



BEA WebLogic Integration™

**B2B Integration □
ジック プラグ イン プ
ログ ラ ミ ン グ ガ イ ド**

著作権

Copyright © 2002, BEA Systems, Inc. All Rights Reserved.

限定的権利条項

本ソフトウェアおよびマニュアルは、BEA Systems, Inc. 又は日本ビー・イー・エー・システムズ株式会社（以下、「BEA」といいます）の使用許諾契約に基づいて提供され、その内容に同意する場合にのみ使用することができ、同契約の条項通りにのみ使用またはコピーすることができます。同契約で明示的に許可されている以外の方法で同ソフトウェアをコピーすることは法律に違反します。このマニュアルの一部または全部を、BEA Systems, Inc. からの書面による事前の同意なしに、複写、複製、翻訳、あるいはいかなる電子媒体または機械可読形式への変換も行うことはできません。

米国政府による使用、複製もしくは開示は、BEA の使用許諾契約、および FAR 52.227-19 の「Commercial Computer Software-Restricted Rights」条項のサブパラグラフ (c)(1)、DFARS 252.227-7013 の「Rights in Technical Data and Computer Software」条項のサブパラグラフ (c)(1)(ii)、NASA FAR 補遺 16-52.227-86 の「Commercial Computer Software--Licensing」条項のサブパラグラフ (d)、もしくはそれらと同等の条項で定める制限の対象となります。

このマニュアルに記載されている内容は予告なく変更されることがあり、また BEA による責務を意味するものではありません。本ソフトウェアおよびマニュアルは「現状のまま」提供され、商品性や特定用途への適合性を始めとする（ただし、これらには限定されない）いかなる種類の保証も与えません。さらに、BEA は、正当性、正確さ、信頼性などについて、本ソフトウェアまたはマニュアルの使用もしくは使用結果に関していかなる確約、保証、あるいは表明も行いません。

商標または登録商標

BEA、Jolt、Tuxedo、および WebLogic は BEA Systems, Inc. の登録商標です。BEA Builder、BEA Campaign Manager for WebLogic、BEA eLink、BEA Manager、BEA WebLogic Commerce Server、BEA WebLogic Enterprise、BEA WebLogic Enterprise Platform、BEA WebLogic Express、BEA WebLogic Integration、BEA WebLogic Personalization Server、BEA WebLogic Platform、BEA WebLogic Server、BEA WebLogic Workshop および How Business Becomes E-Business は、BEA Systems, Inc の商標です。

その他の商標はすべて、関係各社が著作権を有します。

BE2B Integration ロジック プラグイン プログラミング ガイド

パート番号	日付	ソフトウェアのバージョン
なし	2002 年 6 月	7.0

目次

このマニュアルの内容

対象読者.....	v
e-docs Web サイト.....	vi
このマニュアルの印刷方法.....	vi
関連情報.....	vi
サポート情報.....	vii
表記規則.....	vii

1. 概要

アプリケーションの種類.....	1-1
ロジック プラグイン.....	1-2

2. ビジネス メッセージのルーティングとフィルタ処理

ビジネス メッセージおよびメッセージ エンベロープ.....	2-1
実行時のメッセージ処理.....	2-2
送信側.....	2-7
受信側.....	2-12
メッセージコンテキスト ドキュメントでの作業.....	2-15
XPath 式での作業.....	2-16
XPath 式について.....	2-16
メッセージ XPath 式の作成.....	2-20
トレーディング パートナ XPath 式の作成.....	2-21
ビジネス プロトコル XPath 式の作成.....	2-22

3. ロジック プラグインの作成と追加（非推奨）

ロジック プラグインについて.....	3-1
ロジック プラグインの概要.....	3-1
ロジック プラグインが処理するタスク.....	3-2
チェーン.....	3-3
システム ロジック プラグインとカスタム ロジック プラグイン.....	3-6
ロジック プラグイン API.....	3-7
ロジック プラグインについてのルールおよびガイドライン.....	3-9

ロジック プラグインの開発と管理.....	3-12
ロジック プラグインのプログラミング ステップ	3-12
管理タスク	3-18

索引

このマニュアルの内容

このマニュアルでは、ビジネス メッセージを交換し、WebLogic Integration™ システムの B2B Integration をサポートする実行時アクティビティをモニタするためのアプリケーションの開発方法について説明します。

注意： カスタム ロジック プラグインは、WebLogic Integration の本リリースから非推奨になっています。代替機能に関する情報については、『*BEA WebLogic Integration リリース ノート*』を参照してください。

このマニュアルの内容は以下のとおりです。

- 第 1 章「概要」では、WebLogic Integration 環境のためのアプリケーションの開発について概説します。
- 第 2 章「ビジネス メッセージのルーティングとフィルタ処理」では、ルーティングおよびフィルタ処理が WebLogic Integration 環境でどのように B2B Integration をサポートしているかについて説明します。
- 第 3 章「ロジック プラグインの作成と追加（非推奨）」では、B2B 統合のために WebLogic Integration システムを通過するビジネス メッセージを処理する方法について説明します。

対象読者

このマニュアルは主に、次のユーザを対象としています。

- WebLogic Integration Studio を使用して WebLogic Integration 環境に統合できるワークフローを設計するビジネス プロセス設計者
- WebLogic Integration 環境でビジネス メッセージの交換を管理したり、実行時の統計をモニタしたりするための Java アプリケーションを記述するアプリケーション開発者
- WebLogic Integration アプリケーションを設定および管理するシステム管理者

WebLogic Integration のアーキテクチャの概要については、『*WebLogic Integration 入門*』を参照してください。

e-docs Web サイト

BEA の製品マニュアルは、以下の場所から入手できます。

<http://edocs.beasys.co.jp/e-docs/index.html>

このマニュアルの印刷方法

このマニュアルの PDF 版は、Web サイトで入手できます。PDF を Adobe Acrobat Reader で開くと、マニュアルの全体または一部分を印刷できます。Adobe Acrobat Reader がない場合は、次の URL にある Adobe の Web サイトから無料で入手できます。

<http://www.adobe.co.jp/>

または、Web ブラウザの [ファイル | 印刷] オプションを使用すると、Web ブラウザからこのマニュアルの HTML ファイルを一度に 1 ファイルずつ印刷できます。

関連情報

Java 2 Enterprise Edition (J2EE)、eXtensible Markup Language (XML)、および Java プログラミングについての詳細は、次の URL にある Javasoft Web サイトを参照してください。

<http://java.sun.com>

サポート情報

WebLogic Integration のドキュメントに関するユーザからのフィードバックは弊社にとって非常に重要です。質問や意見などがあれば、電子メールで docsupport-jp@bea.com までお送りください。寄せられた意見については、WebLogic Integration のドキュメントを作成および改訂する BEA の専門の担当者が直に目を通します。

電子メールのメッセージには、製品のバージョン番号をお書き添えください。

本リリースの WebLogic Integration について不明な点がある場合、または WebLogic Integration のインストールおよび動作に問題がある場合は、次の URL にある BEA WebSUPPORT を通じて BEA カスタマサポートまでお問い合わせください。

<http://websupport.bea.com/custsupp>

カスタマサポートでは以下の情報をお尋ねしますので、お問い合わせの際はあらかじめご用意ください。

- お名前、電子メール アドレス、電話番号、ファクス番号
- 会社の名前と住所
- お使いの機種とコード番号
- 製品の名前とバージョン
- 問題の状況と表示されるエラー メッセージの内容

表記規則

このマニュアルでは、全体を通して以下の表記規則が使用されています。

表記法	適用
太字	用語集で定義されている用語を示す。

表記法	適用
[Ctrl] + [Tab]	複数のキーを同時に押すことを示す。
<i>斜体</i>	強調または書籍のタイトルを示す。
等幅テキスト	コード サンプル、コマンドとそのオプション、データ構造体とそのメンバー、データ型、ディレクトリ、およびファイル名とその拡張子を示す。等幅テキストはキーボードから入力するテキストも示す。 <i>例</i> <pre>#include <iostream.h> void main () the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float</pre>
太字の等幅テキスト	コード内の重要な箇所を示す。 <i>例</i> <pre>void commit ()</pre>
<i>斜体の等幅テキスト</i>	コード内の変数を示す。 <i>例</i> <pre>String expr</pre>
すべて大文字のテキスト	デバイス名、環境変数、および論理演算子を示す。 <i>例</i> <pre>LPT1 SIGNON OR</pre>
{ }	構文の中で複数の選択肢を示す。実際には、この括弧は入力しない。
[]	構文の中で任意指定の項目を示す。実際には、この括弧は入力しない。 <i>例</i> <pre>buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...</pre>
	構文の中で相互に排他的な選択肢を区切る。実際には、この記号は入力しない。

表記法	適用
...	<p>コマンドラインで以下のいずれかを示す。</p> <ul style="list-style-type: none"> ■ 引数を複数回繰り返すことができる ■ 任意指定の引数が省略されている ■ パラメータや値などの情報を追加入力できる <p>実際には、この省略記号は入力しない。</p> <p><i>例</i></p> <pre>buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...</pre>
. . <p>.</p>	<p>コード サンプルまたは構文で項目が省略されていることを示す。実際には、この省略記号は入力しない。</p>



1 概要

以下の節では、ロジックプラグインのプログラミングの概要を説明します。

- アプリケーションの種類
- ロジックプラグイン

注意： カスタム ロジック プラグインは、**WebLogic Integration** の本リリースから非推奨になっています。代替機能に関する情報については、『*BEA WebLogic Integration リリース ノート*』を参照してください。

アプリケーションの種類

WebLogic Integration では、**B2B Integration** に使用できる以下のアプリケーションが提供されています。

- ロジックプラグイン—カスタマイズされるルーティング、フィルタ処理、および情報処理に使用します。このマニュアルでは、ロジックプラグインの概要を説明します。
- 管理アプリケーション— **B2B Integration** のアクティビティのモニタに使用します。BEA によって実装されている MBeans に基づいています。
- メッセージング アプリケーション— **WebLogic Integration Messaging API** を使用する **XOCP** アプリケーションです。XOCP アプリケーションはトレーディング パートナ ロールを実装し、**B2B** エンジンと直接会話して会話を管理し、ビジネス メッセージを処理します。

管理アプリケーションおよびメッセージング アプリケーションの詳細については、『*B2B Integration 管理アプリケーションプログラミングガイド*』および『*B2B Integration メッセージング アプリケーションプログラミングガイド*』を参照してください。

B2B Integration の概要については、『*B2B Integration 入門*』を参照してください。

ロジック プラグイン

ロジック プラグインは、B2B エンジンを通過するビジネス メッセージに対して特殊な処理を実行する Java クラスです。具体的には、ロジック プラグインはビジネス メッセージが B2B エンジンを通り過ぎるまでのパスにルールおよびビジネス ロジックを挿入します。WebLogic Integration では、各ビジネスプロトコルのためのルータおよびフィルタ ロジック プラグインが提供されています。サービス プロバイダやトレーディング パートナは、カスタム ロジック プラグインを開発およびインストールしてハブ アンド スポーク コンフィグレーションに付加価値を与えることができます（『*B2B Integration 入門*』の「B2B Integration の基礎」の「メッセージ仲介モデル」を参照）。

ロジック プラグインは WebLogic Integration のリポジトリで定義および格納され、B2B エンジンで実行されます。ロジック プラグインの介在はユーザに意識されません。

2 ビジネス メッセージのルーティングとフィルタ処理

以下の節では、トレーディング パートナ間で交換されるビジネス メッセージのフローを制御するために、ルーティング、フィルタ処理、および XPath 式を使用する方法を説明します。

- ビジネス メッセージおよびメッセージ エンベロープ
- 実行時のメッセージ処理
- メッセージコンテキスト ドキュメントでの作業
- XPath 式での作業

ビジネス メッセージおよびメッセージ エンベロープ

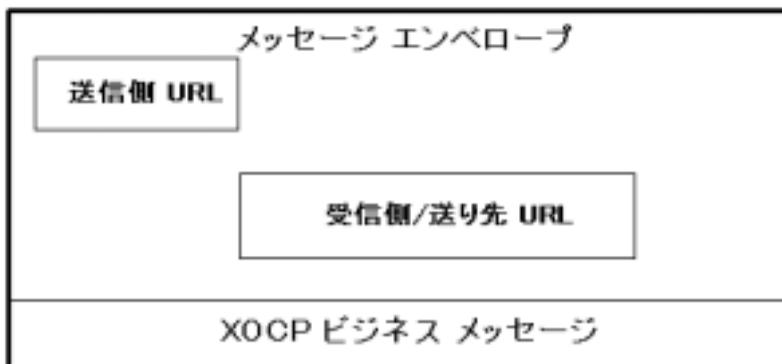
ビジネス メッセージは、会話のトレーディング パートナ間で交換される通信の基本単位です。ビジネス メッセージには、メッセージの受信側のリストがあります。1 つのビジネス メッセージは、**B2B integration API** では `com.bea.b2b.protocol.messaging.Message` インタフェースによって表されます。以下のクラスは、このインタフェースを実装して、プロトコル固有のビジネス メッセージを表します。

- `com.bea.b2b.protocol.xocp.messaging.XOCPMessage`
- `com.bea.b2b.protocol.rosettanet.messaging.RNMessage`

ビジネス メッセージを受信すると、B2B エンジンではメッセージエンベロープを作成します。メッセージエンベロープは、B2B エンジンで処理されるビジネスメッセージのコンテナになります。メッセージエンベロープは、`com.bea.b2b.protocol.messaging.MessageEnvelope` クラスのインスタンスです。

メッセージエンベロープはルーティングのために使用されるもので、手紙の封筒に相当します。メッセージエンベロープには、ビジネスメッセージに加えて、送信側の識別情報（返信先アドレス）およびビジネスメッセージの受信者（送り先アドレス）といったアドレス情報が含まれます。次に図を示します。

図 2-1 XOCP ビジネス メッセージを含んでいるメッセージエンベロープ



また、メッセージエンベロープにはビジネスメッセージに関するその他の情報も含まれます。`MessageEnvelope` クラスの詳細については、『*BEA WebLogic Integration Javadoc*』を参照してください。

実行時のメッセージ処理

WebLogic Integration B2B エンジンではロジック プラグインをルータまたはフィルタとして利用して、トレーディング パートナへのビジネスメッセージのフローを管理します。次の例は、XOCP ビジネスメッセージに対するこの処理の実装を示します。

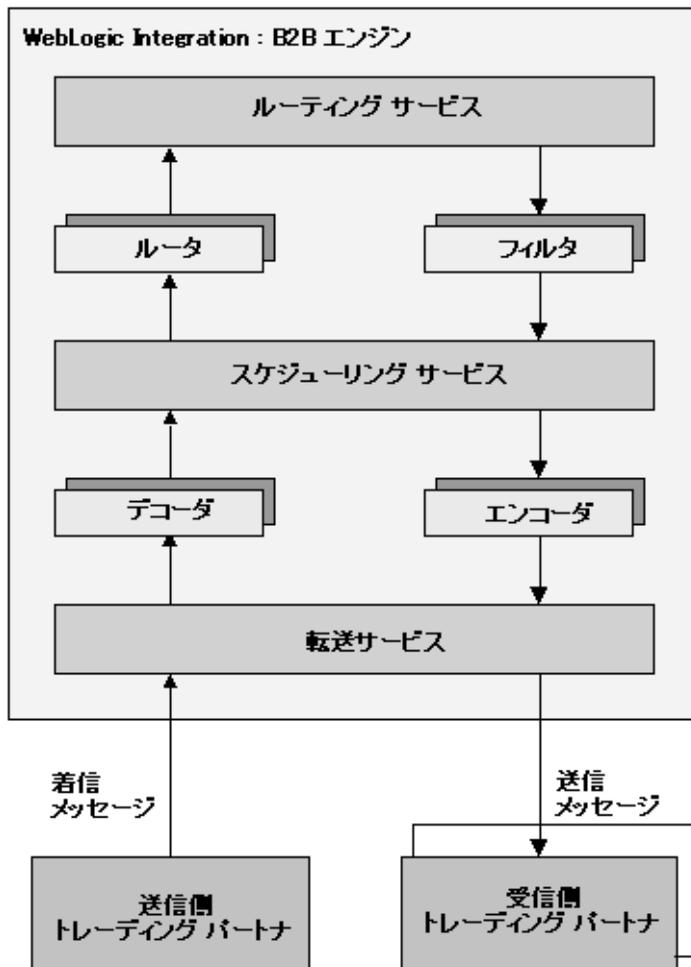
注意： XOCP ビジネス プロトコルは、WebLogic Integration の本リリースより非推奨になりました。代替機能に関する情報については、『*BEA WebLogic Integration リリース ノート*』を参照してください。

- トレーディング パートナが B2B エンジンへ XOCP ビジネス メッセージを送信すると、ロジック プラグインが XOCP ルータとして動作し、メッセージの送り先となるトレーディング パートナを決めます。ルータ ロジック プラグインはメッセージ処理の送信側にあり、メッセージの受信側を決定します。
- WebLogic Integration によってビジネス メッセージが受信側トレーディング パートナへ送信される前に、2 つめのロジック プラグインが、XOCP フィルタとして動作して、当該のトレーディング パートナがメッセージを受信すべきかどうかを決めます。2 つめのフィルタ ロジック プラグインは、メッセージ処理の受信側にあります。フィルタ ロジック プラグインを使用して、特定のトレーディング パートナが特定のビジネス メッセージを受信しないようにできます。

2 ビジネス メッセージのルーティングとフィルタ処理

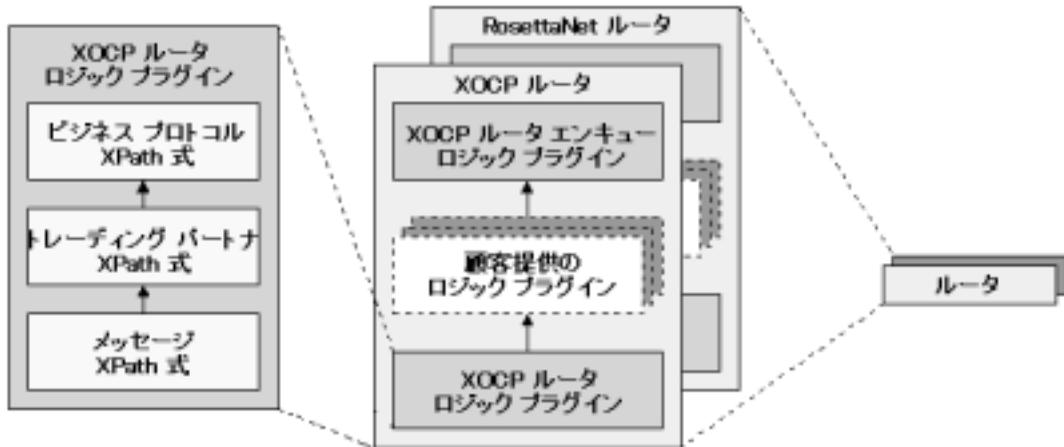
次の図は、B2B エンジンによるメッセージ処理の概要を示します。

図 2-2 メッセージ処理の概要



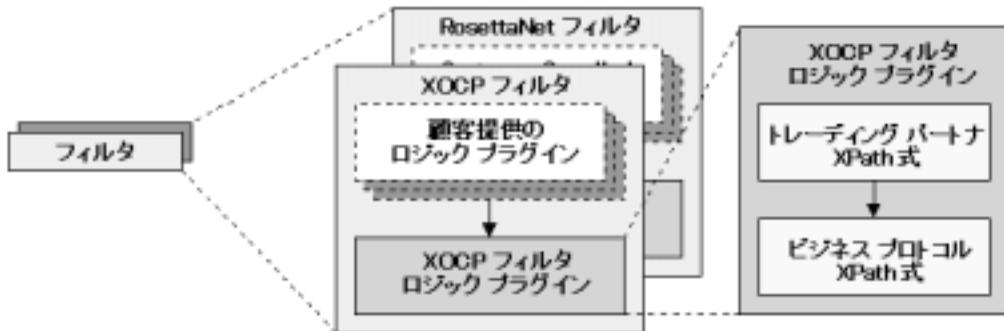
WebLogic Integration では、サポートされるビジネスプロトコルごとにルータが提供されています。次の図は、ルータの詳細を示します。

図 2-3 WebLogic IntegrationB2B ルータ



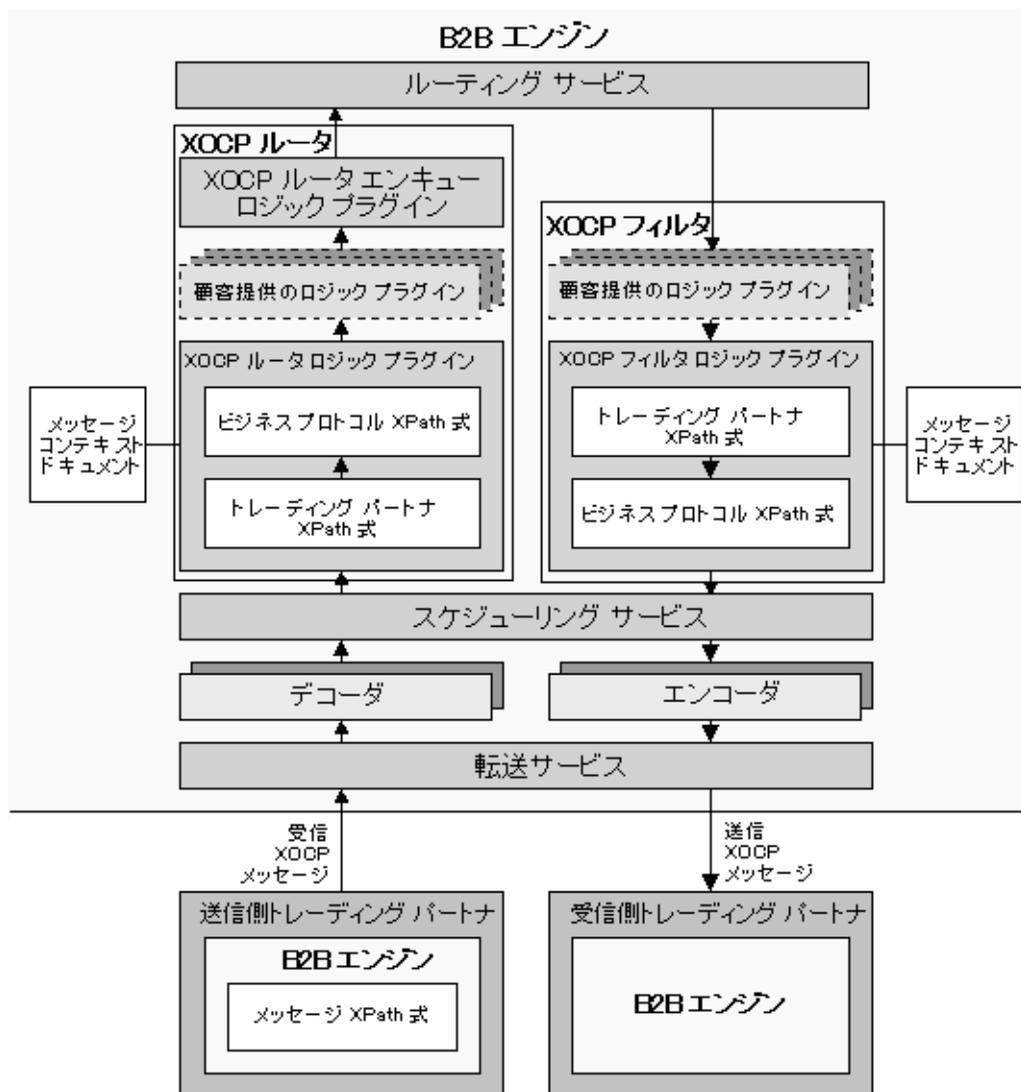
WebLogic Integration では、サポートされるビジネスプロトコルごとにフィルタが提供されています。次の図は、フィルタの詳細を示します。

図 2-4 WebLogic IntegrationB2B フィルタ



次の図は、B2B エンジンによる XOCP ビジネス メッセージ処理の詳細を示します。RosettaNet ビジネス メッセージの処理も同様です。詳細については、『*B2B Integration RosettaNet の実装*』を参照してください。

図 2-5 XOCO メッセージの処理



以下の節では、**B2B** エンジンの送信側および受信側での **XOCP** ビジネス メッセージの処理について説明します。

- 送信側
- 受信側

送信側

以下の節では、**B2B** エンジンの送信側のコンポーネントと **XOCP** ビジネス メッセージの処理について説明します。

- メッセージ XPath 式
- 転送サービス
- デコーダ
- スケジューリング サービス
- **XOCP** ルータ
- ルーティング サービス

メッセージ XPath 式

XOCP ビジネス メッセージを送信する際に、送信側トレーディング パートナのための **B2B** エンジンではビジネス メッセージの受信側を定義するメッセージ XPath 式を指定できます。このメッセージ XPath 式は、**Business Process Management (BPM)** ワークフロー、またはローカルに実行される **B2B Integration** アプリケーションで定義されます。メッセージ XPath 式の詳細については、2-20 ページの「メッセージ XPath 式の作成」を参照してください。

転送サービス

転送サービスは受信 XOCP ビジネス メッセージを読み取って以下の処理を行います。

1. メッセージをメッセージ エンベロープにラップして、B2B エンジンを通過する際に処理されやすくします。
2. XOCP、RosettaNet、または cXML などのビジネス プロトコルに基づいて、メッセージを適切なデコーダへ転送します。転送サービスがメッセージを受信する場所の URL によって、プロトコルおよび配信チャネルが識別されます。配信チャネル/ビジネス プロトコルの組み合わせごとに、ユニークな URL があります。トレーディング パートナはこの URL を使用して、特定のビジネス プロトコルを使用する特定の配信チャネルへアクセスします。

警告： 配信チャネル/ビジネス プロトコルの組み合わせに対応する URL は、B2B エンジンのみが使用できます。顧客提供のソフトウェアがこうした URL を使用しても、メッセージは正しく処理されません。

ビジネス プロトコルのコンフィグレーションの詳細については、『*B2B Integration 管理ガイド*』を参照してください。

デコーダ

デコーダは以下の処理を行います。

1. プロトコル固有のメッセージ ヘッダを処理します。
2. 送信側トレーディング パートナを識別します。
3. 送信側トレーディング パートナを会話に追加します。
4. 送信側へ返信する応答を準備します。
5. メッセージをスケジューリング サービスへ転送します。

スケジューリング サービス

スケジューリング サービスはメッセージをキューに入れて格納し、後に XOCP ルータで取得できるようにします。

XOCP ルータ

XOCP ルータはロジック プラグインをつなげたもので、XOCP ビジネス メッセージの受信側を指定します。各ロジック プラグインは、受信側トレーディング パートナのセットについてトレーディング パートナを追加または削除できます。

XOCP ルータにあるロジック プラグインは、以下の順序で並べられます。

1. XOCP ルータ ロジック プラグイン – WebLogic Integration で提供されているもの
2. 顧客提供のロジック プラグイン – ユーザが作成できるロジック プラグイン (省略可能)
3. XOCP ルータ エンキュー ロジック プラグイン – WebLogic Integration で提供されているもの

以下の節では、これらのロジック プラグインについて説明します。

XOCP ルータ ロジック プラグイン

XOCP ルータ ロジック プラグインは、以下の処理を行います。

1. メッセージコンテキスト ドキュメントを作成します。

メッセージコンテキスト ドキュメントは、XOCP ビジネス メッセージおよびリポジトリ中の関連する情報から XOCP ルータ ロジック プラグインによって作成される XML ドキュメントです。メッセージコンテキスト ドキュメントは、ビジネス プロトコル、会話、送信側トレーディング パートナ、および受信側トレーディング パートナといった、XOCP ビジネス メッセージに関するヘッダおよびコンテキストの情報を示します。XOCP ルータ ロジック プラグインは XPath 式を使用してメッセージコンテキスト ドキュメントを評価します。メッセージコンテキスト ドキュメントの詳細については、2-15 ページの「メッセージコンテキスト ドキュメントでの作業」を参照してください。

2. メッセージコンテキスト ドキュメントを XPath ルーティング式で評価します。これにより、メッセージコンテキスト ドキュメント内の各値と対応する場合があります。この評価の結果、XOCP ビジネス メッセージの受信先となるトレーディング パートナのセットが得られます。

XOCP ルータ ロジック プラグインは XPath ルータ式を次の順序で使用します。

a. メッセージ XPath 式

メッセージ XPath 式はビジネス メッセージに含まれているので、常にビジネス メッセージのルーティングに適用されます。

b. トレーディング パートナ XPath ルータ式

トレーディング パートナ XPath ルータ式は送信側トレーディング パートナ用のリポジトリで定義され、当該のトレーディング パートナによって送信されるすべての XOCP ビジネス メッセージに適用されます。送信側トレーディング パートナは複数のトレーディング パートナ XPath ルータ式を持つことができます。

トレーディング パートナ XPath ルータ式は、メッセージコンテキストドキュメントのさまざまな部分を調べて受信側トレーディング パートナの別のセットを選択します。各式によって生成されたトレーディング パートナは、以前に生成された受信側トレーディング パートナのセットと置き換えられたり、現在のセットに追加されたりできます。

c. ビジネス プロトコル XPath ルータ式

XPath ルータ式はリポジトリで定義され、特定のビジネス プロトコルを使用するすべての XOCP ビジネス メッセージに適用されます。

トレーディング パートナ XPath ルータ式と同様に、ビジネス プロトコル XPath ルータ式はメッセージコンテキストドキュメントのさまざまな部分を調べて受信側トレーディング パートナの別のセットを選択します。各式によって生成されたトレーディング パートナは、以前に生成された受信側トレーディング パートナのセットと置き換えられたり、現在のセットに追加されたりできます。

XOCP ルータ ロジック プラグインが使用するための XPath 式をリポジトリに追加することができます。XPath 式の詳細については、2-16 ページの「XPath 式での作業」を参照してください。

3. メッセージコンテキストドキュメントを破棄します。
4. 受信側トレーディング パートナのセットが空でない限り、B2B エンジン はメッセージの処理を続けます。空である場合、XOCP ルータ ロジック プラグインはメッセージを次のコンポーネントでの処理に転送しません。

顧客提供のルータ ロジック プラグイン

ロジック プラグインを作成して、XOCP ルータに追加することができます。新しいロジック プラグインを作成した場合、それをチェーンに追加する際は場所は XOCP ルータ ロジック プラグインの後で、かつ XOCP ルータ エンキュー ロジック プラグインの前です。XOCP ルータ チェーン内でのロジック プラグインの順序は、XOCP ビジネス プロトコル定義で指定されます。

顧客提供のロジック プラグインは、ルータ機能を持たなくとも XOCP ルータの一部になることができます。たとえば、顧客提供のロジック プラグインは、特定の送信側トレーディング パートナによって送信されたメッセージの数をトラッキングし、該当するメッセージについてトレーディング パートナに請求することで請求処理の機能を提供できます。顧客提供のロジック プラグインは、ルーティングおよびフィルタ処理の機能を提供していなくても、XOCP ルータまたは XOCP フィルタのみに追加できます。顧客提供のロジック プラグインの詳細については、第 3 章「ロジック プラグインの作成と追加（非推奨）」を参照してください。

顧客提供のルータ ロジック プラグインが処理された後で、受信側トレーディング パートナのセットが空でなければ、B2B エンジン はメッセージの処理を続けます。空である場合、顧客提供のルータ ロジック プラグインはメッセージを次のコンポーネントでの処理に転送しません。

XOCP ルータ エンキュー ロジック プラグイン

XOCP ルータ エンキュー ロジック プラグインは、以下の処理を行います。

1. XOCP ビジネス メッセージを受信先のキューに入れます。
2. メッセージをルーティング サービスへ転送します。

ルーティング サービス

ルーティング サービスは以下の処理を行います。

1. メッセージ受信側の最終検証を実行します。
2. 検証済みの受信側トレーディング パートナに個別のメッセージ エンベロープを作成します。
3. メッセージ エンベロープの各コピーを XOCP フィルタへ転送します。

受信側

以下の節では、B2B エンジンの受信側のコンポーネントと XOCP ビジネス メッセージの処理について説明します。

- XOCP フィルタ
- スケジューリング サービス
- エンコーダ
- 転送サービス

XOCP フィルタ

XOCP フィルタはロジック プラグインのチェーンで、XOCP ビジネス メッセージを受信側へ送信するかどうかを決定します。これらのロジック プラグインは、XOCP ルータ ロジック プラグインの後で評価され、XOCP ルータによる結果を変更またはオーバーライドすることができます。各ロジック プラグインで、メッセージの非送信を決定できます。

XOCP フィルタにあるロジック プラグインは、以下の順序で並べられます。

1. 顧客提供のロジック プラグイン – ユーザが作成できるロジック プラグイン (省略可能)
2. XOCP フィルタ ロジック プラグイン – **WebLogic Integration** で提供されているもの

以下の節では、これらのロジック プラグインについて説明します。

XOCP フィルタ ロジック プラグイン

XOCP フィルタ ロジック プラグインは、以下の処理を行います。

1. メッセージコンテキスト ドキュメントを作成します。

メッセージコンテキスト ドキュメントは、XOCP ビジネス メッセージおよびリポジトリ中の関連する情報から XOCP フィルタ ロジック プラグインによって作成される XML ドキュメントです。メッセージコンテキスト ドキュメントは、ビジネス プロトコル、会話、送信側トレーディング パートナ、および受信側トレーディング パートナといった、XOCP ビジネス メッセージ

ジに関するヘッダおよびコンテキストの情報を示します。XOCP フィルタ ロジック プラグインは XPath 式を使用してメッセージコンテキスト ドキュメントを評価します。メッセージコンテキスト ドキュメントの詳細については、2-15 ページの「メッセージコンテキスト ドキュメントでの作業」を参照してください。

2. メッセージコンテキスト ドキュメントを XPath フィルタ式で評価します。これにより、メッセージコンテキスト ドキュメント内の各値と対応する場合があります。この評価の結果、メッセージを受信側へ送信するかどうかが決まります。

XOCP フィルタ ロジック プラグインは XPath フィルタ式を次の順序で使用します。

- a. トレーディング パートナ XPath フィルタ式

トレーディング パートナ XPath フィルタ式は受信側トレーディング パートナ用のリポジトリで定義され、当該のトレーディング パートナ宛のすべての XOCP ビジネス メッセージに適用されます。受信側トレーディング パートナは複数のトレーディング パートナ XPath フィルタ式を持つことができます。

トレーディング パートナ XPath フィルタ式は、メッセージコンテキスト ドキュメントのさまざまな部分を調べてメッセージの受信または拒否を示すブール値の結果を返します。この処理は、**false** と評価される式が出るか、すべての式が処理されるまで継続されます。

- b. ビジネス プロトコル XPath フィルタ式

XPath フィルタ式はリポジトリで定義され、すべての XOCP ビジネス メッセージに適用されます。

トレーディング パートナ XPath フィルタ式と同様に、ビジネス プロトコル XPath フィルタ式はメッセージコンテキスト ドキュメントのさまざまな部分を調べてメッセージの受信または拒否を示すブール値の結果を返します。この処理は、**false** と評価される式が出るか、すべての式が処理されるまで継続されます。

XOCP フィルタ ロジック プラグインが使用するための XPath 式をリポジトリに追加することができます。XPath 式の詳細については、2-16 ページの「XPath 式での作業」を参照してください。

3. メッセージコンテキスト ドキュメントを破棄します。

4. **XOCP** フィルタ ロジック プラグインが受信側への **XOCP** ビジネス メッセージの配信をキャンセルした場合、その **XOCP** フィルタ ロジック プラグインはメッセージを **B2B** エンジン内の次のコンポーネントへ転送しません。メッセージが転送された場合は、**B2B** エンジンでメッセージの処理が継続されま

顧客提供のフィルタ ロジック プラグイン

ロジック プラグインを作成して、**XOCP** フィルタに追加することができます。新しいロジック プラグインを作成した場合、それをチェーンに追加する際は **XOCP** フィルタ ロジック プラグインの前です。**XOCP** フィルタ チェーン内のロジック プラグインの順序は、**XOCP** ビジネス プロトコル定義で指定されます。

顧客提供のロジック プラグインは、フィルタ機能を持たなくとも **XOCP** フィルタの一部になることができます。たとえば、顧客提供のロジック プラグインは、特定の受信側トレーディング パートナへ送信されたメッセージの型をトラッキングすることでサンプリングの機能を提供できます。顧客提供のロジック プラグインは、ルーティングおよびフィルタ処理の機能を提供していても、**XOCP** ルータまたは **XOCP** フィルタのみに追加できます。ロジック プラグインの詳細については、第3章「ロジック プラグインの作成と追加（非推奨）」を参照してください。

顧客提供のロジック プラグインが受信側への **XOCP** ビジネス メッセージの配信をキャンセルした場合、顧客提供のロジック プラグインはメッセージを **B2B** エンジン内の次のコンポーネントへ転送しません。メッセージが転送された場合は、**B2B** エンジンでメッセージの処理が継続されます。

スケジューリング サービス

スケジューリング サービスは以下の処理を行います。

1. サービス品質の問題および会話の管理に関連する内部処理を実行します。サービス品質の詳細については、『*B2B Integration 管理アプリケーション プログラミング ガイド*』を参照してください。
2. メッセージをエンコーダへ転送します。

エンコーダ

エンコーダは、ビジネス プロトコルをサポートする必要に応じてメッセージを変換し、そのメッセージを転送サービスへ転送します。

転送サービス

転送サービスは、メッセージを受信側へ転送します。

メッセージコンテキスト ドキュメントでの作業

メッセージコンテキスト ドキュメントの作成および使用の詳細については、2-9 ページの「XOCP ルータ ロジック プラグイン」および 2-12 ページの「XOCP フィルタ ロジック プラグイン」を参照してください。

次のリストは、メッセージコンテキスト ドキュメント用の文書型定義 (DTD) を示します。

コード リスト 2-1 メッセージコンテキスト ドキュメント用の文書型定義

```
<!--Copyright (c) 2001 BEA Systems, Inc. -->
<!--All rights reserved -->

<!-- この DTD は XPATH ルータ および フィルタ用のメッセージコンテキスト ドキュメントを示す -->

<!ELEMENT wlc (business-protocol, conversation, sender, trading-partner+) >
<!ATTLIST wlc context ( message-router | trading-partner-router | hub-router |
trading-partner-filter | hub-filter ) #REQUIRED >

<!ELEMENT business-protocol EMPTY >
<!ATTLIST business-protocol name CDATA #REQUIRED >
<!ATTLIST business-protocol version CDATA #REQUIRED >

<!ELEMENT conversation EMPTY >
<!ATTLIST conversation name CDATA #REQUIRED >
```

2 ビジネス メッセージのルーティングとフィルタ処理

```
<!ATTLIST conversation version CDATA #REQUIRED >
<!ATTLIST conversation sender-role CDATA #REQUIRED >
<!ATTLIST conversation receiver-role CDATA #REQUIRED >

<!-- 送信側は、会話中のロールからメッセージを送信したトレーディング パートナ -->
<!ELEMENT sender ( trading-partner ) >

<!-- トレーディング パートナはメッセージを送受信する企業などエンティティを表す -->
<!ELEMENT trading-partner ( address, extended-property-set* ) >
<!ATTLIST trading-partner email CDATA #IMPLIED >
<!ATTLIST trading-partner fax CDATA #IMPLIED >
<!ATTLIST trading-partner name ID #REQUIRED >
<!ATTLIST trading-partner phone CDATA #IMPLIED >

<!ELEMENT address ANY >

<!ELEMENT extended-property-set ANY >
<!ATTLIST extended-property-set name CDATA #REQUIRED >
```

XPath 式での作業

この節では、XPath 式とその作成方法について説明します。

- XPath 式について
- メッセージ XPath 式の作成
- トレーディング パートナ XPath 式の作成
- ビジネス プロトコル XPath 式の作成

XPath 式について

XPath は World Wide Web Consortium で定義されている XML パス言語です。XOCP ルータ ロジック プラグインおよび XOCP フィルタ ロジック プラグインは、XPath 式を使用してメッセージコンテキスト ドキュメントを評価します。XOCP ルータ ロジック プラグインおよび XOCP フィルタ ロジック プラグインが使用するための XPath 式をリポジトリに追加することができます。

XOCP ルータ ロジック プラグイン および XOCP フィルタ ロジック プラグインにある XPath 式は、以下の機能を実行します。

- XPath ルータ式は、XPath 構文を使用してメッセージコンテキスト ドキュメントからトレーディング パートナのセットを選択します。選択されたトレーディング パートナは、XOCP ビジネス メッセージの受信側です。各 XPath ルータ式は、トレーディング パートナのセットを評価する必要があります。

XOCP ルータ ロジック プラグインでは、XPath 式はメッセージ配信のビジネス基準を指定します。たとえば、パイヤは XPath ルータ式を使用して入札要求を特定の市外局番のすべてのセラーに送信したり、大口の注文を扱えるセラーに送信したりできます。

- XPath フィルタ式は XPath の構文を使用してメッセージの受信または拒否を示すブール値の結果を返します。各 XPath フィルタ式による評価の結果はブール値の true または false でなければなりません。

XOCP フィルタ ロジック プラグインでは、XPath 式は B2B エンジンが特定のビジネス メッセージを特定のトレーディング パートナへ送信するかどうかを決定します。XOCP フィルタ ロジック プラグイン内の XPath フィルタ式は、受信側トレーディング パートナにとって不要なビジネス メッセージをフィルタするゲートキーパとして動作します。

次の表は、各種の XPath 式の概要を示します。

2 ビジネス メッセージのルーティングとフィルタ処理

表 2-1 各種の XPath 式の概要

XPath 式の種類	XOCP ルータ ロジック プラグイン	XOCP フィルタ ロジック プラグイン
メッセージ	評価順: 最初 XPath 式の数: 1 定義の場所: BPM ワークフローまたは B2B Java アプリケーション 用途: 受信側の定義 適用の対象: 送信側 BPM ワークフローまたは B2B アプリケーションからの XOCP ビジネス メッセージ	適用不可
トレーディング パートナ	評価順: 2 番目 XPath 式の数: 1 つまたは複数 定義の場所: リポジトリ (WebLogic Integration B2B Console または Bulk Loader を使用) 用途: 受信側の追加および削除 適用の対象: 送信側トレーディング パートナからのすべての XOCP ビジネス メッセージ	評価順: 4 番目 XPath 式の数: 1 つまたは複数 定義の場所: リポジトリ (WebLogic Integration B2B Console または Bulk Loader を使用) 用途: メッセージを受信側へ送信するかどうかの決定 適用の対象: 受信側トレーディング パートナへのすべての XOCP ビジネス メッセージ
ビジネス プロトコル	評価順: 3 番目 XPath 式の数: 1 つまたは複数 定義の場所: リポジトリ (WebLogic Integration B2B Console または Bulk Loader を使用) 用途: 受信側の追加および削除 適用の対象: すべての送信側トレーディング パートナからのすべての XOCP ビジネス メッセージ	評価順: 5 番目 XPath 式の数: 1 つまたは複数 定義の場所: リポジトリ (WebLogic Integration B2B Console または Bulk Loader を使用) 用途: メッセージを受信側へ送信するかどうかの決定 適用の対象: すべての受信側トレーディング パートナへのすべての XOCP ビジネス メッセージ

XOCP ルータ ロジック プラグインでは、XPath ルータ式によってメッセージコンテキスト ドキュメントの各部分が調べられ、別の受信側トレーディング パートナのセットが選択されます。各式によって生成されたトレーディング パートナは、以前に生成された受信側トレーディング パートナのセットと置き換えられたり、現在のセットに追加されたりできます。

次の表は、XPath ルータ式がどのように使用されるかをステップを追って示します。

表 2-2 XPath ルータ式に関する例

XPath 式	結果として選択される受信側トレーディング パートナのセット
1. メッセージ XPath 式でトレーディング パートナ A および B が選択される。	A、B
2. 最初のトレーディング パートナ XPath ルータ式でトレーディング パートナ C が追加される。	A、B、C
3. 2 番目のトレーディング パートナ XPath ルータ式で、以前に選択されているすべてのトレーディング パートナがトレーディング パートナ D によって置き換えられる。	D
4. 最初のビジネス プロトコル ルータ式で、トレーディング パートナ B および F が追加される。	D、B、F
5. 2 番目のビジネス プロトコル ルータ式で、トレーディング パートナ F が削除される。	D、B

XOCP フィルタ ロジック プラグインでは、XPath ルータ式によってメッセージコンテキスト ドキュメントの各部分が調べられ、受信側トレーディング パートナへメッセージを転送するかどうか決定されます。XPath フィルタ式は、さまざまな選択基準に基づいて true または false を返します。false が返されると、そのメッセージは評価が中止されて、受信側には送信されません。

XPath 式は、以下の情報を参照できます。

- トレーディング パートナの属性

- 標準的な属性。たとえば、トレーディング パートナ名や郵便番号
- 拡張された属性。たとえば、WebLogic Integration B2B Console で定義されるカスタム属性
- メッセージ情報。たとえばビジネス ドキュメントの型、発注書番号、あるいはインボイス総計

Xpath 式の詳細については、『*B2B Integration 管理ガイド*』の「高度なコンフィグレーション タスク」を参照。

メッセージ XPath 式の作成

XOCP ビジネス メッセージを送信する際に、送信側トレーディング パートナは XPath 式を指定して、ビジネス メッセージの受信側を定義することができます。メッセージ XPath ルータ式は、Business Process Management ワークフローまたは WebLogic Integration B2B Java アプリケーションで定義されています。この XPath 式で、XOCP ルータ ロジック プラグインによって生成される メッセージ コンテキスト XML ドキュメントから <trading-partner> ノードのサブセットが選択されます。

送信側トレーディング パートナはこの XPath 式を定義して、メッセージとともに送信します。XPath 式は、B2B Integration では次のようにして定義されます。

- BPM ワークフローを使用してビジネス メッセージが交換される場合、XPath 式はワークフロー テンプレートで定義されて、トレーディング パートナが別のトレーディング パートナへメッセージを送信するときに適用されます。XPath 式を定義するには、WebLogic Integration Studio の [ビジネス メッセージの送信] ダイアログ ボックスを使用します。詳細については、『*B2B Integration ワークフローの作成*』を参照してください。
- Java アプリケーションを使用してビジネス メッセージを交換する場合、XPath 式は Java アプリケーションで定義されます。
`com.bea.b2b.protocol.messaging.Message` インスタンスの `setExpression` メソッドを呼び出します。このとき、パラメータとして XPath 式を渡します。詳細については、『*B2B Integration ワークフローの作成*』を参照してください。

注意： 多くの場合、トレーディング パートナがビジネス メッセージを送信する相手は、単一で既知の、他のトレーディング パートナです。た

例えば、トレーディング パートナからの要求に応答する場合などで、この場合、送信側トレーディング パートナは、受信側トレーディング パートナの名前を XPath 式の代わりに指定することで、XOCP ルータ ロジック プラグインでの XPath 式の評価を回避することができます。トレーディング パートナ名を指定するには、`com.bea.b2b.protocol.messaging.Message` インスタンスの `setExpression` メソッドではなく `setRecipient` メソッドを呼び出してください。

トレーディング パートナ XPath 式の作成

トレーディング パートナ XPath 式は、トレーディング パートナごとに定義される XPath 式です。ルーティングの場合、トレーディング パートナ XPath 式は XOCP ルータ ロジック プラグインに使用されて、送信側トレーディング パートナについて定義されます。フィルタ処理の場合、トレーディング パートナ XPath 式は XOCP フィルタ ロジック プラグインに使用されて、受信側トレーディング パートナについて定義されます。

トレーディング パートナ XPath 式はリポジトリに定義されます。XOCP ルータ ロジック プラグインおよび XOCP フィルタ ロジック プラグインで使用されるトレーディング パートナ XPath 式を作成するために、以下のツールを使用できます。

- Bulk Loader については、『*B2B Integration 管理ガイド*』の「Bulk Loader の操作」を参照してください。リポジトリ データ ファイルでの XPath 式のフォーマットは次のようになります。

```
<xpath-expression expression="//TradingPartner1"  
location="ROUTER" type="APPEND"/>
```

XPath の構文および使用法の詳細については、World Wide Web Consortium によって公開されている「XML Path Language Specification」を参照してください。この文書は次の URL にあります。

<http://www.w3.org/TR/xpath.html>

- WebLogic Integration B2B Console については、『*B2B Integration Administration Console オンライン ヘルプ*』の「ロジック プラグインの使用」を参照してください。

次の表は、B2B Console を使用して XPath 式を定義するときに設定するプロパティを示します。

表 2-3 B2B Console での XPath 式のプロパティ

コンポーネント	説明
XPath 式	既に説明した形式の、XPath ルータ式または XPath フィルタ式。
Type	XPath 式を評価した結果について、以前の XPath 式を評価した結果に追加するか、または置換するかを指定するフラグ。

たとえば、あるトレーディング パートナが、カリフォルニアにいるトレーディング パートナへの要求をルーティングする必要があるとします。これを実現するために、送信側トレーディング パートナは B2B Console の [トレーディング パートナ] タブにある詳細オプションのウィンドウを使用して、XOCP ルータ ロジック プラグイン用に次の XPath 式を作成できます。

```
/wlc/trading-partner[extended-property-set/state='California']
```

ビジネス プロトコル XPath 式の作成

ビジネス プロトコル XPath 式は、特定のビジネス プロトコルごとに WebLogic Integration リポジトリで定義されている XPath 式です。ビジネス プロトコル XPath ルータ式は、対象となるプロトコルを使用するすべての受信ビジネス メッセージに適用されます。ビジネス プロトコル XPath フィルタ式は、すべての送信 XOCP ビジネス メッセージに適用されます。

ビジネス プロトコル XPath 式はリポジトリに定義されます。XOCP ルータ ロジック プラグインおよび XOCP フィルタ ロジック プラグインで使用される XPath 式を作成するために、以下のツールを使用できます。

- Bulk Loader については、『*B2B Integration 管理ガイド*』の「Bulk Loader の操作」を参照してください。リポジトリ データ ファイルでの XPath 式のフォーマットは次のようになります。

```
<xpath-expression expression="//TradingPartner1"
location="ROUTER" type="APPEND"/>
```

XPath の構文および使用法の詳細については、World Wide Web Consortium によって公開されている「XML Path Language Specification」を参照してください。この文書は次の URL にあります。

<http://www.w3.org/TR/xpath.html>

- WebLogic Integration B2B Console については、『*B2B Integration Administration Console* オンライン ヘルプ』の「ロジックプラグインの使用」を参照してください。

表 2-3 に、B2B Console を使用して XPath 式を定義するときに設定するプロパティを示しています。

たとえば、ある管理者が、出荷側であるトレーディング パートナが出荷要求のみを受信できるようにメッセージをフィルタしつつ、他のすべてのトレーディング パートナについてはすべてのメッセージを受信できるようにする必要があります。これを実現するために、管理者は B2B Console の [ビジネスプロトコル定義] タブを使用して XOCF フィルタ ロジックプラグインに次の XPath 式を作成できます。

```
(/wlc/trading-partner/extended-property-set/business='shipper') OR  
(/wlc/trading-partner/extended-property-set/business!='shipper')
```

3 ロジック プラグインの作成と追加 (非推奨)

以下の節では、WebLogic Integration でロジック プラグインを開発する方法について説明します。

- ロジック プラグインについて
- ロジック プラグイン API
- ロジック プラグインについてのルールおよびガイドライン
- ロジック プラグインの開発と管理

ロジック プラグインについて

以下の節では、ロジック プラグイン、および関連する概念について説明します。

- ロジック プラグインの概要
- ロジック プラグインが処理するタスク
- チェーン
- システム ロジック プラグインとカスタム ロジック プラグイン

ロジック プラグインの概要

注意： カスタムロジック プラグインは本リリースの WebLogic Integration では非推奨となりました。代替りとなる機能については、『*BEA WebLogic Integration リリース ノート .*』を参照してください。

3 ロジック プラグインの作成と追加（非推奨）

ロジック プラグインは、B2B エンジンを通過するビジネス メッセージに対して特殊な処理を実行する Java クラスです。ロジック プラグインは、WebLogic Integration プロバイダまたはトレーディング パートナによって、カスタム サービスとして開発されます。

ロジック プラグインは、WebLogic Integration B2B システムを通過するビジネス メッセージのパスに沿った重要な場所にルールおよびビジネス ロジックを挿入します。ロジック プラグインは、WebLogic Integration でビジネス プロトコルが作成されるときに作成される Java クラスのインスタンスです。ロジック プラグインは、トレーディング パートナの配信チャネルが起動されたときにアクティブ化され、メッセージが B2B エンジンを通過するときに呼び出されます。

ロジック プラグインは、ビジネス プロトコルに個別に対応します。ロジック プラグインでは、特定のビジネス プロトコルを使用して交換されるメッセージのみが処理されます。たとえば、あるプラグインが XOCF プロトコルと関連付けられている場合、そのプラグインは XOCF ビジネス メッセージのみを処理します。

ロジック プラグインが処理するタスク

WebLogic Integration では、サポートされるビジネス プロトコルに対してルータ ロジック プラグインおよびフィルタ ロジック プラグインが提供されています。ルーティング機能やフィルタ機能の他に、カスタム ロジック プラグインがさまざまなサービスを実行できます。たとえば、請求処理が目的であれば、トレーディング パートナから送信されるメッセージの数をカスタム ロジック プラグインでトラッキングできます。

次の表は、ロジック プラグインによって実行されるタスクの種類を示します。

表 3-1 ロジック プラグインによって実行されるタスク

タスク	目的	例
ルート変更	ビジネス メッセージの受信側のリストを変更する。受信側による会話およびコラボレーション アグリーメントの検証を条件とする。この機能は、WebLogic Integration B2B システム プラグインおよびカスタム プラグインによって提供される。	<ul style="list-style-type: none">「コンピュータ チップについて 100 万ドルを超える額の注文を受けたら、NewChipCo を受信側に含めること」「2002 年 1 月 1 日以降、OldChipCo. へは注文を送信しないこと」

表 3-1 ロジック プラグインによって実行されるタスク

タスク	目的	例
検査	<p>ビジネス メッセージの内容を調べ、その結果に基づいて特定のアクションを起こす。この機能はカスタム プラグインによって提供される。</p> <p>注意： 検査されるビジネス メッセージの大多数は、暗号化された内容を含みません。</p>	<ul style="list-style-type: none"> ■ 「請求処理の目的でメッセージの送信側をすべて記録する」 ■ 「特定の型のメッセージについて、会話のバージョン 1 とバージョン 2 の比率は？」

チェーン

ルータとフィルタには、ビジネス メッセージがルータとフィルタを通過するときに実行される任意の数のロジック プラグインがあります。同一のビジネス プロトコルを共有する複数のロジック プラグインは、ロジック プラグイン チェーンとして順に並べられます。

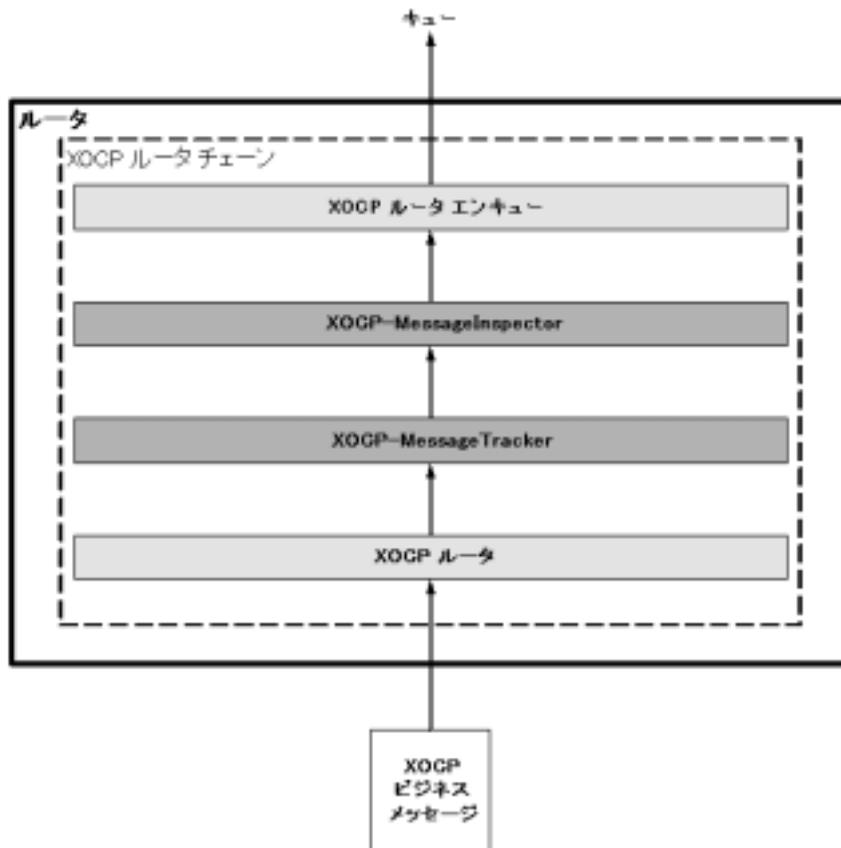
チェーンの中で、ロジック プラグインは実行時に順番に処理されます。1 つのロジック プラグインの実行が完了すると、チェーン内の次のプラグインがアクティブ化されます。連続するロジック プラグインのそれぞれが、**B2B** エンジンでビジネス メッセージが処理される間に共有メッセージ情報に加えられた任意の変更を参照できます。

注意： チェーン内のロジック プラグインの位置は、**WebLogic Integration B2B Console** を使用してリポジトリでコンフィグレーションされます。詳細については、『*B2B Integration Administration Console* オンラインヘルプ』を参照してください。

3 ロジック プラグインの作成と追加 (非推奨)

次の図は、ルータにある XOCP ロジックプラグインのチェーンの例を示します。

図 3-1 XOCP ルータ チェーンの例



ルーティング機能やフィルタ機能を提供しないカスタム ロジック プラグインもルータまたはフィルタ ロジック プラグイン チェーンの一部でなければならない点に注意してください。この例では、チェーンに 4 つのロジック プラグインがあり、次の表に示されている順番で処理されます。

表 3-2 サンプルの XOCP ルータ チェーンにあるロジック プラグイン

ロジック プラグイン	説明
XOCP ルータ	システム ロジック プラグイン。WebLogic Integration によってこのロジック プラグインが提供される。リポジトリでコンフィグレーションされている XPath ルータ式に基づいて XOCP ビジネス メッセージの受信側リストを変更できる。このロジック プラグインは、XOCP ルータ チェーンの先頭になければならない。
XOCP-MessageTracker	カスタム ロジック プラグイン。WebLogic Integration のオーナまたはトレーディング パートナは、こうしたカスタム ロジック プラグインを用意して、各トレーディング パートナから請求処理の目的で送信されるビジネス メッセージの数をトラッキングできる。
XOCP-MessageInspector	カスタム ロジック プラグイン。WebLogic Integration のオーナまたはトレーディング パートナは、こうしたカスタム ロジック プラグインを用意して、B2B エンジンを通じて交換されている各種のビジネスドキュメント（たとえば発注書、インボイスなど）の統計を検査および保持できる。
XOCP ルータ エンキュー	システム ロジック プラグイン。WebLogic Integration によってこのロジック プラグインが提供される。内部の WebLogic Integration B2B ルータ メッセージ キューに XOCP ビジネス メッセージを追加する。このロジック プラグインは、XOCP ルータ チェーンの末尾になければならない。

この例では、XOCP ビジネス メッセージのみがこの XOCP ルータ チェーンにあるロジック プラグインをトリガします。非 XOCP ビジネス メッセージ（たとえば RosettaNet や cXML メッセージ）は、それらを扱うビジネス プロトコルと関連付けられているルータ チェーンで別個に処理されます。

システム ロジック プラグインとカスタム ロジック プラグイン

ビジネス メッセージを処理するための標準的なサービスを提供するために、WebLogic Integration B2B では以下のロジック プラグインが用意されています。

表 3-3 システム ロジック プラグイン

ロジック プラグイン	説明
XOCP ルータ (XOCP router)	リポジトリでコンフィグレーションされている XPATH ルータ式に基づいて XOCP ビジネス メッセージの受信側のリストを変更する。このシステム ロジック プラグインは、ルータ ロジック プラグイン チェーンの前頭である必要がある。先頭であれば、受信側のリストが既知になるので、カスタム ロジック プラグインがビジネス メッセージを処理できる。
XOCP ルータ エンキュー	WebLogic Integration B2B ルータ メッセージキューに XOCP ビジネス メッセージを追加する。このシステム ロジック プラグインは、XOCP ルータ ロジック プラグイン チェーンの末尾になければならない。
XOCP フィルタ (XOCP filter)	リポジトリでコンフィグレーションされている XPATH フィルタ式に基づいて、特定のトレーディング パートナへ XOCP ビジネス メッセージを送信するかどうかを決定する。このシステム ロジック プラグインは、XOCP フィルタ ロジック プラグイン チェーンの末尾になければならない。
RosettaNet ルータ エンキュー	WebLogic Integration B2B ルータ メッセージキューに RosettaNet ビジネス メッセージを追加する。このシステム ロジック プラグインは、RosettaNet ルータ ロジック プラグイン チェーンの末尾になければならない。
RosettaNet フィルタ (RosettaNet filter)	RosettaNet ビジネス メッセージを特定のトレーディング パートナへ送信するかどうかを決定する。このシステム ロジック プラグインは、RosettaNet フィルタ ロジック プラグイン チェーンの末尾になければならない。

システム ロジック プラグインを使用する他に、WebLogic Integration 上に構築されたトレーディング パートナは独自のロジック プラグインを開発して専用のサービスを用意することができます。各ロジック プラグインは、ロジック プラグイン API を実装する Java クラスです。詳細については、3-12 ページの「ロジック プラグインのプログラミング ステップ」を参照してください。

ロジック プラグイン API

WebLogic Integration で提供されるロジック プラグイン API を使用して、WebLogic Integration B2B アプリケーションでは以下のことが可能になります。

- XOCP マルチキャストを使用するときに、対象のトレーディング パートナをメッセージ受信側リストで追加または削除します。
- ビジネス メッセージの各部を取得、検査、および処理します。ビジネス メッセージの内容がプログラムによって変更されたり誤って処理されたりすることを回避するために、ロジック プラグイン API にはビジネス メッセージを検査するためのメソッドは用意されていますが、ビジネス メッセージの内容を変更するためのメソッドはありません。

次の表は、ロジック プラグイン API の各コンポーネントを示します。詳細については、『*BEA WebLogic Integration Javadoc*』を参照してください。

表 3-4 ロジック プラグイン API

クラス/インタフェース	説明
<code>com.bea.b2b.protocol.PlugIn</code>	汎用のロジック プラグインを表すタグ付けインタフェース。つまり、実行するためにルータまたはフィルタに挿入できるコード。
<code>com.bea.b2b.protocol.PlugInException</code>	ロジック プラグインの実行中にエラーが発生すると送出される例外クラス。

3 ロジック プラグインの作成と追加 (非推奨)

表 3-4 ロジック プラグイン API (続き)

クラス/インタフェース	説明
<code>com.bea.b2b.protocol.messaging.MessageEnvelope</code>	ビジネス メッセージのコンテナ (エンベロープ) を表す。 <code>MessageEnvelope</code> には、実際のビジネス メッセージに加えて、関連するルーティングおよび処理の高度な情報が含まれている。たとえば、送信側 URL、および 1 名の受信側のための URL (受信側の各自に 1 つずつメッセージ エンベロープがある) などである。ネイティブなメッセージへのアクセスが必要な場合に備えて、Java の <code>InputStream</code> も利用可能である (ただし、メッセージ内容の変更は不可なので、 <code>OutputStream</code> は一切提供されない)。
<code>com.bea.b2b.protocol.messaging.メッセージ</code>	<code>Message</code> インタフェースには、 WebLogic Integration B2B エンジンでビジネス メッセージを処理するために必要なすべての情報が含まれている。 <code>Message</code> インタフェースによって、トレーディング パートナ間でメッセージを適切にルーティングするために使用される情報が提供される。また、 <code>Message</code> インタフェースには、ビジネス メッセージに使用されている特定のビジネス プロトコルに固有の情報も含まれている。使用されるプロトコルに応じて、通常 <code>Message</code> クラスはメッセージに関するプロトコル固有の追加情報を提供するためのサブクラスを含んでいる。
<code>com.bea.b2b.protocol.messaging.PayloadPart</code>	メッセージ ペイロードのコンポーネントを表す。この情報を実装する特定のクラスは、ビジネス メッセージの各部分ごとに提供されている。たとえば XML 部分と非 XML 部分などである。つまり、ビジネス プロトコル固有の情報へのアクセスを補助するために提供されている。
<code>com.bea.b2b.protocol.conversation.ConversationType</code>	特定の会話定義の中で 1 つのロールを表す。 <code>ConversationType</code> クラスには、会話名、会話バージョン、およびトレーディング パートナロールなどの情報が含まれる。

表 3-4 ロジック プラグイン API (続き)

クラス/インタフェース	説明
<code>com.bea.b2b.tpa.CAInstance</code>	コラボレーション アグリーメントのインスタンスを表す。利用可能なメソッドによって、コラボレーション アグリーメントに関するさまざまな情報を取得できる。この API でコラボレーション アグリーメントを変更することは許可されていないので、取得および検証のためのメソッドのみが提供されている。
<code>com.bea.b2b.tpa.PartyInstance</code>	コラボレーション アグリーメントのパーティを表す。利用可能なメソッドによって、コラボレーション アグリーメントのパーティに関する情報を検証または取得できる。たとえば、コラボレーション アグリーメントのパーティによって使用されている配信チャネルなどである。
<code>com.bea.b2b.tpa.TradingPartnerInstance</code>	実行時のトレーディング パートナのインスタンスを表す。このクラスは <code>PartyInstance</code> とともに使用されるか、ルータまたはフィルタとともにスタンドアロン モードで使用される。

ロジック プラグインについてのルールおよびガイドライン

ロジック プラグインは以下のルールおよびガイドラインに従う必要があります。

- ロジック プラグインはスレッドセーフかつステートレスである必要があります。実行時には、ロジック プラグインのインスタンスが複数のスレッドによってキャッシュおよび共有されます。インスタンス変数の使用は推奨されません。
- 共有されているリソースへのアクセスが必要な場合は、`synchronized` という `Java` キーワードを使用して共有リソースへのアクセスを制限します。共有アクセスが必要なリソースとしては、クラス内部のインスタンス変数、共有オブジェクト、あるいは外部システム リソース（ファイルなど）がありま

3 ロジック プラグインの作成と追加 (非推奨)

す。synchronized キーワードを使用するとアプリケーション全体のパフォーマンスが低下するので、必要な場合にのみ使用してください。

- ロジック プラグインはメッセージ エンベロープおよび受信側リストを変更できますが、メッセージの内容は変更 できません。ビジネス メッセージを変更すると、デジタル署名が無効になります (署名がある場合)。ロジック プラグインの API は、メッセージ エンベロープのみを変更するミューテータ メソッドを提供します。
- ロジック プラグインは自分自身だけで完結している必要があります。他のロジック プラグインと相互依存せず、変数を交換できず、変数を返せません。メッセージ エンベロープが唯一の入力で、かつ唯一の出力です。ロジック プラグインがメッセージ エンベロープに変更を加えた場合は、変更された状態のメッセージ エンベロープが出力されます。
- メインのロジック プラグイン クラスは `com.bea.b2b.protocol.PlugIn` インタフェースを実装している必要があります。
- セキュアなメッセージングを確保するために、通常ロジック プラグインは暗号化されたビジネス メッセージを検査できません。検査されるビジネス メッセージの大半は、暗号化された内容を含まないものです。ビジネス メッセージの暗号化された内容を調べるには、ロジック プラグインでメッセージの暗号化を解釈し、内容を調べた後で、再び暗号化できる必要があります。公開鍵のインフラストラクチャはユーザが独自に用意する必要があります。
- プラグインの提供者は、**WebLogic Integration** にインストールされるすべてのカスタム ロジック プラグインについて、適切にデバッグされ、セキュリティを確保するように設計されていることを確認する義務があります。
- ロジック プラグインは、リポジトリ内の少なくとも 1 つのビジネス プロトコルと常に関連付けられている必要があります。ロジック プラグインは、関連付けられているプロトコルを使用するビジネス メッセージが **B2B** エンジンを通るときにのみトリガされます。たとえば、**RosettaNet** ビジネス メッセージは **XOCP** ロジック プラグインをトリガしませんし、**XOCP** ビジネスメッセージは **RosettaNet** ロジック プラグインをトリガしません。
- 1 つのロジック プラグインは、リポジトリ内の複数のプロトコルと関連付けられることができます。たとえば、`SentMessages` という名前の同じロジック プラグイン クラスが、**XOCP** プロトコルおよび **RosettaNet** プロトコルの両方と関連付けられることも可能です。**WebLogic Integration B2B Console** を使用して各ビジネス プロトコルについて別個のロジック プラグインを定義できますが (たとえば、`XOCP-SentMessages`、`RN-SentMessages`、および

cXML-SentMessages)、各プラグインは同じ `SentMessage` クラスを指し示しています。あるいは、同じロジックプラグインを2つの別のプロトコルチェーンで使用できます。こうしたチェーンは初期化パラメータを共有しますが、別個のインスタンスです。

- 効率的なロジックプラグインは、ビジネスメッセージが処理に適しているかどうかを素早く判別し、適していない場合は直ちに終了します。
- ロジックプラグインは、他のモジュールを呼び出すことができます。こうしたモジュールには、ユーティリティライブラリにある共有メソッド（たとえば、データベースにアクセスするモジュール）も含まれています。
- ロジックプラグインは、配信チャンネルがアクティブ化されるときに1回で初期化されます。
 - 配信チャンネルがシャットダウンされた場合（つまり、関連する `com.bea.b2b.management.hub.runtime.DeliveryChannelMBean` クラスの `shutdown` メソッドが呼び出された場合）、その配信チャンネルと関連付けられている、プロトコル固有のすべてのプラグインもシャットダウンされます。ロジックプラグインをアクティブにするには、配信チャンネルを再起動する必要があります。
 - B2B エンジンがシャットダウンされた場合（つまり、関連する `com.bea.b2b.management.runtime.WLCMBean` クラスで `shutdown` メソッドが呼び出された場合）、その B2B エンジンで実行されているすべてのプラグインもシャットダウンされます。B2B エンジンおよび配信チャンネルを再起動する必要があります。
 - **WebLogic Integration** リポジトリでロジックプラグインの定義が変更された場合、たとえばチェーンが並べ替えられたり、ロジックプラグイン定義が追加または削除されたりした場合には、リポジトリの変更を反映するために配信チャンネルをシャットダウンしてから再起動する必要があります。
- **WebLogic Server** のインスタンスは、アップグレードされたバージョンのロジックプラグインのソースコードがインストールされた場合に、再起動する必要があります（また、Java 仮想マシン (JVM) も再ロードする必要があります）。

ロジック プラグインの開発と管理

カスタム ロジック プラグインを実装するには、開発のタスクと管理のタスクを組み合わせる必要があります。以下のステップで、必要な手順を説明します。

- ロジック プラグインのプログラミング ステップ
- 管理タスク

ロジック プラグインのプログラミング ステップ

この節では、ロジック プラグインのコードに関して実行する必要があるプログラミング ステップを説明します。ロジック プラグインはビジネス メッセージを各自の方法で処理しますが、すべてのロジック プラグインが実行しなければならないタスクもあります。

ロジック プラグインを実装するには、以下のステップを完了してください。

- 手順 1: 必要なパッケージをインポートする
- 手順 2: PlugIn インタフェースを実装する
- 手順 3: 例外処理モデルを指定する
- 手順 4: process メソッドを実装する
- 手順 5: メッセージ エンベロープからビジネス メッセージを取得する
- 手順 6: ビジネス メッセージを検証する
- 手順 7: ビジネス メッセージのプロパティを取得する
- 手順 8: ビジネス メッセージを処理する

この節で引用されるコード例は、次のようなロジック プラグインからのものです。

- WebLogic Integration B2B エンジンを通るビジネス メッセージを途中で取得する。
- メッセージの送信側、受信側、および関連する会話定義の名前を取得する。
- 取得した情報を含んだ 1 行を請求処理データベースに挿入する。

手順 1 : 必要なパッケージをインポートする

少なくとも、ロジック プラグインは以下のパッケージをインポートする必要があります。

- `com.bea.b2b.protocol.*`
- `com.bea.b2b.protocol.messaging.*`

`SentMsgCounter.java` ファイルから引用された以下のリストは、必要なパッケージをインポートする方法を示します。

コード リスト 3-1 必要なパッケージのインポート

```
import java.util.Hashtable;
import com.bea.b2b.protocol.*;
import com.bea.b2b.protocol.messaging.*;
import com.bea.eci.logging.*;
import javax.naming.*;
import javax.sql.DataSource;

// このパッケージは DB プールへのアクセスに必要
import java.sql.*;
```

手順 2 : PlugIn インタフェースを実装する

ロジック プラグインは `com.bea.b2b.protocol.PlugIn` インタフェースを実装する必要があります。次のリストに例を示します。

コード リスト 3-2 PlugIn インタフェースの実装

```
public class SentMsgCounter implements PlugIn
{
    ...
}
```

手順 3 : 例外処理モデルを指定する

以下の場合に、`PlugInException` が送出されます。

- ロジック プラグインによって実行時例外（`NullPointerException` など）が送出され、**WebLogic Integration** の処理コードに補足された場合。
- ロジック プラグインが、ロジック プラグインの処理で発生した問題を示すために例外を送出した場合。ロジック プラグインは、例外を自身で処理するか、**WebLogic Integration** 処理コードに通知することができます。

ロジック プラグインに指定された例外処理モデルは、例外が送出された場合に何が起きるかを決定します。ロジック プラグインは、`exceptionProcessingModel` メソッドを実装して、次の表に示される戻り値のうち 1 つを指定する必要があります。

表 3-5 例外処理モデルのオプション

クラス/インターフェース	説明
EXCEPTION_CONTINUE	<p>PlugInException が送出されても、処理を停止しないようにする。処理は、チェーン内の次のロジックプラグインまで継続される。</p> <p>このオプションは、ロジックプラグインの実行中にエラーが発生した場合でもビジネス メッセージの処理を継続できるようにするために使用する。</p>
EXCEPTION_STOP	<p>PlugInException が送出されたら、現在のロジックプラグインで処理が停止されるようにする。ビジネス メッセージは、チェーン内の次のロジックプラグインへ進まない。</p> <p>このオプションは、メッセージの処理をキャンセルして以後の処理を回避するために使用する。たとえば、ビジネスドキュメントを検証しているロジックプラグインは、不十分または不正なデータがあるドキュメントを拒絶することができる。</p>
EXCEPTION_UNWIND	<p>PlugInException が送出されたら処理を後戻りさせる。ビジネス メッセージは、チェーン内の次のロジックプラグインへ進まない。</p> <p>このオプションを使用して、メッセージの拒絶、B2B エンジンでの以後の処理の回避、およびプラグインによるすべての変更の破棄（チェーン内で先の順番にあるプラグインによるすべての変更も含む）ができる。例外が送出された場合に、このモデルが例外処理モデルであれば、チェーンで現在よりも前にあるすべてのプラグイン（ただし現在のロジックプラグインは除く）で <code>unwind</code> メソッドが呼び出される。呼び出しの順番は、チェーンをさかのぼる形になる。結果的に、チェーンによるすべての変更が後戻りによってキャンセルされる。</p> <p>たとえば、ロジックプラグインによってデータベース テーブルに 1 行が挿入されている場合は、ロジックプラグインの <code>unwind</code> メソッドでその行が削除される。</p> <p>注意： この例外処理モデルを使用するには、チェーン内のすべてのロジックプラグインが <code>unwind</code> メソッドを実装している必要がある。たとえ何もしないメソッドとしてでも、このメソッドの実装が必要です。</p>

3 ロジック プラグインの作成と追加 (非推奨)

ビジネス メッセージが拒絶された場合、次の処理をどうするかは、メッセージと関連付けられているサービス品質の指定だけでなくビジネスプロトコルにも依存します。たとえば、メッセージを送信した B2B アプリケーションにメッセージ配信の失敗が通知されたら、アプリケーションはビジネス メッセージの再送を試みます。

次のリストは、SentMsgCounter プラグインの `exceptionProcessingModel` メソッドの実装を示します。

コード リスト 3-3 例外処理モデルの指定

```
public int exceptionProcessingModel()
{
    return EXCEPTION_CONTINUE;
}
```

手順 4 : process メソッドを実装する

ビジネス メッセージを処理するために、ロジック プラグインは `process` メソッドを実装する必要があります。このメソッドには、ビジネス メッセージのメッセージ エンベロープを唯一のパラメータとして指定できます。次のリストは、SentMsgCounter クラスでの `process` メソッドの実装の冒頭を示します。ここでは、メッセージプロパティを格納するために後で使用される変数が定義されています。

コード リスト 3-4 process メソッドの実装

```
public void process(MessageEnvelope mEnv) throws PlugInException
{
    String sender, conversation;
    String tRecipient;
    Connection conn = null;
    Statement stmt = null;
    Message bMsg = null;
    ...
}
```

注意: ビジネス メッセージを処理するとき、ロジック プラグインはメッセージ エンベロープのみを変更できます。ビジネス メッセージ自体は変更できません。

手順 5: メッセージ エンベロープからビジネス メッセージを取得する

ロジック プラグインでビジネス メッセージの内容を調べる必要がある場合は、`MessageEnvelope` インスタンスの `getMessage` メソッドを呼び出して、ビジネス メッセージを `Message` オブジェクトとして取得する必要があります。

次のリストでは、`SentMsgCounter` クラスが `getMessage` メソッドを呼び出してメッセージ エンベロープからビジネス メッセージを取得します。

コード リスト 3-5 メッセージ エンベロープからビジネス メッセージを取得

```
if((bMsg = mEnv.getMessage())== null)
{
    throw new PlugInException("message is NULL");
}
```

手順 6: ビジネス メッセージを検証する

ロジック プラグインは、あるメッセージが処理されるべき有効なビジネス メッセージであるか、あるいは無視できるシステム メッセージであるかを判別することもできます。ビジネス メッセージを検証するために、ロジック プラグインでは `Message` インスタンスの `isBusinessMessage` メソッドを呼び出すことができます。次のリストでは、`SentMsgCounter` クラスで `isBusinessMessage` メソッドが使用されています。

コード リスト 3-6 ビジネス メッセージの検証

```
if (bMsg.isBusinessMessage())
{
    ...
}
```

手順 7: ビジネス メッセージのプロパティを取得する

ロジック プラグインは、ビジネス メッセージの特定のプロパティを取得するために `MessageEnvelope` または `Message` インスタンスで必要なメソッドを呼び出すことができます。次のリストでは、`SentMsgCounter` クラスが、メッセージの送信元である会話と関連付けられている会話定義の名前、ビジネス メッセージの送信側の名前、および受信側トレーディング パートナの名前を取得しています。

コード リスト 3-7 ビジネス メッセージのプロパティの取得

```
conversation= bMsg.getConversationType().getName();
sender = mEnv.getSender();
tRecipient = mEnv.getRecipient();
```

手順 8: ビジネス メッセージを処理する

ロジック プラグインは、ビジネス メッセージから必要な情報を取得し、処理することができます。たとえば、`SentMsgCounter` プラグインは、収集したメッセージ統計で請求処理データベースを更新します。

管理タスク

管理者がリポジトリにロジック プラグイン定義を追加するには、**WebLogic Integration B2B Console** の [ロジック プラグイン] タブで以下のタスクを実行します。

1. 以下のロジック プラグイン プロパティを指定します。
 - ロジック プラグイン名
 - `PlugIn` インタフェースを実装する **Java** クラス。このクラスはクラス ライブラリで補助クラスを呼び出すことができますが、自身はロジック プラグインの主要なエントリ ポイントでなければなりません。また、**Java** クラス ファイルは `CLASSPATH` で指定された場所に格納されている必要があります。

- Java クラスの初期化に使用するパラメータ名 / 値のペア
2. ビジネス プロトコルにロジック プラグインを割り当てます。
 3. チェーン内でのロジック プラグインの順位を指定します。

管理タスクの詳細については、『*B2B Integration 管理ガイド*』および『*B2B Integration Administration Console オンラインヘルプ*』を参照してください。

3 ロジック プラグインの作成と追加 (非推奨)

索引

A

API 3-7

C

CAInstance クラス 3-9

ConversationType クラス 3-8

E

EXCEPTION_CONTINUE 3-15

EXCEPTION_STOP 3-15

EXCEPTION_UNWIND 3-15

M

MessageEnvelope クラス 3-8

Message インタフェース 3-8

P

PartyInstance クラス 3-9

PayloadPart インタフェース 3-8

PlugInException クラス 3-7

PlugIn インタフェース 3-7, 3-14

process メソッド 3-16

R

RosettaNet

 フィルタ 3-6

 ルータ エンキュー 3-6

T

TradingPartnerInstance クラス 3-9

X

XOCP

 フィルタ 3-6

 ルータ 3-6

 ルータ エンキュー 3-6

XOCP フィルタ処理 フィルタ処理を参照

XOCP メッセージ処理

 XPath 式 2-7

 エンコーダ 2-15

 関連項目 XOCP メッセージ処理

 顧客提供のロジック プラグイン 2-11,
 2-14

 スケジューリング サービス (受信)
 2-14

 スケジューリング サービス (送信)
 2-8

 デコーダ 2-8

 転送サービス 2-8

 転送サービス、受信 2-15

 フィルタ 2-12

 メッセージコンテキスト ドキュメン
 ト 2-9

 ルータ 2-9

 ルータ ロジック プラグイン 2-9

 ルーティング サービス 2-11

XOCP ルーティング ルーティングを参照

XPath 言語 2-16

XPath 式 2-7

 作成 2-16, 2-20

 説明 2-16

 トレーディング パートナ用の作成
 2-21

 プロパティ 2-22

XPath 式の作成 2-16, 2-20

あ

アプリケーション 1-1

い

インタフェース

 PayloadPart 3-8

 PlugIn 3-7, 3-14

 メッセージ 3-8

え

エンキュー

 RosettaNet 3-6

 XOCP 3-6

エンコーダ 2-15

エンベロープ メッセージ エンベロープを
 参照 2-1

か

ガイドライン 3-9

開発タスク 3-12

カスタマ サポート vii

管理タスク 3-18

関連情報 vi

く

クラス

 CAInstance 3-9

 ConversationType 3-8

 MessageEnvelope 3-8

 PartyInstance 3-9

 PlugInException 3-7

 TradingPartnerInstance 3-9

け

言語、XPath 2-16

こ

顧客提供のロジック プラグイン
 フィルタ処理 2-14
 ルーティング 2-11

さ

サービス

 スケジューリング 2-8, 2-14

 転送 2-8, 2-15

サポート

 カスタマ vii

 テクニカル vii

し

システム ロジック プラグイン 3-6
受信側 2-12

す

スケジューリング サービス
 XOCP フィルタ処理 2-14
 XOCP ルーティング 2-8

そ

送信側 2-7

ち

チェーン 3-3

て

テクニカル サポート vii

デコーダ 2-8

転送サービス

 XOCP メッセージ処理 2-8

 XOCP メッセージ処理、受信 XOCP
 2-15

と

トレーディング パートナ、XPath 式の作成 2-21

は

パッケージ、インポート 3-13
パッケージのインポート 3-13

ひ

ビジネス メッセージ
 XOCP 処理 2-2
 概要 2-1
 検証 3-17
 受信 2-15
 送信 2-8
 プロパティ 3-18
 メッセージ エンベロープから取得 3-17
ビジネス メッセージの検証 3-17
ビジネス メッセージの取得 3-17

ふ

フィルタ
 RosettaNet 3-6
 XOCP 2-12, 3-6
フィルタ処理
 スケジューリング サービス、受信 2-14
 顧客提供のロジック プラグイン 2-14
プログラミング ステップ 3-12
プログラミングのステップ 3-12
プログラミングのタスク 3-12
プログラミングの方法 3-12
プロパティ
 XPath 式 2-22
 ビジネス メッセージ 3-18

ま

マニュアル

印刷 vi

 メッセージコンテキスト 2-9, 2-15
マニュアル入手先 vi
マニュアルの印刷 vi

め

メソッド、process 3-16
メッセージ エンベロープ
 概要 2-1
 ビジネス メッセージの取得 3-17
メッセージコンテキスト ドキュメント 2-9, 2-15
メッセージ処理
 XOCP 2-2
 XPath 式 2-7
 関連項目 *XOCP* メッセージ処理
 受信側 2-12
 送信側 2-7
メッセージの受信 2-15
XOCP メッセージの処理 XOCP 2-2
メッセージの送信 2-8
メッセージ ビジネス メッセージを参照

も

モデル、例外処理 3-14

る

ルータ
 RosettaNet エンキュー 3-6
 XOCP 2-9, 3-6
 XOCP エンキュー 3-6
ルータ ロジック プラグイン 2-9
ルーティング
 顧客提供のロジック プラグイン 2-11
 スケジューリング サービス、送信 2-8
ルーティング サービス 2-11
ルール 3-9

れ

例外処理モデル 3-14
連絡先 vii

ろ

ロジック プラグイン

API 3-7

RosettaNet フィルタ 3-6

RosettaNet ルータ エンキュー 3-6

XOCP フィルタ 3-6

XOCP ルータ 3-6

XOCP ルータ エンキュー 3-6

関連項目 フィルタ ロジック プラグイン
2-9

関連項目 ルータ ロジック プラグイン

顧客提供 2-11, 2-14

システム 3-6