

3. Business Process Diagram Concepts

This section provides a summary of the BPMN graphical objects and their relationships. More details on the concepts will be provided in “Business Process Diagram Graphical Objects” on page 34 and “Connecting Objects” on page 81.

One of the goals of BPMN is that the notation be simple and adoptable by business analysts. Also, there is a conflicting requirement that BPMN provide the power to depict complex business processes and map to BPM execution languages. To help understand how BPMN can manage both requirements, the list of BPMN graphic elements is presented in two groups.

First, there is the list of core elements that will support the requirement of a simple notation. These are the elements that define the basic look-and-feel of BPMN. Most business processes will be modeled adequately with these elements. Second, there is the entire list of elements, including the core elements, which will help support requirement of a powerful notation to handle more advanced modeling situations.




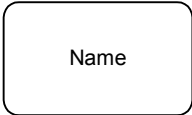
The following are attributes of a Start Event, which extends the set of common object elements (see Table 5):

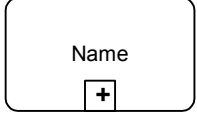
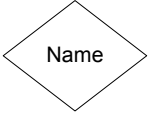
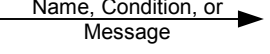
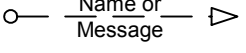


3.1 BPD Core Element Set



It should be emphasized that one of the drivers for the development of BPMN is to create a simple mechanism for creating business process models. Of the core element set, there are three primary modeling elements:

- Events
- Activities
- Gateways

Table 1 displays a list of the core business process concepts that are depicted through the notation:

Element	Description	Notation
Event (three types)	An event is something that “happens” during the course of a business process. These events affect the flow of the process and usually have a cause or an impact. There are three types of events in terms of how they affect the flow: start, intermediate, and end.	<div style="display: flex; flex-direction: column; align-items: center;"> <div style="display: flex; align-items: center; margin-bottom: 10px;"> Start  </div> <div style="display: flex; align-items: center; margin-bottom: 10px;"> Intermediate  </div> <div style="display: flex; align-items: center;"> End  </div> </div>
Task (atomic)	A Task is an atomic activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process Model detail.	

Sub-Process (Compound)	A Sub-Process is a compound activity that is included within a Process. It is compound in that it is broken down into a finer level of detail through a set of sub-activities.	
Decision	Decisions are locations within a business process where the flow of control can take two or more alternative paths.	
Sequence Flow	A Sequence Flow is used to show the order that activities will be performed in a Process.	
Message Flow	A Message Flow is used to show the flow of messages between two entities that are prepared to send and receive them. In BPMN, two separate Pools in the diagram will represent the two entities.	
Pool	A Pool is a “swimlane” and a graphical container for partitioning a set of activities from other Pools, usually in the context of B2B situations.	
Lanes	A Lane is a sub-partition within a Pool and will extend the entire length of the Pool, either vertically or horizontally. Lanes are used to organize and categorize activities within a Pool.	

Element	Description	Notation
Event	An event is something that “happens” during the course of a business process. These events affect the flow of the process and usually have a cause or an impact. Events are circles with open centers for There are three types of events in terms of how they affect the flow: start, intermediate, and end.	
Activity	An activity is a generic term for work that company performs. An activity can be atomic or non-atomic (compound). The types of activities that are a part of a Process Model are: Process, Sub-Process, and Task. Tasks and Sub-Processes are rounded rectangles. Processes are either not bounded or a contained within a Pool.	

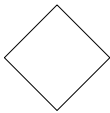

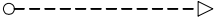

























































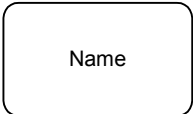

Gateway	A Gateway is used to control the divergence and convergence of multiple Sequence Flow. Thus, it will determine branching, forking, merging, and joining of paths. Icons within the diamond shape will indicate the type of behavior control.	
Sequence Flow	A Sequence Flow is used to show the order that activities will be performed in a Process.	
Message Flow	A Message Flow is used to show the flow of messages between two entities that are prepared to send and receive them. In BPMN, two separate Pools in the diagram will represent the two entities.	
Pool	A Pool is a “swimlane” and a graphical container for partitioning a set of activities from other Pools, usually in the context of B2B situations.	
Lane	A Lane is a sub-partition within a Pool and will extend the entire length of the Pool, either vertically or horizontally. Lanes are used to organize and categorize activities within a Pool.	

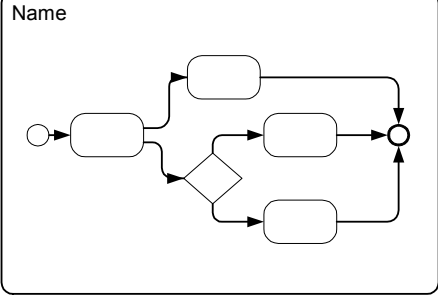
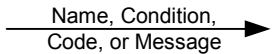
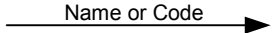
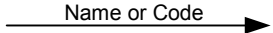
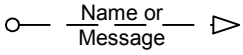
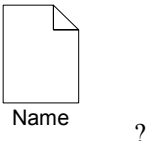
Table 1 BPD Core Element Set

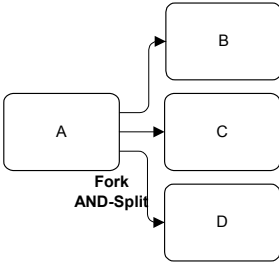
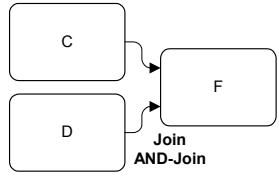
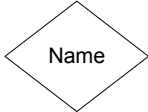
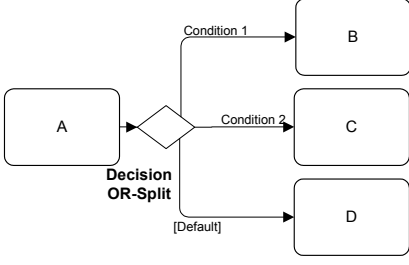
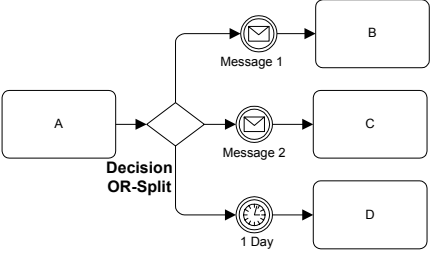
3.2 BPD Complete Set

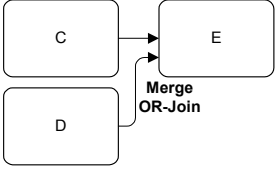
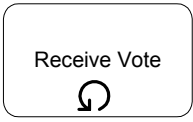
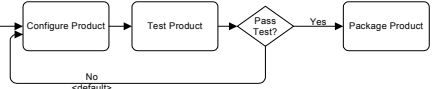

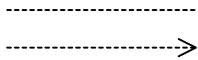
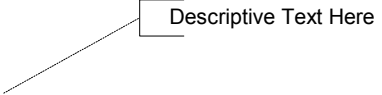
Table 2 displays a more extensive list of the business process concepts that could be depicted through a business process modeling notation.



Element	Description	Notation
Event	An event is something that “happens” during the course of a business process. These events affect the flow of the process and usually have a cause or an impact. There are three types of events in terms of how they affect the flow: start, intermediate, and end.	 Name or Source





<p>Flow Dimension (e.g., Start, Intermediate, End)</p> <p>Start (Message, Timer, Rule, Link, Multiple)</p> <p>Intermediate (Message, Timer, Process Error, Compensate, Rule, Link, Multiple)</p> <p>End (Message, Process Error, Compensate, Link, Multiple)</p>	<p>As the name implies, the Start Event indicates where a particular process will start.</p> <p>Intermediate Events occur between a Start Event and an End Event. This is an event that occurs after a Process has been started. It will affect the flow of the process, but will not start or (directly) terminate the process.</p> <p>As the name implies, the End Event indicates where a process will end.</p>	<p>Start </p> <p>Intermediate </p> <p>End </p>																																
<p>Type Dimension (e.g., Message, Timer, Process Error, Compensate, Rule, Link, Multiple)</p>	<p>Start and Intermediate Events have “Triggers” that define the cause for the event. There are multiple ways that these events can be triggered. End Events may define a “Result” that is a consequence of a Sequence Flow ending.</p>	<table border="1"> <thead> <tr> <th></th> <th>Start</th> <th>Inter- mediate</th> <th>End</th> </tr> </thead> <tbody> <tr> <td>Message</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Timer</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Process Error</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Compensate</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Rule</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Link</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Multiple</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		Start	Inter- mediate	End	Message				Timer				Process Error				Compensate				Rule				Link				Multiple			
	Start	Inter- mediate	End																															
Message																																		
Timer																																		
Process Error																																		
Compensate																																		
Rule																																		
Link																																		
Multiple																																		
<p>Task (Atomic)</p>	<p>A Task is an atomic activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process Model detail.</p>																																	
<p>Process/Sub-Process (non-atomic)</p>	<p>A Sub-Process is a compound activity that is included within a Process. It is compound in that it is broken down into a finer level of detail through a set of sub-activities.</p>	<p>See Next Two Figures</p>																																
<p>Collapsed Sub-Process</p>	<p>The details of the Sub-Process are not visible in the diagram.</p>																																	











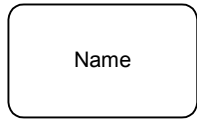
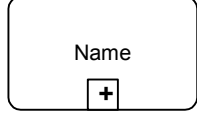
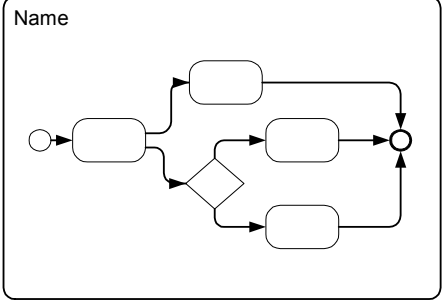
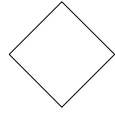
<p>Expanded Sub-Process</p>	<p>The boundary of the Sub-Process is expanded and the details of the Sub-Process are visible within its boundary.</p>	<p>Name</p> 
<p>Sequence Flow</p>	<p>A Sequence Flow is used to show the order that activities will be performed in a Process.</p>	<p>See next three figures</p>
<p>Normal flow</p>	<p>Normal sequence flow refers to the flow that originates from a Start Event and continues through activities via alternative and parallel paths until it ends at an End Event. Conditions (or guards) are only available for Flows exiting a Decision.</p>	
<p>Exception flow</p>	<p>Exception flow occurs outside the normal flow of the Process and is based upon an event (an Intermediate Event) that occurs during the performance of the Process.</p>	
<p>Transaction Compensation flow</p>	<p>Transaction Compensation Flow occurs outside the normal flow of the Process and is based upon an event (an Intermediate Event) that is triggered during the rolling back of a Process that has started, but is later cancelled.</p>	
<p>Message Flow</p>	<p>A Message Flow is used to show the flow of messages between two entities that are prepared to send and receive them. In BPMN, two separate Pools in the diagram will represent the two entities.</p>	
<p>Data Object</p>	<p>Data Objects are considered artifacts because they do not have any direct affect on the Sequence Flow or Message Flow of the Process, but they do provide information about what the Process does.</p>	







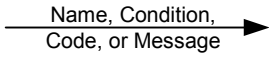
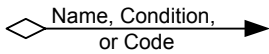
<p>Fork (AND-Split)</p>	<p>BPMN uses the term forking to refer to the dividing of a path into two or more parallel paths (also known as an AND-Split). It is a place in the Process where activities can be performed concurrently, rather than serially.</p>	
<p>Join (AND-Join)</p>	<p>BPMN uses the term joining to refer to the combining of two or more parallel paths into one path (also known as an AND-Join). The Join mechanism is an Open Issue.</p>	
<p>Decision, Branching Point; (OR-Split)</p>	<p>Decisions are locations within a business process where the flow of control can take two or more alternative paths.</p>	
<p>Data-Based Exclusive Condition Default</p>	<p>The set of Decision Alternatives for Data-Based Exclusive Decisions are based on condition expressions. These expressions evaluate the current values of process data to determine which path should be taken. This means that if none of the other condition expressions is true at runtime, then the default expression will be chosen.</p>	
<p>Event-Based Exclusive</p>	<p>This Decision represents a branching point in the process where the Alternatives are based on an Intermediate Event that occurs at that point in the Process. The specific Intermediate Event, usually a message type, determines which of the paths will be taken.</p>	
<p>Inclusive</p>	<p>An Inclusive Decision is a hybrid between a Fork (AND-Split) and a Decision (OR-Split). In some sense it is a grouping of related independent Binary (Yes/No) Decisions. Since each path is independent, all combinations of the paths may be taken, from one to all.</p>	<p>Notation TDB</p>

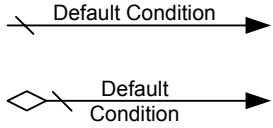
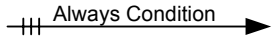
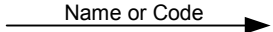
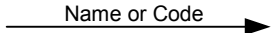
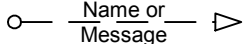
<p>Merging (OR-Join)</p>	<p>BPMN uses the term merging to refer to the combining of two or more alternative paths into one path (also known as an a OR-Join). The Merge mechanism is an Open Issue.</p>	
<p>Looping; Multiple Instances</p>	<p>BPMN provides 2 (two) mechanisms for looping within a Process.</p>	<p>See Next Two Figures</p>
<p>Activity Looping</p>	<p>The properties of Tasks and Sub-Processes will determine if they are repeated or performed once. There are two types of loops: Standard and ForEach.</p>	
<p>Sequence Flow Looping</p>	<p>Loops can be created by connecting a Sequence Flow to an “upstream” object. An object is considered to be upstream if that object has an outgoing Sequence Flow that leads to a series of other Sequence Flows, the last of which is an incoming Sequence Flow to the original object.</p>	
<p>Process Break (something out of the control of the process makes the process pause)</p>	<p>A Process Break is a graphical marker that shows where an expected delay will occur within a Process.</p>	 <p>Other Notation TBD</p>
<p>Transaction group/context</p>		<p>Notation TBD</p>
<p>Nested Process (Inline Block)</p>		<p>Notation TBD, if at all</p>
<p>Group (a box around a group of objects for documentation purposes)</p>	<p>A grouping of activities that does not affect the Sequence Flow. The grouping is generally for documentation or analysis purposes.</p>	<p>Notation TBD, if at all</p>
<p>Off-Page Connector (used within a page?)</p>	<p>Generally used for printing, this object will show where the Sequence Flow leaves one page and then restarts on the next page.</p>	<p>Notation TBD, if at all</p>
<p>Association</p>	<p>An Association is used to associate information with flow objects. Text and graphical non-flow objects can be associated with the flow objects.</p>	
<p>Text Annotation (attached with an Association)</p>	<p>Text Annotations are a mechanism for a modeler to provide additional information for the reader of a BPMN diagram.</p>	

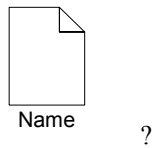
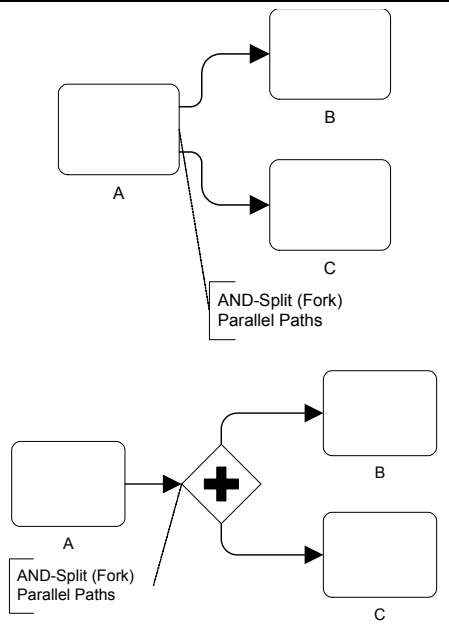
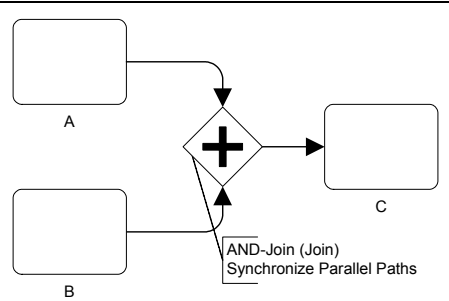
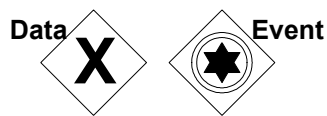
Pool	A Pool is a “swimlane” and a graphical container for partitioning a set of activities from other Pools, usually in the context of B2B situations.	
Lanes	A Lane is a sub-partition within a Pool and will extend the entire length of the Pool, either vertically or horizontally. Lanes are used to organize and categorize activities within a Pool.	

Element	Description	Notation
Event	An event is something that “happens” during the course of a business process. These events affect the flow of the process and usually have a cause or an impact. There are three types of events in terms of how they affect the flow: start, intermediate, and end.	
<p>Flow Dimension (e.g., Start, Intermediate, End)</p> <p>Start (Message, Timer, Rule, Link, Multiple)</p> <p>Intermediate (Message, Timer, Exception, Compensation, Rule, Link, Multiple, Branching)</p> <p>End (Message, Exception, Compensation, Link, Terminate, Multiple)</p>	<p>As the name implies, the Start Event indicates where a particular process will start.</p> <p>Intermediate Events occur between a Start Event and an End Event. This is an event that occurs after a Process has been started. It will affect the flow of the process, but will not start or (directly) terminate the process.</p> <p>As the name implies, the End Event indicates where a process will end.</p>	<p>Start </p> <p>Intermediate </p> <p>End </p>

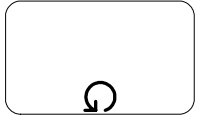
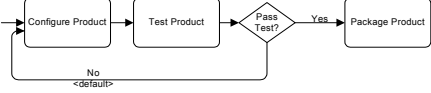
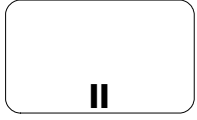

<p>Type Dimension (e.g., Message, Timer, Exception, Compensation, Rule, Link, Terminate, Return, Multiple, Branching)</p>	<p>Start and Intermediate Events have “Triggers” that define the cause for the event. There are multiple ways that these events can be triggered. End Events may define a “Result” that is a consequence of a Sequence Flow ending.</p>	<p>Message </p> <p>Timer </p> <p>Exception </p> <p>Compensation </p> <p>Rule </p> <p>Link </p> <p>Terminate </p> <p>Return </p> <p>Multiple </p> <p>Branching </p>
<p>Task (Atomic)</p>	<p>A Task is an atomic activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process Model detail.</p>	
<p>Process/Sub-Process (non-atomic)</p>	<p>A Sub-Process is a compound activity that is included within a Process. It is compound in that it can be broken down into a finer level of detail through a set of sub-activities.</p>	<p>See Next Two Figures</p>
<p>Collapsed Sub-Process</p>	<p>The details of the Sub-Process are not visible in the diagram.</p>	
<p>Expanded Sub-Process</p>	<p>The boundary of the Sub-Process is expanded and the details of the Sub-Process are visible within its boundary. Note that Sequence Flow cannot cross the boundary of a Sub-Process.</p>	
<p>Gateway</p>	<p>A Gateway is used to control the divergence and convergence of multiple Sequence Flow. Thus, it will determine branching, forking, merging, and joining of paths.</p>	

<p>Gateway Flow Control Types</p>	<p>Icons within the diamond shape will indicate the type of flow control behavior. The types of control include:</p> <ul style="list-style-type: none"> • XOR -- exclusive decision and merging • OR -- inclusive decision • Complex -- complex conditions and situations (e.g., 3 out of 5) • AND -- forking and joining <p>Each type of control affects both the incoming and outgoing Sequence Flow.</p>	<p>Exclusive (XOR) Data-Based  or </p> <p>Event-Based </p> <p>Inclusive (OR) </p> <p>Complex </p> <p>Parallel (AND) </p>
<p>Sequence Flow</p>	<p>A Sequence Flow is used to show the order that activities will be performed in a Process.</p>	<p>See next three figures</p>
<p>Normal flow</p>	<p>Normal Sequence Flow refers to the flow that originates from a Start Event and continues through activities via alternative and parallel paths until it ends at an End Event.</p>	<p></p>
<p>Uncontrolled flow</p>	<p>Uncontrolled flow refers to flow that is not affected by any conditions or does not pass through a Gateway. The simplest example of this is a single Sequence Flow connecting two activities. This can also apply to multiple Sequence Flow that converge on or diverge from an activity. For each uncontrolled Sequence Flow a "Token" will flow from the source object to the target object.</p>	<p>N/A</p>
<p>Conditional flow</p>	<p>Sequence Flow can have condition expressions that are evaluated at runtime to determine whether or not the flow will be used. If the conditional flow is outgoing from an activity, then the Sequence Flow will have a mini-diamond at the beginning of the line (see figure to the right). If the conditional flow is outgoing from a Gateway, then the line will not have a mini-diamond (see figure in the row above).</p>	<p></p>

<p>Default flow</p>	<p>For Data-Based Exclusive Decisions and Inclusive Decisions, one type of flow is the Default condition flow. This flow will be used only if all the other outgoing conditional flow is not true at runtime. If the Default flow is outgoing from a Gateway, then a diagonal slash will be added to the beginning of the line (see the top-right figure). If the Default flow is outgoing from an activity, then a diagonal slash will be added to the beginning of the line, in addition to the mini-diamond (see the bottom-right figure).</p>	
<p>Always flow</p>	<p>For Inclusive Decisions, another type of flow is the Always condition flow. This means that the Sequence Flow will always be used, regardless of the evaluation of the conditions on any other Sequence Flow. If the Always flow is outgoing from an OR Gateway, then the beginning of the line will have three vertical hash marks (see figure to the right). If the Always flow is outgoing from an activity, then the line will not have the three vertical hash marks (see the figure for Normal flow).</p>	
<p>Exception flow</p>	<p>Exception flow occurs outside the normal flow of the Process and is based upon an event (an Intermediate Event) that occurs during the performance of the Process.</p>	
<p>Compensation flow</p>	<p>Compensation Flow occurs outside the normal flow of the Process and is based upon an event (an Intermediate Event) that is triggered during the rolling back of a Process that has started, but is later cancelled.</p>	
<p>Message Flow</p>	<p>A Message Flow is used to show the flow of messages between two entities that are prepared to send and receive them. In BPMN, two separate Pools in the diagram will represent the two entities.</p>	

<p>Data Object</p>	<p>Data Objects are considered artifacts because they do not have any direct affect on the Sequence Flow or Message Flow of the Process, but they do provide information about what the Process does.</p>	
<p>Fork (AND-Split)</p>	<p>BPMN uses the term “fork” to refer to the dividing of a path into two or more parallel paths (also known as an AND-Split). It is a place in the Process where activities can be performed concurrently, rather than serially.</p> <p>There are two options:</p> <p>Multiple Outgoing Sequence Flow can be used (see figure top-right). This represents “uncontrolled” flow is the preferred method for most situations.</p> <p>A forking (AND) Gateway can be used (see figure bottom-right). This will be used rarely, usually in combination with other Gateways.</p>	
<p>Join (AND-Join)</p>	<p>BPMN uses the term “join” to refer to the combining of two or more parallel paths into one path (also known as an AND-Join or synchronization).</p> <p>A joining (AND) Gateway is used to show the joining of multiple flows.</p>	
<p>Decision, Branching Point; (OR-Split)</p>	<p>Decisions are Gateways within a business process where the flow of control can take two or more alternative paths.</p>	<p>See next five rows.</p>
<p>Exclusive</p>	<p>An Exclusive Gateway (XOR) restricts the flow such that <u>only one</u> of a set of alternatives may be chosen during runtime. There are two types of Exclusive Gateways: Data-based and Event-based.</p>	

<p>Data-Based</p>	<p>This Decision represents a branching point where Alternatives are based on conditional expressions contained within the outgoing Sequence Flow. Only one of the Alternatives will be chosen.</p>	<p>OR-Split (Split) Data-Based Exclusive Alternative Paths</p>
<p>Event-Based</p>	<p>This Decision represents a branching point where Alternatives are based on an Event that occurs at that point in the Process. The specific Event, usually a message type, determines which of the paths will be taken. Only one of the Alternatives will be chosen.</p>	<p>OR-Split (Split) Event-Based Exclusive Alternative Paths</p>
<p>Inclusive</p>	<p>This Decision represents a branching point where Alternatives are based on conditional expressions contained within the outgoing Sequence Flow. In some sense it is a grouping of related independent Binary (Yes/No) Decisions. Since each path is independent, all combinations of the paths may be taken, from zero to all. However, it should be designed so that at least one path is taken.</p> <p>There are two versions of this type of Decision. The first uses a collection of conditional Sequence Flow, marked with mini-diamonds (see top-right figure). The second uses an OR Gateway, usually in combination with other Gateways (see bottom-right picture).</p>	<p>OR-Split (Split) Data-Based Inclusive Parallel/Alternative Paths</p> <p>OR-Split (Split) Data-Based Inclusive Parallel/Alternative Paths</p>
<p>Merging (OR-Join)</p>	<p>BPMN uses the term “merge” to refer to the exclusive combining of two or more paths into one path (also known as an a OR-Join).</p> <p>A Merging (XOR) Gateway is used to show the merging of multiple flows.</p> <p>If all the incoming flow is alternative, then a Gateway is not needed. That is, uncontrolled flow provides the same behavior.</p>	

<p>Looping</p>	<p>BPMN provides 2 (two) mechanisms for looping within a Process.</p>	<p>See Next Two Figures</p>
<p>Activity Looping</p>	<p>The properties of Tasks and Sub-Processes will determine if they are repeated or performed once. There are two types of loops: Standard and ForEach. A small looping indicator will be displayed at the bottom-center of the activity.</p>	
<p>Sequence Flow Looping</p>	<p>Loops can be created by connecting a Sequence Flow to an “upstream” object. An object is considered to be upstream if that object has an outgoing Sequence Flow that leads to a series of other Sequence Flows, the last of which is an incoming Sequence Flow to the original object.</p>	
<p>Multiple Instances</p>	<p>The attributes of Tasks and Sub-Processes will determine if they are repeated or performed once. A small parallel indicator will be displayed at the bottom-center of the activity.</p>	
<p>Process Break (something out of the control of the process makes the process pause)</p>	<p>A Process Break is a graphical marker that shows where an expected delay will occur within a Process.</p>	 <p>Other Notation TBD</p>
<p>Transaction</p>	<p>A transaction is an activity that is supported by special protocol that insures that all parties involved have complete agreement that the activity should be completed or cancelled. The attributes of the activity will determine if the activity is a transaction.</p>	<p>There is no special indicator for a Transaction</p>
<p>Nested Sub-Process (Inline Block)</p>	<p>A nested Sub-Process is an activity that shares the same set of data as its parent process. This is opposed to a Sub-Process that is independent, reusable, and referenced from the parent process. Data needs to be passed to the referenced Sub-Process, but not to the nested Sub-Process.</p>	<p>There is no special indicator for nested Sub-Processes</p>

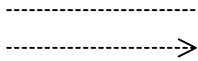
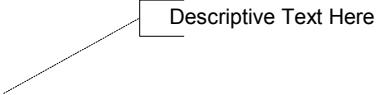


Group (a box around a group of objects for documentation purposes)	A grouping of activities that does not affect the Sequence Flow. The grouping is generally for documentation or analysis purposes.	There is no special graphical indicator for a Group.
Off-Page Connector (used within a page?)	Generally used for printing, this object will show where the Sequence Flow leaves one page and then restarts on the next page.	There is no graphical element for Off-Page Connector.
Association	An Association is used to associate information with flow objects. Text and graphical non-flow objects can be associated with the flow objects.	
Text Annotation (attached with an Association)	Text Annotations are a mechanism for a modeler to provide additional information for the reader of a BPMN diagram.	
Pool	A Pool is a “swimlane” and a graphical container for partitioning a set of activities from other Pools, usually in the context of B2B situations.	
Lanes	A Lane is a sub-partition within a Pool and will extend the entire length of the Pool, either vertically or horizontally. Lanes are used to organize and categorize activities within a Pool.	

Table 2 BPD Complete Element Set

3.3 Use of Text, Color, and Lines in a Diagram

Flow objects and Flows can have labels (e.g., its name) placed inside the shape, or above or below the shape, in any direction or location, depending on the preference of the modeler or modeling tool vendor. Text Annotation objects can be used by the modeler to display additional information about a Process or properties of the objects within the Process.

3.4 Flow Object Connection Rules

An incoming Sequence Flow can connect to any location on a flow object (left, right, top, or bottom). Likewise, an outgoing Sequence Flow can connect from any location on a flow object (left, right, top, or bottom). Message Flows also have this capability. BPMN allows this flexibility, however, we also recommend that modelers use judgment in how flow objects should be connected so that readers of the diagrams will find the behavior clear and easy to follow. This is even more important when a diagram contains Sequence Flows and Message Flows. In these situations it is best to pick a direction of Sequence Flow, either left to right or top to bottom, and then direct the Message Flow at a 90° angle to the Sequence Flow. The resulting diagrams will be much easier to understand.

3.4.1 Sequence Flow Rules

Table 3 displays the BPMN flow objects and shows how these objects can connect to one another through Sequence Flows. The ↗ symbol indicates that the object listed in the row can connect to the object listed in the column. The quantity of connections into an object is specified in the column header with a code letter that precedes the graphical shape. The quantity of connections and out of an object is subject to various configuration dependencies are not specified here. Refer to the sections in the row header with a code letter that follows next chapter for each individual object for more detailed information on the graphical shape. The code letters are: 0 (No Connections); 1 (One Connection); M (Multiple Connections); and M(E) (Multiple Exclusive Connections) appropriate connection rules. Note that if a sub-process has been expanded within a diagram, the objects within the sub-process cannot be connected to objects outside of the sub-process. Nor can Sequence Flows cross a Pool boundary.




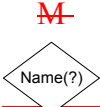
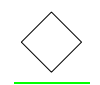









From\To	\emptyset 	M 	M 	M  	1 	M 
 M		↗	↗	↗	↗	↗
 M		↗	↗	↗	↗	↗
 M		↗	↗	↗	↗	↗
  M(E)		↗	↗	↗	↗	↗
 M		↗	↗	↗	↗	↗
 \emptyset						

Table 3 Sequence Flow Connection Rules

Note: Only those objects that can have incoming and/or outgoing Sequence Flow are shown in the table. Thus, Pool, Lane, Data Object, and Text Annotation are not listed in the table.

3.4.2 Message Flow Rules

3.4.2 Message Flow Rules

Table 4 displays the BPMN modeling objects and shows how these objects can connect to one another through Message Flows. The ↗ symbol indicates that the object listed in the row can connect to the object listed in the column. The quantity of connections into an object is specified in the column header with a code letter that precedes the graphical shape. The quantity of connections and out of an object is subject to various configuration dependencies are not specified here. Refer to the sections in the row header with a code letter that follows next chapter for each individual object for more detailed information on the graphical shape. The code letters are: 0 (No Connections); 1 (One Connection in a single direction); M (Multiple Connections in a single direction) appropriate connection rules. Note that Message Flows cannot connect to objects that are within the same Participant Lane boundary.




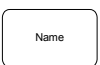








From\To	M 	M 	M 	0-1 	1 	0 
 0						
 M	↗	↗	↗	↗	↗	
 M	↗	↗	↗	↗	↗	
 0-1	↗	↗	↗	↗	↗	
 0						
 M	↗	↗	↗	↗	↗	

Table 4 Message Flow Connection Rules

Note: Only those objects that can have incoming and/or outgoing Message Flow are shown in the table. Thus, Lane, Decision, Data Object, and Text Annotation are not listed in the table.