OASIS BPEL Webinar: Frank Leymann Input

(OASIS Webinar, March 12th, 2007)

Prof. Dr. Frank Leymann
Director, Institute of Architecture of Application Systems
Former IBM Distinguished Engineer
BPEL’s Role in SOA
BPEL in the Web Service Stack

We are here!
A Typical Situation

Often, you need more than one service to achieve your goal!

Which ones to use? How to use them? In which order? ...

© Frank Leymann
BPEL Prescribes Proper Use of Web Services

Wow! This is easy 😊
...and Externalizes Processes as Web Services
BPEL in SOA: Abstract View

“BPEL is a recursive aggregation model for Web services”

- **Aggregation**: A set of Web services can be tied into one or more new Web service by means of a business process model

- **Recursive**: These new Web services can again be tied into other new Web services
BPEL in a Nutshell: Elevator Speech

- BPEL describe in a SOA how your company performs its business processes

- With BPEL, it is straightforward to let your business partners and customers directly participate in your business processes

- With BPEL, it is straightforward to tie in Web services as activities of your business processes
BPEL and Business Process Modeling
A Well-Known Problem

How to make money?
Capital & Work Force,
Business Models,
Organizational Structures

Executive Focus

Business-IT-Gap

Application Systems,
Transactions & Data,
Hardware Infrastructure

How to support business with IT?

IT Personnel Focus
Key Aspect of a Solution

<table>
<thead>
<tr>
<th>Executive Focus</th>
<th>Business-IT-Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>How to make money?</td>
<td>Business Processes</td>
</tr>
<tr>
<td>Capital &amp; Work Force,</td>
<td>IT Personnel Focus</td>
</tr>
<tr>
<td>Business Models,</td>
<td>Application Systems,</td>
</tr>
<tr>
<td>Organizational Structures</td>
<td>Transactions &amp; Data,</td>
</tr>
<tr>
<td>Hardware Infrastructure</td>
<td>How to support business with IT?</td>
</tr>
</tbody>
</table>

© Frank Leymann
Today’s Landscape for Process Management

**Business-Level**

- Modeling Tool₁
- Modeling Tool₂
- Modeling Tool₃
- …
- Modeling Toolₙ

How to make models run on particular engine?

**Managing Business Processes**

- Process Engine₁
- Process Engine₂
- …
- Process Engineₖ

**IT-Level**
What’s the Problem?

- Vast amount of business level methodologies and corresponding tools
  - …which will never go away… (see next slide)
- Process engines have historically grown up
- Process model must be transformed into a format that can be executed within the process engine of your favorite vendor
- Different vendors process engines support different capabilities
  - …different syntax, terminology, concepts,…
  - …different operational semantics…
  - The real problem!
- …And all that is different from business level syntax, terminology, concepts, operational semantics,…
- Thus, transformation from business level into IT level always loses information
- Business level tool vendor has to support multiple transformations
Business-Level Modeling

- Business-level modeling comes with a whole methodology, often developed by business schools.
- Different methodologies compete (“method war” 😊)
- Number of business-level methodologies is very large
  - Chances to unify them is very low
- Very likely, there will always be different business-level modeling tools.
What BPEL Brings to the Table

Business-Level

Modeling Tool\textsubscript{1} \quad \text{Modeling Tool}\textsubscript{2} \quad \text{Modeling Tool}\textsubscript{3} \quad \ldots \quad \text{Modeling Tool}\textsubscript{n}

IT-Level

BPEL Tool

BPEL Tool\textsubscript{1} \quad \text{BPEL Tool}\textsubscript{2} \quad \text{BPEL Tool}\textsubscript{k}

BPEL Engine

BPEL Engine\textsubscript{1} \quad \text{BPEL Engine}\textsubscript{2} \quad \text{BPEL Engine}\textsubscript{k}
BPEL Value: First Glance

- Single business process language for IT-users
  - Single runtime model
  - Single IT-level tooling
  - Single target model all business-level tools can map too

- Reuse of IT-level process models across tools and process engines

- Investment protection
- Reduced total cost of ownership: Simplification of overall infrastructure
Did I Say “Table”?

“BPEL is for process/workflow technology what SQL is for relational databases!”
BPEL and Tool Interoperability
BPEL & Common Operational Semantics

Deployment

BPEL Tool

.BPEL

BPEL Engine_{1}

BPEL Engine_{2}

... 

BPEL Engine_{k}
Usage of Abstract Processes
Hiding Process Details
BPEL Abstract & Executable Processes
Abstract Processes

- An **abstract process** describes “behavior” only
- It may not be executable
- It may omit certain information
  - Omitted information represent modeling artifacts that may be provided later (“completion”)

```
“omission”
```

```
executable process
```

```
abstract process
```

© Frank Leymann
Use Cases Categories (“Pattern”)

- **View** on internal process
  - Only a projection of an internal (executable) process is made visible to the outside
    - …to protect process model as corporate asset
    - …to hide non-optimal parts of a process model

- **Template** as “best practice”
  - Specification of common activities, major data structures, and main control flow
  - Must be refined into an executable processes on a case-by-case basis

- **Constraints** on message exchange
  - Specification about the order in which messages are consumed or produced
    - Business functionality is implemented as a (set of) port types, and operations must be used in a certain order to achieve intended business goal
View ("Export")

- An abstract process is derived from an executable by abstracting away parts that are not part of the behavior one wishes to expose.

- Example:
  - Show a particular business partner the interactions that the partner must follow.
    - Interactions with all other partners are dropped.
  - Use an abstract process to represent common behavior in a set of executables, and drop any non-repeated behavior.
    - An executable process of a more general business model may need parts tagged as points of variability, and those are made explicitly opaque.
Template (“Import”)

- An abstract process is basis to create one or more executables, or more detailed abstract processes
- Example:
  - One needs to create an implementation of an abstract process provided as a behavioral prescription for complying with a known, domain-specific business function
  - Multiple abstract processes can be created in a series of iterative refinements to a design
  - One wants to implement “best practices” while maintaining some company specifics
    - The abstract process may have been purchased from a consulting firm, as a model of an optimized approach to a problem
Service Usage Constraint

- Typically, the operations of a service may not be used in order
  - E.g. it doesn’t make sense to use the `Cancel` operation of an `Order` service before using its `Buy` operation 😊
- To describe such ordering constraint an abstract process can be used that only refers to operations of a single port type
- The port type of a service may be associated with a process which describes the order in which the operations of the port type can be used
End of Document