実行する。</p> <ul style="list-style-type: none"> - データファイルを閉じる。 - 割り当てられたストレージを解放する。 - データベースをコミットする。 - PowerExchange Navigator に成功またはエラーコードを返す。 - エラーが発生した場合に、メッセージバッファにメッセージテキストを送信する。メッセージバッファで 79 文字のメッセージを保持する。 - 呼び出しプログラムに制御を戻す。

PowerExchange では、呼び出すたびにユーザーアクセス方法プログラムにパラメータ構造体を渡します。パラメータ構造体には、ユーザーアクセス方法データマップで定義するパラメータが含まれます。

ユーザーアクセス方法は、呼び出しの成功か失敗を示す、またはデータファイルの末尾に達したことを示すために PowerExchange Navigator に戻りコードを返します。

関連項目：

- [「データマップの追加」 \(ページ 66\)](#)

ユーザーアクセス方法プログラムへ渡されたパラメータ構造体

PowerExchange Navigator は、パラメータ構造体に検索する単一のパラメータを提供します。

以下の表に、ユーザーアクセス方法プログラムに渡されたパラメータ構造体を示します。

変数名	説明	PowerExchange Navigator GUI の対応フィールド	タイプ	サイズ
LGMODE	LGCODE 変数の値 1 と共に使用して、次のように、データファイルを開く方法を示す。 <ul style="list-style-type: none">- 1. 読み取る。- 2. 書き込む。- 3. 更新する。- 4. 付加する。	該当なし	バイナリ	ワード
LGCODE	次のように、ユーザーアクセス方法プログラムにより実行される機能を示す。 <ul style="list-style-type: none">- 1. 開く。- 2. 閉じる。- 3. 読み取る。- 4. 書き込む。	該当なし	バイナリ	ワード
LGUI1	ユーザー定義パラメータ。取得すべきレコード長や行数などの数値情報を提供するために使用できます。	【USER アクセス方式】ダイアログボックスの【整数 1】の値。	バイナリ	ワード
LGUI2	ユーザー定義パラメータ。取得すべきレコード長や行数などの数値情報を提供するために使用できます。	【USER アクセス方式】ダイアログボックスの【整数 2】の値。	バイナリ	ワード
LGREADCT	データファイルから読み取ったレコード数。	該当なし	バイナリ	ワード
LGWRITCT	データファイルに書き込んだレコード数。	該当なし	バイナリ	ワード
LGMLRECL	取得データのレコード長。	ユーザーアクセス方式プログラムによる設定	バイナリ	ワード

変数名	説明	PowerExchange Navigator GUI の対応フィールド	タイプ	サイズ
LGUI1LEN	【USER アクセス方式】 ダイアログボックスの【文字列 1】ボックスに入力したデータの長さ。	【USER アクセス方式】 ダイアログボックスの【文字列 1】値から派生したもの。	バイナリ	ワード
LGUI2LEN	【USER アクセス方式】 ダイアログボックスの【文字列 2】ボックスに入力したデータの長さ。	【USER アクセス方式】 ダイアログボックスの【文字列 2】値から派生したもの。	バイナリ	ワード
LGFNLEN	【USER アクセス方式】 ダイアログボックスの【ファイル名】ボックスに入力したデータファイル名の長さ。	【USER アクセス方式】 ダイアログボックスの【ファイル名】値から派生したもの。	バイナリ	ワード
LGUIDLEN	【データマップのリモートノード】 ダイアログボックスの【ユーザ ID】ボックスに入力したデータの長さ。	【データマップのリモートノード】 ダイアログボックスの【ユーザ ID】値から派生したもの。	バイナリ	ワード
LGPWDLEN	【データマップリモートノード】 ダイアログボックスの【パスワード】ボックスに入力したデータの長さ。	【データマップリモートノード】 ダイアログボックスの【パスワード】値から派生したもの。	バイナリ	ワード
LGSQLEN	【データベース行のテスト】ダイアログボックスの【SQL 文】ボックスに入力したデータの長さ。	【データベース行のテスト】ダイアログボックスの【SQL 文】値から派生したもの。	バイナリ	ワード
LGUIPTR	文字列 1 用に格納されているデータへのポインタ。	【USER アクセス方式】 ダイアログボックスの【文字列 1】の値。	ポインタ	ワード
LGU2PTR	文字列 2 用に格納されているデータへのポインタ。	【USER アクセス方式】 ダイアログボックスの【文字列 2】の値。	ポインタ	ワード
LGFNPTR	ファイル名用に格納されているデータへのポインタ。	【USER アクセス方式】 ダイアログボックスの【ファイル名】の値。	ポインタ	ワード
LGUIDPTR	ユーザー ID 用に格納されているデータへのポインタ。	【データマップリモートノード】 ダイアログボックスの【ユーザー ID】値から派生したもの。	ポインタ	ワード
LGPWDPTR	パスワード用に格納されているデータへのポインタ。	【データマップリモートノード】 ダイアログボックスの【パスワード】値から派生したもの。	ポインタ	ワード
LGSQLPTR	SQL 文用に格納されているデータへのポインタ。	【データベース行のテスト】ダイアログボックスの【SQL 文】値。	ポインタ	ワード

変数名	説明	PowerExchange Navigator GUI の対応フィールド	タイプ	サイズ
LGPUWK1	ユーザー定義データ用のストレージ領域へのポインタ。	ユーザーアクセス方式プログラムによる設定	バイナリ	ワード
LGPUWK2	ユーザー定義データ用のストレージ領域へのポインタ。	ユーザーアクセス方式プログラムによる設定	バイナリ	ワード
LGOPENRC	次のように、OPEN 呼び出しのコードを戻す。: - 1. 成功 - 0. エラー	ユーザーアクセス方式プログラムによる設定	バイナリ	ワード
LGRECLEN	取得したレコードの長さ。	ユーザーアクセス方式プログラムによる設定	バイナリ	ワード
LGRECPTR	取得したレコードデータへのポインタ。	ユーザーアクセス方式プログラムによる設定	ポインタ	ワード
LGMSGBUF	ユーザーアクセス方法プログラムから PowerExchange へ返されたメッセージ。	ユーザーアクセス方式プログラムによる設定	文字	80 バイト

関連項目：

- [「\[データベース行のテスト\] ダイアログボックス」 \(ページ 237\)](#)
- [「データマッププロパティ - USER アクセス方式」 \(ページ 368\)](#)
- [「リモートシステムへのデータマップの送信」 \(ページ 95\)](#)

ユーザーアクセス方式プログラムからの戻りコード

ユーザーアクセス方法により、PowerExchange Navigator に戻りコードを返されます。

PowerExchange Navigator では、ユーザーアクセス方法プログラムから次のうちの 1 つの戻りコードが预期されます。

0

成功 OPEN、READ、WRITE、または CLOSE の呼び出しに成功しました。

255

ファイルの末尾。これ以上読み取り可能なレコードはありません。

その他

エラー OPEN、READ、WRITE、または CLOSE の呼び出しに失敗しました。

ユーザーアクセス方式プログラム - 例

PowerExchange は、アセンブリ言語、C、または PL/I で記述された代表的なユーザーアクセス方法のプログラムを、ヘッダーファイルと JCL と共に提供しています。

uam*. * および dlgif*. * と名付けられたこれらのファイルは、PowerExchange ナビゲータインストールディレクトリのサンプルディレクトリにあります。これらのサンプルプログラムを編集およびコンパイルし、使用環境に応じてカスタマイズします。

付録 C

アプリケーショングループおよびアプリケーション

この付録では、以下の項目について説明します。

- [アプリケーショングループおよびアプリケーションの概要, 345 ページ](#)
- [アプリケーショングループの追加, 346 ページ](#)
- [アプリケーショングループのプロパティの表示または編集, 348 ページ](#)
- [アプリケーションの抽出開始ポイントのリセット, 350 ページ](#)
- [アプリケーショングループの削除, 350 ページ](#)

アプリケーショングループおよびアプリケーションの概要

ODBC 接続を使用して変更データを抽出する場合は、PowerExchange で、アプリケーショングループおよびアプリケーションが使用され、変更データ抽出用に抽出情報が保持されます。

アプリケーショングループでは、データソースのノード場所を定義します。登録グループを追加したときに、PowerExchange Navigator によりアプリケーショングループが追加されます。アプリケーショングループを手動で追加することもできます。

アプリケーションは、アプリケーショングループの下に表示されます。PowerExchange Navigator を使用したアプリケーションの追加は行いません。CDC 抽出を新しいアプリケーション名で実行したとき、または DTLUAPPL ユーティリティを実行したときにアプリケーションを動的に追加します。アプリケーション名は、ODBC 接続の DBQUAL2 パラメータ、または DTLAPP SQL オーバーライドに指定する名前です。

注: Informatica では、再起動の実行を検証するために、それぞれの抽出処理に一意のアプリケーション名を指定することをお勧めします。複数の抽出処理に同じアプリケーション名を使用した場合、間違ったデータが抽出されて他の予期しない結果をもたらすようなエラーが生じる場合があります。

各アプリケーションでは、降順のタイムスタンプで変更データ抽出の履歴が表示されます。各抽出の履歴には、以下が含まれます。

- タイムスタンプ
- 監査証跡
- 抽出ステータス
- キャプチャ登録名およびテーブル名

さらに、各アプリケーションでは次回の抽出開始ポイントを格納して、PowerExchange Navigator で、または DTLUAPPL ユーティリティを使用してそのアプリケーションをリセットすることができます。

CDEP ファイルは制御ファイルです。このファイル内に、PowerExchange によりアプリケーション情報が格納されます。アプリケーション情報には、アプリケーション名、次回の抽出開始ポイント、履歴を含む実行済みの抽出処理に関する情報が含まれています。PowerExchange ODBC では、CDEP ファイルのアプリケーション名に関連付けられたリスタートトークンを使用して、セッション用の開始抽出ポイントを決定します。アプリケーショングループは、PowerExchange Navigator の表示用 CDEP ファイルのアプリケーション情報のフィルタリングに使用されます。

注: PowerExchange Client for PowerCenter (PWXPC) 接続を使用して、推奨どおりに変更データを抽出する場合は、PowerExchange ではアプリケーショングループおよびアプリケーションは使用されません。PWXPC で、PowerExchange により以下の場所で抽出再起動情報が保持されます。

- 状態テーブルにおけるリレーショナルターゲット用
- Integration Service の状態ファイルにおける非リレーショナルターゲット用

詳細については、オペレーティングシステム用 CDC ガイドを参照してください。

関連項目：

- [「登録グループ」 \(ページ 169\)](#)

アプリケーショングループの追加

アプリケーショングループは、名前の付いたアプリケーションのグループで、ODBC 接続を使用する場合に使用されます。

アプリケーショングループにより、ソースのノード場所が定義され、必要に応じてソースにアクセスするためのユーザー ID とパスワードも定義されます。登録グループを追加する場合は、PowerExchange により、アプリケーショングループが同じ名前で追加されます。次の手順を実行して、アプリケーショングループを手動で追加することもできます。

1. リソースエクスプローラの [リソース] タブで、[追加] - [アプリケーショングループ] をクリックします。
2. [アプリケーショングループの追加] ダイアログボックスに、以下の情報を入力します。

名前

アプリケーショングループのユーザー定義名です。最大長は 16 文字です。

場所

変更データを抽出するソースの場所です。ソースが PowerExchange Navigator システムにある場合は、[ローカル] を選択します。

このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。

デフォルトは [ローカル] です。

ユーザー ID

i5/OS または z/OS のソースでは、PowerExchange セキュリティが有効な場合はオペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を(1,x)または(2,x)に設定すると、PowerExchange のセキュリティが有効になります。

DB2 for Linux, UNIX, and Windows、Microsoft SQL Server、または Oracle ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力してください。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

パスワード

指定したユーザー名のパスワードです。

ソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符 (')、二重引用符 (")、または通貨記号を含めることはできません。

z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

ユーザー ID を指定してパスワードを省略した場合、アプリケーショングループを開いたときに PowerExchange Navigator が **【パスワードが入力されていません】** ダイアログボックスにパスワードの入力を求めるメッセージを表示します。

3. **【完了】** をクリックします。

リソースインスペクタの【アプリケーショングループ】 タブで、アプリケーショングループ用のプロパティが表示されます。

ODBC 接続を使用して変更データを抽出する場合は、**リソースエクスプローラの【アプリケーショングループ】** タブに、アプリケーショングループに関連付けられたノード場所のすべてのアプリケーションが表示されます。

関連項目：

- [「登録グループ」](#) (ページ 169)

アプリケーショングループのプロパティの表示または編集

アプリケーショングループのプロパティを表示または編集することができます。

アプリケーションのグループ名、場所、ユーザー ID、およびパスワードは変更できます。

1. アプリケーショングループを開きます。

以下の表に、アプリケーショングループについてリソースインスペクタの **【アプリケーショングループ】** タブに表示されるプロパティを示します。

プロパティ	説明
名前	アプリケーショングループのユーザー定義名です。
場所	変更をキャプチャするソースの場所です。ソースが PowerExchange Navigator システムにある場合は、 【ローカル】 を選択します。 このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。

プロパティ	説明
ユーザー ID	<p>ソースにアクセスできるユーザー ID です。ユーザー ID のタイプは、次のソースタイプおよび PowerExchange のセキュリティ設定によって異なります。</p> <ul style="list-style-type: none"> - ソースが i5/OS システムまたは z/OS システムにあり、PowerExchange セキュリティが有効な場合は、オペレーティングシステムのユーザー ID を入力します。DBMOVER メンバの PowerExchange SECURITY 文を(1,x)または(2,x)に設定すると、PowerExchange のセキュリティが有効になります。 - DB2 for Linux, UNIX, and Windows、Microsoft SQL Server、または Oracle ソースの場合は、必要に応じて有効なデータベースユーザー ID を入力してください。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。
パスワード	<p>指定したユーザー名のパスワードです。</p> <p>ソースが i5/OS または z/OS システムにある場合、パスワードの代わりに有効な PowerExchange パスフレーズを入力できます。i5/OS のパスフレーズは、9～31 文字の範囲で指定できます。z/OS のパスフレーズは、9～128 文字の範囲で指定できます。パスフレーズには次の文字を含めることができます。</p> <ul style="list-style-type: none"> - 英字（大文字と小文字） - 数字（0～9） - スペース - 次に示す特殊文字。 ' - ; # \ , . / ! % & * () _ + { } : @ < > ? <p>注: 先頭に書かれている特殊文字はアポストロフィです。</p> <p>パスフレーズに一重引用符 (')、二重引用符 (")、または通貨記号を含めることはできません。</p> <p>z/OS で IBM IRRPHREX exit に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。</p> <p>注: z/OS で有効な RACF パスフレーズの最大長は 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。</p> <p>パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『PowerExchange リファレンスマニュアル』の「SECURITY 文」を参照してください。</p>

2. プロパティを編集するには、プロパティをクリックするか値を選択します。
プロパティを編集した後に、変更したプロパティの隣に赤い四角が表示されます。
ヒント: 変更を適用する前に取り消す場合は、**[リセット]** をクリックします。
3. 変更を適用するには、**[適用]** をクリックします。
アプリケーショングループ名を編集する場合は、その変更を確認するように求められます。新しい名前が既存のアプリケーショングループ名と一致する場合は、現存グループを上書きするように求められます。
4. **[はい]** をクリックして変更を確定します。
リソースインスペクタにより、変更したフィールドはすべて更新されます。

関連項目：

- [「アプリケーショングループの追加」 \(ページ 346\)](#)

アプリケーションの抽出開始ポイントのリセット

アプリケーションでは、キャプチャ登録の変更データ抽出の履歴が表示され、次回の抽出開始ポイントが格納されます。

【今回の実行】の抽出として表示された次回の抽出開始ポイントを別の抽出開始ポイントにリセットすることができます。

抽出開始ポイントのリセット後、アプリケーションにより、リセット開始ポイント時間で抽出を再開することも、抽出開始ポイントを再度リセットすることもできます。開始ポイントのリセットを取り消すことはできません。

また、最初の抽出後、DTLUAPPL ユーティリティを使用して、アプリケーションの抽出開始ポイントのリセットすることができます。

アプリケーションの抽出開始ポイントのリセットする手順

1. アプリケーショングループを開き、アプリケーションを開きます。
右側のペインは、降順のタイムスタンプでアプリケーションの抽出を一覧表示します。
現在の抽出情報は、**【今回の実行】**の抽出として表示されていますが、空白です。他の抽出はすべてタイムスタンプを表示しています。
2. 開始ポイントのリセットする抽出を選択します。
3. アプリケーショングループ名を右クリックし、**【新しい開始ポイントへのリセット】**をクリックします。
アプリケーショングループ名の隣のアイコンが赤く表示されます。
PowerExchange により、次回の抽出再開ポイントが選択された抽出開始ポイントにリセットされます。
再度アプリケーションを開いたときに、**【今回の実行】**の抽出によって、選択された抽出からの実行情報が表示されます。

アプリケーショングループの削除

アプリケーショングループは削除することができます。

アプリケーショングループを削除する場合は、PowerExchange により、PowerExchange Navigator システムのローカルグループ情報が削除されます。PowerExchange では、グループ、または関連する登録グループにあるアプリケーションは削除されません。別のアプリケーショングループを、削除したグループとして同じプロパティで追加する場合、新しいグループには、削除されたグループ用に存在するすべてのアプリケーションが含まれます。

また、PowerExchange によりアプリケーショングループが生成された場合、PowerExchange では、このアプリケーショングループの作成を生成した登録グループは削除されません。

アプリケーショングループを削除する手順

1. アプリケーショングループが開いている場合は、そのアプリケーショングループを閉じます。
2. **リソース**ウィンドウで、1 つ以上のアプリケーショングループを削除します。

3. グループの削除を確認するように求めるメッセージボックスが表示されたら、**【はい】** をクリックします。
PowerExchange により、単数または複数のアプリケーショングループが削除されます。

付録 D

データマップのプロパティ

この付録では、以下の項目について説明します。

- [データマッププロパティ - ADABAS アクセス方式, 352 ページ](#)
- [データマッププロパティ - DATACOM アクセス方式, 354 ページ](#)
- [データマッププロパティ - DB2 アクセス方式, 355 ページ](#)
- [データマッププロパティ - DB2UNLD アクセス方式, 355 ページ](#)
- [\[DB2 テーブルフィルタ\] ダイアログボックス, 356 ページ](#)
- [\[DB2 テーブルフィルタ\] - \[使用可能なテーブル\] ダイアログボックス, 358 ページ](#)
- [データマップのプロパティ - DL/I バッチまたは、IMS ODBA アクセス方式, 358 ページ](#)
- [データマッププロパティ - ESDS アクセス方式, 360 ページ](#)
- [データマッププロパティ - IDMS アクセス方式, 361 ページ](#)
- [データマッププロパティ - KSDS アクセス方式, 362 ページ](#)
- [データマッププロパティ - MQSeries アクセス方式, 362 ページ](#)
- [データマッププロパティ - RRDS アクセス方式, 364 ページ](#)
- [データマップのプロパティ - SEQ アクセス方式, 365 ページ](#)
- [データマッププロパティ - TAPE アクセス方式, 367 ページ](#)
- [データマッププロパティ - USER アクセス方式, 368 ページ](#)
- [データマッププロパティ - データのデフォルト値, 370 ページ](#)
- [データマップのプロパティ - データチェック, 371 ページ](#)
- [データマッププロパティ - レコードレベルの出口, 373 ページ](#)
- [データマッププロパティ - キー, 374 ページ](#)

データマッププロパティ - ADABAS アクセス方式

Adabas データマップのプロパティを表示または編集します。

データベース ID

Adabas データベース ID です。実行時にデータベースをオーバーライドする場合、または実際のデータベース ID が 0 の場合に、0 に設定します。ランタイム時にデータベースをオーバーライドすることにより、ある Adabas 環境から別の環境へ簡単にデータマップを移行します。

ランタイム時にデータベースをオーバーライドするには、次のうち 1 つの方法を使用します。

- DBMOVE 構成ファイルの ADABAS_DEFAULT_DBID 文
- PowerCenter のセッションプロパティの [データベース ID の上書き] オプション
- ODBC DTLDSN エスケープシーケンスのファイル名のオーバーライド
- ODBC DBQUAL1 パラメータのファイル名のオーバーライド

デフォルトは 1 です。

ファイル番号

Adabas データベースのファイル番号です。

デフォルトは 1 です。

最適化レベル

PowerExchange による Adabas キーの使用方法を指定します。次のオプションがあります。

- **OFF**。PowerExchange では、Adabas キーは使用されません。
- **フル**。SQL WHERE 句が存在する場合、PowerExchange では Adabas キーが使用されます。
- **範囲のみ**。キー値が指定された範囲にある場合、および SQL WHERE 句が存在する場合、PowerExchange では Adabas キーが使用されます。

デフォルトは**範囲のみ**です。

エンコーディング

エンコードのスキームです。次のオプションがあります。

- **デフォルト**。PowerExchange では、オペレーティングシステムのデフォルトのエンコードスキーマが使用されます。
- **EBCDIC**。EBCDIC です。
- **ASCII HIEND**。ASCII ハイエンディアンです。
- **ASCII LOEND**。ASCII リトルエンディアンです。

デフォルトは**デフォルト**です。

コードページ

データベースの文字データの文字セットを示すコードページです。[**デフォルト**] エンコードでは、コードページを選択できません。PowerExchange では、オペレーティングシステムのデフォルトのコードページが使用されます。

デフォルトは**デフォルト**です。

ワイド文字コードページ

ワイド文字データタイプを使用する Adabas フィールドのコードページです。

[**デフォルト**] を選択して、次のオペレーティングシステムのコードページを使用します。

- Linux、UNIX、Windows。UTF-8 です。
- z/OS。デフォルトのオペレーティングシステムコードページです。

デフォルトは**デフォルト**です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - DATACOM アクセス方式

Datcom データマップのプロパティを表示または編集します。

データベース名

必須。CA Datcom/DB データベース名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

データベース ID

必須です。CA Datcom/DB データベースの ID 番号です。

注: データディクショナリ内に CA Datcom/DB データベースを定義する場合は、数値 ID をデータベースに割り当てる必要があります。この ID 値はデータベースの DATACOM-ID 属性値です。1 から 20、1000 から 1020、2000 から 2020、3000 から 3020、4000 から 4020、5000 などの ID は、CA Datcom/DB で使用するために予約されています。

レコードサイズ

テーブルのレコードサイズです。データマップを追加していて、**[レコード定義のインポート]** オプションを選択しない場合は、レコードサイズを入力します。CA Datcom/DB テーブルでは、レコードサイズに次のような制限が適用されます。

- 最小サイズは 1 バイトです。
- 最大サイズは 32720 バイトです。

コードページ

データベース内の文字データに使用する文字セットを示すコードページです。PowerExchange はデフォルトのコードページとして IBM-037 を使用します。

注: マルチバイトコードページが使用できるのは Datcom です。

デフォルト値は **Default** です。

関連項目：

- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - DB2 アクセス方式

DB2 データマップのプロパティを表示または編集します。

DB インスタンス

必須です。DB2 データベースのインスタンス名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

テーブル名

必須です。DB2 テーブル名です。テーブル名を入力するか、[参照] ボタンをクリックして、テーブルに対するフィルタ条件を入力します。

エンコーディング

エンコードのスキームです。次のオプションがあります。

- **デフォルト**。PowerExchange では、オペレーティングシステムのデフォルトのエンコードスキームが使用されます。
- **EBCDIC**。EBCDIC です。
- **ASCII HIEND**。ASCII ハイエンディアンです。
- **ASCII LOEND**。ASCII リトルエンディアンです。

デフォルトは**デフォルト**です。

コードページ

データベースの文字データの文字セットを示すコードページです。[デフォルト] エンコードでは、コードページを選択できません。PowerExchange では、オペレーティングシステムのデフォルトのコードページが使用されます。

デフォルトは**デフォルト**です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - DB2UNLD アクセス方式

DB2 アンロードファイルデータマップのプロパティを表示または編集します。

ファイル名

必須です。DB2 アンロードファイル名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

DB インスタンス

DB2 データベースのインスタンス名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

テーブル名

必須です。DB2 テーブル名です。テーブル名を入力するか、[参照] ボタンをクリックして、テーブルに対するフィルタ条件を入力します。

アンロードの種類

アンロードのタイプです。アンロードファイルへの一括アクセスに対しては、次の IBM および BMC の形式の、

- REORG UNLOAD
- DSNTIAUL/BMC UNLOAD+
- DSNUTILB UNLOAD
- UNDEFINED。デフォルトでは、このタイプは NULL インジケータと共に生成され、カウントフィールドは REORG UNLOAD EXTERNAL タイプと同じポジションになります。

これらのアンロードタイプを使用する代わりに、これらの文を特定の要件に合うように変更することができます。ユーザー定義のアンロード形式に対応するようにこれらの文を移動または削除できますが、PowerExchange では適用するデータフィールドにこれらの文が先行する場合に文を解釈します。

エンコーディング

エンコードのスキームです。次のオプションがあります。

- **デフォルト**。PowerExchange では、オペレーティングシステムのデフォルトのエンコードスキーマが使用されます。
- **EBCDIC**。EBCDIC です。
- **ASCII HIEND**。ASCII ハイエンディアンです。
- **ASCII LOEND**。ASCII リトルエンディアンです。

デフォルトは**デフォルト**です。

コードページ

データベースの文字データの文字セットを示すコードページです。[デフォルト] エンコードでは、コードページを選択できません。PowerExchange では、オペレーティングシステムのデフォルトのコードページが使用されます。

デフォルトは**デフォルト**です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

[DB2 テーブルフィルタ] ダイアログボックス

[DB2 テーブルフィルタ] ダイアログボックスで、DB2 カタログまたは DB2 アンロードファイルデータマップのテーブルフィルタ条件を表示または編集します。

ダイアログボックスには以下のフィールドがあります。

DB インスタンス

DB2 データベースのインスタンス名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

場所

データソースの場所です。ソースが PowerExchange Navigator システムにある場合は、**【ローカル】** を選択します。

このフィールドに、PowerExchange Navigator システムの dbmover.cfg 構成ファイルの NODE 文で定義した場所が表示されます。

デフォルトは **【ローカル】** です。

ユーザー ID

データベースにアクセスできるデータベースユーザー ID です（必要な場合）。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

サポートされている Linux、UNIX、または Windows システムのデータベースで、PowerExchange LDAP ユーザー認証を有効にし、リレーショナルパススルー認証を無効（該当する場合）にしている場合、ユーザー ID はエンタープライズユーザー名になります。

詳細については、『*PowerExchange リファレンスマニュアル*』を参照してください。

パスワード

指定したユーザー名のパスワードです。

z/OS にアクセスする場合は、パスワードではなく有効な PowerExchange パスフレーズを入力できます。PowerExchange パスフレーズは 9～128 文字の長さで指定し、次の文字を含めることができます。

- 英字（大文字と小文字）
- 数字（0～9）
- スペース
- 次に示す特殊文字。

' - ; # \ , . / ! % & * () _ + { } : @ | < > ?

注: 先頭に書かれている特殊文字はアポストロフィです。

パスフレーズに一重引用符（'）、二重引用符（"）、通貨記号を含めることはできません。

IBM IRRPHREX 出口に指定できる文字は、PowerExchange パスフレーズに指定できる文字に影響しません。

注: 有効な RACF のパスフレーズは最長 100 文字です。PowerExchange は、検証のために RACF にパスフレーズを渡すときに 100 文字を超えるパスフレーズを切り捨てます。

パスフレーズを使用するには、DBMOVER メンバのセキュリティ設定を SECURITY=(1,N)以上にして PowerExchange リスナを実行します。詳細については、『*PowerExchange リファレンスマニュアル*』の「SECURITY 文」を参照してください。

データのフィルタ

次のうちいずれかのフィルタボックスにテキストを入力します。

- **クリエータ/スキーマ**
- **DB 名/定義元**
- **テーブル**

i5/OS 用 DB2 では、入力する最大長は、**【クリエータ/スキーマ】** ボックスで 10 文字、**【テーブル】** ボックスで 32 文字です。

DB2 for z/OS では、入力する最大長は、**【クリエータ/スキーマ】** ボックスで 128 文字、**【テーブル】** ボックスで 128 文字です。

どのフィルタボックスでも、以下のワイルドカード文字を使用できます。

- 1つ以上の一致文字を表すアスタリスク (*)
- 1つの一致文字を表す 疑問符 (?)

大文字と小文字を区別する

デフォルトでは、いずれかのフィルタボックスに入力されたテキストは DB2 データソースの標準文字に変換されます。PowerExchange で DB2 データソース用に標準文字を使用しないフィルタでは、**【大文字と小文字を区別する】** オプションを選択します。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

[DB2 テーブルフィルタ] - [使用可能なテーブル] ダイアログボックス

DB2 または DB2 アンロードファイルデータマップ用のテーブルを選択します。

使用可能なテーブル

【DB2 テーブルフィルタ】 ダイアログボックスに入力したフィルタ条件を満たすテーブルが一覧表示されます。

マッピングする DB2 テーブルを選択して、**【完了】** をクリックします。

関連項目：

- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマップのプロパティ - DL/I バッチまたは、IMS ODBA アクセス方式

IMS DL/1 BATCH または **IMS ODBA** のアクセス方式を使用するデータマップのプロパティを表示または編集します。

IMS SSID

IMS のサブシステム ID です。最大長は 4 文字です。

次のアクセス方式に応じて、このプロパティは論理値または実際の SSID になります。

- IMS ODBA アクセス方式を使用する場合、この値は制御範囲に定義する実際の IMS SSID です。この値は、ODBA アクセスを設定した IMS ジョブの DBCTLIB 値と一致し、DBMOVER 構成メンバの IMSID 文に指定した SSID 値と一致する必要があります。

IMS ターゲットで、DTL.DBWRITE.IMS.*imsid* リソースプロファイルを使用して書き込みアクセスを制御する場合は、この IMS SSID 値がリソースプロファイルの *imsid* 値と一致することを確認します。

- DL/I アクセスまたは BMP アクセスを使用する場合、この値は DBMOVER 構成メンバおよび登録グループの IMSID 文に指定した SSID と一致する論理 SSID です。DL/I を使用して IMS データベースにアクセスする場合は使用されません。

IMS ターゲットで、DTL.DBWRITE.IMS リソースプロファイルを使用して書き込みアクセスを制御する場合は、この IMS SSID 値を指定しないでください。

注: バルクデータの移動元または移動先の PowerCenter セッションプロパティに指定された IMS SSID 値をオーバーライドできます。データベース行のテストの詳細パラメータに指定した IMS SSID もオーバーライドできます。

DBD 名

IMS ソースの IMS データベース記述 (DBD) の名前です。IMS アンロードファイルを読み取る場合はこの値を指定します。

PCB 番号

Netport ジョブを利用した DL/I バッチアクセスまたは BMP アクセスには必須です。データベースに関連する PCB 番号を指定します。PSB に COMPAT=YES と指定されている場合は PCB 番号に 1 を追加します。

PSB 名

IMS ODBA アクセスの場合は必須です。データベースの DBD が含まれる PSB の名前を指定します。

最大長は英数字 8 文字です。

PCB 名

IMS ODBA アクセスの場合は必須です。指定した PCB に存在し、指定した DBD を参照する PCB の名前を入力します。PCB のラベルまたは PCB 名のパラメータ値を入力できます。

データコードページ

データベースの文字データの文字セットを示すコードページです。[デフォルト] コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。

デフォルトはデフォルトです。

ファイルの先頭の *n* つのレコードをスキップする

スキップするデータファイルのヘッダレコードの数です。例えば、データファイルがデータを含まないヘッダレコードを 1 つ以上含む場合、スキップするヘッダレコードの数を入力します。

0 から 2147483647 までの有効な値です。デフォルトは 0 です。

ファイルの表示

データマップに関連付けられたデータファイルを表示します。[IMS アンロードデータセット] フィールドにファイル名を入力すると有効になります。

IMS アンロードデータセット

IMS アンロードファイルを処理する場合は、IMS アンロードデータセットの名前を入力してください。

データマッププロパティ - ESDS アクセス方式

VSAM エントリシーケンスのデータセット (ESDS) データマップのプロパティを表示または編集します。

ファイル名

VSAM エントリシーケンスのデータセット (ESDS) の完全修飾データセット名です。

データバッファ数

PowerExchange から VSAM に割り当てが要求される、データ制御間隔の I/O バッファの数です。

デフォルトは 2 です。

CI アクセス

VSAM データセットの読み取り時に、個々のデータレコードではなく制御間隔のコンテンツ全体にアクセスします。このオプションにより、読み込みのパフォーマンスが向上します。

VSAM データセットの読み込み時に個々のデータレコードのコンテンツにアクセスするには、このオプションの選択を取り消します。

注: データセットが圧縮されている場合、**[CI アクセス]** オプションは使用できません。

XRBA 値のプレフィックスレコード

VSAM データセットから読み込まれるすべてのレコードの拡張相対バイトアドレス (XRBA) 値を返します。XRBA 値は、各レコードの最初の 8 バイトに返されます。

注: XRBA 値を含めるために、レコードの先頭に 8 バイトのバイナリフィールドを含める必要があります。このオプションを選択し、レコードの先頭に 8 バイトのバイナリフィールドを含めない場合、データ取得時にデータは正しくマッピングされません。

ファイルリスト処理

データマップにファイルリスト処理を使用します。

データコードページ

データベースの文字データの文字セットを示すコードページです。 **[デフォルト]** コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。

デフォルトは **デフォルト** です。

ファイルの先頭の n つのレコードをスキップする

スキップするデータファイルのヘッダレコードの数です。例えば、データファイルがデータを含まないヘッダレコードを 1 つ以上含む場合、スキップするヘッダレコードの数を入力します。

0 から 2147483647 までの有効な値です。デフォルトは 0 です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップと複数のデータファイルの関連付け」 \(ページ 99\)](#)
- [「VSAM データセットの一括読み込みのパフォーマンス向上」 \(ページ 138\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - IDMS アクセス方式

IDMS データマップのプロパティを表示または編集します。

注: [サブスキーマ名] フィールドが必要です。それ以外のすべてのフィールドは、インストール構成および標準で必要とされない限り、オプションです。

サブスキーマ名

必須。サブスキーマの名前です。有効な文字は ASCII 文字、ポンドおよび円記号です。

DB 名

データベースの名前です。有効な文字は ASCII 文字、ポンドおよび円記号です。

プログラム名

プログラム識別文字列です。有効な文字は ASCII 文字、ポンドおよび円記号です。

DB ノード

Distributed Database System (DDS) ノード名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

ディクショナリ名

ディクショナリの名前です。有効な文字は ASCII 文字、ポンドおよび円記号です。

辞書ノード

DDS 辞書ノード名です。有効な文字は ASCII 文字、ポンドおよび円記号です。

レディモード

最初のリストから、プロテクションモードを選択します。次のオプションがあります。

- **排他。** 他のユーザーによる同時使用を防ぎます。
- **保護。** 実行ユニットが完了するまで、他のプロセスは更新できません。
- **共有。** ロックまたは保護は行われません。

デフォルトは**シェアード**です。

2 番目のリストから、データモードを選択します。次のオプションがあります。

- **取得。** データ取得モードです。
- **更新。** データ更新モードです。

デフォルトは**取得**です。

すべてレディ

すべてのエリアが準備完了であることを示します。

関連項目：

- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - KSDS アクセス方式

VSAM キーシーケンスのデータセット (KSDS) データマップのプロパティを表示または編集します。

ファイル名

VSAM エントリシーケンスのデータセット (KSDS) の完全修飾データセット名です。

データバッファ数

PowerExchange から VSAM に割り当てが要求される、データ制御間隔の I/O バッファの数です。デフォルトは 2 です。

インデックスバッファ数

PowerExchange から VSAM に割り当てが要求される、インデックス制御間隔の I/O バッファの数です。デフォルトは 1 です。

CI アクセス

VSAM データセットの読み取り時に、個々のデータレコードではなく制御間隔のコンテンツ全体にアクセスします。このオプションにより、読み込みのパフォーマンスが向上します。

VSAM データセットの読み込み時に個々のデータレコードのコンテンツにアクセスするには、このオプションの選択を取り消します。

注: データセットが圧縮されている場合、**[CI アクセス]** オプションは使用できません。

データコードページ

データベースの文字データの文字セットを示すコードページです。 **[デフォルト]** コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。

注: マルチバイトコードページは KSDS で使用可能です。

デフォルトは**デフォルト**です。

ファイルの先頭の *n* つのレコードをスキップする

スキップするデータファイルのヘッダレコードの数です。例えば、データファイルがデータを含まないヘッダレコードを 1 つ以上含む場合、スキップするヘッダレコードの数を入力します。

0 から 2147483647 までの有効な値です。デフォルトは 0 です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「VSAM データセットの一括読み込みのパフォーマンス向上」 \(ページ 138\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - MQSeries アクセス方式

MQSeries データマップのプロパティを表示または編集します。

キューマネージャ

必須です。MQ シリーズキューマネージャの名前です。

有効な文字は ASCII 文字、ポンドおよび円記号です。

キュー名

必須です。キューの名前です。

有効な文字は ASCII 文字、ポンドおよび円記号です。

注: キューマネージャ名とキュー名を入力した後、**【キューの表示】** をクリックしてデータファイルを表示します。

フィールド区切り記号

データファイルが文字区切りファイルの場合、このオプションを選択して、ファイルのフィールドを区切るために使用する文字を入力します。

フィールド区切り文字

フィールド内のデータ区切りに使用されている文字を入力します。例えば、二重引用符 (") を使用するとフィールド内の文字列を囲むことができます。

【フィールド区切り記号】 オプションを選択した場合に有効になります。

隣接するセパレータのマージ

連続するセパレータを 1 つのセパレータにマージします。.

【フィールド区切り記号】 オプションを選択した場合に有効になります。

エンコーディング

エンコードのスキームです。次のオプションがあります。

- **デフォルト。** PowerExchange では、オペレーティングシステムのデフォルトのエンコードスキームが使用されます。
- **EBCDIC。** EBCDIC です。
- **ASCII HIEND。** ASCII ハイエンディアンです。
- **ASCII LOEND。** ASCII リトルエンディアンです。

デフォルトは**デフォルト**です。

コードページ

データベースの文字データの文字セットを示すコードページです。 **【デフォルト】** コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。デフォルトは**デフォルト**です。

キューの表示

データファイルが表示されます。

キューマネージャとキュー名を入力した場合に有効になります。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - RRDS アクセス方式

VSAM 相対レコードデータセット (RRDS) データマップのプロパティを表示または編集します。

ファイル名

VSAM 相対レコードデータセット (RRDS) の完全修飾データセット名です。

インデックスバッファ数

PowerExchange から VSAM に割り当てが要求される、インデックス制御間隔の I/O バッファの数です。デフォルトは 1 です。

データバッファ数

PowerExchange から VSAM に割り当てが要求される、データ制御間隔の I/O バッファの数です。デフォルトは 2 です。

CI アクセス

VSAM データセットの読み取り時に、個々のデータレコードではなく制御間隔のコンテンツ全体にアクセスします。このオプションにより、読み込みのパフォーマンスが向上します。

VSAM データセットの読み込み時に個々のデータレコードのコンテンツにアクセスするには、このオプションの選択を取り消します。

注: データセットが圧縮されている場合、**[CI アクセス]** オプションは使用できません。

RRN 値を持つレコードにプレフィックスを付ける

VSAM データセットから読み込まれるすべてのレコードの相対レコード番号 (RRN) 値を返します。

注: RRN 値を含めるために、レコードの先頭に 8 バイトのバイナリフィールドを含める必要があります。

データコードページ

データベースの文字データの文字セットを示すコードページです。 **[デフォルト]** コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。デフォルトは **デフォルト** です。

ファイルの先頭の *n* つのレコードをスキップする

スキップするデータファイルのヘッダレコードの数です。例えば、データファイルの場合。例えば、データファイルがデータを含まないヘッダレコードを 1 つ以上含む場合、スキップするヘッダレコードの数を入力します。

0 から 2147483647 までの有効な値です。デフォルトは 0 です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「VSAM データセットの一括読み込みのパフォーマンス向上」 \(ページ 138\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマップのプロパティ - SEQ アクセス方式

フラットファイルまたはシーケンシャルデータセットに対して定義されるデータマップのプロパティを表示または編集します。

ファイル名

シーケンシャルデータセットまたはフラットファイルの完全パスとファイル名です。ファイルがローカルシステムにある場合、[参照] ボタンをクリックしてファイルを参照します。有効な文字は ASCII 文字、ポンドおよび円記号です。

注意事項:

- 名前を一重引用符で囲まないでください。PowerExchange は、指定された名前を完全修飾ファイル名として使用し、z/OS がユーザー ID をプレフィックスとして追加することを防ぎます。
- IBM i から Db2 ファイルをインポートする場合、インポートするファイルは単レコード形式のファイルである必要があります。

固定長

固定長バイナリデータファイルでは、このオプションを選択します。

[サイズ] ボックスに、レコード長を入力します。

固定長バイナリファイルは指定されたレコード長で正確に分けられます。

変数

可変長データファイルでは、このオプションを選択します。

[可変] リストで、以下のうち 1 つの可変長データファイルタイプを選択します。

- **VB12**。データ長は 2 バイト値で、長さに 2 バイト長が含まれます。
- **VB14**。データ長は 4 バイト値で、長さに 4 バイト長が含まれます。
- **VBX2**。データ長は 2 バイト値で、長さに 2 バイト長は含まれません。
- **VBX4**。データ長は 4 バイト値で、長さに 4 バイト長は含まれません。
- **VS**。可変長ストリームデータファイルです。このデータファイルでは、レコードの境界が存在しません。レコードはフィールドで区切られ、固定長ではありません。

このオプションは、UNIX または Windows システムと i5/OS または z/OS システムの間で転送したデータに使用できます。

UNIX または Windows から z/OS に転送したファイルでは、そのデータセットに定義した LRECL 値が UNIX または Windows ファイルの最長レコードの長さより長いことを確認します。UNIX または Windows から i5/OS に転送したファイルでは、Create Physical File (CRTPF) コマンドで指定したレコードの長さが UNIX または Windows ファイルの最長レコードの長さより長いことを確認します。

このオプションを選択した場合、[ファイルの先頭の *n* つのレコードをスキップする] データマッププロパティは使用できません。また、データベース行のテストを実行する際に、[データベース行のテスト] ダイアログボックスの [ファイルの先頭の *n* 行をスキップする] オプションを使用して、特定数の先頭行をスキップすることもできません。

- **VBP2**。VLI の長さの値には、4 バイトのプレフィックス、2 バイトの VLI、2 バイトの NULLS が含まれます。ブロックされない z/OS 変数と一致します。
- **VBM2**。VLI の長さの値はマイナスで、4 バイトのプレフィックス、2 バイトの VLI、2 バイトの NULL が含まれます。

- **VBB2**。VLI の長さの値には、4 バイトのプレフィックスと 4 バイトのブロック記述子が含まれます。ブロックされる z/OS 変数と一致します。ASCII システムに転送されるファイルに使用して、ブロック記述子とレコード記述子を保持します。

注: Linux、Windows、UNIX では、これらのファイルタイプを使用して可変長バイナリデータを処理または作成できます。VS を除くこれらすべてのファイルタイプで、データは次のいずれかの整数長フィールドによって先行されます。

- 16 ビット短整数 (2 バイト)
- 32 ビット整数 (4 バイト)

このフィールドに含まれる長さでは、実際のデータの長さが提供され、プレフィックスの長さが含まれる場合があります。

デフォルト

PowerExchange でオペレーティングシステムとファイル属性に基づいてファイルタイプが割り当てられるようにする場合は、このオプションを選択します。このオプションは、別のシステムで生成されたのではないメインフレームデータを読み込む場合に使用します。

[**デフォルト**] は、デフォルトで選択されています。

エンコード

エンコードのスキームです。次のオプションがあります。

- **デフォルト**。PowerExchange では、オペレーティングシステムのデフォルトのエンコードスキーマが使用されます。
- **EBCDIC**。EBCDIC です。
- **ASCII HIEND**。ASCII ハイエンディアンです。
- **ASCII LOEND**。ASCII リトルエンディアンです。

注: エンコードのスキームは、レコード記述子およびレコードのデータの両方に適用されます。

デフォルトは**デフォルト**です。

コードページ

データベースの文字データの文字セットを示すコードページです。[**デフォルト**] コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。

注: マルチバイトコードページは、シーケンシャルデータセットまたはフラットファイルで使用可能です。

デフォルトは**デフォルト**です。

フィールドセパレータ

データファイルが文字区切りファイルの場合、このオプションを選択して、データファイルのフィールドを区切るために使用する文字を入力します。

有効な文字は ASCII 文字、ポンドおよび円記号です。

フィールドセパレータが 16 進数の組み合わせである場合、次の形式で入力します。

x'nn'

隣接するセパレータのマージ

隣接するセパレータを 1 つのセパレータにマージします。

[**フィールドセパレータ**] オプションが選択されている場合に有効になります。

フィールド区切り文字

フィールド内のデータ区切りに使用される文字を入力します。例えば、二重引用符 (") を使用するとフィールド内の文字列を囲むことができます。

[フィールドセパレータ] オプションが選択されている場合に有効になります。

有効な文字は ASCII 文字、ポンドおよび円記号です。

ファイルの先頭の n つのレコードをスキップする

ファイルに PowerExchange の操作に不要な 1 つ以上のヘッダレコードがある場合、スキップするヘッダレコードの数を入力します。

有効な値は 0~2147483647 です。

デフォルトは 0 で、行はスキップされません。

ファイルリスト処理

データマップにファイルリスト処理を使用します。

ファイルの表示

データファイルが表示されます。

ファイル名を入力した後に有効になります。

バルクデータ移動用にシーケンシャルファイルのデータマッププロパティを定義する方法の詳細については、『PowerExchange バルクデータ移動ガイド』の「シーケンシャルファイルのバルクデータ移動」の章を参照してください。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップと複数のデータファイルの関連付け」 \(ページ 99\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - TAPE アクセス方式

テープデータセット用に定義されるデータマップのプロパティを表示または編集します。

ファイル名

z/OS のテープデータセットの完全パスとファイル名です。この名前は、GDG データセット名になることがあります。

注: 一重引用符で名前を囲まないでください。PowerExchange では指定された名前が完全修飾ファイル名として使用され、z/OS にユーザー ID がプレフィックスとして追加されるのを防ぎます。

有効な文字は ASCII 文字、ポンドおよび円記号です。

フィールド区切り記号

データファイルが文字区切りファイルの場合、このオプションを選択して、データファイルのフィールドを区切るために使用する文字を入力します。

有効な文字は ASCII 文字、ポンドおよび円記号です。

フィールド区切り記号が 16 進数の組み合わせである場合、次の形式で入力します。

x'nn'

隣接するセパレータのマージ

連続するセパレータを 1 つのセパレータにマージします。

【フィールド区切り記号】 オプションを選択した場合に有効になります。

位置

テープのファイルの 1 に対する相対位置です。

この値は DBMOVE コンフィギュレーションメンバの RELPOS パラメータの値と一致する必要があります。

単位

この値は DBMOVE コンフィギュレーションメンバの UNIT パラメータの値と一致する必要があります。

ボリュームシリアル

この値は DBMOVE コンフィギュレーションメンバの VOLSER パラメータの値と一致する必要があります。

ファイルの先頭の n つのレコードをスキップする

ファイルに 1 つ以上のヘッダレコードがある場合、スキップするヘッダレコードの数を入力します。

0 から 2147483647 までの有効な値です。

デフォルトは 0 です。

ファイルリスト処理

このデータマップにファイルリスト処理を使用します。

関連項目：

- [「データマップと複数のデータファイルの関連付け」 \(ページ 99\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - USER アクセス方式

ユーザーアクセス方法プログラムに対して定義されるデータマップのプロパティを表示または編集します。

出口プログラム名

ランタイム時に呼び出されるユーザーアクセス方法プログラムの名前です。このプログラムは、有効な DLL または共有ライブラリである必要があります。

呼び出し規約

ユーザーアクセス方法プログラムの呼び出し規約です。次のオプションがあります。

- **C**。標準 C 呼び出しです。
- **ASM**。IBM Assembler タイプパラメータリストです。
- **PLI**。PL/I ユーザープログラムです。

ファイル名

ユーザーアクセス方法プログラムによって処理されるデータファイルの完全パスとファイル名です。ファイルがローカルシステムにある場合、[参照] ボタンをクリックしてファイルを参照します。

有効な文字は ASCII 文字、ポンドおよび円記号です。

文字列 1

ランタイム時にユーザーアクセス方法プログラムに渡す文字列です。

整数 1

ランタイム時にユーザーアクセス方式プログラムに渡す整数値です。

文字列 2

ランタイム時にユーザーアクセス方法プログラムに渡す文字列です。

整数 2

ランタイム時にユーザーアクセス方式プログラムに渡す整数値です。

エンコーディング

エンコードのスキームです。次のオプションがあります。

- **デフォルト**。PowerExchange では、オペレーティングシステムのデフォルトのエンコードスキーマが使用されます。
- **EBCDIC**。EBCDIC です。
- **ASCII HIEND**。ASCII ハイエンディアンです。
- **ASCII LOEND**。ASCII リトルエンディアンです。

注: エンコードのスキームは、レコード記述子およびレコードのデータの両方に適用されます。

デフォルトは**デフォルト**です。

コードページ

データファイルの文字データの文字セットを示すコードページです。[デフォルト] コードページでは、PowerExchange により、オペレーティングシステムのデフォルトコードページが使用されます。デフォルトは**デフォルト**です。

ファイルの先頭の n つのレコードをスキップする

データファイルに 1 つ以上の不要なヘッダレコードがある場合に、スキップするヘッダレコードの数です。

0 から 2147483647 までの有効な値です。デフォルトは 0 です。

関連項目：

- [「コードページ」 \(ページ 61\)](#)
- [「データマップの追加」 \(ページ 66\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - データのデフォルト値

データマップのデータデフォルトプロパティを表示または編集します。

日付のデフォルト設定

世紀、年、月、日についてのデフォルトの日付設定です。フィールドマスクを定義し、マスクの値が欠けている場合、欠落値はこれらの設定でデフォルトに設定されます。例えば、フィールドマスクを Y2-MM として定義し、99-04 のインプットデータを提供する場合、世紀と日の値は欠けています。次のように、PowerExchange では、デフォルトの世紀と日の値にそれぞれ 19 と 01 を使用して日付フィールドが構築されます。

19990401

次のデフォルトの日付設定があります。

- 世紀: 19
- 年: 80
- 月: 1
- 日: 1

欠けている世紀

欠けている世紀の値に対し、次のうち 1 つのアクションを指定します。

- **デフォルト。** デフォルトアクションを完了します。
- **固定。** 年を **[Pivot]** 値と照合します。入力した年の値が **[Pivot]** 値より小さい場合は、世紀は 20 と仮定されます。それ以外の場合は、19 と仮定されます
- **移動可能。** 入力した年から現在の年の値を減算して、バウンダリ値を計算します。バウンダリが 0 未満の場合は、バウンダリ値に 100 が追加されます。 **[Pivot]** 値がバウンダリ値より小さい場合は、世紀は 20 と仮定されます。それ以外の場合は、19 と仮定されます

文字の置き換え

データチェックが CHAR フィールドで有効で、フィールドのデータがデータチェックに失敗した場合、**[値]** ボックスの文字は、次のように、フィールドの最初の文字またはすべての文字のどちらかに置き換えられます。

- **単一文字。** フィールドの最初の文字です。
- **すべての文字。** フィールドのすべての文字です。

キーへの厳格な署名

データが適切なデータタイプの正しい署名規約と正確に一致する場合、そのデータを返します。例えば、SPACKED です。

関連項目：

- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマップのプロパティ - データチェック

データマップのデータチェックプロパティを表示または編集します。

レコードマッピングエラーをどのように処理しますか？

次のうち 1 つのオプションを選択し、PowerExchange によるレコードマッピングエラーの処理方法を指定します。

- **行をスキップし、ログファイルメッセージを書き込む。** 行をスキップして、メッセージをログファイルに書き込み、抽出処理を続行します。
- **データ抽出の終了。** 抽出処理を停止します。

デフォルトは**データ抽出の終了**です。

不正なフィールドデータをどのように処理しますか？

次のうち 1 つのオプションを選択し、フィールドで無効なデータが PowerExchange により処理される方法を指定します。

- **データのデフォルト値に従った置換。** CHAR または DATE フィールドでは、[データマップのプロパティ] ダイアログボックスの [データのデフォルト値] タブの値を使用してデータを置換します。フィールドの最初の文字またはすべての文字のどちらを置換するかを指定できます。PowerExchange では、数字フィールドの値はゼロに設定されています。
- **行をスキップし、ログファイルメッセージを書き込む。** 行をスキップして、メッセージをログファイルに書き込み、抽出処理を続行します。
- **データ抽出の終了。** 抽出処理を停止します。
- **NULL への設定。** フィールドを NULL に設定します。

デフォルトは**データ抽出の終了**です。

フィールドの型のチェック

検証を実行するフィールドの型を選択します。PowerExchange により、以下の種類のフィールドの型の検証が実行されます。

CHAR

EBCDIC および ASCII 文字セットでは、PowerExchange により、次の文字のみを含むフィールドが検証されます。

- 0～9
- A～Z
- a～z
- 大なり記号 (>)
- 小なり記号 (<)
- 等号 (=)
- スペース ()

DATE

PowerExchange により、PowerExchange 定義 DATE フォーマットを使用するフィールドが検証されます。

NUMCHAR

PowerExchange により、次の文字のみを含むフィールドが検証されます。

- 0～9
- アスタリスク (*)
- カンマ (,)
- 小数点 (.)
- ドル記号 (\$)
- フォワードスラッシュ (/)
- マイナス記号 (-)
- プラス記号 (+)
- スペース ()

PACKED

PowerExchange により、標準パック 10 進数形式 (S9 COMP-3) の文字のみを含むフィールドが検証されます。

注意: このフィールドの型に対してデータチェックを有効にせず、最後の半バイト (記号を含む) 以外のデータの半バイトに 0～9 以外の値が含まれる場合、PowerExchange はそのフィールドに NULL 値が含まれると解釈します。セッションが非リレーショナルターゲットに書き込み、ターゲットのデータマップが NULL 値を許可しない場合、セッションが失敗します。

例えば、パック 10 進数のデータチェックが無効で、ソースデータの 2 バイトのパック 10 進数フィールドが PIC S9(3) COMP-3 として定義されている場合、PowerExchange によって次の値が PowerCenter に渡されます。

- X'FFFF' - NULL が PowerCenter に渡されます
- X'0000' - 0 が PowerCenter に渡されます
- X'4040' - 404 が PowerCenter に渡されます
- X'F00C' - NULL が PowerCenter に渡されます

セッションの失敗を防ぐには、パック 10 進数フィールドのデータチェックを有効にし、**[不正なフィールドデータをどのように処理しますか?]** で、状況に適したオプションを選択します。

TIME

PowerExchange により、PowerExchange 定義 TIME フォーマットを使用するフィールドが検証されます。

TIMESTAMP

PowerExchange により、PowerExchange 定義 TIMESTAMP フォーマットを使用するフィールドが検証されます。

ZONED

PowerExchange により、ゾーン数値、0 から 9、および記号 (-) のみを使用するフィールドが検証されます。

データログの非表示

データログを非表示にするには、このオプションを選択します。

デフォルトでは選択されていません。

関連項目：

- [「データチェック」 \(ページ 63\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - レコードレベルの出口

データマップに対して定義されたレコードレベルの出口プログラムのプロパティを表示または編集します。

出口プログラム名

ランタイム時に呼び出されるレコードレベルの出口プログラムの名前です。このプログラムは、オペレーティングシステムに応じて、有効な DLL または共有ライブラリである必要があります。

呼び出し規約

レコードレベルの出口プログラムの呼び出し規約です。次のオプションがあります。

- **C**。標準 C 呼び出しです。
- **OS**。IBM Assembler タイプパラメータリストです。
- **PLI**。PL/I ユーザープログラムです。
- **COBOL**。COBOL ユーザープログラムです。

デフォルトは **C** です。

文字列 1

ランタイム時にレコードレベルの出口プログラムに渡される文字列値です。

整数 1

ランタイム時にレコードレベルの出口プログラムに渡される整数値です。

文字列 2

ランタイム時にレコードレベルの出口プログラムに渡される文字列値です。

整数 2

ランタイム時にレコードレベルの出口プログラムに渡される整数値です。

関連項目：

- [「レコードレベルの出口プログラム」 \(ページ 56\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

データマッププロパティ - キー

Adabas データマップのキープロパティを表示または編集します。

記述子

FDT と共にインポートされる、Abadas キーの記述子です。

コンポーネントフィールド

FDT と共にインポートされる、Abadas キーのコンポーネントフィールドです。

関連項目：

- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

付録 E

レコード、フィールドおよびテーブルのプロパティ

この付録では、以下の項目について説明します。

- [レコードプロパティ - 名前, 375 ページ](#)
- [レコードプロパティ - 名前 \(Datacom\) , 376 ページ](#)
- [レコードプロパティ - 名前 \(IDMS\) , 377 ページ](#)
- [\[並べ替えエレメントの詳細\] ダイアログボックス \(IDMS\) , 379 ページ](#)
- [レコードのプロパティ - インポートの詳細, 380 ページ](#)
- [フィールドプロパティ - 名前, 380 ページ](#)
- [フィールドプロパティ - 配列, 387 ページ](#)
- [フィールドプロパティ - 追加のプロパティ, 388 ページ](#)
- [フィールドプロパティ - フィールドチェック, 389 ページ](#)
- [フィールドプロパティ - コードページ, 389 ページ](#)
- [フィールドプロパティ - 可変長, 390 ページ](#)
- [テーブルプロパティ - 定義, 391 ページ](#)
- [テーブルプロパティ - フィルタ, 394 ページ](#)
- [テーブルのプロパティ - IMS オプション, 394 ページ](#)
- [テーブルプロパティ - キー, 395 ページ](#)
- [検索ダイアログボックス, 396 ページ](#)

レコードプロパティ - 名前

レコードの表示または編集プロパティ

レコード名を編集できます。IMS、SEQ、および VSAM レコードに対してはそのほかのプロパティも編集できます。

レコード名

必須です。ユーザー定義のレコード名です。

最大長は 256 文字です。

有効な文字は ASCII 文字、ポンドおよび円記号です。

セグメント名

IMS DL/1 パッチまたは ODBA の場合。必須です。IMS DBD からのセグメント名です。

可変長

IMS DL/1 パッチまたは ODBA の場合。指定されたセグメントが可変長である場合は、このオプションを選択します。

デフォルトでは選択されていません。

親セグメント名

IMS DL/1 パッチまたは ODBA の場合。親セグメント名です。

ルートセグメント

IMS DL/1 パッチまたは ODBA の場合。ルートセグメントの下に、直接指定されたセグメントを位置付けます。

【親セグメント名】ボックスに親セグメント名を入力するには、このオプションの選択を取り消します。

デフォルトでは選択されていません。

親レコード名

SEQ または VSAM でオプションです。親レコード名を選択します。現在選択されているレコード、現在選択されているレコードのすべての子、ベースレコードを再定義するすべてのレコード、すべてのトレーラまたはヘッダレコードは選択できません。

再定義レコード名

SEQ または VSAM でオプションです。現在のレコードを再定義するベースレコードの名前を選択します。現在選択されているレコード、またはベースレコードを再定義するすべてのレコードは選択できません。

レコードが他のレコードによって再定義されている場合、このオプションは使用できません。

ヘッダ/トレーラレコード

SEQ または VSAM でオプションです。**HEADER** または **TRAILER** を選択します。PowerCenter Integration Service マシンで実行されるキャッシュアルゴリズムは、この値を使用して複数レコードの書き込み処理のパフォーマンスを向上させます。

レコードが親であることが判別された場合、このオプションは使用できません。

関連項目：

- [「データマップレコードの追加」 \(ページ 69\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

レコードプロパティ - 名前 (Datacom)

Datacom レコードプロパティを表示または編集します。

長いテーブル名

必須です。テーブルの長い名前です。最大長は 256 文字です。有効な文字は ASCII 文字、ポンドおよび円記号です。

短いテーブル名

必須です。テーブルの短い名前です。最大長は 3 文字です。有効な文字は ASCII 文字、ポンドおよび円記号です。

テーブル ID

必須です。内部テーブル ID です。最低でも 1 を入力する必要があります。デフォルトは 0 です。

テーブルリカバリ

このテーブルの CA DATACOM/DB のリカバリ機能を使用します。デフォルトでは選択されていません。

テーブルの状態

必須です。このテーブルを定義したデータディクショナリ出現のステータスを、以下の値のうち 1 つ選択します。

- 履歴
- 製品
- テスト

デフォルトは「なし」です。

テーブルのバージョン

テーブルのバージョンです。履歴またはテストのステータスでテーブルのテーブルバージョンを入力できます。デフォルトは 1 です。

関連項目：

- [「データマップレコードの追加」 \(ページ 69\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

レコードプロパティ - 名前 (IDMS)

IDMS レコードプロパティを表示または編集します。

レコード名

必須です。ユーザー定義のレコード名です。

最大長は 256 文字です。有効な文字は ASCII 文字、ポンドおよび円記号です。

IDMS レコード名

必須です。レコードの IDMS 名です。

領域名

必須です。レコードのある IDMS 領域の名前です。

レコード ID

オプションです。スキーマ内のレコードタイプを特定する一意のレコード ID です。

有効な文字は ASCII 文字、ポンドおよび円記号です。

場所モード

オプションです。レコードタイプの出現を物理的に格納するために IDMS で使用する方法を選択します。

次のオプションがあります。

- **CALC**
- **DIRECT**
- **VIA**
- **VSAM**
- **VSAM CALC**

重複

[LocationMode] リストの [**CALC**] または [**VSAM CALC**] を選択した場合に有効になります。

[**CALC**] には、次のオプションがあります。

- **DB キー順**
- **First**
- **Last**
- **許可されていません**

[**VSAM CALC**] には、次のオプションがあります。

- **許可されていません**
- **順序指定の解除済み**

VSAM の種類

[LocationMode] リストの [**VSAM CALC**] を選択した場合に有効になります。レコードをネイティブ VSAM レコードとして特定し、そのレコードを含むファイルのファイルタイプを示します。

次のオプションがあります。

- **固定スパン**
- **固定非スパン**
- **可変スパン**
- **可変非スパン**

ページグループ

レコードの IDMS ページグループです。

最大値は 32767 です。デフォルトは 0 です。

基数

レコードの IDMS 基数です。

0 から 12 までの有効な値です。デフォルトは 8 です。

最小ルート長

圧縮レコードの最小ルート長です。デフォルトは 0 です。

データ長

圧縮レコードの固定長レコードのデータ長です。デフォルトは 1 です。

プレフィックス長

圧縮レコードのプレフィックスの長さです。デフォルトは 1 です。

制御部分長

圧縮レコードのプレフィックスを除く制御部分長です。デフォルトは0です。

圧縮

IDMS でレコードを圧縮形式で格納する場合は、このオプションを選択します。デフォルトでは選択されていません。

変数

レコードが可変長レコードである場合は、このオプションを選択します。デフォルトでは選択されていません。

関連項目：

- [「データマップレコードの追加」 \(ページ 69\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

[並べ替えエレメントの詳細] ダイアログボックス (IDMS)

IDMS のオーナーレコードとセットのソートエレメント詳細を表示または編集します。

名前

フィールドエレメント名です。

位置

レコードの開始に対するフィールドエレメントの位置です。デフォルトは1です。

長さ

フィールドエレメントの長さです。デフォルトは1です。

順序

ソート順です。次のオプションがあります。

- なし。
- **ASC**。昇順です。
- **DESC**。降順です。

デフォルトは「なし」です。

関連項目：

- [「データマップレコードの追加」 \(ページ 69\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

レコードのプロパティ - インポートの詳細

レコードのインポートの詳細を表示または編集します。

最後にインポートされたコピーブック

データマップに最後にインポートされたコピーブックの完全パスとファイル名を表示します。

最後にインポートされた DBD

IMS です。データマップに最後にインポートされた DBD の完全パスとファイル名を表示します。

IMS セグメント処理規則

IMS です。IMS DBD からインポートした RULES キーワードの値を表示します。

例えば、DBD が RULES=(,LAST)を指定している場合は、このフィールドでは LAST の値が表示されます。

関連項目：

- [「データマップレコードの追加」 \(ページ 69\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

フィールドプロパティ - 名前

フィールドプロパティを表示または編集します。

フィールド名

フィールドのユーザー定義名です。フィールド名は、レコード内で一意である必要があります。

このフィールドに基づいてテーブルに列を追加する場合は、このフィールド名を参照します。

有効な文字は ASCII 英数字、ポンドおよび円記号です。最初の文字はアルファベット文字である必要があります。最大長は 30 文字です。

Adabas 名 (Adabas のみ)

フィールドの Adabas 名です。

フィールドの型、精度、位取り、長さ

PowerExchange で使用されるフィールドの型、およびそのフィールドの型で有効な精度、位取り、長さの値です。一部のフィールドの型では、精度、位取り、または長さを編集できます。

ヒント: メタデータが DB2 カタログまたは DB2 アンロードファイルからインポートされた DB2 10 for z/OS の TIMESTAMP 列のフィールドプロパティは編集しないでください。

以下の表に、精度、位取り、および長さに対して入力可能な値をフィールドの型ごとに示します。

フィールドの型	精度	スケール	長さ
BIN	-	-	1~32767
BLOB	-	-	1~8000000
CHAR	-	-	1~32767

フィールドの型	精度	スケール	長さ
CLOB	-	-	1~8000000
DOUBLE	-	-	-
FLOAT	-	-	-
GROUP	-	-	-
NUM16	-	0~5	-
NUM16U	-	0~5	-
NUM32	-	0~10	-
NUM32U	-	0~10	-
NUM64	-	0~19	-
NUM64U	-	0~19	-
NUM8	-	0~3	-
NUM8U	-	0~3	-
NUMCHAR	1~100	0~50	1~100
PACKED	1~31	0~31	-
SPACKED	1~31	0~31	-
UNPACKED	1~31	0~31	-
UZONED	1~31	0~31	-
VARBIN	-	-	1~32767
VARCHAR	-	-	1~32767
ZONED	1~31	0~31	-

注: PowerCenter の Decimal トランスフォーメーションデータ型にマップされる PACKED フィールド型など、特定のフィールド型では、精度は 31 まで指定できます。ただし、セッションプロパティで高精度を有効にすると、Decimal データ型では最大 28 の精度がサポートされます。高精度を有効にしなかった場合は、最大 16 の精度がサポートされます。28 より大きい精度を指定した場合は、Decimal ではなく、精度が 16 の Double のトランスフォーメーションデータ型が使用されます。

z/OS データソースでは、フィールドの型は COBOL PIC 句に対応します。以下の表に、フィールドの型ごとの COBOL PIC 句、およびその PIC 句に基づく精度、位取り、固定長を示します。

フィールドの型	COBOL PIC 句	最大精度	スケール	固定長
BIN	X(nn)	-	-	32767
CHAR	X(nn)	-	-	32767
DOUBLE	COMP-2	-	-	8
FLOAT	COMP-1	-	-	4
GROUP	X(nn)	-	-	-
NUM16	S9(4) COMP	4	0～5	2
NUM16	S9(4) COMP-5	4	0～5	2
NUM16U	9(4) COMP	4	0～5	2
NUM16U	9(4) COMP-5	4	0～5	2
NUM32	S9(9) COMP	9	0～10	4
NUM32	S9(9) COMP-5	9	0～10	4
NUM32U	9(9) COMP	9	0～10	4
NUM32U	9(9) COMP-5	9	0～10	4
NUM64	S9(18) COMP	19	0～19	8
NUM64	S9(18) COMP-5	19	0～19	8
NUM64U	9(18) COMP	20	0～19	8
NUM64U	9(18) COMP-5	20	0～19	8
NUMCHAR	9(nn)	100	0～50	50
PACKED ¹	S9(nn) COMP-3	31	0～31	-
SPACKED ²	-	31	0～31	-
UPACKED ³	9(nn) COMP-3	31	0～31	-
UZONED	9(nn)	31	0～31	-
VARBIN	2 バイト長のフィールドで設定された X(nn)	-	-	32767
VARCHAR	2 バイト長のフィールドで設定された X(nn)	-	-	32767

フィールドの型	COBOL PIC 句	最大精度	スケール	固定長
VARCHAR	2 バイト長のフィールドで設定された X(nn)	-	-	32767
ZONED	9(nn)	31	0～31	-

1. PACKED フィールドは、MVS の標準的なパックされたバイナリ 10 進数フィールドです。正の値を示すために C、A、または F ニブルが含まれるか、負の値を示すために B または D ニブルが含まれます。X'12345C' のような奇数の精度や、X'01234C' のような偶数の精度を持つことができます。

2. SPACKED フィールドは、サインニブルのない短い数字です。日付などの情報を格納するために使用されます。X'012345' のような奇数の精度や、X'1234' のような偶数の精度を持つことができます。

3. UPACKED フィールドは、符号のない PACKED フィールドです。ただし、依然として F ニブルは含まれています。X'012345F' のような奇数の精度や、X'01234F' のような偶数の精度を持つことができます。

暗黙の小数点

実際には含まない小数点を示す NUMCHAR フィールドに対してこのオプションを選択します。このオプションを選択することによって、PowerExchange により、フィールドのデータが整列されます。デフォルトでは選択されていません。

フィールドフォーマット

CHAR、VARCHAR、ZONED、UZONED、PACKED、SPACKED、または UPACKED のフィールドの型では、入力データの形式を定義し、適切な出力フィールドを生成する文字マスクを提供します。日付、時刻、またはタイムスタンプの形式を選択できます。

日付形式には、次のものがあります。

- D2-MM-Y2
- D2-MM-Y4
- MM-D2-Y2
- MM-D2-Y4
- Y2-MM-D2
- Y4-MM-D2

時刻形式には、次のものがあります。

- HH-MI-SA
- HH-MI-SS

タイムスタンプ形式には、次のものがあります。

- Y4-MM-D2-HH.MI.SS.N6
- Y4-MM-D2-HH.MI.SS.N9
- Y4-MM-D2-HH.MI.SS.N0～Y4-MM-D2-HH.MI.SS.N12 (DB2 10 for z/OS の拡張精度のタイムスタンプ)
- Y4MMD2HHMISSN6
- Y4MMD2HHMISSN9
- Y4MMD2HHMISSN0～Y4MMD2HHMISSN12 (DB2 10 for z/OS の拡張精度のタイムスタンプ)

注: データマップ用にインポートした DB2 カタログまたはアンロードファイルに基づく、DB2 V10 for z/OS の拡張精度のタイムスタンプは編集しないでください。

【フィールドフォーマット】 リストの右の矢印をクリックして、特定のマスク文字を挿入します。

以下の表に、フィールドフォーマットの文字マスクを示します。また、この表に、フィールドフォーマットに追加できる追加の文字マスクを一覧表示します。

文字マスク	説明	有効な入力値
D2	日	1～31
D3	ユリウス日	1～366 例えば、001 は 1 月 1 日で、032 は 2 月 1 日です。
HH	時間	-
HH12	12 時間形式の時間	1～12
HH24	24 時間形式の時間	0～23
LA	長い形式での、午前または午後のインジケータ	a.m.または p.m.
LM	月（フルスペル）	January、February など
MI	分	0～60
MM	月	1～12
NATDATE	Adabas Natura 日付	-
NATTIME	Adabas Natural 時間	-
N n	小数秒の数	n 値は 1 から 6 までの数字です。
SA	短い形式での、午前または午後のインジケータ	am または pm
SM	月（省略名）	フルスペルの月名の最初の 3 文字 (APR、SEP など)
SS	秒。	0～60
Y2	2 桁の年	00～99
Y4	世紀と年を含む 4 桁の年	0000～9999
YYY	PACKED フィールドの年と世紀のインジケータ (20th、21st などの、th、st)	例: 990823 は、1999 年 8 月 23 日 1000914 は、2000 年 9 月 14 日 1010129 は、2001 年 1 月 29 日

以下の表に、フィールドフォーマット例および対応する入力値と出力値を示します。

フォーマット例	入力	出力
D2-MM-Y2 HH:MI/N3	02-07-09 02:45/123	20090702024500123000
Y2-SM-D2	09-APR-01	20090401
HH:MI:SS LA	01:30:23 p.m.	133023
Y2D3	96366	19990401

注: 文字マスクの入力時に、次のガイドラインを使用します。

- フィールドが数字以外である場合、マスクの英数字を使用します。
- 文字マスクでは、スペースと入力の文字を一致させます。
- マスクが入力フィールドよりも長い場合は、PowerExchange により、入力フィールドに先頭のゼロが追加され、マスクの長さになるように入力フィールドがパディングされます。例えば、マスクが 6 文字の長さ（D2MMY2）であり、入力フィールドが 5 バイトの長さである場合、PowerExchange により、先頭のゼロが追加され、入力フィールドが 6 文字になるように入力フィールドがパディングされます。
- マスクから文字が不足している場合は、PowerExchange により、**【データマップのプロパティ】** ダイアログボックスの **【データのデフォルト値】** タブから値が使用されて欠落値を満たします。

レコード ID の値

フィールドに 1 つ以上のレコード ID 値を割り当てます。これは、PowerExchange により、ランタイム時にレコードをフィルタリングするための演算子と共に使用されます。

【レコード ID 値】 リストで、**[=]** または **[<>]** ボタンを選択して、データをフィルタリングするための演算子を選択します。

- **[=]** ボタンをクリックして、レコード ID 値が指定されたレコード ID 値と等しいレコードを表示します。
- **[<>]** ボタンをクリックして、レコード ID 値が指定されたレコード ID 値より大きい小さい（等しくない）レコードを表示します。

注: データマップには、データではなくレコード ID が格納されます。**【レコード ID】** フィールドは、データの列ビューに表示されない場合があります。WRITE ではレコード ID フィルタリングを使用できません。

16 進数のレコード ID の値「0000」を入力するには、レコード ID として次の値を入力します。

X'0000'

オプションのフィールド

非リレーショナルソースと非リレーショナルターゲットのバッチセッションでは、NRDB マッピング処理でさまざまなサイズのレコードを処理できるようにするためにこのオプションを選択します。**【オプションのフィールド】** を選択した場合、次のようになります。

- マッピング処理がオプションのフィールドを含まない短いレコードを読み取った場合、結果セットの対応するリレーショナルカラムは NULL に設定されます。
- 対応するリレーショナルカラムが NULL の場合、マッピング処理はオプションのフィールドがターゲットに書き込まれないようなレコード長を設定します。
- 空のレコードが許可されない非リレーショナルデータ型では、最初のフィールドをオプションのフィールドにすることはできません。**【オプションのフィールド】** が選択されている場合は無視されます。
- オプションのフィールドは、可変長のレコードの末尾に定義できます。

- あるフィールドがオプションと設定されている場合、同じレコード内の以降のフィールドもオプションに設定する必要があります。

デフォルトでは、このオプションは選択されていません。NRDB マッピング処理は、レコード内にこのフィールドが必ず存在することと想定しています。

- 読み取り時には、レコード長がオプションでないフィールドを保持するには短すぎる場合、NRDB マッピング処理は失敗します。
- 書き込み時には、フィールドを保持するのに十分な長さにマッピング処理がレコード長を設定します。対応するリレーショナルカラムが NULL の場合、デフォルト値が書き込まれます。

データマップでレコード長が固定として定義されている場合、またはマッププロパティがオペレーティングシステムのデフォルトのファイルタイプを使用していてそのファイルタイプが固定長の場合（z/OS での RECFM=FB ファイルなど）は、いずれのフィールドもオプションとして定義できません。

オプションのフィールドは、全体が可変長のレコードに存在するか、まったく存在しないかのいずれかである必要があります。レコード境界が ZONED などの固定長フィールド型内にある場合、マッピングは失敗します。

NULL 可能フィールド

このオプションは、マッピング処理がレコードを書き込むときに、NRDB レコードのフィールドに値を入力するために使用されているリレーショナルカラムが NULL 値を受け取れるかどうかを制御します。マッピング処理がレコードを読み取る場合には影響しません。

PowerCenter や Publisher などのソースのカラム値を使用するターゲットに書き込む際にフィールドに NULL 値を許可するには、このオプションを選択します。このオプションを指定すると、次のようになります。

- フィールドが **【オプションのフィールド】** に設定されている場合、NULL 値のデータを受け取ると、オプションのフィールドが含まれない短いレコードが書き込まれます。
- フィールドがオプションと設定されていない場合、NULL 値データを受け取ると、フィールドにはデータ型に応じて *Not Available* 値（スペースやバイナリゼロ）が入力されます。

デフォルトでは、このオプションは選択されていません。フィールドに入力する処理では、必ず値が提供される必要があります。

【NULL 不可】 に設定されているカラムに NULL 値を書き込もうとすると、リレーショナルデータ検証プロセスで失敗し、その行は NRDB 書き込みエンジンに送信されません。

DB2 アンロードファイルデータマップのフィールドには次のプロパティが表示されます。

Null フィールド制御

【フィールド名】 ボックスを有効にします。デフォルトでは選択されていません。

フィールド名

【Null フィールド制御】 オプションを選択した場合、**【フィールド名】** ボックスでは、そのフィールドの Null インジケータを含む制御フィールドが表示されるか、その制御フィールドに入力できるようになります。

テーブルから非表示

テーブルウィンドウのこのフィールドに基づく列をすべて非表示にします。デフォルトでは選択されていません。

関連項目：

- [「レコードへのフィールドの追加」](#)（ページ 70）
- [「レコード内のフィールドのプロパティの表示または編集」](#)（ページ 87）

フィールドプロパティ - 配列

フィールドプロパティを表示または編集して、フィールドを配列として定義することができます。
使用可能なプロパティは、データマップのデータソースによって異なります。

非 Adabas フィールド

非 Adabas データマップのフィールドの以下のプロパティを表示または編集します。

配列

フィールドが配列である場合は、このオプションを選択します。配列は OCCURS 文を使用して COBOL で定義されます。デフォルトでは選択されていません。

このオプションを選択する場合、次のうち 1 つのオプションを選択します。

固定

配列が固定長配列 (OCCURS x) である場合は、このオプションを選択します。また、**【配列サイズ】** ボックスに配列サイズを入力します。

変数

配列のエレメント数がレコードの別のフィールドによって異なる場合 (OCCURS x DEPENDING ON フィールド- y)、このオプションを選択します。また、**【配列サイズ】** ボックスに配列サイズを入力し、**【カウントフィールド】** ボックスにカウントサイズを入力します。

注: VARCHAR フィールドでは、**【フィールドのプロパティ】** ダイアログボックスの **【名前】** タブの **【オプションのフィールド】** チェックボックスを選択します。

デフォルトは**固定**です。

配列サイズ

配列のサイズです。**【配列】** オプションが選択されている場合に必要となります。

デフォルトは 1 です。

カウントサイズ

カウントサイズです。可変長配列で必要となります。

関連項目：

- [「レコードへのフィールドの追加」 \(ページ 70\)](#)
- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)

Adabas フィールド

Adabas データマップのフィールドの以下のプロパティを表示または編集します。

定期的なグループ/複数値フィールド

フィールドが配列で、そのフィールドが PE または MU フィールドである場合は、このオプションを選択します。

デフォルトでは選択されていません。

このオプションを選択する場合、次のうち 1 つのオプションを選択します。

未使用の値へのパディング

適切な文字を持つ配列の任意の未使用値をパディングします。例えば、数字フィールドでは、未使用値はゼロでパディングされます。

未使用の値を Null に設定する

配列の任意の未使用値を NULL に設定します。

デフォルトは**未使用の値を Null に設定する**です。

配列サイズ

配列のサイズです。デフォルトは 1 です。

配列のオーバーフローをマッピングエラーとして扱う

実際の配列が **【配列サイズ】** ボックスで指定された値よりも大きい場合は、エラーを生成します。

【配列サイズ】 ボックスで指定されたエレメント数を返すには、このオプションの選択を取り消します。デフォルトでは選択されていません。

関連項目：

- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)

フィールドプロパティ - 追加のプロパティ

フィールドの追加のプロパティを表示します。

使用方法

インポートされた COBOL コピーブックで指定されたフィールドの COMP または DISPLAY などのフィールドの使用方法です。

レベル

インポートされた COBOL コピーブックのフィールドのインデントレベルです。

元の名前

インポートされた COBOL コピーブックで指定されたフィールドの元の名前です。

PICTURE

DB2 以外のすべてのデータソースに対するデータマップのフィールド用です。フィールドの COBOL ピクチャ形式です。

DB2 データタイプ

DB2 データソースに対するデータマップのフィールド用です。フィールドのデータタイプを表示します。

関連項目：

- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)
- [「レコードへのフィールドの追加」 \(ページ 70\)](#)

フィールドプロパティ - フィールドチェック

フィールドのフィールドチェックプロパティを表示または編集します。

チェックするフィールドを選択してください。

次のうち 1 つのオプションを選択し、PowerExchange でフィールドのデータチェックを完了する方法を指定します。

- **データマップ規則の使用。** データマップに対して定義されたデータチェックプロパティを使用します。
- **はい。** フィールドチェックを使用します。
- **いいえ。** フィールドチェックは使用しません。

デフォルトは**データマップ規則の使用**です。

フィールドエラーをどのように処理しますか。

次のうち 1 つのオプションを選択し、PowerExchange によるフィールドデータのエラーの処理方法を指定します。

- **データマップ規則の使用。** データマップに対して定義されたデータチェックプロパティを使用します。
- **デフォルト値との置換。** CHAR フィールドでは、[**データマップのプロパティ**] ダイアログボックスの [**データのデフォルト値**] タブの値を使用してデータを置換します。フィールドの最初の文字またはすべての文字のどちらを置換するかを指定できます。PowerExchange では、数字フィールドの値はゼロに設定されています。
- **フィールドをスキップし、ログファイルを書き込む。** フィールドをスキップして、メッセージをログファイルに書き込み、抽出処理を続行します。
- **データ抽出の終了。** 抽出処理を停止します。
- **NULL への設定。** フィールドを NULL に設定します。

デフォルトは**データマップ規則の使用**です。

関連項目：

- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)
- [「データチェック」 \(ページ 63\)](#)
- [「レコードへのフィールドの追加」 \(ページ 70\)](#)

フィールドプロパティ - コードページ

フィールドに対して選択されたコードページを表示または変更します。コードページは、フィールド内の文字データに使用する文字セットを示します。

データマップのデフォルトのコードページを使用する

PowerExchange がこのフィールドにオペレーティングシステムのデフォルトのコードページを使用するよう設定するには、このオプションを選択します。

デフォルトでは、このオプションは選択されています。

選択したコードページを使用する

このフィールドのコードページを選択します。

注: マルチバイトコードページが使用できるのは、Datacom、DB2、VSAM KSDS、シーケンシャルデータセット、およびフラットファイルです。

関連項目：

- [「レコード内のフィールドのプロパティの表示または編集」 \(ページ 87\)](#)
- [「コードページ」 \(ページ 61\)](#)
- [「レコードへのフィールドの追加」 \(ページ 70\)](#)

フィールドプロパティ - 可変長

可変長フィールドのプロパティを表示または編集します。

カウントフィールド

このフィールドの長さの値を含むフィールドの名前を指定することができます。デフォルトでは選択されていません。

名前

フィールドの長さの値を含むフィールドの名前です。【**カウントフィールド**】オプションを選択した場合に有効になります。

区切り文字

フィールドを区切られたフィールドとして定義します。デフォルトは選択されています。

包含

フィールドデータの区切り文字の値を含みます。【**区切り文字**】オプションを選択した場合に有効になります。

排除

フィールドデータの最後から区切り文字の値を除外します。デフォルトは選択されています。【**区切り文字**】オプションを選択した場合に有効になります。

レコードの最後

フィールドがレコードの最後のフィールドである場合、可変長レコードのマッピング時にレコードの最後のマーカーを定義します。【**区切り文字**】オプションを選択し、フィールドがレコードの最後のフィールドである場合に有効になります。このオプションを選択した場合、【**値**】ボックスは無効になります。

値

区切られたフィールドの区切り文字の値です。有効な値は次のとおりです。

- テキスト文字列
- 次のように指定された 16 進値です。

x'nn'

デフォルト値は次のとおりです。

x'00'

【**区切り文字**】オプションを選択した場合に有効になります。

パディング

DB2 などのいくつかのデータソースのデータファイルは、可変長データを含む固定長フィールドを持つ場合があります。この場合、フィールドはパディングされます。このタイプのデータをインポートした場合、このオプションを選択し、PowerExchange により正しくパディングフィールドが処理されることを確認します。例えば、i5/OS システムから順次アクセスされた DB2 データをインポートする場合は、このオプションを選択します。フィールドがパディングされていない可変長フィールドである場合は、このオプションの選択を取り消します。デフォルトでは選択されていません。

テーブルプロパティ - 定義

定義プロパティを表示または編集します。

テーブル名

抽出時に SQL で使用されるテーブル名です。

最大長は 128 文字です。

カラムの生成

テーブルが基にしているレコードのフィールド定義の変更後、PowerExchange でテーブルの列を更新する方法を示します。最初にテーブルを定義する場合、PowerExchange により、そのテーブルが基にしているレコードのフィールド名から列名が引き出されます。ただし、PowerExchange では、配列として定義されたフィールドを含むレコードには特別な命名規則が使用されます。

次のいずれかのオプションを選択します。

- **配列フォーマットの変更の適用。** テーブルが基にしているレコードに、配列として定義される変更フィールドが含まれている場合、これらの変更はテーブルの対応する列に反映されます。
- **不足しているカラムを使用してリフレッシュ。** テーブルが基にしているレコードに新しいフィールドが含まれている場合、対応する列がテーブルに追加されます。
- **デフォルト値にリセット。** PowerExchange により、列名はテーブルが基にしているレコードの対応するフィールド名にリセットされます。テーブル内の変更された列名はすべて破棄されます。
- **非表示カラムの削除。** DB2UNLD では、新しい一連の列が、各フィールドの **【テーブルから非表示】** のデフォルト値で生成されます。

CCK カラムを生成する

IMS では、テーブルが基にしているレコードに連結キー（CCK）フィールドが含まれている場合に、テーブルに CCK フィールドのカラムを表示するかどうかを制御します。

- テーブルに CCK カラムを表示するには、このオプションを選択します。
CCK カラムが含まれるテーブルでデータベース行のテストを実行するには、次のライブラリを APF 許可します。
 - Netport ジョブに BMP と共に IMS DL/1 バッチデータマップを使用する場合は、BMP STEPLIB 連結内のすべてのライブラリを APF 許可する必要があります。このように処理しないと、PowerExchange によって戻りコード 12 でメッセージ PWX-02204 が発行されます。
 - IMS ODBA データマップを使用する場合は、PowerExchange Listener の STEPLIB 連結内のすべてのライブラリを APF 許可する必要があります。このように処理しないと、PowerExchange が異常終了します。
- テーブルで CCK カラムを非表示にするには、このオプションの選択を取り消します。テーブルが基にしているレコードに CCK フィールドが含まれている場合は、CCK カラムを非表示にすることの確定が求められます。

使用可能なレコード

複雑なテーブルにはないデータマップのレコードを一覧表示します。

【**レコードの依存関係**】 リストにレコードを追加するには、【**使用可能なレコード**】 リストで、レコードを右クリックし、【**レコードの追加**】 をクリックします。

親レコードに子レコードを追加する手順

- 【**レコードの依存関係**】 リストで、親レコードを選択します。
- 【**使用可能なレコード**】 リストで、レコードを右クリックし、【**レコードを子として追加**】 をクリックします。レコードが親レコードの子レコードとして【**レコードの依存関係**】 リストに移動します。

IDMS では、レコード取得に使用するシステムインデックスを選択できます。レコードを右クリックし、【**システムインデックスの使用**】 をクリックします。次に、インデックスを選択します。

Datcom で、【**レコードの依存関係**】 リストにレコードを追加するには、【**使用可能なレコード**】 リストでレコードを右クリックし、次のいずれかのコマンドを選択します。

- 【**テーブル読み込みシーケンスを使用したレコードの追加**】 > 【**物理的シーケンシャル**】。データマップにレコードを追加します。データを取得するために、アクセス方式で物理的シーケンシャルによってレコードを取得する Datcom GETPS ルーチンが呼び出されます。ただし、キーに WHERE 句が含まれるクエリの場合、アクセス方式ではインデックスシーケンシャルアクセスが使用されます。
- 【**テーブル読み込みシーケンスを使用したレコードの追加**】 > 【**インデックスシーケンシャル**】。データマップにレコードを追加します。データを取得するために、アクセス方式でインデックスによってレコードを取得する Datcom GETPS ルーチンが呼び出されます。

デフォルトは【**物理的シーケンシャル**】 です。

レコードの依存関係

定義された階層依存関係を持つ、複雑なテーブルのレコードを一覧表示します。

レコード依存関係を削除するには、レコードを右クリックし、【**削除**】 をクリックします。

IDMS の場合、エリア読み取りの方向を反転するには、レコードを右クリックし、【**エリア読み取りの反転**】 をクリックします。セット読み取りの方向を反転するには、レコードを右クリックし、【**セット読み取りの反転**】 をクリックします。

Datcom で、テーブル読み込みシーケンスを変更するには、【**使用可能なレコード**】 リストでレコードを右クリックし、次のいずれかのコマンドを選択します。

- 【**テーブル読み込みシーケンス**】 > 【**物理的シーケンシャル**】
- 【**テーブル読み込みシーケンス**】 > 【**インデックスシーケンシャル**】

選択したレコードの複数インスタンスをどのように処理しますか？

行がデータベース行出力で表示される方法やターゲットに書き込まれる方法を制御します。

次のうち 1 つのオプションを選択します。

- **新しい行**。新しい行が表示されるか、レコードまたはセグメントのインスタンスごとにターゲットに書き込まれます。
- **無視**。レコードまたはセグメントの 2 番目とそれに次ぐインスタンスが表示されるか、ターゲットに書き込まれます。
- **配列**。【**配列**】 リストに指定されたレコードまたはセグメントの数が表示されるか、単一出力行のターゲットに書き込まれます。

PowerExchange により、出力行がいっぱいになるまで入力が行われ、次のうち 1 つのアクションが完了されます。

- **【オーバーフロー時の新しい行】** オプションを選択しない場合、PowerExchange では、それに次ぐレコードまたはセグメントが無視されます。
- **【オーバーフロー時の新しい行】** オプションを選択する場合、PowerExchange では、オーバーフローレコードまたはセグメントのある新しい行が表示されます。

例えば、5 つのインスタンスを持つレコードで、**【配列】** リストに「3」と入力した場合は、PowerExchange により、出力行が 2 つ構築されます。最初の行には、3 つのインスタンスの配列が含まれ、2 番目の行には 2 つのインスタンスの配列が含まれます。

注: 親レコードまたはセグメントを **【配列】** に設定した場合は、子レコードまたはセグメントのすべてを **【無視】** に設定する必要があります。

フィールド

配列またはフルプフィールドの要素がデータベース行出力で表示される方法やターゲットに書き込まれる方法を制御します。

次のいずれかのオプションを選択します。

- 配列に各要素を表示したり、単一の行にグループフィールドを表示するには、フィールドを選択します。
- 配列に各要素を表示したり、別の行にグループフィールドを表示するには、フィールドの選択を取り消します。

注: グループフィールドや配列として定義したフィールドを **【フィールド】** リストで表示するには、**【グループと配列のみ】** オプションを選択します。**【フィールド】** リストのレコードのすべてのフィールドを表示するには、このオプションの選択を取り消します。このオプションでは、**【フィールド】** リストおよび **【テーブルプロパティ】** ダイアログボックスに表示されるフィールドは制御されますが、**データベース行のテストによる出力ウィンドウ**に表示されるフィールドは制御されません。

単一入力行の複数配列

複数配列、または OCCURS 節を含む単一レコードから、複数の出力行を生成します。PowerExchange により、レコードのデータの終了時に、出力フィールドが NULL に設定されます。

複数の OCCURS 句を持つ、インポートされた COPYLIB のあるテーブルで有効になります。

関連項目：

- [「データマップテーブルの追加」 \(ページ 80\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)
- [「配列を含む単一レコードデータマップの追加 - 例」 \(ページ 263\)](#)
- [「IMS レコードへの CCK フィールドの追加」 \(ページ 78\)](#)

テーブルプロパティ - フィルタ

テーブルのフィルタ条件を表示または編集します。

カラム名のフィルタ

列名に基づくデータマップの列をフィルタリングします。次のオプションがあります。

- **インクルード**。指定する名前のパターンと一致する列を含みます。
- **除外**。名前のパターンと一致する列を除外します。

デフォルトは**除外**です。

カラム名

フィルタ条件として使用する名前を入力します。

フィルタ条件で、以下のワイルドカード文字を使用できます。

- アスタリスク (*) は、1 つ以上の一致する文字を表します。
- 疑問符 (?) は、単一の一致する文字を表します。

デフォルトでは、PowerExchange により、FILLER で始まる名前を持つ列はすべて除外されます。

関連項目：

- [「データマップテーブルの追加」 \(ページ 80\)](#)
- [「データマップ設定の定義」 \(ページ 44\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)

テーブルのプロパティ - IMS オプション

テーブルの IMS オプションを表示または編集します。

Update/Insert Options (更新/挿入オプション)

次のうち 1 つのオプションを選択します。

- **最低セグメントのみ**。最低セグメントのアクションを適用します。
- **すべての可能なレベル**。アクションをセグメントレベルに適用します。

デフォルトは**最低セグメントのみ**です。

Update all matching segments (すべての一致セグメントを更新)

一意でない一致するすべてのセグメントを更新します。

注: このオプションは複雑なテーブルでは無効になります。

デフォルトでは選択されていません。

Delete Options (削除オプション)

次のうち 1 つのオプションを選択します。

- **最低セグメントのみ**。最低セグメントの削除アクションを適用します。

- **階層内のすべての子なしセグメント。** 子を持たないすべてのセグメントレベルに削除アクションを適用します。

警告: IMS 階層について十分な知識がない場合は、[削除オプション] オプションのデフォルト値を編集しないでください。この設定を変更した場合、IMS データが失われることがあります。

デフォルトは**最低セグメントのみ**です。

Delete all matching segments (**すべての一致セグメントを削除**)

一意でない一致するすべてのセグメントを削除します。

デフォルトでは選択されていません。

Ignore record not found error on delete (**削除でエラーが見つからないレコードを無視**)

削除アクションで生成される [レコードが見つかりません] エラーをすべて無視します。

デフォルトでは選択されていません。

関連項目：

- [「データマップテーブルの追加」 \(ページ 80\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)
- [「テーブルの IMS オプションの表示または編集」 \(ページ 130\)](#)

テーブルプロパティ - キー

Adabas または Datacom テーブルのキーのプロパティを表示または編集します。

プライマリキー

テーブルのプライマリキーを選択します。

キー詳細

選択したプライマリキーのシーケンス番号と列を表示します。

関連項目：

- [「データマップテーブルの追加」 \(ページ 80\)](#)
- [「データマップテーブル用プロパティの表示または編集」 \(ページ 91\)](#)
- [「テーブル内の列のプロパティの表示または編集」 \(ページ 92\)](#)

検索ダイアログボックス

検索条件を入力し、データマップレコードのフィールド、パーソナルメタデータプロファイルのテーブル、あるいはデータマップテーブル、パーソナルメタデータプロファイルテーブル、抽出マップ、またはキャプチャ登録の列を検索します。

検索にはワイルドカードを含めることができます。次のデフォルトのワイルドカードがあります。

- アスタリスク (*) は、複数の一致する文字を表します。
- クエスチョンマーク (?) は 1 個の一致する文字を表します。

ヒント: テーブル、フィールドまたは列名にワイルドカード文字が含まれている場合、エスケープ文字を持つ名前ワイルドカード文字が先行されます。デフォルトでは、エスケープ文字は波形符号 (~) です。

検索条件ボックスの右にある矢印をクリックし、ワイルドカードまたはエスケープ文字を検索条件に挿入します。

検索対象

データマップレコードのフィールド、あるいはデータマップテーブル、パーソナルメタデータプロファイルテーブル、抽出マップ、またはキャプチャ登録の列を検索するには、検索するフィールドまたは列の名前を入力します。

スキーマ

パーソナルメタデータプロファイルのテーブルを検索するには、スキーマ名を入力します。

マップ名

パーソナルメタデータプロファイルのテーブルを検索するには、マップ名を入力します。

テーブル

パーソナルメタデータプロファイルのテーブルを検索するには、テーブル名を入力します。

詳細

【詳細】 をクリックして、検索用のワイルドカードまたはエスケープ文字を変更します。

関連項目：

- [「テーブルとカラムの表示」 \(ページ 83\)](#)
- [「パーソナルメタデータプロファイル内のテーブルの検索」 \(ページ 32\)](#)
- [「パーソナルメタデータプロファイル内のテーブル内のカラムの検索」 \(ページ 33\)](#)

付録 F

DTL__CAPXTIMESTAMP のタイムスタンプ

- [データソースによって DTL__CAPXTIMESTAMP フィールドに報告されるタイムスタンプ, 397 ページ](#)

データソースによって DTL__CAPXTIMESTAMP フィールドに報告されるタイムスタンプ

変更レコードに生成された DTL__CAPXTIMESTAMP フィールドで PowerExchange が報告するタイムスタンプは、データソースタイプと特定のパラメータ設定によって異なります。

z/OS 上の PowerExchange データソースの場合、UOWC CAPI_CONNECTION の TIMESTAMP パラメータにより、PowerExchange が DTL__CAPXTIMESTAMP フィールドに報告するタイムスタンプのタイプを制御します。TIMESTAMP パラメータを COMMIT に設定すると、PowerExchange は、トランザクションのすべての変更に対して、ソースでのトランザクションコミットのタイムスタンプを報告します。TIMESTAMP パラメータを COMMIT に設定した場合、PowerExchange は、トランザクション内のすべての変更について、ソースのトランザクションコミットのタイムスタンプを報告します。LOG のデフォルトのパラメータ値を使用した場合、PowerExchange はソースデータのベースログからタイムスタンプを取得します。この場合、タイムスタンプの種類はソースの種類に応じて異なります。

以下の表に、TIMESTAMP パラメータにデフォルト値の LOG を使用している場合に PowerExchange によって報告されるタイムスタンプを示します。

データソースタイプ	タイムスタンプのタイプ
Adabas	PLOG ブロックヘッダからの HDDATE タイムスタンプで、ブロックが書き込まれた時刻を示します。 注: 低レベルの更新アクティビティを持つ Adabas 環境では、異なるタイミングで発生した複数の更新に対して同一のタイムスタンプが報告される場合があります。
Datcom テーブルベースの CDC	変更レコードが Datcom の LXX ログに書き込まれたときの協定世界時 (UTC) またはローカル時刻を示します。ECCR コンフィギュレーションメンバ (ECCRD CMP) 内の LOCAL_TIME パラメータは、UTC またはローカル時刻のどちらを使用するかを制御します。
Db2 for i (i5/OS)	変更がジャーナルに記録された時刻を表す i5/OS ジャーナルタイムスタンプです。

データソースタイプ	タイムスタンプのタイプ
Db2 for z/OS	DB2 ECCR が変更データレコードをキャプチャした時刻を示します。UOW 内のそれぞれのレコードには、異なるタイムスタンプがあります。通常、このタイムスタンプは、DB2 for z/OS システムのタイムゾーンを反映する UTC 値です。
IDMS	変更データレコードが IDMS ログファイルに書き込まれた時刻を示します。このタイムスタンプはスタックロック (STCK) タイムスタンプと等価です。この時刻にはローカルタイムゾーンが反映されません。
IMS ログベースの CDC	変更が IMS ログに記録された時刻を示します。
IMS 同期 CDC	変更が発生した時刻を示します。
バッチ VSAM および CICS/VSAM	変更レコードがキャプチャされた時刻を示します。UOW 内のそれぞれのレコードには、異なるタイムスタンプがあります。通常、このタイムスタンプは UTC 値です。

UOWC の CAPI_CONNECTION 文を使用しない他のデータソースに対しては、DTL__CAPXTIMESTAMP フィールドに報告するための適切なタイムスタンプを PowerExchange が決定します。PowerExchange Express CDC for Oracle のソースに対しては、Express CDC コンフィギュレーションファイルの OPTIONS 文で設定される TIME_STAMP_MODE パラメータでタイムスタンプのタイプを制御します。

以下の表に、このようなデータソースに対して PowerExchange が報告するタイムスタンプのタイプを示します。

データソースタイプ	タイムスタンプのタイプ
Linux、UNIX、または Windows 上の Db2	トランザクションコミットのタイムスタンプを示します。このタイムスタンプは Db2 システムの昇順の仮想タイムスタンプ (VTS) で、通常は UTC 値に対応しています。
Microsoft SQL Server	変更が配布データベースに書き込まれた時刻を示します。
MySQL	MySQL がバイナリログに記録した変更イベントのタイムスタンプを示します。
PowerExchange Express CDC for Oracle	タイムスタンプのタイプは Express CDC コンフィギュレーションファイルの OPTIONS 文で設定される TIME_STAMP_MODE パラメータで制御します。 <ul style="list-style-type: none"> - デフォルト値の LOGTIME を使用している場合、PowerExchange は REDO ログに記録されたソースデータベースの変更のタイムスタンプを報告します。このタイムスタンプにはローカルタイムゾーンが反映されます。 - COMMITTIME を指定した場合、PowerExchange はソースデータベースのトランザクションコミットのタイムスタンプを報告します。 - BEGINTIME を指定した場合、PowerExchange は開始 UOW ログレコードのタイムスタンプを報告します。
PostgreSQL	トランザクションコミットの時間を示します。

COBOL コピーブックのインポート 時におけるメモリマップ作成のト レース

PowerExchange Navigator から COBOL コピーブックをデータマップにインポートする際、PowerExchange Navigator は、拡張データメモリマップを示すトレース出力を生成できます。

IBM IGYWC コンパイルプロシージャが生成するデータ分割マップと比較して、トレースメモリマップは、追加情報を提供し、FILLER グループと REDEFINES グループが特定しやすい形式を使用します。次のタスクを実行する際にトレースを使用します。

- z/OS システムではなく PowerExchange Navigator マシンからマッピングエラーのトラブルシューティングを行う
- メモリマップがデータマップと一致していることを確認する
- ストレージ内のコピーブックアイテムのオフセットを確認する
- FILLER アイテムと REDEFINES アイテムをチェックして、これらが正しくマッピングされていることを確認する
- 再定義のために PowerExchange が追加する FILLER_REDEF_ *nn* フィールドがパディング値と一致していることを確認する
- マッピングの必要がないフィールドを特定する
- COBOL レベル、FILLER アイテムと REDEFINES アイテムの場所、パディング、OCCURS 値を含む、IBM IGYWC コンパイラから使用できないマップ情報を表示する

トレースを設定して実行するには、次の手順に従います。

1. PowerExchange Navigator が実行する Windows システム上の dbmover.cfg 構成ファイルで、次の TRACE 文を指定します。
TRACE=(COBDTL,0,99)
2. PowerExchange Navigator で、COBOL コピーブックをデータマップにインポートします。
PowerExchange Navigator が詳細ログにトレース出力を生成します。または、dbmover.cfg ファイルに TRACING 文を指定した場合は、代替ログファイルに生成されます。
3. 生成されたトレース出力をコピーブックと比較し、データストレージが正しくマッピングされていることを確認します。

COBOL メモリマップのトレース出力例

DBMOVER 構成ファイルで TRACE=(COBDTL,0,99)文を指定すると、PowerExchange Navigator でデータマップに COBOL コピーブックをインポートする際、PowerExchange Navigator が、ソースデータレコードのメモリマップを示すトレース出力を生成できます。

たとえば、次の COBOL コピーブックをインポートするとします。

```
01 TESTREC1.
02 REC1-1.
05 ITEM-1.
   10 ITEM-1-1      PIC X(4).
   10 ITEM-1-2      PIC X(4).
05 ITEM-2.
   10 ITEM-2-1      PIC X(4).
05 ITEM-3.
   10 ITEM-3-1      OCCURS 4 TIMES.
       15 ITEM-3-1-1 PIC XX.
05 FILLER REDEFINES ITEM-3.
       15 ITEM-4-1-1 PIC XX.
       15 ITEM-4-1-2 PIC XX.
       15 ITEM-4-1-3 PIC XX.
       15 ITEM-4-1-4 PIC XX.
...
```

この COBOL コピーブックに基づいて、PowerExchange Navigator は次のトレース出力を生成します。

COBOL Memory Map for TESTREC1

1	01	TESTREC1		OCL121	pad 0	type G occurs 0	00000000 (0)	last NEW_2	
...	2	02	REC1_1		OCL42	pad 0	type G occurs 0	00000000 (0)	last ITEM_8_1_2
.....	3	05	ITEM_1		OCL8	pad 0	type G occurs 0	00000000 (0)	last ITEM_1_2
.....	4	10	ITEM_1_1		CL4	pad 0	type D occurs 0	00000000 (0)	
.....	4	10	ITEM_1_2		CL4	pad 0	type D occurs 0	00000004 (4)	
.....	3	05	ITEM_2		OCL4	pad 0	type G occurs 0	00000008 (8)	last ITEM_2_1
.....	4	10	ITEM_2_1		CL4	pad 0	type D occurs 0	00000008 (8)	
*****	3	05	ITEM_3		OCL8	pad 22	type R occurs 0	0000000C (12)	last ITEM_3_1_1
.....	4	10	ITEM_3_1		OCL8	pad 0	type D occurs 4	0000000C (12)	last ITEM_3_1_1
.....	5	15	ITEM_3_1_1		CL2	pad 0	type D occurs 0	0000000C (12)	
=====	3	05	FILLER		OCL8	pad 22	type R occurs 0	0000000C (12)	last ITEM_4_1_4
.....	4	15	ITEM_4_1_1		CL2	pad 0	type D occurs 0	0000000C (12)	
.....	4	15	ITEM_4_1_2		CL2	pad 0	type D occurs 0	0000000E (14)	
.....	4	15	ITEM_4_1_3		CL2	pad 0	type D occurs 0	00000010 (16)	
.....	4	15	ITEM_4_1_4		CL2	pad 0	type D occurs 0	00000012 (18)	
...										

このトレース出力には以下の情報が含まれます。

- 階層ポジションと COBOL レベル。たとえば、次の行では、「1」が階層ポジションで、「01」が COBOL レベルです。
1 01 TESTREC1
- レコード名。例: TESTREC1。
- アイテムまたはグループの長さ。例: OCL8 はグループ、CL4 はアイテム。
- パディング。pad 値は、REDEFINE に、長さの違う複数の再定義が含まれている場合に PowerExchange がデータマップに追加する FILLER_REDEF_nn フィールドを反映します。パディングとオフセットは一致している必要があります。
- アイテムタイプ: グループは G、表示は D、REDEFINES グループは R。
注: 最初の定義グループには、連続するアスタリスク(*)が前に付きます。以降の再定義グループには、等号(=)が前に付きます。
- OCCURS 値。OCCURS 句が定義されていない場合は 0。
- 16 進数と 10 進数のオフセット。10 進数のオフセットは括弧で括ります。
- グループの最後の依存アイテム。

トレース出力を次の COBOL コンパイル出力と比較してみてください。

Source LineID	Hierarchy and Data Name	Base Locator	Displacement Structure	Asmblr Data Definition	Data Type	Data Def Attributes
2	PROGRAM-ID UAMC-----*					
17	1 TESTREC1.		000000000	DS 0CL121	Group	
18	2 REC1-1.		000000000	DS 0CL42	Group	
19	3 ITEM-1.		000000000	DS 0CL8	Group	
20	4 ITEM-1-1.		000000000	DS 4C	Display	
21	4 ITEM-1-2.		000000004	DS 4C	Display	
22	3 ITEM-2.		000000008	DS 0CL4	Group	
23	4 ITEM-2-1.		000000008	DS 4C	Display	
24	3 ITEM-3.		00000000C	DS 0CL8	Group	
25	4 ITEM-3-1.		00000000C	DS 0CL2	Group	0
26	5 ITEM-3-1-1.		00000000C	DS 2C	Display	
27	3 FILLER.		00000000C	DS 0CL8	Group	R
28	4 ITEM-4-1-1.		00000000C	DS 2C	Display	
29	4 ITEM-4-1-2.		00000000E	DS 2C	Display	
30	4 ITEM-4-1-3.		000000010	DS 2C	Display	
31	4 ITEM-4-1-4.		000000012	DS 2C	Display	
...						

トレースには、COBOL レベル、パディング、OCCURS 値、10 進数のオフセット、グループ内の最後のアイテムの名前といった追加情報が含まれています。また、再定義アイテムも明確に特定されています。

索引

数字

- 3 層名
データマップ [52](#)
- 2 層名
データマップ [52](#)

A

- ADABAS アクセス方式
データマッププロパティ [352](#)
- Adabas キー
データマップテーブルへの追加 [80](#), [112](#)
データマップへのインポート [55](#)
- Adabas データベース
データマップ [352](#)
- Adabas データマップ
概要 [110](#)
キーのインポート [55](#)
追加 [111](#)
テーブルへのキーの追加 [112](#)
フィールド定義テーブル (FDT) のインポート [55](#), [148](#)
- Adabas ファイル
データマップ [53](#)
- Adabas フィールド定義テーブル (FDT)
データマップ内のひょうじ [149](#)
データマップ内のリフレッシュ [149](#)
データマップへのインポート [55](#), [148](#)
- ADACMP コピーブック
サポートされたデータマップタイプ [143](#)

C

- CallProg 関数
データマップ内のユーザー定義フィールドでの呼び出し [282](#)
ユーザー定義フィールドの式 [297](#)
- CAMAPS ファイル [195](#)
- CAPTURE_NODE_EPWD パラメータ
暗号化されたパスワードの生成 [48](#)
- CAPXRT 詳細パラメータ
データベース行のテスト用の Adabas パラメータ [253](#)
データベース行のテスト用の AS400 パラメータ [253](#)
データベース行のテスト用の DB2 for Linux, UNIX, and Windows
パラメータ [253](#)
データベース行のテスト用の Microsoft SQL Server パラメータ [254](#)
データベース行のテスト用の Oracle パラメータ [254](#)
データベース行のテスト用の全般パラメータ [249](#)
- CAPX 詳細パラメータ
i5/OS ソースのデータベース行のテスト [248](#)
データベース行のテスト用の全般パラメータ [246](#)
- CheckNumData 関数 [306](#)
- Check 関数
例 [306](#)

- CL ユーザー定義プログラム
リンケージ [336](#)
例 [336](#)
- COBOL コピーブック
サポートされたデータマップタイプ [143](#)
データマップへのインポート [268](#), [272](#), [273](#), [279](#), [283](#)
- COBOL コピーブックの PIC G フィールド
コードページ [153](#)
- COBOL ユーザー定義プログラム
オペレーティングシステムサポート [334](#)
- Concat 関数
例 [307](#)
- CopyData 関数
ユーザー定義フィールドの式 [297](#)

D

- DATACOM アクセス方式
データマッププロパティ [354](#)
- Datcom キャプチャ登録
エレメントの表示 [191](#)
- DATACOM コピーブック
サポートされたデータマップタイプ [143](#)
- Datcom データマップ
概要 [113](#)
追加 [113](#)
レコードからのエレメントの削除 [106](#)
レコードからのキーセグメントの削除 [107](#)
レコードからのキーの削除 [106](#)
レコード内のエレメント [64](#)
レコード内のキー [64](#)
レコード内のキーセグメント [64](#)
レコードへのエレメントの追加 [71](#)
レコードへのキーセグメントの追加 [71](#), [74](#)
レコードへのキーの追加 [71](#), [72](#)
- Datcom データマップレコード
エレメントの追加 [71](#)
プロパティ [376](#)
- Datcom ファイル
データマップ [53](#), [354](#)
- DB2_BIN_CODEPAGE 文
DBMOVE コンフィギュレーションファイル [230](#)
- DB2390 IMG 詳細パラメータ
データベース行のテスト [255](#)
- DB2CODEPAGE 文
DBMOVE コンフィギュレーションファイル [230](#)
- DB2UNLD アクセス方式
データマッププロパティ [355](#)
- DB2 アクセス方式
データマッププロパティ [355](#)
- DB2 アンロードファイル
データマップ [53](#)
- DB2 アンロードファイルデータマップ
概要 [117](#)
追加 [117](#), [119](#)

DB2 カタログコピーブック
サポートされたデータマップタイプ [143](#)
ローカル詳細 [154](#)
DB2 カタログデータマップ
追加 [115](#)
概要 [114](#)
DB2 データベース
データマップ [53](#), [355](#)
DB2 データマップ
テーブルのフィルタ [358](#)
レコードレイアウトの編集 [116](#)
DBD ソースのインポートで作成された IMS 階層
データマップ内の表示 (IMS) [128](#)
DBD ソースのインポートで作成された階層
データマップ内の表示 (IMS) [128](#)
dbmover.cfg ファイル
IMS データにアクセスするための Windows 上での構成 [126](#)
DBMOVER 構成ファイル
暗号化されたパスワードの生成 [48](#)
DBMOVER コンフィギュレーション文
DB2_BIN_CODEPAGE [230](#)
DB2CODEPAGE [230](#)
DL/1 バッチアクセス方式
データマッププロパティ [358](#)
DL/I または BMP を使用して IMS にアクセス
ODBA を使用して IMS データにアクセスするためのタスクの構成
[124](#)
DTL__BI_columnname カラム
説明 [214](#)
DTL__CAPXACTION
説明 [214](#)
DTL__CAPXCASDELIND
説明 [214](#)
DTL__CAPXRESTART1 カラム
説明 [214](#)
DTL__CAPXRESTART2 カラム
説明 [214](#)
DTL__CAPXROWID カラム
説明 [214](#)
DTL__CAPXRRN カラム
説明 [214](#)
DTL__CAPXTIMESTAMP カラム
説明 [214](#)
DTL__CAPXTIMESTAMP フィールド
データソースによって報告されるタイムスタンプのタイプ [397](#)
DTL__CAPXUOW カラム
説明 [214](#)
DTL__CAPXUSER カラム
説明 [214](#)
DTL__CI_columnname カラム
説明 [214](#)
DTL__ST 列
説明 [214](#)
DTLCAMAP ファイル [195](#)
DTLDSN SQL エスケープシーケンス [100](#)
DTLFILELIST SQL エスケープシーケンス [100](#)
DTLREXE ユーティリティ -epwd オプション
暗号化されたパスワードの生成 [48](#)
DTLUAPPL ユーティリティ EPWD オプション
暗号化されたパスワードの生成 [48](#)
DTLUCBRG ユーティリティ EPWD オプション
暗号化されたパスワードの生成 [48](#)
DTLUCUDB ユーティリティ EPWD オプション
暗号化されたパスワードの生成 [48](#)
DTLURDMO ユーティリティ EPWD オプション
暗号化されたパスワードの生成 [48](#)
DTLUTSK ユーティリティ EPWD オプション
暗号化されたパスワードの生成 [48](#)

E

ESDS アクセス方式
データマッププロパティ [360](#)
[Expr(n)] タブ
レコードウィンドウ [81](#)

F

Fragment 関数
例 [309](#)

G

GenVRowKey 関数
例 [310](#)
GetCurrentFileName 関数
例 [311](#)
GetDatabaseKey 関数
RBA の取得 [141](#)
RRN の取得 [141](#)
エイリアス [314](#)
GetDataFlowType 関数 [313](#)
GetDbKeyOfFirstMember 関数 [314](#)
GetDbKeyOfLastMember 関数 [315](#)
GetDbKeyOfNextMember 関数 [315](#)
GetDbKeyOfOwner 関数 [315](#)
GetDbKeyOfPriorMember 関数 [316](#)
GetDbKey 関数 [314](#)
GetFullDbKeyOfFirstMember 関数 [317](#)
GetFullDbKeyOfLastMember 関数 [317](#)
GetFullDbKeyOfNextMember 関数 [318](#)
GetFullDbKeyOfOwner 関数 [318](#)
GetFullDbKeyOfPriorMember 関数 [319](#)
GetFullDbKey 関数 [316](#)
GetIMSRBABByLevel 関数 [319](#)
GetPageGroupOfOwner 関数 [320](#)
GetPageGroup 関数 [320](#)
GetPgGrpAndRdxOfOwner 関数 [321](#)
GetPgGrpAndRdx 関数 [321](#)
GetPgGrpOfOwner 関数 [321](#)
GetRadixOfOwner 関数 [322](#)
GetRadix 関数 [322](#)
GetSeqWithinLevel 関数
例 [323](#)
GetSeqWithinParent 関数
例 [323](#)

I

IBM MQSeries メッセージキュー
データマップ [53](#)
IDMS アクセス方式
データマッププロパティ [361](#)
IDMS キャプチャ登録
ベースレコードの親階層の表示 [192](#)
IDMS コピーブック
サポートされたデータマップタイプ [143](#)
詳細プロパティ [160](#)
IDMS データマップ
CALC エlement [65](#)
オーナーレコードとセット [65](#)
概要 [119](#)
設定の定義 [48](#)
追加 [120](#)

IDMS データマップ (続く)

- レコードからの CALC エLEMENT の削除 [107](#)
- レコードからのオーナーレコードとセットの削除 [107](#)
- レコードへの CALC エLEMENT の追加 [75](#)
- レコードへのオーナーレコードとセットの追加 [75](#)
- レコード用プロパティの表示 [377](#)
- レコード用プロパティの編集 [377](#)

IDMS データベース

- データマップ [53](#), [361](#)

IDMS データマップレコード

- プロパティの編集 [121](#)

IMS ODBA アクセス方式

- データマッププロパティ [358](#)

IMS ODBA データマップレコード

- CCK フィールド [65](#)
- 検索フィールド [65](#)

IMS DL/1 バッチデータマップレコード

- CCK フィールド [65](#)
- 検索フィールド [65](#)

IMS RBA

- データマップレコードでの取得 (IMS) [128](#)

IMSUNLD および NRDB の詳細パラメータ

- データベース行のテスト [255](#)

IMS アクセス方式

- DL/I バッチまたは BMP を使用した IMS データへのアクセスの構成タスク [124](#)
- ODBA を使用して IMS データにアクセスするためのタスクの構成 [125](#)

IMS オプション

- データマップテーブル用の編集 [130](#)

IMS データマップ

- 概要 [123](#)
- コピーブックのインポート [133](#)
- 追加 [127](#), [132](#), [134](#)
- 追加するための前提条件 [124](#)
- テスト [131](#)
- プロパティの編集 [129](#)
- リモートシステムへの送信 [129](#)
- 例 [131](#)
- レコードの表示 [127](#)

IMS データベース

- データマップ [53](#), [358](#)
- データマップ内の階層メタデータの保持 [146](#)
- データの書き込み [130](#)
- ルックアップトランスフォーメーションを使用したデータの検索 [129](#)

IMS のアクセス方式

- DL/I バッチまたは BMP を使用した IMS データへのアクセスの構成タスク [124](#)
- ODBA を使用して IMS データにアクセスするためのタスクの構成 [125](#)

K

KSDS アクセス方式

- データマッププロパティ [362](#)

L

LengthOf 関数 [324](#)

LISTTASK コマンド

- データベース行のテスト内の発行 [234](#)

LTrim 関数

- 例 [324](#)

M

MQSeries アクセス方式

- データマッププロパティ [362](#)

MQSeries メッセージキュー

- データマップ [53](#), [362](#)

O

OCCURS DEPENDING ON 句

- 複数出力行の表示 [272](#)

OCCURS 句

- 配列の定義 [60](#)

ODBA を使用した IMS へのアクセス

- ODBA を使用して IMS データにアクセスするためのタスクの構成 [125](#)

ODBC

- ファイルリストファイル [100](#)

P

PL/I コピーブック

- サポートされたデータマップタイプ [143](#)
- データマップへのインポート [276](#)

PL/I ユーザー定義プログラム

- オペレーティングシステムサポート [337](#)

PowerExchange

- サポートされたコードページ [62](#)
- リソース [30](#)

PowerExchange Navigator

- アプリケーションウィンドウ [19](#)
- インタフェース [17](#)
- ウィンドウの整列 [21](#)
- 概要 [15](#)
- [キャプチャ登録] ウィンドウ [19](#)
- ショートカットキー [29](#)
- ショートカットメニュー [25](#)
- ステータスバー [30](#)
- タイトルバー [21](#)
- 抽出定義ウィンドウ [19](#)
- ツールバー [28](#)
- テーブルウィンドウ [19](#)
- メニューバー [21](#)
- リソースインスペクタ [19](#)
- リソースウィンドウ [19](#)
- リソースエクスプローラ [19](#)
- 例 [258](#)
- レコードウィンドウ [19](#)

PowerExchange が生成した抽出マップのカラム

- DTL__BI_columnname [214](#)
- DTL__CAPXACTION [214](#)
- DTL__CAPXCASDELIND [214](#)
- DTL__CAPXRESTART1 [214](#)
- DTL__CAPXRESTART2 [214](#)
- DTL__CAPXROWID [214](#)
- DTL__CAPXRRN [214](#)
- DTL__CAPXTIMESTAMP [214](#)
- DTL__CAPXUOW [214](#)
- DTL__CAPXUSER [214](#)
- DTL__CI_columnname [214](#)
- DTL__ST [214](#)

PowerExchange リスナのコマンド

- データベース行のテスト内の発行 [234](#)

PowerExchange リソース

- 検索 [31](#)
- 削除 [33](#)

PowerExchange リソース (続く)

閉じる [33](#)

開く [31](#)

PowerExchange 関数

ユーザー定義フィールド [303](#)

ユーザー定義フィールド内の式 [304](#)

PREDICT コピーブック

サポートされたデータマップタイプ [143](#)

pwxcmd -epassword オプション

暗号化されたパスワードの生成 [48](#)

R

REDEFINES 文

COBOL コピーブック [272](#), [273](#)

RPG プログラム

オペレーティングシステムサポート [336](#)

RRDS アクセス方式

データマッププロパティ [364](#)

RTrim 関数

例 [325](#)

S

SEQ アクセス方式

データマッププロパティ [365](#)

SEQ データマップ

追加 [135](#)

SetBitA 関数 [325](#)

SetBitN 関数 [326](#)

Split 関数

例 [327](#)

SQL エスケープシーケンス

DTLDSN [100](#)

DTLFILELIST [100](#)

STOPTASK コマンド

データベース行のテスト内の発行 [234](#)

Strip 関数

例 [328](#)

T

TAPE アクセス方式

データマッププロパティ [367](#)

ToLower 関数 [328](#)

ToUpper 関数 [329](#)

U

UDB_CAPI_CONNECTION 文の EPWD オプション

暗号化されたパスワードの生成 [48](#)

USER アクセス方式

データマッププロパティ [368](#)

データマップ [339](#)

V

VSAM エントリシーケンスのデータセット (ESDS)

データマップ [53](#), [360](#)

VSAM キーシーケンスのデータセット (KSDS)

データマップ [53](#), [362](#)

VSAM 相対レコードデータセット (RRDS)

データマップ [53](#), [364](#)

VSAM データセットレコード

RBA の取得 [140](#), [141](#)

RRN の取得 [140](#), [141](#)

相対バイトアドレス (RBA) の取得 [140](#)

相対レコード番号 (RRN) の取得 [140](#)

VSAM データセット

一括読み込みのパフォーマンスの向上 [138](#)

インデックスバッファ [138](#)

キー [139](#)

制御間隔アクセス [139](#)

データバッファ [138](#)

VSAM データマップ

概要 [134](#), [136](#)

追加 [136](#)

W

windows

アプリケーション [19](#)

キャプチャ登録 [19](#)

抽出定義 [19](#), [202](#), [213](#)

データベース行のテストによる出力 [231-233](#)

テーブル [19](#), [83](#)

リソース [17](#), [19](#), [21](#)

リソースインスペクタ [17](#)

リソースエクスプローラ [17](#)

レコード [19](#), [81](#)

あ

アクセス

複数のテーブルからの DB2 アンロードデータ [118](#)

アクセス方式のタイプ

データソース [53](#), [109](#)

データマップのための選択 [53](#)

アセンブリ言語のユーザー定義プログラム

オペレーティングシステムサポート [335](#)

リンケージ [335](#)

例 [335](#)

アプリケーション

概要 [345](#)

抽出開始ポイントのリセット [350](#)

アプリケーションウィンドウ

概要 [19](#)

アプリケーショングループ

概要 [345](#)

削除 [350](#)

プロパティの表示 [348](#)

プロパティの編集 [348](#)

追加 [346](#)

暗号化されたパスワード

生成 [48](#), [49](#)

い

一括読み込みのパフォーマンス

VSAM データセットの向上 [138](#)

印刷

データマップ [101](#)

データマップテーブル [103](#)

データマップテーブル内の列 [103](#)

データマップレコード [102](#)

データマップレコード内のフィールド [103](#)

インデックスバッファ

VSAM データセット [138](#)

インポート

- IMS データマップへのコピーブック [133](#)
 - PL/I コピーブック データマップ [276](#)
 - エクスポートされたデータマップ [97](#)
 - データマップへの REDEFINES 文を含む COBOL コピーブック [272](#), [273](#)
 - データマップへのコピーブック [144](#)
 - データマップセグメントへのコピーブック [146](#)
 - データマップへの FDT (Adabas) [55](#)
 - データマップへの PL/I コピーブック [276](#)
 - データマップへのキー (Adabas) [55](#)
 - データマップへのフィールド定義テーブル (FDT) (Adabas) [148](#)
 - データマップレコードへのコピーブック [146](#)
- ## インポートの詳細
- データマップレコード [380](#)

う

ウィンドウの整理

- PowerExchange Navigator [21](#)

え

エクスポートされたデータマップ

- インポート [97](#)
- 概要 [64](#)
- 削除 [104](#)

エラー

- ユーザー定義プログラム [329](#)

エレメント

- データマップレコード内のプロパティの表示 (Datacom) [88](#)

お

オーナーレコードとセット内のエレメント (DMS)

- プロパティ [379](#)

か

階層

- SEQ データマップの表示 [136](#)
- VSAM データマップの表示 [138](#)
- 複数レコードデータマップ用の定義 [266](#)

概要

- PowerExchange Navigator [15](#)
- アプリケーション [345](#)
- アプリケーショングループ [345](#)
- エクスポートされたデータマップ [64](#)
- キャプチャ登録 [164](#)
- コードページ [61](#)
- コピーブック [55](#), [142](#)
- ショートカットキー [21](#)
- 抽出グループ [195](#)
- 抽出マップ [195](#)
- データチェック [63](#)
- データマップ [51](#), [52](#)
- データマップ (DB2 カタログ) [114](#)
- データマップ (IMS) [123](#)
- データベース行のテスト [229](#)
- データマップ (Adabas) [110](#)
- データマップ (Datacom) [113](#)
- データマップ (DB2 アンロードファイル) [117](#)
- データマップ (IDMS) [119](#)
- 特定のデータソースのためのデータマップ [109](#)

概要 (続く)

- 配列 [60](#)
- パーソナルメタデータプロファイル [222](#)
- メニューコマンド [21](#)
- ユーザーアクセス方式プログラム [65](#), [339](#)
- ユーザー定義フィールド [56](#)
- レコードレベルの出口プログラム [56](#)
- 登録グループ [164](#)

格納

- データマップ (IMS) [129](#)

可変長プロパティ

- データマップレコード内のフィールド [390](#)

関数

- CallProg [282](#)

関連付け

- 抽出マップを持つキャプチャ登録 [205](#)
- データマップがあるデータファイル [54](#), [263](#), [266](#)
- データマップと複数のデータファイル [54](#), [99](#)
- データマップのあるファイルリストファイル [100](#)

き

キー

- VSAM データセット [139](#)
- データマップテーブルへの追加 (Adabas) [80](#), [112](#)
- データマップへのインポート (Adabas) [55](#)

基数

- データマップレコード内のオーナーレコードとセットに対する取得 (IDMS) [321](#), [322](#)
- データマップレコードのための取得 (IDMS) [321](#), [322](#)

キャプチャ登録

- 削除 [192](#)
- ステータスオプション [181](#)
- タグ名 [165](#)
- 抽出マップのための変更 [205](#)
- プロパティの表示 [186](#)
- プロパティの編集 [186](#)
- 列の検索 [32](#), [396](#)
- 圧縮オプション [182](#)
- 概要 [164](#)
- 追加 [175](#)
- 登録グループ内のすべての削除 [194](#)
- 登録のカラムの表示 [183](#)
- 列の削除 [192](#)
- 列の追加 [192](#)
- 列を変更するための編集 [192](#)

キャプチャ登録 (Datacom)

- エレメントの表示 [191](#)

キャプチャ登録 (IDMS)

- ベースレコードの親階層の表示 [192](#)

[キャプチャ登録] ウィンドウ

- 概要 [19](#)

キャプチャ登録内の列

- 検索 [32](#)
- 削除 [192](#)
- 追加 [192](#)

共有リソース構成

- 作成 [37](#)

切り取りおよび貼り付け

- データマップテーブル [92](#)
- データマップテーブル内の列 [92](#)
- データマップレコード [92](#)
- データマップレコード内のフィールド [92](#)

け

現在の構成

選択 [37](#)

現在の削除により影響をうける参照リスト [105](#)

検索

PowerExchange リソース [31](#)

キャプチャ登録内の列 [32](#)

抽出マップ内の列 [32](#)

データマップテーブル内の列 [32](#)

データマップレコード内のフィールド [31](#)

パーソナルメタデータテーブル内の列 [33](#)

パーソナルメタデータプロファイル内のテーブル [32](#)

検索条件

キャプチャ登録内の列の検索 [396](#)

抽出マップ内の列の検索 [396](#)

データマップテーブル内の列の検索 [396](#)

データマップレコード内のフィールドの検索 [396](#)

パーソナルメタデータプロファイルテーブル内の列の検索 [396](#)

パーソナルメタデータプロファイル内のテーブルの検索 [396](#)

ワイルドカード [31](#)

検索フィールド

データマップレコードからの削除 (IMS) [108](#)

データマップレコード内のプロパティの表示 (IMS) [91](#)

データマップレコード内のプロパティの編集 (IMS) [91](#)

データマップ内の編集 (IMS) [77](#)

データマップレコードへの追加 (IMS) [77, 79](#)

こ

向上

VSAM データセット用の一括読み込みのパフォーマンス [138](#)

構成

IMS データにアクセスするための Windows 上での

PowerExchange dbmover.cfg ファイル [126](#)

構文規則

ファイルリストファイル [99](#)

コードページ

PowerExchange 用のサポート [62](#)

概要 [61](#)

データベース行のテストでのサポート [230](#)

データマップレコード内のフィールド [153, 389](#)

ユーザー定義 [63](#)

コピー

データマップ内の項目 [92](#)

データマップレコード内のフィールドデータ [308](#)

コピーブック

ADACMP [143](#)

COBOL [143, 268, 272, 279, 283](#)

Datacom [156](#)

DATACOM [143](#)

DB2 カタログ [143, 154, 158](#)

DBD [143](#)

IDMS [143, 160](#)

PL/I [143](#)

PREDICT [143](#)

TEXT [143](#)

インポート用のソース詳細 [149](#)

概要 [55, 142](#)

初期設定のデータ定義モジュール (DDM) [143](#)

データマップへのインポート [144, 155](#)

データマップセグメントへのインポート [146](#)

データマップへのインポート (IMS) [133](#)

データマップレコードへのインポート [146](#)

フィールド定義テーブル (FDT) [143](#)

リモートの詳細 [155](#)

ローカル詳細 [153](#)

コピーブックタイプ

データマップ内のサポート [143](#)

コマンド

データベース行のテスト内の発行 [234](#)

コンパイル

ユーザーイグジットプログラム [287](#)

さ

削除

PowerExchange リソース [33](#)

アプリケーショングループ [350](#)

キャプチャ登録 [192](#)

キャプチャ登録からの列 [192](#)

抽出グループ [212](#)

抽出グループ内の抽出マップ [212](#)

抽出マップ [211](#)

データマップ内の項目 [104](#)

データマップレコードからの検索フィールド (IMS) [108](#)

データマップ [104](#)

データマップ、エクスポート [104](#)

データマップセグメント [105](#)

データマップセグメントからのフィールド [105](#)

データマップテーブル [105](#)

データマップテーブルからの列 [106](#)

データマップレコード [105](#)

データマップレコードからの CALC エlement (IDMS) [107](#)

データマップレコードからの Element (Datacom) [106](#)

データマップレコードからのオーナーレコードとセット (IDMS)

[107](#)

データマップレコードからのキー (Datacom) [106](#)

データマップレコードからのキーセグメント (Datacom) [107](#)

データマップレコードからのフィールド [105](#)

登録グループ [193](#)

パーソナルメタデータプロファイル [228](#)

リソース構成 [38](#)

抽出処理からの列 [204](#)

登録グループ内のすべての登録 [194](#)

作成

共有リソース構成 [37](#)

抽出マップからのデータマップ [210](#)

ファイルリストファイル [100](#)

ユーザーイグジットプログラム [287](#)

し

式

PowerExchange 関数 [304](#)

データマップレコードへの追加 (IMS) [128](#)

データマップレコードへの追加 (IDMS) [122](#)

シーケンシャルデータセット

データマップ [53, 365](#)

シーケンス番号

データマップレコード内のフィールドに対する取得 (IDMS) [323](#)

データマップレコード内のフィールドに対する取得 (IMS) [323](#)

実行

データベース行のテスト [229](#)

初期設定のデータ定義モジュール (DDM) コピーブック

サポートされたデータマップタイプ [143](#)

ショートカットキー

PowerExchange Navigator [29](#)

概要 [21](#)

ショートカットメニュー

PowerExchange Navigator [21, 25](#)

す

ステータスバー
PowerExchange Navigator [30](#)
非表示 [30](#)
表示 [30](#)

せ

制御間隔アクセス
VSAM データセット [139](#)
生成
暗号化されたパスワード [48, 49](#)
データマップレコード内のフィールド用の複数のテーブル行 [310](#)
整列の正規化
テスト [282](#)
設定 - [IDMS のプロパティ] ダイアログボックス [48](#)
選択
データマップ用の定義 [44](#)
選択
データマップテーブル内の列(DB2 カタログ) [159](#)
データマップ用アクセス方式 [53](#)
リソース構成 [37](#)

そ

送信
リモートシステムへのデータマップ [64, 95](#)
相対バイトアドレス (RBA)
GetDatabaseKey 関数 [141](#)
データセットレコードの取得 (VSAM) [140](#)
相対レコード番号 (RRN)
GetDatabaseKey 関数 [141](#)
データセットレコードの取得 (VSAM) [140](#)
ソート
データマップ内のテーブル [85](#)
データマップ内のレコード [85](#)
ソース詳細
コピーブックのインポート [149](#)
ソースファイル
データマップレコードのための名前の取得 [310](#)
ソートエレメント詳細
オーナーレコードとセット (IDMS) [379](#)

た

タイトルバー
PowerExchange Navigator [21](#)
タグ名
キャプチャ登録 [165](#)
単一セグメントデータマップ (IMS)
追加 [132](#)
単一レコードデータマップ
追加 [259](#)
定義済み [56](#)
テスト [262](#)
レコードの追加 [260](#)
レコードヘフィールドの追加 [260](#)
単純なテーブル
データマップ [57](#)

ち

抽出開始ポイント
アプリケーションに対するリセット [350](#)
抽出グループ
概要 [195](#)
削除 [212](#)
すべての抽出マップの削除 [212](#)
追加 [198](#)
プロパティの表示 [207](#)
プロパティの編集 [207](#)
抽出定義ウィンドウ
[マップ] タブ [213](#)
[拡張] タブ [213](#)
概要 [19](#)
抽出マップ
BI 列の表示 [213](#)
CI 列の表示 [213](#)
PowerExchange が生成したカラム [214](#)
概要 [195](#)
キャプチャ登録の変更 [205](#)
削除 [211](#)
抽出グループ内のすべての削除 [212](#)
[データベース行のテスト] ダイアログボックス [232](#)
データマップとの統合 [206](#)
データ列への操作前の画像 (BI) の列の追加 [203](#)
データ列への変更インジケータ (CI) の列の追加 [203](#)
テスト [232](#)
データベース行のテストによる出力ウィンドウ [232](#)
データマップのさくせい [210](#)
プロパティの表示 [209](#)
変更データキャプチャ [195](#)
列の検索 [32, 396](#)
抽出マップの追加 [200](#)
表示 [202](#)
名前 [196](#)
抽出マップ内の列
検索 [32](#)
削除 [204](#)
操作前の画像 (BI) の列の削除 [203](#)
操作前の画像 (BI) の列の追加 [203](#)
追加 [204](#)
表示 [202](#)
変更インジケータ (CI) の列の削除 [203](#)
変更インジケータ (CI) の列の追加 [203](#)

つ

追加
アプリケーショングループ [346](#)
キャプチャ登録 [175](#)
キャプチャ登録への列 [192](#)
抽出グループ [198](#)
データマップ (SEQ) [135](#)
データマップレコードへの検索フィールド (IMS) [79](#)
データマップレコードへの式 (IDMS) [122, 128](#)
データマップ [66, 259](#)
データマップ (Adabas) [111](#)
データマップ (Datacom) [113](#)
データマップ (DB2 アンロードファイル) [117, 119](#)
データマップ (DB2 カタログ) [115](#)
データマップ (IDMS) [120](#)
データマップ (IMS) [132, 134](#)
データマップ (VSAM) [136](#)
データマップテーブルへのキー (Adabas) [80, 112](#)
データマップテーブルへの列 [80](#)
データマップレコード [69](#)

追加 (続く)

- データマップレコードへの CALC エlement (IDMS) [75](#)
 - データマップレコードへの Element (Datacom) [71](#)
 - データマップレコードへのオーナーレコードとセット (IDMS) [75](#)
 - データマップレコードへのキー (Datacom) [72](#)
 - データマップレコードへのキーセグメント (Datacom) [74](#)
 - データマップレコードへのフィールド [70, 260](#)
 - データマップレコードへのユーザー定義フィールド [70](#)
 - テーブルをデータマップに [80, 262](#)
 - パーソナルメタデータプロファイル [223](#)
 - 複雑なテーブルがあるデータマップ [265](#)
 - リソース構成 [35](#)
 - 抽出マップ [200](#)
 - 抽出マップ列への操作前の画像 (BI) の列 [203](#)
 - 抽出マップ列への変更インジケータ (CI) の列 [203](#)
 - 抽出処理への列 [204](#)
 - 登録グループ [169](#)
- ## 追加のプロパティ
- データマップレコード内のフィールド [388](#)
- ## ツールバー
- PowerExchange Navigator [28](#)
 - 非表示 [28](#)
 - 表示 [28](#)

て

定義

- IDMS データマップ設定 [48](#)
- ソースまたはターゲット内のデータのレイアウト [55](#)
- データマップ設定 [44](#)
- 複雑なテーブル [134](#)
- リソース構成 [34](#)

データ

- ログオン認証情報 [93](#)

データソース

- データマップ用にサポート [53, 109](#)

データチェック

- 概要 [63](#)
- データマッププロパティ [371](#)
- データマップレコード内のフィールド [389](#)

データデフォルト

- データマップ用の定義 [370](#)

データファイル

- データマップがある複数の関連付け [54](#)
- データマップとの関連付け [263, 266](#)
- データマップと複数のデータファイルの関連付け [99](#)
- 表示 [83](#)
- ファイルリスト処理 [54](#)
- モードの表示 [83](#)

データベースキー

- データマップレコードのための取得 [312](#)
- データマップレコードのための取得 (IDMS) [316](#)
- データマップレコード内のオーナーレコードとセットに対する取得 (IDMS) [315, 318](#)

データマップ (DB2 カタログ)

- 追加 [115](#)
- 概要 [114](#)

データマップ (IMS)

- DBD ソースのインポートで作成された階層の表示 [128](#)
- 概要 [123](#)
- コピーブックのインポート [133](#)
- 追加 [127, 132, 134](#)
- テスト [131](#)
- テーブルの表示 [127](#)
- テーブル用プロパティの編集 [130, 394](#)
- テーブル用メタデータの表示 [130, 394](#)
- プロパティの編集 [129](#)

データマップ (IMS) (続く)

- リモートシステムへの送信 [129](#)

例 [131](#)

- レコードの表示 [127](#)

データマップ (SEQ)

- 階層の表示 [136](#)

追加 [135](#)

データマップセグメント内のフィールド (IMS)

- RBA 値の取得 [319](#)

データマップ内のテーブル (DB2)

- フィルタリング [356](#)

データマップレコード (IDMS)

- 基数の取得 [322](#)
- 式の追加 [122](#)
- シーケンス番号の取得 [323](#)
- データフロー情報の取得 [313](#)
- ページグループおよび基数の取得 [321](#)
- ページグループの取得 [320](#)
- 完全修飾データベースキーの取得 [316](#)

データマップレコード (IMS)

- CCK フィールド [90](#)
- CCK フィールドの編集 [77](#)
- CCK フィールドの追加 [77, 78](#)
- RBA の取得 [128](#)
- 検索フィールドの編集 [77](#)
- 検索フィールドの削除 [108](#)
- 検索フィールドの追加 [77, 79](#)
- 検索フィールド用プロパティの表示 [91](#)
- 検索フィールド用プロパティの編集 [91](#)
- 式の追加 [128](#)
- シーケンス番号の取得 [323](#)

データマップレコード (IMS DL/1 バッチ)

- CCK フィールドの削除 [107](#)

データマップレコード (IMS ODBA)

- CCK フィールドの削除 [107](#)

データマップレコード内の CCK フィールド (IMS ODBA)

- 概要 [65](#)

データマップレコード内の検索フィールド (IMS ODBA)

- 概要 [65](#)

データマップを追加するための前提条件

- IMS データマップ [124](#)

テーブル

- データマップへ追加 [262](#)

テスト

- 抽出マップ [232](#)
- データマップ [231](#)
- データマップ (IMS) [131](#)
- パーソナルメタデータプロファイル [233](#)
- 単レコードデータマップ [262](#)
- 複数レコードデータマップ [267](#)
- データ記述ブロック (DBD) コピーブック
- サポートされたデータマップタイプ [143](#)

データの書き込み

- IMS データベース [130](#)

データのレイアウト

- データのソースまたはターゲット [55](#)

データバッファ

- VSAM データセット [138](#)

データフロー情報

- データマップレコードのための取得 (IDMS) [313](#)

データベース行のテスト

- [データベース行のテスト] ダイアログボックス [237](#)
- [全般] タブの CAPXRT の [詳細パラメータ] [249](#)
- [Adabas] タブの CAPXRT の [詳細パラメータ] [253](#)
- [AS400] タブの CAPXRT の [詳細パラメータ] [253](#)
- CAPX の全般詳細パラメータ [246](#)
- DB2390IMG 詳細パラメータ [255](#)
- [DB2UDB] タブの CAPXRT の [詳細パラメータ] [253](#)

データベース行のテスト (続く)

i5/OS ソースの CAPX の [詳細パラメータ] [248](#)
IMSUNLD および NRDB の詳細パラメータ [255](#)
[MSSQL] タブの CAPXRT の [詳細パラメータ] [254](#)
[Oracle] タブの CAPXRT 詳細パラメータ [254](#)
PowerExchange リスナコマンドの発行 [234](#)

概要 [229](#)

実行 [229](#)

整列の正規化のテスト [282](#)

データマップレコード内のユーザー出口プログラムの結果のテスト
[302](#)

非標準コードページデータ [230](#)

マルチバイトデータ [230](#)

ユーザー定義フィールドに対する実行 [338](#)

リスタートトークンの生成 [235](#)

データベース行のテストによる出力ウィンドウ

非表示 [230](#)

表示 [230](#)

データマップ

3 層名 [52](#)

2 層名 [52](#)

Adabas データベース [352](#)

COBOL コピーブックのインポート [268](#), [273](#), [279](#), [283](#)

Datacom ファイル [354](#)

DB2 データベース [355](#)

GROUP フィールドへの日付マスクの適用 [274](#)

IDMS データベース [361](#)

IMS データベース [358](#)

MQSeries メッセージキュー用 [362](#)

PL/I コピーブックのインポート [276](#)

USER アクセス方式 [339](#)

VSAM エントリシーケンスのデータセット (ESDS) [360](#)

VSAM キーシーケンスのデータセット (KSDS) [362](#)

VSAM 相対レコードデータセット (RRDS) [364](#)

アクセス方式のタイプ [53](#)

印刷 [101](#)

概要 [51](#), [52](#)

コピーブックのインポート [144](#)

削除 [104](#)

サポートされたコピーブックタイプ [143](#)

サポートされるデータソース [53](#)

シーケンシャルデータセット [365](#)

整列による追加 [279](#), [283](#)

セグメント [55](#)

セグメントからのフィールドの削除 [105](#)

セグメントの削除 [105](#)

設定の定義 [44](#)

単一レコード [259](#)

抽出マップからの作成 [210](#)

追加 [66](#), [259](#)

定義されたレコードレベルの出口プログラム [373](#)

データチェック [371](#)

データデフォルト [370](#)

テスト [231](#)

データファイルの関連付け [263](#)

テーブルデータセット [367](#)

テーブルの削除 [105](#)

テーブルのソート [85](#)

特定のデータソース [109](#)

名前 [52](#), [66](#)

配列の定義 [58](#)

ファイルリストファイルの関連付け [54](#), [100](#)

フラットファイル [365](#)

プロパティの表示 [86](#)

プロパティの編集 [86](#)

ユーザーアクセス方式プログラム [368](#)

ユーザー定義フィールドのテスト [338](#)

リモートシステムからのインポート [97](#)

データマップ (続く)

リモートシステムへの送信 [64](#), [95](#)

レコード [55](#)

レコードからのフィールドの削除 [105](#)

レコードの削除 [105](#)

レコードのソート [85](#)

レコードの追加 [69](#), [260](#)

レコードヘフィールドの追加 [70](#)

レコードへのユーザー定義フィールドの追加 [70](#)

ログオン認証情報 [93](#)

抽出マップとの統合 [206](#)

データマップ (Adabas)

概要 [110](#)

キーのインポート [55](#)

追加 [111](#)

テーブルへのキーの追加 [112](#)

フィールド定義テーブル (FDT) のインポート [148](#)

データマップ (Datacom)

概要 [113](#)

追加 [113](#)

レコードからのエレメントの削除 [106](#)

レコードからのキーセグメントの削除 [107](#)

レコードからのキーの削除 [106](#)

レコード内のエレメント [64](#)

レコード内のキー [64](#)

レコードへのエレメントの追加 [71](#)

レコードへのキーセグメントの追加 [71](#), [74](#)

レコードへのキーの追加 [71](#), [72](#)

データマップ (DB2)

テーブルのフィルタ [356](#), [358](#)

レコードレイアウトの編集 [116](#)

データマップ (DB2 アンロードファイル)

概要 [117](#)

追加 [117](#), [119](#)

データマップ (IDMS)

CALC エレメント [65](#)

オーナーレコードとセット [65](#)

オーナーレコードとセット用のソートエレメント詳細 (IDMS)
[379](#)

概要 [119](#)

追加 [120](#)

レコードからの CALC エレメントの削除 [107](#)

レコードからのオーナーレコードとセットの削除 [107](#)

レコードへの CALC エレメントの追加 [75](#)

レコードへのオーナーレコードとセットの追加 [75](#)

レコード用プロパティの表示 [377](#)

レコード用プロパティの編集 [377](#)

データマップ (VSAM)

階層の表示 [138](#)

概要 [134](#), [136](#)

追加 [136](#)

データマップアクセス方式

概要 [109](#)

タイプ [53](#)

データマップ、エクスポート

削除 [104](#)

データマップセグメント

コピーブックのインポート [146](#)

データマップセグメント内のフィールド

概要 [55](#)

削除 [105](#)

データマップテーブル

印刷 [103](#)

コピー [92](#)

削除 [105](#)

追加 [80](#), [262](#)

テーブル定義プロパティ [391](#)

表示 [83](#)

データマップテーブル (続く)

- 複雑 [57, 80](#)
- 不足しているカラムを使用したリフレッシュ [70](#)
- プロパティ [281, 301](#)
- プロパティの表示 [91, 391](#)
- プロパティの編集 [91, 391](#)
- 列の検索 [32, 396](#)
- 列の削除 [106](#)
- 列の追加 [80](#)
- 列のフィルタリング [394](#)
- データマップテーブル (Adabas)
 - キーの追加 [80, 112](#)
- データマップテーブル (DB2 カタログ)
 - 列の選択 [159](#)
- データマップテーブル内のキー (Adabas)
 - プロパティの表示 [395](#)
 - プロパティの編集 [395](#)
- データマップテーブル内のキー (Datacom)
 - プロパティの表示 [395](#)
 - プロパティの編集 [395](#)
- データマップテーブル内の列
 - 印刷 [103](#)
 - 概要 [57](#)
 - 検索 [32](#)
 - コピー [92](#)
 - 削除 [106](#)
 - 選択 (DB2 カタログ) [159](#)
 - 追加 [80](#)
 - 表示 [83](#)
 - プロパティの表示 [92](#)
 - プロパティの編集 [92](#)
- データマップテーブルプロパティ
 - 複数回出現する配列の正規化に使用 [278](#)
- データマップ内のキー (Adabas)
 - プロパティ [374](#)
- データマップ内の項目
 - 印刷 [101](#)
- データマップ内のセグメント
 - 概要 [55](#)
 - コピーブックのインポート [146](#)
 - 削除 [105](#)
 - フィールドの削除 [105](#)
- データマップ内のテーブル
 - Adabas キーの追加 [80](#)
 - 印刷 [103](#)
 - 概要 [57](#)
 - 削除 [105](#)
 - ソート [85](#)
 - 追加 [80](#)
 - 表示 [83](#)
 - プロパティ [281, 301](#)
 - プロパティの表示 [91, 391](#)
 - プロパティの編集 [91, 391](#)
 - 列の印刷 [103](#)
 - 列の検索 [32, 396](#)
 - 列の削除 [106](#)
 - 列の追加 [80](#)
 - 列用プロパティの表示 [92](#)
 - 列用プロパティの編集 [92](#)
- データマップ内のテーブル (Adabas)
 - キーの追加 [112](#)
- データマップ内のテーブル (IMS)
 - 表示 [127](#)
 - プロパティの表示 [130, 394](#)
 - プロパティの編集 [130, 394](#)
- データマップ内のレコード
 - 印刷 [102](#)
 - 概要 [55](#)

データマップ内のレコード (続く)

- コピーブックのインポート [146](#)
- 削除 [105](#)
- ソート [85](#)
- 追加 [69](#)
- 表示 [81](#)
- フィールドの印刷 [103](#)
- フィールドの削除 [105](#)
- フィールドの追加 [70](#)
- フィールドの表示 [81](#)
- フィールド用プロパティの表示 [87](#)
- フィールド用プロパティの編集 [87](#)
- プロパティの表示 [86](#)
- プロパティの編集 [86](#)
- ユーザー定義フィールド [70](#)
- レコードレベルの出口プログラム [56](#)
- データマップ内のレコード (IDMS)
 - CALC エLEMENTの削除 [107](#)
 - CALC エLEMENTの追加 [75](#)
 - CALC エLEMENT用プロパティの表示 [89](#)
 - CALC エLEMENT用プロパティの編集 [89](#)
 - 項目の追加 [75](#)
- データマップ内のレコード (IMS)
 - 表示 [127](#)
- データマップのレコード (Datacom)
 - ELEMENTの削除 [106](#)
 - ELEMENTの追加 [71](#)
 - キーセグメントの削除 [107](#)
 - キーセグメントの追加 [74](#)
 - キーセグメント用プロパティの表示 [88](#)
 - キーセグメント用プロパティの編集 [88](#)
 - キーの削除 [106](#)
 - キーの追加 [72](#)
 - キー用プロパティの表示 [88](#)
 - キー用プロパティの編集 [88](#)
 - 項目の追加 [71](#)
- データマッププロパティ
 - VSAM データセット用の RBA 取得のために使用 [140](#)
 - VSAM データセット用の RRN 取得のために使用 [140](#)
- データマップレコード
 - あるフィールドから別のフィールドへのデータのコピー [308](#)
 - 印刷 [102](#)
 - コピー [92](#)
 - コピーブックのインポート [146](#)
 - 削除 [105](#)
 - ソースファイル名の取得 [310](#)
 - 追加 [69, 260](#)
 - データベースキーの取得 [312](#)
 - データのフィルタリング [263](#)
 - 配列 [265](#)
 - フィールド内の指定オフセットでのビット設定の取得 [325, 326](#)
 - フィールド内のデータの分割 [308](#)
 - フィールドの検索 [31, 396](#)
 - フィールドの追加 [70, 260](#)
 - プロパティの表示 [86, 375](#)
 - プロパティの編集 [86, 375, 380](#)
 - ユーザー定義プログラムの呼び出し [304](#)
 - レコード ID [266](#)
 - レコードの階層の定義 [266](#)
 - 連結フィールド [307](#)
- データマップレコード (Datacom)
 - ELEMENTの追加 [71](#)
 - ELEMENT用プロパティの表示 [88](#)
 - プロパティ [376](#)
- データマップレコード内の CALC ELEMENT (IDMS)
 - 概要 [65](#)
 - 削除 [107](#)
 - 追加 [75](#)

データマップレコード内の CALC エlement (IDMS) (続く)
プロパティの表示 [89](#)
プロパティの編集 [89](#)
データマップレコード内の CCK フィールド (IMS DL/1 バッチ)
概要 [65](#)
データマップレコード内の Element (Datacom)
概要 [64](#)
削除 [106](#)
追加 [71](#)
データマップレコード内のオーナーレコードとセット (IDMS)
概要 [65](#)
削除 [107](#)
ソート Element 詳細 [379](#)
追加 [75](#)
データベースキーの取得 [315](#)
プロパティの表示 [89](#)
プロパティの編集 [89](#)
ページグループおよび基数の取得 [321](#)
ページグループの取得 [320](#)
完全修飾データベースキーの取得 [318](#)
データマップレコード内のキー (Datacom)
概要 [64](#)
削除 [106](#)
追加 [71, 72](#)
プロパティの表示 [88](#)
プロパティの編集 [88](#)
データマップレコード内のキーセグメント (Datacom)
概要 [64](#)
削除 [107](#)
追加 [71, 74](#)
プロパティの表示 [88](#)
プロパティの編集 [88](#)
データマップレコード内の検索フィールド (IMS DL/1 バッチ)
概要 [65](#)
データマップレコード内のフィールド
印刷 [103](#)
階層内のシーケンス番号の取得 [322](#)
概要 [55](#)
可変長プロパティ [390](#)
検索 [31](#)
コードページ [389](#)
コピー [92](#)
コンテンツの取得 [306, 327](#)
削除 [105](#)
追加 [70, 260](#)
データチェックプロパティ [389](#)
長さの取得 [324](#)
配列として定義 [387](#)
日付マスクの適用 [271, 274](#)
表示 [81](#)
複数のテーブル行の生成 [310](#)
プロパティの表示 [87](#)
プロパティの編集 [87, 380, 387, 388](#)
文字の除去 [327](#)
文字列の大文字への変換 [329](#)
文字列の小文字への変換 [328](#)
文字のトリミング [324, 325](#)
データマップレコード内のフィールド (Adabas)
プロパティの編集 [387](#)
データマップレコード内のフィールド (IDMS)
階層内のシーケンス番号の取得 [322](#)
データマップレコード内のフィールド (IMS)
階層内のシーケンス番号の取得 [322](#)
データマップレコード用のオーナーレコードとセット (IDMS)
基数の取得 [322](#)
ページグループの取得 [321](#)
テーブルデータセット
データマップ [53, 367](#)

テーブルウィンドウ
概要 [19](#)
テーブル [83](#)
列 [83](#)

と

閉じる
PowerExchange リソース [33](#)
トリミング
データマップレコード内のフィールドからの文字 [324, 325](#)

は

配列
概要 [60](#)
各 Element 用の新しい行の定義 [264](#)
すべての Element 用の単一行の定義 [265](#)
データマップ内の定義 [58](#)
データマップレコード内のフィールドの定義 [387](#)
パーソナルメタデータテーブル
プロパティの表示 [226](#)
パーソナルメタデータプロファイル
概要 [222](#)
削除 [228](#)
[データベース行のテスト] ダイアログボックス [233](#)
テスト [233](#)
データベース行のテストによる出力ウィンドウ [233](#)
テーブルの検索 [32, 396](#)
テーブル用メタデータの表示 [226, 227](#)
プロパティの表示 [227](#)
プロパティの編集 [227](#)
メタデータのリフレッシュ [228](#)
追加 [223](#)
パーソナルメタデータプロファイルテーブル
列の検索 [33](#)
パーソナルメタデータプロファイルテーブル内の列
検索 [33](#)
パーソナルメタデータプロファイル内のテーブル
検索 [32](#)
列内のメタデータの表示 [226](#)
プロパティの表示 [227](#)
列の検索 [33, 396](#)
パラメータ構造体
ユーザーアクセス方式プログラムへの受け渡し [341](#)
パラメータリスト
ユーザー定義プログラムへの受け渡し [330](#)

ひ

日付マスク
データマップレコード内の GROUP フィールドへの適用 [274](#)
データマップレコード内のフィールドへの適用 [271](#)
非表示
ステータスバー [30](#)
ツールバー [28](#)
データベース行のテストによる出力ウィンドウ [230](#)
リソースエクスプローラ [20](#)
表示
Adabas フィールド定義テーブル (FDT) [149](#)
OCCURS DEPENDING ON 句内の項目に対する複数出力行 [272](#)
SEQ データマップの階層 [136](#)
VSAM データマップの階層 [138](#)
キャプチャ登録用の Element (Datacom) [191](#)
キャプチャ登録用ベースレコードの親階層 (IDMS) [192](#)

表示 (続く)

ステータスバー [30](#)
ツールバー [28](#)
データファイル [83](#)
データマップ内の DBD ソースのインポートで作成された階層 (IMS) [128](#)
データマップレコード内の CCK フィールド (IMS) [90](#)
データマップレコード内のエレメントのプロパティ (Datacom) [88](#)
データマップレコード内の検索フィールドのプロパティ (IMS) [91](#)
データベース行のテストによる出力ウィンドウ [230](#)
データマップテーブル [83](#)
データマップテーブル内の列 [83](#)
データマップレコード [81, 266](#)
データマップレコード (IMS) [127](#)
データマップレコード内のフィールド [81](#)
リソースエクスプローラ [20](#)
抽出マップ [202](#)
開く
PowerExchange リソース [31](#)

ふ

ファイルリストファイル

ODBC [100](#)
構文規則 [99](#)
作成 [100](#)
処理 [99](#)
データマップとの関連付け [54, 99](#)
例 (Linux) [100](#)
例 (UNIX) [100](#)
例 (Windows) [100](#)
例 (z/OS) [100](#)

CCK フィールド

データマップレコードからの削除 (IMS DL/1 バッチ) [107](#)
データマップレコードからの削除 (IMS ODBA) [107](#)
データマップレコード内の表示 (IMS) [90](#)
データマップ内の編集 (IMS) [77, 90](#)
データマップレコードへの追加 (IMS) [77, 78](#)

フィルタリング

データマップテーブル内の列 [394](#)
データマップ内のテーブル (DB2) [356, 358](#)
データマップレコード内のデータ [263](#)

[フィールド] タブ

レコードウィンドウ [81](#)

フィールド定義テーブル (FDT)

データマップのためのリフレッシュ (Adabas) [149](#)
データマップへのインポート (Adabas) [55, 148](#)
データマップ用の表示 (Adabas) [149](#)

フィールド定義テーブル (FDT) コピーブック

サポートされたデータマップタイプ [143](#)

フィールド内のオフセットでのビット設定

データマップレコードのための取得 [325, 326](#)

複雑なテーブル

サポートするデータソース [57](#)

定義 [134](#)
データマップ [57](#)
データマップ内の定義 [265](#)
データマップへ追加 [80](#)

複数回出現する配列

データマッププロパティの定義 [278](#)
正規化を示すユーザー定義フィールドの追加 [280](#)

複数セグメントデータマップ

追加 [134](#)

複数のテーブルからの DB2 アンロードデータ

アクセス [118](#)

複数レコードデータマップ

定義済み [56](#)

テスト [267](#)

フラットファイル

データマップ [53, 365](#)

プロパティ

アプリケーショングループ [348](#)

オーナーレコードとセット内のエレメント (DMS) [379](#)

キャプチャ登録 [186](#)

データマップ (DL/1 バッチアクセス方式) [358](#)

データマップ (IMS ODBA アクセス方式) [358](#)

データマップ (シーケンシャルデータセット) [365](#)

データマップ (フラットファイル) [365](#)

データマップ (ADABAS アクセス方式) [352](#)

データマップ (DATACOM アクセス方式) [354](#)

データマップ (DB2 アクセス方式) [355](#)

データマップ (DB2 アンロードファイル) [355](#)

データマップ (ESDS アクセス方式) [360](#)

データマップ (IDMS アクセス方式) [361](#)

データマップ (KSDS アクセス方式) [362](#)

データマップ (MQSeries アクセス方式) [362](#)

データマップ (RRDS アクセス方式) [364](#)

データマップ (USER アクセス方式) [368](#)

データマップ (テーブルデータセット) [367](#)

データマップについてのデータチェック [371](#)

データマップ用のデータデフォルト [370](#)

データマップに対して定義されたレコードレベルの出口プログラム

[373](#)

データマップ [86](#)

データマップテーブル [91](#)

データマップテーブル (IMS) [394](#)

データマップテーブル内のキー (Adabas) [395](#)

データマップテーブル内の列 [92](#)

データマップテーブル内の列用のフィルタ条件 [394](#)

データマップテーブル用の定義 [391](#)

データマップ内のキー (Adabas) [374](#)

データマップレコード [86, 375, 380](#)

データマップレコード (Datacom) [376](#)

データマップレコード (IDMS) [121, 377](#)

データマップレコード内の CALC エレメント (IDMS) [89](#)

データマップレコード内のオーナーレコードとセット (IDMS) [89](#)

データマップレコード内のキー (Datacom) [88](#)

データマップレコード内のキーセグメント (Datacom) [88](#)

データマップレコード内のフィールド [87, 380, 387, 388, 390](#)

データマップレコード内のフィールド (Adabas) [387](#)

パーソナルメタデータプロファイル [227](#)

パーソナルメタデータプロファイルテーブル [227](#)

抽出グループ [207](#)

抽出マップ [209](#)

登録グループ [184](#)

プロパティの表示

アプリケーショングループ [348](#)

キャプチャ登録 [186](#)

データマップレコード内のフィールド [87](#)

データマップ [86](#)

データマップテーブル [91](#)

データマップテーブル (IMS) [130, 394](#)

データマップテーブル内のキー (Datacom) [395](#)

データマップテーブル内の列 [92](#)

データマップテーブル用の定義 [391](#)

データマップレコード [86, 375](#)

データマップレコード (IDMS) [377](#)

データマップレコード内の CALC エレメント (IDMS) [89](#)

データマップレコード内のオーナーレコードとセット (IDMS) [89](#)

データマップレコード内のキー (Datacom) [88](#)

データマップレコード内のキーセグメント (Datacom) [88](#)

パーソナルメタデータプロファイル [227](#)

パーソナルメタデータプロファイルテーブル [226, 227](#)

プロパティの表示 (続く)

抽出グループ [207](#)

抽出マップ [209](#)

登録グループ [184](#)

プロパティの編集

アプリケーショングループ [348](#)

キャプチャ登録 [186](#)

データマップレコード内のフィールド [87](#)

データマップ [86](#)

データマップ (IMS) [129](#)

データマップテーブル [91](#)

データマップテーブル (IMS) [130](#), [394](#)

データマップテーブル内のキー (Adabas) [395](#)

データマップテーブル内のキー (Datacom) [395](#)

データマップテーブル内の列 [92](#)

データマップテーブル用の定義 [391](#)

データマップレコード [86](#), [375](#)

データマップレコード (IDMS) [121](#), [377](#)

データマップレコード内の CALC エlement (IDMS) [89](#)

データマップレコード内のオーナーレコードとセット (IDMS) [89](#)

データマップレコード内のキー (Datacom) [88](#)

データマップレコード内のキーセグメント (Datacom) [88](#)

パーソナルメタデータプロファイル [227](#)

抽出グループ [207](#)

登録グループ [184](#)

分割

データマップレコード内のフィールドデータ [308](#)

へ

ページグループ

データマップレコード内のオーナーレコードとセットに対する取得 (IDMS) [320](#), [321](#)

データマップレコードのための取得 (IDMS) [320](#), [321](#)

変換

データマップレコード内のフィールドで大文字になる文字 [329](#)

データマップレコード内のフィールドで小文字になる文字 [328](#)

変更

キャプチャ登録内の列 [192](#)

抽出マップ用のキャプチャ登録 [205](#)

リソースウィンドウ用のビュータイプ [21](#)

編集

データマップレコード内の CCK フィールド (IMS) [90](#)

データマップレコード内の検索フィールドのプロパティ (IMS) [91](#)

データマップ内のレコードレイアウト (DB2) [116](#)

リソース構成 [35](#)

列を変更するためのキャプチャ登録 [192](#)

ま

マルチバイトデータ

データベース行のテスト [230](#)

め

命名

データマップ [52](#), [66](#)

メタデータ

パーソナルメタデータプロファイル内のリフレッシュ [228](#)

メニュー

メニューバー [21](#)

メニューコマンド

PowerExchange Navigator [21](#)

概要 [21](#)

メニューバー

概要 [21](#)

も

モードの表示

データファイル [83](#)

戻りコード

CallProg 関数で呼び出されるプログラムのための定義 [305](#)

ユーザーアクセス方式プログラム [343](#)

ユーザー定義プログラム [332](#)

ゆ

ユーザーイグジットプログラム

コンパイル [287](#)

作成 [287](#)

サンプル [287](#)

データマップレコードでの作成と呼び出し [282](#)

ユーザーアクセス方式プログラム

概要 [65](#), [339](#)

データマップ [368](#)

戻りコード [343](#)

例 [344](#)

渡されたパラメータ構造体 [341](#)

ユーザー定義関数

CallProg [56](#), [304](#)

Check [305](#)

CheckNumData [306](#)

Concat [307](#)

CopyData [308](#)

Fragment [308](#)

GenVRowKey [310](#)

GetCurrentFileName [310](#), [311](#)

GetDatabaseKey [312](#)

GetDataFlowType [313](#)

GetDbKey [314](#)

GetDbKeyOfFirstMember [314](#)

GetDbKeyOfLastMember [315](#)

GetDbKeyOfNextMember [315](#)

GetDbKeyOfOwner [315](#)

GetDbKeyOfPriorMember [316](#)

GetFullDbKey [316](#)

GetFullDbKeyOfFirstMember [317](#)

GetFullDbKeyOfLastMember [317](#)

GetFullDbKeyOfNextMember [318](#)

GetFullDbKeyOfOwner [318](#)

GetFullDbKeyOfPriorMember [319](#)

GetIMSRBABByLevel [319](#)

GetPageGroup [320](#)

GetPageGroupOfOwner [320](#)

GetPgGrpAndRdx [321](#)

GetPgGrpAndRdxOfOwner [321](#)

GetPgGrpOfOwner [321](#)

GetRadix [322](#)

GetRadixOfOwner [322](#)

GetSeqWithinLevel [322](#)

GetSeqWithinParent [323](#)

LengthOf [324](#)

LTrim [324](#)

RTrim [325](#)

SetBitA [325](#)

SetBitN [326](#)

Strip [327](#)

ToLower [328](#)

ToUpper [329](#)

ユーザー定義関数 (続く)

分割 [327](#)

ユーザー定義コードページ [63](#)

ユーザー定義フィールド

PowerExchange 関数 [303](#)

概要 [56](#)

式を使用 [297](#)

データマップ内のテスト [338](#)

データマップレコードへの追加 [70](#)

式のための関数 [304](#)

整列の正規化の明示 [280](#)

ユーザー定義プログラム

エラー [329](#)

サンプル C プログラム [333](#)

データマップレコードからの呼び出し [304](#)

戻りコード [332](#)

例 (COBOL 言語) [334](#)

例 (PL/I 言語) [337](#)

例 (RPG 言語) [336](#)

例 (アセンブリ言語) [335](#)

渡されたパラメータリスト [330](#)

ユーザー出口プログラム

テスト [302](#)

り

リスタートトークン

変更データキャプチャのための生成 [235](#)

リセット

アプリケーション用の抽出開始ポイント [350](#)

リソース

PowerExchange [30](#)

リソースインスペクタ

概要 [17](#), [19](#)

リソースウィンドウ

概要 [17](#), [19](#)

ビュータイプの変更 [21](#)

リソースエクスプローラ

概要 [17](#), [19](#)

非表示 [20](#)

表示 [20](#)

リソース構成

共有の作成 [37](#)

削除 [38](#)

選択 [37](#)

追加 [35](#)

定義 [34](#)

編集 [35](#)

リフレッシュ

Adabas フィールド定義テーブル (FDT) [149](#)

パーソナルメタデータプロファイル内のメタデータ [228](#)

不足しているカラムを使用したデータマップテーブル [70](#)

リモート Datacom 詳細

コピーブック [156](#)

リモート DB2 カタログの詳細

コピーブックのインポート [158](#)

リモートシステム

データマップの送信 [95](#)

リモートの詳細

コピーブック [155](#)

る

ルックアップトランスフォーメーション

IMS データベース内のデータの検索に使用 [129](#)

れ

例

Check 関数 [306](#)

Concat 関数 [307](#)

Fragment 関数 [309](#)

GenVRowKey 関数 [310](#)

GetSeqWithinLevel 関数 [323](#)

GetSeqWithinParent 関数 [323](#)

IMS データマップ [131](#)

LTrim 関数 [324](#)

PowerExchange Navigator [258](#)

REDEFINES を含む COBOL コピーブックのインポート [272](#)

RTrim 関数 [325](#)

Split 関数 [327](#)

Strip 関数 [328](#)

単一セグメントデータマップの追加 (IMS) [132](#)

単一レコードデータマップの追加 [259](#)

単一レコードデータマップへのテーブルの追加 [262](#)

単一レコードデータマップへのフィールドの追加 [260](#)

単一レコードデータマップへのレコードの追加 [260](#)

データマップへの PL/I コピーブックのインポート [276](#)

データマップの追加 (DB2 アンロードファイル) [119](#)

データマップの追加 (単一レコード) [259](#)

データマップへの COBOL コピーブックのインポート [268](#)

配列を含む単一レコードデータマップ [263](#)

ファイルリストファイル (Linux) [100](#)

ファイルリストファイル (UNIX) [100](#)

ファイルリストファイル (Windows) [100](#)

ファイルリストファイル (z/OS) [100](#)

複雑なテーブルがある複数レコードデータマップの追加 [265](#)

複数セグメントデータマップのつか (IMS) [134](#)

ユーザーアクセス方式プログラム [344](#)

ユーザー定義プログラム [333](#)

ユーザー定義プログラム (CL 言語) [336](#)

ユーザー定義プログラム (COBOL 言語) [334](#)

ユーザー定義プログラム (C 言語) [333](#)

ユーザー定義プログラム (PL/I 言語) [337](#)

ユーザー定義プログラム (RPG 言語) [336](#)

ユーザー定義プログラム (アセンブリ言語) [335](#)

レコード ID

データマップレコード [266](#)

レコードウィンドウ

[Expr(n)] タブ [81](#)

概要 [19](#)

フィールド [81](#)

[フィールド] タブ [81](#)

レコード [81](#)

レコードレイアウト

データマップ内の編集 (DB2) [116](#)

レコードレベルの出口プログラム

概要 [56](#)

プロパティ [373](#)

ろ

ローカル DB2 カタログ詳細

コピーブックのインポート [154](#)

ログオン認証情報

データへのアクセス [93](#)

データマップへのアクセス [93](#)

わ

ワイルドカード

PowerExchange リソース用の検索条件 [31](#)