

## **NRS Business Process Standards and Guidelines using BPMN**

<b>Last Updated:</b>	June 21, 2016
<b>Version:</b>	1.0.1
<b>Document:</b>	NRS Business Process Standards and Guidelines

# Table of Contents

<b>Version Control.....</b>	<b>3</b>
<b>1. Introduction.....</b>	<b>4</b>
1.1 Purpose.....	4
1.2 Audience .....	4
1.3 Scope/Exclusions .....	4
1.4 Assumptions.....	4
<b>2. Basic Rules of BPMN.....</b>	<b>5</b>
<b>3. Process Modeling Best Practices.....</b>	<b>6</b>
<b>4. BPMN Process Modeling Standards .....</b>	<b>7</b>
<b>5. Verbs to Use in Process Names .....</b>	<b>11</b>
<b>6. Naming Convention Best Practices .....</b>	<b>13</b>
<b>7. Resources .....</b>	<b>15</b>

## Version Control

Document Version	Revision	Date	Author(s)	Change Reference
0.0.1	Draft	2014-10-09	Susan Lindner	Created to support the EA Modeling Standards
0.0.2	Draft	2014-10-10	Lorelei Solomon	Modified
0.0.3	Draft	2014-10-16	Susan Lindner	Modified
1.0.0	Final	2014-10-17	Susan Lindner	Modified
1.0.1	Final	2016-06-16	Noel Carawan-Hubin	Modified

# 1. Introduction

## 1.1 Purpose

This document identifies the elements and terms to use in creating process models using Business Process Model and Notation release 2.0 standard (BPMN 2.0).

It should be noted that the process models created are not for executable code.

## 1.2 Audience

This document is directed at vendors and ministry staff who will be creating or maintaining Business Process Models.

## 1.3 Scope/Exclusions

The scope of this document covers all process models delivered to or maintained by the Sector.

Where conflicts, if any, are perceived between this document and other standards, the Business Portfolio Manager must be consulted.

## 1.4 Assumptions

It is assumed that the audience has working knowledge of the Sector's SDLC process and standards around the content of those documents.

## 2. Basic Rules of BPMN

### Sequence Flows

- Are used to show the order that **Activities** will be performed in a Process
- They cannot cross **Sub-Process** boundaries
- They cannot cross **Pool** boundaries

### Message Flows

- They cannot connect objects that are within the same **Pool**

### Boundary (Edge-mounted) Events

- Must have at most one outgoing **Sequence Flow**
- Must not have any incoming **Sequence Flow**

### Sub-Process

- A **Start Event** in a **Sub-Process** must be of type None

## 3. Process Modeling Best Practices

### Start and End Events

- Always use **Start** and **End** Events
- Distinguish alternative instantiation of the process as separate **Start Events**
- Distinguish various end states as separate **End Events**
- Flows that end in the same end state should be merged to the same **End Event**

### Gateways

- Always use **Gateways** to depict split or merge of flows
- Always place an **Activity** that will determine the diverging condition(s) just before a diverging **Gateway** of type Exclusive, Inclusive and Complex
  - A benefit of this best practice is that this decision **Activity** can now be interrupted if need be

## 4. BPMN Process Modeling Standards

Business Process Diagrams must use those Elements and Attributes selected from the **Descriptive** and **Analytic** sub-classes for Process Modeling.

Object Management Group’s standards for Business Process and Model Notation (BPMN), Version 2.0, defines three conformance sub-classes for Process Modeling: **Descriptive**, **Analytic**, and **Common Executable**. Both **Descriptive** and **Analytic** focus on visible (graphical) elements and a minimal subset of supporting attributes/elements. **Common Executable** focuses on what is needed for executable process models.

All of the **Descriptive** and a subset of **Analytic** sub-class Elements and Attributes were selected for use. (Refer to Tables 2.1 and 2.2 of OMG’s specifications document for BPMN 2.0)

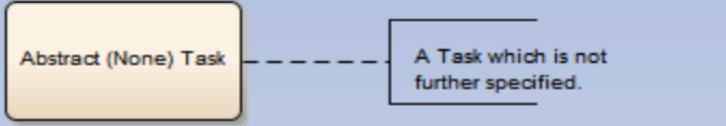
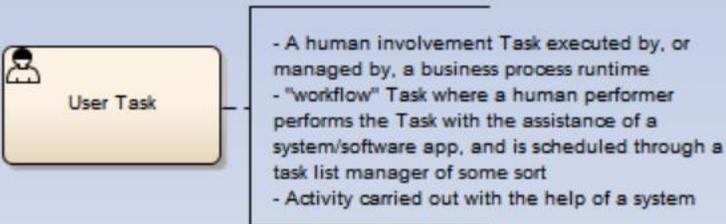
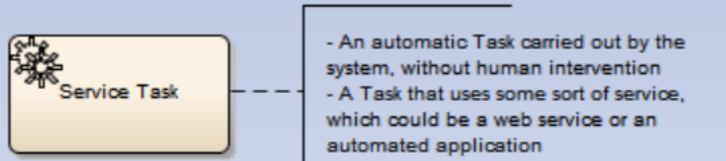
Acceptable Elements and Attributes to use on BPMN 2.0 Business Process Diagrams include the following:

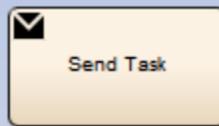
- **Pool**
- **Lane** (Role)
- **Sequence Flow** (*unconditional*) – must not use *default* or *conditional* flows
- **Message Flow** – must not use, attach, Message icon (element)

### Activity Types:

- Task (atomic)
- Collapsed Sub-Process (compound)

### Task Types and Graphical Markers:

<p><b>Abstract (None)</b> Avoid use of this, unless absolutely necessary</p>	
<p><b>User</b></p>	
<p><b>Service</b></p>	

<b>Manual</b>	 <p>Manual Task</p> <ul style="list-style-type: none"> <li>- A human involvement Task that is not managed by any business process engine or aid of any software application</li> <li>- Activity carried out manually</li> </ul>
<b>Send</b>	 <p>Send Task</p> <p>A simple Task that is designed to send a Message to an external Participant (relative to the business process). Once the Message has been sent, the Task is completed.</p>
<b>Receive</b>	 <p>Receive Task</p> <p>A simple Task that is designed to wait for a Message to arrive from an external Participant (relative to the business process).</p>

#### Sub-Process Types:

- 1) **Sub-Process** (aka *embedded* Sub-Process)
  - a) Has unique empty Start Event
  - b) For collapsed Sub-Processes, preferred use of Start and End Events
  - c) For expanded Sub-Processes used to depict parallel path activities, the Sub-Process Activity is unlabeled and there are no Start and End Events
- 2) **Call Activity** (aka *reusable* Sub-Process)

Exception to Sub-Process Rules:

  - a) May also have non-empty Start Events
  - b) Can have multiple pools

#### Collapsed Sub-Process Internal Markers:

- **Ad-Hoc** – limit use of this internal activity marker for those instances where all activities may not be activated, otherwise, use expanded Sub-Process to model this behaviour
- Must not use internal markers: Standard Loop, Multi-Instance Loop, Compensation

#### Gateways:

- Exclusive – For consistency, NO symbol in diamond shape to represent this gateway
- Parallel
- Inclusive
- Event-based

#### EVENTS of Types: Start, Intermediate, End:

##### Start Events:

- None (no trigger)
- Message
- Signal
- Timer

**End Events:**

- None (no trigger)
- Message
- Signal
- Terminate

**Intermediate Events (standalone):**

- Link (catch / throw)
- Message (catch / throw)
- Signal (catch / throw)
- Timer (catch)

**Boundary (Edge-mounted) *Interrupting* Intermediate Events:**

- Message
- Timer
- Signal
- Error
- Must not use *Non-Interrupting*

**Artifacts:**

- Group
- Text Annotation

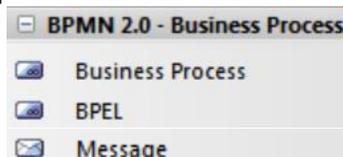
**Data Artifacts:**

- Data Object (None)
- Data Store

**Artifact Connectors:**

- Association
- Data Association

<<DO NOT USE>> Business Process, BPEL (Business Process Execution Language), and Message graphical flow elements:



<<DO NOT USE>> Conversation Link connector:  Conversation Link

<<DO NOT USE>> BPMN 2.0 Type toolbox page:  BPMN 2.0 Types

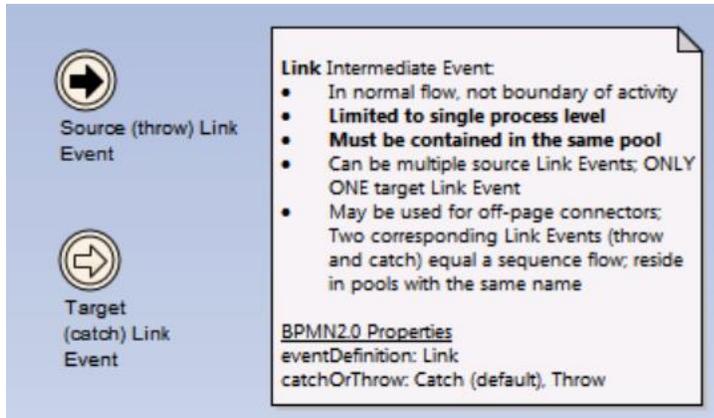
<<DO NOT USE>> Documentation (not a visible element, an attribute of most elements)

## Sending and Receiving Messages

- Use a **Message Event** if the sending or receiving of the message is considered instantaneous
- **Message Task** if the sending or receiving of the message can be interrupted
- From a temporal perspective; an **Event** maps to a time point on a time line and a **Task** maps to a time interval

## Intermediate Events – Catch & Throw Scenarios

### Link Intermediate Event:



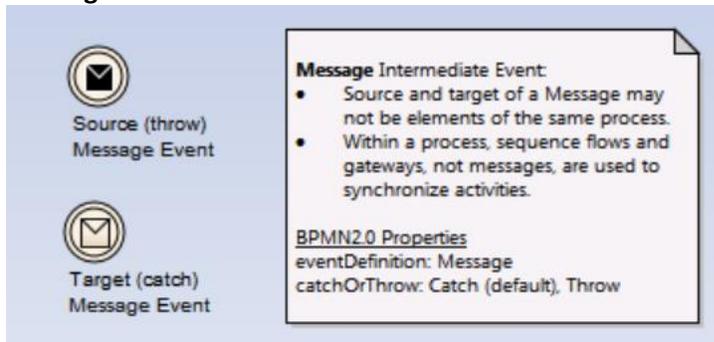
The diagram shows two symbols for Link Intermediate Events: a circle with a right-pointing arrow for 'Source (throw) Link Event' and a circle with a left-pointing arrow for 'Target (catch) Link Event'. A text box on the right provides details:

**Link Intermediate Event:**

- In normal flow, not boundary of activity
- **Limited to single process level**
- **Must be contained in the same pool**
- Can be multiple source Link Events; **ONLY ONE** target Link Event
- May be used for off-page connectors; Two corresponding Link Events (throw and catch) equal a sequence flow; reside in pools with the same name

BPMN2.0 Properties  
eventDefinition: Link  
catchOrThrow: Catch (default), Throw

### Message Intermediate Event:



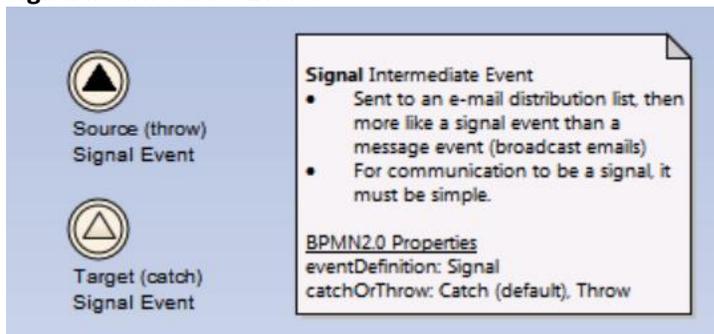
The diagram shows two symbols for Message Intermediate Events: a circle with an envelope icon for 'Source (throw) Message Event' and a circle with an envelope icon for 'Target (catch) Message Event'. A text box on the right provides details:

**Message Intermediate Event:**

- Source and target of a Message may not be elements of the same process.
- Within a process, sequence flows and gateways, not messages, are used to synchronize activities.

BPMN2.0 Properties  
eventDefinition: Message  
catchOrThrow: Catch (default), Throw

### Signal Intermediate Event:



The diagram shows two symbols for Signal Intermediate Events: a circle with a triangle pointing up for 'Source (throw) Signal Event' and a circle with a triangle pointing up for 'Target (catch) Signal Event'. A text box on the right provides details:

**Signal Intermediate Event**

- Sent to an e-mail distribution list, then more like a signal event than a message event (broadcast emails)
- For communication to be a signal, it must be simple.

BPMN2.0 Properties  
eventDefinition: Signal  
catchOrThrow: Catch (default), Throw

## 5. Verbs to Use in Process Names

As a convention, processes are named using verbs and nouns. For example, “Apply for License” or “Identify the Client”.

Verbs generally correspond to actions or events in a business context. A list of common verbs to use in naming processes is provided in table below. The modeler is asked to use those verbs when it makes sense, and to add to the list as needed.

Nouns are used to refer to people, places, events or other things for which business would want to keep records.

Use precise verbs. Verbs such as manage, administer, process, etc. do not convey the precision required for process names. However, they can be used at the function, business domain, level (L1 Business Level).

<b>Accept</b>	<b>Control</b>	<b>Gather</b>	<b>Obtain</b>	<b>Resolve</b>
<b>Adapt</b>	<b>Convert</b>	<b>Generate</b>	<b>Offer</b>	<b>Respond</b>
<b>Adopt</b>	<b>Coordinate</b>	<b>Group</b>	<b>Open</b>	<b>Return</b>
<b>Align</b>	<b>Decide</b>	<b>Guide</b>	<b>Operate</b>	<b>Review</b>
<b>Allocate</b>	<b>Define</b>	<b>Identify</b>	<b>Optimize</b>	<b>Scan</b>
<b>Apply</b>	<b>Deliver</b>	<b>Implement</b>	<b>Order</b>	<b>Schedule</b>
<b>Approve</b>	<b>Deploy</b>	<b>Improve</b>	<b>Organize</b>	<b>Screen</b>
<b>Arrange</b>	<b>Describe</b>	<b>Include</b>	<b>Perform</b>	<b>Search</b>
<b>Assess</b>	<b>Design</b>	<b>Increase</b>	<b>Plan</b>	<b>Secure</b>
<b>Assign</b>	<b>Determine</b>	<b>Initiate</b>	<b>Prepare</b>	<b>Select</b>
<b>Authorize</b>	<b>Develop</b>	<b>Integrate</b>	<b>Present</b>	<b>Serve</b>
<b>Begin</b>	<b>Discuss</b>	<b>Interview</b>	<b>Prioritize</b>	<b>Settle</b>
<b>Build</b>	<b>Distribute</b>	<b>Involve</b>	<b>Process</b>	<b>Share</b>
<b>Calculate</b>	<b>Divide</b>	<b>Issue</b>	<b>Produce</b>	<b>Show</b>
<b>Capture</b>	<b>Enable</b>	<b>Launch</b>	<b>Propose</b>	<b>Simplify</b>
<b>Categorize</b>	<b>Enter</b>	<b>Leverage</b>	<b>Provide</b>	<b>Sort</b>
<b>Change</b>	<b>Establish</b>	<b>License</b>	<b>Purchase</b>	<b>Specify</b>
<b>Choose</b>	<b>Estimate</b>	<b>Link</b>	<b>Qualify</b>	<b>Staff</b>
<b>Circulate</b>	<b>Evaluate</b>	<b>Load</b>	<b>Quantify</b>	<b>Start</b>
<b>Clarify</b>	<b>Examine</b>	<b>Locate</b>	<b>Query</b>	<b>Structure</b>
<b>Classify</b>	<b>Execute</b>	<b>Maintain *</b>	<b>Receive</b>	<b>Submit</b>
<b>Close</b>	<b>Expedite</b>	<b>Manage *</b>	<b>Record</b>	<b>Suggest</b>
<b>Combine</b>	<b>Explore</b>	<b>Maximize</b>	<b>Refer</b>	<b>Supply</b>

<b>Compare</b>	<b>Facilitate</b>	<b>Measure</b>	<b>Register</b>	<b>Support</b>
<b>Compile</b>	<b>Finish</b>	<b>Minimize</b>	<b>Release</b>	<b>Survey</b>
<b>Complete</b>	<b>Focus</b>	<b>Modify</b>	<b>Replace</b>	<b>Test</b>
<b>Conduct</b>	<b>Forecast</b>	<b>Monitor *</b>	<b>Reply</b>	<b>Translate</b>
<b>Confirm</b>	<b>Form</b>	<b>Move</b>	<b>Report</b>	<b>Transmit</b>
<b>Construct</b>	<b>Forward</b>	<b>Name</b>	<b>Request</b>	<b>Update</b>
		<b>Navigate</b>		<b>Validate</b>
<b>Continue</b>	<b>Foster</b>	<b>Notify</b>	<b>Reserve</b>	<b>Verify</b>

\* Avoid use of the verbs **Manage**, **Monitor**, or **Maintain**, as these may suggest an action, however, typically mean things performed *continuously* rather than *repeatedly*.

## 6. Naming Convention Best Practices

### General:

- Always use keywords that are meaningful to the business
- Do not use uncommon abbreviations
- **Do not use the element type in its name (i.e. Activity1)**

### Activities

- All **Activities** should be named (except for expanded Sub-Process)
- Name **Activities** using a Verb-Noun phrase
  - Use the present tense of an active verb of meaning to the business
  - Use a qualified noun of meaning to the business
- Do not name multiple **Activities** with the same name (except for Call Activities)

### Gateways

- **Gateways** do not perform any work or make decisions; it is simply a visualization of divergence or convergence of flow
- **Do not** name converging **Gateways**
  - Associate a **Text Annotation** when convergence logic is not obvious
- Name diverging **Exclusive Gateways** with an interrogative phrase (except where gateway used for directing the flow)

### Sequence Flows

- Name **Sequence flows** coming out of diverging **Gateways** of type Exclusive, Inclusive and Complex using their associated conditions stated as outcomes

### Message Flows

- Label **Message Flows** directly with the name of the message
- The label should be the name of the message, for example, **Rejection notice**, and not the name of the state (i.e. Rejected) or the action of sending or receiving (i.e. Send rejection)

### Pools

- The name of the **Pool** should be the name of the organization, department, branch, external Participant, etc. or the Process Name
- If you use a **Pool** on a child level (sub-process) diagram (Call Activity ONLY), label it with the same name of the upper-level process not the Sub-Process Activity name

### Lanes

- Name **Lanes** using the Category's name
  - **Lanes** are often used to categorize elements by Roles
  - Name Roles using a qualified noun or noun phrase

### Data Objects

- All **Data Objects** should be named
- Name **Data Objects** using a qualified noun that is the name of a business object or information object of meaning to the business

## Events

- All “triggered” **Events** should be named
- Name **Message, Signal, Escalation,** and **Error Events** with a past participle using an active verb
- Name **Link Event** with a noun
- Name paired **Message, Link, Signal, Escalation,** and **Error Events** using a matching name
- Name **Timer Events** using their schedule
- Name **End Events** using the name of the end state

## 7. Resources

- Object Management Group, Inc. (OMG) specifications document for BPMN 2.0: Business Process Model and Notation (BPMN), Version 2.0  
<http://www.omg.org/spec/BPMN/2.0/>
- BPMN Method and Style, Second Edition, with BPMN Implementer's Guide book by Bruce Silver (available as an eBook)