Owl axioms

|  |  |
| --- | --- |
| **Specification in natural language** | **Formalized Specification** |
| A **Gateway** MUST have either multiple *incoming* **Sequence Flows** or multiple *outgoing* **Sequence Flows** (i.e., it MUST merge or split the flow). (pp290) |  |
| A **Gateway** with a gatewayDirection of unspecified MAY have both multiple *incoming* and *outgoing* **Sequence Flows**. |  |
| A **Gateway** with a gatewayDirection of mixed MUST have both multiple *incoming* and *outgoing* **Sequence Flows**. |  |
| A **Gateway** with a gatewayDirection of converging MUST have multiple *incoming* **Sequence Flows**, but MUST NOT have multiple *outgoing* **Sequence Flows**. |  |
| A **Gateway** with a gatewayDirection of diverging MUST have multiple *outgoing* **Sequence Flows**, but MUST NOT have multiple *incoming* **Sequence Flows**. |  |
| An **Event Gateway** MUST have two or more *outgoing* **Sequence Flows**. pp297 |  |
| The *outgoing* **Sequence Flows** of the **Event Gateway** MUST NOT have a conditionExpression. |  |

# Start event

**Start Event** indicates where a particular **Process** will start.

* The **Start Event** starts the flow of the **Process**, and thus, will not have any *incoming* **Sequence Flows**
* The **Start Event** should have at least one *outgoing* **Sequence Flow**
* The **Start Event** of an **Event Sub-Process** MUST have a defined *trigger*.
  + The **Start Event** *trigger* (EventDefinition) MUST be from the following types: Message, Error, Escalation, Compensation, Conditional, Signal, and Multiple (see page 260 for more details).

# End event

* The **End Event** should have at least one incoming Sequence Flow
* The **End Event** will not have any outgoing **Sequence Flow**

# Process

* Each **Process** should have at least start and an end event
* For public Processes, no value has the same semantics as if the value were false. The value MAY not be true for public Processes.

# Activity

* The *default* **Sequence Flow** should not have a conditionExpression.
* If a sequenceFlow is attached to an activity with has\_default, then it MUST be attached to only this activity (max 1 activity)
* The value of completionQuantity and the start quuatity MUST NOT be less than 1.

# Gateway

The gateway concept has its attribute “gatewayDirection” which may have two values: “converging” and “diverging”

* Multiple outgoing connections are only allowed for converging Gateways.
* Multiple outgoing connections are only allowed for diverging Gateways.
* ConditionExpression, allowed only for Sequence Flow out of Gateways, MAY be null.
* Default is an attribute of a sourceRef (exclusive or inclusive) Gateway.

# TimerEventDefinition

Timer attributes are mutually exclusive and if any of the other Timer attributes is set, timeDuration MUST NOT be set (if the isExecutable attribute of the Process is set to true).

# SequenceFlow

"The FlowNode that the Sequence Flow is connecting to.

For a Process: Of the types of FlowNode, only Activities, Gateways, and Events can be the target. However, Activities that are Event Sub-Processes are not allowed to be a target.

has\_targetRef exactly 1 (Activity or Gateway or Event)

ConditionExpression, allowed only for **Sequence Flow** out of **Gateways**, MAY be null. b. Default is an attribute of a sourceRef (exclusive or inclusive) **Gateway**. c. Note that messageRef, an attribute of various message **Events**, is optional and not in the sub-class.

A reference to the InputSets defined by the InputOutputSpecification. Every InputOutputSpecification MUST define at least one InputSet.

The *default* **Sequence Flow** should not have a conditionExpression. Any such Expression SHALL be ignored.

# ResourceRole

* resourceRef: Should not be specified when resourceAssignmentExpression is provided.
* resourceAssignmentExpression: Should not be specified when a resourceRef is provided.
* resourceParameterBinding: is only applicable if a resourceRef is specified.
* The Resource that is associated with **Activity**. Should not be specified when resourceAssignmentExpression is provided.

# ExpressionAssignment

These Expressions MUST return Resource entity related data types, like Users or Groups.

# Event Sub-Process

An **Event Sub-Process** MUST NOT have any *incoming* or *outgoing* **Sequence Flows**.

An **Event Sub-Process** is not part of the *normal flow* of its parent **Process**—there are no *incoming* or *outgoing* **Sequence Flows**.

An **Event Sub-Process** MUST have one and only one **Start Event**.

Ad-Hoc Sub-Processes

restrict the use of **BPMN** elements that would normally be used in **Sub-Processes**.

* The list of **BPMN** elements that MUST be used in an **Ad-Hoc Sub-Process: Activity**.
* The list of **BPMN** elements that MAY be used in an **Ad-Hoc Sub-Process: Data Object, Sequence Flow, Association, Data Association, Group, Message Flow** (as a *source* or *target*), **Gateway**, and **Intermediate Event**.
* The list of **BPMN** elements that MUST NOT be used in an **Ad-Hoc Sub-Process: Start Event, End Event, Conversations** (graphically), **Conversation Links** (graphically), and **Choreography Activities**.

(has\_flowElements max 0 StartEvent) and (has\_flowElements max 0 EndEvent)

# Properties

* The Property class is a DataElement element that acts as a container for data associated with flow elements. Property elements MUST be contained within a FlowElement.

# EventBasedGateway

In order for an **Event Gateway** to instantiate a **Process**, it MUST not have any *incoming* **Sequence Flows**.

The eventGatewayType determines the behavior of the **Gateway** when used to instantiate a **Process** (as described above). The attribute can only be set to parallel when the instantiate attribute is set to *true*.

Only two values exclusive or parallel

# Call Activity

**calledElement**: CallableElement [0..1] The element to be called, which will be either a **Process** or a GlobalTask.

|  |  |
| --- | --- |
| Tout resourceParameter doit appartenir à une et une seule ressource | has\_parameterResource exactly 1 resource |
| Tout resourceParameterBinding doit appartenir à un et un seul resourceRole |  |
| Tout resourceAssignmentExpression doit appartenir à un et un seul resourceRole |  |
| Toute valeur de paramètre d’entrée a un et un seul paramètre | has\_dataInputValue\_parameter exactly 1 resourceParameter |
| Toute valeur de paramètre de sortie a un et un seul paramètre | has\_dataOutputValue\_parameter exactly 1 resourceParameter |
|  |  |
|  |  |
|  |  |

* Ajouter la relation has\_parameterResource inverse de resourceParameter
* Ajouter la relation has\_workStation qui lie une Activity à une workStation

