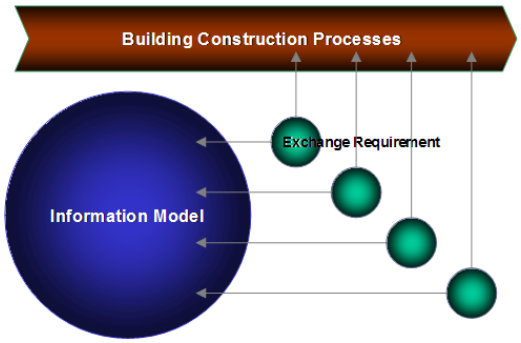



# IDM
# Process Mapping

*Jeffrey Wix*

# What is a process map

- A process map is used to understand:
  - The tasks (activities) performed within a business process
  - The sequence in which they are carried out
  - The actors (people/organizations) involved in the business process
  - The information that is exchanged between actors as a result of activities being completed.
- A process map can also identify:
  - The start and end events in the business process
  - Events at which an information exchange occurs
  - Decisions in the business process

# The way to draw process maps

- There are lots of ways to draw process maps
- The way in which it is drawn is called a ‘notation’
- One common ‘notation’ is IDEF0
  - Most widely used to date for building construction process maps
- But, the notation used in IDM is called ‘Business Process Modelling Notation’ or BPMN
  - Better capability to express business process
  - Supports multiple actors
  - Better able to define information

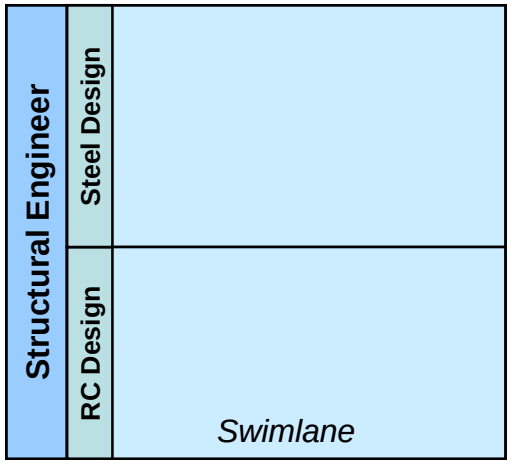
# Key process mapping objects

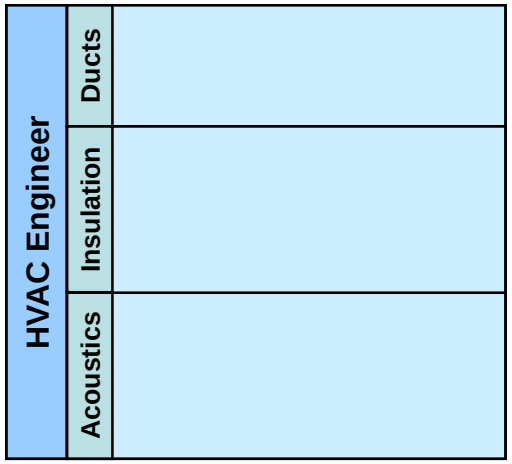
- The key objects used to define IDM process maps are:
  - **Swimming pools/swimlanes** that are used to distinguish actors
  - **Activities** that describe work tasks
  - **Events** which are points in time at which something notable happens
  - **Gateways** which are points at which processes may diverge or converge (decisions)
  - **Connecting objects** that join together the objects in the process map
  - **Data objects** that define the information to be exchanged between activities
- Each of these will now be discussed in detail

# Swimming Pool


- In a process map, the activities for each actor are shown within a ‘container’
- The container is known as a ‘swimming pool’
- A pool may be shown horizontally or vertically in a process map
- Each pool has a name that identifies the actor concerned
  - actor is identified by role, not by name
- Each actor is shown within their own swimming pool

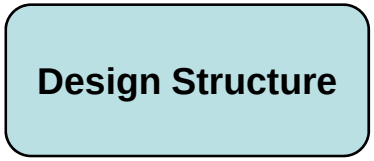
# Swimlane

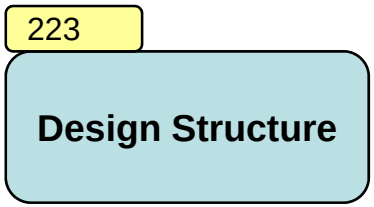



- Activities may be organized in various ways for each actor
- One way of organizing activities is by department or division.
- This can be shown by adding ‘swimlanes’ to the swimming pool
- Each swimlane has a name that identifies the organization unit concerned
  - this is by role, not by name
- Every swimming pool may have its own set of lanes

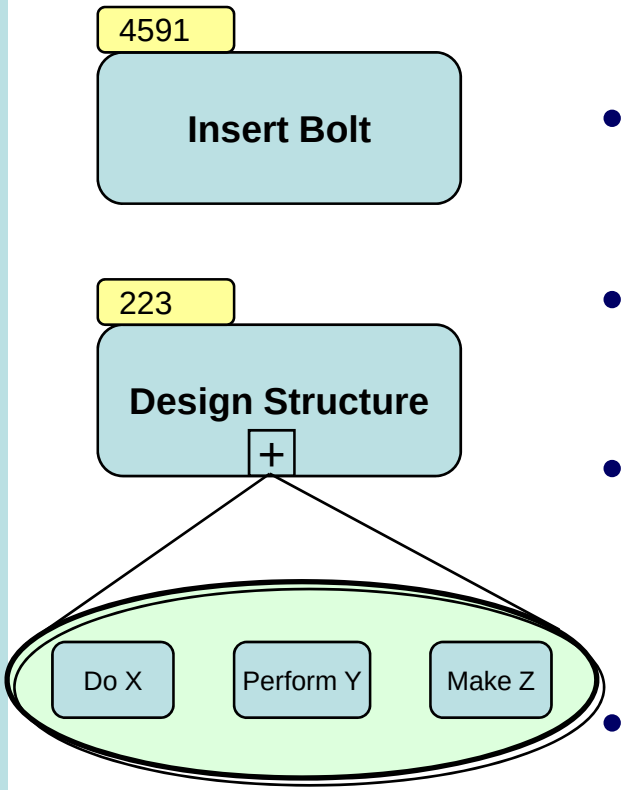
# Activities



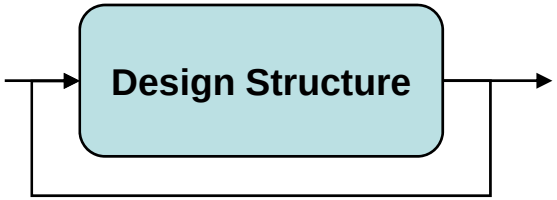

- Activity is a general name given to any work done by an actor that may appear in a process map.
- An activity is shown as a rectangle (normally as a rounded rectangle).
- An activity is named
  - The name of an activity is usually expressed as a ‘verb phrase’ in the form *‘action → object’*
  - That is, the action to be performed followed by the object that it is to be performed on.
- Activities also usually have an identifier other than their name

# Compound Activities

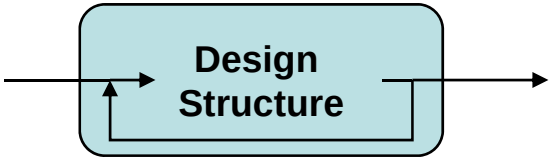



- Activities may be either atomic or compound
- An atomic activity is complete; it cannot be broken down any further
- A compound activity can be further broken down into other activities
- A compound activity is shown with a '+' sign to indicate that it is 'collapsed'
- Eventually, a compound activity will resolve into a set of atomic activities

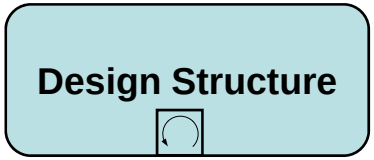
# Activity Feedback



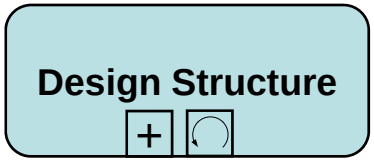

- In practice, most activities will have a feedback loop that enables output to be corrected
- Showing lots of feedback loops would make a cluttered process map




- A lot of feedback is ‘self feedback’ that is effectively within the activity



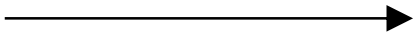

- This can be described by a ‘loop’ marker in the activity



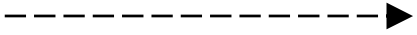

- Loop and ‘collapsed’ markers can be used together in the same activity

# Connecting Activities

- In BPMN, there are two types of connecting objects that can be used to link activities (and other types of flow object)



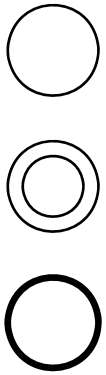
- A sequence arrow is an unbroken line in the direction of the arrow
- It is used to link activities that are within the same swimming pool or swimlane



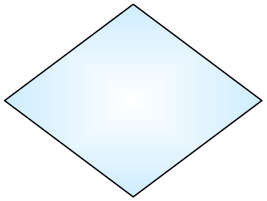
- A message arrow is a dashed line
- It is used to pass information between activities that are in different swimming pools or swimlanes

# Events

- An event is something that “happens” during a business process
- Events affect the flow of the process and usually have a cause (trigger) or an impact (result)
- There are three types of Events, based on when they affect the flow:
  - Start events
    - Occur at the beginning of a process
  - Intermediate events
    - Occur whilst a process is happening
  - End events
    - Occur at the end of, or to cause the end of, a process



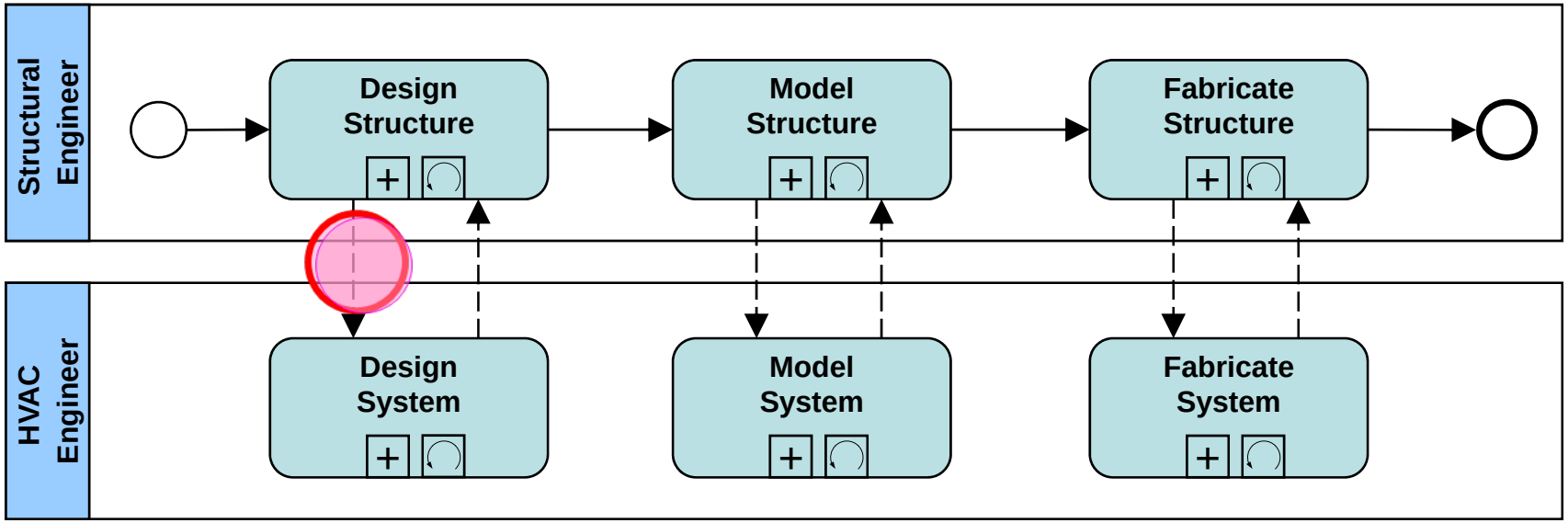
# Gateways / Decisions



- A gateway is used to control the splitting and joining of Sequence Flows
- It is like a decision
- A gateway is shown as a diamond shape symbol
- Internal Markers within gateways can indicate the type of behavior control

# Making a process map

- We now have enough ‘things’ to make a process map
  - Create the swimming pool
  - Identify start and end events (ignore intermediate events)
  - Place activities within the pool
  - Connect activities in sequence
  - Specify messages between activities



# Extending Events (Triggers)

- **Internal Markers within events indicate the type of trigger**

- None
  - No specific trigger

- Message
  - Event triggered by the need for a message

- Timer
  - Event occurring after a specific period of time

- Rule
  - Event triggered by a rule occurrence

- Link

- Multiple
  - Multiple triggers

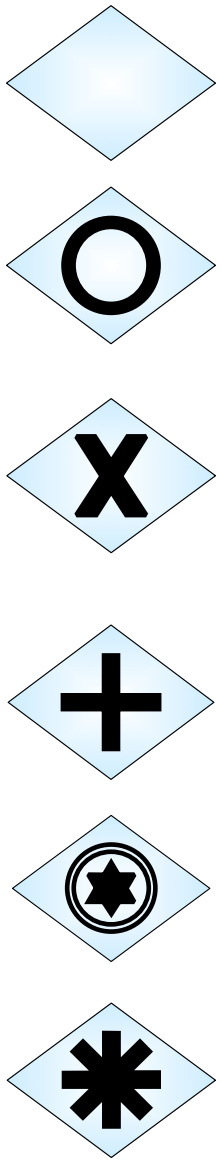
- Error
  - Error triggers event

- Cancel
  - Event to cancel process

- Compensation
  - Event to compensate

- Terminate
  - Termination event

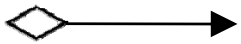
# Extending Gateways (Behavior Controls)



- None
  - No specifically identified behavior
- OR
  - Output sequence will be one route OR another
- Exclusive OR
  - Output sequence will be exclusively one route
- AND
  - All routes are taken
- Event based
  - Output route is selected based on the occurrence of an event
- Complex
  - Output route is selected based on a complex set of choices

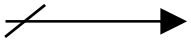
# Extending Connecting Objects

- Connecting objects may be further elaborated depending on how the sequence is driven



*Conditional Sequence Flow*

- A sequence flow that can have conditional expressions that are evaluated at runtime to determine whether or not the flow will be used.



*Default Sequence Flow*

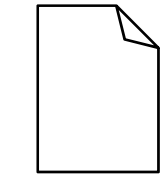
- Sets the Default condition flow in a branching split through a gateway.



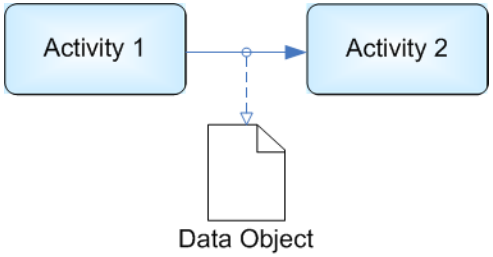
*Association*

- An Association is used to associate information with flow objects.

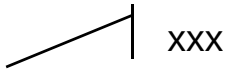
# Artefacts



*Data Object*




- Data Objects provide information about what activities require be performed and/or what they produce.
- Data Objects are connected to sequence flows, usually using message flows and define data between activities.



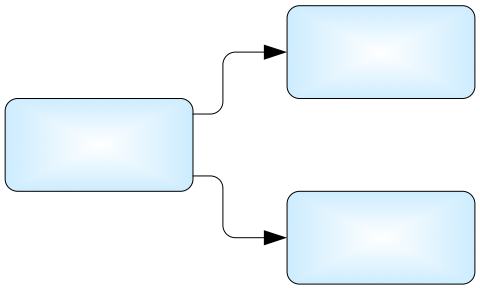

*Annotation*

- Annotations provide additional information for the reader of a BPMN Diagram.

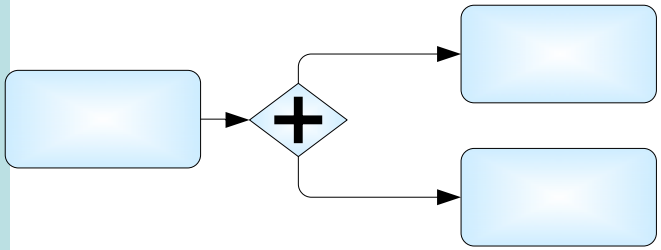
*Group*

- A grouping of activities that does not affect the Sequence Flow.

# Using Gateways (1)

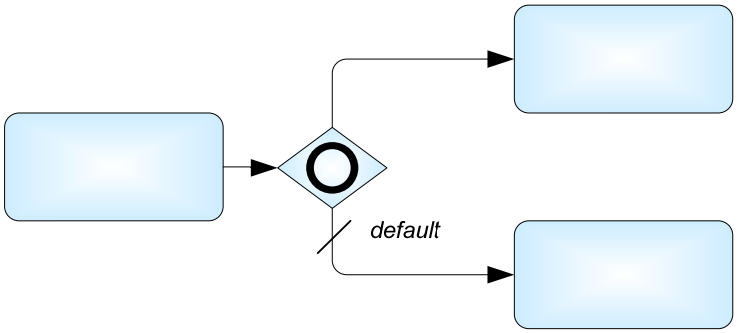


- A ‘split’ enables activities to be performed concurrently, rather than sequentially.
- Multiple outgoing sequence flow can be used
- Each flow is “uncontrolled”
- This is the preferred method for most situations.

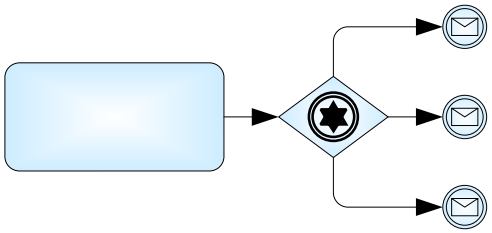


- A ‘branch fork’ also enables activities to be performed concurrently.
- Rarely used

# Using Gateways (2)

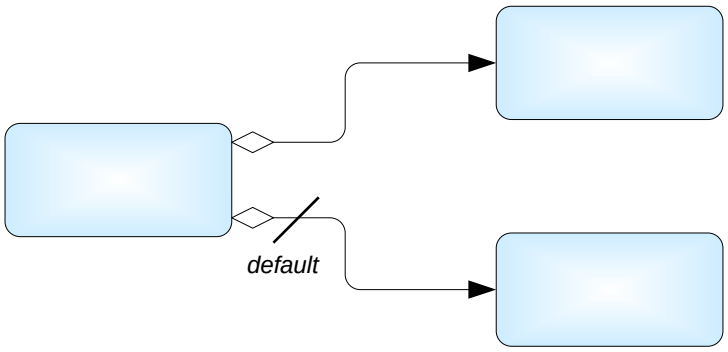



- An OR gateway distributes flow in one direction OR another.
- A default flow direction can be indicated.
- Each flow direction can have conditions applied that determine or control flow direction.
- Can be used for any data based or event based gateway

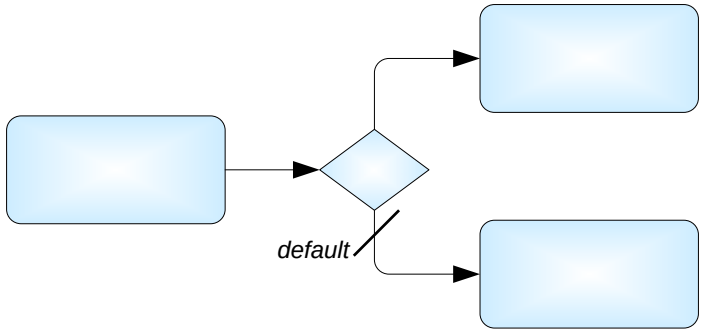


- An event based OR gateway distributes flow to a branch based on the results of a particular event.

# Using Gateways (3)




- A branch where alternatives are based on conditional expressions in the output.
- Generally these are returned as Yes/No or TRUE/FALSE 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 by using a default condition.




- Alternative branching may also be shown at an ‘inclusive’ OR gateway.

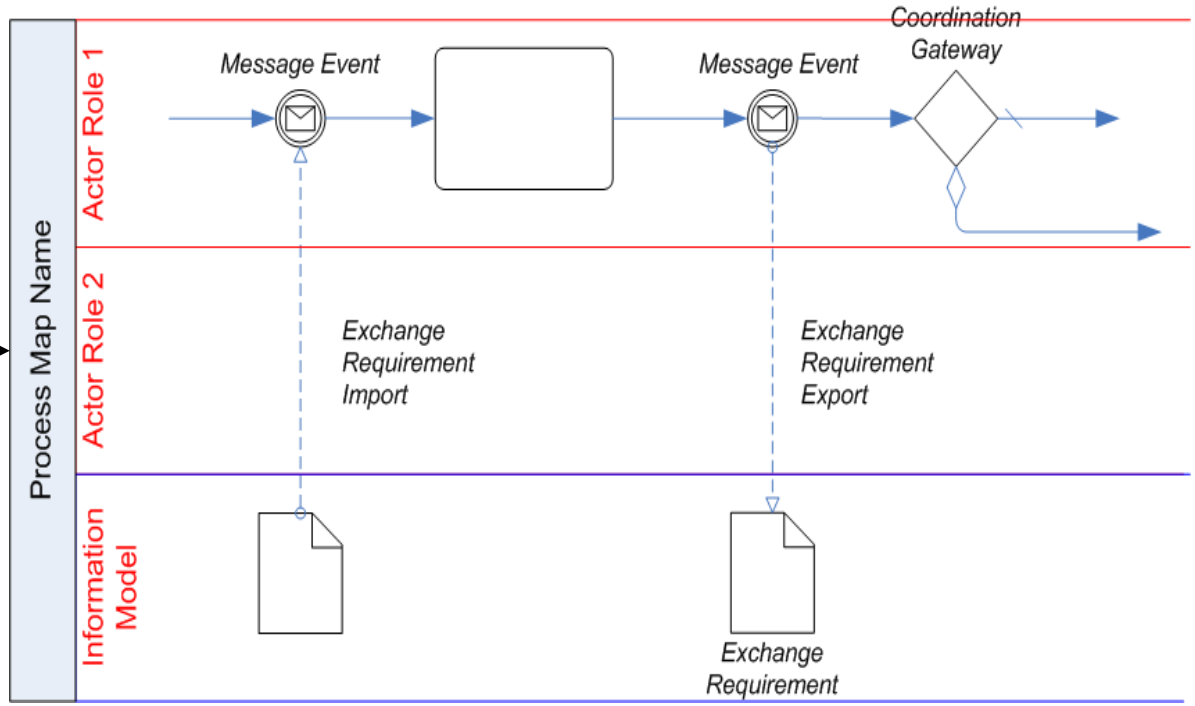
# IDM Specific Usages of BPMN

- IDM uses some techniques of the Business Process Modeling Notation (BPMN) in particular ways.
- These techniques apply some minor restrictions to the overall development.
- However, all of the techniques are legal uses of the notation.

# IDM Swimming Pool

Each process map is developed as a single ‘pool’.

*This is intended to identify each project as an overall organization and each actor participating as a part of that organization.*

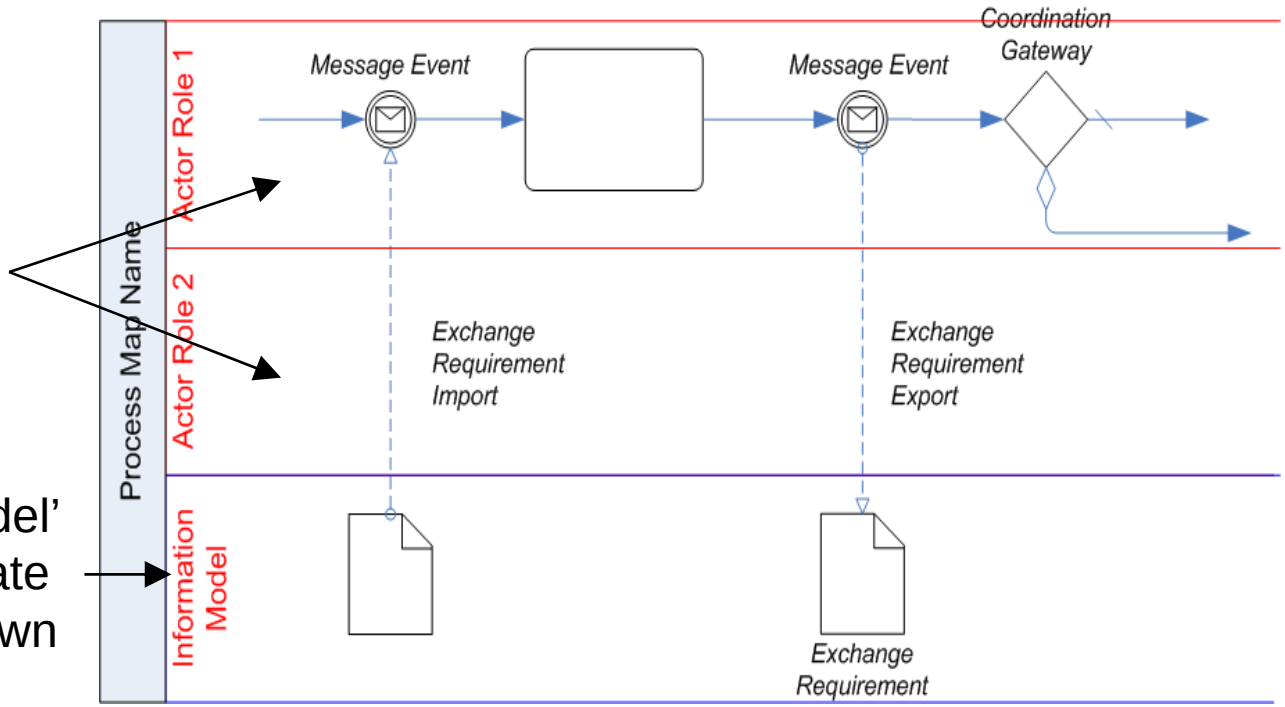



The name of the pool identifies the process map

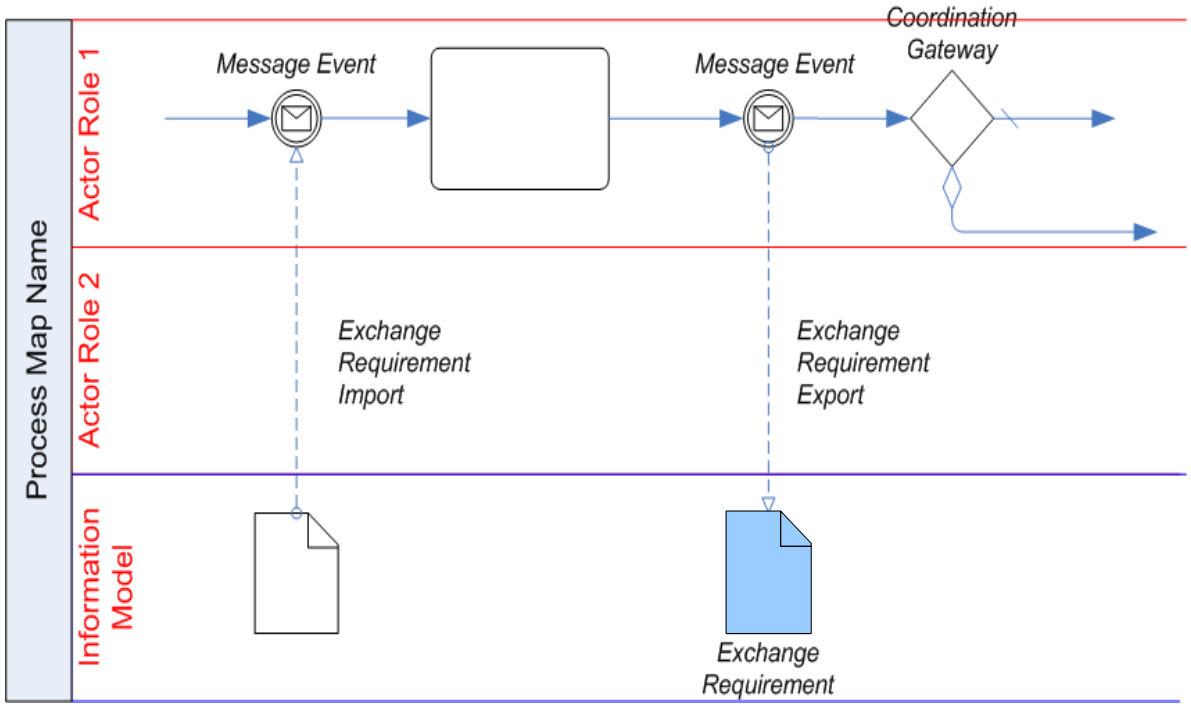
# IDM Swimlanes

Activities for each actor are shown in individual swimlanes

The ‘Information Model’ is shown as a separate actor role within its own swimlane.



# Showing an Exchange Requirement



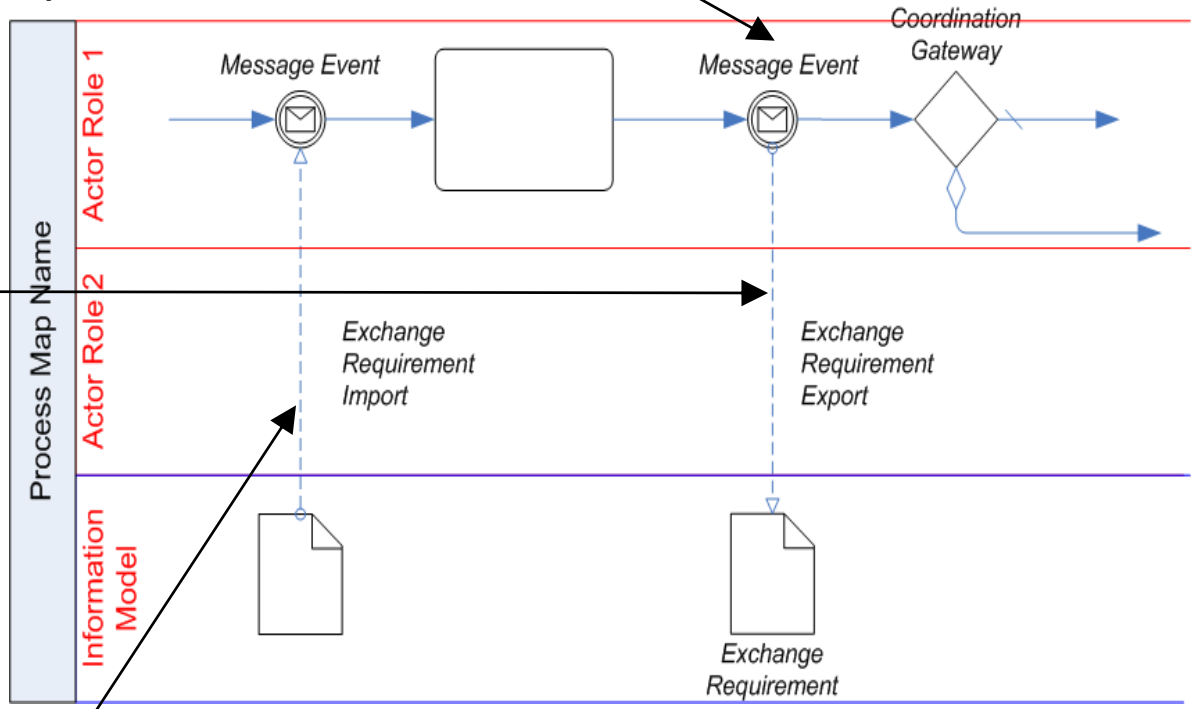

An ‘Exchange Requirement’ is always shown in a process map as a data object within the Information Model swimlane.

# Providing an Exchange Requirement

The provision of an ‘Exchange Requirement’ is always shown by:

A ‘message’ event within the ‘sequence flow’

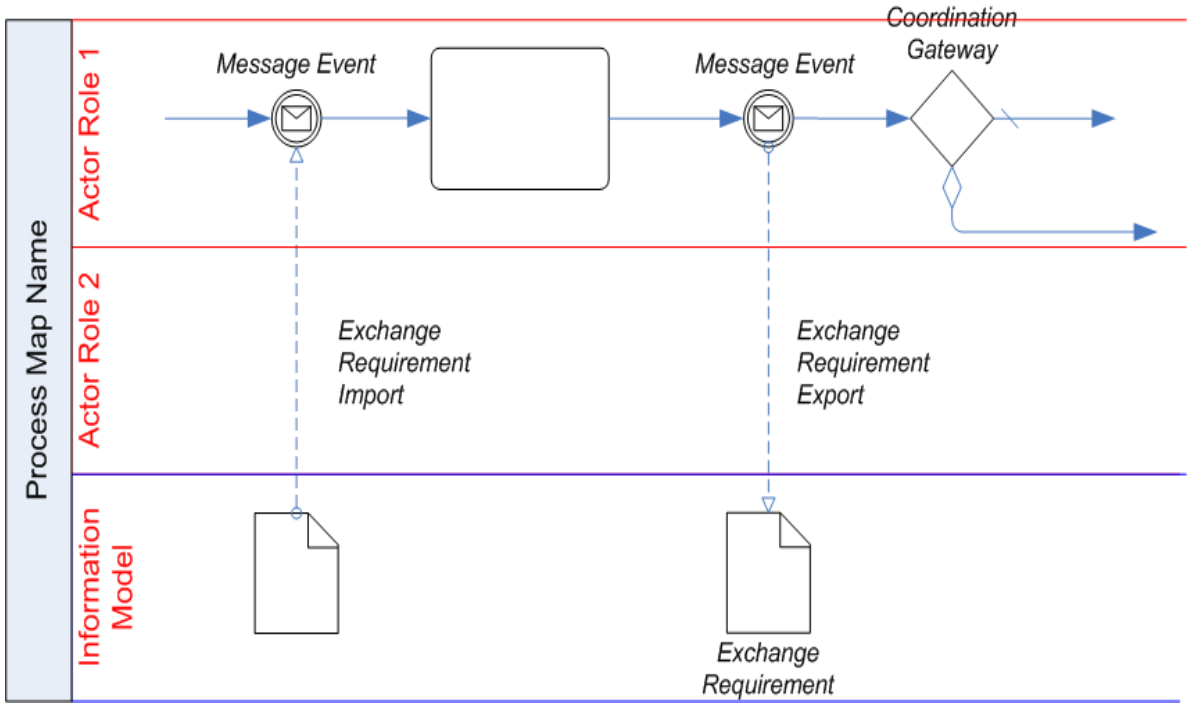
A message flow from the event to the Exchange Requirement data object



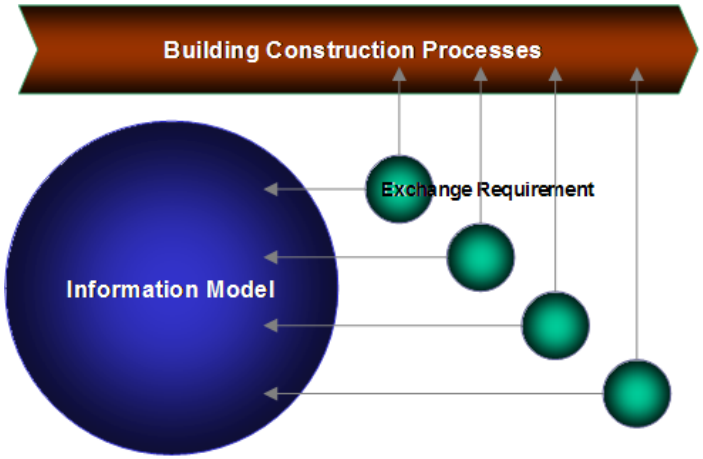

An Exchange Requirement used as input is shown in the same way.

# Coordination Gateway

An Exchange Requirement export event is succeeded immediately by a ‘Coordination Gateway.




*A Coordination Gateway is a point at which information from several exchange requirements may be brought together for examination, consideration or action*




- **Congratulations**
- You can now draw complete IDM process maps.