Semantic Business Process Management

Lecture 3 – BPMN and BPDM

Prof. Dr. Adrian Paschke

Corporate Semantic Web (AG-CSW)
Institute for Computer Science, Freie Universitaet Berlin
paschke@inf.fu-berlin.de
http://www.inf.fu-berlin.de/groups/ag-csw/
Overview

- OMG BPMN 1.x
- Outlook BPMN 2.0
- OMG BPDM 1.0
BPMN

- Business Process Modeling Notation (BPMN) is a graphical representation for specifying business processes as a workflow
  - flowcharting technique quite similar to UML activity diagrams
  - simple diagrams with a small set of graphical elements
    - easy for business users as well as developers to understand the flow and the process
- Developed by Business Process Management Initiative (BPMI)
- Standardized by OMG
  - (BPMI merged with OMG)
- BPMN 1.2 since January 2009
  - major revision process for BPMN 2.0 in progress
Business Process Standards

[Martin Bartonitz/Saperion, 2009]
Basic Types of BPMN Sub-models

- **Private (internal) business processes**
  - Are those internal to a specific organization and are the type of processes that have been generally called workflow or BPM processes.

- **Abstract (public) processes**
  - Represents the interactions between a private business process and another process or participant. Only those activities that communicate outside the private business process are included in the abstract process.

- **Collaboration (global) processes**
  - Depict the interactions between two or more business entities. These interactions are defined as a sequence of activities that represent the message exchange patterns between the entities involved.
Four Basic Categories of Elements

- **Flow Objects**
  - Events, Activities, Gateways

- **Connecting Objects**
  - Sequence Flow, Message Flow, Association

- **Swimlanes**
  - Pool, Lane

- **Artifacts (Artefacts)**
  - Data Object, Group, Annotation
Flow Objects - Events

- **Event**
  - An Event is represented with a circle and denotes something that happens.
  - Icons within the circle denote the type of event (e.g. envelope for message, clock for time).
    - **Catching** Events (e.g. incoming message to Start the process)
    - **Throwing** Events (throw a message at the End of the process)
- **Start event**
  - Acts as a trigger for the process; indicated by a single narrow border; and can only be *Catch*, so is shown with an open (outline) icon.
- **End event**
  - Represents the result of a process; indicated by a single thick or bold border; and can only be *Throw*, so is shown with a solid icon.
- **Intermediate event**
  - Represents something that happens between the start and end events; is indicated by a tramline border; and can *Throw or Catch* (using solid or open icons as appropriate).
Flow Objects - Activities

- **Activity**
  - An Activity is represented with a rounded-corner rectangle and describes the kind of work which must be done.

- **Task**
  - A task represents a single unit of work that is not or cannot be broken down to a further level of business process detail.

- **Sub-process**
  - Used to hide or reveal additional levels of business process detail
    - when collapsed a sub-process is indicated by a plus sign against the bottom line of the rectangle;
    - when expanded the rounded rectangle expands to show all flow objects, connecting objects, and artefacts.
  - Has its own self-contained start and end events, and sequence flows from the *parent* process must not cross the boundary.
Flow Objects - Gateway

- Activity (2)
  - **Transaction**
    - A form of sub-process in which all contained activities must be treated as a whole, i.e., they must all be completed to meet an objective, and if any one of them fails they must all be compensated (**undone**).
    - Transactions are differentiated from expanded sub-processes by being surrounded by a tramline border.
  - **Gateway**
    - A Gateway is represented with a diamond shape and will determine forking and merging of paths depending on the conditions expressed.
Connecting Objects

- Connecting objects
  - Sequence Flow
    - solid line and arrowhead: shows in which order the activities will be performed.
    - The sequence flow may also have a symbol at its start, a small diamond indicates one of a number of conditional flows from an activity while a diagonal slash indicates the default flow from a decision or activity with conditional flows.
  - Message Flow
    - dashed line, an open circle at the start, and an open arrowhead at the end.
    - Messages flow across organisational boundaries (i.e., between pools)
    - A message flow can never be used to connect activities or events within the same pool.
  - Association
    - dotted line; associate an Artifact or text to a Flow Object
    - can indicate some directionality using an open arrowhead (toward the artifact to represent a result, from the artifact to represent an input, and both to indicate it is read and updated).
    - No directionality when the Artifact or text is associated with a sequence or message flow
Swimlanes

- Swimlanes
  - visual mechanism of organizing and categorizing activities, based on cross functional flowcharting
- Pool
  - Represents major participants in a process, typically separating different organizations.
  - A pool contains one or more lanes
  - A pool can be open (i.e., showing internal detail) when it is depicted as a large rectangle showing one or more lanes, or collapsed (i.e., hiding internal detail) when it is depicted as an empty rectangle stretching the width or height of the diagram.
- Lane
  - Used to organize and categorize activities within a pool according to function or role, and depicted as a rectangle stretching the width or height of the pool.
  - A lane contains the Flow Objects, Connecting Objects and Artifacts.
Artifacts

- Artifacts allow developers to bring some more information into the model/diagram.
- **Data Objects**
  - Data Objects show the reader which data is required or produced in an activity.
- **Group**
  - A Group is represented with a rounded-corner rectangle and dashed lines. The Group is used to group different activities but does not affect the flow in the diagram.
- **Annotation**
  - An Annotation is used to give the reader of the model/diagram an understandable impression.
Example - A Process with Normal Flow

- Working Group Active
- Friday at 6 PM Pacific Time
- Check Status of Working Group
- Working Group Still Active?
  - Yes: Send Current Issue List
  - No: Issue List

[Diagram showing the flow from Working Group Active to sending the issue list if the working group is active.]

(Corporate Semantic Web logo in the bottom right corner.)
Example 2 - Collect Votes

1. Check Calendar for Conference Call
2. Conference Call in Voting Week?
   - Yes: Wait until Thursday, 9am
     - Moderate Conference Call Discussion
   - No: Collect Votes
3. Moderate E-mail Discussion
4. Delay 6 Days
5. Moderate E-mail Discussion
6. E-Mail Vote Deadline Warning
7. Receive Vote
8. Increment Tally
9. Vote
10. Vote Tally
Example 3 - Discussion Cycle Process

1. Issue Voting List [0 to 5 Issues]
2. Announce Issues for Discussion
3. Delay 6 days from Announcement
4. E-Mail Discussion Deadline Warning
5. Conference Call in Discussion Week?
   - Yes: Conference Call Discussion
   - No: Wait until Thursday, 9am
4. Check Calendar for Conference Call
5. Calendar
6. 7 Days
7. Moderate E-mail Discussion
8. Evaluate Discussion Progress
9. This Task returns the value of the Discussion Over to True or False
10. Allow 1 week for the discussion of the issues - through e-mail or calls

The Sub-Process will repeat of the Discussion Over variable is False
BPMN 2.0
### BPMN 2.0

#### Events Overview

<table>
<thead>
<tr>
<th>Types</th>
<th>Start</th>
<th>Intermediate</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td><img src="none.png" alt="Image" /></td>
<td><img src="none.png" alt="Image" /></td>
<td><img src="none.png" alt="Image" /></td>
</tr>
<tr>
<td>Message</td>
<td><img src="message.png" alt="Image" /></td>
<td><img src="message.png" alt="Image" /></td>
<td><img src="message.png" alt="Image" /></td>
</tr>
<tr>
<td>Timer</td>
<td><img src="timer.png" alt="Image" /></td>
<td><img src="timer.png" alt="Image" /></td>
<td><img src="timer.png" alt="Image" /></td>
</tr>
<tr>
<td>Error</td>
<td><img src="error.png" alt="Image" /></td>
<td><img src="error.png" alt="Image" /></td>
<td><img src="error.png" alt="Image" /></td>
</tr>
<tr>
<td>Escalation</td>
<td><img src="escalation.png" alt="Image" /></td>
<td><img src="escalation.png" alt="Image" /></td>
<td><img src="escalation.png" alt="Image" /></td>
</tr>
<tr>
<td>Cancel</td>
<td><img src="cancel.png" alt="Image" /></td>
<td><img src="cancel.png" alt="Image" /></td>
<td><img src="cancel.png" alt="Image" /></td>
</tr>
<tr>
<td>Compensation</td>
<td><img src="compensation.png" alt="Image" /></td>
<td><img src="compensation.png" alt="Image" /></td>
<td><img src="compensation.png" alt="Image" /></td>
</tr>
<tr>
<td>Conditional</td>
<td><img src="conditional.png" alt="Image" /></td>
<td><img src="conditional.png" alt="Image" /></td>
<td><img src="conditional.png" alt="Image" /></td>
</tr>
<tr>
<td>Link</td>
<td><img src="link.png" alt="Image" /></td>
<td><img src="link.png" alt="Image" /></td>
<td><img src="link.png" alt="Image" /></td>
</tr>
<tr>
<td>Signal</td>
<td><img src="signal.png" alt="Image" /></td>
<td><img src="signal.png" alt="Image" /></td>
<td><img src="signal.png" alt="Image" /></td>
</tr>
<tr>
<td>Terminate</td>
<td><img src="terminate.png" alt="Image" /></td>
<td><img src="terminate.png" alt="Image" /></td>
<td><img src="terminate.png" alt="Image" /></td>
</tr>
<tr>
<td>Multiple</td>
<td><img src="multiple.png" alt="Image" /></td>
<td><img src="multiple.png" alt="Image" /></td>
<td><img src="multiple.png" alt="Image" /></td>
</tr>
<tr>
<td>Parallel</td>
<td><img src="parallel.png" alt="Image" /></td>
<td><img src="parallel.png" alt="Image" /></td>
<td><img src="parallel.png" alt="Image" /></td>
</tr>
<tr>
<td>Flows</td>
<td><img src="flows.png" alt="Image" /></td>
<td><img src="flows.png" alt="Image" /></td>
<td><img src="flows.png" alt="Image" /></td>
</tr>
</tbody>
</table>
Inclusive Gateways

- BPMN 1.x
  - Wait until no more token can arrive
  - Not clear in complicated cases (e.g. loops) to wait for which token
- BPMN 2.0
  - Defines scopes in which the process needs to wait for tokens:

  „...there is no directed path (formed by Sequence Flow) from a Token to this sequence flow unless:
  - the path visits the inclusive gateway or
  - the path visits a node that has a directed path to a non-empty incoming sequence flow of the inclusive gateway. „

BPMN2.0-v.0.9.12
Complex Gateways

- BPMN 1.x
  - "synchronization is based on a textual description"
- BPMN 2.0
  - state (activationCount, WaitingForStart)
  - Formal expression for synchronization (activationCondition)
  - Conditions on outgoing link
  - Second synchronization when reset
  - Or-Join semantics avoids deadlocks
Example

- a
- b
- c

- isWaitingForStart=true
  - x
- isWaitingForStart=false
  - y
- activationCount = 2
  - z

activationCount: 0
isWaitingForStart: true
activationCondition: activationCount > 1
Example
Example

```
activationCount: 3
isWaitingForStart: false
activationCondition: activationCount > 1
```
Event Sub-Process

- **BPMN 1.x**
  - No optional execution of independent sub processes

- **BPMN 2.0**
  - Embedded in subprocesses
  - Access on surrounding context
  - Enables **BPEL Event-Handler** semantics in BPMN
# Event Escalation

## Table 10-91 – Types of Events and their Markers

<table>
<thead>
<tr>
<th>Types</th>
<th>Start</th>
<th>Intermediate</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top-Level</td>
<td>Event Sub-Process Interrupting</td>
<td>Event Sub-Process Non-Interrupting</td>
</tr>
<tr>
<td>None</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Message</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Timer</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Error</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Escalation</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Cancel</td>
<td></td>
<td></td>
<td>○</td>
</tr>
<tr>
<td>Compensation</td>
<td></td>
<td></td>
<td>○</td>
</tr>
</tbody>
</table>
Event Sub-Process - Types

isInterrupting=false:

isInterrupting=true:
Example
Multiple Start Events

- BPMN 1.x
  - Exclusive semantics
  - Start with several events unclear

- BPMN 2.0
  - Correlation via conversation
  - Special event-based gateway
  - Special start events
Examples

- **Exclusive:**
  - Each received event starts a new process

- **Joint:**
  - Multiple events belong to one process instance
Business Rules Task

Figure 10.8 – A Task object

Figure 10.9 – A Send Task Object

Figure 10.10 – A User Task Object

Figure 10.11 – A Service Task Object

Figure 10.12 – A Receive Task Object

Figure 10.13 – A Manual Task Object

Figure 10.14 – A Script Task Object

Figure 10.15 – A Business Rule Task Object

\[\text{BusinessRuleTaskImplementation = BusinessRuleWebService | WebService | Other | Unspecified}\]
### 2 New Diagram Types in BPMN 2.0

<table>
<thead>
<tr>
<th>Conversation Diagram</th>
<th>Choreography Diagram</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview of complex interaction scenarios</td>
<td>Single Choreographies</td>
</tr>
<tr>
<td>Representation of (all) partner</td>
<td>Ordered sequence of concrete interactions</td>
</tr>
<tr>
<td>Hierarchical modeling of the communication</td>
<td>Defines communication contract between involved partners</td>
</tr>
</tbody>
</table>

„Which conversations between which partners?“

„How to execute the conversation?“

- Choreographies useful extension for distributed processes
- Defines the paths of communication
- Choreographie = globale view
BPMN Conformance

- Process Modelling
- Choreography Modelling
- Process Execution
- BPEL Process Execution
Conformance - Process Modelling
Conformance - Choreography Modelling
Conformance - Process Execution and BPEL Execution

- Execution of BPMN processes
- Activity life cycle
- Interchange format

- Implicit „Process Execution“
- BPMN to BPEL mapping
- „Basic Mapping“ required
  - BPEL Import not required
BPMN 2.0 Structure - Summary Overview
Summary BPMN 2.0

- Conformance Level can be used for evaluating tools with respect to modeling and execution capabilities
- BPMN 2.0 to BPEL mapping is a full transformation
  - (but not vice versa)
  - Statements about the assumptions about a process
  - Detailed mapping down to the technical level
Weakness of BPMN

- BPMN only supports graphical modeling of business processes, but not e.g.:
  - Organization models
  - Data structures
  - Modeling of strategies
  - Business rules
- Ambiguity and confusion in sharing BPMN models
- Support for routine work
- Support for knowledge work, and
- Converting BPMN models to executable environments
  - Roundtripping not part of the standard
OMG BPDM
BPDM 1.0

- **Business Process Definition Metamodel (BPDM)** is a standard definition of concepts used to express business process models
  - relies on a formal method - PSL (Process Specification Language) - a first order logic language to ensure execution consistency of processes
- OMG standard version 1.0
  - BPDM 1.1. revisions under way
- **BPDM provides BPMN with:**
  - An explicit metamodel
  - A serialization mechanism for BPMN concepts
  - Rigorous execution semantic
Goals of BPDM

- Provide a common semantics basis for all process oriented models
- Provide support for the service oriented world
- Integrate rules within processes
- Ensure Execution Interoperability of process models
- Use BPMN as the standard notation for processes
- Leverage other “process” knowledge: UML, BPMN, PSL
BPDM Language Structure

- Two complementary process models
  - Activity Model
    - BPMN Extensions
  - Interaction Protocol Model
- Common Process Elements
  - Happening (Event)
  - Processing Behavior (Steps)
  - Simple Interaction
- Common abstractions
  - Composition Model
  - Course Model
- Use of UML2 infrastructure
  - Element / Type
  - Data Types
  - Primitives Types
  - Package
  - Expressions
Composition Metamodel

- Consistent framework for organizing complex structures
- Ensure uniform execution of models across tools
Course Model

- **Course Model**
  - Share common constructs
  - Differentiate elements representing runtime occurrences (e.g., activities) from elements that constrain the temporal ordering of those occurrences (sequencing, gateways)

- **Used across two parts of the metamodel**
  - Process life cycle events (start/stop)
  - Process Steps (Interactions, Activities)
Three dimensions of Process Analysis

- **Processes happen over time**, 
  - produce changes (start, stop and other changes) 
  - react to changes 

- **Processes specify work to be done** 
  - They have processing steps that occur and can be organized in series 

- **Processes communicate with their environment** 
  - Their delivered services: inputs and outputs 
  - Their required services.
Summary BPDM

- Provides a clear semantics
- A complete coverage of Activity Modeling
  - Using BPMN as its notation
  - Providing clarification for BPMN Execution semantic
- A consistent approach of interactions and activities providing
  - support for choreography
  - support for activities
  - consistent binding between the two
  - support for integrating rules
  - support for process specification versus process realization
- Together, BPDM and BPMN will help leverage process management maturity
Questions?

References

- BPDM 1.0 [http://www.omg.org/spec/BPDM/1.0/](http://www.omg.org/spec/BPDM/1.0/)
- BPMN 2.0 [http://www.omg.org/spec/BPMN/1.2/](http://www.omg.org/spec/BPMN/1.2/)
- BPDM Links [http://www.conradbock.org/#BPDM](http://www.conradbock.org/#BPDM)