



# Java開発者に贈る SOA/BPEL開発の最新動向

日本オラクル株式会社 システム事業推進本部

テクノロジー・エバンジェリスト

佐藤 直生

Naoki.SATO@oracle.com

ORACLE

# お願い

- 後ほど、デモンストレーションを行うので、前方にお座りください!

# スピーカー紹介

- 日本オラクルのJ2EE/SOAのテクノロジー・エバンジェリストとして活動
  - Oracle Application Server (OC4J、TopLink ...)
  - Oracle JDeveloper
  - Oracle BPEL Process Manager、Oracle ESB ...
- 執筆、監訳、講演、プロジェクト支援などの活動を行う
- 稚内北星学園大学の客員助教授として、社会人向けIT教育にも携わる
- blog: <http://d.hatena.ne.jp/satonaoki/>
- mail: [Naoki.SATO@oracle.com](mailto:Naoki.SATO@oracle.com)

# セッション概要

- BPELやJBIといったSOAを実現するためのテクノロジーの動向を、Java開発者の観点から考えます
- さらに、今すぐ使える無償のSOA開発ツール「Oracle JDeveloper 10g」を用いた、WebサービスやBPELプロセスの開発についてもご紹介します

# アンケート

- 挙手をお願いします!

# Agenda

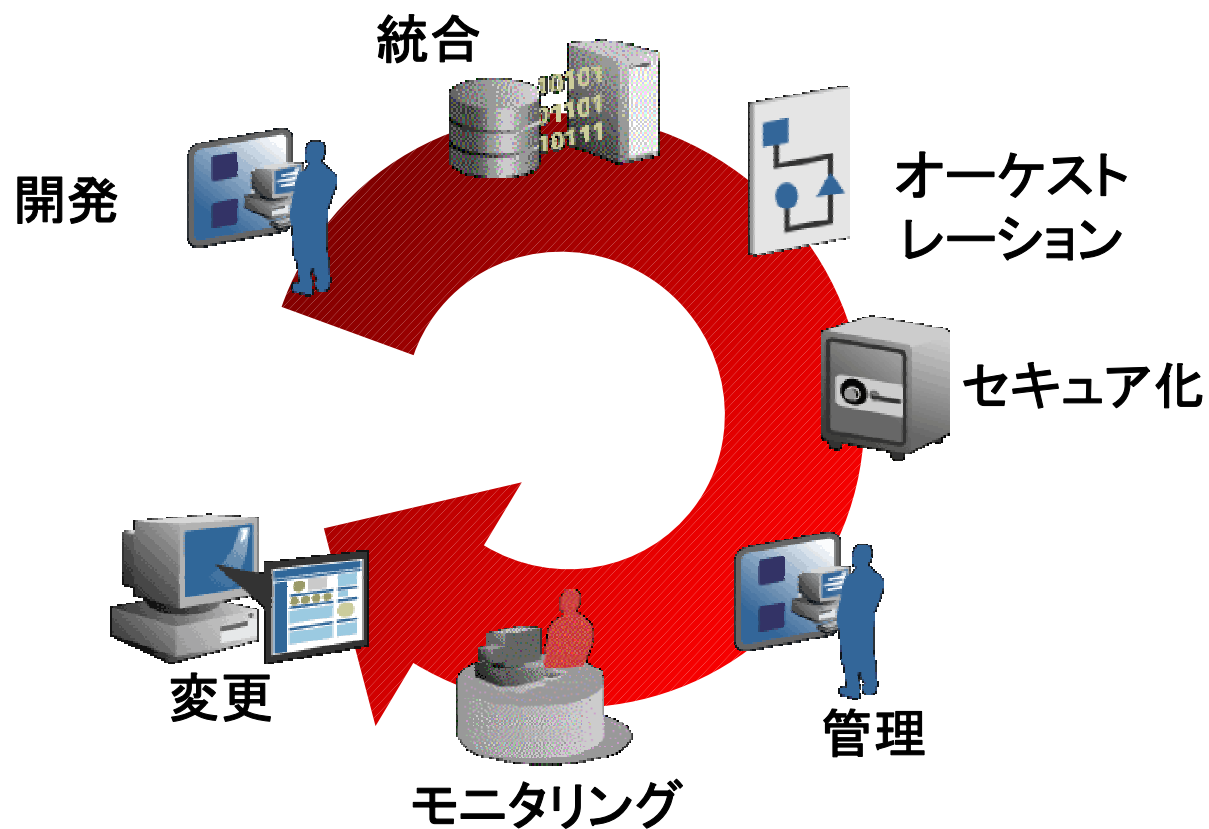
## SOAとそれを支えるテクノロジーの動向

- SOAとは?
- Webサービス
- BPMとBPEL
- ESBとJBI
- Oracle Fusion Middleware
  - Oracle Fusion Architecture
  - OC4J / OracleAS Web Services
  - Oracle BPEL Process Manager
  - Oracle JDeveloper
- デモンストレーション

# 現在の課題

- 業務要件の変更
- 新技術の登場
- 増大する複雑さ
- 変更に対応できるアプリ設計

# SOA – 成功のためのブループリント





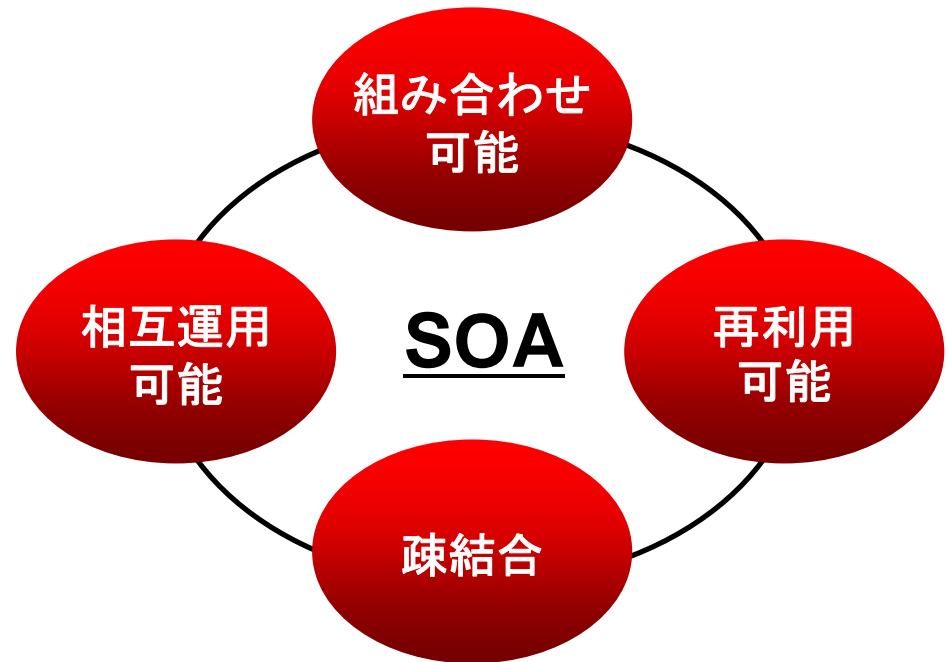
# SOA(サービス指向アーキテクチャ)の定義

サービス指向アーキテクチャ (SOA) は、  
相互作用するソフトウェア・エージェント間の  
疎結合の達成を目的とする、  
アーキテクチャのスタイルです。  
サービスは、サービス・コンシューマの  
ための最終結果を達成するために、  
サービス・プロバイダが行う作業の単位です。

*Dr. Hao He, W3C Web Services Architecture Working Group*

# SOAの特徴

- サービスが自己記述的な  
インタフェースを持つ
- メッセージが明確に定義  
されている
- サービスが発見可能である
- サービスの記述と相互作用  
にプラットフォーム非依存の  
標準 (XML) が使われる
- サービスが任意のプラット  
フォームで提供可能



# SOAの目的

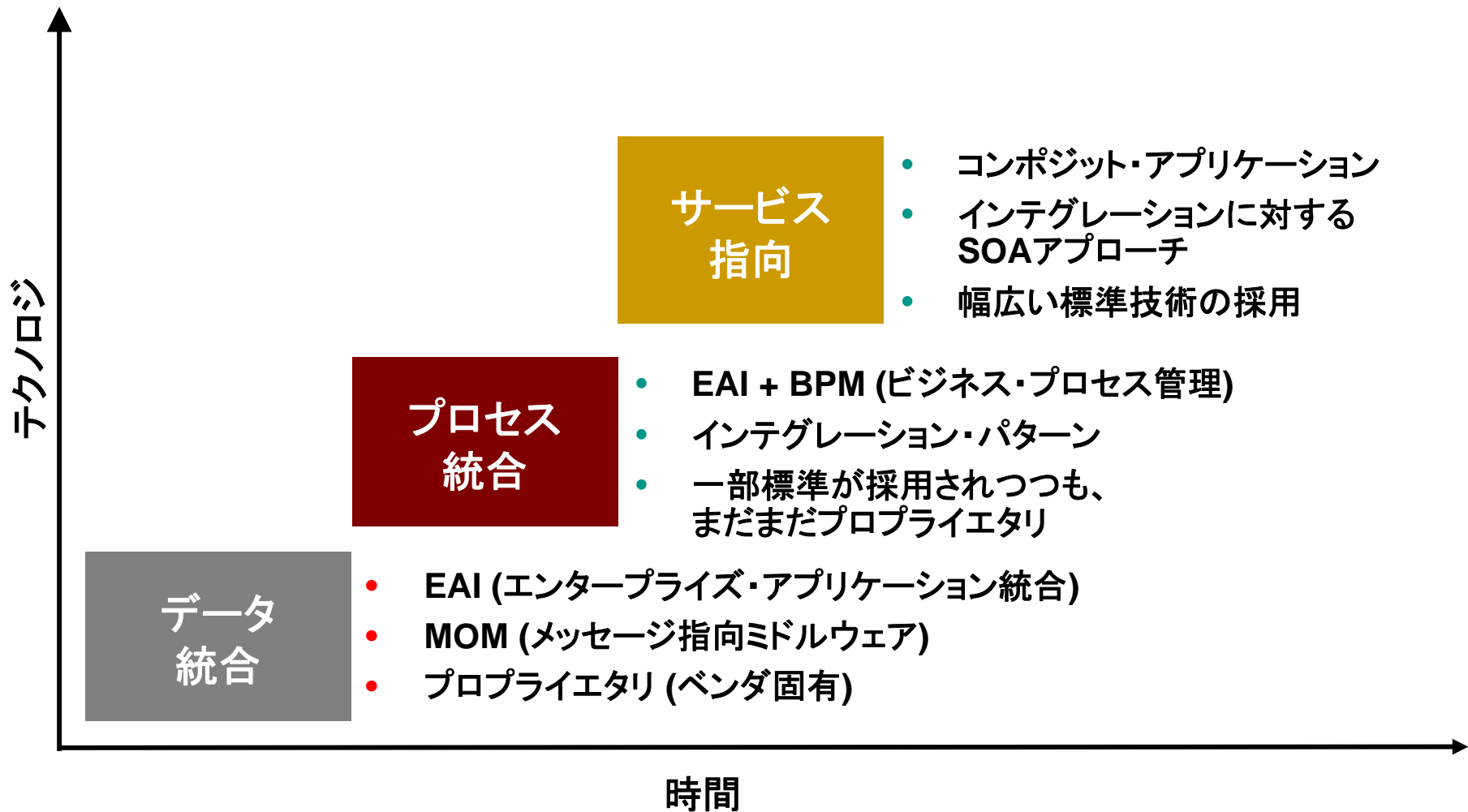
- 疎結合の「サービス」を組み合わせることで、柔軟に低コストでアプリケーションを構築
  - 統合可能な新規アプリケーションを開発
  - 既存アプリケーションからアプリケーションを構築



# SOAの「サービス」と Webサービス

- SOAにおける「サービス」は、必ずしも (SOAPを使う) Webサービスである必要はない
- 疎結合を実現するためのインターフェースとして、WSDLは極めて重要
- Webサービス  $\subset$  SOAの「サービス」

# インテグレーションの進展

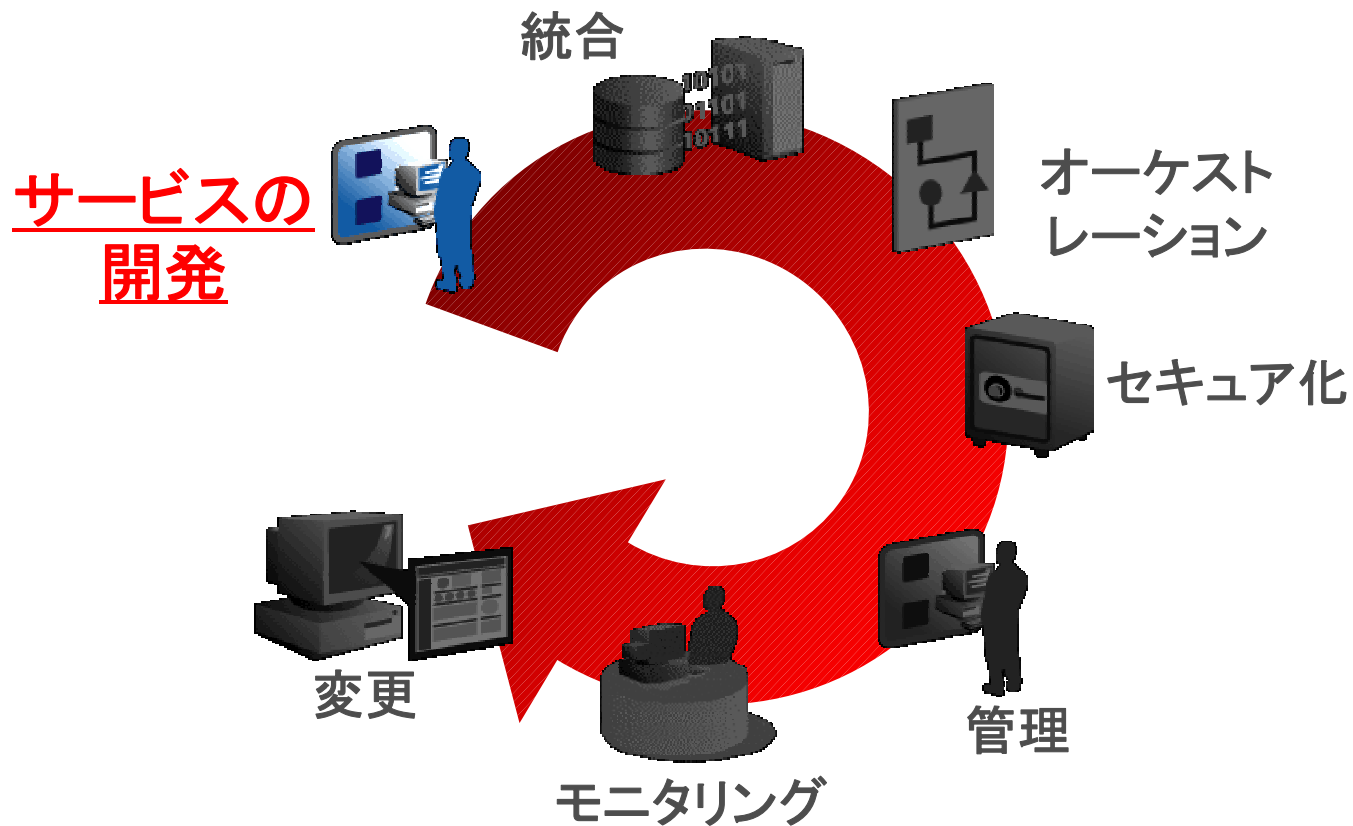


# Agenda

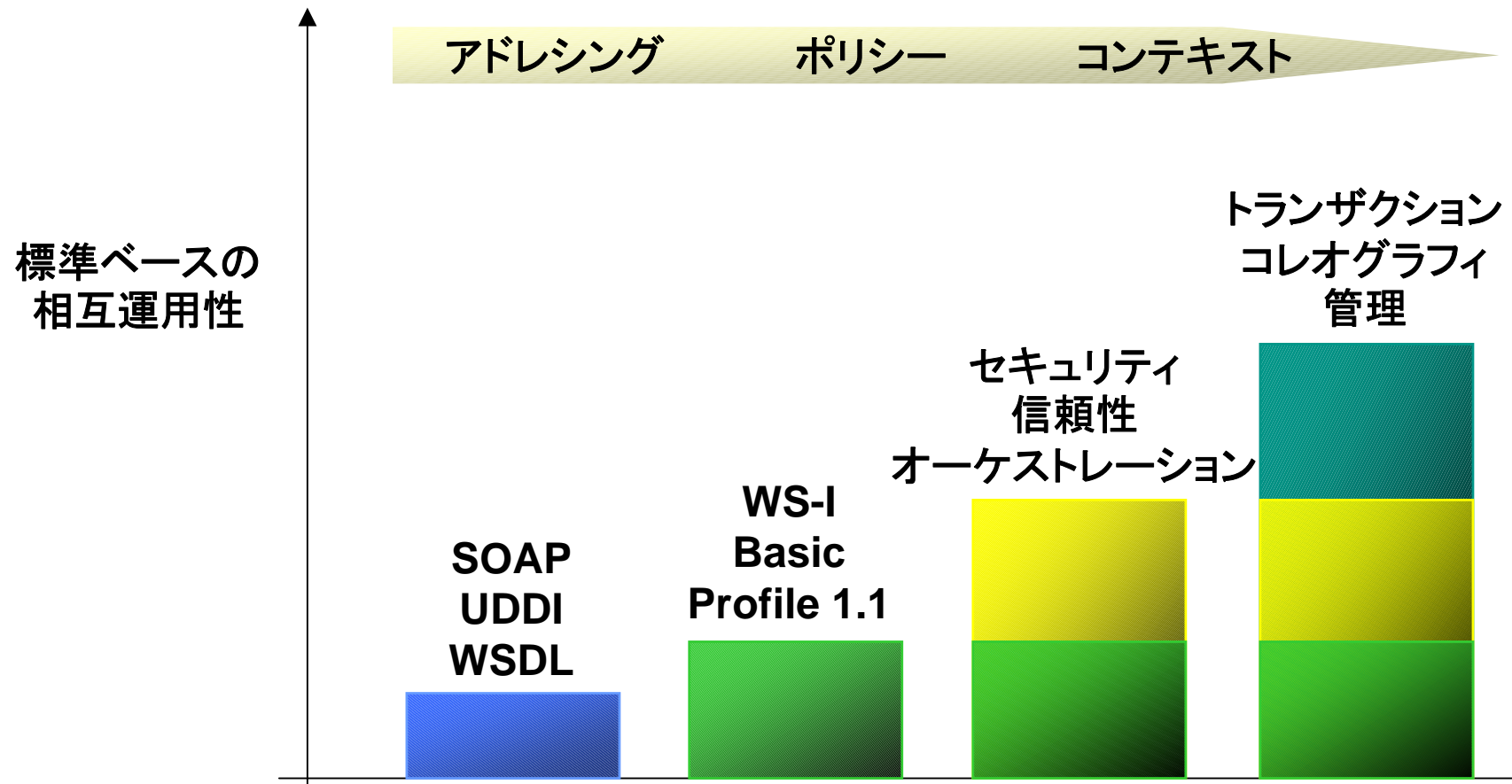
- SOAとそれを支えるテクノロジーの動向
  - SOAとは?
  - Webサービス
  - BPMとBPEL
  - ESBとJBI
- Oracle Fusion Middleware
  - Oracle Fusion Architecture
  - OC4J / OracleAS Web Services
  - Oracle BPEL Process Manager
  - Oracle JDeveloper
- デモンストレーション



# SOA – サービスの開発



# Webサービス標準の進展

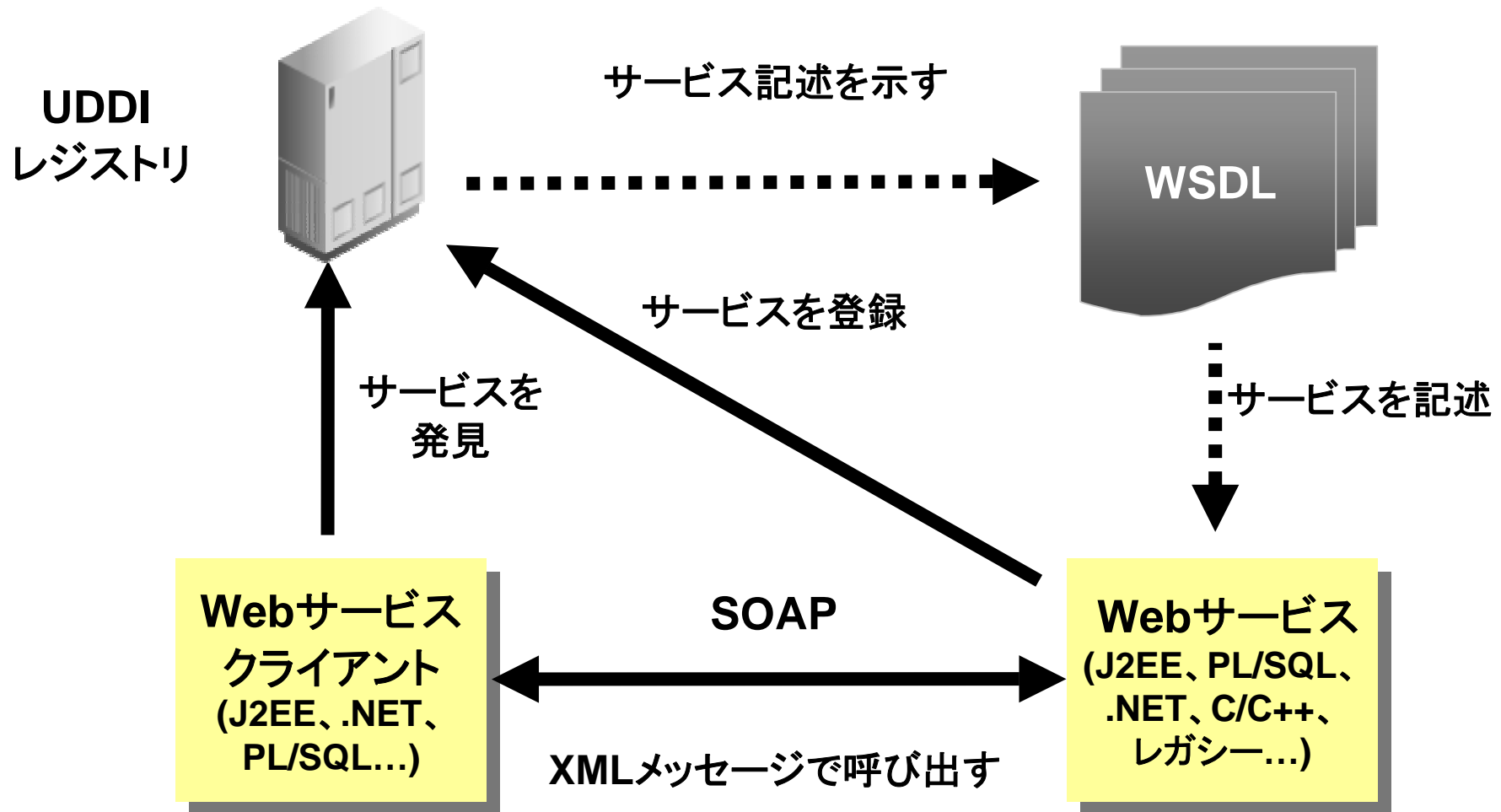




# 基本的なWebサービス標準 (2000-2003年頃)



# 基本的なWebサービス実装



# J2EE 1.4 – JavaにおけるWebサービス標準

API	説明
JAXP	Java API for XML Parsing、XMLパーサ/XSLTトランスフォーマ
JAXB	Java API for XML Data Binding、O/Xマッピング
JAX-RPC	Java API for XML Remote Procedure Call、Webサービス
SAAJ	SOAP API for Attachments in Java、SOAP
JAXR	Java API for XML Registries、UDDIなどのレジストリ
EJB 2.1	ステートレスSession Beanのエンドポイント・モデル
JSR-109	Webサービスのデプロイメント・モデル

# Java EE 5

- JAX-WS (Java API for XML Web Services 2.0)
  - JAX-RPC 1.1の後継
- EJB 3.0
  - 複雑なEJB 2.1を改良し、アノテーションを活用してEoD (開発の容易さ) を実現
  - EJBをPOJO / POJI (普通のJavaクラスやJavaインタフェース) として作成
  - 「Java Persistence API」でPOJOのO/Rマッピングを標準化
- Web Services Metadata 1.1
  - アノテーションでWebサービスを定義

# EJB 3.0のWebサービス

```
@Stateless(name="Hello")
@WebService(name = "MyWebService1",
    serviceName = "MyWebService1",
    targetNamespace = "http://project1/")
@SOAPBinding(style = SOAPBinding.Style.DOCUMENT,
    use = SOAPBinding.Use.LITERAL,
    parameterStyle = SOAPBinding.ParameterStyle.WRAPPED)
public class HelloBean implements Hello {

    @WebResult(name = "result",
        targetNamespace = "http://project1/")
    @WebMethod(operationName = "hello",
        action = "")
    public String hello(
        @WebParam(name = "name",
            mode = WebParam.Mode.IN,
            targetNamespace = "http://project1/",
            header = false)
            String name) {
        return "こんにちは、" + name + "さん";
    }
}
```

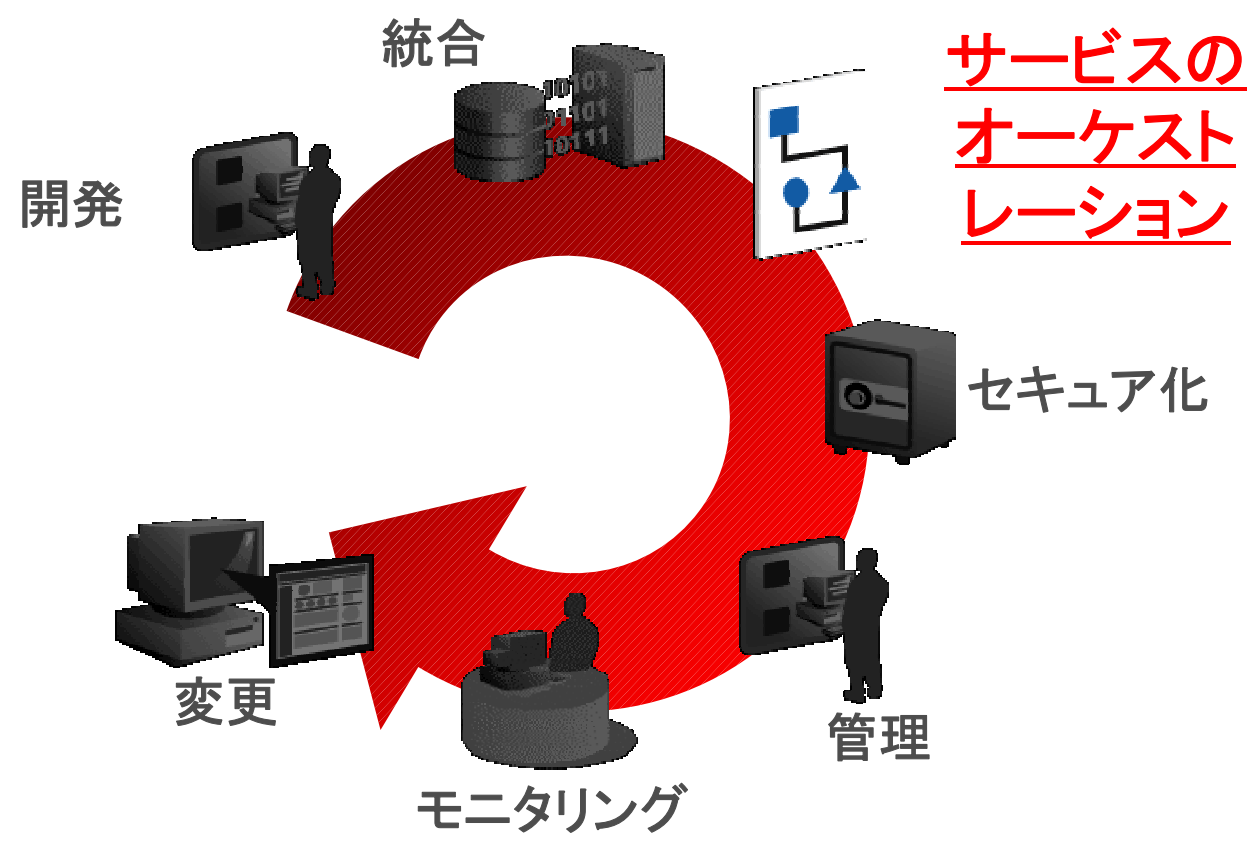
## @Remote

```
public interface Hello {
    String hello(String name);
}
```

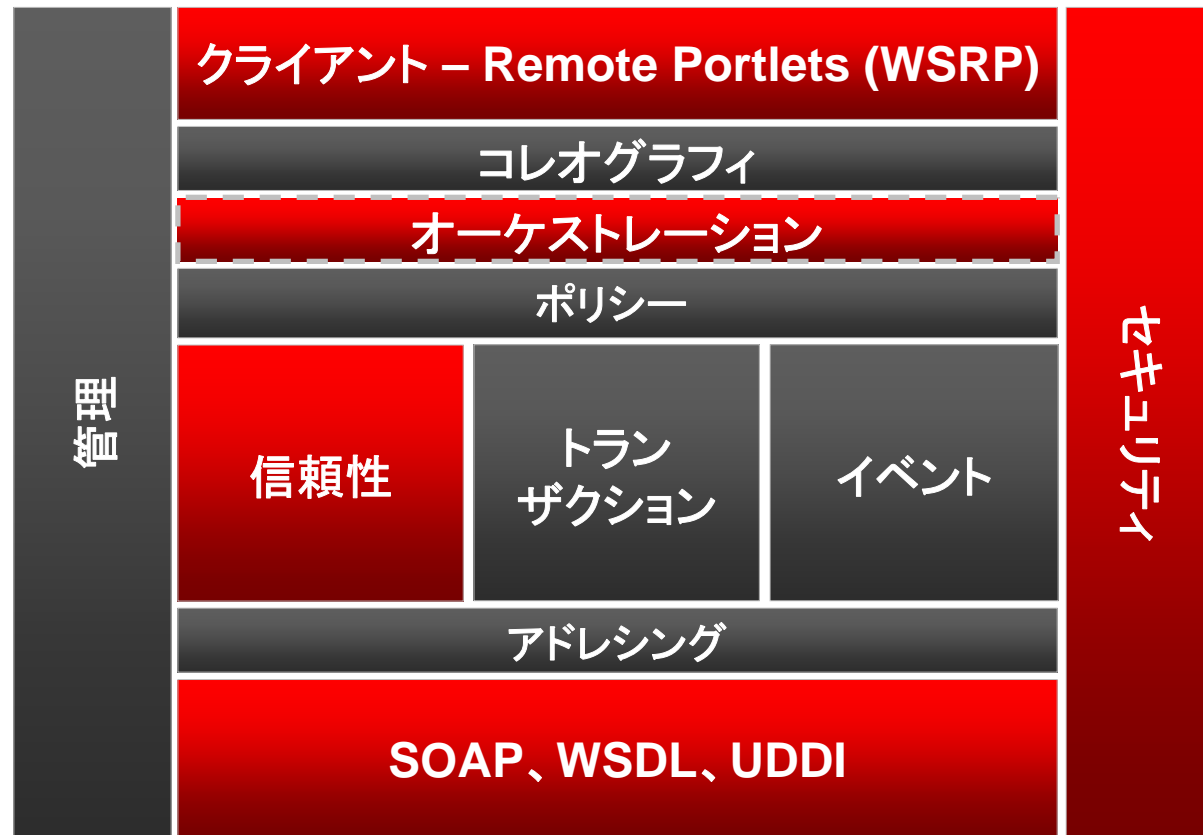
# Agenda

- SOAとそれを支えるテクノロジーの動向
  - SOAとは?
  - Webサービス
  - - BPMとBPEL
  - ESBとJBI
- Oracle Fusion Middleware
  - Oracle Fusion Architecture
  - OC4J / OracleAS Web Services
  - Oracle BPEL Process Manager
  - Oracle JDeveloper
- デモンストレーション

# SOA – サービスのオーケストレーション



# Webサービス標準(2004-2005年頃)

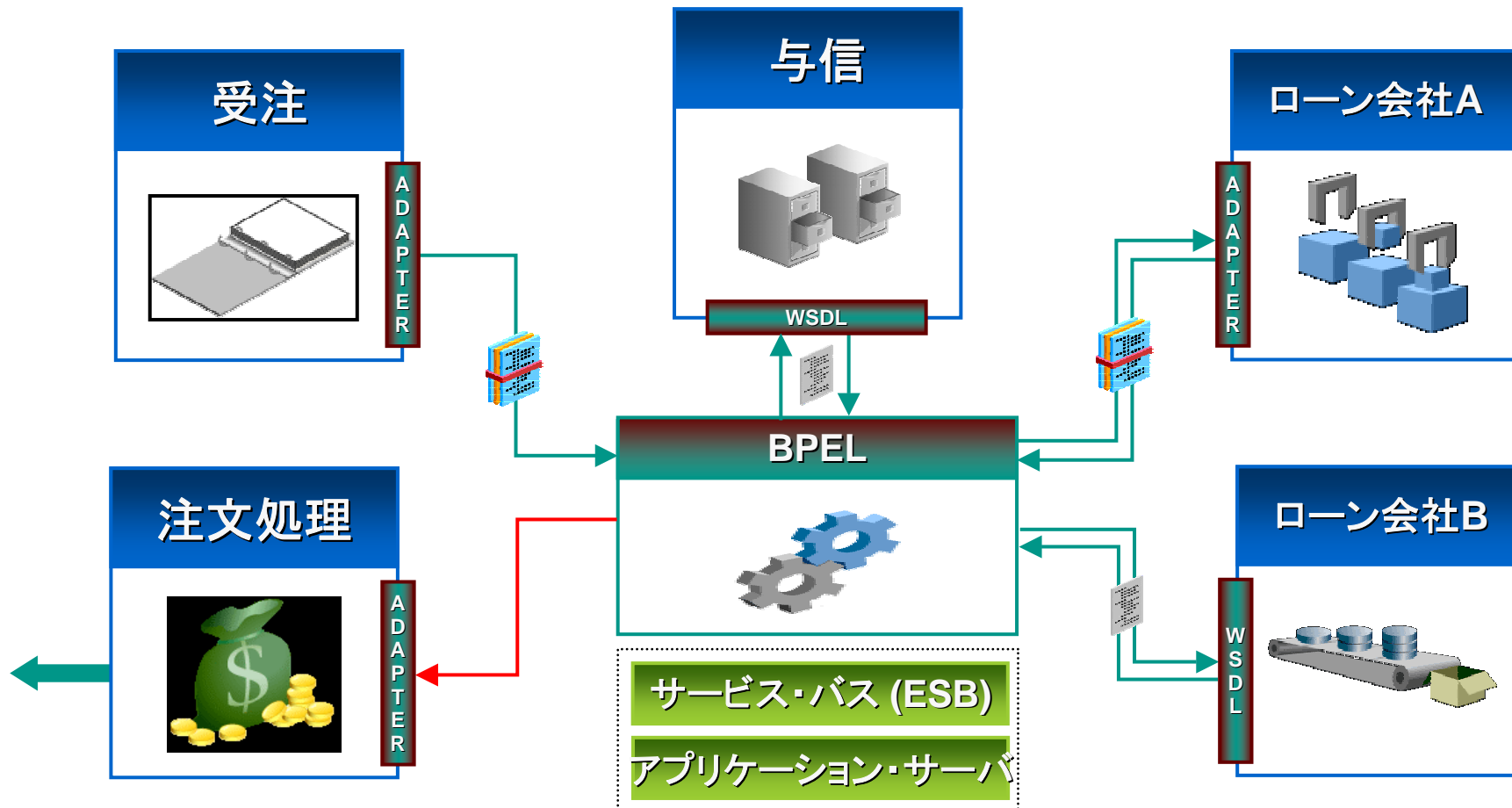


セキュリティ

管理



# サービスのオーケストレーション



# ビジネス・プロセスの課題

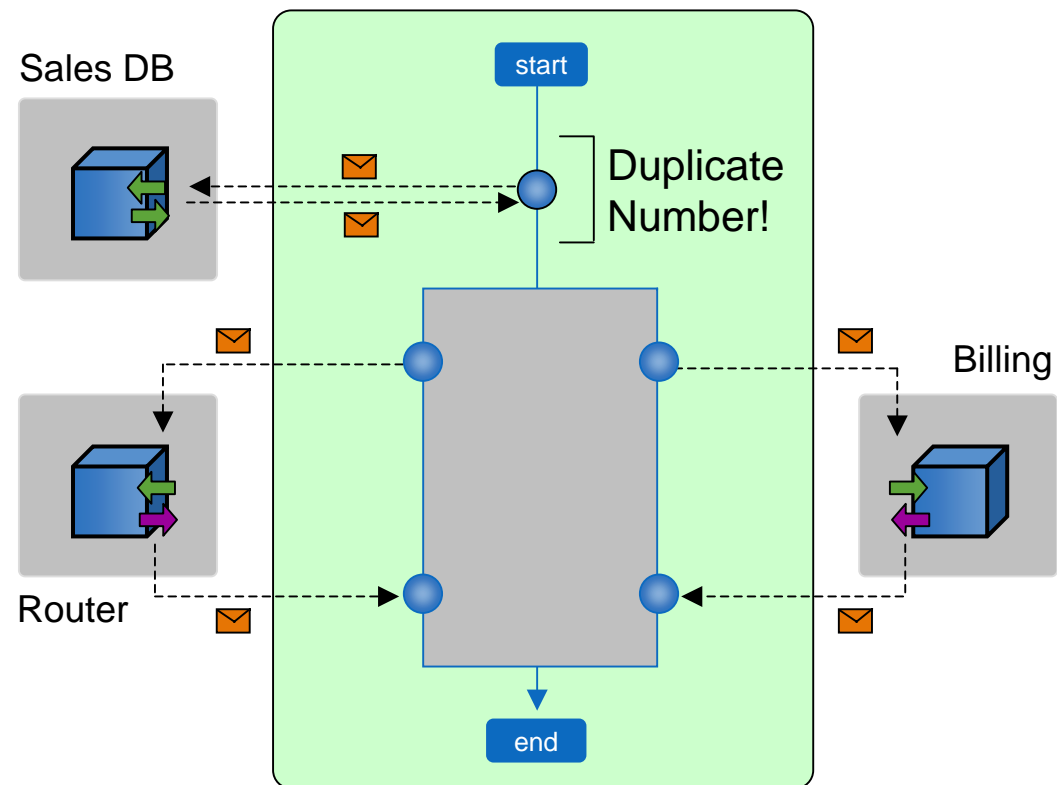
- サービス間の非同期通信の調整
- サービス間のメッセージ交換の相関
- 並列処理の実装
- ...
- 相互作用時のデータの操作/変換
- 長時にわたるビジネス・トランザクションのサポート
- 一貫した例外処理の提供
- ...

# BPEL

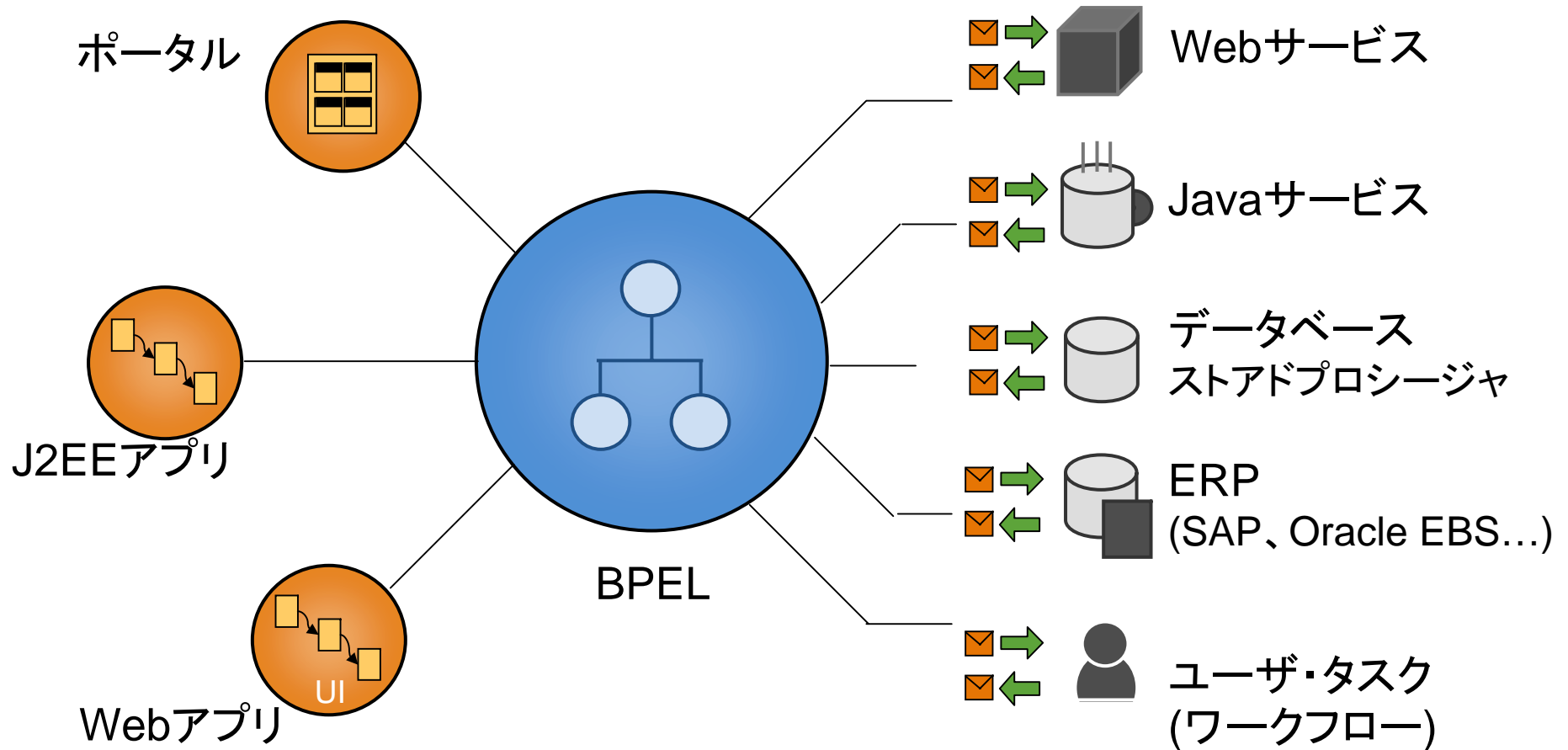
## (Business Process Execution Language)

個別のサービスを組み立て、エンド・トゥ・エンドのプロセス・フローを構築するためのXMLベースのマークアップ言語

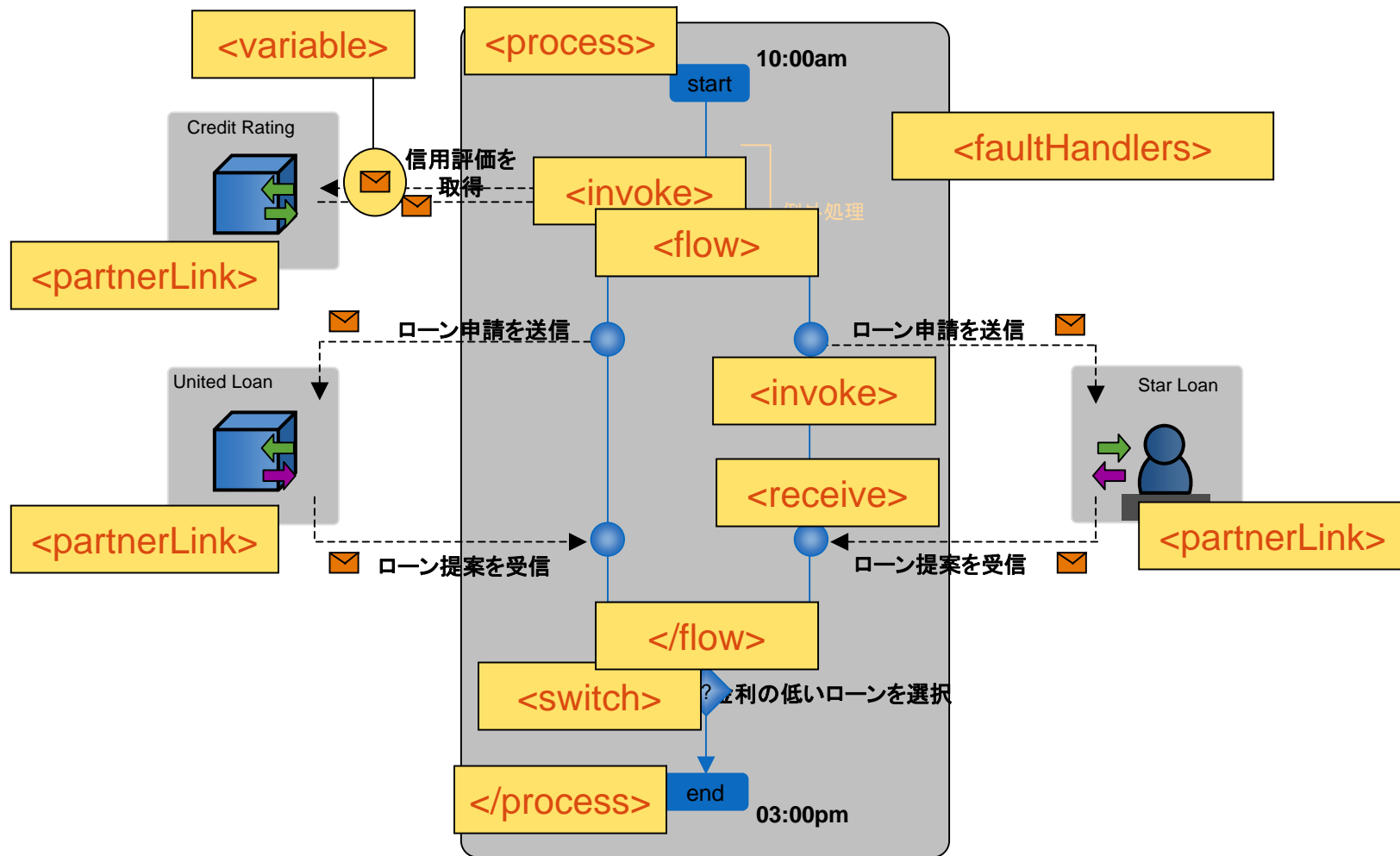
- OASISで標準策定中
- (WS-BPEL 2.0)
- 成熟したビジネス・プロセス定義言語
  - WSFLとXLANGに由来
- 業界での幅広い支持
  - Oracle、IBM、MS、BEA、SAP...
- ベンダ・ロックインを回避
- 再利用と相互運用性



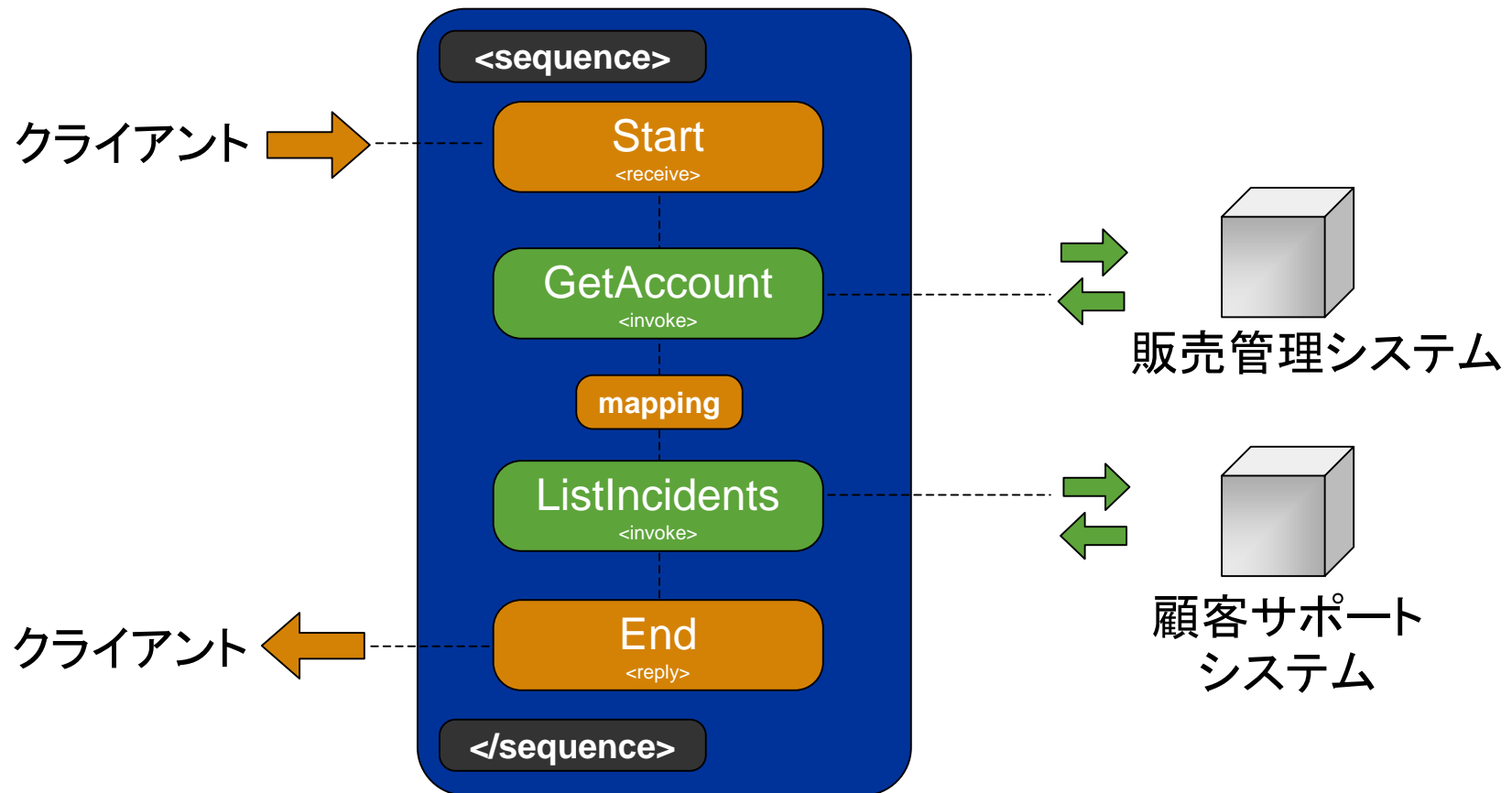
# BPELによるオーケストレーション



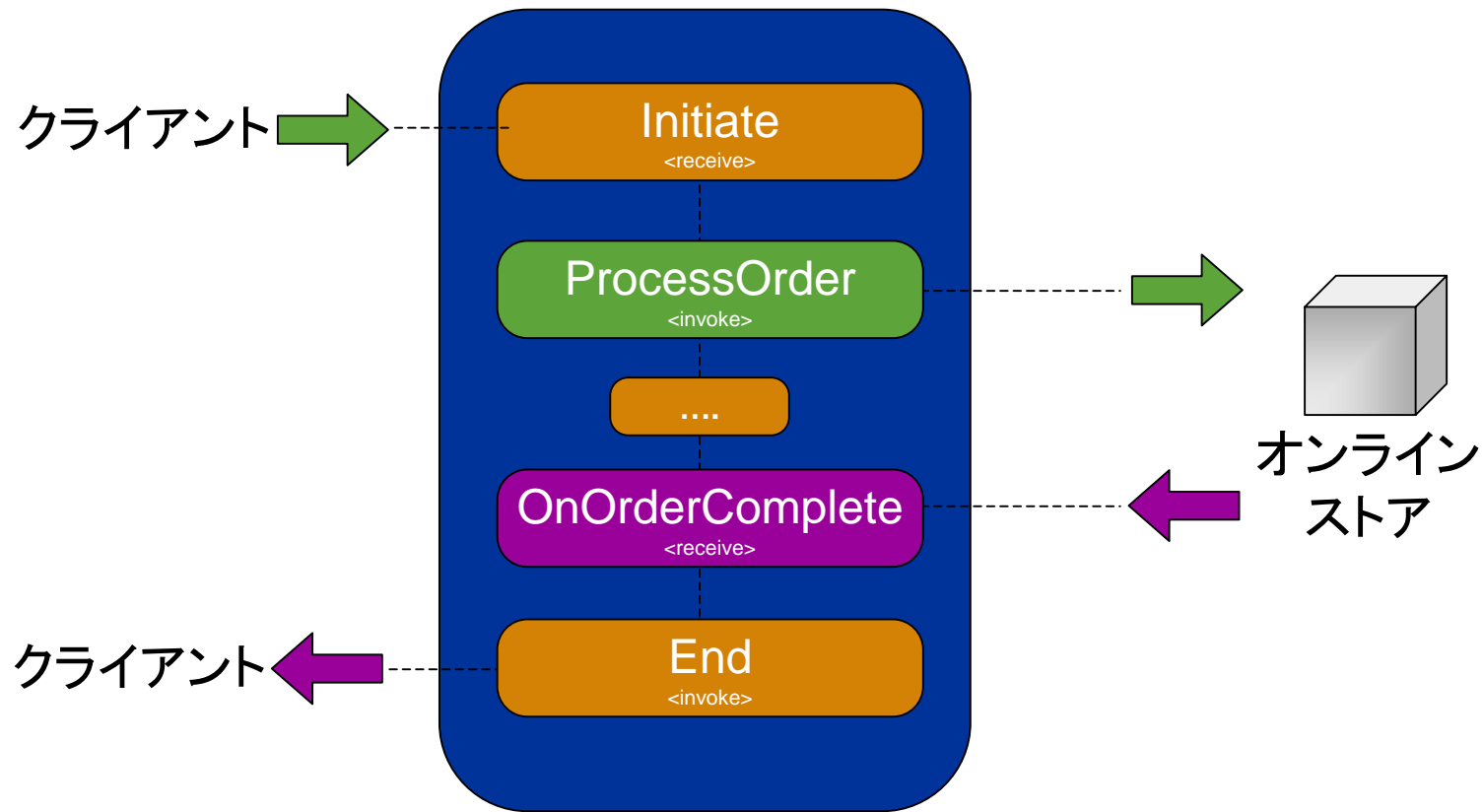
# BPELプロセスの例



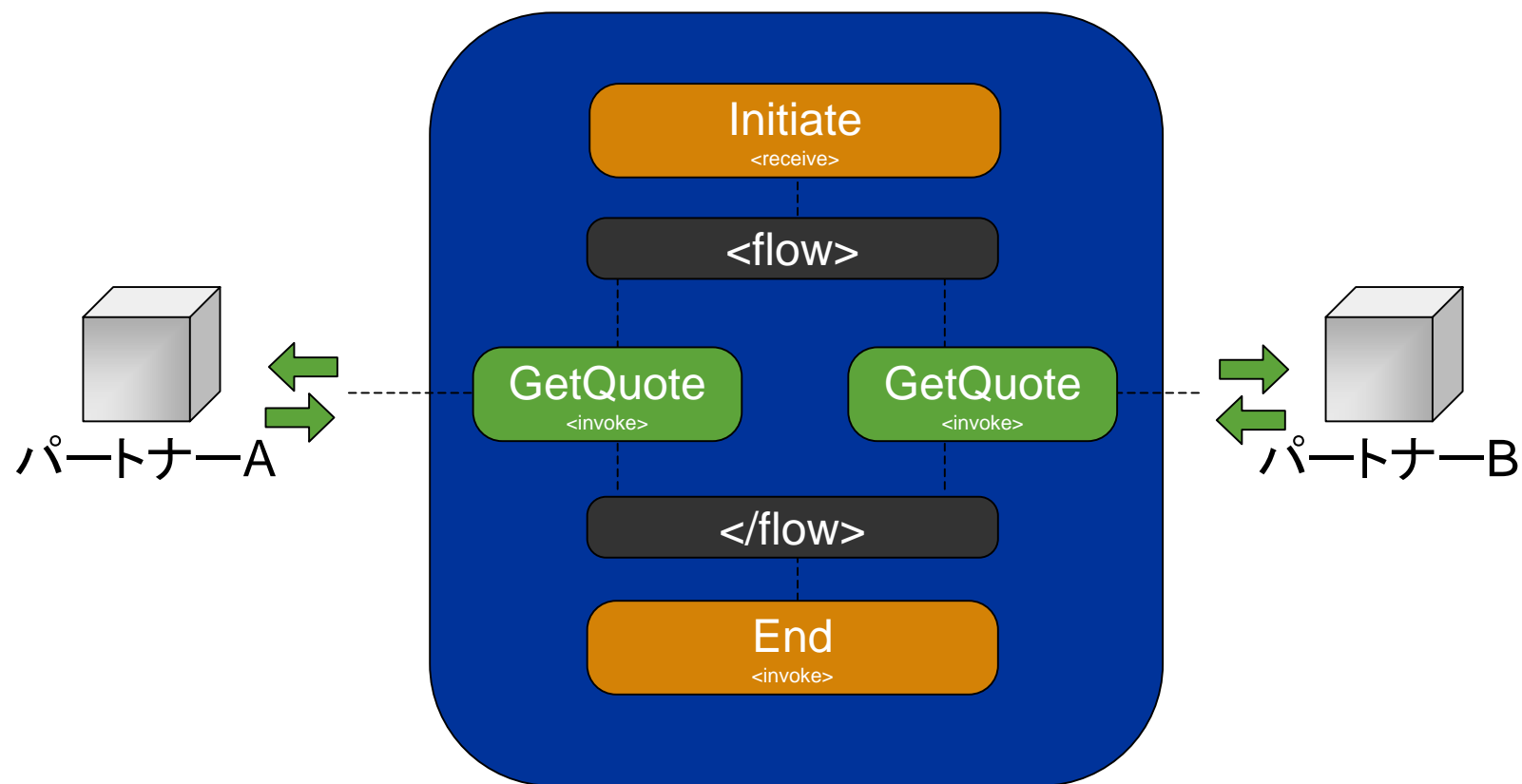
# BPELプロセス(1) 実行のチェーン



# BPELプロセス(2) 非同期



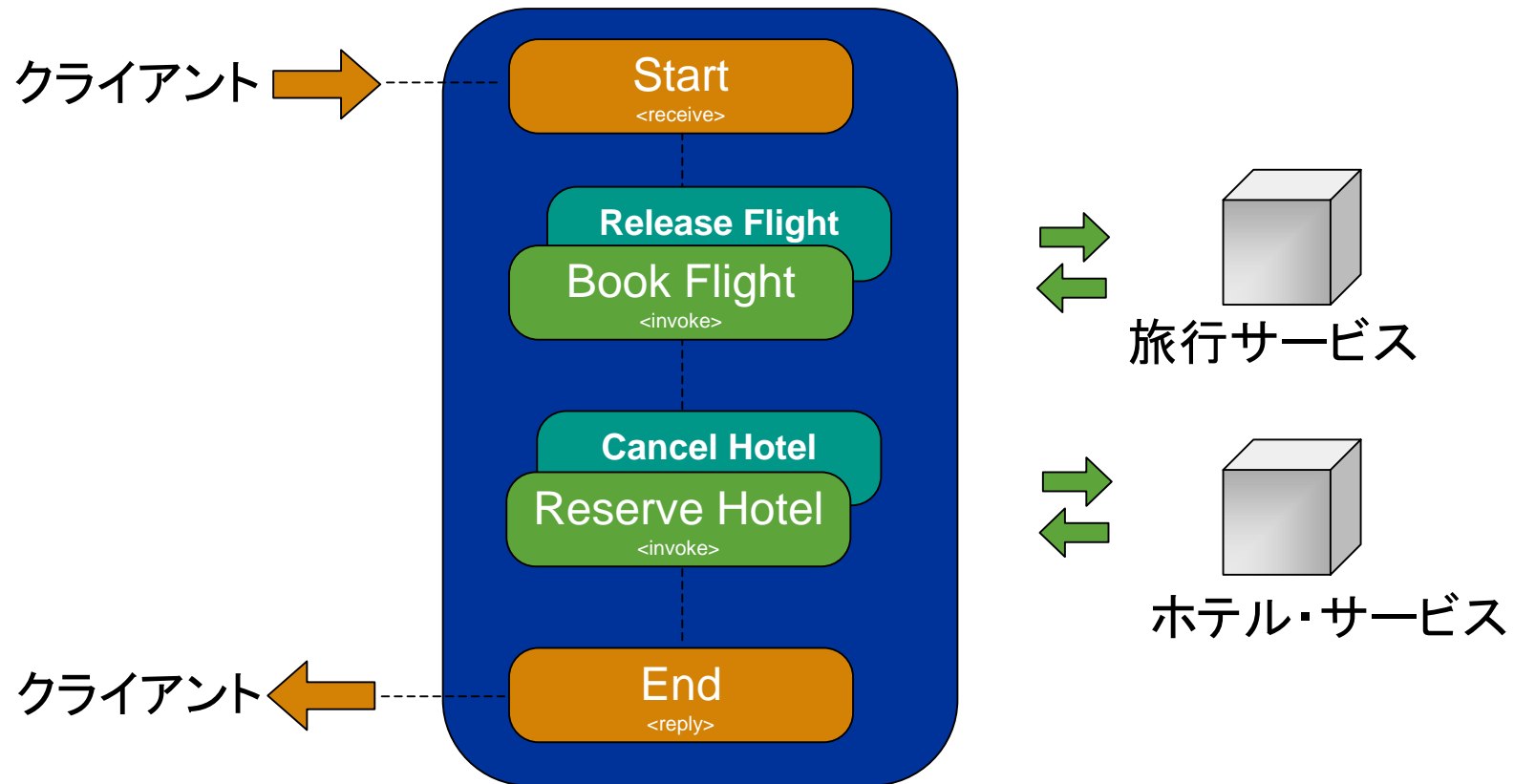
# BPELプロセス(3) 並行処理



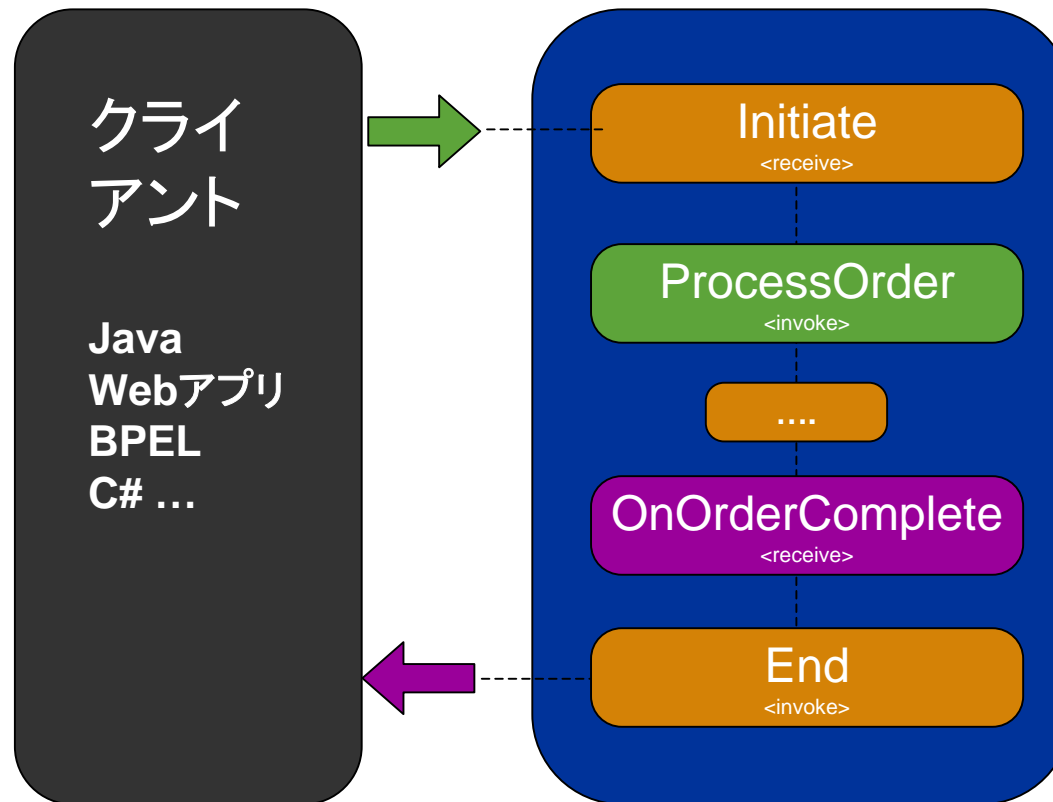


# BPELプロセス(4)

## 例外管理と補償



# BPELプロセス(5) プロセス自体も一つのサービス

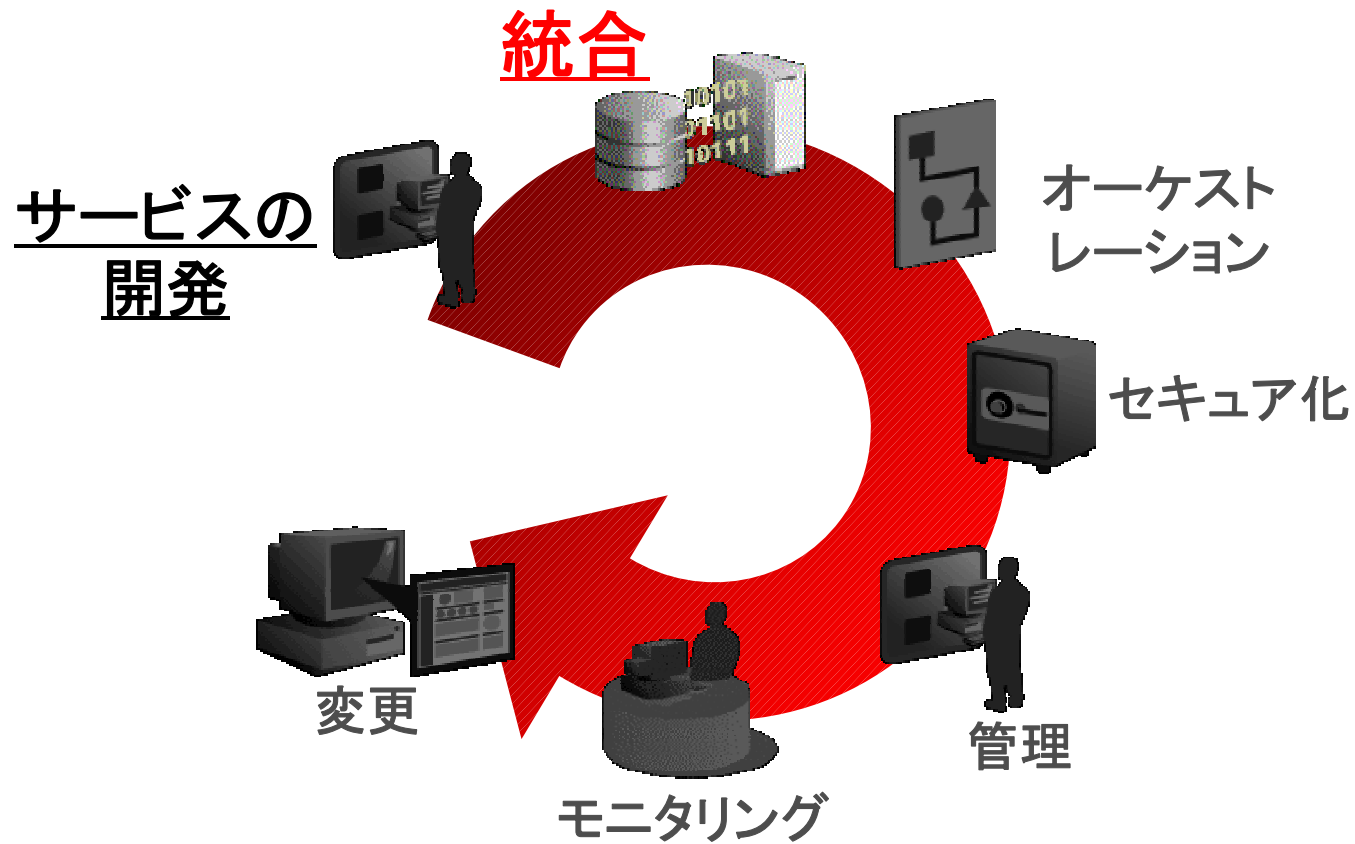


# Agenda

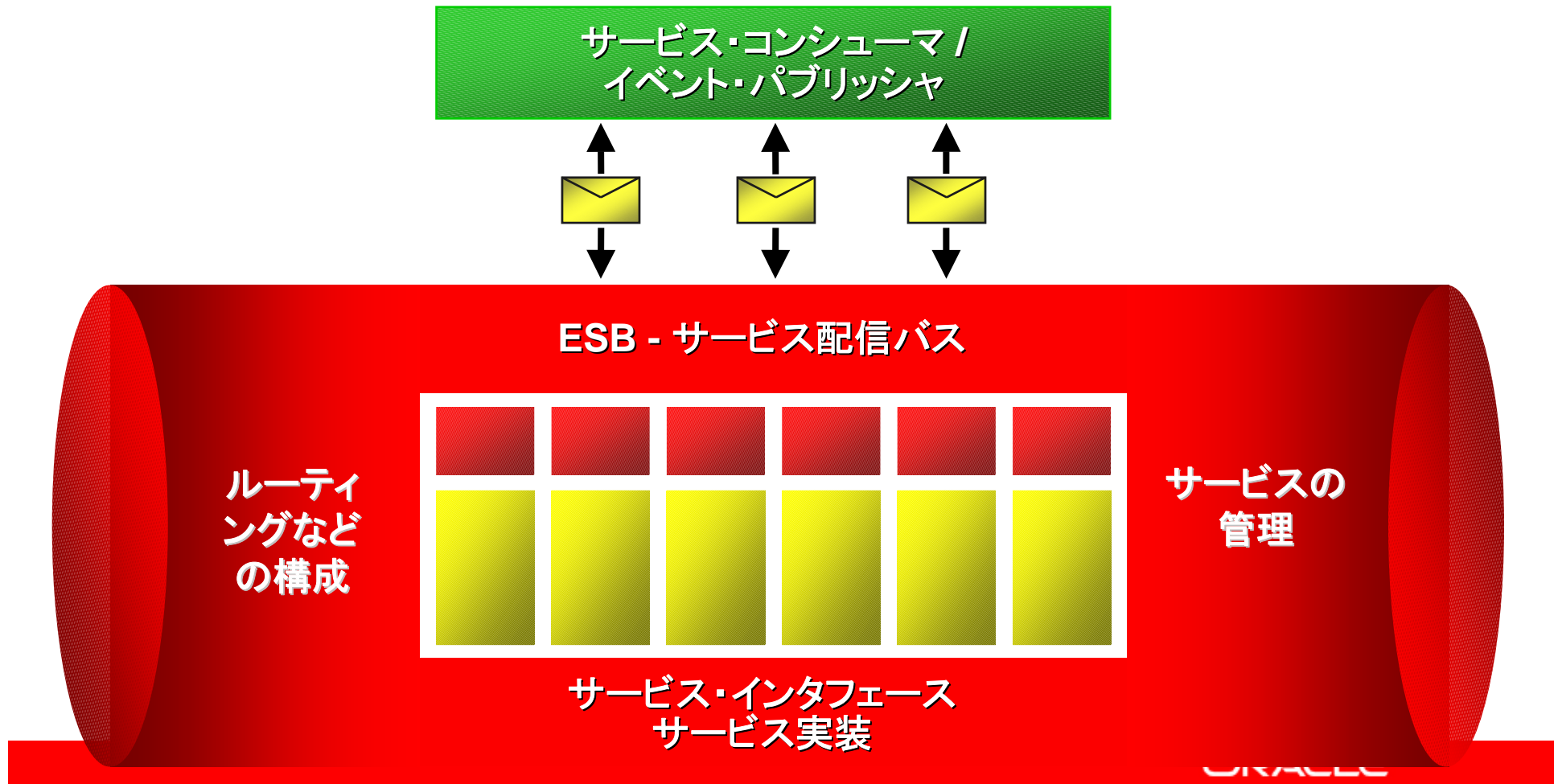
- SOAとそれを支えるテクノロジーの動向
  - SOAとは?
  - Webサービス
  - BPMとBPEL
  - ESBとJBI
- Oracle Fusion Middleware
  - Oracle Fusion Architecture
  - OC4J / OracleAS Web Services
  - Oracle BPEL Process Manager
  - Oracle JDeveloper
- デモンストレーション



# SOA – 統合



# ESB – 仮想的なサービス・バス



# ESBの主な機能

## モニタリング/構成管理



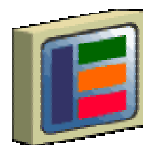
### 接続

アダプタ  
イベント  
メタデータ  
セキュリティ  
サービス管理



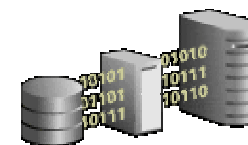
### 加工

変換  
値マッピング  
相互参照  
ビジネス・ルール



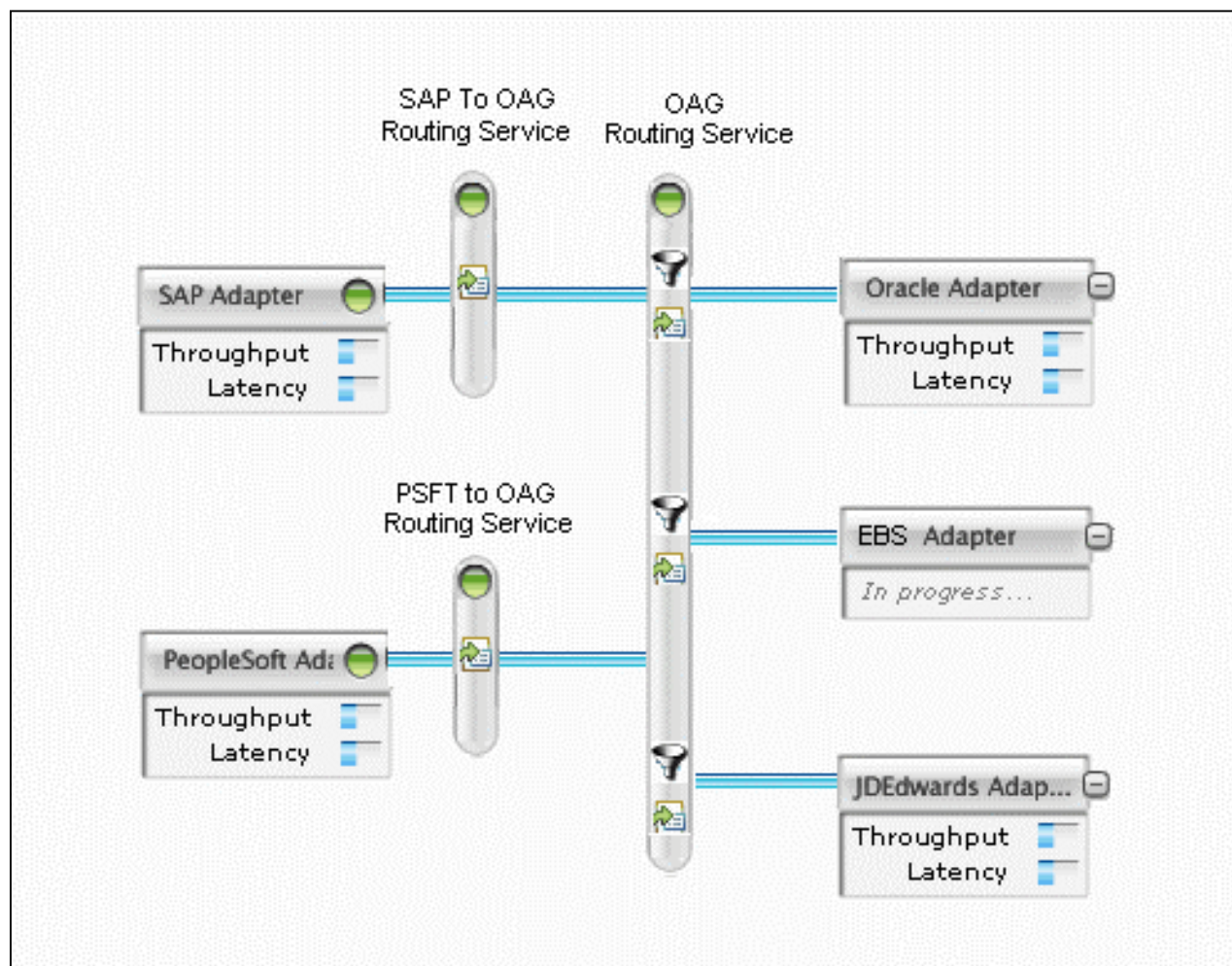
### 分散

ルーティング  
プロトコル・バス  
メッセージング  
ゲートウェイ  
相互運用性



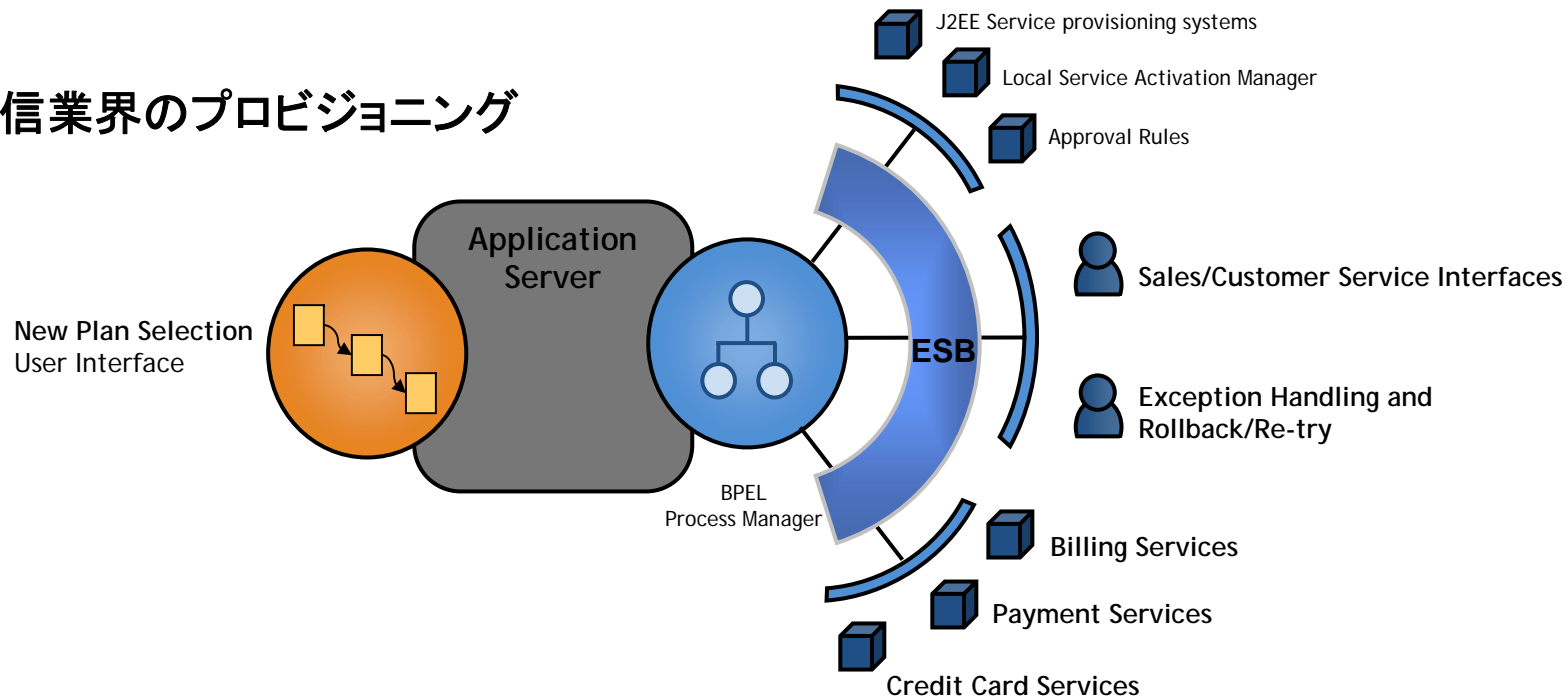
ORACLE

# ESBにおけるルーティング



# BPEL & ESBのユース・ケース

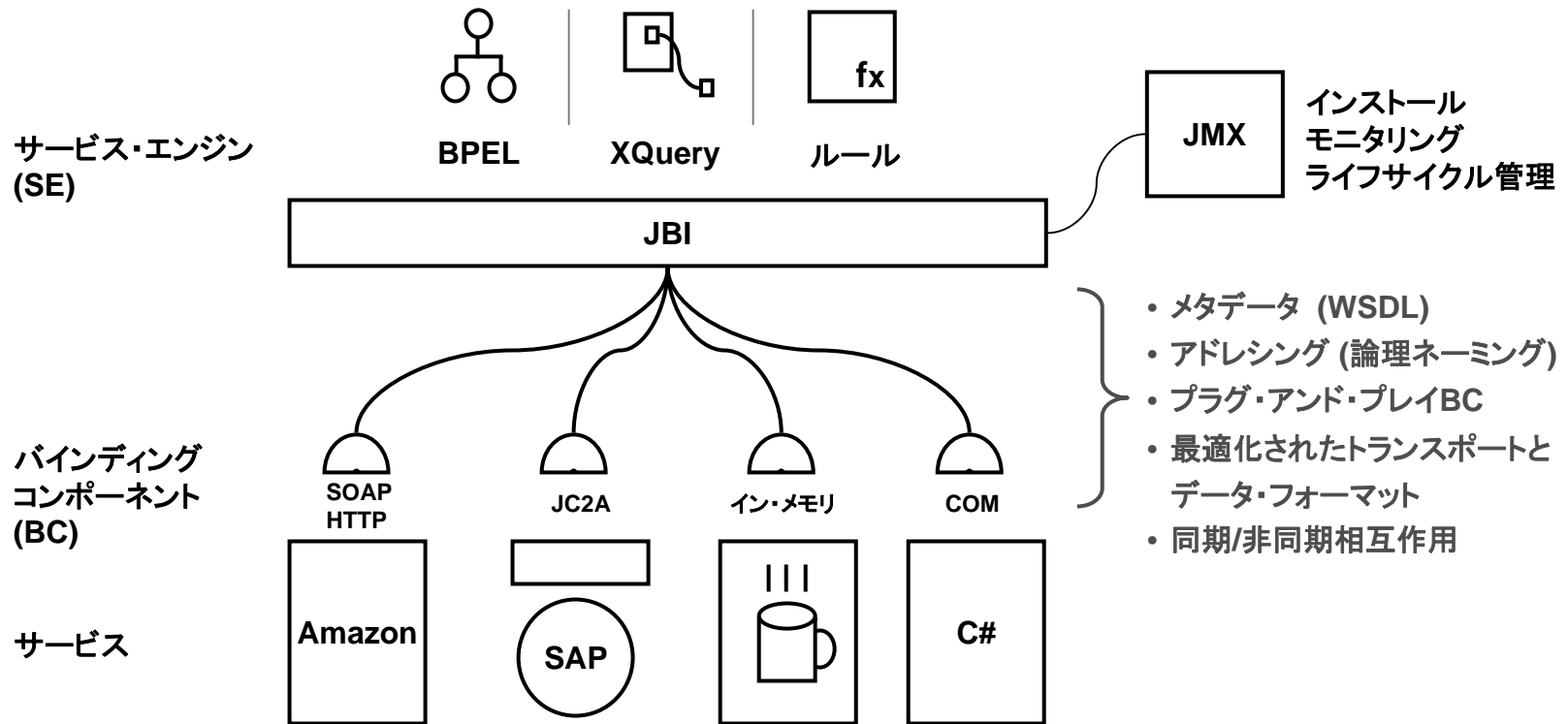
## 通信業界のプロビジョニング



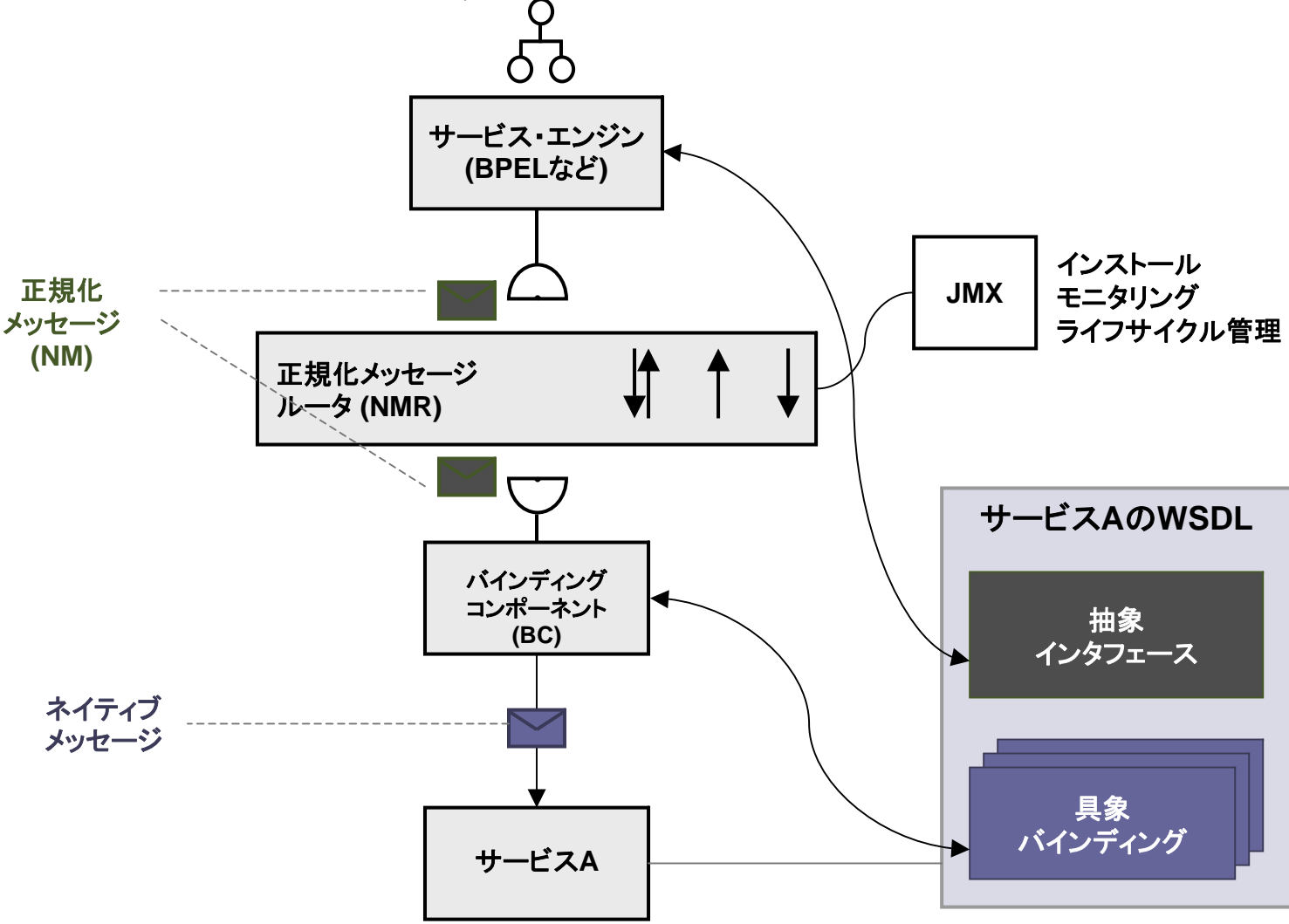
ESBが、アダプタと変換サービスをBPELから分離  
BPELは、オーケストレーションに集中



# JBIとは?



# JBIにおけるプラグ・アンド・プレイ



# BPELとJBI: 補完的な概念

BPEL	JBI
ビジネス・プロセスと その実行に着目	ビジネス上の セマンティクスには 着目しない
サービス自体をホスト せず、メッセージの 仲介もしない	サービスをホストし、 サービス間の メッセージを仲介

# Agenda

- SOAとそれを支えるテクノロジーの動向
  - SOAとは?
  - Webサービス
  - BPMとBPEL
  - ESBとJBI



## Oracle Fusion Middleware

- Oracle Fusion Architecture
- OC4J / OracleAS Web Services
- Oracle BPEL Process Manager
- Oracle JDeveloper
- デモンストレーション



## Oracle Fusion Architecture (OFA)

- ✓ モデル駆動 (Model driven)
- ✓ SOA (サービス指向) /  
EDA (イベント駆動) 対応  
(Service & Event enabled)
- ✓ 標準準拠 (Standards-based)
- ✓ 情報中心 (Information Centric)
- ✓ グリッド対応 (Grid Ready)

# ORACLE FUSION MIDDLEWARE

## コラボレーティブ エンタープライズ ポータル



ポータル、コラボレーション、モバイル、デスクトップ、検索

## 開発ツール



モデリング  
開発ツール  
フレームワーク

## コンポジット アプリケーション プロセス オーケストレーション



BPM、ESB、B2B

## 情報の集約と分析



ETL、ハブ、コンテンツ管理、BI、BAM

## 管理ツール



システム管理

## エンタープライズ アプリケーション サーバ



J2EE、WS-\*、イベント、メタデータ、レジストリ

## セキュリティ



アイデンティティ管理  
サービス管理

## グリッド コンピューティング

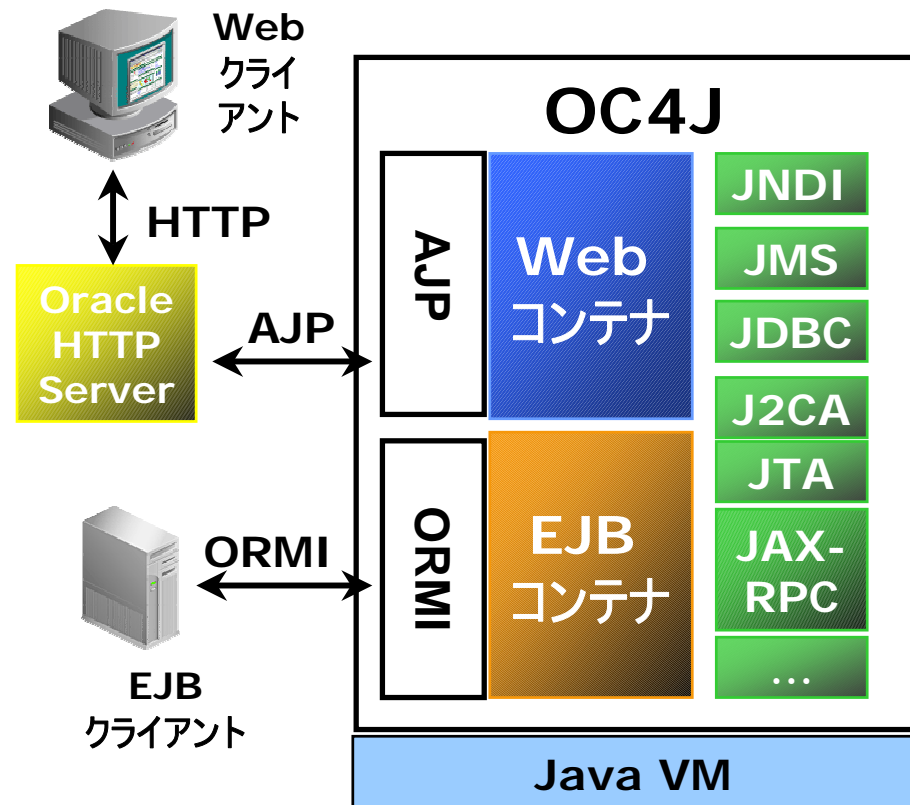


クラスター、リソース管理、高可用性

ORACLE

# J2EE 1.4完全準拠のJ2EEコンテナ Oracle Containers for J2EE (OC4J)

- J2EE標準API群をサポート
  - ✓サーブレット、JSP、EJB、JTA ...
- J2SE 1.4に加え、JDK 5.0もサポート
- EJB 3.0を業界に先駆けてサポート
  - ✓O/Rマッピング・フレームワーク「Oracle TopLink」をベースにした実装
- JSF 1.1をサポート
  - ✓リッチなUIを提供するJSFコンポーネント「ADF Faces」も提供
- JDBCデータソースと接続プーリング
  - Oracle RACと連携した、効率的な接続管理
- 高度なJMS実装を提供
  - ✓インメモリ、ファイル、DBベースの永続化
- ID管理と連携可能なJAAS実装
  - Oracle Identity Managementによるユーザ管理や認証/認可



# J2EE 1.4完全準拠のJ2EEコンテナ Oracle Containers for J2EE (OC4J)

## アプリケーション実行環境

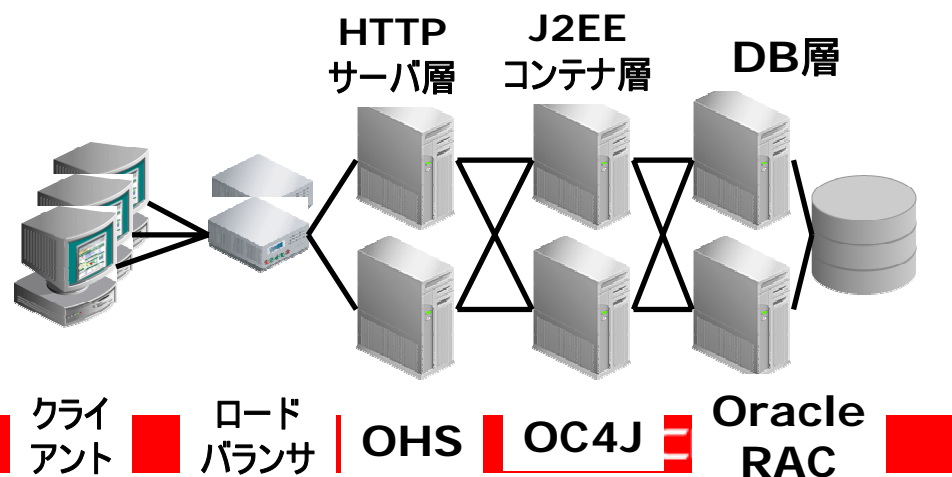
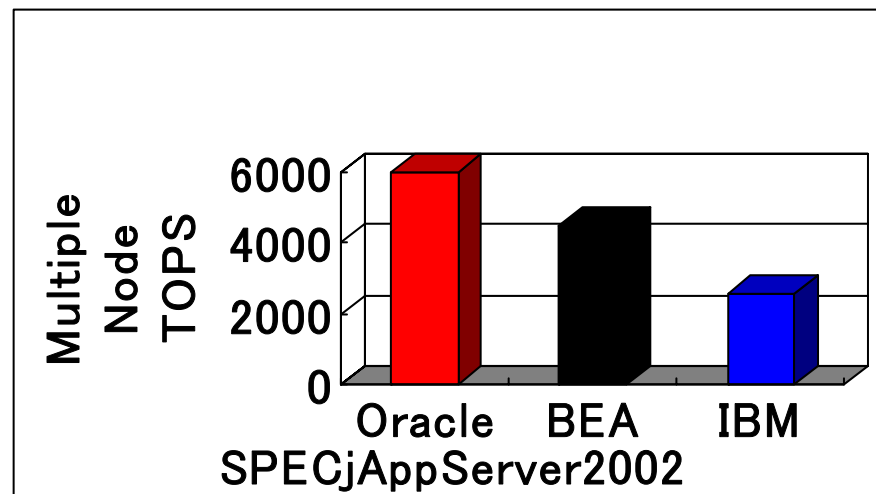
- ▶ カスタムJ2EEアプリケーションのみならず、Oracle Fusion Middlewareの他のコンポーネントの実行環境としても利用

## 優れたパフォーマンス

- ▶ 業界標準J2EEベンチマーク SPECjAppServer で実証済み

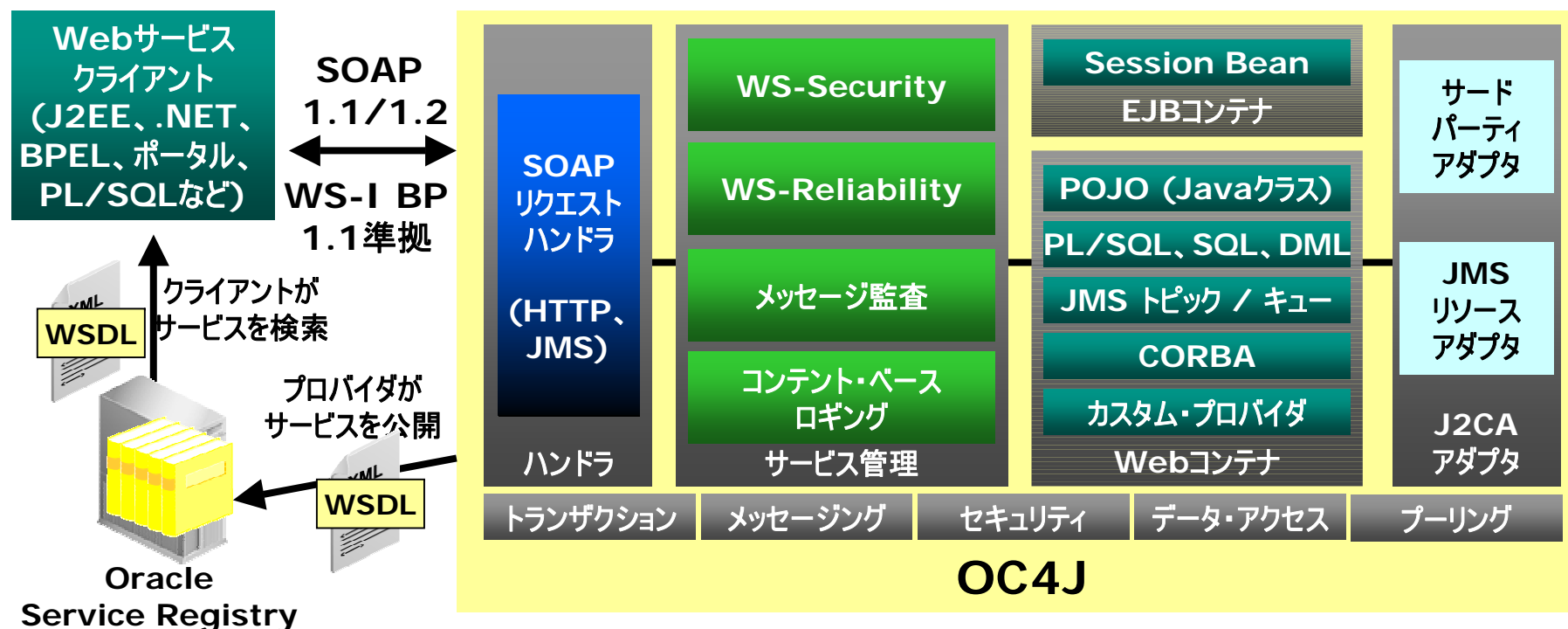
## 高い可用性と拡張性

- ▶ OC4JがOHS (Oracle HTTP Server)、プロセス監視機能 OPMN (Oracle Process Manager and Notification Server) と連携
- ▶ すべての層にわたるクラスタリング/スケール・アウトが可能





# Webサービス実行環境 OracleAS Web Services



## Webサービス関連仕様のサポート

- JAX-RPC 1.1、SOAP 1.1/1.2、WSDL 1.1、UDDI 3.0、WS-Security 1.0、WS-Reliability 1.0など

## 高度な機能

- EJBやJavaクラスのみならず、多様なリソースをWebサービス化可能
- Oracle JDeveloperによる容易なWebサービスやクライアントの開発

# BPEL準拠のBPM実行基盤

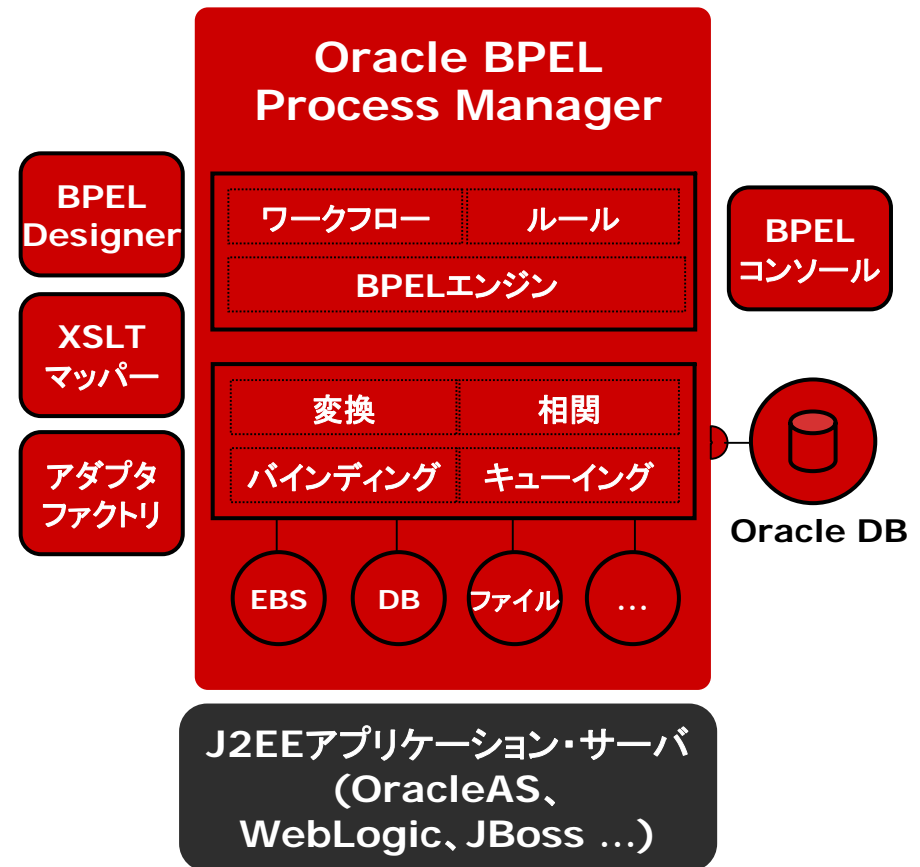
## Oracle BPEL Process Manager

### ネイティブBPEL実行環境

- BPEL4WS 1.1に完全準拠
- BPELをそのまま実行するネイティブ実装
- 他社APサーバ上での利用もサポート
- J2CA準拠のアダプタを介し、多様なシステム/リソースを統合可能
- ヒューマン・ワークフローのサポート
- ルール・エンジン連携 (Oracle、iLog)
- 高い信頼性 (プロセス状態をDBに格納)
- 高いスケーラビリティ (中間層、DB層でスケール・アウト可能)

### SOAライフサイクルをサポート

- JDeveloperによるサービス開発
- BPEL Designerによるプロセス開発
- BPELコンソールによる運用管理



ORACLE

# BPEL準拠のBPM実行基盤

## Oracle BPEL Process Manager

### BPEL Designer

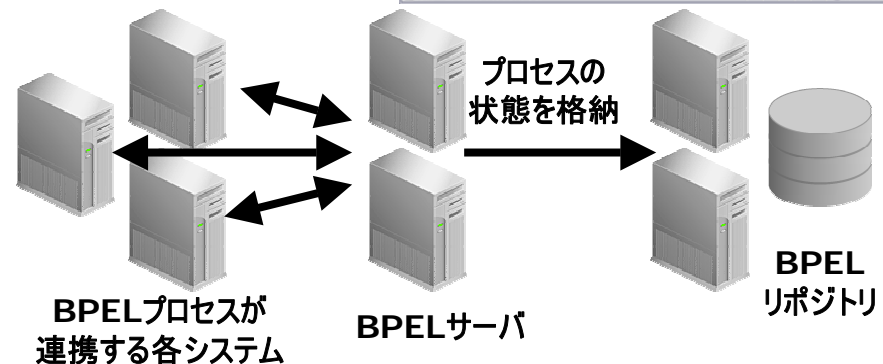
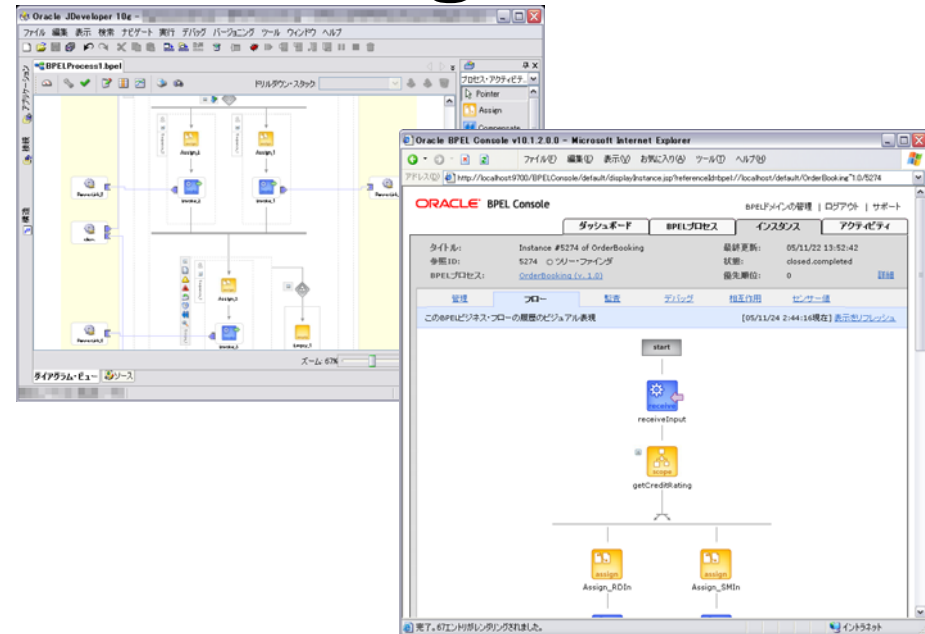
- BPELプロセスの設計・開発ツール
- GUIで簡単に業務フローを定義可能
- ビジネス・プロセスの可視化が可能

### BPELコンソール

- BPELサーバの運用管理ツール
- プロセスのテスト実行や、実行されたプロセスの管理/モニタリングが可能

### 信頼性の高い実行環境

- DBにプロセスの状態を格納することで、BPELサーバに障害が発生してもプロセスを継続可能
- 中間層/DB層 (RAC) でのスケールアウトによる、高可用性とスケーラビリティ



ORACLE

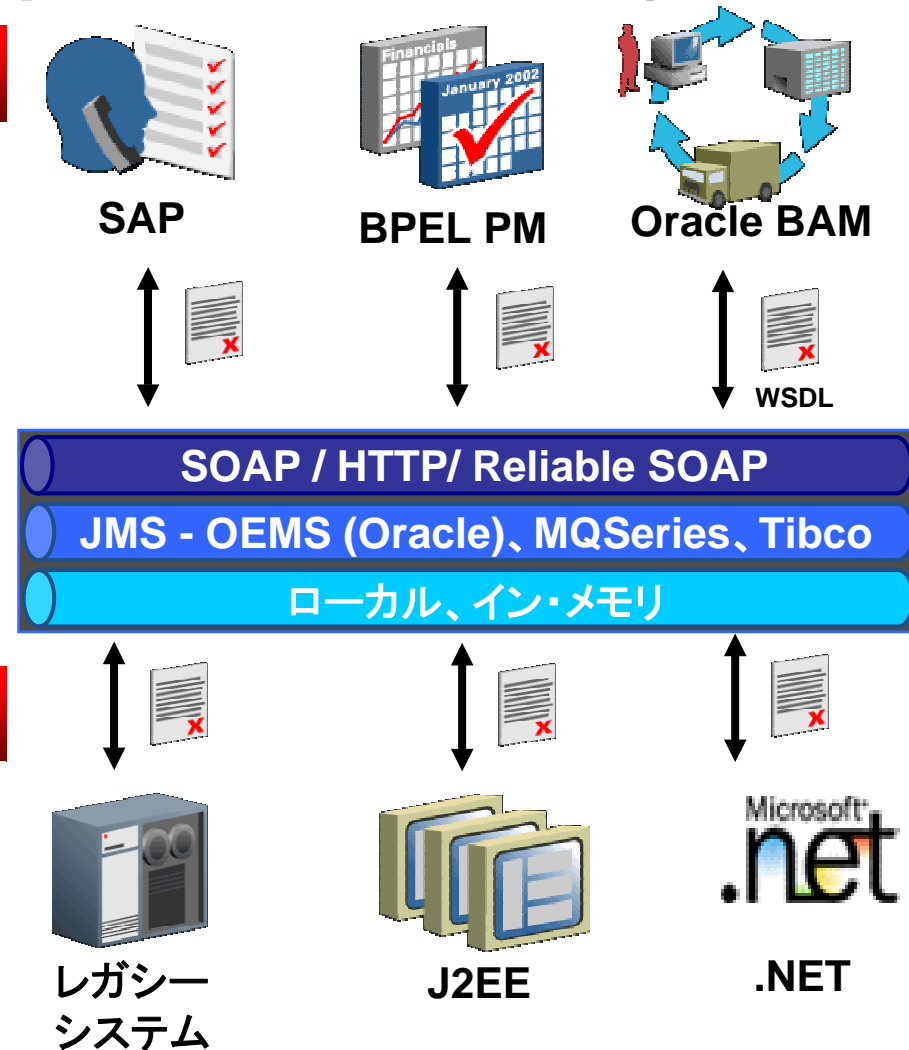
# Oracle ESB (Enterprise Service Bus)

## サービス・バス (ESB)

- SOAを補完するEDA (イベント駆動アーキテクチャ) をサポート
- 多様なシステム/サービス間のメッセージ交換をサポートするサービス・バス
  - ✓ Webサービス、JMS、J2CA準拠のアダプタ
- データ変換 (XSLT)、相互参照、ドメイン値マッピングが可能
- コンテント・ベース・ルーティング (CBR)

## 容易な設計/開発と運用管理

- ESB Designer
  - ✓ ESBの設計/開発のためのJDeveloperベースのGUIツール
- ESBコンソール
  - ✓ 運用管理やモニタリングのためのGUIツール

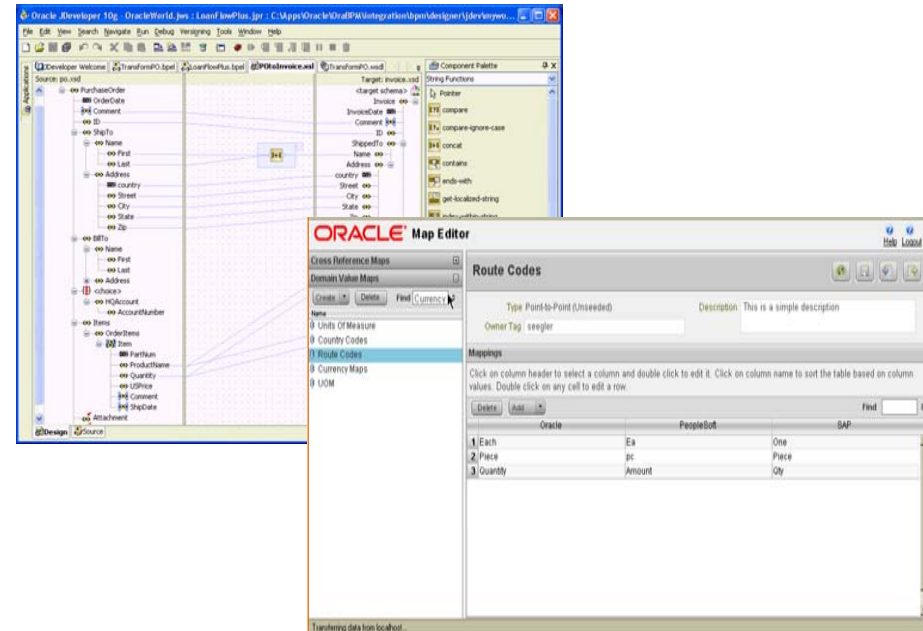


ORACLE

# Oracle ESB (Enterprise Service Bus)

## ESB Designer

- アダプタ・ウィザード (J2CAアダプタの構成)
- XSLTマッパー (データ変換)
- ドメイン値マップ・エディタ
- 相互参照エディタ
- コンテンツ・ベース・ルーティングの設定

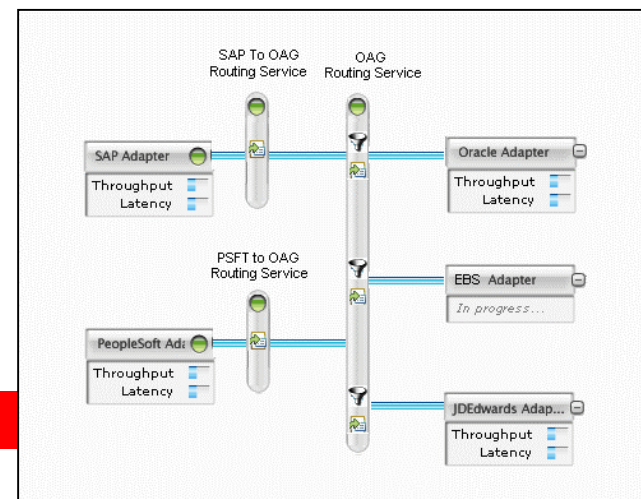


## ESBコンソール

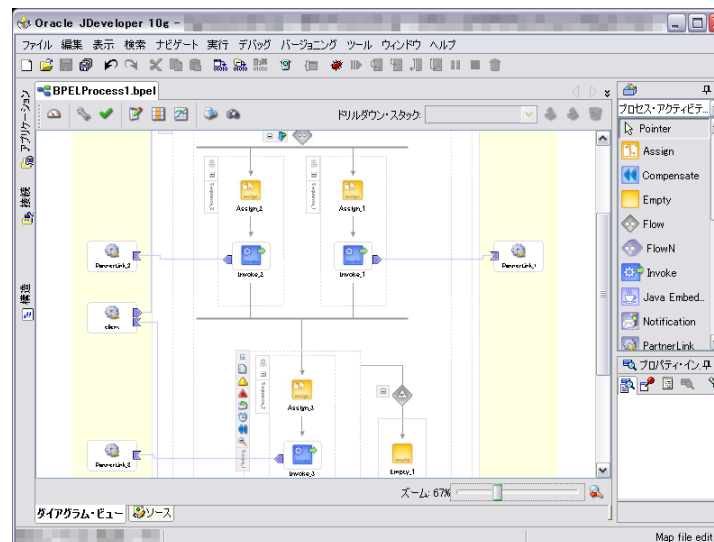
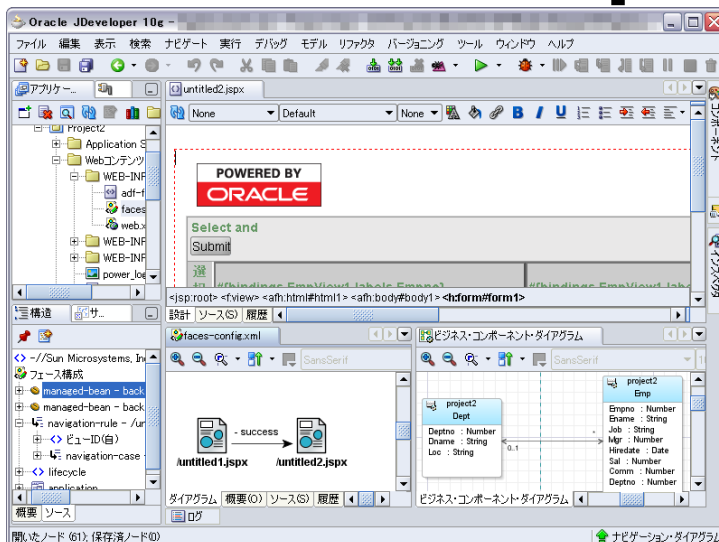
- ESB実行環境の運用管理ツール
- イベント/メッセージのモニタリング

## BPEL Process Manager連携

- ESBで、他システムとのメッセージ交換、単純なルーティングを実行
- BPEL PMで、複雑なプロセス (例外処理、並列処理など) やワークフローを実行



# SOA統合開発環境 (IDE) Oracle JDeveloper



## Java/J2EE開発のためのIDE

- JDK 5.0、J2EE 1.4、JSF 1.1、EJB 3.0などをサポートした無償IDE
- EoDを実現する生産性の高いJ2EEフレームワーク「Oracle ADF」を提供
- 開発ライフサイクルすべてをサポート (モデリング、ソース管理、プロファイリングなど)

## Webサービス/SOA開発

- XML開発 (XML Schemaエディタ、XSLTマップ、XQuery、XPath、JAXB開発など)
- Webサービス開発 (JAX-RPC、JSR-181、クライアント開発、UDDIブラウザなど)
- BPELプロセス開発 (ビジュアルBPEL開発、アダプタ・ウィザード、ワークフロー開発など)

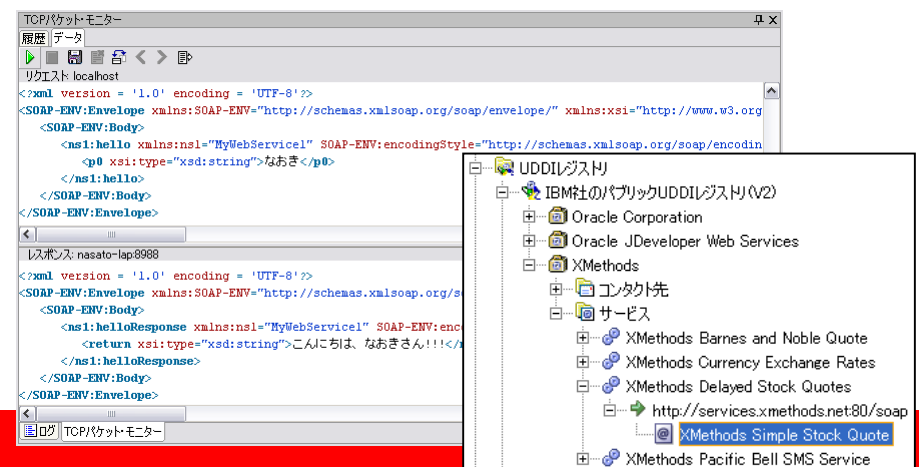
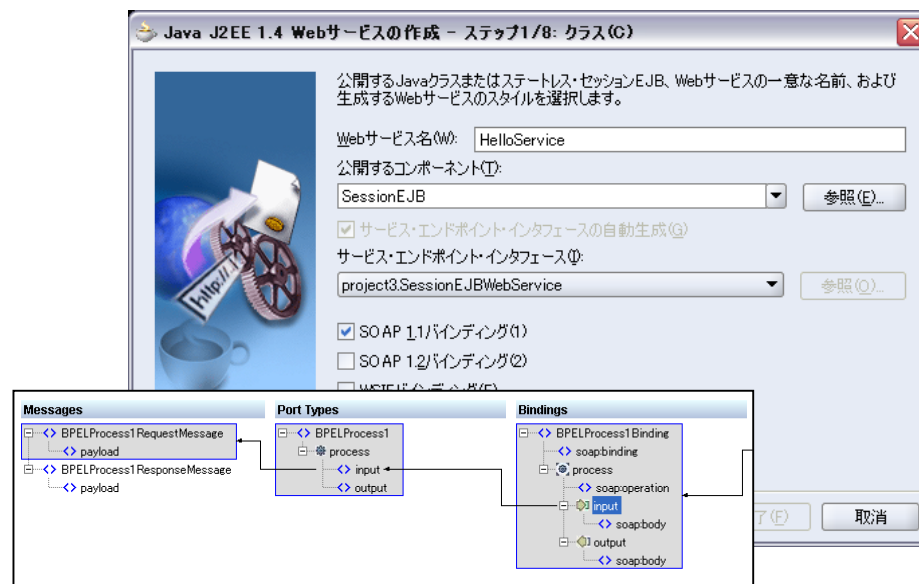
# Oracle JDeveloper Webサービス開発を支援する機能

## Webサービスとクライアントの開発

- ▶ ボトム・アップ開発 (既存のJavaクラス、EJB、PL/SQLなどをWebサービス化)
- ▶ トップ・ダウン開発 (WSDLから実装を生成)
- ▶ クライアント開発 (WSDLからスタブを生成)
- ▶ JAX-RPC 1.1、WS for J2EE 1.1、JSR-181 (アノテーションによる開発)、WSIF、WS-Security、WS-Reliability ...

## 生産性を高める各種機能

- ▶ ビジュアルWSDLエディタ (複雑なWSDLを可視化)
- ▶ UDDIブラウザ (レジストリからの検索、レジストリへの公開)
- ▶ HTTPアナライザ (SOAPメッセージをキャプチャ)





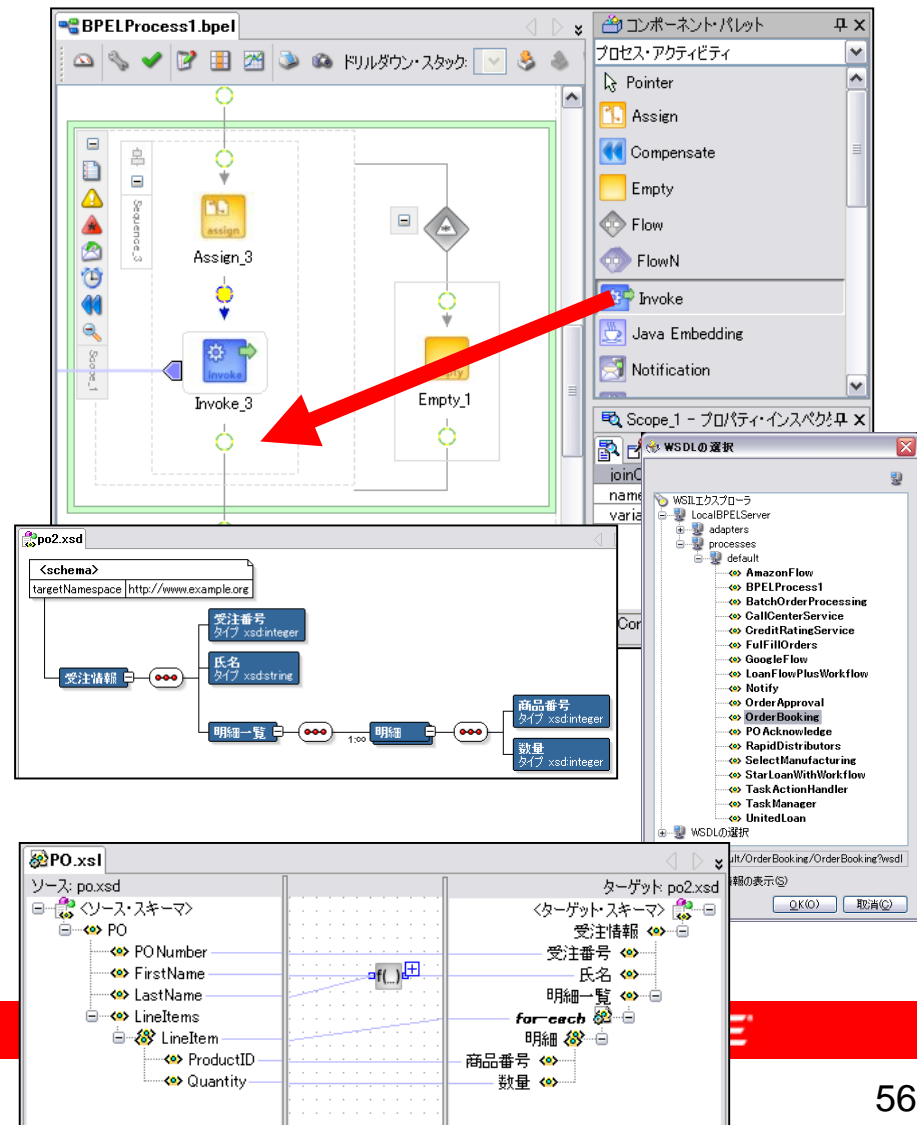
# Oracle JDeveloper BPEL開発機能 (Oracle BPEL Designer)

## ビジュアルBPELプロセス開発

- ドラッグ&ドロップでBPELプロセスを開発
- プロセスの可視化/モデリング
- BPELサーバへのワンクリック・デプロイ
- サード・パーティBPMNモデリング・ツールとの連携も可能

## BPEL開発を支援する各種機能

- XML Schemaエディタ (XML要素を定義)
- XSLTマップ (XML変数間の変換を定義)
- アダプタ・ウィザード (Webサービス以外のリソース/システムとの連携をサポート)
- ヒューマン・ワークフロー・ウィザード
- WSILブラウザ / UDDIブラウザ (既存サービスの検索)





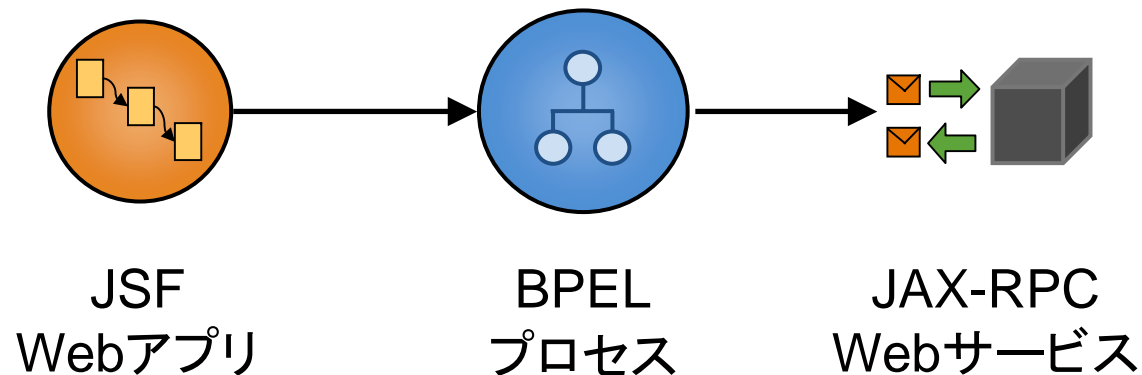
# Agenda

- SOAとそれを支えるテクノロジーの動向
  - SOAとは?
  - Webサービス
  - BPMとBPEL
  - ESBとJBI
- Oracle Fusion Middleware
  - Oracle Fusion Architecture
  - OC4J / OracleAS Web Services
  - Oracle BPEL Process Manager
  - Oracle JDeveloper

 デモンストレーション

# デモンストレーション

1. POJOを作成し、JAX-RPC Webサービスとして公開
2. BPELプロセスを作成し、Webサービスを呼び出す
3. JSFで、BPELプロセスを起動するWebページを作成



# まとめ

- SOAに関連するテクノロジーの動向についてご紹介してきました
- ですが、SOAの本質は、テクノロジーではなく、ITアーキテクチャのコンセプトです
- 標準仕様の動きをウォッチしつつも、それを待っていてはいつまで経っても始められません。
- オラクルが提供するSOA開発/実行環境で、SOAの第一歩を踏み出してみましよう!