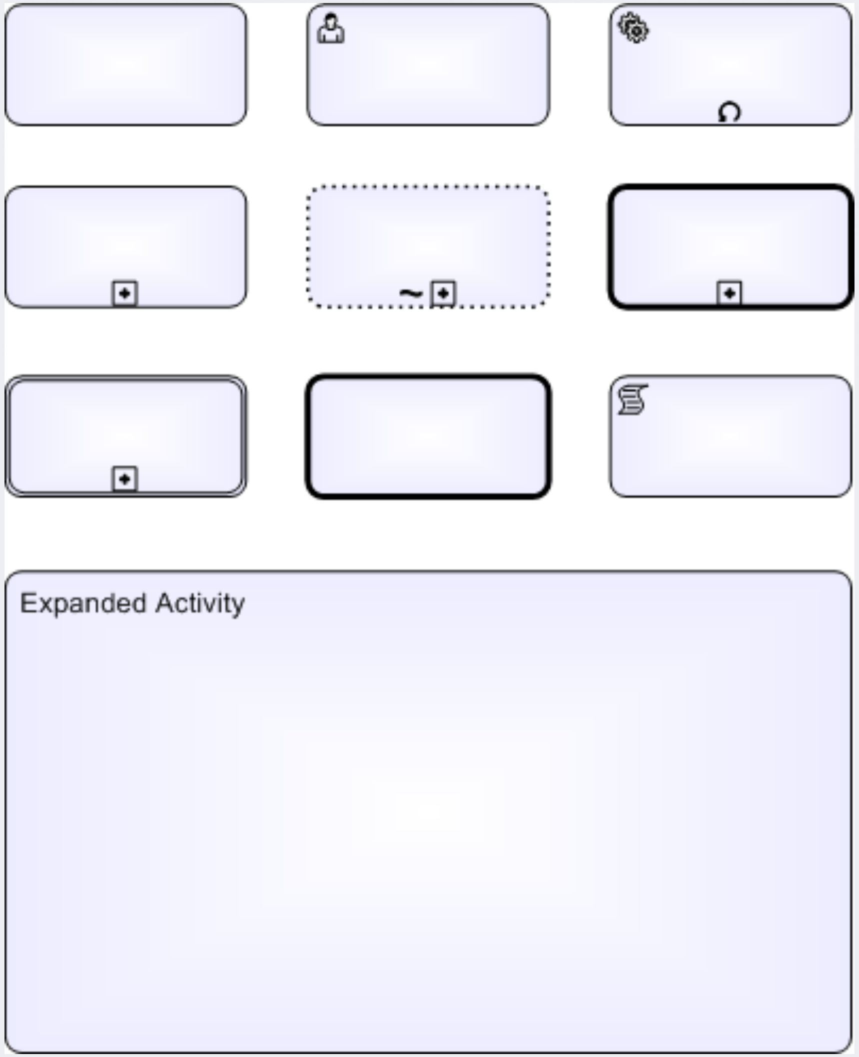
Activities

- **Activity** is a generic term for work that a company performs in a Process. An Activity can be **atomic** or **non-atomic**.

  The type of activities that are part of the process are: **Task** and **Sub-Process**.

  A task can be differentiated by markers that represent its type or associated resource.

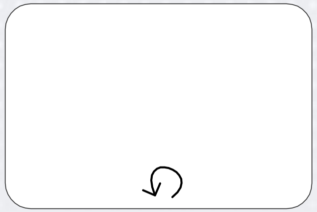
  Sub-Process can be Collapsed or Expanded, and can be differentiated by the kind of elements that join in: **Sub-process**, **Transactions**, **Event Sub Process** and **Call Activities**.

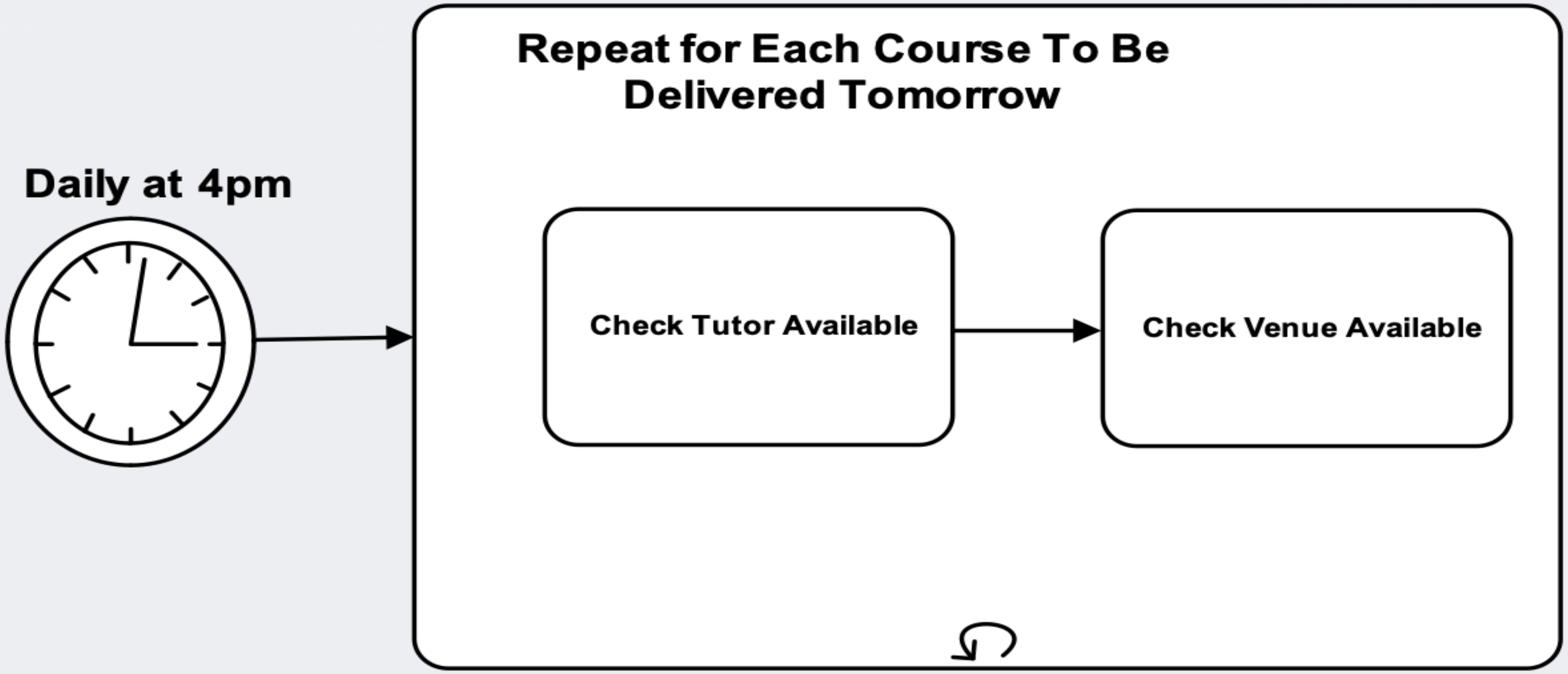


HiCure
Excellence in Health Informatics Integrated Curricula

Iteration

- Loop – process is performed zero, one or many times (serially one after the other)



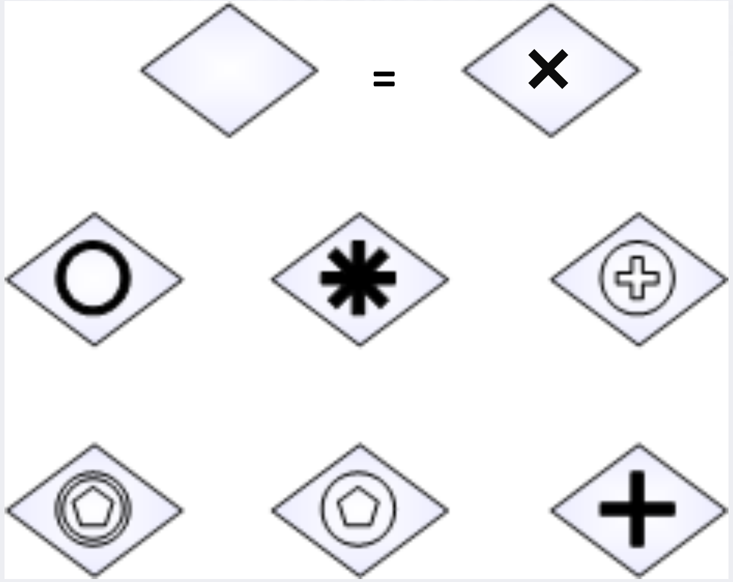


Gateways

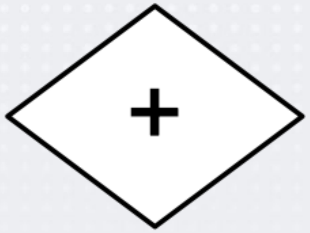
A **Gateway** is used to control the divergence and convergence of sequence flows in a Process or in a choreography.

Gateway will determine branching, forking, merging or joining.

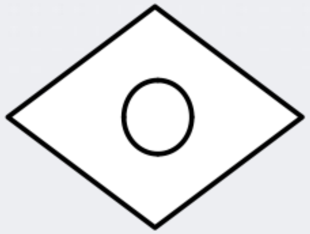
There are 7 kinds of gateways differed by its internal marker: **Exclusive**, **Inclusive**, **Parallel**, **Complex**, **Event-based**, **Parallel Event-based** and **Exclusive Event-based**.



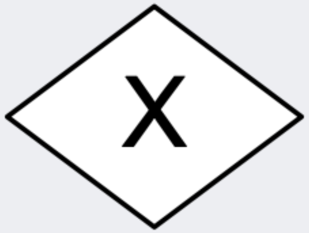
Gateways Types



And - Parallel fork/join

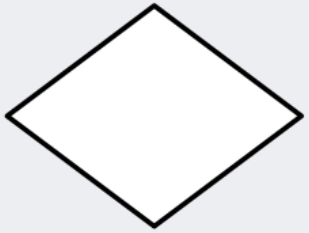


OR - Inclusive Decision/ Merge



XOR - Exclusive Decision/Merge

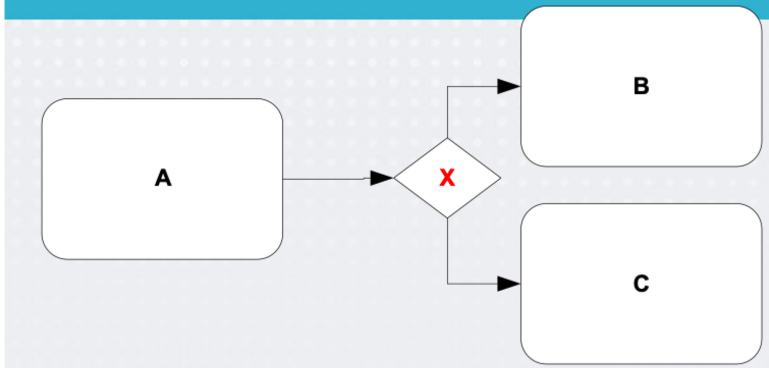
means the same as



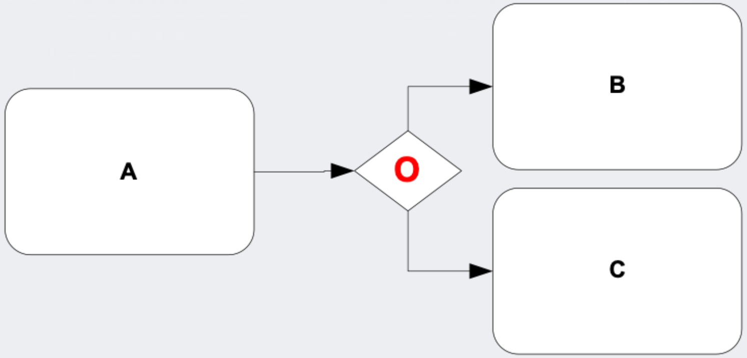
-Gateways control convergence OR divergence of process flows - not equivalent to decision diamonds.

-Decisions are based on the evaluation of a piece of data ***within the preceding process***.

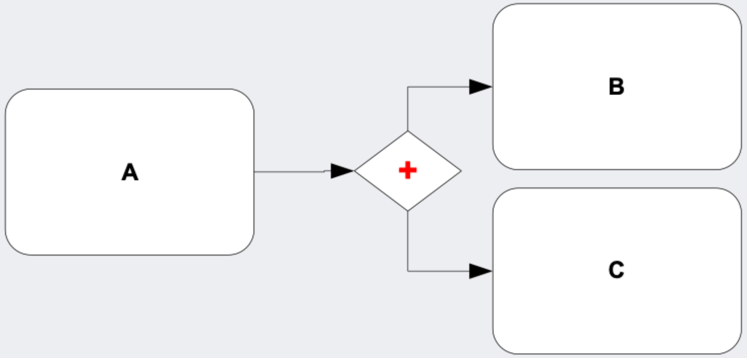
Divergence



Exclusive Or: When A has finished perform B or C



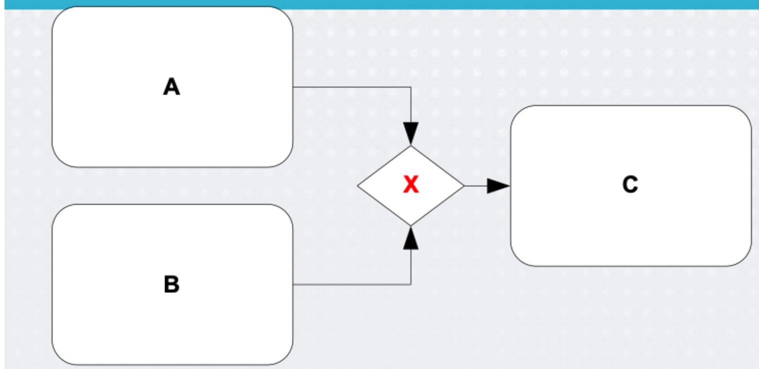
Inclusive Or: When A has finished perform B or C or both



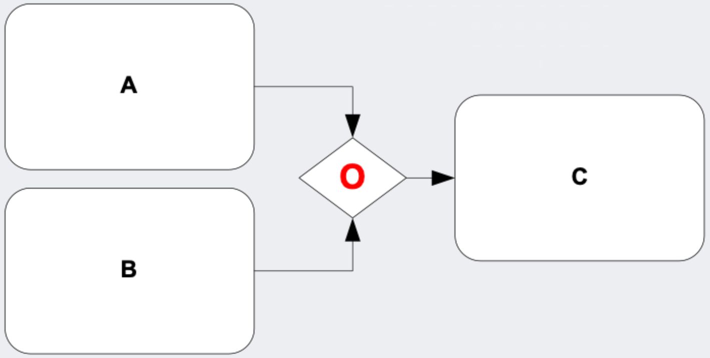
And: When A has finished perform B and C

Excellence in Health Informatics Integrated Curricula

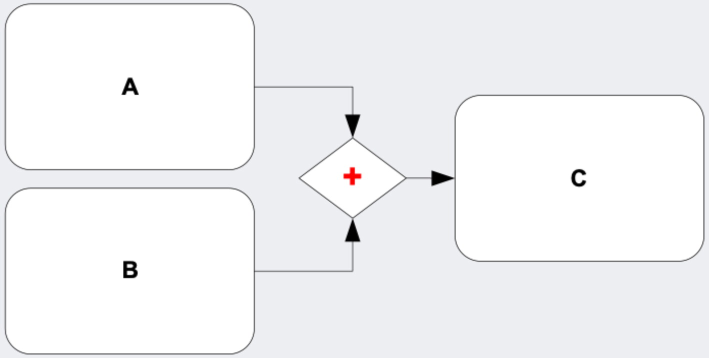
Convergence



Exclusive Or: Perform C when A or B have finished

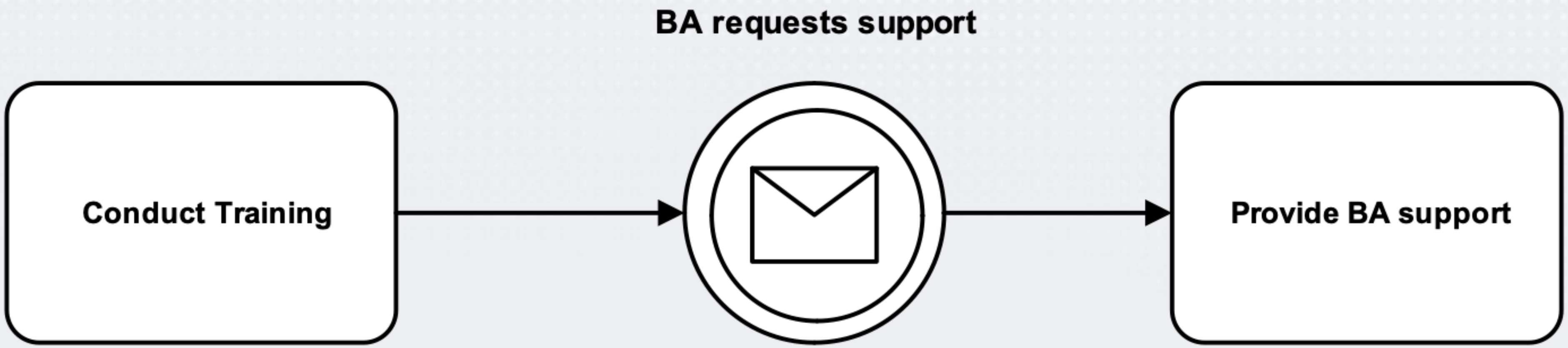


Inclusive Or: Perform C when A or B or Both have finished



And: Perform C when A and B have finished

Process Break

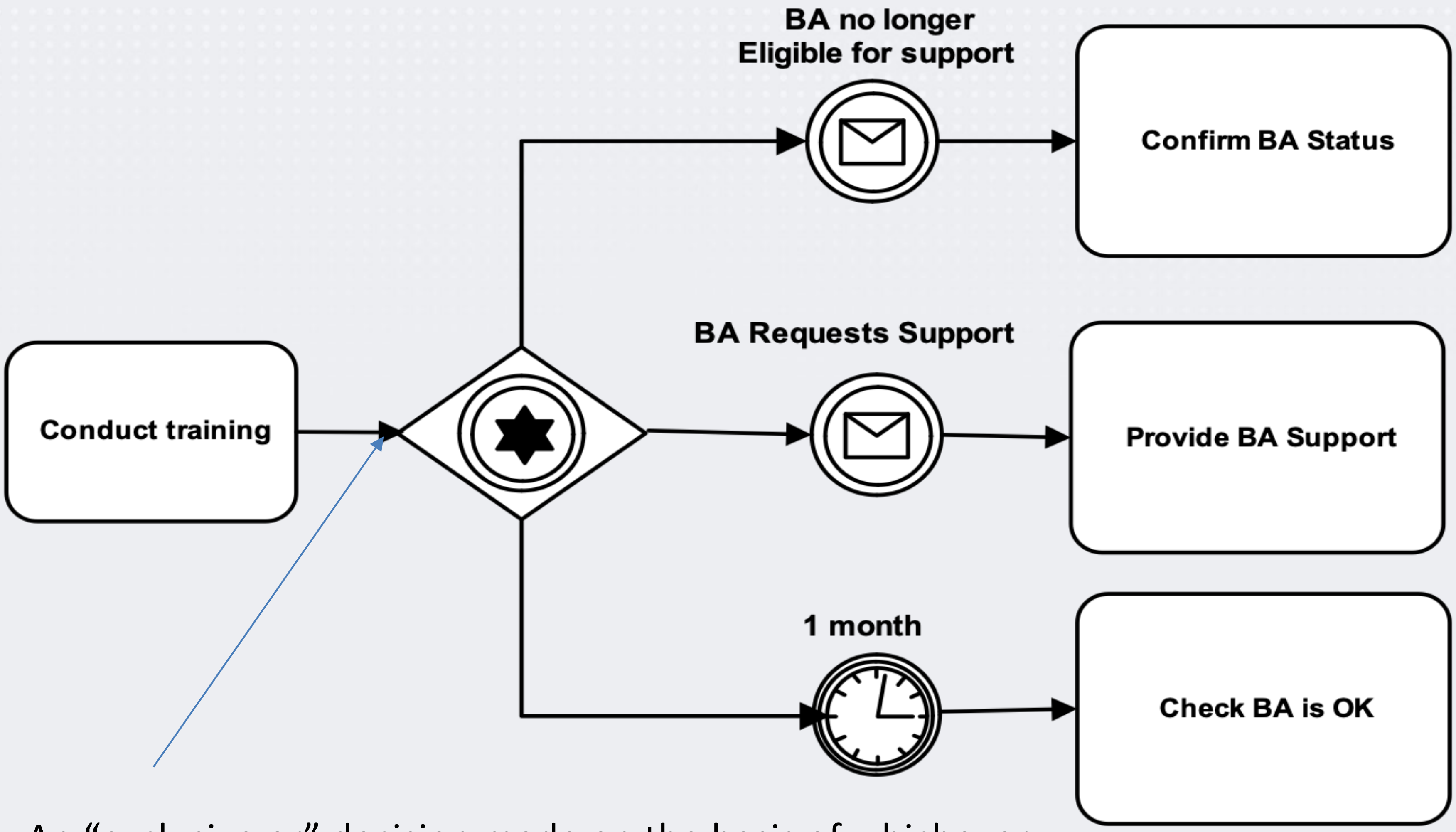


-> An intermediate message can be used to show a process break.

e.g. Having conducted training, the process waits until a BA (Business Administrator) requests support.

Assumption: The next step is *always* BA requests support…?

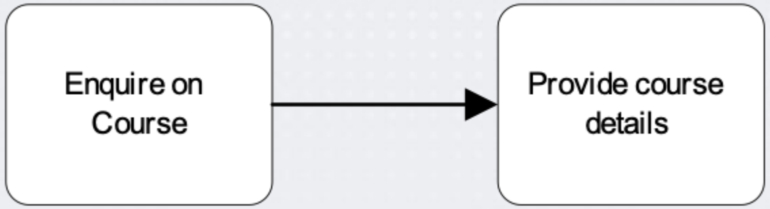
Event Based Gateway



An “exclusive or” decision made on the basis of whichever associated intermediate event occurs first.

Sequence Flow of Control

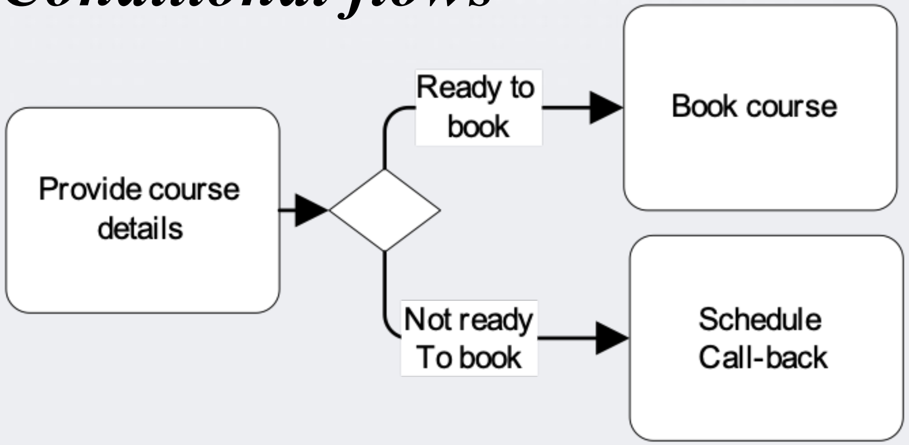
Unconditional flows



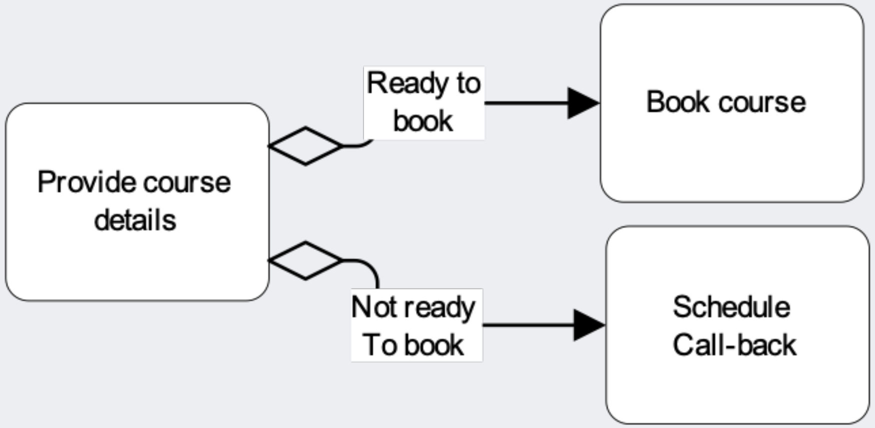
Business rules enforced:

1. when a course enquiry is made, course details are ALWAYS provided.
2. The only time that course details will be provided is when an enquiry is made.

Conditional flows

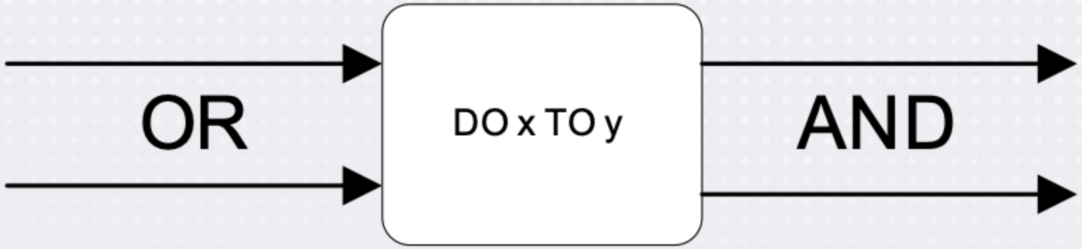


Do these rules reflect business requirements?
Are they workable?



Are these two process flows equivalent?
Which is better?

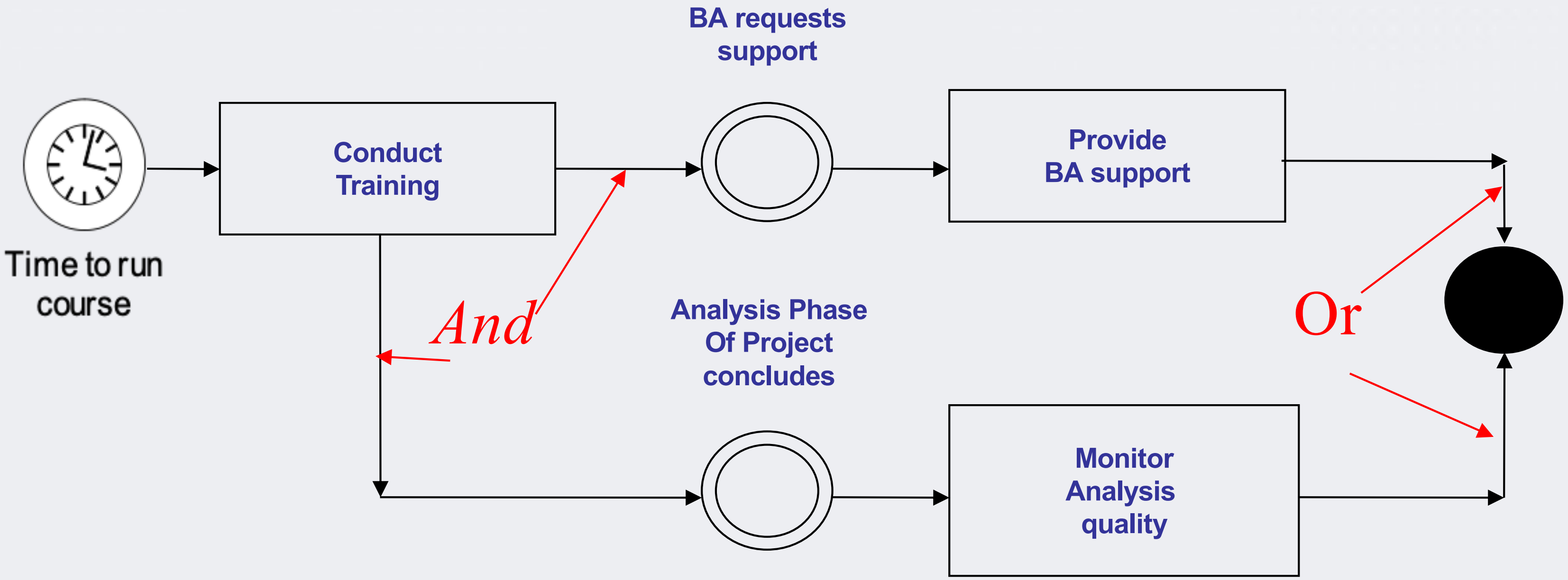
BPMN defaults for Process Flows



OR

DO x TO y

AND



BA requests support

Conduct Training

Provide BA support

Time to run course

And

Analysis Phase Of Project concludes

Or

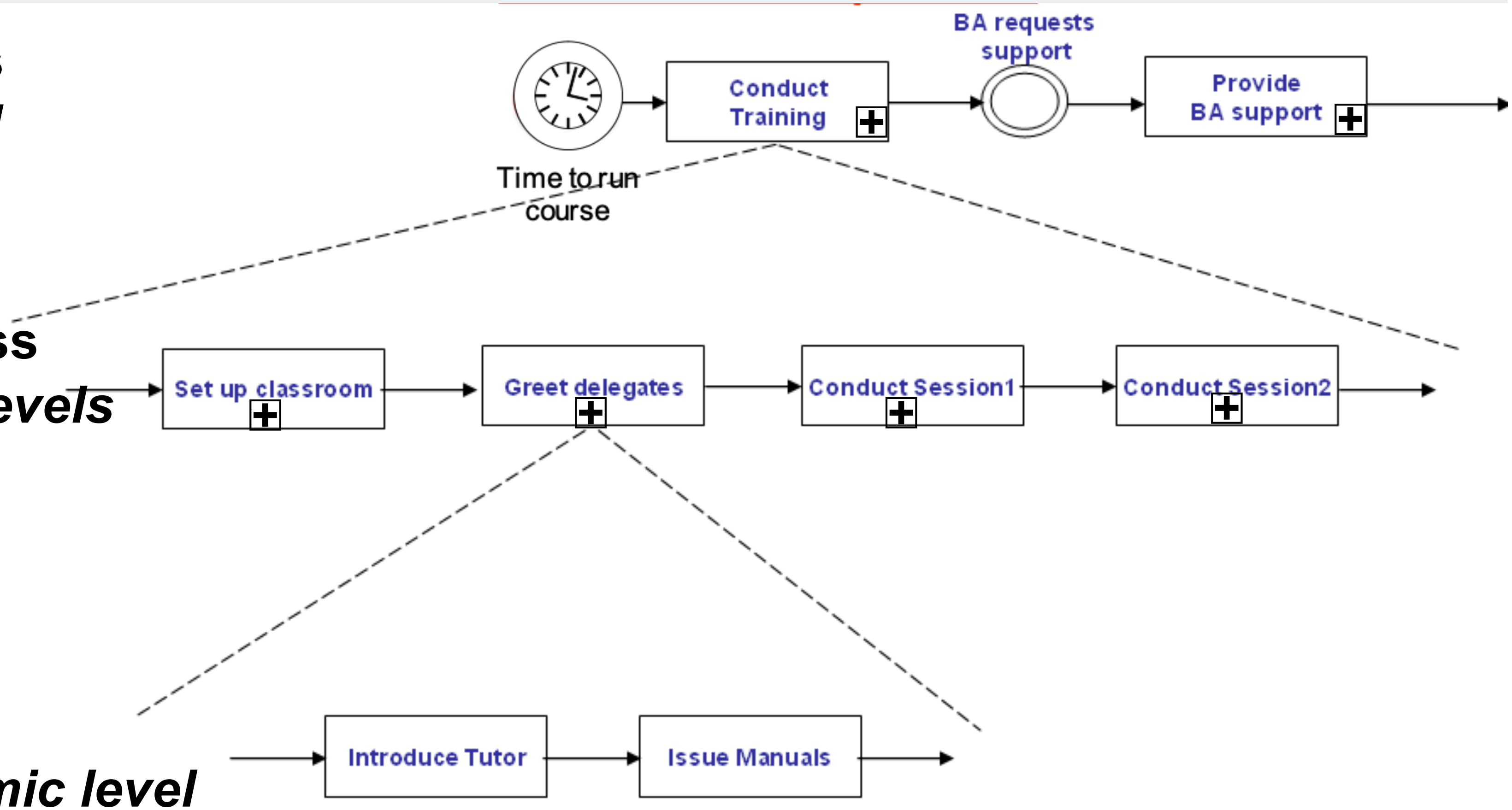
Monitor Analysis quality

Process Decomposition

Activities
Top level

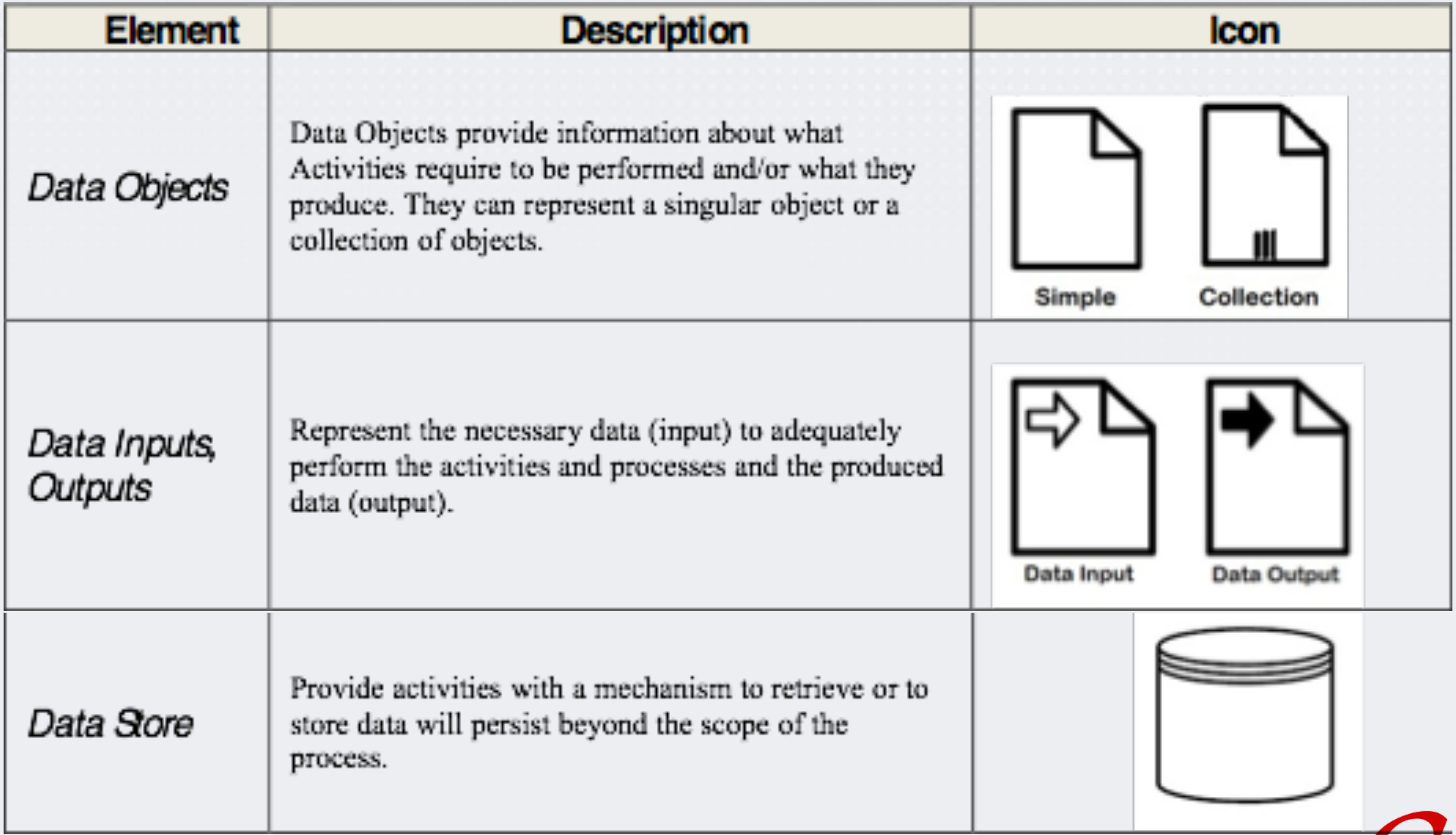
Sub-Process
Intermediate levels

Task
Bottom or atomic level



HiCure
Excellence in Health Informatics Integrated Curricula

Data Objects



Element	Description	Icon
Data Objects	Data Objects provide information about what Activities require to be performed and/or what they produce. They can represent a singular object or a collection of objects.	Simple Collection
Data Inputs, Outputs	Represent the necessary data (input) to adequately perform the activities and processes and the produced data (output).	Data Input Data Output
Data Store	Provide activities with a mechanism to retrieve or to store data will persist beyond the scope of the process.	

HiCure

Excellence in Health Informatics Integrated Curricula

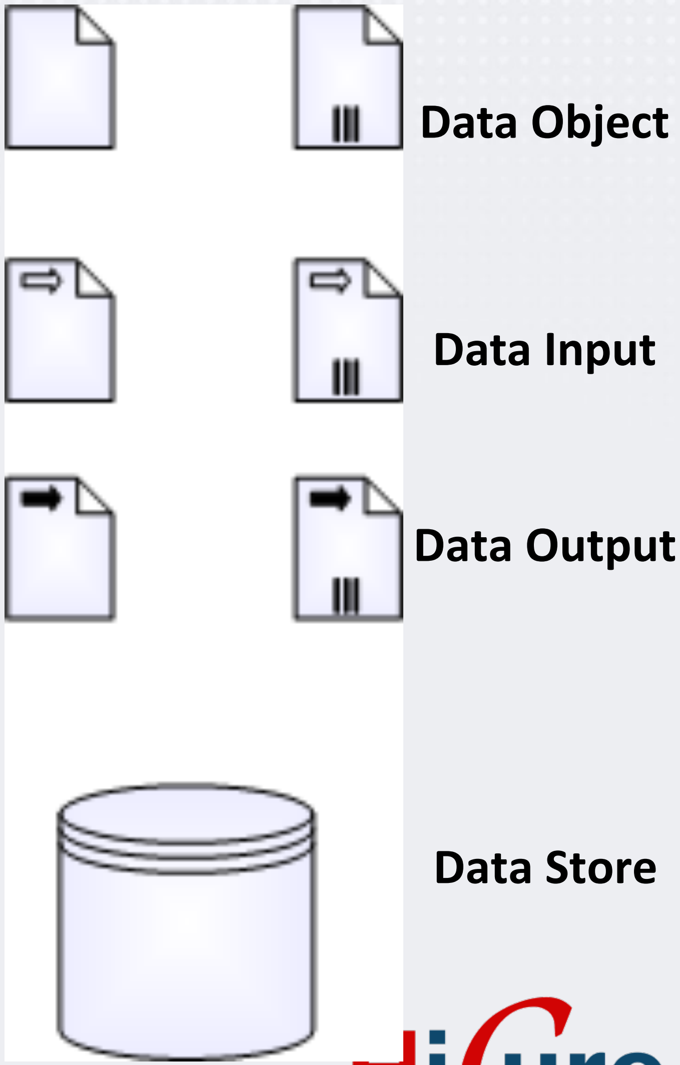
Data Objects

BPMN 2.0 can represent Data in two ways: **Data Stores** and **Data Objects.**

A **Data Store** provides a mechanism for Activities to retrieve or update stored information that will persist beyond the scope of the Process. The same Data Store can be visualised, through a Data Store Reference, in one or more places in the Process.

Data Objects provide information about what Activities require to be performed and/or what they produce, Data Objects can represent a singular object or a collection of objects.

Data Objects can be separated in **Data Input** and **Data Output** only considering direction of the information.



BPMN – Basic Elements

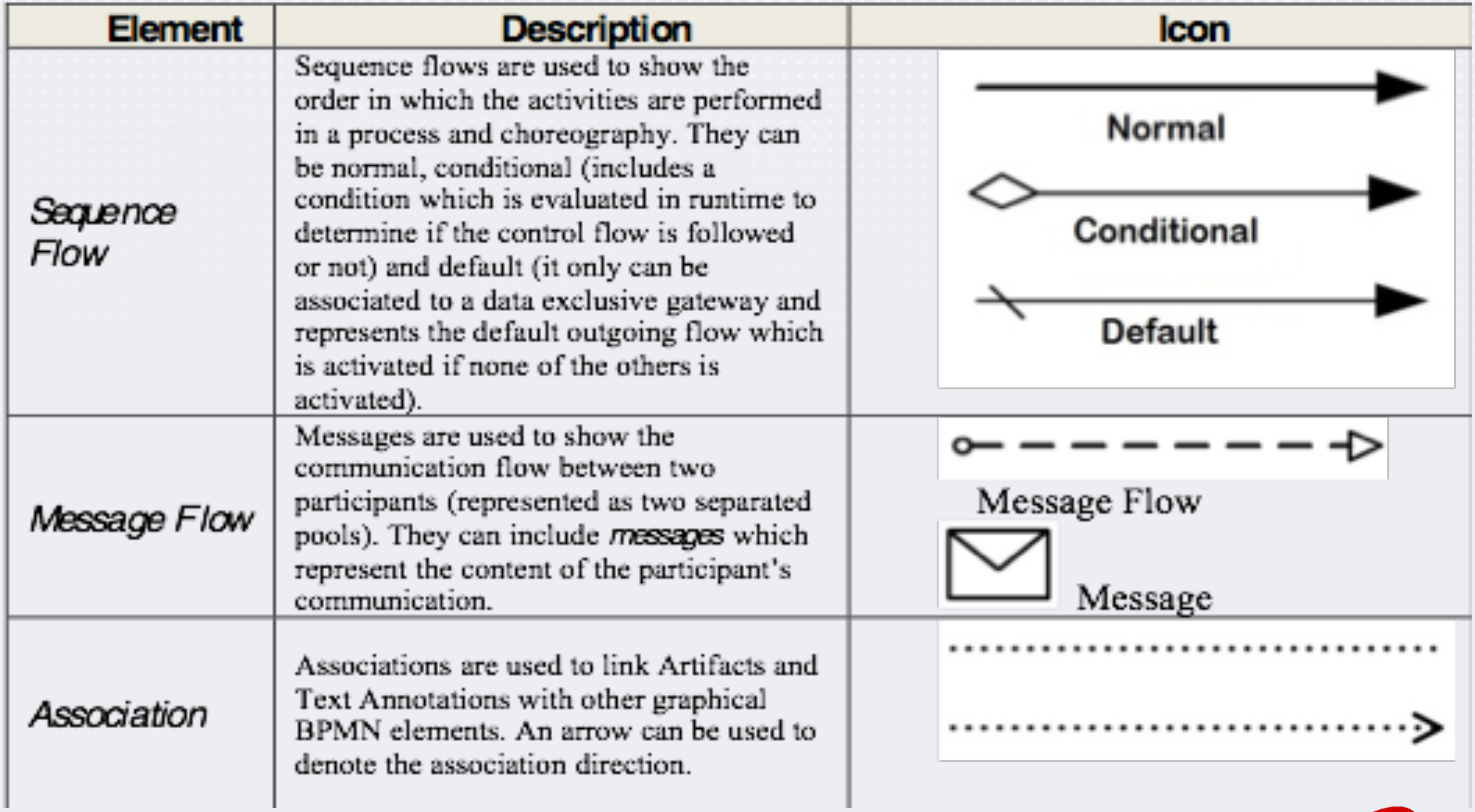
- Connecting Objects:
  - Sequence Flows;
  - Message Flows;
  - Associations;
  - Data Associations.

- **Swimlanes**:
  Used to group the primary modelling elements. Can be of two forms: Pools and Lanes.

- **Artifacts**:

  Used to provide additional information about the process. Include Group, and Text Annotation.

Connecting Objects

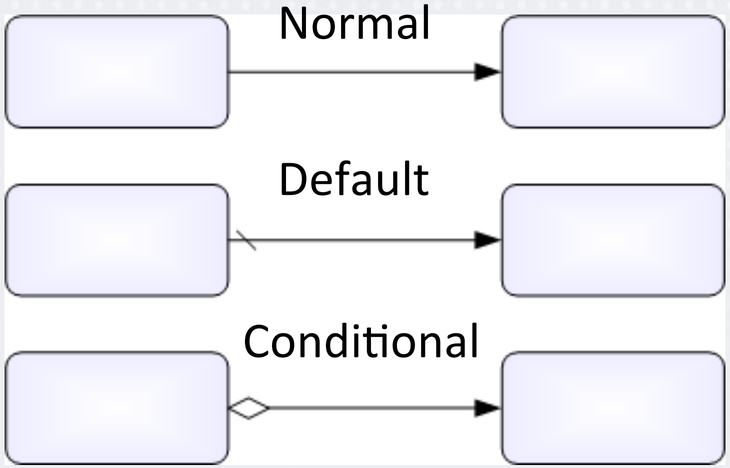


Element	Description	Icon
Sequence Flow	Sequence flows are used to show the order in which the activities are performed in a process and choreography. They can be normal, conditional (includes a condition which is evaluated in runtime to determine if the control flow is followed or not) and default (it only can be associated to a data exclusive gateway and represents the default outgoing flow which is activated if none of the others is activated).	Normal Conditional Default
Message Flow	Messages are used to show the communication flow between two participants (represented as two separated pools). They can include *messages* which represent the content of the participant's communication.	Message Flow Message
Association	Associations are used to link Artifacts and Text Annotations with other graphical BPMN elements. An arrow can be used to denote the association direction.	

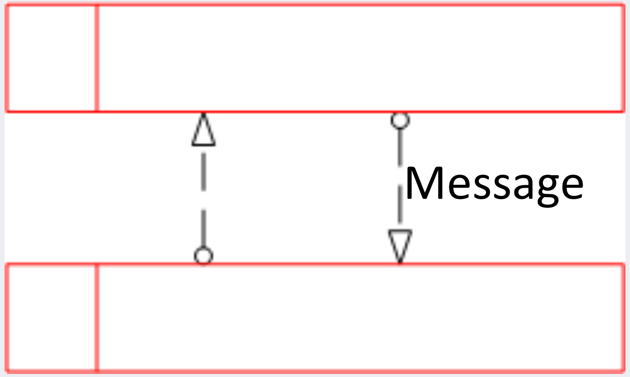
Connection Objects

There are **6 types** or connection objects. All are represented for a **line**.

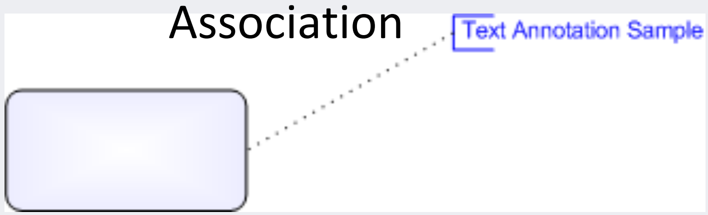
Sequence Flows can be **Normal**, **Default** and **Conditional**, and always have direction, source and target.

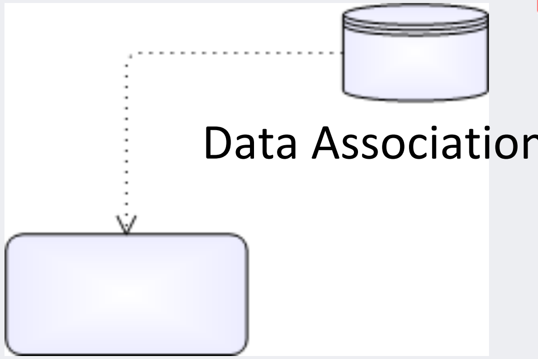


Message Flows **used to represent collaboration between two processes.**

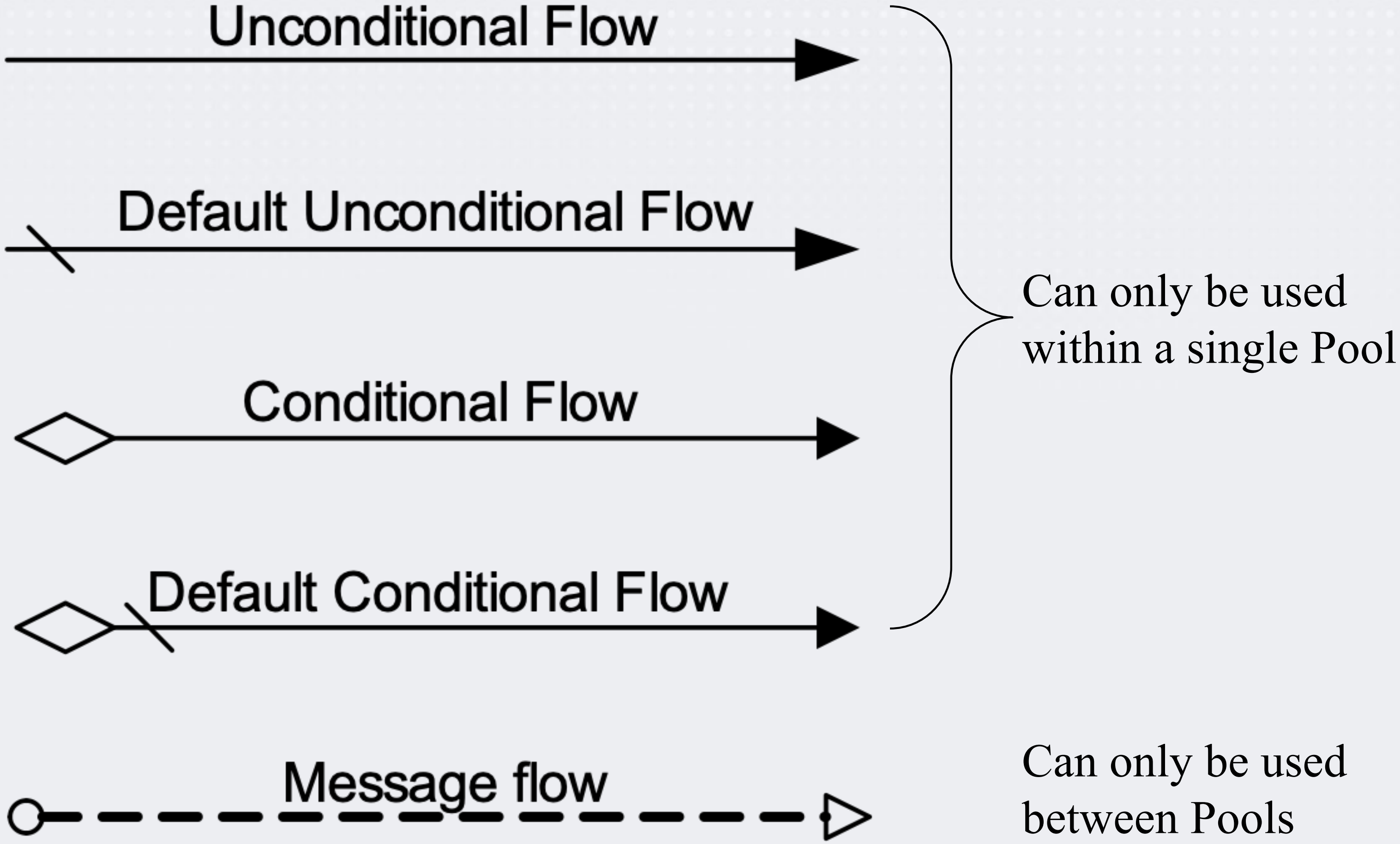


Data Association is a line between a **Data Object** and **an element.**





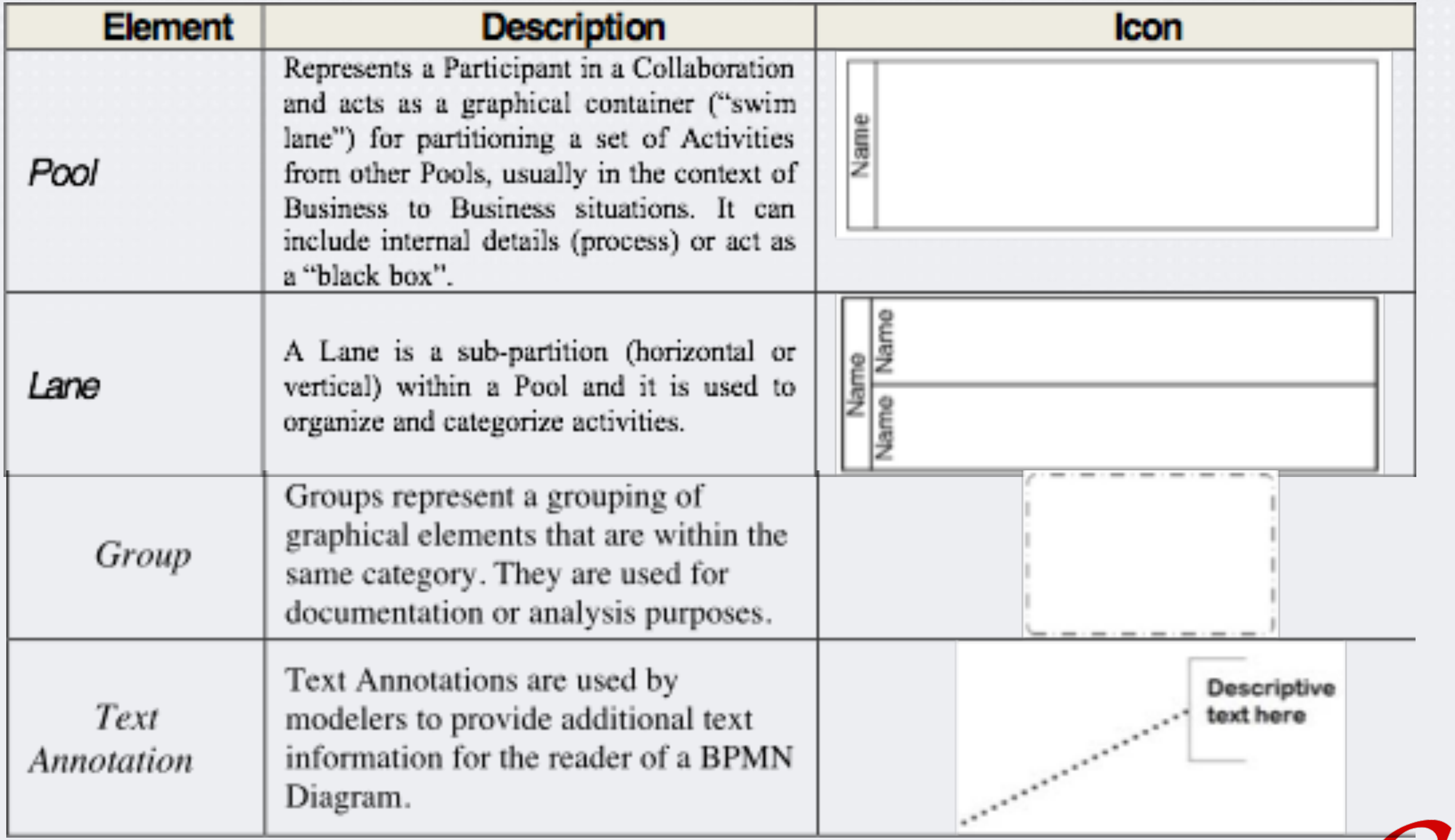
Process flow connectors



Can only be used within a single Pool

Can only be used between Pools

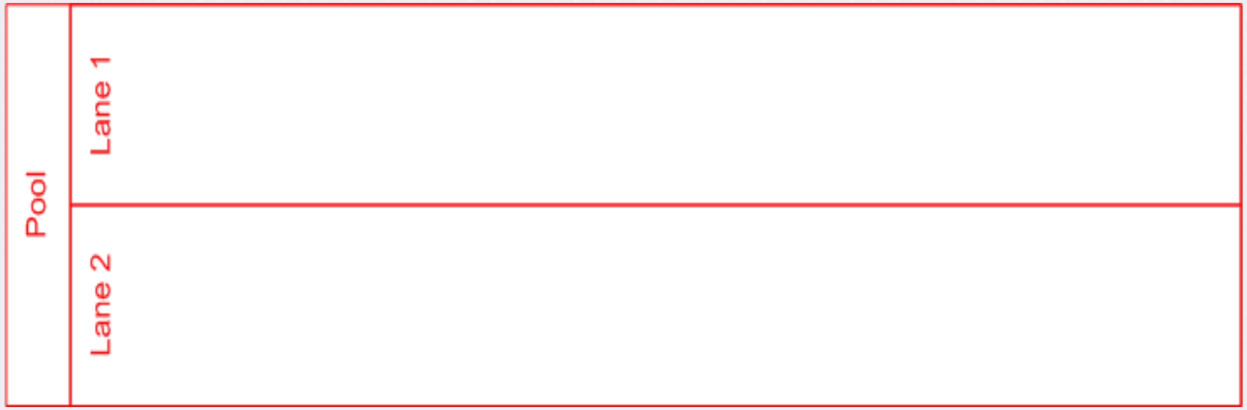
Swimlanes



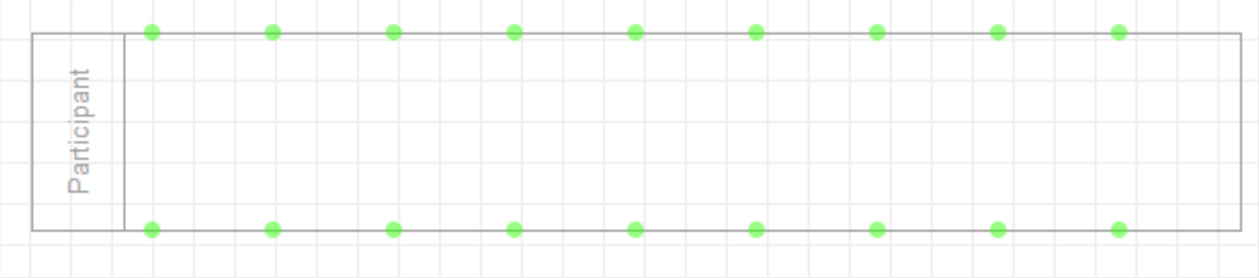
Element	Description	Icon
Pool	Represents a Participant in a Collaboration and acts as a graphical container ("swim lane") for partitioning a set of Activities from other Pools, usually in the context of Business to Business situations. It can include internal details (process) or act as a "black box".	Name
Lane	A Lane is a sub-partition (horizontal or vertical) within a Pool and it is used to organize and categorize activities.	Name Name Name
Group	Groups represent a grouping of graphical elements that are within the same category. They are used for documentation or analysis purposes.	
Text Annotation	Text Annotations are used by modelers to provide additional text information for the reader of a BPMN Diagram.	Descriptive text here

Swimlanes

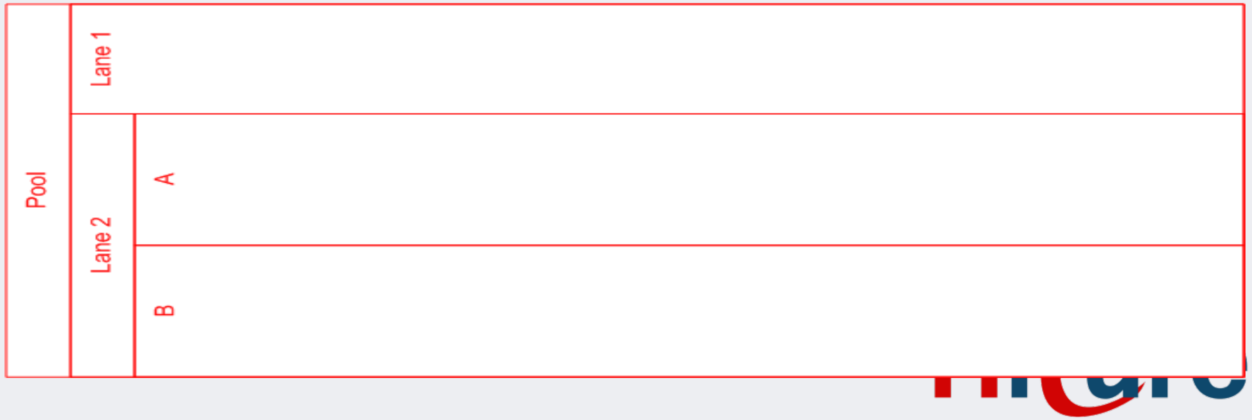
Pool or Laneset is an element representing a process into an organization or company.



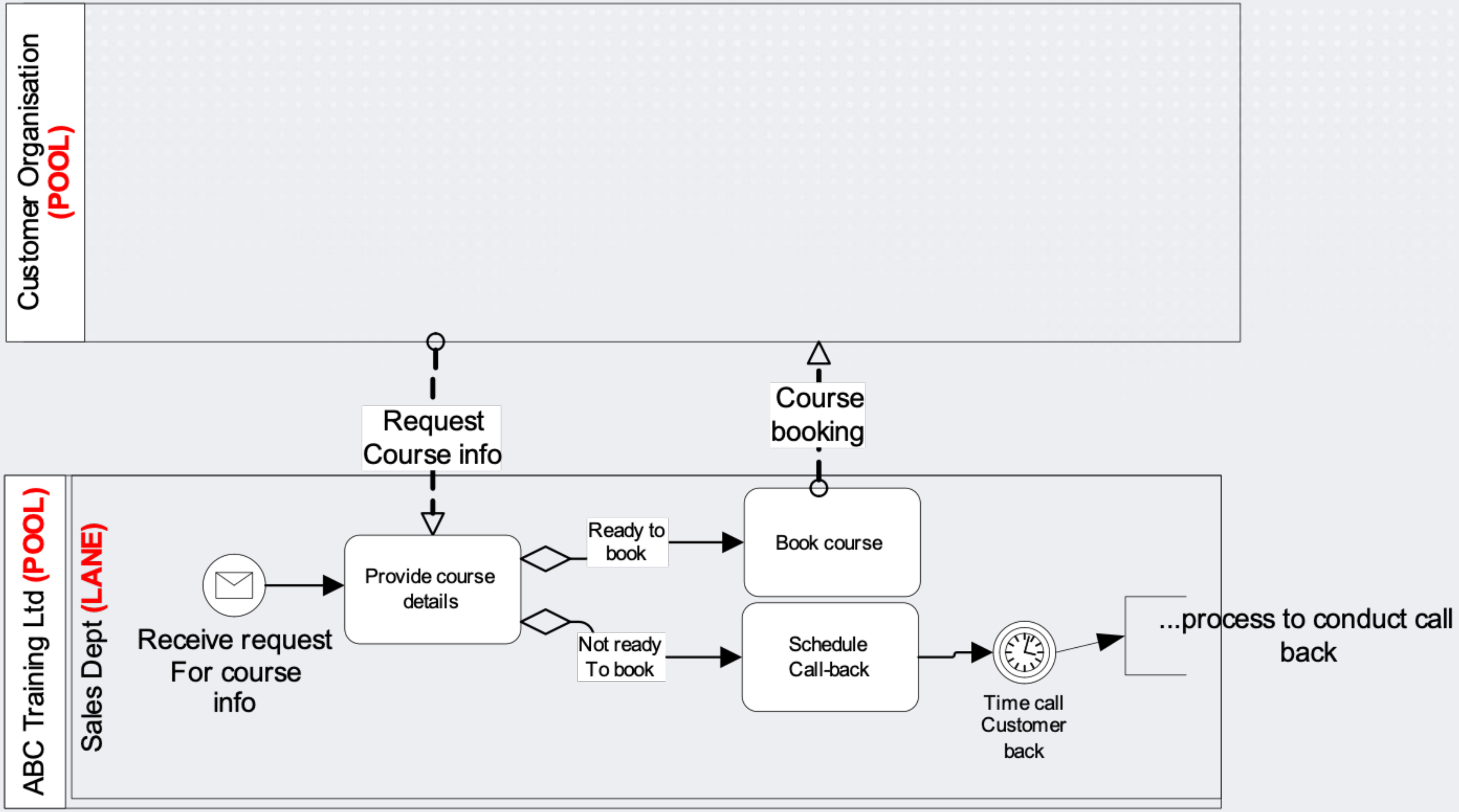
Lane is a representation of an area or department of the company. Some times, a lane can represent a role (of a **participant**) into a process scope.



Participant or Empty Pool is a representation of a process or entity that does not have any action within the process.



Pools, Lanes & Message Flows

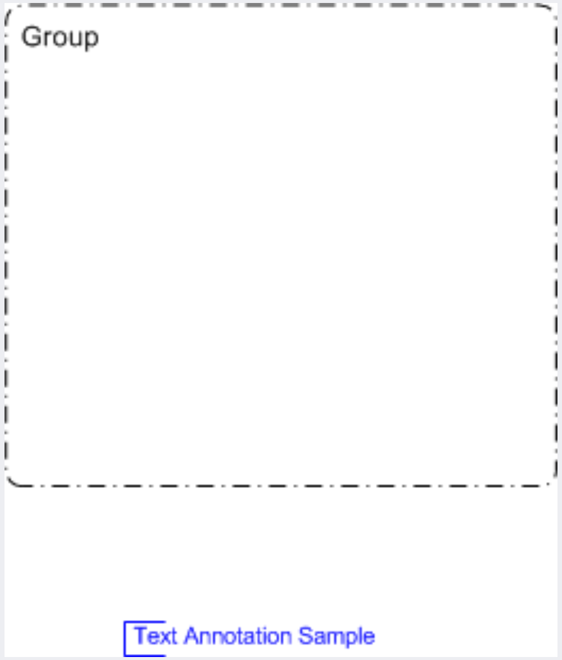


A message cannot flow between two objects in the same Pool

Artifacts

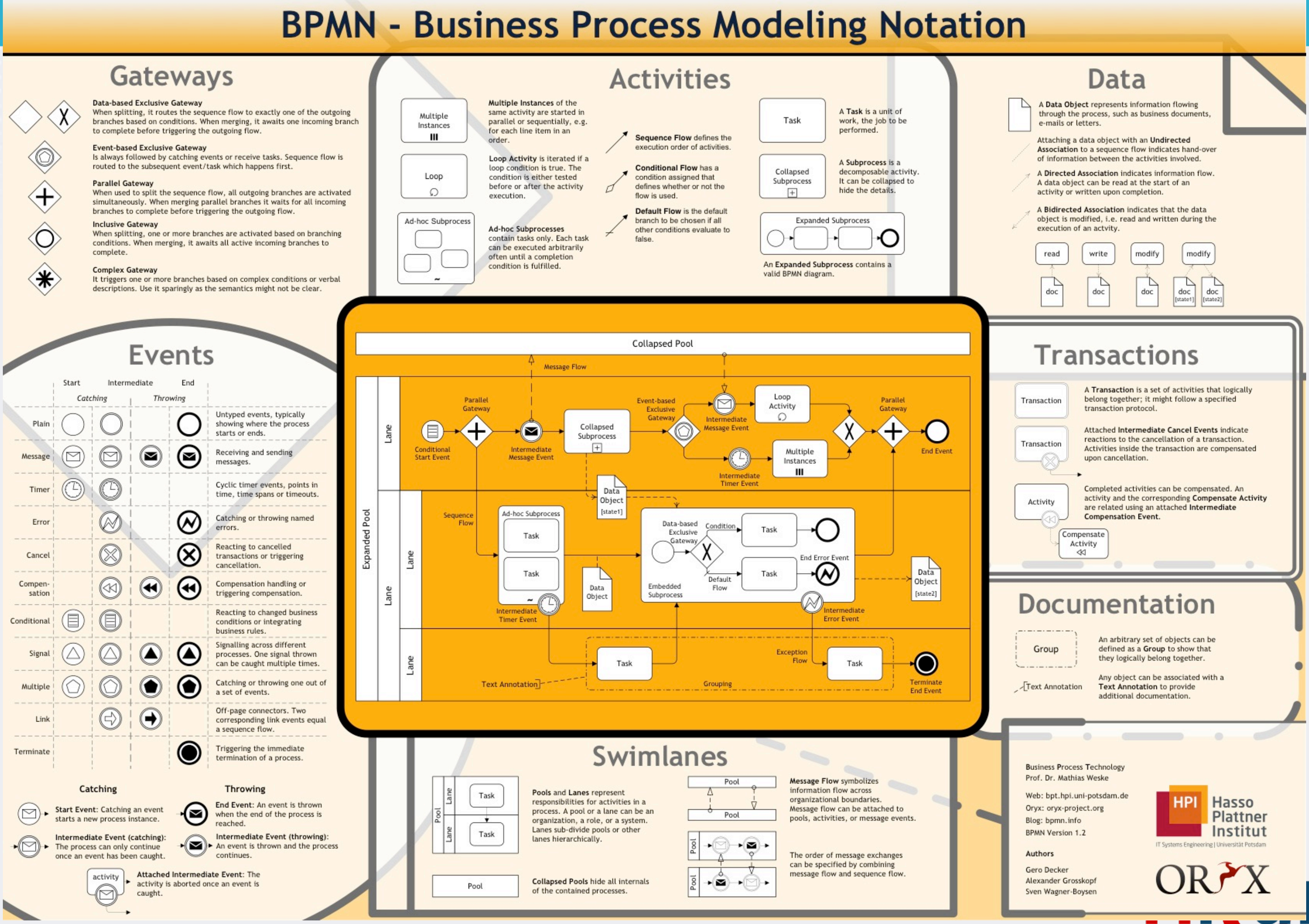
BPMN 2.0 considers 2 types of artifacts: **Groups** and **Text Annotations**.

A **Group** is a grouping of graphical elements that are within the same Category. This type of grouping does not affect the Sequence Flows within the Group. The Category name appears on the diagram as the group label. Categories can be used for documentation or analysis purposes. Groups are one way in which Categories of objects can be visually displayed on the diagram.



Text Annotations are a mechanism for a modeler to provide additional text information for the reader of a BPMN Diagram.

Summary of BPMN 2.0 Notations



BPMN - Business Process Modeling Notation

Gateways

Data-based Exclusive Gateway
When splitting, it routes the sequence flow to exactly one of the outgoing branches based on conditions. When merging, it awaits one incoming branch to complete before triggering the outgoing flow.

Event-based Exclusive Gateway
Is always followed by catching events or receive tasks. Sequence flow is routed to the subsequent event/task which happens first.

Parallel Gateway
When used to split the sequence flow, all outgoing branches are activated simultaneously. When merging parallel branches it waits for all incoming branches to complete before triggering the outgoing flow.

Inclusive Gateway
When splitting, one or more branches are activated based on branching conditions. When merging, it awaits all active incoming branches to complete.

Complex Gateway
It triggers one or more branches based on complex conditions or verbal descriptions. Use it sparingly as the semantics might not be clear.

Activities

Multiple Instances III

Multiple Instances of the same activity are started in parallel or sequentially, e.g. for each line item in an order.

Loop

Loop Activity is iterated if a loop condition is true. The condition is either tested before or after the activity execution.

Ad-hoc Subprocess

Ad-hoc Subprocesses contain tasks only. Each task can be executed arbitrarily often until a completion condition is fulfilled.

Sequence Flow defines the execution order of activities.

Conditional Flow has a condition assigned that defines whether or not the flow is used.

Default Flow is the default branch to be chosen if all other conditions evaluate to false.

Task

A **Task** is a unit of work, the job to be performed.

Collapsed Subprocess ⊞

A **Subprocess** is a decomposable activity. It can be collapsed to hide the details.

Expanded Subprocess

An **Expanded Subprocess** contains a valid BPMN diagram.

Data

A **Data Object** represents information flowing through the process, such as business documents, e-mails or letters.

Attaching a data object with an **Undirected Association** to a sequence flow indicates hand-over of information between the activities involved.

A **Directed Association** indicates information flow. A data object can be read at the start of an activity or written upon completion.

A **Bidirected Association** indicates that the data object is modified, i.e. read and written during the execution of an actvity.

read	write	modify	modify	
doc	doc	doc	doc [state1]	doc [state2]

Events

	Start	Intermediate		End	
	Catching		*Throwing*		
Plain	○	◎		●	Untyped events, typically showing where the process starts or ends.
Message	○	◎	◉	●	Receiving and sending messages.
Timer	○	◎			Cyclic timer events, points in time, time spans or timeouts.
Error		◎		●	Catching or throwing named errors.
Cancel		◎		●	Reacting to cancelled transactions or triggering cancellation.
Compen-sation		◎	◉	●	Compensation handling or triggering compensation.
Conditional	○	◎			Reacting to changed business conditions or integrating business rules.
Signal	○	◎	◉	●	Signalling across different processes. One signal thrown can be caught multiple times.
Multiple	○	◎	◉	●	Catching or throwing one out of a set of events.
Link		◎	◉		Off-page connectors. Two corresponding link events equal a sequence flow.
Terminate				●	Triggering the immediate termination of a process.

Catching

Start Event: Catching an event starts a new process instance.

Intermediate Event (catching): The process can only continue once an event has been caught.

Throwing

End Event: An event is thrown when the end of the process is reached.

Intermediate Event (throwing): An event is thrown and the process continues.

activity

Attached Intermediate Event: The activity is aborted once an event is caught.

Transactions

Transaction

A **Transaction** is a set of activities that logically belong together; it might follow a specified transaction protocol.

Transaction

Attached **Intermediate Cancel Events** indicate reactions to the cancellation of a transaction. Activities inside the transaction are compensated upon cancellation.

Activity

Compensate Activity

Completed activities can be compensated. An activity and the corresponding **Compensate Activity** are related using an attached **Intermediate Compensation Event**.

Documentation

Group

An arbitrary set of objects can be defined as a **Group** to show that they logically belong together.

Text Annotation

Any object can be associated with a **Text Annotation** to provide additional documentation.

Swimlanes

Pool | Lane | Task | Lane | Task

Pools and **Lanes** represent responsibilities for activities in a process. A pool or a lane can be an organization, a role, or a system. Lanes sub-divide pools or other lanes hierarchically.

Pool

Collapsed Pools hide all internals of the contained processes.

Pool | Pool

Message Flow symbolizes information flow across organizational boundaries. Message flow can be attached to pools, activities, or message events.

Pool | Pool

The order of message exchanges can be specified by combining message flow and sequence flow.

Business Process Technology
Prof. Dr. Mathias Weske

Web: bpt.hpi.uni-potsdam.de
Oryx: oryx-project.org
Blog: bpmn.info
BPMN Version 1.2

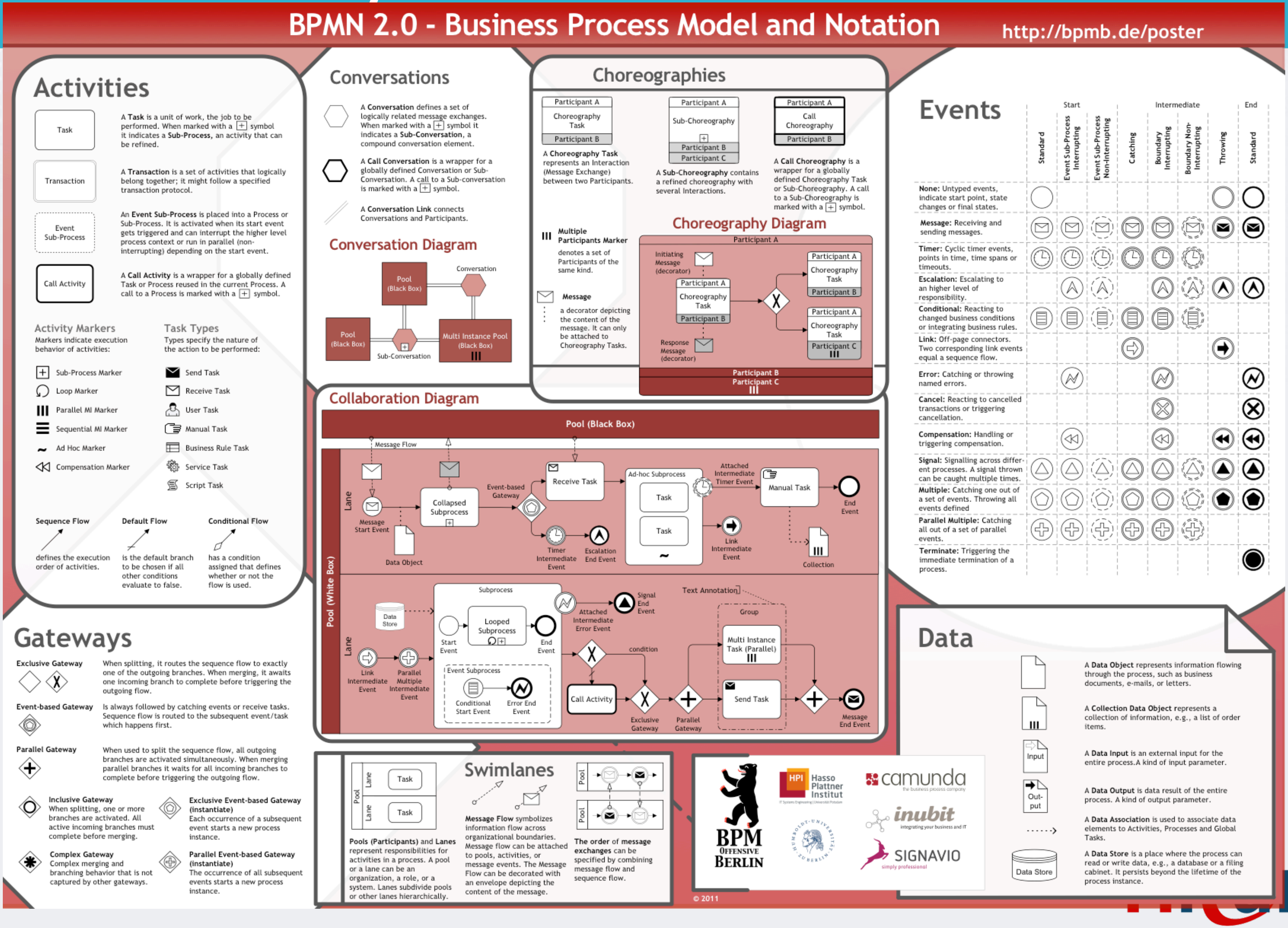
Authors

Gero Decker
Alexander Grosskopf
Sven Wagner-Boysen

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

ORYX

Summary of BPMN 2.0 Notations



BPMN 2.0 - Business Process Model and Notation

http://bpmb.de/poster

Activities

A **Task** is a unit of work, the job to be performed. When marked with a ⊞ symbol it indicates a **Sub-Process**, an activity that can be refined.

A **Transaction** is a set of activities that logically belong together; it might follow a specified transaction protocol.

An **Event Sub-Process** is placed into a Process or Sub-Process. It is activated when its start event gets triggered and can interrupt the higher level process context or run in parallel (non-interrupting) depending on the start event.

A **Call Activity** is a wrapper for a globally defined Task or Process reused in the current Process. A call to a Process is marked with a ⊞ symbol.

Activity Markers

Markers indicate execution behavior of activities:

- Sub-Process Marker
- Loop Marker
- Parallel MI Marker
- Sequential MI Marker
- Ad Hoc Marker
- Compensation Marker

Task Types

Types specify the nature of the action to be performed:

- Send Task
- Receive Task
- User Task
- Manual Task
- Business Rule Task
- Service Task
- Script Task

Sequence Flow

defines the execution order of activities.

Default Flow

is the default branch to be chosen if all other conditions evaluate to false.

Conditional Flow

has a condition assigned that defines whether or not the flow is used.

Gateways

Exclusive Gateway

When splitting, it routes the sequence flow to exactly one of the outgoing branches. When merging, it awaits one incoming branch to complete before triggering the outgoing flow.

Event-based Gateway

Is always followed by catching events or receive tasks. Sequence flow is routed to the subsequent event/task which happens first.

Parallel Gateway

When used to split the sequence flow, all outgoing branches are activated simultaneously. When merging parallel branches it waits for all incoming branches to complete before triggering the outgoing flow.

Inclusive Gateway
When splitting, one or more branches are activated. All active incoming branches must complete before merging.

Complex Gateway
Complex merging and branching behavior that is not captured by other gateways.

Exclusive Event-based Gateway (instantiate)
Each occurrence of a subsequent event starts a new process instance.

Parallel Event-based Gateway (instantiate)
The occurrence of all subsequent events starts a new process instance.

Conversations

A **Conversation** defines a set of logically related message exchanges. When marked with a ⊞ symbol it indicates a **Sub-Conversation**, a compound conversation element.

A **Call Conversation** is a wrapper for a globally defined Conversation or Sub-Conversation. A call to a Sub-conversation is marked with a ⊞ symbol.

A **Conversation Link** connects Conversations and Participants.

Conversation Diagram

Choreographies

A **Choreography Task** represents an Interaction (Message Exchange) between two Participants.

A **Sub-Choreography** contains a refined choreography with several Interactions.

A **Call Choreography** is a wrapper for a globally defined Choreography Task or Sub-Choreography. A call to a Sub-Choreography is marked with a ⊞ symbol.

Multiple Participants Marker

denotes a set of Participants of the same kind.

Message

a decorator depicting the content of the message. It can only be attached to Choreography Tasks.

Choreography Diagram

Events

	Start			Intermediate				End
	Standard	Event Sub-Process Interrupting	Event Sub-Process Non-Interrupting	Catching	Boundary Interrupting	Boundary Non-Interrupting	Throwing	Standard
None: Untyped events, indicate start point, state changes or final states.	○						○	○
Message: Receiving and sending messages.	○	○	○	○	○	○	●	●
Timer: Cyclic timer events, points in time, time spans or timeouts.	○	○	○	○	○	○		
Escalation: Escalating to an higher level of responsibility.		○	○		○	○	●	●
Conditional: Reacting to changed business conditions or integrating business rules.	○	○	○	○	○	○		
Link: Off-page connectors. Two corresponding link events equal a sequence flow.				○			●	
Error: Catching or throwing named errors.		○			○			●
Cancel: Reacting to cancelled transactions or triggering cancellation.					○			●
Compensation: Handling or triggering compensation.		○			○		●	●
Signal: Signalling across different processes. A signal thrown can be caught multiple times.	○	○	○	○	○	○	●	●
Multiple: Catching one out of a set of events. Throwing all events defined	○	○	○	○	○	○	●	●
Parallel Multiple: Catching all out of a set of parallel events.	○	○	○	○	○	○		
Terminate: Triggering the immediate termination of a process.								●

Collaboration Diagram

Swimlanes

Pools (Participants) and **Lanes** represent responsibilities for activities in a process. A pool or a lane can be an organization, a role, or a system. Lanes subdivide pools or other lanes hierarchically.

Message Flow symbolizes information flow across organizational boundaries. Message flow can be attached to pools, activities, or message events. The Message Flow can be decorated with an envelope depicting the content of the message.

The order of message exchanges can be specified by combining message flow and sequence flow.

Data

A **Data Object** represents information flowing through the process, such as business documents, e-mails, or letters.

A **Collection Data Object** represents a collection of information, e.g., a list of order items.

A **Data Input** is an external input for the entire process.A kind of input parameter.

A **Data Output** is data result of the entire process. A kind of output parameter.

A **Data Association** is used to associate data elements to Activities, Processes and Global Tasks.

A **Data Store** is a place where the process can read or write data, e.g., a database or a filing cabinet. It persists beyond the lifetime of the process instance.

BPM Offensive Berlin
HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam
camunda
the business process company
inubit
integrating your business and IT
SIGNAVIO
simply professional