

**IBM WebSphere Business Integration
Adapters**



Adapter for SunGard FRONT ARENA ユーザーズ・ガイド

バージョン 1.0.x

**IBM WebSphere Business Integration
Adapters**



Adapter for SunGard FRONT ARENA ユーザーズ・ガイド

バージョン 1.0.x

お願い

本書および本書で紹介する製品をご使用になる前に、195 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM WebSphere Business Integration Adapter for SunGard FRONT ARENA (プロダクト ID 5724-i48) バージョン 1.0.x、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

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

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

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

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

(URL は、変更になる場合があります)

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： IBM WebSphere Business Integration Adapters
Adapter for SunGard FRONT ARENA User Guide
Version 1.0.x

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2004.7

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

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

© Copyright International Business Machines Corporation 2004. All rights reserved.

© Copyright IBM Japan 2004

目次

本書について	v
対象読者	v
本書の前提条件	v
関連文書	v
表記上の規則	vi
本リリースの新機能	vii
リリース 1.0.x の新機能	vii
第 1 章 概要	1
作業ロードマップ	1
用語	1
アダプターのアーキテクチャー	3
アプリケーションとアダプター間の通信	7
イベント通知	10
保証付きイベント・デリバリー	12
ビジネス・オブジェクト要求	13
第 2 章 アダプターのインストール	19
アダプター環境	19
前提条件となる作業	21
アダプターに対する WebSphere MQ の使用可能化	33
アダプターと関連ファイルのインストール	33
デフォルトのアダプター構成プロパティ	33
インストールの検証	34
第 3 章 FRONT ARENA のビジネス・オブジェクト	39
ビジネス・オブジェクト・メタデータ	39
ビジネス・オブジェクト構造	39
カスタム・ビジネス・オブジェクト・ハンドラー	40
第 4 章 データ・ハンドラーの構成	43
FRONT ARENA メッセージとデータ・ハンドラー	43
データ・ハンドラー・メタオブジェクトの構成	46
データ・ハンドラーのイベント通知サポート	49
第 5 章 コネクターの構成	51
Connector Configurator の概要	51
Connector Configurator の始動	52
System Manager からの Configurator の実行	53
コネクター固有のプロパティ・テンプレートの作成	53
新しい構成ファイルを作成	56
既存ファイルの使用	57
構成ファイルの完成	58
構成ファイル・プロパティの設定	59
構成ファイルの保管	66
構成ファイルの変更	67
構成の完了	67

グローバル化環境における Connector Configurator の使用 67

第 6 章 ブリッジの構成 69

ブリッジのパラメーター	69
構成ファイル	69
イベント処理	72
要求処理	74
ユーザー定義キーおよび FRONT ARENA キー	77
SMTP アラート	78
WBI アダプターとの通信	79
他のアプリケーションとの通信	81

第 7 章 アプリケーションの構成 85

AMB の構成	85
AMBA の構成	85

第 8 章 アダプターの実行 89

アダプターの構成	89
保証付きイベント・デリバリーの使用可能化	94
キューの URI (Uniform Resource Identifiers)	99
メタオブジェクトの構成	101
アダプターの複数インスタンスの作成	113
始動ファイルの構成	115
コネクターの始動	115
コネクターの停止	117
ブリッジの始動	117
ブリッジの停止	118

第 9 章 アダプターのトラブルシューティング 119

アダプターのエラー処理	119
アプリケーションのトレース	120
ErrorQueue の検査	120
アプリケーションへの接続不可	121
アーカイブ・キューの管理	121
アンサブスクライブ・キューの管理	121
エラー処理	121
要求処理	123
要求対応処理	124
アダプターのトレース	125
ブリッジのトレース	126

付録 A. コネクターの標準構成プロパティ 127

新規プロパティと削除されたプロパティ	127
標準コネクター・プロパティの構成	127
標準プロパティの要約	128
標準構成プロパティ	133

付録 B. コネクター固有のプロパティ	147
FRONT ARENA イベント通知のサポート	147
サービス要求	151
付録 C. ビジネス・オブジェクトの作成 および変更	157
概要	157

アダプターのビジネス・オブジェクトの構造	157
特記事項	195
プログラミング・インターフェース情報	197
商標	197
索引	199

本書について

IBM[®] WebSphere[®] Business Integration Adapter ポートフォリオは、先進の e-business テクノロジー、エンタープライズ・アプリケーション、レガシー・システム、およびメインフレーム・システムを統合的に接続する機能を提供します。この製品セットには、ビジネス・プロセス統合のコンポーネントをカスタマイズ、作成、および管理するためのツールやテンプレートが含まれています。

本書では、IBM WebSphere Business Integration Adapter for SunGard[®] FRONT ARENA[™] のインストール、構成、ビジネス・オブジェクト開発、およびトラブルシューティングについて説明します。

対象読者

本書は、お客様のサイトで WebSphere Business Integration システムのサポートおよび管理を担当するコンサルタント、開発者、およびシステム管理者を対象としています。

本書の前提条件

本書の読者は、WebSphere Business Integration システム、ビジネス・オブジェクトとコラボレーションの開発、および WebSphere MQ アプリケーションについて十分な知識と経験を持っている必要があります。

関連文書

この製品に付属する資料の完全セットで、すべての WebSphere Business Integration Adapters のインストールに共通な機能とコンポーネントについて説明します。また、特定のコンポーネントに関する参考資料も含まれています。

以下のサイトから、関連資料をインストールすることができます。

- アダプターの一般情報、WebSphere Message Brokers (WebSphere MQ Integrator、WebSphere MQ Integrator Broker、WebSphere Business Integration Message Broker) でのアダプターの使用、WebSphere Application Server でのアダプターの使用については、次の IBM WebSphere Business Integration Adapters InfoCenter をご覧ください。
<http://www.ibm.com/websphere/integration/wbiadapters/infocenter>
- WebSphere InterChange Server でのアダプターの使用については、次の IBM WebSphere InterChange Server InfoCenter をご覧ください。
<http://www.ibm.com/websphere/integration/wicserver/infocenter>
<http://www.ibm.com/websphere/integration/wbicollaborations/infocenter>
- WebSphere Message Brokers の詳細については、以下をご覧ください。
<http://www.ibm.com/software/integration/mqfamily/library/manualsa/>
- WebSphere Application Server の詳細については、以下をご覧ください。
<http://www.ibm.com/software/webservers/appserv/library.html>

上記のサイトには資料のダウンロード、インストール、および表示に関する簡単な説明が記載されています。

注: 本書の発行後に公開されたテクニカル・サポートの技術情報や速報に、本書の対象製品に関する重要な情報が記載されている場合があります。これらの情報は、WebSphere Business Integration Support Web サイト (<http://www.ibm.com/software/integration/websphere/support/>) にあります。関心のあるコンポーネント・エリアを選択し、「Technotes」セクションと「Flashes」セクションを参照してください。

表記上の規則

本書では、以下のような規則を使用しています。

Courier フォント	コマンド名、ファイル名、入力情報、システムが画面に出力した情報など、記述されたとおりの値を示します。
太字	初出語を示します。
<i>イタリック、イタリック</i>	変数名または相互参照を示します。
青のアウトライン	オンラインで表示したときのみ見られる青のアウトラインは、相互参照用のハイパーリンクです。アウトラインの内側をクリックすると、参照先オブジェクトにジャンプします。
{ }	構文の記述行の場合、中括弧 {} で囲まれた部分は、選択対象のオプションです。1 つのオプションのみを選択する必要があります。
[]	構文の記述行の場合、大括弧 [] で囲まれた部分は、オプションのパラメーターです。
...	構文の記述行の場合、省略符号 ... は直前のパラメーターが繰り返されることを示します。例えば、option[,...] は、複数のオプションをコンマで区切って指定できることを意味します。
< >	命名規則により、1 つの名前の各エレメントを個々に判別できるようにするために、不等号括弧で囲みます。例えば、<server_name><connector_name>tmp.log のように使用します。
/, ¥	本書では、ディレクトリー・パスに円記号 (¥) を使用します。UNIX システムの場合には、円記号をスラッシュ (/) に置き換えてください。すべての製品のパス名は、ご使用のシステムで製品がインストールされたディレクトリーを基準とした相対パス名です。
%text% および \$text	% 記号で囲まれたテキストは、Windows の text システム変数またはユーザー変数の値を示します。UNIX 環境での同等の表記は \$text です。これは、UNIX 環境変数 text の値を示します。
<i>ProductDir</i>	製品のインストール先ディレクトリーを表します。

本リリースの新機能

リリース 1.0.x の新機能

2004 年 6 月:

バージョン 1.0.x は、Adapter for SunGard FRONT ARENA Adapter Guide の最初のリリースです。

第 1 章 概要

- 『作業ロードマップ』
- 『用語』
- 3 ページの『アダプターのアーキテクチャー』
- 7 ページの『アプリケーションとアダプター間の通信』
- 10 ページの『イベント通知』
- 12 ページの『保証付きイベント・デリバリー』
- 13 ページの『ビジネス・オブジェクト要求』

この章では、理解する必要がある用語を説明する概要を示し、アダプターの処理について説明します。アダプターをインストール、構成、および使用する前にアダプターについて理解しておくことが重要です。

作業ロードマップ

Adapter for FRONT ARENA を使用するには、表 1 に説明されている作業を実行する必要があります。

表 1. アダプターの使用: 作業ロードマップ

作業	関連手順 (参照先)	詳細情報 (参照先)
コネクターのインストール	19 ページの『第 2 章 アダプターのインストール』	「 <i>WebSphere Business Integration Adapters</i> インストール・ガイド」
ビジネス・オブジェクトとメタオブジェクトの構成	39 ページの『第 3 章 FRONT ARENA のビジネス・オブジェクト』	「ビジネス・オブジェクト開発ガイド」
データ・ハンドラーの構成	43 ページの『第 4 章 データ・ハンドラーの構成』	「データ・ハンドラー・ガイド」
コネクターの構成	51 ページの『第 5 章 コネクターの構成』	69 ページの『第 6 章ブリッジの構成』、85 ページの『第 7 章 アプリケーションの構成』、147 ページの『付録 B. コネクター固有のプロパティ』、および「コネクター開発ガイド」
コネクターの実行	89 ページの『第 8 章 アダプターの実行』	「コネクター開発ガイド」
コネクターの保守	119 ページの『第 9 章 アダプターのトラブルシューティング』	

用語

アダプターについて理解するには、以下の用語を理解する必要があります。

アダプター

統合ブローカーとアプリケーション（またはテクノロジー）との間の通信をサポートするコンポーネントを備えている、WebSphere Business Integration システムのコンポーネント。アダプターには、コネクター、メッセージ・ファイル、および構成ツールが必ず含まれています。また、Object Discovery Agent (ODA) やデータ・ハンドラーも含まれていることがあります。

アダプター・フレームワーク

IBM が提供する、アダプターの構成と実行のためのソフトウェア。アダプター・フレームワークには、ランタイム・コンポーネントとして、Java ランタイム環境、コネクター・フレームワーク、および Object Discovery Agent (ODA) ランタイムが組み込まれています。コネクター・フレームワークには、コネクターを新規開発するときに必要となるコネクター・ライブラリー (C++ および Java) が含まれています。ODA ランタイムには、ODA を新規開発するときに必要となる Object Development Kit (ODK) のライブラリーが含まれています。構成用コンポーネントとしては、以下のツールが用意されています。

- Business Object Designer
- Connector Configurator
- Log Viewer
- System Manager
- Adapter Monitor
- Test Connector
- アダプターに関連付けられた Object Discovery Agent (ODA) (用意されていない場合もあります)

Adapter Development Kit (ADK)

アダプター開発のための実例をいくつか備えた開発キット。実例には、コネクターや Object Discovery Agent (ODA) の実例も含まれます。

Arena Message Broker (AMB)

FRONT ARENA AMB は、メッセージ指向のミドルウェアで、パブリッシュ/サブスクライブ・プロトコルに基づいて、内部および外部のアプリケーション同士が互いに通信できるストア・アンド・フォワード機能を備えています。

コネクター

ビジネス・オブジェクトを使用して、統合ブローカーにイベント関連の情報を送信し (イベント通知)、統合ブローカーから要求関連の情報を受信する (要求処理)、アダプターのコンポーネント。コネクターは、コネクター・フレームワークと、コネクターのアプリケーション固有のコンポーネントで構成されています。

コネクター・フレームワーク

コネクターのアプリケーション固有のコンポーネントと統合ブローカーの間の相互作用を管理する、コネクターのコンポーネント。このコンポーネントは、必要な管理サービスをすべて備えており、コネクターが必要とするメタデータをリポジトリから取得します。コネクター・フレームワークは Java で記述されており、C++ で記述されたアプリケーション固有のコンポ

ーネットをサポートできるように C++ 拡張が組み込まれています。コネクター・フレームワークのコードはすべてのコネクターで共通です。

コネクター・コントローラー

コラボレーションと相互作用する、コネクター・フレームワークのサブコンポーネント。コネクター・コントローラーは、InterChangeServer 内で動作するもので、アプリケーション固有のビジネス・オブジェクトと汎用ビジネス・オブジェクトの間のマッピングを開始し、コラボレーションのビジネス・オブジェクト定義に対するサブスクリプションを管理します。

統合ブローカー

異種のアプリケーションの間でデータを統合する、WebSphere Business Integration システムのコンポーネント。通常、統合ブローカーはさまざまなサービスを備えています。このサービスには、データをルーティングする機能、統合プロセスを決定する規則のリポジトリ、各種アプリケーションに接続する機能、および統合を容易にする管理機能が含まれます。

WebSphere Business Integration システム

多様なソースの間で情報を移動してビジネス関連の情報を交換し、エンタープライズ環境内の異種のアプリケーションの間で情報の処理とルーティングを行う、エンタープライズ・ソリューション。このビジネス・インテグレーション・システムは、統合ブローカーと 1 つ以上のアダプターで構成されています。

WebSphere MQ Integrator Broker、バージョン 2.2

WebSphere MQ キュー間でメッセージの変換とルーティングを行う、メッセージ・ブローカー製品。このテクノロジーにより、アプリケーションは、リモート・キューとの間でメッセージを送受信して非同期的に通信することができます。MQ Integrator Broker での大きな変更点は、ユーザー定義のロジックに基づいてメッセージをフォーマット、格納、およびルーティングできる機能を追加するメッセージ・フローが追加されたことです。

アダプターのアーキテクチャー

アダプターはメタデータ主導型です。アダプターは、エンタープライズ・メッセージング・システムにアクセスするための API である Java™ Message Service (JMS) の MQ インプリメンテーションを使用します。

Adapter for FRONT ARENA は、WebSphere MQ キューを使用して、FRONT ARENA とブローカーとの間の双方向の非同期データ交換を可能にします。データは、FRONT ARENA のキューとブローカーとの間で、テキスト・メッセージ形式で送信されます。データを任意のブローカーが処理できるビジネス・オブジェクトに変換する場合は、FRONT ARENA データ・ハンドラーを使用します。

同期交換方式でアダプターを使用する場合は、6 ページの『同期した要求および応答による対話』を参照してください。

このアダプターを使用すると、WebSphere MQ Adapter は、FRONT ARENA アプリケーション、バージョン 1.5 とメッセージを交換できます。

Adapter for FRONT ARENA が使用できるソリューションは、適切なアダプターがインストールされている FRONT ARENA 以外の全社的情報システム・アプリケー

ションと FRONT ARENA との間のビジネス・データ交換を統合するために統合ブローカーを使用するソリューションです。

コネクタは、アプリケーション固有のコンポーネントとコネクタ・フレームワークから成り立っています。アプリケーション固有のコンポーネントには、特定のアプリケーションに応じて調整されたコードが含まれます。アダプター・フレームワークのコードは、すべてのアダプターに共通です。コネクタ・フレームワークは、統合ブローカーとアプリケーション固有のコンポーネントの間を中継します。アダプター・フレームワークは、統合ブローカーとアプリケーション固有のコンポーネントの間で以下のサービスを提供します。

- ビジネス・オブジェクトの受信と送信
- 始動メッセージと管理メッセージの交換の管理

本書では、アダプター・フレームワークとコネクタについて説明します。本書では、これらのコンポーネントの両方をアダプターとして参照します。統合ブローカーとアダプターとの関係の詳細については、「*IBM WebSphere InterChange Server システム管理ガイド*」を参照してください。

注: すべての WebSphere Business Integration アダプターは、統合ブローカーと連携して動作します。Adapter for FRONT ARENA は、InterChange Server 統合ブローカーと連携して動作します。詳細は、「*テクニカル入門 (IBM WebSphere InterChange Server)*」に説明されています。

FRONT ARENA から統合ブローカーへの非同期メッセージ

取引データが FRONT ARENA に配置されると、INSERT_TRADE メッセージがネイティブの FRONT ARENA メッセージ形式で生成され、5 ページの図 1 に示すように WebSphere MQ 出力キューに配置されます。(この図では、FRONT ARENA と統合ブローカーは異なるキュー・マネージャーを使用して異なるマシンにインストールすることが前提になっており、したがって FRONT ARENA からの出力に対してリモート・キュー定義を用意する必要があり、統合ブローカーに対してローカルな入力キューを接続することが前提になります。1 つのキューが、FRONT ARENA からの出力キューと統合ブローカーへの入力キューの 2 つの役割を果たします。)

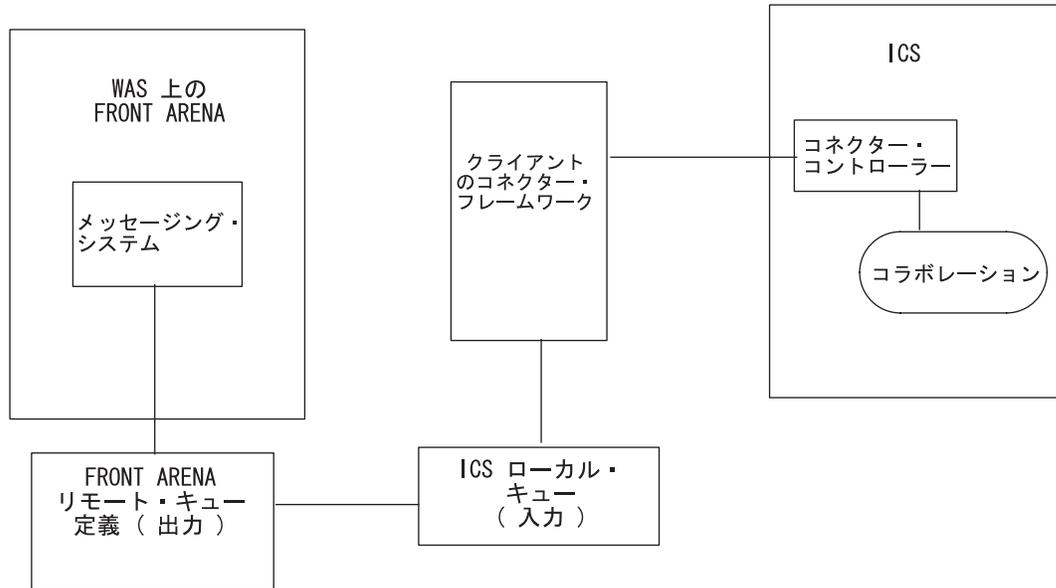


図1. アダプターのアーキテクチャー

FRONT ARENA のデータ・イベントを検出するため、アダプターは FRONT ARENA の出力キューをポーリングして新規メッセージを検索します。アダプターは、新規メッセージを検出すると、それを入力キューに渡し、データ・ハンドラーを呼び出して、このメッセージを FRONT ARENA から発信されているデータの構造に固有のビジネス・オブジェクトに変換します。さらに、このビジネス・オブジェクトを InterChange Server 内部のコネクタに渡します。コネクタはマップを呼び出して、FRONT ARENA 固有のビジネス・オブジェクトから汎用のビジネス・オブジェクトを生成し、これを 1 つ以上のコラボレーション・オブジェクトに配信します。コラボレーション・オブジェクトがビジネス・オブジェクトを処理すると、汎用のビジネス・オブジェクトは、アプリケーション固有のビジネス・オブジェクトにマップされ、これがバックエンド・アプリケーションに対して構成されているアダプター (WebSphere Business Integration Adapter for SAP など) に配信されます。

次の図では、ブリッジ、コネクタ、および WebSphere MQ によって、FRONT ARENA が WBI コンテキスト内に統合される仕組みが視覚化されています。

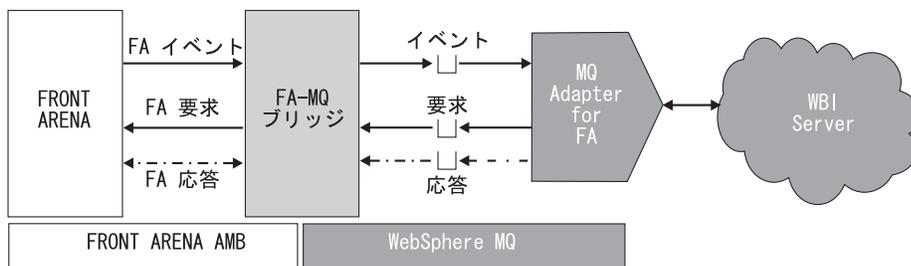


図2. アーキテクチャーの上位概念図

InterChange Server から FRONT ARENA への非同期メッセージ

逆方向では、Adapter for FRONT ARENA はコラボレーションからビジネス・オブジェクトを受け取り、データ・ハンドラーを使用してビジネス・オブジェクトをメッセージに変換し、このメッセージを FRONT ARENA WebSphere MQ キューに配信します。

同期した要求および応答による対話

同期した要求と応答による対話では、次のトピックで説明するように、FRONT ARENA アプリケーションへの機能追加やカスタマイズが必要です。

FRONT ARENA から InterChange Server への要求

アダプターを使用すると、FRONT ARENA メッセージング・システムから InterChange Server またはその他の外部システムへの要求と応答による対話の同期メッセージ・フローをセットアップできます。この方法の詳細については、FRONT ARENA バージョン 1.5 と IBM WebSphere Business Integration システムとの統合の資料を参照してください。

InterChange Server から FRONT ARENA への要求

注: この方法では、ビジネス・オブジェクトが InterChange Server から到着したときに FRONT ARENA で実行されるコマンドのカスタマイズが必要です。これらのコマンドは、メッセージから ReplyTo キューを検索し、ResponseTimeout の時間間隔内に応答をキューに配置します。FRONT ARENA のコマンドの作成およびカスタマイズについては、「*Programmer's Guide for SunGard FRONT ARENA, 1.5, 5.4*」を参照してください。

同期処理と非同期処理との対比

アダプターのブリッジ・コンポーネントを構成すると、要求を同期と非同期の両方の方式で処理できます。非同期モードの場合、要求は単純に ARENA Message Broker (AMB) に転送されます。要求が正常に実行されるかどうかは検査されません。要求の発信元に応答は返送されません。

同期モードの場合は、アダプターのブリッジ・コンポーネントが要求処理の結果を検査し、最初に要求を出したアプリケーションにこの結果を戻します。AMB は非同期のメッセージング・システムであるため、クライアント・アプリケーションは要求に対する同期応答を得られません。応答の到着は後になる場合があるため、クライアント・アプリケーションは対応する要求と応答を関連付ける必要があります。

AMB には、要求と応答の相関サポート機能は組み込まれていません。したがって、相関は、要求メッセージと応答メッセージのペイロードの要素に基づいてのみ実行できます。

イベント通知

FRONT ARENA アプリケーションで発生したデータ・イベントの通知は、アダプターのポーリング機能によって実現されます。アダプターは複数の入力キューをポーリングできます。その際、各入力キューをラウンドロビン方式でポーリングし、各入力キューから指定された数のメッセージを検索します。ポーリング中に検索する

メッセージごとに、アダプターは動的な子メタオブジェクト (ビジネス・オブジェクトに指定されている場合) を追加します。子メタオブジェクトの値はアダプターに対し、メッセージ・フォーマットとメッセージが検索された入力キューの名前を、属性に取り込むよう指示できます。

アダプターは、入力キューのメッセージを取り込むと、メッセージ・ヘッダーに格納されている FORMAT フィールドに関連付けられているビジネス・オブジェクト名を検索します。メッセージ本文は、該当するビジネス・オブジェクトの新規インスタンスと共にデータ・ハンドラーに渡されます。ビジネス・オブジェクト名がフォーマットと関連付けられていない場合は、メッセージ本文のみがデータ・ハンドラーに渡されます。メッセージの内容がビジネス・オブジェクトに正常に取り込まれると、アダプターはそのビジネス・オブジェクトがサブスクライブされているかどうかを検査し、InterChange Server に配信します。

ビジネス・オブジェクトと WebSphere MQ メッセージ・ヘッダー

メッセージの処理に使用されるビジネス・オブジェクトのタイプと動詞は、WebSphere MQ メッセージのヘッダーに含まれる FORMAT フィールドによって決定されます。アダプターはメタオブジェクトの項目を使用して、ビジネス・オブジェクト名および動詞を決定します。ビジネス・オブジェクト名と動詞を格納するメタオブジェクトを構築し、WebSphere MQ メッセージ・ヘッダーの FORMAT フィールドのテキストに関連付けます。

必要に応じて、アダプターに渡されるビジネス・オブジェクトの子として追加される動的なメタオブジェクトを作成できます。子メタオブジェクトの値は、アダプター全体を対象として指定されている静的なメタオブジェクトに指定されている値をオーバーライドします。子メタオブジェクトが定義されていない場合、または子メタオブジェクトに必要な変換プロパティが定義されていない場合、デフォルトでは、アダプターが値の静的メタオブジェクトを調べます。単一の静的なアダプター・メタオブジェクトの代わりに、またはそれを補足するために、1 つ以上の動的な子メタオブジェクトを指定することができます。

アプリケーションとアダプター間の通信

Adapter for FRONT ARENA を使用すると、WebSphere InterChange Server は、FRONT ARENA アプリケーションとビジネス・オブジェクトを交換できるようになります。

Adapter for FRONT ARENA は、WebSphere MQ Adapter、ブリッジ・コンポーネント、およびデータ・ハンドラーで構成されています。アダプターは、WebSphere MQ にインプリメントされている Java Message Service (JMS) を使用します。JMS は、エンタープライズ・メッセージング・システムにアクセスするためのオープン・スタンダード API です。JMS は、ビジネス・アプリケーションがビジネス・データとイベントを非同期的に送受信できるように設計されています。

アダプターのブリッジ・コンポーネントは、FRONT ARENA AMB メッセージング・システムと WebSphere MQ メッセージング・システムとの間で、メッセージを双方向に転送します。メッセージの内容については処理しません。メッセージ内容の変換は実行されません。ただし、メッセージング・システムに依存するヘッダー

のみが交換されます。メッセージ、正確にはメッセージのペイロードが FRONT ARENA システムから送信されるか、または最終的に FRONT ARENA システムに到着すると、このメッセージはいずれの場合もネイティブの FRONT ARENA 形式になります。アダプターを介して FRONT ARENA と通信するアプリケーションは、この形式に対応できる必要があります。その後、キューに格納されているメッセージは、WebSphere MQ Adapter またはその他のサード・パーティー・アプリケーションからアクセスできます。

FRONT ARENA アプリケーションには、専用のミドルウェア層である ARENA Message Broker (AMB) があります。これは、パブリッシュとサブスクライブのパラダイム (枠組み) を採用している非同期のメッセージング・システムです。AMB 提供の API を介して FRONT ARENA にアクセスする場合は、外部アプリケーションの使用をお勧めします。API が提供されているのは、AMB にアクセスし、使用するためです。このことは、データ (取引など) を電子的に供給したり、データを抽出したりするすべてのアプリケーションに推奨される統合のポイントです。

API は、FRONT ARENA システムとその他の金融市場アプリケーションとを同期化するために使用できます。これに対応して FRONT ARENA をセットアップすると、新規に作成した FRONT ARENA オブジェクト、または既存の FRONT ARENA オブジェクト (取引や場所など) に対する変更が AMB 上パブリッシュにされます。外部アプリケーションは、AMB に登録でき、これらの通知のすべてまたは一部に対してサブスクライブできます。反対方向の場合、外部アプリケーションは、適切なメッセージを AMB に書き込むことにより、新規の FRONT ARENA オブジェクトを作成したり、既存のオブジェクトを変更したりできます。

AMB の API は、下位層の C インターフェースであるため、API 呼び出しの前後に広範囲のメモリー管理が必要になります。アダプターのブリッジ・コンポーネントは、その片側を FRONT ARENA AMB に接続し、他方の側を WebSphere MQ キューを介してアダプターに接続します。アダプターのブリッジ・コンポーネントは、一連の構成可能な FRONT ARENA オブジェクトに対する通知を作成、更新、および削除するためにサブスクライブして、FRONT ARENA から受信したすべての通知イベント・メッセージを WebSphere MQ キューに渡します。ブリッジは、アダプターから受信したサービス要求メッセージを FRONT ARENA に転送し、必要に応じて要求の結果をアダプターに戻します。アダプターのデータ・ハンドラー・コンポーネントは、ネイティブの FRONT ARENA メッセージ形式に対応します。

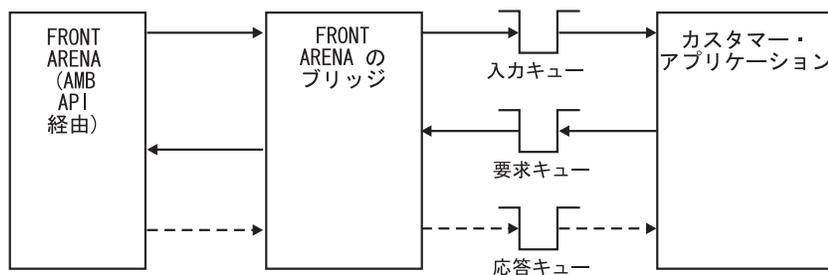


図3. ブリッジ・コンポーネントの直接の使用

ブリッジは、次の 2 つの状況で使用できます。

- FRONT ARENA と WebSphere MQ メッセージで直接通信するカスタマー・アプリケーションが使用する。

- WebSphere Business Integration (WBI) ベースのビジネス・インテグレーション・シナリオで、Adapter for FRONT ARENA と組み合わせて使用する。

メッセージ要求

図 4 に、メッセージ要求通信を示します。doVerbFor() メソッドがコラボレーションからビジネス・オブジェクトを受け取ると、アダプターはそのビジネス・オブジェクトをデータ・ハンドラーに渡します。データ・ハンドラーがビジネス・オブジェクトを FRONT ARENA のネイティブ・テキスト・メッセージに変換すると、アダプターがこれをキューへのメッセージとして送ります。このとき、JMS 層は適切な呼び出しを実行してキュー・セッションを開き、メッセージの経路を指定します。

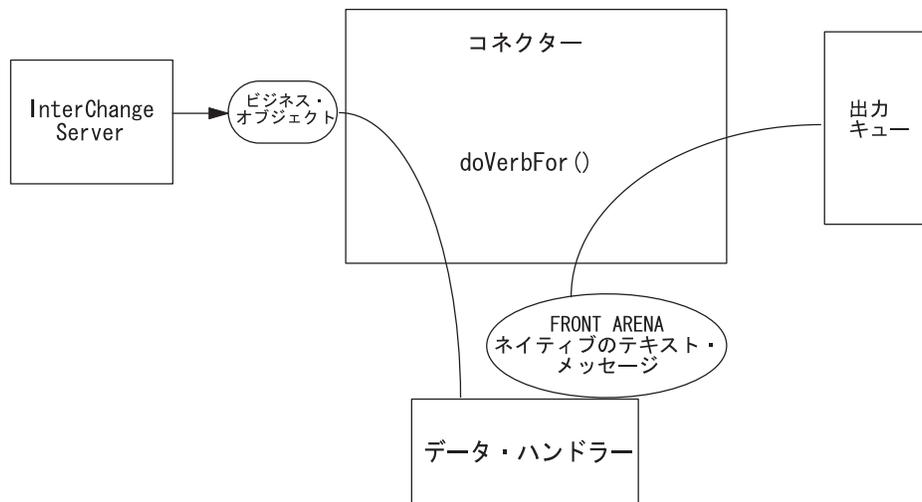


図 4. アプリケーションとアダプター間の通信方法: メッセージ要求

メッセージの戻り

図 5 には、メッセージの戻り方向を示します。pollForEvents() メソッドは、次の該当するメッセージを入力キューから検索します。メッセージは実行中のキューに入れられ、処理が完了するまでキュー内に残ります。アダプターは最初に、静的メタオブジェクトまたは動的メタオブジェクトのいずれかを使用して、そのメッセージ・タイプがサポートされているかどうかを調べます。サポートされている場合、アダプターは構成されているデータ・ハンドラーにメッセージを渡し、データ・ハンドラーがそのメッセージをビジネス・オブジェクトに変換します。設定される動詞には、そのメッセージ・タイプに対して定義されている変換プロパティが反映されます。次に、アダプターは、そのビジネス・オブジェクトがコラボレーションによってサブスクライブされているかどうかを調べます。サブスクライブされている場合、getAppEvents() メソッドがビジネス・オブジェクトを InterChange Server にデリバリーし、実行中のキューからメッセージが削除されます。

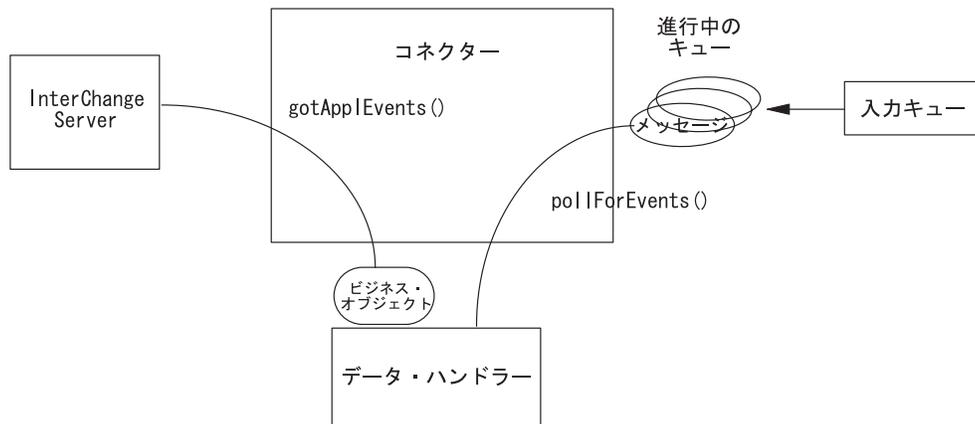


図5. アプリケーションとアダプター間の通信方法: メッセージの戻り

イベント通知

イベント通知の場合、アダプターは、FRONT ARENA AMB に書き込まれたイベントを検出します。イベント通知メッセージは、新規に作成されたオブジェクトまたは変更されたオブジェクトごとに、対象となるオブジェクトのタイプを含む事前定義トピックの下位にある AMB 上にパブリッシュされます。イベント処理は非同期であるため、FRONT ARENA メッセージは、応答を待たずにキューに書き込まれます。

AMB クライアント・アプリケーションは、特定のトピックの下にパブリッシュされているすべてのメッセージに対してサブスクライブできます。クライアント・アプリケーションは、このようにして FRONT ARENA システムの変更内容を認識できるようになります。例えば、FRONT ARENA システムがマスター・アプリケーションになっているシナリオでは、この仕組みに基づいて、その他の金融市場アプリケーションと FRONT ARENA との同期性を保持できます。Adapter for FRONT ARENA のブリッジ・コンポーネントは、AMB クライアント・アプリケーションとして動作します。ブリッジ・コンポーネントは、FRONT ARENA から受信した情報を単独では使用しませんが、WebSphere MQ キューを介してサード・パーティに渡します。ブリッジにサブスクライブし、ブリッジによって転送されるイベントのタイプは、その構成によって異なります。これらの設定は、対応する FRONT ARENA 構成パラメーター (厳密には、対応する FRONT ARENA AMBA パラメーター) と整合している必要があります。

メッセージの検索

アダプターは、pollForEvents() メソッドを使用して、定期的な間隔でメッセージのキューをポーリングします。アダプターは、メッセージを見つけると、それをキューから取り出して調べ、そのフォーマットを判別します。フォーマットがアダプターの統計メタオブジェクトで定義されている場合、アダプターはメッセージ本文と、このフォーマットに関連したビジネス・オブジェクトの新規インスタンスの両

方を FRONT ARENA データ・ハンドラーに渡します。データ・ハンドラーは、ビジネス・オブジェクトのデータを取り込むと想定されています。判別されたフォーマットが静的メタオブジェクトで定義されていない場合、アダプターはメッセージの本文のみをデータ・ハンドラーに渡します。データ・ハンドラーはメッセージに対する正しいビジネス・オブジェクトを判別して作成し、取り込むと想定されています。イベント失敗のシナリオについては、121 ページの『エラー処理』を参照してください。

アダプターは、最初に入力キューとのトランザクション・セッションを開いて、メッセージを処理します。アダプターは、次に、すべてのメッセージを進行中のキューに移動します。メッセージは、処理が完了するまで進行中のキューに保持されます。処理中にアダプターが予期しないエラーでシャットダウンした場合、メッセージは元の入力キューには戻されず、実行中のキュー内に残されます。

注: WebSphere MQ とのトランザクション・セッションでは、キュー内の要求されたアクションがすべて実行され、キューからイベントが除去される前にコミットされる必要があります。したがって、アダプターはキューからメッセージを取得した場合、次の 3 つの条件が揃うまで取得をコミットしません。

1. メッセージがビジネス・オブジェクトに変換された。
2. ビジネス・オブジェクトが `gotApp1Events()` メソッドによって InterChange Server に配信された。
3. 戻り値を受け取った。

メッセージのリカバリー

アダプターの初期化時には、コネクターのシャットダウンなどが原因で完全に処理されなかったメッセージが進行中キュー内にあるかどうかを検査されます。コネクターの構成プロパティ `InDoubtEvents` を使用すると、そのようなメッセージのリカバリー処理に関する 4 つのオプション (`fail on startup`、`reprocess`、`ignore`、または `log error`) のうち、いずれかを指定できます。

始動時に失敗

`Fail on Startup` オプションを使用すると、アダプターの初期化時に進行中キュー内のメッセージが検出された場合、エラー・ログは記録されますが、コネクターは即時にシャットダウンします。ユーザーまたはシステム管理者は、検出されたメッセージを調べ、これらのメッセージを完全に削除するかまたは別のキューに移動するなどの適切な処置を取る必要があります。

メッセージの再処理

`Reprocess` オプションを使用すると、アダプターの初期化時に進行中キュー内のメッセージが検出された場合、以降のポーリング中にこれらのメッセージが最初に処理されます。アダプターは、進行中キュー内のメッセージをすべて処理した後で、入力キューからのメッセージの処理を開始します。

メッセージの無視

`ignore` オプションを指定した場合、初期化の際、アダプターが実行中のキュー内にメッセージを検出すると、アダプターはそれを無視しますが、シャットダウンはしません。

エラーのロギング

Log Error オプションを使用すると、アダプターの初期化時に進行中キュー内のメッセージが検出された場合、エラー・ログが記録されますが、コネクタはシャットダウンしません。

メッセージのアーカイブ

コネクタ・プロパティ `ArchiveQueue` が指定されて有効なキューを示す場合、アダプターは正常に処理されたすべてのメッセージのコピーをアーカイブ・キューに入れます。 `ArchiveQueue` が未定義の場合、メッセージは処理後に破棄されます。アンサブスクライブされたメッセージまたはエラーを含むメッセージのアーカイブの詳細については、121 ページの『エラー処理』を参照してください。

注: JMS 規則により、検索したメッセージを即時に別のキューに送信することはできません。メッセージをアーカイブして再デリバリーできるようにするために、アダプターは、オリジナルのメッセージから本文とヘッダー（該当する場合のみ）を複製した第 2 のメッセージを最初に生成します。FRONT ARENA メッセージング・サービスとの競合を回避するには、JMS に必要なフィールドのみを複製します。したがって、`FORMAT` フィールドは、アーカイブまたは再デリバリーされるメッセージにコピーされる唯一の追加メッセージ・プロパティとなります。

保証付きイベント・デリバリー

保証付きイベント・デリバリー機能を使用すると、コネクタのイベント・ストア、JMS イベント・ストア、および宛先 JMS キュー間でイベントが失われたり、2 度送信されたりすることがなくなることが、コネクタ・フレームワークによって保証されます。JMS 対応にするためには、コネクタ `DeliveryTransport` 標準プロパティに `JMS` を設定する必要があります。コネクタは、構成が完了すると JMS トランスポートを使用するため、コネクタと統合ブローカーとの後続の通信は、すべてこのトランスポートを介して行われます。JMS トランスポートにより、メッセージは最終的に宛先に確実に配送されます。JMS トランスポートの役割は、トランザクションのキュー・セッションの開始後、コミットが発行されるまでメッセージが確実にキャッシュされるようにすることです。障害が起こったとき、あるいはロールバックが発行されたときには、メッセージは廃棄されます。

注: 保証付きイベント・デリバリー機能を使用しない場合、コネクタがイベントをパブリッシュする時間（コネクタが `getApplEvent()` メソッドを自身の `pollForEvents()` メソッド内部で呼び出す時間）と、コネクタがイベント・レコードを削除することによってイベント・ストアを更新する（または「イベント送付済み」状況を使用して更新する）時間との間のわずかな期間に障害が発生する可能性があります。この期間に障害が発生すると、イベントは送信されますが、イベント・レコードは「進行中」状況でイベント・ストア内に残ります。コネクタは再始動時に、このイベント・ストアに残されたイベント・レコードを検出して送信するので、イベントが 2 回送信されることとなります。

保証付きイベント・デリバリー機能を、JMS イベント・ストアあり、またはなしで、JMS 対応コネクタのために構成することができます。保証付きイベント・デ

リバリーのためにコネクタを構成するには、94 ページの『保証付きイベント・デリバリーの使用可能化』を参照してください。

コネクタ・フレームワークが InterChange Server 統合ブローカーにビジネス・オブジェクトを配信できない場合、このビジネス・オブジェクトは (UnsubscribedQueue や ErrorQueue ではなく) FaultQueue に配置され、状況表示と問題の記述を生成します。FaultQueue メッセージは MQRFH2 フォーマットで書き込まれます。

ビジネス・オブジェクト要求

ビジネス・オブジェクト要求は、InterChange Server が doVerbFor() メソッドにビジネス・オブジェクトを送信するときに処理されます。アダプターは、構成されているデータ・ハンドラーを使用してビジネス・オブジェクトを WebSphere MQ メッセージに変換し、発行します。データ・ハンドラーについての要件を除いては、処理されるビジネス・オブジェクトのタイプに関する要件はありません。

動詞の処理

アダプターは、コラボレーションから渡されたビジネス・オブジェクトを、各ビジネス・オブジェクトの動詞に基づいて処理します。アダプターはビジネス・オブジェクト・ハンドラーと doForVerb() メソッドを使用して、アダプターがサポートするビジネス・オブジェクトを処理します。アダプターは以下のビジネス・オブジェクトの動詞をサポートします。

- Create
- Update
- Delete
- Retrieve

注: Create 動詞、Update 動詞、および Delete 動詞を持つビジネス・オブジェクトは、非同期的にも同期的にも送信できます。デフォルト・モードは非同期送信です。アダプターは、Retrieve 動詞、Exists 動詞、または Retrieve by Content 動詞を持つビジネス・オブジェクトの非同期デリバリーをサポートしません。したがって、Retrieve 動詞、Exists 動詞、または Retrieve by Content 動詞のデフォルト・モードは同期デリバリーです。

Create、Update、および Delete 動詞

Create 動詞、Update 動詞、および Delete 動詞を持つビジネス・オブジェクトの処理は、オブジェクトの発行方式が非同期または同期のいずれかによって異なります。

非同期メッセージ・デリバリー

Create 動詞、Update 動詞、および Delete 動詞を持つビジネス・オブジェクトのデフォルトのデリバリー・モードは、非同期です。メッセージは、データ・ハンドラーの使用により、ビジネス・オブジェクトを基にして作成され、その後出力キューに書き込まれます。メッセージが配信されると、アダプターは BON_SUCCESS、さもなければ BON_FAIL を戻します。

注: アダプターには、メッセージが受信されたかどうか、あるいはアクションが実行されたかどうかを確認する手段はありません。

同期メッセージ・デリバリー

注: この方法では、ビジネス・オブジェクトが InterChange Server から到着したときに FRONT ARENA アプリケーションで実行されるコマンドのカスタマイズが必要です。これらのコマンドは、メッセージから ReplyTo キューを検索し、ResponseTimeout の時間間隔内に応答をキューに配置します。FRONT ARENA のコマンドの作成およびカスタマイズについては、「*Programmer's Guide for SunGard FRONT ARENA, 1.5, 5.4*」を参照してください。

コネクター・プロパティに replyToQueue が定義されていて、ビジネス・オブジェクトの変換プロパティに ResponseTimeout が存在する場合、アダプターは同期モードで要求を発行します。アダプターは、その後、応答を待機して、FRONT ARENA によって適切なアクションが実行されていたことを検証します。

注: MQMD はメッセージ記述子です。MQMD には、メッセージがアプリケーション間で送信されるときにアプリケーション・データに添付される制御情報が格納されます。静的メタオブジェクトまたは動的メタオブジェクト内で、MQMD 属性 OutputFormat の値を指定する必要があります。

アダプターは、次の表に示すようなヘッダー付きのメッセージを最初に送出不します。

表 2. 要求メッセージ記述子ヘッダー (MQMD)

フィールド	説明	値
Format	フォーマットの名前。	変換プロパティに定義されている出力フォーマット。ここでは、IBM 要件を満たすために 8 文字に切り捨てられています (例: MQSTR)。
MessageType Report	メッセージ・タイプ。 必要なレポート・メッセージのオプション。	MQMT_DATAGRAM* 応答メッセージが期待される場合、このフィールドには次のように値が取り込まれます。 処理が成功した場合の肯定処理レポートが必要なことを示す、MQRO_PAN*。 処理が失敗した場合の否定処理レポートが必要なことを示す、MQRO_NAN*。
ReplyToQueue	応答キューの名前。	生成されるレポートの相関 ID が始めに発行された要求のメッセージ ID と等しくなければならないことを示す、MQRO_COPY_MSG_ID_TO_CORREL_ID*。 応答メッセージが期待される場合は、このフィールドにはコネクター・プロパティ ReplyToQueue の値が取り込まれます。
Persistence	メッセージのパーススタンス (永続性)。	MQPER_PERSISTENT*
Expiry	メッセージの存続時間。	MQEI_UNLIMITED*

* は、IBM によって定義されている定数を示します。

表2 に示したメッセージ・ヘッダーの後ろに、メッセージ本文が続きます。メッセージの本文は、データ・ハンドラーを使用して直列化されたビジネス・オブジェクトです。

Report フィールドは、FRONT ARENA からの肯定処理レポートと否定処理レポートの両方の返送が期待されていることを示すよう、設定されます。メッセージを発行したスレッドは、FRONT ARENA が要求を処理できるかどうかを示す応答メッセージを待機します。

FRONT ARENA は、アダプターから同期要求を受信すると、ビジネス・オブジェクトのデータを処理して、次の表で説明するようなレポート・メッセージを送出します。

表3. 応答メッセージ記述子ヘッダー (MQMD)

フィールド	説明	値
Format	フォーマットの名前。	変換プロパティーに定義された busObj の入力フォーマット。
MessageType	メッセージ・タイプ。	MQMT_REPORT*

* は、IBM によって定義されている定数を示します。

表4. 応答メッセージに含まれるデータ

動詞	Feedback フィールド	メッセージの本文
Create、Update、または Delete	SUCCESS VALCHANGE VALDUPES FAIL	(オプション) 変更を反映する、直列化されたビジネス・オブジェクト。 (オプション) エラー・メッセージ。

表5. WebSphere MQ のフィードバック・コードおよび InterChange Server の応答値

WebSphere MQ フィードバック・コード	等価な InterChange Server の応答*
MQFB_NONE (フィードバック・コードが指定されない場合のデフォルトです)	VALCHANGE
MQFB_PANまたは MQFB_APPL_FIRST	SUCCESS
MQFB_NANまたは MQFB_APPL_FIRST + 1	FAIL
MQFB_APPL_FIRST + 2	VALCHANGE
MQFB_APPL_FIRST + 3	VALDUPES
MQFB_APPL_FIRST + 4	MULTIPLE_HITS
MQFB_APPL_FIRST + 5	FAIL_RETRIEVE_BY_CONTENT
MQFB_APPL_FIRST + 6	BO_DOES_NOT_EXIST
MQFB_APPL_FIRST + 7	UNABLE_TO_LOGIN
MQFB_APPL_FIRST + 8	APP_RESPONSE_TIMEOUT (コネクターが即時終了します)
MQFB_NONE	応答メッセージにフィードバック・コードが指定されていない場合にコネクターが受け取る情報

ビジネス・オブジェクトを処理できる場合、アプリケーションは、フィードバック・フィールドが MQFB_PAN (または特定の InterChange Server 値) に設定されたレポート・メッセージを作成します。オプションで、アプリケーションはメッセージ本文に変更箇所を含むビジネス・オブジェクトを直列化して取り込むことができます。ビジネス・オブジェクトを処理できない場合、アプリケーションは、フィードバック・フィールドが MQFB_NAN (または特定の InterChange Server 値) に設定され

たレポート・メッセージを作成し、必要に応じてメッセージ本文にエラー・メッセージを挿入します。いずれの場合も、アプリケーションはメッセージの correlationID フィールドをアダプター・メッセージの messageID に設定し、replyTo フィールドで指定されたキューにメッセージを送信します。

応答メッセージを検索時に、アダプターは、応答メッセージの correlationID を要求メッセージの messageID に突き合わせます。次に、アダプターは要求を発行したスレッドを通知します。アダプターは、応答のフィードバック・フィールドの設定によって、メッセージの本文にビジネス・オブジェクトとエラー・メッセージのどちらが含まれているかを予測します。ビジネス・オブジェクトが予測されていたが、メッセージ本文のデータが取り込まれていない場合、アダプターは、Request 操作のために、本来は InterChange Server によって送出されたのと同じビジネス・オブジェクトをそのまま戻します。エラー・メッセージが含まれていると予測したにもかかわらず、メッセージの本文にエラー・メッセージが取り込まれていなかった場合、InterChange Server には応答コードと汎用エラー・メッセージが返送されません。

カスタムのフィードバック・コード: WebSphere MQ のフィードバック・コードを拡張すると、接続プロパティである FeedbackCodeMappingMO を指定することにより、15 ページの表 5 に示すデフォルトの解釈を変更できます。このプロパティを使用すると、InterChange Server 固有の戻り状況値が WebSphere MQ のフィードバック・コードにマップされるメタオブジェクトを作成できます。

フィードバック・コードに割り当てられた戻り状況値は、InterChange Server に渡されます。詳細については、92 ページの『FeedbackCodeMappingMO』を参照してください。

Retrieve 動詞

Retrieve 動詞を持つビジネス・オブジェクトが同期デリバリーのみをサポートする場合、これらの動詞を持つビジネス・オブジェクトのコネクターによる処理は、Create、Update および Delete に対して定義された同期デリバリーの場合と同様になります。ただし、Retrieve 動詞、Exists 動詞、および Retrieve by Content 動詞を使用する場合は、responseTimeout および replyToQueue が必要です。さらに、Retrieve 動詞および Retrieve by Content 動詞の場合は、メッセージ本文に直列化されたビジネス・オブジェクトを取り込んで、トランザクションを完了する必要があります。

表 6 に、これらの動詞に対応する応答メッセージを示します。

表 6. 応答メッセージに含まれるデータ

動詞	Feedback フィールド	メッセージの本文
Retrieve	FAIL	(オプション) エラー・メッセージ。

データ・ハンドラー・メッセージの処理

Adapter for FRONT ARENA は、ネイティブの FRONT ARENA メッセージに対応できる、新規開発のデータ・ハンドラーを使用します。このデータ・ハンドラーは、メッセージからビジネス・オブジェクトへの変換とその逆方向の変換の両方をサポートします。

メッセージをビジネス・オブジェクトに変換する場合、データ・ハンドラーは、作成されるビジネス・オブジェクトのタイプを特定のメッセージ・フィールドから判別する新規開発のネーム・ハンドラーに依存します。このコネクタ固有のプロパティおよびその他のコネクタ固有のプロパティの詳細については、43ページの『第4章 データ・ハンドラーの構成』を参照してください。データ・ハンドラー開発の詳細については、「データ・ハンドラー・ガイド」を参照してください。

第 2 章 アダプターのインストール

- 『アダプター環境』
- 21 ページの『前提条件となる作業』
- 33 ページの『アダプターに対する WebSphere MQ の使用可能化』
- 33 ページの『アダプターと関連ファイルのインストール』
- 33 ページの『デフォルトのアダプター構成プロパティー』
- 34 ページの『インストールの検証』

本章では、Adapter for FRONT ARENA のインストール方法について説明します。

アダプター環境

アダプターをインストール、構成、使用する前に、環境要件を理解しておく必要があります。

アダプター・コンポーネント

Adapter for FRONT ARENA は、次の付属品と共に出荷されます。

- 「Adapter for SunGard FRONT ARENA ユーザーズ・ガイド」(この資料は、「Adapter for WebSphere MQ ユーザーズ・ガイド」と併せてお読みください)。
- アダプターのデータ・ハンドラー関連メタオブジェクトである
MO_DataHandler_FrontArenaDefaultConfig.xsd、
MO_DataHandler_FrontArenaNativeConfig.xsd、
および MO_FrontArena_DefaultConfig.xsd。
- アダプターによって使用され、JAR ファイル FrontArenaDataHandler.jar として付属する特殊なデータ・ハンドラー。
- FA_trade のアプリケーション固有のビジネス・オブジェクト定義の例 (FA_trade.xsd)。FA_trade 全体のアプリケーション固有の子ビジネス・オブジェクトを含む。

ブローカーとの互換性

アダプターが使用するアダプター・フレームワークは、アダプターと通信する統合ブローカーのバージョンとの互換性を備えている必要があります。Adapter for FRONT ARENA バージョン 1.0 は、以下のアダプター・フレームワークと統合ブローカーでサポートされます。

- **アダプター・フレームワーク:**
WebSphere Business Integration Adapter Framework バージョン 2.3.1、および 2.4
- **統合ブローカー:**
 - WebSphere InterChange Server、バージョン 4.1.1、4.2、4.2.1、4.2.2
 - WebSphere MQ Integrator、バージョン 2.1.0
 - WebSphere MQ Integrator Broker、バージョン 2.1.0
 - WebSphere Business Integration Message Broker、バージョン 5.0

- WebSphere Application Server Enterprise、バージョン 5.0.2
(WebSphere Studio Application Developer Integration Edition、バージョン 5.0.1 と併用)

例外については、「リリース情報」を参照してください。

注: 統合ブローカーおよびその前提条件のインストールに関する説明については、以下のガイドを参照してください。

WebSphere InterChange Server については、「*IBM WebSphere InterChange Server システム・インストール・ガイド (UNIX 版)*」または「*IBM WebSphere InterChange Server システム・インストール・ガイド (Windows 版)*」を参照してください。

WebSphere Message Broker については、「*WebSphere Message Brokers 使用アダプター・インプリメンテーション・ガイド*」を参照してください。

WebSphere Application Server については、「*IBM WebSphere Business Integration Adapters アダプター実装ガイド (WebSphere Application Server)*」を参照してください。

アダプター・プラットフォーム

このアダプターは以下のソフトウェアでサポートされます。

オペレーティング・システム:

以下のアプリケーション・プラットフォームのいずれか:

- AIX 5.1、AIX 5.2
- Solaris 8.0
- HP UX 11.0、HP UX 11i
- Windows 2000

サード・パーティー・ソフトウェア:

- FRONT ARENA バージョン 1.5

アダプターの依存関係

Adapter for FRONT ARENA には、WebSphere Application Server ライブラリーおよび API が必要です。

ロケール依存データ

コネクターは国際化され、2 バイト文字セットをサポートし、特定の言語でメッセージ・テキストを配信できるようになっています。ある文字コードを使用する場所から別の文字コード・セットを使用する場所へデータを転送する場合、コネクターは、そのデータの意味が伝わるように文字変換を実行します。

Java 仮想マシン (JVM) 内部の Java ランタイム環境では、Unicode 文字コード・セットでデータを表現します。Unicode は、既知の文字コード・セットのほとんど (単一バイトおよびマルチバイトの両方) に対応するエンコード方式を含んでいます。

IBM WebSphere Business Integration システムのほとんどのコンポーネントは Java で書かれています。したがって、大部分のインテグレーション・コンポーネント間で行われるデータ転送には、文字変換は必要ありません。

エラー・メッセージや通知メッセージを個々の国や地域に合った適切な言語で記録するには、個々の環境に合わせて Locale 標準構成プロパティを構成する必要があります。構成プロパティの詳細については、127 ページの『付録 A. コネクタの標準構成プロパティ』参照してください。

グローバル化

グローバル化は、標準の WebSphere MQ テクノロジーのアダプターによってサポートされる限り、サポートされます。

前提条件となる作業

プリインストールされている WebSphere MQ アダプターを FRONT ARENA に変換するには、次の手順を実行する必要があります。

必要なタスクは以下のとおりです。

1. 次のアプリケーション固有ビジネス・オブジェクト定義をロードします。
 - FA_trade.xsd
 - MO_DataHandler_FrontArenaDefaultConfig.xsd、
MO_DataHandler_FrontArenaNativeConfig.xsd
(ロード元は MO_DataHandler_FrontArenaDefaultConfig.xsd)、および
MO_FrontArena_DefaultConfig.xsd
2. コネクタの標準プロパティ ApplicationName を、アダプターに割り当てられた名前 (例えば、FrontArenaConnector) に設定します。
3. コネクタのアプリケーション構成プロパティ DataHandlerConfigMO、DataHandlerMIMIType、および ConfigurationMetaObject を、33 ページの『デフォルトのアダプター構成プロパティ』に説明されているように設定します。
4. 手順 1 でインポートしたオブジェクトを、アダプターのサポート対象ビジネス・オブジェクトとして定義します (「エージェント・サポート」をオンにする)。
5. WebSphere Interchange Server のインストール環境にある connectors ディレクトリーの messages サブディレクトリーに、アダプターのメッセージ・ファイルをコピーします。
6. WebSphere Interchange Server のインストール環境にある connectors ディレクトリーの Datahandlers サブディレクトリーに、アダプターのデータ・ハンドラー JAR ファイルをコピーします。

WebSphere MQ キューの構成

アダプターを使用するために必要な WebSphere MQ キューの構成は、FRONT ARENA と IBM WebSphere InterChange Server インストール環境のトポロジーによって一部が異なります。次のトポロジーのいずれかを使用できます。

- マシン 1 台

FRONT ARENA、IBM WebSphere InterChange Server、およびアダプターがすべて同じマシンにインストールされます。

- マシン 2 台と 2 つのキュー・マネージャー

FRONT ARENA が 1 台のマシンにインストールされ、IBM WebSphere InterChange Server とコネクタがもう 1 台のマシンにインストールされます。マシンごとに異なるキュー・マネージャーが使用されます。

- マシン 2 台と 1 つのキュー・マネージャー

FRONT ARENA が 1 台のマシンにインストールされ、IBM WebSphere InterChange Server とコネクタがもう 1 台のマシンにインストールされます。2 台のマシンのキューを管理するために、同じキュー・マネージャーを使用します。

マシン 1 台のトポロジー

このトポロジーでは、FRONT ARENA、InterChange Server、および Adapter for FRONT ARENA は、1 台のマシンにインストールされます。このソリューションで使用されるすべての WebSphere MQ キューは、1 つのキュー・マネージャーによって処理されます。InterChange Server をインストールしたときにセットアップしたキュー・マネージャーを使用することをお勧めします。

このトポロジーでは、次の役割を果たすキューが必要です。

- インバウンド・キュー

FRONT ARENA では、このキューが存在することが必要です。ただし、アダプターは、このソリューションではこのキューを使用しません。

- 並列インバウンド・キュー

FRONT ARENA では、このキューが存在することが必要です。ただし、アダプターは、このソリューションではこのキューを使用しません。

- 直列インバウンド・キュー

これは、InterChange Server から FRONT ARENA に送信されたメッセージを受信するためのキューです。

- アウトバウンド・キュー

これは、FRONT ARENA から InterChange Server へメッセージを送信するためのキューです。

- 処理実行中キュー

FRONT ARENA から InterChange Server に送信された有効なメッセージの原版は、アダプターが処理を完了するまでここに格納されます。処理が完了すると、元のメッセージはローカル・アーカイブ・キューに移動します。

- Archive_Queue

アダプターによってメッセージが完全に処理され、FRONT ARENA から InterChange Server に送信されると、メッセージの原版はここに格納されます。

- Unsubscribe_Queue

メッセージは正常に解析されたが、アダプターによってサポートされているどのビジネス・オブジェクトにも対応しない場合、そのメッセージはここに格納されます。

- ICS_Error_Queue

メッセージは、ビジネス・オブジェクトに正常に変換されずに InterChange Server に送信されると、ここに格納されます。

- WCS_Error_Queue

FRONT ARENA が正常に処理できなかったメッセージを格納します。

- ReplyToQueue

同期データ交換に対してセットアップされた構成内でのみ使用されます。

前述のすべてのキューは、マシン 1 台のトポロジーではローカルです。キューを手動で作成する場合は、キューに付ける名前を選択します。このソリューションに付属のバッチ・ファイルを使用する場合 (後述) は、バッチ・ファイルによって事前に割り当てられた名前を持つキューが作成されます。

アダプターを Windows 環境で使用する場合は、バッチ・ファイルを使用して、マシン 1 台のトポロジーに適したキューを生成できます。バッチ・ファイルは、IBM InterChange Server インストール環境で使用されるルート・ディレクトリー内の `¥Connector¥WebSphereCommerce¥Utilities` サブディレクトリーに置かれている製品パッケージによってインストールされます。バッチ・ファイルを使用してキューを作成するには、次に示すように、`ConfigureWebSphereCommerceAdapter.bat` というファイルを実行します。

コマンド・プロンプトで、次のコマンドを入力します。

```
ConfigureWebSphereCommerceAdapter  
<InterChangeServerName>.queue.manager
```

ここで、`<InterChangeServerName>` は、使用している WebSphere InterChange Server の名前です。

この結果、`InterChange ServerName.queue.manager` という名前のキュー・マネージャーが作成され、このキュー・マネージャーによって WebSphere MQ キューが作成されます。作成されるキューの名前は、バッチ・ファイルによって作成された場合と同様に、次の表で確認できます。

表 7. WebSphere MQ キュー

WebSphere MQ キュー名	説明
WC_MQCONN.IN_PROGRESS	アダプターの処理実行中キュー。
WC_MQCONN.ERROR	アダプターの InterChange Server エラー・キュー。
WC_MQCONN.ARCHIVE	アダプターのアーカイブ・キュー。
WC_MQCONN.REPLY	アダプターの応答先キュー。
MC_MQCONN.UNSUBSCRIBED	アダプターのアンサブスクライブ・キュー。
WCS_Serial_Inbound	FRONT ARENA の直列インバウンド・キュー。28 ページの『WebSphere Application Server の JMS 設定の構成』で説明するように、このキュー名は、FRONT ARENA に対して定義されている JMS キュー名と一致する必要があります。

表7. WebSphere MQ キュー (続き)

WebSphere MQ キュー名	説明
WCS_Outbound	FRONT ARENA のアウトバウンド・キュー。28 ページの『WebSphere Application Server の JMS 設定の構成』で説明するように、このキュー名は、FRONT ARENA に対して定義されている JMS キュー名と一致する必要があります。
WCS_Parallel_Inbound	FRONT ARENA のインバウンド並列キュー。28 ページの『WebSphere Application Server の JMS 設定の構成』で説明するように、このキュー名は、FRONT ARENA に対して定義されている JMS キュー名と一致する必要があります。
WCS_Error	FRONT ARENA のエラー・キュー。28 ページの『WebSphere Application Server の JMS 設定の構成』で説明するように、このキュー名は、FRONT ARENA に対して定義されている JMS キュー名と一致する必要があります。
WCS_Inbound	FRONT ARENA のインバウンド・キュー。28 ページの『WebSphere Application Server の JMS 設定の構成』で説明するように、このキュー名は、FRONT ARENA に対して定義されている JMS キュー名と一致する必要があります。

2 台のマシン、2 つのキュー・マネージャーのトポロジー

このトポロジーでは、FRONT ARENA が 1 台のマシンにインストールされ、IBM WebSphere InterChange Server と Adapter for FRONT ARENA がもう 1 台のマシンにインストールされます。

WebSphere MQ はマシンごとにインストールする必要があり、各インストール環境では、異なるキュー・マネージャーが使用されます。これらは、マシンごとに作成するキューです。

25 ページの表 8 では、キュー名は各キューの役割を示しますが、これらのキュー名が、現在の FRONT ARENA システムで使用されている JMS キュー名と同期する限り、異なるキュー名を設定できます。

この表で、WCS というプレフィックスは、FRONT ARENA システムのインストール先マシンに作成されているキューと、そのマシンに存在するキュー・マネージャーによって管理されるキューを示します。ICS というプレフィックスは、InterChange Server およびコネクタのインストール先マシンに作成されているキューと、そのマシンに存在するキュー・マネージャーによって管理されるキューを示します。

表 8. FRONT ARENA および InterChange Server のキュー名

FRONT ARENA マシン上のキュー	InterChange Server マシン上のキュー
<p>WCS_Outbound キュー</p> <p>FRONT ARENA から InterChange Server へメッセージを送信するためのキュー。このキューは、リモート・キューとして InterChange Server マシンの ICS_Inbound を指すリモート・キュー定義として作成されます。</p>	<p>ICS_Inbound キュー</p> <p>FRONT ARENA から InterChange Server に送信されたメッセージを受信するためのキュー。</p>
<p>WCS_Serial Inbound キュー</p> <p>InterChange Server から FRONT ARENA に送信されたメッセージを受信するためのキュー。</p>	<p>ICS_Outbound キュー</p> <p>InterChange Server から FRONT ARENA にメッセージを送信するためのキュー。このキューは、リモート・キューとして FRONT ARENA マシンの WCS_Serial Inbound を指すリモート・キュー定義として作成されます。</p>
<p>宛先が InterChange Server の場合</p> <p>FRONT ARENA から InterChange Server への送信キュー。</p> <p>WCS_Error_queue</p> <p>FRONT ARENA が正常に処理できなかったメッセージを格納します。</p> <p>WCS_Parallel Inbound</p>	<p>宛先が FRONT ARENA の場合</p> <p>InterChange Server から FRONT ARENA システムへの送信キュー。</p> <p>ICS_Error_queue</p> <p>メッセージは、ビジネス・オブジェクトに正常に変換されなかった場合、ここに格納されます。アダプターは、並列インバウンド・キューを使用しません。</p> <p>ICS_InProgress キュー</p> <p>FRONT ARENA から InterChange Server に送信された有効なメッセージの原版は、アダプターが処理を完了するまでここに格納されます。処理が完了すると、元のメッセージはローカル・アーカイブ・キューに移動します。</p> <p>ICS_Archive_queue</p> <p>アダプターによってメッセージが完全に処理され、FRONT ARENA から InterChange Server に送信されると、メッセージの原版はここに格納されます。</p>

2 つのシステム間の通信を使用可能にするには、チャンネルおよび送信キューを使用します。

次の表に示す役割を果たすチャンネルは、このトポロジーの各マシン上で作成する必要があります。

表9. チャネルの役割

FRONT ARENA マシン上のチャネル	InterChange Server マシン上のチャネル
Sender_WCS	Sender_ICS
Receiver_ICS	Receiver_WCS

チャネルの作成: 次の手順の例では、特定のサーバー名およびキュー・マネージャー名を指定することによって、これらの名前が、さまざまなマシンやキューと関連を取ることができます。チャネルは、キューが互いに正しいキューを参照していることを確認するために使用します。実際の情報を保持するには、キューのローカル版を使用します。

FRONT ARENA マシンの構成作業:

注: ここで使用するチャネル名は、例にすぎません。

1. WebSphere MQ Explorer を使用して、FRONT ARENA システムに 2 つのチャネルを作成します。一方は WCS という名前の送信側チャネルで、もう一方は ICS という名前の受信側チャネルです。
2. ローカル・キューを作成します。例えば、ToICSSystem という名前を使用します。
3. ToICSSystem キューを送信キューとして設定します。
4. 次のプロパティを WCS_Outbound キューに設定します。
 - a. リモート・キュー名 ICS_Inbound。リモート・キュー・マネージャー名 ICS_server_name.queue.manager。例えば、ICS.queue.manager です。
 - b. 伝送キュー名のプロパティを、手順 2 で作成したように ToICSSystem に設定します。
5. 送信側のチャネルを構成するには、次の手順を実行します。
 - a. IP アドレスとポート番号を使用して接続名を指定します。例えば、9.182.12.235(1414) とします。ここで、9.182.12.235 は InterChange Server が稼働しているマシンの IP アドレスで、1414 はデフォルトのリスナー・ポート番号です。
 - b. 送信キュー名を ToICSSystem と指定します。

これで、FRONT ARENA マシンの構成作業は完了です。

InterChange Server マシンの構成作業:

1. WebSphere MQ Explorer を使用して、2 つのチャネルを作成します。一方は ICS という名前の送信側チャネルで、もう一方は WCS という名前の受信側チャネルです。

注: WebSphere Business Integration システムの送信側チャネルの名前は、FRONT ARENA の受信側チャネルの名前と同一にする必要があります。WebSphere Business Integration システムの受信側チャネルの名前は、FRONT ARENA の送信側チャネルの名前と同一にする必要があります。

2. 新規のローカル・キュー、例えば ToICSSystem を作成します。ToICSSystem キューを送信キューとして設定します。

3. WebSphere Business Integration システムにリモート定義キューを作成します。このリモート定義キューは、出力キューとしてコネクタ・コンポーネントに使用する必要があります。以下のプロパティを設定します。
 - a. リモート・キュー名 WCS_SerialInbound
 - b. リモート・キュー・マネージャー名 <wcssystems_Q_manager_name>。例えば、QM_wcsfvt3 とします。
 - c. 送信キュー名のプロパティを ToICSSystem に設定します。
4. 送信側のチャンネルを構成するには、次の手順を実行します。
 - a. IP アドレスとポート番号を使用して接続名を指定します。例えば、9.182.12.18(1414) とします。ここで、9.182.12.18 は FRONT ARENA が稼働しているマシンの IP アドレスで、1414 はデフォルトのリスナー・ポート番号です。
 - b. 送信キュー名を ToICSSystem と指定します。
5. FRONT ARENA マシンと InterChange Server マシンの両方の WebSphere MQ キューおよびチャンネルを構成したら、次の手順を実行します。
 - a. 受信側チャンネルを開始します。
 - b. 次に、送信側チャンネルを開始します。

これで、InterChange Server マシンの構成作業は完了です。

2 台のマシン、1 つのキュー・マネージャーのトポロジー

このトポロジーでは、FRONT ARENA が 1 台のマシンにインストールされ、InterChange Server と Adapter for FRONT ARENA がもう 1 台のマシンにインストールされます。動作する WebSphere MQ のインスタンスは 1 つのみであり、2 台のマシンが使用するキューは、1 つのキュー・マネージャーによって管理されます。このシナリオでは、ローカル・キューのみを使用します。

このトポロジーでは、次の役割を果たすキューが必要です。

- インバウンド・キュー

FRONT ARENA では、このキューが存在することが必要です。ただし、アダプターは、このソリューションではこのキューを使用しません。

- 直列インバウンド・キュー

このキューは、InterChange Server から FRONT ARENA にメッセージを送信するためのキューです。

- アウトバウンド・キュー

このキューは、FRONT ARENA から InterChange Server に送信されたメッセージを受信するためのキューです。

- 進行中のキュー

FRONT ARENA から InterChange Server に送信された有効なメッセージの原版は、アダプターが処理を完了するまでここに格納されます。処理が完了すると、元のメッセージはローカル・アーカイブ・キューに移動します。

- Archive_Queue

アダプターによってメッセージが完全に処理され、FRONT ARENA から InterChange Server に送信されると、メッセージの原版はここに格納されます。

- Unsubscribed_Queue

メッセージは正常に解析されたが、アダプターによってサポートされているどのビジネス・オブジェクトにも対応しない場合、そのメッセージはここに格納されます。

- FRONT ARENA エラー・キュー
- ICS_Error_Queue

メッセージは、ビジネス・オブジェクトに正常に変換されずに InterChange Server に送信されると、ここに格納されます。

- WCS_Error_Queue

このキューには、FRONT ARENA が正常に処理できなかったメッセージが格納されます。

- ReplyToQueue

このキューは、同期データ交換に対してセットアップされた構成内でのみ使用されます。

WebSphere Application Server の JMS 設定の構成

WebSphere Application Server を構成して Java Messaging Service Connection Factory および JMS キューを作成し、WebSphere MQ と連動するようするには、次の手順を実行します。

1. コマンド・プロンプトで、次の手順を実行します。

- a. 次のコマンドを 1 行に入力して、クラスパス変数を更新します。

```
set classpath= %classpath%;
MQ_install_path%java%lib%com.ibm.mqjms.jar;
MQ_install_path%java%lib%com.ibm.mq.jar;
MQ_install_path%java%lib%com.ibm.mq.iopp.jar;
MQ_install_path%java%lib%com.ibm.ibmorb.jar;WAS_install_path%lib%ns.jar
```

MQ_install_path WebSphere MQ のインストール先のパス

WAS_install_path WebSphere Application Server のインストール先のパス

- b. 次のコマンドを入力して、MQ_JAVA_INSTALL_PATH という新しい環境変数を追加します。

```
set MQ_JAVA_INSTALL_PATH=MQ_install_path%java
```

MQ_install_path WebSphere MQ のインストール先のパス

- c. 次のコマンドを入力して、WebSphere Application Server に付属の JDK (Java Development Kit) を使用するために環境を更新します。

```
set PATH = WAS_Intall_Path%Java%bin;%PATH%
```

WAS_install_path WebSphere Application Server のインストール先のパス

2. WebSphere Application Server が稼働していることと、前述の手順 1 で定義した正しいクラスパスおよび環境変数が追加されていることを確認します。さらに、

使用する JDK は WebSphere Application Server に付属の JDK であることを `java -version` を実行して確認し、`WAS_Install_Path\Java\bin` にある JDK のバージョンを確認します。

3. `MQ_install_path\java\bin` ディレクトリーで、`JMSAdmin.config` ファイルを開き、次の値を設定します。

```
INITIAL_CONTEXT_FACTORY=com.ibm.ejs.ns.jndi.CNInitialContextFactory
PROVIDER_URL=iiop://localhost:900
```

この値では、FRONT ARENA と WebSphere MQ が同じマシンにインストールされていることを前提にしています。

```
SECURITY_AUTHENTICATION=none
```

4. コマンド行の入力として `JMSAdmin.config` ファイルを次のように指定して、`MJSAdmin` プログラムを実行します。

```
CommandPrompt:> JMSAdmin -cfg JMSAdmin.config -t -v
```

このコマンドを実行すると、WebSphere Application Server に用意されている JNDI (Java Naming and Directory Interface) サービスを検索できます。InitCtx> というプロンプトが表示されます。これを使用すると、JMS 管理コマンドを実行できます。

5. 次のコマンドを入力して、`QueueConnectionFactory` を登録し、コード化文字セット ID を設定します。

- `define qcf (JMS_QueueConnectionFactory) qmanager (Your_QueueManager_Name)`
- `alter qcf (JMS_QueueConnectionFactory) ccsid(1208)`

ここで、`JMS_QueueConnectionFactory` は、`MQQueueConnectionFactory` JMS オブジェクトの名前です。

これらの一連のコマンドを実行すると、このキュー接続ファクトリーの項目が、`BINDINGBEANTBL` テーブルの下の WebSphere Application Server データベースに作成されます。これらのオブジェクトは、WebSphere Application Server データベースに登録されます。

6. 次の `JMSQueues` を定義して、WebSphere MQ キューや、使用している WebSphere MQ キュー・マネージャーに対して設定した名前にマップします。`JMSQueues` 名は、各管理者独自の要件に応じてカスタマイズできますが、これらの名前の定義先となる WebSphere MQ 名は、大文字と小文字の違いを含めて、WebSphere MQ に設定したキュー名と完全に一致する必要があります。マシン 1 台のトポロジーに対して適切な WebSphere MQ キューを作成するために、このアダプターに付属のバッチ・ファイルを使用する場合は、後述の表で説明するように、生成されたこれらの WebSphere MQ キュー名を、JMS キューの定義先にする値として使用するようになしてください。

JMS キューは、次の構文で定義されます。

- 注:** 2 台のマシン、2 つのキュー・マネージャーのトポロジーの場合のように、アウトバウンド・キューに対してリモート・キュー定義を使用している場合は、

JMS_Outbound_Queue をローカルの WebSphere MQ キューに定義しないでください。リモート・キュー定義を使用している場合、アウトバウンド・キューの構文は次のとおりです。

```
define q(JMS_Outbound_Queue)qmanager(Your_Queue_Manager_Name)

define q(JMS_Outbound_Queue)qmanager
  (Your_Queue_Manager_Name)
  queue (Your_Outbound_QueueName)

define q(JMS_Inbound_Queue)qmanager
  (Your_Queue_Manager_Name)
  queue (Your_Inbound_QueueName)

define q(JMS_Parallel_Inbound_Queue)qmanager
  (Your_Queue_Manager_Name)queue
  (Your_Parallel_Inbound_Queue_Name)

define q(JMS_Serial_Inbound_Queue)qmanager
  (Your_Queue_Manager_Name)queue
  (Your_Serial_Inbound_Queue_Name)

define q(JMS_Error_Queue)qmanager
  (Your_Queue_Manager_Name)
  queue (Your_Error_Queue_Name)
```

表 10. JMS キュー名の定義

<i>Your_Outbound_QueueName</i>	アウトバウンド・キューに対して作成された WebSphere MQ キュー。デフォルトでは、これは、FRONT ARENA からのメッセージを受け取って InterChange Server に渡すためにアダプターがポーリングするキューになります。バッチ・ファイルによって作成されたデフォルトの WebSphere MQ キュー・セットアップの場合、この値は WCS_Outbound になります。
<i>Your_Serial_Inbound_Queue</i>	直列インバウンド・キューに対して作成された WebSphere MQ キュー。これは、InterChange Server から FRONT ARENA へ送信されたメッセージを Adapter for FRONT ARENA が格納する格納先キューです。バッチ・ファイルによって作成されたデフォルトの WebSphere MQ キュー・セットアップの場合、この値は WCS_Serial_Inbound になります。
<i>Your_Parallel_Inbound_Queue_Name</i>	これは、並列インバウンド・キューに対して作成された WebSphere MQ キューです。
<i>Your_Error_Queue_Name</i>	エラー・キューに対して作成された WebSphere MQ キュー。これは、Adapter for FRONT ARENA がメッセージにエラーを検出したときに、メッセージを送信する場所です。バッチ・ファイルによって作成されたデフォルトの WebSphere MQ キュー・セットアップの場合、この値は WCS_Error になります。

表 10. JMS キュー名の定義 (続き)

<i>Your_Queue_Manager_Name</i>	<p>FRONT ARENA システムのセットアップで WebSphere MQ キューを処理するキュー・マネージャーの名前。バッチ・ファイル <code>ConfigureAdapterQueues.bat</code> で作成したセットアップなど、通常のマシン 1 台のセットアップでは、FRONT ARENA システムのキューも、ICS に設定したキュー・マネージャーを使用して管理します。このようなセットアップの場合、デフォルトは <code><InterchangeServerName>.queue.manager</code> になります。</p>
--------------------------------	---

キューを作成したら、JMSAdmin コンソールを使用して、アウトバウンド・キューやエラー・キューに対して次のプロパティーを設定します。この手順では、JMS がネイティブの WebSphere MQ アプリケーションを処理することが指定されます。

- `alter q(JMSOutboundQueue) targclient(MQ)`
- `alter q(JMSErrorQueue) targclient(MQ)`

`end` と入力して、JMSAdmin ツールを終了します。これで、Java Messaging Service と、FRONT ARENA を実行している WebSphere Application Server とを構成するための作業は完了です。

FRONT ARENA 内部での JMS ConnectionSpec の構成

注: JMSAdmin 名および JMS 接続ファクトリーは、インスタンス XML ファイルで、Commerce Configuration Manager の `connectionSpec` セクションに入力した値と同じにする必要があります。詳細については、FRONT ARENA Configuration Manager の `Transports` セクションの下を参照してください。後述の手順も参照してください。

FRONT ARENA Administration Console を開始します。Site Administrator としてログインし、「構成 (Configuration)」セクションに移動して、「トランスポート (Transport)」オプションを選択します。トランスポートとして「WebSphere MQ」を選択し、状況をアクティブに変更します。Administration Console からログアウトします。

FRONT ARENA ソリューションでは、「*FRONT ARENA Installation Guide*」に説明されているように、「ストア」を作成して使用することが必要です。『Publishing a Sample Store』のセクションに説明されているようにストアのパブリッシュを完了したら、今度は Store Administrator として Administration Console にログインし、使用するストアを選択します。「構成 (Configuration)」セクションで、ストアに MQ Transport を追加します。このための項目が STORETRANS テーブルに作成されます。

メッセージング・システムのトランスポート・アダプターを使用可能にするには、FRONT ARENA Configuration Manager を起動して、次の手順を事項します。

1. 「ホスト名」->「インスタンス」の順に選択します。
2. 「コンポーネント」フォルダーを開きます。

3. TransportAdapter を選択します。
4. 「コンポーネントを使用可能にする (Enable Component)」チェック・ボックスを選択します。
5. 「適用」をクリックします。

このインスタンスの connectionSpec に使用している値を使用して、JMSQueue 名と JMS Connection Factory を次のように構成します。

1. 「ホスト名」->「インスタンス」の順に選択します。
2. 「トランスポート (Transport)」を選択します。
3. 「アウトバウンド」->「JMS」の順に展開します。
4. ConnectionSpec を選択します。
5. WebSphere Application Server に JMS 設定を構成するときに作成された ConnectionFactory 名を入力します。
6. これまでに作成した、Inbound、Error、および Outbound のキュー名を入力します。
7. 「適用」をクリックします。
8. 「インバウンド」->「JMSInbound CCF Connector」-「シリアル (Serial)」の順に展開します。
9. ConnectionSpec を選択します。
10. ConnectionFactory Name、SerialInbound、Error および Output JMS キューを入力します。
11. 「適用」をクリックします。
12. 「インバウンド」-「JMSInbound CCF Connector」-「並列 (Parallel)」を展開します。
13. ConnectionSpec を選択します。
14. ConnectionFactory、ParallelInbound、Error および Output JMS キューを入力します。
15. 「適用」をクリックします。

Configuration Manager を終了します。

FRONT ARENA JVM 設定の更新

WebSphere Application Server クラスのパスをインスタンスに対して更新し、追加の JAR ファイル項目を追加する必要があります。これを実行するには、WebSphere Application Server Advanced Administrative Console を開き、次の手順を実行します。

1. FRONT ARENA インスタンスを実行しているホストを選択します。
2. 「WebSphere 管理可能ドメイン (WebSphere Administrative Domain)」を選択します。
3. 「ノード (Nodes)」を選択します。
4. ホスト名を選択します。
5. 「アプリケーション・サーバー (Application Servers)」を選択します。

6. 「FRONT ARENA Server instance_name」を選択します。ここで、instance_name は、FRONT ARENA インスタンスの名前です。
7. このインスタンスの JVM 設定に移動します。
8. 「新規システム・プロパティの追加 (Add a new system property)」を選択します。
9. 次のシステム・プロパティを入力します。

```
name= ws.ext.dirs value=MQ_INSTALL_PATH/java/lib
```

ここで、MQ_INSTALL_PATH は、WebSphere WebSphere MQ のインストール先のパスです。

10. すべての変更内容を有効にするために、WebSphere Application Server サービスを再始動します。

アダプターに対する WebSphere MQ の使用可能化

Adapter for FRONT ARENA は、カスタマイズされた WebSphere MQ テクノロジーのアダプターです。WebSphere MQ アダプターのインスタンスが Adapter for FRONT ARENA のベースとしてインストールされていることが前提になります。これには、アダプターが使用するすべての WebSphere MQ キューの定義も含まれます。

WebSphere MQ アダプターのインストール方法については、対応する資料を参照してください。これは、WebSphere Interchange Server による配信の一部になっています。アダプターには、FrontArenaConnector のように記述名を付けることをお勧めします。

アダプターと関連ファイルのインストール

WebSphere Business Integration Adapter 製品のインストールについては、「*WebSphere Business Integration Adapters インストール・ガイド*」を参照してください。この資料は、次の Web サイトの WebSphere Business Integration Adapters Infocenter にあります。

<http://www.ibm.com/websphere/integration/wbiadapters/infocenter>

デフォルトのアダプター構成プロパティ

Adapter for FRONT ARENA は、WebSphere MQ テクノロジーのアダプターの変形として、そのベース・アダプターと同じコネクタ構成プロパティを持っています。FRONT ARENA の MQ アダプターに対して特定の関連を持つプロパティのみを次に示します。

表 11. 標準のプロパティ

プロパティ名	説明	デフォルト値
ApplicationName	コネクタの名前。	FrontArenaConnector
MessageFileName	メッセージ・ファイルはコネクタに使用されません。	FrontArenaConnector.txt (アダプターに付属しているメッセージ・ファイルの名前です。)

表 12. コネクタ固有のプロパティ

プロパティ名	説明	デフォルト値
DataHandlerConfigMO	データ・ハンドラーが使用するメタオブジェクト。	MO_DataHandler_FrontArena Config (このデータ・ハンドラー・メタオブジェクトは、アダプターに付属しています。)
DataHandlerMimeType	MIME タイプに基づくデータ・ハンドラーの選択。	FA/native
ConfigurationMetaObject	デフォルトの構成メタオブジェクト。	MO_FrontArena_DefaultConfig (この構成メタオブジェクトは、アダプターに付属しています。)

表 13. サポートされるビジネス・オブジェクト

ビジネス・オブジェクト名	説明
MO_DataHandler_FrontArenaDefaultConfig	デフォルトのデータ・ハンドラー・メタオブジェクト。
MO_DataHandler_FrontArenaNativeConfig	ネイティブの FRONT ARENA メッセージ形式に対応するデータ・ハンドラー・メタオブジェクト。
MO_FrontArena_DefaultConfig FA_trade	デフォルトの構成メタオブジェクト。 アプリケーション固有のビジネス・オブジェクトの例。

注: サポートされているすべてのビジネス・オブジェクトについて、エージェントのサポートをオンにする必要があります。

インストールの検証

以下のセクションでは、インストール後の製品のパスとファイル名について説明します。

注: Windows 環境でも UNIX 環境でも、一般に、WebSphere MQ と JMS は異なるディレクトリーにインストールされています。例えば AIX システムの場合、デフォルトでは、WebSphere MQ は /var/mqm/ にインストールされ、JMS は /usr/mqm/java/lib にインストールされています。JMS のインストールを /var/mqm/java/lib にリダイレクトすることにより、ルーチン /usr に関連するシステム管理タスクによる削除を防止することができます。同様に Windows でも、通常、WebSphere MQ は %Program Files%\WebSphere MQ の下にインストールされ、JMS は %Program Files%\IBM\MQSeries\Java の下にインストールされます。これに従って、WebSphere MQ コネクタ始動スクリプト内のクラスパスを変更してください。

Windows のファイル構造

インストーラーは、コネクタに関連付けられた標準ファイルをご使用のシステムにコピーします。

ユーティリティーにより、コネクタが `ProductDir¥connectors¥WebSphereMQConnector` ディレクトリーにインストールされ、コネクタへのショートカットが「スタート」メニューに追加されます。

次の表に、コネクタが使用する Windows ファイル構造の説明と、インストーラーからコネクタのインストールを選択した場合に自動的にインストールされるファイルを示します。

<i>ProductDir</i> のサブディレクトリー	説明
<code>WICS¥connectors¥WebSphereMQ¥CWWebSphereMQ.jar</code>	WebSphere MQ コネクタによってのみ使用されるクラスを含む。
<code>WICS¥connectors¥WebSphereMQConnector¥start_WebSphereMQ.bat</code>	コネクタ (NT/2000) の始動スクリプト。
<code>WICS¥connectors¥messages¥WebSphereMQConnector.txt</code>	コネクタのメッセージ・ファイル。
<code>WICS¥repository¥WebSphereMQ¥CN_WebSphereMQ.txt</code>	コネクタのリポジトリー定義。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥WebSphereMQConnector.cfg</code>	WebSphere MQ 構成ファイルの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥PortConnector.cfg</code>	ポートのコネクタ構成ファイルの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥Sample_WebSphereMQ_LegacyContact.xsd</code>	スキーマの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥Sample_WebSphereMQ_MO_Config.xsd</code>	メタオブジェクトの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥Sample_WebSphereMQ_MO_DataHandler.xsd</code>	データ・ハンドラー・メタオブジェクトの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥Sample_WebSphereMQ_MO_DataHandler_DelimitedConfig.xsd</code>	区切り文字で区切られているデータ・ハンドラー・メタオブジェクトの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥Sample_WebSphereMQ_DynMO_Config.xsd</code>	動的メタオブジェクトの例。
<code>WICS¥connectors¥WebSphereMQ¥samples¥LegacyContact¥JMSPropertyPairs.xsd</code>	JMS プロパティーの例。
<code>WICS¥DataHandlers¥FrontArenaDataHandler.jar</code>	データ・ハンドラー。
<code>WICS¥repository¥FrontArena¥MO_DataHandler_FrontArenaDefaultConfig.xsd</code>	データ・ハンドラー・メタオブジェクト。
<code>WICS¥repository¥FrontArena¥MO_DataHandler_FrontArenaNativeConfig.xsd</code>	データ・ハンドラー・メタオブジェクト。
<code>WICS¥repository¥FrontArena¥MO_FrontArenaDefaultConfig.xsd</code>	アダプター構成メタオブジェクト。
<code>WICS¥repository¥FA_trade.xsd</code>	アプリケーション固有のビジネス・オブジェクトの例。
<code>WICS¥repository¥FA_instrument.xsd</code>	アプリケーション固有のビジネス・オブジェクトの例。
<code>WICS¥bin¥Data¥App¥FrontArenaConnectorTemplate</code>	構成テンプレート。

表 14. FRONT ARENA ファイルのブリッジ

<i>ProductDir</i> のサブディレクトリー	説明
<code>WICS¥other¥FrontArena¥bia_bfa.exe</code>	ブリッジの実行可能ファイル。
<code>WICS¥other¥FrontArena¥bia_bfaMessages.txt</code>	ブリッジのメッセージ・ファイル。
<code>WICS¥other¥FrontArena¥bia_BFAAlert.jar</code>	SMTP E メール・アラート・メカニズムに対応するブリッジの JAR ファイル。
<code>WICS¥other¥FrontArena¥config.sample</code>	ブリッジの構成ファイルの例。

注: すべての製品パス名は、システム上の、製品のインストール先ディレクトリーを基準とした相対パス名です。ブリッジの場合は、別のマシンにインストールされる可能性が高くなります。

UNIX のファイル構造

インストーラーは、コネクタに関連付けられた標準ファイルをご使用のシステムにコピーします。

ユーティリティーにより、コネクターが *ProductDir/connectors/WebSphereMQConnector* ディレクトリーにインストールされます。

次の表に、コネクターが使用する UNIX ファイル構造の説明と、インストーラーからコネクターのインストールを選択した場合に自動的にインストールされるファイルを示します。

<i>ProductDir</i> のサブディレクトリー	説明
connectors/WebSphereMQ/CWWebSphere MQ.jar	WebSphere MQ コネクターによってのみ使用されるクラスを含む。
connectors/WebSphereMQ/start_WebSphereMQ.sh	コネクターのシステム始動スクリプト。このスクリプトは、汎用のコネクター・マネージャー・スクリプトから呼び出されます。 System Manager の「コネクター構成」画面をクリックすると、このコネクター・マネージャー・スクリプトのカスタマイズ済みラッパーがインストーラーによって作成されます。コネクターの始動および停止には、このカスタマイズされたラッパーを使用してください。
connectors/messages/WebSphereMQ/Connector.txt	コネクターのメッセージ・ファイル。
repository/WebSphereMQ/CN_WebSphere MQ.txt	コネクターのリポジトリ定義。
connectors/WebSphereMQ/samples/LegacyContact/WebSphereMQConnector.cfg	WebSphere MQ 構成ファイルの例。
connectors/WebSphereMQ/samples/LegacyContact/PortConnector.cfg	ポートのコネクター構成ファイルの例。
connectors/WebSphereMQ/samples/LegacyContact/Sample_WebSphereMQ_LegacyContact.xsd	スキーマの例。
connectors/WebSphereMQ/samples/LegacyContact/Sample_WebSphereMQ_MO_Config.xsd	メタオブジェクトの例。
connectors/WebSphereMQ/samples/LegacyContact/Sample_WebSphereMQ_MO_DataHandler.xsd	データ・ハンドラー・メタオブジェクトの例。
connectors/WebSphereMQ/samples/LegacyContact/Sample_WebSphereMQ_MO_DataHandler_DelimitedConfig.xsd	区切り文字で区切られているデータ・ハンドラー・メタオブジェクトの例。
connectors/WebSphereMQ/samples/LegacyContact/Sample_WebSphereMQ_DynMO_Config.xsd	動的メタオブジェクトの例。
connectors/WebSphereMQ/samples/LegacyContact/JMSPropertyPairs.xsd	JMS プロパティの例。
WICS/DataHandlers/FrontArenaDataHandler.jar	データ・ハンドラー。
WICS/repository/FrontArena/MO_DataHandler_FrontArenaDefaultConfig.xsd	データ・ハンドラー・メタオブジェクト。
WICS/repository/FrontArena/MO_FrontArena_DefaultConfig.xsd	アダプター構成メタオブジェクト。
WICS/repository/FA_trade.xsd	アプリケーション固有のビジネス・オブジェクトの例。
WICS/bin/Data/App/FrontArenaConnectorTemplate	コネクターのテンプレート。
WICS/other/FrontArena/bia_bfa	ブリッジの実行可能ファイル。
WICS/other/FrontArena/bia_bfaMessages.txt	ブリッジのメッセージ・ファイル。
WICS/other/FrontArena/bia_BFAAlert.jar	SMTP E メール・アラート・メカニズムに対応するブリッジの JAR ファイル。
WICS/other/FrontArena/config.sample	ブリッジの構成ファイルの例。
WICS/other/FrontArena/start_bfa.sh	ブリッジの始動スクリプト。

表 15. FRONT ARENA ファイルのブリッジ

<i>ProductDir</i> のサブディレクトリー	説明
WICS/other/FrontArena/bia_bfa	ブリッジの実行可能ファイル。
WICS/other/FrontArena/bia_bfaMessages.txt	ブリッジのメッセージ・ファイル。
WICS/other/FrontArena/bia_BFAAlert.jar	SMTP E メール・アラート・メカニズムに対応するブリッジの JAR ファイル。

表 15. FRONT ARENA ファイルのブリッジ (続き)

ProductDir のサブディレクトリー	説明
WICS/other/FrontArena/config.sample	ブリッジの構成ファイルの例。
WICS/other/FrontArena/start_bfa.sh	ブリッジの始動スクリプト。

注: すべての製品パス名は、システム上の、製品のインストール先ディレクトリーを基準とした相対パス名です。ブリッジの場合は、別のマシンにインストールされる可能性が高くなります。

第 3 章 FRONT ARENA のビジネス・オブジェクト

- 『ビジネス・オブジェクト・メタデータ』
- 『ビジネス・オブジェクト構造』
- 40 ページの『カスタム・ビジネス・オブジェクト・ハンドラー』

この章では、アダプターによるビジネス・オブジェクトの処理方法と、コネクタの前提事項について説明します。この情報は、新規ビジネス・オブジェクトをインプリメントする際のガイドとして利用できます。

アダプターには、ビジネス・オブジェクトの例のみが付属しています。固有のビジネス・オブジェクトを作成する必要があります。ビジネス・オブジェクト定義は、FRONT ARENA データベースに格納されている FRONT ARENA ビジネス・オブジェクトと一致する必要があります。ビジネス・オブジェクト定義は、Business Object Designer を使用して手動で作成することも、JDBCODA を使用して自動的に作成することもできます。

ビジネス・オブジェクト・メタデータ

アダプターはメタデータ主導型です。このコンテキストでは、メタデータはアプリケーションに関するデータであり、ビジネス・オブジェクト定義に格納され、アダプターとアプリケーション間の対話を支援します。メタデータ主導型のコネクタによって処理される各ビジネス・オブジェクトは、コネクタ内にハードコーディングされた命令ベースではなく、ビジネス・オブジェクト定義内にエンコードされたメタデータをベースにしてサポートされています。

ビジネス・オブジェクトのメタデータには、ビジネス・オブジェクトの構造、属性プロパティの設定、およびアプリケーション固有テキストの内容が含まれています。コネクタは、メタデータ主導型であるため、アダプターのコードを修正せずに、新規または変更済みのビジネス・オブジェクトを処理できます。アダプターに構成されたデータ・ハンドラーは、コネクタのビジネス・オブジェクトの構造、オブジェクトのカーディナリティー、アプリケーション固有テキストのフォーマット、およびビジネス・オブジェクトのデータベース表記について、ある条件を前提として動作します。したがって、アダプターのビジネス・オブジェクトを作成または変更する場合、変更の内容はアダプターが従うよう定められている規則に適合している必要があります。適合していない場合、アダプターは新規のビジネス・オブジェクトや変更されたビジネス・オブジェクトを正しく処理できません。

ビジネス・オブジェクト構造

Adapter for FRONT ARENA は、キューから WebSphere MQ メッセージを取得して、メッセージ内に格納されているビジネス・データを、(メタオブジェクトによって定義された) ビジネス・オブジェクトに取り込もうとします。

アダプターによって交換された MQ Series メッセージ内のビジネス・データは、XML 文書内に格納されます。アダプターは、FRONT ARENA データ・ハンドラー

を使用して、FRONT ARENA ネイティブ・メッセージからのデータをビジネス・オブジェクトに変換し、ビジネス・オブジェクトから FRONT ARENA ネイティブ・メッセージに変換します。

ビジネス・オブジェクト定義の構造は、FRONT ARENA ビジネス・オブジェクトの要件に適合する必要があります。JDBCODA の使用方法や、FRONT ARENA データベースからビジネス・オブジェクト定義を生成する方法については、「*WebSphere Business Integration Adapters インストール・ガイド*」および「*Adapter for JDBC ユーザーズ・ガイド*」を参照してください。

FRONT ARENA の通常の Trade ビジネス・オブジェクト定義で使用するプロパティや構造を参照するには、ファイル `/DataHandlers/repository/FrontArena/FA_trade.xsd` を開きます。

カスタム・ビジネス・オブジェクト・ハンドラー

アダプターにいくつかの基本的な変換プロパティを指定するには、静的メタオブジェクトを使用します。

表 16. 静的構成メタオブジェクト、*MO_FrontArena_DefaultConfig*

プロパティ名	アプリケーション固有情報
デフォルト	<code>OutputQueue=queue://DENALI/FRONTARENA.OUT?</code> <code>targetClient=1;ResponseTimeout=60000;</code>
FA_trade_Create	<code>InputFormat=INSERT;</code>
FA_trade_Update	<code>InputFormat=UPDATE;</code>
FA_trade_Delete	<code>InputFormat=DELETE;</code>
FA_trade_Retrieve	<code>InputFormat=SELECT;</code>

静的メタオブジェクトは、ビジネス・オブジェクトごとに定義された変換プロパティのリストで構成されています。ビジネス・オブジェクトの変換プロパティを定義するには、ストリング属性を作成し、`<ビジネス・オブジェクト名>_<動詞>` という構文を使用して、このストリング属性に名前を付けます。例えば、FA_trade オブジェクトと動詞 Create の組み合わせに対して変換プロパティを定義するには、FA_trade_Create という属性を作成します。

属性のアプリケーション固有情報には、実際の変換プロパティを指定できます。さらに、Default という名前の予約済み属性名を、メタオブジェクトに定義することもできます。この属性があると、そのプロパティはすべてのビジネス・オブジェクトの変換プロパティのデフォルト値として使用されます。

前述したように、この例のメタオブジェクトでは、ビジネス・オブジェクト FA_trade およびサポートされているすべての動詞の変換プロパティが定義されます。サポートの対象となる追加のビジネス・オブジェクトと動詞の組み合わせについては、適切な項目を追加する必要があります。

Default プロパティには、サービス要求をアダプターのブリッジ・コンポーネントに送信するときに使用する WebSphere MQ キューの名前である OutputQueue が含まれます。キュー名は、カスタマーのセットアップに適合している必要があります。

ただし、どのような場合のキュー名にも、targetClient=1 というキュー名を追加する必要があります。これにより、アダプターのブリッジ・コンポーネントに送信されたメッセージは、確実に FRONT ARENA アプリケーションが予想するフォーマットになります。

注: アダプターのブリッジ・コンポーネントは、MQMD ヘッダーとメッセージ本文のみで構成されるネイティブの WebSphere MQ メッセージを予想します。このパラメーターを指定しなかった場合は、アダプターによって JMS スタイルのメッセージが作成されます。このメッセージには、追加の MQRFH2 ヘッダーがあります。

パラメーター ResponseTimeout は、サービス要求に対する応答を待つ場合にタイムアウトになるまでの時間を、ミリ秒単位で示します。このプロパティーが未定義のままか、またはゼロよりも小さい値に設定されている場合、コネクタは応答を待機せず、SUCCESS を即時に戻します。サービス要求を同期方式で処理する場合は、このパラメーターを設定する必要があります。

サポートの対象となる追加のビジネス・オブジェクトと動詞の組み合わせごとに、対応する新規のプロパティーを追加する必要があります。パラメーター InputFormat のみを指定する必要があります。このパラメーターは、FRONT ARENA の動詞 INSERT、UPDATE、DELETE、および SELECT をマップするときに使用します。これらの動詞は、ブリッジから WBI の動詞 Create、Update、Delete、および Select に送信されたメッセージの FORMAT フィールドに格納されます。

これらのパラメーターの詳細については、「*Adapter for WebSphere MQ ユーザーズ・ガイド*」の資料を参照してください。

第 4 章 データ・ハンドラーの構成

- 『FRONT ARENA メッセージとデータ・ハンドラー』
- 46 ページの『データ・ハンドラー・メタオブジェクトの構成』
- 49 ページの『データ・ハンドラーのイベント通知サポート』

この章では、データ・ハンドラーを構成して、メッセージを処理および変換する方法について説明します。

FRONT ARENA メッセージとデータ・ハンドラー

FRONT ARENA データ・ハンドラーは、アダプターと FRONT ARENA アプリケーションを組み合わせる際に必要です。データ・ハンドラーを構成する方法は 2 つあります。

- コネクタ固有のプロパティ `DataHandlerClassName` 内で、データ・ハンドラー・クラス名を指定します。詳細については、90 ページの『コネクタ固有のプロパティ』を参照してください。
- MIME タイプを指定し、さらに、この MIME タイプの構成をコネクタ固有のプロパティ `DataHandlerMimeType` および `DataHandlerConfigMO` のそれぞれで定義するデータ・ハンドラー・メタオブジェクトを指定します。詳細については、表 38 および「データ・ハンドラー・ガイド」を参照してください。

データ・ハンドラーの処理方法を理解するには、FRONT ARENA メッセージの構造を理解することが不可欠です。これらのメッセージには、次の特性があります。

- メッセージは、基本的なエレメントまたはネストされたリストを含むリストです。
- リストの先頭と末尾は、メッセージ・エレメント `<list>` および `</list>` で示されます。`<list>` は、リストの名前を表します。リスト名は任意であり、固有である必要はありません。
- 単純なリスト・エレメントは、「キーワード = 値」というフォーマットになっています。
- トップレベルのリストの名前は、必ず MESSAGE になります。リストには、次の 5 つのエレメントがあります。
 - `TYPE=<作成/変更/削除/検索したオブジェクトの名前と、このオブジェクトに適用される操作の名前>`
 - `VERSION=<FRONT ARENA AMB のバージョン>`
 - `TIME=<メッセージ作成のタイム・スタンプ>`
 - `SOURCE=<メッセージ・オリジネーターの名前>`
 - 作成または変更したオブジェクトを表すリスト
- 従属リストは、その親リストの末尾に置く必要があります。従属リストの後に単純なリスト・エレメントを置くことはできません。ただし、子リストをいくつか置くことはできます。

メッセージ全体は、テキスト形式になっています。メッセージの論理形式とは関係なく、単純なエレメントのすべての値は、ストリングで表現されます。リスト・タグとキーワードは、必ず大文字になります。

次の未完成メッセージは、新規の取引の作成を示しています。

```
[MESSAGE]
TYPE=INSERT_TRADE
VERSION=1.0
TIME=2003-09-15 18:21:36
SOURCE=ATLAS
[TRADE]
STATUS=F0 confirmed
TRADER_USERNBR.USERID=INTAS
QUANTITY=200
PRICE=89.57
TIME=2003-09-15 12:05:41
[INSTRUMENT]
ISIN=FR123456780
EXPIRY_DATE=2001-05-12
[/INSTRUMENT]
[/TRADE]
[/MESSAGE]
```

基本オブジェクトとリスト TRADE の他に、このメッセージには INSTRUMENT というネストされたリストが含まれています。

FRONT ARENA メッセージからビジネス・オブジェクトへの変換

ある特定の FRONT ARENA メッセージの場合、ネーム・ハンドラーは、データ・ハンドラーのメタオブジェクトの BOPrefix 属性の値、および MESSAGE の (単一の) 子リストの名前を小文字に変換したものを基にして、関連のビジネス・オブジェクトの名前を構成します。

注: 名前の 2 つの構成要素は、下線で分離されます。

例えば、メタオブジェクト属性 BOPrefix に FA という値があることを前提とすると、ビジネス・オブジェクト FA_trade は、前記のメッセージを基にして作成されます。

動詞は、静的構成メタオブジェクトに指定されている変換プロパティに従って、メッセージの FORMAT フィールドの内容を基にして導出されます。

注: アダプターにいくつかの基本的な変換プロパティを指定するには、静的メタオブジェクトを使用します。

データ・ハンドラーは、FRONT ARENA メッセージのリストのエレメントごとに、対応するビジネス・オブジェクト属性を検出しようとします。ビジネス・オブジェクト属性とリスト・エレメントとを関連付けるには、ビジネス・オブジェクト属性名とエレメント名 (単純属性のキーワード) を同一にするか、またはビジネス・オブジェクト属性のアプリケーション固有情報によって、この属性に対応する FRONT ARENA メッセージ・エレメントを識別する必要があります。このような明示的なマッピングは、ビジネス・オブジェクト属性のアプリケーション固有の情報に CN=<FRONTARENA element name> を指定することによって実行されます。

注: <FRONT ARENA element name> は、大/小文字を区別しません。CN は大文字にする必要があります。

ネストされたリストに対応するビジネス・オブジェクト属性は、子ビジネス・オブジェクトになることが想定されています。そうでない場合は、エラーが通知されます。つまり、FRONT ARENA メッセージのリスト階層は、ビジネス・オブジェクト定義に反映される必要があります。

FRONT ARENA メッセージの構造と関連のビジネス・オブジェクト定義の構造が完全に同一である必要はありません。ビジネス・オブジェクト定義に反映されていない FRONT ARENA メッセージの要素や部分は、スキップされるだけです。この処理は、単純属性と従属リスト (のツリー) の両方に適用されます。アダプターのログ・ファイルには、スキップされたメッセージ・要素ごとに通知項目が追加されます。

元のメッセージに示されていないビジネス・オブジェクト属性は、定数値 CxIgnore に設定されます。

ビジネス・オブジェクトから FRONT ARENA メッセージへの変換

ビジネス・オブジェクトが実際に処理される前に、トップレベルの MESSAGE リスト (ビジネス・オブジェクトには反映されていないリスト) およびその単純要素は、次のように作成されます。

- TYPE 要素は、ビジネス・オブジェクト動詞のアプリケーション固有情報のパラメーター FAVERB の値と、ビジネス・オブジェクトのアプリケーション固有情報のパラメーター TN の値で構成され、この 2 つの値が下線で区切られています。TYPE の値は大文字にします。
- VERSION は、定数値 1.0 に設定されます。
- TIME は現在の日時に設定されます。
- SOURCE は、データ・ハンドラー・メタオブジェクトの SourceName 属性の値に設定されます。

ビジネス・オブジェクトは、順次処理されます。単純属性ごとに、キーワード=値のペアが追加されます。このペアは、アプリケーション固有情報のパラメーター CN の値または属性の名前 (パラメーターが指定されていない場合) と属性の値で構成されます。

子ビジネス・オブジェクトが検出されると、子リストが開きます。このリストの名前は、属性のアプリケーション固有情報のパラメーター CN の値になります。または、このパラメーターが指定されていない場合は属性の名前になります。ビジネス・オブジェクトのすべての属性が処理されると、関連リストの終了タグがメッセージに追加されます。

メッセージ・ビジネス・オブジェクトの例

前述のメッセージ例のビジネス・オブジェクトの内容は、次のようになります。

表 17. ビジネス・オブジェクト FA_trade

一般オブジェクト・レベルの アプリケーション固有情報	TN=trade
動詞名	アプリケーション固有情報

表 17. ビジネス・オブジェクト FA_trade (続き)

一般オブジェクト・レベルの アプリケーション固有情報	TN=trade
Create	FAVERB=insert
Delete	FAVERB=delete
Retrieve	FAVERB=select
Update	FAVERB=update
属性名	アプリケーション固有情報
trader	CN=trader_usrnbr.userid
quantity	CN=quantity
price	CN=price
instrument (子ビジネス・オブジェクト)	CN=instrument

表 18. 子ビジネス・オブジェクト FA_instrument

一般オブジェクト・レベルの アプリケーション固有情報	TN=trade
動詞名	アプリケーション固有情報
Create	FAVERB=insert
Delete	FAVERB=delete
Retrieve	FAVERB=select
Update	FAVERB=update
属性名	アプリケーション固有情報
isin	
expirationdate	FAVERB=insert

ビジネス・オブジェクト定義には、TRADE エレメントである STATUS および TIME に対応する属性が存在しません。前述したように、ビジネス・オブジェクトには、対応する FRONT ARENA メッセージのすべての属性が格納されている必要はありません。

子ビジネス・オブジェクト FA_instrument の属性 isin の場合は、アプリケーション固有情報のマッピングは必要ありません。これは、属性名と対応する FRONT ARENA メッセージ・エレメントの名前が同一であるためです。

データ・ハンドラー・メタオブジェクトの構成

データ・ハンドラーは、次の 2 つのメタオブジェクトによって構成されます。

- MO_DataHandler_FrontArenaDefaultConfig
- MO_DataHandler_FrontArenaNativeConfig

表 19. データ・ハンドラー・メタオブジェクト *MO_DataHandler_FrontArenaDefaultConfig*

プロパティ名	説明	指定可能な値	デフォルト値
FA_native	汎用のトップレベル・データ・ハンドラー・メタオブジェクト。現在サポートしているのは、MIME タイプが FA/native の場合のみです。	MO_DataHandler_FrontArenaNativeConfig (後述する子メタオブジェクトを指します)。	N/A

表 20. データ・ハンドラー・メタオブジェクト *MO_DataHandler_FrontArenaNativeConfig* (MIME タイプが *FA/native* の場合)

プロパティ名	説明	指定可能な値	デフォルト値
ClassName	データ・ハンドラーのインプリメンテーションを格納しているクラス。	com.ibm.adapters.datahandlers.frontarenaFANativeDataHandler	com.ibm. adapters. datahandlers.frontarena.FANativeDataHandler
NameHandler Class	ネーム・ハンドラーのインプリメンテーションを格納しているクラス。	com.ibm.adapters.datahandlers.frontarena.FANativeNameHandler	com. ibm. adapters. datahandlers.frontarena.FANativeNameHandler
BOPrefix	FRONT ARENA メッセージの要素を区切り、終了する区切り文字。	任意の一連の文字。	FA
Attribute Delimiter	FRONT ARENA メッセージの単純な要素に使用されるキーワードと値の分離文字。	FRONT ARENA メッセージの要素を区切る一連の文字。 注: FRONT ARENA バージョン 1.x および 2.0 では、分離文字として改行文字を使用します。このプロパティの値を変更する必要があるのは、将来の FRONT ARENA のバージョンで別の分離文字が使用される場合のみです。	¥n (改行)

表 20. データ・ハンドラー・メタオブジェクト *MO_DataHandler_FrontArenaNativeConfig* (MIME タイプが *FA/native* の場合) (続き)

プロパティ			
名	説明	指定可能な値	デフォルト値
KeyValue Separator	トップレベルのリストである MESSAGE の単純なリスト・エレメント SOURCE の値。	FRONT ARENA メッセージの単純なエレメントのキーワードと値の間で使用される分離文字。 注: FRONT ARENA バージョン 1.x および 2.0 では、分離文字として等号を使用します。このプロパティの値を変更する必要があるのは、将来の FRONT ARENA のバージョンで別の分離文字が使用される場合のみです。	=
SourceName DefaultVerb	デフォルトの動詞 デフォルトの動詞	任意の一連の文字。 Create、Update、Delete、または Retrieve。	WBI Create
CxIgnore	ビジネス・オブジェクトから変換する場合、データ・ハンドラーは、CxIgnore 定数をその値として保持するビジネス・オブジェクトを検出すると、この属性をリスト・エレメントに割り当てます。ビジネス・オブジェクトに変換する場合、この属性の値がデータ・ハンドラーの値として関連のメッセージ・エレメントに存在すると、データ・ハンドラーは CxIgnore 定数をビジネス・オブジェクト属性に割り当てます。	任意の一連の文字。	CxIgnore

表 20. データ・ハンドラー・メタオブジェクト *MO_DataHandler_FrontArenaNativeConfig* (MIME タイプが *FA/native* の場合) (続き)

プロパティ			
名	説明	指定可能な値	デフォルト値
SkipCxIgnore	この属性を true に設定すると、データ・ハンドラーは、CxIgnore という値を持つすべてのビジネス・オブジェクト属性/リスト・エレメントを無視します。	true または false。	false

通知のない要求の送信

アダプターを構成して、通知なしで要求を送信する方法 (デフォルトの非同期モード。「応答不要送信」とも呼ばれる) は、次のとおりです。

- 送信する要求を表し、データ・ハンドラーと互換性のあるビジネス・オブジェクトを作成します。
- 静的メタオブジェクトまたは動的メタオブジェクトを使用して、宛先のキューとフォーマットを指定します。静的メタオブジェクトと動的メタオブジェクトの詳細については、101 ページの『メタオブジェクトの構成』と 109 ページの『動的子メタオブジェクトの作成の概要』を参照してください。
- (静的または動的) メタオブジェクト内のプロパティ `ResponseTimeout` を `-1` に設定します。この設定では、コネクタは発行したビジネス・オブジェクトの戻りをチェックしません。

データ・ハンドラーのイベント通知サポート

FRONT ARENA アプリケーションは、適宜にセットアップされると、新規オブジェクトが作成されるか、既存のオブジェクトが変更または削除されるたびに、その AMB ミドルウェアを介してメッセージをパブリッシュします。アダプターのプリッジ・コンポーネントは、これらのすべてのメッセージにサブスクライブします。受信された各イベント・メッセージは、WebSphere MQ キューに転送されます。

アダプターによるそれらのイベントの処理を可能にするには、(アダプターのコネクタ固有プロパティ内で) アダプターを構成して、その `InputQueue` として上記のキューを使用するようにする必要があります。

注: イベントが失われないようにするため、このキューは永続的キューとしてセットアップします。

第 5 章 コネクタの構成

- 『Connector Configurator の概要』
- 52 ページの『Connector Configurator の始動』
- 53 ページの『コネクタ固有のプロパティ・テンプレートの作成』
- 56 ページの『新しい構成ファイルを作成』
- 57 ページの『既存ファイルの使用』
- 58 ページの『構成ファイルの完成』
- 59 ページの『構成ファイル・プロパティの設定』
- 67 ページの『構成の完了』
- 67 ページの『グローバル化環境における Connector Configurator の使用』

Connector Configurator の概要

Connector Configurator では、次の統合ブローカーで使用するアダプターのコネクタ・コンポーネントを構成できます。

- WebSphere InterChange Server (ICS)
- WebSphere MQ Integrator、WebSphere MQ Integrator Broker、および WebSphere Business Integration Message Broker (WebSphere Message Brokers (WMQI) と総称)
- WebSphere Application Server (WAS)

Connector Configurator を使用して次の作業を行います。

- コネクタを構成するための**コネクタ固有のプロパティ・テンプレート**を作成する。
- **コネクタ構成ファイル**を作成します。インストールするコネクタごとに 1 つ構成ファイルを作成する必要があります。
- 構成ファイル内のプロパティを設定する。
場合によっては、コネクタ・テンプレートでプロパティに対して設定されているデフォルト値を変更する必要があります。また、サポートされるビジネス・オブジェクト定義と、ICS の場合はコラボレーションとともに使用するマップを指定し、必要に応じてメッセージング、ロギング、トレース、およびデータ・ハンドラー・パラメーターを指定する必要があります。

Connector Configurator の実行モードと使用する構成ファイルのタイプは、実行する統合ブローカーによって異なります。例えば、使用している統合ブローカーが WMQI の場合、Connector Configurator を System Manager から実行するのではなく、直接実行します (52 ページの『スタンドアロン・モードでの Configurator の実行』を参照)。

コネクタ構成プロパティには、標準の構成プロパティ (すべてのコネクタがもつプロパティ) と、コネクタ固有のプロパティ (特定のアプリケーションまたはテクノロジーのためにコネクタに必要なプロパティ) とが含まれます。

標準プロパティはすべてのコネクタにより使用されるので、標準プロパティを新規に定義する必要はありません。ファイルを作成すると、Connector Configurator により標準プロパティがこの構成ファイルに挿入されます。ただし、Connector Configurator で各標準プロパティの値を設定する必要があります。

標準プロパティの範囲は、ブローカーと構成によって異なる可能性があります。特定のプロパティに特定の値が設定されている場合にのみ使用できるプロパティがあります。Connector Configurator の「標準のプロパティ」ウィンドウには、特定の構成で設定可能なプロパティが表示されます。

ただしコネクタ固有プロパティの場合は、最初にプロパティを定義し、その値を設定する必要があります。このため、特定のアダプターのコネクタ固有プロパティのテンプレートを作成します。システム内ですでにテンプレートが作成されている場合には、作成されているテンプレートを使用します。システム内でまだテンプレートが作成されていない場合には、53 ページの『新規テンプレートの作成』のステップに従い、テンプレートを新規に作成します。

注: Connector Configurator は、Windows 環境内でのみ実行されます。UNIX 環境でコネクタを実行する場合には、Windows で Connector Configurator を使用して構成ファイルを変更し、このファイルを UNIX 環境へコピーします。

Connector Configurator の始動

以下の 2 種類のモードで Connector Configurator を開始および実行できます。

- スタンドアロン・モードで個別に実行
- System Manager から

スタンドアロン・モードでの Configurator の実行

どのブローカーを実行している場合にも、Connector Configurator を個別に実行し、コネクタ構成ファイルを編集できます。

これを行うには、以下のステップを実行します。

- 「スタート」>「プログラム」から、「IBM WebSphere InterChange Server」>「IBM WebSphere Business Integration Tools」>「Connector Configurator」をクリックします。
- 「ファイル」>「新規」>「コネクタ構成」を選択します。
- 「システム接続: Integration Broker」の隣のプルダウン・メニューをクリックします。使用しているブローカーに応じて、ICS、WebSphere Message Brokers、または WAS を選択します。

Connector Configurator を個別に実行して構成ファイルを生成してから、System Manager に接続してこの構成ファイルを System Manager プロジェクトに保存することもできます (58 ページの『構成ファイルの完成』を参照)。

System Manager からの Configurator の実行

System Manager から Connector Configurator を実行できます。

Connector Configurator を実行するには、以下のステップを実行します。

1. System Manager を開きます。
2. 「System Manager」ウィンドウで、「統合コンポーネント・ライブラリー」アイコンを展開し、「コネクタ」を強調表示します。
3. System Manager メニュー・バーから、「ツール」>「**Connector Configurator**」をクリックします。「Connector Configurator」ウィンドウが開き、「**新規コネクタ**」ダイアログ・ボックスが表示されます。
4. 「システム接続: **Integration Broker**」の隣のプルダウン・メニューをクリックします。使用しているブローカーに応じて、ICS、WebSphere Message Brokers、または WAS を選択します。

既存の構成ファイルを編集するには、以下のステップを実行します。

- 「System Manager」ウィンドウの「コネクタ」フォルダーでいずれかの構成ファイルを選択し、右クリックします。Connector Configurator が開き、この構成ファイルの統合ブローカー・タイプおよびファイル名が上部に表示されます。
- Connector Configurator で「ファイル」>「開く」を選択します。プロジェクトまたはプロジェクトが保管されているディレクトリーからコネクタ構成ファイルを選択します。
- 「標準のプロパティ」タブをクリックし、この構成ファイルに含まれているプロパティを確認します。

コネクタ固有のプロパティ・テンプレートの作成

コネクタの構成ファイルを作成するには、コネクタ固有プロパティのテンプレートとシステム提供の標準プロパティが必要です。

コネクタ固有プロパティのテンプレートを新規に作成するか、または既存のコネクタ定義をテンプレートとして使用します。

- テンプレートの新規作成については、53 ページの『新規テンプレートの作成』を参照してください。
- 既存のファイルを使用する場合には、既存のテンプレートを変更し、新しい名前でのこのテンプレートを保管します。既存のテンプレートは `¥WebSphereAdapters¥bin¥Data¥App` ディレクトリーにあります。

新規テンプレートの作成

このセクションでは、テンプレートでプロパティを作成し、プロパティの一般特性および値を定義し、プロパティ間の依存関係を指定する方法について説明します。次にそのテンプレートを保管し、新規コネクタ構成ファイルを作成するためのベースとして使用します。

Connector Configurator でテンプレートを作成するには、以下のステップを実行します。

1. 「ファイル」>「新規」>「コネクタ固有プロパティ・テンプレート」をクリックします。
2. 「コネクタ固有プロパティ・テンプレート」 ダイアログ・ボックスが表示されます。
 - 「新規テンプレート名を入力してください」の下の「名前」フィールドに、新規テンプレートの名前を入力します。テンプレートから新規構成ファイルを作成するためのダイアログ・ボックスを開くと、この名前が再度表示されます。
 - テンプレートに含まれているコネクタ固有のプロパティ定義を調べるには、「テンプレート名」表示でそのテンプレートの名前を選択します。そのテンプレートに含まれているプロパティ定義のリストが「テンプレートのプレビュー」表示に表示されます。
3. テンプレートを作成するときには、ご使用のコネクタに必要なプロパティ定義に類似したプロパティ定義が含まれている既存のテンプレートを使用できます。ご使用のコネクタで使用するコネクタ固有のプロパティが表示されるテンプレートが見つからない場合は、自分で作成する必要があります。
 - 既存のテンプレートを変更する場合には、「変更する既存のテンプレートを選択してください: 検索テンプレート」の下の「テンプレート名」テーブルのリストから、テンプレート名を選択します。
 - このテーブルには、現在使用可能なすべてのテンプレートの名前が表示されます。テンプレートを検索することもできます。

一般特性の指定

「次へ」をクリックしてテンプレートを選択すると、「プロパティ: コネクタ固有プロパティ・テンプレート」ダイアログ・ボックスが表示されます。このダイアログ・ボックスには、定義済みプロパティの「一般」特性のタブと「値」の制限のタブがあります。「一般」表示には以下のフィールドがあります。

- **一般:**
プロパティ・タイプ
更新されたメソッド
説明
- **フラグ**
標準フラグ
- **カスタム・フラグ**
フラグ

プロパティの一般特性の選択を終えたら、「値」タブをクリックします。

値の指定

「値」タブを使用すると、プロパティの最大長、最大複数値、デフォルト値、または値の範囲を設定できます。編集可能な値も許可されます。これを行うには、以下のステップを実行します。

1. 「値」タブをクリックします。「一般」のパネルに代わって「値」の表示パネルが表示されます。
2. 「プロパティを編集」表示でプロパティの名前を選択します。
3. 「最大長」および「最大複数値」のフィールドに値を入力します。

新規プロパティ値を作成するには、以下のステップを実行します。

1. 「プロパティを編集」リストでプロパティを選択し、右マウス・ボタンでクリックします。
2. ダイアログ・ボックスから「追加」を選択します。
3. 新規プロパティ値の名前を入力し、「OK」をクリックします。右側の「値」パネルに値が表示されます。

「値」パネルには、3つの列からなるテーブルが表示されます。

「値」の列には、「プロパティ値」ダイアログ・ボックスで入力した値と、以前に作成した値が表示されます。

「デフォルト値」の列では、値のいずれかをデフォルトとして指定することができます。

「値の範囲」の列には、「プロパティ値」ダイアログ・ボックスで入力した範囲が表示されます。

値が作成されて、グリッドに表示されると、そのテーブルの表示内から編集できるようになります。

テーブルにある既存の値の変更を行うには、その行の行番号をクリックして行全体を選択します。次に「値」フィールドを右マウス・ボタンでクリックし、「値の編集 (Edit Value)」をクリックします。

依存関係の設定

「一般」タブと「値」タブで変更を行ったら、「次へ」をクリックします。「依存関係: コネクター固有プロパティ・テンプレート」ダイアログ・ボックスが表示されます。

依存プロパティは、別のプロパティの値が特定の条件に合致する場合にのみ、テンプレートに組み込まれて、構成ファイルで使用されるプロパティです。例えば、テンプレートに PollQuantity が表示されるのは、トランスポート機構が JMS であり、DuplicateEventElimination が True に設定されている場合のみです。プロパティを依存プロパティとして指定し、依存する条件を設定するには、以下のステップを実行します。

1. 「使用可能なプロパティ」表示で、依存プロパティとして指定するプロパティを選択します。
2. 「プロパティを選択」フィールドで、ドロップダウン・メニューを使用して、条件値を持たせるプロパティを選択します。
3. 「条件演算子」フィールドで以下のいずれかを選択します。

== (等しい)

!= (等しくない)

> (より大)

< (より小)

>= (より大か等しい)

<= (より小か等しい)

4. 「条件値」フィールドで、依存プロパティをテンプレートに組み込むために必要な値を入力します。
5. 「使用可能なプロパティ」表示で依存プロパティを強調表示させて矢印をクリックし、「依存プロパティ」表示に移動させます。
6. 「完了」をクリックします。Connector Configurator により、XML 文書として入力した情報が、Connector Configurator がインストールされている %bin ディレクトリーの %data¥app の下に保管されます。

新しい構成ファイルを作成

構成ファイルを新規に作成するには、構成ファイルの名前を指定し、統合ブローカーを選択する必要があります。

- 「System Manager」ウィンドウで「コネクタ」フォルダーを右クリックし、「新規コネクタの作成」を選択します。Connector Configurator が開き、「新規コネクタ」ダイアログ・ボックスが表示されます。
- スタンドアロン・モードの場合は、Connector Configurator で「ファイル」>「新規」>「コネクタ構成」を選択します。「新規コネクタ」ウィンドウで、新規コネクタの名前を入力します。

また、統合ブローカーも選択する必要があります。選択したブローカーによって、構成ファイルに記述されるプロパティが決まります。ブローカーを選択するには、以下のステップを実行します。

- 「Integration Broker」フィールドで、ICS 接続、WebSphere Message Brokers 接続、WAS 接続のいずれかを選択します。
- この章で後述する説明に従って「新規コネクタ」ウィンドウの残りのフィールドに入力します。

コネクタ固有のテンプレートからの構成ファイルの作成

コネクタ固有のテンプレートを作成すると、テンプレートを使用して構成ファイルを作成できます。

1. 「ファイル」>「新規」>「コネクタ構成」をクリックします。
2. 以下のフィールドを含む「新規コネクタ」ダイアログ・ボックス表示されます。

- **名前**

コネクタの名前を入力します。名前では大文字と小文字が区別されます。入力する名前は、システムにインストールされているコネクタのファイル名に対応した一意の名前でなければなりません。

重要: Connector Configurator では、入力された名前のスペルはチェックされません。名前が正しいことを確認してください。

- **システム接続**

ICS 接続、WebSphere Message Brokers 接続、WAS のいずれかをクリックします。

- 「コネクタ固有プロパティ・テンプレート」を選択します。

ご使用のコネクタ用に設計したテンプレートの名前を入力します。「テンプレート名」表示に、使用可能なテンプレートが表示されます。「テンプレート名」表示で名前を選択すると、「プロパティ・テンプレートのプレビュー」表示に、そのテンプレートで定義されているコネクタ固有のプロパティが表示されます。

使用するテンプレートを選択し、「OK」をクリックします。

3. 構成しているコネクタの構成画面が表示されます。タイトル・バーに統合ブローカーとコネクタの名前が表示されます。ここですべてのフィールドに値を入力して定義を完了するか、ファイルを保管して後でフィールドに値を入力するかを選択できます。
4. ファイルを保管するには、「ファイル」>「保管」>「ファイルに」をクリックするか、「ファイル」>「保管」>「プロジェクトに」をクリックします。プロジェクトに保管するには、System Manager が実行中でなければなりません。ファイルとして保管する場合は、「ファイル・コネクタを保管」ダイアログ・ボックスが表示されます。*.cfg をファイル・タイプとして選択し、「ファイル名」フィールド内に名前が正しいスペル (大文字と小文字の区別を含む) で表示されていることを確認してから、ファイルを保管するディレクトリにナビゲートし、「保管」をクリックします。Connector Configurator のメッセージ・パネルの状況表示に、構成ファイルが正常に作成されたことが示されます。

重要: ここで設定するディレクトリ・パスおよび名前は、コネクタの始動ファイルで指定するコネクタ構成ファイルのパスおよび名前に一致する必要があります。

5. この章で後述する手順に従って、「Connector Configurator」ウィンドウの各タブにあるフィールドに値を入力し、コネクタ定義を完了します。

既存ファイルの使用

使用可能な既存ファイルは、以下の 1 つまたは複数の形式になります。

- コネクタ定義ファイル。
コネクタ定義ファイルは、特定のコネクタのプロパティと、適用可能なデフォルト値がリストされたテキスト・ファイルです。コネクタの配布パッケージの `¥repository` ディレクトリ内には、このようなファイルが格納されていることがあります (通常、このファイルの拡張子は .txt です。例えば、XML コネクタの場合は CN_XML.txt です)。
- ICS リポジトリ・ファイル。
コネクタの以前の ICS 実装で使用した定義は、そのコネクタの構成で使用されたリポジトリ・ファイルで使用可能になります。そのようなファイルの拡張子は、通常 .in または .out です。
- コネクタの以前の構成ファイル。
これらのファイルの拡張子は、通常 *.cfg です。

これらのいずれのファイル・ソースにも、コネクタのコネクタ固有プロパティのほとんど、あるいはすべてが含まれますが、この章内の後で説明するように、コネクタ構成ファイルは、ファイルを開いて、プロパティを設定しない限り完成しません。

既存ファイルを使用してコネクタを構成するには、Connector Configurator でそのファイルを開き、構成を修正し、そのファイルを再度保管する必要があります。

以下のステップを実行して、ディレクトリーから *.txt、*.cfg、または *.in ファイルを開きます。

1. Connector Configurator 内で、「ファイル」>「開く」>「ファイルから」をクリックします。
2. 「ファイル・コネクタを開く」ダイアログ・ボックス内で、以下のいずれかのファイル・タイプを選択して、使用可能なファイルを調べます。
 - 構成 (*.cfg)
 - ICS リポジトリー (*.in、*.out)

ICS 環境でのコネクタの構成にリポジトリー・ファイルが使用された場合には、このオプションを選択します。リポジトリー・ファイルに複数のコネクタ定義が含まれている場合は、ファイルを開くとすべての定義が表示されます。

- すべてのファイル (*.*)

コネクタのアダプター・パッケージに *.txt ファイルが付属していた場合、または別の拡張子で定義ファイルが使用可能である場合は、このオプションを選択します。

3. ディレクトリー表示内で、適切なコネクタ定義ファイルへ移動し、ファイルを選択し、「開く」をクリックします。

System Manager プロジェクトからコネクタ構成を開くには、以下のステップを実行します。

1. System Manager を始動します。System Manager が開始されている場合にのみ、構成を System Manager から開いたり、System Manager に保管したりできます。
2. Connector Configurator を始動します。
3. 「ファイル」>「開く」>「プロジェクトから」をクリックします。

構成ファイルの完成

構成ファイルを開くか、プロジェクトからコネクタを開くと、「Connector Configurator」ウィンドウに構成画面が表示されます。この画面には、現在の属性と値が表示されます。

構成画面のタイトルには、ファイル内で指定された統合ブローカーとコネクタの名前が表示されます。正しいブローカーが設定されていることを確認してください。正しいブローカーが設定されていない場合、コネクタを構成する前にブローカー値を変更してください。これを行うには、以下のステップを実行します。

1. 「標準のプロパティ」タブで、BrokerType プロパティの値フィールドを選択します。ドロップダウン・メニューで、値 ICS、WMQI、または WAS を選択します。
2. 選択したブローカーに関連付けられているプロパティが「標準のプロパティ」タブに表示されます。ここでファイルを保管するか、または 62 ページの『サポートされるビジネス・オブジェクト定義の指定』の説明に従い残りの構成フィールドに値を入力することができます。
3. 構成が完了したら、「ファイル」>「保管」>「プロジェクトに」を選択するか、または「ファイル」>「保管」>「ファイルに」を選択します。

ファイルに保管する場合は、*.cfg を拡張子として選択し、ファイルの正しい格納場所を選択して、「保管」をクリックします。

複数のコネクタ構成を開いている場合、構成をすべてファイルに保管するには「すべてファイルに保管」を選択し、コネクタ構成をすべて System Manager プロジェクトに保管するには「すべてプロジェクトに保管」をクリックします。

Connector Configurator では、ファイルを保管する前に、必須の標準プロパティすべてに値が設定されているかどうかを確認されます。必須の標準プロパティに値が設定されていない場合、Connector Configurator は、検証が失敗したというメッセージを表示します。構成ファイルを保管するには、そのプロパティの値を指定する必要があります。

構成ファイル・プロパティの設定

新規のコネクタ構成ファイルを作成して名前を付けるとき、または既存のコネクタ構成ファイルを開くときには、Connector Configurator によって構成画面が表示されます。構成画面には、必要な構成値のカテゴリに対応する複数のタブがあります。

Connector Configurator では、すべてのブローカーで実行されているコネクタで、以下のカテゴリのプロパティに値が設定されている必要があります。

- 標準のプロパティ
- コネクタ固有のプロパティ
- サポートされるビジネス・オブジェクト
- トレース/ログ・ファイルの値
- データ・ハンドラー (保証付きイベント・デリバリーで JMS メッセージングを使用するコネクタの場合に該当する)

注: JMS メッセージングを使用するコネクタの場合は、データをビジネス・オブジェクトに変換するデータ・ハンドラーの構成に関して追加のカテゴリが表示される場合があります。

ICS で実行されているコネクタの場合、以下のプロパティの値も設定されている必要があります。

- 関連付けられたマップ
- リソース
- メッセージング (該当する場合)

重要: Connector Configurator では、英語文字セットまたは英語以外の文字セットのいずれのプロパティ値も設定可能です。ただし、標準のプロパティおよびコネクタ固有プロパティ、およびサポートされるビジネス・オブジェクトの名前では、英語文字セットのみを使用する必要があります。

標準プロパティとコネクタ固有プロパティの違いは、以下のとおりです。

- コネクタの標準プロパティは、コネクタのアプリケーション固有のコンポーネントとブローカー・コンポーネントの両方によって共有されます。すべてのコネクタが同じ標準プロパティのセットを使用します。これらのプロパティの説明は、各アダプター・ガイドの付録 A にあります。変更できるのはこれらの値の一部のみです。
- アプリケーション固有のプロパティは、コネクタのアプリケーション固有コンポーネント (アプリケーションと直接対話するコンポーネント) のみに適用されます。各コネクタには、そのコネクタのアプリケーションだけで使用されるアプリケーション固有のプロパティがあります。これらのプロパティには、デフォルト値が用意されているものもあれば、そうでないものもあります。また、一部のデフォルト値は変更することができます。各アダプター・ガイドのインストールおよび構成の章に、アプリケーション固有のプロパティおよび推奨値が記述されています。

「標準プロパティ」と「コネクタ固有プロパティ」のフィールドは、どのフィールドが構成可能であるかを示すために色分けされています。

- 背景がグレーのフィールドは、標準のプロパティを表します。値を変更することはできますが、名前の変更およびプロパティの除去はできません。
- 背景が白のフィールドは、アプリケーション固有のプロパティを表します。これらのプロパティは、アプリケーションまたはコネクタの特定のニーズによって異なります。値の変更も、これらのプロパティの除去も可能です。
- 「値」フィールドは構成できます。
- プロパティごとに「更新メソッド」フィールドが表示されます。これは、変更された値をアクティブにするためにコンポーネントまたはエージェントの再始動が必要かどうかを示します。この設定を構成することはできません。

標準コネクタ・プロパティの設定

標準のプロパティの値を変更するには、以下の手順を実行します。

1. 値を設定するフィールド内でクリックします。
2. 値を入力するか、ドロップダウン・メニューが表示された場合にはメニューから値を選択します。
3. 標準のプロパティの値をすべて入力後、以下のいずれかを実行することができます。
 - 変更内容を破棄し、元の値を保持したままで Connector Configurator を終了するには、「ファイル」>「終了」をクリックし (またはウィンドウを閉じ)、変更内容を保管するかどうかを確認するプロンプトが出されたら「いいえ」をクリックします。
 - Connector Configurator 内の他のカテゴリーの値を入力するには、そのカテゴリーのタブを選択します。「標準のプロパティ」 (またはその他のカテゴリー) で入力した値は、次のカテゴリーに移動しても保持されます。ウィンドウ

を閉じると、すべてのカテゴリで入力した値を一括して保管するかまたは破棄するかを確認するプロンプトが出されます。

- 修正した値を保管するには、「ファイル」>「終了」をクリックし（またはウィンドウを閉じ）、変更内容を保管するかどうかを確認するプロンプトが出されたら「はい」をクリックします。「ファイル」メニューまたはツールバーから「保管」>「ファイルに」をクリックする方法もあります。

アプリケーション固有の構成プロパティの設定

アプリケーション固有の構成プロパティの場合、プロパティ名の追加または変更、値の構成、プロパティの削除、およびプロパティの暗号化が可能です。プロパティのデフォルトの長さは 255 文字です。

1. グリッドの左上端の部分で右マウス・ボタンをクリックします。ポップアップ・メニュー・バーが表示されます。プロパティを追加するときは「追加」をクリックします。子プロパティを追加するには、親の行番号で右マウス・ボタンをクリックし、「子を追加」をクリックします。
2. プロパティまたは子プロパティの値を入力します。
3. プロパティを暗号化するには、「暗号化」ボックスを選択します。
4. 60 ページの『標準コネクタ・プロパティの設定』の説明に従い、変更内容を保管するかまたは破棄するかを選択します。

各プロパティごとに表示される「更新メソッド」は、変更された値をアクティブにするためにコンポーネントまたはエージェントの再始動が必要かどうかを示します。

重要: 事前設定のアプリケーション固有のコネクタ・プロパティ名を変更すると、コネクタに障害が発生する可能性があります。コネクタをアプリケーションに接続したり正常に実行したりするために、特定のプロパティ名が必要である場合があります。

コネクタ・プロパティの暗号化

「コネクタ固有プロパティ」ウィンドウの「暗号化」チェック・ボックスにチェックマークを付けると、アプリケーション固有のプロパティを暗号化することができます。値の暗号化を解除するには、「暗号化」チェック・ボックスをクリックしてチェックマークを外し、「検証」ダイアログ・ボックスに正しい値を入力し、「OK」をクリックします。入力された値が正しい場合は、暗号化解除された値が表示されます。

各プロパティとそのデフォルト値のリストおよび説明は、各コネクタのアダプター・ユーザズ・ガイドにあります。

プロパティに複数の値がある場合には、プロパティの最初の値に「暗号化」チェック・ボックスが表示されます。「暗号化」を選択すると、そのプロパティのすべての値が暗号化されます。プロパティの複数の値を暗号化解除するには、そのプロパティの最初の値の「暗号化」チェック・ボックスをクリックしてチェックマークを外してから、「検証」ダイアログ・ボックスで新規の値を入力します。入力値が一致すれば、すべての複数值が暗号化解除されます。

更新メソッド

付録『コネクターの標準構成プロパティー』の 127 ページの『プロパティー値の設定と更新』にある更新メソッドの説明を参照してください。

サポートされるビジネス・オブジェクト定義の指定

コネクターで使用するビジネス・オブジェクトを指定するには、Connector Configurator の「サポートされているビジネス・オブジェクト」タブを使用します。汎用ビジネス・オブジェクトと、アプリケーション固有のビジネス・オブジェクトの両方を指定する必要があり、またそれらのビジネス・オブジェクト間のマップの関連を指定することが必要です。

注: コネクターによっては、アプリケーションでイベント通知や (メタオブジェクトを使用した) 追加の構成を実行するために、特定のビジネス・オブジェクトをサポートされているものとして指定することが必要な場合もあります。詳細は、「コネクター開発ガイド (C++ 用)」または「コネクター開発ガイド (Java 用)」を参照してください。

ご使用のブローカーが ICS の場合

ビジネス・オブジェクト定義がコネクターでサポートされることを指定する場合や、既存のビジネス・オブジェクト定義のサポート設定を変更する場合は、「サポートされているビジネス・オブジェクト」タブをクリックし、以下のフィールドを使用してください。

ビジネス・オブジェクト名: ビジネス・オブジェクト定義がコネクターによってサポートされることを指定するには、System Manager を実行し、以下の手順を実行します。

1. 「ビジネス・オブジェクト名」リストで空のフィールドをクリックします。
System Manager プロジェクトに存在するすべてのビジネス・オブジェクト定義を示すドロップダウン・リストが表示されます。
2. 追加するビジネス・オブジェクトをクリックします。
3. ビジネス・オブジェクトの「エージェント・サポート」(以下で説明) を設定します。
4. 「Connector Configurator」ウィンドウの「ファイル」メニューで、「プロジェクトに保管」をクリックします。追加したビジネス・オブジェクト定義に指定されたサポートを含む、変更されたコネクター定義が、System Manager の ICL (Integration Component Library) プロジェクトに保管されます。

サポートされるリストからビジネス・オブジェクトを削除する場合は、以下の手順を実行します。

1. ビジネス・オブジェクト・フィールドを選択するため、そのビジネス・オブジェクトの左側の番号をクリックします。
2. 「Connector Configurator」ウィンドウの「編集」メニューから、「行を削除」をクリックします。リスト表示からビジネス・オブジェクトが除去されます。
3. 「ファイル」メニューから、「プロジェクトの保管」をクリックします。

サポートされるリストからビジネス・オブジェクトを削除すると、コネクター定義が変更され、削除されたビジネス・オブジェクトはコネクターのこの実装で使用不

可になります。コネクターのコードに影響したり、そのビジネス・オブジェクト定義そのものが System Manager から削除されることはありません。

エージェント・サポート: ビジネス・オブジェクトがエージェント・サポートを備えている場合、システムは、コネクター・エージェントを介してアプリケーションにデータを配布する際にそのビジネス・オブジェクトの使用を試みます。

一般に、コネクターのアプリケーション固有ビジネス・オブジェクトは、そのコネクターのエージェントによってサポートされますが、汎用ビジネス・オブジェクトはサポートされません。

ビジネス・オブジェクトがコネクター・エージェントによってサポートされるよう指定するには、「エージェント・サポート」ボックスにチェックマークを付けます。「Connector Configurator」ウィンドウでは「エージェント・サポート」の選択の妥当性は検査されません。

最大トランザクション・レベル: コネクターの最大トランザクション・レベルは、そのコネクターがサポートする最大のトランザクション・レベルです。

ほとんどのコネクターの場合、選択可能な項目は「最大限の努力」のみです。

トランザクション・レベルの変更を有効にするには、サーバーを再始動する必要があります。

ご使用のブローカーが WebSphere Message Broker の場合

スタンドアロン・モードで作業している (System Manager に接続していない) 場合、手動でビジネス・オブジェクト名を入力する必要があります。

System Manager を実行している場合、「サポートされているビジネス・オブジェクト」タブの「ビジネス・オブジェクト名」列の下にある空のボックスを選択できます。コンボ・ボックスが表示され、コネクターが属する統合コンポーネント・ライブラリー・プロジェクトから選択可能なビジネス・オブジェクトのリストが示されます。リストから必要なビジネス・オブジェクトを選択します。

「メッセージ・セット ID」は、WebSphere Business Integration Message Broker 5.0 のオプションのフィールドです。この ID が提供される場合、一意である必要はありません。ただし、WebSphere MQ Integrator および Integrator Broker 2.1 の場合は、一意の ID を提供する必要があります。

ご使用のブローカーが WAS の場合

使用するブローカー・タイプとして WebSphere Application Server を選択した場合、Connector Configurator にメッセージ・セット ID は必要ありません。「サポートされているビジネス・オブジェクト」タブには、サポートされるビジネス・オブジェクトの「ビジネス・オブジェクト名」列のみが表示されます。

スタンドアロン・モードで作業している (System Manager に接続していない) 場合、手動でビジネス・オブジェクト名を入力する必要があります。

System Manager を実行している場合、「サポートされているビジネス・オブジェクト」タブの「ビジネス・オブジェクト名」列の下にある空のボックスを選択できます。コンボ・ボックスが表示され、コネクターが属する統合コンポーネント・ライ

ブラリー・プロジェクトから選択可能なビジネス・オブジェクトのリストが示されます。このリストから必要なビジネス・オブジェクトを選択します。

関連付けられたマップ (ICS のみ)

各コネクタは、現在 WebSphere InterChange Server でアクティブなビジネス・オブジェクト定義、およびそれらの関連付けられたマップのリストをサポートします。このリストは、「**関連付けられたマップ**」タブを選択すると表示されます。

ビジネス・オブジェクトのリストには、エージェントでサポートされるアプリケーション固有のビジネス・オブジェクトと、コントローラーがサブスクライブ・コラボレーションに送信する、対応する汎用オブジェクトが含まれます。マップの関連によって、アプリケーション固有のビジネス・オブジェクトを汎用ビジネス・オブジェクトに変換したり、汎用ビジネス・オブジェクトをアプリケーション固有のビジネス・オブジェクトに変換したりするときに、どのマップを使用するかが決定されます。

特定のソースおよび宛先ビジネス・オブジェクトについて一意的に定義されたマップを使用する場合、表示を開くと、マップは常にそれらの該当するビジネス・オブジェクトに関連付けられます。ユーザーがそれらを変更する必要はありません (変更できません)。

サポートされるビジネス・オブジェクトで使用可能なマップが複数ある場合は、そのビジネス・オブジェクトを、使用する必要のあるマップに明示的にバインドすることが必要になります。

「**関連付けられたマップ**」タブには以下のフィールドが表示されます。

- **ビジネス・オブジェクト名**

これらは、「**サポートされているビジネス・オブジェクト**」タブで指定した、このコネクタでサポートされるビジネス・オブジェクトです。「**サポートされているビジネス・オブジェクト**」タブでビジネス・オブジェクトを追加指定した場合、その内容は、「Connector Configurator」ウィンドウの「**ファイル**」メニューから「**プロジェクトに保管**」を選択して、変更を保管した後に、このリストに反映されます。

- **関連付けられたマップ**

この表示には、コネクタの、サポートされるビジネス・オブジェクトでの使用のためにシステムにインストールされたすべてのマップが示されます。各マップのソース・ビジネス・オブジェクトは、「**ビジネス・オブジェクト名**」表示でマップ名の左側に表示されます。

- **明示的**

場合によっては、関連付けられたマップを明示的にバインドすることが必要になります。

明示的バインディングが必要なのは、特定のサポートされるビジネス・オブジェクトに複数のマップが存在する場合のみです。ICS は、ブート時、各コネクタでサポートされるそれぞれのビジネス・オブジェクトにマップを自動的にバインドしようとします。複数のマップでその入力データとして同一のビジネス・オブ

ジェクトが使用されている場合、サーバーは、他のマップのスーパーセットである 1 つのマップを見つけて、バインドしようとします。

他のマップのスーパーセットであるマップがないと、サーバーは、ビジネス・オブジェクトを単一のマップにバインドすることができないため、バインディングを明示的に設定することが必要になります。

以下の手順を実行して、マップを明示的にバインドします。

1. 「明示的 (Explicit)」列で、バインドするマップのチェック・ボックスにチェックマークを付けます。
2. ビジネス・オブジェクトに関連付けるマップを選択します。
3. 「Connector Configurator」ウィンドウの「ファイル」メニューで、「プロジェクトに保管」をクリックします。
4. プロジェクトを ICS に配置します。
5. 変更を有効にするため、サーバーをリポートします。

リソース (ICS)

「リソース」タブでは、コネクター・エージェントが、コネクター・エージェント並列処理を使用して同時に複数のプロセスを処理するかどうか、またどの程度処理するかを決定する値を設定できます。

すべてのコネクターがこの機能をサポートしているわけではありません。複数のプロセスを使用するよりも複数のスレッドを使用する方が通常は効率的であるため、Java でマルチスレッドとして設計されたコネクター・エージェントを実行している場合、この機能を使用することはお勧めできません。

メッセージング (ICS)

メッセージング・プロパティは、DeliveryTransport 標準プロパティの値として MQ を設定し、ブローカー・タイプとして ICS を設定した場合にのみ、使用可能です。これらのプロパティは、コネクターによるキューの使用方法に影響しません。

トレース/ログ・ファイル値の設定

コネクター構成ファイルまたはコネクター定義ファイルを開くと、Connector Configurator は、そのファイルのログおよびトレースの値をデフォルト値として使用します。Connector Configurator 内でこれらの値を変更できます。

ログとトレースの値を変更するには、以下の手順を実行します。

1. 「トレース/ログ・ファイル」タブをクリックします。
2. ログとトレースのどちらでも、以下のいずれかまたは両方へのメッセージの書き込みを選択できます。
 - コンソールに (STDOUT):
ログ・メッセージまたはトレース・メッセージを STDOUT ディスプレイに書き込みます。

注: STDOUT オプションは、Windows プラットフォームで実行しているコネクターの「トレース/ログ・ファイル」タブでのみ使用できます。

- ファイルに:
ログ・メッセージまたはトレース・メッセージを指定されたファイルに書き込みます。ファイルを指定するには、ディレクトリー・ボタン (省略符号) をクリックし、指定する格納場所へ移動し、ファイル名を指定し、「保管」をクリックします。ログ・メッセージまたはトレース・メッセージは、指定した場所の指定したファイルに書き込まれます。

注: ログ・ファイルとトレース・ファイルはどちらも単純なテキスト・ファイルです。任意のファイル拡張子を使用してこれらのファイル名を設定できます。ただし、トレース・ファイルの場合、拡張子として `.trc` ではなく `.trace` を使用することをお勧めします。これは、システム内に存在する可能性がある他のファイルとの混同を避けるためです。ログ・ファイルの場合、通常使用されるファイル拡張子は `.log` および `.txt` です。

データ・ハンドラー

データ・ハンドラー・セクションの構成が使用可能となるのは、`DeliveryTransport` の値に `JMS` を、また `ContainerManagedEvents` の値に `JMS` を指定した場合のみです。すべてのアダプターでデータ・ハンドラーを使用できるわけではありません。

これらのプロパティーに使用する値については、付録 A の『標準構成プロパティー』の `ContainerManagedEvents` の下の説明を参照してください。その他の詳細は、「コネクター開発ガイド (C++ 用)」または「コネクター開発ガイド (Java 用)」を参照してください。

構成ファイルの保管

コネクターの構成が完了したら、コネクター構成ファイルを保管します。`Connector Configurator` では、構成中に選択したブローカー・モードでファイルを保管します。`Connector Configurator` のタイトル・バーには現在のブローカー・モード (`ICS`、`WMQI`、または `WAS`) が常に表示されます。

ファイルは XML 文書として保管されます。XML 文書は次の 3 通りの方法で保管できます。

- `System Manager` から、統合コンポーネント・ライブラリーに `*.con` 拡張子付きファイルとして保管します。
- 指定したディレクトリーに保管します。
- スタンドアロン・モードで、ディレクトリー・フォルダーに `*.cfg` 拡張子付きファイルとして保管します。デフォルトでは、このファイルは `¥WebSphereAdapters¥bin¥Data¥App` に保管されます。
- `WebSphere Application Server` プロジェクトをセットアップしている場合には、このファイルを `WebSphere Application Server` プロジェクトに保管することもできます。

`System Manager` でのプロジェクトの使用法、および配置の詳細については、以下のインプリメンテーション・ガイドを参照してください。

- `ICS`: 「*WebSphere InterChange Server* システム・インプリメンテーション・ガイド」

- WebSphere Message Brokers: 「WebSphere Message Brokers 使用アダプター・インプリメンテーション・ガイド」
- WAS: 「アダプター実装ガイド (WebSphere Application Server)」

構成ファイルの変更

既存の構成ファイルの統合ブローカー設定を変更できます。これにより、他のブローカーで使用する構成ファイルを新規に作成するときに、このファイルをテンプレートとして使用できます。

注: 統合ブローカーを切り替える場合には、ブローカー・モード・プロパティーと同様に他の構成プロパティーも変更する必要があります。

既存の構成ファイルでのブローカーの選択を変更するには、以下の手順を実行します (オプション)。

- Connector Configurator で既存の構成ファイルを開きます。
- 「標準のプロパティー」タブを選択します。
- 「標準のプロパティー」タブの「**BrokerType**」フィールドで、ご使用のブローカーに合った値を選択します。
現行値を変更すると、プロパティー画面の利用可能なタブおよびフィールド選択がただちに變更され、選択した新規ブローカーに適したタブとフィールドのみが表示されます。

構成の完了

コネクターの構成ファイルを作成し、そのファイルを変更した後で、コネクターの始動時にコネクターが構成ファイルの位置を特定できるかどうかを確認してください。

これを行うには、コネクターが使用する始動ファイルを開き、コネクター構成ファイルに使用されている格納場所とファイル名が、ファイルに対して指定した名前およびファイルを格納したディレクトリーまたはパスと正確に一致しているかどうかを検証します。

グローバル化環境における Connector Configurator の使用

Connector Configurator はグローバル化され、構成ファイルと統合ブローカー間の文字変換を処理できます。Connector Configurator では、ネイティブなエンコード方式を使用しています。構成ファイルに書き込む場合は UTF-8 エンコード方式を使用します。

Connector Configurator は、以下の場所で英語以外の文字をサポートします。

- すべての値のフィールド
- ログ・ファイルおよびトレース・ファイル・パス (「**トレース/ログ・ファイル**」タブで指定)

CharacterEncoding および Locale 標準構成プロパティーのドロップ・リストに表示されるのは、サポートされる値の一部のみです。ドロップ・リストに、サポート

される他の値を追加するには、製品ディレクトリーの ¥Data¥Std¥stdConnProps.xml ファイルを手動で変更する必要があります。

例えば、Locale プロパティの値のリストにロケール en_GB を追加するには、stdConnProps.xml ファイルを開き、以下に太文字で示した行を追加してください。

```
<Property name="Locale"
isRequired="true"
updateMethod="component restart">
  <ValidType>String</ValidType>
  <ValidValues>
    <Value>ja_JP</Value>
    <Value>ko_KR</Value>
    <Value>zh_CN</Value>
    <Value>zh_TW</Value>
    <Value>fr_FR</Value>
    <Value>de_DE</Value>
    <Value>it_IT</Value>
    <Value>es_ES</Value>
    <Value>pt_BR</Value>
    <Value>en_US</Value>
    <Value>en_GB</Value>
  </ValidValues>
  <DefaultValue>en_US</DefaultValue>
</Property>
```

第 6 章 ブリッジの構成

- 『ブリッジのパラメーター』
- 『構成ファイル』
- 72 ページの『イベント処理』
- 74 ページの『要求処理』
- 77 ページの『ユーザー定義キーおよび FRONT ARENA キー』
- 78 ページの『SMTP アラート』
- 79 ページの『WBI アダプターとの通信』
- 81 ページの『他のアプリケーションとの通信』

この章では、アダプターのブリッジ・コンポーネントの構成プロパティ値を設定する方法について説明します。FRONT ARENA と WebSphere MQ との間のブリッジは、ネイティブの FRONT ARENA メッセージ形式で、メッセージの読み取りや書き込みを MQ キューとの間で行います。キューに格納されたメッセージは、その後、Adapter for FRONT ARENA またはその他のサード・パーティー・アプリケーションからアクセスできます。

注: 本書では、ディレクトリー・パスに円記号 (¥) を使用します。UNIX システムを使用している場合は、円記号をスラッシュ (/) に置き換えてください。また、各オペレーティング・システムの規則に従ってください。

ブリッジのパラメーター

Adapter for FRONT ARENA のブリッジ実行可能コンポーネントに存在するパラメーターは 1 つで、それは、使用する構成ファイルの名前と場所です。

このパラメーターは必須です。このパラメーターを指定していない場合、ブリッジは使用法のメッセージを表示して、即時に終了します。無効またはアクセス不能な構成ファイルを指定した場合も、ブリッジは即時に終了します。終了した理由を説明するメッセージがコマンド・ウィンドウに表示されます。

構成ファイル

ブリッジには、構成ファイルに指定されている構成オプションが多数存在します。この構成ファイルは、スタンザまたはセクションで分割されます。

スタンザは、大括弧で囲まれた名前が始まります (例: [stanza1])。ただし、最初のスタンザは例外で、これには明示的なヘッダーは必要ありません。最初のスタンザは、新しいスタンザが始まるか、または構成ファイルが終了すると終了します。

各スタンザには、論理的に関連した 1 組の構成パラメーターが、キーワード=値の形式で格納されます。キーワードはスタンザごとに固有にする必要がありますが、異なるスタンザに対しては同じキーワードを使用できます。

コメントは構成ファイルの任意の場所に挿入できます。コメント行では、最初の列の先頭にハッシュ・プラグマ (#) を付ける必要があります。コメント行は、入力パーサーによって単純に無視されます。任意の場所に挿入できるブランク行についても、同じように無視されます。

ブリッジの構成パラメーター

構成プロパティは、プログラムの開始時にブリッジによって読み取られます。構成ファイルに変更を適用しても、ブリッジを停止または再始動しない限り無視されます。必須のスタンザまたはプロパティが欠落していると、Adapter for FRONT ARENA は即時に終了します。

注: スタンザ名、キーワード、および値は大文字と小文字を区別します。

スタンザの順に並べ替えられた構成パラメーターは、ブリッジによってサポートされています。

一般的な構成の設定

メイン (無名) のスタンザには、FRONT ARENA の AMB コンポーネントと通信するとき使用するいくつかの汎用構成プロパティおよびパラメーターが格納されています。

表 21. メイン (無名) スタンザ

キーワード	必須/オプション/デフォルト値	指定可能な値	説明
Mode	必須	Event、Request、Both	ブリッジによってサポートされている処理のタイプは、次のとおりです。 <ul style="list-style-type: none"> • Event: イベント処理のみ • Request: 要求処理のみ • Both: イベント処理と要求処理の両方
FAHost	必須	有効な URL	FRONT ARENA のホスティングを実行するシステムの URL。
FAAMBPport	必須	1 から 65536	FRONT ARENA AMB サーバーのポート番号。
FAUserID	オプション	1 から 65536。有効な FRONT ARENA (AMB) ユーザー ID	AMB にログインするとき使用するユーザー ID。 注: ユーザー ID とパスワードが指定されていない場合は、AMB.ini
FAPassword	オプション	有効な FRONT ARENA (AMB) パスワード	ファイルからの値が適用されず。AMB にログインするとき使用するパスワード。

表 21. メイン (無名) スタンザ (続き)

キーワード	必須/オプション/デフォルト値	指定可能な値	説明
FAMonitoring Port	オプション	1 から 65536	FRONT ARENA System Monitor のポート番号。 注: これがサポートされるのは、FRONT ARENA 1.5 以上のみです。
FARetries	オプション デフォルト: 3	整数値: 値がゼロの場合は、際限なく再試行するという意味になります。	AMB への接続を確立できない場合または中止する場合の再試行回数。
FARetryInterval	オプション デフォルト: 60000	整数値 > 0	AMB 接続を確立するための連続した再試行間の時間 (ミリ秒単位)。
PollFrequency	オプション デフォルト: 10000	整数値 > 0	AMB または WebSphere MQ キューからメッセージを連続して読み取る時間間隔 (ミリ秒単位)。トラフィックが混雑している場合、この時間間隔は、アプリケーションによって自動的に縮小されます。
TraceFile	オプション 注: トレースのスイッチを入れた場合 (TraceLevel>0) は、このプロパティを指定する必要があります。	(パス名および) ファイル名	トレース・ファイルの名前。
TraceLevel	オプション デフォルト: 0	0: トレースのスイッチがオフ 1: エラー・メッセージ 2: 現在は未使用 3: 現在は未使用 4: トレース呼び出しシーケンス 5: 通知メッセージ	トレース・レベル。 注: 高いトレース・レベルの場合、低いトレース・レベルからのメッセージがすべて対象になります。
LogFile	必須	(パス名および) ファイル名	ログ・ファイルの名前。
LogLevel	オプション デフォルト: 3	1: ログ・エラーのみ 2: ログ・エラーおよび警告 3: ログ・エラー、警告、および通知メッセージ	ログ・レベル。

イベント処理

表 22 に示すスタanzasは、ブリッジをイベント処理のために構成する場合、例えば、Mode パラメーターを Event または Both に設定する場合に指定する必要があります。

スタanzas EventHandler には、主に、ターゲット・アプリケーションとの WebSphere MQ 接続を確立するために必要な WebSphere MQ パラメーターが格納されています。この WebSphere MQ 接続は、FRONT ARENA イベント通知をターゲット・アプリケーションに転送するときを使用します。

表 22. スタanzas EventHandler

キーワード	必須/オプション/デフォルト値	指定可能な値	説明
Name	必須	既存の AMB クライアントの名前。	FRONT ARENA との通信に使用するクライアント名。詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。
WMQueueManager	必須	既存の WebSphere MQ キュー・マネージャーの名前。	ターゲット・アプリケーションとの通信に使用する WebSphere MQ キュー・マネージャー。詳細については、79 ページの『WBI アダプターとの通信』を参照してください。
WMChannel	必須	<channel/ <transport>/ hostname(port)	特定のキュー・マネージャーとの通信に使用する WebSphere MQ チャネル/トランスポート (通常は TCP)/ キュー・マネージャーをホスティングするシステムの URL + キュー・マネージャー・リスナー・ポート。詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。
WMQueue	必須	特定のキュー・マネージャーが所有する既存の WebSphere MQ キュー・マネージャーの名前。	イベント・メッセージをターゲット・アプリケーションに転送するとき使用する WebSphere MQ キュー。詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。

表 22. スタンザ *EventHandler* (続き)

キーワード	必須/オプション/デフォルト値	指定可能な値	説明
WMQUserID	オプション	特定のキュー・マネージャーに認識されているユーザー ID。	ブリッジによって送信されたメッセージで使用するユーザー ID。
WMQRetries	オプション デフォルト: 3	整数値。 負の値の場合は、 際限なく再試行するという意味。	WebSphere MQ 接続を確立できない場合または中止する場合の再試行回数。
WMQRetry Interval	オプション デフォルト: 60000	整数値 > 0	WebSphere MQ 接続を確立するための連続した再試行間の時間 (ミリ秒単位)。

イベント・サブスクリプションおよびフィルター処理

FRONT ARENA では、トピックの下にイベント通知をパブリッシュします。トピックは次の部分から構成されます。

- 後ろにスラッシュ (/) の付いた FRONT ARENA AMBA 構成パラメーターによって定義されているプレフィックス。構成パラメーターの詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。
- イベントの影響を受けるオブジェクトのタイプ。

注: TRADE オブジェクトに対する更新内容は、トピック AMBA/TRADE の下にパブリッシュされます (AMBA は 1 からのプレフィックスであることが前提です)。

この単一トピックにサブスクライブしているクライアント・アプリケーションは、TRADE オブジェクトに対する変更の通知を受け取りますが、INSTRUMENT オブジェクトの変更は通知しません。スタンザ *EventHandler* および *Filter* のインスタンスの主な用途は、アプリケーションのサブスクライブ先のトピックがどれであるかをアダプターのブリッジ・コンポーネントに通知することです。この方法で定義されたイベント・フィルター処理では結果が粗すぎる場合、このスタンザに追加のフィルター処理基準を指定できます。

各イベント通知メッセージには、次の属性が格納されています。

- **TYPE:** イベント通知を発生させた操作と、この操作の適用先であったオブジェクトのタイプが、次の構文で記述されます。
`<operation>_<object type>`
- **SOURCE:** イベント通知の発信元が格納されます (通常は FRONT ARENA AMBA の名前になります)。

前述の例では、TYPE は UPDATE_TRADE となり、SOURCE は、例えば ProdAmba になります。このスタンザに指定できる追加のフィルター処理基準では、これら 2 つのメッセージ属性が使用されます。これらのメッセージ属性を使用すると、TRADE オブジェクトの更新によって発生したイベント通知と、ProdAmba によってパブリッ

シュされるイベント通知のみを転送するようにブリッジが構成されます。この例では、ブリッジの構成が次の設定によって実行されます。

- Topics は AMBA/TRADE に
- Verbs は UPDATE に
- Sources は ProdAmba に

スタンザ EventHandler/Filter<n> (<n> は整数値):

表 23. イベント通知

キーワード	必須/オプション/デフォルト	指定可能な値	説明
Topics	必須	任意の FRONT ARENA トピック名 (例えば、EXT/TRADE)。	ブリッジのサブスクリプション先にする FRONT ARENA トピック名 (セミコロン区切り) です。
Sources	オプション	任意の FRONT ARENA ソース名 (例えば、AMBA)。	これらのソース (セミコロン区切り) からのメッセージのみが検討されます。
Objects	オプション	任意の FRONT ARENA オブジェクト (例えば、TRADE)。	これらのオブジェクト (セミコロン区切り) のみが検討されます。注: これは、前述の <operation>_<object type> の組からの 2 番目のトークンです。
Verbs	オプション	任意の FRONT ARENA 動詞 (INSERT、UPDATE、および DELETE)。	これらの動詞 (セミコロン区切り) のみが検討されます。注: これは、前述の <operation>_<object type> の組からの最初のトークンです。

このスタンザのいくつかのインスタンス (名前は EventHandler/Filter1 や EventHandler2 など) を指定できます。

前述のように、各スタンザはフィルターを定義します。Topics プロパティ内にサブスクリプションされたイベントの組は、Sources、Objects、Verbs の各パラメーターによって制限できます。いくつかのスタンザが用意されている場合、アダプターのブリッジ・コンポーネントによって転送されたイベントの組は、別のスタンザによって指定された組の共用体です。このスタンザのすべてのプロパティには、複数の値をセミコロン区切りで保持できます。

要求処理

次に示すスタンザは、アダプターのブリッジ・コンポーネントを要求処理のために構成する場合、例えば、Mode パラメーターを Request または Both に設定する場合に指定する必要があります。

要求の処理を同期方式または非同期方式のどちらにするかは、パラメーター WMQReplyQueue および WMQCorrelationQueue を指定するか省略するかによって決ま

ります。スタanzasを指定するには、以下の複数の表を参照してください。

表 24. 要求処理

パラメーター WMQReplyQueue および WMQCorrelationQueue	要求処理
両方とも指定しない	非同期
両方とも指定する	同期
一方を指定し、もう一方は指定しない	不可 (構成エラー)

表 25. Stanzas *RequestHandler*

キーワード	必須/オプション/ デフォルト値	指定可能な値	説明
Name	必須	既存の AMB クライアントの名前。	FRONT ARENA アプリケーションとの通信に使用するクライアント名。詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。
WMQQueueManager	必須	既存の WebSphere MQ キュー・マネージャーの名前。	ターゲット・アプリケーションとの通信に使用する WebSphere MQ キュー・マネージャー。詳細については、79 ページの『WBI アダプターとの通信』を参照してください。
WMQChannel	必須	<channel/ <transport>/ hostname(port)	特定のキュー・マネージャーとの通信に使用する WebSphere MQ チャネル/ トランスポート (通常は TCP)/ キュー・マネージャーをホスティングするシステムの URL + キュー・マネージャー・リスナー・ポート。詳細については、79 ページの『WBI アダプターとの通信』を参照してください。
WMQRequestQueue	必須	特定のキュー・マネージャーが所有する既存の WebSphere MQ キュー・マネージャーの名前。	アダプターのブリッジ・コンポーネントに要求を送信するときに使用する WebSphere MQ キュー。

表 25. スタンザ RequestHandler (続き)

キーワード	必須/オプション/ デフォルト値	指定可能な値	説明
WMQReplyQueue	オプション 注: このパラメーターは、WMQCorrelationQueueを指定した場合に指定する必要があります。	特定のキュー・マネージャーが所有する既存の WebSphere MQ キュー・マネージャーの名前。	関連した要求の送信元への応答を返すときに使用する WebSphere MQ キュー。
WMQCorrelationQueue	オプション 注: このパラメーターは、WMQReplyQueueを指定した場合に指定する必要があります。	特定のキュー・マネージャーが所有する既存の WebSphere MQ キュー・マネージャーの名前。	要求と応答の関連のためにアダプターのブリッジ・コンポーネントが内部で使用する WebSphere MQ キュー。
WMQUserID	オプション	特定のキュー・マネージャーに認識されているユーザー ID。	ブリッジ・コンポーネントによって送信されたメッセージで使用されるユーザー ID。
WMQRetries	オプション デフォルト: 3	整数値。負の値の場合は、際限なく再試行するという意味。	WebSphere MQ 接続を確立できない場合または中止する場合の再試行回数。
WMQRetryInterval	オプション	整数値 > 0	WebSphere MQ 接続を確立するための連続した再試行間の時間 (ミリ秒単位)。
FAPublicationSubject	必須	String	要求メッセージがブリッジによって AMB 上にパブリッシュされるトピック。詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。
FAAMBASenderSource	オプション	String	FRONT ARENA が要求の応答をパブリッシュする件名のプレフィックス。詳細については、85 ページの『第 7 章 アプリケーションの構成』を参照してください。

WMQQueueManager、WMQChannel などのパラメーターが、すでに EventHandler スタンザに指定されていた場合でも、再度指定する必要があります。値は同じでも異なっていたりもかまいません。

注: 大半のシナリオでは、同じ値を使用するのが適切です。

ユーザー定義キーおよび FRONT ARENA キー

オブジェクトは、コラボレーションを基にして FRONT ARENA システム内に作成できます。このようなオブジェクトは、オブジェクト・タイプとサポートされている動詞ごとにキーを定義することによって、更新および削除できます。このキーは、応答と発信元の要求との相関をとるときに必要です。FRONT ARENA アプリケーションは、これに対して一貫性のある内部機構を備えていません。

FRONT ARENA キー

FRONT ARENA システム内に作成されたオブジェクトごとに、FRONT ARENA アプリケーションによって内部キーが生成されます。このキーは、オブジェクト・タイプごとに固有です。例えば、すべての TRADE オブジェクトには、異なる内部キーが存在します。内部キーは、オブジェクトの 1 次データベース・テーブルに永続的に格納されます。キーを格納するために使用する列は、オブジェクト・タイプによって異なります。

TRADE オブジェクトの内部キーの例は、TRADE データベース・テーブルの列 `trdnbr` に格納されています。INSTRUMENT オブジェクトの内部キーの例は、INSTRUMENT データベース・テーブルの列 `insaddr` に格納されています。

AMB メッセージ ID

FRONT ARENA AMB バスに書き込まれる各メッセージには、AMB によって固有のメッセージ ID が割り当てられます。このメッセージ ID は、メッセージを送出する AMB クライアント・アプリケーションに対して使用できます。

メッセージが FRONT ARENA によって拒否された場合、システムから戻されるエラー応答には、原因となるメッセージのメッセージ ID が格納されています。

このことは、要求とエラー応答との相関をとるために、ブリッジによって活用されます。ただし、AMB メッセージ ID がブリッジによって外部化されることはありません。この ID は、Adapter for FRONT ARENA に送信された応答メッセージの一部でも、アダプターによって作成されたビジネス・オブジェクトの一部でもありません。

ユーザー定義キーとしての FRONT ARENA キー

要求と応答の相関を取るための作成 (挿入) シナリオに FRONT ARENA キーを使用できないことは明らかです。更新や削除の場合には使用できますが、その場合は要求側がこの内部キーを認識していることが前提になります。

すべての種類の要求が含まれるシナリオでは、通常、FRONT ARENA キーは使用されません。関係テーブルは、FRONT ARENA オブジェクトと、他のシステムの同じオブジェクトとの相関をとるために必要なコラボレーションによって保持され、使用される必要があります。通常、ユーザー定義キーは、これらのテーブルにある FRONT ARENA オブジェクトのキーとしても使用されます。

異なるタイプの要求に対して異なるキーを使用する場合は、これらのテーブルのいくつかが必要になります。したがって、コラボレーションの複雑さは大幅に増加します。

ユーザー定義キーの候補

固有値を割り当てることのできる（または内部的に割り当てられている）オブジェクトの属性だけがユーザー定義キーの候補になることは明らかです。それに加えて、対応するデータベースの列を索引の一部にする必要があります。

候補は、「FRONT ARENA Data Model Reference」(FCA 1051) の『Keys and Indexes』の章に記載されています。

例:

- TRADE オブジェクトでは、どのような場合でも `optional_key` を使用できます。
- 更新要求および削除要求では、前述のように `trdnbr` を使用できます。
- INSTRUMENT オブジェクトでは、どのような場合でも `isin`、`extern_id1`、および `extern_id2` を使用できます。
- 更新要求および削除要求では、`insaddr` および `insid` を使用できます。

表 26. スタンザ `RequestHandler/ObjectKeys`:

キーワード	必須/オプション/ デフォルト値	指定可能な値	説明
<code><requesttype>_<object></code>	必須	<code><object></code> の属性の名前。	タイプ <code><request type></code> の要求を <code><object></code> に対して実行する場合、ユーザー・キーとして使用する <code><object></code> の属性の名前。

有効な `<request type>`: INSERT、UPDATE、および DELETE。

有効な `<object>`: TRADE、INSTRUMENT など。(任意の FRONT ARENA オブジェクト)

例:

```
[RequestHandler/ObjectKeys]
INSERT_TRADE = OPTIONAL_KEY
UPDATE_TRADE = TRDNBR
DELETE_TRADE = TRDNBR
```

SMTP アラート

次のスタンザは、アダプターのブリッジがエラーを検出したときに SMTP メッセージが送出される場合、指定する必要があります。

表 27.

キーワード	必須/オプション/ デフォルト値	指定可能な値	説明
HostName	必須	有効な URL。	使用の対象となる SMTP サーバーの URL。
Port	オプション デフォルト: 25	SMTP ポート (通常は 25)。	SMTP ポートの E メール送信者の識別文字列。
Sender	必須	任意のストリング。	E メール送信者の識別文字列。

表 27. (続き)

キーワード	必須/オプション/デフォルト値	指定可能な値	説明
Receivers	オプション デフォルト: FRONT ARENA - WebSphere MQ ブリッジのアラ ート。	有効な E メール受 取人アドレス。	
Subject	オプション デフォルト: FRONT ARENA - WebSphere MQ ブリッジのアラ ート。	任意のストリング。	Eメールの件名。
UserID	オプション		メール・サーバーのユーザー ID。
Password	オプション		メール・サーバーのパスワー ド。

WBI アダプターとの通信

ブリッジ・コンポーネントとコネクタとの通信を確立するには、いくつかの構成プロパティを適合させる必要があります。通信パラメーターのほかに、アダプターによってサポートされているビジネス・オブジェクトは、ブリッジによってサポートされているビジネス・オブジェクトにも対応する必要があります。

イベント処理

次のイベントの表には、ブリッジ構成プロパティとアダプターのコネクタ固有プロパティとの相関を示します。

表 28. イベント処理の通信

アダプターのコネクタ 固有プロパティ	ブリッジ 構成プロパティ (スタンザ EventHandler)	説明
HostName	N/A (WMQChannel の一部)	キュー・マネージャーが存在するホスト。
Port	N/A (WMQChannel の一部)	キュー・マネージャーのリスナーが listen するポート。
Channel	WMQChannel (フォーマット: <チャ ネル>/ <トランスポー ト>/<ホスト名>(<ポー ト>)	キュー・マネージャーとの通信に使用され るチャンネル。
N/A (InputQueue の一部)	WMQQueueManager	ブリッジとアダプターのコネクタ固有プロ パティとの通信に使用されるキュー・ マネージャー。

表 28. イベント処理の通信 (続き)

アダプターのコネクタ 固有プロパティ	ブリッジ 構成プロパティ (スタンザ EventHandler)	説明
InputQueue (フォーマット: キュー:// <キュー・マネージャー >/キュー)	WMQueue	ブリッジとアダプターのコネクタ固有プロパティとの通信に使用されるキュー。

表 29 の例では、一貫性のあるアダプター・アプリケーション構成プロパティとブリッジ構成プロパティを示します。

表 29. プロパティの例

アダプターのコネクタ固有プロパティ	ブリッジ構成プロパティ
HostName=fahost Port=1414 Channel=CHANNEL1 InputQueue=queue://TestQMgr/ FRONTARENA.IN	WMQChannel=CHANNEL1/ TCP/fahost(1414) WMQueueManager=TestQMgr WMQueue=FRONTARENA.IN

要求処理

このセクションでは、要求処理のプロパティを示します。

表 30. 要求処理の通信

アダプターのコネクタ 固有プロパティ	ブリッジ構成プロパティ (スタンザ RequestHandler)	説明
HostName	N/A (WMQChannel の一部)	キュー・マネージャーが存在する ホスト。
Port	N/A (WMQChannel の一部)	キュー・マネージャーのリスナー が listen するポート。
Channel	Channel WMQChannel (フォーマット: <チャンネル>/ < トランスポート>/<ホスト名 >(<ポート>)	キュー・マネージャーとの通信に 使用されるチャンネル。
N/A (キュー名の一部)	WMQueueManager	ブリッジとアダプターのコネクタ 固有プロパティとの通信に使用 されるキュー・マネージャー。
ReplyToQueue (フォーマット: キュー://<キュー・ マネージャー>/キュー)	WMQReplyQueue	ブリッジからアダプターのコネク タ固有プロパティに (要求に 対する) 応答を送信するときに使 用するキュー。

表 31. 属性およびブリッジの構成

ブリッジ構成プロパティ		
静的変換メタオブジェクトで の属性のパラメーター	RequestHandler	説明
OutputQueue (フォーマット: キュー:// <キ ュー・マネージャー>/キュー ー)	WMQRequestQueue	アダプターからブリッジに要求を 送信するときに使用するキュー。

タイムアウト

同期イベント処理に対して WBI アダプターをセットアップすると、アダプターの静的変換メタオブジェクトには、サービス要求に対する応答をアダプターが待機する時間の最大値を定義する `ResponseTimeout` の値が格納されます。ブリッジ構成パラメーター `PollFrequency` の値は、この値よりも大幅に小さくする必要があります。

WebSphere MQ の許可

WebSphere MQ キュー・マネージャーと通信できるようにするには、次のいずれかを適用する必要があります。

- ブリッジの動作時に使用しているユーザー ID は、キュー・マネージャーのアクセスを許可されている必要があります。これを実行するには、次の 2 つの方法があります。
 - このユーザー ID へのアクセス権を実際に認可する。
 - キュー・マネージャーへのアクセスが許可されているユーザー ID の代わりに、ブリッジによってキューにメッセージを書き込む。

注: これを実行するには、2 番目のユーザー ID と通信するときに使用するチャネルのプロパティである `MCA User Id` を設定します。

- 対応する権限を持つユーザー ID を、ブリッジ構成ファイルの `EventHandler` スタンザまたは `RequestHandler` スタンザのいずれかまたは両方のパラメーター `WMQUserID` の値として指定する必要があります。

他のアプリケーションとの通信

アダプターと連携して使用するその他のアプリケーションは、メッセージのレイアウト規則に従う必要があります。

アダプターのブリッジ・コンポーネントによって作成され、受け入れられた WebSphere MQ メッセージは、WebSphere MQ MQMD ヘッダーとその後のメッセージ本文で構成されます。追加のヘッダーまたはメッセージ・エレメントは許可されません。

イベント通知メッセージ

次に示す MQMD ヘッダーのイベント通知メッセージのフィールドは、アダプターのブリッジ・コンポーネントによって、イベント通知メッセージに対して設定されます。

表 32. イベント通知メッセージ

フィールド	説明	値
Format	メッセージ・フォーマット	次の 3 つの値があります。 <ul style="list-style-type: none"> • INSERT。イベントが FRONT ARENA オブジェクトの新規作成を示す場合。 • UPDATE。イベントが既存の FRONT ARENA オブジェクトの変更を示す場合。 • DELETE。イベントが既存の FRONT ARENA オブジェクトの削除を示す場合。
Persistence UserId	メッセージのパーシスタンス メッセージ作成者のユーザー ID	MQPER_AS_Q_DEF ブリッジ構成にユーザー ID が指定されている場合は、この値が挿入されます。そうでない場合は、プログラムを実行している ID が挿入されます。

注: 表 32 に示すように、FORMAT フィールドには WebSphere MQ の事前定義値は指定されていませんが、FRONT ARENA の発信通知を発行する操作のタイプが指定されています。この情報は、ブリッジと連携して使用できる Adapter for FRONT ARENA に必要です。この結果、コード・ページ変換とエンコード変換はサポートされていません。メッセージ本文には、AMB から読み取られた FRONT ARENA メッセージが、変更を加えられることなく格納されています。

要求メッセージ

メッセージ本文には、FRONT ARENA AMB メッセージと、FRONT ARENA アプリケーションが要求を実行するために必要なすべての情報を格納する必要があります。

次に示す MQMD ヘッダーのヘッダー・フィールドは、アダプターのブリッジ・コンポーネントに送信された要求メッセージに対して設定する必要があります。

表 33. ヘッダー・フィールドの説明および値

フィールド	説明	値
MessageId	メッセージ ID	メッセージが作成されたときに WebSphere MQ によって生成される固有 ID。

応答メッセージ

次に示す MQMD ヘッダーのフィールドは、応答メッセージにあるアダプターのブリッジ・コンポーネントによって設定されます。

表 34. ヘッダー・フィールドの説明および値

フィールド	説明	値
CorrelationId	メッセージ相関 ID	対応する要求メッセージのメッセージ ID が入ります。
MessageType	メッセージ・タイプ	MQMT_DIAGRAM
Feedback	フィードバック・コード	後述の表である、表 35 を参照してください。

その他のすべてのヘッダー・フィールドの値は、対応する要求メッセージからコピーされます。ただし、新規に生成されるメッセージ ID は除きます。

表 35. アダプターのフィードバック・コードおよび説明

アダプターのフィードバック・コード	説明
MQFB_PAN	INSERT 要求処理または SELECT 要求処理は正常に実行されました。
MQFB_APPL_FIRST + 2	UPDATE 処理または DELETE 処理は正常に実行されました。
MQFB_APPL_FIRST + 4	SELECT 要求: ヒットはありませんでした。
MQFB_APPL_FIRST + 6	SELECT 要求: 複数のヒットがありました。
MQFB_NAN	要求処理は失敗しました。

注: 表 35 に示したフィードバック・コードは、アダプターによって予想されます。

メッセージ本文の内容は次のとおりです。

- 正常に処理できる、INSERT、UPDATE、DELETE のいずれかの要求の要求メッセージの本文。
- SELECT 要求に対して AMB から受信した FRONT ARENA メッセージ。これによって厳密に 1 ヒットが配信されました。
- それ以外の場合のエラー・メッセージ。

第 7 章 アプリケーションの構成

- 『AMB の構成』
- 『AMBA の構成』

本章では、FRONT ARENA アプリケーションの構成方法について説明します。特に、このアプリケーションの 2 つのコンポーネントである FRONT ARENA Message Broker (AMB) と FRONT ARENA AMB Adapter (AMBA) を構成します。

AMB の構成

ブリッジ・コンポーネントの構成ファイルの EventHandler スタンザと RequestHandler スタンザには、必須プロパティである Name が格納されています。このパラメーターの値は、FRONT ARENA AMB に読み取りプロセスと書き込みプロセスとして接続する場合、クライアント名として使用されます。

この名前は、事前に FRONT ARENA に通知しておく必要があります。AMB データベースのテーブル・システムには、この名前が入った項目が格納されている必要があります。この項目を作成するには、AMB データベースに接続した後に、次のような SQL コマンドを発行します。INSERT システム (名) VALUES (<config.file からのクライアント名)。

AMBA の構成

FRONT ARENA AMB Adapter (AMBA) とは、イベント通知を作成してパブリッシュし、AMB に送信された外部の要求メッセージを処理する FRONT ARENA サブシステムのことです。AMBA は高度な構成が可能であるため、アダプターのブリッジ・コンポーネントと一致するセットアップを実行して、イベント処理と要求処理の両方が機能するようにする必要があります。

注: AMBA パラメーターの設定方法の詳細については、FRONT ARENA アプリケーションの資料を参照してください。

イベント処理

イベント処理では、86 ページの表 36 に示す次の AMBA パラメーターを、事前に定義された方法で設定する必要があります。または、パラメーターの値に、ブリッジの構成パラメーターとの互換性を持たせる必要があります。

表 36. イベント処理のための AMBA パラメーター

AMBA パラメーター	値	説明
-sender_source	AMBA パラメーターの値は、ブリッジ・パラメーターの値である EventHandler/Filter<n>/Topics に指定されているすべてのトピックのプレフィックスである必要があり、ブリッジ・パラメーター EventHandler/Filter<n>/Sources の有効な値になります。	出力メッセージの送信側ソース。すべてのトピックのプレフィックスの形式は、次のようになります。 <sender source>/...
-sender_off	0	AMBA からの出力メッセージを使用不可または使用可能に設定します。
-show_changes	0	生成されるメッセージに差異情報を組み込みます。

要求処理

要求処理では、表 37 に示す次の AMBA パラメーターを、事前に定義された方法で設定する必要があります。または、パラメーターの値に、ブリッジの構成パラメーターとの互換性を持たせる必要があります。

表 37. 要求処理のための AMBA パラメーター

AMBA パラメーター	値	説明
-receiver_sources	ブリッジ・パラメーター RequestHandler/FAPublicationSubject の値は、この AMBA パラメーターの値のいずれかにする必要があります。	listen 先のトピックのプレフィックス。メッセージのトピックが次の形式の場合、このメッセージを受信します。 <receiver source>/... (<receiver source> は、このパラメーターに複数の値が存在する場合、そのいずれかになります)。
-receiver_off	0	メッセージの受信を使用不可または使用可能に設定します。
-db_writes_off	0	着信メッセージの FRONT ARENA データベースへの書き込みを使用不可または使用可能に設定します。
-force_update	1(*)	すべての値に変更がない場合でも、データベース・レコードを更新します。
-show_changes	0(*)	生成されるメッセージに差異情報を組み込みます。
-sender_source	ブリッジ・パラメーター RequestHandler/FAAMBA SenderSource の値は、この AMBA パラメーターの値と同一である必要があります (*)。	出力メッセージの送信側ソース。次の形式を持つすべてのメッセージ・トピックのプレフィックス。 <sender source>/...

表 37. 要求処理のための AMBA パラメーター (続き)

AMBA パラメーター	値	説明
-sender_off	0(*)	AMBA からの出力メッセージを使用不可または使用可能に設定します。

(*) このようにマークされたパラメーターの値は、同期要求処理がサポートされる場合にのみ関係があります。

メッセージ・フォーマット

AMBA によって受け入れられ、パブリッシュされたメッセージの形式は、高度な構成が可能です。AMBA によるメッセージの構成は、外部アプリケーションによって作成され、認識されるメッセージの形式と合致する必要があります。例えば、この構成は、WebSphere Business Integration のシナリオで Adapter for FRONT ARENA が使用するビジネス・オブジェクト定義に対応する必要があります。

第 8 章 アダプターの実行

- 『アダプターの構成』
- 94 ページの『保証付きイベント・デリバリーの使用可能化』
- 99 ページの『キューの URI (Uniform Resource Identifiers)』
- 101 ページの『メタオブジェクトの構成』
- 113 ページの『アダプターの複数インスタンスの作成』
- 115 ページの『始動ファイルの構成』
- 115 ページの『コネクターの始動』
- 117 ページの『コネクターの停止』

この章では、アダプターのインストールと構成の方法、およびコネクターと FRONT ARENA アプリケーションの連動を可能にする方法について説明します。

アダプターの構成

コネクターの構成プロパティには、標準構成プロパティとコネクター固有の構成プロパティの 2 種類があります。一部のプロパティはデフォルト値を持っており、変更を加えなくても使用できます。その他のプロパティについては、アダプターの実行前に値を設定する必要があります。統合ブローカーとして InterChange Server を使用し、コネクター・プロパティを構成するには、Connector Configurator を使用してください。詳細については、147 ページの『付録 B. コネクター固有のプロパティ』を参照してください。

コネクターは、始動時に構成値を取得します。実行時セッション中に、1 つ以上のコネクター・プロパティの値の変更が必要になることがあります。AgentTraceLevel など一部のコネクター構成プロパティへの変更は、即時に有効になります。その他のコネクター・プロパティへの変更を有効にするには、変更後にコンポーネントまたはシステムを再始動する必要があります。プロパティが動的 (即時に有効になる) か、静的 (コネクター・コンポーネントの再始動またはシステムの再始動を必要とする) かどうかを判断するには、147 ページの『付録 B. コネクター固有のプロパティ』を参照してください。

Adapter for FRONT ARENA 用のコネクター・プロパティを構成する場合は、以下の基準が満足されていることを確認します。

- コネクター・プロパティ HostName に対して指定されている値が、ご使用の WebSphere MQ サーバーのホストの名前と一致していること。
- コネクター・プロパティ Port に対して指定されている値が、アダプターの使用するキュー・マネージャーのリスナーに対するポートの値と一致していること。
- コネクター・プロパティ Channel に対して指定されている値が、アダプターの使用するキュー・マネージャーのサーバー接続チャンネルと一致していること。
- キューに対して指定されている値が、キューの作成時に使用した名前と一致していること。

- コネクタ・プロパティ InputQueue、InProgressQueue、ArchiveQueue、ErrorQueue、および UnsubscribeQueue に対するキュー URI が有効で実際に存在すること。詳細については、『コネクタ固有のプロパティ』を参照してください。

標準コネクタ・プロパティ

標準構成プロパティには、コネクタが使用する情報が用意されています。これらのプロパティの詳細については、127 ページの『付録 A. コネクタの標準構成プロパティ』を参照してください。

このアダプターは、統合ブローカーとして InterChange Server しかサポートしていないため、それに関連する構成プロパティは、InterChange Server 用に限られません。

コネクタ固有のプロパティ

コネクタ固有の構成プロパティには、コネクタが実行時に必要とする情報が用意されています。また、エージェントの再コーディングや再ビルドを行わなくても、アダプター内で静的な情報やロジックを変更するための手段も用意されています。

以下の表は、アダプター用のコネクタ固有の構成プロパティをリストしています。プロパティの説明については、以下の各セクションを参照してください。

注: これらのプロパティには、デフォルトのキュー名値が示されています。それらの値は、ご使用のセットアップで実際に使用するキュー名に一致するように変更する必要があります。

表 38. コネクタ固有のプロパティ

名前	指定可能な値	デフォルト値	必須
ApplicationPassword	ログイン・パスワード		いいえ
ApplicationUserName	ログイン・ユーザー ID		いいえ
ArchiveQueue	正常に処理されたメッセージのコピーが送信されるキュー	queue://<queue_manager_name>/WC_MQCONN.ARCHIVE	いいえ
Channel	MQ サーバー・コネクタ・チャンネル		はい
ConfigurationMetaObject	構成メタオブジェクトの名前		はい
DataHandlerClassName	データ・ハンドラー・クラス名	com.crossworlds.DataHandlers.text.xml	いいえ
DataHandlerConfigMO	データ・ハンドラー・メタオブジェクト	MO_DataHandler_Default	はい
DataHandlerMimeType	ファイルの MIME タイプ	text/xml	いいえ
ErrorQueue	未処理のメッセージのキュー	queue://<queue_manager_name>/WC_MQCONN.ERROR	いいえ
FeedbackCodeMappingMO	フィードバック・コード・メタオブジェクト		いいえ
HostName	WebSphere MQ サーバー		はい
InDoubtEvents	FailOnStartup Reprocess Ignore LogError	Reprocess	いいえ

表 38. コネクタ固有のプロパティ (続き)

名前	指定可能な値	デフォルト値	必須
InputQueue	ポーリング・キュー	queue://<queue_manager_name>/ WC_MQCONN.IN	いいえ
InProgressQueue	進行中イベント・キュー	queue://<queue_manager_name>/ WC_MQCONN.IN_PROGRESS	はい
PollQuantity	<i>InputQueue</i> プロパティで指定された各キューから検索するメッセージの数	1	いいえ
Port	<i>WebSphere MQ</i> リスナーのために確立するポート		はい
ReplyToQueue	アダプターからの要求発行時に応答メッセージが配信されるキュー	queue://<queue_manager_name>/ WC_MQCONN.REPLYTO	いいえ
UnsubscribedQueue	アンサブスクライブされたメッセージが送信されるキュー	queue://<queue_manager_name>/ WC_MQCONN.UNSUBSCRIBE	いいえ
UseDefaults	true または false	false	

ApplicationPassword

WebSphere MQ にログインするために UserID と一緒に使用するパスワードです。

デフォルト = なし。

ApplicationPassword が空白のままか、または除去された場合、アダプターは、WebSphere MQ によって提供されるデフォルトのパスワードを使用します。

ApplicationUserName

WebSphere MQ にログインするために、Password と一緒に使用するユーザー ID です。

デフォルト = なし。

ApplicationUserName が空白のままか、または除去された場合、アダプターは、WebSphere MQ によって提供されるデフォルトのユーザー ID を使用します。

ArchiveQueue

正常に処理されたメッセージのコピーが送信されるキューです。

デフォルト = queue://<queue_manager_name>/WC_MQCONN.ARCHIVE

Channel

アダプターが、WebSphere MQ との通信を行うときに経由する MQ サーバー・アダプター・チャンネルです。

デフォルト = なし。

Channel の値が空白のままか、または除去された場合、アダプターは WebSphere MQ によって提供されるデフォルトのサーバー・チャンネルを使用します。

ConfigurationMetaObject

コネクター用の構成情報を含む静的メタオブジェクトの名前です。

デフォルト = なし。

DataHandlerClassName

ビジネス・オブジェクトとの間でのメッセージ変換に使用するデータ・ハンドラー・クラスです。

デフォルト = com.crossworlds.DataHandlers.text.xml

DataHandlerConfigMO

構成情報を提供するためにデータ・ハンドラーに渡されるメタオブジェクトです。

デフォルト = MO_DataHandler_Default

DataHandlerMimeType

これにより、特定の MIME タイプに基づいて、データ・ハンドラーを要求することが可能になります。FRONT ARENA との使用には、XML データ・ハンドラーが必要となります。

デフォルト = text/xml

ErrorQueue

処理されなかったメッセージが送信されるキューです。

デフォルト = queue://<queue_manager_name>/WC_MQCONN.ERROR

FeedbackCodeMappingMO

このプロパティーにより、メッセージの受信を InterChange Server に対して同期応答するために使用されるデフォルトのフィードバック・コードのオーバーライドと再割り当てが可能になります。このプロパティーを使用すると、フィードバック・コードを表示するために各属性名を解釈するときのメタオブジェクトを指定できます。フィードバック・コードの対応する値は、InterChange Server に渡される戻り状況値です。デフォルトのフィードバック・コードのリストについては、14 ページの『同期メッセージ・デリバリー』を参照してください。アダプターは、WebSphere MQ 固有のフィードバック・コードを表す、次の属性値を受け入れます。

- MQFB_APPL_FIRST
- MQFB_APPL_FIRST_OFFSET_N、N は整数 (MQFB_APPL_FIRST + N の値として解釈される)

アダプターは、次の InterChange Server 固有の状況コードを、メタオブジェクトの属性値として受け入れます。

- SUCCESS
- FAIL
- APP_RESPONSE_TIMEOUT
- MULTIPLE_HITS
- UNABLE_TO_LOGIN

- VALCHANGE
- VALDUPES

以下の表は、メタオブジェクトの例を示しています。

表 39. フィードバック・コード・メタオブジェクト属性の例

属性名	デフォルト値
MQFB_APPL_FIRST	SUCCESS
MQFB_APPL_FIRST + 1	FAIL
MQFB_APPL_FIRST + 2	UNABLE_TO_LOGIN

デフォルト = なし。

HostName

WebSphere MQ をホスティングしているサーバーの名前です。

デフォルト = なし。

InDoubtEvents

このプロパティは、予期しないアダプターのシャットダウンにより、処理が完了していない進行中イベントの処理方法を指定します。初期化中に進行中のキューにイベントが見つかった場合に実行するアクションを、以下の 4 つから選択してください。

- FailOnStartup: エラーを記録し、即時にシャットダウンします。
- Reprocess: 残っているイベントを最初に処理し、続いて入力キュー内のメッセージを処理します。
- Ignore: 進行中のキューのメッセージをすべて無視します。
- LogError: エラー・ログを記録しますが、シャットダウンはしません。

デフォルト = Reprocess。

InputQueue

アダプターが新規のメッセージの有無を確認するためにポーリングするメッセージ・キューです。このパラメーターは、FRONT ARENA のブリッジ構成ファイルの EventHandler/WMQQueue パラメーターと同じ値に設定します。

InputQueue プロパティが指定されない場合でも、コネクタは正常に始動しますが、警告メッセージを出力し、要求処理のみを実行します。イベント処理は一切実行しません。

デフォルト= queue://<queue_manager_name>/WC_MQCONN.IN

InProgressQueue

処理中にメッセージが保留されるメッセージ・キューです。

デフォルト= queue://<queue_manager_name>/WC_MQCONN.IN_PROGRESS

PollQuantity

pollForEvents スキャン時に InputQueue プロパティで指定された各キューから取得するメッセージの数。

デフォルト =1

Port

WebSphere MQ リスナーのために確立するポート。

デフォルト = なし。

ReplyToQueue

アダプターからの要求発行時に応答メッセージが配信されるキューです。このパラメーターは、FRONT ARENA のブリッジ構成ファイルの RequestHandler/WMQReplyQueue パラメーターと同じ値に設定します。

デフォルト = queue://<queue_manager_name>/WC_MQCONN.REPLY

UnsubscribedQueue

アンサブスクライブされたメッセージが送信されるキューです。

デフォルト = queue://<queue_manager_name>/WC_MQCONN.UNSUBSCRIBED

注: * WebSphere MQ によって提供される値は誤っていたり不明である可能性があるため、常にチェックする必要があります。値が誤っていたり不明な場合は、値を暗黙的に指定してください。

UseDefaults

Create 操作の場合、UseDefaults を true に設定すると、コネクターは、各 isRequired ビジネス・オブジェクト属性に有効値またはデフォルト値が指定されているかどうかをチェックします。値が指定されている場合、Create 操作は成功します。このパラメーターを false に設定すると、コネクターは有効値の有無だけをチェックし、有効値が指定されていない場合、Create 操作は失敗します。デフォルト値は false です。

保証付きイベント・デリバリーの使用可能化

保証付きイベント・デリバリー機能は、以下の方法のいずれかで、JMS 対応コネクターに対して構成することができます。

- コネクターが (JMS ソース・キューとして実装されている) JMS イベント・ストアを使用している場合は、コネクター・フレームワークが、JMS イベント・ストアを管理できます。詳細については、95 ページの『JMS イベント・ストアによるコネクターの保証付きイベント・デリバリー』を参照してください。
- コネクターが、(例えば、JDBC 表、電子メールのメールボックス、またはフラット・ファイルとして実装されている) 非 JMS イベント・ストアを使用している場合は、コネクター・フレームワークが、JMS モニター・キューを使用して、重複イベントが発生しないようにすることができます。詳細については、97 ページの『非 JMS イベント・ストアによるコネクターの保証付きイベント・デリバリー』を参照してください。

JMS イベント・ストアによるコネクターの保証付きイベント・デリバリー

JMS 対応のコネクターが JMS キューを使用してイベント・ストアを実装している場合、コネクター・フレームワークはコンテナーとして機能して、JMS イベント・ストア (JMS ソース・キュー) を管理できます。コネクターは、1 つの JMS トランザクションで、メッセージをソース・キューから除去し、そのメッセージを宛先キューに入れることができます。このセクションでは、JMS イベント・ストアを備えた JMS 対応コネクターの保証付きイベント・デリバリー機能の使用について、以下の内容を説明します。

- 『JMS イベント・ストアを使用するコネクターでの機能の使用可能化』
- 96 ページの『イベント・ポーリングへの影響』

JMS イベント・ストアを使用するコネクターでの機能の使用可能化

JMS イベント・ストアを備えた JMS 対応コネクターの保証付きイベント・デリバリー機能を使用可能にするには、表 40 に示されている値にコネクター構成プロパティーを設定します。

表 40. JMS イベント・ストアを使用するコネクター用の保証付きイベント・デリバリー・コネクター・プロパティー

コネクター・プロパティー	値
DeliveryTransport	JMS
ContainerManagedEvents	JMS
PollQuantity	イベント・ストアの 1 回のポーリングで処理されるイベントの数。
SourceQueue	JMS ソース・キュー (イベント・ストア) の名前。コネクター・フレームワークはこのキューをポーリングし、処理対象イベントを取り出します。 注: ソース・キューと他の JMS キューは、同じキュー・マネージャーの一部になっていなければなりません。異なるキュー・マネージャーに保管されているイベントをコネクターのアプリケーションで生成する場合は、リモート・キュー・マネージャー用のリモート・キュー定義を定義する必要があります。次に WebSphere MQ は、リモート・キューから取得したイベントを、JMS 対応のコネクターが統合ブローカーへの送信用に使用するキュー・マネージャーに転送します。リモート・キュー定義の構成方法については、「Adapter for WebSphere MQ ユーザーズ・ガイド」の資料を参照してください。

また、コネクターの構成に加えて、JMS ストア内のイベントとビジネス・オブジェクト間の変換を行うデータ・ハンドラーの構成も必要です。このデータ・ハンドラー情報は、表 41 に示すコネクター構成プロパティーから構成されます。

表 41. 保証付きイベント・デリバリー用のデータ・ハンドラー・プロパティー

データ・ハンドラー・プロパティー	値	必須
MimeType	データ・ハンドラーによって処理される MIME タイプ。呼び出し対象となるデータ・ハンドラーは、この MIME タイプで識別されます。	はい
DHClass	データ・ハンドラーを実装する Java クラスのフルネーム。	はい
DataHandlerConfigMOName	MIME タイプとそのデータ・ハンドラーを関連付けるトップレベルのメタオブジェクトの名前。	オプション

注: データ・ハンドラー構成プロパティーは、他のコネクタ構成プロパティーと一緒に、コネクタ構成ファイル内に存在します。

JMS イベント・ストアを備えているコネクタを保証付きイベント・デリバリーを使用するように構成する場合は、表 40 および 表 41 の説明に従って、コネクタ・プロパティーを設定する必要があります。これらのコネクタ構成プロパティーの設定には、Connector Configurator ツールを使用します。Connector Configurator は、表 40 のコネクタ・プロパティーを「標準プロパティー」タブに表示します。また、表 41 のコネクタ・プロパティーを「データ・ハンドラー (Data Handler)」タブに表示します。

注: Connector Configurator は、DeliveryTransport コネクタ構成プロパティーが JMS、および ContainerManagedEvents が JMS にそれぞれ設定されている場合に限り、その「データ・ハンドラー」タブのフィールドをアクティブにします。

Connector Configurator の詳細については、147 ページの『付録 B. コネクタ固有のプロパティー』を参照してください。

イベント・ポーリングへの影響

ContainedManagedEvents を JMS に設定して、保証付きイベント・デリバリーを使用するコネクタの場合は、その機能を使用しないコネクタとは動作が多少異なります。コネクタ・フレームワークは、コンテナ管理のイベントを提供するために、次のステップに従ってイベント・ストアをポーリングします。

1. JMS トランザクションを開始します。
2. イベント・ストアから JMS メッセージを読み取ります。

イベント・ストアは、JMS ソース・キューとして実装されます。JMS メッセージにはイベント・レコードが含まれています。JMS ソース・キューの名前は、SourceQueue コネクタ構成プロパティーから取得されます。

3. イベントをビジネス・オブジェクトに変換するための、データ・ハンドラーを呼び出します。

コネクタ・フレームワークは、表 41 で示されたプロパティーで構成されたデータ・ハンドラーを呼び出します。

4. WebSphere MQ Integrator Broker が統合ブローカーである場合は、構成されているワイヤー・フォーマット (XML) に基づいて、ビジネス・オブジェクトをメッセージに変換します。
5. 変換後のメッセージを JMS 宛先キューに送信します。
 WebSphere InterChange Server 統合ブローカーを使用している場合、JMS 宛先キューに送信されるメッセージは、ビジネス・オブジェクトになります。
 WebSphere MQ Integrator ブローカーを使用している場合、JMS 宛先キューに送信されるメッセージは、(データ・ハンドラーが生成した) XML メッセージになります。
6. JMS トランザクションをコミットします。

 JMS トランザクションがコミットされると、同一のトランザクションで、メッセージが JMS 宛先キューに書き込まれ、JMS リソース・キューから除去されません。
7. ステップ 1 から 6 までのループを繰り返します。このループの繰り返し回数は、PollQuantity コネクタ・プロパティによって決定されます。

重要: ContainerManagedEvents プロパティが JMS に設定されているコネクタは、イベントのポーリングを実行する際に pollForEvents() メソッドを呼び出しません。コネクタの基底クラスに pollForEvents() メソッドが組み込まれている場合、このメソッドは起動されません。

非 JMS イベント・ストアによるコネクタの保証付きイベント・デリバリー

JMS 対応コネクタが、非 JMS ソリューションを使用して、(JDBC イベント表、電子メールのメールボックス、またはフラット・ファイルなどの) そのイベント・ストアを実装している場合は、コネクタ・フレームワークが重複イベント除去を使用して、重複イベントが発生しないようにすることができます。このセクションでは、非 JMS イベント・ストアを使用する JMS 対応コネクタでの保証付きイベント・デリバリー機能の使用について、以下の内容を説明します。

- 『非 JMS イベント・ストアによるコネクタの保証付きイベント・デリバリー機能の使用可能化』
- 96 ページの『イベント・ポーリングへの影響』

非 JMS イベント・ストアによるコネクタの保証付きイベント・デリバリー機能の使用可能化: 非 JMS イベント・ストアを備えた JMS 対応コネクタの保証付きイベント・デリバリー機能を使用可能にするには、表 42 に示されている値にコネクタ構成プロパティを設定する必要があります。

表 42. 非 JMS イベント・ストアを使用するコネクタ用の保証付きイベント・デリバリー・コネクタ・プロパティ

コネクタ・プロパティ	値
DeliveryTransport	JMS
DuplicateEventElimination	true
MonitorQueue	コネクタ・フレームワークが処理対象のビジネス・オブジェクトの ObjectEventId を保管する JMS モニター・キューの名前。

保証付きイベント・デリバリーを使用するようにコネクタを構成する場合は、表 42 の説明に従って、コネクタ・プロパティを設定する必要があります。これらのコネクタ構成プロパティの設定には、Connector Configurator ツールを使用します。このツールは、これらのコネクタ・プロパティを「標準プロパティ (Standard Properties)」タブに表示します。Connector Configurator の詳細については、147 ページの『付録 B. コネクタ固有のプロパティ』を参照してください。

イベント・ポーリングへの影響: DuplicateEventElimination を true に設定して、保証付きイベント・デリバリーを使用するコネクタの場合は、その機能を使用しないコネクタとは動作が多少異なります。重複イベント回避機能を提供するために、コネクタ・フレームワークは JMS モニター・キューを使用してビジネス・オブジェクトを追跡します。JMS モニター・キューの名前は、MonitorQueue コネクタ構成プロパティから取得されます。

コネクタ・フレームワークは、アプリケーション固有のコンポーネントから (pollForEvents() メソッドの getApplEvent() の呼び出しを介して) ビジネス・オブジェクトを受信すると、(getApplEvents() から受信した) 現在のビジネス・オブジェクトが重複イベントであるかどうかを判別する必要があります。このことを判別するために、コネクタ・フレームワークは、JMS モニター・キューからビジネス・オブジェクトを取り出し、その ObjectEventId と現在のビジネス・オブジェクトの ObjectEventId を比較します。

- これら 2 つの ObjectEventId が同一の場合、現在のビジネス・オブジェクトは重複イベントを表しています。この場合、コネクタ・フレームワークは、現行のビジネス・オブジェクトが表すイベントを無視し、このイベントを統合ブローカーに送信しません。
- ObjectEventId が同一でない場合、ビジネス・オブジェクトは重複イベントを表していません。この場合、コネクタ・フレームワークは、現在のビジネス・オブジェクトを JMS モニター・キューにコピーし、これを JMS デリバリー・キューに配信します。これらの一連の処理は、すべて同一の JMS トランザクションの一部として実行されます。JMS デリバリー・キューの名前は、DeliveryQueue コネクタ構成プロパティから取得されます。getApplEvent() メソッドの呼び出し後、コネクタの pollForEvents() メソッドへ制御が戻されます。

JMS 対応のコネクタで重複イベント回避機能サポートするには、コネクタの pollForEvents() メソッドに必ず次のステップを含めるようにしてください。

- 非 JMS イベント・ストアから取得したイベント・レコードからビジネス・オブジェクトを作成する場合、イベント・レコードの固有のイベント ID をビジネス・オブジェクトの ObjectEventId 属性として保管します。

アプリケーションはこのイベント ID を生成し、イベント・ストアのイベント・レコードを一意に識別します。統合ブローカーへのイベント送信後で、しかもこのイベント・レコードの状況が変更可能となる前にコネクタに障害が発生した場合、このイベント・レコードは「進行中 (In-Progress)」状況のままイベント・ストアに残されます。コネクタが復旧した際に、「進行中」のイベントをリカバリーする必要があります。コネクタは、ポーリングを再開すると、イベント・ストアに残っているイベント・レコードのビジネス・オブジェクトを生成します。ただし、すでに送信済みのビジネス・オブジェクトと新規ビジネス・オブジェクトの両方が ObjectEventId として同じイベント・レコードを持っているた

め、コネクタ・フレームワークは新規ビジネス・オブジェクトを重複オブジェクトと認識し、統合ブローカーに送信しない場合があります。

- コネクタのリカバリー時には、コネクタが新規イベントのためのポーリングを開始する前に、「進行中」のイベントを処理するようにしてください。

コネクタの開始時に、「進行中」のイベントが「ポーリング可能」状況に変更されない限り、ポーリング・メソッドは再処理のためにイベント・レコードを受信しません。

キューの URI (Uniform Resource Identifiers)

キューの URI は、シーケンス `queue://` で始まり、それに続いて以下のものが記述されます。

- キューが存在しているキュー・マネージャーの名前
- 別の /
- キューの名前
- 残りのキュー・プロパティを設定する、名前と値のペアのリスト (オプション)。

例えば、次の URI を指定した場合、キュー・マネージャー `<queue.manager.name>` に存在するキュー `IN` に接続し、すべてのメッセージが優先順位 5 の WebSphere MQ メッセージとして送信されます。

```
queue://<queue.manager.name>/WC_MQCONN.IN?targetClient=1&priority=5
```

以下の表に、キュー URI のプロパティ名を示します。

表 43. キュー URI の WebSphere MQ 固有コネクタ・プロパティ名

プロパティ名	説明	値
<code>expiry</code>	ミリ秒で表した、メッセージの存続時間。	0 = 無制限 正の整数 = タイムアウト (ミリ秒単位)。
<code>priority</code>	メッセージの優先順位。	0 から 9 で、1 が最高の優先順位。値 -1 は、このプロパティがキューの構成によって決定されることを意味します。値 -2 は、コネクタ自身のデフォルト値を使用できるよう指定します。
<code>persistence</code>	メッセージをディスクに永続化するかを示します。	1 = 非永続 2 = 永続 値 -1 は、このプロパティがキューの構成によって決定されることを意味します。値 -2 は、コネクタ自身のデフォルト値を使用できるよう指定します。

表 43. キュー URI の WebSphere MQ 固有コネクタ・プロパティ名 (続き)

プロパティ名	説明	値
CCSID	宛先の文字セット。	整数: 有効な値は、「 <i>Adapter for WebSphere MQ ユーザーズ・ガイド</i> 」の資料にリストされています。
targetClient	受信アプリケーションが JMS 対応かそうでないかを示します。	0 = JMS (MQRFH2 ヘッダー) 1 = MQ (MQMD ヘッダーのみ)
encoding	数値フィールドの表現方法を指示します。	整数: 有効な値は、「 <i>Adapter for WebSphere MQ ユーザーズ・ガイド</i> 」の資料にリストされています。

アダプターは、MQMessage 内のデータの文字セット (CCSID) またはエンコード属性を制御できません。データ変換はデータがメッセージ・バッファーから検索されるかメッセージ・バッファーにデリバリーされる時に行われるため、アダプターは JMS の IBM WebSphere MQ インプリメンテーションに依存してデータ変換を行います (IBM WebSphere MQ Java クライアント・ライブラリーの資料を参照してください)。したがって、これらの変換は、ネイティブ WebSphere MQ API がオプション MQGMO_CONVERT を使用して実行する変換と双方向で等しくなければなりません。

アダプターは、変換プロセスにおける差異または失敗を制御できません。アダプターは、特別な変更を必要とせずに、WebSphere MQ によってサポートされるすべての CCSID またはエンコードのメッセージ・データを検索できます。特定の CCSID またはエンコードのメッセージを送信するには、出力キューが完全修飾 URI であり、CCSID および encoding の値を指定している必要があります。アダプターはこの情報を WebSphere MQ に渡し、WebSphere MQ は MQMessage を送達するためにデータをエンコードするときに、この情報を使用します (JMS API を介して)。

多くの場合、CCSID およびエンコードのサポートの欠如は、IBM の Web サイトから最新バージョンの IBM WebSphere MQ Java クライアント・ライブラリーをダウンロードすることによって解決できます。CCSID やエンコードに固有の問題が解決されない場合は、テクニカル・サポートに連絡し、代わりに Java 仮想マシンを使用してアダプターを実行できるかどうかお問い合わせください。

キュー URI の構成

コネクタと使用するキューを構成するには、以下のようにします。

- すべてのキューを URI (Uniform Resource Identifier) として指定します。構文は、以下のとおりです。

```
queue://<InterChangeServerName.queue.manager>/<actual queue>
```

- コネクタ固有の構成プロパティに、キュー・マネージャーのホストを指定します (表 38 を参照)。

注: ターゲット・アプリケーションが MQMD ヘッダーのみを予期していて、JMS クライアントが使用する拡張 MQRFH2 ヘッダーを処理できない場合は、?targetClient=1 をキュー URI に付加します。

メタオブジェクトの構成

コネクターは、メタオブジェクト・エントリーを使用して、メッセージに関連付けるビジネス・オブジェクトを決定します。メッセージの処理に使用されるビジネス・オブジェクトのタイプと動詞は、WebSphere MQ メッセージのヘッダーに含まれる FORMAT フィールドによって決定されます。ビジネス・オブジェクト名と動詞を格納するメタオブジェクト属性を構成し、WebSphere MQ メッセージ・ヘッダーの FORMAT フィールドのテキストに関連付けます。メタオブジェクト属性には、メッセージ処理のガイドラインも含まれます。

入力キューからメッセージが検索されると、コネクターは、FORMAT テキスト・フィールドに関連付けられているビジネス・オブジェクト名を調べます。次に、ビジネス・オブジェクト名とともに、メッセージがデータ・ハンドラーに渡されます。ビジネス・オブジェクトにメッセージの内容が正常に取り込まれると、コネクターはそのビジネス・オブジェクトがサブスクライブされているかどうかをチェックしてから、gotAppEvents() メソッドを使用して統合ブローカーにデリバリーします。

コネクターは、2 種類のメタオブジェクトを認識し、読み取ることができます。

- 静的なコネクター・メタオブジェクト
- 動的な子メタオブジェクト

動的な子メタオブジェクトの属性値は、静的なメタオブジェクトの属性値と重複し、それらをオーバーライドします。

ご自分の実装にはどちらのメタオブジェクトが最適であるかを判断する際には、以下のことを考慮してください。

- **静的メタオブジェクト**
 - 各種メッセージのメタデータがすべて固定されており、かつ構成時に指定可能である場合に役立ちます。
 - ビジネス・オブジェクト・タイプごとにしか値を指定できません。例えば、Customer タイプのオブジェクトのすべてが同一の宛先に送られます。
- **動的メタオブジェクト**
 - ビジネス・プロセスからメッセージ・ヘッダー内の情報にアクセスできます。
 - ビジネス・オブジェクト・タイプに関係なく、実行時にビジネス・プロセスでメッセージの処理を変更できます。例えば、動的メタオブジェクトを使用すると、アダプターに送信された Customer タイプのオブジェクトのそれぞれに、別々の宛先を指定することができます。
 - サポートするビジネス・オブジェクトの構造を変更する必要があります。この変更によって、マップとビジネス・プロセスの変更が必要になる場合があります。
 - カスタム・データ・ハンドラーに変更を加える必要があります。

メタオブジェクト・プロパティ

表 44 に、メタオブジェクトでサポートされるプロパティをすべて含むリストを示します。メタオブジェクトを実装するときには、これらのプロパティの説明を参照してください。メタオブジェクトには、表 44 に示すプロパティが 1 つ以上含まれていなければなりません。

一部のプロパティは、静的メタオブジェクトと動的メタオブジェクトのいずれかでしか使用できません。また、メッセージ・ヘッダーからの読み取りやメッセージ・ヘッダーへの書き込みが不可能なプロパティもあります。特定のプロパティについて、コネクタでどのように解釈および使用されるかを判断するには、イベント処理と要求処理に関する適切なセクションを参照してください。

表 44. WebSphere MQ アダプター・メタオブジェクトのプロパティ

プロパティ名	静的メタオブジェクトで定義可能か	動的メタオブジェクトで定義可能か	説明
CollaborationName	はい	はい	<p>CollaborationName は、ビジネス・オブジェクトと動詞の組み合わせに対する属性のアプリケーション固有のテキスト内で指定される必要があります。例えば、ユーザーが動詞 Create 付きのビジネス・オブジェクト Customer の同期イベントを処理しようとしている場合、静的メタデータ・オブジェクトは Customer_Create という名前の属性を含んでいる必要があります。</p> <p>Customer_Create 属性は、名前と値のペアを含むアプリケーション固有のテキストを含んでいる必要があります。例えば、CollaborationName=MyCustomerProcessingCollab です。構文の詳細については、106 ページの『静的メタオブジェクトの作成の概要』のセクションを参照してください。</p> <p>この条件が満たされていない場合は、コネクタが Customer ビジネス・オブジェクトに関する要求を同期的に処理しようとするランタイム・エラーが発生します。</p> <p>注: このプロパティは、同期要求にのみ利用可能です。</p>

表 44. WebSphere MQ アダプター・メタオブジェクトのプロパティー (続き)

プロパティー名	静的メタオブジェクトで定義可能か	動的メタオブジェクトで定義可能か	説明
DataHandlerConfigMO	はい	はい	構成情報を提供するためにデータ・ハンドラーに渡されるメタオブジェクトです。静的なメタオブジェクトに指定された場合、この値は DataHandlerConfigMO コネクター・プロパティーに指定された値をオーバーライドします。このメタオブジェクト・プロパティーは、さまざまなタイプのビジネス・オブジェクトを処理するために複数の異なるデータ・ハンドラーが必要な場合に使用します。データ形式が実際のビジネス・データに依存する可能性がある場合は、要求処理には動的な子メタオブジェクトを使用します。指定するビジネス・オブジェクトは、コネクター・エージェントでサポートされるものでなければなりません。147 ページの『付録 B. コネクター固有のプロパティー』の説明を参照してください。
DataHandlerMimeType	はい	はい	使用すると、特定の MIME タイプに基づいたデータ・ハンドラーを要求できます。メタオブジェクトに指定されている場合、この値は DataHandlerMimeType コネクター・プロパティーに指定されている値をオーバーライドします。このメタオブジェクト・プロパティーは、さまざまなタイプのビジネス・オブジェクトを処理するために複数の異なるデータ・ハンドラーが必要な場合に使用します。データ形式が実際のビジネス・データに依存する可能性がある場合は、要求処理には動的な子メタオブジェクトを使用します。DataHandlerConfigMO に指定されたビジネス・オブジェクトは、このプロパティーの値に対応する属性を含める必要があります。147 ページの『付録 B. コネクター固有のプロパティー』の説明を参照してください。
DataHandlerClassName	はい	はい	147 ページの『付録 B. コネクター固有のプロパティー』の説明を参照してください。

表 44. WebSphere MQ アダプター・メタオブジェクトのプロパティー (続き)

プロパティー名	静的メタオブジェクトで定義可能か	動的メタオブジェクトで定義可能か	説明
InputFormat	はい	はい	特定のビジネス・オブジェクトに関連付けるインバウンド (イベント) メッセージのフォーマットまたはタイプです。この値は、メッセージ内容の識別に役立つものであり、メッセージを生成するアプリケーションによって決まります。検索されたメッセージがこのフォーマットである場合、そのメッセージは (可能であれば) 特定のビジネス・オブジェクトに変換されます。ビジネス・オブジェクトにこのフォーマットが指定されていない場合、コネクターは特定のビジネス・オブジェクトのサブスクリプション・デリバリーを処理しません。このプロパティーを設定するときは、デフォルトのメタオブジェクト変換プロパティーを使用しないでください。デフォルトのメタオブジェクト変換プロパティーの値は、着信メッセージをビジネス・オブジェクトに一致させるために使用されます。コネクターがメッセージのフォーマットを定義していると見なすフィールドは、コネクター固有のプロパティー <code>MessageFormatProperty</code> を使用してユーザーが定義できます。
OutputFormat	はい	はい	アウトバウンド・メッセージに取り込まれるフォーマットです。OutputFormat が指定されていない場合、使用可能であれば入力フォーマットが使用されます。
InputQueue	はい	はい	コネクターが、新しいメッセージを検出するためにポーリングする入力キュー。このプロパティーは、着信メッセージとビジネス・オブジェクトを一致させる目的でのみ使用されます。このプロパティーを設定するときは、デフォルトの変換プロパティーを使用しないでください。デフォルトの変換プロパティーの値は、着信メッセージをビジネス・オブジェクトに一致させるために使用されます。 注: コネクター固有のプロパティーにある <code>InputQueue</code> プロパティーは、アダプターがポーリングするキューを定義します。これは、アダプターがポーリングするキューを決定するのに使用する唯一のプロパティーです。静的 MO では、InputQueue プロパティーおよび InputFormat プロパティーは、アダプターが指定されたメッセージを特定のビジネス・オブジェクトにマップする条件として使用できます。この機能を実装するには、コネクター固有のプロパティーを使用して複数の入力宛先を構成し、さらに、必要に応じて、着信メッセージの入力フォーマットを基に各入力宛先に別個のデータ・ハンドラーをマップします。詳細については、108 ページの『データ・ハンドラーの入力キューへのマッピングの概要』を参照してください。
OutputQueue	はい	はい	特定のビジネス・オブジェクトから派生したメッセージが送信されるキューです。

表 44. WebSphere MQ アダプター・メタオブジェクトのプロパティー (続き)

プロパティー名	静的メタオブジェクトで定義可能か	動的メタオブジェクトで定義可能か	説明
ResponseTimeout	はい	はい	同期要求処理で応答を待機するときにタイムアウトとするまでの待機時間を、ミリ秒単位で示します。このプロパティーが未定義のままか、またはゼロよりも小さい値に設定されている場合、コネクタは応答を待機せず、SUCCESS を即時に戻します。
TimeoutFatal	はい	はい	同期要求処理で、応答の受信がないためコネクタからエラー・メッセージを戻す動作が起動される時に使用されます。このプロパティーの値が True の場合、コネクタは、ResponseTimeout に指定されている時間内に応答の受信がなければ、APPRESPONSETIMEOUT をブローカーに戻します。このプロパティーが未定義の場合、または False に設定されている場合、コネクタは応答タイムアウトが発生すると要求を失敗させます。ただし、終了させることはありません。デフォルト値は False です。
DataEncoding			DataEncoding は、メッセージの読み取りおよび書き込みに使用されるエンコードです。このプロパティーが静的メタオブジェクトで指定されていない場合、コネクタは特定のエンコードを使用せずにメッセージを読み取ろうとします。動的子メタオブジェクトで定義された DataEncoding は、静的メタオブジェクトで定義された値をオーバーライドします。デフォルト値は Text です。この属性の値のフォーマットは、messageType[:enc] です。例えば、Text:ISO8859_1、Text:UnicodeLittle、Text、または Binary のようになります。このプロパティーは内部的に InputFormat プロパティーに関連します。InputFormat ごとに 1 つの DataEncoding のみを指定します。
<p>以下に示すのは、JMS メッセージ・ヘッダーのみにマップされるフィールドです。具体的な説明や値の解釈などの詳細情報については、JMS API の仕様を参照してください。フィールドによっては、JMS プロバイダー間で解釈が異なることがあります。ご使用の JMS プロバイダーの資料を参照して、解釈の違いがないかどうかを確認してください。</p>			
ReplyToQueue		はい	要求に対する応答メッセージの送信先となるキューです。
Type		はい	メッセージのタイプ。通常はユーザーがメッセージのタイプを定義できます (JMS プロバイダーによって異なります)。
MessageID		はい	メッセージの固有 ID (JMS プロバイダー固有) です。
CorrelationID	はい	はい	応答メッセージで、その応答の開始理由にあたる要求メッセージの ID を示すために使用されます。
Delivery Mode	はい	はい	メッセージを MOM システム内で永続化するかどうかを指定します。許容値には、以下のものがあります。 1 = 非永続 2 = 永続 その他の値も使用できる場合があります (JMS プロバイダーによって異なります)。

表 44. WebSphere MQ アダプター・メタオブジェクトのプロパティー (続き)

プロパティー名	静的メタオブジェクトで定義可能か	動的メタオブジェクトで定義可能か	説明
Priority		はい	メッセージの優先順位を数値で表現したものです。許容値は、(両端を含め) 0 から 9 (低い優先順位から高い優先順位) です。
Destination		はい	MOM システムでのメッセージの現在または最後 (削除された場合) の位置です。
Redelivered		はい	以前に JMS プロバイダーからクライアントに対してメッセージのデリバリーが試行されたが、受取の確認がなかった可能性が高いことを示すものです。
Timestamp		はい	時刻メッセージが JMS プロバイダーに引き渡されたことを示します。
UserID		はい	メッセージを送信したユーザーの ID です。
AppID		はい	メッセージを送信したアプリケーションの ID です。
DeliveryCount		はい	デリバリーを試行した回数です。
GroupID		はい	メッセージ・グループの ID です。
GroupSeq		はい	GroupID に指定されたメッセージ・グループにおける順位です。
JMSProperties		はい	111 ページの『JMS プロパティー』を参照してください。

静的メタオブジェクトの作成の概要

WebSphere MQ アダプター構成のメタオブジェクトは、さまざまなビジネス・オブジェクト用に定義された変換プロパティーのリストで構成されます。静的メタオブジェクトの例を表示するには、Business Object Designer を起動して、アダプター connectors¥WebSphereMQ¥samples¥LegacyContact¥WebSphereMQ_MO_Config.xsd に同梱の以下の例を開きます。

コネクターがサポートする静的メタオブジェクトは、常に 1 つだけです。静的メタオブジェクトを実装するには、そのメタオブジェクトの名前を、コネクター・プロパティー ConfigurationMetaObject に指定します。

静的メタオブジェクトは、各属性が、それぞれ 1 つのビジネス・オブジェクトと動詞の組み合わせを、そのオブジェクトの処理に関連するメタデータのすべてとともに示す構造になっています。各属性の名前は、Customer_Create のように、ビジネス・オブジェクト・タイプと動詞の間を下線で区切った名前にする必要があります。属性のアプリケーション固有情報には、このオブジェクトと動詞の固有の組み合わせに対して指定するメタデータ・プロパティーを表す名前と値のペアを、セミコロンで区切って 1 つ以上含める必要があります。

表 45. 静的メタオブジェクト構造

属性名	アプリケーション固有のテキスト
<business object type>_<verb>	property=value;property=value;...
<business object type>_<verb>	property=value;property=value;...

例えば、次のようなメタオブジェクトがあるとします。

表 46. 静的メタオブジェクト構造の例

属性名	アプリケーション固有情報
Customer_Create	OutputFormat=CUST;OutputDestination=QueueA
Customer_Update	OutputFormat=CUST;OutputDestination=QueueB
Order_Create	OutputFormat=ORDER;OutputDestination=QueueC

この例のメタオブジェクトは、コネクタに対して、動詞 Create を持つタイプ Customer の要求ビジネス・オブジェクトを受信したときに、そのビジネス・オブジェクトをフォーマット CUST のメッセージに変換した後、宛先の QueueA に置くように指示します。Customer タイプのオブジェクトの動詞が Update である場合、変換後のメッセージは QueueB に入れます。オブジェクトのタイプが Order で動詞が Create である場合には、コネクタはそのオブジェクトをフォーマットが ORDER のメッセージに変換し、QueueC にデリバリーします。それ以外のビジネス・オブジェクトがコネクタに渡された場合には、アンサブスクライブされているものとして処理されます。

また、Default 属性を 1 つ指定して、アプリケーション固有情報 (ASI) のプロパティを 1 つ以上割り当てることもできます。このデフォルト属性のプロパティと、オブジェクトと動詞の組み合わせごとの属性のプロパティが結合されて、メタオブジェクトの属性の最終的なプロパティになります。オブジェクトと動詞の組み合わせに関係なく汎用的に適用するプロパティが 1 つ以上ある場合には、このデフォルト属性を使用すると便利です。次の例の場合、コネクタは、オブジェクトと動詞の組み合わせ Customer_Create と Order_Create については、それぞれに個別に指定されているメタデータ・プロパティに加えて OutputDestination=QueueA が指定されていると見なします。

表 47. 静的メタオブジェクト構造の例

属性名	アプリケーション固有情報
Default	OutputDestination=QueueA
Customer_Update	OutputFormat=CUST
Order_Create	OutputFormat=ORDER

静的メタオブジェクトにアプリケーション固有情報として指定できるプロパティは、102 ページの表 44 に記載されています。

注: 静的なメタオブジェクトが指定されていない場合、コネクタはポーリング中にある特定のメッセージ・フォーマットを特定のビジネス・オブジェクト・タイプにマップできません。この場合、コネクタはビジネス・オブジェクトを指定せずに、メッセージ・テキストを構成済みのデータ・ハンドラーに渡します。データ・ハンドラーがテキストのみに基づいたビジネス・オブジェクトを作成できない場合、コネクタはこのメッセージ・フォーマットが認識されていないことを表すエラーを報告します。

静的メタオブジェクトの構成

静的メタオブジェクトは、ユーザーがビジネス・オブジェクトに関して指定したアプリケーション固有の情報と、コネクタによるビジネス・オブジェクトの処理方法についての情報を格納します。静的メタオブジェクトは、コネクタに、ビジネス・オブジェクトを処理するために必要なすべての情報を、コネクタの始動時に提供します。

実装の時点で、それぞれのビジネス・オブジェクトの送信先であるキューが判明している場合は、静的メタオブジェクトを使用してください。静的メタオブジェクトの作成および構成を行う場合は、コネクタがその静的メタオブジェクトを確実にサブスクライブするように、コネクタ固有プロパティ `DataHandlerConfigMO` にその静的メタオブジェクトの名前を指定します。詳細については、90 ページの『コネクタ固有のプロパティ』を参照してください。

静的メタオブジェクトの作成のステップ

1. Business Object Designer を起動します。詳細については、「ビジネス・オブジェクト開発ガイド」を参照してください。
2. メタオブジェクトの例
`connectors\WBIMB\samples\LegacyItem\Sample_WBIMB_MO_Config.xsd` を開きます。
3. 必要な要件が反映されるように属性と ASI を編集して (102 ページの表 44 を参照)、メタオブジェクト・ファイルを保管します。
4. 保管したメタオブジェクト・ファイルの名前を、コネクタ・プロパティ `ConfigurationMetaObject` の値に指定します。

データ・ハンドラーの入力キューへのマッピングの概要

静的メタオブジェクトのアプリケーション固有情報で `InputQueue` プロパティを使用することにより、データ・ハンドラーと入力キューを関連付けることができます。この機能は、異なる書式や変換要件を持つ複数の取引先と取り引きする場合に役立ちます。

データ・ハンドラーの入力キューへのマッピングのステップ

`InputQueue` にデータ・ハンドラーをマップするには、以下のステップを実行します。

1. コネクタ固有のプロパティ (93 ページの『`InputQueue`』参照) を使用して 1 つ以上の入力キューを構成します。
2. Business Object Designer で静的メタオブジェクトを開きます。
3. 静的メタオブジェクト内の各入力キューに対して、以下を指定します。
 - キュー・マネージャー
 - 入力キュー名
 - データ・ハンドラー・クラス名
 - アプリケーション固有情報内の MIME タイプ

例えば、次に示す静的メタオブジェクトの属性は、データ・ハンドラーと、`CompReceipts` という名前の `InputQueue` を関連付けています。

```

[Attribute]
Name = Cust_Create
Type = String
Cardinality = 1
MaxLength = 1
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = InputQueue=//queue.manager/CompReceipts;DataHandlerClassName=
com.crossworlds.DataHandlers.WBIMB.disposition_notification;DataHandlerMimeType=
message/
disposition_notification
IsRequiredServerBound = false
[End]

```

動的子メタオブジェクトの作成の概要

コネクタがシナリオに応じて異なるビジネス・オブジェクト処理を実行する必要がある場合は、動的メタオブジェクトを使用します。これは、ビジネス・オブジェクトに追加する子オブジェクトです。動的メタオブジェクトは、実行時に要求の処理方法をコネクタに指示します。静的メタオブジェクトは、コネクタがビジネス・オブジェクトを処理するために必要なすべての情報を、コネクタに提供します。これに対して、動的メタオブジェクトは、特定のシナリオの処理を実行するために必要なロジックの追加部分だけを提供します。

動的メタオブジェクトを使用すると、要求処理では、コネクタがビジネス・オブジェクトの処理に使用するメタデータを要求ごとに変更できるようになります。また、イベント処理では、イベント・メッセージに関する情報の検索が可能になります。

コネクタは、コネクタに渡されるトップレベルのビジネス・オブジェクトに子として追加された動的メタオブジェクトから、変換プロパティを認識して読み取ります。この動的な子メタオブジェクトの属性値は、コネクタの構成に使用される静的なメタオブジェクトに指定可能であった変換プロパティと重複します。

動的な子メタオブジェクトのプロパティは静的なメタオブジェクトから検出されるプロパティをオーバーライドするため、動的な子メタオブジェクトを指定する場合は、静的なメタオブジェクトを指定するコネクタ・プロパティを組み込む必要はありません。したがって、動的な子メタオブジェクトは、静的なメタオブジェクトとは無関係に使用することができ、その逆もまた同様です。

動的メタオブジェクトにアプリケーション固有情報として指定できるプロパティは、102 ページの表 44 に記載されています。

動的メタオブジェクトは、各属性がそれぞれ 1 つのメタデータ・プロパティと値を meta-object property name =meta-object property value の形式で表す構造になっています。

注: IBM WebSphere の標準のデータ・ハンドラーは、いずれも、cw_mo_ タグを認識すると、その後指定されている動的メタオブジェクトを表す属性をビジネス・データ用の属性として処理しないように設計されています。アダプターで使用するカスタム・データ・ハンドラーを開発するときは、同様に設計する必要があります。

動的メタオブジェクトの作成および構成

動的メタオブジェクトを作成および構成するには、次のステップを実行します。

- 動的メタオブジェクトを作成し、それを子オブジェクトとして要求ビジネス・オブジェクトに追加します。
- ターゲット・キュー、メッセージ・フォーマット、その他のタスクなどの情報を動的メタオブジェクトに取り込む追加ロジックで、コラボレーションをプログラムしてから、コネクターにコラボレーションを発行します。

コネクターは動的メタオブジェクトをチェックし、その情報を使用してビジネス・オブジェクトの処理方法を判別します。

ポーリング時の動的子メタオブジェクトの取り込み

ポーリング中に検索されたメッセージについてさらに詳しい情報をコラボレーションに提供するため、コネクターは、作成されたビジネス・オブジェクトに動的なメタオブジェクトが定義済みである場合、その特定の属性に値を取り込みます。

表 48 に、動的子メタオブジェクトがポーリングのためにどのように構成されるかを示します。

表 48. ポーリング用の動的子メタオブジェクトの構造

プロパティ名	値の例
InputFormat	CUST_IN
InputQueue	MYInputQueue
OutputFormat	CxIgnore
OutputQueue	CxIgnore
ResponseTimeout	CxIgnore
TimeoutFatal	CxIgnore

表 48 に示すように、動的子メタオブジェクトに、追加属性 `Input_Format` および `InputQueue` を定義することができます。`Input_Format` には検索されたメッセージのフォーマットが取り込まれ、`InputQueue` 属性には特定のメッセージが検索されたキューの名前が含まれます。これらのプロパティが子メタオブジェクトで定義されていない場合は、これらのプロパティは取り込まれません。

シナリオ例:

- コネクターは、キュー `MyInputQueue` からフォーマット `CUST_IN` でメッセージを取得します。
- コネクターはこのメッセージを `Customer` ビジネス・オブジェクトに変換し、アプリケーション固有のテキストを調べてメタオブジェクトが定義されているかどうかを判断します。
- メタオブジェクトが定義されている場合、コネクターはこのメタオブジェクトのインスタンスを作成し、定義に基づいて `InputQueue` および `InputFormat` 属性に値を取り込んで、ビジネス・オブジェクトを使用可能なコラボレーションにパブリッシュします。

JMS ヘッダーと動的子メタオブジェクトの属性

動的メタオブジェクトに属性を追加すると、メッセージ・トランスポートの詳細情報を取得したりメッセージ・トランスポートを詳細に制御したりすることができます。このセクションでは、これらの属性、および属性がイベント通知と要求処理に及ぼす影響について説明します。

JMS プロパティ: 動的メタオブジェクトの他の属性と異なり、JMSProperties は単一カーディナリティー子オブジェクトを定義する必要があります。この子オブジェクトの各属性は、以下のように JMS メッセージ・ヘッダーの可変部分で読み取り/書き込みを行う単一プロパティを定義する必要があります。

1. 属性の名前はセマンティック値を持ちません。
2. 属性のタイプは、JMS プロパティ・タイプに無関係に必ず String でなければなりません。
3. 属性のアプリケーション固有情報は、属性をマップする JMS メッセージ・プロパティの名前と形式を定義する 2 つの名前と値の組を含まなければなりません。名前はユーザーが定義できます。値の型は次のいずれかでなければなりません。
 - Boolean
 - String
 - Int
 - Float
 - Double
 - Long
 - Short
 - Byte

以下の表に、JMSProperties オブジェクトの属性に対して定義する必要があるアプリケーション固有情報プロパティを示します。

表 49. JMS プロパティ属性のアプリケーション固有情報

属性	指定可能な値	ASI	コメント
名前	任意の有効な JMS プロパティ名 (有効とは、プロパティの型と ASI に定義した型が矛盾しないこと)	<code>name=<JMS プロパティ名>;type=<JMS プロパティの型></code>	ベンダーによっては、拡張機能を提供するために特定のプロパティを予約している場合があります。一般に、ユーザーはベンダー固有の機能にアクセスする場合以外は、JMS で開始するカスタム・プロパティを定義してはなりません。

表 49. JMS プロパティ属性のアプリケーション固有情報 (続き)

属性	指定可能な値	ASI	コメント
タイプ	String	type=<コメントを参照>	これは JMS プロパティのタイプです。 JMS API は、JMS メッセージに値を設定するための多くのメソッドを提供します (例: setIntProperty、setLongProperty、setStringProperty)。 ここで指定される JMS プロパティのタイプによって、これらのどのメソッドを使用してメッセージのプロパティ値を設定するかが決まります。

次の例では、Customer オブジェクトに JMSProperties 子オブジェクトを定義して、メッセージ・ヘッダーのユーザー定義フィールドにアクセスできるようにしています。

```
Customer (ASI = cw_mo_conn=MetaData)
  -- Id
  -- FirstName
  -- LastName
  -- ContactInfo
  -- MetaData
    -- OutputFormat = CUST
    -- OutputDestination = QueueA
    -- JMSProperties
      -- RoutingCode = 123 (ASI= name=RoutingCode;type=Int)
      -- Dept = FD (ASI= name=RoutingDept;type=String)
```

もう 1 つの例として、図 6 に、動的メタオブジェクトに含まれる属性 JMSProperties と、JMS メッセージ・ヘッダーの 4 つのプロパティ (ID、GID、RESPONSE、および RESPONSE_PERSIST) の定義を示します。属性のアプリケーション固有情報はそれぞれの名前およびタイプを定義します。例えば、属性 ID はタイプ String の JMS プロパティ ID にマップされます。

	Pos	Name	Type	Key	Reqd	Card	App Spec Info
1	1	JMSProperties	TeamCenter_JMS_Properties	<input type="checkbox"/>	<input type="checkbox"/>	1	
1.1	1.1	ID	String	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		name=ID,type=String
1.2	1.2	GID	String	<input type="checkbox"/>	<input checked="" type="checkbox"/>		name=GID,type=String
1.3	1.3	RESPONSE	String	<input type="checkbox"/>	<input checked="" type="checkbox"/>		name=RESPONSE,type=Boolean
1.4	1.4	RESP_PERSIST	String	<input type="checkbox"/>	<input checked="" type="checkbox"/>		name=RESPONSE_PERSIST,type=Boolean
1.5	1.5	ObjectEventId	String				
2	2	OutputFormat	String	<input checked="" type="checkbox"/>	<input type="checkbox"/>		

図 6. 動的メタオブジェクトの JMS プロパティ属性

動的メタオブジェクトの作成のステップ

動的メタオブジェクトを作成するには、次のステップを実行します。

1. **Business Object Designer** を起動します。詳細については、「ビジネス・オブジェクト開発ガイド」を参照してください。
2. 処理するときに動的メタオブジェクトを作用させる必要があるトップレベル・ビジネス・オブジェクトを開きます。
3. 開いたトップレベル・オブジェクトに動的メタオブジェクトを子オブジェクトとして追加し、名前と値のペア `cw_mo_conn=<MO attribute>` をそのトップレベル・オブジェクトの ASI に追加します。ここで、`<MO attribute>` は、そのトップレベル・オブジェクトに含まれる、動的メタオブジェクトを表現する属性の名前です。例えば、次のようになります。

```
Customer (ASI = cw_mo_conn=MetaData)
  -- Id
  -- FirstName
  -- LastName
  -- ContactInfo
  -- MetaData
    -- OutputFormat = CUST
    -- OutputDestination = QueueA
```

コネクタは、上の定義にあてはまる要求を受信すると、その要求 (Customer オブジェクト) をフォーマットが CUST のメッセージに変換し、キュー QueueA にそのメッセージを入れます。

4. トップレベル・ビジネス・オブジェクトを保管します。

注: 複数のビジネス・オブジェクトで同じ動的メタオブジェクトを使用することも、それぞれで異なる動的メタオブジェクトを使用することもできます。また、動的メタオブジェクトを一切使用しなくてもかまいません。

アダプターの複数インスタンスの作成

コネクタの複数のインスタンスを作成する作業は、いろいろな意味で、カスタム・コネクタの作成と同じです。以下に示すステップを実行することによって、コネクタの複数のインスタンスを作成して実行するように、ご使用のシステムを設定することができます。次のようにする必要があります。

- コネクタ・インスタンス用に新規ディレクトリを作成します。
- 必要なビジネス・オブジェクト定義が設定されていることを確認します。
- 新規コネクタ定義ファイルを作成します。
- 新規始動スクリプトを作成します。

新規ディレクトリの作成

それぞれのコネクタ・インスタンスごとにコネクタ・ディレクトリを作成する必要があります。このコネクタ・ディレクトリには、次の名前を付けなければなりません。

```
ProductDir%connectors%connectorInstance
```

ここで `connectorInstance` は、コネクタ・インスタンスを一意的に示します。

コネクタに、コネクタ固有のメタオブジェクトがある場合、コネクタ・インスタンス用のメタオブジェクトを作成する必要があります。メタオブジェクトをファイルとして保管する場合は、次のディレクトリを作成して、ファイルをそこに格納します。

```
ProductDir¥repository¥connectorInstance
```

ビジネス・オブジェクト定義の作成

各コネクタ・インスタンスのビジネス・オブジェクト定義がプロジェクト内にまだ存在しない場合は、それらを作成する必要があります。

1. 初期コネクタに関連付けられているビジネス・オブジェクト定義を変更する必要がある場合は、適切なファイルをコピーし、**Business Object Designer** を使用してそれらのファイルをインポートします。初期コネクタの任意のファイルをコピーできます。変更を加えた場合は、名前を変更してください。
2. 初期コネクタのファイルは、次のディレクトリに入っていないとなりません。

```
ProductDir¥repository¥initialConnectorInstance
```

作成した追加ファイルは、`ProductDir¥repository` の適切な `connectorInstance` サブディレクトリ内に存在する必要があります。

コネクタ定義の作成

Connector Configurator 内で、コネクタ・インスタンスの構成ファイル (コネクタ定義) を作成します。これを行うには、以下のステップを実行します。

1. 初期コネクタの構成ファイル (コネクタ定義) をコピーし、名前変更します。
2. 各コネクタ・インスタンスが、サポートされるビジネス・オブジェクト (および関連メタオブジェクト) を正しくリストしていることを確認します。
3. 必要に応じて、コネクタ・プロパティをカスタマイズします。

始動スクリプトの作成

始動スクリプトは以下のように作成します。

1. 初期コネクタの始動スクリプトをコピーし、コネクタ・ディレクトリの名前を含む名前を付けます。

```
dirname
```

2. この始動スクリプトを、113 ページの『新規ディレクトリの作成』で作成したコネクタ・ディレクトリに格納します。
3. 始動スクリプトのショートカットを作成します (Windows のみ)。
4. 初期コネクタのショートカット・テキストをコピーし、新規コネクタ・インスタンスの名前に一致するように (コマンド行で) 初期コネクタの名前を変更します。

これで、ご使用の統合サーバー上でコネクタの両方のインスタンスを同時に実行することができます。

カスタム・コネクタ作成の詳細については、「コネクタ開発ガイド (C++ 用)」または「コネクタ開発ガイド (Java 用)」を参照してください。

始動ファイルの構成

Adapter for FRONT ARENA を始動する前に、始動ファイルを構成する必要があります。クライアント・ライブラリーの場所を指すように、start_connector スクリプトを必ず変更します。複数のバージョン、または現在使用している WebSphere MQ サーバーに対応しないバージョンのクライアント・ライブラリーをインストールしないようにしてください。

Windows

Windows プラットフォーム用にアダプターの構成を実行するには、始動ファイル (アダプターに添付されている start_WebSphereMQ.bat または start_WebSphereCommerce.bat) を以下のように変更する必要があります。

1. start_WebSphereCommerceAdapter.bat ファイルを開きます。
2. スクロールして「Set the directory containing your WebSphere MQ Java client libraries」で始まるセクションに移動し、WebSphere MQ Java クライアント・ライブラリーの場所を指定します。
3. パラメーター JCLASSES にスクロールし、行の最後に以下を追加します。
;%CROSSWORLDS%¥DataHandlers¥bia_FrontArenaDataHandler.jar

UNIX

UNIX プラットフォーム用にアダプターの構成を実行するには、始動ファイル (アダプターに添付されている start_WebSphereCommerceAdapter.sh または start_WebSphereCommerce.sh) を以下のように変更する必要があります。

1. start_WebSphereCommerceAdapter.sh ファイルを開きます。
2. スクロールして「Set the directory containing your WebSphere MQ Java client libraries」で始まるセクションに移動し、WebSphere MQ Java クライアント・ライブラリーの場所を指定します。
3. パラメーター CLASSPATH にスクロールし、行の最後に以下を追加します。
:\${CROSSWORLDS}/DataHandlers/bia_FrontArenaDataHandler.jar

コネクタの始動

コネクタは、コネクタ始動スクリプトを使用して明示的に始動する必要があります。始動スクリプトは、次に示すようなコネクタのランタイム・ディレクトリに存在していなければなりません。

```
ProductDir¥connectors¥connName
```

ここで、connName はコネクタを示します。始動スクリプトの名前は、表 50 に示すように、オペレーティング・システム・プラットフォームによって異なります。

表 50. コネクタの始動スクリプト

オペレーティング・システム	始動スクリプト
UNIX ベースのシステム	connector_manager_connName

表 50. コネクターの始動スクリプト (続き)

オペレーティング・システム	始動スクリプト
Windows	start_connName.bat

コネクタ始動スクリプトは、以下に示すいずれかの方法で起動することができます。

- Windows システムで「スタート」メニューから。

「プログラム」>「IBM WebSphere Business Integration Adapters」>「アダプター」>「コネクタ」を選択します。デフォルトでは、プログラム名は「IBM WebSphere Business Integration Adapters」となっています。ただし、これはカスタマイズすることができます。あるいは、ご使用のコネクタへのデスクトップ・ショートカットを作成することもできます。

- コマンド行から。

– Windows システム:

```
start_connName connName brokerName [-cconfigFile ]
```

– UNIX ベースのシステム:

```
connector_manager_connName -start
```

ここで、*connName* はコネクタの名前であり、*brokerName* は以下のご使用の統合ブローカーを表します。

- WebSphere InterChange Server の場合は、*brokerName* に ICS インスタンスの名前を指定します。
- WebSphere Message Brokers (WebSphere MQ Integrator、WebSphere MQ Integrator Broker、または WebSphere Business Integration Message Broker) または WebSphere Application Server の場合は、*brokerName* にブローカーを示すストリングを指定します。

注: Windows システム上の WebSphere Message Broker または WebSphere Application Server の場合は、*-c* オプションに続いてコネクタ構成ファイルの名前を指定しなければなりません。ICS の場合は、*-c* はオプションです。

- Adapter Monitor から (WebSphere Business Integration Adapters 製品のみ)。
Adapter Monitor は System Manager 始動時に起動されます。

このツールを使用して、コネクタのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- System Monitor から (WebSphere InterChange Server 製品のみ)。

このツールを使用して、コネクタのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- Windows システムでは、Windows サービスとして始動するようにコネクタを構成することができます。この場合、Windows システムがブートしたとき (自動サービスの場合)、または Windows サービス・ウィンドウを通じてサービスを始動したとき (手動サービスの場合) に、コネクタが始動します。

コマンド行の始動オプションなどのコネクターの始動方法の詳細については、以下の資料のいずれかを参照してください。

- WebSphere InterChange Server については、「システム管理ガイド」を参照してください。
- WebSphere Message Brokers については、「*WebSphere Message Brokers 使用アダプター・インプリメンテーション・ガイド*」を参照してください。
- WebSphere Application Server については、「*アダプター実装ガイド (WebSphere Application Server)*」を参照してください。

コネクターの停止

コネクターを停止する方法は、以下に示すように、コネクターが始動された方法によって異なります。

- コマンド行からコネクターを始動した場合は、コネクター始動スクリプトを用いて、以下の操作を実行します。
 - Windows システムでは、始動スクリプトを起動すると、そのコネクター用の別個の「コンソール」ウィンドウが作成されます。このウィンドウで、「Q」と入力して Enter キーを押すと、コネクターが停止します。
 - UNIX ベースのシステムでは、コネクターはバックグラウンドで実行されるため、別ウィンドウはありません。代わりに、次のコマンドを実行してコネクターを停止します。

```
connector_manager_connName -stop
```

ここで、*connName* はコネクターの名前です。

- Adapter Monitor から (WebSphere Business Integration Adapters 製品のみ)。Adapter Monitor は System Manager 始動時に起動されます。

このツールを使用して、コネクターのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- System Monitor から (WebSphere InterChange Server 製品のみ)

このツールを使用して、コネクターのロード、アクティブ化、非アクティブ化、休止、シャットダウン、または削除を行うことができます。

- Windows システムでは、Windows サービスとして始動するようにコネクターを構成することができます。この場合、Windows システムのシャットダウン時に、コネクターは停止します。

ブリッジの始動

FRONT ARENA 実行可能ファイル用のブリッジ *bia_bfa* は、始動するときに 1 つのコマンド行パラメーター (使用する構成ファイルの完全修飾名) を取ります。

アプリケーションは、以下を実行して始動します。

- Windows の場合: *bia_bfa config.file* と入力します。
- Solaris の場合:
 1. *start_bfa.sh* を編集して、*exec* 行に構成ファイルの名前を指定します。

2. CLASSPATH を編集して、CLASSPATH 行に `bia_BFAAlert.jar` を追加します。
3. `start_bfa.sh` を実行します。

構成ファイルが存在しないか、アクセスできない場合、ブリッジは、エラー・メッセージを表示した後、すぐに終了します。構成ファイルで、必須プロパティのいずれかが欠落または無効な場合も、同じ処理になります。

ブリッジの始動が成功すると、その肯定証明が、コマンド行メッセージ「アプリケーションは正常に始動しました (Application started up successfully)」によって示されます。

始動後に、ブリッジで発生した問題は、ログ・ファイルと (トレースが使用可能になっている場合) トレース・ファイルに記録されます。適切に構成すると、SMTP 経由のアラートがサポートされます。

ブリッジの停止

ブリッジは、ブリッジ・コマンド行で `stop` コマンドを発行すると停止されます。構成プロパティ `PollFrequency` 次第で、アプリケーションの終了に時間がかかる場合があります。ブリッジは、`FRONT ARENA AMB` と連動しているスレッドがウェイクアップし、メッセージの処理を終了するまでは停止できません。

第 9 章 アダプターのトラブルシューティング

- 『アダプターのエラー処理』
- 120 ページの『アプリケーションのトレース』
- 120 ページの『ErrorQueue の検査』
- 121 ページの『アプリケーションへの接続不可』
- 121 ページの『アーカイブ・キューの管理』
- 121 ページの『アンサブスクライブ・キューの管理』
- 121 ページの『エラー処理』
- 123 ページの『要求処理』
- 124 ページの『要求対応処理』
- 125 ページの『アダプターのトレース』

この章では、コネクタ、発生する可能性のあるその他のシナリオをトラブルシューティングする方法について説明します。

アダプターのエラー処理

アダプターのブリッジ・コンポーネントは、以下の 3 タイプのエラーに直面する可能性があります。

- FRONT ARENA との通信の失敗。
- WebSphere MQ との通信の失敗。
- 内部エラー、例えば、メモリー不足状態。

どの場合も、エラーがログに記録されます。ブリッジの構成に従って、アダプターのブリッジ・コンポーネントは、FRONT ARENA アプリケーションのアラート機構を使用して、SMTP メールを追加して送信し、問題の存在を示すことができます。

回復可能な通信問題の場合には、ブリッジは回復を試行します。回復が失敗した場合、または回復不能なエラーが発生した場合、アダプターのブリッジ・コンポーネントは終了します。

ブリッジ・コンポーネントとアダプター内の他のコンポーネントの間の通信が非同期であるため、ブリッジの終了は、非同期サービス要求のタイムアウトの場合を除いて、すぐには通知されません。ただし、これは、他の理由でも発生する可能性があります。

エラーの発生時に SMTP メールを送出するようにブリッジが構成されている場合は、E メール・アダプターがそのメールを受信し、事後処理を開始することができます。

他のアダプターのエラー処理

アダプターは、以下の 3 つのタイプのエラーを区別します。

- FRONT ARENA イベント通知のビジネス・オブジェクトへの変換の失敗。

- アダプターとアダプターのブリッジ・コンポーネントまたはサーバー・サイドのコネクター・エージェントとの通信の失敗。
- FRONT ARENA メッセージにサービス要求を変換できないエラー。

FRONT ARENA イベント通知のビジネス・オブジェクトへの変換が失敗する最初のケースでは、処理中のメッセージは、アダプターのアプリケーション構成プロパティに、有効なキューが指定されている場合、アダプターの **ErrorQueue** に転送されます。それ以外の場合、イベント・メッセージは単純に廃棄されます。

破壊されたイベント・メッセージ以外にも、アダプターの静的変換メタオブジェクト内の <object>_<verb> エントリーに欠落したエントリーも、この障害の原因である可能性があります。

エラーがデータ・ハンドラーによって検出された場合は、障害の詳細がアダプターのログ・ファイルに記録されている可能性があります。

アダプターとアダプターのブリッジ・コンポーネントまたはサーバー・サイドのコネクター・エージェントとの通信が失敗するケースでは、アダプターがログ・ファイル・エントリーを作成します。処理中のイベント・メッセージは、アダプターの **InProgressQueue** に残ります。

サービス要求を FRONT ARENA メッセージに変換できない 3 番目のケースでは、ログ・ファイル・エントリーが作成され、エラーのサービス要求は廃棄されます。否定応答が、発信したコラボレーションに送信されます。

イベントをサブスクライブしているコラボレーションがない場合、イベントは、**UnsubscribedQueue** がアダプター構成に定義されている場合、そのキューに格納されます。それ以外の場合、イベントは単純に廃棄されます。

アプリケーションのトレース

FRONT ARENA - WebSphere MQ Bridge は、そのログ・ファイルで検出されたエラーをすべて記録します。構成されたログ・レベルに従って、主要なエラーでないイベントも報告されます。

さらに、ブリッジは、その処理の詳細な追跡を可能にするトレースを備えています。さまざまなトレース・レベルがサポートされています。ログとトレースの両レベルは、アプリケーションの構成ファイルで初期設定されます。両方の設定内容は、コマンドを使用して動的に変更できます。

ErrorQueue の検査

アダプターが処理できないイベント通知は、アダプターの **ErrorQueue** に格納されます。アダプターには、そのようなエラーを処理するための追加サポートは用意されていません。

アプリケーションへの接続不可

最初のバージョンのアダプターがイベント処理のみをサポートし、アダプター内部の通信が非同期であるため、アダプターは接続に関する問題を通知しません。ブリッジが接続を再確立できない場合に、SMTP メールが適切に設定されているときは、ブリッジはこれを送出します。E メール・アダプターがこのメールを横取り処理することができるため、どのような事後処理の開始も可能です。

アーカイブ・キューの管理

正常に処理されたイベント通知は、アダプターの `ArchiveQueue` に格納されます。それ以上のアーカイブ・サポートは提供されていません。必要に応じて、アーカイブ・キューの内容をデータベースまたはファイルに保管するのは、お客様に任されています。

アンサブスクライブ・キューの管理

イベントをサブスクライブしているコラボレーションがない場合、イベントは、`UnsubscribedQueue` がアダプター構成に定義されている場合、そのキューに格納されます。それ以外の場合、イベントは単純に廃棄されます。後の時点で、例えば、それらのイベントをサブスクライブしているコラボレーションがアクティブになった後などに再度処理する場合は、イベントを `InputQueue` に移動する必要があります。

エラー処理

アダプターが生成するエラー・メッセージはすべて、`FrontArena.txt` という名前のメッセージ・ファイルに保管されます。ファイル名は、`LogFileName` 標準コネクター構成プロパティによって決定されます。各エラーはエラー番号が付けられ、その後エラー・メッセージが表示されます。

Message number
Message text

アダプターは、以降のセクションで説明する方法で特定のエラーを処理します。

アプリケーションのタイムアウト

以下の場合に、エラー・メッセージ「`BON_APPRESPONSETIMEOUT`」が戻されません。

- アダプターは、メッセージ検索時に `WebSphere MQ` への接続を確立できなかった。
- コネクターはビジネス・オブジェクトをメッセージに正常に変換しましたが、接続切断が原因でメッセージを出力キューにデリバリーできなかった。
- アダプターはメッセージを発行したが、変換プロパティ `TimeoutFatal` の値が `True` であるビジネス・オブジェクトの応答待ちがタイムアウトになった。
- アダプターは、`APP_RESPONSE_TIMEOUT` または `UNABLE_TO_LOGIN` に等しい戻りコードを持つ応答メッセージを受け取った。

アンサブスクライブされたビジネス・オブジェクト

アンサブスクライブされたビジネス・オブジェクトに関連付けられているメッセージを検索した場合、アダプターは `UnsubscribedQueue` プロパティで指定されたキューにメッセージをデリバリーします。

注: `UnsubscribedQueue` が定義されていない場合、アンサブスクライブされたメッセージは破棄されます。

データ・ハンドラーの変換

データ・ハンドラーがメッセージをビジネス・オブジェクトに変換できなかった場合や (WebSphere MQ ではなく) ビジネス・オブジェクトに固有の処理エラーが発生した場合、メッセージは、`ErrorQueue` で指定されたキューに送信されます。`ErrorQueue` が定義されていない場合、エラーが原因で処理できなかったメッセージは破棄されます。

データ・ハンドラーがビジネス・オブジェクトからメッセージへの変換に失敗した場合、`BON_FAIL` が戻されます。

イベント・メッセージからビジネス・オブジェクトへの変換の失敗

この場合、処理中のメッセージは、アダプターのアプリケーション構成プロパティに有効なキューが指定されていると、アダプターの `ErrorQueue` に転送されます。それ以外の場合、イベント・メッセージは単純に廃棄されます。

破壊されたイベント・メッセージ以外にも、アダプターの静的変換メタオブジェクト内の `<object>_<verb>` エントリーに欠落したエントリーも、この障害の原因である可能性があります。エラーがデータ・ハンドラーによって検出された場合は、障害の詳細がアダプターのログ・ファイルに記録されている可能性があります。

推奨処置: エラー・キューにあるメッセージはすべて、通常は不適切な構成の兆候であるため、分析する必要があります。例えば、予想外のアプリケーションが入力キューにメッセージを送信する場合、またはメタオブジェクト定義内にエラーが存在する場合、またはメッセージのバージョンを間違えてアダプターと比較する場合などがあります。

アダプターと InterChange Server 間の通信障害またはアダプターのクラッシュ

この場合、アダプターは、ログ・ファイル・エントリーを作成して、シャットダウンします。処理中のイベント・メッセージは、アダプターの `InProgressQueue` に残ります。`InProgressQueue` 内のすべてのメッセージは、サーバーによる処理が終了していないため、アダプターが再始動されると、自動的に再度処理されます。

推奨処置: なし。

サブスクリプションの欠落

イベントをサブスクライブしているコラボレーションがない場合、イベントは、`UnsubscribedQueue` がアダプター構成に定義されている場合、そのキューに格納されます。それ以外の場合、イベントは単純に廃棄されます。

推奨処置: 「スリープしている」コラボレーションがこの状況の原因と考えられる場合は、メッセージをアダプターの入力キューに移動して戻し、コラボレーションを開始した後に再処理を強制的に実行します。UnsubscribedQueue から、ビジネス・オブジェクトと動詞の不適切な組み合わせを持つメッセージ (例えば、統合シナリオ内部の宛先のないメッセージなど) を除去し、それらのメッセージの送信元を停止します。

イベント関連キューの問題

イベント関連キューのいずれかが作動可能でない場合、アダプターは、ログ・ファイルにメッセージを書き込み、シャットダウンします。

推奨処置: アダプターが終了した場合は、アダプターのログ・ファイルも考慮に入れ、慎重に分析する必要があります。

イベントの重複デリバリー

イベントの 2 回または複数回のデリバリーが発生する可能性は非常に低いですが、不可能とはいえません。ブリッジが、アダプターのイベント・キューにイベント・メッセージを挿入する時点と FRONT ARENA の AMB から除去する時点の間にはわずかなギャップがあります。ブリッジが、これらの 2 つの近接する命令の間にクラッシュした場合、ブリッジが再始動されるとすぐに、イベント・メッセージは再配信されます。このギャップは、FRONT ARENA が外部トランザクション・コーディネーターに何の手掛かりも与えないため、ブリッジによって閉じることはできません。

推奨処置: ブリッジがイベントを 2 回配信する場合、アダプターは、処理済みイベントの履歴を維持していないため、そのことを検出できません。重複イベントの処理は、関係表と連動して、コラボレーションで実行する必要があります。FRONT ARENA は、システム内で作成された各オブジェクトに固有キーを割り当てます。システム、より正確には、AMBA が適切に構成されると、このキーはオブジェクトの作成、変更、または削除を示すイベント・メッセージの一部となります。やはりイベント・メッセージの一部である作成時刻または更新時刻と併せて、この情報は、イベント・メッセージが以前にも受信されたことがあるかを検証するために使用できます。

例: TRADE オブジェクトの場合、メッセージ・フィールド trdnbr には、固有の内部 FRONT ARENA キーが含まれ、update_time には、オブジェクトが作成、更新、または削除された時を示すタイム・スタンプが含まれています。

要求処理

以下のセクションでは、要求処理の問題について説明します。

始動しないアダプター

アダプターが始動されないと、コラボレーションは、サービス要求を処理できません。フローは失敗フローで終了し、アダプターを始動した後に再開する必要があります。

推奨処置: アダプターを始動し、失敗したフローをそこから再開します。

要求後のアダプターの終了

アダプターが、要求の後、要求に対する応答を指示する前に終了します。アダプターは要求の状況が不明であることを示し、フローは失敗フローとして終了します。

推奨処置: 問題の修復後、失敗フローを再開します。これは、重複要求を示すこともあります。重複要求は対処可能です。

アプリケーション固有ビジネス・オブジェクトの直列化

アプリケーション固有ビジネス・オブジェクトの直列化が、(例えば、データ・ハンドラーの問題があるため) 失敗した場合、アダプターは例外をスローして、そのログ・ファイルにエントリーを書き込みます。フローは、失敗フローとして終了します。

推奨処置: ログのメッセージを分析します。その後、アダプターのバグを修正し、アダプターを再始動します。最終的に、失敗したフローを再開します。

要求キューへの要求メッセージの挿入不可

要求キューにメッセージを挿入できない場合、アダプターは終了します。アダプターは、それ自体の構成に従って、自動的に再始動します。要求フローは、キュー・アクセス問題があったことを示して、失敗フローとして終了します。

推奨処置: WebSphere MQ キューの問題を訂正し、失敗したフローを再開します。

要求対応答処理

以下のセクションは、要求対応答処理に関する問題について説明します。

タイムアウト

要求に対する応答の待機中に最大待機時間が経過した場合、要求は失敗したものと見なされます。タイムアウトがコラボレーションに戻され、フローは失敗フローとして終了します。

タイムアウトが発生した場合でも、要求が、ターゲット・アプリケーションによって正常に処理されている可能性もあります。失敗したフローを再開すると、要求の繰り返しになります。

アダプターは、要求履歴を維持していないため、重複要求を検出できません。FRONT ARENA では、重複オブジェクト要求は問題とはなりません。FRONT ARENA は、オブジェクトに対する 2 番目の作成要求を、オブジェクトに変更を一切適用しない更新要求に内部的に変換します。重複した更新要求は、何の効果もありません。2 番目の削除要求は、オブジェクトが存在しないため失敗します。

推奨処置: 環境が現在も稼働中であることを確認した後、失敗フローを再開します。

タイムアウトした要求に関連する応答キュー内の応答メッセージ

要求に対する応答が、アダプターに対して構成されている最大待機時間の経過後に到着した場合、アダプターは、その要求を無視します。応答キュー内の応答メッセ

ージが、すでにタイムアウトした要求に関連する場合は、タイムアウト・メッセージがコラボレーションに送信された後、失敗フローとして終了します。

推奨処置: 応答キューを定期的にクリーンアップします。失敗したフローが存在する場合、診断目的にそれらのメッセージを使用することが可能です。

失敗要求を示す応答

失敗した要求がリターン情報をフィードバック・フィールドに持っていることを示す応答、およびメッセージ本文の一部としてのエラー・メッセージ。

推奨処置: コラボレーションは、文書化されているエラー状況を処理する必要があります。詳細については、69 ページの『第 6 章 プリッジの構成』を参照してください。

要求キューからのメッセージの取得不可

メッセージが要求キューから読み取れない場合、アダプターは終了します。アダプターは、それ自体の構成に従って、自動的に再始動します。要求フローは、キュー・アクセス問題があったことを示して、失敗フローとして終了します。

推奨処置: WebSphere MQ キューの問題を訂正し、失敗したフローを再開します。

アダプターのトレース

トレースはオプションのデバッグ機能で、オンにするとアダプターの振る舞いを詳細にトレースできます。デフォルトでは、トレース・メッセージは `STDOUT` に書き込まれます。トレース・メッセージの構成に関する詳細については、コネクタ構成プロパティを参照してください。トレースに関する情報、トレースを有効化して設定する方法について詳しくは、「コネクタ開発ガイド」を参照してください。

次に、トレース・メッセージに推奨する内容を示します。

- | | |
|-------|--|
| レベル 0 | このレベルは、アダプターのバージョンを示すトレース・メッセージに使用されます。 |
| レベル 1 | 処理される各ビジネス・オブジェクトについての重要な情報を提供するトレース・メッセージや、ポーリング・スレッドが入力キュー内で新しいメッセージを検出するたびに記録されるトレース・メッセージに使用します。 |
| レベル 2 | ビジネス・オブジェクトが <code>gotApp1Event()</code> または <code>executeCollaboration()</code> から InterChange Server に送付されるたびに記録されるトレース・メッセージに使用します。 |
| レベル 3 | メッセージからビジネス・オブジェクトへの変換およびビジネス・オブジェクトからメッセージへの変換に関する情報を提供するトレース・メッセージや、出力キューへのメッセージのデリバリーに関する情報を提供するトレース・メッセージに使用します。 |
| レベル 4 | このレベルは、アダプターがある関数を入力または出力する場合を示すトレース・メッセージに使用します。 |

- レベル 5 このレベルは、アダプターの初期化を示すトレース・メッセージ、アプリケーション内で実行されるステートメントを示すトレース・メッセージ、メッセージがキューから取り出されたりキューに入れられたりしたときにそれを記録するトレース・メッセージ、ビジネス・オブジェクトのダンプを記録するトレース・メッセージなどに使用されます。

ブリッジのトレース

トレースはオプションのデバッグ機能で、オンにするとブリッジの振る舞いを詳細にトレースできます。トレース・メッセージは、ブリッジ構成ファイル・パラメーター `TraceFile` に指定されているファイルに書き込まれます。トレース・レベルは、構成ファイル・パラメーター `TraceFile` で設定するか、コマンド・ウィンドウでトレース・コマンドを入力して設定することができます。トレース・メッセージの構成の詳細については、69 ページの『第 6 章 ブリッジの構成』でブリッジ構成プロパティを参照してください。

次に、トレース・メッセージに推奨する内容を示します。

- レベル 0 トレース・スイッチがオフです。
- レベル 1 トレース・エラー・メッセージにはこのレベルを使用してください。
- レベル 2 使用されません。
- レベル 3 使用されません。
- レベル 4 ブリッジが関数を実行または終了する時を示すトレース・メッセージには、このレベルを使用してください。
- レベル 5 情報メッセージを示すトレース・メッセージには、このレベルを使用してください。

付録 A. コネクターの標準構成プロパティ

- 新規プロパティと削除されたプロパティ
- 標準コネクタ・プロパティの構成
- 標準プロパティの要約
- 標準構成プロパティ

新規プロパティと削除されたプロパティ

以下の標準プロパティは、本リリースで追加されました。

新規プロパティ

- XMLNamespaceFormat

削除されたプロパティ

- RestartCount

標準コネクタ・プロパティの構成

アダプター・コネクタには 2 つのタイプの構成プロパティがあります。

- 標準構成プロパティ
- コネクタ固有のプロパティ

このセクションでは、標準構成プロパティについて説明します。コネクタ固有の構成プロパティについては、該当するアダプターのユーザーズ・ガイドを参照してください。

Connector Configurator の使用

Connector Configurator からコネクタ・プロパティを構成します。Connector Configurator には、System Manager からアクセスします。Connector Configurator の使用法の詳細については、付録の『Connector Configurator』を参照してください。

注: Connector Configurator と System Manager は、Windows システム上でのみ動作します。コネクタを UNIX システム上で稼働している場合でも、これらのツールがインストールされた Windows マシンが必要です。UNIX 上で動作するコネクタのコネクタ・プロパティを設定する場合は、Windows マシン上で System Manager を起動し、UNIX の統合ブローカーに接続してから、コネクタ用の Connector Configurator を開く必要があります。

プロパティ値の設定と更新

プロパティ・フィールドのデフォルトの長さは 255 文字です。

コネクタは、以下の順序に従ってプロパティの値を決定します (最も番号の大きい項目が他の項目よりも優先されます)。

1. デフォルト

2. リポジトリ (WebSphere InterChange Server が統合ブローカーである場合のみ)
3. ローカル構成ファイル
4. コマンド行

コネクタは、始動時に構成値を取得します。実行時セッション中に 1 つ以上のコネクタ・プロパティの値を変更する場合は、プロパティの**更新メソッド**によって、変更を有効にする方法が決定されます。標準コネクタ・プロパティには、以下の 4 種類の更新メソッドがあります。

- **動的**

変更を System Manager に保管すると、変更が即時に有効になります。コネクタが System Manager から独立してスタンドアロン・モードで稼働している場合 (例えば、いずれかの WebSphere Message Brokers と連携している場合) は、構成ファイルでのみプロパティを変更できます。この場合、動的更新は実行できません。

- **エージェント再始動 (ICS のみ)**

アプリケーション固有のコンポーネントを停止して再始動しなければ、変更が有効になりません。

- **コンポーネント再始動**

System Manager でコネクタを停止してから再始動しなければ、変更が有効になりません。アプリケーション固有コンポーネントまたは統合ブローカーを停止、再始動する必要はありません。

- **サーバー再始動**

アプリケーション固有のコンポーネントおよび統合ブローカーを停止して再始動しなければ、変更が有効になりません。

特定のプロパティの更新方法を確認するには、「Connector Configurator」ウィンドウ内の「更新メソッド」列を参照するか、次に示す129 ページの表 51 の「更新メソッド」列を参照してください。

標準プロパティの要約

129 ページの表 51 は、標準コネクタ構成プロパティの早見表です。標準プロパティの依存関係は RepositoryDirectory に基づいているため、コネクタによっては使用されないプロパティがあり、使用する統合ブローカーによってプロパティの設定が異なる可能性があります。

コネクタを実行する前に、これらのプロパティの一部の値を設定する必要があります。各プロパティの詳細については、次のセクションを参照してください。

注: 129 ページの表 51 の「注」列にある「Repository Directory は REMOTE」という句は、ブローカーが InterChange Serverであることを示します。ブローカーが WMQI または WAS の場合には、リポジトリ・ディレクトリーは LOCAL に設定されます。

表 51. 標準構成プロパティの要約

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
AdminInQueue	有効な JMS キュー名	CONNECTORNAME /ADMININQUEUE	コンポ ネント 再始動	Delivery Transport は JMS
AdminOutQueue	有効な JMS キュー名	CONNECTORNAME/ADMINOUTQUEUE	コンポ ネント 再始動	Delivery Transport は JMS
AgentConnections	1 から 4	1	コンポ ネント 再始動	Delivery Transport は MQ および IDL: Repository Directory は <REMOTE> (ブローカーは ICS)
AgentTraceLevel	0 から 5	0	動的	
ApplicationName	アプリケーション名	コネクタ・アプリケーション名 として指定された値	コンポ ネント 再始動	
BrokerType	ICS、WMQI、WAS		コンポ ネント 再始動	
CharacterEncoding	ascii7、ascii8、SJIS、 Cp949、GBK、Big5、 Cp297、Cp273、Cp280、 Cp284、Cp037、Cp437 注: これは、 サポートされる値 の一部です。	ascii7	コンポ ネント 再始動	
ConcurrentEventTriggeredFlows	1 から 32,767	1	コンポ ネント 再始動	Repository Directory は <REMOTE> (ブローカーは ICS)
ContainerManagedEvents	値なしまたは JMS	値なし	コンポ ネント 再始動	Delivery Transport は JMS
ControllerStoreAndForwardMode	true または false	true	動的	Repository Directory は <REMOTE> (ブローカーは ICS)
ControllerTraceLevel	0 から 5	0	動的	Repository Directory は <REMOTE> (ブローカーは ICS)
DeliveryQueue		CONNECTORNAME/DELIVERYQUEUE	コンポ ネント 再始動	JMS トランス ポートのみ

表 51. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
DeliveryTransport	MQ、IDL、または JMS	JMS	コンポ ネント 再始動	Repository Directory が ローカルの 場合は、値は JMS のみ
DuplicateEventElimination	true または false	false	コンポ ネント 再始動	JMS トランス ポートのみ、 Container Managed Events は <NONE> でなければ ならない
FaultQueue		CONNECTORNAME/FAULTQUEUE	コンポ ネント 再始動	JMS トランス ポートのみ
jms.FactoryClassName	CxCommon.Messaging.jms .IBMMQSeriesFactory ま たはCxCommon.Messaging .jms.SonicMQFactory または任意の Java クラス 名	CxCommon.Messaging. jms.IBMMQSeriesFactory	コンポ ネント 再始動	JMS トランス ポートのみ
jms.MessageBrokerName	FactoryClassName が IBM の場合は crossworlds.queue. manager を使用。 FactoryClassName が Sonicの場合 localhost:2506 を使用。	crossworlds.queue.manager	コンポ ネント 再始動	JMS トランス ポートのみ
jms.NumConcurrentRequests	正整数	10	コンポ ネント 再始動	JMS トランス ポートのみ
jms.Password	任意の有効なパスワード		コンポ ネント 再始動	JMS トランス ポートのみ
jms.UserName	任意の有効な名前		コンポ ネント 再始動	JMS トランス ポートのみ
JvmMaxHeapSize	ヒープ・サイズ (メガバイ ト単位)	128m	コンポ ネント 再始動	Repository Directory は <REMOTE> (ブローカーは ICS)
JvmMaxNativeStackSize	スタックのサイズ (キロバ イト単位)	128k	コンポ ネント 再始動	Repository Directory は <REMOTE> (ブローカーは ICS)
JvmMinHeapSize	ヒープ・サイズ (メガバイ ト単位)	1m	コンポ ネント 再始動	Repository Directory は <REMOTE> (ブローカーは ICS)

表 51. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
ListenerConcurrency	1 から 100	1	コンポーネント再始動	Delivery Transport は MQ でなければならぬ
Locale	en_US、ja_JP、ko_KR、zh_CN、zh_TW、fr_FR、de_DE、it_IT、es_ES、pt_BR 注: これは、サポートされるロケールの一部です。	en_US	コンポーネント再始動	
LogAtInterchangeEnd	true または false	false	コンポーネント再始動	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
MaxEventCapacity	1 から 2147483647	2147483647	動的	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
MessageFileName	パスまたはファイル名	CONNECTORNAMEConnector.txt	コンポーネント再始動	
MonitorQueue	任意の有効なキュー名	CONNECTORNAME/MONITORQUEUE	コンポーネント再始動	JMS トランスポートのみ: DuplicateEvent Elimination は true でなければならぬ
OADAutoRestartAgent	true または false	false	動的	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
OADMaxNumRetry	正数	1000	動的	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
OADRetryTimeInterval	正数 (単位: 分)	10	動的	Repository Directory は <REMOTE> でなければならぬ (ブローカーは ICS)
PollEndTime	HH:MM	HH:MM	コンポーネント再始動	

表 51. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
PollFrequency	正整数 (単位: ミリ秒) no (ポーリングを使用不可にする) key (コネクタのコマンド・プロンプト・ウィンドウで文字 p が入力された場合にのみポーリングする)	10000	動的	
PollQuantity	1 から 500	1	エージェント再始動	JMS トランスポートのみ: Container Managed Events を指定
PollStartTime	HH:MM (HH は 0 から 23、MM は 0 から 59)	HH:MM	コンポーネント再始動	
RepositoryDirectory	メタデータ・リポジトリの場所		エージェント再始動	ICS の場合は <REMOTE> に設定する。 WebSphere MQ Message Brokers および WAS の場合: C:\crossworlds¥repository に設定する
RequestQueue	有効な JMS キュー名	CONNECTORNAME/REQUESTQUEUE	コンポーネント再始動	Delivery Transport は JMS
ResponseQueue	有効な JMS キュー名	CONNECTORNAME/RESPONSEQUEUE	コンポーネント再始動	Delivery Transport が JMS の場合: Repository Directory が <REMOTE> の場合のみ必要
RestartRetryCount	0 から 99	3	動的	
RestartRetryInterval	適切な正数 (単位: 分): 1 から 2147483547	1	動的	
RHF2MessageDomain	mrm、xml	mrm	コンポーネント再始動	Delivery Transport が JMS であり、かつ WireFormat が CwXML である。

表 51. 標準構成プロパティの要約 (続き)

プロパティ名	指定可能な値	デフォルト値	更新メソッド	注
SourceQueue	有効な WebSphere MQ 名	CONNECTORNAME/SOURCEQUEUE	エージェント再始動	Delivery Transport が JMS であり、かつ Container Managed Events が指定されている場合のみ
SynchronousRequestQueue		CONNECTORNAME/ SYNCHRONOUSREQUESTQUEUE	コンポーネント再始動	Delivery Transport は JMS
SynchronousRequestTimeout	0 以上の任意の数値 (ミリ秒)	0	コンポーネント再始動	Delivery Transport は JMS
SynchronousResponseQueue		CONNECTORNAME/ SYNCHRONOUSRESPONSEQUEUE	コンポーネント再始動	Delivery Transport は JMS
WireFormat	CwXML、CwBO	CwXML	エージェント再始動	Repository Directory が <REMOTE> でない場合は CwXML。Repository Directory が <REMOTE> であれば CwBO
WsifSynchronousRequestTimeout	0 以上の任意の数値 (ミリ秒)	0	コンポーネント再始動	WAS のみ
XMLNameSpaceFormat	short、long	short	エージェント再始動	WebSphere MQ Message Brokers および WAS のみ

標準構成プロパティ

このセクションでは、各標準コネクタ構成プロパティの定義を示します。

AdminInQueue

統合ブローカーからコネクタへ管理メッセージが送信されるときに使用されるキューです。

デフォルト値は CONNECTORNAME/ADMININQUEUE です。

AdminOutQueue

コネクタから統合ブローカーへ管理メッセージが送信されるときに使用されるキューです。

デフォルト値は CONNECTORNAME/ADMINOUTQUEUE です。

AgentConnections

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

AgentConnections プロパティは、orb.init[] により開かれる ORB (オブジェクト・リクエスト・ブローカー) 接続の数を制御します。

このプロパティのデフォルト値は 1 に設定されます。必要に応じてこの値を変更できます。

AgentTraceLevel

アプリケーション固有のコンポーネントのトレース・メッセージのレベルです。デフォルト値は 0 です。コネクタは、設定されたトレース・レベル以下の該当するトレース・メッセージをすべてデリバリーします。

ApplicationName

コネクタのアプリケーションを一意的に特定する名前です。この名前は、システム管理者が WebSphere Business Integration システム環境をモニターするために使用されます。コネクタを実行する前に、このプロパティに値を指定する必要があります。

BrokerType

使用する統合ブローカー・タイプを指定します。オプションは ICS、WebSphere Message Brokers (WMQI、WMQIB または WBIMB) または WAS です。

CharacterEncoding

文字 (アルファベットの文字、数値表現、句読記号など) から数値へのマッピングに使用する文字コード・セットを指定します。

注: Java ベースのコネクタでは、このプロパティは使用しません。C++ ベースのコネクタでは、現在、このプロパティに `ascii7` という値が使用されています。

デフォルトでは、ドロップダウン・リストには、サポートされる文字エンコードの一部のみが表示されます。ドロップダウン・リストに、サポートされる他の値を追加するには、製品ディレクトリーにある `¥Data¥Std¥stdConnProps.xml` ファイルを手動で変更する必要があります。詳細については、Connector Configurator に関する付録を参照してください。

ConcurrentEventTriggeredFlows

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

コネクタがイベントのデリバリー時に並行処理できるビジネス・オブジェクトの数を決定します。この属性の値を、並行してマップおよびデリバリーできるビジネス・オブジェクトの数に設定します。例えば、この属性の値を 5 に設定すると、5 個のビジネス・オブジェクトが並行して処理されます。デフォルト値は 1 です。

このプロパティを 1 よりも大きい値に設定すると、ソース・アプリケーションのコネクタが、複数のイベント・ビジネス・オブジェクトを同時にマップして、複

数のコラボレーション・インスタンスにそれらのビジネス・オブジェクトを同時にデリバリーすることができます。これにより、統合ブローカーへのビジネス・オブジェクトのデリバリーにかかる時間、特にビジネス・オブジェクトが複雑なマップを使用している場合のデリバリー時間が短縮されます。ビジネス・オブジェクトのコラボレーションに到達する速度を増大させると、システム全体のパフォーマンスを向上させることができます。

ソース・アプリケーションから宛先アプリケーションまでのフロー全体に並行処理を実装するには、次のようにする必要があります。

- `Maximum number of concurrent events` プロパティの値を増加して、コラボレーションが複数のスレッドを使用できるように構成します。
- 宛先アプリケーションのアプリケーション固有コンポーネントが複数の要求を並行して実行できることを確認します。つまり、このコンポーネントがマルチスレッド化されているか、またはコネクタ・エージェント並列処理を使用でき、複数プロセスに対応するよう構成されている必要があります。`Parallel Process Degree` 構成プロパティに、1 より大きい値を設定します。

`ConcurrentEventTriggeredFlows` プロパティは、順次に行われる単一スレッド処理であるコネクタのポーリングでは無効です。

ContainerManagedEvents

このプロパティにより、JMS イベント・ストアを使用する JMS 対応コネクタが、保証付きイベント・デリバリーを提供できるようになります。保証付きイベント・デリバリーでは、イベントはソース・キューから除去され、単一 JMS トランザクションとして宛先キューに配置されます。

デフォルト値はありません。

`ContainerManagedEvents` を JMS に設定した場合には、保証付きイベント・デリバリーを使用できるように次のプロパティも構成する必要があります。

- `PollQuantity` = 1 から 500
- `SourceQueue` = /SOURCEQUEUE

また、`MimeType`、`DHClass` (データ・ハンドラー・クラス)、および `DataHandlerConfigMOName` (オプションのメタオブジェクト名) プロパティを設定したデータ・ハンドラーも構成する必要があります。これらのプロパティの値を設定するには、`Connector Configurator` の「データ・ハンドラー」タブを使用します。

これらのプロパティはアダプター固有ですが、例の値は次のようになります。

- `MimeType` = `text/xml`
- `DHClass` = `com.crossworlds.DataHandlers.text.xml`
- `DataHandlerConfigMOName` = `MO_DataHandler_Default`

「データ・ハンドラー」タブのこれらの値のフィールドは、`ContainerManagedEvents` を JMS に設定した場合にのみ表示されます。

注: ContainerManagedEvents を JMS に設定した場合、コネクターはその pollForEvents() メソッドを呼び出さなくなるため、そのメソッドの機能は使用できなくなります。

このプロパティーは、DeliveryTransport プロパティーが値 JMS に設定されている場合にのみ表示されます。

ControllerStoreAndForwardMode

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

宛先側のアプリケーション固有のコンポーネントが使用不可であることをコネクター・コントローラーが検出した場合に、コネクター・コントローラーが実行する動作を設定します。

このプロパティーを true に設定した場合、イベントが ICS に到達したときに宛先側のアプリケーション固有のコンポーネントが使用不可であれば、コネクター・コントローラーはそのアプリケーション固有のコンポーネントへの要求をブロックします。アプリケーション固有のコンポーネントが作動可能になると、コネクター・コントローラーはアプリケーション固有のコンポーネントにその要求を転送します。

ただし、コネクター・コントローラーが宛先側のアプリケーション固有のコンポーネントにサービス呼び出し要求を転送した後でこのコンポーネントが使用不可になった場合、コネクター・コントローラーはその要求を失敗させます。

このプロパティーを false に設定した場合、コネクター・コントローラーは、宛先側のアプリケーション固有のコンポーネントが使用不可であることを検出すると、ただちにすべてのサービス呼び出し要求を失敗させます。

デフォルト値は true です。

ControllerTraceLevel

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

コネクター・コントローラーのトレース・メッセージのレベルです。デフォルト値は 0 です。

DeliveryQueue

DeliveryTransport が JMS の場合のみ適用されます。

コネクターから統合ブローカーへビジネス・オブジェクトが送信されるときに使用されるキューです。

デフォルト値は CONNECTORNAME/DELIVERYQUEUE です。

DeliveryTransport

イベントのデリバリーのためのトランスポート機構を指定します。指定可能な値は、WebSphere MQ の MQ、CORBA IIOP の IDL、Java Messaging Service の JMS です。

- RepositoryDirectory がリモートの場合は、DeliveryTransport プロパティの指定可能な値は MQ、IDL、または JMS であり、デフォルトは IDL になります。
- RepositoryDirectory がローカル・ディレクトリーの場合は、指定可能な値は JMS のみです。

DeliveryTransport プロパティに指定されている値が、MQ または IDL である場合、コネクタは、CORBA IIOP を使用してサービス呼び出し要求と管理メッセージを送信します。

WebSphere MQ および IDL

イベントのデリバリー・トランスポートには、IDL ではなく WebSphere MQ を使用してください (1 種類の製品だけを使用する必要がある場合を除きます)。

WebSphere MQ が IDL よりも優れている点は以下のとおりです。

- 非同期 (ASYNC) 通信:
WebSphere MQ を使用すると、アプリケーション固有のコンポーネントは、サーバーが利用不能である場合でも、イベントをポーリングして永続的に格納することができます。
- サーバー・サイド・パフォーマンス:
WebSphere MQ を使用すると、サーバー・サイドのパフォーマンスが向上します。最適化モードでは、WebSphere MQ はイベントへのポインターのみをリポジトリ・データベースに格納するので、実際のイベントは WebSphere MQ キュー内に残ります。これにより、サイズが大きい可能性のあるイベントをリポジトリ・データベースに書き込む必要がありません。
- エージェント・サイド・パフォーマンス:
WebSphere MQ を使用すると、アプリケーション固有のコンポーネント側のパフォーマンスが向上します。WebSphere MQ を使用すると、コネクタのポーリング・スレッドは、イベントを選出した後、コネクタのキューにそのイベントを入れ、次のイベントを選出します。この方法は IDL よりも高速で、IDL の場合、コネクタのポーリング・スレッドは、イベントを選出した後、ネットワーク経由でサーバー・プロセスにアクセスしてそのイベントをリポジトリ・データベースに永続的に格納してから、次のイベントを選出する必要があります。

JMS

Java Messaging Service (JMS) を使用しての、コネクタとクライアント・コネクタ・フレームワークとの間の通信を可能にします。

JMS をデリバリー・トランスポートとして選択した場合は、

`jms.MessageBrokerName`、`jms.FactoryClassName`、`jms.Password`、`jms.UserName` などの追加の JMS プロパティが Connector Configurator 内に表示されます。このうち最初の 2 つは、このトランスポートの必須プロパティです。

重要: 以下の環境では、コネクタに JMS トランスポート機構を使用すると、メモリ制限が発生することもあります。

- AIX 5.0
- WebSphere MQ 5.3.0.1
- ICS が統合ブローカーの場合

この環境では、WebSphere MQ クライアント内でメモリーが使用されるため、(サーバー側の) コネクター・コントローラーと (クライアント側の) コネクターの両方を始動するのは困難な場合があります。ご使用のシステムのプロセス・ヒープ・サイズが 768M 未満である場合には、次のように設定することをお勧めします。

- CWSHaredEnv.sh スクリプト内で LDR_CNTRL 環境変数を設定する。

このスクリプトは、製品ディレクトリー配下の %bin ディレクトリーにあります。テキスト・エディターを使用して、CWSHaredEnv.sh スクリプトの最初の行として次の行を追加します。

```
export LDR_CNTRL=MAXDATA=0x30000000
```

この行は、ヒープ・メモリーの使用量を最大 768 MB (3 セグメント * 256 MB) に制限します。プロセス・メモリーがこの制限値を超えると、ページ・スワッピングが発生し、システムのパフォーマンスに悪影響を与える場合があります。

- IPCCBaseAddress プロパティーの値を 11 または 12 に設定する。このプロパティーの詳細については、「システム・インストール・ガイド (UNIX 版)」を参照してください。

DuplicateEventElimination

このプロパティーを true に設定すると、JMS 対応コネクターによるデリバリー・キューへの重複イベントのデリバリーが防止されます。この機能を使用するには、コネクターに対し、アプリケーション固有のコード内でビジネス・オブジェクトの **ObjectEventId** 属性として一意のイベント ID が設定されている必要があります。これはコネクター開発時に設定されます。

このプロパティーは、false に設定することもできます。

注: DuplicateEventElimination を true に設定する際は、MonitorQueue プロパティーを構成して保証付きイベント・デリバリーを使用可能にする必要があります。

FaultQueue

コネクターでメッセージを処理中にエラーが発生すると、コネクターは、そのメッセージを状況表示および問題説明とともにこのプロパティーに指定されているキューに移動します。

デフォルト値は CONNECTORNAME/FAULTQUEUE です。

JvmMaxHeapSize

エージェントの最大ヒープ・サイズ (メガバイト単位)。このプロパティーは、RepositoryDirectory の値が <REMOTE> の場合にのみ適用されます。

デフォルト値は 128M です。

JvmMaxNativeStackSize

エージェントの最大ネイティブ・スタック・サイズ (キロバイト単位)。このプロパティーは、RepositoryDirectory の値が <REMOTE> の場合にのみ適用されます。

デフォルト値は 128K です。

JvmMinHeapSize

エージェントの最小ヒープ・サイズ (メガバイト単位)。このプロパティは、RepositoryDirectory の値が <REMOTE> の場合にのみ適用されます。

デフォルト値は 1M です。

jms.FactoryClassName

JMS プロバイダーのためにインスタンスを生成するクラス名を指定します。JMS をデリバリー・トランスポート機構 (DeliveryTransport) として選択する際は、このコネクタ・プロパティを必ず 設定してください。

デフォルト値は CxCommon.Messaging.jms.IBMMQSeriesFactory です。

jms.MessageBrokerName

JMS プロバイダーのために使用するブローカー名を指定します。JMS をデリバリー・トランスポート機構 (DeliveryTransport) として選択する際は、このコネクタ・プロパティを必ず 設定してください。

デフォルト値は crossworlds.queue.manager です。ローカル・メッセージ・ブローカーに接続する場合は、デフォルト値を使用します。

リモート・メッセージ・ブローカーに接続すると、このプロパティは次の (必須) 値をとります。

QueueMgrName:<Channel>:<HostName>:<PortNumber>

各変数の意味は以下のとおりです。

QueueMgrName: キュー・マネージャー名です。

Channel: クライアントが使用するチャンネルです。

HostName: キュー・マネージャーの配置先のマシン名です。

PortNumber: キュー・マネージャーが listen に使用するポートの番号です。

例えば、次のようになります。

```
jms.MessageBrokerName = WBIMB.Queue.Manager:CHANNEL1:RemoteMachine:1456
```

jms.NumConcurrentRequests

コネクタに対して同時に送信することができる並行サービス呼び出し要求の数 (最大値) を指定します。この最大値に達した場合、新規のサービス呼び出し要求はブロックされ、既存のいずれかの要求が完了した後に処理されます。

デフォルト値は 10 です。

jms.Password

JMS プロバイダーのためのパスワードを指定します。このプロパティの値はオプションです。

デフォルトはありません。

jms.UserName

JMS プロバイダーのためのユーザー名を指定します。このプロパティの値はオプションです。

デフォルトはありません。

ListenerConcurrency

このプロパティは、統合ブローカーとして ICS を使用する場合の MQ Listener でのマルチスレッド化をサポートしています。このプロパティにより、データベースへの複数イベントの書き込み操作をバッチ処理できるので、システム・パフォーマンスが向上します。デフォルト値は 1 です。

このプロパティは、MQ トランスポートを使用するコネクタにのみ適用されます。DeliveryTransport プロパティには MQ を設定してください。

Locale

言語コード、国または地域、および、希望する場合には、関連した文字コード・セットを指定します。このプロパティの値は、データの照合やソート順、日付と時刻の形式、通貨記号などの国/地域別情報を決定します。

ロケール名は、次の書式で指定します。

ll_TT.codeset

ここで、

<i>ll</i>	2 文字の言語コード (普通は小文字)
<i>TT</i>	2 文字の国または地域コード (普通は大文字)
<i>codeset</i>	関連文字コード・セットの名前。名前のこの部分は、通常、オプションです。

デフォルトでは、ドロップダウン・リストには、サポートされるロケールの一部のみが表示されます。ドロップダウン・リストに、サポートされる他の値を追加するには、製品ディレクトリーにある `¥Data¥Std¥stdConnProps.xml` ファイルを手動で変更する必要があります。詳細については、Connector Configurator に関する付録を参照してください。

デフォルト値は `en_US` です。コネクタがグローバル化に対応していない場合、このプロパティの有効な値は `en_US` のみです。特定のコネクタがグローバル化に対応しているかどうかを判別するには、以下の Web サイトにあるコネクタのバージョン・リストを参照してください。

<http://www.ibm.com/software/websphere/wbiadapters/infocenter>、または
<http://www.ibm.com/websphere/integration/wicsserver/infocenter>

LogAtInterchangeEnd

RepositoryDirectory が <REMOTE> の場合のみ適用可能です。

統合ブローカーのログ宛先にエラーを記録するかどうかを指定します。ブローカーのログ宛先にログを記録すると、電子メール通知もオンになります。これにより、

エラーまたは致命的エラーが発生すると、InterchangeSystem.cfg ファイルに指定された MESSAGE_RECIPIENT に対する電子メール・メッセージが生成されます。

例えば、LogAtInterChangeEnd を true に設定した場合にコネクタからアプリケーションへの接続が失われると、指定されたメッセージ宛先に、電子メール・メッセージが送信されます。デフォルト値は false です。

MaxEventCapacity

コントローラー・バッファ内のイベントの最大数。このプロパティはフロー制御が使用し、RepositoryDirectory プロパティの値が <REMOTE> の場合にのみ適用されます。

値は 1 から 2147483647 の間の正整数です。デフォルト値は 2147483647 です。

MessageFileName

コネクタ・メッセージ・ファイルの名前です。メッセージ・ファイルの標準位置は、製品ディレクトリーの %connectors%messages です。メッセージ・ファイルが標準位置に格納されていない場合は、メッセージ・ファイル名を絶対パスで指定します。

コネクタ・メッセージ・ファイルが存在しない場合は、コネクタは InterchangeSystem.txt をメッセージ・ファイルとして使用します。このファイルは、製品ディレクトリーに格納されています。

注: 特定のコネクタについて、コネクタ独自のメッセージ・ファイルがあるかどうかを判別するには、該当するアダプターのユーザズ・ガイドを参照してください。

MonitorQueue

コネクタが重複イベントをモニターするために使用する論理キューです。このプロパティは、DeliveryTransport プロパティ値が JMS であり、かつ DuplicateEventElimination が TRUE に設定されている場合にのみ使用されます。

デフォルト値は CONNECTORNAME/MONITORQUEUE です。

OADAutoRestartAgent

RepositoryDirectory が <REMOTE> の場合のみ有効です。

コネクタが自動再始動およびリモート再始動機能を使用するかどうかを指定します。この機能では、MQ により起動される Object Activation Daemon (OAD) を使用して、異常シャットダウン後にコネクタを再始動したり、System Monitor からリモート・コネクタを始動したりします。

自動再始動機能およびリモート再始動機能を使用可能にするには、このプロパティを true に設定する必要があります。MQ により起動される OAD 機能の構成方法については、「システム・インストール・ガイド (Windows 版)」または「システム・インストール・ガイド (UNIX 版)」を参照してください。

デフォルト値は false です。

OADMaxNumRetry

RepositoryDirectory が <REMOTE> の場合のみ有効です。

異常シャットダウンの後で MQ により起動される OAD がコネクタの再始動を自動的に試行する回数の最大数を指定します。このプロパティを有効にするためには、OADAutoRestartAgent プロパティを true に設定する必要があります。

デフォルト値は 1000 です。

OADRetryTimeInterval

RepositoryDirectory が <REMOTE> の場合のみ有効です。

MQ により起動される OAD の再試行時間間隔の分数を指定します。コネクタ・エージェントがこの再試行時間間隔内に再始動しない場合は、コネクタ・コントローラーはコネクタ・エージェントを再び再始動するように OAD に要求します。OAD はこの再試行プロセスを OADMaxNumRetry プロパティで指定された回数だけ繰り返します。このプロパティを有効にするためには、OADAutoRestartAgent プロパティを true に設定する必要があります。

デフォルト値は 10 です。

PollEndTime

イベント・キューのポーリングを停止する時刻です。形式は HH:MM です。ここで、HH は 0 から 23 時を表し、MM は 0 から 59 分を表します。

このプロパティには必ず有効な値を指定してください。デフォルト値は HH:MM ですが、この値は必ず変更する必要があります。

PollFrequency

これは、前回のポーリングの終了から次のポーリングの開始までの間の間隔です。PollFrequency は、あるポーリング・アクションの終了から次のポーリング・アクションの開始までの時間をミリ秒単位で指定します。これはポーリング・アクション間の間隔ではありません。この論理を次に説明します。

- ポーリングし、PollQuantity の値により指定される数のオブジェクトを取得します。
- これらのオブジェクトを処理します。一部のアダプターでは、これは個別のスレッドで部分的に実行されます。これにより、次のポーリング・アクションまで処理が非同期に実行されます。
- PollFrequency で指定された間隔にわたって遅延します。
- このサイクルを繰り返します。

PollFrequency は以下の値のいずれかに設定します。

- ポーリング・アクション間のミリ秒数 (整数)。
- ワード key。コネクタは、コネクタのコマンド・プロンプト・ウィンドウで文字 p が入力されたときのみポーリングを実行します。このワードは小文字で入力します。

- ワード no。コネクタはポーリングを実行しません。このワードは小文字で入力します。

デフォルト値は 10000 です。

重要: 一部のコネクタでは、このプロパティの使用が制限されています。このようなコネクタが存在する場合には、アダプターのインストールと構成に関する章で制約事項が説明されています。

PollQuantity

コネクタがアプリケーションからポーリングする項目の数を指定します。アダプターにコネクタ固有のポーリング数設定プロパティがある場合、標準プロパティの値は、このコネクタ固有のプロパティの設定値によりオーバーライドされます。

電子メール・メッセージもイベントと見なされます。コネクタは、電子メールに関するポーリングを受けたときには次のように動作します。

コネクタは、1 回目のポーリングを受けると、メッセージの本文を選出します。これは、本文が添付とも見なされるからです。本文の MIME タイプにはデータ・ハンドラーが指定されていないので、コネクタは本文を無視します。

コネクタは PO の最初の添付を処理します。この添付の MIME タイプには対応するデータ・ハンドラーがあるので、コネクタはビジネス・オブジェクトを Visual Test Connector に送信します。

2 回目のポーリングを受けると、コネクタは PO の 2 番目の添付を処理します。この添付の MIME タイプには対応するデータ・ハンドラーがあるので、コネクタはビジネス・オブジェクトを Visual Test Connector に送信します。

一度受け入れると、PO の 3 番目の添付が届きます。

PollStartTime

イベント・キューのポーリングを開始する時刻です。形式は HH:MM です。ここで、HH は 0 から 23 時を表し、MM は 0 から 59 分を表します。

このプロパティには必ず有効な値を指定してください。デフォルト値は HH:MM ですが、この値は必ず変更する必要があります。

RequestQueue

統合ブローカーが、ビジネス・オブジェクトをコネクタに送信するときに使用されるキューです。

デフォルト値は CONNECTOR/REQUESTQUEUE です。

RepositoryDirectory

コネクタが XML スキーマ文書を読み取るリポジトリの場所です。この XML スキーマ文書には、ビジネス・オブジェクト定義のメタデータが含まれています。

統合ブローカーが ICS の場合はこの値を <REMOTE> に設定する必要があります。これは、コネクタが InterChange Server リポジトリからこの情報を取得するためです。

統合ブローカーが WebSphere Message Broker または WAS の場合は、この値を `<local directory>` に設定する必要があります。

ResponseQueue

DeliveryTransport が JMS の場合のみ適用可能で、RepositoryDirectory が `<REMOTE>` の場合のみ必須です。

JMS 応答キューを指定します。JMS 応答キューは、応答メッセージをコネクタ・フレームワークから統合ブローカーへデリバリーします。統合ブローカーが ICS の場合、サーバーは要求を送信し、JMS 応答キューの応答メッセージを待ちます。

RestartRetryCount

コネクタによるコネクタ自体の再始動の試行回数を指定します。このプロパティを並列コネクタに対して使用する場合、コネクタのマスター側のアプリケーション固有のコンポーネントがスレーブ側のアプリケーション固有のコンポーネントの再始動を試行する回数が指定されます。

デフォルト値は 3 です。

RestartRetryInterval

コネクタによるコネクタ自体の再始動の試行間隔を分単位で指定します。このプロパティを並列コネクタに対して使用する場合、コネクタのマスター側のアプリケーション固有のコンポーネントがスレーブ側のアプリケーション固有のコンポーネントの再始動を試行する間隔が指定されます。指定可能な値の範囲は 1 から 2147483647 です。

デフォルト値は 1 です。

RHF2MessageDomain

WebSphere Message Brokers および WAS でのみ使用されます。

このプロパティにより、JMS ヘッダーのドメイン名フィールドの値を構成できます。JMS トランスポートを介してデータを WMQI に送信するときに、アダプター・フレームワークにより JMS ヘッダー情報、ドメイン名、および固定値 `mrm` が書き込まれます。この構成可能なドメイン名により、ユーザーは WMQI ブローカーによるメッセージ・データの処理方法を追跡できます。

サンプル・ヘッダーを以下に示します。

```
<mcd><Msd>mrm</Msd><Set>3</Set><Type>
Retek_POPhyDesc</Type><Fmt>CwXML</Fmt></mcd>
```

デフォルト値は `mrm` ですが、このプロパティには `xml` も設定できます。このプロパティは、DeliveryTransport が JMS に設定されており、かつ WireFormat が CwXML に設定されている場合にのみ表示されます。

SourceQueue

DeliveryTransport が JMS で、ContainerManagedEvents が指定されている場合のみ適用されます。

JMS イベント・ストアを使用する JMS 対応コネクタでの保証付きイベント・デリバリーをサポートするコネクタ・フレームワークに、JMS ソース・キューを指定します。詳細については、135 ページの『ContainerManagedEvents』を参照してください。

デフォルト値は `CONNECTOR/SOURCEQUEUE` です。

SynchronousRequestQueue

`DeliveryTransport` が JMS の場合のみ適用されます。

同期応答を要求する要求メッセージを、コネクタ・フレームワークからブローカーに配信します。このキューは、コネクタが同期実行を使用する場合にのみ必要です。同期実行の場合、コネクタ・フレームワークは、`SynchronousRequestQueue` にメッセージを送信し、`SynchronousResponseQueue` でブローカーから戻される応答を待機します。コネクタに送信される応答メッセージには、元のメッセージの ID を指定する相関 ID が含まれています。

デフォルトは `CONNECTORNAME/SYNCHRONOUSREQUESTQUEUE` です。

SynchronousResponseQueue

`DeliveryTransport` が JMS の場合のみ適用されます。

同期要求に対する応答として送信される応答メッセージを、ブローカーからコネクタ・フレームワークに配信します。このキューは、コネクタが同期実行を使用する場合にのみ必要です。

デフォルトは `CONNECTORNAME/SYNCHRONOUSRESPONSEQUEUE` です。

SynchronousRequestTimeout

`DeliveryTransport` が JMS の場合のみ適用されます。

コネクタが同期要求への応答を待機する時間を分単位で指定します。コネクタは、指定された時間内に応答を受信できなかった場合、元の同期要求メッセージをエラー・メッセージとともに障害キューに移動します。

デフォルト値は 0 です。

WireFormat

トランスポートのメッセージ・フォーマットです。

- `RepositoryDirectory` がローカル・ディレクトリーの場合は、設定は `CwXML` になります。
- `RepositoryDirectory` の値が `<REMOTE>` の場合には、設定値は `CwBO` です。

WsifSynchronousRequestTimeout

WAS 統合ブローカーでのみ使用されます。

コネクタが同期要求への応答を待機する時間を分単位で指定します。コネクタは、指定された時間内に応答を受信できなかった場合、元の同期要求メッセージをエラー・メッセージとともに障害キューに移動します。

デフォルト値は 0 です。

XMLNameSpaceFormat

WebSphere Message Brokers および WAS 統合ブローカーでのみ使用されます。

ビジネス・オブジェクト定義の XML 形式でネーム・スペースを short と long のどちらにするかをユーザーが指定できるようにするための、強力なプロパティです。

デフォルト値は short です。

付録 B. コネクタ固有のプロパティ

- 『FRONT ARENA イベント通知のサポート』
- 151 ページの『サービス要求』

この付録では、コネクタ固有プロパティについて説明します。

FRONT ARENA イベント通知のサポート

適切にセットアップされている FRONT ARENA アプリケーションの場合は、新規のオブジェクトの作成または既存のオブジェクトの変更か削除が実行されると、その AMB ミドルウェアを使用してメッセージをパブリッシュします。アダプターのブリッジ・コンポーネントが、そのようなメッセージのすべてにサブスクライブされています。受信された各イベント・メッセージは、WebSphere MQ キューに転送されます。

アダプターによるそれらのイベントの処理を可能にするには、(アダプターのコネクタ固有プロパティ内で) アダプターを構成して、その InputQueue として上記のキューを使用するようにする必要があります。

注: イベントが失われないようにするため、このキューは、永続的 キューとしてセットアップします。

アダプターの標準プロパティ PollStartTime、PollEndTime、および PollFrequency の設定値、およびコネクタ固有プロパティ PollQuantity の値に応じて、このキューのエントリは、アダプターから検索されます。イベント・メッセージは、ビジネス・オブジェクトに変換された後、そのタイプのビジネス・オブジェクトを処理できるすべてのコラボレーションに転送されます。

表 52. コネクタ構成: 標準プロパティ

プロパティ名	説明	有効な値
PollFrequency	イベント・ポーリング・アクション間の時間の長さです。	<ul style="list-style-type: none">• 整数値 (ミリ秒単位)。• No、ポーリングを使用不可にします。• Key、コネクタのコマンド・プロンプト・ウィンドウで、英字 p が入力されたときに限りポーリングを行います。
PollStartTime	イベント・キューのポーリングを開始する時刻です。	HH:MM (HH は 0 から 23、MM は 0 から 59 です) このパラメーターは、有効な時刻が指定されないと使用不可になります。

表 52. コネクタ構成: 標準プロパティ (続き)

プロパティ名	説明	有効な値
PollEndTime	イベント・キューのポーリングを停止する時刻です。	HH:MM (HH は 0 から 23、MM は 0 から 59 です) このパラメーターは、有効な時刻が指定されないと使用不可になります。

表 53. コネクタ構成: コネクタ固有のプロパティ

プロパティ名	説明	デフォルト
InputQueue	1 つ以上のイベント入力キュー。	N/A
PollQuantity	pollForEvents スキャン時に InputQueue プロパティで指定された各キューから取得するメッセージの数。	1

コンポーネント

WebSphere MQ キューは、アダプターのブリッジ・コンポーネントの報告するイベントをアダプターに通知するために使用されます。

アダプターのコネクタ固有プロパティの属性 InputQueue は、このキューを指し示す必要があります。

トリガー

アダプターの InputQueue 内の各エントリは、イベントと見なされます。アダプターの標準プロパティ PollFrequency の値に従って、このキューに対して、ポーリングは、定期的に (プロパティ値 HH:MM で) 行われるか、ユーザーによる明示的な要求 (プロパティ値 KEY の指定時) があったときに行われます。定期的にキューに対してポーリングが行われる場合、標準プロパティの PollStartTime と PollEndTime を使用してポーリングの開始時刻と終了時刻を構成することができます。

Create 通知

新規の FRONT ARENA オブジェクトの作成を示すイベントが検出されると、アダプターは、対応するビジネス・オブジェクトの作成を試行します。ビジネス・オブジェクト名は、前に説明したように、データ・ハンドラーのメタオブジェクトの属性 BOPrefix の値およびネーム・ハンドラーが抽出したオブジェクト名から構成されます。

属性の <business object name>_<verb> が、静的構成メタオブジェクトに存在し、

- その <business object name> 部分が (上記で決定された) ビジネス・オブジェクト名と一致し、
- そのアプリケーション固有情報に、イベント・メッセージの FORMAT フィールドの内容に一致するパラメーター値 (この場合、INSERT) を含むパラメーターの InputFormat を持っている場合、

ビジネス・オブジェクトが作成されます。

新規に作成されたビジネス・オブジェクトの動詞は、メタオブジェクト属性の <verb> 部分となります。ビジネス・オブジェクトは、すべての関係するコラボレーションに送信されるか、アクティブなサブスクリバースが存在しない場合は、UnsubscribedQueue に挿入されます。いずれでもない場合は、イベント・メッセージは拒否され、処理されることなく、ErrorQueue に移動されます。

表 54. 静的構成メタオブジェクト: **MO_FrontArena_DefaultConfig**

プロパティ名	アプリケーション固有情報
FA_trade_Create	InputFormat=INSERT;

Update 通知

新規の FRONT ARENA オブジェクトの更新を示すイベントが検出されると、アダプターは、対応するビジネス・オブジェクトの作成を試行します。ビジネス・オブジェクト名は、前に説明したように、データ・ハンドラーのメタオブジェクトの属性 BOPrefix の値およびネーム・ハンドラーが抽出したオブジェクト名から構成されます。

属性の <business object name>_<verb> が、静的構成メタオブジェクトに存在し、

- その <business object name> 部分が上記で決定されたビジネス・オブジェクト名と一致し、
- そのアプリケーション固有情報に、イベント・メッセージの FORMAT フィールドの内容に一致するパラメーター値 (この場合、UPDATE) を含むパラメーターの InputFormat を持っている場合、

ビジネス・オブジェクトが作成されます。

その動詞は、メタオブジェクト属性の <verb> 部分となります。ビジネス・オブジェクトは、すべての関係するコラボレーションに送信されるか、アクティブなサブスクリバースが存在しない場合は、UnsubscribedQueue に挿入されます。

いずれでもない場合は、イベント・メッセージは拒否され、処理されることなく、ErrorQueue に移動されます。

表 55. 静的構成メタオブジェクト: **MO_FrontArena_DefaultConfig**

プロパティ名	アプリケーション固有情報
FA_trade_Update	InputFormat=UPDATE;

Delete 通知

新規の FRONT ARENA オブジェクトの削除を示すイベントが検出されると、アダプターは、対応するビジネス・オブジェクトの作成を試行します。ビジネス・オブジェクト名は、前に説明したように、データ・ハンドラーのメタオブジェクトの属性 BOPrefix の値およびネーム・ハンドラーが抽出したオブジェクト名から構成されます。

属性の <business object name>_<verb> が、静的構成メタオブジェクトに存在し、

- その <business object name> 部分が上記で決定されたビジネス・オブジェクト名と一致し、
- そのアプリケーション固有情報に、イベント・メッセージの FORMAT フィールドの内容に一致するパラメーター値 (この場合、DELETE) を含むパラメーターの InputFormat を持っている場合、

ビジネス・オブジェクトが作成されます。

その動詞は、メタオブジェクト属性の <verb> 部分となります。ビジネス・オブジェクトは、すべての関係するコラボレーションに送信されるか、アクティブなサブスクライバーが存在しない場合は、UnsubscribedQueue に挿入されます。

表 56. 静的構成メタオブジェクト: **MO_FrontArena_DefaultConfig**

プロパティ名	アプリケーション固有情報
FA_trade_Delete	InputFormat=DELETE;

アーカイブ

有効な WebSphere MQ キューが、アダプターのアプリケーション構成プロパティアーカイブQueue に指定されている場合は、正常に処理されたイベントが、指定されたキューにアーカイブされます。それ以外の場合、アダプターはアーカイブを実行しません。

注: アーカイブ・キューは定期的にクリーンアップすることを推奨します。

リカバリー

アダプターは、WebSphere MQ アダプターの標準的なリカバリー振る舞いを行います。無効または非サポートのフォーマットのために処理できなかったイベント・メッセージは、ErrorQueue がアダプター構成に定義されている場合、そのキューに格納されます。それ以外の場合は、単純に廃棄されます。

アダプターで発生するその他のタイプのエラーは、アダプターのログ・ファイルに記録されます。エラーのタイプと重大度に従って、アダプターは、操作の実行を再試行するか、終了します。

アンサブスクライブ・イベント

イベントをサブスクライブしているコラボレーションがない場合、イベントは、UnsubscribedQueue がアダプター構成に定義されている場合、そのキューに格納されます。それ以外の場合は、廃棄されます。

サブスクライバーが、後の時点でアクティブになると、格納されたイベントは **InputQueue** に移動されて再度処理される場合があります。

サービス要求

アダプターは、コラボレーションから受信したサービス要求をすべて、FRONT ARENA メッセージに変換します。それらのメッセージは、アダプターのブリッジ・コンポーネントに転送されます。静的構成メタオブジェクトに定義されている `OutputQueue` が、その目的に使用されます。詳細については、40 ページの『カスタム・ビジネス・オブジェクト・ハンドラー』を参照してください。

アダプターは、サービス要求を同期式にも非同期式にも処理するようにセットアップすることができます。これは、静的構成メタオブジェクトにパラメーター `ResponseTimeout` を指定するか、省略して行います。詳細については、40 ページの『カスタム・ビジネス・オブジェクト・ハンドラー』を参照してください。そのパラメーターを設定する場合は、アダプターのコネクタ固有プロパティに `ReplyToQueue` も指定する必要があります。

サービス要求を非同期処理するようにセットアップされているアダプターの場合は、要求が成功したと見なされるのは、要求を `OutputQueue` に転送できるときです。 `BON_SUCCESS` が戻されます。

同期モードでは、アダプターは、`ResponseTimeout` に指定された時間の間、アダプターのブリッジ・コンポーネントからの応答を待機します。応答が指定された時間間隔内に受信されると、応答は、呼び出し側コラボレーションに渡されます。それ以外の場合は、障害が発生されたものと想定され、否定応答 (`BON_FAIL`) がコラボレーションに戻されます。

アダプターのブリッジ・コンポーネントは、サービス要求を同期式にも非同期式にも処理するようにセットアップすることができます。両方のコンポーネントのセットアップは、期待通りに処理を実行させるためには、一致する必要があります。

表 57. コネクタ構成: コネクタ固有のプロパティ

プロパティ名	説明	デフォルト
<code>ReplyToQueue</code>	アダプターが、アダプターのブリッジ・コンポーネントに送信された、サービス要求に対する応答をその中に期待するキューです。	N/A

表 58. 静的構成メタオブジェクト: `MO_FrontArena_DefaultConfig`

プロパティ名	アプリケーション固有情報
デフォルト	<code>OutputQueue=queue://DENALI/ FRONTARENA.OUT?targetClient=1;ResponseTimeout =60000;</code>

注: `OutputQueue` 名の後に `?targetClient=1` が続くことが重要です。これにより、アダプターのブリッジ・コンポーネントに送信されるメッセージが予想通りのフォーマットを持つことが保証されます。

Create

Create 動詞を含むビジネス・オブジェクトの処理は、オブジェクトが非同期または同期のどちらの方式で発行されているかによって異なります。

非同期処理: Create の動詞を含むビジネス・オブジェクトに関しては、これがデフォルトのデリバリー・モードです。メッセージは、データ・ハンドラーを使用して、ビジネス・オブジェクトから作成された後、変換メタオブジェクトに指定されている OutputQueue に書き込まれます。詳細については、40 ページの『カスタム・ビジネス・オブジェクト・ハンドラー』を参照してください。メッセージが配信されると、アダプターは BON_SUCCESS、さもなければ BON_FAIL を戻します。

注: 非同期処理では、コネクタには、メッセージが受信されたかどうか、または操作が実行されたかを確認する手段がありません。

同期処理: コネクタ・プロパティに ReplyToQueue が定義されていて、ビジネス・オブジェクトの変換プロパティに ResponseTimeout が存在する場合、アダプターは同期モードで要求を発行します。アダプターはその後で応答を待機して、受信アプリケーションによって適切なアクションが実行されることを確認します。

アダプターのブリッジ・コンポーネントに送信される要求メッセージは、MQMD ヘッダーとその後に続くメッセージ本文から構成され、メッセージ本文は、データ・ハンドラーによってビジネス・オブジェクトから作成された FRONT ARENA メッセージを含みます。

MQMD ヘッダーは、以下の表の情報に基づいて初期化されます。

表 59. 要求メッセージ記述子ヘッダー (MQMD):

フォーマット	説明	値
Format	メッセージ・フォーマット。	指定の <business object name>_<verb> 組み合わせに属する変換メタオブジェクト属性のパラメーター InputFormat の値。この値は、対応する OutputFormat パラメーターを明示的に設定して変更することができます。 注: このフィールドの値は、アダプターのブリッジ・コンポーネントによって無視されます。
Message type	メッセージ・タイプ。	MQMT_DIAGRAM

表 59. 要求メッセージ記述子ヘッダー (MQMD): (続き)

フォーマット	説明	値
Report	必要なレポート・メッセージのオプション。	<p>応答メッセージが期待される場合、このフィールドには次のように値が取り込まれます。</p> <ul style="list-style-type: none"> • 処理が成功した場合に肯定処理レポートが必要なことを示す MQRO_PAN • 処理が失敗した場合に否定処理レポートが必要なことを示す MQRO_NAN • 生成されるレポートの関連 ID を始めに発行された要求のメッセージ ID と等しくすることを示す MQRO_COPY_MSG_ID_TO_CORREL_ID
ReplyToQueue	応答キューの名前。	<p>応答メッセージが期待される場合は、このフィールドにはコネクタ固有プロパティ ReplyToQueue の値が取り込まれます。このフィールドの値は、アダプターのブリッジ・コンポーネントによって無視されます。</p> <p>注: 要求処理結果を戻すため、その構成内に定義されている固定の応答 キューを使用します。</p>
Persistence	メッセージのパーシスタンス	MQPER_PERSISTENT
Expiry	メッセージの存続時間	MQEI_UNLIMITED

ビジネス・オブジェクトの処理が可能な場合は、アダプターのブリッジ・コンポーネントが、MQFB_PAN の設定されたフィードバック・フィールドを含むレポート・メッセージを作成します。さらに、アプリケーションが、作成されたばかりのオブジェクトの属性を含む直列化したビジネス・オブジェクトをメッセージ本文に取り込みます。

注: ただし、この応答メッセージ内の属性のリストは、FRONT ARENA アプリケーション構成に応じて、要求メッセージ内のものとは異なる場合があります。

ビジネス・オブジェクトの処理が不可能な場合は、MQFB_NAN の設定されたフィードバック・フィールドを含むレポート・メッセージが戻されます。メッセージ本文には、障害の原因を説明するエラー・メッセージが含まれます。

いずれの場合も、アプリケーションは、応答メッセージの CorrelationID フィールドを要求メッセージの MessageID に設定し、ReplyToQueue フィールドで指定されたキューにメッセージを送信します。

応答メッセージを検索時に、アダプターは、応答メッセージの correlationID を要求メッセージの messageID に突き合わせます。次に、アダプターは要求を発行したスレッドを通知します。コネクタは、応答のフィードバック・フィールドの設定によって、メッセージの本文にビジネス・オブジェクトとエラー・メッセージのどちらが含まれているかを予測します。

Update

Update 動詞を持つビジネス・オブジェクトの処理は、Create 動詞を持つビジネス・オブジェクトの処理と類似しています。

要求の処理が正常に行われる場合は、応答メッセージ・ヘッダー内のフィードバック・フィールドは、この例では、MQFB_APPL_FIRST + 2 (BON_VALCHANGE) に設定されます。この場合も、メッセージ本文に、更新されたオブジェクトの詳細が含まれます。

Delete

Delete 動詞を持つビジネス・オブジェクトの処理は、Create 動詞を持つビジネス・オブジェクトの処理と類似しています。

Retrieve

Retrieve 動詞を持つビジネス・オブジェクトは、同期処理のみをサポートします。基本的に、アダプターは、この動詞を持つビジネス・オブジェクトを、Create、Update、および Delete の各動詞に対して定義されている同期処理の場合と同様に処理します。ただし、戻されるフィードバック・コードが異なります。

Exists

この動詞は、サポートされていません。

Retrieve by content

この動詞は、サポートされていません。FRONT ARENA アプリケーションの観点からは、Retrieve と RetrieveByContent の間に違いはありません。

以下の表は、アダプターのブリッジ・コンポーネントが配信するフィードバック・コードの概要を示しています。

表 60. FRONT ARENA とブリッジ・コンポーネント間のフィードバック・コード

処理結果	フィードバック・コード	メッセージの本文
MQFB_PAN	BON_SUCCESS	Insert、Delete、または Retrieve が成功しました。(Insert、Delete、or Retrieve successful.)
MQFB_NAN	BON_FAIL	Insert、Update、Delete、または Retrieve が失敗しました。(Insert、Update、Delete or Retrieve failed.) Retrieve の場合: 複数の一致 と BO 以外のエラーは存在しません。(For Retrieve:Failures other than multiple hits and BO do not exist.)
MQFB_APPL_FIRST + 2	BON_VALCHANGE	更新は成功しました (The update successful)
MQFB_APPL_FIRST + 4	BON_MULTIPLE_HITS	検索では、一致するオブジェクトが複数検出されました。(The retrieve found several matching objects.)

表 60. FRONT ARENA とブリッジ・コンポーネント間のフィードバック・コード (続き)

処理結果	フィードバック・コード	メッセージの本文
MQFB_APPL_FIRST + 6	BON_BO_DOES_ NOT_EXIST	検索では、一致するオブジェクト が検出されませんでした。(The retrieve did not find any matching objects.)

付録 C. ビジネス・オブジェクトの作成および変更

- 『概要』
- 『アダプターのビジネス・オブジェクトの構造』

この付録では、ビジネス・オブジェクトとその構造について説明します。

概要

コネクターには、ビジネス・オブジェクトの例のみが付属しています。システム・インテグレーター、コンサルタント、またはお客様が、ビジネス・オブジェクトを構築する必要があります。

コネクターは、メタデータ主導型コネクターです。WebSphere Business Integration システムのビジネス・オブジェクトにおいては、メタデータはアプリケーションに関するデータであり、ビジネス・オブジェクト定義に格納され、コネクターとアプリケーションの間の通信を支援します。メタデータ主導型のコネクターによって処理される各ビジネス・オブジェクトは、コネクター内にハードコーディングされた命令ベースではなく、ビジネス・オブジェクト定義内にエンコードされたメタデータをベースにしてサポートされています。

ビジネス・オブジェクトのメタデータには、ビジネス・オブジェクトの構造、その属性プロパティの設定、およびアプリケーション固有情報の内容が含まれます。コネクターはメタデータ主導型のため、コネクターのコーディングを変更しなくても、新規ビジネス・オブジェクトや変更されたビジネス・オブジェクトを処理できます。しかし、コネクターに構成されたデータ・ハンドラーは、コネクターのビジネス・オブジェクトの構造、オブジェクトのカーディナリティー、アプリケーション固有情報のフォーマット、およびビジネス・オブジェクトのデータベース表記について、ある条件を前提として動作します。したがって、WebSphere MQ のビジネス・オブジェクトを作成または変更する場合、変更の内容はコネクターに対して定められている規則に準拠している必要があります。準拠していない場合、コネクターは新規ビジネス・オブジェクトや変更されたビジネス・オブジェクトを正しく処理できません。

この章では、コネクターによるビジネス・オブジェクトの処理方法と、コネクターの前提事項について説明します。この情報は、新規ビジネス・オブジェクトをインプリメントする際のガイドとして利用できます。

アダプターのビジネス・オブジェクトの構造

アダプターをインストールした後で、ビジネス・オブジェクトを作成する必要があります。構成されるデータ・ハンドラーについての要件を除いては、ビジネス・オブジェクトの構造に関する要件はありません。コネクターが処理するビジネス・オブジェクトは、InterChange Server によって許可されている任意の名前を持つことができます。

アダプターはキューからメッセージを検索し、(メタオブジェクトによって定義されている) ビジネス・オブジェクトにメッセージの内容を取り込もうとします。厳密に言えば、コネクターはビジネス・オブジェクトの構造を制御したり、ビジネス・オブジェクトの構造に影響を及ぼすことはありません。それらの機能は、コネクターのデータ・ハンドラーの要件と、メタオブジェクト定義によって提供されます。実際には、ビジネス・オブジェクト・レベルのアプリケーション情報はありません。より正確に言えば、ビジネス・オブジェクトを検索して渡すときのコネクターの主な役割は、メッセージをビジネス・オブジェクトに変換する (およびその逆の) 処理中に発生するエラーをモニターすることです。

ビジネス・オブジェクトの例

このセクションでは、Name-Value データ・ハンドラーを持つコネクターが処理するビジネス・オブジェクトの例を示します。

```
[ReposCopy]
Version = 3.0.0
[End]
[BusinessObjectDefinition]
Name = FA_trade
Version = 3.0.0
AppSpecificInfo = TN=trade

    [Attribute]
    Name = creattime
    Type = Date
    Cardinality = 1
    MaxLength = 255
    IsKey = false
    IsForeignKey = false
    IsRequired = false
    AppSpecificInfo = CN=creat_time
    IsRequiredServerBound = false
    [End]

    [Attribute]
    Name = creatusrnbr
    Type = Integer
    Cardinality = 1
    MaxLength = 255
    IsKey = false
    IsForeignKey = false
    IsRequired = false
    AppSpecificInfo = CN=creat_usrnbr
    IsRequiredServerBound = false
    [End]

    [Attribute]
    Name = updattime
    Type = Date
    Cardinality = 1
    MaxLength = 255
    IsKey = false
    IsForeignKey = false
    IsRequired = false
    AppSpecificInfo = CN=updat_time
    IsRequiredServerBound = false
    [End]

    [Attribute]
    Name = updatusrnbr
    Type = Integer
    Cardinality = 1
```

```
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=updat_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = protection
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=protection
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = ownerusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=owner_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = foureyeon
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=four_eye_on
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = authorizerusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=authorizer_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = trdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = true
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = prfnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=prfnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = insaddr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = instr_insid
Type = String
Cardinality = 1
MaxLength = 39
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr.insid
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = instr_instype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr.instype
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = instr_isin
Type = String
Cardinality = 1
MaxLength = 20
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr.isin
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = instr_externid1
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
```

```
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr.extern_id1
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = instr_externid2
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr.extern_id2
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = acquireday
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=acquire_day
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = acquirerptynbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=acquirer_ptynbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = acquirer_ptyid
Type = String
Cardinality = 1
MaxLength = 19
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=acquirer_ptynbr.ptyid
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = curr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=curr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
```

Name = curr_insid
Type = String
Cardinality = 1
MaxLength = 39
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=curr.insid
IsRequiredServerBound = false
[End]

[Attribute]
Name = valueday
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=value_day
IsRequiredServerBound = false
[End]

[Attribute]
Name = time
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=time
IsRequiredServerBound = false
[End]

[Attribute]
Name = quantity
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=quantity
IsRequiredServerBound = false
[End]

[Attribute]
Name = price
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=price
IsRequiredServerBound = false
[End]

[Attribute]
Name = premium
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false

```
AppSpecificInfo = CN=premium
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = fee
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=fee
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = bofxrate
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=bo_fx_rate
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = botrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=bo_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = status
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=status
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = counterpartyptynbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=counterparty_ptynbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = traderusrnbr
Type = Integer
```

```
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=trader_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = trader_userid
Type = String
Cardinality = 1
MaxLength = 19
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=trader_usrnbr.userid
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = optkey1chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=optkey1_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = optkey2chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=optkey2_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = optkey3chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=optkey3_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = optkey4chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=optkey4_chlnbr
IsRequiredServerBound = false
```

[End]

```
[Attribute]
Name = hedgetrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=hedge_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = trxtrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=trx_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = openingbotrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=opening_bo_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = optionalkey
Type = String
Cardinality = 1
MaxLength = 31
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=optional_key
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = type
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = marketptynbr
Type = Integer
Cardinality = 1
MaxLength = 255
```

```
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=market_ptynbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = origin
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=origin
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = assinf
Type = String
Cardinality = 1
MaxLength = 35
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=assinf
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = brokerptynbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=broker_ptynbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = bookvalue
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=book_value
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = bookday
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=book_day
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = documenttypechlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=document_type_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = guarantorptynbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=guarantor_ptynbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = reclaccnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rec1_accnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rec2accnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rec2_accnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = pay1accnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=pay1_accnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = pay2accnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
```

```
IsRequired = false
AppSpecificInfo = CN=pay2_accnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = calcagent
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=calcagent
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = cfnetting
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=cfnetting
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = creditnetting
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=credit_netting
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = specialterms
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=special_terms
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = paymentcount
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=payment_count
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = text1
```

```
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=text1
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = text2
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=text2
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = yourref
Type = String
Cardinality = 1
MaxLength = 31
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=your_ref
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = hedgecfwnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=hedge_cfwnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = contracttrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=contract_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = clearingtypechlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=clearing_type_chlnbr
```

```
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = clearinglocationchlnbr  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=clearing_location_chlnbr  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = executiontime  
Type = Date  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=execution_time  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = netpremium  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=net_premium  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = aggregate  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=aggregate  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = aggregatetrnbr  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=aggregate_trdnbr  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = aggregatepl  
Type = Double  
Cardinality = 1
```

```
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=aggregate_pl
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = discount
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=discount
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = salescredit
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=sales_credit
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = salespersonusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=sales_person_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = ordnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ordnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = settleseqnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=settle_seqnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = archivestatus
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=archive_status
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = originalcurr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=original_curr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = tradecurr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=trade_curr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = odseqnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=od_seqnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = mirrortrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=mirror_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = correctiontrdnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
```

```
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=correction_trdnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = haircut
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=haircut
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rightofsubstitution
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=right_of_substitution
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = salesmargin
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=sales_margin
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = iasclasschlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ias_class_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = flataccrued
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=flat_accrued
IsRequiredServerBound = false
[End]
```

```
[Attribute]
```

```

Name = instrument
Type = FA_instrument
ContainedObjectVersion = 3.0.0
Relationship = Containment
Cardinality = N
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=instrument
IsRequiredServerBound = false
[End]

[Attribute]
Name = ObjectEventId
Type = String
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
IsRequiredServerBound = false
[End]

[Verb]
Name = Create
AppSpecificInfo = FAVERB=insert
[End]

[Verb]
Name = Delete
AppSpecificInfo = FAVERB=delete
[End]

[Verb]
Name = Retrieve
AppSpecificInfo = FAVERB=select
[End]

[Verb]
Name = Update
AppSpecificInfo = FAVERB=update
[End]
[End]
[BusinessObjectDefinition]
Name = FA_instrument
Version = 3.0.0
AppSpecificInfo = TN=instrument

[Attribute]
Name = creattime
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=creat_time
IsRequiredServerBound = false
[End]

[Attribute]
Name = creatusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false

```

```
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=creat_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = updatime
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=updat_time
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = updatusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=updat_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = protection
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=protection
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = ownerusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=owner_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = foureyeon
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=four_eye_on
IsRequiredServerBound = false
[End]
```

```
[Attribute]
```

```
Name = authorizerusrnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=authorizer_usrnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = insid
Type = String
Cardinality = 1
MaxLength = 39
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insid
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = insaddr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = true
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=insaddr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = generic
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=generic
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = notional
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=notional
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = instype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
```

```
AppSpecificInfo = CN=instype
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = curr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=curr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = quotetype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=quote_type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = ratetype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rate_type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = otc
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=otc
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = mtmfromfeed
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=mtm_from_feed
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = spotbankingdaysoffset
Type = Integer
```

```
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=spot_banking_days_offset
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = externid1
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=extern_id1
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = externid2
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=extern_id2
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = suspended
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=suspended
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = productchlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=product_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = issuer
Type = String
Cardinality = 1
MaxLength = 29
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=issuer
IsRequiredServerBound = false
```

```

[End]

[Attribute]
Name = issuerptynbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=issuer_ptynbr
IsRequiredServerBound = false
[End]

[Attribute]
Name = contrsize
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=contr_size
IsRequiredServerBound = false
[End]

[Attribute]
Name = physcontrsize
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=phys_contr_size
IsRequiredServerBound = false
[End]

[Attribute]
Name = undinsaddr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=und_insaddr
IsRequiredServerBound = false
[End]

[Attribute]
Name = undinstype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=und_instype
IsRequiredServerBound = false
[End]

[Attribute]
Name = monthchar
Type = String
Cardinality = 1
MaxLength = 1

```

```
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=month_char
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = settlement
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=settlement
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = paytype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=paytype
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = expday
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=exp_day
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = expperiodunit
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=exp_period_unit
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = expperiodcount
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=exp_period_count
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = paydayoffset
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=pay_day_offset
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = payperiodunit
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=pay_period_unit
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = payperiodcount
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=pay_period_count
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = calloption
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=call_option
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = exercisetype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=exercise_type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = strikeprice
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
```

```
IsRequired = false
AppSpecificInfo = CN=strike_price
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = strikecurr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=strike_curr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = striketype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=strike_type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = exotictype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=exotic_type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = digital
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=digital
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = barrier
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=barrier
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = payout
```

```
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=payout
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = callable
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=callable
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = putable
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=putable
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = indexfactor
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=index_factor
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = indextype
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=index_type
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = roundclean
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=round_clean
```

```

IsRequiredServerBound = false
[End]

[Attribute]
Name = roundpremium
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=round_premium
IsRequiredServerBound = false
[End]

[Attribute]
Name = beta
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=beta
IsRequiredServerBound = false
[End]

[Attribute]
Name = coupons
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=coupons
IsRequiredServerBound = false
[End]

[Attribute]
Name = couprate
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=coup_rate
IsRequiredServerBound = false
[End]

[Attribute]
Name = facevalue
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=face_value
IsRequiredServerBound = false
[End]

[Attribute]
Name = issueday
Type = Date
Cardinality = 1

```

```
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=issue_day
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = lastcoupday
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=last_coup_day
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = daycountmethod
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=daycount_method
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = accruedarrear
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=accrued_arrear
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = freetext
Type = String
Cardinality = 1
MaxLength = 19
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=free_text
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = settlecalnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=settle_calnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = openend
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=open_end
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = categorychlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=category_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = excoupmethod
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ex_coup_method
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = excoupperiodunit
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ex_coup_period_unit
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = excoupperiodcount
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ex_coup_period_count
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rate
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
```

```
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rate
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = refprice
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ref_price
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = refvalue
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ref_value
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rating1chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rating1_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rating2chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rating2_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rating3chlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rating3_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
```

Name = ytmmethod
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=ytm_method
IsRequiredServerBound = false
[End]

[Attribute]
Name = archivestatus
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=archive_status
IsRequiredServerBound = false
[End]

[Attribute]
Name = totalissued
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=total_issued
IsRequiredServerBound = false
[End]

[Attribute]
Name = pricedifflimitabs
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=price_diff_limit_abs
IsRequiredServerBound = false
[End]

[Attribute]
Name = pricedifflimitrel
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=price_diff_limit_rel
IsRequiredServerBound = false
[End]

[Attribute]
Name = shortsell
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false

```
AppSpecificInfo = CN=short_sell  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = payoffsetmethod  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=pay_offset_method  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = startday  
Type = Date  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=start_day  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = settlecategorychlnbr  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=settle_category_chlnbr  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = conversionratio  
Type = Double  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=conversion_ratio  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = originalcurr  
Type = Integer  
Cardinality = 1  
MaxLength = 255  
IsKey = false  
IsForeignKey = false  
IsRequired = false  
AppSpecificInfo = CN=original_curr  
IsRequiredServerBound = false  
[End]
```

```
[Attribute]  
Name = combcategorychlnbr  
Type = Integer
```

```
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=comb_category_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = pricefindingchlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=price_finding_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = isin
Type = String
Cardinality = 1
MaxLength = 20
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=isin
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = dividendfactor
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=dividend_factor
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = fixfxrate
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=fix_fx_rate
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = rebate
Type = Double
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=rebate
IsRequiredServerBound = false
```

[End]

```
[Attribute]
Name = exptime
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=exp_time
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = fixfx
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=fix_fx
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = roundaccrued
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=round_accrued
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = bondfuturemarket
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=bond_future_market
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = incomplete
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=incomplete
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = senioritychlnbr
Type = Integer
Cardinality = 1
MaxLength = 255
```

```
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=seniority_chlnbr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = originalinsaddr
Type = Integer
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=original_insaddr
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = datefrom
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=date_from
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = dateto
Type = Date
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
AppSpecificInfo = CN=date_to
IsRequiredServerBound = false
[End]
```

```
[Attribute]
Name = ObjectEventId
Type = String
Cardinality = 1
MaxLength = 255
IsKey = false
IsForeignKey = false
IsRequired = false
IsRequiredServerBound = false
[End]
```

```
[Verb]
Name = Create
AppSpecificInfo = FAVERB=insert
[End]
```

```
[Verb]
Name = Delete
AppSpecificInfo = FAVERB=delete
[End]
```

```
[Verb]
Name = Retrieve
AppSpecificInfo = FAVERB=select
```

```
[End]
[Verb]
Name = Update
AppSpecificInfo = FAVERB=update
[End]
[End]
```

特記事項

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

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。

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

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 Burlingame Laboratory Director
IBM Burlingame Laboratory
577 Airport Blvd., Suite 800
Burlingame, CA 94010
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せんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

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

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

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。

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。

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、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・サービスを含むアプリケーション・ソフトウェアを書くことができます。

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

以下は、IBM Corporation の商標です。

IBM
IBM ロゴ
AIX
CrossWorlds
DB2
DB2 Universal Database
Lotus
Lotus Domino
Lotus Notes
MQIntegrator
MQSeries
Tivoli
WebSphere

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

MMX、Pentium および ProShare は、Intel Corporation の米国およびその他の国における商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

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。



WebSphere Business Integration Adapter Framework V2.4.0

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

アダプター
アーキテクチャー 3
サポートされる動詞 13
トレース 125
フレームワーク 2
用語 1
アプリケーション固有ビジネス・オブジェクトの直列化 124
アプリケーションとアダプター間の通信 10
アプリケーションのタイムアウト 121
アンサブスクライブ
キュー管理 121
ビジネス・オブジェクト 122
イベント
関連キュー 123
サブスクリプションおよびフィルター処理 73
サブスクリプションの欠落 122
重複デリバリー 123
通知 6, 10
表 97
ブリッジによる処理 72
メッセージの失敗 122
ID 98
イベント・ストア
障害 12
電子メール・メールボックス 97
フラット・ファイル 97
JMS 12, 95, 99
エラー
アダプター 119
アプリケーションのトレース 120
ブリッジ 119
メッセージ処理 121
ロギング 12
ErrorQueue 120
エラーのロギング 12
応答メッセージ記述子ヘッダー (MQMD) 15

[カ行]

関連文書 v
記述子ヘッダー (MQMD) 15
キューの URI (Uniform Resource Identifiers)
URI 99
構成
コネクター固有のプロパティ 89, 90
静的メタオブジェクト 108
標準コネクター・プロパティ 89
ブリッジ 69
InDoubtEvents 11
コネクター
コントローラー 3
再始動 12
プロパティ
JMS イベント・ストア 95
コネクター構成プロパティ
データ・ハンドラー 95
AgentTraceLevel 89
ApplicationUserName 91
ArchiveQueue 91
channel 91
ConfigurationMetaObject 92
ContainerManagedEvents 95
DataHandlerClassName 92
DataHandlerConfigMO 92
DataHandlerConfigMOName 96
DataHandlerMimeType 92
DeliveryTransport 12, 95, 97
DHClass 96
DuplicateEventElimination 97
ErrorQueue 92
FeedbackCodeMappingMO 92
MimeType 96
MonitorQueue 97
PollQuantity 95, 97
SourceQueue 95
コネクター固有のプロパティリスト 151

[サ行]

再始動
コネクター 12
作業ロードマップ 1
サポートされていない動詞
Exists 154
RetrieveByContent 154

失敗
始動時 11
始動
失敗 11
処理
データ・ハンドラー 16
同期 6
非同期 6
スタンザ 70
静的構成
メタオブジェクト 151
静的なコネクター 101
静的メタオブジェクト
構成 40, 108
構造 107
作成 106, 108
セッション、トランザクション 11

[タ行]

タイムアウト 81, 121, 124
要求 125
多目的インターネット・メール
拡張仕様 (MIME) フォーマット 96
重複イベントの初期化 97
通信
アプリケーションとアダプター間 7, 10
障害 122
接続の喪失 121
ブリッジ 81
通知
イベント 6
要求 49
データ・ハンドラー 39, 40
構成
メタオブジェクト 46
処理 16
動作の仕組み 8
入力キューへのマッピング 108
メッセージの処理 43
メッセージ変換 122
JMS ストア 95
データ・ハンドラー・プロパティ
保証付きイベント・デリバリー 95
同期した要求による対話 6
統合ブローカー 3
動詞
create, update, delete 15
動的な子メタオブジェクト 101

トポロジー
マシン 1 台 21
トランザクション・セッション 11

[ナ行]

入力キュー
データ・ハンドラーのマッピング 108

[ハ行]

ビジネス・オブジェクト
アンサブスクライブ 122
構造 39
サポートされる動詞 13
処理 15
データ・ハンドラー
処理 45
定義の作成 39
ネストされたリスト 45
メタデータ 39
メッセージ変換 9
メッセージ・ヘッダーとの関連 7
要求 13
例 45
create、update、delete 動詞 13
delete 154
exists 154
retrieve 154
RetrieveByContent 154
update 動詞 154
非同期メッセージ
デリバリー 13
FRONT ARENA 4
表記上の規則 vi
標準コネクタ・プロパティ 89
フィードバック・コード
カスタマイズ 16
リスト 154
WebSphere MQ 15
フィールドの説明
Format フィールド 15
MessageType 15
ブリッジ 8
イベント処理 72
イベント通知メッセージ 81
応答メッセージ 82
構成 69、70
構成ファイル 69
トレース 126
必須パラメーター 69
メッセージ 41、69
要求処理 74
要求メッセージ 82

ポーリング
重複イベントの初期化 97
保証付きイベント・デリバリー 96、98
保証付きイベント・デリバリー 12、94
構成プロパティ 95
データ・ハンドラー・プロパティ 95
非 JMS イベント・ストア
非 JMS イベント・ストア 97
ポーリング効果 96
ContainedManagedEvents 96
JMS イベント・ストア 95

[マ行]

マシン 1 台のトポロジー 21
メソッド
doForVerb() 13
メタオブジェクト
構成 101
サポートされているプロパティ 102
静的構成 151
静的なコネクタ 101
データ・ハンドラーの構成 46
動的な子 101
メタデータ 39
メッセージ
アーカイブ 12
アーカイブ、ArchiveQueue 12
アダプターのトレース 125
エラー処理 121
検索 10
pollForEvents() 10
構造 43
再処理 11
失敗した要求 125
データ・ハンドラー障害 122
同期デリバリー 14
非同期デリバリー 13
ブリッジ 41、69
ブリッジによる要求処理 74
ブリッジのイベント通知 81
ブリッジの応答メッセージ 82
ブリッジのトレース 126
ブリッジの要求メッセージ 82
ヘッダー 7、15
変換 9、44
無視 11
戻り 9
pollForEvents() 9
要求 9
要求キュー 125
リカバリー 11
AMB ミドルウェア 49
AMB メッセージ ID 77

メッセージ (続き)
InProgressQueue 93
InputQueue 93
MQMD 14
MQSeries 39
PollQuantity 94
ReplyToQueue 94
SMTP 78
UnsubscribedQueue 94
UseDefaults 94
メッセージのアーカイブ 12
メッセージの再処理 11
文字コード・セット 20

[ヤ行]

要求
アダプターの終了 124
処理 (AMBA) 86
処理上の問題 123
タイムアウト 125
通知 49
ビジネス・オブジェクト 13
MQMD 14
要求メッセージ記述子ヘッダー (MQMD) 14
用語
コネクタ 2
コネクタ・フレームワーク 2

[ラ行]

ロケール依存データ 20

[数字]

2 バイト文字セット 20

A

Adapter Development Kit (ADK) 2
AgentTraceLevel 89
AMB
アダプター (AMBA) 85
クライアント・アプリケーション 10
構成 85
ミドルウェア 49
メッセージ ID 77
AMBA
イベント処理 85
構成 85
パラメーター 85
要求処理 86
ApplicationUserName 91
ArchiveQueue 12、91

Arena Message Broker (AMB) 2

C

channel 91
ConfigurationMetaObject 92
ContainerManagedEvents
構成 95
create 動詞 152

D

DataHandlerClassName 92
DataHandlerConfigMO 92
DataHandlerConfigMOName 96
DataHandlerMimeType 92
delete 動詞 154
DeliveryTransport 12, 95, 97
DHClass 構成プロパティ 96
doForVerb() 13
DuplicateEventElimination 97

E

ErrorQueue 92
exists 動詞 154

F

FeedbackCodeMappingMO 92
FRONT ARENA
構成 85
AMB 10, 85
AMB アダプター 85

G

gotApplEvent() 9

H

HostName 93

I

InDoubtEvents 11, 93
InProgressQueue 93
InputQueue 93
InterChange Server (ICS)
統合ブローカー 90
InterChange Server 固有の状況コード 92

J

Java Message Service
(JMS) 7
JMS
イベント・ストア 12, 95
キュー 12
キュー・セッション 9
保証付きイベント・デリバリー 95
Java Message Service 7

M

MessageType フィールド 15
MimeType 96
MonitorQueue コネクター構成プロパティ
ー 97
MQMD ヘッダーの初期化 152

P

pollForEvents() 9, 10, 98
例外 97
PollQuantity 94, 95
コネクター構成 97

R

ReplyToQueue 94
retrieve 動詞 154
RetrieveByContent 動詞 154

S

SMTP メッセージのアラート 78
SourceQueue コネクター構成プロパティ
ー 95

T

TRADE エレメント 46

U

UnsubscribedQueue 94
update 動詞 154
URI
キュー・プロパティ名 99
UseDefaults 94

W

WebSphere Business Integration システム

WebSphere MQ

キュー・プロパティ名 99
許可 81
フィードバック・コード 15
Integrator Broker 3



Printed in Japan