JWT metamodel compared to BPMN metamodel

Philipp Kretschmer

University of Augsburg

Programming distributed Systems Lab

Augsburg, Germany

philipp.rolf.kretschmer@student.uni-augsburg.de

1 Introduction

Speaking genarally a business process is a procedure which has an important role for the economic value added of an organisation. Being more precise a business process is a set of activities which are specific for a company. The activities are target oriented and have a logical and temporal context. A better understanding of processes can be achieved if they are representet by suitable models. A model also contains further characteristics of a process e.g. involved persons, events, notices.

There are several modeling notations available and every single one has its own rules. Examples of other notions are Event-driven Process Chain (EPC), XML Process Definition Language (XPDL), UML Activity Diagram and Business Process Modeling Notation (BPMN).

This paper discusses the mapping between the BPMN specification 1.0 and the metamodel that is underlying the Eclipse Java Workflow Tooling (JWT) project in version 1.3.0. The principles that led the discussion are:

- How does the metamodel of JWT match the BPMN metamodel?
- How can the BPMN metamodel be matched to JWT?

BPMN lets people design and manage business processes with the Business Process Diagram (BPD). The BPD provides the graphical objects and their interrelationship. At the moment there is no extensive description of the metamodel for the BPD available. Therefore i took the metamodel from Korherr and List [KL] in their paper "'Extending the epc and the bpmn with

business goals and performance measures"'. Their extension to core BPMN metamodel is left out of this consideration because it is not our focus.

To avoid ambiguous names in this paper a namespace notation is used like xml documents have it, e.g. JWT:Action or BPMN:Action. This way it is possible to distinguish between two different items which have the same name in both models.

2 Overview

In this section I provide a short JWT and BPMN overview. The intension is to give a summary for both business process modeling languages. After this section you will know the basics about the two different approaches in business process modeling.

2.1 JWT

AgilPro is a joint project with the University of Augsburg, Germany and eMundo GmbH, Unterhaching. It is funded by the high-tech offensive of Bavaria and fosters business process modeling in small and medium-sized enterprises (SMEs). Cause most companies use proprietary software like MS Excel or Word they are not so easy to integrate in a process. The integration in the process is not supported but the data could be integrated with import and export functions. This is where AgilPro comes into play.

AgilPro is an RCP application developed on the Eclipse platform. The model for AgilPro is designed with EMF (Eclipse Modeling Language). It offers different views for different purposes like technical or manager view. The graphical representation is done with Eclipse Graphical Editing Framework (GEF). AgilPro is one conerstone of Eclipse Java Workflow Tooling (JWT) project for Eclipse. JWT consists of two parts, the Workflow Editing (WE) and WAM (Workflow Administration and Monitoring tools). WE is responsible for the workflow modeling part with several different views and a well defined API. WAMs part in the game is the connection to a process engine, the deployment, administration etc.

For details on the JWT metamodel see [BLR] which is appended to this paper.

2.2 BPMN

The Business Process Modeling Notation (BPMN) has as a main goal the intense to be an easy to understand modeling language for all business participants. Starting with the analyst going to the developer, designing a program

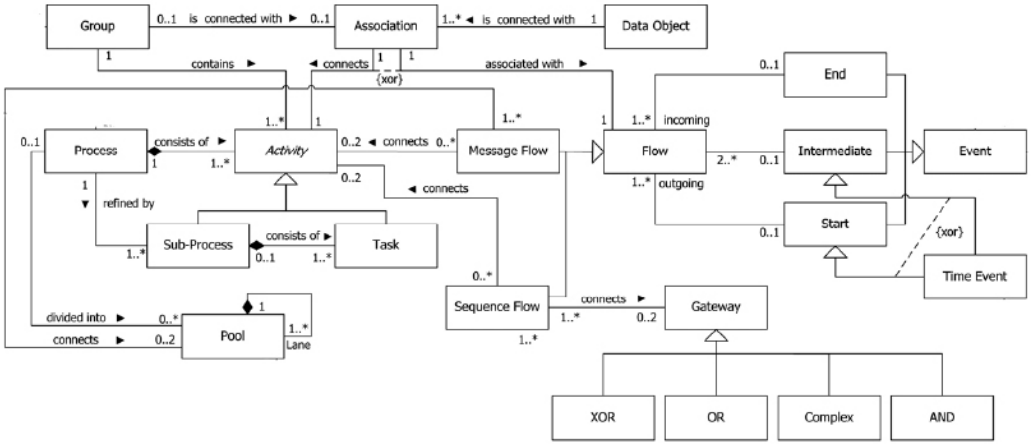


Fig. 1: BPMN Metamodel from [KL]

that executes these processes, to the people managing the processes in every day life, it should be appropriate for these different concerns. BPMN acts as a standardized notation for both, the design and the implementation of business processes, and is therefore able to close the gap in between.

BPMN also aims to offer XML languages like BPEL which stand for the execution of a business process the possibility to get a graphical representation of their processes.

In BPMN there is a Business Process Diagram (BPD) defined with semantic and graphical notion. It respects the different tasks a participant has to fulfill and lets them communicate in a standardized and easy way.

Since 2005 the specification is made by the Business Process Management Initiative (BPMI) and the Object Management Group (OMG). They put together their activities in the Business Modeling & Integration (BMI) Domain Task Force (DTF). They try to bring the best from several other modeling languages together e.g. UML Activity Diagram, ebXML BPSS, Activity-Decision Flow (ADF) Diagram.

Figure 1 shows the metamodel from [KL] which is uses for the following metamodel discussion. I cut out the model extending performance measures and goals because they are not further needed.

3 Model mapping

The first impression comparing the models is that the BPMN model has no attributes but the JWT model has many of them. I think that the attributes are only left out for concerns of complexity. So I tried to get them

where needed from the BPMN specification or a BPMN metamodel from IBM [IBM06], a member of BPMN task force. The BPMN model defines three main event groups: start, intermediate and end. These groups are further specified through triggers or an eventresult like message or error. In JWT Event is a class which can be further specified to taggle specified BPMN events, like a time-out.

BPMN:Process matches JWT:Activity Both classes define the business process itself as they compose everything a business process represents.

BPMN:Task maps to JWT:Action JWT:Action and BPMN:Task are atomic activities in a process model.

BPMN:Sub-Process maps to JWT:StructuredActivityNode In BPMN a process is refined by one or many Sub-Processes like it is in JWT that an Activity may contain StructuredActivityNodes. The intent of both is to further structure the process fine grained parts which logically stick together.

BPMN:Group maps to JWT:Group A Group is a visual arrangement that groups elements of the diagram informally and adds no constraint or other value. The only optional attribute is a name.

BPMN:Pools, BPMN:Lanes maps to JWT:Role, JWT:OrganisationUnit Responsibilities for JWT:Actions / BPMN:Tasks are put to the model in different ways. In JWT there are Roles associated to the Actions which themselves can be grouped in OrganisationUnits. An OrganisationUnit itself may be composed through subUnits, whereas in BPMN there is the concept of Swimlanes which includes Pools and Lanes. Pools represent participants in the process and Lanes divide the Pool into sub partitions, e.g. university is a Pool and one Lane for each professor.

I think the best way to match a Pool, is to match it to an OrganisationUnit cause both represent the container for participants in the process for certain JWT:Actions/BPMN:Activites. The Lane should therefore be matched to Role as both are connected with a higer abstraction, OrganisationUnit and Pool. In JWT it is possible to divide an OrganisationUnit in subUnits but in BPMN Pools cannot contain "'SubPools"'. For this purpose one could match an JWT:OrganisationUnit to a BPMN:Pool and one level deep JWT:subUnits to BPMN:Lanes but then the JWT:Role could not be matched, because Lanes could not be further divided in parts.

We must be careful with the interaction of pools because sequence flow is not allowed to cross the border of a Pool. With Lanes it is no problem to have sequence flow crossing the border. I would say this means that a process has to be modeled within a Pool and the interaction between two processes is modeled with message flows, which are allowed to cross pool borders.

BPMN:Sequence Flow maps to JWT:ActivityEdge A BPMN:SequenceFlow models the transition from an Event or a Gateway to an Activity and vice versa. It is also possible to connect two Activities with a SequenceFlow. A SubProcess can also be interconnected with the elements I just mentioned. In JWT an ActivityEdge connects Actions, ControlNodes and StructuredActivityNodes. Events in JWT are also ExectutableNodes and can therefore be part of a transition.

BPMN:SeqenceFlow Attributes maps to JWT:Guard The main paper I use for this review says nothing about conditions or guards. In the IBM metamodel the attributes from the specification are added to the SequenceConnector to describe the conditions on a SequenceFlow. Namely these are ConditionType, ConditionExpression and Quantity. A BPMN:SequenceFlow can have a condition with one limitation existing, if one defnines a condition on a SequenceFlow there has to be at least one other SequenceFlow leaving. In contrast to the normal arrow symbol the conditional SequenceFlow has a mini diamond at the begining. The default marker has a backslash at the beginning.

The ConditionType indicates whether there is an evaluation at runtime or not. It is a string type and can have the predefined values, None, Expression or Default. If Default is set, then the default marker is shown. Is the type set to Expression then the ConditionExpression will be evaluated at runtime. For None nothing happens. this is the default type. The specification defines an Expression in the following way:

> An Expression MUST be entered to provide a mathematical expression to be either tested as True or False or to be evaluated to update the value of Properties (e.g., assignment).

According to my opinion it would be possible to divide the BPMN ConditionExpression, lets say (a>2 && b<9) in a JWT:GuardSpecification and two subSpecifications. In the GuardSpecification the subSpecificationConnector will get the boolean value AND and the two subSpecifications get the attribute values a and b with the values 2 and 9. Finally the operation attributes must be accordingly set.

In JWT a Guard takes the duty of specifying conditions for an ActivityEdge. The expressions that can be done with an JWT:Guard seem to be more comprehensive than the one possible with the attributes in BPMN.

Looping In BPMN looping is possible in different ways. First there is Activity Looping and second there is Sequence Flow Looping.

Looking at Activity Looping in BPMN I found it difficult to match this concept of the BPMN specification to JWT. The first confusing thing about loops was that Activity Looping is divided in Standard Loops and Multi-Instance Loops. The clear part was that Standard Loops are BPMN:Tasks which are performed more than once. At the first sight I thought Multi-Instance Loops are the same as Seqence Flow Loops only compressed to a BPMN:SubProcess node. On the second view it is obvious that there could be activites and transitions organised in a sub-process not having a link back but BPMN:Tasks heading to an endpoint or sending a message to other processes.

Sequence Flow Looping is having a Sequence Flow connection back to a previous performed BPMN:Task. The endpoint of the connection must be another task than the starting point of the connection uses.

These BPMN concepts could not be found in the paper [KL] but the IBM-Metamodel includes the concept ActivityLoop subclasses of it.

BPMN:Start map to JWT:InitialNode Start in BPMN is a Subclass of Event where as in JWT the InitialNode is a Subclass of ControlNode. Both types model the beginning of a business process. In BPMN it is possible to model the start event more detailed e.g. Timer, Message.

BPMN:End maps to JWT:Final The ends of a process are modeled with these classes. There is no difference between a normal termination and an abortion in JWT at the moment. Like start events there is a more precise description possible in BPMN, e.g. Terminate, Error.

BPMN:XOR maps to JWT:DecisionNode If someone has to make an exclusive choice he should choose these classes to split the process path.

BPMN:XOR maps to JWT:MergeNode If u need to end an exclusive choice, this node reconnects the different pathes of a process.

BPMN:AND maps to JWT:ForkNode The split for parallel pathes is done with these two classes.

BPMN:AND maps to JWT:JoinNode A JoinNode brings parallel pathes back together just like the AND node does.

BPMN:OR / BPMN:Complex The inclusive or has no opposite in the JWT model. It has been left out because it is not decidable at runtime or simulation time. With Complex it is possible to provide complex expressions that determine the merging and/or splitting behavior of the Gateway [BPM04]. Dependend on the decision at the gateway several outgoing Sequence Flows can be choosen. The expression should be designed in a way that at least one way can be taken.

I think that in the rare case a Complex Gateway appears it will be possible to modell it with some effort with a combination of parallel and exclusive choices. An exception must be made where the inclusive or is needed because as described above it is not matched in JWT.

BPMN:DataObject maps to JWT:Data In JWT it is possible to have multiple Data inputs and outputs for an Action. DataObject belongs to the category artifacs like group does. In BPMN DataObjects are connected to Activities. In the BPMN model from [KL] it is only possible to connect one Data Object but I think this is not appropriate. In the model from IBM it is like in JWT that an Activity can have several input and output DataObjects.

BPMN:Event maps to JWT:Event JWT models the Initial- and FinalNode as a subclass of ControlNode but in BPMN these two nodes are Events (Start and End) as mentioned earlier. Apart from this the event classes are the same. In BPMN there are different kinds of events Start, End and Intermediate. All three types have triggers. Examples for triggers are rule, timer or message. When a message arrives an Intermediate Event occurs if the trigger is message. There are little differences for start, intermediate and end events. End events do not have the types timer, rule and cancel.

A Trigger has attributes which can be encapsulated in a Trigger object and then linked with the JWT:Event. The timer Trigger for example has a specific time or cycle.

Intermediate events are events which are neither start nor end events. They appear to happen between the start and end event and can have the same types as a start event.

BPMN:Association maps to JWT:Reference An association connects a BPMN:Activity with object like DataObjects or Groups, which belong to the category of artifacs. The specification of BPMN says, that BPMN shows the

association of data artifacts with activities. This is the same in JWT where a reference connects e.g. Data with the Action through a ReferenceEdge.

JWT:Application / JWT:Parameter - BPMN:DataObject I found nothing equal for Application in BPMN. But the BPMN specification says generally for DataObjects that they provide the information an activity needs for beeing performed and then for what they produce. This description of DataObject led me to the decision that I would map an Application and a Parameter to it cause they exactly fit in this role. The Application and Parameter provide much information for the JWT:Action for beeing performed. A DataObject has not to be only electronic but may also be a physical item.

An DataObject is in the category of artifacts cause it does not affect the sequence or message flow. An BPMN:Assiciation connects it to the Flow Objects.

BPMN:MessageFlow Interaction between a business process and another business process or between a business process and a participant is done with Message Flow. It can be attached to two participants which are prepared to send or recieve a message. Message Flow has to connect objects that are not in the same participant lane boundary. There is no equivalent in JWT.

4 Concrete syntax

The concrete syntax of BPMN is shown in the figures 2 and 3. There you can find the basic elements and the corresponding graphical notation in the way it is defined in the specification.

5 Table overview

Concept	BPMN	JWT
General concepts		
Process	Process	Activity
Process behavior	—	—
Link to another process		
Included Process	Sub-Process	StructuredActivityNode
Group	Group	Group
Activity	Task	Action

continued on next page

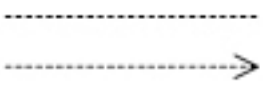
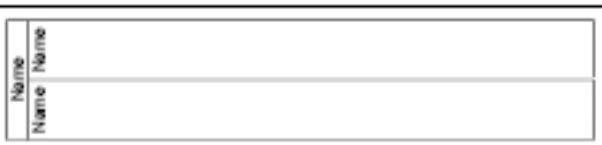
Element	Notation
Event	
Activity	
Gateway	
Sequence Flow	
Message Flow	
Association	
Pool	
Lane	

Fig. 2: concrete syntax BPMN

Element	Notation
Data Object	
Group (a box around a group of objects for documentation purposes)	
Text Annotation (attached with an Association)	

Fig. 3: concrete syntax BPMN

continued from previous page

Concept	BPMN	JWT
Transition	Sequence Flow	ActivityEdge
Guard on Transition	SequenceFlow Attributes[a]	Guard
Loops	—[b]	look detailed description
Control Nodes		
Process start	Start	InitialNode
Process finish	End	FinalNode
Process flow abort	End(Error)/Intermediate(Error)	finalNode
XOR-Split	XOR	DecisionNode
XOR-Join	XOR	MergeNode
AND-Split	AND	ForkNode
AND-Join	AND	JoinNode
OR-Split	OR	
OR-Join	OR	
Complex-Split	Complex	
Complex-Join	Complex	
IOPE		
Input data	Data Object	Data
Output data	Data Object	Data
Precondition		
Effect		
Events		
Event	Event	Event
Message Event	MessageTrigger	
Timer Event	TimerTrigger	
Rule Event	RuleTrigger	
Link Event	LinkTrigger	
Multiple Event		
Compensate Event	CompensationTrigger	
Error Event	ErrorTrigger	
Cancel Event	CancelTrigger	
Intermediate Event	Intermediate	
Business specific		
References	Association	Reference

continued on next page

[a] In the IBM model there is a Guard defined as an attribute of the transition/SequenceConnector

[b] Looped tasks are defined; in the IBM model there is an ActivityLoop

continued from previous page		
Concept	**BPMN**	**JWT**
Business Function		Function
Role	Pool/Lane	Role
Organisation	Pool	OrganisationUnit
Application	DataObject	Application
Parameter	DataObject	Parameter
Interactions		
Interaction		
Message channel	MessageFlow	
Interaction Role		
Flow Binding		

Tab. 1: Comparision of BPMN and JWT

References

[BLR] Bernhard Bauer, Florian Lautenbacher, and Stephan Roser. Agilpro - agile processes in the context of erp – agilpro metamodel description. 'http://wiki.eclipse.org/images/2/2f/AgilPro_MetamodelDescription.pdf.

[BPM04] BPMI. Business process modeling notation (bpmn). 2004.

[Egb07] Steve Egbert. Agilpro metamodel vs. xpdl 1.0 specification. 2007.

[IBM06] IBM. Bpmnmodel uml documentation. 2006.

[KL] Birgit Korherr and Beate List. Extending the epc and the bpmn with business goals and performance measures. 'http://www.wit.at/people/korherr/publications/iceis2007.pdf.

[Lau07] Florian Lautenbacher. Jwt - java workflow tooling – comparision of business process metamodels. 2007.

[Pro] Eclipse JWT Project. Eclipse jwt project. 'http://www.eclipse.org/jwt.

[Whi06] Stephen A. White. Introduction to bpmn. 2006.