



BEA WebLogic Integration™

Application Integration 入門

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。

Application Integration 入門

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.....	viii

1. Application Integration の概要

2. 統合フレームワーク

アダプタ.....	2-1
アプリケーションビュー.....	2-1
アプリケーションビュー.....	2-2
アプリケーションビューの主な機能.....	2-3
アプリケーション間の共通言語としての XML の使用.....	2-4
アプリケーションの機能を公開するためのサービスおよびイベント 定義の使用.....	2-4
サービスとイベントのデータ定義のための XML スキーマの使用.....	2-5
アダプタの双方向通信のサポート.....	2-5
アダプタおよびアプリケーションビューを使用するユーザ.....	2-5
システム管理者.....	2-5
アダプタ開発担当者.....	2-6
アダプタのユーザ.....	2-6
複数のユーザによる責任範囲の分担方法.....	2-6
設計時 GUI.....	2-7
設計時 GUI によるアプリケーションビューの作成.....	2-7
コンソールによるアプリケーションビューの管理.....	2-7
アプリケーションビュー定義と独自コード作成の使い分け.....	2-8
アプリケーションビューを使用する状況.....	2-8
アプリケーションビューを定義せずに独自のコードを作成する状況	

3. ADK

設計時フレームワーク	3-1
実行時フレームワーク	3-2
ロギングおよび監査フレームワーク	3-2
パッケージ化フレームワーク	3-2

4. 開発キット アダプタ

キット アダプタの開発手段	4-1
キット アダプタの使い方	4-1
DBMS アダプタ	4-2
サンプル アダプタ	4-2

5. BPM の Application Integration プラグイン

索引

このマニュアルの内容

『*Application Integration 入門*』の内容は、以下のとおりです。

- 「Application Integration の概要」に、Application Integration の概要を示します。Application Integration の主なコンセプト、特徴、およびコンポーネントアーキテクチャについて詳しく説明します。
- 「統合フレームワーク」に、アダプタの設計時 GUI を使用してアプリケーションを統合する方法を示します。
- 「ADK」に、WebLogic Integration の Adapter Development Kit (ADK) を使用してアダプタを開発する方法を示します。
- 「開発キット アダプタ」に、ADK が提供する完全な機能を持った EIS アダプタについての情報を示します。
- 「BPM の Application Integration プラグイン」に、Business Process Management (BPM) ツールによりアプリケーション統合機能を利用できるようにするプラグインについての情報を示します。

対象読者

このマニュアルは、次のユーザを対象としています。

- テクニカル アナリスト—インテグレーション ソリューションの構築、そしてアプリケーション サーバ、Application Integration 用ツール、ビジネスプロセス モデリング、B2B Integration を含む WebLogic Integration 環境の管理を行います。このガイドでは、テクニカルアナリストはシステム全体について完全に把握しているものと想定しています。
- ビジネス アナリスト—テクニカルアナリストと共に作業し、ビジネス インタフェースの機能をより正確に定義します。また、アプリケーションビュー、そしてそのアプリケーションビューを使用するビジネスプロセスの要件も定義します。

-
- システム管理者— WebLogic Integration server のセットアップ、チューニング、保守を行います。また、WebLogic Integration で使用するアダプタのインストールおよび保守も行います。

e-docs Web サイト

BEA 製品のドキュメントは、BEA Systems, Inc. の Web サイトで入手できます。BEA のホーム ページで [製品のドキュメント] をクリックするか、または「e-docs」という製品ドキュメント ページ (<http://edocs.beasys.co.jp/e-docs/index.html>) を直接表示してください。

このマニュアルの印刷方法

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を一度に 1 ファイルずつ印刷できます。

このマニュアルの PDF 版は、Web サイトで入手できます。Application IntegrationPDF を Adobe Acrobat Reader で開くと、マニュアルの全体（または一部分）を書籍の形式で印刷できます。PDF を表示するには、WebLogic Integration ドキュメントのホーム ページを開き、[PDF 版] をクリックして、印刷するマニュアルを選択します。

Adobe Acrobat Reader がない場合は、Adobe の Web サイト (<http://www.adobe.co.jp/>) で無料で入手できます。

関連情報

以下の関連情報があります。

- BEA WebLogic Server ドキュメント (<http://edocs.beasys.co.jp/e-docs/index.html>)

-
- BEA WebLogic Integration ドキュメント
(<http://edocs.beasys.co.jp/e-docs/index.html>)
 - XML スキーマの仕様 (<http://www.w3.org/TR/xmlschema-1/>)
 - Sun Microsystems, Inc. の J2EE コネクタ アーキテクチャの仕様
(<http://java.sun.com/j2ee/connector/>)
 - Sun Microsystems, Inc. の Java 関連サイト (<http://www.javasoft.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、ご使用の WebLogic Integration のリリースをお書き添えください。

本バージョンの 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、BEA WebSupport (<http://websupport.bea.com/custsupp>) 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。カスタマサポートへの連絡方法については、製品パッケージに同梱されているカスタマサポートカードにも記載されています。

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
- WebLogic Integration 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
[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> <code>chmod u+w *</code> <code>c:\startSever</code> <code>.doc</code> <code>wls.doc</code> <code>BITMAP</code> <code>float</code>
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> <code>void commit ()</code>
<i>斜体の等幅</i> <i>テキスト</i>	コード内の変数を示す。 <i>例</i> <code>String <i>expr</i></code>
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> <code>LPT1</code> <code>SIGNON</code> <code>OR</code>

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><i>例</i></p> <pre>import com.bea.adapter.cci.* . . . ;</pre>
. . .	<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 Application Integration の概要

WebLogic Integration は、企業内、および企業間のアプリケーションを協調させるための、標準化された統合ソリューションを提供します。特にこのソリューションが提供するのは、**WebLogic Integration Studio** により定義されるプロセスフローまたはカスタム コードで通信のエンドポイントを定義することによりアプリケーションを統合する手段です。

WebLogic Integration はアプリケーションを統合するツールとして以下を提供します。

- 統合フレームワーク
- Adapter Development Kit (ADK)
- EIS アダプタ
- WebLogic Integration Studio 用の Application Integration プラグイン

これらのツールを使えば、企業内のエンタープライズ情報システム (EIS) をすべて統合することができます。典型的な IT 組織は、複数の高度に専門化されたアプリケーションを使用しています。共通の統合プラットフォームを使用しなければ、このようなアプリケーションの統合には膨大で高度に専門化された作業が要求されます。

WebLogic Integration は、アダプタを使用して今使用されているアプリケーションと今後使用されるアプリケーションを統合する、企業規模のフレームワークを 1 つ構成します。アダプタを使用すれば、1 つのアプリケーション サーバへの統合が可能になり、統合が容易になります。また、すべてのアプリケーションをそれぞれ互いに統合する必要がありません。

WebLogic Integration ではこれらのアダプタを使用し、ビジネスにフォーカスをあてた EIS インタフェースを定義することができます。これらのインタフェースはアプリケーション ビューと呼ばれ、アプリケーションのサービスおよびイベントに対して単純でわかりやすく整合性の取れたインタフェースを提供します。アダプタ ユーザとは異なり、アプリケーション ビュー ユーザには EIS やクライアント インタフェースに関する詳細な知識は不要です。このため、テクニカル

アナリストのようにプログラマではないユーザがアプリケーションビューを使うことができます。アダプタの場合は、EIS についての詳細な知識がなければ使いこなすのは難しいと言えます。

次のセクションでは、**WebLogic Integration** が備えているアプリケーション統合ツールをそれぞれ簡単に説明します。詳細については、『*アダプタの開発*』と『*Application Integration ユーザーズガイド*』を参照してください。

2 統合フレームワーク

WebLogic Integration で提供される統合フレームワークは、アプリケーションビューを動作させるための標準アーキテクチャを提供します。アプリケーションビューは企業のアプリケーションに対する実務指向のインタフェースです。

この付録のトピックは以下のとおりです。

- アダプタ
- アプリケーションビュー
- 設計時 GUI

アダプタ

Application Integration は、企業のアプリケーションを統合するアダプタと、アプリケーションビューをベースにした統合アプリケーションを使用します。統合フレームワークでは、企業内のシステムをケーブル等で接続するのではなく、アプリケーションサーバとシステムを接続するアダプタと呼ばれるソフトウェアコンポーネントを構築します。EIS のアダプタをデプロイすると、他のコンポーネントやアプリケーションがこのアダプタを使用して、その EIS のアダプタにアクセスできるようになります。

アプリケーションビュー

アプリケーションビューは、アダプタとそのアダプタの EIS 機能間の抽象化レイヤとしての役目を持っています。アプリケーションビューを使うと、アダプタへのアクセス方法を単純化できます。EIS へのアクセス機能を直接起動せずに、アダプタのアプリケーションビューの編集、新規アプリケーションビュー

の作成、古くなったビューの削除だけで済みます。アプリケーションビューで構成される抽象化レイヤを使用すれば、プログラマでなくても、アダプタが提供するサービスやイベントの保守ができるようになります。

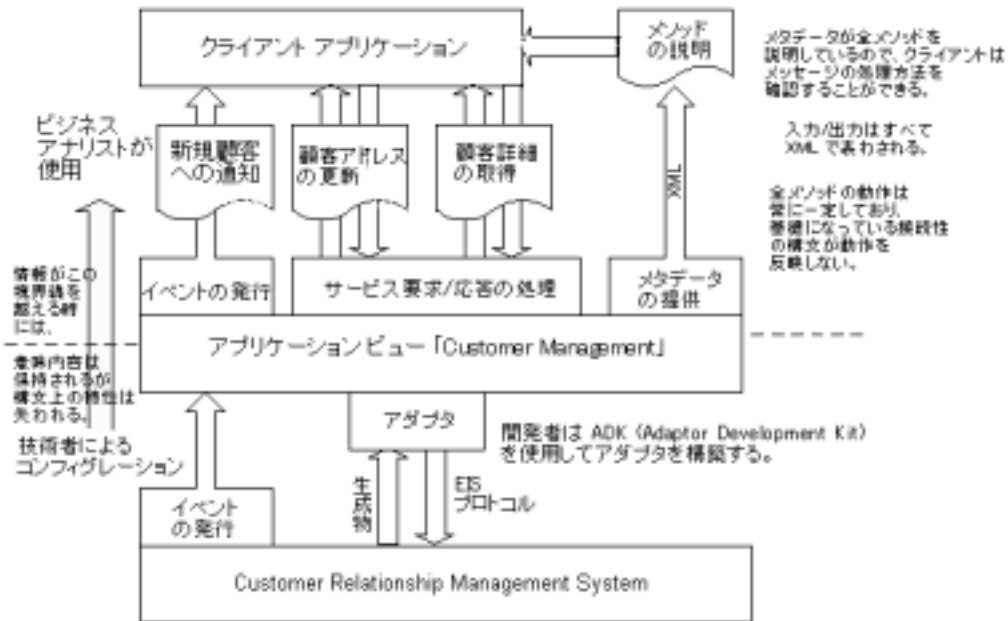
アプリケーションビューはそれぞれ、1つのアダプタのEIS上のビジネス機能セットを定義します。アダプタ作成後、Webベースインタフェースを使用して、独自のアプリケーションビューを定義できます。アプリケーションビューを使うと、アダプタが提供するアプリケーション機能を表示できます。

ビジネスアナリストまたはテクニカルアナリストのどちらかがアダプタを使用してアプリケーションを提供する場合、ビジネス上のある特定の目的向けにアプリケーションビューをカスタマイズすることができます。ビジネスの目的はビジネスアナリストが定義します。たとえばCRM(カスタマリレーションシップマネージメント)システムのアダプタに「顧客管理」アプリケーションビューを定義する場合、通常このアプリケーションビューには、顧客管理関連のサービスおよびイベントだけを含むようにします。ただし、必要に応じた機能を含むようにアプリケーションビューを作成することもできます。アプリケーションビューはビジネス上の特定の目的向けにカスタマイズことができ、この点で他の多くのEAIシステムが提供している「ワンサイズフィツオール」型のアプローチよりもはるかに優れています。

アプリケーションビュー

アプリケーションの機能に対応するビジネスレベルのビューによって、プログラマとテクニカルアナリストの役割が論理的に区分されます。たとえば、ビジネスレベルのビューにより、テクニカルアナリストはSQLについての知識を持たなくてもデータベース上にレコードを作成することができます。Figure 2-1はapplication integration環境での作業で使用されるアプリケーションビューのダイアグラムを提供します。

図 2-1 アプリケーション ビューの役割



アプリケーション ビューの主な機能

WebLogic Integration では、アプリケーション ビューをアダプタの主要なユーザインタフェースとして使用するため、競合する EAI テクノロジーが通常提供していないいくつかの機能を持ちます。

- アプリケーション間の共通言語としての XML の使用
- アプリケーションの機能を公開するためのサービスおよびイベント定義の使用
- サービスとイベントのデータ定義のための XML スキーマの使用
- アダプタの双方向通信のサポート

ここからは、これらの機能について説明します。

アプリケーション間の共通言語としての XML の使用

EAI のシナリオでは、1 つの共通データ フォーマットを使用して各 EIS と WebLogic Server の統合を実行するほうが、各 EIS がそれぞれ独自のデータ フォーマットを使用して他の EIS との統合を実行するよりもはるかに容易でありまた効率的です。共通のデータ フォーマットを使用すると、すべてのアプリケーションが標準言語を使用して通信できるようになります。WebLogic Integration は広く採用されつつあるデータ交換フォーマットである XML を共通のデータ フォーマットとして使用しています。

WebLogic Integration の環境では、ほぼすべてのメッセージが XML ドキュメントとして送信されます。

- 各サービスについて、アプリケーション ビューは XML 要求メッセージを送信し、XML 応答メッセージを返します。
- イベントが生成されると、登録されているイベント リスナはイベント情報を XML メッセージとして受け取ります。アプリケーション ビューはアダプタを使用して EIS 固有のフォーマットと XML の相互変換を行います。

アダプタは XML を使用してアプリケーションのデータ フォーマットを変換するので、ビジネス アナリストがフォーマットを熟知している必要がありません。ビジネス アナリストがアダプタを使用する場合に必要な知識は、アプリケーション ビューの定義方法と使用方法です。とりわけ、すべてのアダプタが、アプリケーション ビュー定義のために同じ Web ベース インタフェースを使用しているため、現在使用されているアダプタの使用法だけでなく、今後提供されるアダプタについても容易に習得することができます。このように、XML は開発者にとってもビジネスアナリストにとっても EAI を使いやすくしてくれます。

アプリケーションの機能を公開するためのサービスおよびイベント定義の使用

アプリケーション ビューは、特定のビジネス用途向けのイベントおよびサービスを、ベースとなるアダプタを介してサポートします。イベントは、アプリケーションが生成するメッセージを、後で説明するパブリッシュ/サブスクライブ モデルで管理するために使用します。サービスは、ユーザによって呼び出されるビジネス機能です。サービスを呼び出すことによって、メッセージが要求/応答モデルの後に続くアプリケーションに送信されます。イベント、サービス要求/応答のどちらも XML ドキュメントとしてシステムに渡されます。

サービスとイベントのデータ定義のための XML スキーマの使用

各アプリケーションビューでは、イベント、サービス要求およびサービス応答の XML 情報を指すメタデータとして XML スキーマを使用します。このメタデータを使用すると、アプリケーションビュー イベントまたはサービスがどのようなデータを必要とするかがわかります。

アダプタの双方向通信のサポート

現時点では、J2EE コネクタ アーキテクチャ仕様バージョン 1.0 は EIS がアプリケーション サーバまたはクライアントと通信を開始するためのガイドラインを提供していません。WebLogic Integration ではイベント アダプタでこの通信機能を実現しています。

アダプタおよびアプリケーションビューを使用するユーザー

企業内では、複数の社員が、アダプタ、アプリケーションビュー、そしてこれらのアプリケーションビューのサービスとイベントの保守を分担して行います。多くの企業では、これらの社員は以下の 3 つのグループのいずれかに該当します。

- システム管理者
- アダプタ開発担当者
- アダプタのユーザー

システム管理者

このマニュアル、そして『*Application Integration ユーザーズガイド*』でシステム管理者として説明されるのは、WebLogic Integration Adapter Development Kit (ADK) のインストールの責任者です。

アダプタ開発担当者

ソフトウェア開発者、または上位レベルの技術者の場合はアダプタ開発者になります。アダプタ開発者は通常、このアダプタに付随する **Common Client Interface (CCI)** の開発と設計を行います。CCI とは、リソース アダプタと関連付けられた機能を表わすアプリケーション ビューを設計するためにアダプタ ユーザが対話するインタフェースです。詳細については、『アダプタの開発』を参照してください。

アダプタのユーザ

ビジネス アナリスト、EIS のスペシャリスト、テクニカル アナリストの場合はアダプタのユーザになります。アダプタのユーザは、通常アダプタの作成は行いませんが、開発者に対してアダプタの仕様を提供することができます。アダプタが作成されると、多くの場合、ユーザがアプリケーション ビューを定義して管理することができます。

複数のユーザによる責任範囲の分担方法

表 2-1 は複数のユーザによるタスクと責任の分担方法を示しています。

表 2-1 共通のジョブとそのオーナー

タスク	システム 管理者	アダプタ開発担当 者	アダプタ ユーザ
WebLogic Integration インストール	X		
アダプタの開発		X	
アプリケーション ビュー定義用ユーザ インタフェースの開発		X	
アダプタをデプロイするための WebLogic Integration のコンフィグレーション	X	X	
アプリケーション ビューの定義			X
アプリケーション ビューの接続パラメータ のコンフィグレーション			X
サービスとイベントの追加およびテスト			X

表 2-1 共通のジョブとそのオーナー

タスク	システム 管理者	アダプタ開発担当 者	アダプタ ユーザ
アプリケーション ビュー フォルダのセッ トアップ			X

設計時 GUI

WebLogic Integration の統合フレームワークは、各アダプタの Common Client Interface (CCI) を作成する手段を開発者に提供します。CCI により、アプリケーション コンポーネントおよび Enterprise Application integration (EAI) フレームワークが共通のクライアント API を使用して、異種 EIS 間での対話を実現します。アダプタの設計時 GUI を使用すれば、プログラマでなくてもアプリケーション ビューの作成、デプロイ、テストおよび編集を素早く行うことができ、またサービスやイベントを追加してカスタマイズすることができます。

設計時 GUI によるアプリケーション ビューの作成

アダプタの設計時 GUI の主な目的は、アプリケーション ビューの定義、デプロイおよびテストです。アプリケーション ビュー定義の詳細情報は、『*Application Integration ユーザーズ ガイド*』を参照してください。

コンソールによるアプリケーション ビューの管理

Application View Console を使用して、企業内の全アプリケーション ビューへのアクセス、編成、および編集を行います。Application View Console を使用してフォルダを新しく作成し、新しくアプリケーション ビューを追加することができます。このようなフォルダを使用すると、使用されるアダプタとは関係なく、アプリケーション ビューを独自のナビゲーション スキームでまとめることができます。

アプリケーション ビュー管理の詳細な情報については、『*Application Integration ユーザーズ ガイド*』の「WebLogic Integration Application View Console の使い方」を参照してください。

アプリケーション ビュー定義と独自コード作成の使い分け

EIS の機能を公開する方法は、アダプタの設計時 GUI を使用する方法だけではありませんが、通常はこれが最も便利な方法です。サービスの要求とイベントをサポートするには、アプリケーション ビューを定義する方法と、同様の機能を持つコードを独自に作成する方法があります。アダプタのアプリケーションが提供する機能を公開するには、最低でも各アダプタにアプリケーション ビューを定義する必要があります。ただし、ユーザが通常以上に管理を行うことを希望する場合は、アダプタのリソースへアクセスできるよう、カスタム コードを作成することもできます。企業のニーズを満たすのが、アプリケーション ビューを定義することか、独自のコードを作成することか、あるいは両方を組み合わせた方法を導入するべきかを見極める必要があります。

アプリケーション ビューを使用する状況

多くの EIS アプリケーションは、アプリケーション ビューを定義することで容易にシステムへ統合することができます。次のような場合、アプリケーション ビューを定義する必要があります。

- 企業内で複数の EIS システムを使用しており、そのシステム全体の詳細を把握している開発者がいない。
- ビジネスプロセスの構築と管理に WebLogic Integration Studio を使用したい。
- アダプタのパラメータまたはそのプロセスを更新する必要がある。

アプリケーション ビューを定義せずに独自のコードを作成する状況

一般的に、アダプタのインタフェースとして独自のコードを作成するのは以下のような場合に限ります。

- 企業内で EIS システムを 1 つだけ使用している。
- 開発者が対象のビジネス プロセスに関連する各 EIS の詳細を完全に把握している。
- **WebLogic Integration Studio** で作成したコードを使用する必要がない。
- コードが変更される可能性が無い。

3 ADK

アダプタ プロバイダまたは開発者の場合は、**WebLogic Integration Adapter Development Kit (ADK)** を使用して独自の J2EE 互換の EIS アダプタを作成することができます。ADK とは、ツールと Java クラスで構成される 4 つのフレームワークの集まりです。これらのフレームワークを組み合わせ、アダプタを迅速に開発し、簡単にテスト、パッケージ化、配布することができます。ADK フレームワークを使用するアダプタ作成についての詳細は、『**アダプタの開発**』を参照してください。

このセクションでは、次の 4 つのコンポーネント フレームワークについて説明します。

- 設計時フレームワーク
- 実行時フレームワーク
- ロギングおよび監査フレームワーク
- パッケージ化フレームワーク

設計時フレームワーク

アダプタの設計時インタフェースは、**WebLogic Integration** をホストする **WebLogic Server** のアプリケーションビューの定義とデプロイに使用します。**WebLogic Integration ADK** で任意のアダプタを使用してビルドすると、設計時ユーザインタフェースも開発することができます。アダプタの設計時ユーザインタフェースは一般的な **Web** ブラウザであればどこからでもアクセスできます。このインタフェースにより、プログラマでないユーザでもコードを書くという作業を行わずにアダプタを対話的に操作することができます。ユーザインタフェースを使えば、ビジネスアナリストがアダプタにログインし、独自のカスタムアプリケーションビューを定義することができます。設計時ユーザインタフェースがなければ、高度な技術を持つユーザ以外はアダプタを使いこなすことはできません。

ADK には、アダプタの設計時ユーザ インタフェース開発を容易にするための、設計時フレームワークと呼ばれる Java クラスおよびツールが含まれます。このフレームワークはプログラマでないユーザでもアダプタを使用できるという強力な機能を備えています。開発プロセスが単純化されることにより、ADK の設計時フレームワークはアダプタのユーザ層を広げ、ビジネス プロセス導入時のビジネス アナリストの役割をより高度なものにします。

実行時フレームワーク

ADK の実行時フレームワークは、イベント アダプタの開発をサポートするすべての機能のほか、拡張が可能なイベント ジェネレータを提供します。また、サービス アダプタの開発をサポートするため、実行時フレームワークは完全かつ最小限の機能を備えた J2EE 準拠アダプタを提供します。アダプタ開発者は、この基本フレームワークから開始し、これを企業のニーズに合わせて拡張することによってコーディングとデバッグの時間を短縮できます。

ロギングおよび監査フレームワーク

管理者は、アダプタから出される警告メッセージを後で監査できるように自動的にログに記録するようにしておく必要があります。ログの記録と監査をサポートする機能が組み込まれたアダプタの開発を容易にするため、ADK にはロギングと監査のフレームワークが用意されています。開発するアダプタはすべて、インターナショナルライズおよびローカライズされた警告メッセージを生成することができ、このメッセージを複数の送信先に配信することができます。

パッケージ化フレームワーク

サードパーティのアダプタプロバイダであれば、カスタマに提供するアダプタの設定に ADK パッケージ化フレームワークを使用できます。このフレームワークを使用すると、パッケージ化に必要なアーカイブ ファイルと環境ファイルの作成が容易になります。

4 開発キット アダプタ

BEA WebLogic Integration Adapter Development Kit (ADK) にはサンプルアダプタが含まれており、これを使用してユーザ独自のアダプタ、DBMS アダプタとサンプルアダプタを開発できます。ADK を使用して独自のアダプタを開発する場合は、まずキットに含まれているアダプタで学習することから始めます。これらのアダプタは一般的でシンプルなものですが、ADK でビルドするアダプタのサンプルとしては優れています。

キット アダプタの開発手段

開発キットのアダプタはすべて、ADK を使用して開発されています。ADK を使用すれば高度なアダプタを開発できますが、キットに含まれているアダプタは分析や理解がしやすいようにあえてシンプルなものになっています。

キット アダプタの使い方

アダプタ プロバイダまたは開発者は、まずキット アダプタについてよく研究して ADK についての知識を深め、また独自のモデルとしてどのようにアダプタを使えばよいかを決定します。キット アダプタはすべて Sun Microsystems の Java 2 Platform, Enterprise Edition (J2EE) バージョン 1.3 をベースにしています。ADK についての詳細は、『アダプタの開発』を参照してください。次のセクションでは、キット アダプタについて詳しく説明します。

DBMS アダプタ

WebLogic Integration には、WebLogic Server を EIS を使用する簡単なリレーショナル データベースと統合する DBMS アダプタがあります。DBMS アダプタは、アダプタ開発者とアダプタ プロバイダが、個々のデータベースについて学習せずにアダプタと ADK についての知識を得るための適切なサンプルです。

DBMS アダプタで使用する適切なデータベースがない場合は、WebLogic Integration に含まれる PointBase データベースを使用することができます。

DBMS アダプタは次の機能をサポートします。

- 全データベース レコードの取得
- データベース レコード セットの取得
- データベース レコードの書き込み
- テーブルからのレコードの更新、挿入、削除通知の受信

詳細については、『アダプタの開発』の「DBMS アダプタを使用したアダプタ開発方法の学習」を参照してください。

サンプル アダプタ

サンプル アダプタは新しいアダプタのテンプレートとして提供されています。設計時コンポーネント、サービス アダプタ、イベント アダプタがこれに含まれます。ソース コードはサービス アダプタとイベント アダプタの構造を明確に説明しています。

また、サンプル アダプタの EIS 実装はシンプルであり、サービスの呼び出しによって関連イベントをトリガする方法のデモになっています。各コンポーネントには独自のアダプタを作成する場合にロジックをどこに追加するかを示すコメントが付けられています。

5 BPM の Application Integration プラグイン

WebLogic Integration はユーザのアプリケーションの統合に必要なツールに加えて、プロセスレベルで統合を行えるようにする強力なツールセットも備えています。Business Process Management (BPM) ツールの中で中心となるのは WebLogic Integration Studio です。これは、ビジネスアナリストがアプリケーションを統合するビジネスプロセスワークフローを視覚的に構築し、保守できるようにするためのグラフィカルユーザインタフェースです。Studioの詳細については、『*WebLogic Integration Studio ユーザーズガイド*』を参照してください。

WebLogic Integration Studio を始めとする BPM ツールは、Application Integration プラグインをデプロイしなければアプリケーションビューサービスやイベントと一緒に使用することができません。いったんプラグインをデプロイすれば、ビジネスアナリストなどプログラマ以外の人でも、Studio が提供する使い易い調和の取れたインタフェースを使用してサービスを呼び出したり、ワークフローからイベントの通知を受信したりできます。詳細な情報については、『*WebLogic Integration Studio ユーザーズガイド*』の「ワークフローリソースのコンフィグレーション」を参照してください。

WebLogic Integration 機能とツールは次の手順で使用します。

1. ユーザの企業における全アプリケーションに対して、当マニュアルで説明するツールを使用してアプリケーションレベルの統合を実行する。
2. 当マニュアルで説明するツールを使用して、アダプタを開発し、アプリケーションビューを定義する。
3. WebLogic Integration Studio を立ち上げてビジネスプロセスワークフローを作成する。
4. ユーザのアプリケーションビューサービスとイベントをワークフローに組み込む。
5. ワークフローをデプロイする。

6. 既存のワークフロー、アダプタ、サービス、イベントを更新する場合は、**Studio** で実行できる。この方法で変更すれば、**Studio** を使用する場合は必要のない、時間のかかるコンパイルを行わずに済みます。

WebLogic Integration Studio を使用して、アプリケーション ビュー サービスとイベントを持つビジネス プロセス ワークフローを構築するためには、『*Application Integration ユーザーズ ガイド*』の「**Studio** でのアプリケーション ビューの使用」を参照してください。

索引

A

Adapter Development Kit (ADK)
 実行時フレームワーク 3-2
Application View Console 2-7

E

edocs vi, vii

G

GUI
 Application View Console 2-7

J

J2EE コネクタ アーキテクチャの仕様 vii

S

Sun Microsystems vii
Sun Microsystems, Inc. の Java 関連サイト
 vii

W

WebLogic Server vi

X

XML
 スキーマ 2-5
XML スキーマの仕様 vii

あ

アーキテクチャ 2-1

アプリケーション ビュー
 イベントおよびサービス 2-4
 使用する状況 2-8

い

イベント 2-4
イベント ジェネレータ 3-2
印刷、製品のマニュアル vi

え

エンドポイント 1-1

か

カスタマ サポート情報 vii
関連情報 vi
 J2EE コネクタ アーキテクチャの仕様
 vii
 Sun Microsystems の Java 関連サイト
 vii
 WebLogic Server ドキュメント vi
 XML スキーマの仕様 vii

さ

サービス 2-4
サポート
 テクニカル vii

し

システム管理者 vi

て

テクニカルアナリスト v

と

統合フレームワーク 2-1

設計時 GUI 2-7

独自のコード

アプリケーションビュー定義の代替

手段 2-8

ひ

ビジネスアナリスト v, vi

ふ

プロセスフロー 1-1

ま

マニュアル入手先 vi

ゆ

ユーザインタフェース 2-7