

## 第7回 XMLコンソーシアムDay

# UMTP BPMN 研究会の紹介 ～モデリングパターン化への取り組み～

2005年12月16日（15:30～16:30）

UMTP/Japan BPMN研究会  
日揮情報ソフトウェア(株) 技術本部  
明庭 聡  
akeniwa.satoshi@jsys.co.jp

- 特定非営利活動法人 UMLモデリング推進協議会  
(UMTP/Japan)
- BPMN研究会
- BPMN (Business Process Modeling Notation)
- BPMNによってシステム開発がどう変わるのか
- BPMNとBPELのマッピング
- 研究会活動
  - BPMNモデリングの実践
    - ◇ ワイン配送センター問題
  - モデリングパターン化への取り組み
    - ◇ ローレベルBPMNパターン

# UMTP/Japan

- 2003年5月19日設立（2003年10月 NPO認可）
- 設立発起人（21法人・個人）
  - 日本IBM、日立製作所、富士通、NEC、NEXS、NTTデータ、サントリー、日本オラクル、サンモアテック、キャッツ、テクノロジックアート、東芝、東陽テクニカ、日本ユニシス、日本ラショナルソフトウェア、NRI、豆蔵、Aithent Inc. Japan.、ラーニングアーキテクチャ研究所、堀内一東京国際大学教授、オーシス総研
- 協賛
  - JISA, JUAS, ECOM, JAHIS, JCMA, CBOP
- オブザーバ
  - 経済産業省
- 協力
  - 中国武漢大学、ソフトウェア工学研究所、 韓国高麗大学

- 本法人は、国内だけでなく中国、韓国などアジア諸国の関連団体などと連携し、地域におけるビジネス連携を容易とすべく、「**共通モデリング技術の策定**」と「**統一的なモデリング技術者育成とその技能認定の仕組の運営**」を通じて、わが国産業のグローバル化とアジア地域におけるソフトウェアビジネスの有機的連携確立を期すものである。

## ■ 認定事業

- モデリング技術者認定
- 教材及びコンテンツ認定

## ■ モデリング技術普及事業

- モデリング技術調査研究
- モデリング技術体系化とスキル体系
- 育成カリキュラム策定
- 技術者フォーラム

## ■ モデル共有促進事業

- 各分野のベストプラクティス・モデル開発支援
- モデル登録と共有化支援

## ■ 国際連携

- UMTP/Asiaの運営（中国、韓国、インド）
- OMGなど海外団体との連携

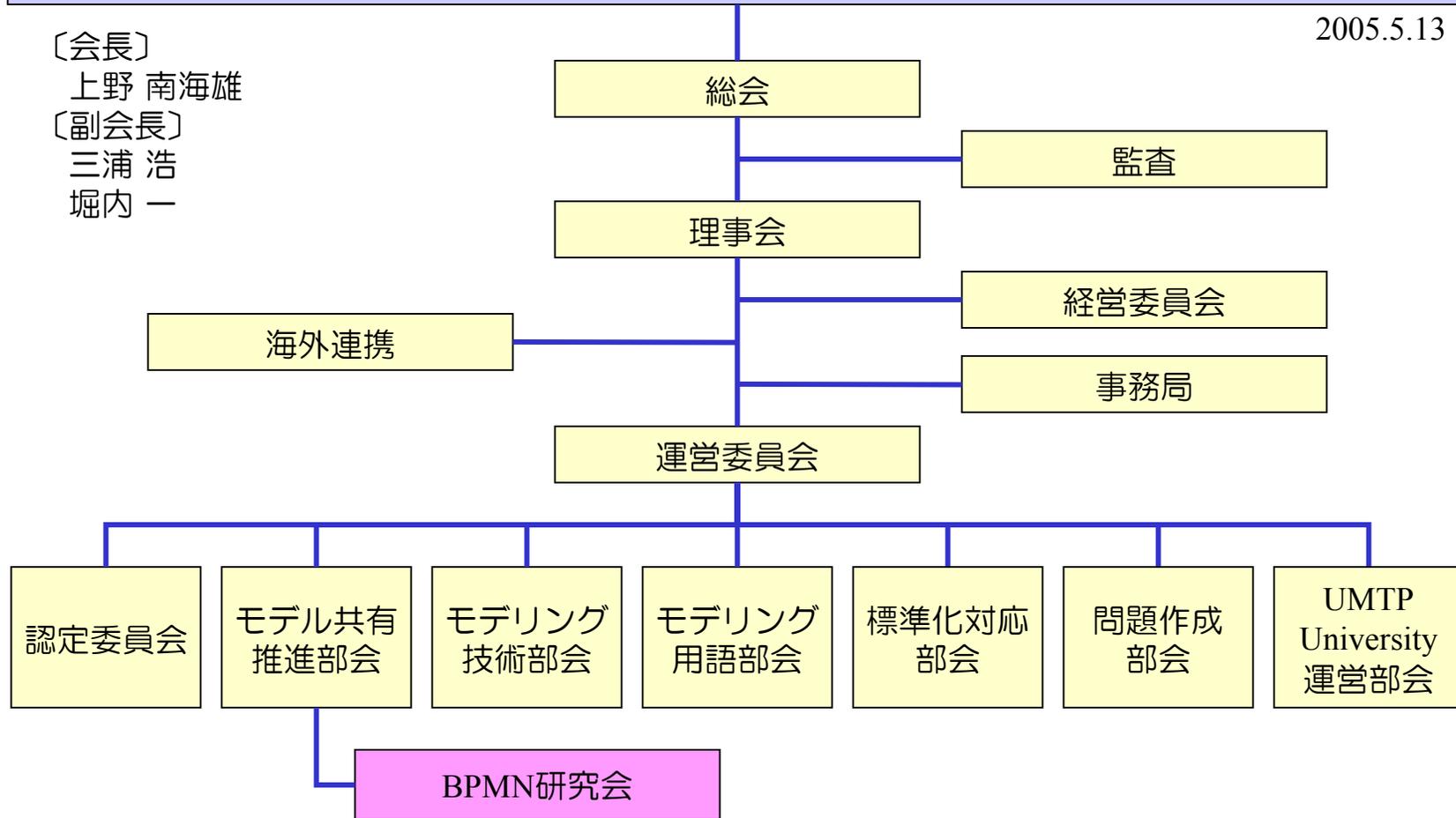
# 2005年度 組織



特定非営利活動法人 UMLモデリング推進協議会

2005.5.13

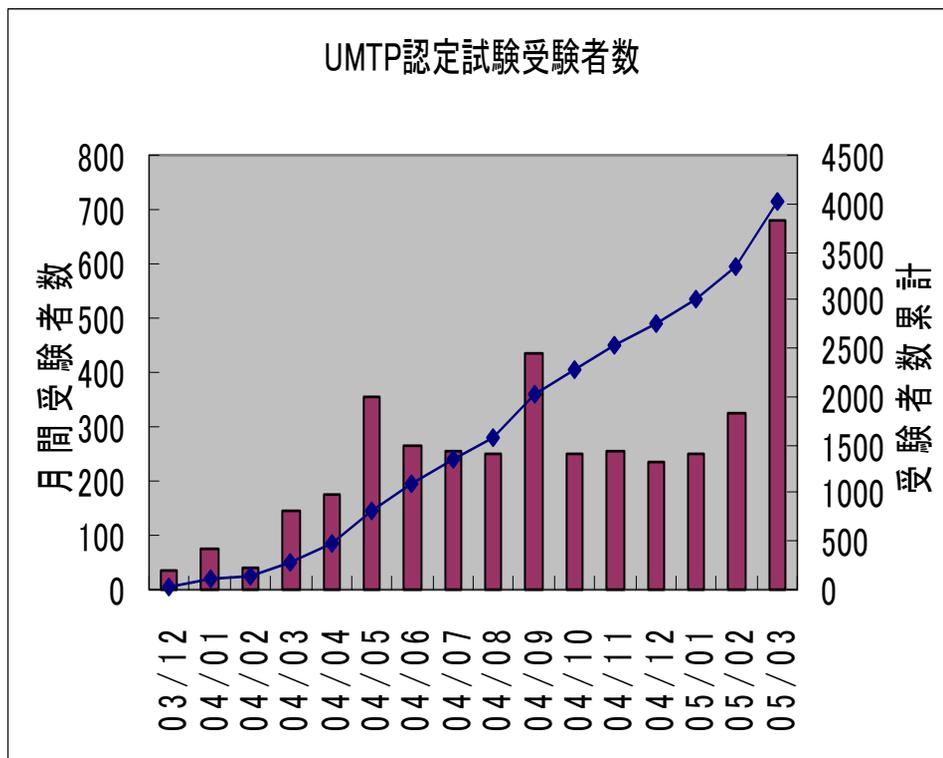
〔会長〕  
上野 南海雄  
〔副会長〕  
三浦 浩  
堀内 一



# 会員数の遷移

|       | 2003年度 | 2004年度 |
|-------|--------|--------|
| 団体正会員 | 29     | 54     |
| 団体準会員 | 3      | 4      |
| 個人正会員 | 0      | 3      |
| 個人準会員 | 4      | 14     |
| 協賛他   | 1      | 1      |
| 合計    | 37     | 76     |

# UMLモデリング技能認定試験



**UMTP  
L4 MODELER**

実践に基づいてモデリングを  
指導できる

**UMTP  
L3 MODELER**

実務でモデリングが実践できる

**UMTP  
L2 MODELER**

UMLモデルの読み書きが  
普通にできる

**UMTP  
L1 MODELER**

簡単なUMLモデルの意味が  
分かる

実施中

※UMLはOMGの商標です。

# BPMN研究会

## ■ 背景

- ビジネスプロセスの分析/設計から実装に至る一連の設計仕様を表記できる新しいダイアグラム技法『BPMN (Business Process Modeling Notation)』への期待
- ビジネスとITのギャップを埋める一手法としてのビジネスプロセス・モデリングに対する期待
- BPM、SOA など、ワークフローエンジンを中核にサービスをオーケストレーションする新技術の台頭

## ■ 目的

- 欧米を中心に注目されているが国内では未だ活用事例がないBPMNに関して、研究会メンバーによる実践を通じて活用するための課題と解決策を研究する

## ■ 設立

- 2005年1月

## ■ 主査

岩田アキラ（日揮情報ソフトウェア株式会社）

## ■ 副主査

富澤雅彦（NECネクサソリューションズ株式会社）

## ■ 研究会員 計34名（2005年6月24日現在）

NECネクサソリューションズ株式会社、株式会社NTTデータ、  
株式会社オージス総研、サントリー株式会社、株式会社シーイーシー、  
セントラル・コンピュータ・サービス株式会社、東京国際大学、  
東京電力株式会社、日揮情報ソフトウェア株式会社、  
株式会社Nixシステム研究所、  
日本テレコムネットワーク情報サービス株式会社、日本プロセス株式会社、  
日本ユニシス株式会社、株式会社日立製作所

# 活動状況

| 2004年度 |    |    | 2005年度 |    |    |    |    |    |     |     |     |    |     |
|--------|----|----|--------|----|----|----|----|----|-----|-----|-----|----|-----|
| 1月     | 2月 | 3月 | 4月     | 5月 | 6月 | 7月 | 8月 | 9月 | 10月 | 11月 | 12月 | 1月 | ... |

▲ 第1回      ▲ 第2回   ▲ 第3回   ▲ 第4回   ▲ 第5回   ▲ 第6回   ▲ 第7回      ▲ 第8回      △ 第9回  
(1/12)

## BPMN表記法習得

◇ BPMNモデリングの実践  
『ワイン配送センター問題』

研究テーマ  
検討

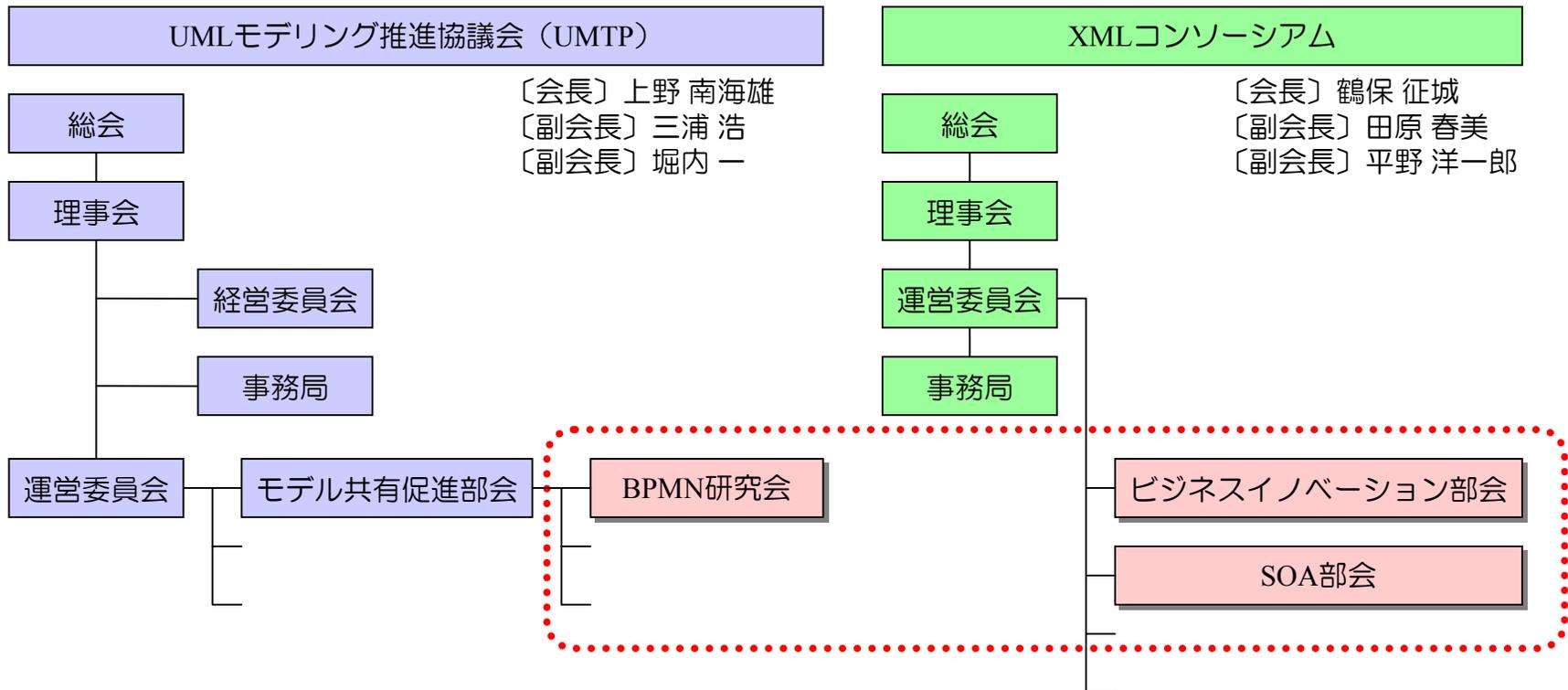
- ◇ BPMN関連情報の紹介
- ・ BPMNの概要
  - ・ BPMNとUMLとの比較
  - ・ BPMI.org 『Miami Think Tank』への参加報告
  - ・ BPMNとBPELのマッピング

研究活動  
『ローレベルBPMNパターン』

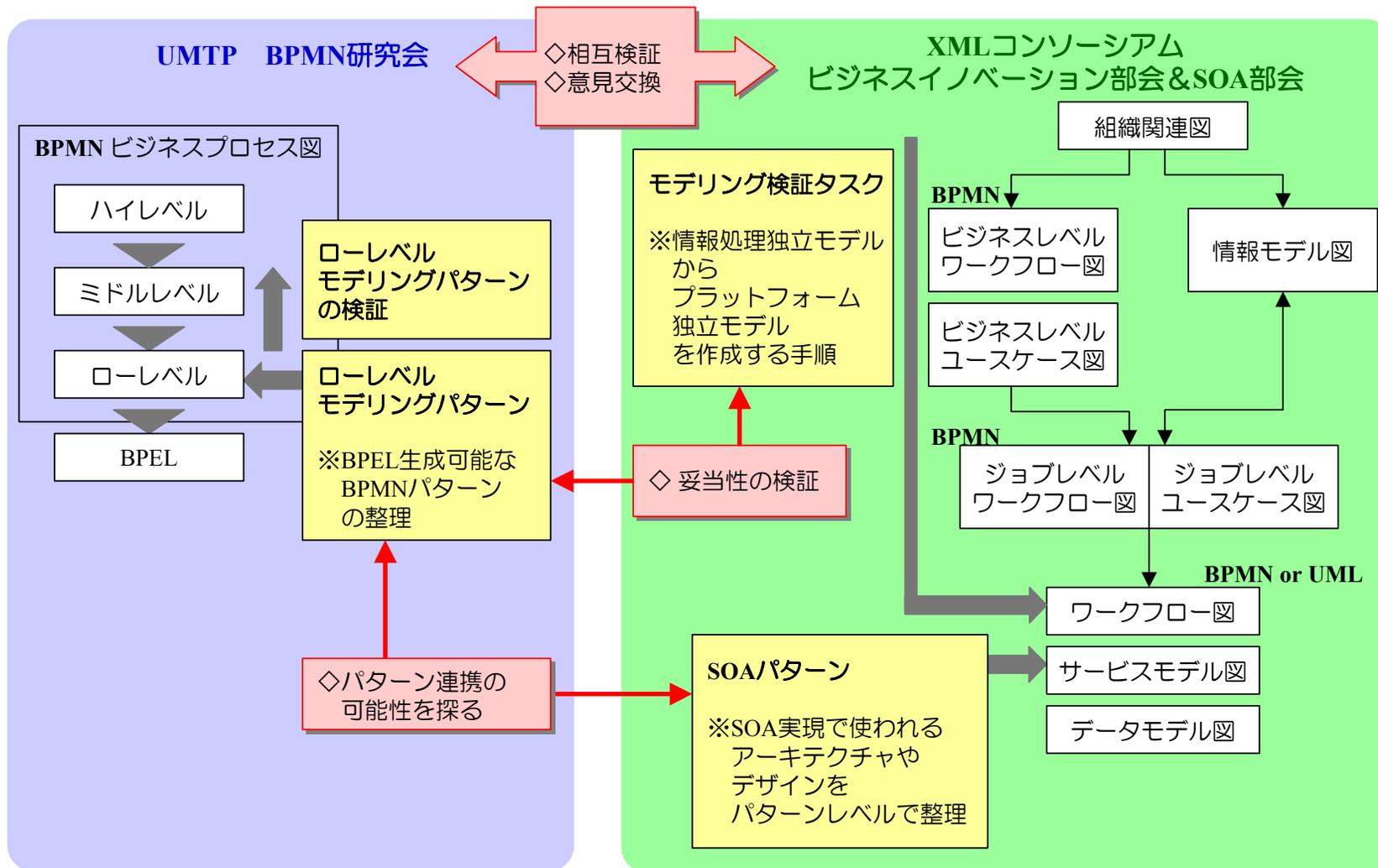
# XMLコンソーシアムとの協業



- UMLモデリング推進協議会とXMLコンソーシアム、BPMとSOAの実現化の研究・調査の協業に合意 (2005年9月)

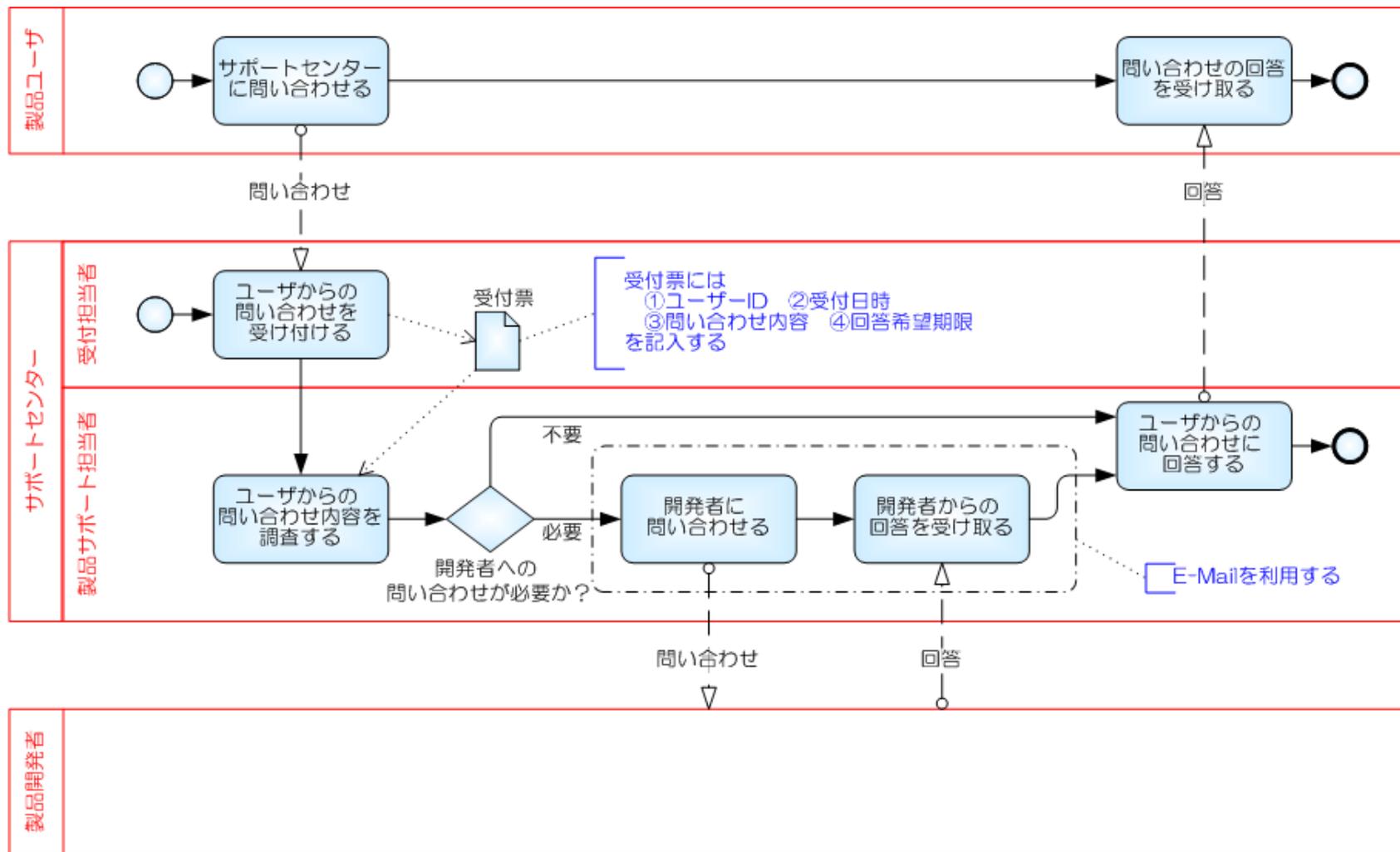


# 研究活動における連携



# BPMN (Business Process Modeling Notation)

# BPMNサンプルモデル



## ■ Business Process Management Initiative (BPML.org)

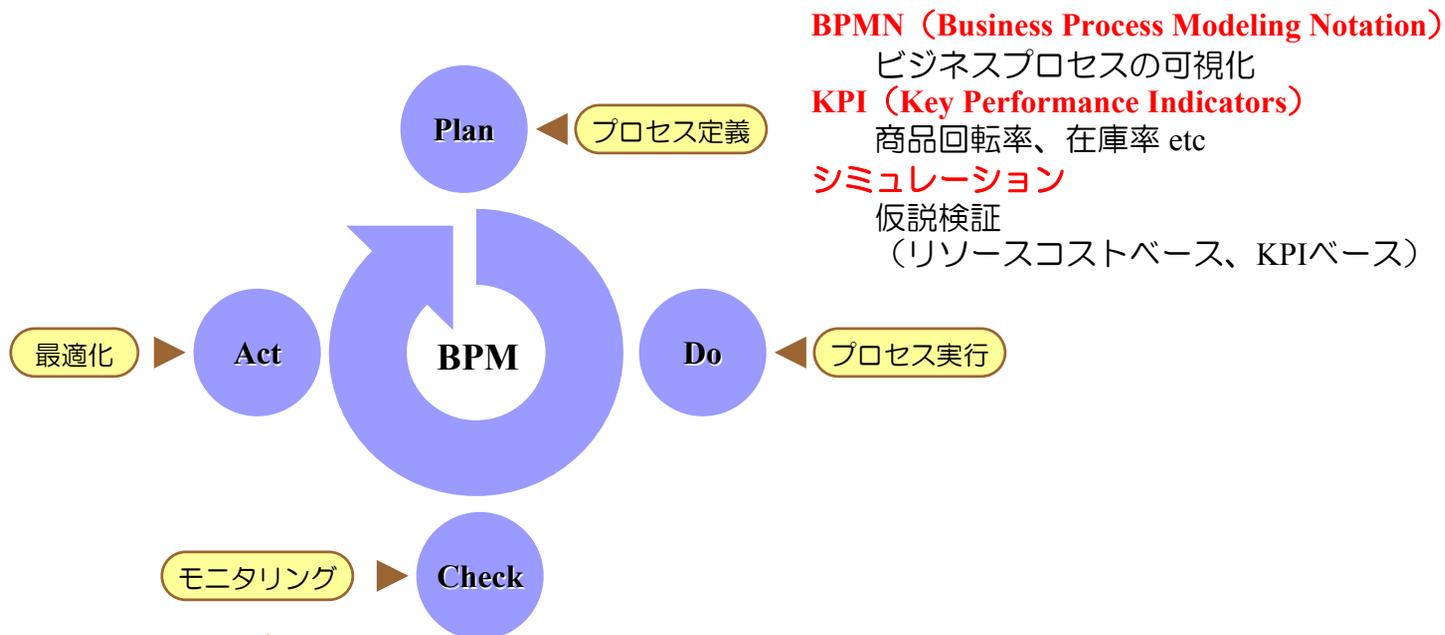
- モデル記述言語ワーキンググループ（座長：Intalio）
  - ◇ 2002年11月13日、BPML 1.0を発行
  - ◇ ワーキングメンバー：Intalio, SAP, Sun, SeeBeyondなど
- モデル表記ワーキンググループ（座長：IBM）
  - ◇ 2003年8月25日、パブリックレビューのため草案を公開
  - ◇ 2004年5月3日、BPMN 1.0を発行
  - ◇ ワーキングメンバー：IBMのほか、モデリングツール、BPMベンダなど

## ■ 参考にした表記、方法論

- UML Activity Diagram、UML EDOC Business Processes、IDEF、ebXML BPSS、Activity Decision Flow(ADF) Diagram、RosettaNet、LOVeM、Event-Process Chains(EPCs)

## ① BPM (Business Process Management)

- ビジネスプロセスの継続的な改善を可能にする概念、仕組み、行為

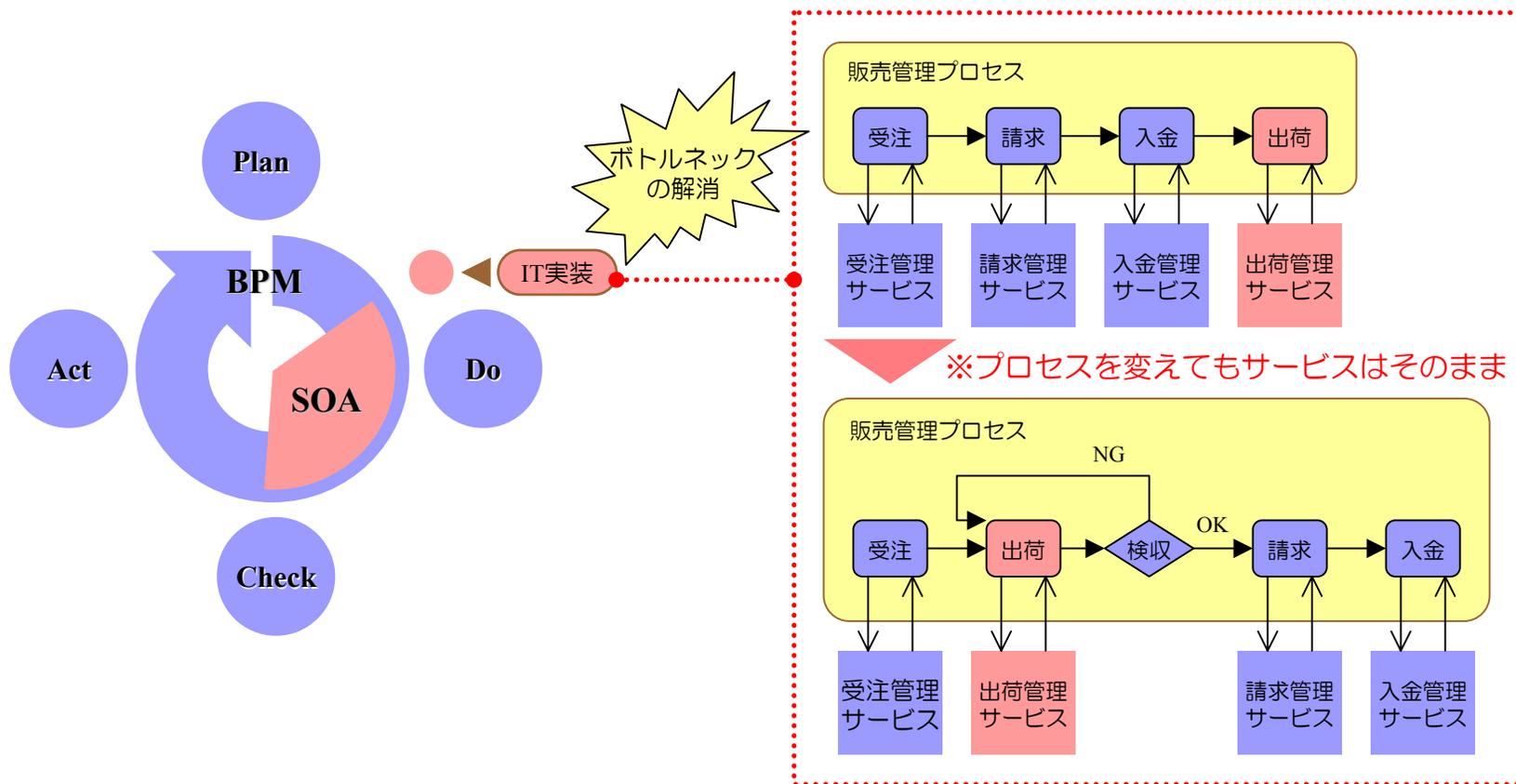


### BAM (Business Activity Monitoring)

プロセスの進捗、滞留状況、処理時間、KPI 実績値 の把握  
(ビジュアルな表示、リアルタイムな通知、レポート等 etc)

## ② シームレスなPDCAサイクル

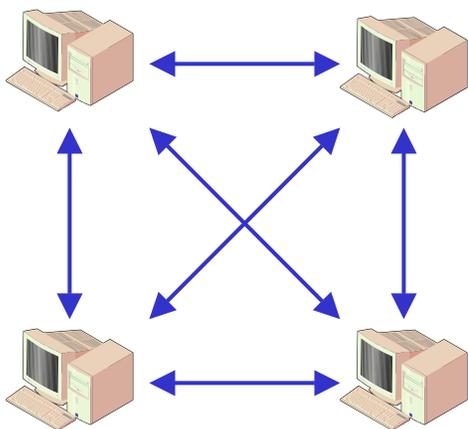
- SOA (Service-Oriented Architecture)によるシステム構築/改善における「迅速性の向上」と「低コスト化」



## ③ SOAにおけるビジネスプロセス

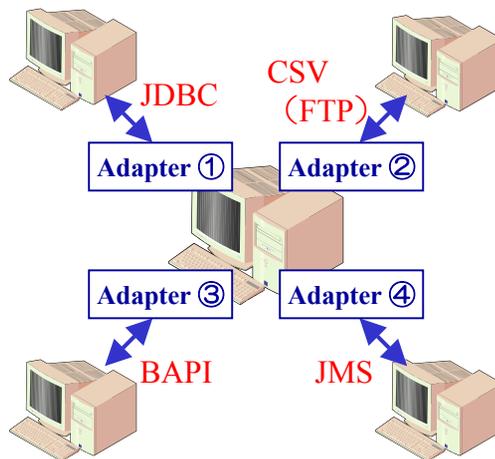
- 企業アプリケーション統合における進化形
- SOAのサーバが企業システムの中核になる
- サービス制御ロジック = ビジネスプロセス (BPEL : Business Process Execution Language)

### Peer to Peer システム間連携



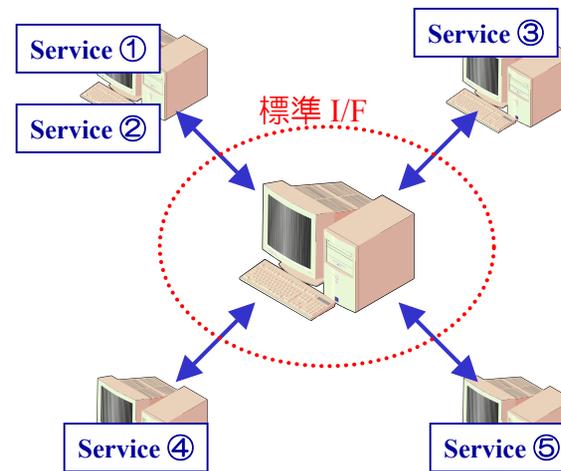
### EAI (集約化)

※EAIサーバ (裏方) が  
周辺システムに合わせる



### SOA (標準化)

※SOAサーバ (主役) に  
周辺サービスが合わせる



## ④ 新たな表記法の必要性

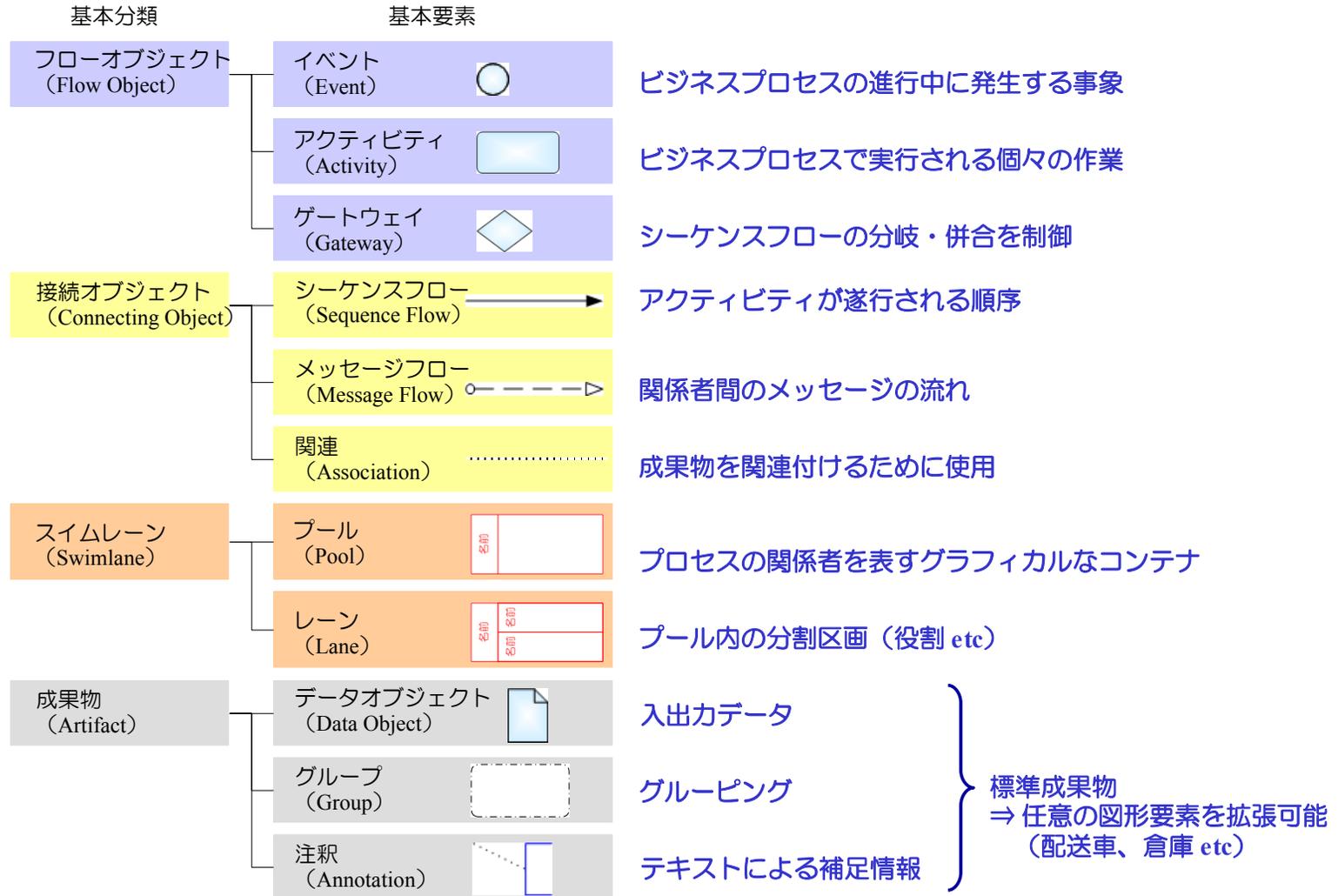
### ■ 共通言語としての表記標準

- BPMサイクルに係わるビジネスユーザ、プロセス設計者、IT技術者の共通言語
  - ◇ 現場のビジネスユーザが読み書きできる表記法
- 企業間プロセス連携、ビジネス参照モデルの流通、汎用的なサービスの流通
  - ◇ プラットフォームに依存しない標準表記

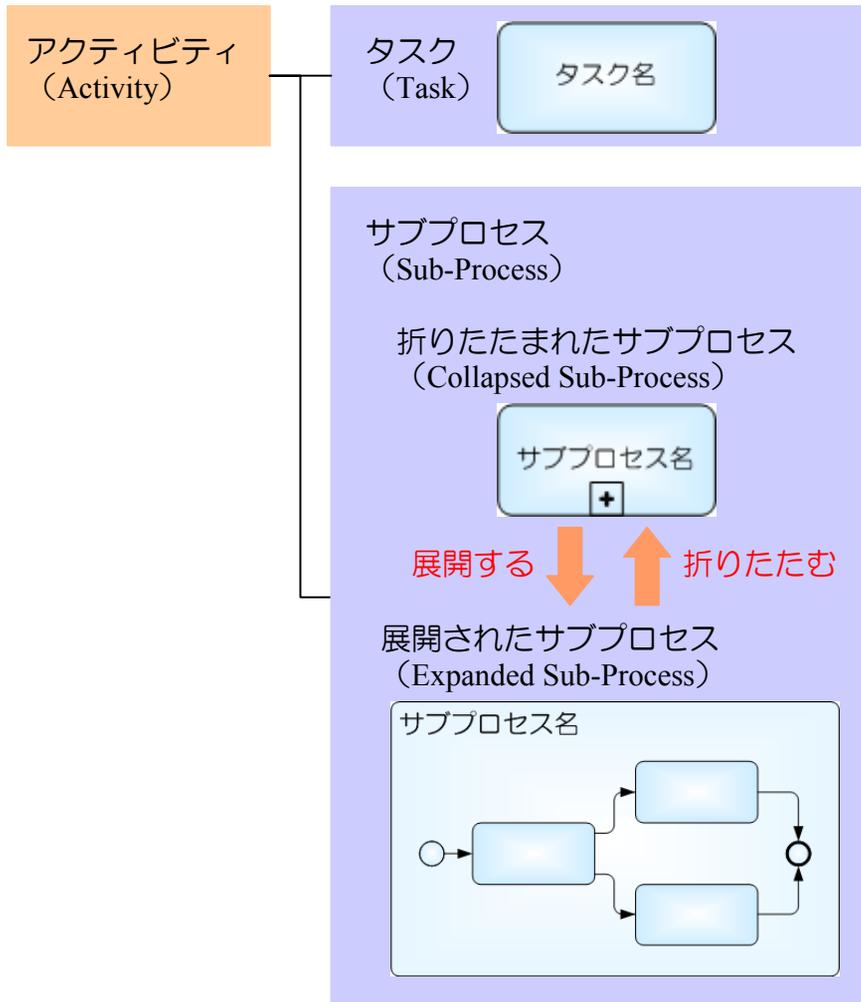
### ■ ビジネスプロセス駆動型開発のための表記標準

- ビジネス環境の変化に迅速に対応するBPMサイクルの実現
- ビジネスプロセス実行言語（BPEL）との親和性

# BPMNの基本セット



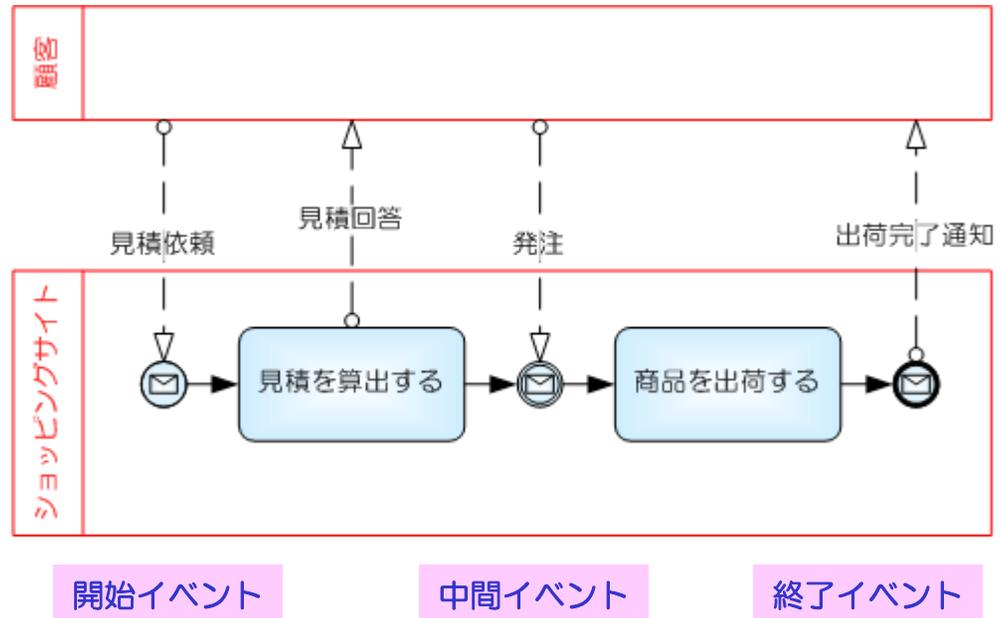
# アクティビティのバリエーション



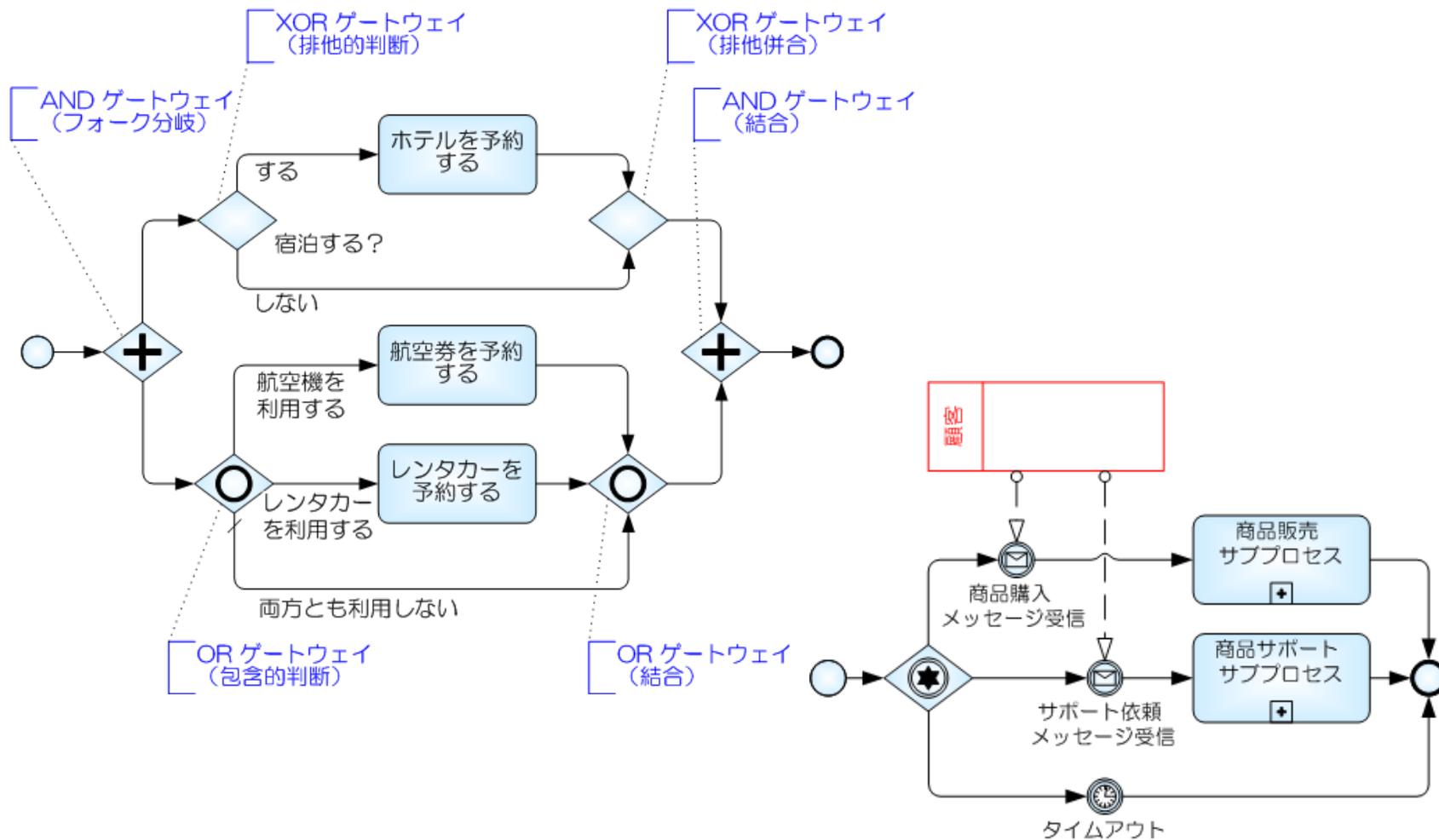
| マーカー名                         | タスク | サブプロセス |
|-------------------------------|-----|--------|
| ループ (Loop)                    |     |        |
| マルチインスタンス (Multiple Instance) |     |        |
| アドホック (Ad-Hoc)                |     |        |
| 補償 (Compensation)             |     |        |

# イベントのバリエーション

| マーカー名             | 開始 | 中間 | 終了 |
|-------------------|----|----|----|
| なし (None)         |    |    |    |
| メッセージ (Message)   |    |    |    |
| タイマー (Timer)      |    |    |    |
| 停止 (Terminate)    |    |    |    |
| リンク (Link)        |    |    |    |
| ルール (Rule)        |    |    |    |
| 補償 (Compensation) |    |    |    |
| キャンセル (Cancel)    |    |    |    |
| エラー (Error)       |    |    |    |
| 複合 (Multiple)     |    |    |    |



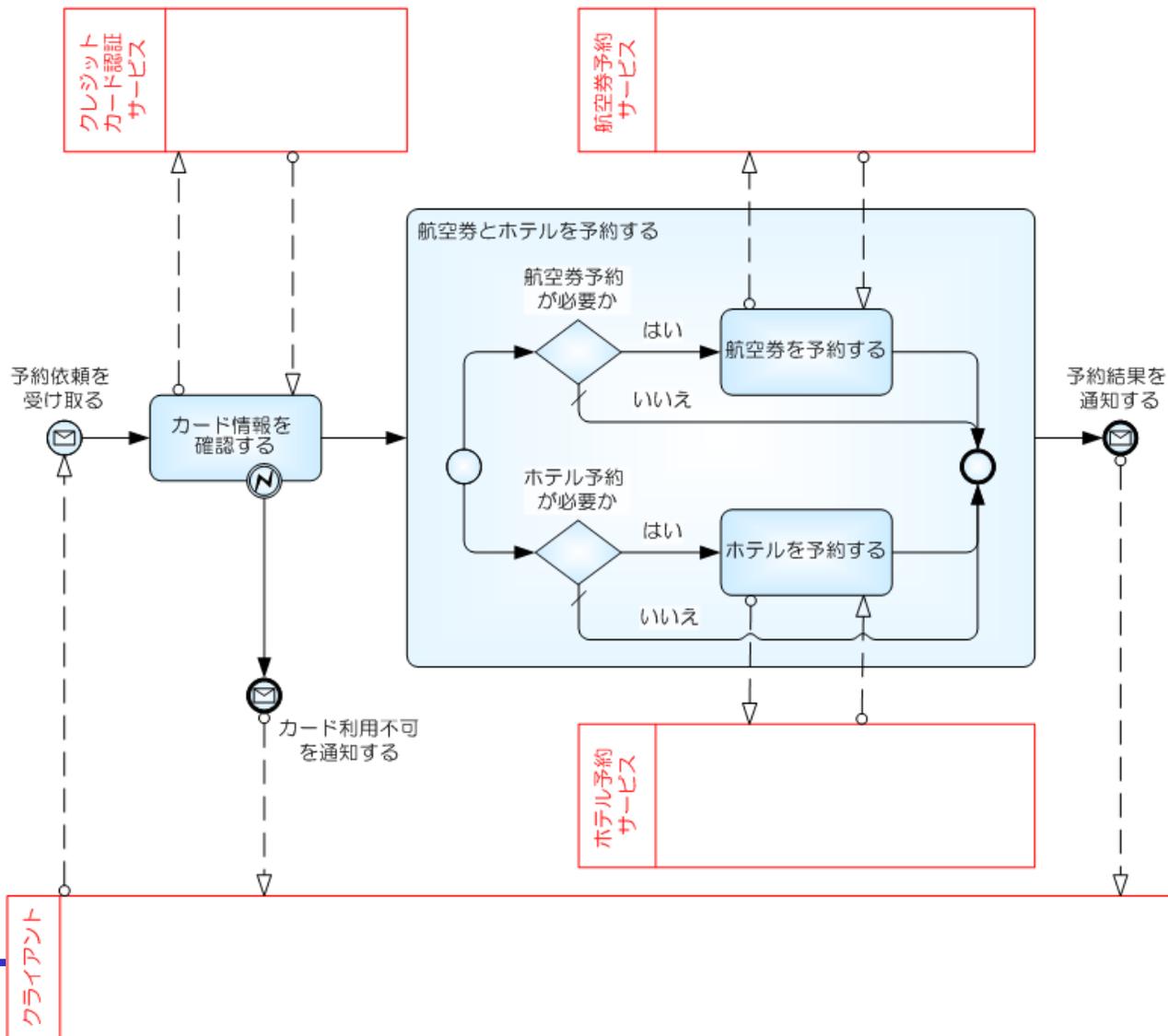
# ゲートウェイのバリエーション



システム開発がどのように変わるのか



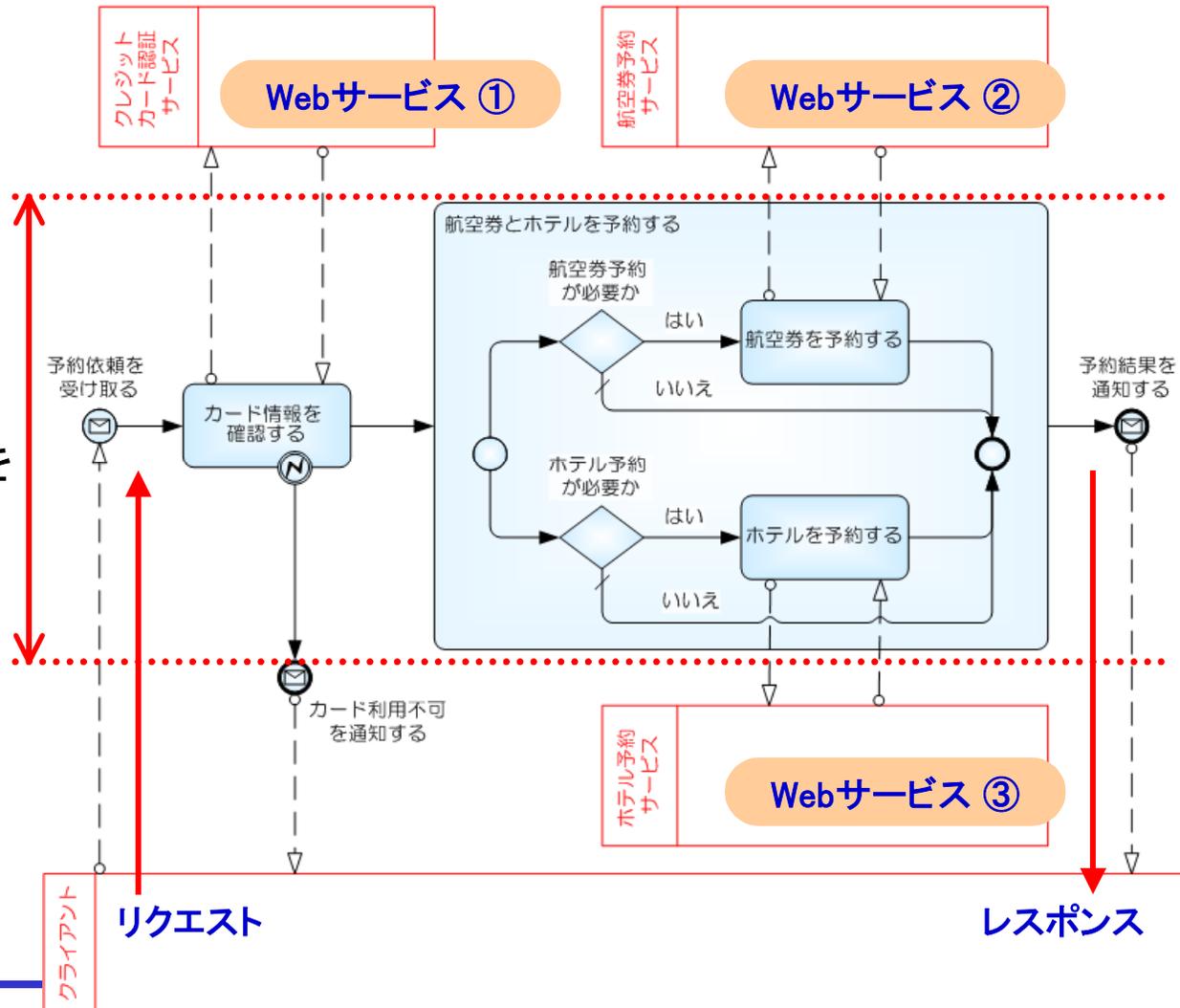
# デモンストレーション サンプルBPMNモデル



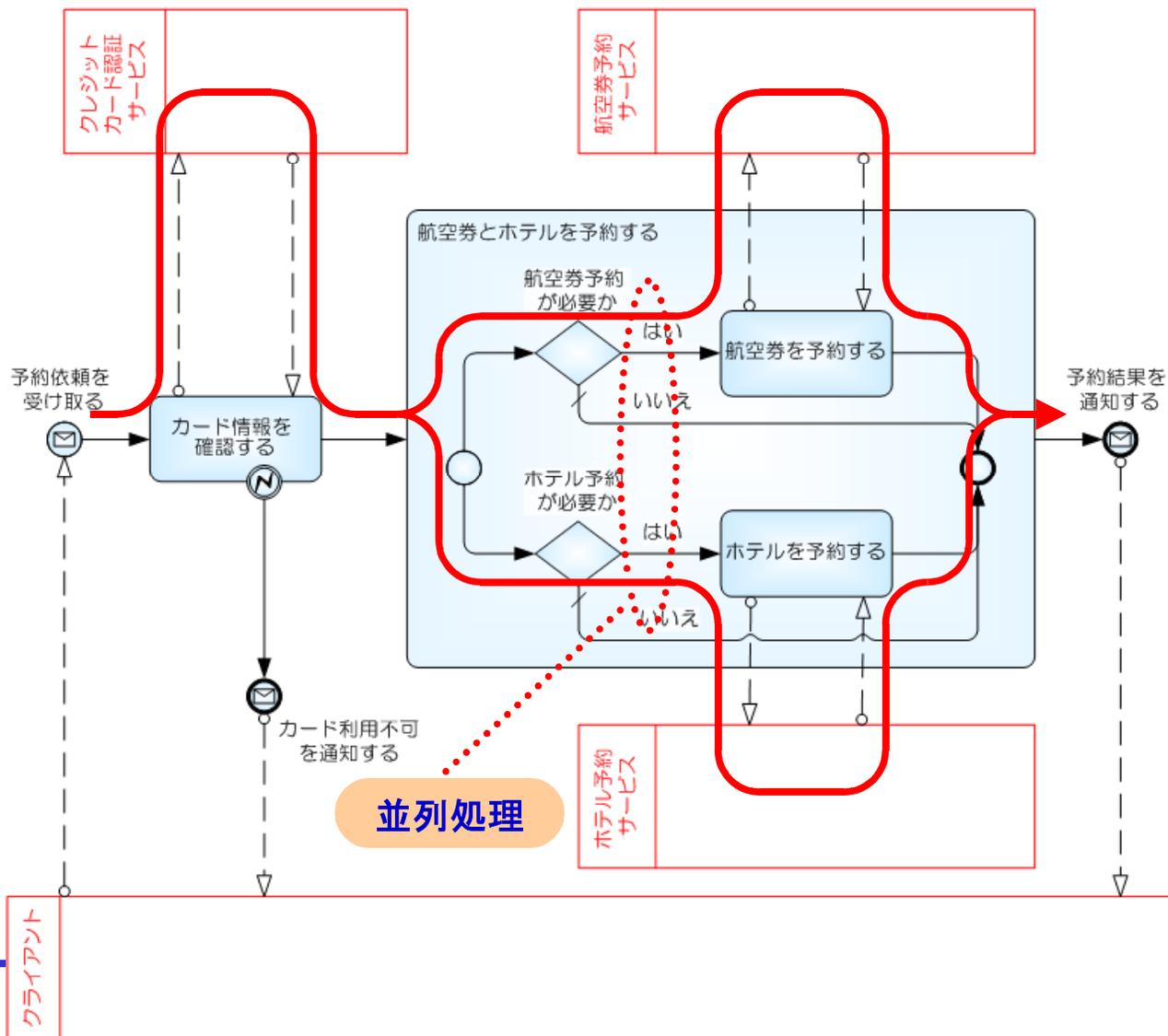
# デモンストレーション

## Webサービスのオーケストレーション

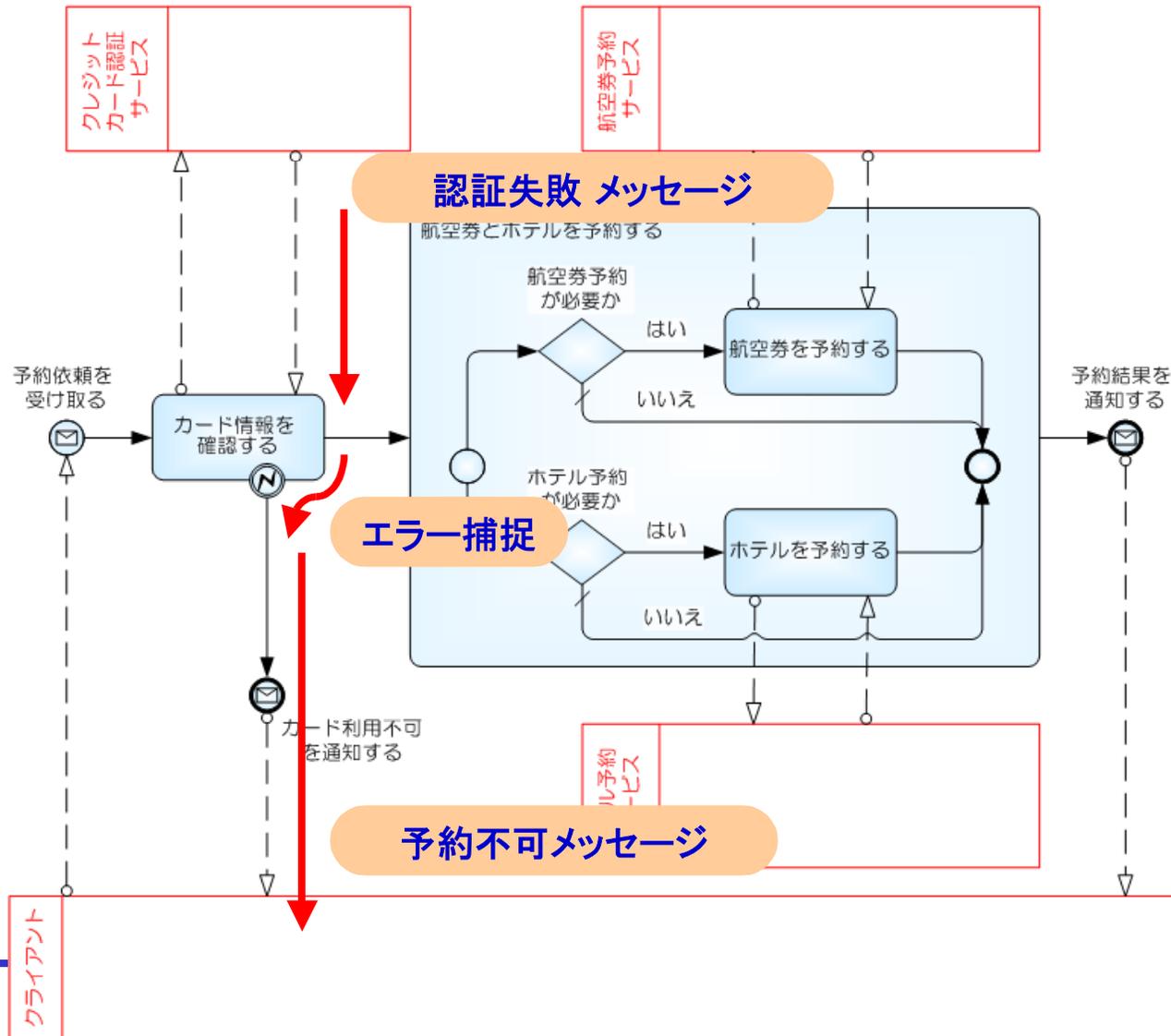
クライアントから  
Webサービスによる  
要求を受け、  
3つの Webサービスを  
オーケストレーション  
するプロセス



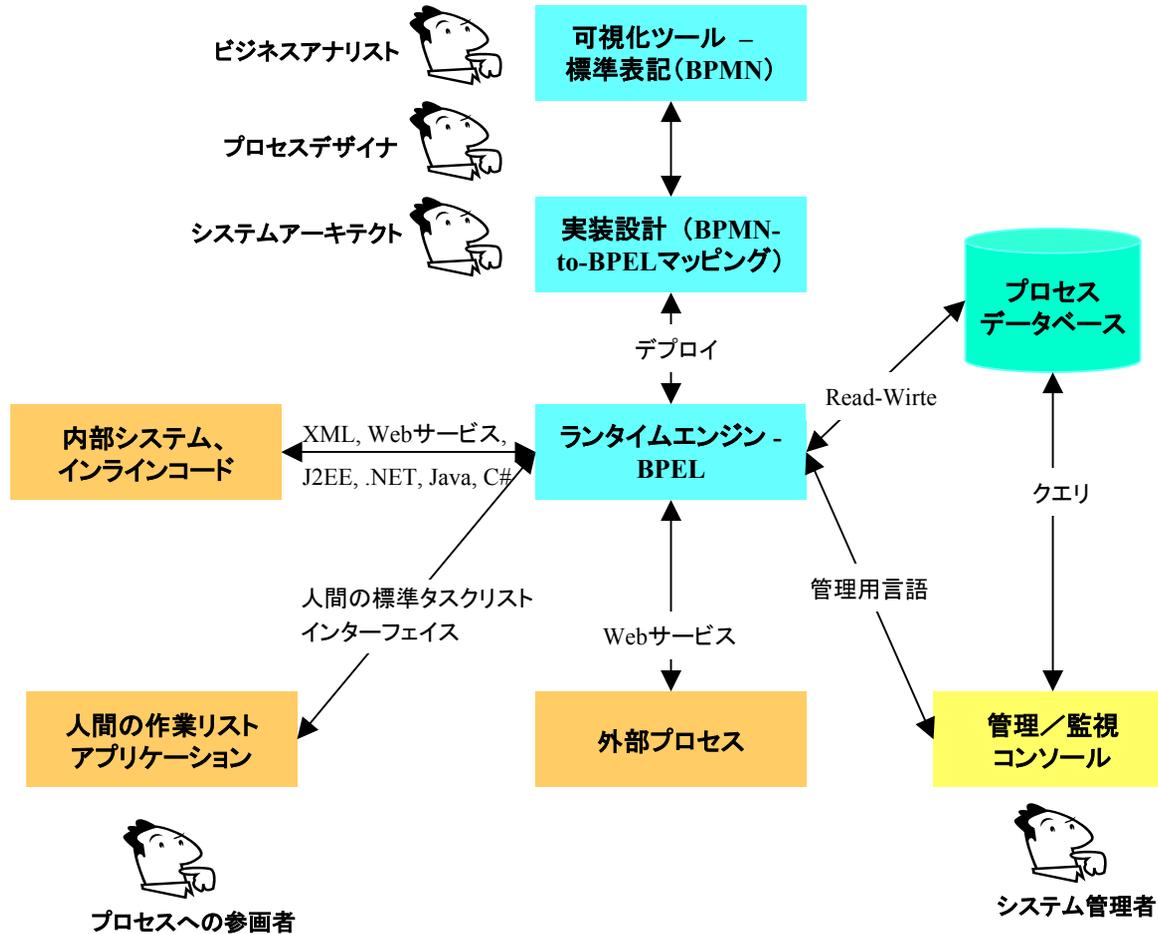
# デモンストレーション 正常処理の流れ



# デモンストレーション 例外処理の流れ



# 理想的なBPM/SOA のアーキテクチャ

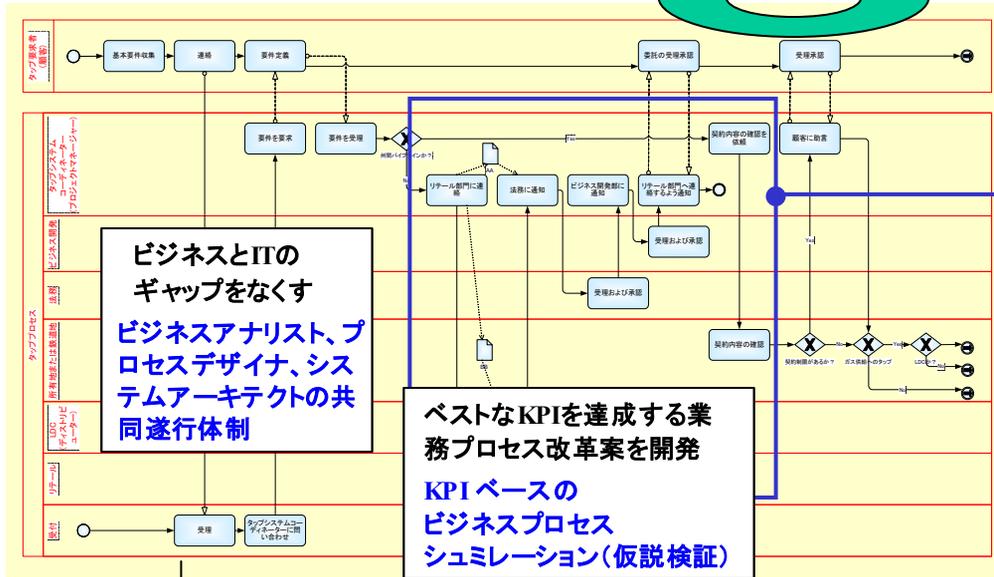


by Mike Havey, author of Essential Business Process Modeling

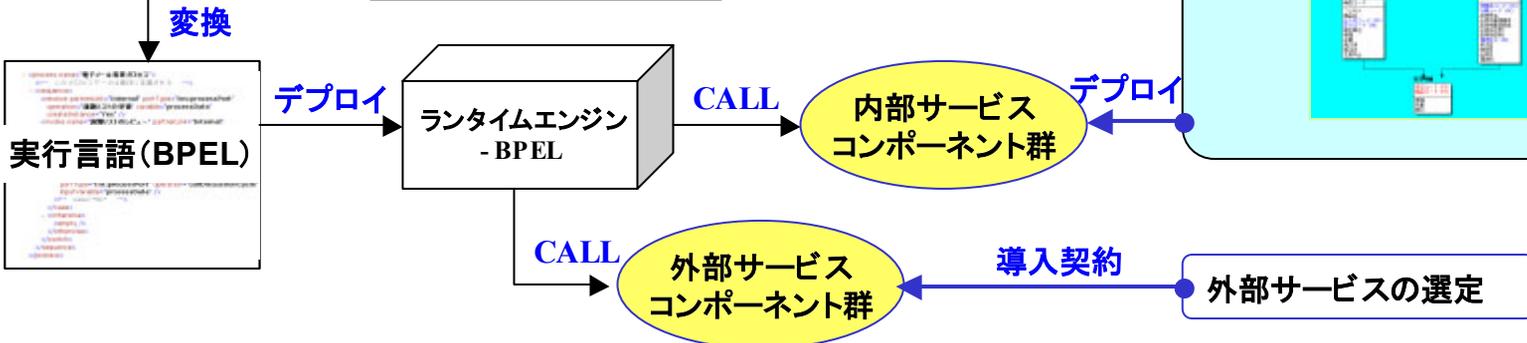
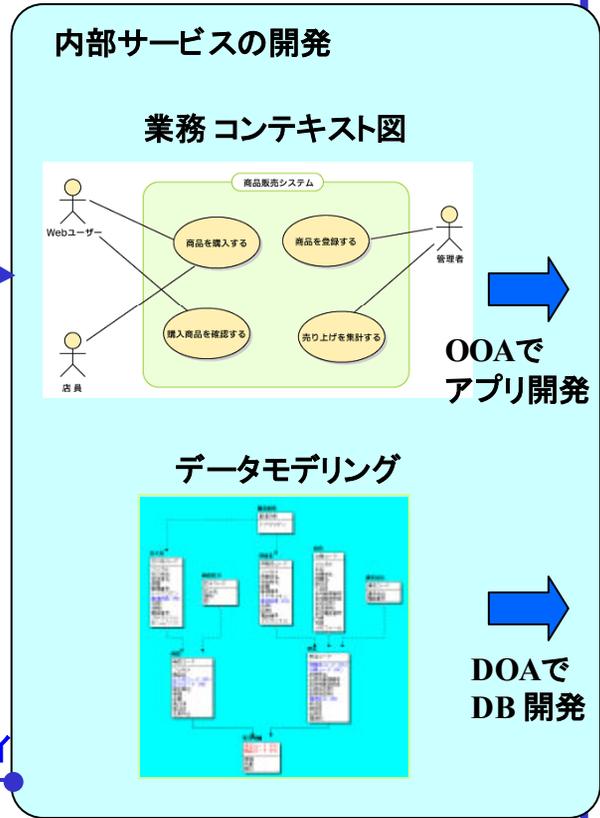
# SOA時代の新しいモデリング体系



## ビジネスプロセスモデリング (BPMN)

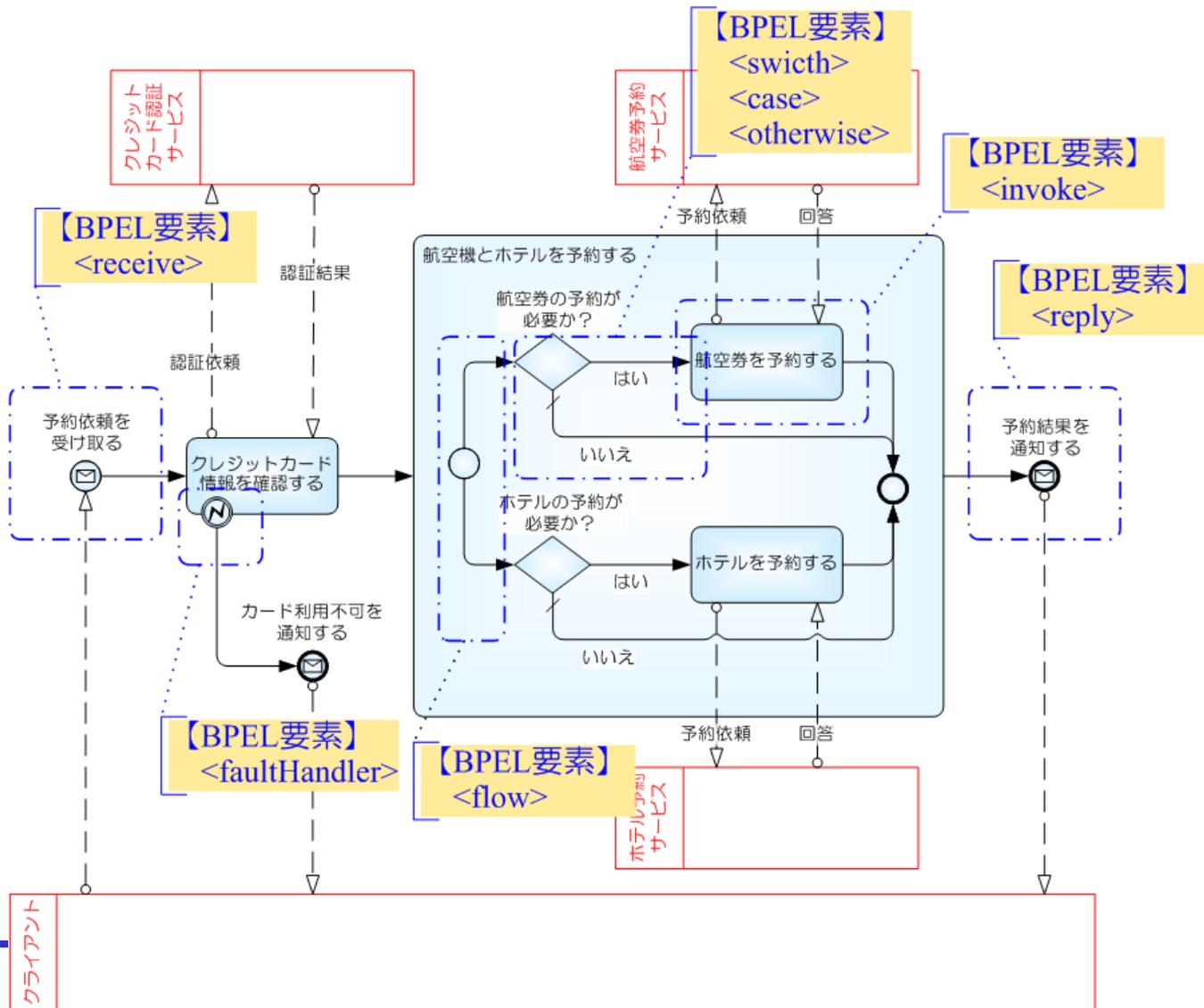


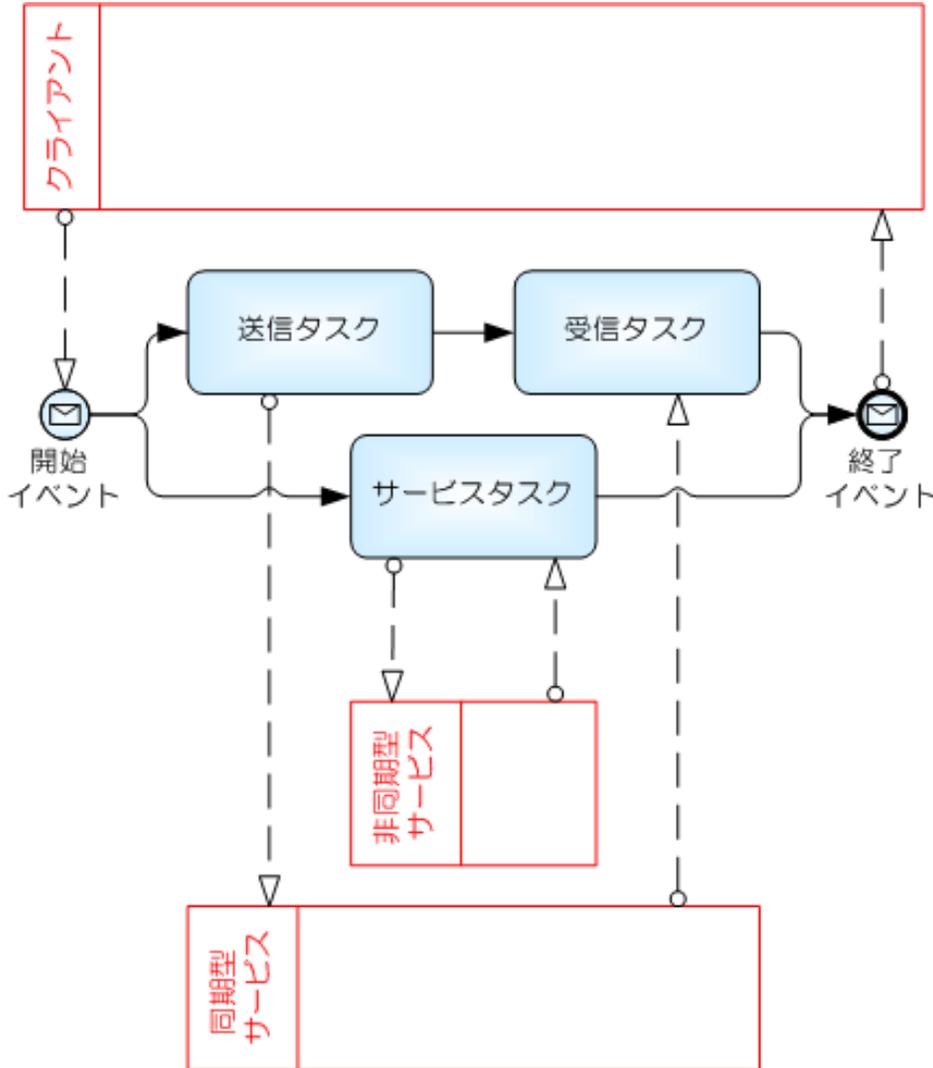
新規要求サービス



# BPMNとBPELのマッピング

# BPMNとBPELのマッピング例



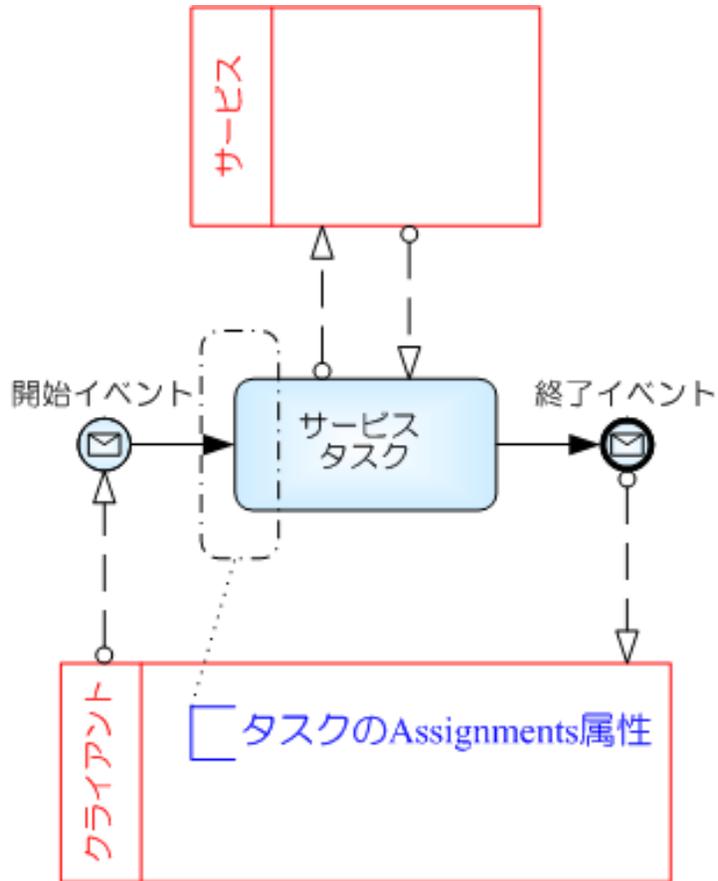


```

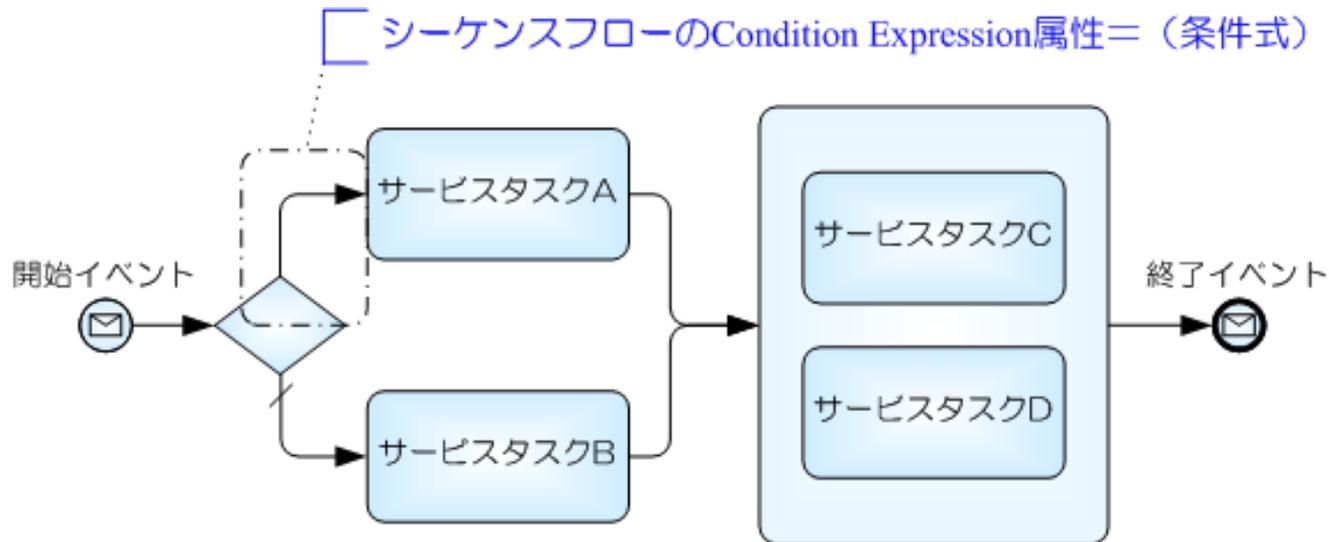
process>
<sequence>
  <receive name="開始イベント"/>
  <flow>
    <sequence>
      <invoke name="サービスタスク"/>
    </sequence>
    <sequence>
      <invoke name="送信タスク"/>
      <receive name="受信タスク"/>
    </sequence>
  </flow>
  <reply name="終了イベント"/>
</sequence>
/process>
    
```

# 主なBPEL要素

| BPEL要素                      | 説明   |
|-----------------------------|--|
| <b>sequence</b>             | 逐次実行   |
| <b>flow</b>                 | 並列実行   |
| <b>invoke</b>               | サービス呼び出し                                       |
| <b>receive</b>              | 受信   |
| <b>reply</b>                | 応答（同期型）  |
| <b>switch</b>               | 条件分岐   |
| <b>while</b>                | 繰り返し実行   |
| <b>assign</b>               | 割り当て（データ作成/変更）                                 |
| <b>pick</b>                 | イベントによる選択（ <b>onMessage</b> , <b>onAlarm</b> ） |
| <b>throw</b>                | 障害通知   |
| <b>fault handler</b>        | 障害捕捉   |
| <b>compensate</b>           | 補償（補正）通知                                       |
| <b>compensation handler</b> | 補償（補正）捕捉                                       |
| <b>scope</b>                | 実行コンテキスト                                       |
| <b>wait</b>                 | 待機（時間）   |
| <b>terminate</b>            | 即時終了   |
| <b>empty</b>                | 空のアクティビティ                                      |
| <b>correlation</b>          | 送受信メッセージ相関                                     |



```
<sequence>  
  <receive name="開始イベント"/>  
  <assign>  
    <copy>  
      <from variable="コピー元" />  
      <to variable="コピー先" />  
    </copy>  
  </assign>  
  <invoke name="サービスタスク" />  
  <reply name="終了イベント" />  
</sequence>
```

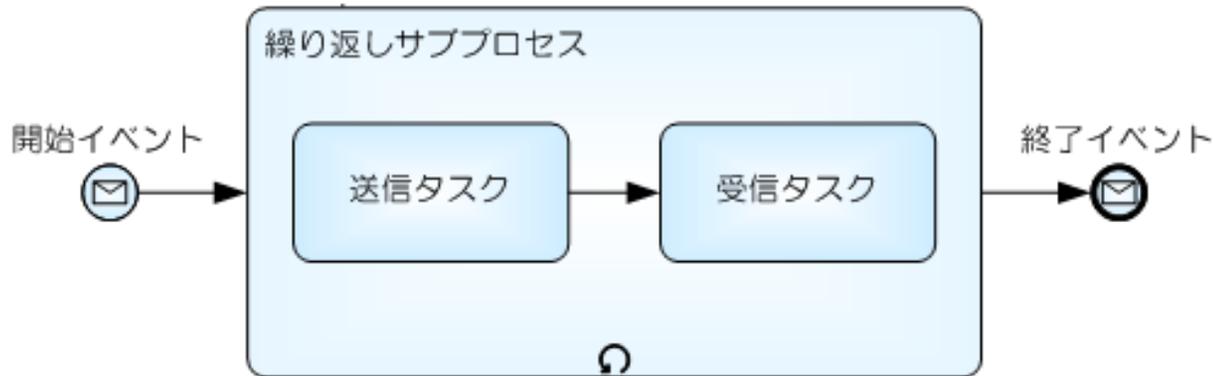


```
<switch name="XORゲートウェイ">
  <case condition="条件式">
    <invoke name="サービスタスクA"/>
  </case>
  <otherwise>
    <invoke name="サービスタスクB"/>
  </otherwise>
</switch>
```

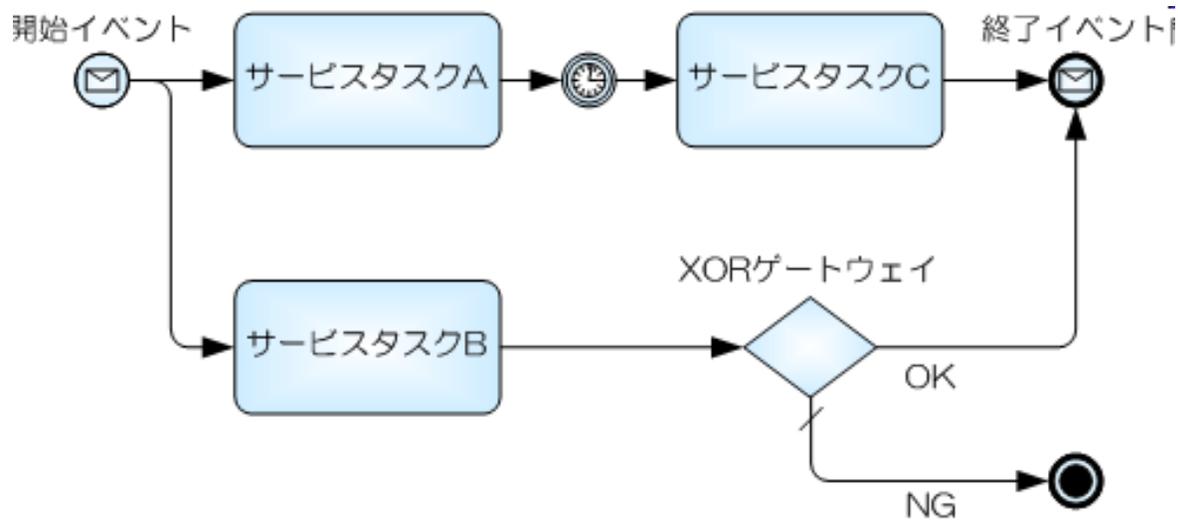
```
<flow>
  <invoke name="サービスタスクC"/>
  <invoke name="サービスタスクD"/>
</flow>
```

# while

サブプロセスのLoop Condition Expression属性= (条件式)



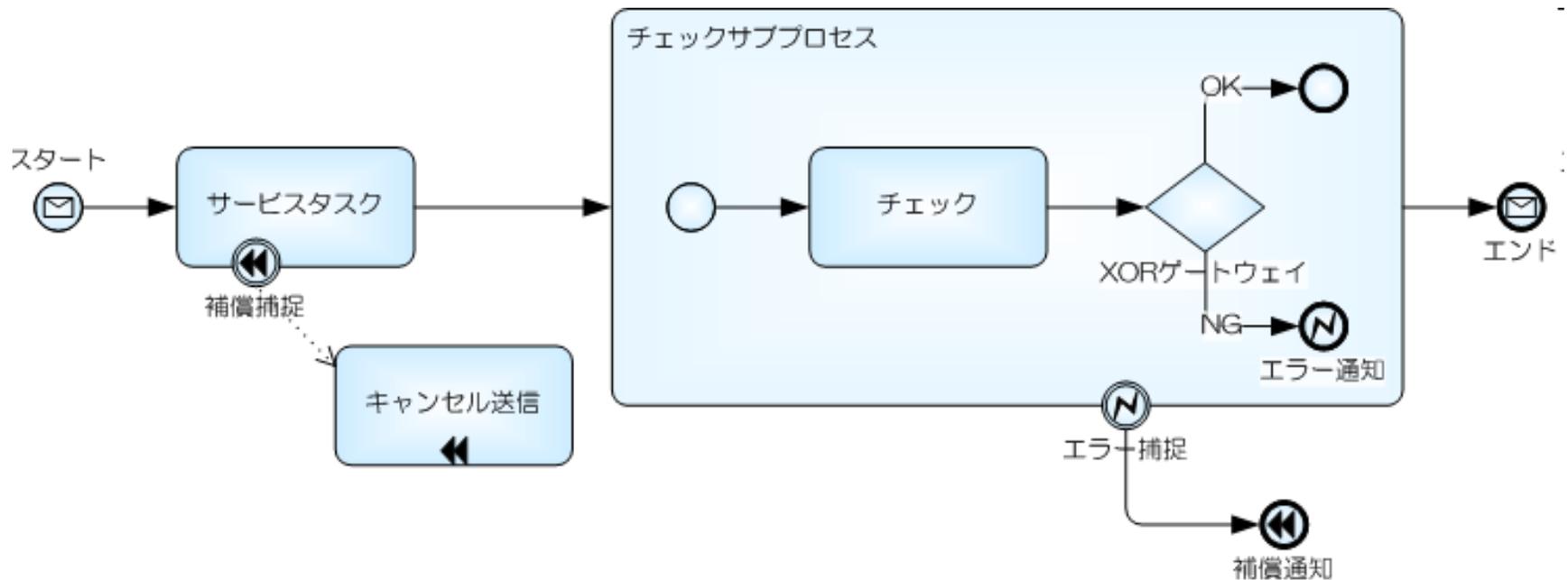
```
<while condition="条件式">
  <scope name="繰り返しサブプロセス">
    <sequence>
      <invoke name="送信タスク"/>
      <receive name="受信タスク"/>
    </sequence>
  </scope>
</while>
```



```
<sequence>  
  <invoke name="サービスタスクA"/>  
  <wait for="待ち時間" />  
  <invoke name="サービスタスクC"/>  
</sequence>
```

```
<sequence>  
  <invoke name="サービスタスクB"/>  
  <switch name="XORゲートウェイ">  
    <case condition="条件式">  
      <terminate />  
    </case>  
  </switch>  
</sequence>
```

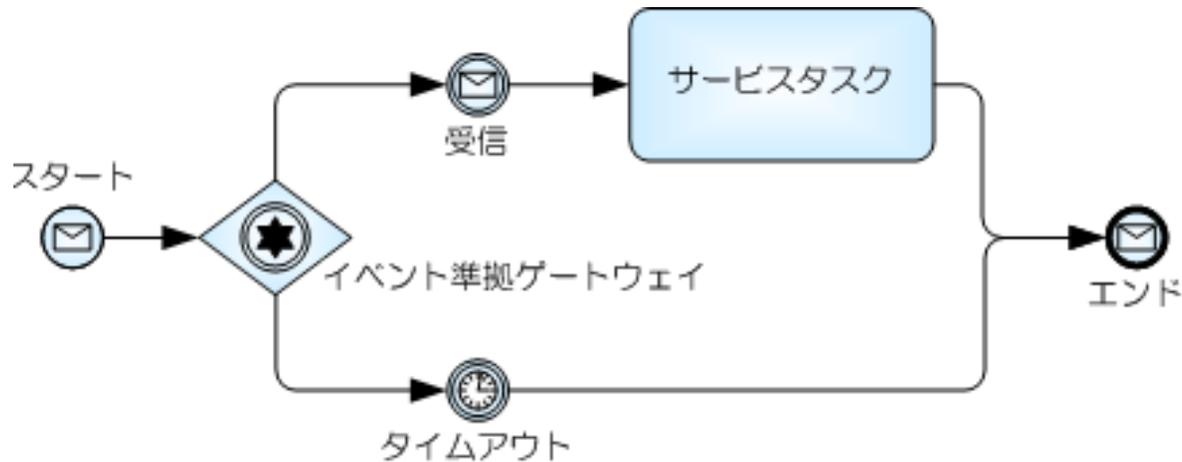
# throw / faultHandlers compensate / compensateHandlers



```
<scope name="サービスタスク">  
  <compensationHandlers>  
    <invoke name="キャンセル送信"/>  
  </compensationHandlers>  
  <sequence>  
    <invoke name="サービスタスク"/>  
  </sequence>  
</scope>
```

```
<scope name="チェックサブプロセス">  
  <faultHandlers>  
    <catch faultName="エラーコード">  
      <compensate scope="サービスタスク" />  
    </catch>  
  </faultHandlers>  
  <sequence>  
    <empty name="チェック" />  
    <switch name="XORゲートウェイ">  
      <case condition="true()">  
        <throw faultName="エラーコード" />  
      </case>  
    </switch>  
  </sequence>  
</scope>
```

# pick / onMessage / onAlarm



```
<pick name="イベント準備ゲートウェイ">  
  <onMessage partnerLink= >  
    <invoke name="サービスタスク"/>  
  </onMessage>  
  <onAlarm until="15:00">  
    <empty />  
  </onAlarm>  
</pick>
```

# BPMN研究会の活動

## ■ ワイン配送センター問題

- 各員自らBPMNのモデリングを実践し、BPMNの表記法を習得する

[1] このワイン配送センターは、ワイナリーからワインを調達し、適切な在庫を保持し、注文に応じて小売店へワインを配送することを業務としている。

[2] 原則的には、小売店からの注文は翌日配送される。また、適切な在庫水準を保つために、毎夕、ワイナリーへ必要な発注を行う。なお、請求/支払処理、売掛金/買掛金管理等の業務は、対象外とする。

[3] より具体的な業務記述は以下の通り。

[4] 当センターは、午前9時から午後5時まで、電話にて小売店からの注文を受け付ける。

[5] 小売店からの注文は、図1に示される注文受付書にて受け付ける。

[6] 各商品の未引当在庫量が常に把握されていて、小売店からの注文を受け付ける際、各注文商品に対して、十分な未引当在庫量があるかどうかチェックされる。

[7] 十分な未引当在庫量がある場合は、在庫引当が行われ、当該注文商品は、「引当済」台帳に登録される。（「引当済」台帳に登録されたものは、翌日配送される。）

[8] 十分な未引当在庫量がない場合は、小売店がバックオーダーを希望する場合のみ、注文商品は、「入荷待ち」台帳に登録されバックオーダーとなる。（「入荷待ち」台帳に登録されたものは、入荷の翌日に配送される。）

...

| 注文受付書  |       |    |
|--------|-------|----|
| 注文日付:  | _____ |    |
| 受注番号:  | _____ |    |
| 顧客番号:  | _____ |    |
| 顧客名:   | _____ |    |
| 電話番号:  | _____ |    |
| 配送先住所: | _____ |    |
| 商品番号   | 商品名   | 数量 |
|        |       |    |
|        |       |    |
|        |       |    |
|        |       |    |
|        |       |    |
|        |       |    |

図1. 小売店からの注文受付書

| 発注書    |       |    |
|--------|-------|----|
| 発注日付:  | _____ |    |
| 発注番号:  | _____ |    |
| ワイナリー: | _____ |    |
| 電話番号:  | _____ |    |
| 商品番号   | 商品名   | 数量 |
|        |       |    |
|        |       |    |
|        |       |    |
|        |       |    |
|        |       |    |
|        |       |    |

××ワイン配送株式会社  
山梨県小淵沢市・・・  
0488-26-XXXX

図2. ワイナリーへの発注書



## ■ モデリング指針の必要性

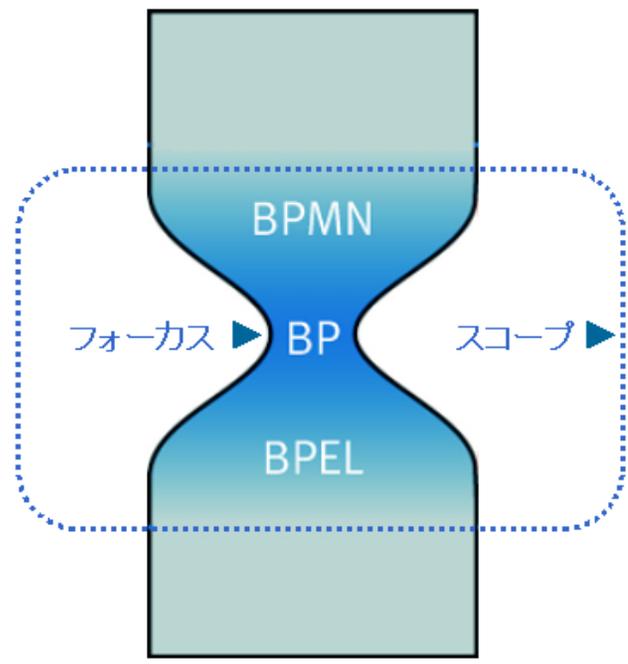
- 設計方法論（BPMN＝表記法）
- モデラーの役割
  - ◇ ビジネスアナリスト（ハイレベルBPMNモデル）
  - ◇ プロセスデザイナー（ミドルレベルBPMNモデル）
  - ◇ システムアーキテクト（ローレベルBPMNモデル）
- モデリングの手本
  - ◇ モデリングパターン
  - ◇ 参照モデル
  - ◇ サンプルモデル

## BPMI.org BPMI.org の砂時計

参画者:

- 戦略コンサルタント
- ビジネスアナリスト
- プロセスデザイナー
- システムアーキテクト
- ソフトウェアエンジニア

ビジネス環境



目的:

↑  
モデリング

↓  
実行

# モデリングの目的と視点

## モデリングの目的

## 主なモデリングの視点

ハイレベルBPMNモデル

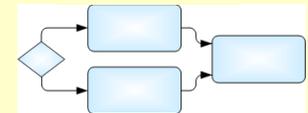
ビジネスアナリスト

ビジネスプロセス  
の可視化

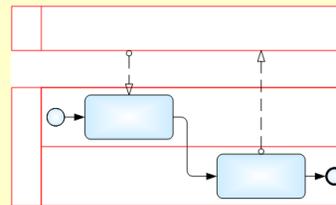
作業の順序性



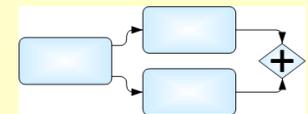
条件判断



プロセス内外における  
役割分担と相互作用



作業の同時並行性



ミドルレベルBPMNモデル

プロセスデザイナー

ビジネスプロセスの  
仮説検証、モニタリング

- ・ 作業の開始/終了タイミング（イベント）
  - ・ 繰り返す作業（ループ）
- などを含めたプロセスの詳細化

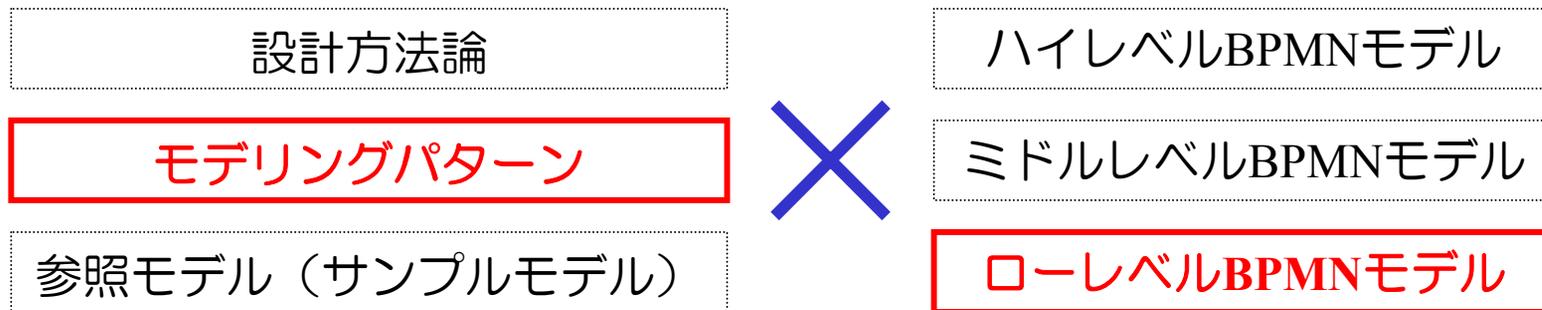
ローレベルBPMNモデル

システムアーキテクト

BPELの生成

- ・ Webサービスとのやり取り
- ・ BPEL生成に必要なBPMN属性の定義
- ・ システム的な例外や補償処理

## ■ ローレベルBPMNパターン



## ■ 目的

- モデリング指針の明示
- 技術習得のためのドキュメント
  - ◇ BPMN表記法
  - ◇ BPMNとBPELのマッピング

# 目次とパターン一覧【案】

## 【目次】

1. 本書の目的/位置づけ
2. BPMN概説
3. BPEL概説
4. ビジネスプロセス設計プロセス  
(ビジネスプロセス図における3つの記述レベル)
5. ローレベルBPMNパターン
  5. 1. パターンを分類するカテゴリ
  5. 2. パターン一覧
  5. 3. 記述項目の説明
  5. 4. パターン集
6. BPELにマッピングされないBPMN要素

| カテゴリ名     | No | パターン名         | 備考                     |
|-----------|----|---------------|------------------------|
| 基本制御      | 1  | 逐次実行          |                        |
|           | 2  | 並列実行(AND)     |                        |
| 経路選択      | 3  | 排他的選択(XOR)    |                        |
|           | 4  | 包含的選択(OR)     |                        |
|           | 5  | イベント選択        |                        |
| 経路併合      | 6  | 同期結合(AND)     |                        |
|           | 7  | 排他的併合(XOR)    |                        |
|           | 8  | 包含的併合(OR)     |                        |
| ループ制御     | 9  | 標準ループ         |                        |
|           | 10 | マルチインスタンスループ  | MI_Ordering=Sequential |
| イベント制御    | 11 | メッセージイベント待機   | ルール、リンクを含む             |
|           | 12 | タイマーイベント待機    |                        |
| 例外制御      | 13 | 例外メッセージ       |                        |
|           | 14 | 例外タイマー        |                        |
|           | 15 | エラー処理         | ※パターン分けが必要か?           |
|           | 16 | 補償処理          |                        |
| 終了制御      | 17 | (一部)フロー終了     |                        |
|           | 18 | プロセス即時終了      |                        |
| コミュニケーション | 19 | 非同期型サービス受信/応答 |                        |
|           | 20 | 同期型サービス受信/応答  |                        |
|           | 21 | 非同期型サービス呼出    |                        |
|           | 22 | 同期型サービス要求/受信  |                        |

# パターン集の雛型【案】

|           |  |
|-----------|--|
| カテゴリ      | 経路選択   |
| パターン名称    | イベント選択   |
| 目的        | 複数のイベント発生を待ち、最初に発生したイベントにより後続パスを決定する振る舞いを表現するために使用する。イベントが発生すると、その他イベントの後続パスは無効になる排他的な選択を表す。   |
| ビジネスプロセス図 | <p>ダイアグラム</p>  |
|           | <p>シナリオ</p> <ol style="list-style-type: none"> <li>(1) 「メッセージ中間イベント A」「メッセージ中間イベント B」「タイマー中間イベント C」の発生を待つ</li> <li>(2) 最初に発生したイベントにより後続パスの何れに進むかを決定する             <ol style="list-style-type: none"> <li>① 「メッセージ中間イベント A」が最初に発生した場合には「タスク A」を実行する</li> <li>② 「メッセージ中間イベント B」が最初に発生した場合には「タスク B」を実行する</li> <li>③ 「タイマー中間イベント C」が最初に発生した場合には「タスク C」を実行する</li> </ol> </li> </ol> |

|         |  |
|---------|--|
| BPELコード | <pre> &lt;pick name="イベント準拠 XOR ゲートウェイ" createInstance="no"&gt;   &lt;onMessage partnerLink="*****" portType="*****"     operation="*****" variable="*****"&gt;     &lt;sequence&gt;       &lt;empty name="タスク A" /&gt;     &lt;/sequence&gt;   &lt;/onMessage&gt;   &lt;onMessage partnerLink="*****" portType="*****"     operation="*****" variable="*****"&gt;     &lt;sequence&gt;       &lt;empty name="タスク B" /&gt;     &lt;/sequence&gt;   &lt;onAlarm for="*****"&gt;     &lt;sequence&gt;       &lt;empty name="タスク C" /&gt;     &lt;/sequence&gt;   &lt;/onAlarm&gt; &lt;/pick&gt; </pre> |
| 特記事項    | <p><b>【BPMN 仕様】</b></p> <ul style="list-style-type: none"> <li>イベント準拠 XOR ゲートウェイに後続する中間イベントは、メッセージ、タイマーの他、ルール、リンク、エラーを使用することができる。</li> <li>メッセージ中間イベントを TaskType 属性が Receive であるタスクで代替することができる。</li> </ul> <p><b>【BPEL マッピング】</b></p> <ul style="list-style-type: none"> <li>イベント準拠 XOR ゲートウェイに後続するイベントとして、ルール中間イベント、リンク中間イベントを使用した場合には、ルールが満たされたこと、またはリンクにトークンが発生したことをメッセージとして受け取ることを前提としている。BPEL マッピングはメッセージ中間イベントと同様、<code>&lt;onMessage&gt;</code>にマッピングされる。</li> <li>BPMN 仕様ではイベント準拠 XOR ゲートウェイに後続するイベントとしてエラー中間イベントを使用できるが、BPEL マッピングの対象外となる。</li> </ul>                                       |

- □ローレベルBPMNパターン
  - 2006年1月12日 初版レビュー
  - 2006年3月末 第1.0版
    - ◇ XMLコンソーシアム（ビジネスイノベーション研究部会&SOA部会）との相互検証
- 2006年度の研究【候補】
  - ミドルレベルBPMNモデルを対象
    - ◇ ローレベルBPMNパターンの検証（ミドルレベルBPMNモデルへの適用）
    - ◇ ミドルレベルBPMNパターン化