Business Process Mapping using the BPMN 2.0 Modelling Notation

Tutorial
Contents

◆ BPMN and Business Process Diagrams (BPD)
◆ Private Processes
◆ Start and End Events
◆ Activities and Sequence Flows
◆ Exclusive Gateways
◆ Intermediate Events
◆ Public Processes, Pools and Lanes
◆ Message Flows
◆ Collaboration Processes
◆ Sub-processes
◆ Parallel Gateways
◆ Process Steps as Tasks
◆ Human versus System Pools
◆ Choreographies, Conversations and BPEL
The Business Process Model and Notation

- Syntax only
- Process independent
- Tool independent
- Developed by the OMG
- Also responsible for
  - UML
  - SysML
- Version 1.0 - May 2004 - adopted 6/2/06
- Version 1.1 - Feb 2008
- Version 2.0 - Jan 2011 - current
Business Process Diagram (BPD)

- Private Process
- Start and End Events
- Activities
- Sequence Flows
- Exclusive Gateways
- Intermediate Events

BPMN and the BPMN logo are trademarks of the Object Management Group
Private Processes

- **Internal to a specific organisation**
- **Does not show explicit interaction with external entities**
- **Non-executable**
  - Modelled for the purpose of documenting process behaviour at a modeller-defined level of detail
- **Executable**
  - Modelled for the purpose of being executed in business process management or other system
Start and End Events

◆ An event is instantaneous
  ● Mostly a status phrase e.g.
  ● “Customer Enquiry Received”
  ● “Sale Complete”
  ● Maybe a command e.g.
  ● “Start Process”
  ● Often associated with the arrival or sending of information

◆ Start Event
  ● Indicates where a process will start
  ● Optional but recommended
  ● No incoming sequence flows
  ● May be more than one

◆ End Events
  ● Indicates where a path of a process will end
  ● No outgoing sequence flows
  ● Optional but recommended
  ● May be more than one
Activities and Sequence Flows

- An activity is work that is performed within a business process
- A sequence flow shows the order in which activities will be performed in a business process
- It defines the end of one activity and the beginning of the next
- No activity occurs on the sequence flow
- Think of a sequence flow as the thread of activity
- A sequence flow is normally only named if it is the outgoing flow from a gateway
- Use an event or a conditional sequence flow to qualify it
Exclusive Gateways

- Gateways control the flow of the thread of activity
  - 1-N sequence flow in (‘merging’ of flows)
  - 1-N sequence flow out (‘switching’ of flows)
  - Can be named with a question
  - Can occur sequentially
- No work is done in a gateway
- A simple gateway is exclusive by default
- An exclusive gateway has non-overlapping conditions defined for each possible exit
- Think of it as switching the flow
- If no condition is met the thread is held up

Explicitly Exclusive
Intermediate Events

- Indicate where something happens between the start and end of a process
- Indicates the termination of the previous activity or the start of the next
- All events are instantaneous and no work is done
- May occur as the result of receiving or sending data (message flows)
- May be ‘throw’ or ‘catch’
Public Processes

◆ Pools, lanes and message flows

CRaG Systems

BPMN 2.0 Tutorial

BPMN and the BPMN logo are trademarks of the Object Management Group

© Copyright CRaG Systems ’97–’18. All rights reserved
Pools and Lanes

- Pools represent participants in a collaboration
- Pools may be empty (black box), or show a process
- A public process shows external entities as empty pools with messages to and from the pool
- Lanes may be used to organise activities within a pool
- The meaning of the lanes is up to the modeller
- Lanes may be nested e.g. a role within a department
- The assignment of an activity to a pool or lane indicates the allocation of responsibility

Car Dealership
Salesperson
Customer

- Customer enquiry
- Handle Enquiry
- Make Offer
- Offer
- Offer Acc

Mechanic
Accounts Clerk

- Process Payment
- payment accepted

© Copyright CRaG Systems '97–'18. All rights reserved
Message Flows

- are used to show the flow of messages between two participants in a collaboration
- must connect two separate pools and not two objects within the same pool
- connect either to the pool boundary or to flow objects within the pool
- may have an attached message
**Collaboration Process**

- Shows the internal processes of more than one participant (pool) in the collaboration.
- The participants can be different organisations or different departments within the same organisation.
- Message flows can cross the pool boundary to attach to the appropriate activity or event in another pool.
- Start and end events for each pool.
‘Take Order’ and ‘Prepare Vehicle’ are collapsed sub-processes

A collapsed sub-process can be “opened up” to show a lower-level process either as a separate process diagram, or on the parent process diagram

The events starting and ending the parent activities should be consistent with the child diagrams
Concurrency: Parallel Gateway

- Splits thread (token) into parallel paths
- Following activities all start immediately
- Each prior state must be complete at the merging gateway for the thread to continue
- Re-sync the threads with a parallel gateway before leaving the diagram
Process Steps as Tasks

- Tasks are activities that do not decompose
- User task is performed by a human with the assistance of an application
- Service task uses a web-service or automated application
- Business rule task provides input to and gets output from a business rules engine
- Script task executed by a business process engine
Human versus System Pools

- Optional grouping of user/automated tasks into a system pool
- May show data flow between user and system for each step
Choreographies and Conversations

- Added in version 2.0
- Choreography emphasises the messaging between pools in a collaboration process
- Conversations show collections of choreographies
Business Process Execution Language (BPEL)

- Allows the process to be exported as source for Business Process Management system
- Also WS-BPEL for Web Services
- Not all BPMN processes will map to BPEL
- The process must be sound (executable)
  - No deadlocks
  - No lack of synchronisation

```xml
<wSDL:portType name="[if-name]">
  <operation name="[op1-name]">
    <wSDL:input message="[msg1i-name]"/>
    <wSDL:output message="[msg1o-name]"/>
    <wSDL:Fault name="[error1a-faultname]" message="[error1a-name]"/>
    ...
  </operation>
  ...
</wSDL:portType>
```