BPMN 2.0 in the @enterprise Process Editor

November 2014

Groiss Informatics GmbH

Groiss Informatics GmbH

Strutzmannstraße 10/4
9020 Klagenfurt
Austria

Tel: +43 463 504694 - 0
Fax: +43 463 504594 - 10
Email: support@groiss.com

Document Version 8.0.17728

The information in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Groiss Informatics GmbH does not warrant that this document is error-free.

1 Introduction

Historically, @enterprise has been using the proprietary **W**orkflow **D**efinition **N**otation (WDN) along with its textual counterparts WDL and XWDL.

Since @enterprise 8.0 (build 8583) processes can also be modeled using the standard graphical elements of the BPMN 2.0 (**B**usiness **P**rocess **M**odel and **N**otation).

This standard is developed by Object Management Group (OMG), its specification can be found at *http://www.omg.org/spec/BPMN/2.0*. The intent of the BPMN standard - as defined by OMG - is " ... to standardize a business process model and notation in the face of many different modeling notations and viewpoints". So one big advantage of that standard is that workflow participants do not need to learn proprietary notations with different semantics but have one defined standard. BPMN has gained broad acceptance in the industry and @enterprise is committed to convergency between the proprietary WDN and the standard BPMN.

This document will give an overview of the mapping between the modeling elements in WDN and their pendants in BPMN. It will not give an introduction to BPMN 2.0 - for such purposes a a large set of books and other publications is available, e.g. "*BPMN Method & Style*" by Bruce Silver or "*BPMN 2.0 Business Process Model and Notation*" by Thomas Allweyer or "*BPMN 2.0 Praxishandbuch*" (in German) by Jakob Freund and Bernd Rücker.

2 Notation Mapping

To express a certain aspect in BPMN, there are usually several possible modelling alternatives. For example, explicit gateways can be used when splitting the flow in a process. While such gateways increase the readability, their use is not mandatory. Another example are loop constructs which can be modeled using gateways or by employing a BPMN subprocess with a loop decoration.

This flexible modeling space might come handy in early analysis phases of the development, the goals of downstream implementation activities of the process models will usually call for a consensus on the usage of standardized conversion patterns.

The @enterprise process editor is not a mere drawing tool which allows to create single BPMN nodes and to connect them by arbitrary edges. The main paradigm of the @enterprise Process Editor is to support the modeling of processes which can be directly executed in the Business Process Engine of @enterprise. Processes are modeled in a structured way where entire constructs (e.g. loops) consisting of multiple nodes and edges are created in one step.

There is not that much difference between the symbols and constructs used to model a process in either one of the notations. So workflow designers who are used to model processes in WDN should have virtually no difficulty to accomplish the switch to BPMN. But it should be kept in mind, that endusers will also get to see the new graphics via the 'Process' tab in the detail view of work items. So, a considerate and gentle introduction of the new notation may be called for.

The following tables show the mapping between WDN and BPMN constructs. The presentation order of the constructs corresponds to their order in the function list of process editor:

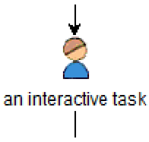
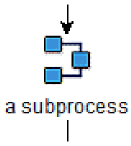
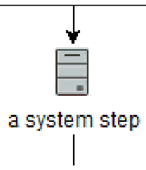
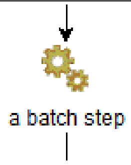
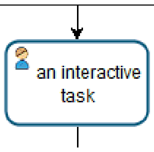
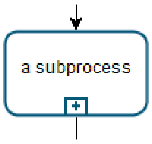
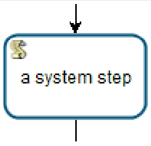
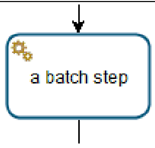
Activities	Task	Subprocess	System	Batch
WDN				
BPMN				

Table 2.1: Mapping for Activities

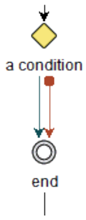
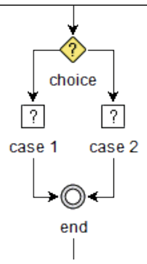
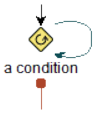
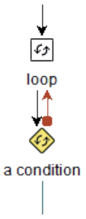
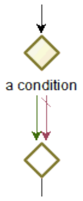
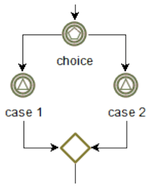
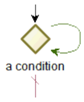
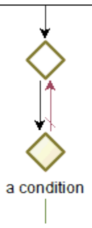
Control Structures	If	Choice	While Loop	Loop
WDN				
BPMN				

Table 2.2: Mapping for Control Structures

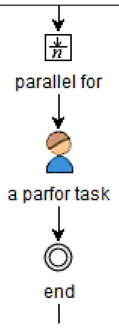
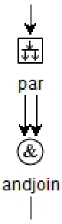
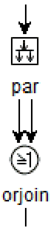
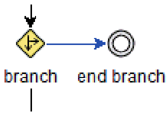
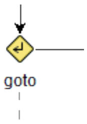
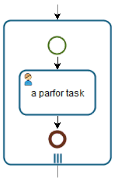
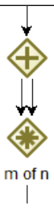
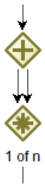
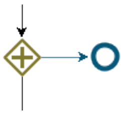
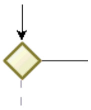
Control Structures	Parallel for	AND-Parallelism	OR-Parallelism	Branch	Goto
WDN					
BPMN					

Table 2.3: Mapping for Control Structures (2)

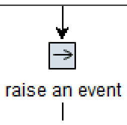
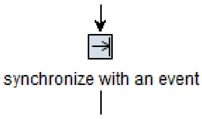
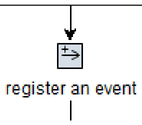
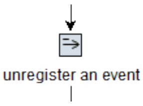
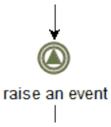
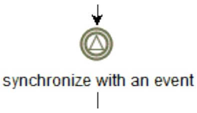
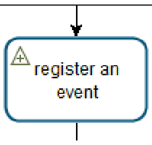
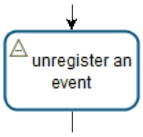
Events	Raise	Synchronize	Register	Unregister
WDN				
BPMN				

Table 2.4: Mapping for Events

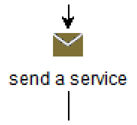
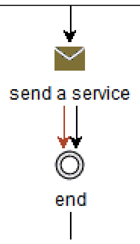
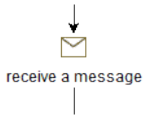
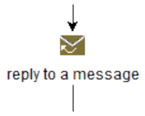
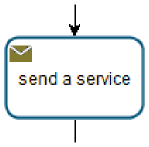
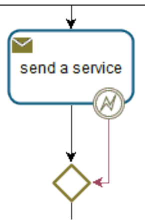
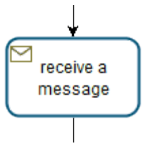
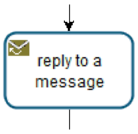
Web-Services	Outgoing Message	Outgoing Message with Exception Flow	Incoming Message	Reply Message
WDN				
BPMN				

Table 2.5: Mapping for Web-Services

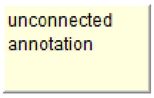
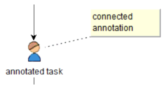
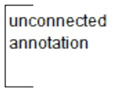
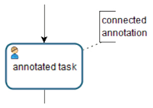
Annotations	Unconnected	Connected
WDN		
BPMN		

Table 2.6: Mapping for Annotations

3 Which notation to use?

The choice of which notation to use can be made for each individual process definition. This can be done via the ’Common’ panel in the properties of a process definition (see ’Administration Handbook’). At this time, the following default notation selection for process definitions is made:
WDN is used:

1. for all existing process definitions - no automatic notation change is performed.
2. for all new process definitions created via the process editor if WDN is set as default in the ’settings’ dialog of the editor. This property can be set per user.
3. for all new process definitions created via loading an XWDL file in which the notation to use is set to WDN or in which it is not specified.
4. for all new process definitions created via XML-Import in which the process definition(s) state WDN as notation or do not specify the notation to be used.

BPMN is used for all process definitions which:

1. are created via the process editor if BPMN is set as default in the ’settings’ dialog of the editor - which is also the system default value for this property.
2. are created via loading a WDL or BPEL file
3. are created via loading a XWDL file in which the notation is set to BPMN
4. are created via XML-Import in which the process definition(s) specify BPMN as the notation to use

So as can be seen, the notation of existing process definitions does not change automatically. After a manual change of a process definitions notation, it is advisable to check the visual consequences for the graphical representation of that process within the process editor. Due to the different shapes and sizes of the various modeling elements the previous layout may deteriorate and call for manual adjustments. The majority of such changes will be necessary because of one the following reasons:

- the text of activities is not fully visible within the default bounds of the activity - manual resizing may be needed.

- elements positioned very close to each other may overlap due to the larger shapes used in BPMN - manual repositioning may be needed.

- a ParFor is now one big element within the editor which may now overlap other elements - again, manual repositioning may be needed.

- in BPMN a GOTO within a ParFor must not point to a node which is not an element of that ParFor. Although this cannot be modeled using the process editor with BPMN notation, this can be the case if the notation of an existing process containing such a GOTO has been changed to BPMN. This situation will lead to ugly effects in the graphical representation of that process so such processes should not use BPMN as notation.