Business Process Modeling

Jaelson Castro

jbc@cin.ufpe.br

Objectives

- Business processes
- Modeling concurrency and synchronization in business activities
- BPMN Diagrams

Business Processes

- Business Process modeling involves capturing an ordered sequence of activities and supporting information [White and Miers 2008].
- Applications
  - Automation
  - Improvement
  - Re-engineering

Business Process – Applications

- Business Process Automation
  - Leave existing business processes as they are
  - Look for opportunities to automate parts of the process
  - Can make an organization more efficient; has least impact on the business
- Business Process Improvement
  - Make moderate changes to the way the organization operates
    - E.g. improve efficiency and/or effectiveness of existing process
  - Techniques:
    - Duration analysis;
    - Activity-based costing;
    - Benchmarking

Business Process – Applications

- Business Process Re-engineering
  - Fundamental change to the way the organization operates
  - Techniques:
    - Outcome analysis - focus on the real outcome from the customer's perspective
    - Technology analysis - look for opportunities to exploit new technology
  - Activity elimination - consider each activity in turn as a candidate for elimination

Modeling Business Processes

- Business processes involve:
  - Multiple actors (people, business units, ...)
  - Concurrent activities
  - Explicit synchronization points
    - E.g. some task cannot start until several other concurrent tasks are complete
    - End-to-end flow of activities

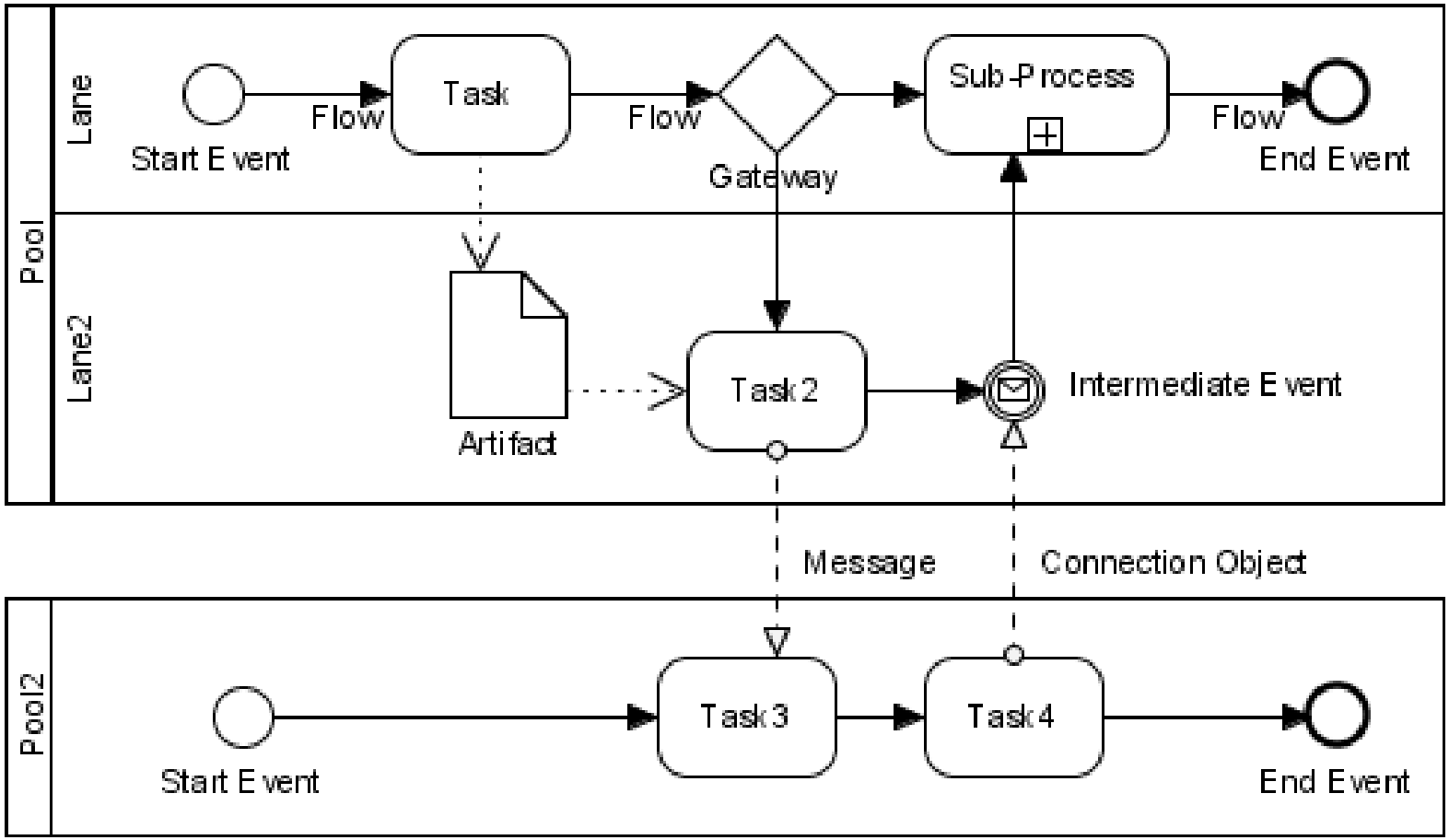
Business Process Model & Notation (BPMN)

- Standard released in 2004
- Adds many detailed modeling elements to basic activity diagrams
- Offers notation that is understandable by all business process users
  - Process analysts
  - IS developers
  - Process managers
- Fills gap between business models and their implementation

Definition of BPMN

- The Business Process Modeling Notation (BPMN) is a graphical notation that depicts the steps in a business process.
  - BPMN depicts the end to end flow of a business process.
  - Notation designed to coordinate the sequence of processes and the messages that flow between different process participants in a related set of activities
- BPMN is an agreement between multiple modeling tools vendors, who had their own notations, to use a single notation for the benefit of end-user understand and training

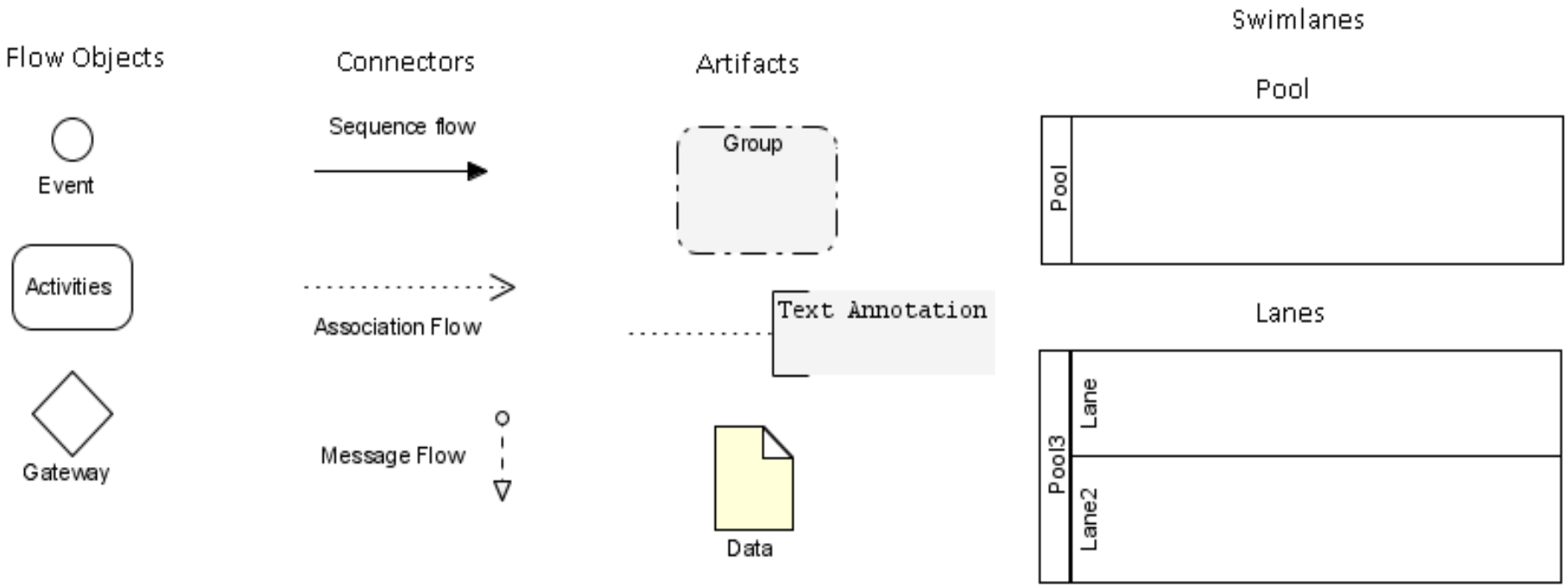
BPMN Model

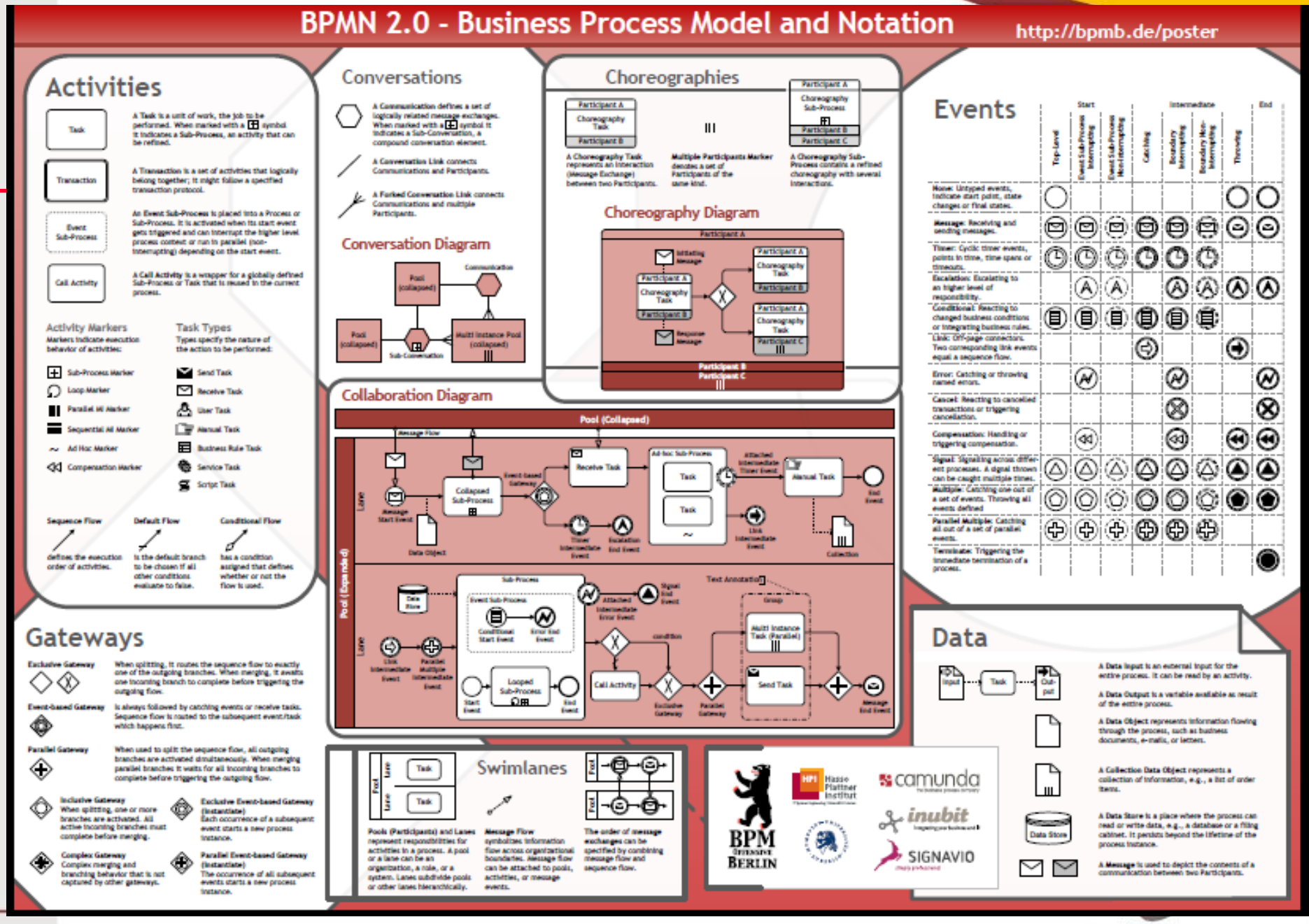

Pool
Lane
Start Event
Flow
Task
Flow
Gateway
Sub-Process
Flow
End Event
Lane2
Artifact
Task2
Intermediate Event
Message
Connection Object
Pool2
Start Event
Task3
Task4
End Event

Core Set of Diagram Elements

- The core set of modeling elements enable the easy development
  - Simple Business Process Diagrams that will look familiar to most Business Analysts (a flowchart diagram)

Diagram Elements



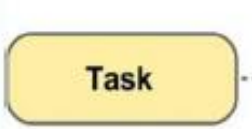

BPMN 2.0 - Business Process Model and Notation
http://bpmb.de/poster
Activities
Task
A Task is a unit of work, the job to be performed. When marked with a ⊞ symbol it indicates a Sub-Process, an activity that can be refined.
Transaction
A Transaction is a set of activities that logically belong together; it might follow a specified transaction protocol.
Event Sub-Process
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.
Call Activity
A Call Activity is a wrapper for a globally defined Sub-Process or Task that is reused in the current process.
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.
Conversations
A Communication defines a set of logically related message exchanges. When marked with a ⊞ symbol it indicates a Sub-Conversation, a compound conversation element.
A Conversation Link connects Communications and Participants.
A Forked Conversation Link connects Communications and multiple Participants.
Conversation Diagram
Communication
Pool (collapsed)
Sub-Conversation
Multi Instance Pool (collapsed)
Choreographies
Participant A
Choreography Task
Participant B
A Choreography Task represents an Interaction (Message Exchange) between two Participants.
Multiple Participants Marker denotes a set of Participants of the same kind.
Participant C
Choreography Sub-Process
A Choreography Sub-Process contains a refined choreography with several interactions.
Choreography Diagram
Initiating Message
Response Message
Collaboration Diagram
Pool (Collapsed)
Message Flow
Lane
Collapsed Sub-Process
Event-based Gateway
Receive Task
Ad-hoc Sub-Process
Task
Attached Intermediate Timer Event
Manual Task
End Event
Message Start Event
Data Object
Timer Intermediate Event
Escalation End Event
Link Intermediate Event
Collection
Pool (Expanded)
Sub-Process
Data Store
Event Sub-Process
Conditional Start Event
Error End Event
Attached Intermediate Error Event
Signal End Event
Text Annotation
Group
Multi Instance Task (Parallel)
Looped Sub-Process
Start Event
Call Activity
condition
Exclusive Gateway
Parallel Gateway
Send Task
Message End Event
Events
Start
Intermediate
End
Top-Level
Event Sub-Process Interrupting
Event Sub-Process Non-Interrupting
Catching
Boundary Interrupting
Boundary Non-Interrupting
Throwing
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 cancelation.
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.
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.
Exclusive Event-based Gateway (instantiate)
Each occurrence of a subsequent event starts a new process instance.
Complex Gateway
Complex merging and branching behavior that is not captured by other gateways.
Parallel Event-based Gateway (instantiate)
The occurrence of all subsequent events starts a new process instance.
Swimlanes
Pool
Lane
Task
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 order of message exchanges can be specified by combining message flow and sequence flow.
Data
Input
Task
Output
A Data Input is an external input for the entire process. It can be read by an activity.
A Data Output is a variable available as result of the entire process.
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.
Data Store
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.
A Message is used to depict the contents of a communication between two Participants.
BPM Offensive Berlin
Hasso Plattner Institut
camunda
inubit
SIGNAVIO

Activities

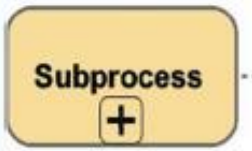
- Work that is performed within a BP
- Atomic or non-atomic (compound)
  - Sub-Process
  - Task
  - Transaction

Activity Symbol

An Activity is work that is performed within a Business Process. An Activity can be atomic or non-atomic (compound).



Single unit of work that is not or cannot be broken down to a further level of business process detail without diagramming the steps in a procedure (not the purpose of BPMN)



Hide or reveal additional levels of business process detail - when collapsed a sub-process is indicated by a plus sign against the bottom line of the rectangle; when expanded the rounded rectangle expands to show all flow objects, connecting objects, and artifacts.

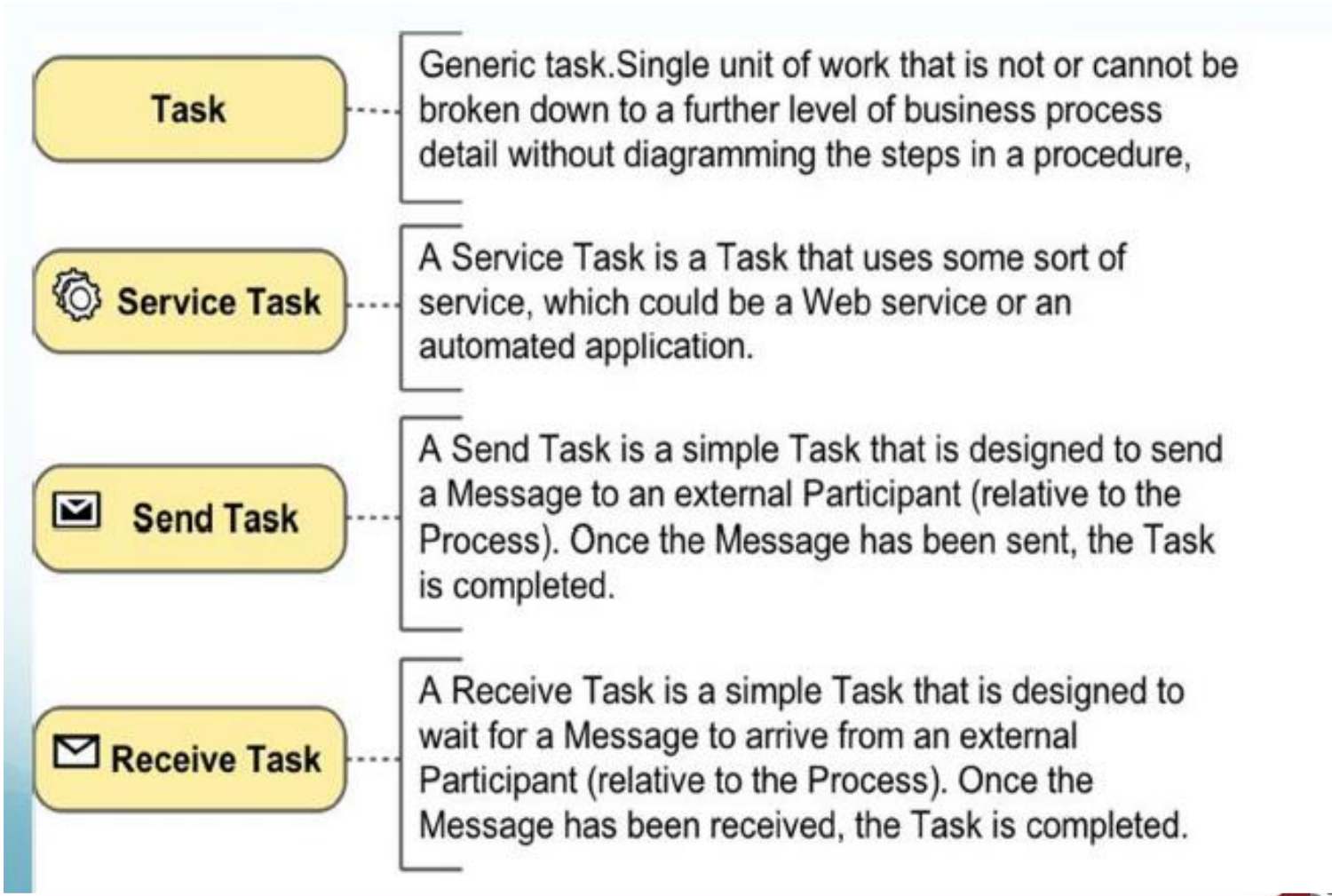
Has its own self-contained start and end events, and sequence flows from the parent process must not cross the boundary.

A form of sub-process in which all contained activities must be treated as a whole, i.e., they must all be completed to meet an objective, and if any one of them fails they must all be compensated (undone). Transactions are differentiated from expanded sub-processes by being surrounded by a tramline border.

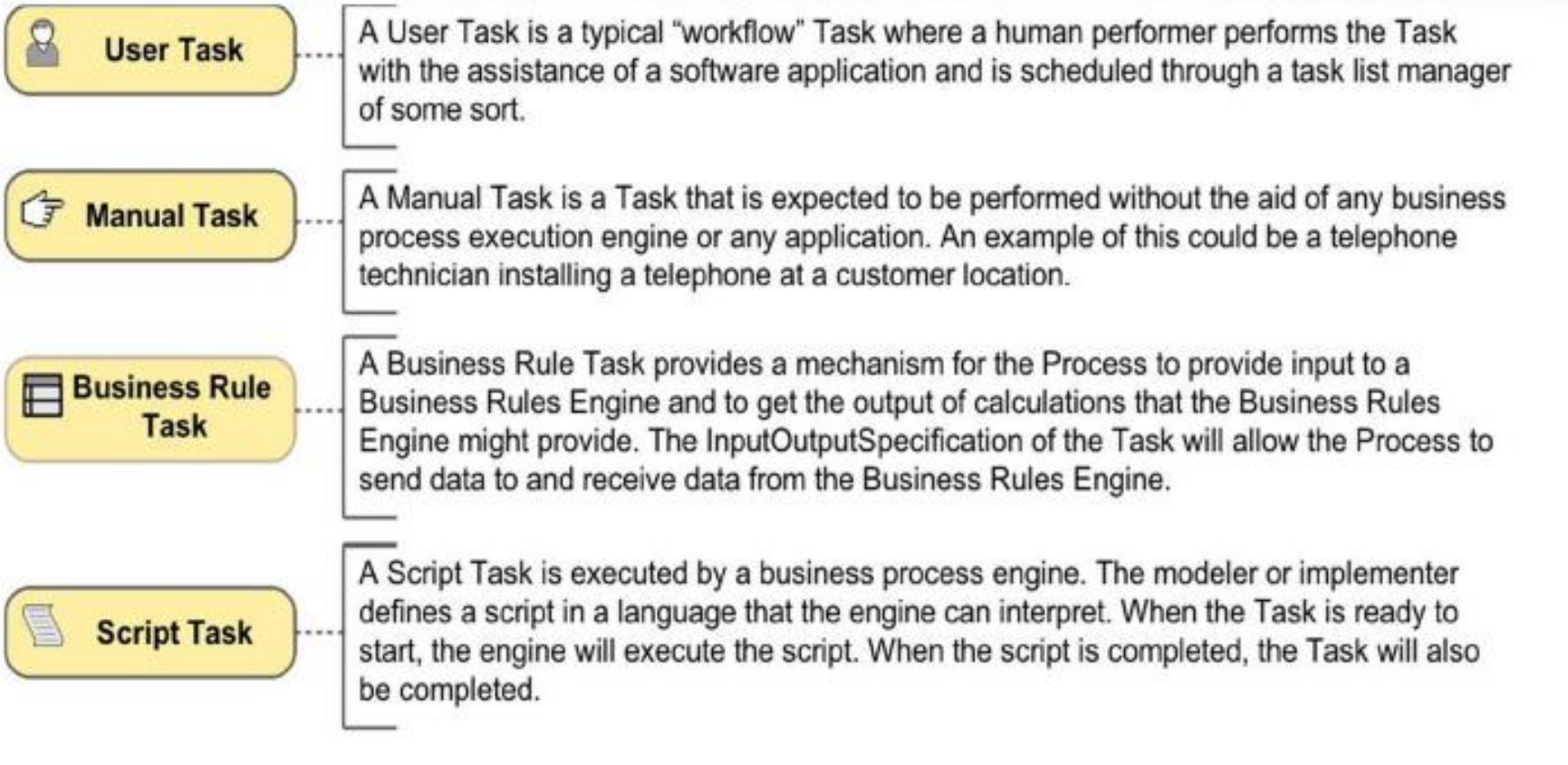
Tasks

- Definition
  - Task is used when the work in the Process is not broken down to a finer level of detail
- Atomic activity
- There are specialized types of Tasks
  - sending and receiving
  - user-based Tasks, etc
- Markers or icons to help identify the type of Task

Task Symbols (1 of 2)

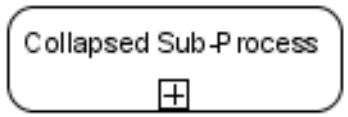


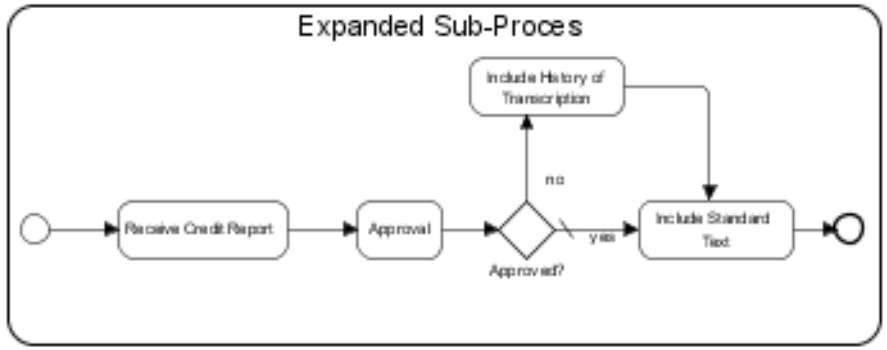
Task Symbols (2 of 2)



Sub-Processes

- Enable Hierarchical Process development
- Compound activity
- Broken down into a finer level of detail
- Collapsed version
  - Details are not visible in the Diagram
  - A plus sign in the lower-center
- Expanded version
  - Details are visible



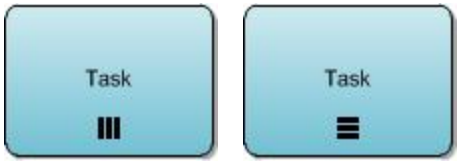


BPMN Task Activity Markers

Task: Smallest piece of work; Activity

Looping task: (Do while):

- Perform activity at least once
- Evaluate / test for condition
- If condition is true, repeat task again
- Condition may be a Boolean Expression or a Numeric Cap
- Condition may be evaluated at beginning or end of task



Multiple instance task: (For each):

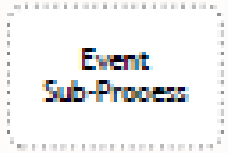
- Perform activity X times, where X is known at onset of task starting
- Task is complete when all X iterations are done
  Next task does NOT begin until all iterations are done
- Instances of activity can occur in sequence or in parallel with each other
- Each instance must complete the task before next task begins

Compensation

- Undoes tasks that were already successfully completed
- Outside normal process flow
- Marker shows used for compensation only

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 c Sub-Process. It is activated when its start event gets triggered and can interrupt the higher leve process context or run in parallel (non-interrupting) depending on the start event.

A Call Activity is a wrapper for a globally defin Sub-Process or Task that is reused in the curren process.

Activity Markers

Markers indicate execution behavior of activities:

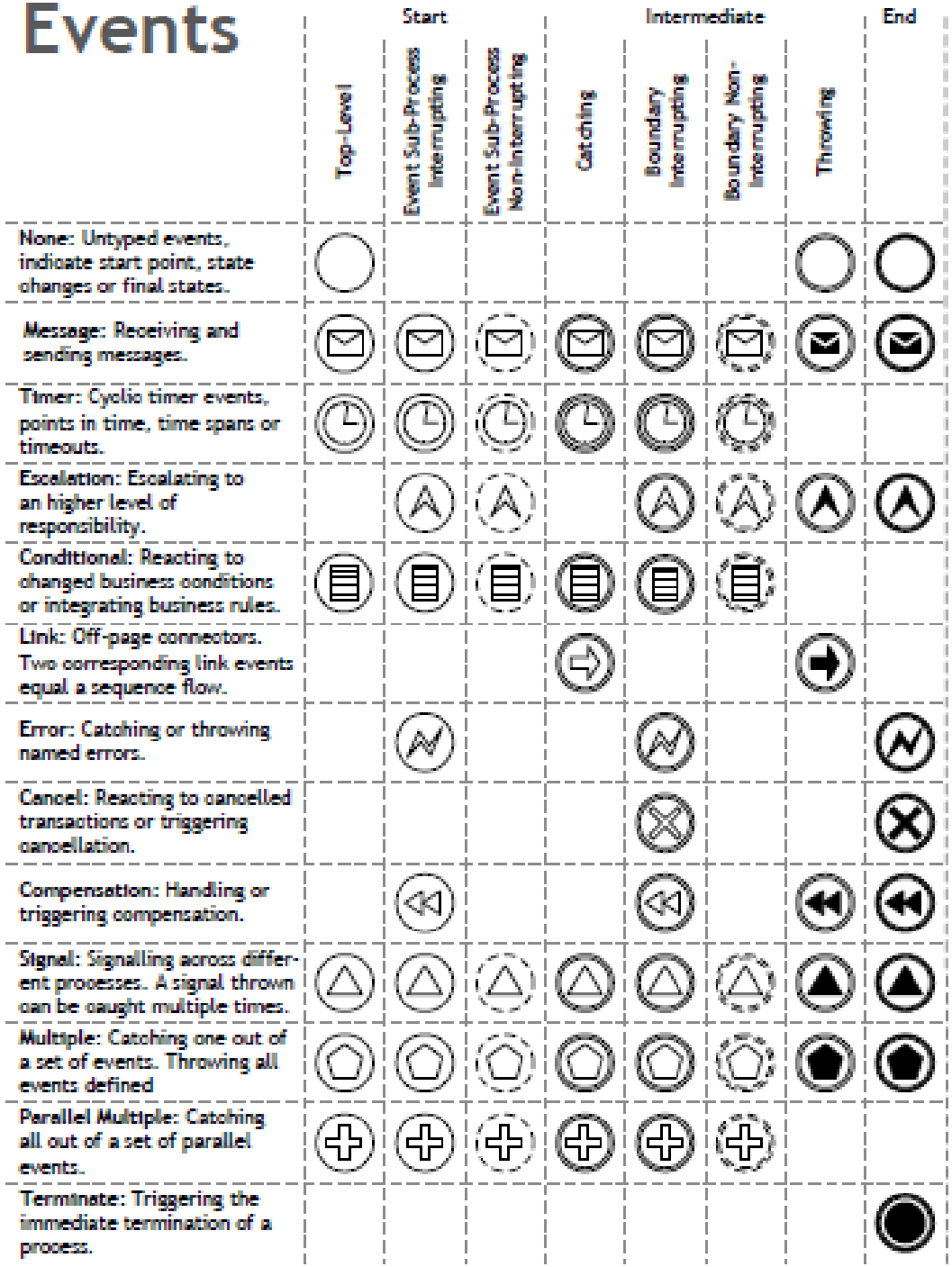
- Sub-Process Marker
- Loop Marker
- Parallel MI Marker
- Sequential MI Marker
- Ad Hoc Marker
- Compensation Marker

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

Events



Events

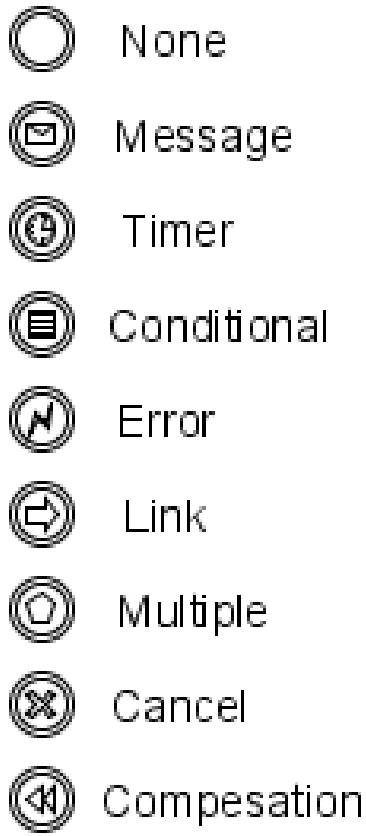
- An Event is something that happens during the course of a business process.
  - Events affect the flow of the Process and usually have a trigger or a result.
- They can start, interrupt, or end the flow
- Events are circles
- The type of boundary determines the type of Event

Start Event

- Start Events indicate where a Process will begin
- Different Triggers to specific circumstances
- None Start Event to start Sub-processes
- Any Event can trigger a Multiple Start Event
- Link Start Event will be removed in BPMN 2.0

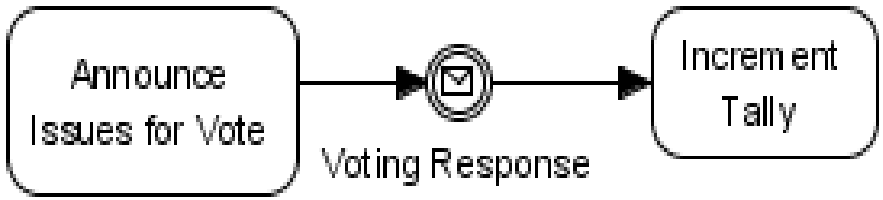
Intermediate Event

- Intermediate Events occur after a process has been started and before a process is ended
- Can be placed in the normal flow or attached to the boundary



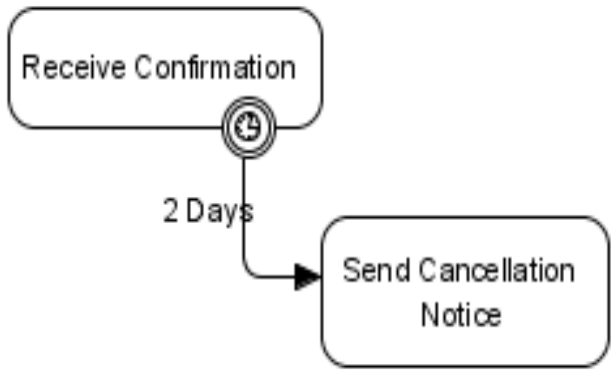
Intermediate Event (normal flow)

- Events that are placed within the process flow represent things that happen during the normal operations of the process
- They can represent:
  - the response to the Event (i.e., the receipt of a message)
  - the creation of the Event (i.e., the sending of a message)



Intermediate Event (Attached to Boundary)

- Events that are attached to the boundary of an activity indicate that the activity should be interrupted when the Event is triggered
- Attached to Tasks of Sub-Processes
- Used for error handling, exception handling, and compensation



End Event

- End Event indicates where a Process will end
- In terms of Sequence Flows, the End Event ends the flow of the Process, and thus, will not have any outgoing Sequence Flows
- No Sequence Flow can connect from an End Event

End Event

- End Events indicates where a process will end
- Different Results that indicate the specific circumstances that end the Process
- None End Events are used to mark that end of Sub-Processes

Gateways

- Gateways are used to control how Sequence flows interact as they converge or diverge within a Process
- Define decisions/branching (exclusive, inclusive, and complex), merging, forking, and joining.

Gateway Symbols

Gateways are used to control how Sequence Flows interact as they converge and diverge within a Process.

Exclusive

A diverging Exclusive Gateway (Decision) is used to create alternative paths within a Process flow. This is basically the "diversion point in the road" for a Process. For a given instance of the Process, only one of the paths can be taken.

Inclusive

Inclusive Gateway (Inclusive Decision) can be used to create alternative but also parallel paths within a Process flow. The true evaluation of one condition Expression does not exclude the evaluation of other condition Expressions. All Sequence Flows with a true evaluation will be traversed by a token.

Complex

The Complex Gateway can be used to model complex synchronization behavior. An Expression activationCondition is used to describe the precise behavior.

Event Based

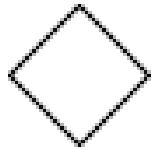
The Event-Based Gateway represents a branching point in the Process where the alternative paths that follow the Gateway are based on Events that occur, rather than the evaluation of Expressions using Process data (as with an Exclusive or Inclusive Gateway). A specific Event, usually the receipt of a Message, determines the path that will be taken.

Parallel

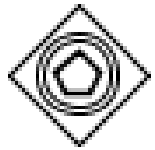
A Parallel Gateway is used to synchronize (combine) parallel flows and to create parallel flows.

Gateways

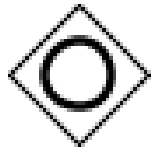
- Controls how Sequence Flows interact
- All types of Gateways are diamonds
- Different internal markers indicate different types of behavior
- Used to split and merge the flow
- If the flow does not need to be controlled, then a Gateway is not needed



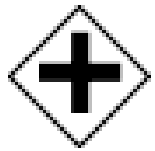
Exclusive Data-based



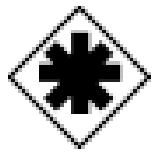
Exclusive Event-based



Inclusive



Parallel



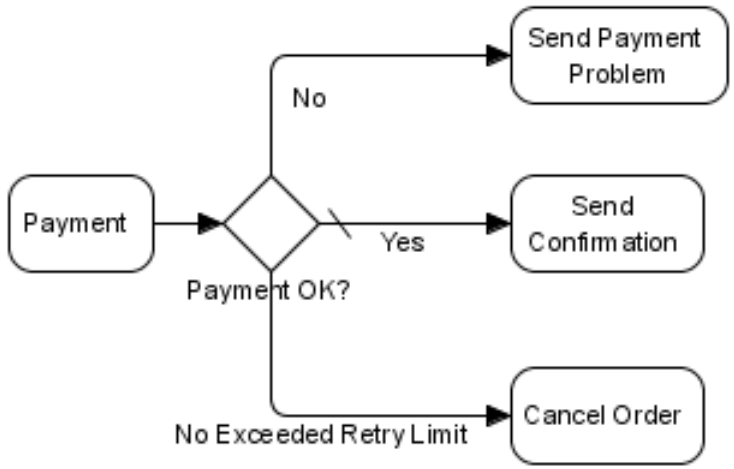
Complex

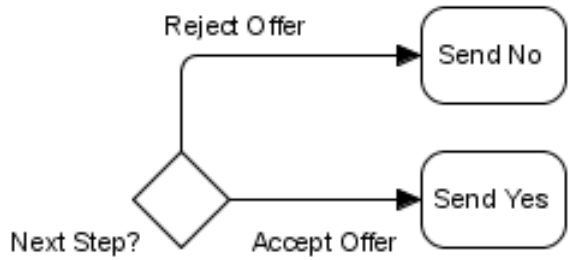
Exclusive Gateways

- They are locations within a business process where the Sequence Flow can take two or more alternative paths. This is basically the fork in the process.
- Only one of the possible outgoing paths can be taken when the Process is performed
- There are two types decision mechanism:
  - Data (e.g., condition expressions)
  - Events (e.g., the receipt of alternative messages)

Exclusive Gateways, based on Data

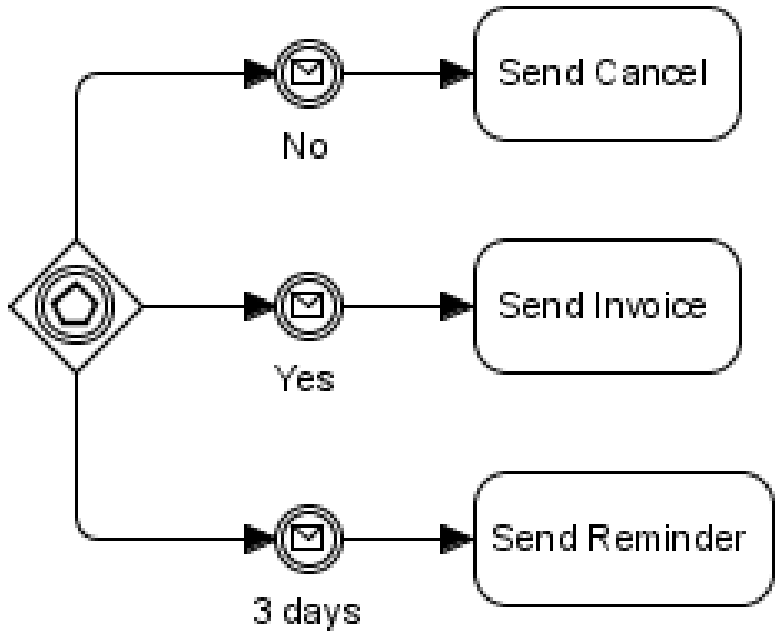
- Most commonly used Gateway
- With or without an internal marker
- Creates alternatives path based on defined conditions (Decision)





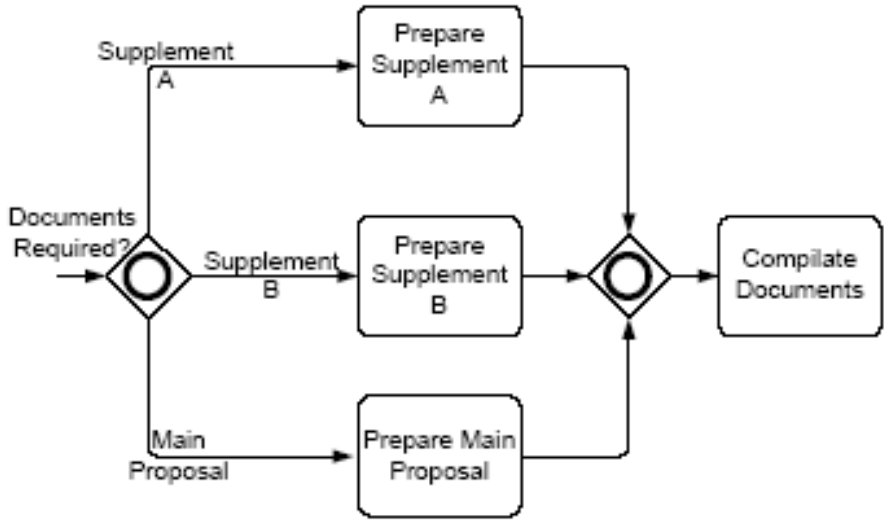
Exclusive Gateways, based on Events

- Branching point in the process where the alternatives are based on events
- The Multiple Intermediate Event is used to identify this Gateway
- The Event that follow the Gateway Diamond determine the chosen path
- First Event triggered wins



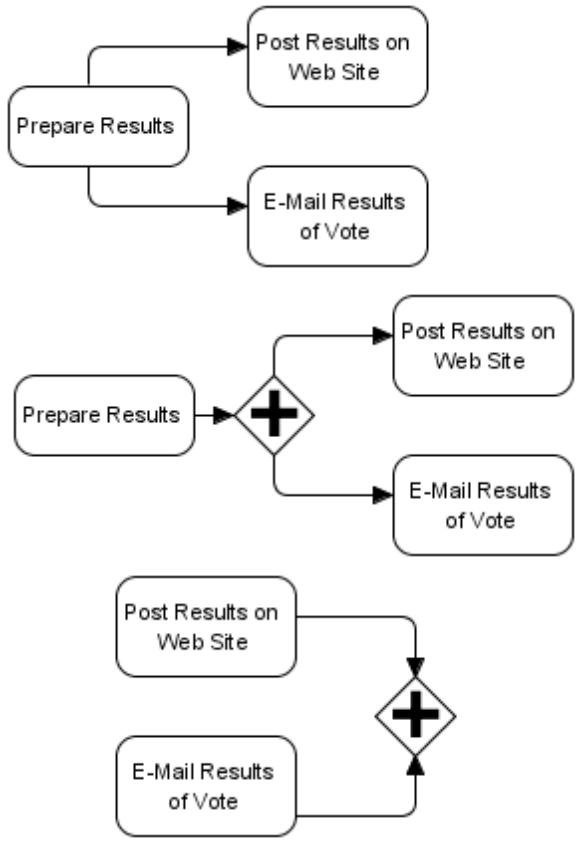
Inclusive Gateways

- Inclusive Gateways are Decisions where there is more than one possible outcome
- The “O” marker is used to identify this Gateway
- They are usually followed by a corresponding merging Inclusive Gateway



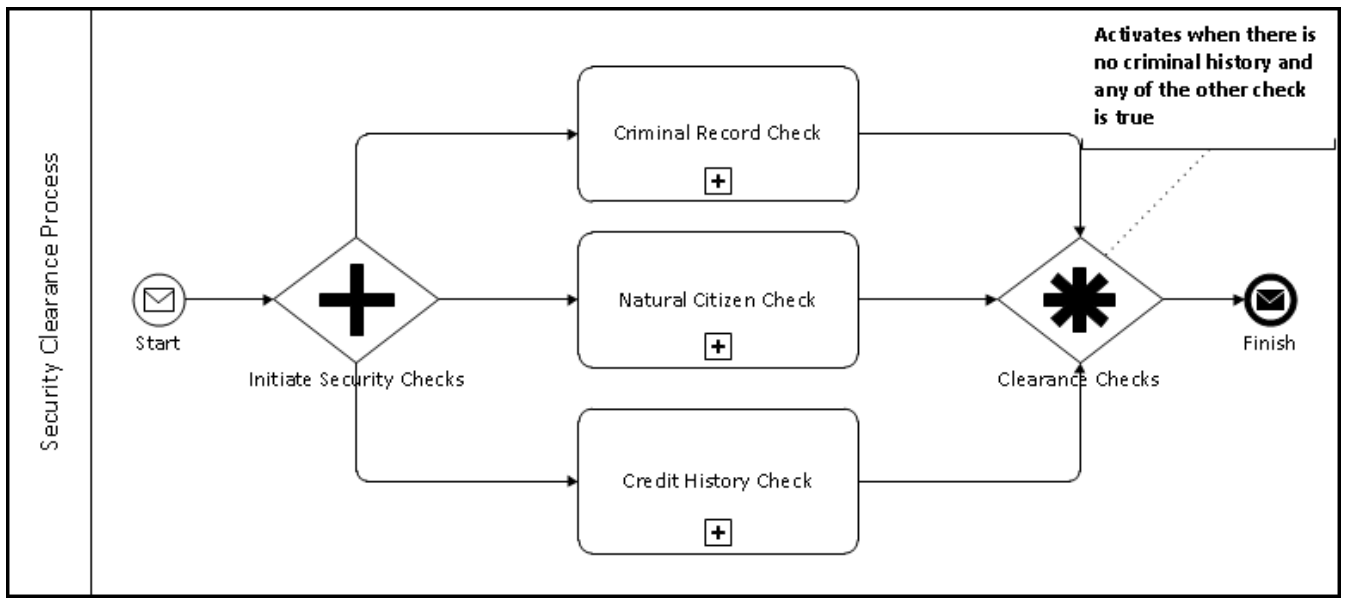
Parallel Gateways

- They are where multiple parallel paths are defined
- They are required for forking in most situations
- They can be used for methodological purposes
- The Gateway is also used to synchronize (wait for) parallel paths



Complex Gateway

- Complex Gateways are Decisions where more advanced definitions of behavior can be defined
- The asterisk marker is used to identify this Gateway



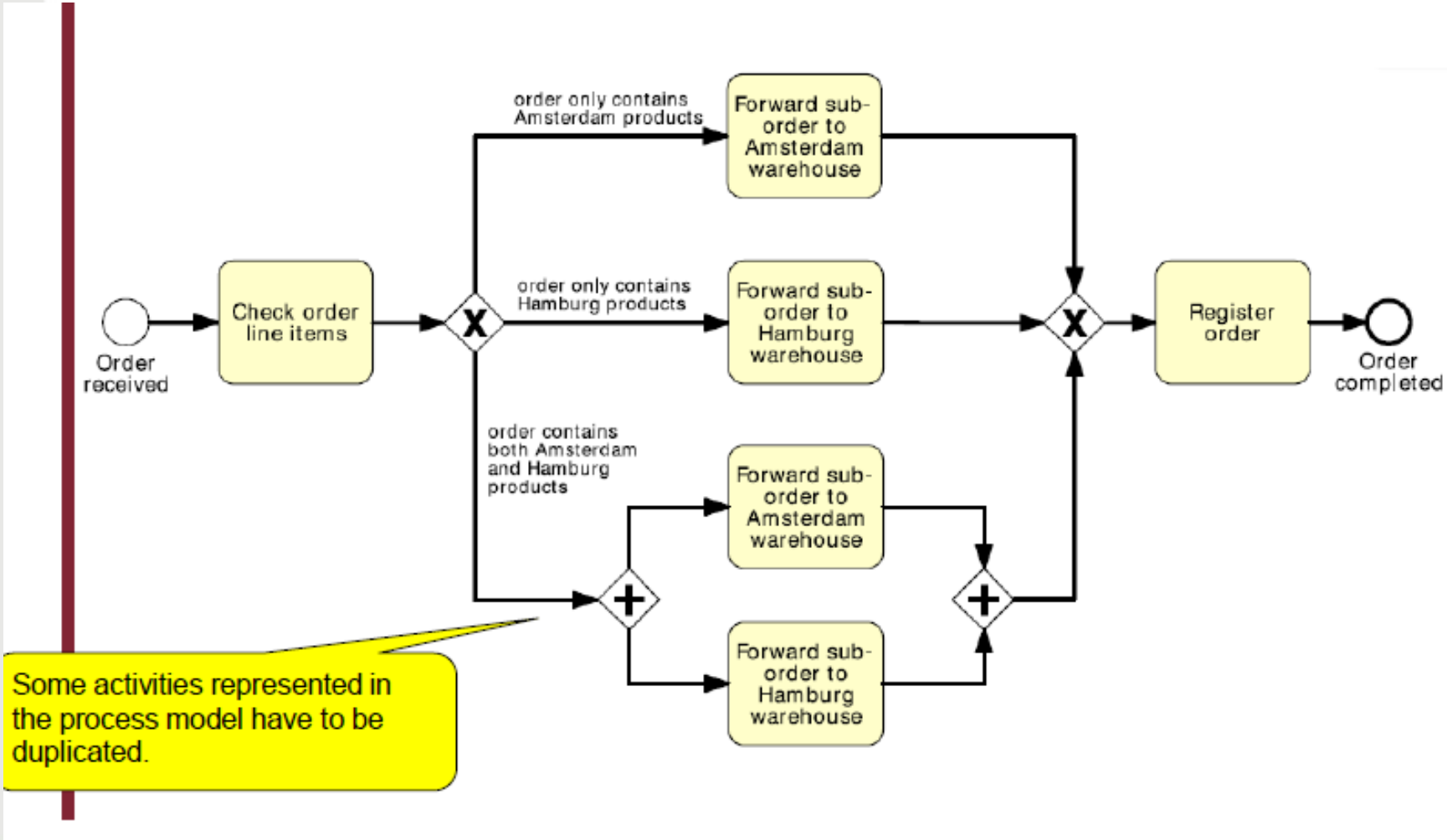
Exercise

A company has two warehouses that store different products: Amsterdam and Hamburg.

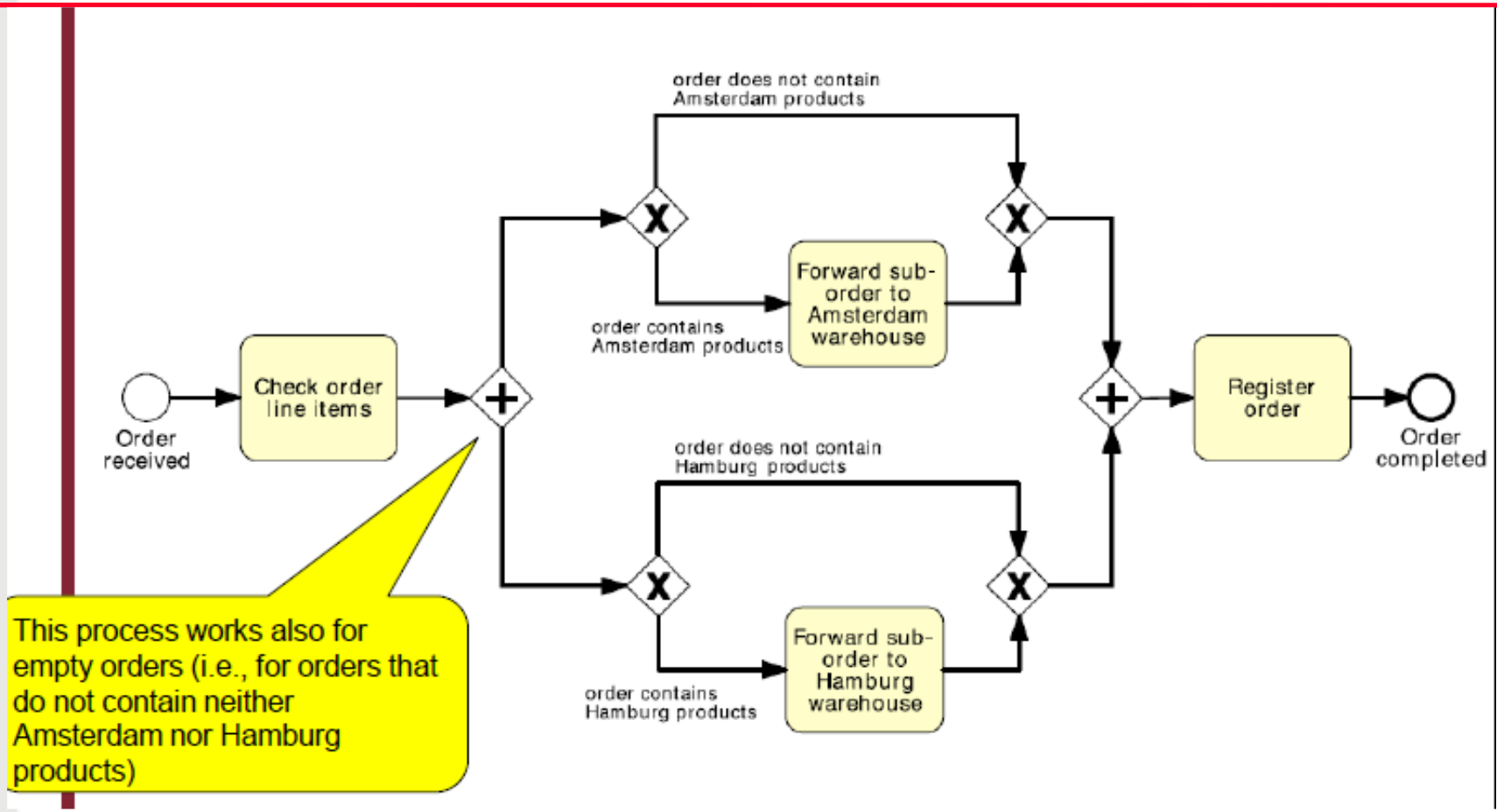
When an order is received, it is distributed across these warehouses: if some of the relevant products are maintained in Amsterdam, a sub-order is sent there; likewise, if some relevant products are maintained in Hamburg, a sub-order is sent there.

Afterwards, the order is registered and the process completes.

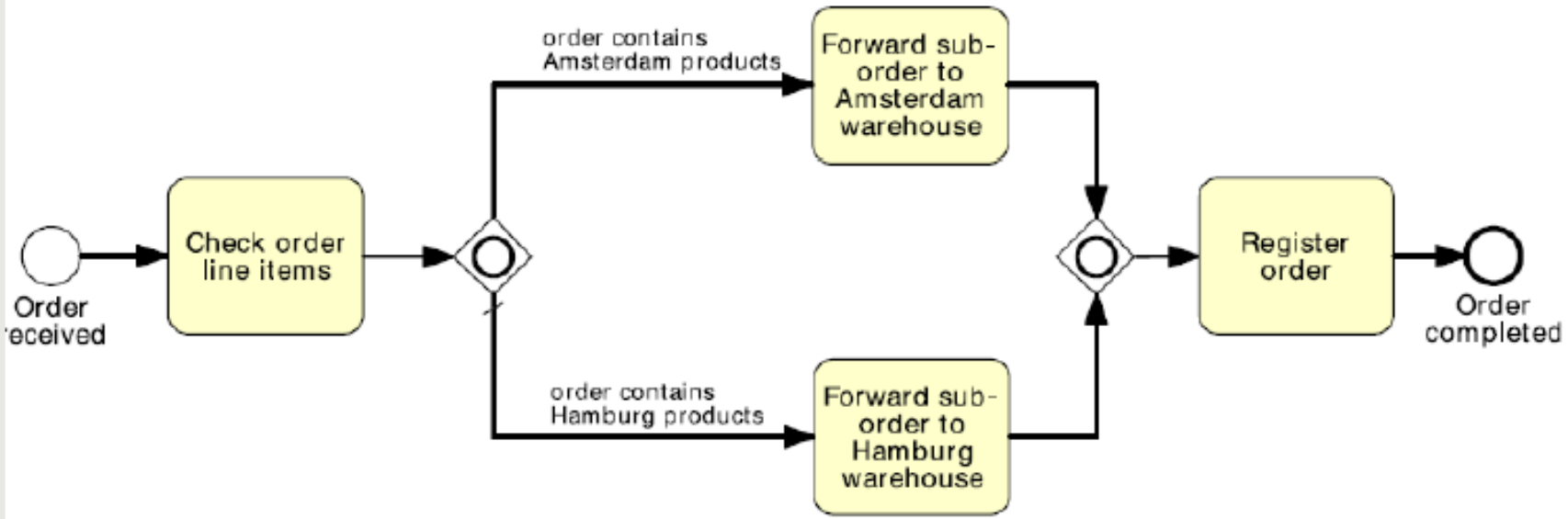
First Solution



Second Solution

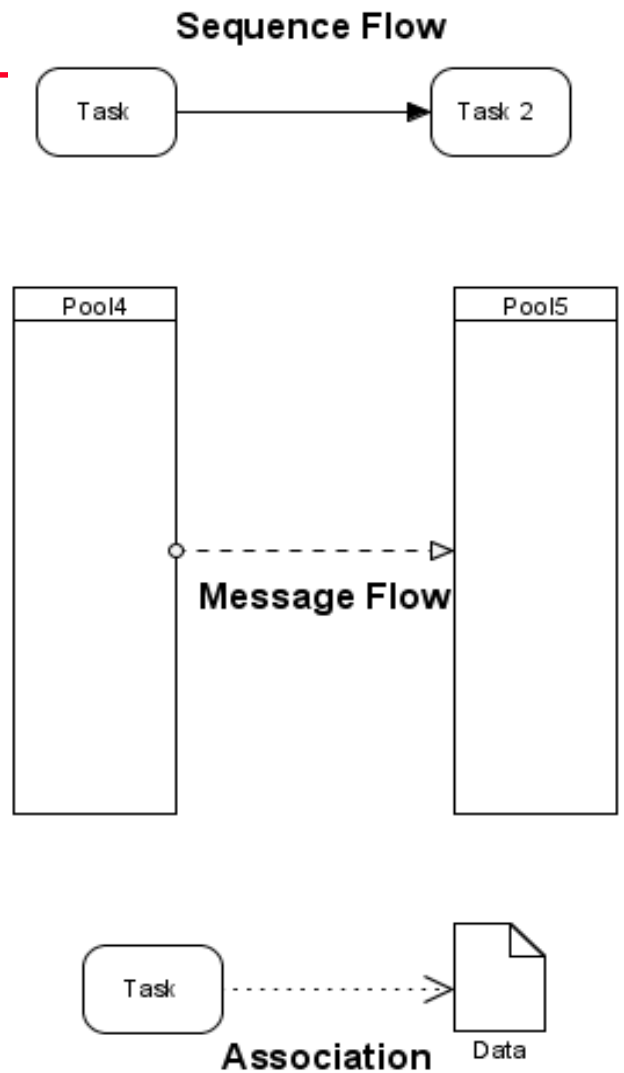


Third Solution with Inclusive Gateways



Connectors

- A Sequence Flow is used to show the order that activities will be performed in a Process
- A Message Flow is used to show the flow of messages between two entities that are prepared to send and receive them
- An Association is used to associate information and artifacts with flow objects

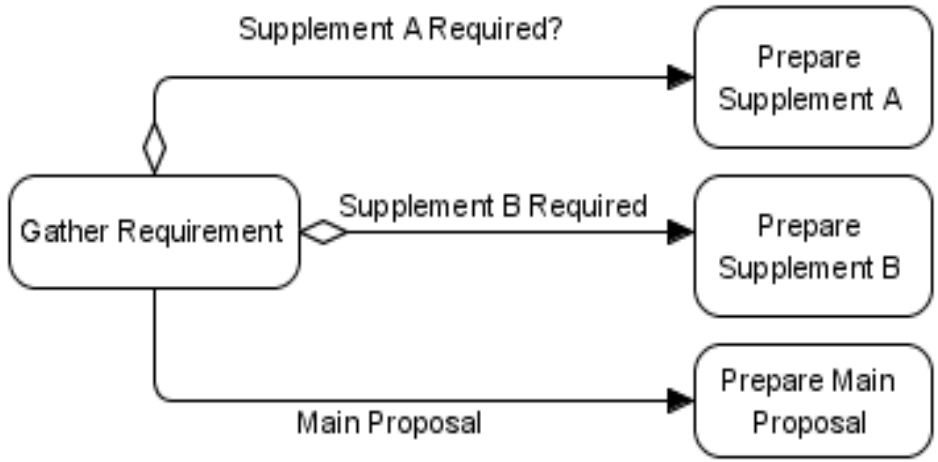


Sequence Flow

- A Sequence Flow is used to show the order that activities will be performed in a Process.
- The source and target must be one of the following objects:
  - Events, Activities, and Gateways
- A Sequence Flow cannot cross a Sub-Process boundary or a Pool boundary

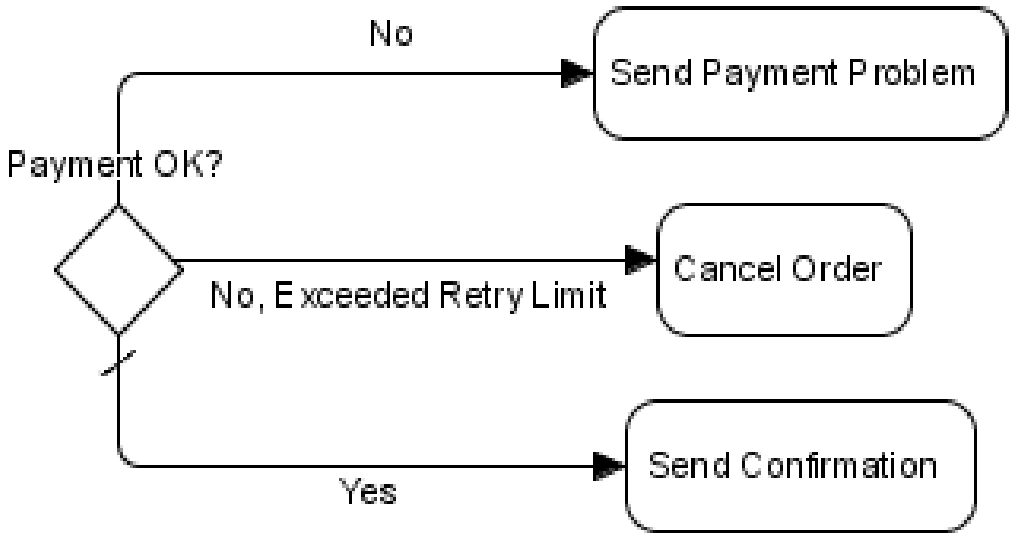
Conditional Sequence Flow

- Sequence Flow may have a condition
- At least one of the outgoing Sequence Flow must be chosen
- The condition has to be True to allow the flow to continue
- A mini-diamond shows that the flow has a condition

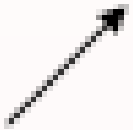
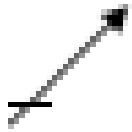
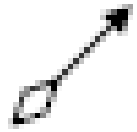


Default Sequence Flow

- Sequence Flow may be defined as being the default path
- A hatch mark at the line beginning shows the default Sequence Flow
- A default path is chosen only if all the other conditions are False

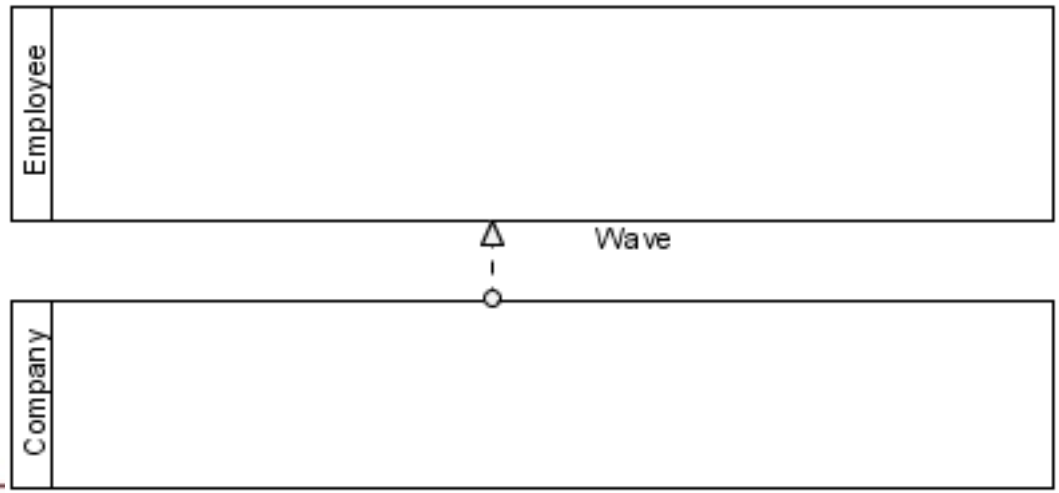


Flows

Sequence Flow	Default Flow	Conditional Flow
		
defines the execution order of activities.	is the default branch to be chosen if all other conditions evaluate to false.	has a condition assigned that defines whether or not the flow is used.

Message Flow

- Shows the flow of messages between two Participants of Process (pools)
- A Message Flow can connect to the boundary of the Pool or to an object within the Pool
- Message Flows are not allowed between objects within a single Pool



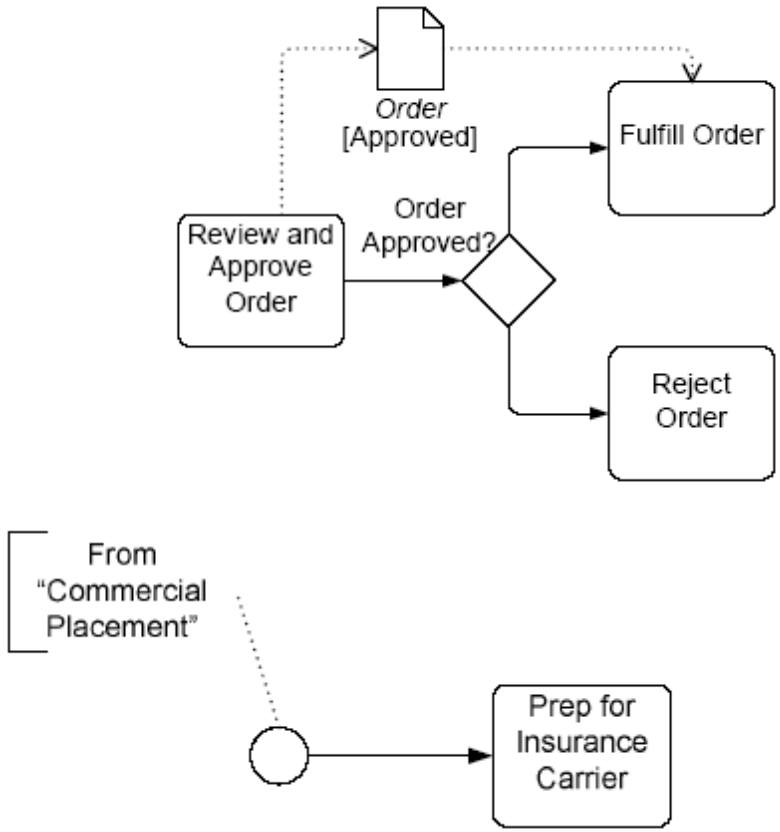
Associations

- They Associate objects to one another
- Show how data is input to and output from Activities

- Text Annotations can be Associated with objects

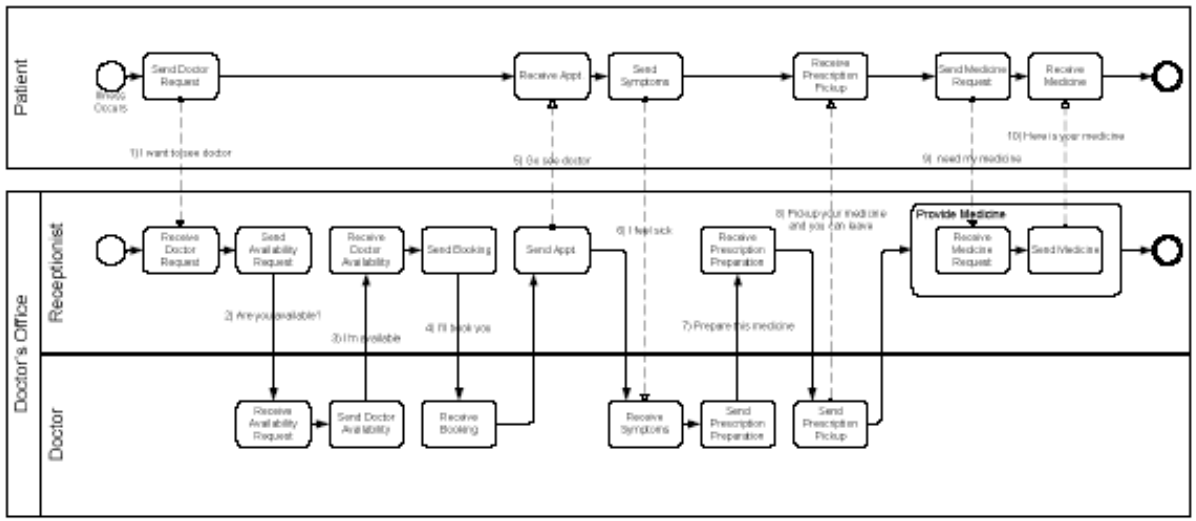
Associations

- An Association is used to associate objects to one another (such as Artifacts and Activities)
- Associations are used to show how data is input to and output from Activities
- Text Annotations can be Associated with objects



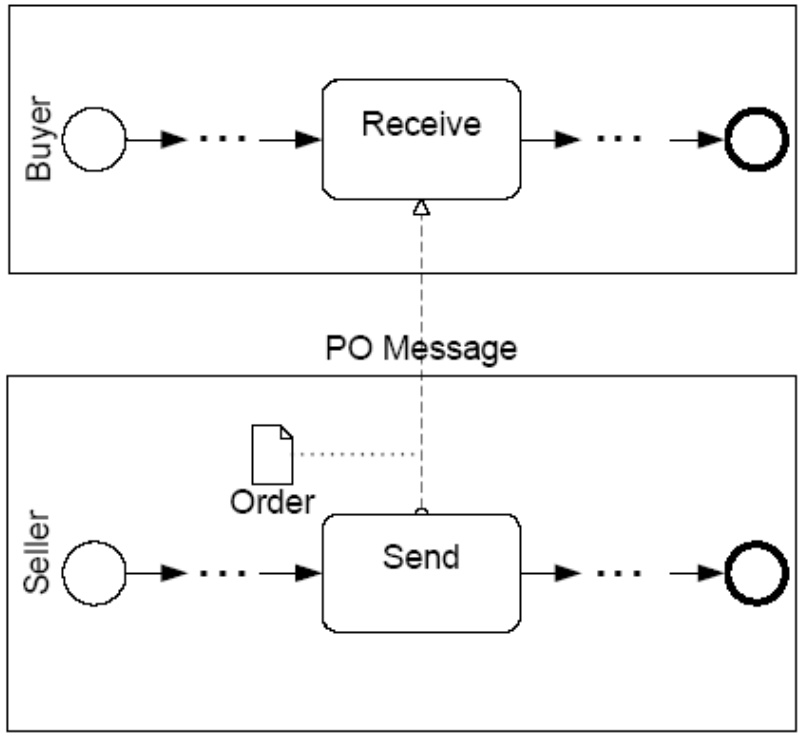
Swimlanes

- BPMN uses the concept known as “swimlanes” to help partition and/organize activities
- There are two main types of swimlanes: Pool and Lane
  - Pools represent Participants in an interactive (B2B) Business Process Diagram
  - Lanes represent sub-partitions for the objects within a Pool



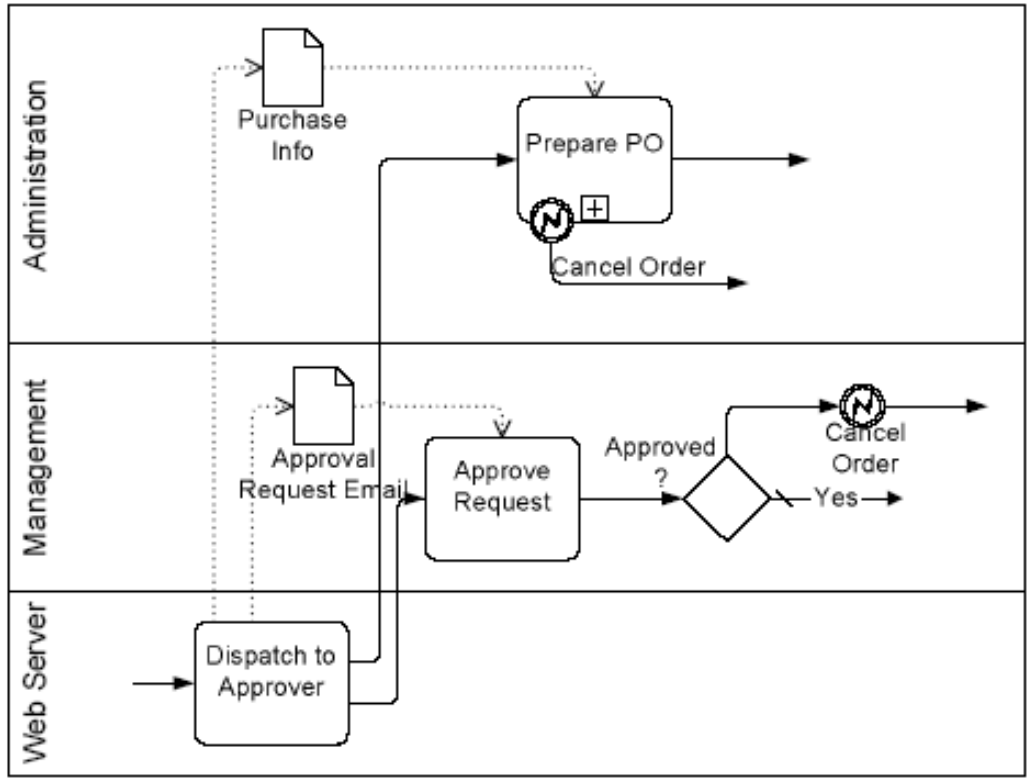
Pool

- Pools represent Participants in an interactive (B2B) Business Process Diagram
  - A Participant may be a business role (e.g., “buyer” or “seller”) or may a business entity (e.g., “IBM” or “OMG”)
- A Pool may be a “black box” or may contain a Process
- Interaction between Pools is handled through Message Flow
- Sequence Flow cannot cross the boundary of a Pool (i.e., a Process is fully contained within a Pool)

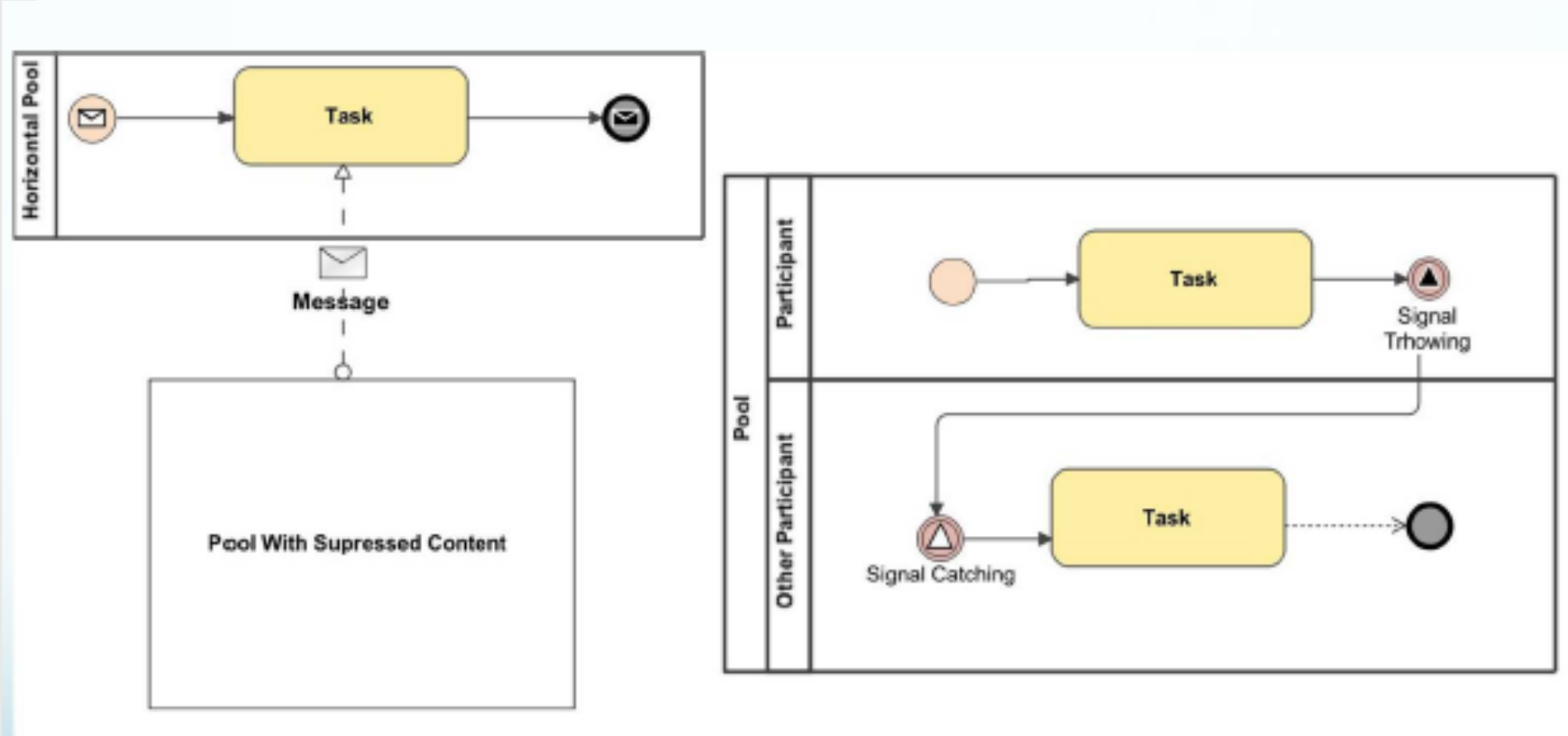


Lanes

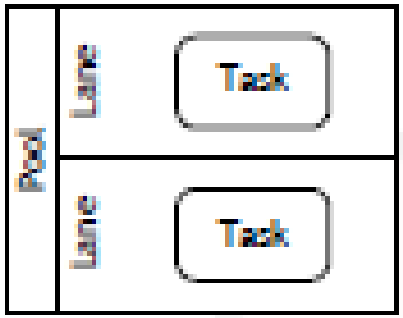
- Lanes represent sub-partitions for the objects within a Pool
- They often represent organization roles (e.g., Manager, Associate), but can represent any desired Process characteristic
- Sequence Flow can cross Lane boundaries



Pools & Lanes


Horizontal Pool
Task
Message
Pool With Supressed Content
Pool
Participant
Task
Signal Trhowing
Other Participant
Signal Catching
Task

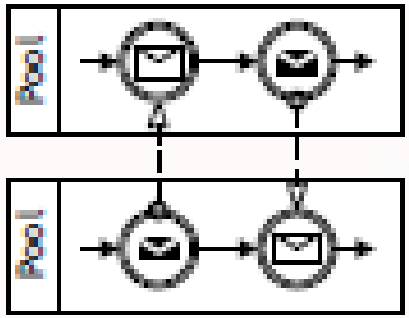
Pools & Lanes



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.

Swimlanes

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



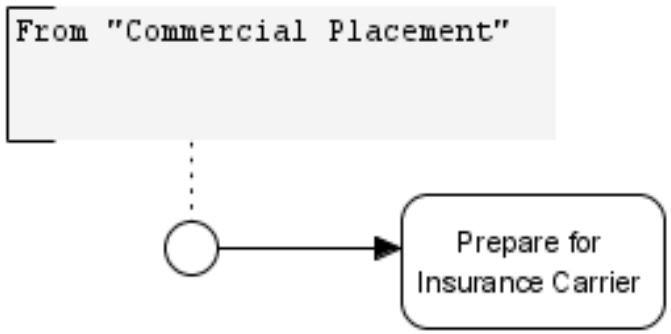
The order of message exchanges can be specified by combining message flow and sequence flow.

Artifacts

- They provide the capability to show information beyond the basic flow-chart structure of the Process
- Three standard Artifacts:
  - Text Annotations
  - Data Objects
  - Groups
- A modeler or tool can extend BPMN by defining new Artifacts

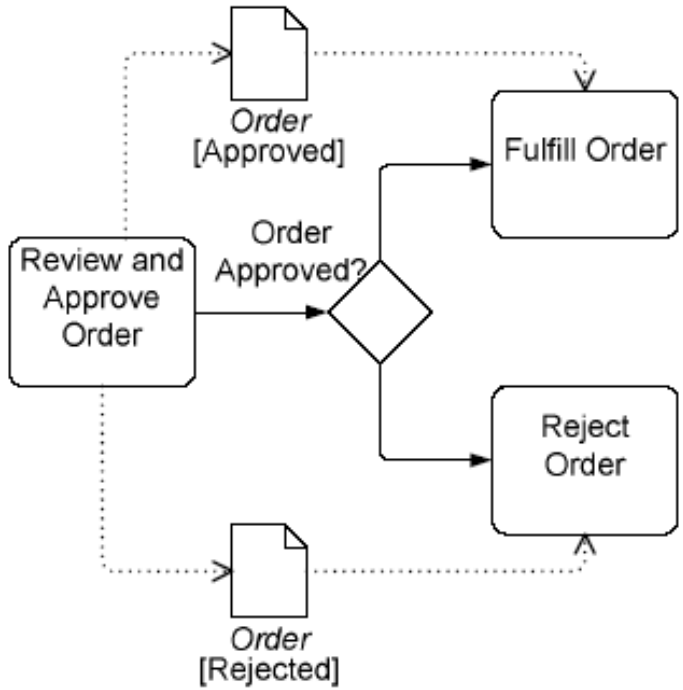
Text Annotations

- Mechanism for a modeler to provide additional information
- Can be connected to a specific object on the Diagram with an Association



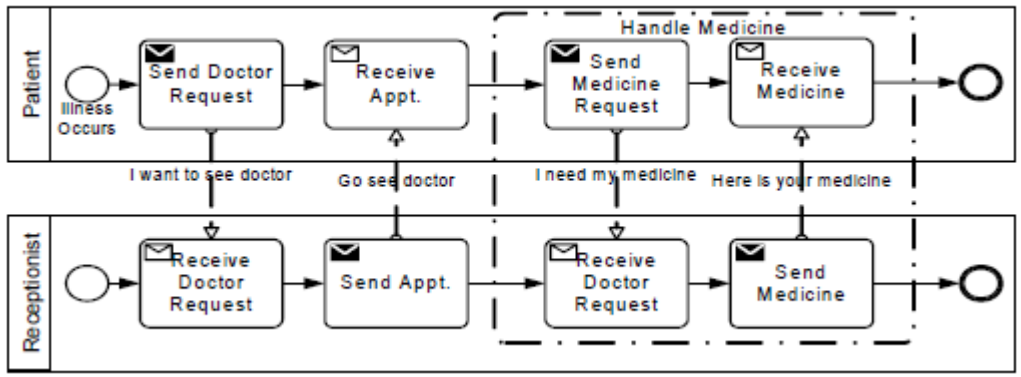
Data Objects

- Data Objects are Artifacts that are used to show how data and documents are used within a Process
- Data Objects can be used to define inputs and outputs of activities
- Data Objects can be given a “state” that shows how a document may be changed or updated within the Process



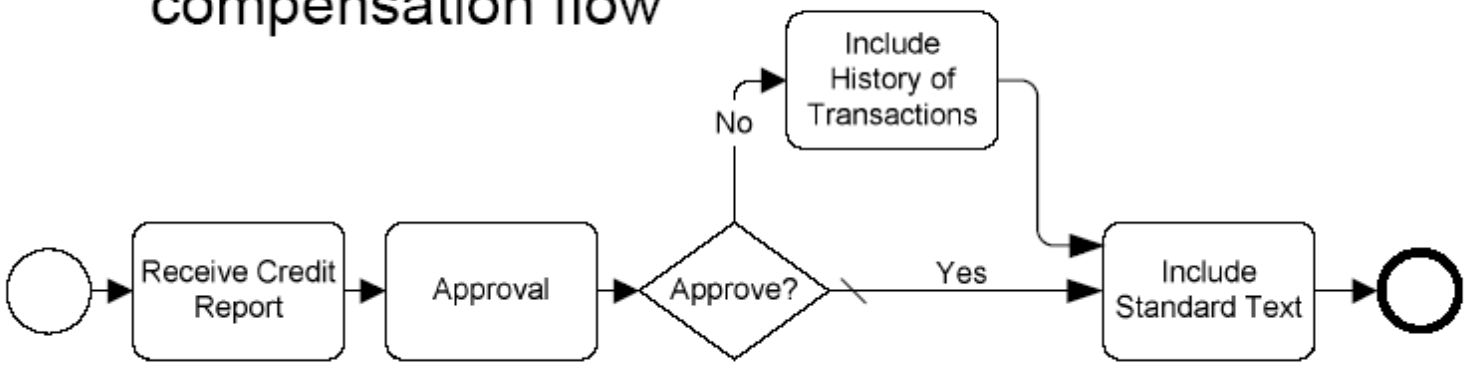
Groups

- Highlight certain sections of a Diagram without adding additional constraints for performance
- Categorize elements for reporting purposes
- Not constrained by restrictions of Pools and Lanes



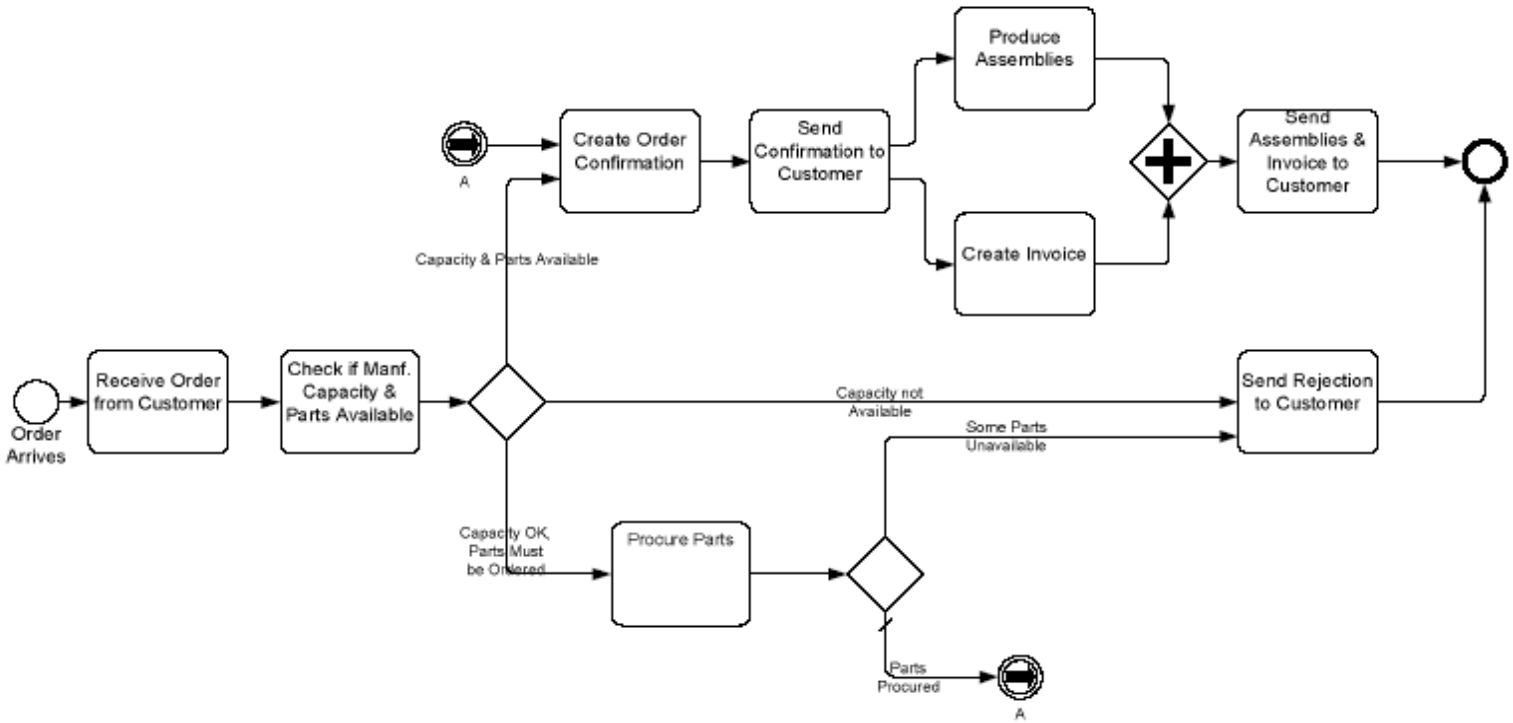
Normal flow

- 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
  - Normal Flow does not include exception flow or compensation flow



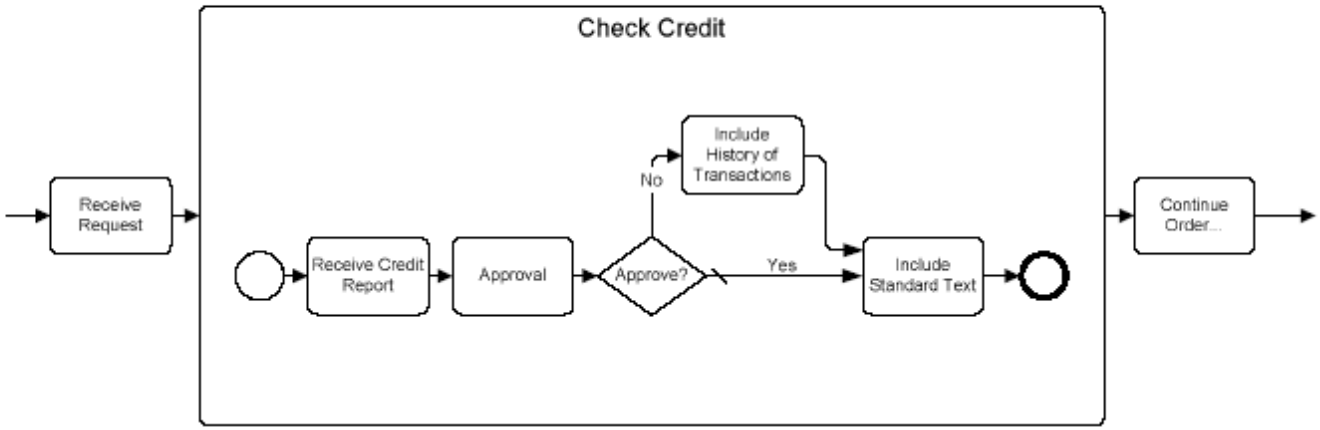
Link events

- Link Events can be used for Off-Page connectors
- Link Events can be used as “Go-To” objects

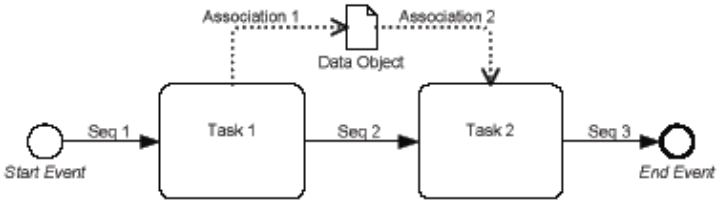


Process levels

- Processes can be developed hierarchically, with multiple levels through Sub-Processes
- Sequence Flow cannot cross a Sub-Process boundary
  - Message Flow and Associations can cross Sub-Process boundaries

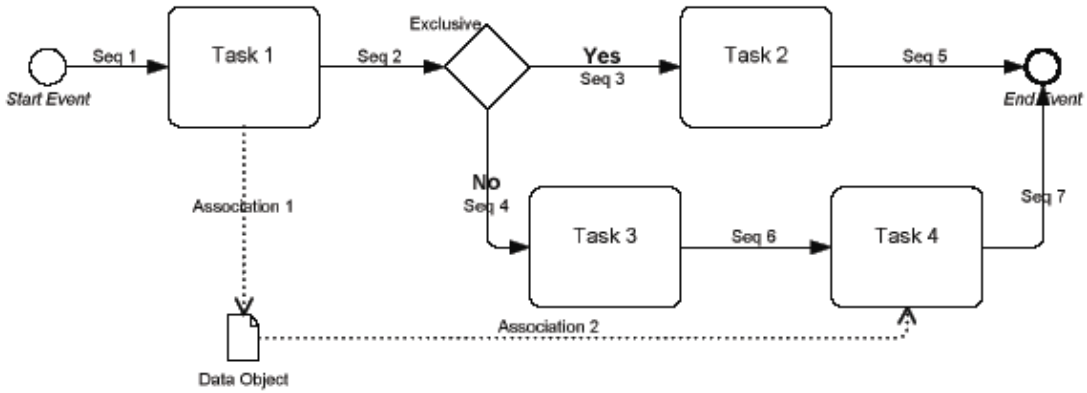


Data flow



Sequence Flow and Data Flow are decoupled

They can be bound together

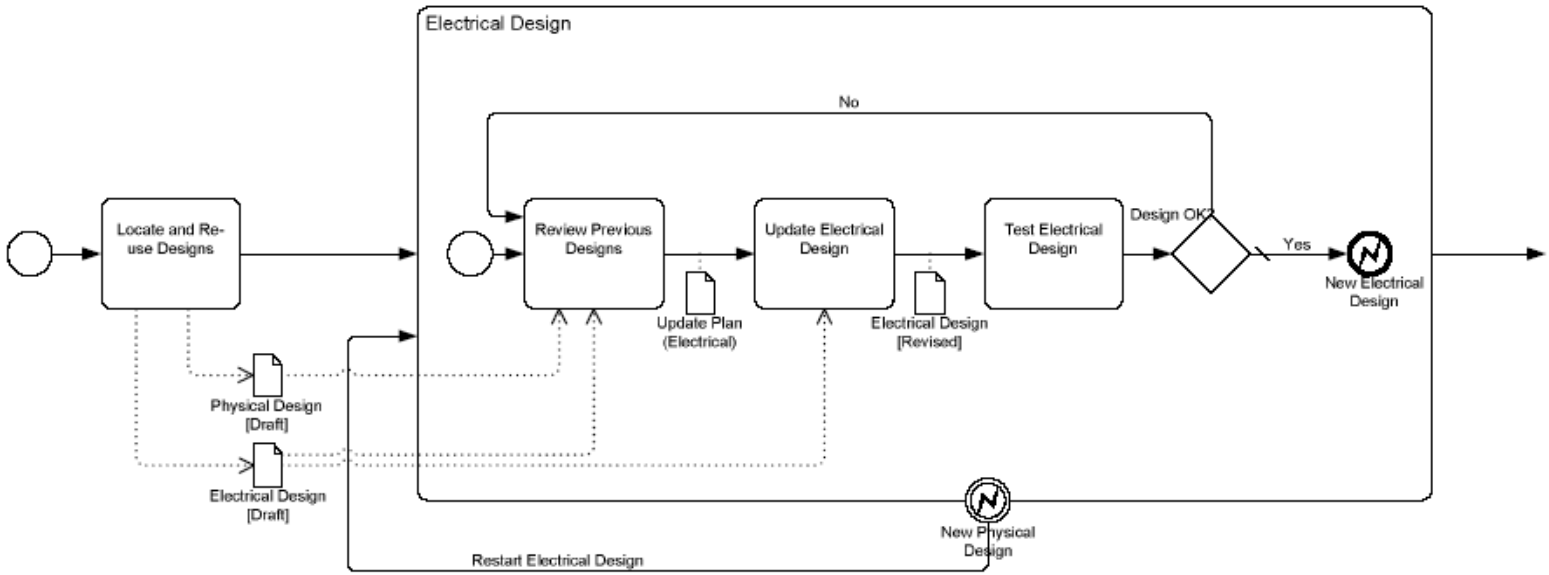


Use case for decoupling

Exception Handling

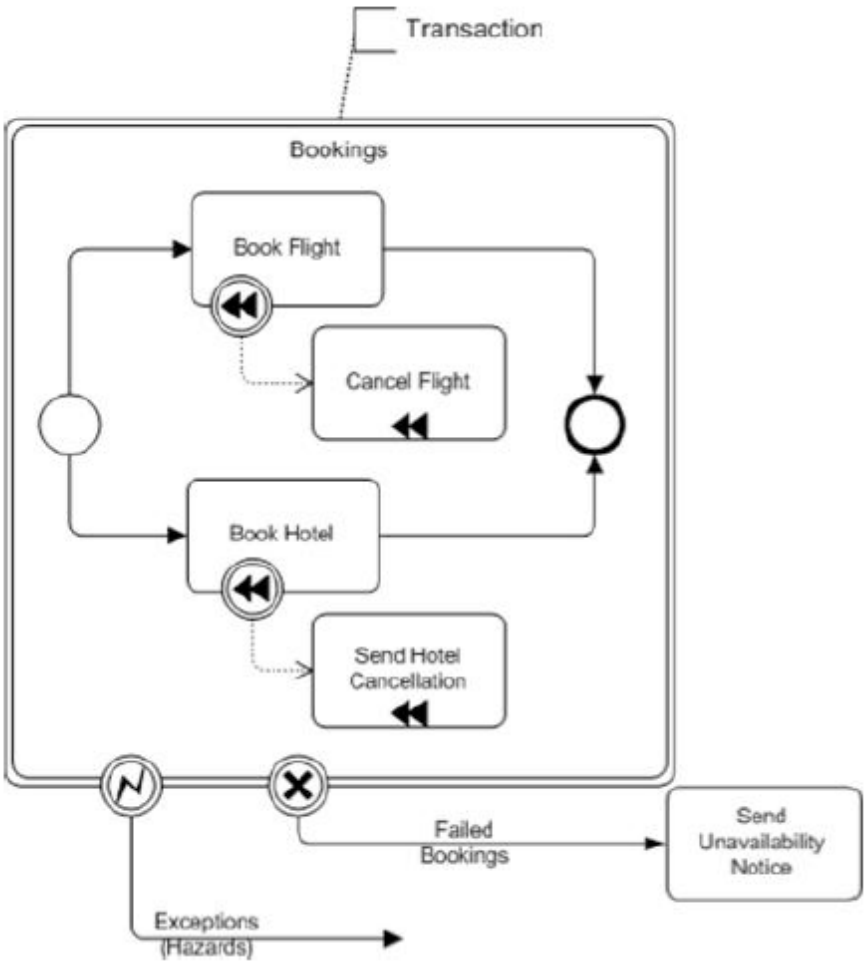
- Intermediate Events attached to the boundary of an activity represent triggers that can interrupt the activity
- All work within the activity will be stopped and flow will proceed from the Event.
- Timer, Exceptions, Messages, etc. can be Triggers.

Exceptions



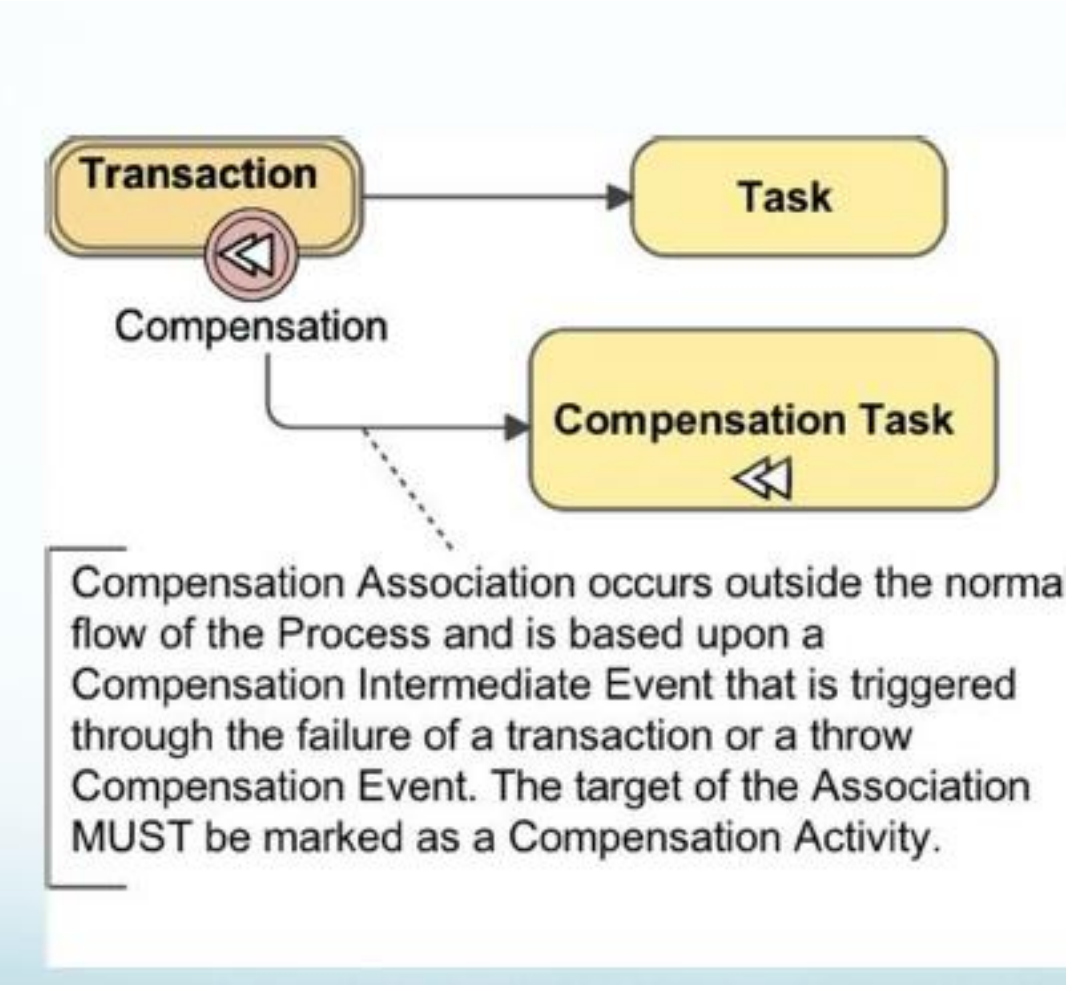
Intermediate Events attached to the boundary of an activity represent triggers that can interrupt the activity. All work within the activity will be stopped and flow will proceed from the Event. Timer, Errors, Messages, etc. can be Triggers.

Compensation handling & transactions



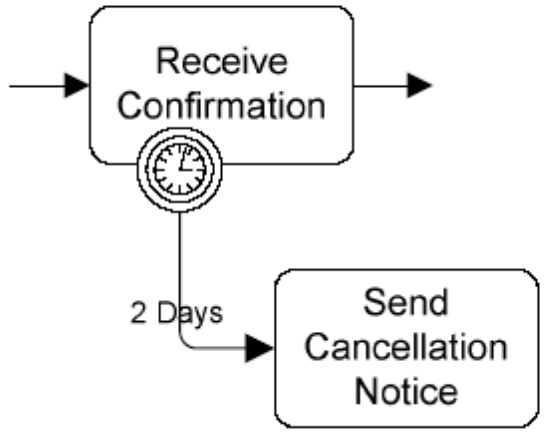
- A Transaction is an activity that has a double border. Transactions are supported by a transaction protocol (e.g., WS-Transaction)
- Normal Outgoing Sequence Flow represents the path to follow a successful completion
- A Cancel Intermediate Event represents the path to follow a cancelled completion
- An Exception Intermediate Event represents the path to follow a transaction hazard (but no compensation is performed)
- Activities used for compensate (with marker) are outside normal flow and are Associated normal activities. Compensation flows "backwards."

Compensation Flow



Timers

Timers to add delays in the Process



Timeouts for exception handling

BPMN v2.0.2 (Dec 2013) What is new? I

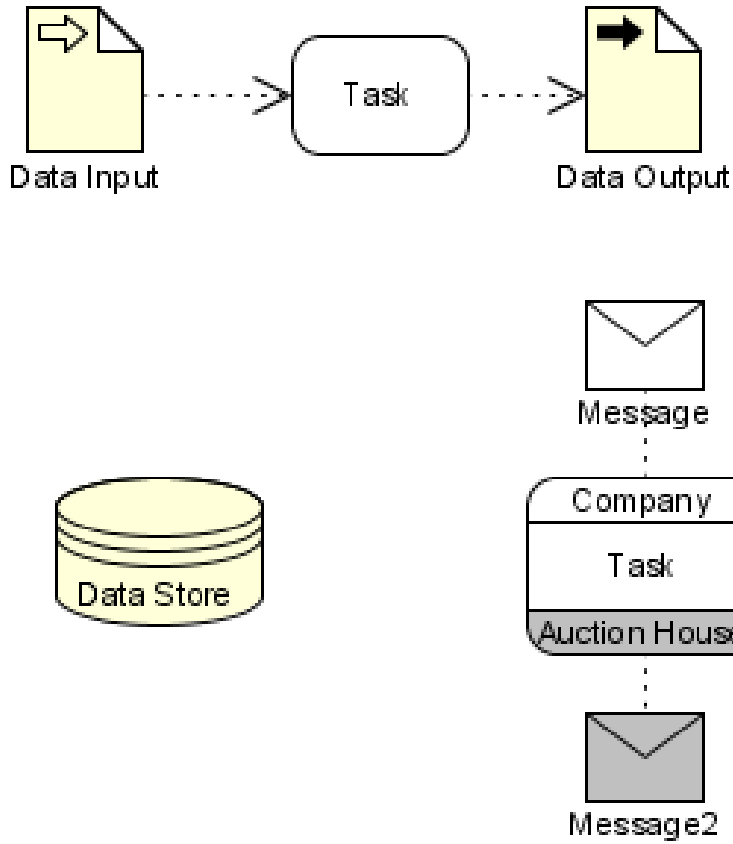
- The formalization of the implicit metamodel of BPMN 1.2
- The refinement and formalization of the BPMN execution semantics
- Interchange of BPMN models, including the diagram layout, through XML and XMI Schemas (and XSLT transformation)
- A set of conformance levels to support different process modeling markets

BPMN 2.0 - What is new? II

- New Process Elements:
  - Non-Interrupting Events (attached to Activities)
  - Optional Event Sub-Processes
  - Graphical markers for individual Task types
  - New data representation
- Improved support for human interactions
- Enhanced modeling of services
- Upgrade support for process data

New Data Representation

- **Data Input** is an external input for the entire process. It can be read by an activity
- **Data Output** is a variable available as result of the entire process
- 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.
- A **Message** is used to depict the contents of a communication between two Participants.



Data

Data Object

Primary construct for modeling data within the Process flow. DataObject has a well-defined lifecycle, with resulting access constraints.

Data Store

Provides a mechanism for Activities to retrieve or update stored information that will persist beyond the scope of the Process.

Data Input

Data Input is a declaration that a particular kind of data will be used as input of the InputOutputSpecification.

Data Output

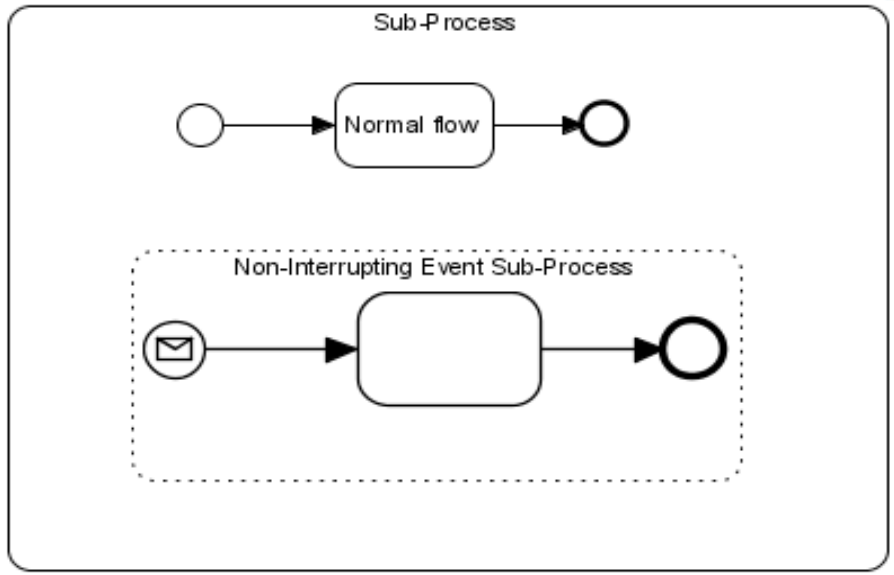
Data Output is a declaration that a particular kind of data can be produced as output of the InputOutputSpecification.

Call Activity

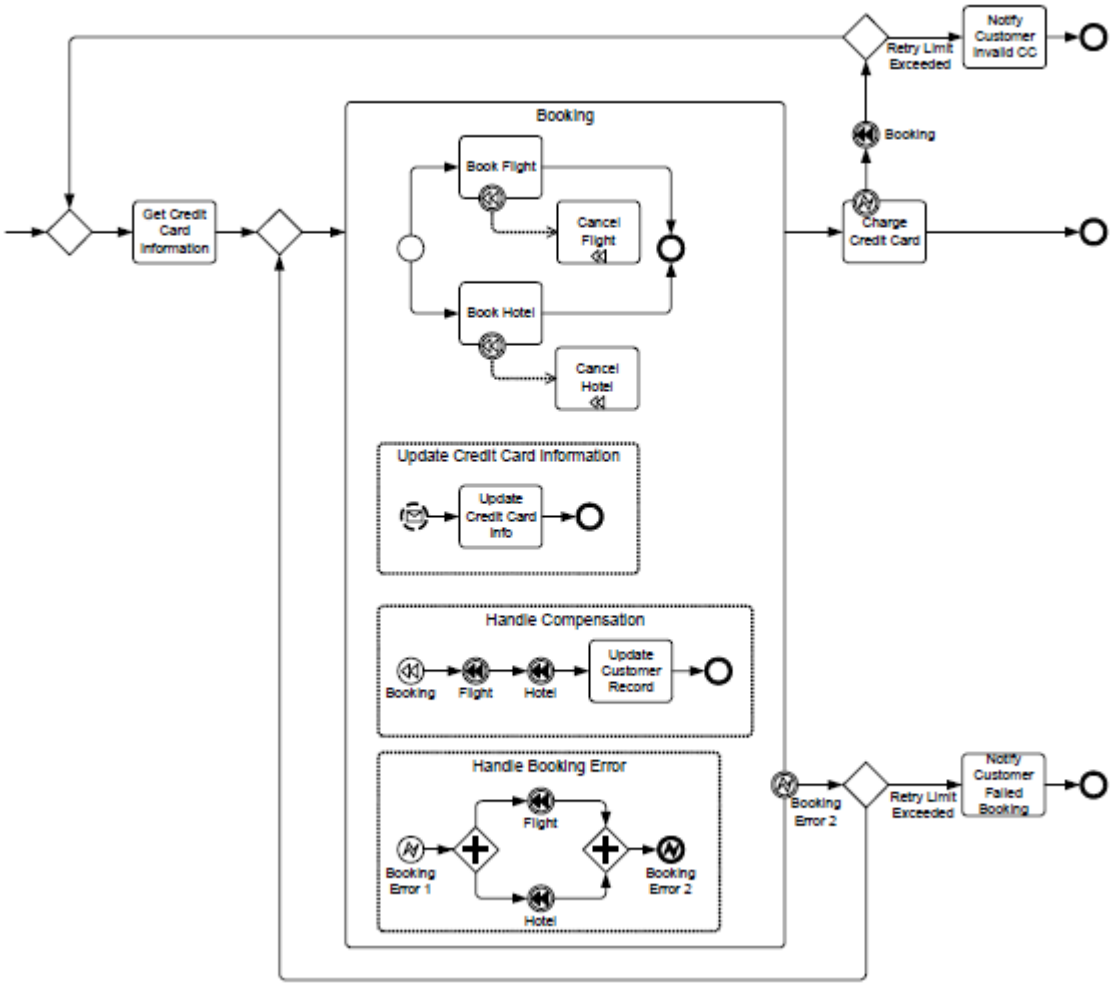
- Replaces the Reusable and Reference Sub-process
- The collapsed shape represents the calling activity
- The expanded view represents the called global task of process
- Data must be explicitly passed from the Call Activity to the global task or called process

Event Subprocess

- Run in the context of a activity to listening
- Align the event handling of BPMN and BPEL
- Event subprocess is drawn as a floating box within a expanded view of a regular subprocess

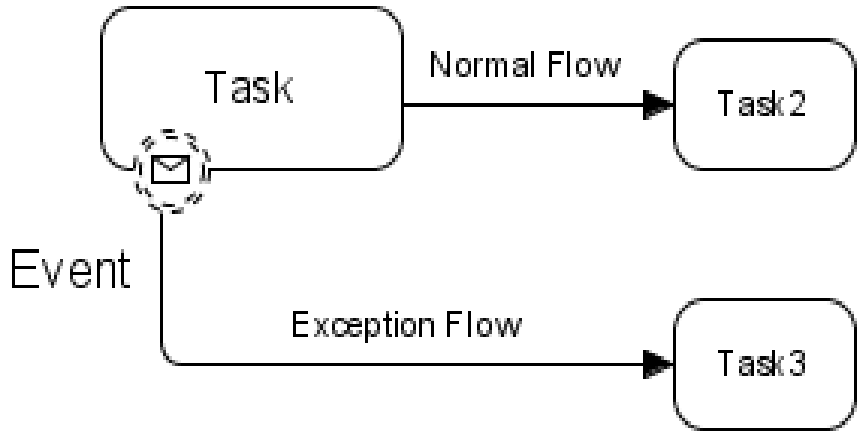


Event Subprocess



Non-Interrupting Event

- Intermediate Event
- Maintain the main flow and throws exception flow in parallel
- Dashed shape



Escalation Event

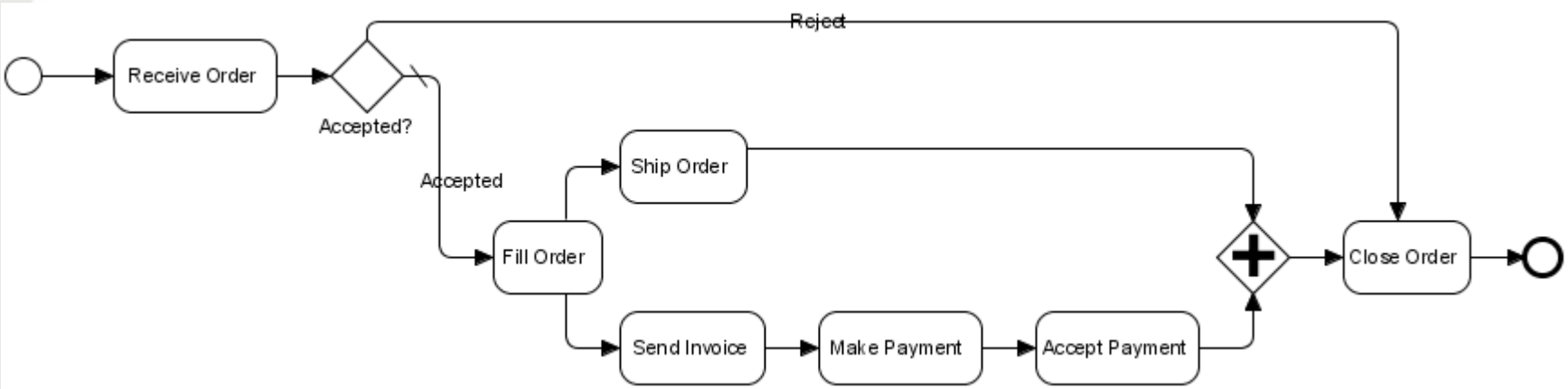
- Variant of error event
- Escalation does not imply an error, just some additional processing triggered during a process activity

BPMN 2.0 - Diagrams

- Process diagram
- Collaboration diagram
- Conversation diagram
- Choreography diagram

Process Diagram

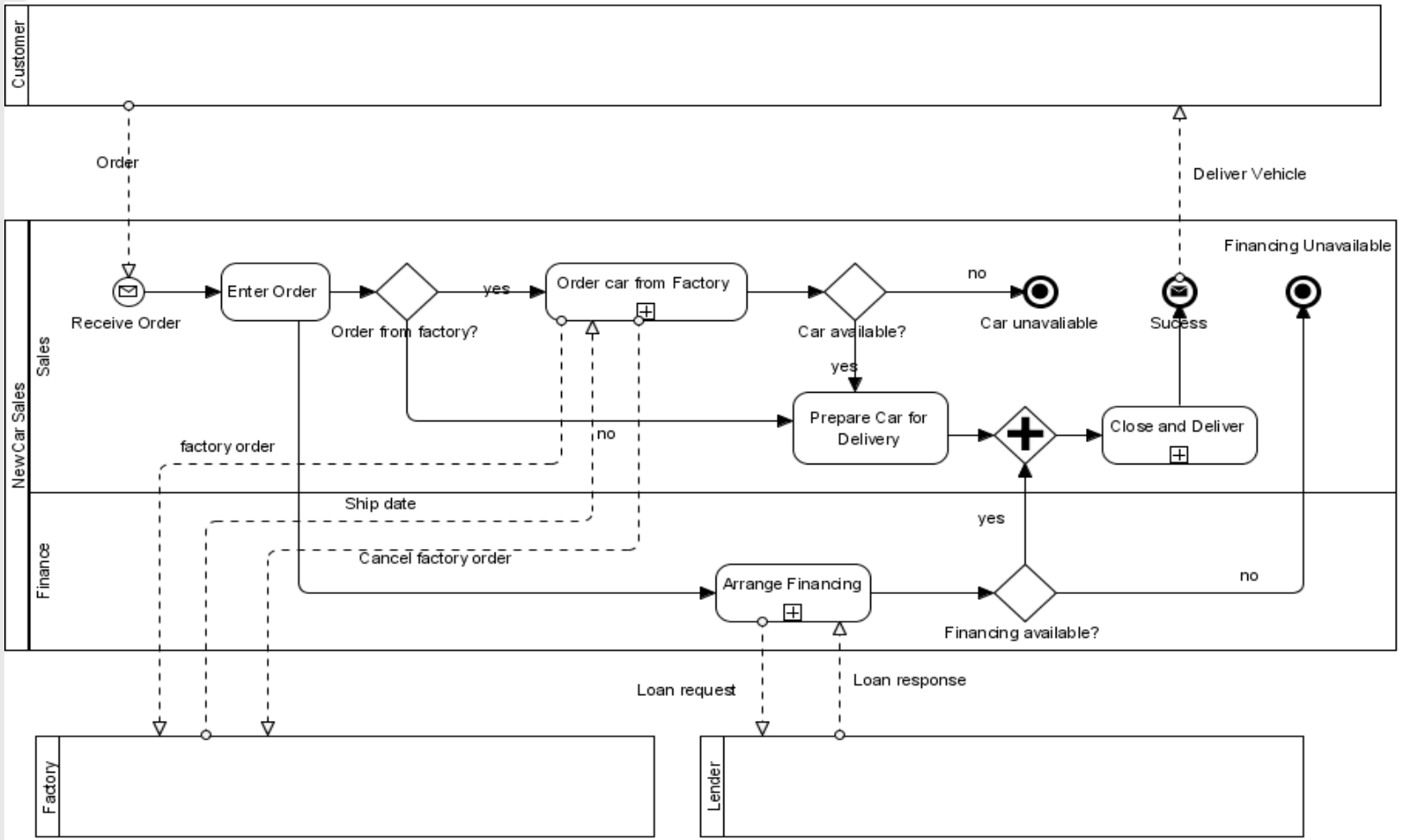
- Basic flowchart model
- Build inside or outside of the swimlanes
- Represents just a process without interaction of participants



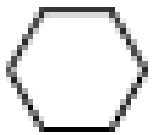
Collaboration Diagram

- Provides a view of the interactions (the flow of messages) between two or more business partners (i.e., pools)
- Collaborations can be combined with Processes to show how the interactions are related to the internal Process activities
- The pool with a process is visible, other pools are collapsed

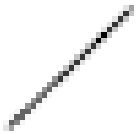
Collaboration Diagram



Conversations



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



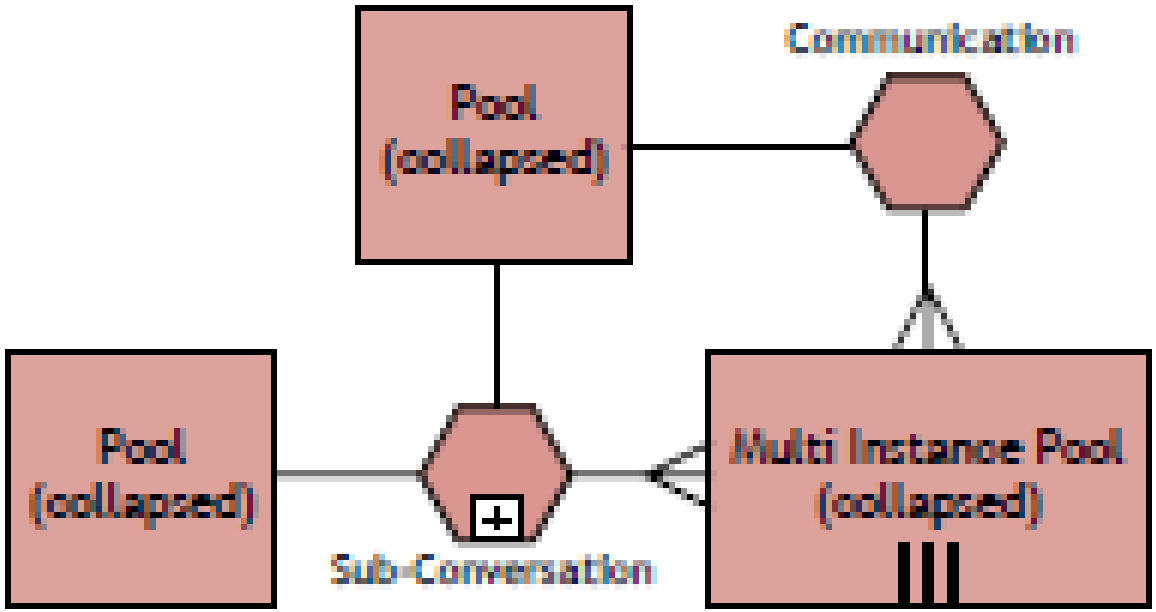
A Conversation Link connects Communications and Participants..

A Forked Conversation Link connects Communications and multiple Participants.

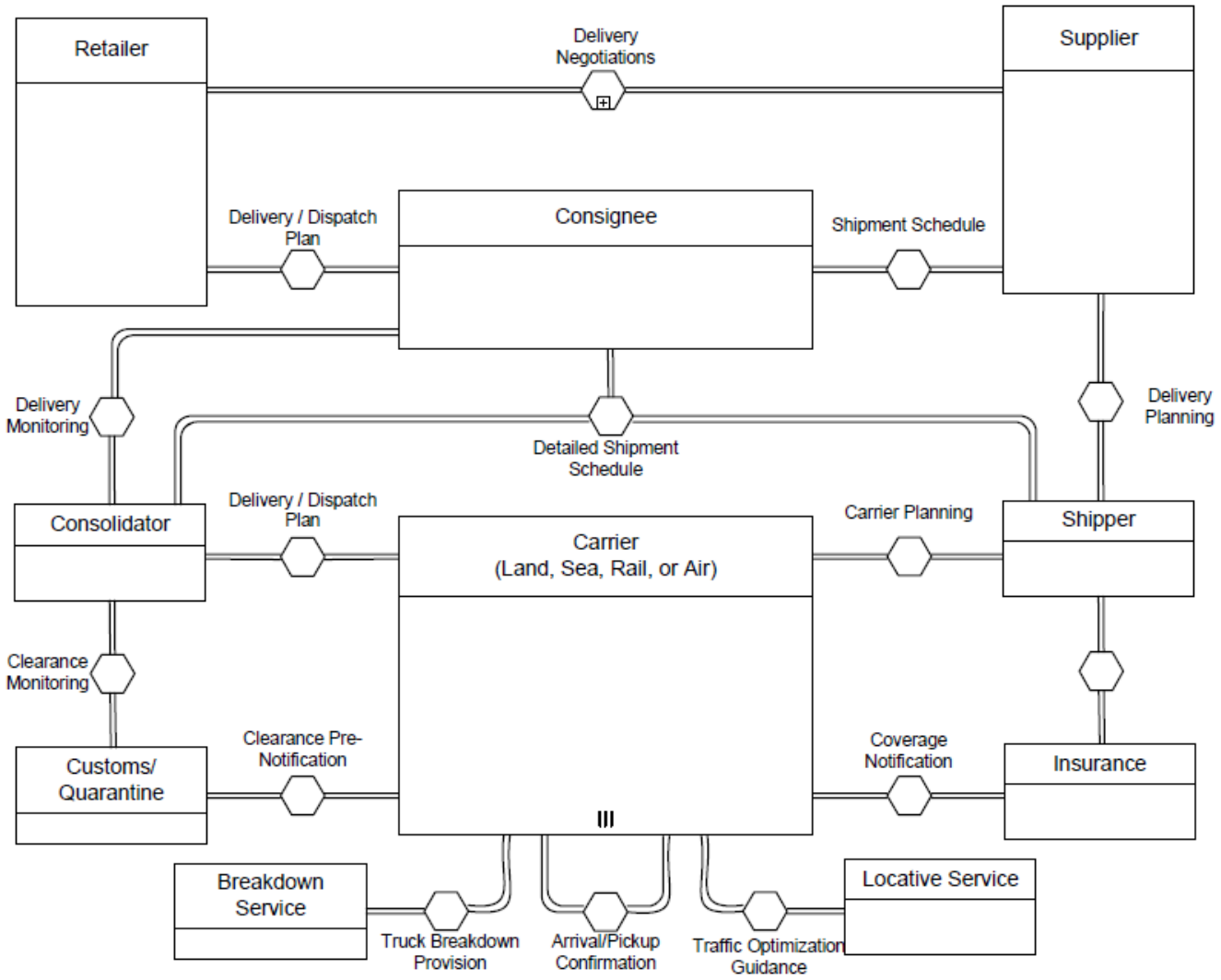
Conversation Diagram

- Conversation Diagram is used to model a "who with whom and what"
- All process participants can be included in a compact form of modeling as long as denoting which communication is involved
- All process participants can be included in a compact form of modeling as long as denoting which communication is involved.
- Process participants are represented as Pools (collapsed) in Conversation diagrams.
- The Communication-Shape defines a set of logically related message exchanges which are tied to Pools via Conversation Links.
- Furthermore Sub-Conversations can be used to define abstractions.

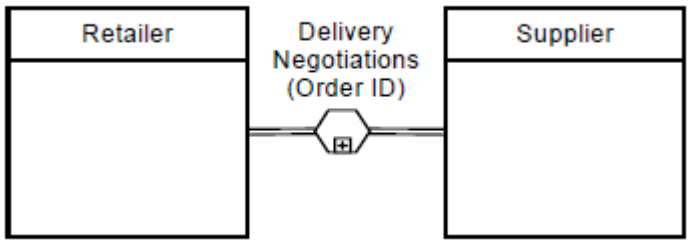
Conversation Diagram



Conversation Diagram



Conversation Diagram



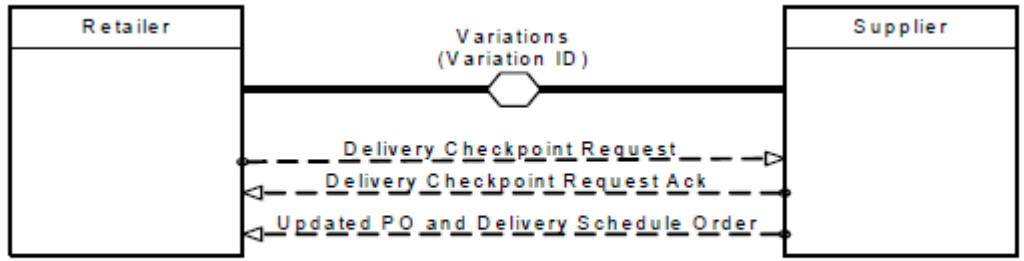


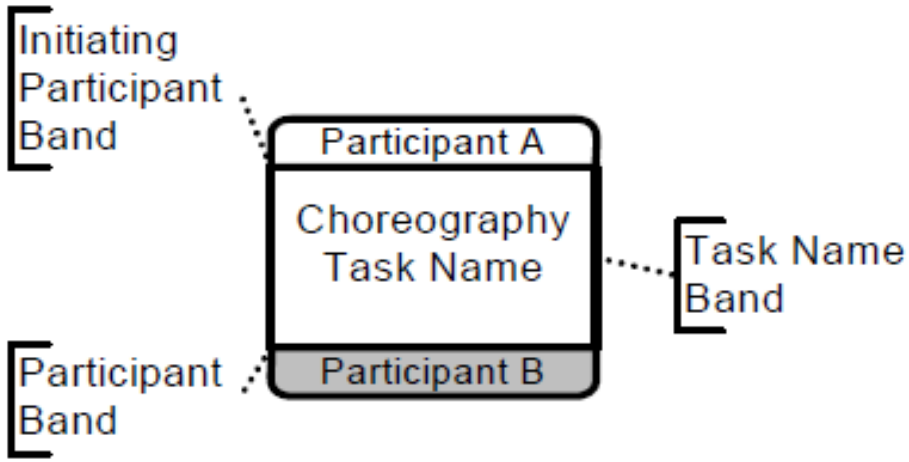
Figure 9.20 – An example of a Sub-Conversation expanded to a Conversation and Message Flow

Figure 9.21 – An example of a Sub-Conversation that is fully expanded

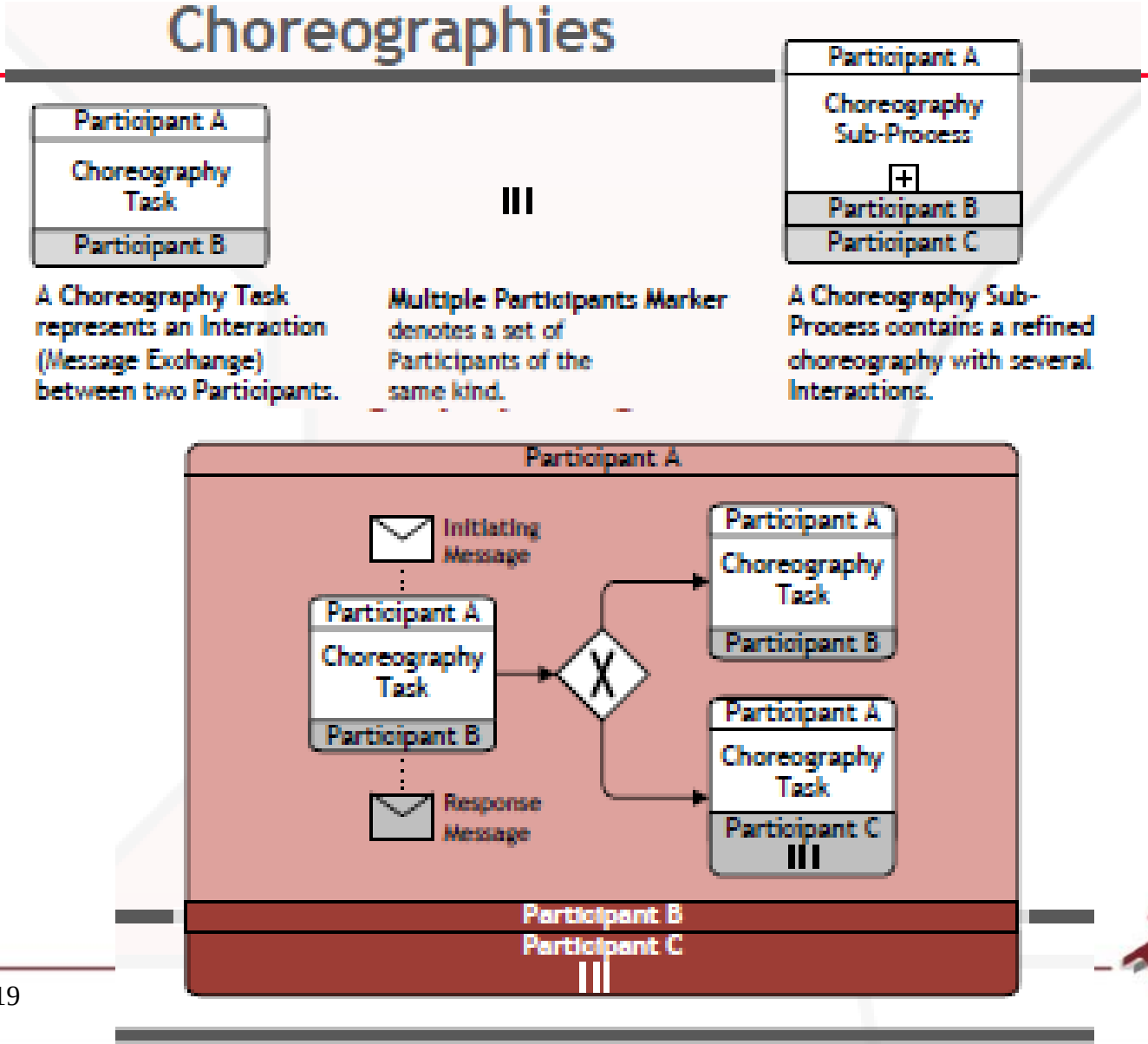
Choreography Diagram

- Represents more detailed information than Conversation Diagram
- A Choreography usually stands for exactly one Choreography whereas Conversation diagram implicitly hint at multiple Choreographies
- Choreography diagram do show the sequence of certain interactions
- Contracts about communication can be arranged between process participants

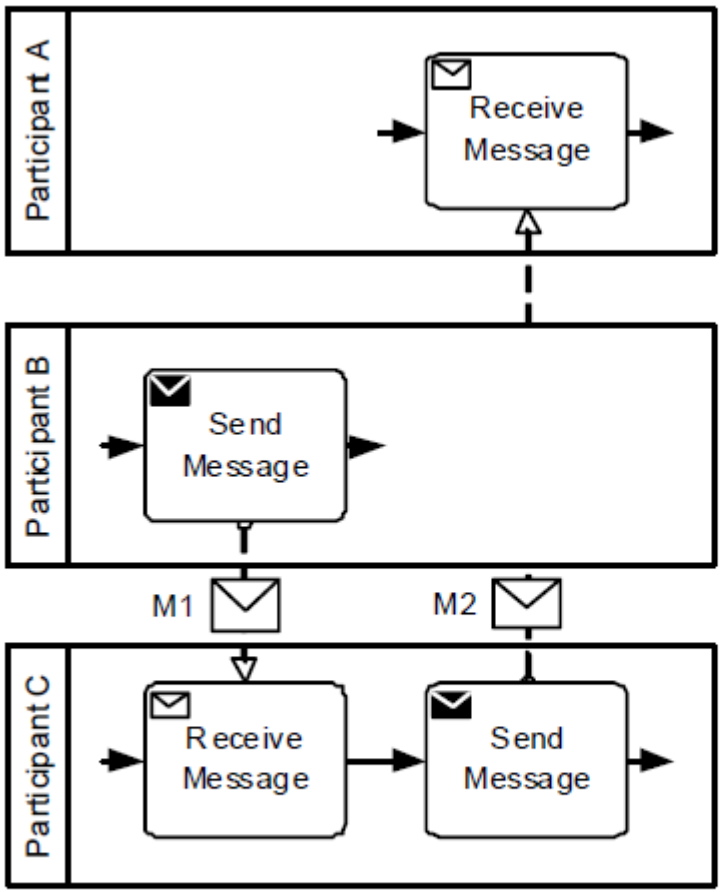
Choreography Task



Choreography Diagram



Choreography Diagram



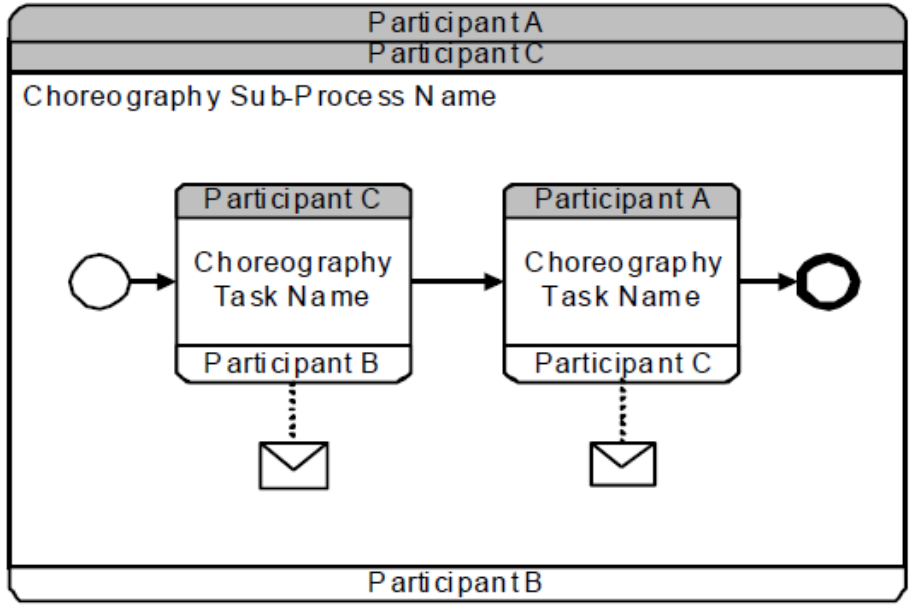


Figure 11.19 – An expanded Sub-Choreography

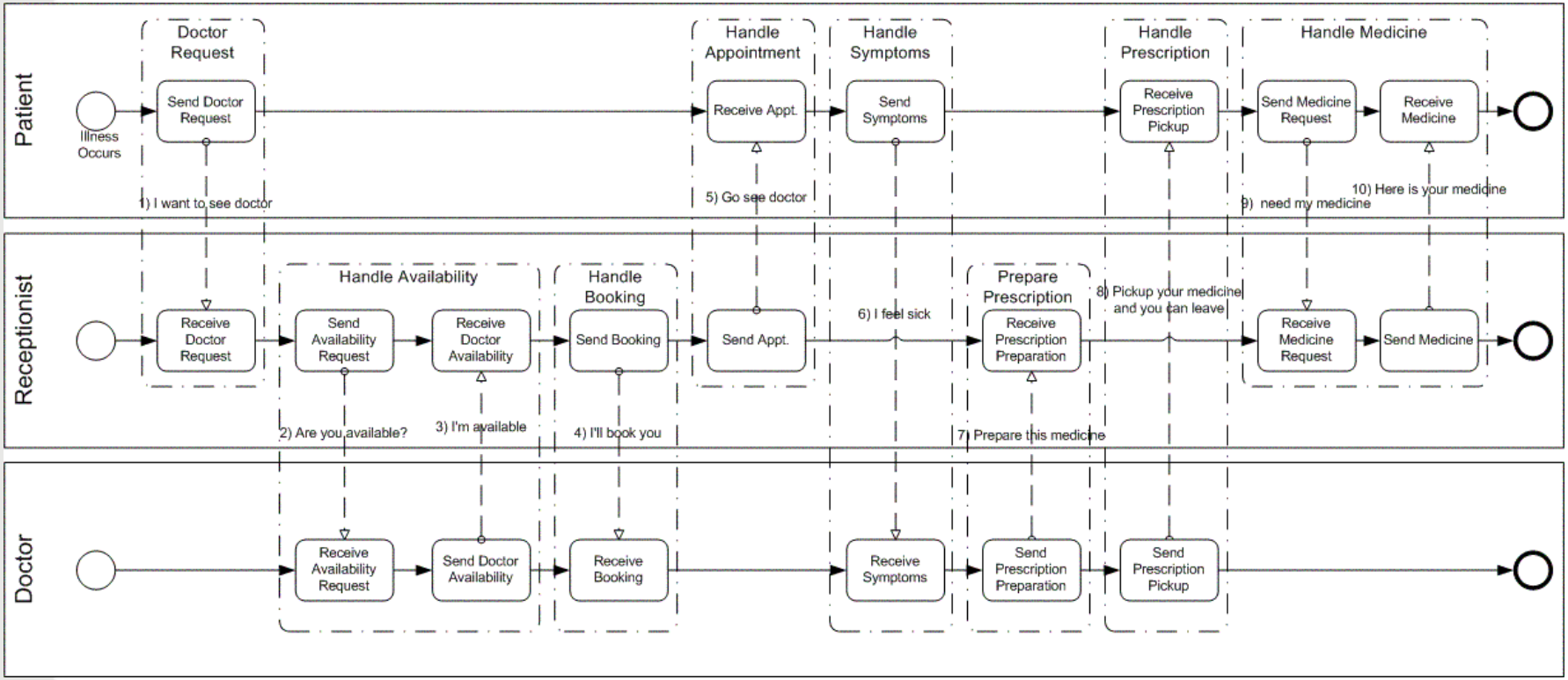
Figure 11.20 – A Collaboration view of an expanded Sub-Choreography

Choreography Diagram

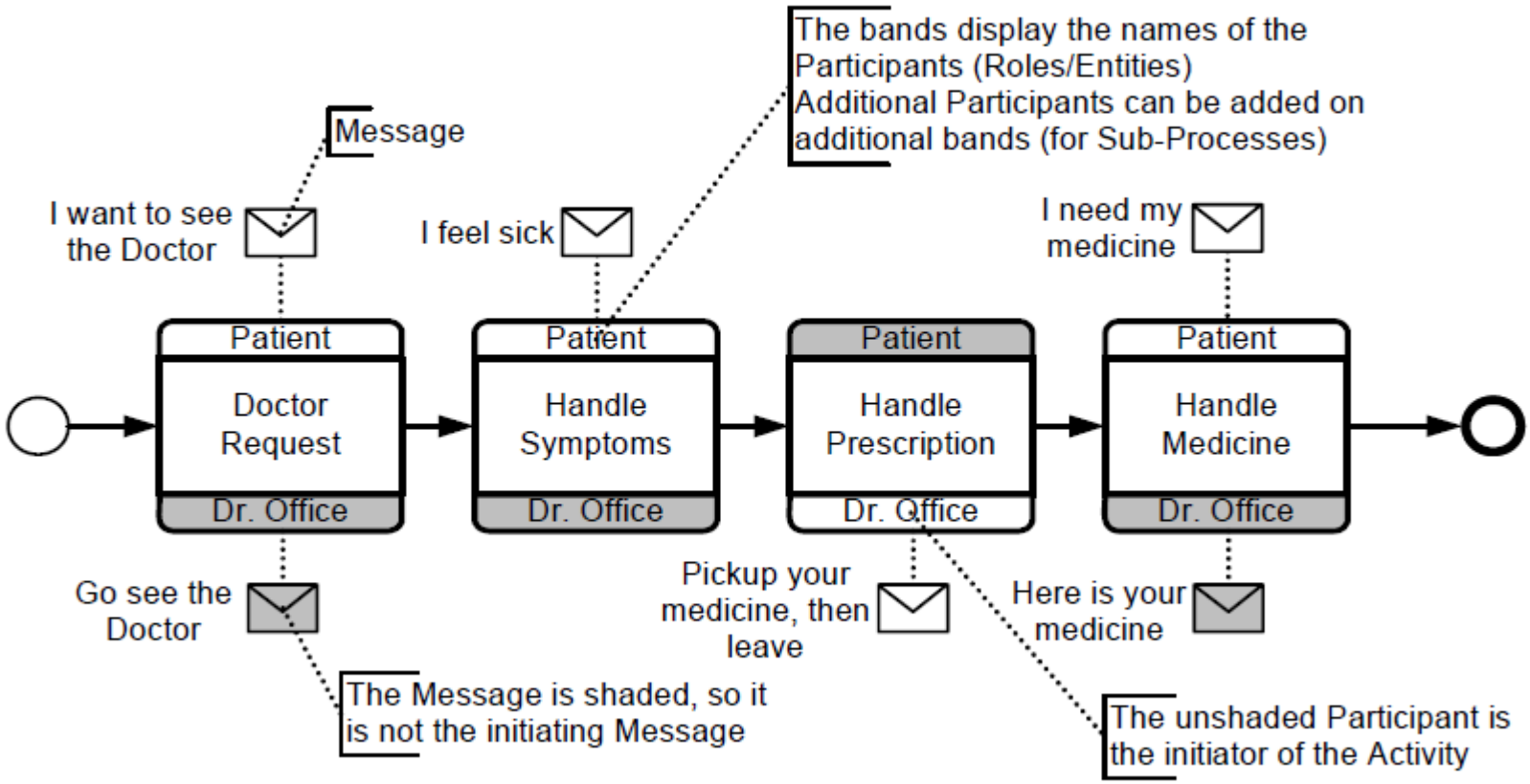
A text description of the choreography was presented as so:

1) Patient send a "I want to see doctor" message to the Receptionist
2) Receptionist send a "Are you available ?" message to a a list of Doctors
3) One doctor send a "I'm available" message to the Receptionist.
4) Receptionist send a "I'll book you" message to the Doctor.
5) Receptionist send a "Go see doctor" message to the Patient
6) Patient send a "I feel sick" message to Doctor
7) Doctor send a "Prepare this medicine" message to Receptionist
8) Doctor send a "Pickup your medicine and you can leave" message to Patient
9) Patient send a "I need my medicine" message to Receptionist
10) Receptionist send a "Here is your medicine" message to Patient

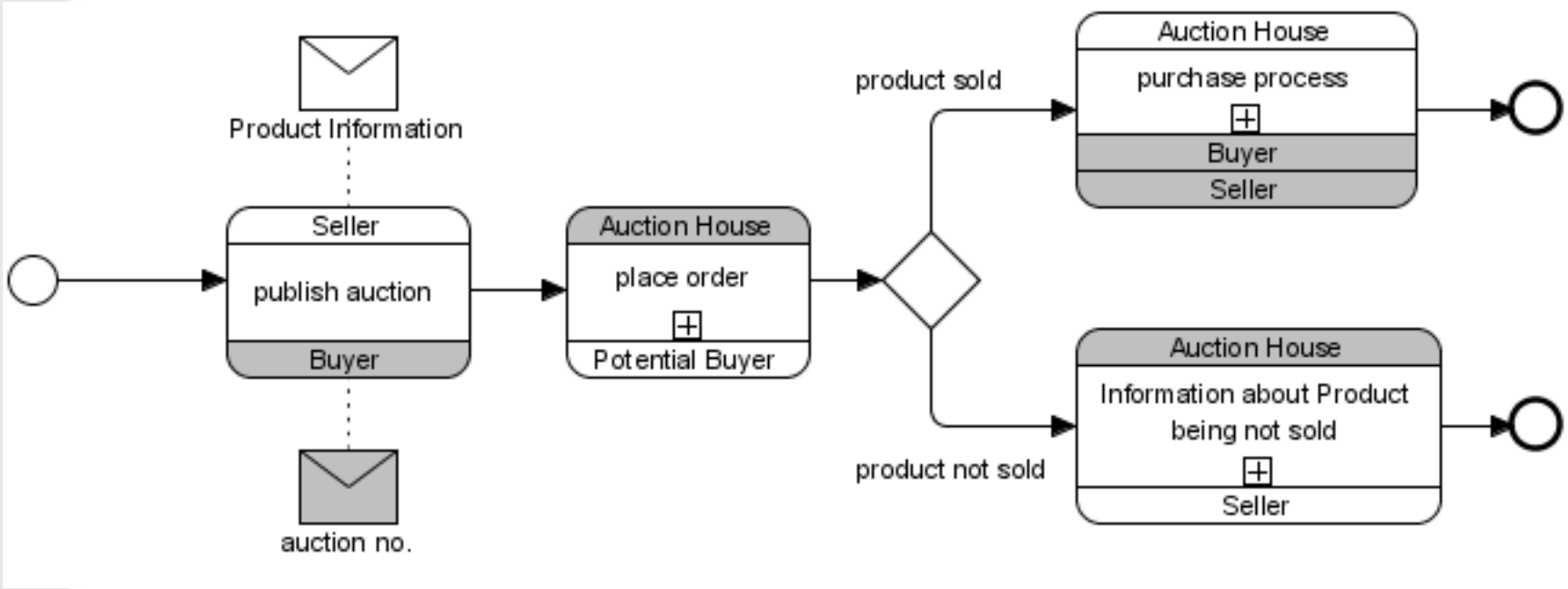
Collaboration Diagram



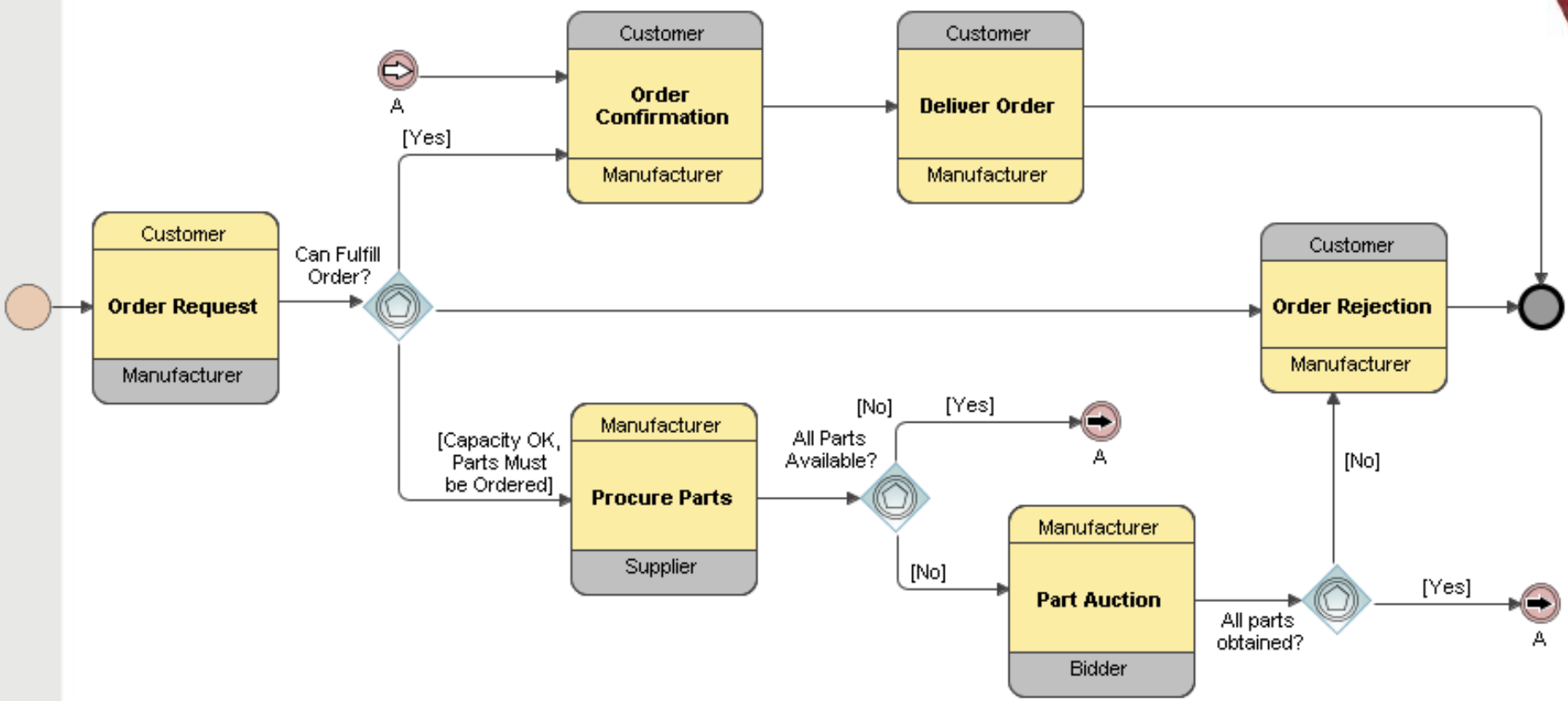
Choreography Diagram



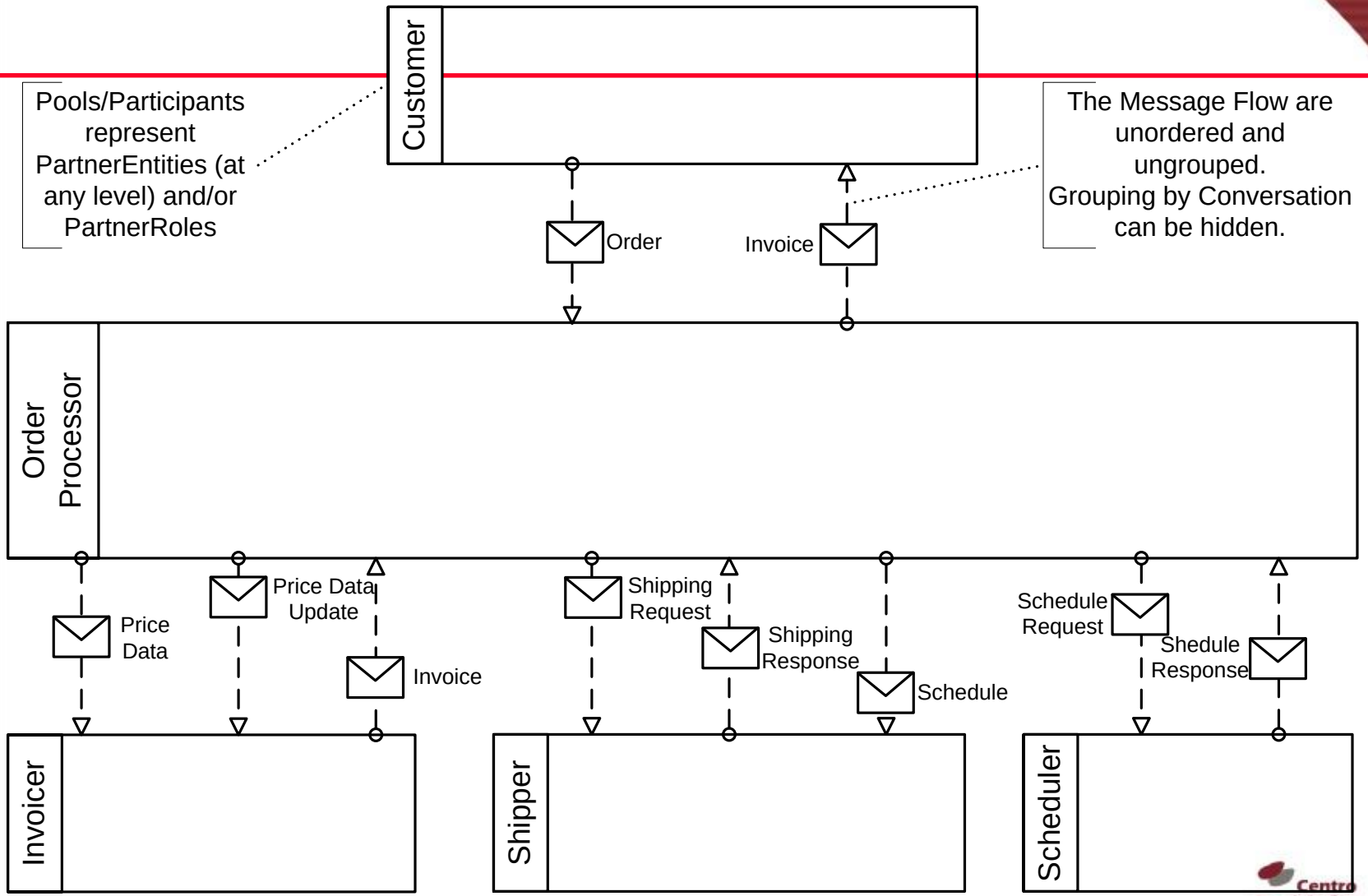
Choreography Diagram



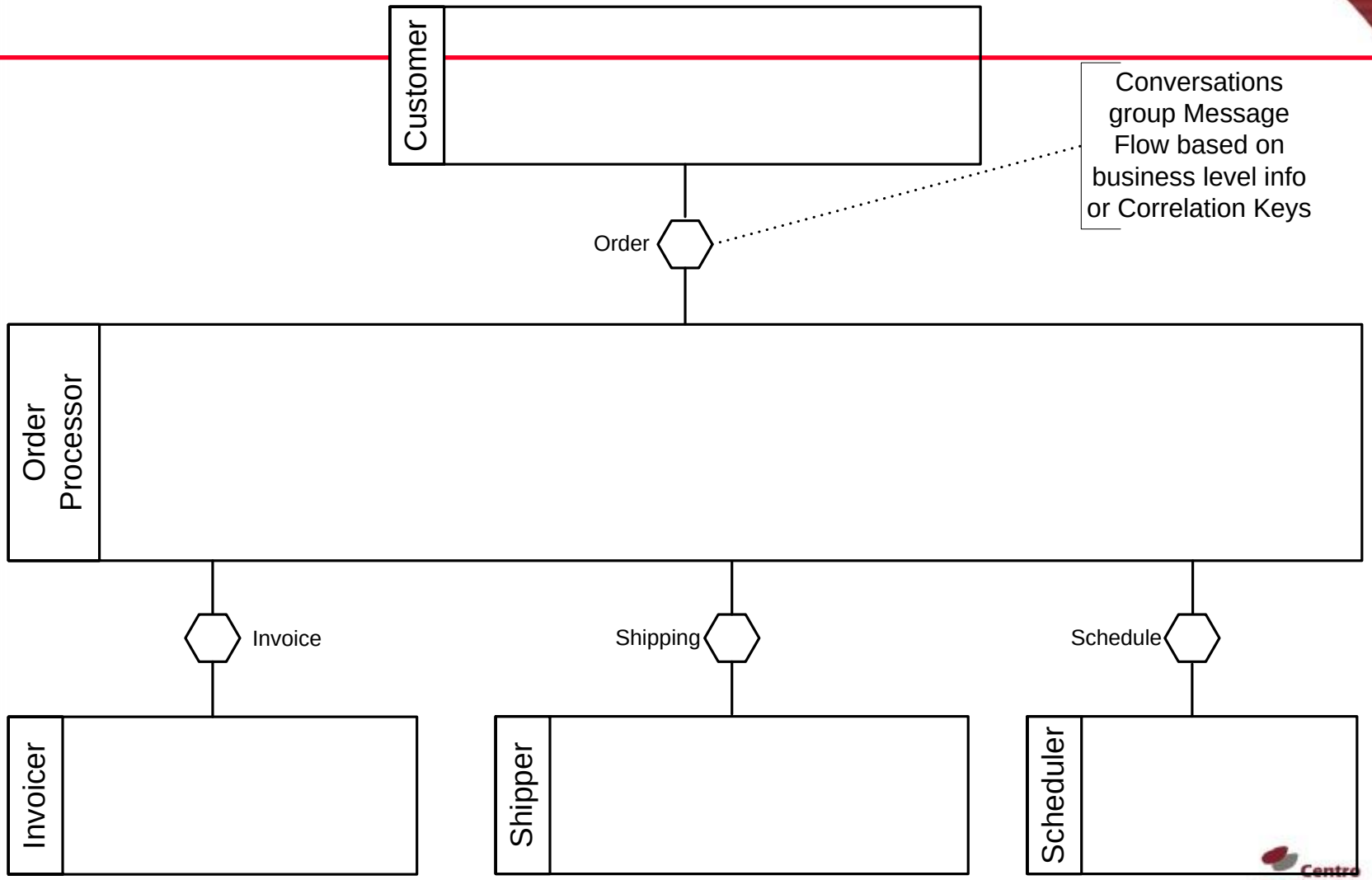
Choreography Diagram



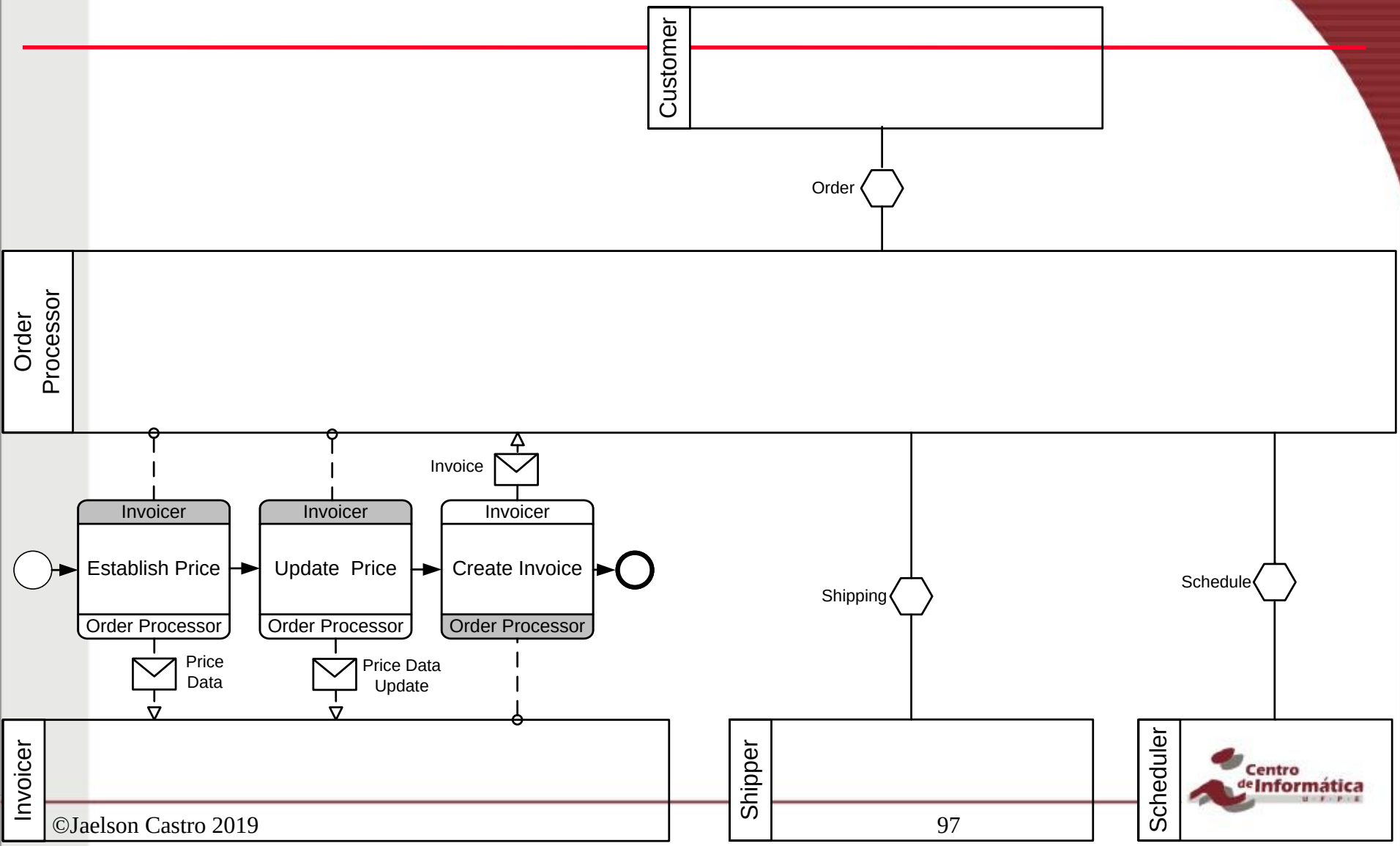
Associated BPMN 2.0 Collaboration



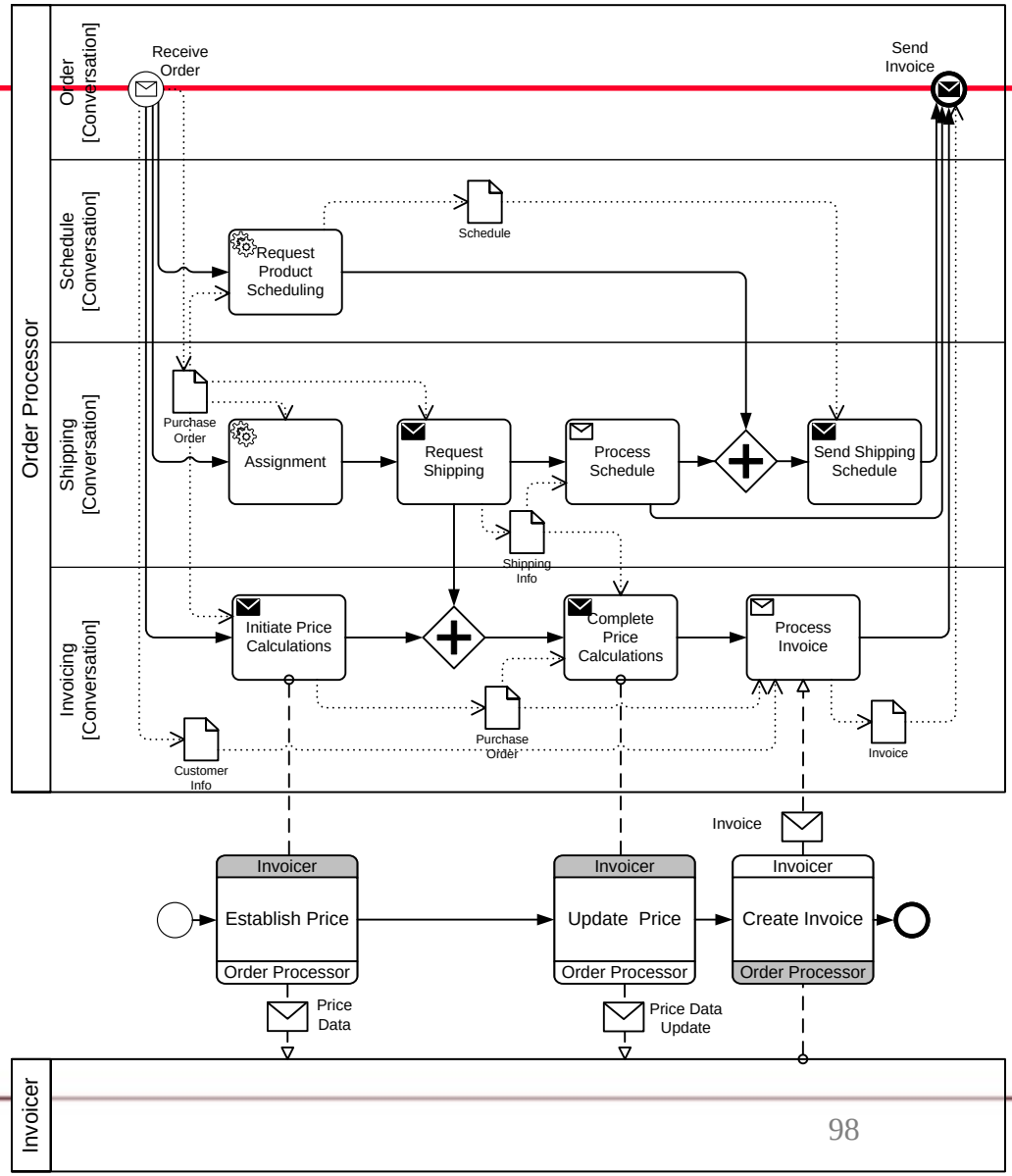
Associated BPMN 2.0 Conversation



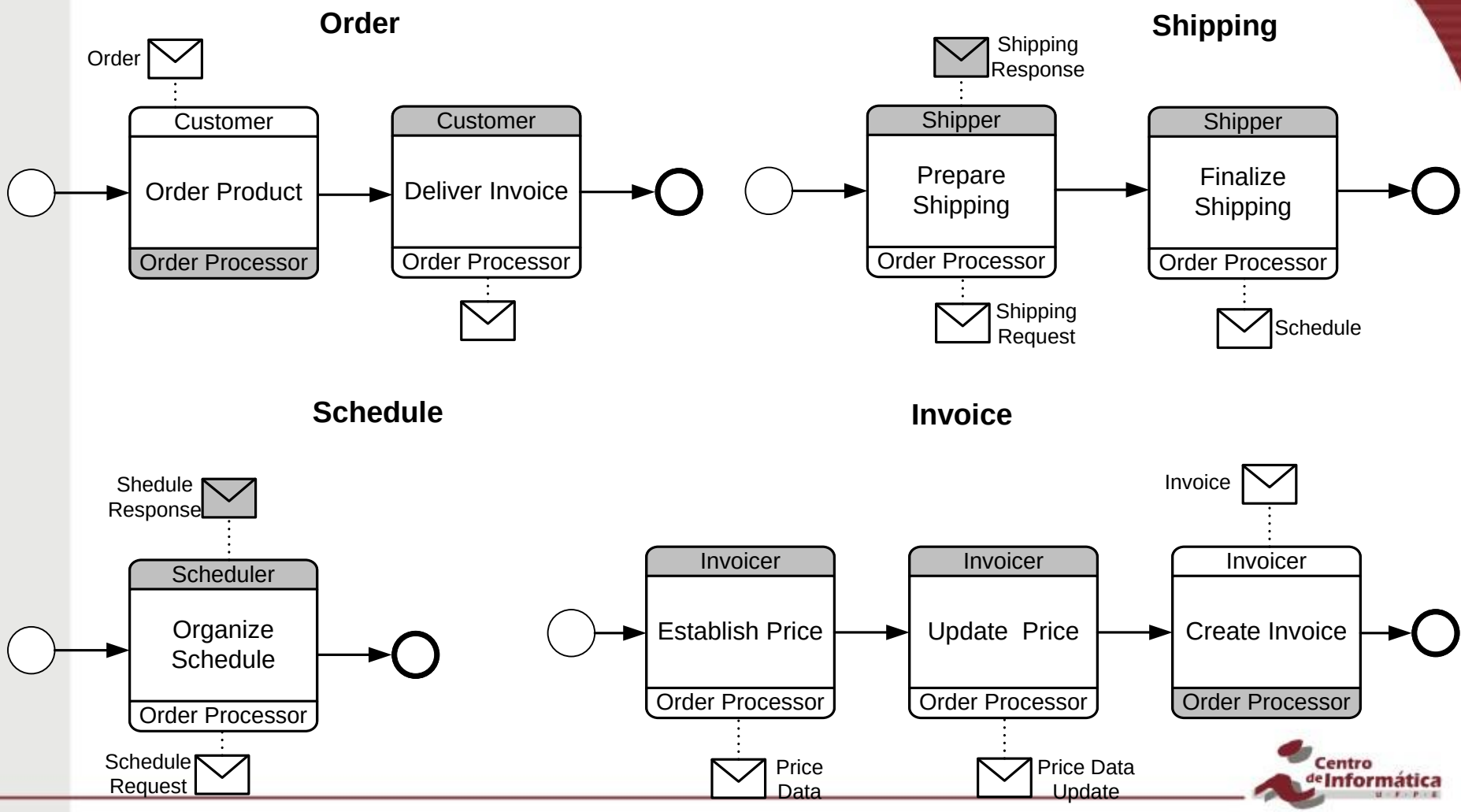
Associated BPMN 2.0 Conversation (showing mini-Choreography)



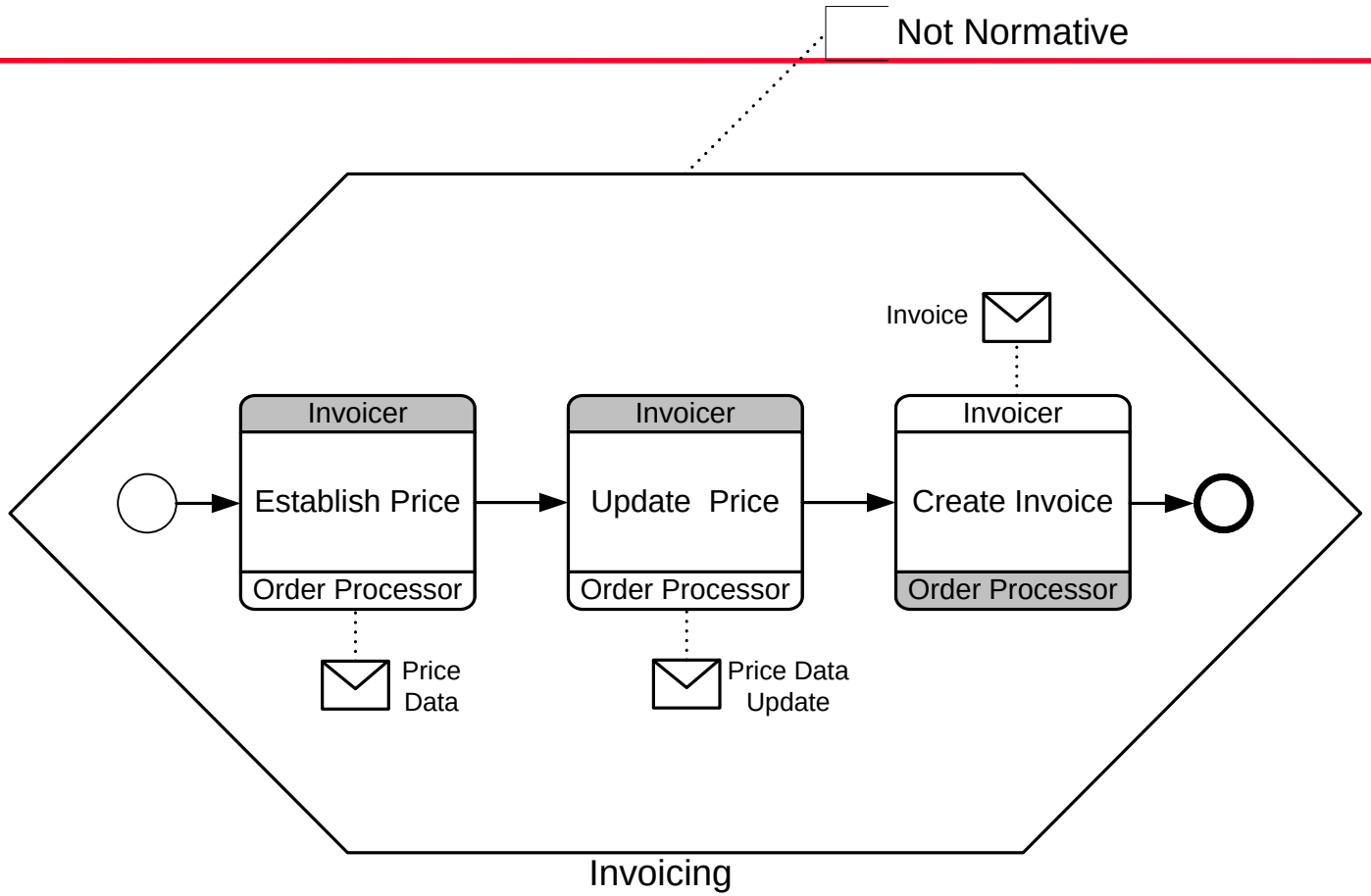
BPMN 2.0 Process (with Definitional Collaboration and Choreography)



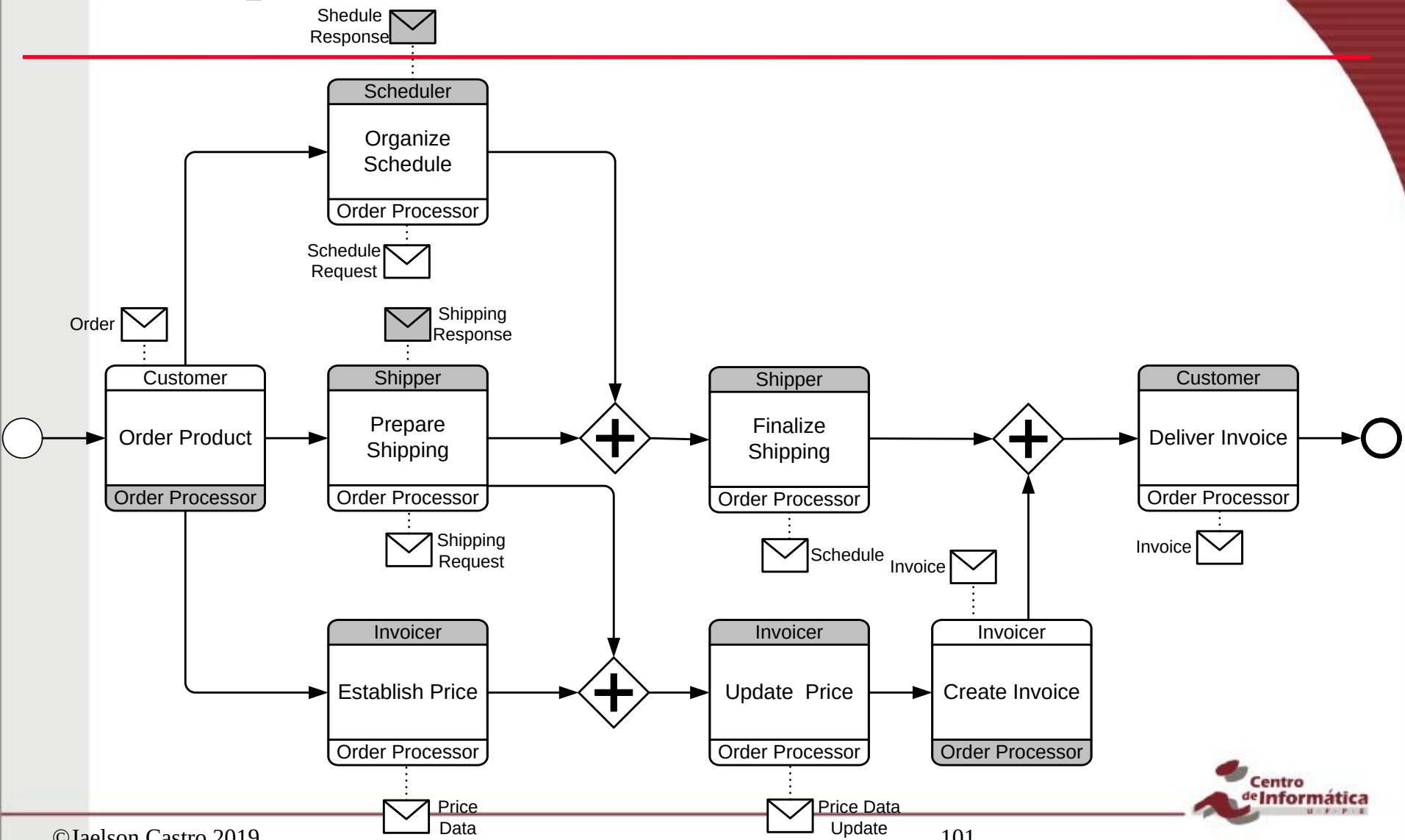
Associated BPMN 2.0 Choreographies (per Conversation)



Encapsulated Conversation (and Choreography)



Associated BPMN 2.0 Choreography (Complete)



Summary

- Need to understand business processes
- Existing business process to understand the problem
- Potential changes to the business process to investigate alternative solutions