

Oracle® BPEL Process Manager

クイック・スタート・ガイド

10g (10.1.3.1.0)

部品番号 : B31876-01

2006 年 11 月

Oracle BPEL Process Manager クイック・スタート・ガイド, 10g (10.1.3.1.0)

部品番号 : B31876-01

原本名 : Oracle BPEL Process Manager Quick Start Guide, 10g (10.1.3.1.0)

原本部品番号 : B28983-01

原本著者 : Mark Kennedy

原本協力者 : Guru Balse, Dave Shaffer, Raj Venkatesan

Copyright © 2005, 2006, Oracle. All rights reserved.

制限付権利の説明

このプログラム（ソフトウェアおよびドキュメントを含む）には、オラクル社およびその関連会社に所有権のある情報が含まれています。このプログラムの使用または開示は、オラクル社およびその関連会社との契約に記された制約条件に従うものとします。著作権、特許権およびその他の知的財産権と工業所有権に関する法律により保護されています。

独立して作成された他のソフトウェアとの互換性を得るために必要な場合、もしくは法律によって規定される場合を除き、このプログラムのリバース・エンジニアリング、逆アセンブル、逆コンパイル等は禁止されています。

このドキュメントの情報は、予告なしに変更される場合があります。オラクル社およびその関連会社は、このドキュメントに誤りが無いことの保証は致し兼ねます。これらのプログラムのライセンス契約で許諾されている場合を除き、プログラムを形式、手段（電子的または機械的）、目的に関係なく、複製または転用することはできません。

このプログラムが米国政府機関、もしくは米国政府機関に代わってこのプログラムをライセンスまたは使用する者に提供される場合は、次の注意が適用されます。

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このプログラムは、核、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーションへの用途を目的としておりません。このプログラムをかかるとして使用する際、上述のアプリケーションを安全に使用するために、適切な安全装置、バックアップ、冗長性（redundancy）、その他の対策を講じることは使用者の責任となります。万一かかるとしてプログラムの使用に起因して損害が発生いたしましても、オラクル社およびその関連会社は一切責任を負いかねます。

Oracle、JD Edwards、PeopleSoft、Siebel は米国 Oracle Corporation およびその子会社、関連会社の登録商標です。その他の名称は、他社の商標の可能性ががあります。

このプログラムは、第三者の Web サイトへリンクし、第三者のコンテンツ、製品、サービスへアクセスすることがあります。オラクル社およびその関連会社は第三者の Web サイトで提供されるコンテンツについては、一切の責任を負いかねます。当該コンテンツの利用は、お客様の責任になります。第三者の製品またはサービスを購入する場合は、第三者と直接の取引となります。オラクル社およびその関連会社は、第三者の製品およびサービスの品質、契約の履行（製品またはサービスの提供、保証義務を含む）に関しては責任を負いかねます。また、第三者との取引により損失や損害が発生いたしましても、オラクル社およびその関連会社は一切の責任を負いかねます。

目次

はじめに	iii
対象読者	iv
ドキュメントのアクセシビリティについて	iv
関連ドキュメント	iv
表記規則	v
サポートおよびサービス	v
1 Oracle BPEL Process Manager の概要	
BPEL: SOA の基礎	1-2
BPEL の概要	1-2
Oracle BPEL Process Manager	1-3
Oracle BPEL Process Manager の概要	1-3
Oracle BPEL Process Manager の特長	1-4
Oracle BPEL Process Manager の機能の概要	1-4
Oracle Application Server SOA Suite および Oracle JDeveloper のインストール	1-5
Oracle Application Server SOA Suite のインストール	1-6
ポートの識別	1-6
Oracle JDeveloper のインストール	1-6
Oracle Application Server および統合サーバー接続のインストール	1-7
Oracle BPEL Process Manager コンポーネントの起動および停止	1-8
Web ブラウザ設定でのホスト名の確認	1-10
ネームスペースとネームスペース URI について	1-10
2 信用フローのチュートリアル	
信用フローの概要	2-2
チュートリアルの使用方法	2-2
Oracle BPEL Server と Oracle JDeveloper の起動	2-2
サービスの起動とテスト	2-2
アプリケーションとプロジェクトの作成	2-3
WSDL ファイルのソース・コードの確認	2-4
BPEL ファイルのソース・コードの確認	2-5
XSD ファイルの編集	2-6
設計フェーズにおけるプロジェクトの検証	2-8

信用格付けサービス用のパートナ・リンクの作成と構成	2-9
信用格付けサービスのパートナ・リンクの作成	2-9
Scope アクティビティの作成	2-11
Scope アクティビティ内での Invoke アクティビティの作成	2-11
Scope アクティビティ内での 1 つ目の Assign アクティビティの作成	2-12
Scope アクティビティ内での 2 つ目の Assign アクティビティの作成	2-13
信用フロー・プロセスの検証、コンパイルおよびデプロイ	2-14
信用フロー・プロセスの実行	2-15

3 融資調達 BPEL プロセスの確認

LoanFlowPlus プロセスの概要	3-2
Oracle JDeveloper でのプロセスの作成と確認	3-3
サービスの起動とテスト	3-3
Oracle JDeveloper での LoanFlowPlus プロセスのオープン	3-4
フォルト管理	3-5
非同期融資プロセッサ・サービスとの対話	3-6
並列処理の確認	3-8
ヒューマン・タスクの確認	3-8
Web サービスの起動フレームワークの確認	3-10
BPEL プロセスを開始するためのカスタム・ユーザー・インタフェースの作成	3-11
LoanFlowPlus プロセスの開始	3-11
ビジュアル監査証跡の確認	3-13
進行中の BPEL プロセスのデバッグ	3-15
非同期の StarLoan サービスの実行	3-15
ヒューマン・タスクの実行	3-17
パフォーマンス・チューニング	3-18
プロセスのライフサイクルの管理	3-19
プロセス・レポートの作成と表示	3-19
レポート・ダッシュボードの作成	3-19
Oracle Enterprise Manager 10g BPEL Control のレポートの表示	3-20
BAM のレポートの表示	3-21
開発環境における BPEL プロセスのテスト	3-22
Web サービス管理ソリューションとの統合	3-24
デモおよびチュートリアル	3-26

索引

はじめに

このマニュアルは、Oracle BPEL Process Manager の概要および使用方法について説明します。

- [対象読者](#)
- [ドキュメントのアクセシビリティについて](#)
- [関連ドキュメント](#)
- [表記規則](#)
- [サポートおよびサービス](#)

対象読者

このマニュアルは、Oracle BPEL Process Manager をインストールして使用するユーザーを対象としています。

ドキュメントのアクセシビリティについて

オラクル社は、障害のあるお客様にもオラクル社の製品、サービスおよびサポート・ドキュメントを簡単にご利用いただけることを目標としています。オラクル社のドキュメントには、ユーザーが障害支援技術を使用して情報を利用できる機能が組み込まれています。HTML 形式のドキュメントで用意されており、障害のあるお客様が簡単にアクセスできるようにマークアップされています。アクセシビリティの標準規格は改善されつつあります。オラクル社はドキュメントをすべてのお客様がご利用できるように、市場をリードする他の技術ベンダーと積極的に連携して技術的な問題に対応しています。詳細は、Oracle Accessibility Program の Web サイト <http://www.oracle.com/accessibility/> を参照してください。

ドキュメント内のサンプル・コードのアクセシビリティについて

スクリーン・リーダーは、ドキュメント内のサンプル・コードを正確に読めない場合があります。コード表記規則では閉じ括弧だけを行に記述する必要があります。しかし一部のスクリーン・リーダーは括弧だけの行を読まない場合があります。

外部 Web サイトのドキュメントのアクセシビリティについて

このドキュメントにはオラクル社およびその関連会社が所有または管理しない Web サイトへのリンクが含まれている場合があります。オラクル社およびその関連会社は、それらの Web サイトのアクセシビリティに関しての評価や言及は行っておりません。

Oracle サポート・サービスへの TTY アクセス

アメリカ国内では、Oracle サポート・サービスへ 24 時間年中無休でテキスト電話 (TTY) アクセスが提供されています。TTY サポートについては、(800)446-2398 にお電話ください。

関連ドキュメント

詳細は、次の Oracle ドキュメントを参照してください。

- 『Oracle BPEL Process Manager Order Booking チュートリアル』
- 『Oracle BPEL Process Manager 開発者ガイド』

リリース・ノート、インストール関連ドキュメント、ホワイト・ペーパーまたはその他の関連ドキュメントは、OTN-J (Oracle Technology Network Japan) から、無償でダウンロードできます。OTN-J を使用するには、オンラインでの登録が必要です。登録は、次の Web サイトから無償で行えます。

<http://otn.oracle.co.jp/membership/>

Oracle BPEL Process Manager のドキュメント、テクニカル・ノートまたはその他の関連ドキュメントは、Oracle Technology Network (OTN) の Oracle BPEL Process Manager のサイトからダウンロードできます。

<http://www.oracle.com/technology/bpel/>

すでに OTN-J のユーザー名およびパスワードを取得している場合は、次の URL で OTN-J Web サイトのドキュメントのセクションに直接接続できます。

<http://www.oracle.co.jp/document/>

次の URL で入手可能な Business Process Execution Language for Web Services Specification を参照してください。

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnbizspec/html/bpel1-1.asp>

表記規則

このマニュアルでは次の表記規則を使用します。

規則	意味
太字	太字は、操作に関連する Graphical User Interface 要素、または本文中で定義されている用語および用語集に記載されている用語を示します。
イタリック	イタリックは、ユーザーが特定の値を指定するプレースホルダ変数を示します。
固定幅フォント	固定幅フォントは、段落内のコマンド、 URL 、サンプル内のコード、画面に表示されるテキスト、または入力するテキストを示します。

サポートおよびサービス

次の各項に、各サービスに接続するための URL を記載します。

Oracle サポート・サービス

オラクル製品サポートの購入方法、および Oracle サポート・サービスへの連絡方法の詳細は、次の URL を参照してください。

<http://www.oracle.co.jp/support/>

製品マニュアル

製品のマニュアルは、次の URL にあります。

<http://otn.oracle.co.jp/document/>

研修およびトレーニング

研修に関する情報とスケジュールは、次の URL で入手できます。

<http://www.oracle.co.jp/education/>

その他の情報

オラクル製品やサービスに関するその他の情報については、次の URL から参照してください。

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

<http://otn.oracle.co.jp>

注意： ドキュメント内に記載されている URL や参照ドキュメントには、Oracle Corporation が提供する英語の情報も含まれています。日本語版の情報については、前述の URL を参照してください。

Oracle BPEL Process Manager の概要

この章では、Business Process Execution Language (BPEL) および Oracle BPEL Process Manager を使用して、同期および非同期サービスをエンドツーエンド BPEL プロセス・フローに発展させることにより、サービス指向アーキテクチャ (SOA) ベースのアプリケーションを設計できるようにする方法の概要を説明します。また、Oracle BPEL Process Manager、Oracle BPEL Server および他の主要なコンポーネントのインストールおよび起動方法についても説明します。

この章の内容は次のとおりです。

- [BPEL: SOA の基礎](#)
- [Oracle BPEL Process Manager](#)
- [Oracle Application Server SOA Suite および Oracle JDeveloper のインストール](#)
- [Oracle BPEL Process Manager コンポーネントの起動および停止](#)
- [Web ブラウザ設定でのホスト名の確認](#)
- [ネームスペースとネームスペース URI について](#)

BPEL: SOA の基礎

多くの企業では、複合型アプリケーションの構築に伴う統合要件に対処する方法として Web サービスと SOA に注目しています。SOA は 10 年以上前から存在していましたが、採用するインタフェースについては混乱がありました。BPEL 標準と Web サービス標準は、オープンで移植性が高く標準的な方法で一般的なアプリケーション要件に対応することにより、このジレンマを解決します。SOA は、既存のリソースを活用しながら、新しいアプリケーションのデプロイ・コストを最小限に抑えることで、ビジネス・パフォーマンスを最適化します。これらの標準およびアーキテクチャの手法を採用している企業では、Java/Java 2 Platform, Enterprise Edition (J2EE) の Web アプリケーションの構築に使用したのと同じ標準ベースの手法を継続型アプリケーションの構築に使用することで、著しい投資収益率 (ROI) を得ています。

Web サービスを機能させるには、次の 2 段階の処理が必要です。

1. サービスの公開。

サービスを公開するとは、既存のアプリケーションやシステムの機能を利用し、標準的な方法でサービスを使用可能にすることです。

2. サービスからのビジネス・フローの組立て (オーケストレーション)。

オーケストレーションとは、複数のサービスからエンドツーエンドのビジネス・プロセスを組み立てることです。

Web Services Description Language (WSDL)、eXtensible Markup Language (XML)、Simple Object Access Protocol (SOAP) などの Web サービス標準が、サービスを公開するための効率的で相互運用性の高いプラットフォームとして出現しています。さらに、パフォーマンスの高いバインディング・フレームワークにより、SOAP インタフェースでラップすることなくレガシー・システムやネイティブの Java コードにアクセスできます。

BPEL の概要

BPEL は、複数の同期および非同期サービスから相互に連携したトランザクション形式のプロセス・フローを構成するための明確な標準として現れました。BPEL には、前身となる言語である XLANG および WSFL を改良する、15 年にもわたる研究が活かされています。BPEL には次の機能があります。

- コンポーネント・モデルとしての Web サービス /WSDL
- データ・モデルとしての XML (データの疎結合)
- 同期および非同期のメッセージ交換パターン
- 決定的および非決定的なフロー調整
- 階層的な例外管理
- 長期間にわたる作業ユニット / 補正

BPEL の仕様は、2003 年 3 月に Organization for the Advancement of Structured Information Standards (OASIS) に提出されて以来、ほとんどすべての業界主要ベンダーから支持を得ています。この仕様により、企業には、従来では考えられないほどベンダー固有のルールから解放され、標準的で移植性の高い方法でビジネス・プロセスを実装できるといった大きな利点もたらされます。

Oracle BPEL Process Manager

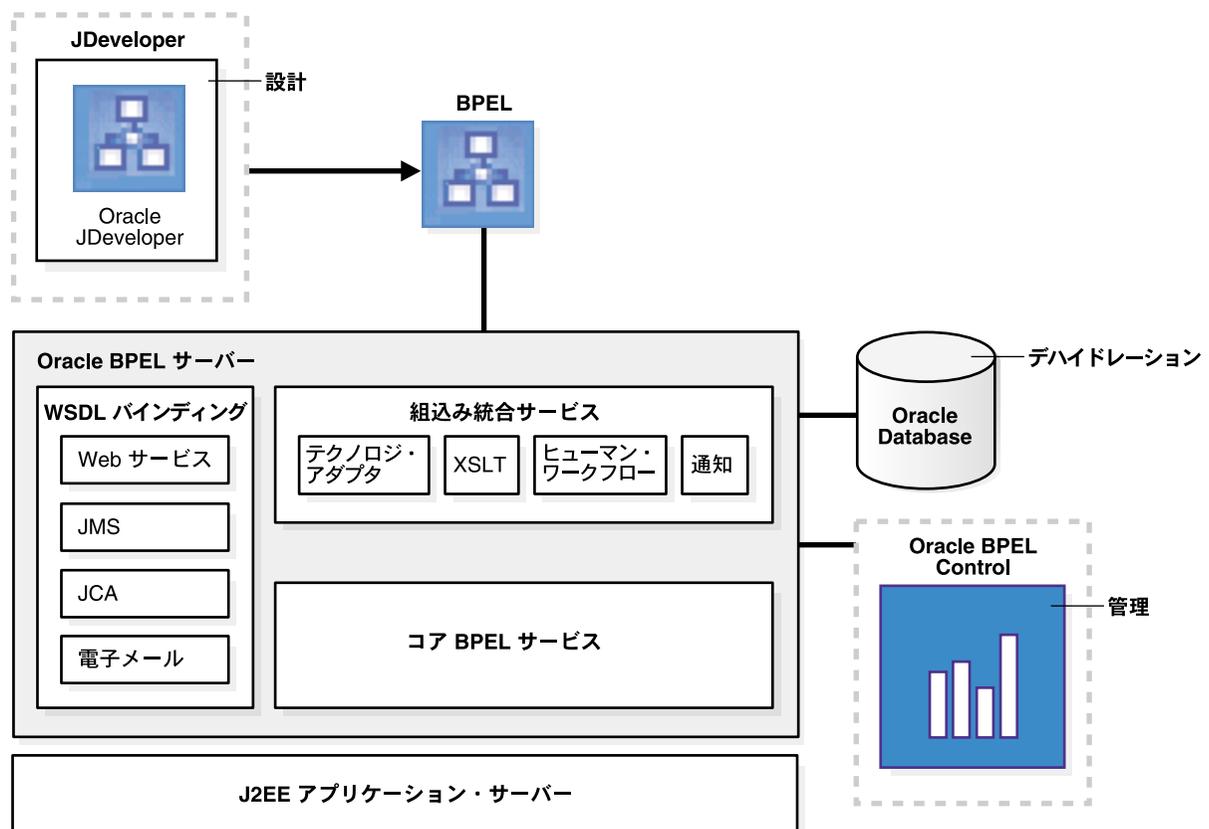
この項の内容は次のとおりです。

- Oracle BPEL Process Manager の概要
- Oracle BPEL Process Manager の特長
- Oracle BPEL Process Manager の機能の概要

Oracle BPEL Process Manager の概要

Oracle BPEL Process Manager は、BPEL ビジネス・プロセスを設計、デプロイおよび管理するための、ユーザー・フレンドリで信頼性の高いソリューションを提供します。図 1-1 に、Oracle BPEL Process Manager を構成する主要なコンポーネントを示します。

図 1-1 Oracle BPEL Process Manager



Oracle BPEL Process Manager は、Oracle JDeveloper にロードされます。これによって Oracle JDeveloper の機能が拡張され、BPEL を使用するビジネス・プロセスのモデル化、編集および設計が可能になります。Oracle JDeveloper では、グラフィカルかつユーザー・フレンドリな方法で BPEL プロセスを作成できます。Oracle JDeveloper の特徴は、BPEL をネイティブ・フォーマットとして使用することです。つまり、Oracle JDeveloper で作成されたプロセスは 100% 移植可能です。また、BPEL ソースの表示や変更を行う際にも Oracle JDeveloper はツールとしての便利さを損なうことなく使用できます。コア BPEL エンジンには、完成度、スケーラビリティおよび堅牢性の点で、現在入手可能な最も優れた BPEL サーバーの実装です。Oracle BPEL Process Manager は、標準の BPEL プロセスを実行します。また、Oracle BPEL Process Manager では長期間にわたるフローの状態をデータベース内に自動的に保存するデハイドレーション機能が提供されるため、フェイルオーバーとスケーラビリティの両方を目的としたクラスタ化が可能になります。Oracle BPEL Server は基礎となる J2EE アプリケーション・サーバーを使用し、ほとんどの主要なアプリケーション・サーバーの入手可能なバージョンをサポートします。

組込み統合サービスを使用すると、接続や変換などの高度な機能を標準 BPEL プロセスで使用できます。使用可能な機能には、XSLT および XQuery トランスフォーメーションや、Java コネクタ・アーキテクチャ (JCA) アダプタとネイティブ・プロトコルによる何百ものレガシー・システムへのバインディングのサポートが含まれます。ヒューマン・タスク・サービスが組込み BPEL サービスとして提供されており、人と手動タスクを BPEL フローに統合することができます。

拡張可能な WSDL バインディング・フレームワークにより、SOAP 以外のプロトコルやメッセージ形式への接続が可能になります。バインディングは、JMS、電子メール、JCA、HTTP GET、HTTP POST およびその他多数のプロトコルに対して利用できるため、何百ものバックエンド・システムに簡単に接続できます。

Oracle Enterprise Manager 10g BPEL Control では、成熟した Web ベースのインタフェースで、Oracle BPEL Server にデプロイされたプロセスの管理、処理、デバッグが実行できます。監査証跡とプロセスの履歴 / レポート情報は自動的に保持され、Oracle Enterprise Manager 10g BPEL Control と Java API の両方を使用して参照できます。

Oracle BPEL Process Manager の特長

Oracle BPEL Process Manager には、次の主要な特長があります。

- オープン・スタンダードの威力

ビジネス・プロセスを BPEL に取り込むことで、ベンダーへの依存を回避しつつ、知的財産や投資を保護します。BPEL とビジネス・プロセス管理の関係は、SQL とデータ管理の関係のようなものです。
- 比類のない可視性と管理

Oracle Enterprise Manager 10g BPEL Control により、ビジネス・プロセスをデプロイしたり、管理するためのコストや複雑さが軽減します。各 BPEL プロセスの実行の視覚的な監視、監査証跡へのドリルダウンによる各対話の詳細の確認、あるいは、BPEL 実装に対する実行フローのデバッグが可能です。
- オープンで柔軟性の高いバインディング・フレームワーク

XML Web サービスの他、Java/J2EE コンポーネント、ポータル、JCA インタフェース、Java Message Service (JMS) の送信先に対してもオーケストレーションを実行できます。また、バックエンド・システムと接続できます。Java のスキルやアプリケーション・サーバーへの投資を活用することもできます。

Oracle BPEL Process Manager の機能の概要

Oracle BPEL Process Manager には、次の機能があります。

- Oracle JDeveloper
 - BPEL のネイティブ・サポート
 - ドラッグアンドドロップ方式のプロセス・モデラー
 - Universal Description, Discover, and Integration (UDDI) および Web Services Inspection Language (WSIL) サービス・ブラウザ
 - ビジュアル XPath エディタ
 - ワンクリックのビルドとデプロイ
- Oracle Enterprise Manager 10g BPEL Control
 - ビジュアル・モニタリング
 - 監査
 - BPEL デバッグ
 - プロセスのバージョンニング
 - 実行中管理

- プロセス・パフォーマンス・モニタリング統計
- パーティション / ドメイン
- 組込み統合サービス
 - Java コードの埋込み
 - 電子メールおよび JMS メッセージング・サービス
 - eXtensible Stylesheet Language Transformation (XSLT) および XQuery 変換サービス
 - ワークフロー・タスクおよびポータル統合
 - 拡張可能な WSIF バインディング・フレームワーク
- Oracle BPEL Server
 - 包括的な BPEL バージョン 1.1
 - 同期および非同期メッセージング
 - コンテキストのデハイドレーション
 - 高度な例外管理
 - 並列バージョンニング
 - 巨大な XML ドキュメント

Oracle Application Server SOA Suite および Oracle JDeveloper のインストール

この項では、Oracle Application Server SOA Suite および Oracle JDeveloper をインストールして、Oracle BPEL Process Manager を Oracle JDeveloper で使用できるように構成する方法について説明します。

この項の内容は次のとおりです。

- [Oracle Application Server SOA Suite のインストール](#)
- [ポートの識別](#)
- [Oracle JDeveloper のインストール](#)
- [Oracle Application Server および統合サーバー接続のインストール](#)

注意： Oracle BPEL Process Manager for Developers インストール・タイプをインストールしてこのチュートリアルを使用する場合、ポート (9700)、Oracle BPEL Server の起動コマンドおよび Oracle Enterprise Manager 10g BPEL Control のユーザー名 (bpeladmin) がこのチュートリアルに示されているものと異なる場合があります。

Oracle Application Server SOA Suite のインストール

1. Oracle Application Server SOA Suite のインストール手順の詳細は、使用しているオペレーティング・システム用の『Oracle Application Server インストール・ガイド 10g リリース 3 (10.1.3.1)』を参照してください。
2. 「Oracle Application Server SOA Suite 10.1.3.1.0 インストール」ウィンドウの「**基本インストール**」を選択します。

注意： Oracle BPEL Process Manager のサンプルを使用する場合、**oc4jadmin** ユーザーに対して「Oracle Application Server SOA Suite 10.1.3.1.0 インストール」ウィンドウの「**AS 管理者のパスワード**」フィールドに、「**welcome1**」と入力してください。異なるパスワードを入力すると、サンプルによっては正常に動作させるために変更が必要になる場合があります。

3. 「インストール」をクリックします。
4. インストールが完了したら「**終了**」をクリックします。

ポートの識別

インストール時に自動構成されたいくつかのポート番号を識別します。これらのポート番号は、アプリケーション・サーバーおよび Oracle BPEL Server への統合サーバー接続を作成するときに使用します。

1. テキスト・エディタを使用して
`SOA_Oracle_Home\bpel\utilities\ant-orabpel.properties` ファイルを開きます。
2. 次のプロパティで使用されているポート番号をメモします。
 - `http.port`
 - `opmn.requestport`

これらのポート値は、後で 1-7 ページの「[Oracle Application Server および統合サーバー接続のインストール](#)」で接続を作成するときに必要です。

Oracle JDeveloper のインストール

Oracle JDeveloper 10.1.3.1 Studio を個別にインストールする必要があります。

1. Oracle JDeveloper を、ホスト上に解凍します。
2. `JDev_Oracle_Home\JDev\bin\jdev.exe` をクリックするか、ショートカットを作成して Oracle JDeveloper を起動します。

注意： Oracle JDeveloper は、空白を含むディレクトリ・パス（たとえば `C:\Program Files\JDev`）にインストールしないでください。インストールすると、BPEL プロセスのコンパイル時にエラーになります。

Oracle Application Server および統合サーバー接続のインストール

1. Oracle JDeveloper で、「表示」メイン・メニューから「接続ナビゲータ」を選択します。
2. 「アプリケーション・サーバー」を右クリックします。
3. 「アプリケーション・サーバー接続の作成」を選択します。
4. 「ようこそ」ページで「次へ」をクリックします。
5. アプリケーション・サーバーへ接続するための、わかりやすい名前を指定します。この名前は、1-8 ページの手順 19 で統合サーバー接続を作成するときを選択するので、覚えておきます。
6. 「接続タイプ」リストから、「Oracle Application Server 10g 10.1.3」を選択します。
7. 「次へ」をクリックします。
8. 次の情報を入力します。

フィールド	値
ユーザー名	oc4jadmin
パスワード	1-6 ページの「Oracle Application Server SOA Suite のインストール」で入力したものと同一管理者パスワードを入力します。
パスワードを配布	このチェック・ボックスを選択します。選択しないと、プロセスをデプロイするたびに認証が必要となります。

9. 「次へ」をクリックします。
10. 次の情報を入力します。

フィールド	値
ホスト名	localhost
OPMN ポート	1-6 ページの「ポートの識別」の手順 2 で識別した opmn.requestport プロパティで使用されているものと同じポート値を入力します。これには、ポート番号 6003 がよく使用されます。
グループ/OC4J インスタンス名	home
シングル・インスタンス	このラジオ・ボタンを選択します。
グループ	このラジオ・ボタンは選択しないままにします。

11. 「次へ」をクリックします。
12. 「接続のテスト」をクリックして接続をテストします。接続に成功した場合は、次のメッセージが表示されます。
成功
13. 「終了」をクリックします。
14. 「接続ナビゲータ」の「統合サーバー」を右クリックします。
15. 「新規統合サーバー接続」を選択します。
BPEL プロセスをすべてデプロイする場合に、この統合サーバー接続を使用します。
16. 「ようこそ」ページで「次へ」をクリックします。
17. サーバーへ接続するための、わかりやすい名前を指定します。
18. 「次へ」をクリックします。

19. 1-7 ページの手順 5 で指定したアプリケーション・サーバー接続名を、「アプリケーション・サーバー」リストから選択します。

20. 次の情報を入力します。

フィールド	値
ホスト名	localhost
ポート	1-6 ページの「 ポートの識別 」の手順 2 で識別した http.port プロパティで使用されるものと同じポート値を入力します。これには、ポート番号 8888 がよく使用されます。
プロキシ例外リストにホスト名を追加	このチェック・ボックスを選択して、このホストが「ツール」→「設定」→「Web ブラウザとプロキシ」の下の「例外」フィールドに示されたプロキシ例外のリストに自動的に追加されるようにします。Oracle JDeveloper の最初の起動時に、使用ブラウザに対して設定されたこれらのプロキシ例外がデフォルトで使用されます。

21. 「次へ」をクリックします。

22. 「接続のテスト」をクリックして接続をテストします。接続に成功した場合は、次のメッセージが表示されます。

アプリケーション・サーバー : OK
 BPEL Process Manager サーバー : OK
 ESB サーバー : OK

23. 「終了」をクリックします。

Oracle BPEL Process Manager コンポーネントの起動および停止

表 1-1 の手順に従って、Oracle BPEL Process Manager コンポーネントを起動および停止します。

表 1-1 Oracle BPEL Process Manager コンポーネントの起動および停止

アクセス対象	Windows の場合	UNIX の場合
Oracle BPEL Server	「スタート」→「すべてのプログラム」→「Oracle - Oracle_Home」→「Start SOA suite」を選択します。	Oracle BPEL Server を起動するには、次のようにします。 <code>\$ORACLE_HOME/bpel/bin</code> から、次のスクリプトを実行します。 <code>startorabpel.sh</code> Oracle BPEL Server を停止するには、次のようにします。 <code>\$ORACLE_HOME/bpel/bin</code> から、次のスクリプトを実行します。 <code>shutdownorabpel.sh</code>
Oracle JDeveloper	1-6 ページの「 Oracle JDeveloper のインストール 」の手順 2 を参照してください。	<code>\$ORACLE_HOME/jdev/bin/jdev</code>

表 1-1 Oracle BPEL Process Manager コンポーネントの起動および停止 (続き)

アクセス対象	Windows の場合	UNIX の場合
Oracle Enterprise Manager 10g BPEL Control	<p>最初に Oracle BPEL Server を起動する必要があります。</p> <p>Oracle Enterprise Manager 10g BPEL Control を起動するには、次のようにします。</p> <ol style="list-style-type: none"> 1. 「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「BPEL Control」 を選択します。 <p>または、使用しているインストールの URL で Oracle Enterprise Manager 10g BPEL Control を起動することもできます。この URL は、<code>SOA_Oracle_Home¥install¥bpelsetupinfo.txt</code> にあります。</p>	<p>最初に Oracle BPEL Server を起動します。</p> <p>Oracle Enterprise Manager 10g BPEL Control を起動するには、次のようにします。</p> <ul style="list-style-type: none"> ▪ <code>bpelsetupinfo.txt</code> に記載されているインストール用の URL にログオンします。
Developer Prompt	<p>「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「Developer Prompt」 を選択して、コマンド・プロンプトを <code>SOA_Oracle_Home¥bpel¥samples</code> ディレクトリで開きます。このプロンプトを使用すると、デモンストレーションへのアクセスや、必要な Web サービスの起動が簡単にできます。</p>	<p>次のようにして、Bourne シェルで Developer Prompt を設定します。</p> <pre>\$ ORACLE_HOME=/home/oracle/installs/midtier \$ export ORACLE_HOME \$ PATH=\$ORACLE_HOME/bpel/bin:\$PATH \$ export PATH</pre>
Oracle BPEL Process Manager のサンプルおよびチュートリアル	<p>BPEL のサンプルおよび追加チュートリアルの詳細を使用するには、次のようにします。</p> <p>「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「Getting Started with Samples」 を選択します。</p>	<p>次の URL にログインします。</p> <pre>\$ORACLE_HOME/bpel/samples/sampleshome.html</pre>
Worklist Application のサンプル	<p>Oracle BPEL Worklist Application のログイン・ウィンドウにアクセスするには、次のようにします。</p> <p>「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「Sample Worklist Application」 を選択します。</p> <p>または、使用しているインストールの URL で Oracle BPEL Worklist Application を起動することもできます。この URL は、<code>SOA_Oracle_Home¥install¥bpelsetupinfo.txt</code> にあります。</p>	<p>最初に Oracle BPEL Server を起動します。</p> <p>Oracle BPEL Worklist Application を起動するには、次のようにします。</p> <ul style="list-style-type: none"> ▪ <code>bpelsetupinfo.txt</code> に記載されているインストール用の URL にログオンします。

注意： このマニュアルでオペレーティング・システムのコマンド・プロンプトを開くには、常に「Developer Prompt」を使用してください。これで、必要なパスがすべて設定されます。これ以外の方法でオペレーティング・システムのコマンド・プロンプトを開くことはサポートされていません。

Web ブラウザ設定でのホスト名の確認

使用しているコンピュータのホスト名は、インストール時に自動的に Oracle JDeveloper のプリファレンス設定に追加されます。ホスト名がプリファレンス設定に表示されない場合、パートナー・リンクの作成時に「サービス・エクスプローラ」ウィンドウで WSDL ファイルを選択すると、解析エラーが表示されることがあります。

1. 「ツール」メイン・メニューから「設定」を選択します。
2. 「Web ブラウザとプロキシ」をクリックします。
3. 「例外」フィールドにホスト名が表示されていることを確認します。たとえば、ホスト名が myhost-pc の場合、次のように入力します。

```
us.acme.com|*.us.acme.com|127.0.0.1|myhost-pc|localhost
```

4. 「例外」フィールドに localhost が表示されていることも確認します。
5. 「OK」をクリックします。

これで、BPEL プロセスを設計する準備ができました。

ネームスペースとネームスペース URI について

Oracle JDeveloper を使用すると、**ns1:**、**ns2:**、**client:** などのネームスペース値を、**Assign** アクティビティでコピー操作を作成するときの「式」フィールドや「変数」ナビゲーション・ツリー、あるいはその他のアクティビティのフィールドに表示できます。

Oracle BPEL Process Manager では、ネームスペースとの相互作用の削減を試行します。ネームスペースが「式」フィールドや「変数」ナビゲーション・ツリーに表示される場合は、問合せや式の一部を入力する際に表示されるネームスペース接頭辞を使用します。

XML ネームスペースは、重複する要素タイプおよび属性名を区別する方法です。このような重複は、たとえば、2つの異なる DTD の要素タイプおよび属性が含まれる XSLT スタイル・シートやドキュメントなどで発生します。<refuse> 要素が、融資業者用の <refuse> とはまったく異なる清掃業者用のものを指す場合があります。そのため、これらの要素は、**ns1:**、**ns2:** または **client:** などの異なるネームスペース接頭辞で区別されます。

URI は、一意の識別子を作成する方式です。

信用フローのチュートリアル

このチュートリアルでは、Oracle JDeveloper を使用して、初めての BPEL プロセスを設計、デプロイおよびテストする方法について説明します。

このチュートリアルの内容は次のとおりです。

- [信用フローの概要](#)
- [チュートリアルの使用方法](#)

信用フローの概要

このチュートリアルでは、Oracle JDeveloper を使用して、初めての BPEL プロセスを作成、デプロイおよびテストする方法について説明します。このプロセスは、信用格付けサービスをコールするように作成するフローです。このプロセスを実行する際には、社会保障番号を HTML ユーザー・インタフェースに入力します。信用格付けサービスは、その社会保障番号を受け取り、信用格付けを返します。このプロセスは、第 3 章「融資調達 BPEL プロセスの確認」で示すような、さらに高度なアプリケーションを作成するための第 1 歩として作成します。

チュートリアルの使用方法

このチュートリアルの内容は次のとおりです。

- Oracle BPEL Server と Oracle JDeveloper の起動
- サービスの起動とテスト
- アプリケーションとプロジェクトの作成
- WSDL ファイルのソース・コードの確認
- BPEL ファイルのソース・コードの確認
- XSD ファイルの編集
- 設計フェーズにおけるプロジェクトの検証
- 信用格付けサービス用のパートナ・リンクの作成と構成
- 信用フロー・プロセスの検証、コンパイルおよびデプロイ
- 信用フロー・プロセスの実行

Oracle BPEL Server と Oracle JDeveloper の起動

Oracle JDeveloper と Oracle BPEL Server が起動していることを確認します。手順は、1-8 ページの「Oracle BPEL Process Manager コンポーネントの起動および停止」を参照してください。

サービスの起動とテスト

このチュートリアル中に、設計する BPEL プロセスと 2-2 ページの「信用フローの概要」で説明した信用格付けサービスとの通信が行われます。まず、このサービスを起動し、動作することをテストする必要があります。

1. 「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「Developer Prompt」を選択して、コマンド・プロンプトを `SOA_Oracle_Home\bpel\samples` ディレクトリで開きます。

2. `utils\CreditRatingService` サブディレクトリに移動します。

```
cd utils\CreditRatingService
```

3. 次のコマンドを入力します。

```
ant
```

これにより、このチュートリアルで使用するために、信用格付けサービスがデプロイおよび起動されます。正常に行われた場合は、次のようなメッセージが最後に表示されます。

```
BUILD SUCCESSFUL
```

4. 「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「BPEL Control」を選択して Oracle Enterprise Manager 10g BPEL Control にログインします。

5. Oracle Enterprise Manager 10g BPEL Control にログインするための次の情報を入力して「ログイン」をクリックします。

フィールド	値
ユーザー名	oc4jadmin
パスワード	password

`password` は、インストール時に 1-6 ページの「Oracle Application Server SOA Suite のインストール」で入力した値です。

CreditRatingService、**TaskActionHandler** および **TaskManager** の各サービスが「ダッシュボード」タブに表示されていれば、サービスはすべて稼働しています。

アプリケーションとプロジェクトの作成

はじめに、アプリケーションとプロジェクトを作成する必要があります。BPEL プロセスの基本的な要素は、プロジェクトによって自動的に作成されます。これには、次のものが含まれます。

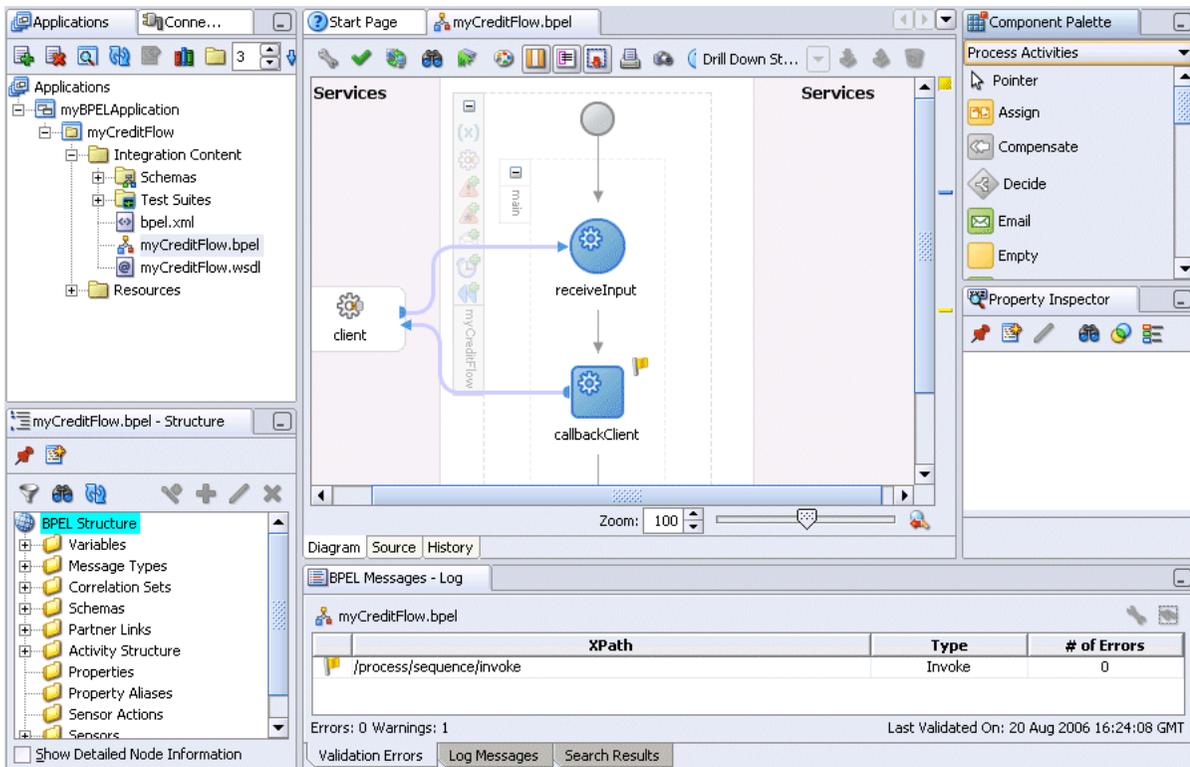
- BPEL プロセスのソース (`projectname.bpel`)
- Web Services Description Language (WSDL) クライアント・インタフェース (`projectname.wsdl`)
- BPEL プロセスのデプロイメント・ディスクリプタ (`bpel.xml`)

次の手順に従って、信用フロー・プロジェクトを新規作成します。

注意： プロジェクト名に特殊文字（ピリオドなど）を含めないでください。特殊文字を含めると、プロジェクトのコンパイル時にエラーが表示されます。

1. Oracle JDeveloper に戻ります。
2. メイン・メニューから「ファイル」→「新規」を選択します。
3. 「項目」ウィンドウで「アプリケーション」をダブルクリックして、「アプリケーションの作成」ウィンドウを表示します。
4. 「アプリケーション名」フィールドに **myBPELApplication** と入力し、「ディレクトリ名」フィールドではデフォルト・パスを使用します。
5. 「OK」をクリックします。
6. 「プロジェクトの作成」ウィンドウで「取消」をクリックします。
7. 「アプリケーション・ナビゲータ」セクションで「**myBPELApplication**」を右クリックします。
8. 「新規プロジェクト」を選択して、新しい BPEL プロセス・プロジェクトを定義します。
9. 「項目」ウィンドウで「BPEL プロセス・プロジェクト」をダブルクリックして、「BPEL プロジェクト作成ウィザード」ウィンドウを表示します。このエントリが表示されない場合は、最初に「一般」→「プロジェクト」を選択します。
10. 「名前」フィールドに **myCreditFlow** と入力します。その他すべてのフィールドは、デフォルトで非同期 BPEL プロセスを作成するための確な値になります。
11. 「終了」をクリックします。BPEL プロセス・ファイルが、`JDev_Oracle_Home\jdev\mywork\myBPELApplication\myCreditFlow\bpel` ディレクトリに作成されます。

12. 次のセクションが表示されます。これらのセクションが表示されない場合は、Oracle JDeveloper で「**ダイアグラム**」をクリックして、「**アプリケーション・ナビゲータ**」→「**インテグレーション・コンテンツ**」セクションで「**myCreditFlow.bpel**」をクリックします。Oracle JDeveloper の各セクションの詳細は、Oracle BPEL Process Manager 開発者ガイドを参照してください。



WSDL ファイルのソース・コードの確認

ここでは、WSDL ファイルのセクションを確認します。

- 「**アプリケーション・ナビゲータ**」セクションで「**myCreditFlow.wsdl**」をダブルクリックした後、ウィンドウ下部の「**ソース**」をクリックします。
- 次の部分のコードを確認します。2つのポート・タイプが、それぞれ一方の操作で定義されています。一方の操作は、非同期プロセスを初期化します。他方の操作は、非同期レスポンスでクライアントへコールバックします。

```
<portType name="myCreditFlow">
  <operation name="initiate">
    <input message="client:myCreditFlowRequestMessage"/>
  </operation>
</portType>

<!-- portType implemented by the requester of myCreditFlow BPEL process
for asynchronous callback purposes
-->
<portType name="myCreditFlowCallback">
  <operation name="onResult">
    <input message="client:myCreditFlowResponseMessage"/>
  </operation>
</portType>
```

3. 次の部分のコードを確認します。この非同期プロセスの `partnerLinkType` には、2つのロールがあります。一方のロールはサービス・プロバイダ用、他方のロールはリクエスト元用です。

```
<plnk:partnerLinkType name="myCreditFlow">
  <plnk:role name="myCreditFlowProvider">
    <plnk:portType name="client:myCreditFlow"/>
  </plnk:role>
  <plnk:role name="myCreditFlowRequester">
    <plnk:portType name="client:myCreditFlowCallback"/>
  </plnk:role>
</plnk:partnerLinkType>
```

BPEL ファイルのソース・コードの確認

ここでは、BPEL ファイルのセクションを確認します。

1. 「**myCreditFlow.bpel**」をダブルクリックします。
2. Designer ウィンドウ下部の「**ソース**」をクリックします。
3. 次の部分のコードを確認します。クライアント・インタフェース用に作成された `partnerLink` には、`myRole` と `partnerRole` の2つのロールがあります。通常、非同期 BPEL プロセスには、クライアント・インタフェース用のロールが2つあります。一方はフロー自体のためのロールで、入力操作を公開します。他方はクライアント用のロールで、非同期にコールバックされます。

```
<partnerLinks>
  <partnerLink name="client" partnerLinkType="client:myCreditFlow"
    myRole="myCreditFlowProvider" partnerRole="myCreditFlowRequester"/>
</partnerLinks>
```

4. 次の部分のコードを確認します。プロセスの本体の `<receive>` アクティビティに続いて、リクエスト元への非同期コールバックを実行する `<invoke>` アクティビティが指定されています。非同期プロセスと同期プロセスの違いに注意してください。同期プロセスは、`<reply>` アクティビティを使用して、コール元へ同期的に応答します。

```
<sequence name="main">

  <!-- Receive input from requestor.
  Note: This maps to operation defined in myCreditFlow.wsdl
  -->
  <receive name="receiveInput" partnerLink="client"
    portType="client:myCreditFlow" operation="initiate" variable="inputVariable"
    createInstance="yes"/>

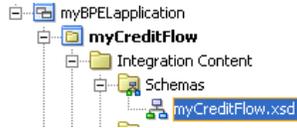
  <!-- Asynchronous callback to the requester.
  Note: the callback location and correlation id is transparently handled
  using WS-addressing.
  -->
  <invoke name="callbackClient" partnerLink="client"
    portType="client:myCreditFlowCallback" operation="onResult"
    inputVariable="outputVariable"/>
</sequence>
</process>
```

5. 「**ダイアグラム**」をクリックします。これで、BPEL プロセスを設計する準備ができました。

XSD ファイルの編集

BPEL プロジェクトを作成したときに、デフォルトの XSD ファイルが作成されています。

1. 「アプリケーション・ナビゲータ」で、「myCreditFlow」 → 「インテグレーション・コンテンツ」 → 「スキーマ」の順に開きます。
2. 「myCreditFlow.xsd」をダブルクリックします。



3. 「ソース」をクリックします。

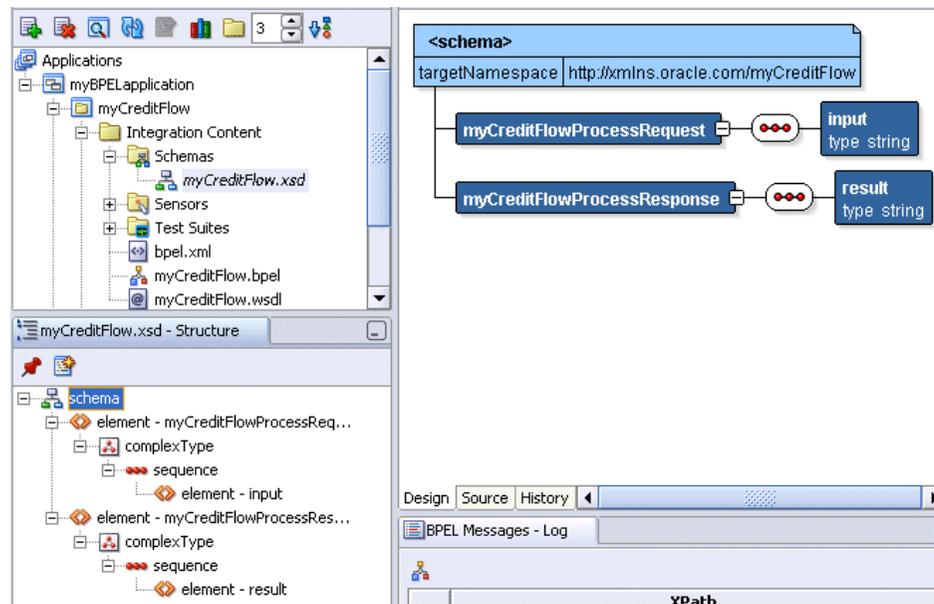
myCreditFlowProcessRequest 要素が入力に使用されます。
myCreditFlowProcessResponse 要素が出力として返されます。

```
<schema attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://xmlns.oracle.com/myCreditFlow"
  xmlns="http://www.w3.org/2001/XMLSchema">
  <element name="myCreditFlowProcessRequest">
    <complexType>
      <sequence>
        <element name="input" type="string"/>
      </sequence>
    </complexType>
  </element>
  <element name="myCreditFlowProcessResponse">
    <complexType>
      <sequence>
        <element name="result" type="string"/>
      </sequence>
    </complexType>
  </element>
</schema>
```

ここでは、WSDL ファイルの入出力メッセージを編集します。この WSDL ファイルは、2-15 ページの「信用フロー・プロセスの実行」で BPEL プロセスを実行し、信用格付けをリクエストするときの対話に使用するユーザー・インタフェースに相当します。

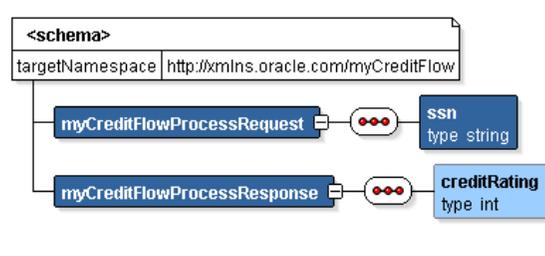
4. 「設計」をクリックします。

XSD ファイルのグラフィカル・ビューが「設計」ウィンドウに表示されます。左下の「構造」セクションに、完全に開いた状態で編集できるように XSD ファイルの要素が表示されます。



5. 「構造」ウィンドウの最初の「要素 - input」をダブルクリックして、「elementのプロパティ」ウィンドウを表示します。
6. **input** を **ssn** に変更します。
7. 「OK」をクリックします。
8. 「構造」ウィンドウの2つ目の「要素 - result」をダブルクリックして、「elementのプロパティ」ウィンドウを表示します。
9. **result** を **creditRating** に変更します。
10. 「アドバンスド・プロパティ」タブをクリックします。
11. 「type」セクションに移動して、「string」と表示されていることを確認します。
12. 「string」をクリックします。
13. 「string」の右にある「...」をクリックして「type」ウィンドウを表示します。
14. 「名前」フィールドで、出力として返す「string」を「int」に置き換えます。
15. 「OK」をクリックして、「type」ウィンドウと「elementのプロパティ」ウィンドウを閉じます。

変更内容が XSD ファイルの設計ビューに表示されます。



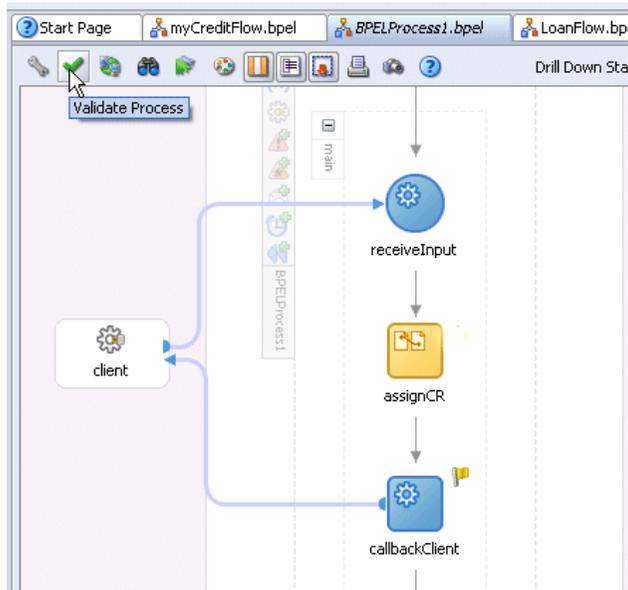
16. 「ファイル」メイン・メニューから「保存」を選択します。
17. Designer ウィンドウ上部の「myCreditFlow.xsd」の「x」をクリックして、XSD ファイルを閉じます。
18. 「アプリケーション・ナビゲータ」セクションで「myCreditFlow.bpel」をダブルクリックします。
19. 「ダイアグラム」をクリックします。

設計フェーズにおけるプロジェクトの検証

BPEL プロジェクトの設計時に、黄色の小さなフラグがアクティビティの上側に表示される場合があります。これらのアクティビティは、正常に構成しても検証が済んでいない場合があります。

1. プロセスを検証するには、Designer ウィンドウの上部の「検証」をクリックします。

図 2-1 プロセスの検証



アクティビティが正しく検証されると、フラグの表示がなくなります。

2. 黄色のフラグが残っている場合は、アクティビティに検証エラーがあるか、プロセスの設計が未完成の可能性があります。
3. 検証エラーの詳細と解決方法を表示するには、アクティビティの上にある黄色のフラグをクリックします。

注意： プロセスをデプロイするとき、2-14 ページの「信用フロー・プロセスの検証、コンパイルおよびデプロイ」に説明されているように、プロセスが自動的に検証、コンパイルされ、正常に終了した場合にはデプロイも実行されます。

信用格付けサービス用のパートナ・リンクの作成と構成

ここでは、同期型の信用格付けサービス用のパートナ・リンクを作成および構成します。

この項では、次のタスクについて説明します。

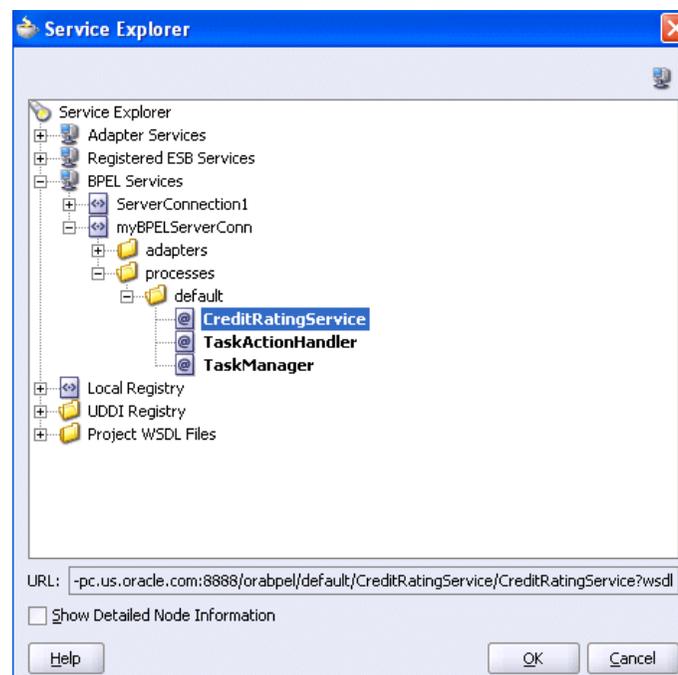
- 信用格付けサービスのパートナ・リンクの作成
- Scope アクティビティの作成
- Scope アクティビティ内での Invoke アクティビティの作成
- Scope アクティビティ内での 1 つ目の Assign アクティビティの作成
- Scope アクティビティ内での 2 つ目の Assign アクティビティの作成

信用格付けサービスのパートナ・リンクの作成

注意： パートナ・リンクは、BPEL プロセスと対話する外部サービスを定義します。信用格付けサービス用にパートナ・リンクを作成する必要があります。

1. Oracle JDeveloper の右上にある「コンポーネント・パレット」セクションのドロップダウン・リストで、「サービス」が選択されていることを確認します。
2. 「サービス」セクションの下にある **PartnerLink** アクティビティを、Oracle JDeveloper の右側にドラッグ・アンド・ドロップします。
「パートナ・リンクの作成」ウィンドウが表示されます。
3. 次の値を入力して、信用格付けサービス用のパートナ・リンクを作成します。

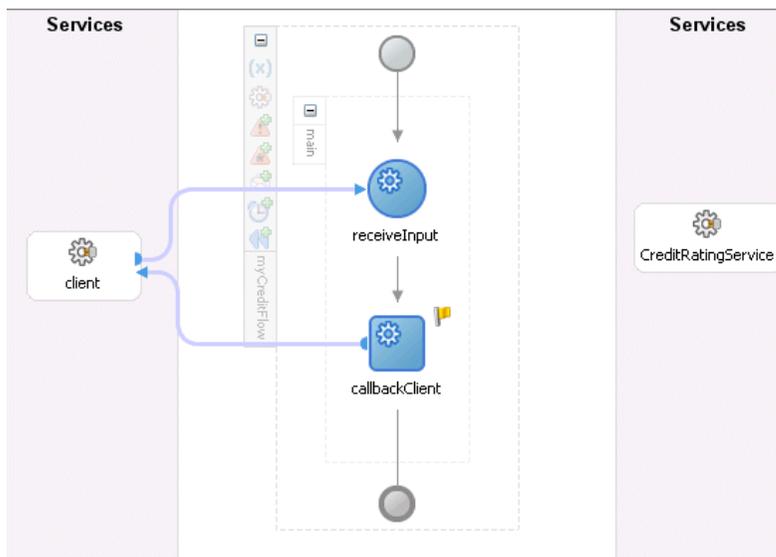
注意： 下側の「WSDL ファイル」フィールドでは、2-2 ページの「サービスの起動とテスト」でデプロイした信用格付けサービスを自動的に選択するために、懐中電灯（左から 2 番目の「サービス・エクスプローラ」という名前のアイコン）をクリックして、次の図に示す「サービス・エクスプローラ」ウィンドウにアクセスします。



フィールド	値
名前	CreditRatingService
WSDL ファイル	<p>この URL にアクセスするには、「サービス・エクスプローラ」の懐中電灯アイコンをクリックして BPEL Services → 「<i>my_integration_server_connection</i>」 → 「プロセス」 → 「デフォルト」 → 「CreditRatingService」の順に開いて選択します。</p> <p><i>my_integration_server_connection</i> は、1-7 ページの手順 17 で指定したサーバー接続名です。</p> <p>http://localhost:8888/orapel/default/CreditRatingService/CreditRatingService?wsdl</p> <p>関連項目: 「サービス・エクスプローラ」 ウィンドウで WSDL ファイルを追加しようとして、解析エラーが表示された場合は、1-10 ページの「Web ブラウザ設定でのホスト名の確認」を参照してください。</p>
パートナ・リンク・タイプ	CreditRatingService
パートナ・ロール	CreditRatingServiceProvider
マイ・ロール	未指定のままにします。これは同期パートナ・リンクであり、ロールは必要ないためです。

4. 「OK」をクリックします。

「CreditRatingService」パートナ・リンクが設計ウィンドウの右側に表示されます。



5. 「ファイル」メイン・メニューから「保存」を選択します。

Scope アクティビティの作成

概要: この項では、まず、**Scope** アクティビティを作成します。**Scope** アクティビティは、独自のローカル変数、フォルト・ハンドラなどを保持できるアクティビティの集まりで構成されています。**Scope** アクティビティは、プログラミング言語での { } ブロックに似ています。

この **Scope** アクティビティ内で、クライアントの信用格付けのバックグラウンドが識別されます。

1. 「コンポーネント・パレット」セクションに移動して「プロセス・アクティビティ」を選択します。
2. **Scope** アクティビティを、**receiveInput** アクティビティと **callbackClient** アクティビティの間にドラッグ・アンド・ドロップします。
3. この **Scope** アクティビティをダブルクリックして、「Scope」ウィンドウを表示します。
4. 「一般」タブの「名前」フィールドに、**GetCreditRating** と入力します。
5. 「OK」をクリックします。
6. 「ファイル」メイン・メニューから「保存」を選択します。

Scope アクティビティ内での Invoke アクティビティの作成

概要: この項では、**Invoke** アクティビティを作成します。**Invoke** アクティビティを使用すると、(パートナ・リンクによって識別された) サービスを起動するための操作を指定できます。

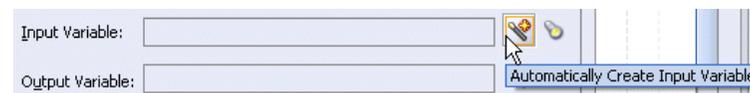
この **Invoke** アクティビティは、クライアントと信用格付けサービスとの間の最初の相互作用の操作となります。

1. 「+」記号をクリックして、**GetCreditRating Scope** アクティビティを開きます。
2. **Invoke** アクティビティを、「コンポーネント・パレット」セクションから **GetCreditRating Scope** アクティビティ内にドラッグ・アンド・ドロップします。
3. この **Invoke** アイコンをダブルクリックして、「起動」ウィンドウを表示します。
4. 次の情報を入力します。

フィールド	値
名前	invokeCRS
パートナ・リンク	CreditRatingService

「操作」(process) フィールドは自動的に入力されます。

5. 「入力変数」フィールドの右側にある最初のアイコンをクリックします。これは、変数自動作成アイコンです。



6. 表示される「変数の作成」ウィンドウで「OK」をクリックします。

invokeCRS_process_InputVariable という名前の変数が「入力変数」フィールドに自動的に作成されます。この変数には、メッセージ・タイプの **CreditRatingServiceRequestMessage** が自動的に割り当てられます。

7. 「出力変数」フィールドの右側にある最初のアイコンをクリックします。

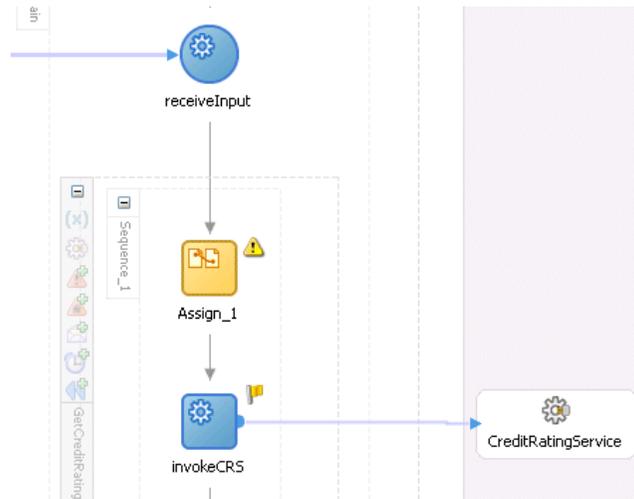
8. 表示される「変数の作成」ウィンドウで「OK」をクリックします。
invokeCRS_outputVariable という名前の変数が「出力変数」フィールドに自動的に作成されます。この変数には、メッセージ・タイプの **CreditRatingServiceResponseMessage** が自動的に割り当てられます。
9. 「OK」をクリックします。
10. 「ファイル」メイン・メニューから「保存」を選択します。

Scope アクティビティ内での 1 つ目の Assign アクティビティの作成

概要： この項では、このチュートリアル の 2 つの **Assign** アクティビティの 1 つ目を作成します。**Assign** アクティビティは、ある変数の内容を別の変数にコピーするなど、単純なデータ操作を行うための方法を提供します。

この **Assign** アクティビティは、クライアントの社会保障番号を入力として取り込み、信用格付けを割り当てるために信用格付けサービスにコンタクトします。

1. **Assign** アクティビティを、「コンポーネント・パレット」セクションから **invokeCRS Invoke** アクティビティの上側にドラッグ・アンド・ドロップします。



2. この **Assign** アイコンをダブルクリックして、「割当て」ウィンドウを表示します。
3. 「一般」タブの「名前」フィールドに、**assignSSN** と入力します。
4. 「適用」をクリックします。
5. 「コピー操作」タブをクリックします。
6. 「作成」をクリックし「コピー操作」を選択して、「コピー操作の作成」ウィンドウを表示します。

7. 次の値を入力します。

フィールド	値
送信元	
■ タイプ	変数
■ 変数	「変数」 → 「inputVariable」 → 「ペイロード」 → 「client:myCreditFlowProcessRequest」 → 「client:ssn」の順に開いて選択します。 注意: ネームスペース番号の値 (ns1、ns2 など) は変わります。自動的に表示されるネームスペース値を使用してください。
宛先	
■ タイプ	変数
■ 変数	「変数」 → 「invokeCRS_process_InputVariable」 → 「payload」 → 「ns1:ssn」の順に開いて選択します。

8. 「OK」をクリックして、「コピー操作」ウィンドウと「割当て」ウィンドウを閉じます。
9. 「ファイル」メイン・メニューから「保存」を選択します。

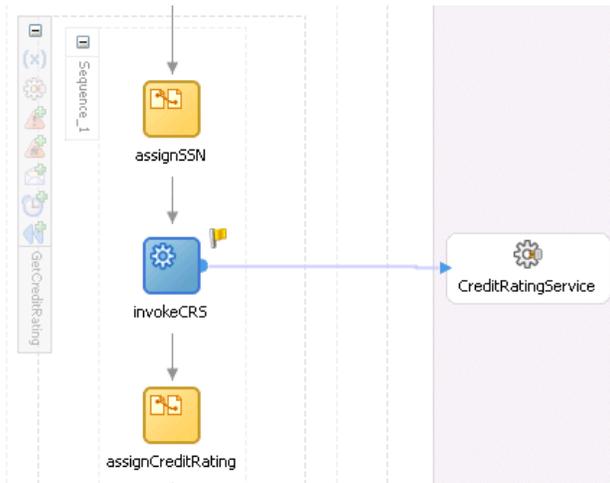
Scope アクティビティ内での2つ目の Assign アクティビティの作成

注意: この Assign アクティビティは、クライアントに割り当てられた信用等级付けに関する詳細情報を返します。

- 2つ目の Assign アクティビティを、「コンポーネント・パレット」セクションから **GetCreditRating Scope** アクティビティ内の **invokeCRS Invoke** アクティビティの下にドラッグ・アンド・ドロップします。
- この Assign アイコンをダブルクリックして、「割当て」ウィンドウを表示します。
- 「一般」タブの「名前」フィールドに、**assignCreditRating** と入力します。
- 「適用」をクリックします。
- 「コピー操作」タブをクリックします。
- 「作成」をクリックし「コピー操作」を選択して、「コピー操作の作成」ウィンドウを表示します。
- 次の値を入力します。

フィールド	値
送信元	
■ タイプ	変数
■ 変数	「変数」 → 「invokeCRS_process_OutputVariable」 → 「ペイロード」 → 「ns1:rating」の順に開いて選択します。 注意: ネームスペース番号の値 (ns1、ns2 など) は変わります。自動的に表示されるネームスペース値を使用してください。
宛先	
■ タイプ	変数
■ 変数	「変数」 → 「outputVariable」 → 「payload」 → 「client:myCreditFlowProcessResponse」 → 「client:creditRating」の順に開いて選択します。

8. 「OK」をクリックして、「コピー操作」ウィンドウと「割当て」ウィンドウを閉じます。
この **Scope** アクティビティは次のようになります。



9. 「ファイル」メイン・メニューから「保存」を選択します。
これで、信用格付けサービスの設計が完了し、次の項で説明する検証、コンパイルおよびデプロイの各タスクに進む準備ができました。

信用フロー・プロセスの検証、コンパイルおよびデプロイ

この時点で、BPEL プロセスをデプロイする準備ができています。

1. 「アプリケーション・ナビゲータ」セクションに移動します。
2. 「myCreditFlow」を右クリックします。
3. 「デプロイ」→「my_integration_server_connection」→「default ドメインにデプロイ」の順に選択します。

my_integration_server_connection は、1-7 ページの手順 17 で指定した統合サーバー接続名です。

4. 「OK」をクリックします。
これにより、BPEL プロセスがコンパイルされます。エラーがないかウィンドウの下部を確認します。エラーがない場合は、次のメッセージが表示されます。

```
BUILD SUCCESSFUL
```

5. ウィンドウ下部のボタンをクリックしてエラーの有無の確認します。「メッセージ」ボタンで表示されないエラーを、「Apache Ant」ボタンで表示できる場合があります。
エラーがない場合、デプロイメントは成功しました。エラーがある場合は、そのエラーをダブルクリックしてエラーのタイプと場所の詳細を表示します。
6. 訂正して、再度デプロイします。

信用フロー・プロセスの実行

この時点で、融資の申込みと融資提案の受取りのプロセスを起動する準備ができています。

1. 「スタート」→「すべてのプログラム」→「Oracle - Oracle_Home」→「Oracle BPEL Process Manager」→「BPEL Control」を選択して Oracle Enterprise Manager 10g BPEL Control にログインします。
2. 次の情報を入力して Oracle Enterprise Manager 10g BPEL Control にログインします。

フィールド	値
ユーザー名	oc4jadmin
パスワード	password

Oracle Enterprise Manager 10g BPEL Control の「ダッシュボード」タブが表示されます。

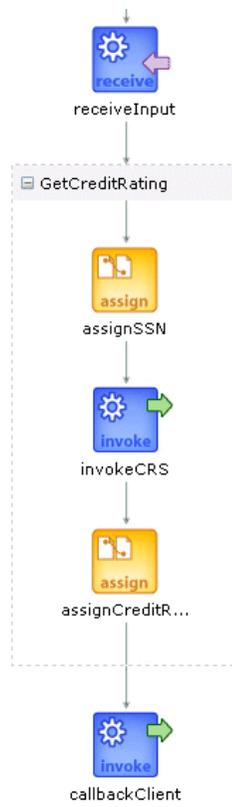
3. 「デプロイ済の BPEL プロセス」リストで「myCreditFlow」をクリックします。
4. HTML フォームの「ssn」フィールドに、先頭がゼロでない9桁の整数値を入力し、「XML メッセージの転送」をクリックします。ユーザー・インタフェースに表示される名前の ssn は、2-6 ページの「XSD ファイルの編集」で入力した値であることに注意してください。

「BPEL プロセス」タブに次のようなメッセージが表示されます。

テスト・インスタンスが開始されました

インスタンス '39e706a46ad531be:858bf1:fcc240f310:-7ffc' は非同期で処理中です。

5. プロセス・インスタンスの現在の状態のビジュアル監査証跡を表示するには、「ビジュアル・フロー」をクリックします。



プロセスの現在の状態を示す監査証跡が表示されます。これは、信用格付けサービスが正常に起動したことを示します。

6. 監査証跡で「callbackClient」をクリックし、信用格付けサービスからクライアントに返された信用格付け（560）を含めた融資提案の結果を参照します。

callbackClient

```
[2006/07/21 16:23:28]
Skipped callback "onResult" on partner "client".
- <outputVariable>
- <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="payload">
- <myCreditFlowProcessResponse xmlns="http://xmlns.oracle.com/myCreditFlow">
  <ns0:creditRating xmlns="http://services.otn.com"
    xmlns:ns0="http://xmlns.oracle.com/myCreditFlow">560</ns0:creditRating>
</myCreditFlowProcessResponse>
</part>
</outputVariable>
```

[Copy details to clipboard](#)

7. 追加情報を確認するには、上部の「監査」リンクをクリックします。

```
<process>
  <sequence>
    receiveInput
      [2006/07/21 16:23:28] Received "inputVariable" call from partner "client" More...
      <scope name="GetCreditRating">
        <sequence>
          assignSSN
            [2006/07/21 16:23:28] Updated variable "invokeCRS_process_InputVariable" More...
          invokeCRS
            [2006/07/21 16:23:28] Invoked 2-way operation "process" on partner "CreditRatingService". More...
          assignCreditRating
            [2006/07/21 16:23:28] Updated variable "outputVariable" More...
        </sequence>
      </scope>
    callbackClient
      [2006/07/21 16:23:28] Skipped callback "onResult" on partner "client". More...
  </sequence>
  [2006/07/21 16:23:28] BPEL process instance "7" completed
</process>
```

融資調達 BPEL プロセスの確認

この章では、第2章「信用フローのチュートリアル」で作成した BPEL プロセスより高度なプロセスの実行方法について説明します。フォルト処理、非同期サービスとの対話、ユーザーのタスクと手動のタスク、Java と J2EE コードの統合およびカスタム・ユーザー・インタフェースとポータルは、すべてが、多くのビジネス・フローにとって主要な要件となります。この章では、これらすべての要件を満たす BPEL プロセスの作成、テスト、デバッグおよび管理の方法を説明します。

この章の内容は次のとおりです。

- [LoanFlowPlus プロセスの概要](#)
- [Oracle JDeveloper でのプロセスの作成と確認](#)
- [Web サービスの起動フレームワークの確認](#)
- [BPEL プロセスを開始するためのカスタム・ユーザー・インタフェースの作成](#)
- [LoanFlowPlus プロセスの開始](#)
- [ビジュアル監査証跡の確認](#)
- [進行中の BPEL プロセスのデバッグ](#)
- [非同期の StarLoan サービスの実行](#)
- [ヒューマン・タスクの実行](#)
- [パフォーマンス・チューニング](#)
- [プロセスのライフサイクルの管理](#)
- [プロセス・レポートの作成と表示](#)
- [開発環境における BPEL プロセスのテスト](#)
- [Web サービス管理ソリューションとの統合](#)
- [デモおよびチュートリアル](#)

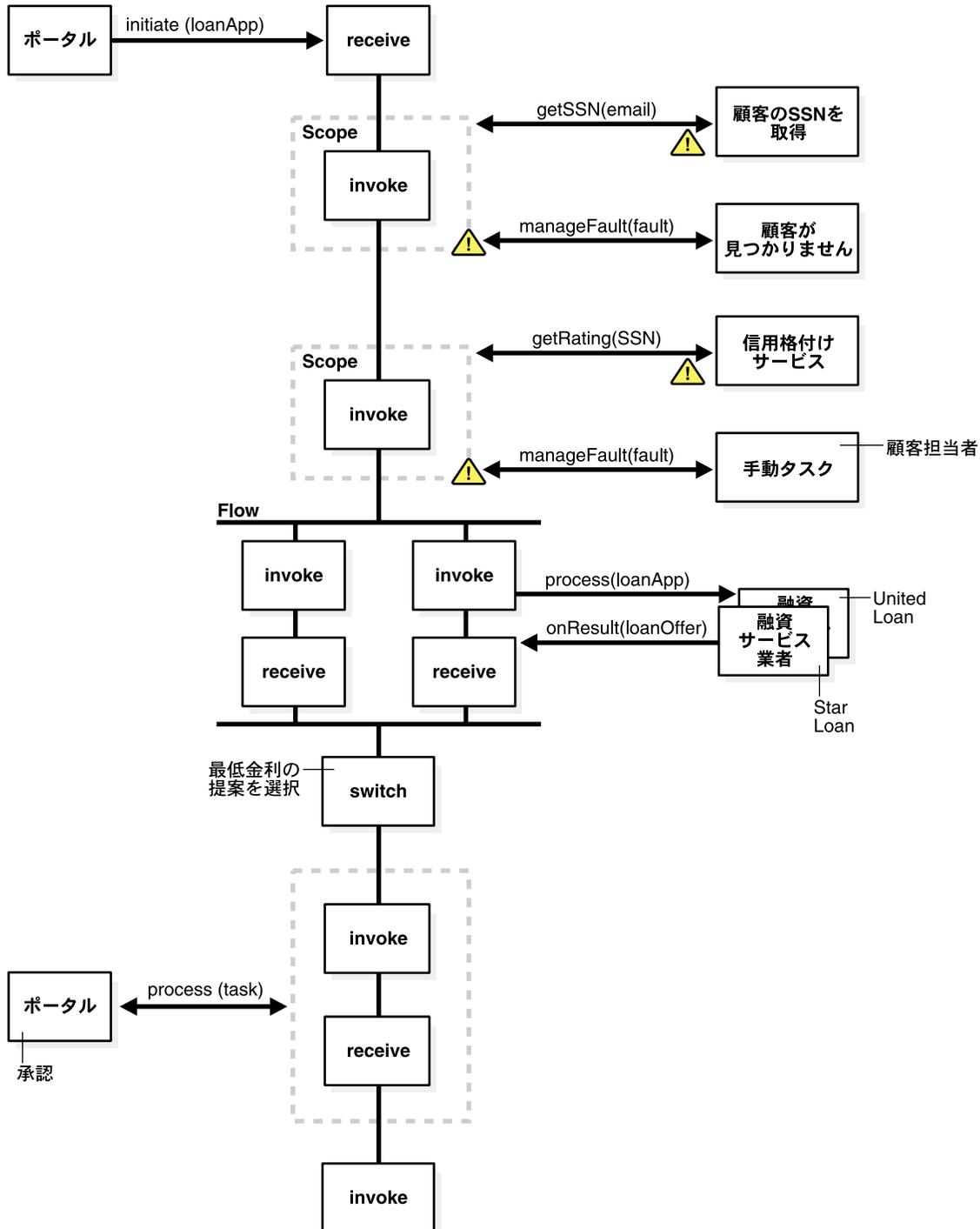
LoanFlowPlus プロセスの概要

この章では LoanFlowPlus プロセスについて説明します。このプロセスは Oracle BPEL Process Manager のサンプルとともに出荷され、インストール後に次のディレクトリで使用可能になります。

`SOA_Oracle_Home\bpel\samples\demos\LoanDemoPlus`

図 3-1 に、LoanFlowPlus で実装される要件を示します。

図 3-1 LoanFlowPlus の概要



このプロセスは実行時間の長い BPEL フローで、融資申込書を入力として受け取り、選択および承認された融資提案を非同期で返します。

このフローでは、プロセス実行の最初に、(ネイティブな EJB Web Services Invocation Framework (WSIF) バインディングを介してサービスとしてアクセスした) Enterprise Java Bean (EJB) から顧客の社会保障番号を取得し、第 2 章「信用フローのチュートリアル」で説明した同期の信用格付けサービスにアクセスして顧客の信用格付けをリクエストします。

このプロセスは、顧客に信用上のマイナス事例（破産など）が確認された場合に信用格付けサービスから通知される可能性があるマイナスの信用フォルトを処理できるようにコーディングされています。このような例外が通知された場合、LoanFlowPlus プロセスは顧客サービス担当者が手動で例外を処理するためのヒューマン・タスクを開始し、そのタスクの完了を待機します。顧客サービス担当者は、ダッシュボード・ユーザー・インタフェースを使用してタスクを表示し、信用格付けを指定したり申込みを取り消すことができます。

信用格付けが提示された後は、手動または自動のアクティビティのどちらを介したものでプロセスは自動処理に入ります。次にプロセスは、完了した融資申込みを 2 つの融資プロセッサに送付します。各プロセッサが融資提案を返すまでにかかる時間は不定です。融資プロセッサは、実行時間が長い場合があるため、非同期のサービスとして実装され、BPEL プロセスから並列起動されます。

この BPEL フローのケースでは、両方の融資提案の受信を待った後、簡単なブランチ・ロジックを使用して最適な提案（最も低金利のもの）を選択します。最後に、別のヒューマン・タスクによって、顧客は選択された融資提案を確認し承認することができます。このタスクが完了した後、選択および承認された提案がフローの結果として返されます。

Oracle JDeveloper でのプロセスの作成と確認

この項では、BPEL プロセスで使用する外部サービスを作成およびデプロイする方法と、Oracle JDeveloper でプロセスを起動および表示する方法を説明します。

サービスの起動とテスト

第 2 章「信用フローのチュートリアル」と同様に、LoanFlowPlus プロセスは、ant コマンドを使用してすぐに作成およびデプロイできるいくつかの外部サービスに依存しています。これを実行するには、次のようにします。

1. 「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「Developer Prompt」を選択して、コマンド・プロンプトを `SOA_Oracle_Home¥bpel¥samples` ディレクトリで開きます。

2. `demos¥LoanDemoPlus` ディレクトリへ移動します。

```
cd demos¥LoanDemoPlus
```

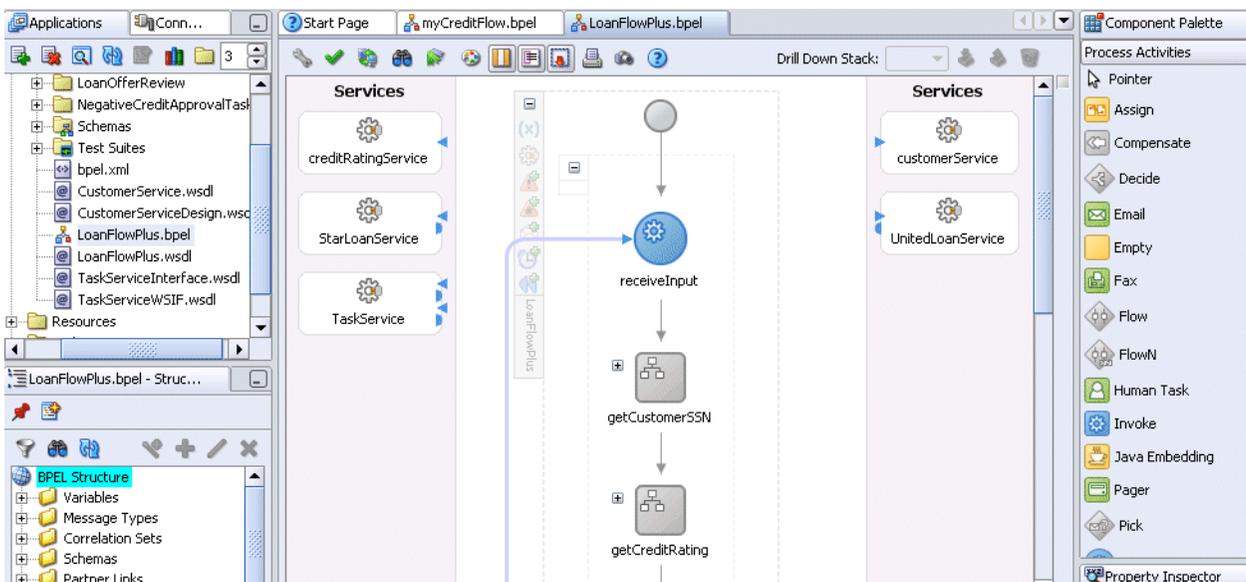
3. 次のコマンドを入力します。

```
ant
```

これで、LoanFlowPlus プロセス自体も含め、プロセスに必要な外部の融資サービスが作成およびデプロイされます。

Oracle JDeveloper での LoanFlowPlus プロセスのオープン

1. Oracle JDeveloper に戻ります。
2. 「アプリケーション・ナビゲータ」で「myBPELApplication」を選択します。
3. 「ファイル」メイン・メニューから「開く」を選択します。
「開く」ウィンドウが表示されます。
4. SOA_Oracle_Home¥bpel¥samples¥demos¥LoanDemoPlus¥LoanFlowPlus ディレクトリに移動します。
5. 「LoanFlowPlus.jpri」を選択して「開く」をクリックします。
この LoanFlowPlus のプロジェクト・ファイルが、Oracle JDeveloper で表示されます。
6. 「アプリケーション・ナビゲータ」で、「LoanFlowPlus」→「インテグレーション・コンテンツ」の順に開きます。
7. 「LoanFlowPlus.bpel」をダブルクリックします。
これでプロセスの「ダイアグラム」ビューが表示されます。



creditRatingService パートナ・リンクは、第2章「信用フローのチュートリアル」で使用したものと同期のサービスです。ただし、**StarLoanService** と **UnitedLoanService** のパートナ・リンクは非同期のサービスで、一方向の開始操作と融資提案の準備完了後に各サービスからコールされる非同期の **onResult** コールバックをサポートしています。これは任意の場所から利用でき、数秒から数日以上かかることがあります。前のプロセスと同様に、**LoanFlowPlus** に対するクライアント・インターフェースも非同期です。

8. 全体プロセス中央のスィムレーンのビジュアルを見てください。Oracle JDeveloper では、標準的な BPEL ソースをネイティブ・フォーマットで使用します。したがって、随時 BPEL の「ソース」ビューを選択して、基礎となる BPEL のソース・コードを表示したり、ビューでコードを変更してすぐに別のビューでこれらの変更を表示することもできます。「ダイアグラム」ビューには、プロセスが開始していることや、社会保障番号を取得するためのロジックが（例外処理コードも含めて）BPEL スコープ内にカプセル化されていることなどが示されます。

以降の各項では、このプロセスを実装するコードを確認します。

フォルト管理

LoanFlowPlus プロセスは、社会保障番号と信用格付けサービスによって通知された例外を、BPEL のフォルト処理を使用して捕捉および管理します。フォルト・ハンドラは特定の **scope** アクティビティに関連付けられており、あるスコープ内で発生してそのスコープで処理されないフォルトは包含するスコープに渡されます。これは、Java の **try-catch** とまったく同じです。

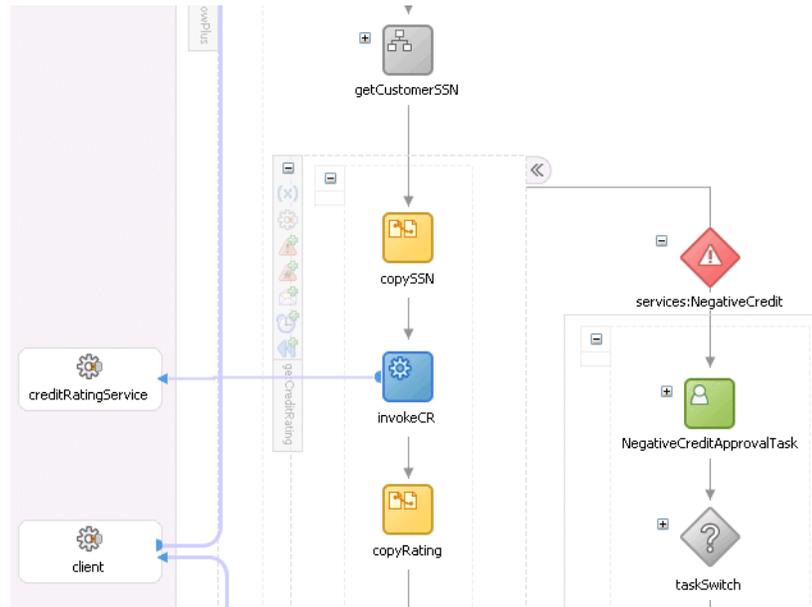
getCreditRating 用に定義された **NegativeCredit** フォルト・ハンドラを表示するには、次のようにします。

1. 「+」記号をクリックして、**getCreditRating** のスコープを開きます。



これでそのスコープのビューが開きます。

2. 実行中のメイン・フローがスコープ内で上から下へ表示されます。また、二次フローが右側に拡張して表示されます。このフローの最上部には、**error** アクティビティ **services:NegativeCredit** が含まれています。



3. 例外フローの中で **scope** アクティビティを開き、このフォルト処理のロジックとして実行された手順を表示します。BPEL 内のどのアクティビティでも（複雑なもの、実行時間が短いものや長いものなど）フォルト・ハンドラで使用できます。この場合、顧客サービス担当者用に例外を手動で処理するためのヒューマン・タスクが定義されています。また、フォルトを自動化ロジックで処理したり、再通知することもできます。

フォルト処理を実装するコードの一部を示します。

```
<scope name="getCreditRating" variableAccessSerializable="no">
  <variables>
    . . .
    . . .
  </variables>
  <!-- Watch for faults (exceptions) being thrown from creditRatingService -->
  <faultHandlers>
    <catch faultName="services:NegativeCredit" faultVariable="crError">
      . . .
    </catch>
  </faultHandlers>
</scope>
```

```

</faultHandlers>
<sequence>
. . .
. . .
<!-- Invoke the CreditRating Service, the URL of this service's
      WSDL is specified in the deployment descriptor -->
  <invoke name="invokeCR" partnerLink="creditRatingService"
         portType="services:CreditRatingService" operation="process"
         inputVariable="crInput" outputVariable="crOutput"/>
. . .
</sequence>
</scope>

```

関連資料: 例外管理、信頼性およびクラッシュ・テストの詳細は、次の URL にある、レジリエンスがある BPEL プロセスの作成と Oracle BPEL Server 上でのクラッシュ・テストに関する TechNotes（技術情報）を参照してください。

<http://www.oracle.com/technology/bpel>

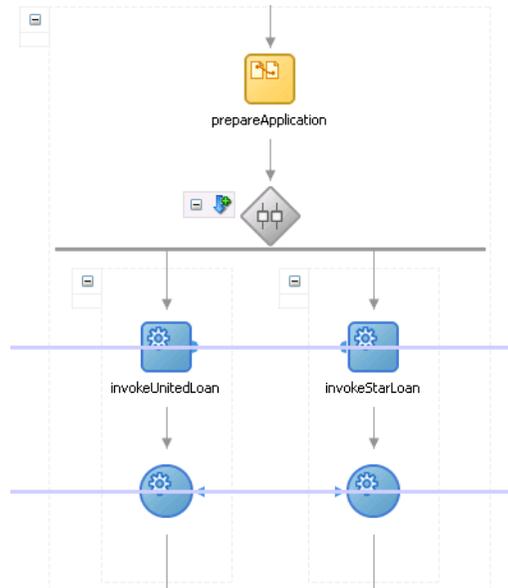
非同期融資プロセッサ・サービスとの対話

記入が完了すると、融資申込書は 2 つの非同期融資プロセッサ・サービスへ送付されます。この項では、BPEL プロセス内の非同期サービスとの対話に必要な事項について検討します。前述のとおり、BPEL は根本的に非同期のアクティビティをサポートし、Oracle BPEL Process Manager はデハイドレーション機能をサポートします。それは、フローが非同期イベントを待機しているときは常に、現在の状態情報とともに確実かつ効率的にデータ・ストアに永続格納されることを意味しています。

また、BPEL では非同期メッセージの相関についていくつかの標準的な方法がサポートされるため、非同期コールバックは適切な待機プロセス・インスタンスを探し出すことができます。具体的には、Oracle BPEL Server では次のメカニズムがサポートされています。

- メッセージの内容を相関に使用する相関セット・メカニズム
 - 非同期メッセージの相関に SOAP メッセージのヘッダーを使用する WS-Addressing 仕様
- LoanFlowPlus** プロセスによる **UnitedLoan** サービスの起動方法と非同期コールバックの待機方法を確認するには、次のようにします。

1. **scope_collectOffers scope** アクティビティを開きます。
2. **collectOffers flow** アクティビティを開きます。
3. 左側にある **invokeUnitedLoan invoke** アクティビティをダブルクリックします。



「起動」ウィンドウで、このアクティビティが融資申込書を入力変数 (**loanApplication**) として **UnitedLoanService** に渡します。「開始」操作はすぐに戻ります。ただし、次のアクティビティである **onResult** コールバックの **receive** アクティビティは、融資提案とともにサービスがコールバックされるまで待機します。

WS-Addressing はメッセージの相関に対して (デフォルトで) 使用され、Oracle BPEL Server で完全に透過的に処理されます。WS-Addressing の相関情報を確認する場合は、サービスと置き換えられた SOAP メッセージを TCP トンネルを使用して確認できます。

UnitedLoanService から返される提案を受け取る **invoke** アクティビティおよび **receive** アクティビティの BPEL ソース・コードは、BPEL の「ソース」ビューを選択するか、「ビュー」→「ソースに移動」を選択して確認できます。このソース・コードは次のとおりです。

```
<sequence>
  <!-- initiate the remote service -->
  <invoke name="invokeUnitedLoan" partnerLink="UnitedLoanService"
    portType="services:LoanService" operation="initiate"
    inputVariable="loanApplication"/>
  . . .
  <!-- receive the result of the remote service -->
  <receive partnerLink="UnitedLoanService"
    portType="services:LoanServiceCallback" operation="onResult"
    variable="loanOffer1" createInstance="no"/>
</sequence>
```

関連資料： サービスと置き換えられた SOAP メッセージを TCP トンネルを使用して確認する方法は、Oracle BPEL Process Manager 開発者ガイドを参照してください。

並列処理の確認

3-2 ページの「[LoanFlowPlus プロセスの概要](#)」の説明にあるように、**LoanFlowPlus** は2つの融資サービス・プロバイダを並列起動する必要があります。これは、プロバイダの処理時間が不定なためです。これには BPEL で flow アクティビティを使用しますが、このアクティビティによりいくつかのアクションの並列実行が可能になります。この例が、3-6 ページの「[非同期融資プロセッサ・サービスとの対話](#)」にある **collectOffers flow** アクティビティに示されています。この flow には、2つの並列アクティビティ、つまり **UnitedLoanService** を起動するシーケンスと **StarLoanService** を起動するシーケンスが含まれています。

この flow アクティビティを実装する簡単な BPEL ソース・コードは、次のとおりです。

```
<flow name="collectOffers">
  <!-- *****
    Invoke the first loan provider (UnitedLoan)
    ***** -->
  <!-- invoke first loan provider -->
  <sequence>
    <!-- initiate the remote service -->
    <invoke name="invokeUnitedLoan" partnerLink="UnitedLoanService"
      portType="services:LoanService" operation="initiate"
      inputVariable="loanApplication"/>
    <!-- receive the result of the remote service -->
    <receive partnerLink="UnitedLoanService"
      portType="services:LoanServiceCallback" operation="onResult"
      variable="loanOffer1" createInstance="no"/>
  </sequence>
  <!-- *****
    Invoke the second loan provider (StarLoan)
    ***** -->
  <sequence>
    <!-- initiate the remote service -->
    <invoke name="invokeStarLoan" partnerLink="StarLoanService"
      portType="services:LoanService" operation="initiate"
      inputVariable="loanApplication"/>
    <!-- receive the result of the remote service -->
    <receive partnerLink="StarLoanService"
      portType="services:LoanServiceCallback" operation="onResult"
      variable="loanOffer2" createInstance="no"/>
  </sequence>
</flow>
```

ヒューマン・タスクの確認

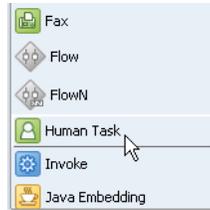
BPEL には非同期サービスに対する基本的なサポートが用意されており、人と手動タスクを BPEL プロセスに統合できます。一連の組込みサービスが Oracle BPEL Server に統合されており、プロセスにヒューマン・ワークフローを含めることができます。これを真の BPEL サービスとして実装することにより、これらのサービスに対するインタフェースが WSDL で記述され、人を BPEL の 100% 標準のプロセスに含めることができます。BPEL プロセスには、人と手動タスクは他の非同期 Web サービスと同じように見えます。

LoanFlowPlus プロセスでは、手動タスクがいくつかの場所で示されています。

- 例外管理の場合、3-5 ページの「[フォルト管理](#)」の説明にあるとおりです。
- 承認ステップの場合、選択された融資提案の顧客による確認が完了するまでフローが待機します。

ヒューマン・ワークフローの実装方法を確認するには、次のようにします。

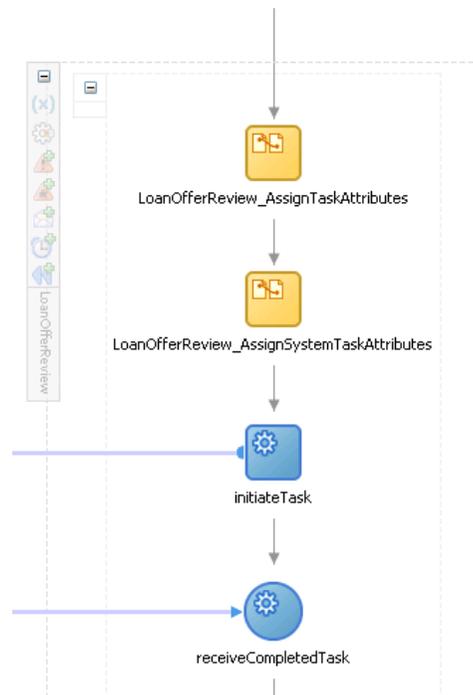
1. 「コンポーネント・パレット」から「プロセス・アクティビティ」を選択します。
2. 「ヒューマン・タスク」アクティビティを確認してください。これをプロセス・フローにドロップできます。これにより、ヒューマン・タスク・エディタでタスク定義を作成できるようになります。このエディタを使用すると、割当て先、承認パターン、通知チャンネルなどを選択することができます。



「ヒューマン・タスク」というパレット要素は、開発者がヒューマン・タスクをより高いレベルの抽象概念として処理できるようにするテンプレート・アクティビティですが、一連の BPEL アクティビティとして実装されています。

3. LoanOfferReview human task アクティビティを開きます。

最初にいくつかの **assign** アクティビティがあります。次にプロセスが **TaskService** パートナ・リンクをコールします。このパートナ・リンクが、タスクに対して必要な操作を公開します。



ユーザーに割り当てられたタスクの間合せやタスク・データの更新およびタスクの実行または取消しをクライアントが実行できるように、API が用意されています。さらに、タスクに関するグループとロールのサポート有効期間のチェック、エスカレーションおよびその他の一般的なヒューマン・タスクの要件がタスク・サービスによって実装されています。

LoanFlowPlus プロセスでは、ユーザーの対話はすべてカスタム Java Server Pages で Java API を使用して処理されます。顧客による確認中に、次の項目を実行します。

- **LoanFlowPlusUI** インタフェース (<http://localhost:8888/LoanFlowPlusUI>) を介したプロセスとの対話
- **ExceptionDashboardUI** インタフェース (<http://localhost:8888/ExceptionDashboardUI>) を介したプロセスとの対話

オラクルでは、特別な設定なしでワークフロー対応プロセス内でのヒューマン・インタフェースの処理に使用（およびカスタマイズ）できる **Oracle BPEL Worklist Application** も提供しています。

関連資料： BPEL 内のヒューマン・ワークフローおよびタスク・サービス・クライアントの作成の詳細は、**Oracle BPEL Process Manager 開発者ガイド**のワークフロー・サービスに関する章を参照してください。

Web サービスの起動フレームワークの確認

企業では、バックエンド・システムへのアクセスのために既存の Java コードまたは Java API を BPEL プロセス内で使用する場合があります。このためには、一般的に次の 3 つの方法があります。

- Java コードを Web サービスとしてラッピングする。
- Java コードを BPEL プロセスにインライン化する。
- WSIF 起動フレームワークを使用して Java コードを Web サービスのようにコールする。

最初の 2 つの方法は、Oracle BPEL Process Manager で提供されるいくつかのコード例にわかりやすく示されています。

3 つ目の方法の例として、**LoanFlowPlus** プロセスでは WSIF を使用して EJB を起動し、これが顧客の社会保障番号を返します。BPEL プロセスから、EJB サービスを WSDL インタフェースを持つ他のサービスの場合と同じように扱い、そのためのパートナ・リンクを BPEL プロセス内に作成することができます。EJB により提供される機能は、WSDL がインタフェースをとるサービスに EJB バインディングを介して直接マップされます。「アプリケーション・ナビゲータ」内の **CustomerService.wsdl** ファイルに、このバインディングが示されています。

```
<!-- binding declns -->
<binding name="EJBBinding" type="tns:CustomerService">
  <ejb:binding/>
  <format:typeMapping encoding="Java" style="Java">
    <format:typeMap typeName="xsd:string" formatType="java.lang.String" />
    <format:typeMap typeName="tns:CustomerNotFoundException"
formatType="com.otn.samples.CustomerNotFoundException" />
  </format:typeMapping>
  <operation name="getCustomerSSN">
    <ejb:operation
      methodName="getCustomerSSN"
      parameterOrder="email"
      interface="remote"
      returnPart="ssn" />
    <input name="GetCustomerSSNRequest"/>
    <output name="GetCustomerSSNResponse"/>
    <fault name="CustomerNotFoundException"/>
  </operation>
</binding>

<!-- service decln -->
<service name="CustomerService">
  <port name="EJBPort" binding="tns:EJBBinding">
    <!-- Put appserver vendor-specific deployment information here -->
    <ejb:address className="com.otn.samples.CustomerServiceHome"
      jndiName="ejb/session/CustomerService"
      initialContextFactory="com.evermind.server.rmi.RMIInitialContextFactory"
      jndiProviderURL="ormi://{hostname}/CustomerService"/>
  </port>
</service>

  <plnk:partnerLinkType name="CustomerService">
    <plnk:role name="CustomerServiceProvider">
      <plnk:portType name="tns:CustomerService"/>
    </plnk:role>
  </plnk:partnerLinkType>
</definitions>
```

このコードは、EJB が WSDL で記述されたサービスとしてアクセスされていることを意味しています。ここでは、WSIF が操作を EJB メソッドへマップしています。

注意： Oracle JDeveloper 10.1.3.1.0 では、現在サポートされている SOAP バインディングに加えて、POJO (Plain Old Java Objects) および EJB 用の Java および EJB バインディングを備えた WSDL を自動的に生成します。

BPEL プロセスを開始するためのカスタム・ユーザー・インタフェースの作成

BPEL プロセスはすべて、それ自身が Web サービスです。また Oracle BPEL Process Manager には、デプロイされた BPEL フローの起動や、アクティブ・インスタンスの状態とステータス情報のフェッチなどのための Java API が提供されています。BPEL プロセスは、ポータルまたはその他のカスタム・ユーザー・インタフェースからインスタンス化されることがよくあります。たとえば、LoanFlowPlus プロセスには JSP 融資インタフェースがあり、顧客はこれを使用して新規の融資申込みの開始や、受け取った提案の確認、提案の承認を行うことができます。LoanFlowPlus のユーザー・インタフェースは、Oracle BPEL Process Manager と同じアプリケーション・サーバー上にデプロイされ、その全ソースが次の場所のサンプル内にあります。

`SOA_Oracle_Home\bpel\samples\demos\LoanDemoPlus\LoanFlowPlusUI`

デプロイされた BPEL プロセスを開始するための Java API の JavaDocs は、次のファイルにあります。

`SOA_racle_Home\bpel\docs\apidocs\index.html`

さらに、この Java API を JSP から簡単に使用するための JSP タグ・ライブラリが利用できます (このタグ・ライブラリは、LoanFlowPlusUI のサンプルで使用されています)。開発者は、Web サービスの SOAP および WSDL API を使用して BPEL プロセスを起動することもできます。Web サービスを使用する方法では、Web サービスをサポートする任意の言語またはツール・キットから BPEL フローを起動したりアクセスできます。

LoanFlowPlus プロセスの開始

ここでは、ポータル・ユーザー・インタフェースを使用して LoanFlowPlus インスタンスを開始および実行し、Oracle Enterprise Manager 10g BPEL Control を使用してステータス、監査証跡、デバッグ情報、パフォーマンス・メトリックおよび Oracle BPEL Server で自動収集されたその他の情報を表示します。

LoanFlowPlus BPEL プロセスは、3-3 ページの「サービスの起動とテスト」で ant コマンドを実行したときに、依存するサービスとポータル・ユーザー・インタフェースとともにコンパイルおよびデプロイされています。Oracle JDeveloper からも、第 2 章「信用フローのチュートリアル」の場合と同じようにプロセスを作成およびデプロイできます。

LoanFlowPlus インスタンスを開始および実行するには、次のようにします。

1. 次のポータル・ユーザー・インタフェースに移動します。

`http://localhost:8888/LoanFlowPlusUI/Homepage.html`

2. ポータル・ページの「Initiate New BPEL Loan Flow」リンクをクリックします。

融資申込みを発行して新しい LoanFlowPlus インスタンスを開始するための Web ページが表示されます。

3. ユーザー・インタフェース内の各フィールドを必要に応じて変更し、「**Submit Loan Application**」 ボタンをクリックして新しいプロセス・インスタンスを開始します。

ビジュアル監査証跡の確認

開発者または管理者のタスクを実行し、Oracle Enterprise Manager 10g BPEL Control を使用してこのプロセスに関する監査証跡とその他のステータス情報を確認します。

- 次のいずれかの方法で Oracle Enterprise Manager 10g BPEL Control にアクセスします。
 - 「スタート」 → 「すべてのプログラム」 → 「Oracle - Oracle_Home」 → 「Oracle BPEL Process Manager」 → 「BPEL Control」 を選択します。
 - 次の URL に移動します。

`http://localhost:8888/BPELConsole/`

- Oracle Enterprise Manager 10g BPEL Control へのログインに必要な次の情報を入力して「ログイン」をクリックします。

フィールド	値
ユーザー名	bpeladmin
パスワード	password

password は、インストール時に 1-6 ページの「Oracle Application Server SOA Suite のインストール」で入力した値です。

BPEL プロセスのインスタンスで、進行中のものと終了したものがいくつか表示されます。

- 「インスタンス」列で **LoanFlowPlus** プロセスのアクティブなインスタンスをクリックします。

Dashboard				BPEL Processes	Instances	Activities
Deployed BPEL Processes		In-Flight BPEL Process Instances 1 - 3				
Name	Instance	BPEL Process	Last Modified ↑			
CreditRatingService	13 : Instance #13 of StarLoan	StarLoan (v. 1.0)	7/22/06 10:03:36 AM			
LoanFlowPlus	11 : Loan Flow Plus- John Smith	LoanFlowPlus (v. 1.0)	7/22/06 10:03:36 AM			
StarLoan	9 : Instance #9 of LoanFlowPlus	LoanFlowPlus (v. 1.0)	7/22/06 10:01:22 AM			
TaskActionHandler						
TaskManager						
UnitedLoan						
Recently Completed BPEL Process Instances (More...)						
	✓ 14 : Instance #14 of UnitedLoan	UnitedLoan (v. 1.0)	7/22/06 10:03:34 AM			
	✓ 12 : Instance #12 of CreditRatingService	CreditRatingService (v. 1.0)	7/22/06 10:03:33 AM			
	✓ 10 : Instance #10 of CreditRatingService	CreditRatingService (v. 1.0)	7/22/06 10:01:20 AM			
	7 : Instance #7 of myCreditFlow	myCreditFlow (v. 1.0)	7/21/06 4:23:28 PM			
	8 : Instance #8 of CreditRatingService	CreditRatingService (v. 1.0)	7/21/06 4:23:28 PM			

Oracle Enterprise Manager 10g BPEL Control に表示されるものはすべて、API でも利用できます。実際に、Oracle Enterprise Manager 10g BPEL Control は JSP の集合であり、Oracle BPEL Server API の使用方法を示すコード例として機能します。開発者がこれらの API を使用して頻繁にプロセスのダッシュボードや検索画面およびその他のインタフェースをカスタマイズするため、ユーザーや管理者はより効率的にプロセス・インスタンスにアクセスおよび管理できます。これは特に、管理対象のプロセス・インスタンスが何千または何百万にもなる可能性がある本番環境にアプリケーションがデプロイされる場合によくあることです。

Oracle Enterprise Manager 10g BPEL Control で **LoanFlowPlus** インスタンスを選択すると、プロセスの現在のステータスとその実行履歴を示したビジュアル監査証跡が表示されます。この監査証跡は、Oracle BPEL Server によって自動的に生成、保守されますが、記録される情報は制御可能です。

- ビジュアル監査証跡内のアクティビティをクリックして、そのアクティビティ（この例では **receiveInput**）の詳細情報を表示します。たとえば次のウィンドウでは、クライアントの開始が選択されており、フローを開始するために送信された XML 入力メッセージが表示されています。

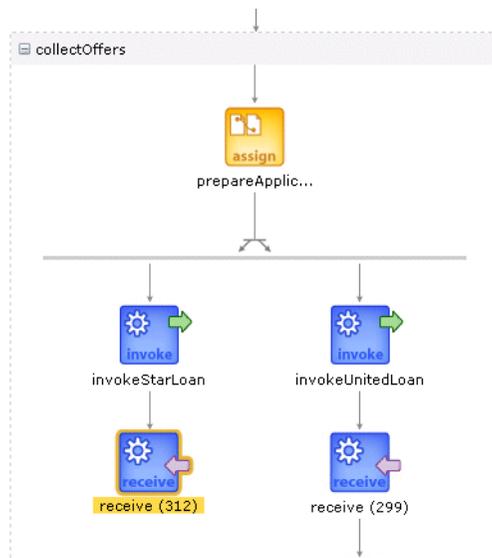
The screenshot shows the Oracle BPEL Process Manager interface. On the left, the 'Flow' tab is active, displaying a visual representation of the business flow. The flow starts with a 'start' activity, followed by a 'receiveInput' activity (highlighted with a dashed box), then a 'getCustomerSSN' activity, and finally an 'assignRequest' activity. On the right, the 'Activity Audit Trail' window is open, showing details for the 'receiveInput' activity. The details include a timestamp [2006/08/20 09:51:35], the event 'Received "input" call from partner "client"', and the XML payload:

```

<input>
  <part
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="payload">
    <loanApplication
      xmlns="http://www.autoloan.com/ns/autoloan">
      <SSN/>
      <email>demo1@otn.com</email>
      <customerName>John Smith</customerName>
      <loanAmount>100000</loanAmount>
      <carModel>BMW Z8</carModel>
      <carYear>2006</carYear>
      <creditRating>0</creditRating>
    </loanApplication>
    </part>
  </input>
  
```

Below the XML is a link: [Copy details to clipboard](#).

- ビジュアル監査証跡の一番下までスクロールして、このインスタンスの最新のステータスを確認します。この場合、StarLoan サービスが開始していますが、フローはレスポンスの受信を待機しています（オレンジ色で強調表示されています）。ただし UnitedLoan サービスは、すでに融資提案とともにコールバックされています。



- テキストの監査証跡を表示するには、「監査」タブをクリックします。テキストの監査証跡には、サービスとやりとりしたメッセージや、各アクティビティが開始または終了した時のタイムスタンプなどの情報が含まれています。これは、API でも構成およびアクセスできます。

進行中の BPEL プロセスのデバッグ

1. 「デバッグ」タブをクリックして BPEL Debugger を表示します。このプロセスを実装する BPEL ソース・コードが取得され、この特定のインスタンスの状態にあわせて表示されます。実行が現在停止しているコードの位置が黄色で強調表示されます。このコードの一番下に示されているとおり、プロセスは現在 StarLoan サービスによる融資提案のコールバックを待機しています。



2. デバッガで、変数を選択してその現在値を調べます。この図に示されているのは融資提案変数 (`loanOffer1`) の値で、UnitedLoan サービスのコールバック操作の結果が含まれています。図からわかるとおり、UnitedLoan は融資の申込みを承認し、金利 5.7% の融資提案を返しています。

非同期の StarLoan サービスの実行

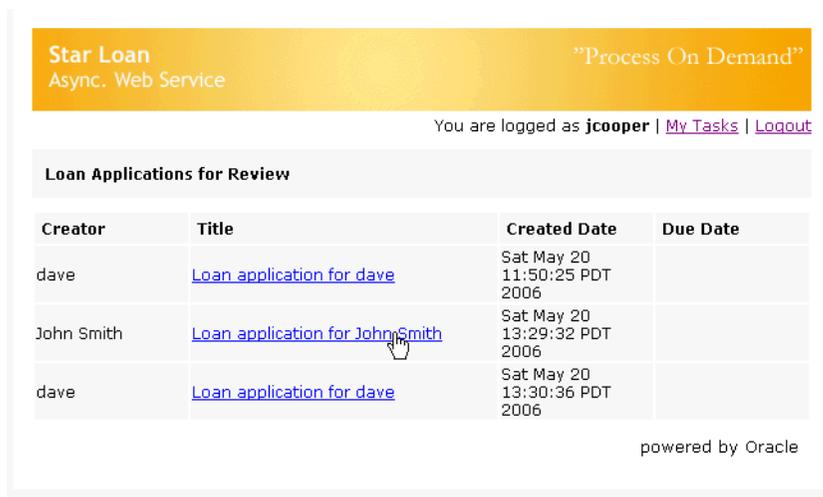
ここでプロセスを完了させる場合は、StarLoan の顧客サービス担当者のダッシュボードを開始します。これは、StarLoan では融資申込みの手動処理が必要なためです。StarLoan は BPEL プロセスとしても実装されており（ただし StarLoan が Microsoft .Net、Apache Axis などの他の Web サービス・ツール・キットで作成された場合に実装できます）、別の BPEL コード例として機能します。

StarLoan の融資担当者の役割を実行するには、次のようにします。

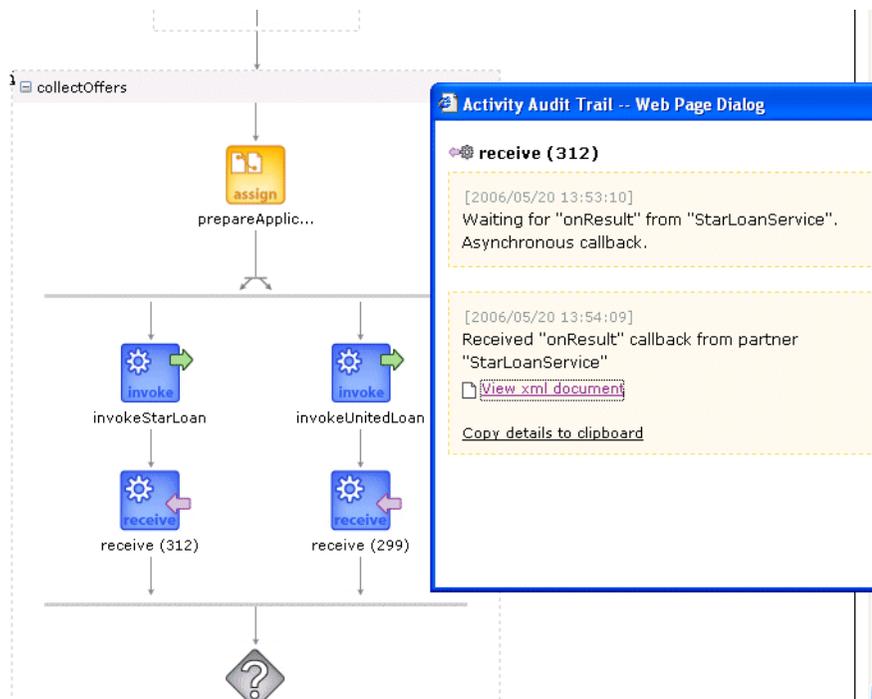
1. 次の URL に移動します。
<http://localhost:8888/StarLoanUI/home.jsp>
2. ログインに必要な次の情報を入力して「発行」をクリックします。

フィールド	値
ユーザー名	jcooper
パスワード	password

保留中の融資申込みが表示されます。



3. 融資申込みを選択して金利を入力し（自分自身の融資申込みを承認するので低い金利を入力します）、「Approve」をクリックします。これで、StarLoan サービスは LoanFlowPlus プロセスにコールバックを返します。
4. Oracle Enterprise Manager 10g BPEL Control で監査証跡をリフレッシュして、処理が進んで 2 番目の融資提案を含んだ StarLoan コールバックを受け取り、最適な融資提案が選択されたことを確認します。現在、プロセスは顧客による提案承認を待機しています。



5. 融資提案に関する詳細を特に表示するには、「XML 文書の表示」をクリックします。

```

<?xml version="1.0" encoding="UTF-8" ?>
- <loanOffer xmlns="http://www.autoloan.com/ns/autoloan">
  <providerName>Star Loan</providerName>
  <selected />
  <approved>true</approved>
  <APR>5.5</APR>
</loanOffer>
    
```

ヒューマン・タスクの実行

1. 顧客融資のポータル・ホームページをリフレッシュします。これで、選択された融資提案でレビューを待機しているものが表示されます。これがプロセスの最終ステップです。

2. この融資提案を選択すると、顧客がこの提案を受け入れるためのページが表示されます。これで、LoanFlowPlus BPEL プロセスのこのインスタンスの実行が完了します。

Provider Name	Star Loan
APR	5.5
Approved	true

パフォーマンス・チューニング

パフォーマンス（スループットと応答時間の要件を含む）は、BPEL プロセスを効率よく実装するための重大な要件となることがよくあります。Oracle BPEL Server と Oracle Enterprise Manager 10g BPEL Control に組み込まれているのが、パフォーマンス・チューニングのフレームワークです。このフレームワークにより、開発者はそのアプリケーションのパフォーマンスを理解し、（開発者のコード、外部サービスまたは Oracle BPEL Server に存在する）ボトルネックを特定して解決することができます。

1. Oracle Enterprise Manager 10g BPEL Control に戻ります。
2. 右上隅の「BPEL ドメインの管理」をクリックして、「統計」をクリックします。
完了したフローのパフォーマンス・データを含んだページが表示されます。

BPEL Domain: default	
Statistics:	6 Active Processes 0 Retired Processes
Build version:	10.1.3.1.0 [build #0] - type: release

Configuration	XPath Library	Logging	Threads	Statistics	Adapter Stats
Runtime Statistics for this BPEL Domain Print Friendly Refresh View Clear Statistics 350 Hotspots <input type="checkbox"/> Filter					
Synchronous BPEL process statistics					Average
CreditRatingService 1.0 (3 stats, min = 109 ms, max = 281 ms)					171.66 ms
Asynchronous process statistics					Average
UnitedLoan 1.0 (3 stats, min = 235 ms, max = 296 ms)					271.0 ms
LoanFlowPlus 1.0 (1 stat, min = 2242531 ms, max = 2242531 ms)					2242531.0 ms
StarLoan 1.0 (3 stats, min = 57312 ms, max = 7338315 ms)					2880146.33 ms
0 hotspots found					
Request breakdown					Average
eng-composite-request (35 stats, min = 109 ms, max = 4687 ms)					714.14 ms
log-sync-stats (3 stats, min = 0 ms, max = 0 ms)					0.0 ms
eng-single-request (133 stats, min = 0 ms, max = 3938 ms)					159.63 ms
scope-marshall (13 stats, min = 0 ms, max = 297 ms)					42.0 ms

Oracle Enterprise Manager 10g BPEL Control では、デプロイしたすべてのプロセスを、自動生成された HTML フォームのインタフェースを介して、または特定の XML メッセージ・コンテンツを渡すことでテストできます。このテスト・ページにはストレス・テスト機能が含まれており、デプロイされた BPEL プロセスの負荷テストを実行し、ストレスを受けた状態のフローのパフォーマンス統計を表示することができます。

3. 「ダッシュボード」タブをクリックします。
4. 「デプロイ済の BPEL プロセス」リストで「LoanFlowPlus」をクリックします。
5. 一番下にスクロールして「ストレス・テストを実行」を選択します。

注意： Oracle Application Server SOA の「基本インストール」タイプに含まれる Oracle Database Lite データベースでは、この組み込みストレス・テスト機能を使用しないでください。

Perform stress test

Number of concurrent threads? (threads)

Number of loops? (loops)

Constant delay between each invocation? (ms)

Clear statistics? Yes No

プロセスのライフサイクルの管理

Oracle BPEL Process Manager と Oracle Enterprise Manager 10g BPEL Control では、プロセスのライフサイクル管理がサポートされています。たとえば、プロセスの新しい実装を、現在進行中の古い実装のインスタンスに影響を与えずにホット・デプロイできるように並列バージョンがサポートされています。さらに、Oracle Enterprise Manager 10g BPEL Control では、プロセス全体の停止、プロセスのアンデプロイ、または既存のインスタンスは終了可能だが新しいインスタンスは認めない、プロセスのリタイアを実行できます。

1. デプロイ済の BPEL プロセスの名前を Oracle Enterprise Manager 10g BPEL Control で選択し、左側の「管理」タブをクリックします。

次のページが表示されます。

BPEL Process: LoanFlowPlus Version: 1.0 Lifecycle: Active
 Statistics: [4 Open Instances](#) | [1 Closed Instances](#)

Manage Initiate Descriptor WSDL Sensors Source Test Suites Reports

Managing this BPEL Process

Process Lifecycle:
 When the process lifecycle is **active** instances may be instantiated from the process; when it is **retired**, no new instances may be instantiated but existing instances are permitted to complete normally.

Active Retired

Process State:
 The process state controls overall access to the process. When the state is **on** instances may be instantiated and accessed; when it is **off** no new instances may be instantiated and access to existing instances and activities belonging to the process will be denied.

On Off

Apply

関連資料： プロセスのライフサイクルの詳細は、Oracle BPEL Process Manager 開発者ガイドを参照してください。

プロセス・レポートの作成と表示

Oracle BPEL Process Manager には、次のプロセス・レポート機能が用意されています。

- [レポート・ダッシュボードの作成](#)
- [Oracle Enterprise Manager 10g BPEL Control のレポートの表示](#)
- [BAM のレポートの表示](#)

レポート・ダッシュボードの作成

ビジネス・プロセスの自動化で重要な機能は、多くの場合、プロセスの状態情報がよりわかりやすく表示されることにあります。これは、集約サマリー情報やプロセス・インスタンスの状態情報を表示するプロセス・ダッシュボードを作成することで実現される場合がよくあります。たとえば、経営者にとって、処理の各段階にある現在の融資数、顧客への融資提案に要する平均時間、または進行中のインスタンスと完了したインスタンスに関するその他の主要なパフォーマンス・インディケータが必要な場合があります。

Oracle BPEL Process Manager では、プロセスの状態に関する大量の情報が保持され、この情報を API を介して使用することができます。また、簡単なレポート・ダッシュボードのテンプレートがいくつか Oracle BPEL Server に付属しており、専門のサービス組織やサポート組織からも入手できます。

LoanFlowPlus プロセスに含まれるテンプレートに、次の URL でアクセスします。

<http://localhost:8888/LoanFlowPlusUI/dashboard.jsp>



Process Status	# of Instances
1-Credit Rating	0
2-Collecting Offers	2
3-Confirming Offer	0
4-Completed	1
5-Others	2

Title	Email	Started	Finished	Process Status
Instance #101 of LoanFlowPlus	demo1@otn.com	May 20 06, 11:50 AM		5-Others
Loan Flow Plus- John Smith	demo1@otn.com	May 20 06, 01:29 PM	May 20 06, 02:06 PM	4-Completed
Instance #205 of LoanFlowPlus	demo1@otn.com	May 20 06, 01:30 PM		2-Collecting Offers
Instance #209 of LoanFlowPlus	demo1@otn.com	May 20 06, 01:53 PM		5-Others
Instance #213 of LoanFlowPlus	demo1@otn.com	May 20 06, 02:43 PM		2-Collecting Offers

Oracle Enterprise Manager 10g BPEL Control のレポートの表示

Oracle BPEL Process Manager では、プロセスの最適化を可能にするための組込みレポートも Oracle Enterprise Manager 10g BPEL Control に提供されています。

1. Oracle Enterprise Manager 10g BPEL Control に戻ります。
2. 「ダッシュボード」タブの「デプロイ済の BPEL プロセス」リストで「LoanFlowPlus」をクリックします。
3. 「レポート」タブをクリックします。
4. 特定のレポート・タイプを選択し、特定の問合せ詳細を指定して「実行」をクリックします。レポート・タイプとしてプロセスの時間分布の例を次に示します。

Activity Name	Count	Average Execution Time (seconds)	Average Execution Time Details
assignRequest	18	7.44	■
assignResponse	18	6.39	■
copyRating	18	1.72	■
copySSN	18	5.61	■
getCustomerSSN	18	7.39	■
initiateTask	18	0	
invokeCR	18	5.56	■
invokeStarLoan	18	1.06	■
invokeUnitedLoan	18	1.06	■
prepareApplication	18	1.06	■
prepareTask	18	0	
prepareTitle	18	0	
readPayload	18	0	
receive (181)	18	5.39	■
receive (187)	18	217.33	■
receiveInput	18	9.33	■
receiveTaskResult	18	184.56	■

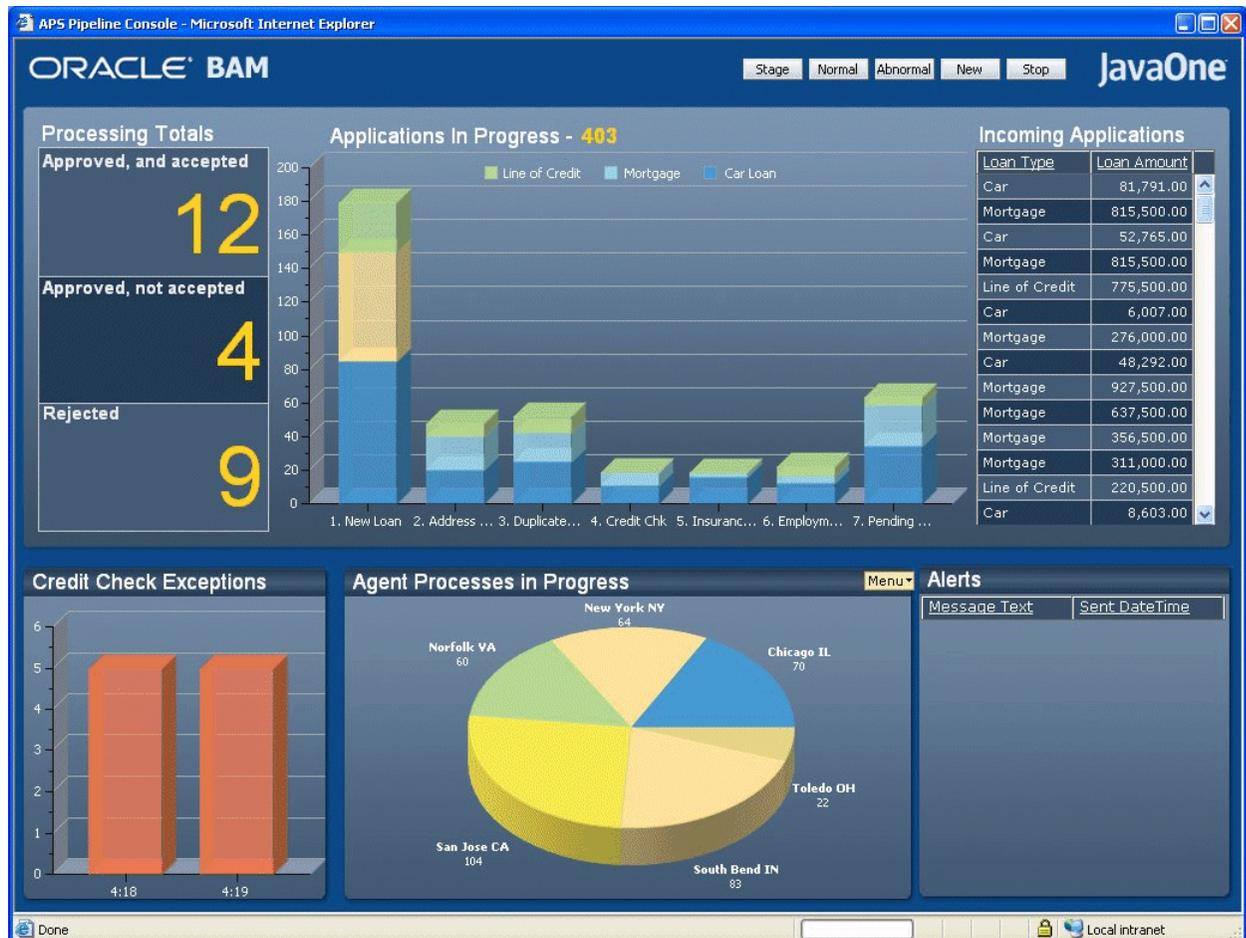
関連資料： Oracle Enterprise Manager 10g BPEL Control のレポートの詳細は、Oracle BPEL Process Manager 開発者ガイドを参照してください。

BAM のレポートの表示

オラクルでは、Oracle BPEL Process Manager と統合された Business Activity Monitoring (BAM) コンポーネントも提供しています。Oracle BAM により、企業の経営陣は、企業内のビジネス・イベントおよびプロセスの監視、メッセージの関連付けと KPI (キー・パフォーマンス・インディケータ: 重要業績評価指数) の定義、およびライブ・データつまりシン・クライアントの Web インタフェースを介してアクセスされたリアルタイム情報に基づいた警告の定義と修正アクションの実行が可能になります。

図 3-2 に、情報を表示している Oracle BAM のユーザー・インタフェースの例を示します。

図 3-2 Oracle BAM のレポート



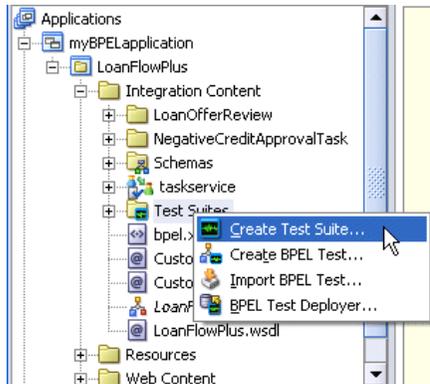
Oracle JDeveloper には、BPEL プロセスから Oracle BAM Server に簡単にイベントを送信するためのセンサーのサポートが含まれています。ただし、Oracle BAM Server は、BPEL プロセス・イベントに限らず、企業内のどのソースのイベントでも (JMS メッセージ、データベース・イベントなど) JMS キューに書き込めるものはすべて収集できます。

開発環境における BPEL プロセスのテスト

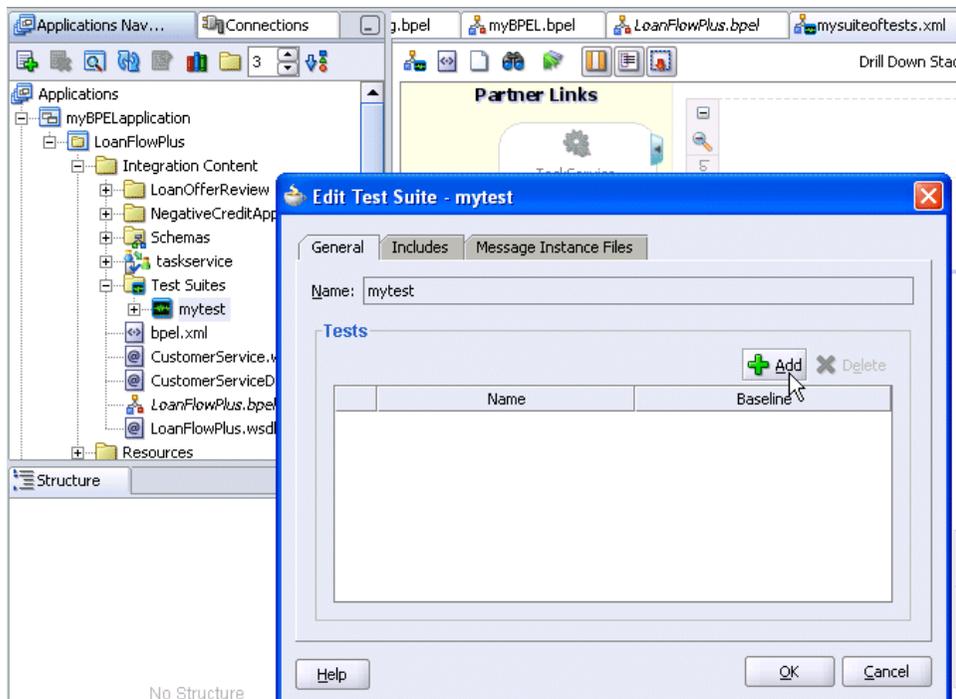
Oracle BPEL Process Manager では、BPEL プロセスを含んだテスト・スイートを作成し Oracle BPEL Server にデプロイできます。テスト・スイートは、開発環境で BPEL プロセスのテストを自動化するための XML テスト・ファイルで構成されます。テストには、パートナー Web サービスとの対話のシミュレーション、テスト・データを使用したプロセス・アクションの検証、および単純なアクティビティにおけるソース・コードの実行率の計算を行うために作成した手順が含まれます。このテストにより、本番環境へのデプロイの前に、プロセスが正常に動作することを確認できます。

テスト・スイートを作成およびデプロイするには、次のようにします。

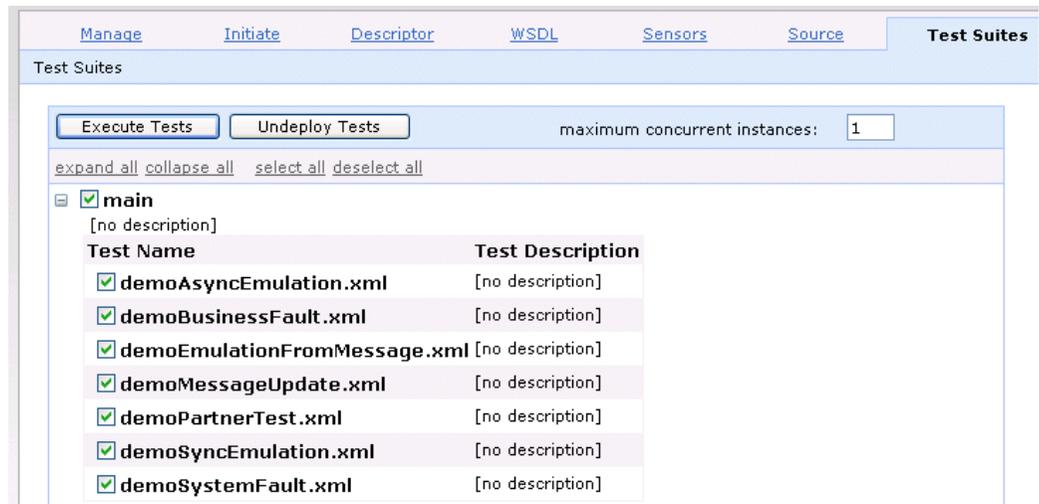
1. Oracle JDeveloper で「テスト・スイート」を右クリックし、「テスト・スイートの作成」を選択します。



2. テスト・スイートの名前を入力して「OK」をクリックします。
3. 作成したテスト・スイートを右クリックし、「テスト・スイートの編集」を選択します。



4. ベースラインおよびエミュレートされたメッセージ・インスタンスのテスト・ファイルをテスト・スイートに追加します。
5. BPEL プロセスに対するテスト・スイートを Oracle BPEL Server にデプロイします。
6. Oracle Enterprise Manager 10g BPEL Control (BPEL Processes 「BPEL プロセス」 タブ → 「process_instance_name」 → 「テスト・スイート」) または ant から BPEL プロセスにテスト・スイートを実行し、XML 文書のレポートを作成します。



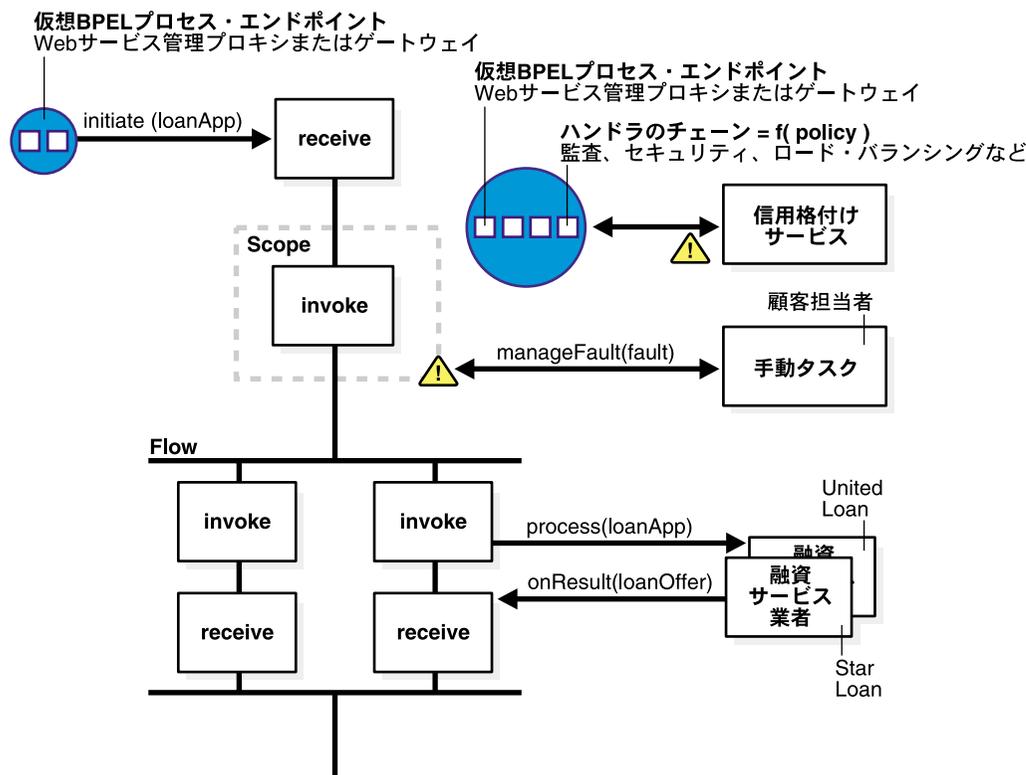
レポート結果を確認し、エラーがあれば修正し、テストの実行を繰り返すことができます。

関連資料： BPEL プロセスのテストの詳細は、Oracle BPEL Process Manager 開発者ガイドを参照してください。

Web サービス管理ソリューションとの統合

サービス指向アーキテクチャ (SOA) を採用している企業内で、サービスとそれを統合する BPEL プロセスの、両方のセキュリティ保護と管理をどのように実行するのかという質問がよく出されます。Oracle Web Services Manager (OWSM) などの Web サービス管理 (WSM) ソリューションでは、認証、認可、暗号化、ロギングなどの機能がポリシー定義を介してサービスとプロセスに提供されます。図 3-3 に示されているとおり、Web サービス管理ソリューションは一般に、Web サービスの前にプロキシまたはゲートウェイを提供します。これには、様々なテクノロジーで実装可能なバックエンド Web サービスと、Oracle BPEL Server やその他の BPEL サーバーにデプロイされた BPEL プロセスの両方が含まれます。これは、BPEL プロセス自身が Web サービスであるためです。

図 3-3 Web サービス管理



OWSM は、ここで管理対象エンドポイントを提供しているため、BPEL プロセスには基礎となる Web サービスとまったく同じように見えます。BPEL プロセスは、サービス自体ではなく管理対象エンドポイントに接続していることを認識する必要はありません。OWSM (または同じ標準をサポートするその他の製品) は、暗号化、認証、サービスの仮想化およびその他の主要な要件を同じ方法で実装できるように、ステップのパイプラインを提供しています。同様に、OWSM は、論理的に Oracle BPEL Process Manager の前に配置して BPEL プロセスへの接続を管理できます。

図 3-4 と図 3-5 に、Oracle Web Services Manager のユーザー・インタフェースを示します。

図 3-4 Oracle Web Services Manager のユーザー・インターフェース

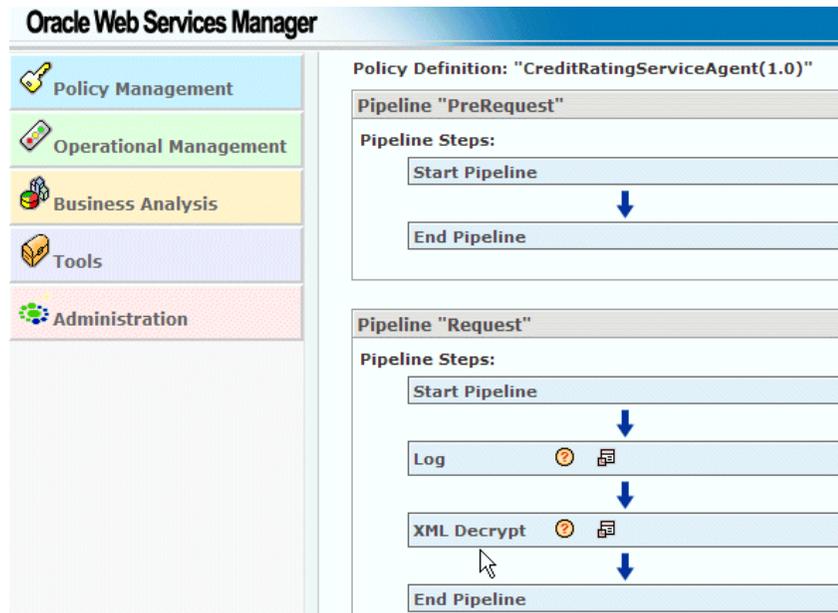
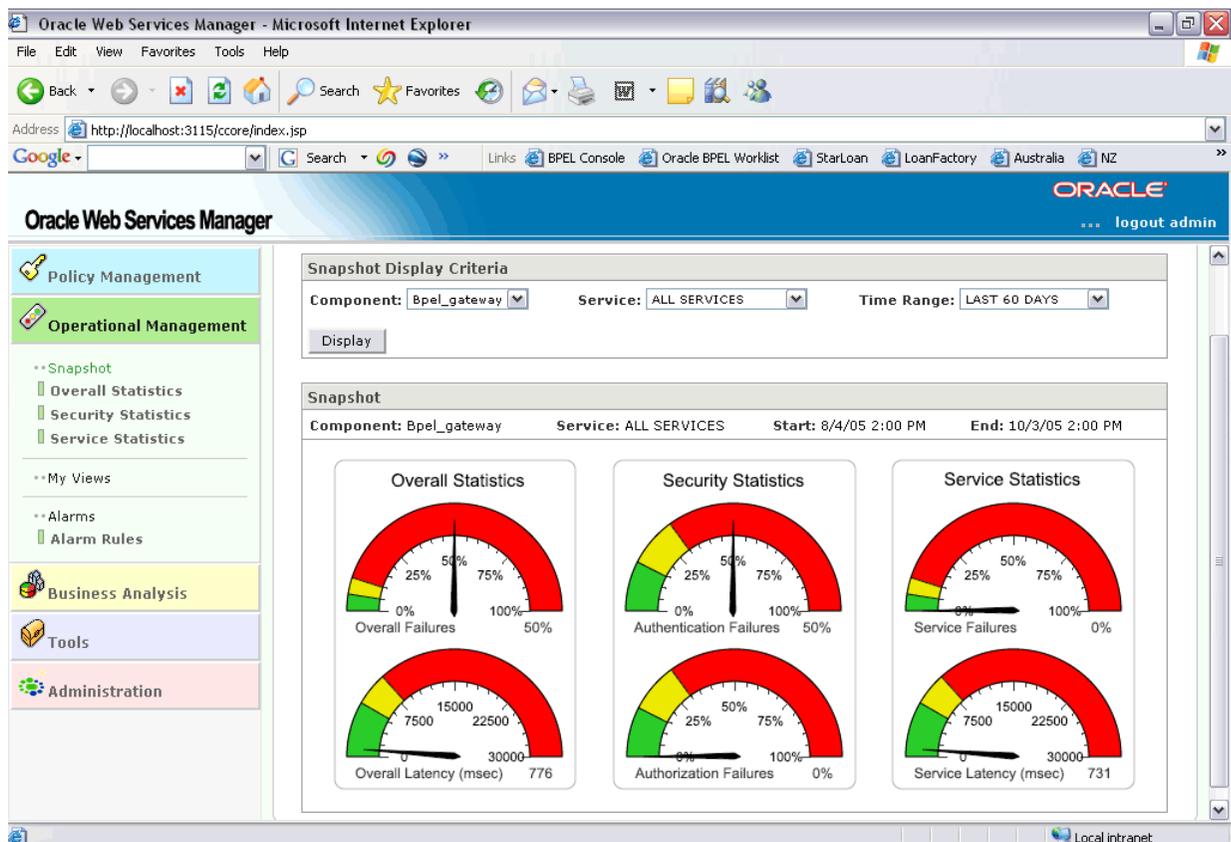


図 3-5 Oracle Web Services Manager のユーザー・インターフェース



デモおよびチュートリアル

一連のデモ、アクティビティと概念の参照資料、およびチュートリアルも、Oracle BPEL Process Manager の概要に関する知識と実地経験を増やすために用意されています。これらの資料は、Oracle BPEL Process Manager とともに、`SOA_Oracle_Home\bpel\samples` ディレクトリにインストールされます。

関連資料： Oracle BPEL Process Manager とともに提供されているデモ、アクティビティと概念の参照資料およびチュートリアルの説明は、Oracle BPEL Process Manager 開発者ガイドの第 1 章を参照してください。

索引

A

ant

信用フローのチュートリアル用に実行, 2-2

assign アクティビティ

作成, 2-12, 2-13

説明, 2-12

B

BPEL

SOA の基礎, 1-2

説明, 1-2

「BPEL エラー」セクション

検証、コンパイルおよびデプロイ時のエラー, 2-14

BPEL ファイル

ソース・コードの確認, 2-5, 3-7

BPEL プロセス

テスト, 3-22

デバッグ, 3-15

Business Process Execution Language

「BPEL」を参照

D

Developer Prompt

起動, 1-8

I

invoke アクティビティ

作成, 2-11

説明, 2-11

O

Oracle BAM

レポートの表示, 3-21

Oracle BPEL Control

アクセス, 2-15

起動, 1-8

機能の概要, 1-4

説明, 1-4

レポートの表示, 3-20

Oracle BPEL Process Manager

起動, 1-8

機能の概要, 1-4

主要なコンポーネント, 1-3

説明, 1-3

Oracle BPEL Server

起動, 1-8

機能の概要, 1-5

Oracle JDeveloper

起動, 1-8

説明, 1-3

Oracle Web Services Manager, 3-24

S

scope アクティビティ

作成, 2-11

説明, 2-11

SOA

説明, 1-2

W

Web サービス

機能させるための処理, 1-2

Web ブラウザ設定

設定, 1-10

WSDL ファイル

ソース・コードの確認, 2-4, 3-10

あ

アプリケーション

作成, 2-3

え

エラー

検証、コンパイルおよびデプロイ時, 2-14

パートナー・リンクの作成時の解析エラー, 2-10

か

カスタム・ユーザー・インタフェース

BPEL プロセスを開始するために作成, 3-11

監査証跡

確認, 3-13

け

検証

プロセス, 2-14

こ

コンパイル

プロセス, 2-14

さ

サービス指向アーキテクチャ

「SOA」を参照

し

実行

プロセス, 2-15

た

ダッシュボード

レポート・ダッシュボードの作成, 3-19

「ダッシュボード」タブ

使用方法, 2-15

ち

チュートリアル

loan flow plus

Oracle Jdeveloper での作成, 3-3

概要, 3-2

監査証跡の確認, 3-13

進行中のプロセスのデバッグ, 3-15

パフォーマンス・チューニング, 3-18

非同期サービスとの対話, 3-6

非同期サービスの実行, 3-15

ヒューマン・タスクの確認, 3-8

ヒューマン・タスクの実行, 3-17

フォルト管理, 3-5

プロセスのライフサイクルの管理, 3-19

並列処理の確認, 3-8

レポートの作成と表示, 3-19

信用フロー

scope アクティビティ内での 2 つ目の assign アク

ティビティの作成, 2-13

scope アクティビティ内での assign アクティビ

ティの作成, 2-12

scope アクティビティ内での invoke アクティビ

ティの作成, 2-11

scope アクティビティの作成, 2-11

アプリケーションとプロジェクトの作成, 2-3

概要, 2-2

検証、コンパイルおよびデプロイ, 2-14

サービスの起動とテスト, 2-2

実行, 2-15

使用方法, 2-2

ネームスペースの使用, 1-10

パートナ・リンクの作成, 2-9

変数の自動作成, 2-11

チューニング

パフォーマンス, 3-18

て

テスト

BPEL プロセス, 3-22

デバッグ

進行中のプロセス, 3-15

デプロイ

プロセス, 2-14

ね

ネームスペース

説明, 1-10

チュートリアルでの使用, 1-10

は

パートナ・リンク

信用格付けサービス用に作成, 2-9

説明, 2-9

パートナ・リンクの作成時の解析エラー, 1-10

パートナ・リンクの作成時の解析エラー, 1-10

パフォーマンス

チューニング, 3-18

ひ

非同期サービス

対話, 3-6

ヒューマン・タスク

確認, 3-8

実行, 3-17

ふ

フォルト

管理, 3-5

プロジェクト

信用フローのチュートリアル用に作成, 2-3

プロセス

テスト, 3-22

プロセスのライフサイクル

管理, 3-19

へ

並列処理

確認, 3-8

変数

自動作成, 2-11

れ

レポート

BAM のレポートの表示, 3-21

Oracle BPEL Control のレポートの表示, 3-20

レポート・ダッシュボードの作成, 3-19