

# マイグレーション・マネージャーを使用した マイグレーションのユースケース

Tivoli Service Request Manager、CCMDB、

Tivoli's process automation engine

(Tivoli/Maximo Base Service) のマイグレーション

マイグレーション・マネージャーの  
実装戦略を学ぶ

実践的なシナリオを試す

マイグレーションに関する  
トラブルシューティングの  
ヒントを学ぶ



Bradley Downing

Burak Bolek

David A Gutierrez

John Dibble

Kylie Cameron

Sampath Sriramadhesikan

Scott Tyrrell

Thadeu de Russo e Carmo

Vasfi Gucer





International Technical Support Organization

マイグレーション・マネージャーを使用した  
マイグレーションのユースケース

2011 年 2 月

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

## 第 1 刷 2011.2

本版は、IBM Tivoli Change and Configuration Management Database V7.2.0 および V7.2.1、IBM Tivoli Service Request Manager V7.2.0 and V7.2.1、Tivoli's process automation engine V7.1.1.6 and V7.1.1.7 に適用されます。

本書は 2010 年 12 月 17 日版の日本語訳です。※原文（英語）は 2011 年 2 月に更新されています。

# 目次

図 .....	.ix
表 .....	.xv
例 .....	.xvii
<b>特記事項 .....</b>	.xix
商標 .....	.xx
<b>前書き .....</b>	.xxi
本書の執筆に携わったチームについて .....	.xxi
執筆にご協力ください .....	.xxv
ご意見をお寄せください .....	.xxv
IBM Redbooks 関連情報 .....	.xxvi
<b>第1章 マイグレーション戦略 .....</b>	.1
1.1 要件 .....	.3
1.2 マイグレーションをサポートするツールの選択 .....	.3
1.2.1 データ・ロード・ツール .....	.4
1.2.2 データ・ロードの順序 .....	.6
1.3 コンテンツ戦略 .....	.8
1.3.1 構成コンテンツ間の依存関係 .....	.8
1.3.2 ベスト・プラクティス・コンテンツ .....	.9
1.3.3 構成コンテンツの完全性 .....	.10
1.3.4 コンテンツの検証 .....	.11
1.3.5 コンテンツおよびソース制御システム .....	.13
1.4 マイグレーション方法 .....	.16
1.4.1 スナップショットの特性 .....	.16
1.4.2 変更の特性 .....	.16
1.4.3 混成マイグレーション戦略 .....	.17
1.5 マイグレーション計画 .....	.18
1.5.1 マイグレーション計画の内容 .....	.19
1.5.2 マイグレーションの時間枠での作業 .....	.25
1.6 マイグレーション・マネージャーに関する参考資料 .....	.27
<b>第2章 データ・ディクショナリーのマイグレーション .....</b>	.29
2.1 要件 .....	.30
2.2 ソリューション .....	.30
2.3 構成アプリケーション .....	.31

2.4 オブジェクト構造 .....	32
2.5 マイグレーション・グループ .....	36
2.6 パッケージ定義 .....	37
2.7 デプロイメント .....	48
2.8 デプロイメントに関する考慮事項 .....	49
2.8.1 組織および清算勘定 .....	49
2.8.2 クロスオーバー・ドメインおよびテーブル・ドメイン .....	49
<b>第3章 セキュリティ構成データのマイグレーション .....</b>	<b>51</b>
3.1 新規セキュリティ・グループのマイグレーション .....	52
3.1.1 要件 .....	52
3.1.2 ソリューション .....	52
3.1.3 構成アプリケーション .....	53
3.1.4 オブジェクト構造 .....	55
3.1.5 マイグレーション・グループ .....	56
3.1.6 パッケージ定義 .....	57
3.1.7 デプロイメント .....	59
3.1.8 デプロイメントに関する考慮事項 .....	59
3.2 条件付きユーザー・インターフェースのマイグレーション .....	61
3.2.1 要件 .....	62
3.2.2 ソリューション .....	62
3.2.3 構成アプリケーション .....	62
3.2.4 オブジェクト構造 .....	66
3.2.5 マイグレーション・グループ .....	66
3.2.6 パッケージ定義 .....	67
3.2.7 デプロイメント .....	70
3.2.8 デプロイメントに関する考慮事項 .....	70
3.3 グローバルなデータ制限のマイグレーション .....	70
3.3.1 要件 .....	71
3.3.2 ソリューション .....	71
3.3.3 構成アプリケーション .....	71
3.3.4 オブジェクト構造 .....	73
3.3.5 マイグレーション・グループ .....	74
3.3.6 パッケージ定義 .....	75
3.3.7 デプロイメント .....	77
3.3.8 デプロイメントに関する考慮事項 .....	77
3.4 アクセス権限定義およびLDAP情報のマイグレーション .....	78
3.4.1 要件 .....	79
3.4.2 ソリューション .....	79
3.4.3 構成アプリケーション .....	79
3.4.4 オブジェクト構造 .....	80
3.4.5 マイグレーション・グループ .....	80
3.4.6 パッケージ定義 .....	82

3.4.7 デプロイメント	83
3.4.8 デプロイメントに関する考慮事項	84
3.5 セキュリティー・マイグレーションに関する考慮事項	85
<b>第4章 エスカレーションのマイグレーション</b>	87
4.1 要件	89
4.2 ソリューション	89
4.3 構成アプリケーション	89
4.3.1 アクション	89
4.3.2 担当者	90
4.3.3 担当者グループ	90
4.3.4 ロール	90
4.3.5 通信テンプレート	90
4.3.6 エスカレーション	90
4.3.7 その他のアプリケーション	91
4.4 オブジェクト構造	92
4.5 マイグレーション・グループ	94
4.6 パッケージ定義	97
4.6.1 アクションが関連付けられているエスカレーションのSQL基準の定義	98
4.6.2 通知が関連付けられているエスカレーションのSQL基準の定義	101
4.7 デプロイメント	104
4.7.1 エスカレーションとクーロン・タスク	104
<b>第5章 アプリケーションとスタート・センターのマイグレーション</b>	115
5.1 基本的なアプリケーション変更と署名オプション	116
5.1.1 要件	116
5.1.2 ソリューション	118
5.1.3 このソリューションに使用する構成アプリケーション	118
5.1.4 オブジェクト構造	118
5.1.5 マイグレーション・グループ	119
5.1.6 パッケージ定義	121
5.1.7 デプロイメント	123
5.1.8 デプロイメントに関する考慮事項	123
5.2 メニュー、ルックアップ、グローバルなデータ制限などの複雑なアプリケーション変更	124
5.2.1 要件	124
5.2.2 ソリューション	126
5.2.3 このソリューションに使用する構成アプリケーション	126
5.2.4 オブジェクト構造	128
5.2.5 マイグレーション・グループ	131
5.2.6 パッケージ定義	134
5.2.7 デプロイメント	137

5.2.8 デプロイメントに関する考慮事項 .....	138
5.3 スタート・センターとクエリーのマイグレーション .....	139
5.3.1 要件 .....	139
5.3.2 ソリューション .....	140
5.3.3 このソリューションに使用する構成アプリケーション .....	140
5.3.4 オブジェクト構造 .....	141
5.3.5 マイグレーション・グループ .....	142
5.3.6 パッケージ定義 .....	144
5.3.7 デプロイメント .....	146
5.3.8 デプロイメントに関する考慮事項 .....	147
<b>第6章 ワークフローのマイグレーション .....</b>	<b>149</b>
6.1 ワークフローに対する特定の変更のマイグレーション .....	150
6.1.1 要件 .....	150
6.1.2 ソリューション .....	150
6.1.3 このソリューションで使用する構成アプリケーション .....	151
6.1.4 オブジェクト構造 .....	155
6.1.5 マイグレーション・グループ .....	155
6.1.6 パッケージの定義 .....	156
6.1.7 デプロイメント .....	157
6.1.8 デプロイメントに関する考慮事項 .....	158
6.2 ワークフローとすべての関連コンテンツのマイグレーション .....	158
6.2.1 要件 .....	158
6.2.2 ソリューション .....	159
6.2.3 このソリューションで使用する構成アプリケーション .....	159
6.2.4 オブジェクト構造 .....	160
6.2.5 マイグレーション・グループ .....	160
6.2.6 パッケージの定義 .....	161
6.2.7 デプロイメント .....	162
6.2.8 デプロイメントに関する考慮事項 .....	163
<b>第7章 分類のマイグレーション .....</b>	<b>165</b>
7.1 シナリオ .....	166
7.1.1 要件 .....	166
7.1.2 ソリューション .....	167
7.2 構成アプリケーション .....	167
7.3 オブジェクト構造 .....	169
7.4 マイグレーション・グループ .....	171
7.5 パッケージ定義 .....	171
7.6 デプロイメント .....	173
7.7 デプロイメントに関する考慮事項 .....	173
<b>第8章 サービス・オファリング・コンテンツのマイグレーション .....</b>	<b>175</b>
8.1 サービス・オファリング・シナリオ .....	177

8.2 要件 . . . . .	177
8.3 ソリューション . . . . .	177
8.4 構成アプリケーション . . . . .	180
8.5 オブジェクト構造 . . . . .	182
8.5.1 PMSC_JOBPLAN . . . . .	182
8.5.2 PMSC_OFFERING . . . . .	183
8.5.3 PMSC_DOCINFO . . . . .	184
8.6 マイグレーション・グループ . . . . .	184
8.7 パッケージ定義 . . . . .	185
8.7.1 SQL 定義の定義 . . . . .	187
8.8 デプロイメント . . . . .	188
8.8.1 デプロイメントの考慮事項 . . . . .	188
<b>第 9 章 サービス・カタログのマイグレーション . . . . .</b>	<b>189</b>
9.1 要件 . . . . .	191
9.2 ソリューション . . . . .	191
9.3 構成アプリケーション . . . . .	193
9.3.1 セキュリティー・グループ . . . . .	194
9.3.2 カタログ . . . . .	194
9.4 オブジェクト構造 . . . . .	194
9.5 マイグレーション・グループ . . . . .	201
9.6 パッケージ定義 . . . . .	202
9.6.1 PMSC72_OFFERING . . . . .	202
9.6.2 PMSC72_CATALOG . . . . .	206
9.7 デプロイメント . . . . .	206
9.7.1 デプロイメントに関する考慮事項 . . . . .	207
<b>第 10 章 一般的なトピック . . . . .</b>	<b>211</b>
10.1 統合フレームワークとマイグレーション・マネージャーの比較 . . . . .	212
10.1.1 シナリオ：統合フレームワークまたはマイグレーション・マネージャーの選択 . . . . .	213
10.2 複数言語に関する考慮事項 . . . . .	215
10.3 スナップショットと変更 . . . . .	218
10.4 埋め込み URL . . . . .	222
10.5 クラスター環境に関する考慮事項 . . . . .	223
10.6 開発環境での変更トラッキングと随時レポート . . . . .	224
10.6.1 変更パッケージを使用するタイミング . . . . .	224
10.6.2 変更パッケージを使用した開発アクティビティのトラッキング . . . . .	225
10.6.3 変更ロールの構成 . . . . .	225
10.6.4 変更パッケージ定義 . . . . .	227
10.6.5 変更イベントのトラッキング . . . . .	228
10.6.6 変更イベントの随時レポートの作成 . . . . .	231

10.7 管理モード .....	235
10.8 階層データのマイグレーション .....	236
10.9 マイグレーション・マネージャーでのロギングのセットアップ .....	244
10.10 スタート・センターでの構成の表示 .....	250
<b>第 11 章 トラブルシューティング .....</b>	<b>253</b>
11.1 一般的なマイグレーション・パッケージ・エラー .....	254
11.1.1 インストールの違いによるパッケージ・デプロイメント・エラー .....	254
11.1.2 誤ったソースの指定によるパッケージ・デプロイメント・エラー .....	257
11.1.3 マニフェスト・データとパッケージ・ファイル名の不整合によるパッケージ・デプロイメント・エラー .....	257
11.1.4 オブジェクト構造の誤った組み合わせによるパッケージ・デプロイメント・エラー .....	258
11.1.5 セキュリティー構成のマイグレーション・エラー .....	258
11.1.6 セキュリティー・グループおよびグループ再割り当てのエラー .....	259
11.1.7 無効または非推奨の MAXVARS の影響 .....	260
11.1.8 添付ドキュメントおよびドキュメント・タイプ .....	261
11.2 マイグレーション・パッケージ・エラーの解決方法 .....	263
11.2.1 ロギング・アプリケーションの使用 .....	263
11.2.2 マイグレーション・パッケージの XML の変更 .....	267
11.3 マイグレーション・パッケージ・エラーの防止方法 .....	269
11.3.1 同一の環境での開始 .....	269
11.3.2 マイグレーション・グループとオブジェクト構造の関係の理解 .....	269
11.3.3 個別のパッケージの独立した操作 .....	270
11.3.4 パッケージ・グループの SQL Where 文節の使用 .....	270
<b>付録 A. 追加資料 .....</b>	<b>271</b>
Web 資料の場所 .....	271
Web 資料の使用 .....	272
Web 資料のダウンロードに関するシステム要件 .....	272
Web 資料の使用法 .....	272
<b>関連資料 .....</b>	<b>273</b>
オンライン資料 .....	273
Redbooks の入手方法 .....	273
IBM からの支援 .....	274
<b>索引 .....</b>	<b>275</b>



1-1 データ・ロードおよびマイグレーションの一般的な順序 .....	6
1-2 構成とマイグレーションの相互作用 .....	8
1-3 構成における依存関係 .....	9
1-4 Integrity Checker エラーの評価 .....	11
1-5 パッケージ・デプロイメントにおける検証の役割 .....	12
1-6 バージョン管理を行うためのソース制御モデル .....	14
1-7 マイグレーション・パッケージをパブリッシュするためのソース制御 モデル .....	15
1-8 マイグレーション・モード .....	18
1-9 マイグレーション作業の内訳 .....	26
2-1 DMMAXSERVICE .....	32
2-2 DMLANGUAGE .....	32
2-3 DMMAXMESSAGES .....	32
2-4 DMMAXLOOKUPMAP .....	33
2-5 DMCURRENCY .....	33
2-6 DMSETS .....	33
2-7 DMORGANIZATION .....	33
2-8 DMMAXVARS .....	34
2-9 DMMAXDOMAIN .....	34
2-10 DMMAXOBJECTCFG .....	34
2-11 DMMAXSEQUENCE .....	35
2-12 DMMAXRELATIONSHIP .....	35
2-13 DMMAXINTOBJECT .....	35
2-14 DMCONDITION .....	35
2-15 DMGLCONFIGURE .....	35
2-16 「データ・ディクショナリー」グループ .....	36
2-17 データ・ディクショナリーのマイグレーション・オブジェクト .....	37
2-18 データ・ディクショナリー変更パッケージ .....	38
2-19 データ・ディクショナリー変更パッケージのトラッキング・レコード .....	39
2-20 新規の通貨 .....	40
2-21 既存の組織に対する新規のサイト .....	41
2-22 新規の部品セットおよび企業セット .....	41
2-23 最初のスナップショット・パッケージの Where 文節 .....	42
2-24 新規の条件式 .....	43
2-25 新規のクロスオーバー・ドメイン .....	43
2-26 新規の従業員オブジェクト .....	44
2-27 新規のルックアップ・マップ .....	44
2-28 従業員オブジェクトの新規の関係 .....	45

2-29 担当者オブジェクトの新規の関係	45
2-30 既存の担当者レポート・オブジェクト構造での従業員オブジェクトへの 新規の結合	46
2-31 2番目のデータ・ディクショナリー・スナップショット・ パッケージ	47
2-32 3番目のデータ・ディクショナリー・スナップショット・ パッケージ	48
2-33 クロスオーバー・ドメインおよびテーブル・ドメインの マイグレーションに必要な除外または組み込み	50
3-1 条件オブジェクト構造	56
3-2 ユーザー・オブジェクト構造	56
3-3 セキュリティ・グループ・オブジェクト構造	56
3-4 RBSECURITY01 オブジェクト構造	57
3-5 オブジェクト構造のクエリー	58
3-6 コントロールのプロパティーの操作	65
3-7 DMSOPTION オブジェクト構造	66
3-8 DMCTRLGROUP オブジェクト構造	66
3-9 オブジェクト構造のクエリー	68
3-10 RBGLOBALSE01 マイグレーション・グループ・ツリー	74
3-11 RBGLOBALSE02 マイグレーション・グループ	75
3-12 RBGLOBALSE02 マイグレーション・グループ・ツリー	75
3-13 パッケージ RBGLOBALSEC01PKG のクエリー	76
3-14 RBGLOBALSEC02PKG オブジェクト構造のクエリー	77
3-15 RBAUTHORIZATIONS01 マイグレーション・グループ・ツリー	81
3-16 RBAUTHORIZATIONS02 マイグレーション・グループ・ツリー	81
3-17 パッケージ RBAUTH01PKG のクエリー	83
3-18 パッケージ RBAUTH02PKG のクエリー	83
4-1 エスカレーションと関連アプリケーション	92
4-2 DMACTION オブジェクト構造	92
4-3 DMACTIONGROUP オブジェクト構造	93
4-4 DMPERSON オブジェクト構造	93
4-5 DMPERSONGROUP オブジェクト構造	93
4-6 DMROLE オブジェクト構造	93
4-7 DMCOMMTEMPLATE オブジェクト構造	94
4-8 DMESCALATION オブジェクト構造	94
4-9 BPM マイグレーション・グループ	95
4-10 RESOURCES マイグレーション・グループ	95
4-11 不要な BPM オブジェクト構造および依存関係	96
4-12 不要な RESOURCES 依存関係	96
4-13 エスカレーションのマイグレーションをサポートするオブジェクト 構造およびマイグレーション・グループ	97
4-14 オブジェクト構造の順序	97
4-15 INCCLOSE サンプル・エスカレーション	98

4-16 INCIDENT CLOSE アクション .....	98
4-17 INCCLOSEGRP アクション・グループとそのメンバーの INCIDENT CLOSED アクション .....	99
4-18 INCCLOSE エスカレーションをマイグレーションするための SQL 基準 .....	99
4-19 SLAREVIEW サンプル・エスカレーション .....	101
4-20 SLAREVIEW 通信テンプレート .....	101
4-21 SLAREVIEW 通信テンプレートの SLACONTACT ロール .....	102
4-22 SLACONTACT ロールの詳細 .....	102
4-23 SLAREVIEW エスカレーションをマイグレーションするための SQL 基準 .....	103
4-24 DMCRONTASKDEF オブジェクト構造が含まれるマイグレーション・ グループ .....	106
4-25 ESCALATION クーロン・タスクをマイグレーションするための SQL 基準 .....	107
4-26 ソース環境で最初に作成されたエスカレーション 1001 .....	108
4-27 ターゲット環境にマイグレーションされたエスカレーション 1001 ..	108
4-28 ターゲット環境で直接変更されたエスカレーション 1001 .....	109
4-29 ソース環境で変更されたエスカレーション 1001.....	109
4-30 「置換」処理アクションを使用してマイグレーションした変更済み エスカレーション 1001 .....	110
4-31 「追加変更」処理アクションを使用してマイグレーションした 変更済みエスカレーション 1001 .....	110
4-32 マイグレーションの後にエスカレーションをアクティビ化したときに 検出されたエラー .....	112
5-1 このケース・シナリオのオブジェクト構造 .....	118
5-2 APPLICATION および APPSECURITY マイグレーション・グループ ..	120
5-3 新規マイグレーション・グループ MYAPPLICATION .....	121
5-4 WOBASICAPP マイグレーション・パッケージの Where 文節条件 ..	123
5-5 新規のマイグレーション・オブジェクト構造 MYMAXAPPS .....	129
5-6 MYMAXPRESENTATION オブジェクト構造 .....	130
5-7 DATADICTONARY マイグレーション・グループ .....	132
5-8 DATADICTONARY マイグレーション・オブジェクト .....	133
5-9 MYNEWAPP マイグレーション・オブジェクト .....	134
5-10 MYQUERY オブジェクト構造 .....	142
5-11 新規マイグレーション・グループ MYAPPSC .....	144
5-12 システム・メッセージ BMXAA6137W .....	147
6-1 ワークフロー・デザイナーのキャンバス .....	152
6-2 接続線アクションのプロパティー .....	153
6-3 通信テンプレート .....	154
6-4 ロール .....	154
6-5 RB_BPM のオブジェクト構造 .....	155
6-6 RB_BPM グループのマイグレーション・オブジェクト .....	156

6-7 パッケージの Where 文節の SQL クエリー .....	157
6-8 新しいマイグレーション・グループ MYBPM .....	161
7-1 分類の階層の例 .....	168
7-2 分類データのサポート・オブジェクト .....	168
7-3 DMCLASSIFICATION オブジェクト構造 .....	169
7-4 「アクションの選択」メニューの複製オプション .....	169
7-5 オブジェクト構造の「自己参照」オプション .....	170
7-6 関係の名前 .....	170
7-7 MYCLASSIFICATION オブジェクト構造 .....	170
7-8 分類データのマイグレーション・グループ .....	171
7-9 分類マイグレーション SQL ステートメント .....	172
7-10 Where 文節ウィンドウ .....	172
8-1 マイグレーション・マネージャー・アプリケーションへのメニュー・ナビゲーション .....	178
8-2 サービス・テンプレート・オファリング .....	179
8-3 サービス・オファリング・テンプレートのマイグレーション・グループ (Migration Group) .....	179
8-4 「作業標準 (Job Plan)」マイグレーションのためのカスタム Tivoli Service Request Manager クラス .....	183
8-5 PMSC ネストされた作業標準関係 .....	183
8-6 PMSC_OFFERING の専用クラス .....	183
8-7 PMSCOFFERING オブジェクト構造の関係 .....	183
8-8 PMSC_DOCINFO の専用クラス .....	184
8-9 サービス・オファリング・マイグレーション・グループのオブジェクト順序 .....	185
8-10 マイグレーション・パッケージ定義 .....	186
8-11 MAXLAUNCHENTRY が選択された状態の「Where 文節の設定 (Set Where clause)」ダイアログ・ボックス .....	186
9-1 マイグレーション・パッケージ・アプリケーションへのメニュー・ナビゲーション .....	192
9-2 サービス・カタログ・パッケージ定義 .....	193
9-3 PMSC72_CatalogTemplate マイグレーション・グループ .....	193
9-4 DMMAXGROUP オブジェクト構造 .....	195
9-5 PMSCCATALOG オブジェクト構造 .....	195
9-6 DMMMAXLAUNCHENTRY オブジェクト構造 .....	195
9-7 DMSIGOPTION オブジェクト構造 .....	196
9-8 DMSIGOPTFLAG オブジェクト構造 .....	196
9-9 PMSC_JACTION オブジェクト構造 .....	196
9-10 PMSC_JOBPLAN オブジェクト構造 .....	197
9-11 PMSC_COMMODITIES オブジェクト構造 .....	197
9-12 DMWFPROCESS オブジェクト構造 .....	198
9-13 DMTKTEMPLATE オブジェクト構造 .....	198
9-14 PMSC_OFFERING オブジェクト構造 .....	199

9-15 PMSC_DOCINFO オブジェクト構造 .....	199
9-16 DMMAXGROUP オブジェクト構造 .....	199
9-17 DMCLASSIFICATION オブジェクト構造 .....	199
9-18 PMSC_ASSETATTRIBUTE オブジェクト構造 .....	200
9-19 PMSC_AUTOSCRIPT オブジェクト構造 .....	200
9-20 DMACTION オブジェクト構造 .....	200
9-21 DMACTIONGROUP オブジェクト構造 .....	200
9-22 PMSC72_OFFERING および PMSC72_CATALOG マイグレーション・グループ構造 .....	201
9-23 マイグレーション・グループ・アプリケーションの「マイグレーション・グループ順序」フィールド .....	209
10-1 DMMAXAPPS オブジェクト .....	216
10-2 スナップショット・パッケージの定義 .....	218
10-3 変更パッケージの定義 .....	219
10-4 「イベント・トラッキング・レコードの表示 (View Event Tracking Records)」オプション .....	220
10-5 変更パッケージに記録されたイベント .....	221
10-6 「イベント・トラッキング・レコードのリセット (Reset Event Tracking Records)」オプション .....	221
10-7 通信テンプレートの例 .....	222
10-8 DEVELOPER ロール .....	226
10-9 DEVELOP 担当者グループ .....	226
10-10 変更パッケージ MYCHANGES-DD .....	227
10-11 変更パッケージ MYCHANGES-APP .....	228
10-12 パッケージ MYCHANGES-DD のイベント .....	229
10-13 パッケージ MYCHANGES-APP のイベント .....	229
10-14 レポート用の DMPKGEVENTS オブジェクト構造 .....	231
10-15 レポート・オブジェクト構造用のアプリケーション .....	232
10-16 DMPKGEVENTS のレポート・オブジェクト・セキュリティー ..	232
10-17 MM_ALL_EVENT クエリー・ベース・レポートの 「スタイル (Style)」タブ .....	233
10-18 QBR 用の MM_ALL_EVENTS 選択済みフィールド .....	234
10-19 変更パッケージ・リスト・トラッキング・イベント .....	234
10-20 MM_ALL_EVENT クエリー・ベース・レポートの出力 .....	235
10-21 ロケーション階層の例 .....	237
10-22 MXOPERLOC オブジェクト構造 .....	238
10-23 ロケーションの CHILDREN 関係 .....	239
10-24 新しい自己参照関係 .....	240
10-25 マイグレーション・サービスの新しいオブジェクト構造 .....	241
10-26 新しいオブジェクト構造を移行するための新しいマイグレーション・ パッケージ .....	242
10-27 限定されたマイグレーション・パッケージで複数のグループに対して Where 文節を設定する例 .....	243

10-28 ロケーション階層のマイグレーションの例 .....	244
10-29 ルート・ロガーの管理 .....	245
10-30 ルート・ロガー・ダイアログ・ボックス .....	245
10-31 ユーザーによって定義されたルート・ロギング・フォルダー .....	246
10-32 「アベンダーの管理 (Manage Appenders)」ダイアログ・ボックス .....	247
10-33 新しいアベンダーの例 .....	248
10-34 アベンダーの設定 .....	249
10-35 サンプル・エラー・ログ .....	250
10-36 ビジネス・オブジェクト、ワークフロー・プロセス、および セキュリティー・グループを表示するサンプルのスタート・センター .....	252
11-1 システム情報へのアクセス .....	254
11-2 「システム情報 (System Information)」画面 .....	255
11-3 「システム情報 (System Information)」画面 .....	256
11-4 パッケージ・マニフェストの XML .....	257
11-5 ログ・ファイルの読み取りとエラーの発見 .....	264
11-6 エラーのある構成ファイルの特定 .....	265
11-7 XML ファイルの確認 .....	265
11-8 マイグレーション・オブジェクトの定義がないパッケージ定義構造 .....	266
11-9 オブジェクト構造のマイグレーション・グループ .....	267
11-10 エラーが発生したキー .....	268
11-11 SQL Where 文節の例 .....	270

# 表

1-1 ツールおよびユースケース .....	5
1-2 改訂履歴の記録 .....	20
1-3 環境、製品、および改訂情報を記録する列 .....	21
1-4 マイグレーション・タスク .....	22
3-1 新規条件値 .....	53
3-2 新規セキュリティー・グループの値 .....	54
3-3 新規オブジェクト制限 .....	54
3-4 新規ユーザーの値 .....	55
3-5 新規マイグレーショングループ .....	57
3-6 RBSECPKG01 マイグレーション・パッケージの値 .....	58
3-7 新規条件 値 .....	63
3-8 新規セキュリティー・グループの値 .....	63
3-9 新規 SIGOPTION .....	64
3-10 「実際の開始 (Actual Start)」フィールドの新規コントロール・ セキュリティー条件 .....	65
3-11 RBUIPKG01 マイグレーション・パッケージの値 .....	67
3-12 RBOOKCOND2 .....	72
3-13 オブジェクト制限 .....	72
3-14 DMSECURITYRESTRICT オブジェクト構造 .....	73
3-15 RBGLOBALSE01 マイグレーション・グループ .....	74
3-16 RBGLOBALSE02 マイグレーション・グループ .....	74
3-17 RBGLOBALSEC01PKG マイグレーション・パッケージの値 .....	75
3-18 RBGLOBALSEC02PKG マイグレーション・パッケージの値 .....	76
3-19 DMAAPPLICATIONAUTH オブジェクト構造 .....	80
3-20 RBAUTHORIZATIONS01 マイグレーション・グループ .....	81
3-21 RBAUTHORIZATIONS02 マイグレーション・グループ .....	81
3-22 RBAUTH01PKG マイグレーション・パッケージの値 .....	82
3-23 RBAUTH02PKG マイグレーション・パッケージの値 .....	82
5-1 基本的なアプリケーション・パッケージに対する Where 文節定義の 設定 .....	122
5-2 MYTESTS-DD パッケージの Where 文節定義 .....	135
5-3 MYTESTS-APP パッケージの Where 文節定義 .....	136
5-4 MYTESTS-SEC パッケージの Where 文節定義 .....	137
5-5 MYSC-DD パッケージの Where 文節定義 .....	144
5-6 MYTESTS-APP パッケージの Where 文節定義 .....	146
6-1 MYTESTS-WF パッケージの Where 文節の条件 .....	162
7-1 分類の属性の例 .....	166

8-1 構成アプリケーション .....	180
9-1 構成アプリケーション .....	194
9-2 すぐに使用可能な PMSC72_OFFERING の Where 文節 .....	203
9-3 すぐに使用可能な PMSC72_CATALOG の Where 文節 .....	206
10-1 統合フレームワークとマイグレーション・マネージャーの比較 .....	212
10-2 プレゼンテーション・エレメントおよび対応するマイグレーション・ マネージャー・オブジェクト構造 .....	217
10-3 スナップショット・パッケージ内のデータ・ディクショナリー・ オブジェクト用の Where 文節 .....	230
10-4 スナップショット・パッケージ内のアプリケーション・オブジェクト用の Where 文節 .....	230

# 例

8-1 キャプションの例 .....	187
8-2 キャプションの例 .....	187
8-3 キャプションの例 .....	187
11-1 ログ・ファイル .....	263
11-2 元のエラー .....	265



# 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権（特許出願中のものを含む）を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒242-8502 神奈川県大和市下鶴間1623番14号 日本アイ・ビー・エム株式会社  
法務・知的財産・知的財産権ライセンス部外

**以下の保証は、国または地域の法律に沿わない場合は、適用されません。** IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態で提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは默示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、隨時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものとします。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

## 著作権使用許諾：

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

## 商標

IBM、IBM ロゴおよび ibm.com は、世界の多くの国で登録された International Business Machines Corp. の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

ITIL は英國 Office of Government Commerce の登録商標および共同体登録商標であって、米国特許商標庁にて登録されています。

IT Infrastructure Library は英國 Office of Government Commerce の一部である the Central Computer and Telecommunications Agency の登録商標です。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。

# 前書き

マイグレーション・マネージャーを利用すると、ある製品環境から別の製品環境に構成コンテンツをマイグレーションできます。一般的な利用法として、Tivoli's process automation engine (Tivoli/Maximo Base Service) とそのアプリケーション (IBM Tivoli Change and Configuration Management Database (CCMDB)、IBM Tivoli Service Request Manager など) の構成コンテンツを、開発環境からテスト環境、さらに実稼働環境にマイグレーションすることができます。マイグレーションの目的は、実稼働環境がエンド・ユーザーのニーズを完全に満たせるようにすることです。

この IBM Redbooks 資料では、マイグレーション・マネージャーを使用した最も一般的なマイグレーション・ユースケースを扱います。もちろん、これらのユースケースは、マイグレーション・マネージャーで実現できるマイグレーション・シナリオのごく一部に過ぎませんが、マイグレーション・マネージャーの代表的な機能を示す目的で選ばれています。

本書ではこれらのユースケースに加え、マイグレーション戦略を紹介とともに、マイグレーション・マネージャーの使用時に発生する可能性のあるマイグレーション問題のトラブルシューティングを包括的に扱う章を用意しています。

他の章を読む前に、まず『第1章 マイグレーション戦略』を読むことを強くお勧めします。この章を読むことで、本書で扱うすべてのマイグレーション・シナリオの理解に役立つ基礎知識が得られます。

IT スペシャリストや IT 設計者がマイグレーション・マネージャーを使用して、ある製品環境から別の製品環境に構成コンテンツをマイグレーションする際には、本書を参照してください。

## 本書の執筆に携わったチームについて

本書は、International Technical Support Organization Raleigh Center に世界各地から参加しているスペシャリストのチームによって作成されました。



**Bradley Downing** は現在、IBM の Tivoli Advanced Technology Group - ISM Lab Services に勤務しています。彼のキャリアは 15 年前、カリフォルニアの小規模なインテグレーター会社で「Maximo」製品を担当することから始まりました。彼は 2001 年に MRO Software に入社し、2006 年の吸収合併により IBM の一員となりました。この間、彼はさまざまな業種のクライアントを扱う業務に専念し、特にレポート作成、Mobile Maximo、統合フレームワークの使用、および最近ではマイグレーション・マネージャー・モジュールを扱っています。彼は経営管理の MBA を取得しています。



**Burak Bolek** は、IS Bank で主任開発者および管理者として勤務しています。彼は Maximo に関して 3 年の経験があります。また、Java 開発者としてさまざまなクライアントの実装にも携わっています。彼は Tivoli's process automation engine (Tivoli/Maximo Base Service) と ITIL の認定を受けています。



**David A Gutierrez** は IBM Global Business Services の技術コンサルタントであり、Maximo v7.x のデプロイメント・プロフェッショナルとして IBM 認定を受けています。彼には、IBM の内外を問わず民間セクターと公共セクターの両方で、パッケージ・ソリューションの実装、デプロイ、構成、マイグレーション、および管理の経験が 6 年あります。David は 25 年以上にわたり IT プロフェッショナルとして、システムやネットワーキングの技術サポートをクライアントに提供しています。David はペルーのリマ大学で産業エンジニアリングの理学士号を取得し、現在はカリフォルニア州サンディエゴに住んでいます。



**John Dibble** は MRO から IBM Tivoli に参加しました。MRO では MainControl ITAM 製品と Maximo EAM の統合（最終的には Maximo 6 のリリース）を指揮しました。IBM が MRO を吸収合併したことで、彼は新しいテクノロジーの応用を目的に IBM Global Technology Services グループに異動しました。彼は現在、Tivoli's process automation engine (Tivoli/Maximo Base Service)、CCMDB、TSRM、および TAMIT ソリューションの主任設計者として活躍しています。



**Kylie Cameron** は、オーストラリアのシドニーを拠点として、IBM Global Business Services の Maximo クライアント対応コンサルタントとして働いています。彼女はビジネス情報システムと E-コマースを専攻とする情報通信技術の学士号を取得しています。Kylie は Maximo 認定コンサルタントとして、公共事業と製薬業のプロジェクト経験が 3 年以上あり、これまでに在庫管理 / 調達リーダー、環境マネージャー、リリース・マネージャー、トレーニング・リーダーなどの職務についています。



**Sampath Sriramadhesikan** は IBM Software Group の設計者として、マサチューセッツの IBM Littleton キャンパスで働いています。彼は Tivoli's process automation engine (Tivoli/Maximo Base Service) と Maximo 製品ポートフォリオの主要コンポーネントの設計および開発を指揮しており、特にコンテンツ管理機能とプロモーション機能を重点的に扱っています。彼はソフトウェア開発業界で 17 年以上の経験があり、これまでに製品開発、製品管理、研修などさまざまな職務についています。彼はマイソール大学でコンピューター・サイエンスの理学修士号、ノースイースタン大学で MBA を取得しています。

**Scott Tyrrell** は Fortune 500 にランкиングされる企業で Maximo システム管理者をしており、現在はサービス・プロバイダーと共に TSRM、TAMIT、CCMDB の製品スイートに関する日常的な運用上の問題とアプリケーション開発をサポートしています。元ソフトウェア開発者の Scott は ITIL の認定を受けており、ここ数年は ITIL のプロセス開発と実装に携わっています。彼はクリスチャン・ブラザーズ大学で電気工学の理学士号、リトルロックにあるアーカンソー大学で MBA を取得しています。



**Thadeu de Russo e Carmo** は、ブラジルのサンパウロを拠点とする IBM Brazil Software Lab で、IBM Software Group のスタッフ・ソフトウェア・エンジニアとして勤務しています。彼はコンピューター・サイエンスの理学士号を取得しており、現在はサンパウロ大学 (USP) で大学院生として分散システム分野を研究しています。彼は IBM Maximo Asset Management の開発に 4 年間従事しており、IBM Maximo for Nuclear Power Plants、IBM Maximo for Life Sciences、IBM Maximo Mobile など多数のアドオンや業種別ソリューションの開発に開発者リーダーとして携わっています。また、Java Enterprise の開発経験が 6 年以上ある彼は Java 認定開発者および設計者でもあり、Foundations of Tivoli Process Automation Engine の認定も受けています。



**Vasfi Gucer** は、International Technical Support Organization Austin Center でプロジェクト・リーダーをしています。彼は 1999 年 1 月から ITSO に勤務しています。システム管理、ネットワーキング・ハードウェア、メインフレームや分散型プラットフォーム上のソフトウェアなどの分野で 12 年以上の経験があります。彼はアメリカでシステム設計者として、さまざまな IBM Tivoli クライアント・プロジェクトを担当しています。また、幅広い執筆活動のほか、Tivoli ソフトウェアに関する IBM クラスの研修も世界中で行っています。Vasfi はまた、IBM 認定の上級 IT スペシャリスト、PMP、および ITIL エキスパートでもあります。

このプロジェクトにご協力いただいた以下の方々に感謝します。

Tamikia Barrow、David Bennin、Emma Jacobs、Stephen Smith、Margaret Ticknor、  
Debbie Willmschen

**International Technical Support Organization**

Scott Dickerson、Tom Ellingwood、Adel Fahmy、Stephen Ridgill  
**IBM USA**

Marieli Alves de Lucena  
**IBM Brazil**

Thomas Gehrke  
**IBM Germany**

Andrzej Wieclaw  
**Framsteg AB**

## 執筆にご協力ください

スキルを生かしてキャリアを伸ばし、それと同時に執筆に協力できる機会がここにあります。ITSO の実習プログラムに参加しませんか。ご自分の専門分野に関する資料の作成を手伝いながら、最先端のテクノロジーを実地に体験することができます。技術担当者や関係者との人脈を開拓しながら、製品をお客様にとってより使いやすいものにして、お客様の満足度を向上させるために協力してください。実習期間は 2 週間から 6 週間で、直接参加することも、ご自宅からリモート実習生として参加することもできます。

実習プログラムについて詳しくは、次の URL で実習に関する索引を参照して、オンラインで申し込んでください。

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## ご意見をお寄せください

以下は英語のみの対応となります。皆様の貴重なご意見をお寄せください。

IBM では、本書ができる限り皆様のお役に立つものになるように努力しています。本書または他の IBM Redbooks 資料についてのご意見を以下のいずれかの方法でお寄せください。

- ▶ 以下の URL からオンラインの「お問い合わせ (Contact us)」Redbooks レビュー・フォームを使用する。

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ 以下の宛先に E メールで送信する。

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- ▶ 以下の住所に郵送する。

IBM Corporation, International Technical Support Organization  
Dept. HYTD Mail Station P099  
2455 South Road  
Poughkeepsie, NY 12601-5400

## IBM Redbooks 関連情報

- ▶ Facebook で探す。  
<http://www.facebook.com/IBMRibooks>
- ▶ Twitter でフォローする。  
<http://twitter.com/ibmredbooks>
- ▶ LinkedIn で探す。  
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ IBM Redbooks の週刊ニュースレターで新しい Redbooks 資料、実習、ワーク ショップを調べる。  
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ RSS フィードで最近の Redbooks 資料に関する最新情報を入手する。  
<http://www.redbooks.ibm.com/rss.html>



# マイグレーション戦略

この章では、クライアント環境全体で Tivoli システム構成を正常にマイグレーションするために、マイグレーション・マネージャーが他のツールと連携してどのように機能するかについて説明します。通常、このような環境には、開発環境(単体テスト)、テスト環境(統合テスト)、ステージ環境(最終統合およびパフォーマンス・テスト)、および実稼働環境があります。

マイグレーション・マネージャーは、Tivoli's process automation engine (Tivoli/Maximo Base Service) のコンポーネントです。これは、開発環境から後続の環境(ステージ環境、ユーザー受け入れテスト環境、および実稼働環境など)にシステム構成をプロモートするために使用する、一連のツールです。

システム構成とは、製品環境において、アプリケーション機能を実現し、アプリケーション動作を制御する、メタデータ・セットを指します。例えば、サービス要求アプリケーションのユーザー・インターフェースは、アプリケーション・プレゼンテーション・メタデータを通じて有効化およびレンダリングされます。

実稼働環境を実際に稼働する前に、システム構成をマイグレーションする際には、マイグレーション・マネージャーを重点的に使用します。ただし、実稼働環境の稼働後も、システム構成に対する変更をマイグレーションする際に使用します。本書では、この両方のシナリオにおいて、マイグレーション・マネージャーが担う役割について説明します。

通常、マイグレーション対象のシステム構成が生成される製品環境を、ソース環境と呼びます。また、システム構成を取り込む製品環境のことを、通常はターゲット環境と呼びます。

この章には以下のセクションがあります。

- ▶ 3 ページの『1.1 要件』
- ▶ 3 ページの『1.2 マイグレーションをサポートするツールの選択』
- ▶ 8 ページの『1.3 コンテンツ戦略』
- ▶ 16 ページの『1.4 マイグレーション方法』
- ▶ 18 ページの『1.5 マイグレーション計画』
- ▶ 27 ページの『1.6 マイグレーション・マネージャーに関する参考資料』

## 1.1 要件

システム構成を開発環境から実稼働環境にマイグレーションする際のクライアント要件を以下に示します。

- ▶ マイグレーションは、システム・ランドスケープを構成するすべての環境において反復可能かつ効率的である必要があります。
- ▶ マイグレーションは、実稼働環境の稼働前も稼働後も活用可能である永続的なものである必要があります。
- ▶ マイグレーションは、すべてのタイプのシステム構成を、同じ方法で識別、収集、パッケージ化、配布、およびデプロイできる安定的なものである必要があります。
- ▶ マイグレーションでは、ターゲット環境を常に使用可能な完全な状態に維持する必要があります。システム構成の完全性に問題が生じてはなりません。
- ▶ マイグレーションでは、ターゲット環境のシステム・ダウン時間を最小限に抑えるか、完全に排除する必要があります。

テスト・データ（開発環境での単体テストに使用するデータなど）はプロモート対象ではありません。Java ソースの形式のソース・コードまたはコンパイル済み Java クラス・ファイルは、従来のソース制御システムを使用することで個別に管理できます。

## 1.2 マイグレーションをサポートするツールの選択

前述の要件を満たすために、製品の展開を担当する設計者は、企業内で Tivoli に求められている動作および機能を評価する必要があります。Tivoli は、企業との間でデータ・レコードの交換を行います。マイグレーション中および稼働後にこのデータをどのように保守するかを定義する必要があります。これは、ある製品環境から別の環境にシステムを構築する際の方法に影響します。例えば、担当者レコードは決してマイグレーションしないとします。代わりに、それぞれの製品環境で、ディレクトリーの担当者レコードを同期化します。あるいは、レガシー・システムから製品環境に担当者レコードが常にロードされるようにします。これは、マイグレーションの計画、範囲、および作業内容に影響します。

ツールの観点から見ると、主にマイグレーション・マネージャーは、開発環境における構成およびカスタマイズのアクティビティーの結果として生成されるシステム構成メタデータをマイグレーションする役割を担います。また、マイグレーション・マネージャーでは、アプリケーション・コンテンツのマイグレーションも可能です。ただし、マイグレーション・マネージャーを使用して

そのようなアプリケーション・コンテンツをマイグレーションする必要があるかどうかは、以下に示す基準に従って判断する必要があります。

- ▶ アプリケーションが静的であるかどうか。  
静的数据が変更されることはありません。一度のみセットアップされ、それ以降は、すべての製品環境において同じ静的フォームで使用されます。
- ▶ アプリケーション・コンテンツが開発環境で作成されるかどうか。  
開発環境で作成されたアプリケーション・コンテンツは、プロモートが必要な場合があります。例えば、分類の階層が作業標準アプリケーションをサポートする場合です。
- ▶ アプリケーション・コンテンツをマスター・データまたはトランザクション・データとして見なすことができるかどうか。  
マスター・データの例としては、担当者、部品マスター、および企業レコードなどがあります。トランザクション・データの例としては、サービス要求、インシデント、注文書、および請求書などがあります。テスト・データを除き、これらのデータは開発環境ではセットアップされません。

アプリケーション・コンテンツが静的であり、開発環境で作成され、マスター・データでもトランザクション・データでもない場合、そのようなコンテンツは、マイグレーション・マネージャーを使用してターゲット環境にマイグレーションする対象と考えることができます。

### 1.2.1 データ・ロード・ツール

テスト環境や実稼働環境などのターゲット環境を、データ・ロードを通じて準備する観点から、マイグレーション・マネージャーの他に多数の IBM ツールが提供されています。これらのデータ・ロード・ツールは、特定のデータ・ロード機能を提供することを目的としています。主なツールには、以下のものがあります。

- ▶ Tivoli's process automation engine (Tivoli/Maximo Base Service) 統合フレームワーク
- ▶ Tivoli Integration Composer
- ▶ Tivoli Directory Integrator

場合によっては、ビジネス・アプリケーションにアプリケーション固有のデータ・ロード機能が備わっていることがあります。例えば、アプリケーション・デザイナーには、アプリケーションおよびシステム・プレゼンテーションをサポートするエクスポート機能およびインポート機能が備わっています。

ツールの選択は、具体的なデータ・ロード要件に応じて異なります。表 1-1 に、特定のツールとそのユースケースを示します。

表 1-1 ツールおよびユースケース

データ・ロードのタイプ	Tivoli's process automation engine (Tivoli/Maximo Base Service) マイグレーション・マネージャー	Tivoli's process automation engine (Tivoli/Maximo Base Service) 統合フレームワーク	Tivoli Integration Composer	Tivoli Directory Integrator
検出ツールからのデータのバッチ・ロード			X	
マスター・データとトランザクション・データ、またはレガシー・データのバッチ・ロード		X		
外部システムとの、メッセージングに基づいた統合		X		X
Tivoli's process automation engine (Tivoli/Maximo Base Service) をベースとする製品環境間でのシステム構成のプロモーション	X			

ここで示すツールやそのユースケースから、後続の環境にデータをロードする際に使用するツールは、マイグレーション・マネージャーだけではないことが分かります。データ・ロードのニーズに応じた適切なツールを選択することで、実稼働環境の稼働を滞りなく行うことができます。

データ・ロード・ツールおよび機能の詳細については、ベスト・プラクティスを紹介した以下の資料を参照してください。

## 1.2.2 データ・ロードの順序

ほとんどの実稼働環境は、最初は複数のデータ・ロード・ツールを組み合わせて使用して構築されます。6ページの図 1-1 に、マイグレーション・マネージャーを使用したマイグレーションなど、初期の一般的なデータ・ロード作業を示します。

Integration Framework	Chart of Accounts, Currency, Classification	Locations	ERP data – Companies (vendors)	Legacy items and assets				
Migration Manager	Data Dictionary (including Org, Domains)	Data Dictionary (Domains 2 <sup>nd</sup> pass)	Applications	Global restrictions	Start Centers and queries	Application security and person groups	Service catalog	BPM
Other data loaders	LDAP-based person /user /groups	ITIC-based discovery data						
Manual seeding	Org/Site/Currency/GL structure	Location System	Application Designer system XML					

図 1-1 データ・ロードおよびマイグレーションの一般的な順序

以下に、この一般的なロード / マイグレーションの順序の詳細説明を示します。

1. 組織名と通貨を手動で指定します。
2. GL 勘定科目構造(セグメント)を手動でセットアップします。
3. 定義されている各組織について、デフォルトの勘定科目一覧を手動で作成します。
4. ロケーション・システムを手動で定義します。
5. 統合フレームワークのオブジェクト構造を使用して、地域などの物理的なロケーション階層をロードします。

6. Integrity Checker を実行して、エラーを探します。
7. エラーを手動で修正するか、Integrity Checker を修復モードで実行して修正します。
8. 手動による手順を完了すると、エンタープライズ LDAP ディレクトリー・サーバーが製品と同期化されます(クライアントがディレクトリーを使用している場合)。担当者、ユーザー・グループ、およびセキュリティー・グループをロードできます。
9. LDAP が稼働していない場合、担当者、担当者グループ、およびセキュリティー・グループ(従業員および監督者など)の各構成をマイグレーションします(管理レポート)。
10. ドメインなどのデータ・ディクショナリーをマイグレーションします。
11. クロスオーバー・ドメインおよびテーブル・ドメインをマイグレーションして、データ・ディクショナリーのマイグレーションを完了します。
12. 製品環境に Tivoli Service Request Manager (TSRM) が存在する場合、サービス・カタログのオブジェクト構造のマイグレーションを実行します。この時点で、構造的なマイグレーションは完了します。
13. 会計勘定科目一覧をロードします。
14. 通貨および換算レートをロードします。
15. メニューやルックアップなど、すべてのアプリケーション構成をマイグレーションします。
16. 署名オプションなどのアプリケーション・セキュリティー構成をマイグレーションします。
17. スタート・センターおよびクエリーをマイグレーションします。
18. Tivoli Service Request Manager サービス・カタログ・テンプレートをマイグレーションします。
19. Tivoli Service Request Manager サービス・カタログの単一サービス構成をマイグレーションします。
20. ワークフローおよびその関連ロール、アクション、通信テンプレートをマイグレーションします。
21. Integrity Checker を実行してエラー探し、見つかったエラーを修正します。

この順序には、データ・ロードおよびマイグレーションにおいて重要な手順が組み込まれています。この順序に沿った手順を実行する際、多数のツールを活用することになります。このようなツールがあれば、実稼働環境の準備を迅速に行うことができます。計画段階においてこれらの順序を考慮することは、マイグレーション作業全体を成功させる上で必要不可欠です。ツール、アプリケーション、およびコンテンツに関する事前知識もなく、その場しのぎで作業を実行すると、マイグレーション作業の効率性が失われ、顧客の満足度の低下にもつながります。

## 1.3 コンテンツ戦略

マイグレーションは、構成アクティビティーの結果として付随します。構成アクティビティーでは、さまざまな構成アプリケーションを使用して、製品データベース内にコンテンツを作成します。このようなコンテンツには、IBMにより製品とともに提供されるものもあれば、クライアントの開発環境で作成されるものもあります。これらの構成について詳細に把握し、制御することは、結果として付随するマイグレーション作業を効率的に行う上で非常に重要です。8ページの図 1-2 に、構成とマイグレーションの間の相互作用について示します（いずれの場合も、基礎となる管理対象作成物はコンテンツです）。

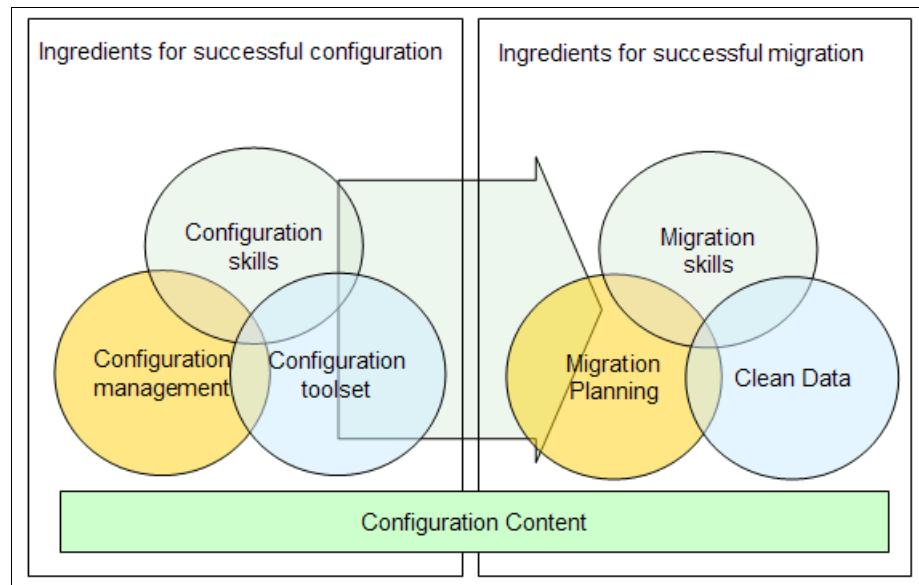


図 1-2 構成とマイグレーションの相互作用

構成の観点から見ると、クライアントの開発環境において構成を効率的に完了するには、構成スキル、構成管理、および構成ツール・セットの適切な使用が不可欠です。構成管理の実施手法が向上すれば、マイグレーション計画もより適切なものになります。構成アプリケーションおよびツールを一貫して使用することで、データの明確性を確保でき、マイグレーション作業の効率化につながります。

### 1.3.1 構成コンテンツ間の依存関係

完全に独立した製品構成というものは、ほとんど存在しません。多くの製品構成には、他の構成に対する依存関係があります。構成コンテンツを製品データベースにロードする際に、製品検証を行うことで、これらの依存関係を強制適

用します。構成間の依存関係と、それを強制適用する検証は、マイグレーション作業に影響します。9ページの図 1-3 に、膨大な量の製品機能を実行する構成における、多数の依存関係を示します。

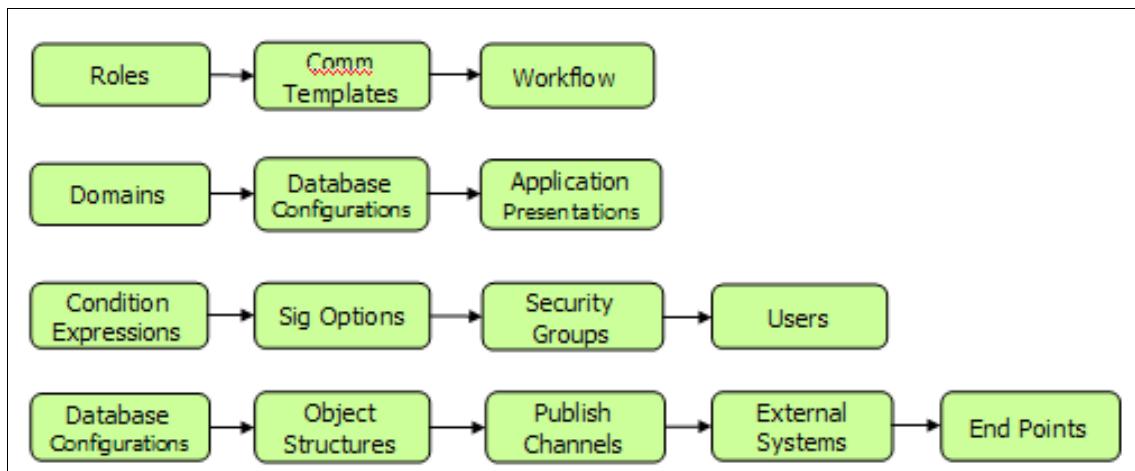


図 1-3 構成における依存関係

図 1-3 では、ワークフロー・プロセス構成が通信テンプレートおよびロールに依存しています。アプリケーション・プレゼンテーションはビジネス・オブジェクトに依存し、一方このオブジェクトの属性は、ドメインの形式で値リストに関連付けられています。セキュリティー・グループは署名オプションに関連付けられています。一方、この署名オプションは、条件式に基づいて、セキュリティー・グループに条件付きで適用することができます。統合フレームワークを使用して識別される外部システムはパブリッシュ・チャネルに関連付けられており、このパブリッシュ・チャネルはオブジェクト構造およびビジネス・オブジェクトによりサポートされています。

構成間に複雑な依存関係が存在する場合、マイグレーションの順序付けが非常に重要となります。

### 1.3.2 ベスト・プラクティス・コンテンツ

Tivoli の方針で、製品にはベスト・プラクティス・コンテンツが用意されています。これにより、クライアントは特定の構成やカスタマイズを行うことなく、製品を迅速に準備でき、すぐに使用を開始できます。この一例として、すぐに使用可能な Tivoli Service Request Manager があります。これによって、一般的な IT Infrastructure Library (ITIL) のシナリオをカバーする、事前構成オファリングの IBMPMSC 組織、PMSC サイト、および PMSC セットの作成をインストール時に含めることができます。

最新のクライアントでは、これらがロードされており、選択するだけでオーフアーリングを完全にカスタマイズできます。これにより、マイグレーションを行う際に、このコンテンツをどのように扱えばよいかという問題が生じますが、選択肢はいくつかしかありません。事前構成のものを除外してマイグレーション・パッケージを作成するか、現在提供されているツールを使用して事前構成のものをクライアント組織に追加するか、削除するという選択肢のみです。

**注:** 前述の例はインストール・プロセスの一部であるため、インストール段階で特に削除しない限り、通常はターゲット環境に適用されます。

別の例としては、Intellectual Capital (iCAP) を使用して、ITIL ガイドラインに準拠するベスト・プラクティス・ビジネス・プロセス・コンテンツ (ワークフロー・ベース) を提供することが挙げられます。これはすべての環境でインストール時にロードされるため、環境においてプロモートが必要となるのは、インストール以降のクライアント変更のみです。

### 1.3.3 構成コンテンツの完全性

このセクションでは、IBM 提供のツールを使用して、製品データベースにおいて確保する必要がある構成の完全性について説明します。

多数の構成の完全性を判別する際に使用する主要なツールは、Integrity Checker です。Integrity Checker ツールは製品と共に提供されており、コンソールのコマンド行から起動できます。このツールは、データの完全性に関する警告およびエラーを含む詳細なログを作成します。すべての製品環境で Integrity Checker を定期的に実行して、基礎となる製品データベースの完全性を継続的にモニターすることが重要です。

ベスト・プラクティスは、Integrity Checker によりレポートされたエラーをスプレッドシート形式で取り込み、エラー・メッセージを検討し、それらのエラーの解決策を特定することです。このスプレッドシートを準備することで、エラーがマイグレーションに与える影響を評価することができます。図 1-4 に、エラー・メッセージを収集し、解決策も提示したサンプル・スプレッドシートを示します。

Error message	Reason for error	Impacts migration?	TACTICAL FIX 1	TACTICAL FIX 2
			Migration Remediation	Integrity Checker Remediation
BMXAA0443E - ERROR -- BMXAA0465E - The following Maximo indexes are missing from the database: Mon Nov 03 08:39:34 CST 2008 PLUSGCFW_NDX4 Mon Nov 03 08:39:34 CST 2008 PLUSGREGULAT_NDX4 Mon Nov 03 08:39:34 CST 2008	MAXSYSINDEXES table contains two entries, but underlying database does not	YES, if you attempt to migrate the PLUS* tables, the missing indexes will cause package deployment failures.	Do not carry PLUS* tables in your migration packages; use SQL conditions to exclude such tables for DMMAXOBJECTCFG	Correct error as per Integrity Checker troubleshooting/correction steps.
BMXAA0443E - ERROR -- BMXAA0467E - The database has a different UNIQUE property for the following maximo indexes: Mon Nov 03 08:39:34 CST 2008 Table MXIN_INTER_TRANS Keys TRANSID Mon Nov 03 08:39:34 CST 2008 Table MXOUT_INTER_TRANS Keys TRANSID Mon Nov 03 08:39:34 CST 2008	The uniqueness of the index in MAXSYSINDEXES is different from the uniqueness defined at the database level.	MAYBE, only in an update scenario where the same indexes already exist in target	For these two errors, use SQL conditions to exclude MXIN_INTER_TRAN S and MXOUT_INTER_TRA NS tables for DMMAXOBJECTCFG	Correct error as per Integrity Checker troubleshooting/correction steps.
BMXAA0443E - ERROR -- BMXAA0470E - Non-sequential primary key sequence on table(s): Mon Nov 03 08:40:09 CST 2008 DMPACKAGEDEF Mon Nov 03 08:40:09 CST 2008 MAXPROCCOLS Mon Nov 03 08:40:09 CST 2008	PRIMARYKEYCOLSEQ values in MAXATTRIBUTE table for the primary key columns are not in sequence	YES, if you attempt to migrate the MAXOBJECTCFG data for these two tables, it is highly likely package deployment failures will occur	Correct the error in both source and target environments such that the primary key col seq values are in sequence (1, 2, 3, etc)	Correct error as per Integrity Checker troubleshooting/correction steps.

図 1-4 Integrity Checker エラーの評価

図 1-4において、「エラー・メッセージ(Error Message)」列では、Integrity Checker ログ・ファイルから取得した Integrity Checker エラーを複製しています。「エラーの理由(Reason for Error)」列では、エラー・メッセージ番号と Integrity Checker エラー・テーブルを関連付けることで、Integrity Checker 製品資料を複製しています。「マイグレーションへの影響(Impacts Migration)」列は、特定のエラーがマイグレーション・マネージャーのアクティビティに影響するかどうかを示しています。「戦術的修正 1(TACTICAL FIX 1)」列は、マイグレーション・マネージャーを制御して、エラーによるマイグレーション障害を回避できるかどうかを示しています。「戦術的修正 2(TACTICAL FIX 2)」列は、その解決策により基礎となるデータベース内の完全性エラーを直接修正し、マイグレーションを実行できるかどうかを示しています。多くの場合、データベース内で完全性エラーを修正してから、マイグレーションを実行する必要があります。

Integrity Checker ツールの説明と使用法については、以下より参照できる詳細な製品資料を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v32r1/topic/com.ibm.srm.doc\\_721/reference/6\\_to\\_7\\_upgrade.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v32r1/topic/com.ibm.srm.doc_721/reference/6_to_7_upgrade.pdf)

### 1.3.4 コンテンツの検証

このセクションでは、検証の重要性と、マイグレーション作業への影響について説明します。

マイグレーション・パッケージの構成コンテンツは、マイグレーション・マネージャーを使用して製品環境にデプロイします。このデプロイメントは、マイグレーション・パッケージのコンテンツを抽出し、それらのコンテンツを特定の順序で処理するための、十分に定義された自動的な手順を通じて行います。

コンテンツのタイプによっては、管理者が手動タスクを行うためにこの手順を一時的に停止し、手動タスクが完了したら手順を開始する場合があります。基礎となる製品データベースにデプロイするすべてのコンテンツは、検証対象です。このような検証により、製品データベースの完全性が常に維持されます。この検証を省略することはできません。12 ページの図 1-5 に、デプロイメントにおける検証の役割を示します。

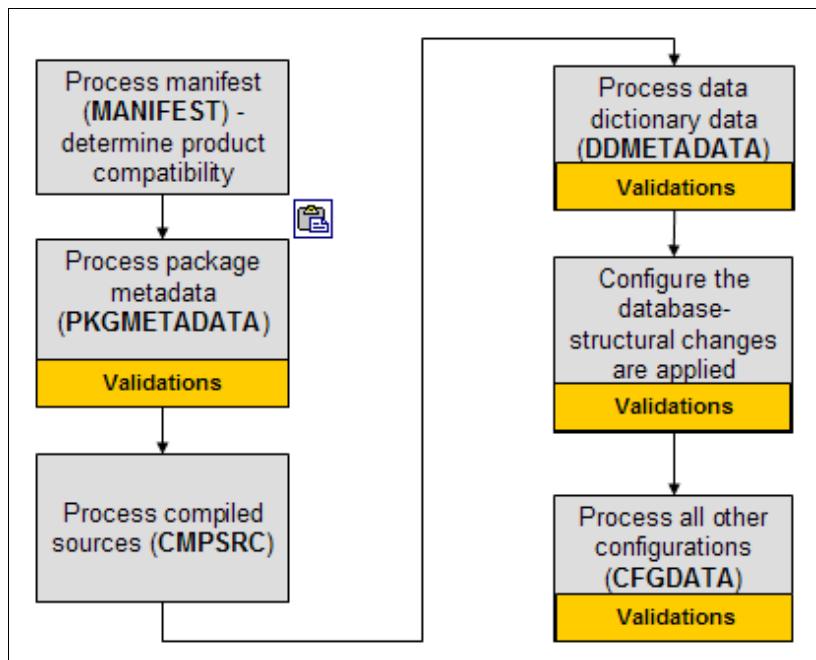


図 1-5 パッケージ・デプロイメントにおける検証の役割

図 1-5 では、重要な検証が、データ・ディクショナリ構成、データベース構成プロセス、およびその他すべての構成（担当者レコード、セキュリティー・グループ・レコード、ワークフロー・プロセスなど）に適用されています。

検証が失敗すると、マイグレーション・マネージャーはデプロイメント・エラーをレポートします。デプロイメント・エラーは、構成アプリケーションが、デプロイメント対象の特定の構成に対してデータ検証ルールを正常に適用できないことを意味します。検証時の障害の結果であるデプロイメント・エラーをレポートする役割は、マイグレーション・マネージャーが担います。

このような検証エラーの根本原因是、以下のいずれか、または以下の組み合わせである場合があります。

1. 構成コンテンツのパッケージ化が不完全である
2. マイグレーションの順序付けが誤っている

3. ソースの製品環境のデータが不正である
4. マイグレーション・マネージャーの制約
5. 製品の問題点

いずれの場合でも、適切な改善処置を実施してエラーを解決し、マイグレーション作業を継続できるようにします。

順序付けの誤りや、不完全なパッケージ化は、製品に対するスキルを向上させることで解決できます。一般に、不正なデータは、データベース・コマンドを使用してデータをデータベースに直接ロードしていることが原因です。この慣例は必ず避けるべきです。また、不正なデータの原因是、製品で提供されるデータベース・スクリプトにある場合もあります。一部の不正なデータは、Integrity Checker を使用することで解決できる場合があります。不正なデータが製品と共に提供されていた場合、IBM はこれを製品の問題点として処理します。マイグレーション・マネージャーの制約が原因である場合は、他のツールを利用して、必要な構成または関連データをロードすることで解決できます。製品の問題点である場合、IBM が標準的な手順に則って対処します。デプロイメント・エラーの性質を理解することは、エラーの解決に非常に役立ちます。

### 1.3.5 コンテンツおよびソース制御システム

構成の管理およびマイグレーションの実行を行う実務担当者やクライアントから頻繁に話題に出されるトピックとして、ソース制御システムの役割があります。Tivoli's process automation engine (Tivoli/Maximo Base Service) には、バージョン管理機能は組み込まれていません。構成は、基礎となる製品データベース内の指定されたテーブルに保存されるだけです(その際、前の値は上書きされます)。構成のバージョンはどのように保守すればよいでしょうか。マイグレーション・マネージャーが担う役割は何でしょうか。

マイグレーション・マネージャーは、特定のソース制御モデルを使用可能にしたり、このモデルに対する指示を出したりすることはありません。ただし、変更管理の観点から見ると、ソース制御モデルを実装したり、既存のソース制御モデルを拡張したりすることは、実務担当者やクライアントにとってマイグレーション作業だけに留まらないメリットがあります。

このセクションでは、2つのソース制御モデルについて説明します。

- ▶ ソース制御による構成のバージョンの管理
- ▶ ソース制御によるマイグレーション・パッケージのパブリッシュ

#### ソース制御による構成のバージョンの管理

製品の一部の構成コンテンツは、チーム・メンバー間で共有されます。このコンテンツに対し、短期間で複数の編集作業が行われる場合があります。構成コンテンツが製品データベースに保管されるとき、構成を保存した最後のチー

ム・メンバーが行った変更のみが、データベースに書き込まれることになります。これは、共有されている製品開発環境における、変更管理に関する重要な課題です。構成コンテンツを前の改訂内容に戻す必要がある状況があるからです。これは、ソース制御モデルを実装することでのみ、対処できます。

このソース制御モデルでは、選択した構成コンテンツが製品環境からファイルとしてエクスポートされ、ソース制御システムにチェックインされます。そのコンテンツに対して変更を行う必要がある実務担当者または開発者は、そのファイルをチェックアウトし、製品環境にインポートし、適切なアプリケーションを使用して変更を実装する必要があります。完了後、構成を再度エクスポートし、ソース制御側に戻します。14 ページの図 1-6 に、このモデルを示します。

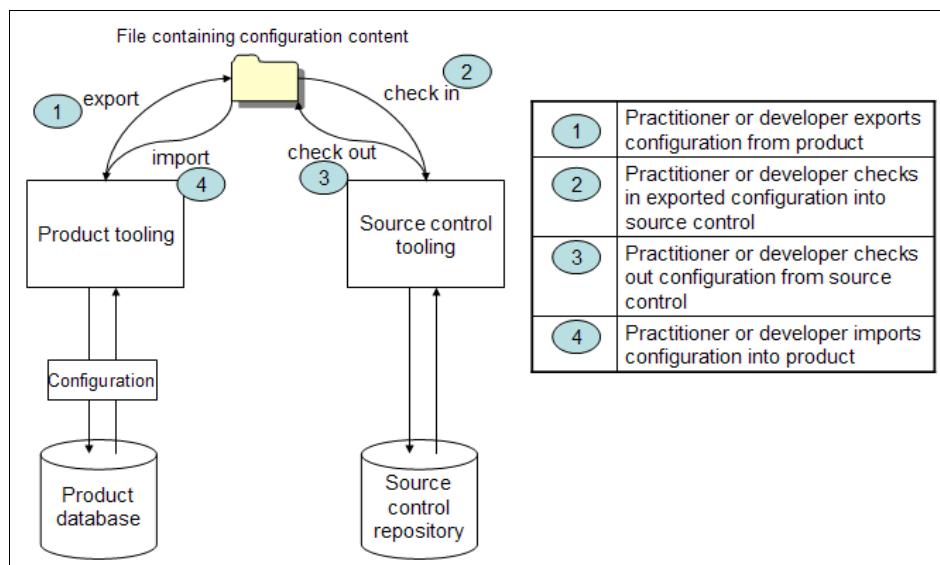


図 1-6 バージョン管理を行うためのソース制御モデル

手間はかかりますが、このモデルを実装することで、変更を追跡することができます、それらの変更について作成者に実装サイクル中の責任を負わせることができます。構成の改訂を管理するこの手法は、全体的なコンテンツ戦略の一部と考えてください。

一般的に、このモデルは共有されているアプリケーション・デザイナー構成 (library.xml、lookups.xml、および menus.xml) に適用されます。これらの 3 つの構成ファイルは、アプリケーション・デザイナーを使用してアクセスおよび管理される共有構成です。

## ソース制御によるマイグレーション・パッケージのパブリッシュ

クライアントによっては、マイグレーション・マネージャーにより提供される圧縮ファイルの形式で、マイグレーション・パッケージ全体をソース制御システムにチェックインする場合もあります。この手法を選択するのは、マイグレーションの実行時に使用する必要があるマイグレーション・パッケージの許可セットを、ソース制御システムのみで保持する場合です。ファイル・システムまたはデータベース配布を利用して、ソース環境からターゲット環境にパッケージを直接配布することはできません。15 ページの図 1-7 に、このモデルを示します。

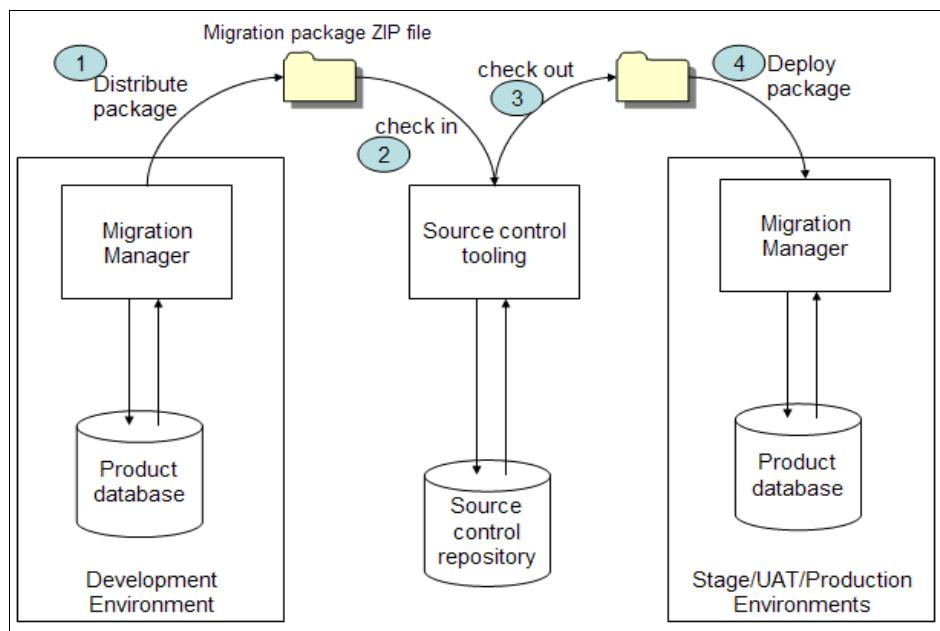


図 1-7 マイグレーション・パッケージをパブリッシュするためのソース制御モデル

この手法では、柔軟性がメリットとなります。この手法を実施する前に、全体的な実装のタイムラインおよびリソース制約を考慮しておく必要があります。ソース制御システムを通じてコンテンツを配布するプロセスが既に確立されているクライアントでは、この手法はさらに容易に実施できます。

提示した 2 つのソース制御モデルにおいて、重要な構成のバージョン管理を行うことは、開発作業を成功させるために必要不可欠です。共有構成を適切に管理できないと、開発作業を続行できなくなる可能性があります。

## 1.4 マイグレーション方法

このセクションでは、推奨されるマイグレーション方法について説明します。マイグレーション・マネージャには、2つの異なる操作モードがあります。

- ▶ スナップショット
- ▶ 変更

### 1.4.1 スナップショットの特性

スナップショット・モードは、文字どおり、パッケージの作成時点のシステムを示します。マイグレーション・マネージャはプログラムの後半で使用されるものであり、スナップショットにはインストール以降のすべての変更が表示されるため、このモードは頻繁に使用されます。スナップショットのデメリットは、(フィルタリングなしで)すべてをマイグレーションするか、そうでなければ、行われた変更のみを選択するフィルターを実務担当者が定義しなければならないことです。命名規則(すべての構造的な変更およびその他のデータベース変更に対してクライアント接頭部を付けるなど)を使用することで、このようなWhere文節を作成し、特定の変更セットについて、マイグレーション・パッケージのサイズを、対応する変更モード・パッケージにより作成されたサイズに制限することができます。

スナップショット・パッケージの考慮事項を以下に示します。

- ▶ スナップショット・パッケージでは、(1つ以上を対象として)フィルタリングしたり、完全な構成コンテンツをマイグレーションしたりすることができます。ソースとターゲットの同期化を確保することができます。
- ▶ スナップショット処理アクションを指定することで、構成コンテンツを完全に置き換えるのか、ターゲットの製品環境にマージするのかを決定できます。
- ▶ コンテンツ・エレメントの正確な名前に至るまで、マイグレーション要件全体を明確に定義、識別、および文書化します。
- ▶ 開発に時間がかかる膨大な量の構成がソース環境に存在し、バルク・マイグレーションが適切であり、以前のすべての特性をそのまま維持します。

### 1.4.2 変更の特性

変更モードにはイベント・リスナーが組み込まれており、これを有効にすると、無効にするまでの間、ビジネス・オブジェクトに対する変更が追跡されます。これにより、誰が、いつ、何をするかが追跡されます。コンテンツは、追跡された変更のみに基づいて抽出されます。

変更パッケージの考慮事項を以下に示します。

- ▶ このタイプのパッケージがアクティブ化された時点で、ソース環境とターゲット環境が既に同期されている場合に役立ちます(前述のように、実稼働前に最終マイグレーションが行われている場合など)。
- ▶ アクティブなパッケージがオーバーラップせず(同じオブジェクトを listen していない)、開発作業もオーバーラップせず(各開発作業で独自の変更パッケージを使用している)、以前のすべての特性をそのまま維持する場合にも、役立ちます。
- ▶ オーバーラップする開発作業が多すぎて、オーバーラップしないパッケージを定義するのが難しい場合、単一の変更パッケージを使用することで、変更の追跡およびマイグレーション要件の定義が容易になります。これは、変更パッケージを単なる追跡デバイスとして使用する、変更モードの混成手法です。実際のマイグレーションは、スナップショットを使用して行われます。

本書のシナリオの大部分は、スナップショット・パッケージとして設計および実装されています。

詳細については、224 ページの『10.6 開発環境での変更トラッキングと随時レポート』を参照してください。

### 1.4.3 混成マイグレーション戦略

混成マイグレーション戦略では、変更パッケージとスナップショット・パッケージを同時に使用します。この目的は、変更情報を自動的に取り込むことで、実務担当者や開発者に対して、自分たちが作成した個々の構成を文書化する手間隙を削減することです。この手法を実施することで、若干のコストは必要になりますが(1つではなく2つのパッケージ定義が必要になるため)、開発およびマイグレーションの作業時に大きなメリットを受けることができます。

この戦略では、両方のパッケージ定義で同じタイプのコンテンツを識別します。変更パッケージは排他的に定義され、最終的にスナップショット・パッケージでパッケージ化されるまったく同じ構成コンテンツを追跡します。特定の変更パッケージにより追跡される変更を表示するために、随時レポートが構成されます。

18 ページの図 1-8 に、マイグレーション・モードのライフサイクルを示します。

**ヒント:** 推奨される手法は、スナップショット・モードと変更モードの両方を使用することです(混成マイグレーション戦略)。実稼働サイクルでは開発用にスナップショット・モードを使用し、稼働後の開発やフィックスパックのプロモーション、マイナーな更新などでは(開発システム上で)変更モードに切り替えます。

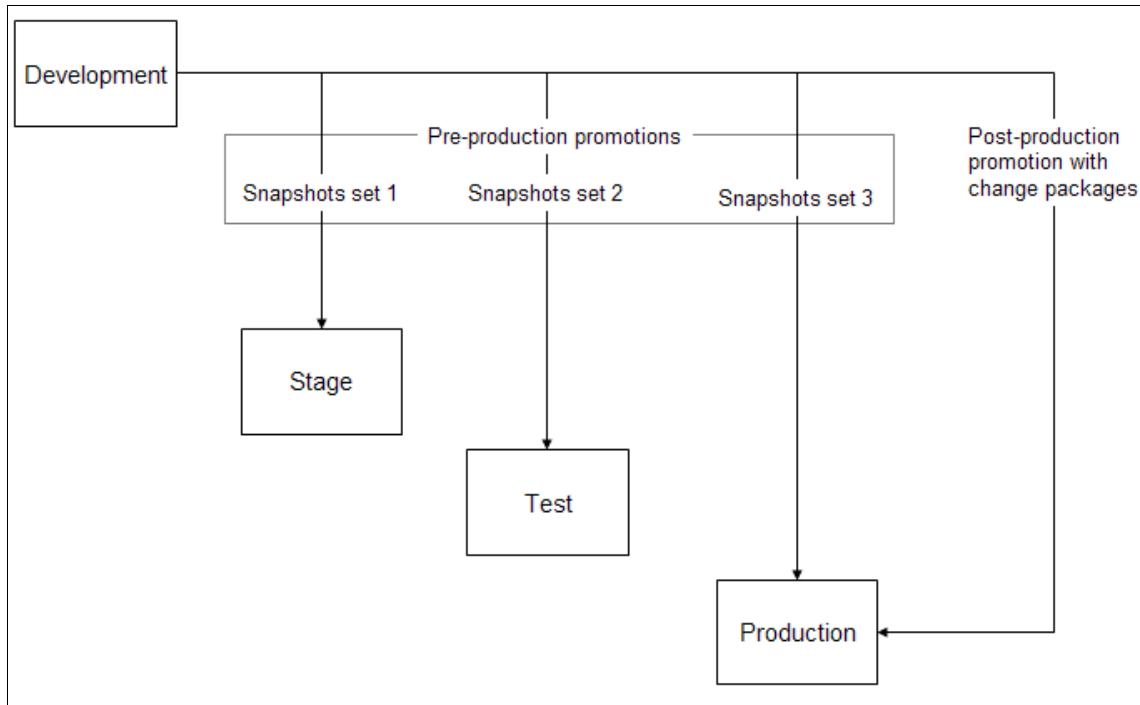


図 1-8 マイグレーション・モード

**注：**構造的な構成およびその他の構成に対して命名規則を使用する考え方は、構成の状態を検索および追跡するクエリーと組み合わせて使用することもできます。

詳細については、224 ページの『10.6 開発環境での変更トラッキングと随時レポート』を参照してください。

## 1.5 マイグレーション計画

マイグレーション計画とは、開発環境から後続の環境（ステージ、テスト、実稼働、またはトレーニングなど）に、構成コンテンツをプロモートする一連のアクションを系統立てたものです。マイグレーション計画は、管理された実際の文書の形式で明示します。これは、スキルを備えた人材が実行するマイグレーション・マネージャー・タスクへの入力として機能します。ただし、これはマイグレーション・マネージャーのタスクのみに限定されるものではありません。この計画では、マイグレーション・マネージャー機能の範囲外で実行す

る手動アクティビティーについても明示します。このような手動アクティビティーには、以下のものがあります。

- ▶ 対応する構成アプリケーションを使用した構成の作成(組織など)
- ▶ SQLスクリプトの実行(一連の担当者レコードのロードなど)
- ▶ 統合フレームワーク・ベースのデータ・ロードの実行(レガシー・システムからの一連のGL勘定科目または一連の企業レコードのロードなど)
- ▶ Javaクラスの作成プロセスの実行(製品のデプロイ可能EARファイルの作成など)

計画において重要な位置を占めているのは、このようなすべてのタスクを実行する順序と、タスク・オーナーの明確な識別です。目標は、明確でない部分を可能な限り排除し、後続の環境を準備してエンド・ユーザーに提供する作業を合理化することです。

この計画を策定することにより、実稼働前のマイグレーション、および実稼働後の定期的なマイグレーションが容易になります。

このようなマイグレーション計画文書を作成し、保守することで、以下のような重要なメリットがあります。

- ▶ マイグレーションの範囲を明確に設定できる
- ▶ 必要なツールおよびスキルを明確に特定できる
- ▶ タスクのオーナーを明確に識別できる
- ▶ マイグレーションの進行状況をモニターできる
- ▶ 関係者が最新の状態を把握できる

このIBM Redbookには、マイグレーション計画文書の内容に関する指針が示されています。本書で提供されているサンプルを利用するこどもできます。これらのサンプルをダウンロードには、271ページの『付録A追加資料』を参照してください。

### 1.5.1 マイグレーション計画の内容

マイグレーション計画文書には、スプレッドシート形式が適しています。このスプレッドシートは、共有環境で作成し、運用します。実務担当者によっては、スプレッドシートの改訂が管理され、異なるバージョンのスプレッドシートを保守および取得できる場合もあります。スプレッドシートの改訂を管理できるようにすることで、関係者は、さまざまなバージョンのスプレッドシートを比較したり、管理対象文書に対する許可アクセスや更新を徹底したりできます。

実務担当者の手法、クライアントの要件、およびITベース組織のポリシーに応じて、さまざまな種類のスプレッドシートを使用できます。ただし、すべてに共通する趣旨は、以下の詳細情報を提供することです。

- ▶ タスク
- ▶ タスクの順序
- ▶ オーナー
- ▶ ツール
- ▶ 日付

以下のセクションでは、合理的かつ再利用可能なスプレッドシート・ベースのマイグレーション計画について説明します。

スプレッドシートには、以下のワークシートが含まれます。

- ▶ マイグレーション計画の改訂履歴
- ▶ マイグレーション環境、製品、およびバージョン
- ▶ マイグレーション・タスク

### マイグレーション計画の改訂履歴

このワークシートでは、マイグレーション計画への変更と、その各変更の作成者を記録します。表 1-2 に、改訂履歴を記録する列を示します。

表 1-2 改訂履歴の記録

改訂	日付	作成者	説明	確認者	承認者

メリット：

- ▶ 計画に対する変更の監査証跡を提供する
- ▶ 確認および承認情報により責任所在を明確にする

### マイグレーション環境、製品、およびバージョン

このワークシートでは、IT ランドスケープの一部である各マイグレーション環境を記録します。少なくとも、ここには開発、テスト、および実稼働の各環境を含める必要があります。それぞれの環境について、インストールされる製品のリストを記載します。この情報は、製品環境の Web ユーザー・インターフェース (ユーザーのスタート・センターにある「システム情報」メニュー項目) から容易に取得できます。また、環境のタイプを記録します (マイグレーションの観点から、特定の環境がソースであるのかターゲットであるのか)。21 ページの表 1-3 に、環境、製品、および改訂に関する情報の例を示します。

表 1-3 環境、製品、および改訂情報を記録する列

環境のタイプ	製品およびバージョン	製品 URL	マイグレーション・パッケージの配布	パッケージ・ファイルの場所
開発	SRM Problem Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 SRM Incident Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 SRM Service Request Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 IBM Tivoli Common Process Components 7.2.0.01 Build 20100109D DB Build V7201-02 Base Services 7.1.1.6-LA20100312-0937 Build 20091208-1415 DB Build V7116-173 HFDB Build HF7116-04	<a href="http://hostname:9999/maximo">http://hostname:9999/maximo</a>	ファイル	H:\share\migration\packages

メリット：

- ▶ 関係者が、IT ランドスケープに関する情報を一目で確認できる
- ▶ ソースおよびターゲットに関して、実装者間の混乱を回避できる
- ▶ デプロイメント問題に関する情報を収集する時間を節約できる
- ▶ 実装リソースを移行する際のプロセス停止を回避できる

## マイグレーション・タスク

このワークシートは、計画の最も重要な要素です。このワークシートでは、下記のようなマイグレーション作業に関する多数の側面を記録します。

- ▶ 製品環境に対するデータの手動入力
- ▶ 統合フレームワークを使用した製品環境へのレガシー・データのインポート
- ▶ マイグレーション・マネージャーを使用した製品環境へのマイグレーション・パッケージのデプロイメント
- ▶ 必要なアプリケーション・サーバーの再始動
- ▶ 従属データに適応するために、マイグレーションの標準的な順序を変更する必要がある特殊な状況

表 1-4 に、推奨される手法に基づく、このワークシートの内容例を示しています。ワークシート内の行に記載の項目は、単なるサンプルです。構成の範囲に応じて、手動で行う手順の数、統合フレームワークを使用したデータ・インポートの必要性、およびマイグレーション作業を完了するために必要なマイグレーション・パッケージの数は、それぞれ異なる場合があります。このワークシートは、データ・マイグレーションに内在する不確実性を削減し、マイグレーション・アクティビティーを促進する上で、非常に役立ちます。

表 1-4 マイグレーション・タスク

順序	要約	説明	方法	オーナー
1	Integrity Checker	ターゲット・データベースで Integrity Checker をレポート・モードで実行し、エラーを修正します	手動	製品の実装者
2	Integrity Checker エラーおよび警告の修正	Integrity Checker を修復モードで使用して修正するか、SQL ツールを使用して手動で修正します	手動	製品の実装者
3	データベースのバックアップ	ターゲットの製品データベースをバックアップします	手動	DBA
4	EAR の準備	製品 EAR ファイルを作成します	手動	製品の実装者
5	EAR のデプロイ	製品 EAR をデプロイし、アプリケーション・サーバーを再始動します	手動	製品の実装者
6	通貨および換算レート	通貨および換算レートをロードします	手動	製品の実装者
7	組織 / 通貨	組織名と通貨を指定します	手動	製品の実装者
8	GL 構造	GL 勘定科目構造をセットアップします	手動	製品の実装者
9	勘定科目一覧	各組織について勘定科目一覧を作成します	手動	製品の実装者
10	デフォルトの勘定科目一覧	各組織についてデフォルトの勘定科目一覧を作成します	手動	製品の実装者
11	勘定科目一覧	統合フレームワークを使用して追加の勘定科目をロードします	統合フレームワーク	マイグレーション・マネージャー
12	システムおよびロケーション	システムおよびロケーションを定義またはロードします	手動または統合フレームワーク	製品の実装者

13	Integrity Checker の実行	開発データベースに対し Integrity Checker をレポート・モードで実行し、エラーを修正します	手動	製品の実装者
14	Integrity Checker エラーおよび警告の修正	Integrity Checker を修復モードで使用して修正するか、SQL ツールを使用して手動で修正します	手動	製品の実装者、開発者
15	データ・ディクショナリー(フェーズ1)	条件式をマイグレーションします	パッケージ	マイグレーション・マネージャー
16	データ・ディクショナリー(フェーズ2)	テーブルとクロスオーバーを除くすべてのドメインをマイグレーションします	パッケージ	マイグレーション・マネージャー
17	データ・ディクショナリー(フェーズ3)	オブジェクトと属性などのデータ・ディクショナリー構成をマイグレーションします	パッケージ	マイグレーション・マネージャー
18	データ・ディクショナリー(フェーズ4)	テーブル・ドメインおよびクロスオーバー・ドメインをマイグレーションします	パッケージ	マイグレーション・マネージャー
19	アプリケーション	アプリケーション・デザイナー・システム XML をマイグレーションします	手動	アプリケーション・デザイナー
20	アプリケーション	システム XML 以外のすべてのアプリケーションおよびメニューをマイグレーションします(製品環境がクラスター化されている場合、サーバーの再始動が必要です)	パッケージ	マイグレーション・マネージャー
21	担当者、ユーザー、およびセキュリティー・グループ	LDAP が有効なクライアントについて LDAP が同期化されている場合、手順 X に進みます	システム	製品の実装者
14	担当者、ユーザー、およびセキュリティー・グループ	LDAP が有効でない場合、担当者、ユーザー、およびセキュリティー・グループをマイグレーションします	パッケージ	マイグレーション・マネージャー
17	グローバルなデータ制限	セキュリティー・グループ・マイグレーションに関するセクションを参照してください	パッケージ	マイグレーション・マネージャー

18	アプリケーション・セキュリティ・パッケージ	署名オプションおよびクエリー用の、アプリケーション・セキュリティ・パッケージ	パッケージ	マイグレーション・マネージャー
19	スタート・センター	スタート・センターおよびクエリー・パッケージ(個々のユーザーがそれぞれスタート・センターを更新して変更を適用する必要があります)	パッケージ	マイグレーション・マネージャー
20	統合(フェーズ1)	統合オブジェクト構造をマイグレーションします	パッケージ	マイグレーション・マネージャー
21	統合(フェーズ2)	その他のすべての統合構成をマイグレーションします	パッケージ	マイグレーション・マネージャー
22	サービス・カタログ	サービス・カタログのオブジェクト構造パッケージ	パッケージ	マイグレーション・マネージャー
23	BPM(フェーズ1)	ワークフローを起動するアプリケーション・アクションを除く、ワークフローおよびその関連ロール、アクション、通信テンプレートをマイグレーションするためのBPM(ワークフロー)パッケージ	パッケージ	マイグレーション・マネージャー
24	BPM(フェーズ2)	ワークフローを起動するアプリケーション・アクションをマイグレーションします	パッケージ	マイグレーション・マネージャー
25	マイグレーション後のタスク	通信テンプレートおよびワークフローをアクティブ化します	手動	製品の実装者
26	分類	分類をマイグレーションします	パッケージ	マイグレーション・マネージャー
27	サービス・カタログ・テンプレート	サービス・カタログ・テンプレート・パッケージ	パッケージ	マイグレーション・マネージャー

計画文書の作成後、作成者が中心となり、関係者を集めてミーティングを開催し、計画の内容を検討することが推奨されます。これにより、すべての関係者に重要な情報を周知できるとともに、計画文書に足りない情報や誤った情報を見つけることができます。以降、定期的にミーティングを開くことで、マイグレーション・プロセスを継続的にモニターできます。

## 24 マイグレーション・マネージャーを使用したマイグレーションのユースケース

## 1.5.2 マイグレーションの時間枠での作業

マイグレーション作業は、常に、費やされた時間の観点からクライアントにより判断されるものであり、プロジェクト計画の一部として適切に調査する必要があります。

プロジェクトの稼働前フェーズにおいて、構成の作成には数週間を要する場合があり、複数の実務担当者や開発者が関与することになります。マイグレーションの計画とアクティビティにとって非常に重要であるのは、このフェーズ中に開発アクティビティとの密接な関係を維持することです。開発が完了するまでマイグレーション・アクティビティを延期することは、最も一般的な間違いです。クライアントの大部分は、マイグレーション・アクティビティが完了し、実稼働環境がエンド・ユーザーによるアクセス可能状態に仕上がるまでの期間を、数日間として計画しています。開発アクティビティとマイグレーション・アクティビティが分離してしまうと、クライアントがマイグレーションに求めている期待に応えることはできません。

**ベスト・プラクティス:** プロジェクトの稼働前フェーズにおいて、プロジェクトの開発アクティビティおよび構成アクティビティの完了とともに、マイグレーション計画文書を完成させます。

稼働後フェーズでは、いくつかの構成および開発アクティビティが進行することになります。このようなアクティビティは、エンド・ユーザーまたはアプリケーション・マネージャーからのフィードバックに基づいて行われます。既存の実稼働環境に対して、そのような保守や追加構成などをプロモートするために、小規模なマイグレーションも行われます。多くのクライアントは、このマイグレーションが特定の保守期間内で開始および終了することを求めていきます。最も一般的な保守期間は、2時間、4時間、または6時間です。通常、保守期間の開始は、エンド・ユーザー・アクティビティが最小となり、業務への影響を出さずに停止できる週末に設定されます。クライアントにより指定された保守期間内に保守アクティビティを完了することは、最優先事項です。保守期間のマイグレーション計画を作成することは、非常に有益です。この計画は、前のセクションで概要を説明した稼働前計画より簡素化できます。

多くの要因が、あらゆるタイプのマイグレーションの期間に影響します。その結果、実装作業が、求められているマイグレーションの時間枠を越えてしまう可能性もあります。マイグレーション・マネージャーまたは統合フレームワークには直接関与しない、非常に長い時間を要する要因としては、以下のものがあります。

- ▶ アプリケーション・サーバーの再始動

通常、実稼働環境では、クラスター化されたアプリケーション・サーバーで実行される複数の Java 仮想マシン (JVM) が使用されます。クラスター外部に、クーロン・タスク、統合、およびレポート作成機能をサポートする、追

加の JVM が存在する場合もあります。すべての JVM をシャットダウンして再始動する作業には、非常に長い時間がかかる場合があり、マイグレーション時間枠を削減させる要因となります。

▶ マイグレーション後のタスク

マイグレーションの完了後、通常は、手動で実行するタスク（ワークフロー・プロセス、エスカレーション、クーロン・タスクのアクティビ化など）が存在します。これらは製品のアプリケーション・ユーザー・インターフェースを利用して行うものであり、一定量のマイグレーション時間枠を消費する要因となります。

▶ Integrity Checker などのツールの実行

Integrity Checker などのツールを、定期的に実行する場合があります。製品データベースのサイズに応じて、ツールの実行が完了するまでの時間が変わるために、この作業も一定量のマイグレーション時間枠を消費する要因となります。

▶ その他の要因

DBA は、基礎となる製品データベースに対する構造的な変更を確認しなければならない場合があります。また、必要に応じて、データベース・オブジェクト（インデックス、テーブル・スペースなど）を変更する必要があります。この目的は、データベースの最適なパフォーマンスを確保するためです。このようなアクティビティーも、一定量のマイグレーション時間枠を消費する要因となります。

図 1-9 に、マイグレーション作業全体でさまざまなタイプのタスクが占める割合の内訳を示します。この図から、総合的にマイグレーションを実現するツールは、マイグレーション・マネージャーだけではないことが分かります。

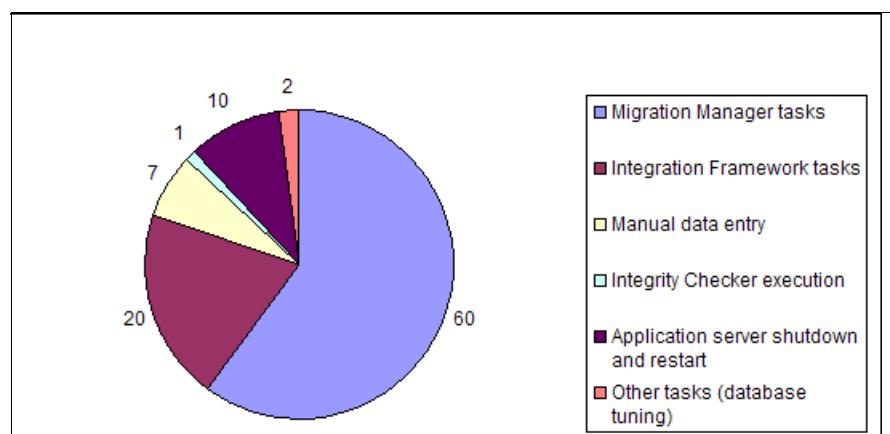


図 1-9 マイグレーション作業の内訳

これらの要因に留意してマイグレーション作業を管理することは、マイグレーション・マネージャーを使用して実際のマイグレーション・タスクを実行するのと同様に、複数の制約があるクライアント要件を管理することになります。クライアントとの間で率直な意見交換を行い、同意を得ていくことは、マイグレーションを予定どおりに成功させるために必要不可欠な作業です。

## 1.6 マイグレーション・マネージャーに関する参考資料

以下のリストは、マイグレーション・マネージャーに関する参考資料の一部です。これらの大部分は、IBM Tivoli サポート Web サイトのマイグレーション・マネージャーに関する技術情報として公開されています。

- ▶ マイグレーション・マネージャー V7.1.1.5 の機能 :

[http://www-01.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&dc=DB560&dc=DB520&uid=swg21393047&loc=en\\_US&cs=utf-8&lang=en](http://www-01.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&dc=DB560&dc=DB520&uid=swg21393047&loc=en_US&cs=utf-8&lang=en)

この資料は、Base Services V7.1.1.5 で導入されたユーザビリティ機能について説明しています。

- ▶ マイグレーション・マネージャー V7.1.1.5 チケット・テンプレート

[http://www-01.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&dc=DB560&dc=DB520&uid=swg21393063&loc=en\\_US&cs=utf-8&lang=en](http://www-01.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&dc=DB560&dc=DB520&uid=swg21393063&loc=en_US&cs=utf-8&lang=en)

この資料は、チケット・テンプレートをマイグレーションするためのサポートについて説明しています。

- ▶ マイグレーション・マネージャー V7.1.1.5 の分類 :

[http://www-01.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&dc=DB560&dc=DB520&uid=swg21393048&loc=en\\_US&cs=utf-8&lang=en](http://www-01.ibm.com/support/docview.wss?rs=3214&context=SSLKT6&dc=DB560&dc=DB520&uid=swg21393048&loc=en_US&cs=utf-8&lang=en)

この資料は、分類をマイグレーションするためのサポートについて説明しています。

- ▶ マイグレーション・マネージャーのロギングのセットアップ :

<http://www-01.ibm.com/support/docview.wss?uid=swg21397377>

この資料は、マイグレーション・マネージャー機能のロギングを構成する手順について説明しています。

- ▶ スタート・センター・マイグレーション :

<http://www-01.ibm.com/support/docview.wss?uid=swg21427580>

この資料は、スタート・センター・マイグレーションに関する追加情報を説明しています。

- ▶ 大規模なマイグレーション・マネージャー・パッケージをアップロードする際のサイズ制限の変更：

<http://www-01.ibm.com/support/docview.wss?uid=swg21408312>

この資料は、大規模なパッケージ・ファイルをアップロードする方法について説明しています。

- ▶ マイグレーション・マネージャー・プレビュー V7.1.1.6:

<http://www-01.ibm.com/support/docview.wss?uid=swg21414562>

この資料は、Base Services 7.1.1.6 で導入された、マイグレーション・マネージャーのプレビュー機能について説明しています。

- ▶ マイグレーション・マネージャーに先立つ Integrity Checker の実行：

<http://www-01.ibm.com/support/docview.wss?uid=swg21299691>

この資料は、マイグレーションを行う前に Integrity Checker ツールを実行する必要性について説明しています。

- ▶ Oracle の長セマンティクスによるマイグレーション・マネージャーへの影響：

<http://www-01.ibm.com/support/docview.wss?uid=swg21411696>

この資料は、Oracle の文字長セマンティクスがマイグレーションに与える影響について説明しています。

- ▶ MEA インターフェース・テーブル・マイグレーションの非サポート：

<http://www-01.ibm.com/support/docview.wss?uid=swg21299697>

この資料は、インターフェース・テーブル定義のマイグレーションに関するマイグレーション・マネージャーの制限について説明しています。

- ▶ 条件のマイグレーション：

<http://www-01.ibm.com/support/docview.wss?uid=swg21389955>

この資料は、条件式をマイグレーションする方法について説明しています。



## データ・ディクショナリーの マイグレーション

データ・ディクショナリーには、環境のビルディング・ブロックが含まれています。データ・ディクショナリーは、データベース・オブジェクト、属性、およびそれらの関係のほかに、組織、サイト、通貨などの基本的なユーザー情報を含むさまざまな情報で構成されています。データ・ディクショナリー・マイグレーションを使用するいくつかの例として考えられるのは、カスタム・アプリケーションを作成するために他のマイグレーション済みデータと組み合わせて使用する新規オブジェクト、その属性、関係、ドメインなどです。あるいは、新規の国際サイトをそのデフォルトの言語および通貨と一緒に追加することも考えられます。

この章には以下のセクションがあります。

- ▶ 30 ページの『2.1 要件』
- ▶ 30 ページの『2.2 ソリューション』
- ▶ 31 ページの『2.3 構成アプリケーション』
- ▶ 32 ページの『2.4 オブジェクト構造』
- ▶ 36 ページの『2.5 マイグレーション・グループ』
- ▶ 37 ページの『2.6 パッケージ定義』
- ▶ 48 ページの『2.7 デプロイメント』
- ▶ 49 ページの『2.8 デプロイメントに関する考慮事項』

## 2.1 要件

データ・ディクショナリー・オブジェクトは、開発環境で定義し、その後、実稼働環境にマイグレーションします。このシナリオでは、すぐに使用可能なデータ・ディクショナリー・マイグレーション・パッケージを使用します。パッケージ内のすべてのマイグレーション・オブジェクトをこのシナリオで明示的に扱うわけではありませんが、主なオブジェクトについては言及しています。

新しい開発環境では、必ず変更マイグレーション・パッケージをセットアップすることをお勧めします。変更パッケージを使用すると、開発環境のアクティビ化以降に行われた変更をマイグレーションできます。変更パッケージは、データ・マイグレーションの用途に使用しない場合でも、マイグレーション・マネージャーによってサポートされている環境の変更を追跡する目的で使用できます。

変更パッケージは、マイグレーションの対象となる構成の正確なリストが容易に作成できるように、新しいコンテンツを作成する前に、このシナリオでセットアップされています。

## 2.2 ソリューション

2セットの構成のマイグレーションを容易にするため、複数のパッケージが作成されます。

最初のセットは、新規サイトを既存の組織に追加するとプロモートされる標準コンポーネントの追加から成ります。これらのオブジェクトは、以下の順序でマイグレーションされます。

1. 通貨
2. セット
3. サイト

2番目のセットは、表付きの新しいタブを既存のアプリケーションに追加することをサポートする目的で、通常はデータ・ディクショナリーからマイグレーションされる構成と、特定のレコードの既存の値に基づいてデータをフィルター処理するためのいくつかの条件から成ります。マイグレーションされるオブジェクトは、順に以下のとおりです。

1. ルックアップ・マップ
2. ドメイン
3. オブジェクト

4. 関係
5. 条件式
6. ドメイン

ドメインは、このリストに2回出現します。クロスオーバー・ドメインおよびテーブル・ドメインは複雑であるため、これらを正しくマイグレーションするには、追加の手順が必要です。詳しくは、48ページの『2.7 デプロイメント』を参照してください。

データ・ディクショナリー・パッケージは、他のパッケージが必要とする基本的な情報を移動するため、マイグレーションの一環として満たす必要がある外部オブジェクト依存関係は存在しません。しかし、その他のパッケージは、データ・ディクショナリー・マイグレーションの一環として移動する構成に左右されます。すべてのマイグレーションを正しい順序で確実に実行するには、適切な計画を立てる必要があります。

## 2.3 構成アプリケーション

データベース構成アプリケーションは、オブジェクト、属性、インデックス、ビュー、テーブル、列、自動キー、関係、サービス、ルックアップ・マップ、メッセージ、順序、およびGL勘定科目構成を管理します。このアプリケーションにアクセスするには、スタート・センターで、「リンク先」→「システムの構成」→「プラットフォームの構成」→「データベースの構成」にナビゲートします。

ドメイン・アプリケーションは、ドメインを管理します。これらのドメインは、「データベースの構成」で属性と関連付けることができます。ドメインにアクセスするには、スタート・センターで、「リンク先」→「システムの構成」→「プラットフォームの構成」→「ドメイン」にナビゲートします。

セット・アプリケーションは、組織に関連付けられている部品セットおよび企業セットを管理します。セットにアクセスするには、スタート・センターで、「リンク先」→「管理」→「セット」にナビゲートします。

通貨コード・アプリケーションは、組織に関連付けられている通貨コードを管理します。通貨コードにアクセスするには、スタート・センターで、「リンク先」→「会計」→「通貨コード」にナビゲートします。

組織アプリケーションは、組織、サイト、および住所を管理します。これらにアクセスするには、スタート・センターで、「リンク先」→「管理」→「組織」にナビゲートします。

条件式マネージャー・アプリケーションは、条件式を管理します。条件式は、いくつかの他の構成オブジェクトと組み合わせて使用できます。条件式にアクセスするには、スタート・センターで、「リンク先」→「管理」→「条件式マネージャー」にナビゲートします。

オブジェクト構造アプリケーションは、データをインポートするために、統合フレームワークによって使用されるオブジェクト構造を管理します。あるいは、テーブルを結合して随時レポートを作成するために、「クエリー・ベース・レポート」機能によって使用されるオブジェクト構造を管理します。オブジェクト構造にアクセスするには、スタート・センターで、「リンク先」→「統合」→「オブジェクト構造」にナビゲートします。

## 2.4 オブジェクト構造

すぐに使用可能なデータ・ディクショナリー・マイグレーション・パッケージの一部である、すぐに使用可能なオブジェクト構造は、以下の 15 種類あります。

1. サービスのマイグレーションをサポートする DMMAXSERVICE オブジェクト構造(図 2-1):

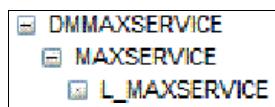


図 2-1 DMMAXSERVICE

2. 言語のマイグレーションをサポートする DMLANGUAGE オブジェクト構造(図 2-2):

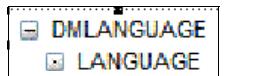


図 2-2 DMLANGUAGE

3. メッセージのマイグレーションをサポートする DMMAXMESSAGE オブジェクト構造(図 2-3):

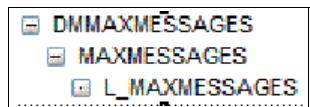


図 2-3 DMMAXMESSAGE

4. ルックアップのマイグレーションをサポートする DMMAXLOOKUPMAP オブジェクト構造(図 2-4):

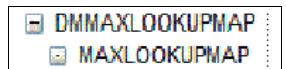


図 2-4 DMMAXLOOKUPMAP

5. 通貨のマイグレーションをサポートする DMCURRENCY オブジェクト構造(33 ページの図 2-5):

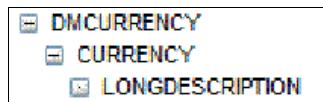


図 2-5 DMCURRENCY

6. 部品セットおよび企業セットのマイグレーションをサポートする DMSETS オブジェクト構造(図 2-6):

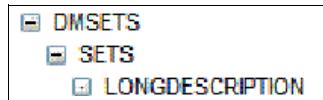


図 2-6 DMSETS

7. 組織、住所、およびサイトのマイグレーションをサポートする DMORGANIZATION オブジェクト構造(図 2-7):

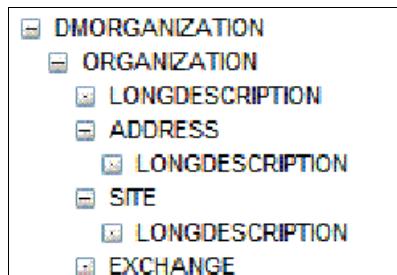


図 2-7 DMORGANIZATION

8. maxvars のマイグレーションをサポートする DMMAXVARS オブジェクト構造(図 2-8):

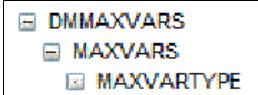


図 2-8 DMMAXVARS

9. ドメインのマイグレーションをサポートする DMMAXDOMAIN オブジェクト構造 (34 ページの図 2-9):

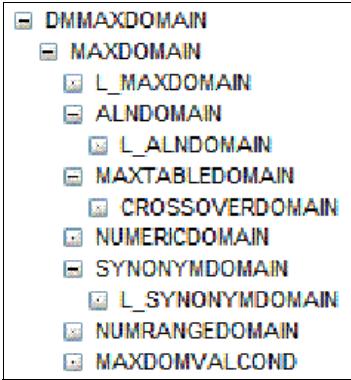


図 2-9 DMMAXDOMAIN

10. テーブル、ビュー、列、属性、およびインデックスのマイグレーションをサポートする DMMAXOBJECTCFG オブジェクト構造 (図 2-10):

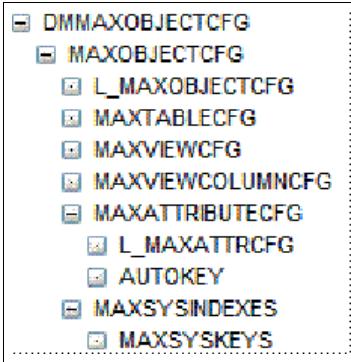


図 2-10 DMMAXOBJECTCFG

11. 順序のマイグレーションをサポートする DMMAXSEQUENCE オブジェクト構造 (図 2-11):

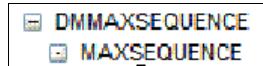


図 2-11 DMMAXSEQUENCE

12. 関係のマイグレーションをサポートする DMMAXRELATIONSHIP オブジェクト構造(図 2-12):

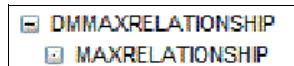


図 2-12 DMMAXRELATIONSHIP

13. 統合オブジェクトのマイグレーションをサポートする DMMAXINTOBJECT オブジェクト構造(図 2-13):

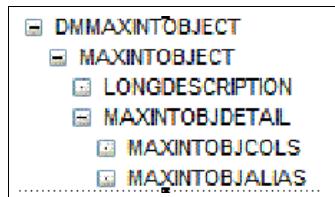


図 2-13 DMMAXINTOBJECT

14. 条件式のマイグレーションをサポートする DMCONDITION オブジェクト構造(図 2-14):

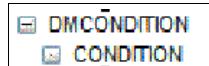


図 2-14 DMCONDITION

15. 総勘定元帳(GL)コンポーネントのマイグレーションをサポートする DMGLCONFIGURE オブジェクト構造:

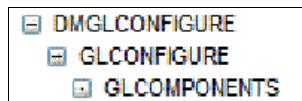


図 2-15 DMGLCONFIGURE

## 2.5 マイグレーション・グループ

前のセクションに記載したオブジェクト構造は、「データ・ディクショナリー」マイグレーション・グループの一部です。「データ・ディクショナリー」マイグレーション・グループは、すべてのマイグレーション・パッケージ・セットのうち、インポートされる最初のグループであるため、すぐに使用可能なパッケージをそのまま使用できるか、それをコピーしてから、マイグレーション対象の構成専用に変更できます。

図 2-16 に、すぐに使用可能な「データ・ディクショナリー」グループを示します。



図 2-16 「データ・ディクショナリー」グループ

マイグレーション・パッケージによって移行されるデータを制限する方法は、2つあります。最初の方法は、初期パッケージを複製し、すぐに使用可能なパッケージからオブジェクト構造を削除する方法です。2番目の方法は、このシナリオで使用します。この方法については、2.6、『パッケージ定義』で詳しく説明します。マイグレーション・マネージャー・アプリケーションの機能の1つである、既存のマイグレーション・パッケージを複製する機能は、この状況で役立ちます。個々の機能をマイグレーションするには、複数のデータ・ディクショナリー・パッケージが使用されるためです。データ・ディクショナリー・パッケージには依存関係が存在しないため、複製機能を使用して、パッケージ内の Where 文節を変更すると、処理時間をある程度節約できます。

マイグレーション・グループ内でのオブジェクト構造の順序を 37 ページの図 2-17 に示します。

Migration Object	Description	Migration Object Order
DMMAXSERVIC	System service migration object for Migration Manager	1
DMLANGUAGE	Language migration object for Migration Manager	2
DMMAXMESSA	System message migration object for Migration Manager	3
DMCURRENCY	Currency migration object for Migration Manager	4
DMSETS	Set migration object for Migration Manager	5
DMORGANIZAT	Organization and site migration object for Migration Manager	6
DMMAXVARS	MAXVARS migration object for Migration Manager	7
DMCONDITION	Condition expression migration object for Migration Manager	8
DMMAXDOMAIN	Domain migration object for Migration Manager	9
DMMAXOBJECT	Description: Condition expression migration object for Migration Manager Data dictionary migration object for Migration Manager	10
DMMAXSEQUEN	Data dictionary sequence migration object for Migration Manager	11
DMMAXLOOKU	Lookup map migration object for Migration Manager	12
DMMAXRELATI	Data dictionary relationship migration object for Migration Manager	13
DMMAXINTOBJ	Object structure migration object for Migration Manager	14
DMGLCONFIGU	GL configuration and component migration object for Migration Manager	15

図 2-17 データ・ディクショナリーのマイグレーション・オブジェクト

注: 37 ページの図 2-17 は、通常 5 つのオブジェクトのスクリーン・ショットを個別に 3 つ作成する代わりに、オブジェクト構造内にある 15 のオブジェクトをすべて表示するように変更されています。

## 2.6 パッケージ定義

すぐに使用可能なデータ・ディクショナリー・マイグレーション・パッケージを使用すると、データ・ディクショナリーに対して新規の変更マイグレーション・パッケージが作成されます。この処理は、このシナリオの一環として開発から実動にプロモートされるすべての構成を作成する前に実行されています。

マイグレーション・マネージャー・アプリケーションにアクセスするには、スタート・センターで、「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」にナビゲートします。

変更パッケージを作成するための処理は、きわめて単純であり、製品の資料に従って実行します。38 ページの図 2-18 に、新しい変更パッケージの「パッケージ定義 (Package Definition)」タブを示します。

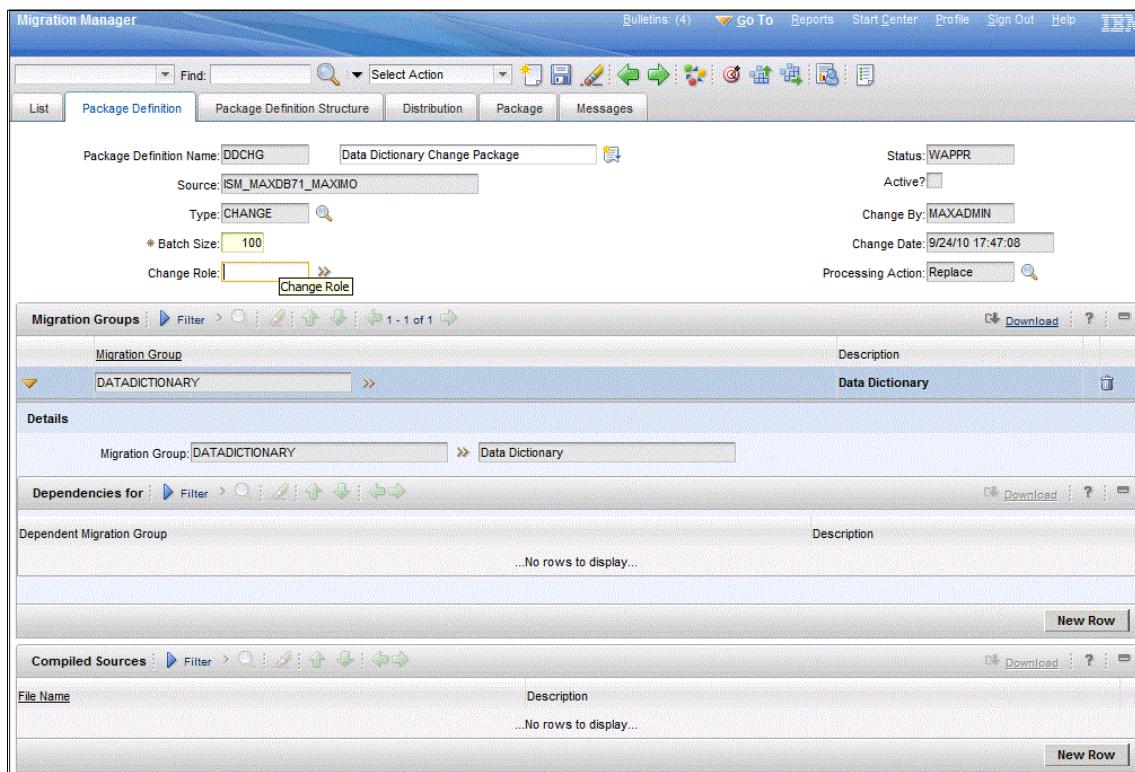


図 2-18 データ・ディクショナリー変更パッケージ

パッケージは「ステータスの変更」ツールバーまたは「アクションの選択」メニュー項目を使用して承認し、「パッケージ定義のアクティブ化」という「アクションの選択」メニュー項目を使用してアクティブ化します。いったん「アクティブ」になると、変更パッケージはデータ・ディクショナリーの変更点を追跡するようになります。

マイグレーション対象の構成を作成した後で変更パッケージにアクセスし、「イベント・トラッキング・レコードの表示」を選択すると、39 ページの図 2-19 に示すように、新しい構成が表示されます。

View Event Tracking Records							
Root Event Tracking Records			Filter > 🔍		1 - 10 of 10	Download	?
Object	Action	Primary Keys	Event Date	Event User	Object	Object Id	trash
▶ CONDITION	ADD	CONDITIONNUM=SUPERVISOR	9/24/10 19:45:26	MAXADMIN	CONDITION	979	trash
▶ CURRENCY	ADD	CURRENCYCODE=HKD	9/24/10 18:23:57	MAXADMIN	CURRENCY	18	trash
▶ MAXDOMAIN	ADD	DOMAINID=SUPERVISOR	9/24/10 19:55:59	MAXADMIN	MAXDOMAIN	768	trash
▶ MAXINTOBJECT	UPDATE	INTOBBJECTNAME=REP_PERSON	9/24/10 20:19:42	MAXADMIN	MAXINTOBJECT	203	trash
▶ MAXLOOKUPMAP	ADD	TARGET=EMPLOYEE,LOOKUPATTR=EMPLOYEE,TARGETATTR=EMPLOYEE,SOURCE=PERSON	9/24/10 20:09:46	MAXADMIN	MAXLOOKUPMAP	888,945	trash
▶ MAXOBJECTCFG	UPDATE	OBJECTNAME=PERSON	9/24/10 19:10:35	MAXADMIN	MAXOBJECTCFG	26	trash
▶ MAXOBJECTCFG	ADD	OBJECTNAME=EMPLOYEE	9/24/10 19:04:42	MAXADMIN	MAXOBJECTCFG	1,557	trash
▶ ORGANIZATION	UPDATE	ORGID=TEST	9/24/10 18:40:24	MAXADMIN	ORGANIZATION	8	trash
▶ SETS	ADD	SETID=ITEM1	9/24/10 18:28:35	MAXADMIN	SETS	12	trash
▶ SETS	ADD	SETID=COMP1	9/24/10 18:30:54	MAXADMIN	SETS	13	trash
Child Event Tracking Records			Filter > 🔍		1 - 2 of 2	Download	?
Object	Action	Primary Keys	Event Date	Event User	Parent Object	Parent Object Id	trash
MAXINTOBJDETAIL	ADD	INTOBBJECTNAME=REP_PERSON,OBJECTID=6	9/24/10 20:19:42	MAXADMIN	MAXINTOBJECT	203	trash
MAXINTOBJCOLS	ADD	INTOBBJECTNAME=REP_PERSON,OBJECTNAME=EMPLOYEE,NAME=HASLD,INTOBJFLDTYPE=EXCLUDE	9/24/10 20:19:42	MAXADMIN	MAXINTOBJDETAIL	919	trash

OK Cancel

図 2-19 データ・ディクショナリー変更パッケージのトラッキング・レコード

この時点で、プロモーション対象の項目を選択し、パッケージを作成して配布することができます。

データをマイグレーションするために、3つのパッケージが作成されます。最初のパッケージでは、通貨、組織、およびセットがマイグレーションされます。このパッケージは、複数のユーザーがデータ・ディクショナリーの個別セクションを、個々のプロジェクトごとに必要に応じて作成およびマイグレーションできることを示すために、他のパッケージとは分離しています。このパッケージには、新規の「組織」または「清算勘定」のマイグレーションは含まれていません。これに関しては問題があり、詳しくは 49 ページの『2.8 デプロイメントに関する考慮事項』の『組織および清算勘定』で説明しています。

2 番目と 3 番目のパッケージは、構成の残りの部分をマイグレーションするために使用されます。2 つのパッケージが必要なのは、クロスオーバー・ドメインを組み込むためです。クロスオーバー / テーブル・ドメインのマイグレー

ションに関する問題については、49 ページの『2.8 デプロイメントに関する考慮事項』を参照してください。

最初のステップは、パッケージの作成です。この作成では、製品資料に定義されている標準の手順に従います。前述の変更パッケージのセットアップに非常に似ていますが、例外が 2 つあります。1 つはパッケージの「タイプ」が「スナップショット」であること、2 番目はパッケージの Where 文節を定義するのに SQL の定義を使用することです。このことについては、以下で詳しく説明します。

- 最初のパッケージでは、更新する「組織」の名前、および追加する「セット」と「通貨」の名前は、最初に作成された変更パッケージのイベント・トランкиングか、または開発者がプロジェクト用に作成した資料にあります。これらの名前は、「通貨」、「組織」および「セット」の値を定義するために、SQL で使用されます。マイグレーションしないその他のオブジェクトは、すべて 1=2 と設定されます。以下に示す図 2-20 から 41 ページの図 2-22 までの図は、このパッケージでマイグレーションされるデータを示しています。



図 2-20 新規の通貨

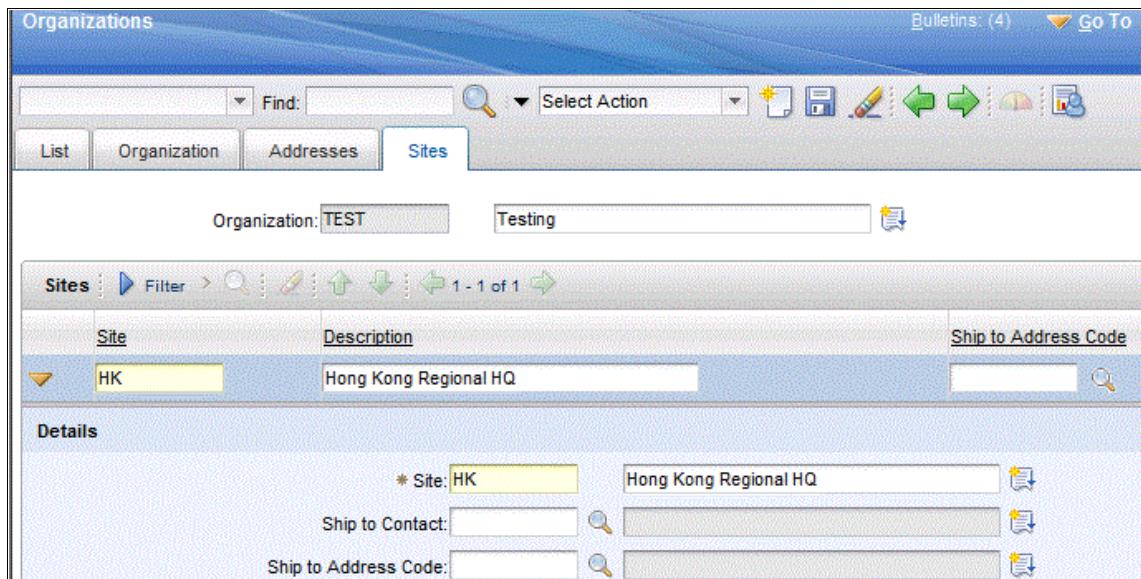


図 2-21 既存の組織に対する新規のサイト

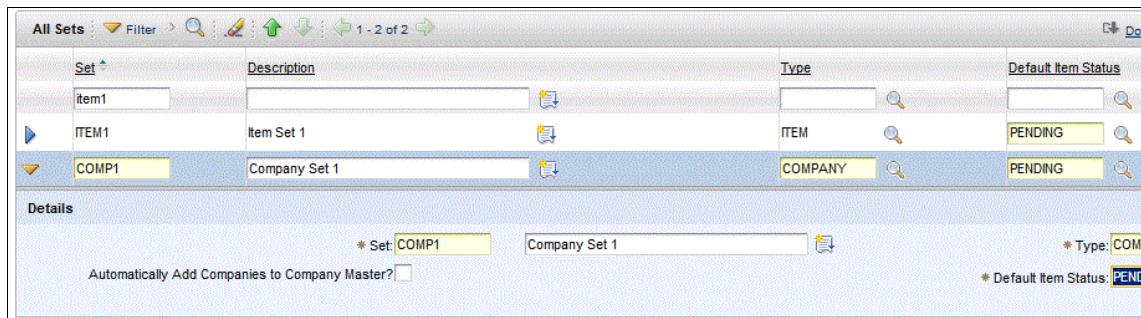


図 2-22 新規の部品セットおよび企業セット

2. これらの名前は、「通貨」、「組織」および「セット」の値を定義するために、SQL で使用されます。マイグレーションしないその他のオブジェクトは、すべて 1=2 と設定されます。各オブジェクトの Where 文節で使用された列名は、それぞれのアプリケーションでの更新ごとに、変更パッケージのイベント・キャッシングまたはフィールド・ヘルプで見つかります。42 ページの図 2-23 を参照してください。

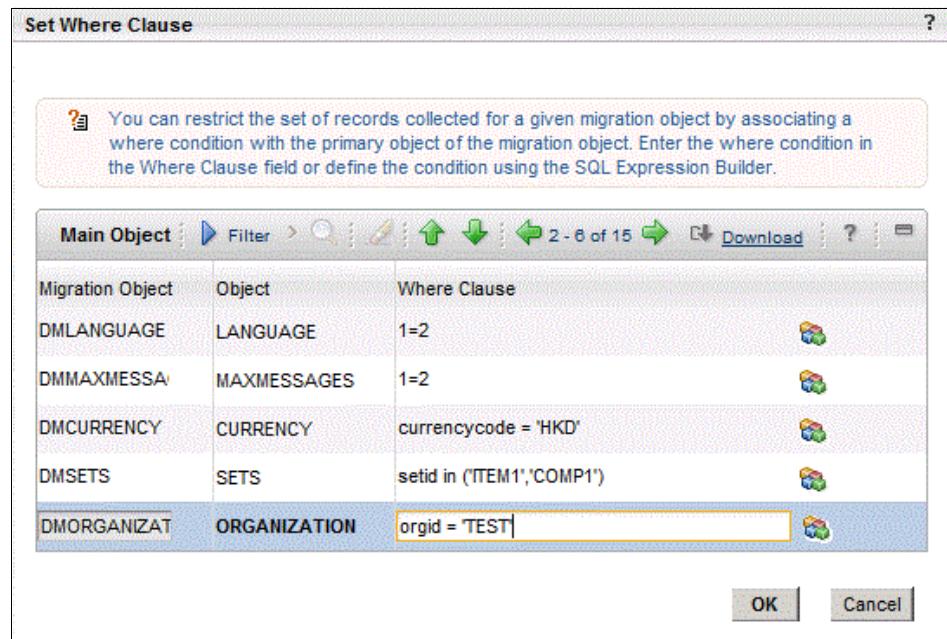


図 2-23 最初のスナップショット・パッケージの Where 文節

3. SQL 基準が更新され、すべての基本フィールドが定義されると、パッケージは保存されます。
4. その他のパッケージは、「アクションの選択」メニューの「パッケージ定義の複製」オプションを使用することにより、最近作成したスナップショット・パッケージから作成できます。こうすると、既存のパッケージ内のすべての情報が、「パッケージ定義名」と説明を除いて複製されます。これらは手動で入力する必要があります。
5. 2 番目のパッケージの Where 文節での SQL は、49 ページの『2.8.2 クロス オーバー・ドメインおよびテーブル・ドメイン』で説明するように、そのほかのすべてのオブジェクト(ただし、DOMAIN を除く)をマイグレーションするために更新されます。2 番目のパッケージのデプロイメント前にオブジェクト構造が変更され、3 番目のパッケージのデプロイメント前に元に戻されます。43 ページの図 2-24 から 47 ページの図 2-31 に、このパッケージでマイグレーションされたオブジェクトを示します。

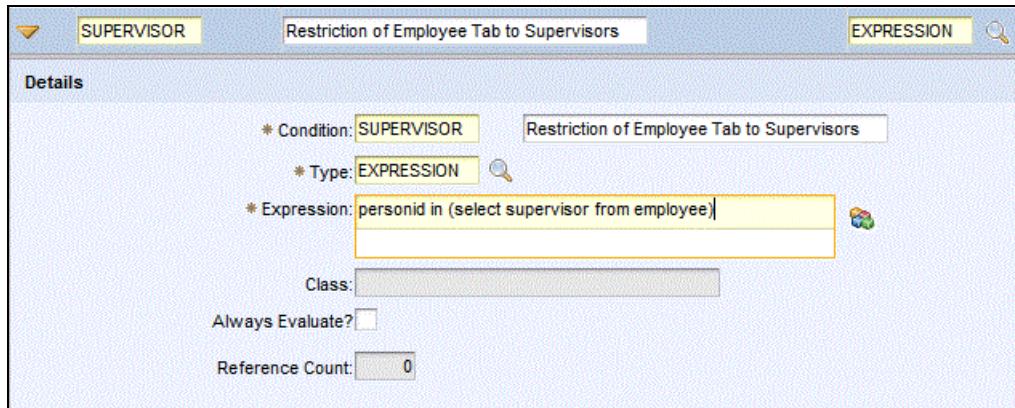


図 2-24 新規の条件式

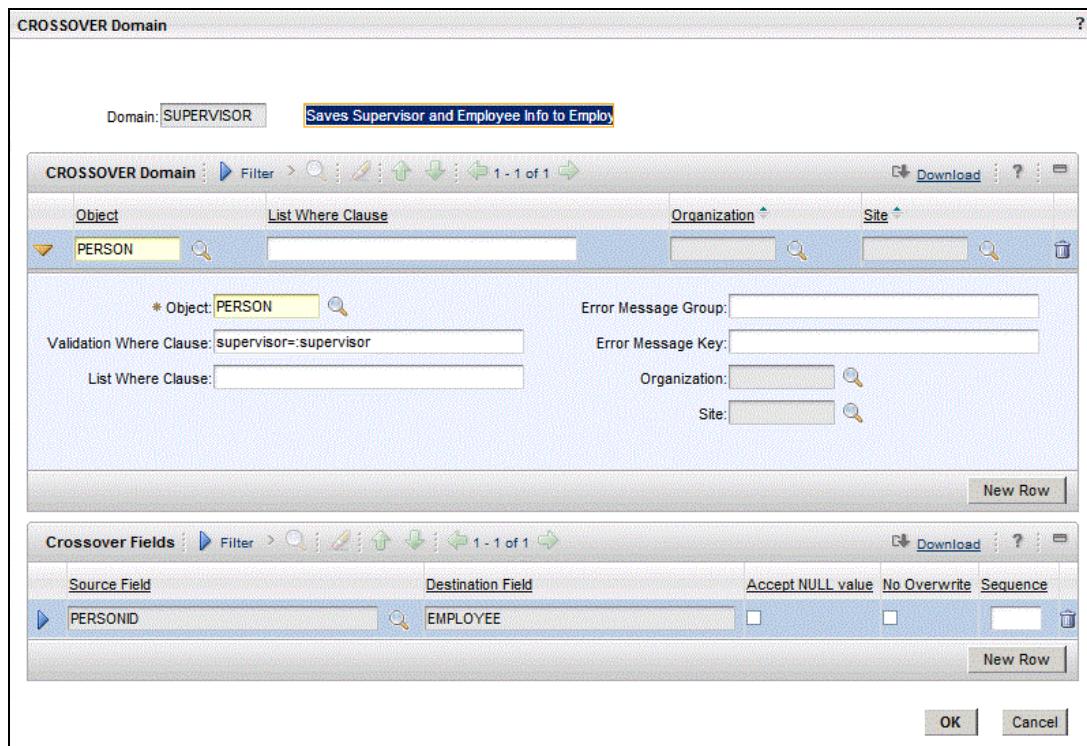


図 2-25 新規のクロスオーバー・ドメイン

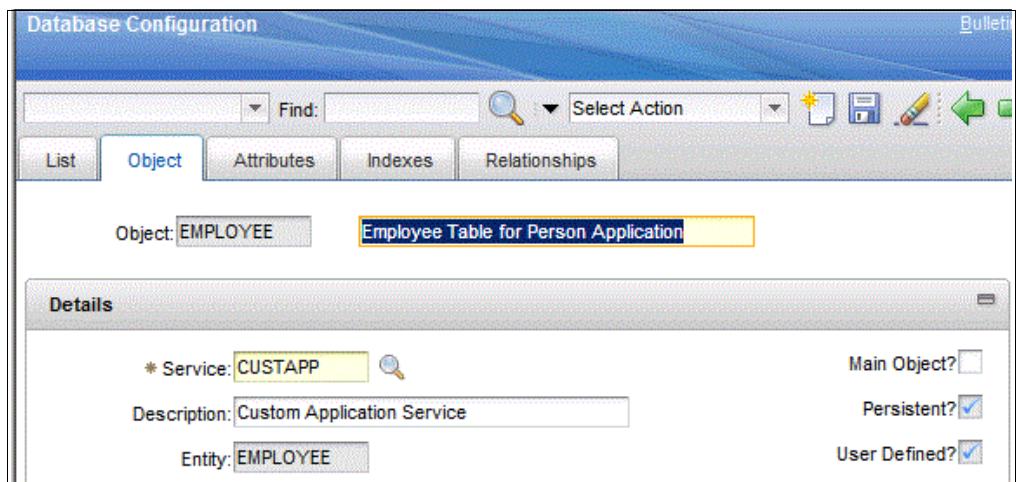


図 2-26 新規の従業員オブジェクト

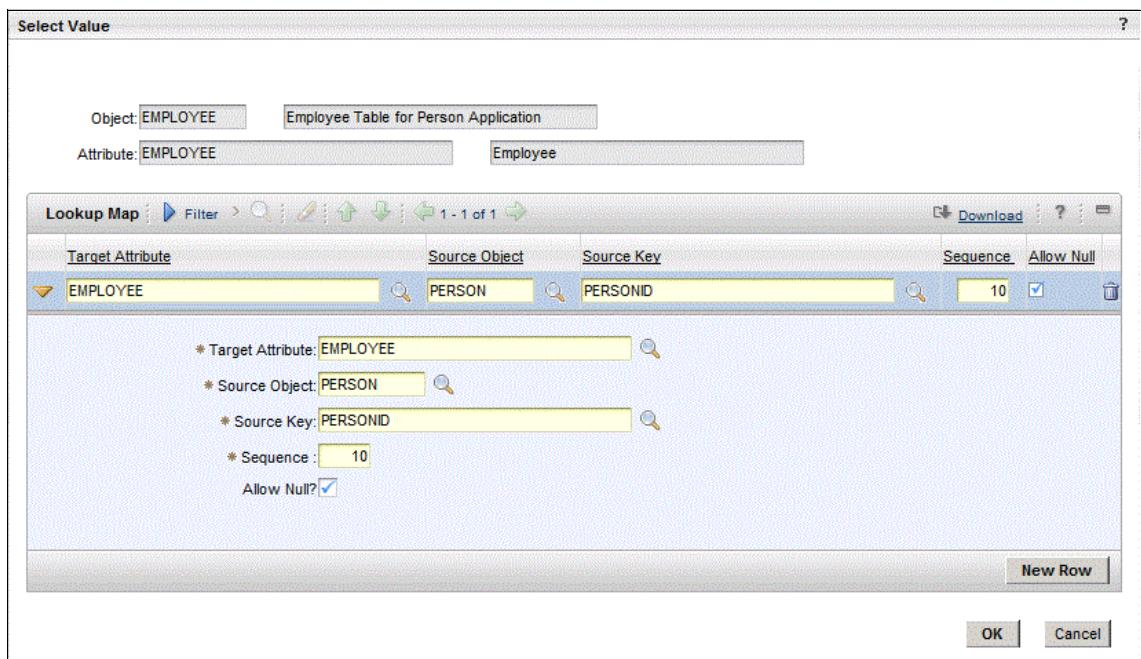


図 2-27 新規のルックアップ・マップ

Object: EMPLOYEE Employee Table for Person Application Status: To Be Added

Relationships Filter 1 - 1 of 1 Download ?

Relationship	Child Object	Where Clause	Remarks
▼ PERSON	PERSON	employee=:personid	Connection to the Person Table

**Details**

Relationship: PERSON	Child Object: PERSON
Where Clause: employee=:personid	Remarks: Connection to the Person Table

New Row

図 2-28 従業員オブジェクトの新規の関係

▼ EMPLOYEE EMPLOYEE personid=:employee

Details

* Relationship: EMPLOYEE	* Child Object: EMPLOYEE
Where Clause: personid=:employee	Remarks: Connection to Employee table.

図 2-29 担当者オブジェクトの新規の関係

Object Structures

Bulletins: (4) Go To Reports Start Center Profile Sign Out Help IBM

List Object Structure

Object Structure: REP\_PERSON Person Details Query Only?  User Defined?   
 \* Consumed By: REPORTING Reporting Self Reference?  Configurable?   
 Application: PERSON Support Flat Structure?  Alias Conflict?   
 Outbound Definition Class:  
 Inbound Processing Class:

Source Objects for REP\_PERSON Filter 2 - 6 of 6 Download ?

Object	Parent Object	Object Location Path	Relationship	Object Order	User Defined
PHONE	PERSON	PERSON/PHONE	PHONE	1	<input checked="" type="checkbox"/>
EMAIL	PERSON	PERSON/EMAIL	EMAIL	2	<input checked="" type="checkbox"/>
SMS	PERSON	PERSON/SMS	SMS	3	<input checked="" type="checkbox"/>
ASSETUSERCU	PERSON	PERSON/ASSETUSERCU	ASSETUSERCU	4	<input checked="" type="checkbox"/>
EMPLOYEE	PERSON	PERSON/EMPLOYEE	EMPLOYEE	5	<input checked="" type="checkbox"/>

Details

\* Object: EMPLOYEE Relationship: EMPLOYEE  
 Parent Object: PERSON Object Order: 5  
 Object Location Path: PERSON/EMPLOYEE Alternate Key:  
 Reporting Description: Employees for a Supervisor User Defined?   
 Cardinality: MULTIPLE Database Join Required?

図 2-30 既存の担当者レポート・オブジェクト構造での従業員オブジェクトへの新規の結合

the Where Clause field or define the condition using the SQL Expression Builder.

Main Object	Object	Where Clause	
DMMAXSERVIC	MAXSERVICE	1=2	
DMLANGUAGE	LANGUAGE	1=2	
DMMAXMESSA	MAXMESSAGES	1=2	
DMCURRENCY	CURRENCY	1=2	
DMSETS	SETS	1=2	
DMORGANIZAT	ORGANIZATION	1=2	
DMMAXVARS	MAXVARS	siteid = 'HK'	
DMCONDITION	CONDITION	conditionnum = 'SUPERVISOR'	
DMMAXDOMAIN	MAXDOMAIN	domainid = 'SUPERVISOR'	
DMMAXOBJECT	MAXOBJECTCFG	objectname in ('PERSON','EMPLOYEE')	
DMMAXSEQUEN	MAXSEQUENCE	tbname = 'EMPLOYEE'	
DMMAXLOOKU	MAXLOOKUPMAP	target = 'EMPLOYEE'	
DMMAXRELATI	MAXRELATIONSHIP	parent = 'EMPLOYEE' or (parent = 'PERSON' and	
DMMAXINTOBI	MAXINTOBJECT	intobjectname = 'REP_PERSON'	
DMGLCONFIGU	GLCONFIGURE	1=2	

OK Cancel

図 2-31 2 番目のデータ・ディクショナリー・スナップショット・パッケージ

6. これで、SQL 基準は更新されて 2 番目のパッケージに関連付けられたため、保存できます。
7. このパッケージが保存されると、最初のパッケージまたは 2 番目のパッケージの複製を基にして、3 番目のパッケージが作成されます。繰り返しになりますが、「パッケージ定義名」および説明は複製されないため、手動で入力する必要があります。
8. 最後のパッケージの SQL 基準は、MAXOBJECTCFG の Where 文節のみです。オブジェクト構造の除外が元に戻ると、このことによって、デプロイ後、オブジェクトとドメインとの間にリンクが作成されます。それ以外のオブジェクトは、マイグレーションの対象から除外するために、すべて 1=2 に設定されます。48 ページの図 2-32 を参照してください。

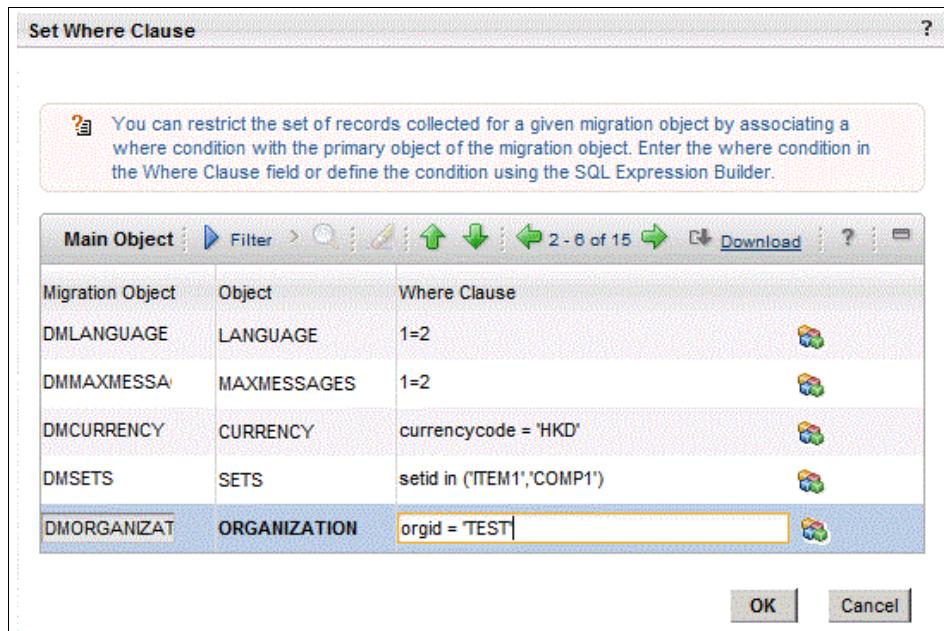


図 2-32 3 番目のデータ・ディクショナリー・スナップショット・パッケージ

9. これで、最後のパッケージの SQL 基準が更新されたため、パッケージを保存できます。

## 2.7 デプロイメント

パッケージは、保存したら、ソース・システムからエクスポートできます。クロスオーバー・ドメインのマイグレーションでは、49 ページの『2.8 デプロイメントに関する考慮事項』に定義されている特別なステップが必要です。2 番目のパッケージをエクスポートする前に、DMMAXOBJECTCFG オブジェクト構造から DOMAINID を除外しておく必要があります。さらに、3 番目のパッケージをエクスポートする前に、DMMAXOBJECTCFG オブジェクト構造に DOMAINID をもう一度組み込んでおく必要があります。

データベースに構造変更を加えるデータ・ディクショナリー・パッケージでは、マイグレーションを実行するために「管理モード」または再始動が必要です。この例では、構造変更が行われます。「管理モード」がアクティベート化したら、パッケージをインポートできます。デプロイメントのための正確なステップは、製品の資料に記載されています。

**ヒント:** データベースに対して大幅な構造変更を行う場合のベスト・プラクティスは、新規のマイグレーション・パッケージをインポートする前にバックアップを取ることです。

## 2.8 デプロイメントに関する考慮事項

以下では、このマイグレーションのデプロイメントに関する考慮事項について説明します。

### 2.8.1 組織および清算勘定

ここでマイグレーションされない 2 つの項目は、組織と最初の総勘定元帳 (GL) 勘定科目で、後者は組織内で清算勘定として識別されます。GL 勘定科目は組織と関連付ける必要がありますが、組織をアクティブ化してから GL 勘定科目を関連付けることはできません。マイグレーション・マネージャーは組織をアクティブ化する機能を備えていないため、関連を作成して組織をアクティブ化するには、手操作による介入が必要です。これらの項目については、新しい環境で値を手動で入力する方が効率的な場合があります。

### 2.8.2 クロスオーバー・ドメインおよびテーブル・ドメイン

クロスオーバー・ドメイン / テーブル・ドメインは、マイグレーション済みオブジェクトの 1 つです。これらのドメインを関連の属性と正しく関連付けるには、特別な配慮が必要です。このためには、ドメインを含むものと含まないものの 2 つのパッケージが必要です。これらは、以下のように作成され、デプロイされます。

- 最初のパッケージをドメイン参照なしで作成し、エクスポートして、新しい環境へインポートします。ソース環境内のオブジェクト構造 DMMAXOBJECTCFG からドメイン参照を一時的に除外するには、「フィールドの除外 / インクルード」ポップアップ・ダイアログを使用してオブジェクト構造を開き、MAXATTRIBUTECFG から DOMAINID フィールドを除外します。
- 2 番目のパッケージをドメイン参照ありで作成し、エクスポートして、新しい環境へインポートします。ドメイン参照を再度追加するには、最初のパッケージから DOMAINID フィールドを除外するために使用した前述の手順の逆に従います。この結果、属性とドメインとの間に適切なリンクが作成されます。

クロスオーバー・ドメインおよびテーブル・ドメインのマイグレーションに必要な除外または組み込みを 50 ページの図 2-33 に示します。

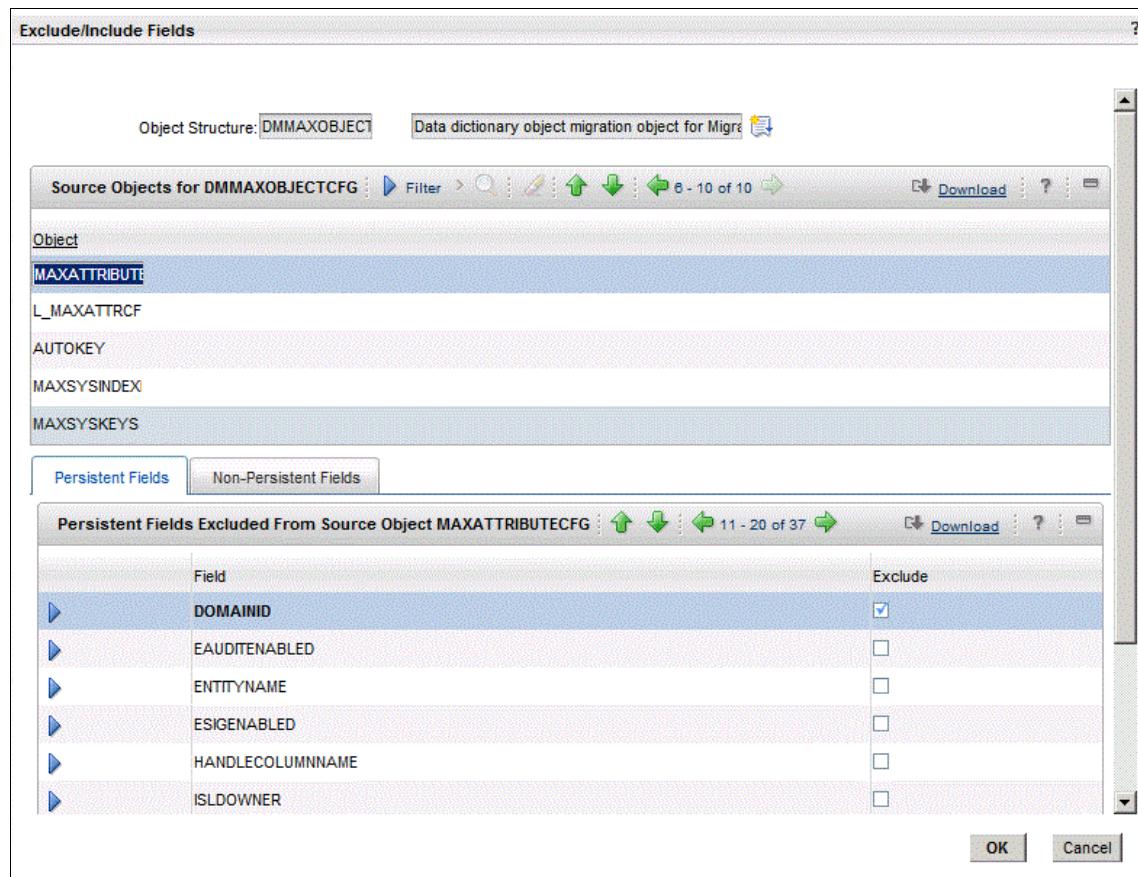


図 2-33 クロスオーバー・ドメインおよびテーブル・ドメインのマイグレーションに必要な除外または組み込み



## セキュリティー構成データのマイグレーション

この章では、セキュリティー構成データのマイグレーションのユースケース・シナリオについて説明します。管理者は、ユーザーを作成し、それらのユーザーをセキュリティー・グループに分類し、権限を付与することで、セキュリティー構成を実施します。説明するシナリオを以下に示します。

- ▶ 52 ページの『3.1 新規セキュリティー・グループのマイグレーション』
- ▶ 61 ページの『3.2 条件付きユーザー・インターフェースのマイグレーション』
- ▶ 70 ページの『3.3 グローバルなデータ制限のマイグレーション』
- ▶ 78 ページの『3.4 アクセス権限定義および LDAP 情報のマイグレーション』
- ▶ 85 ページの『3.5 セキュリティー・マイグレーションに関する考慮事項』

## 3.1 新規セキュリティー・グループのマイグレーション

システム管理者は、業務上の要件に基づいてシステムへのユーザーのアクセス権限を構成します。業務上の要件の例を以下に示します。

- ▶ Planning セキュリティー・グループに所属するユーザーのみが作業標準アプリケーションとルート・アプリケーションを使用できる。
- ▶ MAINTENANCE セキュリティー・グループに所属するユーザーが作業指示書を開始および完了できる。
- ▶ FINANCIAL セキュリティー・グループに所属するユーザーのみが勘定科目一覧アプリケーションを使用できる。
- ▶ MAINTENANCE セキュリティー・グループに所属するユーザーに対し、終了した作業指示書およびキャンセルされた作業指示書が表示されない。

このシナリオでは、新規ユーザーからなる新規セキュリティー・グループのマイグレーションと、既存のアプリケーション・オプションのアクセス権限定義を説明します。また、このシナリオではデータ制限に使用される条件のマイグレーションも説明します。

ここで説明する作業を行うには、管理ユーザーとしてログインする必要があります。

### 3.1.1 要件

このマイグレーションの要件として、新規ユーザー RBUSER が含まれている新規セキュリティー・グループ RBGROUP を定義する必要があります。このグループには、作業指示書管理アプリケーションのすべてのオプションへのフルアクセスがありますが、グループ・メンバーにはデータ制限が設定されており、承認待ちステータスの作業指示書を変更することはできません。

説明するすべての構成は、開発環境で作成され、その後テスト環境および実稼働環境にプロモートされます。

### 3.1.2 ソリューション

定義されている手法では、新規セキュリティー・グループとその関連データが1つのスナップショット・マイグレーション・パッケージでマイグレーションされます。

コンテンツの観点から見ると、このシナリオにおけるセキュリティー・グループのマイグレーションでは、以下の構成がこの順序でマイグレーションされます。

1. 条件

2. セキュリティー・グループ
3. ユーザー

**注:** セキュリティー・グループには、デフォルトのスタート・センター・テンプレート、このグループが許可されているサイトのリスト、およびアプリケーションが設定されていることがあります。サイト、スタート・センター・テンプレート、およびアプリケーションのマイグレーションは、このシナリオの要件ではありません。ここでは、これらの構成が既にマイグレーション済みであるか、またはターゲット環境内に既に存在していることを前提としています。詳しくは、59 ページの『3.1.8 デプロイメントに関する考慮事項』を参照してください。

### 3.1.3 構成アプリケーション

以降のセクションでは、このマイグレーション・シナリオの例として作成されたデータについて説明します。

#### 条件式マネージャー

条件とは、データ・アクセスやデータ編集を制限する場合などに使用される SQL 式またはカスタム Java クラスのいずれかです。条件によって定義される制限は、条件式が True と評価される場合にのみ適用されます。条件の作成および管理には、条件式マネージャー・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「管理」→「条件式マネージャー」をクリックします。

このシナリオのために作成した条件とその値を、表 3-1 に示します。

表 3-1 新規条件値

フィールド	値
条件	RBOOKCOND
説明	IBM Redbooks Condition
タイプ	EXPRESSION
式	:status = 'WAPPR'
常に評価	チェック

#### セキュリティー・グループ

セキュリティー・グループの作成および管理には、セキュリティー・グループ・アプリケーションを使用します。このアプリケーションを使用するには、

「リンク先」→「セキュリティ」→「セキュリティ・グループ」をクリックします。

このシナリオのために作成した新規グループとその値を、表 3-2 に示します。

表 3-2 新規セキュリティ・グループの値

フィールド	値
グループ	RBGROUP
説明	IBM Redbooks Group

「サイト」タブの「グループにすべてのサイトを許可」チェック・ボックスにチェック・マークが付いています。

「データ制限」タブで、WORKORDER オブジェクトのオブジェクト制限が作成されました。このデータ制限が True として評価されると、承認待ちの作業指示書の編集設定が読み取り専用に変更されます。表 3-3 に、この新規データ制限の値を示します。

表 3-3 新規オブジェクト制限

フィールド	値
オブジェクト	WORKORDER
タイプ	READONLY
条件	RBOOKCOND

**重要:** 同一条件を複数のデータ制限に使用できます。この例では「オブジェクト」フィールドに WORKORDER を設定することで、この関連付けが作成されます。

この条件を適用できる他のオブジェクトに、この条件を関連付けることができます。

「アプリケーション」タブで、作業指示書管理アプリケーションのすべてのオプションへのアクセス権限がこのグループに付与されました。

「GL 勘定科目構成」タブのすべてのコンポーネントのチェック・ボックスにチェック・マークが付いています。

## ユーザー

ユーザーの作成および管理には、ユーザー・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「セキュリティ」→「ユーザー」をクリックします。

このシナリオのために作成した新規ユーザーとその値を、表 3-4 に示します。

表 3-4 新規ユーザーの値

フィールド	値
ユーザー	RBUSER1
ユーザー名	RBUSER1
タイプ	TYPE 1

### 重要:

- ▶ 「担当者」フィールドは必須フィールドです。新規担当者を作成するかどうかを尋ねられます。このシナリオでは、担当者アプリケーションで作成したデータのマイグレーションは行いません。既にターゲット環境に存在しており、ステータスがアクティブである担当者が選択されることを前提としています。
- ▶ 本製品には、DMPERSON オブジェクト構造を持つ事前定義マイグレーション・グループ RESOURCES が用意されています。
- ▶ パスワード・フィールドが必須フィールドであることがあります。この場合は、ご使用のシステムの構成に基づいて有効な値を入力する必要があります。この場合、すべての必須データを使用して新規ユーザーを作成できることを前提としています。
- ▶ 参照データとパスワードのマイグレーションについて詳しくは、59 ページの『3.1.8 デプロイメントに関する考慮事項』を参照してください。

「グループ」タブに、新規グループ RBGROUP が追加されました。

### 3.1.4 オブジェクト構造

このシナリオでのアプリケーション・セキュリティ・マイグレーションをサポートするために製品と共に提供されているオブジェクト構造を以下に示します。

- ▶ 条件をサポートする DATADICTIONARY マイグレーション・グループの DMCONDITION オブジェクト構造を図 3-1 に示します。

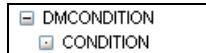


図 3-1 条件オブジェクト構造

- ▶ ユーザーをサポートする APPSECURITY マイグレーション・グループの DMMAXUSER オブジェクト構造を図 3-2 に示します。

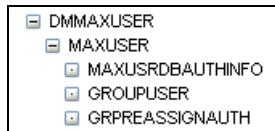


図 3-2 ユーザー・オブジェクト構造

- ▶ セキュリティ・グループをサポートする APPSECURITY マイグレーション・グループの DMMAXGROUP オブジェクト構造を図 3-3 に示します。

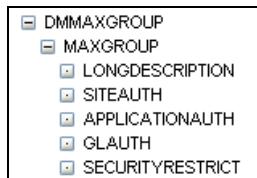


図 3-3 セキュリティ・グループ・オブジェクト構造

### 3.1.5 マイグレーション・グループ

3.1.4. 『オブジェクト構造』に示すオブジェクト構造は、説明からわかるように 2 つの個別のマイグレーション・グループに含まれています。本製品で使用可能なマイグレーション・グループには、その他のマイグレーション・グループへの依存関係が含まれていますが、このシナリオではこの依存関係は必要ありません。

作成したデータを適切かつ効率的にマイグレーションする方法として、このシナリオのマイグレーション要件を実現することが確認されたオブジェクト構造のみを含む新規マイグレーション・グループを定義する方法があります。

新規マイグレーション・グループを作成するには、マイグレーション・グループ・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・グループ」をクリックします。

**ヒント:**既存のグループ APPSECURITY を複製し、すべての依存関係を削除することができます。

新規マイグレーション・グループの値を表 3-5 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-5 新規マイグレーショングループ

フィールド	値
マイグレーション・グループ	RBSECURITY01
説明	IBM Redbooks Security 01

新規グループに含まれるオブジェクト構造とその順序を図 3-4 に示します。

Migration Object	Description	Migration Object Order	Internal
DMCONDITION >>	Condition expression migration object for Migration Manager	1	<input type="checkbox"/>
DMSIGOPTION >>	Signature option migration object for Migration Manager	2	<input type="checkbox"/>
DMMAXSERVSI >>	Service security migration object for Migration Manager	3	<input type="checkbox"/>
DMMAXGROUP >>	Security group migration object for Migration Manager	4	<input type="checkbox"/>
DMMAXUSER >>	User migration object for Migration Manager	5	<input type="checkbox"/>
DMSIGOPTFLAT >>	Advanced signature option migration object for Migration Manager	6	<input type="checkbox"/>
DMCTRLGROUP >>	Control security migration object for Migration Manager	7	<input type="checkbox"/>

図 3-4 RBSECURITY01 オブジェクト構造

### 3.1.6 パッケージ定義

この時点で、データ・セットとマイグレーション・グループの両方が定義されたため、マイグレーション・プロセスを開始できます。最初にマイグレーション・マネージャー・アプリケーションを使用してパッケージを定義します。マイグレーション・マネージャー・アプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」をクリックします。

1つのスナップショット・パッケージで、作成したセキュリティ構成をマイグレーションできます。このパッケージには、56 ページの『3.1.5 マイグレーション・グループ』で作成したマイグレーション・グループを含める必要があ

ります。このパッケージの値を表 3-6 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-6 RBSECPKG01 マイグレーション・パッケージの値

フィールド	値
パッケージ定義名	RBSECPKG01
説明	IBM Redbooks Security Pkg 01
タイプ	SNAPSHOT
処理アクション	AddChange

「マイグレーション・グループ」タブに、マイグレーション・グループ RBSECURITY01 が追加されました。

## SQL 基準の定義

マイグレーション・グループに含まれる各オブジェクト構造に対して作成されたクエリーを図 3-5 に示します。

Migration Object	Object	Where Clause
DMCONDITION	CONDITION	conditionnum in ('RBOOKCOND')
DMSIGOPTION	SIGOPTION	1=2
DMMAXSERVSI	MAXSERVSECURITY	1=2
DMMAXGROUP	MAXGROUP	groupname in ('RBGROUP')
DMMAXUSER	MAXUSER	status <> 'DELETED' and userid in(select userid from groupuser where groupname in ('RBGROUP'))
DMSIGOPTFLAG	SIGOPTFLAG	1=2
DMCTRLGROUP	CTRLGROUP	1=2

図 3-5 オブジェクト構造のクエリー

マイグレーション・オブジェクト DMMAXUSER の完全な Where 文節を以下に示します。

```
status <> 'DELETED' and userid in(select userid from groupuser where groupname in ('RBGROUP'))
```

**ヒント:**一部のクエリーは `1=2` と定義されています。その他のセキュリティ・マイグレーション関連シナリオのために新規マイグレーション・グループを作成する必要はありません。別のパッケージを作成し、異なるフィルターを定義してください。

入力された SQL Where 文節が正しくない場合、マイグレーション・マネージャー・アプリケーションからエラー BMXAA7010E が報告されます。パッケージ定義を保存するには、SQL Where 文節を訂正する必要があります。

**ベスト・プラクティス:**どんな場合も、マイグレーション・パッケージにはステータスが `DELETED` のユーザーを含めないようにしてください。このようなユーザーが含まれているマイグレーション・パッケージをターゲット環境にデプロイしようとすると、エラー BMXAA3837E が報告されます。

また、システム・ユーザー MAXADMIN および MAXINTADM をマイグレーションしないようにしてください。これらのシステム・ユーザーが含まれているマイグレーション・パッケージをターゲット環境にデプロイしようとすると、エラー BMXAA3829E が報告されます。

### 3.1.7 デプロイメント

パッケージ定義の保存と承認が完了したら、パッケージの作成、配布、およびデプロイに進むことができます。

### 3.1.8 デプロイメントに関する考慮事項

このシナリオでは、セキュリティ構成がソース環境と同様にターゲット環境で機能できるようにするために、このセクションで説明する重要なポイントに従ってください。

#### 基本的な依存関係

前述したように、実行したマイグレーションは、ターゲット環境に存在する必要があるデータに基づいています。基本的な依存関係を以下に示します。

- ▶ ターゲット環境では管理モードがオンである必要がある。
- ▶ デプロイ前に、新規アプリケーション、新規オブジェクト、および既存のオブジェクトの変更内容をマイグレーションする必要がある。

**ベスト・プラクティス:** ターゲットにはソース環境と同じライセンス・キーを適用してください。ライセンスが異なる場合、異なるアプリケーション・セットが使用可能または使用不可になり、これが原因でデプロイメント・エラーが発生します。

- ▶ セキュリティー・グループにデフォルト・スタート・センター・テンプレートを使用している場合は、デプロイ前にこのスタート・センターをターゲット環境にマイグレーションするか、またはターゲット環境で手動で作成する必要があります。スタート・センター・テンプレートのマイグレーションについて詳しくは、115ページの『第5章 アプリケーションとスタート・センターのマイグレーション』を参照してください。
- ▶ デプロイ前に、グループが許可されているサイトをターゲット環境にマイグレーションするか、またはターゲット環境で手動で作成する必要があります。
- ▶ この例で作成したグループは、GL 勘定科目コンポーネントの使用を許可します。デプロイ前に、GL 勘定科目構成をターゲット環境にマイグレーションするか、またはターゲット環境で手動で作成する必要があります。
- ▶ この例で作成したユーザーは、既存の担当者に関連付けられています。デプロイ前に、すべての関連担当者をターゲット環境にマイグレーションするか、またはターゲット環境で手動で作成する必要があります。担当者の操作については、87ページの『第4章 エスカレーションのマイグレーション』を参照してください。

**注:** アプリケーション・オプション (SIGOPTION) のマイグレーションは、このシナリオのマイグレーション要件ではありません。この構成は、本製品に含まれている DMSIGOPTION オブジェクト構造を使用しており、61ページの『3.2 条件付きユーザー・インターフェースのマイグレーション』で説明するマイグレーション要件の一部です。

## パスワードの処理

マイグレーション・マネージャー・アプリケーションでは、セキュリティー上の理由からパスワードはマイグレーションされません。マイグレーションされるすべてのユーザーのパスワードはリセットされ、新規の一時パスワードを通知する電子メール・メッセージが各ユーザーに送信されます。この一時パスワードは、初回ログインの後で有効期限切れになります。このため、セキュリティー・パッケージのマイグレーション前にターゲット環境の電子メール・サーバーを構成しておく必要があります。また、マイグレーションされるすべてのユーザーの担当者レコードに、有効な電子メール・アドレスが含まれている必要があります。

**注:** ご使用のシステムが LDAP を使用して構成されていないか、またはパスワードが自動生成されない状態で、電子メール・サーバーが構成されていないか、またはユーザーが一時パスワードを受信しない場合、管理権限が付与されているユーザーが一時パスワードを手動で定義できます。

LDAP のシナリオについては、78 ページの『3.4 アクセス権限定義および LDAP 情報のマイグレーション』を参照してください。

## 属性の制限

条件を使用して、オブジェクトと同様に属性に対してもデータ制限を定義できます。属性データ制限は、オブジェクト制限と同様の方法でマイグレーションされます。属性データ制限は、オブジェクト全体ではなく属性のみに適用される点が異なります。

### マイグレーションされないデータ

マイグレーションされないセキュリティー関連データを以下に示します。

- ▶ パスワードのヒント(質問と答えの両方)
- ▶ ログイン・トラッキング
- ▶ ネイティブ・データベース・ユーザー
- ▶ 保管場所のアクセス権
- ▶ 作業員権限
- ▶ 限度額と許容額
- ▶ ステータス履歴
- ▶ 購買 GL

## 3.2 条件付きユーザー・インターフェースのマイグレーション

製品構成の一部として、ユーザーは既存の画面の再定義、新規アプリケーション・オプションの作成、および業務上の要件に基づいてユーザー・インターフェース・コントロールを表示 / 非表示にする条件の設定を行うことができます。業務上の要件の例を以下に示します。

- ▶ ユーザーが ENGINEERS グループに所属しており、作業指示書が予防保全である場合に、作業指示書管理アプリケーションの「故障レポート」タブを非表示にする。
- ▶ 未完了の作業指示書を表示するときに、作業指示書管理アプリケーションのメインタブの「実際の開始」フィールドと「実際の終了」フィールドを非表示にする。

このシナリオでは、新規アプリケーション・オプション、新規条件付きユーザー・インターフェース定義、新規セキュリティー・グループ、および新規アクセス権限定義のマイグレーションを説明します。

ここで説明する作業を行うには、管理ユーザーとしてサインインする必要があります。

### 3.2.1 要件

このシナリオの要件は、作業指示書にアクセスするユーザーが RBGROUP1 のメンバーであり、この作業指示書が未完了の場合に「実際の開始」フィールドと「実際の終了」フィールドを非表示にすることです。この要件を構成する方法として、新規セキュリティー・グループ RBGROUP1 と作業指示書管理アプリケーションの新規オプション RBOPTION1 を定義し、作業指示書のステータスに基づく条件を使用してフィールドを非表示にするように条件付きユーザー・インターフェースを構成する方法があります。

説明するすべての構成は、開発環境で作成され、その後テスト環境および実稼働環境にプロモートされます。

### 3.2.2 ソリューション

定義されている手法では、一連の新規アプリケーション・オプション、条件付きユーザー・インターフェース、新規グループおよびその関連データが 1 つのスナップショット・マイグレーション・パッケージでマイグレーションされます。

このシナリオで説明する条件付きユーザー・インターフェースのマイグレーションは、以下の構成がこの順序でマイグレーションされることを意味します。

1. 条件
2. SIGOPTION
3. セキュリティー・グループ
4. コントロール・セキュリティー

### 3.2.3 構成アプリケーション

以降のセクションでは、このマイグレーション・シナリオの例として作成された構成について説明します。

## 条件式マネージャー

条件とは、データ・アクセスやデータ編集を制限する場合などに使用される SQL 式またはカスタム Java クラスのいずれかです。条件によって定義される制限は、条件式が True と評価される場合にのみ適用されます。条件の作成および管理には、条件式マネージャー・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「管理」→「条件式マネージャー」をクリックします。

このシナリオのために作成した条件とその値を、表 3-7 に示します。

表 3-7 新規条件 値

フィールド	値
条件	RBOOKCOND1
説明	IBM Redbooks Condition 1
タイプ	EXPRESSION
式	:status = 'COMP'
常に評価	チェック

## セキュリティー・グループ

セキュリティー・グループの作成および管理には、セキュリティー・グループ・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「セキュリティー」→「セキュリティー・グループ」をクリックします。

このシナリオのために作成した新規グループとその値を、表 3-8 に示します。

表 3-8 新規セキュリティー・グループの値

フィールド	値
グループ	RBGROUP1
説明	IBM Redbooks Group 1

「サイト」タブの「グループにすべてのサイトを許可」にチェック・マークが付いています。

**注:**すべてのサイトを許可しない場合は、その前に必ずそれらのサイトをマイグレーションしてください。詳しくは 59 ページの『3.1.8 デプロイメントに関する考慮事項』を参照してください。

「アプリケーション」タブで、作業指示書管理アプリケーションのすべてのオプションへのアクセス権限がこのグループに付与されました。

「GL 勘定科目構成」タブのすべてのコンポーネントのチェック・ボックスにチェック・マークが付いています。

## アプリケーション・デザイナー

アプリケーションの作成および管理には、アプリケーション・デザイナー・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アプリケーション・デザイナー」をクリックします。

アプリケーション WOTRACK を開いたら、「アクションの選択メニュー」→「署名オプションの追加 / 変更」をクリックして新規アプリケーション・オプションを作成できます。

このシナリオのために作成した作業指示書管理アプリケーション (WOTRACK) のアプリケーション・オプションを表 3-9 に示します。

表 3-9 新規 SIGOPTION

オプション	説明	署名の詳細オプション
RBOPTION1	IBM Redbooks Opt1	なし

ユーザー・インターフェース・コントロールの条件付きプロパティーを構成するには、最初にコントロールを選択し、アプリケーション・デザイナーのツールバーにある「コントロールのプロパティー」ボタンをクリックします(65 ページの図 3-6 参照)。

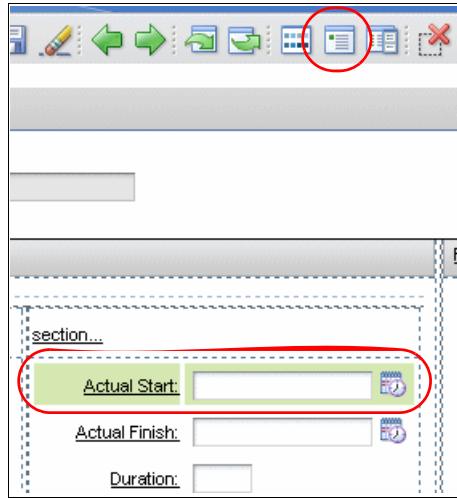


図 3-6 コントロールのプロパティの操作

「実際の開始 (Actual Start)」フィールドと「実際の終了 (Actual Finish)」フィールドの両方に条件構成を追加する必要があります。このため、「作業指示書」タブの最初のフィールドを選択してから「コントロールのプロパティ」ツールバー・ボタンをクリックします。コントロール条件構成を表示するには、ダイアログの「条件付きプロパティの構成」ボタンをクリックします。このフィールドのために作成した構成を表 3-10 に示します。

表 3-10 「実際の開始 (Actual Start)」フィールドの新規コントロール・セキュリティー

フィールド / テーブル	値
署名オプション	RBOPTION1
セキュリティー・グループ	RBGROUP1
セキュリティー・グループ RBGROUP1 の条件	RBOOKCOND1
条件が True の場合のプロパティ値	display - true
条件が False の場合のプロパティ値	display - false

#### 条件

次に、2 番目のフィールドに対しても、表 3-10 に示す値を使用して同じ手順を実行します。

セキュリティー・グループ・アプリケーションに戻り、RBOPTION1へのアクセス権限をグループ RBGROUP1 に付与します。マイグレーション要件での説明に従って、アクセス権限定義をマイグレーションする必要があります。

### 3.2.4 オブジェクト構造

このシナリオの条件付きユーザー・インターフェースのマイグレーションをサポートするために製品と共に提供されているオブジェクト構造を以下に示します。

- ▶ アプリケーション・オプションをサポートする APPSECURITY マイグレーション・グループの DMSIGOPTION オブジェクト構造を図 3-7 に示します。



図 3-7 DMSIGOPTION オブジェクト構造

- ▶ コントロール・セキュリティーをサポートする APPSECURITY マイグレーション・グループの DMCTRLGROUP オブジェクト構造を 66 ページの図 3-8 に示します。

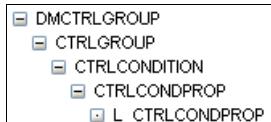


図 3-8 DMCTRLGROUP オブジェクト構造

このシナリオで必要なオブジェクト構造 DMCONDITION および DMMAXGROUP については、55 ページの『3.1.4 オブジェクト構造』を参照してください。

### 3.2.5 マイグレーション・グループ

2 つのマイグレーション・グループに含まれているオブジェクト構造を 66 ページの『3.2.4 オブジェクト構造』に示します。本製品で使用可能なマイグレーション・グループには、その他のマイグレーション・グループへの依存関係が含まれていますが、このシナリオではこの依存関係は必要ありません。

作成した構成を適切かつ効率的にマイグレーションする方法として、このシナリオの目的に対応していることが確認されたオブジェクト構造のみを含む新規マイグレーション・グループを定義する方法があります。

このマイグレーション・グループは、56 ページの『3.1.5 マイグレーション・グループ』で作成したグループと同等です。

### 3.2.6 パッケージ定義

この時点では、データ・セットとマイグレーション・グループの両方が定義されたため、マイグレーション・プロセスを開始できます。最初にマイグレーション・マネージャー・アプリケーションを使用してパッケージを定義します。マイグレーション・マネージャー・アプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」をクリックします。

1 つのスナップショット・パッケージで、作成した条件付きユーザー・インターフェース構成をマイグレーションできます。このパッケージには、3.2.5, 『マイグレーション・グループ』で作成したマイグレーション・グループを含める必要があります。このパッケージの値を表 3-11 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-11 RBUIPKG01 マイグレーション・パッケージの値

フィールド	値
パッケージ定義名	RBUIPKG01
説明	IBM Redbooks UI Security Pkg 01
タイプ	SNAPSHOT
処理アクション	AddChange

「マイグレーション・グループ」タブに、マイグレーション・グループ RBSECURITY01 が追加されました。

### SQL 基準の定義

この例で作成したデータをフィルター処理するためのクエリーを図 3-9 に示します。

Migration Object	Object	Where Clause
DMCONDITION	<b>CONDITION</b>	conditionnum in ('RBOOKCOND1')
DMSIGOPTION	SIGOPTION	app in ('WOTRACK') and optionname in ('RBOP')
DMMAXSERVSI	MAXSERVSECURITY	1=2
DMMAXGROUP	MAXGROUP	groupname in ('RBGROUP1')
DMMAXUSER	MAXUSER	1=2
DMSIGOPTFLAG	SIGOPTFLAG	1=2
DMCTRLGROUP	<b>CTRLGROUP</b>	app in ('WOTRACK') and groupname in ('RBGR')

図 3-9 オブジェクト構造のクエリー

マイグレーション・オブジェクト DMSIGOPTION の Where 文節の内容を以下に示します。

```
app in ('WOTRACK') and optionname in ('RBOPTION1')
```

マイグレーション・オブジェクト DMCTRLGROUP の Where 文節の内容を以下に示します。

```
app in ('WOTRACK') and groupname in ('RBGROUP1') and optionname in ('RBOPTION1')
```

**ヒント:**一部のクエリーは `I=2` と定義されています。その他のセキュリティ・マイグレーション関連シナリオのために新規マイグレーション・グループを作成する必要はありません。別のパッケージを作成し、異なるフィルターを定義してください。

入力された SQL Where 文節が正しくない場合、マイグレーション・マネージャー・アプリケーションからエラー BMXAA7010E が報告されます。パッケージ定義を保存するには、SQL Where 文節を訂正する必要があります。

これらの例の SQL Where 文節をコピーして貼り付ける場合には注意してください。検証エラーを防ぐため、この文節の内容を手動で入力することをお勧めします。

この例のように、既存のアプリケーションの新規オプションを作成した場合、デプロイメントの問題を防ぐため、これらのオプションをフィルター処理する必要があります。新規アプリケーションを作成するか、または既存のアプリケーションを複製し、このアプリケーションのすべてのオプションをマイグレーションする場合、オプションを個別にフィルター処理する必要はありません。以下のようにアプリケーション名を使用してフィルター処理を実行します。

```
app in ('MYNEWAPP1', 'MYNEWAPP2')
```

これにより、両方のアプリケーションのすべてのオプションがフィルター処理されます。追加情報については、70 ページの『3.2.8 デプロイメントに関する考慮事項』を参照してください。

**注:** DMSIGOPTION オブジェクト構造クエリーでは、新規アプリケーションの条件付きユーザー・インターフェース構成をマイグレーションする際に、アプリケーション名およびグループ名を使用してフィルター処理を実行できます。

```
app in ('MYNEWAPP1', 'MYNEWAPP2') and groupname in ('MYGRP1', 'MYGRP2')
```

これにより、グループとアプリケーションのすべてのコントロール・セキュリティーがフィルター処理されます。

### 3.2.7 デプロイメント

パッケージ定義の保存と承認が完了したら、パッケージの作成、配布、およびデプロイに進むことができます。

### 3.2.8 デプロイメントに関する考慮事項

このシナリオでは、UI セキュリティーがソース環境と同様に機能できるようにするために、このセクションで説明する重要なポイントに従ってください。

**注:** 59 ページの『3.1.8 デプロイメントに関する考慮事項』で説明したデプロイメントに関する考慮事項はすべてこのシナリオにも適用されます。

#### 新規アプリケーション

新規アプリケーションの SIGOPTION を作成する場合は、このマイグレーションの前にターゲット環境にアプリケーションが存在することを確認してください。新規アプリケーションのパッケージにこのアプリケーションの SIGOPTION を含めてマイグレーションできます。追加情報については、115 ページの『第 5 章 アプリケーションとスタート・センターのマイグレーション』を参照してください。

#### 新規ユーザー

このシナリオでは新規ユーザーを追加しませんが、52 ページの『3.1 新規セキュリティー・グループのマイグレーション』で説明する手法などを使用して、新規ユーザーを追加できます。

## 3.3 グローバルなデータ制限のマイグレーション

特定のグループ・セットまたはシステム全体に対するデータ制限を作成できます。これらの制限はオブジェクト・レベルまたはアプリケーション・レベルです。グローバルなデータ制限とは、業務上の要件に基づいて作成され、システム内のすべてのグループに適用されるオブジェクト・レベルの制限です。業務上の要件の例を以下に示します。

- ▶ レコードをリストするために使用するアプリケーションまたはユーザー・グループとは関係なく、キャンセルされた作業指示書または終了した作業指示書をすべてのリストからフィルター処理により除外する。
- ▶ レコードをリストするために使用するアプリケーションまたはユーザー・グループとは関係なく、削除されたユーザーをすべてのリストからフィルター処理により除外する。

このシナリオでは、新規グローバル・データ制限とその関連データのマイグレーションを説明します。ここで説明する作業を行うには、管理ユーザーとしてサインインする必要があります。アプリケーション・レベルでのグローバルなデータ制限については、115 ページの『第 5 章 アプリケーションとスタート・センターのマイグレーション』を参照してください。

### 3.3.1 要件

このマイグレーションの要件として、終了された作業指示書およびキャンセルされた作業指示書をすべてのリストからフィルター処理によって除外するための新規グローバル・データ制限を定義する必要があります。グローバルなデータ制限では、このマイグレーション要件でもある新規条件が使用されます。

グローバルなデータ制限と条件は、開発環境で定義され、その後テスト環境および実稼働環境にプロモートされます。

### 3.3.2 ソリューション

定義されている手法では、新規グローバル制限とその関連条件が 2 つのスナップショット・マイグレーション・パッケージでマイグレーションされます。

このシナリオで説明するグローバルなデータ制限のマイグレーションは、以下に示す構成が最初のマイグレーション・パッケージでマイグレーションされることを意味します。

1. オブジェクト構造
- 2 番目のマイグレーション・パッケージの内容を以下に示します。
  1. 条件
  2. セキュリティー制限

**注:** セキュリティー・グループに関連付けられているセキュリティー制限は、そのグループとともにマイグレーションされます。詳しくは、52 ページの『3.1 新規セキュリティー・グループのマイグレーション』を参照してください。

### 3.3.3 構成アプリケーション

以降のセクションでは、このマイグレーション・シナリオの例として作成されたデータについて説明します。

## 条件式マネージャー

条件とは、データ・アクセスやデータ編集を制限する場合などに使用される SQL 式またはカスタム Java クラスのいずれかです。条件によって定義される制限は、条件式が True と評価される場合にのみ適用されます。条件の作成および管理には、条件式マネージャー・アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「管理」→「条件式マネージャー」をクリックします。

このシナリオのために作成した条件とその値を表 3-12 に示します。

表 3-12 RBOOKCOND2

フィールド	値
条件	RBOOKCOND2
説明	IBM Redbooks Condition 2
タイプ	EXPRESSION
オブジェクト名	WORKORDER
式	:status = 'CLOSE' または :status = 'CAN'
常に評価	チェック

## セキュリティー・グループ

グローバルなデータ制限の作成および管理には、セキュリティー・グループ・アプリケーションを使用します。セキュリティー・グループ・アプリケーションを使用するには、「リンク先」→「セキュリティー」→「セキュリティー・グループ」をクリックし、さらに「アクションの選択」→「グローバルなデータ制限」をクリックします。

このシナリオのために作成したオブジェクト・データ制限とその値を 72 ページの表 3-13 に示します。

表 3-13 オブジェクト制限

フィールド	値
オブジェクト	WORKORDER
タイプ	HIDDEN
再評価	チェック
条件	RBOOKCOND2

**重要:** 同一条件を複数のデータ制限に使用できます。この例では「オブジェクト」フィールドに WORKORDER を設定することで、この関連付けが作成されます。

この条件を適用できる他のオブジェクトに、この条件を関連付けることができます。

### 3.3.4 オブジェクト構造

このシナリオでは、グローバルなデータ制限のための新規オブジェクト構造を作成する必要があります。新規オブジェクト構造の作成には、オブジェクト構造アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」をクリックします。

グローバルなデータ制限の構成は、テーブル SECURITYRESTRICT に現在含まれています。このテーブルは、オブジェクト構造 DMMAXGROUP の下位階層の 1 つとして定義されています。このオブジェクト構造内には、セキュリティ・グループと、このグループに基づくデータ制限の間の関係がありますが、これはこのシナリオの要件ではありません。

使用できる事前定義オブジェクト構造が本製品に含まれていないことから、マイグレーションを実現するために、表 3-14 に示す値を使用して新規オブジェクト構造を作成しました。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-14 DMSECURITYRESTRICT オブジェクト構造

フィールド	値
オブジェクト構造	DMSECURITYRESTRICT
説明	グローバル制限オブジェクト構造
使用者	MIGRATIONMGR

「ソース・オブジェクト」タブに、SECURITYRESTRICT オブジェクトが追加されました。

このシナリオで必要な DMCONDITION オブジェクト構造については、55 ページの『3.1.4 オブジェクト構造』を参照してください。

### 3.3.5 マイグレーション・グループ

作成したデータを適切かつ効率的にマイグレーションする方法として、2つの新規マイグレーション・グループを定義する方法があります。

最初のマイグレーション・グループの値を表 3-15 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-15 RBGLOBALSE01 マイグレーション・グループ

フィールド	値
マイグレーション・グループ	RBGLOBALSE01
説明	IBM Redbooks Global Secur Obj Structure

このグループには、マイグレーション・オブジェクト DMMAXINTOBJECT のみが含まれています。このグループのマイグレーション・ツリーを図 3-10 に示します。

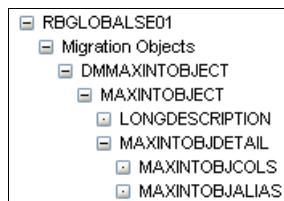


図 3-10 RBGLOBALSE01 マイグレーション・グループ・ツリー

2 番目のマイグレーション・グループの値を表 3-16 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-16 RBGLOBALSE02 マイグレーション・グループ

フィールド	値
マイグレーション・グループ	RBGLOBALSE02
説明	IBM Redbooks Global Security Data

このグループには、オブジェクト構造 DMCONDITION および DMSECURITYRESTRICT が含まれています(75 ページの図 3-11 参照)。

Migration Object	Description	Migration Object Order	Internal
DMCONDITION >>	Condition expression migration object for Migra	1	<input type="checkbox"/>
DMSECURITYREF >>	Global Restrictions Object Structure	2	<input type="checkbox"/>

図 3-11 RBGLOBALSE02 マイグレーション・グループ

このグループのマイグレーション・ツリーを図 3-12 に示します。

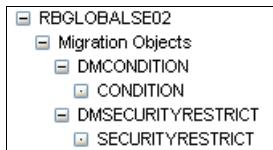


図 3-12 RBGLOBALSE02 マイグレーション・グループ・ツリー

### 3.3.6 パッケージ定義

この時点でのデータ・セットとマイグレーション・グループの両方が定義されたため、マイグレーション・プロセスを開始できます。最初にマイグレーション・マネージャー・アプリケーションを使用してパッケージを定義します。マイグレーション・マネージャー・アプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」をクリックします。

新規オブジェクト構造をマイグレーションするにあたり、グローバルなデータ制限のマイグレーションのために 2 つのスナップショット・パッケージが必要です。最初のパッケージでは新規オブジェクト構造がマイグレーションされ、2 番目のパッケージではセキュリティ制限がマイグレーションされます。

**注:** 新規オブジェクト構造をマイグレーションせずに、ターゲット環境でのオブジェクト構造を直接作成することもできます。追加情報については、77 ページの『3.3.8 デプロイメントに関する考慮事項』を参照してください。

最初のパッケージの値を表 3-17 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-17 RBGLOBALSEC01PKG マイグレーション・パッケージの値

フィールド	値
パッケージ定義名	RBGLOBALSEC01PKG

フィールド	値
説明	IBM Redbooks Global Security object structure
タイプ	SNAPSHOT
処理アクション	AddChange

「マイグレーション・グループ」タブに、マイグレーション・グループ RBGLOBALSE01 が追加されました。

2 番目のパッケージの値を 76 ページの表 3-18 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-18 RBGLOBALSEC02PKG マイグレーション・パッケージの値

フィールド	値
パッケージ定義名	RBGLOBALSEC02PKG
説明	IBM Redbooks Global Security Data
タイプ	SNAPSHOT
処理アクション	AddChange

「マイグレーション・グループ」タブに、マイグレーション・グループ RBGLOBALSE02 が追加されました。

## SQL 基準の定義

パッケージ RBGLOBALSEC01PKG のデータのフィルター処理に使用したクエリを図 3-13 に示します。

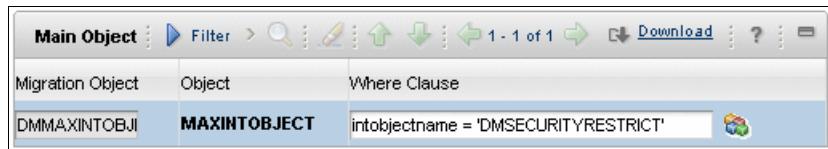


図 3-13 パッケージ RBGLOBALSEC01PKG のクエリー

パッケージ RBGLOBALSEC02PKG のデータのフィルター処理に使用したクエリを 77 ページの図 3-14 に示します。

Main Object	Object	Where Clause
Migration Object	Object	
DMCONDITION	CONDITION	conditionnum in ('RBOOKCOND2')
DMSECURITYRESTRICT	<b>SECURITYRESTRICT</b>	groupname is null and app is null and conditionnum in ('RBOOKCOND2')

図 3-14 RBGLOBALSEC02PKG オブジェクト構造のクエリー

DMSECURITYRESTRICT オブジェクト構造をフィルター処理するクエリー全体を以下に示します。

```
groupname is null and app is null and conditionnum in ('RBOOKCOND2')
```

### 3.3.7 デプロイメント

2つのパッケージ定義の保存と承認が完了したら、パッケージの作成、配布、およびデプロイに進むことができます。パッケージはそれぞれ個別に次の順序でマイグレーションする必要があります。

1. RBGLOBALSEC01PKG
2. RBGLOBALSEC02PKG

### 3.3.8 デプロイメントに関する考慮事項

このシナリオでは、グローバル・セキュリティーがソース環境と同様に機能できるようにするために、このセクションで説明する重要なポイントに従ってください。

#### 管理モード

システム接続ユーザーに対して副作用が及ばないようにするために、パッケージのデプロイ前にターゲット環境で管理モードをオンにしておく必要があります。

#### 新規オブジェクト構造のマイグレーション

グローバル・セキュリティーのマイグレーション前に、作成した新規オブジェクト構造をターゲット環境にマイグレーションする必要があります。パッケージ作成中に、データ・ディクショナリーをマイグレーションするかどうか尋ねるプロンプトが出されます。これは新規カスタム・オブジェクト構造をマイグレーションするたびに表示される警告メッセージであるため、操作を続行してください。マイグレーションする必要があるデータ・ディクショナリーの保留中の変更がないことを前提としています。

グローバル・セキュリティ構成を含むパッケージをデプロイする前に、新規オブジェクト構造を含むパッケージをデプロイする必要があります。

新規オブジェクト構造が含まれているパッケージをマイグレーションする代わりに、ターゲット環境でこのオブジェクト構造を手動で作成し、2番目のパッケージのみをマイグレーションすることができます。

## グローバルな属性制限のマイグレーション

このセクションでは、グローバルな属性制限のマイグレーションについて説明します。属性制限はオブジェクト制限と同じテーブルに格納されているため、オブジェクト制限と同様の方法でマイグレーションできます。唯一の相違点は、定義したクエリーが、条件 RBOOKCOND2 に基づいてフィルター処理を行う点です。この条件を参照する属性レベルの制約がある場合は、その制約もマイグレーションされます。

## 条件式の依存関係

グローバル・セキュリティ制限は式の条件と関連しています。このシナリオで示すマイグレーション・パッケージに含まれていない条件は、グローバル制限パッケージのマイグレーション前に、ターゲット環境に存在している必要があります。

## 3.4 アクセス権限定義および LDAP 情報のマイグレーション

本製品での認証を行うには、システムのネイティブ認証を使用するか、またはアプリケーション・サーバー・セキュリティおよび *Lightweight Directory Access Protocol (LDAP)* を使用します。

LDAP ディレクトリー・サーバーに接続したアプリケーション・サーバー・セキュリティを使用して認証を実行するようにシステムを構成するときには、ディレクトリー・サーバーで変更されたグループおよびユーザーを使用してシステムを更新する仮想メンバー管理と呼ばれるクーロン・タスクがあります。

グループとユーザーをシステムにロードしたら、これらのグループとユーザーのアクセス権限定義を構成できます。これは、LDAP ベースの認証を使用する場合に非常に一般的なシナリオです。

ここで説明する作業を行うには、管理ユーザーとしてサインインする必要があります。

### 3.4.1 要件

このシナリオの要件として、アプリケーション・オプションのアクセス権限定義のみをマイグレーションする必要があります。

アクセス権限定義は、開発環境で定義され、その後テスト環境および実稼働環境にプロモートされます。

**重要 :** LDAP 同期が設定されており、アプリケーション・サーバー・セキュリティーが有効な場合、セキュリティー・グループとユーザーのマイグレーションに、マイグレーション・マネージャー・アプリケーションは使用されません。

このプロセスでは外部ツールが使用されるため、本書では説明しません。

### 3.4.2 ソリューション

定義されている手法では、新規アクセス権限定義とその関連条件が 2 つのスナップショット・マイグレーション・パッケージでマイグレーションされます。

このシナリオで説明するアクセス権限定義のマイグレーションは、以下の構成が最初のマイグレーション・パッケージでマイグレーションされることを意味します。

- ▶ オブジェクト構造

2 番目のマイグレーション・パッケージの内容を以下に示します。

- ▶ 権限

### 3.4.3 構成アプリケーション

以降のセクションでは、このマイグレーション・シナリオの例として作成されたデータについて説明します。

#### セキュリティー・グループ

このシナリオでは、53 ページの『セキュリティー・グループ』で作成したグループを使用します。このコンテキストでは、これは LDAP サーバー・ディレクトリーからロードしたグループです。

**重要:** このグループがターゲット環境に存在しており、アクセス権限定義がないことを前提とします。追加情報については、3.4.8、『デプロイメントに関する考慮事項』を参照してください。

「アプリケーション」タブで、担当者アプリケーションのすべてのオプションへのアクセス権限がこのグループに付与されます。

#### 3.4.4 オブジェクト構造

このシナリオでは、アクセス権限定義のための新規オブジェクト構造を作成する必要があります。新規オブジェクト構造の作成には、オブジェクト構造アプリケーションを使用します。このアプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」をクリックします。

データ・アクセス権限の構成は、現在テーブル APPLICATIONAUTH に含まれています。このテーブルは、オブジェクト構造 DMAXGROUP で下位階層の 1 つとして定義されています。このオブジェクト構造内には、セキュリティー・グループと、このグループに基づくアクセス権限定義の間の関係がありますが、これはこのシナリオの要件ではありません。

使用できる事前定義オブジェクト構造が本製品に含まれていないことから、マイグレーションを実現するために、表 3-19 に示す値を使用して新規オブジェクト構造を作成しました。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-19 DMAPPLICATIONAUTH オブジェクト構造

フィールド	値
オブジェクト構造	DMAPPLICATIONAUTH
説明	権限
使用者	MIGRATIONMGR

「ソース・オブジェクト」タブに APPLICATIONAUTH オブジェクトが追加されました。

#### 3.4.5 マイグレーション・グループ

作成したデータを適切かつ効率的にマイグレーションする方法として、2 つの新規マイグレーション・グループを定義する方法があります。

最初のマイグレーション・グループの値を表 3-20 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-20 RBAUTHORIZATIONS01 マイグレーション・グループ

フィールド	値
マイグレーション・グループ	RBAUTHORIZATIONS01
説明	IBM Redbooks Authorizations Obj Structure

このグループにはオブジェクト構造 DMMAXINTOBJECT のみが含まれています。

このグループのマイグレーション・ツリーを図 3-15 に示します。

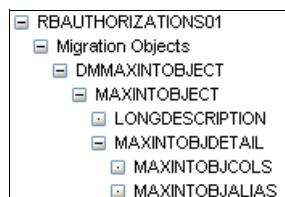


図 3-15 RBAUTHORIZATIONS01 マイグレーション・グループ・ツリー

2 番目のマイグレーション・グループの値を表 3-21 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-21 RBAUTHORIZATIONS02 マイグレーション・グループ

フィールド	値
マイグレーション・グループ	RBAUTHORIZATIONS02
説明	IBM Redbooks Access Definitions Data

このグループには、オブジェクト構造 DMSECURITYRESTRICT のみが含まれています。このグループのマイグレーション・ツリーを 81 ページの図 3-16 に示します。



図 3-16 RBAUTHORIZATIONS02 マイグレーション・グループ・ツリー

### 3.4.6 パッケージ定義

この時点で、データ・セットとマイグレーション・グループの両方が定義されたため、マイグレーション・プロセスを開始できます。最初にマイグレーション・マネージャー・アプリケーションを使用してパッケージを定義します。マイグレーション・マネージャー・アプリケーションを使用するには、「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」をクリックします。

データ・アクセス権限をマイグレーションするために2つのスナップショット・パッケージが必要です。最初のパッケージでは作成した新規オブジェクト構造がマイグレーションされ、2番目のパッケージではアクセス権限定義と条件がマイグレーションされます。

**注:** 新規オブジェクト構造をマイグレーションせずに、ターゲット環境でこのオブジェクト構造を直接作成することもできます。追加情報については、84ページの『3.4.8 デプロイメントに関する考慮事項』を参照してください。

最初のパッケージの値を表 3-22 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-22 RBAUTH01PKG マイグレーション・パッケージの値

フィールド	値
パッケージ定義名	RBAUTH01PKG
説明	IBM Redbooks Auth object structure
タイプ	SNAPSHOT
処理アクション	AddChange

「マイグレーション・グループ」タブに、マイグレーション・グループ RBAUTHORIZATIONS01 が追加されました。

2番目のパッケージの値を表 3-23 に示します。その他のすべてのフィールドにはデフォルト値が入ります。

表 3-23 RBAUTH02PKG マイグレーション・パッケージの値

フィールド	値
パッケージ定義名	RBAUTH02PKG
説明	IBM Redbooks Auth data

フィールド	値
タイプ	SNAPSHOT
処理アクション	AddChange

「マイグレーション・グループ」タブに、マイグレーション・グループ RBAUTHORIZATIONS02 が追加されました。

### SQL 基準の定義

パッケージ RBAUTH01PKG のデータのフィルター処理に使用したクエリーを図 3-17 に示します。

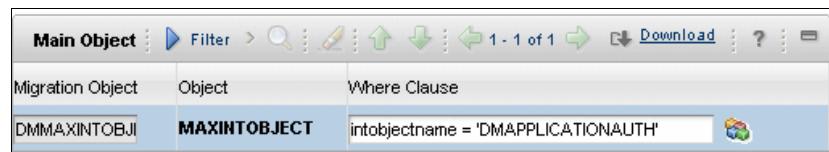


図 3-17 パッケージ RBAUTH01PKG のクエリー

パッケージ RBAUTH02PKG のデータのフィルター処理に使用したクエリーを 83 ページの図 3-18 に示します。



図 3-18 パッケージ RBAUTH02PKG のクエリー

DMAPPLICATIONAUTH マイグレーション・オブジェクトをフィルター処理するクエリーを以下に示します。

```
app = 'PERSON' and groupname in ('RBGROUP')
```

### 3.4.7 デプロイメント

2 つのパッケージ定義の保存と承認が完了したら、パッケージの作成、配布、およびデプロイに進むことができます。パッケージはそれぞれ個別に次の順序でマイグレーションする必要があります。

1. RBAUTH01PKG
2. RBAUTH02PKG

### 3.4.8 デプロイメントに関する考慮事項

このシナリオでは、グローバル・セキュリティーがソース環境と同様に機能できるようにするために、このセクションで説明する重要なポイントに従ってください。

#### 管理モード

システム接続ユーザーに対して副作用が及ばないようにするために、パッケージのデプロイ前にターゲット環境で管理モードをオンにしておく必要があります。

#### 新規オブジェクト構造のマイグレーション

アクセス権限定義のマイグレーション前に、作成した新規オブジェクト構造をターゲット環境にマイグレーションする必要があります。パッケージ作成中に、データ・ディクショナリーをマイグレーションするかどうかを尋ねるプロンプトが出されます。これは新規カスタム・オブジェクト構造をマイグレーションするたびに表示される警告メッセージであるため、操作を続行してください。マイグレーションする必要があるデータ・ディクショナリーの保留中の変更がないことを前提としています。

データ・アクセス権限構成を含むパッケージをデプロイする前に、新規オブジェクト構造を含むパッケージをデプロイする必要があります。

新規オブジェクト構造が含まれているパッケージをマイグレーションする代わりに、ターゲット環境でこのオブジェクト構造を手動で作成することができます。

#### メタデータに関する前提事項

製品構成で、APPLICATIONAUTH テーブルの 1 次キーが変更されておらず、またオブジェクトが拡張されていないことを前提としています。

#### アプリケーションおよびセキュリティー・グループの依存関係

アプリケーション定義、セキュリティー・グループ、およびアクセス権限を定義したアプリケーション・オプションが、ターゲット環境に存在している必要があります。

#### 条件式の依存関係

アクセス権限定義は式の条件と関連しています。条件がマイグレーション・パッケージに含まれていない場合、アクセス権限定義パッケージのマイグレーション前に、これらの条件がターゲット環境に存在している必要があります。

## **アクセス権限定義のマイグレーションに関する考慮事項**

このシナリオでは、アクセス権限定義の初めてのマイグレーションについて説明しますが、この場合マイグレーションするデータがターゲット環境に存在していません。

ソース環境を変更し、新規マイグレーションを実行する必要がある場合は、このセクションで説明する方法とほぼ同じ方法を使用できます。ただし、「置換」処理アクションの使用を検討するか、または変更パッケージを使用してください。追加情報については、211 ページの『第 10 章 一般的なトピック』を参照してください。

## **3.5 セキュリティー・マイグレーションに関する考慮事項**

この章では、さまざまなセキュリティー・マイグレーションのシナリオについて説明しました。各シナリオは個別に説明され、依存関係はありませんが、これらのシナリオを組み合わせるかまたは拡張して、新たなマイグレーション・シナリオを作成できます。





## エスカレーションのマイグレーション

エスカレーションとは、Tivoli's process automation engine (Tivoli/Maximo Base Service) ベースの任意の製品でアクションおよび通知を自動化するサーバー・サイド機能のことです。クライアントや実務担当者は、エスカレーションを幅広く使用することにより、Tivoli's process automation engine (Tivoli/Maximo Base Service) ベースの製品によってサポートされている業務のさまざまな側面を自動化できます。例えば、エスカレーションにより、サービス・デスクがサービス・レベル契約 (SLA) の確約に準拠しているかどうかを事前にモニターします。優先度の高いサービス要求 (SR) の応答時間が、確約に定義されている SLA しきい値に近づくと、エスカレーションによってアクションおよび通知が実行されます。通知によってサービス・デスク管理者に、非準拠に近づいているというアラートが出され、アクションによって SR の重大度が引き上げられることがあります。

この章では、エスカレーションをマイグレーションする場合のベスト・プラクティスについて説明します。この章には以下のセクションがあります。

- ▶ 89 ページの『4.1 要件』
- ▶ 89 ページの『4.2 ソリューション』
- ▶ 89 ページの『4.3 構成アプリケーション』
- ▶ 92 ページの『4.4 オブジェクト構造』
- ▶ 94 ページの『4.5 マイグレーション・グループ』
- ▶ 97 ページの『4.6 パッケージ定義』

- ▶ 104 ページの『4.7 デプロイメント』
- ▶ 104 ページの『4.7.1 エスカレーションとクーロン・タスク』

## 4.1 要件

エスカレーションは、開発環境で定義され、その後、テスト環境および実稼働環境にプロモートされます。エスカレーション構成および関連するすべての構成を、(エラーや障害が発生することなく)高い信頼性で、(必須のコンテンツが欠落することなく)正確にマイグレーションするには、反復可能な処理が必要です。

## 4.2 ソリューション

定義されている手法では、エスカレーション構成および関連するすべての構成が1つのスナップショット・マイグレーション・パッケージでマイグレーションされます。

コンテンツの観点から見ると、エスカレーションのマイグレーションでは、主に以下の構成のマイグレーションを順序どおりにマイグレーションする必要があります。

1. アクション
2. アクション・グループ
3. 担当者
4. 担当者グループ
5. ロール
6. 通信テンプレート
7. エスカレーション

## 4.3 構成アプリケーション

このセクションでは、エスカレーション構成をサポートするアプリケーションをリストします。また、エスカレーションのマイグレーションを推進するこれらのアプリケーション間の依存関係についても説明します。

### 4.3.1 アクション

アクションおよびアクション・グループにアクセスするには、該当する製品のスタート・センターで、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アクション」をクリックします。

### 4.3.2 担当者

ロールまたは通信テンプレートを担当者レコードに関連付けた場合、担当者レコードの作成および管理は、担当者アプリケーションを使用して実行します。担当者レコードにアクセスするには、該当する製品のスタート・センターで、「リンク先」→「管理」→「リソース」→「担当者」をクリックします。

### 4.3.3 担当者グループ

担当者グループ・レコードの作成および管理は、担当者グループ・アプリケーションを使用して実行します。担当者グループにアクセスするには、該当する製品のスタート・センターで、「リンク先」→「管理」→「リソース」→「担当者グループ」をクリックします。

ここまでに確認した構成および管理側アプリケーションは、エスカレーションの主要な依存関係を構成します。

### 4.3.4 ロール

通信テンプレートをロールに関連付けた場合、ロールはロール・アプリケーションを使用して定義します。ロールにアクセスするには、該当する製品のスタート・センターで、「リンク先」→「システムの構成」→「プラットフォームの構成」→「ロール」をクリックします。

ロールは、担当者レコードまたは担当者グループ・レコードから作成できます。

### 4.3.5 通信テンプレート

通知は、通信テンプレート・アプリケーションを使用して管理されます。通信テンプレートにアクセスするには、該当する製品のスタート・センターで、「リンク先」→「システムの構成」→「プラットフォームの構成」→「通信テンプレート」をクリックします。

通信テンプレートでは、電子メール・アドレス、ロール、担当者、または担当者グループの各レコードの形式で、1人以上の受信者を定義できます。

### 4.3.6 エスカレーション

エスカレーションは、エスカレーション・アプリケーションを使用して作成および管理されます。エスカレーションにアクセスするには、該当する製品のスタート・センターで、「リンク先」→「システムの構成」→「プラットフォームの構成」→「エスカレーション」をクリックします。

エスカレーションでは、アクションおよび通知が1回以上実行されます。このため、すべてのエスカレーションは、対応する一連のアクションまたは通知に関連付けられます。

#### 4.3.7 その他のアプリケーション

前述したアプリケーションのほかに、エスカレーションは、一連の追加構成にも依存します。これらの構成は、製品と共に提供されていた場合があることや、以前にマイグレーションされていた場合があるという意味で2次的なものです。これらの2次構成について、以下に説明します。

- ▶ クーロン・タスク

エスカレーションでは、エスカレーションを定期的に実行するために、常にクーロン・タスク・インスタンスが必要です。クーロン・タスクは、**クーロン・タスクのセットアップ・アプリケーション**を使用して定義します。すべてのエスカレーション・クーロン・タスク・インスタンスは、製品と共に提供されている**ESCALATION**クーロン・タスクから派生します。エスカレーションをアクティビ化すると、クーロン・タスク・インスタンスが作成されます(存在しない場合)。クーロン・タスク・インスタンスを動的に作成することは、エスカレーションに関連付けられているクーロン・タスク・インスタンスをマイグレーションする別の方法があることを示しています。これらの方法については、このシナリオの「デプロイメントに関する考慮事項」セクションで説明します。

- ▶ 組織およびサイト

エスカレーションでは、オプションで組織またはサイトへの関連付けを指定できます。組織およびサイトは、**組織**アプリケーションを使用して定義します。このマイグレーション・シナリオは、組織およびサイトは既にマイグレーション済みであることを前提としています。

- ▶ ビジネス・オブジェクト

エスカレーションでは、エスカレーションのアクションおよび通知のターゲットであるオブジェクトが必ず指定されます。実際に、ロール、通信テンプレート、およびエスカレーションは、すべてビジネス・オブジェクトが存在することに依存しています。ビジネス・オブジェクトは、**データベースの構成**アプリケーションを使用して定義します。ロール、通信テンプレート、およびエスカレーションの大多数は、製品と共に提供されているビジネス・オブジェクトに対して定義されています。このマイグレーション・シナリオは、ビジネス・オブジェクト定義は既にマイグレーション済みであることを前提としています。

これで、エスカレーションを単独で作成するのではなく、他のアプリケーションを介して管理されている構成と組み合わせて作成するように設定できました。エスカレーションのマイグレーションでは、これらの関連構成をすべて考慮に入れる必要があります。92ページの図 4-1 に、これらの依存関係を示します。

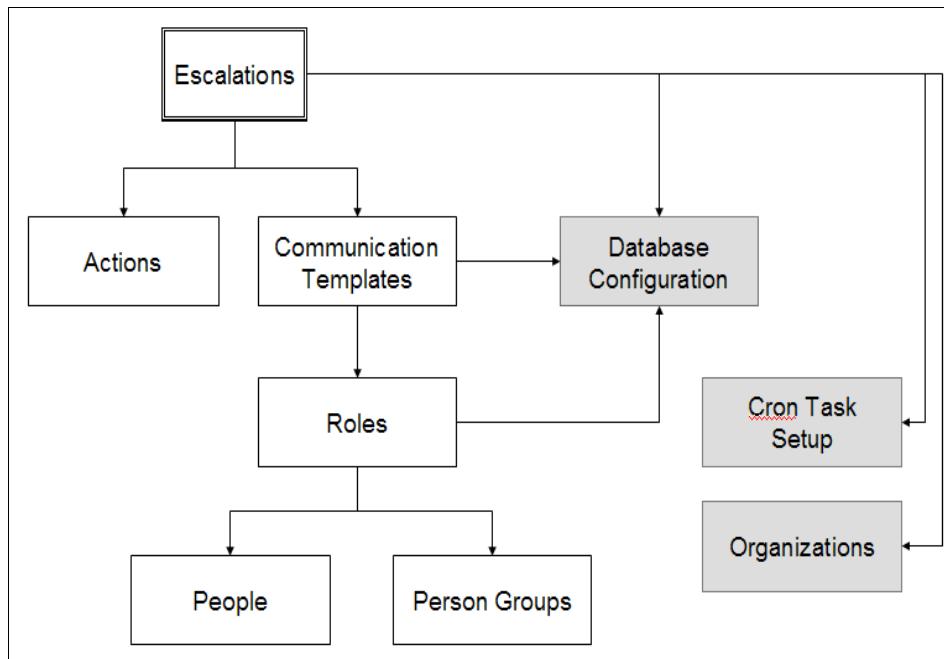


図 4-1 エスカレーションと関連アプリケーション

## 4.4 オブジェクト構造

以下のオブジェクト構造は、製品と共に提供されています。

- ▶ アクション・グループのマイグレーションをサポートする **BPM** マイグレーション・グループの一部である **DMACTION** オブジェクト構造。図 4-2 に、**DMACTION** オブジェクト構造のコンテンツを示します。

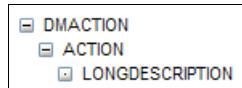


図 4-2 DMACTION オブジェクト構造

- ▶ アクションのマイグレーションをサポートする **BPM** マイグレーション・グループの一部である **DMACTIONGROUP** オブジェクト構造。93 ページの図 4-3 に、**DMACTIONGROUP** オブジェクト構造のコンテンツを示します。

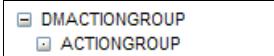


図 4-3 DMACTIONGROUP オブジェクト構造

- ▶ 担当者構成情報のマイグレーションをサポートする **RESOURCES** マイグレーション・グループの一部である **DMPERSON** オブジェクト構造。図 4-4 に、**DMPERSON** オブジェクト構造のコンテンツを示します。

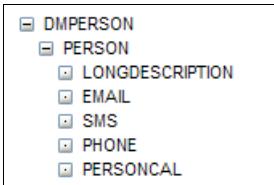


図 4-4 DMPERSON オブジェクト構造

- ▶ 担当者グループのマイグレーションをサポートする **RESOURCES** マイグレーション・グループの一部である **DMPERSONGROUP** オブジェクト構造。図 4-5 に、**DMPERSONGROUP** オブジェクト構造のコンテンツを示します。

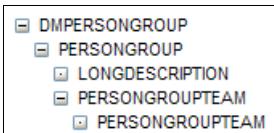


図 4-5 DMPERSONGROUP オブジェクト構造

- ▶ ロールのマイグレーションをサポートする **BPM** マイグレーション・グループの一部である **DMROLE** オブジェクト構造。図 4-6 に、**DMROLE** オブジェクト構造のコンテンツを示します。

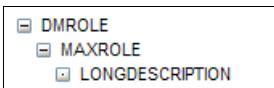


図 4-6 DMROLE オブジェクト構造

- ▶ 通信テンプレートのマイグレーションをサポートする **BPM** マイグレーション・グループの一部である **DMCOMMTEMPLATE** オブジェクト構造。94 ページの図 4-7 に、**DMCOMMTEMPLATE** オブジェクト構造のコンテンツを示します。



図 4-7 DMCOMMTEMPLATE オブジェクト構造

- ▶ エスカレーションのマイグレーションをサポートする **BPM** マイグレーション・グループの一部である **DMESCALATION** オブジェクト構造。図 4-8 に、**DMESCALATION** オブジェクト構造のコンテンツを示します。

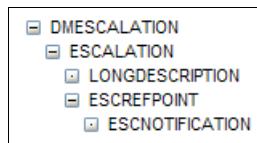


図 4-8 DMESCALATION オブジェクト構造

## 4.5 マイグレーション・グループ

エスカレーションのマイグレーションをサポートするオブジェクト構造は、以下の 2 つのマイグレーション・グループに属しています。

- ▶ BPM
- ▶ RESOURCES

95 ページの図 4-9 に、製品と共に提供されている BPM マイグレーション・グループとその依存関係を示します。

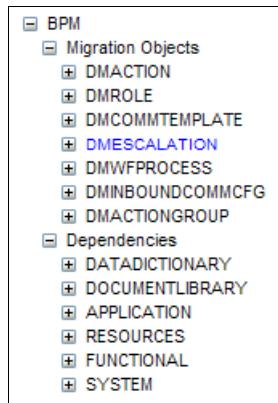


図 4-9 BPM マイグレーション・グループ

図 4-10 に、製品と共に提供されている RESOURCES マイグレーション・グループとその依存関係を示します。

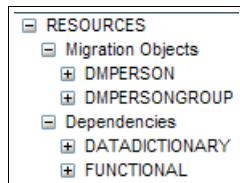


図 4-10 RESOURCES マイグレーション・グループ

以下の理由により、BPM マイグレーション・グループと RESOURCES マイグレーション・グループを使用するのは非効率的です。

- ▶ マイグレーションが必要なのは、構成の一部分のみであるため
- ▶ この 2 つのマイグレーション・グループと共にリストされている依存関係に属する構成は、既にマイグレーション済みになっている場合があるため

こうした理由から、BPM に属するオブジェクト構造およびマイグレーション・グループのうち、エスカレーションのマイグレーション・シナリオには必要なないものを 96 ページの図 4-11 に示します。

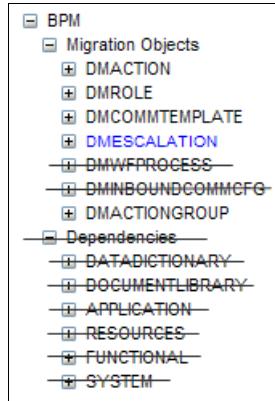


図 4-11 不要な BPM オブジェクト構造および依存関係

**注:**96 ページの図 4-11 では、説明の便宜上、取り消し線を使用しています。製品には、ここに示すような視覚的な方法でオブジェクト構造およびマイグレーション・グループを削除する機能はありません。

図 4-12 に、RESOURCES に属するオブジェクト構造およびマイグレーション・グループのうち、このマイグレーション・シナリオには必要のないものを示しています。

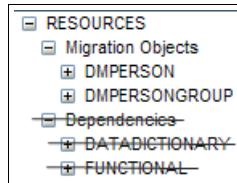


図 4-12 不要な RESOURCES 依存関係

エスカレーションのマイグレーション・シナリオの場合は、必要なオブジェクト構造のみが含まれる新規のマイグレーション・グループを作成します。新規のマイグレーション・グループは、既存の BPM マイグレーション・グループを複製してそのコンテンツを変更することにより、簡単に作成できます。97 ページの図 4-13 に、新規の MYESCALATION マイグレーション・グループとそのコンテンツを示します。

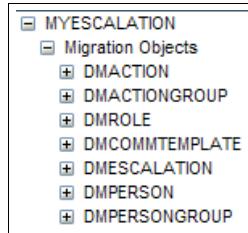


図 4-13 エスカレーションのマイグレーションをサポートするオブジェクト構造およびマイグレーション・グループ

図 4-14 に、この新規マイグレーション・グループ内のオブジェクト構造の順序付けを示します。

Migration Object	Description	Migration Object Order
DMACTION >>	Action migration object for Migration Manager	1
DMACTIONGRC >>	Actiongroup migration object for Migration Man	2
DMPERSON >>	Person migration object for Migration Manager	3
DMPERSONGR(>>	Person group migration object for Migration Ma	4
DMROLE >>	Role migration object for Migration Manager	5
DMCOMMTEMPI >>	Communication template migration object for Mi	6
DMESCALATIO >>	Escalation migration object for Migration Manag	7

図 4-14 オブジェクト構造の順序

注：97 ページの図 4-14 は、新規のマイグレーション・グループ内のオブジェクト構造とその順序付けを説明するために使用したもので、それ以外の目的はありません。製品のユーザー・インターフェースとは異なる場合があります。

## 4.6 パッケージ定義

コンテンツを確認して整理したら、マイグレーションを開始できます。最初のステップとして、スナップショット・マイグレーション・パッケージを定義します。この定義には、新規に作成したマイグレーション・グループが含まれます。SQL 基準は、新規のマイグレーション・グループのメンバーである適切なオブジェクト構造と関連付けられます。

**注:**マイグレーション・マネージャー・アプリケーションがパッケージ定義に対して提示しているすべてのデフォルト値をそのまま設定してください。

以下のセクションでは、製品と共に提供されているサンプル・エスカレーションを使用して、SQL 基準の設定を説明します。

#### 4.6.1 アクションが関連付けられているエスカレーションの SQL 基準の定義

この説明では、サンプル・エスカレーションである INCCLOSE を使用します。このエスカレーションは、製品のインシデント・レコードのうち、10 日間より長く解決済み状態のままになっているレコードを終了することを目的としています。図 4-15 に、INCCLOSE エスカレーションのエスカレーション・ヘッダーを示します。

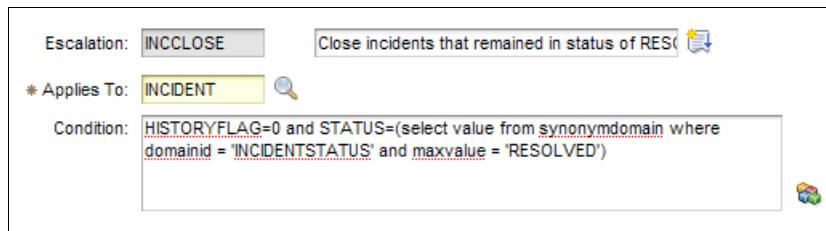


図 4-15 INCCLOSE サンプル・エスカレーション

INCCLOSE エスカレーションには、アクションを実行するエスカレーション・ポイントが 1 つありますが、関連付けられた通知はありません。図 4-16 に、エスカレーションに関連付けられている INCIDENT CLOSED アクションを示します。

Action	Description	Type
INCIDENT CLOS	TIDE Incident Closed Action	CHANGESTATUS

図 4-16 INCIDENT CLOSE アクション

このアクションは、アクション・グループの一部です。アクション・グループを使用すると、エスカレーションの実行時に複数のアクションを順番に実行できます。

**注:** エスカレーションは、個別のアクションを1つ以上含むアクション・グループに必ず関連付けられます。

99ページの図4-17に、INCIDENT CLOSED アクションがメンバーになっているINCCLOSEGRP アクション・グループを示します。

Actions		Filter >     1 - 1 of 1	Down
	Action	Description	
	INCIDENT CLOS >>	TIDE Incident Closed Action	

図4-17 INCCLOSEGRP アクション・グループとそのメンバーの INCIDENT CLOSED アクション

INCCLOSE エスカレーションのマイグレーションには、関連付けられているアクション INCIDENT CLOSED およびアクション・グループ INCCLOSEGRP のマイグレーションが含まれます。このため、SQL 基準を、マイグレーション・パッケージの一部であるオブジェクト構造に対して設定できます。図4-18に、エスカレーション、アクション、およびアクション・グループの各オブジェクト構造に対して設定されている個別の SQL 基準を示します。

Main Object	Object	Where Clause
DMACTION	ACTION	action in (select action from ACTIONGROUP w)
DMACTIONGRC	ACTIONGROUP	action in (select action from ESCREFPOINT wh)
DMPERSON	PERSON	1=0)
DMPERSONGR	PERSONGROUP	1=0)
DMROLE	MAXROLE	1=0)
DMCOMMTEMPI	COMMTEMPLATE	1=0)
DMESCALATIO	ESCALATION	escalation in ('INCCLOSE'))

図4-18 INCCLOSE エスカレーションをマイグレーションするための SQL 基準

DMACTION オブジェクト構造に対する完全な SQL Where 文節は、以下のとおりです。

```
action in (select action from ACTIONGROUP where action in (select
action from ESCREFPOINT where escalation in      ('INCCLOSE') ) ) or
```

```
action in (select member from actiongroup where action in (select  
action from esrefpoint where escalation in ('INCCLOSE')))
```

**注 :** DMACTION オブジェクト構造の SQL Where 文節は、エスカレーションに必要なアクション・グループと個々のアクションの両方を収集することを目的としています。アクション・グループおよび個々のアクションの項目は、基礎となる製品データベースの同じ ACTION テーブルにあります。

DMACTIONGROUP オブジェクト構造に対する完全な SQL Where 文節は、以下のとおりです。

```
action in (select action from ESCREFPOINT where escalation in  
('INCCLOSE') )
```

DMESCALATION オブジェクト構造に対する完全な SQL Where 文節は、以下のとおりです。

```
escalation in ('INCCLOSE')
```

INCCLOSE エスカレーションのマイグレーションでは、その他の関連構成を収集する必要はないため、マイグレーション・パッケージ内のその他のオブジェクト構造に対する SQL Where 文節は以下のように指定します。

```
1=0
```

この SQL Where 文節により、DMPERSON、DMPERSONGROUP、DMROLE、および DMCOMMTEMPLATE オブジェクト構造の構成レコードは収集されません。

**注 :** Where 文節を空のままにしておくことは、オブジェクト構造の主要なビジネス・オブジェクトの構成レコードがマイグレーション用にすべて収集されることを意味します。これは望ましくない結果になる場合があります。収集する構成レコードがオブジェクト構造にない場合は、SQL Where 文節を 1=0 と指定してください。

SQL Where 文節の作成では、特定のエスカレーションの名前のみを指定する必要があります。別のエスカレーションをマイグレーションする場合には、INCCLOSE エスカレーション ID を目的のエスカレーションの ID に置き換えて、同じマイグレーション・パッケージ定義を再使用できます。

**ヒント :** マイグレーション・グループ内にあるさまざまなオブジェクト構造が互いに関連している限り、これらのオブジェクト構造が主要なオブジェクト構造を参照するように、これらのオブジェクト構造の SQL 節を指定してください。このような SQL 節を作成するには、さまざまなアプリケーションのデータ・モデルの知識が必要です。

パッケージ定義を保存します。これで、残りのマイグレーション手順を実行できるようになりました。

#### 4.6.2 通知が関連付けられているエスカレーションのSQL基準の定義

この説明では、サンプル・エスカレーションである SLAREVIEW を使用します。このエスカレーションは、SLA の有効期限より前に、SLA を検討する指定の担当者に SLA 検討通知を事前に送信することを目的としています。101 ページの図 4-19 に、SLAREVIEW エスカレーションのエスカレーション・ヘッダーを示します。

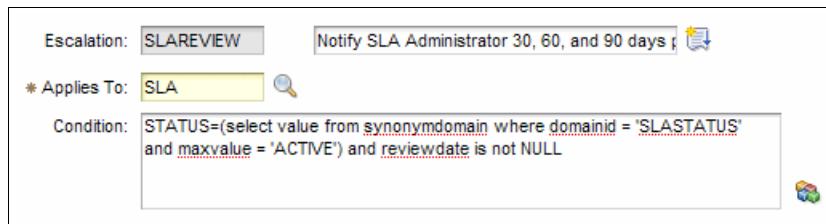


図 4-19 SLAREVIEW サンプル・エスカレーション

SLAREVIEW エスカレーションにはエスカレーション・ポイントが 3 つあり、それぞれが通知を送信しますが、関連付けられているアクションはありません。3 つのエスカレーション・ポイントは、すべて同じ通知を送信しますが、それぞれの通知は異なる時刻に送出されます。図 4-20 に、エスカレーションに関連付けられている SLAREVIEW 通信テンプレートを示します。

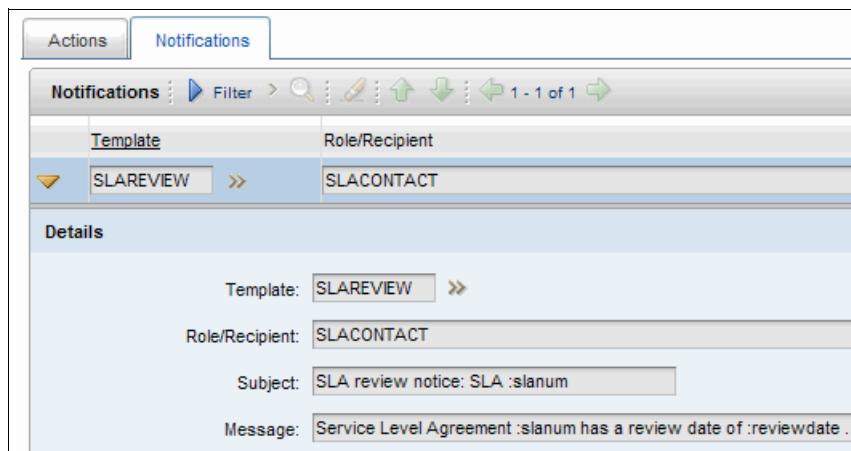


図 4-20 SLAREVIEW 通信テンプレート

通信テンプレート・アプリケーションを使用して SLAREVIEW 通信テンプレートを開くには、「テンプレート」フィールドの右側にある「詳細」メニュー・アイコンを使用します。「通信テンプレート」タブに、「件名」や「メッセージ」情報などの通知のコンテンツが表示され、「受信者」タブにはさまざまな受信者タイプがリストされます。「ロール」セクションを開くと、この通知の送信先のロールがリストされます。102 ページの図 4-21 に、「ロール」セクションを示します。

Role(s) for Communication Template					
	Role	Description	To	cc	bcc
▶	SLACONTACT	Contact person named on SLA	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<a href="#">Select Roles</a>					

図 4-21 SLAREVIEW 通信テンプレートの SLACONTACT ロール

ロール・アプリケーションを使用して SLACONTACT ロールを開くには、「ロール」フィールドの右側にある「詳細」メニュー・アイコンを使用します。「ロール」タブに、ロールの内容が表示されます。このロールは、担当者または担当者グループの構成には依存しません。図 4-22 に、SLACONTACT ロールを示します。

Role:	SLACONTACT	Contact person named on SLA
* Type:	A set of data relat	
Object:	SLA	
* Value:	:SLACONTACT	»
Parameter:		
E-mail?	<input type="checkbox"/>	
Broadcast?	<input checked="" type="checkbox"/>	

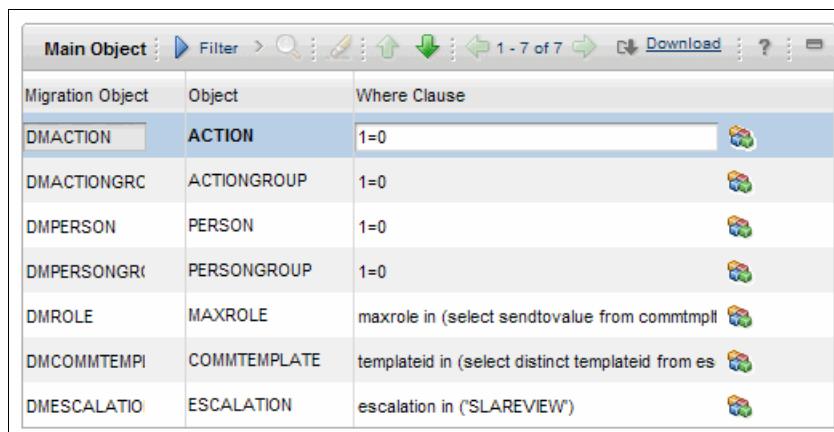
図 4-22 SLACONTACT ロールの詳細

SLAREVIEW エスカレーションのマイグレーションには、関連付けられている通信テンプレート SLAREVIEW およびロール SLACONTACT のマイグレーションが含まれます。

SLAREVIEW エスカレーションは、以前に定義したのと同じパッケージ定義を使用してマイグレーションします。別のパッケージ定義を作成する必要はありません。既存のパッケージ定義を再使用するには、パッケージ定義でさまざまなオブジェクト構造に関連付けられている SQL 基準を変更するだけで済みます。

**注:** SLAREVIEW エスカレーションをマイグレーションする前に、物理パッケージを作成してから SQL 基準を変更する必要があります。これにより、INCCLOSE エスカレーションに属する構成が目的どおりに抽出されます。物理パッケージを作成しないで SQL 基準を変更すると、INCCLOSE エスカレーションに属する構成を抽出できなくなります。あるいは、SQL 基準を結合して、INCCLOSE エスカレーションと SLAREVIEW エスカレーションの両方に属する構成を収集できます。どちらを選択するかは、実装要件に左右されます。

このように、SQL 基準は、マイグレーション・パッケージの一部であるオブジェクト構造に対して設定できます。図 4-23 に、エスカレーション、アクション、およびアクション・グループの各オブジェクト構造に対して設定されている個別の SQL 基準を示します。



The screenshot shows a software interface with a toolbar at the top containing various icons for filtering, saving, and navigating. Below the toolbar is a table with the following data:

Main Object	Object	Where Clause
DMACTION	ACTION	1=0
DMACTIONGRC	ACTIONGROUP	1=0
DMPERSON	PERSON	1=0
DMPERSONGR(	PERSONGROUP	1=0
DMROLE	MAXROLE	maxrole in (select sendtovalue from commtmpltsendto where templateid in (select distinct templateid from escnotification where escalation in ('SLAREVIEW'))))
DMCOMMTEMPL	COMMTEMPLATE	templateid in (select distinct templateid from escnotification where escalation in ('SLAREVIEW')))
DMESCALATIO	ESCALATION	escalation in ('SLAREVIEW')

図 4-23 SLAREVIEW エスカレーションをマイグレーションするための SQL 基準

DMROLE オブジェクト構造に対する完全な SQL Where 文節は、以下のとおりです。

```
maxrole in (select sendtovalue from commtmpltsendto where templateid in
(select distinct templateid from escnotification where escalation in
('SLAREVIEW'))))
```

DMCOMMTEMPLATE オブジェクト構造に対する完全な SQL Where 文節は、以下のとおりです。

```
templateid in (select distinct templateid from escnotification where
escalation in ('SLAREVIEW')))
```

DMESCALATION オブジェクト構造に対する完全な SQL Where 文節は、以下のとおりです。

```
escalation in ('SLAREVIEW')
```

**ヒント:** 誤った SQL Where 文節を入力すると、マイグレーション・マネージャーにより、エラー BMXAA7010E が通知されます。パッケージ定義を保存するには、その前に SQL Where 文節を訂正する必要があります。

パッケージ定義を保存します。これで、残りのマイグレーション手順を実行できるようになりました。

## 4.7 デプロイメント

パッケージを作成し、配布して、デプロイします。エスカレーション構成との関連構成は、稼働中の製品環境にデプロイできます。このセクションでは、デプロイメントに関する考慮事項について説明します。

### 4.7.1 エスカレーションとクーロン・タスク

すべてのエスカレーションは、基礎となるクーロン・タスク・インスタンスを活用して定期的に実行されます。すべてのエスカレーション・クーロン・タスク・インスタンスは、製品と共に提供されている ESCALATION クーロン・タスクを基にして作成されます。

エスカレーションが初めてアクティブ化されたときに、対応するクーロン・タスク・インスタンスも作成され、ESCALATION クーロン・タスクにリンクされます。クーロン・タスク・インスタンス名は、対応するエスカレーションの名前と、その前にある 3 文字の接頭部「ESC」が含まれます。クーロン・タスク・インスタンス名への参照は、ESCALATION テーブルの INSTANCENAME 列に保持されます。例えば、SLAREVIEW エスカレーションは、参照を ESCSLAREVIEW クーロン・タスク・インスタンスに伝達します。エスカレーションは、非アクティブ化されると、以前に作成されたクーロン・タスク・インスタンスへの参照を引き続き保持します。

エスカレーションのクーロン・タスク・インスタンスに対する依存関係を解決する方法として、以下の 2 つがあります。

- ▶ エスカレーション・アプリケーションを活用して、クーロン・タスク・インスタンスを作成する
- ▶ エスカレーションをサポートするクーロン・タスクとそのインスタンスをマイグレーションする

## エスカレーション・アプリケーションを活用して、クーロン・タスク・インスタンスを作成する

この方法では、エスカレーション・アプリケーションの機能を活用して、エスカレーションの定期的な実行をサポートするのに必要なクーロン・タスクを作成します。この方法が有効なのは、エスカレーションを初めてマイグレーションした結果、ターゲットの製品環境にさまざまな構成レコードが挿入された場合です。実行手順は以下のとおりです。

1. ソースおよびターゲットの両方の製品環境で、DMESCALATION オブジェクト構造から INSTANCE 属性を除外します。
  - a. オブジェクト構造アプリケーションを開きます。
  - b. 「一覧」タブで、DMESCALATION オブジェクト構造を見つけます。
  - c. 「オブジェクト構造」タブで、DMESCALATION オブジェクト構造を開きます。
  - d. 「アクションの選択」メニューから、「フィールドの除外 / インクルード」アクションを実行します。
  - e. 「フィールドの除外 / インクルード」ポップアップ・ダイアログで、ESCALATION テーブル・ウィンドウの「ソース・オブジェクト」に、ESCALATION オブジェクトが選択されていることを確認します。
  - f. 「永続フィールド」サブタブで、INSTANCE フィールドが見つかるまでスクロールします。
  - g. 「除外」チェック・ボックスにチェック・マークを付けます。
  - h. 「OK」をクリックして変更を保存し、「フィールドの除外 / インクルード」ポップアップ・ダイアログを閉じます。
2. ソースの製品環境で、目的のエスカレーションを非アクティブ化します。
3. マイグレーション・パッケージを定義し、マイグレーション・グループのオブジェクト構造に適切な SQL 基準を関連付けます。
4. マイグレーション・パッケージを承認し、アクティブ化して作成します。
5. マイグレーション・パッケージをターゲットの製品環境に配布します。
6. マイグレーション・パッケージをデプロイします。
7. デプロイメントが完了したら、マイグレーションしたエスカレーションをエスカレーション・アプリケーションで開き、それぞれをアクティブ化します。

アクティブ化すると、エスカレーション・アプリケーションは、クーロン・タスク・インスタンスが必要かどうかを調べ、インスタンスを作成し、エスカレーションにバック・リンクします。これで、ターゲットの製品環境でエスカレーションが有効になりました。

**注:** この方法は、エスカレーションが挿入されるのではなく更新されるマイグレーション・シナリオの場合にも有効です。マイグレーションの更新シナリオの場合は、クーロン・タスク・インスタンスがエスカレーションと既に関連付けられており、この関連付けはマイグレーション時にもそのままの状態で残ります。

## エスカレーション・クーロン・タスク・インスタンスと一緒にエスカレーションをマイグレーションする

この方法では、ESCALATION クーロン・タスクと関連付けられているすべてのクーロン・タスク・インスタンスが収集され、エスカレーション自体と一緒にマイグレーションされます。この方法では追加手順を実行する必要があります。その結果、マイグレーションにより多くの手間がかかります。手順は以下のとおりです。

1. ソースの製品環境で、エスカレーションのマイグレーションをサポートする新規に作成されたマイグレーション・グループに、DMCRONTASKDEF オブジェクト構造を追加します。DMCRONTASKDEF が先頭のオブジェクト構造になるように、マイグレーション・グループのオブジェクト構造を並べ替えます。106 ページの図 4-24 に、更新されたマイグレーション・グループを示します。



The screenshot shows a software interface titled "Migration Objects for MYESC". It displays a list of migration objects with their descriptions and migration object orders. The objects listed are:

	Migration Object	Description	Migration Object Order
▶	DMCRONTASK	Cron task migration object for Migration Manager	1
▶	DMACTION	Action migration object for Migration Manager	2
▶	DMACTIONGRC	Actiongroup migration object for Migration Manager	3
▶	DMPERSON	Person migration object for Migration Manager	4
▶	DMPERSONGRC	Person group migration object for Migration Manager	5
▶	DMROLE	Role migration object for Migration Manager	6
▶	DMCOMMTEMPI	Communication template migration object for Migration Manager	7
▶	DMESCALATIO	Escalation migration object for Migration Manager	8

図 4-24 DMCRONTASKDEF オブジェクト構造が含まれるマイグレーション・グループ

2. DMCRONTASKDEF オブジェクト構造の追加が反映された既存のマイグレーション・パッケージ定義を再使用します。図 4-25 に、DMCRONTASKDEF オブジェクト構造に関連付けられている SQL Where 文節を示します。

Migration Object	Object	Where Clause
DMCRONTASK	CRONTASKDEF	crontaskname in ('ESCALATION')
DMACTION	ACTION	1=0
DMACTIONGRC	ACTIONGROUP	1=0
DMPERSON	PERSON	1=0
DMPERSONGR	PERSONGROUP	1=0
DMROLE	MAXROLE	maxrole in (select sendtovalue from commtmp)
DMCOMMTEMPI	COMMTEMPLATE	templateid in (select distinct templateid from es)
DMESCALATIO	ESCALATION	escalation in ('SLAREVIEW')

図 4-25 ESCALATION クーロン・タスクをマイグレーションするための SQL 基準

3. パッケージの作成、配布、およびデプロイを行うための標準のマイグレーション手順を実行します。

デプロイメントが完了すると、ESCALATION クーロン・タスクに関連付けられているすべてのクーロン・タスク・インスタンスは、ターゲットの製品データベースに挿入されるか、更新されます。この方法の重大な欠点は、マイグレーションするエスカレーションに関連付けられている特定のクーロン・タスク・インスタンスのみを対象にできないことです。マイグレーション・マネージャーは、オブジェクト構造の主要オブジェクトのみを対象にして、サブオブジェクトは対象にしないように SQL 基準を指定する機能を備えています。

**注:** クーロン・タスクをマイグレーションしない方針の場合は、  
DMCRONTASKDEF オブジェクト構造に対する SQL Where 文節で 1=0 を指定  
して、クーロン・タスク構成を収集しないようにしてください。

**ベスト・プラクティス** 構成の作成または関連付けを自動的に行う機能がビジネス・アプリケーションに用意されている場合は、この機能をマイグレーションのサポートに活用してください。これにより、時間およびリソースの点でコストを削減できます。

## スナップショット処理アクションとエスカレーションでの効果

スナップショット・マイグレーション・パッケージ定義は、必ず処理アクションと関連付けられます。処理アクションは、ソースの製品環境から抽出されたコンテンツが、ターゲットの製品環境へのデプロイメント時にどのように処理

されるかを決定します。スナップショット・マイグレーション・パッケージ定義に関連付けられているデフォルトの処理アクションは「置換」です。ソースの製品環境では、パッケージ定義を承認する前か、パッケージの作成中に処理アクションを「追加変更」に変更できます。各処理アクションにより、マイグレーション中に生成される結果が異なる可能性があります。

図 4-26 に、ソースの製品環境で作成されたエスカレーション 1001 を示します。

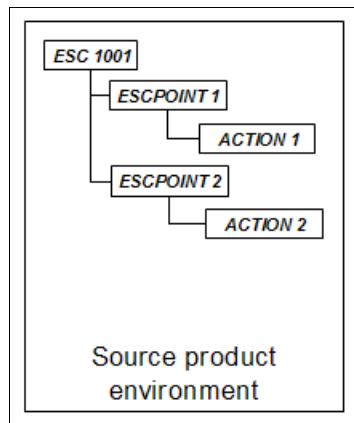


図 4-26 ソース環境で最初に作成されたエスカレーション 1001

エスカレーション 1001 は、ターゲットの製品環境にマイグレーションされます。図 4-27 に、マイグレーションの結果を示します。

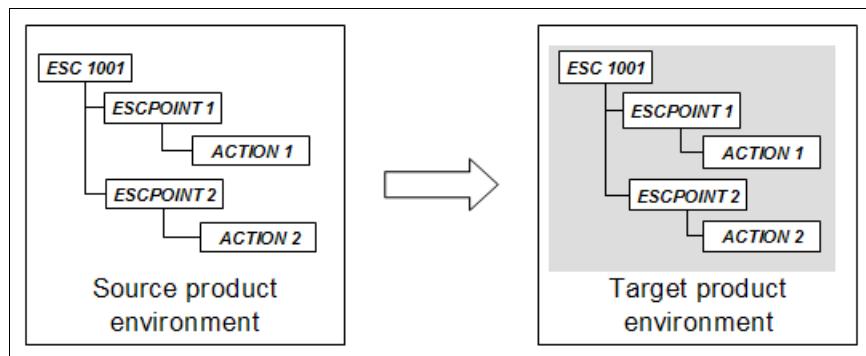


図 4-27 ターゲット環境にマイグレーションされたエスカレーション 1001

ターゲットの製品環境で、エスカレーション 1001 に新規のエスカレーション・ポイントが追加されます。図 4-28 に、更新されたエスカレーションを示します。

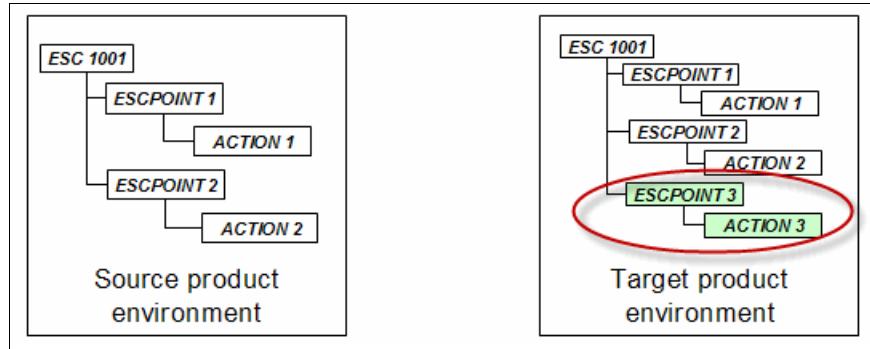


図 4-28 ターゲット環境で直接変更されたエスカレーション 1001

ソースの製品環境で、エスカレーション 1001 に新規のエスカレーション・ポイントが追加されます。109 ページの図 4-29 に、更新されたエスカレーションを示します。

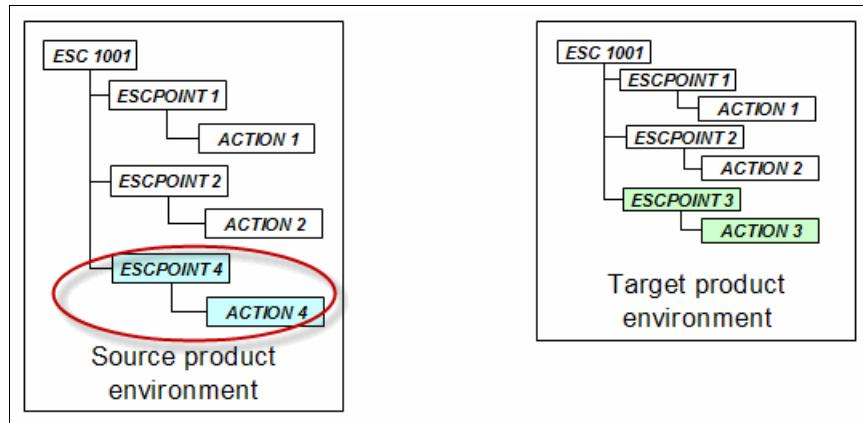


図 4-29 ソース環境で変更されたエスカレーション 1001

エスカレーション 1001 を「置換」処理アクションを使用してソースからターゲットへマイグレーションすると、ターゲットのエスカレーション・ポイント 'ESCPPOINT3' は削除されます。「置換」処理アクションの目的は、ターゲットの製品環境の構成をマイグレーション・パッケージのコンテンツで完全に置き換えることです。図 4-30 に、「置換」処理アクションの効果を示します。

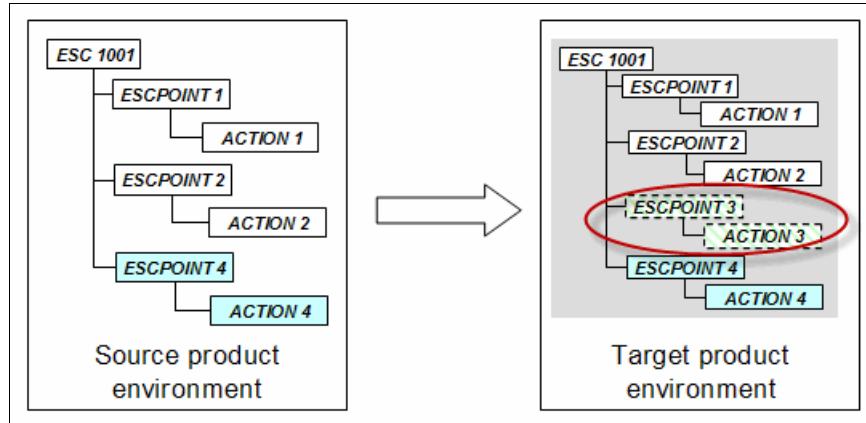


図 4-30 「置換」処理アクションを使用してマイグレーションした変更済みエスカレーション 1001

これは望ましい結果ではありません。ターゲットの製品環境（実稼働環境など）で直接行った変更は、保持する必要があります。エスカレーション 1001 を「追加変更」処理アクションを使用してソースからターゲットへマイグレーションした場合は、エスカレーション・ポイント ESCPOINT3 は保持されます。「追加変更」処理アクションの目的は、ターゲットの製品環境内の構成の要素をそのままの状態で残すことです。110 ページの図 4-31 に、「追加変更」処理アクションの効果を示します。

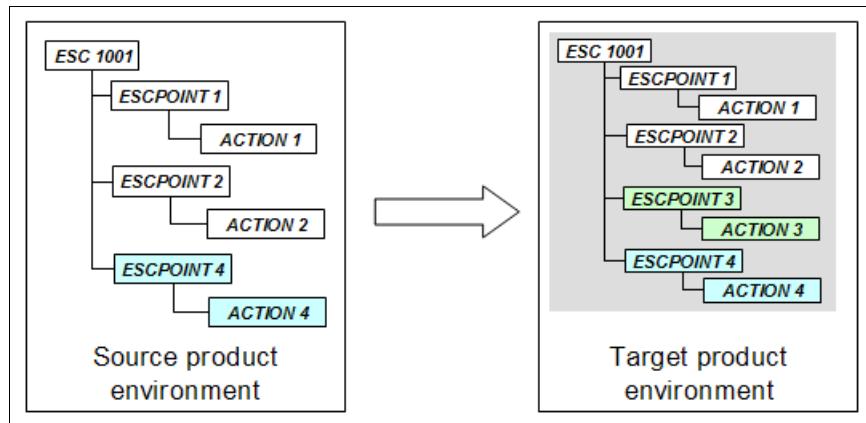


図 4-31 「追加変更」処理アクションを使用してマイグレーションした変更済みエスカレーション 1001

**ベスト・プラクティス** デプロイメントを稼働中の製品環境に対して実行する場合は、スナップショット・マイグレーション・パッケージに対して「追加変更」処理アクションを指定してください。

## 添付ドキュメントの収集

一部の構成は、添付ドキュメントに関連付けられる場合があります。例えば、特定の通信テンプレートには、操作マニュアル・ドキュメントが添付として常に組み込まれる場合があります。この通信テンプレートを使用して通知を送信すると、ドキュメントはメール通知と一緒に受信者に配信されます。マイグレーション時には、添付ドキュメントを構成コンテンツと一緒に組み込む機能は非常に貴重です。

Maximo Base Services Fix Pack 7.1.1.6 のリリースで、DMCOMMTEMPLATE オブジェクト構造に、通信テンプレートとその添付ドキュメントとの関連付けを維持する DOCLINKS オブジェクトが組み込まれました。これにより、添付ドキュメント・ファイルは自動的に収集され、マイグレーション・パッケージに組み込まれます。

すべての添付ドキュメント・ファイルは、フォルダ名およびファイル名の情報に関連付けられます。この情報は、DOCINFO テーブルに保管されます。通信テンプレートとその添付ドキュメントをマイグレーションすると、物理ファイルは、DOCINFO テーブルで指定されたフォルダー構造にコピーされます。ファイル・システム上のフォルダー構造は、ソース環境とターゲット環境の両方で同じになるようにしてください。フォルダー構造が異なると、ファイルは保存されますが、通信テンプレート・アプリケーションから添付ドキュメントとして表示できません。

添付ドキュメントに関するオブジェクト構造のサポートについて詳しくは、以下の資料を参照してください。

<http://www-01.ibm.com/support/docview.wss?uid=swg24027858>

**ベスト・プラクティス** マイグレーション時には、オブジェクトの添付ドキュメントを自動的に収集する機能を活用します。ファイル・システム上の添付ドキュメントのフォルダー構造は、ソースとターゲットの製品環境の両方で同じになるようにしてください。この方法は、マイグレーション・パッケージのコンパイル済みソースを編成して収集するよりも効率的です。

## エスカレーションとワークフロー・プロセス

アクションを介してワークフロー・プロセスを起動するように、エスカレーションを設計できます。ワークフロー・プロセスを起動するエスカレーションおよびアクションをマイグレーションする場合は、ワークフロー・プロセスがエスカレーションおよびアクションよりも先にマイグレーションされるように

してください。2つのマイグレーション・パッケージを使用し、最初のパッケージで目的のワークフロー・プロセスをマイグレーションし、2番目のパッケージでエスカレーションおよびアクションをマイグレーションするようにします。

## エスカレーションと Java クラス

エスカレーションに関連付けられているアクションおよびロールは、カスタムの Java クラスを実行する場合があります。通常、Java クラスは、製品の開発環境またはビルド環境のファイル・システム上にあるフォルダー構造を指定して管理します。さまざまなバージョンの Java クラスを管理し、Java ソース・コードへのアクセスを制御するには、ソース制御システムを使用します。

Java クラスは、コンパイル済みソースの形式で、マイグレーション・パッケージ内に収集できます。パッケージがデプロイされると、マイグレーション・マネージャーは、マイグレーション・パッケージ内の Java クラスを、Java デプロイメント・エンタープライズ・アーカイブ (EAR) を作成できる作成元フォルダーにダウンロードするよう、ユーザーに要求します。マイグレーション・マネージャーでは、EAR ファイルの作成処理やアプリケーション・サーバーへのデプロイメントは一切サポートされていません。

あるいは、目的の Java クラスをソース制御システムから直接取得して、作成手順を実行する方法もあります。

## アクティブなエスカレーションをマイグレーションする場合の落とし穴

前述した方法以外の方法でアクティブなエスカレーションのマイグレーションを実行した場合でも、エラー・フリーのデプロイメントは得られます。ただし、エスカレーションを正常にアクティビ化することはできません。エスカレーション・アプリケーションは、エラー BMXAA4214E を通知します。112 ページの図 4-32 に、エラー・メッセージを示します。

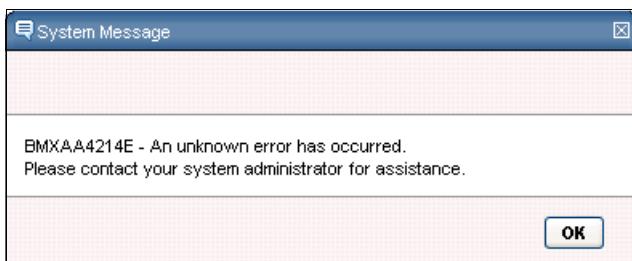


図 4-32 マイグレーションの後にエスカレーションをアクティビ化したときに検出されたエラー

製品のログ・ファイルまたはシステム・コンソールには、以下の Java スタック・トレースが示されます。

```
java.lang.NullPointerException
    at
psdi.app.escalation.Escalation.activateCronTaskInstance(Escalation.java:140)
    at
psdi.app.escalation.Escalation.actDeactEscalation(Escalation.java:104)
)
    at
psdi.webclient.beans.escalation.EscalationAppBean.ADESC(EscalationAppBean.java:48)
```

このエラーが発生するのは、エスカレーションの構成にクーロン・タスク・インスタンスへの参照が含まれているにもかかわらず、クーロン・タスク自体はマイグレーションされなかつたためです。エスカレーションがアクティブ化されると、エスカレーション・アプリケーションは、この存在しないクーロン・タスク・インスタンスをアクティブ化しようとするわけです。

ベスト・プラクティスに従ってエスカレーションをマイグレーションすることにより、この落とし穴は回避できます。





# アプリケーションとスター ト・センターのマイグレー ション

本製品では豊富な標準アプリケーション・ユーザー・インターフェース (UI) 一式がインストールされます。ユーザーはこのインターフェースを利用して、ジョブの実行に必要なタスクの大半を実行できます。ただし、UI の操作性を簡素化したり、ビジネス・プロセス・ルールに従つたりする目的で、ニーズに合わせたインターフェースの変更が必要になる場合もあります。

この章では、アプリケーション UI に関するマイグレーション・コンテンツとその関連メニュー項目 (該当する場合) を判別するためのユースケースと、それを別の環境にプロモートする目的の達成に必要なタスクについて説明します。

各ケースでは最も一般的なシナリオを扱うよう努めていますが、状況によっては、マイグレーション・マネージャーに加えて他のマイグレーション・ツールの使用も変更の範囲に含まれる場合があります。

この章のセクションは以下のとおりです。

- ▶ 116 ページの『5.1 基本的なアプリケーション変更と署名オプション』
- ▶ 124 ページの『5.2 メニュー、ルックアップ、グローバルなデータ制限などの複雑なアプリケーション変更』
- ▶ 139 ページの『5.3 スタート・センターとクエリーのマイグレーション』

## 5.1 基本的なアプリケーション変更と署名オプション

アプリケーションを起動すると、ユーザーとの対話用にアプリケーションUIのレンダリング画面が表示されます。このユースケース・シナリオでは、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アプリケーション・デザイナー」をクリックして表示されるアプリケーション・デザイナー・ツールのみを使用して、アプリケーションUIの基本的な画面表示変更をマイグレーションする方法について説明します。

### 5.1.1 要件

このユースケースでは、*作業指示書管理*アプリケーションの変更内容のマイグレーションを扱います。マイグレーション要件を分析できるように、まず変更要件の要因となる仮定のビジネス・ニーズについて検討しましょう。例えば、アプリケーションUIに以下のような単純な変更を加える必要があるとします。

- ▶ 「故障レポート」画面は不要なので削除する。
- ▶ 「作業タイプ」フィールドの名前を単に「タイプ」とする。
- ▶ 「顧客」フィールドを常に必須にする。
- ▶ 「ライセンス」フィールドは使用しないので、「作業指示書」画面と「詳細検索」の「検索条件の入力」から削除する。
- ▶ 「報告日」をリスト画面の「説明」と「ロケーション」の間に追加し、並べ替え可能にする。
- ▶ 最初に検索結果リストを「報告日」フィールドで並べ替える。
- ▶ 「SLA の適用」の署名オプションがセキュリティに対して表示されないようにする。
- ▶ リスト結果の表示時に「KPI の作成」ツールバー・ボタンをクリックすると、このアクションがすべてのリスト・レコードに影響する旨の警告が出されるようになる。
- ▶ 「レポートの実行」メニュー・オプションを「アクションの選択」メニューの最初の項目にする。
- ▶ リスト画面の「ステータスの変更」ボタンを無効にし、このボタンを「作業指示書」画面でのみ使用可能にする。
- ▶ 作業指示書の「着手」、「承認」、「完了」、および「クローズ」の各ツールバー・ボタンを削除する。
- ▶ 「Where 文節」オプションを「詳細検索」メニューの最初の項目にする。

このニーズを満たすには、ビジネス・ニーズに基づいてアプリケーションUIに以下の基本的な変更を加えます。

1. アプリケーション・デザイナーを使用して、作業指示書管理アプリケーションを検索し、変更を加えるために開きます。このアプリケーションの名前は以下のとおりです。

a. WOTRACK

2. アプリケーション・デザイナーのワークスペースを使用して、要件を満たすように WOTRACK アプリケーションに以下の変更を加えます。
  - a. 「一覧」タブ画面で、既存のオブジェクト属性「報告日」に対する新しいテーブル列コントロールを、「説明」列と「ロケーション」列の間に追加します。クリック・アンド・ドラッグやカット・アンド・ペーストなどの編集テクニックを使用して、画面内のコントロールのレイアウトを変更できます。
  - b. 追加した「報告日」コントロールのプロパティーを編集して、「並べ替え可能」属性のチェック・マークを外し、「入力モード」属性を「読み取り専用」にします。

**注:** 最も一般的に変更されるコントロールは、テキスト・ボックス、チェック・ボックス、セクション、静的テキスト、ヘルプ・グリッド、テーブル、プッシュボタンです。

- c. 「メイン」タブ画面から「ライセンス」フィールドの不要なコントロールを削除します。
- d. 署名オプション *APPLYSLA* の拡張署名オプションを変更して、2番目のオプション「このアクションの選択時に警告を表示する (Warning appears when this action is selected)」を選択します。
- e. 「可視」オプションのチェック・マークを外して、「アクションの選択」メニュー・オプションのキー値 *APPLYSLA* の可視性を変更します。
- f. ツールバー・ボタンを *MAIN* タブでのみ使用可能にして、「ツールバー」メニュー・オプションのキー値 *STATUS* の可視性を変更します。
- g. 下位位置の値をリスト内の最初のキー値よりも低い数値に変更して、「検索」メニュー・オプションのキー値 *SEARCHWHERE* の位置を変更します。

**注:** 各ソリューションでの変更方法は各固有要件の詳細によって異なる場合があるため、本書ではこれらの変更方法の詳細は扱いません。アプリケーション・デザイナー・ツールの使用方法について詳しくは、以下の URL で「*IBM Maximo Asset Management 7.1, IBM Tivoli Asset Management for IT 7.1, IBM Tivoli Change and Configuration Management Database 7.1.1, IBM Tivoli Service Request Manager 7.1* アプリケーション開発者ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_app\\_dev\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_app_dev_guide.pdf)

これらの変更は、製品に付属のアプリケーション・デザイナー・ツールのみを使用して変更要件を満たせるという点で、比較的小さな変更と考えられます。

## 5.1.2 ソリューション

作業指示書管理・アプリケーションに対する変更内容を確認したら、単一のスナップショット・マイグレーション・パッケージを使用してすべての関連構成コンテンツがマイグレーションされるような方法を取ります。

コンテンツの観点から見ると、変更されたアプリケーションをマイグレーションするには、以下の構成をマイグレーションする必要があります。

1. アプリケーション

## 5.1.3 このソリューションに使用する構成アプリケーション

アプリケーション構成にアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アプリケーション・デザイナー」をクリックします。

## 5.1.4 オブジェクト構造

このユースケース・シナリオで定義されたソリューションのマイグレーションをサポートするために、製品に用意されている以下の標準オブジェクト構造が使用されます。

- ▶ DMMAXAPPS
- ▶ DMMAXMENU
- ▶ DMSIGOPTFLAG

図 5-1 に、これら 3 つのオブジェクトの詳しいオブジェクト構造を示します。マイグレーション・オブジェクト構造にアクセスするには、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」をクリックします。

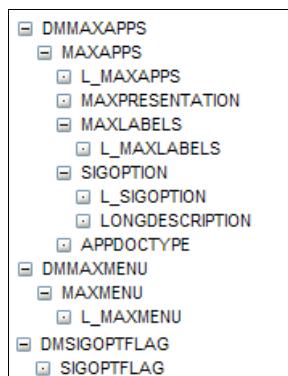


図 5-1 このケース・シナリオのオブジェクト構造

**注:**マイグレーション・オブジェクト構造について詳しくは、以下の URL で「マイグレーション・マネージャー・ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)

### 5.1.5 マイグレーション・グループ

変更要件が明確に特定されたら、次に、オブジェクト構造のグループを定義および作成します。これらのオブジェクト構造は、マイグレーションする必要のある構成の論理コンテナーとなります。

このマイグレーションのオブジェクト構造は、*APPLICATION* と *APPSECURITY* という 2 つの標準マイグレーション・グループに組み込まれます。これらのグループは、アプリケーション変更とセキュリティー変更をサポートするために製品と共に提供されています。

アプリケーション・デザイナーで定義された拡張署名オプションの構成コンテナは、標準 *APPLICATION* グループには組み込まれませんが、*APPSECURITY* グループには組み込まれます。

これらのマイグレーション・グループとその依存関係は、構成のサブセットを選択的にマイグレーションする必要があるシナリオではそれほど有用でないため、現状のままにしてください。図 5-2 に、これらのマイグレーション・グループのオブジェクト構造リストを示します。

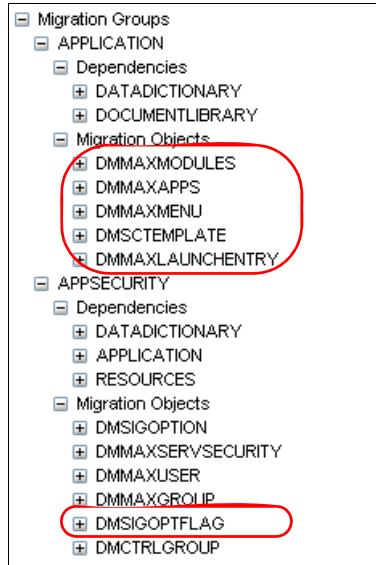


図 5-2 APPLICATION およびAPPSECURITY マイグレーション・グループ

このユースケース・シナリオのマイグレーション・パッケージを作成するためには必要なオブジェクト構造が組み込まれた、新規アプリケーション・マイグレーション・グループが必要です。新規グループは前述の 2 つのグループをベースにしたもので、図 5-2 で強調表示されたオブジェクト構造が組み込まれます。マイグレーション・グループ・アプリケーションを使用して、APPLICATION を複製し、依存関係にあるマイグレーション・グループを削除して、新規マイグレーション・グループを作成します。

次に、この新規グループに、拡張署名オプションの構成が含まれたオブジェクト構造を組み込みます。DMSIGOPTFLAG オブジェクトを新規グループの DMMAXLAUNCHENTRY オブジェクトの後に追加して保存します。図 5-3 に、この新規グループを示します。

The screenshot shows the SAP GUI interface for defining a migration group. At the top, there is a table titled 'Migration Object' with the following data:

Migration Object	Description	Migration Object Order
DMMAXMODUL >>	Application module migration object for Migration	1
DMMAXAPPS >>	Application migration object for Migration Mana	2
DMMAXMENU >>	Application menu migration object for Migration	3
DMSCTEMPLATE >>	Start Center template migration object for Migra	4
DMMAXLAUNCI >>	Launch entry migration object for Migration Ma	5
DMSIGOPTFLA( >>	Advanced signature option migration object for	6

Below this is another table titled 'Dependent Migration Group' which displays the message "...No rows to display...".

図 5-3 新規マイグレーション・グループ MYAPPLICATION

**ヒント:** マイグレーション・グループは、マイグレーションされる特定のコンテンツ（またはオブジェクト）の定義に使用するシェルです。これらのグループは、さまざまなマイグレーション・シナリオに何度も再利用することができます。

## 5.1.6 パッケージ定義

オブジェクト構造の特定と編成が完了したら、マイグレーション・プロセスを開始することができます。新しく定義したマイグレーション・グループが含まれた、新規スナップショット・パッケージを作成してください。このパッケージを使用して、ターゲット環境にマイグレーションされる具体的なコンテンツを定義します。

次に、グループに含まれる各オブジェクト構造内に、コンテンツを選択するための SQL 基準を定義します。作業指示書管理アプリケーションの名前 *WOTRACK* を使用して、SQL 条件内の基準を作成します。

122 ページの表 5-1 に、各オブジェクトの *Where* 文節の SQL コマンド・ステートメント条件を示します。これらの条件によって、マイグレーション・オブジェクトごとに構成コンテンツのレコードが 1 つまたは複数生成されます。

**注:** マイグレーション・パッケージには、特定のマイグレーション・オブジェクトのレコードが含まれない場合もあります。Where 文節が **1=2** の場合、エラーは生成されません。この文節は、特定のオブジェクトに関連する構成情報が含まれていない場合によく使用されます。

表 5-1 基本的なアプリケーション・パッケージに対する Where 文節定義の設定

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMMAXMODULES	MAXMODULES	<b>1=2</b>
DMMAXAPPS	MAXAPPS	<code>app in ('WOTRACK')</code>
DMMAXMENU	MAXMENU	<code>moduleapp in ('WOTRACK')</code>
DMSCTEMPLATE	SCTEMPLATE	<b>1=2</b>
DMMAXLAUNCHENTRY	MAXLAUNCHENTRY	<b>1=2</b>
DMSIGOPTFLAG	SIGOPTFLAG	<code>app in ('WOTRACK')</code>

このケースでは、MAXAPPS、MAXMENU、SIGOPTFLAG の各オブジェクトに格納される構成コンテンツのみをマイグレーションするため、各オブジェクトの条件に従って、パッケージに格納される正確なコンテンツが生成されます。

**ヒント:** 複数のアプリケーションを変更し、それらをマイグレーション対象としてプロモートする場合は、各アプリケーション名をコンマで区切って追加することができます。以下に例を示します。

`app in ('WOTRACK', 'CHANGE', 'JOBPLAN')`

DMMAXMODULES、DMSCTEMPLATE、および DMLAUNCHENTRY の *Where* 文節は、論理的な false ステートメント (**1=2**) で定義されているため、結果は生成されません。この特定のケースでは、これは意図したものです。

SQL 条件を入力するには、MYAPPLICATION マイグレーション・グループの「*Where* 文節の設定」アイコンをクリックします(123 ページの図 5-4 を参照)。

Migration Object	Object	Where Clause
DMMAXMODUL	MAXMODULES	1=2
DMMAXAPPS	MAXAPPS	app in ('WOTRACK')
DMMAXMENU	MAXMENU	moduleapp='WOTRACK'
DMSCTEMPLAT	SCTEMPLATE	1=2
DMMAXLAUNCI	MAXLAUNCHENTRY	1=2
DMSIGOPTFLA(	SIGOPTFLAG	app in ('WOTRACK')

図 5-4 WOBASICAPP マイグレーション・パッケージの Where 文節条件

### 5.1.7 デプロイメント

パッケージ定義が承認され、アクティブ化されたら、パッケージの作成と配布を行うことができます。その後、パッケージをターゲット環境にデプロイします。パッケージをデプロイする前に、**管理モード**をオンにしてください。MAXAPPS オブジェクトが含まれたパッケージをデプロイする際には、マイグレーション・マネージャーを**管理モード**にする必要があります。

**注:** 本書では、パッケージのデプロイに必要な標準ステップは繰り返しません。詳しくは、以下の URL で「マイグレーション・マネージャー・ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)

### 5.1.8 デプロイメントに関する考慮事項

このユースケース・シナリオでは、任意のアプリケーションに対する基本的な変更内容をマイグレーションする方法を示します。ただし、以下のセクションで説明する考慮事項に留意してください。

#### 言語に関する考慮事項

言語に関する考慮事項について詳しくは、215 ページの『10.2 複数言語に関する考慮事項』を参照してください。

## 5.2 メニュー、ルックアップ、グローバルなデータ制限などの複雑なアプリケーション変更

この新しいユースケース・シナリオでは、アプリケーション・デザイナー・ツールを使用する前後に他のアプリケーションを使用する必要のある、より複雑な構成コンテンツのマイグレーションについて説明します。

ここでは、新しいメニュー項目を作成し、新しいライブラリー・ルックアップを構築し、セキュリティー制限をデータ属性に適用します。

構成サイクルにおけるこのイベント・シーケンスによって、マイグレーションの依存関係が発生します。これらの依存関係は、マイグレーション要件とソリューションで分析し、さらにマイグレーション・パッケージの定義とデプロイメントに関する考慮事項でも扱います。

### 5.2.1 要件

このユースケースでは、資産アプリケーションに対する変更内容をマイグレーションします。マイグレーション要件を分析できるように、まず仮定の変更要件について検討しましょう。この変更要件では、シリアルライズされた資産のテスト結果を記録し、これらのテストを追跡し、テストが失敗した場合は通知を送信する必要があります。

- ▶ 資産テーブルからこの新しいテスト結果テーブルへの1対多の関係を使用して、新規テーブルを作成する。
- ▶ この新規テーブルに、シリアルライズされた資産に関するテスト結果、日付、テストのタイプ、テストの説明、障害コード（テストが失敗した場合）などの情報を格納する。
- ▶ テスト結果は、テスト担当チームのみが使用できる新しいアプリケーションUIを使用して記録される。
- ▶ テスト・タイプとテスト結果は事前定義値と照合して検証され、日付フィールドはデフォルトで現在の日付に設定される。
- ▶ テスト結果の削除や複製はできない。
- ▶ 基本的な検索オプションとリスト・オプションは、「一覧」タブから使用できる。
- ▶ 障害コードを除き、すべての情報を必須とする。障害コードは、テストが失敗した場合のみ必須とする。
- ▶ 資産に関連するテスト結果を表示するための新しいタブを資産アプリケーションに追加する。このタブにはテスト担当チームのみがアクセスできる。

このシナリオのビジネス・ニーズを達成するために必要な最小の構成変更は、以下のとおりです。

1. ドメイン・アプリケーションを使用して、2つの新規 **ALN ドメイン**を作成します。これらのドメインは、テスト結果とテスト・タイプの検証値に使用されます。これらのドメインの名前は以下のとおりです。
  - a. *MYTESTRESULT*
  - b. *MYTESTTYPE*
2. ドメイン・アプリケーションを使用して、新規 **テーブル・ドメイン**を作成します。このドメインは、新規テーブルのリンクされた資産フィールドの値を検索する際に使用されます。このドメインの名前は以下のとおりです。
  - a. *MYASSETTB*
3. データベースの構成アプリケーションを使用して、新規テーブル・オブジェクトを作成します。このオブジェクトには、すべての必須属性、新規ドメインの関連付け、資産のシリアル番号の表示に使用する ASSET テーブルとの新規関係、およびその他の基本的なオブジェクト定義が含まれます。新規テーブル・オブジェクトの名前は以下のとおりです。
  - a. *MYTESTS*
4. アプリケーション・デザイナー・アプリケーションを使用して、前のステップで作成されたテーブル・ドメインを指す新規テーブル ID 参照を、LOOKUPS システム・ライブラリー XML に作成します。
5. アプリケーション・デザイナー・アプリケーションを使用して、「パワー・アプリケーション」タイプの新規アプリケーションを作成します。このアプリケーションには、新規テーブル・オブジェクトの必須構成がすべて含まれます。新規アプリケーションの名前は以下のとおりです。
  - a. *MYTESTS*
6. アプリケーション・デザイナー・アプリケーションを使用して、既存の資産アプリケーションを、資産にリンクされたテスト結果レコードの表示に必要な新規タブとコントロールを含むように構成します。この新規タブに対してセキュリティー制限を構成するために、新しい署名オプションを作成します。既存のアプリケーションの名前は以下のとおりです。
  - a. *ASSET*
7. セキュリティー・グループ・アプリケーションを使用して、2つの新規セキュリティー・グループを作成します。これらのグループは、新しいテスト結果アプリケーションと、資産アプリケーション内の新規タブに対して、署名セキュリティー・アクセスを構成するために使用します。2つの新規セキュリティー・グループの名前は以下のとおりです。
  - a. *TESTMGR*
  - b. *TESTTECH*

8. 条件式マネージャー・アプリケーションを使用して、値が *FAILED* の場合にテスト結果属性を評価する新規条件を作成します。新規の条件式の名前は以下のとおりです。
  - a. *MYFAILED*
9. セキュリティー・グループ・アプリケーションを使用して、新規のグローバルなデータ制限を作成します。この制限では、データの入力時に、前のステップで作成した条件に基づいてテスト結果フィールドを評価し、テスト障害コード・フィールドの入力モードを動的に「必須」に変更します。属性制限の対象となるアプリケーションの名前は以下のとおりです。
  - a. *MYTEST*

上記からわかるように、少なくとも 5 種類のアプリケーションから生成された構成コンテンツをマイグレーションする必要があります。また、この構成を作成する順序が非常に重要となります。

### 5.2.2 ソリューション

このケース・シナリオの変更内容を確認したら、複数のスナップショット・マイグレーション・パッケージを使用してすべての関連構成コンテンツがマイグレーションされるような方法を取ります。これらのパッケージは、特定の順序でターゲット環境にデプロイされます。

コンテンツの観点から見ると、変更されたアプリケーションをマイグレーションするには、以下の構成を順序どおりにマイグレーションする必要があります。

1. ドメイン
2. データベース表、シーケンス、関係
3. 条件式
4. システム・ライブラリー・ルックアップ
5. アプリケーション、メニュー、署名オプション
6. グローバルなセキュリティー制限
7. セキュリティー・グループ

### 5.2.3 このソリューションに使用する構成アプリケーション

このソリューションに使用する以下の構成アプリケーションと、これらのアプリケーションにアクセスする方法を確認してください。

## ドメイン

ドメイン・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「ドメイン」をクリックします。

## データベース表、シーケンス、関係

データベースの構成アプリケーションを起動するには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「データベースの構成」をクリックします。

## 条件式

条件式マネージャー・アプリケーションにアクセスするには、「リンク先」→「管理」→「条件式マネージャー」をクリックします。

## システム・ライブラリー・ルックアップ

アプリケーション・デザイナー・アプリケーションを起動するには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アプリケーション・デザイナー」をクリックします。アプリケーション・デザイナーから、「アクションの選択」→「システム XML のエクスポート」を選択します。

## アプリケーション、メニュー、署名オプション

アプリケーション、メニュー、署名オプションは、アプリケーション・デザイナーから構成できます。このアプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アプリケーション・デザイナー」をクリックします。

## グローバルなセキュリティー制限

グローバルなセキュリティー制限を構成するには、「リンク先」→「システムの構成」→「セキュリティー・グループ」をクリックし、「アクションの選択」→「グローバルなデータ制限」を選択します。

## セキュリティー・グループ

セキュリティー・グループ・アプリケーションを起動するには、「リンク先」→「システムの構成」→「セキュリティー・グループ」をクリックします。

## オブジェクト構造

オブジェクト構造アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」をクリックします。

## 5.2.4 オブジェクト構造

このシナリオの構成コンテンツをマイグレーションするために、いくつかの標準オブジェクト構造を特定します。ただし、グローバルなデータ制限とシステム・ライブラリー・ロックアップの構成コンテンツには特別な注意が必要です。

**ヒント:** 製品に用意されている標準オブジェクトと標準グループは、開発者によって作成された論理的なビジネス・オブジェクト構造と関連付けを反映しているため、これらを使用するようにしてください。

### 標準オブジェクト構造

マイグレーション要件をサポートするために、製品に用意されている以下の標準オブジェクト構造が使用されます。

- ▶ DMMAXOBJECTCFG
- ▶ DMMAXDOMAIN
- ▶ DMMAXRELATIONSHIP
- ▶ DMCONDITION
- ▶ DMMAXAPPS
- ▶ DMCONDITION
- ▶ DMMAXMENU
- ▶ DMMAXGROUP

### アプリケーションに関連するグローバルなデータ制限のコンテンツ

この構成コンテンツは現在、標準オブジェクト構造 DMMAXGROUP で定義された *SECURITYRESTRICT* テーブルに含まれており、その主要な関係はセキュリティー・グループです。データ制限がグローバルな場合は、セキュリティー・グループには関係しません。

このユースケースでは、セキュリティー・グループではなくアプリケーション名にリンクされたグローバルなデータ制限を構成したため、標準オブジェクト構造を使用してこのコンテンツをマイグレーションすることはできません。

**注:** 標準のマイグレーション・オブジェクト構造 DMMAXGROUP について詳しくは、51 ページの『第 3 章 セキュリティー構成データのマイグレーション』を参照してください。

したがって、グローバルなセキュリティー制限をマイグレーションするには、新しいマイグレーション・オブジェクト構造を作成する必要があります。この新規オブジェクトは DMMAXAPPS オブジェクトの複製であり、

SECURITYRESTRICT テーブルのコンテンツをアプリケーション名で関連付けるために、このテーブルが追加されます。

以下の特性を使用して、DMMAXAPPS を複製することで新規のマイグレーション・オブジェクト構造を作成します。

**オブジェクト構造** MYMAXAPPS

**説明** データ・セキュリティー制限のあるアプリケーション

以下の特性を使用して、MYMAXAPPS のソース・オブジェクトのリストの末尾に新規のソース・オブジェクトを追加します。

**オブジェクト** SECURITYRESTRICT

**親オブジェクト** MAXAPPS

**関係** SECURITYRESTRICT

**オブジェクト順序** 6(またはデフォルト)

新規のオブジェクト構造を保存します。

図 5-5 に、新規の MYMAXAPPS オブジェクト構造リストを示します。

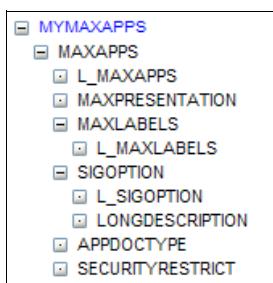


図 5-5 新規のマイグレーション・オブジェクト構造 MYMAXAPPS

ここで作成した新規オブジェクト構造は、その構成コンテンツをマイグレーションする必要があります。このオブジェクト構造は、標準オブジェクト構造 DMMAXINTOBJECT に含まれます。

### システム・ライブラリー・ルックアップのオブジェクト構造

システム・ルックアップを構成するには、アプリケーション・デザイナーを使用します。同じ方法でターゲット環境にマイグレーションすることもできます。ただし、このシナリオでは、マイグレーション・グループに含めるマイグレーション・オブジェクト構造を作成します。

この構成は、XML コンテンツの形式で MAXPRESENTATION テーブルに格納されます。このテーブルには、各アプリケーションの XML 表示も含まれます。このテーブルは、MYMAXAPPS オブジェクト構造に含まれます(図 5-5 を参

照)。LOOKUPS ライブラリー用に構築されたアプリケーションはないため、表示テーブルではアプリケーション名でコンテンツが識別されます。

すべてのデフォルト値と以下の特性を使用して、MYMAXPRESENTATION という名前の新規マイグレーション・オブジェクト構造を作成します。

**オブジェクト構造** MYMAXPRESENTATION

**説明** XML 表示

**利用側** MIGRATIONMGR

デフォルト値はすべてそのままで、以下の特性を使用して、  
MYMAXPRESENTATION のソース・オブジェクトのリストの末尾に新規のソース・オブジェクトを追加します。

**オブジェクト** MAXPRESENTATION

**オブジェクト順序** 1( またはデフォルト )

新規のオブジェクト構造を保存します。

図 5-6 に、新規の MYMAXPRESENTATION オブジェクト構造を示します。

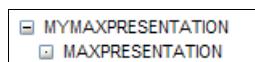


図 5-6 MYMAXPRESENTATION オブジェクト構造

ここで作成した新規オブジェクト構造は、その構成コンテンツをマイグレーションする必要があります。このマイグレーションには、標準オブジェクト構造 DMMAXINTOBJECT を使用します。

### マイグレーション・オブジェクト構造

以下に、マイグレーション要件のサポートに使用するオブジェクト構造の改訂リストを示します( 特定の順序ではありません )。

- ▶ DMMAXOBJECTCFG
- ▶ DMMAXINTOBJECT
- ▶ DMMAXDOMAIN
- ▶ DMMAXSEQUENCE
- ▶ DMMAXRELATIONSHIP
- ▶ DMCONDITION
- ▶ DMMAXMENU
- ▶ DMMAXGROUP
- ▶ MYMAXAPPS
- ▶ MYMAXPRESENTATION

**注:**マイグレーション・オブジェクト構造について詳しくは、以下の URL で「マイグレーション・マネージャー・ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)

### 5.2.5 マイグレーション・グループ

変更要件を確認し、オブジェクト構造を特定したら、次に、パッケージ・コンテンツの定義に使用するグループを定義します。

最初のシナリオ 5.1、『基本的なアプリケーション変更と署名オプション』では、変更内容が 1 つの構成ツールに限られており、他の構成から独立していたため、1 つのグループと 1 つのパッケージしか使用しませんでした。

今度は、ソース環境で構成が実行された順序を反映したシーケンスで、個別の構成グループをマイグレーションします。製品構成サイクルの 3 つの個別領域（データベース構成、アプリケーション、セキュリティーなど）に分けて構成を定義すると、マイグレーション・プロセスを制御しやすくなり、明確に理解できるようになります、さらに再利用も可能になります。

**注:**マイグレーション・グループは、マイグレーションされる特定のコンテンツ（またはオブジェクト）の定義に使用されるシェルです。これらのグループは、さまざまなマイグレーション・シナリオに何度も再利用することができます。

使用する 3 つのグループは以下のとおりです。

#### データ・ディクショナリー・グループ

最初に使用するグループは、製品に用意されている標準の *DATADICTONARY* グループです。図 5-7 に、このグループに含まれる必須オブジェクトを強調表示します。

Migration Groups
DATAIDICTIONARY
Migration Objects
+ DMAXSERVICE
+ DMLANGUAGE
+ DMMAXMESSAGES
+ DMMAXLOOKUPMAP
+ DMURRENCY
+ DMSETS
+ DMORGANIZATION
+ DMAXVARS
+ DMAXDOMAIN
+ DMAXOBJECTCFG
+ DMAXSEQUENCE
+ DMAXRELATIONSHIP
+ DMAXINTOBJECT
+ DMCONDITION
+ DMGLCONFIGURE

図 5-7 DATAIDICTIONARY マイグレーション・グループ

## アプリケーション・グループ

これは、セクション 5.1.5, 『マイグレーション・グループ』の基本的な変更のシナリオで、標準グループから作成された新規マイグレーション・グループです。ただしここでは、条件式とシステム・ルックアップに関する工夫があります。

*CONDITION* テーブルには条件式が格納されます。このシナリオでは、この新規テーブルを評価する目的で作成された新しい式をマイグレーションする必要があります。図 5-7 でわかるように、マイグレーション・マネージャーは、最初のグループに含まれる DMAXOBJECTCFG を使用してこの新規テーブルを作成します。同様に、DMCONDITION もこのグループに含まれていますが、グループのマイグレーション順序が逆になります。つまり、この新規テーブルは条件の後に作成されるため、このグループと共に条件をマイグレーションしようととしても、条件の処理時には新規テーブルがまだ作成されていないためマイグレーションは失敗します。133 ページの図 5-8 に、マイグレーション・オブジェクト構成とオブジェクト順序を示します。

Migration Object	Description	Migration Object Order	Int.
DMMAXSERVICE	System service migration object for Migration Manager	1	<input checked="" type="checkbox"/>
DMLANGUAGE	Language migration object for Migration Manager	2	<input checked="" type="checkbox"/>
DMMAXMESSAGE	System message migration object for Migration Manager	3	<input checked="" type="checkbox"/>
DMCURRENCY	Currency migration object for Migration Manager	4	<input checked="" type="checkbox"/>
DMSETS	Set migration object for Migration Manager	5	<input checked="" type="checkbox"/>
DMORGANIZATION	Organization and site migration object for Migration Manager	6	<input checked="" type="checkbox"/>
DMMAXVARS	MAXVARS migration object for Migration Manager	7	<input checked="" type="checkbox"/>
<b>DMCONDITION</b>	Condition expression migration object for Migration Manager	8	<input checked="" type="checkbox"/>
DMMAXDOMAIN	Domain migration object for Migration Manager	9	<input checked="" type="checkbox"/>
<b>DMMAXOBJECT</b>	Data dictionary object migration object for Migration Manager	10	<input checked="" type="checkbox"/>

図 5-8 DATADICTONARY マイグレーション・オブジェクト

この問題に注意するため、データ・ディクショナリー・グループではなくアプリケーション・グループ内の条件式をマイグレーションします。また、この新規グループと共に、システム・ルックアップ・ライブラリーのマイグレーション用に作成された新規オブジェクト構造も組み込みます。

まず標準の APPLICATIONS グループを複製し、デフォルトはすべてそのままにして新規グループを作成したら、以下の手順を実行します。

- 依存関係にあるグループをすべて削除します。
- オブジェクト構造 DMCONDITION をオブジェクト・リストに追加し、追加されたオブジェクトが最初になり、それ以外のオブジェクトが 1 つずつずれるようにオブジェクト順序を再調整します。
- オブジェクト DMMAXAPPS を削除します。このオブジェクトを削除する前に、このオブジェクト順序をメモしてください。
- 5.2.4. 『オブジェクト構造』で作成したオブジェクト構造 MYMAXAPPS を追加します。オブジェクト順序を DMMAXAPPS の場合と同じようにします。
- オブジェクト構造 MYMAXPRESENTATION をオブジェクト・リストに追加します。リストの末尾に置いてください。
- このグループを保存します。

134 ページの図 5-9 に、新規作成されたグループ・オブジェクト・リストと、マイグレーション・オブジェクトの順序を示します。

Migration Objects for MYNEWAPP				
	Migration Object	Description	Migration Object Order	Int
▶	DMCONDITION	Condition expression migration object for Migration	1	
▶	DMMAXMODUL	Application module migration object for Migration	2	
▶	MYMAXAPPS	Application with data security restriction	3	
▶	DMMAXMENU	Application menu migration object for Migration	4	
▶	DMSCTEMPLAT	Start Center template migration object for Migration	5	
▶	DMMAXLAUNCI	Launch entry migration object for Migration Manager	6	
▶	MYMAXPRESEN	System XML migration object for Migration Manager	7	

図 5-9 MYNEWAPP マイグレーション・オブジェクト

## アプリケーション・セキュリティー・グループ

3番目のグループは、セキュリティー・グループのマイグレーション用に製品に用意されている標準グループ *APPSECURITY* をベースにして作成される新規グループです。新規グループを作成するには、標準の APPSECURITY グループを複製して、デフォルトはすべてそのままにします。ただし、以下の手順を実行します。

1. 依存関係にあるグループをすべて削除します。
2. この新規グループを保存します。

**注:** アプリケーション・セキュリティーに関するその他のユースケース・シナリオについては、51 ページの『第 3 章 セキュリティー構成データのマイグレーション』を参照してください。

これで、このユースケース・シナリオに使用するマイグレーション・グループの定義が完了し、マイグレーション・プロセスを開始する準備が整いました。

### 5.2.6 パッケージ定義

このユースケース・シナリオでは、3つの個別のスナップショット・パッケージをデプロイするマイグレーション方法を取ります。各パッケージは、5.2.5、『マイグレーション・グループ』で定義されたマイグレーション・グループのいずれかを使用して作成され、マイグレーション要件を反映した SQL ステートメント条件基準を使用して構成されています。

マイグレーション用に正確な構成コンテンツを持つ新規パッケージを作成するには、以下の各セクションのプロセスに従ってください。

## データ・ディクショナリー・パッケージ

MYTESTS-DD という名前の新規パッケージ定義を作成します。これは、マイグレーション・グループ DATAIDICTIONARY が含まれるスナップショット・タイプのパッケージです。

次に、グループに含まれる各オブジェクト構造内に、SQL 基準を定義します。セクション 124 ページの『5.2.1 要件』で収集した情報を使用し、各オブジェクト（新規または変更済み）の名前を使用して SQL 条件を構成します。

表 5-2 に、各オブジェクトの *Where* 文節の SQL コマンド・ステートメント条件を示します。これらの条件によって、マイグレーション・オブジェクトごとに構成コンテンツのレコードが 1 つまたは複数生成されます。

表 5-2 MYTESTS-DD パッケージの *Where* 文節定義

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMMAXSERVICE	MAXSERVICE	<b>1=2</b>
DMLANGUAGE	LANGUAGE	<b>1=2</b>
DMMAXMESSAGES	MAXMESSAGES	<b>1=2</b>
DMCURRENCY	CURRENCY	<b>1=2</b>
DMSETS	SETS	<b>1=2</b>
DMORGANIZATION	ORGANIZATION	<b>1=2</b>
DMMAXVARS	MAXVARS	<b>1=2</b>
DMCONDITION	CONDITION	<b>1=2</b>
DMMAXDOMAIN	MAXDOMAIN	<b>domainid in ('MYASSETTB', 'MYTESTRESULT', 'MYTESTTYPE')</b>
DMMAXOBJECTCFG	MAXOBJECTCFG	<b>objectname in ('MYTESTS')</b>
DMMAXSEQUENCE	MAXSEQUENCE	<b>tbname in ('MYTESTS')</b>
DMMAXLOOKUPMAP	MAXLOOKUPMAP	<b>1=2</b>
DMMAXRELATIONSHIP	MAXRELATIONSHIP	<b>parent in ('MYTESTS') or child in ('MYTESTS')</b>
DMMAXINTOBJECT	MAXINTOBJECT	<b>intobjectname in ('MYMAXAPPS', 'MYMAXPRESENTATION')</b>
DMGLCONFIGURE	GLCONFIGURE	<b>1=2</b>

各オブジェクトの条件によって、このパッケージ用の正確なコンテンツが生成されます。この特定のケースではマイグレーションする関連コンテンツがないため、論理的な false ステートメント (**1=2**) で定義されたオブジェクトの *Where* 文節によって生成される結果はありません。

「**Where 文節の設定**」アイコンをクリックし、135 ページの表 5-2 に示すように SQL 条件を入力して、パッケージ定義を保存します。

## アプリケーション・パッケージ

次に、MYTESTS-APP という名前の新規パッケージ定義を作成します。これは、このシナリオ用に作成された新規マイグレーション・グループ MYNEWAPP が含まれるスナップショット・タイプのパッケージです。

次に、グループに含まれる各オブジェクト構造内に、SQL 基準を定義します。セクション 124 ページの『5.2.1 要件』で収集した情報を使用し、各オブジェクト(新規または変更済み)の名前を使用して SQL 条件を構成します。

表 5-3 に、各オブジェクトの *Where* 文節の SQL コマンド・ステートメント条件を示します。これらの条件によって、マイグレーション・オブジェクトごとに構成コンテンツのレコードが 1 つまたは複数生成されます。

表 5-3 MYTESTS-APP パッケージの *Where* 文節定義

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMCONDITION	CONDITION	<code>conditionnum in ('MYFAILED')</code>
DMMAXMODULES	MAXMODULES	<code>1=2</code>
MYMAXAPPS	MAXAPPS	<code>app in ('MYTESTS', 'ASSET')</code>
DMMAXMENU	MAXMENU	<code>moduleapp in ('MYTESTS') or keyvalue in ('MYTESTS')</code>
DMSCTEMPLATE	SCTEMPLATE	<code>1=2</code>
DMMAXLAUNCHENTRY	MAXLAUNCHENTRY	<code>1=2</code>
MYMAXPRESENTATION	MAXPRESENTATION	<code>app in ('LOOKUPS')</code>

各オブジェクトの条件によって、このパッケージ用の正確なコンテンツが生成されます。この特定のケースではマイグレーションする関連コンテンツがないため、論理的な false ステートメント (**1=2**) で定義されたオブジェクトの *Where* 文節によって生成される結果はありません。

「**Where 文節の設定**」アイコンをクリックし、表に示すように SQL 条件を入力して、パッケージ定義を保存します。

## セキュリティー・パッケージ

次に、MYTESTS-SEC という名前の新規パッケージ定義を作成します。これは、このシナリオ用に作成された新規マイグレーション・グループ MYAPPSECURITY が含まれるスナップショット・タイプのパッケージです。

次に、グループに含まれる各オブジェクト構造内に、SQL 基準を定義します。セクション 124 ページの『5.2.1 要件』で収集した情報を使用し、各オブジェクト（新規または変更済み）の名前を使用して SQL 条件を構成します。

表 5-4 に、各オブジェクトの *Where* 文節の SQL コマンド・ステートメント条件を示します。これらの条件によって、マイグレーション・オブジェクトごとに構成コンテンツのレコードが 1 つまたは複数生成されます。

表 5-4 MYTESTS-SEC パッケージの *Where* 文節定義

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMSIGOPTION	SIGOPTION	<b>1=2</b>
DMMAXSERVSECURITY	MAXSERVSECURITY	<b>1=2</b>
DMMAXGROUP	MAXGROUP	<b>groupname in ('TESTMGR', 'TESTTECH')</b>
DMMAXUSER	MAXUSER	<b>1=2</b>
DMSIGOPTFLAG	SIGOPTFLAG	<b>1=2</b>
DMCTRLGROUP	CTRLGROUP	<b>1=2</b>

各オブジェクトの条件によって、このパッケージ用の正確なコンテンツが生成されます。この特定のケースではマイグレーションする関連コンテンツがないため、論理的な *false* ステートメント (**1=2**) で定義されたオブジェクトの *Where* 文節によって生成される結果はありません。

「**Where 文節の設定**」アイコンをクリックし、表に示すように SQL 条件を入力して、パッケージ定義を保存します。

### 5.2.7 デプロイメント

すべてのパッケージ定義の作成、承認、アクティビ化を行ったら、各パッケージを作成してターゲットに配布します。ターゲット環境からパッケージ入手できるようになったら、デプロイを行います。

本書の目的のため、必要な変更プロトコルとリリース・プロトコル（存在する場合）はすべて設定済みで、ターゲット環境のユーザーにはデプロイメント期間中にシステムが使用できなくなることを通知済みであるものとします。

このユースケース・シナリオでは、以下のデプロイメント・ステップを実行します。

1. まず、マイグレーション・マネージャー・アプリケーションから管理モードをオンにします。  
構成コンテンツはデータ・ディクショナリーに変更を加えるだけでなく、UI のアプリケーション・プレゼンテーションにも影響を与えるため (MAXAPPS オブジェクトがパッケージに含まれるため)、デプロイメント中はシステムをこの操作モードにしておく必要があります。また、どのユーザーもこの環境に接続させないようにして、アプリケーション・インフラストラクチャーのトラフィックを最小量にすることをお勧めします。
2. システムが管理モードで操作中であることを確認したら、データ・ディクショナリー・パッケージ MYTESTS-DD をロードおよびデプロイします。エラーが発生した場合は操作を停止し、トラブルシューティング手順を実行してエラーを修正してください。エラーがない場合にのみ、次のステップに進んでください。
3. 最初のパッケージをエラーなくデプロイしたら、コンテンツがマイグレーションされていることを検証します。コンテンツの検証が完了するまでは、次に進まないでください。
4. 次に、アプリケーション・パッケージ MYTESTS-APP をロードおよびデプロイします。ここでも、エラーが発生した場合は操作を停止し、トラブルシューティング手順を実行してください。エラーがない場合にのみ、次のステップに進んでください。
5. 2番目のパッケージをエラーなくデプロイしたら、コンテンツがマイグレーションされていることを検証します。コンテンツの検証が完了するまでは、次に進まないでください。
6. 最後に、セキュリティー・パッケージ MYTESTS-SEC をロードおよびデプロイします。ここでも、エラーが発生した場合は操作を停止し、トラブルシューティング手順を実行してください。
7. すべてのコンテンツを検証したら、システムの管理モードをオフにして、ユーザーがログインできるようにします。

トラブルシューティングについて詳しくは、253 ページの『第 11 章 トラブルシューティング』を参照してください。

### 5.2.8 デプロイメントに関する考慮事項

以下に、このマイグレーションでのデプロイメントに関する考慮事項について説明します。

## 管理モード

データ・ディクショナリーやアプリケーション・プレゼンテーションなど、構造の変更を含むパッケージをデプロイする際には、管理モードでの操作をオンにしてください。単体の JVM サーバー環境では問題ありませんが、マルチサーバー環境(つまりクラスター)を使用している場合は、クラスター内のすべてのサーバーをこのモードに切り替える必要があります。

このモード中は、管理者の許可がなければ、どのユーザーもアプリケーションにログインすることができません。デプロイメントが完了したら、必ずこの操作モードをオフにしてください。

このトピックについて詳しくは、一般的なトピックを参照してください。

## 5.3 スタート・センターとクエリーのマイグレーション

スタート・センターは、ユーザーが製品にログインする際に表示される起動画面ポータルです。ユーザーがアクセスするスタート・センターは数種類あり、これらはセキュリティー構成に基づいています。スタート・センターのレイアウトとコンテンツは、アプリケーション UI と同じように XML 表示形式で格納されます。

このケース・シナリオでは、新しいスタート・センター・テンプレートと、結果セット用に構成されたアプリケーション・クエリーのマイグレーションについて検討します。

### 5.3.1 要件

前のセクションで検討したケース・シナリオでは、テスト結果用の新しいアプリケーションを作成しました。同じ要件の一部として、以下のような新しいスタート・センターをマイグレーションする必要があります。

- ▶ テスト・マネージャーと技術者用に新しい「テスト・ポータル」スタート・センターを作成し、これらのロールの対象として特定されたセキュリティー・グループにリンクさせる。
- ▶ このスタート・センターに、「お気に入りのアプリケーション」ポートレットと、新しいテスト結果アプリケーションおよび資産アプリケーションへのリンクを含める。
- ▶ このスタート・センターに、通常の「掲示板」ポートレットと「タスク」ポートレットを含める。
- ▶ このスタート・センターに結果セット用のポートレットを含め、このポートレットに、新しいテスト結果アプリケーションで作成した新規クエリーを含

める。このクエリーによって、テスト・テーブルをフィルタリングして結果が「失敗」のレコードに絞る。

この要件を満たすには、以下の最小構成変更を行います。

1. (5.2, 『メニュー、ルックアップ、グローバルなデータ制限などの複雑なアプリケーション変更』で作成した) テスト結果アプリケーションを使用して、「失敗」の結果をフィルタリングする新しいクエリーを作成します。このクエリーの文節名は以下のとおりです。

a. *FAILEDTEST*

2. スタート・センター・アプリケーションを使用して、左側に「お気に入りのアプリケーション」ポートレットが表示され、右側に「掲示板」、「インボックス / 割り当て」、「結果セット」の各ポートレットが表示される「ナローウィド」レイアウトの新規テンプレートを作成します。「結果セット」ポートレットは、テスト結果アプリケーション内の新しいクエリーを指すようにします。このテンプレートの名前は以下のとおりです。

a. *Template-20100927205646*

このケース・シナリオでは、既存のアプリケーションのクエリー機能と、スタート・センター・テンプレート構成を使用して、構成を作成しました。繰り返しますが、マイグレーション時には構成の作成順序を考慮することが非常に重要です。

### 5.3.2 ソリューション

このシナリオでの変更内容を確認したら、5.1, 『基本的なアプリケーション変更と署名オプション』の基本的なユースケース・シナリオと同様に、アプリケーション・グループを使用して1つのパッケージを使用するマイグレーション方法を取ります。

ただし、構成オブジェクトの内容を注意深く確認すると、クエリー・オブジェクトをマイグレーションするための新しいマイグレーション・オブジェクトを作成する必要があることに気付きました。このオブジェクトは、製品と共に提供されていません。この新しいオブジェクト構造を作成するには、「データ・ディクショナリー」マイグレーション・グループを使用して、このオブジェクト構造を別個のパッケージにマイグレーションする必要があります。

### 5.3.3 このソリューションに使用する構成アプリケーション

このソリューションに使用する以下の構成アプリケーションと、これらのアプリケーションにアクセスする方法を確認してください。

## テスト結果アプリケーション用のクエリー

クエリーを作成するには、「リンク先」→「資産」→「テスト結果」をクリックし、次に「クエリーの保存」をクリックします。

## スタート・センター・テンプレート

スタート・センター・テンプレートを変更または新規作成するには、「スタート・センター」→「新規テンプレートの作成 / テンプレートの変更 (Create New/Modify Template)」をクリックします。

## オブジェクト構造

オブジェクト構造アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」をクリックします。

### 5.3.4 オブジェクト構造

スタート・センター・テンプレートの構成コンテンツのマイグレーション用に、標準オブジェクト構造が特定されています。ただし、クエリーの構成コンテンツには特別な注意が必要です。

**ヒント:** 製品と共に提供されている標準オブジェクトと標準グループは、開発者によって作成された論理的なビジネス・オブジェクト構造と関連付けを反映しているため、これらを使用するようにしてください。

## 標準オブジェクト構造

このマイグレーション・グループで使用する標準オブジェクト構造は以下のとおりです。

- ▶ DMSCTEMPLATE
- ▶ DMMAXINTOBJECT

## クエリー・オブジェクト構造

クエリーは、アプリケーションとユーザー ID にリンクされます。コンテンツは QUERY テーブルに格納されます。

すべてのデフォルト値と以下の特性を使用して、MYQUERY という名前の新規マイグレーション・オブジェクト構造を作成します。

**オブジェクト構造**      MYQUERY

**説明**                    クエリー・オブジェクト構造

**使用者**                MIGRATIONMGR

デフォルト値はすべてそのまで、以下の特性を使用して、MYQUERY のソース・オブジェクトのリストに新規のソース・オブジェクトを追加します。

**オブジェクト**      QUERY  
**オブジェクト順序**    1( またはデフォルト )  
新規のオブジェクト構造を保存します。

図 5-10 に、新規の MYQUERY オブジェクト構造を示します。

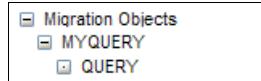


図 5-10 MYQUERY オブジェクト構造

ここで作成した新規オブジェクト構造は、その構成コンテンツをマイグレーションする必要があります。このマイグレーションには、標準オブジェクト構造 DMMAXINTOBJECT を使用します。

### マイグレーション・オブジェクト構造

以下に、マイグレーション要件のサポートに使用するオブジェクト構造の改訂リストを示します(特定の順序ではありません)。

- ▶ DMSCTEMPLATE
- ▶ DMMAXINTOBJECT
- ▶ MYQUERY

**注:** マイグレーション・オブジェクト構造について詳しくは、以下の URL で「マイグレーション・マネージャー・ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)

### 5.3.5 マイグレーション・グループ

次に、パッケージ定義に使用するマイグレーション・グループを定義します。オブジェクト構造を特定した後、新規オブジェクト構造を作成したため、2つのグループが使用されます。最初のグループでは、データ・ディクショナリーのコンテンツ(このケースでは新規オブジェクト構造の定義)がマイグレーションされます。次に、クエリー・コンテンツを組み込む若干の変更を加えて、アプリケーション・グループでスタート・センターのコンテンツがマイグレーションされます。

**ヒント:** 新規オブジェクト構造をマイグレーションしたら、次に、結果セット用のクエリーが組み込まれたスタート・センター・テンプレートをマイグレーションします。このマイグレーションでは、データ・ディクショナリー・グループは不要です。必要なのは、スタート・センター・テンプレートとクエリー用のグループ 1 つのみです。

このケース・シナリオに必要なグループは以下のとおりです。

### データ・ディクショナリー・グループ

最初のグループは標準の DATADICTONARY グループです。このデプロイメントで構成するオブジェクト構造は DMMAXINTOBJECT のみです。

### アプリケーション・グループ

前のシナリオと同様に、標準マイグレーション・グループを変更して、依存関係にあるグループを削除し、欠落している(ただしコンテンツに関連する)オブジェクト構造を追加します。

スタート・センター・テンプレートのコンテンツは、既にこの標準グループ(DMSCTEMPLATE オブジェクト構造)に含まれています。このグループに、クエリー・コンテンツを追加します。

マイグレーション・オブジェクト構造(Migration Object Structures) アプリケーションを使用して、標準の APPLICATIONS グループを複製して、新規グループを作成します。グループ名は MYAPPSC です。それ以外の構成は以下のとおりです。

1. 依存関係にあるグループをすべて削除します。
2. 5.3.4.『オブジェクト構造』で作成したオブジェクト構造 MYQUERY を追加します。このオブジェクトをリストの末尾に置いてください。
3. このグループを保存します。

図 5-11 に、新規作成されたグループ・オブジェクト・リストと、マイグレーション・オブジェクトの順序を示します。

Migration Objects for MYAPPSC		Filter	Search	Print	Up	Down	First	Last	Download
	Migration Object	Description	Migration Object Order		Intern				
▶	DMMAXMODUL	Application module migration object for Migration	1						
▶	DMMAXAPPS	Application migration object for Migration Manager	2						
▶	DMMAXMENU	Application menu migration object for Migration	3						
▶	DMSCTEMLAT	Start Center template migration object for Migration	4						
▶	DMMAXLAUNCI	Launch entry migration object for Migration Manager	5						
▶	MYQUERY	Query object for Migration	6						

図 5-11 新規マイグレーション・グループ MYAPPSC

### 5.3.6 パッケージ定義

このユースケース・シナリオでは、2つの個別のスナップショット・パッケージをデプロイするマイグレーション方法を取ります。最初のパッケージではデータ・ディクショナリー・コンテンツをマイグレーションし、2番目のパッケージではアプリケーション・コンテンツをマイグレーションします。

#### データ・ディクショナリー・パッケージ

MYSC-DD という名前の新規パッケージ定義を作成します。これは、マイグレーション・グループ DATAADCTIONARY が含まれるスナップショット・タイプのパッケージです。

次に、グループに含まれる各オブジェクト構造内に、SQL 基準を定義します。セクション 139 ページの『5.3.1 要件』で収集した情報を使用し、各オブジェクトの名前を使用して SQL 条件を構成します。

表 5-5 に、各オブジェクトの *Where* 文節の SQL コマンド・ステートメント条件を示します。これらの条件によって、マイグレーション・オブジェクトごとに構成コンテンツのレコードが 1つまたは複数生成されます。

表 5-5 MYSC-DD パッケージの *Where* 文節定義

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMMAXSERVICE	MAXSERVICE	1=2
DMLANGUAGE	LANGUAGE	1=2
DMMAXMESSAGES	MAXMESSAGES	1=2

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMCURRENCY	CURRENCY	<b>1=2</b>
DMSETS	SETS	<b>1=2</b>
DMORGANIZATION	ORGANIZATION	<b>1=2</b>
DMMAXVARS	MAXVARS	<b>1=2</b>
DMCONDITION	CONDITION	<b>1=2</b>
DMMAXDOMAIN	MAXDOMAIN	<b>1=2</b>
DMMAXOBJECTCFG	MAXOBJECTCFG	<b>1=2</b>
DMMAXSEQUENCE	MAXSEQUENCE	<b>1=2</b>
DMMAXLOOKUPMAP	MAXLOOKUPMAP	<b>1=2</b>
DMMAXRELATIONSHIP	MAXRELATIONSHIP	<b>1=2</b>
DMMAXINTOBJECT	MAXINTOBJECT	<code>intobjectname in ('MYQUERY')</code>
DMGLCONFIGURE	GLCONFIGURE	<b>1=2</b>

各オブジェクトの条件によって、このパッケージ用の正確なコンテンツが生成されます。この特定のケースではマイグレーションする関連コンテンツがないため、論理的な `false` ステートメント (**1=2**) で定義されたオブジェクトの *Where* 文節によって生成される結果はありません。

「**Where 文節の設定**」アイコンをクリックし、144 ページの表 5-5 に示すように SQL 条件を入力して、パッケージ定義を保存します。

## アプリケーション・パッケージ

次に、MYSC-APP という名前の新規スナップショット・パッケージ定義を作成します。このパッケージには、このシナリオ用に作成された新規マイグレーション・グループ MYAPPSC が含まれます。

次に、グループに含まれる各オブジェクト構造内に、SQL 基準を定義します。セクション 139 ページの『5.3.1 要件』で収集した情報を使用し、各オブジェクトの名前を使用して SQL 条件を構成します。

146 ページの表 5-6 に、各オブジェクトの *Where* 文節の SQL コマンド・ステートメント条件を示します。これらの条件によって、マイグレーション・オブジェクトごとに構成コンテンツのレコードが 1 つまたは複数生成されます。

表 5-6 MYTESTS-APP パッケージの Where 文節定義

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMMAXMODULES	MAXMODULES	<b>1=2</b>
MYMAXAPPS	MAXAPPS	<b>1=2</b>
DMMAXMENU	MAXMENU	<b>1=2</b>
DMSCTEMPLATE	SCTEMPLATE	<code>name in ('Template-20100927205646')</code>
DMMAXLAUNCHENTRY	MAXLAUNCHENTRY	<b>1=2</b>
MYQUERY	QUERY	<code>clausename='FAILEDTESTS' and app='MYTESTS'</code>

各オブジェクトの条件によって、このパッケージ用の正確なコンテンツが生成されます。この特定のケースではマイグレーションする関連コンテンツがないため、論理的な false ステートメント (**1=2**) で定義されたオブジェクトの Where 文節によって生成される結果はありません。

「Where 文節の設定」アイコンをクリックし、表に示すように SQL 条件を入力して、パッケージ定義を保存します。

### 5.3.7 デプロイメント

2つのパッケージ定義の作成、承認、アクティブ化を行ったら、各パッケージを作成してターゲットに配布します。ターゲット環境からパッケージ入手できるようになり、変更管理の準備をすべて実行したら、デプロイを行います。

このユースケース・シナリオでは、以下のデプロイメント・ステップを実行します。

- まず、マイグレーション・マネージャー・アプリケーションから管理モードをオンにします。  
データベース構成または画面表示の更新を伴うデプロイメントの場合は、必ずこの操作が必要となります。
- システムが管理モードで操作中であることを確認したら、データ・ディクショナリー・パッケージ MYSC-DD をロードおよびデプロイします。エラーが発生した場合は操作を停止し、トラブルシューティング手順を実行してエラーを修正してください。エラーがない場合にのみ、次のステップに進んでください。

3. 最初のパッケージをエラーなくデプロイしたら、コンテンツがマイグレーションされていることを検証します。コンテンツの検証が完了するまでは、次に進まないでください。
4. 最後に、アプリケーション・パッケージ(MYSC-APP)をロードおよびデプロイします。ここでも、エラーが発生した場合は操作を停止し、トラブルシューティング手順を実行してください。エラーがない場合にのみ、次のステップに進んでください。
5. すべてのコンテンツを検証したら、システムの管理モードをオフにして、ユーザーがログインできるようにします。

トラブルシューティングについて詳しくは、253 ページの『第 11 章 トラブルシューティング』を参照してください。

### 5.3.8 デプロイメントに関する考慮事項

ユーザー定義のマイグレーション・オブジェクトを含むパッケージの承認、作成、またはデプロイ時には、マイグレーション・マネージャーに図 5-12 のようなシステム・メッセージが表示されます。

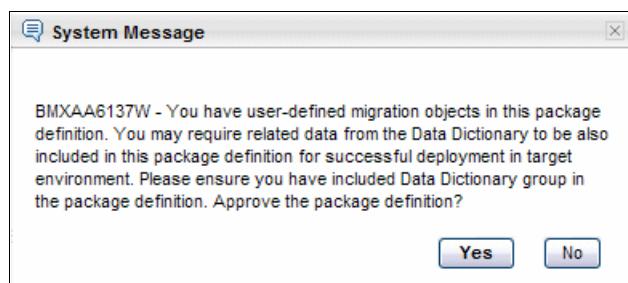


図 5-12 システム・メッセージ BMXAA6137W

先に進むために、必ず「**はい**」をクリックしてください。





# ワークフローのマイグレー ション

ワークフローとは、*Maximo* ビジネス・オブジェクト(MBO) 上に階層化された、定義済みのビジネス・プロセスを表します。ワークフローを広範囲に使用することで、業務のさまざまな局面が自動化されます。例えば、ワークフロー・プロセスを使用して、エンド・ユーザーが送信した購買要求書や変更要求の承認のルーティングを自動化できます。

最も単純な形式のワークフロー・プロセスは、送信されたレコードの基準に基づく、ユーザーとの対話を必要としない条件付きアクションで構成されます。より複雑なワークフロー・プロセスは、サブプロセスに分割することができ、このようなプロセスを使用すると、複数の割り当て、アクション(ステータスの更新など)、電子メールやポケベルを通じた通知などを含む高度な対話式の変更管理承認プロセスを、あらかじめ定義された通信テンプレートによって管理できます。

この章は以下のセクションで構成されています。

- ▶ 150 ページの『6.1 ワークフローに対する特定の変更のマイグレーション』
- ▶ 158 ページの『6.2 ワークフローとすべての関連コンテンツのマイグレー  
ション』

## 6.1 ワークフローに対する特定の変更のマイグレーション

ワークフローは、通常、開発環境で定義された後、テスト環境や実稼働環境にプロモートされます。ワークフロー・プロセスのマイグレーションを確実(エラーや障害を発生させずに)かつ正確に(必要なコンテンツを喪失せずに)実行するためには、反復可能なプロセスが必要です。

### 6.1.1 要件

このケース・シナリオでは、変更前のワークフローとして PRSTATUS を使用します。このシナリオの要件は、アクション行に変更(要求者に要求のステータスを通知する通信テンプレートなど)を加えたこのワークフローの複製をマイグレーションすることです。

### 6.1.2 ソリューション

この事例では、ターゲット環境に元のワークフロー (PRSTATUS) が既に存在する状況で、既存のワークフローに基づく新しいワークフローをマイグレーションする必要があります。このため、マイグレーションの対象となるのは、新しいワークフロー・プロセスと、追加されたコンポーネントのみです。変更のない既存のコンポーネントは、マイグレーションの対象ではありません。

ここでは、ワークフロー構成とそれに関連する新しい構成を、单一のスナップショット・マイグレーション・パッケージでマイグレーションするように、解決方法が定義されています。

コンテンツの観点から見ると、ワークフローのマイグレーションは、ワークフロー構成を以下の順序でマイグレーションすることも意味します。

1. アクション
2. アクション・グループ(データ・ディクショナリーをマイグレーションする場合)
3. ロール
4. 通信テンプレート
5. ワークフロー・プロセス

マイグレーションの対象コンテンツを定義する際のベースとしては、標準のワークフロー・グループである *BPM* を使用します。このグループには、オブジェクト構造 *INBOUNDCOMMCONFIG* および *ESCALATIONS* も含まれています。これらの構造は、ここで説明するマイグレーション・ユースケースに対応するために必要ではありませんが、グループ定義からは削除しません(ただし、SQL クエリー・フィルターを使用して除外します)。

さらに、ワークフローでは、担当者またはロール・プレイヤー(担当者または担当者グループとして定義)を対象とした割り当ておよび通信の各テンプレートを使用して、ユーザーとの対話も指定されています。この文書では、『マイグレーション戦略』セクションで示すように、これらの依存関係やその他の依存関係もマイグレーションされているか、ターゲット・システムにロードされているものとします。

### 6.1.3 このソリューションで使用する構成アプリケーション

このセクションでは、例に対応するために変更を加えるときに使用するアプリケーションについて説明します。

ワークフロー・デザイナー・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「ワークフロー・デザイナー」の順にナビゲートします。

ワークフロー・デザイナーから、ワークフロー PRSTATUS を開いて複製し、*REDBOOK\_WF* という名前を付けます。これをバージョン 1 として保存します。

ここでの目的は、*SUB\_APPR* タスクと *CLOSE* タスクの間の接続線内のアクションに、通知を追加することです(図 6-1.. 参照)。

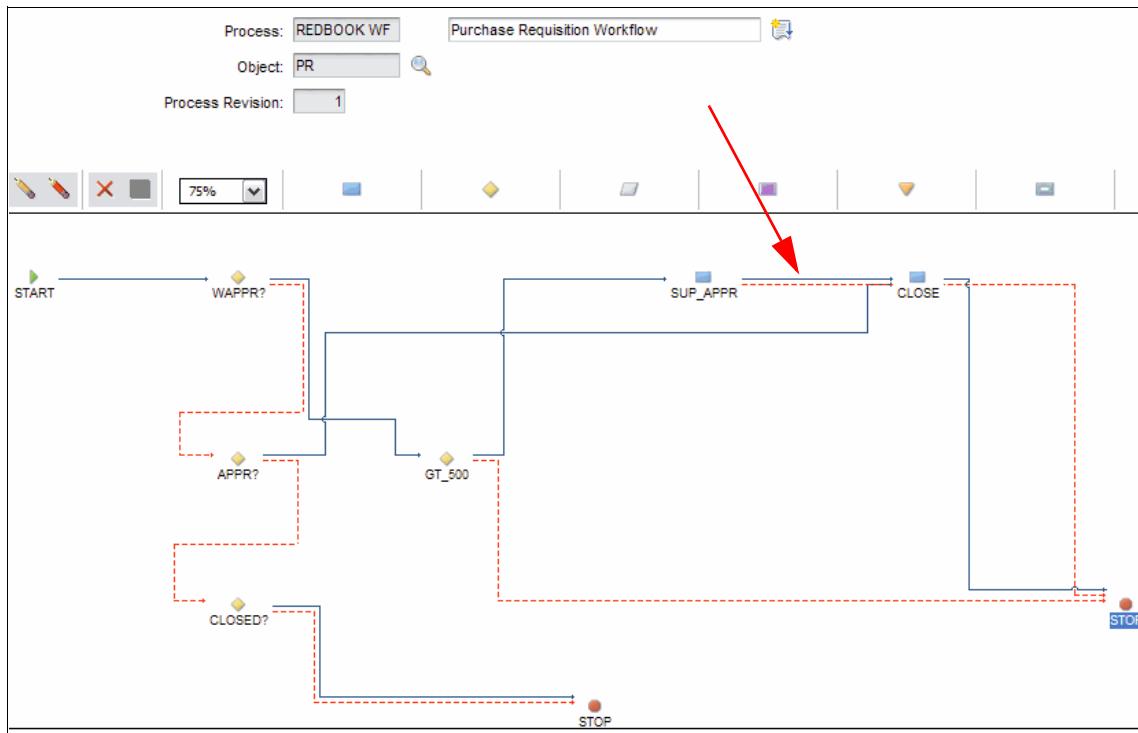


図 6-1 ワークフロー・デザイナーのキャンバス

この通知には、新しい通信テンプレート *RB\_REQ\_APP* が含まれます(153 ページの図 6-2 参照)。

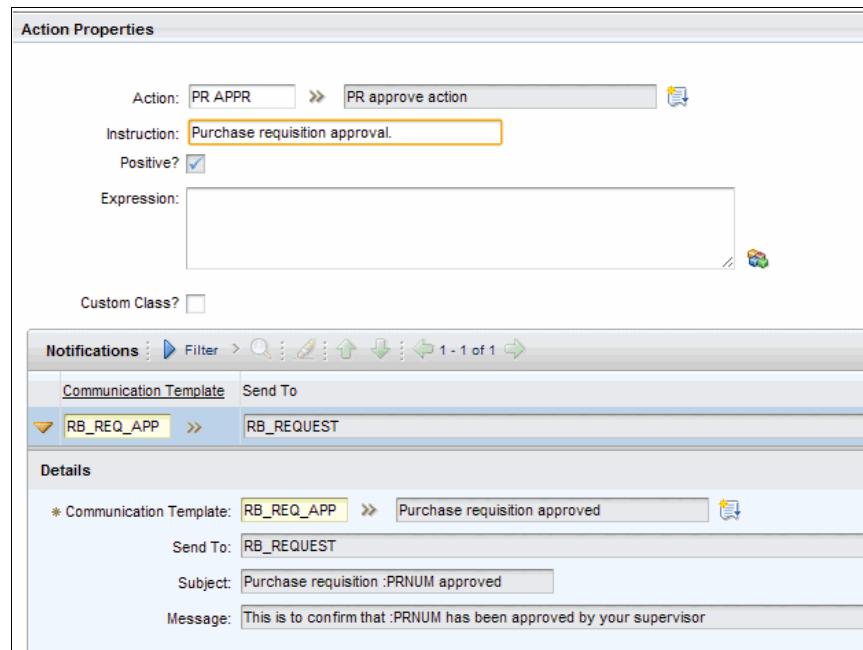


図 6-2 接続線アクションのプロパティー

後述する選択マイグレーション・プロセスに対応するために、接頭部 *RB* を付ける命名規則が適用されます。このことは、この章で後述する選択マイグレーション・プロセスに対応するための措置であり、特に重要です。

通知の管理は、通信テンプレート・アプリケーションを使用して行います。通信テンプレートにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「通信テンプレート」の順にナビゲートします。

通信テンプレートでは、1つ以上の宛先を、電子メール・アドレス、ロール、担当者、または担当者グループの形で定義できます。この例では、通信テンプレート *RB\_REQ\_APP* を、2つのタスク・ノード間の接続線内のアクション *PR\_APPR* に追加します。さらに、購買要求書の要求者にロールを関連付けます。154 ページの図 6-3 を参照してください。

Template: RB\_REQ\_APP Purchase requisition approved

\* Applies To: PR

\* Accessible From: ALL

Comm Log Entry?

To: RB\_REQUEST

cc:

bcc:

\* Send From: maxadmin@us.ibm.com

Reply To:

Subject: Purchase requisition :PRNUM approved

Message: This is to confirm that :PRNUM has been approved by your supervisor

図 6-3 通信テンプレート

通信テンプレートをロールに関連付ける場合、ロール・アプリケーションを使用して、そのロールを定義する必要があります。ロール・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「ロール」の順にクリックします。ロールは、以前に定義した担当者または担当者グループのレコード、または担当者レコードに関連付けられたロールに基づいて作成できます。この例では、レコードに関連付けられたロール・プレイヤー(PR.Requestedby)を使用することで、以前にロール・ベースの方法でロードした購買要求書レコードを使用できるようになります。この場合、購買要求書の要求者がロール・プレイヤーになります。図 6-4 を参照してください。

Role: RB\_REQUEST Requester of requisition

\* Type: A set of data relate

\* Object: PR

\* Value: :requestedby

Parameter:

E-mail?

Broadcast?

図 6-4 ロール

## 6.1.4 オブジェクト構造

このユースケースのマイグレーションに対応するには、以下の標準的なオブジェクト構造(図 6-5 参照)が必要です。

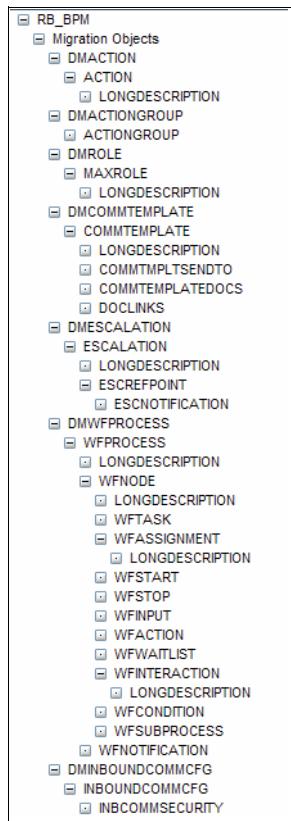


図 6-5 RB\_BPM のオブジェクト構造

### 6.1.5 マイグレーション・グループ

オブジェクト構造は、1つのマイグレーション・グループ、つまり *RB\_BPM* に含まれています。このワークフローのマイグレーション・シナリオでは、すぐに使用可能なマイグレーション・グループを使用することは効率的ではありません。このようなグループには、デプロイメントが失敗しないように用意された他のグループへの依存関係が含まれているからです。作業対象のシナリオでは、ターゲットにワークフロー要素が既に含まれており、このマイグレーションを行う前に、依存関係オブジェクトをマイグレーションするための予防措置が実行されています。

既存の *BPM* グループを複製して、新しいマイグレーション・グループを作成します。依存関係は単純に削除し、オブジェクトの順序はそのまま維持します。

156 ページの図 6-6 に、作成したマイグレーション・グループ *RB\_BPM* を示します。このグループには、このワークフローのマイグレーションに対応するために必要なオブジェクト構造のみが含まれています。

Migration Object	Description	Migration Object Order	Internal?
DMACTION	Action migration object for Migration Manager	1	<input type="checkbox"/>
DMACTIONGRO	Actiongroup migration object for Migration Manager	2	<input type="checkbox"/>
DMROLE	Role migration object for Migration Manager	3	<input type="checkbox"/>
DMCOMMTEMPL	Communication template migration object for Migration Manager	4	<input type="checkbox"/>
DMESCALATIO	Escalation migration object for Migration Manager	5	<input type="checkbox"/>
DMWFPROCESS	Workflow process definition migration object for Migration Manager	6	<input type="checkbox"/>
DMINBOUNDCO	E-mail listener definition migration object for Migration Manager	7	<input type="checkbox"/>

図 6-6 *RB\_BPM* グループのマイグレーション・オブジェクト

### 6.1.6 パッケージの定義

この時点では、構成コンテンツをアセンブルおよび編成するために必要なコンポーネントは揃っています。このプロセスの次の手順では、マイグレーション・マネージャー・アプリケーションを使用して、スナップショット・マイグレーション・パッケージを定義します。このユースケースで特定したすべての構成コンテンツをマイグレーションするには、単一のパッケージ定義で十分です。マイグレーション・パッケージの定義を作成するための作業については、製品資料に詳しく記載されているため、ここでは説明しません。このパッケージ定義に含まれるグループは、新しく作成したマイグレーション・グループ *RB\_BPM* のみです。

ここでは、新しいパッケージの名前も *RB\_BPM* とします。*RB\_BPM* パッケージ定義を保存した後は、グループ内の各オブジェクト構造に、必要に応じて SQL フィルター基準を関連付けることで、マイグレーションが必要な構成のサブセットを特定できます。

最適な SQL 基準を判断するために、ワークフローの内容を検討する方法の例を、以下に示します。

1. 変更した接続線のアクションに、通信テンプレート *RB\_REQ\_APP* を含める必要があるため、このテンプレートが、対応する新規ロール *RB\_REQUEST* と共に追加されています。
2. ワークフロー・デザイナー・アプリケーションの「ワークフロー」フィルードに、作成したワークフローのプロセス名として *REDBOOK\_WF* が指定されているため、これを SQL 文節で使用します。ワークフローには多くのリビジョンがありますが、そのうちアクティブなものは 1 つのみであるため、ワークフローネームを指定するときは、ステータス ACTIVE も指定します。
3. さまざまなオブジェクト構造の SQL 基準を指定するには、マイグレーション・マネージャー・アプリケーションの「**Where 文節の設定**」機能を使用します。Where 文節を指定すると、エクスポート対象となる特定の構成レコードを識別できます。エクスポートが不要な場合は、文節 1=0 または 1=2 を指定します。
4. ワークフローをマイグレーションするために必要な SQL 基準を入力したら、パッケージ定義を保存できます(図 6-7 参照)。

Migration Object	Object	Where Clause	
DMACTION	ACTION	1=2	
DMACTIONGRO	ACTIONGROUP	1=2	
DMROLE	MAXROLE	maxrole = 'RB_REQUEST'	
DMCOMMTEML	COMMTEMPLATE	templateid = 'RB_REQ_APP'	
DMESCALATIO	ESCALATION	1=2	
DMWFPROCES	WFPROCESS	processname = 'RB_BPM'	
DMINBOUNDCO	INBOUNDCOMMCFG	1=2	

図 6-7 パッケージの Where 文節の SQL クエリー

### 6.1.7 デプロイメント

パッケージ定義の保存と承認が完了したら、マイグレーション・マネージャー・アプリケーションを使用して、物理パッケージを作成し、これを配布およびデプロイできます。ワークフローとそれに関連付けられた構成の稼動環境へのデプロイメントは、アプリケーション・サーバーを再始動したり、管理モードをオンにしたりすることなく実行できます。

## 6.1.8 デプロイメントに関する考慮事項

ワークフローのマイグレーションを実行する前に、以下のマイグレーション・オプションを検討することが重要です。

### バージョン

複数バージョンのワークフロー・プロセスを開発、テストすることは、よく行われる手法です。ただし、マイグレーションされるのは選択したアクティブ・バージョンであり、ターゲット環境ではこのバージョンが「1」にリセットされます。

### アクティブ化

アクティブ化手順は手動で行う必要がありますが、これはワークフローのコンテンツを検証するよい機会でもあり、ターゲット環境で、ワークフローをタイミングよくアクティブ化するだけで済みます。

## 6.2 ワークフローとすべての関連コンテンツのマイグレーション

このシナリオでは、変更された既存のワークフローのマイグレーションについて説明します。このワークフローには、メイン・プロセスから呼び出されるサブプロセスも含まれています。ここでは、サブプロセスとすべてのワークフロー・コンポーネントを同じマイグレーション・パッケージに含める方法を示します。

### 6.2.1 要件

この例では、メイン・プロセス ISMACCEPT と、そのすべてのサブプロセスが、大幅に変更されています。変更要件を満たすために、多くのアクション、ロール、および通信テンプレートが変更され、プロセスに追加されています。

しかし、正確な変更内容を記載した資料はなく、マイグレーション・マネージャーの変更パッケージを使用したこれらの構成変更の追跡も行われていません。わかっていることは、ワークフロー・プロセス ISMACCEPT の構成が変更されたため、マイグレーションの必要があるということだけです。

ワークフロー・プロセスの変更を、確実（エラーや障害を発生させずに）かつ正確に（必要なコンテンツを喪失せずに）マイグレーションするためには、復可能なプロセスが必要です。

## 6.2.2 ソリューション

このケース・シナリオのマイグレーション要件を検討した結果、スナップショット・マイグレーション・パッケージを使用して、関連するすべての構成コンテンツをマイグレーションするように、解決方法が定義されています。パッケージは、ビジネス・プロセス・マネジメント(BPM)用の標準的なマイグレーション・グループ(Tivoli's process automation engine (Tivoli/Maximo Base Service)によってインストールされます)を使用するように構成します。

コンテンツの観点から見ると、変更されたワークフローと関連コンテンツをマイグレーションするには、以下の構成オブジェクトをマイグレーションする必要があります。

1. ロール
2. アクション
3. 通信テンプレート
4. ワークフロー・プロセスおよびサブプロセス

## 6.2.3 このソリューションで使用する構成アプリケーション

このソリューションで使用する構成アプリケーションと、これらのアプリケーションにアクセスする方法を、以下に示します。

### ロール

ロール・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「ロール」の順にクリックします。

### アクション

アクション・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「アクション」の順にクリックします。

### 通信テンプレート

通信テンプレート・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「通信テンプレート」の順にクリックします。

### ワークフロー

ワークフロー・デザイナー・アプリケーションにアクセスするには、「リンク先」→「システムの構成」→「プラットフォームの構成」→「ワークフロー・デザイナー」の順にクリックします。

## 6.2.4 オブジェクト構造

このマイグレーション要件に対応するために、製品に用意された以下の標準的なオブジェクト構造を使用します。

- ▶ DMACTION
- ▶ DMACTIONGROUP
- ▶ DMROLE
- ▶ DMCOMMTEMPLATE
- ▶ DMWFPROCESS

**注:**マイグレーション・オブジェクト構造について詳しくは、以下の URL にある「マイグレーション・マネージャー・ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)

## 6.2.5 マイグレーション・グループ

変更要件の検討とオブジェクト構造の特定が完了したら、次はパッケージ・コンテンツの定義に使用するグループを定義します。

製品と共に提供されている標準グループ BPM には、前述のマイグレーション・オブジェクトが含まれています。ここでは、シナリオに対応するために、このグループを基にして新しいグループを作成します。

まず、標準の BPM グループを、すべてデフォルトのままで複製して、新しいグループを作成します。次に、以下の作業を行います。

1. グループに **MYBPM** という名前を付け、「依存関係のないビジネス・プロセス・モジュール」という説明を記載します。
2. すべての依存関係グループを削除します。
3. グループを保存します。

複製した BPM グループから既存のオブジェクトを削除しないままにすることで、今後、他の類似の構成（エスカレーションなど）をマイグレーションするときに、この新しいグループを使用できます。

また、依存関係グループを削除することで、これらのグループに含まれるすべての構成コンテンツを除外します。ただし、マイグレーション・プロセスには、必ず依存関係が存在します。このケース・シナリオでは、操作対象が既存のワークフローであるため、依存関係にあるワークフロー要素を含む構成が、ターゲット環境に既に存在すると想定しています。例えば、新しい表属性が作成された場合や、ロールに新しい担当者グループが追加された場合は、操作対

象のワークフローをマイグレーションする前に、これらの構成をマイグレーションする必要があります。

図 6-8 は、新しいマイグレーション・グループを示しています。

Migration Group: MYBPM		BPM without dependency groups	Change By: MAXADMIN		
* Migration Group Order: 209		Change Date: 10/8/10 13:55:44			
Internal? <input type="checkbox"/>					
Migration Objects for MYBPM					
	Migration Object	Description	Migration Object Order		
▶	DMACTION	» Action migration object for Migration Manager	1		
▶	DMACTIONGRO	» Actiongroup migration object for Migration Manager	2		
▶	DMROLE	» Role migration object for Migration Manager	3		
▶	DMCOMMTEMPL	» Communication template migration object for Migration Manager	4		
▶	DMESCALATIO	» Escalation migration object for Migration Manager	5		
▶	DMWFPROCES	» Workflow process definition migration object for Migration Manager	6		
▶	DMINBOUNDCO	» E-mail listener definition migration object for Migration Manager	7		
<a href="#">New Row</a>					
Dependency					
	Dependent Migration Group	Description			
...No rows to display...					

図 6-8 新しいマイグレーション・グループMYBPM

## 6.2.6 パッケージの定義

このユースケース・シナリオでは、1つのスナップショット・パッケージをデプロイすることでマイグレーションを行います。

新しいパッケージ定義を作成し、MYTESTS-WF という名前を付けます。これはスナップショット・タイプのパッケージで、マイグレーション・グループ MYBPM が含まれています。

次に、グループの各オブジェクト構造内で SQL 基準を定義します。この基準を作成するために必要な情報は、メイン・ワークフローの名前(ISMACCEPT)のみです。関連する他のワークフロー依存関係(ロール、アクション、テンプ

レートなど)の条件は、各オブジェクト内の情報を取得するために作成されたインライン SQL ステートメントを使用して決定されます。

表 6-1 に、各オブジェクトの *Where* 文節 SQL コマンド・ステートメントの条件をリストします。これらの条件を指定すると、各マイグレーション・オブジェクトの構成コンテンツの 1 つ以上のレコードが結果として返されます。

表 6-1 MYTESTS-WF パッケージの *Where* 文節の条件

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMACTION	ACTION	<code>action in (select action from ACTIONGROUP where action in (select action from WFACTION where processname in ('ISMACCEPT') or processname in (select subprocessname from wfsubprocess where processname in ('ISMACCEPT')))) or action in (select member from actiongroup where action in (select action from WFACTION where processname in ('ISMACCEPT') or processname in (select subprocessname from wfsubprocess where processname in ('ISMACCEPT')))))</code>
DMACTIONGROUP	ACTIONGROUP	<code>action in (select action from WFACTION where processname in ('ISMACCEPT') or processname in (select subprocessname from wfsubprocess where processname in ('ISMACCEPT')))</code>
DMROLE	MAXROLE	<code>maxrole in (select sendtovalue from COMMPLTSENTO where type='ROLE' and templateid in (select templateid from COMMTEMPLATE where templateid in (select templateid from WFNOTIFICATION where processname in ('ISMACCEPT') or processname in (select subprocessname from wfsubprocess where processname in ('ISMACCEPT')))))</code>
DMCOMMTEMPLATE	COMMTEMPLATE	<code>templateid in (select templateid from WFNOTIFICATION where processname in ('ISMACCEPT') or processname in (select subprocessname from wfsubprocess where processname in ('ISMACCEPT')))</code>
DMESCALATION	ESCALATION	<code>1=2</code>
DMWFPROCESS	WFPROCESS	<code>(processname in ('ISMACCEPT') and active=1) or processname in (select subprocessname from wfsubprocess where processname in ('ISMACCEPT'))</code>
DMINBOUNDCOMMCFG	INBOUNDCOMMCFG	<code>1=2</code>

各オブジェクトの条件によって、このパッケージの正確なコンテンツが生成されます。オブジェクトの *Where* 文節で論理 False ステートメント (`1=2`) が定義されると、結果が生成されません。この場合、エスカレーションまたはインバウンド通信のコンテンツが、マイグレーションの対象とならないためです。

### 6.2.7 デプロイメント

すべてのパッケージ定義の保存、承認、およびアクティビ化が完了したら、次はパッケージを作成してターゲット環境に配布します。ターゲット環境でパッケージが使用可能な状態になったら、そのパッケージをデプロイします。

ターゲット環境に MYTESTS-WF パッケージをロードしてデプロイします。デプロイメント時に管理モードをオンにする必要はありませんが、変更管理プロセスの一環として、管理モードをオンにすることをお勧めします。

### 6.2.8 デプロイメントに関する考慮事項

今回の更新対象は既存のパッケージであるため、ターゲット環境では、このワークフローの下位にトランザクションが存在する可能性があります。影響を受けるレコードでは、以前のバージョンのワークフローが引き続き使用されます。

この場合、新しいワークフロー・バージョンはアクティブ化されません。以前のワークフローを使用するレコードがワークフロー・プロセスから外れると、新しいワークフローをアクティブ化できるようになります。

更新されたワークフローは、非アクティブかつ使用不可の状態でデプロイされ、サブプロセスも有効化されません。デプロイメントを完了するには、以前のバージョンのワークフローを無効化および非アクティブ化する必要があります。





## 分類のマイグレーション

業務データは日々増加します。レポート作成を容易にし、データの理解を深めるために、ビジネス・ニーズに応じてデータを分類する必要があります。この章では、マイグレーション・マネージャーを使用した分類データのマイグレーションに関する情報を説明します。この章には以下のセクションがあります。

- ▶ 166 ページの『7.1 シナリオ』
- ▶ 167 ページの『7.2 構成アプリケーション』
- ▶ 169 ページの『7.3 オブジェクト構造』
- ▶ 171 ページの『7.4 マイグレーション・グループ』
- ▶ 171 ページの『7.5 パッケージ定義』
- ▶ 173 ページの『7.6 デプロイメント』
- ▶ 173 ページの『7.7 デプロイメントに関する考慮事項』

## 7.1 シナリオ

分類は、データのグループ化および分類を可能にする階層データです。ユーザーは分類データを使用することで、より簡単に問題分布のレポートを作成し理解することができます。分類は多くのアプリケーションで使用できます。また、それぞれの分類に異なる属性を定義することもできます。属性とはデータの特性です。属性はデータの整合性を保ち、データを検索するうえで役立ちます。例えば、2つの異なる分類があるとします。1つはサービス分類 NET.NETWORKSERVICE で、もう1つはアプリケーション・サーバー分類 APP.J2EE.WEBSPHERE.WEBSPHERESERVER です。分類の属性の例を表 7-1 に示します。

表 7-1 分類の属性の例

NET.NETWORKSERVICE 分類の属性	APP.J2EE.WEBSPHERE.WEBSPHERESERVER 分類の属性
サービス名	ステータス
コンテキスト IP	製品名
双方向 (BIDI) フラグ	製品バージョン

これらの属性を使用して、構成アイテム (CI) を検索することができます。例えば、ステータスがアクティブでバージョンが 6.0.2.1 の CI を分類に従って検索することができます。

データの整合性を確保するために、属性に異なるドメインが付加されている場合があります。詳しくは、29 ページの『第 2 章 データ・ディクショナリーのマイグレーション』を参照してください。

データをグループ化するための分類は、グループ化のニーズに従って作成する必要があります。

### 7.1.1 要件

分類データは階層データです。分類は最大で 1 つの上位階層を持つことができます。しかし、下位階層分類は無限を持つことができます。分類は開発環境から実稼働環境へと順番に転送する必要があります。プロモート対象の分類は、中断されることなくマイグレーションされる必要があります。

### 7.1.2 ソリューション

分類データのマイグレーションでは、すぐに使用可能な *DMCLASSIFICATION* オブジェクト構造をそのまま使用することはできません。分類データに下位階層および上位階層が存在し、このオブジェクト構造では階層データ構造に対して十分ではないからです。下位階層データおよび上位階層データが、ソース環境でランダムな順序で作成された可能性があります。これにより分類がランダムな順序でエクスポートされる可能性があります。下位階層分類が上位階層分類よりも先にエクスポートされた場合、マイグレーション処理で下位階層分類の処理中にデータベース内に上位階層分類が見つからないため、エラーになります。

この状況を回避するには、データを正しくマイグレーションするための新しいオブジェクト構造を作成します。

新しいオブジェクト構造は再帰的にする必要があります。最上位分類から始まらなければなりません。マイグレーション・マネージャーは、最上位レベルを処理するときに、下位レベルの分類がなくなるまで、次のレベルの処理を行う必要があります。

分類データには、組織、サイト、または担当者が所有者グループとして設定され、分類データ自体にサービス・グループが付加されている場合があります。分類データをマイグレーションする前にこれらをマイグレーションしておく必要があります。そうしないと分類のマイグレーション中に検証エラーが発生する可能性があります。

## 7.2 構成アプリケーション

分類アプリケーションでは、分類を定義、管理および削除することができます。分類アプリケーションを使用するには、「リンク先」→「管理」→「分類」にナビゲートします。

図 7-1 に分類の階層の例を示します。

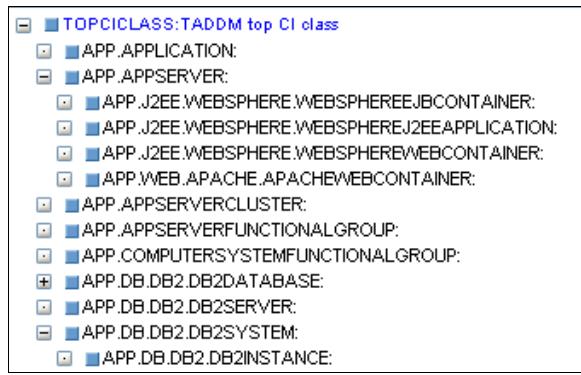


図 7-1 分類の階層の例

組織、担当者グループ、ドメイン、計測単位、およびサービス・グループのオブジェクトを分類または属性定義で使用した場合は、分類のマイグレーションの前にこれらをマイグレーションする必要があります。

168 ページの図 7-2 にサポート・オブジェクトを示します。

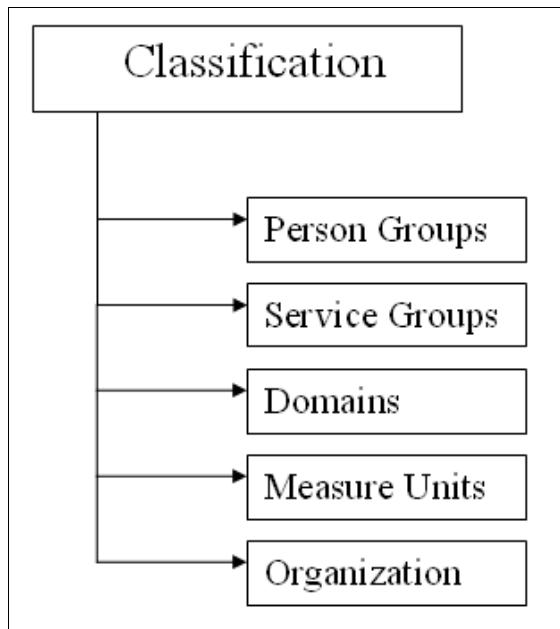


図 7-2 分類データのサポート・オブジェクト

## 7.3 オブジェクト構造

分類データをマイグレーションするには、製品と共に提供されている *DMCLASSIFICATION* オブジェクト構造を複製して変更する必要があります。

すぐに使用可能なオブジェクト構造 *DMCLASSIFICATION* を図 7-3 に示します。

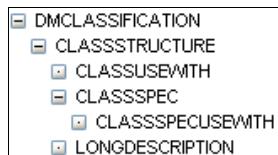


図 7-3 *DMCLASSIFICATION* オブジェクト構造

オブジェクト構造を複製するには、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」にナビゲートします。

*DMCLASSIFICATION* オブジェクト構造を選択し、図 7-4 に示すように「アクションの選択 (Select Action)」メニューから「オブジェクト構造の複製 (Duplicate Object Structure)」を選択します。

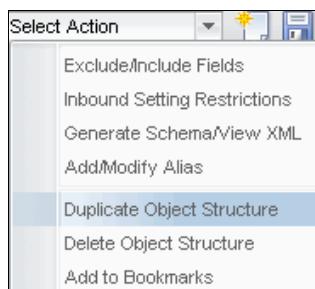


図 7-4 「アクションの選択」メニューの複製オプション

複製する際に、*MYCLASSIFICATION*などの名前を付け、次に 170 ページの図 7-5 に示すように「自己参照 (Self Reference)」オプションを選択して、再帰的なオブジェクト構造にします。

Object Structure:	MYCLASSIFICA	Classification migration object for Migration Ma		Query Only? <input type="checkbox"/>
* Consumed By:	MIGRATIONMGF		Migration Manager Application	<input checked="" type="checkbox"/> Self Reference?
Outbound Definition Class:				<input type="checkbox"/> Support Flat Structure?
Inbound Processing Class:	psdi.dn.procclass.DMMaxClassificationProces			

図 7-5 オブジェクト構造の「自己参照」オプション

チェック・ボックスにチェック・マークを付けると、オブジェクト構造の最初のオブジェクトの横にある「関係 (Relationship)」フィールドが編集可能になります。図 7-6 に示すように、最初のオブジェクトの横にある「関係 (Relationship)」フィールドに「下位階層 (CHILDREN)」と入力します。

Source Objects for MYCLASSIFICATION			
Object	Parent Object	Object Location Path	Relationship
CLASSSTRUCT		CLASSTRUCTURE	CHILDREN
CLASSUSEWITH	CLASSSTRUCT	CLASSTRUCTURE/CLASSUSEWITH	CLASSUSEWITH
CLASSSPEC	CLASSSTRUCT	CLASSTRUCTURE/CLASSSPEC	CLASSSPEC
CLASSSPECUS	CLASSSPEC	CLASSTRUCTURE/CLASSSPEC/CLASSSPEC	CLASSSPECUSEWITH
LONGDESCRIP1	CLASSSTRUCT	CLASSTRUCTURE/LONGDESCRIPTION	LONGDESC

図 7-6 関係の名前

これでマイグレーション・グループを作成するための正しいオブジェクト構造が作成されました。

図 7-7 に新しいオブジェクト構造を示します。

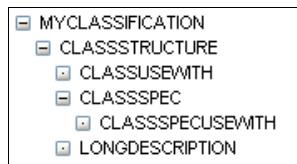


図 7-7 MYCLASSIFICATION オブジェクト構造

## 7.4 マイグレーション・グループ

分類データをマイグレーションするために、新しいオブジェクト構造を含む新規マイグレーション・グループを作成します。

1. 「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・グループ」にナビゲートします。
2. 新規マイグレーション・グループを作成します。
3. MYCLASSIFICATIONGROUPなどの名前を付けます。
4. MYCLASSIFICATION オブジェクト構造をマイグレーション・グループに追加します。

図 7-8 に示すように、マイグレーション・グループには 1 つだけマイグレーション・オブジェクトが含まれます。

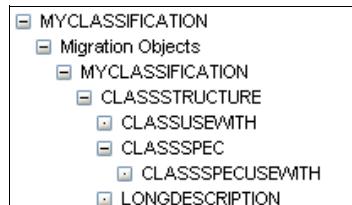


図 7-8 分類データのマイグレーション・グループ

このマイグレーション・グループは他のグループに依存しません。グループ順序を指定することなく、このグループを直接マイグレーションできます。

## 7.5 パッケージ定義

マイグレーション処理を開始するためには、マイグレーション・パッケージを定義する必要があります。単一のマイグレーション・パッケージを作成し、*MYCLASSIFICATIONGROUP* マイグレーション・グループをパッケージに追加します。残りのパッケージ定義については、以下にある「マイグレーション・マネージャー・ガイド」で説明されています。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_migration_mgr_guide.pdf).

パッケージを定義した後、マイグレーションする分類データを定義する SQL ステートメントを作成する必要があります。

次に、図 7-9 に示されたものと同じようなステートメントを作成します。

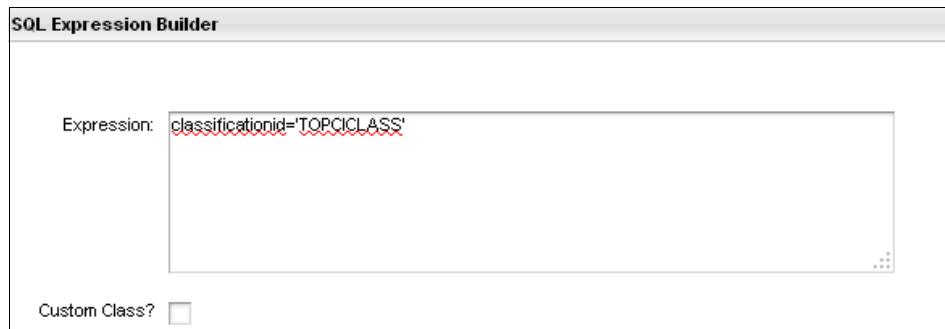


図 7-9 分類マイグレーション SQL ステートメント

**警告 :**すべての分類を正常にマイグレーションするには、SQL 文に最上位分類を指定する必要があります。マイグレーション・マネージャーは自動的に下位階層分類をマイグレーションします。

マイグレーション・マネージャーは SQL 文を検証します。SQL 文の検証に失敗した場合は、パッケージの保存前にエラー・メッセージが表示されます。

この方法では、分類データは結果的に最上位から最下位へとマイグレーションされます。マイグレーション・マネージャーはすべての分類を再帰的に取得します。図 7-10 に Where 文節ステートメント・ウィンドウを示します。

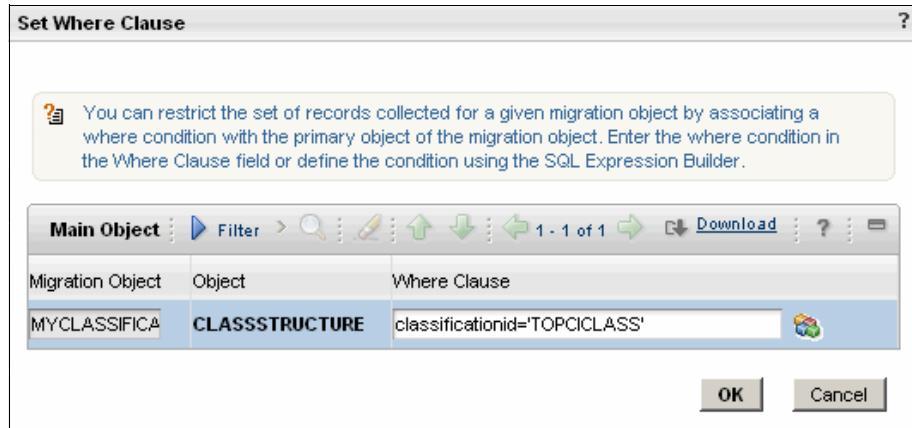


図 7-10 Where 文節ウィンドウ

これでパッケージを保存してアクティブ化し、パッケージを作成することができます。

## 7.6 デプロイメント

パッケージを作成したら、パッケージを配布してターゲット環境に置く必要があります。パッケージは、ファイル・タイプ配布またはデータベース・タイプ配布を介して配布できます。詳しくは、211 ページの『第 10 章 一般的なトピック』を参照してください。

データを正常にマイグレーションするために、オブジェクト構造 *myclassification* をターゲット環境へとマイグレーションします。

**注:** 分類をマイグレーションする際、管理モードをオンにしたり、サーバーを再始動したりする必要はありません。

ターゲット環境へのデプロイメント後、ユーザーはプロモート済みの分類を新しい環境で使い始めることができます。

## 7.7 デプロイメントに関する考慮事項

新規の分類を正常にターゲット環境にマイグレーションするには、決してターゲット環境で手動で分類を作成してはなりません。すべてのデータを最初にソース環境で作成してから、ターゲット環境にマイグレーションする必要があります。そうしないと、分類データのターゲット環境へのマイグレーション中に、インデックスの問題が起こる場合があります。

**注:** 分類のマイグレーションには常に、スナップショット・マイグレーションを使用する必要があります。マイグレーション・マネージャーは、変更パッケージを使用した分類のマイグレーションでは階層構造を処理することができません。

デプロイメント前に、オブジェクト構造で CLASSSTRUCTURE オブジェクトに HIERARCHYPATH 非永続フィールドが含まれていることを確認してください。含まれていない場合、「BMXAA5592E - インバウンド・データに無効な階層パスがあります」エラーが発生します。マイグレーション・マネージャーはこの属性を基に、挿入または更新のどちらを実行する必要があるかを判断します。

スナップショット・マイグレーションを使用する場合、マイグレーション・マネージャーは、分類マイグレーションで処理アクションが「置換」なのか「追加変更」なのかは関知しません。マイグレーション・マネージャーは、ターゲット環境で同一の HIERARCHYPATH を見つけた場合は、レコードを更新します。





# サービス・オファリング・コントンツのマイグレーション

本章では、IBM Tivoli Service Request Manager 製品の IBM サービス・オファリング・コンテンツをマイグレーションする際のプロセス・ステップ、問題、および考慮事項の概要を説明します。

サービス・オファリングは技術的にはコンテンツ型のデータであるため、このデータをある環境から別の環境に移動するには統合フレームワークを使用しなければなりません。ただし、IBM は、IBM 配布コンテンツのマイグレーションに特化したオブジェクト構造を持つツールに統合サービス・マネジメント (ISM) を組み込むことにより、ISM マーケットプレイスに対する専門知識を提供します。

本章では、ソースとターゲットの両方に IBM コンテンツが含まれている場合のシナリオを示します。ターゲット環境とソース環境に含まれる製品インストールのコンテンツ・パッケージが異なる場合の対処方法の詳細については、253 ページの『第 11 章 トラブルシューティング』を参照してください。

ここでは 2 つのシナリオのみを示しますが、このプロセスを利用して他のコンテンツ・パッケージに必要な操作を理解することができます。

本章には以下のセクションが含まれています。

- ▶ 177 ページの『8.1 サービス・オファリング・シナリオ』
- ▶ 177 ページの『8.2 要件』

- ▶ 177 ページの『8.3 ソリューション』
- ▶ 180 ページの『8.4 構成アプリケーション』
- ▶ 182 ページの『8.5 オブジェクト構造』
- ▶ 184 ページの『8.6 マイグレーション・グループ』
- ▶ 185 ページの『8.7 パッケージ定義』
- ▶ 188 ページの『8.8 デプロイメント』

## 8.1 サービス・オファリング・シナリオ

IBM が Tivoli Service Request Manager で提供するサービス・オファリング・コンテンツでは、データベース内の多数のオブジェクトを使用します。これらのオブジェクト構造をマイグレーションする場合は、どのサービス・オファリングの使用をマイグレーションするかに注意する必要があります。すぐに使用可能な(*OOTB*)マイグレーション・パッケージ・テンプレートを構成する各種のマイグレーション・グループには、さまざまな構成と、コンテンツ型のオブジェクト構造が多数含まれています。このマイグレーションを管理するために、ユーザーは適切なデータを入力し、パッケージ定義内の *Where* 文筋を操作することでそのデータを組み込む(または除外する)必要があります。

## 8.2 要件

Tivoli Service Request Manager 実装内の追加および変更されたサービス・オファリングをマイグレーションします。その際、提供または開発される各種のサービス・オファリングのテンプレートとして用意されているサンプル・マイグレーション・パッケージを使用します。

## 8.3 ソリューション

マイグレーション・パッケージ・アプリケーションにナビゲートして、テンプレート・マイグレーション・パッケージにアクセスします。

178 ページの図 8-1 に示すように、「リンク先 (Go To)」→「システムの構成 (System Configuration)」→「マイグレーション (Migration)」→「マイグレーション・マネージャー (Migration Manager)」を選択します。

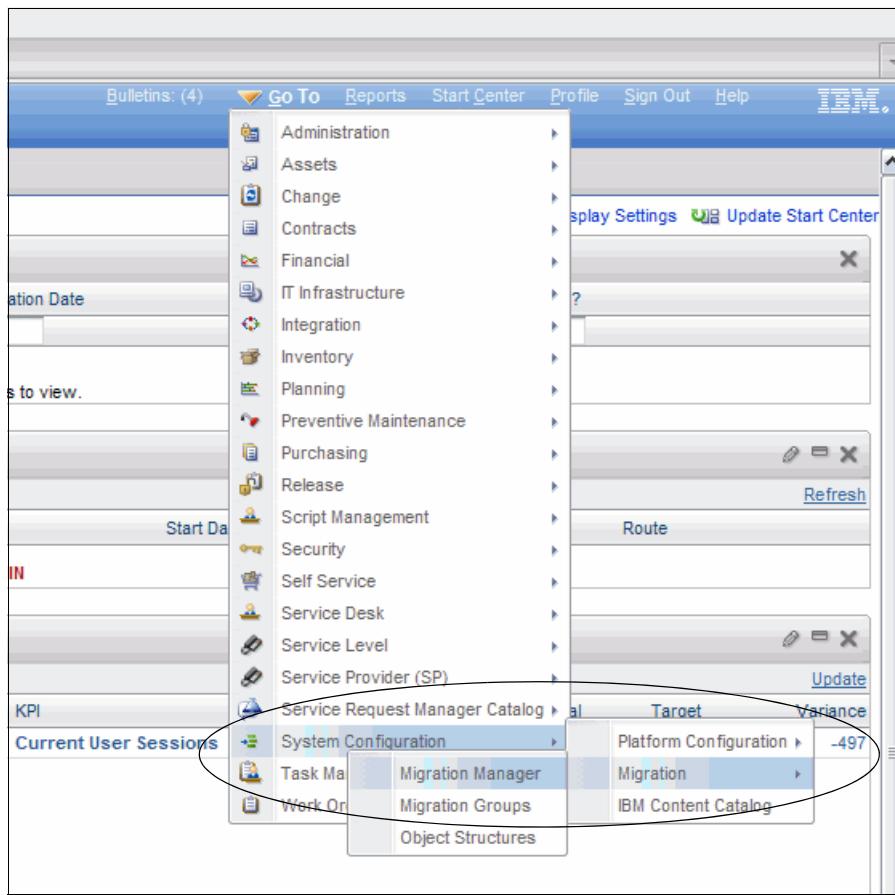


図 8-1 マイグレーション・マネージャー・アプリケーションへのメニュー・ナビゲーション

このシナリオには、PMSC72\_ServiceTemplate パッケージが必要です。このパッケージの名前は実際の環境によって異なる場合があるため、179 ページの図 8-2 に示すように検索することができます。

The screenshot shows the Migration Manager application window. At the top, there are tabs for 'List', 'Package Definition', 'Package Definition Structure', 'Distribution', 'Package', and 'Messages'. Below these are search and filter tools, including 'Advanced Search' and 'Save Query'. The main area displays a table titled 'Package Definitions' with two rows:

Package Definition Name	Source	Description
pmSC		
<a href="#">PMSC72_ServiceTemplate</a>	PMSC72_SampleTemplate	Template for SRM 72 Service Offering migration
<a href="#">PMSC72_CatalogTemplate</a>	PMSC72_SampleTemplate	Template for SRM 72 Catalog migration

A checkbox labeled 'Select Records' is located at the bottom left of the table.

図 8-2 サービス・オファリング・テンプレート

「マイグレーション・グループ (Migration Groups)」構造を調べると、図 8-3 に示すように、このシナリオには非常に多くの考慮事項があることがわかります。

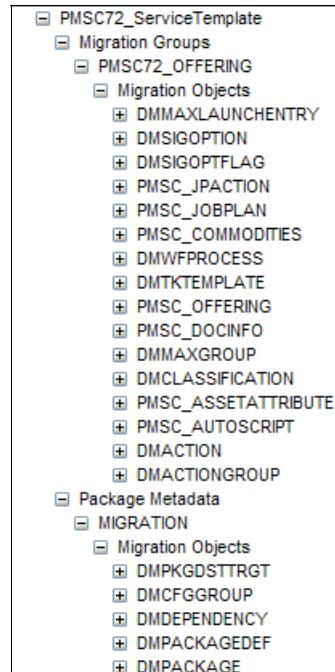


図 8-3 サービス・オファリング・テンプレートのマイグレーション・グループ (Migration Group)

## 8.4 構成アプリケーション

マイグレーション・グループを理解するために、サービス・オファリングで構成されている各種のアプリケーションを表 8-1 に示します。

表 8-1 構成アプリケーション

アプリケーション名	オブジェクト構造
コンテキストでの起動	<ul style="list-style-type: none"><li>□ DMMAXLAUNCHENTRY<ul style="list-style-type: none"><li>□ MAXLAUNCHENTRY</li><li>□ LONGDESCRIPTION</li><li>□ MAXLECONTEXT</li></ul></li></ul>
作業標準	<ul style="list-style-type: none"><li>□ PMSC_JOBPLAN<ul style="list-style-type: none"><li>□ JOBLAN</li><li>□ JOBTASK<ul style="list-style-type: none"><li>□ JOBTASKSPEC</li><li>□ LONGDESCRIPTION</li></ul></li><li>□ JOBITEM</li><li>□ JOBPLANCLASS</li><li>□ JOBPLANSPEC</li><li>□ JOBSERVICE</li><li>□ JOBLABOR</li><li>□ JOBMATERIAL</li><li>□ JOBTOOL</li><li>□ JPASSETSLINK</li><li>□ JPTASKRELATION</li></ul></li></ul>
ドキュメントの表示	<ul style="list-style-type: none"><li>□ PMSC_DOCINFO<ul style="list-style-type: none"><li>□ DOCINFO</li><li>□ DOCLINKS</li></ul></li></ul>
オファリング	<ul style="list-style-type: none"><li>□ PMSC_OFFERING<ul style="list-style-type: none"><li>□ PMSCOFFERING</li><li>□ PMSCOFFERINGAUTH</li><li>□ PMSCITEMSPEC</li><li>□ PMSCOFFDIALOG</li><li>□ IMGLIB</li><li>□ LONGDESCRIPTION</li></ul></li></ul>
アクション  注：このアプリケーションには2つのオブジェクト構造があります。	<ul style="list-style-type: none"><li>□ PMSC_JPACTION<ul style="list-style-type: none"><li>□ ACTION<ul style="list-style-type: none"><li>□ LONGDESCRIPTION</li></ul></li></ul></li><li>□ DMACTION<ul style="list-style-type: none"><li>□ ACTION<ul style="list-style-type: none"><li>□ LONGDESCRIPTION</li></ul></li><li>□ DMACTIONGROUP</li><li>□ ACTIONGROUP</li></ul></li></ul>

アプリケーション名	オブジェクト構造
分類 分類 (SP)	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> DMCLASSIFICATION</li> <li><input type="checkbox"/> CLASSSTRUCTURE</li> <li><input type="checkbox"/> CLASSUSEWITH</li> <li><input checked="" type="checkbox"/> CLASSSPEC</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> </ul>
ワークフロー・デザイナー	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> DMWFPROCESS</li> <li><input type="checkbox"/> WFWORKFLOW</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> <li><input checked="" type="checkbox"/> WFNODE</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> <li><input type="checkbox"/> WFTASK</li> <li><input checked="" type="checkbox"/> WFASSIGNMENT</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> <li><input type="checkbox"/> WFSTART</li> <li><input type="checkbox"/> WFSTOP</li> <li><input type="checkbox"/> WFINPUT</li> <li><input type="checkbox"/> WFACTION</li> <li><input type="checkbox"/> WFWAITLIST</li> <li><input checked="" type="checkbox"/> WFINTERACTION</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> <li><input type="checkbox"/> WFCCONDITION</li> <li><input type="checkbox"/> WFSUBPROCESS</li> <li><input type="checkbox"/> WFNOTIFICATION</li> </ul>
チケット・テンプレート	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> DMTKTEMPLATE</li> <li><input type="checkbox"/> TKTEMPLATE</li> <li><input type="checkbox"/> TKTEMPLATESPEC</li> <li><input checked="" type="checkbox"/> TKTEMPLTACTIVITY</li> <li><input type="checkbox"/> TKTEMPLTACTYSPEC</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> </ul>
サービス・グループ	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> PMSC_OFFERING</li> <li><input type="checkbox"/> PMSCOFFERING</li> <li><input type="checkbox"/> PMSCOFFERINGAUTH</li> <li><input type="checkbox"/> PMSCITEMSPEC</li> <li><input type="checkbox"/> PMSCOFFDIALOG</li> <li><input type="checkbox"/> IMGLIB</li> <li><input type="checkbox"/> LONGDESCRIPTION</li> </ul>
自動化スクリプト	<ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> PMSC_AUTOSCRIPT</li> <li><input type="checkbox"/> AUTOSCRIPT</li> </ul>

これらのアプリケーションには、次のナビゲーション・リンクを使用してアクセスできます。

**コンテキストでの起動：**「リンク先」→「システムの構成」→「プラットフォームの構成」→「コンテキストでの起動」を選択します。

**作業標準**: 「リンク先」→「計画」→「作業標準」を選択します。

**ドキュメントの表示**: 「リンク先」→「管理」→「ドキュメントの表示」を選択します。

**オファリング**: 「リンク先」→「Service Request Manager カタログ (Service Request Manager Catalog)」→「オファリング」を選択します。

**アクション**: 「リンク先」→「システムの構成」→「プラットフォームの構成」→「アクション」を選択します。

**分類 (または分類 (SP))**: 「リンク先」→「管理」→「分類」を選択します。

**ワークフロー・デザイナー**: 「リンク先」→「システムの構成」→「プラットフォームの構成」→「ワークフロー・デザイナー」を選択します。

**チケット・テンプレート**: 「リンク先」→「サービス・デスク」→「チケット・テンプレート」を選択します。

**サービス・グループ**: 「リンク先」→「Service Request Manager カタログ (Service Request Manager Catalog)」→「サービス・グループ」を選択します。

**自動化スクリプト**: 「リンク先」→「スクリプト管理」→「自動化スクリプト」を選択します。

## 8.5 オブジェクト構造

このセクションでは、Tivoli's process automation engine (Tivoli/Maximo Base Service) では配布されない、Tivoli Service Request Manager 固有のオブジェクト構造のみを取り上げます。これらのオブジェクト構造は、サービス・オファリング・コンテンツのマイグレーションに固有のものです。これらの構造を以下に示します。

- ▶ PMSC\_JOBPLAN
- ▶ PMSC\_OFFERING
- ▶ PMSC\_DOCINFO

### 8.5.1 PMSC\_JOBPLAN

このオブジェクト構造には、注目すべき機能がいくつかあります。第一に、サービス・オファリング・コンテンツに固有の、インバウンド処理およびアウ

トバウンド処理用のクラスがあります。これらは他のオブジェクト構造または目的には使用されません。図 8-4 は、指定されたクラスを示しています。

Outbound Definition Class:	com.ibm.ism.pmsc.migration.PMSC_JOBPLAN_
Inbound Processing Class:	com.ibm.ism.pmsc.migration.PMSC_JOBPLAN_

図 8-4 「作業標準(Job Plan)」マイグレーションのためのカスタム Tivoli Service Request Manager クラス

さらに IBM は、サービス・オファリングを正しくマイグレーションできるようにするために、Tivoli Service Request Manager 製品オファリング用の新しい関係を提供しています。この関係を図 8-5 に示します。

Object	Parent Object	Object Location Path	Relationship
JOBPLAN	🔍	» JOBPLAN	PMSC_NESTEDJOBPLAN

図 8-5 PMSC ネストされた作業標準関係

データベースの構成アプリケーションを使用して、定義を個別に確認することをお勧めします。

### 8.5.2 PMSC\_OFFERING

*PMSC\_Offering* オブジェクト構造では、標準製品とは別の追加のクラスと関係が使用されることを理解しておく必要があります。図 8-6 を参照してください。

Outbound Definition Class:	com.ibm.ism.pmsc.migration.PMSC_OffAndCat_
Inbound Processing Class:	com.ibm.ism.pmsc.migration.PMSC_OffAndCat_

図 8-6 PMSC\_OFFERING の専用クラス

このオブジェクト構造の関係の一覧を 183 ページの図 8-7 に示します。

Object	Parent Object	Object Location Path	Relationship
PMSCOFFERING	🔍	» PMSCOFFERING	
PMSCOFFERING	PMSCOFFERING	» PMSCOFFERING/PMSCOFFERINGAUTH	PMSCOFFERINGAUTH
PMSCITEMSPEC	PMSCOFFERING	» PMSCOFFERING/PMSCITEMSPEC	PMSCITEMSPEC
PMSCOFFDIALOG	PMSCOFFERING	» PMSCOFFERING/PMSCOFFDIALOG	PMSCOFFDIALOG
IMGLIB	🔍	» PMSCOFFERING/IMGLIB	IMGLIB

図 8-7 PMSCOFFERING オブジェクト構造の関係

### 8.5.3 PMSC\_DOCINFO

Tivoli Service Request Manager のサービス・オファリング・コンテンツでは、PMSC\_DOCINFO オブジェクト構造用に図 8-8 に示すようなクラスが提供されています。

Outbound Definition Class:	com.ibm.ism.pmsc.migration.PMSC_DocInfo_Ou
Inbound Processing Class:	com.ibm.ism.pmsc.migration.PMSC_DocInfo_Inf

図 8-8 PMSC\_DOCINFO の専用クラス

これらのクラスにより、構成データのマイグレーション時にコンテンツ型データが正しく処理されます。

**重要:** このオブジェクト構造は、PMSCOFFERING オブジェクトと密結合されています。これを PMSCOFFERING によって所有されていない docinfo オブジェクトに使用しようとすると、作成フェーズで失敗します。提供されているすぐに使用可能な Where 文節は以下のとおりです。

```
docinfooid in (select docinfooid from doclinks where
ownertable='PMSCOFFERING' and ownerid in (select itemid from
pmescoffering where itemnum='SERVICENAME'))
```

## 8.6 マイグレーション・グループ

サービス・オファリング用のマイグレーション・グループは PMSC\_OFFERING マイグレーション・グループです。これは、必要な場合にのみ複製および変更できるように提供されています。オブジェクト構造の順序は、185 ページの図 8-9 のようになっています。

Object	Description	Migration Object Order
AUNC >>	Launch entry migration object for Migration Manager	5
TION >>	Signature option migration object for Migration Manager	10
TFLAI >>	Advanced signature option migration object for Migration Manager	15
ITOSC >>	PMSC Automation Scripts	20
SSETA >>	Asset attribute migration object	25
SIFICA >>	Classifications migration object	30
ACTIC >>	Action migration object for Migration Manager	35
BPLA >>	Jobplan migration object for Migration Manager	40
MMMOI >>	Commodities migration object	45
N >>	Action migration object for Migration Manager	50
NGRC >>	Action Group migration object for Migration Manager	55
OCES! >>	Workflow process definition migration object for Migration Manager	60
IPLAT >>	Ticket Template migration object	65
ROUP >>	Security group migration object	70
FERIN >>	SRM 72 Service Offering migration object	75

図 8-9 サービス・オファリング・マイグレーション・グループのオブジェクト順序

この順序は重要であるほか、順序の番号付けに空きがあることに注意してください。これにより、ユーザーは必要なオブジェクト構造を簡単に追加できます。

## 8.7 パッケージ定義

このセクションでは、ユーザーが実行する大部分の作業について説明します。IBM 提供の SQL 定義では、クライアントが変更または追加したオファリングをマイグレーションするために、定義を変更する必要があります。

### パッケージへのアクセス

マイグレーション・マネージャー・アプリケーションにナビゲートし(「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」)、*PMSC72\_ServiceTemplate* を検索します。見つかったら、「パッケージ定義」タブをクリックし、186 ページの図 8-10 に示すように矢印アイコンをクリックします。

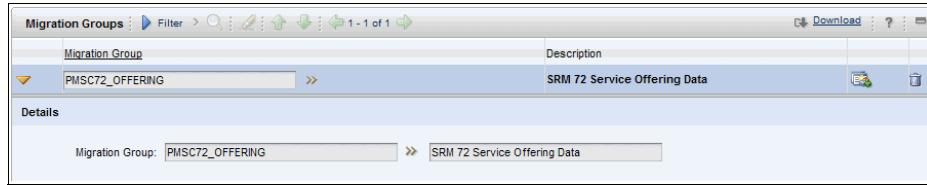


図 8-10 マイグレーション・パッケージ定義

さまざまなパッケージの SQL 基準ステートメントの 1 つを、図 8-11 に示します。

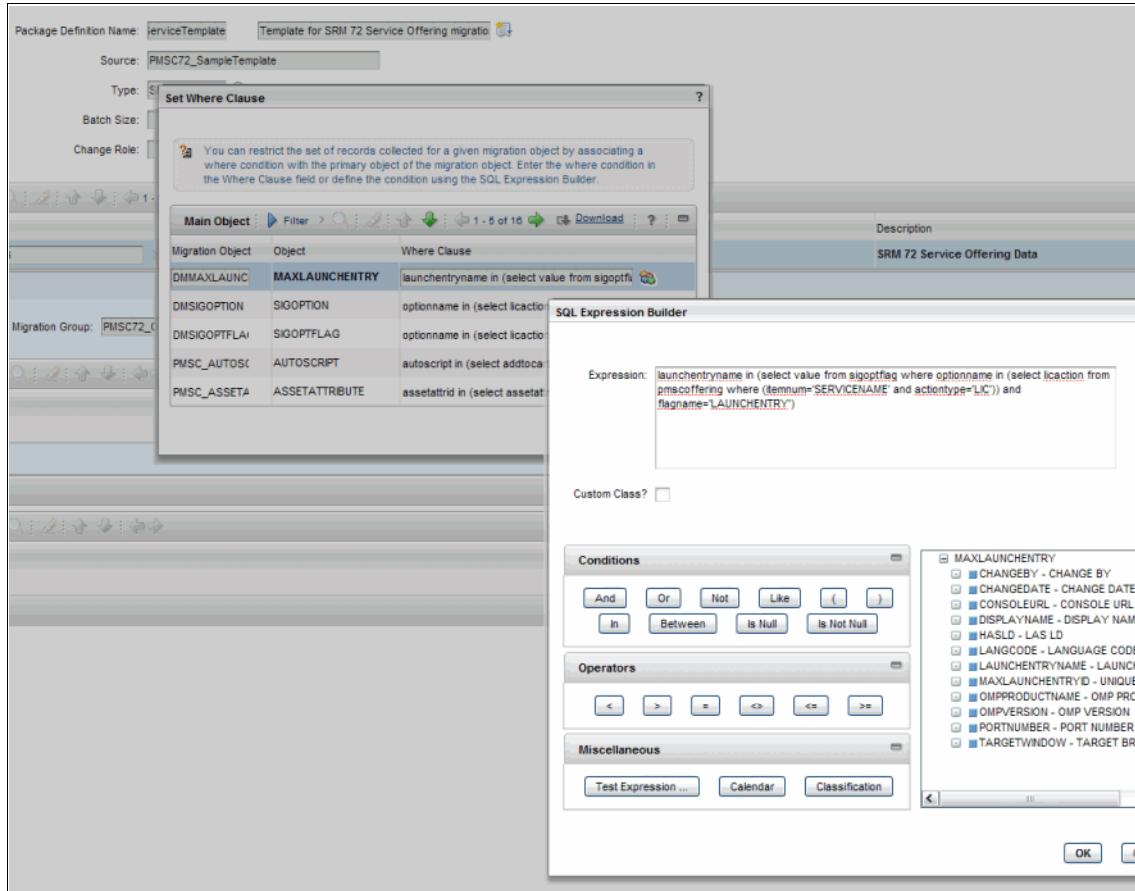


図 8-11 MAXLAUNCHENTRY が選択された状態の「Where 文節の設定 (Set Where clause)」ダイアログ・ボックス

## 8.7.1 SQL 定義の定義

各行の右端にある式ビルダーのアイコンをクリックして、式ビルダーにアクセスします。ここで、グループ内のマイグレーション・オブジェクトごとにSQLを操作します。

### 例 8-1 キャプションの例

```
launchentryname in (select value from sigoptflag where optionname in  
(select licaction from pmscoffering where (itemnum='SERVICENAME' and  
actiontype='LIC')) and flagname='LAUNCHENTRY')
```

この例の ‘SERVICENAME’ を、個別の ITEMNUM 値か、レコード選択基準を満たす他の構成体に置き換えます。

次の例の SQL 構文を使用すると、マイグレーションするオファリングを複数選択できます。

### 例 8-2 キャプションの例

```
launchentryname in (select value from sigoptflag where optionname in  
(select licaction from pmscoffering where (itemnum in  
('SERVICENAME1','SERVICENAME2','SERVICENAME3','SERVICENAME4') and  
actiontype='LIC')) and flagname='LAUNCHENTRY')
```

この方法では、4つのサービス・オファリングをマイグレーションします。マイグレーションが正しく行われるためには、この基準が他のオブジェクト構造の基準と一致していなければなりません。

この特定のマイグレーション・オブジェクトのすべてのオファリングをマイグレーションする方法を次の例に示します。

### 例 8-3 キャプションの例

```
launchentryname in (select value from sigoptflag where optionname in  
(select licaction from pmscoffering where (actiontype='LIC')) and  
flagname='LAUNCHENTRY')
```

**注：**正しいレコード・セットがマイグレーションされるようにするには、オブジェクト構造間で選択基準が一致していなければならないという点に注意してください。レコード・セットの作成に使用されるSQLの構文および各種の成果物は異なる場合がありますが、基準は同じである必要があります。

## 8.8 デプロイメント

パッケージを作成したら、パッケージを配布してターゲット環境に送信します。これにはファイル配布を使用する必要があります。この手順は、「マイグレーション・マネージャー・ガイド」で説明されています。

**注:** サービス・オファリングのマイグレーションでは、**管理モード**および**サーバー再始動**は必要ありません。

ターゲット環境へのデプロイメント後、ユーザーはターゲット環境でサービス・オファリングの使用を開始できます。

### 8.8.1 デプロイメントの考慮事項

この特定のマイグレーション・パッケージの主な考慮事項として、以下のものが存在することを確認する必要があります。

1. マイグレーション・マネージャーがコンテンツを正しくマイグレーションするため必要なマイグレーション・グループ成果物の変更
2. ある環境から別の環境への移動が必要なコンテンツの変更

コンテンツはシステムの初回セットアップ時に IBM 製品インストーラーによってインストールされるため、これらの条件のどちらにも当てはまらない場合はコンテンツをマイグレーションする必要はほとんどありません。

これはコンテンツ・ベースの資料であるため、ユーザーおよび実装チームは、実稼働環境に変更を取り込むために単体テストおよび統合テストでコンテンツが必要かどうかを慎重に検討する必要があります。

ビジネス・プロセスが正しく機能するためにコンテンツが必要な場合、本章が重要となります。



# サービス・カタログのマイグレーション

この章では、サービス・カタログのマイグレーションに関連するベスト・プラクティスを説明します。

サービス・カタログは、Tivoli Service Request Manager のコア・コンポーネントを形成する多数の標準データベース・オブジェクトから構成されています。サービス・カタログは、論理的にグループ化されたサービス・オファリングで構成されます。サービス・カタログの典型的な例は、サーバー・ハードウェアおよび電子メール・アカウント作成サービス・オファリングなどが含まれる IT サービス・カタログです。

Tivoli Service Request Manager ソリューションは、すぐに使用可能な標準パッケージ定義テンプレートと共に提供されており、さらにサービス・カタログ・マイグレーションにデータを組み込んだり除外したりするための構成可能な Where 文節も備えています。

この章には以下のセクションがあります。

- ▶ 191 ページの『9.1 要件』
- ▶ 191 ページの『9.2 ソリューション』
- ▶ 193 ページの『9.3 構成アプリケーション』
- ▶ 194 ページの『9.4 オブジェクト構造』
- ▶ 201 ページの『9.5 マイグレーション・グループ』

- ▶ 202 ページの『9.6 パッケージ定義』
- ▶ 206 ページの『9.7 デプロイメント』
- ▶ 207 ページの『9.7.1 デプロイメントに関する考慮事項』

## 9.1 要件

サービス・カタログは開発環境で定義し、その後テストおよび実稼働環境にプロモートします。Tivoli Service Request Manager には、サービス・カタログのマイグレーションに必要な 2 つのマイグレーション・グループが標準として提供されています。175 ページの『第 8 章 サービス・オファリング・コンテンツのマイグレーション』では、サービス・オファリングのマイグレーションについて検討しました。サービス・カタログはサービス・オファリングで構成されます。したがって、PMSC72\_CATALOG マイグレーション・グループに加えて、同じ PMSC72\_OFFERING マイグレーション・グループを活用できます。

事前定義パッケージは以下の概念に関連したデータをマイグレーションします。

- ▶ コンテキストでの起動
- ▶ 署名オプション
- ▶ アクション(作業標準)
- ▶ 作業標準
- ▶ 商品
- ▶ ワークフロー・プロセス
- ▶ チケット・テンプレート
- ▶ オファリング
- ▶ ドキュメント情報
- ▶ セキュリティー・グループ
- ▶ 分類
- ▶ 資産属性
- ▶ 自動化スクリプト
- ▶ アクション
- ▶ アクション・グループ
- ▶ セキュリティー・グループ
- ▶ サービス・カタログ

## 9.2 ソリューション

PMSC72\_CatalogTemplate パッケージ定義は、オブジェクト構造またはマイグレーション・オブジェクトの変更を必要としません。オブジェクト構造に関連付けられた、すぐに使用可能な Where 文節を構成して、マイグレーションするサービス・カタログ (CATALOGNAME) を指定する必要があります。

テンプレート・マイグレーション・パッケージにアクセスするには、以下の方法でマイグレーション・パッケージ・アプリケーションにナビゲートします。

「リンク先 (Go To)」→「システムの構成 (System Configuration)」→「マイグレーション (Migration)」→「マイグレーション・パッケージ (Migration Package)」(図 9-1 を参照 )

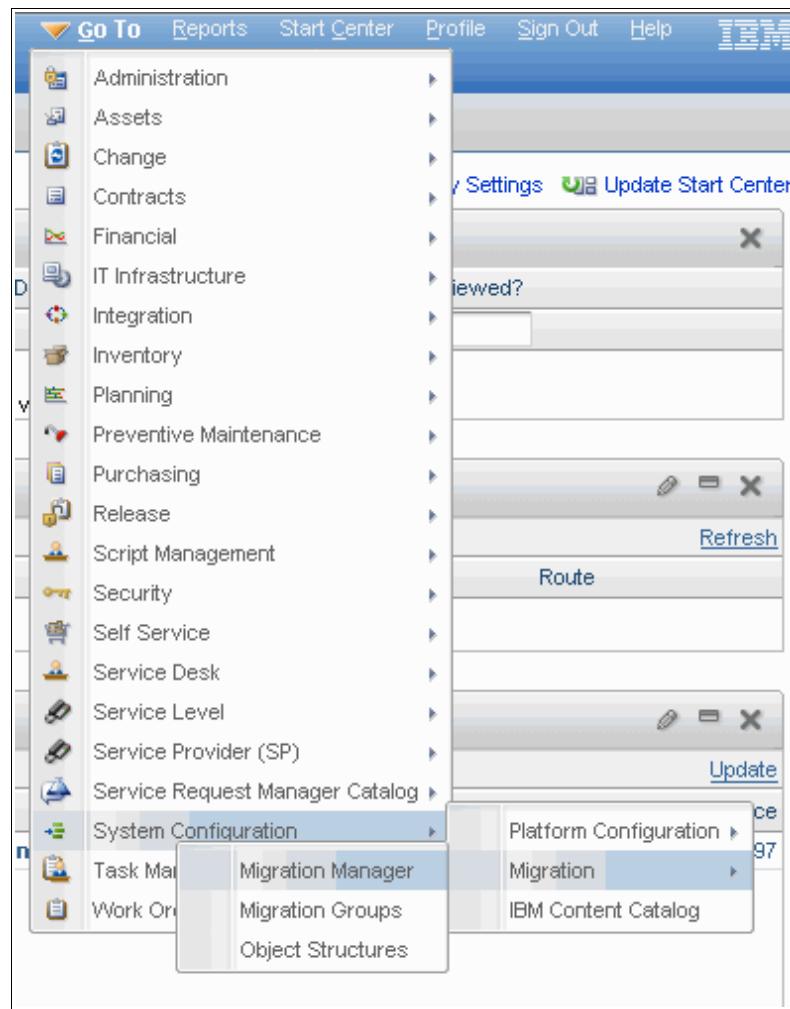


図 9-1 マイグレーション・パッケージ・アプリケーションへのメニュー・ナビゲーション

193 ページの図 9-2 に示すように、すぐに使用可能なマイグレーション・パッケージ・テンプレートを使用してサービス・カタログをマイグレーションします。

Package Definition Name	Source	Description
pmsc		
PMSC72_ServiceTemplate	PMSC72_SampleTemplate	Template for SRM 72 Service Offering migration
<b>PMSC72_CatalogTemplate</b> PMSC72_SampleTemplate      Template for SRM 72 Catalog migration		
<input type="checkbox"/> Select Records		

図9-2 図2 サービス・カタログ・パッケージ定義

図 9-3 に示すとおり、PMSC72\_CatalogTemplate は PMSC72\_CATALOG マイグレーション・グループおよび PMSC72\_OFFERING マイグレーション・グループで構成されます。

Migration Group	Description
PMSC72_CATALOG	SRM 72 Catalog
PMSC72_OFFERING	SRM 72 Service Offering

図9-3 図3 - PMSC72\_CatalogTemplate マイグレーション・グループ

## 9.3 構成アプリケーション

この章の前のシナリオのサービス・オファリング・マイグレーションでは、個々のサービス・オファリング (PMSC72\_OFFERING マイグレーション・オブジェクト) に関連付けられた構成アプリケーションを検討しました。同様に、サービス・カタログでは、サービス・オファリングを定義するために使用した構成アプリケーションに加えて、セキュリティー・グループ・アプリケーションおよびカタログ・アプリケーションを活用します。194 ページの表 9-1 に、追加の構成アプリケーションおよび関連するオブジェクト構造を示します。

表 9-1 構成アプリケーション

アプリケーション名	オブジェクト構造
セキュリティ・グループ	<ul style="list-style-type: none"><li>□ DMMAXGROUP</li><li>□ MAXGROUP<ul style="list-style-type: none"><li>□ LONGDESCRIPTION</li><li>□ SITEAUTH</li><li>□ APPLICATIONAUTH</li><li>□ GLAUTH</li><li>□ SECURITYRESTRICT</li></ul></li></ul>
カタログ	<ul style="list-style-type: none"><li>□ PMSC72_CATALOG</li><li>□ PMSCCATALOG<ul style="list-style-type: none"><li>□ PMSCCATALOGAUTH</li><li>□ PMSCCATALOGOFFMAP</li><li>□ IMGLIB</li></ul></li></ul>

### 9.3.1 セキュリティ・グループ

セキュリティ・グループ・アプリケーションは、該当する製品のスタート・センターで「リンク先」メニューを以下のようにナビゲートしてアクセスします。

「セキュリティ」 → 「セキュリティ・グループ」

### 9.3.2 カタログ

カタログ・アプリケーションは、該当する製品のスタート・センターで「リンク先」メニューを以下のようにナビゲートしてアクセスします。

「Service Request Manager カタログ (Service Request Manager Catalog)」「カタログ」

## 9.4 オブジェクト構造

以下のオブジェクト構造は、すぐに使用可能な PMSC72\_CatalogTemplate パッケージ定義と共に提供されます。

サービス・カタログのマイグレーションをサポートする PMSC72\_CATALOG マイグレーション・グループの一部である DMMAXGROUP オブジェクト構造。195 ページの図 9-4 に DMMAXGROUP オブジェクト構造のコンテンツを示します。

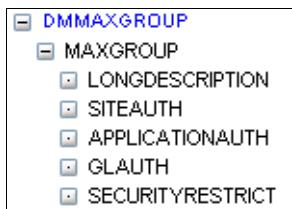


図 9-4 DMMAXGROUP オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_CATALOG マイグレーション・グループの一部である PMSCATALOG オブジェクト構造。図 9-5 に PMSCATALOG オブジェクト構造のコンテンツを示します。

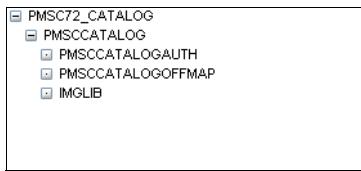


図 9-5 PMSCCATALOG オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMMAXLAUNCHENTRY オブジェクト構造。図 9-6 に DMMAXLAUNCHENTRY オブジェクト構造のコンテンツを示します。

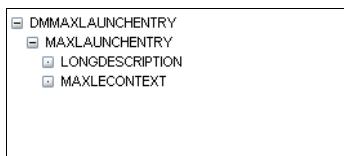


図 9-6 DMMAXLAUNCHENTRY オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMSIGOPTION オブジェクト構造。図 9-7 に DMSIGOPTION オブジェクト構造のコンテンツを示します。

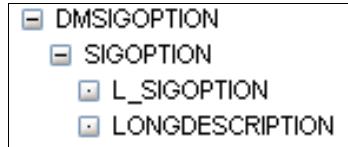


図 9-7 DMSIGOPTION オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMSIGOPTFLAG オブジェクト構造。図 9-8 に DMSIGOPTFLAG オブジェクト構造のコンテンツを示します。



図 9-8 DMSIGOPTFLAG オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_JPACTION オブジェクト構造。図 9-9 に PMSC\_JPACTION オブジェクト構造のコンテンツを示します。

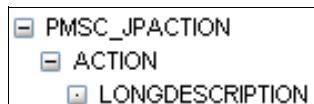


図 9-9 PMSC\_JPACTION オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_JOBPLAN オブジェクト構造。図 9-10 に PMSC\_JOBPLAN オブジェクト構造のコンテンツを示します。

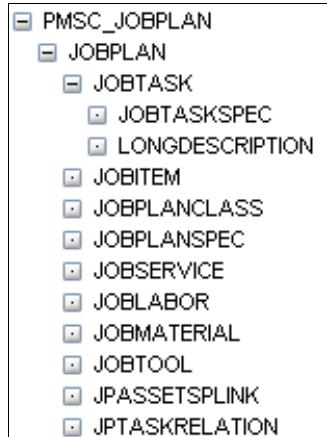


図 9-10 PMSC\_JOBPLAN オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_COMMODITIES オブジェクト構造。図 9-11 に PMSC\_COMMODITIES オブジェクト構造のコンテンツを示します。

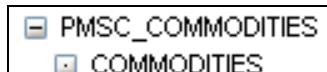


図 9-11 PMSC\_COMMODITIES オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMWFPROCESS オブジェクト構造。198 ページの図 9-12 に DMWFPROCESS オブジェクト構造のコンテンツを示します。

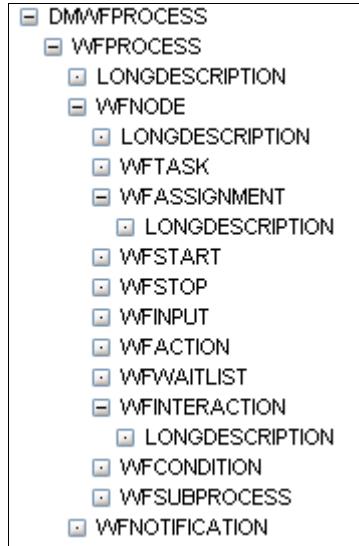


図 9-12 DMWFPROCESS オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMTKTEMPLATE オブジェクト構造。図 9-13 に DMTKTEMPLATE オブジェクト構造のコンテンツを示します。

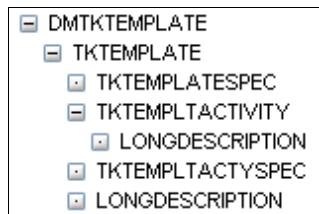


図 9-13 DMTKTEMPLATE オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_OFFERING オブジェクト構造。199 ページの図 9-14 に PMSC\_OFFERING オブジェクト構造のコンテンツを示します。

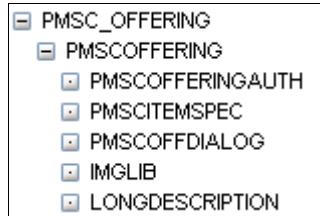


図 9-14 PMSC\_OFFERING オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_DOCINFO オブジェクト構造。図 9-15 に PMSC\_DOCINFO オブジェクト構造のコンテンツを示します。

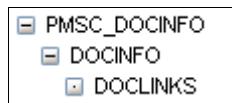


図 9-15 PMSC\_DOCINFO オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMMAXGROUP オブジェクト構造。図 9-16 に DMMAXGROUP オブジェクト構造のコンテンツを示します。

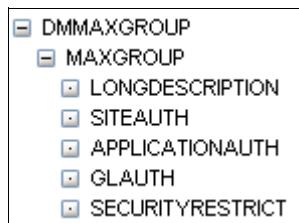


図 9-16 DMMAXGROUP オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMCLASSIFICATION オブジェクト構造。図 9-17 に DMCLASSIFICATION オブジェクト構造のコンテンツを示します。

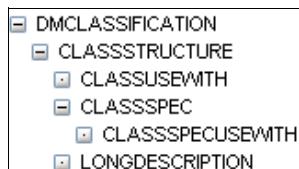


図 9-17 DMCLASSIFICATION オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_ASSETATTRIBUTE オブジェクト構造。図 9-18 に PMSC\_ASSETATTRIBUTE オブジェクト構造のコンテンツを示します。

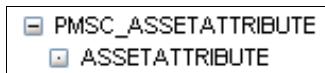


図 9-18 PMSC\_ASSETATTRIBUTE オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である PMSC\_AUTOSCRIPT オブジェクト構造。図 9-19 に PMSC\_AUTOSCRIPT オブジェクト構造のコンテンツを示します。



図 9-19 PMSC\_AUTOSCRIPT オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMACTION オブジェクト構造。

図 9-20 に DMACTION オブジェクト構造のコンテンツを示します。

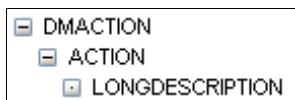


図 9-20 DMACTION オブジェクト構造

サービス・カタログのマイグレーションをサポートする PMSC72\_OFFERING マイグレーション・グループの一部である DMACTIONGROUP オブジェクト構造。図 9-21 に DMACTIONGROUP オブジェクト構造のコンテンツを示します。

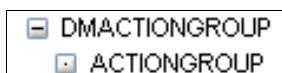


図 9-21 DMACTIONGROUP オブジェクト構造

ユーザーは、サービス・カタログをマイグレーションするために、追加のオブジェクト構造を作成したり、既存のオブジェクト構造を変更したりする必要はありません。

前のシナリオで、PMSC\_OFFERING オブジェクト構造の特別な機能を検討しました。これらの機能はサービス・カタログのマイグレーション・シナリオにも適用できます。

## 9.5 マイグレーション・グループ

このセクションでは、指定したサービス・カタログをマイグレーションする、すぐに使用可能なマイグレーション・グループの概略を説明します。このマイグレーション・シナリオを実行する際に、以下にリストされたマイグレーション・グループに変更を加える必要は一切ありません。

サービス・カタログのマイグレーションをサポートするオブジェクト構造は、次の 2 つのマイグレーション・グループに属します。

- ▶ PMSC72\_OFFERING
- ▶ PMSC72\_CATALOG

図 9-22 に OOTB PMSC72\_OFFERING および PMSC72\_CATALOG マイグレーション・グループとその依存関係を示します。

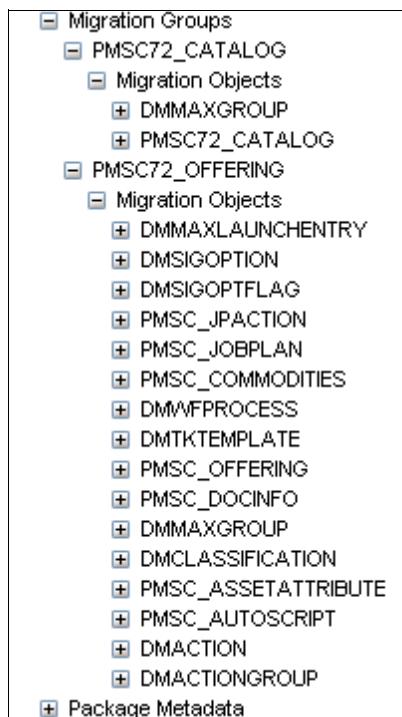


図 9-22 PMSC72\_OFFERING および PMSC72\_CATALOG マイグレーション・グループ構造

## 9.6 パッケージ定義

このセクションでは、読者が実行する大部分の作業について説明します。指定したサービス・カタログをマイグレーションするには、IBM から提供される Where 文節の値を変更する必要があります。

### パッケージへのアクセス

マイグレーション・マネージャー・アプリケーションにナビゲートし(「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」)、次に **PMSC72\_CatalogTemplate** を検索して選択します。

PMSC72\_CatalogTemplate は、2 つのすぐに使用可能なマイグレーション・オブジェクト (PMSC72\_OFFERING および PMSC72\_CATALOG) から構成されています。

### 9.6.1 PMSC72\_OFFERING

PMSC72\_OFFERING マイグレーション・オブジェクトは、マイグレーション・オブジェクト内の各オブジェクト構造に対する標準 Where 文節と共に提供されています。ユーザーは、これらの Where 文節を編集し、マイグレーションするカタログの名前を指定する必要があります。このためには、カタログ名を(データベースに表示されるとおりに) 各 Where 文節の「CATALOGUENAME」が現れる部分に挿入します。例えば、SERVICE CATALOG1 サービス・カタログをマイグレーションする場合、Where 文節 A を Where 文節 B で置き換えます。

#### Where 文節 A

```
launchentryname in (select value from sigoptflag where optionname in (select licaction from pmscoffering where actiontype='LIC' and itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) and flagname='LAUNCHENTRY')
```

#### Where 文節 B

```
launchentryname in (select value from sigoptflag where optionname in (select licaction from pmscoffering where actiontype='LIC' and itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='SERVICE CATALOG1')) and flagname='LAUNCHENTRY')
```

203 ページの表 9-2 に PMSC72\_OFFERING マイグレーション・オブジェクトを構成するオブジェクト構造と、それらに関連付けられている Where 文節をマイグレーションされる順番にリストします。

表9-2 すぐに使用可能なPMSC72\_OFFERING のWhere文節

マイグレーション順序	オブジェクト構造	Where文節
	DMMAXLAUNCH	launchentryname in (select value from sigoptflag where optionname in (select licaction from pmscoffering where actiontype='LIC' and itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) and flagname='LAUNCHENTRY')
10	DMSIGOPTION	optionname in (select licaction from pmscoffering where actiontype='LIC' and itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))
15	DMSIGOPTIONFLAG	optionname in (select licaction from pmscoffering where actiontype='LIC' and itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) and flagname='LAUNCHENTRY'
20	PMSC_AUTOSCRIPT	autoscript in (select addtocartscript from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) or autoscript in (select offsubmitscript from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) or autoscript in (select prepopulationscript from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) or autoscript in (select validationattr from pmscoffdialog where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))
25	PMSC_ASSETATTRIBUTE	assetattrid in (select assetattrid from pmscitemspec where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))

マイグレーション順序	オブジェクト構造	Where 文節
30	DMCLASSIFICATION	classstructureid in (select classstructureid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) or classstructureid in (select classstructureid from pmsscatalogoffmap where itemnum='CATALOGNAME') or classstructureid in (select classstructureid from tktemplate where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))) or classstructureid in (select classstructureid from tktempltactivity where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))) or classstructureid in (select classstructureid from jobplan where jnum in (select jnum from tktempltactivity where templateid in (select templateid from tktemplate where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))))) or classstructureid in (select classstructureid from jobtask where jnum in (select jnum from tktempltactivity where templateid in (select templateid from tktemplate where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))))))
35	PMSC_JPACTION	action in (select flowaction from jobplan where jnum in (select jnum from tktempltactivity where templateid in (select templateid from tktemplate where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))))) or action in (select flowaction from jobtask where jnum in (select jnum from tktempltactivity where templateid in (select templateid from tktemplate where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))))))
40	PMSC_JOBPLAN	jnum in (select jnum from tktempltactivity where templateid in (select templateid from tktemplate where templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))))

マイグレーション順序	オブジェクト構造	Where 文節
45	PMSC_COMMODITIES	commodity in (select commodity from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) or commodity in (select commoditygroup from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))
50	DMACTION	action in (select action from wfaction where processname in ((select mgrapprprocess from pmscofferingext where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) union (select srapprprocess from pmscofferingext where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))) or action in (select member from actiongroup where action in (select action from wfaction where processname in ((select mgrapprprocess from pmscofferingext where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) union (select srapprprocess from pmscofferingext where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))))
55	DMACTIONGROUP	action in (select action from wfaction where processname in ((select mgrapprprocess from pmscofferingext where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) union (select srapprprocess from pmscofferingext where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))))
60	DMWFPROCESS	processname in (select srapprprocess from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')) or processname in (select mgrapprprocess from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))
65	DKTICKETTEMPLATE	templateid in (select templateid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))
70	DMMAXGROUP	groupname in (select groupname from pmscofferingauth where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME'))
75	PMSC_OFFERING	itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')

マイグレーション順序	オブジェクト構造	Where 文節
80	PMSC_DOCINFO	docinfoid in (select docinfoid from doclinks where ownertable='PMSCOFFERING' and ownerid in (select itemid from pmscoffering where itemnum in (select offeringnum from pmsscatalogoffmap where itemnum='CATALOGNAME')))

**注:** マイグレーション順序の値は、事前定義オブジェクト構造の間に追加のオブジェクト構造を将来加えることができるよう、5 単位で大きくなっています。

## 9.6.2 PMSC72\_CATALOG

PMSC72\_CATALOG マイグレーション・オブジェクトは、マイグレーション・オブジェクト内の各オブジェクト構造に対する標準 Where 文節と共に提供されています。PMSC72\_OFFERING マイグレーション・グループと同様に、ユーザーはこれらの Where 文節を編集し、マイグレーションするサービス・カタログの名前を指定する必要があります。このためには、カタログ名を(データベースに表示されるとおりに)各 Where 文節に挿入します。

表 9-3 に PMSC72\_CATALOG マイグレーション・オブジェクトを構成するオブジェクト構造と、それらに関連付けられている Where 文節をマイグレーションされる順番にリストします。

表 9-3 すぐに使用可能な PMSC72\_CATALOG の Where 文節

マイグレーション順序	マイグレーション・オブジェクト	Where 文節
	DMMAXGROUP	groupname in (select groupname from pmsscatalogauth where itemnum='CATALOGNAME')
10	PMSCCATALOG	itemnum='CATALOGNAME'

## 9.7 デプロイメント

このシナリオに関するデプロイメントの考慮事項、特にマイグレーションの前提条件については、パッケージをターゲット環境にデプロイする前に理解しておく必要があります。

パッケージ定義が承認され、アクティブ化されたら、パッケージの作成と配布を行うことができます。その後、パッケージをターゲット環境にデプロイします。

**注:**本書では、パッケージのデプロイに必要な標準の手順については繰り返しません。詳細については、以下にある「マイグレーション・マネージャー・ガイド」を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)

ターゲット環境へのデプロイメント後、ユーザーはターゲット環境でサービス・カタログおよびその関連するサービス・オファリングを使い始めることができます。

### 9.7.1 デプロイメントに関する考慮事項

以下に示されたデプロイメントに関する考慮事項について、PMSC72\_CatalogOffering のデプロイメントの前に対応する必要があります。そうしない場合、パッケージのマイグレーションは失敗します。

#### マイグレーションの前提条件

PMSC72\_CatalogueTemplate は、サービス・カタログに直接関連付けられた追加レコードをマイグレーションします。OOTB パッケージ定義は Maximo 内のすべてをマイグレーションする包括的なパッケージではありません。そのため、以下のものを手動でもしくは他のマイグレーション・パッケージを使用して確実にマイグレーションすることが重要です。

- ▶ **作業標準 - PMSC72\_OFFERING** マイグレーション・オブジェクトは作業標準を丸ごとマイグレーションしますが(ネストされた作業標準を除く)、参照される作業員、部品、サービス、および工具はターゲット環境でそれぞれのデータベース・オブジェクト内に作成されません。
- ▶ **ネストされた作業標準 -** 作業標準が作業標準タスクに対して指定された場合(ネストされた作業標準)、ネストされた作業標準は PMSC72\_CatalogueTemplate の一部としてマイグレーションされません。サービス・カタログに適用されたチケット・テンプレートに直接関連付けられた作業標準のみが、パッケージ内でマイグレーションされます。ネストされた作業標準はこれよりも前にマイグレーションする必要があります。
- ▶ **メタデータ -** データ・ディクショナリー・オブジェクトは PMSC72\_CatalogueTemplate に含まれません。サービス・カタログのマイグレーションの前にメタデータをマイグレーションするために使用する、すぐに使用可能なマイグレーション・グループがあります。

- ▶ **応答計画** - すぐに使用可能なサービス・カタログ・マイグレーション・パッケージに含まれません。サービス・グループおよびサービス・オファリングに関連付けられた応答計画は、PMSC72\_CatalogTemplate をターゲット環境にデプロイする前にマイグレーションする必要があります。
- ▶ **上位階層分類** - PMSC\_OFFERING マイグレーション・オブジェクトは、サービス・オファリングによって参照される個々の分類をマイグレーションします。ただし、参照される分類に関連付けられた分類の階層全体はマイグレーションされません。上位階層分類は、その下位階層分類よりも前に作成する必要があります。
- ▶ **循環関係** - アクションがワークフローを参照し、ワークフローがアクションを参照する場合、問題が発生します。PMSC72\_OFFERING は、関連するワークフローに関連付けられた第 1 レベルのアクションをマイグレーションします。ユーザーはサービス・カタログをマイグレーションする前に、まずアクションに参照されるワークフロー、そしてワークフローに参照されるアクションをマイグレーションする必要があります。

## Where 文節による組み込みと除外

Where 文節により、マイグレーション・パッケージのデータが組み込まれたり、除外されたりします。PMSC72\_CatalogTemplate は、各マイグレーション・オブジェクトに関連付けられたそれぞれの Where 文節で指定された「CATALOGUENAME」に関連するオブジェクトのみをマイグレーションします。例えば、作業標準、カタログのオファリング、またはカタログ自身に対して分類が指定されていない場合、マイグレーション・グループは DMCLASSIFICATION オブジェクト構造内のオブジェクトは一切マイグレーションしません。

## PMSC72\_OFFERING マイグレーション・グループの複製

PMSC72\_CatalogTemplate は PMSC72\_OFFERING マイグレーション・グループと共に開発され、PMSC72\_CATALOG マイグレーション・グループよりも低いマイグレーション順序が設定されています。パッケージが適切にマイグレーションされるためには、PMSC72\_OFFERING パッケージが PMSC72\_CATALOG よりも前に処理される必要があります。

特定のマイグレーション・ニーズに対応するために、OOTB PMSC72\_OFFERING マイグレーション・グループを変更する必要がある場合があります。ただし、このマイグレーション・グループは PMSC72\_ServiceTemplate でも使用されているため、複製することが推奨されます。複製して PMSC72\_CatalogTemplate に追加すると、複製された PMSC72\_OFFERING マイグレーション・グループの自動番号が PMSC72\_CATALOG マイグレーション・グループの自動番号よりも大きくなることに注意してください。したがって、複製された PMSC72\_OFFERING マイグレーション・グループよりも前に PMSC72\_CATALOG が処理されます。この場

合、ターゲット環境にすぐに使用可能なマイグレーション・グループ用の前提データが既に存在していないと、パッケージは失敗します。

この問題を回避するには、PMSC72\_OFFERING マイグレーション・グループを複製した後、PMSC72\_CATALOG マイグレーション・グループの「マイグレーション・グループ順序(Migration Group Order)」フィールドの数字を、複製された PMSC72\_OFFERING マイグレーション・グループの数字よりも大きく設定する必要があります。209 ページの図 9-23 に、複製された PMSC72\_OFFERING マイグレーション・グループに自動的に設定された「マイグレーション・グループ順序(Migration Group Order)」を示します。

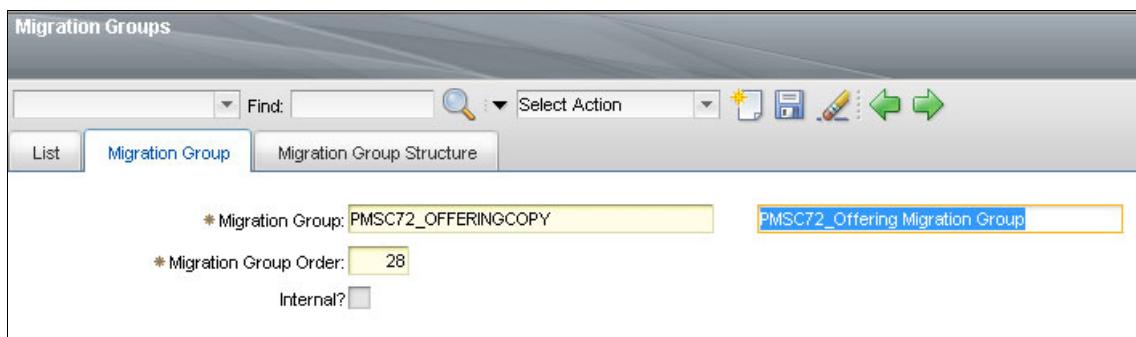


図9-23 マイグレーション・グループ・アプリケーションの「マイグレーション・グループ順序」フィールド





## 一般的なトピック

マイグレーション・マネージャーには、多数の機能が組み込まれています。そのほとんどは、ほぼすべてのマイグレーション・シナリオで使用されます。また、認識しておく必要があるいくつかの考慮事項があります。この章では、以下の一般的なトピックについて概説します。

- ▶ 10.1, 『統合フレームワークとマイグレーション・マネージャーの比較』
- ▶ 215 ページの『10.2 複数言語に関する考慮事項』
- ▶ 218 ページの『10.3 スナップショットと変更』
- ▶ 222 ページの『10.4 埋め込み URL』
- ▶ 223 ページの『10.5 クラスター環境に関する考慮事項』
- ▶ 224 ページの『10.6 開発環境での変更トラッキングと随時レポート』
- ▶ 235 ページの『10.7 管理モード』
- ▶ 236 ページの『10.8 階層データのマイグレーション』
- ▶ 244 ページの『10.9 マイグレーション・マネージャーでのロギングのセットアップ』
- ▶ 250 ページの『10.10 スタート・センターでの構成の表示』

## 10.1 統合フレームワークとマイグレーション・マネージャーの比較

統合フレームワークとマイグレーション・マネージャーは、Tivoli's process automation engine (Tivoli/Maximo Base Service) の 2 つの主要なコンポーネントです。この 2 つのコンポーネントには、さまざまなエクスポート機能とインポート機能があります。それぞれのコンポーネントは、異なる目的で設計されています。表 10-1 では、2 つのコンポーネントの主要な機能を比較します。

表 10-1 統合フレームワークとマイグレーション・マネージャーの比較

	統合フレームワーク	マイグレーション・マネージャー
目的	<ul style="list-style-type: none"><li>▶ 外部アプリケーションとのトランザクションの統合</li><li>▶ 外部アプリケーションまたはレガシー・アプリケーションとの間のエクスポートとインポート</li><li>▶ サービスの呼び出し</li></ul>	同様の製品環境間でのパッケージ・ベースの構成のエクスポートとインポート
プロセス / 分野	エンタープライズ・アプリケーション統合およびサービス指向アーキテクチャー	変更管理
サポートする主なコンテンツ	マスター・データとトランザクション・データ	メタデータ (構成データ)
その他のコンテンツ	すべてのビジネス・オブジェクトを統合の対象にできる	すべてのビジネス・オブジェクトをマイグレーションの対象にできる
作業単位	トランザクション、ファイル、またはメッセージ(单一のドキュメントまたはレコードの場合あり)	パッケージ(複数のレコード)
典型的なユーザー	プログラミング・スキル、アプリケーション統合の経験、ERP システムの知識を備えた実務担当者または実装者	製品構成のスキルおよび変更管理の知識を備えた実務担当者または実装者
フレームワーク	ゼロから構築された統合フレームワーク	ゼロから構築されたマイグレーション・フレームワーク。一部の統合構造を活用する。
主要な機能の違い	Java、XSL、またはルールに基づく広範なマッピング機能と変換機能	マッピング機能または変換機能なし
	特定のアプリケーションで発生するデータベース・イベント・ベースの単一トランザクション	複数のアプリケーション間でのデータベース・イベントに基づいた変更のトラッキングと集約(変更パッケージ・タイプ)

	統合フレームワーク	マイグレーション・マネージャー
	JMS キューを使用した非同期処理	キューなし
	Tivoli's process automation engine (Tivoli/Maximo Base Service) 添付ドキュメントを上位階層レコードで抽出して処理できる	Tivoli's process automation engine (Tivoli/Maximo Base Service) 添付ドキュメントを上位階層レコードで抽出して処理できる。また、任意の添付ファイル (Java クラス、イメージ・ファイル) をパッケージに配置できる。
		パッケージ・デプロイメント時にデータベース構成を自動化する
	Web サービス (WSDL または REST ベース) を公開する	
	Web サービスを呼び出す	
	トランザクション・レベルでの広範なエラー訂正および再送信機能 (メッセージ再処理アプリケーション)	パッケージ・レベルでの限定的なエラー訂正機能 (場合によってはソース環境でパッケージの再作成が必要)
	単一オブジェクト構造の処理 (複数ファイルに対してエクスポートされたデータをインポート用に手動で順序付ける)	複数のオブジェクト構造が単一のマイグレーション・グループにまとめられ、グループ内のすべてのオブジェクト構造が順序付けられるため手動での順序付けが不要。マイグレーション・グループも順序付けられる。
	データを含んでいる XML ファイルを編集し、データ・エレメントとともに実行されるアクションを決定するアクション属性に関連付けることができる	パッケージには編集対象でない複数の XML ファイルが含まれている。アクション属性はマイグレーション・マネージャーによって自動的にセットアップされる。

### 10.1.1 シナリオ：統合フレームワークまたはマイグレーション・マネージャーの選択

このセクションでは、いくつかの主要なシナリオをレビューし、必要なタスクの実行に利用する適切なコンポーネントを識別します。

**シナリオ 1:** 開発者が、Tivoli's process automation engine (Tivoli/Maximo Base Service) のワークフロー・デザイナーおよび関連するアプリケーションを使用して、ITIL ベースのインシデント管理ワークフローを実装しました。このワークフローを実稼働環境にプロモートする必要があります。どのツールを使用すればよいでしょうか。

ワークフローと関連する構成データをパッケージ化する場合に選択するツールは、マイグレーション・マネージャーです。1 つのパッケージに、ワークフ

ローと、任意のアクション、ロール、および通信テンプレートを、顧客が Maximo EAR にビルドする必要があるカスタム・コードとともにに入れることができます。

**シナリオ 2:** レガシー・チケット管理アプリケーションを、Tivoli Service Request Manager 製品に統合する必要があります。過去のチケットを Tivoli 製品にロードする必要があります。また、レガシー・チケット管理アプリケーションは Tivoli Service Request Manager の実稼働が開始されるまで使用されます。

選択するツールは統合フレームワーク (MEA) です。チケットなどのレガシー・データは、フラット・ファイル、インターフェース・テーブル、XML などの複数の方法で、Tivoli Service Request Manager にロードできます。レガシー・ツールから定期的にデータをファイル形式でエクスポートして指定のフォルダーに入れることができます。統合フレームワークはこれを自動的に処理して Tivoli Service Request Manager に送ります。

**シナリオ 3:** 顧客が、在庫 / 購買管理をはじめとする複数の SAP R/3 モジュールを実装しています。特定の SAP モジュールを Maximo Asset Management に統合して、購買および在庫プロセスの全体的な統合を実現することが要求されています。

選択するツールは、統合フレームワーク (MEA) および SAP R/3 アダプターです。SAP R/3 アダプターは、Maximo から SAP R/3 への事前に構築された統合ポイントを提供します。また、カスタムの統合ポイントを構築できます。

**シナリオ 4:** クライアントの開発環境で、開発者チームが CCMDB 製品を構成しています。構成には、新しいオブジェクト (テーブル)、属性 (カラム)、およびドメインの追加が含まれます。また、このクライアントは、既存の CI アプリケーションに新規のタブとダイアログを追加したいと考えています。このすべての構成を、実稼働環境にプロモートする前に、ユーザー受け入れテスト (UAT) 環境にマイグレーションする必要があります。

選択するツールはマイグレーション・マネージャーです。このツールは、構成を開発環境から UAT を経て実稼働環境にプロモートする処理の制御という要求を満たすよう設計されています。また、このツールは、新しいオブジェクトと属性を追加した結果必要となる、基礎となるデータベースへの構造変更を自動化します。これによって、可変のコンテンツを含むパッケージをシームレスにデプロイできます。

**シナリオ 5:** クライアントは、「基本」データ (実装データとも呼ばれます) を開発環境から実稼働環境にマイグレーションすることを必要としています。これは、ロケーションや分類などのデータを実稼働環境で再入力する手間を省くためです。

一般に、基本データは、他のデータとの複数の関係または深い関係がない離散的データ・セットで構成されます。マイグレーション・マネージャーでは、こ

のような基本データをマイグレーションできます。このタイプのデータをサポートするオブジェクト構造がない場合、実務担当者または実装者は、オブジェクト構造を作成し、必要に応じて Java ベースの処理コードを作成する必要があります。このようなオブジェクト構造は、統合フレームワークからも利用できます。統合フレームワークのキュー・ベースの処理、エラー訂正、および再送信機能を使用して基本データをロードすると、マイグレーション・マネージャーを使用する場合より効率的です。

## 10.2 複数言語に関する考慮事項

Tivoli's process automation engine (Tivoli/Maximo Base Service) ベースの製品は、単一の環境およびデータベースで複数の言語をサポートできます。製品環境の言語構成は、製品のインストール時にセットアップされ、基礎となる製品データベースに保管されます。すべての製品環境には 1 つのベース言語が関連付けられており、適切な言語パックをインストールすることで、必要に応じて追加の言語を使用可能になります。追加の言語がインストールされるたびに、翻訳された一連のコンテンツが製品データベース内の専用の複数言語テーブルに追加されます。

複数言語の使用可能化は、ビジネス・オブジェクトの属性に対して構成されます。特定の属性に対して複数言語が使用可能になっている場合は、専用の複数言語テーブルが作成されて、対象の属性の複数言語コンテンツが保管されます。ベース・テーブルには、常にベース言語のコンテンツが保持されます。いくつかの属性に対してあらかじめ複数言語サポートが使用可能になっています。これらの属性によって、追加の言語がインストールされて使用可能になったときに、製品の主要機能がエンド・ユーザーの選択したローカル言語で表示され、使用できるようになります。

この製品の複数言語機能および構成について詳しくは、以下の場所にある資料を参照してください。

[http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.srm.doc\\_7.1/reference/mam71\\_sys\\_admin\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/topic/com.ibm.srm.doc_7.1/reference/mam71_sys_admin_guide.pdf)

マイグレーション・マネージャーでは、ソース製品環境とターゲット製品環境のベース言語が同じであることが必要となります。ターゲット環境のベース言語がソース環境と異なる場合は、マイグレーション・マネージャーからエラー・メッセージ BMXAA5651E が出され、パッケージのデプロイメントが中止されます。

製品環境の複数言語構成は、マイグレーションの観点からサポートする必要があります。複数言語コンテンツをマイグレーションする際には、2 種類のテーブルをサポートする必要があります。

- ▶ LONGDESCRIPTION
- ▶ L\_ 複数言語テーブル

製品と共に提供されている各マイグレーション・マネージャー・オブジェクト構造は、複数言語テーブルを組み込むように設計されています（このようなテーブルが製品とともに出荷された場合）。216 ページの図 10-1 は、LONGDESCRIPTION テーブルと複数言語テーブルの両方を含んだ DMMAXAPPS オブジェクト構造を示しています。

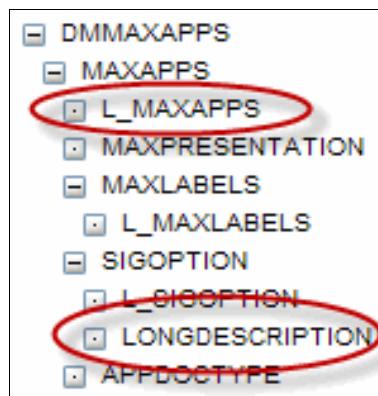


図 10-1 DMMAXAPPS オブジェクト

追加の言語が使用可能になっている製品環境から構成をマイグレーションすると、マイグレーション・マネージャによって、ベース言語のコンテンツが抽出されてベース・テーブルに組み込まれ、さらに追加言語のコンテンツが抽出されて L\_ 複数言語テーブルに組み込まれます。このように、パッケージが作成されるときに、マイグレーション・マネージャによって自動的に複数言語対応のコンテンツがエクスポートされます。

**注:** マイグレーション・マネージャは、L\_ テーブルから使用可能なすべての複数言語コンテンツを抽出します。一部の言語の複数言語コンテンツのみを抽出する機能はありません。

マイグレーション・マネージャ・オブジェクト構造に対するこの手法は、ビジネス・アプリケーションの最も目に触れる部分となる構成をマイグレーションする際に特に重要となります。アプリケーションの各種表示エレメントは、アプリケーション・デザイナーを使用して構成されます。アプリケーション・プレゼンテーションをマイグレーションする場合は、アプリケーション・プレゼンテーションの各種エレメントのローカライズされたコンテンツもマイグレーションする必要があります。表 10-2 では、これらのプレゼンテーション・エレメントを列挙し、複数言語コンテンツを収集する場合に使用可能にする対応するマイグレーション・マネージャ・オブジェクト構造を示します。

表 10-2 プレゼンテーション・エレメントおよび対応するマイグレーション・マネージャー・オブジェクト構造

ユーザー・インターフェース・エレメント	マイグレーション・マネージャー・オブジェクト構造	マイグレーション・グループ
プレゼンテーション XML	DMMAXAPPS	APPLICATION
ラベル	DMMAXAPPS の一部	APPLICATION
すべてのメニュー	DMMAXMENU	APPLICATION
シグニチャー・オプション	DMSIGOPTION	APPSECURITY
条件付きユーザー・インターフェース	DMCTRLGROUP	APPSECURITY
ドメイン	DMMAXDOMAIN	DATADICTIONARY

マイグレーションを実行する場合は、事前に以下の点を考慮する必要があります。

- ソース環境で追加言語が使用可能になっていた場合は、他の構成をマイグレーションする前に、DMLANGUAGE オブジェクト構造を使用して LANGUAGE テーブルに保管されている情報をターゲット環境にマイグレーションする必要があります。DMLANGUAGE オブジェクト構造は、DATADICTIONARY マイグレーション・グループの一部です。ターゲット環境で使用可能になっている言語がソース環境より少ない場合、あるいは別の言語が使用可能になっている場合に、ソース環境からターゲット環境へのマイグレーションを実行すると、マイグレーションは失敗します。
- 複数言語テーブルがサイトで構成されていた場合は、複数言語テーブルを組み込むために更新する必要がある、対応するマイグレーション・マネージャー・オブジェクト構造が存在するかどうかを確認します。複数言語テーブルは、そのテーブルを構成したベース・テーブルの下位階層として追加する必要があります。
- ビジネス・オブジェクトに対して詳細説明機能が構成されていた場合は、LONGDESCRIPTION テーブルを組み込むために更新する必要がある、対応するマイグレーション・マネージャー・オブジェクト構造が存在するかどうかを確認します。LONGDESCRIPTION テーブルは、そのテーブルを構成したベース・テーブルの下位階層として追加する必要があります。

**注:** ソース環境で複数言語テーブルを追加するために既存のマイグレーション・マネージャー・オブジェクト構造を変更する場合は、同じようにターゲット環境のオブジェクト構造を変更する必要があります。

## 10.3 スナップショットと変更

マイグレーション・マネージャー・アプリケーションには、以下の 2 つのタイプのパッケージがあります。

- ▶ スナップショット・パッケージ
- ▶ 変更パッケージ

スナップショット・マイグレーション・パッケージには、パッケージ作成時にソース環境に存在していた構成データ・コンテンツが含まれています。スナップショット・パッケージは、環境で大きな変更がある場合に使用します。例えば、新しい環境や、一連の新しいアプリケーションまたはワークフローを作成する場合などに、このパッケージを使用します。スナップショット・マイグレーションでは、SQL Where 文節を使用して、マイグレーションするコンテンツをフィルターに掛けることができます。図 10-2 は、スナップショット・パッケージ定義の例を示しています。

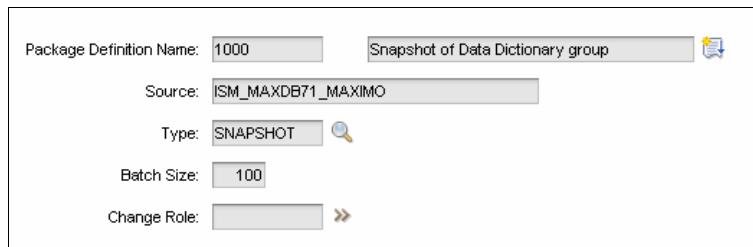


図 10-2 スナップショット・パッケージの定義

**ヒント:** コンテンツをマイグレーションしない場合は、選択したマイグレーション・グループの SQL 文で  $I=0$  を使用します。

スナップショット・パッケージを使用するには、マイグレーション要件全体が、コンテンツ・エレメントの正確な名前まで明確に定義され、識別され、文書化されている必要があります。開発にある程度の時間を要したソース環境では、スナップショット・パッケージを構成するためにかなりの量の構成が必要となります。

変更マイグレーション・パッケージには通常、スナップショット・パッケージに比例して少ないコンテンツ・データが含まれています。このパッケージには、パッケージがアクティブ化された後に作成される構成データが保管されます。

図 10-3 は、変更パッケージ定義の例を示しています。

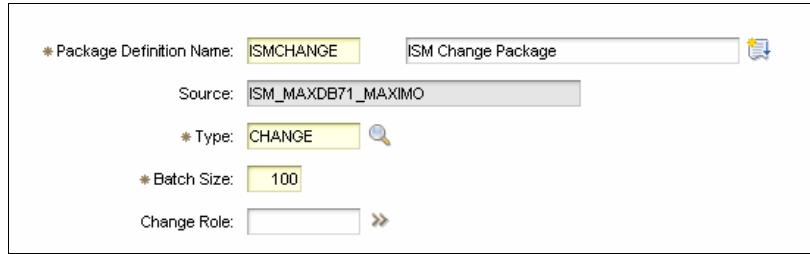


図 10-3 変更パッケージの定義

**注：**「タイプ(Type)」フィールドは、一度定義すると読み取り専用になり、変更できなくなります。

変更パッケージ定義では、「ロールの変更(Change Role)」フィールドが編集可能で、このフィールドにより、特定のロールの変更をトラッキングできます。

変更パッケージは、このタイプのパッケージがアクティブ化された時点でソース環境とターゲット環境が既に同期している場合に役立ちます。異なる変更パッケージがオーバーラップしないようにする必要があります。例えば、複数のパッケージが同じオブジェクトを listen しないようにする必要があります。各開発者が独自の変更パッケージを用意することも役に立ちます。

**注：**管理モードがオンになっているときは、変更パッケージで変更をトラッキングできません。管理モードでは、すべてのイベント・ブロードキャストがシャットオフされます。変更パッケージがアクティブな場合に管理モードをオンにしようとすると、警告が表示されます。

スナップショット・パッケージで使用するために作成されたオブジェクト構造には、変更パッケージで使用できないものがあります。マイグレーション・マネージャーは、変更パッケージ内的一部のデータを処理できません。例えば、階層データを変更パッケージでマイグレーションすることはできません。変更パッケージでは階層順序が認識されません。このため、検証エラーが発生します。階層データの例として、分類、ロケーション、資産などがあります。検証エラーは、同じコンテンツを持つパッケージがオーバーラップしたことによって発生する可能性があります。デプロイメントの前に必ずデプロイメントの変更をプレビューして、コンテンツが検証されるかどうかを確認してください。

**注:** スナップショット・パッケージでは、履歴データが保存されないため、DELETE ステートメントをトラッキングできません。削除された構成コンテントを把握したい場合は、スナップショット・パッケージではなく変更パッケージを使用してください。

変更パッケージ内の変更を表示して管理するには、以下のステップを実行します。

1. マイグレーション・マネージャー・アプリケーションに移動します。「リンク先」→「システムの構成」→「マイグレーション」→「マイグレーション・マネージャー」を選択してください。
2. 関連する変更パッケージを選択し、図 10-4 に示すように、「アクションの選択 (Select Action)」メニューの「イベント・トラッキング・レコードの表示 (View Event Tracking Records)」を選択します。

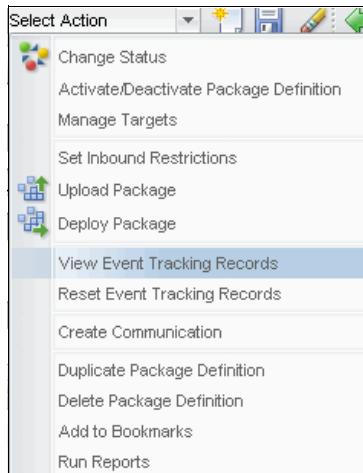


図 10-4 「イベント・トラッキング・レコードの表示 (View Event Tracking Records)」オプション

3. 221 ページの図 10-5 は、変更パッケージのイベント・トラッキング・レコードを示しています。この図から、変更されるオブジェクトおよび関連オブジェクトがわかります。マイグレーションしたくないレコードがある場合は、行の横の「ごみ箱」アイコンをクリックします。下位階層イベント・トラッキング・レコードは自動的に削除されます。

View Event Tracking Records						
Root Event Tracking Records			Child Event Tracking Records			
Object	Action	Primary Keys	Event Date	Event User	Object	Object Id
CLASSTRUCTURE	DELETE	CLASSTRUCTUREID=2524	24.09.2010 16:30:34	MAXADMINIII	CLASSTRUCTURE	866.406
<b>Child Event Tracking Records</b>						
Object	Action	Primary Keys	Event Date	Event User	Parent Object	Parent Object Id
CLASSTRUCTURE	UPDATE	CLASSTRUCTUREID=2522	24.09.2010 16:30:34	MAXADMINIII	CLASSTRUCTURE	866.406
CLASSSPEC	DELETE	CLASSTRUCTUREID=2524,ASSETATTRID=C16AT,TRIBUTE,SECTION=,ORGID=,SITEID=	24.09.2010 16:30:34	MAXADMIN	CLASSTRUCTURE	866.406
CLASSUSEWITH	DELETE	CLASSTRUCTUREID=2524,OBJECTNAME=CI	24.09.2010 16:30:34	MAXADMIN	CLASSTRUCTURE	866.406
CLASSSPECUSEWITH	DELETE	CLASSTRUCTUREID=2524,ASSETATTRID=C16AT,TRIBUTE,SECTION=,ORGID=,SITEID=,OBJECTNA ME=CI	24.09.2010 16:30:34	MAXADMIN	CLASSPEC	17.721.985

図 10-5 変更パッケージに記録されたイベント

- すべてのイベント・トラッキング・レコードを削除する場合は、図 10-6 に示すように、「アクションの選択 (Select Action)」メニューの「イベント・トラッキング・レコードのリセット (Reset Event Tracking Records)」を選択します。

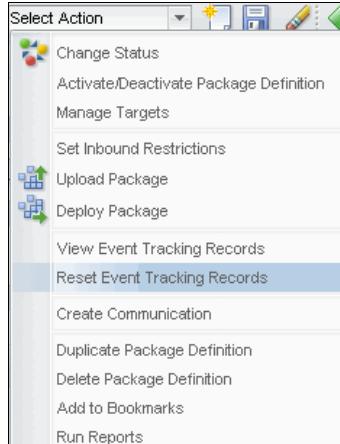


図 10-6 「イベント・トラッキング・レコードのリセット (Reset Event Tracking Records)」オプション

- このオプションを選択すると、確認ダイアログが表示されます。トラッキングされたイベントを削除する場合は、「はい」をクリックします。

## 10.4 埋め込み URL

構成データに埋め込み URL が含まれている場合があります。例えば、システムへのリンクを含んだ電子メールをユーザーに送信するとします。これを行うには、通信テンプレート・アプリケーションを使用します。図 10-7 は、URL が埋め込まれた通信テンプレートの例を示しています。

The screenshot shows the 'Template' configuration screen for 'Escalation for Tracked Survey'. The 'Template' field is set to 'ESCINCSURVEY'. The 'Applies To' field is set to 'INCIDENT'. The 'Accessible From' field is set to 'ALL'. The 'Comm Log Entry?' checkbox is checked.

**Message Details**

To: AFFECTEDBY, REPORTEDBY

cc:

bcc:

\* Send From: maxadmin@mro.com

Reply To:

Subject: :CLASS :TICKETID has been closed. Please give

Message: Report Date - :REPORTDATE Classification - :CLASSSTRUCTUREDESCRIPTION Summary - :D  
http://maximoserver/maximo/ui/maximo.jsp?event=loadapp&value=surveyreq&uniqueid=2&recordkey=:TICKETID

図 10-7 通信テンプレートの例

通信テンプレート、エンドポイント、およびコンテキストでの起動 (LIC) で使用される URL は、一般に環境によって異なります。

このようなオブジェクトをマイグレーションした場合は、後でこれらの URL を手動で訂正する必要があります。通信テンプレートは、マイグレーション後に非アクティブとなります。通信テンプレート内の URL を確認してから、ステータスをアクティブに変更します。LIC やエンドポイントなどの他のオブジェクトはステートレスです。ターゲット環境でこれらのオブジェクトを使用する場合は注意してください。

## 10.5 クラスター環境に関する考慮事項

クラスター環境では、マイグレーション・シナリオとは動作が異なる場合があります。クラスター環境の動作が異なるシナリオは複数あります。

パッケージには、構造化されたコンテンツと構造化されていないコンテンツが含まれている場合があります。構造化されたコンテンツと一部の構造化されていないコンテンツは、パフォーマンスを向上させるために、Java 仮想マシン (JVM) によってキャッシュされます。キャッシュされるコンテンツの例は以下のとおりです。

- ▶ テーブル、ビュー、カラムなどの構造
- ▶ アプリケーション・プレゼンテーション
- ▶ ドメイン

単一サーバー環境でこのタイプのコンテンツをマイグレーションする場合は、そのコンテンツを使用する際に問題は生じません。マイグレーションが完了すると直ちに、このコンテンツを使用できます。

クラスター環境では、各メンバー JVM サーバーが同じバックエンド・データベースに接続してこのコンテンツを読み取り、それぞれのキャッシュにロードします。新しいパッケージをデプロイする場合は、各クラスター・サーバーでキャッシュされたコンテンツをリフレッシュする必要があります。

クラスター内の各 JVM を再起動して、マイグレーション・パッケージ内の変更が各 JVM に反映されるようにします。そうしないと、他の JVM を使用するユーザーは変更の影響を受けなくなります。クラスター環境では、すべての JVM を再起動できるように、変更をスケジュールする（停止期間）必要があります。完全な停止期間をスケジュールできない場合は部分的または定期的な停止手法を使用して、JVM サーバーが段階的に再起動され、常に 1 つ以上のサーバーが使用可能となるようにすることができます。この手法では、負荷分散装置を使用する必要があります。

**ヒント：**完全な停止期間でパッケージをデプロイする間は、クラスター内の JVM を 1 つを除いてすべて停止します。これは、マイグレーション・プロセス全体が終了するまで使用されるデプロイメント JVM です。マイグレーション・プロセス完了後は、必ず管理モードをオフにして JVM を始動してください。

**警告：**各 JVM にログインする最初のユーザーは、サーバーがキャッシュを構築するまで処理の遅さを経験することになります。その後はパフォーマンスが向上します。

## 10.6 開発環境での変更トラッキングと随時レポート

一般に誤解されているのは、マイグレーション・マネージャーが後から生まれた機能であり、構成が完了した後に使用されるツールであるということです。

ほとんどのシナリオでは、構成変更のプロモートにスナップショット・パッケージが使用されますが、マイグレーションを実行するのは常に開発者とは限りません。また複数の開発者が関与するプロジェクトでは開発者が複数存在します。この文書に示す例からわかるように、スナップショット・パッケージを効果的に使用できるようにするには、マイグレーション要件を構成エレメントの正確な名前まで明確に定義し、識別し、文書化する必要があります。

このセクションでは、マイグレーション・マネージャーの変更パッケージが開発サイクルの重要なエレメントであること、およびその効果的な使い方を説明します。

### 10.6.1 変更パッケージを使用するタイミング

変更パッケージは、ソース環境とターゲット環境が同期し、かつ開発作業が始まっていない場合に役立ちます。ただし、これを効果的に行うには、マイグレーション・マネージャーを開発計画に組み込んで、開発サイクルでその機能を利用する必要があります。

最も単純かつ理想的なシナリオは、開発環境の構成アクティビティーを変更パッケージで収集するように開発計画で決定し、ターゲット環境へ組織的にデプロイした場合です。

ロールが割り当てられ、パッケージ・コンテンツの定義が作成されています。最も重要な点としては、開発者のロールはオーバーラップしません。つまり、異なる変更パッケージが同じマイグレーション・オブジェクトを listen している場合でも、2人の開発者が同じ構成コンテンツで作業することはできません。

ロールが不要である場合（ロールが開発者など1つのみである場合）は、変更パッケージを1つだけ定義して、マイグレーション要件の一部として識別されたすべてのマイグレーション・オブジェクトを含める方が適切です。このタイプのパッケージをデプロイすると、トラッキングされるすべての構成イベントが1回のデプロイメントで実行されます。

それぞれが特定のマイグレーション・グループを1つだけトラッキングする複数のパッケージを、マイグレーション・オブジェクトをオーバーラップさせることなく定義することもできます。このタイプの変更パッケージをデプロイするには、デプロイメントの手順の中でグループの依存関係を考慮する必要があります。部分的に完了した開発をプロモートしてパッケージを作成し、

その後にパッケージをリセットし、アクティブ化してさらに開発を続ける場合は、組織的な取り組みが必要となることもあります。

## 10.6.2 変更パッケージを使用した開発アクティビティーのトラッキング

開発環境は、特定の期間内に製品を構成して製品ソリューションの要件を満たすために、複数の開発者が連係して作業する共有環境です。

この最も一般的なシナリオでは、開発アクティビティーが非常に急速に開始されます。マイグレーション・マネージャー変更パッケージの計画と組み込みを最初から行うことは不可能です。

ほとんどのマイグレーション・マネージャーの実装と同様に、スナップショット機能は構成をプロモートするために使用されます。特定のコンテンツをプロモートするためには、*SQL Where* 文筋コマンドを使用して特定の構成コンテンツを対象にすることが不可欠ですが、このコンテンツを定義して構成キー値の名前を識別することは大きな課題となります。

マイグレーションの必要なレコードは、命名規則(例えば、接頭部「CUST」で始まるオブジェクト・キーなど)によって識別するか、またはスプレッドシートやその他の表形式ドキュメントで明確に文書化する必要があります。この情報がないと、マイグレーションを開始するまでにかなりの時間と手間がかかります。

変更パッケージでは、特定の構成コンテンツ・レコードに影響するデータベース・イベント(追加、更新、削除)を収集することができます。この機能は、作成、配布、デプロイを行わないトラッキング・デバイスとしてのみ使用できます。その後に、トラッキングされた変更に基づいてスナップショット・パッケージを作成できます。

## 10.6.3 変更ロールの構成

変更パッケージ機能では、特定のユーザーがロールを使用して行った変更をトラッキングすることができます。これは複数の開発者がいるチームで非常に役立つことがわかっています。各開発者には、ログインして構成を実行するためのユーザー ID が割り当てられます。

担当者または担当者グループには、ロールを割り当てるすることができます。使用的グループのタイプは、チームで決定されます。

次の例では、図 10-9 に示す *DEVELOP* という担当者グループに基づいて、図 10-8 に示すように「担当者グループ(Person Group)」タイプの *DEVELOPER* ロールを作成します。このグループには 2 つの担当者 ID が割り

当てられており、それぞれの ID は 2 人の開発者のユーザー ID と一致します。この例でのユーザーは、WILSON と LWAYNE です。

Role: DEVELOPER      Developer Group Role  
\* Type: Person Group  
Object:    
\* Value: DEVELOP   
Parameter:  
E-mail?   
Broadcast?

図 10-8 DEVELOPER ロール

**重要:** 図 10-8 でブロードキャスト・フラグのチェック・マークが外されると、マイグレーション・マネージャーはグループのすべてのメンバーがトリガーするイベントを記録しません。グループで指定されたデフォルトの担当者のイベントのみが記録されます。

Person	Name	Use for Organization	Use for Site
WILSON	Mike Wilson		
LWAYNE	Lisa Wayne		

図 10-9 DEVELOP 担当者グループ

変更パッケージでは「担当者」タイプのロールも使用できます。担当者グループの利点は、動的な環境でユーザーを追加および除去できることです。

ここで定義する 2 つのロールは、データ・ディクショナリーとアプリケーション・フレームワークに影響するアプリケーションで変更を構成する 2 人の開発者を対象としています。

この例では、2 人の作業はオーバーラップしていません。WILSON はデータ・ディクショナリー・グループ内のオブジェクトを変更し、LWAYNE はアプリケーション側に対する構成を行います。これは単に開発をどのように組織化で

きるかを示す例であり、両方のユーザーが両方のグループで構成を実行する可能性があります。ここでの重要なポイントは、これらのイベントをトラッキングできるようにすることです。

#### 10.6.4 変更パッケージ定義

ロールを定義したら、次に変更パッケージを作成します。変更パッケージでは、パッケージに割り当てられたロールで定義されているユーザーが特定の構成オブジェクトに対して行った変更がトラッキングされます。

この例では、標準の DATADICTONARY マイグレーション・グループに含まれているすべてのオブジェクト構造に対する変更をトラッキングするために、このグループで変更パッケージ定義を作成します。この定義は、DEVELOPER ロールに関連付けられます。

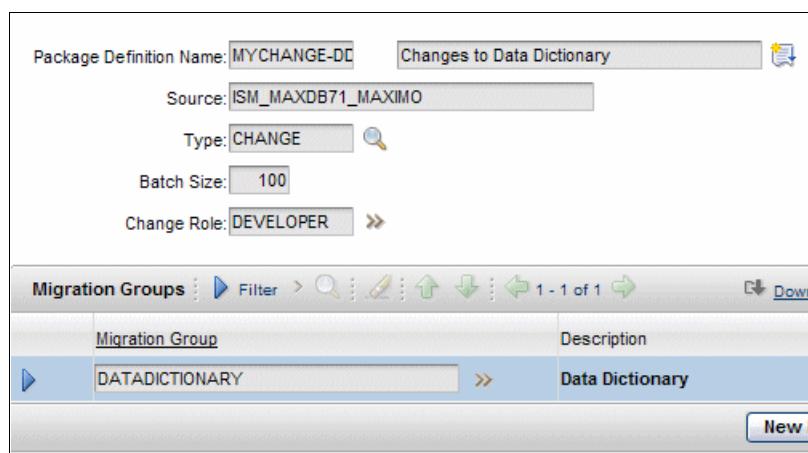


図 10-10 変更パッケージ MYCHANGES-DD

アプリケーションに対する変更をトラッキングするための 2 番目の変更パッケージ定義を作成します。新しいマイグレーション・グループが、APPLICATIONS マイグレーション・グループのコピーとして使用されます。この新しいグループは *MYAPPLICATIONS* と名付けられ、従属グループはすべて除去されています。また、このパッケージは、228 ページの図 10-11 に示すように DEVELOPER ロールに関連付けられます。

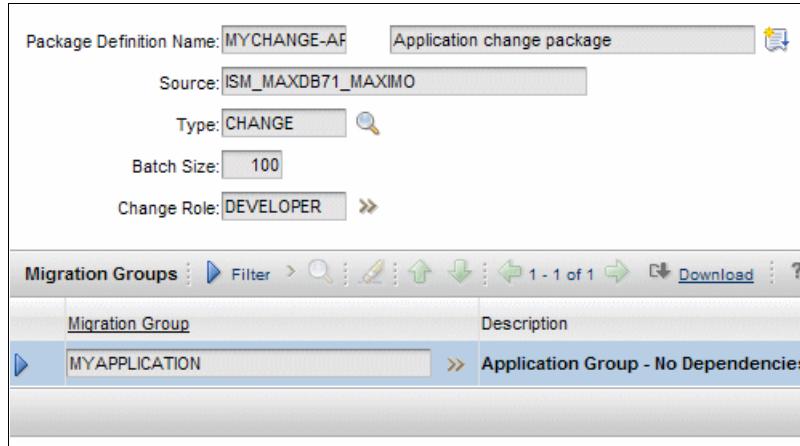


図 10-11 変更パッケージ MYCHANGES-APP

**注:** 変更トラッキングは、リストで指定されたマイグレーション・グループのすべての Maximo ビジネス・オブジェクト (MBO) に対して実行されます。パッケージ定義ステータスが WAPPR である間は、必要に応じてグループを追加できます。

### 10.6.5 変更イベントのトラッキング

開発者は、パッケージを承認してアクティブ化した後に構成作業を開始できます。これらのイベントはトラッキングされ、イベント・テーブルに保管されます。

データベースの構成アプリケーションを使用してデータベースの構造変更を行う場合は、その変更を適用するために管理モードをオンにする必要があります。この間はトラッキングが停止します。データベース構成が完了し、システムで管理モードがオフになるまでは、変更を実行しないようにする必要があります。

トラッキングに変更パッケージを使用する方法を説明するために、必要とされた変更を以下に要約します。

- ▶ ALN タイプの新しいドメイン *MYRESULTS* が追加されました。値は PASSED と FAILED です。
- ▶ ASSET テーブルに *MYRESULTS* という新しい属性が作成されました。この属性にはテスト値の結果が保持され、検証用に新しいドメインが構成されます。

- ▶ テスト結果用の新しい属性が ASSET アプリケーションに追加され、検証するため構成されます。

開発者が構成を完了すると、変更パッケージ定義から正確なイベントを確認できるようになります。そのためには、「**アクションの選択**」→「**イベント・トラッキング・レコードの表示 (View Event Tracking Records)**」をクリックします。

Object	Action	Primary Keys	Event Date	Event User	Object	Object Id
MAXDOMAIN	ADD	DOMAINID=MYRESULTS	10/5/10 16:00:02	WILSON	MAXDOMAIN	768
MAXOBJECTCFG	UPDATE	OBJECTNAME=ASSET	10/5/10 16:02:04	WILSON	MAXOBJECTCFG	192

図 10-12 パッケージ MYCHANGES-DD のイベント

Object	Action	Primary Keys	Event Date	Event User	Object	Object Id
MAXAPPS	UPDATE	APP=ASSET	10/5/10 16:18:21	LWAYNE	MAXAPPS	8

図 10-13 パッケージ MYCHANGES-APP のイベント

この例は、2人の開発者が行ったイベントのサンプル構成を示しています。各パッケージ定義には、アプリケーションでのこれらの記録されたイベントをリストする機能があります。

**注:** 戦略によっては、単一のロールおよび変更パッケージを利用して変更イベントをトラッキングすることができます。単一の変更パッケージには、構成済みとして識別されたすべてのマイグレーション・グループが含まれている必要があります。

ここで重要な情報は、「オブジェクト (Object)」カラムと「プライマリー・キー (Primary Keys)」カラムです。この情報によって、スナップショット・パッケージのマイグレーション要件を定義できます。

例えば、229 ページの図 10-12 に示すイベントから、パッケージに必要なオブジェクトと SQL 条件は以下のようになります。

表 10-3 スナップショット・パッケージ内のデータ・ディクショナリー・オブジェクト用の Where 文節

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMMAXDOMAIN	MAXDOMAIN	<code>domainid in ('MYRESULTS')</code>
DMMAXOBJECTCFG	MAXOBJECTCFG	<code>objectname in ('ASSET')</code>

また、229 ページの図 10-13 に示すイベントから、スナップショット・パッケージに必要なオブジェクトと SQL 条件(表 10-4 を参照)を判別するための十分な情報を収集できます。

表 10-4 スナップショット・パッケージ内のアプリケーション・オブジェクト用の Where 文節

マイグレーション・オブジェクト	オブジェクト	Where 文節
DMMAXAPPS	MAXAPPS	<code>app in ('ASSET')</code>

複数の開発者がいる環境では、複数の構成が同時に行われます。多くの場合、開発者は他のツールやアプリケーションを習得する必要があり、またタスクの遂行に集中しなければなりません。

変更パッケージ機能を使用すると、文書化されていない場合もある必須情報を得ることができ、開発サイクル全体を通して、あるいは日常業務や保守作業の一環として、継続的にスナップショット・パッケージの準備とデプロイを行うことができます。

何が誰によって変更されたかを記録することは適切な習慣であり、それによつて責任所在と変更プロセスの文書化が可能になります。ただし、このトラッキング情報はデータベースに保管され、トラッキング・スプレッドシート形式でしか文書化できません。

次に、この情報を文書化およびスナップショット・パッケージの準備の目的で抽出する方法を説明します。

## 10.6.6 変更イベントの随時レポートの作成

随時レポートまたはクエリー・ベース・レポート (QBR) を構成して、すべてのパッケージを対象に全ユーザーについてトラッキングされたすべての変更を収集することができます。このような随時レポートを構成して実行するために、1回限りのタスクが複数実行されます。

### レポート・オブジェクト構造の定義

DMPKGEVENTS オブジェクト構造を作成するため、「リンク先」→「システムの構成」→「マイグレーション」→「オブジェクト構造」をクリックします。新しいオブジェクト構造の詳細は以下のとおりです。

1. DMPKGEVENTS オブジェクト構造の「使用者 (Consumed By)」フィールドで、REPORTING を指定します。
2. DMPACKAGEDEF MBO を上位階層、DMPKGEVENTTRK MBO を下位階層として指定します。
3. DMPKGEVENTTRK 下位階層 MBO に対し、「関係 (Relationship)」に DMPKGEVENTTRK\_ROOT を指定します。これにより、ルート(またはプライマリー) MBO に対する変更のみが随時レポートに表示されるようになります。
4. オブジェクト構造を保存します。

新しいオブジェクト構造を図 10-14 に示します。

Object	Parent Object	Object Location Path	Relationship	Object Order	User
DMPACKAGEDEF		DMPACKAGEDEF		1	<input checked="" type="checkbox"/>
DMPKGEVENTT	DMPACKAGEDEF	DMPACKAGEDEF/DMPKGEVENTTRK	DMPKGEVENTTRK_ROOT	1	<input checked="" type="checkbox"/>

図 10-14 レポート用の DMPKGEVENTS オブジェクト構造

図 10-15 に示すように、レポート・オブジェクトとアプリケーションの関連付けが、Maximo Base Services Fix Pack 7.1.1.6 で新しくなりました。

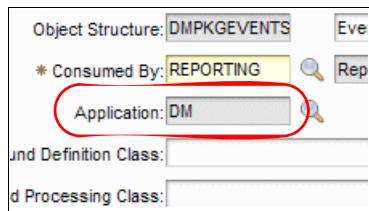


図 10-15 レポート・オブジェクト構造用のアプリケーション

このステップが完了すると、マイグレーション・マネージャー・アプリケーションに新たなレポート・アイコンが追加されます。

### レポート・オブジェクト構造に対するセキュリティー・アクセス権限の付与

このステップは、Maximo Base Services Fix Pack 7.1.1.6 のユーザーが対象です。

1. 「リンク先」 → 「管理」 → 「レポート」 → 「レポート管理」をクリックし、「一覧」タブから「アクションの選択」→「レポート・オブジェクト構造セキュリティーの設定」をクリックし、マイグレーション・マネージャー・アプリケーションをフィルターに掛けます。
2. セキュリティー・リストに MAXADMIN グループを追加し、「OK」をクリックします。図 10-16 にこのセキュリティー・オブジェクトを示します。

Report Object Structure Security	
Report Object Structures	
Description	Application
	migration
Event tracking object	Migration Manager
Report Object Structure Security	
Group	Description
MAXADMIN	Maximo Administrators (Super Users)
<a href="#">New Row</a>	

図 10-16 DMPKGENVENTS のレポート・オブジェクト・セキュリティー

## 随时レポートの構成

マイグレーション・マネージャー・アプリケーションで、「レポートの作成」アイコンをクリックして「クエリー・ベース・レポート (Query Based Report)」ポップアップ・ダイアログを開きます。Maximo Base Services Fix Pack 7.1.1.6 では、このダイアログは複数のタブで構成されています。

最初のタブは「スタイル (Style)」タブです。以下のように構成します。

1. 随時レポートの名前 (MM\_ALL\_EVENTS) を指定します。
2. このレポートをパブリックにするために、「パブリック (Public?)」チェック・ボックスにチェック・マークを付けます。
3. 「レポートの保存 (Save Report)」を選択して、後続のユーザーがレポート設計を使用できるようにします。
4. レコードを単純な行 / カラムの形式で表示する場合は、「要約レポート (Summary Report)」を選択します。

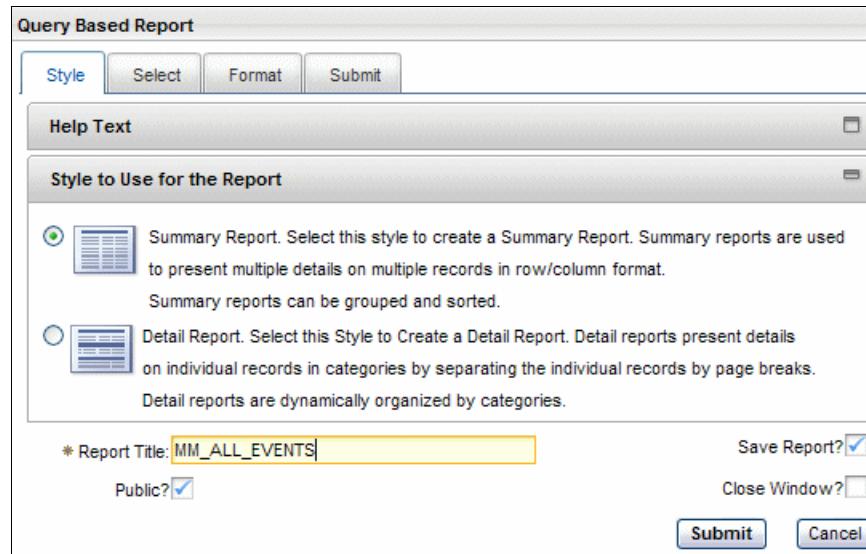


図 10-17 MM\_ALL\_EVENT クエリー・ベース・レポートの「スタイル (Style)」タブ

5. 「選択 (Select)」タブで、図 10-18 に示すように「選択済みフィールド (Selected Fields)」リストを構成します。

Selected Fields	Column Order	Report Label	Field Value	Category
	1	Package Definition Name	PKGDEFNAME	DMPACKAGEDEF
	2	Object	OBJECT	DMPKGEVENTTRK
	3	Primary Keys	PRIMARYKEYS	DMPKGEVENTTRK
	4	Action	ACTION	DMPKGEVENTTRK
	5	Event User	EVENTUSER	DMPKGEVENTTRK
	6	Event Date	EVENTDATE	DMPKGEVENTTRK

図 10-18 QBR 用の MM\_ALL\_EVENTS 選択済みフィールド

6. オプション「**アプリケーションから現在のクエリーとフィルターを適用**」にチェック・マークを付けておくと、マイグレーション・マネージャー・アプリケーションの「一覧」タブからパッケージ定義のフィルター処理を使用できるようになります。

この例では、234 ページの図 10-19 に示すように、パッケージの「タイプ (Type)」と「アクティブ (Active)」に対してフィルターが掛けられています。

Package Definition Name	Source	Description	Type	Status	Active
			change		Y
MYCHANGE-DD	ISM_MAXDB71_MAXIMO	Changes to Data Dictionary	CHANGE	APPR	<input checked="" type="checkbox"/>
MYCHANGE-APP	ISM_MAXDB71_MAXIMO	Application change package	CHANGE	APPR	<input checked="" type="checkbox"/>
<input type="checkbox"/> Select Records					

図 10-19 変更パッケージ・リスト・トラッキング・イベント

## 随時レポートの実行

この時点で、レポートを PDF レポートとして直ちに実行するように送信できます。定義を閉じると、マイグレーション・マネージャー・アプリケーションでレポートを呼び出せるようになります。その場合は、「**アクションの選択**」→「**レポートの実行**」オプションをクリックし、保存されているレポート MM\_ALL\_EVENTS をクリックします。この出力結果を 235 ページの図 10-20 に示します。

Package Definition Name	Object	Primary Keys	Event User	Event Date
MYCHANGE-DD	MAXDOMAIN	DOMAINID=MYRESULTS	WILSON	10/5/10 4:00:02 PM
MYCHANGE-DD	MAXOBJECTCFG	OBJECTNAME=ASSET	WILSON	10/5/10 4:02:04 PM
MYCHANGE-APP	MAXAPPS	APP=ASSET	LWAYNE	10/5/10 4:18:21 PM
MYCHANGE-APP	MAXPRESENTATION	APP=ASSET	LWAYNE	10/5/10 4:18:20 PM

**Number of Records:** 4  
**Saved Where Clause:** ( dmpackagedef.active = 1 and dmpackagedef.type like '%CHANGE%' )  
**Dynamic Where Clause:**  
 10/6/10 16:55:32      1 / 1

図 10-20 MM\_ALL\_EVENT クエリー・ベース・レポートの出力

このレポートは、スプレッドシート (XLS) としても出力できます。また、スケジュールすることもできます。

## 10.7 管理モード

Tivoli's process automation engine (Tivoli/Maximo Base Service) でバックエンド・データベースのオブジェクトに対する構造変更を構成するためには、そのデータベースに排他的にアクセスする必要があります。以前は、アプリケーション・サーバーをシャットダウンして(ユーザー・アクセスを制限するため)構成変更を正常に適用した後に、この構成が行われていました。

現在では、管理モードを使用することで、アプリケーション・サーバーをシャットダウンすることなく、アプリケーションでユーザー・アクセスを制限し、対応するデータベース構成を行えるようになりました。

また管理モードでは、モードがオフになるまでクーロン・ジョブが中断され、イベント・リスナーが停止します。

この操作モードは、マイグレーション・マネージャーにも影響します。マイグレーション・マネージャーでデータベース・オブジェクトに対して行われた構成変更(構造変更とも呼ばれます)をデプロイする間は、このモードをオンにする必要があります。また、MAXAPPS オブジェクト、つまりアプリケーショ

ン・デザイナーでの変更を含むパッケージをデプロイするときも、このモードをオンにする必要があります。

したがって、デプロイされるパッケージのタイプによっては、ユーザーのセッションで変更を反映させるために管理モードが必要となります。ただし、パッケージをデプロイする間は常にこのモードをオンにして、ユーザーが変更を予期できるようにした方がよいでしょう。

複数サーバー環境またはクラスター環境で作業する場合は、ほかにも考慮事項があります。223 ページの『10.5 クラスター環境に関する考慮事項』を参照してください。

## 10.8 階層データのマイグレーション

以前は、階層データのマイグレーションは非常に困難でした。現在ではこの問題は解決されています。以下のセクションでは、ロケーション階層をマイグレーションするシナリオを示します。

この作業を行うために、階層データで生じる特有の状態があります。つまり、オブジェクト構造は自己参照する必要があり、自己参照するためには構造自体にデータベース参照が必要となります。

階層データのマイグレーションを作成するために必要なステップを以下に示します。

1. 階層データを識別します。
2. テンプレートとして使用するオブジェクト構造を識別します。
3. 自己参照関係を作成します。
4. ステップ 2 で識別したオブジェクト構造を複製します。
5. オブジェクト構造をマイグレーションします。
6. データをマイグレーションします。

### マイグレーションする階層データの識別

このシナリオでは、6 階の研修施設が完成したと想定します。資産を実稼働させる必要があるため、階層をテストから実稼働に移行する必要があります。

図 10-21 では、ロケーション階層をドリルダウン・ビューで示しています。

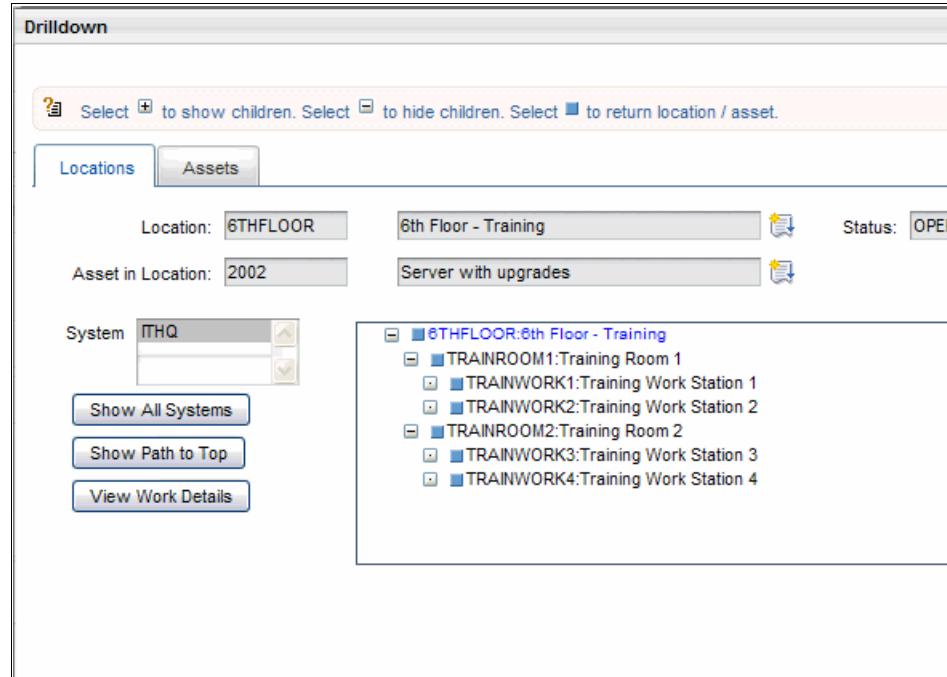


図 10-21 ロケーション階層の例

### テンプレートとして必要なオブジェクト構造の識別

マイグレーションする階層データのタイプを識別したら、テンプレートとして使用する適切なオブジェクト構造を識別する必要があります。ロケーション・データを移行することから、238 ページの図 10-22 は MXOPERLOC オブジェクト構造を新しい構造のテンプレートとして使用する必要があることを示しています。

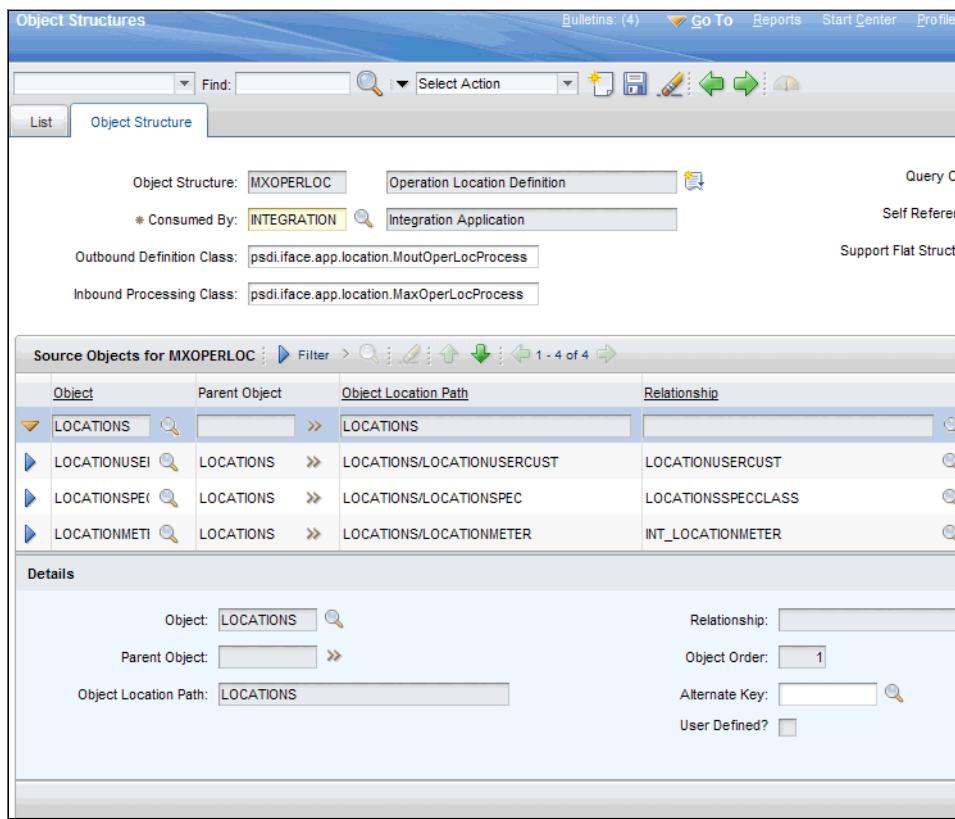


図 10-22 MXOPERLOC オブジェクト構造

オブジェクト構造を識別したので、新しいオブジェクト構造を作成するためにはどの関係が必要であるかを判別できます。

### 新しい自己参照関係の作成

テンプレート・オブジェクト構造で規定されている関係の性質を判別したら、データベースの構成アプリケーションを使用してその関係を作成する必要があります。ここではロケーション階層データを使用しているため、既存の CHILDREN 関係を使用して自己参照用の新しい関係を作成できます。239 ページの図 10-23 にその CHILDREN 関係を示しています。

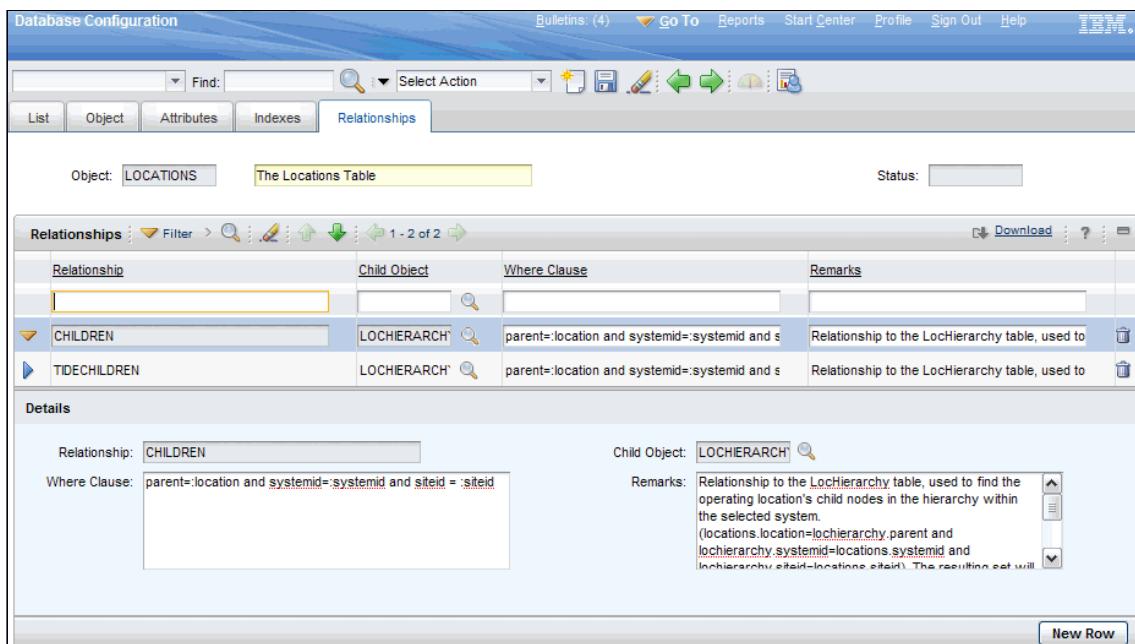


図 10-23 ロケーションの CHILDREN 関係

- 最初に「新規行 (New Row)」をクリックします。新しい関係の名前を入力します。「下位階層のオブジェクト (Child Object)」に LOCATIONS と入力します。現在のロケーションの下位階層をすべて選択する新しい Where 文節を入力します。以下に例を示します。

```
location in (select location from lochierarchy where parent=
:location and systemid = :systemid and siteid = :siteid)
```

- 関係の目的を定義する注釈を入力します。その設定例を 240 ページの図 10-24 に示します。

**注 :** 関係の下位階層テーブルは上位階層と同じであるため、アウトバウンド処理クラスを使用して選択されたレコードごとに、すべての下位階層も選択されます。その結果、この処理クラスによってツリー内のすべての下位階層が再帰的に選択されます。

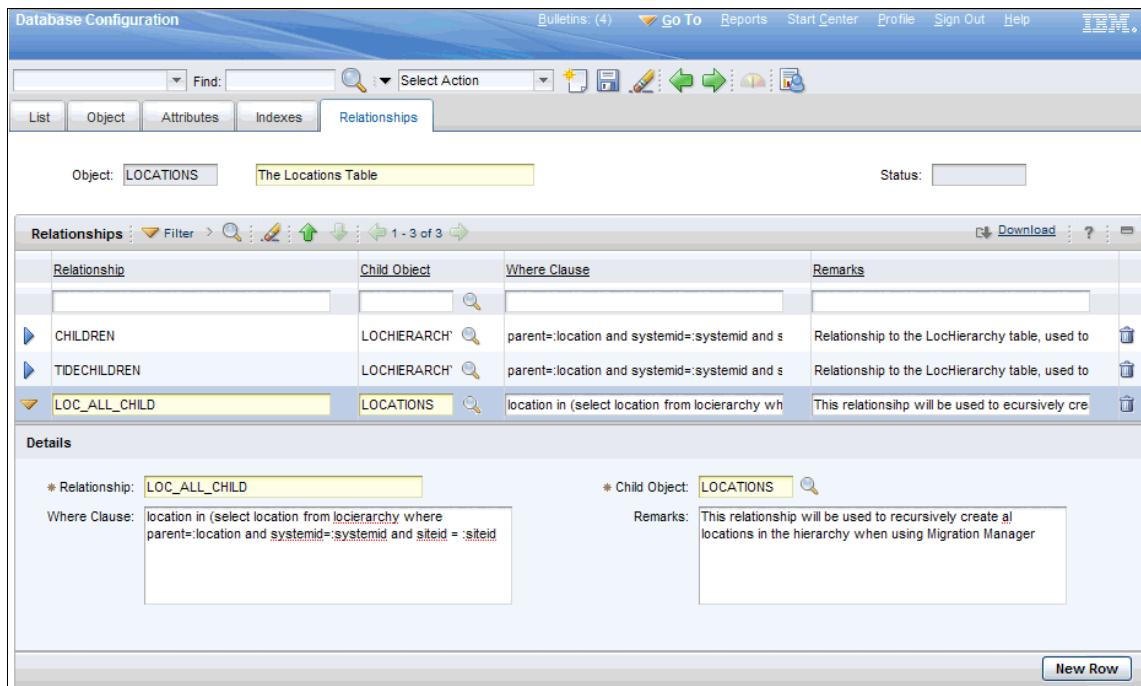


図 10-24 新しい自己参照関係

レコードを保存し、オブジェクト構造アプリケーションに戻ります。

### 新しいオブジェクト構造の作成

新しい関係を作成したので、238 ページの図 10-22 で識別したオブジェクト構造を複製できます。

- 「アクションの選択」→「オブジェクト構造の複製」をクリックします。新しいオブジェクト構造は、MXOPERLOC オブジェクト構造と同一であることがわかります。
- 241 ページの図 10-25 に示すように、オブジェクト構造の新しい名前を入力します。

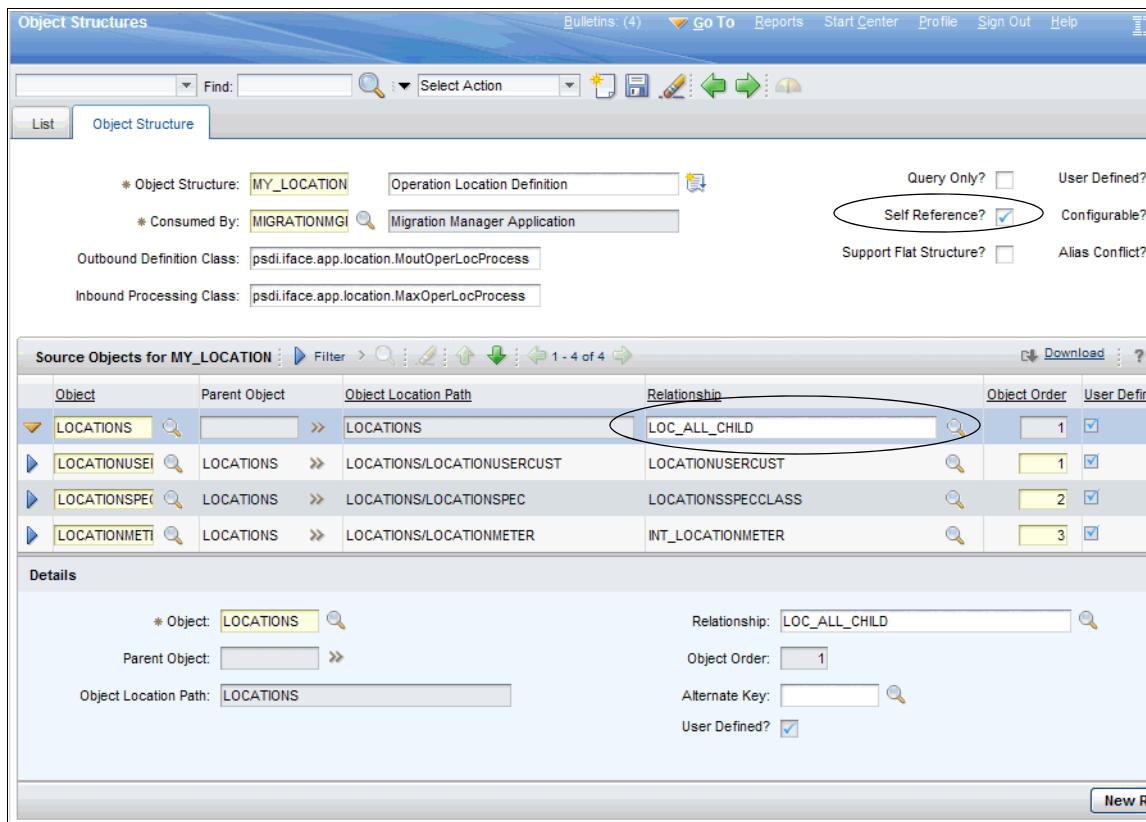


図 10-25 マイグレーション・サービスの新しいオブジェクト構造

ここでは 2 つの点に注目します。

- ▶ 「自己参照 (Self Reference?)」チェック・ボックスにチェック・マークが付いている点
- ▶ 直前に作成した関係を使用している点

## オブジェクト構造のマイグレーション

オブジェクト構造を作成したので、次にオブジェクト構造と関係をマイグレーションするマイグレーション・パッケージを作成する必要があります。

これを行うには、一連の新しいマイグレーション・グループを作成するか、または定義済みのマイグレーション・グループを使用します。定義済みのグループを使用する場合は、送信するレコードを限定するために、SQL Where 文節値の数を ‘1=2’ に設定する必要があります。

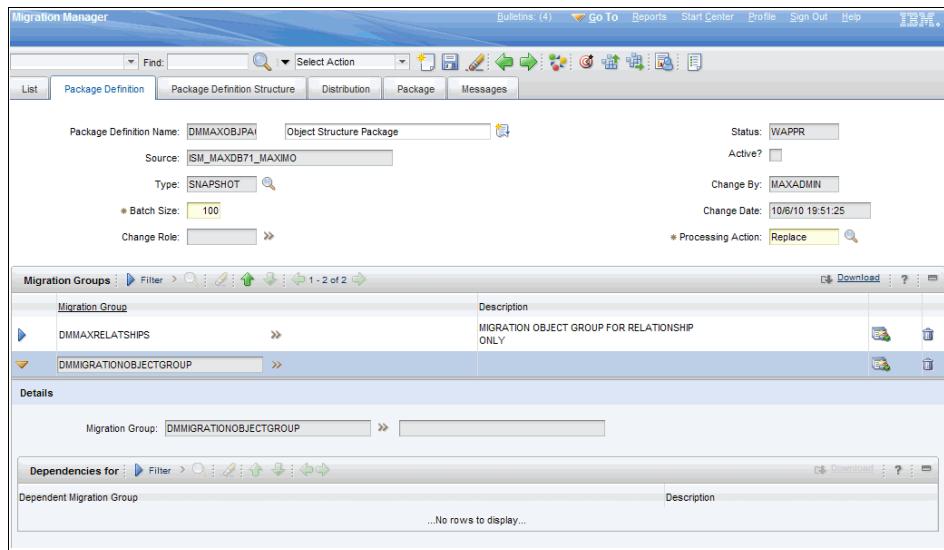


図 10-26 新しいオブジェクト構造を移行するための新しいマイグレーション・パッケージ

この例では、関係とオブジェクト構造のみがグループ内のマイグレーション・オブジェクトであることがわかります。同様に、243 ページの図 10-27 で示すように Where 文節によってグループが限定されています。

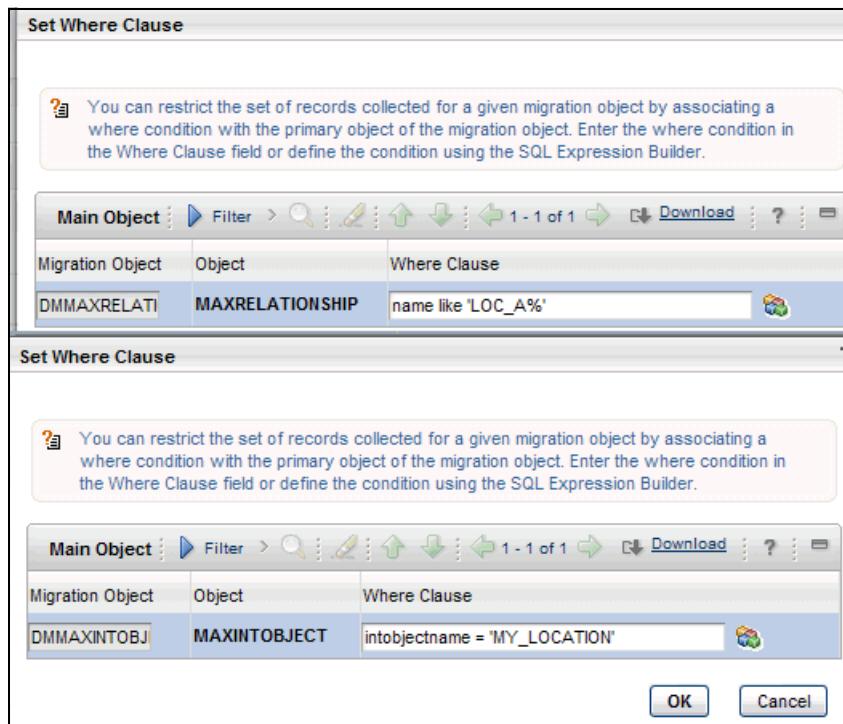


図 10-27 限定されたマイグレーション・パッケージで複数のグループに対して Where 文節を設定する例

オブジェクト構造が限定された新しいマイグレーション・グループを作成し、その限定されたマイグレーション・グループに対して SQL Where 文節を設定する方が簡単です。これにより、パッケージをきわめて迅速に処理できます。

### オブジェクト構造のマイグレーション

これで、ロケーション階層をマイグレーションできるようになりました。244 ページの図 10-28 に示すように、オブジェクト構造が 1 つしかないマイグレーション・グループでパッケージを作成してください。

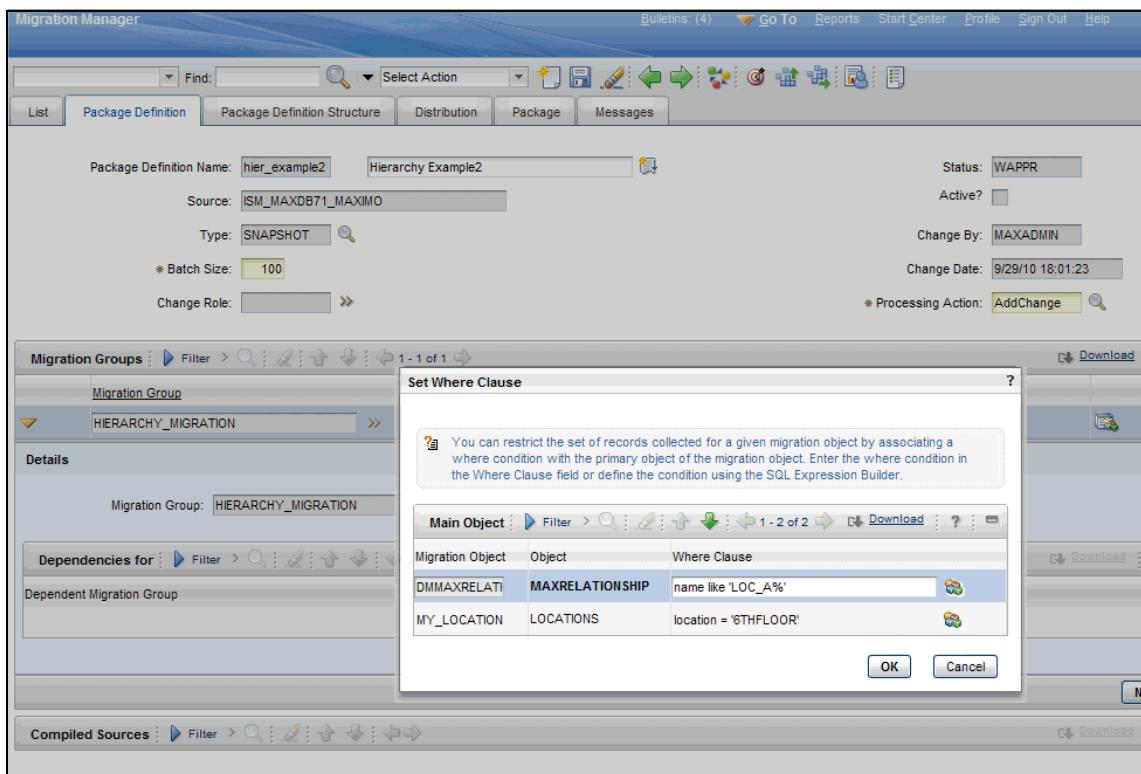


図 10-28 ロケーション階層のマイグレーションの例

この時点でのこの文書で示した他のすべての例と同様にマイグレーション・パッケージを処理します。

## 10.9 マイグレーション・マネージャーでのロギングのセットアップ

「マイグレーション・マネージャー・ガイド」([http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamdoc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamdoc_7.1/pdf/mam71_migration_mgr_guide.pdf)) および「IBM Maximo Asset Management、IBM Tivoli Asset Management for IT、IBM Tivoli Service Request Manager システム管理者ガイド」([http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamdoc\\_7.1/pdf/mam71\\_sys\\_admin\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamdoc_7.1/pdf/mam71_sys_admin_guide.pdf)) では、ロギング・アプリケーションの一般的な使用について説明されています。このセクションでは、失敗したマイグレーション・パッケージのトラブルシューティングを行う際に必要

な出力を得られるようにシステムをセットアップするための具体的な方法を説明します。

最適な結果を得るためにロギングをセットアップするには、以下を実行する必要があります。

- ▶ ルート・ロガーのセットアップ
- ▶ ルート・ロギング・フォルダーの設定
- ▶ マイグレーション・マネージャーの出力をロギングするためのアッパnderのセットアップ
- ▶ マイグレーション・マネージャー・サービスでのロギング・レベルの設定

## ルート・ロガーの管理

ルート・ロガーを管理するには、以下のステップを実行します。

1. ロギング・アプリケーションに移動し、図 10-21 に示すように「**アクションの選択 (Select Action)**」ドロップダウン・ボックスをクリックします。

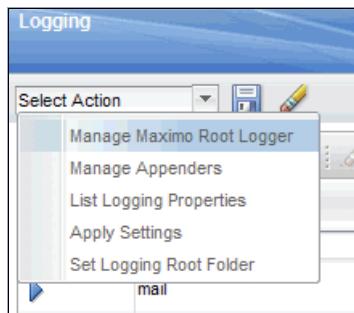


図 10-29 ルート・ロガーの管理

2. 245 ページの図 10-30 に示す新しいダイアログ・ボックスが表示されます。

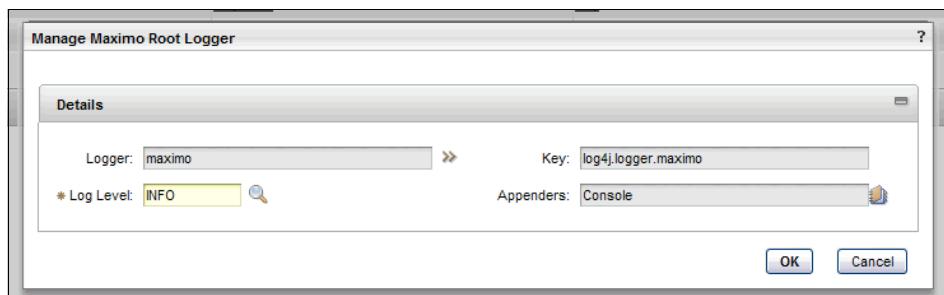


図 10-30 ルート・ロガー・ダイアログ・ボックス

3. ここでは、ルート・ロギング・レベルを設定し、メインのアッパenderを変更できます。どのサービスにも具体的な設定を行わない場合は、システムのロギング全体に影響するので注意してください。

## ロギング・フォルダーの設定

次に、ルート・ロギング・フォルダーを設定する必要があります。この設定を行わない場合は、ルート・フォルダーを設定しない場合にデフォルトで設定される WebSphere のログのロケーションが、通常は以下のフォルダー構造にあることに注意してください。

\ibm\WebSphere\AppServer\profiles\ctgAppSrv01\maximo\logs

242 ページの図 10-26 は、ルート・フォルダーを他のロケーションに設定する場合のルート・ロギング・フォルダーの設定方法を示しています。

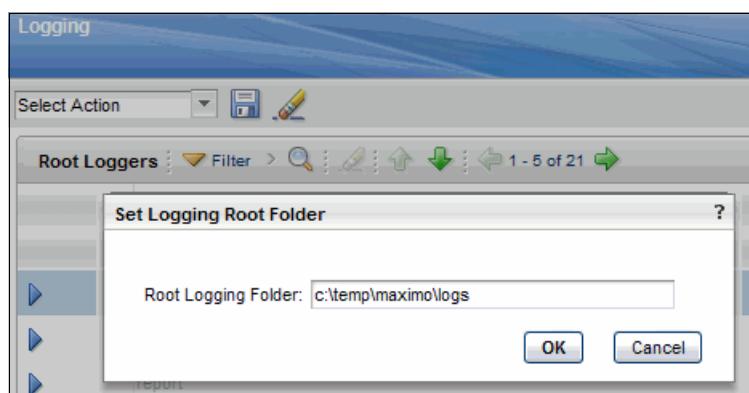


図 10-31 ユーザーによって定義されたルート・ロギング・フォルダー

このダイアログ・ボックスを表示するには、「アクションの選択」→「ロギング・ルート・フォルダーの設定」をクリックします。

## 新しいアッパenderの作成

次のステップでは、マイグレーション・マネージャーのアクティビティーを他のすべてのロギング・アクティビティーから分離するために、ユーザー固有のアッパenderを作成します。これにより、エラー・インシデントを分離する必要があるときに、エラー・ログが見やすくなります。

1. 247 ページの図 10-32 に示す「アッパenderの管理 (Manage Appender)」ダイアログ・ボックスを開くために、「アクションの選択」→「アッパenderの管理」をクリックします。

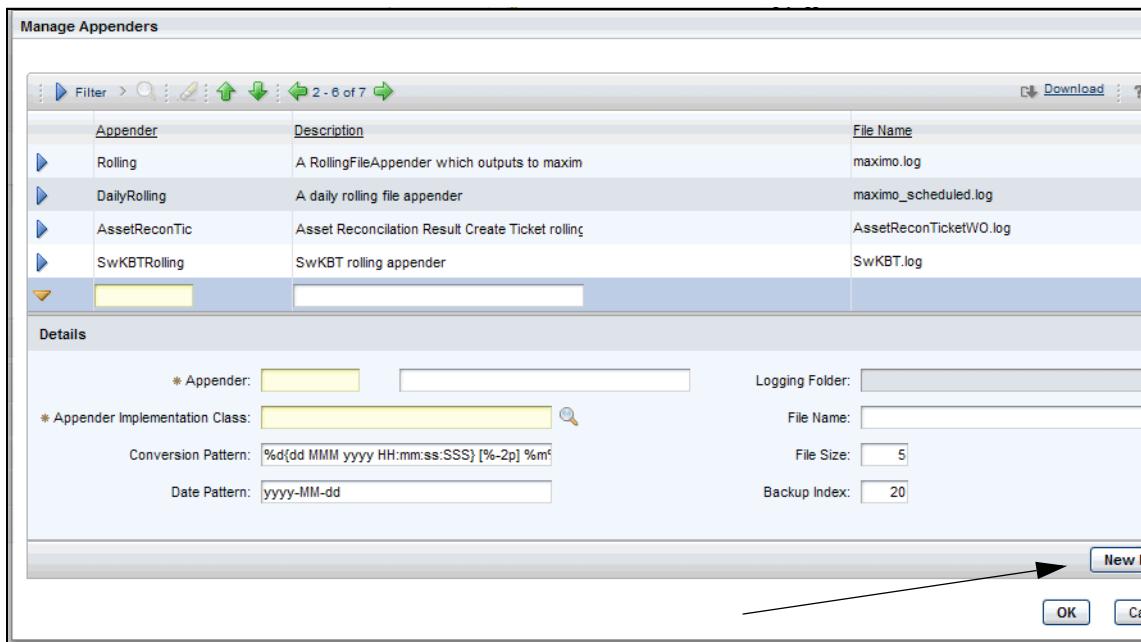


図 10-32 「アッパenderの管理 (Manage Appenders)」ダイアログ・ボックス

2. 「新規行 (New Row)」ボタンをクリックし、作成するアッパenderの名前を入力します。説明を追加し、実装クラスを選択します。
3. 最後にアッパenderで使用するファイル名を追加し、「OK」をクリックします。参考のために、244 ページの図 10-28 に例を示しています。

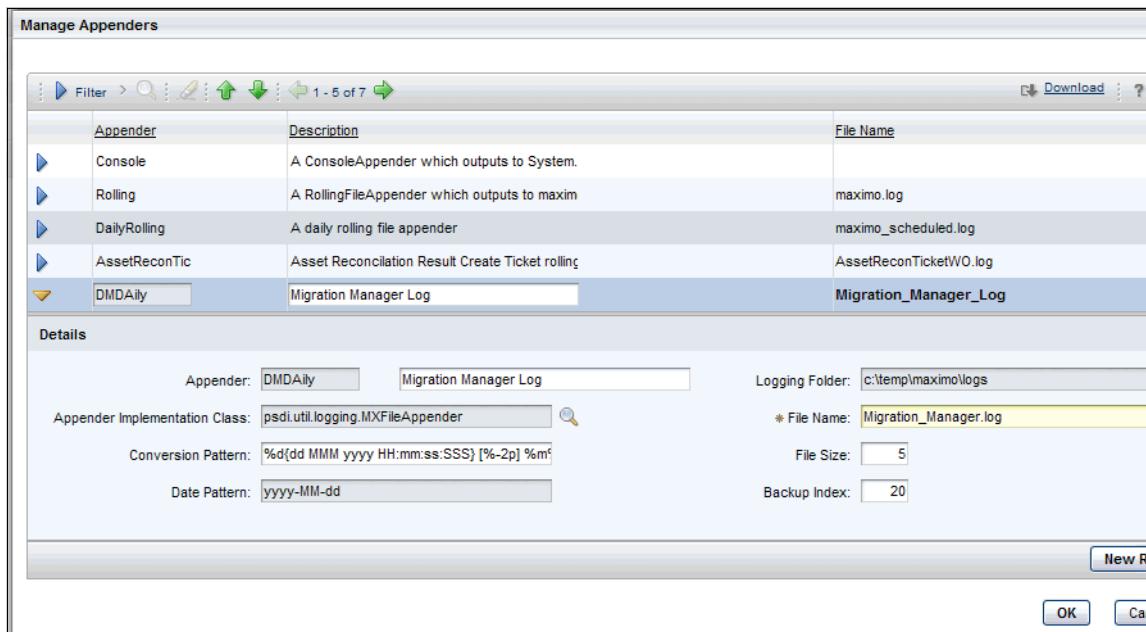


図 10-33 新しいアベンダーの例

この図からわかるように、ファイル・サイズとバックアップ・インデックスも設定できます。経験から変更の必要性を判断できるようになるまでは、デフォルトを使用することをお勧めします。

### マイグレーション・マネージャー・サービスでのロギング・レベルの設定

最後に、マイグレーション・マネージャーの特定のルート・ロガーにアクセスして、リストで dm を探します。

1. 249 ページの図 10-34 に示すように、矢印をクリックしてエントリーを展開します。ログ・レベルを「DEBUG」に設定します。

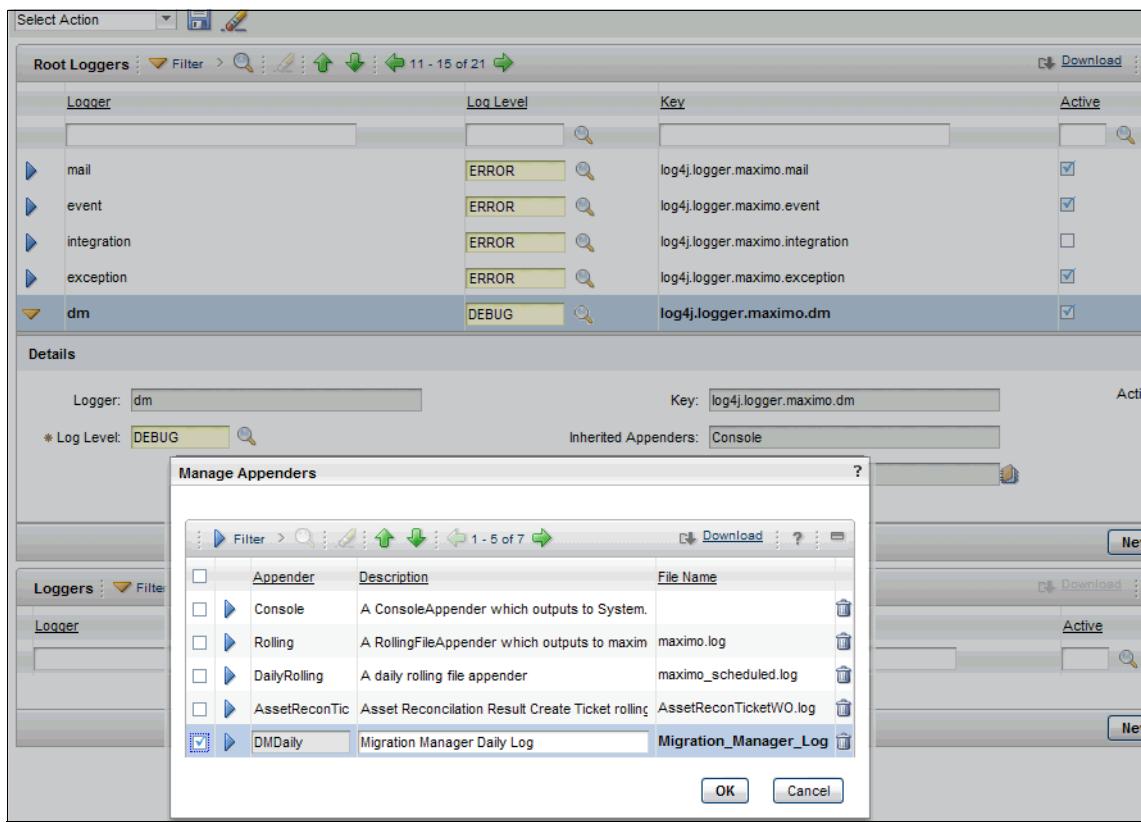


図 10-34 アッパenderの設定

2. 次に、「アッパender (Appender)」フィールドの横にあるアイコンをクリックし、作成したアッパenderを選択します。チェック・ボックスをクリックし、「OK」をクリックします。
3. 「IBM Maximo Asset Management、IBM Tivoli Asset Management for IT、IBM Tivoli Service Request Manager システム管理者ガイド」([http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_sys\\_admin\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_sys_admin_guide.pdf))で説明されているように、「保存」アイコンをクリックし、「アクションの選択」→「設定の適用」をクリックします。
4. ダイアログ・ボックスの応答を受け入れて、システムの使用を開始します。
5. 250 ページの図 10-35 は、前のプロセスでエラーのあるパッケージを処理した後の結果を示しています。この図は、253 ページの『第 11 章 プロセス・モニタリング』(264 ページの図 5)でもう一度示します。

```

20 Sep 2010 17:12:09:360 [DEBUG] Thu Sep 20 17:12:09 EDT 2010*****DMAppBean.deployPackage(): Deploy begins *****
20 Sep 2010 17:12:09:360 [INFO] Migration Manager deployment started for package hierarchy_Example_ISM_MAXIMO_20100928160956, type CFGDATA and configuration order 1.
20 Sep 2010 17:12:09:360 [DEBUG] Object name MAXRELATIONSHIP
20 Sep 2010 17:12:09:360 [DEBUG] Operation Sync
20 Sep 2010 17:12:09:360 [DEBUG] Processing record LOCATIONS-LOC_ALL_CHILD
20 Sep 2010 17:12:09:360 [DEBUG] ZONE is reported
20 Sep 2010 17:12:09:360 [DEBUG] *****MAXRELATIONSHIP*****
20 Sep 2010 17:12:09:360 [DEBUG] Object name MAXRELATIONSHIP
20 Sep 2010 17:12:09:360 [DEBUG] Operation Sync
20 Sep 2010 17:12:09:360 [DEBUG] Processing record SPRELATEDDATASET-LOC PARENT
20 Sep 2010 17:12:09:360 [DEBUG] ZONE is reported
20 Sep 2010 17:12:09:360 [DEBUG] *****
20 Sep 2010 17:12:09:360 [INFO] Migration Manager deployment started for package hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956, type CFGDATA and configuration order 2.
20 Sep 2010 17:12:09:360 [ERROR] Could not deploy configuration data for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956. Please check log entries for more information.
psdi.util.MxApplicationException: [ERROR] Could not import configuration records for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956.
please see log for details.
at psdi.util.MxApplicationException: BMXA1281E - Object Structure MY_LOCATION does not exist.
at psdi.iface.mos.MboMUtil.getObjectName(MboMUtil.java:453)
at psdi.iface.mos.MiService.loaddata(MiService.java:156)
at psdi.pkg.DMPackage.deployStagingData(DMPackage.java:1414)
at psdi.pkg.DMPackage.deployPackage(DMPackage.java:906)
at psdi.webclient.beans.dm.DMAppBean.deployPackage(DMAppBean.java:321)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:618)
at psdi.webclient.system.session.AsyncProcess.run(AsyncProcess.java:95)
at java.lang.Thread.run(Thread.java:810)

30 Sep 2010 17:12:09:393 [ERROR] Could not import configuration records for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956.
please see log for details.
at psdi.util.MxApplicationException: BMXA1281E - Object Structure MY_LOCATION does not exist.
at psdi.iface.mos.MboMUtil.getObjectName(MboMUtil.java:453)
at psdi.iface.mos.MiService.loaddata(MiService.java:156)
at psdi.pkg.DMPackage.deployStagingData(DMPackage.java:1414)
at psdi.pkg.DMPackage.deployPackage(DMPackage.java:906)
at psdi.webclient.beans.dm.DMAppBean.deployPackage(DMAppBean.java:321)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:618)
at psdi.webclient.system.session.AsyncProcess.run(AsyncProcess.java:95)
at java.lang.Thread.run(Thread.java:810)

30 Sep 2010 17:12:09:393 [ERROR] Import fails for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956 with type CFGDATA and configuration data order 2.
30 Sep 2010 17:12:09:640 [ERROR] Could not deploy package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956. Please check log entries for more information.
null
30 Sep 2010 17:12:09:640 [ERROR] Could not deploy package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928160956. Please check log entries for more information.
java.lang.Exception: psdi.util.MxApplicationException: BMXA1281E - Object Structure MY_LOCATION does not exist.
at psdi.util.MxApplicationException: BMXA1281E - Object Structure MY_LOCATION does not exist.
at psdi.util.MxApplicationException: BMXA1281E - Object Structure MY_LOCATION does not exist.
at psdi.pkg.DMPackage.deployStagingData(DMPackage.java:1542)
at psdi.pkg.DMPackage.deployPackage(DMPackage.java:906)
at psdi.webclient.beans.dm.DMAppBean.deployPackage(DMAppBean.java:321)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:618)
at psdi.webclient.system.session.AsyncProcess.run(AsyncProcess.java:95)
at java.lang.Thread.run(Thread.java:810)
```

図10-35 サンプル・エラー・ログ

## 10.10 スタート・センターでの構成の表示

管理者、開発者、および実装者にとって、開発環境での重要な構成設定を、ワークフロー・デザイナーのような特定のアプリケーションだけでなくスタート・センターでも表示できると便利です。重要な構成が「一目でわかる」ようになり、スタート・センターで表示されることで、関係者は構成の数や範囲を判別でき、その結果としてスナップショット・マイグレーション・パッケージを準備できます。

各種の構成を作成する際に一貫した命名規則が使用されていれば、このような表示を行うスタート・センターを比較的速く設計できます。使用された命名規則に基づいて目的の構成を取得するようにクエリーを構築し、それらのクエリーにスタート・センター結果セット・ポートレットを関連付けることができます。各ポートレットは、開発およびマイグレーションの観点から最も関連性の高い情報を表示するように構成できます。

例えば、マイグレーションが必要となる可能性があるワークフロー・プロセスを取得して表示するには、以下のように単純なクエリーを作成します。

```
processname like 'RB%' and active = '1'
```

このクエリーを、以下のような主要なワークフロー・プロセス情報を表示するスタート・センター結果セット・ポートレットに関連付けることができます。

- ▶ プロセス

- ▶ 説明
- ▶ オブジェクト
- ▶ プロセスの改訂
- ▶ 有効
- ▶ アクティブ
- ▶ 変更者
- ▶ 変更日

スタート・センターを保存して適切なセキュリティ・グループに関連付けると、製品環境にログインしたユーザーに直ちに主要な情報が表示されます。252ページの図 10-36 は、ポートレットによってビジネス・オブジェクト、ワークフロー・プロセス、およびセキュリティ・グループを表示するスタート・センターの例を示しています(画面の一部分を掲載)。

Accelerator Objects			
Object	Extends Object	Entity	Class
TKOWNERHISTORY		TKOWNERHISTORY	com.ibm.tsd.app.ticket.PmOwnerHistoryS
SOLUTION		SOLUTION	com.ibm.tsd.pmsol.app.PmSolutionSet
TICKET		TICKET	psdi.app.ticket.TicketSet
PROBLEM	TICKET	PROBLEM	com.ibm.tsd.pmpro.app.PmProblemSet
SR	TICKET	SR	com.ibm.tsd.pmsrv.app.PmSrvReqSet
TKSTATUS		TKSTATUS	com.ibm.tsd.app.ticket.PmTKSLAHoldStat
RELATEDRECORD		RELATEDRECORD	psdi.app.ticket.RelatedRecordSet
WORKLOG		WORKLOG	psdi.app.ticket.WorkLogSet
CLASSSTRUCTURE		CLASSSTRUCTURE	com.ibm.ism.cci.app.classetcatalog.CCIC
INBOUNDCOMM		INBOUNDCOMM	psdi.common.emailstner.InboundCommSe

Accelerator Workflow			
Process	Description	Object	Process Revision
A61REPLY	User Replied	SR	1
A61	Request Fulfillment	SR	5
A612	Receive and Approve Request	SR	6
A612.1	Receive Service Request and Apply Response Plan	SR	3
A612.2	Validate User Profile	SR	1
A612.3	Confirm Configuration Information	SR	1
A612.4	Verify Support Status	SR	2
A612.6	Decline Further Action	SR	2
A612.7	Categorize Service Request	SR	4
A613	Fulfill or Route Service Request	SR	10

Accelerator Security Groups	
Group	Description
A61.REQUESTOR	RF Requestor Group
A61.ANALYST	RF Analyst Group
A61.ADMIN	RF Administrator Group
A61.MANAGER	RF Manager Group
A65 ANALYST	Incident Analyst Group
A65.ADMIN	Incident Administrator Group
A65.MANAGER	Incident Manager Group
A65.REQUESTOR	Incident Requestor Group
A65.OWNER	Incident Owner Group

図 10-36 ビジネス・オブジェクト、ワークフロー・プロセス、およびセキュリティー・グループを表示するサンプルのスタート・センター



## トラブルシューティング

本章では、マイグレーション・パッケージが意図したとおりにマイグレーションされない場合に実務担当者が適用する基本的なアプローチを説明します。ここでは、これまでに発生した最も一般的な問題を示し、マイグレーションの失敗原因を特定するのに役立つ手法の詳細と、パッケージを再処理してマイグレーションを正常に実行するためのパッケージ修正方法を説明します。

本章の内容は以下のとおりです。

- ▶ 254 ページの『11.1 一般的なマイグレーション・パッケージ・エラー』
- ▶ 263 ページの『11.2 マイグレーション・パッケージ・エラーの解決方法』
- ▶ 269 ページの『11.3 マイグレーション・パッケージ・エラーの防止方法』

## 11.1 一般的なマイグレーション・パッケージ・エラー

このセクションでは、これまでに発生した最も一般的なエラーについて詳しく説明します。「マイグレーション・マネージャー・ガイド」([http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_migration_mgr_guide.pdf))では、アプリケーションの使用時に発生する主要な3つのタイプのエラーについて簡単に説明しています。これらのエラーは、パッケージの作成、配布、およびデプロイメントという3つのカテゴリーに分類されます。このセクションでは、デプロイメントが失敗する一般的な状況について説明します。

### 11.1.1 インストールの違いによるパッケージ・デプロイメント・エラー

インストールおよび製品の資料には、ソース・システムとターゲット・システムのソフトウェア・インストールが同じである場合にのみマイグレーション・マネージャーが機能することが明記されています。しかし、そうではない場合がしばしば見受けられました。その場合は、次のアプローチを利用して問題を修正することができます。

#### インストールの違いの特定

ターゲット・システムとソース・システムの両方で、システム情報を確認します。これを行うには、アプリケーションの右上隅にある「ヘルプ (Help)」→「システム情報 (System Information)」をクリックします。図 11-1 にその方法を示します。

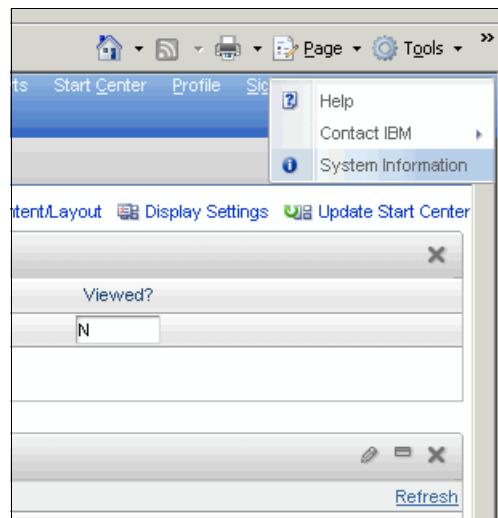


図 11-1 システム情報へのアクセス

次に、ソース・システムにインストールされているコンポーネントを確認して、ターゲット・システムと比較する必要があります。255 ページの 図 11-2 に示されている情報をよく確認してください。



図 11-2 「システム情報 (System Information)」画面

この例では、ソース・システムに多くの製品が含まれています。いくつかのコンテンツ・パッケージをインストールした後で、一部のコンテンツ・パッケージを使用しないことを決める場合もあります。そして、そのコンテンツをターゲット・システムにインストールしないことにします。256 ページの 図 11-3 を参照してください。

System Information	
App Server	IBM WebSphere Application Server 6.1
User Name	MAXADMIN
Version	Service Desk Integration MEA 7.2.0.00 Build 20091111D DB Build V7200-03 IBM Content Catalog 7.1.null.00 Build 1 DB Build V710-04 SRM Service Request Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 IBM Tivoli Configuration Management 7.2.0.01 Build 20100218D2 DB Build V7201-01 IBM Tivoli Change Management Content 7.2.0.00 Build 20091111D DB Build V7200-14 IBM Tivoli Integration Composer 7.2.1.0 Build 201003231145 DB Build V7210-02 IMBot SRM Service Desk 7.2.0.00 Build \${env.BUILD_NAME} DB Build V7200-05 IBM Tivoli Common Process Components 7.2.0.01 Build 20100109D DB Build V7201-02 SRM Service Desk Content - Best Practice Users 7.2.0.0 Build 20091111D DB Build V7200-05 Naming Reconciliation Services 1.0.0.0 Build 200907211418 DB Build V7200-04 SRM Service Desk 7.2.0.1 Build 20100316D2 DB Build V7201-04 IBM Tivoli Asset Management for IT 7.2.1.0 Build 201003231509 DB Build V7210-96 SRM Screen Capturer 7.2.0.0 Build 20091111D DB Build V7200-02 IBM Tivoli Common Process Components for Service Providers 7.2.0.0 Build 20091105D DB Build V7200-04 SRM Problem Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 SRM Solution 7.2.0.1 Build 20100316D2 DB Build V7201-01 SRM Incident Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 SRM SLA Hold 7.2.0.1 Build 20100316D2 DB Build V7201-03 SRM Service Catalog Base 7.2.0.1 Build 20100316D DB Build V7201-05 IBM Tivoli Asset Management for IT for Service Providers 7.2.1.0 Build 201003231509 DB Build V7210-11 SRM Service Desk for Service Providers 7.2.0.0 Build 20091111D DB Build V7120-03 CI PMP Content 7.2.0.00 Build 200911081214 DB Build V712-01 SRM Search 7.2.0.1 Build 20100316D2 DB Build V7201-02 SRM Service Desk Demo Data 7.2.0.0 Build 20091111D DB Build V7200-04 SRM Survey Management 7.2.0.1 Build 20100316D2 DB Build V7201-01 IBM Tivoli ITL V3 Support 7.2.0.0 Build 201004112300 DB Build V7200-02 IBM Tivoli UI Widgets 7.2.0.01 Build 201002251643 DB Build V7201-02 IBM Tivoli Change Management 7.2.0.01 Build 20100215D2 DB Build V7201-02 ServiceProvider 7.1.1.1 Build BUILD DB Build V7121-03 HFDB Build HF7121-01 IMI for SRM Service Desk 7.2.0.00 Build 201002200030 DB Build V7200-02 CCI PMP 7.2.0.01 Build 201002151749 DB Build V7201-05 SRM Service Catalog Content 7.2.0.1 Build 20100316D DB Build V7201-06 Base Services 7.1.1.6-LA20100506-1424 Build 20091208-1415 DB Build V7116-173 HFDB Build HF7116-0
Server OS	Windows Server 2003 5.2 build 3790 Service Pack 2
License #	BMXAA4039E - MRO Software Internal Use Only
(c) Copyright IBM Corp. 2007	
<b>OK</b>	

図 11-3 「システム情報 (System Information)」画面

ソースとターゲットを注意深く調べると、ターゲット・システムにソースと同じインストール・パッケージが含まれていないことがわかります。この場合、マイグレーション固有の問題が発生します。なぜなら、マイグレーション・マ

ネージャーが最初に行う処理は、パッケージ・マニフェストをデータベース(MAXVARS テーブル)と照合して、製品インストールが同じかどうかを確認することであるためです。

### 11.1.2 誤ったソースの指定によるパッケージ・デプロイメント・エラー

パッケージを正しく選択していない場合、デプロイメントが失敗します。この状況は、複数の環境用のパッケージを中央ファイル・サーバーを使用して格納している場合に発生します。これは珍しいことではなく、プロジェクトの開発時に多くのパッケージをすべて同じフォルダーに格納している場合に最もよく起ります。

#### パッケージ・マニフェストの確認

パッケージ・マニフェストには、パッケージのコンテンツを分析するために必要な情報が含まれています。パッケージ・マニフェストを調べることで、ソース・システムがコンシューム側環境であることが判明する場合がしばしばあります。マニフェストを使用することにより、パッケージを正しくデプロイすることができます。図 11-4 は、ソース・システムの情報を示しています。

```
<?xml version="1.0" encoding="UTF-8" ?>
- <PACKAGEMANIFEST>
- <PACKAGEHEADER>
<PKGDEFNAME>Hierarchy_Example</PKGDEFNAME>
<SOURCE>ISM_MAXDB71_MAXIMO</SOURCE>
<PACKAGE>Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057</PACKAGE>
<BASELANGUAGE>EN</BASELANGUAGE>
<CREATEBY>MAXADMIN</CREATEBY>
<CREATEDATE>2010-09-28T17:00:57-04:00</CREATEDATE>
- <COMPONENTVERSION>
<RDBMS>DB2/NT : SQL09053</RDBMS>
```

図 11-4 パッケージ・マニフェストの XML

この XML タグのソースには、ターゲット・システムに対する検証を行うために必要な情報が含まれています。

### 11.1.3 マニフェスト・データとパッケージ・ファイル名の不整合によるパッケージ・デプロイメント・エラー

マイグレーション・パッケージ・ファイルをソース・サーバーからコンピューターにダウンロードするときに、ファイル名が予期せず変更されることがあります。これによって不整合が生じ、デプロイメントが失敗します。

ファイル名が変更されるのは、ダウンロードするファイルがダウンロード場所に既に存在することをブラウザーが認識したときに、ブラウザーの設定によってファイルの名前が変更される場合です。これは、メタデータで最もよく発生します。

この状況に該当するかどうかを調べるには、ダウンロード場所にナビゲートし、パッケージのファイル名をモニタします。次に、パッケージを開き、マニフェストを抽出します。マニフェスト内のパッケージの名前が zip ファイルの名前と一致していない場合は、パッケージ zip ファイルの名前を、マニフェスト内の情報と一致するように変更します。

#### 11.1.4 オブジェクト構造の誤った組み合わせによるパッケージ・ディプロイメント・エラー

あるマイグレーション・グループを、依存関係がある別のマイグレーション・グループに依存関係を使用せずに関連付けると、次のエラーが発生します。このエラーが発生するのは、依存関係が、そのデータに依存する次のグループの前に処理されてデータベースにコミットされるためです。

これは、ユーザーにとって難しい問題です。従属データ・グループのサイズが大きいが、大きいデータ・マイグレーション・グループの小さな部分だけに依存している場合、マイグレーション・グループまたはパッケージに依存関係を設定するとパフォーマンスに影響します。これは、従属データ・グループをフィルタリングする方法がないためです。したがって、この問題を軽減するには 2 つの選択肢があります。

1 つは、問題となる特定のオブジェクト構造のみを含んだマイグレーション・パッケージを作成する方法です。次に、パッケージをフィルタリングして、特定の小さなレコード・セットを抽出します。このパッケージをマイグレーションした後で、依存関係なしでオリジナル・パッケージをマイグレーションします。元のセットが依存しているデータは処理済みのため、問題は発生しません。

もう 1 つは、このまま依存関係を作成し、データを処理する方法です。これは時間のかかる方法であり、依存関係のデータの量によっては、マイグレーションの処理に長時間を要する可能性があります。

#### 11.1.5 セキュリティー構成のマイグレーション・エラー

このセクションでは、セキュリティー構成に関連するマイグレーション・エラーについて説明します。

## セキュリティー・グループとスタート・センターの関連付け

すべてのセキュリティー・グループは、対応するスタート・センターに関連付けられます。あるユーザーが複数のセキュリティー・グループに属している場合、各セキュリティー・グループに関連付けられたスタート・センターが累積されることにより、そのユーザーに複数のスタート・センターが割り当てられる可能性があります。

セキュリティー・グループをマイグレーションする場合は、関連付けられたスタート・センターが別のパッケージであらかじめマイグレーションされているか、または現在のパッケージ内でセキュリティー・グループ構成よりも前にあることを慎重に確認する必要があります。これを行わないと、以下のエラーが発生してセキュリティー・グループのマイグレーションが失敗します。

```
[エラー] java.lang.NullPointerException が  
psdi.dm.procclass.DMMaxGroupProcess.setAdditionalData で発生しました  
([ERROR] java.lang.NullPointerException at  
psdi.dm.procclass.DMMaxGroupProcess.setAdditionalData)  
(DMMaxGroupProcess.java:114)
```

セキュリティー・グループを含むマイグレーション・パッケージをデプロイメントする際に、マイグレーション・マネージャーは、セキュリティー・グループとそれに関連付けられたスタート・センター間のリンクを確立しようと試みます。ターゲット製品環境にはスタート・センターが存在しないため、マイグレーション・マネージャーはデプロイメント・エラーを報告します。

セキュリティー・グループのマイグレーションに対する別のアプローチとして、マイグレーションを実行する前にスタート・センターへの参照を除外する方法もあります。これを行うには、オブジェクト構造アプリケーションにアクセスして、DMMAXGROUP オブジェクト構造を表示します。「アクションの選択」メニューから、「フィールドの除外 / インクルード」アクションを実行します。表示されたポップアップ・ダイアログ・ボックスで、MAXGROUP ビジネス・オブジェクトから SCTYPEMPATEID 属性を除外します。「OK」をクリックして変更を保存し、ダイアログを閉じます。これにより、スタート・センター・テンプレートの参照がソース環境のマイグレーション・パッケージに含まれないようにすることができます。ただし、マイグレーションの完了後に、セキュリティー・グループとそれに対応するスタート・センター間のリンクを確立するための手動タスクを実行する必要があります。この重要なトレードオフは、マイグレーション計画アクティビティの一部として考慮する必要があります。

### 11.1.6 セキュリティー・グループおよびグループ再割り当てのエラー

ユーザーをマイグレーションするときに、以下のデプロイメント・エラーが報告されます。

BMXAA6695E - オブジェクト GROUPUSER の MBO をバッチ検証できませんでした。エラーは GROUPNAME  
psdi.util.MXApplicationException です。BMXAA0028E - 現在のセキュリティー権限では、選択したオプションにはアクセスできません。

ユーザーは、DMMAXUSER オブジェクト構造を使用してマイグレーションされます。ユーザーがマイグレーションされる前に、セキュリティー・グループが DMMAXGROUP オブジェクト構造を使用してマイグレーションされます。DMMAXUSER オブジェクト構造内では、GROUPUSER ビジネス・オブジェクトによってユーザーとその所属先のセキュリティー・グループが関連付けられます。ユーザーをグループに割り当てるには、特定のセキュリティー許可が必要です。1人の特定ユーザーが、選択したグループに他のユーザーを追加する許可を与えられます。

マイグレーション・マネージャーを使用してユーザーをマイグレーションする場合、マイグレーション・パッケージのデプロイメントを実行するユーザーまたは管理者は、ユーザーをセキュリティー・グループに割り当てる許可を与えられていなければなりません。許可が与えられていない場合、前述のデプロイメント・エラーが発生し、GROUPUSER MBO に対する検証ルールが失敗します。マイグレーションを実行しているユーザーが、必要な許可を与えられた管理ユーザーであることを確認してください。通常、クライアントがマイグレーションを実行するときは、MAXADMIN ユーザー・アカウントまたは MAXADMIN から複製されたアカウントを使用します。

### 11.1.7 無効または非推奨の MAXVARS の影響

マイグレーション・パッケージに DMMAXVARS オブジェクト構造と MAXVARS テーブルのデータが組み込まれている場合に、デプロイメント・エラーが発生しました。報告されたエラーは以下のとおりです。

BMXAA4116E - Maxvar タイプが無効です。

MAXVARS テーブルは、システム全体、組織レベル、またはサイト・レベルのフラグを多数保持しています。これらのフラグの一部は、組織アプリケーションの「アクションの選択」メニューから設定できます。MAXVARS テーブルのすべての項目は、対応する MAXVARTYPE テーブルの項目に関連付けられています。MAXVARTYPE テーブルの項目により、特定の MAXVAR がシステム全体、組織レベル、またはサイト・レベルのどれであるかが判別されます。MAXVARS ビジネス・オブジェクトは、MAXVAR テーブルと MAXVARTYPE テーブルの項目が一致することを確認する検証を行います。

MAXVARS および MAXVARTYPE の項目は、インストール・プログラムまたはアップグレード・プログラムによってのみ作成されます。同じ製品のインストールまたはアップグレードが実稼働環境で完了している場合、2つの製品環境間で MAXVARS が正確に一致していないかもしれません。MAXVARS ビジネ

ス・オブジェクトは、MAXVARS テーブルに MAXVAR の新しい項目を「追加」する機能を提供していません。

したがって、マイグレーション・マネージャーは、MAXVARS ビジネス・オブジェクトによる検証の対象である既存の MAXVAR 項目の更新のみをサポートしています。

いくつかの製品リリースを経る間に、MAXVARS のサブセットは非推奨になりました(製品に使用されなくなりました)。

IBM では、Tivoli's process automation engine (Tivoli/Maximo Base Service) で提供されているアップグレード・スクリプトに問題点があることを認識しています。この問題により、リリース 7.1.x にアップグレードした後に孤立または非推奨の MAXVARS 項目が製品環境に残ります。これらの項目を含む MAXVARS をマイグレーションすると、マイグレーションが失敗します。この問題点は、Maximo Base Services (MBS) Fix Pack 7.1.1.6 のリリースに伴って修正されています。詳細については、IBM サポート Web サイトの以下のリンクを参照してください。

<http://www-01.ibm.com/support/docview.wss?uid=swg1IZ61692>

クライアントが実施できる予防策は、両方の環境で以下の SQL ステートメントを実行して、一致する MAXVARS 項目の数を調べることです。

```
select count(*) from maxvars
```

以下の SQL ステートメントを実行すると、MAXVARS テーブル内の項目に対応する MAXVARTYPE テーブル内の項目が少なくとも 1 つ存在することを確認できます。

```
select count(*) from maxvars where exists (select * from maxvartype  
where maxvars.varname=maxvartype.varname)
```

これらの 2 つのステートメントから返されるレコードの数が一致している必要があります。

MAXVARS のマイグレーションを正しく実行するためのもう 1 つの考慮事項は、開発環境で行われた変更を追跡し、変更された MAXVARS のみをマイグレーションすることです。

### 11.1.8 添付ドキュメントおよびドキュメント・タイプ

クライアントが「置換」処理アクションを使用してスナップショット・パッケージを移行する際に、以下のデプロイメント・エラーが発生しました。

DMDOCTYPES の処理中にエラーが発生しました(オブジェクト構造番号 1。プライマリー・オブジェクト: DOCTYPES。キー: Attachments)。(Error occurred while processing DMDOCTYPES (Object Structure number 1. Primary Object

is: DOCTYPES. Key is: Attachments.) エラー: BMXAA0876E - このレコードは削除できません。 (Error is: BMXAA0876E - Cannot delete this record)

DMDOCTYPES オブジェクト構造は、添付ドキュメント・タイプのマイグレーションをサポートしています。DOCINFO ビジネス・オブジェクトはこのオブジェクト構造のメンバーであり、DOCTYPES ビジネス・オブジェクトの下位階層です。「置換」処理アクションを使用してスナップショット・パッケージをマイグレーションするときに、基礎となるターゲット環境データベースに項目が存在しない場合、マイグレーション・マネージャーは DOCINFO テーブルに項目を追加します。そして DOCINFO テーブル内の既存の項目を更新し、最後に、マイグレーション・パッケージに含まれている XML 文書に存在しない DOCINFO テーブルの項目を削除しようとします。

ターゲット製品環境に、DOCINFO の項目が実際のアプリケーション・レコード(サービス要求や注文書アプリケーションなど)にリンクされているような稼働中の添付ドキュメントが含まれている場合は、DOCINFO ビジネス・オブジェクトの検証ルールにより、その DOCINFO レコードの削除が阻止されます。そのため上記のエラーが発生し、スナップショット・マイグレーション・パッケージのデプロイメントが失敗します。

予防策は以下のとおりです。

- ▶ 実稼動の前に、最初の DOCTYPES のマイグレーションを実行します。これにより、アプリケーション・レコードに結合された稼働中の添付ドキュメントが存在しないようにすることができます。
- ▶ 実稼動の前後に、AddModify 処理アクションを使用してすべてのスナップショット・マイグレーションを実行します。これにより、マイグレーション・マネージャーが、ターゲット・データベースに存在するがデプロイメント時に処理される XML 文書には存在しない DOCINFO レコードの削除を試みないようにすることができます。

## 11.2 マイグレーション・パッケージ・エラーの解決方法

以下の手法を使用して、前のセクションで示したパッケージ・インストール・エラーを修正することができます。順序は重要ではなく、必要に応じて以下の手法を任意の組み合わせで使用することができます。エラーが発生した場合は、アプリケーションの「メッセージ」タブに表示されているエラーをメモしてください。

### 11.2.1 ロギング・アプリケーションの使用

「マイグレーション・マネージャー・ガイド」([http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_migration_mgr_guide.pdf))では、ロギング・アプリケーションへのアクセス方法と、エラー発生時のシステム分析に使用するログ・ファイルを作成するためのロギング・アプリケーションのセットアップ方法を簡単に説明しています。マイグレーション・マネージャーのロギング・アプリケーションをセットアップする方法については、このマニュアルの『マイグレーション・マネージャー・アプリケーション・ロガー』を参照してください。このセクションでは、ログ・ファイルの分析方法と、ログ・ファイルに含まれる情報に基づいてさまざまなアクションを実行する方法を説明します。

#### ログ・ファイルの読み取り

マイグレーションでエラーが発生した場合は、ロギング・アプリケーションにセットアップしたログ・ファイルに直ちにアクセスしてください。ログ・ファイルを開いたら末尾に移動し、逆方向にエラーを探します。エラーは 264 ページの 図 11-5 のように表示されます。通常、エラーが発生するといくつかのスタック・トレースが作成されます。これは、標準のログ・ファイル項目と見分けることができます。エラーが発生した場合は、以下の例のようなテキストを注意深く探してください。

##### 例 11-1 ログ・ファイル

---

```
[9/28/10 17:06:55:359 EDT] 00000092 SystemOut      0 28 Sep 2010
17:06:55:359 [エラー] タイプCFGDATA と構成データ順序 4 を持つパッケージ Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057 のインポートが失敗しました。
[9/28/10 17:06:55:390 EDT] 00000092 SystemOut      0 28 Sep 2010
17:06:55:390 [エラー] パッケージ
Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057 をデプロイできませんでした。詳細は、ログのエントリーを確認してください。
```

---

以下の点に注意してください。

- ▶ 第1に、「パッケージのインポートが失敗しました (Import fails for package)」というフレーズにより、パッケージ内のエラーの場所を警告するログの部分を判別できます。
- ▶ 第2に、このセンテンスには、問題のあるファイルを示すパッケージ・ファイルIDが含まれています。

```
[9/28/10 17:06:55:265 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:265 [DEBUG] ****
[9/28/10 17:06:55:265 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:265 [INFO] Management Manager deployment started for packageHierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057, type CFGDATA and configuration order 4.
[9/28/10 17:06:55:281 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:281 [ERROR] Could not deploy configuration data for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057.
psdi.util.MXApplicationException: MXAA1281E - Object Structure MY_LOCATION does not exist.
    at psdi.util.MXApplicationException.getMXAA1281E(MXUtil.java:45)
    at psdi.iface.mic.MicService.loadData(MicService.java:1506)
    at psdi.dn.pkg.DMPackage.deployStagingData(DMPackage.java:1414)
    at psdi.dn.pkg.DMPackage.deployPackage(DMPackage.java:906)
    at psdi.webclient.beans.dn.DMAppBean.deployPackage(DMAppBean.java:321)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at psdi.webclient.system.session.AsyncProcess.rui(AsyncProcess.java:95)
    at java.lang.Thread.run(Thread.java:810)
[9/28/10 17:06:55:359 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:359 [ERROR] Could not import configuration records for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057.
Please check log file for more detail.
psdi.util.MXApplicationException: MXAA1281E - Object Structure MY_LOCATION does not exist.
    at psdi.util.MXApplicationException.getMXAA1281E(MXUtil.java:45)
    at psdi.iface.mic.MicService.loadData(MicService.java:1506)
    at psdi.dn.pkg.DMPackage.deployStagingData(DMPackage.java:1414)
    at psdi.dn.pkg.DMPackage.deployPackage(DMPackage.java:906)
    at psdi.webclient.beans.dn.DMAppBean.deployPackage(DMAppBean.java:321)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at psdi.webclient.system.session.AsyncProcess.rui(AsyncProcess.java:95)
    at java.lang.Thread.run(Thread.java:810)
[9/28/10 17:06:55:359 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:359 [ERROR] Import fails for package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057 with type CFGDATA and configuration data order 4.
[9/28/10 17:06:55:390 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:390 [ERROR] Could not deploy package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057. Please check log entries for more information.
null
[9/28/10 17:06:55:406 EDT] 00000092 SystemOut 0 28 Sep 2010 17:06:55:406 [ERROR] Could not deploy package Hierarchy_Example_ISM_MAXDB71_MAXIMO_20100928170057. Please check log entries for more information.
java.lang.Exception: psdi.util.MXApplicationException: MXAA1281E - Object Structure MY_LOCATION does not exist.
    at psdi.util.MXApplicationException.getMXAA1281E(MXUtil.java:1542)
    at psdi.dn.pkg.DMPackage.deployPackage(DMPackage.java:906)
    at psdi.webclient.beans.dn.DMAppBean.deployPackage(DMAppBean.java:321)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:79)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at psdi.webclient.system.session.AsyncProcess.rui(AsyncProcess.java:95)
    at java.lang.Thread.run(Thread.java:810)
Caused by:
psdi.util.MXApplicationException: MXAA1281E - Object Structure MY_LOCATION does not exist.
    at psdi.util.MXApplicationException.getMXAA1281E(MXUtil.java:453)
    at psdi.iface.mic.MicService.loadData(MicService.java:1506)
    at psdi.dn.pkg.DMPackage.deployStagingData(DMPackage.java:1414)
    ... 10 more
[9/28/10 17:07:12:856 EDT] 00000056 SystemOut 0 28 Sep 2010 17:06:55:546 [DEBUG] Tue Sep 28 17:06:55 EDT 2010*****finally block ****
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] CLASSIFICATION: mbosets (3), mbos (5)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] CLASSSTRUCTURE: mbosets (2), mbos (4)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] CRONTASKDEF: mbosets (38), mbos (76)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] CRONTASKINSTANCE: mbosets (50), mbos (105)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] CRONTASKPARAM: mbosets (46), mbos (78)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] DRCFGGROUP: mbosets (6), mbos (7)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] DRCFGGROUPCT: mbosets (1), mbos (1)
[9/28/10 17:07:12:859 EDT] 00000056 SystemOut 0 28 Sep 2010 17:07:12:859 [INFO] DRCFGPDEFECT: mbosets (1), mbos (1)
```

図11-5 ログ・ファイルの読み取りとエラーの発見

## エラーの解釈

これでファイルの場所が判明したので、zipファイル・エディターを使用してマイグレーション・パッケージを開き、問題のあるXMLファイルを開きます。  
265ページの図11-6を参照してください。

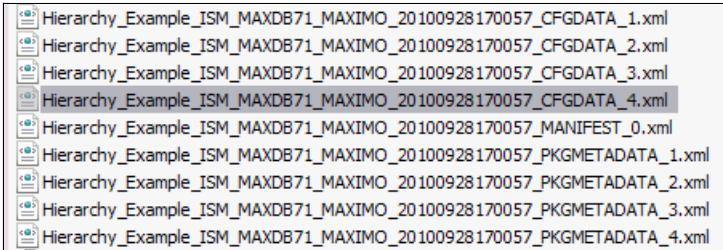


図 11-6 エラーのある構成ファイルの特定

ログ・ファイルに示されているとおり、CFGDATA の 4 番にエラーがあります。

## ソリューションの分析

任意のエディターまたはビューアーで、この XML ファイルを開きます。次の図 11-7 に、XML が Internet Explorer でどのように表示されるかを示します。

```
<?xml version="1.0" encoding="UTF-8" ?>
- <SyncMY_LOCATION creationDateTime="2010-09-28T17:02:24-04:00" transLanguage="ja-JP" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.sap.com/ewm/locations">
- <MY_LOCATIONSet>
- <LOCATIONS action="AddChange">
<AUTOWOGEN>0</AUTOWOGEN>
<BILLTOADDRESSCODE />
<BILLTOLABORCODE />
<CALNUM />
<CHANGEBY>SMITH</CHANGEBY>
<CHANGEDATE>2004-05-13T18:59:06-04:00</CHANGEDATE>
<CHILDREN>0</CHILDREN>
<CONTROLACC />
<CURVARACC />
```

図 11-7 XML ファイルの確認

この時点で、例 2 に示すように元のエラーを参照します。

### 例 11-2 元のエラー

---

BMXAA1281E - オブジェクト構造 MY\_LOCATION は存在しません。

---

XML ファイルの一番上を見ると、アクションは MY\_LOCATION オブジェクト構造の SYNC であることがわかります。

同期アクションが行われる場合、データがシステム内に存在するか、少なくとも検証参照点を持っている必要があります。

このケースでは、マイグレーション・マネージャーがオブジェクト構造自体をまだ処理していなかったために検証が失敗しました。これは、データがパッケージに組み込まれていなかったか、または処理順序が正しくなかったことを意味します。このケースでは、オブジェクト構造の定義が組み込まれていませんでした。266 ページの 図 11-8 を参照してください。

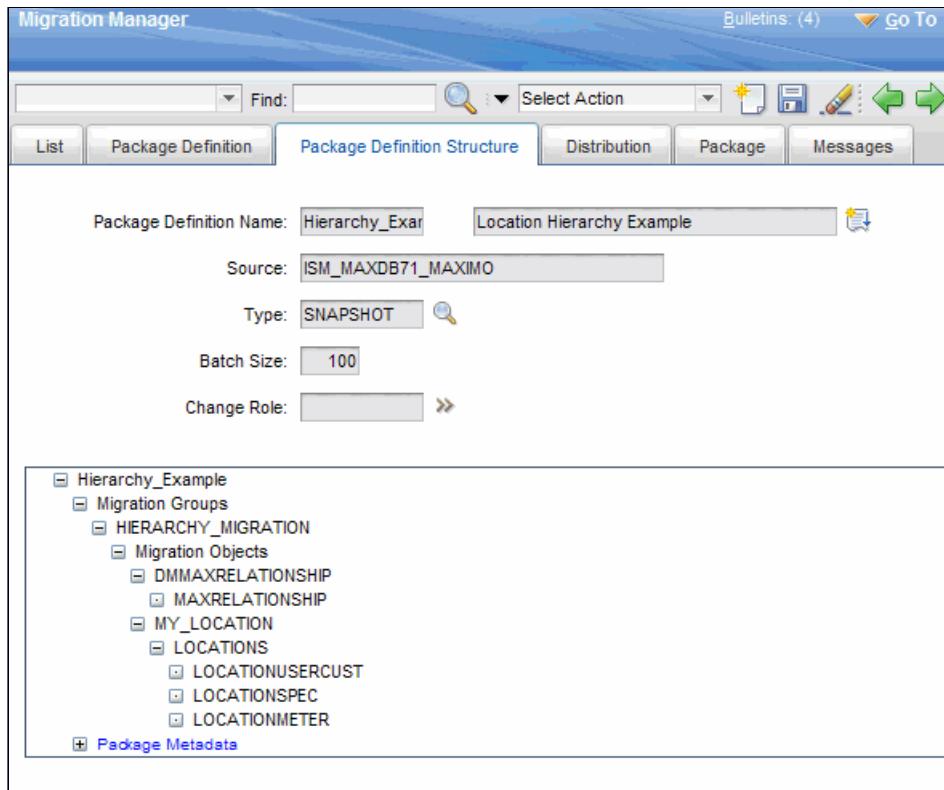


図 11-8 マイグレーション・オブジェクトの定義がないパッケージ定義構造

この状況は、新しいオブジェクト構造が作成され、データ・ディクショナリーのマイグレーションの一部としてマイグレーションされていない場合に発生します。これを修正するには、次の 2 つの方法のいずれかを使用します。

### ソリューションの適用

1 つは、データ・ディクショナリーのマイグレーション・グループに対する依存関係を作成する方法です。このような非常に独立性の高いシナリオでは、この方法は時間がかかり、お勧めできません。

もう 1 つは、必要なオブジェクト構造をマイグレーション・パッケージ定義に単純に追加する方法です。この手法を使用すると、データベース全体をマイグ

レーションする必要なく個別のデータ成果物をマイグレーションすることができます。ただし、これを行うと同じエラーが発生します。マイグレーション・グループの処理方法には処理対象データの再取り出しが組み込まれていないため、パッケージのエラーが再び発生するからです。

そのソリューションは、DMMAXINTOBJ オブジェクト構造専用のパッケージを個別に作成することです。この構造のみを含むマイグレーション・グループを作成します。次に、その新しいグループのみを含むパッケージを作成します。この作業が完了したら、ソリューション・パッケージをマイグレーションする前に、このパッケージを別にマイグレーションすることができます。

このパッケージの作成方法を、次の図 11-9 に示します。

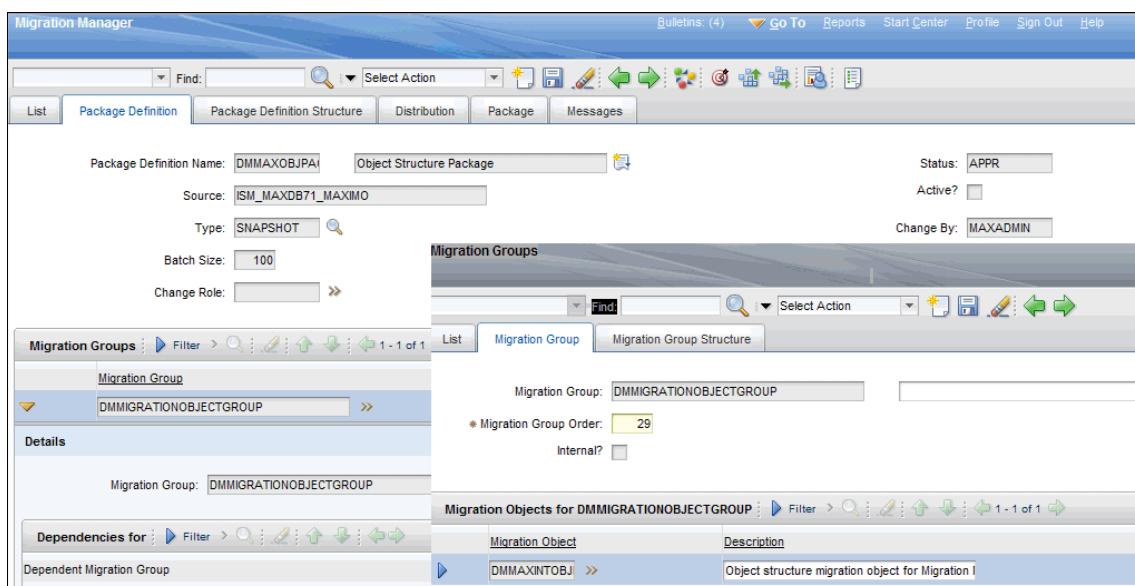


図 11-9 オブジェクト構造のマイグレーション・グループ

これは、マイグレーション・マネージャーの柔軟性および機能性と、さまざまなマイグレーション・シナリオをトラブルシューティングする場合のマイグレーション・マネージャーの使用方法を示しています。

### 11.2.2 マイグレーション・パッケージの XML の変更

254 ページの『11.1 一般的なマイグレーション・パッケージ・エラー』で、インストールの違いの問題について説明しました。マイグレーション・マネージャーは異なるインストール・パッケージ間のマイグレーションのサポートを目的としていませんが、どうしても避けられない場合もあります。この状況に対処するには、以下の操作を実行します。

1. DMMAXVAR オブジェクト構造で、データ・ディクショナリーの SQL Where 文節を変更し、欠落しているインストール・パッケージの項目をフィルター処理で除外します。他のすべてのオブジェクト構造に対してこの処理を繰り返します。
2. マニフェスト XML ファイル内でその項目に移動し、異なるパッケージ項目を削除します。
3. SQL Where 文節の変更によってパッケージからフィルタリングされない、コンテンツに関する限定情報が含まれる各種の XML ファイルを変更します。
4. 問題のある XML タグ項目がすべて除去されるまで処理を繰り返します。

## XML ファイルの特定

前のセクションの 264 ページの『図 11-5 ログ・ファイルの読み取りとエラーの発見』を参照して、エラーを含んでいる XML ファイルを判別します。

## パッケージ・マニフェスト XML の変更

まず、XML ファイルを開かなければなりません。次に、問題のある項目を削除します。このとき、XML エディターを使用するとファイルを簡単に操作できます。

## XML の直接変更

マニフェストを修正したら、XML ファイルを修正する必要があります。パッケージを再処理して、発生したエラーをメモします。ログ・ファイルを開き、問題のある項目を調べます。ログ・ファイル内で、エラー・メッセージに示されたキーを探します。図 11-10 に、キーを強調表示した項目を示します。

```
:19:46:342 [DEBUG] *****
:19:46:348 [DEBUG] Object name MAXMESSAGES
:19:46:348 [DEBUG] Operation Sync
:19:46:348 [DEBUG] Keys StatusChangeFailure~solution
:19:46:355 [DEBUG] Action is Replace
:19:46:355 [DEBUG] *****
:19:46:362 [DEBUG] Object name MAXMESSAGES
:19:46:362 [DEBUG] Operation Sync
:19:46:362 [DEBUG] Keys srvdef$shouldBeActive~pmcsrvdef
:19:46:365 [DEBUG] Action is Add
:19:46:365 [DEBUG] *****
```

図 11-10 エラーが発生したキー

次に、ログ・ファイルで XML ファイルが示されます(264 ページの 図 11-5 を参照してください)。XML ファイルを開き、キーを探します。必要に応じてエラーを修正するか、項目を削除します。ファイルを保存し、zip ファイル・パッケージに戻します。

**パッケージが正常にマイグレーションされるまで処理を繰り返します。**

パッケージを再処理します。

**注:** これは時間のかかる単調な処理です。辛抱強く実行してください。

すべてのパッケージ・メタデータを簡単に識別できるわけではありません。データ検証処理は MAXIMO ビジネス・オブジェクト (MBO) によって管理されます。したがって、すべてのパッケージが処理されるまで、パッケージを順番に辛抱強く処理しなければなりません。これにより、実稼動環境に移行できる「クリーンな」環境が得られます。

## 11.3 マイグレーション・パッケージ・エラーの防止方法

このセクションでは、これまでに紹介した知識を利用して、マイグレーション・パッケージ・エラーを防止する方法を説明します。この資料でこれまでに蓄積した知識を使いこなすことで、マイグレーションに成功されることを願っています。

### 11.3.1 同一の環境での開始

可能な限り、常に同一の環境を使用してください。製品の観点からは、異なる環境間のマイグレーションを行わないことを強くお勧めします。このことは、2 次「コンテンツ」のインストールに特に当てはまります。

### 11.3.2 マイグレーション・グループとオブジェクト構造の関係の理解

製品に用意されているオブジェクト構造は徹底的にテストされており、設計どおりに機能します。しかし、本書のシナリオで示したように、データの一部だけをマイグレーションして、すべてのパッケージを一度にマイグレーションしないことはよくあります。

つまり、すべての依存関係で構成されているオブジェクト構造全体を含む完全なパッケージが IBM によって提供されているからといって、すべての依存関係を使用してすべてのパッケージを処理しなければならないわけではありません。

このことを念頭に置いて、パッケージの作成は、従属データを別のパッケージで最初にマイグレーションできることを理解したうえで行ってください。これを行った場合(そして事前にマイグレーションした構成を変更していない場合)、そのデータを再マイグレーションする必要はありません。再マイグレー

ションが必要なのは、そのマイグレーション・グループを依存関係として組み込む場合です。

この点を考慮すると、エラーを防止するだけでなく、マイグレーションの速度を大幅に向向上することができます。

### 11.3.3 個別のパッケージの独立した操作

マイグレーション・グループ間の構造上の関係を理解する方法について上で説明しました。この理解に基づいて、構成データの小さな部分を迅速かつ正確にマイグレーションできるように、順次処理を必要とするさまざまなグループごとに個別のパッケージを作成することができます。

### 11.3.4 パッケージ・グループの SQL Where 文節の使用

構成の変更をマイグレーションするために必要な各種パッケージをすべて作成した場合、最後に行う手順は、マイグレーションする構成の変更のみにパッケージのコンテンツを限定するように SQL Where 文節を変更することです。

SQL Where 文節を適切に使用することにより、必要なデータのみをマイグレーションし、従属データ・グループが適切な検証のために必要なデータを持つようになります。図 11-11.

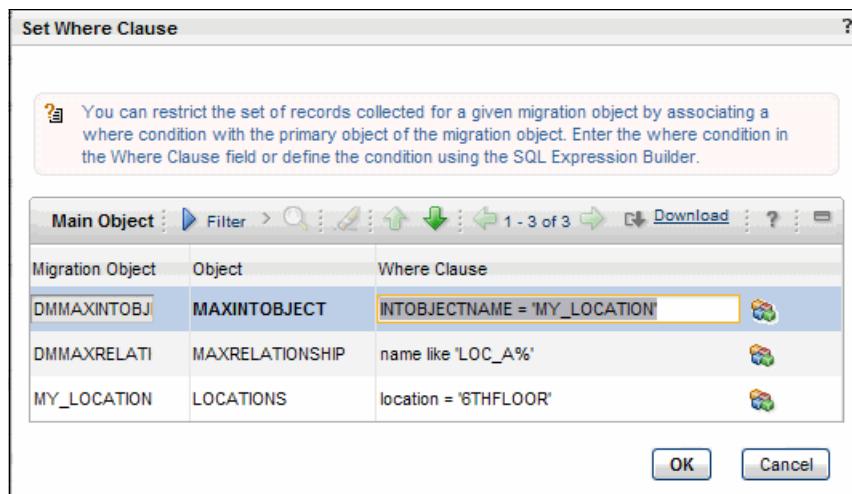


図 11-11 SQL Where 文節の例



A

## 追加資料

本書で参照している追加資料は、ここに示す方法でインターネットからダウンロードできます。

### Web 資料の場所

本書に関連した Web 資料は、インターネット上の IBM Redbooks Web サーバーからソフトコピーで入手できます。Web ブラウザーで、以下にアクセスしてください。

<ftp://www.redbooks.ibm.com/redbooks/SG247906>

以下の IBM Redbooks Web サイトにアクセスすることもできます。

[ibm.com/redbooks](http://ibm.com/redbooks)

「**追加資料 (Additional materials)**」を選択し、IBM Redbooks 資料番号 SG247906 に対応するディレクトリーを開いてください。

# Web 資料の使用

本書に付属する追加の Web 資料には、以下のファイルが収録されています。

ファイル名	説明
<b>SG247906.zip</b>	マイグレーション・サンプルを収録した圧縮ファイル

## Web 資料のダウンロードに関するシステム要件

推奨されるシステム構成は以下のとおりです。

ハード・ディスク・スペース :10 MB 以上  
オペレーティング・システム :Windows/Linux/Unix

## Web 資料の使用法

ワークステーションにサブディレクトリー(フォルダー)を作成し、Web 資料の zip ファイルの内容をこのフォルダーに unzip します。

# 関連資料

このセクションにリストされている資料は、この Redbook で扱われているトピックに関する詳細を学ぶのに、特に適していると思われる資料です。

## オンライン資料

以下は英語のみの対応となります。以下の Web サイトも、追加の情報源として役立ちます。

- ▶ IBM Maximo Asset Management 7.1、IBM Tivoli Asset Management for IT 7.1、IBM Tivoli Change and Configuration Management Database 7.1.1、IBM Tivoli Service Request Manager 7.1 アプリケーション開発者ガイド  
[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_app\\_dev\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_app_dev_guide.pdf)
- ▶ マイグレーション・マネージャー・ガイド  
[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc\\_7.1/pdf/mam71\\_migration\\_mgr\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.mam.doc_7.1/pdf/mam71_migration_mgr_guide.pdf)
- ▶ IBM Maximo Asset Management、IBM Tivoli Asset Management for IT、IBM Tivoli Service Request Manager システム管理者ガイド  
[http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc\\_7.1/pdf/mam71\\_sys\\_admin\\_guide.pdf](http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/topic/com.ibm.tamit.doc_7.1/pdf/mam71_sys_admin_guide.pdf)
- ▶ データ・ロード・ツールおよび機能のベスト・プラクティスに関する資料  
<http://www.ibm.com/developerworks/wikis/download/attachments/130515354/TpaeEcosystemDataIntegrationBestPractices.pdf?version=2>
- ▶ 添付ドキュメントに関するオブジェクト構造のサポート  
<http://www-01.ibm.com/support/docview.wss?uid=swg24027858>

## Redbooks の入手方法

以下は英語のみの対応となります。以下の Web サイトでは、Redbooks、Redpapers、技術情報、ドラフト文書、およびその他の追加資料の検索、閲覧、またはダウンロードができるほか、ハードコピー版の Redbooks を注文できます。

[ibm.com/redbooks](http://ibm.com/redbooks)

## IBM からの支援

IBM サポート & ダウンロード

[ibm.com/support](http://ibm.com/support)

IBM グローバル・サービス

[ibm.com/services](http://ibm.com/services)

# 索引

## A

A xxii, 2, 29, 53, 89, 115, 150, 189, 213, 260  
APPLICATIONAUTH オブジェクト構造 80  
APPLICATIONS グループ 143  
APPLICATIONS マイグレーション・グループ 227  
APPLICATION マイグレーション・グループ 119  
APPLYSLA キー値 117  
APPSECURITY 56, 119, 217  
  複製 57  
  マイグレーション・グループ 66  
APPSECURITY グループ 134  
APPSECURITY マイグレーション・グループ 119

## B

BPM マイグレーション・グループ 94

## C

CATALOGNAME サービス・カタログ 191  
CatalogTemplate 202  
CONDITION テーブル 132

## D

DATADICTIONARY 55, 135, 217  
  グループ 143  
  マイグレーション・グループ 131, 227  
DMACTIONGROUP オブジェクト構造 92, 100, 160, 200  
DMACTION オブジェクト構造 92, 99, 160, 200  
DMAPPLICATIONAUTH オブジェクト構造 80  
DMAPPLICATIONAUTH マイグレーション・オブジェクト 83  
DMCLASSIFICATION 208  
DMCLASSIFICATION オブジェクト構造 167, 169, 199  
DMCOMMTEMPLATE オブジェクト構造 93, 103, 111, 160  
DMCONDITION オブジェクト構造 55, 74  
DMCONDITION マイグレーション・オブジェクト 136  
DMCRONTASKDEF オブジェクト構造 106  
DMCTRLGROUP 217  
  オブジェクト構造 66, 68

DMCTRLGROUP マイグレーション・オブジェクト 137  
DMCURRENCY オブジェクト構造 33  
DMDOCTYPES オブジェクト構造 262  
DMESCALATION オブジェクト構造 94, 100, 103, 105  
DMLANGUAGE オブジェクト構造 217  
DMMAXGROUP オブジェクト構造 56, 73, 128, 195, 199  
DMMAXGROUP マイグレーション・オブジェクト 137  
DMMAXOBJECT オブジェクト構造 81, 129, 142  
DMMAXOBJECT マイグレーション・オブジェクト 74  
DMMAXLAUNCHENTRY オブジェクト構造 195  
DMMAXLAUNCHENTRY マイグレーション・オブジェクト 136  
DMMAXMENU マイグレーション・オブジェクト 136  
DMMAXMODULES マイグレーション・オブジェクト 136  
DMMAXSERVSECURITY マイグレーション・オブジェクト 137  
DMMAXUSER オブジェクト構造 56, 58, 260  
DMMAXUSER マイグレーション・オブジェクト 137  
DMPERSON オブジェクト構造 55, 93  
DMPKGEVENTS オブジェクト構造 231  
DMROLE オブジェクト構造 93, 103, 160  
DMSCTEMPLATE マイグレーション・オブジェクト 136  
DMSECURITYRESTRICT オブジェクト構造 74, 81  
DMSECURITYRESTRICT オブジェクト構造 73, 77  
DMSIGOPTFLAG オブジェクト構造 196  
DMSIGOPTFLAG マイグレーション・オブジェクト 137  
DMSIGOPTION オブジェクト構造 60, 66, 68, 69, 195  
DMSIGOPTION マイグレーション・オブジェクト 137  
DMTKTEMPLATE オブジェクト構造 198  
DMWFPROCESS オブジェクト構造 197, 160  
DOCINFO テーブル 111

DOCINFO レコード 262

## E

E xxiii  
EAR ファイル、作成 112  
ESCALATION クーロン・タスク 106, 107

## G

GL 勘定科目構成 54

## H

HIERARCHYPATH 173  
HIERARCHYPATH 非永続フィールド 173

## I

ISMACCEPT プロセス 158

## J

Java 仮想マシン (JVM) 25  
Java デプロイメント・エンタープライズ・アーカイブ (EAR) 112

## L

LANGUAGE テーブル 217  
Lightweight Directory Access Protocol (LDAP) 78  
LONGDESCRIPTION テーブル 217  
L\_ 複数言語テーブル 216

## M

Maximo Asset Management 214  
Maximo EAR 214  
MAXIMO ビジネス・オブジェクト (MBO) 269  
MAXOBJECTCFG 47, 135, 230  
MAXPRESENTATION テーブル 129  
MAXVARS テーブル 257, 260  
MAXVARTYPE テーブル 260  
MIGRATIONMGR 141  
MXOPERLOC オブジェクト構造 237  
MYESCALATION マイグレーション・グループ 96  
MYMAXAPPS マイグレーション・オブジェクト 136  
MYMAXPRESENTATION オブジェクト構造 130  
MYQUERY オブジェクト構造 142

## P

PMSC\_ASSETATTRIBUTE オブジェクト構造 200  
PMSC\_AUTOSCRIPT オブジェクト構造 200  
PMSC\_COMMODITIES オブジェクト構造 197  
PMSC\_DOCINFO 182  
PMSC\_DOCINFO オブジェクト構造 184, 199  
PMSC\_JOBPLAN 182  
PMSC\_JOBPLAN オブジェクト構造 182, 196  
PMSC\_JACTION オブジェクト構造 196  
PMSC\_OFFERING 182  
PMSC\_OFFERING オブジェクト構造 198, 200  
PMSC\_Offering オブジェクト構造 183  
PMSC\_OFFERING マイグレーション・グループ 184  
PMSC72\_CatalogOffering 207  
PMSC72\_CatalogTemplate 193, 208  
PMSC72\_CatalogTemplate パッケージ定義 191  
PMSC72\_CATALOG マイグレーション・グループ 191, 193, 201  
PMSC72\_OFFERING マイグレーション・グループ 191, 193, 201  
PMSCCATALOG オブジェクト構造 195

## R

R 214  
RBAUTHORIZATIONS01 マイグレーション・グループ 81  
RBGLOBALSE01 マイグレーション・グループ 76  
RBSECPKG01 パッケージ 58  
RBSECURITY01 オブジェクト構造 57  
Redbooks Web サイト 273  
お問い合わせ (Contact us) xxv  
RESOURCES マイグレーション・グループ 94

## S

SCTEMPLATEID 属性 259  
SEARCHWHERE キー値 117  
SECURITYRESTRICT オブジェクト構造 73  
SECURITYRESTRICT テーブル 73, 128, 129  
sigoptflag 203  
SIGOPTION 62  
SQL Where 文節 270  
SQL Where 文節コマンド 225  
SQL データ・ディクショナリー 268  
SQL フィルター基準 156  
STATUS キー値 117

## T

Tivoli Service Request Manager、オブジェクト構造  
182

## U

URL の埋め込み 222

## W

Web サービス 213  
Where 文節の設定 145, 146, 157  
WORKORDER オブジェクト 54

## あ

アクション 159, 182  
アプリケーション・デザイナー 64, 116, 118  
アプリケーションの作成 64  
アベンダーの管理 246

## い

依存関係にあるグループ 143  
イベント・トラッキング・レコードの表示 229  
イベント・トラッキング・レコードのリセット  
221

## う

埋め込み URL 222

## え

エスカレーション・ポイント 110  
エスカレーション、アクセス 90

## お

応答計画 208  
オファリング 182  
オブジェクト構造 6, 32, 55, 92, 118, 150, 175,  
191, 213, 258  
APPLICATIONAUTH 80  
DMACTION 92, 200  
DMACTIONGROUP 92, 200  
DMAPPLICATIONAUTH 80  
DMCOMMTEMPLATE 93  
DMCONDITION 55  
DMCRONTASKDEF 106  
DMCTRLGROUP 66, 68  
DMESCALATION 94  
DMMAXGROUP 56, 73, 128

## DMMAXINTOBJECT 81, 129

DMMAXUSER 56, 58  
DMPERSON 55, 93  
DMROLE 93  
DMSECURITYRESTRICT 81  
DMSIGOPTION 60, 66, 68  
DMWFPROCESS 197  
PMSC\_ASSETATTRIBUTE 200  
PMSC\_AUTOSCRIPT 200  
PMSC\_COMMODITIES 197  
PMSC\_DOCINFO 184  
PMSC\_JOBPLAN 182, 196  
PMSC\_JACTION 196  
PMSC\_Offering 183  
PMSCATALOG 195  
RBSECURITY01 57  
SECURITYRESTRICT 73  
アクセス 141  
作成 73, 80, 118, 128  
システム・ロックアップ 129  
標準 128, 141, 155  
複製 129, 169, 240  
マイグレーション 130, 241  
マイグレーション・マネージャー 216  
オブジェクト構造の複製 240

## か

階層データのマイグレーション 236  
カタログ 194  
管理モードをオン 146

## く

クエリー、作成 141  
クエリーの保存 141  
グループ xxvi, 7, 36, 51, 89, 119, 150, 168, 177,  
191, 213, 259  
新規グループの追加 55  
マイグレーション 56  
グローバル・セキュリティー制限 78  
グローバルなセキュリティー制限  
マイグレーション 128  
グローバルなデータ制限 70, 72  
クーロン・タスク 91, 104

## こ

構成アイテム (CI) 166  
故障レポート 61

コントロールのプロパティー 64

## さ

作業指示書 65  
作業指示書管理 116  
作業標準 182, 207  
サービス・カタログ 189  
サービス・グループ 182  
サービスの呼び出し 212

## し

式ビルダー 187  
「システム情報」画面 256  
システム・ルックアップ 129  
自動化スクリプト 182  
循環関係 208  
上位階層分類 208  
条件 28, 30, 52, 121, 162, 188, 230  
  管理 72  
条件式 53, 132  
条件式マネージャー 53, 63, 72  
条件付きプロパティーの構成 65  
署名オプション 65  
  追加または変更 64  
新規スナップショット・パッケージ 145  
新規テンプレートの作成 / テンプレートの変更  
(Create New/Modify Template) 141

## す

すぐに使用可能なマイグレーション・パッケージ・  
テンプレート 192  
スタート・センター 140  
スナップショット・パッケージ 224  
スナップショット・マイグレーション・  
  パッケージ 218

## せ

セキュリティー・グループ 63, 194  
  作成 53  
セキュリティー構成のマイグレーション・エラー  
  258

## そ

ソース・オブジェクト 80

## た

担当者グループ、アクセス 90  
担当者レコード、アクセス 90

## ち

チケット・テンプレート 182

## つ

通信テンプレート 101, 153, 159  
通信テンプレート、アクセス 90  
通知 153

## て

テスト結果 140  
データ制限 54  
データ・ディクショナリー 23, 30, 138, 150, 207,  
  266  
  SQL 268  
データ・ディクショナリーのコンテンツ 142  
電子メール・アカウント作成サービス・  
  オファリング 189

## と

統合フレームワーク (MEA) 213, 214  
ドキュメントの表示 182

## ね

ネストされた作業標準 207

## は

パッケージ 10, 30, 59, 111, 126, 175, 191, 215,  
  256  
  RBSECPKG01 58  
パッケージ・デプロイメント・エラー 258

## ひ

非推奨の MAXVARS 260  
標準オブジェクト構造 141  
標準的なオブジェクト構造 155

## ふ

複数の XML ファイル 213  
分類 167, 182  
分類データ、マイグレーション 171

## へ

変更管理 212  
変更パッケージ 212, 224  
変更パッケージのイベント・トラッキング 41  
変更マイグレーション・パッケージ 218

## ほ

報告日 117

## ま

マイグレーション・パッケージ・エラー 263  
マイグレーション xxi, 3, 29, 51, 89, 115, 150,  
167, 177, 189, 211, 253, 272  
グループ 56  
マイグレーション・オブジェクト構造 141  
マイグレーション・オブジェクト構造(Migration  
Object Structures) アプリケーション 143  
マイグレーション・グループ 67, 82  
マイグレーション計画文書 24  
マイグレーション順序の値 206  
マイグレーション・パッケージ 192  
マイグレーション・マネージャー 67, 75, 112,  
177, 185, 215, 220  
マイグレーション、DOCTYPE の 262

## む

無効な MAXVARS 260

## め

メタデータ 207

## ゆ

ユーザー、作成および管理 55

## る

ルート・ロガーの管理 245  
ルート・ロガーへのアクセス 248

## れ

レポート・オブジェクト構造セキュリティーの  
設定 232  
レポートの作成 233  
レポートの実行 234  
レポートの保存 233

## ろ

ロギング・アプリケーション 263  
ログ・ファイルの読み取り 263  
ロギング・ルート・フォルダーの設定 246  
ログ・ファイル、読み取り 263  
ロール 154, 159  
ロール、アクセス 90

## わ

ワークフロー・デザイナー 151, 159, 182  
ワークフローの変更 150



本の背幅を決定する際には、本のページ数を用紙のPPIで除算します。例えば、Plainfield不透明50#平滑化を使用した250ページの本のPPIが526の場合、250を526で除算して得られた0.4752インチが背幅となります。この場合は、0.5インチの背表紙を使用します。次に、本の背幅を選択し、それ以外を非表示にします（「特殊(Special)」>「条件付きテキスト(Conditional Text)」>「表示/非表示>Show/Hide」>「背景紙サイズ(SpineSize)」（=>「非表示(Hide)」）>「設定(Set)」）。変更した「条件付きテキスト(Conditional Text)」設定を選択し、「条件付きテキスト(Conditional Text)」設定のみを書籍ファイルに適用します。これを行うには、spine.fmがまだ開いている状態で書籍ファイルを開き、「ファイル」>「インポート(Imports)」>「レビュー用ドRAFT資料 2010年12月16日午前11:00」

7906spine.fm 265



# マイグレーシヨン・マネージャーを使用したマイグレーションのユースケース



## マイグレーシヨン・マネージャーを使用したマイグレーションのユースケース

(1.5インチの背表紙)  
1.5" <-> 1.998"  
789 <-> 1051 ページ



## マイグレーシヨン・マネージャーを使用したマイグレーションのユースケース

(1.0インチの背表紙)  
0.875" <-> 1.498"  
460 <-> 788 ページ



## マイグレーシヨン・マネージャーを使用したマイグレーションのユースケース

(0.5インチの背表紙)  
0.475" <-> 0.875"  
90 <-> 249 ページ  
250 <-> 459 ページ

(0.1インチの背表紙)  
0.1" <-> 0.169"  
53 <-> 89 ページ



Redbooks

マイグレーショントラック  
マイグレーショントラック

(2.0インチの背表紙)  
2.0" <-> 2.498"  
1052 <-> 1314 ページ



Redbooks

マイグレーショントラック  
マイグレーショントラック

(2.5インチの背表紙)  
2.5" <-> 2.498"  
1315 <-> 1314 ページ

本の背幅を決定する際には、本のページ数を用紙の PPI で除算します。例えば、Plainfield 不透明 50# 平滑化を使用した 250 ページの本の PPI が 526 の場合、250 を 526 で除算して得られた 0.4752 インチが背幅となります。この場合は、0.5 インチの背表紙を使用します。次に、本の背幅を選択し、それ以外を非表示にします（「特殊 (Special)」>「条件付きテキスト (Conditional Text)」>「表示 / 非表示 (Show/Hide)」>「書表紙サイズ (SpineSize)」(>>「非表示 (Hide)」)>「設定 (Set)」）。変更した「条件付きテキスト (Conditional Text)」設定を選択し、「条件付きテキスト (Conditional Text)」設定のみを書表紙ファイルに適用します。これを行うには、spine.fm がまだ開いている状態で書表紙ファイルを開き、「ファイル」>「インポート」>「フォーマット (Formats)」>「ドロップ (Drop)」>「条件付きテキスト (Conditional Text)」>「選択 (Select)」>「適用 (Apply)」>「OK」をクリックします。

7906spine.fm 266





# マイグレーション・マネージャーを使用したマイグレーションのユースケース

Tivoli Service Request Manager、CCMDB、Tivoli's process automation engine (Tivoli/Maximo Base Service) のマイグレーション



Redbooks®

## マイグレーション・マネージャーの実装戦略を学ぶ

## 実践的なシナリオを試す

## マイグレーションに関するトラブル シューティングのヒントを学ぶ

マイグレーション・マネージャーを利用すると、ある製品環境から別の製品環境に構成コンテンツをマイグレーションできます。一般的な利用法として、Tivoli's process automation engine (Tivoli/Maximo Base Service) とそのアプリケーション (IBM Tivoli Change and Configuration Management Database (CCMDB)、IBM Tivoli Service Request Manager など) の構成コンテンツを、開発環境からテスト環境、さらに実稼働環境にマイグレーションすることができます。マイグレーションの目的は、実稼働環境がエンド・ユーザーのニーズを完全に満たせるようにすることです。

この IBM Redbooks 資料では、マイグレーション・マネージャーを使用した最も一般的なマイグレーション・ユースケースを扱います。もちろん、これらのユースケースは、マイグレーション・マネージャーで実現できるマイグレーション・シナリオのごく一部に過ぎませんが、マイグレーション・マネージャーの代表的な機能を示す目的で選ばれています。

本書ではこれらのユースケースに加え、マイグレーション戦略を紹介するとともに、マイグレーション・マネージャーの使用時に発生する可能性のあるマイグレーション問題のトラブル シューティングを包括的に扱う章を用意しています。

他の章を読む前に、まず『第1章 マイグレーション戦略』を読むことを強くお勧めします。この章を読むことで、本書で扱うすべてのマイグレーション・シナリオの理解に役立つ基礎知識が得られます。

IT スペシャリストや IT 設計者がマイグレーション・マネージャーを使用して、ある製品環境から別の製品環境に構成コンテンツをマイグレーションする際には、本書を参照してください。

INTERNATIONAL  
TECHNICAL  
SUPPORT  
ORGANIZATION

実務経験に基づく  
技術情報の構築

IBM Redbooks は、IBM International Technical Support Organization により開発されています。世界中の IBM の専門家、お客様、およびパートナーが、現実的なシナリオに基づいたタイムリーな技術情報を作成しています。具体的な推奨事項が提供されているため、ご使用の環境でより効率よく IT ソリューションを実装することができます。

詳しくは次にアクセスしてください。  
[ibm.com/redbooks](http://ibm.com/redbooks)