







**WebSphere Process Server の管理**

お願い

本書に記載されている情報をご使用になる前に、本書末尾の特記事項セクションに記載されている情報をお読みください。

新しい版で明記されるまで、WebSphere® Process Server for Multiplatforms バージョン 6、リリース 2、モディフィケーション 0 (製品番号 5724-L01) 以降のすべてのリリースとモディフィケーションが本書の対象となります。

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原典： WebSphere® Process Server for Multiplatforms  
Version 6.2.0  
Administering WebSphere Process Server

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

第1刷 2009.1

© Copyright International Business Machines Corporation 2005, 2008.

---

## PDF ブックおよびインフォメーション・センター

PDF ブックは、印刷およびオフラインでの参照用に提供されています。最新情報は、オンラインのインフォメーション・センターを参照してください。

セットとして、PDF ブックには、インフォメーション・センターと同一の内容が含まれます。

PDF 資料は、バージョン 6.0 またはバージョン 6.1 など、インフォメーション・センターのメジャー・リリースの後の四半期以内にご利用いただけます。

PDF 資料の更新頻度は、インフォメーション・センターより低いですが、Redbooks® よりも頻繁に更新されます。通常、PDF ブックはブックに十分な変更が累積されたときに更新されます。

PDF ブックの外部にあるトピックへのリンクを選択すると、Web 上のインフォメーション・センターに移動します。PDF ブックの外部にあるターゲットへのリンクには、そのターゲットが PDF ブックと Web ページのどちらなのかを示すアイコンによるマークが付いています。

表 1. 本書の外部にあるトピックへのリンクのプレフィックスとなるアイコン

アイコン	説明
	<p data-bbox="540 254 1325 281">インフォメーション・センターのページを含む、Web ページへのリンク。</p> <p data-bbox="540 310 1427 407">インフォメーション・センターへのリンクは、ターゲット・トピックが新しい場所に移動した場合でもその機能を保つように、間接参照ルーティング・サービスを経由します。</p> <p data-bbox="540 436 1427 604">ローカルのインフォメーション・センターでリンク先ページを見つけたい場合は、リンクのタイトルを検索することができます。あるいは、トピック ID を検索することもできます。検索の結果、タイプが異なる製品についてのトピックがいくつか見つかった場合は、検索結果の「グループ別 (Group by)」コントロールを使用して、表示するトピック・インスタンスを識別できます。以下に例を示します。</p> <ol data-bbox="540 619 1427 905" style="list-style-type: none"> <li>1. リンク URL をコピーします。例えば、リンクを右クリックして「リンク先をコピーする (Copy link location)」を選択します。例: <code>http://www14.software.ibm.com/webapp/wsbroker/redirect?version=wbpm620&amp;product=wesb-dist&amp;topic=tins_apply_service</code></li> <li>2. <code>&amp;topic=</code> の後のトピック ID をコピーします。例: <code>tins_apply_service</code></li> <li>3. ローカル・インフォメーション・センターの検索フィールドに、トピック ID を貼り付けます。文書機能がローカルにインストールされている場合は、検索結果にそのトピックが表示されます。以下に例を示します。</li> </ol> <div data-bbox="581 911 1427 1108" style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p data-bbox="589 926 818 947">1 result(s) found for</p> <p data-bbox="589 974 1102 1020">Group by: None   Platform   Version   Product Show Summary</p> <p data-bbox="589 1045 1346 1092">Update Installer を使用したフィックスパックおよびリフレッシュ・パックのインストール</p> </div> <ol data-bbox="540 1142 1183 1169" style="list-style-type: none"> <li>4. 検索結果のリンクをクリックしてトピックを表示します。</li> </ol>
	<p data-bbox="540 1184 799 1211">PDF ブックへのリンク。</p>

# 目次

PDF ブックおよびインフォメーション・センター . . . . .	iii
------------------------------------	-----

図 . . . . .	ix
-------------	----

表 . . . . .	xi
-------------	----

## 第 1 章 WebSphere Process Server の管理の概要 . . . . . 1

管理コンソール . . . . .	1
管理コンソールの領域 . . . . .	1
管理コンソールのビジネス・インテグレーション領域 . . . . .	3
管理コンソールのガイド付きアクティビティ . . . . .	4
管理コンソール・ページ . . . . .	5
管理コンソールのボタン . . . . .	6
コマンド行ツール、スクリプト、およびプログラミング・インターフェース . . . . .	8
Business Process Choreographer Explorer の概要 . . . . .	9
ビジネス・ルール・マネージャー . . . . .	10
構成情報 . . . . .	10

## 第 2 章 管理インターフェース入門 . . . . . 13

管理コンソール入門 . . . . .	13
管理コンソールの始動と停止 . . . . .	14
管理コンソール設定の指定 . . . . .	16
管理コンソール・フィルターの設定 . . . . .	17
「マイ・タスク」の使用法 . . . . .	17
管理コンソールからの製品情報とヘルプへのアクセス . . . . .	18
管理コンソールからコマンド・アシスタンスにアクセスする操作 . . . . .	20
Business Process Choreographer Explorer 入門 . . . . .	22
Business Process Choreographer Explorer のユーザー・インターフェース . . . . .	23
Business Process Choreographer Explorer の開始 . . . . .	32
Business Process Choreographer Explorer のカスタマイズ . . . . .	33
Business Process Choreographer Observer 入門 . . . . .	43

## 第 3 章 サーバーおよびクラスターの管理 49

サーバーの作成 . . . . .	49
サーバーおよびクラスターの開始または停止 . . . . .	49
スタンドアロン・サーバーの開始 . . . . .	50
スタンドアロン・サーバーの停止 . . . . .	51
デプロイメント・マネージャーの始動 . . . . .	51
デプロイメント・マネージャーの停止 . . . . .	52
ノード・エージェントの開始 . . . . .	53
ノード・エージェントの停止 . . . . .	54
ノード・エージェントの再開 . . . . .	54

デプロイメント環境の開始と停止 . . . . .	54
コマンド行を使用してデプロイメント環境を開始する操作 . . . . .	55
コマンド行を使用してデプロイメント環境を停止する操作 . . . . .	56
クラスターの開始 . . . . .	58
クラスターの停止 . . . . .	58
サーバーの開始 . . . . .	58
サーバーの停止 . . . . .	60

## 第 4 章 デプロイメント環境の管理 . . . . . 63

カスタム・デプロイメント環境のレイアウト構成 . . . . .	64
デプロイメント環境の作成 . . . . .	70
デプロイメント・トポロジーの表示 . . . . .	72
デプロイメント環境の編集 . . . . .	73
データ・ソース構成の編集 . . . . .	75
データベース・プロバイダーの編集 . . . . .	76
デプロイメント環境内のデータ・ソースの編集 . . . . .	77
デプロイメント・マネージャーの停止と再始動 . . . . .	77
クラスター・メンバーの停止および再始動 . . . . .	78
デプロイメント環境の開始と停止 . . . . .	79
管理コンソールを使用したデプロイメント環境定義のエクスポート . . . . .	80
コマンド行を使用したデプロイメント環境定義のエクスポート . . . . .	82
管理コンソールを使用したデプロイメント環境定義のインポート . . . . .	83
コマンド行を使用したデプロイメント環境定義のインポート . . . . .	86
デプロイメント環境の除去 . . . . .	87

## 第 5 章 アプリケーションおよびアプリケーション・サービスの管理 . . . . . 89

サービス・アプリケーションおよびサービス・モジュールの管理 . . . . .	89
サービス・アプリケーションのバージョン管理 . . . . .	90
管理インターフェースのサービス・アプリケーション機能 . . . . .	91
サービス・モジュールの管理 . . . . .	93
エンタープライズ・アプリケーションの管理 . . . . .	103
SCA 要求のスルーputの管理 . . . . .	105
サービス・アプリケーションおよびサービス・モジュールの追加処理 . . . . .	109
デプロイされた SCA モジュールの WebSphere MQ JMS 宛先の変更 . . . . .	125
Service Component Architecture モジュールと WebSphere MQ . . . . .	126
ターゲットの使用 . . . . .	128
インポート・ターゲットの変更 . . . . .	128
J2C アクティベーション・スペックの削除 . . . . .	130
SIBus 宛先の削除 . . . . .	131

エンタープライズ・アプリケーションの管理 . . . . .	132
Application Scheduler の管理 . . . . .	134
Application Scheduler へのアクセス . . . . .	134
Application Scheduler MBean インターフェース を使用した Application Scheduler へのアクセス . . . . .	135
管理コンソールを使用したスケジューラー・エン トリーの表示 . . . . .	136
スケジュール済みイベントの作成 . . . . .	137
スケジュール済みイベントの変更 . . . . .	139
スケジュール済みイベントの削除 . . . . .	140
リレーションシップの管理 . . . . .	140
リレーションシップの表示 . . . . .	141
リレーションシップ詳細の表示 . . . . .	142
ロール詳細の表示 . . . . .	143
リレーションシップの照会 . . . . .	143
リレーションシップ・インスタンスの表示 . . . . .	147
リレーションシップ・インスタンス詳細の表示 . . . . .	148
リレーションシップ・インスタンスの詳細の編集 . . . . .	149
リレーションシップ・インスタンスの新規作成 . . . . .	150
リレーションシップ・インスタンスの削除 . . . . .	151
リレーションシップ・インスタンス・データのロ ールバック . . . . .	151
ロール・インスタンス詳細の表示 . . . . .	152
ロール・インスタンス・プロパティの編集 . . . . .	153
ロール・インスタンスの新規作成 . . . . .	154
ロール・インスタンスの削除 . . . . .	155
リポジトリからのリレーションシップ・インス タンス・データの除去 . . . . .	155
チュートリアル: リレーションシップ・マネー ジャーの管理 . . . . .	158
リレーションシップ・サービスの管理 . . . . .	160
リレーションシップ・サービスで管理されるリ レーションシップの表示 . . . . .	162
リレーションシップ・プロパティの表示 . . . . .	162

**第 6 章 Business Process  
Choreographer の管理 . . . . . 165**

**第 7 章 Common Event  
Infrastructure の構成および管理 . . . . . 167**

**第 8 章 サービス・コンポーネントの管  
理 . . . . . 169**

ビジネス・ステート・マシンの管理 . . . . .	169
ビジネス・ステート・マシンのインスタンスの検 索 . . . . .	170
表示状態の表示 . . . . .	170
ビジネス・ルールおよびセレクターの管理 . . . . .	171
ビジネス・ルールおよびセレクターを含むモジ ュールの考慮事項 . . . . .	171
ビジネス・ルールの概要 . . . . .	174
ビジネス・ルール・マネージャー . . . . .	177
セレクター・コンポーネントの概要 . . . . .	214

**第 9 章 バインディングの操作 . . . . . 223**

エクスポートおよびインポート・バインディングの 概要 . . . . .	225
エクスポートおよびインポート・バインディング構 成 . . . . .	229
インポートおよびエクスポートでのデータ・フォ ーマット変換 . . . . .	230
エクスポート・バインディングでの関数セレクタ ー . . . . .	235
障害の処理 . . . . .	238
EIS バインディング . . . . .	244
EIS バインディング: 全体像 . . . . .	245
EIS バインディングの主な特徴 . . . . .	245
JCA 対話仕様および接続仕様の動的プロパティ ー . . . . .	249
EIS バインディングの管理 . . . . .	251
EIS バインディングを持つ外部クライアント . . . . .	251
JMS バインディング . . . . .	252
JMS バインディング: 全体像 . . . . .	253
JMS 統合とリソース・アダプター . . . . .	256
JMS バインディングの主な特徴 . . . . .	257
JMS バインディングの管理 . . . . .	257
JMS ヘッダー . . . . .	260
外部クライアント . . . . .	261
JMS バインディングのトラブルシューティング 例外の処理 . . . . .	263
汎用 JMS バインディング . . . . .	265
汎用 JMS バインディング: 全体像 . . . . .	266
汎用 JMS バインディングの主な機能 . . . . .	269
汎用 JMS バインディングの管理 . . . . .	271
汎用 JMS ヘッダー . . . . .	274
汎用 JMS バインディングのトラブルシューティ ング . . . . .	275
例外の処理 . . . . .	277
WebSphere MQ JMS バインディング . . . . .	277
WebSphere MQ JMS バインディング: 全体像 . . . . .	278
WebSphere MQ JMS バインディングの主な特徴 . . . . .	280
WebSphere MQ JMS バインディングの管理 . . . . .	282
JMS ヘッダー . . . . .	285
外部クライアント . . . . .	287
WebSphere MQ JMS バインディングのトラブル シューティング . . . . .	287
例外の処理 . . . . .	288
WebSphere MQ バインディング . . . . .	289
WebSphere MQ バインディング: 全体像 . . . . .	289
WebSphere MQ バインディングの主な特徴 . . . . .	292
WebSphere MQ バインディングの管理 . . . . .	294
WebSphere MQ ヘッダー . . . . .	298
Service Component Architecture モジュールと WebSphere MQ . . . . .	299
外部クライアント . . . . .	300
WebSphere MQ バインディングのトラブルシュー ティング . . . . .	301
例外の処理 . . . . .	302
EJB バインディング . . . . .	303
EJB バインディング: 全体像 . . . . .	303
EJB バインディングの操作 . . . . .	307

EJB バインディングの管理 . . . . .	307
外部クライアント . . . . .	309
HTTP バインディング . . . . .	309
HTTP バインディング: 全体像 . . . . .	310
HTTP バインディングの管理 . . . . .	311
HTTP ヘッダー . . . . .	314
WebSphere Transformation Extender データ・ハンド ラー . . . . .	319
WebSphere Transformation Extender マップおよび データ・ハンドラー . . . . .	321
データ・バインディング記述子の設定 . . . . .	324

## 第 10 章 サーバーによってサポートさ れるアダプター . . . . . 333

WebSphere Adapters . . . . .	336
WebSphere Business Integration Adapter . . . . .	337
WebSphere Business Integration Adapter の管理 . . . . .	337

## 第 11 章 イベントの使用 . . . . . 339

イベントの順次処理 . . . . .	339
例: イベント順序付け . . . . .	340
イベント順序付けを実装する場合の考慮事項 . . . . .	342
WebSphere Process Server におけるイベント順序 付けの有効化 . . . . .	347
ロックのリスト、解放、および削除 . . . . .	350
イベント順序付けのトラブルシューティング . . . . .	351
失敗イベントの管理 . . . . .	353
リカバリーのセキュリティ考慮事項 . . . . .	357
失敗イベントの検索 . . . . .	358
失敗イベントのデータの操作 . . . . .	361
失敗イベントの再サブミット . . . . .	366
Business Process Choreographer の停止イベント の管理 . . . . .	367
失敗イベントに関連するビジネス・プロセス・イ ンスタンスの検索 . . . . .	368
失敗イベントに関連する Common Base Event の 検索 . . . . .	369

失敗イベントの削除 . . . . .	369
Failed Event Manager のトラブルシューティング . . . . .	370

## 第 12 章 WebSphere Process Server のトラブルシューティング管理 . . . . . 373

アプリケーションのクロス・コンポーネント・トレ ースの使用 . . . . .	373
クロス・コンポーネント・トレースの使用可能化 . . . . .	373
データ・スナップショット付きクロス・コンポー ネント・トレースの使用可能化 . . . . .	374
クロス・コンポーネント・トレースの使用不可化 . . . . .	375
クロス・コンポーネント・トレースで収集された データの削除 . . . . .	376
Service Component Architecture (SCA) 処理と呼 び出しチェーンのトラブルシューティング . . . . .	376
Failed Event Manager のトラブルシューティング . . . . .	377
Service Component Architecture と WebSphere MQ の通信のトラブルシューティング . . . . .	379
イベント順序付けのトラブルシューティング . . . . .	380
オブジェクト・リクエスト・ブローカー (ORB) の サービス設定のトラブルシューティング . . . . .	382
デプロイメント環境のトラブルシューティング . . . . .	383
ビジネス・ルール・マネージャーのトラブルシュー ティング . . . . .	384
ログイン・エラーの解決 . . . . .	384
ログイン競合エラーの解決 . . . . .	385
アクセス競合エラーの解決 . . . . .	385
メッセージング・バインディングのトラブルシュー ティング . . . . .	386
JMS バインディングのトラブルシューティング . . . . .	386
汎用 JMS バインディングのトラブルシューティ ング . . . . .	387
WebSphere MQ バインディングのトラブルシュ ーティング . . . . .	389
プロファイル固有のログ・ファイル . . . . .	390

## 特記事項 . . . . . 395





1. ガイド付きアクティビティ . . . . .	5	17. JMS エクスポート・バインディングのリソース . . . . .	256
2. 別のメディエーション・モジュールと対話する メディエーション・モジュールの例 . . . . .	93	18. 単純なユース・ケース・シナリオ: 外部クライアントとサーバー・アプリケーションの対話 . . . . .	262
3. ビジネス・ルール・マネージャーのイベント・シーケンス . . . . .	178	19. 汎用 JMS インポート・バインディング・リソース . . . . .	267
4. デシジョン・テーブル . . . . .	207	20. 汎用 JMS エクスポート・バインディング・リソース . . . . .	269
5. HTTP バインディングを持つエクスポート . . . . .	223	21. WebSphere MQ JMS インポート・バインディングのリソース . . . . .	279
6. HTTP バインディングを持つインポート . . . . .	224	22. WebSphere MQ JMS エクスポート・バインディングのリソース . . . . .	280
7. エクスポートを経由したコンポーネントへの要求のフロー . . . . .	226	23. WebSphere MQ インポート・バインディングのリソース . . . . .	291
8. エクスポートを経由して戻る応答のフロー . . . . .	227	24. WebSphere MQ エクスポート・バインディングのリソース . . . . .	292
9. インポートを経由したコンポーネントからサービスへのフロー . . . . .	228	25. SCA アプリケーションから Web アプリケーションへの要求フロー . . . . .	310
10. インポートを経由して戻る応答のフロー . . . . .	229	26. クライアント・アプリケーションから Web サービスへの要求フロー . . . . .	311
11. 関数セクター . . . . .	236	27. WebSphere Adapter の詳細な図式 . . . . .	334
12. 障害情報をエクスポート・バインディングを経由してコンポーネントからクライアントに送信する方法 . . . . .	239	28. WebSphere Business Integration Adapter の詳細な図式 . . . . .	335
13. 2 つの障害を持つインターフェース . . . . .	240		
14. 障害情報をインポートを経由してサービスからコンポーネントに送信する方法 . . . . .	241		
15. Siebel システムから SAP システムへのフロー . . . . .	245		
16. JMS インポート・バインディングのリソース . . . . .	255		



## 表

1.	本書の外部にあるトピックへのリンクのプレフィックスとなるアイコン . . . . .	iv	21.	汎用 JMS インポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名 . . . . .	271
2.	コンソールのコレクション・ページ上部のグラフィック・ボタン . . . . .	7	22.	汎用 JMS エクスポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名 . . . . .	272
3.	コンソール・ページ下部のボタン . . . . .	7	23.	MQ JMS インポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名 . . . . .	283
4.	WebSphere Process Server 構成ファイル	11	24.	MQ JMS エクスポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名 . . . . .	283
5.	デプロイメント環境のコンポーネントの関係	67	25.	WebSphere MQ インポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名 . . . . .	295
6.	エンタープライズ・アプリケーションを管理するためのボタン . . . . .	104	26.	WebSphere MQ エクスポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名 . . . . .	296
7.	Service Integration Bus Browser のアイコン	113	27.	EJB インポート・バインディングの主なコンポーネント . . . . .	303
8.	エンタープライズ・アプリケーションを管理するためのボタン . . . . .	133	28.	障害ハンドラーの戻り値 . . . . .	306
9.	リレーションシップ・データベース・ビューの列 . . . . .	145	29.	EJB インポートの JNDI 名の構成 . . . . .	308
10.	機能ボタン . . . . .	182	30.	提供される HTTP ヘッダー情報 . . . . .	315
11.	事前定義データ・ハンドラー . . . . .	232	31.	WebSphere Adapter および WebSphere Business Integration Adapter 間の差異 . . . . .	335
12.	JMS バインディング用の事前定義データ・バインディング . . . . .	233	32.	Network Deployment 環境でのイベント順序付けのサポート . . . . .	345
13.	WebSphere MQ バインディング用の事前定義データ・バインディング . . . . .	234	33.	esAdmin listLocks コマンドの出力例 . . . . .	351
14.	HTTP バインディング用の事前定義データ・バインディング . . . . .	235	34.	検索基準 . . . . .	360
15.	JMS バインディング用の事前定義関数セレクター . . . . .	236	35.	実行時に更新されるプロファイル固有のログ・ファイル . . . . .	391
16.	WebSphere MQ バインディング用の事前定義関数セレクター . . . . .	237			
17.	HTTP バインディング用の事前定義関数セレクター . . . . .	237			
18.	プリパッケージされている障害セレクター	242			
19.	インポート・バインディングの値の例	258			
20.	エクスポート・バインディングの値の例	258			



---

## 第 1 章 WebSphere Process Server の管理の概要

WebSphere® Process Server を管理する場合、アプリケーションとリソースのデプロイ先である環境の準備、モニター、変更などを実行します。また、アプリケーションとリソースの管理も行います。以下のトピックを読み、管理タスクに使用されるインターフェースと構成ファイルについて理解してください。

---

### 管理コンソール

管理コンソールは、アプリケーション、サービス、およびその他のリソースをセル、ノード、サーバー、またはクラスターの有効範囲で管理するために使用されるブラウザ・ベースのインターフェースです。管理コンソールは、スタンドアロン・プロセス・サーバーで使用することもできますし、ネットワーク環境のセル内のすべてのサーバーを管理するデプロイメント・マネージャーで使用することもできます。

**注:** 管理コンソールは、全体的に見れば Integrated Solutions Console フレームワークの一部であり、部分的に見れば WebSphere Application Server 管理コンソールの一部です。そのため、多数の管理タスク (例えば、セキュリティの設定、ログの表示、およびアプリケーションのインストール) は、WebSphere Process Server および WebSphere Enterprise Service Bus など、コンソールを使用するすべての製品で同一の内容です。これらのタスクは、WebSphere Application Server インフォメーション・センターに記載されています。

スタンドアロン・プロファイルをインストールしている場合、「セル」という独自の管理可能ドメイン内に単一のノードがあります。管理コンソールを使用して、管理可能ドメイン内のアプリケーション、バス、サーバー、およびリソースを管理します。

同様に、ネットワーク・デプロイメント・セルをインストールおよび構成している場合、同一セル内に 1 つのデプロイメント・マネージャー・ノードと 1 つ以上の管理対象ノードがあります。管理コンソールは、アプリケーションの管理、セル内の管理対象ノードのセットアップ、およびこれらのノードやそのリソースのモニターと制御のために使用します。

管理コンソールでは、タスク・フィルターによって、単純化されたユーザー・エクスペリエンスが提供され、段階的に機能を開示することによって、基礎になる WebSphere Application Server 管理機能にアクセスできるようになります。

### 管理コンソールの領域

管理コンソールを使用して、リソース、アプリケーション、およびサーバーなどのオブジェクトを作成および管理できます。さらに、管理コンソールを使用して、製品メッセージを表示します。このトピックでは、管理コンソールに表示されるメイン領域について説明します。

管理コンソールを表示するには、管理コンソール用のサーバーが稼働している必要があります。スタンドアロン・サーバーを構成した場合、コンソールはそのサーバーで実行されます。ネットワーク・デプロイメント・セルを構成した場合、コンソールはデプロイメント・マネージャー・サーバーで実行されます。

Web ブラウザーで管理コンソールの Web アドレスを指定し、「ログイン」ページでユーザー ID およびパスワード (セキュリティーが使用可能な場合) を入力します。

ナビゲーション・ツリーとワークスペースの間で境界線を左右にドラッグすると、それらの幅を同時にサイズ変更することができます。幅の変更は、管理コンソール・ユーザー・セッション間では保持されません。

コンソールには、以下のメイン領域があります。

### **タスクバー**

タスクバーは、コンソールのログアウト、製品情報へのアクセス、およびサポートへのアクセスのためのオプションを提供します。

### **ナビゲーション・ツリー**

コンソールの左側にあるナビゲーション・ツリーは、コンソール・ページへのリンクを提供します。このページを使用して、セル内でコンポーネントを作成および管理できます。

フォルダーまたは項目のツリーを展開するには、フォルダーまたは項目の横の正符号 (+) をクリックします。

フォルダーまたは項目のツリーを縮小するには、フォルダーまたは項目の横の負符号 (-) をクリックします。

ツリー表示の項目をクリックして、そのコンソール・ページを表示します。これにより、項目が展開ツリーと縮小ツリーの間で切り替わります。

### **ワークスペース**

コンソールの右側のワークスペースには、サーバーおよびリソースなどの構成オブジェクトを作成および管理するために使用するページが含まれています。

さまざまなタイプの構成済みオブジェクトを表示するには、ナビゲーション・ツリーのリンクをクリックします。

構成済みオブジェクトの構成、実行時の状況、およびオプションを表示するには、ワークスペース内で構成済みオブジェクトをクリックします。ボタンをクリックすると、選択したオブジェクトでアクションが実行されます。

ナビゲーション・ツリーの「ようこそ」をクリックして、ワークスペースのホーム・ページを表示できます。このページには、製品の使用に関する情報へのリンクが含まれています。

## 管理コンソールのビジネス・インテグレーション領域

WebSphere Process Server および WebSphere Enterprise Service Bus が使用するビジネス・インテグレーション・リソースは、管理コンソールのいくつかの領域にグループ化されます。

ナビゲーション・ツリーを使用して、以下のようにビジネス・インテグレーション・リソースを探します。

- 「**サーバー**」 > 「**デプロイメント環境**」：デプロイメント環境管理機能と、新規デプロイメント環境の作成を支援するウィザードを使用できます。

このオプションは、WebSphere Application Server Network Deployment をインストールしている場合にのみ使用可能です。

- 「**サーバー**」 > 「**アプリケーション・サーバー**」 > 「*server\_name*」：ここから以下を利用できます。
  - ビジネス・プロセスおよびヒューマン・タスク用のコンテナ設定
  - ビジネス・インテグレーション構成 (デプロイメント・ターゲット機能のタブ付きページ)
  - ビジネス・スペース構成
  - システム REST サービス・エンドポイント構成
  - サービス・コンポーネント・アーキテクチャーの構成
  - Common Event Infrastructure サーバーおよび宛先構成
  - Business Process Choreographer 構成
  - ビジネス・ルール構成
  - セレクター
  - WebSphere Business Integration Adapter サービス
  - Application Scheduler
- 「**サーバー**」 > 「**クラスター**」 > 「*cluster\_name*」：ここから以下を利用できます。
  - ビジネス・プロセスおよびヒューマン・タスク用のコンテナ設定
  - ビジネス・インテグレーション構成 (デプロイメント・ターゲット機能のタブ付きページ)
  - ビジネス・スペース構成
  - システム REST サービス・エンドポイント構成
  - サービス・コンポーネント・アーキテクチャーの構成
  - Common Event Infrastructure サーバーおよび宛先構成
  - Business Process Choreographer 構成
  - ビジネス・ルール構成
- 「**アプリケーション**」 → 「**SCA モジュール**」：ここから、以下の項目にアクセスすることができます。
  - SCA モジュールと、それに関連するサービス・アプリケーション
  - SCA モジュール・インポート (インターフェースやバインディングなど)
  - SCA モジュール・エクスポート (インターフェースやバインディングなど)
  - SCA モジュール・プロパティ

- 「リソース」：ここから以下を利用できます。
  - WebSphere Business Integration Adapter
  - 担当者ディレクトリー・プロバイダー
  - リモート成果物
- 「統合アプリケーション」：ここから以下を利用できます。
  - 失敗イベント・マネージャー
  - リレーションシップ・マネージャー
  - Common Base Event ブラウザー
- 「サービス統合」：ここから、以下の項目にアクセスすることができます。
  - WebSphere Service Registry and Repository (WSRR) の定義
  - Service Integration Bus Browser

## 管理コンソールのガイド付きアクティビティー

ガイド付きアクティビティーは、複数の管理コンソール・ページを使用する必要がある一般的な管理タスクの実行を支援します。

ガイド付きアクティビティーは、タスクを実行するために必要な各管理コンソール・ページを表示し、そのタスクを適切に実行する上で役立つ次の情報を提供します。

- タスクとそれに関連する重要な概念の紹介
- タスクを実行する状況と理由の説明
- 現行タスクの実行前および実行後に行う必要がある他のタスクのリスト
- タスクを完了するための主なステップ
- 問題の回避または問題からの回復のためのヒント
- オンライン資料にあるフィールドの説明およびタスクに関する詳細情報へのリンク

5 ページの図 1 に、ガイド付きアクティビティーが表示された管理コンソールの例を示します。

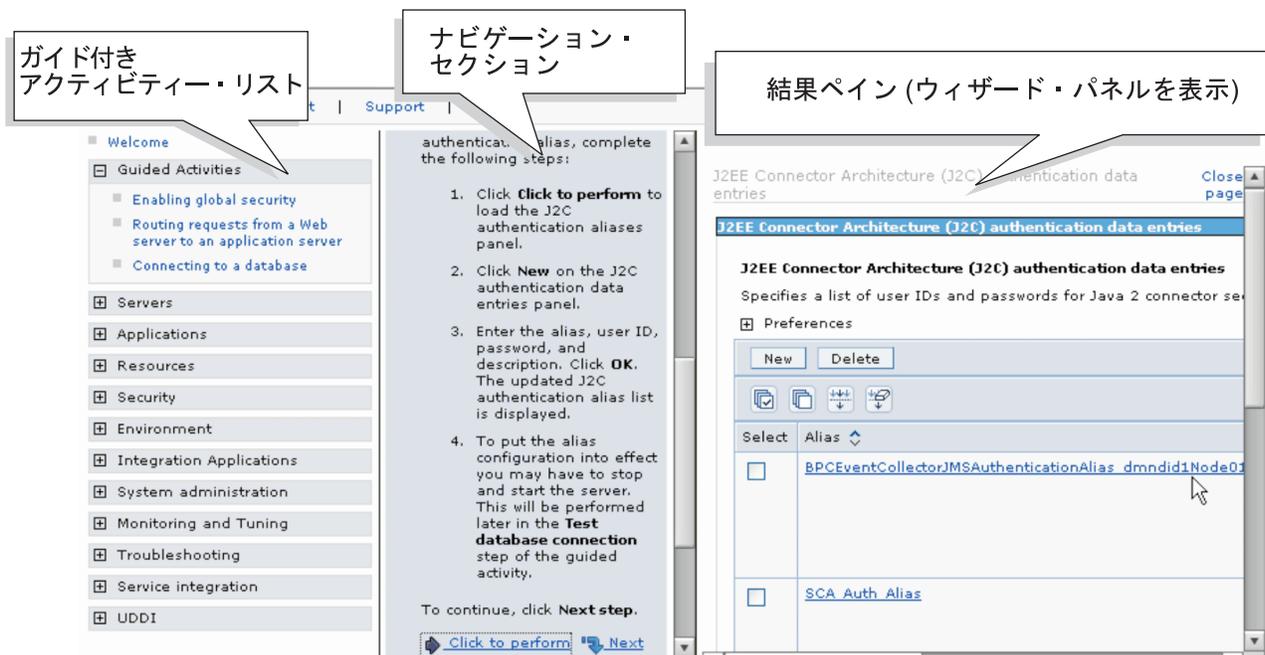


図1. ガイド付きアクティビティ

## 管理コンソール・ページ

管理コンソール・ページは、コレクション・ページ、詳細ページ、およびウィザード・ページの3つのいずれかのフォーマットになっています。それぞれのタイプのページのレイアウトおよび振る舞いを理解すると、効果的に使用できるようになります。

- 『コレクション・ページ』
- 6ページの『詳細ページ』
- 6ページの『ウィザード・ページ』

## コレクション・ページ

コレクション・ページは既存の管理オブジェクト (例えば、リレーションシップ、失敗イベント、またはリソース・アダプター) のコレクションを管理します。これには、次のエレメントが1つ以上含まれています。

### スコープと設定

有効範囲および設定によって、どの管理オブジェクトを表に表示するか、およびそれらをどのように表示するかを決定できます。

### 既存オブジェクトの表

この表は、コレクション・ページで指定したタイプの、既存の管理オブジェクトを示しています。表の列は、これらのオブジェクトの主要な設定値をまとめたものです。オブジェクトがまだ存在していない場合は、表は空です。選択可能なボタンを使用して、新規オブジェクトを作成してください。

### アクションを実行するためのボタン

標準的なボタンは6ページの『管理コンソールのボタン』で説明されています。

ます。ほとんどの場合、コレクション・テーブルから 1 つ以上のオブジェクトを選択してから、ボタンをクリックする必要があります。アクションは、選択されたすべてのオブジェクトに適用されます。

### ソートのトグル・ボタン

表の各列見出しの後には、項目を昇順 (v) または降順 (v) にソートするためのアイコンがあります。デフォルトでは、オブジェクト名などの項目はアルファベット順に降順でソートされます。

## 詳細ページ

詳細ページは、オブジェクトに関する詳細を表示したり、特定のオブジェクト (アプリケーション・サーバーまたはリスナー・ポート拡張など) を構成するために使用されます。通常、これには次のエレメントが 1 つ以上含まれています。

### 構成タブの付いたページ

このタブ付きページは、管理オブジェクトの構成を変更するために使用されます。各構成ページには、オブジェクト固有の一連の一般プロパティが含まれています。構成している管理オブジェクトのタイプに応じて、ページに追加のプロパティを表示できます。

このタブ付きページへの変更を有効にするには、サーバーの再始動が必要になることがあります。

### ランタイム・タブの付いたページ

このタブ付きページには、管理オブジェクトの現在使用中の構成が表示されます。これは、読み取り専用であることがあります。一部の詳細ページにランタイム・タブがないことに注意してください。

このタブ付きページへの変更は、即時に有効になります。

### ローカル・トポロジー・タブの付いたページ

このタブ付きページには、管理オブジェクトの現在使用中のトポロジーが表示されます。異なるレベルのトポロジーを展開および縮小表示して、トポロジーを表示します。一部の詳細ページにローカル・トポロジー・タブがないことに注意してください。

### アクションを実行するためのボタン

特定のアクションを実行するボタンは、構成タブの付いたページとランタイム・タブの付いたページにのみ表示されます。標準的なボタンは『管理コンソールのボタン』で説明されています。

## ウィザード・ページ

ウィザード・ページは、いくつかのステップからなる構成プロセスの実行に役立ちます。ウィザードは、構成している特定のオブジェクトの特性に応じて、特定のステップを表示したり非表示にしたりすることができます。4 ページの『管理コンソールのガイド付きアクティビティ』を参照してください。

## 管理コンソールのボタン

管理コンソール・インターフェースには、現在表示されているページに応じて、多数のボタンが含まれています。このトピックでは、使用可能なコンソール・ボタンについて説明します。

以下のグラフィック・ボタンは、サーバー関連リソースを表示するテーブルの上部に配置されています。

表2. コンソールのコレクション・ページ上部のグラフィック・ボタン

ボタン	結果のアクション
すべてにチェック・マークを付ける	テーブルにリストされているそれぞれのリソースを選択します (例えば、失敗イベント、またはリレーションシップ・インスタンス)。これは、これらのリソースに対してアクションを実行するための準備です。
チェック・マークをすべて外す	すべての選択済みリソースをクリアします。これにより、リソースに対するアクションは実行されなくなります。
フィルター・ビューを表示する	フィルターを設定するためのダイアログ・ボックスが開きます。フィルターは、テーブルに表示するリソースのサブセットを指定するために使用します。17 ページの『管理コンソール・フィルターの設定』を参照してください。
フィルター・ビューを隠す	フィルターを設定するために使用するダイアログ・ボックスを非表示にします。
フィルター値をクリア	フィルターに対して行ったすべての変更をクリアし、最後に保管した値を復元します。

以下のボタンが、管理コンソール・ページの下部に表示されます。すべてのボタンがすべてのページに表示されるわけではありません。

表3. コンソール・ページ下部のボタン

ボタン	結果のアクション
追加	選択または入力した項目をリストに追加するか、または項目をリストに追加するためのダイアログ・ボックスを表示します。
適用	ページを終了せずに、ページへの変更を保管します。
戻る	シーケンスの直前のページまたは項目を表示します。管理コンソールでは、Web ブラウザーの「戻る」オプションおよび「進む」オプションの使用をサポートしていません。これらを使用すると、偶発的な問題が起こる可能性があります。代わりに、コンソールの「戻る」または「キャンセル」ボタンを使用します。
キャンセル	現在のページまたはダイアログ・ボックスを終了します。保管されていないすべての変更は破棄されます。管理コンソールでは、Web ブラウザーの「戻る」オプションおよび「進む」オプションの使用をサポートしていません。これらを使用すると、偶発的な問題が起こる可能性があります。代わりに、コンソールの「戻る」または「キャンセル」ボタンを使用します。
クリア	変更をクリアして、最後に保管された値を復元します。

表 3. コンソール・ページ下部のボタン (続き)

ボタン	結果のアクション
選択をクリア	このタブ付きページの表で選択されているセルをすべてクリアします。
閉じる	ダイアログを閉じます。
削除	選択したインスタンスを除去します。
OK	変更を保管して、ページを終了します。
リセット	タブまたはページ上の変更を消去し、最後に保管された値を復元します。
保管	ローカル構成の変更をマスター構成に保管します。

管理コンソールで使用するすべての製品とリソースの管理用ボタンの完全なリストについては、WebSphere Application Server インフォメーション・センターの『管理コンソールのボタン』を参照してください。

## コマンド行ツール、スクリプト、およびプログラミング・インターフェース

WebSphere Process Server には、ランタイム環境を管理するためのコマンド行ツール、スクリプト・インターフェース、およびプログラミング・インターフェース (管理プログラム) が組み込まれています。

### コマンド行ツール

コマンド行ツールは、特定のタスクを実行するために、オペレーティング・システムのコマンド行プロンプトから実行する簡単なプログラムです。このツールを使用して、アプリケーション・サーバーの始動と停止、サーバー状況の確認、ノードの追加または除去、およびその他のタスクを実行できます。

WebSphere Process Server コマンド行ツールには、`serviceDeploy` コマンドが含まれており、このコマンドは、WebSphere Integration Developer 環境からエクスポートされた `.jar`、`.ear`、`.war` および `.rar` ファイルを処理して、それらをプロダクション・サーバーにインストールするための準備をします。

コマンド行ツールについての詳細は、このインフォメーション・センターの『コマンドおよびスクリプト』を参照してください。

### スクリプト (wsadmin)

WebSphere 管理 (wsadmin) スクリプト・プログラムは、非グラフィカル・コマンド・インタプリタ環境であり、スクリプト言語で管理オプションを実行したり、スクリプト言語プログラムの実行を依頼したりすることができます。これは、管理コンソールと同じタスクをサポートします。wsadmin は、実稼働環境および不在操作向けのツールです。

プログラミング・インターフェースについての詳細は、このインフォメーション・センターの『コマンドおよびスクリプト』を参照してください。

## 管理プログラム

管理プログラムは、Java Management Extensions (JMX) 仕様に基づく一連の Java™ クラスおよびメソッドであり、Service Component Architecture (SCA) およびビジネス・オブジェクトを管理するためのサポートを提供します。各プログラミング・インターフェースには、その目的の説明、インターフェースまたはクラスを使用する方法を示す例、および個々のメソッド説明への参照が含まれています。

プログラミング・インターフェースについての詳細は、このインフォメーション・センターの、『生成された API 文書と SPI 文書 (Generated API and SPI documentation)』を参照してください。

---

## Business Process Choreographer Explorer の概要

Business Process Choreographer Explorer は、ビジネス・プロセスおよびヒューマン・タスクと対話するための汎用 Web ユーザー・インターフェースを実装している Web アプリケーションです。

また、オプションのレポート作成機能が組み込まれていますが、この機能はこれまでは Business Process Choreographer Observer と呼ばれていました。

サーバーまたはクラスター上で Business Process Choreographer Explorer インスタンスを 1 つ以上構成できます。WebSphere Process Server プロファイルを持つ WebSphere Process Server のインストール済み環境、または WebSphere Process Server クライアントのインストール済み環境があれば十分です。Business Process Choreographer をサーバーまたはクラスター上に構成する必要はありません。クライアントを WebSphere Process Server に接続する必要がある唯一のインフラストラクチャーは WebSphere Process Server クライアント・インストールですが、これには Business Process Choreographer Explorer は含まれません。WebSphere Process Server クライアント・インストールの場合でも、デプロイメント・マネージャーを使用して Business Process Choreographer Explorer をサーバーにインストールしてください。

単一の Business Process Choreographer Explorer は、1 つの Business Process Choreographer 構成に接続できますが、ローカル構成に接続する必要はありません。ただし、Business Process Choreographer Explorer の複数のインスタンスを同じサーバーまたはクラスター上で構成して、各インスタンスを異なる Business Process Choreographer 構成に接続できます。

Business Process Choreographer Explorer を始動した場合、ユーザー・インターフェースに表示されるオブジェクトおよび実行できるアクションは、所属先のユーザー・グループとそのグループに与えられている権限に応じて異なります。例えば、ビジネス・プロセス管理者の場合は、デプロイされたビジネス・プロセスの処理を円滑に進める責任があります。したがって、プロセス・テンプレート、タスク・テンプレート、プロセス・インスタンス、タスク・インスタンス、およびこれらの関連オブジェクトに関する情報を表示できます。これらのオブジェクトを処理することもできます。例えば、新規プロセス・インスタンスの開始、タスクの作成および開始、失敗したアクティビティの修復および再開、作業項目の管理、完了したプ

ロセス・インスタンスやタスク・インスタンスの削除などを実行できます。ただし、ユーザーの場合は、割り当てられているタスクについてのみ、表示や処理が可能になります。

---

## ビジネス・ルール・マネージャー

ビジネス・ルール・マネージャーは、ビジネス・アナリストがビジネス・ルール値を表示および変更する際に使用する Web ベースのツールです。このツールは、WebSphere Process Server のオプションであり、プロファイル作成時またはサーバーのインストール後に選択してインストールできます。

ビジネス・ルールは、WebSphere Integration Developer 内で、if/then ルール・セットおよびデシジョン・テーブルを使用して操作を実装することにより設計および開発します。ビジネス・ルールは、WebSphere Business Modeler 内で作成することもできます。ただし、Modeler では、ビジネス・ルール・タスクの作成のみがサポートされます (ビジネス・ルール・タスクは、Modeler からエクスポートしたときにルール・セットになります)。ルール・セットおよびデシジョン・テーブルはテンプレートに設定します。テンプレートは、変更可能なビジネス・ルールの側面と変更の程度を制御します。それらによって、デシジョン・テーブルの if/then ルール、条件ケース、およびアクションの構造を定義します。

テンプレートは、ビジネス・ルール・マネージャーでのビジネス・ルール・ランタイム・オーサリングのメカニズムを提供します。テンプレートを使用すると、ビジネス・ルールの値を変更したり、ルール・セット内に新しいルールを作成したり、デシジョン・テーブル内に新しい条件またはアクションを作成したり、実行時にビジネス・ルール定義に対する変更を公開したりできます。

ビジネス・ルールはビジネス・ルール・グループとして編成されます。ビジネス・ルール・グループは、ルールに対するインターフェースとして、ルールを呼び出すために使用されます。ルール・セットおよびデシジョン・テーブルが直接呼び出されることはありません。

ビジネス・ルールの構築およびデプロイについての詳細は、WebSphere Integration Developer インフォメーション・センターを参照してください。

---

## 構成情報

WebSphere Process Server 用の構成データは、構成リポジトリ・ツリー (マスター・リポジトリ) のディレクトリーに保管されている XML ファイルに格納されています。

構成ファイルが配置されているディレクトリーによって、その有効範囲 (データが適用される範囲) が決まります。

- 個々のサーバーのディレクトリー内のファイルは、そのサーバーにのみ適用されます。
- アプリケーション・ディレクトリー内のファイルは、そのアプリケーションにのみ適用されます。
- クラスター・レベルのディレクトリー内のファイルは、そのクラスターにのみ適用されます。

- ノード・レベルのディレクトリー内のファイルは、そのノード上のすべてのサーバーに適用されます。
- セル・ディレクトリー内のファイルは、セル全体内のすべてのノード上のすべてのサーバーに適用されます。

表 4. WebSphere Process Server 構成ファイル

構成ファイル	説明
server-wbi.xml	サーバーとそのコンポーネント (Adaptive Entity Service、拡張メッセージング・サービス、ビジネス・ルールとセレクター監査サービス、WebSphere Business Integration Adapter Service 構成を含む) を識別します。
resources-wbi.xml	WebSphere Process Server 用のオペレーティング環境リソースを定義し、セル・スコープ、ノード・スコープ、およびサーバー・スコープに存在します。これには、拡張メッセージング・プロバイダー、および WebSphere Business Integration Adapters が含まれます。
cell-wbi.xml	セルを識別します。このファイルは、リレーシヨニシッ・サービス構成を格納するために使用され、セル・スコープにのみ存在します。
server-bpc.xml	Business Process Choreographer コンテナおよびそのコンポーネントを識別します。
resources-bpc.xml	Business Process Choreographer コンテナ用の稼働環境リソース (担当者ディレクトリー・プロバイダー用の構成情報を含む) を定義します。このファイルは、セル・スコープ、ノード・スコープ、およびサーバー・スコープに存在します。
deployment-bpc.xml	ビジネス・プロセス・コンテナ用のアプリケーション・デプロイメント設定を構成します。
server-core.xml	コア WebSphere Process Server 構成 (Artifact Loader Service および Business Context Data Service を含む) の構成情報を識別します。

WebSphere Process Server 構成ファイルは、管理コンソール、wsadmin、およびスクリプトを使用して編集できます。手動で編集する必要はありません。

詳しくは、『構成ファイルの説明』を参照してください。



---

## 第 2 章 管理インターフェース入門

WebSphere Process Server をセットアップして管理する場合は、以下のトピックの情報を参照してください。

---

### 管理コンソール入門

管理コンソールを使用して WebSphere Process Server リソースの管理を開始する際には、このトピックのタスクを使用します。

以下のタスクは、サーバーと管理コンソールの始動、コンソールのスコープと設定の変更、およびマスター・リポジトリへの作業の保存に役立ちます。

- **サーバーを始動します。**

管理コンソールを使用するには、その前にスタンドアロン・サーバーまたはデプロイメント・マネージャーを始動しておく必要があります。スタンドアロン・サーバーを始動する方法については、『スタンドアロン・サーバーの開始』を参照してください。Network Deployment の場合は、『デプロイメント・マネージャーの始動』を参照してください。

- **管理コンソールを始動します。**

詳しくは、14 ページの『管理コンソールの始動と停止』を参照してください。

- **コンソールの設定を指定します。**

設定により、管理コンソールでのデータの表示方法と、ワークスペースの振る舞いを制御します。16 ページの『管理コンソール設定の指定』を参照してください。

- **コンソール・スコープを設定します。**

スコープは、リソースが管理コンソールに表示されるレベルを指定します。リソースは、セル、ノード、クラスター、サーバーいずれかのスコープでコンソール・コレクション・テーブルに表示されます。詳しくは、『管理コンソールの有効範囲設定』を参照してください。

- **情報を表示するためのフィルターを作成します。**

フィルターは、コレクション・ページの列にどのデータを表示するかを指定します。17 ページの『管理コンソール・フィルターの設定』を参照してください。

- **オプション: コンソール用のセッション・タイムアウトを設定します。**

デフォルトでは、非アクティブのまま 30 分経過すると、コンソール・セッションがタイムアウトになります。『構成文書の説明』で説明されているように、deployment.xml 構成ファイルを編集してこの値を変更することができます。

- **自分の作業をマスター・リポジトリに保管します。**

変更内容をマスター・リポジトリに保管するまで、コンソールは、ローカル・ワークスペースを使用して変更内容を追跡します。変更内容を保管するには、

「システム管理」 > 「変更内容をマスター・リポジトリに保管 (Save Changes to Master Repository)」をクリックして「保管」ページを表示してから、「保管」をクリックします。

## 管理コンソールの始動と停止

管理コンソールを利用するには、管理コンソールを始動してログインする必要があります。コンソールでの作業が終了したら、作業内容を保管してログアウトします。

### 始める前に

管理コンソールが必要とするアプリケーション・サーバーを始動していることを確認します。

### このタスクについて

以下の手順に従い、コンソールの起動、ログイン、ログアウトを行います。

#### 手順

1. 管理コンソールを始動します。
  - a. 管理コンソールへのアクセスに使用する Web ブラウザーで、Cookie を使用可能にします。
  - b. オプション: JavaScript™ を使用可能にします。管理コンソールのすべての機能が使用可能になるように、JavaScript を使用可能にすることをお勧めします。
  - c. Cookie を使用可能にした Web ブラウザーで、以下のように入力します。

`http://your_fully_qualified_server_name:portNumber/ibm/console`

ここで、*your\_fully\_qualified\_server\_name* には、管理サーバーが入っているマシンの完全修飾ホスト名を指定し、*portNumber* は、管理コンソールのポート番号です。管理コンソールがローカル・マシン上にある場合、セキュリティが有効になっていなければ、*your\_fully\_qualified\_server\_name* は `localhost` になります。

Windows プラットフォームで `localhost` が認識されない場合は、実際のホスト名を使用してください。

セキュリティが有効になっていれば、要求は `https://your_fully_qualified_server_name:secure_portNumber/ibm/console` にリダイレクトされます。ここで、*your\_fully\_qualified\_server\_name* は管理サーバーが入っているマシンの完全修飾ホスト名、*secure\_portNumber* は管理コンソールのセキュア・ポート番号です。

**注:** 非セキュア管理コンソールのデフォルト・ポート番号は、ポート 9060 です。セキュア管理コンソールのデフォルト・ポート番号は、9043 です。プロファイル作成中にデプロイされた新規の管理コンソールには、それぞれ新規の非セキュア・ポート番号が割り当てられます。プロファイル作成中にセキュリティを有効にした場合には、新規のセキュア・ポート番号が割り当てられます。

- d. コンソール・アプリケーションを実行するサーバーの System.Out.log ファイルをチェックして、コンソール・アプリケーションが正常に開始したことを確認します。正常に開始した場合には、「WSVR0221I: アプリケーションが始動されました: isclite」というメッセージが生成されます。

マシン上で既に実行中のアプリケーションとコンソール・ポートが競合するためにコンソールを始動できない場合は、以下のファイル内でポート番号を変更します。

- `profile_root/config/cells/cell_name/nodes/node_name/serverindex.xml`
- `profile_root/config/cells/cell_name/virtualhosts.xml`

プロファイル作成中に選択したポート (デフォルトでは 9060) をファイル内で検索し、そのポート番号をすべてコンソールのポートに変更します。または、競合するポートを使用している別のアプリケーションをシャットダウンした後で、管理コンソールを始動します。

**15/05** コンソール・ポートがマシン上で既に実行中のアプリケーションと競合するために管理コンソールを始動できない場合は、`chgwassvr` スクリプト・コマンドを使用してポート番号を変更します。詳しくは、『Changing the ports associated with an application server』を参照してください。または、競合するポートを使用している別のアプリケーションをシャットダウンした後で、アプリケーション・サーバーを始動します。

管理コンソールがブラウザにロードされ、ログイン・ページが表示されます。

2. コンソールにログインします。
- a. 「**ユーザー ID**」フィールドに、ユーザー名またはユーザー ID を入力します。ユーザー ID は、ログインするために使用された対象セッションの期間中にしか存続しません。

**注:** 既に使用中 (およびセッション中) の ID を入力すると、以下のいずれかを行うことを求めるプロンプトが出ます。

- 同じユーザー ID を持つもう一方のユーザーをログアウトする。そのユーザーのセッション中に加えられた変更を回復できます。
- ログイン・ページに戻り、別のユーザー ID を入力する。

サーバー構成に加えられたすべての変更は、ユーザー ID に保管されます。セッションがタイムアウトになった場合も、サーバー構成はユーザー ID に保管されます。

- b. コンソールのセキュリティが有効な場合は、「**パスワード**」フィールドにパスワードを入力する必要があります。
- c. 「**OK**」をクリックします。

ここで、管理コンソールに「ようこそ」ページが表示されます。

3. コンソールからログオフします。
- このセッション中の作業内容を保管するには、「**システム管理**」>「**変更内容をマスター・リポジトリに保管 (Save Changes to Master Repository)**」>「**保管**」をクリックしてから、「**ログアウト**」をクリックしてコンソールを終了します。

- ・ 変更内容をリポジトリに保管しないでコンソールを終了するには、「**ログアウト**」をクリックします。

作業内容を保管する前にブラウザを閉じた場合は、同じユーザー ID を使用して次にログインしたときに、保管しなかった変更内容を回復できます。

## 管理コンソール設定の指定

管理コンソール設定によって、コレクション・ページ (データまたはリソースのコレクションをテーブルに表示するページ) 上のデータの表示をカスタマイズできます。設定はユーザー・レベルで指定され、通常、管理コンソールの領域ごとに別個に設定する必要があります。

### このタスクについて

コレクション・ページに対して以下の表示設定を指定できます。

- ・ 「**最大行数**」: コレクションの規模が大きい場合に、表示される行の最大数を指定します。行数が指定した最大値を超える場合、後続のページに表示されます。デフォルト値は 20 です。
- ・ 「**フィルター基準の保存**」: フィルター機能で入力された最後の検索基準が保存されるかどうかを指定します。保存が指定されている場合、コンソール・コレクション・ページでは、設定に従ってテーブルにデータを表示する際に最初に保存されているフィルター基準を使用します。詳しくは、17 ページの『管理コンソール・フィルターの設定』を参照してください。
- ・ 「**最大結果セット・サイズ**」: 検索の結果として戻されるリソースの最大数を指定します。デフォルト値は 500 です。
- ・ 「**最大列幅**」: コレクション列で表示可能な文字の最大数を指定します。デフォルト値は 18 です。

以下のステップを実行して、コレクション・ページ用の表示設定を指定します。

#### 手順

1. 任意のコレクション・ページから「**設定**」をクリックします。

このページは設定フィールドを表示するために拡大されます。

2. 「**最大行数**」、「**フィルター基準の保存**」、「**最大結果セットサイズ**」、および「**最大列幅**」フィールドの値を必要に応じて変更します。
3. 「**適用**」をクリックします。

コレクション・テーブルが更新され、指定した値に従って表示されます。

### 次のタスク

ワークスペースの自動更新やデフォルトで使用するスコープなど、管理コンソールのグローバル設定も指定できます。管理コンソールで「設定」ページにアクセスするには、「**システム管理**」>「**コンソール設定**」>「**設定**」をクリックします。これらの設定の指定方法の資料については、WebSphere Application Server インフォメーション・センターを参照してください。

## 管理コンソール・フィルターの設定

管理コンソールのコレクション・ページの各テーブルには、WebSphere Process Server のデータまたはリソースのリストが表示されます。フィルターを使用すると、テーブルの特定の列に、どのリソースまたはデータを表示するかを正確に指定できます。1 つの列のみに対してフィルターを設定可能です。

### 手順

1. テーブルの上部のボタンから「**ビューのフィルター (Filter the view)**」をクリックします。

「フィルター」ダイアログ・ボックスが、テーブルの一番上の行の上にかきまします。

2. 「**フィルター**」ドロップダウン・メニューを使用して、フィルターに含める列を選択します。
3. 「**検索語**」フィールドにフィルター基準を指定します。

この基準はストリングで、名前にこのストリングが含まれているテーブル項目のみが表示されます。ストリングでは、ワイルドカード文字としてパーセント記号 (%)、アスタリスク (\*)、または疑問符 (?) 記号を使用できます。例えば、「リソース・アダプター (Resource Adapter)」ページで、「名前」列のフィルター基準として \*JMS\* を入力すると、名前にストリング JMS が含まれるすべてのリソース・アダプターを検索できます。

検索を実行する正規表現エンジンが正しく検索基準との突き合わせを行えるように、ストリングの一部として現れる ( ) ^ \* % { } ¥ + & の各文字の前には円記号 (¥) を置きます。

例えば、プロバイダー名に (XA) が含まれているすべての Java DataBase (JDBC) プロバイダーを検索する場合は、次のようなストリングを「検索語」フィールドに指定します。

\*¥(XA¥)\*

4. 「**実行 (Go)**」をクリックします。

テーブルが更新され、選択された列でフィルター基準に適合する項目のみが表示されます。

## 「マイ・タスク」の使用法

コンソール・ナビゲーションをカスタマイズするには、タスク・ビューを作成および編集します。

### このタスクについて

「マイ・タスク」では、コンソール・ナビゲーションに表示するタスクのリストを作成および編集できます。タスクには、そのタスクを完了するために使用される 1 つ以上の Web アプリケーション (コンソール・モジュール) を表示するページが含まれます。コンソールを初めて表示すると、アクセスできるすべてのタスクがナビゲーションに表示されます。「マイ・タスク」は、使用する頻度の高いタスクのみを表示するようにナビゲーションをカスタマイズするときに特に便利です。タスク

のカスタマイズ後には、コンソールにログインするたびに、最初に「マイ・タスク」が表示されます。

#### 手順

1. ナビゲーション・ツリーで、「ようこそ」リンクをクリックします。
2. ナビゲーションの「表示」選択リストから「マイ・タスク」を選択します。「マイ・タスク」を初めて使用する場合に「マイ・タスク」を開くには、「タスクの追加」をクリックする必要があります。
3. 「マイ・タスク」リストに追加するタスクを選択します。
4. 変更を保存するには、「適用」をクリックします。
5. 変更を取り消すには、「リセット」をクリックします。

#### タスクの結果

「適用」をクリックすると、カスタマイズされたタスク・リストがナビゲーションに表示されます。管理コンソールをシャットダウンしてから再始動する必要はありません。

## 管理コンソールからの製品情報とヘルプへのアクセス

管理コンソールからは、それぞれのページとフィールドのオンライン・ヘルプだけでなく、製品資料にもアクセスすることができます。ヘルプは、コンソールのヘルプ・ブラウザーか、WebSphere Process Server インフォメーション・センターで表示できます。

#### このタスクについて

製品情報と管理コンソールのヘルプ・トピックにアクセスするには、以下のステップを実行します。

#### 手順

1. 製品情報にアクセスするには、以下のタスクを実行します。
  - a. 管理コンソールのナビゲーション・ツリーの「ようこそ」をクリックします。ナビゲーション・ツリーの右のワークスペースに、インストール済み製品に関する情報が表示されます。
  - b. 適切なリンクをクリックして、製品のインフォメーション・センターと developerWork の関連技術情報にアクセスします。

2. 以下のいずれかの方法で、製品ヘルプにアクセスします。

オプション	説明
管理コンソールのフィールド・レベル・ヘルプにアクセスする	<ul style="list-style-type: none"> <li>• フィールド上にカーソルを移動すると、そのフィールドの吹き出しヘルプが表示されます。</li> <li>• フィールド上にカーソルを移動して、疑問符（「?」）アイコンが表示されるのを待ちます。アイコンが表示されたら、フィールド名をクリックします。ヘルプ・ポータル（ワークスペースの一番右側のパネル）に、そのフィールドに関する簡単なヘルプが表示されます。  <b>注:</b> そのフィールド、またはページ全体および関連タスクに関する拡張情報を表示するには、ヘルプ・ポータルの下部にある「このページの追加情報 (More information about this page)」リンクをクリックします。</li> </ul>
スタンドアロン・ヘルプ・ブラウザにアクセスする	<p>コンソール・タスクバーから「ヘルプ」をクリックすると、新しく Web ブラウザーが開いてオンライン・ヘルプが表示されます。ここでは、以下のオプションを選択できます。</p> <ul style="list-style-type: none"> <li>• 「索引 (Index)」タブに表示するトピックをブラウズします。トピックのリンクをクリックすると、ブラウザの右パネルにそのトピックが開きます。</li> <li>• 「検索」タブに 1 つ以上のキー・ワードを指定して、トピックを検索します。一致したすべてのトピックが、ナビゲーション・ツリーに表示されます。トピックを表示するには、トピック・リンクをクリックします。</li> </ul>
WebSphere Process Server インフォメーション・センターのオンライン・ヘルプを表示する	<ul style="list-style-type: none"> <li>• ブラウザーを使用して、WebSphere Process Server インフォメーション・センターに直接ナビゲートします。オンライン・ヘルプのトピックは、「参照」セクションにあります。</li> <li>• ヘルプ・ブラウザで表示したヘルプ・ファイルの「このトピックの更新情報を確認してください」リンクをクリックします。</li> </ul>
コマンド・アシスタンスの表示	<p>コマンド・アシスタンスが使用可能な場合は、右側のヘルプ・ポータルにある「最後のアクションの管理スクリプトを表示」をクリックします。</p>

### 関連タスク

20 ページの『管理コンソールからコマンド・アシスタンスにアクセスする操作』

コマンド・アシスタンスを使用すれば、管理コンソールの各アクションに対応する `wsadmin` スクリプト・コマンドを表示できます。そのようにしてコマンドを表示する機能は、`wsadmin` ユーティリティからサーバーを管理するために必要なコマンド行ツールを開発するときに便利です。

## 管理コンソールからコマンド・アシスタンスにアクセスする操作

コマンド・アシスタンスを使用すれば、管理コンソールの各アクションに対応する `wsadmin` スクリプト・コマンドを表示できます。そのようにしてコマンドを表示する機能は、`wsadmin` ユーティリティからサーバーを管理するために必要なコマンド行ツールを開発するときに便利です。

### 始める前に

コマンド・アシスタンスを使用する前に、以下の作業を行います。

- WebSphere Process Server と管理コンソールを開始します。
- コマンド・アシスタンス・データをログ・ファイルに保存するかどうかを決めます。ロギングを有効にすると、コマンド・アシスタンス・データを生成したページのタイム・スタンプとパンくずリストが、コンソールを実行中のプロセスの `logs` ディレクトリーにある `commandAssistanceJythonCommands_username.log` ファイルの `wsadmin` データに記録されます。

コマンド・アシスタンス・データをログ・ファイルに保存するには、「システム管理」 → 「コンソール設定」 → 「コマンド・アシスタンスのコマンドをログに記録する」をクリックします。

- コマンド・アシスタンスから Java Management Extensions (JMX) 通知を生成するかどうかを決めます。通知を有効にすると、自動化スクリプトを記述するための製品ツール (WebSphere Application Server Toolkit Jython エディターなど) との統合が可能になります。通知タイプは `websphere.command.assistance.jython.user_name` です (`user_name` は、現在の管理コンソール・ユーザーです)。

注: このオプションは、非実稼働環境でのみ使用することをお勧めします。

JMX 通知を有効にするには、「システム管理」 → 「コンソール設定」 → 「コマンド・アシスタンス通知を使用可能にする」をクリックします。

### このタスクについて

コマンド・アシスタンスを使用すれば、管理コンソールの多くのページで最後に実行したアクションに対応する `wsadmin` スクリプト・コマンドを Jython 言語で表示できます。

ヘルプ・ポートレットにコマンド・アシスタンスのリンクが表示されていれば、最後に実行したコンソール・アクションに対応する `wsadmin` コマンドが存在し、そのアクションに関するコマンド・アシスタンスを利用できます。

アクションの例としては、ボタンのクリックや、ナビゲーション・バー、コレクション・ページ、詳細ページのリンクのクリックなどがあります。フォームの編集はユーザー・アクションではないので、コマンド・アシスタンスによるデータ収集の対象ではありません。

wsadmin スクリプト・コマンドは、Jython 言語で 2 次ウィンドウに表示されます。「管理スクリプト・コマンド」ウィンドウを起動した後に管理アクションを実行した場合は、最新のコンソール・アクションに合わせてそのウィンドウのコマンド・リストが自動的に更新されます。

**ヘルプ・ポートレットでコマンド・アシスタンスを使用できない場合:** 一部のコンソール・アクションには wsadmin コマンドが直接関連付けられていません。管理コンソール・ページの右側にあるヘルプ・ポートレットにコマンド・アシスタンスのリンクがない場合は、最後のコンソール・アクションに対応するコマンド・アシスタンス・データがありません。

コンソールでコマンド・アシスタンスを使用するには、以下の手順を実行します。

#### 手順

1. オプション: コマンド・アシスタンス・データをログ・ファイルに取り込むには、コンソール設定を次のように構成します。
  - a. 「システム管理」 → 「コンソール設定」をクリックして「設定」ページを開きます。
  - b. 「コマンド・アシスタンスのコマンドをログに記録する」を選択します。
2. オプション: コマンド・アシスタンスから Java Management Extensions (JMX) 通知を生成するには、コンソール設定を次のように構成します。
  - a. 「システム管理」 → 「コンソール設定」をクリックして「設定」ページを開きます。
  - b. 「コマンド・アシスタンス通知を使用可能にする」を選択して、`websphere.command.assistance.jython.user_name` 通知を生成します。
3. コマンド・アシスタンスを使用するコンソール・ページにナビゲートします。
4. ページの右側にあるヘルプ・ポートレットで「最後のアクションの管理スクリプト・コマンドを表示」をクリックします。「管理スクリプト・コマンド」ウィンドウが開き、関連する wsadmin スクリプト・コマンドが Jython で表示されます。
5. オプション: 特定の wsadmin コマンドの説明を確認するには、そのコマンドの上にカーソルを配置して吹き出しヘルプを表示します。

#### タスクの結果

管理コンソールから wsadmin スクリプト・コマンドを表示しました。オプションとして、コマンドをログ・ファイルに記録し、コマンド・アシスタンスから JMX 通知を生成しました。

#### 次のタスク

管理タスクを自動化するために wsadmin スクリプトを作成するときに、コマンド・アシスタンスで表示した情報を利用できます。

#### 関連資料



コマンド・アシスタンスを使用した管理コンソール

コマンド・アシスタンスは、管理コンソール・アクションのサブセットに対して使用することができます。コマンド・アシスタンスが使用可能な場合、実行した

最後のコンソール・アクションの `wsadmin` スクリプト・コマンドが表示されません。このデータを使用して、特定の管理タスクを自動化する `wsadmin` スクリプトを作成できます。

### 関連情報

 コマンド・アシスタンスを備えた管理コンソール・アクション (WebSphere Application Server)

 スクリプトの使用 (`wsadmin`)

---

## Business Process Choreographer Explorer 入門

定義されているユーザー役割に応じて、Business Process Choreographer Explorer を使用してビジネス・プロセスとヒューマン・タスクを管理したり、割り当てられたタスクを処理することができます。ビジネス・プロセスおよびタスクの実行中に、WebSphere はプロセス・インスタンスとそれに関連するアクティビティの状態変更に関する情報が入ったイベントを発行できます。レポート作成機能を使用して、これらのイベントに基づく統計情報を取得し、プロセスおよびアクティビティに関するレポートを作成できます。

### このタスクについて

Business Process Choreographer Explorer を使用して、以下のタスクを実行できます。

- ビジネス管理者である場合は、ビジネス・プロセスのライフ・サイクルを管理し、ビジネス・プロセスを修復することができます。例えば、単一のアクティビティの実行を再始動または強制実行するか、ビジネス・プロセス全体を補正することができます。補正に失敗した場合は、そのプロセス・インスタンスを再試行するか、スキップするか、停止することができます。さらに、ビジネス・プロセスおよびアクティビティのカスタム・プロパティを追加および更新することもできます。
- ヒューマン・タスク管理者であれば、ヒューマン・タスクのライフ・サイクルを管理でき、作業割り当てを管理できます。例えば、ユーザーに責任を割り当てたり、ユーザーの不在処理や代替ユーザーを管理したりすることができます。また、ヒューマン・タスクの優先順位とビジネス・カテゴリーを変更し、カスタム・プロパティの追加または更新を行うこともできます。
- Business Process Choreographer Explorer のレポート機能を使用すると、プロセス・インスタンス、アクティビティ・インスタンス、またはインライン・ヒューマン・タスクの履歴をモニターできます。Business Process Choreographer Explorer の構成にレポート機能が含まれる場合は、独自のレポートを定義するか、ドリルダウン機能を使用することにより、特定のプロセス・インスタンス、アクティビティ・インスタンス、またはインライン・ヒューマン・タスクに関する詳細な情報を取得できます。また、報告された結果をエクスポートし、外部でさらに処理することもできます。
- ビジネス・ユーザーである場合は、Business Process Choreographer Explorer を使用して、割り当てられたタスクを処理できます。例えば、ビジネス・プロセス、サービス、およびヒューマン・タスクを開始することができ、ヒューマン・タスク

クの処理、編集、保存、実行、または解放を行うことができます。そのほか、不在のフラグを立て、代理人を定義できます。

さらに、Business Process Choreographer Explorer には検索機能があり、これを使用すると、注意が必要なビジネス・プロセスとそれに関連するアクティビティーおよびヒューマン・タスクを検出できます。例えば、それらのインスタンスの状況を検査し、関連するインスタンスおよびテンプレートの間をナビゲートし、関連するアクティビティーおよびヒューマン・タスクを含んでいるプロセス状態のグラフィカル・ビューを取得することができます。

### 関連タスク



#### タスク・テンプレートとタスク・インスタンスの管理

タスク・テンプレートを管理するには、管理コンソールまたは管理コマンドを使用します。タスク・インスタンスを処理するには、Business Process Choreographer Explorer を使用します。



#### 作業割り当ての管理

タスクの開始後に、例えば、ワークグループのメンバー間でワークロードを効率よく分散するために、そのタスクに対する作業割り当ての管理が必要になる場合があります。



#### タスク・インスタンスの作成と開始

タスク・インスタンスは、使用を許可されている任意のタスク・テンプレートから作成し、開始することができます。



#### タスクの処理

タスクを処理するには、タスクを要求してから、タスクの完了に必要なアクションを実行する必要があります。



#### ビジネス・プロセスおよびアクティビティーについてのレポート作成

ビジネス・プロセスおよびアクティビティーの処理中に、プロセス、アクティビティー、またはタスクの状態が変わると、イベントが生成されることがあります。これらのイベントは保管され、Business Process Choreographer Explorer によるレポート作成に利用できます。例えば、プロセスのパフォーマンス問題の分析レポートや、アクティビティーから呼び出されるサービスの信頼性評価のレポートなどです。

## Business Process Choreographer Explorer のユーザー・インターフェース

Business Process Choreographer Explorer は、ビジネス・プロセスおよびヒューマン・タスクを管理し、プロセスおよびアクティビティー・イベントに関するレポートを作成するための管理機能のセットを提供するスタンドアロン Web アプリケーションです。このインターフェースは、タスクバー、ナビゲーション・ペイン、およびワークスペースで構成されます。

以下の図は、Business Process Choreographer Explorer ユーザー・インターフェースのレイアウトを示しています。



このユーザー・インターフェースの主領域について、以下に説明します。

### タスクバー

すべてのユーザーは、タスクバーを使用して、 Business Process Choreographer Explorer からログアウトしたり、オンライン・ヘルプにアクセスしたりすることができます。また、「代理人」オプションおよび「代理人の定義」オプションで、不在設定を指定することもできます。これらのオプションは、 Business Process Choreographer で Human Task Manager に対して代替が有効になっていて、WebSphere Application Server セキュリティーで Virtual Member Manager サービスが構成されている場合に選択できます。

**代理人** このオプションは、ユーザーのタスクに対して代理人を指定するために選択します。

#### 代理人の定義

このオプションは、ユーザーの不在設定を定義する場合に選択します。

システム管理者権限を保持している場合には、タスクバーに以下のオプションも組み込まれます。

#### カスタマイズ

このオプションは、 Business Process Choreographer Explorer のこのインスタンスのナビゲーション・ペインにビューを追加したり、ナビゲーション・ペインからビューを除去したりするために選択します。ユーザーがログインしたときに、ユーザーに対して表示されるビューを定義することもできます。

#### ビューの定義

このオプションは、ユーザー・グループのカスタマイズ・ビューを定義する場合に選択します。

## ナビゲーション・ペイン

「ビュー」タブを選択すると、ナビゲーション・ペインには、オブジェクト (例えば、開始済みのプロセス・インスタンス、あるいは、管理する許可が与えられているヒューマン・タスク) の管理に使用するビューへのリンクが表示されます。デフォルトのユーザー・インターフェースには、ビジネス・プロセスやタスクの事前定義ビューへのリンクがあります。

システム管理者は、ナビゲーション・ペインに対して事前定義ビューの追加および除去を行い、カスタム・ビューを定義してナビゲーション・ペインに追加することにより、ナビゲーション・ペインのコンテンツをカスタマイズできます。すべてのユーザーは、ナビゲーション・ペインから個別設定ビューを定義できます。

「レポート」タブを選択すると、ナビゲーション・ペインには、作成するレポートの種類 (例えば、アクティビティ・インスタンスのデータを図表に表示できるレポート) を選択するために使用するリンクが表示されます。ランタイム・エンティティの状態やイベント情報 (例えば、プロセスやアクティビティのスナップショット・グラフ) を取得するには、事前に定義したリストや図表を使用します。

「レポート」タブは、レポート作成が構成されている場合のみ表示されます。レポート作成機能は、Business Process Choreographer Explorer の構成時に構成できますが、後で構成することもできます。

## ページ・タイトル

「ビュー」タブを選択すると、ワークスペースには、ビジネス・プロセスおよびヒューマン・タスク関連オブジェクトを表示および管理するために使用するページが表示されます。これらのページにアクセスするには、ナビゲーション・ペインのリンクをクリックする方法、アクション・バーのアクションをクリックする方法、またはワークスペース・ページ内のリンクをクリックする方法があります。ページについて詳しくは、それぞれのページの「ヘルプ」アイコン  をクリックしてください。

「レポート」タブを選択すると、ワークスペースには、事前定義のリストや図表を表示したり、レポート定義を指定したり、レポートを表示したりするためのページが表示されます。これらのページにアクセスするには、ナビゲーション・ペインのリンクをクリックする方法、アクション・バーのアクションをクリックする方法、またはワークスペース・ページ内のリンクをクリックする方法があります。ページについて詳しくは、それぞれのページの「ヘルプ」アイコン  をクリックしてください。

### 関連資料

 Business Process Choreographer Explorer レポート作成機能および Event Collector の構成

Business Process Choreographer Explorer レポート作成機能の使用はオプションです。しかし、使用するには、事前にデータベースをセットアップし、アプリケーションをインストールしておく必要があります。

## Business Process Choreographer Explorer の「ビュー」タブ

Business Process Choreographer Explorer の「ビュー」タブを使用すると、ビジネス・プロセスおよびヒューマン・タスク・オブジェクト（プロセス・インスタンスや作業割り当てなど）の管理に使用するビューにアクセスできます。デフォルトのユーザー・インターフェースには、ビジネス・プロセスやタスクの事前定義ビューへのリンクがあります。ナビゲーション・ペインに追加する専用の個別設定ビューを定義することもできます。さらに、システム管理者は、すべてのユーザーが使用できるカスタマイズ・ビューを定義できます。

### 使用可能なアクション

ナビゲーション・ペインでは、以下のアクションを使用できます。

- グループを縮小表示または展開する。

ナビゲーション・ペイン内のある項目を展開または縮小表示するには、その項目の横にある矢印をクリックします。

- ビューにナビゲートする。

あるビューにナビゲートするには、そのビュー名をクリックします。

- 新規の検索を定義する。

オブジェクトの検索や個別設定ビューの定義を行うには、「**新規検索**」アイコン () をクリックします。

ビューの種類に応じて、その他のアクションをポップアップ・メニューから選択できます。「**ポップアップ・メニューの表示**」アイコン () は、ポップアップ・メニューが使用可能であることを示します。

- ビューを削除するには、「**削除**」アイコン () をクリックします。
- ビューを変更するには、「**編集**」アイコン () をクリックします。
- ビューのコピーを作成して、コピーを変更するには、「**コピー**」アイコン () をクリックします。
- リスト内でのビューの位置を上下に移動するには、「**上へ**」アイコン () または「**下へ**」アイコン () をクリックします。

### ビューの種類

ナビゲーション・ペインには、以下の種類のビューを収容できます。ビューによっては、ポップアップ・メニューからその他のアクションを選択できます。

#### デフォルトのナビゲーション・ペイン内の事前定義ビュー

これらのビュー・グループはナビゲーション・ペインで使用できますが、最初はポップアップ・メニューに表示されません。「**カスタマイズ**」を使用し

てナビゲーション・ペインが変更されると、これらの事前設定ビューの前に「事前定義ビュー」アイコン  が表示され、ビューを上下に移動できます。

### システム管理者によりナビゲーション・ペインに追加されたカスタマイズ・ビューおよび事前定義ビュー

ビジネス・ユーザーは、ビュー名をクリックしてビューまでナビゲートできます。システム管理者の場合は、ポップアップ・メニューを使用できます。

- これらの事前定義ビューは、「事前定義ビュー」アイコン  で示されます。システム管理者は、ポップアップ・メニューを使用して、ナビゲーション・ペイン内でのこれらのビューの位置を変更できます。
- カスタマイズ・ビューは、「カスタム・ビュー」アイコン  で示されます。システム管理者は、これらのビューの削除、編集、コピー、および移動を実行できます。

### 個別設定ビュー

これらのビューは、「カスタム・ビュー」アイコン  で示されます。これらのビューは、これらのビューを作成したユーザーに対してのみ表示されます。ユーザーは、これらのビューの削除、編集、コピー、および移動を実行できます。

### ナビゲーション・ペインの事前定義ビュー

デフォルトのナビゲーション・ペインには、以下のビュー・グループがあります。ご使用の Business Process Choreographer Explorer のナビゲーション・ペインに表示されるビューは、システム管理者がナビゲーション・ペインに対してビューを追加または削除したかどうかによって異なります。すべてのビューでは、追加フィルターとは関係なく、許可された項目が表示されます。例えば、強制終了したプロセスのうち、表示が許可されたものだけが表示されます。ビューのグループにビューが定義されていない場合、そのグループは表示されません。

#### プロセス・テンプレート

プロセス・テンプレート・グループには、以下のビューがあります。

##### プロセス・テンプレート

このビューには、プロセス・テンプレートのリストが表示されます。このビューからは、プロセス・テンプレートとその構造に関する情報の表示、テンプレートに関連付けられているプロセス・インスタンスのリストの表示、およびプロセス・インスタンスの開始を実行できます。

#### プロセス・インスタンス

プロセス・インスタンス・グループには、以下のビューがあります。

##### ユーザーが開始

このビューには、開始したプロセス・インスタンスが表示されます。このビューからは、プロセス・インスタンスの進行状況のモニター、プロセス・インスタンスに関連するアクティビティー、プロセス、またはタスクのリスト表示を実行できます。

### ユーザーが管理

このビューには、管理する権限があるプロセス・インスタンスが表示されます。このビューからは、プロセス・インスタンスを処理できます。例えば、プロセスの中断および再開、またはプロセス・インスタンスでのアクティビティの進行状況のモニターを実行できます。

### 重大なプロセス

このビューには、実行状態のプロセス・インスタンスのうち、停止状態のアクティビティを含むものが表示されます。このビューからは、プロセス・インスタンスを処理したり、アクティビティのリストを表示後にアクティビティを処理したりすることができます。

### 強制終了したプロセス

このビューには、強制終了状態のプロセス・インスタンスが表示されます。このビューからは、これらのプロセス・インスタンスを処理できます。

### 失敗した補正

このビューには、microflow について失敗した補正アクションが表示されます。

### アクティビティ・インスタンス

アクティビティ・インスタンス・グループには、以下のビューがあります。

#### 失敗したアクティビティ

このビューには、失敗状態にあるアクティビティが表示されません。停止状態のアクティビティは含まれません。

### タスク・テンプレート

タスク・テンプレート・グループには、以下のビューがあります。

#### ユーザーのタスク・テンプレート

このビューには、タスク・テンプレートのリストが表示されます。このビューからは、タスク・インスタンスの作成および開始、テンプレートに関連付けられているタスク・インスタンスのリストの表示を実行できます。

### タスク・インスタンス

タスク・インスタンス・グループには、以下のビューがあります。

#### ユーザーの予定

このビューには、操作する権限があるタスク・インスタンスのリストが表示されます。このビューからは、タスク・インスタンスの処理、要求したタスク・インスタンスのリリース、または他のユーザーへのタスク・インスタンスの転送を実行できます。タスクの優先順位の変更と、タスクのビジネス・カテゴリーの変更もできます。

#### すべてのタスク

このビューには、お客様が所有者、潜在的所有者、編集者のいずれかになっているすべてのタスクが表示されます。このビューからは、タスク・インスタンスの処理、要求したタスク・インスタンス

のリリース、または他のユーザーへのタスク・インスタンスの転送を実行できます。タスクの優先順位の変更と、タスクのビジネス・カテゴリの変更もできます。

#### ユーザーが開始

このビューには、開始したタスク・インスタンスが表示されます。このビューからは、タスク・インスタンスの処理、要求したタスク・インスタンスのリリース、または他のユーザーへのタスク・インスタンスの転送を実行できます。タスクの優先順位の変更と、タスクのビジネス・カテゴリの変更もできます。

#### ユーザーが管理

このビューには、管理する権限があるタスク・インスタンスが表示されます。このビューからは、タスク・インスタンスを処理できます。例えば、プロセスの中断および再開、タスク・インスタンスの作業項目の作成、またはタスク・インスタンスの現在の作業項目リストの表示を実行できます。タスクの優先順位の変更と、タスクのビジネス・カテゴリの変更もできます。

#### ユーザーのエスカレーション

このビューには、ログオンしているユーザーのすべてのエスカレーションが表示されます。

## Business Process Choreographer Explorer の「レポート」タブ

Business Process Choreographer Explorer の「レポート」タブを使用して、Business Process Choreographer で処理された特定のプロセスおよびアクティビティーについてのレポートを管理します。作成するレポートの種類（プロセス・レポートやアクティビティー・レポートなど）を選択することができます。独自のレポート定義を保管して、これらの定義をナビゲーション・ペインに追加することもできます。ドリルダウン方式の事前定義のリストおよび図表は、ランタイム・エンティティーの状態およびイベント情報を入手するために使用します。例えば、リスト、プロセスおよびアクティビティーのスナップショット・グラフ、期間グラフによるプロセス・インスタンスおよびアクティビティー・インスタンスが選択できます。「レポート」タブは、レポート作成が構成されている場合にのみ表示されます。レポート作成機能は、Business Process Choreographer Explorer の構成時に構成できますが、後で構成することもできます。

### 使用可能なアクション

ナビゲーション・ペインでは、以下のアクションを使用できます。

- グループを縮小表示または展開する。

ナビゲーション・ペイン内のある項目を展開または縮小表示するには、その項目の横にある矢印をクリックします。

- 事前定義のリストまたはグラフにナビゲートする。

レポートを作成するインスタンスの種類をクリックします。

- プロセス・レポート・ウィザードまたはアクティビティー・レポート・ウィザードへナビゲートする。

レポートのタイプ、レポートの内容、レポートのフィルター基準を指定するには、「新規レポート」アイコン (  ) をクリックします。

- 保存済みのプロセス・レポートまたはアクティビティ・レポートを実行する。

レポートを実行するには、レポート名をクリックします。

- 保存済みのプロセス・レポート定義またはアクティビティ・レポート定義のポップアップ・メニューを開く。

保存済みのレポート定義を操作するには、「ポップアップ・メニューの表示」アイコン (  ) をクリックします。

- レポート定義を削除するには、「削除」アイコン (  ) をクリックします。
- レポート定義を編集するには、「編集」アイコン (  ) をクリックします。
- レポート定義をコピーするには、「コピー」アイコン (  ) をクリックします。
- レポート結果をエクスポートするには、「エクスポート」アイコン (  ) をクリックします。
- レポートを非同期で実行するには、「非同期レポート (Asynchronous Report)」アイコン (  ) をクリックします。
  - 非同期レポートが正常に完了すると、ナビゲーション・ペインに「非同期レポート完了 (Asynchronous Report Completed)」アイコン (  ) が表示されます。レポートの名前をクリックして、結果を表示します。
  - 非同期レポートが正常に完了しなかった場合は、「非同期レポート失敗 (Asynchronous Report Failed)」アイコン (  ) が表示されます。

## ナビゲーション・ペインの事前定義リストおよびグラフ

ナビゲーション・ペインには、以下に示す事前定義リストおよびグラフのグループがあります。

**リスト** このグループには、以下のリストがあります。

### プロセス

このリストは、指定した期間内にプロセス・イベントを出力したプロセスを表示する場合に使用します。プロセスのリストは、プロセス状態に応じて表示されます。

### アクティビティ

このリストは、選択したアクティビティが、指定した期間内に到達した状態を表示する場合に使用します。アクティビティのリストは、アクティビティ状態に応じて表示されます。

### ユーザー

このリストは、指定した期間内に、選択したユーザーが実行したアクティビティと、そのアクティビティが到達した状態を表示する場合に使用します。アクティビティは、その状態に応じて表示されます。対応するユーザーは、アクティビティごとに表示されます。

**グラフ** このグループには、以下のグラフがあります。

#### プロセス・スナップショット

このグラフは、指定した時刻に状態の異なるプロセス・インスタンスの数を調べる場合に使用します。データは棒グラフまたは円グラフで表示できます。

#### 期間によるプロセス

このグラフは、指定した期間内に指定した状態に到達したプロセス・インスタンスの分布を調べる場合に使用します。各インスタンスは、それが指定の状態に到達したタイム・スライス内に表示されます。データは折れ線グラフ、棒グラフ、または円グラフで表示できます。

#### アクティビティ・スナップショット

このグラフは、指定した時刻に状態の異なるアクティビティ・インスタンスの数を調べる場合に使用します。データは棒グラフまたは円グラフで表示できます。

#### 期間によるアクティビティ

このグラフは、指定した期間内に指定した状態に到達したアクティビティ・インスタンスの分布を調べる場合に使用します。各インスタンスは、それが指定の状態に到達したタイム・スライス内に表示されます。データは折れ線グラフ、棒グラフ、または円グラフで表示できます。

### プロセス・レポートおよびアクティビティ・レポート

ナビゲーション・ペインには、以下のレポート・ウィザードへのリンクがあります。

レポート・ウィザードは「**新規レポート**」アイコン (  ) で示されます。

#### プロセス・レポート

プロセス・レポートは、プロセス・インスタンス・イベントを照会する場合に使用します。これらのイベントには、プロセス・インスタンスの状態変更が記述されています。作成するレポートのデータを定義する場合は、レポート・ウィザードを使用してください。レポート定義を保管して検索することができます。

#### アクティビティ・レポート

アクティビティ・インスタンス・イベントを照会する場合は、アクティビティ・レポートを使用します。これらのイベントには、アクティビティ・インスタンスの状態変更が記述されています。個々のレポートを指定する場合は、レポート・ウィザードを使用します。レポート定義を保管して検索することができます。

#### 関連タスク

43 ページの『Business Process Choreographer Observer 入門』

ビジネス・プロセスおよびタスクの実行中に、WebSphere はプロセス・インスタンスとそれに関連するアクティビティーの状態変更に関する情報が入ったイベントを発行できます。Business Process Choreographer Observer を使用すると、それらのイベントに基づいた統計情報を取り出し、プロセスとアクティビティーに関するレポートを作成できます。

### 関連資料



Business Process Choreographer Explorer レポート作成機能および Event Collector の構成

Business Process Choreographer Explorer レポート作成機能の使用はオプションです。しかし、使用するには、事前にデータベースをセットアップし、アプリケーションをインストールしておく必要があります。

## Business Process Choreographer Explorer の開始

Business Process Choreographer Explorer は、ビジネス・プロセス・コンテナの構成の一部としてインストールできる Web アプリケーションです。Web ブラウザーから Business Process Choreographer Explorer の使用を開始するには、前もってビジネス・プロセス・コンテナ、ヒューマン・タスク・コンテナ、および Business Process Choreographer Explorer アプリケーションのインストールが完了済みで、さらにこのアプリケーションが稼働している必要があります。レポート作成機能を使用するには、Event Collector アプリケーションがインストールされ、動作している必要があります。

### このタスクについて

Business Process Choreographer Explorer を開始するには、以下の手順を実行します。

#### 手順

1. Web ブラウザーで、Business Process Choreographer Explorer の URL にアクセスします。

この URL は、次の形式をとります。URL の値は、使用するシステムで仮想ホストとコンテキスト・ルートがどのように構成されているかによって異なります。

```
http://app_server_host:port_no/context_root
```

各部の意味は、次のとおりです。

*app\_server\_host*

作業に使用するビジネス・プロセス・アプリケーションを提供するアプリケーション・サーバーのホストのネットワーク名。

*port\_no*

Business Process Choreographer Explorer が使用するポート番号。ポート番号はシステム構成によって異なります。デフォルト・ポート番号は 9080 です。

*context\_root*

Business Process Choreographer Explorer アプリケーションの、アプリケーション・サーバー上のルート・ディレクトリー。デフォルトは `bpc` です。

2. セキュリティーが有効になっている場合は、ユーザー ID とパスワードを入力して、「ログイン」をクリックする必要があります。

## タスクの結果

Business Process Choreographer Explorer の最初のページが表示されます。デフォルトでは、このページには「ユーザーの予定」ビューが表示されます。

## Business Process Choreographer Explorer のカスタマイズ

Business Process Choreographer Explorer は、管理者がビジネス・プロセスとヒューマン・タスクを管理し、ビジネス・ユーザーが割り当てられたタスクを処理するためのユーザー・インターフェースを提供します。これは汎用インターフェースであるため、特定の Business Process Choreographer Explorer インスタンスに割り当てられたユーザー・グループのビジネス・ニーズに合わせて、インターフェースをカスタマイズする必要がある場合があります。さらに、構成時に (または後で)、ユーザーは、レポート作成機能を追加して、プロセスとアクティビティーに関するレポートを作成したり、イベントに基づく統計情報を取得したりすることもできます。

### このタスクについて

このユーザー・インターフェースは、さまざまな方法でカスタマイズできます。

### さまざまなユーザー・グループ用の Business Process Choreographer Explorer インターフェースのカスタマイズ

デフォルトの Business Process Choreographer Explorer ユーザー・インターフェースのナビゲーション・ペインには、事前定義ビューへの一連のリンクが入っています。「ユーザーの予定」ビューは、ログイン後に表示される「ビュー」タブのデフォルト・ビューです。Business Process Choreographer のシステム管理者ロールのいずれかを持っている場合は、タスクバーの「カスタマイズ」を使用して、ナビゲーション・ペインに表示されるリンク、およびユーザーのログイン後に表示されるビューをカスタマイズできます。また、「ビューの定義」を使用すると、ナビゲーション・ペインで情報、フィルター、およびソート基準を指定してユーザーに表示されるビューと、ビュー内で必要なアクションを定義できます。

### 始める前に

インターフェースをカスタマイズするには、Business Process Choreographer のシステム管理者になる必要があります。

### このタスクについて

例えば、Business Process Choreographer Explorer のデフォルトのユーザー・インターフェースには、ビジネス・ステート・マシンで作業するためのビューが含まれていません。その場合、ビジネス・ステート・マシンのプロセス・テンプレートおよびプロセス・インスタンスを操作するための事前定義ビューを追加できます。

あるいは、カスタマー・オーダーを処理するユーザーに、カスタマー・サービスの照会を処理するユーザーとは異なるインターフェースを提供する必要がある場合があります。Business Process Choreographer Explorer のインスタンスを、そのインスタンスに割り当てられたユーザーのワークフロー・パターンに合うようにカスタマイズできます。

Business Process Choreographer Explorer のデフォルトのユーザー・インターフェースをカスタマイズするには、以下の手順を実行します。

#### 手順

1. ナビゲーション・ペイン内の一連のビューとデフォルトのログイン・ビューをカスタマイズします。
  - a. タスクバーで「**カスタマイズ**」をクリックします。
  - b. 「ナビゲーション・ツリーおよびログイン・ビューのカスタマイズ」ページで、組み込むビューを選択し、ナビゲーション・ペインから除去するビューの選択を解除します。
  - c. ユーザーが Business Process Choreographer Explorer にログインしたときに表示されるビューを選択します。

リストには前のステップで選択したビューと、「カスタマイズ・ビューの検索および定義」ページで作成したカスタマイズ済みビュー (手順 2 を参照) が含まれています。

- d. 変更を保存するには、「**保管**」をクリックします。

変更を保管すると、各事前定義ビューがナビゲーション・ペイン内に表示され、ビューの前にはアイコンが付きます。これらのアイコンを使用すれば、事前定義ビューをリスト内で上下に移動させることができます。

このインスタンスのビューをデフォルトのビューに戻すには、「**デフォルトの復元**」をクリックします。このアクションは、ナビゲーション・ペインを事前定義ビューのリストにリセットします。ナビゲーション・ペイン内のカスタマイズされたビューは、このアクションの影響を受けません。

2. ビューをカスタマイズします。

この Business Process Choreographer Explorer インスタンスのビューに表示される情報を指定できます。

- a. タスクバーで「**ビューの定義**」をクリックします。
- b. 「カスタマイズ・ビューの検索および定義」ページで、例えばプロセス・テンプレートなど、カスタマイズしたいビューのタイプを選択します。
- c. 「XXX の検索およびカスタマイズ・ビューの定義」ページ (XXX はビューのタイプで、例えば「プロセス・テンプレート」) で、検索基準を指定します。

「プロセスの基準」タブ、「タスクの基準」タブ、および「プロパティ・フィルター」タブを使用して、検索結果を限定します (例えば、特定のプロセス・テンプレートに限定)。また、インスタンス・ビューの定義時に、「ユーザー役割」タブを使用して、検索結果をユーザー、グループ、または役割に限定することもできます。

- d. 「プロパティの表示」タブを使用して、ビューに組み込む順序付けプロパティや結果のしきい値などのリスト列とリスト・プロパティを選択します。

また、「ビューの設定」で、ビューのアクション・バーに追加するアクションを指定できます。ビューに組み込むアクション、または実行しようとしている検索を選択するには、以下を実行します。

- 「使用可能なアクション」で、1 つまたは複数のアクションを選択して「追加」をクリックします。
- アクションを削除するには、「ビューのアクション」内でアクションを選択して、「削除」をクリックします。
- アクション・バー内のアクションの順序は、アクションを「ビューのアクション」内で上下に移動させて指定できます。

これが、タスク、プロセス、またはアクティビティ・インスタンス・ビューの場合は、「ビューの設定」をクリックして、システム管理者およびシステム・モニター用のビューに組み込まれる項目を指定します。

- システム管理者およびシステム・モニターの場合は、検索結果を各自のインスタンスに制限できます。
    - 検索基準に一致するすべての項目をビューに表示するには、「すべてのインスタンス」を選択します。システム管理者が当該の項目についての作業項目を持っているかどうかに関係なく、すべての項目が表示されます。
    - ログオン・ユーザーが作業項目を持っている項目だけを表示するには、「個人用インスタンス」を選択します。
- e. 「ビュー名」フィールドにビューの表示名を入力して、「保管」をクリックします。

「要約」タブを使用して、現在ビューに設定されている内容を確認します。

新しいビューがナビゲーション・ペインに表示されます。ユーザーが次回 Business Process Choreographer Explorer にログオンした際に、この新しいビューが表示されます。ビューは、ナビゲーション・ペイン内で上下に移動させることができます。

#### ビジネス・ステート・マシンのプロセス・テンプレート用ビューの定義:

ビジネス・ステート・マシンのプロセス・テンプレートには事前定義ビューが用意されていますが、このタイプのテンプレート用に独自のビューを定義することもできます。

#### 始める前に

カスタマイズ・ビューを作成するには、システム管理者のいずれかのロールが必要です。

## このタスクについて

Business Process Choreographer Explorer で次のステップを実行します。

### 手順

1. タスクバーで「ビューの定義」をクリックします。
2. 「カスタマイズ・ビューの検索および定義」ページで、「プロセス・テンプレートの検索およびカスタマイズ・ビューの定義」を選択します。
3. 「プロパティ・フィルター」 → 「カスタム・プロパティ・フィルター」をクリックします。
  - a. 以下の設定でカスタム・プロパティを追加します。
    - 「プロパティ名」フィールドに generatedBy と入力します。
    - 「プロパティ値」フィールドに BusinessStateMachine と入力します。
  - b. 「追加」をクリックします。
  - c. 必要に応じて他のカスタム・プロパティを追加します。
4. 「プロパティの表示」 → 「リスト列」をクリックします。
  - a. カスタム・プロパティのリスト列で、以下の設定により、カスタム・プロパティを追加します。
    - 「プロパティ名」フィールドに generatedBy と入力します。
    - 「表示名」フィールドに列の表示名を入力し、「追加」をクリックします。
  - b. 他の列を追加するか、選択した列のリストから列を除去します。
5. 照会の表示名を「ビュー名」フィールドに入力し、「保管」をクリックします。

### タスクの結果

デフォルトでは、新しいビューへのリンクがナビゲーション・ペインの「プロセス・テンプレート」グループに追加されます。ユーザーが次回に Business Process Choreographer Explorer にログインすると、このビューが表示されます。

### ビジネス・ステート・マシンのプロセス・インスタンス用ビューの定義:

ビジネス・ステート・マシンのプロセス・インスタンスには、事前定義ビューが用意されていますが、このタイプのプロセス・インスタンス用に独自のビューを定義することもできます。

### 始める前に

カスタマイズ・ビューを作成するには、システム管理者のいずれかのロールが必要です。

## このタスクについて

Business Process Choreographer Explorer で次のステップを実行します。

### 手順

1. タスクバーで「ビューの定義」をクリックします。

2. 「カスタマイズ・ビューの検索および定義」 ページで、「プロセス・インスタンスの検索およびカスタマイズ・ビューの定義」を選択します。
3. 「カスタム・プロパティ・フィルター」 → 「カスタム・プロパティ・フィルター」をクリックします。
  - a. 以下の設定でカスタム・プロパティを追加します。
    - 「プロパティ名」フィールドに `generatedBy` と入力します。
    - 「プロパティ値」フィールドに `BusinessStateMachine` と入力します。
  - b. 「追加」をクリックします。
  - c. 必要に応じて他のカスタム・プロパティを追加します。
4. 「プロパティの表示」 → 「リスト列」をクリックします。
  - a. 照会プロパティのリスト列で、以下の照会プロパティを追加します。
    - ビューにビジネス状態情報を追加するには、「プロパティ名」フィールドに `name`、「変数名」フィールドに `DisplayState`、「ネーム・スペース」フィールドに `tns` と入力します。ここで、`tns` は `-process` のサフィックスが付けられたビジネス・ステート・マシンのターゲット名前空間です。また、「表示名」フィールドに列の表示名も指定し、「追加」をクリックします。
    - ビューに関連情報を追加するには、「プロパティ名」フィールド、「変数名」フィールド、および「ネーム・スペース」フィールドに該当する情報を入力します。これらの値は、ビジネス・ステート・マシンの定義から派生します。また、「表示名」フィールドで列の表示名も指定します。

#### プロパティ名

ビジネス・ステート・マシンに定義した関連プロパティの名前。

**変数名** 関連セットが着信パラメーターによって開始される場合、変数名は以下の形式になります。

`operation_name_Input_operation_parameter_name`

ここで、`operation_name` は、初期状態からの遷移操作の名前です。

関連セットが発信パラメーターによって開始される場合、変数名は以下の形式になります。

`operation_name_Output_operation_parameter_name`

#### ネーム・スペース

照会プロパティの名前空間。ここで、`tns` は `-process` のサフィックスが付けられたビジネス・ステート・マシンのターゲット名前空間です。

- b. 他のカスタム・プロパティまたは照会プロパティを追加するか、選択した列のリストに列を追加するか、リストから列を除去します。
5. 照会の名前を「ビュー名」フィールドに入力し、「保管」をクリックします。

#### タスクの結果

デフォルトでは、新しいビューへのリンクがナビゲーション・ペインの「プロセス・インスタンス」グループに追加されます。ユーザーが次回に `Business Process Choreographer Explorer` にログインすると、このビューが表示されます。

## Business Process Choreographer Explorer インターフェースの個別設定

デフォルトの Business Process Choreographer Explorer ユーザー・インターフェースのナビゲーション・ペインには、事前定義ビューおよびシステム管理者が定義したビューへの一連のリンクが入っています。ナビゲーション・ペインには、ユーザーのロールに関係なく、独自のビューを追加することができます。例えば、特定のタスクまたはプロセスの進行状況をモニターするために、新規のビューを追加することができます。ビューに表示される情報と、ビューで提供されるフィルター、ソート基準、およびアクションを指定できます。

### このタスクについて

ユーザー・インターフェースの個別設定を行うには、Business Process Choreographer Explorer で以下の手順を実行します。

#### 手順

1. 新しいビューを定義する「ビュー」タブのナビゲーション・ペインのセクション (例: 「プロセス・テンプレート」) で、「新規検索」アイコン (  ) をクリックします。
2. ビューの「... の検索および個別設定ビューの定義 (Search For ... and Define Personalized Views)」ページ (例: 「プロセス・テンプレートの検索および個別設定ビューの定義」) で、検索基準を指定します。

「プロセスの基準」タブ、「タスクの基準」タブ、および「プロパティ・フィルター」タブを使用して、検索結果を限定します (例えば、特定のプロセス・テンプレートに限定)。また、インスタンス・ビューの定義時に、「ユーザー役割」タブを使用して、検索結果をユーザー、グループ、または役割に限定することもできます。

3. 「プロパティの表示」タブを使用して、ビューに組み込む順序付けプロパティや結果のしきい値などのリスト列とリスト・プロパティを選択します。

また、「ビューの設定」で、ビューのアクション・バーに追加するアクションを指定できます。ビューに組み込むアクション、または実行しようとしている検索を選択するには、以下を実行します。

- 「使用可能なアクション」で、1 つまたは複数のアクションを選択して「追加」をクリックします。
- アクションを削除するには、「ビューのアクション」内でアクションを選択して、「削除」をクリックします。
- アクション・バー内のアクションの順序は、アクションを「ビューのアクション」内で上下に移動させて指定できます。

これが、タスク、プロセス、またはアクティビティ・インスタンス・ビューの場合は、「ビューの設定」をクリックして、システム管理者およびシステム・モニター用のビューに組み込まれる項目を指定します。システム管理者またはシステム・モニターであれば、検索結果を自分自身のインスタンスに限定できます。

- ・ 検索基準に一致するすべての項目をビューに表示するには、「すべてのインスタンス」を選択します。システム管理者が当該の項目についての作業項目を持っているかどうかに関係なく、すべての項目が表示されます。
  - ・ ログオン・ユーザーが作業項目を持っている項目だけを表示するには、「個人用インスタンス」を選択します。
4. 「ビュー名」フィールドにビューの表示名を入力して、「保管」をクリックします。

「要約」タブを使用して、現在ビューに設定されている内容を確認します。

## タスクの結果

新しいビューがナビゲーション・ペインに表示されます。

## デフォルトの Web アプリケーションの外観の変更

Business Process Choreographer Explorer は、JavaServer Pages (JSP) ファイルと JavaServer Faces (JSF) コンポーネントに基づいた、すぐに使用できる Web ユーザー・インターフェースを備えています。この Web インターフェースのレンダリング方法は、カスケーディング・スタイル・シート (CSS) によって制御されます。このスタイル・シートを変更して、ユーザー・インターフェースを特定のルック・アンド・フィールに合わせることができ、新たにコードを記述する必要はありません。

### 始める前に

スタイル・シートを変更するには、カスケーディング・スタイル・シートに関する深い知識が必要です。

### このタスクについて

CSS の変更は、例えば、デフォルト・インターフェースをコーポレート・アイデンティティのガイドラインに合わせるために行うことができます。

### 手順

スタイル・シートを変更します。デフォルトのスタイル・シートである `style.css` には、ヘッダー、ナビゲーション・ペイン、およびコンテンツ・ペイン内に置かれるエレメントのスタイルが入っています。

#### 関連概念

23 ページの『Business Process Choreographer Explorer のユーザー・インターフェース』

Business Process Choreographer Explorer は、ビジネス・プロセスおよびヒューマン・タスクを管理し、プロセスおよびアクティビティ・イベントに関するレポートを作成するための管理機能のセットを提供するスタンドアロン Web アプリケーションです。このインターフェースは、タスクバー、ナビゲーション・ペイン、およびワークスペースで構成されます。

**Business Process Choreographer Explorer のインターフェースで使用されるスタイル:**

style.css ファイルには、デフォルトのユーザー・インターフェースのルック・アンド・フィールを適合させるために変更できるスタイルが収録されています。

style.css ファイルには、以下に示すデフォルトのユーザー・インターフェースの要素のスタイルが収録されています。

- 『バナー』
- 『フッター』
- 『メニュー・バー』
- 41 ページの『ログイン・ページ』
- 41 ページの『ナビゲーター』
- 41 ページの『コンテンツ・パネル』
- 42 ページの『コマンド・バー』
- 42 ページの『リスト』
- 42 ページの『詳細パネル』
- 42 ページの『メッセージ・データ』
- 42 ページの『タブ付きペイン』
- 43 ページの『検索ページ』
- 43 ページの『エラー詳細』

このファイルは、次のディレクトリーにあります。

```
<profile_root>%installedApps%<node_name>%<explorer_instance>%bpcexplorer.war%theme
```

### バナー

スタイル名	説明
.banner	バナーの分割線。
.banner_left	バナー内の分割線。アプリケーションのタイトル画像を埋め込むときに使用します。
.banner_right	バナー内の分割線。例えば、ロゴを追加して表示するときに使用できます。

### フッター

スタイル名	説明
.footer	フッターの分割線。
.footer_left	フッター内の分割線。例えば、アプリケーションの会社ロゴを表示する場合に使用できます。
.footer_right	フッター内の分割線。例えば、ロゴを追加して表示するときに使用できます。

### メニュー・バー

スタイル名	説明
.menubar	JSF サブビュー。

スタイル名	説明
.menuContainer	ラベルやリンクなどのメニュー項目を含むコンテナ・パネル。
.menuItem	メニュー・バーに表示される項目。

## ログイン・ページ

スタイル名	説明
.loginPanel	ログイン・フォームを含むパネル。
.loginTitle	フォームのタイトル。
.loginText	手順を示すテキスト。
.loginForm	入力コントロールが組み込まれているフォーム。
.loginValues	コントロールのレイアウトを決定するテーブル。
.loginField	ログオン・フィールドに使用されるラベル (「名前」や「パスワード」など)。
.loginValue	テキスト入力フィールド。

## ナビゲーター

スタイル名	説明
.pageBodyNavigator	ナビゲーターが表示される領域。
.navigator	ナビゲーターの JSF サブビュー。リストへのリンクが収録されている。
.navigatorTitle	各ナビゲーター・ボックスのタイトル。
.taskNavigatorTitle	あるクラスに属する、ナビゲーション・ボックスのタイトル。これらは、ビジネス・プロセス・オブジェクトのリストへのリンクとヒューマン・タスク・オブジェクトのリストへのリンクを区別するときに使用します。
.navigatorFrame	各ナビゲーター・ボックスの分割線 (例えば、境界線を引くための線)。
.navigatorLink	ナビゲーター・ボックス内のリンク。
.expanded	ナビゲーター・ボックスが展開されている場合に使用される。
.collapsed	ナビゲーター・ボックスが縮小表示されている場合に使用される。

## コンテンツ・パネル

スタイル名	説明
.pageBodyContent	コンテンツが表示される領域。
.panelContainer	リスト、詳細、メッセージのいずれかが入っている分割パネル。
.panelTitle	表示されるコンテンツのタイトル。例えば、「ユーザーの予定」。

スタイル名	説明
.panelHelp	ヘルプ・テキストおよびアイコンが入っている分割コンテナ。
.panelGroup	コマンド・バー、コマンド・リスト、詳細、またはメッセージが入っている分割コンテナ。

## コマンド・バー

スタイル名	説明
.commandbar	コマンド・バー領域を囲む分割コンテナ。
.button	コマンド・バーのボタンに使用されるスタイル。

## リスト

スタイル名	説明
.list	行のある表。
.listHeader	リストの見出し行に使用されるスタイル。
.ascending	リストをこの列で昇順にソートする場合のリスト・ヘッダー・クラスのスタイル。
.descending	リストをこの列で降順にソートする場合のリスト・ヘッダー・クラスのスタイル。
.unsorted	リストをこの列でソートしない場合のリスト・ヘッダー・クラスのスタイル。

## 詳細パネル

スタイル名	説明
.details	詳細パネルを囲む分割コンテナ。
.detailsProperty	プロパティ名のラベル。
.detailsValue	プロパティ値のテキスト。

## メッセージ・データ

スタイル名	説明
.messageData	メッセージを囲む分割コンテナ。
.messageDataButton	メッセージ・フォームでの「追加」ボタンおよび「除去」ボタンのボタン・スタイル。
.messageDataOutput	レンダリングの読み取り専用テキストに使用。
.messageDataValidInput	有効なメッセージ値に使用。
.messageDataInvalidInput	無効なメッセージ値に使用。

## タブ付きペイン

スタイル名	説明
.tabbedPane	すべてのタブ付きペインを囲む分割コンテナ。

スタイル名	説明
.tabHeader	タブ付きペインのタブ・ヘッダー。
.selectedTab	アクティブなタブ・ヘッダー。
.tab	非アクティブのタブ・ヘッダー。
.tabPane	タブ付きペインを囲む分割コンテナ。
.tabbedPaneNested	検索ページで使用されるネストされたタブ付きペインを囲む分割コンテナ。
.tabHeaderSimple	ネストされたタブ付きペインのタブ・ヘッダー。
tabHeaderProcess	プロセス・フィルター用のネストされたタブ付きペインのタブ・ヘッダー。
.tabHeaderTask	タスク・フィルター用のネストされたタブ付きペインのタブ・ヘッダー。
.tabPaneSimple	ネストされたタブ付きペインを囲む分割コンテナ。

### 検索ページ

スタイル名	説明
.searchPane	検索パネル用のタブ付きペイン。タブ付きペインも参照。
.searchPanelFilter	検索書式のテーブル・コンテナ。
.searchLabel	検索書式コントロールのラベル。
.summary	検索要約ペインを囲むコンテナ。
.summaryTitle	検索要約ペインのすべてのタイトルの共通スタイル。
.summaryTitleProcess	検索要約ペインのプロセス関連セクションのタイトルのスタイル。
.summaryTitleTask	検索要約ペインのタスク関連セクションのタイトルのスタイル。

### エラー詳細

スタイル名	説明
.errorPage	エラー・ページのタブ付きペイン。
.errorLink	ページのボタン・リンクをレンダリングするときに使用するスタイル。
.errorDetails	エラー詳細のあるタブ付きペイン。
.errorDetailsStack	例外スタックのあるタブ付きペイン。
.errorDetailsMessage	エラー・メッセージのテキスト・スタイル。

## Business Process Choreographer Observer 入門

ビジネス・プロセスおよびタスクの実行中に、WebSphere はプロセス・インスタンスとそれに関連するアクティビティの状態変更に関する情報が入ったイベントを発行できます。Business Process Choreographer Observer を使用すると、それらのイベントに基づいた統計情報を取り出し、プロセスとアクティビティに関するレポートを作成できます。

## このタスクについて

特定のプロセス・インスタンス、アクティビティ・インスタンス、またはインライン・ヒューマン・タスクに関する詳細な情報を取得するために、独自のレポートを定義するか、ドリルダウン・アプローチを使用できます。また、報告された結果をエクスポートし、外部でさらに処理することもできます。

Business Process Choreographer Observer は、IT レベルのモニターとビジネス・レベルのモニターの間をギャップを埋めます。また、Business Flow Manager コンポーネントでのイベントを報告する手段を提供することにより、Business Process Choreographer 内で起きていることを理解するのに役立ちます。

### 関連概念

『Business Process Choreographer Observer のユーザー・インターフェース』

Business Process Choreographer Observer は、プロセス・イベントおよびアクティビティ・イベントを通知するための一連の機能を備えたスタンドアロン Web アプリケーションです。このインターフェースは、タスクバー、ナビゲーション・ペイン、およびワークスペースで構成されます。

29 ページの『Business Process Choreographer Explorer の「レポート」タブ』  
Business Process Choreographer Explorer の「レポート」タブを使用して、Business Process Choreographer で処理された特定のプロセスおよびアクティビティについてのレポートを管理します。作成するレポートの種類 (プロセス・レポートやアクティビティ・レポートなど) を選択することができます。独自のレポート定義を保管して、これらの定義をナビゲーション・ペインに追加することもできます。ドリルダウン方式の事前定義のリストおよび図表は、ランタイム・エンティティの状態およびイベント情報を入手するために使用します。例えば、リスト、プロセスおよびアクティビティのスナップショット・グラフ、期間グラフによるプロセス・インスタンスおよびアクティビティ・インスタンスが選択できます。「レポート」タブは、レポート作成が構成されている場合のみ表示されます。レポート作成機能は、Business Process Choreographer Explorer の構成時に構成できますが、後で構成することもできます。

### 関連タスク

 ビジネス・プロセスおよびアクティビティについてのレポート作成  
ビジネス・プロセスおよびアクティビティの処理中に、プロセス、アクティビティ、またはタスクの状態が変わると、イベントが生成されることがあります。これらのイベントは保管され、Business Process Choreographer Explorer によるレポート作成に利用できます。例えば、プロセスのパフォーマンス問題の分析レポートや、アクティビティから呼び出されるサービスの信頼性評価のレポートなどです。

## Business Process Choreographer Observer のユーザー・インターフェース

Business Process Choreographer Observer は、プロセス・イベントおよびアクティビティ・イベントを通知するための一連の機能を備えたスタンドアロン Web アプリケーションです。このインターフェースは、タスクバー、ナビゲーション・ペイン、およびワークスペースで構成されます。

次の図に、Business Process Choreographer Observer のユーザー・インターフェースのレイアウトを示します。



このユーザー・インターフェースの主領域について、以下に説明します。

### タスクバー

タスクバーによって提供されるオプションは、Business Process Choreographer Observer からのログアウト、および一般的な「ヘルプ」ページへのリンクです。

### ナビゲーション・ペイン

ユーザー・インターフェースの左側にあるナビゲーション・ペインには、作成するレポートの種類を選択するときに使用するリンクが表示されます。例えば、アクティビティ・インスタンスのデータをグラフで表示できます。

### ワークスペース

ユーザー・インターフェースの右側にあるワークスペースには、レポート定義を指定してレポートを表示するときに使用するページが表示されます。これらのページにアクセスするには、ナビゲーション・ペインのリンクをクリックします。ページについて詳しくは、それぞれのページの「ヘルプ」アイコン  をクリックしてください。

#### 関連タスク

43 ページの『Business Process Choreographer Observer 入門』

ビジネス・プロセスおよびタスクの実行中に、WebSphere はプロセス・インスタンスとそれに関連するアクティビティの状態変更に関する情報が入ったイベン

トを発行できます。Business Process Choreographer Observer を使用すると、それらのイベントに基づいた統計情報を取り出し、プロセスとアクティビティーに関するレポートを作成できます。

## Business Process Choreographer Explorer の「レポート」タブ

Business Process Choreographer Explorer の「レポート」タブを使用して、Business Process Choreographer で処理された特定のプロセスおよびアクティビティーについてのレポートを管理します。作成するレポートの種類（プロセス・レポートやアクティビティー・レポートなど）を選択することができます。独自のレポート定義を保管して、これらの定義をナビゲーション・ペインに追加することもできます。ドリルダウン方式の事前定義のリストおよび図表は、ランタイム・エンティティーの状態およびイベント情報を入手するために使用します。例えば、リスト、プロセスおよびアクティビティーのスナップショット・グラフ、期間グラフによるプロセス・インスタンスおよびアクティビティー・インスタンスが選択できます。「レポート」タブは、レポート作成が構成されている場合にのみ表示されます。レポート作成機能は、Business Process Choreographer Explorer の構成時に構成できますが、後で構成することもできます。

### 使用可能なアクション

ナビゲーション・ペインでは、以下のアクションを使用できます。

- グループを縮小表示または展開する。

ナビゲーション・ペイン内のある項目を展開または縮小表示するには、その項目の横にある矢印をクリックします。

- 事前定義のリストまたはグラフにナビゲートする。

レポートを作成するインスタンスの種類をクリックします。

- プロセス・レポート・ウィザードまたはアクティビティー・レポート・ウィザードへナビゲートする。

レポートのタイプ、レポートの内容、レポートのフィルター基準を指定するに

は、「新規レポート」アイコン (  ) をクリックします。

- 保存済みのプロセス・レポートまたはアクティビティー・レポートを実行する。

レポートを実行するには、レポート名をクリックします。

- 保存済みのプロセス・レポート定義またはアクティビティー・レポート定義のポップアップ・メニューを開く。

保存済みのレポート定義を操作するには、「ポップアップ・メニューの表示」アイコン (  ) をクリックします。

– レポート定義を削除するには、「削除」アイコン (  ) をクリックします。

– レポート定義を編集するには、「編集」アイコン (  ) をクリックします。

- レポート定義をコピーするには、「コピー」アイコン (  ) をクリックします。
- レポート結果をエクスポートするには、「エクスポート」アイコン (  ) をクリックします。
- レポートを非同期で実行するには、「非同期レポート (Asynchronous Report)」アイコン (  ) をクリックします。
  - 非同期レポートが正常に完了すると、ナビゲーション・ペインに「非同期レポート完了 (Asynchronous Report Completed)」アイコン (  ) が表示されます。レポートの名前をクリックして、結果を表示します。
  - 非同期レポートが正常に完了しなかった場合は、「非同期レポート失敗 (Asynchronous Report Failed)」アイコン (  ) が表示されます。

## ナビゲーション・ペインの事前定義リストおよびグラフ

ナビゲーション・ペインには、以下に示す事前定義リストおよびグラフのグループがあります。

**リスト** このグループには、以下のリストがあります。

### プロセス

このリストは、指定した期間内にプロセス・イベントを出力したプロセスを表示する場合に使用します。プロセスのリストは、プロセス状態に応じて表示されます。

### アクティビティ

このリストは、選択したアクティビティが、指定した期間内に到達した状態を表示する場合に使用します。アクティビティのリストは、アクティビティ状態に応じて表示されます。

### ユーザー

このリストは、指定した期間内に、選択したユーザーが実行したアクティビティと、そのアクティビティが到達した状態を表示する場合に使用します。アクティビティは、その状態に応じて表示されます。対応するユーザーは、アクティビティごとに表示されます。

**グラフ** このグループには、以下のグラフがあります。

### プロセス・スナップショット

このグラフは、指定した時刻に状態の異なるプロセス・インスタンスの数を調べる場合に使用します。データは棒グラフまたは円グラフで表示できます。

### 期間によるプロセス

このグラフは、指定した期間内に指定した状態に到達したプロセス・インスタンスの分布を調べる場合に使用します。各インスタン

スは、それが指定の状態に到達したタイム・スライス内に表示されます。データは折れ線グラフ、棒グラフ、または円グラフで表示できます。

### アクティビティ・スナップショット

このグラフは、指定した時刻に状態の異なるアクティビティ・インスタンスの数を調べる場合に使用します。データは棒グラフまたは円グラフで表示できます。

### 期間によるアクティビティ

このグラフは、指定した期間内に指定した状態に到達したアクティビティ・インスタンスの分布を調べる場合に使用します。各インスタンスは、それが指定の状態に到達したタイム・スライス内に表示されます。データは折れ線グラフ、棒グラフ、または円グラフで表示できます。

## プロセス・レポートおよびアクティビティ・レポート

ナビゲーション・ペインには、以下のレポート・ウィザードへのリンクがあります。レポート・ウィザードは「**新規レポート**」アイコン () で示されます。

### プロセス・レポート

プロセス・レポートは、プロセス・インスタンス・イベントを照会する場合に使用します。これらのイベントには、プロセス・インスタンスの状態変更が記述されています。作成するレポートのデータを定義する場合は、レポート・ウィザードを使用してください。レポート定義を保管して検索することができます。

### アクティビティ・レポート

アクティビティ・インスタンス・イベントを照会する場合は、アクティビティ・レポートを使用します。これらのイベントには、アクティビティ・インスタンスの状態変更が記述されています。個々のレポートを指定する場合は、レポート・ウィザードを使用します。レポート定義を保管して検索することができます。

### 関連タスク

43 ページの『Business Process Choreographer Observer 入門』

ビジネス・プロセスおよびタスクの実行中に、WebSphere はプロセス・インスタンスとそれに関連するアクティビティの状態変更に関する情報が入ったイベントを発行できます。Business Process Choreographer Observer を使用すると、これらのイベントに基づいた統計情報を取り出し、プロセスとアクティビティに関するレポートを作成できます。

### 関連資料



Business Process Choreographer Explorer レポート作成機能および Event Collector の構成

Business Process Choreographer Explorer レポート作成機能の使用はオプションです。しかし、使用するには、事前にデータベースをセットアップし、アプリケーションをインストールしておく必要があります。

---

## 第 3 章 サーバーおよびクラスターの管理

サーバーの管理には、サーバーの作成、開始、および停止が含まれます。

---

### サーバーの作成

管理対象ノードでサーバーを作成するには、管理コンソールの「新規アプリケーション・サーバーの作成」ウィザード、または `wsadmin createApplicationServer` コマンドを使用します。

#### 始める前に

カスタム・プロファイルを作成した後、それをデプロイメント・マネージャーに統合して、管理対象ノードを作成しておく必要があります。

#### このタスクについて

以下の手順は、デプロイメント・マネージャーで管理されるサーバーを作成するためのものです。スタンドアロン・サーバーを作成するには、スタンドアロン・サーバー・プロファイルを作成します。

管理コンソールを使用する方法以外の方法でも、サーバーを作成できます (`wsadmin createApplicationServer` コマンドなど)。詳しくは、『AdminTask オブジェクトのコマンド』を参照してください。

#### 手順

1. 管理コンソールで「アプリケーション・サーバー」 → 「新規」をクリックして、「新規アプリケーション・サーバーの作成」ウィザードを開始します。
2. 『アプリケーション・サーバーの作成』の説明に従い、「サーバー・テンプレートの選択」ページで「defaultProcessServer」テンプレートを選択するか、適切なユーザー定義テンプレートを選択します。
3. **オプション:** ビジネス・プロセスまたはヒューマン・タスクを含むアプリケーションがサーバーで実行される場合は、Business Process Choreographer を構成します。
4. **オプション:** 1 つ以上のサーバーに SchedulerCalendars アプリケーションをインストールします。

#### 次のタスク

これで、サーバーを始動し、モジュールをサーバーにデプロイできます。

---

### サーバーおよびクラスターの開始または停止

サーバーでアプリケーションを開始する前に、サーバーが稼働している必要があります。サーバーを開始する方法はいくつかありますが、スタンドアロン・サーバーと管理対象サーバーのいずれであるかによって、使用する方法が異なります。管理対象サーバーの場合、サーバーを開始する前に、ノード・エージェントが稼働して

いる必要があります。管理対象サーバーは、デプロイメント・マネージャーの管理コンソールから開始できます。デプロイメント環境またはクラスターを導入している場合、デプロイメント・マネージャーの管理コンソールですべてのサーバーを一括に開始または停止できます。

## このタスクについて

クラスターを使用している場合、「アプリケーション・サーバー」サブコンポーネント（「サーバー」>「アプリケーション・サーバー」>「*server\_name*」>「管理」>「サーバー・コンポーネント」>「アプリケーション・サーバー」）の「初期状態」プロパティは、クラスターの開始時にクラスター内の個々のサーバーの状態を制御する目的では使用できません。これは、サーバーの「アプリケーション・サーバー」サブコンポーネントの状態を制御するためのものです。クラスターの個々のメンバーを開始および停止するには、管理コンソールの「サーバー」のオプションを使用するか、またはコマンド行のコマンド (**startServer** および **stopServer**) を使用する方法をお勧めします。

## スタンドアロン・サーバーの開始

管理コンソールを使用する前、およびサーバーでアプリケーションを実行する前に、サーバー・プロセスを開始しておく必要があります。

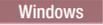
### 始める前に

この時点でサーバーが実行されてはなりません。インストール検査プログラムを実行すると、サーバーが開始される点に注意してください。

### このタスクについて

スタンドアロン・サーバーを開始する方法はいくつかあります。複数のスタンドアロン・サーバー・プロファイルを作成したホストの場合、正しいプロファイルが使用されていることを確認してください。

サーバーを開始するには、以下のいずれかのアクションを実行します。

-  「スタート」メニューから、「IBM WebSphere」→「Process Server 6.2」→「プロファイル」→「*profile\_name*」→「サーバーの起動」の順にクリックします。
- ファースト・ステップ・コンソールで、「サーバーの始動」をクリックします。
- **startServer** コマンドを使用します。

*install\_root/profiles/profile\_name/logs/server\_name/startServer.log* に進行状況が記録されます。サーバーが正常に始動すると、「サーバー *server\_name* が e-business 用にオープンされました。プロセス ID は *nnnn* です。」というメッセージがログに記録されます。

### 次のタスク

これで、管理コンソールを始動し、スタンドアロン・サーバーを管理し、アプリケーションをサーバーにデプロイできます。WebSphere Process Server を今回初めて使用する場合、既にサンプルをインストールしていれば、ファースト・ステップ・コンソールからサンプル・ギャラリーを開くことができます。

## スタンドアロン・サーバーの停止

サーバーの構成を変更する場合、またはサーバーにデプロイされているモジュールを変更する場合は、サーバー・プロセスを停止する必要があります。

### 始める前に

- サーバーが稼働していることを確認します。
- 処理が完了するまで待ちます。十分な期間にわたって待機するか、または Performance Monitoring Infrastructure のカウンターを使用して、キューに格納されているすべての処理が完了したかどうかを判別します。
- 新規処理が開始しないようにします。

### このタスクについて

サーバーを停止する方法はいくつかあります。

サーバーを停止するには、以下のいずれかのアクションを実行します。

-  「スタート」メニューで「IBM WebSphere」 → 「Process Server 6.2」 → 「プロファイル」 → 「*profile\_name*」 → 「サーバーの停止」の順に選択します。
- ファースト・ステップ・コンソールで、「サーバーの停止」をクリックします。
- stopServer コマンドを使用します。

サーバー・ログ `install_root/profiles/profile_name/logs/server_name/stopServer.log` に進行状況が記録されます。サーバーが正常に停止すると、「サーバー *server\_name* の停止が完了しました。」というメッセージがサーバー・ログに記録されます。

### 次のタスク

サーバーが停止するまで待ち、停止したら、サーバーを再始動します。

## デプロイメント・マネージャーの始動

デプロイメント・マネージャーの管理コンソールを使用してセルを管理する前に、そのデプロイメント・マネージャーを始動する必要があります。

### 始める前に

デプロイメント・マネージャーが稼働してはなりません。インストール検査プログラムを実行すると、デプロイメント・マネージャーが始動される点に注意してください。

### このタスクについて

デプロイメント・マネージャーを始動する方法はいくつかあります。

デプロイメント・マネージャーを開始するには、以下のいずれかのアクションを実行します。

- **Windows** 「スタート」メニューで「IBM WebSphere」 → 「Process Server 6.2」 → 「プロファイル」 → 「*profile\_name*」 → 「デプロイメント・マネージャーの開始」の順に選択します。
- ファースト・ステップ・コンソールで、「デプロイメント・マネージャーの始動」をクリックします。
- startManager コマンドを使用します。

*install\_root/profiles/profile\_name/logs/server\_name/startServer.log* に進行状況が記録されます。サーバーが正常に始動すると、「サーバー *server\_name* が e-business 用にオープンされました。プロセス ID は *nnnn* です。」というメッセージがログに記録されます。

## 次のタスク

これで、管理コンソールを開始し、セルを管理できます。ノードをデプロイメント・マネージャーに統合してから、サーバーとクラスターを作成できます。

## デプロイメント・マネージャーの停止

デプロイメント・マネージャーはサーバー・プロセスです。このため、オペレーティング・システムの保守作業を行う場合などに停止する必要があります。デプロイメント・マネージャーは、ドメイン内のその他のサーバーの運用に影響せずいつでも停止できます。

### 始める前に

デプロイメント・マネージャーは稼働している必要があります。

### このタスクについて

デプロイメント・マネージャーを停止する方法はいくつかあります。

デプロイメント・マネージャーを停止するには、以下のいずれかのアクションを実行します。

- **Windows** 「スタート」メニューで、「IBM WebSphere」 → 「Process Server 6.2」 → 「プロファイル」 → 「*profile\_name*」 → 「デプロイメント・マネージャーの停止」の順に選択します。
- ファースト・ステップ・コンソールで、「デプロイメント・マネージャーの停止」をクリックします。
- 管理コンソールで「システム管理」 → 「デプロイメント・マネージャー」 → 「停止」 → 「OK」の順にクリックします。管理コンソールが閉じ、その後サーバーが実行停止します。
- stopManager コマンドを使用します。

サーバー・ログ *install\_root/profiles/profile\_name/logs/server\_name/stopServer.log* に進行状況が記録されます。サーバーが正常に停止すると、「サーバー *server\_name* の停止が完了しました。」というメッセージがサーバー・ログに記録されます。

## 次のタスク

デプロイメント・マネージャーが停止するまで待ち、停止したら、サーバーを再始動します。

## ノード・エージェントの開始

管理対象ノードのノード・エージェントは、ノードでサーバーを開始する前に開始する必要があるサーバー・プロセスです。デプロイメント・マネージャーがノード・エージェントと通信するには、ノード・エージェントが開始されている必要があります。

### 始める前に

ノード・エージェントが稼働してはなりません。管理コンソールからノード・エージェントが稼働しているかどうかを判断する手順をここで説明します。

### このタスクについて

ノード・エージェントは、ノードが構成されているホストのコマンド行で、`install_root/bin` ディレクトリーから開始する必要があります。

#### 手順

1. デプロイメント・マネージャーの管理コンソールからサーバーの正しいノード・エージェントを判別するには、以下のようにします。「サーバー」 → 「アプリケーション・サーバー」をクリックします。
2. ノード・エージェントが稼働しているかどうかを判断するには、以下のようにします。「システム管理」 → 「ノード・エージェント」をクリックします。
3. ノード・エージェントを開始するには、次のようにします。 `startNode` コマンドを実行します。このコマンドで使用できるオプションを確認するには、`-help` オプションを使用します。オプションを指定しないと、ノードが開始されます。「サーバー `nodeagent` が `e-business` 用にオープンされました。プロセス ID は `nnnn` です。」というメッセージが表示されるまで待機します。

#### 例

- デフォルト・プロファイルのノード・エージェントを開始するには、`startNode` と入力します。
- オプションのリストを表示するには、`startNode -help` と入力します。
- `Custom03` プロファイルのノード・エージェントを開始するには、`startNode -profileName Custom03` と入力します。
- `Custom03` プロファイルのノード・エージェントを開始し、トレース情報をログ・ファイル `install_root/profiles/Custom03/logs/startServer.log` に書き込むには、`startNode -logfile -profileName Custom03` と入力します。

## 次のタスク

これで、デプロイメント・マネージャーからこのノードを管理できます (ノード上のサーバーを開始することもできます)。

## ノード・エージェントの停止

管理対象ノードのノード・エージェントは、停止する必要があるサーバー・プロセスです。

### 始める前に

- ノード・エージェントは実行されていなければなりません。
- すべてのサーバーが停止している必要があります。

### 手順

1. デプロイメント・マネージャーの管理コンソールからノード・エージェントを停止するには、以下のようにします。「システム管理」 → 「ノード・エージェント」を選択します。
2. ノード・エージェントを選択します。
3. 「停止」をクリックします。

## ノード・エージェントの再開

管理対象ノードのノード・エージェントは、再始動する必要があるサーバー・プロセスです。

### 始める前に

ノード・エージェントは実行されていなければなりません。

### 手順

1. デプロイメント・マネージャーの管理コンソールからノード・エージェントを再開するには、以下のようにします。「システム管理」 → 「ノード・エージェント」をクリックします。
2. ノード・エージェントを選択します。
3. 「再始動」をクリックします。

## デプロイメント環境の開始と停止

IBM 提供のパターンに基づいて、管理コンソールから直接デプロイメント環境を開始または停止することができます。この手順で、カスタム・デプロイメント環境を管理することはできません。

### 始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールに移動し、「サーバー」 → 「Deployment Environments」をクリックします。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

デプロイメント環境を開始または停止するには、デプロイメント環境が存在しなければなりません。

## このタスクについて

IBM 提供のパターンに基づいて、デプロイメント環境を開始または停止する場合は必ず、以下のステップを使用します。

**注:** カスタム・デプロイメント環境を開始または停止するには、個々のクラスターを開始および停止する必要があります。

### 手順

1. 開始および停止するデプロイメント環境の名前の隣にあるチェック・ボックスを選択します。
2. 以下のいずれかのアクションを行います。

アクション	結果
「開始」をクリックします。	デプロイメント・マネージャーにより、デプロイメント環境を構成するクラスターが始動されます。
「停止」をクリックします。	デプロイメント・マネージャーにより、デプロイメント環境を構成するクラスターが停止されます。

**注:** このプロセスは、ご使用のデプロイメント環境のサイズによっては数分かかる場合があります。

## タスクの結果

画面が最新表示され、デプロイメント環境の状況が示されます。

### 関連情報

-  Administration Thin Client の使用
-  デプロイメント環境の状況情報
-  デプロイメント環境の機能の状況 (Deployment environment function status)
-  デプロイメント環境の状況
-  startDeploymentEnv コマンド
-  stopDeploymentEnv コマンド

## コマンド行を使用してデプロイメント環境を開始する操作

wsadmin コマンドを使用して、デプロイメント環境を開始することができます。

### 始める前に

管理クライアントから、開始するデプロイメント環境のデプロイメント・マネージャーに接続する必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者権限またはオペレーター権限を持つユーザー ID とパスワードを使用する必要があります。

## このタスクについて

このタスクでは、wsadmin コマンドを使用してデプロイメント環境を開始します。

### 手順

1. コマンド・ウィンドウを開きます。 wsadmin コマンドは、<WPS>/profiles/<dmgr profile>/bin ディレクトリーまたは <WPS>/bin ディレクトリーにあります。
2. コマンド・プロンプトで wsadmin コマンドを入力して、コマンド環境に入ります。

**注:** wsadmin を接続モードで実行する場合は、正しいデプロイメント・マネージャーに接続していることを確認してください。

3. startDeploymentEnv コマンドを使用して、デプロイメント環境を開始します。

**注:** 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力を求めるプロンプトが表示されます (コマンド内で指定していない場合)。

### 例

この例では、管理セキュリティを有効にしてホスト (**myDmgr**) のデプロイメント環境 (**MyDepEnv**) を開始します。

**注:** デプロイメント・マネージャーの bin フォルダーから管理クライアントを実行している場合は、コマンドに -host パラメーターと -port パラメーターを組み込む必要はありません。

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgradmin -password dmgrpass  
> $AdminTask startDeploymentEnv {-topologyName myDepEnv}
```

-connType パラメーターでは、使用する接続のタイプを指定します。デフォルトの引数は SOAP です。

**注:** デフォルトが SOAP なので、使用する接続タイプが SOAP の場合は、値を明示的に指定する必要はありません。

-host パラメーターでは、SOAP または RMI の接続で使用するホストを指定します。-host のデフォルト値はローカル・ホストです。

**注:** ローカル・ホストでノードを実行している場合は、-host を指定する必要はありません。

**注:** 管理セキュリティが無効になっている場合は、ユーザー ID とパスワードを指定する必要はありません。

## コマンド行を使用してデプロイメント環境を停止する操作

wsadmin コマンドを使用して、デプロイメント環境を停止することができます。

## 始める前に

管理クライアントから、停止するデプロイメント環境のデプロイメント・マネージャーに接続する必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者権限またはオペレーター権限を持つユーザー ID とパスワードを使用する必要があります。

## このタスクについて

このタスクでは、`wsadmin` コマンドを使用してデプロイメント環境を停止します。

### 手順

1. コマンド・ウィンドウを開きます。 `wsadmin` コマンドは、`<WPS>/profiles/<dmgr profile>/bin` ディレクトリーまたは `<WPS>/bin` ディレクトリーにあります。
2. コマンド・プロンプトで `wsadmin` コマンドを入力して、コマンド環境に入ります。

**注:** `wsadmin` を接続モードで実行する場合は、正しいデプロイメント・マネージャーに接続していることを確認してください。

3. `stopDeploymentEnv` コマンドを使用して、デプロイメント環境を停止します。

**注:** 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力を求めるプロンプトが表示されます (コマンド内で指定していない場合)。

## 例

この例では、管理セキュリティを有効にしてホスト (`myDmgr`) のデプロイメント環境 (`MyDepEnv`) を停止します。

**注:** デプロイメント・マネージャーの `bin` フォルダーから管理クライアントを実行している場合は、コマンドに `-host` パラメーターと `-port` パラメーターを組み込む必要はありません。

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgradmin -password dmgrpass  
> $AdminTask stopDeploymentEnv {-topologyName myDepEnv}
```

`-connType` パラメーターでは、使用する接続のタイプを指定します。デフォルトの引数は `SOAP` です。

**注:** デフォルトが `SOAP` なので、使用する接続タイプが `SOAP` の場合は、値を明示的に指定する必要はありません。

`-host` パラメーターでは、`SOAP` または `RMI` の接続で使用するホストを指定します。`-host` のデフォルト値はローカル・ホストです。

**注:** ローカル・ホストでノードを実行している場合は、`-host` を指定する必要はありません。

**注:** 管理セキュリティが無効になっている場合は、ユーザー ID とパスワードを指定する必要はありません。

## クラスターの開始

クラスター内のすべてのサーバー (クラスター・メンバー) を 1 回の操作で開始できます。**Ripplestart** オプションを指定すると、各サーバーが停止してから、順に開始されます。クラスターを開始すると、ワークロード管理が自動的に有効になります。

### 始める前に

- ノード・エージェントが稼働していることを確認します。
- クラスターにデプロイされているアプリケーションに必要なすべてのリソースが使用可能であることを確認し、すべての前提条件サブシステムを開始します。

### 手順

1. デプロイメント・マネージャーの管理コンソールからクラスターを開始するには、以下のようにします。「サーバー」 → 「クラスター」をクリックします。
2. クラスターを選択します。
3. サーバーが稼働していない場合は、「開始」をクリックします。
4. サーバーが稼働している場合は、「**ripple 始動**」をクリックします。まず各サーバーが停止され、その後順に開始されます。

### 次のタスク

これで、アプリケーションをクラスターで開始できるようになりました。いずれかのサーバーで実行中のプロセスをトレースおよびデバッグする必要がある場合は、Java デバッグ可能にできます。

## クラスターの停止

クラスター内のすべてのサーバー (クラスター・メンバー) を 1 回の操作で停止できます。

### 始める前に

1. 処理が完了するまで待ちます。十分な期間にわたって待機するか、または Performance Monitoring Infrastructure のカウンターを使用して、キューに格納されているすべての処理が完了したかどうかを判別します。
2. 新規処理が開始しないようにします。

### 手順

1. デプロイメント・マネージャーの管理コンソールからクラスターを停止するには、以下のようにします。「サーバー」 → 「クラスター」をクリックします。
2. クラスターを選択します。
3. 「停止」をクリックします。

## サーバーの開始

管理対象サーバーでアプリケーションを実行する前に、サーバー・プロセスを開始しておく必要があります。単一サーバー (クラスター・メンバーを含む) を開始できます。

## 始める前に

- ノード・エージェントが稼働していることを確認します。
- サーバーにデプロイされているアプリケーションに必要なすべてのリソースが使用可能であることを確認し、すべての前提条件サブシステムを開始します。
- サーバーが停止していることを確認します。サーバーで障害が発生しており、新規処理の開始前にリカバリー機能を完了させたい場合は、startServer コマンドと **-recovery** オプションを使用してサーバーをリカバリー・モードで再始動する必要があります。

## 手順

1. デプロイメント・マネージャーの管理コンソールからサーバーを開始するには、以下のようにします。「サーバー」→「アプリケーション・サーバー」をクリックします。
2. サーバーを選択します。
3. 「開始」をクリックします。 `install_root/profiles/profile_name/logs/server_name/startServer.log` に進行状況が記録されます。サーバーが正常に始動すると、「サーバー `server_name` が e-business 用にオープンされました。プロセス ID は `nnnn` です。」というメッセージがログに記録されます。

## 次のタスク

注: その他のプロセスでは、新規開始サーバーが直ちにディスクカバーされないことがあります。

これで、アプリケーションをサーバーで開始できるようになりました。サーバーで実行中のプロセスをトレースおよびデバッグする必要がある場合は、Java デバッグを可能にできます。

## i5/OS コマンド行からのサーバーの開始

i5/OS コマンド行からサーバーを開始するには、以下の 2 つの方法のいずれかを実行します。

### 始める前に

- ノード・エージェントが稼働していることを確認します。
- サーバーにデプロイされているアプリケーションに必要なすべてのリソースが使用可能であることを確認し、すべての前提条件サブシステムを開始します。
- サーバーが停止していることを確認します。
- startServer Qshell コマンドを使用します。
- または
- Submit Job (SBMJOB) CL コマンドを使用します。以下の CL コマンドを i5/OS® コマンド行から実行できます。

```
SBMJOB CMD(CALL PGM(QWBI61/QWASSTRSVR) PARM('-profilePath'  
      'profile_root' '-server' 'server_name')) JOB(server_name)  
      JOBQ(QWAS61/QWASJOBQ) JOBQ(QWAS61/QWASJOBQ) USER(QEJBSVR)  
      LANGID(*USRPRF)  
      CENTRYID(*USRPRF) CCSID(*USRPRF) OUTQ(QWAS61/QWASOUTQ) ALWMLTTHD(*YES)
```

`server_name` はサーバーの名前です。

`profile_root` はサーバーのプロファイル・ディレクトリーです。

## サーバーを Java デバッグ可能にする

サーバーの始動後に、サーバーをトレースおよびデバッグする必要がある場合は、Java デバッグ可能にできます。

### 始める前に

サーバーは実行されていない必要があります。

#### 手順

1. デプロイメント・マネージャーの管理コンソールから Java デバッグを可能にするには、以下のようにします。「サーバー」 → 「アプリケーション・サーバー」をクリックします。
2. サーバーを選択します。
3. 「サーバー・インフラストラクチャー」 → 「Java およびプロセス管理」 → 「プロセス定義」 → 「Java 仮想マシン」をクリックします。
4. 標準 Java デバッガーを有効にするには、以下のようにします。「Java 仮想マシン」ページで「デバッグ・モード」をクリックします。
5. 必要に応じて「デバッグ・モード」の引数を設定します。
6. 「OK」をクリックします。
7. 変更を保管します。

### 次のタスク

この時点でサーバーを停止する必要があります。停止後、スタンドアロン・サーバーの場合は再始動してください。

## サーバーの停止

サーバーの構成を変更する場合、またはサーバーにデプロイされているモジュールを変更する場合は、サーバー・プロセスを停止する必要があります。単一サーバー（クラスター・メンバーを含む）を停止できます。

### 始める前に

- サーバーが稼働していることを確認します。
- 処理が完了するまで待ちます。十分な期間にわたって待機するか、または Performance Monitoring Infrastructure のカウンターを使用して、キューに格納されているすべての処理が完了したかどうかを判別します。
- 以下の手順で、新規処理が開始しないようにします。
  - IBM HTTP Server を使用している場合は、`plugin_cfg.xml` ファイルを変更し、HTTP トラフィックのためにサーバーを除去します。その他の HTTP サーバーを使用している場合は、ご使用のサーバーの手順に従いサーバーを除去します。
  - IIOF トラフィックの場合は、サーバーのランタイム・ウェイトをゼロに設定します。
  - サービス統合バスを静止します。

## 手順

1. デプロイメント・マネージャーの管理コンソールからサーバーを停止するには、以下のようにします。「サーバー」 → 「アプリケーション・サーバー」をクリックします。
2. サーバーを選択します。
3. 「停止」をクリックします。 `install_root/profiles/profile_name/logs/server_name/stopServer.log` に進行状況が記録されます。サーバーが正常に停止すると、「サーバー `server_name` の停止が完了しました。」というメッセージがログに記録されます。

## 次のタスク

サーバーが停止するまで待ち、停止したら、サーバーを再始動します。



## 第 4 章 デプロイメント環境の管理

デプロイメント・マネージャー上の管理コンソールによって、そのデプロイメント・マネージャーに対して定義されているデプロイメント環境を管理します。また、管理コンソールでデプロイメント環境を作成、削除、インポート、およびエクスポートすることもできます。

### 始める前に

デプロイメント・マネージャーを開始し、管理コンソールにログインしていることを確認してください。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

### このタスクについて

デプロイメント・マネージャーによって管理されているデプロイメント環境を更新する必要がある場合は、デプロイメント環境を管理してください。「**デプロイメント環境**」管理コンソール・ページは、特定のデプロイメント・マネージャーに対して定義されているデプロイメント環境の管理と定義に関連するすべてのタスクの開始点です。

### 手順

1. 管理コンソールで、「**サーバー**」>「**デプロイメント環境**」をクリックします。
2. デプロイメント環境のコンポーネントを表示するには、名前をクリックします。
3. 既存の環境の場合は、管理するデプロイメント環境の隣にあるチェック・ボックスを選択した後、次のボタンのいずれかをクリックします。

機能	タスク
開始 または 停止	デプロイメント環境の開始および停止
削除	デプロイメント環境からのリソースの除去。 このオプションは、リソースを削除するわけではありません。
エクスポート	デプロイメント環境のエクスポート

4. 新しいデプロイメント環境をデプロイメント・マネージャーに追加するには、「**新規作成**」または「**インポート**」を使用します。

### 次のタスク

デプロイメント環境エンティティを管理します。

#### 関連情報

 デプロイメント環境トポロジーの更新

-  ホスト別名の構成
-  デプロイメント環境の認証別名の構成
-  カスタム・デプロイメント環境の構成
-  デプロイメント環境の据え置かれた構成の構成
-  コマンド行を使用したデプロイメント環境の構成

---

## カスタム・デプロイメント環境のレイアウト構成

この概要では、カスタム・デプロイメント環境の構成における 2 つの主要な考慮事項について説明します。環境で使用するクラスターおよび単一サーバーの選択と、デプロイメント環境構成の指定です。これらの考慮事項を理解すれば、デプロイメント環境を効率的に計画および実装できます。

65 ページの『デプロイメント環境で使用するクラスターと単一サーバーの選択』では、デプロイメント環境を構成するクラスターとサーバーを定義します。機能ごとにクラスターを作成するパターンに基づくデプロイメント環境とは異なり、カスタム・デプロイメント環境では、機能を実行するために必要なクラスターとサーバーを追加します。

65 ページの『デプロイメント環境構成の定義』では、クラスターとサーバーに構成する機能について説明します。これらの機能は、メッセージング、Common Event Infrastructure、およびアプリケーション・サポートです。

システム内でデプロイメント環境構成を生成して構成を完了する前であれば、構成に戻って変更することができます。システム内でデプロイメント環境構成を生成した後でも、現行構成を確認できます。また、サーバーとクラスターを追加したり、追加の機能を構成したりすることもできます。このデプロイメント環境での管理対象からサーバーとクラスターを除去することもできます。生成済みの機能構成を取り消すことや、デプロイメント環境の別のサーバーまたはクラスターにまだ必要なデプロイメント環境定義からサーバーまたはクラスターを除去することはできません。

### すべてのカスタム・デプロイメント環境における要件

カスタム・デプロイメント環境レイアウトには、以下の制限があります。

- ユーザーがデプロイメント環境を生成して構成を完了すると、関連するコントロールがチェックされ、使用不可になります。つまり、構成を取り消すことができなくなります。
- デプロイメント環境を生成後、コンポーネントのコントロールがチェックされず、使用不可にならない場合は、次の順番で関連する機能を構成する必要があります。関連するメッセージング・エンジンを構成して、Common Event Infrastructure (CEI) を構成し、次にアプリケーション・サポート (このトピックの以降で説明します) を構成します。

- システムに存在する構成は、トポロジー・レイアウト構成よりも優先されます。したがって、カスタム・トポロジーをエクスポートすると、トポロジーに含まれるサーバーの実際の構成が反映されます。

管理コンソールの「トポロジー・レイアウト」ページには、カスタム・トポロジーにおいて構成する必要のある 4 つのセクションがあります。

- クラスタおよび単一サーバーの選択
- メッセージング
- Common Event Infrastructure
- コンポーネント

以下のセクションで、カスタム・トポロジー・レイアウト構成を完了するためのその他の要件を説明しています。

## デプロイメント環境で使用するクラスタと単一サーバーの選択

「トポロジー・レイアウト」ページの「クラスタおよび単一サーバーの選択」セクションを使用して、デプロイメント環境内のクラスタとサーバーを管理し、クラスタとサーバーが提供する機能を定義します。

「トポロジー・レイアウト」ページの「クラスタおよび単一サーバーの選択」セクションには、デプロイメント環境の一部として構成する、使用可能なクラスタおよびサーバーのリストが含まれています。機能構成のコラボレーション単位にクラスタとサーバーを割り当てます。各コラボレーション単位は、デプロイメント環境において 1 つの機能を提供するクラスタおよびサーバーのグループ全体を表します。デプロイメント環境からクラスタまたはサーバーを削除することもできます。ただし、除去できるクラスタおよびサーバーは、構成内の他のクラスタまたはサーバーで必要としなくなったもののみです。

## デプロイメント環境構成の定義

「トポロジー・レイアウト」ページの「デプロイメント環境の構成の指定」セクションを使用して、デプロイメント環境の特定の機能に参加するクラスタまたはサーバーを定義します。

### メッセージング

**注:** 分割されたメッセージング・エンジンはサポートされません。

選択したターゲットのメッセージング宛先を構成するには、「メッセージング」タブのフィールドを使用します。各テーブルはコラボレーション単位を表し、「メッセージング」セクションには複数のテーブルを含むことができます。それぞれのユニットのローカル構成のオプションには、ターゲット (クラスタ/サーバー) を 1 つのみ選択する必要があります。このユニットのその他のすべてのターゲットでは、リモート宛先が想定されます。アプリケーションがリモート宛先構成のターゲットにメッセージを送信する際、システムはメッセージをそのユニットのローカル・ターゲットにルーティングします。

メッセージング構成は、Service Component Architecture (SCA)、CEI、Business Process Choreographer システム・バスに適用されます。

トポロジー構成内のローカル宛先との競合を避けるために、以下のルールが適用されます。

- SCA システム・バス・メッセージング・エンジン構成により、ローカル宛先ロケーションとリモート宛先ロケーションが決定します。SCA アプリケーション、CEI バス構成、および Business Process Choreographer バス構成は、SCA システム・バス構成に従います。
- 他のバス用のメッセージング・エンジンを単位内の別のターゲットに配置する場合、その単位内の他のターゲットはリモート宛先の役割であると想定されます。CEI または Business Process Choreographer バスの構成が異なる場合、情報メッセージは指定されたバスのメッセージング・エンジンが SCA メッセージング・エンジンと同じターゲットに見つからないことを示します。
- リモートまたはローカル宛先が既に構成されており、指定のユニットの現在のバス設定と競合するターゲットを追加しようとした場合、システムはエラー・メッセージを生成します。

### Common Event Infrastructure

CEI タブに CEI を構成します。「メッセージング」と同じです。CEI は複数のテーブルを保有でき、各テーブルが単位を表します。各テーブルで、CEI クラスターまたはサーバーを 1 つ選択します（「クラスター/サーバー」列）。これは、「サーバー」ラジオ・ボタンを選択すると、サーバーとして機能します。サーバーとして構成されていないすべてのターゲットでは、宛先ロールが想定されます。対応するターゲットでは、このターゲットで発行される Common Base Event が該当するコラボレーション単位のサーバーに送信されるように、Event Infrastructure エミッター・ファクトリーの Java Naming and Directory Interface (JNDI) 名が構成されています。

### アプリケーション・サポート

「アプリケーション・サポート」タブには、特定のデプロイメント・ターゲットに対して構成可能なすべてのコンポーネントがリストされます。関連するコラボレーション単位内のコンポーネント機能を構成します。例えば、単位内で構成されている Business Process Choreographer Container から発行される Common Base Event を収集するには、同じ単位内で Business Process Choreographer Event Collector を構成します。各コンポーネント構成には、他のコンポーネント構成に対する要件と依存関係があります。依存関係は、クリアされて使用不可のコントロールにより表現されます。このようなコントロールを使用可能にするには、依存コントロールを最初に構成する必要があります。

注: 依存コントロールは、「メッセージング」タブまたは「CEI」タブに構成されます。

コンポーネント間の関係を 67 ページの表 5 に示します。

表 5. デプロイメント環境のコンポーネントの関係

コンポーネント	目的	関連コンポーネント	考慮事項
Service Component Architecture (SCA)	<p>SCA アプリケーション・サポート用にデプロイメント・ターゲットを構成します。</p> <p>SCA システムおよびアプリケーションのバス・メンバーは、対応するメッセージング構成がローカルの場合はローカルに構成されます。それ以外の場合は、対応するメッセージング単位の指定に基づき、リモート宛先を使用してリモートに構成されます。</p>	メッセージング	<p>デプロイメント・ターゲットがメッセージング用に構成されていない場合、SCA 構成は使用できません。</p>
Business Process Choreographer Container	<p>ビジネス・フローとヒューマン・タスクの両方をサポートするようにデプロイメント・ターゲットを構成します。</p> <p>この構成では、SCA 構成に従って Business Process Choreographer システム・バスをセットアップします。</p>	<p>メッセージング</p> <p>Service Component Architecture</p> <p>Business Process Choreographer Explorer</p>	<p>デプロイメント・ターゲットがメッセージング用に構成されていない場合、または Service Component Architecture サポート用に構成されていない場合、Business Process Choreographer 構成は使用できません。</p> <p>1 つのコラボレーション単位が 1 つの Business Process Choreographer 構成をサポートします。 「アプリケーション・サポート」タブで、必要なだけ単位を追加します。</p> <p>Container を管理する場合は、Business Process Choreographer Explorer を構成することを検討してください。</p>

表 5. デプロイメント環境のコンポーネントの関係 (続き)

コンポーネント	目的	関連コンポーネント	考慮事項
Business Process Choreographer Explorer	<p>選択されているデプロイメント・ターゲットで Business Process Choreographer Explorer を構成します。</p> <p>Business Process Choreographer Explorer は、同じコラボレーション単位内で構成されている Business Process Choreographer Container を管理する Web アプリケーションです。</p> <p>これには、以前は <i>Business Process Choreographer Observer</i> と呼ばれていたオプションのレポート作成機能 (Business Process Choreographer Explorer レポート作成) が組み込まれています。</p>	Business Process Choreographer Container	<p>同じコラボレーション単位内の Business Process Choreographer Container 構成を選択すると、Business Process Choreographer Explorer 構成が使用可能になります。</p> <p>Web アプリケーション・サポート用にデプロイメント・ターゲットを構成する必要があります。</p> <p>任意のデプロイメント・ターゲットで、必要な数の Business Process Choreographer Explorer のインスタンスを構成できます。コンテナが構成されているコラボレーション単位にデプロイメント・ターゲットを追加し、Business Process Explorer の構成コントロールをチェックします。</p>

表 5. デプロイメント環境のコンポーネントの関係 (続き)

コンポーネント	目的	関連コンポーネント	考慮事項
Business Process Choreographer Event Collector	<p>選択されているデプロイメント・ターゲットで Business Process Choreographer Event Collector を構成します。</p> <p>Business Process Choreographer Event Collector は、同じコラボレーション単位内で構成されている Business Process Choreographer Container から発行される Common Base Event を収集します。監視対象 Container に関する統計情報がデータベースに記録されます。</p>	<p>Business Process Choreographer Container</p> <p>Common Event Infrastructure</p>	<p>最初に、Business Process Choreographer Event Collector に使用するのと同じデプロイメント・ターゲットで Common Event Infrastructure サーバーを構成します。Business Process Choreographer Event Collector を使用可能にするには、同じコラボレーション単位内で Business Process Choreographer Container を構成する必要があります。</p> <p>特定の Business Process Choreographer Container を監視する必要があるかどうかはわからない場合は、この機能を後で構成できます。</p>
ビジネス・ルール・マネージャー	<p>選択されているデプロイメント・ターゲットでビジネス・ルール・マネージャーを構成します。</p> <p>ビジネス・ルール・マネージャーでは、ビジネス・プロセスの動作を決定するビジネス・ルールを構成できます。</p>	<p>Service Component Architecture</p>	<p>同じデプロイメント・ターゲットで SCA サポートを構成すると、ビジネス・ルール・マネージャー構成コントロールが使用可能になります。</p> <p>ビジネス・ルール・マネージャーは、1つのデプロイメント環境で1つのみ構成できます。</p> <p>1つのインスタンスでセル全体のビジネス・ルール構成を管理できるため、システムで構成する必要があるビジネス・ルール・マネージャーは1つのみです。</p>

## デプロイメント環境の作成

デプロイメント・パターンを選択したら、「デプロイメント環境構成」ウィザードを使用して、そのパターンに基づいたデプロイメント環境を作成します。

### 始める前に

デプロイメント・マネージャーの管理コンソールで、「サーバー」>「デプロイメント環境」にナビゲートします。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

### このタスクについて

このタスクは、特定のパターンに基づいたデプロイメント環境を作成し、「デプロイメント環境構成」ウィザードを使用します。

**注:** ウィザードで作業中に操作を間違えた場合は、「戻る」をクリックして前に戻ることができます。

### 手順

1. 「デプロイメント環境」ページで「新規」をクリックして、「デプロイメント環境構成」ウィザードを起動します。

システムは、「デプロイメント環境構成」ウィザードの最初のページを表示します（「新規デプロイメント環境の作成」が選択されています）。

2. デプロイメント環境に固有の名前を付けて、リストからランタイムの機能を選択し、「次へ」をクリックします。

「ランタイムの機能」に、次のいずれの値を指定します。

- **WESB.** これを指定すると、メディエーションをサポートするデプロイメント環境が作成されます。
- **WPS.** これを指定すると、メディエーション、ビジネス・プロセス、ヒューマン・タスク、およびビジネス・ルールをサポートするデプロイメント環境が作成されます。

「ランタイムの機能」のデフォルト値はデプロイメント・マネージャーの機能に一致します。

3. 「デプロイメント環境パターン」ページで、このデプロイメント環境のデプロイメント環境パターンを選択して、「次へ」をクリックします。

このパターンは、計画フェーズでこの環境に選択したパターンに一致させます。パターンについてよく理解していない場合は、『デプロイメント環境パターン』を参照してください。

**注:** カスタム・パターンを選択した場合、ユーザーがカスタム・デプロイメント環境構成を定義します。詳しくは、『デプロイメント環境パターン』を参照してください。

4. 「ノードの選択」ページで、このデプロイメント環境に含めるノードを選択して、「次へ」をクリックします。

ノードを含めるには、ノード名の隣にあるチェック・ボックスを選択します。選択されているノードを別のノード名にマップするには、「ノード・マッピング」を使用します。

5. 「クラスター」ページで、デプロイメント環境の各機能に使用する各ノードのクラスター・メンバーの数を割り当てます。

デフォルトでは、各機能について、各ノード上に 1 つのクラスター・メンバーが割り当てられます。この数を変更するには、各列の数値を置換します。機能についてよく理解していない場合は、『デプロイメント環境の機能』を参照してください。ノードが 0 (ゼロ) の場合、ノードはその機能に寄与しません。

6. 「データベース」ページで、デプロイメント環境のデータベースを構成して、「次へ」をクリックします。

このページで、このデプロイメント環境に組み込まれたコンポーネントに対してデータベース情報を定義します。可能であれば、ウィザードにパラメーターのデフォルト情報が表示されますが、環境の計画時に定義した値に合うようにそれらの値を変更してください。

7. オプション: Business Process Choreographer の構成を定義して、「次へ」をクリックします。

このページでは、以下の値を指定します。

- コンテキスト・ルート
- セキュリティー・ロール
- 認証別名
- Human Task Manager メール・セッション (必要な場合)

**注:** このページは、「WPS」値が「ランタイムの機能」に選択されている場合のみ、表示されます。

8. オプション: 「ビジネス・ルール・マネージャー」ページで、ビジネス・ルール・マネージャーのコンテキスト・ルートを指定して、「次へ」をクリックします。

**注:** このページは、「WPS」値が「ランタイムの機能」に選択されている場合のみ、表示されます。

ウィザードは「要約」ページを表示します。

9. 「要約」ページの情報が正しいことを確認したあと、「終了して環境を生成」をクリックして、デプロイメント環境の構成を保存して完了します。構成を完了せずに終了するには、「終了」をクリックします。

## タスクの結果

構成が完了すると、構成ファイルを調べて、変更を表示できます。

## 次のタスク

変更をマスター構成に保管または廃棄します。

### 関連概念

10 ページの『ビジネス・ルール・マネージャー』  
ビジネス・ルール・マネージャーは、ビジネス・アナリストがビジネス・ルール値を表示および変更する際に使用する Web ベースのツールです。このツールは、WebSphere Process Server のオプションであり、プロファイル作成時またはサーバーのインストール後に選択してインストールできます。

### 関連情報

-  デプロイメント環境構成の一部としてのビジネス・スペースの構成
-  ビジネス・スペースのデータベース表の構成
-  デプロイメント環境パターン
-  デプロイメント・パターン内のデプロイメント環境機能
-  データベース仕様書
-  Business Process Choreographer の構成

---

## デプロイメント・トポロジーの表示

IBM 提供パターンの構成情報を確認するには、「デプロイメント・トポロジー」ページを使用します。

### 始める前に

デプロイメント・マネージャーの管理コンソールに移動して、「サーバー」 → 「デプロイメント環境」 → 「*deployment\_environment\_name*」 → 「プロパティ」 → 「デプロイメント・トポロジー」をクリックします。

### このタスクについて

このページを使用して、必要に応じてデプロイメント環境にノードを追加します。各ノードの特定の機能に関係するクラスター・メンバーの数を変更することもできます。

- 目的を選択し、関連付けられたアクションを実行します。

目的	アクション
名前なしノードまたは空のノードの追加	「名前なしノード」をクリックし、「追加」をクリックします。
新規ノードの追加	「新規ノード」をクリックし、ノード名を入力し、「追加」をクリックします。
既存のノードの追加	「既存のノード」をクリックし、リストからノード名を選択し、「追加」をクリックします。

目的	アクション
デプロイメント環境におけるノードの置換	置換するノードの横の「選択」をクリックし、選択したノードを置換する方法に基づいて「名前なしノード」、「新規ノード」、または「既存のノード」のいずれかをクリックし、「選択したものを置換」をクリックします。
各機能に関するクラスター・メンバーの数の変更	ラベルの付いた列の下の入力フィールドに数を入力します。 <ul style="list-style-type: none"> <li>アプリケーション・デプロイメントのターゲット</li> <li>メッセージング・インフラストラクチャー</li> <li>サポート・インフラストラクチャー</li> </ul> <b>要確認:</b> 各機能には 1 つ以上のクラスター・メンバーが割り当てられている必要があります。

- 更新内容を保持し、引き続き「デプロイメント・トポロジー」ページを表示するには、「適用」をクリックします。更新内容を保持し、前のページに戻るには、「OK」をクリックします。

### 次のタスク

変更を保存または破棄します。

## デプロイメント環境の編集

デプロイメント環境の要件の発展とともに、デプロイメント環境を編集してリソースを追加することができます。

### 始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールで、「サーバー」>「デプロイメント環境」にナビゲートします。
- デプロイメント環境からノードを削除する前に、削除するノードを完全に停止する必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

### このタスクについて

特定のデプロイメント環境を構成するリソースを変更するように、デプロイメント環境を編集することができます。以下のことが可能です。

- サーバーおよびクラスターの追加または除去。
- 特定の機能に参加するノードの変更。

- データ・ソースの構成の変更。
- 認証別名の変更。
- データベースまたはテーブルの構成方法に関する情報の取得 (操作を遅延している場合)。

#### 手順

1. 編集するデプロイメント環境を選択するには、名前をクリックします。 システムは、「デプロイメント環境構成」ページを表示して、以下の内容をリストします。
  - デプロイメント環境
  - デプロイメント環境パターン
  - 説明
  - デプロイメント環境の状況
  - デプロイメント環境の機能
  - 構成ページへのリンク
2. 変更するデプロイメント環境の構成領域を選択します。変更が完了するまで、各リンクを選択します。

構成の領域	使用可能なアクション
「追加プロパティ」	<p><b>デプロイメント・トポロジー</b> IBM 提供パターンをベースにしたデプロイメント環境の構成を変更するには。</p> <p><b>据え置かれた構成</b> このデプロイメント環境の構成を完了するために必要な手動ステップを決定するには。</p>
関連項目	<p><b>データ・ソース</b> デプロイメント環境内のさまざまなコンポーネントのデータ・ソース構成を変更するには。</p> <p><b>認証別名</b> デプロイメント環境内のコンポーネントの認証別名またはパスワードを変更するには。</p>

3. 構成を完了するには、必要な結果に応じて、オプションを選択します。

注: 「環境の生成」をクリックするまでは、構成は完了しません。

アクション	結果
「OK」または「適用」をクリックします。	両方のオプションとも、構成を保存します。「適用」をクリックしたときは現在のページのままですが、「OK」をクリックしたときは「デプロイメント環境」ページに戻ります。

アクション	結果
「環境の生成」をクリックします。	構成を保存して、構成プロセスを開始します。 注: デプロイメント環境が最低限の制約を満たしていない場合、またはデプロイメント環境が不完全の場合は、このオプションを選択できません。

## 次のタスク

デプロイメント環境を管理します。

### 関連情報

-  デプロイメント環境トポロジーの更新
-  カスタム・デプロイメント環境の構成
-  デプロイメント環境のデータ・ソースの構成
-  デプロイメント環境の認証別名の構成
-  デプロイメント環境の据え置かれた構成の構成

---

## データ・ソース構成の編集

デプロイメント環境を作成したら、データ・ソース構成を編集できます。「データ・ソース」ページには、デプロイメント環境のすべてのデータ・ソースがリストされ、このページで複数の編集を実行できます。

### 始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールに移動して、「サーバー」 → 「デプロイメント環境」 → 「*deployment\_environment\_name*」 → 「関連項目」 → 「データ・ソース」をクリックします。

**このタスクに必要なセキュリティー・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

### このタスクについて

「データ・ソース」ページで、デプロイメント環境のすべてのデータ・ソースのコレクションを編集できます。このページでは、データ・ソースの編集はできますが、新規のデータ・ソースを追加することはできません。テキスト・フィールドの数は、各データ・ソースのコンポーネントおよびデータ・ソース・プロバイダーによって異なる場合があります。

**重要:** 別のデータ・ソースによって使用されるスキーマ名を使用しているなどの競合する編集を行う場合、警告メッセージが表示されます。変更を保存することはできませんが、競合が解決するまでメッセージはそのまま表示されます。

#### 手順

1. 「データ・ソース」ページで、編集するデータ・ソースを含むコンポーネントを選択します。
2. 必要に応じて変更を加えます。
3. 「適用」または「OK」をクリックして変更を保存します。

#### 関連情報

 デプロイメント環境のデータ・ソースの構成

## データベース・プロバイダーの編集

「データベース・ソース・プロバイダー構成」ページを使用して、データベース・プロバイダーに変更を加えます。

#### 始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールに移動して、「サーバー」 → 「デプロイメント環境」 → 「*deployment\_environment\_name*」 → 「関連項目」 → 「データ・ソース」をクリックします。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

#### このタスクについて

データ・ソースが使用するデータベース・プロバイダーの構成を変更する必要がある場合は、以下の手順を実行します。「データベース・プロバイダー構成」ページの一部のセクション（「コンポーネント固有のプロパティ」など）では、データベース・プロバイダーによってテキスト・ボックスの数が異なる場合があります。

#### 手順

1. 「データ・ソース」ページで、編集するデータ・ソース・プロバイダーのデータ・ソースを選択します。
2. 「プロバイダーの編集 (Edit Provider)」をクリックして、「データベース・ソース・プロバイダーの構成」ページを開きます。
3. 必要に応じて変更を加えます。
4. 「適用」または「OK」をクリックして、変更を保存します。

#### 関連情報

 データベース仕様書

 共通データベースの仕様

## デプロイメント環境内のデータ・ソースの編集

データ・ソースのプロパティを編集するには、「データ・ソース」ページを使用します。

### 始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールに移動して、「サーバー」 → 「デプロイメント環境」 → 「*deployment\_environment\_name*」 → 「関連項目」 → 「データ・ソース」をクリックします。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

### このタスクについて

この手順を使用してデプロイメント環境が使用しているデータ・ソースを、「データ・ソース」ページで更新できます。「データ・ソース」ページでは、新しいデータ・ソースを追加することはできません。データ・ソース情報を編集するには、データ・ソース名をクリックするか、またはコンポーネントを選択します。一部のテキスト・ボックスは使用不可になっており、それらの値を変更することはできません。

**重要:** 別のデータ・ソースによって使用されるスキーマ名を使用しているなどの競合する編集を行う場合、警告メッセージが表示されます。変更を保存することはできませんが、競合が解決するまでメッセージはそのまま表示されます。

### 手順

1. 「データ・ソース」ページで、変更するデータ・ソースを含むコンポーネントを選択し、「編集」をクリックします。
2. 該当する情報を編集します。
3. 「適用」または「OK」をクリックして、変更を保存します。

---

## デプロイメント・マネージャーの停止と再始動

デプロイメント・マネージャーで構成を変更した後は、変更を有効にするために、デプロイメント・マネージャーを停止してから再始動する必要があります。

### 始める前に

デプロイメント・マネージャーを開始し、管理コンソールにログインしていることを確認してください。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

### 手順

1. デプロイメント・マネージャーの停止方法を選択します。

方法	アクション
管理コンソールの使用	<ol style="list-style-type: none"> <li>1. 「システム管理」 → 「デプロイメント・マネージャー」にナビゲートします。</li> <li>2. 「停止」をクリックします。</li> </ol>
コマンド行の使用	<ol style="list-style-type: none"> <li>1. デプロイメント・マネージャーの <code>profile_root/bin</code> ディレクトリーにナビゲートします。</li> <li>2. ご使用のオペレーティング・システムの <code>stopManager</code> コマンドを入力します。 注: 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力を求めるプロンプトがシステムによって表示されます。</li> </ol>

2. デプロイメント・マネージャーの停止確認を待機します。
3. デプロイメント・マネージャーの `profile_root/bin` ディレクトリーにナビゲートします。
4. ご使用のオペレーティング・システムの `startManager` コマンドを入力します。

注: 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力を求めるプロンプトがシステムによって表示されます。

## 次のタスク

アプリケーション・デプロイメント・ターゲットのクラスターが開始可能なことを確認します。

---

## クラスター・メンバーの停止および再始動

構成を変更した場合、クラスター・メンバーを停止して、再始動します。

### 始める前に

1. 以下の手順で、新規処理がクラスター・メンバーに割り振られないようにします。
  - IBM® HTTP Server を使用している場合は、`plugin_cfg.xml` ファイルを変更して、HTTP トラフィック用のクラスター・メンバーを除去します。別の HTTP サーバーを使用している場合は、ご使用の HTTP サーバーの説明に従ってクラスター・メンバーを除去します。
  - IOP トラフィックの場合、クラスター・メンバーのランタイム・ウェイトをゼロに設定します。
  - サービス統合バスを静止します。
2. クラスター・メンバーに定義された作業が完了していることを確認します。ある期間待機するか、または Performance Monitoring Infrastructure のカウンターを使用して、キューに入れられたすべての処理をクラスターが完了するタイミングを判断します。

## このタスクについて

構成変更を有効にするために、サーバー・プロセスの停止および再始動が必要な場合があります。これには、デプロイメント・マネージャー、クラスター・メンバー、およびノード・エージェントの停止および再始動が含まれます。

**注:** すべてのコマンド・ファイル (管理コンソールの使用に対する代替) は `install_root/bin` サブディレクトリーにあります。

### 手順

1. 管理コンソールで、「サーバー」 → 「アプリケーション・サーバー」をクリックします。
2. 停止するサーバーまたはクラスター・メンバーを選択し、「停止」をクリックします。
3. サーバーまたはクラスター・メンバーが停止するまで待機します。
4. 再始動するサーバーまたはクラスター・メンバーを選択し、「開始」をクリックします。
5. サーバーまたはクラスター・メンバーが開始されるまで待機します。

**注:** 代替方法として、オペレーティング・システムでコマンド行から `stopServer` および `startServer` コマンドを使用するか、または管理コンソールのクラスター・パネルで「サーバー」 → 「クラスター」 → `cluster_name` → `cluster_member_name` を選択することによってクラスター・メンバーを停止および再始動することができます。

---

## デプロイメント環境の開始と停止

IBM 提供のパターンに基づいて、管理コンソールから直接デプロイメント環境を開始または停止することができます。この手順で、カスタム・デプロイメント環境を管理することはできません。

### 始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールに移動し、「サーバー」 → 「Deployment Environments」をクリックします。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

デプロイメント環境を開始または停止するには、デプロイメント環境が存在しなければなりません。

### このタスクについて

IBM 提供のパターンに基づいて、デプロイメント環境を開始または停止する場合は必ず、以下のステップを使用します。

注: カスタム・デプロイメント環境を開始または停止するには、個々のクラスターを開始および停止する必要があります。

#### 手順

1. 開始および停止するデプロイメント環境の名前の隣にあるチェック・ボックスを選択します。
2. 以下のいずれかのアクションを行います。

アクション	結果
「開始」をクリックします。	デプロイメント・マネージャーにより、デプロイメント環境を構成するクラスターが始動されます。
「停止」をクリックします。	デプロイメント・マネージャーにより、デプロイメント環境を構成するクラスターが停止されます。

注: このプロセスは、ご使用のデプロイメント環境のサイズによっては数分かかる場合があります。

#### タスクの結果

画面が最新表示され、デプロイメント環境の状況が示されます。

##### 関連情報

-  Administration Thin Client の使用
-  デプロイメント環境の状況情報
-  デプロイメント環境の機能の状況 (Deployment environment function status)
-  デプロイメント環境の状況
-  startDeploymentEnv コマンド
-  stopDeploymentEnv コマンド

---

## 管理コンソールを使用したデプロイメント環境定義のエクスポート

デプロイメント環境定義をエクスポートすると、各デプロイメント・マネージャー上の構成を最小限に抑えることができるので、デプロイメント環境をより迅速に実装できるようになります。エクスポートしたデプロイメント環境は、デプロイメント環境のテンプレートとして、他のデプロイメント・マネージャー上で使用できます。また、これにより同じデプロイメント環境構成を大規模に複製できます。

#### 始める前に

- 1 つ以上のデプロイメント環境をデプロイメント・マネージャー上で定義します。

- デプロイメント環境定義のエクスポート元のデプロイメント・マネージャーの管理コンソールにログインします。
- デプロイメント・マネージャーの管理コンソールで、「サーバー」>「デプロイメント環境」にナビゲートします。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

## このタスクについて

同じ設計に基づくデプロイメント環境を多数実装する場合は、それらのデプロイメント環境定義をエクスポートすることにより、他のデプロイメント・マネージャー上でデプロイメント環境のテンプレートとして使用できます。

### 手順

1. 「デプロイメント環境」ページで、エクスポートするデプロイメント環境定義の隣にあるチェック・ボックスを選択します。
2. 「エクスポート」をクリックします。 システムの応答は、以下のいずれを選択したかに応じて異なります。

エクスポートするデプロイメント環境の数	アクション
1 つ	プロンプトで、エクスポートされたファイルの名前を入力します。デフォルトの名前は、 <i>deployment_environment_name.xml</i> です。デフォルト名を変更するには、絶対ファイル・パスを指定します。
複数	プロンプトで、デプロイメント環境定義を含む、エクスポートする圧縮ファイルを配置する出力ディレクトリーを入力します。デフォルトでは、システムは、圧縮ファイルの名前を <i>first_env_name.zip</i> のように設定します。デフォルト名を変更するには、絶対ファイル・パスを指定します。 <b>注:</b> 圧縮ファイルを直接インポートすることはできません。デプロイメント環境定義を抽出してから、それをターゲット・ファイル・システムに取り込む必要があります。

3. システムがファイルを作成済みであることを確認します。

## 次のタスク

エクスポートしたファイルを他のデプロイメント・マネージャーにインポートできます。

---

## コマンド行を使用したデプロイメント環境定義のエクスポート

`wsadmin` コマンドを使用して、デプロイメント環境定義をエクスポートすることができます。`wsadmin` コマンドを使用して、管理コンソールで実行するものと同じ定義エクスポート・タスクを実行できます。この機能により、スクリプトを使用して、大量のデプロイメント環境定義をデプロイメント・マネージャーからエクスポートできます。これにより、管理コンソールを他のタスク用に空け、作業構成を別のデプロイメント・マネージャーに複製することが可能です。

### 始める前に

デプロイメント環境定義のエクスポート元のデプロイメント・マネージャーを起動しておく必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者権限またはオペレーター権限を持つユーザー ID とパスワードを使用する必要があります。

### このタスクについて

コマンド行を使用して、デプロイメント環境定義をエクスポートできるのは、以下の場合です。

- 複数のデプロイメント環境定義をエクスポートし、コマンド行を使用する必要がある場合。
- コマンド行を使用して、1つのデプロイメント環境定義をエクスポートしたい場合。
- 大量のデプロイメント環境定義をエクスポートする必要がある場合 (`wsadmin` を使用すれば、タスクの実行時間を削減できます)。

### 手順

1. コマンド・ウィンドウを開きます。

`wsadmin` コマンドは、`<WPS>/profiles/<dmgr profile>/bin` ディレクトリーまたは `<WPS>/bin` ディレクトリーにあります。

2. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を入力します。
3. `exportDeploymentEnvDef` コマンドを使用して、デプロイメント・マネージャーから出力ファイルにデプロイメント環境定義をエクスポートします。ファイル名の形式は `depEnvName.xml` です。

**注:** 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力を求めるプロンプトが表示されます (コマンド内で指定していない場合)。

### 例

この例では、管理セキュリティが有効なホスト `myDmgr` のデプロイメント環境 `myDepEnv` がエクスポートされます。

注: デプロイメント・マネージャーの bin フォルダから管理クライアントを実行している場合は、コマンドに `-host` パラメーターと `-port` パラメーターを組み込む必要はありません。

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgradmin -password dmgrpass  
> $Admintask exportDeploymentEnvDef {-filePath c:/dmgr01/DeploymentEnvs  
-topologyName myDepEnv}
```

`-connType` パラメーターでは、使用する接続のタイプを指定します。デフォルトの引数は SOAP です。

注: デフォルトが SOAP なので、使用する接続タイプが SOAP の場合は、値を明示的に指定する必要はありません。

`-host` パラメーターでは、SOAP または RMI の接続で使用するホストを指定します。`-host` のデフォルト値はローカル・ホストです。

注: ローカル・ホストでノードを実行している場合は、`-host` を指定する必要はありません。

注: 管理セキュリティが無効になっている場合は、ユーザー ID とパスワードを指定する必要はありません。

#### 関連情報



exportDeploymentEnvDef コマンド

---

## 管理コンソールを使用したデプロイメント環境定義のインポート

既存のデプロイメント環境定義を別のデプロイメント・マネージャーからインポートして、それを新しいデプロイメント環境構成の基礎として使用することができます。

### 始める前に

- デプロイメント・マネージャーの管理コンソールで、「サーバー」>「デプロイメント環境」にナビゲートします。
- 別のデプロイメント・マネージャーからエクスポートされたデプロイメント環境定義のコピーが必要です。
- デプロイメント環境定義をインポートするデプロイメント・マネージャーからデプロイメント環境定義ファイルにアクセスできる必要があります。
- デプロイメント環境定義をインポートするデプロイメント・マネージャーは、少なくともデプロイメント環境定義で定義されているすべての機能をサポートしている必要があります。例えば、WebSphere Enterprise Service Bus デプロイメント・マネージャーで作成されたデプロイメント環境定義を WebSphere Process Server デプロイメント・マネージャーにインポートできますが、この逆はできません。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

**重要:** 1 つの圧縮ファイルから複数のデプロイメント環境定義を同時にインポートすることはできません。圧縮ファイルから定義を抽出したら、それらの定義を一度に 1 つずつインポートする必要があります。

## このタスクについて

新規に作成するために既存のデプロイメント環境定義をインポートすると、デプロイメント環境の構成に費やす時間を最小限に抑えることができます。作成する環境に似た既存の環境があれば、エクスポートして、次に構成するデプロイメント・マネージャーにインポートします。

### 手順

1. 「デプロイメント環境」ページの「インポート」をクリックして、「デプロイメント環境構成」ウィザードを起動します。

このウィザードは、「外部のデプロイメント環境定義のロード」が選択された状態で起動します。

2. 「参照」をクリックして、ファイル・ダイアログを開き、インポートするデプロイメント環境定義ファイルを選択するか、またはそのファイルへの絶対パスを入力します。
3. 「次へ」をクリックします。構成が読み込まれ、「デプロイメント環境のインポート」ウィザードが開始されます。

すべてのノード名が現在の統合ノードに一致しない場合、ウィザードは「ノードの選択」ページを表示します。すべてのノードが一致する場合、ウィザードは「データベース」ページを表示します。

**重要:** ウィザードのどのパネルでも、「構成」をクリックすると、現行値を使用してデプロイメント環境が構成されます。

4. オプション: 「ノードの選択」ページで、使用可能なノードのリストから、デプロイメント環境に含めるノードを選択して「次へ」をクリックします。

ノードを含めるには、ノード名の隣にあるチェック・ボックスを選択します。選択されているノードを別のノード名にマップするには、「ノード・マッピング」を使用します。

**重要:** 選択したノードが、インポートするデプロイメント環境定義によって課せられた制約を満たしていない場合は、「次へ」が使用可能になりません。例えば、デプロイメント環境に『Mandatory\_Node』という名前のノードと、それ以外に任意の名前の 3 つのノードを含めるという要件がある場合、

『Mandatory\_Node』 および、他の 3 つのノードを選択するまで続行できません。

5. 「データベース」ページで、デプロイメント環境のデータベースを構成して、「次へ」をクリックします。

このページで、このデプロイメント環境に組み込まれたコンポーネントに対してデータベース情報を定義します。可能であれば、ウィザードにパラメーターのデフォルト情報が表示されますが、環境の計画時に定義した値に合うようにそれらの値を変更してください。

6. オプション: Business Process Choreographer の構成を定義して、「次へ」をクリックします。

このページでは、以下の値を指定します。

- コンテキスト・ルート
- セキュリティー・ロール
- 認証別名
- Human Task Manager メール・セッション (必要な場合)

注: このページは、「WPS」値が「ランタイムの機能」に選択されている場合のみ、表示されます。

7. オプション: 「ビジネス・ルール・マネージャー」ページで、ビジネス・ルール・マネージャーのコンテキスト・ルートを指定して、「次へ」をクリックします。

注: このページは、「WPS」値が「ランタイムの機能」に選択されている場合のみ、表示されます。

ウィザードは「要約」ページを表示します。

8. 「要約」ページの情報が正しいことを確認したあと、「終了して環境を生成」をクリックして、デプロイメント環境の構成を保存して完了します。構成を完了せずに終了するには、「終了」をクリックします。

## タスクの結果

構成が完了すると、構成ファイルを調べて、変更を表示できます。

## 次のタスク

変更をマスター構成に保管または廃棄します。

### 関連概念

10 ページの『ビジネス・ルール・マネージャー』

ビジネス・ルール・マネージャーは、ビジネス・アナリストがビジネス・ルール値を表示および変更する際に使用する Web ベースのツールです。このツールは、WebSphere Process Server のオプションであり、プロファイル作成時またはサーバーのインストール後に選択してインストールできます。

### 関連タスク

80 ページの『管理コンソールを使用したデプロイメント環境定義のエクスポート』

デプロイメント環境定義をエクスポートすると、各デプロイメント・マネージャー上の構成を最小限に抑えることができるので、デプロイメント環境をより迅速に実装できるようになります。エクスポートしたデプロイメント環境は、デプロイメント環境のテンプレートとして、他のデプロイメント・マネージャー上で使用できます。また、これにより同じデプロイメント環境構成を大規模に複製できます。

### 関連情報

 デプロイメント環境パターン

-  デプロイメント・パターン内のデプロイメント環境機能
-  データベース仕様書
-  Business Process Choreographer の構成

---

## コマンド行を使用したデプロイメント環境定義のインポート

`wsadmin` コマンドを使用して、デプロイメント環境定義をインポートすることができます。 `wsadmin` コマンドを使用して、管理コンソールで実行するものと同じ定義インポート・タスクを実行できます。この機能により、スクリプトを使用して、大量のデプロイメント環境定義をデプロイメント・マネージャーにインポートできます。これにより、管理コンソールを他のタスク用に空け、作業構成を別のデプロイメント・マネージャーに複製することが可能です。

### 始める前に

- エクスポートしたデプロイメント環境定義のコピーを持っている必要があります。
- デプロイメント環境定義のインポート先のデプロイメント・マネージャーを起動しておく必要があります。
- インポートするデプロイメント環境定義と同名のデプロイメント環境がこのデプロイメント・マネージャーに存在しないことを確認します。
- デプロイメント環境定義をインポートするデプロイメント・マネージャーは、デプロイメント環境定義で定義されているすべての機能をサポートしている必要があります。例えば、WebSphere Enterprise Service Bus デプロイメント・マネージャーで作成されたデプロイメント環境を WebSphere Process Server デプロイメント環境にインポートできますが、この逆はできません。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者権限またはオペレーター権限を持つユーザー ID とパスワードを使用する必要があります。

### このタスクについて

コマンド行を使用して、デプロイメント環境定義をインポートできるのは、以下の場合です。

- 複数のデプロイメント環境定義をインポートし、コマンド行を使用する必要がある場合。
- コマンド行を使用して、複数のデプロイメント環境のテンプレートとして 1 つのデプロイメント環境定義をインポートしたい場合。
- 大量のデプロイメント環境定義をインポートする必要がある場合 (`wsadmin` を使用すれば、タスクの実行時間を削減できます)。

### 手順

1. コマンド・ウィンドウを開きます。

`wsadmin` コマンドは、`<WPS>/profiles/<dmgr profile>/bin` ディレクトリーまたは `<WPS>/bin` ディレクトリーにあります。

2. システムにインポートするデプロイメント環境定義ファイルをコピーします。
3. `wsadmin` コマンドを入力し、`wsadmin` 環境を入力します。
4. `importDeploymentEnvDef` コマンドを使用して、デプロイメント・マネージャーにコピーしたファイルからデプロイメント環境定義をインポートします。デプロイメント環境をインポートするときは、その名前を変更できます。

**注:** 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力を求めるプロンプトが表示されます (コマンド内で指定していない場合)。

## 例

この例では、管理セキュリティが有効になっているデプロイメント・マネージャー `myDmgr` にデプロイメント環境 `myDepEnv` がインポートされ、名前が `eastDepEnv` に変更されます。

```
wsadmin -connType SOAP -host myDmgr -port 8879 -user dmgrAdmin -password -dmgrPass  
> $AdminTask importDeploymentEnvDef {-filePath  
c:/dmgr01/importedEnvironments/myDepEnv.xml -topologyName eastDepEnv}
```

`-connType` パラメーターでは、使用する接続のタイプを指定します。デフォルトの引数は `SOAP` です。

**注:** デフォルトが `SOAP` なので、使用する接続タイプが `SOAP` の場合は、値を明示的に指定する必要はありません。

`-host` パラメーターでは、`SOAP` または `RMI` の接続で使用するホストを指定します。`-host` のデフォルト値はローカル・ホストです。

**注:** ローカル・ホストでノードを実行している場合は、`-host` を指定する必要はありません。

**注:** 管理セキュリティが無効になっている場合は、ユーザー ID とパスワードを指定する必要はありません。

## 次のタスク

**オプション:** インポートしたデプロイメント環境を妥当性検査します。

### 関連情報

 ノード・エージェントの管理

 `importDeploymentEnvDef` コマンド

---

## デプロイメント環境の除去

デプロイメント環境を除去すると、デプロイメント環境の管理エンティティが除去されます。デプロイメント環境を削除しても、デプロイメント環境を構成するサーバー、ノード、およびクラスターの構成が除去または変更されることはありません。デプロイメント環境の削除は、デプロイメント環境をあるデプロイメント・マネージャーから別のデプロイメント・マネージャーへと移動する最終段階といえます。

## 始める前に

- デプロイメント・マネージャーの管理コンソールで、「サーバー」>「デプロイメント環境」にナビゲートします。
- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- リカバリーの目的で、デプロイメント環境定義をエクスポートしておくことを検討してください。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

## このタスクについて

特定のデプロイメント環境のリソースをグループとして管理する必要がなくなったら、そのデプロイメント環境定義をデプロイメント・マネージャーから除去します。

### 手順

1. 「デプロイメント環境」ページで、除去するデプロイメント環境の横にあるチェック・ボックスを選択し、「**除去**」をクリックします。

システムは、画面からデプロイメント環境を削除します。

2. この変更内容をマスター構成に保存するには「**保存**」、マスター構成が更新されないようにするには「**破棄**」をクリックします。

### 関連タスク

80 ページの『管理コンソールを使用したデプロイメント環境定義のエクスポート』

デプロイメント環境定義をエクスポートすると、各デプロイメント・マネージャー上の構成を最小限に抑えることができるので、デプロイメント環境をより迅速に実装できるようになります。エクスポートしたデプロイメント環境は、デプロイメント環境のテンプレートとして、他のデプロイメント・マネージャー上で使用できます。また、これにより同じデプロイメント環境構成を大規模に複製できます。

---

## 第 5 章 アプリケーションおよびアプリケーション・サービスの管理

WebSphere Process Server のアプリケーションでは、WebSphere Application Server の J2EE アプリケーションの場合と同様の管理タスクとインターフェースが必要ですが、その他に、サービス・アプリケーションやサービス・モジュール、そして WebSphere MQ 宛先などのリソースに個別に関連する追加タスクがあります。

---

### サービス・アプリケーションおよびサービス・モジュールの管理

サービス・アプリケーションおよび関連したサービス・モジュールの表示および管理には、管理ツールを使用します。

#### 始める前に

サービス・モジュールをランタイム環境にデプロイします。

#### このタスクについて

サービス・モジュールは、実行時にサービスを提供する Service Component Architecture (SCA) モジュールです。サービス・モジュールを WebSphere Process Server にデプロイするときには、Enterprise ARchive (EAR) ファイルとしてパッケージされた関連サービス・アプリケーションをビルドします。

サービス・モジュールはデプロイメントの基本単位であり、関連したサービス・アプリケーションが使用するコンポーネント、ライブラリー、およびステージング・モジュールを含めることができます。サービス・モジュールにはエクスポート (オプションでインポートも) が格納され、モジュールとサービス・リクエスター/サービス・プロバイダー間の関係が定義されます。WebSphere Process Server は、ビジネス・サービスのモジュールおよびメディエーション・モジュールをサポートします。モジュールおよびメディエーション・モジュールは、いずれも SCA モジュールの一種です。メディエーション・モジュールは、サービス起動をターゲットが理解する形式に変換し、要求をターゲットに渡して結果をオリジネーターに戻すことによって、アプリケーション間の通信を可能にします。ビジネス・サービス用のモジュールは、ビジネス・プロセスのロジックを実装します。ただし、メディエーション・モジュール内にパッケージ可能なものと同じメディエーション・ロジックをモジュールに格納することもできます。

#### 関連タスク

##### 実動サーバーへのモジュールのインストール

このトピックでは、テスト・サーバーからアプリケーションを取り出して実稼働環境にデプロイするステップについて説明します。

## サービス・アプリケーションのバージョン管理

サービス・アプリケーションではバージョン管理をサポートしています。モジュールおよびその成果物のバージョンを 1 つ以上開発して、特定のクライアントが使用できるようにランタイム環境にデプロイすることができます。

### バージョン管理の可能な範囲

モジュールはバージョン番号を持つことができ、これはモジュール内の SCA インポート・バインディングおよびエクスポート・バインディングについても同じです。SCA バインディングは関連付けられたモジュールからバージョン情報を継承します。

注: 現時点では、SCA バインディングのタイプだけがバージョン管理可能です。バージョン管理は 6.2.x モジュールについてのオプションです。WebSphere Integration Developer および WebSphere Process Server 6.1.x で開発およびデプロイしたモジュールはバージョンを持つことができず、現行の性質で機能し続けます。詳しくは、マイグレーションのトピックを参照してください。

ライブラリーもバージョン管理できます。ライブラリーを使用するモジュールは、そのライブラリーの特定のバージョンに依存し、ライブラリーも他のライブラリーの特定のバージョンに依存することもあります。ライブラリーのバージョン管理について詳しくは、WebSphere Integration Developer インフォメーション・センターを参照してください。

### バージョン指定したモジュールをデプロイするときの考慮事項

バージョン指定したモジュールを 6.2.x ランタイムにデプロイして、管理コンソールの「SCA モジュール」ページから管理できます。WebSphere Process Server では以下に示すバージョン指定のデプロイメント・シナリオがサポートされています。

- バージョン管理されたモジュールを、セル内のサーバーまたはクラスターにインストールする
- あるモジュールの同一バージョンを、セル内の 1 つ以上のサーバーまたはクラスターにそれぞれ 1 回インストールする
- あるモジュールの異なるバージョンを同一サーバーまたは同一クラスターにインストールする

モジュールの新規バージョンをデプロイしても、モジュールの以前のバージョンは置換されません。セルを対象範囲としたアプリケーション成果物 (このケースではビジネス・ルール) の以前のバージョンは上書きされます。

バージョンを変更せずにアプリケーションを更新する (例えば、小さな修正または改良を加える) 場合、更新されたアプリケーションおよびその成果物は既存のアプリケーションおよび成果物を置換しますが、定義済みのセキュリティー・ポリシーについては例外です。すべてのセキュリティー・ポリシー成果物はアプリケーション更新中に保持されます。

バージョン管理情報を保持するために、インストール・プロセスではモジュール名が (serviceDeploy または createVersionedSCAModule コマンドによって) 自動的に変

更され、モジュール名がセル内で固有であるようにします。この変更は、バージョン番号、固有のセル ID、またはその両方を元のモジュール名に追加することによって行われます。

`moduleName_vversionValue_uniqueCellID`

### バージョン指定したモジュールをバインドするときの考慮事項

あるモジュールの複数バージョンを 1 つのサーバーにデプロイするか、またはあるモジュールの複数インスタンスをクラスター全体にデプロイした後、モジュールの特定バージョンをクライアント (バージョンの指定ありまたはなし) にどのようにバインドするかについて検討します。

- 静的バインディング: 静的バインディングを使用している場合、単に既存の管理ツールを使用して、バージョン指定したモジュールをクライアントにバインドします。静的バインディングではモジュールのバージョン番号を指定する必要があります。
- 動的バインディング: バージョン指定されたモジュールで動的バインディングを使用するには、モジュール・バージョンのメタデータ (`versionValue` および `versionProvider`) およびサービス・バージョンを認識するルーティング機能を含む、メディエーション・フロー・コンポーネントを使用します。サービス・バージョンを認識するルーティング機能を使用して、バージョン指定したモジュールを動的にバインドするには、すべてのモジュールを WebSphere Service Registry and Repository (WSRR) で登録しておく必要があります。

## 管理インターフェースのサービス・アプリケーション機能

WebSphere Process Server では、管理コンソールを使用してサービス・アプリケーションおよびサービス・モジュールの特徴を表示および変更することができます。

サービス・アプリケーションによってサービスが提供されます。サービス・アプリケーションは、サービス・モジュール (Service Component Architecture (SCA) モジュールとも呼ばれる) と関連付けられています。

### 表示可能なモジュールの詳細

SCA モジュールを含む EAR (Enterprise ARchive) ファイルをデプロイした後は、SCA モジュールの詳細を表示できます。すべての SCA モジュールと関連アプリケーションをリストし、特定の SCA モジュールの詳細を表示できます。

表示可能な SCA モジュールの詳細には、以下のような情報が含まれます。

- SCA モジュール名
- 関連アプリケーション
- SCA モジュール インポート:
  - インターフェース
  - バインディング
- SCA モジュール エクスポート:
  - インターフェース
  - バインディング
- SCA モジュール・プロパティー

## 変更可能なモジュールの詳細

SCA モジュールを含む EAR ファイルをデプロイ済みの場合は、EAR ファイルを再デプロイしなくても、管理コンソールを使用して、以下の SCA モジュールの詳細を変更できます。

- SCA タイプのインポート・バインディング
  - インポート・バインディングを変更することにより、サービス対話を変更できます。
  - SCA バインディングでは、SCA モジュールを他の SCA モジュールに接続します。1 つの SCA モジュールは、2 番目の SCA モジュールと相互作用でき、別の SCA モジュールと相互作用するように変更することもできます。
  - Web サービス・バインディングは、SOAP を使用して SCA モジュールを外部サービスに接続します。
- Web サービス (WS) タイプのインポート・バインディング
  - インポート・バインディングを変更すると、サービス対話を変更できるようになります。
  - WS インポート・バインディングによって SCA モジュールは Web サービスにアクセスできます。WS インポート・バインディングは、指定されたエンドポイントにあるサービスを呼び出します。エンドポイントを変更してバインディングが別のエンドポイントにあるサービスを呼び出すようにすることができますほか、インターフェースに互換性があるまったく別のサービスを呼び出すようにすることもできます。
- JMS、WebSphere MQ JMS、汎用 JMS、WebSphere MQ、および HTTP の各タイプのエクスポート・バインディングおよびインポート・バインディングの変更できる属性
- メディエーション・モジュール・プロパティ
  - メディエーション・モジュールのプロパティは、関連付けられているメディエーション・プリミティブに属します。ただし、WebSphere Process Server 管理コンソールでは、これらのプロパティの一部は SCA モジュールの追加プロパティとして表示されます。WebSphere Process Server からメディエーション・プリミティブ・プロパティを表示できるようにするには、Integration Developer がプロパティにプロモート済みというフラグを立てる必要があります。
  - メディエーション・モジュール・プロパティを変更することにより、メディエーションの動作を変更できます。行うことができるメディエーションの変更は、プロモートされたプロパティによって異なります。

**注:** バインディングが指定されていないエクスポートは、ランタイムにより SCA バインディングを持つエクスポートと解釈されます。

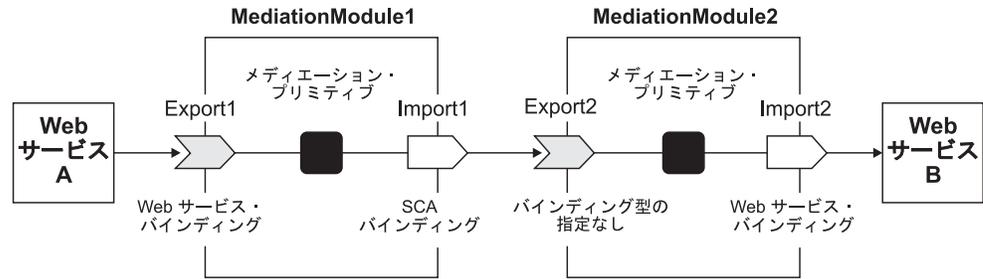


図2. 別のメディエーション・モジュールと対話するメディエーション・モジュールの例：  
Mediation Module1 は Mediation Module2 に接続します。

## サービス・モジュールの管理

WebSphere Process Server にデプロイされたサービス・モジュールをリストし、個々のサービス・モジュールに関連する情報を表示して、一部のインポート・バインディングを変更することができます。

### このタスクについて

サービス・アプリケーションをデプロイした後は、管理コンソールを使用して、関連するすべてのサービス・モジュール（メディエーション・モジュールを含む）をリストおよび管理します。

#### 手順

1. 管理コンソールを開きます。
2. 「アプリケーション」 → 「SCA モジュール」をクリックし、使用可能なサービス・モジュールをリストします。

### タスクの結果

WebSphere Process Server にデプロイされているサービス・モジュールが、コンテンツ・ペインに表示されます。また、モジュールが関連付けられているアプリケーション、およびアプリケーションが実行されているかどうかを確認することができます。

### サービス・モジュール詳細の表示

WebSphere Process Server にデプロイされたサービス・モジュールに関する情報を表示できます。

### このタスクについて

デプロイ済みのサービス・モジュールに関する詳細を表示するには、管理コンソールを使用して、以下のステップを実行します。

#### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開して、SCA モジュールを表示します。
2. コンテンツ・ペインで SCA モジュールをクリックして、SCA モジュールを選択します。

## タスクの結果

コンテンツ・ペインには、SCA モジュールの名前と説明、関連するエンタープライズ・アプリケーションの名前、インポートおよびエクスポートの展開可能なリスト、およびモジュール・プロパティ・リンクが表示されます。

## サービス・モジュール用アプリケーション詳細の表示

サービス・モジュールを WebSphere Process Server にデプロイするために使用するアプリケーションに関する詳細を表示できます。

## このタスクについて

サービス・モジュールのデプロイに使用されるアプリケーションにより、モジュールおよび関連コンポーネントの使用に影響する構成プロパティの範囲が定義されます。このアプリケーションをインストールした際に、そのプロパティの値のすべてではないがほとんどを指定しています。

アプリケーションのインストール後にプロパティを検討して、必要があればいくつかの値を変更します。

サービス・モジュールのデプロイに使用するアプリケーションの詳細を表示するには、管理コンソールを使用して以下のステップを実行します。

### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールを表示します。
2. 「アプリケーション」というラベルの付いた列で、アプリケーション名をクリックして SCA モジュールを選択します。

## タスクの結果

コンテンツ・ペインにはアプリケーションの詳細ページが表示され、このページに、アプリケーションの構成プロパティとローカル・トポロジー (該当する場合) が記載されています。

## 次のタスク

『アプリケーションの構成』で説明されているように、このページから、アプリケーション用の構成プロパティを確認し、必要に応じて変更したり、その他のコンソール・ページにリンクしたりすることができます。

## サービス・モジュールの開始と停止

「停止」状況のサービス・モジュールを開始したり、「開始済み」状況のサービス・モジュールを停止したりできます。サービス・モジュールの状況を変更するには、モジュールをデプロイするために使用されるアプリケーションを開始または停止します。

## 始める前に

サービス・モジュールに使用するアプリケーションを開始または停止するには、事前にそのモジュールを WebSphere Process Server にデプロイしておく必要があります。

## このタスクについて

サービス・モジュールおよび関連するコンポーネントのサービスを使用するには、関連するアプリケーションを開始します。デフォルトでは、サーバーの始動時にアプリケーションが自動的に開始します。

手動でのアプリケーションの開始および停止には、以下の管理ツールを使用できます。

- 管理コンソール
- `wsadmin startApplication` コマンドおよび `stopApplication` コマンド
- ApplicationManager または AppManagement MBeans を使用する Java プログラム

サービス・モジュールを開始または停止するには、管理コンソールを使用して以下のステップを実行します。

### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. 開始または停止する SCA モジュールのチェック・ボックスを選択します。
3. 「開始」ボタンまたは「停止」ボタンをクリックします。

オプション	説明
開始	アプリケーションを実行して、アプリケーションの状態を「開始済み」に変更します。アプリケーションをデプロイしているサーバーのうち、一部のサーバーが稼働していない場合は、状態が「一部開始済み ( <i>partially started</i> )」に変更されます。
停止	アプリケーションの処理を停止して、アプリケーションの状態を「停止済み」に変更します。

4. 「停止」をクリックするか、再始動するアプリケーションを選択し、次に「開始」をクリックして実行中のアプリケーションを再始動します。

## タスクの結果

アプリケーションの状態が変更され、アプリケーションが開始済みまたは停止済みであることを示すメッセージがページの上部に表示されます。

## 次のタスク

アプリケーションが存在するサーバーの始動時に自動的にアプリケーションを開始するかどうかを変更できます。WebSphere アプリケーションの開始および停止の詳細については、『アプリケーションの始動と停止』を参照してください。

## サービス・モジュール・プロパティの表示

WebSphere Process Server にデプロイされたサービス・モジュールのプロパティを表示できます。

### このタスクについて

サービス・アプリケーションを実行する前に、適切なプロパティ値が設定されているかどうかチェックできます。

デプロイ済みサービス・モジュールのプロパティを表示するには、管理コンソールを使用して、以下のステップを実行します。

#### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開して、SCA モジュールを表示します。
2. SCA モジュールを選択するには、コンテンツ・ペインで必要な SCA モジュールをクリックします。
3. コンテンツ・ペインの「追加プロパティ」で「モジュール・プロパティ」をクリックし、SCA モジュール・プロパティのリストを表示します。
4. オプション: 表示するプロパティが属するグループを展開します。プロパティがグループに属している場合、拡張可能なセクションの内部に表示されます。プロパティがグループに属していない場合、すぐに表示できます。

### タスクの結果

コンテンツ・ペインで、SCA モジュールの更新可能なプロパティが、プロパティの名前、タイプ、および値を記載したテーブルに表示されます。管理コンソールから更新できるのはプロパティ値だけです。プロパティのグループ、名前、タイプを変更するには、WebSphere Integration Developer を使用します。更新できるプロパティがない場合はメッセージが表示されます。

#### 関連概念

 [メディエーション・モジュール](#)

メディエーション・モジュールは、サービス要求のフォーマット、内容、またはターゲットを変更できる Service Component Architecture (SCA) モジュールです。

#### 関連資料

 [プロモート可能なプロパティ](#)

## サービス・モジュール・プロパティの変更

一部のサービス・モジュール・プロパティは値を変更できます。

### このタスクについて

ランタイム環境の変更に合わせて、プロパティ値の変更も必要なことがあります。

サービス・モジュール・プロパティの値を変更するには、管理コンソールを使用して以下のステップを実行します。

#### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュールを選択するには、コンテンツ・ペインで SCA モジュールをクリックします。
3. 「追加プロパティ」で、コンテンツ・ペインの「モジュール・プロパティ」を選択し、SCA モジュール・プロパティを表示します。これにより、更新できるモジュール・プロパティが表示されます。コンテンツ・ペインには、プロパティのグループ、名前、タイプ、および値が表示されますが、更新できるのはプロパティ値だけです。プロパティのグループ、名前、タイプを変更するには、WebSphere Integration Developer を使用します。
4. オプション: 更新するプロパティが属するグループを展開します。プロパティがグループに属している場合、拡張可能なセクションの内部に表示されます。プロパティがグループに属していない場合、すぐに表示できます。
5. プロパティ値を選択するには、「プロパティ」テーブルでプロパティ値をクリックします。
6. プロパティ・タイプに準拠する値を入力して、プロパティ値を変更します。
7. 「OK」をクリックして変更を保存します。次に、変更をマスター構成に保管します。

#### タスクの結果

プロパティの値が変更されます。通常、プロパティの変更がデプロイメント・マネージャーのセル内で行われた場合を除き、プロパティの変更内容はメディエーション・フローによってただちに使用されます。デプロイメント・マネージャー・セル内に変更が生じた場合、セル内の各ノードで同期化が行われるとそのノード上で変更が有効になります。プロパティ値の変更時点で処理中のメディエーション・フローは、前の値を引き続き使用します。

#### 関連概念



メディエーション・モジュール

メディエーション・モジュールは、サービス要求のフォーマット、内容、またはターゲットを変更できる Service Component Architecture (SCA) モジュールです。

#### 関連資料



プロモート可能なプロパティ

#### インポートの使用

WebSphere Process Server にデプロイされたサービス・モジュールのインポートをリストできます。また、インポート・インターフェースを表示したり、インポート・バインディングの詳細を変更したりすることもできます。

## このタスクについて

デプロイしたサービス・モジュールのインポートをリストするには、管理コンソールを使用して以下のステップを実行します。

### 手順

1. ナビゲーション・ペインで、「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. 使用する SCA モジュールを選択します。
3. コンテンツ・ペインのモジュール・コンポーネントの下で「インポート」を展開して、SCA モジュール・インポートをリストします。

## タスクの結果

インポートのリストがコンテンツ・ペインに表示されます。インポートがない場合、情報メッセージが表示されます。

### 関連概念



#### インポートとインポート・バインディング

インポートは、Service Component Architecture (SCA) モジュールとサービス・プロバイダー間の対話を定義します。SCA モジュールは、インポートを使用することにより、コンポーネントがローカルの表記を使用して外部サービス (SCA モジュールの外部にあるサービス) にアクセスできるようにします。インポート・バインディングでは、外部サービスにアクセスする特定の方法を定義します。



#### HTTP バインディング

HTTP バインディングは、Service Component Architecture (SCA) と HTTP の接続を提供する目的で設計されています。これにより、既存のまたは新規作成された HTTP アプリケーションをサービス指向アーキテクチャー (SOA) 環境内に組み込むことができます。

### インポート・インターフェースの表示:

WebSphere Process Server にデプロイされたサービス・モジュールのインポート・インターフェースを表示できます。

## このタスクについて

デプロイしたサービス・モジュールのインポート・インターフェースを表示するには、管理コンソールを使用して以下のステップを実行します。

### 手順

1. ナビゲーション・ペインで、「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュールを選択します。
3. コンテンツ・ペインで、モジュール・コンポーネントの「インポート」を展開し、SCA モジュール・インポートのリストを表示します。
4. コンテンツ・ペインのモジュール・コンポーネントで、必要なインポートを展開してインポートの詳細を表示します。

5. 「インターフェース」を展開して、インポート・インターフェースを表示します。
6. インターフェースを選択します。

### タスクの結果

コンテンツ・ペインに WSDL (Web サービス記述言語) インターフェースが表示されます。

### インポート・バインディングの表示:

サービス・モジュールを WebSphere Process Server にデプロイした後は、Web サービス、SCA、JMS、アダプター各タイプのインポート・バインディングを表示できます。

### このタスクについて

サービス・モジュールのインポート・バインディングを表示するには、管理コンソールを使用して以下のステップを実行します。

### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュールを選択します。
3. コンテンツ・ペインで、モジュール・コンポーネントの「インポート」を展開し、SCA モジュール・インポートのリストを表示します。
4. コンテンツ・ペインのモジュール・コンポーネントで、必要なインポートを展開してインポートの詳細を表示します。
5. 「バインディング」を展開して、インポート・バインディングを表示します。
6. バインディングを選択します。

### タスクの結果

コンテンツ・ペインにインポート・バインディングの詳細が表示されます。

### 関連概念

223 ページの『第 9 章 バインディングの操作』

サービス指向アーキテクチャの中核をなすのは「サービス」の概念で、これはコンピューター・デバイス間での対話によって実行される機能単位のことです。

「エクスポート」はモジュールの外部インターフェース (またはアクセス・ポイント) を定義し、これによってモジュール内の SCA コンポーネントは外部クライアントにサービスを提供できます。「インポート」はモジュール外部のサービスへのインターフェースを定義し、これによってサービスはモジュール内部から呼び出すことができます。インポートおよびエクスポートとともにプロトコル固有の「バインディング」を使用して、データをモジュールの内部または外部に移送する方法を指定します。

### SCA インポート・バインディングの変更:

管理コンソールで、SCA インポート・バインディングの詳細を変更できます。その他のタイプのバインディングの詳細の変更方法については、『バインディング』のセクションを参照してください。

### このタスクについて

サービス・モジュールの SCA インポート・バインディングを変更するには、管理コンソールを使用して、以下のステップを完了します。

#### 手順

1. ナビゲーション・ペインで、「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュールを選択します。
3. モジュール・コンポーネントのコンテンツ・ペインで「インポート」を展開し、SCA モジュール・インポートのリストを表示します。
4. モジュール・コンポーネントのコンテンツ・ペインで、必要なインポートを展開し、インポートの詳細を表示します。
5. 「バインディング」を展開して、インポート・バインディングを表示します。
6. SCA インポート・バインディングを選択します。SCA インポート・バインディングは、ID [SCA] を使用して示されます。
7. 新規ターゲットの SCA モジュールを選択します。「ターゲット」ドロップダウン・メニューからモジュールを選択します。異なる SCA モジュールを選択すると、表示されるエクスポートおよびエクスポート・インターフェースが変更されます。
8. 「エクスポート」ドロップダウン・メニューからエクスポートを選択します。
9. 変更内容をマスター構成に保管します。

#### タスクの結果

選択した SCA モジュールのインポート用に、SCA インポート・バインディングが変更されました。

WebSphere Process Server では、エクスポート・インターフェースに対応しない各インポート・インターフェースに対して警告が出されます。WebSphere Process Server では、インポートとエクスポートの WSDL (Web サービス記述言語) ポート・タイプ名が比較されます。ポート・タイプ名が異なる場合は警告が出されますが、この警告は無視してかまいません。ただし、ポート・タイプ名が一致する場合、WebSphere Process Server では、行われた操作を等価とみなし警告は出されません。

#### 関連概念

223 ページの『第 9 章 バインディングの操作』

サービス指向アーキテクチャの中核をなすのは「サービス」の概念で、これはコンピューター・デバイス間での対話によって実行される機能単位のことです。

「エクスポート」はモジュールの外部インターフェース (またはアクセス・ポイント) を定義し、これによってモジュール内の SCA コンポーネントは外部クライアントにサービスを提供できます。「インポート」はモジュール外部のサービスへのインターフェースを定義し、これによってサービスはモジュール内部から

呼び出すことができます。インポートおよびエクスポートとともにプロトコル固有の「バインディング」を使用して、データをモジュールの内部または外部に移送する方法を指定します。

### Web サービス・インポート・バインディングの変更:

管理コンソールを使用して、Web サービス・インポート・バインディングのエンドポイント URL を変更できます。その他のタイプのバインディングの詳細の変更方法については、『バインディング』のセクションを参照してください。

#### このタスクについて

サービス・モジュールの Web サービス・インポート・バインディングを変更するには、管理コンソールを使用して、以下のステップを実行します。

#### 手順

1. ナビゲーション・ペインで、「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュールを選択します。
3. モジュール・コンポーネントのコンテンツ・ペインで「インポート」を展開し、SCA モジュール・インポートのリストを表示します。
4. モジュール・コンポーネントのコンテンツ・ペインで、必要なインポートを展開し、インポートの詳細を表示します。
5. 「バインディング」を展開して、インポート・バインディングを表示します。
6. Web サービス・インポート・バインディングを選択します。Web サービス・インポート・バインディングは、ID [Web service] によって示されます。
7. エンドポイント URL を変更します。エンドポイントが正しい形式の URL であることを確認します。
8. 変更内容をマスター構成に保管します。

#### タスクの結果

選択した SCA モジュールのインポート用に Web サービス・インポート・バインディングが変更されました。

変更内容は、マスター構成を更新し SCA モジュールを再始動した後で有効になります。

#### 関連概念

223 ページの『第 9 章 バインディングの操作』

サービス指向アーキテクチャの中核をなすのは「サービス」の概念で、これはコンピューター・デバイス間での対話によって実行される機能単位のことです。「エクスポート」はモジュールの外部インターフェース (またはアクセス・ポイント) を定義し、これによってモジュール内の SCA コンポーネントは外部クライアントにサービスを提供できます。「インポート」はモジュール外部のサービスへのインターフェースを定義し、これによってサービスはモジュール内部から呼び出すことができます。インポートおよびエクスポートとともにプロトコル固有の「バインディング」を使用して、データをモジュールの内部または外部に移送する方法を指定します。

## エクスポートの使用

WebSphere Process Server にデプロイされたサービス・モジュールのエクスポートをリストできます。エクスポート・インターフェースおよびエクスポート・バインディングも表示できます。

### このタスクについて

デプロイ済みサービス・モジュールのエクスポートをリストするには、管理コンソールを使用して以下のステップを実行します。

#### 手順

1. ナビゲーション・ペインで、「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. 参照したい特定の SCA モジュールを選択します。
3. コンテンツ・ペインの モジュール・コンポーネントの下で「エクスポート」を展開し、SCA モジュール・エクスポートのリストを表示します。

### タスクの結果

エクスポートのリストがコンテンツ・ペインに表示されます。エクスポートがない場合は情報メッセージが表示されます。

#### 関連概念



#### エクスポートとエクスポート・バインディング

エクスポートは、Service Component Architecture (SCA) モジュールとサービス要求元間の対話を定義します。SCA モジュールはエクスポートを使用して、他のモジュールにサービスを提供します。エクスポート・バインディングは、SCA モジュールがサービス要求元によってアクセスされる際の方法を定義します。



#### HTTP バインディング

HTTP バインディングは、Service Component Architecture (SCA) と HTTP の接続を提供する目的で設計されています。これにより、既存のまたは新規作成された HTTP アプリケーションをサービス指向アーキテクチャー (SOA) 環境内に組み込むことができます。

### エクスポート・インターフェースの表示:

WebSphere Process Server にデプロイされたサービス・モジュールのエクスポート・インターフェースを表示できます。

### このタスクについて

デプロイしたサービス・モジュールのエクスポート・インターフェースを表示するには、管理コンソールを使用して以下のステップを実行します。

#### 手順

1. ナビゲーション・ペインで「アプリケーション」 → 「SCA モジュール」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュール を選択します。

3. コンテンツ・ペインのモジュール・コンポーネントで「**エクスポート**」を展開し、SCA モジュール・エクスポートのリストを表示します。
4. コンテンツ・ペインのモジュール・コンポーネントで、必要なエクスポートを展開し、エクスポートの詳細を表示します。
5. 「**インターフェース**」を展開して、エクスポート・インターフェースを表示します。
6. インターフェースを選択します。

### タスクの結果

コンテンツ・ペインに WSDL (Web サービス記述言語) インターフェースが表示されます。

### エクスポート・バインディングの表示:

サービス・モジュールを WebSphere Process Server にデプロイした後、一部のタイプのエクスポート・バインディングを表示できます。

### このタスクについて

サービス・モジュールのエクスポート・バインディングを表示するには、管理コンソールを使用して以下のステップを実行します。

### 手順

1. ナビゲーション・ペインで「**アプリケーション**」 → 「**SCA モジュール**」を展開し、SCA モジュールのリストを表示します。
2. SCA モジュール を選択します。
3. コンテンツ・ペインで、モジュール・コンポーネントの「**エクスポート**」を展開し、SCA モジュール・エクスポートのリストを表示します。
4. コンテンツ・ペインのモジュール・コンポーネントで、必要なエクスポートを展開し、エクスポートの詳細を表示します。
5. 「**バインディング**」を展開して、エクスポート・バインディングを表示します。
6. バインディングを選択します。

## エンタープライズ・アプリケーションの管理

コンソールの「エンタープライズ・アプリケーション」ページ (「**アプリケーション**」 > 「**エンタープライズ・アプリケーション**」とクリックして表示) を使用して、サーバーにインストールされているエンタープライズ・アプリケーションの表示および管理を行います。

アプリケーションの構成で指定した値を表示するには、リストからアプリケーション名をクリックします。「**アプリケーション詳細**」ページが開き、アプリケーションの構成プロパティ、および該当する場合はローカル・トポロジーが表示されます。このページから既存値を変更して、アプリケーションを構成する追加コンソール・ページにリンクできます。

エンタープライズ・アプリケーションを管理するには、その名前の横にあるチェック・ボックスをクリックしてから、以下のいずれかのボタンを使用します。

表 6. エンタープライズ・アプリケーションを管理するためのボタン

ボタン	結果のアクション
開始	<p>アプリケーションの実行を試みます。アプリケーションが正常に開始すると、アプリケーションの状態は以下のいずれかに変わります。</p> <ul style="list-style-type: none"> <li>• 開始 - アプリケーションは、すべてのデプロイメント・ターゲットで開始されました。</li> <li>• 一部開始 - アプリケーションは、1 つ以上のデプロイメント・ターゲットでまだ開始中です。</li> </ul>
停止	<p>アプリケーションの処理の停止を試みます。アプリケーションが正常に停止すると、アプリケーションの状態は以下のいずれかに変わります。</p> <ul style="list-style-type: none"> <li>• 停止 - アプリケーションは、すべてのデプロイメント・ターゲットで停止されました。</li> <li>• 一部停止 - アプリケーションは、1 つ以上のデプロイメント・ターゲットでまだ停止中です。</li> </ul>
インストール	<p>ウィザードが開き、エンタープライズ・アプリケーションまたはモジュール (.jar, .war, または .ear ファイルなど) をサーバーにデプロイできるようにします。</p>
アンインストール	<p>WebSphere Application Server の構成リポジトリからアプリケーションを削除し、さらに、構成の保管後にアプリケーション・モジュールのインストール先のすべてのノードのファイル・システムから、そのアプリケーション・バイナリーを削除します。</p>
更新	<p>ウィザードが開き、サーバー上にデプロイされているアプリケーション・ファイルを更新できるようにします。アプリケーション全体を更新することも、単一モジュール、単一ファイル、あるいはアプリケーションの一部のみを更新することもできます。新規ファイルまたはモジュールの名前が、サーバー上に既に存在するファイルまたはモジュールの名前と同じである場合は、既存のファイルまたはモジュールが、その新規ファイルまたはモジュールに置き換えられます。同じ名前のファイルまたはモジュールがない場合は、デプロイされているアプリケーションに追加されません。</p>
ファイルの除去	<p>デプロイされたアプリケーションまたはモジュールから、ファイルを削除します。このボタンは、構成リポジトリ、およびファイルがインストールされているすべてのノードのファイル・システムから、ファイルを削除します。</p>
エクスポート	<p>「アプリケーション EAR ファイルのエクスポート」ページが開きます。このページでエンタープライズ・アプリケーションを EAR ファイルにエクスポートできます。デプロイされているアプリケーションをバックアップしたり、そのバイnding情報を保存したりする場合にも、「エクスポート」アクションを使用します。</p>

表 6. エンタープライズ・アプリケーションを管理するためのボタン (続き)

ボタン	結果のアクション
DDL のエクスポート	「アプリケーション DDL ファイルのエクスポート」ページが開きます。このページでエンタープライズ・アプリケーションの EJB モジュール内の DDL ファイルをエクスポートできます。

アプリケーションの管理について詳しくは、WebSphere Application Server インフォメーション・センターを参照してください。

## SCA 要求のスループットの管理

WebSphere Process Server にデプロイされた各 SCA モジュールでは、処理中の要求はキュー・ポイントおよびメッセージング・エンジンのデータ・ストアに保持されます。SCA 要求のデータを表示したり、適宜、何らかの措置を取って SCA 要求のスループットを管理したりすることができます。

### このタスクについて

SCA モジュールをエンタープライズ・サービス・バスで実行している場合は、通常、そのエンタープライズ・サービス・バスを介して要求が送信されるため、管理の必要はありません。必要に応じて、要求のスループットの確認、要求内容の確認、または何らかの問題が発生した場合は要求の削除を行うことができます。要求のスループット全体のモニター、要求の信頼性設定の変更などのアクションを実行しなければならない場合もあります。

要求は、基盤となる WebSphere Application Server のサービス統合テクノロジーにより、メッセージとして処理されます。そのため、要求管理アクションは、サービス統合メッセージを処理する WebSphere Application Server タスクを使用して管理されます。

このトピックでは、必要に応じて実行するメインタスクの概要を説明し、詳細を参照するための WebSphere Application Server タスクへのリンクが記載されています。

- メッセージ・ポイント上のメッセージのリスト表示

処理中の SCA 要求は SCA.SYSTEM.bus のキュー・ポイントに保持されます。SCA 要求は、SCA モジュールのコンポーネントのキュー宛先、またはキュー・ポイントをホストするメッセージング・エンジンを使用してリストできます。例:  
「サービス統合 (Service integration)」 → 「バス」 →  
「SCA.SYSTEM.localhostNode01Cell.Bus」 → 「宛先」 →  
「StockQuoteService\_Export」 → 「キュー・ポイント (Queue points)」 →  
「StockQuoteService\_Export@localhostNode01.server1-SCA.SYSTEM.localhostNode01Cell.Bus」 → 「ランタイム」 → 「メッセージ」

- メッセージ・ポイント上のロックされたメッセージの解決

問題が発生した場合は、SCA 要求が処理中のキュー・ポイントにロックされたままになる場合があります。メッセージの **State** プロパティを表示して、要求がロックされているかどうかを確認できます。必要があればアクションを実行して問題を解決します。

- メッセージ・ポイントのメッセージの削除

特別な事情により、選択したバス宛先またはメッセージング・エンジンのメッセージ・ポイントにある 1 つ以上のメッセージを削除する必要がある場合があります。通常は、メッセージ・ポイントにあるメッセージを削除する必要はありません。このタスクはトラブルシューティング手順の一部として実行するものです。

- メッセージング・エンジンのデータ・ストア内のデータの表示

メッセージング・エンジンはメッセージ、トランザクション状態、通信チャネル状態など、非永続データと永続データの両方をデータ・ストア内で保守します。データベース・ツールを使用して、メッセージング・エンジンに関するデータ・ストア内のデータを表示できます。

- 宛先のメッセージ信頼性の変更

メッセージには、メッセージ配信の信頼性を指定するサービス品質属性が備わっています。配信保証の要件およびシステム性能に応じて、信頼性を選択できます。

## データ・ストア内のデータの表示

メッセージング・エンジンはメッセージ、トランザクション状態、通信チャネル状態など、非永続データと永続データの両方をデータ・ストア内で保守します。データベース・ツールを使用して、メッセージング・エンジンに関するデータ・ストア内のデータを表示できます。

### 始める前に

ij ツールを使用して、メッセージング・エンジンの組み込み Derby データ・ストア内のデータを表示するには、事前にメッセージング・エンジンを停止しておく必要があります。

### このタスクについて

メッセージング・エンジンが停止すると、非永続データは失われます。データは、制御されながら失われる場合もあれば、制御されずに失われる場合もあります。永続データは、サーバーの再始動後に使用可能になります。

場合によっては、メッセージング・エンジンにより処理中のメッセージを検査するために、データ・ストア内のデータを表示する必要が生じることがあります。

データ・ストア用のデータベース・ツールを使用して、メッセージング・エンジンに関するデータ・ストア内のデータを表示できます。例えば、メッセージング・エンジンが組み込み Derby データベースを使用している場合は、ij ツールを使用して要求メッセージを表示できます。

### 手順

1. ij ツールを開始します。Windows® では、以下のサブステップを実行します。

- a. コマンド・ウィンドウを開きます。
- b. ディレクトリーを `profile_root\derby\bin\embedded` に変更します。
- c. `ij.bat` と入力します。

Windows 以外のプラットフォームでは、以下のサブステップを実行します。

- a. コマンド・ウィンドウを開きます。
  - b. ディレクトリーを、`profile_root/derby/bin/embedded` に変更します。
  - c. `./ij.sh` と入力します。
2. メッセージング・エンジンのデータ・ストアを開きます。 `ij` ツールを使用して、以下のサブステップを実行します。
    - a. 必要なデータベース・ファイルに接続します。

メッセージング・エンジンの場合、データベースはディレクトリー `profile_root/profiles/profile_name/databases/com.ibm.ws.sib` にメッセージング・エンジンの名前で格納されます。例えば、Windows でのデフォルトのスタンドアロン・プロファイルの場合、メッセージング・エンジン `localhostNode01.server1-SCA.SYSTEM.localhostNode01Cell.Bus` (SCA.SYSTEM バス上のサーバー 1) のデータベース・ファイルのディレクトリーは次のとおりです。

`profile_root\profiles\default\databases\com.ibm.ws.sib\localhostNode01.server1-SCA.SYSTEM.localhostNode01Cell.Bus`

- b. `ij` ツールを使用して、SQL コマンドを発行し、データを表示します。
  - 1) ディレクトリーを、`install_root/derby/bin/embedded` に変更します。
  - 2) `./ij.sh` と入力します。
  - 3) `protocol 'jdbc:derby:'` ; と入力します。
  - 4) `connect 'profile_root/profiles/profile_name/databases/com.ibm.ws.sib/database_name'` ; と入力します。
- c. オプション: `ij` の使用方法に関するヘルプを表示するには、`ij>` プロンプトで `help` ; と入力します。

## バス宛先のメッセージ信頼性の変更

メッセージには、メッセージ配信の信頼性を指定するサービス品質属性が備わっています。配信保証の要件およびシステム性能に応じて、信頼性を選択できます。

### このタスクについて

管理者がバス宛先上で信頼性設定を指定するか、個々の作成者により (通常、API 呼び出しによるアプリケーション制御のもとで) 信頼性を指定できます。管理者は、宛先に設定されたデフォルトの信頼性を作成者がオーバーライド可能かどうか、および接続している作成者が要求できる信頼性の上限を指定できます。

宛先のメッセージの信頼性設定を参照または変更するには管理コンソールを使用して、以下のステップを完了します。

#### 手順

1. ナビゲーション・ペインの「サービス統合」 → 「バス」 をクリックします。
2. コンテンツ・ペインで、宛先が存在するバスの名前をクリックします。
3. 「宛先」 をクリックします。

- 宛先名をクリックします。宛先の詳細ページが表示されます。
- 信頼性プロパティを調べます。次のプロパティによって、宛先のメッセージの信頼性が制御されます。

**Default reliability**

製作者が信頼性を明示的に設定していないときに、この宛先に生成されるメッセージに割り当てる信頼性。

**Maximum reliability**

この宛先が受け入れるメッセージの最大の信頼性。

これらのプロパティには、以下にリストされた値を指定できます。

**Best effort nonpersistent**

メッセージング・エンジンが停止または障害が発生すると、メッセージが破棄されます。また、メッセージ送信用の接続が使用不可になった場合、およびシステム・リソースに余裕のない場合も、メッセージが破棄されることがあります。

**Express nonpersistent**

メッセージング・エンジンが停止または障害が発生すると、メッセージが破棄されます。また、メッセージ送信用の接続が使用不可になった場合も、メッセージが破棄されることがあります。

**Reliable nonpersistent**

メッセージング・エンジンが停止または障害が発生すると、メッセージが破棄されます。

**Reliable persistent**

メッセージング・エンジンに障害が発生すると、メッセージが破棄される場合があります。

**Assured persistent**

メッセージは破棄されません。

メッセージの信頼性を制御するこれらのプロパティの使用については、『メッセージ信頼性レベル』を参照してください。

- デフォルトの信頼性設定を作成者がオーバーライドできるようにするかどうかを検討します。

**プロデューサーによるデフォルトの信頼性のオーバーライドを可能にする**

プロデューサーが、宛先に設定されているデフォルトの信頼性をオーバーライドできるようにするには、このオプションを選択します。

- オプション: **オプション:** 必要に応じて、宛先プロパティを変更します。

『バス宛先の構成』の説明に従って、必要に応じてその他のプロパティを設定することにより、宛先をより詳細に構成できます。

- 「OK」をクリックします。
- 変更内容をマスター構成に保管します。

## サービス・アプリケーションおよびサービス・モジュールの追加処理

WebSphere 管理コンソールを使用すると、サービス・モジュール本体だけでなく、モジュールが使用するリソースや、モジュールが含まれるアプリケーションまで管理できます。このタスクはコマンドでも実行できます。

### このタスクについて

サービス・モジュールを管理するためのルーチン・タスクは、93 ページの『サービス・モジュールの管理』で説明しています。高度な作業については、以下のサブトピックを参照してください。

### サービス・モジュールのリソースの管理

サービス・モジュールは、WebSphere Application Server のサービス統合テクノロジーが提供するリソースを利用します。またサービス・モジュールは、Java Message Service (JMS) および Common Event Infrastructure によって提供されるリソースを含む、さまざまなリソースも利用します。サービス・モジュールのリソースは、WebSphere 管理コンソール、コマンド、スクリプト・ツールを使用して管理することができます。

サービス・モジュールのリソースの管理について詳しくは、関連するトピックを参照してください。

### サービス統合テクノロジー

バス宛先などのサービス統合リソースにより、サービス・モジュールでサービス統合テクノロジーを使用することができます。サービス・モジュールによって使用される SCA ランタイムは、コンポーネントとモジュール間の非同期対話をサポートする堅固なインフラストラクチャーとして、キュー宛先を使用します。サービス・モジュールを WebSphere Process Server にインストールすると、モジュールによって使用される宛先が、SCA.SYSTEM.bus のメンバー上に定義されます。これらのバス宛先は、非同期対話を使用するサービス・モジュールのコンポーネントに対して処理されるメッセージの保管用に使用されます。

#### Queue *sca/module\_name*

モジュール *module\_name* に送信される非同期要求をバッファーに入れるために使用する宛先です。

#### Queue *sca/module\_name/exportlink/export\_name*

これは、非同期要求をモジュールに送信するためにエクスポートが使用する宛先です。要求は、そのエクスポートにリンクされたコンポーネント・ターゲットに送付されます。

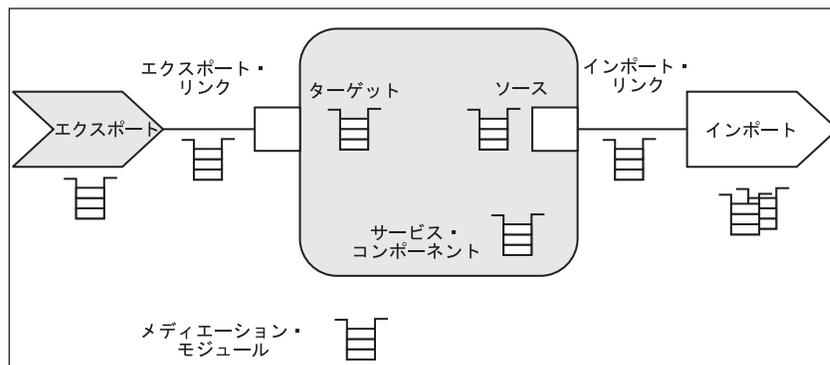
#### Queue *sca/module\_name/importlink/import\_name*

モジュールからの非同期要求を送信するインポートによって使用される宛先です。要求は、そのインポートにリンクされたモジュール・エクスポートに送付されます。

#### Queue *sca/module\_name/import/sca/dynamic/import/scaimport* [for SCA binding]

#### Queue *sca/module\_name/import/sca/dynamic/import/wsimport* [for Web service binding]

Queue sca/contextStore/module\_name



宛先ごとに、キュー・ポイントも作成され、該当するバス・メンバーのメッセージング・エンジンで定義されます。

サービス・モジュールをデプロイして使用する際に、これらのリソースを管理する必要はありません。ただし、リソースの構成を調整したり（例えば、使用されるサービスの最大メッセージング品質を変更する場合）、トラブルシューティングでメッセージを見つける際にリソースを使用することもできます。

## Java Message Service (JMS)

JMS リソースにより、Java Message Service (JMS) プログラミング・インターフェースに基づく通信方式として、サービス・モジュールで非同期メッセージングを使用することができます。使用される JMS サポートは、モジュールの JMS バインディングによって異なります。例えば、JMS バインディングを含むモジュールは、基盤となる WebSphere Application Server により提供されるデフォルトのメッセージング・プロバイダー上で構成されている JMS 接続ファクトリーを使用し、WebSphere MQ JMS バインディングを含むモジュールは、WebSphere MQ 上で JMS プロバイダーとして構成されている JMS 接続ファクトリーを使用します。Java Message Service の使用状態を管理するため、次のリソースを管理できます。

### JMS 接続ファクトリー

JMS 接続ファクトリーは、Point-to-Point メッセージングとパブリッシュ/サブスクライブ・メッセージングの両方において、対応する JMS 宛先の JMS プロバイダーとの接続を作成するために使用します。接続ファクトリー管理オブジェクトを使用して、プロバイダーの JMS 接続ファクトリーを管理します。

### JMS キュー (JMS queue)

JMS キューは、Point-to-Point メッセージングの宛先として使用されます。JMS キュー宛先管理オブジェクトを使用して、プロバイダーの JMS キューを管理します。

### JMS トピック

JMS トピックは、パブリッシュ/サブスクライブ・メッセージングの宛先として使用されます。トピック宛先管理オブジェクトを使用して、プロバイダーの JMS トピックを管理します。

## JMS アクティベーション・スペック

JMS アクティベーション・スペックは、1 つ以上のメッセージ駆動型 Bean に関連付けられており、これらの Bean がメッセージを受信するのに必要な構成を提供します。

## JMS リスナー・ポート

JMS リスナー・ポートは、接続ファクトリー、宛先、およびメッセージ駆動型 Bean の間の関連を定義します。これにより、ポートに関連付けられているデプロイ済みのメッセージ駆動型 Bean が、宛先からメッセージを取得できます。

## Common Event Infrastructure (CEI)

CEI リソースにより、イベント・データを管理するための標準の形式と機構をサービス・モジュールで使用することができます。Common Event Infrastructure の使用を管理するため、次のリソースを管理できます。

### データ・ストア・プロファイル

デフォルトのデータ・ストアが使用するプロパティを定義します。デフォルトのデータ・ストアは、Common Event Infrastructure によって提供されるデータ・ストアです。

### エミッター・ファクトリー・プロファイル

このプロファイルは、イベント・エミッターのオプションを定義します。

### イベント・バス伝送プロファイル

このプロファイルは、イベント・バスへの EJB エントリを定義します。

### イベント・グループ・プロファイル

このプロファイルは、セレクター式で判別されるイベントのリストを定義します。JMS キューおよび JMS トピックを各イベント・グループに関連付けることができます。イベント・サーバー配布サービスが使用可能で、イベントがイベント・グループに一致する場合、イベントはそのイベント・グループ用に構成されたトピックまたはキューに配布されます。

### イベント・サーバー・プロファイル

このプロファイルは、イベント・サーバーのプロパティを定義します。

### フィルター・ファクトリー・プロファイル

このプロファイルは、フィルターのプロパティを定義します。フィルターはフィルター構成ストリングを使用して、イベントがバスに渡されるかどうかを判別します。

## JMS 伝送プロファイル

プロファイルは、イベント・バス内に JMS キュー項目を定義します。これは、JMS キューおよびキュー接続ファクトリーの JNDI 名を定義します。

### 関連タスク



#### 実動サーバーへのモジュールのインストール

このトピックでは、テスト・サーバーからアプリケーションを取り出して実稼働環境にデプロイするステップについて説明します。



#### Common Event Infrastructure の管理

## 関連資料



### serviceDeploy コマンド行ユーティリティー

serviceDeploy コマンドを使用して、Service Component Architecture (SCA) 対応モジュールを、サーバーにインストール可能な Java アプリケーションとしてパッケージします。このコマンドは、wsadmin を使用してバッチ・インストールを実行する場合に役立ちます。



### プロモート可能なプロパティー

## アプリケーションにおけるサービス統合の管理

このトピックでは、サービス統合テクノロジーについて説明します。サービス統合は、セル内のアプリケーション・サーバーで実行しているメッセージング・エンジンのグループ (通常は 1 台のサーバーに対して 1 つのエンジン) として実装されます。

サービス統合バスは、同期メッセージングと非同期メッセージングによってサービス統合をサポートする管理対象通信の一形態です。バスは、バス・リソースを管理する相互接続メッセージング・エンジンで構成されます。サービス統合バスのメンバースは、メッセージング・エンジンが定義されているアプリケーション・サーバーとアプリケーション・クラスターです。

### Service Integration Bus Browser:

Service Integration Bus Browser は、サービス統合バスにおける日常の操作タスクの参照および実行を行うための単一ロケーションを提供します。

日常の操作の例として、サービス統合バスの参照、メッセージング・エンジンのランタイム・プロパティーの表示、またはメッセージ・ポイント上のメッセージの管理などがあります。ブラウザーはバス構成ツール用のものではありません。

「サービス統合 (Service Integration)」 → 「Service Integration Bus Browser」をクリックして Service Integration Bus Browser にアクセスすると、標準のコンソール・ナビゲーション・ペインの右側に以下の 2 つのペインが開きます。

#### ナビゲーション・ツリー・ペイン

このペインには、システム上に構成されたサービス統合バスを参照できるナビゲーション・ツリーが表示されます。

#### コンテンツ・ペイン

このペインには、バスおよびその個別コンポーネント、例えばメッセージング・エンジン、キュー・ポイント、宛先、公開ポイント、およびメディエーション・ポイントなどについてのコレクション・ページおよび詳細ページが表示されます。

ナビゲーション・ツリーのリンクからアクセスしたすべてのページが編集可能というわけではありません。編集可能な複数バージョンのページにアクセスする方法などの詳細については、ブラウザーのオンライン・ヘルプを参照してください。

ナビゲーション・ツリー・ペインで項目をクリックすると、対応するコレクション・ページまたは詳細ページがコンテンツ・ペインに表示されます。

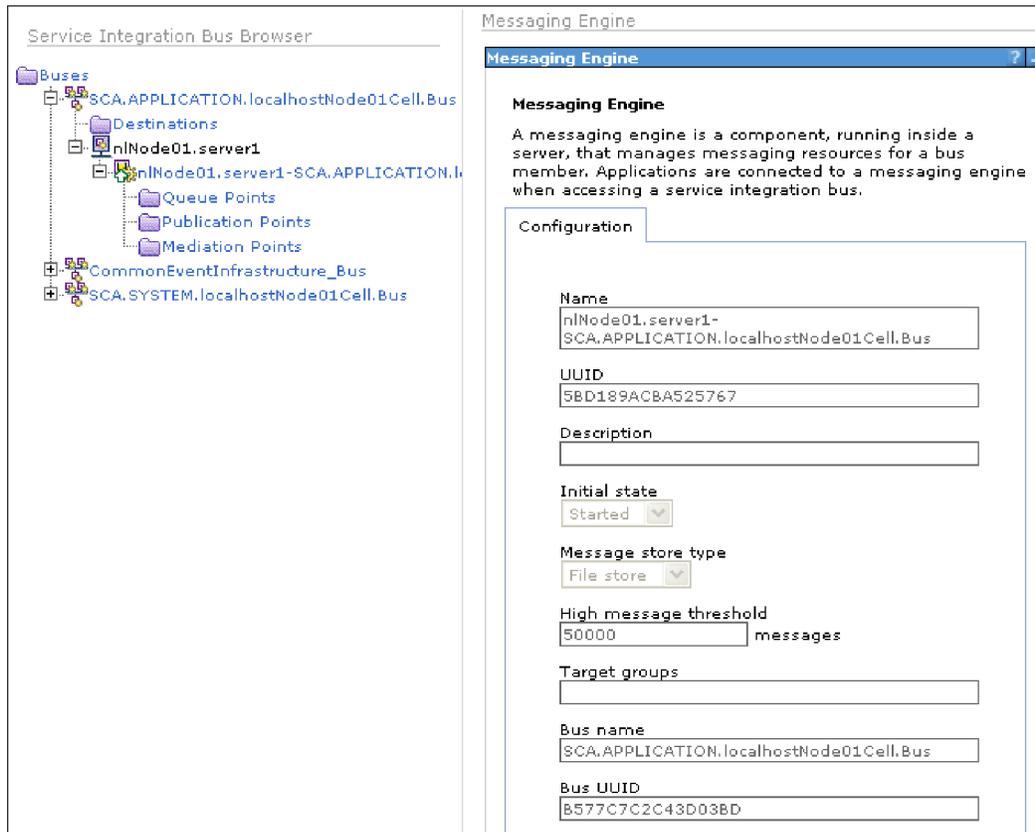


表7では、各項目に関連付けられたナビゲーション・ツリー内のアイコンについてリストし説明します。

表7. Service Integration Bus Browser のアイコン

アイコン	説明
	ナビゲーション・ツリー内で表示される場所によって、バス、宛先、キュー・ポイント、公開ポイント、またはメディエーション・ポイントのコレクションを表します。
	サービス統合バスを表します。
	メッセージング・エンジンを表します。
	サービス統合バス・メンバーを表します。

### 関連タスク

 サービス統合バスについて

## コマンドを使用したサービス・アプリケーション管理

コマンドを使用してサービス・アプリケーションを管理できます。コマンドはスクリプト内で使用できます。

### 始める前に

`wsadmin` ツールを使用して、サービス・アプリケーション・コマンドを実行します。

### このタスクについて

`wsadmin` ツールはさまざまな方法で使用できます。個々のコマンドとして、またはスクリプトで、このツールを対話式に使用できます。複数のマシンを管理している場合は、スクリプト内で複数のコマンドを実行すると便利です。

WebSphere Process Server には、SCA モジュールとそのインポートおよびエクスポートを表示し、インポート・バインディングとエクスポート・バインディングの詳細を変更するコマンドがあります。

注: `wsadmin` スクリプトの `Jacl` 構文は使用すべきではありません。今後のリリースでサポートが廃止される予定です。 `Jython` が `wsadmin` スクリプトの戦略的構文です。

### 手順

1. SCA 管理コマンドをリストします。 `$AdminTask help SCAAdminCommands`
2. 特定のコマンドの詳細ヘルプを表示します。 `$AdminTask help command_name`

### 例

```
$AdminTask help listSCAModules
```

### コマンドを使用したサービス・モジュールの管理:

コマンド行により WebSphere Process Server にデプロイされたサービス・モジュールをリストできます。各サービス・モジュールに関連する情報を表示し、一部のインポート・バインディングを変更することもできます。

### 始める前に

`wsadmin` ツールを使用して WebSphere Process Server コマンドを実行します。

### このタスクについて

コマンドは個別にも実行できますし、スクリプトでも実行できます。複数のホストを管理する場合や、定期的にレポートを作成する場合には、スクリプトで複数のコマンドを実行すると便利です。

### コマンドを使用したサービス・モジュールのリスト表示:

コマンドを使用して、WebSphere Process Server にデプロイされたサービス・モジュールをリストできます。

## 始める前に

wsadmin ツールを開始します。このツールを使用して、必要なコマンドを実行します。

## このタスクについて

デプロイされているサービス・モジュールをすべてリストするには、以下の手順を実行します。

## 手順

デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`

## タスクの結果

WebSphere Process Server にデプロイされている SCA モジュールおよびそれに関連したアプリケーションをリストします。出力は、`module name:application name` のフォーマットで戻されます。このようにリストすることにより、以降のコマンドでスクリプトを使用する際に、出力の解析処理と名前の抽出処理が容易になります。

## コマンドを使用したサービス・モジュールの詳細の表示:

コマンドを使用して、サービス・モジュールの属性を表示できます。

## 始める前に

wsadmin ツールを開始します。このツールを使用して、必要なコマンドを実行します。

## このタスクについて

特定のサービス・モジュールの説明を表示するには、モジュール名を知る必要があります。

`listSCAModules` コマンドにより、WebSphere Process Server にデプロイされている SCA モジュールがリストされます。

## 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールの詳細を表示します。 `$AdminTask showSCAModule {-moduleName moduleName}`

**注:** モジュール名を指定するほか、アプリケーション名を指定することもできます。アプリケーション名を指定することにより、パフォーマンスが向上します。

## タスクの結果

指定した SCA モジュールの名前および説明が表示されます。

## 例

```
$AdminTask showSCAModule {-moduleName myModule -applicationName  
myApplication}
```

### コマンドを使用したサービス・モジュールのプロパティの表示:

コマンドを使用して、指定したサービス・モジュールのプロパティを表示できます。

### 始める前に

wsadmin ツールを開始します。このツールを使用して、必要なコマンドを実行します。

### このタスクについて

特定のサービス・モジュールのプロパティを表示するには、モジュール名を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされている SCA モジュールがリストされます。

### 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのプロパティを表示します。 `$AdminTask showSCAModuleProperties {-moduleName moduleName}`

**注:** モジュール名を指定するほか、アプリケーション名を指定することもできます。アプリケーション名を指定することにより、パフォーマンスが向上します。

### タスクの結果

指定した SCA モジュールのプロパティが表示されます。

## 例

```
$AdminTask showSCAModuleProperties {-moduleName myModule -applicationName  
myApplication}
```

### コマンドを使用したサービス・モジュール・プロパティの変更:

コマンドを使用して、指定したサービス・モジュールのプロパティ値を変更できます。

### 始める前に

wsadmin ツールを開始します。このツールを使用して、必要なコマンドを実行します。

## このタスクについて

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストできます。

### 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのプロパティをリストします。 `$AdminTask showSCAModuleProperties {-moduleName moduleName}`

注: モジュール名を指定するほか、アプリケーション名を指定することもできます。アプリケーション名を指定することにより、パフォーマンスが向上します。

3. SCA モジュールのモジュール・プロパティを変更します。 `$AdminTask modifySCAModuleProperty -moduleName moduleName -propertyName propertyName -newPropertyValue newPropertyValue`

注: オプションで、アプリケーション名も指定できます。アプリケーション名を指定することにより、パフォーマンスが向上します。

### タスクの結果

指定した SCA モジュール・プロパティのプロパティ値が変更されます。

### 例

```
$AdminTask modifySCAModuleProperty {-moduleName myModule -applicationName myApplication -propertyName myPropertyName -newPropertyValue myNewPropertyValue}
```

### コマンドを使用したインポートのリスト表示:

コマンドを使用して、WebSphere Process Server にデプロイされている任意のサービス・モジュールのインポートをリストさせることができます。

### 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

## このタスクについて

wsadmin ツールを使用してサービス・モジュールのインポートをリストするには、以下のステップを実行します。

サービス・モジュールのインポートをリストするには、モジュールの名前を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストし、listSCAImports コマンドにより、特定の SCA モジュールのインポートをリストします。SCA モジュールにインポートが存在しない場合もあります。

## 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのインポートをリストします。 `$AdminTask listSCAImports {-moduleName moduleName}`

注: *moduleName* の指定に加えて、オプションで *applicationName* も指定できます。 *applicationName* を指定することにより、パフォーマンスが向上します。

## タスクの結果

特定の SCA モジュールのインポートをリストします。

## 例

```
$AdminTask listSCAImports {-moduleName myModule -applicationName myApplication}
```

### コマンドを使用したインポートの詳細の表示:

コマンドを使用して、WebSphere Process Server にデプロイされているサービス・モジュールのインポートの詳細を表示させることができます。

## 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

### このタスクについて

wsadmin ツールを使用してサービス・モジュールのインポートの詳細を表示するには、以下のステップに従います。

特定のサービス・モジュール・インポートの詳細を表示するには、モジュール名およびインポート名を知る必要があります。

`listSCAModules` コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストし、`listSCAImports` コマンドにより、特定の SCA モジュールのインポートをリストします。 SCA モジュールにインポートが存在しない場合もあります。

## 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのインポートをリストします。 `$AdminTask listSCAImports {-moduleName moduleName}`

注: *moduleName* の指定に加えて、オプションで *applicationName* も指定できます。 *applicationName* を指定することにより、パフォーマンスが向上します。

3. 特定の SCA モジュール・インポートの詳細を表示します。 `$AdminTask showSCAImport {-moduleName moduleName -import importName}`

*moduleName* および *importName* の指定に加えて、オプションで *applicationName* も指定できます。

注: *applicationName* を指定することにより、パフォーマンスが向上します。

## タスクの結果

特定の SCA モジュール・インポートのインポート詳細を表示します。

### 例

```
$AdminTask showSCAImport {-moduleName myModule -applicationName  
myApplication -import myImport}
```

### コマンドを使用したインポート・バインディングの表示:

コマンドを使用して、WebSphere Process Server にデプロイされているサービス・モジュールのインポート・バインディングを表示させることができます。

### 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

### このタスクについて

wsadmin ツールを使用して特定のサービス・モジュールのインポート・バインディングを表示するには、以下のステップに従います。

特定のサービス・モジュール・インポートのインポート・バインディングを表示するには、モジュール名およびインポート名を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストし、listSCAImports コマンドにより、特定の SCA モジュールのインポートをリストします。SCA モジュールにインポートが存在しない場合もあります。

### 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのインポートをリストします。 `$AdminTask listSCAImports {-moduleName moduleName}`

*moduleName* の指定に加えて、オプションで *applicationName* も指定できます。

注: *applicationName* を指定することにより、パフォーマンスが向上します。

3. 特定のインポートのインポート・バインディングを表示します。 `$AdminTask showSCAImportBinding {-moduleName moduleName -import importName}`

*moduleName* および *importName* の指定に加えて、オプションで *applicationName* も指定できます。

注: *applicationName* を指定することにより、パフォーマンスが向上します。

## タスクの結果

特定の SCA モジュール・インポートのインポート・バインディングを表示します。

### 例

```
$AdminTask showSCAImportBinding {-moduleName myModule -applicationName  
myApplication -import myImport}
```

### コマンドを使用した SCA インポート・バインディングの変更:

コマンドを使用して、WebSphere Process Server にデプロイされているサービス・モジュールの SCA インポート・バインディングを変更することができます。

### 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

### このタスクについて

SCA バインディングによって、ある Service Component Architecture (SCA) モジュールが別の SCA モジュールに接続されます。

特定のサービス・モジュールで異なるサービス・モジュールを呼び出す場合は、SCA インポート・バインディングを変更する場合があります。インポート・バインディングを変更する場合は、インポートとエクスポートが一致するように、つまり、同等の操作を行うようにしなければなりません。それには、WSDL の検討が必要となる場合があります。

特定のサービス・モジュール・インポートのバインディングを変更するには、ソースおよびターゲット・サービス・モジュール、および特定のインポートとエクスポートの名前を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストできます。listSCAImports コマンドにより、特定の SCA モジュールのすべてのインポートが、listSCAExports コマンドにより、特定の SCA モジュールのすべてのエクスポートがリストされます。

### 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのインポートをリストします。 `$AdminTask listSCAImports {-moduleName moduleName}`

注: `moduleName` の指定に加えて、オプションで `applicationName` も指定できます。 `applicationName` を指定することにより、パフォーマンスが向上します。

3. SCA タイプの SCA モジュール・インポート・バインディングを変更します。  
`$AdminTask modifySCAImportSCABinding -moduleName moduleName -import  
importName -targetModule targetModuleName -targetExport targetExportName`

オプションで、`applicationName` および `targetApplicationName` も指定できます。

注: *applicationName* および *targetApplicationName* を指定することにより、パフォーマンスが向上します。

## タスクの結果

特定の SCA モジュール・インポートに対する SCA インポート・バインディングが変更されました。

WebSphere ESB では、エクスポート・インターフェースに対応しない各インポート・インターフェースに対して警告が出されます。WebSphere Process Server では、インポートとエクスポートの WSDL ポート・タイプ名が比較され、同じでない場合は警告が出されます。ただし、ポート・タイプ名が一致する場合、WebSphere Process Server では、行われた操作を等価とみなし警告は出されません。

## 例

```
$AdminTask modifySCAImportSCABinding {-moduleName myModule -applicationName myApplication -import myImport -targetModule myTargetModule -targetApplicationName myTargetApplication -targetExport myTargetExport}
```

## コマンドを使用したインポート Web サービス・バインディングの変更:

コマンドを使用して、WebSphere Process Server にデプロイされているサービス・モジュールの Web サービス インポート・バインディングを変更することができます。

## 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

## このタスクについて

特定のサービス・モジュールで異なる Web サービスを呼び出す場合、インポート Web サービス・バインディングを変更する場合があります。インポート・バインディングを変更する場合、インポートとエクスポートを一致させる (つまり提供されている操作が同等である) 必要があります。それには、WSDL の検討が必要となる場合があります。wsadmin ツールを使用して、以下のステップを実行します。

特定のサービス・モジュール・インポートの Web サービス・バインディングを変更するには、ターゲット・エンドポイントの特定の URL を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストできます。listSCAImports コマンドにより、特定の SCA モジュールのすべてのインポートが、listSCAExports コマンドにより、特定の SCA モジュールのすべてのエクスポートがリストされます。

## 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのインポートをリストします。 `$AdminTask listSCAImports {-moduleName moduleName}`

*moduleName* の指定に加えて、オプションで *applicationName* も指定できます。

注: *applicationName* を指定することにより、パフォーマンスが向上します。

3. WS タイプのインポート・バインディングを変更します。 `$AdminTask modifySCAImportWSBinding -moduleName moduleName -import importName -endpoint targetEndpointName`

### タスクの結果

特定の Web サービス・インポートのインポート Web サービス・バインディングが変更されました。

WebSphere ESB では、エクスポート・インターフェースに対応しない各インポート・インターフェースに対して警告が出されます。WebSphere Process Server では、インポートとエクスポートの WSDL ポート・タイプ名が比較され、同じでない場合は警告が出されます。ただし、ポート・タイプ名が一致する場合、WebSphere Process Server では、行われた操作を等価とみなし警告は出されません。

### 例

```
$AdminTask modifySCAImportWSBinding {-moduleName myModule -applicationName myApplication -import myImport -endpoint http://myTargetEndpoint}
```

### コマンドを使用したエクスポートのリスト表示:

コマンドを使用して、WebSphere Process Server にデプロイされている任意のサービス・モジュールのエクスポートをリストさせることができます。

### 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

### このタスクについて

wsadmin ツールを使用してサービス・モジュールのエクスポートをリストするには、以下のステップに従います。

特定のサービス・モジュールのエクスポートをリストするには、モジュール名を知る必要があります。

`listSCAModules` コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストし、`listSCAExports` コマンドにより、特定の SCA モジュールのエクスポートをリストします。SCA モジュールにエクスポートが存在しない場合もあります。

### 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのエクスポートをリストします。 `$AdminTask listSCAExports {-moduleName moduleName}`

*moduleName* の指定に加えて、オプションで *applicationName* も指定できます。

注: *applicationName* を指定することにより、パフォーマンスが向上します。

## タスクの結果

特定の SCA モジュールのエクスポートをリストします。

## 例

```
$AdminTask listSCAExports {-moduleName myModule -applicationName myApplication}
```

## コマンドを使用したエクスポートの詳細の表示:

コマンドを使用して、WebSphere Process Server にデプロイされているサービス・モジュールのエクスポートの詳細を表示させることができます。

## 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

## このタスクについて

wsadmin ツールを使用してサービス・モジュールのエクスポートの詳細を表示するには、以下のステップを実行します。

特定のサービス・モジュール・エクスポートの詳細を表示するには、モジュール名およびエクスポート名を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストし、listSCAExports コマンドにより、特定の SCA モジュールのエクスポートをリストします。SCA モジュールにエクスポートが存在しない場合もあります。

## 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのエクスポートをリストします。 `$AdminTask listSCAExports {-moduleName moduleName}`

注: *moduleName* の指定に加えて、オプションで *applicationName* も指定できます。*applicationName* を指定することにより、パフォーマンスが向上します。

3. 特定の SCA モジュール・エクスポートの詳細を表示します。 `showSCAExport -moduleName moduleName -export exportName`

*moduleName* および *exportName* の指定に加えて、オプションで *applicationName* も指定できます。

注: *applicationName* を指定することにより、パフォーマンスが向上します。

## タスクの結果

特定の SCA モジュール・エクスポートのエクスポート詳細を表示します。

## 例

```
$AdminTask showSCAExport {-moduleName myModule -applicationName  
myApplication -export myExport}
```

### コマンドを使用したエクスポート・バインディングの表示:

コマンドを使用して、WebSphere Process Server にデプロイされているサービス・モジュールのエクスポート・バインディングを表示させることができます。

### 始める前に

wsadmin ツールを使用して、以下のコマンドを実行します。

### このタスクについて

サービス・モジュールのエクスポート・バインディングを表示するには、wsadmin ツールを使用して以下のステップを実行します。

特定のサービス・モジュール・エクスポートのエクスポート・バインディングを表示するためには、モジュール名およびエクスポート名を知る必要があります。

listSCAModules コマンドにより、WebSphere Process Server にデプロイされているすべての SCA モジュールをリストし、listSCAExports コマンドにより、特定の SCA モジュールのエクスポートをリストします。

### 手順

1. デプロイされている SCA モジュールをリストします。 `$AdminTask listSCAModules`
2. 特定の SCA モジュールのエクスポートをリストします。 `$AdminTask listSCAExports {-moduleName moduleName}`

*moduleName* の指定に加えて、オプションで *applicationName* も指定できます。

**注:** *applicationName* を指定することにより、パフォーマンスが向上します。

3. 特定のエクスポートのエクスポート・バインディングを表示します。  
`$AdminTask showSCAExportBinding {-moduleName moduleName -export  
exportName}`

**注:** *moduleName* および *export* の指定に加えて、オプションで *applicationName* も指定できます。*applicationName* を指定することにより、パフォーマンスが向上します。

### タスクの結果

特定の SCA モジュール・エクスポートのエクスポート・バインディングを表示します。表示される情報は、バインディングのタイプによって異なります。指定されたバインディングがエクスポートに存在しない場合は、そのバインディングが SCA タイプであるとランタイムによりみなされます。

## 例

```
$AdminTask showSCAExportBinding {-moduleName myModule -applicationName  
myApplication -export myExport}
```

---

## デプロイされた SCA モジュールの WebSphere MQ JMS 宛先の変更

ビジネス環境の発展に応じて、Service Component Architecture (SCA) モジュールが使用する WebSphere MQ 宛先を変更する必要があることがあります。

### 始める前に

変更する WebSphere MQ JMS 宛先を、どの SCA モジュールが使用しているかを知っておく必要があります。デフォルトでは、これらの宛先は、SCA モジュールをサーバーまたはクラスターにインストールしたときに作成されます。宛先の形式は、以下のとおりです。

- 片方向のエクスポートの場合:
  - *modulename.exportname\_MQEXPORT\_CF*
  - *modulename.exportname\_MQ\_RECEIVE\_D*
- これに加えて、両方向のエクスポートの場合:
  - *modulename.exportname\_MQ\_SEND\_D*
- 片方向のインポートの場合:
  - *modulename.importname\_MQIMPORT\_CF*
  - *modulename.importname\_MQ\_SEND\_D*
- これに加えて、両方向のインポートの場合:
  - *modulename.importname\_MQ\_RECEIVE\_D*

このタスクは、管理コンソールを使用して構成を変更することを前提とします。

**制約事項:** 宛先を変更するときには、以下の事項に注意してください。

- ターゲット・クライアントを **MQ** に設定した状態でキュー宛先を構成する必要があります。
- TCP/IP クライアント接続を使用して、WebSphere MQ に接続する必要があります。詳しくは、「WebSphere MQ 相互通信」を参照してください。
- チャンネル圧縮は使用できません。
- データを正しく処理するには、必要なチャンネル出口をすべて設定する必要があります。これを行うには、「WebSphere MQ メッセージング・プロバイダー」 → 「WebSphere MQ 接続ファクトリー・オブジェクト (WebSphere MQ connection factory objects)」 → 「カスタム・プロパティ」をクリックして、以下のカスタム・プロパティを設定します。
  - **SENDEXIT**. このプロパティには、値 `com.ibm.ws.sca.internal.mq.exit.MQInternalSendExitImpl` を設定します。
  - **RECEXIT**. このプロパティには、値 `com.ibm.ws.sca.internal.mq.exit.MQInternalReceiveExitImpl` を設定します。

- **SENDEXITINIT**。このプロパティには、選択したテキスト・ストリングを設定します。このプロパティを設定すると、トレースで出口を見分けやすくなります。
- **RECEXITINIT**。このプロパティには、選択したテキスト・ストリングを設定します。このプロパティを設定すると、トレースで出口を見分けやすくなります。
- 出口では **JMS MQ** バインディングは禁止されるため、**WebSphere MQ** バインディングと **Java Message Service (JMS) MQ** バインディングに同じ接続ファクトリーを使用することはできません。

## このタスクについて

SCA モジュールをデプロイした後で、新規のビジネス要件を満たすために、**WebSphere MQ JMS** 宛先の変更が必要になることがあります。

### 手順

1. 変更する宛先を使用しているすべてのアプリケーションを停止します。『エンタープライズ・アプリケーションの管理』で説明する手順を実行します。
2. 宛先が進行中のメッセージの処理を完了することを許可します。
3. **WebSphere MQ JMS** 宛先を表示します。管理コンソールでこのページにナビゲートするには、「リソース」>「**JMS プロバイダー**」>「**WebSphere MQ**」>「**WebSphere MQ キュー宛先**」を使用します。
4. 変更する宛先のスコープを選択します。

宛先を含む SCA モジュールが単一サーバー上にインストールされている場合は、「**サーバー・スコープ (Server scope)**」を選択します。

宛先を含む SCA モジュールがクラスター上にインストールされている場合は、「**クラスター・スコープ (Cluster scope)**」を選択します。

5. 変更する宛先をリストから選択します。
6. 次のページの各種フィールドを新規の値に変更します。
7. 変更する宛先ごとに、ステップ 5 と 6 を繰り返します。
8. 構成変更を保管します。

## 次のタスク

ステップ 1 で停止したアプリケーションを再始動します。

### 関連情報

 [WebSphere MQ WebSphere の MQ 相互通信](#)

## Service Component Architecture モジュールと WebSphere MQ

SCA モジュールと **WebSphere MQ** キューを接続して、相互にサービスを提供することができます。

Service Component Architecture (SCA) モジュールは、他の SCA モジュールに対するのと同じように WebSphere MQ アプリケーションと通信できます。WebSphere MQ アプリケーションに要求を送信するモジュールは、そのアプリケーションに関連付けられた正しい応答および要求キューで構成されたインポートを使用します。同様に SCA モジュールは、適切なアプリケーション要求および応答キューで構成されたエクスポートを使用して、WebSphere MQ アプリケーションにサービスを提供できます。モジュールを作成するときは、SCA モジュールと WebSphere MQ キュー間の接続を定義します。

WebSphere MQ キュー・マネージャーの観点からは、SCA モジュールは通常の MQ クライアントであるかのように見えます。SCA モジュールの側からは、WebSphere MQ キューは他のサービスと同じように見えます。SCA モジュールと WebSphere MQ キューとの間のシールドを更に強化するには、SCA モジュールと WebSphere MQ キューの間にメディエーション・モジュールを使用します。これにより、メディエーションが元の SCA 要求をターゲット・キューに適した正しい形式に変換し、応答が使用可能になったときにそれを処理できるようになります。

**制約事項:** インポートおよびエクスポート用に WebSphere MQ を構成する場合は、以下の点に注意してください。

- ターゲット・クライアントを **MQ** に設定した状態でキュー宛先を構成する必要があります。
- TCP/IP クライアント接続を使用して、WebSphere MQ に接続する必要があります。詳しくは、「WebSphere MQ 相互通信」を参照してください。
- チャネル圧縮は使用できません。
- データを正しく処理するには、必要なチャネル出口をすべて設定する必要があります。これを行うには、「WebSphere MQ メッセージング・プロバイダー」 → 「WebSphere MQ 接続ファクトリー・オブジェクト (WebSphere MQ connection factory objects)」 → 「カスタム・プロパティ」をクリックして、以下のカスタム・プロパティを設定します。
  - **SENDEXIT**。このプロパティには、値 `com.ibm.ws.sca.internal.mq.exit.MQInternalSendExitImpl` を設定します。
  - **RECEXIT**。このプロパティには、値 `com.ibm.ws.sca.internal.mq.exit.MQInternalReceiveExitImpl` を設定します。
  - **SENDEXITINIT**。このプロパティには、選択したテキスト・ストリングを設定します。このプロパティを設定すると、トレースで出口を見分けやすくなります。
  - **RECEXITINIT**。このプロパティには、選択したテキスト・ストリングを設定します。このプロパティを設定すると、トレースで出口を見分けやすくなります。
- 出口では JMS MQ バインディングは禁止されるため、WebSphere MQ バインディングと Java Message Service (JMS) MQ バインディングに同じ接続ファクトリーを使用することはできません。

#### 関連情報



WebSphere MQ WebSphere の MQ 相互通信

---

## ターゲットの使用

ターゲットを使用すると、参照用に構成されたターゲットを変更して処理を変更できるため、柔軟性が向上します。

コンポーネントは、別のモジュール内のコンポーネントを呼び出すことができます。これにより、アプリケーション作成の時間とコストを最小限に抑えることができます。また、管理コンソールを使用することにより、アプリケーションを再作成または再デプロイすることなく、モジュール間呼び出しのエンドポイントを変更できます。これにより、インストール済みのアプリケーションは、処理の進歩やその他の変更からの利点を得られます。ターゲットを使用すると、こうしたさらなる柔軟性が得られます。

このような柔軟性を十分に生かすには、システムによるターゲットの命名方法について理解しておく必要があります。呼び出しモジュールからのリンクは、正しいターゲットに接続されている必要があります。

### ターゲット名

ターゲット名は、呼び出しコンポーネントがターゲットを呼び出す方法が元になっています。名前の形式は以下のとおりです。

#### 呼び出しタイプ

##### 名前の形式

**同期** Java Naming and Directory Interface (JNDI) 形式に準拠する名前。例えば、以下のような形式です。

```
folder/export/fullpath_to_target/target_component_name
```

**非同期** 次の形式の名前。

```
folder/calling_component_name/  
full_path_to_target_component/target_component_name
```

#### 複数の宛先

この名前は、非同期呼び出しと同じです。ただし、ターゲットは、実際にメッセージを複数の宛先コンポーネントに送信します。

#### 関連タスク

『インポート・ターゲットの変更』

参照のターゲットを変更することにより、アプリケーションの再コンパイルや再インストールを行わなくとも、更新されたコンポーネントをアプリケーションが柔軟に利用できるようになります。

## インポート・ターゲットの変更

参照のターゲットを変更することにより、アプリケーションの再コンパイルや再インストールを行わなくとも、更新されたコンポーネントをアプリケーションが柔軟に利用できるようになります。

### 始める前に

参照のターゲットを変更する前に、以下を実行する必要があります。

- 新規ターゲットが同じデータ・オブジェクト・タイプを使用することを確認する。

- モジュールがターゲットを同期的に起動しているか非同期的に起動しているかを確認する。
- 参照が単一のサービスをターゲットにしているか、複数のサービスをターゲットにしているかを確認する。

## このタスクについて

元のターゲットと同じインターフェースを持つ別のサービスが、モジュールで使用可能な新機能または改良機能を提供する場合、モジュールからのインポートのターゲットを変更します。

### 手順

1. 変更する参照を含むモジュールを停止します。
  - a. 管理コンソールを使用して、Service Component Architecture (SCA) モジュールを表示します。  
  
「アプリケーション」>「SCA モジュール」を使用してこのパネルにナビゲートします。
  - b. モジュールを選択し、「停止」を押します。表示は更新され、アプリケーションが停止済みとして表示されます。
2. 参照のターゲット宛先を変更します。

変更を行う方法は、モジュールがターゲットを呼び出す方法によって異なります。

呼び出しのタイプ	変更方法
単一ターゲット・サービス	<ol style="list-style-type: none"> <li>1. 管理コンソールを使用して「SCA モジュール」を表示します。「アプリケーション」&gt;「SCA モジュール」を使用してこのパネルにナビゲートします。</li> <li>2. 表示されたリストから、変更するターゲットを参照するインポートを含むモジュールを選択します。</li> <li>3. 「インポート」の横の正符号 (+) をクリックして、インポートのリストを展開します。</li> <li>4. 変更するインポートをリストから選択します。</li> <li>5. 「ターゲット」領域で、リストから「モジュール」を選択します。</li> <li>6. 「エクスポート」リストが最新の内容に更新されたら、新規ターゲット用のエクスポートを選択します。</li> <li>7. 「OK」をクリックして変更を保管します。</li> </ol>

呼び出しのタイプ	変更方法
複数のターゲット・サービス	<ol style="list-style-type: none"> <li>1. モジュールが存在するシステムのバスを表示します。「サービス統合」&gt;「バス」を使用してパネルにナビゲートします。</li> <li>2. <b>SCA.System.cellname.Bus</b> を選択します。</li> <li>3. 「宛先」をクリックしてバスの宛先ターゲットを表示します。</li> <li>4. 呼び出し側モジュールをターゲットに接続するインポートを表す宛先を選択します。この ID は <code>import</code> という語を含みます。</li> <li>5. 「コンテキスト・プロパティ (Context properties)」をクリックしてプロパティのリストを表示します。</li> <li>6. リスト内のターゲット・プロパティをクリックして、変更するプロパティを選択します。</li> <li>7. 「コンテキスト値 (Context value)」フィールドを新規宛先ターゲットに変更します。</li> <li>8. 「OK」をクリックして、「コンテキスト・プロパティ (Context properties)」パネルに戻ります。</li> <li>9. 「OK」をクリックして変更を保管します。</li> </ol>

3. 変更を保管します。プロンプトが出されたら、「保管」をクリックします。

### 次のタスク

モジュールを始動し、モジュールが予期される結果を受け取ることを確認します。

## J2C アクティベーション・スペックの削除

サービスを含むアプリケーションをインストールすると、システムによって J2C アプリケーションの仕様が作成されます。アプリケーションを再インストールする前に、この仕様を削除する必要がある場合があります。

### 始める前に

アプリケーションのインストールに失敗したために仕様を削除する場合、Java Naming and Directory Interface (JNDI) 名の中のモジュールとインストールできなかったモジュールの名前とが一致するようにしてください。JNDI 名の 2 番目の部分が、宛先をインプリメントしたモジュールの名前に相当します。例えば、`sca/SimpleBOCrsmA/ActivationSpec` の場合、**SimpleBOCrsmA** がモジュール名です。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

## このタスクについて

サービスを含むアプリケーションをインストールした後で間違って構成を保管したが、J2C アクティベーション・スペックが不要な場合は、その仕様を削除します。

### 手順

1. 削除するアクティベーション・スペックを見つけます。

仕様は「リソース・アダプター」パネルに表示されます。「リソース」>「リソース・アダプター」をクリックして、このパネルにナビゲートします。

- a. 「**Platform Messaging Component SPI Resource Adapter**」を見つけます。

このアダプターを見つけるには、スタンドアロン・サーバーの「ノード」スコープ、またはデプロイメント環境の「サーバー」スコープで作業する必要があります。

2. Platform Messaging Component SPI Resource Adapter に関連した J2C アクティベーション・スペックを表示します。

リソース・アダプター名をクリックすると、次のパネルが表示され、関連した仕様が表示されます。

3. 削除するモジュール名に一致した「**JNDI 名**」の仕様をすべて削除します。

- a. 該当する仕様の横にあるチェック・ボックスをクリックします。

- b. 「**削除**」をクリックします。

## タスクの結果

システムは、選択された仕様を表示から削除します。

## 次のタスク

変更を保管します。

---

## SIBus 宛先の削除

SIBus 宛先とは、アプリケーションでサービスを使用可能にするための関連付けのことです。宛先の削除が必要になる場合があります。

### 始める前に

アプリケーションのインストールに失敗したために宛先を削除する場合、宛先名の中のモジュールとインストールできなかったモジュールの名前とが一致するようにしてください。宛先の 2 番目の部分が、宛先をインプリメントしたモジュールの名前に相当します。例えば、`sca/SimpleBOCrsmA/component/test/sca/cros/simple/cust/Customer` の場合、**SimpleBOCrsmA** がモジュール名です。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

## このタスクについて

サービスを含むアプリケーションのインストール後に不注意で構成を保管した場合、または SIBus 宛先を必要としなくなった場合、その宛先を削除します。

**注:** このタスクは、SCA システム・バスからのみ宛先を削除します。サービスを含むアプリケーションを再インストールする前に、アプリケーション・バスからもその項目を削除する必要があります (このインフォメーション・センターの『管理』セクションの『J2C アクティベーション・スペックの削除』を参照してください)。

### 手順

1. 管理コンソールにログインします。
2. SCA システム・バスの宛先を表示します。

「サービス統合」>「バス」をクリックして、パネルにナビゲートします。

3. SCA システム・バスの宛先を選択します。

画面上で、「SCA.SYSTEM.cellname.Bus」をクリックします。ここで、*cellname* は、削除しようとしている宛先を持つモジュールが含まれているセルの名前です。

4. 削除するモジュールと一致するモジュール名を含む宛先を削除します。
  - a. 該当する宛先の横にあるチェック・ボックスをクリックします。
  - b. 「削除」をクリックします。

## タスクの結果

パネルには残りの宛先のみが表示されます。

## 次のタスク

これらの宛先を作成したモジュールに関連した J2C アクティベーション・スペックを削除します。

---

## エンタープライズ・アプリケーションの管理

コンソールの「エンタープライズ・アプリケーション」ページ (「アプリケーション」>「エンタープライズ・アプリケーション」とクリックして表示) を使用して、サーバーにインストールされているエンタープライズ・アプリケーションの表示および管理を行います。

アプリケーションの構成で指定した値を表示するには、リストからアプリケーション名をクリックします。「アプリケーション詳細」ページが開き、アプリケーションの構成プロパティ、および該当する場合はローカル・トポロジーが表示されます。このページから既存値を変更して、アプリケーションを構成する追加コンソール・ページにリンクできます。

エンタープライズ・アプリケーションを管理するには、その名前の横にあるチェック・ボックスをクリックしてから、以下のいずれかのボタンを使用します。

表 8. エンタープライズ・アプリケーションを管理するためのボタン

ボタン	結果のアクション
開始	<p>アプリケーションの実行を試みます。アプリケーションが正常に開始すると、アプリケーションの状態は以下のいずれかに変わります。</p> <ul style="list-style-type: none"> <li>• 開始 - アプリケーションは、すべてのデプロイメント・ターゲットで開始されました。</li> <li>• 一部開始 - アプリケーションは、1 つ以上のデプロイメント・ターゲットでまだ開始中です。</li> </ul>
停止	<p>アプリケーションの処理の停止を試みます。アプリケーションが正常に停止すると、アプリケーションの状態は以下のいずれかに変わります。</p> <ul style="list-style-type: none"> <li>• 停止 - アプリケーションは、すべてのデプロイメント・ターゲットで停止されました。</li> <li>• 一部停止 - アプリケーションは、1 つ以上のデプロイメント・ターゲットでまだ停止中です。</li> </ul>
インストール	<p>ウィザードが開き、エンタープライズ・アプリケーションまたはモジュール (.jar, .war, または .ear ファイルなど) をサーバーにデプロイできるようにします。</p>
アンインストール	<p>WebSphere Application Server の構成リポジトリからアプリケーションを削除し、さらに、構成の保管後にアプリケーション・モジュールのインストール先のすべてのノードのファイル・システムから、そのアプリケーション・バイナリーを削除します。</p>
更新	<p>ウィザードが開き、サーバー上にデプロイされているアプリケーション・ファイルを更新できるようにします。アプリケーション全体を更新することも、単一モジュール、単一ファイル、あるいはアプリケーションの一部のみを更新することもできます。新規ファイルまたはモジュールの名前が、サーバー上に既に存在するファイルまたはモジュールの名前と同じである場合は、既存のファイルまたはモジュールが、その新規ファイルまたはモジュールに置き換えられます。同じ名前のファイルまたはモジュールがない場合は、デプロイされているアプリケーションに追加されません。</p>
ファイルの除去	<p>デプロイされたアプリケーションまたはモジュールから、ファイルを削除します。このボタンは、構成リポジトリ、およびファイルがインストールされているすべてのノードのファイル・システムから、ファイルを削除します。</p>
エクスポート	<p>「アプリケーション EAR ファイルのエクスポート」ページが開きます。このページでエンタープライズ・アプリケーションを EAR ファイルにエクスポートできます。デプロイされているアプリケーションをバックアップしたり、そのバイnding情報を保存したりする場合にも、「エクスポート」アクションを使用します。</p>

表 8. エンタープライズ・アプリケーションを管理するためのボタン (続き)

ボタン	結果のアクション
DDL のエクスポート	「アプリケーション DDL ファイルのエクスポート」ページが開きます。このページでエンタープライズ・アプリケーションの EJB モジュール内の DDL ファイルをエクスポートできます。

アプリケーションの管理について詳しくは、WebSphere Application Server インフォメーション・センターを参照してください。

## Application Scheduler の管理

Application Scheduler によって、管理者は、WebSphere Process Server にインストールされているアプリケーションの開始および停止をスケジュールに入れることができます。管理コンソールの「Application Scheduler」パネルを使用して、インストール済みアプリケーションのスケジューリングを制御します。

また、スケジューラー・エントリーは、WebSphere InterChange Server のスケジューラー・エントリーが格納された WebSphere InterChange Server のリポジトリをマイグレーションする際に生成することができます (WebSphere InterChange Server からのマイグレーションと reposMigrate コマンドに関するトピックを参照してください)。管理コンソールの「Application Scheduler」パネルを使用して、これらのマイグレーション済みスケジューラー・エントリーを管理することもできます。

スタンドアロン・サーバー環境では、Application Scheduler は自動的にインストールされます。スタンドアロン・サーバーのプロファイルを作成すると、このサーバー上に Application Scheduler がインストールされて構成されます。

Network Deployment 環境の場合、すべての管理対象サーバーと作成済みのクラスター・メンバーに Application Scheduler が自動的にインストールされます。追加のアクションを実行する必要はありません。新しい管理対象サーバーおよびクラスター・メンバーの作成手順については、関連するタスクを参照してください。

WebSphere InterChange Server では、コラボレーション・オブジェクトやコネクタが定義されたアプリケーションを、コンポーネント・レベルで起動、一時停止、停止することができました (つまり、アプリケーションの残りの部分を続行する場合でも、コンポーネントを停止することができました)。WebSphere Process Server の場合、Application Scheduler を介してイベントのスケジュールを行います。

Application Scheduler により、アプリケーション・レベルでプロセスの開始と停止を制御することができます。

## Application Scheduler へのアクセス

Application Scheduler には、Application Scheduler MBean インターフェースを使用してプログラマチックにアクセスするか、または管理コンソールの「Application Scheduler」パネルを使用してアクセスします。

# Application Scheduler MBean インターフェースを使用した Application Scheduler へのアクセス

Application Scheduler MBean を起動するには、コマンド行から起動します。

## このタスクについて

Application Scheduler MBean を起動するには、以下を実行します。

### 手順

1. クラス `com.ibm.wbiserver.migration.ics.Parameters` に、プロパティー `SOAP_HOSTNAME` および `SOAP_PORT` を設定します。

このクラスは、`WAS_HOME\lib` ディレクトリーの `migration-wbi-ics.jar` ファイル内にあります。`SOAP_HOSTNAME` は、Application Scheduler が実行されているホストの名前です。`SOAP_PORT` は、Application Scheduler が実行されているポートです。

```
Parameters.instance.setProperty(Parameters.SOAP_HOSTNAME, "localhost");
Parameters.instance.setProperty(Parameters.SOAP_PORT, "8880");
```

**注:** セキュリティーがオンになっている場合、ロケーション

`WAS_HOME\profiles\profiles\properties\soap.client.props` にある `SOAP` プロパティー・ファイル内でユーザー ID およびパスワードを指定する必要があります。

このプロパティー・ファイル名を、以下に示す `Parameters` インスタンスに設定する必要があります。

```
Parameters.instance.setProperty(Parameters.SOAP_PROPERTIES,
"WAS_HOME\profiles\profiles\properties\soap.client.props");
```

2. `AppScheduler Mbean` への呼び出しを実装するクラス `com.ibm.wbiserver.migration.ics.utils.MBeanUtil` のインスタンスを作成します。

`MBeanUtil` のインスタンスを生成するには、コンストラクターにこの照会ストリングを渡す必要があります。これにより、コンストラクターが `name`、`type`、`server name`、および `node name` に基づいて正しい `Mbean` を起動します。

```
protected static final String WEBSHERE_MB_QUERY_CONSTANT = "WebSphere:*";
protected static final String NAME_QUERY_CONSTANT = ",name=";
protected static final String WBI_SCHED_MB_NAME = "Scheduler_AppScheduler";
protected static final String TYPE_QUERY_CONSTANT = ",type=";
protected static final String WBI_SCHED_MB_TYPE = "WASScheduler";
protected static final String SERVER_QUERY_CONSTANT = ",process=";
serverName = "<server1>";
protected static final String NODE_QUERY_CONSTANT = ",node=";
nodeName = "<myNode>";
```

```
String queryString = new StringBuffer(WEBSHERE_MB_QUERY_CONSTANT)
    .append(NAME_QUERY_CONSTANT)
    .append(WBI_SCHED_MB_NAME)
    .append(TYPE_QUERY_CONSTANT)
    .append(WBI_SCHED_MB_TYPE)
    .append(SERVER_QUERY_CONSTANT)
    .append(serverName)
    .append(NODE_QUERY_CONSTANT)
    .append(nodeName).toString();
```

```
MBeanUtil mbs = new MBeanUtil(queryString.toString());
```

3. MbeanUtil インスタンスの invoke() メソッドを使用して Mbean メソッドを呼び出し、メソッドの名前を渡します。

## 例

以下に、Scheduler Mbean の createSchedulerEntry メソッドを起動する例を示します。最初のステップでは、SchedulerEntry を作成し、name、type、version、transition、entry status、recurrence type、recurrence week、recurrence period、initial date、repeat interval、component id などの各種パラメーターを設定します。

```
try
{
    //First we set up the Schedule entry

    ScheduleEntry entry1 = new ScheduleEntry();
    entry1.setCName("BPEWebClient_localhost_server1");
    entry1.setCType("Application");
    entry1.setCVersion("ver1");
    entry1.setCTransition("startApplication");
    entry1.setSchedulerNumberOfRepeats(3); // Fire Three times
    entry1.setScheduleEntryStatus(TaskStatus.SCHEDULED);
    entry1.setRType(Recurrence.MINUTES);
    entry1.setRWeekNumber(-1);
    entry1.setRPeriod(2);
    entry1.setInitialDate(new Date(System.currentTimeMillis()+SIXTY_SECOND_OFFSET));
    entry1.setRepeatInterval(entry1.getInitialDate(), entry1.getRType(),
    entry1.getRWeekNumber(),
    entry1.getRPeriod());
    entry1.setComponentID(entry1.getCName(), entry1.getCType(),
    entry1.getCVersion(), entry1.getCTransition());
```

その後、Mbean の createSchedulerEntry メソッドを呼び出します。ScheduleEntry クラスの名前とともに、スケジューラー・エントリー entry1 をパラメーターとして渡します。

```
mbs.invoke(schedulerExtMBName, "createScheduleEntry", new Object[]{entry1},
    new String[]{"com.ibm.wbiserver.scheduler.common.ScheduleEntry"});
```

最後に、readAllScheduleEntries メソッドを呼び出して、追加したばかりのエントリーを含め、すべてのスケジュール・エントリーを読み込みます。

```
        result = mbs.invoke("readAllScheduleEntries", null, null);
    }
    catch (MigrationException e)
    {
        e.printStackTrace();
    }
```

## 管理コンソールを使用したスケジューラー・エントリーの表示

管理コンソールの「Application Scheduler」パネルを使用して、スケジューラー・イベントを作成、変更、または削除します。

### 始める前に

このタスクを実行するには、ユーザーはサーバーの管理コンソールを使用している必要があります。

### 手順

1. 「サーバー」 → 「アプリケーション・サーバー」 → 「*ServerName*」を選択します。
2. 副見出し「**Business Integration**」の「**Application Scheduler**」を選択します。
3. 表示するエントリーの有効範囲 (セル、ノード、サーバー) を選択します。

通常、スケジューラー項目はサーバー・スコープで定義されます。

## タスクの結果

そのスコープの既存のスケジュール済みイベントがリストされます。

これで、新しいスケジューラー・イベントの作成、既存のスケジューラー・イベントの編集、または既存のスケジューラー・イベントの削除を行うことができます。

## スケジュール済みイベントの作成

管理コンソールは、新規のスケジュール済みイベントを作成するためのパネルを提供します。

### 始める前に

新規のスケジュール済みイベントを作成するには、サーバー用管理コンソールの「Application Scheduler」コレクション・パネルを使用する必要があります。

### このタスクについて

特定のニーズを満たすイベントを作成しなければならないことがあります。新規のスケジュール済みイベントを作成するには、以下の手順に従います。

注: パネル上の「\*」が付いたフィールドは、必須フィールドです。

### 手順

1. 「新規」をクリックします。「追加」パネルが開きます。
2. スケジュール済みイベントを構成します。
  - a. 「グループ・アプリケーション」を選択します。
  - b. 「状況」を選択します。
  - c. 「初期日付」を *month, dd, yyyy* という形式で入力します (*month* の部分は月の略称を指定)。例えば 2005 年 4 月 15 日の場合は、**Apr 15, 2005** と入力します。
  - d. 12 時間表記の *hh:mm* を使用して「初期時刻」を入力し、am または pm のいずれかを指定して時間帯を入力します。

注: このフィールドから別のフィールドに移動すると、次の起動時刻が自動的に計算されます。

- e. 「アクション」を選択します。
- f. オプション: 「繰り返し」パラメーターを指定します。

#### • 開始期間

イベントの起動予定時刻に Application Scheduler または Process Server が稼働していない場合は、開始期間パラメーターにより、イベントの起動予

定時刻からの時間またはウィンドウの長さ (分) が指定されます。この時間内に Application Scheduler または Process Server が操作を再開した場合は、イベントが実行されます。ただし、この時間が経過するまでに Application Scheduler または Process Server が操作を再開しない場合は次の起動時刻が計算され、次回はその時刻にイベントが実行されます。

例えば、夜の 12 時に起動予定のイベントに対して開始期間を 60 分に設定した状態で、その時刻にサーバーがダウンしたとします。この場合は、サーバーが午前 1:00 より前にオンラインに復帰すれば、イベントが実行されます。

- スケジュールされた項目を指定した時刻に繰り返す必要があるかどうかを確認します。
  - 毎分、毎時、日次、月次、年次単位で 1 回以上。
  - 月単位の特定の週 (第 1 週、第 2 週、第 3 週、第 4 週、最終週) の特定の曜日 (日曜から土曜)。
  - 月単位の最終日

3. 「適用」または「OK」をクリックして、イベントを設定します。

注: 別のイベントを作成するには、「リセット」をクリックしてパネルをクリアします。

## タスクの結果

Application Scheduler によって、新規のスケジュール済みイベントが作成され、「Application Scheduler」パネルに表示されます。

## イベント状況およびアクションの説明

各イベントには、状況とアクションを指定する必要があります。

### 状況

「状況」フィールドには、イベントの状況が表示されます。この状況により、イベントをモニターします。次の表にそれぞれの状況を示します。

状況	説明
スケジュール済み	事前に指定された日時と間隔でタスクが実行されます。2 回目以降の実行時刻は計算によって求められます。
中断	タスクが中断された状態です。状況が「スケジュール済み」に変更されるまで、このタスクは実行されません。
完了	タスクが完了した状態です。
キャンセル済み	タスクがキャンセルされた状態です。このタスクは実行されず、再開することもできませんが、ページすることはできます。

状況	説明
無効	通常は、タスクがパージされているかタスクの照会に使用された情報が無効である場合に、この状況になります。
実行中	タスクが実行されている状態です。 <b>注:</b> イベントがモニターされるのは起動時の非常に短い時間に限られるため、この状況が表示されることはほとんどありません。

## アクション

各イベントにはアクションを関連付ける必要があります。アクションにより、イベントの処理方法が指定されます。イベントに対して実行できるのは、以下の 2 つのアクションだけです。

- **アプリケーションの開始** - システムのデプロイメント・マネージャーが管理するすべてのアプリケーションを開始します。
- **アプリケーションの停止** - システムのデプロイメント・マネージャーが管理するすべてのアプリケーションを停止します。

## スケジュール済みイベントの変更

管理コンソールから、移行したスケジュール済みイベントまたは既存のスケジュール済みイベントを変更します。

### 始める前に

スケジュール済みイベントを変更するには、サーバー用管理コンソールの「Application Scheduler」コレクション・パネルを使用する必要があります。

### 手順

1. 変更するイベントの「**スケジュール・エントリ ID**」をクリックします。「イベント」パネルが開きます。
2. 以下のフィールドのいずれかを変更します。

**注:** サーバー上のすべてのアプリケーションがリストされるため、既存のイベントの状況を変更する際には注意が必要です。サーバーで実行中のアプリケーションが停止する可能性があります。

- **グループ・アプリケーション**
- **状況**
- **初期日付** (*month, dd, yyyy* の形式で指定。month の部分には月の略称を指定)
- **初期時刻** (12 時間表記の *hh:mm* 形式で指定)
- **アクション**

**オプション: Recurrence** パラメーターに記入することもできます。

3. 「適用」または「OK」をクリックして、イベントの変更を設定します。

注: スケジュール済みイベントを変更する場合、新規の「スケジュール・エントリー ID」がサーバーから割り当てられます。サーバーは、現在スケジュールされているイベントを削除して、新規 ID を持つ新規のイベントをスケジュールに入れます。

## タスクの結果

新規 ID を持つ変更済みイベントが、「Application Scheduler」コレクション・パネルに表示されます。

## スケジュール済みイベントの削除

Application Scheduler は、スケジュール済みイベントを削除するためのパネルを提供します。

### 始める前に

スケジュール済みイベントを削除するには、サーバー用管理コンソールの「Application Scheduler」コレクション・パネルを使用する必要があります。

### このタスクについて

イベントが使用されなくなった場合、コレクション・パネルのイベントのリストからそのイベントを削除できます。スケジュール済みイベントを削除するには、以下の手順に従います。

#### 手順

1. 「選択」列で、削除するスケジュール・エントリーを選択します。
2. 「削除」をクリックします。

### タスクの結果

スケジュール・エントリーが削除されます。

---

## リレーションシップの管理

リレーションシップ・マネージャーは、手動でリレーションシップ・データを制御および操作するためのツールです。このツールで、自動化されたリレーションシップ管理で検出されたエラーを訂正し、より完全なリレーションシップ情報を提供します。特に、このツールには、リレーションシップ・インスタンス・データの検索および変更を行う機能があります。

### リレーションシップ・マネージャーが機能する仕組み

リレーションシップ・マネージャーによって、リレーションシップ・ランタイム・データ (ロールおよびそれらのデータを含む) を構成、照会、表示、および操作できます。Relationship Editor を使用して、リレーションシップの定義を作成します。実行時に、リレーションシップのインスタンスには、別のアプリケーションの情報を関連付けるデータが取り込まれます。このリレーションシップ・インスタンス・データは、マップまたは他の WebSphere Process Server コンポーネントが実行され、そこでリレーションシップ・インスタンスが必要とされる場合に作成されます。リ

リレーションシップ・サービスは、一連のアプリケーション・プログラミング・インターフェース (API) を公開して、リレーションシップ・メタデータの検索、インスタンス・データの作成、検索、および操作を行います。データは、リレーションシップ定義で指定されているリレーションシップ・テーブルに保管されます。リレーションシップ・マネージャーには、リレーションシップおよびリレーションシップ・インスタンスと対話するためのグラフィカル・ユーザー・インターフェースがあります。

リレーションシップ・マネージャーは、リレーションシップ・インスタンスごとに、ロールの階層リストを表示できます。リレーションシップの各ロールには、インスタンス・データ、プロパティ、およびキー属性があります。リレーションシップ・ツリーは、エンティティのタイプ、その値、最終変更日などの、リレーションシップ・インスタンス内の各ロールに関する詳細な情報も提供します。リレーションシップ・インスタンス ID は、リレーションシップ・インスタンスがリレーションシップ・テーブル内に保管されるときに、自動的に生成されます。リレーションシップ・マネージャーは、リレーションシップ・ツリーの最上位にこのインスタンス ID を表示します。

## リレーションシップ・マネージャーの使用

リレーションシップ・マネージャーを使用すると、全レベルでエンティティを管理できます。つまり、リレーションシップ・インスタンス、ロール・インスタンス、属性データ、プロパティ・データの各レベルです。例えば、リレーションシップ・マネージャーを使用して、以下を行うことができます。

- 既存のリレーションシップの値の表示および検査。
- リレーションシップ・インスタンスの作成および削除。
- リレーションシップ・インスタンスの内容の変更 (ロール・インスタンスの追加および削除など)。
- ロール・プロパティや論理状態などのリレーションシップ・ロール・インスタンスのデータの編集。
- ロール・インスタンスのアクティブ化および非アクティブ化。
- キー属性、開始日と終了日、およびプロパティ値を指定したロール・インスタンスの取得。
- 問題発生時の状態の修復。例えば、ソース・アプリケーションの破損データや不整合データが汎用および宛先アプリケーション・リレーションシップ・テーブルに送信された場合、リレーションシップ・マネージャーを使用して、データの信頼性が確信できる時点でデータをロールバックできます。

リレーションシップについて詳しくは、WebSphere Integration Developer インフォメーション・センター、および WebSphere Process Server インフォメーション・センターのリレーションシップ・サービスについてのトピックを参照してください。

## リレーションシップの表示

リレーションシップ名、表示名、静的属性、識別属性など、システム内のリレーションシップのリストを表示するには、このタスクを実行します。

## 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

## このタスクについて

システム内のリレーションシップのリストを表示するには、以下のステップを実行します。

### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。

情報は表形式で表示されます。各リレーションシップ・タイプはリンクです。

**ヒント:** 一度に表示される行数をカスタマイズできます。「設定」をクリックし、「最大行数」フィールド値を変更します。デフォルトは 25 です。

## リレーションシップ詳細の表示

リレーションシップ名、表示名、関連するロールとその属性、プロパティ値、静的属性、識別属性など、選択したリレーションシップの詳細情報を表示するには、このタスクを実行します。

## 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

## このタスクについて

選択したリレーションシップの詳細情報を表示するには、以下のステップを実行します。

### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. 次の 2 つの方法で、リレーションシップの詳細を表示できます。
  - a. リレーションシップ名をクリックします。
  - b. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「詳細」をクリックします。

リレーションシップの詳細には、表形式で表示されるロール属性が含まれます。ロール属性には、ロールの表示名、オブジェクト名、および管理対象属性設定が含まれます。

リレーションシップのリストに戻るには、ページ上部のパスから「リレーションシップ」をクリックするか、「戻る」をクリックします。

## ロール詳細の表示

リレーションシップ名、ロール名、表示名、プロパティ値、キー、ロール・オブジェクト・タイプ、および管理対象属性設定などの、選択されたロールに関する詳細情報を表示するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

### このタスクについて

選択したロールの詳細情報を表示するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーの「リレーションシップ」ページを開きます。
4. リレーションシップ名をクリックして、「リレーションシップの詳細」ページを開きます。
5. 「ロール・スキーマ情報」で、関連するロール名をクリックして、「ロールの詳細」ページを開きます。

### 次のタスク

「リレーションシップの詳細」ページに戻るには、ページの上部のパスから「リレーションシップの詳細」をクリックするか、または「戻る」をクリックします。

## リレーションシップの照会

リレーションシップ・ベースのインスタンス照会を実行するには、このタスクを使用します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

## このタスクについて

照会オプション (すべて、ID ごと、プロパティごと、またはロールごと) を選択して、リレーションシップ用のすべてのインスタンス・データまたはそのサブセットを取得します。その照会の結果セットが戻され、各行が 1 つのリレーションシップ・インスタンスを表す表形式で表示されます。

リレーションシップを照会するには、以下のステップを実行します。

### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれかをクリックし、検索基準を指定します。

オプション	説明
「すべて」タブ	リレーションシップのすべてのインスタンスのリストを取得します。表示する内容を選択できます (すべてのアクティブ化されたリレーションシップ・インスタンス・データ、すべての非アクティブ化されたリレーションシップ・インスタンス・データ、またはすべてのアクティブ化されたおよび非アクティブ化されたリレーションシップ・インスタンス・データ)。
「ID ごと」タブ	開始インスタンス ID と終了インスタンス ID の間の範囲のリレーションシップ・インスタンスを取得します。1 つのフィールドを空白のままにしておいた場合は、照会によって、インスタンスが 1 つのみ戻されます。照会によって、検出されたインスタンス用のすべてのロールが戻されます。
「プロパティごと」タブ	特定のプロパティ値に基づいてリレーションシップ・インスタンスを取得します。
「ロールごと」タブ	ロール名、キー属性値、ロールが作成または変更された日付範囲、または特定のプロパティ値に基づいてリレーションシップ・インスタンスを取得します。

6. 照会パラメーターを指定した後に行うオプションとして、以下があります。
  - 「OK」をクリックして、照会からの結果データを表示します。
  - 「キャンセル」をクリックして、すべての変更内容を破棄し、リレーションシップのリストに戻ります。

## データベース・ビューを使用したリレーションシップ・データの照会

データベースのビューを使用して、リレーションシップ・マネージャーを使用せずにリレーションシップ・データを照会することができます。

データベース・ビューを使用して、データベースに保管されたリレーションシップ・データを直接照会できます。これらのビューは基本的に、データベース表に保管されたリレーションシップ・データをカプセル化したものです。これらのビューでは以下の方法によって、リレーションシップ・データを入力したり、照会したりできます。

- DB クライアントで (例えば DB2 コマンド・センターで)、SQL ステートメントを使用
- Java プログラムで、SQL ステートメントを実行する JDBC を使用

いずれの場合も、表と同じように SQL ビューを使用できます。詳しくは、WebSphere Process Server バージョン 6.2.0 の技術情報を参照してください。

リレーションシップ・データベース SQL ビューは、データ・ソースの他の場所に配置された表に含まれるデータに基づいて作成されます。このビューは、データベース表自体が空であったとしても存在します。ビューにはそれぞれに固有の名前があります。名前は "V\_"+*relationship\_display\_name*+"\_role\_display\_name+"\_uuid という命名規則に従います (変数は下線文字 "\_" で連結されることに注意してください)。表示名は両方とも最大 20 文字の英数字で、UUID は両方の表示名の組み合わせから生成された数値です。したがって、各ビュー名は、データ・ソース内で固有となります。この命名規則に従って、例えば以下の変数を使用したとします。

- *relationship\_display\_name* = SAMPLECUSTID
- *role\_display\_name* = MYCUSTOMER
- *uuid* = 80C (この数値はサーバーによって自動的に生成)

この場合、ビュー名は "V\_SAMPLECUSTID\_MYCUSTOMER\_80C" となります。それぞれのリレーションシップには、2 つの対応するビューがあります。この 2 つのビューには同じリレーションシップ表示名が含まれる一方、ロールの表示名と UUID はそれぞれに異なります。

注: Oracle データベースの場合の命名規則は、*relationship\_display\_name* および *role\_display\_name* の最初の 10 文字のみが使用されるという点で異なります。

各ビューには、以下の表に記載する列 (タイプ、値、ヌル可能の関連プロパティを含む) が含まれます。

表 9. リレーションシップ・データベース・ビューの列

名前	データ型	値	ヌル可能かどうか
INSTANCEID	Integer	異なるアプリケーション間でインスタンス・データを相関させるために使用する ID 番号。	いいえ

表9. リレーションシップ・データベース・ビューの列 (続き)

名前	データ型	値	ヌル可能かどうか
ROLE_ATTRIBUTE_COLUMNS • 動的リレーションシップ - ビジネス・オブジェクトで定義 • 静的リレーションシップ - DATA	• 動的リレーションシップ - ビジネス・オブジェクトで定義 • 静的リレーションシップ - VARCHAR	列名と列タイプはロール定義に依存します。列名はキー属性名に基づく一方、列タイプは、ロール定義で定義されたキー属性タイプを基にマップされたデータベース・データ・タイプです。	いいえ
STATUS	Integer	0-4 注: ビューを使用してインスタンスを設定する場合には、この列の値が0であることを確認してください。	はい
LOGICAL_STATE	Integer	• 0 = アクティブ化 • 1 = 非アクティブ化  データベースにデータを設定するときには、適切な値が設定されることを確実にしてください。	いいえ
LOGICAL_STATE_TIMESTAMP	タイム・スタンプ	論理状態列のデータが最後に更新された日時。	はい
CREATE_TIMESTAMP	タイム・スタンプ	ロール・インスタンスが作成された日時。	はい
UPDATE_TIMESTAMP	タイム・スタンプ	ロール・インスタンスが最後に更新された日時。	はい
ROLEID	Integer	ロール・インスタンスを識別するために使用する ID 番号。	いいえ

関連概念

## リレーションシップ

リレーションシップとは、ビジネス・オブジェクトとその他のデータの間の関連のモデル化および保守に使用されるサービスです。

140 ページの『リレーションシップの管理』

リレーションシップ・マネージャーは、手動でリレーションシップ・データを制御および操作するためのツールです。このツールで、自動化されたリレーションシップ管理で検出されたエラーを訂正し、より完全なリレーションシップ情報を提供します。特に、このツールには、リレーションシップ・インスタンス・データの検索および変更を行う機能があります。

## リレーションシップ・インスタンスの表示

リレーションシップ照会に一致するリレーションシップ・インスタンスのリストを表示するには、このタスクを実行します。テーブル・ビューに表示される結果には、リレーションシップ・インスタンス ID とインスタンスに関連付けられたプロパティ値が含まれています。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

### このタスクについて

リレーションシップ照会に一致するリレーションシップ・インスタンスのリストを表示するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「Relationship Manager」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックし、検索基準を指定します。照会オプションの説明については、143 ページの『リレーションシップの照会』を参照してください。
6. 「OK」をクリックして、「リレーションシップ・インスタンス」ページを開きます。

### タスクの結果

照会と一致するリレーションシップ・インスタンスのリストがテーブル・ビューに表示されます。各リレーションシップ・インスタンスが 1 つの行に対応します。ページと戻されたインスタンスの合計カウントが、ページの下部に表示されます。

**ヒント:** 一度に表示される行数をカスタマイズできます。「設定」をクリックし、「行」フィールド値を変更し、「適用」をクリックします。デフォルトは 25 です。一度に表示されるレコードの最小数は 1 であり、最大数はすべてのレコードです。

ページをナビゲートするには、以下の操作を行います。

- インスタンスの次のセットを表示するには、進む矢印をクリックします。
- 前のページのインスタンスを表示するには、戻る矢印をクリックします。

**制約事項:** ソートを実行するにはサーバーから全照会結果セットを取得する必要があるため、多数のリレーションシップ・インスタンスをフィルタリングまたはソートすると、パフォーマンスの問題が発生する可能性があります。例えば、20,000 個のリレーションシップ・インスタンスを戻す照会のリレーションシップ・インスタンス・データをソートするには、その 20,000 個のインスタンスをソートする必要があります。合計カウント (ページの下部) は、予期されるリレーションシップ・インスタンス数の推定値を示すため、大量のデータ・セットのソートまたはフィルタリングによって長い待ち時間が発生するかどうかを判断できます。

サーバーから一度に読み取るインスタンス数をカスタマイズできるように照会ブロック・サイズのパラメーターを設定する方法については、リレーションシップ・サービスの構成についてのヘルプ・トピックを参照してください。

## リレーションシップ・インスタンス詳細の表示

リレーションシップ名、リレーションシップ・インスタンス ID、プロパティー値、参加しているロール、およびロール・インスタンス値 (ロール・インスタンス ID、論理状態、キー属性、およびプロパティー値) などの、選択されたリレーションシップ・インスタンスに関する詳細情報を表示するには、このタスクを実行します。複数のロールを同時に表示できます。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

### このタスクについて

選択したリレーションシップ・インスタンスの詳細情報を表示するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。

5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックして、検索基準を指定します。次に、「OK」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 次の 2 つの方法で、リレーションシップ・インスタンスの詳細を表示できます。
  - リレーションシップ・インスタンス ID をクリックします。
  - リレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択して、「詳細」をクリックします。

リレーションシップ・インスタンスのリストに戻るには、ページ上部のパスから「リレーションシップ・インスタンス」をクリックします。

## リレーションシップ・インスタンスの詳細の編集

選択されたリレーションシップ・インスタンスに関するプロパティ値を編集するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

選択されたリレーションシップ・インスタンスに関するプロパティ値を編集するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックして、検索基準を指定します。次に、「OK」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 次の 2 つの方法のいずれかで、リレーションシップ・インスタンスの詳細を表示します。
  - リレーションシップ・インスタンス ID をクリックします。
  - リレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択して、「詳細」をクリックします。
7. 必要に応じて、リレーションシップ・インスタンス・プロパティ値を変更します。

**制約事項:** 以前にリレーションシップ・インスタンスに対してプロパティ値が定義された場合のみ、プロパティ値を編集できます。

リレーションシップ・インスタンスを削除するには、ページ下部の「削除」をクリックします。

このページから、新規ロール・インスタンスを作成したり、既存のロール・インスタンスを削除したりすることもできます。作成または削除するには、そのロール・インスタンスを選択し、ロール・テーブルの下にある「作成」または「削除」をそれぞれクリックします。「作成」をクリックすると、新規ロール・インスタンスのキー属性値とプロパティ値を入力するための「新規ロール・インスタンス」ページが開きます。選択されたロール・インスタンス ID をクリックすると、ロール・インスタンスのプロパティ値を編集できます。

8. インスタンスおよびインスタンスのロールでの変更が終了した後に行うオプションとして、以下があります。
  - 「OK」をクリックして、変更内容をシステムに即時に保管します。
  - 「キャンセル」をクリックして、すべての変更内容を破棄し、「リレーションシップ・インスタンス」ページに戻ります。

## リレーションシップ・インスタンスの新規作成

新しいリレーションシップ・インスタンスを作成するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

リレーションシップ・インスタンスを新規作成するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して「作成」をクリックすると、「新規リレーションシップ・インスタンス」ページが開きます。
5. デフォルト値以外の値が必要な場合は、「値」フィールドにプロパティ値の情報を追加し、「OK」をクリックして、新規リレーションシップ・インスタンスをローカル側で保管します。

**注:** ロール・インスタンスがないとリレーションシップ・インスタンスを持つことはできないので、リレーションシップ・インスタンス用のロール・インスタンスを作成する必要もあります。

## リレーションシップ・インスタンスの削除

選択したリレーションシップ・インスタンスを削除するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

選択したリレーションシップ・インスタンスを削除するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックして、検索基準を指定します。次に、「OK」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 削除するリレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択します。
7. 「削除」をクリックします。

リレーションシップ・インスタンスは、システムから即時に削除されます。

## リレーションシップ・インスタンス・データのロールバック

このタスクを実行すると、リレーションシップ・インスタンス・データが指定の日に時にロールバックされます。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

ロールバック中に以下のアクションが実行されます。

- 特定の期間内に作成されたリレーションシップ・インスタンスがデータベースから削除される（物理的に削除される）。

- アクティブにされたリレーションシップ・インスタンスがデータベースから削除される (物理的に削除される)。
- 特定の期間内に非アクティブにされたリレーションシップ・インスタンスがアクティブにされる。

リレーションシップ・インスタンス・データをロールバックするには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. リレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「**ロールバック**」をクリックします。
5. 「**開始日**」および「**終了日**」フィールドにロールバックの期間を入力します。

**重要:** WebSphere Process Server サーバーとデータベース・サーバーが、同じ時間帯に設定されていることを確認してください。同じ時間帯に設定されていない場合は、ロールバックが失敗します。

6. 「**OK**」をクリックします。

リレーションシップ内のインスタンス・データのうち、指定された日時より後に作成されたすべてのインスタンス・データは、非アクティブ状態としてマークされます。

## ロール・インスタンス詳細の表示

ロール名、ロール・エレメント、キー属性とプロパティ値、状況、および論理状態などの、選択されたロール・インスタンスに関する詳細情報を表示するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、モニター、オペレーター、コンフィギュレーター、または管理者としてログインする必要があります。

### このタスクについて

選択したロール・インスタンスの詳細情報を表示するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。

3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックして、検索基準を指定します。次に、「OK」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 次の 2 つの方法のいずれかで、リレーションシップ・インスタンスの詳細を表示します。
  - リレーションシップ・インスタンス ID をクリックします。
  - リレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択して、「詳細」をクリックします。
7. ロール・インスタンスに関する詳細を表示するには、ロール・インスタンス・テーブルで、関連する ID をクリックします。

## ロール・インスタンス・プロパティの編集

選択されたロール・インスタンスに関するプロパティ値を編集するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

選択されたロール・インスタンスに関するプロパティ値を編集するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックして、検索基準を指定します。次に、「OK」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 次の 2 つの方法のいずれかで、リレーションシップ・インスタンスの詳細を表示します。
  - リレーションシップ・インスタンス ID をクリックします。

- リレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択して、「詳細」をクリックします。
7. ロール・インスタンス・テーブルで、ロール・インスタンス ID をクリックして、ロール・インスタンスの詳細を表示します。
  8. 必要に応じてロール・インスタンス・プロパティ情報を編集し、「OK」をクリックして、この変更をローカル側で保管します。

**制約事項:** 以前にリレーションシップ・インスタンスに対してプロパティ値が定義された場合のみ、プロパティ値を編集できます。

## ロール・インスタンスの新規作成

リレーションシップ用の新しいロール・インスタンスを作成するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

リレーションシップのロール・インスタンスを新規作成するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「Relationship Manager」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
5. 照会オプション・タブのいずれか（「すべて」、「ID ごと」、「プロパティごと」、または「ロールごと」）をクリックして、検索基準を指定します。次に、「OK」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 次の 2 つの方法のいずれかで、リレーションシップ・インスタンスの詳細を表示します。
  - リレーションシップ・インスタンス ID をクリックします。
  - リレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択して、「詳細」をクリックします。
7. 新規インスタンスを作成するロールを指定して、ロール・テーブルの下にある「作成」をクリックすると、「新規ロール・インスタンス」ページが開きます。
8. キー属性とロール・プロパティの値をそれぞれの「値」フィールドに入力し、「OK」をクリックして、新規ロール・インスタンスをローカル側で保管します。

**制約事項:** キー属性値は、ロール・インスタンスの作成時にのみ設定できます。変更内容をデータベースに適用した後では、この情報は変更できません。ただし、プロパティ値は後で編集できます。

## ロール・インスタンスの削除

リレーションシップの選択したロール・インスタンスを削除するには、このタスクを実行します。

### 始める前に

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーターまたは管理者としてログインする必要があります。

### このタスクについて

リレーションシップの選択したロール・インスタンスを削除するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「**統合アプリケーション**」 > 「**Relationship Manager**」をクリックします。
3. そのリレーションシップ・サービス MBean の横にある「**リレーションシップ**」をクリックして、管理するサーバーのリレーションシップ・ページを開きます。
4. リレーションシップ名の隣にあるラジオ・ボタンを選択して、「**照会**」をクリックします。
5. 照会オプション・タブのいずれか（「**すべて**」、「**ID ごと**」、「**プロパティごと**」、または「**ロールごと**」）をクリックして、検索基準を指定します。次に、「**OK**」をクリックして「リレーションシップ・インスタンス」ページを開きます。
6. 次の 2 つの方法のいずれかで、リレーションシップ・インスタンスの詳細を表示します。
  - リレーションシップ・インスタンス ID をクリックします。
  - リレーションシップ・インスタンス ID の隣にあるラジオ・ボタンを選択して、「**詳細**」をクリックします。
7. ロール・インスタンスを削除する対象のロールを指定します。
8. 削除するロール・インスタンスの隣にあるラジオ・ボタンをクリックして、ロール・テーブルの下にある「**削除**」をクリックします。

ロール・インスタンスは、ローカル側で削除されます。

## リポジトリからのリレーションシップ・インスタンス・データの除去

リレーションシップを使用するアプリケーションは、リポジトリ内のリレーションシップ・スキーマとデータを関連付けています。リポジトリは、リレーションシップ・インスタンス・データを保持するように構成されたデータベースです。こ

のようなアプリケーションを実動サーバーからアンインストールしても、サーバーは、リレーションシップ・スキーマとデータをリポジトリから除去しません。これらを除去するには、既存のリレーションシップ・スキーマを手動で除去する必要があります。

## 始める前に

リレーションシップ・スキーマを使用するアプリケーションが、そのスキーマにアクセスするすべてのサーバーからアンインストールされていることを確認します。

## このタスクについて

リレーションシップを含むアプリケーションをインストールすると、サーバーは、テーブル、索引、シーケンス、およびストアド・プロシージャを含む、対応するデータベース・スキーマ・オブジェクトを作成します。実行時、このテーブルには、リレーションシップ・インスタンス・データが設定されます。リレーションシップを含むアプリケーションをアンインストールする場合、テーブルおよびインスタンス・データは、データベースから除去されません。この設計は、リレーションシップまたはロール定義の変更後にアプリケーションの再インストールを試みる場合に問題を起こすことがあります。

**注:** WebSphere Integration Developer (WID) の Unit Test Environment (UTE) テスト・サーバーを使用する場合、アプリケーション・プロジェクトを除去すると、リレーションシップ・スキーマとデータがリポジトリから除去されます。

同じリレーションシップを持つアプリケーションを再インストールする場合、古いスキーマは再利用されます。ただし、リレーションシップまたはロール定義を変更して、既存のスキーマと互換性がなくなるようにした場合、リレーションシップ・サービスは、例外をスローして、リレーションシップのインストールを終了します。ログには、以下の例外とメッセージが示されます。

```
RelationshipServiceException("table <tablename> already exists, but the table schema is different from current role definition")
```

この問題を解決するには、リポジトリのデータベース・プラットフォームが提供するツールを使用して、既存のリレーションシップ・スキーマ成果物を手動で除去し、アプリケーションを再インストールします。

既存のリレーションシップ・スキーマをリポジトリから除去するには、以下のステップを実行します。

### 手順

1. データベースのロケーションを探します。 データベースのロケーションは、データベース・プラットフォームによって異なります。

オプション	説明
データベース・プラットフォーム	ロケーション
Derby	WASHOME¥derby¥databases¥RepositoryDB

オプション	説明
その他のデータベース	<p>リレーシオンは、サーバーのインストールおよびプロファイル作成時に構成されます。例えば、サーバーを自動的に構成し、デフォルト・データベース名を選択した場合、データベースの名前は WPRCSDB となります。</p> <p>DB2® for i5/OS の場合、参照先コンテナは、データベースではなくコレクションです。これは、インストールおよびプロファイル作成中に構成されたデータベース名ではなく、コレクション名です。つまり、デフォルトの WPRCSDB という名前のデータベースではなく、コレクションです。</p>

2. リレーシオンシップを構成するデータベース成果物を削除します。指定のリレーシオンシップのデータベース・オブジェクトをすべて削除するには、データベース・プラットフォームのツールを使用して、以下のステップを実行します。
  - a. 以下のステップでデータベースからデータを除去する前に、データベースのバックアップを作成します。

**注:** DB2 for i5/OS の場合、データを除去する前にコレクションのバックアップを作成してください。
  - b. リレーシオンシップ・テーブルを検索します。リレーシオンシップのインストール時に、以下のテーブルが作成されます。

テーブル	フォーマット
リレーシオンシップ・プロパティ用に 1 つのテーブル	_ <b>&lt;relnam&gt;</b> _P_uniqueidentifier
各リレーシオンシップのインスタンス ID の生成用に 1 つのテーブル (Derby)	_ <b>&lt;relnam&gt;</b> _S_uniqueidentifier
アプリケーション固有の各ロールのロール・プロパティ用に 1 つのテーブル	_ <b>&lt;relnam&gt;</b> _<b>rolenam</b>_P_uniqueidentifier
アプリケーション固有のロールごとに 1 つのテーブル (静的リレーシオンシップの場合は、汎用ロール用にも 1 つのテーブルが作成されます)	_ <b>&lt;relnam&gt;</b> _<b>rolenam</b>_RT_uniqueidentifier

**制約事項:** リレーシオンシップ名は、先頭の 4 文字のみが使用されます。複数のリレーシオンシップ用のテーブルをデータベースが保持している場合は、リレーシオンシップ名を先頭の 4 文字以内で区別する必要があります。

- c. ストアード・プロシージャを検索します。ストアード・プロシージャ・オブジェクトの形式は以下のとおりです。

\_**<relnam>**\_RS\_uniqueidentifier または  
\_**<relnam>**\_<b>rolenam</b>\_RS\_uniqueidentifier

- d. シーケンスを検索します。シーケンス・オブジェクトの形式は、以下のとおりです。

\_`<relnam>`\_S\_uniqueidentifier

**制約事項:** Derby では、シーケンスはサポートされません。

- e. データベース・プラットフォームのツールを使用して、以下を削除します。
  - 1) テーブル
  - 2) ストアド・プロシージャ
  - 3) シーケンス (Derby を除く)

## タスクの結果

リレーションシップ・インスタンス・データが、データベース・リポジトリから除去されます。

## 次のタスク

以上で、アプリケーションを再インストールできます。

## チュートリアル: リレーションシップ・マネージャーの管理

リレーションシップ・マネージャーを使用して、同じデータ項目に対して複数の異なる環境から取得された ID を相関させるリレーションシップのインスタンスを追加、変更、および除去できます。このチュートリアルでは、リレーションシップ・マネージャーの基本機能を説明します。

このチュートリアルでは、WebSphere Process Server リレーションシップ・マネージャーの基本機能を説明します。リレーションシップは、同じデータ項目に対して複数の異なる環境から取得された ID を相関させるために使用されます。例えば、1つの環境では、州名は 2 文字の略語 (AZ, TX) として識別されます。別の環境では、異なる略語 (Ariz., Tex.) が使用されます。リレーションシップは、最初の環境の「AZ」を 2 番目の環境の「Ariz」と相関させるために作成されます。

ここで説明するサンプル・リレーションシップは、カスタマー ID を相関させるリレーションシップです。多くのビジネス・アプリケーションでは、カスタマーのデータベースを管理し、各カスタマーにそのアプリケーション独自の ID を割り当てています。エンタープライズ環境では、同じカスタマーがビジネス・アプリケーションごとに異なる ID を持つ可能性があります。このチュートリアルでは、カスタマー ID を相関させるリレーションシップが定義されます。リレーションシップ名は「SampleCustID」です。このリレーションシップには、2 つのロールが定義されています。1 つのロールは Customer Information System (CIS) 用で、もう 1 つのロールは General Ledger (GL) アプリケーション用です。このリレーションシップは、リレーションシップ・サービス・サンプルに基づいて作成され、ロールと少量のサンプル・データも提供されています。

リレーションシップ・マネージャーは、リレーションシップ・インスタンスを追加、変更、および除去できるだけでなく、リレーションシップ・インスタンスのロール・インスタンスも追加、変更、および除去できるように設計されています。新規のリレーションシップ定義を作成およびデプロイするには、WebSphere Integration Developer を使用する必要があります。定義は XML ファイルとして保管され、J2EE アプリケーションの一部として特定のサーバーにデプロイされます。

## このチュートリアルの目的

このチュートリアルを完了すると、リレーションシップ・インスタンスの値を変更できるようになります。

## このチュートリアルを完了するのに必要な時間

このチュートリアルを完了するのに必要な時間は、約 10 分です。

## 前提条件

このチュートリアルでは、リレーションシップ・サービス・テクニカル・サンプルに基づいて作成されたリレーションシップを使用します。このチュートリアルを手順に従って実行する前に、サンプル・ギャラリーにアクセスし、リレーションシップ・サービス・サンプルで説明されている手順を実行して、必要なリレーションシップおよびロールを作成します。

### 関連タスク



サンプル・ギャラリーのインストールおよびアクセス

統合アプリケーション成果物のサンプルは サンプル・ギャラリーに用意されています。このギャラリーは、本製品のインストール時に オプションでインストールできます。統合アプリケーション成果物のサンプルは サンプル・ギャラリーに用意されています。

## 例: リレーションシップ・インスタンスの値の変更

リレーションシップ・インスタンスでは、キー属性の値を管理コンソールの「リレーションシップ・インスタンス」ページで変更できます。この例では、このページを使用して、リレーションシップ・インスタンスの値を変更する方法を示します。

## このタスクについて

ある顧客の CIS アプリケーション内での顧客 ID が A004 であるとします。この顧客の GL アプリケーション内での顧客 ID は 801 です。しかし、データ入力エラーがあったため、この顧客の顧客 ID を関連させるリレーションシップ・インスタンスでは、現時点で GL 顧客 ID の値が 801 ではなく 901 になっています。このチュートリアルでは、リレーションシップ内のこの入力を訂正する手順を説明します。

## 手順

1. 管理コンソールを開きます。
2. セキュリティーが使用可能に設定されている場合は、管理者特権を持ったユーザーとしてログインします。
3. ナビゲーション・ペインで、「統合アプリケーション」 → 「**Relationship Manager**」をクリックします。
4. 管理するサーバーのリレーションシップ・ページを開きます。 そのリレーションシップ・サービス MBean の横にある「リレーションシップ」をクリックします。

SampleCustID という名前のリレーションシップが表示されます。

5. SampleCustID の隣にあるラジオ・ボタンを選択して、「照会」をクリックします。
6. 顧客のリレーションシップ・インスタンスを見つけます。
  - a. 照会の「**ロールごと**」タブをクリックします。
  - b. 「**ロール名**」フィールドで、ドロップダウン・リストから「MyGLCustomer\_0」を選択します。
  - c. 「**キー属性**」の下の「**値**」フィールドに 901 と入力します。
  - d. 「**OK**」をクリックします

これによって、要求されたカスタマーのリレーションシップ・インスタンスが検索され、「リレーションシップ・インスタンス」ページが開きます。

7. リレーションシップ・インスタンス ID をクリックします。

これによって、GL アプリケーション内のカスタマー ID 901 のリレーションシップ・インスタンス・データ (すべての関連するロール・インスタンスを含む) が表示されます。

8. MyGLCustomer\_0 ロール・テーブルで、キー属性値 901 を持つロール・インスタンス ID を選択し、ロール・テーブルの下にある「**削除**」をクリックします。

注: 関連付けられたプロパティ値がないことを確認してください。他のデータが表示された場合は、ロール・インスタンスを確認して、保持したいデータをすべて記録する必要があります。

9. 「**作成**」をクリックすると、このリレーションシップ・インスタンスのロール・インスタンスを新規作成するための「**新規ロール・インスタンス**」ページが開きます。
10. 「**キー属性**」の下にある「**値**」フィールドに 801 と入力し、「**OK**」をクリックします。

新規ロール・インスタンスが保管され、テーブル内に新規ロール・インスタンスが表示されます。

## タスクの結果

リレーションシップ・インスタンス内の GL アプリケーションのカスタマー ID 値が訂正されました。

---

## リレーションシップ・サービスの管理

リレーションシップ・サービスは、システム内のリレーションシップおよびロールを保守します。これは、リレーションシップとロールの定義およびメタデータを管理します。また、リレーションシップの定義を指定して、定義から導き出されるインスタンスの操作を可能にします。

リレーションシップ・サービスにより、さまざまなオブジェクトの間のリレーションシップを取り込むことができます。リレーションシップの参加者は、果たすロールによって識別されます。例えば、Person オブジェクト「ジョー」は Car オブジェクト「スバル (ナンバー・プレート、XYZ 123)」との間に所有権リレーションシッ

プを持つことができます。この例では、ジョーが「所有者」のロールでリレーションシップに参加し、自動車が「所有オブジェクト」のロールでリレーションシップに関与しています。

## リレーションシップおよびロール定義

リレーションシップおよびロールは、WebSphere Integration Developer のリレーションシップ・エディター・ツールのグラフィカル・インターフェースで設計した定義で記述されます。リレーションシップ定義は、リレーションシップの概要を記述し、リレーションシップの各参加者が実行可能なロールを識別するテンプレートです。ロール定義により、参加者についての構造および制約要件を指定します。リレーションシップの定義は XML ファイルとして保管され、J2EE アプリケーションの一部として特定のサーバーにデプロイされます。

リレーションシップの作成、リレーションシップ・タイプの識別、およびリレーションシップ・エディターの使用に関する背景情報およびタスク情報については、WebSphere Integration Developer インフォメーション・センターを参照してください。

## リレーションシップの機能の仕組み

マップまたは WebSphere Process Server の他のコンポーネントの実行時にリレーションシップ・インスタンスが必要になると、リレーションシップのインスタンスは、シナリオに応じて作成または取得されます。リレーションシップおよびロール・インスタンス・データは、以下の 3 つの方法で操作できます。

- WebSphere Process Server コンポーネント Java スニペットによるリレーションシップ・サービス API の呼び出し
- WebSphere Process Server ビジネス・オブジェクト・マッピング・サービスでのリレーションシップ変換
- リレーションシップ・マネージャー・ツールの使用

リレーションシップおよびロールのインスタンス・データは、リレーションシップ・サービスの構成時に指定したデフォルト・データ・ソースのデータベースに格納されるリレーションシップ・テーブルに保管されます。

リレーションシップ・サービスは、各サーバー上でセル・レベルで実行されます。「リレーションシップ・マネージャー」ホーム・ページの「リレーションシップ・サービスについて (About)」セクションに、リレーションシップ・サービスを実行中のセル内のサーバー数が表示されます。「リレーションシップ」セクションに、リレーションシップ・サービスを実行中の各サーバー名が表示されます。リレーションシップ・インスタンスで作業する前に、管理するリレーションシップおよびロールのインスタンスを持つサーバーを選択する必要があります。

リレーションシップ・マネージャーの使用については、WebSphere Process Server インフォメーション・センターの リレーションシップ・マネージャー に関するトピックを参照してください。

以下のトピックでは、ご使用の WebSphere Process Server 環境のリレーションシップ・サービスのために実行する必要がある構成タスクについて説明します。

## リレーションシップ・サービスで管理されるリレーションシップの表示

このリレーションシップ・サービスで管理される既存のリレーションシップのリストを表示するには、このタスクを実行します。

### 始める前に

このタスクに必要なセキュリティ・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、どの WebSphere セキュリティー・ロールでもこの構成を表示できます。

### このタスクについて

リレーションシップ・リストを表示するには、以下のステップを実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」 > 「**Relationship Manager**」をクリックします。
3. 「リレーションシップ・サービス構成」 > 「リレーションシップ」をクリックします。

「リレーションシップ・コレクション」ページが表示されます。各行は、関連するリレーションシップのバージョンとデータ・ソースを示します。

**ヒント:** 一度に表示される行数をカスタマイズするには、「設定」をクリックします。「最大行数」フィールドの値を変更し、「適用」をクリックします。デフォルトは 25 です。このリレーションシップ・サービスによって管理されるリレーションシップの合計カウントがページの下部に表示されます。

### 次のタスク

リレーションシップの構成プロパティーを表示するには、リレーションシップ・コレクション・テーブルのリレーションシップ名をクリックします。

## リレーションシップ・プロパティーの表示

リレーションシップ・サービスがリレーションシップ・サービス・レベル (リレーションシップ・サービスに適用されるとき) と個々のリレーションシップ・レベル (個々のリレーションシップに適用されるとき) の両方のレベルで管理する構成プロパティーを表示するには、このタスクを実行します。

### 始める前に

このタスクに必要なセキュリティ・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、どの WebSphere セキュリティー・ロールでもこの構成を表示できます。

## このタスクについて

構成プロパティを表示するには、以下のステップを実行します。

### 手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. 「リレーションシップ・サービス構成」>「リレーションシップ」をクリックします。
4. リレーションシップ・コレクション・テーブルで、プロパティを表示したいリレーションシップの名前をクリックします。

タブ付きの構成ページが表示され、そのリレーションシップの現在使用中の名前、バージョン、およびデータ・ソース (読み取り専用) が示されます。

**注:** バージョンは、マイグレーションの目的で使用されます。新しいシステム内で古いリレーションシップ・データを共存させる必要がある場合は、古いインフラストラクチャーのバージョンが古いバージョンに設定されます。それ以外の場合は、現行バージョンに設定されます。

5. 「リレーションシップ・コレクション」ページに戻るには、「戻る」をクリックします。



---

## 第 6 章 Business Process Choreographer の管理

Business Process Choreographer を管理する方法については、WebSphere Process Server for Multiplatforms バージョン 6.1 のインフォメーション・センターにアクセスし、『**WebSphere Process Server の管理**』>『**Business Process Choreographer の管理**』のトピックを参照してください。この情報は、「*Business Process Choreographer*」の PDF にも記載されています。



---

## 第 7 章 Common Event Infrastructure の構成および管理

Common Event Infrastructure を構成し、管理する方法については、WebSphere Process Server for Multiplatforms バージョン 6.1 のインフォメーション・センターにアクセスし、『**WebSphere Process Server の管理**』>『**Common Event Infrastructure の構成**』、および『**WebSphere Process Server の管理**』>『**Common Event Infrastructure の管理**』のトピックを参照してください。この情報は、「*Common Event Infrastructure*」の PDF にも記載されています。



---

## 第 8 章 サービス・コンポーネントの管理

このセクションのトピックを使用して、サービス・コンポーネントを管理します。

ビジネス・プロセスおよびヒューマン・タスクの管理については、WebSphere Process Server for Multiplatforms バージョン 6.1 のインフォメーション・センターで『WebSphere Process Server の管理』>『サービス・コンポーネントの管理』のトピックを参照するか、「Business Process Choreographer」の PDF を参照してください。

---

### ビジネス・ステート・マシンの管理

ビジネス・ステート・マシンのインスタンスのデバッグおよび管理を行うために、相関セットの値および表示状態の変数を表示できます。

ビジネス・ステート・マシンは、イベント・ドリブン・ビジネス・プロセスを表すために使用されます。ビジネス・ステート・マシン内には多くのインスタンスがあります。以下を使用すると、ビジネス・ステート・マシンのインスタンスの管理およびデバッグが可能です。

- 相関セットのプロパティ
- 表示状態

#### 相関セットのプロパティ

ビジネス・ステート・マシンのインスタンスを相互に区別するために、相関セットを使用してステート・マシンのインスタンスを一意的に識別します。例えば、顧客 ID と状態を相関セットのプロパティにすることができます。特定のインスタンスを管理する場合は、相関セット・プロパティの値が必要です。相関セット・プロパティは WebSphere Integration Developer で定義され、Business Process Choreographer Explorer に表示されます。

WebSphere Integration Developer で定義できる相関セットは 1 つのみです。複数の相関セットは使用できません。

#### 表示状態

表示状態変数は、特定のビジネス・ステート・マシンのインスタンスの現行の状態を示します。最後にコミットされた状態を調べると、ビジネス・ステート・マシンのデバッグや管理に役立ちます。表示状態は WebSphere Integration Developer で定義され、Business Process Choreographer Explorer に表示されます。

表示状態変数は、必ずしもビジネス・ステート・マシンのインスタンスの最新の状態を示すとは限りません。インスタンスが次々とイベントを処理している場合は、表示状態変数のメモリー内コピーが最後にコミットされた値と異なる場合があります。Business Process Choreographer Explorer に表示されるのは、最後にディスクに書き込まれた表示状態の値です。ビジネス・ステート・マシンのインスタンスがイベントを処理している場合には、トランザクションが完了するまで、変数のメモリー

一内の値はディスクに書き込まれません。

## ビジネス・ステート・マシンのインスタンスの検索

特定のビジネス・ステート・マシンのインスタンスを検索して管理するための、関連セット・プロパティを表示します。

### 始める前に

WebSphere Integration Developer 内で関連セットを定義し、モジュールを保管します。モジュールをサーバーにデプロイします。

### このタスクについて

関連セット・プロパティの値によって、1 つのビジネス・ステート・マシンのインスタンスがライフ・サイクルにわたって他のインスタンスから区別されます。特定のビジネス・ステート・マシンのインスタンスを終了させる必要がある場合、関連セット・プロパティの値によって正しいインスタンスを識別できます。Business Process Choreographer Explorer を使用して、関連セット・プロパティを表示するには、以下の手順を実行します。

**制約事項:** 1 つのビジネス・ステート・マシンに対して定義できる関連セットは 1 つのみです。複数の関連セットは使用できません。

### 手順

1. 「プロセス・テンプレート」の下で、使用しているビジネス・ステート・マシンを表すプロセス・テンプレートを選択します。
2. 「プロセス・テンプレート名」の下で、プロセス・テンプレートを選択し、「インスタンス」をクリックして、システム内で現在もアクティブな、存在するすべてのインスタンスを表示します。
3. インスタンスごとに、そのインスタンスをクリックしてから「照会プロパティ」タブをクリックして、「プロパティ名」の下に関連セット・プロパティを表示します。

### 次のタスク

管理用タスクを実行します。

## 表示状態の表示

ビジネス・ステート・マシンのインスタンスを管理したり、デバッグしたりする場合に、表示状態を表示します。

### 始める前に

WebSphere Integration Developer で表示状態変数を初期化して、モジュールを保管します。モジュールをサーバーにデプロイします。

### このタスクについて

表示状態変数を使用すると、アクティブなビジネス・ステート・マシンのインスタンスの現在の状態を表示できます。例えば、あるビジネス・ステート・マシンのイ

インスタンスが予想通りに応答しない場合は、アクティブなビジネス・ステート・マシンのインスタンスを表示して、現在の状態を判別し、問題をデバッグできます。そのアクティブなビジネス・ステート・マシンのインスタンスの関連セット・プロパティの値が必要です。アクティブなビジネス・ステート・マシンのインスタンスの現在の状態を表示するには、Business Process Choreographer Explorer で次の手順を実行します。

#### 手順

1. 「プロセス・テンプレート」の下で、使用しているビジネス・ステート・マシンを表すプロセス・テンプレートを選択します。
2. 「プロセス・テンプレート名」の下で、プロセス・テンプレートを選択し、「インスタンス」をクリックして、システム内で現在もアクティブな、存在するすべてのインスタンスを表示します。
3. インスタンスごとに、そのインスタンスをクリックしてから「照会プロパティ」タブをクリックして、「プロパティ名」の下に関連セット・プロパティと表示状態を表示します。

#### 次のタスク

管理用タスクを実行します。

---

## ビジネス・ルールおよびセレクトターの管理

ビジネス・ルールおよびセレクトターは、基準に基づいて処理の結果を変更できるため、ビジネス・プロセスに柔軟性が提供されます。ビジネス・ルールおよびセレクトターのコンポーネントを含むアプリケーションをインストールする前に、ビジネス・ルール動的リポジトリをインストールする必要があります。スタンドアロン・サーバー用または Network Deployment 用のビジネス・ルール動的リポジトリをインストールできます。

ビジネス・ルールまたはセレクトターを含むモジュールをインストールしたり、またはサーバーのビジネス・ルールおよびセレクトターを変更するたびに、システム・ログ、またはビジネス・ルールおよびセレクトター監査ロギングの構成時に指定した別のログに、更新内容が記録されます。

### ビジネス・ルールおよびセレクトターを含むモジュールの考慮事項

ここでは、ビジネス・ルールおよびセレクトターを含むモジュールをインストールまたは削除するときに考慮すべき情報をいくつか説明します。

ビジネス・ルールおよびセレクトターによって、モジュールの柔軟性が増します。柔軟性が増したことにより、モジュールをインストールまたは削除する方法に影響が及びます。サーバーが、中央リポジトリにビジネス・ルールおよびセレクトターを保管しているためです。

### ビジネス・ルールまたはセレクトターの変更に関する考慮事項

ビジネス・ルールおよびセレクトターは、影響があるモジュールを再構成および再インストールすることなく、実稼働環境で変更できます。これらの変更は、直接リポジトリに対して行われ、ビジネス・ルールまたはセレクトターを含むファイルには

コピーされません。ビジネス・ルールまたはセクターを変更後、そのビジネス・ルールまたはセクターをエクスポートし、開発環境にインポートしてください。ビジネス・ルールおよびセクターのエクスポートおよびインポートに習熟していない場合は、これらのタスクについて説明しているトピックを参照してください。

## **ビジネス・ルールまたはセクターを含むモジュールの置換に関する考慮事項**

ビジネス・ルールまたはセクターを含むモジュールを置換する場合、サーバーはリポジトリ内のビジネス・ルールおよびセクターのコピーを上書きします。動的に行ったすべての変更は、モジュールを置換するときに失われます。変更を失わないようにするには、モジュールが使用するビジネス・ルールおよびセクターをエクスポートして、開発環境に再度インポートし、実動システム上のモジュールを置換する前に、そのモジュールを再ビルドしてください。

あるモジュールによって実装されているビジネス・ルールまたはセクターを変更した場合、サーバー内で稼働中の他のモジュールには、そのビジネス・ルールまたはセクターの現在のコピーが必要です。この場合、更新したモジュールをサーバーにインストールする時に、このモジュールが他のモジュールに影響を与えないよう、別のリポジトリを構成する必要があります。『環境の構成』トピックでは、データベースの構成について説明します。

## **ビジネス・ルールまたはセクターを含むモジュールの削除に関する考慮事項**

ビジネス・ルールまたはセクターを含むモジュールをサーバーから削除しても、サーバーはビジネス・ルールおよびセクターをリポジトリから削除しません。サーバーは、他のアプリケーションまたはモジュールがこれらのビジネス・ルールを必要としているかどうかを判別できないため、これらの成果物を保持します。

ビジネス・ルールまたはセクターを必要とするものがないと判断した場合は、リポジトリから除去してください。『リポジトリからのビジネス・ルールおよびセクター・データの除去』では、不要なビジネス・ルールまたはセクターの消去方法について説明します。

## **リポジトリからのビジネス・ルールおよびセクターのデータの除去**

ビジネス・ルールやセクターを使用するアプリケーションをアンインストールする場合、サーバーはリポジトリからその成果物を除去しません。未使用の成果物を使用するアプリケーションをアンインストールした後で、データベースからその成果物を手動で削除します。リポジトリのデータベース・プラットフォームで提供されるツールを使用して成果物を除去します。この理由として、ビジネス・ルールとセクターに含まれているビジネス・ロジックが、アプリケーションのインストール時に更新されている可能性があること、またアプリケーションが除去された時点でこの重要なビジネス・データを削除せずにおく必要があることがあります。

### **始める前に**

除去されるビジネス・ルールまたはセクターを使用するアプリケーションのコピーをすべてアンインストールします。ビジネス・ルールまたはセクターの成果物

を削除する前に、それらのバックアップを作成してください。これを行うには、管理コンソールまたは `wsadmin` コマンドを使用して、ビジネス・ルールまたはセクターの成果物をサーバーからエクスポートします。

## このタスクについて

ビジネス・ルールまたはセクターの成果物を含むアプリケーションをインストールする場合、サーバーはこれらの成果物をデータベース表に格納するため、アプリケーションを変更しなくてもその成果物を動的に更新できます。また、これによって、他のサーバーはその成果物を共有できます。アプリケーションをアンインストールする場合、サーバーはデータベース表からその成果物を自動的に除去しません。アプリケーションがまだ別のサーバーにインストールされていて、実行されている可能性があるためです。その場合、データベースから成果物を削除してしまうと、アプリケーションの別の実行中のコピーがビジネス・ルールまたはセクターの使用を試行したときにそれらのアプリケーションは失敗します。

不要なビジネス・ルールおよびセクターの成果物をリポジトリから除去するには、以下のステップを実行します。

### 手順

1. 行を削除する以下のデータベース表のロケーションを探します。

#### **BYTESTORE**

ビジネス・ルールおよびセクターの成果物を含むメイン・テーブル

#### **BYTESTOREOVERFLOW**

メイン・テーブル用のオーバーフロー・テーブル

#### **APPTIMESTAMP**

ビジネス・ルールおよびセクターの成果物を含むインストール済みアプリケーションのタイム・スタンプを保持するテーブル

#### **CUSTPROPERTIES**

ビジネス・ルール・グループ、ルール・セット、またはデシジョン・テーブルに対するユーザー定義のカスタム・プロパティ、およびシステム・プロパティを保持するテーブル

2. ご使用のデータベース・プラットフォーム用のツールを使用して、所定のアプリケーション用のすべてのビジネス・ルールおよびセクターの成果物を削除するには、以下のステップを実行します。
  - a. **BYTESTORE** テーブルで、**APPNAME** 列の値がアプリケーション名と同じ行をすべて検索します。
  - b. 検索されたすべての行の基本キー列の値を記録します。**BYTESTORE** テーブルの基本キー列は、**ARTIFACTNS**、**ARTIFACTNAME**、および **ARTIFACTTYPE** です。
  - c. **BYTESTORE** テーブルから、ステップ 2a で検索された行を削除します。
  - d. ステップ 2b で記録した基本キー値の各セットに対して、対応する列に同じ値を持つ **BYTESTOREOVERFLOW** テーブルの行を探します。

注: 基本キー値の各セットに対して、**BYTESTOREOVERFLOW** テーブルに対応する行がない場合、1 行、または複数行に対応する場合があります。

- e. BYTESTOREOVERFLOW テーブルから、ステップ 2d で検索された行を削除します。
- f. ステップ 2b で記録した基本キー値の各セットに対して、対応する列に同じ値を持つ CUSTPROPERTIES テーブルの行を探します。
- g. CUSTPROPERTIES テーブルから、ステップ 2f で検索された行を削除します。
- h. **APPNAME** 列がアプリケーションの名前と同じである APPTIMESTAMP テーブルの行を削除します。

## タスクの結果

データベース表からビジネス・ルールおよびセレクターの不要な成果物を除去しました。

## ビジネス・ルールの概要

ビジネス・ルールを使用して、ビジネス・プラクティスの振る舞いを制御します。

### ビジネス・ルールとは

ビジネス・プラクティスに構造を与えたり、ビジネス・プラクティスの振る舞いを制御したりするものはすべてビジネス・ルールです。ルールは、ビジネス・ポリシーの適用、組織内での共通のガイドラインの確立、ビジネス環境でのアクセスの制御を実行できます。

### ビジネス・ルールを使用する場合

ビジネス・ルールは、頻繁に変わるビジネス・プラクティス (ビジネスに伴って当然生じる場合もあれば、監督官庁などの外部から要求される場合もある) に対応するために使用します。ビジネス・ルールの標準的な使用法は次のとおりです。

- 現在の金利を確認する
- 製品の割引を計算する
- 消費税を計算する
- 高齢者や得意先など特別なグループを判別する

### ビジネス・ルールの使用法

ビジネス・ルールは、WebSphere Integration Developer の Eclipse ベースのビジネス・ルール・エディターを使用して開発およびデプロイします。ビジネス・ルールの値は、WebSphere Process Server のオプションである Web ベースのビジネス・ルール・マネージャーを使用して管理および変更します。これらのツールについて詳しくは、それぞれ WebSphere Integration Developer インフォメーション・センターおよび WebSphere Process Server インフォメーション・センターの該当トピックを参照してください。

### ビジネス・ルール・コンポーネントの表示

ビジネス・ルール・グループを管理する際には、まず、ビジネス・ルール・コンポーネントを表示します。その表示画面から、ビジネス・ルール・グループの一部ま

たはすべてをエクスポートまたはインポートしたり、ビジネス・ルール・グループを構成するテーブルを表示したりすることができます。

## 始める前に

このタスクを実行するには、ユーザーは WebSphere Process Server の管理コンソールを使用していなければなりません。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

## このタスクについて

サーバーに存在するビジネス・ルール・グループを確認するには、以下のステップを実行します。

### 手順

1. 管理コンソールから「サーバー」>「アプリケーション・サーバー」をクリックします。
2. サーバー名 をクリックして、ビジネス・ルールを表示するサーバー・リストからサーバーを選択します。
3. 「Business Integration」の下の「ビジネス・ルール」をクリックします。

## タスクの結果

コンソールに、各グループの説明を使用して定義されたすべてのビジネス・ルール・コンポーネントのリストが表示されます。

### 管理コンソールを使用したビジネス・ルールのエクスポート:

ビジネス・ルール・テーブルを変更したら、ビジネス・ルール・コンポーネントをエクスポートします。これにより、開発環境にインポートできるファイルが作成され、開発の成果物を実際の実動システムの成果物と同期させることができます。

## 始める前に

このタスクを開始するには、『ビジネス・ルール・コンポーネントの表示』の説明に従って、ビジネス・ルール・コンポーネントを事前に表示しておく必要があります。「サーバー」>「アプリケーション・サーバー」> *servername* >「ビジネス・ルール」>「ビジネス・ルール」をクリックします。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。セキュリティが有効でないときは、ユーザー ID を指定して管理コンソールにログインする必要があります。

## このタスクについて

管理コンソールを使用してビジネス・ルールをエクスポートするには、以下のステップを実行します。

**ヒント:** コマンド行を使用してビジネス・ルールをエクスポートすることもできます。『`exportBusinessRuleArtifacts.jacl` コマンド』を参照してください。

### 手順

- 1 つ以上のビジネス・ルール・グループの隣にあるチェック・ボックスを選択して、「エクスポート」をクリックします。

ブラウザーには、選択したビジネス・ルール・グループへの HTML リンクのリストが表示されます。これは、「ビジネス・ルールのエクスポート」ページです。各ビジネス・ルール・グループには、ファイル拡張子 `.zip` が付加されています。

2. 各ファイル名をクリックして、ファイルをシステムにダウンロードします。ファイルを保存するためのプロンプトがシステムから出されたら、「OK」をクリックします。

**注:** ファイルの保存を選択した場合は、ファイルのダウンロード時にファイル名を変更できます。

3. 「戻る」をクリックして、ビジネス・ルール・グループのリストに戻ります。

### タスクの結果

指定した場所にファイルが保管されます。その後、そのファイルをテスト・システムにコピーできます。

### 次のタスク

ファイルは WebSphere Integration Developer 環境にインポートする必要があります。詳しくは、WebSphere Integration Developer インフォメーション・センターを参照してください。

### 管理コンソールを使用したビジネス・ルールのインポート:

インストールしたビジネス・ルールを、アプリケーションを再インストールせずに更新するには、ビジネス・ルールをインポートします。

### 始める前に

管理コンソールで作業する必要があり、エクスポート機能によって作成された圧縮ファイルのロケーションを知っている必要があります。

ビジネス・ルールをインポートする前に、以下の条件が満たされていることを確認します (条件が満たされていない場合は、インポートが失敗します)。

- ファイルの拡張子が `zip` である。
- サーバーからビジネス・ルールをエクスポートして圧縮ファイルが作成されている。
- ビジネス・ルール・グループを使用するアプリケーションがセル内のサーバーに既にインストールされている。

**このタスクに必要なセキュリティー・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

## このタスクについて

インストール済みのアプリケーションが使用しているビジネス・ルールを変更した後で、それらの変更内容を別のクラスターまたはサーバーにも適用したい場合に、ビジネス・ルールをインポートします。ご使用の開発環境と実稼働環境内の変更内容を同期させる場合にも、この機能を使用できます。

管理コンソールを使用してビジネス・ルールをインポートするには、以下のステップを実行します。

**ヒント:** ビジネス・ルールは、コマンド行を使用してインポートすることもできます。『importBusinessRuleArtifacts.jacl コマンド』を参照してください。

### 手順

1. ビジネス・ルールのインポート先のサーバー上のビジネス・ルールを表示します。「サーバー」>「アプリケーション・サーバー」> *servername* >「ビジネス・ルール」>「ビジネス・ルール」をクリックします。
2. 「インポート」をクリックします。
3. 「ビジネス・ルールのインポートの準備」 ページ上でファイルへのパスを指定します。

### 次のタスク

ビジネス・ルールを表示して、変更されたルールを確認します。

## ビジネス・ルール・マネージャー

ビジネス・ルール・マネージャーは、ビジネス・アナリストがビジネス・ルール値を表示および変更する際に使用する Web ベースのツールです。このツールは、WebSphere Process Server のオプションであり、プロファイル作成時またはサーバーのインストール後に選択してインストールできます。

ビジネス・ルールは、WebSphere Integration Developer 内で、if/then ルール・セットおよびデシジョン・テーブルを使用して操作を実装することにより設計および開発します。ビジネス・ルールは、WebSphere Business Modeler 内で作成することもできます。ただし、Modeler では、ビジネス・ルール・タスクの作成のみがサポートされます (ビジネス・ルール・タスクは、Modeler からエクスポートしたときにルール・セットになります)。ルール・セットおよびデシジョン・テーブルはテンプレートに設定します。テンプレートは、変更可能なビジネス・ルールの側面と変更の程度を制御します。それらによって、デシジョン・テーブルの if/then ルール、条件ケース、およびアクションの構造を定義します。

テンプレートは、ビジネス・ルール・マネージャーでのビジネス・ルール・ランタイム・オーサリングのメカニズムを提供します。テンプレートを使用すると、ビジネス・ルールの値を変更したり、ルール・セット内に新しいルールを作成したり、デシジョン・テーブル内に新しい条件またはアクションを作成したり、実行時にビジネス・ルール定義に対する変更を公開したりできます。

ビジネス・ルールはビジネス・ルール・グループとして編成されます。ビジネス・ルール・グループは、ルールに対するインターフェースとして、ルールを呼び出すために使用されます。ルール・セットおよびデシジョン・テーブルが直接呼び出されることはありません。

ビジネス・ルールの構築およびデプロイについての詳細は、WebSphere Integration Developer インフォメーション・センターを参照してください。

### ビジネス・ルール・マネージャーが機能する仕組み

ビジネス・ルール・マネージャーは、ビジネス・アナリストがランタイム・ルール・オーサリングに使用する主要な WebSphere Process Server ツールです。

ビジネス・ルール・マネージャーを使用して、以下のタスクを実行します。

- リポジトリからビジネス・ルールのコピーを検索する
- ビジネス・ルールを参照および編集する
- リポジトリへビジネス・ルールを公開する

以下の図は、ビジネス・ルール・マネージャーがルールを呼び出し、公開する仕組みを示しています。

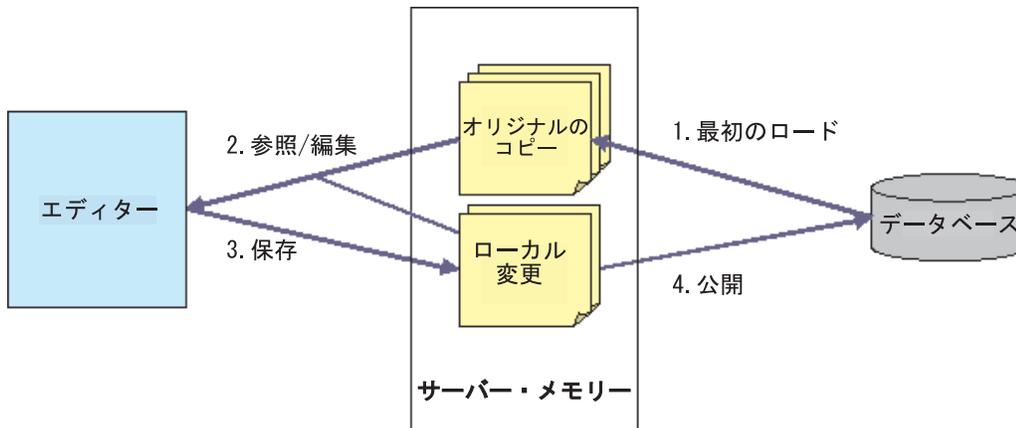


図3. ビジネス・ルール・マネージャーのイベント・シーケンス

ビジネス・ルール・マネージャーにログオンした後に、ビジネス・ルールを変更すると、以下のイベントが発生します。

1. ビジネス・ルールを選択すると、ビジネス・ルール・マネージャーは、リポジトリからビジネス・ルール・グループにアクセスして、それをオリジナルのコピーとしてサーバー・メモリーに保管します。
2. ビジネス・ルール・グループとルール・ロジックが編集できるようになります。
3. ルール・セット、デシジョン・テーブル、およびビジネス・ルール・グループへの変更をコピーとしてサーバー・メモリーに保存できます。
4. ローカル・コピーを再びデータ・ソースに公開します。あるいは、更新を実行せずに、変更を取り消すことも可能です。

## ビジネス・ルール・マネージャーへのアクセス

ビジネス・ルール・マネージャーには、Web ブラウザーを使用してアクセスします。

### 始める前に

サーバーとクライアントが両方とも正しく構成されていることを確認してください。

### このタスクについて

ビジネス・ルール・マネージャーにアクセスできるデフォルトの URL は、以下のとおりです。 URL は環境によって異なる場合があります。

`http://hostname:port/br`

「hostname」は現行ホスト・システムの名前（または IP アドレス）、「port」はアプリケーションがインストールされたアプリケーション・サーバーのポートです。

例えばサーバーが 1 つのみのスタンドアロン環境では、このリンクは次のようになります。

`http://hostname:9080/br`

**注:** 管理セキュリティが有効になっている場合、前述のリンクが自動的にセキュア・リンクに切り替わります。例えばサーバーが 1 つのみのスタンドアロン環境では、`https://hostname:9443/br` です。

管理セキュリティが使用可能でない場合は、「ビジネス・ルール・グループ」ページが開きます。管理セキュリティがサーバー上で使用可能な場合は、「ログイン」ページが開きます。

管理セキュリティが使用可能な場合、ログインするには、以下のステップを実行します。

### 手順

1. 「ログイン」ページで、「ユーザー ID」を入力します。
2. 「パスワード」を入力します。
3. 「ログイン」をクリックします。

### タスクの結果

ビジネス・ルール・マネージャーの最初のページが開きます。ナビゲーション・ペインには、既存のビジネス・ルール・グループがリストされています。

### 次のタスク

ビジネス・ルール操作をブラウズおよび編集し、ビジネス・ルールをテンプレート化できます。

## 「ビジネス・ルール・グループ」ページおよび「ビジネス・ルール・マネージャー」ページのレイアウト

ビジネス・ルール・マネージャーが開くと、「ビジネス・ルール・グループ」ページが表示されます。このページでは、ビジネス・ルール・グループとその定義済みの操作をすべて確認できます。

「ビジネス・ルール・グループ」ページはナビゲーションの第 1 レベルです。このページのレイアウトには、他のビジネス・ルール・マネージャー・ページにも共通する多くのエレメントが含まれます。

### ツールバー

ツールバーのコンポーネントは以下のとおりです。

#### ようこそ

現在ログオンしているユーザーの名前が表示されます。

#### ユーザー識別

「ウェルカム・ユーザー名 (Welcome User Name)」の後に現在のユーザーの名前が表示されます。

#### ログアウト

管理セキュリティーが使用可能である場合に、「ログイン」ページを開きます。

**重要:** 公開せずにログアウトしようとする、確認を求めるダイアログ・ボックスが表示されます。

**検索** 「ビジネス・ルール・グループの検索」ページを開きます。指定した一連の操作対象ビジネス・ルール・グループを迅速に検索したり、その絞り込みを行うことができます。

**ヘルプ** WebSphere Process Server インフォメーション・センターのビジネス・ルールに関するトピックにアクセスできます。

### ナビゲーション・ペイン

ナビゲーション・ペインは、左側にあるペインです。「公開および復帰」ページおよび使用可能なビジネス・ルール・グループへのアクセスを提供します。このナビゲーション・ツリーを使用して、必要なルール・レベルにドリルダウンできます。

**注:** ナビゲーション・ペインは、編集モードのページには表示されません。

**重要:** バージョン番号が現行モデルよりも大きいビジネス・ルール成果物を取得する場合は、シェルと呼ばれるビジネス・ルール成果物が、ナビゲーション・ペインでフラット・テキスト項目になります。その結果、シェルをさらに公開できなくなります。現在の WebSphere Process Server を最新版 (シェルのバージョンと同じかまたはそれより高いバージョン) に更新してください。

#### 公開および復帰

「公開および復帰」ページを開きます。このページでは、ビジネス・ルール・グループおよびルール・スケジュールの変更をデータベースに公開したり、ビジネス・ルール・グループまたはルール・スケジュールをデータベース上にあった元のコピーに戻したりすることができます。

## ビジネス・ルール・グループ

「ビジネス・ルール・グループ」ページを開きます。このページはブラウザのトップレベルです。ビジネス・ルール・グループはナビゲーション・ツリーにリストされます。表示名の隣にある正符号 (+) または負符号 (-) をクリックすると、ビジネス・ルール・グループを展開または縮小できます。これにより、関連するルールをすべて表示することが可能です。左のペインのナビゲーション・ツリーでビジネス・ルール・グループを選択すると、右のペインにすべての子「ルール・スケジュール」ページ (ビジネス・ルール操作) と関連するすべてのルール・セットやデシジョン・テーブルが表示されます。これらをクリックすると、対応する編集用ページが開きます。

## コンテンツ領域

コンテンツ領域は、右側にあるペインであり、これが主に表示および編集を行う領域です。コンテンツ領域には「タイトル」セクション、「一般情報」セクション、およびページ固有のセクションがあります。

**注:** コンテンツ領域に表示される情報は、「ビジネス・ルール・グループ」ページ、「ルール・スケジュール」ページ、「ルール・セット」ページ、「デシジョン・テーブル」ページ、「公開および復帰」ページ、および「ビジネス・ルール・グループの検索」ページのいずれを表示しているかによって異なります。

### 「タイトル」セクション

「タイトル」セクションには以下の情報があります。

#### パス情報

ページに対するパス (ビジネス・ルール・グループおよび「ルール・スケジュール」ページの名前など) を以下の形式で表示します。

`BusinessRuleGroup01 > Table1_operation1`

**例:** `CalculateDiscountBRG > CalculateDiscount`

#### ルール・タイトル

リソース表示名およびビジネス・ルールのタイプを以下の形式で指定します。

`Ruleset112 - Ruleset`

**例:** `calculateDiscount-Rule Schedule, CalculateDiscountRS - Rule Set`

#### 機能ボタン

個々のページの目的に応じて各種のアクションを使用可能にします。すべての機能ボタンがページで使用できるわけではありません。ボタンによっては、コンテンツ領域の別のセクションに表示されるものもあります。ページで使用可能な機能ボタンを以下の表にリストします。

表 10. 機能ボタン

ボタン名	機能
プロパティの追加	「ビジネス・ルール・グループ」ページでビジネス・ルール・グループにプロパティを追加するか、または「ビジネス・ルール・グループの検索」ページで検索照会を作成します。
戻る	前のページに戻ります。
キャンセル	リソースへの変更を破棄して前のページに戻ります。
コピー	新規デシジョン・テーブルまたはルール・セットを作成するため、デシジョン・テーブルまたはルール・セットをコピーします。デシジョン・テーブルまたはルール・セットを新規に作成するには、既存のデシジョン・テーブルまたはルール・セットをコピーしてから、値を変更する必要があります。
編集	ビジネス・ルール・グループ、ルール・スケジュール、ルール・セット、またはデシジョン・テーブルが編集できるようになります。
公開	ビジネス・ルール・グループまたはルール・スケジュールをリポジトリに公開します。
復帰	ローカル環境に保存されたルールに対する変更をすべて取り消し、サーバー・メモリーに存在する元のコピーにルールを戻します。公開後にルールを復帰することはできません。
保管	変更内容を検証してからローカル・コピーに保存し、前のページに戻ります。稼働中のサーバーの状態は変更されていないことに注意してください。サーバーの状態を変更する方法については、「公開」を参照してください。
検索	「ビジネス・ルール・グループの検索」ページの検索照会を開始し、照会と一致するビジネス・ルール・グループを同じ「ビジネス・ルール・グループの検索」ページに検索結果として戻します。
ソート	ビジネス・ルール・グループのプロパティを、プロパティ名を基準にして、アルファベットの昇順でソートします。

#### メッセージ・フィールド

ルールに対してとられたアクションの状況またはエラーが発生したことを表示します。状況メッセージの例を以下に示します。

"calculateDiscount" が一時的に保管されました。

「公開および復帰」ページから変更を公開できます。

## 「一般情報」セクション

「一般情報」セクションには以下の情報があります。

**注:** WebSphere Process Server 6.1 以降では、「ビジネス・ルール・グループ」ページに「一般情報」セクションが含まれます。「ビジネス・ルール・グループの検索」ページと「公開および復帰」ページには、このセクションは含まれません。

**表示名** Websphere Process Server 6.1 以降のビジネス・ルール・グループ、ルール・セット、またはデシジョン・テーブルの表示名を示します。「ビジネス・ルール・グループ」、「ルール・セット」、および「デシジョン・テーブル」ページにおいて、表示名は、表示モードでは読み取り専用ですが、編集モードでは変更することが可能です。表示名には、どのストリング値でも使用でき、特殊文字を含めることもできます。同じタイプのビジネス・ルール成果物の表示名を固有にする必要はありません。ただし、ユース・ケースでは、ビジネス・ルール成果物の名前を固有にする必要があります。

表示名を設定した場合、その表示名は、名前の値が使用されるすべての場所(ナビゲーション・ペインを含む)で、名前の値の代わりに使用されます。また、成果物が詳しく表示されるときにも、その表示名が使用されます。ビジネス・ルール成果物の表示名を設定しない場合は、その名前の値が代わりに使用されます。「名前との同期」チェック・ボックスを選択すると、ターゲットのビジネス・ルール・グループ、ルール・セット、またはデシジョン・テーブルの対応する名前の値に表示名が同期されます。編集ページで行った変更を保存すると、ビジネス・ルール・マネージャーのすべてのページで新しい名前が有効になります。

### 最後に公開

ビジネス・ルール・グループ、ルール・スケジュール、ルール・セット、またはデシジョン・テーブルの最終公開日を表示します。

**状況** ルール・スケジュール、ルール・セット、またはデシジョン・テーブルが編集モードにあるかまたは公開されたかを表示します。

**説明** ビジネス・ルール・グループ、ルール・スケジュール、ルール・セット、またはデシジョン・テーブルの要旨を表示します。これらのページの編集モードで、説明を編集できます。

**制約事項:** ビジネス・ルール・マネージャーでビジネス・ルール・グループ・コンポーネントおよびビジネス・ルールの説明フィールドを編集するときには、CDATA タグを使用しないでください。使用すると、ビジネス・ルール・グループおよびビジネス・ルールが編集不能になります。CDATA タグが存在する場合は、XML エディターでビジネス・ルール・グループまたはビジネス・ルールを開き、説明フィールドから CDATA タグを手動で除去します。

## ページ固有情報のセクション

ページ固有情報セクションの内容は、「ビジネス・ルール・グループ」ページ、「ルール・スケジュール」ページ、「ルール・セット」ページ、「デシジョン・テーブル」ページのいずれを表示しているかによって異なります。これらの各ページに関する具体的な説明については、個々のトピックを参照してください。

「ビジネス・ルール・グループ」ページの場合、このセクションには以下の情報が表示されます。

#### ビジネス・ルール・リソース

ルール・スケジュール、ルール・セット、およびデシジョン・テーブルの表示名をリストします。

**説明** リソースの要旨または名前を指定します。

#### アクション

対応するビジネス・ルール・リソースについて選択可能なアクションを表示します。最初は空になっていますが、ビジネス・ルール・グループを展開すると、それぞれのルールの横に「編集」ボタンが表示されます。

#### 「公開および復帰」ページ:

「公開および復帰」ページは、ローカル環境に保存されているビジネス・ルール・グループおよびルール・スケジュールに対する変更内容をリポジトリに公開するために使用します。また、ビジネス・ルール・リソースをローカル環境に保管する前にサーバー・メモリーにあった元のコピーにビジネス・ルール・グループおよびルール・スケジュールを戻すためにも使用します。

コンテンツ領域のページ固有情報セクションには、以下のエレメントがあります。

#### 「変更されたビジネス・ルール・リソース」セクション

このセクションでは、公開または復帰が可能なビジネス・ルール・グループおよびルール・スケジュールのリストを表示し、以下の情報を提供します。

#### ビジネス・ルール・リソース

変更したビジネス・ルール・グループおよびルール・スケジュールの名前を表示します。公開する準備が整っているリソースの隣には、公開を選択または選択解除するためのチェック・ボックスがあります。

**状況** リソースが元の状態であるかローカル環境で変更されたかを示します。

**説明** リソースの要旨を提供します。

#### アクション

復帰可能なリソースを示します。リソースには、対応する「アクション」フィールドに「復帰」ボタンがあります。

#### 「ビジネス・ルール・グループ」ページ:

「ビジネス・ルール・グループ」ページには、ビジネス・ルール・グループに関連するビジネス・ルール・リソースがすべて表示されます。

このページでは、ビジネス・ルール・グループの情報や関連するビジネス・ルール・リソースの情報を参照するだけでなく、編集ページを開いて情報を変更することもできます (ビジネス・ルール・グループのカスタム・プロパティの追加、削除、変更も含まれます)。

コンテンツ領域のページ固有情報セクションには、以下のエレメントがあります。

## 「プロパティ」セクション

このセクションでは、ビジネス・ルール・グループのカスタム定義プロパティを表示します。

**制約事項:** ビジネス・ルール・グループにカスタム・プロパティがない場合、またはそのカスタム・プロパティのリストが空の場合、表示モードでは「プロパティ」セクションは表示されません。また、ビジネス・ルール・グループが WebSphere Process Server 6.1 より前のバージョンに属する場合、「ビジネス・ルール・グループ」ページには、ビジネス・ルール・グループの「プロパティ」セクションと「編集」ボタンは表示されません。

**名前** プロパティの名前を指定します。この名前は固有である必要があります。また、空にすることはできません。各プロパティは、ビジネス・ルール・グループで 1 回のみ定義できます。

**値** プロパティの値を指定します。プロパティごとに定義された値が必要です。空ストリングまたは長さゼロにすることはできますが、ヌルであってはなりません。プロパティをヌルに設定することは、プロパティを削除することと同じです。

## 「ビジネス・ルール・リソース」セクション

このセクションでは、ビジネス・ルール・グループに関連するルール・スケジュール、ルール・セット、およびデシジョン・テーブルのリストを表示します。

### ビジネス・ルール・リソース

ビジネス・ルール・グループに関連するルール・スケジュール、ルール・セット、およびデシジョン・テーブルの表示名をリストします。

**説明** ビジネス・ルール・グループ、ルール・スケジュール、ルール・セット、およびデシジョン・テーブルの要旨または名前を表示します。

### アクション

対応するビジネス・ルールのリストについて選択可能なアクションを表示します。最初は空になっていますが、グループを展開すると、それぞれのルールの横に「編集」ボタンが表示されます。

### 「ルール・スケジュール」ページ:

「ルール・スケジュール」ページには、スケジュール済みルール・ロジック項目内のビジネス・ルール・グループの値を変更するためのインターフェースがあります。情報は表形式で表示されます。

「ルール・スケジュール」ページからは、ビジネス・ルールの有効な日付の表示、変更、追加、分割、または削除などの操作を実行できます。また、既存のビジネス・ルールをコピーすることにより、新しいビジネス・ルールを作成することもできます。

コンテンツ領域のページ固有情報セクションには、以下のエレメントがあります。

## 「スケジュール済みルール・ロジック」セクション

このセクションでは、ルールを構築するための基本単位である有効なビジネス・ルールのリストを表示します。ここでは、スケジュール済みルール・ロジック項目の追加やソートなどの操作が可能です。

**注:** ビジネス・ルール・マネージャーのルール・ロジック選択日時の値は、現地時間 (Web ブラウザーを実行しているクライアントの時間帯を使用します)、または協定世界時 (UTC) のいずれかで指定できます。

### 開始日/時刻

特定の日付または「開始日無し」のいずれかを選択できます。

**注:** 「開始日無し」は、ターゲット・ルール・ロジックが終了日より前のすべての日に対して有効であることを意味します。

### 終了日/時刻

特定の日付または「終了日なし」のいずれかを選択できます。

**注:** 「終了日なし」は、ルール・ロジックが開始日およびそれ以降のすべての日に対して有効であることを意味します。

### 有効なルール・ロジック

対応する時間フレーム内で有効なルール・セットまたはデシジョン・テーブルを指定します。

### アクション

スケジュール済みルール・ロジック項目を分割および削除するためのオプションを提供します。

### デフォルト・ルール・ロジック

他のルール・ロジックを適用できない場合のデフォルトのルール・ロジックを提供します。これは、他のスケジュール済みルール・ロジック項目のいずれとも日付が一致しない場合に選択されます。

## 「使用可能なルール・ロジック」セクション

このセクションでは、特定のビジネス・ルールに適用可能なルール・セットまたはデシジョン・テーブルのリストを、関連する記述やアクションとともに表示します。

### ルール・ロジック

ルール・セットまたはデシジョン・テーブルの名前を指定します。

**説明** ルール・セットまたはデシジョン・テーブルの要旨を表示します。

### アクション

ルールを容易に編集またはコピーするオプションを提供します。

### 「ルール・セット」ページ:

「ルール・セット」ページには、ビジネス・ルールのルール「インスタンス」とそれらの実行順序、およびそのルール・セットに関連付けられているテンプレートがリストされます。

「ルール・セット」ページからは、テンプレートを使用した既存のルール・インスタンスの表示/編集、選択したテンプレートからの新規ルール・インスタンスの作成、ルール実行順序の指定、ルールまたはルール・セットの名前変更、ルール・セットの表示名またはルール・セットのルールの表示/編集、ルール・セット、ルールの説明、テンプレート・パラメーターの説明の表示/編集、ルール・セットの作業用コピーとしての保存、ルールの削除が可能です。

コンテンツ領域のページ固有情報セクションには、以下のエレメントがあります。

### 「ルール」セクション

このセクションでは、関連するルールのリストを表示し、以下の情報を提供します。

**名前** ルールの名前を提供します。このフィールドは、編集モードでのみ表示されます。

**表示名** ルールの表示名を指定します。表示名を指定しなかった場合は、「名前」の値が設定されます。表示モードでは読み取り専用ですが、編集モードでは編集が可能です。表示名には、どのストリング値でも使用でき、特殊文字を含めることもできます。固有である必要はありません。「アクション」フィールドの「名前の同期」チェック・ボックスを選択すると、表示名が、対応する名前と同期されます。

**ルール** ルールを定義する変数、制約、範囲、および列挙をリストします。

**説明** ルール・セット内の各ルールの情報を指定します。表示モードでは読み取り専用ですが、編集モードでは編集が可能です。

### アクション

関連するボタンをクリックすることにより、ルールの再配列、ルールの削除、および表示名と名前の同期が可能です。このアクションは、編集モードでのみ使用可能です。

### 「テンプレート」セクション

このセクションでは、既存のテンプレートを使用して、編集モードで新規ルールを容易に作成できます。また、ルールに対して以下の情報を指定するためのフィールドがあります。

#### テンプレート名

既存のテンプレートの名前を提供します。

**名前** ルールの名前を入力および変更するためテキスト域を提供します。

**表示名** ルールの表示名を入力するためテキスト域を提供します。表示名を指定しない場合は、「名前」の値が設定されます。表示名には、どのストリング値でも使用でき、特殊文字を含めることもできます。固有である必要はありません。「名前の同期」チェック・ボックスを選択すると、表示名が、ルール名の値と同期されます。編集ページで行った変更を保存すると、ビジネス・ルール・マネージャーのすべてのページで新しい名前が有効になります。

**注:** 「名前の同期」チェック・ボックスを選択すると、ルールの表示名が使用不可になり、変更できなくなります。

**ルール** ルールを定義する変数、制約、範囲、および列挙を指定するためテキスト域を提供します。

**説明** 各テンプレート・パラメーターのルールの情報を指定します。これが表示されるのは、ルール・セットが編集モードであり、マウスをターゲット・テンプレート・パラメーターの上に移動したときに限られます。読み取り専用です。

#### **アクション**

テンプレートへのルールの追加、テンプレートからのルールの削除、またはルール名の値と表示名との同期が可能です。

#### **「デシジョン・テーブル」 ページ:**

「デシジョン・テーブル」 ページには、条件ケースとアクション、その方向 (行と列)、およびデシジョン・テーブルに関連付けられたテンプレートがあります。「デシジョン・テーブル」 ページは「ルール・スケジュール」 ページから開きます。

「デシジョン・テーブル」 ページでは、テンプレートを使用した既存の条件またはアクションの表示または編集、そのデシジョン・テーブルに定義されたテンプレートを使用した新規の条件の追加、条件の削除、条件の順序の変更、方向の変更、関連付けられているテンプレートを使用した初期設定アクションのルールの変更、デシジョン・テーブルと初期設定のルールの表示名およびその説明の表示と編集、デシジョン・テーブルの保存 (作業用コピーとして) が可能です。

コンテンツ領域のページ固有情報セクションには、以下のエレメントがあります。

#### **「初期設定のルール」 セクション**

このセクションでは、デシジョン・テーブルの初期設定のルールが示されます。ビジネス・ルール定義が WebSphere Integration Developer 内で初期設定アクションを指定して設計されている場合にのみ、初期設定のルールが表示されます。初期設定のルールは、デシジョン・テーブル・ロジックが発行される前に直接呼び出されます。初期設定のルールを使用して、デシジョン・テーブルで使用される変数やアクションを初期化できます。編集モードでは、以下の情報を変更するためのフィールドが存在します。

**名前** 初期設定のルールの名前を指定します。

**表示名** ルールの表示名を指定します。表示名を指定しなかった場合は、「名前」の値が設定されます。表示名には、任意のストリング値を使用でき、特殊文字を含めることもできます。また、固有名である必要はありません。「アクション」フィールドの「名前の同期」チェック・ボックスを選択すると、表示名が、対応する名前と同期されます。編集ページで行った変更を保存すると、新しい名前が有効になります。

**注:** 「名前の同期」チェック・ボックスを選択すると、ルールの表示名が使用不可になり、変更できなくなります。

**ルール** 初期設定のルールを定義する変数、制約、範囲、および列挙をリストします。

**説明** 各初期設定のルールの情報を指定します。表示モードでは読み取り専用ですが、デシジョン・テーブルの編集モードでは編集が可能です。

## アクション

「**名前の同期**」チェック・ボックスを選択することにより、表示名と名前を同期させることができます。

## 「デシジョン・テーブル」セクション

このセクションでは、テーブルの条件ケース（行見出しおよび列見出しとして表されます）とアクション（それらの条件ケースの交点として表されます）が示されます。「**方向**」アイコンを使用して、条件行の方向を水平方向から垂直方向、またはその逆に切り替えることができます。

## その他の場合

このデシジョン・テーブルの「**その他の場合**」条件を示します。「**その他の場合**」条件は、デシジョン・テーブル内のほかの条件が適用されない場合にデフォルトで入力される特別な条件です。「**その他の場合**」条件は、WebSphere Integration Developer で設計されたデシジョン・テーブル定義で指定された場合にのみ表示されます。ビジネス・ルール・マネージャーで、デシジョン・テーブルの「**その他の場合**」条件の列を動的に追加または除去することはできません。

## 「テンプレート」セクション

このセクションでは、既存のテンプレートを使用して新規ルールを容易に追加できます。

## 「ビジネス・ルール・グループの検索」ページ:

「**ビジネス・ルール・グループの検索**」ページでは、検索照会を作成することにより、指定した一連の操作対象ビジネス・ルール・グループを検索したり、その絞り込みを行ったりすることができます。「**ビジネス・ルール・グループの検索**」ページを開くには、ビジネス・ルール・マネージャーの上部にあるツールバーの「**検索**」をクリックします。

「**ビジネス・ルール・グループの検索**」ページでは、ターゲット名前空間、ビジネス・ルール・グループ名、カスタム・プロパティ、またはこれらの任意の組み合わせを基準にして検索を実行できます。また、1 つ以上のカスタム・プロパティを追加したり、名前を基準にしてアルファベットの昇順でカスタム・プロパティをソートしたり、プロパティ・テーブル内でプロパティを上下に移動したり、カスタム・プロパティを削除したりできます。

「**ビジネス・ルール・グループの検索**」ページのコンテンツ領域には、「**メッセージ**」フィールドとページ固有の情報セクションがあり、以下のエレメントが含まれています。

## 「データの検索」セクション

このセクションには、以下のエレメントが含まれています。

**名前** 検索対象のビジネス・ルール・グループの名前を入力するためテキスト域を提供します。この値を空のままにすると、ビジネス・ルール・グループは検索コンテキストに含まれなくなります。入力した値は、名前と表示名の両方として使用されます。したがって、検索を実行すると、入力した名前の値と

一致する名前または表示名を持つビジネス・ルール・グループが検索されます。名前または表示名のいずれか一方で具体的に検索する場合はプロパティ名を使用して、そのような検索を指定する必要があります。

**例:** プロパティの名前として `IBMSysName`、そのプロパティの値として `VIPGroup` を入力した場合、ビジネス・ルール・マネージャーは、`VIPGroup` に一致する名前 (表示名ではない) を持つビジネス・ルール・グループを検索します。

#### ターゲット名前空間

ビジネス・ルール・グループの URL を入力するためテキスト域を提供します。この値を空のままにすると、ビジネス・ルール・グループは検索コンテキストに含まれなくなります。

#### 「プロパティ」セクション

このセクションは、「プロパティの追加」をクリックすると開きます。以下のエレメントが表示されます。

#### 論理演算子

「And」、「Or」、または「Not」を選択するためのドロップダウン・リストが表示されるので、複数のプロパティを含む検索照会を作成できます。

**名前** プロパティの名前を入力するためテキスト域を提供します。検索コンテキストのプロパティ・テーブル内では、名前は固有でなければなりません。また、名前を空にすることはできません。

#### 照会演算子

検索データ・フィールドごとに 4 つの照会演算子から選択するためのドロップダウン・リストが表示されます。照会演算子は以下のとおりです。

照会演算子	説明
次と等しい	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと厳密に一致しなければならないことを示します。
次を含む	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと類似するビジネス・ルール・グループを照会で検索することを示します。このストリングにはワイルドカード文字を含めることができます。任意の数の文字を表すワイルドカードを指定する場合はパーセント文字 ('%')、単一文字のワイルドカードを指定する場合は下線文字 ('_') を使用します。これらのワイルドカード文字は、SQL 構文に従う必要があります。
次と等しくない	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと一致してはならないことを示します。

照会演算子	説明
次を含まない	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと類似しないビジネス・ルール・グループを照会で検索することを示します。このストリングには、「次に近い」演算子で定義されたワイルドカード文字を含めることができます。

**値** プロパティ値を入力するためテキスト域を提供します。値は空にすることができます。また、値は検索コンテキストに取り込まれます。

**例:** プロパティ `PayMethod` の値を空のままにして、その照会演算子を「次と等しくない」に設定した場合は、`PayMethod` プロパティの値が空以外のストリングに設定されたビジネス・ルール・グループがすべて検索されます。

#### アクション

プロパティ・テーブル内でプロパティを上下に移動したり、カスタム・プロパティを削除したりできます。

#### 「検索結果」セクション

このセクションには、以下のエレメントが含まれています。

#### ルール・グループ

検索照会で戻されたビジネス・ルール・グループの名前をリストします。

**状況** ランタイムから検索結果として戻されたビジネス・ルール・グループの状況を表示します。以下の 4 種類のいずれかの状況になります。

**ヒント:** 結果のビジネス・ルール・グループをクリックすると、そのビジネス・ルール・グループのページが開きます。

状況	説明
ローカルと同じ	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに既に存在しており、その内容と結果のビジネス・ルール・グループの内容が厳密に同じであることを示します。したがって、検索後にさらにアクションが実行されることはありません。
ランタイムから変更	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに既に存在していることを示します。ただし、別のユーザー・セッションでマスター・コピーが変更されたため、ローカルの内容と結果のビジネス・ルール・グループの内容は異なります。ビジネス・ルール・マネージャーは、ランタイムからの新しい変更を取得するためにローカル・コピーを自動的に更新します。

状況	説明
ローカルで変更	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに既に存在していることを示します。ただし、それが現行ユーザーによって変更されています。ビジネス・ルール・マネージャーは、ユーザーによる以降のすべてのアクションに対してローカル・コピーを使用します。
ローカルで新規	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに存在していないことを示します。この場合、ビジネス・ルール・マネージャーは、結果のビジネス・ルール・グループのローカル・コピーを作成して、それをナビゲーション・ペインに表示します。

**説明** ビジネス・ルール・グループの追加情報を提供します。

## ビジネス・ルール・グループのプロパティの追加、削除、および変更

検索において、ビジネス・ルール・グループのカスタム・プロパティを使用すると、表示および変更したいビジネス・ルール・グループのサブセットを取得できます。新しいカスタム・プロパティを追加したり、既存のプロパティを削除または変更するには、ビジネス・ルール・グループの編集ページを使用します。ビジネス・ルール・グループのカスタム・プロパティの数の制限はありません。

### 始める前に

ビジネス・ルール・グループの編集モードにする必要があります。

**制約事項:** ビジネス・ルール・グループのプロパティ・サポートは、6.1 ビジネス・ルール・グループ以降で使用できます。

### このタスクについて

ビジネス・ルール・グループのプロパティを追加、削除、または変更するには、以下のステップを実行します。

#### 手順

1. 以下のオプションから選択します。

オプション	説明
オプション	ステップ

オプション	説明
プロパティのルールへの追加	<ol style="list-style-type: none"> <li>1. 「プロパティの追加」をクリックします。</li> <li>2. 固有の「名前」を指定します。名前は空にすることはできません。</li> <li>3. 固有の「値」を指定します。各プロパティは、ビジネス・ルール・グループで1回のみ定義できます。各プロパティには、定義済みの値を指定する必要があります。値は、空ストリングまたは長さゼロにすることはできますが、ヌルであってはなりません。プロパティをヌルに設定することは、プロパティを削除することと同じです。</li> </ol>
プロパティの削除	選択したプロパティの「アクション」フィールドで、「削除」をクリックします。
プロパティの変更	対応するフィールドに新しい名前と値を入力します。
プロパティのソート	「ソート」をクリックすると、ビジネス・ルール・グループのプロパティが、プロパティ名を基準にしてアルファベットの昇順でソートされます。

2. 「保管」をクリックします。

## タスクの結果

ビジネス・ルール・マネージャーは、ルールを検証してからプロパティをサーバーに送信します。

## ビジネス・ルール・グループの検索

ビジネス・ルール・グループに対して検索照会を実行すると、指定した一連の操作対象ビジネス・ルール・グループを検索したり、その絞り込みを行ったりすることができます。検索照会は、名前、ターゲット名前空間、カスタム・プロパティ、またはこれらの任意の組み合わせに基づいて作成します。

## 始める前に

「ビジネス・ルール・グループの検索」ページを起動する必要があります。このページを開くには、ビジネス・ルール・マネージャーのツールバーの「検索」をクリックします。

## このタスクについて

検索照会を作成するには、以下のステップを実行します。

### 手順

1. 「名前」フィールドで、検索対象のビジネス・ルール・グループの名前を入力します。この値は、空のままにできます。ただし、検索コンテキストには含まれません。入力した値は、名前と表示名の両方として使用されます。したがっ

て、検索を実行すると、入力した名前の値と一致する名前または表示名を持つビジネス・ルール・グループが検索されます。名前または表示名のいずれか一方で具体的に検索する場合は、プロパティ名を使用して、そのような検索を指定する必要があります。

**例:** プロパティの名前として `IBMSysName`、そのプロパティの値として `VIPGroup` を入力した場合、ビジネス・ルール・マネージャーは、`VIPGroup` に一致する名前 (表示名ではない) を持つビジネス・ルール・グループを検索します。

2. 「ターゲット名前空間」フィールドにビジネス・ルール・グループの URL を入力します。この値は、空のままにできます。ただし、検索コンテキストには含まれません。
3. それぞれの「データの検索」フィールドで、以下の 4 つの照会演算子のいずれかを選択します。

オプション	説明
照会演算子	説明
次と等しい	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと厳密に一致しなければならないことを示します。
次を含む	<p>ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと類似するビジネス・ルール・グループを照会で検索することを示します。このストリングにはワイルドカード文字を含めることができます。任意の数の文字を表すワイルドカードを指定する場合はパーセント文字 (<code>%</code>)、単一文字のワイルドカードを指定する場合は下線文字 (<code>_</code>) を使用します。これらのワイルドカード文字は、SQL 構文に従う必要があります。</p> <p><b>例:</b></p> <ol style="list-style-type: none"> <li>1. ビジネス・ルール・グループ名として「次を含む」「Discount」、ターゲット名前空間として「<code>http://calculateDiscounts</code>」を入力した場合、検索を実行すると、そのストリングを含み、またその URL を持つビジネス・ルール・グループがすべて戻されます。</li> <li>2. ビジネス・ルール・グループ名として「次を含む」「<code>%Discount%</code>」を入力した場合、検索を実行すると、<code>AirlineTicketDiscount</code> や <code>MovieTicketDiscountRules</code> などの名前を持つすべてのビジネス・ルール・グループが戻されます。</li> </ol>

オプション	説明
次と等しくない	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと一致してはならないことを示します。
次を含まない	ビジネス・ルール・グループ名、ターゲット名前空間、またはプロパティの値が、指定したストリングと類似しないビジネス・ルール・グループを照会で検索することを示します。このストリングには、「次に近い」演算子で定義されたワイルドカード文字を含めることができます。

4. **オプション:** 検索コンテキストに必要な数だけプロパティを追加するには、「**プロパティの追加**」をクリックします。
- 「名前」を指定します。検索コンテキストのプロパティ・テーブル内では、名前は固有でなければなりません。また、名前を空にすることはできません。
  - 「照会演算子」を指定します。
  - 「値」を指定します。値は空にすることができます。また、値は検索コンテキストに取り込まれます。

**例:** プロパティ `PayMethod` の値を空のままにして、その照会演算子を「次と等しくない」に設定した場合は、`PayMethod` プロパティの値が空以外のストリングに設定されたビジネス・ルール・グループがすべて検索されます。

- 「アクション」フィールドで上矢印と下矢印をクリックして、プロパティを適切な順番に並べます。

**ヒント:** 「論理演算子」フィールドで「And」、「Or」、または「Not」を使用してプロパティを結合すると、複数のプロパティを含む検索照会を作成できます。

**例:** ターゲット名前空間「`http://calculateDiscounts`」内のビジネス・ルール・グループのうち、`DiscountedItem` プロパティにストリング「`men T-Shirts`」が含まれており、`Ship Handling` プロパティに値「`Free`」が設定されているものをすべて検索するには、論理プロパティ「And」を使用します。

**注:** 「ビジネス・ルール・グループの検索」ページでプロパティを追加、削除、または変更しても、それは検索コンテキスト内でしか適用されません。ビジネス・ルール・マネージャー内のルール・オブジェクトのプロパティには、影響は及びません。

- 「検索」をクリックします。

## タスクの結果

検索照会に一致したビジネス・ルール・グループは、「ビジネス・ルール・グループの検索」ページの「検索結果」セクションに表示されます。ランタイムから検索結果として戻されたビジネス・ルール・グループの状況は、以下の 4 種類の状況の

いずれかになります。

状況	説明
ローカルと同じ	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに既に存在しており、その内容と結果のビジネス・ルール・グループの内容が厳密に同じであることを示します。したがって、検索後にさらにアクションが実行されることはありません。
ランタイムから変更	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに既に存在していることを示します。ただし、別のユーザー・セッションでマスター・コピーが変更されたため、ローカルの内容と結果のビジネス・ルール・グループの内容は異なります。ビジネス・ルール・マネージャーは、ランタイムからの新しい変更を取得するためにローカル・コピーを自動的に更新します。
ローカルで変更	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに既に存在していることを示します。ただし、それが現行ユーザーによって変更されています。ビジネス・ルール・マネージャーは、ユーザーによる以降のすべてのアクションに対してローカル・コピーを使用します。
ローカルで新規	結果のビジネス・ルール・グループのコピーがビジネス・ルール・マネージャーに存在していないことを示します。この場合、ビジネス・ルール・マネージャーは、結果のビジネス・ルール・グループのローカル・コピーを作成して、それをナビゲーション・ペインにも表示します。

**注:** ビジネス・ルール・グループの変更の同期は、戻される検索結果と同時に実行され、ビジネス・ルール・マネージャーのコンテキストに適用されます。つまり、影響を受けるビジネス・ルール・グループに対する次の操作では、そのビジネス・ルール・グループの最新の更新が使用されます。

## 例

**例:** 次のプロパティが指定された 4 つのビジネス・ルール・グループがインストールされているとします。

### ビジネス・ルール・グループ 1

- 名前: BRDCR002BRG2.brg
- ターゲット名前空間: <http://BRDCR002BRG2/com/ibm/br/rulegroup>
- プロパティ:
  - organization, 7GAA
  - department, accounting

- ID, 0000047
- ID\_cert45, ABC
- region, NorthRegion

#### ビジネス・ルール・グループ 2

- 名前: BRDCR002BRG3.brg
- ターゲット名前空間: <http://BRDCR002BRG3/com/ibm/br/rulegroup>
- プロパティ:
  - organization, 7FAB
  - department, finance
  - ID, 0000053
  - ID\_app45, DEF
  - region, NorthCentralRegion

#### ビジネス・ルール・グループ 3

- 名前: BRDCR002BRG4.brg
- ターゲット名前空間: <http://BRDCR002BRG4/com/ibm/br/rulegroup>
- プロパティ:
  - organization, 7HAA
  - department, shipping
  - ID, 0000023
  - ID\_app45, GHI
  - region, SouthRegion

#### ビジネス・ルール・グループ 4

- 名前: BRDCR002BRG5.brg
- ターゲット名前空間: <http://BRDCR002BRG5/com/ibm/br/rulegroup>
- プロパティ:
  - organization, 8JAA
  - department, claims
  - ID, 00000567
  - region, SouthCentralRegion
  - manager, Joe Bean

1 つのプロパティを使用してビジネス・ルール・グループを取得します。

論理演算子	名前	照会演算子	値
	department	次と等しい	accounting

これは、ビジネス・ルール・グループ 1 を戻します。

2 つのプロパティと複数文字ワイルドカード「%」を使用してビジネス・ルール・グループを取得します。

論理演算子	名前	照会演算子	値
	region	次を含む	%Region
AND	ID	次を含む	00000%

これは、ビジネス・ルール・グループ 1 から 4 を戻します。

単一文字ワイルドカード「\_」を使用してビジネス・ルール・グループを取得します。

論理演算子	名前	照会演算子	値
	ID	次を含む	00000_3

これは、ビジネス・ルール・グループ 2 と 3 を戻します。

複数の単一文字ワイルドカード「\_」を使用してビジネス・ルール・グループを取得します。

論理演算子	名前	照会演算子	値
	region	次を含む	__uth%Region

これは、ビジネス・ルール・グループ 3 と 4 を戻します。

単一文字ワイルドカード「\_」と NOT 演算子を使用してビジネス・ルール・グループを取得します。

論理演算子	名前	照会演算子	値
	organization	次を含まない	7_A

これは、ビジネス・ルール・グループ 4 を戻します。

複数文字ワイルドカード「%」と NOT 演算子を使用してビジネス・ルール・グループを取得します。

論理演算子	名前	照会演算子	値
	organization	次を含まない	7%

これは、ビジネス・ルール・グループ 4 を戻します。

## 次のタスク

結果のビジネス・ルール・グループをクリックすると、そのビジネス・ルール・グループのページが開きます。

## スケジュール済みルール・ロジック項目の使用

スケジュール済みルール・ロジック項目は、ルールの有効な日付やルールに関連付けられた if/then ルール・セットまたはデシジョン・テーブルなど、ルールに関する情報を示します。

## このタスクについて

ビジネス・ルール・マネージャーを使用して、スケジュール済みルール・ロジック項目の作成、変更、または削除を行います。

## スケジュール済みルール・ロジック項目の作成:

スケジュール済みルール・ロジック項目は、既存の項目から作成します。

### 始める前に

ルールを作成するには、編集モードにする必要があります。

### このタスクについて

新しいスケジュール済みルール・ロジック項目を作成するには、以下のステップを実行します。

### 手順

1. 「ルール・スケジュール」 ページで、「**選択レコードの追加**」をクリックします。

新しいスケジュール済みルール・ロジック項目がリストの下部に追加されます。

「**開始日/時刻**」および「**終了日/時刻**」フィールドは、「**1 月 1 日**」に設定されています。日付/時刻フィールドの値が無効であることを示すメッセージが「**メッセージ**」フィールドに表示されます。

2. 「**開始日/時刻**」フィールドを設定します。
  - a. ドロップダウン・リストから月を選択します。
  - b. ドロップダウン・リストから日を選択します。
  - c. 年を入力します。
  - d. 時刻を入力します (24 時間形式)。
3. 「**終了日/時刻**」フィールドを設定します。
  - a. ドロップダウン・リストから月を選択します。
  - b. ドロップダウン・リストから日を選択します。
  - c. 年を入力します。
  - d. 時刻を入力します (24 時間形式)。

**制約事項:** 任意の一時点で 1 つのルール・ロジックのみが有効です。複数のルールの日付/時刻の範囲を重複させることはできません。

**注:** 日付/時刻の範囲に間隔が空いても構いません。デフォルトのルール・ロジックが指定されている場合は、そのロジックがこの間隔で使用されます。ベスト・プラクティスとして、常にデフォルトのルール・ロジックを指定してください。

4. ドロップダウン・リストから「**有効なルール・ロジック**」を選択します。
5. 「**保管**」をクリックします。

### タスクの結果

スケジュール済みルール・ロジック項目が一時的に保管されたことや、変更内容を「**公開および復帰**」ページから公開できることを示すメッセージが「**メッセージ**」フィールドに表示されます。

### 関連タスク

## 211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

### スケジュール済みルール・ロジック項目の変更:

既存のスケジュール済みルール・ロジック項目の日付および時刻の値を変更できます。

#### 始める前に

ルールを変更するには、編集モードにする必要があります。

#### このタスクについて

スケジュール済みルール・ロジック項目を変更するには、以下のステップを実行します。

#### 手順

1. 「ルール・スケジュール」ページで、スケジュール済みルール・ロジック項目の「開始日/時刻」を編集します。
  - a. ドロップダウン・リストから月を選択します。
  - b. ドロップダウン・リストから日を選択します。
  - c. 年を入力します。
  - d. 時刻を入力します (24 時間形式)。
2. スケジュール済みルール・ロジック項目の「終了日/時刻」を編集します。
  - a. ドロップダウン・リストから月を選択します。
  - b. ドロップダウン・リストから日を選択します。
  - c. 年を入力します。
  - d. 時刻を入力します (24 時間形式)。

**制約事項:** 任意の一時点で 1 つのルール・ロジックのみが有効です。複数のルールの日付/時刻の範囲を重複させることはできません。

**注:** 日付/時刻の範囲に間隔が空いても構いません。デフォルトのルール・ロジックが指定されている場合は、そのロジックがこの間隔で使用されます。ベスト・プラクティスとして、常にデフォルトのルール・ロジックを指定してください。

3. 「保管」をクリックします。

**注:** 「日付/時刻」フィールドが無効な場合、そのフィールドは赤色に変わり、さらに日付/時刻フィールド値が無効なことを示すメッセージが「メッセージ」フィールドに表示されます。

## タスクの結果

スケジュール済みルール・ロジック項目がローカル側で保管されます。スケジュール済みルール・ロジック項目をリポジトリに公開する準備ができました。詳しくは、211 ページの『ビジネス・ルールの公開と復帰』を参照してください。

## 次のタスク

ビジネス・ルールの日付の設定について詳しくは、202 ページの『ビジネス・ルールの日付の分割』を参照してください。

### 関連タスク

211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

## 日付/時刻の選択:

ビジネス・ルールを選択するには、日付/時間を指定します。

日付は、ビジネス・ルール・グループの演算パラメーターの一部として定義されるか、実行時に派生されます。日付は、常に UTC で表現され、特定の時点を示します。どの時点であっても、操作に対して有効にできるルール・ロジックは 1 つのみです。また、どの時点であっても、有効なルール・ロジックが見つからない場合は、デフォルトのルール・ロジックが使用されます。

ビジネス・ルール・グループは、以下の日付/時刻オプションをサポートします。これらのオプションにアクセスするには、「開始日/時刻」および「終了日/時刻」フィールドのアイコンをクリックします。

### 「日付/時刻の指定」

日付を手動で指定します。

### 「連続」

終了日をスケジュール済みルール・ロジック項目よりも後の一番早い開始日に設定する自動日付計算を使用します。連続的な日付選択は、「終了日/時刻」フィールドでのみ可能です。

**注:** 連続選択は、2 つのスケジュール済みルール・ロジック項目の日付範囲が連続している場合に使用します。連続属性は、最初のスケジュール済みルール・ロジック項目の終了日に設定されます。この属性が設定されている場合は、2 番目のスケジュール済みルール・ロジック項目の開始日は、最初のスケジュール済みルール・ロジック項目の終了日に設定されます。そのため、両方の日付を指定する必要はありません。

### 「開始日なし」または「終了日なし」

開始または終了 (選択内容に応じて決まります) の境界を設定しません。

**制約事項:** ビジネス・ルール・グループは有効な日付のみをサポートします。他の種類の選択を行う必要がある場合は、セクター・コンポーネントを使用してください。

### ビジネス・ルールの日付の分割:

ビジネス・ルールの日付を分割すると、ビジネス・ルールを別の目的のために簡単に変更できます。

#### 始める前に

ルールを変更するには、編集モードにする必要があります。

#### このタスクについて

スケジュール済みルール・ロジック項目を分割するには、以下のステップを実行します。

#### 手順

1. スケジュール済みルール・ロジック項目の横にある「分割」をクリックします。

開始日が 1 月 1 日で、各フィールドが赤色の新しいスケジュール済みルール・ロジック項目が作成されます。また、日付/時刻フィールド値が無効であることを示すメッセージが「メッセージ」フィールドに表示されます。

2. 新しいスケジュール済みルール・ロジック項目の開始日/時刻を選択します。

元のスケジュール済みルール・ロジック項目の終了日/時刻が継続 から新しいスケジュール済みルール・ロジック項目の開始日/時刻に変更され、新しいスケジュール済みルール・ロジック項目の終了日/時刻が前のスケジュール済みルール・ロジック項目の終了日/時刻に変更されます。

3. 新しいスケジュール済みルール・ロジック項目の「日付/時刻」を変更します。
4. 新規ルールのニーズに合うように「有効なルール・ロジック」を変更します。

### ルール・セット

ルール・セットは *if/then* ステートメント (ルール) のグループです。 *if* がルールの条件で、 *then* がルールのアクションです。ルール・セットは、条件節が非常に少ないビジネス・ルールに最適です。

条件が満たされると、アクションが実行されます。このとき、ルール・セットによって複数のアクションが実行される場合があります。ルール処理の順序は、*if/then* ルール・セット内のルール・ステートメントの順序によって決定されます。したがって、ルールを変更または追加する場合は、正しい順序で配置する必要があります。

ルール・セットは、*if/then* ルールとアクション・ルールの 2 種類のルールを持つことができます。

- *if/then* ルールは、実行するアクションを着信メッセージの条件に従って決定します。
- アクション・ルールは、実行するアクションを着信メッセージに無関係に決定します。

ルール内の条件は条件式を含みます。条件式は単純なストリング、*and*、*or*、または *not* にすることができます。

ルール・セットに定義されたテンプレートを使用して、新規ルール・セットを作成したりビジネス・ルール・マネージャーにある既存のルール・セットを変更したりすることができます。テンプレートは、ルール・セットがどのように機能するかを決定する構造を提供します。ルール・テンプレートはルール・セット間で共有されません。

### ルール・セット項目の作成:

新規のルール・セット項目を作成するには、既存のルール・セット項目をコピーし、その値を変更します。

### このタスクについて

新規ルール・セット項目を作成するには、以下のステップを実行します。

#### 手順

1. 選択したルール・セットのスケジュール済みルール・ロジック項目の横にある「**コピー**」をクリックします。

「編集モード:Copy\_of\_TableName- ルール・セット (Edit Mode:Copy\_of\_TableName-Ruleset)」というタイトルの新規項目の編集ページが開きます。

2. 「名前」フィールドで、新規ルール・セット項目の固有の名前を入力します。
3. 「表示名」フィールドで、新規ルール・セット項目の表示名を入力します。ルール・セットでは、表示名は固有である必要はありません。どのストリング値でも使用でき、特殊文字を含めることもできます。表示名を指定しない場合は、「名前」の値が表示名として使用されます。

注: 表示名と、ルール・セットの対応する名前とを同期させるには、「名前との同期」チェック・ボックスを選択します。

4. 新しいルール・セットの項目の簡単な説明を「説明」フィールドに入力します。
5. 各条件の値を変更します。

ヒント: 各値のパラメーター設定を表示するには、フィールドにカーソルを合わせます。変数の型および範囲を示すロールオーバー・メッセージが表示されません。

6. 上または下矢印をクリックして、ルールを正しい順序に配置します。
7. 「保管」をクリックします。

### タスクの結果

ルール・セット項目が一時的に保管されたことや、変更内容を「公開および復帰」ページから公開できることを示すメッセージが「メッセージ」フィールドに表示されます。

#### 関連タスク

### 211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

### テンプレートからのルール・セット内のルールの作成:

ルール・セットに関連するルール・テンプレートを使用して、そのルール・セット内に新規ルールを作成します。

#### 始める前に

ルール・セットを変更するには、編集モードにする必要があります。

#### このタスクについて

テンプレートから新規のルールを作成するには、以下のステップを実行します。

#### 手順

1. 「**テンプレートからの新規ルール**」をクリックして、ルールに対して使用可能なテンプレートのリストを表示します。
2. テンプレートを選択し、以下を実行します。
  - a. 「**名前**」フィールドで、新規ルールの名前を入力します。
  - b. 「**表示名**」フィールドで、新規ルールの表示名を入力します。ルールでは、表示名は固有である必要はありません。どのストリング値でも使用でき、特殊文字を含めることもできます。表示名を指定しない場合は、「**名前**」の値が表示名として使用されます。

注: 表示名と名前の値とを同期させるには、「**アクション**」フィールドで対応する「**名前の同期**」チェック・ボックスを選択します。このチェック・ボックスが選択されている場合は、ルールの表示名が使用不可になり、変更できません。
  - c. 入力フィールド内にルールの値を指定するか、またはドロップダウン・リストから変数を選択します。
  - d. ルールの説明を入力します。
  - e. 「**追加**」をクリックします。
3. 「**アクション**」フィールドで上矢印または下矢印をクリックして、ルールを適切な順序で配置します。

注: ルール処理の順序は、if/then ルール・セット内のルール・ステートメントの順序によって決定されます。したがって、ルールを変更または追加する場合は、正しい順序で配置する必要があります。

4. 「**保管**」をクリックします。

## 次のタスク

ルール・セットを公開する準備ができました。詳しくは、211 ページの『ビジネス・ルールの公開と復帰』を参照してください。

### 関連タスク

211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

### テンプレートを使用したルール・セット内のルールの変更:

ルール・セットに関連するテンプレートを使用して、そのルール・セット内のルールを変更します。

### 始める前に

ルール・セットを変更するには、編集モードにする必要があります。

### このタスクについて

既存のテンプレートを使用してルールを変更するには、以下のステップを実行します。

### 手順

1. 値を編集します。入力フィールド内の既存の値に上書きして入力するか、またはフィールド内に表示される下矢印をクリックし、ドロップダウン・リストから値を選択します。
2. 必要に応じて、上または下矢印をクリックして、ルールを適切な順序に配置します。

**注:** ルール処理の順序は、if/then ルール・セット内のルール・ステートメントの順序によって決定されます。したがって、ルールを変更または追加する場合は、正しい順序で配置する必要があります。

3. 「保管」をクリックします。

## 次のタスク

変更されたルール・セットを公開する準備ができました。詳しくは、211 ページの『ビジネス・ルールの公開と復帰』を参照してください。

### 関連タスク

211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック

ク」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

## デシジョン・テーブル

デシジョン・テーブルは、表形式のスケジュール済みルール・ロジック項目です。このテーブルでは、条件が行見出しおよび列見出しとして表され、アクションがそれらの条件ケースの交点として表されます。デシジョン・テーブルは、複数の条件を持つビジネス・ルールに最適です。新しい条件を追加する場合も、新しい行または列を追加するだけで済みます。

if/then ルール・セットと同様に、デシジョン・テーブルは条件とアクションの相互作用によって駆動されます。主な違いは、デシジョン・テーブルでは、アクションが複数の条件によって決定され、条件の各セットに複数のアクションを関連付けることができる点です。条件が満たされると、対応するアクション（複数可）が実行されます。

## テンプレート

ビジネス・ルール・マネージャーでデシジョン・テーブル値を変更するには、テンプレートを使用します。テンプレートは、WebSphere Integration Developer で設計され、ビジネス・ルール定義に含まれています。テンプレートは、変更可能なデシジョン・テーブルの側面を決定し、選択可能な有効な値のリストを提供します。デシジョン・テーブルに定義されたテンプレートに基づいて、テーブルに新規の行/列、または新規のアクションを作成し、テンプレートで作成した既存の条件またはアクションを変更します。デシジョン・テーブル・テンプレートはデシジョン・テーブル間で共有されません。

## 初期化アクション・ルール

デシジョン・テーブルは初期化アクション・ルールの使用をサポートします。初期化アクション・ルールはデシジョン・テーブルの実行前に実行され、ビジネス・オブジェクトの作成や初期値の設定などの前処理を行うことができます。ビジネス・ルール定義が初期化アクション付きで WebSphere Integration Developer で設計されている場合、ビジネス・ルール・マネージャーで初期化アクション・ルールを変更できます。

単一のテンプレートから作成できる初期化アクション・ルールは 1 つだけですが、アクション・ルールは複数のアクション式を持つことができるため、複数のアクションを実行できます。初期化ルール・テンプレートが特定のデシジョン・テーブルに定義されている場合は、そのテーブルのみで使用できます。

### 「その他の場合」条件

「その他の場合」条件は、デシジョン・テーブル内の他の条件が適用されない場合にデフォルトで入力される特別な条件です。

WebSphere Integration Developer 内で設計されたデシジョン・テーブル定義に「その他の場合」条件がある場合にのみ、この条件がビジネス・ルール・マネージャー内に表示されます。ビジネス・ルール・マネージャーでは、ルールを動的に追加したり除去したりできません。

ただし、「その他の場合」条件のテンプレートで定義したアクションを取り込むことはできません。「その他の場合」条件は、検査する条件 1 つにつき 0 回または 1 回使用できます。

下図は、デフォルト・メンバー・タイプをシルバーに設定する初期化アクション・ルールと、500 ドル未満を使うゴールド顧客とシルバー顧客に適用される「その他の場合」条件を持つデシジョン・テーブルを示します。条件 PurchaseAmount および MemberType は、1 番目と 2 番目の行に沿って存在します。アクション Discount は、3 番目の行に沿って存在します。条件とアクションの方向を矢印によって示します。

Initialization Rule						
Display Name	Rule	Description				
Rule1	Default Member Type = Silver					

Decision Table						
PurchaseAmount	>= 500 && < 2000		>= 2000		Otherwise	
MemberType	Gold	Silver	Gold	Silver	Gold	Silver
Discount	8 %	3 %	10 %	5 %	2 %	0 %

図 4. デシジョン・テーブル

この例では、500 ドルから 1999 ドルを使うゴールド顧客は 8% の割引を受け、500 ドルから 2000 ドルを使うシルバー顧客は 3% の割引を受けることが示されています。2000 ドル以上を使うゴールド顧客は 10% の割引を受け、2000 ドル以上を使うシルバー顧客は 5% の割引を受けます。500 ドル未満を使うゴールド顧客は 2% の割引を受け、500 ドル未満を使うシルバー顧客は 0% の割引を受けます。

#### デシジョン・テーブル項目の作成:

新規のデシジョン・テーブル項目を作成するには、既存のデシジョン・テーブル項目をコピーし、その値を変更します。

#### このタスクについて

デシジョン・テーブル項目を作成するには、以下のステップを実行します。

#### 手順

1. 選択したデシジョン・テーブルのスケジュール済みルール・ロジック項目の横にある「コピー」をクリックします。

「編集モード:Copy\_of\_TableName - デシジョン・テーブル

(Edit Mode:Copy\_of\_TableName-Decision Table)」というタイトルの新規項目の編集ページが開きます。

2. 「名前」フィールドで、新規デシジョン・テーブル項目の名前を入力します。
3. 「表示名」フィールドで、新規デシジョン・テーブル項目の表示名を入力します。デシジョン・テーブルでは、表示名は固有である必要はありません。どのストリング値でも使用でき、特殊文字を含めることもできます。表示名を指定しない場合は、「名前」の値が表示名として使用されます。

注: 表示名と名前の値とを同期させるには、対応する「名前との同期」チェック・ボックスを選択します。

4. 新しいデシジョン・テーブルの項目の簡単な説明を「説明」フィールドに入力します。
5. 各条件の値を変更します。

ヒント: 各値のパラメーター設定を表示するには、フィールドにカーソルを合わせます。変数の型および範囲を示すロールオーバー・メッセージが表示されません。

6. 「保管」をクリックします。

### タスクの結果

デシジョン・テーブル項目が一時的に保管されたことや、変更内容を「公開および復帰」ページから公開できることを示すメッセージがメッセージ・フィールドに表示されます。詳しくは、211 ページの『ビジネス・ルールの公開と復帰』を参照してください。

### 関連タスク

211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

### 「特別なアクション (Special actions)」メニュー:

「デシジョン・テーブル」ページには、デシジョン・テーブルの値を編集したり、テンプレートの構造および変数を変更したりするための「特別なアクション (Special actions)」メニューがあります。

「特別なアクション (Special actions)」メニューは、デシジョン・テーブルが編集モードのときに、横に「特別なアクション (Special actions)」アイコンが表示されるすべてのフィールドで使用可能です。フィールドの「特別なアクション (Special actions)」アイコンをクリックすると、そのフィールドで使用可能なオプションのリストが開きます。有効なオプションを以下の表にリストします。

注: 列または行を再配列しても、表のビジュアル表示にのみ影響し、条件およびアクションが処理される順序には影響しません。

メニュー・オプション	説明	条件の変更	アクションの変更
下に追加	新規の条件値 (行) を現行セルの下に追加します (方向は垂直)	可能	

メニュー・オプション	説明	条件の変更	アクションの変更
右に追加	新規の条件値をセルの右側に追加します (水平方向に追加)	可能	
テンプレートの変更	セル値を変更できます	可能	可能
上へ移動	条件値または変数を上の行に移動します (方向は垂直)	可能	
下へ移動	条件値または変数を下の行に移動します (方向は水平)	可能	
左へ移動	条件値または変数を左に移動します (方向は水平)	可能	
右へ移動	条件値または変数を右に移動します (方向は垂直)	可能	
削除	条件値または変数を削除します	可能	
メニューを閉じる	メニューを閉じます	可能	可能

### デシジョン・テーブル項目の変更:

デシジョン・テーブルを編集するには、該当する入力フィールドに新規の値を直接入力するか、またはフィールドのリスト・ボックス・オプションから値を選択します。

#### 始める前に

デシジョン・テーブルを変更するには、編集モードにする必要があります。

#### このタスクについて

デシジョン・テーブルの値を変更するには、以下のステップを実行します。

#### 手順

1. 値を編集します。入力フィールド内の既存の値に上書きして入力するか、またはフィールド内に表示される下矢印をクリックし、ドロップダウン・リストから値を選択します。

#### 制約事項:

- 初期設定のルールは、WebSphere Integration Developer 内で設計されたビジネス・ルール定義に初期設定のルールがある場合にのみ、デシジョン・テーブルに表示されます。単一のテンプレートに関連付けることができる初期化アクション・ルールは 1 つのみですが、アクション・ルールの内部には複数のアクション式を含めることができます。

- 「その他の場合」条件は、WebSphere Integration Developer 内で設計されたビジネス・ルール定義に「その他の場合」条件がある場合にのみ、デシジョン・テーブルに表示されます。ビジネス・ルール・マネージャーで「その他の場合」条件を追加または除去することはできません。ただし、「その他の場合」条件のテンプレートで定義したアクションを取り込むことはできます。
2. フィールドの横にある「特別なアクション (Special actions)」アイコンをクリックして、フィールドで使用可能なオプションのリストを開き、必要なアクションを選択します。

注: 列または行を再配列するオプションの選択は、テーブルのビジュアル表示にのみ影響し、条件およびアクションが処理される順序には影響しません。

3. 「保管」をクリックします。

### タスクの結果

ルールがローカル側で変更されます。ルールをリポジトリに公開する準備ができました。詳しくは、211 ページの『ビジネス・ルールの公開と復帰』を参照してください。

#### 関連タスク

211 ページの『スケジュール済みルール・ロジック項目の削除』

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

#### デシジョン・テーブルのテンプレート値の変更:

デシジョン・テーブル・テンプレートの構造と値を変更するには、「特別なアクション (Special actions)」メニューを使用し、該当する入力フィールドに値を直接入力します。

#### 始める前に

デシジョン・テーブルを変更するには、編集モードにする必要があります。

#### このタスクについて

デシジョン・テーブル・テンプレートを変更するには、以下のステップを実行します。

#### 手順

1. 変更するデシジョン・テーブル・フィールドの横にある「特別なアクション (Special actions)」アイコンをクリックして、フィールドのオプションのリスト・ボックスを開き、「テンプレートの変更」を選択します。
2. 入力フィールド内の既存の値にテンプレートの新規の値を上書きして入力します。

3. 「アクション」列の「変更」をクリックします。
4. 「保管」をクリックします。

### タスクの結果

デシジョン・テーブル・テンプレートは変更され、公開する準備ができました。詳しくは、『ビジネス・ルールの公開と復帰』を参照してください。

### スケジュール済みルール・ロジック項目の削除

既存のスケジュール済みルール・ロジック項目をスケジュール済みルール・ロジック・テーブルから削除できます。スケジュール済みルール・ロジック項目を削除した場合、それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。スケジュール済みルール・ロジック項目は、デフォルトのルール・ロジックとして、または特定の日時を指定して再度追加できます。

### 始める前に

ルールを削除するには、編集モードにする必要があります。

### このタスクについて

スケジュール済みルール・ロジックを削除するには、以下のステップを実行します。

#### 手順

1. 「ルール・スケジュール」ページで、スケジュール済みルール・ロジックを選択し、「削除」をクリックします。

スケジュール済みルール・ロジックが削除されます。それに関連付けられているルール・セットまたはデシジョン・テーブル定義はルール・グループに残り、そのページの「使用可能なルール・ロジック」セクション内にリストされます。

**注:** ビジネス・ルール・グループに対する各操作には、スケジュール済みルール・ロジック項目として、またはデフォルトのルール・ロジックとして関連付けられた 1 つ以上のアクティブなビジネス・ルールが存在する必要があります。すべてのスケジュール済みルール・ロジック項目を削除しようとする、エラーになります。

2. 「保管」をクリックします。

### タスクの結果

スケジュール済みルール・ロジック項目が一時的に保管され、リポジトリに公開する準備が整います。

### ビジネス・ルールの公開と復帰

ビジネス・ルール・グループの一部を保管するときには、変更内容はローカル側で保管されます。アプリケーション・サーバー上にあるデータ・ソースに変更内容を格納するには、変更内容を公開する必要があります。また、ビジネス・ルールをその元の状態に復帰させることにより、ルールに対してローカル側で保管された変更内容を取り消すこともできます。

## 始める前に

ナビゲーション・ペインを持つ任意のビジネス・ルール・マネージャー・ページ上で作業する必要があります。

## このタスクについて

サーバーは、ビジネス・ルール・グループ・レベルとルール・スケジュール・レベルで変更を公開します。公開ステージでは、ビジネス・ルール・マネージャーは検証を行う必要はありません。なぜなら、ビジネス・ルール・マネージャーは、各編集ページで入力された変更内容を、情報の保存時にすべて検証するからです。

ビジネス・ルール・グループまたはルール・スケジュールへの変更内容を公開するには、以下のステップを実行します。

### 手順

1. 「公開および復帰」をクリックします。
2. 「公開および復帰」ページで、コンテンツ領域の左端の列にあるチェック・ボックスをクリックして、リポジトリに送信するビジネス・ルール・グループおよびルール・スケジュールを選択します。すべてのビジネス・ルール・グループおよびルール・スケジュールをまとめて単一トランザクションとして公開するか、またはそれらのサブセットのみを公開するかを選択できます。

**注:** ビジネス・ルール・グループまたはルール・スケジュールに対してローカル側で保存された変更をすべて取り消して、変更されたリソースをサーバー・メモリー内の元のコピーで置き換えるには、ビジネス・ルール・グループまたはルール・スケジュールのチェック・ボックスを選択して、「復帰」をクリックします。ビジネス・ルール・グループおよびルール・スケジュールを公開すると、サーバー・メモリー内にある元のコピーが変更されるため、公開の後でそれらを復帰させることはできません。

3. 「公開」をクリックします。

選択されているビジネス・ルール・グループとルール・スケジュールがサーバー・メモリーに書き込まれます。

## 次のタスク

ビジネス・ルールをデータ・ソースにエクスポートする準備ができました。

## ビジネス・ルール・マネージャーのトラブルシューティング

ビジネス・ルール・マネージャーの使用中に発生する可能性のある問題には、ログイン・エラー、ログイン競合、アクセス競合などがあります。

これらの問題のトラブルシューティングには、さまざまな対策を取ることができます。

### ログイン・エラーの解決:

ログイン・エラーは、ログイン時に発生します。

## 始める前に

### このタスクについて

ログイン・エラー・メッセージは、以下のとおりです。

ログインを処理できません。ユーザー ID とパスワードを確認して、再試行してください。

**注:** ログイン・エラーは、管理セキュリティが使用可能になっていて、ユーザー ID とパスワードのいずれかまたは両方に誤りがある場合にのみ発生します。

ログイン・エラーを解決するには、以下のステップを実行します。

### 手順

1. エラー・メッセージで「OK」をクリックして、「ログイン」ページに戻ります。
2. 「ユーザー ID」と「パスワード」に有効な値を入力します。
  - パスワードで大/小文字を区別する必要がある場合は、Caps Lock キーがオンになっていないことを確認してください。
  - ユーザー ID とパスワードのスペルが正しいことを確認してください。
  - システム管理者に問い合わせ、ユーザー ID とパスワードが正しいことを確認してください。
3. 「ログイン」をクリックします。

### 次のタスク

ログイン・エラーが解決された場合は、ビジネス・ルール・マネージャーにログインできるようになります。エラーが解決されない場合は、システム管理者に連絡してください。

### ログイン競合エラーの解決:

ログイン競合エラーは、同じユーザー ID を持つ別のユーザーが既にアプリケーションにログインしている場合に発生します。

## 始める前に

### このタスクについて

ログイン競合メッセージは、以下のとおりです。

同じユーザー ID を使って、別のユーザーが現在ログインしています。次の中から選択してください。

通常、ユーザーがログアウトせずにブラウザを閉じると、このエラーが発生します。この状態が発生した場合、セッションがタイムアウトする前に次のログインが試行されると、ログイン競合になります。

**注:** ログイン競合エラーは、管理セキュリティが使用可能になっている場合にのみ発生します。

ログイン競合エラーを解決するには、以下の 3 つのオプションから選択します。

- ログイン・ページに戻る。

別のユーザー ID を使用してアプリケーションを開く場合は、このオプションを使用します。

- 同じユーザー ID を持つ別のユーザーをログアウトする。

別のユーザーをログアウトし、新しいセッションを開始する場合は、このオプションを使用します。

**注:** 他のセッションで行われた未公開のローカル変更は、すべて失われます。

- 同じユーザー ID を持つ別のユーザーのコンテキストを継承し、そのユーザーをログアウトする。

進行中の作業を続行する場合は、このオプションを使用します。前のセッションで保管された未公開のローカル変更は、すべて保持されます。ビジネス・ルール・マネージャーは、前のセッションで表示された最後のページを開きます。

#### アクセス競合エラーの解決:

1 人のユーザーがデータ・ソース内のビジネス・ルールを更新しているときに、別のユーザーが同時に同じルールを更新しようとすると、アクセス競合エラーが発生します。

#### 始める前に

ローカル側の変更をリポジトリに公開すると、このエラーが報告されます。

#### このタスクについて

アクセス競合エラーを解決するには、以下のアクションを実行します。

- エラーの原因となっているビジネス・ルールのソースを検索し、ローカル・マシンでの変更がまだ有効かどうかチェックします。別のユーザーが変更を行った後では、ローカル側で変更する必要がない場合があります。
- ビジネス・ルール・マネージャーで作業を続行する場合は、エラーが発生したビジネス・ルール・グループおよびルール・スケジュールをデータ・ソースから再ロードする必要があります。これは、エラーが発生したビジネス・ルール・グループおよびルール・スケジュールのローカル変更は使用できなくなるからです。エラーが報告されたルールの「公開および復帰」ページで、「再ロード」をクリックして、ビジネス・ルール・グループまたはルール・スケジュールのページを再ロードします。エラー状態でない他のビジネス・ルール・グループおよびルール・スケジュールのローカル変更は引き続き使用できます。

## セレクター・コンポーネントの概要

ビジネスの変更に応じて、ビジネスを駆動するビジネス・プロセスも変更する必要があります。プロセスの設計は変更することなく、特定のプロセスが元の設計とは異なる結果を戻すよう変更しなければならない場合もあります。セレクター・コンポーネントは、このような状況に柔軟に対応するためのフレームワークを提供します。

セレクター・コンポーネントは、特定の基準に基づいて結果を変更するサービスに対する単一のインターフェースを提供します。セレクター・コンポーネントは、インターフェースおよびセレクター・テーブルに含まれています。セレクター・テーブルは、要求を処理するコンポーネント (ターゲット・コンポーネントと呼ばれる) を基準に基づいて決定します。サーバーは、ターゲット・コンポーネントによって提供された処理結果をクライアントに戻します。

ビジネス・プロセスの構築時に、ソリューション設計者はセレクター・コンポーネントの要件を識別し、セレクター・コンポーネントが処理を完了するために使用するインターフェースおよびセレクター・テーブルを定義します。セレクター・コンポーネントの開発に関連する作業については、WebSphere Integration Developer インフォメーション・センターを参照してください。

セレクター・コンポーネントの管理は、セレクター・コンポーネントに関連した作業またはセレクター・テーブルに関連した作業で構成されます。

**制約事項:** セレクター・コンポーネント・ページにアクセスするには、管理コンソールにログインする際にユーザー ID を指定する必要があります。ユーザー ID を指定せずにログインすると、ログアウトして有効なユーザー ID でもう一度ログインするよう、警告を受信します。

## セレクター・コンポーネントの表示

セレクター・コンポーネントの表示は、セレクター・コンポーネントを管理する最初のステップです。この表示から、セレクター・コンポーネントの一部またはすべてをエクスポートしたり、セレクター・コンポーネントから構成されるセレクター・テーブルを表示したりできます。

### 始める前に

このタスクを実行するには、ユーザーは WebSphere Process Server の管理コンソールを使用していなければなりません。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

### このタスクについて

サーバーに存在するセレクター・コンポーネントを確認するには、以下のステップを実行します。

**制約事項:** セレクター・コンポーネント・ページにアクセスするには、管理コンソールにログインする際にユーザー ID を指定する必要があります。ユーザー ID を指定せずにログインすると、ログアウトして有効なユーザー ID でもう一度ログインするよう、警告を受信します。

### 手順

1. ナビゲーション・ペインで「サーバー」をクリックして、さまざまなサーバー・タイプを表示します。

2. 「アプリケーション・サーバー」をクリックして、アプリケーション・サーバー・リストを展開します。
3. サーバー・リスト内のサーバーの名前をクリックします。
4. 「Business Integration」の下で「セクター」>「セクター」をクリックします。

コンソールには、すべてのセクター・コンポーネントと各コンポーネントの説明が表示されます。

## セクター・テーブルの表示

セクター・テーブルの表示は、テーブルを管理する最初のステップです。ターゲット・コンポーネントのリストが表示されるので、処理基準の変更、特定の基準で実行されるターゲット・コンポーネントの変更、新規ターゲット・コンポーネントの追加、テーブルからのターゲット・コンポーネントの削除 (基準の削除) などをご画面から実行できます。

### 始める前に

このタスクを実行するには、ユーザーは WebSphere Process Server の管理コンソールを使用していなければなりません。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

### このタスクについて

セクター・テーブルを表示し、テーブルを構成する項目を確認して、その他のセクター・テーブル関連のタスクを実行します。セクター・テーブルを表示するには、以下のステップを実行します。

#### 手順

1. 「サーバー」>「アプリケーション・サーバー」>「*servername*」>「Business Integration」>「セクター」>「セクター」をクリックして、セクター・コンポーネントを表示します。
2. セクター・コンポーネントの表示からセクター名をクリックして、選択したコンポーネントのセクター・テーブルを表示します。
3. 画面上のセクター・テーブルを 1 つクリックして、セクター・テーブルを構成するターゲット・コンポーネントを表示します。

## ターゲット・コンポーネントの変更

ターゲット・コンポーネントを変更すると、セクター・コンポーネント処理を変更できます。そのためには、特定のターゲット・コンポーネントの選択基準を変更するか、選択基準に対してターゲット・コンポーネントを変更するか、または選択基準とターゲット・コンポーネントの両方を変更します。

### 始める前に

このタスクを実行するには、セクター・テーブルが存在している必要があります。

**このタスクに必要なセキュリティー・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者としてログインする必要があります。

## このタスクについて

ターゲット・コンポーネントを変更して、選択基準を変更したり、セレクトター・テーブルの項目に別のターゲット・コンポーネントを使用したりします。ターゲット・コンポーネントを変更するには、以下のステップを実行します。

**重要:** 長期実行アプリケーションのターゲット・コンポーネントを変更する場合は、アプリケーションを停止します。長期実行アプリケーションの処理中には、ターゲット・コンポーネントを変更しないでください。

### 手順

1. 『セレクトター・テーブルの表示』で説明されているように、セレクトター・テーブルを表示します。「サーバー」>「アプリケーション・サーバー」> *servername* > 「**Business Integration**」> 「セレクトター」> 「セレクトター」> *selectername* をクリックします。
2. 画面上のセレクトター・テーブルを 1 つクリックして、セレクトター・テーブルを構成するターゲット・コンポーネントを表示します。
3. 変更するターゲット・コンポーネントのターゲット ID をクリックします。
4. 項目を変更します。

### 項目の変更する部分 ターゲット宛先

### 変更の手順

1. ターゲット・コンポーネント・リストの隣にある矢印をクリックして、適格なターゲット・コンポーネントを表示します。
2. リストからターゲット・コンポーネントを選択します。

### 選択基準

1. 「開始日」、「終了日」、またはその両方を上書きします。入力する日付はシステムのロケールによって異なり、ロケール形式に応じて表示されます。米国英語ロケールの場合、表示される形式は以下のとおりです。
  - 月
  - 日
  - 年 (YYYY 形式)
  - 時刻 (HH:MM:SS 形式)
  - 時間帯

**重要:** 「開始日」は、「終了日」より前の日付を指定する必要があります。そうしないと、この変更をコミットできません。

## 項目の変更する部分 ターゲット宛先および選択基準

## 変更の手順

1. ターゲット・コンポーネント・リストの隣にある矢印をクリックして、適格なターゲット・コンポーネントを表示します。
2. リストからターゲット・コンポーネントを選択します。
3. 「開始日」、「終了日」、またはその両方を上書きします。入力する日付はシステムのロケールによって異なり、ロケール形式に応じて表示されます。米国英語ロケールの場合、表示される形式は以下のとおりです。
  - 月
  - 日
  - 年 (YYYY 形式)
  - 時刻 (HH:MM:SS 形式)
  - 時間帯

**重要:** 「開始日」は、「終了日」より前の日付を指定する必要があります。そうしないと、この変更をコミットできません。

5. オプション: 「デフォルト」チェック・ボックスをクリックして、これをデフォルトのターゲット・コンポーネントにします。

選択基準が他のターゲット・コンポーネントの範囲に含まれない場合、セレクトター・コンポーネントはこのターゲット・コンポーネントを使用します。

6. この画面で作業を継続するには「適用」、ターゲット・コンポーネントの画面に戻るには「OK」をクリックします。
7. ターゲット・コンポーネント画面で「保管」をクリックして、変更内容をセレクトター・テーブルに保管します。

## タスクの結果

セレクトター・テーブル・ファイルには、更新された選択基準とターゲット・コンポーネントが格納されます。セレクトター・コンポーネントは、更新されたセレクトター・テーブルを使用して、受信した次の要求を処理します。

## ターゲット・コンポーネントの追加

現在セレクトター・テーブルに存在するものと異なる選択基準に合う追加処理が必要な場合、ターゲット・コンポーネントを追加します。

## 始める前に

このタスクを実行するには、セレクトター・テーブルが存在している必要があります。

**このタスクに必要なセキュリティー・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者としてログインする必要があります。

## このタスクについて

ビジネス・プロセスにさらに柔軟性が必要な場合、ターゲット・コンポーネントを追加します。セクター・コンポーネントがアクティブである間、新規コンポーネントを追加できます。

ターゲット・コンポーネントを追加するには、以下のステップを実行します。

### 手順

1. 『セクター・テーブルの表示』で説明されているように、セクター・テーブルを表示します。「サーバー」>「アプリケーション・サーバー」> *servername* >「**Business Integration**」>「セクター」>「セクター」> *selectorname* をクリックします。
2. 画面上のセクター・テーブルを 1 つクリックして、セクター・テーブルを構成するターゲット・コンポーネントを表示します。
3. 「新規」をクリックして、データが事前に入力されたターゲット・コンポーネント・ページを表示します。
4. 『ターゲット・コンポーネントの変更』で説明されているように、アプリケーション要件に合わせてターゲット宛先情報を編集します。
5. 「OK」をクリックしてターゲット・コンポーネントを保管し、ターゲット・コンポーネント画面に戻ります。

### タスクの結果

セクター・テーブルに新規ターゲット・コンポーネントが組み込まれます。セクター・コンポーネントは、更新されたセクター・テーブルを使用して、受信した次の要求を処理します。

### ターゲット・コンポーネントの削除

セクター・テーブル内で特定の選択基準に合う項目を削除することによってターゲット・コンポーネントを削除すると、セクター・コンポーネントの処理が変更されます。

### 始める前に

このタスクを実行するには、セクター・テーブルが存在している必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者としてログインする必要があります。

## このタスクについて

ビジネス・プロセスでターゲット・コンポーネントの処理が不要になったら、ターゲット・コンポーネントを削除します。ターゲット・コンポーネントの削除後は、要求が入力されてそれが他のどの特定の選択基準とも一致しない場合、デフォルトの基準によって要求が処理されます。

ターゲット・コンポーネントを削除するには、以下のステップを実行します。

## 手順

1. 『セレクトター・テーブルの表示』で説明されているように、ターゲット・コンポーネントを表示します。
2. 削除するターゲット・コンポーネントの横にあるチェック・ボックスをクリックし、「削除」をクリックします。

システムによってページが更新され、残りのターゲット・コンポーネントが表示されます。

3. 「保管」をクリックします。

残りのターゲット・コンポーネントの項目を表示している更新されたセレクトター・テーブルが、システムによって保管されます。

## タスクの結果

セレクトター・テーブル・ファイルには、残りのターゲット・コンポーネントのみが含まれるようになります。セレクトター・コンポーネントは、更新されたセレクトター・テーブルを使用して、受信した次の要求を処理します。

## 管理コンソールを使用したセレクトター・コンポーネントのエクスポート

セレクトター・テーブルを変更したら、セレクトター・コンポーネントをエクスポートします。これにより、開発環境にインポートできるファイルが作成され、開発の成果物を実際の実動システムの成果物と同期させることができます。

## 始める前に

このタスクを開始するには、『セレクトター・コンポーネントの表示』の説明に従って、セレクトター・コンポーネントを事前に表示しておく必要があります。「サーバー」>「アプリケーション・サーバー」> *servername* > 「Business Integration」> 「セレクトター」> 「セレクトター」をクリックします。

**このタスクに必要なセキュリティ・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。セキュリティが有効でないときは、ユーザー ID を指定して管理コンソールにログインする必要があります。

## このタスクについて

管理コンソールを使用してセレクトターをエクスポートするには、以下のステップを実行します。

## 手順

1. 1 つ以上のセレクトターの隣にあるチェック・ボックスを選択して、「エクスポート」をクリックします。

ブラウザには、選択したセレクトター・コンポーネントへの HTML リンクのリストが表示されます。これは、「エクスポートするセレクトター」ページです。各セレクトターのファイル拡張子は、.zip です。

2. 各ファイル名をクリックして、ファイルをファイル・システムにダウンロードします。ファイルを保存するためのプロンプトがシステムから出されたら、「OK」をクリックします。

注: ファイルの保存を選択した場合は、ファイルのダウンロード時にファイル名を変更できます。

3. 「戻る」をクリックして、セレクトターのリストに戻ります。

### タスクの結果

指定した場所にファイルが保管されます。

### 管理コンソールを使用したセレクトター・コンポーネントのインポート

インストールしたセレクトター・コンポーネントを、アプリケーションを再インストールせずに更新するには、セレクトターをインポートします。

### 始める前に

管理コンソールで作業する必要がある、エクスポート機能によって作成された圧縮ファイルのロケーションを知っている必要があります。

**このタスクに必要なセキュリティー・ロール:** セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

### このタスクについて

インストール済みのアプリケーションが使用しているセレクトターを変更した後で、それらの変更内容を別のクラスターまたはサーバーにも適用したい場合に、セレクトターをインポートします。ご使用の開発環境と実稼働環境内の変更内容を同期させるときにも、この機能を使用できます。

管理コンソールを使用してセレクトターをインポートするには、以下のステップを実行します。

**ヒント:** セレクトター・コンポーネントは、コマンド行を使用してインポートすることもできます。

### 手順

1. 『セレクトター・コンポーネントの表示』で説明されているように、セレクトター・コンポーネントのインポート先のサーバー上のセレクトターを表示します。「サーバー」>「アプリケーション・サーバー」> *servername* > 「Business Integration」> 「セレクトター」> 「セレクトター」をクリックします。
2. 「インポート」をクリックします。
3. 「セレクトターのインポートの準備」ページ上でファイルへのパスを指定します。

### 次のタスク

更新されたセレクトターのセレクトター・テーブルを表示して、変更内容を確認します。



## 第 9 章 バインディングの操作

サービス指向アーキテクチャの中核をなすのは「サービス」の概念で、これはコンピュータ・デバイス間での対話によって実行される機能単位のことです。「エクスポート」はモジュールの外部インターフェース (またはアクセス・ポイント) を定義し、これによってモジュール内の SCA コンポーネントは外部クライアントにサービスを提供できます。「インポート」はモジュール外部のサービスへのインターフェースを定義し、これによってサービスはモジュール内部から呼び出すことができます。インポートおよびエクスポートとともにプロトコル固有の「バインディング」を使用して、データをモジュールの内部または外部に移送する方法を指定します。

### エクスポート

外部クライアントがインテグレーション・モジュール内の SCA コンポーネントを呼び出すとき、さまざまなプロトコル (HTTP、JMS、MQ、および RMI/IIOP など) を経由して、さまざまなデータ・フォーマット (XML、CSV、COBOL、および JavaBean など) が使用されます。エクスポートはこれらの要求を外部ソースから受信し、SCA プログラミング・モデルを使用して WebSphere Process Server コンポーネントを呼び出すコンポーネントです。

例えば以下の図で、エクスポートはクライアント・アプリケーションから HTTP プロトコルを経由して要求を受信します。データは SCA コンポーネントによって使用されるフォーマットであるビジネス・オブジェクトに変換されます。そのコンポーネントはそのデータ・オブジェクトによって呼び出されます。

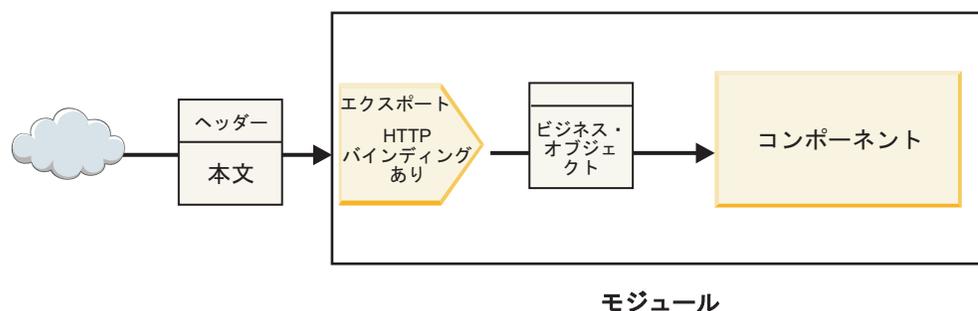


図5. HTTP バインディングを持つエクスポート

### インポート

SCA コンポーネントから、SCA とは異なるフォーマットのデータを要求する SCA 以外の外部サービス呼び出すことが必要な場合があります。インポートは SCA プログラミング・モデルを使用して外部サービス呼び出すために、SCA コンポーネントによって使用されます。インポートは次に、ターゲット・サービスが要求する方法でサービス呼び出します。

例えば以下の図で、SCA コンポーネントからの要求がインポートによって外部サービスに送信されます。SCA コンポーネントによって使用されるフォーマットである

ビジネス・オブジェクトは、サービスが要求するフォーマットに変換され、サービスが呼び出されます。

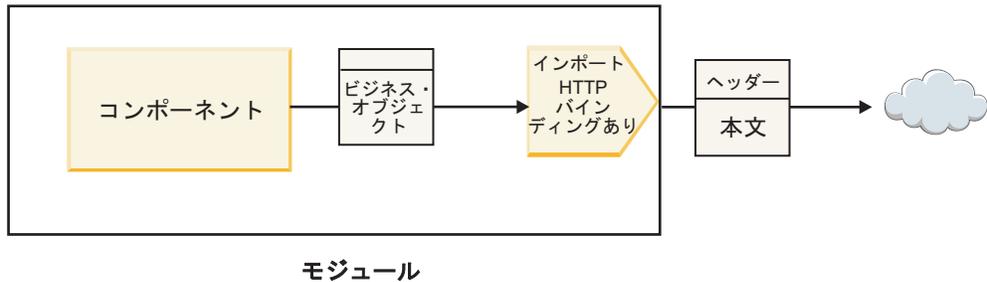


図6. HTTP バインディングを持つインポート

## バインディングのリスト

インポートまたはエクスポートのバインディングを生成して、バインディングを構成するには、WebSphere Integration Developer を使用します。使用可能なバインディングのタイプを以下の一覧に記載します。

- SCA

SCA バインディングはデフォルトのバインディングで、使用するサービスを他の SCA モジュールのサービスと通信させることができます。インポートと SCA バインディングを使用して、別の SCA モジュールのサービスにアクセスします。エクスポートと SCA バインディングを使用して、別の SCA モジュールにサービスを提供します。

SCA バインディングについて詳しくは、『SCA バインディング』を参照してください。

- Web サービス

Web サービス・バインディングを使用すれば、相互運用可能な SOAP メッセージおよびサービス品質を使用して外部サービスにアクセスすることができます。Web サービス・バインディングでは SOAP/HTTP (SOAP over HTTP) または SOAP/JMS (SOAP over JMS) のいずれかのトランスポート・プロトコルを使用できます。

Web サービス・バインディングについて詳しくは、『Web サービス・バインディング』を参照してください。

- HTTP

HTTP バインディングを使用すれば、SOAP 以外のメッセージが使用されている場合、または直接 HTTP アクセスが必要な場合に、HTTP プロトコルを使用して外部サービスにアクセスすることができます。このバインディングは、HTTP モデルに基づく Web サービス (GET、PUT、DELETE などのよく知られた HTTP インターフェース操作を使用するサービス) を処理するときに使用されます。

- EIS

EIS (エンタープライズ情報システム) バインディングを JCA リソース・アダプターと一緒に使用すると、エンタープライズ情報システム上のサービスにアクセスしたり、サービスを EIS で使用可能にすることができます。

- JMS バインディング

Java Message Service (JMS)、汎用 JMS、および WebSphere MQ JMS (MQ JMS) バインディングは、メッセージ・キューを経由した非同期通信が信頼性の維持に欠かせない場合に、メッセージング・システムとの対話に使用されます。

これらの JMS バインディングのいずれかを使用するエクスポートは、キューにメッセージが到着するのを監視し、応答 (該当する場合) を応答キューに非同期に送信します。これらの JMS バインディングのいずれかを使用するインポートは、メッセージを構築して JMS キューに送信し、キューに応答 (該当する場合) が到着するのを監視します。

- JMS

- JMS バインディングを使用すれば、WebSphere 組み込み JMS プロバイダーにアクセスすることができます。

- 汎用 JMS

- 汎用 JMS バインディングでは、IBM 以外のベンダーのメッセージング・システムにアクセスできます。

- MQ JMS

- MQ JMS バインディングを使用すれば、WebSphere MQ メッセージング・システムの JMS サブセットにアクセスできます。ご使用のアプリケーションにとって JMS サブセットの機能で十分な場合、このバインディングを使用します。

- MQ

WebSphere MQ バインディングにより、MQ ネイティブ・アプリケーションと通信できるようになるため、これらのアプリケーションがサービス指向アーキテクチャ・フレームワークに組み込まれ、MQ 固有のヘッダー情報にアクセスできるようになります。MQ のネイティブ機能を使用する必要がある場合に、このバインディングを使用します。

- Enterprise Java Bean

Enterprise Java Bean (EJB) インポート・バインディングを使用すれば、SCA コンポーネントが、J2EE サーバー上で実行中の J2EE ビジネス・ロジックによって提供されたサービスを呼び出すことができます。

---

## エクスポートおよびインポート・バインディングの概要

エクスポートによってインテグレーション・モジュール内のサービスが外部クライアントに使用可能となり、インポートによってインテグレーション・モジュール内の SCA コンポーネントが外部サービスを呼び出すことができるようになります。エクスポートまたはインポートに関連付けられたバインディングは、プロトコル・メッセージとビジネス・オブジェクトとの関係を指定します。また、操作と障害を選択する方法も指定します。

## エクスポートによる情報のフロー

エクスポートは、インポートがワイヤリングされたコンポーネントを対象とする要求を受信します。要求の受信に試用される特定のトランスポート（例えば、HTTP）は、エクスポートに関連付けられたバインディングによって決まります。

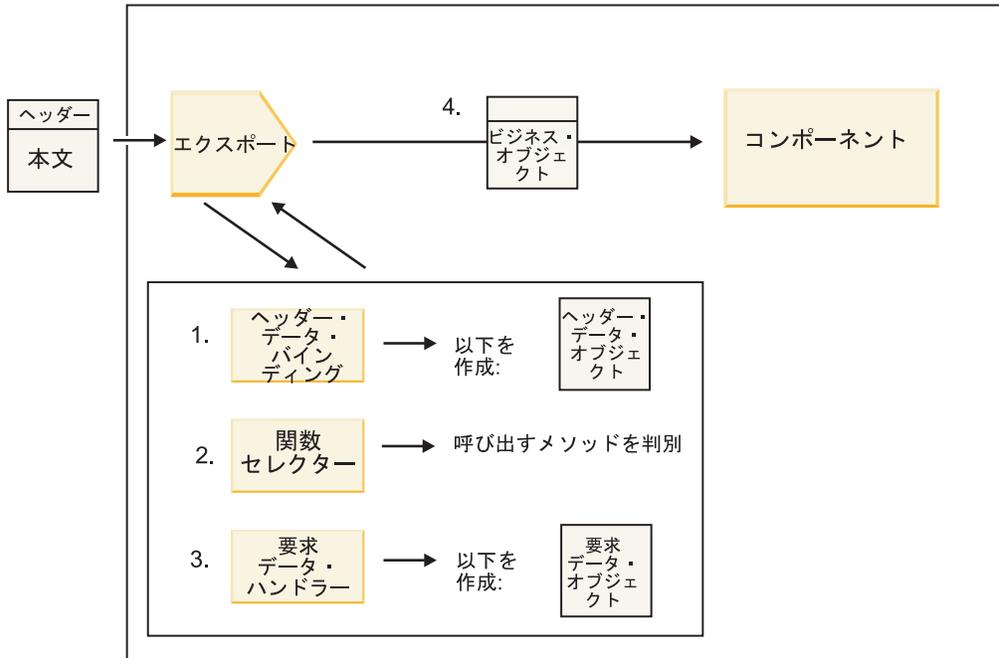


図7. エクスポートを経由したコンポーネントへの要求のフロー

エクスポートが要求を受信するとき、以下の一連のイベントが発生します。

1. WebSphere MQ バインディングの場合に限り、ヘッダー・データ・バインディングがプロトコル・ヘッダーをヘッダー・データ・オブジェクトに変換します。
2. 関数セレクターがプロトコル・メッセージからネイティブ・メソッド名を判別します。ネイティブ・メソッド名は、エクスポート構成により、エクスポートのインターフェース上の操作名にマップされています。
3. メソッドの要求データ・ハンドラーまたは要求データ・バインディングは、要求を要求ビジネス・オブジェクトに変換します。
4. エクスポートは要求ビジネス・オブジェクトでコンポーネント・メソッドを呼び出します。
  - HTTP エクスポート・バインディングと Web サービス・エクスポート・バインディングは、SCA コンポーネントに対して同期呼び出しを行います。
  - JMS、汎用 JMS、MQ JMS、および WebSphere MQ エクスポート・バインディングは、SCA コンポーネントに対して非同期呼び出しを行います。

エクスポートは、コンテキスト伝搬が有効になっている場合、プロトコル経由で受信したヘッダーおよびユーザー・プロパティーを伝搬できます。その後、エクスポートにワイヤリングされたコンポーネントは、これらのヘッダーおよびユーザー・プロパティーにアクセスできます。詳しくは、WebSphere Integration Developer インフォメーション・センターのトピック『伝搬』を参照してください。

これが両方向操作の場合、コンポーネントは応答を返します。

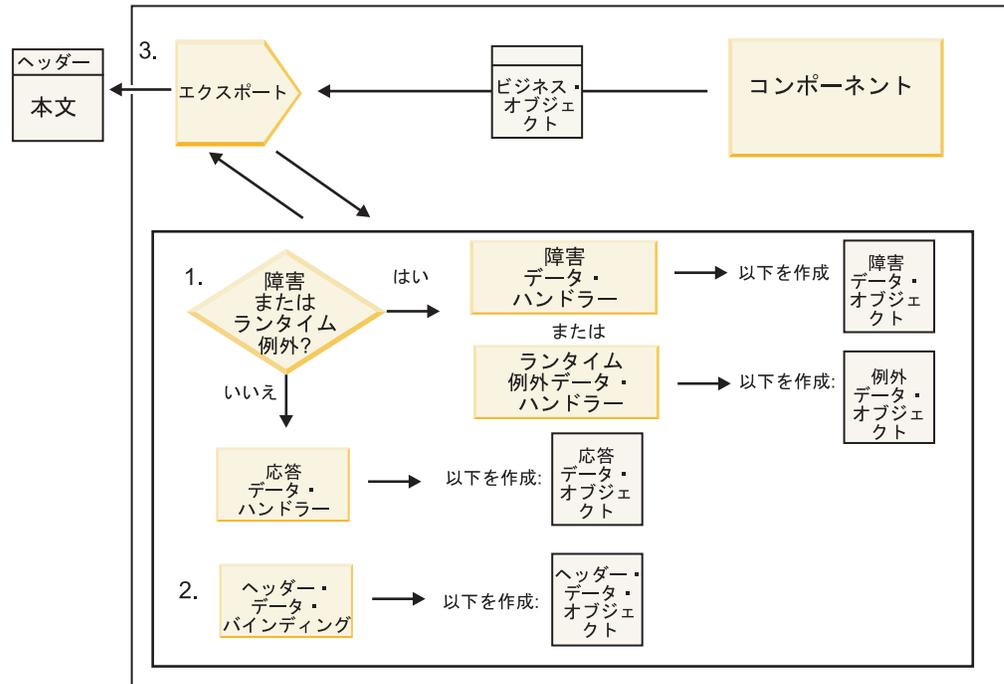


図8. エクスポートを経由して戻る応答のフロー

以下の一連のステップが発生します。

1. 通常の応答メッセージがエクスポート・バインディングによって受信されると、メソッドの応答データ・ハンドラーまたは応答データ・バインディングは、ビジネス・オブジェクトを応答に変換します。

応答が障害の場合、メソッドの障害データ・ハンドラーまたは障害データ・バインディングは障害を障害応答に変換します。

HTTP エクスポート・バインディングにのみ適用されることですが、応答がランタイム例外の場合、ランタイム例外データ・ハンドラー（構成済みの場合）が呼び出されます。

2. WebSphere MQ バインディングの場合に限り、ヘッダー・データ・バインディングがヘッダー・データ・オブジェクトをプロトコル・ヘッダーに変換します。
3. エクスポートはサービス応答をトランスポート経由で送信します。

## インポートによる情報のフロー

コンポーネントはインポートを使用して、モジュール外部のサービスに要求を送信します。要求は、関連付けられたバインディングによって決まる特定のトランスポートを使用して送信されます。

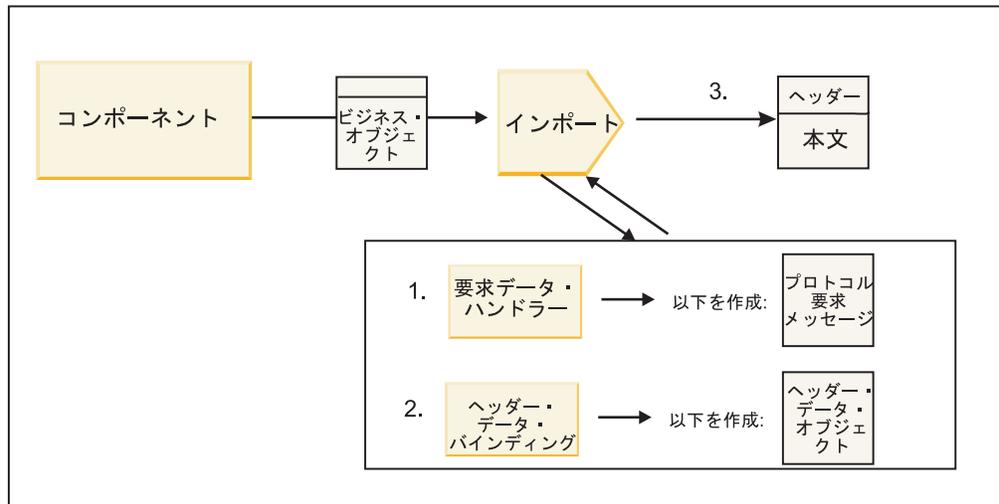


図9. インポートを経由したコンポーネントからサービスへのフロー

コンポーネントは、要求ビジネス・オブジェクトでインポートを呼び出します。

注:

- HTTP インポート・バインディング、Web サービス・インポート・バインディング、および EJB インポート・バインディングは、呼び出し側コンポーネントが同期方式で呼び出す必要があります。
- JMS、汎用 JMS、MQ JMS、および WebSphere MQ インポート・バインディングは、非同期方式で呼び出す必要があります。

コンポーネントがインポートを呼び出した後、以下の一連のイベントが発生します。

1. メソッドの要求データ・ハンドラーまたは要求データ・バインディングは、要求ビジネス・オブジェクトをプロトコル要求メッセージに変換します。
2. WebSphere MQ バインディングの場合に限り、メソッドのヘッダー・データ・バインディングがヘッダー・ビジネス・オブジェクトをプロトコル・ヘッダーに設定します。
3. インポートはトランスポートを経由して、サービス要求でサービスを呼び出します。

これが両方向操作の場合、サービスは応答を戻し、以下の一連のステップが発生します。

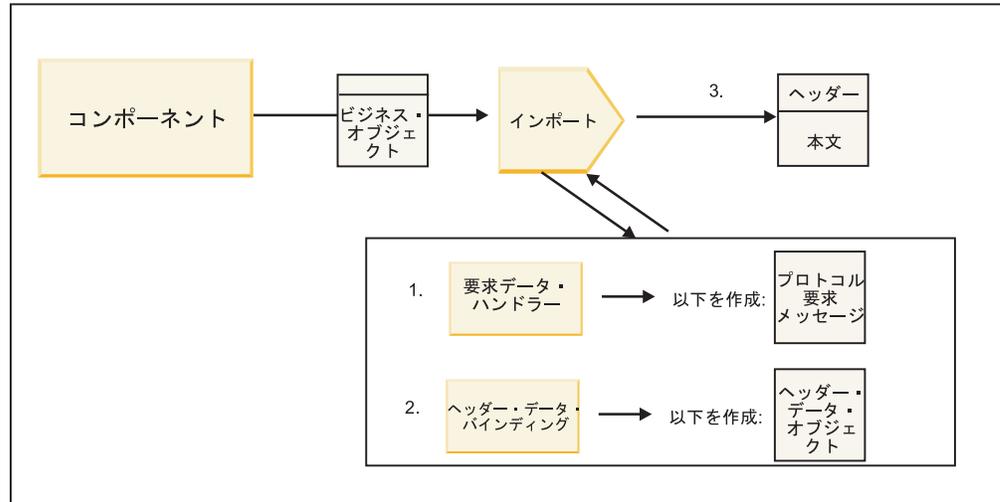


図 10. インポートを経由して戻る応答のフロー

1. WebSphere MQ バインディングの場合に限り、ヘッダー・データ・バインディングがプロトコル・ヘッダーをヘッダー・データ・オブジェクトに変換します。
2. 応答が障害かどうかを判別します。
  - 応答が障害の場合、障害セクターは障害を検査して、どの WSDL 障害にマップするかを決定します。次にメソッドの障害データ・ハンドラーは、障害を障害応答に変換します。
  - 応答がランタイム例外の場合、ランタイム例外データ・ハンドラー (構成済みの場合) が呼び出されます。
3. メソッドの応答データ・ハンドラーまたは応答データ・バインディングは、応答を応答ビジネス・オブジェクトに変換します。
4. インポートは応答ビジネス・オブジェクトをコンポーネントに返します。

#### 関連情報



伝搬

## エクスポートおよびインポート・バインディング構成

エクスポートおよびインポート・バインディングの重要な側面の 1 つはデータ・フォーマットの変換です。これによって、データをネイティブ・ワイヤー・フォーマットからビジネス・オブジェクトにマップ (非直列化) する方法、またはビジネス・オブジェクトからネイティブ・ワイヤー・フォーマットにマップ (直列化) する方法が指定されます。エクスポートに関連したバインディングでは、データに対して実行すべき操作を指示する関数セクターも指定できます。エクスポートまたはインポートに関連したバインディングでは、処理中に発生する障害の処理方法を指示することができます。

さらに、トランスポート固有情報をバインディングに指定できます。例えば HTTP バインディングでは、エンドポイント URL を指定します。詳しくは、WebSphere Integration Developer インフォメーション・センターを参照してください。例えば

HTTP バインディングの場合、トランスポート固有の情報は『HTTP インポート・バインディングの生成』トピックと『HTTP エクスポート・バインディングの生成』に記載されています。

#### 関連情報

-  [HTTP インポート・バインディングの生成](#)
-  [HTTP エクスポート・バインディングの生成](#)
-  [Web サービス・バインディングの生成](#)
-  [JMS バインディングの操作](#)
-  [MQ JMS バインディングの操作](#)
-  [MQ バインディングの操作](#)
-  [汎用 JMS バインディングの操作](#)

## インポートおよびエクスポートでのデータ・フォーマット変換

WebSphere Integration Developer でエクスポートまたはインポート・バインディングを構成するときには、構成プロパティの 1 つとしてバインディングで使用するデータ・フォーマットを指定します。

- エクスポート・バインディングでは、クライアント・アプリケーションが要求を SCA コンポーネントに送信し、応答を SCA コンポーネントから受信しますが、この場合、ネイティブ・データのフォーマットを指示します。システムはこのフォーマットに基づいて、ネイティブ・データからビジネス・オブジェクト (SCA コンポーネントによって使用される) に変換し、逆にビジネス・オブジェクトをネイティブ・データ (クライアント・アプリケーションへの応答) に変換するための適切なデータ・ハンドラーまたはデータ・バインディングを選択します。
- インポート・バインディングでは、SCA コンポーネントが要求をモジュール外部のサービスに送信し、応答をモジュール外部のサービスから受信しますが、この場合、ネイティブ・データのフォーマットを指示します。システムはこのフォーマットに基づいて、ビジネス・オブジェクトからネイティブ・データへ、およびその逆に変換するための適切なデータ・ハンドラーまたはデータ・バインディングを選択します。

WebSphere Process Server には事前定義された一連のデータ・フォーマットと、そのフォーマットをサポートする対応データ・ハンドラーまたはデータ・バインディングが提供されています。また、固有のカスタム・データ・ハンドラーを作成して、そのデータ・ハンドラーのデータ・フォーマットを登録することもできます。詳しくは、WebSphere Integration Developer インフォメーション・センターのトピック『データ・ハンドラーの開発』を参照してください。

- データ・ハンドラーはプロトコルに中立的で、1 つのフォーマットから別のフォーマットにデータを変換します。WebSphere Process Server では、データ・ハンドラーはネイティブ・データ (XML、CSV、および COBOL) からビジネス・オブジェクトへ、およびビジネス・オブジェクトからネイティブ・データへの変換を行うのが標準的です。これらはプロトコルに中立的なため、いろいろなエクス

ポートおよびインポート・バインディングで同じデータ・ハンドラーを再利用することができます。例えば、同一の XML データ・ハンドラーを HTTP エクスポートまたはインポート・バインディング、あるいは JMS エクスポートまたはインポート・バインディングで使用できます。

- データ・バインディングもネイティブ・データからビジネス・オブジェクトへの変換 (およびその逆) を行いますが、これらはプロトコルに固有です。例えば、HTTP データ・バインディングは HTTP エクスポートまたは HTTP インポート・バインディングのみで使用できます。データ・ハンドラーとは異なり、HTTP データ・バインディングは MQ エクスポートまたは MQ インポート・バインディングでは再利用できません。

前に述べたように、必要に応じてカスタム・データ・ハンドラーを作成することができます。また、カスタム・データ・バインディングを作成することもできますが、カスタム・データ・ハンドラーは複数のバインディングで使用できるため、カスタム・データ・ハンドラーを作成することをお勧めします。

### 関連情報

 データ・ハンドラーの開発

## データ・ハンドラー

データ・ハンドラーは、プロトコルに中立的な方法でデータ・フォーマットを変換するために、エクスポートおよびインポート・バインディングに対して構成されます。いくつかのデータ・ハンドラーが製品の一部として提供されていますが、必要に応じて固有のデータ・ハンドラーを作成することもできます。データ・ハンドラーをエクスポートまたはインポート・バインディングに関連付けるレベルは 2 つあります。一方のレベルでは、エクスポートまたはインポートのインターフェースにあるすべての操作に関連付けることが可能です。もう一方のレベルでは、要求または応答に対する特定の操作に関連付けることができます。

## 事前定義データ・ハンドラー

使用するデータ・ハンドラーの指定は、WebSphere Integration Developer を使用して行います。

データ変換用のデータ・ハンドラーは、以下のケースを除いて指定できます。

- JMS StreamMessage および JMS MapMessage を変換する場合。これらのタイプはデータ・ハンドラーではサポートされないため、カスタムのデータ・バインディングを記述する必要があります。
- 事前定義されたデータ・バインディングが製品にもともと付属するが、対応するデータ・ハンドラーがない場合。例えば、WrappedBytes のデータ・バインディングは提供されていますが、対応するデータ・ハンドラーはありません。

使用できるように事前定義されたデータ・ハンドラーを以下の表で示します。表には、各データ・ハンドラーがインバウンドおよびアウトバウンドのデータを変換する方法についても記載されています。

注: 注記されていない限り、これらのデータ・ハンドラーは、JMS、汎用 JMS、MQ JMS、WebSphere MQ、および HTTP バインディングで使用できます。

詳しくは、WebSphere Integration Developer インフォメーション・センターのトピック

ク『データ・ハンドラー』を参照してください。

表 11. 事前定義データ・ハンドラー

データ・ハンドラー	ネイティブ・データからビジネス・オブジェクトへ	ビジネス・オブジェクトからネイティブ・データへ
ATOM	ATOM フィードを ATOM フィード・ビジネス・オブジェクトに解析します。	ATOM フィード・ビジネス・オブジェクトを ATOM フィードに直列化します。
Delimited	区切り文字で区切られているデータをビジネス・オブジェクトに解析します。	ビジネス・オブジェクトを区切り文字で区切られているデータ (CSV など) に直列化します。
Fixed Width	固定長データをビジネス・オブジェクトに解析します。	ビジネス・オブジェクトを固定長データに直列化します。
WTX により処理	データ・フォーマット変換を WebSphere Transformation Extender (WTX) に委任します。WTX マップ名は、データ・ハンドラーから派生します。	データ・フォーマット変換を WebSphere Transformation Extender (WTX) に委任します。WTX マップ名は、データ・ハンドラーから派生します。
WTX 呼び出し側により処理	データ・フォーマット変換を WebSphere Transformation Extender (WTX) に委任します。WTX マップ名は、ユーザーが提供します。	データ・フォーマット変換を WebSphere Transformation Extender (WTX) に委任します。WTX マップ名は、ユーザーが提供します。
JAXB	JAXB 仕様を使用して Java Bean をビジネス・オブジェクトに変換します。	JAXB 仕様を使用してビジネス・オブジェクトを Java Bean に変換します。
JAXWS 注: JAXWS データ・ハンドラーは、EJB バインディングでのみ使用できます。	応答 Java オブジェクトまたは例外 Java オブジェクトを応答ビジネス・オブジェクトに変換します。	ビジネス・オブジェクトを出力 Java メソッド・パラメーターに変換します。
JSON	JSON データをビジネス・オブジェクトに解析します。	ビジネス・オブジェクトを JSON データに直列化します。
SOAP	SOAP メッセージ (およびヘッダー) をビジネス・オブジェクトに解析します。	ビジネス・オブジェクトを SOAP メッセージに直列化します。
XML	XML データをビジネス・オブジェクトに解析します。	ビジネス・オブジェクトを XML データに直列化します。
UTF8XMLDataHandler	UTF-8 にエンコードされた XML データをビジネス・オブジェクトに解析します。	メッセージの送信時に、ビジネス・オブジェクトを UTF-8 にエンコードされた XML データに直列化します。

## データ・ハンドラーの作成

データ・ハンドラーの作成に関する詳細情報については、WebSphere Integration Developer インフォメーション・センターのトピック『データ・ハンドラーの開発』を参照してください。

### 関連情報

 データ・ハンドラー

 データ・ハンドラーの開発

## データ・バインディング

データ・バインディングは、データ・フォーマットを変換するために、エクスポートおよびインポート・バインディングに対して構成されます。データ・バインディングは、それぞれのプロトコルに固有です。いくつかのデータ・バインディングが製品の一部として提供されていますが、必要に応じて固有のデータ・バインディングを作成することもできます。データ・バインディングをエクスポートまたはインポート・バインディングに関連付けるレベルは 2 つあります。一方のレベルでは、エクスポートまたはインポートのインターフェースにあるすべての操作に関連付けることが可能です。もう一方のレベルでは、要求または応答に対する特定の操作に関連付けることができます。

使用するデータ・バインディングの指定または固有のデータ・バインディングの作成は、WebSphere Integration Developer を使用して行います。データ・バインディングの作成については、WebSphere Integration Developer インフォメーション・センターの『JMS、MQ JMS、および汎用 JMS データ・バインディングの概要』セクションを参照してください。

## JMS バインディング

次の表は、以下のバインディングで使用できるデータ・バインディングの一覧です。

- JMS バインディング
- 汎用 JMS バインディング
- WebSphere MQ JMS バインディング

また表には、データ・バインディングが実行するタスクの説明も記載されています。

表 12. JMS バインディング用の事前定義データ・バインディング

データ・バインディング	ネイティブ・データからビジネス・オブジェクトへ	ビジネス・オブジェクトからネイティブ・データへ
直列化 Java オブジェクト	Java 直列化オブジェクトをビジネス・オブジェクト (WSDL で入力または出力タイプとしてマップされているビジネス・オブジェクト) に変換します。	ビジネス・オブジェクトを JMS オブジェクト・メッセージの Java 直列化オブジェクトに直列化します。

表 12. JMS バインディング用の事前定義データ・バインディング (続き)

データ・バインディング	ネイティブ・データからビジネス・オブジェクトへ	ビジネス・オブジェクトからネイティブ・データへ
ラップされたバイト	着信 JMS バイト・メッセージからバイトを抽出して、これらを JMSBytesBody ビジネス・オブジェクトにラップします。	JMSBytesBody ビジネス・オブジェクトからバイトを抽出して、これらを発信 JMS バイト・メッセージにラップします。
ラップされたマップ項目	着信 JMS マップ・メッセージの各項目の名前、値、およびタイプ情報を抽出して、MapEntry ビジネス・オブジェクトのリストを作成します。次に、そのリストを JMSMapBody ビジネス・オブジェクトにラップします。	JMSMapBody ビジネス・オブジェクトの MapEntry リストから名前、値、およびタイプ情報を抽出して、発信 JMS マップ・メッセージ内に対応する項目を作成します。
ラップされたオブジェクト	着信 JMS オブジェクト・メッセージからオブジェクトを抽出して、これを JMSObjectBody ビジネス・オブジェクトにラップします。	JMSObjectBody ビジネス・オブジェクトからオブジェクトを抽出して、これを発信 JMS オブジェクト・メッセージにラップします。
ラップされたテキスト	着信 JMS テキスト・メッセージからテキストを抽出して、これを JMSTextBody ビジネス・オブジェクトにラップします。	JMSTextBody ビジネス・オブジェクトからテキストを抽出して、これを発信 JMS テキスト・メッセージにラップします。

## WebSphere MQ バインディング

次の表は、WebSphere MQ で使用できるデータ・バインディングの一覧と、データ・バインディングが実行するタスクを説明したものです。

表 13. WebSphere MQ バインディング用の事前定義データ・バインディング

データ・バインディング	ネイティブ・データからビジネス・オブジェクトへ	ビジネス・オブジェクトからネイティブ・データへ
直列化 Java オブジェクト	着信メッセージからの Java 直列化オブジェクトをビジネス・オブジェクト (WSDL で入力または出力タイプとしてマップされているビジネス・オブジェクト) に変換します。	ビジネス・オブジェクトを出力メッセージの Java 直列化オブジェクトに変換します。
ラップされたバイト	構造化されていない MQ バイト・メッセージからバイトを抽出して、これらを JMSTextBody ビジネス・オブジェクトにラップします。	JMSTextBody ビジネス・オブジェクトからテキストを抽出して、テキストを構造化されていない発信 MQ バイト・メッセージにラップします。

表 13. WebSphere MQ バインディング用の事前定義データ・バインディング (続き)

データ・バインディング	ネイティブ・データからビジネス・オブジェクトへ	ビジネス・オブジェクトからネイティブ・データへ
ラップされたテキスト	構造化されていない MQ テキスト・メッセージからテキストを抽出して、これを JMSTextBody ビジネス・オブジェクトにラップします。	JMSTextBody ビジネス・オブジェクトからテキストを抽出して、これを構造化されていない MQ テキスト・メッセージにラップします。
ラップされたストリーム項目	着信 JMS ストリーム・メッセージの各項目の名前およびタイプ情報を抽出して、StreamEntry ビジネス・オブジェクトのリストを作成します。次に、そのリストを JMSStreamBody ビジネス・オブジェクトにラップします。	JMSStreamBody ビジネス・オブジェクトの StreamEntry リストから名前およびタイプ情報を抽出して、発信 JMSStreamMessage 内に対応する項目を作成します。

234 ページの表 13 にリストされたデータ・バインディングの他、WebSphere MQ はヘッダー・データ・バインディングも使用します。詳しくは、WebSphere Integration Developer インフォメーション・センターを参照してください。

## HTTP バインディング

次の表は、HTTP で使用できるデータ・バインディングの一覧と、データ・バインディングが実行するタスクを説明したものです。

表 14. HTTP バインディング用の事前定義データ・バインディング

データ・バインディング	ネイティブ・データからビジネス・オブジェクトへ	ビジネス・オブジェクトからネイティブ・データへ
ラップされたバイト	着信 HTTP メッセージの本体からバイトを抽出して、これらを HTTPBytes ビジネス・オブジェクトにラップします。	HTTPBytes ビジネス・オブジェクトからバイトを抽出して、これらを発信 HTTP メッセージの本体に追加します。

その他のデータ・フォーマットについては、カスタムのデータ・バインディングおよびデータ・ハンドラーを使用する必要があります。

### 関連情報



JMS、MQ JMS、汎用 JMS バインディングの概要



カスタム MQ データ・バインディングの例

## エクスポート・バインディングでの関数セクター

関数セクターは、要求メッセージのデータに対して実行すべき操作を指示するために使用します。関数セクターは、エクスポート・バインディングの一部として構成されます。

インターフェースを公開する SCA エクスポートを例として考えてみます。このインターフェースには、作成および更新の 2 つの操作が組み込まれています。エクスポートの JMS バインディングは、キューから読み取られます。

メッセージがキューに到着すると、エクスポートには関連するデータが渡されますが、ワイヤリングされたコンポーネントで、エクスポートのインターフェースからの操作を呼び出すべきかの情報は渡されません。その操作は、関数セクターとエクスポート・バインディング構成によって決まります。

関数セクターはネイティブの関数名 (メッセージを送信したクライアント・システムの関数名) を返します。ネイティブの関数名は、エクスポートが関連付けられたインターフェースの操作名または関数名にマップされます。例えば以下の図で、関数セクターは着信メッセージからネイティブの関数名 (CRT) を返し、ネイティブの関数名は作成操作にマップされ、ビジネス・オブジェクトは作成操作を持つ SCA コンポーネントに送信されます。

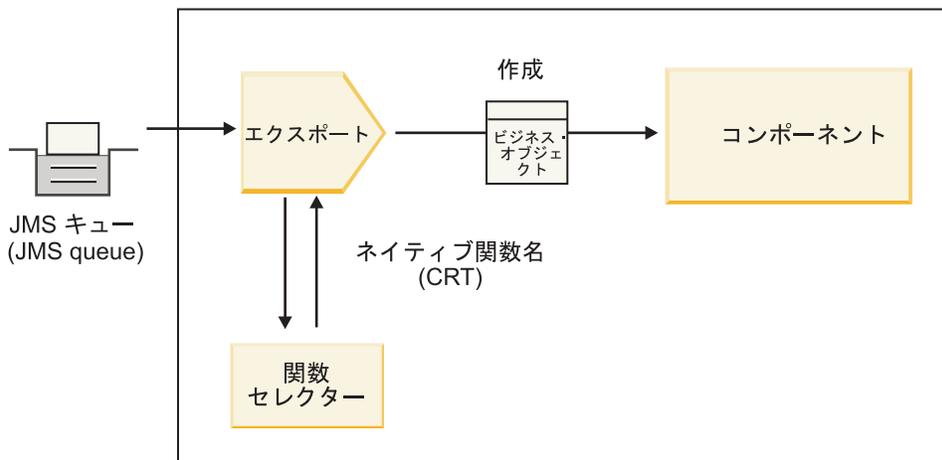


図 11. 関数セクター

インターフェースに 1 つの操作しかない場合には、関数セクターを指定する必要はありません。

複数の関数セクターがプリパッケージされています。以降のセクションで、これらの関数セクターをリストします。

## JMS バインディング

次の表は、以下のバインディングで使用できる関数セクターの一覧です。

- JMS バインディング
- 汎用 JMS バインディング
- WebSphere MQ JMS バインディング

表 15. JMS バインディング用の事前定義関数セクター

関数セクター	説明
単純な JMS データ・バインディング用 JMS 関数セクター	メッセージの JMSType プロパティを使用して操作名を選択します。

表 15. JMS バインディング用の事前定義関数セレクター (続き)

関数セレクター	説明
JMS ヘッダー・プロパティ関数セレクター	ヘッダーの JMS ストリング・プロパティ、TargetFunctionName の値を返します。
JMS サービス・ゲートウェイ関数セレクター	クライアントによって設定された JMSReplyTo プロパティを調べ、要求が片方向操作と両方向操作のどちらであるかを判別します。

## WebSphere MQ バインディング

次の表は、WebSphere MQ バインディングで使用できる関数セレクターの一覧です。

表 16. WebSphere MQ バインディング用の事前定義関数セレクター

関数セレクター	説明
MQ handleMessage 関数セレクター	handleMessage を値として返します。この値は、エクスポート・メソッド・バインディングを使用してインターフェースの操作名にマップされています。
MQ が JMS デフォルト関数セレクターを使用	MQRFH2 ヘッダーのフォルダーの TargetFunctionName プロパティからネイティブ操作を読み取ります。
MQ がメッセージ本体のフォーマットをネイティブ関数として使用	最後のヘッダーの Format フィールドを検索し、そのフィールドをストリングとして返します。
MQ タイプ関数セレクター	MQRFH2 ヘッダーで検出した Msd、Set、Type、および Format プロパティを含む URL を取得して、エクスポート・バインディング内にメソッドを作成します。
MQ サービス・ゲートウェイ関数セレクター	MQMD ヘッダー内の MsgType プロパティを使用して操作名を判断します。

## HTTP バインディング

次の表は、HTTP バインディングで使用できる関数セレクターの一覧です。

表 17. HTTP バインディング用の事前定義関数セレクター

関数セレクター	説明
TargetFunctionName ヘッダーに基づく HTTP 関数セレクター	クライアントからの TargetFunctionName HTTP ヘッダー・プロパティを使用して、実行時にエクスポートから呼び出す操作を判断します。
URL および HTTP メソッドに基づく HTTP 関数セレクター	クライアントからの HTTP メソッドに付加された URL の相対パスを使用して、エクスポートに定義されたネイティブ操作を判断します。

表 17. HTTP バインディング用の事前定義関数セクター (続き)

関数セクター	説明
操作名が設定された URL に基づく HTTP サービス・ゲートウェイ関数セクター	要求 URL に「operationMode = oneWay」が付加されている場合、その URL に基づいて呼び出すメソッドを判断します。

注: WebSphere Integration Developer を使用して固有の関数セクターを作成することもできます。関数セクターの作成に関する詳細情報については、WebSphere Integration Developer インフォメーション・センターを参照してください。例えば、WebSphere MQ バインディング用の関数セクターの作成に関する説明は、『MQ 関数セクターの概要』に記載されています。

### 関連情報

 MQ 関数セクター

## 障害の処理

ご使用のインポート・バインディングおよびエクスポート・バインディングについて、障害データ・ハンドラーを指定することによって、処理中に発生する障害 (ビジネス例外など) を処理するように構成できます。障害データ・ハンドラーは、3 つのレベルでセットアップできます。具体的には、障害データ・ハンドラーを障害に関連付けるか、操作に関連付けるか、またはバインディングを使用するすべての操作を対象に関連付けます。

障害データ・ハンドラーは障害データを処理し、エクスポート・バインディングまたはインポート・バインディングによって送信される正しいフォーマットに変換します。

- エクスポート・バインディングでは、障害データ・ハンドラーはコンポーネントから送信された例外ビジネス・オブジェクトを、クライアント・アプリケーションによって使用できる応答メッセージに変換します。
- インポート・バインディングでは、障害データ・ハンドラーはサービスから送信された障害データまたは応答メッセージを、SCA コンポーネントによって使用できる例外ビジネス・オブジェクトに変換します。

インポート・バインディングの場合、バインディングは障害セクターを呼び出します。障害セクターは、応答メッセージが通常応答、ビジネス障害、またはランタイム例外のいずれであるかを判別します。

障害データ・ハンドラーは、特定の障害、操作、およびバインディングを使用するすべての操作について指定することができます。

- 障害データ・ハンドラーが 3 つすべてのレベルで設定されている場合、特定の障害に関連付けられたデータ・ハンドラーが呼び出されます。
- 障害データ・ハンドラーが操作およびバインディングのレベルで設定されている場合、操作に関連付けられたデータ・ハンドラーが呼び出されます。

WebSphere Integration Developer での障害処理の指定には、2 つのエディターが使用されます。インターフェース・エディターは、操作に障害があるかどうかを示すのに使用されます。このインターフェースでバインディングが生成された後、プロパ

ティー・ビューのエディターによって、障害をどのように処理するかを構成できます。詳しくは、WebSphere Integration Developer インフォメーション・センターのトピック『障害セレクター』を参照してください。

### 関連情報

 障害セレクター

## エクスポート・バインディングでの障害の処理方法

クライアント・アプリケーションからの要求の処理中に障害が発生したとき、エクスポート・バインディングは障害情報をクライアントに返すことができます。障害を処理してクライアントに返す方法を指定するようにエクスポート・バインディングを構成します。

エクスポート・バインディングの構成は WebSphere Integration Developer を使用して行います。

要求を処理するとき、クライアントは要求でエクスポートを呼び出し、エクスポートは SCA コンポーネントを呼び出します。要求の処理中に、SCA コンポーネントはビジネス応答を返すか、またはサービス・ビジネス例外あるいはサービス・ランタイム例外をスローすることができます。これが発生すると、エクスポート・バインディングは例外を障害メッセージに変換して、クライアントに送信します。これについて以下の図で示し、その次のセクションで説明します。

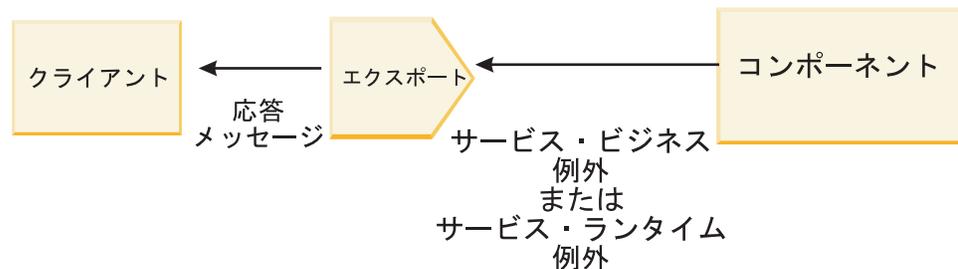


図 12. 障害情報をエクスポート・バインディングを経由してコンポーネントからクライアントに送信する方法

障害を処理するカスタムのデータ・ハンドラーまたはデータ・バインディングを作成することができます。

## ビジネス障害

ビジネス障害とは、処理中に発生するビジネス・エラーまたはビジネス例外です。

createCustomer 操作を持つ以下のインターフェースについて考えてみます。この操作には、CustomerAlreadyExists および MissingCustomerId の 2 つのビジネス障害が定義されています。

## 操作

操作およびパラメーター

	名前	タイプ
▼ createCustomer		
📥 入力	input	CustomerInfo
📤 出力	output	CustomerInfo
🚫 障害	Customer Already Exists	Customer Already ExistsBO
🚫 障害	MissingCustomerID	MissingCustomerIDBO

図 13. 2 つの障害を持つインターフェース

この例で、クライアントが顧客を作成する要求を（この SCA コンポーネントに）送信したときにその顧客が既に存在する場合、コンポーネントは CustomerAlreadyExists 障害をエクスポートにスローします。エクスポートはこのビジネス障害を、呼び出し側のクライアントに伝搬して返す必要があります。エクスポートはこれを行うために、エクスポート・バインディングに設定された障害データ・ハンドラーを使用します。

ビジネス障害がエクスポート・バインディングによって受信されると、以下の処理が行われます。

1. バインディングは、どの障害を障害データ・ハンドラーを呼び出して処理させるかを決定します。サービス・ビジネス例外に障害名が含まれている場合、その障害について設定されているデータ・ハンドラーが呼び出されます。サービス・ビジネス例外が障害名を含まない場合、障害名は障害タイプをマッチングすることによって導き出されます。
2. バインディングは、サービス・ビジネス例外からのデータ・オブジェクトを使用して、障害データ・ハンドラーを呼び出します。
3. 障害データ・ハンドラーは障害データ・オブジェクトを応答メッセージに変換し、これをエクスポート・バインディングに返します。
4. エクスポートは応答メッセージをクライアントに返します。

サービス・ビジネス例外に障害名が含まれている場合、その障害について設定されているデータ・ハンドラーが呼び出されます。サービス・ビジネス例外が障害名を含まない場合、障害名は障害タイプをマッチングすることによって導き出されま

## ランタイム例外

ランタイム例外とは、要求の処理中に SCA アプリケーション内で発生する、ビジネス障害に対応しない例外のことです。ビジネス障害とは異なり、ランタイム例外はインターフェースで定義されません。

シナリオによっては、これらのランタイム例外をクライアント・アプリケーションに伝搬して、クライアント・アプリケーションが適切なアクションを実行できるようにしたい場合もあります。

例えば、クライアントが顧客を作成する要求を（SCA コンポーネントに）送信したとき、要求の処理中に権限エラーが発生した場合、コンポーネントはランタイム例

外をスローします。このランタイム例外は呼び出し側のクライアントに戻すよう伝搬させて、クライアントが権限に関して適切なアクションを実行できるようにする必要があります。これはランタイム例外データ・ハンドラーをエクスポート・バインディングに構成することによって実現できます。

注: ランタイム例外データ・ハンドラーは、HTTP バインディングでのみ構成できます。

ランタイム例外の処理は、ビジネス障害の処理と似ています。ランタイム例外データ・ハンドラーが設定されている場合、以下の処理が実行されます。

1. エクスポート・バインディングはサービス・ランタイム例外について適切なデータ・ハンドラーを呼び出します。
2. データ・ハンドラーは障害データ・オブジェクトを応答メッセージに変換し、これをエクスポート・バインディングに返します。
3. エクスポートは応答メッセージをクライアントに返します。

障害処理およびランタイム例外処理はオプションです。障害またはランタイム例外を呼び出し側のクライアントに伝搬させたくない場合、障害データ・ハンドラーまたはランタイム例外データ・ハンドラーを構成しないでください。

## インポート・バインディングでの障害の処理方法

コンポーネントはインポートを使用して、要求をモジュール外部のサービスに送信します。要求の処理中に障害が発生すると、サービスはインポート・バインディングに障害を返します。障害を処理してコンポーネントに返す方法は、インポート・バインディングを構成して指定することができます。

インポート・バインディングの構成は WebSphere Integration Developer を使用して行います。障害データ・ハンドラー (または障害データ・バインディング) を指定できますが、障害セクターを指定することもできます。

## 障害データ・ハンドラー

要求を処理するサービスがインポート・バインディングに障害情報を送信するときには、例外の形式または障害データが含まれる応答メッセージが使用されます。

インポート・バインディングはサービス例外または応答メッセージをサービス・ビジネス例外またはサービス・ランタイム例外に変換します。これについて以下の図で示し、その次のセクションで説明します。

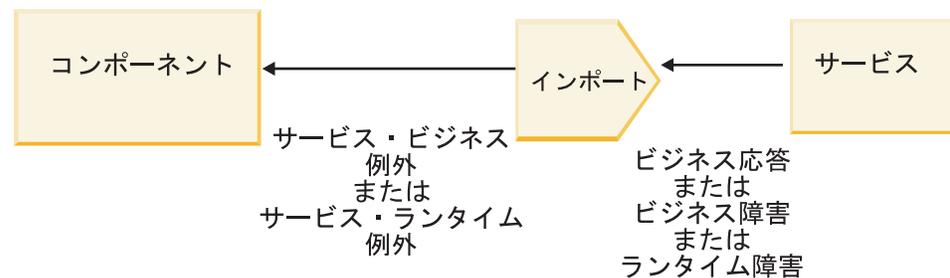


図 14. 障害情報をインポートを経由してサービスからコンポーネントに送信する方法

障害を処理するカスタムのデータ・ハンドラーまたはデータ・バインディングを作成することができます。

## 障害セレクター

インポート・バインディングを構成するとき、障害セレクターを指定することができます。障害セレクターは、インポートの応答が実際の応答か、ビジネス例外か、またはランタイム障害であるかを判別します。またこれは、応答の本体またはヘッダーからネイティブの障害名を判別します。ネイティブの障害名は、バインディング構成によって、関連するインターフェースの障害名にマップされます。

JMS、MQ JMS、汎用 JMS、WebSphere MQ、および HTTP インポートでは、プリパッケージされた 2 つのタイプの障害セレクターを使用できます。

表 18. プリパッケージされている障害セレクター

障害セレクター・タイプ	説明
ヘッダー・ベース	着信応答メッセージのヘッダーに基づいて、応答メッセージがビジネス障害、ランタイム例外、または通常のメッセージのいずれであるかを判別します。
SOAP	応答 SOAP メッセージが通常応答、ビジネス障害、またはランタイム例外のいずれであるかを判別します。

以下は、ヘッダー・ベースの障害セレクターおよび SOAP 障害セレクターの例です。

- ヘッダー・ベースの障害セレクター

アプリケーションで、着信メッセージがビジネス障害であることを示す場合、着信メッセージにはビジネス障害に対応する以下の 2 つのヘッダーが含まれます。

```
Header name = FaultType, Header value = Business  
Header name = FaultName, Header value = <user defined native fault name>
```

アプリケーションで、着信応答メッセージがランタイム例外であることを示す場合、着信メッセージには以下に示す 1 つのヘッダーが含まれます。

```
Header name = FaultType, Header value = Runtime
```

- SOAP 障害セレクター

ビジネス障害は、SOAP メッセージの一部として送信できます。それには、SOAP メッセージに以下のカスタム SOAP ヘッダーを設定します。この場合の障害名は、「CustomerAlreadyExists」です。

```
<ibmSoap:BusinessFaultName  
xmlns:ibmSoap="http://www.ibm.com/soap">CustomerAlreadyExists  
</ibmSoap:BusinessFaultName>
```

障害セレクターはオプションです。障害セレクターを指定しないと、インポート・バインディングは応答のタイプを判別できません。そのため、バインディングは応答をビジネス応答として扱い、応答データ・ハンドラーまたはデータ・バインディングを呼び出します。

カスタム障害セレクターを作成することができます。カスタム障害セレクターを作成するステップは、WebSphere Integration Developer インフォメーション・センターのトピック『カスタム障害セレクターの開発』にあります。

## ビジネス障害

要求の処理にエラーがあると、ビジネス障害が発生することがあります。例えば、顧客を作成する要求を送信したときにその顧客が既に存在する場合、サービスはインポート・バインディングにビジネス例外を送信します。

ビジネス例外がバインディングによって受信された場合、処理ステップは障害セレクターがバインディングに設定されているかどうかによって依存します。

- 障害セレクターが設定されていない場合、バインディングは応答データ・ハンドラーまたは応答データ・バインディングを呼び出します。
- 障害セレクターが設定されている場合、以下の処理が実行されます。
  1. インポート・バインディングは障害セレクターを呼び出して、応答がビジネス障害、ビジネス応答、またはランタイム障害のいずれであるかを判別します。
  2. 応答がビジネス障害である場合、インポート・バインディングは、ネイティブ障害名を提供するために障害セレクターを呼び出します。
  3. インポート・バインディングは、障害セレクターによって返されたネイティブ障害名に対応する WSDL 障害を判別します。
  4. インポート・バインディングは、この WSDL 障害用に構成された障害データ・ハンドラーを判別します。
  5. インポート・バインディングは障害データについて、この障害データ・ハンドラーを呼び出します。
  6. 障害データ・ハンドラーは障害データをデータ・オブジェクトに変換し、これをインポート・バインディングに返します。
  7. インポート・バインディングはデータ・オブジェクトおよび障害名を使用して、サービス・ビジネス例外オブジェクトを構成します。
  8. インポートはサービス・ビジネス例外オブジェクトをコンポーネントに返します。

## ランタイム例外

ランタイム例外は、サービスとの通信に問題があるときに発生することがあります。ランタイム例外の処理は、ビジネス例外の処理と似ています。障害セレクターが設定されている場合、以下の処理が実行されます。

1. インポート・バインディングは例外データについて適切なランタイム例外データ・ハンドラーを呼び出します。
2. ランタイム例外データ・ハンドラーは例外データをサービス・ランタイム例外オブジェクトに変換し、これをインポート・バインディングに返します。
3. インポートはサービス・ランタイム例外オブジェクトをコンポーネントに返します。

### 関連情報

 [カスタム障害セレクターの開発](#)

---

## EIS バインディング

エンタープライズ情報システム (EIS) のバインディングは、SCA コンポーネントと外部 EIS 間の接続を提供します。この通信を可能にするために使用する手段は、JCA 1.5 リソース・アダプターおよび WebSphere Adapters をサポートする EIS エクスポートおよび EIS インポートです。

SCA コンポーネントでは、外部 EIS との間でのデータの転送が必要になる場合があります。このような接続を必要とする SCA モジュールを作成する場合は、SCA コンポーネントに加えて、特定の外部 EIS との通信のために EIS バインディングを使用したインポートまたは EIS エクスポートを組み込む必要があります。

WebSphere Integration Developer のリソース・アダプターは、インポートまたはエクスポートのコンテキスト内で使用されます。インポートまたはエクスポートは外部サービス・ウィザードで開発しますが、これを開発するときにリソース・アダプターを組み込みます。アプリケーションが EIS システムのサービスを呼び出せるようにする EIS インポート、または EIS システムのアプリケーションが WebSphere Integration Developer で開発されたサービスを呼び出せるようにする EIS エクスポートは、特定のリソース・アダプターで作成されます。例えば、JD Edwards システムのサービスを呼び出すインポートを作成するには、JD Edwards アダプターを使用することになります。

外部サービス・ウィザードを使用すると、EIS バインディング情報が自動的に作成されます。また別のツールとして、バインディング情報を追加または変更するにはアSEMBリー・エディターを使用できます。詳しくは、WebSphere Integration Developer インフォメーション・センターを参照してください。

EIS バインディングが含まれる SCA モジュールをサーバーにデプロイすると、管理コンソールを使用して、バインディングに関する情報を表示したり、バインディングを構成したりできます。

### 関連概念

252 ページの『JMS バインディング』

Java Message Service (JMS) プロバイダーにより、Java Messaging Service API およびプログラミング・モデルに基づいたメッセージングが可能になります。JMS プロバイダーは、JMS 宛先への接続を作成し、メッセージを送受信するための J2EE 接続ファクトリーを提供します。

265 ページの『汎用 JMS バインディング』

汎用 JMS バインディングにより、サード・パーティーの JMS 1.1 準拠プロバイダーに対する接続が提供されます。汎用 JMS バインディングの操作は、JMS バインディングの操作と似ています。

277 ページの『WebSphere MQ JMS バインディング』

WebSphere MQ JMS バインディングは、WebSphere MQ JMS ベースのプロバイダーを使用する外部アプリケーションとの統合を提供します。

289 ページの『WebSphere MQ バインディング』

WebSphere MQ バインディングは、WebSphere MQ アプリケーションと Service Component Architecture (SCA) との接続を提供します。

### 関連タスク

WebSphere Process Server におけるイベント順序付けの有効化

関連情報

アダプターを使用したサービスの開発

## EIS バインディング: 全体像

EIS (エンタープライズ情報システム) バインディングを JCA リソース・アダプターと一緒に使用すると、エンタープライズ情報システム上のサービスにアクセスしたり、サービスを EIS で使用可能にすることができます。

以下は、Siebel システムと SAP システム間の接続情報を同期化する方法を説明する ContactSyncModule という名前の SCA モジュールの例です。

1. ContactSync という名前の SCA コンポーネントは、(Siebel Contact という名前の EIS アプリケーション・エクスポートを介して) Siebel への接続に対する変更を listen します。
2. ContactSync SCA コンポーネント自体は、SAP の接続情報を適宜更新するために、(EIS アプリケーション・インポートを介して) SAP アプリケーションを利用します。

接続情報の保管用に使用されるデータ構造は、Siebel システムと SAP システムでは異なるため、ContactSync SCA コンポーネントがマッピングを行う必要があります。

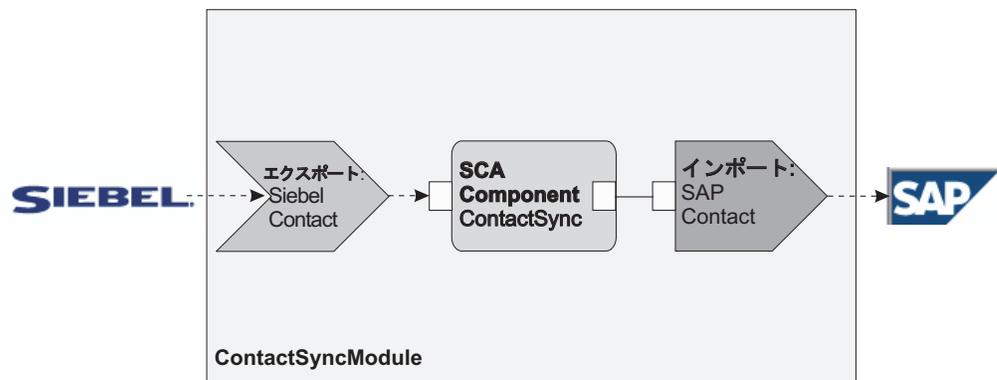


図 15. Siebel システムから SAP システムへのフロー

Siebel Contact エクスポートと SAP Contact インポートには、適切なリソース・アダプターが構成されます。

## EIS バインディングの主な特徴

EIS インポートとは、Service Component Architecture (SCA) モジュール内のコンポーネントが SCA モジュール外部で定義された EIS アプリケーションを使用できるようにする SCA インポートです。EIS インポートは、SCA コンポーネントから外部 EIS へのデータ転送に使用されます。EIS エクスポートは、外部 EIS から SCA モジュールへのデータ転送に使用されます。

## インポート

EIS インポートの役割は、SCA コンポーネントと外部 EIS システムの間のすき間を埋めることです。外部アプリケーションを EIS エクスポートとして取り扱うことができます。この場合、外部アプリケーションはそのデータを定期的な通知という形で送信します。

EIS インポートは、SCA コンポーネントがモジュール外部のアプリケーションを統一された形式で認識できるようにします。これによりコンポーネントは、SAP、Siebel、PeopleSoft などの外部 EIS と、一貫性のある SCA モデルを使用して通信できるようになります。

インポートのクライアント側には、EIS インポート・アプリケーションによって公開されるインターフェースがあります。ここには 1 つ以上のメソッドがあり、各メソッドがデータ・オブジェクトを引数および戻り値として受け入れます。実装側には、リソース・アダプターによって実装される共通クライアント・インターフェース (CCI) があります。

EIS インポートのランタイム実装は、クライアント側のインターフェースとこの CCI を接続します。インポートにより、インターフェース上のメソッドの呼び出しが CCI 上の呼び出しにマップされます。

バインディングは 3 つのレベルで作成されます (インターフェース・バインディングがそれに含まれるメソッド・バインディングを使用し、さらにそのメソッド・バインディングがデータ・バインディングを使用します)。

インターフェース・バインディングは、インポートのインターフェースを、アプリケーションを提供する EIS システムとの接続に関連付けます。これは、インターフェースによって表されるアプリケーション・セットが EIS の特定のインスタンスによって提供されていて、このインスタンスには接続を介してアクセスできるという事実を反映しています。バインディング・エレメントは、接続を作成するための十分な情報を指定したプロパティを持ちます (このプロパティは、`javax.resource.spi.ManagedConnectionFactory` インスタンスの一部です)。

メソッド・バインディングは、メソッドを、EIS システムとの特定の対話に関連付けます。J2C では、この対話の特徴は、`javax.resource.cci.InteractionSpec` インターフェース実装のプロパティ・セットによって記述されます。メソッド・バインディングの対話エレメントには、これらのプロパティとともにクラスの名前が含まれています。これにより、対話を実行するための十分な情報を提供します。メソッド・バインディングは、データ・バインディングを使用して、インターフェース・メソッドの引数と結果の EIS 表現へのマッピングを記述します。

EIS インポートのランタイム・シナリオを以下に示します。

1. インポート・インターフェースのメソッドが、SCA プログラミング・モデルを使用して呼び出されます。
2. EIS インポートに送信される要求には、メソッド名と引数が指定されています。
3. インポートは最初にインターフェース・バインディングの実装を作成し、次にインポート・バインディングのデータを使用して `ConnectionFactory` を作成し、この 2 つを関連付けます。したがって、インポートにより、インターフェース・バインディングの `setConnectionFactory` が呼び出されることとなります。

4. 呼び出されたメソッドと一致するメソッド・バインディングの実装が作成されず。
5. `javax.resource.cci.InteractionSpec` インスタンスが作成され、データが取り込まれます。次に、データ・バインディングを使用して、リソース・アダプターが認識できるフォーマットにメソッド引数がバインドされます。
6. CCI インターフェースを使用して対話が実行されます。
7. 呼び出しが戻されたら、データ・バインディングによって呼び出しの結果が作成され、結果が呼び出し元に戻されます。

## エクスポート

EIS エクスポートの役割は、SCA コンポーネントと外部 EIS 間のギャップを埋めることです。外部アプリケーションを EIS エクスポートとして取り扱うことができます。この場合、外部アプリケーションはそのデータを定期的な通知という形で送信します。EIS エクスポートは、EIS からの外部要求を `listen` するサブスクリプション・アプリケーションであると考えられます。EIS エクスポートを利用する SCA コンポーネントは、EIS エクスポートをローカル・アプリケーションとして認識します。

EIS エクスポートは、SCA コンポーネントがモジュール外部のアプリケーションを統一された形式で認識できるようにします。これによりコンポーネントは、SAP、Siebel、PeopleSoft などの EIS と、一貫性のある SCA モデルを使用して通信できるようになります。

エクスポートは、EIS から要求を受け取るリスナー実装を特徴としています。リスナーは、JMS エクスポート用の `javax.jms.MessageListener` インターフェースまたはリソース・アダプター固有のリスナー・インターフェースのいずれかを実装します。また、エクスポートには、エクスポートを介して EIS に公開されるインターフェースを実装するコンポーネントも含まれます。

EIS エクスポートのランタイム実装は、インターフェースを実装するコンポーネントにリスナーを接続します。エクスポートにより、EIS 要求が、コンポーネント上の該当する操作の呼び出しにマップされます。バインディングは 3 つのレベルで作成されます (リスナー・バインディングがそれに含まれるネイティブ・メソッド・バインディングを使用し、さらにそのネイティブ・メソッド・バインディングがデータ・バインディングを使用します)。

リスナー・バインディングは、要求を受け取るリスナーを、エクスポートを介して公開されるコンポーネントに関連付けます。エクスポート定義には、コンポーネントの名前が含まれています。これにより、ランタイムがコンポーネントを検索し、コンポーネントに要求を転送します。

ネイティブ・メソッド・バインディングは、ネイティブ・メソッドまたはリスナーが受け取ったイベント・タイプを、エクスポートを介して公開されたコンポーネントによって実装される操作に関連付けます。リスナーで呼び出されるメソッドとイベント・タイプの間には関係はなく、すべてのイベントはリスナーの 1 つ以上のメソッドを使用して受信されます。ネイティブ・メソッド・バインディングは、エクスポートで定義された関数セレクターを使用してインバウンド・データからネイティ

ブ・メソッド名を取り出し、データ・バインディングを使用して EIS のデータ・フォーマットをコンポーネントが認識できるフォーマットにバインドします。

EIS エクスポートのランタイム・シナリオを以下に示します。

1. EIS 要求は、リスナー実装のメソッドの呼び出しをトリガーします。
2. リスナーは エクスポートを検出して呼び出し、すべての呼び出し引数を渡します。
3. エクスポートは、リスナー・バインディングの実装を作成します。
4. エクスポートは、関数セクターをインスタンス化してリスナー・バインディング上に設定します。
5. エクスポートは、ネイティブ・メソッド・バインディングを初期化してリスナー・バインディングに追加します。各ネイティブ・メソッド・バインディングについて、データ・バインディングも初期化されます。
6. エクスポートは、リスナー・バインディングを呼び出します。
7. リスナー・バインディングは、エクスポートされたコンポーネントを検出し、関数セクターを使用してネイティブ・メソッド名を取得します。
8. この名前を使用して、ネイティブ・メソッド・バインディングが検出されます。その後、ネイティブ・メソッド・バインディングによってターゲット・コンポーネントが呼び出されます。

アダプターの対話スタイルでは、EIS エクスポート・バインディングで、ターゲット・コンポーネントを非同期方式 (デフォルト) または同期方式のどちらでも呼び出すことができます。

## リソース・アダプター

外部サービス・ウィザードを使用してインポートやエクスポートを開発する際に、リソース・アダプターを組み込みます。WebSphere Integration Developer に付属するリソース・アダプターは、CICS、IMS、JD Edwards、PeopleSoft、SAP、Siebel の各システムへのアクセスに使用され、開発とテストのみを目的としています。これらのリソース・アダプターを、アプリケーションの開発とテスト以外の目的では使用しないでください。

デプロイしたアプリケーションを実行するには、ライセンスが交付されたランタイム・アダプターが必要になります。ただし、サービスを構築する際に、このアダプターをサービスに組み込むことができます。アダプターのライセンス交付により、組み込まれたアダプターを、ライセンスが交付されたランタイム・アダプターとして使用することができます。これらのアダプターは、Java Connector Architecture (J2C 1.5) に準拠しています。オープン・スタンダードである J2C は、EIS 接続のための J2EE 標準です。J2C は、管理されたフレームワークを提供します。つまり、サービスの品質 (QoS) がアプリケーション・サーバーによって提供され、それによってライフ・サイクル管理とセキュリティがトランザクションに対して提供されます。また、これらのアダプターは、Enterprise Metadata Discovery 仕様にも準拠しています (IBM CICS ECI リソース・アダプターおよび IBM IMS Connector for Java を除く)。

以下の IBM WebSphere アダプターが WebSphere Integration Developer でサポートされています。

- IBM CICS ECI リソース・アダプター、バージョン 7.1.0.2
- IBM IMS TM リソース・アダプター、バージョン 9.1.0.2
- IBM WebSphere Adapter for Email バージョン 6.2.0
- IBM WebSphere Adapter for FTP バージョン 6.2.0
- IBM WebSphere Adapter for Flat Files バージョン 6.2.0
- IBM WebSphere Adapter for IBM バージョン 6.2.0
- IBM WebSphere Adapter for JDBC™ バージョン 6.2.0
- IBM WebSphere Adapter for JD Edwards EnterpriseOne® バージョン 6.2.0
- IBM WebSphere Adapter for Oracle® E-Business Suite バージョン 6.2.0
- IBM WebSphere Adapter for PeopleSoft Enterprise バージョン 6.2.0
- IBM WebSphere Adapter for SAP® Software バージョン 6.2.0
- IBM WebSphere Adapter for Siebel® Business Applications バージョン 6.2.0

従来のアダプターのセットである WebSphere Business Integration Adapters もウィザードでサポートされています。

## J2EE リソース

EIS モジュール (EIS モジュール・パターンに準拠する SCA モジュール) は、J2EE プラットフォームにデプロイできます。

EIS モジュールを J2EE プラットフォームにデプロイすると、アプリケーションが EAR ファイルとしてパッケージされ、サーバーにデプロイされるので、アプリケーションを実行する準備が整います。すべての J2EE 成果物およびリソースが作成され、アプリケーションが構成され、実行の準備が整います。

## JCA 対話仕様および接続仕様の動的プロパティー

EIS バインディングは、ペイロードに付随する明確に定義された子データ・オブジェクトを使用することによって指定された InteractionSpec および ConnectionSpec に対する入力を受け入れることができます。これにより、InteractionSpec を介したリソース・アダプターとの動的な要求応答対話と、ConnectionSpec を介したコンポーネント認証が可能になります。

javax.cci.InteractionSpec は、リソース・アダプターとの対話要求の処理方法に関する情報を保持します。また、要求後に対話を行う方法に関する情報も保持します。対話によるこれらの両方向通信は、会話とも呼ばれます。

EIS バインディングはペイロードを必要とします。このペイロードは、properties という子データ・オブジェクトを格納するためのリソース・アダプターに対する引数になります。このプロパティー・データ・オブジェクトは、名前と値のペアを含みます。この名前は、特定のフォーマットでの対話仕様プロパティーの名前になります。フォーマット設定の規則は以下のとおりです。

- 名前はプレフィックス「IS」で開始し、その後にプロパティー名が続かなければなりません。例えば、InteractionId という Java Bean プロパティーを持つ対話仕様は、プロパティー名を ISInteractionId と指定します。
- 名前と値のペアは、対話仕様プロパティーの名前と単純型の値を表します。

この例ではインターフェースによって、操作の入力は「Account」データ・オブジェクトと指定されます。このインターフェースは、値が「xyz」である「workingSet」という動的 InteractionSpec プロパティを送受信するために、EIS インポート・バインディング・アプリケーションを呼び出します。

サーバーのビジネス・グラフまたはビジネス・オブジェクトには下記の「properties」ビジネス・オブジェクトが含まれていて、このビジネス・オブジェクトによってペイロードを持つプロトコル固有データを送信できます。この「properties」ビジネス・オブジェクトは組み込みのものであり、ビジネス・オブジェクトを構成するときに XML スキーマで指定する必要はありません。単に作成するだけで使用できます。XML スキーマに基づいて独自のデータ型を定義している場合は、必要な名前と値のペアを含む「properties」エレメントを指定する必要があります。

```
B0Factory dataFactory = (B0Factory) ¥
serviceManager.locateService("com/ibm/websphere/bo/B0Factory");
//Wrapper for doc-lit wrapped style interfaces,
//skip to payload for non doc-lit
DataObject docLitWrapper = dataFactory.createByElement /
("http://mytest/eis/Account", "AccountWrapper");
```

ペイロードを作成します。

```
DataObject account = docLitWrapper.createDataObject(0);
DataObject accountInfo = account.createDataObject("AccountInfo");
//Perform your setting up of payload
```

```
//Construct properties data for dynamic interaction
```

```
DataObject properties = account.createDataObject("properties");
```

名前「workingSet」に対して予想される値「xyz」を設定します。

```
properties.setString("ISworkingSet", "xyz");
```

```
//Invoke the service with argument
```

```
Service accountImport = (Service) ¥
serviceManager.locateService("AccountOutbound");
DataObject result = accountImport.invoke("createAccount", docLitWrapper);
```

```
//Get returned property
DataObject retProperties = result.getDataObject("properties");
```

```
String workingset = retProperties.getString("ISworkingSet");
```

ConnectionSpec プロパティは動的コンポーネント認証に使用できます。上記と同じ規則が適用されます。ただし、プロパティ名のプレフィックスは「CS」にする必要があります（「IS」ではありません）。ConnectionSpec プロパティは両方向ではありません。同じ「properties」データ・オブジェクトに IS プロパティと CS プロパティの両方を入れることができます。

ConnectionSpec プロパティを使用するには、インポート・バインディングで指定する resAuth を「Application」に設定します。また、リソース・アダプターがコンポーネント許可をサポートする必要があります。詳しくは、「J2EE Connector Architecture Specification」の第 8 章を参照してください。

## EIS バインディングの管理

EIS インポート・バインディングは、SCA アプリケーションの一部としてサーバーにインストールされます。管理コンソールからインポートを管理します。

### 始める前に

この作業を実行するには、マスター構成に対して変更を行う権限が必要です。

### このタスクについて

EIS インポート・モジュールを含んだインストール済みのアプリケーションが必要です。

モジュールの一部としてアダプターをデプロイした後で構成プロパティーを変更するには、ランタイム環境の管理コンソールを使用します。リソース・アダプター・プロパティー (アダプター操作全般で使用)、管理接続ファクトリー・プロパティー (アウトバウンド処理で使用)、アクティベーション・スペック・プロパティー (インバウンド処理で使用) を更新することができます。

**注:** また、スタンドアロン・アダプターのインストール後に構成プロパティーを設定することもできます。これを行うには、管理コンソールで「リソース」 → 「リソース・アダプター」を展開し、構成したいプロパティーを持つアダプターを選択します。

### 手順

1. 管理の対象とするインポートを表示します。「アプリケーション」を展開し、「SCA モジュール」をクリックし、管理する SCA モジュールの名前をクリックします。
2. 「モジュール・コンポーネント」という見出しの下にあるモジュール・コンポーネント「インポート」または「エクスポート」を展開してから、管理するインポートまたはエクスポートを展開します。インターフェースまたはバインディングの管理を選択できます。
3. オプション: インポート・インターフェースの WSDL を表示します。「インターフェース」を展開して、表示するインターフェースを選択します。インターフェースの WSDL が表示されます。WSDL は管理コンソールでは編集できませんが、テキスト・エディターを使用して変更できます。
4. オプション: バインディングを管理します。「バインディング」を展開して、管理するインポート・バインディングまたはエクスポート・バインディングをクリックします。インポートまたはエクスポートされたサービスのポートまたは名前を変更できます。
5. 変更を有効にするために、変更内容を構成に保管します。

## EIS バインディングを持つ外部クライアント

サーバーは、EIS バインディングを使用して、外部クライアントとの間でメッセージを送受信できます。

外部クライアント (Web ポータルや EIS など) は、サーバーの SCA モジュールにメッセージを送信するか、サーバー内からコンポーネントによって呼び出される必要があります。

クライアントは、動的起動インターフェース (DII) または Java インターフェースを使用して、その他のアプリケーションの場合と同様に EIS インポートを呼び出します。

1. 外部クライアントが `ServiceManager` のインスタンスを作成し、その参照名を使用して EIS インポートを検索します。検索の結果は、サービス・インターフェースの実装です。
2. クライアントが入力引数 (汎用データ・オブジェクト) を作成します。この引数は、データ・オブジェクト・スキーマを使用して動的に作成されます。このステップは、サービス・データ・オブジェクトの `DataFactory` インターフェースの実装を使用して行われます。
3. 外部クライアントが EIS を呼び出して必要な結果を取得します。

代わりに、クライアントは Java インターフェースを使用して EIS インポートを呼び出すこともできます。

1. クライアントが `ServiceManager` のインスタンスを作成し、その参照名を使用して EIS インポートを検索します。検索の結果は、EIS インポートの Java インターフェースです。
2. クライアントは入力引数および型付きデータ・オブジェクトを作成します。
3. クライアントが EIS を呼び出して必要な結果を取得します。

EIS エクスポート・インターフェースは、外部の EIS アプリケーションで使用可能なエクスポート済み SCA コンポーネントのインターフェースを定義します。このインターフェースは、(SAP や PeopleSoft などの) 外部アプリケーションが EIS エクスポート・アプリケーション・ランタイムの実装を介して呼び出すインターフェースと考えることができます。

エクスポートは `EISExportBinding` を使用してエクスポート済みサービスを外部の EIS アプリケーションにバインドします。これにより、EIS サービス要求を `listen` するよう、SCA モジュールに含まれるアプリケーションをサブスクライブできます。EIS エクスポート・バインディングは、リソース・アダプターによって (J2EE コネクター・アーキテクチャー・インターフェースを使用して) 認識されるインバウンド・イベントの定義と SCA 操作の呼び出しの間のマッピングを指定します。

`EISExportBinding` では、外部の EIS サービスが J2EE コネクター・アーキテクチャー 1.5 のインバウンド規約に基づいている必要があります。`EISExportBinding` では、データ・ハンドラーまたはデータ・バインディングをバインディング・レベルまたはメソッド・レベルで指定する必要があります。

---

## JMS バインディング

Java Message Service (JMS) プロバイダーにより、Java Messaging Service API およびプログラミング・モデルに基づいたメッセージングが可能になります。JMS プロバイダーは、JMS 宛先への接続を作成し、メッセージを送受信するための J2EE 接続ファクトリーを提供します。

以下の 3 つの JMS バインディングが提供されています。

- JMS JCA 1.5 に準拠したサービス統合バス (SIB) プロバイダー・バインディング (JMS バインディング)
- JMS 1.1 に準拠した非 JCA の汎用 JMS バインディング (汎用 JMS バインディング)
- WebSphere MQ JMS バインディング。WebSphere MQ に対し JMS プロバイダー・サポートを提供し、J2EE アプリケーションとの相互運用性を実現します。(WebSphere MQ JMS バインディング)

JMS エクスポートおよびインポート・バインディングにより、Service Component Architecture (SCA) モジュールは、外部 JMS システムに対する呼び出しを実行したり、外部 JMS システムからメッセージを受信したりできます。

また、WebSphere MQ バインディング (WebSphere MQ バインディング) もサポートされています。このバインディングにより、ネイティブ MQ ユーザーは、着信メッセージと発信メッセージの任意のフォーマットを処理できます (WebSphere MQ が必要です)。

JMS インポートおよびエクスポート・バインディングは、WebSphere Application Server の一部である JCA 1.5 ベースの SIB JMS プロバイダーを使用して JMS アプリケーションとの統合を行います。その他の JCA 1.5 ベースの JMS リソース・アダプターはサポートされません。

#### 関連タスク



WebSphere Process Server におけるイベント順序付けの有効化

## JMS バインディング: 全体像

JMS バインディングは、Service Component Architecture (SCA) 環境と JMS システムの間の接続を提供します。

### JMS バインディング

JMS インポート・バインディングおよび JMS エクスポート・バインディングの主なコンポーネントは、以下のとおりです。

- リソース・アダプター: SCA モジュールと外部 JMS システムの間の管理された両方向接続を使用可能にします。
- 接続: クライアントとプロバイダー・アプリケーションの間の仮想接続をカプセル化します。
- 宛先: 生成するメッセージの宛先またはコンシュームするメッセージの送信元を指定するためにクライアントが使用します。
- 認証データ: バインディングへのアクセスを保護するために使用します。

### JMS インポート・バインディング

JMS インポート・バインディングでは、SCA モジュール内で使用する外部 JMS アプリケーションをインポートできます。JMS インポート・バインディングにより、SCA モジュール内のコンポーネントは、外部 JMS アプリケーションが提供するサービスと通信できるようになります。

JMS 宛先に関連する JMS プロバイダーへの接続を作成するには、JMS 接続ファクトリーを使用します。デフォルト・メッセージング・プロバイダーの JMS 接続ファクトリーを管理するには、接続ファクトリーの管理オブジェクトを使用します。

外部 JMS システムとの対話では、要求を送信し、応答を受信するための宛先が使用されます。

JMS インポート・バインディングでは、呼び出されている操作のタイプに応じた以下の 2 種類の使用シナリオがサポートされています。

- 片方向: JMS インポートは、インポート・バインディングに構成された送信宛先にメッセージを送信します。JMS ヘッダーの `replyTo` フィールドには何も設定しません。
- 両方向 (要求/応答): JMS インポートは、送信宛先にメッセージを送信し、その後 SCA コンポーネントから受信した応答を維持します。

`receive` 宛先がアウトバウンド・メッセージの `replyTo` ヘッダー・プロパティーに設定されます。受信宛先で `listen` するためのメッセージ駆動型 Bean (MDB) をデプロイします。応答を受信すると、MDB は応答をコンポーネントに返します。

インポート・バインディングは、要求メッセージ ID (デフォルト) または要求メッセージ関連 ID から応答メッセージ関連 ID がコピーされていることを期待するように (WebSphere Integration Developer の「**応答関連スキーム**」フィールドを使用して) 構成することができます。

片方向と両方向のいずれのシナリオを使用する場合も、動的および静的ヘッダー・プロパティーを指定できます。静的プロパティーは、JMS インポート・メソッド・バインディングから設定できます。これらのプロパティーの一部は、SCA JMS ランタイムで特別な意味を持ちます。

重要な点として、JMS は非同期バインディングであることに注意してください。呼び出し側コンポーネントが JMS インポートを同期的に呼び出すと (両方向操作の場合)、呼び出し側コンポーネントは、JMS サービスからの応答が返されるまでブロックされます。

255 ページの図 16 は、インポートがどのように外部サービスにリンクされているのかを示しています。

## JMS インポート

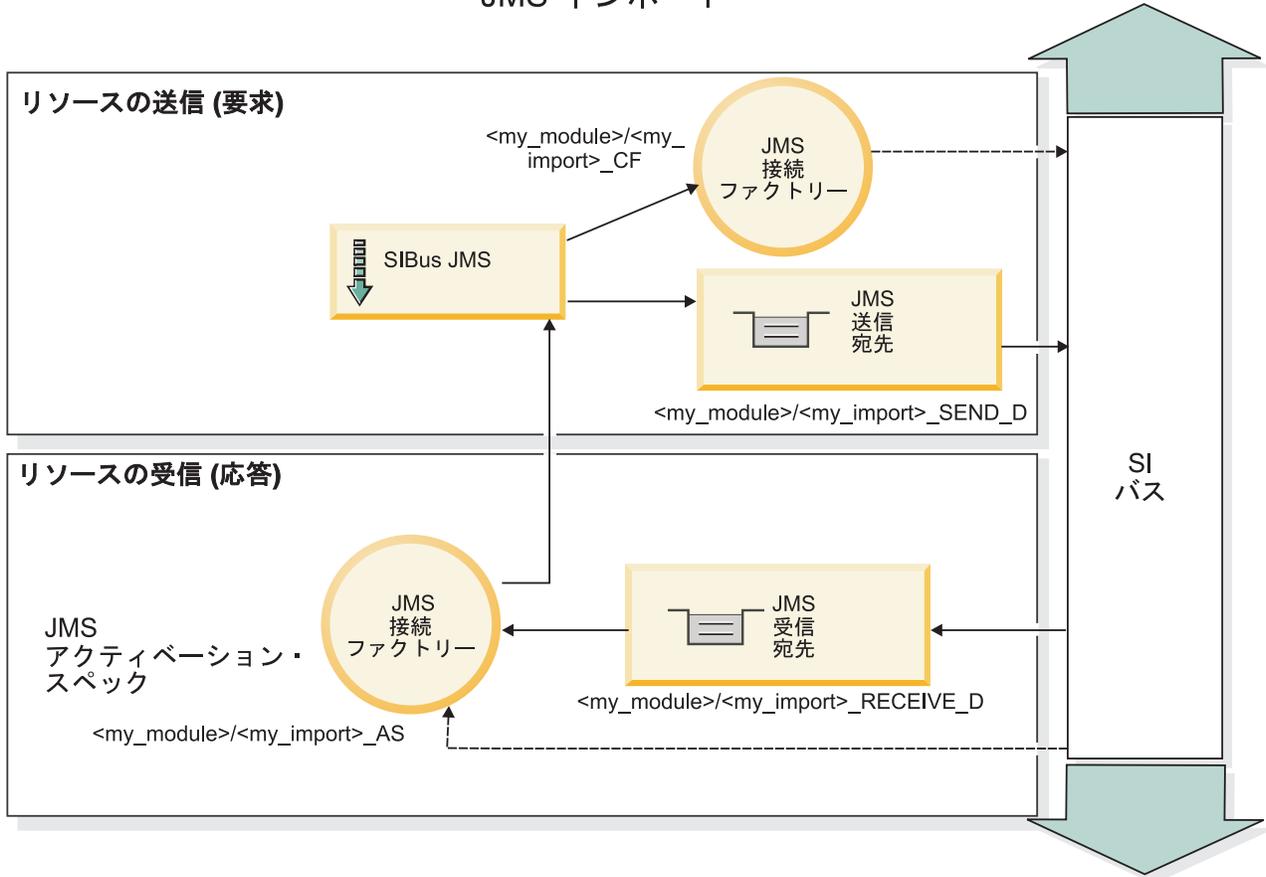


図 16. JMS インポート・バインディングのリソース

### JMS エクスポート・バインディング

JMS エクスポート・バインディングは、SCA モジュールが外部 JMS アプリケーションにサービスを提供する手段を提供します。

JMS エクスポートの接続部分は、構成可能なアクティベーション・スペックです。

JMS エクスポートには、send 宛先と receive 宛先があります。

- receive 宛先は、送信先コンポーネントに対する着信メッセージを格納する宛先です。
- send 宛先は、応答を送信する宛先です。ただし、着信メッセージで replyTo ヘッダー・プロパティを使用してこの宛先をオーバーライドしている場合は、その宛先が優先されます。

エクスポート・バインディングで指定された receive 宛先に着信する要求を listen するため、MDB がデプロイされます。send フィールドで指定された宛先は、呼び出されたコンポーネントが応答を返す場合にインバウンド要求に対する応答を送信するために使用されます。着信メッセージの replyTo フィールドで指定された宛先は、send で指定された宛先をオーバーライドします。

図 17 は、外部の要求側がどのようにエクスポートにリンクされているのかを示しています。

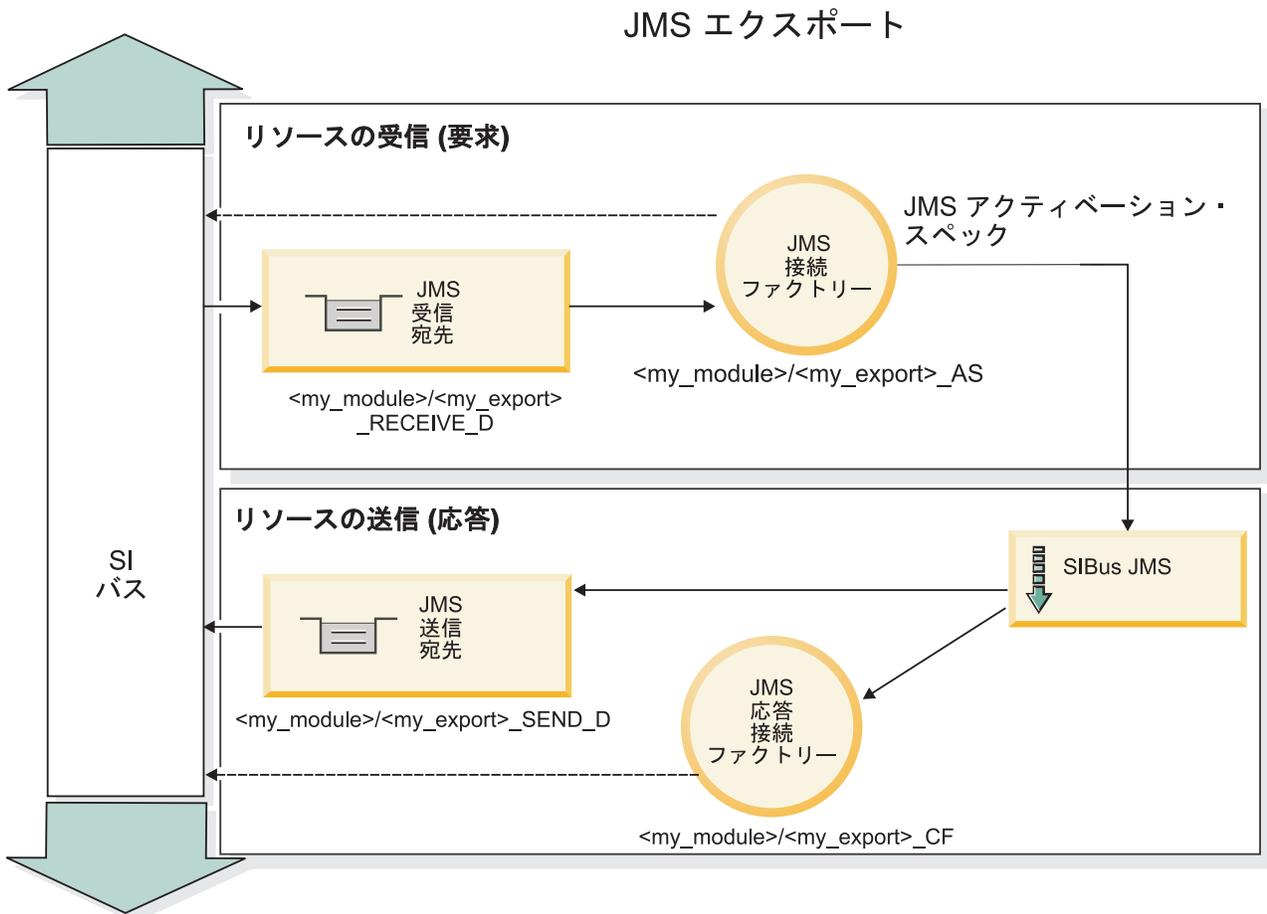


図 17. JMS エクスポート・バインディングのリソース

## JMS 統合とリソース・アダプター

Java Message Service (JMS) は、使用可能な JMS JCA 1.5 ベースのリソース・アダプターを介して統合を提供します。JMS 統合のための完全なサポートが、サービス統合バス (SIB) JMS リソース・アダプターに対して用意されています。

JCA 1.5 リソース・アダプター用の JMS プロバイダーは、外部 JCA 1.5 準拠 JMS システムと統合する場合に使用します。JCA 1.5 に準拠した外部サービスは、SIB JMS リソース・アダプターを使用してメッセージを送受信でき、Service Component Architecture (SCA) コンポーネントと統合します。

他のプロバイダー固有の JCA 1.5 リソース・アダプターの使用はサポートされません。

JMS モジュールは J2SE 環境にデプロイできません。それらのモジュールは J2EE 環境にのみデプロイできます。

## JMS バインディングの主な特徴

JMS のインポート・バインディングおよびエクスポート・バインディングの主な特徴は、ヘッダーと作成される J2EE リソースです。

### 特殊ヘッダー

特殊ヘッダーのプロパティは、JMS インポートおよびエクスポートで使用され、ターゲットに対してメッセージの処理方法を指示します。

例えば、TargetFunctionName がネイティブ・メソッドから操作メソッドにマップされます。

### J2EE リソース

JMS インポートおよびエクスポートを J2EE 環境にデプロイすると、さまざまな J2EE リソースが作成されます。

#### ConnectionFactory

クライアントが JMS プロバイダーとの接続を作成するために使用します。

#### ActivationSpec

インポートでは、要求に対する応答を受信するときに使用されます。エクスポートでは、メッセージング・システムとの対話で MDB を表すメッセージ・エンドポイントを構成するときに使用されます。

#### 宛先

- 送信宛先: インポートの場合は、要求または出力メッセージが送信される宛先になります。エクスポートの場合は、応答メッセージが送信される宛先になります。ただし、着信メッセージの JMSReplyTo ヘッダー・フィールドにより置き換えられた場合は、その宛先が優先されます。
- 受信宛先: 着信メッセージが格納される宛先です。インポートの場合は応答になり、エクスポートの場合は要求になります。
- コールバック宛先: 関連情報の保管に使用される SCA JMS システム宛先です。この宛先に対して、読み取りまたは書き込みを行わないでください。

インストール・タスクでは、ConnectionFactory および 3 つの宛先を作成します。また、ActivationSpec を作成して、ランタイム MDB が受信宛先で応答を listen できるようにします。これらのリソースのプロパティは、インポート・ファイルまたはエクスポート・ファイルに指定されます。

## JMS バインディングの管理

JMS インポートおよびエクスポート・バインディングは、サーバー内から管理できます。

管理コンソールを使用して、JMS インポートおよびエクスポート・バインディングを構成および管理します。

JMS インポートおよびエクスポート・バインディングの詳しい作成手順は、WebSphere Integration Developer インフォメーション・センターの「**WebSphere Integration Developer**」 > 「**Developing integration applications**」 > 「**Accessing**

external services with messaging systems」 > 「Java Message Service (JMS)」 > 「Working with JMS bindings」を参照してください。

## JMS バインディングのプロパティ

JMS のインポート・バインディングおよびエクスポート・バインディングは、デプロイメント中すべての必要な接続ファクトリーが作成された状態でインストールできます。または、既存のリソースのセットを指すように構成することもできます。

一般に、JMS インポート・バインディングおよび JMS エクスポート・バインディングは WebSphere Integration Developer で作成されます。バインディングを構成するときには、JMS バインディングに必要な接続と宛先を作成するか（「**新規メッセージング・プロバイダー・リソースの構成 (Configure new messaging provider resources)**」(デフォルト) を選択)、あるいは「**事前構成メッセージング・プロバイダー・リソースの使用 (Use pre-configured messaging provider resources)**」を選択します。事前構成を選択する場合は、接続ファクトリーの JNDI 名と送信宛先を追加するか (片方向操作の場合)、送信宛先と受信宛先を追加します (要求/応答操作の場合)。

このどちらの方法を選択するかによって、JMS バインディングの構成が異なります。

以下のリストに、「事前構成メッセージング・プロバイダー・リソースの使用 (Use pre-configured messaging provider resources)」を選択した場合に指定するリソースに例を記載します。これらの例でのモジュール名は Inventory です。

表 19. インポート・バインディングの値の例

プロパティ	例
接続ファクトリーの場合の JNDI 名	Inventory/Inventory_CF
送信宛先の場合の JNDI 名	Inventory/Inventory_SEND_D
受信宛先の場合の JNDI 名	Inventory/Inventory_RECEIVE_D

表 20. エクスポート・バインディングの値の例

プロパティ	例
アクティベーション・スベックの場合の JNDI 名	Inventory/Inventory_AS
受信宛先の場合の JNDI 名	Inventory/Inventory_RECEIVE_D
送信宛先の場合の JNDI 名	Inventory/Inventory_SEND_D

注: リソースは、サーバー・スコープで作成されます。管理コンソールのスコープは、初期設定では「すべてのスコープ」に設定されています。新規リソースを作成するには、このスコープを「セル」または「ノード」に設定する必要があります。デフォルトのリストから既存のリソースを選択できます。

## JMS バインディングの構成

JMS インポート/エクスポート・バインディングを構成し、リソースの特殊機能を適用できます。管理用タスクは、WebSphere 管理コンソールを使用して実行します。

## 始める前に

管理コンソールでプロファイルに対して変更を行い、変更内容を保管する権限を持っている必要があります。

## このタスクについて

JMS バインディングは、サーバー・プロファイル内の Service Component Architecture (SCA) アプリケーションの一部としてインストールする必要があります。

### 手順

1. SCA モジュールを選択します。管理コンソールから「アプリケーション」 → 「SCA モジュール」をクリックした後、*modulename* をクリックします。構成ページが開きます。
2. 「モジュール・コンポーネント」の下で、「インポート」または「エクスポート」を展開します。インストールされているインポートまたはエクスポートのリストが表示されます。
3. インポートまたはエクスポートを展開した後、「バインディング」を展開します。管理可能なバインディングのリストが表示されます。
4. 管理するバインディングをクリックします。バインディングの一般プロパティが表示されます。
  - 「送信リソース」カテゴリには、「接続ファクトリー」と「送信宛先」が含まれています。
  - 「受信リソース」カテゴリには、「応答接続ファクトリー」と「アクティベーション・スペック」が含まれています。
  - 「拡張リソース」カテゴリには、コールバック・リソースとその他の使用可能なリソースが含まれています。

**注:** リソースにアクセスするもう 1 つの方法として、テキスト・ボックスに JNDI 名を入力する方法があります。ただしこの方法では、まだ構成されていないリソースの名前の入力も可能になります。

5. 必要なリソースを管理します。
  - a. 「参照」をクリックすると、JNDI 名のリストを含むウィンドウが開きます。このウィンドウで、必要な JNDI 名を選択します。
  - b. 「構成」をクリックすると、この JNDI 名によって参照される、対応するパネルが開きます。

「構成」を選択すると、対応するサーバー・パネルが開きます。

6. 設定を管理する準備ができている場合は、「適用」または「OK」をクリックします。

## 次のタスク

JNDI バインディングを反映するため、アプリケーションが自動的に再始動されません。

## JMS ヘッダー

JMS メッセージには、2 つのタイプのヘッダーが含まれます。1 つは JMS システム・ヘッダー、もう 1 つは複数の JMS プロパティです。メディエーション・モジュールでは、いずれのタイプのヘッダーにも、サービス・メッセージ・オブジェクト (SMO) 内、または ContextService API を使用してアクセスできます。

### JMS システム・ヘッダー

SMO では、JMS システム・ヘッダーは JMSHeader エlementによって表されます。この Element には、JMS ヘッダーに通常あるすべてのフィールドが含まれます。これらのフィールドはメディエーション (または ContextService) で変更できますが、SMO に設定された一部の JMS システム・ヘッダー・フィールドは、システムまたは静的な値によってオーバーライドされるため、アウトバウンド JMS メッセージでは伝搬されません。

メディエーション (または ContextService) で更新可能な JMS システム・ヘッダーのキー・フィールドには以下があります。

- **JMSType** および **JMSCorrelationID** – 特定の事前定義メッセージ・ヘッダー・プロパティの値
- **JMSDeliveryMode** – 送達モードの値 (persistent または nonpersistent。デフォルトは persistent)
- **JMSPriority** – 優先度の値 (0 から 9。デフォルトは JMS\_Default\_Priority)

### JMS プロパティ

JMS プロパティは、SMO ではプロパティ・リスト内のエントリとして表されます。プロパティは、メディエーション内、または ContextService API を使用して追加、更新、または削除できます。

プロパティは、JMS バインディングに静的に設定することもできます。静的に設定されたプロパティは、動的に設定される (同じ名前の) 設定をオーバーライドします。

他のバインディング (例えば HTTP バインディング) から伝搬されるユーザー・プロパティは、JMS バインディングでは JMS プロパティとして出力されます。

### ヘッダー伝搬の設定

JMS システム・ヘッダーおよびプロパティのインバウンド JMS メッセージからダウンストリームのコンポーネントへの伝搬、またはアップストリームのコンポーネントからアウトバウンド JMS メッセージへの伝搬は、バインディングのプロトコル・ヘッダー伝搬フラグで制御できます。

プロトコル・ヘッダー伝搬を設定すると、以下のリストに説明するようにヘッダー情報をメッセージまたはターゲット・コンポーネントに流すことができます。

- JMS エクスポート要求

メッセージ内で受信した JMS ヘッダーは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。メッセージ内で受信した JMS プロパティは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。

- JMS エクスポート応答

コンテキスト・サービスに設定されたすべての JMS ヘッダー・フィールドは、JMS エクスポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。コンテキスト・サービスに設定されたすべてプロパティは、JMS エクスポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。

- JMS インポート要求

コンテキスト・サービスに設定されたすべての JMS ヘッダー・フィールドは、JMS インポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。コンテキスト・サービスに設定されたすべてプロパティは、JMS インポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。

- JMS インポート応答

メッセージ内で受信した JMS ヘッダーは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。メッセージ内で受信した JMS プロパティは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。

#### 関連概念

274 ページの『汎用 JMS ヘッダー』

汎用 JMS ヘッダーは、すべての汎用 JMS メッセージ・プロパティを含むサービス・データ・オブジェクト (SDO) です。これらのプロパティはインバウンド・メッセージからのものか、アウトバウンド・メッセージに適用されるプロパティにすることができます。

298 ページの『WebSphere MQ ヘッダー』

WebSphere MQ ヘッダーには、Service Component Architecture (SCA) メッセージへの変換に関する特定の規則が組み込まれています。

## 外部クライアント

サーバーは、JMS バインディングを使用して、外部クライアントとの間でメッセージを送受信できます。

外部クライアント (Web ポータルやエンタープライズ情報システムなど) は、サーバーの SCA モジュールにメッセージを送信できます。また、サーバー内からコンポーネントによって外部クライアントを呼び出すこともできます。

JMS エクスポート・コンポーネントは、エクスポート・バインディングで指定された受信宛先に着信する要求を listen するためのメッセージ駆動型 Bean (MDB) をデプロイします。送信フィールドで指定された宛先は、呼び出されたアプリケーションが応答を返す場合にインバウンド要求に対する応答を送信するために使用されま

す。したがって、外部クライアントは、エクスポート・バインディングを使用してアプリケーションを呼び出すことができます。

JMS インポートは外部クライアントにバインドし、外部クライアントにメッセージを配信できます。このメッセージは、外部クライアントからの応答を要求してもしなくても構いません。

## 外部クライアントの使用

外部クライアント (サーバー外部のクライアント) とサーバーにインストールされたアプリケーションの対話が必要になる場合があります。

### このタスクについて

外部クライアントとサーバー上のアプリケーションが対話する場合の、非常に簡単なシナリオを以下に示します。以下の図は、標準的な単純なシナリオを表しています。

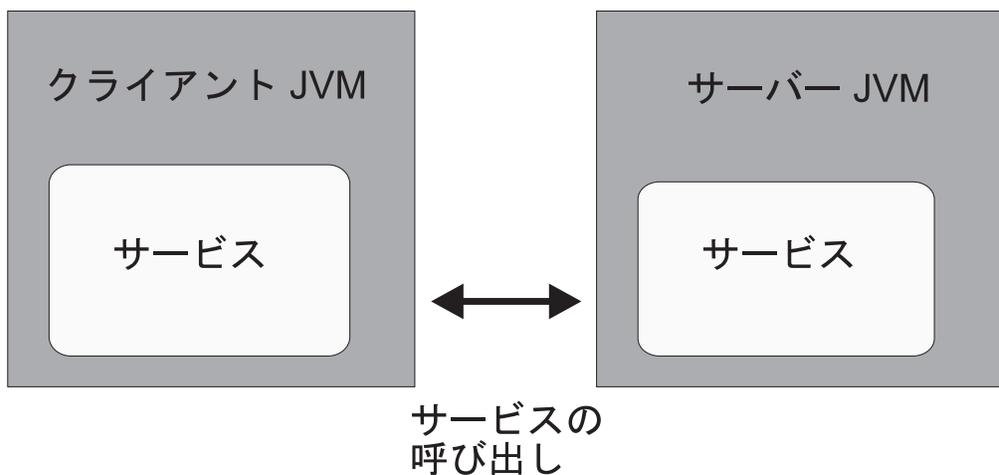


図 18. 単純なユース・ケース・シナリオ: 外部クライアントとサーバー・アプリケーションの対話

SCA アプリケーションには、JMS バインディングによるエクスポートが含まれています。これにより、外部クライアントからアプリケーションを使用できるようになります。

外部クライアントがサーバーとは別個の Java 仮想マシン (JVM) 内にある場合、JMS エクスポートとの接続および対話を作成するためにいくつかのステップを実行する必要があります。クライアントは、正しい値が指定された `InitialContext` を取得し、JNDI を使用してリソースを検索します。次にクライアントは、JMS 1.1 仕様のクライアントを使用して宛先にアクセスし、宛先の送信メッセージと受信メッセージにアクセスします。

ランタイムにより自動作成されるリソースのデフォルトの JNDI 名は、このセクションの構成に関するトピックにリストされています。ただし、事前に作成済みのリソースがある場合は、その JNDI 名を使用します。

## 手順

1. メッセージを送信するため、JMS の宛先と接続ファクトリーを構成します。
2. JNDI コンテキスト、SIB リソース・アダプターのポート、およびメッセージング・ブートストラッピング・ポートが正しいことを確認します。

サーバーはいくつかのデフォルト・ポートを使用しますが、それより多くのサーバーがそのマシンにインストールされている場合は、他のサーバー・インスタンスとの競合を避けるために、インストール時に代替のポートが作成されます。管理コンソールを使用して、サーバーが使用しているポートを調べることができます。「サーバー」→「アプリケーション・サーバー」→ *your\_server\_name* →「構成」を選択して、「通信 (Communication)」の下で「ポート」をクリックします。これで、使用するポートを編集できます。

3. クライアントは、正しい値が指定された初期コンテキストを取得して、JNDI を使用してリソースを検索します。
4. クライアントは、JMS 1.1 仕様に基づいて宛先にアクセスし、宛先の送信メッセージと受信メッセージにアクセスします。

## JMS バインディングのトラブルシューティング

JMS バインディングで発生した問題を診断し、修正できます。

### 実装例外

JMS のインポートおよびエクスポート実装は、さまざまなエラー状態に応じて、以下の 2 種類の例外のうちのいずれかを戻すことがあります。

- サービス・ビジネス例外: サービス・ビジネス・インターフェース (WSDL ポート・タイプ) で指定された障害が発生した場合に、この例外が戻されます。
- サービス・ランタイム例外: その他のすべてのケースで生成されます。ほとんどの場合、cause 例外には元の例外 (JMSEException) が含まれます。

例えばインポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。そのため、複数の応答を受信した場合や遅延応答 (SCA の応答有効期限が切れた応答) を受信した場合は、サービス・ランタイム例外がスローされます。この場合、トランザクションはロールバックされ、応答メッセージはキューからバックアウトされるか、または Failed Event Manager によって処理されます。

### 主な障害状態

JMS バインディングの主な障害状態は、トランザクションの意味構造、JMS プロバイダー構成、またはその他のコンポーネントの既存動作への参照に基づいて判別されます。主な障害状態は以下のとおりです。

- JMS プロバイダーまたは宛先に接続できない。

JMS プロバイダーに接続してメッセージを受信できない場合は、MDB リスナー・ポートを開始することができません。この状態は、WebSphere Application Server ログに記録されます。正常に取得されるまで (または期限切れとなるまで)、永続メッセージは宛先に残ります。

JMS プロバイダーに接続してアウトバウンド・メッセージを送信できない場合、送信操作を制御するトランザクションがロールバックされます。

- インバウンド・メッセージを解析できないか、アウトバウンド・メッセージを構成できない。

データ・バインディングまたはデータ・ハンドラーが失敗すると、作業を制御するトランザクションがロールバックされます。

- アウトバウンド・メッセージを送信できない。

メッセージを送信できないと、関連するトランザクションがロールバックされません。

- 複数のメッセージまたは予期しない遅延応答メッセージが返される。

インポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。また、応答を受信できる有効期間は、要求の SCA 応答有効期限修飾子によって決まります。応答を受信したとき、または有効期間を超えたときに、相関レコードが削除されます。予期しない応答メッセージを受信した場合や遅れて応答を受信した場合は、サービス・ランタイム例外がスローされます。

## Failed Event Manager に表示されない JMS ベースの SCA メッセージ

JMS の対話の失敗によって SCA メッセージが発生した場合は、Failed Event Manager でこのメッセージを見つけることとなります。Failed Event Manager にこのようなメッセージが表示されない場合は、JMS 宛先の基礎となる SIB 宛先の最大配信失敗回数値に 1 よりも大きい値が設定されているかどうかを確認してください。この値を 2 以上に設定すると、JMS バインディングに対して SCA を呼び出す際に、Failed Event Manager と対話することができます。

### 関連概念



Failed Event Manager について

Failed Event Manager は、呼び出しの失敗を処理および再サブミットするための Web ベースのクライアントです。

## 例外の処理

バインディングがどのように構成されているかによって、データ・ハンドラーまたはデータ・バインディングによる例外の処理方法が決まります。また、メディエーション・フローの性質により、そのような例外がスローされたときのシステムの振る舞いが決まります。

データ・ハンドラーまたはデータ・バインディングがバインディングによって呼び出されるときには、さまざまな問題が発生する可能性があります。例えば、データ・ハンドラーが、ペイロードが破損しているメッセージを受信したり、誤った形式のメッセージを読み取るなどです。

バインディングによるそのような例外の処理方法は、データ・ハンドラーおよびデータ・バインディングの実装方法によって決まります。推奨される振る舞いは、`DataBindingException` をスローするようにデータ・バインディングを設計することです。

`DataBindingException` などのランタイム例外がスローされた場合:

- メディエーション・フローがトランザクションとして構成されている場合、デフォルトでは、手動でやり直し、または削除できるように JMS メッセージは `Failed Event Manager` に保管されます。

**注:** バインディングのリカバリー・モードを変更することによって、メッセージが `Failed Event Manager` に保管される代わりに、ロールバックされるようになります。

- メディエーション・フローがトランザクションではない場合、例外はログに記録され、メッセージは失われます。

データ・ハンドラーの場合も同様の状況になります。データ・ハンドラーはデータ・バインディングによって呼び出されるため、すべてのデータ・ハンドラー例外はデータ・バインディング例外内にラップされます。したがって、`DataHandlerException` は `DataBindingException` として報告されます。

---

## 汎用 JMS バインディング

汎用 JMS バインディングにより、サード・パーティーの JMS 1.1 準拠プロバイダーに対する接続が提供されます。汎用 JMS バインディングの操作は、JMS バインディングの操作と似ています。

JMS バインディングを介して提供されるサービスにより、`Service Component Architecture (SCA)` モジュールは、外部システムに対する呼び出しを実行したり、外部システムからメッセージを受信したりできます。このシステムとして、外部 JMS システムを使用できます。

汎用 JMS バインディングは、JCA 1.5 非準拠 JMS プロバイダー (JMS プロバイダーが JMS 1.1 をサポートし、オプションの `JMS Application Server Facility` を実装している場合) との統合を実現します。汎用 JMS バインディングは、JCA 1.5 をサポートしないが JMS 1.1 仕様の `Application Server Facility` をサポートする JMS プロバイダーをサポートしています (これには、Oracle AQ、TIBCO、SonicMQ、WebMethods、BEA WebLogic、WebSphere MQ などが含まれます)。JCA 1.5 JMS プロバイダーである SIB は、サポートされていません。

SCA 環境内で JCA 1.5 非準拠の JMS ベース・システムと統合するときに、この汎用バインディングを使用します。これにより、ターゲットの外部アプリケーションがメッセージを送受信でき、SCA コンポーネントと統合できます。

### 関連タスク



WebSphere Process Server におけるイベント順序付けの有効化

## 汎用 JMS バインディング: 全体像

汎用 JMS バインディングは、Service Component Architecture (SCA) 環境と JMS システム (JMS 1.1 に準拠し、オプションの JMS Application Server Facility を実装している場合) の間の接続を提供する非 JCA JMS バインディングです。

### 汎用 JMS バインディング

汎用 JMS インポートおよびエクスポート・バインディングの主な側面は以下のとおりです。

- リソース・アダプター: エンタープライズ情報システム (EIS) と J2EE コンポーネントの間の管理対象双方向接続を使用可能にします。
- 接続: クライアントとプロバイダー・アプリケーションの間の仮想接続をカプセル化します。
- 宛先: 生成するメッセージの宛先またはコンシュームするメッセージの送信元を指定するためにクライアントが使用します。
- 認証データ: バインディングへのアクセスを保護するために使用します。

### 汎用 JMS インポート・バインディング

汎用 JMS インポート・バインディングにより、SCA モジュール内のコンポーネントは、外部の JCA 1.5 非準拠の JMS プロバイダーが提供するサービスと通信できるようになります。

JMS インポートの接続部分は、接続ファクトリーです。接続ファクトリーとは、クライアントがプロバイダーへの接続を作成するために使用するオブジェクトであり、管理者によって定義された一連の接続構成パラメーターをカプセル化します。各接続ファクトリーは、ConnectionFactory、QueueConnectionFactory、または TopicConnectionFactory インターフェースのインスタンスです。

外部 JMS システムとの対話では、要求を送信し、応答を受信するための宛先が使用されます。

汎用 JMS インポート・バインディングでは、呼び出されている操作のタイプに応じた以下の 2 種類の使用シナリオがサポートされています。

- 片方向: 汎用 JMS インポートは、インポート・バインディングに構成された送信宛先にメッセージを送信します。JMS ヘッダーの replyTo フィールドには何も送信しません。
- 両方向 (要求/応答): 汎用 JMS インポートは、送信宛先にメッセージを送信し、その後 SCA コンポーネントから受信した応答を維持します。

receive 宛先がアウトバウンド・メッセージの replyTo ヘッダー・プロパティーに設定されます。受信宛先で listen するためのメッセージ駆動型 Bean (MDB) をデプロイします。応答を受信すると、MDB は応答をコンポーネントに返します。

インポート・バインディングは、要求メッセージ ID (デフォルト) または要求メッセージ関連 ID から応答メッセージ関連 ID がコピーされていることを期待するように (WebSphere Integration Developer の「応答関連スキーム」フィールドを使用して) 構成することができます。

片方向と両方向のいずれのシナリオを使用する場合も、動的および静的ヘッダー・プロパティを指定できます。静的プロパティは汎用 JMS インポート・メソッド・バインディングから設定できます。これらのプロパティのなかには、SCA JMS ランタイムにとって特殊な意味を持つものがあります。

重要な点として、汎用 JMS は非同期バインディングであることに注意してください。呼び出し側コンポーネントが汎用 JMS インポートを同期的に呼び出すと (両方向操作の場合)、呼び出し側コンポーネントは、JMS サービスからの応答が返されるまでブロックされます。

図 19 は、インポートがどのように外部サービスにリンクされているのかを示しています。

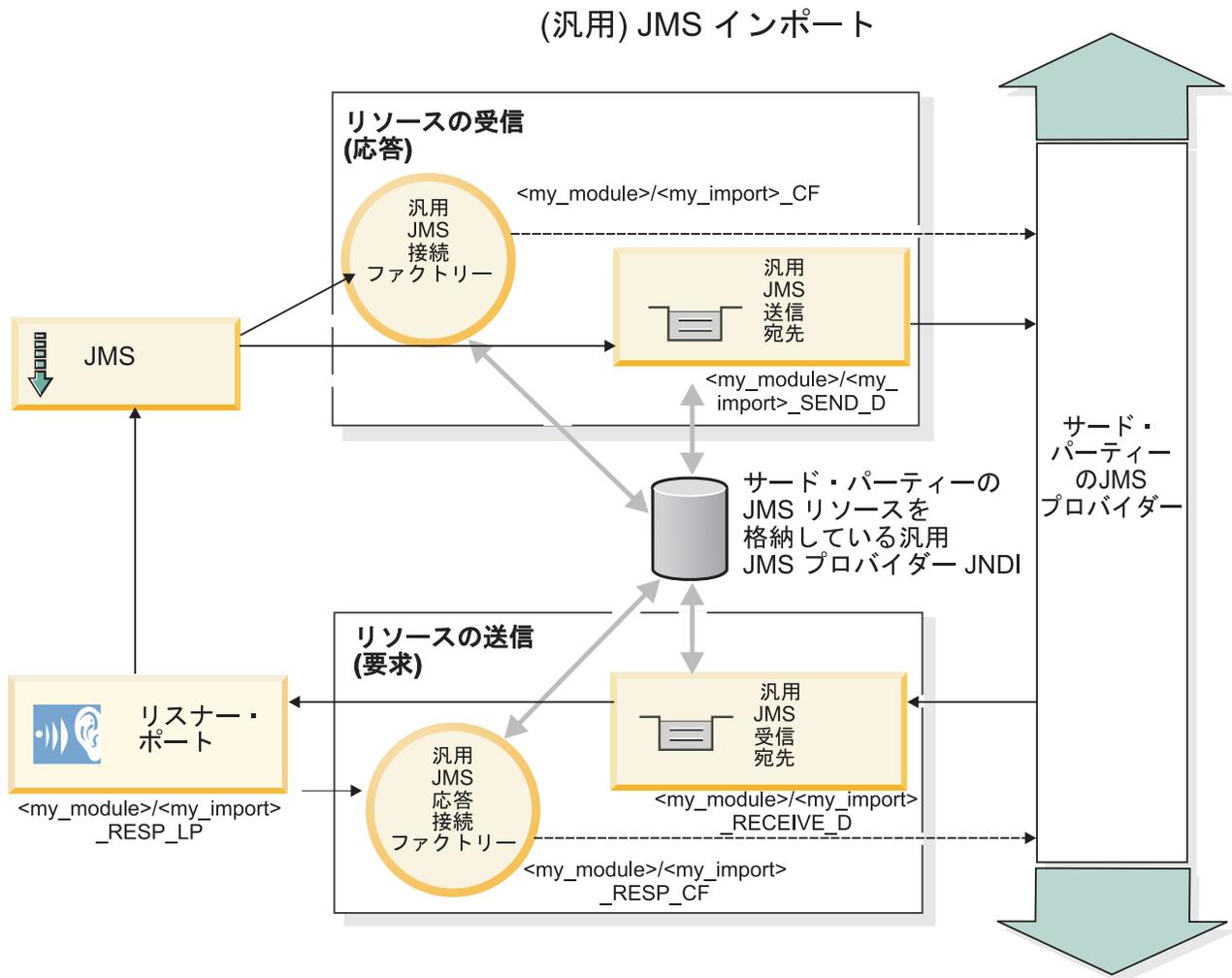


図 19. 汎用 JMS インポート・バインディング・リソース

### 汎用 JMS エクスポート・バインディング

汎用 JMS エクスポート・バインディングは、SCA モジュールが外部 JMS アプリケーションにサービスを提供する手段を提供します。

JMS エクスポートの接続部分は、ConnectionFactory および ListenerPort から構成されます。

汎用 JMS エクスポートには、send 宛先と receive 宛先があります。

- receive 宛先は、送信先コンポーネントに対する着信メッセージを格納する宛先です。
- send 宛先は、応答を送信する宛先です。ただし、着信メッセージで replyTo ヘッダー・プロパティを使用してこの宛先をオーバーライドしている場合は、その宛先が優先されます。

エクスポート・バインディングで指定された receive 宛先に着信する要求を listen するため、MDB がデプロイされます。

- send フィールドで指定された宛先は、呼び出されたコンポーネントが応答を返す場合にインバウンド要求に対する応答を送信するために使用されます。
- 着信メッセージの replyTo フィールドで指定された宛先は、send フィールドで指定された宛先をオーバーライドします。
- 要求/応答シナリオの場合、応答が要求 message ID を応答メッセージの correlation ID フィールドにコピーする(デフォルト) ことを期待するように、(WebSphere Integration Developer の「**応答関連スキーマ**」フィールドを使用して) インポート・バインディングを構成できます。または、応答が要求の correlation ID を応答メッセージの correlation ID フィールドにコピーすることもできます。

269 ページの図 20 は、外部の要求側がどのようにエクスポートにリンクされているのかを示しています。

## (汎用) JMS エクスポート

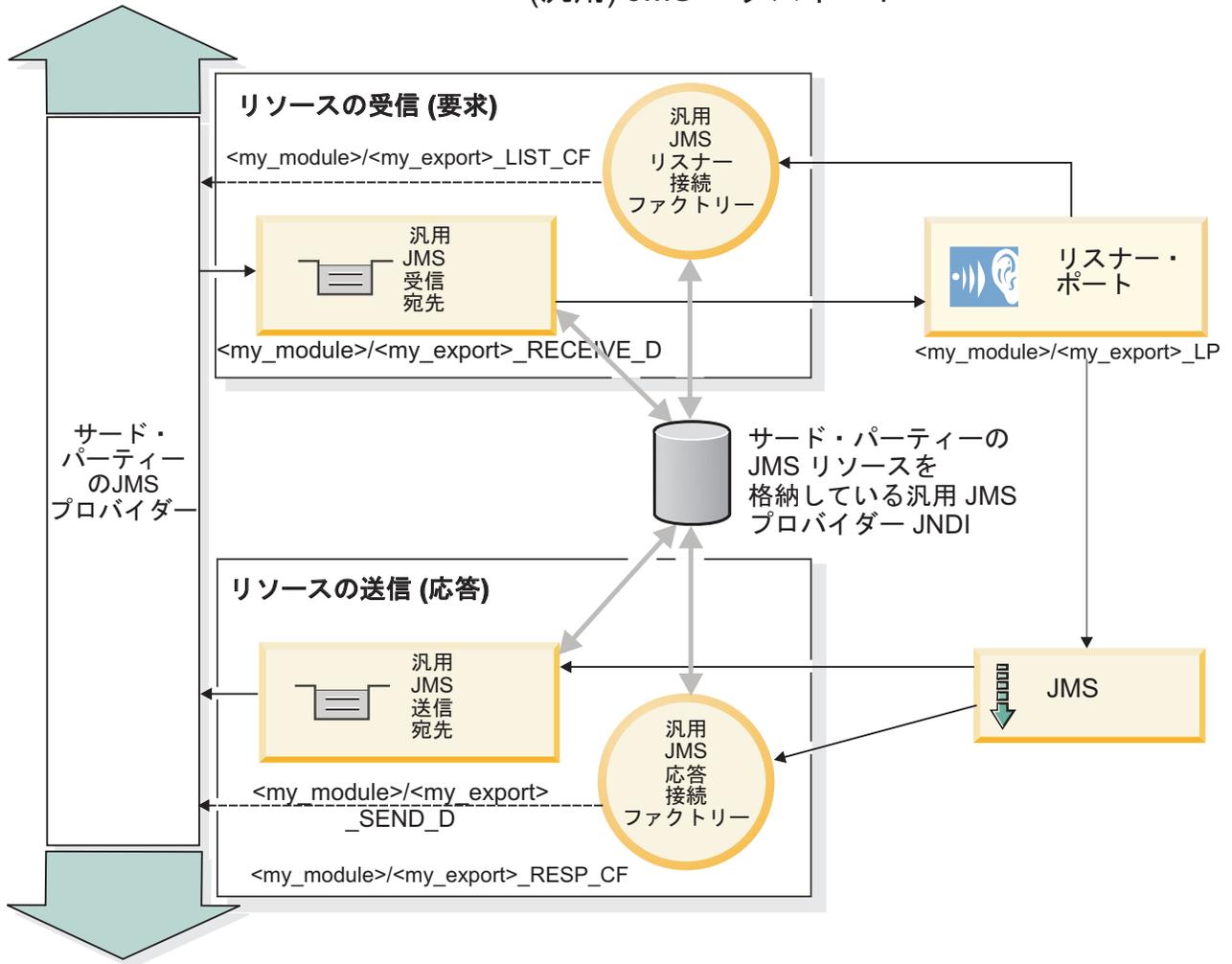


図 20. 汎用 JMS エクスポート・バインディング・リソース

## 汎用 JMS バインディングの主な機能

汎用 JMS インポート・バインディングとエクスポート・バインディングの機能は、WebSphere 埋め込み JMS と MQ JMS インポート・バインディングの機能と整合性があります。主な機能は、ヘッダー定義および既存の J2EE リソースへのアクセスです。ただし、汎用という性質上、JMS プロバイダー固有の接続オプションはありません。また、このバインディングでは、デプロイメントおよびインストール時にリソースを生成できる機能が限定されています。

### 汎用インポート

MQ JMS インポート・アプリケーションと同様に、汎用 JMS 実装は非同期であり、3 つの呼び出し (片方向、両方向 (要求/応答ともいう)、コールバック) をサポートしています。

JMS インポートがデプロイされる時、ランタイム環境によって提供されるメッセージ駆動型 Bean がデプロイされます。MDB は要求メッセージに対する応答を

listen します。MDB は、要求とともに送信される宛先に関連付けられています (この宛先を listen します)。この宛先は、JMS メッセージの replyTo ヘッダー・フィールドで指定されます。

## 汎用エクスポート

汎用 JMS エクスポート・バインディングは、結果の戻りの処理方法が EIS エクスポート・バインディングとは異なります。汎用 JMS エクスポートは、着信メッセージに指定された replyTo 宛先に応答を明示的に送信します。この宛先が指定されていない場合は、送信宛先が使用されます。

汎用 JMS エクスポートがデプロイされるときに、メッセージ駆動型 Bean (汎用 JMS インポートに使用されるものとは異なる MDB) がデプロイされます。この MDB は受信宛先で着信要求を listen し、次にその要求が SCA ランタイムで処理されるようにディスパッチします。

## 特殊ヘッダー

特殊ヘッダーのプロパティは、汎用 JMS インポートとエクスポートで使用され、ターゲット・バインディングに対してメッセージの処理方法を指示します。

例えば、TargetFunctionName プロパティは、呼び出されているエクスポートのインターフェース内の操作名を識別する際に、デフォルトの関数セクターによって使用されます。

注: インポート・バインディングを構成して、各操作名に TargetFunctionName ヘッダーを設定することができます。

## J2EE リソース

JMS バインディングを J2EE 環境にデプロイすると、いくつかの J2EE リソースが作成されます。

- インポートの場合は受信宛先 (応答、両方向のみ)、エクスポートの場合は受信宛先 (要求) で listen するためのリスナー・ポート
- outboundConnection (インポート) および inboundConnection (エクスポート) の汎用 JMS 接続ファクトリー
- 送信宛先 (インポート) および受信宛先 (エクスポート、両方向のみ) の汎用 JMS 宛先
- responseConnection の汎用 JMS 接続ファクトリー (両方向のみ、オプション。これ以外の場合は、インポートに outboundConnection が使用され、エクスポートに inboundConnection が使用される)
- 受信宛先 (インポート) および送信宛先 (エクスポート) の汎用 JMS 宛先 (両方向のみ)
- SIB コールバック・キュー宛先にアクセスするときに使用されるデフォルトのメッセージング・プロバイダー・コールバック JMS 宛先 (両方向のみ)
- コールバック JMS 宛先にアクセスするときに使用されるデフォルトのメッセージング・プロバイダー・コールバック JMS 接続ファクトリー (両方向のみ)
- 応答処理中に使用される要求メッセージに関する情報を格納するための SIB コールバック・キュー宛先 (両方向のみ)

インストール・タスクにより、インポート・ファイルおよびエクスポート・ファイルの情報から ConnectionFactory、3 つの宛先、および ActivationSpec が作成されます。

## 汎用 JMS バインディングの管理

汎用 JMS バインディングはサーバーから管理できます。

管理コンソールを使用して、汎用 JMS インポートおよびエクスポート・バインディングを構成および管理します。

汎用 JMS インポートおよびエクスポート・バインディングの詳しい作成手順は、WebSphere Integration Developer インフォメーション・センターの「**WebSphere Integration Developer**」 > 「**Developing integration applications**」 > 「**Accessing external services with messaging systems**」 > 「**Generic JMS**」を参照してください。

### 汎用 JMS バインディングのプロパティ

汎用 JMS のインポート・バインディングおよびエクスポート・バインディングは、デプロイメント中すべての必要な接続ファクトリーが作成された状態でインストールできます。または、既存のリソースのセットを指すように構成することもできます。

一般に、汎用 JMS バインディングは WebSphere Integration Developer で作成されます。JMS バインディングに必要な接続と宛先を、コンポーネントがサーバーにインストールされるときに作成する方法を選択できます。または、JMS インポートまたはエクスポートで使用する予定のサーバー上のリソースの JNDI 名を指定できます。

このどちらの方法を選択するかによって、汎用 JMS バインディングの構成が異なります。

新しいメッセージ・プロバイダー・リソースが作成される場合（つまり、インストール時にサーバー上にリソースが作成される場合）は、リソースが存在するため、管理コンソールを使用してその位置を見つけ、管理できます。生成される成果物の JNDI 名を以下の表に示します。

表 21. 汎用 JMS インポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名

リソース	生成されたリソース JNDI 名
アウトバウンド接続	[moduleName]/[importName]_CF
応答接続	[moduleName]/[importName]_RESP_CF
送信宛先	[moduleName]/[importName]_SEND_D
受信宛先	[moduleName]/[importName]_RECEIVE_D
コールバック宛先	[moduleName]/[importName]_CALLBACK_D

表 22. 汎用 JMS エクスポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名

リソース	生成されたリソース JNDI 名
インバウンド接続	[moduleName]/[exportName]_LIS_CF
応答接続	[moduleName]/[exportName]_RESP_CF
受信宛先	[moduleName]/[exportName]_RECEIVE_D
送信宛先	[moduleName]/[exportName]_SEND_D
コールバック宛先	[moduleName]/[exportName]_CALLBACK_D

注: リソースは、サーバー・スコープで作成されます。管理コンソールのスコープは、初期設定では「すべてのスコープ」に設定されています。新規リソースを作成するには、このスコープを「セル」または「ノード」に設定する必要があります。デフォルトのリストから既存のリソースを選択できます。

他方のオプションを選択する場合は、JMS インポートが必要なリソースをサーバー上で検出することになるため、これらのリソースを事前にインストールし、インポート・ファイルおよびエクスポート・ファイルで JNDI 名を指定する必要があります。これにより、JMS バインディングとリソースの間の関連付けが行われます。

## 汎用 JMS バインディングの接続のセットアップ

汎用 JMS バインディングを使用するには、サード・パーティー JMS プロバイダーとの接続をセットアップする必要があります。

### 始める前に

管理コンソールでプロファイルに対して変更を行い、変更内容を保管する権限を持っている必要があります。WebSphere Integration Developer および WebSphere Application Server で変更を行い、保存するための適切な権限が必要です。

### このタスクについて

このタスクは手順の概要のみを示します。このトピックでは、個々のサード・パーティーの JMS プロバイダー固有の手順については説明しません。

このシナリオのアプリケーションは、汎用 JMS バインディングにより、メディエーション・コンポーネントの両端で他のアプリケーションに接続されています。このアプリケーションには、単一の両方向操作とのインターフェースが組み込まれています。

### 手順

1. サード・パーティー JMS プロバイダー固有のツールを使用して、キュー・マネージャー、キュー、および JMS 接続ファクトリーと宛先を作成するようにサード・パーティー JMS プロバイダーを構成します。
2. WebSphere Application Server で汎用メッセージング・プロバイダーを定義する必要があります。
3. WebSphere Integration Developer で、以下のタスクを実行する必要があります。
  - a. アプリケーションにエクスポートとインポートを追加し、以前に実装されたメディエーション・コンポーネントに接続します。

- b. 「バインディングの生成 (Generate binding)」 → 「メッセージング・バインディング (Messaging binding)」 → 「JMS バインディングの生成 (Generate JMS binding)」 を選択して、エクスポートとインポートの両方に汎用 JMS バインディングを追加します。
  - c. インポートとエクスポートの両方で、WebSphere Application Server で以前に定義されているプロパティと一致するように `genericMessagingProviderName` プロパティを設定します。
  - d. サード・パーティー JMS プロバイダーのツールでの定義と一致するように、接続と送信宛先/受信宛先の `ExternalJNDIName` を設定します。
4. アプリケーションを 1 つのサーバーにデプロイします。 サード・パーティー JMS プロバイダーのキュー・マネージャーが稼働しており、接続に対して使用可能であること、および WebSphere Application Server で汎用メッセージング・プロバイダー定義が指し示すコンテキストが使用可能であることを確認します。 WebSphere Integration Developer またはサービス・デプロイを使用して、アプリケーションの構築とデプロイを行います。
  5. アプリケーションを開始します。
  6. アプリケーションを実行します。

## タスクの結果

アプリケーションを実行するには、汎用 JMS エクスポートの受信宛先で定義されているサード・パーティーの JMS プロバイダー・キューにメッセージを格納します。応答が汎用 JMS エクスポート送信宛先に戻されます。

同様に、アプリケーションから汎用 JMS インポートの送信宛先に対して要求が発行されます。アプリケーションは、汎用 JMS インポートの受信宛先への応答を待機します。

## 汎用 JMS バインディングの構成

汎用 JMS インポート・バインディングとエクスポート・バインディングを管理することにより、リソースの特殊機能を構成することができます。管理用タスクは、管理コンソールを使用して実行します。

## 始める前に

管理コンソールでプロファイルを変更し、変更内容を保存するための権限が必要です。また、接続セットアップ手順を完了している必要があります。

## このタスクについて

汎用 JMS インポートまたはエクスポートは、サーバー・プロファイル内の Service Component Architecture (SCA) アプリケーションの一部としてインストールする必要があります。

## 手順

1. SCA モジュールを選択します。管理コンソールから「アプリケーション」 → 「SCA モジュール」を選択した後、`modulename` を選択します。構成ページが開きます。

2. 「モジュール・コンポーネント」の下で、「インポート」または「エクスポート」を展開します。インストールされているインポートまたはエクスポートのリストが表示されます。
3. インポートまたはエクスポートを展開した後、「バインディング」を展開します。管理可能なバインディングのリストが表示されます。
4. 管理するバインディングをクリックします。バインディングの一般プロパティが表示されます。
  - 「送信リソース」カテゴリには、「接続ファクトリー」と「送信宛先」が含まれています。
  - 「受信リソース」カテゴリには、「応答接続ファクトリー」、「リスナー・ポート」、および「アクティベーション・スペック」が含まれています。
  - 「拡張リソース」カテゴリには、コールバック・リソースとその他の使用可能なリソースが含まれています。

注: リソースにアクセスするもう 1 つの方法として、テキスト・ボックスに JNDI 名を入力する方法があります。ただしこの方法では、まだ構成されていないリソースの名前の入力も可能になります。

5. 必要なリソースを管理します。
  - a. 「参照」をクリックすると、JNDI 名のリストを含むウィンドウが開きます。このウィンドウで、必要な JNDI 名を選択します。選択された名前が、該当するテキスト・フィールドに取り込まれます。
  - b. 「構成」をクリックすると、この JNDI 名によって参照される、対応するパネルが開きます。ほとんどのリソースはクラスター有効範囲で構成できます。リスナー・ポートで「構成」オプションを選択すると、すべてのリスナー・ポートと、特定のクラスターのすべてのメンバーのクラスター・リスナー・ポート名を示すパネルが表示され、1 つのリスナー・ポートを選択できます。

「構成」を選択すると、対応する WebSphere Application Server パネルが開きます。
6. 設定を管理する準備ができている場合は、「適用」または「OK」をクリックします。

## 次のタスク

JNDI バインディングを反映するため、アプリケーションが自動的に再始動されます。

## 汎用 JMS ヘッダー

汎用 JMS ヘッダーは、すべての汎用 JMS メッセージ・プロパティを含むサービス・データ・オブジェクト (SDO) です。これらのプロパティはインバウンド・メッセージからのものか、アウトバウンド・メッセージに適用されるプロパティにすることができます。

このヘッダーは、システム・プログラミング・インターフェースを使用して提供されるため、ヘッダーを設定または取得するには、Service Component Architecture メッセージにアクセスします。ヘッダーは、クライアント・プログラムからは設定ま

たは取得できません。ただし、エンタープライズ・サービス・バス・メディエーションを使用して、ヘッダー値を取得するか、または値を発信要求に設定することができます。

methodBinding で静的に設定されるプロパティを以下に示します。

- JMSType
- JMSCorrelationID
- JMSDeliveryMode
- JMSPriority

また、汎用 JMS バインディングでは、JMS および MQ JMS バインディングと同じ方法で JMS ヘッダーとプロパティの動的変更がサポートされます。

一部の汎用 JMS プロバイダーでは、アプリケーションにより設定できるプロパティとその組み合わせが制限されます。詳しくは、サード・パーティー製品の資料を参照してください。ただし、methodBinding には、新しいプロパティ ignoreInvalidOutboundJMSProperties (例外の伝搬を許可するプロパティ) が追加されています。

汎用 JMS ヘッダーとメッセージのプロパティは、基本 Service Component Architecture の SCDL バインディング・スイッチがオンになっている場合のみ使用されます。スイッチがオンになっているとき、コンテキスト情報が伝搬されます。デフォルトでは、このスイッチはオンになっています。コンテキスト情報の伝搬を回避するには、値を false に変更します。

コンテキスト伝搬を使用可能にすると、ヘッダー情報をメッセージまたはターゲット・コンポーネントに流すことができます。コンテキスト伝搬のオン/オフを切り替えるには、インポート・バインディングおよびエクスポート・バインディングの contextPropagationEnabled 属性に true または false を指定します。以下に例を示します。

```
<esbBinding xsi:type="eis:JMSImportBinding" contextPropagationEnabled="true">
```

デフォルトは true です。

#### 関連概念

260 ページの『JMS ヘッダー』

JMS メッセージには、2 つのタイプのヘッダーが含まれます。1 つは JMS システム・ヘッダー、もう 1 つは複数の JMS プロパティです。メディエーション・モジュールでは、いずれのタイプのヘッダーにも、サービス・メッセージ・オブジェクト (SMO) 内、または ContextService API を使用してアクセスできません。

298 ページの『WebSphere MQ ヘッダー』

WebSphere MQ ヘッダーには、Service Component Architecture (SCA) メッセージへの変換に関する特定の規則が組み込まれています。

## 汎用 JMS バインディングのトラブルシューティング

汎用 JMS バインディングで発生した問題は、診断して修正することができます。

## 実装例外

汎用 JMS のインポート実装とエクスポート実装は、さまざまなエラー状態に応じて、以下の 2 種類の例外のいずれかを戻すことがあります。

- サービス・ビジネス例外: サービス・ビジネス・インターフェース (WSDL ポート・タイプ) で指定された障害が発生した場合に、この例外が戻されます。
- サービス・ランタイム例外: その他のすべてのケースで生成されます。ほとんどの場合、cause 例外には元の例外 (JMSEException) が含まれます。

## 汎用 JMS メッセージの有効期限のトラブルシューティング

JMS プロバイダーの要求メッセージには有効期限が設定されます。

要求有効期限 は、JMS プロバイダーの要求メッセージの JMSEExpiration の時刻に達した場合の、その要求メッセージの有効期限を指します。その他の JMS バインディングの場合と同様に汎用 JMS バインディングも、インポートによって格納されたコールバック・メッセージの有効期限を発信要求と同じ有効期限に設定することにより、要求の有効期限を処理しています。コールバック・メッセージの有効期限の通知により、要求メッセージの有効期限が切れていることが示されます。クライアントには、ビジネス例外により期限切れを通知する必要があります。

コールバック宛先がサード・パーティー・プロバイダーに移動すると、このタイプの要求有効期限はサポートされなくなります。

応答有効期限 は、JMS プロバイダーの応答メッセージの JMSEExpiration の時刻に達した場合の、その応答メッセージの有効期限を指します。

汎用 JMS バインディングでは応答有効期限はサポートされていません。これは、サード・パーティー JMS プロバイダーの正確な期限切れ動作が定義されていないためです。ただし、応答を受信する場合、応答を受信する時点で応答有効期限が切れていないかどうかを確認できます。

アウトバウンド要求メッセージでは、JMSEExpiration 値は待機時間と、asyncHeader の requestExpiration 値 (設定されている場合) から算出されます。

## 汎用 JMS 接続ファクトリー・エラーのトラブルシューティング

汎用 JMS プロバイダーで特定のタイプの接続ファクトリーを定義すると、アプリケーションの開始時にエラー・メッセージを受信することがあります。この問題を回避するには、外部接続ファクトリーを変更します。

アプリケーションの起動時に、以下のエラー・メッセージを受け取ります。

```
MDB リスナー・ポート JMSConnectionFactory タイプが JMSDestination タイプと一致しません (MDB Listener Port JMSConnectionFactory type does not match JMSDestination type)
```

この問題は、外部接続ファクトリーの定義時に発生します。特に、JMS 1.1 (統合) 接続ファクトリー (Point-to-Point 通信およびパブリッシュ/サブスクライブ通信の両方をサポート可能な接続ファクトリー) ではなく JMS 1.0.2 トピック接続ファクトリーを作成すると、例外がスローされることがあります。

この問題を解決するには、以下のステップを実行します。

1. 使用している汎用 JMS プロバイダーにアクセスします。
2. 定義されている JMS 1.0.2 トピック接続ファクトリーを JMS 1.1 (統合) 接続ファクトリーに置き換えます。

新規に定義した JMS 1.1 接続ファクトリーを使用してアプリケーションを起動すると、エラー・メッセージは表示されないはずで

## JMS ベース・イベント

JMS の対話の失敗によって SCA メッセージが発生した場合は、JMS プロバイダーの管理機能を使用してイベントを管理します。

## 例外の処理

バインディングがどのように構成されているかによって、データ・ハンドラーまたはデータ・バインディングによる例外の処理方法が決まります。また、メディエーション・フローの性質により、そのような例外がスローされたときのシステムの振る舞いが決まります。

データ・ハンドラーまたはデータ・バインディングがバインディングによって呼び出されるときには、さまざまな問題が発生する可能性があります。例えば、データ・ハンドラーが、ペイロードが破損しているメッセージを受信したり、誤った形式のメッセージを読み取るなどです。

バインディングによるそのような例外の処理方法は、データ・ハンドラーおよびデータ・バインディングの実装方法によって決まります。推奨される振る舞いは、`DataBindingException` をスローするようにデータ・バインディングを設計することです。

データ・ハンドラーの場合も同様の状況になります。データ・ハンドラーはデータ・バインディングによって呼び出されるため、すべてのデータ・ハンドラー例外はデータ・バインディング例外内にラップされます。したがって、`DataHandlerException` は `DataBindingException` として報告されます。

---

## WebSphere MQ JMS バインディング

WebSphere MQ JMS バインディングは、WebSphere MQ JMS ベースのプロバイダーを使用する外部アプリケーションとの統合を提供します。

WebSphere MQ JMS エクスポートおよびインポート・バインディングは、サーバー環境から外部 JMS または MQ JMS システムと直接統合する場合に使用します。これにより、サービス統合バスの MQ リンクまたはクライアント・リンク機能を使用する必要がなくなります。

コンポーネントがインポートによって WebSphere MQ JMS ベースのサービスと対話するときには、WebSphere MQ JMS インポート・バインディングはデータの送信先の宛先と応答を受信できる宛先を使用します。JMS メッセージと間のデータ変換は、JMS データ・ハンドラーまたはデータ・バインディング・エッジ・コンポーネントを介して行います。

SCA モジュールが WebSphere MQ JMS クライアントにサービスを提供するときには、WebSphere MQ JMS エクスポート・バインディングは、要求の受信と応答の送信ができる宛先を使用します。JMS メッセージとの間のデータ変換は、JMS ハンドラーまたはデータ・バインディングを介して行います。

関数セレクターは、呼び出すターゲット・コンポーネントに対する操作へのマッピングを提供します。

#### 関連概念

289 ページの『WebSphere MQ バインディング』

WebSphere MQ バインディングは、WebSphere MQ アプリケーションと Service Component Architecture (SCA) との接続を提供します。

#### 関連タスク



WebSphere Process Server におけるイベント順序付けの有効化

## WebSphere MQ JMS バインディング: 全体像

WebSphere MQ JMS バインディングは、WebSphere MQ JMS プロバイダーを使用する外部アプリケーションとの統合を提供します。

### WebSphere MQ 管理用タスク

WebSphere MQ システム管理者は、基盤となる WebSphere MQ キュー・マネージャーを作成する必要があります。このキュー・マネージャーは、WebSphere MQ JMS バインディングが含まれるアプリケーションを実行する前に、これらのバインディングによって使用されます。

### WebSphere MQ JMS インポート・バインディング

WebSphere MQ JMS インポートにより、SCA モジュール内のコンポーネントは、WebSphere MQ JMS ベースのプロバイダーが提供するサービスと通信できるようになります。サポートされているバージョンの WebSphere MQ を使用していることを確認してください。詳細なハードウェアおよびソフトウェア要件については、IBM サポート・ページを参照してください。

WebSphere MQ JMS インポート・バインディングでは、呼び出されている操作のタイプに応じた以下の 2 種類の使用シナリオがサポートされています。

- 片方向: WebSphere MQ JMS インポートは、インポート・バインディングに構成された送信宛先にメッセージを送信します。JMS ヘッダーの `replyTo` フィールドには何も送信しません。
- 両方向 (要求/応答): WebSphere MQ JMS インポートは、送信宛先にメッセージを送信します。

`receive` 宛先が `replyTo` ヘッダー・フィールドに設定されます。受信宛先で `listen` するためのメッセージ駆動型 Bean (MDB) をデプロイします。応答を受信すると、MDB は応答をコンポーネントに返します。

インポート・バインディングは、要求メッセージ ID (デフォルト) または要求メッセージ関連 ID から応答メッセージ関連 ID がコピーされていることを期待するように (WebSphere Integration Developer の「応答関連スキーム」フィールドを使用して) 構成することができます。

片方向と両方向のいずれのシナリオを使用する場合も、動的および静的ヘッダー・プロパティを指定できます。静的プロパティは JMS インポート・メソッド・バインディングから設定できます。これらのプロパティのなかには、SCA JMS ランタイムにとって特殊な意味を持つものがあります。

重要な点として、WebSphere MQ JMS は非同期バインディングであることに注意してください。呼び出し側コンポーネントが WebSphere MQ JMS インポートを同期的に呼び出すと (両方向操作の場合)、呼び出し側コンポーネントは、JMS サービスからの応答が返されるまでブロックされます。

図 21 は、インポートがどのように外部サービスにリンクされているのかを示しています。

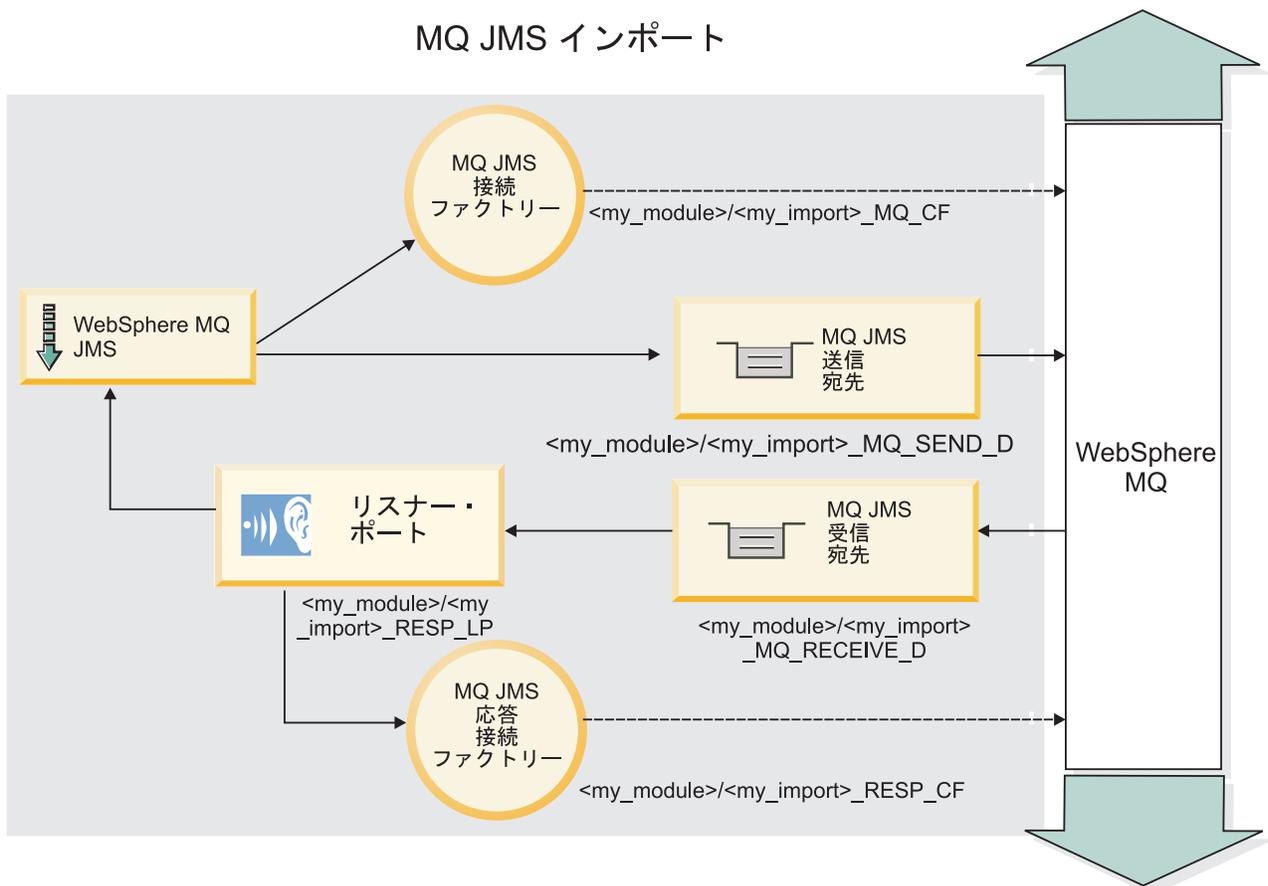


図 21. WebSphere MQ JMS インポート・バインディングのリソース

## WebSphere MQ JMS エクスポート・バインディング

WebSphere MQ JMS エクスポート・バインディングは、SCA モジュールが WebSphere MQ ベースの JMS プロバイダーで外部 JMS アプリケーションにサービスを提供する手段を提供します。

エクスポート・バインディングで指定された receive 宛先に着信する要求を listen するため、MDB がデプロイされます。send フィールドで指定された宛先は、呼び出されたコンポーネントが応答を返す場合にインバウンド要求に対する応答を送信するために使用されます。応答メッセージの replyTo フィールドで指定された値は、send フィールドで指定された宛先をオーバーライドします。

図 22 は、外部の要求側がどのようにエクスポートにリンクされているのかを示しています。

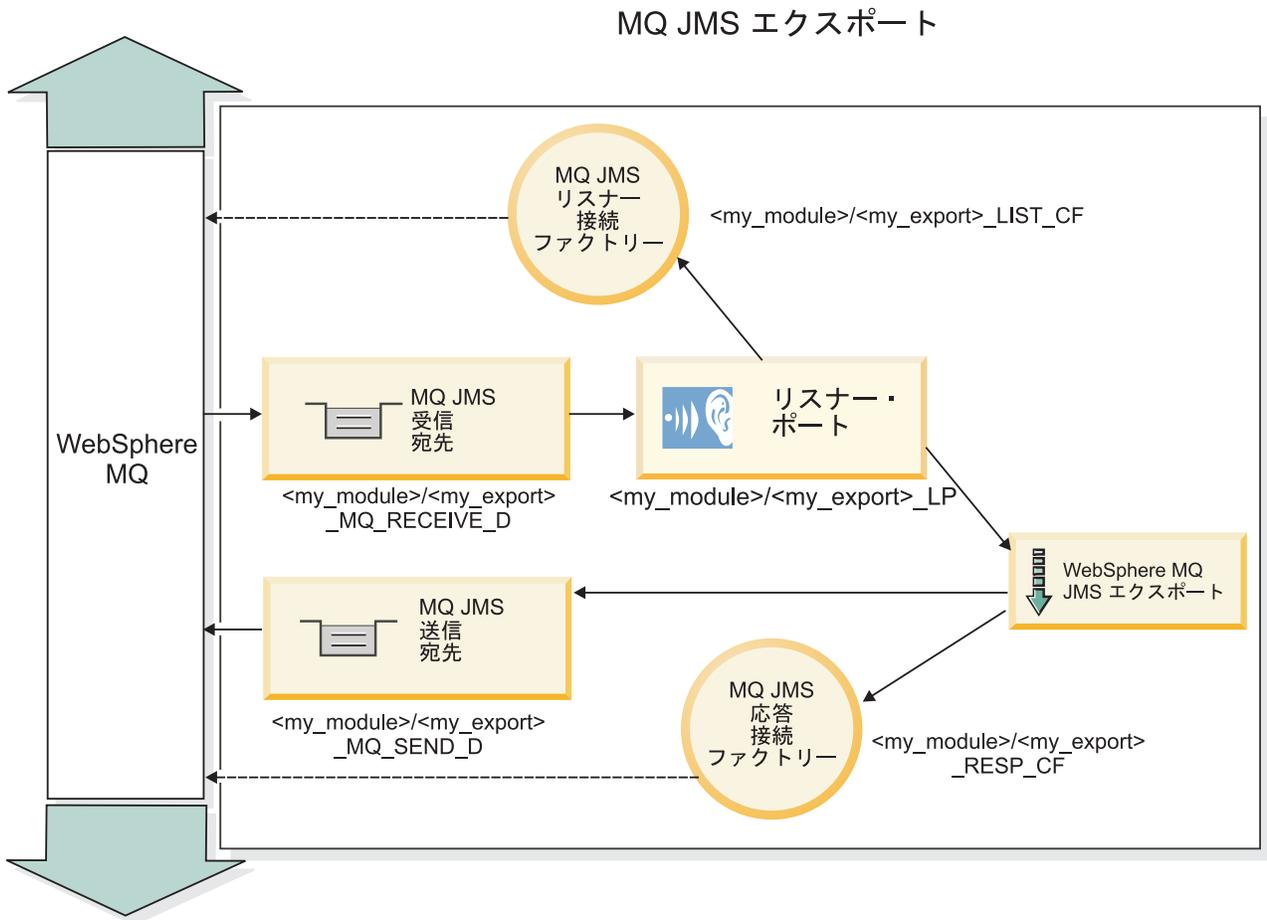


図 22. WebSphere MQ JMS エクスポート・バインディングのリソース

## WebSphere MQ JMS バインディングの主な特徴

WebSphere MQ JMS バインディングの主な特徴は、ヘッダー、J2EE 成果物、および作成される J2EE リソースです。

### ヘッダー

JMS メッセージ・ヘッダーには、いくつかの定義済みフィールドがあります。これらのフィールドには、クライアントとプロバイダーの両方でメッセージの識別と送

付に使用される値が格納されます。バインディング・プロパティを使用して、固定値でこれらのヘッダーを設定するか、またはヘッダーを実行時に動的に指定することができます。

### **JMSCorrelationID**

関連メッセージへのリンクです。通常、このフィールドは、応答の対象となるメッセージのメッセージ ID のストリングに設定されます。

### **TargetFunctionName**

このヘッダーは、呼び出された操作を識別する際に、指定されたいずれかの関数セクターによって使用されます。TargetFunctionName JMS ヘッダー・プロパティを JMS エクスポートに送信されたメッセージ内に設定することにより、この関数セクターが使用可能になります。このプロパティは、JMS クライアント・アプリケーションに直接設定することも、JMS バインディングが指定されたインポートを JMS エクスポートに接続する際に設定することもできます。この場合、JMS インポート・バインディングを構成し、操作名とのインターフェースにおいて、各操作ごとに TargetFunctionName ヘッダーを設定する必要があります。

## **関連スキーム**

WebSphere MQ JMS バインディングは、要求メッセージと応答メッセージを関連させる方法を決定するためのさまざまな関連スキームを提供します。

### **RequestMsgIDToCorrelID**

JMSMessageID は JMSCorrelationID フィールドにコピーされます。これはデフォルト設定です。

### **RequestCorrelIDToCorrelID**

JMSCorrelationID は JMSCorrelationID フィールドにコピーされます。

## **J2EE リソース**

MQ JMS インポートを J2EE 環境にデプロイすると、いくつかの J2EE リソースが作成されます。

### **パラメーター**

#### **MQ 接続ファクトリー**

クライアントが MQ JMS プロバイダーとの接続を作成するために使用します。

#### **応答接続ファクトリー**

送信宛先が受信宛先とは異なるキュー・マネージャー上にある場合に、SCA MQ JMS ランタイムが使用します。

#### **リスナー・ポート**

接続ファクトリー、つまり宛先とメッセージ駆動型 Bean の間の関連を指定します。これにより、ポートに関連付けられているデプロイ済みのメッセージ駆動型 Bean が、宛先からメッセージを取得できます。

#### **宛先**

- 送信宛先:
  - インポート: 要求または出力メッセージが送信される宛先です。

- エクスポート: 応答メッセージが送信される宛先です。ただし、着信メッセージの `JMSReplyTo` ヘッダー・フィールドにより置き換えられた場合は、その宛先が優先されます。
- 受信宛先:
  - インポート: 応答メッセージまたは着信メッセージが格納される宛先です。
  - エクスポート: 着信メッセージまたは要求メッセージが格納される宛先です。

## WebSphere MQ JMS バインディングの管理

WebSphere MQ JMS バインディングは、WebSphere Process Server で管理できます。

WebSphere MQ JMS バインディングにアクセスするには、管理コンソールを使用します。

WebSphere MQ JMS のインポートとエクスポートの生成方法については、WebSphere Integration Developer インフォメーション・センターで『MQ JMS バインディングの処理』を参照してください。

### 関連概念

294 ページの『WebSphere MQ バインディングの管理』

WebSphere MQ バインディングはサーバー内から管理できます。

### 関連タスク



WebSphere Process Server におけるイベント順序付けの有効化

## MQ JMS バインディングのプロパティ

MQ JMS バインディングは、デプロイメント中すべての必要な接続ファクトリーが作成された状態でインストールできます。または、既存のリソースのセットを指すように構成することもできます。

一般に、MQ JMS バインディングは WebSphere Integration Developer で作成されます。JMS バインディングに必要な接続と宛先を、コンポーネントがサーバーにインストールされるときに作成する方法を選択できます。または、MQ JMS インポートまたはエクスポートで使用する予定のサーバー上のリソースの JNDI 名を指定できます。

このどちらの方法を選択するかによって、MQ JMS バインディングの構成が異なります。

新しいメッセージ・プロバイダー・リソースが作成される場合 (つまり、インストール時にサーバー上にリソースが作成される場合) は、リソースが存在するため、管理コンソールを使用してその位置を見つけ、管理できます。生成される成果物の JNDI 名を以下の表に示します。

以下の表に、生成される成果物の JNDI 名の例を示します。

表 23. MQ JMS インポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名

リソース	モジュール名	インポート名	リソースのグローバル JNDI 名
アウトバウンド接続ファクトリー	mqjms.module	my/import	mqjms.module/my/import_MQ_CF
応答リスナー・ポート	mqjms.module	my/import	mqjms.module.my.import_RESP_LP (注: これは JNDI ではなく、単なる名前です)
応答接続ファクトリー	mqjms.module	my/import	mqjms.module/my/import_RESP_CF
送信	mqjms.module	my/import	mqjms.module/my/import_MQ_SEND_D
受信	mqjms.module	my/import	mqjms.module/my/export_MQ_RECEIVE_D
SIB コールバック宛先	mqjms.module	my/import	mqjms.module/my/import_MQ_CALLBACK_D
SIB コールバック接続ファクトリー	すべてのモジュール	my/import	SCA.MQJMS/Callback_CF

表 24. MQ JMS エクスポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名

リソース	モジュール名	エクスポート名	リソースのグローバル JNDI 名
リスナー・ポート	mqjms.module	my/export	mqjms.module.my.export_LP (注: これは JNDI ではなく、単なる名前です)
インバウンド接続ファクトリー	mqjms.module	my/export	mqjms.module/my/export_LIS_CF
応答接続ファクトリー	mqjms.module	my/export	mqjms.module/my/export_RESP_CF
受信	mqjms.module	my/export	mqjms.module/my/export_MQ_RECEIVE_D
送信	mqjms.module	my/export	mqjms.module/my/export_MQ_SEND_D
SIB コールバック宛先	mqjms.module	my/export	mqjms.module/my/export_MQ_CALLBACK_D
SIB コールバック接続ファクトリー	すべてのモジュール	my/export	SCA.MQJMS/Callback_CF

**注:**

- リソースは、サーバー・スコープで作成されます。管理コンソールでのデフォルトのスコープはセルです。リソースを見つけ、管理するためには、スコープを変更する必要があります。
- SIB コールバック宛先および SIB コールバック接続ファクトリーは、SIB JMS リソースです。表の中のその他の項目は、MQ JMS リソースです。この 2 つのタイプのリソースは、管理コンソールとは別に管理されます。

他方のオプションを選択する場合は、MQ JMS インポートまたはエクスポート・バイディングが使用するリソースをサーバー上で検出することになるため、これらのリソースを事前にインストールし、インポート・ファイルで JNDI 名を指定する必要があります。これにより、MQ JMS インポートとリソースの間の関連付けが行われます。

## MQ JMS バインディングの構成

MQ JMS バインディングを管理し、リソースの特殊機能を構成できます。管理用タスクは、管理コンソールを使用して実行します。

### 始める前に

管理コンソールでプロファイルに対して変更を行い、変更内容を保管する権限を持っている必要があります。

キューおよびキュー・マネージャーは自動的に生成されません。WebSphere MQ 管理者が WebSphere MQ 内で作成する必要があります。

### このタスクについて

MQ JMS インポートまたはエクスポートは、サーバー・プロファイル内の Service Component Architecture (SCA) アプリケーションの一部としてインストールする必要があります。

#### 手順

1. SCA モジュールを選択します。管理コンソールから「アプリケーション」→「SCA モジュール」をクリックした後、*modulename* をクリックします。構成ページが開きます。
2. 「モジュール・コンポーネント」の下で、「インポート」または「エクスポート」を展開します。インストールされているインポートまたはエクスポートのリストが表示されます。
3. インポートまたはエクスポートを展開した後、「バイディング」を展開します。管理可能なバイディングのリストが表示されます。
4. 管理するバイディングをクリックします。バイディングの一般プロパティが表示されます。
  - 「送信リソース」カテゴリーには、「接続ファクトリー」と「送信宛先」が含まれています。
  - 「受信リソース」カテゴリーには、「応答接続ファクトリー」、「リスナー・ポート」、および「アクティベーション・スペック」が含まれています。
  - 「拡張リソース」カテゴリーには、コールバック・リソースとその他の使用可能なリソースが含まれています。

**注:** リソースにアクセスするもう 1 つの方法として、テキスト・ボックスに JNDI 名を入力する方法があります。ただしこの方法では、まだ構成されていないリソースの名前の入力も可能になります。

5. 必要なリソースを管理します。

- a. 「参照」をクリックすると、JNDI 名のリストを含むウィンドウが開きます。このウィンドウで、必要な JNDI 名を選択します。選択された名前が、該当するテキスト・フィールドに取り込まれます。
  - b. 「構成」をクリックすると、この JNDI 名によって参照される、対応するパネルが開きます。ほとんどのリソースはクラスター有効範囲で構成できます。リスナー・ポートで「構成」オプションを選択すると、すべてのリスナー・ポートと、特定のクラスターのすべてのメンバーのクラスター・リスナー・ポート名を示すパネルが表示され、1 つのリスナー・ポートを選択できます。
- 「構成」を選択すると、対応する WebSphere Application Server パネルが開きます。
6. 設定を管理する準備ができている場合は、「適用」または「OK」をクリックします。

### 次のタスク

JNDI バインディングを反映するため、アプリケーションが自動的に再始動されます。

## JMS ヘッダー

JMS メッセージには、2 つのタイプのヘッダーが含まれます。1 つは JMS システム・ヘッダー、もう 1 つは複数の JMS プロパティです。メディエーション・モジュールでは、いずれのタイプのヘッダーにも、サービス・メッセージ・オブジェクト (SMO) 内、または ContextService API を使用してアクセスできます。

### JMS システム・ヘッダー

SMO では、JMS システム・ヘッダーは JMSHeader エlement によって表されます。この Element には、JMS ヘッダーに通常あるすべてのフィールドが含まれます。これらのフィールドはメディエーション (または ContextService) で変更できますが、SMO に設定された一部の JMS システム・ヘッダー・フィールドは、システムまたは静的な値によってオーバーライドされるため、アウトバウンド JMS メッセージでは伝搬されません。

メディエーション (または ContextService) で更新可能な JMS システム・ヘッダーのキー・フィールドには以下があります。

- **JMSType** および **JMSCorrelationID** – 特定の事前定義メッセージ・ヘッダー・プロパティの値
- **JMSDeliveryMode** – 送達モードの値 (persistent または nonpersistent。デフォルトは persistent)
- **JMSPriority** – 優先度の値 (0 から 9。デフォルトは JMS\_Default\_Priority)

### JMS プロパティ

JMS プロパティは、SMO ではプロパティ・リスト内のエントリとして表されます。プロパティは、メディエーション内、または ContextService API を使用して追加、更新、または削除できます。

プロパティは、JMS バインディングに静的に設定することもできます。静的に設定されたプロパティは、動的に設定される (同じ名前の) 設定をオーバーライドします。

他のバインディング (例えば HTTP バインディング) から伝搬されるユーザー・プロパティは、JMS バインディングでは JMS プロパティとして出力されます。

## ヘッダー伝搬の設定

JMS システム・ヘッダーおよびプロパティのインバウンド JMS メッセージからダウンストリームのコンポーネントへの伝搬、またはアップストリームのコンポーネントからアウトバウンド JMS メッセージへの伝搬は、バインディングのプロトコル・ヘッダー伝搬フラグで制御できます。

プロトコル・ヘッダー伝搬を設定すると、以下のリストに説明するようにヘッダー情報をメッセージまたはターゲット・コンポーネントに流すことができます。

- JMS エクスポート要求

メッセージ内で受信した JMS ヘッダーは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。メッセージ内で受信した JMS プロパティは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。

- JMS エクスポート応答

コンテキスト・サービスに設定されたすべての JMS ヘッダー・フィールドは、JMS エクスポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。コンテキスト・サービスに設定されたすべてプロパティは、JMS エクスポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。

- JMS インポート要求

コンテキスト・サービスに設定されたすべての JMS ヘッダー・フィールドは、JMS インポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。コンテキスト・サービスに設定されたすべてプロパティは、JMS インポート・バインディングに設定された静的プロパティによってオーバーライドされていない限り、アウトバウンド・メッセージ内で使用されます。

- JMS インポート応答

メッセージ内で受信した JMS ヘッダーは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。メッセージ内で受信した JMS プロパティは、コンテキスト・サービスを介してターゲット・コンポーネントに伝搬されます。

### 関連概念

274 ページの『汎用 JMS ヘッダー』

汎用 JMS ヘッダーは、すべての汎用 JMS メッセージ・プロパティを含むサ

ービス・データ・オブジェクト (SDO) です。これらのプロパティはインバウンド・メッセージからのものか、アウトバウンド・メッセージに適用されるプロパティにすることができます。

298 ページの『WebSphere MQ ヘッダー』

WebSphere MQ ヘッダーには、Service Component Architecture (SCA) メッセージへの変換に関する特定の規則が組み込まれています。

## 外部クライアント

サーバーは、WebSphere MQ JMS バインディングを使用して、外部クライアントとの間でメッセージを送受信できます。

外部クライアント (Web ポータルやエンタープライズ情報システムなど) は、エクスポートによって、アプリケーションの SCA コンポーネントにメッセージを送信できます。また、アプリケーション内の SCA コンポーネントがインポートによって外部クライアントを呼び出すこともできます。

WebSphere MQ JMS エクスポート・バインディングは、エクスポート・バインディングで指定された receive 宛先に着信する要求を listen するためのメッセージ駆動型 Bean (MDB) をデプロイします。send フィールドで指定された宛先は、呼び出されたアプリケーションが応答を返す場合にインバウンド要求に対する応答を送信するために使用されます。したがって、外部クライアントは、エクスポート・バインディングを介してアプリケーションを呼び出すことができます。

WebSphere MQ JMS インポートは外部クライアントにバインドし、外部クライアントにメッセージを配信できます。このメッセージは、外部クライアントからの応答を要求してもしなくても構いません。

WebSphere MQ を使用して外部クライアントと対話する方法については、WebSphere MQ インフォメーション・センターを参照してください。

## WebSphere MQ JMS バインディングのトラブルシューティング

WebSphere MQ JMS バインディングで発生した問題を診断し、修正できます。

### 実装例外

MQ JMS のインポートおよびエクスポート実装は、さまざまなエラー状態に応じて、以下の 2 種類の例外のうちのいずれかを戻すことがあります。

- サービス・ビジネス例外: サービス・ビジネス・インターフェース (WSDL ポート・タイプ) で指定された障害が発生した場合に、この例外が戻されます。
- サービス・ランタイム例外: その他のすべてのケースで生成されます。ほとんどの場合、cause 例外には元の例外 (JMSEException) が含まれます。

例えばインポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。そのため、複数の応答を受信した場合や遅延応答 (SCA の応答有効期限が切れた応答) を受信した場合は、サービス・ランタイム例外がスローされます。この場合、トランザクションはロールバックされ、応答メッセージはキューからバックアウトされるか、または Failed Event Manager によって処理されます。

## Failed Event Manager に表示されない JMS ベースの WebSphere MQ SCA メッセージ

WebSphere MQ JMS の対話の失敗によって SCA メッセージが発生した場合は、Failed Event Manager でこのメッセージを見つけることとなります。Failed Event Manager にこのようなメッセージが表示されない場合は、基礎となるリスナー・ポートの最大再試行回数プロパティの値が 1 以上になっているかどうかを確認してください。この値を 1 以上に設定すると、MQ JMS バインディングに対して SCA を呼び出す際に、Failed Event Manager と対話することができます。

### 誤用例: WebSphere MQ バインディングとの比較

WebSphere MQ JMS バインディングは、WebSphere MQ に対してデプロイされている JMS アプリケーションと相互協調処理するよう設計されています。これにより、メッセージは JMS メッセージ・モデルに基づいて公開されます。これに対し、WebSphere MQ インポートおよびエクスポートは、ネイティブ WebSphere MQ アプリケーションと相互協調処理することができ、WebSphere MQ メッセージ本体の内容全体をメディエーションに公開するように設計されています。

以下のシナリオでは、WebSphere MQ バインディングではなく WebSphere MQ JMS バインディングを使用して作成する必要があります。

- JMS メッセージ駆動型 Bean (MDB) を SCA モジュールから呼び出す。この MDB は、WebSphere MQ JMS プロバイダーに対してデプロイされています。WebSphere MQ JMS インポートを使用します。
- SCA モジュールを J2EE コンポーネント・サーブレットまたは EJB から JMS を介して呼び出すことができるようにする。WebSphere MQ JMS エクスポートを使用します。
- WebSphere MQ 上で転送中の JMS MapMessage の内容のメディエーションを実行する。WebSphere MQ JMS エクスポートとインポート、および適切なデータ・ハンドラーまたはデータ・バインディングを組み合わせ使用します。

WebSphere MQ バインディングと WebSphere MQ JMS バインディングの相互協調処理が予期される状況があります。特に、J2EE WebSphere MQ アプリケーションと非 J2EE WebSphere MQ アプリケーション間をブリッジングする場合は、WebSphere MQ エクスポートと WebSphere MQ JMS インポート (あるいはこの逆) を、適切なデータ・バインディングまたはメディエーション・モジュール (あるいはこの両方) と組み合わせ使用します。

### 例外の処理

バインディングがどのように構成されているかによって、データ・ハンドラーまたはデータ・バインディングによる例外の処理方法が決まります。また、メディエーション・フローの性質により、そのような例外がスローされたときのシステムの振る舞いが決まります。

データ・ハンドラーまたはデータ・バインディングがバインディングによって呼び出されるときには、さまざまな問題が発生する可能性があります。例えば、データ・ハンドラーが、ペイロードが破損しているメッセージを受信したり、誤った形式のメッセージを読み取るなどです。

バインディングによるそのような例外の処理方法は、データ・ハンドラーおよびデータ・バインディングの実装方法によって決まります。推奨される振る舞いは、`DataBindingException` をスローするようにデータ・バインディングを設計することです。

データ・ハンドラーの場合も同様の状況になります。データ・ハンドラーはデータ・バインディングによって呼び出されるため、すべてのデータ・ハンドラー例外はデータ・バインディング例外内にラップされます。したがって、`DataHandlerException` は `DataBindingException` として報告されます。

---

## WebSphere MQ バインディング

WebSphere MQ バインディングは、WebSphere MQ アプリケーションと Service Component Architecture (SCA) との接続を提供します。

WebSphere MQ エクスポートおよびインポート・バインディングは、サーバー環境から WebSphere MQ ベースのシステムと直接統合する場合に使用します。これにより、サービス統合バスの MQ リンクまたはクライアント・リンク機能を使用する必要がなくなります。

コンポーネントがインポートによって WebSphere MQ サービスと対話するときには、WebSphere MQ インポート・バインディングはデータの送信先のキューと応答を受信できるキューを使用します。

SCA モジュールが WebSphere MQ クライアントにサービスを提供するときには、WebSphere MQ エクスポート・バインディングは、要求の受信と応答の送信ができるキューを使用します。関数セレクターは、呼び出すターゲット・コンポーネントに対する操作へのマッピングを提供します。

MQ メッセージと間のペイロード・データの変換は、MQ 本体データ・ハンドラーまたはデータ・バインディングを使用して行われます。MQ メッセージと間のヘッダー・データの変換は、MQ ヘッダー・データ・バインディングを使用して行われます。

サポートされる WebSphere MQ バージョンについては、Web サイトの WebSphere Process Server システム要件を参照してください。

### 関連概念

277 ページの『WebSphere MQ JMS バインディング』

WebSphere MQ JMS バインディングは、WebSphere MQ JMS ベースのプロバイダーを使用する外部アプリケーションとの統合を提供します。

## WebSphere MQ バインディング: 全体像

WebSphere MQ バインディングにより、ネイティブ MQ ベースのアプリケーションとの統合が実現します。

### WebSphere MQ 管理用タスク

WebSphere MQ システム管理者は、基盤となる WebSphere MQ キュー・マネージャーを作成する必要があります。このキュー・マネージャーは、WebSphere MQ バ

インデイングが含まれるアプリケーションを実行する前に、これらのバインディングによって使用されます。

## WebSphere 管理用タスク

WebSphere MQ の MQ\_INSTALL\_ROOT 環境変数を、サーバーによってサポートされる WebSphere MQ バージョンに変更して、サーバーを再始動する必要があります。これにより、サポートされるバージョンの WebSphere MQ のライブラリーが使用されることが保証されます。詳細なハードウェアおよびソフトウェア要件については、IBM サポート・ページを参照してください。

## WebSphere MQ インポート・バインディング

WebSphere MQ インポート・バインディングにより、SCA モジュール内のコンポーネントは、外部の WebSphere MQ ベースのアプリケーションが提供するサービスと通信できるようになります。サポートされているバージョンの WebSphere MQ を使用していることを確認してください。詳細なハードウェアおよびソフトウェア要件については、IBM サポート・ページを参照してください。

外部 WebSphere MQ システムとの対話では、要求を送信し、応答を受信するためのキューが使用されます。

WebSphere MQ インポート・バインディングでは、呼び出されている操作のタイプに応じた以下の 2 種類の使用シナリオがサポートされています。

- 片方向: WebSphere MQ インポートは、インポート・バインディングの「送信宛先キュー (Send destination queue)」フィールドに構成されたキューにメッセージを送信します。MQMD ヘッダーの replyTo フィールドには何も送信しません。
- 両方向 (要求/応答): WebSphere MQ インポートは、「送信宛先キュー (Send destination queue)」フィールドに構成されたキューにメッセージを送信します。

receive キューは、replyTo MQMD ヘッダー・フィールドに設定されます。受信キューで listen するためのメッセージ駆動型 Bean (MDB) をデプロイします。応答を受信すると、MDB は応答をコンポーネントに返します。

インポート・バインディングは、要求メッセージ ID (デフォルト) または要求メッセージ関連 ID から応答メッセージ関連 ID がコピーされていることを期待するように (「応答関連スキーム」フィールドを使用して) 構成することができます。

重要な点として、JMS は非同期バインディングであることに注意してください。呼び出し側コンポーネントが JMS インポートを同期的に呼び出すと (両方向操作の場合)、呼び出し側コンポーネントは、JMS サービスからの応答が返されるまでブロックされます。

291 ページの図 23 は、インポートがどのように外部サービスにリンクされているのかを示しています。

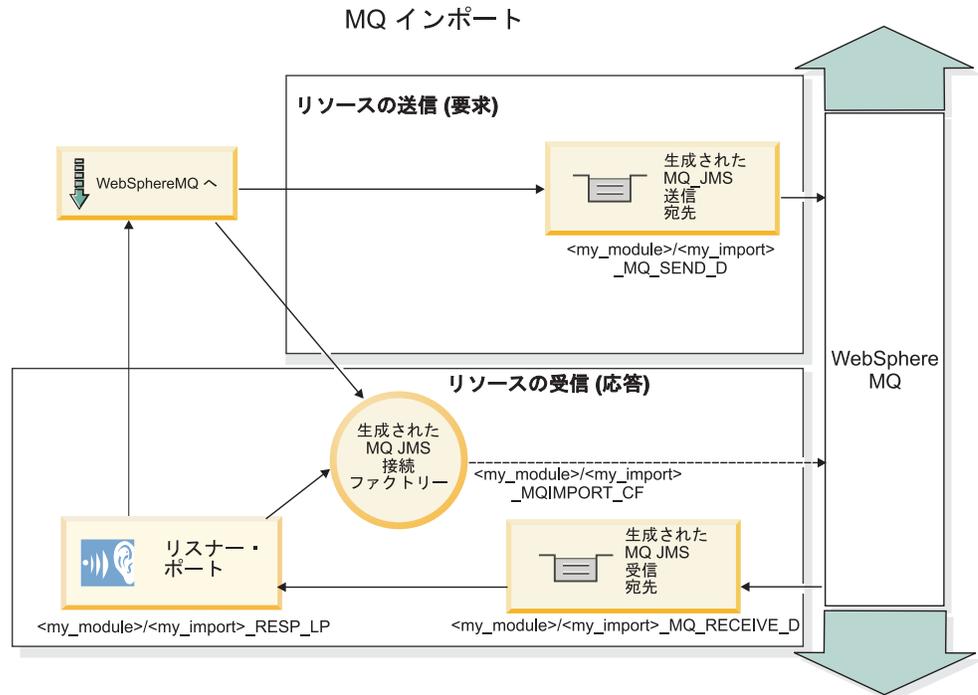


図 23. WebSphere MQ インポート・バインディングのリソース

## WebSphere MQ エクスポート・バインディング

WebSphere MQ エクスポート・バインディングは、SCA モジュールが外部の WebSphere MQ ベースにアプリケーションにサービスを提供する手段を提供します。

エクスポート・バインディングで指定された「受信宛先キュー (Receive destination queue)」に着信する要求を listen するため、MDB がデプロイされます。「送信宛先キュー (Send destination queue)」フィールドで指定されたキューは、呼び出されたコンポーネントが応答を返す場合にインバウンド要求に対する応答を送信するために使用されます。応答メッセージの replyTo フィールドで指定されたキューは、「送信宛先キュー (Send destination queue)」フィールドで指定されたキューをオーバーライドします。

292 ページの図 24 は、外部の要求側がどのようにエクスポートにリンクされているのかを示しています。

## MQ エクスポート

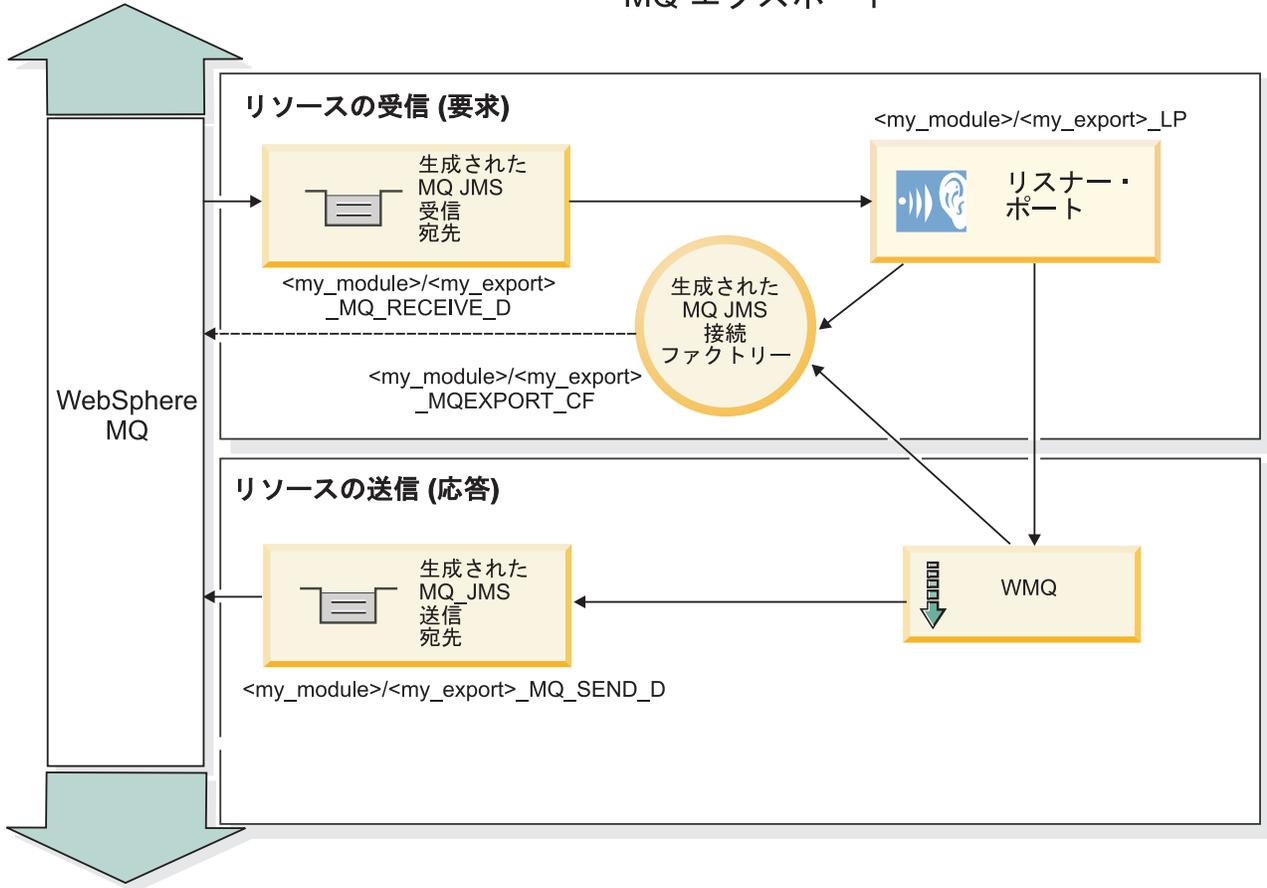


図 24. WebSphere MQ エクスポート・バインディングのリソース

## WebSphere MQ バインディングの主な特徴

WebSphere MQ バインディングの主な特徴は、ヘッダー、J2EE 成果物、および作成される J2EE リソースです。

### 関連スキーム

WebSphere MQ 要求/応答アプリケーションでは、応答メッセージと要求を関連付けるさまざまな手法の 1 つを使用できます。これらの手法には、MQMD MessageID フィールドと CorrelID フィールドが関連します。ほとんどの場合、要求側はキュー・マネージャーに MessageID を選択させ、応答アプリケーションがこれを応答の CorrelID にコピーすることを想定します。多くの場合、要求側と応答アプリケーションは、使用される関連手法を暗黙的に認識しています。応答アプリケーションは、要求の Report フィールドに設定されているさまざまなフラグの指示に従って、これらのフィールドを処理することもあります。

WebSphere MQ メッセージのエクスポート・バインディングを構成するときには、以下のオプションを使用できます。

応答の MsgId のオプション:

### 新規 MsgID (New MsgID)

キュー・マネージャーが応答の固有の MsgId を選択できるようにします (デフォルト)。

### 要求の MsgID からコピーする (Copy from Request MsgID)

要求の MsgId フィールドから MsgId フィールドをコピーします。

### SCA メッセージからコピーする (Copy from SCA message)

SCA 応答メッセージの WebSphere MQ ヘッダーに含まれる値を MsgId に設定します。この値が存在しない場合は、キュー・マネージャーに新規 Id を定義させます。

### Report オプションを使用 (As Report Option)

要求の MQMD の Report フィールドを検査して、MsgId の処理方法を認識します。MQRO\_NEW\_MSG\_ID オプションと MQRO\_PASS\_MSG\_ID オプションがサポートされています。これらのオプションは、それぞれ「新規 MsgID (New MsgID)」および「要求の MsgID からコピーする (Copy from Request MsgID)」と同様に動作します。

### 応答の CorrelId のオプション:

#### 要求の MsgID からコピーする (Copy from Request MsgID)

要求の MsgId フィールドから CorrelId フィールドをコピーします (デフォルト)。

#### 要求の CorrelID からコピーする (Copy from Request CorrelID)

要求の CorrelId フィールドから CorrelId フィールドをコピーします。

#### SCA メッセージからコピーする (Copy from SCA message)

SCA 応答メッセージの WebSphere MQ ヘッダーに含まれる値を CorrelId に設定します。または、この値が存在しない場合は、空白のままにします。

#### Report オプションを使用 (As Report Option)

要求の MQMD の Report フィールドを検査して、CorrelId の処理方法を認識します。MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID オプションと MQRO\_PASS\_CORREL\_ID オプションがサポートされています。これらのオプションは、それぞれ「要求の MsgID からコピーする (Copy from Request MsgID)」および「要求の CorrelID からコピーする (Copy from Request CorrelID)」と同様に動作します。

WebSphere MQ メッセージのインポート・バインディングを構成するときには、以下のオプションを使用できます。

### 要求の MsgId のオプション:

#### 新規 MsgID (New MsgID)

キュー・マネージャーが要求の固有の MsgId を選択できるようにします (デフォルト)。

#### SCA メッセージからコピーする (Copy from SCA message)

SCA 要求メッセージの WebSphere MQ ヘッダーに含まれる値を MsgId に設定します。この値が存在しない場合は、キュー・マネージャーに新規 Id を定義させます。

### 応答関連のオプション:

### 応答の CorrelID を MsgId からコピーする (Response has CorrelID copied from MsgId)

応答メッセージの CorrelId フィールドが、要求の MsgId ごとに設定されていることを想定します (デフォルト)。

### 応答の MsgID を MsgId からコピーする (Response has MsgID copied from MsgId)

応答メッセージの MsgId フィールドが、要求の MsgId ごとに設定されていることを想定します。

### 応答の CorrelID を CorrelId からコピーする (Response has CorrelID copied from CorrelId)

応答メッセージの CorrelId フィールドが、要求の CorrelId ごとに設定されていることを想定します。

## J2EE リソース

WebSphere MQ バインディングを J2EE 環境にデプロイすると、いくつかの J2EE リソースが作成されます。

### パラメーター

#### MQ 接続ファクトリー

クライアントが WebSphere MQ プロバイダーとの接続を作成するために使用します。

#### 応答接続ファクトリー

送信宛先が受信宛先とは異なるキュー・マネージャー上にある場合に、SCA MQ ランタイムが使用します。

#### リスナー・ポート

接続ファクトリー、つまり宛先とメッセージ駆動型 Bean の間の関連を指定します。これにより、ポートに関連付けられているデプロイ済みのメッセージ駆動型 Bean が、宛先からメッセージを取得できます。

#### 宛先

- 送信宛先: インポートの場合は、要求または出力メッセージが送信される宛先です。エクスポートの場合は、応答メッセージが送信される宛先です。ただし、着信メッセージの MQMD ReplyTo ヘッダー・フィールドにより置き換えられた場合は、その宛先が優先されます。
- 受信宛先: 応答/要求または着信メッセージが格納される宛先です。

## WebSphere MQ バインディングの管理

WebSphere MQ バインディングはサーバー内から管理できます。

WebSphere MQ バインディングにアクセスするには、管理コンソールを使用します。

WebSphere MQ インポートおよびエクスポートの作成手順については、WebSphere Integration Developer インフォメーション・センターの「**WebSphere Integration Developer**」 > 「**Developing integration applications**」 > 「**Accessing external services with messaging systems**」 > 「**WebSphere MQ**」 > 「**Working with MQ bindings**」を参照してください。

## 関連概念

282 ページの『WebSphere MQ JMS バインディングの管理』

WebSphere MQ JMS バインディングは、WebSphere Process Server で管理できます。

## 関連タスク

 WebSphere Process Server におけるイベント順序付けの有効化

## WebSphere MQ バインディングのプロパティ

WebSphere MQ バインディングは、デプロイメント中すべての必要な接続ファクトリーが作成された状態でインストールできます。または、既存のリソースのセットを指すように構成することもできます。

一般に、WebSphere MQ インポート・バインディングおよび WebSphere MQ エクスポート・バインディングは WebSphere Integration Developer で作成されます。WebSphere MQ バインディングに必要な接続と宛先を、コンポーネントがサーバーにインストールされるときに作成する方法を選択できます。または、WebSphere MQ バインディングで使用する予定のサーバー上のリソースの JNDI 名を指定できます。

このどちらの方法を選択するかによって、WebSphere MQ バインディングの構成が異なります。

新しいメッセージ・プロバイダー・リソースが作成される場合（つまり、インストール時にサーバー上にリソースが作成される場合）は、リソースが存在するため、管理コンソールを使用してその位置を見つけ、管理できます。

以下の表に、生成される成果物の JNDI 名の例を示します。

表 25. WebSphere MQ インポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名

リソース	モジュール名	インポート名	リソースのグローバル JNDI 名
アウトバウンド接続ファクトリー	mq.module	my/import	mq.module/my/import_MQ_CF
応答リスナー・ポート	mq.module	my/import	mq.module.my.import_RESP_LP (注: これは JNDI ではなく、単なる名前です)
応答接続ファクトリー	mq.module	my/import	mq.module/my/import_RESP_CF
送信	mq.module	my/import	mq.module/my/import_MQ_SEND_D
受信	mq.module	my/import	mq.module/my/export_MQ_RECEIVE_D
SIB コールバック宛先	mq.module	my/import	mq.module/my/import_MQ_CALLBACK_D
SIB コールバック接続ファクトリー	すべてのモジュール	my/import	SCA.MQ/Callback_CF

表 26. WebSphere MQ エクスポート: サーバーへのインストール時に作成されるリソースの名前および JNDI 名

リソース	モジュール名	エクスポート名	リソースのグローバル JNDI 名
リスナー・ポート	mq.module	my/export	mq.module.my.export_LP (注: これは JNDI ではなく、単なる名前です)
インバウンド接続ファクトリー	mq.module	my/export	mq.module/my/export_LIS_CF
応答接続ファクトリー	mq.module	my/export	mq.module/my/export_RESP_CF
受信	mq.module	my/export	mq.module/my/export_MQ_RECEIVE_D
送信	mq.module	my/export	mq.module/my/export_MQ_SEND_D
SIB コールバック宛先	mq.module	my/export	mq.module/my/export_MQ_CALLBACK_D
SIB コールバック接続ファクトリー	すべてのモジュール	my/export	SCA.MQ/Callback_CF

**注:**

- リソースは、サーバー・スコープで作成されます。管理コンソールでのデフォルトのスコープはセルです。リソースを見つけ、管理するためには、スコープを変更する必要があります。
- SIB コールバック宛先および SIB コールバック接続ファクトリーは、SIB JMS リソースです。表の中のその他の項目は、WebSphere MQ リソースです。この 2 つのタイプのリソースは、管理コンソールとは別に管理されます。

他方のオプションを選択する場合、WebSphere MQ バインディングが使用するリソースをサーバー上で検出することになるため、これらのリソースを事前にインストールし、インポート・ファイルまたはエクスポート・ファイルで JNDI 名を指定する必要があります。これにより、WebSphere MQ バインディングとリソースの間の関連付けが行われます。

### WebSphere MQ バインディングの構成

WebSphere MQ インポート・バインディングとエクスポート・バインディングを管理して、リソースの特殊機能を調整したり設定したりすることができます。管理用タスクは、管理コンソールを使用して実行します。

### 始める前に

管理コンソールでプロファイルに対して変更を行い、変更内容を保管する権限を持っている必要があります。

キューおよびキュー・マネージャーは自動的に生成されないため、WebSphere MQ 管理者が WebSphere MQ 内で作成する必要があります。

## このタスクについて

WebSphere MQ インポートは、サーバー・プロファイル内の Service Component Architecture (SCA) アプリケーションの一部としてインストールする必要があります。

### 手順

1. 管理コンソールのデフォルトのメッセージング・プロバイダー設定パネルを開きます。

「**JMS プロバイダー**」を展開して、「**WebSphere MQ**」をクリックします。

2. オプション: WebSphere MQ 接続ファクトリーを管理します。

追加プロパティのリストにある「**WebSphere MQ 接続ファクトリー**」をクリックします。このパネルには、WebSphere MQ 接続ファクトリーのリストが、その構成プロパティの要約とともに表示されます。管理の対象とする MQ 接続ファクトリーをクリックするか、「**新規**」をクリックして新しい接続ファクトリーを作成します。

続いて表示されるパネルを使用して、JMS プロバイダーとしての WebSphere MQ で使用する、選択した接続ファクトリーの構成プロパティを表示または変更します。この構成プロパティは、関連したキューへの接続を作成する方法を制御します。

アプリケーションのリソース参照のために、このプロパティをバインディングに設定します。既存のアプリケーション用のバインディングを変更しない場合は、これらのプロパティを見つけることができる J2C パネルでこの接続ファクトリーを探します。

3. オプション: WebSphere MQ キュー接続ファクトリーを管理します。

追加プロパティのリストにある「**WebSphere MQ キュー接続ファクトリー**」をクリックします。このパネルには、WebSphere MQ キュー接続ファクトリーのリストが、その構成プロパティの要約とともに表示されます。管理の対象とする WebSphere MQ キュー接続ファクトリーをクリックするか、「**新規**」をクリックして新しいキュー接続ファクトリーを作成します。

続いて表示されるパネルを使用して、WebSphere MQ の JMS プロバイダーで使用する、選択したキュー接続ファクトリーの構成を表示または変更します。この構成プロパティは、関連したキューへの接続を作成する方法を制御します。

WebSphere MQ キュー接続ファクトリーは、WebSphere MQ が Point-to-Point メッセージング用に提供するキューへの JMS 接続を作成するために使用されます。WebSphere MQ の JMS プロバイダー用のキュー接続ファクトリーを管理するには、WebSphere MQ キュー接続ファクトリーの管理オブジェクトを使用します。

4. オプション: WebSphere MQ キュー宛先を管理します。

追加プロパティのリストにある「**WebSphere MQ キューの宛先**」をクリックします。このパネルには、WebSphere MQ キュー宛先のリストが、その構成プロ

パティエの要約とともに表示されます。管理の対象とするキューの宛先をクリックするか、「新規」をクリックして新しい WebSphere MQ キューの宛先を作成します。

続いて表示されるパネルを使用して、WebSphere MQ をメッセージング・プロバイダーとする Point-to-Point メッセージングのための、選択したキュー宛先の構成プロパティを表示または変更します。

WebSphere MQ のキュー宛先は、キューのプロパティを構成するために使用されます。キューへの接続は、メッセージング・プロバイダーとしての WebSphere MQ 用の関連するキュー接続ファクトリーによって作成されます。

5. 変更内容をマスター・プロファイルに保管し、必要に応じてサーバーを再始動します。

## WebSphere MQ ヘッダー

WebSphere MQ ヘッダーには、Service Component Architecture (SCA) メッセージへの変換に関する特定の規則が組み込まれています。

WebSphere MQ メッセージは、システム・ヘッダー (MQMD)、場合によっては 1 つ以上のその他の MQ ヘッダー (システムまたはカスタム)、 およびメッセージ本体で構成されます。メッセージ内に複数のメッセージ・ヘッダーがある場合、ヘッダーの順序が意味を持ちます。

各ヘッダーには、次のヘッダーの構造を記述する情報が含まれています。MQMD は最初のヘッダーを記述します。

### MQ ヘッダーの解析方法

MQ ヘッダーの解析には、MQ ヘッダー・データ・バイndingが使用されます。以下のヘッダーは自動的にサポートされます。

- MQRFH
- MQRFH2
- MQCIH
- MQIIIH

MQH で始まるヘッダーについては、処理方法が異なります。ヘッダーの特定のフィールドは解析されません。これらのフィールドは未解析バイトとして維持されます。

その他の MQ ヘッダーについては、カスタム MQ ヘッダー・データ・バイndingを作成して解析することができます。

### MQ ヘッダーへのアクセス方法

製品内で MQ ヘッダーにアクセスするには、以下の 2 つの方法のいずれかを使用します。

- メディエーション内でサービス・メッセージ・オブジェクト (SMO) を使用
- ContextService API を使用

MQ ヘッダーは、内部では SMO MQHeader エlementによって表されます。MQHeader は MQControl を拡張するヘッダー・データのコンテナですが、anyType の値Elementが含まれています。これには MQMD、MQControl (MQ メッセージ本体制御情報)、およびその他の MQ ヘッダーのリストが含まれます。

- MQMD は、WebSphere MQ メッセージ記述の内容を表します。ただし、本体の構造とエンコードを定義する情報は含まれません。
- MQControl には、メッセージ本体の構造とエンコードを定義する情報が含まれています。
- MQHeaders には、MQHeader オブジェクトのリストが含まれています。

MQ ヘッダー・チェーンはアンワインドされるため、SMO 内部では各 MQ ヘッダーが独自の制御情報 (CCSID、Encoding、および Format) を持つこととなります。ヘッダーは簡単に追加または削除できます。他のヘッダー・データを変更する必要はありません。

#### 関連概念

260 ページの『JMS ヘッダー』

JMS メッセージには、2 つのタイプのヘッダーが含まれます。1 つは JMS システム・ヘッダー、もう 1 つは複数の JMS プロパティです。メディアエーション・モジュールでは、いずれのタイプのヘッダーにも、サービス・メッセージ・オブジェクト (SMO) 内、または ContextService API を使用してアクセスできません。

274 ページの『汎用 JMS ヘッダー』

汎用 JMS ヘッダーは、すべての汎用 JMS メッセージ・プロパティを含むサービス・データ・オブジェクト (SDO) です。これらのプロパティはインバウンド・メッセージからのものか、アウトバウンド・メッセージに適用されるプロパティにすることができます。

## Service Component Architecture モジュールと WebSphere MQ

SCA モジュールと WebSphere MQ キューを接続して、相互にサービスを提供することができます。

Service Component Architecture (SCA) モジュールは、他の SCA モジュールに対するのと同じように WebSphere MQ アプリケーションと通信できます。WebSphere MQ アプリケーションに要求を送信するモジュールは、そのアプリケーションに関連付けられた正しい応答および要求キューで構成されたインポートを使用します。同様に SCA モジュールは、適切なアプリケーション要求および応答キューで構成されたエクスポートを使用して、WebSphere MQ アプリケーションにサービスを提供できます。モジュールを作成するときは、SCA モジュールと WebSphere MQ キュー間の接続を定義します。

WebSphere MQ キュー・マネージャーの観点からは、SCA モジュールは通常の MQ クライアントであるかのように見えます。SCA モジュールの側からは、WebSphere MQ キューは他のサービスと同じように見えます。SCA モジュールと WebSphere MQ キューとの間のシールドを更に強化するには、SCA モジュールと WebSphere MQ キューの間にメディアエーション・モジュールを使用します。これに

より、メデイエーションが元の SCA 要求をターゲット・キューに適した正しい形式に変換し、応答が使用可能になったときにそれを処理できるようになります。

**制約事項:** インポートおよびエクスポート用に WebSphere MQ を構成する場合は、以下の点に注意してください。

- ターゲット・クライアントを **MQ** に設定した状態でキュー宛先を構成する必要があります。
- TCP/IP クライアント接続を使用して、WebSphere MQ に接続する必要があります。詳しくは、「WebSphere MQ 相互通信」を参照してください。
- チャネル圧縮は使用できません。
- データを正しく処理するには、必要なチャネル出口をすべて設定する必要があります。これを行うには、「WebSphere MQ メッセージング・プロバイダー」 → 「WebSphere MQ 接続ファクトリー・オブジェクト (WebSphere MQ connection factory objects)」 → 「カスタム・プロパティー」をクリックして、以下のカスタム・プロパティーを設定します。
  - **SENDEXIT**。このプロパティーには、値 `com.ibm.ws.sca.internal.mq.exit.MQInternalSendExitImpl` を設定します。
  - **RECEXIT**。このプロパティーには、値 `com.ibm.ws.sca.internal.mq.exit.MQInternalReceiveExitImpl` を設定します。
  - **SENDEXITINIT**。このプロパティーには、選択したテキスト・ストリングを設定します。このプロパティーを設定すると、トレースで出口を見分けやすくなります。
  - **RECEXITINIT**。このプロパティーには、選択したテキスト・ストリングを設定します。このプロパティーを設定すると、トレースで出口を見分けやすくなります。
- 出口では JMS MQ バインディングは禁止されるため、WebSphere MQ バインディングと Java Message Service (JMS) MQ バインディングに同じ接続ファクトリーを使用することはできません。

#### 関連情報



WebSphere MQ WebSphere の MQ 相互通信

## 外部クライアント

WebSphere Process Server は、WebSphere MQ バインディングを使用して、外部クライアントとの間でメッセージを送受信できます。

外部クライアント (Web ポータルやエンタープライズ情報システムなど) は、エクスポートによって、アプリケーションの SCA コンポーネントにメッセージを送信できます。また、アプリケーション内の SCA コンポーネントがインポートによって外部クライアントを呼び出すこともできます。

WebSphere MQ エクスポート・バインディングは、エクスポート・バインディングで指定された `receive` 宛先に着信する要求を `listen` するためのメッセージ駆動型 Bean (MDB) をデプロイします。`send` フィールドで指定された宛先は、呼び出されたアプリケーションが応答を返す場合にインバウンド要求に対する応答を送信する

ために使用されます。したがって、外部クライアントは、エクスポート・バインディングを介してアプリケーションを呼び出すことができます。

WebSphere MQ インポートは外部クライアントにバインドし、外部クライアントにメッセージを配信できます。このメッセージは、外部クライアントからの応答を要求してもしなくても構いません。

WebSphere MQ を使用して外部クライアントと対話する方法については、WebSphere MQ インフォメーション・センターを参照してください。

## WebSphere MQ バインディングのトラブルシューティング

WebSphere MQ バインディングで発生する障害または失敗の状態を診断し、このような状態を修正できます。

### 主な障害状態

WebSphere MQ バインディングの主な障害状態は、トランザクションの意味構造、WebSphere MQ 構成、またはその他のコンポーネントの既存の動作への参照に基づいて判別されます。主な障害状態は以下のとおりです。

- WebSphere MQ キュー・マネージャーまたはキューに接続できない。

WebSphere MQ に接続してメッセージを受信できない場合は、MDB リスナー・ポートを開始することができません。この状態は、WebSphere Application Server ログに記録されます。永続メッセージは、正常に取得されるまで (または WebSphere MQ により期限切れとなるまで) WebSphere MQ キューに残ります。

WebSphere MQ に接続してアウトバウンド・メッセージを送信できない場合は、送信操作を制御するトランザクションがロールバックされます。

- インバウンド・メッセージを解析できないか、アウトバウンド・メッセージを構成できない。

データ・バインディングが失敗すると、作業を制御するトランザクションがロールバックされます。

- アウトバウンド・メッセージを送信できない。

メッセージを送信できないと、関連するトランザクションがロールバックされません。

- 複数の応答メッセージまたは予期しない応答メッセージが返される。

インポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。そのため、複数の応答を受信した場合や遅延応答 (SCA の応答有効期限が切れた応答) を受信した場合は、サービス・ランタイム例外がスローされます。この場合、トランザクションはロールバックされ、応答メッセージはキューからバックアウトされるか、または Failed Event Manager によって処理されます。

### 誤用例: WebSphere MQ JMS バインディングとの比較

通常、WebSphere MQ インポートおよびエクスポートは、ネイティブ WebSphere MQ アプリケーションと相互協調処理し、WebSphere MQ メッセージ本体の内容を

体をメディエーションに公開するように設計されています。一方、WebSphere MQ JMS バインディングは、WebSphere MQ に対してデプロイされている JMS アプリケーションと相互協調処理するように設計されています。これにより、メッセージは JMS メッセージ・モデルに基づいて公開されます。

以下のシナリオでは、WebSphere MQ バインディングではなく WebSphere MQ JMS バインディングを使用して作成する必要があります。

- JMS メッセージ駆動型 Bean (MDB) を SCA モジュールから呼び出す。この MDB は、WebSphere MQ JMS プロバイダーに対してデプロイされています。WebSphere MQ JMS インポートを使用します。
- SCA モジュールを J2EE コンポーネント・サーブレットまたは EJB から JMS を介して呼び出すことができるようにする。WebSphere MQ JMS エクスポートを使用します。
- WebSphere MQ 上で転送中の JMS MapMessage の内容のメディエーションを実行する。WebSphere MQ JMS エクスポートとインポート、および適切なデータ・バインディングを組み合わせ使用します。

WebSphere MQ バインディングと WebSphere MQ JMS バインディングの相互協調処理が予期される状況があります。特に、J2EE WebSphere MQ アプリケーションと非 J2EE WebSphere MQ アプリケーション間をブリッジングする場合は、WebSphere MQ エクスポートと WebSphere MQ JMS インポート (あるいはこの逆) を、適切なデータ・バインディングまたはメディエーション・モジュール (あるいはこの両方) と組み合わせ使用します。

## 未配布メッセージ

構成エラーなどが原因で WebSphere MQ がメッセージを対象の宛先に配信できない場合、メッセージは指定されている送達不能キューに送信されます。

このとき、メッセージ本体の先頭には、送達不能ヘッダーが追加されます。このヘッダーには、失敗の原因や元の宛先などの情報が含まれています。

## 例外の処理

バインディングがどのように構成されているかによって、データ・ハンドラーまたはデータ・バインディングによる例外の処理方法が決まります。また、メディエーション・フローの性質により、そのような例外がスローされたときのシステムの振る舞いが決まります。

データ・ハンドラーまたはデータ・バインディングがバインディングによって呼び出されるときには、さまざまな問題が発生する可能性があります。例えば、データ・ハンドラーが、ペイロードが破損しているメッセージを受信したり、誤った形式のメッセージを読み取るなどです。

バインディングによるそのような例外の処理方法は、データ・ハンドラーおよびデータ・バインディングの実装方法によって決まります。推奨される振る舞いは、`DataBindingException` をスローするようにデータ・バインディングを設計することです。

データ・ハンドラーの場合も同様の状況になります。データ・ハンドラーはデータ・バインディングによって呼び出されるため、すべてのデータ・ハンドラー例外はデータ・バインディング例外内にラップされます。したがって、`DataHandlerException` は `DataBindingException` として報告されます。

## EJB バインディング

EJB インポート・バインディングによって、SCA コンポーネントは J2EE サーバー上で実行中の J2EE ビジネス・ロジックによって提供されたサービスを呼び出すことができます。

ご使用のビジネス・ロジックが EJB 2.1 サーバー上で実行中の場合、EJB インポート・バインディングは EJB 2.1 プログラミング・モデルを使用して J2EE ビジネス・ロジックを呼び出します。

### EJB バインディング: 全体像

EJB インポート・バインディングによって、Service Component Architecture (SCA) モジュールは、モジュール外部の EJB 実装を呼び出すことができます。

#### EJB インポート・バインディング

EJB インポート・バインディングでは、コンシュームする側のモジュールとインポート・コンポーネントとをバインドする方法を指定することによって、SCA モジュールが EJB 実装を呼び出すことができます。以下に例を示します。

- SCA コンポーネント -> EJB 2.1 import == RMI/IIOP ==> EJB 2.1

外部 EJB 実装 (モジュール外部の EJB 実装) からサービスをインポートすることによって、ユーザーはビジネス・ロジックを WebSphere Process Server 環境に接続でき、ビジネス・プロセスに参加することができます。EJB インポート・バインディングでは、以下の 2 つの使用シナリオがサポートされています。

- 片方向: インポートで指定された EJB にメッセージを送信します。応答は必要ありません。
- 両方向 (要求応答): インポートで指定された EJB にメッセージを送信します。メソッド呼び出しからの応答データは、呼び出し側のコンポーネントにメッセージとして返されます。

EJB インポートに含まれる主なコンポーネントを以下に示します。

`JAXWSDataHandler`、`EJBFaultSelector`、および `EJBImportFunctionSelector` は WebSphere Process Server によって提供されます。

表 27. EJB インポート・バインディングの主なコンポーネント

コンポーネント	目的
EJB 実装	ビジネス・インターフェースを実装する EJB
JNDI 名	EJB 参照を指定します

表 27. EJB インポート・バインディングの主なコンポーネント (続き)

コンポーネント	目的
305 ページの『JAX-WS データ・ハンドラー』	<p>Java オブジェクトをビジネス・オブジェクト (片方向) に、または応答データをビジネス・オブジェクト (両方向) に変化させるためにサーバーに必要なデータ変換を実行します。このデータ・ハンドラーは、JAX-WS および JAXB の仕様を使用して、SCA から Java へのデータをマップします。</p> <p><b>注:</b> JAX-WS 仕様で定義された XSD タイプから Java タイプへのマッピングをカスタマイズするには、ビジネス・ニーズに合うように JAXB アノテーションを変更します。JAXWSDataHandler は xs:any、xs:anyType、および xs:anySimpleType をサポートしません。現在のサポートは JAX-WS 1.0 仕様および JAXB 2.0 仕様に制限されています。</p>
307 ページの『EJB 障害セレクター』	<p>EJB 呼び出しの結果が、障害、ランタイム例外、または正常な応答のいずれかを判別します。障害が検出された場合、障害セレクターがネイティブの障害名をサーバーに返すため、JAXWSDataHandler が例外オブジェクトを障害ビジネス・オブジェクトに変換することができます。</p>
307 ページの『EJB インポート関数セレクター』	<p>EJB インポート上の SCA コンポーネントによって呼び出された WSDL メソッド名に基づき、EJB メソッド・タイプを導き出します。@WebMethod アノテーションが存在する場合はこれを使用して、WSDL メソッドのための正しい Java メソッド・マッピングを判別します。@WebMethod アノテーションがない場合、関数セレクターは WSDL メソッド名と同じ名前の Java メソッドを返します。</p>

ユーザーのシナリオで JAX-WS がサポートされず、一部のカスタム・マッピングが必要な場合、サポートがある場合に JAXWSDataHandler、SLSBFFaultSelector、および SLSBImportFunctionSelector によって実行されるタスクを実行するために、カスタムのデータ・ハンドラー、関数セレクター、および障害セレクターが必要になることがあります。これにはカスタム・マッピング・アルゴリズムによって通常実行されるマッピングが含まれます。

### データ・ハンドラー規約

データ変換が必要な状況において、EJB バインディングとデータ・ハンドラーの間の規約は Java Object 配列 (Object[]) を使用して実行されます。

アウトバウンド通信中は、以下のステップが実行されます。

1. EJB バインディングは、WSDL で指定された内容と一致するように、期待されるタイプ、期待されるエレメント、およびターゲット・メソッド名を `BindingContext` に設定します。
2. EJB バインディングが、データ変換の必要なデータ・オブジェクトのための変換メソッドを呼び出します。
3. データ・ハンドラーは、(メソッド内部での定義の順で) メソッドのパラメーターを表す `Object[]` を返します。
4. EJB バインディングは `Object[]` を使用して、ターゲット EJB インターフェース上のメソッドを呼び出します。

バインディングは EJB 呼び出しからの応答を処理するために `Object[]` を作成します。`Object[]` の最初のエレメントは、Java メソッド呼び出しからの戻り値です。その後の値はメソッドの入力パラメーターを表します。これは `In/Out` タイプおよび `Out` タイプのパラメーターをサポートするために必要です。`Out` タイプのパラメーターの場合、値は応答データ・オブジェクト内で返される必要があります。データ・ハンドラーは `Object[]` 内で検出した値を処理および変換して、データ・オブジェクトに応答を返します。

EJB バインディング・レベルで指定されたデータ・ハンドラーは、要求、応答、障害、およびランタイム例外を処理するために使用されます。障害については、`faultBindingType` 構成プロパティを指定することによって、障害ごとに特定のデータ・ハンドラーを指定できます。この設定は、EJB バインディング・レベルで指定された値をオーバーライドします。

## JAX-WS データ・ハンドラー

EJB バインディングに WSDL インターフェースがある場合、デフォルトで JAX-WS データ・ハンドラーが使用されます。このデータ・ハンドラーは JAX-WS (Java API for XML-based WebService) 仕様に準拠しており、WSDL メソッド呼び出しをインポート用の Java メソッド呼び出し (アウトバウンド処理) に変換するために使用します。またこれは、障害セクターが `ResponseType.RUNTIME` の例外を返したときに障害を処理することを目的とした、Java 例外から障害データ・オブジェクトおよびランタイム例外への変換も処理します。このデータ・ハンドラーは、JAX-WS 呼び出しを表す SOAP メッセージからデータ・オブジェクトへの変換には使用できません。

データ・ハンドラーは `xs:AnyType`、`xs:AnySimpleType`、および `xs:Any` のほか、他の XSD データ型もサポートします。`xs:Any` へのサポートを使用可能にするには、Java コードの bean プロパティで `@XmlAnyElement (lax=true)` を使用します。以下に例を示します。

```
public class TestType {
    private Object[] object;

    @XmlAnyElement (lax=true)
    public Object[] getObject() {
        return object;
    }

    public void setObject (Object[] object) {
        this.object=object;
    }
}
```

これによって、TestType のプロパティ・オブジェクトが xs:any フィールドになります。xs:any フィールド内で使用される Java クラスの値は、@XmlAnyElement アノテーションを持つ必要があります。例えば Address がオブジェクト配列の設定で使用される Java クラスである場合、Address クラスはアノテーション @XmlRootElement を持つ必要があります。

JAX-WS データ・ハンドラーには以下の制約が適用されます。

- ヘッダー属性の @WebParam アノテーションをサポートしない。
- ビジネス・オブジェクト・スキーマ・ファイル (XSD ファイル) の名前空間には、Java パッケージ名からのデフォルト・マッピングが含まれない。  
package-info.java のアノテーション @XMLSchema も機能しません。名前空間で XSD を作成するための方法は、@XmlType および XmlRootElement アノテーションを使用する方法だけです。@XmlRootElement は Java bean タイプのグローバル・エレメントのターゲット名前空間を定義します。
- EJB インポート・ウィザードは、関連しないクラスについては XSD ファイルを作成しません。バージョン 2.0 では @XmlSeeAlso アノテーションがサポートされないため、子クラスが親クラスから直接参照されない場合、XSD は作成されません。この問題の解決方法は、そのような子クラスのために SchemaGen を実行することです。SchemaGen は、所定の Java Bean の XSD ファイルを作成するために提供されたコマンド行ユーティリティです。これは WPS\_Install\_Home/bin ディレクトリにあります。この解決方法を実現させるためには、これらの XSD をビジネス・インテグレーション・モジュールに手動でコピーする必要があります。

## 障害ハンドラー規約

EJB バインディングと障害ハンドラーの間の規約は、Java オブジェクト配列 (Object[]) を使用して実行します。

インポートが正常な場合、バインディングは Object[] を組み立てます。Object[] の最初のエレメントは、Java メソッド呼び出しからの戻り値です。その後の値はメソッドの入力パラメータを表します。これは In/Out タイプおよび Out タイプのパラメータをサポートするために必要です。

例外シナリオの場合、バインディングは Object[] を組み立て、最初のエレメントはメソッドによってスローされた例外を表します。障害セクターは以下のどの値でも返すことができます。

表 28. 障害ハンドラーの戻り値

タイプ	戻り値
障害	ResponseType.FAULT
ランタイム例外	ResponseType.RUNTIME
通常応答	ResponseType.RESPONSE

障害セクターが ResponseType.FAULT の値を返す場合、getFaultName() はネイティブの障害名を返します。このネイティブの障害名は、対応する WSDL 障害名をモデルから判別し、正しい障害データ・ハンドラーを呼び出すためにバインディングによって使用されます。

## EJB 障害セレクター

この障害セレクターは、SLSB インポート・バインディングとの障害セレクター規約に従います。渡された `Object[]` に例外オブジェクトが含まれる場合、これは `ResponseType.BUSINESSFAULT` を返します。例外オブジェクトが、メソッド上で宣言されたどの例外タイプとも一致しない場合、`ResponseType.RUNTIME_EXCEPTION` を返します。それ以外の場合、`ResponseType.RESPONSE` を返します。

## インポート関数セレクター規約

SLSB バインディングとインポート関数セレクターの間の規約は、JAVA オブジェクト配列 (`Object[]`) を使用して実行します。

`Object[]` の最初の要素は、WSDL の名前です。`Object[]` の 2 番目の要素は入力ビジネス・オブジェクトです。関数セレクターは EJB インターフェースのメソッドを表す `java.lang.reflect.Method` オブジェクトを返します。

## EJB インポート関数セレクター

この関数セレクターは、EJB インポート・バインディングとの関数セレクター規約に従います。`Object[]` の最初の要素は、WSDL の名前を持つ Java メソッドです。これは JAX-WS アノテーションに従って所定の WSDL メソッドについての Java メソッドを導き出し、入力データ・オブジェクトのアノテーションを使用して、EJB インターフェースを定義する Java クラスの名前を導き出します。

## EJB バインディングの操作

EJB インポートはサーバーで管理できます。EJB インポートは管理コンソールで構成可能です。

EJB インポートの作成について詳しくは、WebSphere Integration Developer インフォメーション・センターで EJB バインディングの処理について扱われているセクションを参照してください。

## EJB バインディングの管理

EJB インポート・バインディングは、WebSphere 管理コンソールで構成できます。

### 始める前に

EJB バインディングの確認または編集を行うには、サーバー・プロファイル内の Service Component Architecture (SCA) アプリケーションの一部としてそのバインディングがインストールされている必要があります。

### このタスクについて

#### EJB インポート・バインディングの構成

EJB インポートで呼び出す EJB は、以下のいずれかの組み合わせで実行できます。どのシナリオでも、以下のようにして正しい JNDI 名を指定することが重要です。

表 29. EJB インポートの JNDI 名の構成

SLSB シナリオ	JNDI 構成情報
別の J2EE モジュールの WebSphere Process Server	グローバルの名前空間に合わせて EJB インポート・バインディングの JNDI 名を設定します。さらに、EJB インポート・バインディングで指定する JNDI 名が、J2EE モジュールのバインディング・ファイルで指定されている名前と一致していることも確認します。
リモートの WebSphere Process Server または WebSphere Application Server	WebSphere Process Server 管理コンソールを使用して、EJB バインディング・タイプの名前空間バインディングを作成します。そのため場所は、「環境」→「ネーミング」→「名前空間」にあります。名前空間バインディングの「名前空間」フィールドで指定する名前は、EJB インポートのバインディング構成で指定されている JNDI 名と一致していなければなりません。
リモートの J2EE サーバー (WebSphere Process Server または WebSphere Application Server 以外)	WebSphere Process Server 管理コンソールを使用して、J2EE サーバーに COSNaming インターフェースが用意されている場合はタイプ CORBA の名前空間バインディング、そうでない場合は間接タイプの名前空間バインディングを作成します。そのため場所は、「環境」→「ネーミング」→「名前空間」にあります。名前空間バインディングの「名前空間」フィールドで指定する名前は、EJB インポートのバインディング構成で指定されている JNDI 名と一致していなければなりません。

実装に WebSphere Application Server がかわっている場合は、WebSphere Application Server 管理コンソールを使用した追加の構成が必要になる可能性もあります。

注: EJB インポート・バインディングの JNDI 名以外のすべてのプロパティは読み取り専用です。DataHandler、FunctionSelector、FaultSelector の各プロパティは表示可能ですが、構成はできません。

WebSphere Process Server 管理コンソールを使用して、EJB インポートのプロパティの表示または構成を行うには、以下の手順を実行します。

#### 手順

1. SCA モジュールを選択します。管理コンソールから「アプリケーション」→「SCA モジュール」をクリックした後、*modulename* をクリックします。
2. 「モジュール・コンポーネント」の下で、「インポート」を展開します。インストールされているインポートのリストが表示されます。
3. インポートを展開した後、「バインディング」を展開します。管理可能なバインディングのリストが表示されます。
4. 確認対象または編集対象のバインディングをクリックします。

## 外部クライアント

WebSphere Process Server では EJB インポート・バインディングを使用して、Web ポータルまたはエンタープライズ情報システム (EIS) などの外部クライアントにメッセージを送信します。

サーバーから外部クライアントにメッセージを送信するとき、EJB インポートはコンシュームする側のモジュールとインポート・コンポーネントとをバインドする方法を指定します。EJB インポートが呼び出されると、EJB インポートは Enterprise Java Bean (EJB) を呼び出し、EJB インポートでメソッドを呼び出すたびに、WSDL の対応するメソッド (JAX-WS マッピングによって定義される) が EJB 上で呼び出されます。

---

## HTTP バインディング

HTTP バインディングは、Service Component Architecture (SCA) と HTTP の接続を提供する目的で設計されています。これにより、既存のまたは新規作成された HTTP アプリケーションをサービス指向アーキテクチャー (SOA) 環境内に組み込むことができます。

Hypertext Transfer Protocol (HTTP) は、Web 上での情報転送プロトコルとして広く使用されています。HTTP プロトコルを使用する外部アプリケーションを扱う場合には、HTTP バインディングが必要です。HTTP バインディングは、メッセージとして渡されたデータのネイティブ・フォーマットから SCA アプリケーションのビジネス・オブジェクトへの変換を処理します。着信メッセージの場合、HTTP バインディングは、ビジネス・オブジェクトとして渡されたデータを、外部アプリケーションが期待するネイティブ・フォーマットに変換することもできます。

以下に、HTTP バインディングを使用する一般的なシナリオをリストして説明します。

- SCA によってホストされるサービスは、HTTP インポートを使用して HTTP アプリケーションを呼び出すことができます。
- SCA によってホストされるサービスは、HTTP 対応アプリケーションとして自己公開することができます。その場合、HTTP クライアントは HTTP エクスポートを使用して、これらのサービスを使用できるようになります。
- WebSphere Process Server と WebSphere Enterprise Service Bus は、HTTP インフラストラクチャーで相互通信し、ユーザーが企業標準に従って通信を管理することを可能にします。
- WebSphere Process Server および WebSphere Enterprise Service Bus は、HTTP 通信のメディアエーターとしてメッセージを変換およびルーティングし、HTTP ネットワークを使用するアプリケーションの統合を強化することができます。
- WebSphere Process Server および WebSphere Enterprise Service Bus は、HTTP とその他のプロトコル (SOAP/HTTP Web サービス、Java Connector Architecture (JCA) ベースのリソース・アダプター、JMS など) との間のブリッジとして使用できます。

HTTP インポートおよびエクスポート・バインディングの作成に関する詳細情報については、WebSphere Integration Developerインフォメーション・センターを参照し

てください。『統合アプリケーションの開発』 → 『HTTP を使用した外部サービスへのアクセス』トピックを参照してください。

## HTTP バインディング: 全体像

HTTP バインディングは、HTTP がホストするアプリケーションに接続を提供します。HTTP アプリケーション間の通信をメディエーションし、既存の HTTP ベースのアプリケーションをモジュールから呼び出せるようにします。

### HTTP インポート・バインディング

HTTP インポート・バインディングは、Service Component Architecture (SCA) アプリケーションから HTTP サーバーまたはアプリケーションへのアウトバウンド接続を提供します。

インポートは、HTTP エンドポイントの URL を呼び出します。この URL は、以下の 2 つの方法のいずれかを指定できます。

- インポートの構成プロパティとして URL を指定します。
- HTTP ヘッダーで動的に URL を設定します。

本来、この呼び出しは常に同期呼び出しとなります。

HTTP 呼び出しは常に要求/応答となりますが、HTTP インポートは片方向操作と両方向操作の両方をサポートし、操作が片方向の場合には応答を無視します。

### HTTP エクスポート・バインディング

HTTP エクスポート・バインディングは、HTTP アプリケーションから SCA アプリケーションへのインバウンド接続を提供します。

URL は、HTTP エクスポートで定義されます。要求メッセージをエクスポートに送信する必要がある HTTP アプリケーションは、この URL を使用してエクスポートを呼び出します。

HTTP エクスポートは ping もサポートします。

### 実行時の HTTP バインディング

実行時に HTTP バインディングを使用するインポートは、メッセージ本体 (データの有無にかかわらず) を使用して、SCA アプリケーションから外部 Web サービスに要求を送信します。要求は、図 25 に示すように、SCA アプリケーションから外部 Web サービスに対して行われます。

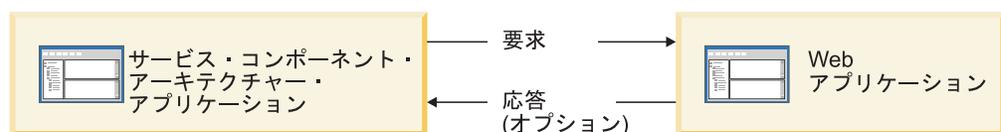


図 25. SCA アプリケーションから Web アプリケーションへの要求フロー

オプションで、HTTP バインディングを使用するインポートは要求に対する応答として Web アプリケーションからのデータを受信することができます。

エクスポートの場合は、図 26 に示すように、クライアント・アプリケーションが Web サービスに対して要求を行います。

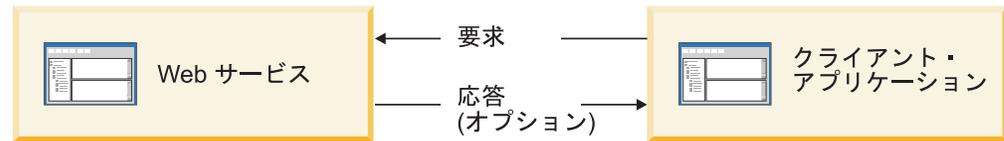


図 26. クライアント・アプリケーションから Web サービスへの要求フロー

Web サービスは、サーバーで稼働する Web アプリケーションです。この Web アプリケーションでは、エクスポートはサーブレットとして実装されるため、クライアントはその要求を URL アドレスに送信します。サーブレットは、実行時に要求を SCA アプリケーションに渡します。

オプションで、エクスポートは要求に対する応答として、クライアント・アプリケーションにデータを送信することができます。

## HTTP バインディングの管理

HTTP インポート・バインディングとエクスポート・バインディングを管理して、リソースを調整したり、リソースの特殊機能を設定したりすることができます。管理タスクは、管理コンソールを使用して実行します。

HTTP インポートおよびエクスポートを作成するには、WebSphere Integration Developer を使用します。

### HTTP バインディングの表示

アプリケーションのデプロイ後に、HTTP バインディングを検査して、それが正しいことを確認する必要がある場合があります。

#### 始める前に

管理コンソールを表示する必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティおよびロール・ベースの許可が有効になっている場合、このタスクを実行するには、オペレーター、管理者、またはコンフィギュレーターとしてログインする必要があります。

#### このタスクについて

既存の HTTP バインディングを変更する最初のステップとして、HTTP バインディングを表示します。

#### 手順

1. Service Component Architecture (SCA) アプリケーションを表示します。「アプリケーション」を展開して、「SCA モジュール」をクリックします。

インストール済みのアプリケーションが表示されます。

2. バインディングを表示する `module_name` をクリックします。

モジュールの構成が表示されます。

- 表示するバインディングに応じて、「インポート」または「エクスポート」を展開します。
- `path` を展開した後、「バインディング」を展開すると、モジュールに含まれるインポートまたはエクスポートのバインディングが表示されます。

含まれているインポート・バインディングまたはエクスポート・バインディングがすべて表示されます。

- バインディング構成を表示するバインディング名をクリックします。

### タスクの結果

選択したインポートまたはエクスポートのバインディングの構成が表示されます。

### 次のタスク

必要に応じてインポートまたはエクスポートを変更します。

### HTTP エクスポート・バインディングの変更

管理コンソールでは、HTTP エクスポート・バインディングの構成を変更するときに、元のソースを変更してからアプリケーションを再デプロイする必要はありません。

### 始める前に

『HTTP バインディングの表示』の説明に従って、モジュールの HTTP バインディングを表示します。

**このタスクに必要なセキュリティ・ロール:** セキュリティおよびロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

### このタスクについて

バインディングのメソッドが `ping` 可能であるかどうかを変更する必要があるとき、またはメソッドもしくはバインディングがサポートするエンコードを変更する必要があるときは、HTTP エクスポート・バインディングを変更します。

### 手順

- 『HTTP バインディングの表示』の説明に従って、モジュールのバインディングを表示します。
- 変更するバインディングを選択します。バインディングの名前をクリックします。
- バインディングの構成を変更します。

オプション	説明
バインディング・スコープの構成を変更するには	「バインディング・スコープ」タブをクリックします。
メソッド・スコープの構成を変更するには	「メソッド・スコープ」タブをクリックします。

両方の構成が存在するときは、メソッド・スコープの構成がバインディング・スコープの構成より優先されます。

4. 構成を変更した後、同じページに留まる場合は「適用」を、前のページに戻る場合は「OK」をクリックします。

## タスクの結果

バインディングの構成が変更されます。

**制約事項:** モジュールを再デプロイすると、構成が新しい構成で置き換えられます。各デプロイメントでこの変更をモジュールで維持するには、WebSphere Integration Developer を使用してモジュールのソース・コードを変更する必要があります。

## HTTP インポート・バインディングの変更

管理コンソールでは、HTTP インポート・バインディングの構成を変更するときに、元のソースを変更してからアプリケーションを再デプロイする必要はありません。

### 始める前に

『HTTP バインディングの表示』の説明に従って、モジュールの HTTP バインディングを表示します。

**このタスクに必要なセキュリティ・ロール:** セキュリティーおよびロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

### このタスクについて

Service Component Architecture (SCA) モジュールで使用される HTTP アプリケーションのバインディング・プロパティが変更されたときは、HTTP インポート・バインディングを変更します。

### 手順

1. 『HTTP バインディングの表示』の説明に従って、モジュールのバインディングを表示します。
2. 変更するバインディングを選択します。バインディングの名前をクリックします。
3. バインディングの構成を変更します。

オプション	説明
バインディング・スコープの構成を変更するには	「バインディング・スコープ」タブをクリックします。
メソッド・スコープの構成を変更するには	「メソッド・スコープ」タブをクリックします。

両方の構成が存在するときは、メソッド・スコープの構成がバインディング・スコープの構成より優先されます。

4. 構成を変更した後、同じページに留まる場合は「適用」を、前のページに戻る場合は「OK」をクリックします。

## タスクの結果

バインディングの構成が変更されます。

**制約事項:** モジュールを再デプロイすると、構成が新しい構成で置き換えられます。各デプロイメントでこの変更をモジュールで維持するには、WebSphere Integration Developer を使用してモジュールのソース・コードを変更する必要があります。

## HTTP ヘッダー

HTTP インポートおよびエクスポート・バインディングを使用すると、HTTP ヘッダーを構成し、その値をアウトバウンド・メッセージに使用できます。HTTP インポートはそれらのヘッダーを要求に使用し、HTTP エクスポートは応答に使用します。

静的に構成されたヘッダーと制御情報は、実行時に動的に設定される値よりも優先されます。しかし、動的オーバーライド URL、バージョン、およびメソッド制御の値は、静的な値をオーバーライドします。それ以外の場合、静的な値がデフォルト値と見なされます。

バインディングは、実行時に HTTP ターゲットの URL、バージョン、およびメソッド値を判断することにより、HTTP インポート URL の動的性をサポートします。これらの値は、エンドポイント参照、バージョン、メソッドの値を抽出して判別されます。

- エンドポイント参照については、`com.ibm.websphere.sca.addressing.EndpointReference` API を使用するか、または SMO ヘッダーの `/headers/SMOHeader/Target/address` フィールドを設定します。

**注:** 動的呼び出しについて、および URL のフォーマット、構文、使用方法について固有の情報は、『関連概念』を参照してください。

- バージョンおよびメソッドについては、Service Component Architecture (SCA) メッセージの HTTP 制御パラメーター・セクションを使用します。

HTTP エクスポートおよびインポート・バインディングのアウトバウンド・メッセージに関する制御およびヘッダー情報は、次の順序で処理されます。

1. HTTP 動的オーバーライド URL、バージョン、およびメソッドを SCA メッセージから除外するヘッダーおよび制御情報 (最低の優先順位)
2. エクスポート/インポート・レベルでの管理コンソールからの変更
3. エクスポート/インポートのメソッド・レベルでの管理コンソールからの変更
4. エンドポイント参照または SMO ヘッダーからの HTTP 動的オーバーライド URL
5. Service Component Architecture (SCA) メッセージからのバージョンおよびメソッド
6. データ・ハンドラーまたはデータ・バインディングからのヘッダーおよび制御情報 (最高の優先順位)

HTTP エクスポートおよびインポートは、contextPropagationEnabled が True に設定されている場合、インバウンド方向のヘッダーと制御パラメーターに、着信メッセージからのデータ (HTTPExportRequest および HTTPImportResponse) の取り込みのみを行います。逆に、アウトバウンドのヘッダーと制御パラメーター (HTTPExportResponse および HTTPImportRequest) については、contextPropagationEnabled が True に設定されている場合、HTTP エクスポートおよびインポートは読み取りと処理のみを行います。

注: インポート応答またはエクスポート要求のなかでデータ・ハンドラーまたはデータ・バインディングがヘッダーまたは制御パラメーターに変更されても、インポートまたはエクスポート・バインディング内でのメッセージの処理命令が変更されることはありません。そのため、データ・ハンドラーまたはデータ・バインディングは、変更された値をダウンストリームの SCA コンポーネントに伝搬する目的のみで使用する必要があります。

コンテキスト・サービスは、コンテキスト (HTTP ヘッダーなどのプロトコル・ヘッダーや、アカウント ID などのユーザー・コンテキストを含む) を Service Component Architecture (SCA) 呼び出しパスに沿って伝搬します。コンテキスト・サービスは、WebSphere Integration Developer の開発中に、インポートおよびエクスポート・プロパティに設定できます。詳しくは、WebSphere Integration Developer インフォメーション・センターのインポートおよびエクスポート・バインディングに関する情報を参照してください。

## 提供される HTTP ヘッダー構造とサポート

表 30 に、HTTPImport と HTTPExport の要求および応答の要求/応答パラメーターを項目別に示します。

表 30. 提供される HTTP ヘッダー情報

制御名	HTTPImport 要求	HTTPImport 応答	HTTPExport 要求	HTTPExport 応答
URL	無視される	設定されない	要求メッセージから読み取られる。 注: 照会ストリングも URL 制御パラメーターの一部です。	無視される
バージョン (可能な値: 1.0、1.1。デフォルトは 1.1)	無視される	設定されない	要求メッセージから読み取られる	無視される
メソッド	無視される	設定されない	要求メッセージから読み取られる	無視される

表 30. 提供される HTTP ヘッダー情報 (続き)

制御名	HTTPImport 要求	HTTPImport 応答	HTTPExport 要求	HTTPExport 応答
動的オーバーライド URL	データ・ハンドラーまたはデータ・バインディングに設定された場合、HTTP Import URL をオーバーライドする。要求行のメッセージに書き込まれる。 <b>注:</b> 照会ストリングも URL 制御パラメーターの一部です。	設定されない	設定されない	無視される
動的オーバーライド・バージョン	設定された場合、HTTP Import バージョンをオーバーライドする。要求行のメッセージに書き込まれる。	設定されない	設定されない	無視される
動的オーバーライド・メソッド	設定された場合、HTTP Import メソッドをオーバーライドする。要求行のメッセージに書き込まれる。	設定されない	設定されない	無視される
メディア・タイプ (この制御パラメーターは、Content-Type HTTP ヘッダーの値の一部を搬送します。)	存在する場合、メッセージに Content-Type ヘッダーの一部として書き込まれる。 <b>注:</b> この制御エレメント値は、データ・ハンドラーまたはデータ・バインディングによって提供してください。	応答メッセージ、Content-Type ヘッダーから読み取られる	要求メッセージ、Content-Type ヘッダーから読み取られる	存在する場合、メッセージに Content-Type ヘッダーの一部として書き込まれる。 <b>注:</b> この制御エレメント値は、データ・ハンドラーまたはデータ・バインディングによって提供してください。

表 30. 提供される HTTP ヘッダー情報 (続き)

制御名	HTTPImport 要求	HTTPImport 応答	HTTPExport 要求	HTTPExport 応答
文字セット (デフォルト: UTF-8)	存在する場合、メッセージに Content-Type ヘッダーの一部として書き込まれる。 注: この制御エレメント値は、データ・バインディングによって提供してください。	応答メッセージ、Content-Type ヘッダーから読み取られる	要求メッセージ、Content-Type ヘッダーから読み取られる	サポート対象。メッセージに Content-Type ヘッダーの一部として書き込まれる。 注: この制御エレメント値は、データ・バインディングによって提供してください。
転送エンコード (可能な値: chunked、identity。デフォルトは identity)	存在する場合、メッセージにヘッダーとして書き込まれ、メッセージ変換のエンコードの方法を制御する。	応答メッセージから読み取られる	要求メッセージから読み取られる	存在する場合、メッセージにヘッダーとして書き込まれ、メッセージ変換のエンコードの方法を制御する。
コンテンツ・エンコード (可能な値: gzip、x-gzip、deflate、identity。デフォルトは identity)	存在する場合、メッセージにヘッダーとして書き込まれ、ペイロードのエンコードの方法を制御する。	応答メッセージから読み取られる	要求メッセージから読み取られる	存在する場合、メッセージにヘッダーとして書き込まれ、ペイロードのエンコードの方法を制御する。
Content-Length	無視される	応答メッセージから読み取られる	要求メッセージから読み取られる	無視される
StatusCode (デフォルト: 200)	サポート対象外	応答メッセージから読み取られる	サポート対象外	存在する場合、応答行のメッセージに書き込まれる。
ReasonPhrase (デフォルト: OK)	サポート対象外	応答メッセージから読み取られる	サポート対象外	制御値は無視される。メッセージ応答行の値は StatusCode から生成される。

表 30. 提供される HTTP ヘッダー情報 (続き)

制御名	HTTPImport 要求	HTTPImport 応答	HTTPExport 要求	HTTPExport 応答
認証 (複数のプロパティーを含む)	存在する場合、基本認証ヘッダーの作成に使用される。 <b>注:</b> このヘッダーの値は、HTTP プロトコル上でのみエンコードされます。SCA では、これはデコードされ、平文として渡されます。	適用外	要求メッセージの基本認証ヘッダーから読み取られる。このヘッダーの存在は、ユーザーが認証済みであることを示すものではありません。認証は、サーブレット構成で制御する必要があります。 <b>注:</b> このヘッダーの値は、HTTP プロトコル上でのみエンコードされます。SCA では、これはデコードされ、平文として渡されます。	適用外
プロキシ (次のような複数のプロパティーを含む: Host、Port、Authentication)	存在する場合、プロキシによる接続を確立するために使用される。	適用外	適用外	適用外
SSL (次のような複数のプロパティーを含む: Keystore、Keystore Password、Trustore、Trustore Password、ClientAuth)	入力され、宛先 URL が HTTPS の場合、これは SSL による接続を確立するために使用される。	適用外	適用外	適用外

### 関連概念

#### 動的起動

WebSphere Process Server は、静的に定義されたエンドポイントを動的にオーバーライドしたり、ターゲット・インポートを使用して動的に起動することにより、メッセージの再ルーティングをサポートします。

---

## WebSphere Transformation Extender データ・ハンドラー

エクスポート経由で SCA モジュールに取り込まれるデータには、COBOL や EDI などさまざまなフォーマットのデータがあります。これらのデータは、サーバー・コンポーネントで処理する前に、データ・オブジェクトに変換する必要があります。同様に、サーバーからインポートを介して渡されるデータは、外部システムが認識できるフォーマットに変換する必要があります。WebSphere Transformation Extender でこのような変換を実行できます。

モジュールと外部の境界に位置するエクスポートとインポートにより、ネイティブ・データとデータ・オブジェクトの間の変換が処理されます。エクスポートとインポートには、変換に使用されるデータ・ハンドラーとデータ・バインディングが格納されます。

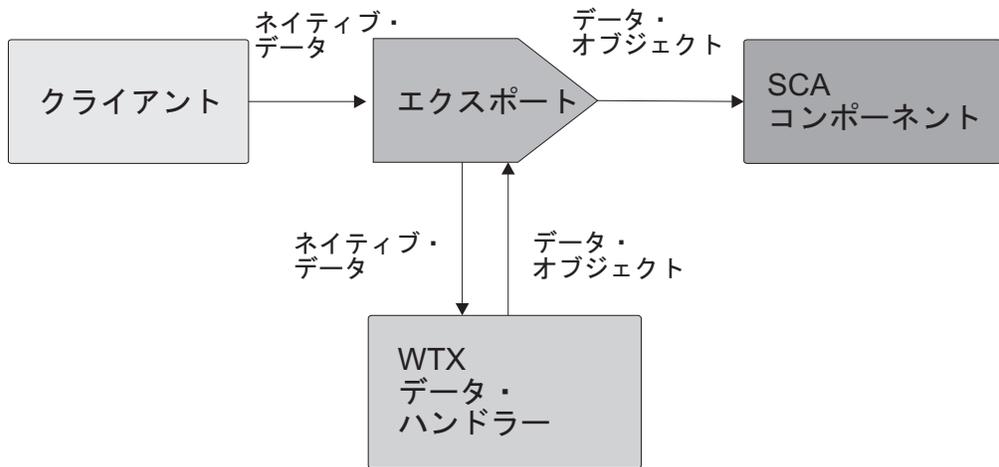
インポートとエクスポート内でネイティブ・データをさまざまなデータ・フォーマットに変換するには、プリパッケージされたデータ・ハンドラーやデータ・バインディングを使用するか、独自のデータ・バインディングを作成するか、または WebSphere Transformation Extender データ・ハンドラーを使用して、モジュールの境界でネイティブ・データとデータ・オブジェクト間の変換を実行します。

WebSphere Transformation Extender は、トランザクション指向のデータ統合ソリューションであり、ハンド・コーディングを必要とせずに、大容量の複雑なトランザクションの変換を自動化します。企業内に分散する複数のアプリケーション、データベース、メッセージング・ミドルウェア、通信テクノロジーからのデータを、リアルタイムに統合することができます。WebSphere Transformation Extender は、メタデータを利用したツール指向の手法でデータを変換します。

WebSphere Transformation Extender は独立したライセンス製品であり、WebSphere Process Server には含まれていません。

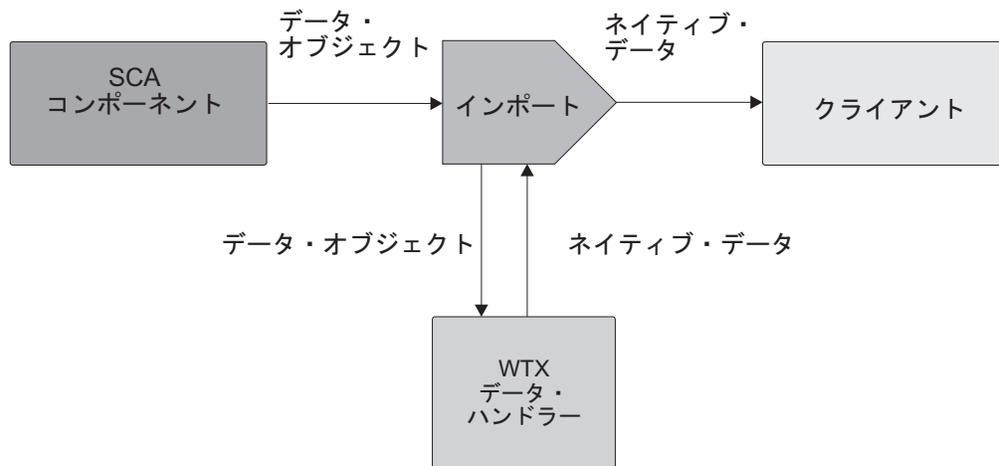
WebSphere Process Server と WebSphere Transformation Extender を連携させるには、WebSphere Transformation Extender for WebSphere Process Server のインストーラーを使用する必要があります。

以下の図は、WebSphere Transformation Extender と WebSphere Transformation Extender データ・ハンドラーを使用して、着信データをデータ・オブジェクトに変換する場合のエクスポート構成の概略図です。



クライアントは、任意のネイティブ・フォーマットでデータをエクスポートに渡します。データを受信したエクスポートは、WebSphere Transformation Extender データ・ハンドラーを介して、同じネイティブ・フォーマットでこのデータを WebSphere Transformation Extender に渡します。WebSphere Transformation Extender は、このデータをデータ・オブジェクト・フォーマットに変換してエクスポートに戻します。エクスポートは、このデータ・オブジェクトを該当する SCA コンポーネントに渡します。

以下の図は、WebSphere Transformation Extender と WebSphere Transformation Extender データ・ハンドラーを使用して、発信データをデータ・オブジェクト・フォーマットからクライアントに必要なネイティブ・フォーマットに変換する場合のインポート構成の概略図です。



SCA コンポーネントは、データ・オブジェクトをインポートに送信します。インポートは、WebSphere Transformation Extender データ・ハンドラーを介して、このデータ・オブジェクトを WebSphere Transformation Extender に渡します。WebSphere Transformation Extender は、この情報をクライアントのネイティブ・フォーマットに変換してインポートに戻します。インポートは、このネイティブ・データをクライアントに送信します。

注: WebSphere Transformation Extender の使用方法についての詳細は、WebSphere Transformation Extender 製品ライブラリーを参照してください。

## WebSphere Transformation Extender データ・ハンドラーのサポート

WebSphere Transformation Extender を使用すると、サーバーで必要なデータ・オブジェクトをさまざまなフォーマットにデータ変換することができます。ここでは、このようなデータ・ハンドラーのサポートについて説明します。

### サポートされているプラットフォーム

WebSphere Transformation Extender データ・ハンドラーをサポートするプラットフォームは、WebSphere Transformation Extender と WebSphere Process Server の両方でサポートされているプラットフォームのみです。この 2 つの製品でサポートされるプラットフォームのリストを『WebSphere Transformation Extender システム要件』と『WebSphere Process Server』で確認してください。

注: WebSphere Transformation Extender は独立したライセンス製品であり、WebSphere Process Server には含まれていません。

**i5/OS** WebSphere Transformation Extender は i5/OS ではサポートされていないため、このソリューションは i5/OS プラットフォームには適用されません。

WebSphere Transformation Extender のバージョンが 8.2 以上である必要があります。

### インポートおよびエクスポート

WebSphere Transformation Extender を使用したデータ変換をサポートするインポートとエクスポートを以下に示します。

- JMS
- 汎用 JMS
- WebSphere MQ JMS
- ネイティブ MQ (本体データ・バインディングのみ)
- EIS フラット・ファイル
- EIS FTP
- EIS E メール
- HTTP

WebSphere Transformation Extender データ・ハンドラーは、上記のすべてのインポートとエクスポートに対して適用することができます。これらのデータ・ハンドラーは、カスタマイズすることができます。詳しくは、WebSphere Integration Developer の資料を参照してください。

## WebSphere Transformation Extender マップおよびデータ・ハンドラー

WebSphere Transformation Extender データ・ハンドラーは、WebSphere Transformation Extender を呼び出します。WebSphere Transformation Extender では、データのフォーマット変換にマップが使用されます。マップは、システムに対

して変換の詳細を提供します。既存の WebSphere Transformation Extender マップをソリューションに適用できる場合は、既存のマップを使用できます。

WebSphere Transformation Extender データ・ハンドラーは、WebSphere Transformation Extender Engine のコンテキストで、WebSphere Transformation Extender のいずれかのマップを呼び出します。データ・ハンドラーは WebSphere Transformation Extender マップ名と入力データを WebSphere Transformation Extender Engine に送信し、このエンジンがフォーマット変換を実行します。変換の結果がデータ・ハンドラーに戻され、このバインディングが変換結果をエクスポートまたはインポートに公開します。

エクスポート側では、WebSphere Transformation Extender データ・ハンドラーが WebSphere Transformation Extender マップを呼び出して、ネイティブ・データ XML に変換します。データ・ハンドラーは次に XML をデータ・オブジェクトに非直列化します。また、インポートは、要求/応答シナリオの応答メッセージでこの方法でデータ・ハンドラーを使用します。

インポート側では、データ・ハンドラーがデータ・オブジェクトを XML に直列化し、この XML を WebSphere Transformation Extender マップに供給します。するとこのマップがネイティブ・データに変換して公開します。エクスポートでも、応答メッセージではこの方法でデータ・ハンドラーを使用します。

WebSphere Transformation Extender マップの作成方法の詳細については、WebSphere Transformation Extender の資料を参照してください。

マップ命名規則の作成、SCA モジュールでのマップの使用、および WebSphere Transformation Extender データ・ハンドラー の構成の詳細については、WebSphere Integration Developer の資料を参照してください。

## **WebSphere Transformation Extender データ・ハンドラーの使用 前の考慮事項**

ご使用の WebSphere Process Server で XML 以外のデータが送受信される場合は、WebSphere Transformation Extender データ・ハンドラーが理想的な選択です。WebSphere Transformation Extender を使用可能にしている必要があります。

WebSphere Process Server 用の WebSphere Transformation Extender をインストールする前に、WebSphere Process Server をインストールしている必要があります。また、WebSphere Transformation Extender 用の有効なライセンスを取得している必要もあります。WebSphere Transformation Extender は、独立した製品であるため、別にインストールする必要があります。

ご使用のサーバーを WebSphere Transformation Extender と連携して動作するようにするには、WebSphere Process Server 用の WebSphere Transformation Extender インストーラーを使用する必要があります。また、この処理では、WebSphere Transformation Extender がインストールされる他に、これが WebSphere Process Server にアクセス可能になるように、WebSphere Transformation Extender の Java クライアント・ライブラリーも WebSphere Process Server 製品内の OSGi バンドルとしてインストールされます。

WebSphere Process Server 用の WebSphere Transformation Extender のインストールには、以下の 2 つのステップがあります。

1. Software Developers Kit (SDK) 用の WebSphere Transformation Extender を実行します。これにより、WebSphere Transformation Extender 用の SDK と OSGi バンドルが WebSphere Process Server プラグインにインストールされます。
2. WebSphere Transformation Extender Design Studio 用のインストーラーを実行します。このステップでは、ツリー型とマップの設計および作成に必要なすべてのツールが提供されます。

**注:** WebSphere Transformation Extender データ・ハンドラーを利用するすべてのノードで 1 番目のステップを実行してください。2 番目のステップは、マップを作成および編集したいノードでのみ必要です。

WebSphere Transformation Extender のインストールでは、WebSphere Transformation Extender を指し示すシステム・パスを構成する必要があります。インストール場所 (例えば、C:\Program Files\IBM\WebSphere Transformation Extender 8.2) がシステム・パスの一部になっていることを確認してください。

WebSphere Transformation Extender データ・ハンドラーでは、Java Native Interface (JNI) を使用して、WebSphere Transformation Extender を呼び出します。このため、ご使用のシステムのパフォーマンスへ影響を与えることがあります。

WebSphere Transformation Extender データ・ハンドラーには、以下のメモリー要件があります。

- ネイティブ・データからビジネス・オブジェクトへの変換の場合、メモリーは、最低でもネイティブ・データを 2 倍したサイズに、直列化されたビジネス・オブジェクトを 2 倍したサイズを加えたものがが必要です。
- ビジネス・オブジェクトからネイティブ・データへの変換の場合、メモリーは、最低でも直列化されたビジネス・オブジェクトを 2 倍したサイズに、ネイティブ・データを 2 倍したサイズを加えたものがが必要です。

ご使用の WebSphere Process Server 環境で XML 以外のデータが送受信される場合は、WebSphere Transformation Extender データ・ハンドラーが理想的な選択です。XML データの場合、JMS、WebSphere MQ、および HTTP のインポートとエクスポートに XML データ・ハンドラーを使用し、EIS バインディングに XMLDataHandler を使用してください。

WebSphere Transformation Extender データ・ハンドラーの使用のためのその他の考慮事項。

- ご使用のデータが WebSphere Transformation Extender でサポートされる業界標準形式の場合は、WebSphere Transformation Extender データ・ハンドラーが理想的なソリューションです。業界パックのリストについては関連リンクを参照してください。
- 区切り文字、csv、および固定幅などの形式の場合は、WebSphere Transformation Extender データ・ハンドラーではなく、WebSphere Process Server の一部として提供されているデータ・バインディングとデータ・ハンドラーを使用してください。

- COBOL の場合は、COBOL のデータ・バインディング生成プログラムを使用して、COBOL データ・バインディングを生成します。
- 他のすべてのカスタム形式の場合は、メモリー要件、および作成の容易さについて WebSphere Transformation Extender マップとカスタム・データ・バインディングとを比較した結果に基づいて評価してください。

## データ・バインディング記述子の設定

WebSphere Transformation Extender データ・ハンドラーをインスタンス・レベルで構成するには、WebSphere Transformation Extender データ・ハンドラーの構成プロパティーではなく、WebSphere Transformation Extender データ・バインディング記述子を使用します。さまざまなインポートおよびエクスポートでは、特定の位置で WebSphere Transformation Extender データ・バインディング記述子を設定する必要があります。

### このタスクについて

WebSphere Transformation Extender の呼び出しでは、呼び出すマップ名を決定することが重要です。そのためには、WebSphere Integration Developer で WebSphere Transformation Extender データ・ハンドラーのプロパティーを構成します。バインディングのプロパティーで設定されたマップは、その操作のエクスポートまたはインポートに入るすべてのデータ・インスタンスに適用されます。例えば、操作 `Create(Customer)` があるエクスポートがあるとします。この操作を呼び出すたびに同じ WebSphere Transformation Extender マップを各顧客に対して使用するには、WebSphere Transformation Extender データ・ハンドラーのプロパティーを WebSphere Integration Developer で構成することをお勧めします。

ただし、着信顧客オブジェクトに応じて異なるマップを呼び出したいとします。データ・ハンドラーのプロパティーではこのような状況に対応した構成はできないため、代わりにデータ・バインディング記述子を使用する必要があります。

使用するインポートまたはエクスポートのタイプに基づいて、WebSphere Transformation Extender データ・バインディング記述子を以下の位置に挿入してください。

- JMS の WebSphere Transformation Extender データ・ハンドラーでは、記述子を **JMSType** または **JMS プロパティー DataBindingDescriptor** に設定する必要があります。

JMS エクスポートの場合:

- 要求の場合、着信メッセージの **JMSType**、または **JMS プロパティー DataBindingDescriptor** がクライアントにより設定される必要があります。

これらのプロパティーは、クライアントからサーバーにメッセージが送信される時点で設定されます。

- 応答の場合、インスタンス・レベルでバインディングを構成することはできません。構成プロパティーを使用して WebSphere Transformation Extender を呼び出す必要があります。

JMS インポートの場合:

- 要求の場合、インスタンス・レベルでバインディングを構成することはできません。構成プロパティを使用して WebSphere Transformation Extender を呼び出す必要があります。
- 応答の場合、着信メッセージの **JMSType**、または **JMS プロパティ DataBindingDescriptor** がクライアントにより設定される必要があります。

これらのプロパティは、クライアントからサーバーにメッセージが送信される時点で設定されます。

- WebSphere Transformation Extender Native MQ データ・ハンドラーでは、記述子を MQRFH2 ヘッダーに設定する必要があります。

Native MQ エクスポートの場合:

- 要求の場合、MQRFH2 ヘッダーの **DatabindingDescriptor** プロパティがクライアントにより設定される必要があります。
- 応答の場合、インスタンス・レベルでバインディングを構成することはできません。バインディング自体の構成プロパティを使用して WebSphere Transformation Extender を呼び出す必要があります。

Native MQ インポートの場合:

- 要求の場合、インスタンス・レベルでバインディングを構成することはできません。バインディング自体の構成プロパティを使用して WebSphere Transformation Extender を呼び出す必要があります。
  - 応答の場合、MQRFH2 ヘッダーの **DatabindingDescriptor** プロパティがクライアントにより設定される必要があります。
- WebSphere Transformation Extender HTTP データ・ハンドラーでは、カスタム・ヘッダー **DatabindingDescriptor** が必要です。このヘッダーがないと、BusinessObject および StreamType の URL のパラメーター、またはマップ名パラメーターが検索されます。例: `https://host[port]/sca module name/exportname/?businessObject=value1&contentType=value2`

HTTP エクスポートの場合:

- 要求の場合、カスタム・ヘッダー **DatabindingDescriptor** がクライアントにより設定されている必要があります。あるいは、URL でカスタム・プロパティを設定できます。
- 応答の場合、インスタンス・レベルでバインディングを構成することはできません。バインディング自体の構成プロパティを使用して WebSphere Transformation Extender を呼び出す必要があります。

HTTP インポートの場合:

- 要求の場合、インスタンス・レベルでバインディングを構成することはできません。バインディング自体の構成プロパティを使用して WebSphere Transformation Extender を呼び出す必要があります。
- 応答の場合、カスタム・ヘッダー **DatabindingDescriptor** がクライアントにより設定されている必要があります。

## WebSphere Transformation Extender データ・バインディング記述子

WebSphere Transformation Extender データ・バインディング記述子は、着信メッセージを記述する URI です。

### 目的

WebSphere Transformation Extender データ・バインディング記述子は、クライアントによって着信メッセージのヘッダーに設定されます。WebSphere Transformation Extender 内で着信メッセージに正しいマップを関連付けるために必要な情報が含まれています。

WebSphere Transformation Extender データ・バインディング記述子を使用する方法は、マップの呼び出しを設定する 2 つの方法のうちの 1 つです。もう 1 つの方法は、WebSphere Integration Developer で WebSphere Transformation Extender データ・ハンドラーのプロパティを構成する方法です。ただし、操作の呼び出しごとに異なるマップを使用する場合は、WebSphere Transformation Extender データ・バインディング記述子を使用してください。

例えば、操作 `Create(Customer)` があるエクスポートがあるとします。この操作を呼び出すたびに同じ WebSphere Transformation Extender マップを各顧客に対して使用するには、WebSphere Transformation Extender データ・ハンドラーのプロパティを WebSphere Integration Developer で構成することをお勧めします。

ただし、顧客に応じて異なるマップを呼び出したいとします。この場合、顧客を指定して操作が呼び出され、呼び出しごとに異なるマップが使用されます。データ・バインディングのプロパティではこのような状況に対応した構成はできないため、代わりにデータ・バインディング記述子を使用する必要があります。

### 構文

WebSphere Transformation Extender データ・バインディング記述子の構文を以下に示します。

**databinding://domain/property?queryParameters**

*domain* は、このデータ・バインディングの場合 **WTX** です。

*property* の値は、この場合 **map** です。

*queryParameters* は、`name=mapname` (*mapname* は必要なマップの名前) または `businessObject=Customer&contentType=format` (*Customer* はビジネス・オブジェクトの名前、*format* はデータ・ストリームのフォーマット (COBOL、EDI など)) です。

### 例

WebSphere Transformation Extender データ・バインディング記述子の例を以下に示します。

```
databinding://WTX/map?name=WTX/CustomerToCOBOL
```

注: WTX というディレクトリーがモジュールの最上位に存在し、このバインディングに必要なすべてのマップがこのディレクトリーに含まれている必要があります。

```
databinding://WTX/map?businessObject=Customer&contentType=COBOL
```

```
databinding://WTX/map?contentType=COBOL
```

## EIS バインディングのバインディング記述子の設定

EIS インポート・バインディングまたは EIS エクスポート・バインディングには、データ・バインディング記述子を設定することはできません。

### このタスクについて

EIS インポートおよびエクスポートを使用する場合は、EIS バインディング定義の構成プロパティーを使用して、バインディングと WebSphere Transformation Extender の間の対話を定義してください。接続をインスタンス・レベルで構成することはできません。

## JMS インポート・バインディングのバインディング記述子の設定

JMS インポート・バインディングは、メッセージの JMSType または JMSProperties 部分の WebSphere Transformation Extender データ・バインディング記述子に渡すことができます。

### 始める前に

JMS インポート、汎用 JMS インポート、MQ JMS インポートには WebSphere Transformation Extender JMS データ・バインディングを使用します。

WebSphere Transformation Extender データ・バインディング記述子は、JMS インポートの応答メッセージでのみ設定できます。

### このタスクについて

メッセージを個別に構成する場合は、JMS インポートの応答メッセージにデータ・バインディング記述子を設定します。

メッセージを個別に構成する必要がない場合は、バインディング自体の中でカスタム・プロパティーを使用して、WebSphere Transformation Extender との関連付けを作成しておくことをお勧めします。

### 手順

1. 要求メッセージをインスタンス・レベルで構成することはできません。バインディングのカスタム・プロパティーを使用して、使用する特定の WebSphere Transformation Extender マップとの関連付けを作成する必要があります。
2. 応答メッセージを構成します。 応答メッセージを構成するには、以下の 2 つの方法があります。

オプション	例
着信メッセージの JMSType を設定します。	<pre>jmsMessage.setJMSType("databinding://WTX/map?businessObject=Customer&amp;contentType=COBOL");</pre> この例には改行が追加されています。

オプション	例
<b>JMS プロパティ</b> <b>DataBindingDescriptor</b> を設定 します。	<code>jmsMessage.setStringProperty ("DataBindingDescriptor", "databinding: //WTX/map?businessObject=Customer&amp;contentType=COBOL");</code> この例には改行が追加されています。

これらのプロパティは、クライアントから WebSphere Process Server にメッセージが送信される時点で設定されます。

## JMS エクスポート・バインディングのバインディング記述子の設定

JMS エクスポート・バインディングは、メッセージの JMSType または JMSProperties 部分の WebSphere Transformation Extender データ・バインディング記述子に渡すことができます。

### 始める前に

JMS エクスポート、汎用 JMS エクスポート、MQ JMS エクスポートには WebSphere Transformation Extender JMS データ・バインディングを使用します。

WebSphere Transformation Extender データ・バインディング記述子は、JMS エクスポートの応答メッセージでのみ設定できます。

### このタスクについて

メッセージを個別に構成する場合は、JMS エクスポートの応答メッセージにデータ・バインディング記述子を設定します。

メッセージを個別に構成する必要がない場合は、バインディング自体の中でカスタム・プロパティを使用して、WebSphere Transformation Extender との関連付けを作成しておくことをお勧めします。

### 手順

1. 要求メッセージを構成します。 要求メッセージを構成するには、以下の 2 つの方法があります。

オプション	例
着信メッセージの JMSType を設定します。	<code>jmsMessage.setJMSType("databinding: //WTX/map?businessObject=Customer&amp;contentType=COBOL");</code> この例には改行が追加されています。
<b>JMS プロパティ</b> <b>DataBindingDescriptor</b> を設定 します。	<code>jmsMessage.setStringProperty ("DataBindingDescriptor", "databinding: //WTX/map?businessObject=Customer&amp;contentType=COBOL");</code> この例には改行が追加されています。

2. 応答メッセージをインスタンス・レベルで構成することはできません。 JMS エクスポート応答メッセージに対して WebSphere Transformation Extender 変換を使用するには、エクスポートの構成プロパティを設定する必要があります。

## ネイティブ MQ インポートのバインディング記述子の設定

WebSphere Transformation Extender と特定の方法で対話するように各メッセージを構成するには、WebSphere Transformation Extender データ・バインディング記述子

を介して WebSphere Transformation Extender MQ データ・バインディングを使用するようにメッセージを構成する必要があります。

### 始める前に

ネイティブ MQ インポートには WebSphere Transformation Extender MQ データ・バインディングを使用します。

WebSphere Transformation Extender データ・バインディング記述子は、ネイティブ MQ インポートの応答メッセージでのみ設定できます。

### このタスクについて

メッセージを個別に構成する場合は、ネイティブ MQ インポートの応答メッセージにデータ・バインディング記述子を設定します。

メッセージを個別に構成する必要がない場合は、バインディング自体の中でカスタム・プロパティを使用して、WebSphere Transformation Extender との関連付けを作成しておくことをお勧めします。

### 手順

1. 要求メッセージをインスタンス・レベルで構成することはできません。バインディングのカスタム・プロパティを使用して、使用する特定の WebSphere Transformation Extender マップとの関連付けを作成する必要があります。
2. 応答メッセージを構成します。着信メッセージの MQRFH2 ヘッダーに DataBindingDescriptor を設定します。

### タスクの結果

これらのプロパティは、MQ クライアントから WebSphere Process Server にメッセージが送信される時点で設定されます。

### ネイティブ MQ エクスポートのバインディング記述子の設定

WebSphere Transformation Extender と特定の方法で対話するように各メッセージを構成するには、WebSphere Transformation Extender データ・バインディング記述子を介して WebSphere Transformation Extender MQ データ・バインディングを使用するようにメッセージを構成する必要があります。

### 始める前に

ネイティブ MQ エクスポートには WebSphere Transformation Extender MQ データ・バインディングを使用します。

WebSphere Transformation Extender データ・バインディング記述子は、ネイティブ MQ エクスポートの要求メッセージでのみ設定できます。

### このタスクについて

メッセージを個別に構成する場合は、ネイティブ MQ エクスポートの要求メッセージにデータ・バインディング記述子を設定します。

メッセージを個別に構成する必要がない場合は、バインディング自体の中でカスタム・プロパティを使用して、WebSphere Transformation Extender との関連付けを作成しておくことをお勧めします。

#### 手順

1. 要求メッセージを構成します。着信メッセージの MQRFH2 ヘッダーに DataBindingDescriptor を設定します。
2. 要求メッセージをインスタンス・レベルで構成することはできません。バインディングのカスタム・プロパティを使用して、使用する特定の WebSphere Transformation Extender マップとの関連付けを作成する必要があります。

#### タスクの結果

これらのプロパティは、MQ クライアントから WebSphere Process Server にメッセージが送信される時点で設定されます。

#### HTTP インポートのバインディング記述子の設定

インポートに対する同一操作に送られる各 HTTP メッセージを、異なる WebSphere Transformation Extender マップを使用するように構成するには、WebSphere Transformation Extender データ・バインディング記述子を使用するようにメッセージを構成する必要があります。

#### 始める前に

HTTP インポートには WebSphere Transformation Extender HTTP データ・バインディングを使用します。

WebSphere Transformation Extender データ・バインディング記述子は、HTTP インポートの応答メッセージでのみ設定できます。

#### このタスクについて

メッセージを個別に構成する場合は、HTTP インポートの応答メッセージにデータ・バインディング記述子を設定します。

メッセージを個別に構成する必要がない場合は、バインディング自体の中でカスタム・プロパティを使用して、WebSphere Transformation Extender との関連付けを作成しておくことをお勧めします。

#### 手順

1. 要求メッセージをインスタンス・レベルで構成することはできません。バインディングのカスタム・プロパティを使用して、使用する特定の WebSphere Transformation Extender マップとの関連付けを作成する必要があります。
2. 応答メッセージを構成します。

DataBindingDescriptor というカスタム HTTP ヘッダーを HTTP インポート内に作成します。以下に例を挙げます。

```
databinding://WTX/map?businessObject=Customer&
contentType =EDI
```

## HTTP エクスポートのバインディング記述子の設定

エクスポートの同じ操作に対して公開された各 HTTP メッセージを構成し、異なる WebSphere Transformation Extender マップを使用するには、WebSphere Transformation Extender データ・バインディング記述子を使用するようにメッセージを構成する必要があります。

### 始める前に

HTTP エクスポートには WebSphere Transformation Extender データ・バインディングを使用します。

WebSphere Transformation Extender データ・バインディング記述子は、HTTP エクスポートの要求メッセージでのみ設定できます。

要求を HTTP エクスポートに送信する HTTP クライアントにより、要求メッセージ・ヘッダーまたは URL にデータ・バインディング記述子が設定されます。

### このタスクについて

メッセージを個別に構成する場合は、HTTP エクスポートの要求メッセージにデータ・バインディング記述子を設定します。

メッセージを個別に構成する必要がない場合は、バインディング自体の中でカスタム・プロパティを使用して、WebSphere Transformation Extender との関連付けを作成しておくことをお勧めします。

### 手順

1. 要求メッセージを構成します。 要求メッセージでデータ・バインディング記述子を設定するには、以下の 2 つの方法があります。

オプション	例
<b>DataBindingDescriptor</b> というカスタム HTTP ヘッダーを HTTP エクスポート内に作成します。	<code>databinding://WTX/ map?businessObject=Customer&amp; contentType=EDI</code>
URL に照会パラメーターを追加します。	<code>http://www.ibm.com/Export1/ map?businessObject=Customer &amp;contentType=EDI</code>

一般に、カスタム・ヘッダーを作成する方法は実用的ではありません。URL に照会パラメーターを設定する方法をお勧めします。

2. 応答メッセージをインスタンス・レベルで構成することはできません。バインディングのカスタム・プロパティを使用して、使用する特定の WebSphere Transformation Extender マップとの関連付けを作成する必要があります。



---

## 第 10 章 サーバーによってサポートされるアダプター

WebSphere Process Server では、WebSphere Adapters と WebSphere Business Integration Adapters の 2 種類のアダプターがサポートされています。ビジネス・アプリケーションは、アダプターを利用することにより、さまざまなエンタープライズ情報システム (EIS) (データベース、エンタープライズ・リソース・プランニング・システム、ファイル・システム、電子メール・システムなど) へ接続して、サービスとして機能します。

アプリケーションと EIS は、アダプターを利用することにより、「相互に対話」できます。つまり、一貫した方法で情報を送受信できます。アプリケーションをサービスとして機能させるために、アダプターはアプリケーションを WebSphere Process Server に接続します。これにより、Service Oriented Architecture (SOA) が実現します。アダプターを使用すれば、各 EIS またはアプリケーション・サーバー用に専用の接続ユーティリティを提供する (またはカスタム接続ユーティリティを作成する) 必要はなくなります。

### 関連概念

244 ページの『EIS バインディング』

エンタープライズ情報システム (EIS) のバインディングは、SCA コンポーネントと外部 EIS 間の接続を提供します。この通信を可能にするために使用する手段は、JCA 1.5 リソース・アダプターおよび Websphere Adapters をサポートする EIS エクスポートおよび EIS インポートです。

---

## WebSphere Adapter および WebSphere Business Integration Adapter 間の差異

WebSphere Adapter および WebSphere Business Integration Adapter は、いずれもコンポーネントとエンタープライズ情報システムの間の通信を仲介します。これら 2 種類のアダプターは、統合、JCA への準拠、データ・モデル、接続の管理などの点で異なっています。

WebSphere Adapter および WebSphere Business Integration Adapter 間には、いくつかの差異があります。これらの違いは、アプリケーションの開発時に最も重要です。実行中のサーバーにアプリケーションをデプロイする場合、使用するアダプターの性質によって、実行する必要があるステップは異なってきます。

アダプターは、Enterprise Information System (EIS) と WebSphere アプリケーション間の通信メカニズムを提供します。アダプターの操作を説明するために、334 ページの図 27 と 335 ページの図 28 では、2 つのタイプのアダプターについてサーバーと EIS の間の通信の詳細を示しています。

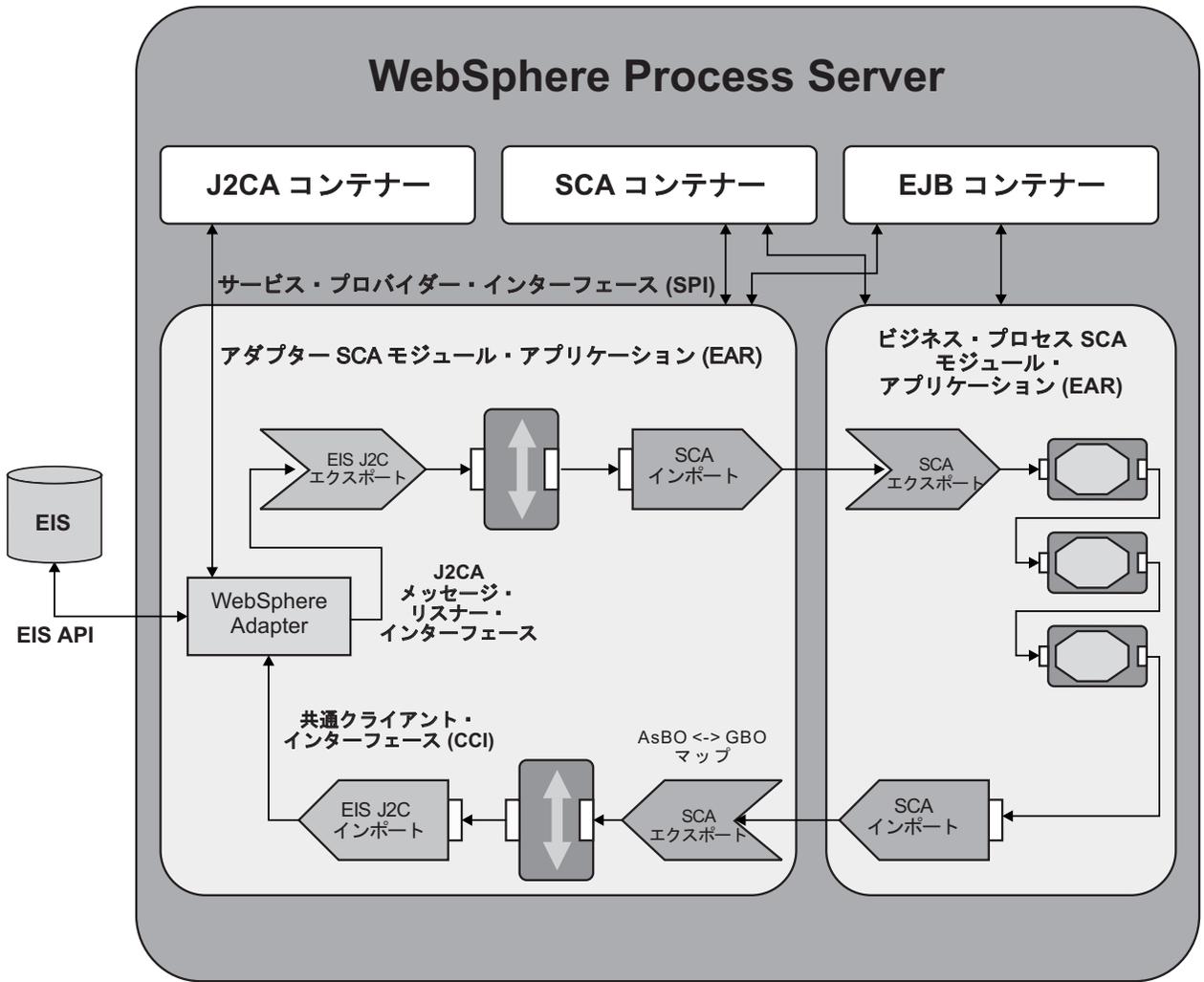


図 27. WebSphere Adapter の詳細な図式

図 27 は、サーバーでサポートされる J2EE コンポーネントと EIS 間の接続を管理する WebSphere Adapter を示しています。WebSphere Adapter はサーバー内にあります。

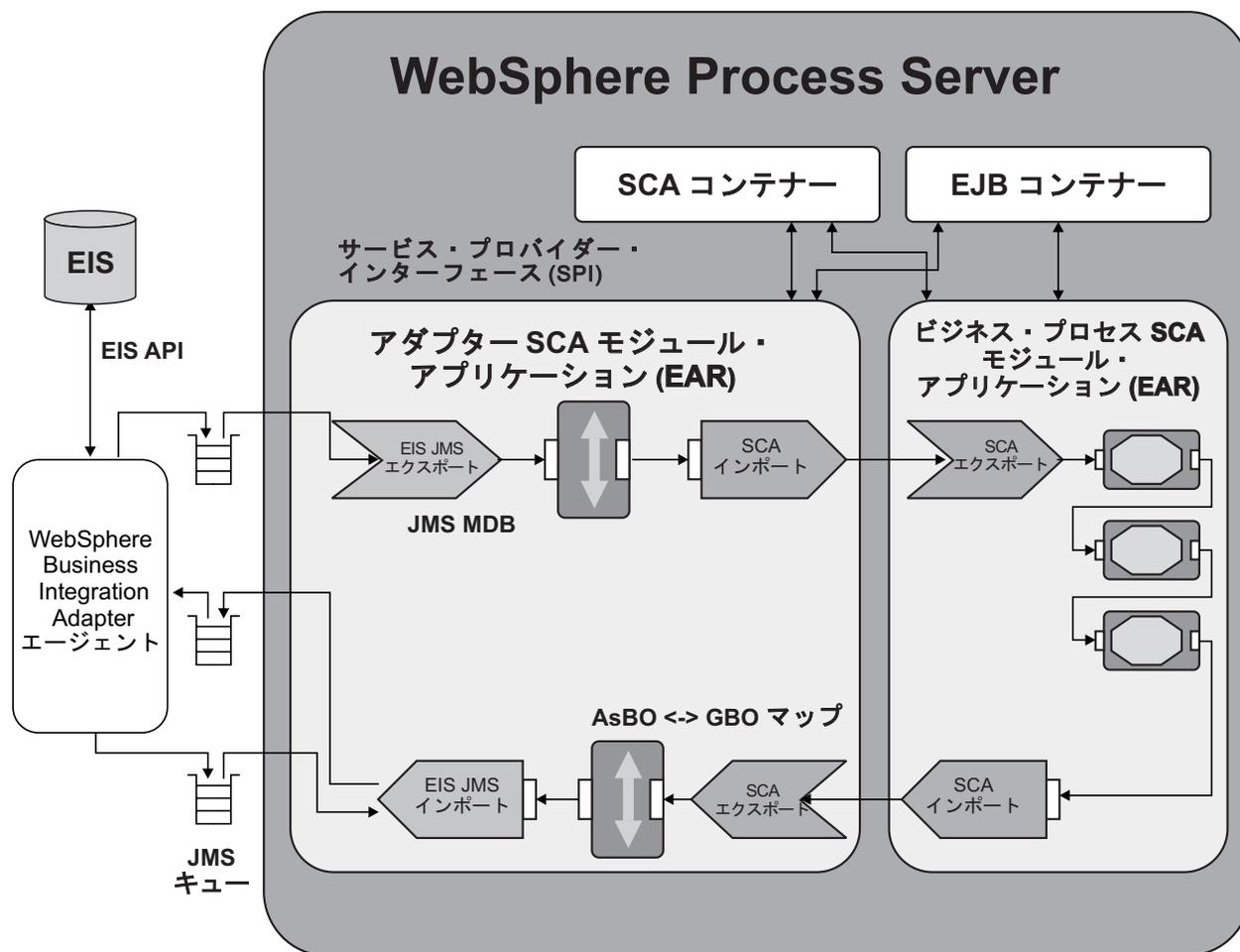


図 28. WebSphere Business Integration Adapter の詳細な図式

図 28 は、WebSphere Integration Broker と EIS 間の通信を仲介する WebSphere Business Integration Adapter を示しています。統合ブローカーは、Java Message Service (JMS) トランスポート・レイヤーを使用することにより、WebSphere Business Integration Adapter と通信します。

表 31 に、2 つのアダプター・タイプの相違点を示します。

表 31. WebSphere Adapter および WebSphere Business Integration Adapter 間の差異

フィーチャー	WebSphere Adapter	WebSphere Business Integration Adapter
JCA Compliance	JCA 完全準拠 (バージョン 1.5)	JCA 非準拠
Connectivity Manager	標準 JCA 規約を利用し、開始および停止などのライフ・サイクル・タスクを管理します。	WebSphere Adapter Framework を利用し、接続を管理します。
Event Notification	EventStore サブクラスを使用して、EIS からイベントを取り出します。	pollFor Events メソッドを使用してイベント通知を管理します。

表 31. WebSphere Adapter および WebSphere Business Integration Adapter 間の差異 (続き)

フィーチャー	WebSphere Adapter	WebSphere Business Integration Adapter
Request Processing	クライアントは、いくつかの対話規約の 1 つを直接起動し、EIS のデータを照会または変更します。	統合サーバーおよび WebSphere Adapter Framework を利用して、処理要求を開始および支援します。
Data Models	Enterprise Metadata Discovery (EMD) ユーティリティを使用し、EIS を解析し、Service Data Object (SDO) およびその他有用な成果物を開発します。EMD は、WebSphere Adapter の実装の一部です。	別の Object Discovery Agent (ODA) を使用して、EIS をイントロスペクトし、ビジネス・オブジェクト定義スキーマを生成します。
Integration	サーバーを実行します。	サーバー外部にあります。サーバーまたは統合ブローカーは、Java Message Service (JMS) トランスポート・レイヤーを介してアダプターと通信します。

WebSphere Adapters は推奨製品です。

## WebSphere Adapters

WebSphere Adapters (リソース・アダプターともいう) は、エンタープライズ情報システム (EIS) と、サーバーでサポートされている J2EE コンポーネントの間で、管理された両方向接続を実現します。

WebSphere Adapters は、WebSphere Business Integration Adapters よりも優先されるため、この情報ライブラリーでは WebSphere Adapters について説明します。

### 詳細情報の参照先

WebSphere Adapters の構成方法と使用方法については、WebSphere Integration Developer インフォメーション・センターで、『アダプターの構成および使用』を参照してください。ご使用のアダプターのアダプター・ガイドで、ナビゲーションを展開し「**アダプター・モジュールの管理 (Administering the adapter module)**」をクリックしてください。

アダプターの一般情報については、WebSphere Integration Developer インフォメーション・センターの『アダプターを使用した外部サービスへのアクセス』を参照してください。

WebSphere Adapters で SCA コンポーネントと EIS 間の接続性を提供するために使用される EIS バインディングについては、244 ページの『EIS バインディング』を参照してください。

---

## WebSphere Business Integration Adapter

WebSphere Business Integration Adapter は、一連のソフトウェア、アプリケーション・プログラミング・インターフェース (API)、およびツールで構成されています。これによって、アプリケーションは、統合ブローカー経由でビジネス・データを交換できます。

それぞれのビジネス・アプリケーションがビジネス・インテグレーション・プロセスに関与するためには、そのアプリケーション固有のアダプターが必要です。現在の WebSphere Business Integration Adapter Framework および Development Kit System Manager ツールを使用して、アダプターをインストール、構成、およびテストできます。WebSphere Integration Developer を使用して既存のビジネス・オブジェクトおよびコネクタ構成ファイルをインポートすることにより、成果物を生成し、WebSphere Process Server のためのソリューションをアSEMBルできます。

WebSphere Business Integration Adapter 用の操作コマンドは、管理コンソールの一部です。

### 詳細情報の参照先

これらのアダプターを使用した作業について詳しくは、『WebSphere Business Integration Adapters の使用』を参照してください。

## WebSphere Business Integration Adapter の管理

実行中の WebSphere Business Integration Adapter を管理コンソールから管理できます。

### 始める前に

管理するためには WebSphere Business Integration Adapter が稼働している必要があります。

### このタスクについて

リソースを管理し、リソースに対してさまざまな管理アクションを実行するには、以下の手順に従います。

#### 手順

1. 管理するリソース (1 つまたは複数) を選択します。管理コンソールの最上位から、以下の手順を実行します。
  - a. 「サーバー」を展開します。
  - b. 「アプリケーション・サーバー」を選択します。
  - c. サーバーのリストから、管理したいリソースが存在するサーバーを選択します。

対象となるリソースをホストするサーバーの名前をクリックします。

- d. 「**WebSphere Business Integration Adapter サービス**」を選択します。

「構成」タブの副見出し「ビジネス・インテグレーション」の下で「**WebSphere Business Integration Adapter サービス**」を選択します。

- e. 「WebSphere Business Integration Adapter リソースの管理」を選択します。
- f. リソースのリストから、管理したいリソースを選択します。

管理するリソースに関連したチェック・ボックスを選択します。

2. 選択したリソースを管理します。

選択したリソースに適用するコマンド・ボタンの 1 つをクリックします。

コマンド	説明
非アクティブ化	選択したリソースの状況をアクティブから休止または非アクティブに変更します。
アクティブ化	選択したリソースの状況を非アクティブからアクティブに変更します。
中断	選択したリソースの状況をアクティブから休止に変更します。
再開	選択したリソースの状況を休止からアクティブに変更します。
シャットダウン	選択したリソースの状況をアクティブから使用不可に変更します。

---

## 第 11 章 イベントの使用

イベントは、1 つのコンポーネントから別のコンポーネントへ送信される要求または応答です。イベントは特定の順序で処理できます。また、失敗イベントを表示、破棄、変更、または再サブミットできます。

---

### イベントの順次処理

イベント順序付けにより、WebSphere Process Server コンポーネントが非同期呼び出しからのイベントを配信順に処理することが保証されます。イベントの順序は、ビジネス・インテグレーション・シナリオ全体で維持されます。

「イベント」は、1 つのコンポーネントから別のコンポーネントへ送信される要求または応答です。イベントは、データおよび呼び出しメタデータ (ターゲット・コンポーネントの名前、操作、パラメーターなど) をカプセル化します。

注: イベントの順序付けは、非同期呼び出しにより送信される要求でのみサポートされます。

#### イベント順序付けを使用する理由

実装によっては、ソース・アプリケーションによって送信された順序と同じ順序でターゲット・コンポーネントがイベントを処理する必要がある場合があります。誤った順序で処理すると、エラーや例外が発生する可能性があります。例えば、ソース・アプリケーションが、ビジネス・オブジェクトを作成するイベントを生成した後、そのビジネス・オブジェクトを更新するイベントを生成した場合は、作成イベントを最初に処理する必要があります。

非同期呼び出しでは、イベントはサービス統合バスの宛先に格納され、メッセージ駆動型 Bean (MDB) の複数のインスタンスによって処理できます。その結果、正しい順序で処理されず、失敗してしまう可能性があります。この問題を避けるために、イベント順序付けを使用します。

#### イベント順序付けの仕組み

イベント順序付けを有効にするには、WebSphere Integration Developer で使用可能な `イベント順序付け修飾子` を使用します。修飾子は、イベント順序付けが必要な各メソッドで設定する必要があります。修飾子は、それらのメソッドに対する呼び出しを順序付ける必要があることをランタイム環境に指示します。

各修飾子には、イベントの順序付けの方法を決定するイベント順序付けキーが含まれます。キーの値は、呼び出しに関連したビジネス・オブジェクトの 1 つ以上の属性から構成されます。同じキーを共有するすべてのイベントがグループ化され、順次処理されます。イベント順序付けキーを持たないイベントは、引き続き通常どおりに (順序付けされたイベントと並列に) 処理されます。

順序付けされたイベントは、ロックを獲得してから、ターゲット・コンポーネントに送信されて処理されます。そのイベントに対するビジネス・ロジックの実行が完

了すると、ロックはただちに解放され、同じイベント順序付けキーを持つ次のイベントに割り当てられます。イベントが必要なロックを獲得できない場合は、ロックを獲得するまで呼び出しの実行が中断されます。

### 関連概念

『例: イベント順序付け』

イベント順序付けの仕組みを理解するために、ソース・アプリケーション (コンポーネント A) がターゲット・アプリケーション (コンポーネント B) を非同期的に呼び出して新規オーダーを作成した後、変更されたデータでそれらのオーダーを更新する場合について考えます。

342 ページの『イベント順序付けを実装する場合の考慮事項』

ビジネス・インテグレーション環境でイベント順序付けの計画、実装、およびトラブルシューティングを行うには、ここに含まれるトピックに記載されている情報を使用してください。

## 例: イベント順序付け

イベント順序付けの仕組みを理解するために、ソース・アプリケーション (コンポーネント A) がターゲット・アプリケーション (コンポーネント B) を非同期的に呼び出して新規オーダーを作成した後、変更されたデータでそれらのオーダーを更新する場合について考えます。

コンポーネント A はコンポーネント B を検索し、`create` メソッドを呼び出してオーダーを作成し、その際に `Order` ビジネス・オブジェクトを使用します。`Order` ビジネス・オブジェクトは以下の属性を持ちます。

属性	タイプ
ID	string
customer	string
productName	string
quantity	integer

コンポーネント A は次に `update` メソッドを呼び出し、新規に作成したオーダーのデータを更新します。

この例では、コンポーネント A からコンポーネント B に、5 つの別個のイベントが以下に示す順序で送信されたと想定します。

- Create1: この呼び出しは `create` メソッドを呼び出し、ID が 1 で quantity が 10 の `Order` ビジネス・オブジェクトを渡します。
- Create2: この呼び出しは `create` メソッドを呼び出し、ID が 2 で quantity が 8 の `Order` ビジネス・オブジェクトを渡します。
- Update1: この呼び出しは `update` メソッドを呼び出し、ID が 1 で quantity が 15 の `Order` ビジネス・オブジェクトを渡します。
- Update2 : 3 番目の呼び出しは `update` メソッドを呼び出し、ID が 1 で quantity が 12 の `Order` ビジネス・オブジェクトを渡します。
- Update3: この呼び出しは `update` メソッドを呼び出し、ID が 2 で quantity が 10 の `Order` ビジネス・オブジェクトを渡します。

イベントごとに、メッセージは、呼び出しと同じ順序でサービス統合バス宛先に格納されます。メッセージ駆動型 Bean (MDB) がメッセージを読み取り、ターゲット・コンポーネント (この場合はコンポーネント B) に送信して処理させます。モジュールごとの MDB の数は 1 つのみですが、その MDB のインスタンスは複数存在し、これら 5 つのメッセージは並列に処理されます。Update2 のメッセージを処理している MDB スレッドが、Create1 イベントのメッセージを処理しているスレッドより先に完了する可能性があります。その場合にはオーダーがまだ作成されていないため、Update2 イベントは失敗します。

このようなエラーを防ぐために、この例ではイベント順序付けを実装しています。下記のコンポーネント定義の例では、create メソッドと update メソッドの両方にイベント順序付け修飾子を指定しています。これらのメソッドはいずれも同じイベント順序付けキー (Order ビジネス・オブジェクトの ID 属性に設定されます) を使用し、同じイベント順序付けグループに配置されます。3 番目のメソッド (retrieve) は順序付けしていません。

```
<interfaces>
  <interface xsi:type="wsdl:WSDLPortType" portType="ns1:ProcessOrder">
    <method name="create">
      <scdl:interfaceQualifier xsi:type="es:EventSequencingQualifier">
        <es:eventSequencing sequencingGroup="default" continueOnError="true">
          <keySpecification>
            <parameter name="Order">
              <xpath>ID</xpath>
            </parameter>
          </keySpecification>
        </es:eventSequencing>
      </scdl:interfaceQualifier>
    </method>
    <method name="update"/>
      <scdl:interfaceQualifier xsi:type="es:EventSequencingQualifier">
        <es:eventSequencing sequencingGroup="default" continueOnError="true">
          <keySpecification>
            <parameter name="Order">
              <xpath>ID</xpath>
            </parameter>
          </keySpecification>
        </es:eventSequencing>
      </scdl:interfaceQualifier>
    <method name="retrieve"/>
  </interface>
</interfaces>
```

イベント順序付けを有効にすると、この例の 5 つのイベントは以下のように処理されます。

1. コンポーネント A が Create1 要求を送信します。これは宛先に置かれ、MDB のインスタンスによって処理されます。
2. Create1 イベントがロックを獲得し、コンポーネント B に送信して処理させます。
3. コンポーネント A が Update1 要求を送信します。これは宛先に置かれ、MDB のインスタンスによって処理されます。
4. Update1 イベントがロックの獲得を試みます。Create1 イベント (Update1 と同じイベント順序付けキー値を共用) がまだロックを保持している場合は、Create1 に対するロックが解放されるまで、このイベントの処理は中断されます。
5. コンポーネント A が Create2 要求を送信します。これは宛先に置かれ、MDB のインスタンスによって処理されます。

6. Create2 要求 (イベント順序付けキーの値が異なる) がロックを獲得し、コンポーネント B に送信されて処理されます。
7. コンポーネント A が Update2 要求を送信します。これは宛先に置かれ、MDB のインスタンスによって処理されます。
8. Update2 イベントがロックの獲得を試みます。Create1 イベントまたは Update1 イベントのいずれか (Update2 と同じイベント順序付けキー値を共用) がまだロックを保持している場合は、このイベントの処理は中断されます。Update1 イベントがロックを獲得して処理され、ロックが解放されるまで処理は行われません。
9. コンポーネント A が Update3 要求を送信します。Create2 イベント (Update3 と同じイベント順序付けキー値を共用) がまだロックを保持している場合は、Create2 に対するロックが解放されるまで、このイベントの処理は中断されません。

#### 関連概念

339 ページの『イベントの順次処理』

イベント順序付けにより、WebSphere Process Server コンポーネントが非同期呼び出しからのイベントを配信順に処理することが保証されます。イベントの順序は、ビジネス・インテグレーション・シナリオ全体で維持されます。

## イベント順序付けを実装する場合の考慮事項

ビジネス・インテグレーション環境でイベント順序付けの計画、実装、およびトラブルシューティングを行うには、ここに含まれるトピックに記載されている情報を使用してください。

#### 関連概念

339 ページの『イベントの順次処理』

イベント順序付けにより、WebSphere Process Server コンポーネントが非同期呼び出しからのイベントを配信順に処理することが保証されます。イベントの順序は、ビジネス・インテグレーション・シナリオ全体で維持されます。

## サポートされるコンポーネントと呼び出し

イベント順序付けを実装する前に、使用する呼び出しおよびコンポーネントのタイプと、それらが順序付けをサポートするかどうかを考慮してください。

イベント順序付けは、以下の要件を満たす Service Component Architecture (SCA) コンポーネントからのすべての要求でサポートされます。

- コンポーネントが WSDL (Web サービス記述言語) インターフェースを使用する必要があります。
- コンポーネントが非同期呼び出しを使用する必要があります。

**注:** イベントが SCA 宛先に格納される前は、クライアントがイベントの順序を管理する必要があります。順序付けが必要な場合は、クライアントが単一スレッド内で SCA 呼び出しを実行する必要があります。

応答ではサポートされません。

同期が実装されているコンポーネントに対する同期呼び出し中に暗黙のうちに順序付けされるイベントに対してイベント順序付けを使用する必要はありません。クラ

イベントが呼び出しに単一スレッドを使用する場合、呼び出しは、ターゲットがイベントの処理を完了するまで自動的に待機します。イベントが返されるまで、他の呼び出しは実行できません。

## コンポーネントのイベント順序付け宣言

コンポーネント上のどのメソッドがイベント順序付けを使用する必要があるかを判別したら、WebSphere Integration Developer を使用して、それらの各メソッドにイベント順序付け修飾子を含めるよう、コンポーネント定義を更新します。

**重要:** コンポーネントでイベントの順序付けを宣言する場合は、コンポーネントが管理対象スレッドで呼び出されることを確認します。管理対象スレッドは、イベントを適切に順序付けるために必要なセッション情報を提供します。

イベント順序付け修飾子は Service Component Definition Language (SCDL) で定義されているタイプを拡張し、Service Component Architecture (SCA) コンポーネントに対するサービス品質を向上させます。

イベント順序付け修飾子には `keySpecification` エレメントが含まれ、このエレメントによって順序付けするイベントを識別します。イベント順序付けを使用するメソッドごとに 1 つの `keySpecification` エレメントが必要です。各 `keySpecification` では `parameter` エレメントを使用します。このエレメントは、イベント順序付けキーの値を提供するビジネス・オブジェクト属性 (複数可) を示します。

順序付け機能をさらに拡張するには、イベント順序付け修飾子の属性を使用します。例えば `sequencingGroup` 属性は、まとめて順序付けする必要があるメソッドをグループ化します。同じグループのメソッドによって生成されるイベントは、すべて順次処理されます。`continueOnError` 属性は、順序付けされたイベントが未宣言の実行時例外で失敗した後にイベント処理を続行するかどうかを決定します。`false` に設定すると、失敗イベントが解決されるまで、これ以降のすべてのイベントの処理が中断されます。

下記のコンポーネント定義の例では、`create` メソッドおよび `update` メソッドでイベント順序付けを定義しています。両方のメソッドの `keySpecification` エレメントが、パラメーターを `newOrder` ビジネス・オブジェクトの ID 属性として定義します。

```
<interfaces>
  <interface xsi:type="wsdl:WSDLPortType" portType="ns1:ProcessOrder">
    <method name="create">
      <scdl:interfaceQualifier xsi:type="es:EventSequencingQualifier">
        <es:eventSequencing sequencingGroup="default" continueOnError="true">
          <keySpecification>
            <parameter name="newOrder">
              <xpath>ID</xpath>
            </parameter>
          </keySpecification>
        </es:eventSequencing>
      </scdl:interfaceQualifier>
    </method>
    <method name="update"/>
      <scdl:interfaceQualifier xsi:type="es:EventSequencingQualifier">
        <es:eventSequencing sequencingGroup="default" continueOnError="true">
          <keySpecification>
            <parameter name="newOrder">
              <xpath>ID</xpath>
            </parameter>
          </keySpecification>
        </es:eventSequencing>
      </scdl:interfaceQualifier>
    </method>
  </interface>
</interfaces>
```

```
        </es:eventSequencing>
        </scdl:interfaceQualifier>
    </interface>
</interfaces>
```

コンポーネントでイベントの順序付けを宣言する場合は、コンポーネントが管理対象スレッドで呼び出されることを確認します。管理対象スレッドは、イベントを適切に順序付けるために必要なセッション情報を提供します。

## エクスポート・バインディングによるイベント順序付け

イベント順序付けは、EIS、JMS、WebSphere MQ、および WebSphere MQ JMS エクスポート・バインディングでサポートされます。エクスポートに正しい順序でメッセージを処理および配信させるには、エクスポート・バインディングを適切に構成する必要があります。

エクスポート・バインディングを処理するターゲット・コンポーネントでイベント順序付けを使用する場合は、以下の要件を考慮に入れてください。

- ターゲット・コンポーネントでイベント順序付けを使用する場合は、アダプター・コンポーネントが Java Message Service (JMS) エクスポートの最適化されていないパスを使用する必要があります。
- JMS エクスポート・バインディングのイベント順序付けを有効にするには、着信メッセージを処理する並行メッセージ駆動型 Bean (MDB) の数を制限する必要があります。このためには、ActivationSpec の maxConcurrency カスタム・プロパティの値を 1 に設定します。
- WebSphere MQ JMS エクスポートのイベント順序付けを有効にするには、メッセージ駆動型 Bean にメッセージを配信する並行リスナー・スレッドの数を制限する必要があります。このためには、maxSessions プロパティの値を 1 に設定します。
- ネイティブ MQ エクスポートのイベント順序付けを有効にするには、WebSphere Integration Developer を使用して eventSequencing プロパティを設定する必要があります。

### 関連情報

 [EIS エクスポート・バインディングでのイベントの順序付けの有効化](#)

 [JMS エクスポート・バインディングでのイベントの順序付けの有効化](#)

 [WebSphere MQ JMS エクスポート・バインディングでのイベントの順序付けの有効化](#)

## Network Deployment 環境でのイベント順序付け

イベント順序付けは、HA マネージャーの有無にかかわらず Network Deployment 環境で使用できます。このトピックの表を参照して、使用するトポロジーがサポートされることを確認してください。

イベント順序付けを使用するコンポーネントの Service Component Architecture (SCA) 宛先は分割できません。したがって、クラスターを使用している場合は、クラスターごとに 1 つだけアクティブ・メッセージング・エンジンを持つことができます。

表 32. Network Deployment 環境でのイベント順序付けのサポート

トポロジ	イベント順序付けのサポートの有無
スタンドアロン・サーバー	あり
クラスターなし	あり
アプリケーションをクラスター化している。メッセージング・エンジンと宛先はクラスター化していない。	あり
メッセージング・エンジンをクラスター化している。アプリケーションと宛先はクラスター化していない。	あり
メッセージング・エンジンと宛先をクラスター化している。アプリケーションはクラスター化していない。	なし。クラスター化された宛先は分割されており、使用の際にイベント順序付けを行うことはできません。
アプリケーションとメッセージング・エンジンをクラスター化している (同じクラスター)。宛先はクラスター化していない。	あり
アプリケーション、メッセージング・エンジン、および宛先をクラスター化している (同じクラスター)。	なし。クラスター化された宛先は分割されており、使用の際にイベント順序付けを行うことはできません。
アプリケーションとメッセージング・エンジンをクラスター化している (別のクラスター)。宛先はクラスター化していない。	あり
アプリケーション、メッセージング・エンジン、および宛先をクラスター化している (別のクラスター)。	なし。クラスター化された宛先は分割されており、使用の際にイベント順序付けを行うことはできません。

## 高可用性環境でのイベント順序付けの使用

高可用性 (HA) サポートとは、イベント順序付けランタイムなどのシステム・サブコンポーネントの可用性が高められており、ノードまたはデーモンの障害発生時にワークロードの分散が可能であることを意味します。

イベント順序付けを行う場合は、宛先でシングルトン・サービスがイベント・メッセージを処理する必要がありますが、この処理を Single Point of Failure としないために必要なサービスを HA マネージャーが提供します。その代わりに、システム異常終了の場合には、イベント順序付けランタイムはクラスター内の別のサーバーにフェイルオーバーします。

## 失敗した順序付けされたイベント

処理でエラーが発生したり、リソースを使用できない場合、順序付けされたイベントが失敗する可能性があります。残りのイベントがどのように処理されるかは、イベント順序付け修飾子の `continueOnError` 属性によって決まります。

`continueOnError` 属性の有効な値は、以下の 2 つです。

**True** イベントが失敗したかどうかにかかわらず一連のイベントの処理を継続する場合は、この値を使用します。

**False** 失敗が解決されるまで従属イベントの処理を中断する場合は、この値を使用

します。Failed Event Manager を使用すると、失敗した順序付けされたイベントを迅速に識別し、処理を再サブミットすることができます。

continueOnError 属性が False に設定されている状態で、順序付けされたイベントの処理が失敗して Failed Event Manager に送信された場合は、以下のいずれかの方法でそのイベントを処理することができます。

- 変更を行わずに再サブミットする
- 変更を行ってから再サブミットする (イベント順序付けキー ID の変更は不問)
- 削除する (Recovery サブシステムは、イベント順序付けコールバックを使用して削除済みイベントに関連付けられているロックを削除し、残りのイベントを処理します)

再サブミットが成功すると、キュー内の順序付けされた元の位置でイベントが処理されます。

## イベント順序付けの制限

特定のタイプのコンポーネントおよび呼び出しでは、イベント順序付けのサポートが制限されます。

現行リリースのイベント順序付けには、以下の制限があります。

- 非開始 receive を行う BPEL (Business Process Execution Language) プロセスにバインドされた操作で、イベント順序付けを使用することはお勧めしません。存続期間の長いビジネス・プロセスでは、イベント順序付けは作業完了規約に基づいてロックの解放時期を判別します。この作業完了規約は、新規プロセス・インスタンスが作成されるときに必ずアクティブ化されます。しかし、非開始 receive を行う場合は新規プロセス・インスタンスが作成されません。結果として、イベント順序付けランタイムが、完了済みの作業規約を正確に検出することが困難となり、ロックの解放が早すぎたり遅すぎたりする可能性があります。
- 非開始 receive を行うビジネス・ステート・マシンにバインドされた操作で、イベント順序付けを使用することはお勧めしません。
- 作業完了規約は、BPEL コンポーネントの場合にのみサポートされます。非同期呼び出しを行う他のタイプのコンポーネントでイベント順序付けを効果的に使用するには、要求応答メソッド・シグニチャーの使用をお勧めします。イベント順序付けランタイムは、応答を作業完了のシグナルとして解釈し、ロックを解放します。

**注:** メソッドを要求応答操作として宣言できない場合は、ダウストリーム・コンポーネントでイベント順序付けを指定する必要があります。その場合は、すべてのメソッドに同じイベント順序付けキーを使用していることを確認します。

以下のシナリオでは、イベントの順序付けはサポートされていません。

- 非管理対象スレッドまたは非 SCA バインディングを使用して、適切なセッション・コンテキストなしで宛先にイベントを送信する場合。
- 非同期呼び出しを行うコンポーネントに対して同期呼び出しを使用する場合。

## WebSphere Process Server におけるイベント順序付けの有効化

イベントの順序付けにより、WebSphere Process Server の SCA コンポーネントに対する着信イベントの順序付け機能が有効になります。エクスポート・バインディングは、ターゲット SCA コンポーネントへのエントリー・ポイントです。イベントの順序付けを有効にするには、エクスポートによってメッセージを受信順に処理して送信する必要があります。

### イベントの順序付けの有効化: EIS エクスポート

WebSphere Adapter は、アクティベーション・スペック・プロパティを指定することにより、WebSphere Process Server でのイベントの順序付けを可能にするメカニズムを提供します。また、エクスポートによってメッセージを受信順に処理して送信する必要があります。

#### このタスクについて

アクティベーション・スペック・プロパティについて詳しくは、WebSphere Adapter の資料を参照してください。JCA 1.5 リソース・アダプターの場合、イベントの順序付けを可能にするアダプターの構成方法の詳細を、特定のプロバイダーの資料で確認してください。

一般に、Network Deployment 環境でイベントの順序付けが必要な場合は、エクスポートが含まれているモジュールを、スタンドアロン・サーバー、または高可用性のために有効にされているアクティブ・サーバーが 1 つのみのクラスターに移動する必要があります。

### イベントの順序付けの有効化: JMS エクスポート

WebSphere Process Server で JMS エクスポート・バインディングのイベント順序付けを有効にするには、バインディングのプロパティを構成する必要があります。また、エクスポートによってメッセージを受信順に処理して送信する必要があります。

#### 始める前に

JMS エクスポート・バインディングでのイベントの順序付けは、宛先が区画に分割されていない場合に限りクラスター環境でサポートされています。クラスターを含む Network Deployment 環境でイベントの順序付けが機能するためには、クラスターあたりのアクティブなメッセージング・エンジンは 1 つのみです。クラスターにないサーバーが含まれている Network Deployment 環境でイベントの順序付けが機能するためには、各サーバーにアクティブなメッセージ・エンジンを含めることができます。

#### このタスクについて

イベントの順序付けを行う場合、イベントをターゲット・コンポーネントにディスパッチして処理する前に、イベントでロックを獲得する必要があります。処理が完了すると、イベントのロックが解放されます。イベントでロックを獲得できない場合は、呼び出し処理が中断されます。その後イベントがロックを獲得すると、イベントがディスパッチされます。

特定のコンポーネントの特定のメソッドでイベントの順序付けが必要であることを宣言するには、コンポーネント定義でそのメソッドにイベントの順序付けの修飾子を追加します。

- `keySpecification` 属性は、順序付けする必要があるイベントの識別に使用するキーを定義します。
- `parameter` 属性は、キー属性の抽出元パラメーターを指定します。
- `name` 属性はパラメーターの名前です。
- `xpath` 属性は、キーの一部となる値の抽出元パラメーターに適用されます。

パラメーターごとに、キーの一部となるパラメーター・エレメントを指定する必要があります。

**esadmin** コマンド行ユーティリティでは、アクティブなロックとキューに入れているロックをリストおよび削除できます。

WebSphere Integration Developer からの JMS エクスポートのイベント順序付けを有効にします。

#### 手順

1. WebSphere Integration Developer で、エクスポートの「プロパティー」タブをクリックします。
2. 「メッセージ構成 (Message configuration)」セクションの「イベントの順序付けが必要 (Event sequence required)」チェック・ボックスを選択します。

#### タスクの結果

これで、バインディングのイベント順序付けが有効になりました。

注: 例外宛先を除去すると、何らかの障害が発生するたびにすべての着信メッセージが停止します。

### イベントの順序付けの有効化: Generic JMS エクスポート

WebSphere Process Server で汎用 JMS エクスポート・バインディングのイベント順序付けを有効にするには、バインディングのプロパティーを構成する必要があります。また、エクスポートによってメッセージを受信順に処理して送信する必要があります。

#### 始める前に

Network Deployment 環境で Generic JMS エクスポート・バインディングのイベント順序付けが必要な場合、エクスポートが格納されているモジュールをスタンドアロン・サーバーに移動するか、または高可用性のために有効に設定されているアクティブ・サーバーが 1 つのみのクラスターに移動する必要があります。

#### このタスクについて

WebSphere Integration Developer からの汎用 JMS エクスポートのイベント順序付けを有効にします。

#### 手順

1. WebSphere Integration Developer で、エクスポートの「プロパティ」タブをクリックします。
2. 「メッセージ構成 (Message configuration)」セクションの「イベントの順序付けが必要 (Event sequence required)」チェック・ボックスを選択します。

### イベントの順序付けの有効化: WebSphere MQ JMS エクスポート

WebSphere Process Server で WebSphere MQ JMS エクスポート・バインディングのイベント順序付けを有効にするには、バインディングのプロパティを構成する必要があります。また、エクスポートによってメッセージを受信順に処理して送信する必要があります。

#### 始める前に

Network Deployment 環境でこれらのエクスポート・バインディングのイベント順序付けが必要な場合、エクスポートが格納されているモジュールをスタンドアロン・サーバーに移動するか、または高可用性のために有効に設定されているアクティブ・サーバーが 1 つのみのクラスターに移動する必要があります。

#### このタスクについて

WebSphere Integration Developer からの WebSphere MQ JMS エクスポート・バインディングのイベント順序付けを有効にします。

#### 手順

1. WebSphere Integration Developer で、エクスポートの「プロパティ」タブをクリックします。
2. 「メッセージ構成 (Message configuration)」セクションの「イベントの順序付けが必要 (Event sequence required)」チェック・ボックスを選択します。

### イベントの順序付けの有効化: WebSphere MQ エクスポート

WebSphere Process Server で WebSphere MQ エクスポート・バインディングのイベント順序付けを有効にするには、バインディングのプロパティを構成する必要があります。また、エクスポートによってメッセージを受信順に処理して送信する必要があります。

#### 始める前に

Network Deployment 環境で WebSphere MQ エクスポート・バインディングのイベント順序付けが必要な場合、エクスポートが格納されているモジュールをスタンドアロン・サーバーに移動するか、または高可用性のために有効に設定されているアクティブ・サーバーが 1 つのみのクラスターに移動する必要があります。

#### このタスクについて

WebSphere Integration Developer からの WebSphere MQ エクスポートのイベント順序付けを有効にします。

#### 手順

1. WebSphere Integration Developer で、エクスポートの「プロパティ」タブをクリックします。

2. 「メッセージ構成 (Message configuration)」セクションの「イベントの順序付けが必要 (Event sequence required)」チェック・ボックスを選択します。

## ロックのリスト、解放、および削除

ロック・マネージャーはイベント順序付けのロックを処理します。esAdmin コマンドを使用すると、ロック・マネージャー内の任意のロックをリスト、削除、またはアンロックできます。

ロック・マネージャーは、イベント・ロックに対する以下の 2 つの操作をサポートします。

- **ロック:** ロック操作はロックの獲得を試み、データベースにロック要求を格納します。ロックが付与されると、ロックを要求した呼び出しの処理が再開します。
- **アンロック:** アンロック操作は現行のロックを解放し、そのロックを次のロック要求に対して付与します。

同じロックに対する要求は、受信順にキューに入れられます。サーバーに障害が発生した場合でもロックを回復できるようにするため、ロックはデフォルトの WebSphere Process Server データベースおよびデータ・ソースに永続性をもって保管されます。

esAdmin コマンドにより、現在ロック・マネージャーでアクティブなロックおよびキューに入っているロックを管理できます。以下のセクションでは、esAdmin の使用方法を詳細に説明します。

**注:** 区画に分割されたデータベースを使用している場合は、デプロイメント・ターゲットごとに 1 回ずつ esAdmin コマンドを実行します。クラスター環境では、このコマンドを任意のクラスター上で実行できますが、デプロイメント・マネージャー上では実行しないでください。

### ロックのリスト

esAdmin コマンドは、ロック・マネージャーのアクティブなロックとキューに入っているロックをすべてリストできるほか、特定のモジュール、コンポーネント、またはメソッドに関連したロックのみをリストすることもできます。

esAdmin では以下のいずれかのメソッドを使用します。

- **listAll:** ロック・マネージャーのアクティブなロックおよびキューに入っているロックをすべてリストします。
- **listLocks:** ロック・マネージャーのアクティブなロックおよびキューに入っているロックのサブセットをリストします。以下の 1 つ以上のパラメーターを指定すると、ロックのリストがフィルターに掛けられて返されます。

- moduleName
- componentName
- methodName

例えば、

```
esAdmin listLocks CustMod CustComp
```

を使用すると、CusMod モジュールの一部である CustComp コンポーネントについて、アクティブなロックとキューに入っているロックのリストが返されます。

このコマンドは、以下のような出力を返します。

表 33. `esAdmin listLocks` コマンドの出力例

ロック ID	シーケンス ID	所有者 ID	モジュール	コンポーネント	メソッド	システム・メッセージ ID
7564504	2	695376	CustMod	CustComp	createCust	A09-427BE_5002
7564504	3	232757	CustMod	CustComp	createCust	ADF-053RT_5004

上記の出力において、シーケンス ID はロック要求がキューに入った順序です。シーケンスで最小の番号が、現在ロックを保持しています。システム・メッセージ ID は、対応するサービス統合バス・メッセージの ID を指定します。この情報を使用すると、ロック要求と宛先のメッセージを相互に関連付けることができます。

## ロックの解放

`esAdmin` コマンドを使用して、ロックを 1 つ解放するには、以下のようになります。

```
esAdmin unlock lockId
```

ここで、`lockId` は `esAdmin listLock` または `esAdmin listAll` コマンドによって戻される固有のロック ID です。

このコマンドは、デッドロックが発生した場合に役立ちます。デッドロック状態のロックを解放し、そのロックをキュー内の次のロック要求に対して付与できます。

## ロックの削除

1 つ以上のロックを削除する必要がある場合は、まず、ロックに関連付けられたモジュールを停止します。次に、以下のよう `esAdmin` コマンドを使用してデータベースからロックを削除します。

```
esAdmin deleteLocks moduleName
```

宛先にイベント・メッセージの処理を再開させるには、モジュールを再始動する必要があります。

`esAdmin deleteLocks` コマンドを使用するときには十分に注意してください。指定したモジュールのすべてのロックがロック・マネージャー・データベースから削除されます。

## イベント順序付けのトラブルシューティング

イベント順序付けで問題が発生した場合は、このトピックに記載されている情報を参照してください。

### イベント順序付け修飾子に関する問題

以下のようにして、コンポーネント定義が正しいことを確認してください。

- メソッドでイベント順序付け修飾子が設定されていますか? インターフェースで修飾子が誤って設定されていると、イベント順序付けの検証に失敗します。
- パラメーター名は有効ですか?
- xpath エlementが有効であり、正しくプリミティブに解決されますか?
- メソッドに eventSequencing Elementが 1 つだけありますか? 各メソッドがサポートする eventSequencing Elementは 1 つのみです。
- メソッドに keySpecification Elementが 1 つだけありますか? 各メソッドがサポートする keySpecification Elementは 1 つのみです。

## デッドロック

ロックを行う操作が呼び出され、さらにその操作が、同じコンポーネントに対し、同じイベント順序付けキーおよびグループを使用して別の操作を呼び出すと、デッドロックが発生します。デッドロックを解決するには、esAdmin コマンドを使用して現行のロックをリストして解放します。

デッドロックを避けるため、イベント順序付けを実装するときには依存関係を十分に考慮してください。循環依存関係を持つ操作は必ず別のイベント順序付けグループに入れてください。

### BPEL プロセスでのデッドロック

BPEL (Business Process Execution Language) プロセスでイベント順序付けを使用する場合にデッドロックが発生することがあります。以下の両方のアクティビティーに対応する操作でイベント順序付け修飾子を設定すると、デッドロックが発生します。

- 複数インスタンス化 receive または pick アクティビティーで createInstance 属性を yes に設定している
- 相関セット指定で initiation 属性を join に設定している

このタイプのデッドロックを解決するには、esAdmin コマンドを使用して現行のロックをリストし、解放します。以後のデッドロックを防ぐため、必ずこれらのタイプの従属操作を別のイベント順序付けグループに入れてください。

### イベント順序付けコールバックでロックを解放できない

Recovery サブシステムで失敗した順序付けされたイベントを削除する際に、イベント順序付けコールバックによってイベントのロックを解放できない場合があります。これは通常、ターゲット・アプリケーションが削除されている場合や、システムの他のコンポーネント (データベースなど) が使用不可になっている場合に発生します。

この場合、Failed Event Manager によってエラー・メッセージが生成されます。esAdmin コマンドを使用して、失敗イベントに関連付けられているロックを手動で削除します。

## パフォーマンスの問題

イベント順序付けコンポーネントで使用されるメッセージング・エンジン・サーバーでメモリーの問題が発生する場合は、`install_root/properties/eventsequencing.properties` ファイルのランタイム・イベント順序付けプロパティを変更します。

`maxActiveMessages` プロパティは、コンポーネント宛先でロックされる現行メッセージの数を定義します。大きなメッセージが多数ロックされていると、パフォーマンスが低下し、メモリーの問題が発生する可能性があります。値 0 (ゼロ) を指定すると、ロックされるメッセージの数が無制限になる点に注意してください。デフォルトでは、`maxActiveMessages` プロパティは 100 に設定されています。値を変更する場合は、以下の式を使用することをお勧めします。

$$\text{average\_number\_of\_ES\_keys} * \text{average\_number\_of\_potential\_queued\_events\_per\_key} + \text{delta}$$

この `delta` は、同時処理可能な同一の順序付けキーを持つ順序付けされたイベント予想数の見積もり精度の標準偏差です。

`workItemRetryCount` プロパティにより、検証作業の試行回数の上限を設定します。検証作業項目は、非同期イベントがアンロックされ、処理を待機している従属イベントが存在するときに生成されます。この場合、ロックの作成と削除は別の作業単位で行われ、次のイベントが処理される前に作業検証タスクによって各作業単位が処理されます。`workItemRetryCount` は、デフォルトで -1 (再試行) に設定されています。

`workItemSleepTime` プロパティにより、作業検証の再試行間隔を指定します。`workItemSleepTime` は、デフォルトで 10 秒に設定されています。この値を低く設定すると、パフォーマンスが低下する可能性があります。

これらのプロパティを変更するには、以下の手順を実行します。

1. テキスト・エディターで `eventsequencing.properties` ファイルを開きます。
2. ご使用の環境に合わせて変更します。
3. ファイルを保存して閉じます。
4. 変更を有効にするため、イベント順序付けコンポーネントの一部であるアプリケーションをすべて停止してから再始動します。

---

## 失敗イベントの管理

WebSphere Process Server Recovery サービスは、失敗イベントに関するデータを収集します。その後、Failed Event Manager を使用して、失敗イベントを表示、変更、再サブミット、または削除できます。

WebSphere Process Server Recovery サービスは、Service Component Architecture (SCA) コンポーネント間の失敗操作、失敗した JMS イベント、および長時間実行されているビジネス・プロセス内の失敗操作を管理します。

### 失敗した SCA イベント

SCA のコンテキストでは、イベントとは、サービス・アプリケーションによって受信された要求または応答です。外部ソース (インバウンドのアプリケーション・ア

ダブターなど) または Web サービスへの外部呼び出しが発生元である可能性もあります。イベントは、操作対象となるビジネス・ロジックへの参照とデータで構成されており、サービス・データ・オブジェクト (ビジネス・オブジェクト) に保管されています。イベントを受信すると、イベントは適切なアプリケーション・ビジネス・ロジックによって処理されます。

単一の実行スレッドは複数のブランチ (またはスレッド) に分岐できます。個々のブランチは、同じセッション・コンテキストによってメインの起動イベントにリンクされます。

これらのいずれかのブランチのビジネス・ロジックが、システム障害、コンポーネント障害、またはコンポーネントが使用不可であったために完全に実行できない場合、イベントは失敗の状態に変わります。複数のブランチに障害が起こった場合、それぞれについて失敗イベントが作成されます。Recovery サービスは、以下のタイプの失敗した SCA イベントを処理します。

- SCA 操作の非同期呼び出し中に発生したイベント障害
- ランタイム例外が原因のイベント障害 (すなわち、ビジネス・ロジックで 사용되는メソッドで宣言されなかった例外すべて)

Recovery サービスは、同期呼び出しで発生した障害を処理しません。

通常、失敗した SCA イベントには、ソースおよび宛先情報が関連付けられています。ソースおよび宛先は、障害点 (呼び出しが失敗したロケーション) に基づくものであり、対話のタイプとは無関係です。以下の例について考慮してください。コンポーネント A は非同期にコンポーネント B を呼び出します。要求メッセージは A から B に送信され、応答 (コールバック) メッセージは B から A に送信されます。

- 最初の要求中に例外が発生した場合、Failed Event Manager にとっては、コンポーネント A がソースでコンポーネント B が宛先になります。
- 応答中に例外が発生した場合、Failed Event Manager にとっては、コンポーネント B がソースでコンポーネント A が宛先になります。

このことは、すべての非同期呼び出しで当てはまります。

Recovery サービスは、失敗した SCA 非同期対話を、SCA システム・バス (SCA.SYSTEM.cell\_name.Bus) 上に作成された失敗イベントの宛先に送信します。失敗イベントのデータは、失敗イベントのデータベース (デフォルトでは WPCRSDB) に格納され、Failed Event Manager インターフェースを通じて管理目的で使用できます。

## 失敗した JMS イベント

Java Message Service (JMS) のコンテキストでは、JMS データ・バインディングまたは関数セレクター操作が失敗した場合、またはダウンストリーム・コンポーネントで発生した例外が JMS エクスポートまたはインポートにスローされた場合に、失敗イベントが生成されます。

JMS データ・バインディングは、データから JMS メッセージへの変換 (またはその逆) を実行します。JMS インポート・バインディングは JMS メッセージを構成

し、それらをターゲット JMS 宛先に送信します。JMS エクスポート・バインディングは、JMS から SCA システムへのインバウンド接続を提供します。

ダウンストリーム・コンポーネントの対話スタイルによっては、JMS エクスポートまたはインポートからの失敗ダウンストリームで `ServiceRuntimeException` が生成される場合があります。JMS の失敗イベントとして、例外が `Failed Event Manager` に送信されます。一般に、バインディングと障害発生のポイントとの間に非同期ホップがある場合、例外は SCA により処理されます。そうでない場合、例外はバインディングにより処理されます。SCA 最適化パスにより、バインディングからダウンストリーム・コンポーネントへの最初の呼び出しは、対話スタイルが非同期として指定される場合でも、同期します。ダウンストリーム・コンポーネントからの実行時例外は、バインディングに逆に伝搬され、JMS 失敗イベントが生成されます。JMS メッセージへの再試行はありません。

インバウンド通信中のイベントの順序は、以下のとおりです。

1. `JMSImport` が要求メッセージを送信する。
2. `JMSExport` が要求メッセージを受信する。
3. `JMSExport` が SCA コンポーネントを呼び出す。
4. SCA コンポーネントが `JMSExport` に応答を返す。
5. `JMSExport` が応答メッセージを送信する。
6. `JMSImport` が応答メッセージを受信する。

データ・バインディング例外が `JMSExport` または `JMSImport` で発生する場合があります。いずれの場合も `ServiceRuntimeException` 例外がスローされ、JMS の失敗イベントが生成されます。

`JMSExport` の例外で考えられる原因は、以下のとおりです。

- メッセージの書式の不備 (最も可能性の高い原因)
- データ・バインディングまたは関数セレクターの設定が間違っている
- (同期呼び出しでの) SCA コンポーネントの例外

`JMSImport` での例外は、要求メッセージがインポートからエクスポートに送信され、応答メッセージがインポートに送信されても、応答メッセージを正常に処理できない場合に発生することがあります。考えられる原因は、以下のとおりです。

- データ・バインディング・エラー
- インポートとエクスポート間のデータ・バインディング構成タイプに互換性がない

Recovery サービスは、JMS メッセージを収集して共通データベースの Recovery テーブルに保管します。さらに、失敗イベントのモジュール名、コンポーネント名、操作名、失敗した時刻、例外の詳細、および JMS プロパティーも収集します。

Recovery ランタイムが JMS の失敗イベントを保存できない場合、メッセージは元のキューにロールバックされます。

WebSphere Integration Developer には、オーサリング時に各バインディングのリカバリーを有効または無効に設定できるリカバリー・バインディング・プロパティーが用意されています。`recoveryMode` プロパティーは、以下のいずれかの値に設定しま

す。

bindingManaged	失敗したメッセージのリカバリーの管理をバインディングに許可します。
unmanaged	失敗したメッセージのトランスポート固有のリカバリーに依存します。

JMS バインディングのリカバリーは、デフォルトで有効になっています。リカバリーを無効にするには、WebSphere Integration Developer で recoveryMode プロパティを unmanaged に設定して、明示的に無効にする必要があります。 recoveryMode プロパティがない場合 (バージョン 6.2 以前のアプリケーションの場合)、リカバリー機能は有効と認識されます。

Failed Event Manager を使用して JMS の失敗イベントを管理することも、カスタム・プログラムを使用することもできます。

## Failed Business Process Choreographer イベント

Business Process Choreographer のコンテキストでは、処理ロジックによって処理されていない場合に、アクティビティの停止またはプロセス・インスタンスの失敗の原因となる例外が発生することがあります。長時間実行している Business Process Execution Language (BPEL) プロセスが失敗し、以下のいずれかの状況が発生した場合に、失敗イベントが生成されます。

- プロセス・インスタンスの状態が失敗または終了になる場合
- アクティビティが停止状態になった場合

Recovery サービスは、Business Process Choreographer の失敗イベントのモジュール名およびコンポーネント名を収集します。失敗イベントのデータは、Business Process Choreographer データベース (BPEDB) に保管されます。

Recovery サービスは、ビジネス・プロセスおよびヒューマン・タスク非同期要求/応答呼び出しの障害を処理しません。

## Business Flow Manager 保留キュー・メッセージ

失敗イベント・マネージャーを使用して、Business Flow Manager の保留キューに格納されているナビゲーション・メッセージを管理することができます。以下の場合には、ナビゲーション・メッセージが保留キューに格納されていることがあります。

- データベースなどのインフラストラクチャーが利用できない場合。
- メッセージが損傷している場合。

長時間実行プロセスの場合、Business Flow Manager は、後続のナビゲーションを起動する要求メッセージを Business Flow Manager 自体に送信することができます。これらのメッセージにより、プロセス関連のアクション (障害ハンドラーの起動など) またはアクティビティ関連のアクション (アクティビティにおけるプロセス・ナビゲーションの続行など) のいずれかが実行されます。ナビゲーション・メッセージには、関連するプロセス・インスタンス ID (piid) が必ず含まれています。

メッセージによってアクティビティに関連したアクションがトリガーされる場合は、アクティビティ・テンプレート ID (atid) とアクティビティ・インスタンス ID (aiid) も含まれます。

Failed Event Manager を使用して Business Flow Manager 保留キュー・メッセージを管理することも、カスタム・プログラムを使用することもできます。

Business Flow Manager 保留キュー・メッセージは、失敗したイベント・マネージャーから直接削除することはできません。関連するプロセス・インスタンスが存在しない場合、保留キュー・メッセージの再生により、メッセージは削除されます。

## 失敗イベントの管理方法

管理者は、Failed Event Manager を使用して、失敗イベントの参照と管理を行います。失敗イベントを管理する一般的なタスクは、以下のとおりです。

- すべての失敗イベントのブラウズ
- 特定の基準で失敗イベントの検索
- 失敗イベントのデータの編集
- 失敗イベントの再サブミット
- 失敗イベントの削除

Failed Event Manager にアクセスするには、「統合アプリケーション」 → 「**Failed Event Manager**」をクリックします。

### 関連概念

#### セッション・モニター

Common Base Event ブラウザーを使用して、Common Event Infrastructure データベースで同じセッション ID 属性を持つすべてのイベントを検索することによって、同じセッションの一部である複数のイベントをモニターすることができます。

#### インフラストラクチャー障害からの回復

長期実行プロセスは、複数のトランザクションにわたっています。Business Flow Manager には、インフラストラクチャー障害が原因でトランザクションが失敗した場合に、それらの障害から自動的に回復するための機能が用意されています。

## リカバリーのセキュリティー考慮事項

WebSphere Process Server アプリケーションおよび環境でセキュリティーを使用可能にしている場合は、ロール・ベースのアクセスとユーザー ID が Recovery サブシステムにどのような影響を与えるかを理解することが重要です。

### Failed Event Manager のロール・ベースのアクセス

Failed Event Manager は、失敗イベントのデータおよびタスクに対して、ロール・ベースのアクセス制御を使用します。管理者ロールおよびオペレーター・ロールのみが、Failed Event Manager 内のタスクの実行を許可されています。管理者またはオペレーターのいずれかとしてログインしたユーザーは、失敗イベントに関連したすべてのデータを表示でき、すべてのタスクを実行できます。

## イベント ID とユーザー権限

失敗イベントは、要求を発信したユーザーについての情報をカプセル化します。失敗イベントを再サブミットする場合には、イベントを再サブミットしたユーザーを反映するように、ID 情報が更新されます。管理者またはオペレーターとしてログインした異なるユーザーがイベントを再サブミットできるため、これらのユーザーには、イベントを処理するために必要な下流のコンポーネントに対する権限を付与する必要があります。

セキュリティーの実装について詳しくは、『アプリケーションとその環境の保護』を参照してください。

## 失敗イベントの検索

失敗イベントはデータベースに格納され、Failed Event Manager の検索機能を使用して検索されます。セル内のすべてのサーバー上の失敗したイベントをすべて検索するか、あるいはイベントの特定のサブセットを検索するかを選択できます。

### 始める前に

管理セキュリティーが有効になっている状態でこのタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

### このタスクについて

このトピックでは、セル内のすべての失敗イベントを検索する方法について説明します。デフォルトの照会を実行すると、SCA と JMS の失敗イベントがすべて返されます。

Business Process Choreographer がインストールされている場合に照会を実行すると、Business Process Choreographer の失敗イベント、強制終了イベント、および停止イベントも返されます。

失敗イベントの完全なリストを取得するには、以下の手順を実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. 「統合アプリケーション」 → 「Failed Event Manager」をクリックして、Failed Event Manager を始動します。
3. 「このサーバー上の失敗イベント」ボックスで、「失敗したすべてのイベントを取得」をクリックします。

### タスクの結果

「検索結果」ページが開き、セル内にある WebSphere Process Server の失敗したすべてのイベントのリストが表示されます。

### 次のタスク

失敗イベント内のデータの表示 (場合によっては変更)、失敗イベントの再サブミット、または削除を行うことができます。

## 基準によるイベントの検索

指定した基準と一致するイベントだけを検索するには、Failed Event Manager の「検索」ページを使用します。失敗イベントのタイプを指定して検索することも、基準 (失敗した時刻、イベントの宛先またはソース、例外またはビジネス・オブジェクトのタイプ、セッション ID、イベント順序付け修飾子 (WebSphere Process Server の場合のみ) など) を指定して検索することもできます。

### 始める前に

管理セキュリティが有効になっている状態でこのタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

### このタスクについて

サーバー上の失敗イベントの特定のサブセットを検索するには、以下の手順を実行します。

#### 手順

1. 管理コンソールが稼働していることを確認します。
2. 「統合アプリケーション」 → 「Failed Event Manager」をクリックして、Failed Event Manager を始動します。
3. 「このサーバー上の失敗イベント」ボックスで、「失敗したイベントの検索」をクリックします。
4. 「失敗したイベントの検索」ページの「イベント・タイプ」ボックスで、検索したいイベントのタイプを 1 つ以上選択します。
  - SCA
  - JMS
  - Business Process Choreographer
  - Business Flow Manager 保留キュー・メッセージ
5. Business Process Choreographer イベントを検索する場合は、「イベント状況」ボックスで選択されたイベント状況を確認します。Failed Event Manager は、Business Process Choreographer のすべての失敗イベント、停止イベント、強制終了イベントをデフォルトで返しますが、特定の状況のイベントだけを返すように変更することができます。
6. オプション: その場合、追加の検索基準を指定します。以下の表に、使用可能なオプションを示します。複数の基準を指定すると、照会時に AND 演算子が使用され、Failed Event Manager はすべての基準を満たすイベントだけを返します。

表 34. 検索基準

検索基準	使用するフィールド	サポートされるイベント・タイプ	使用上の注意
イベント失敗時の対象モジュール、コンポーネント、またはメソッド	モジュール コンポーネント 操作	SCA JMS Business Process Choreographer Business Flow Manager 保留キュー	これらの 1 つ以上のフィールドを使用して、特定のモジュール、コンポーネント、またはメソッドに関連付けられた失敗イベントを検索します。
イベントが失敗した期間	開始日 終了日	SCA JMS Business Process Choreographer Business Flow Manager 保留キュー	日時の形式はロケール固有です。各フィールドに例が示されています。指定された値の形式が正しくない場合、Failed Event Manager は警告を表示して、そのフィールドのデフォルト値に置き換えます。  時刻は常に、サーバーのローカル時間が使用されます。管理コンソールを実行している各マシンのローカル時刻によって時刻が更新されることはありません。
イベントが失敗したセッション	セッション ID	SCA	なし
イベントの発生元のモジュールまたはコンポーネント	ソース・モジュール ソース・コンポーネント	SCA	特定のソース・モジュールまたはコンポーネントから発生した失敗イベントだけを検索するには、これらのいずれかのフィールドまたは両方のフィールドを使用します。Failed Event Manager は、対話のタイプに関係なく、障害発生ポイントを基にしてソースを判断します。
失敗イベント内のビジネス・オブジェクトのタイプ	ビジネス・オブジェクト・タイプ	SCA	なし

表 34. 検索基準 (続き)

検索基準	使用するフィールド	サポートされるイベント・タイプ	使用上の注意
イベントにイベント順序付け修飾子が指定されているかどうか	イベント順序付け修飾子付き	SCA	この検索基準は、WebSphere Process Server の場合のみ有効です。
イベントが失敗したときにスローされた例外	例外テキスト	SCA	例外テキストの全部または一部をフィールドに指定して、その例外に関連付けられているすべてのイベントを検索します。

各フィールドと指定できる値について詳しくは、Failed Event Manager の「検索」ページのオンライン・ヘルプを参照してください。

7. 「OK」をクリックして検索を開始します。

### 次のタスク

失敗イベント内のデータの表示 (場合によっては変更)、失敗イベントの再サブミット、または削除を行うことができます。

## 失敗イベントのデータの操作

失敗イベントには、それぞれに関連付けられたデータがあります。多くの場合、そのデータはイベントを再サブミットする前に編集可能です。失敗イベントでは、2つのデータの基本型があります。イベントに関するデータと、ビジネス・データです。

### 失敗イベントに関するデータ

失敗イベントの各タイプには、以下のデータが関連付けられています。

- SCA イベント:
  - イベント ID、タイプ、状況、およびセッション ID
  - SCA コンポーネント間のサービス呼び出しタイプ
  - イベントが発生したモジュール名およびコンポーネント (ソース) 名
  - イベントの宛先モジュール名、コンポーネント名、およびメソッド名
  - イベントが失敗した時刻
  - イベントに関連付けられているデプロイメント・ターゲット
  - このイベントに対してイベント順序付け修飾子が宣言されているかどうかの指定
  - イベントの再サブミット先の宛先モジュール
  - 相関 ID (存在する場合)
  - イベントが失敗したときにスローされた例外
  - 再サブミットされたイベントの有効期限 (このデータは編集可能)

- イベントのトレース制御設定 (このデータは編集可能)
- JMS イベント:
  - イベント ID、タイプ、および状況
  - 対話タイプ
  - イベントの宛先モジュール名、コンポーネント名、およびメソッド名
  - イベントが失敗した時刻
  - イベントに関連付けられているデプロイメント・ターゲット
  - イベントが失敗したときにスローされた例外
  - イベントの再サブミット先の宛先モジュール
  - 相関 ID (存在する場合)
  - 再サブミットされたイベントの有効期限 (このデータは編集可能)
  - 失敗イベントに関連付けられている JMS 固有のプロパティ: 再配信カウント、送達モード、メッセージ優先順位、タイプ、replyTo 宛先、および再配信標識 (true または false)
- Business Process Choreographer イベント:
  - イベント ID、タイプ、および状況
  - イベントの宛先モジュール名およびコンポーネント名
  - イベントが失敗した時刻
  - イベントに関連付けられているデプロイメント・ターゲット
  - イベントに関連付けられているプロセス・インスタンス名
  - イベントに関連付けられている最上位プロセス ID
- Business Flow Manager 保留キュー・イベント:
  - イベント ID、タイプ、および状況
  - イベントが失敗した時刻
  - イベントに関連付けられているデプロイメント・ターゲット
  - プロセス・インスタンス ID (プロセス・インスタンスが存在しない場合は、0 が返される)
  - プロセス・インスタンスの名前および状態
  - 関連するプロセス・テンプレートの名前
  - アクティビティ・インスタンス名および ID
  - アクティビティ・テンプレート ID

## ビジネス・データ

SCA および Business Process Choreographer の失敗イベントには、通常、ビジネス・データが含まれています。ビジネス・データは、ビジネス・オブジェクト内にカプセル化することも、ビジネス・オブジェクトの一部でない単純データとすることも可能です。SCA の失敗イベントのビジネス・データは、Failed Event Manager の使用可能なビジネス・データ・エディターで編集できます。

## 失敗イベントのデータのブラウズ

失敗イベント・データや、そのイベントに関連したビジネス・データを表示する場合は、Failed Event Manager を使用します。

## 始める前に

管理セキュリティが有効になっている状態でこのタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

## このタスクについて

失敗イベント・データを表示するには、以下の手順を実行します。

### 手順

1. Failed Event Manager が開いていること、および失敗イベントのリストをシステム上で取得したことを確認します。
2. Failed Event Manager の「検索結果」ページで、データを参照する失敗イベントの ID (「イベント ID」列の項目) をクリックします。

「失敗イベントの詳細」ページが開き、イベントに関する情報のすべてが表示されます。

3. 失敗イベントにビジネス・データが含まれている場合は、「**ビジネス・データを編集**」をクリックして参照することができます。

「ビジネス・データ・エディター・コレクション」ページが開き、失敗イベントに関連付けられたビジネス・データを表示します。階層内の各パラメーター名はリンクです。パラメーターが単純データ型である場合、名前をクリックするとフォームが開き、パラメーターの値を編集できます。パラメーターが複合データ型の場合、名前をクリックすると、さらに階層が展開します。

## 失敗した SCA イベントのトレース・データまたは有効期限データの編集

「失敗イベントの詳細」ページでは、失敗イベントに関連付けられたトレース制御および有効期限の値を設定または変更できます。

## 始める前に

このタスクを実行するには、管理者またはオペレーターとしてログインしている必要があります。

## このタスクについて

**重要:** トレースまたは有効期限データに対する編集の内容は、イベントを再サブミットするまでローカルにのみ保管されます。イベントを再サブミットする前にその他の何らかのアクションを実行した場合、すべての編集は失われます。

失敗した SCA イベントは、イベントの処理のモニターに役立つようにトレース付きで再サブミットできます。トレースは、サービスまたはコンポーネントに対して設定でき、ログまたは Common Event Infrastructure (CEI) サーバーに送ることができます。「失敗イベントの詳細」ページで失敗イベント・データを表示すると、そのイベントのデフォルトのトレース値 `SCA.LOG.INFO;COMP.LOG.INFO` が表示されます。このデフォルトの設定値でイベントを再サブミットした場合、セッションが SCA サービスを呼び出すか、またはコンポーネントを実行したときに、トレースは行われません。

失敗した SCA イベントには有効期限が設定されているものもあります。ユーザーが、イベントを送信する非同期呼び出しについて有効期限を指定した場合、イベントが失敗してもデータは存在し続け、有効期限時刻は「失敗イベントの詳細」ページの「再サブミット有効期限時刻」フィールドに表示されます。有効期限が切れた失敗イベントは、正常に再サブミットできません。イベントを再サブミットするときに有効期限が切れないように、イベントの有効期限を編集することにより、2 度目の失敗を回避できます。

失敗イベントのトレース・データまたは有効期限データを編集するには、以下の手順を実行します。

#### 手順

1. Failed Event Manager が開いていること、および失敗イベントのリストをシステム上で取得したことを確認します。
2. Failed Event Manager の「検索結果」ページで、データを編集したい失敗イベントの ID (「イベント ID」列の項目) をクリックします。

「失敗イベントの詳細」ページが開きます。

3. イベントに有効期限があり、再サブミットする前に有効期限が切れる場合、「再サブミット有効期限時刻」フィールドの有効期限を編集します。

表示されている有効期限時刻は、サーバーのローカル時間です。このフィールドの値は、指定したロケールに応じたフォーマットにする必要があります。ロケールに応じた正しいフォーマットの例が、フィールドの上に表示されます。

4. 失敗イベントのトレースを使用可能に設定する場合、「トレース制御」フィールドに新しい値を指定します。トレース値の詳細情報については、WebSphere Business Process Management インフォメーション・センター内の、モニターに関するトピックを参照してください。
5. 以下のいずれかを実行します。
  - 編集したデータが正しく、イベントを再サブミットする場合、「再サブミット」をクリックして、サーバー・レベルで変更を行います。
  - 加えた変更を除去する場合、「ローカルの変更を取り消す」をクリックします。

編集済みの失敗イベントは、処理されるように再サブミットされ、Failed Event Manager から除去されます。

#### 関連タスク

358 ページの『失敗イベントの検索』

失敗イベントはデータベースに格納され、Failed Event Manager の検索機能を使用して検索されます。セル内のすべてのサーバー上の失敗したイベントをすべて検索するか、あるいはイベントの特定のサブセットを検索するかを選択できます。

### 失敗した SCA イベントのビジネス・データの編集

ビジネス・データは、ビジネス・オブジェクト内にカプセル化したものであっても、ビジネス・オブジェクトの一部でない単純データであってもかまいません。失敗イベントには、単純データとそれに関連付けられたビジネス・オブジェクトの両

方が含まれることもあります。失敗イベントを再サブミットする前に、ビジネス・データ・エディターを使用して、失敗イベントに関連したビジネス・データを編集します。

## 始める前に

管理セキュリティが有効になっている状態でこのタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

## このタスクについて

それぞれの失敗イベントについて、エディターでは関連ビジネス・データが階層形式で表示されます。表の上部のナビゲーション・ツリーが、パラメーターをナビゲートするにつれて更新されるため、階層内のどの場所にいるかを明確に把握できません。

編集できるのは、単純データ型 (String、Long、Integer、Date、Boolean など) ののみです。データ型が複合型 (配列またはビジネス・オブジェクトなど) の場合、配列またはビジネス・オブジェクトを構成する単純データ型に達するまでビジネス・データ階層をナビゲートする必要があります。複合データは、「パラメーター値」列に省略符号 (...) で示されます。

Failed Event Manager を使用して、Business Process Choreographer イベントのビジネス・データを編集することはできません。代わりに、失敗イベントの詳細ページの「呼び出しプロセスを **Business Process Explorer** で開きます」リンクをクリックし、Business Process Choreographer Explorer を使用してデータを変更します。

**重要:** ビジネス・データに対する編集の内容はすべて、ローカルで保管されます。失敗イベントを再サブミットするまで、サーバー上の対応するビジネス・データは変更されません。

失敗した SCA イベントに関連したビジネス・データを編集するには、以下の手順を実行します。

### 手順

1. Failed Event Manager が開いていること、および失敗イベントのリストをシステム上で取得したことを確認します。
2. Failed Event Manager の「検索結果」ページで、データを編集したい失敗イベントの ID (「イベント ID」列の項目) をクリックします。
3. 失敗イベントの詳細ページで「**ビジネス・データを編集**」をクリックして、「ビジネス・データ・エディター・コレクション」ページにアクセスします。

このページでは、失敗イベントに関連付けられたすべてのデータが階層図で表示されます。

4. 各パラメーターの名前をクリックして、ビジネス・データ階層をナビゲートします (これらは「パラメーター名」列でリンクとして表示されます)。値を編集したいパラメーターを見つけたら、その名前をクリックします。

パラメーターの値が編集可能な場合、「ビジネス・データ・エディター」ページが開きます。

5. 「パラメーター値」フィールドで、パラメーターの新しい値を指定します。
6. 「OK」をクリックします。

変更内容はローカルに保管され、「ビジネス・データ・エディター・コレクション」ページに戻ります。

7. 加えた変更を除去する場合、「ローカル・ビジネス・データの変更を取り消す」をクリックします。

すべての編集内容が除去され、ビジネス・データは元の状態に戻ります。

8. 編集したビジネス・データが正しければ、「再サブミット」をクリックして、サーバー・レベルで変更を行います。

編集済みの失敗イベントは、処理されるように再サブミットされ、Failed Event Manager から除去されます。

## 失敗イベントの再サブミット

イベントの送信を再試行する場合は、Failed Event Manager からイベントを再サブミットする必要があります。イベントを変更せずに再サブミットすることも、ビジネス・データ・パラメーターを編集してから再サブミットすることもできます。

失敗イベントを再サブミットすると、イベント全体ではなく、失敗したブランチに対してのみ処理が再開されます。

再サブミットした SCA イベントでは、イベント処理のモニターに役立つようにトレースが使用可能になっています。トレースはサービスまたはコンポーネントに対して設定でき、その出力はログまたは Common Event Infrastructure (CEI) サーバーに送信できます。

また、イベント固有のイベント ID を使用して、イベントの成功または失敗を追跡できます。再サブミットしたイベントが再び失敗した場合、そのイベントは、元のイベント ID および更新された失敗の時刻とともに Failed Event Manager に戻されます。

### 未変更の失敗イベントの再サブミット

1 つ以上の未変更の失敗イベントを再サブミットして、再度処理させることができます。イベント全体ではなく、失敗したブランチに対してのみ、処理が再開されます。

### このタスクについて

管理セキュリティが有効になっている状態でこのタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

#### 手順

1. Failed Event Manager が開いていること、および失敗イベントのリストをシステム上で取得したことを確認します。
2. 「検索結果」ページで、再サブミットしようとする失敗イベントの横のチェック・ボックスをそれぞれ選択します。
3. 「再サブミット」をクリックします。

## タスクの結果

選択した各イベントは処理されるように再サブミットされ、Failed Event Manager から除去されます。

## 失敗した SCA イベントのトレース付きでの再サブミット

失敗した SCA イベントの再サブミットをモニターすることにより、このイベントが成功したかどうかを確認することができます。Failed Event Manager は、すべての失敗イベント用に、トレースのオプションを提供しています。

## このタスクについて

トレースは、サービスまたはコンポーネントに対して設定でき、ログまたは Common Event Infrastructure (CEI) サーバーに出力できます。トレースの設定および表示について詳しくは、インフォメーション・センターのモニターに関するトピックを参照してください。

管理セキュリティが有効になっている状態でこのタスクを実行するには、管理者またはオペレーターとしてログインする必要があります。

## 手順

1. Failed Event Manager が開いていること、および失敗イベントのリストをシステム上で取得したことを確認します。
2. 「検索結果」ページで、再サブミットしようとする失敗イベントの横のチェック・ボックスをそれぞれ選択します。
3. 「トレース付き再サブミット」をクリックします。
4. 「トレース付き再サブミット」ページで、使用するトレースのレベルを「トレース制御」フィールドに指定します。

デフォルトでは、この値は SCA.LOG.INFO;COMP.LOG.INFO です。この設定値では、セッションが SCA サービスを呼び出すか、またはコンポーネントを実行したときに、トレースは行われません。

5. 「OK」をクリックして失敗イベントを再サブミットし、「検索結果」ページに戻ります。

## 次のタスク

再サブミットしたイベントのトレース・ログを表示するには、対応するコンポーネントのロガーを開くか、または CEI ログ・ビューアーを使用します。

## Business Process Choreographer の停止イベントの管理

Failed Event Manager および Business Process Choreographer Explorer を使用して、Business Process Choreographer の停止イベントを任意のプロセス状態で管理します。Business Process Execution Language (BPEL) インスタンスで例外が発生し、1 つ以上のアクティビティが停止状態になると、停止イベントが発生します。

## このタスクについて

Business Process Choreographer の停止イベントに関連付けられているプロセス・インスタンスを表示、補正、または強制終了することができます。さらに、そのイベントに関連付けられているアクティビティを、必要に応じて表示、変更、再実行して処理することができます。

長時間実行 BPEL プロセスから発生する停止イベントを管理するには、以下の手順を実行します。

### 手順

1. 管理コンソールが稼働していることを確認します。
2. 「統合アプリケーション」 → 「Failed Event Manager」をクリックして、Failed Event Manager を開きます。
3. 検索を実行して、管理する Business Process Choreographer の停止イベントを検索します。
4. 管理する停止イベントごとに、以下の操作を実行します。
  - a. 「検索結果」ページの「イベント ID」列にある停止イベント ID をクリックします。
  - b. イベントの詳細ページで、「Business Process Choreographer Explorer で呼び出しプロセスを開く」をクリックします。
  - c. Business Process Choreographer Explorer を使用して、イベントと関連アクティビティを管理します。

## 失敗イベントに関連するビジネス・プロセス・インスタンスの検索

失敗イベントがビジネス・プロセスから生成された場合、Failed Event Manager は、Business Process Choreographer Explorer にそのビジネス・プロセス・インスタンスを表示するためのリンクを提供します。

### 始める前に

このタスクを実行するには、管理者またはオペレーターとしてログインしている必要があります。

## このタスクについて

失敗イベントを生成したビジネス・プロセス・インスタンスを調べることで、イベントが失敗した状況や原因に関する追加情報が得られます。ビジネス・プロセス・インスタンスと失敗イベントは、共通のセッション ID によってリンクされます。

**注:** すべての失敗イベントがビジネス・プロセス・インスタンスから生成されるわけではありません。

失敗イベントに関連するビジネス・プロセス・インスタンスを検索して調べるには、以下の手順を実行します。

### 手順

1. 管理コンソールの内部から、Failed Event Manager を使用して、調べようとする失敗イベントを検索します。失敗イベントの検索方法の説明については、358ページの『失敗イベントの検索』を参照してください。
2. そのイベントの「失敗イベントの詳細」ページから、「呼び出しプロセスを **Business Process Explorer** で開きます」をクリックします。

## タスクの結果

新規ブラウザ・ウィンドウで Business Process Choreographer Explorer が開き、関連プロセス・インスタンスについての情報が表示されます。

## 失敗イベントに関連する Common Base Event の検索

失敗イベントは、1 つ以上の Common Base Event に関連していることがあります。Failed Event Manager は、Common Base Event ブラウザー内に関連 Common Base Event を表示するためのリンクを提供します。

### 始める前に

このタスクを実行するには、管理者またはオペレーターとしてログインしている必要があります。

### このタスクについて

関連 Common Base Event を調べることによって、元のイベントが失敗した状況や原因に関する追加情報が得られます。失敗イベントと関連 Common Base Event は、同じセッション ID によってリンクされます。

関連 Common Base Event を見つけて表示するには、以下の手順を実行します。

#### 手順

1. 管理コンソールの内部から、Failed Event Manager を使用して、調べようとする失敗イベントを検索します。失敗イベントの検索方法の説明については、358ページの『失敗イベントの検索』を参照してください。
2. そのイベントの「失敗イベントの詳細」ページから、「**関連の Common Base Event の参照**」をクリックします。

## タスクの結果

新規ブラウザ・ウィンドウで Common Base Event ブラウザーが開き、失敗した元のイベントに関連するすべての Common Base Event がリストされます。

## 失敗イベントの削除

失敗イベントを再サブミットしない場合、または失敗イベントの有効期限が切れた場合、Failed Event Manager を使用してサーバーから削除します。Failed Event Manager は、失敗イベントを削除する 3 つのオプションを提供しています。

### 始める前に

このタスクを実行するには、管理者またはオペレーターとしてログインしている必要があります。

## このタスクについて

1 つ以上の失敗イベントを削除するには、以下の手順を実行します。

### 手順

1. Failed Event Manager が開いていること、および失敗イベントのリストをシステム上で取得したことを確認します。
2. Failed Event Manager の「検索結果」ページから、以下のいずれかを実行します。
  - 1 つ以上の特定の失敗イベントを削除する場合、各イベントの横のチェック・ボックスを選択してから、「削除」をクリックします。
  - 失敗イベントのうち有効期限が切れたものだけを削除する場合、「期限切れイベントを削除」をクリックします。このようにすると、現在の一連の検索結果から、有効期限が切れたイベントのみが削除されます。
  - サーバー上のすべての失敗イベントを削除するには、「すべてクリア」をクリックします。

## Failed Event Manager のトラブルシューティング

このトピックでは、Failed Event Manager の使用中に発生する可能性がある問題について説明します。

**注:** このトピックでは、システムで失敗イベントを検出、変更、再サブミット、または削除するために Failed Event Manager を使用方法については説明しません。失敗イベントの管理について詳しくは、インフォメーション・センターの『*WebSphere Process Server の失敗イベントの管理*』を参照してください。

以下の表から、発生した問題を選択します。

問題	参照先
「検索」ページの「日付ごと」タブに値を入力できない	『入力した値が正しくない場合に、「日付ごと」フィールドと「開始日」フィールドの値が自動的にデフォルトに変更される』
期限切れのイベントを削除できない	371 ページの『「期限切れイベントを削除」機能を使用すると Failed Event Manager が中断するように見える』
失敗イベントが作成されない	371 ページの『失敗イベントが作成されない』

### 入力した値が正しくない場合に、「日付ごと」フィールドと「開始日」フィールドの値が自動的にデフォルトに変更される

検索ページの「開始日」フィールドと「終了日」フィールドには、ロケール固有の値を正しい形式で指定する必要があります。値の形式に不整合があると（年数に 2 桁ではなく 4 桁の数字を使用した、時間値を省略した、など）、Failed Event Manager は以下の警告メッセージを発行し、フィールドをデフォルト値に置き換えます。

CWMAN0017E: 入力された日付 *your\_incorrectly\_formatted\_date* が正しく解析されませんでした。日付 *default\_date* が使用されます。

「開始日」フィールドのデフォルト値は、January 1, 1970, 00:00:00 GMT に定義されています。

**重要:** Failed Event Manager の実装で表示される実際のデフォルト値は、ロケールおよびタイム・ゾーンによって異なります。例えば、米東部標準時 (EST) のタイム・ゾーンにあるマシンでロケールが en\_US に設定されている場合、「開始日」フィールドのデフォルト値は 12/31/69 7:00 PM になります。

「終了日」フィールドのデフォルト値は、常に現在の日付および時刻であり、ご使用のロケールおよびタイム・ゾーンに応じた形式で表示されます。

この問題を回避するには、各フィールドの上に示される例に従って、日付と時刻を常に注意深く入力してください。

### 「期限切れイベントを削除」機能を使用すると Failed Event Manager が中断するよう見える

現在の検索結果に失敗イベントが多く含まれている場合、または失敗イベントに大量のビジネス・データが含まれている場合に、「期限切れイベントを削除」ボタンを使用すると、Failed Event Manager が無期限に中断しているように見える場合があります。

この場合、Failed Event Manager は大量のデータ・セットを処理しているので、実際には中断していません。コマンドが完了すると、結果セットが最新表示されます。

### 失敗イベントが作成されない

失敗イベントが Recovery サブシステムによって作成されない場合には、以下のチェックリストを参照して、問題の原因を確認してください。

- wpsFEMgr アプリケーションが実行中であることを確認します。必要な場合は、再始動します。
- Failed Event Manager のデータベースが作成済みであり、接続がテスト済みであることを確認します。
- 必要とされる失敗イベントの宛先が SCA システム・バス上で作成済みであることを確認します。各デプロイメント・ターゲットに対して 1 つの失敗イベントの宛先が必要です。
- Recovery サービスに処理させるイベントに参加するすべての Service Component Architecture (SCA) 実装、インターフェース、またはパートナー参照に対して、サービス品質 (QoS) の **Reliability** 修飾子が Assured に設定されていることを確認します。



---

## 第 12 章 WebSphere Process Server のトラブルシューティング管理

トラブルシューティングは、問題の原因を検出して除去するプロセスです。以下の各トピックは、標準的な管理タスク中に発生する可能性のある問題を識別して解決するために役立ちます。

Business Process Choreographer または Common Event Infrastructure コンポーネントのトラブルシューティングについては、以下のいずれかを参照してください。

- WebSphere Process Server for Multiplatforms バージョン 6.1 インフォメーション・センター
- *Business Process Choreographer* の PDF
- *Common Event Infrastructure* の PDF

---

### アプリケーションのクロス・コンポーネント・トレースの使用

クロス・コンポーネント・トレースを使用すると、WebSphere Process Server および WebSphere Enterprise Service Bus のモジュールおよびコンポーネントに関連付けられている trace.log データを識別できます。trace.log データには、Service Component Architecture (SCA) 処理中に収集されるエラーおよびイベント情報 (破損データやランタイム例外など) が含まれている場合があります。WebSphere Process Server と WebSphere Enterprise Service Bus のコンポーネント間で受け渡しされる入力データおよび出力データも収集して、WebSphere Integration Developer を使用した問題判別に使用することができます。

### クロス・コンポーネント・トレースの使用可能化

クロス・コンポーネント・トレースを使用可能に設定して、処理中に収集される WebSphere Process Server および WebSphere Enterprise Service Bus のモジュールおよびコンポーネントに関連付けられたエラーおよびイベント情報を収集します。

#### 始める前に

このタスクを実行するには、管理者としてログインしている必要があります。

#### このタスクについて

クロス・コンポーネント・トレースを使用可能に設定するには、以下の手順を実行します。

#### 手順

1. 管理コンソールが実行中であることを確認してから、「トラブルシューティング」→「クロス・コンポーネント・トレース」をクリックして、「クロス・コンポーネント・トレース」ページを表示します。
2. トレースを使用可能に設定するサーバーを選択します。「状況」列で選択したサーバーの状況 (「実行中」または「実行されていない (not running)」) に注目

します。状況が「実行されていない (not running)」になっている場合、そのサーバーのランタイム・トレースは使用不可に設定されているため、指定できるのは構成値のみです。この場合、トレース・レベルが有効になるのは、サーバーの始動時または再始動時のみです。

3. 「構成」または「ランタイム」列で、トレースを使用可能に設定する各サーバーのドロップダウン・ボックスから「使用可能にする」を選択します。サーバーの始動時または再始動時にデータを収集する場合は、「構成」でトレースを使用可能に設定します。現在、サーバーが稼働している場合は、「ランタイム」でトレースを使用可能に設定してデータを収集します。
4. 設定を指定したら、「OK」をクリックして設定を保存します。

## タスクの結果

収集されたデータは trace.log ファイルに追加され、このファイルがパージされるときにデータもパージされます。詳しくは、376 ページの『Service Component Architecture (SCA) 処理と呼び出しチェーンのトラブルシューティング』を参照してください。

## データ・スナップショット付きクロス・コンポーネント・トレースの使用可能化

データ・スナップショット付きでクロス・コンポーネント・トレースを使用可能に設定して、Service Component Architecture (SCA) 処理に関連付けられたデータを収集し、WebSphere Process Server および WebSphere Enterprise Service Bus のモジュールおよびコンポーネントに関連付けられたチェーン・データと呼び出します。

### 始める前に

このタスクを実行するには、管理者としてログインしている必要があります。

### このタスクについて

クロス・コンポーネント・トレースを使用可能に設定するには、以下の手順を実行します。

#### 手順

1. 管理コンソールが実行中であることを確認してから、「トラブルシューティング」→「クロス・コンポーネント・トレース」をクリックして、「クロス・コンポーネント・トレース」ページを表示します。
2. トレースを使用可能に設定するサーバーを選択します。「状況」列で選択したサーバーの状況（「実行中」または「実行されていない (not running)」）に注目します。状況が「実行されていない (not running)」になっている場合、そのサーバーのランタイム・トレースは使用不可に設定されているため、指定できるのは構成値のみです。この場合、トレース・レベルが有効になるのは、サーバーの始動時または再始動時のみです。
3. 「構成」または「ランタイム」列で、トレースを使用可能に設定する各サーバーのドロップダウン・ボックスから「データ・スナップショット付きで使用可能 (enable with data snapshot)」を選択します。サーバーの始動時または再始動時

にデータを収集する場合は、「構成」でトレースを使用可能に設定します。現在、サーバーが稼働している場合は、「ランタイム」でトレースを使用可能に設定してデータを収集します。

4. 設定を指定したら、「OK」をクリックして設定を保存します。

## タスクの結果

収集された SCA データは systemout.log および trace.log ファイルに追加され、これらのファイルがパージされるときにデータもパージされます。WebSphere Process Server と WebSphere Enterprise Service Bus のコンポーネント間で受け渡しされる入力データおよび出力データが収集され、追加のファイルが logs\XCT ディレクトリに作成されます。これらのデータは、WebSphere Integration Developer での問題判別に使用できます。これらのファイルが不要になった場合は、管理者が削除します。詳しくは、376 ページの『Service Component Architecture (SCA) 処理と呼び出しチェーンのトラブルシューティング』を参照してください。

## クロス・コンポーネント・トレースの使用不可化

クロス・コンポーネント・トレースを使用不可に設定して、Service Component Architecture (SCA) 処理中に収集される WebSphere Process Server および WebSphere Enterprise Service Bus のモジュールおよびコンポーネントに関連付けられたエラーおよびイベント情報の収集を停止します。

### 始める前に

このタスクを実行するには、管理者としてログインしている必要があります。

### このタスクについて

クロス・コンポーネント・トレースを使用不可に設定するには、以下の手順を実行します。

#### 手順

1. 管理コンソールが実行中であることを確認してから、「トラブルシューティング」→「クロス・コンポーネント・トレース」をクリックして、「クロス・コンポーネント・トレース」ページを表示します。
2. トレースを使用不可に設定するサーバーを選択します。「状況」列で選択したサーバーの状況（「実行中」または「実行されていない (not running)」）に注目します。状況が「実行されていない (not running)」になっている場合、そのサーバーのランタイム・トレースは使用不可に設定されているため、指定できるのは構成値のみです。この場合、トレースの使用不可化が実施されるのは、サーバーの始動時または再始動時のみです。
3. 「構成」または「ランタイム」列で、トレースを使用不可に設定する各サーバーのドロップダウン・ボックスから「使用不可 (disable)」を選択します。
4. 設定を指定したら、「OK」をクリックして設定を保存します。

## タスクの結果

選択したサーバーのクロス・コンポーネント・トレースが使用不可に設定されます。データは収集されません。

## クロス・コンポーネント・トレースで収集されたデータの削除

データ収集が有効になった状態で、クロス・コンポーネント・トレースによりログ・ファイルにデータが追加されると、logs¥XCT ディレクトリーに追加のファイルが作成されます。systemout.log および trace.log ファイルは、WebSphere Application Server によって自動的に削除されるため、これらのファイルに追加されたデータを削除する必要はありません。データ収集ファイルが不要になったら、logs¥XCT ディレクトリーから手動で削除する必要があります。これらのファイルの削除は、WebSphere Integration Developer を使用して問題判別を行った後、またはこれらのファイルを参照するログ・ファイルが WebSphere Application Server によって削除された後に行います。

### 始める前に

各サーバーのログ・ディレクトリーに対する読み取りおよび書き込み権限が必要です。

### このタスクについて

クロス・コンポーネント・トレースによって収集されたデータを削除するには、以下の手順を実行します。

#### 手順

1. データの収集先の logs¥XCT ディレクトリーに移動し、問題判別のために WebSphere Integration Developer から表示可能な場所にディレクトリーの内容を移動します。
2. 収集されたデータが問題判別に不要であると判断した場合は、logs¥XCT ディレクトリーの内容を手動で削除します。

### タスクの結果

収集されたデータが削除されます。

## Service Component Architecture (SCA) 処理と呼び出しチェーンのトラブルシューティング

クロス・コンポーネント・トレースにより、WebSphere Process Server および WebSphere Enterprise Service Bus のモジュールとコンポーネントに関連付けられた systemout.log または trace.log データを識別できます。trace.log データには、SCA 処理中に収集された、破損データやランタイム例外などのエラーおよびイベント情報が記録されています。WebSphere Process Server と WebSphere Enterprise Service Bus のコンポーネント間で受け渡しされる入力データおよび出力データも収集して、WebSphere Integration Developer を使用した問題判別に使用することができます。

次のようなイベントを収集可能です。

- データが破損しているために処理中に発生したエラー。
- リソースが使用不可か、または障害が発生している場合のエラー。
- コード・パスの解釈。

管理コンソールから「クロス・コンポーネント・トレース」ページにアクセスし、次いで「トラブルシューティング」→「クロス・コンポーネント・トレース」をクリックできます。このページでは、トレース・データの収集元となるサーバーを選択できます。「構成」列を使用して、サーバーが始動または再始動するときに使用されるサーバーのトレース設定を指定してください。「ランタイム」列を使用して、実行中のサーバーのトレース設定を指定してください。

サーバーごとに、以下の設定のいずれかを使用できます。

#### 使用可能にする

この設定により、SCA 処理のトレースを使用可能にします。この設定から収集されるデータは、systemout.log ファイルと trace.log ファイルに追加され、それらのファイルがパージされるときにパージされます。

#### データ・スナップショット付きで使用可能

この設定により、SCA 処理のトレース、および WebSphere Process Server と WebSphere Enterprise Service Bus のコンポーネント間で渡される入出力データのトレースを使用可能にします。SCA 処理からのデータは systemout.log ファイルと trace.log ファイルに追加されます。WebSphere Process Server と WebSphere Enterprise Service Bus のコンポーネントからの入出力データは、logs¥XCT ディレクトリーに作成されるファイルに保存されます。

#### 使用不可

この設定により、選択したサーバーでのトレースを使用不可にします。

#### 収集データの処理および削除

- SCA 処理から収集されるデータは、systemout.log ファイルと trace.log ファイルに追加され、それらのファイルがパージされるときにパージされます。
- logs¥XCT ディレクトリーのファイルとして追加される入出力データは、問題判別のために WebSphere Integration Developer で表示可能な場所に移動できます。ファイルはその後、手動で削除できます。logs¥XCT にある入力ファイルと出力ファイルは、同時に作成された systemout.log ファイルおよび trace.log ファイルと関連があります。WebSphere Application Server が古い systemout.log ファイルと trace.log ファイルを削除すれば、logs¥XCT にある関連付けられた入力ファイルと出力ファイルも削除できます。一般に、指定された systemout.log ファイルに対して、多くの入力ファイルと出力ファイルが存在します。systemout.log ファイルと trace.log ファイルのタイム・スタンプは、削除する入力ファイルと出力ファイルを識別するために使用できます。systemout.log ファイルと trace.log ファイルの最も古い日付より古い入力ファイルと出力ファイルは、すべて削除するのが安全です。

---

## Failed Event Manager のトラブルシューティング

このトピックでは、Failed Event Manager の使用中に発生する可能性がある問題について説明します。

**注:** このトピックでは、システムで失敗イベントを検出、変更、再サブミット、または削除するために Failed Event Manager を使用方法については説明しませ

ん。失敗イベントの管理について詳しくは、インフォメーション・センターの『WebSphere Process Server の失敗イベントの管理』を参照してください。

以下の表から、発生した問題を選択します。

問題	参照先
「検索」ページの「日付ごと」タブに値を入力できない	370 ページの『入力した値が正しくない場合に、「日付ごと」フィールドと「開始日」フィールドの値が自動的にデフォルトに変更される』
期限切れのイベントを削除できない	371 ページの『「期限切れイベントを削除」機能を使用すると Failed Event Manager が中断するよう見える』
失敗イベントが作成されない	371 ページの『失敗イベントが作成されない』

## 入力した値が正しくない場合に、「日付ごと」フィールドと「開始日」フィールドの値が自動的にデフォルトに変更される

検索ページの「開始日」フィールドと「終了日」フィールドには、ロケール固有の値を正しい形式で指定する必要があります。値の形式に不整合があると（年数に 2 桁ではなく 4 桁の数字を使用した、時間値を省略した、など）、Failed Event Manager は以下の警告メッセージを発行し、フィールドをデフォルト値に置き換えます。

CWMAN0017E: 入力された日付 *your\_incorrectly\_formatted\_date* が正しく解析されませんでした。日付 *default\_date* が使用されます。

「開始日」フィールドのデフォルト値は、January 1, 1970, 00:00:00 GMT に定義されています。

**重要:** Failed Event Manager の実装で表示される実際のデフォルト値は、ロケールおよびタイム・ゾーンによって異なります。例えば、米東部標準時 (EST) のタイム・ゾーンにあるマシンでロケールが en\_US に設定されている場合、「開始日」フィールドのデフォルト値は 12/31/69 7:00 PM になります。

「終了日」フィールドのデフォルト値は、常に現在の日付および時刻であり、ご使用のロケールおよびタイム・ゾーンに応じた形式で表示されます。

この問題を回避するには、各フィールドの上を示される例に従って、日付と時刻を常に注意深く入力してください。

## 「期限切れイベントを削除」機能を使用すると Failed Event Manager が中断するよう見える

現在の検索結果に失敗イベントが多く含まれている場合、または失敗イベントに大量のビジネス・データが含まれている場合に、「期限切れイベントを削除」ボタンを使用すると、Failed Event Manager が無期限に中断しているよう見える場合があります。

この場合、Failed Event Manager は大量のデータ・セットを処理しているので、実際には中断していません。コマンドが完了すると、結果セットが最新表示されます。

## 失敗イベントが作成されない

失敗イベントが Recovery サブシステムによって作成されない場合には、以下のチェックリストを参照して、問題の原因を確認してください。

- wpsFEMgr アプリケーションが実行中であることを確認します。必要な場合は、再始動します。
- Failed Event Manager のデータベースが作成済みであり、接続がテスト済みであることを確認します。
- 必要とされる失敗イベントの宛先が SCA システム・バス上で作成済みであることを確認します。各デプロイメント・ターゲットに対して 1 つの失敗イベントの宛先が必要です。
- Recovery サービスに処理させるイベントに参加するすべての Service Component Architecture (SCA) 実装、インターフェース、またはパートナー参照に対して、サービス品質 (QoS) の **Reliability** 修飾子が Assured に設定されていることを確認します。

---

## Service Component Architecture と WebSphere MQ の通信のトラブルシューティング

Service Component Architecture (SCA) モジュールと WebSphere MQ キュー・マネージャーの間の通信は、SCA モジュール内のインポートおよびエクスポートと WebSphere MQ サーバー内のキューとの間のバインディングに依存します。この情報を使用して、WebSphere MQ メッセージを処理していないサーバーを判別します。

### 始める前に

このタスクは、管理コンソールにアクセスできるユーザーが、WebSphere MQ に依存する要求が処理されていないことを認識した場合を前提とします。また、このユーザーは、WebSphere MQ キュー・マネージャーの変更を自分で実行できるか、あるいは変更を WebSphere MQ 管理者に依頼できる必要があります。

### このタスクについて

Service Component Architecture (SCA) モジュールは、サーバーと WebSphere MQ キュー・マネージャーの間のバインディングに依存します。2 つのエンティティーの間の通信が原因で、メッセージが完全に処理されないことがあります。以下のステップは、分断の原因を特定し、メッセージが再び処理されるようにするために役立ちます。

### 手順

1. WebSphere MQ と通信している SCA モジュールを表示して、このモジュールが現在も処理中であることを確認します。このページにナビゲートするには、「アプリケーション」>「SCA モジュール」を使用します。

2. キュー・マネージャーを表示して、キュー・マネージャーが現在も作動可能であることを確認します。このタスクは、WebSphere MQ 管理コンソールで実行します。
3. SCA モジュールとキュー・マネージャーの間のバインディングを表示して、バインディングが正しいことを確認します。バインディングが正しくない場合は、バインディングを変更します。このページへ移動するには、「アプリケーション」→「SCA モジュール」>「*moduleName*」>「インポート|エクスポート」>「*importName|exportName*」>「バインディング」>「*bindingName*」を使用します。
4. 失敗したトランザクションを示している可能性のあるメッセージを見つけます。システム、SCA 固有のメッセージ領域、WebSphere MQ 固有のメッセージ領域、失敗イベント・キュー、およびその他の場所を調べて、障害の対象を特定する必要があります。
  - a. SystemOut.log を調べて、処理の障害を示すメッセージがないか確認します。

WebSphere MQ エラーがあった場合、スタック・トレース内のどこかの場所にリンクした MQException と WebSphere MQ 理由コード (例: 2059 は「キュー・マネージャーが使用不可です (queue manager unavailable)」) が存在します。

- b. AMQERRxx.LOG および WebSphere MQ FFDC ファイルをチェックして、WebSphere MQ エラーの原因を確認します。
- c. アプリケーション・キューを調べて、未処理のメッセージがあるかどうか確認します。WebSphere MQ とサービス統合バス (SIB) キューの両方を必ず調べます。
- d. WebSphere MQ の送達不能キューと SIB 例外の宛先を調べます。
- e. 失敗イベント・キューを調べて、対象となるアプリケーションに関連したメッセージがあるかどうか確認します。失敗イベントを探す方法については、『失敗イベントの検索』を参照してください。失敗イベントの検索については、『失敗イベントの管理』を参照してください。

---

## イベント順序付けのトラブルシューティング

イベント順序付けで問題が発生した場合は、このトピックに記載されている情報を参照してください。

### イベント順序付け修飾子に関する問題

以下のようにして、コンポーネント定義が正しいことを確認してください。

- メソッドでイベント順序付け修飾子が設定されていますか? インターフェースで修飾子が誤って設定されていると、イベント順序付けの検証に失敗します。
- パラメーター名は有効ですか?
- xpath エlementが有効であり、正しくプリミティブに解決されますか?
- メソッドに eventSequencing Elementが 1 つだけありますか? 各メソッドがサポートする eventSequencing Elementは 1 つのみです。
- メソッドに keySpecification Elementが 1 つだけありますか? 各メソッドがサポートする keySpecification Elementは 1 つのみです。

## デッドロック

ロックを行う操作が呼び出され、さらにその操作が、同じコンポーネントに対し、同じイベント順序付けキーおよびグループを使用して別の操作を呼び出すと、デッドロックが発生します。デッドロックを解決するには、`esAdmin` コマンドを使用して現行のロックをリストして解放します。

デッドロックを避けるため、イベント順序付けを実装するときには依存関係を十分に考慮してください。循環依存関係を持つ操作は必ず別のイベント順序付けグループに入れてください。

### BPPEL プロセスでのデッドロック

BPPEL (Business Process Execution Language) プロセスでイベント順序付けを使用する場合にデッドロックが発生することがあります。以下の両方のアクティビティーに対応する操作でイベント順序付け修飾子を設定すると、デッドロックが発生します。

- 複数インスタンス化 `receive` または `pick` アクティビティーで `createInstance` 属性を `yes` に設定している
- 相関セット指定で `initiation` 属性を `join` に設定している

このタイプのデッドロックを解決するには、`esAdmin` コマンドを使用して現行のロックをリストし、解放します。以後のデッドロックを防ぐため、必ずこれらのタイプの従属操作を別のイベント順序付けグループに入れてください。

### イベント順序付けコールバックでロックを解放できない

Recovery サブシステムで失敗した順序付けされたイベントを削除する際に、イベント順序付けコールバックによってイベントのロックを解放できない場合があります。これは通常、ターゲット・アプリケーションが削除されている場合や、システムの他のコンポーネント (データベースなど) が使用不可になっている場合に発生します。

この場合、Failed Event Manager によってエラー・メッセージが生成されます。`esAdmin` コマンドを使用して、失敗イベントに関連付けられているロックを手動で削除します。

### パフォーマンスの問題

イベント順序付けコンポーネントで使用されるメッセージング・エンジン・サーバーでメモリーの問題が発生する場合は、`install_root/properties/eventsequencing.properties` ファイルのランタイム・イベント順序付けプロパティを変更します。

`maxActiveMessages` プロパティは、コンポーネント宛先でロックされる現行メッセージの数を定義します。大きなメッセージが多数ロックされていると、パフォーマンスが低下し、メモリーの問題が発生する可能性があります。値 0 (ゼロ) を指定すると、ロックされるメッセージの数が無制限になる点に注意してください。デフォルトでは、`maxActiveMessages` プロパティは 100 に設定されています。値を変更する場合は、以下の式を使用することをお勧めします。

$average\_number\_of\_ES\_keys * average\_number\_of\_potential\_queued\_events\_per\_key + delta$

この *delta* は、同時処理可能な同一の順序付けキーを持つ順序付けされたイベント予想数の見積もり精度の標準偏差です。

**workItemRetryCount** プロパティにより、検証作業の試行回数の上限を設定します。検証作業項目は、非同期イベントがアンロックされ、処理を待機している従属イベントが存在するときに生成されます。この場合、ロックの作成と削除は別の作業単位で行われ、次のイベントが処理される前に作業検証タスクによって各作業単位が処理されます。**workItemRetryCount** は、デフォルトで -1 (再試行) に設定されています。

**workItemSleepTime** プロパティにより、作業検証の再試行間隔を指定します。**workItemSleepTime** は、デフォルトで 10 秒に設定されています。この値を低く設定すると、パフォーマンスが低下する可能性があります。

これらのプロパティを変更するには、以下の手順を実行します。

1. テキスト・エディターで `eventsequencing.properties` ファイルを開きます。
2. ご使用の環境に合わせて変更します。
3. ファイルを保存して閉じます。
4. 変更を有効にするため、イベント順序付けコンポーネントの一部であるアプリケーションをすべて停止してから再始動します。

---

## オブジェクト・リクエスト・ブローカー (ORB) のサービス設定のトラブルシューティング

管理コンソールのオブジェクト・リクエスト・ブローカー (ORB) のサービス・ページで「参照による受け渡し」を `true` に設定すると、オブジェクトの直列化と非直列化で問題が発生する場合があります。

### オブジェクトの直列化の問題と「参照による受け渡し」プロパティ

SCA は、ORB メソッド `javax.rmi.CORBA.Util.copyObject()` を呼び出してオブジェクトをコピーします。チェック・ボックスを選択して「参照による受け渡し」処理を有効にした場合、ディープ・コピーは作成されません。このため、オブジェクトの直列化と非直列化で問題が発生します。

オブジェクトの直列化と非直列化の問題は、Service Component Architecture (SCA) モジュール間の通信に問題を引き起こす可能性があります。例えば、`ServiceBusinessException` 例外がスローされた場合、クライアント・エンドでその例外が反映されず、`ServiceRuntimeException` 例外が発生する可能性があります。

### 「参照による受け渡し」を `true` に設定してオブジェクト直列化の問題を解決する

WebSphere Process Server におけるオブジェクトの直列化の問題を回避するには、「参照による受け渡し」をデフォルト値に設定します。「参照による受け渡し」のデフォルト設定は `false` です。これは、「参照による受け渡し」のチェック・ボックスが選択されていない状態です。

以下のステップで、「参照による受け渡し」プロパティ設定の確認方法を説明します。

1. 管理コンソールの「ORB サービス」ページにナビゲートします。  
「アプリケーション・サーバー」 → [ServerName] → 「コンテナ・サービス」  
→ 「ORB サービス」を選択します。
2. 「参照による受け渡し」のチェック・ボックスが選択されていないことを確認します。

Service Component Architecture は、「参照による受け渡し」プロパティの設定に応じて、メッセージ・コピーを作成します。

---

## デプロイメント環境のトラブルシューティング

処理が遅くなっているように見えたり、要求が失敗したりするときは、焦点を絞った方法を使用して、環境の問題の原因を判別します。ここで説明する方法は、非スタンドアロン・サーバー環境向けです。

### 始める前に

このタスクを実行するには、デプロイメント・マネージャーの管理コンソールにログインする必要があります。

**このタスクに必要なセキュリティ・ロール:** セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはオペレーターとして管理コンソールにログインする必要があります。

### このタスクについて

以下の症状に気付いた場合は、デプロイメント環境の状態を調査してください。

- アプリケーションが使用できない
- アプリケーションの動作が遅い
- アプリケーションが停止する
- スループットが低下する
- パフォーマンスが低下する

### 手順

1. このデプロイメント環境を記述したトポロジー・レイアウトを表示して、トポロジーの状況を確認します。
2. トポロジーを表示して、トポロジー内の各種のロールの状態を確認します。以降の調査のために、予期しない状態のロール、または警告状態のロールをメモしておきます。
3. 各ロールのエラー状態の原因となっているノードを探します。
4. すべてのノードが同期化されていることを確認します。

管理コンソールの「ノード」ページで、すべての非同期ノードを選択し、「同期」をクリックします。

5. すべてのバスに関連付けられているメッセージング・エンジンが稼働していることを確認します。

メッセージング・エンジンが稼働していない場合は、停止してから再始動してください。

6. エラー状態のノードに関連付けられているログを探し、そのログを表示してエラー・メッセージがないかどうかを確認します。
7. 訂正に影響するエラー・メッセージで説明されたアクションをすべて実行します。
8. エラーをすべて訂正して、影響を受けるノードを再始動します。

## タスクの結果

以前エラー状態だったノードが始動して、トポロジーの状況が「実行中」になります。

## 次のタスク

影響を受けるアプリケーションをすべて再始動します。

---

## ビジネス・ルール・マネージャーのトラブルシューティング

ビジネス・ルール・マネージャーの使用中に発生する可能性のある問題には、ログイン・エラー、ログイン競合、アクセス競合などがあります。

これらの問題のトラブルシューティングには、さまざまな対策を取ることができます。

## ログイン・エラーの解決

ログイン・エラーは、ログイン時に発生します。

### 始める前に

#### このタスクについて

ログイン・エラー・メッセージは、以下のとおりです。

ログインを処理できません。ユーザー ID とパスワードを確認して、再試行してください。

**注:** ログイン・エラーは、管理セキュリティが使用可能になっていて、ユーザー ID とパスワードのいずれかまたは両方に誤りがある場合にのみ発生します。

ログイン・エラーを解決するには、以下のステップを実行します。

#### 手順

1. エラー・メッセージで「OK」をクリックして、「ログイン」ページに戻ります。
2. 「ユーザー ID」と「パスワード」に有効な値を入力します。
  - パスワードで大/小文字を区別する必要がある場合は、Caps Lock キーがオンになっていないことを確認してください。
  - ユーザー ID とパスワードのスペルが正しいことを確認してください。
  - システム管理者に問い合わせ、ユーザー ID とパスワードが正しいことを確認してください。

3. 「ログイン」をクリックします。

### 次のタスク

ログイン・エラーが解決された場合は、ビジネス・ルール・マネージャーにログインできるようになります。エラーが解決されない場合は、システム管理者に連絡してください。

## ログイン競合エラーの解決

ログイン競合エラーは、同じユーザー ID を持つ別のユーザーが既にアプリケーションにログインしている場合に発生します。

### 始める前に

#### このタスクについて

ログイン競合メッセージは、以下のとおりです。

同じユーザー ID を使って、別のユーザーが現在ログインしています。次の中から選択してください。

通常、ユーザーがログアウトせずにブラウザを閉じると、このエラーが発生します。この状態が発生した場合、セッションがタイムアウトする前に次のログインが試行されると、ログイン競合になります。

**注:** ログイン競合エラーは、管理セキュリティが使用可能になっている場合にのみ発生します。

ログイン競合エラーを解決するには、以下の 3 つのオプションから選択します。

- ログイン・ページに戻る。

別のユーザー ID を使用してアプリケーションを開く場合は、このオプションを使用します。

- 同じユーザー ID を持つ別のユーザーをログアウトする。

別のユーザーをログアウトし、新しいセッションを開始する場合は、このオプションを使用します。

**注:** 他のセッションで行われた未公開のローカル変更は、すべて失われます。

- 同じユーザー ID を持つ別のユーザーのコンテキストを継承し、そのユーザーをログアウトする。

進行中の作業を続行する場合は、このオプションを使用します。前のセッションで保管された未公開のローカル変更は、すべて保持されます。ビジネス・ルール・マネージャーは、前のセッションで表示された最後のページを開きます。

## アクセス競合エラーの解決

1 人のユーザーがデータ・ソース内のビジネス・ルールを更新しているときに、別のユーザーが同時に同じルールを更新しようとすると、アクセス競合エラーが発生します。

## 始める前に

ローカル側の変更をリポジトリに公開すると、このエラーが報告されます。

### このタスクについて

アクセス競合エラーを解決するには、以下のアクションを実行します。

- エラーの原因となっているビジネス・ルールのソースを検索し、ローカル・マシンでの変更がまだ有効かどうかチェックします。別のユーザーが変更を行った後では、ローカル側で変更する必要がない場合があります。
- ビジネス・ルール・マネージャーで作業を続行する場合は、エラーが発生したビジネス・ルール・グループおよびルール・スケジュールをデータ・ソースから再ロードする必要があります。これは、エラーが発生したビジネス・ルール・グループおよびルール・スケジュールのローカル変更は使用できなくなるからです。エラーが報告されたルールの「公開および復帰」ページで、「再ロード」をクリックして、ビジネス・ルール・グループまたはルール・スケジュールのページを再ロードします。エラー状態でない他のビジネス・ルール・グループおよびルール・スケジュールのローカル変更は引き続き使用できます。

---

## メッセージング・バインディングのトラブルシューティング

バインディングでは、特定のタイプのバインディングに固有の各種エラー状態が発生する可能性があります。

### このタスクについて

エラー状態の処理方法は、該当するバインディングのタイプによって異なります。

## JMS バインディングのトラブルシューティング

JMS バインディングで発生した問題を診断し、修正できます。

### 実装例外

JMS のインポートおよびエクスポート実装は、さまざまなエラー状態に応じて、以下の 2 種類の例外のうちのいずれかを戻すことがあります。

- サービス・ビジネス例外: サービス・ビジネス・インターフェース (WSDL ポート・タイプ) で指定された障害が発生した場合に、この例外が戻されます。
- サービス・ランタイム例外: その他のすべてのケースで生成されます。ほとんどの場合、cause 例外には元の例外 (JMSEException) が含まれます。

例えばインポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。そのため、複数の応答を受信した場合や遅延応答 (SCA の応答有効期限が切れた応答) を受信した場合は、サービス・ランタイム例外がスローされます。この場合、トランザクションはロールバックされ、応答メッセージはキューからバックアウトされるか、または Failed Event Manager によって処理されます。

## 主な障害状態

JMS バインディングの主な障害状態は、トランザクションの意味構造、JMS プロバイダー構成、またはその他のコンポーネントの既存動作への参照に基づいて判別されます。主な障害状態は以下のとおりです。

- JMS プロバイダーまたは宛先に接続できない。

JMS プロバイダーに接続してメッセージを受信できない場合は、MDB リスナー・ポートを開始することができません。この状態は、WebSphere Application Server ログに記録されます。正常に取得されるまで (または期限切れとなるまで)、永続メッセージは宛先に残ります。

JMS プロバイダーに接続してアウトバウンド・メッセージを送信できない場合、送信操作を制御するトランザクションがロールバックされます。

- インバウンド・メッセージを解析できないか、アウトバウンド・メッセージを構成できない。

データ・バインディングまたはデータ・ハンドラーが失敗すると、作業を制御するトランザクションがロールバックされます。

- アウトバウンド・メッセージを送信できない。

メッセージを送信できないと、関連するトランザクションがロールバックされます。

- 複数のメッセージまたは予期しない遅延応答メッセージが返される。

インポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。また、応答を受信できる有効期間は、要求の SCA 応答有効期限修飾子によって決まります。応答を受信したとき、または有効期間を超えたときに、相関レコードが削除されます。予期しない応答メッセージを受信した場合や遅れて応答を受信した場合は、サービス・ランタイム例外がスローされます。

## Failed Event Manager に表示されない JMS ベースの SCA メッセージ

JMS の対話の失敗によって SCA メッセージが発生した場合は、Failed Event Manager でこのメッセージを見つけることとなります。Failed Event Manager にこのようなメッセージが表示されない場合は、JMS 宛先の基礎となる SIB 宛先の最大配信失敗回数の値に 1 よりも大きい値が設定されているかどうかを確認してください。この値を 2 以上に設定すると、JMS バインディングに対して SCA を呼び出す際に、Failed Event Manager と対話することができます。

### 関連概念



Failed Event Manager について

Failed Event Manager は、呼び出しの失敗を処理および再サブミットするための Web ベースのクライアントです。

## 汎用 JMS バインディングのトラブルシューティング

汎用 JMS バインディングで発生した問題は、診断して修正することができます。

## 実装例外

汎用 JMS のインポート実装とエクスポート実装は、さまざまなエラー状態に応じて、以下の 2 種類の例外のいずれかを戻すことがあります。

- サービス・ビジネス例外: サービス・ビジネス・インターフェース (WSDL ポート・タイプ) で指定された障害が発生した場合に、この例外が戻されます。
- サービス・ランタイム例外: その他のすべてのケースで生成されます。ほとんどの場合、cause 例外には元の例外 (JMSEException) が含まれます。

## 汎用 JMS メッセージの有効期限のトラブルシューティング

JMS プロバイダーの要求メッセージには有効期限が設定されます。

要求有効期限 は、JMS プロバイダーの要求メッセージの JMSEExpiration の時刻に達した場合の、その要求メッセージの有効期限を指します。その他の JMS バインディングの場合と同様に汎用 JMS バインディングも、インポートによって格納されたコールバック・メッセージの有効期限を発信要求と同じ有効期限に設定することにより、要求の有効期限を処理しています。コールバック・メッセージの有効期限の通知により、要求メッセージの有効期限が切れていることが示されます。クライアントには、ビジネス例外により期限切れを通知する必要があります。

コールバック宛先がサード・パーティー・プロバイダーに移動すると、このタイプの要求有効期限はサポートされなくなります。

応答有効期限 は、JMS プロバイダーの応答メッセージの JMSEExpiration の時刻に達した場合の、その応答メッセージの有効期限を指します。

汎用 JMS バインディングでは応答有効期限はサポートされていません。これは、サード・パーティー JMS プロバイダーの正確な期限切れ動作が定義されていないためです。ただし、応答を受信する場合、応答を受信する時点で応答有効期限が切れていないかどうかを確認できます。

アウトバウンド要求メッセージでは、JMSEExpiration 値は待機時間と、asyncHeader の requestExpiration 値 (設定されている場合) から算出されます。

## 汎用 JMS 接続ファクトリー・エラーのトラブルシューティング

汎用 JMS プロバイダーで特定のタイプの接続ファクトリーを定義すると、アプリケーションの開始時にエラー・メッセージを受信することがあります。この問題を回避するには、外部接続ファクトリーを変更します。

アプリケーションの起動時に、以下のエラー・メッセージを受け取ります。

```
MDB リスナー・ポート JMSConnectionFactory タイプが JMSDestination タイプと一致しません (MDB Listener Port JMSConnectionFactory type does not match JMSDestination type)
```

この問題は、外部接続ファクトリーの定義時に発生します。特に、JMS 1.1 (統合) 接続ファクトリー (Point-to-Point 通信およびパブリッシュ/サブスクライブ通信の両方をサポート可能な接続ファクトリー) ではなく JMS 1.0.2 トピック接続ファクトリーを作成すると、例外がスローされることがあります。

この問題を解決するには、以下のステップを実行します。

1. 使用している汎用 JMS プロバイダーにアクセスします。
2. 定義されている JMS 1.0.2 トピック接続ファクトリーを JMS 1.1 (統合) 接続ファクトリーに置き換えます。

新規に定義した JMS 1.1 接続ファクトリーを使用してアプリケーションを起動すると、エラー・メッセージは表示されないはずですが。

## JMS ベース・イベント

JMS の対話の失敗によって SCA メッセージが発生した場合は、JMS プロバイダーの管理機能を使用してイベントを管理します。

## WebSphere MQ バインディングのトラブルシューティング

WebSphere MQ バインディングで発生する障害または失敗の状態を診断し、このような状態を修正できます。

### 主な障害状態

WebSphere MQ バインディングの主な障害状態は、トランザクションの意味構造、WebSphere MQ 構成、またはその他のコンポーネントの既存の動作への参照に基づいて判別されます。主な障害状態は以下のとおりです。

- WebSphere MQ キュー・マネージャーまたはキューに接続できない。

WebSphere MQ に接続してメッセージを受信できない場合は、MDB リスナー・ポートを開始することができません。この状態は、WebSphere Application Server ログに記録されます。永続メッセージは、正常に取得されるまで (または WebSphere MQ により期限切れとなるまで) WebSphere MQ キューに残ります。

WebSphere MQ に接続してアウトバウンド・メッセージを送信できない場合は、送信操作を制御するトランザクションがロールバックされます。

- インバウンド・メッセージを解析できないか、アウトバウンド・メッセージを構成できない。

データ・バインディングが失敗すると、作業を制御するトランザクションがロールバックされます。

- アウトバウンド・メッセージを送信できない。

メッセージを送信できないと、関連するトランザクションがロールバックされません。

- 複数の応答メッセージまたは予期しない応答メッセージが返される。

インポートの場合、要求メッセージごとに 1 つの応答メッセージだけが戻されることを前提としています。そのため、複数の応答を受信した場合や遅延応答 (SCA の応答有効期限が切れた応答) を受信した場合は、サービス・ランタイム例外がスローされます。この場合、トランザクションはロールバックされ、応答メッセージはキューからバックアウトされるか、または Failed Event Manager によって処理されます。

## 誤用例: WebSphere MQ JMS バインディングとの比較

通常、WebSphere MQ インポートおよびエクスポートは、ネイティブ WebSphere MQ アプリケーションと相互協調処理し、WebSphere MQ メッセージ本体の内容全体をメディエーションに公開するように設計されています。一方、WebSphere MQ JMS バインディングは、WebSphere MQ に対してデプロイされている JMS アプリケーションと相互協調処理するように設計されています。これにより、メッセージは JMS メッセージ・モデルに基づいて公開されます。

以下のシナリオでは、WebSphere MQ バインディングではなく WebSphere MQ JMS バインディングを使用して作成する必要があります。

- JMS メッセージ駆動型 Bean (MDB) を SCA モジュールから呼び出す。この MDB は、WebSphere MQ JMS プロバイダーに対してデプロイされています。WebSphere MQ JMS インポートを使用します。
- SCA モジュールを J2EE コンポーネント・サーブレットまたは EJB から JMS を介して呼び出すことができるようにする。WebSphere MQ JMS エクスポートを使用します。
- WebSphere MQ 上で転送中の JMS MapMessage の内容のメディエーションを実行する。WebSphere MQ JMS エクスポートとインポート、および適切なデータ・バインディングを組み合わせて使用します。

WebSphere MQ バインディングと WebSphere MQ JMS バインディングの相互協調処理が予期される状況があります。特に、J2EE WebSphere MQ アプリケーションと非 J2EE WebSphere MQ アプリケーション間をブリッジングする場合は、WebSphere MQ エクスポートと WebSphere MQ JMS インポート (あるいはこの逆) を、適切なデータ・バインディングまたはメディエーション・モジュール (あるいはこの両方) と組み合わせて使用します。

### 未配布メッセージ

構成エラーなどが原因で WebSphere MQ がメッセージを対象の宛先に配信できない場合、メッセージは指定されている送達不能キューに送信されます。

このとき、メッセージ本体の先頭には、送達不能ヘッダーが追加されます。このヘッダーには、失敗の原因や元の宛先などの情報が含まれています。

---

## プロファイル固有のログ・ファイル

個々のプロファイルの特性とランタイム・アクティビティの詳細を記述したログ・ファイルがあります。これらのログ・ファイルは、各プロファイルのプロファイル・ディレクトリー内にあります。

各プロファイルごとにいくつかのログ・ファイルが作成されます。これらのログの一部では、プロファイルの作成に使用されたパラメーターが記述されています。このようなタイプのログ・ファイルは通常、プロファイルの構成が完了した後に変更されることはありません。その他のプロファイル固有のログは、実行時に送出されたエラー、警告、および情報メッセージを取り込むように継続的に更新されます。

これらのログ・ファイルの一部は、モニター対象として選択された Common Base Event (ビジネス・オブジェクト・データを含む場合があります) を取り込むためにも使用されます。

以下の表に、さまざまなタイプのプロファイル固有のログ・ファイルと、製品内部でのそれらの場所を示します。表では、変数 *install\_root* は、WebSphere Process Server のインストール・ディレクトリーを表します。変数 *profile\_root* は、プロファイルのルートの場所を表します。

**i5/OS** **i5/OS プラットフォームの場合:** 変数 *user\_data\_root* は、デフォルトのユーザー・データ・ディレクトリーを表します。

詳しくは、製品、プロファイル、およびツールのデフォルト・インストール・ディレクトリーを参照してください。

表 35. 実行時に更新されるプロファイル固有のログ・ファイル

ログ	内容
<p>First Failure Data Capture (ffdc) ログおよび例外ファイル (すべてのプロファイルに共通) は以下のディレクトリーにあります。</p> <ul style="list-style-type: none"> <li> <b>Linux</b> <b>UNIX</b> <b>Linux®</b> および <b>UNIX®</b> プラットフォームの場合: <i>profile_root/logs/ffdc</i> </li> <li> <b>Windows</b> <b>Windows</b> プラットフォームの場合: <i>profile_root¥logs¥ffdc</i> </li> <li> <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <i>profile_root/logs/ffdc</i> </li> </ul>	<p>個々のプロファイルの ffdc ログおよび例外ファイルが格納されています。2 つのタイプの ffdc ログがあります。プロファイル実行時に発生したすべてのエラーをまとめた単一のログ・ファイルと、スタック・トレースなどの詳細を含む多数のテキスト・ファイルです。さまざまなタイプのプロファイルの命名規則は、両方のファイルで、以下のように示されています。</p> <ul style="list-style-type: none"> <li> <b>デプロイメント・マネージャーのプロファイル:</b> <ul style="list-style-type: none"> <li>ログ・ファイル - <i>deployment_manager_name_exception.log</i>.</li> <li>テキスト・ファイル - <i>deployment_manager_name_hex_id_date_time.txt</i></li> </ul> </li> <li> <b>カスタム・プロファイル:</b> <ul style="list-style-type: none"> <li>ログ・ファイル - <i>node_agent_name_exception.log</i> および <i>server_name_exception.log</i></li> <li>テキスト・ファイル - <i>node_agent_name</i> (または) <i>server_name_hex_id_date_time.txt</i></li> </ul> </li> <li> <b>スタンドアロン・プロファイル:</b> <ul style="list-style-type: none"> <li>ログ・ファイル - <i>server_name_exception.log</i></li> <li>テキスト・ファイル - <i>server_name_hex_id_date_time.txt</i></li> </ul> </li> </ul>

表 35. 実行時に更新されるプロファイル固有のログ・ファイル (続き)

ログ	内容
<p>デプロイメント・マネージャー・ログ (デプロイメント・マネージャー・プロファイルのみ) は以下のディレクトリーにあります。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs/deployment_manager_name</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs¥deployment_manager_name</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs/deployment_manager_name</code></li> </ul>	<p>ユーザーは次のディレクトリー内の 4 つのログ・ファイルで主に作業することになります。</p> <ul style="list-style-type: none"> <li>• <code>startServer.log</code> —</li> </ul> <p>システムで検出されたシステム・パラメーターと、開始プロセス中にデプロイメント・マネージャーが送出したメッセージが含まれています。</p> <ul style="list-style-type: none"> <li>• <code>stopServer.log</code> —</li> </ul> <p>システムで検出されたシステム・パラメーターと、デプロイメント・マネージャーのシャットダウン時に送出されたメッセージが含まれています。</p> <ul style="list-style-type: none"> <li>• <code>SystemErr.log</code> —</li> </ul> <p>実行時にデプロイメント・マネージャーが生成したエラーおよび例外メッセージが含まれています。サーバーの稼働中は継続的に更新されます。</p> <ul style="list-style-type: none"> <li>• <code>SystemOut.log</code> —</li> </ul> <p>実行時にデプロイメント・マネージャーが生成したエラー、警告、および情報メッセージを含むすべてのメッセージが含まれています。サーバーの稼働中は継続的に更新されます。</p>
<p>ノード・エージェントのログ (カスタム・プロファイルのみ) は以下のディレクトリーにあります。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs/node_agent_name</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs¥node_agent_name</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs/node_agent_name</code></li> </ul>	<p>ユーザーは次のディレクトリー内の 4 つのログ・ファイルで主に作業することになります。</p> <ul style="list-style-type: none"> <li>• <code>startServer.log</code> —</li> </ul> <p>システムで検出されたシステム・パラメーターと、開始プロセス中にノード・エージェントが送出したメッセージが含まれています。</p> <ul style="list-style-type: none"> <li>• <code>stopServer.log</code> —</li> </ul> <p>システムで検出されたシステム・パラメーターと、ノード・エージェントのシャットダウン時に送出されたメッセージが含まれています。</p> <ul style="list-style-type: none"> <li>• <code>SystemErr.log</code> —</li> </ul> <p>実行時にノード・エージェントが生成したエラーおよび例外メッセージが含まれています。ノード・エージェントの稼働中は継続的に更新されます。</p> <ul style="list-style-type: none"> <li>• <code>SystemOut.log</code> —</li> </ul> <p>実行時にノード・エージェントが生成したエラー、警告、および情報メッセージを含むすべてのメッセージが含まれています。ノード・エージェントの稼働中は継続的に更新されます。</p>

表 35. 実行時に更新されるプロファイル固有のログ・ファイル (続き)

ログ	内容
<p>サーバー・ログ (カスタム・プロファイルおよびスタンドアロン・プロファイルのみ) は以下のディレクトリーにあります。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs/server_name</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs¥server_name</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs/server_name</code></li> </ul>	<p>ユーザーは次のディレクトリー内の 4 つのログ・ファイルで主に作業することになります。</p> <ul style="list-style-type: none"> <li>• <code>startServer.log</code> — システムで検出されたシステム・パラメーターと、開始プロセス中にサーバーが送出したメッセージが含まれています。</li> <li>• <code>stopServer.log</code> — システムで検出されたシステム・パラメーターと、サーバーのシャットダウン時に送出されたメッセージが含まれています。</li> <li>• <code>SystemErr.log</code> — 実行時にサーバーが生成したエラーおよび例外メッセージが含まれています。サーバーの稼働中は継続的に更新されます。</li> <li>• <code>SystemOut.log</code> — 実行時にサーバーが生成したエラー、警告、および情報メッセージを含むすべてのメッセージが含まれています。Common Event Infrastructure (CEI) から送出されたモニター対象のイベントも含まれています (Common Base Event フォーマット)。これらのイベントには、モニター用に指定されたレベルのビジネス・オブジェクト・データ (FINE、FINER、または FINEST) も含まれる場合があります。サーバーの稼働中は継続的に更新されます。</li> </ul>
<p>ノード統合のログ・ファイルは、以下のディレクトリーにあります (デプロイメント・マネージャー以外のプロファイルのみに適用されます)。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs</code></li> </ul>	<p>カスタム・プロファイル、拡張されたプロファイル、またはスタンドアロン・プロファイルをデプロイメント・マネージャーに統合しようと試みると、以下の 2 つのログ・ファイルが生成されます。</p> <ul style="list-style-type: none"> <li>• <code>addNode.log</code> — 関連のあるサーバー環境情報と、プロファイルの統合を試みたときに生成されたメッセージが含まれています。</li> <li>• <code>isFederated.log</code> — デプロイメント・マネージャーがプロファイルを統合するために使用したコマンドがリストされています。</li> </ul>

表 35. 実行時に更新されるプロファイル固有のログ・ファイル (続き)

ログ	内容
<p>Integrated Solutions Console アプリケーションのデプロイメント・ログ・ファイルの場所を以下に示します (デプロイメント・マネージャー・プロファイルおよびスタンドアロン・プロファイルの場合のみ)。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs/iscinstall.log</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs¥iscinstall.log</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs/iscinstall.log</code></li> </ul>	<p>iscinstall.log ファイルには、デプロイメント・マネージャー・プロファイルまたはスタンドアロン・プロファイル内にある管理コンソール・アプリケーションのデプロイメントに関する情報が含まれています。</p>
<p>インストール検査ツールのログ・ファイル場所を以下に示します (デプロイメント・マネージャー・プロファイルおよびスタンドアロン・プロファイルの場合のみ)。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs/ivtClient.log</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs¥ivtClient.log</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs/ivtClient.log</code></li> </ul>	<p>このログ・ファイルには、インストール検査ツールが生成した出力が含まれています。デプロイメント・マネージャー・プロファイルまたはスタンドアロン・プロファイルを作成した後に、ファースト・ステップ・コンソールからこのプログラムを開始できます。このログには、基本的な構成情報と、ツールの実行時に表示されたメッセージが含まれています。</p>
<p>プロファイル作成用に生成されたコマンドについて詳しく記述したログ・ファイルの場所を以下に示します。</p> <ul style="list-style-type: none"> <li>• <b>Linux</b> <b>UNIX</b> <b>Linux</b> および <b>UNIX</b> プラットフォームの場合: <code>profile_root/logs/updateserverpolicy.log</code></li> <li>• <b>Windows</b> <b>Windows</b> プラットフォームの場合: <code>profile_root¥logs¥updateserverpolicy.log</code></li> <li>• <b>i5/OS</b> <b>i5/OS</b> プラットフォームの場合: <code>profile_root/logs/updateserverpolicy.log</code></li> </ul>	<p>このファイルには、サーバー環境変数を設定してプロファイルを作成するために製品が使用するコマンドのシーケンスが含まれています。すべてのプロファイル・タイプがこのファイルを含みます。</p>

---

## 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。

〒106-8711

東京都港区六本木 3-2-12

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渉外

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を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

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、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

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

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

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM Corporation  
1001 Hillsdale Blvd., Suite 400  
Foster City, CA 94404  
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

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

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の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。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。(c) (お客様の会社名) (西暦年)。このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。(C) Copyright IBM Corp. \_年を入れる\_。All rights reserved.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

### プログラミング・インターフェース情報

プログラミング・インターフェース情報は、このプログラムを使用してアプリケーション・ソフトウェアを作成する際に役立ちます。

一般使用プログラミング・インターフェースにより、お客様はこのプログラム・ツールのサービスを利用するアプリケーション・ソフトウェアを作成できます。

ただし、この情報には、診断、修正、および調整情報が含まれている場合があります。診断、修正、調整情報は、お客様のアプリケーション・ソフトウェアのデバッグ支援のために提供されています。

**警告:** 診断、修正、調整情報は、変更される場合がありますので、プログラミング・インターフェースとしては使用しないでください。

### 商標

IBM、IBMロゴ、および [ibm.com](http://ibm.com) は、International Business Machines Corporation の米国およびその他の国における商標または登録商標です。これらおよび他の IBM 商標に、この情報の最初に現れる個所で商標表示 (® または ™) が付されている場合、これらの表示は、この情報が公開された時点で、米国において、IBM が所有する登録商標またはコモン・ロー上の商標であることを示しています。このような商標は、その他の国においても登録商標またはコモン・ロー上の商標である可能性があります。現時点での IBM の商標リストについては、[www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml) の「Copyright and trademark information」をご覧ください。

Microsoft および Windows は、Microsoft Corporation の米国およびその他の国における登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における登録商標です。

Java および JavaScript は、Sun Microsystems, Inc. の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

この製品には、Eclipse Project (<http://www.eclipse.org>) により開発されたソフトウェアが含まれています。



IBM WebSphere Process Server for Multiplatforms バージョン 6.2





Printed in Japan