INF5120
”Modellbasert Systemutvikling”
”Modelbased System development”

Lecture 6-2: 20.02.2017

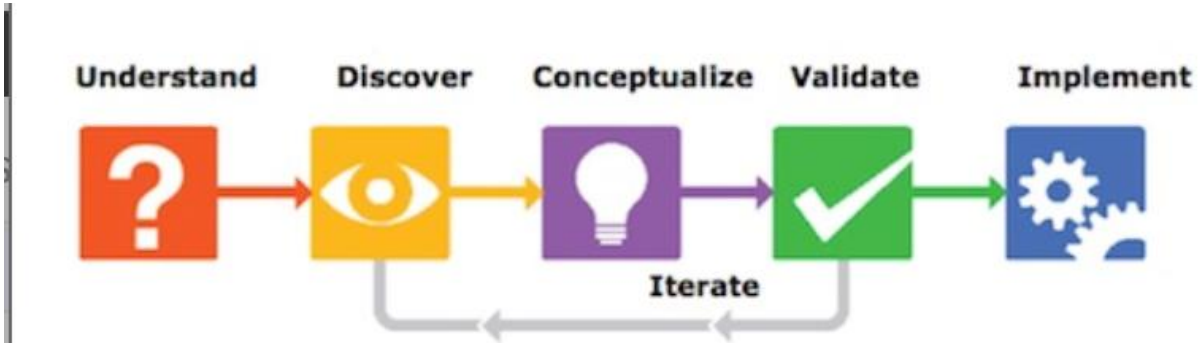
Arne-Jørgen Berre

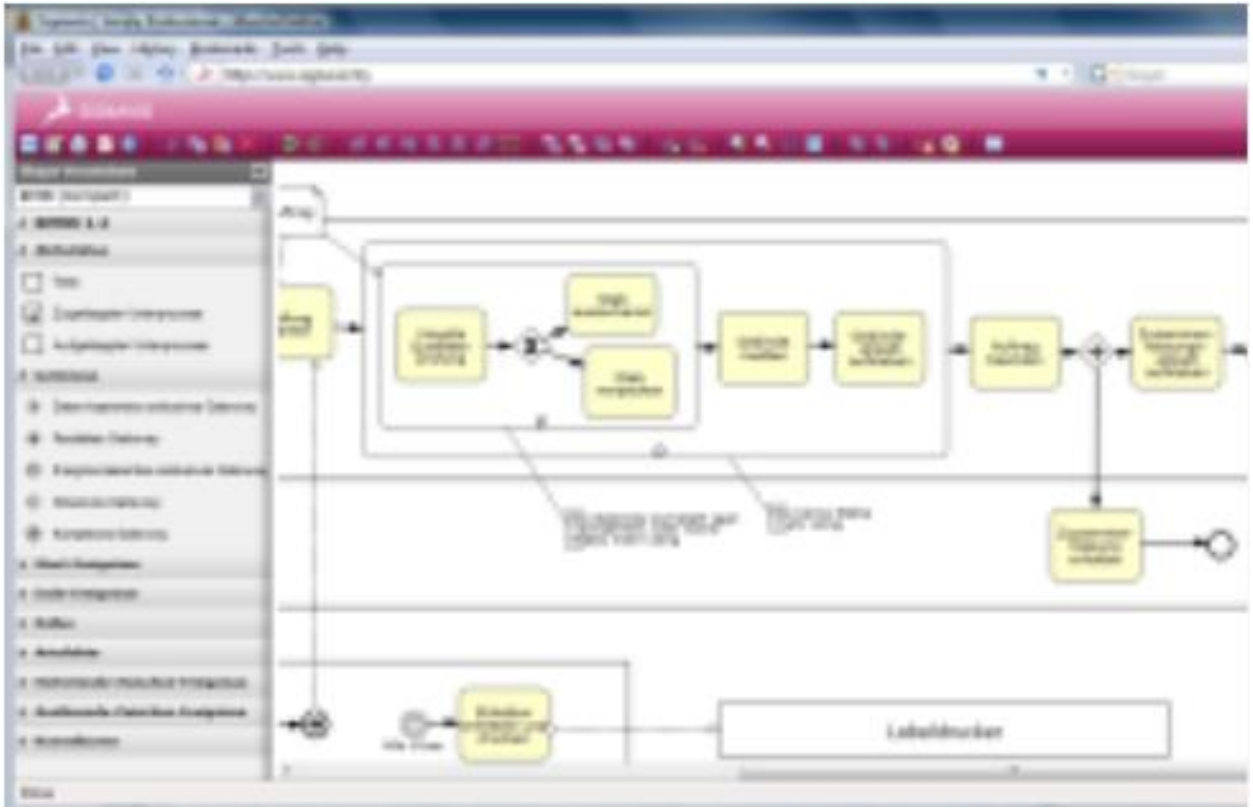
arneb@ifi.uio.no or Arne.J.Berre@sintef.no

Course parts (16 lectures)

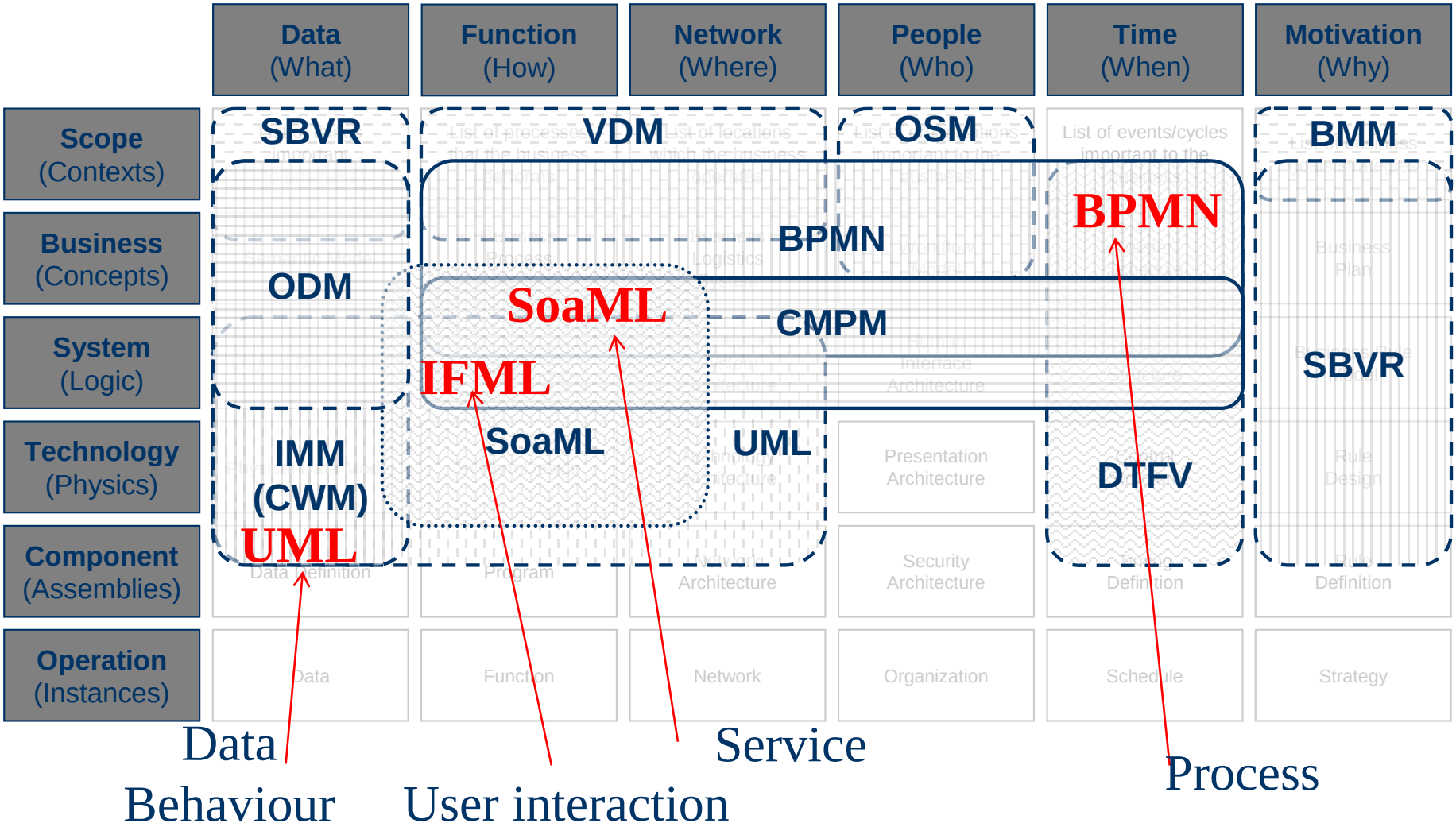
- January – February (1-7) (BAE/WebRatio):
- MDE-1: Introduction to INF5120
- MDE-2: Modeling structure and behaviour (UML and UML 2.0 and metamodeling) (B. Hjelle)
- BAE-1: Business Architecture – Business Model Canvas - Strategyzer tool.
- SAE-1: WebRatio for Mobile App development (Get an App up and running!)
- BAE-2: Essence, Scrum, User stories and Use cases 2.0, Backlog, with Someone
- BAE-3: BPMN process, VDML and UML Activ.Diagrams, … (MD/EA, Smaply and Balsamiq)
- BAE-4: Service Design, AT ONE,Touchpoints, UI, UX, Smaply and Balsamiq (Ragnhild)
- Oblig 1: BA Spec, WebRatio App1 (individual) (end of February, March 7th), Agile Scrum
- March (8,9) (MDE/IFML/Client-Side):
- MDE-3: Model driven engineering – Metamodels, DSL, UML Profiles, EMF, Sirius Editors
- SAE-2: IFML – Interaction Flow Modeling Language, WebRatio advanced
- April (10, 11,12,13) (BPMN, SAE/UML/Server-side):
- SAE-3 (Lecture 10, April 4th): BPMN and WebRatio BPM platform/Magicdraw BPMN
- Oblig 2: Sirius DSL Editor for IFML +/- (indivual), WebRatio/IFML App2 UI (simple) (April 11th)
- SAE-4: UML Service Modeling, ServiceML,SoaML, REST, UML 2.0 Composition, MagicDraw
- MDE-4: Guest lecture: DSL and ThingML, Franck Fleurey) and Web Meet with project from Florida Atlantic University, FAU, Boca Raton, FL, USA (at 1700)
- SAE-5: MDE transformations, Non Functional requirements – OCL and PLanguage
- Oblig 3: SA Spec (More models), WebRatio/IFML App 3 Server (May 2nd)
- May (14,15,16): (Bringing it together)
- SAE-6: Final WebRatio App demo and discussion day (May 2nd)
- MDE-5: Enterprise Architecture, TOGAF, UPDM, SysML – DSLs etc. – Big picture
- MDE-6: Conclusions/Summary of the course
- Exam (4 hours), (June 6th)

Customer Journey & adapted BPMN





OMG Modeling languages and Zachman Framework



BPMN (Business Process Model and Notation)

See also:

INF5181 - **Process improvement and agile methods in systems development**

http://www.uio.no/studier/emner/matnat/ifi/INF5181/index-eng.html

and

http://en.wikipedia.org/wiki/Business_Process_Model_and_Notation#Elements

INF5181 - Process improvement and agile methods in systems development

ma. 31. aug.	12:15–15:00	Software process models, Software process modelling (BPMN)	OJD Seminarrom C	▪ D. Sjøberg ▪ Y. Lindsjørn	Lecture notes Photo for homework exercise Comparison of five standard software models Poster - BPMN Exercise - BPMN Suggested solution Vis mer...

The need of process modeling

- Process improvement is created with better understanding, communication, and organization
- Modeling is an important aspect of these
- Modeling translates verbal or tacit understanding into simple metaphors that assist these objectives
- A metaphor is a way of reducing the dimensions of the description of a process to a more understandable and visible basis
- Metaphors bridge complex concepts and build an understanding of the relationships between them

Three manners of thinking - Process

- Can be defined as an organization of activities that happen in a series, relevant to a business's goals and objectives
- At a fundamental level, a process diagram represents a single instance of a process
- For example, a purchase order process reflects an instance of a single purchase order, not an organization processing their work load of purchase orders

Three manners of thinking - Event

- From another perspective, a process is actually a connected sequence of events that respond to states, causes, and conditions
- In an event-based view, the process is a linkage of the transitions from one processing state to another

Three manners of thinking - Decision

- From yet another viewpoint, all activities and responses to events should be the result of a conscious decision by the organization.
- The decisions are an assemblage of business rules

- A process model is not merely a scenario;
- It is a scenario that exists within the context of the process, events, and decisions
- All these different perspectives are appropriately incorporated in a robust process model

Business Process

- A business process is a sequence of activities that carry out a business goal
  - “A business process is an organized, coordinated flow of activities, conducted by participants, acting on and deciding with data, information, and knowledge, to achieve a business goal”

Business rules

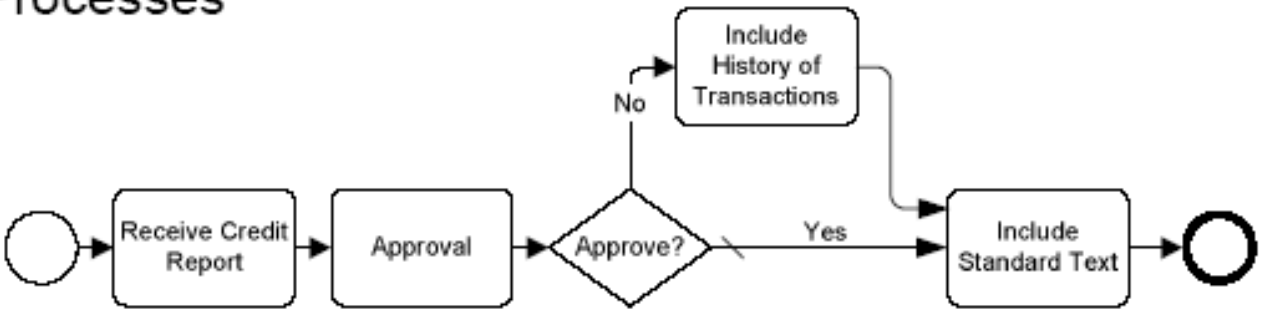
- A business rule is a mediator of information in computer systems for decision-making process participants, such as managers, employees, and salespeople
- More accurately, from the viewpoint of the business process:
  - “a business rule is an atomic logic step that uses data and knowledge to evaluate part of a proposition about a process decision”
- The business rule “meets” the process through the decision – when you change the business rule, you change decision outcome
- Think of a set of business rules as conditions that match data and create conclusions

Business Event

- In a modern process modeling approach, opportunities, conditions, and factors that events must respond to are handled or managed with business events
  - “A business even is an event that is meaningful for conducting commercial, industrial, and governmental, or trade activities”
- In BPMN, we have start, intermediary, non-interrupting, and end events
- Correspond directly to a process instance

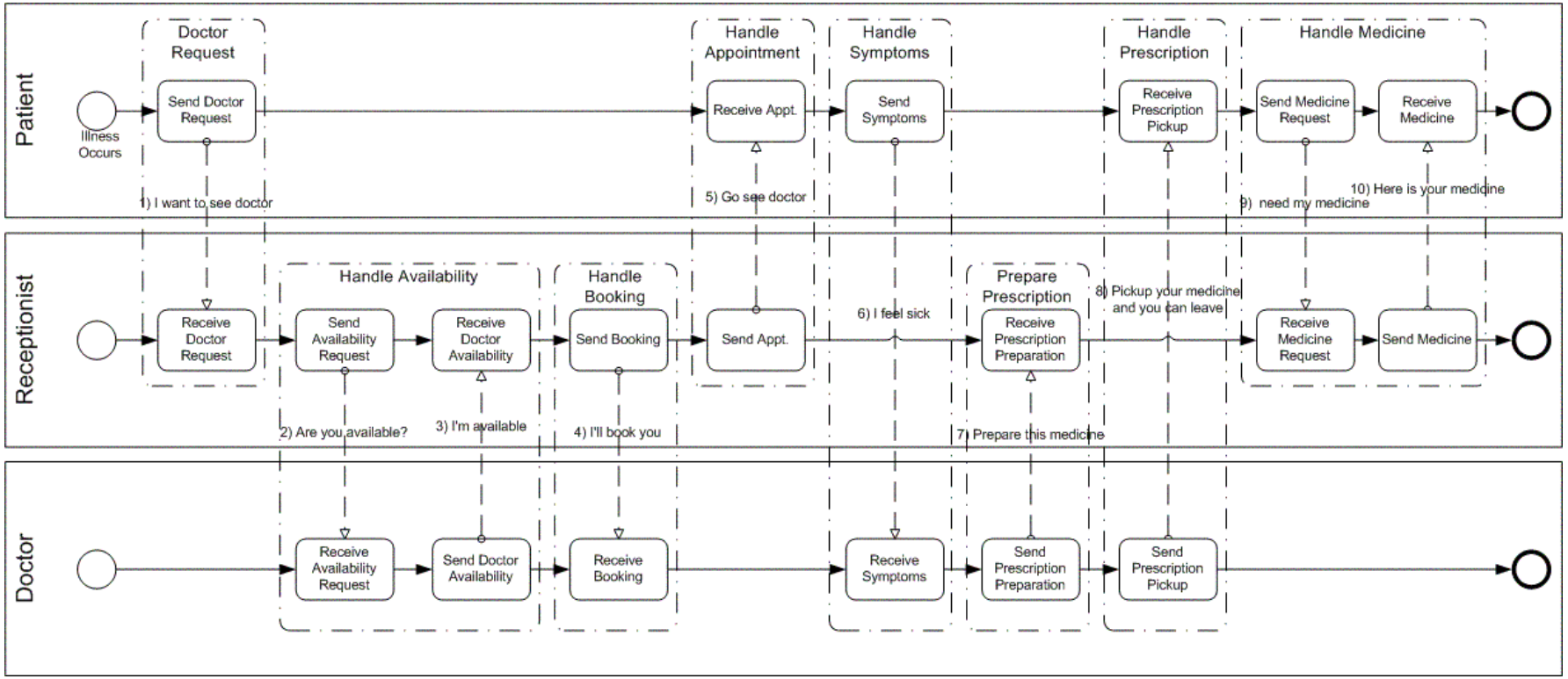
What is BPMN ?

- BPMN is flow-chart based notation for defining Business Processes



- 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 provides a mechanism to generate an executable Business Process (BPEL) from the business level notation
  - A Business Process developed by a business analyst can be directly applied to a BPM engine instead of going through *human* interpretations and translations into other languages

BPMN example



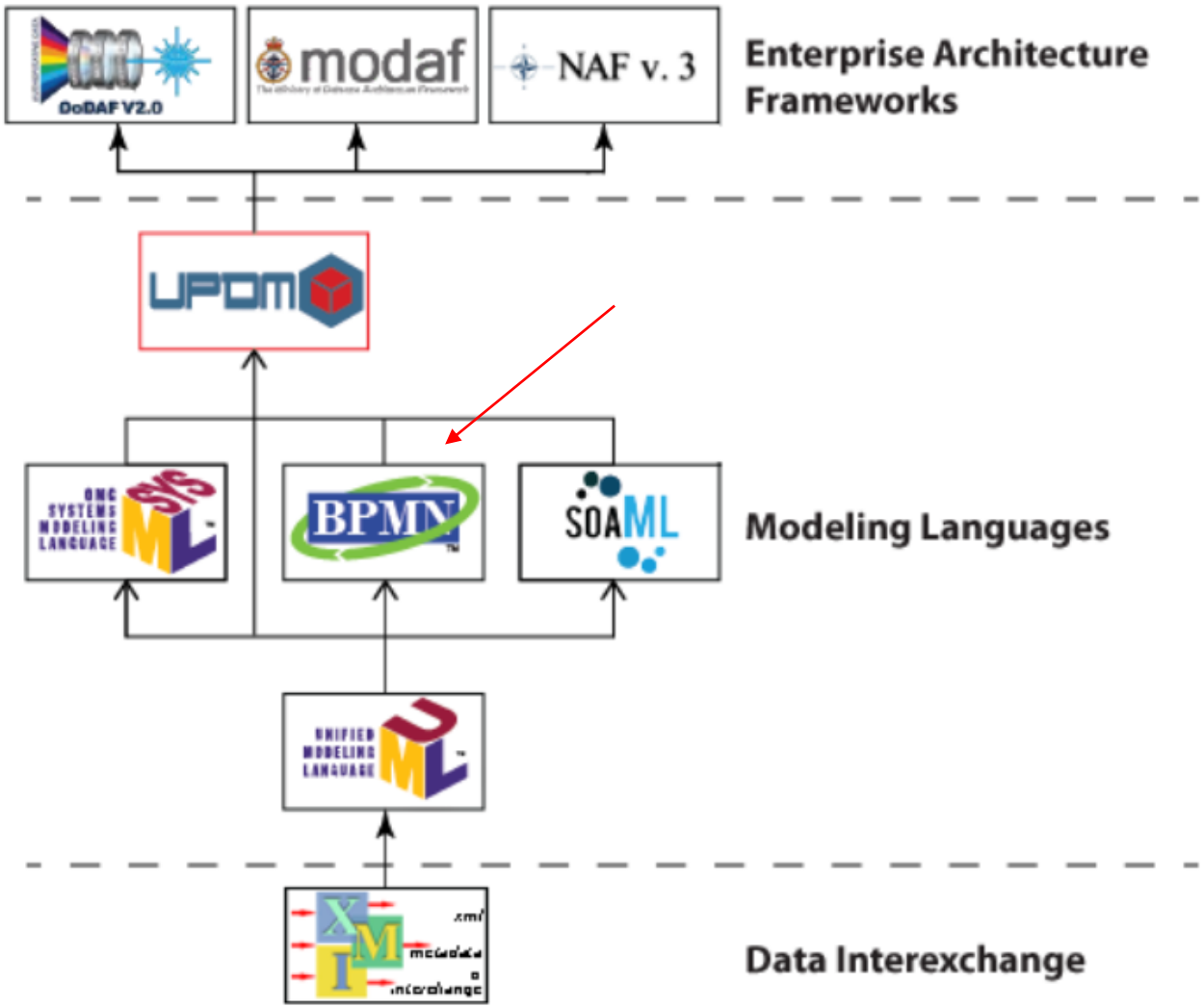
BPMN 2.0 and SoaML tools today

- BPMN 2.0
  - Signavio has 2.0 Conversation and Choreography diagrams – a SaaS solution
  - Most BPMN 1.2 are doing stepwise migration, making existing parts 2.0 compliant

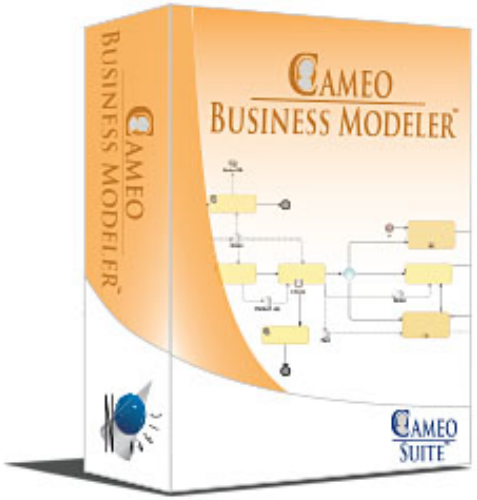
- SoaML (in most UML tools)
  - Magic Draw (Cameo), Enterprise Architect, IBM RSA/RSM, Modelio, …

  - We will use MagicDraw - Cameo Enterprise Architecture - version in the course in spring 2015

Cameo Enterprise Architecture



Cameo Business Modeler



Download FREE Cameo Business Modeler Analyst

INTRO | FEATURES | STANDARDS | EDITIONS | REQUIREMENTS | DEMOS | RESOURCES | RELATED

The world of business process management and modeling is changing. The Business Analyst needs more data and better access to Enterprise Architecture information. The Business Analyst is also a key member of the development team especially in light of the fact that Service Oriented Architecture (SOA) is a primary business driver.

At No Magic we believe that the Business Analyst should be able to use all data any time, in many different standards, all in one tool.

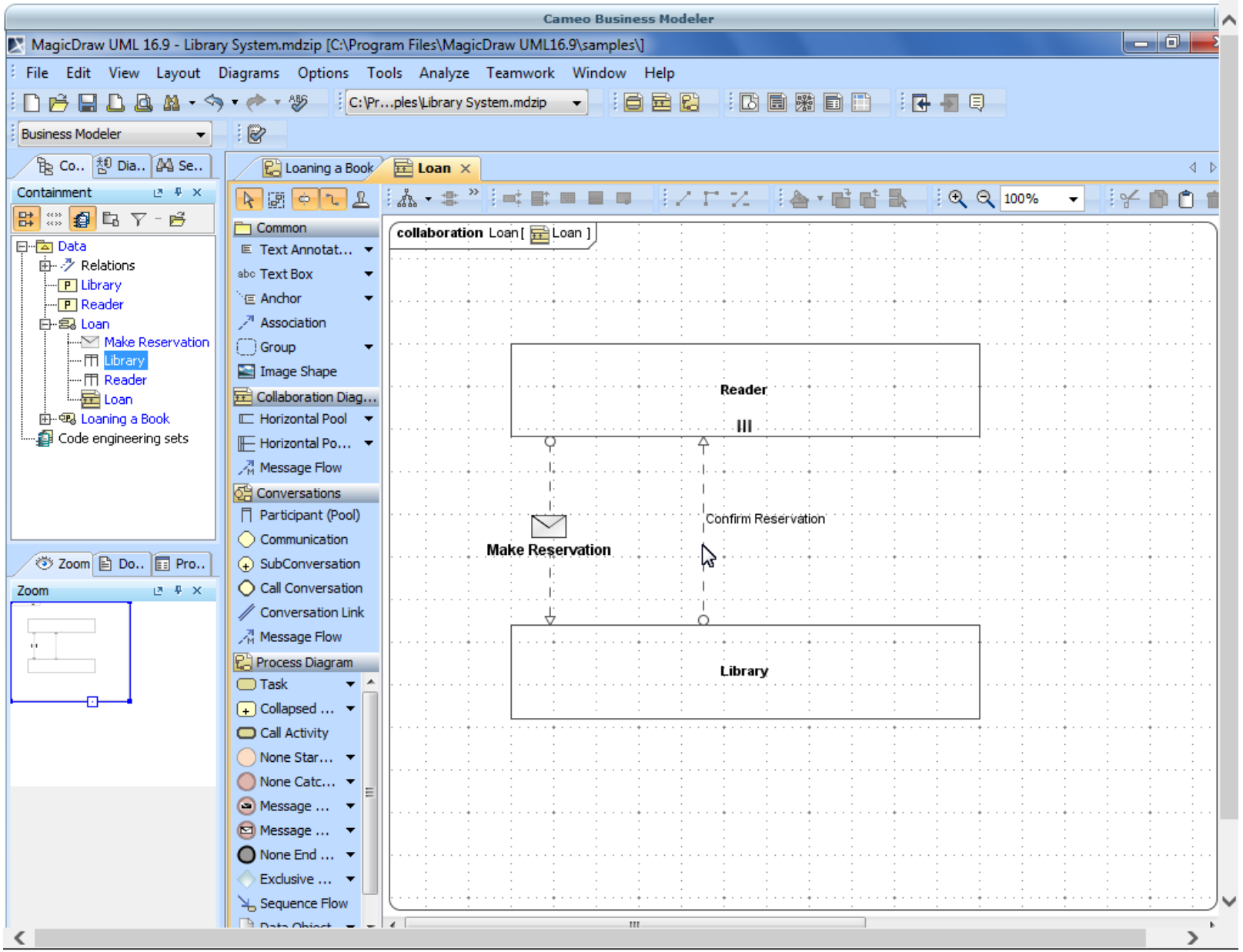
Cameo Business Modeler product family is the answer to these changes. We have integrated the latest business modeling standard (BPMN 2.0) into MagicDraw supporting the latest SOA and Enterprise Architecture standards.

Cameo Business Modeler for MagicDraw supports all the following:

- BPMN 2.0 symbols and diagrams
- Business model analysis tables and matrices
- Unprecedented ease of use of user perspectives
- Additional usability features specifically for BPM modeling
- User manual and BPMN samples
- Organization structure and business data diagrams

OBJECT MANAGEMENT GROUP BPM CERTIFICATION PROGRAM

No Magic is an OCEB sponsor - OMG Certified expert in BPM.



http://www.nomagic.com/products/cameo-business-modeler.html

WEBRATIO BPM PLATFORM

BUILD TAILORED BPM APPS WITH A FINE-GRAINED UX DEFINITION

In many cases, a BPM suite is too heavy and rigid to use for building a tailored BPM Web or mobile App, or for embedding a rich user interaction. WebRatio BPM Platform is designed to support you in building high-end BPM Web and mobile Apps with a tailored User Experience.

TRY IT NOW

WEBRATIO BPM PLATFORM FEATURES

- BPMN BASED
- BINDING BETWEEN BPMN AND IFML
- PLUGGABLE BPM ENGINE
- MOBILE BPM
- HIGH-PERFORMANCE BPM ENGINE
- RESTFUL API SUPPORT

Vacation Request BPM Tutorial

by Antonella Antico

Modeler | Beginner | Model Business Process | 666 views | Published on Oct 19, 2015

BPM | BPMN

Applies to: 8.0 or higher

Table of contents

Introduction
Step 1: Create the BPMN Project
Step 2: Define Process Actors
- Define Lanes
- Define Roles

Step 3: Define Process Parameters
Step 4: Define the BPM Process
- Define the "New Vacation Request" start event
- Define the "Register Vacation Request" user task
- Define the "Check Day Availability" service task
- Define the "Check Days Availability" exclusive gateway
- Define the "Approve Vacation Request" user task
- Define the "Check Request Approval" exclusive gateway
- Define the "Inform Reject Reason" user task
- Define the "Make Administrative Task" user task
- Define the "Vacation Request End" end event
- Add comments to the model

Step 5: Generate the BPM Engine API

https://www.webratio.com/learn/learningobject/vacation-request-bpm-tutorial-v-80

Your First Business Process

Getting Started with WebRatio BPM Platform

Modeler | Beginner | Model Business Process

219 views Published on Jul 24, 2015 | Time 7 min

BPM | BPMN | Process | Process Parameter | Task

Applies to: 8.0 or higher

Create your first Business Process application and discover the main features available for your application in WebRatio BPM Platform. Try yourself how easy and quick is to get your BPM application watching this tutorial.

Progress

x1

Transcript

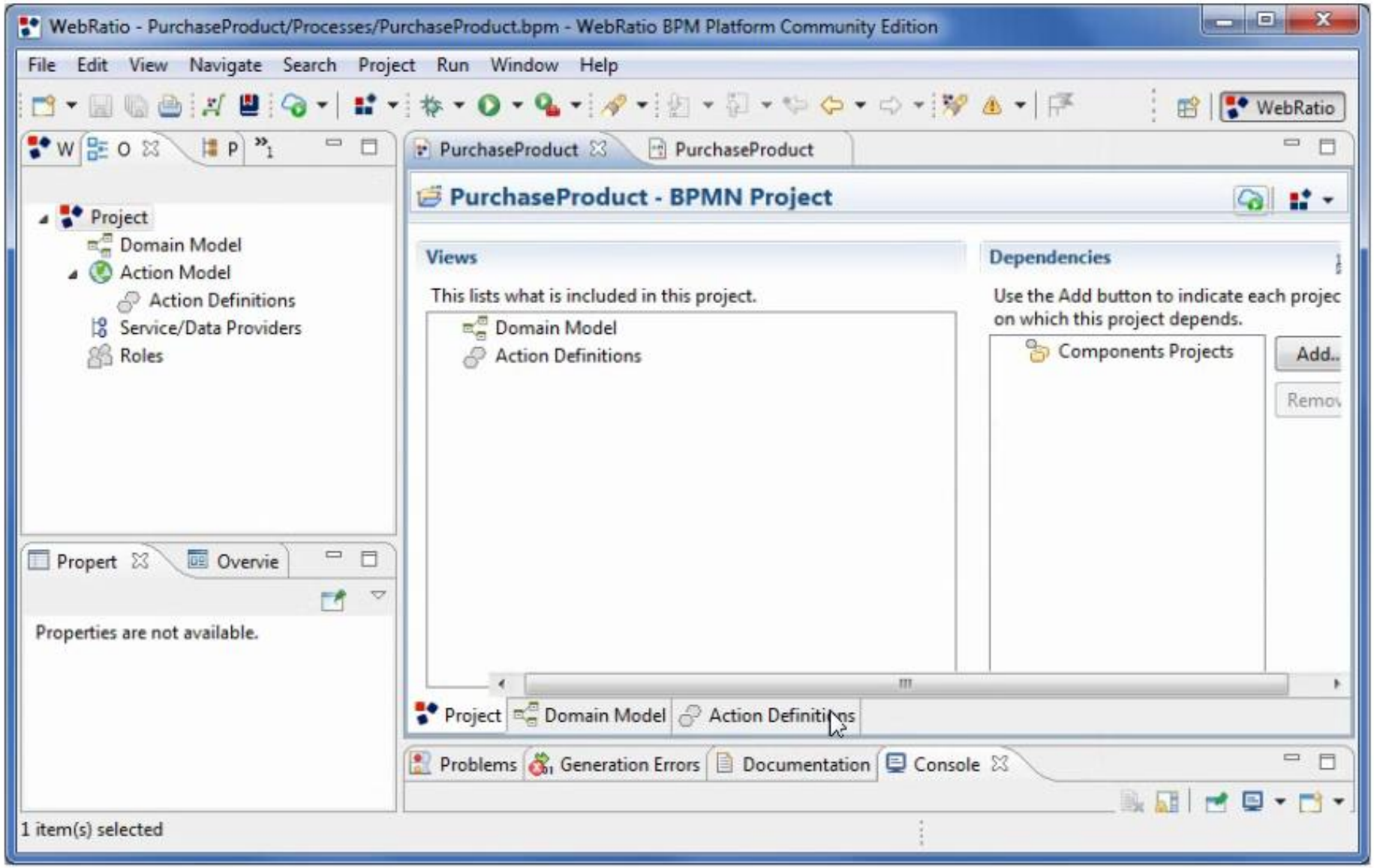
Table of contents

Introduction

How to Create a BPMN Project

How to Create a Business Process

Steps to Run a BPM Application


WebRatio - PurchaseProduct/Processes/PurchaseProduct.bpm - WebRatio BPM Platform Community Edition
File Edit View Navigate Search Project Run Window Help
WebRatio
Project
Domain Model
Action Model
Action Definitions
Service/Data Providers
Roles
Propert
Overvie
Properties are not available.
PurchaseProduct
PurchaseProduct
PurchaseProduct - BPMN Project
Views
This lists what is included in this project.
Domain Model
Action Definitions
Dependencies
Use the Add button to indicate each projec
on which this project depends.
Components Projects
Add..
Project
Domain Model
Action Definitions
Problems
Generation Errors
Documentation
Console
1 item(s) selected

Step 5: Generate the BPM Engine API

Once the business process is completely modeled, it's time to generate it. The generation process lets you get the BPM engine implementing the business process. Move to the Project View and press the "Generate and Run on Cloud" button. The BPM engine publishes a Rest API that can be used by Web and Mobile applications to run the business processes. The API also includes a set of administrative methods that can be used only by the business process administrator to manage the processes. At the end of the generation process, you ill gain access a Web representation of the engine's API, generated by Swagger. Using this Web interface, you can try the business process execution directly from your Web browser. To learn more, watch the Your First Business Process lesson.

BPMN History

- BPMN 1.0 (BPMI) – Mai 2004
- BPMN1.x
  - BPMN 1.1 (OMG) – Januar 2008
  - BPMN 1.2 (OMG) – Januar 2009
- BPMN 2.0 final Juni 2010
- http://www.omg.org/spec/BPMN/2.0/

History for BPMN

- The Business Process Management Institute (BPMI—now a part of the OMG) develops BPML (an XML process execution language) and realizes need for a graphical representation
  - BPML was later replaced by BPEL as the target execution language
- August, 2001, the Notation Working Group is formed. The group was composed of 35 companies, organizations, or individuals.
- BPMN 1.0
  - May, 2004, the BPMN 1.0 specification was released to the public.
  - February, 2006, BPMN 1.0 was adopted as an OMG standard
  - Currently, there are 39 companies that have implementations of BPMN

BPMI.org Hourglass

Audiences:

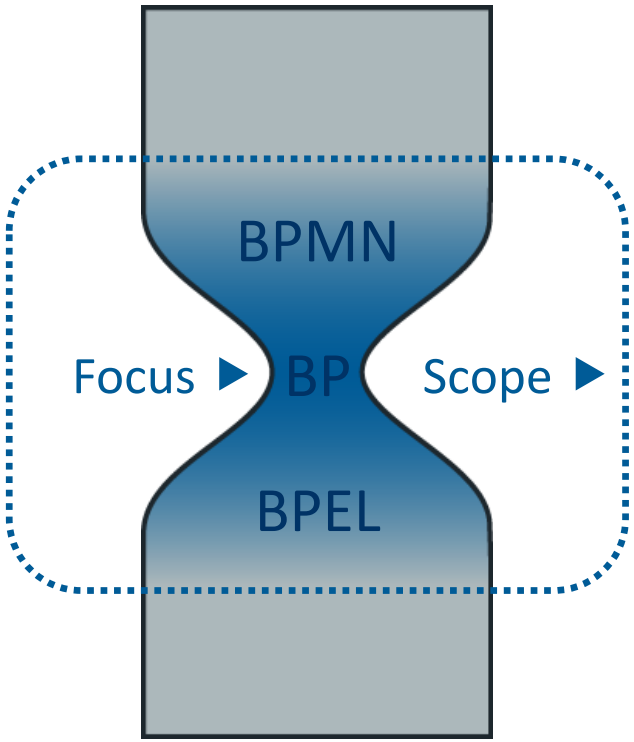
rategy Consultants

Business Analysts

Process Designers

System Architects

oftware Engineers



Purposes:

Modeling

Execution

BPMN requirements

- Must be acceptable and usable by the business community
- Must be able to generate executable processes (e.g., BPEL) through a BPMN Model (a combination of graphical elements and supporting information (attributes))
- Although executable processes triggered the development of BPMN, it was expected that BPMN would be used for more general business purposes
- BPM is intended to be Methodology Agnostic
  - Methodologies will give guidance as to the purpose and level of detail for modeling
  - BPMN is as complex as it needs to be. Just use what you need…

BPMN Building blocks of the foundation

- Some of the concepts are part of the definitions of business process
  - “An event-activated flow of coordinated activities, conducted by participants, and acting on and deciding with data, information, and knowledge that achieve a goal”
- Participant, Activity, Flow, Process event, Data

Participant

- A participant is an actor or a person that interacts in a process
- The actor includes any human, digital, or virtual resource that involved in a business process

Participant examples

"People" participants:

- Inventory receipt clerk inspecting the order
- Employee filling out a request
- Patient in hospital
- Manager approving a requisition
- Technician restoring a disk drive

"System" Participants:

- SAP, PeopleSoft
- DB server
- Rules engines
- A Web service
- A custom-build UI
- A telephony queuing switch

Activity

- An “activity” is work the participant performs with business process
- Is the basic units of process work, can be
  - Atomic (lowest level, indivisible unit of work)
  - Non-atomic (involving many steps)
- Process and subprocesses are compound activities
- In BPMN, the types of process activities include:
  - Tasks – is the atomic activity
  - Subprocesses – is compound activity, might contain other activities

Activity examples

- An activity can be manual, as a human participant completes the activity, or
- It might be automated by a system participant
- Examples:
  - Inspecting material delivery
  - Restoring a server
  - Completing contract requisition
  - Reviewing and approving a requisition
  - Reviewing loan application

Flow

- Is the order (and data) in which the activities or process steps are performed
- Multiple flows might occur within multiple participants roles
- Two types of flows in BPMN diagram
- Sequence – defines the order in which activities are performed for any given process participants
  - Sequence flow never occurs between participants in different pools
- Message – defines the flow of information and message between participants within a process
  - Messages never occur within the same pool

Flow: transition

- Describes the hand-off between activities
- Transition means that one activity has stopped and another has started
- Transition never occurs between multiple participants
- e.g,a work area with people and workstations for each person's activities (tasks)
  - As each task is completed, the person transitions to the next task at another workstation
- Any communication is an interaction, not a transition

Flow: interaction

- Is the communication between participants
- Interactions occur between two or more participants in the form of message
- Interactions never occur from one participant back to itself
- Note: a flow from one participant back to itself is an activity transition, not an interaction

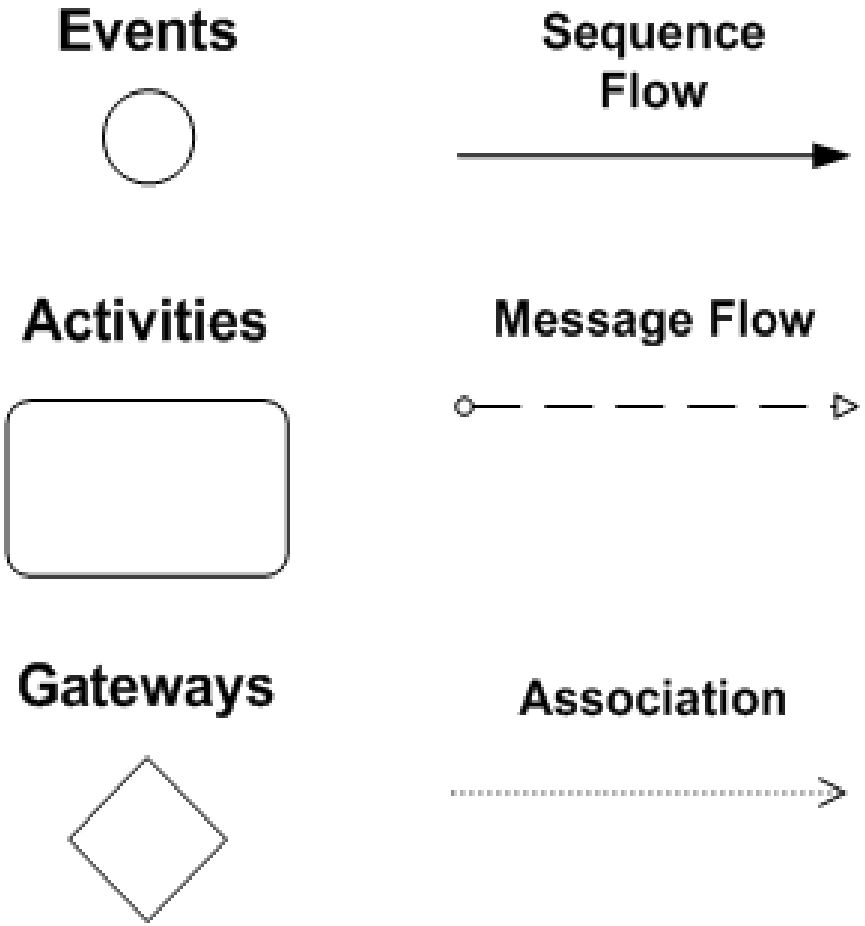
Process Event

- An event is something that happens
- A process event defines a point where the process is either started, stopped, halted, or continued
- Events define occurring activities “of interest”
- Participant actions, choices, or activities define or create events
- Examples:
  - Contract order submitted
  - DB unavailable
  - Requisition rejected

Data

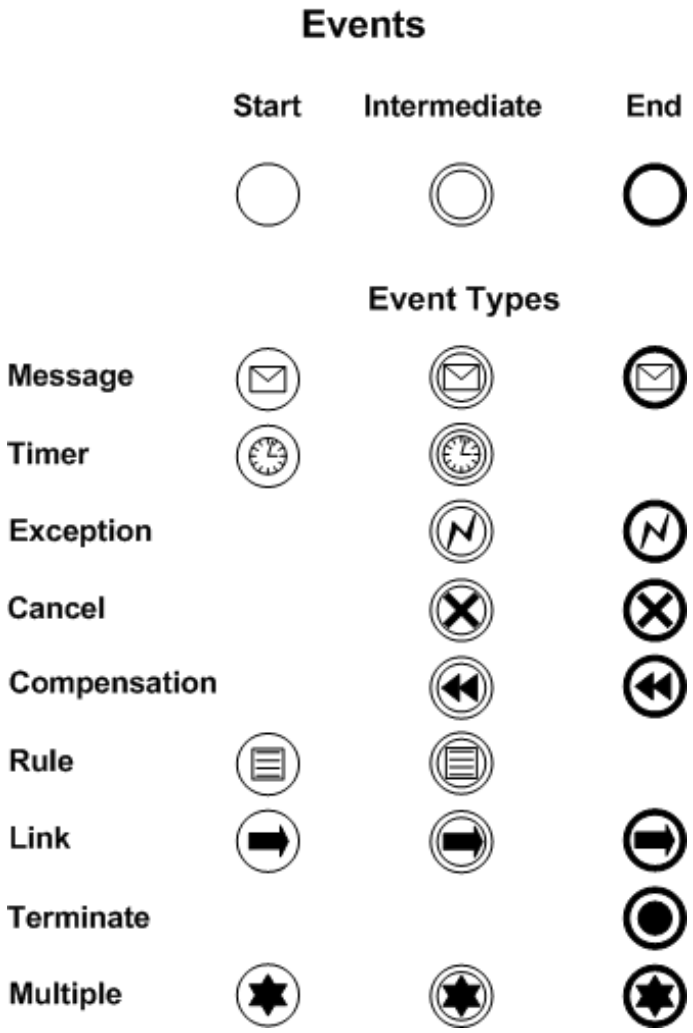
- Data shapes in BPMN are artifacts, meaning, or an effect of process events occurring
- Data is never a cause of process activity occurring. Events trigger activity, resulting in data
- Data mostly originates from events, for example:
  - An airplane is cleared for a final approach. This event is added to the flight log (data)
  - The log data is a chorological series of event snapshots

Core Set of Diagram Elements



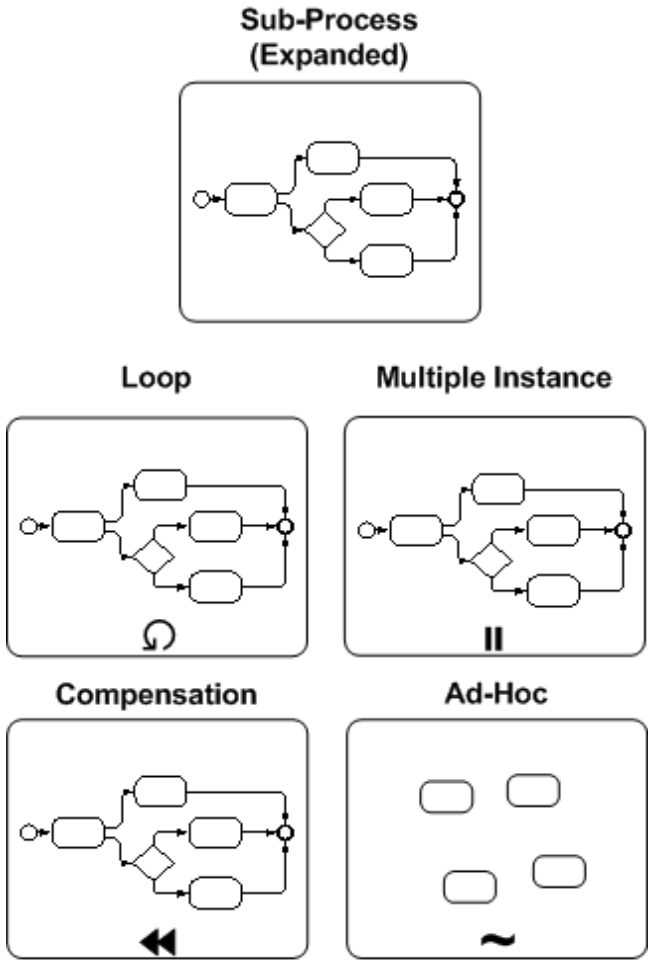
- The core set of modeling elements enable the easy development simple Business Process Diagrams that will look familiar to most Business Analysts (a flowchart diagram)

Complete Set of Diagram Elements, Events



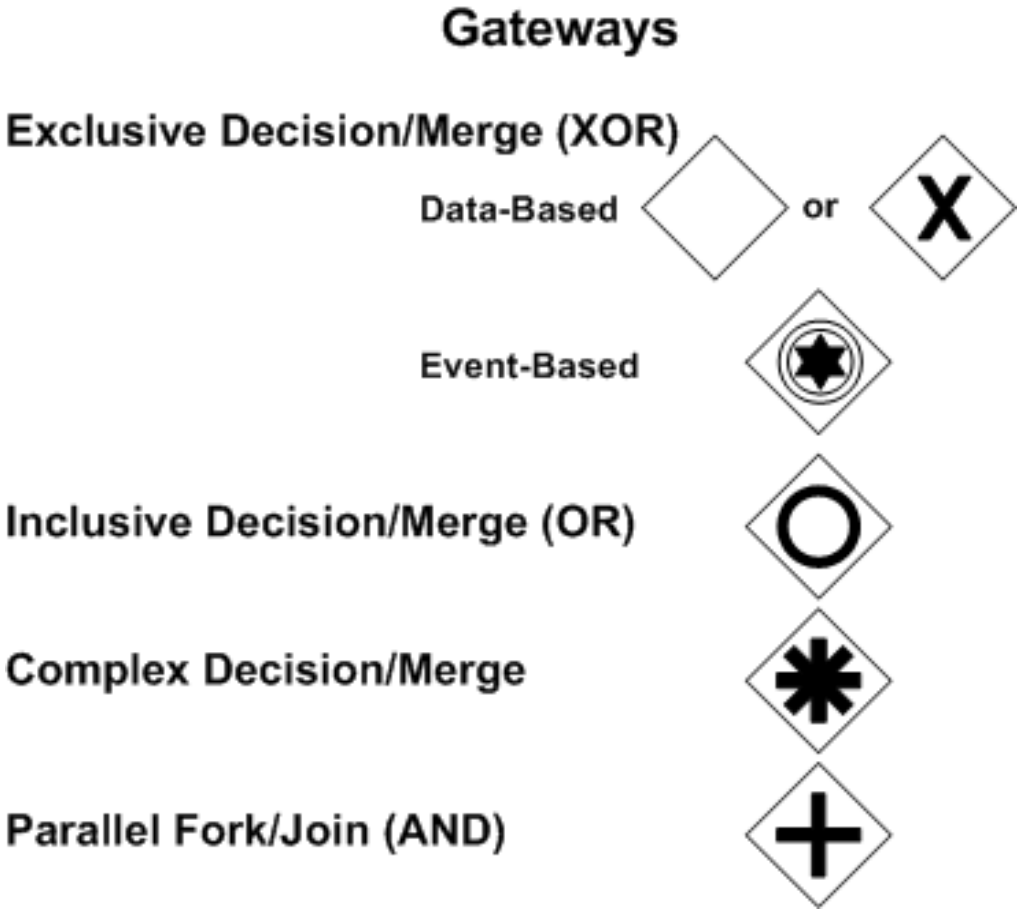
- An Event is something that "happens" during the course of a business process. These Events affect the flow of the Process and usually have a trigger or a result. They can start, interrupt, or end the flow.

Complete Set of Diagram Elements, Activities, Cont.



- A Sub-Process can be in an expanded form that shows the process details of the a lower-level set of activities.

Complete Set of Diagram Elements, Gateways



- Gateways are modeling elements that are used to control how Sequence Flows interact as they converge and diverge within a Process. If the flow does not need to be controlled, then a Gateway is not needed.

BPMN Diagram elements

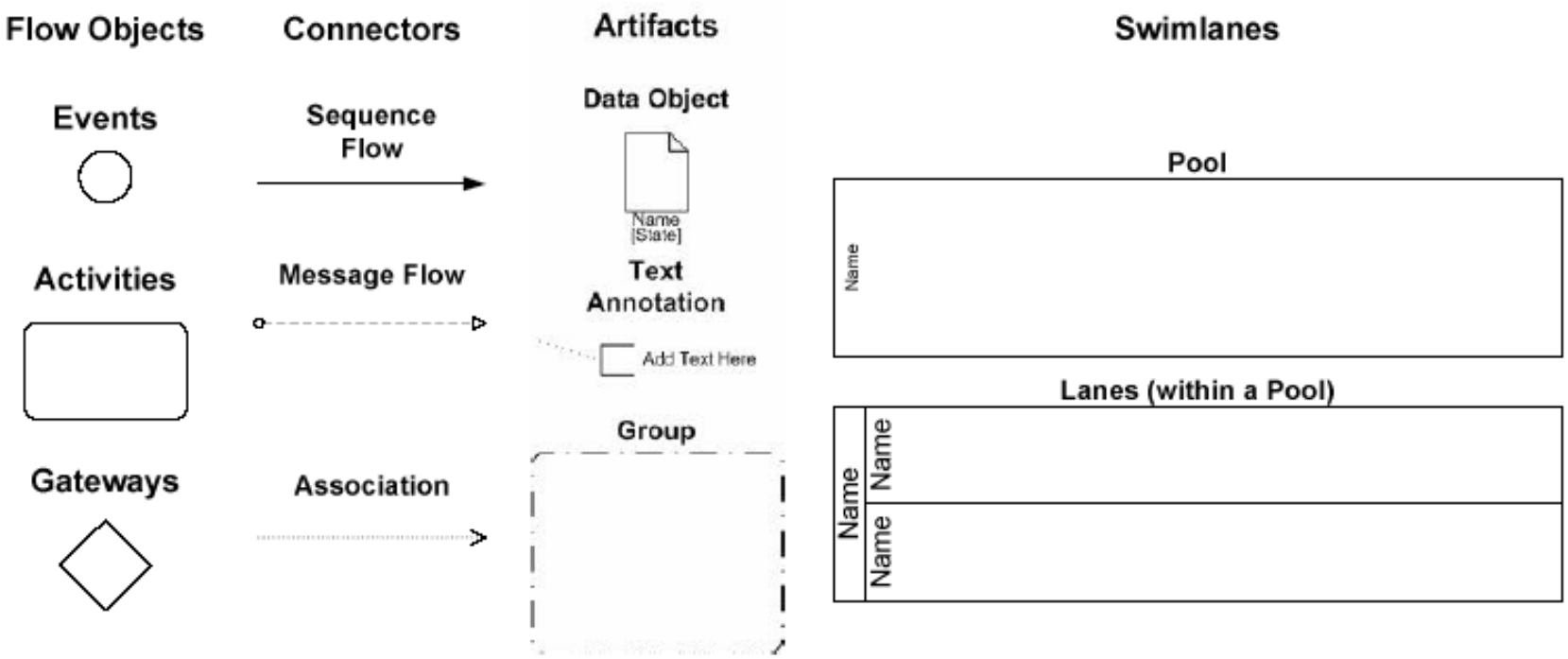
Activities

Events

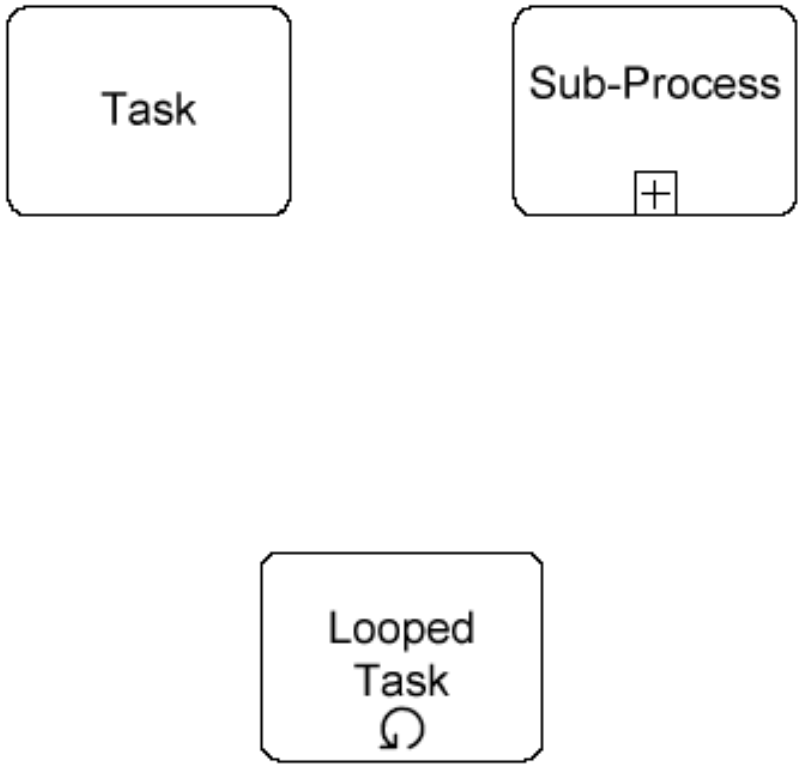
Gateways

Connectors

Diagram elements (2)



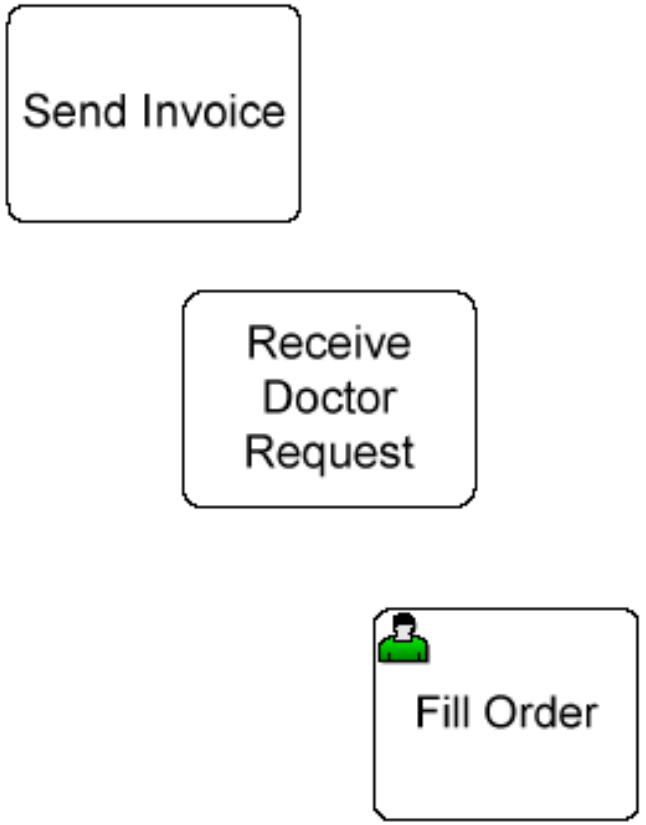
Activities



- An activity is work that is performed within a business process. An activity can be atomic or non-atomic (compound). The types of activities that are a part of a Process Model are: **Sub-Process**, and **Task**
- Activities are rounded rectangles
- They can be performed once or can have internally defined loops

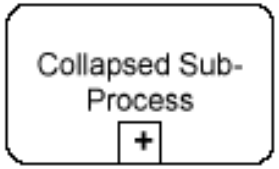
Task

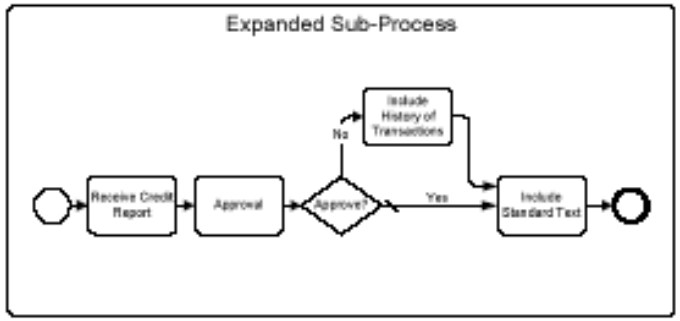
- A Task is an atomic activity that is included within a Process. A Task is used when the work in the Process is not broken down to a finer level of Process Model detail
- There are specialized types of Tasks for sending and receiving, or user-based Tasks, etc.
- Markers or icons can be added to Tasks to help identify the type of Task
  - Markers must not change the footprint of the Task or conflict with any other standard BPMN element



Sub-processes

- Sub-Processes enable hierarchical Process development
- A Sub-Process is a compound activity that is included within a Process. It is compound in that it can be broken down into a finer level of detail (a Process) through a set of sub-activities
- For a collapsed version of a Sub-Process, The details of the Sub-Process are not visible in the Diagram. A “plus” sign in the lower-center of the shape indicates that the activity is a Sub-Process and has a lower-level of detail.
- For an expanded version of a Sub-Process, the details (a Process) are visible within its boundary.
- There are two types of Sub-Processes: Embedded and Independent (Re-usable)





Events

- An **Event** is something that "happens" during the course of a business process. These Events affect the flow of the Process and usually have a 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 Events

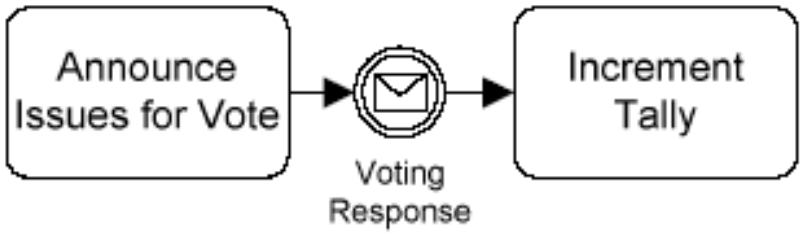
- Start Events indicate where a Process will begin
- There are different “Triggers” that indicate the specific circumstances that start the Process
  - None Start Events are used to mark the start of Sub-Processes or when the start is undefined
  - The Link Start Event will be removed in the next version of BPMN
  - Any one of the Triggers included in a Multiple Start Event will start the Process

Intermediate Events

- Intermediate Events occur after a process has been started and before a process is ended
- There are different “Triggers” that indicate the specific circumstances of the Event
- They can be placed in the normal flow of the Process or attached to the boundary of an activity

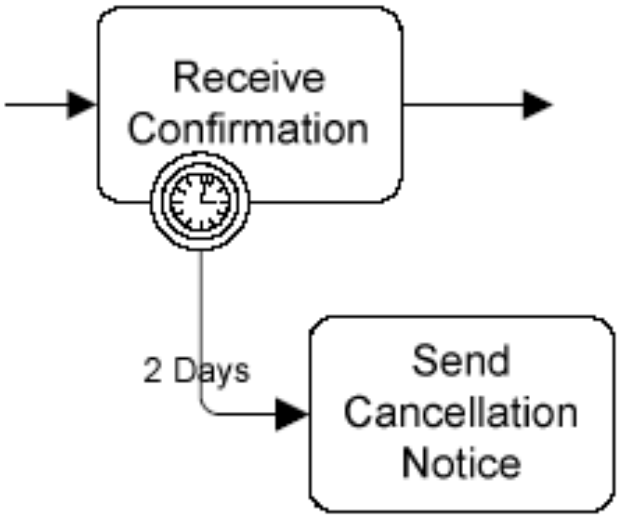
Intermediate events (normal flow)

- 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)
- They can represent the creation of the Event (i.e., the sending of a message)



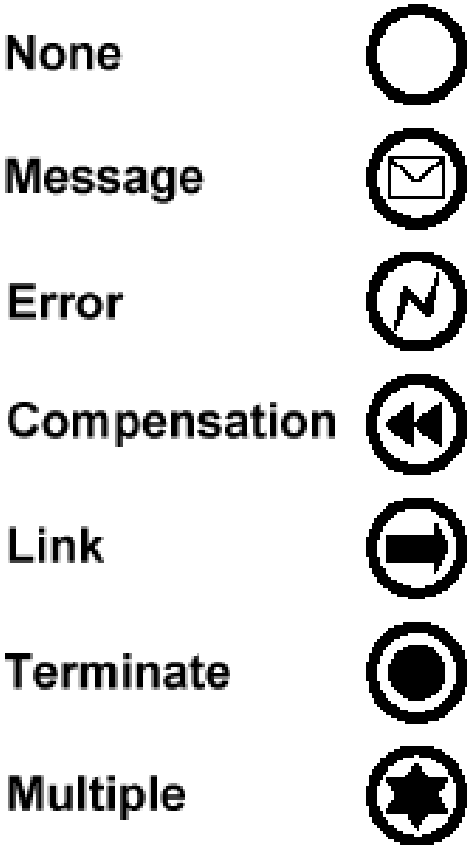
Intermediate events (linked to Boundary)

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

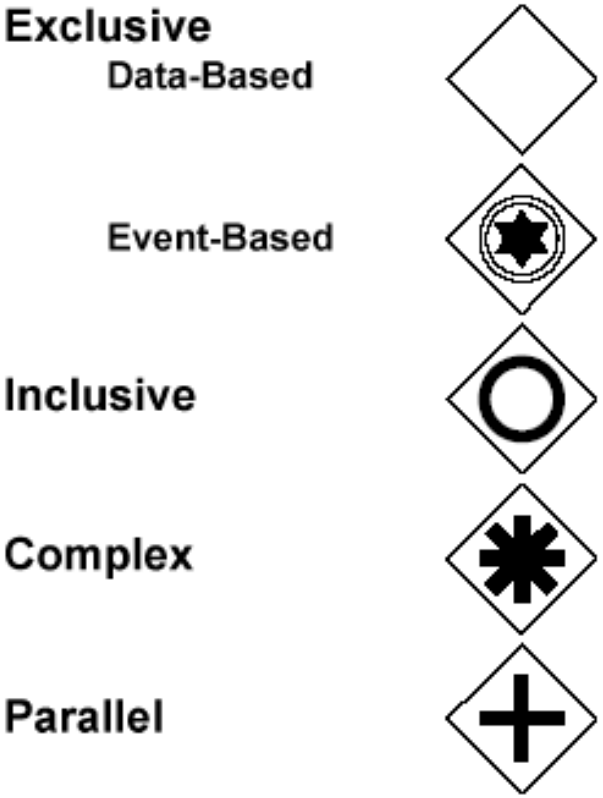


End events

- End Events indicates where a process will end
- There are different “Results” that indicate the specific circumstances that end the Process
  - None Start Events are used to mark the start of Sub-Processes or when the start is undefined
  - The Link End Event will be replaced in the next version of BPMN (probably with a Signal)



Gateways



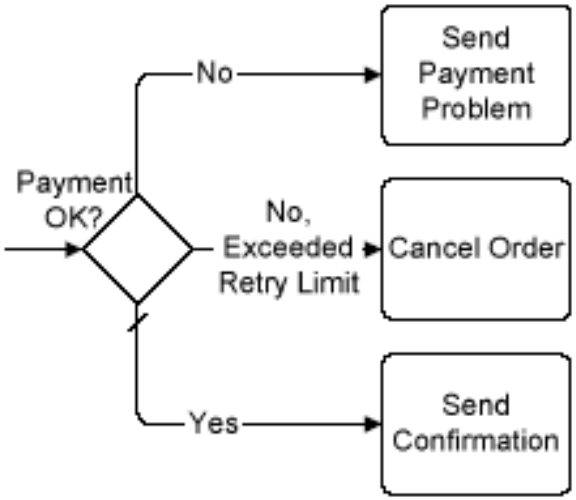
- **Gateways** are modeling elements that are used to control how Sequence Flows interact as they converge and diverge within a Process
- All types of Gateways are diamonds
  - Different internal markers indicate different types of behavior
  - All Gateways both split and merge the flow
- *If the flow does not need to be controlled*, then a Gateway is not needed. Thus, a diamond represents a place where control is needed

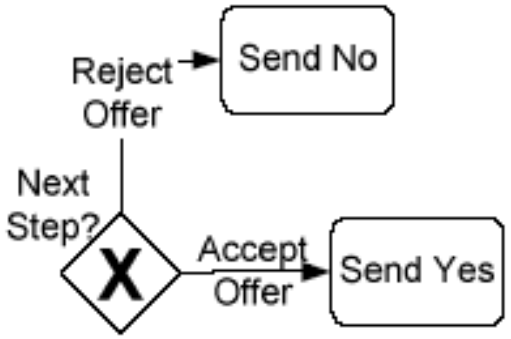
Exclusive Gateways

- Exclusive Gateways (Decisions) are locations within a business process where the Sequence Flow can take two or more alternative paths. This is basically the “fork in the road” for a 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)
- They are also used to merge Sequence Flow
  - The merging behavior may change in the next version of BPMN

Exclusive Gateways, based on data

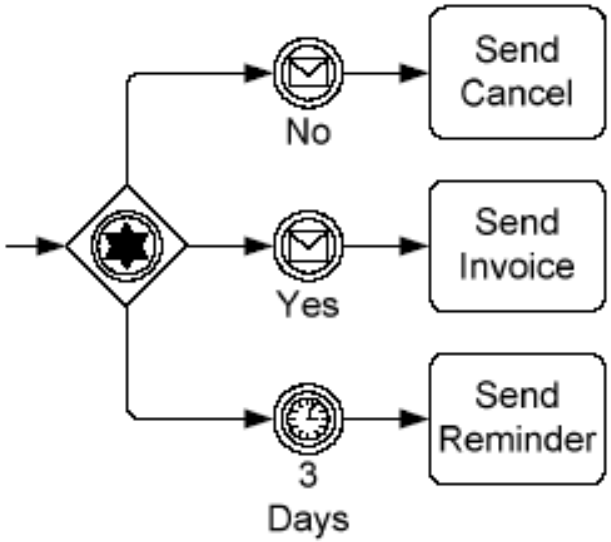
- These are the most commonly used type of Gateways.
  - They can be shown with or without an internal “X” marker. Without is the most common use.
- The Gateway (Decision) creates alternative paths based on defined conditions





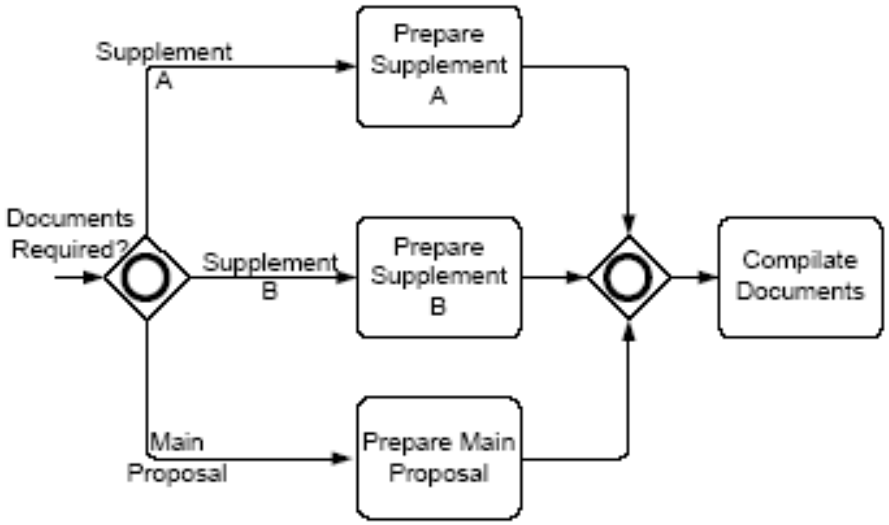
Exclusive Gateways, based on events

- This type of Decision represents a branching point in the process where the alternatives are based on events that occurs at that point in the Process, rather than conditions
- The Multiple Intermediate Event is used to identify this Gateway
- The Event that follow the Gateway Diamond determine the chosen path
  - The first Event triggered wins



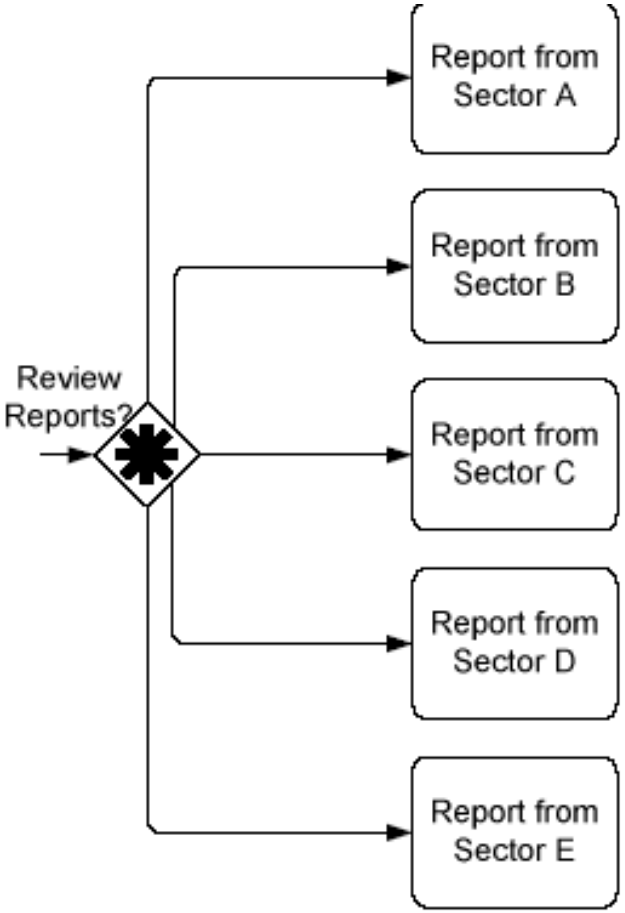
Inclusive Gateways

- 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



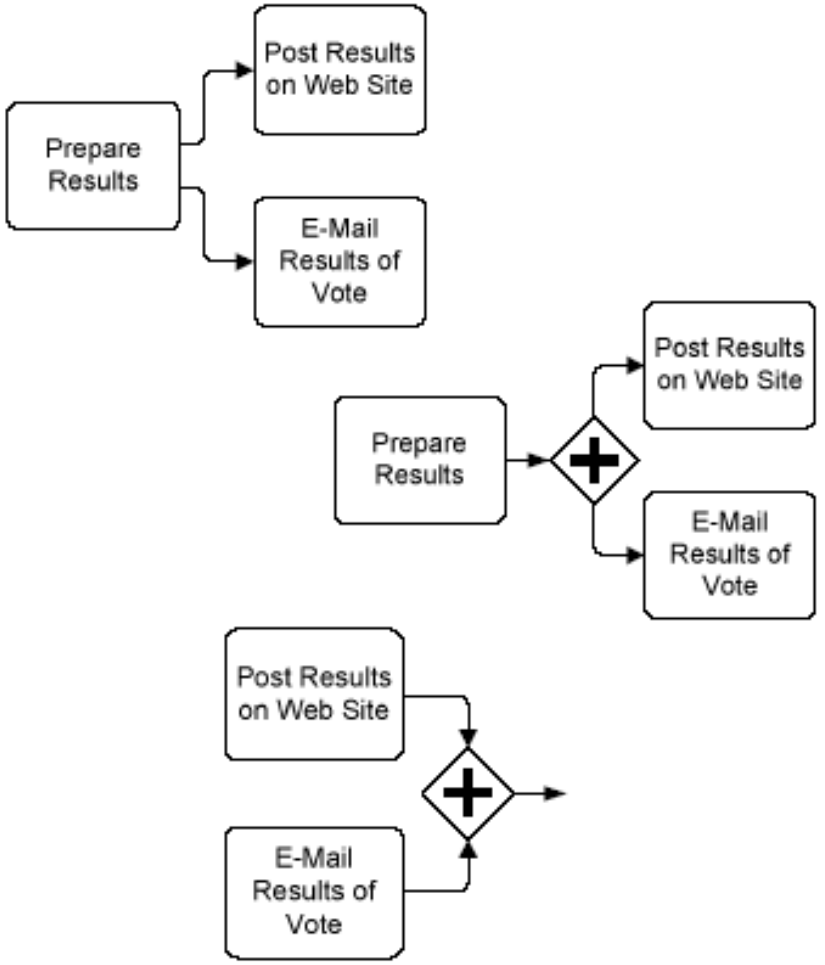
Complex Gateways

- Complex Gateways are Decisions where there is more advanced definitions of behavior can be defined
- The asterisk marker is used to identify this Gateway
- Complex behavior can be defined for both the merging and splitting behavior



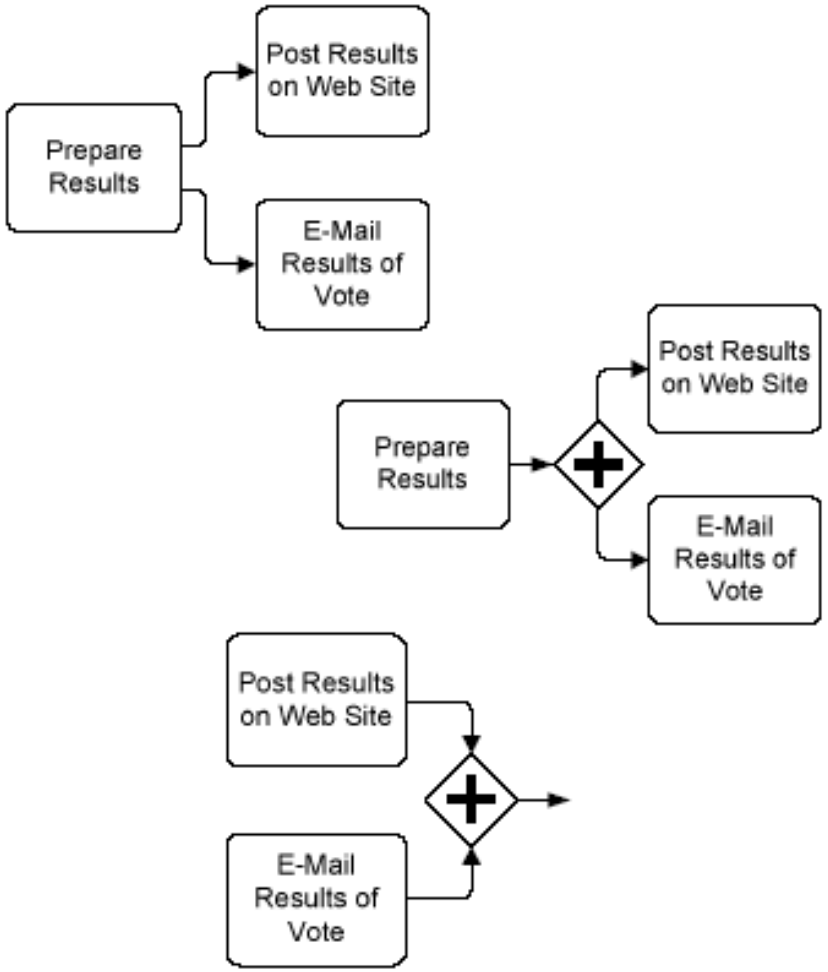
Complex Gateways

- Parallel Gateways are places in the Process where multiple parallel paths are defined
  - They are not required for forking in most situations.
  - They can be used for methodological purposes
- The "+" marker is used to identify this Gateway
- The Gateway is also used to synchronize (wait for) parallel paths



Parallell Gateways

- Parallel Gateways are places in the Process where multiple parallel paths are defined
  - They are not required for forking in most situations.
  - They can be used for methodological purposes
- The “+” marker is used to identify this Gateway
- The Gateway is also used to synchronize (wait for) parallel paths

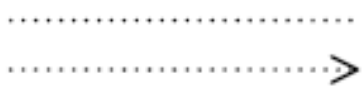


Conectors

Sequence Flow

Message Flow

Association



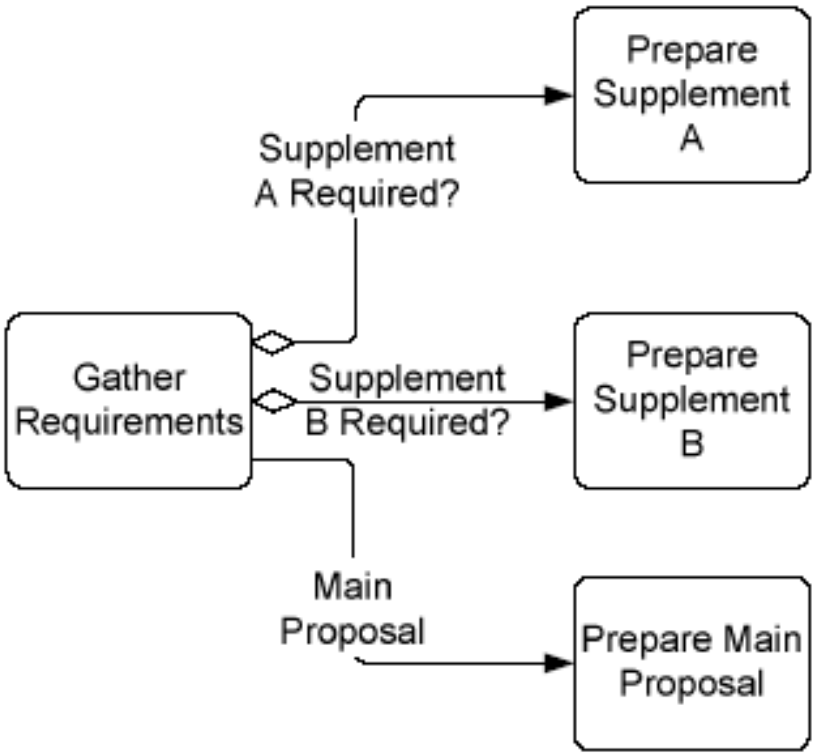
- A **Sequence Flow** is used to show the order that activities will be performed in a Process
- A **Message Flow** is used to show the flow of messages between two entities that are prepared to send and receive them
- An **Association** is used to associate data, information and artifacts with flow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

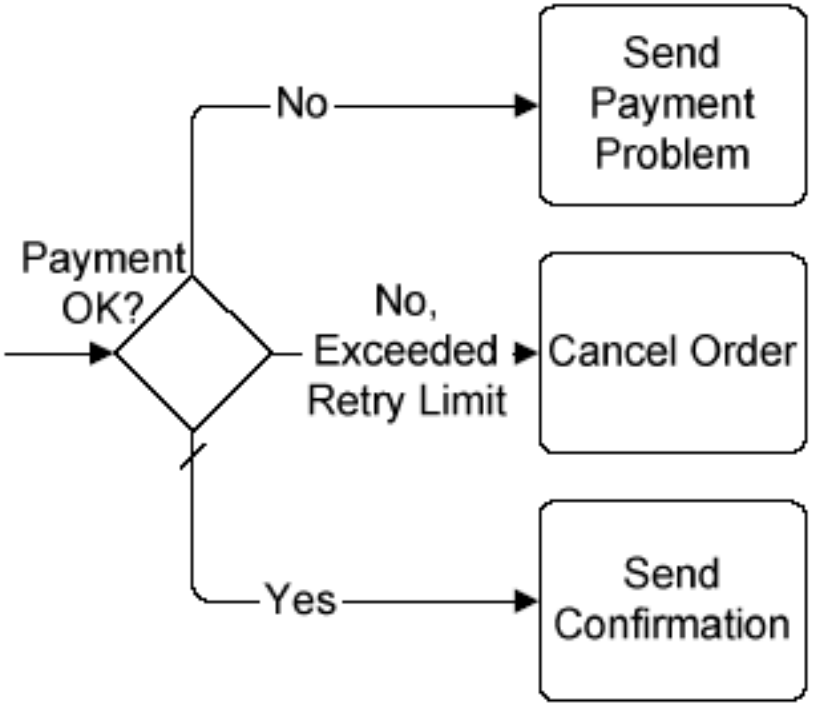
Conditions in sequence flow

- A Sequence Flow MAY have a defined condition if it exits an activity
  - Such an activity must have at least two Sequence Flows
- The condition has to be True to allow the flow to continue down the Sequence Flow
  - A mini-diamond shows that the Sequence Flow has a condition
- At least one of the outgoing Sequence Flow must be chosen during Process performance



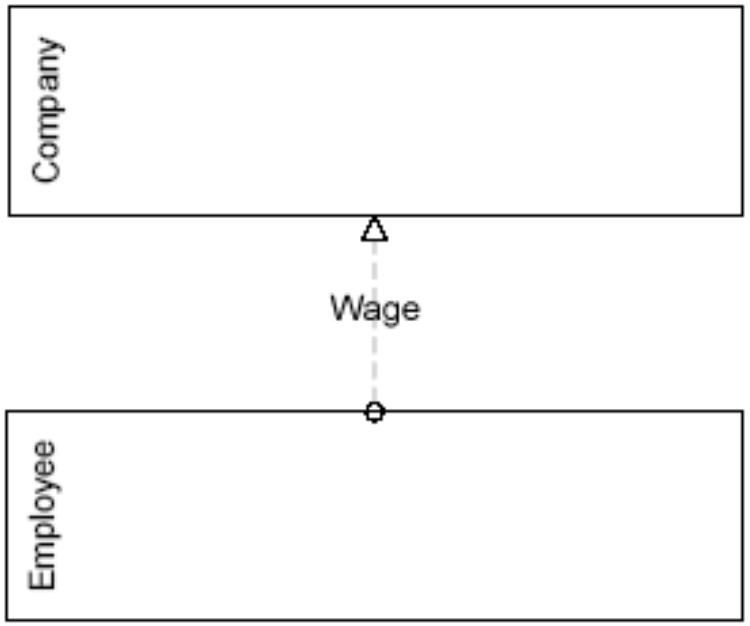
Default sequence flow

- A Sequence Flow that exits an Exclusive or Inclusive Gateway may be defined as being the default path
  - A hatch mark at the line beginning shows the default Sequence Flow
- The default path is chosen only if all the other conditions of the Gateway are False



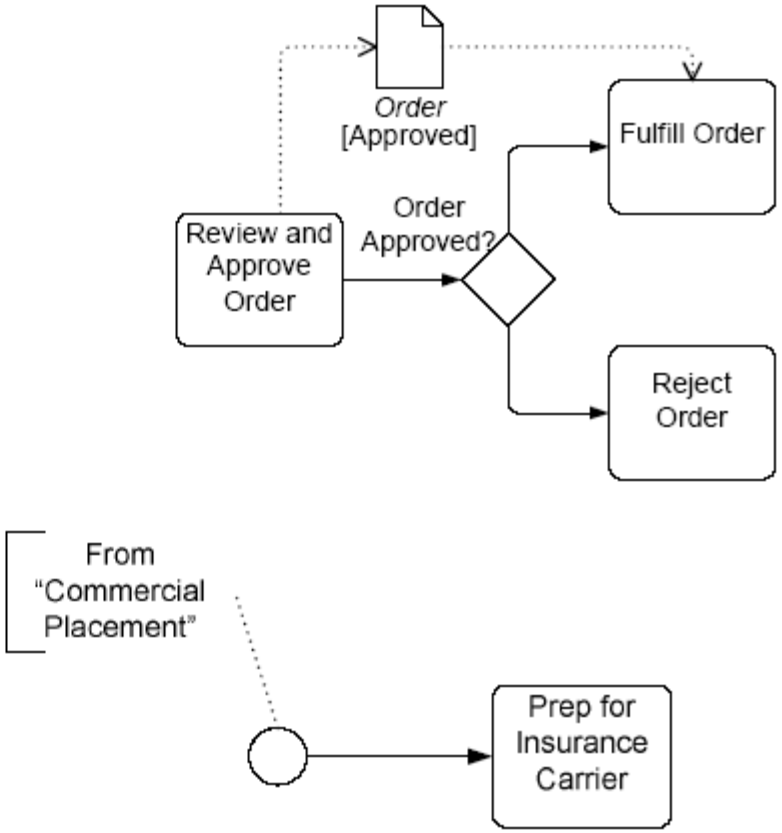
Message flow

- A Message Flow is used to show the flow of messages between two Participants of Process
  - In BPMN, separate Pools are used to represent the Participants
- A Message Flow can connect to the boundary of the Pool or to an object within the Pool
- Message Flow are not allowed between objects within a single Pool



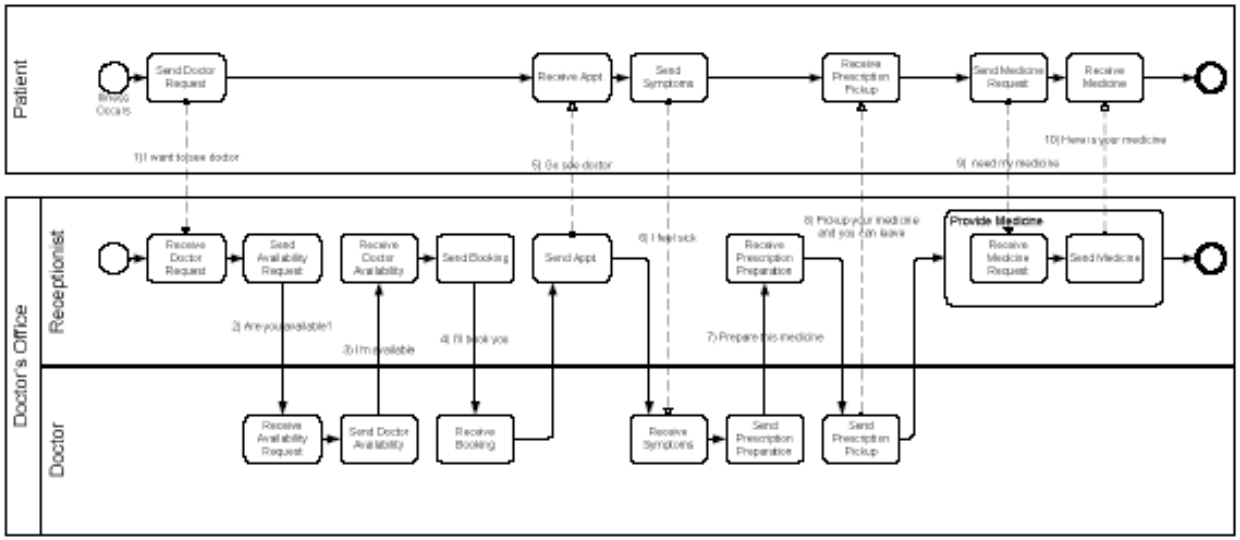
Associations

- 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



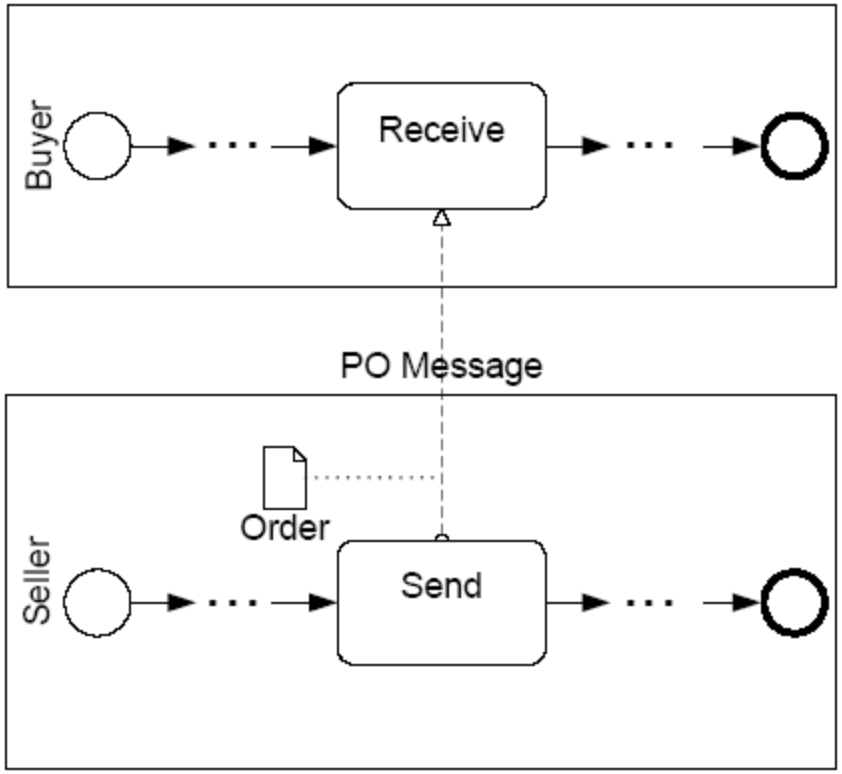
Swim lanes

- 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



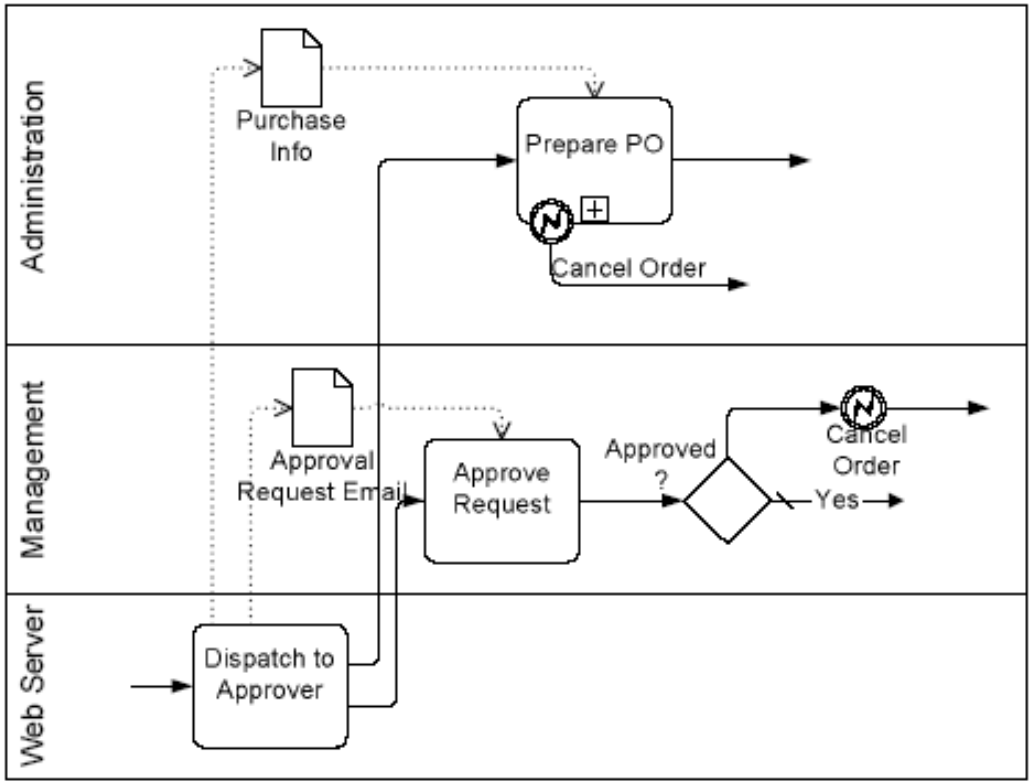
Pool

- 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

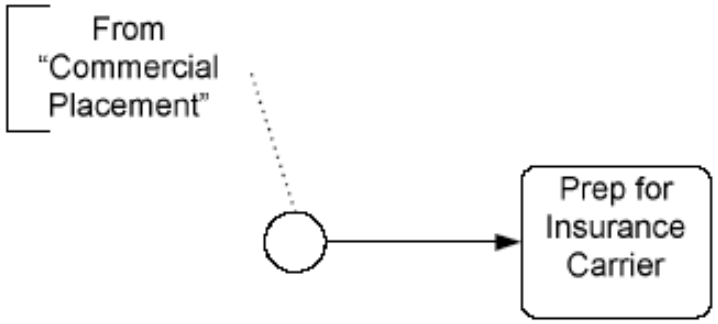


Artifacts

- Artifacts provide the capability to show information beyond the basic flow-chart structure of the Process
- There are currently three standard Artifacts in BPMN: Data Objects, Groups, and Annotations
  - Additional Artifacts may be standardized in later version
  - Sets of vertical market Artifacts may also be developed
- A modeler or tool can extend BPMN by defining new Artifacts

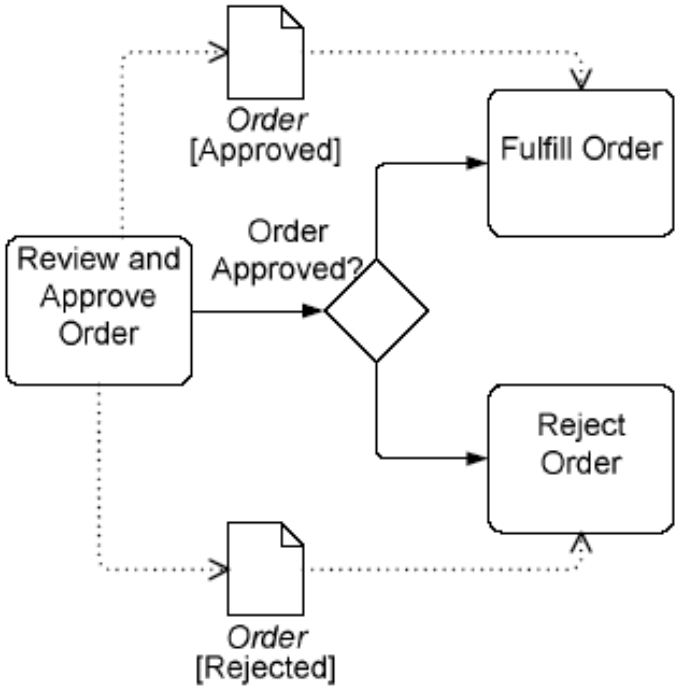
Text annotations

- Text Annotations are a mechanism for a modeler to provide additional information about a Process
- Text Annotations can be connected to a specific object on the Diagram with an Association



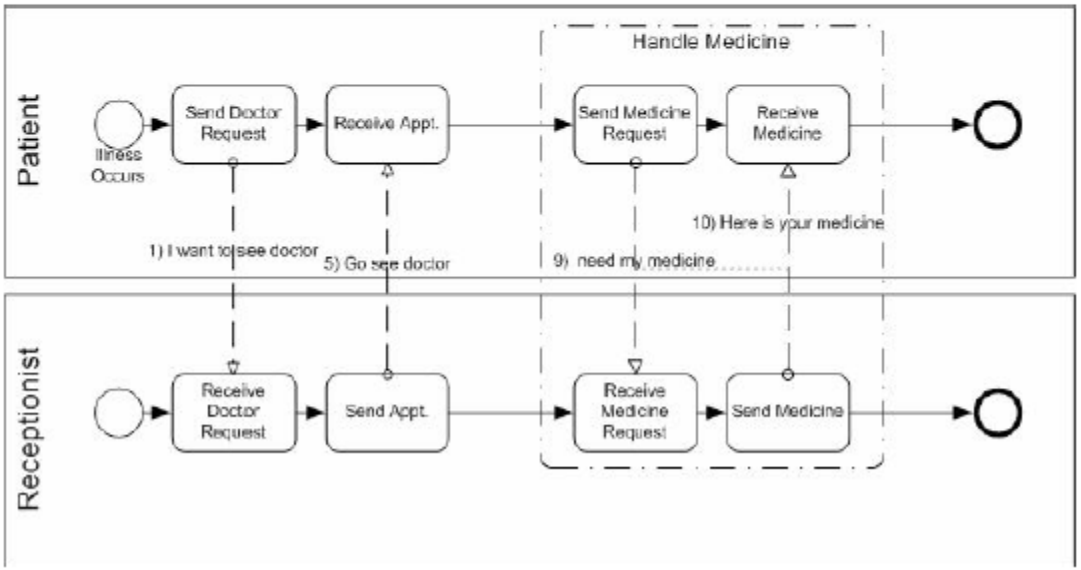
Data objects

- 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



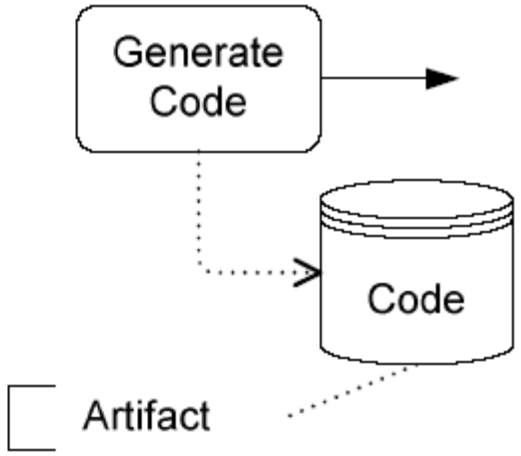
Groups

- Groups are Artifacts that are used to highlight certain sections of a Diagram without adding additional constraints for performance – as a Sub-Process would
  - Groups can be used to categorize elements for reporting purposes
- Groups are not constrained by restrictions of Pools and Lanes



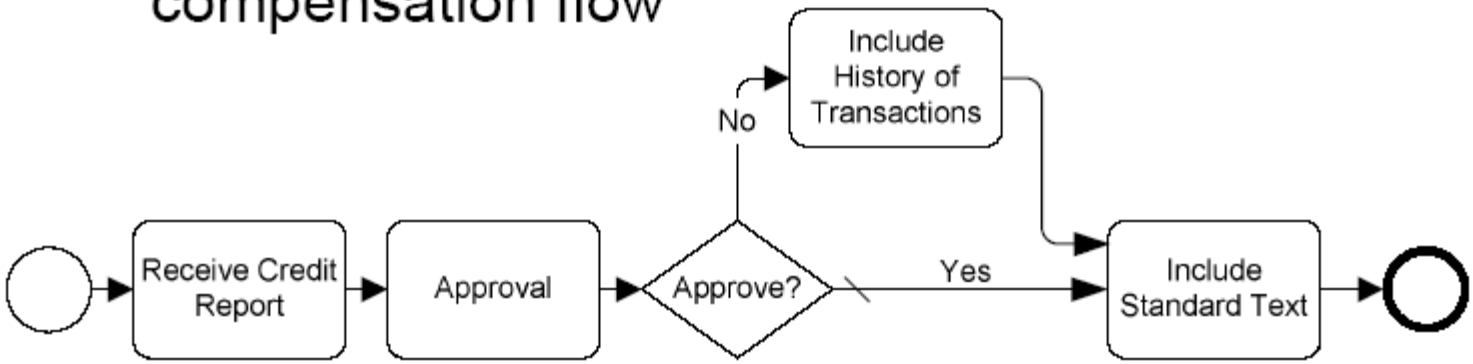
Extended artifacts

- Modelers and Modeling Tools can add new Artifacts to a diagram
  - Specific industries or markets may have their own set of Artifacts
- Their shapes must not conflict with existing shapes
- They are not part of normal flow, but can be associated with other elements



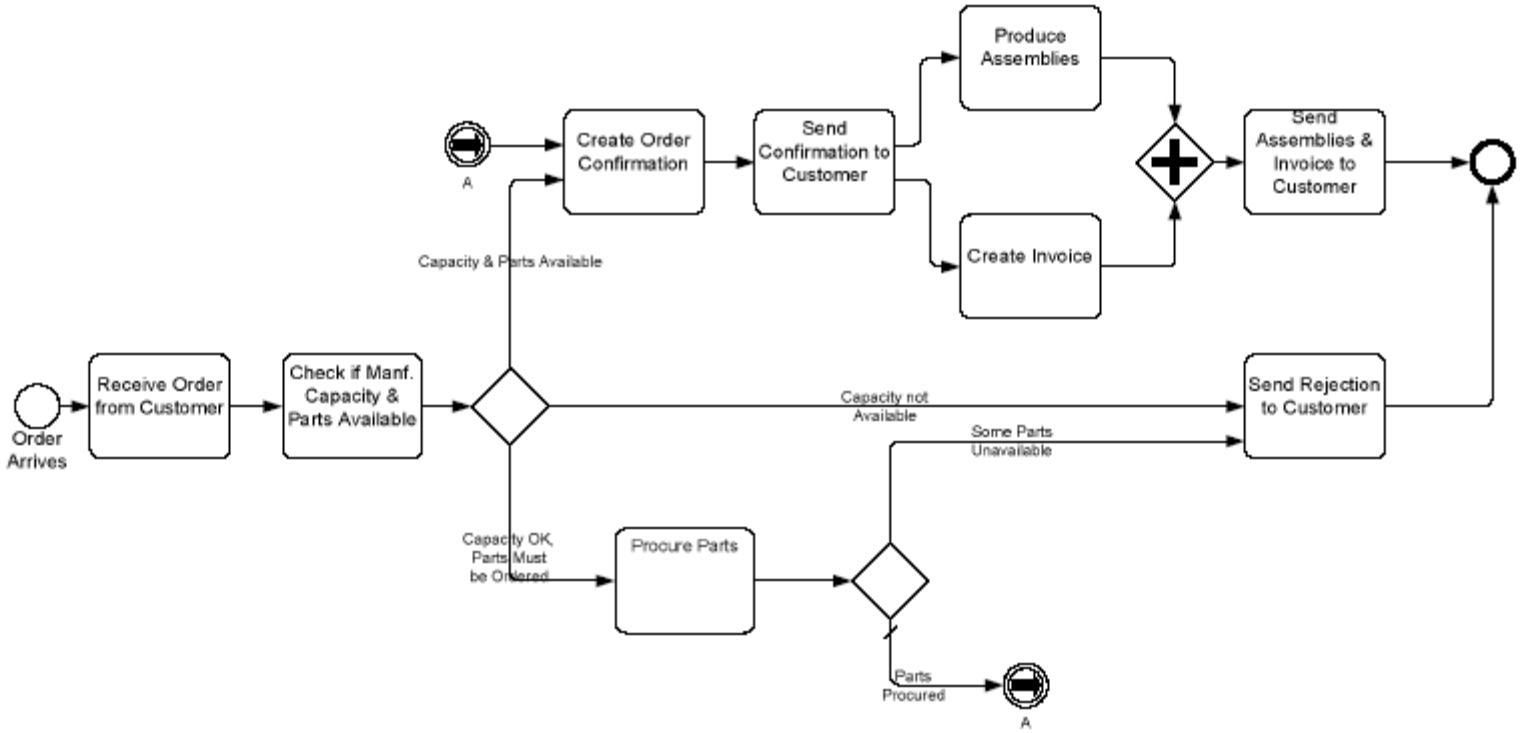
Normal flow

- 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



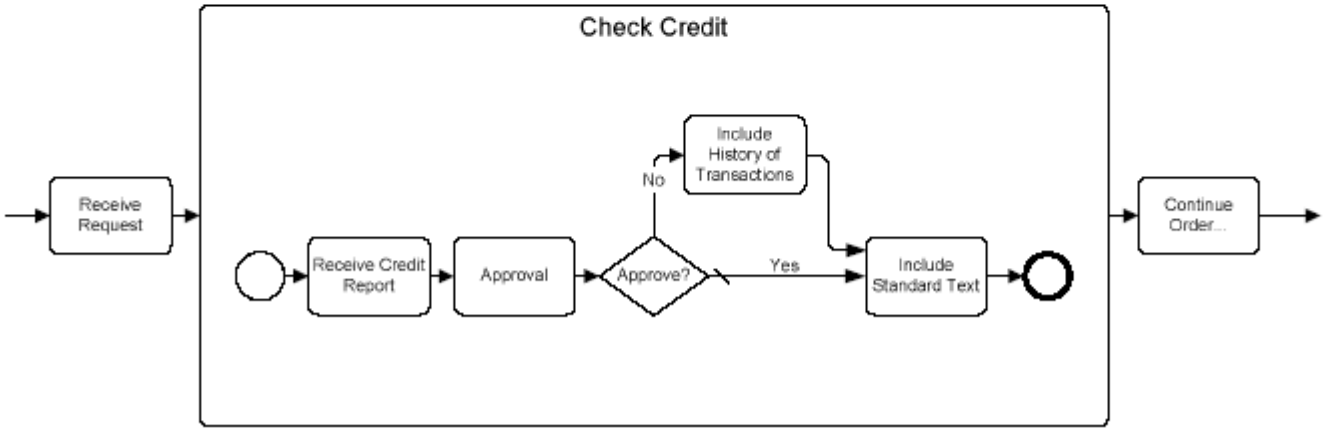
Link events

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

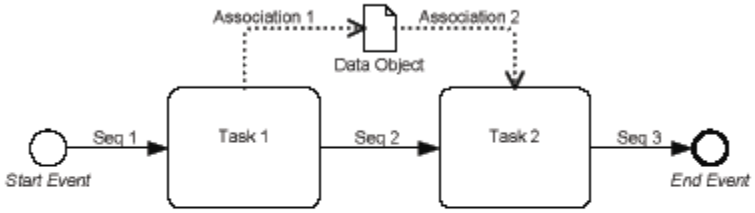


Process leves

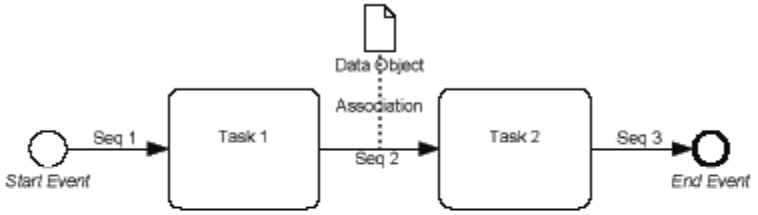
- Processes can be developed hierarchically, with multiple levels through Sub-Processes
- Sequence Flow cannot cross a Sub-Process boundary
  - Message Flow and Associations can cross Sub-Process boundaries



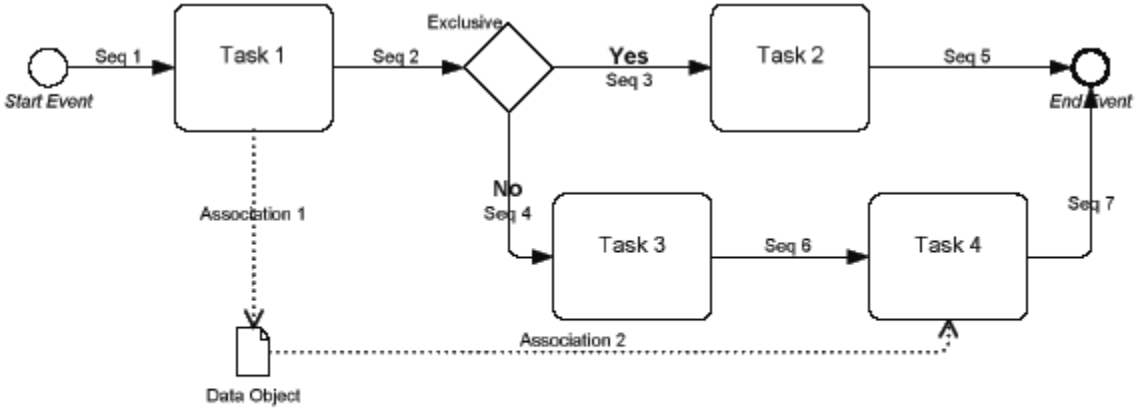
Data flow



Sequence Flow and Data Flow are decoupled

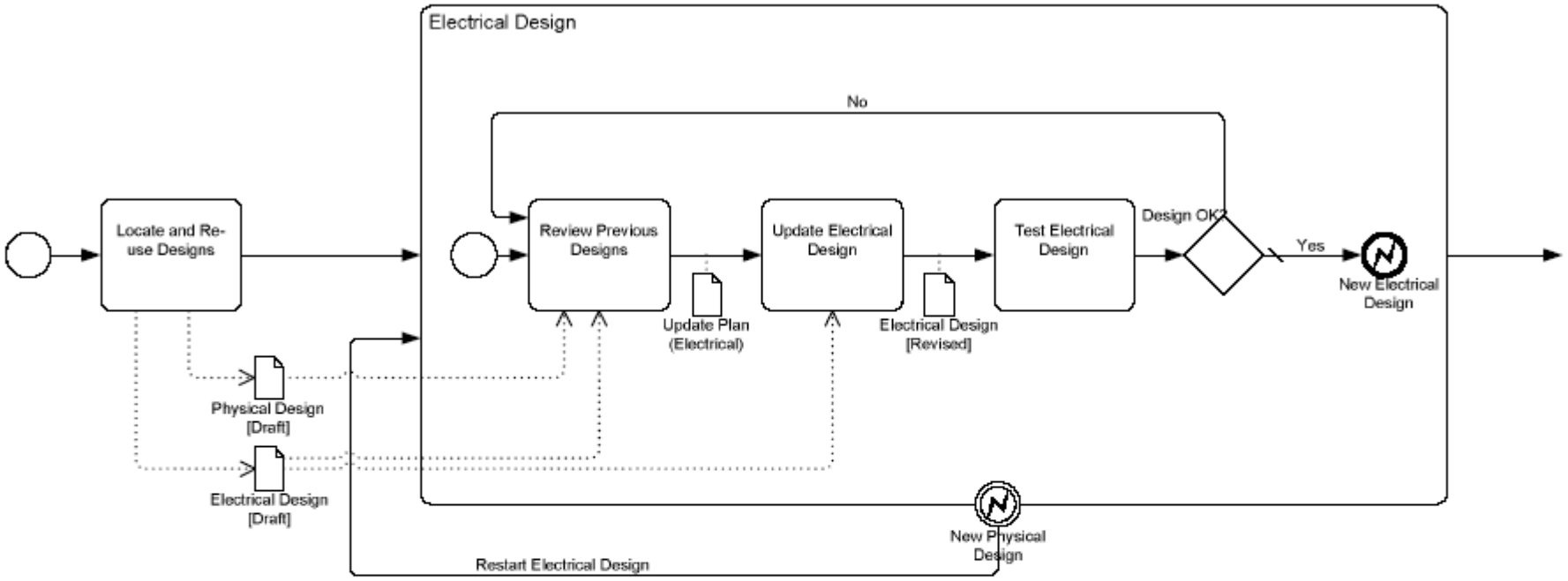


They can be bound together



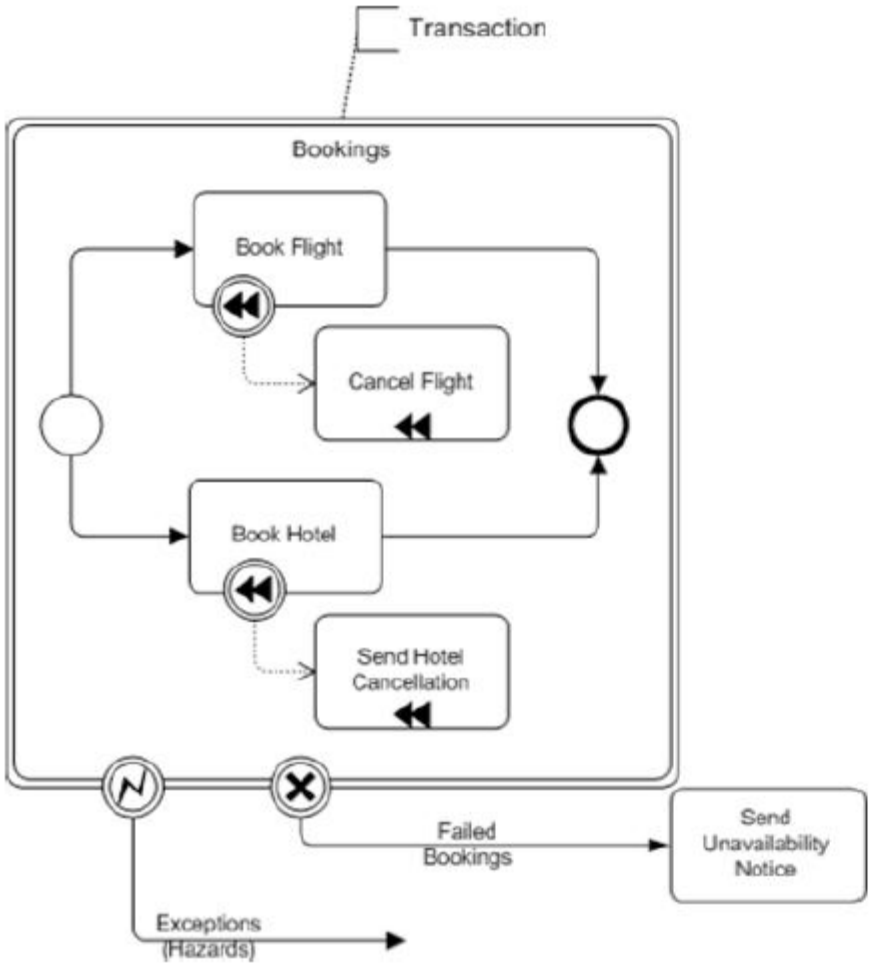
Use case for decoupling

Exceptions



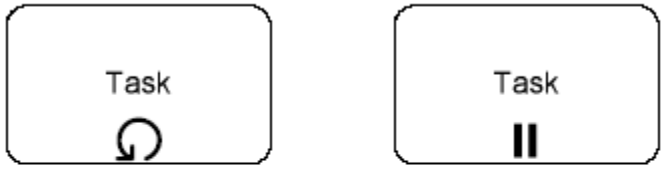
Intermediate Events attached to the boundary of an activity represent triggers that can interrupt the activity. All work within the activity will be stopped and flow will proceed from the Event. Timer, Errors, Messages, etc. can be Triggers.

Compenations and transacations

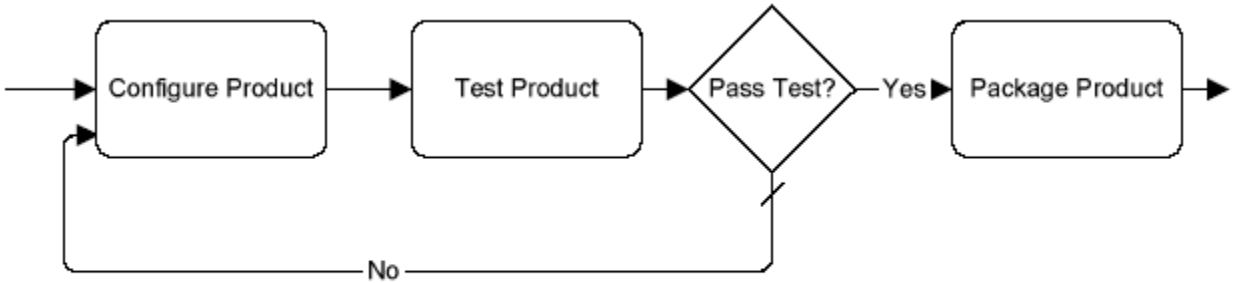


- A Transaction is an activity that has a double border. Transactions are supported by a transaction protocol (e.g., WS-Transaction)
- Normal Outgoing Sequence Flow represents the path to follow a successful completion
- A Cancel Intermediate Event represents the path to follow a cancelled completion
- 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.”

Loops



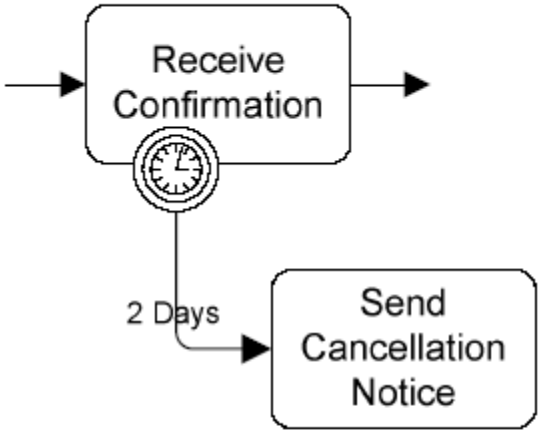
Activity Looping: Do-While;
While-Do; Multiple Instance



Sequence Flow Looping

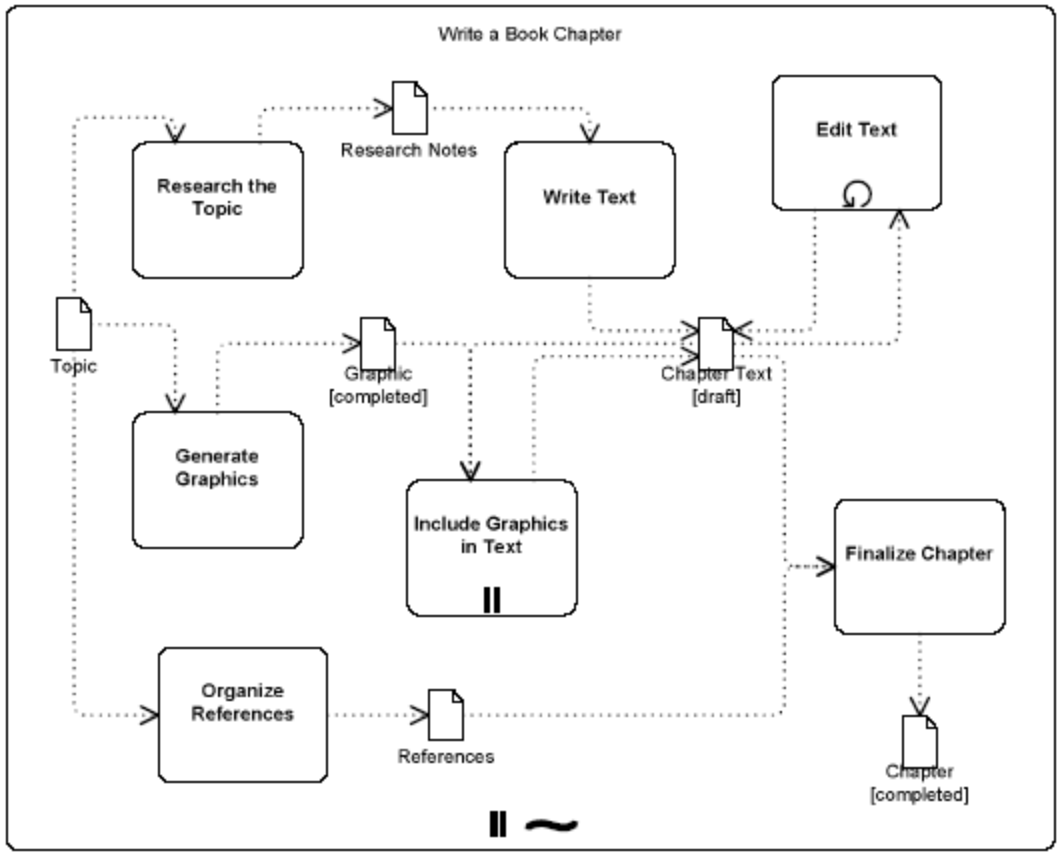
Timers

Timers to add delays in the Process



Timeouts for exception handling

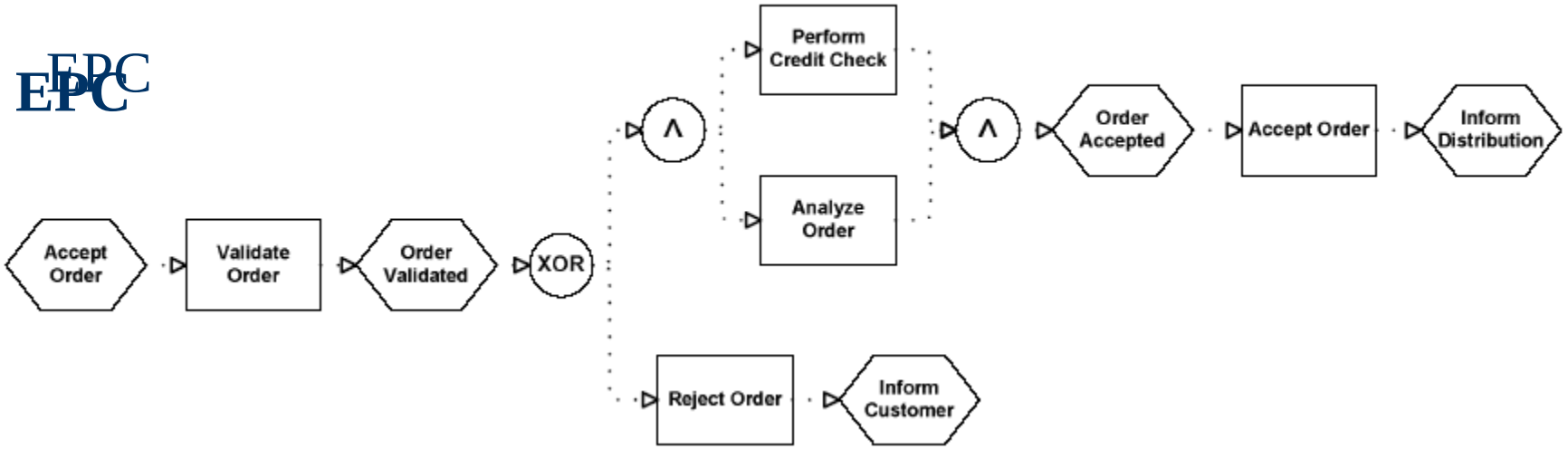
Ad hoc processes



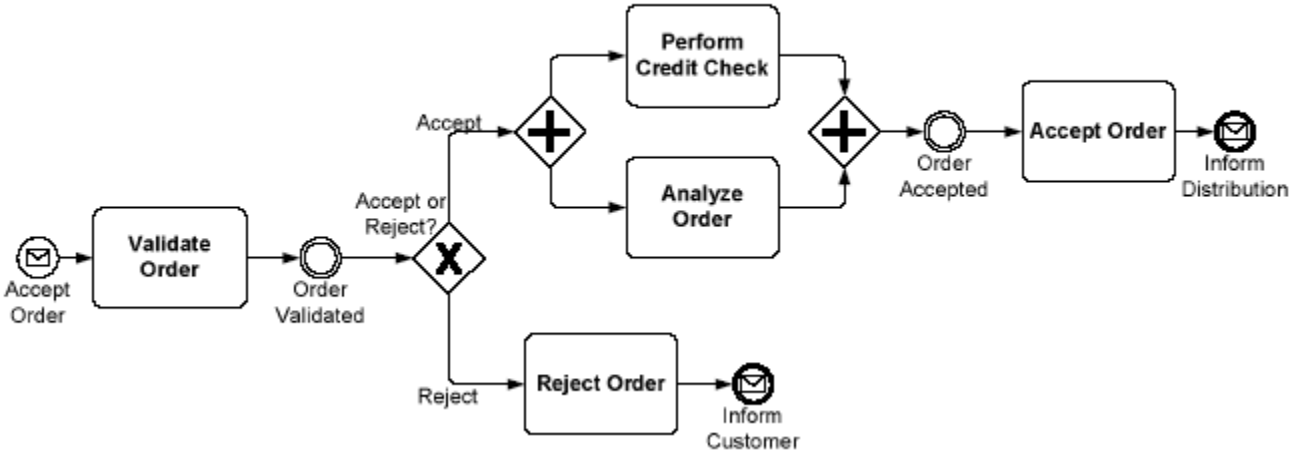
There is no pre-defined Sequence Flow

EPC og BPMN

EPC



BPMN

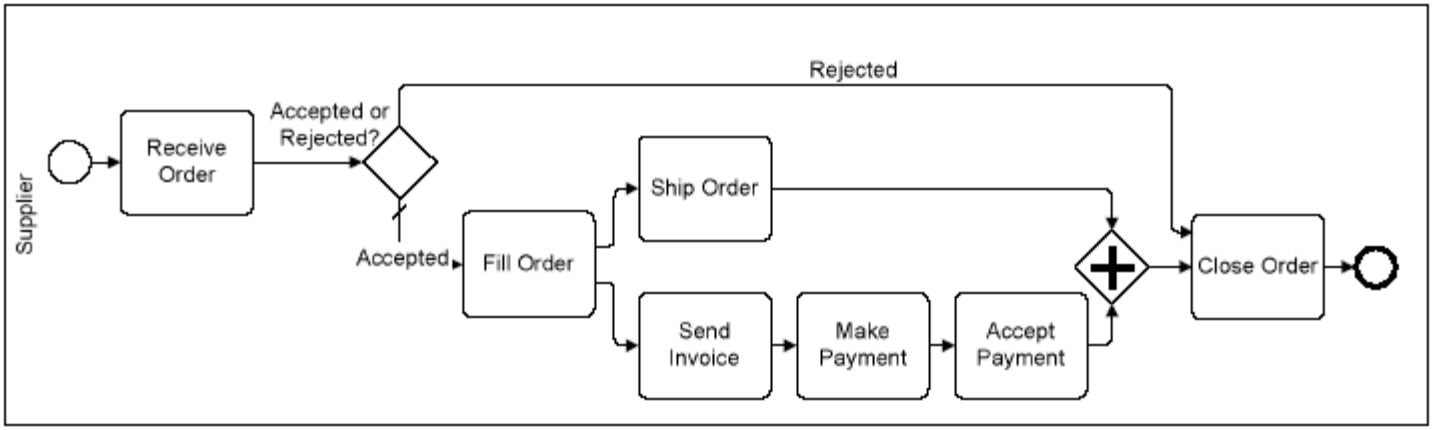


Orchestration versus Choreography

- Orchestration: Workflow, internal processes, private processes
  - Contained within one Pool
- Choreography: Collaboration, global processes, B2B processes
  - Defined by the interaction between Pools

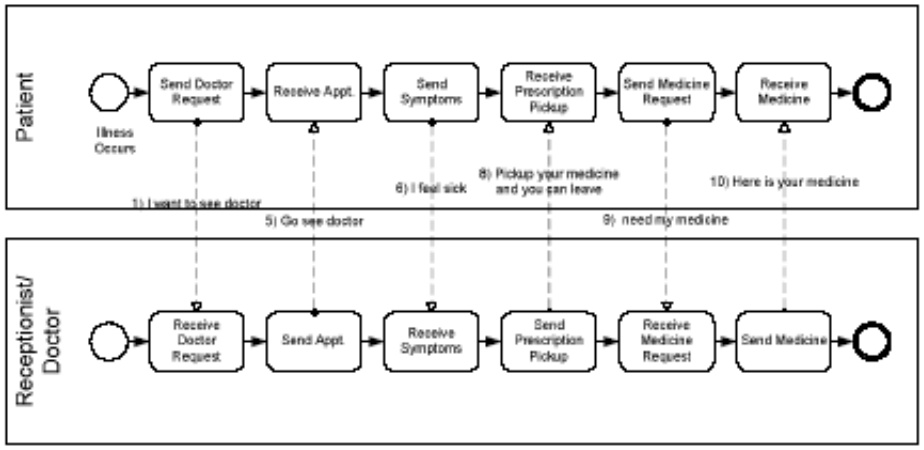
Orchestration

- Orchestration defines processes that are internal to a specific organization
  - Thus, they are contained within a single Pool



Choreography

- A Choreography process depicts the interactions between two or more business entities (as modeled with Pools)
  - Shown by the Message Flow between the Pools



- Or a sequence of interaction (global) types of activities
- BPMN V2.0 will likely update how Choreographies are modeled

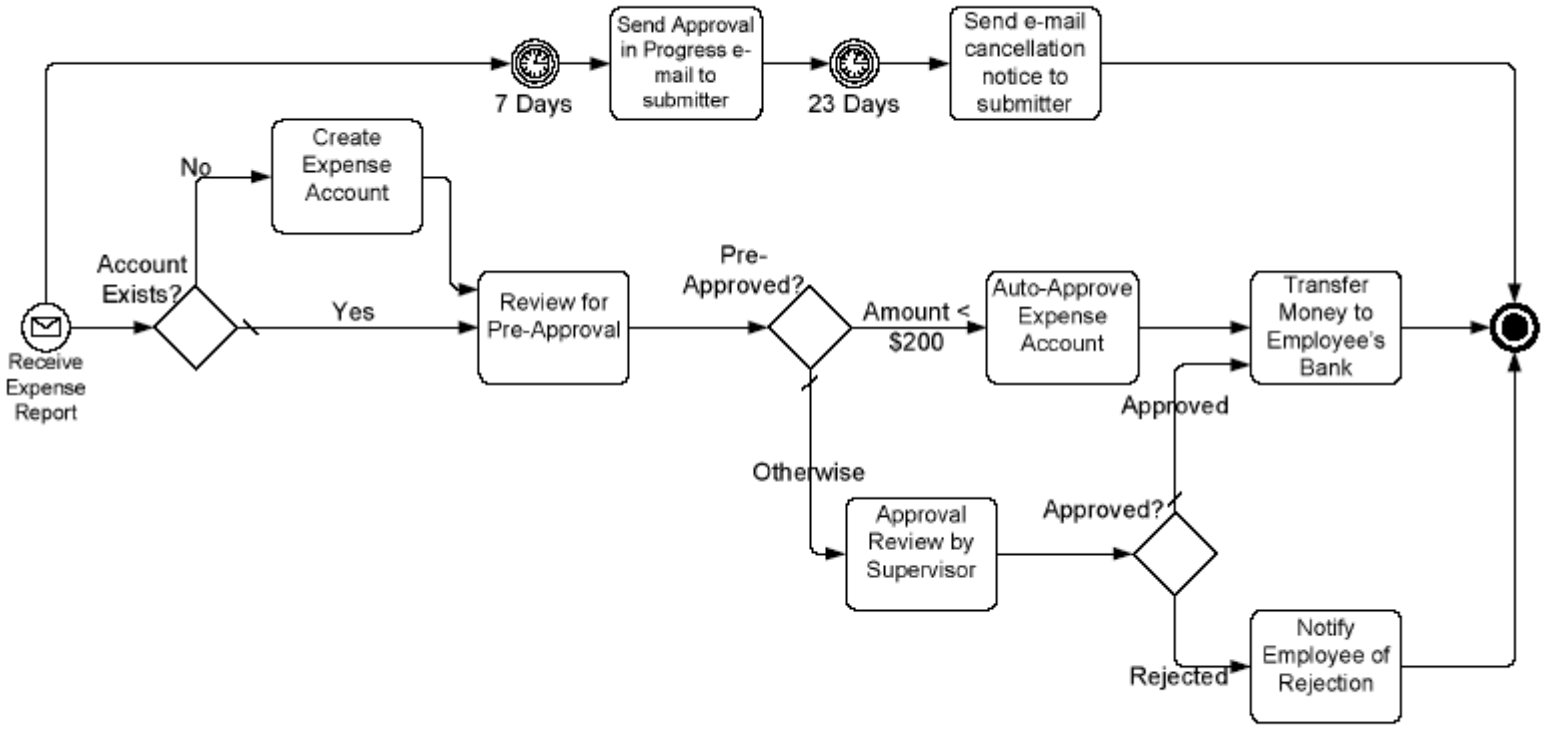
Example

- In this exercise you will read a text descriptive information about a process and will map the process on paper
- The process is a sample expense reimbursement process:
  - This process provides for reimbursement of expenses incurred by employees for the company. For example buying a technical book, office supplies or software. In a normal day there are several hundreds of instances of this process created.
- Concentrate on the basic flow of the Process…

Process information

- After the Expense Report is received, a new account must be created if the employee does not already have one
- The report is then reviewed for automatic approval
  - Amounts under $200 are automatically approved
  - Amounts equal to or over $200 require approval of the supervisor
    - In case of rejection, the employee must receive a rejection notice by email
- The reimbursement goes to the employee's direct deposit bank account
- If no action has happened in 7 days, then the employee must receive an approval in progress email
- If the request is not finished within 30 days, then the process is stopped and the employee receives an email cancellation notice and must re-submit the expense report

Proposal


Send Approval in Progress e-mail to submitter
Send e-mail cancellation notice to submitter
7 Days
23 Days
No
Create Expense Account
Account Exists?
Yes
Review for Pre-Approval
Pre-Approved?
Amount < $200
Auto-Approve Expense Account
Transfer Money to Employee's Bank
Receive Expense Report
Approved
Otherwise
Approval Review by Supervisor
Approved?
Notify Employee of Rejection
Rejected

BPMN Examples …

The overall Business Process represents the global "Bird's Eye" view of the expected behavior that is agreed upon by the Participants.

The "floating" objects are part of the "main" or "background" Pool, which does not have a boundary. The other Pools can be positioned anywhere within the diagram.

The main Pool MUST contain a Process of type "Global."

Assignment Expressions cannot be calculated within a Ca or a "Global" Process.

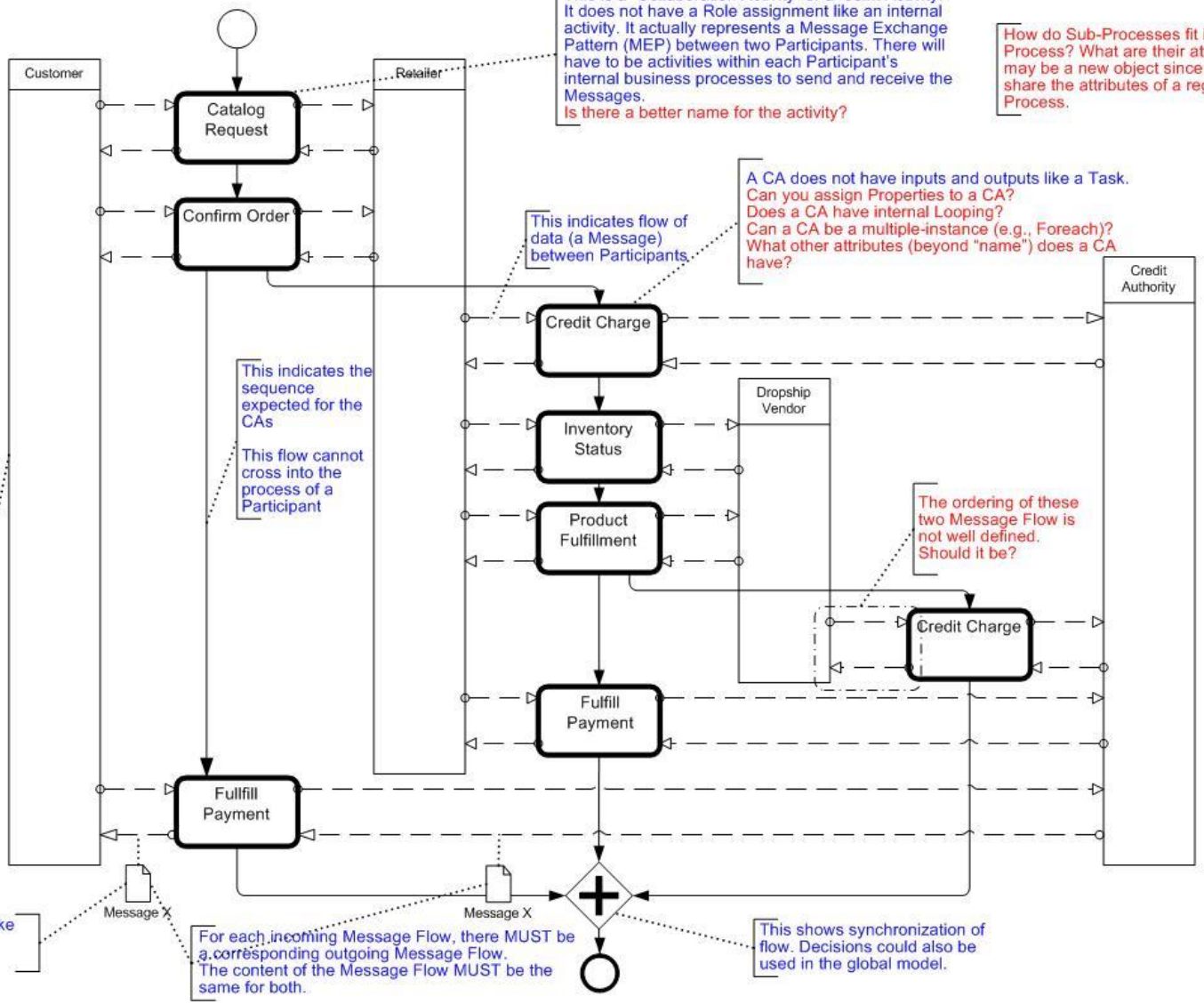
A Pool represents a Participant in the global process: The Participant could be a Business Role or a specific Business Entity. This type of Role is different than the types that are internal to a Participant.

The contents of the box (a business process) could be displayed. The Process can be of any type: "Abstract;" "Private" (executable); "Ad Hoc;" or "Global" (? This would mean Pools within Pools).

If a Process is within the Pool, then the Message Flow MUST connect to an internal Activity and not the boundary of the Pool (unless the Process is a "Global" Process)

Multiple Pools can reference the same Process (if the Participant fulfills multiple Roles in the Global Process)

The Message does not have to be displayed like this. But this does give a place to define the structure of the Message



This is a "Collaboration Activity" or a "Joint Activity." It does not have a Role assignment like an internal activity. It actually represents a Message Exchange Pattern (MEP) between two Participants. There will have to be activities within each Participant's internal business processes to send and receive the Messages.
Is there a better name for the activity?

How do Sub-Processes fit in a "Global" Process? What are their attributes? This may be a new object since it probably not share the attributes of a regular Sub-Process.

A CA does not have inputs and outputs like a Task.
Can you assign Properties to a CA?
Does a CA have internal Looping?
Can a CA be a multiple-instance (e.g., Foreach)?
What other attributes (beyond "name") does a CA have?

This indicates flow of data (a Message) between Participants

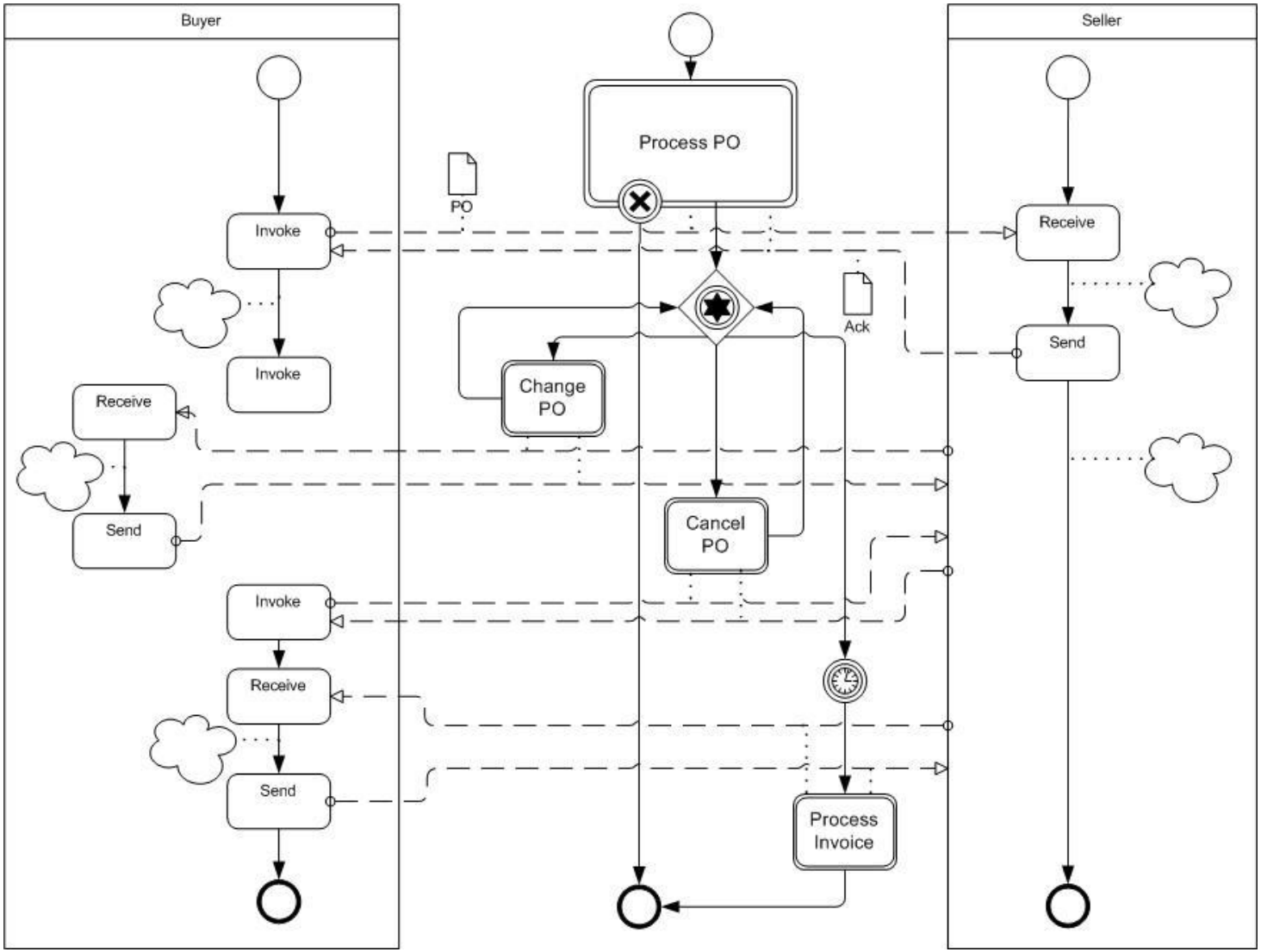
This indicates the sequence expected for the CAs

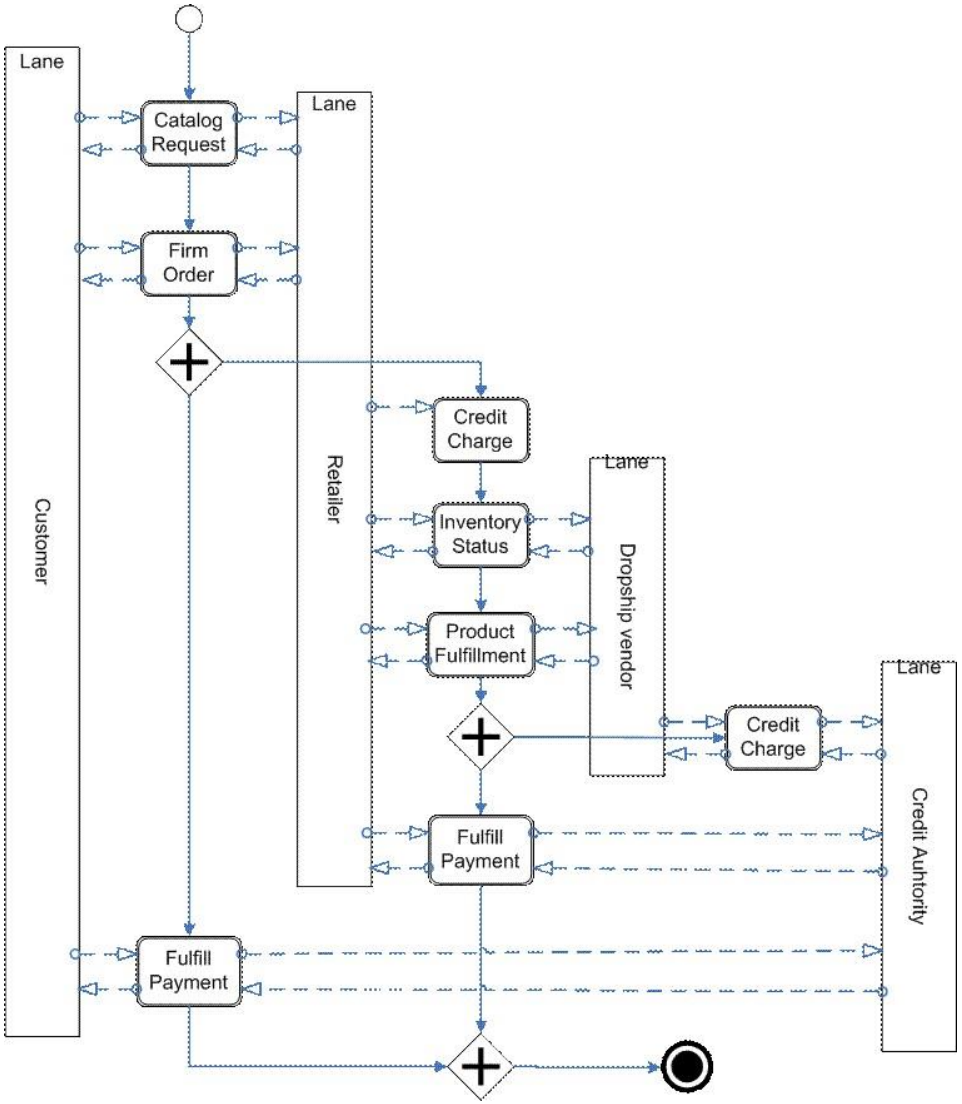
This flow cannot cross into the process of a Participant

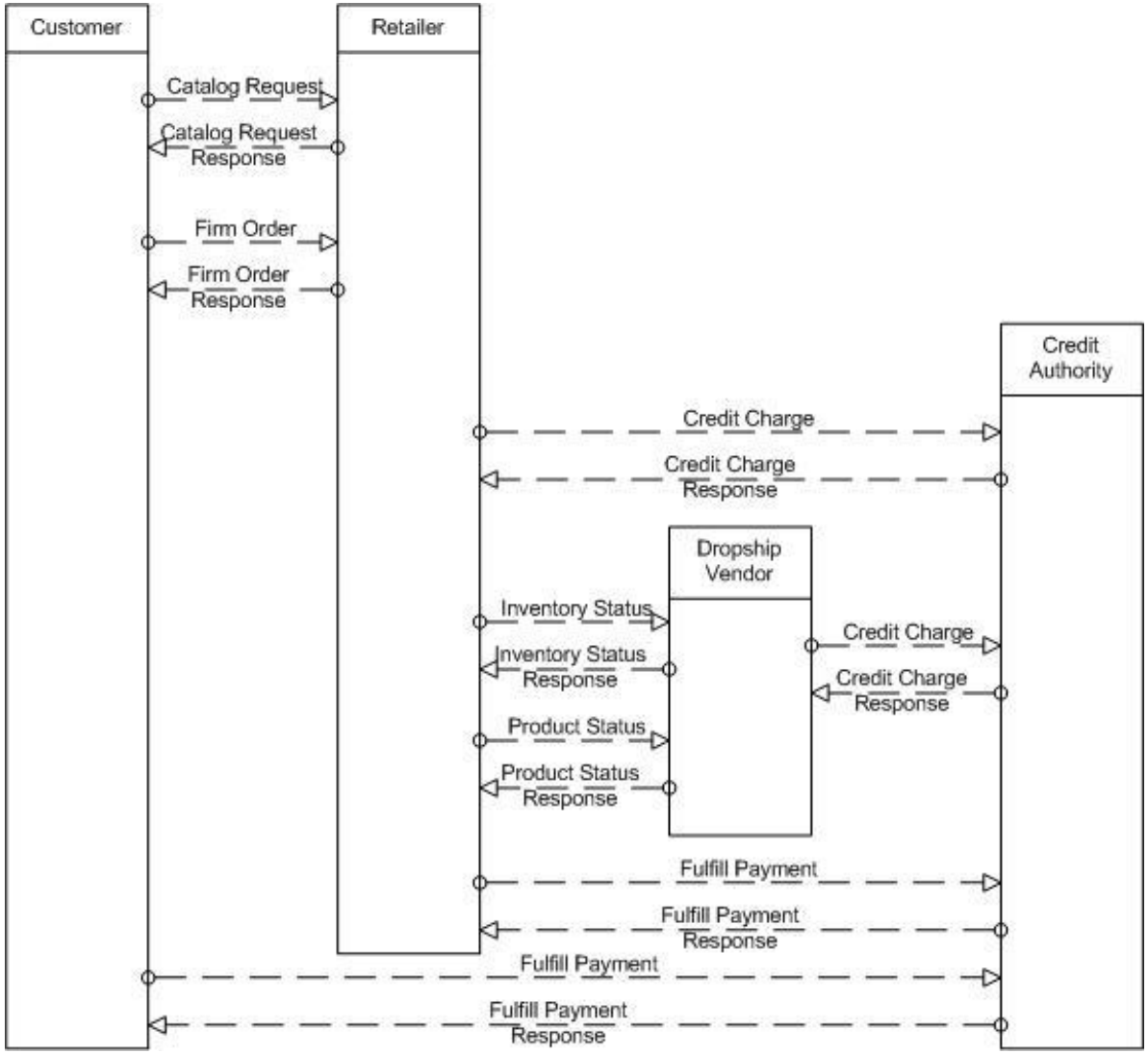
The ordering of these two Message Flow is not well defined. Should it be?

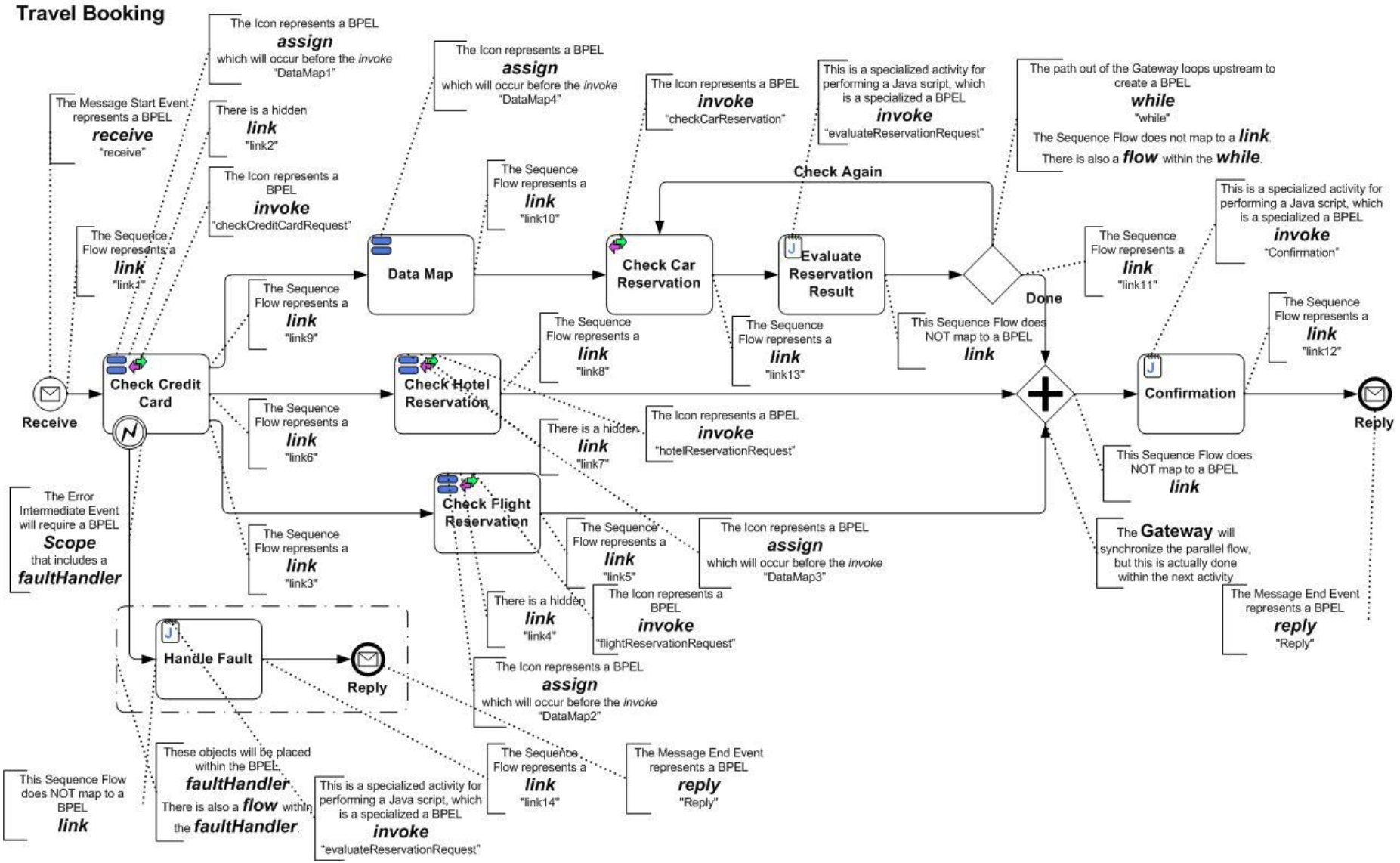
For each incoming Message Flow, there MUST be a corresponding outgoing Message Flow. The content of the Message Flow MUST be the same for both.

This shows synchronization of flow. Decisions could also be used in the global model.


Buyer
Seller
Process PO
PO
Invoke
Invoke
Receive
Send
Change PO
Ack
Receive
Send
Cancel PO
Invoke
Receive
Send
Process Invoice

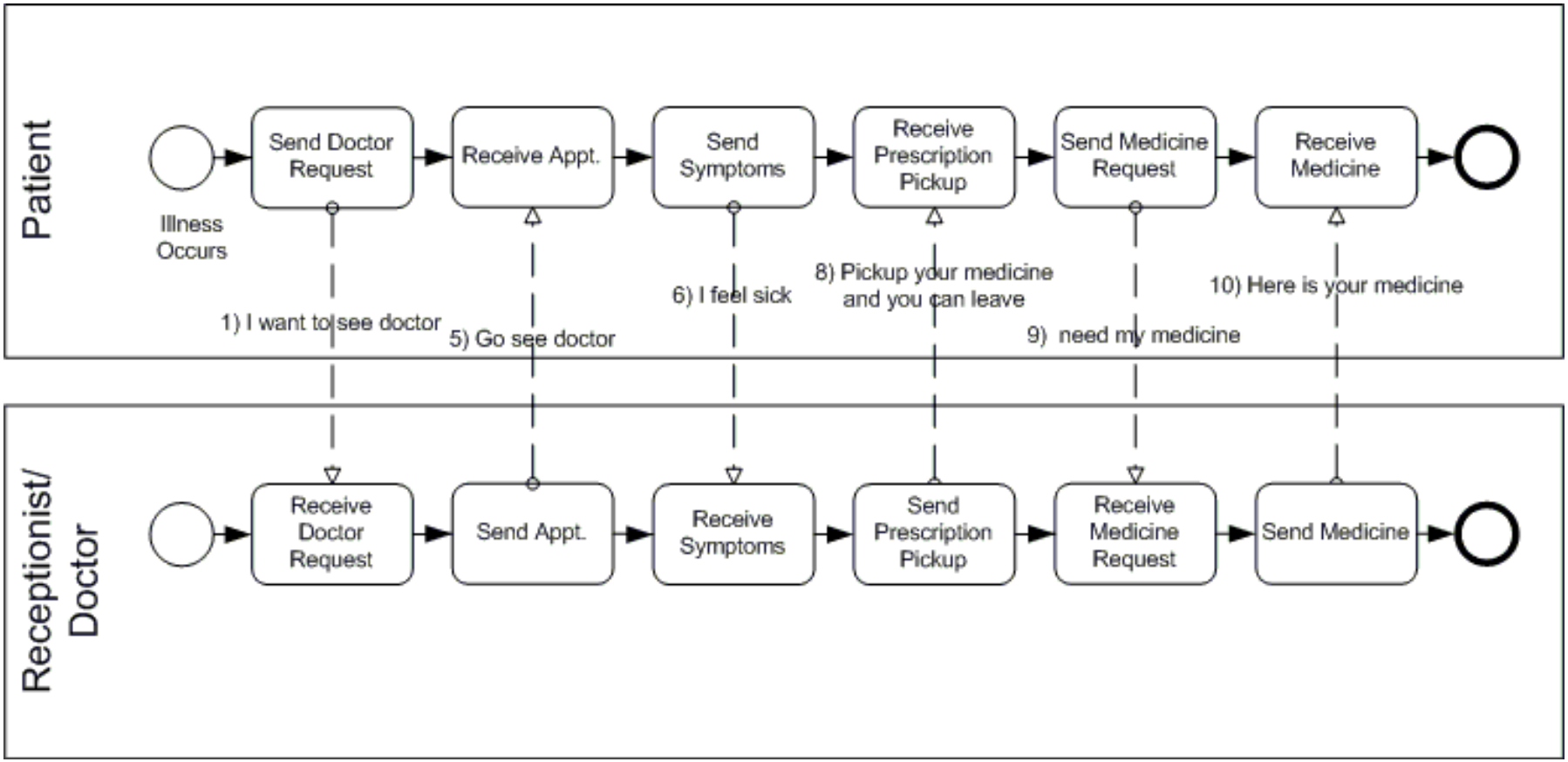

Lane
Customer
Lane
Retailer
Lane
Dropship vendor
Lane
Credit Auhtority
Catalog Request
Firm Order
Credit Charge
Inventory Status
Product Fulfillment
Credit Charge
Fulfill Payment
Fulfill Payment

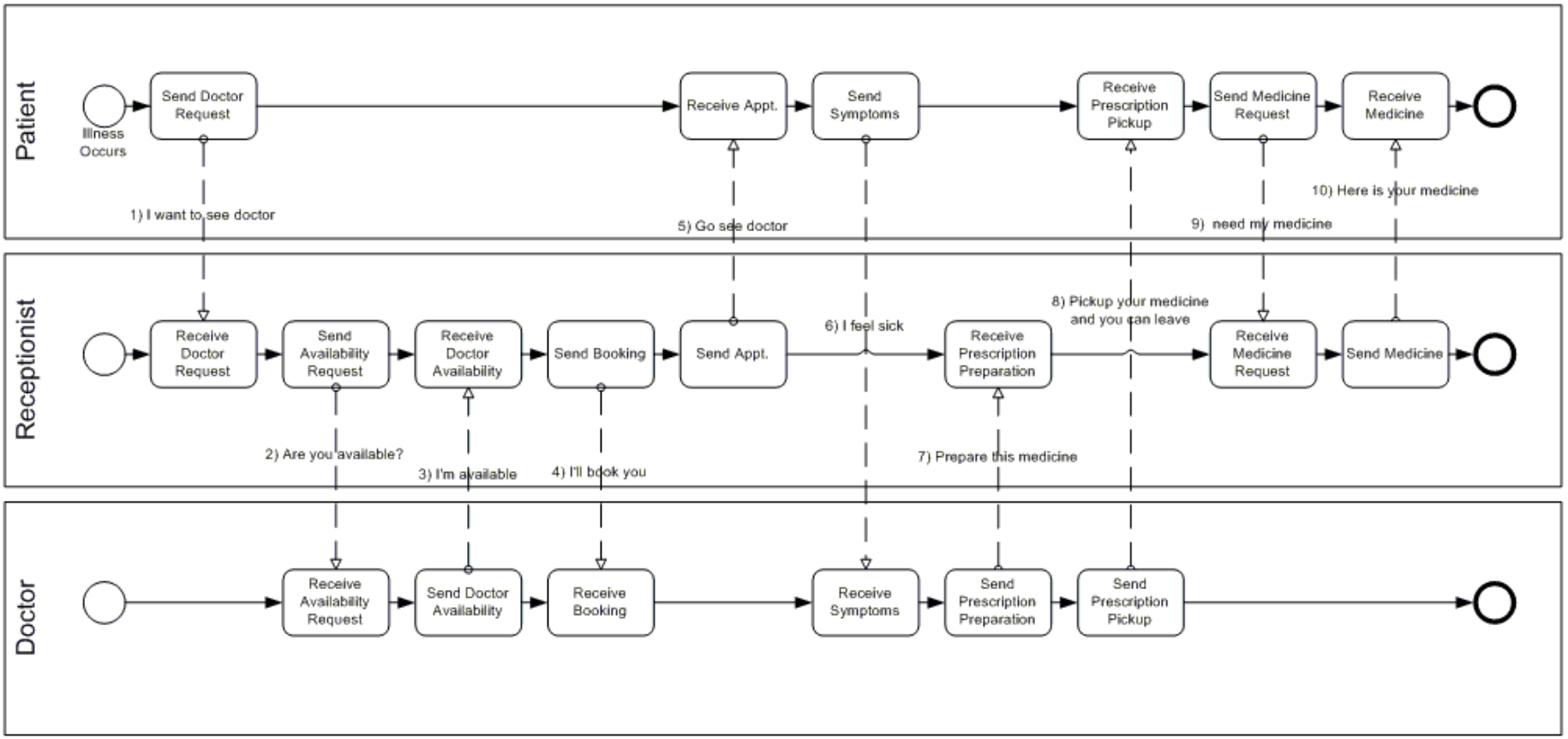

Customer
Retailer
Credit Authority
Dropship Vendor
Catalog Request
Catalog Request Response
Firm Order
Firm Order Response
Credit Charge
Credit Charge Response
Inventory Status
Inventory Status Response
Credit Charge
Credit Charge Response
Product Status
Product Status Response
Fulfill Payment
Fulfill Payment Response
Fulfill Payment
Fulfill Payment Response

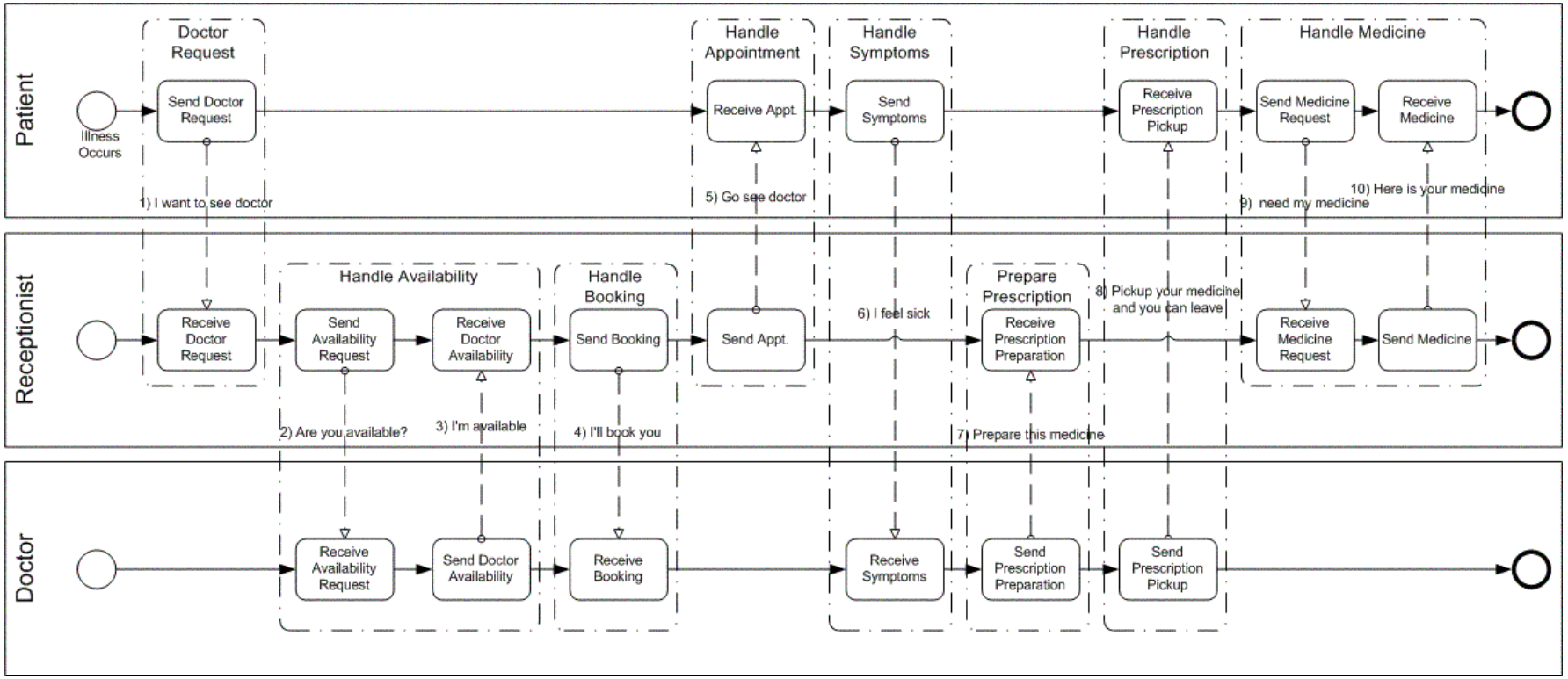

Travel Booking
The Icon represents a BPEL assign which will occur before the invoke "DataMap1"
The Message Start Event represents a BPEL receive "receive"
There is a hidden link "link2"
The Icon represents a BPEL invoke "checkCreditCardRequest"
The Sequence Flow represents a link "link1"
The Icon represents a BPEL assign which will occur before the invoke "DataMap4"
The Sequence Flow represents a link "link10"
Data Map
The Icon represents a BPEL invoke "checkCarReservation"
This is a specialized activity for performing a Java script, which is a specialized a BPEL invoke "evaluateReservationRequest"
Check Car Reservation
Evaluate Reservation Result
Check Again
The path out of the Gateway loops upstream to create a BPEL while "while"
The Sequence Flow does not map to a link.
There is also a flow within the while.
Done
The Sequence Flow represents a link "link11"
This is a specialized activity for performing a Java script, which is a specialized a BPEL invoke "Confirmation"
The Sequence Flow represents a link "link9"
The Sequence Flow represents a link "link8"
The Sequence Flow represents a link "link13"
This Sequence Flow does NOT map to a BPEL link
The Sequence Flow represents a link "link12"
Receive
Check Credit Card
Check Hotel Reservation
Confirmation
Reply
The Sequence Flow represents a link "link6"
There is a hidden link "link7"
The Icon represents a BPEL invoke "hotelReservationRequest"
This Sequence Flow does NOT map to a BPEL link
The Error Intermediate Event will require a BPEL Scope that includes a faultHandler
Check Flight Reservation
The Sequence Flow represents a link "link3"
The Sequence Flow represents a link "link5"
The Icon represents a BPEL assign which will occur before the invoke "DataMap3"
The Gateway will synchronize the parallel flow, but this is actually done within the next activity
There is a hidden link "link4"
The Icon represents a BPEL invoke "flightReservationRequest"
The Message End Event represents a BPEL reply "Reply"
Handle Fault
Reply
The Icon represents a BPEL assign which will occur before the invoke "DataMap2"
These objects will be placed within the BPEL faultHandler. There is also a flow within the faultHandler.
This Sequence Flow does NOT map to a BPEL link
This is a specialized activity for performing a Java script, which is a specialized a BPEL invoke "evaluateReservationRequest"
The Sequence Flow represents a link "link14"
The Message End Event represents a BPEL reply "Reply"

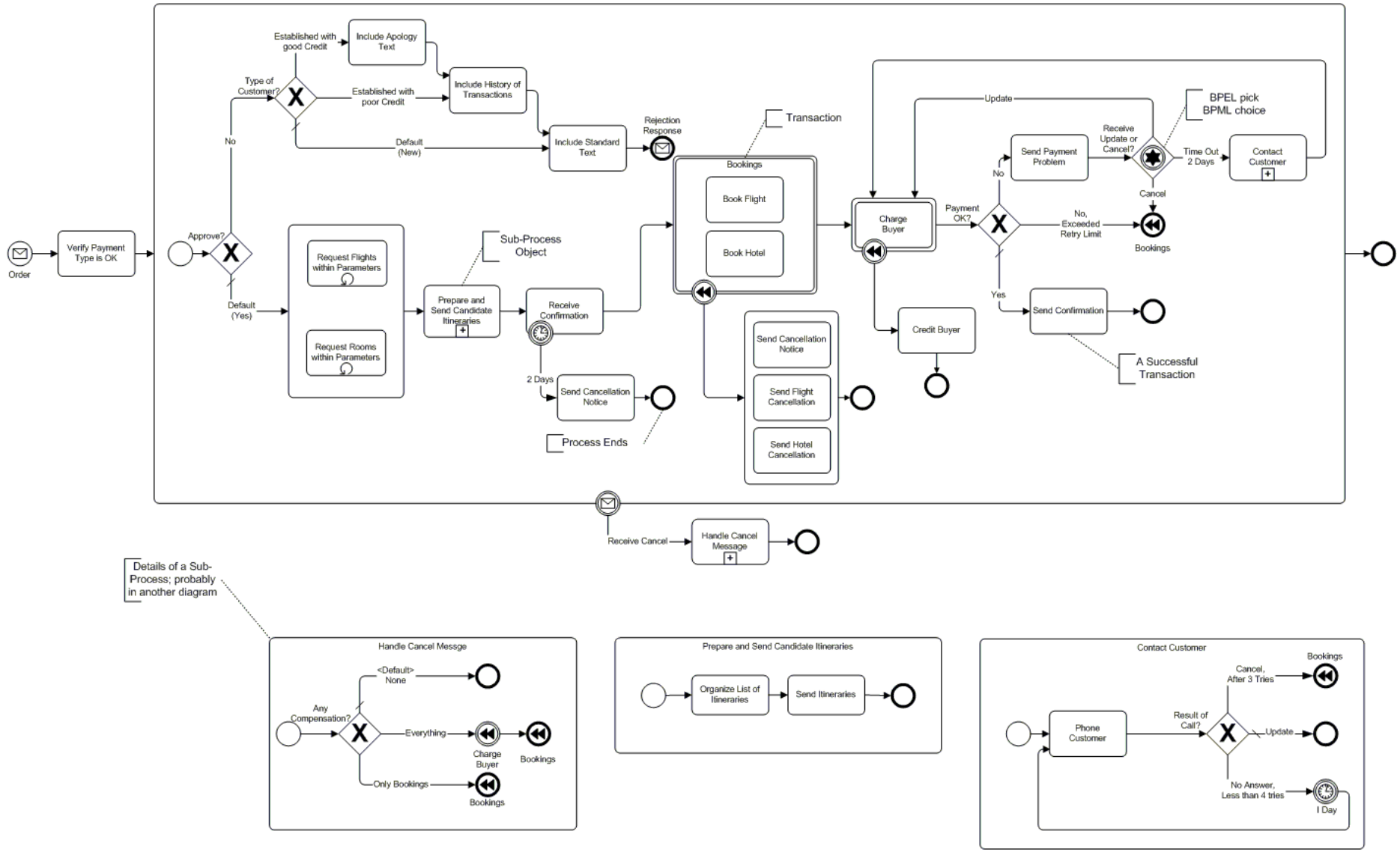
Example – doctor's office

- 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

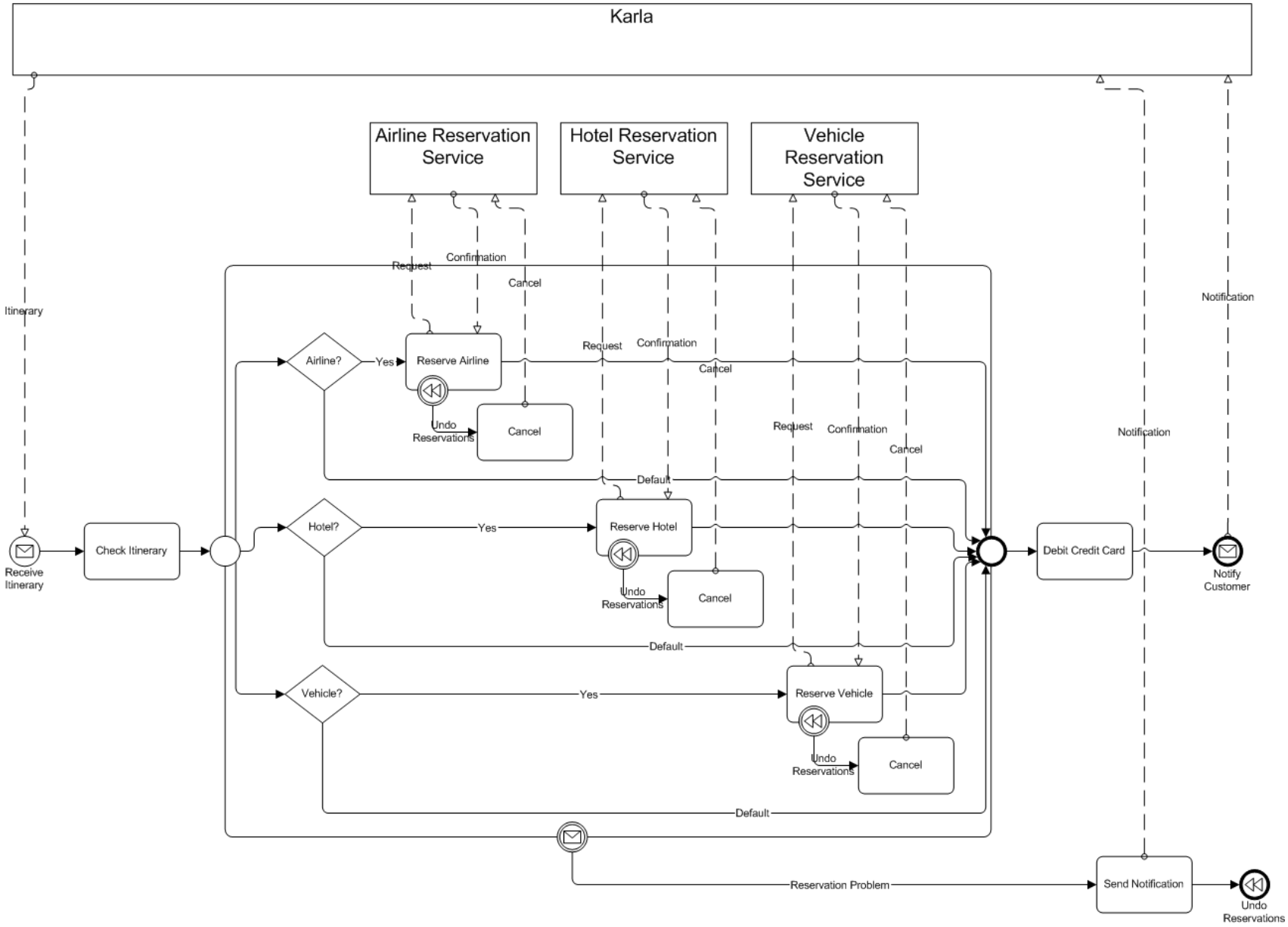

Patient
Send Doctor Request
Receive Appt.
Send Symptoms
Receive Prescription Pickup
Send Medicine Request
Receive Medicine
Illness Occurs
1) I want to see doctor
5) Go see doctor
6) I feel sick
8) Pickup your medicine and you can leave
9) need my medicine
10) Here is your medicine
Receptionist/ Doctor
Receive Doctor Request
Send Appt.
Receive Symptoms
Send Prescription Pickup
Receive Medicine Request
Send Medicine


Patient
Illness Occurs
Send Doctor Request
Receive Appt.
Send Symptoms
Receive Prescription Pickup
Send Medicine Request
Receive Medicine
1) I want to see doctor
5) Go see doctor
9) need my medicine
10) Here is your medicine
Receptionist
Receive Doctor Request
Send Availability Request
Receive Doctor Availability
Send Booking
Send Appt.
6) I feel sick
Receive Prescription Preparation
8) Pickup your medicine and you can leave
Receive Medicine Request
Send Medicine
2) Are you available?
3) I'm available
4) I'll book you
7) Prepare this medicine
Doctor
Receive Availability Request
Send Doctor Availability
Receive Booking
Receive Symptoms
Send Prescription Preparation
Send Prescription Pickup


Patient
Receptionist
Doctor
Doctor Request
Handle Appointment
Handle Symptoms
Handle Prescription
Handle Medicine
Handle Availability
Handle Booking
Prepare Prescription
Illness Occurs
Send Doctor Request
Receive Appt.
Send Symptoms
Receive Prescription Pickup
Send Medicine Request
Receive Medicine
1) I want to see doctor
5) Go see doctor
9) need my medicine
10) Here is your medicine
Receive Doctor Request
Send Availability Request
Receive Doctor Availability
Send Booking
Send Appt.
6) I feel sick
Receive Prescription Preparation
8) Pickup your medicine and you can leave
Receive Medicine Request
Send Medicine
2) Are you available?
3) I'm available
4) I'll book you
7) Prepare this medicine
Receive Availability Request
Send Doctor Availability
Receive Booking
Receive Symptoms
Send Prescription Preparation
Send Prescription Pickup



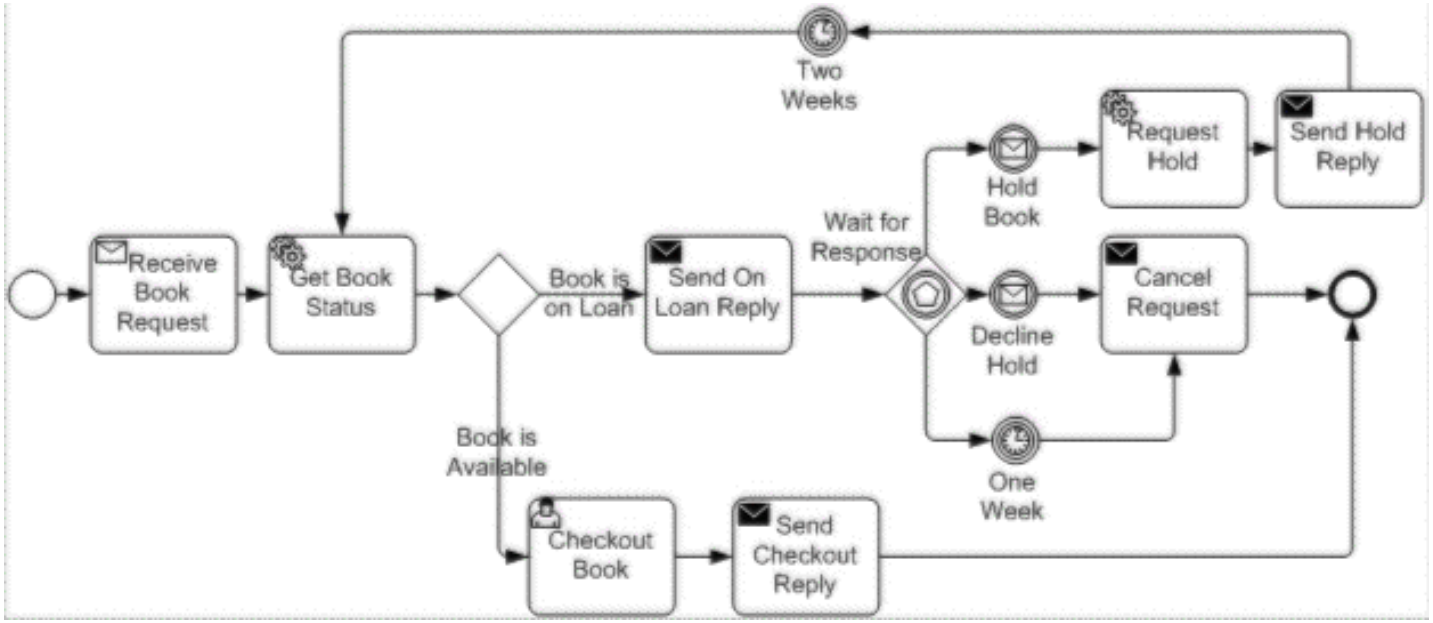
Details of a Sub-Process; probably in another diagram


Karla
Airline Reservation Service
Hotel Reservation Service
Vehicle Reservation Service
Itinerary
Request
Confirmation
Cancel
Notification
Airline?
Yes
Reserve Airline
Undo Reservations
Cancel
Default
Receive Itinerary
Check Itinerary
Hotel?
Reserve Hotel
Debit Credit Card
Notify Customer
Vehicle?
Reserve Vehicle
Reservation Problem
Send Notification
Undo Reservations

BPMN 2.0: Major changes from BPMN1.x

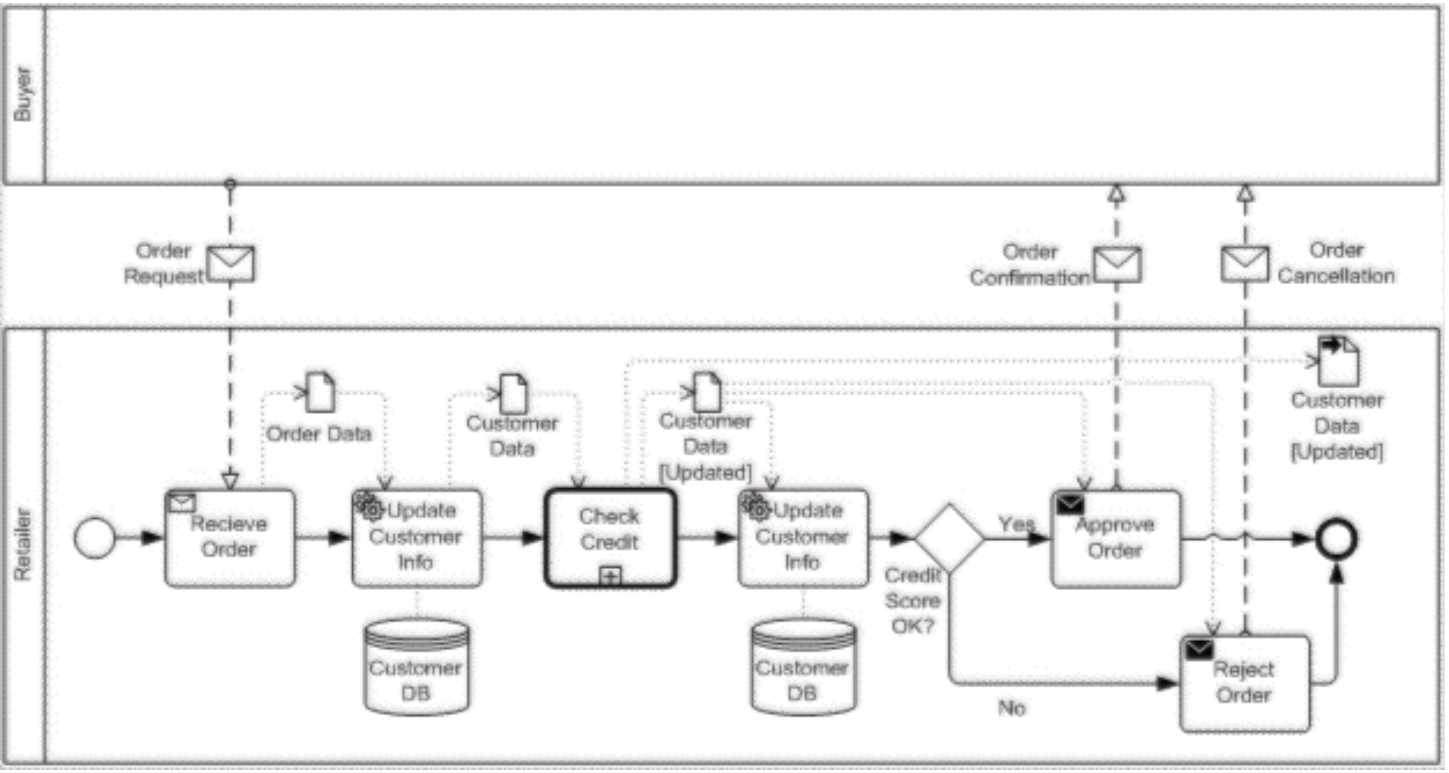
- Notational changes
  - New diagrams for Choreography and Conversation
  - New event-types (escalation, …)
  - Non-interrupting events
  - Event sub-process
  - Call Activity– replaces linked/reusable activity
- Technical changes
  - Formal metamodel – specified in UML
  - Interchange formats for semantic model interchange (XMI, XSD)
  - Interchange formats for diagram interchange (XMI, XSD)
  - XSLT transformations between XMI and XSD formats

Process diagram



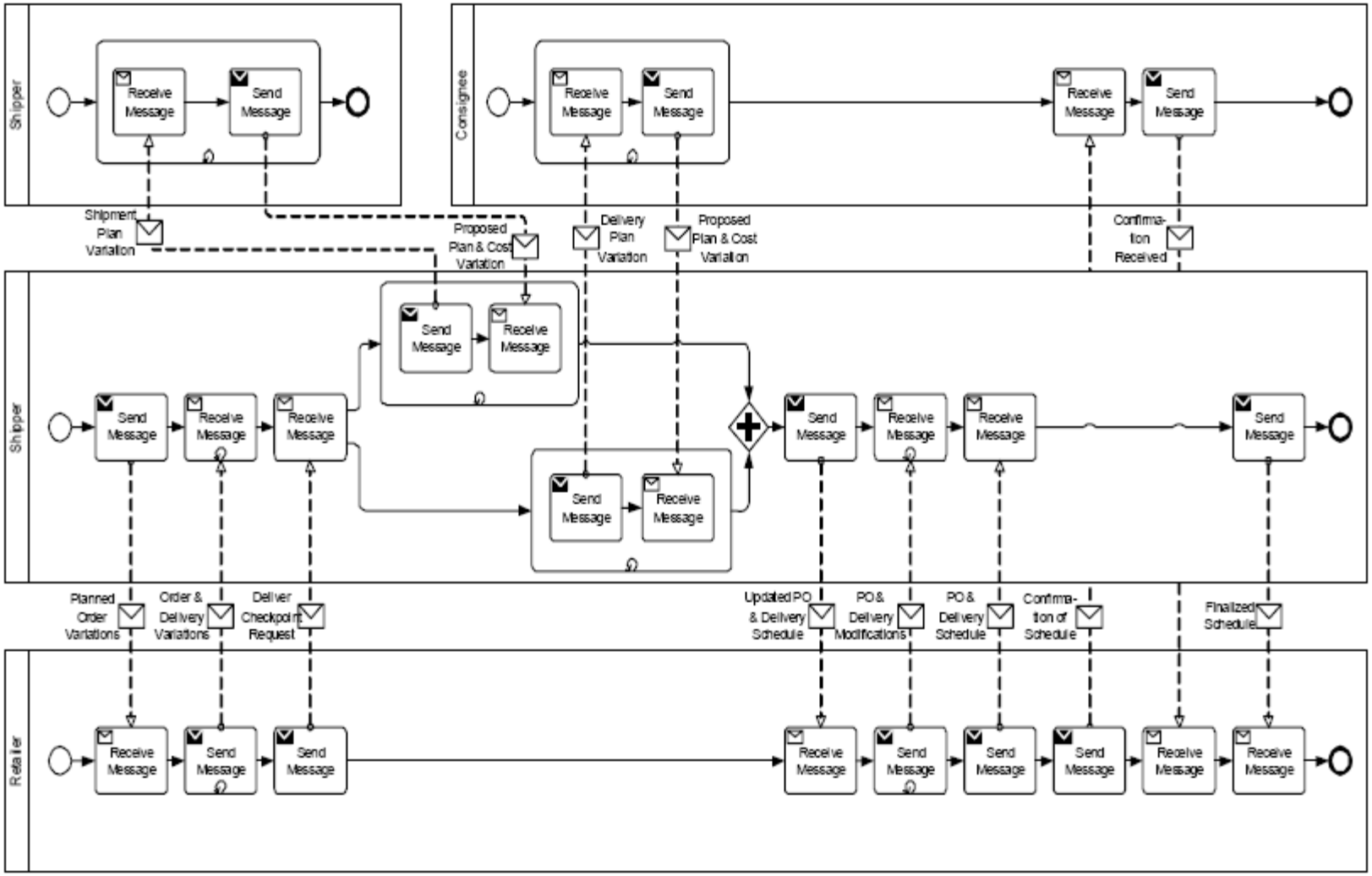
- Flowchart view to sequence activities within an organization
- Support the modeling of simple processes
- Enhanced by BPMN to handle more complex concepts, such as exception handling, transactions, and compensation.

Collaboration diagram

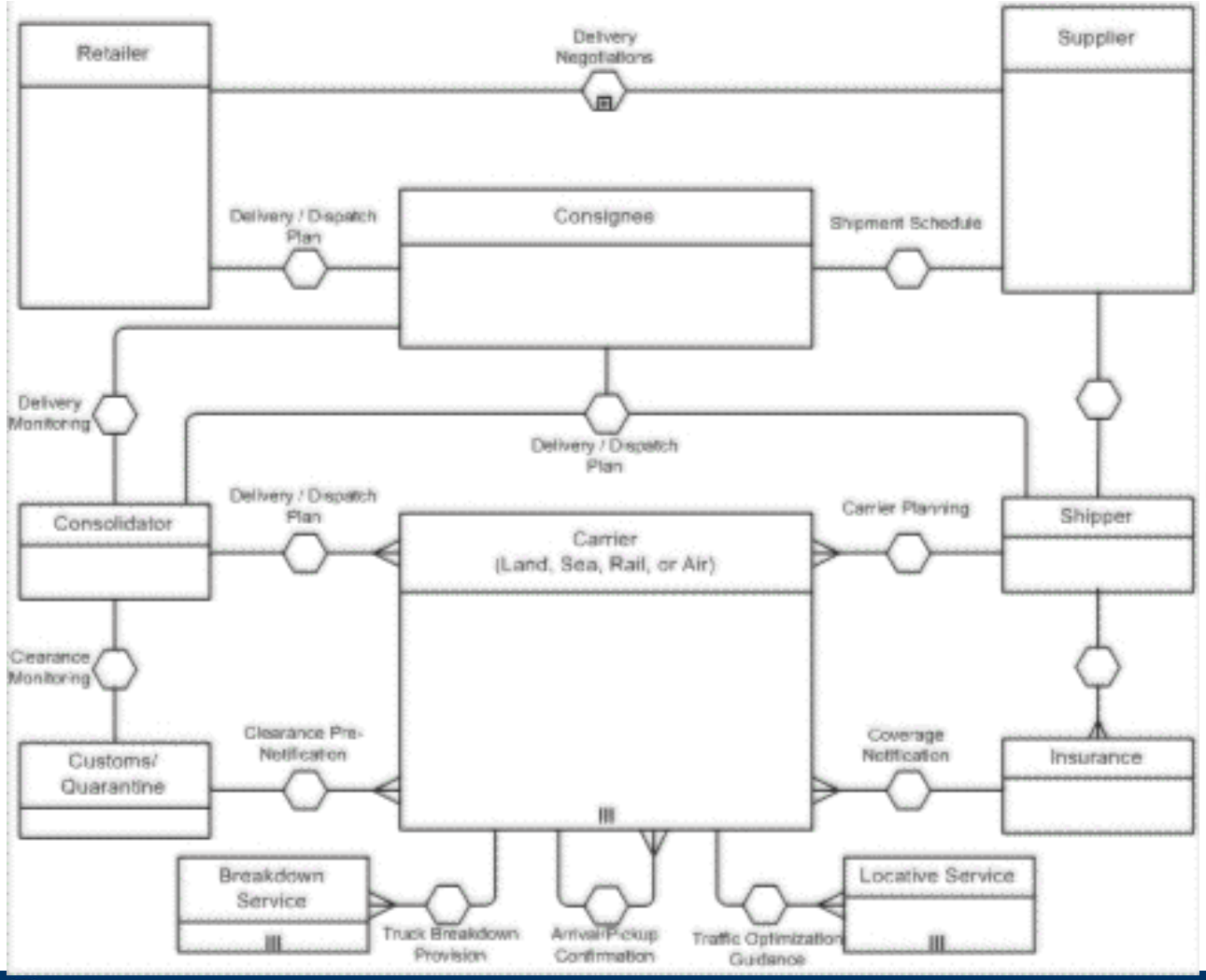


- Provides a view of the interactions (flow of messages) between two or more business partners (Participants).
- Collaborations can be combined with Processes to show how the interactions are related to the internal Process activities.

Collaboration diagram example

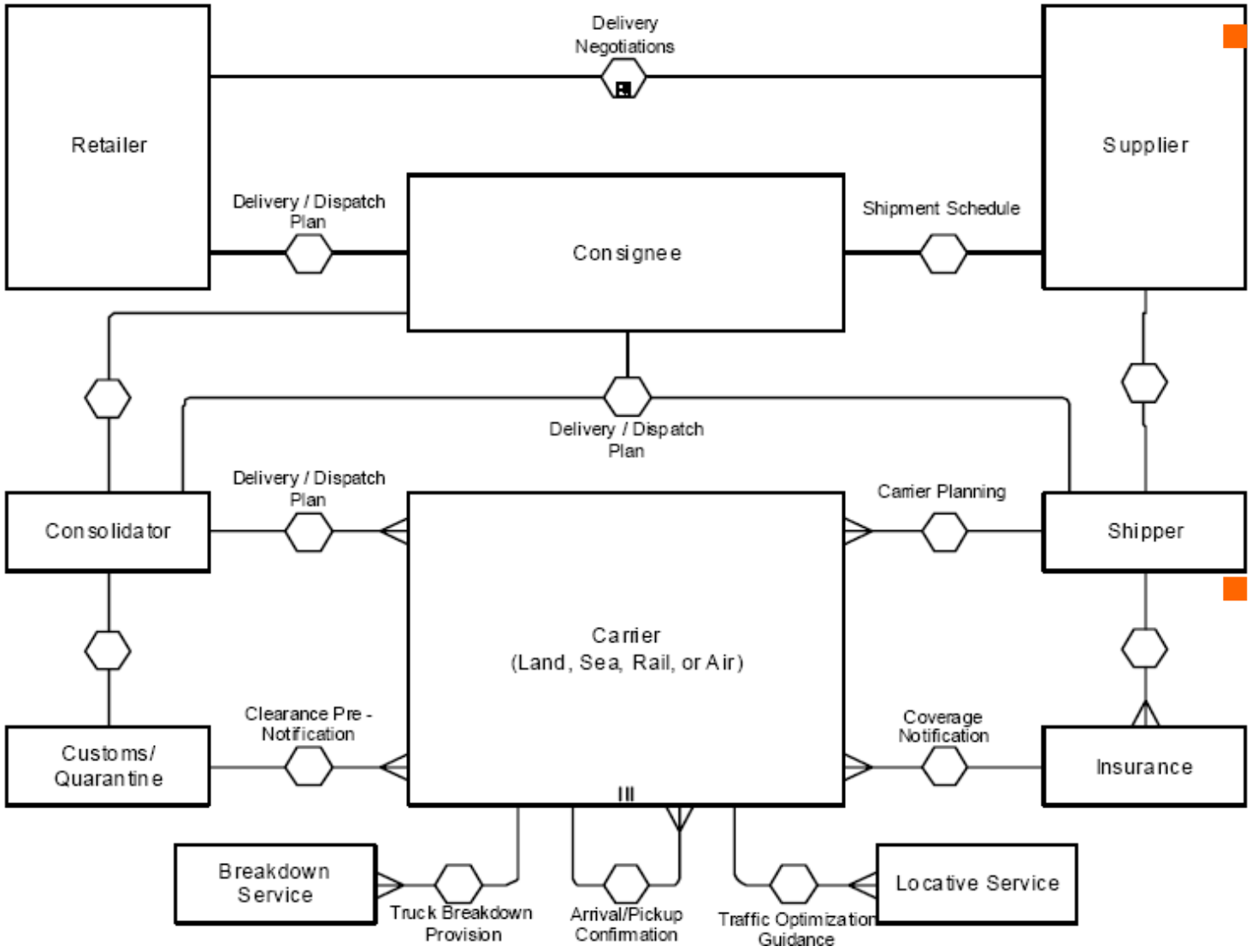


Conversation diagram



- Allows a modeler to group Collaboration interactions between two or more Participants, which together achieve a common goal, e.g. “negotiate delivery”
- The grouping can be based on business keys such as customer id or shipping id.

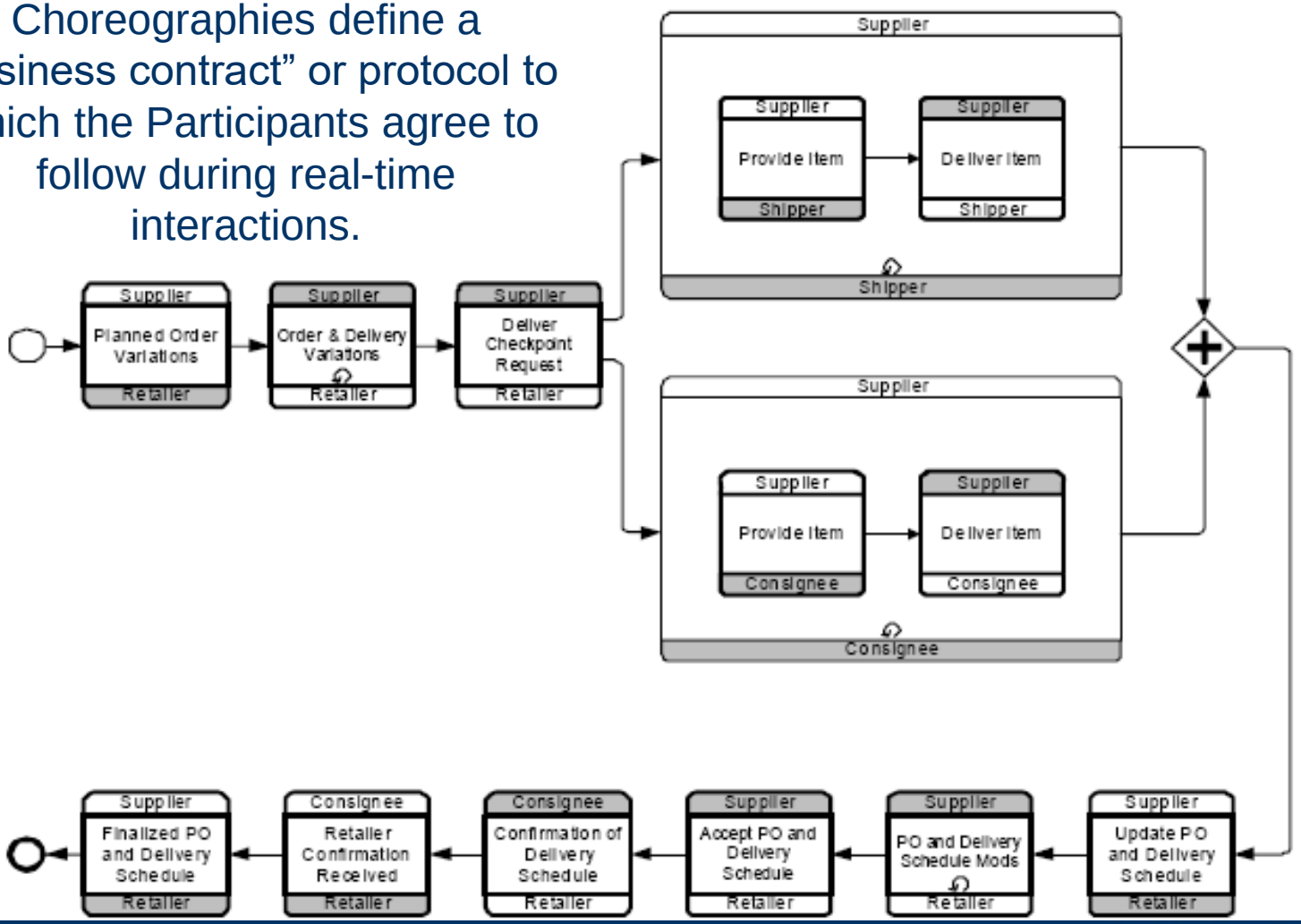
Conversation diagram



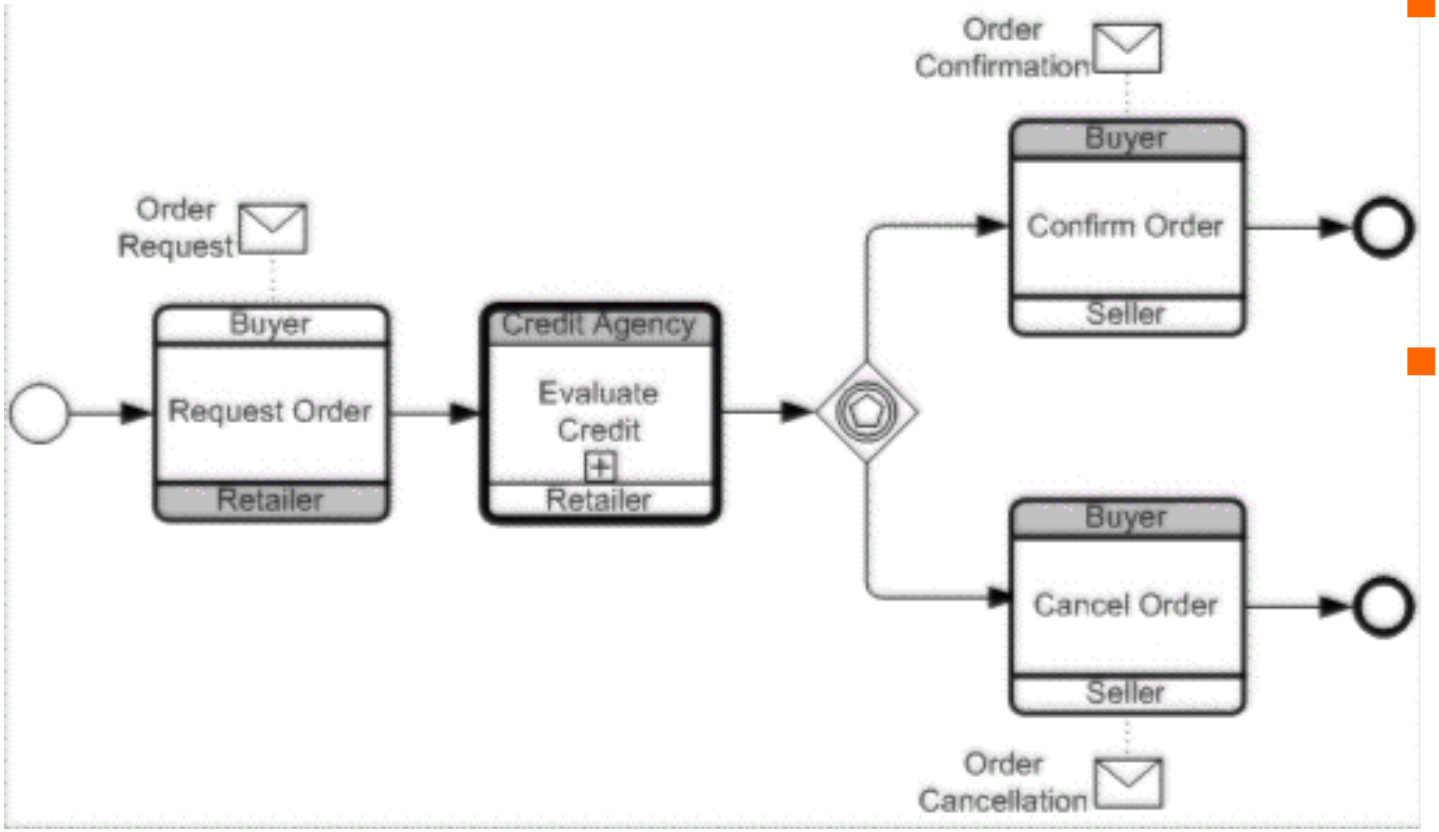
- Allows a modeler to group Collaboration interactions between two or more Participants, which together achieve a common goal, e.g. “negotiate delivery”
- The grouping can be based on business keys such as customer id or shipping id.

Corresponding choreography example

- Provides a flowchart view to sequence interactions between Participants
- Choreographies define a "business contract" or protocol to which the Participants agree to follow during real-time interactions.



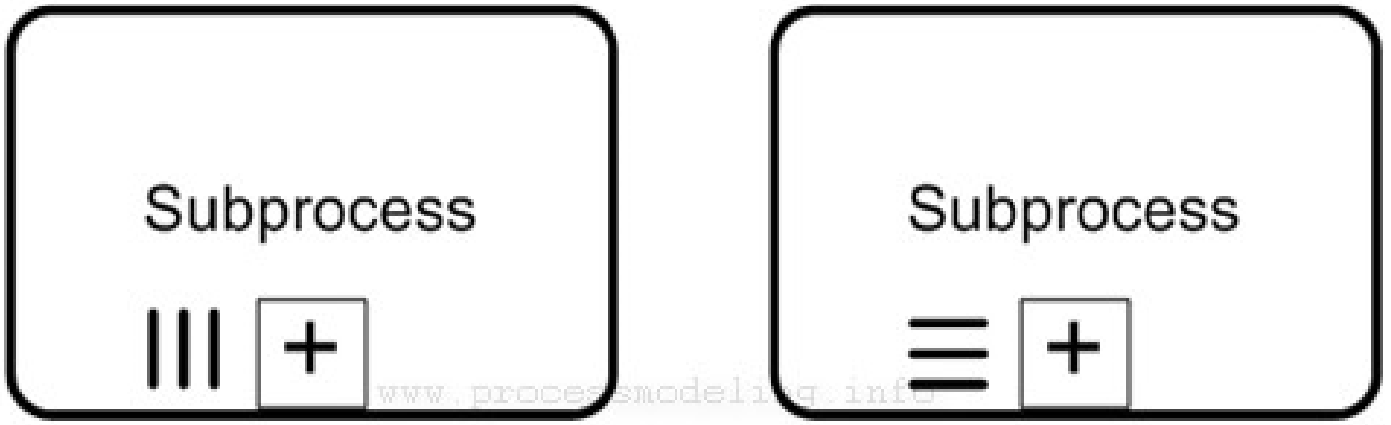
Choreography diagram



- Provides a flowchart view to sequence interactions between Participants
- Choreographies define a “business contract” or protocol to which the Participants agree to follow during real-time interactions.

Activity types - visualised

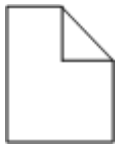
Multi-instance activites - visualised



Parallel

Serial

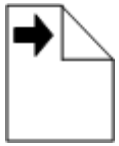
New Artifact Shapes



Data Artifact

Data Artifact Collection

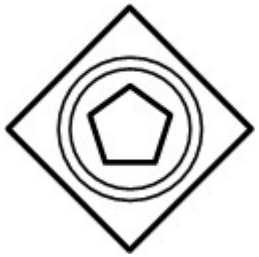
Data Artifact Input



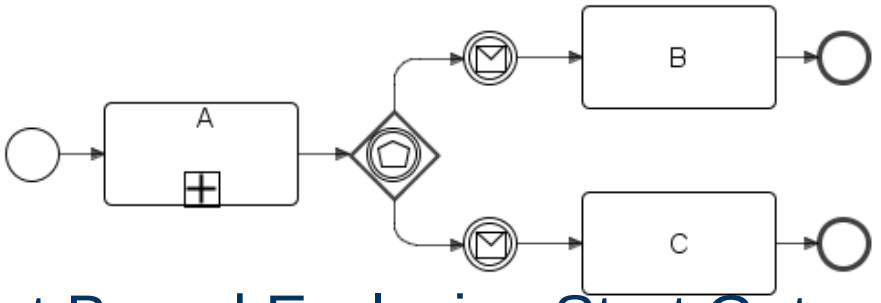
Data Artifact Output

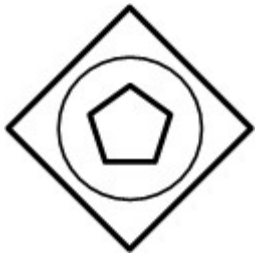
Data Source Artifact

New Event Gateways



- Current event-based gateway
  - Event Based Exclusive Intermediate

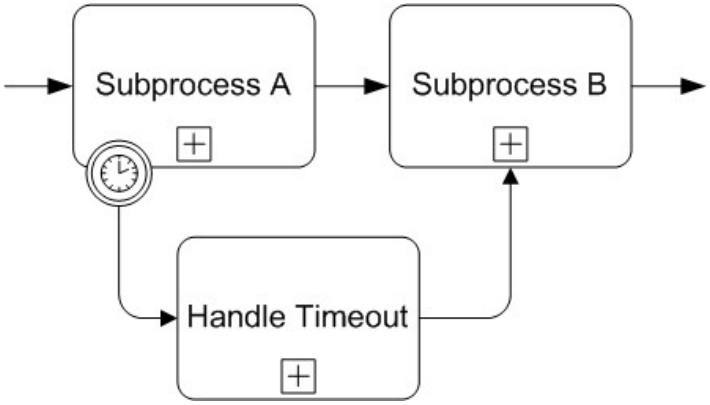


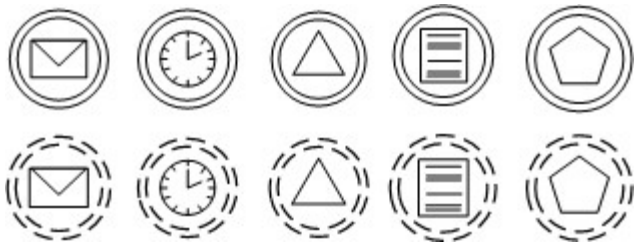


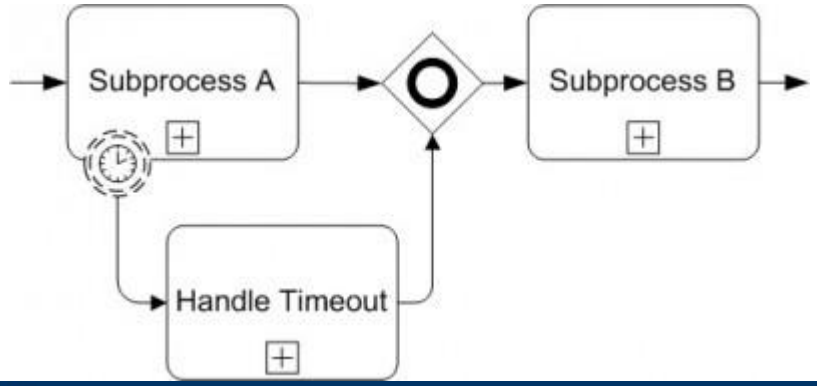
- Event Based Exclusive Start Gateway

- Event Based Inclusive Start
- Event Based Inclusive Intermediate

Non-interrupting Intermediate Events

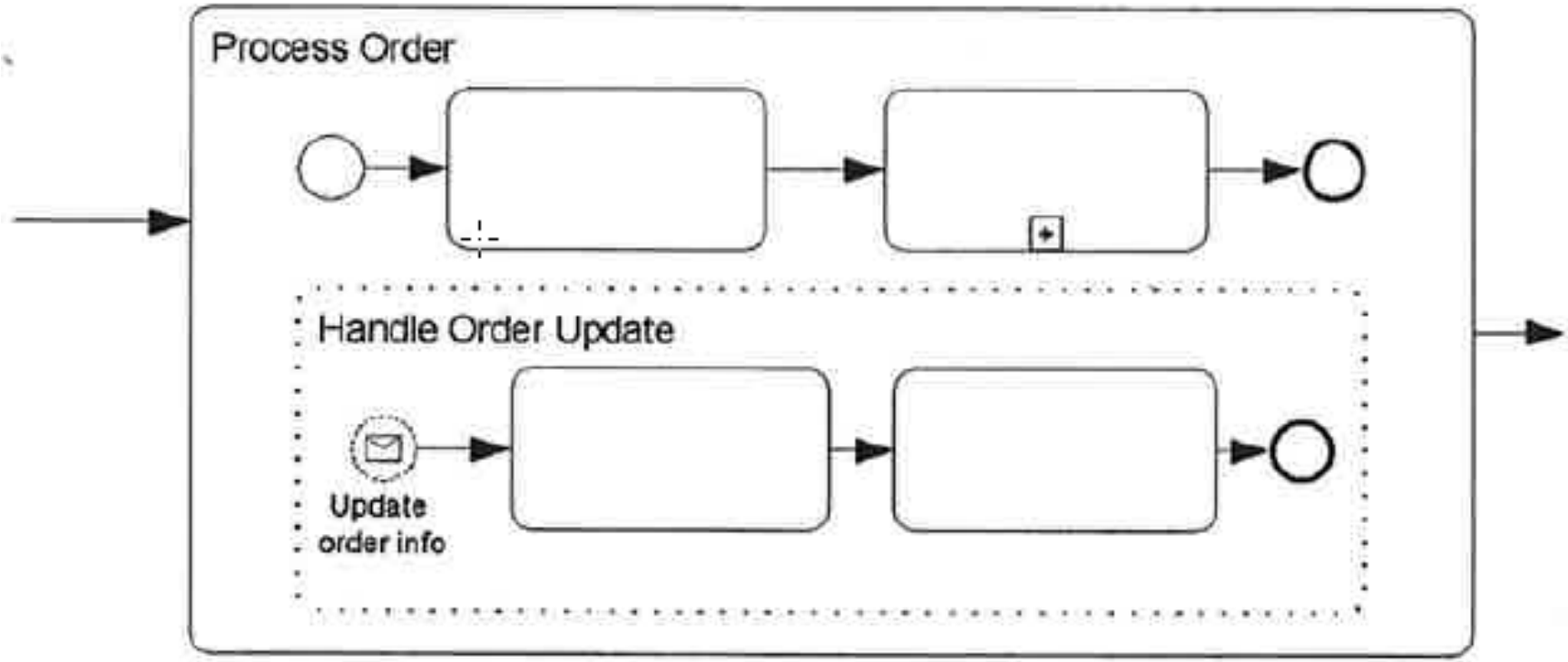




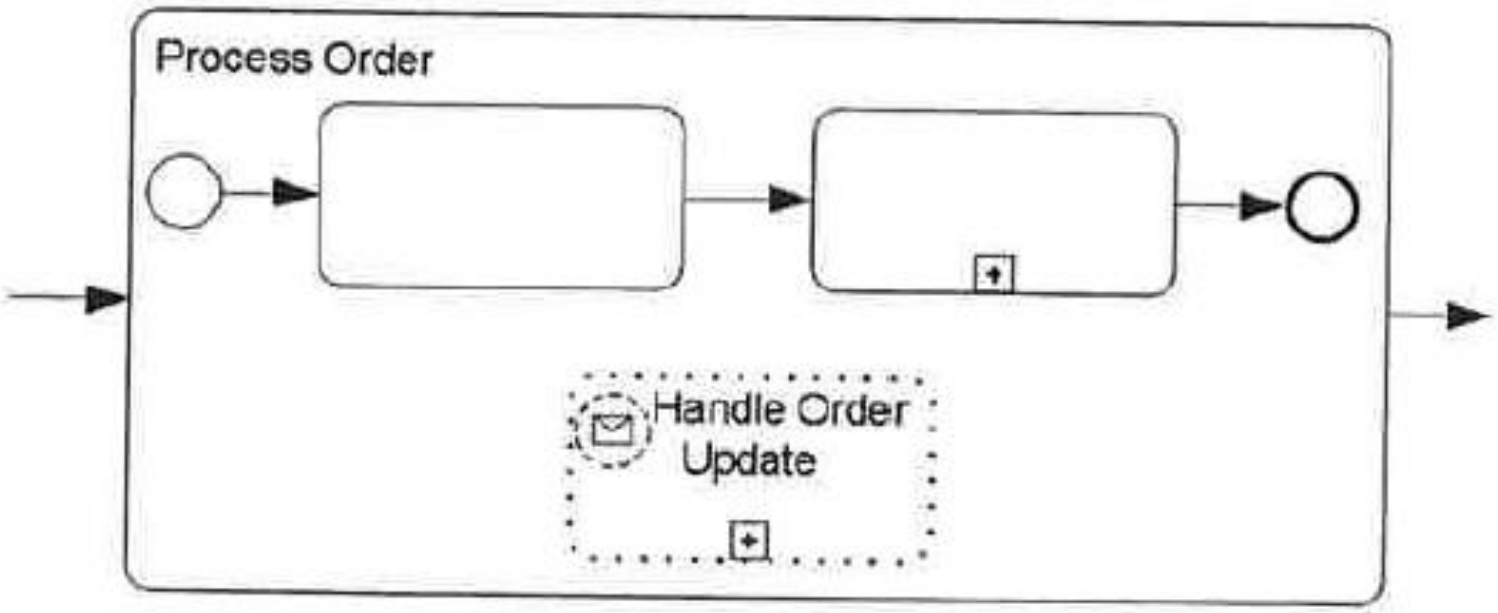


- Boundary intermediate events in BPMN 1.0-1.2 are interrupting
- BPMN2.0 introduces new non-interrupting intermediate events
  - Boundary events
  - Catching
- Example of use

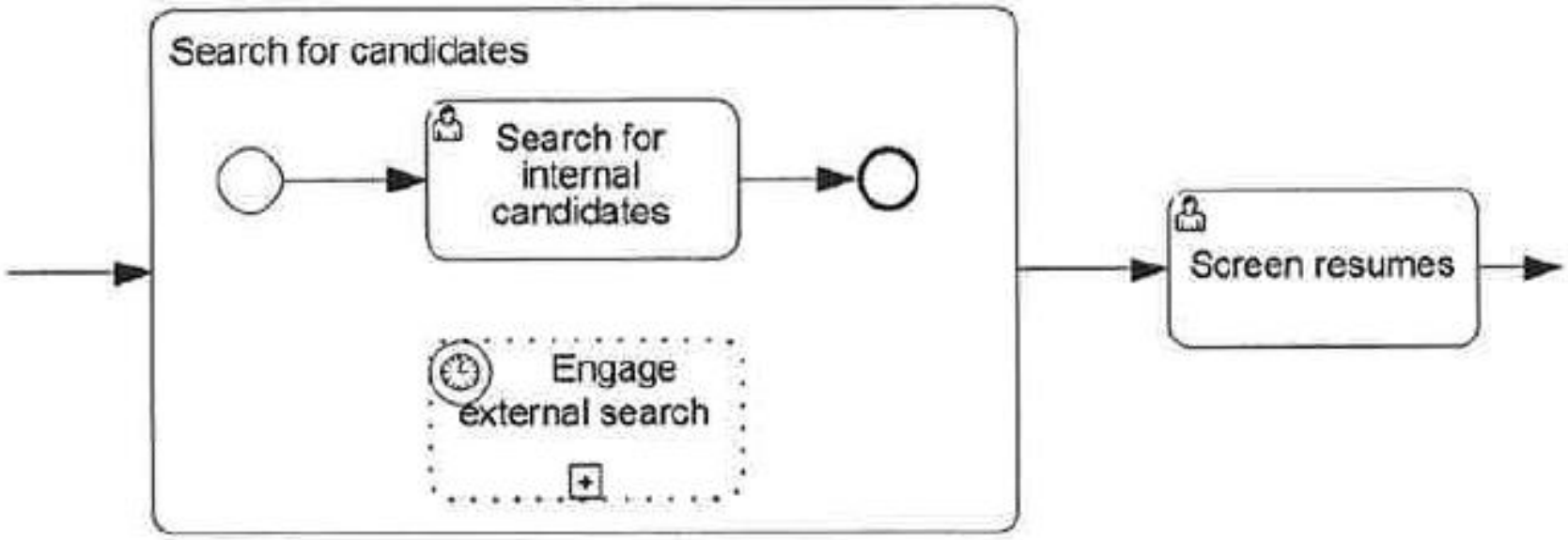
Non-interrupting Event Sub-process (expanded)



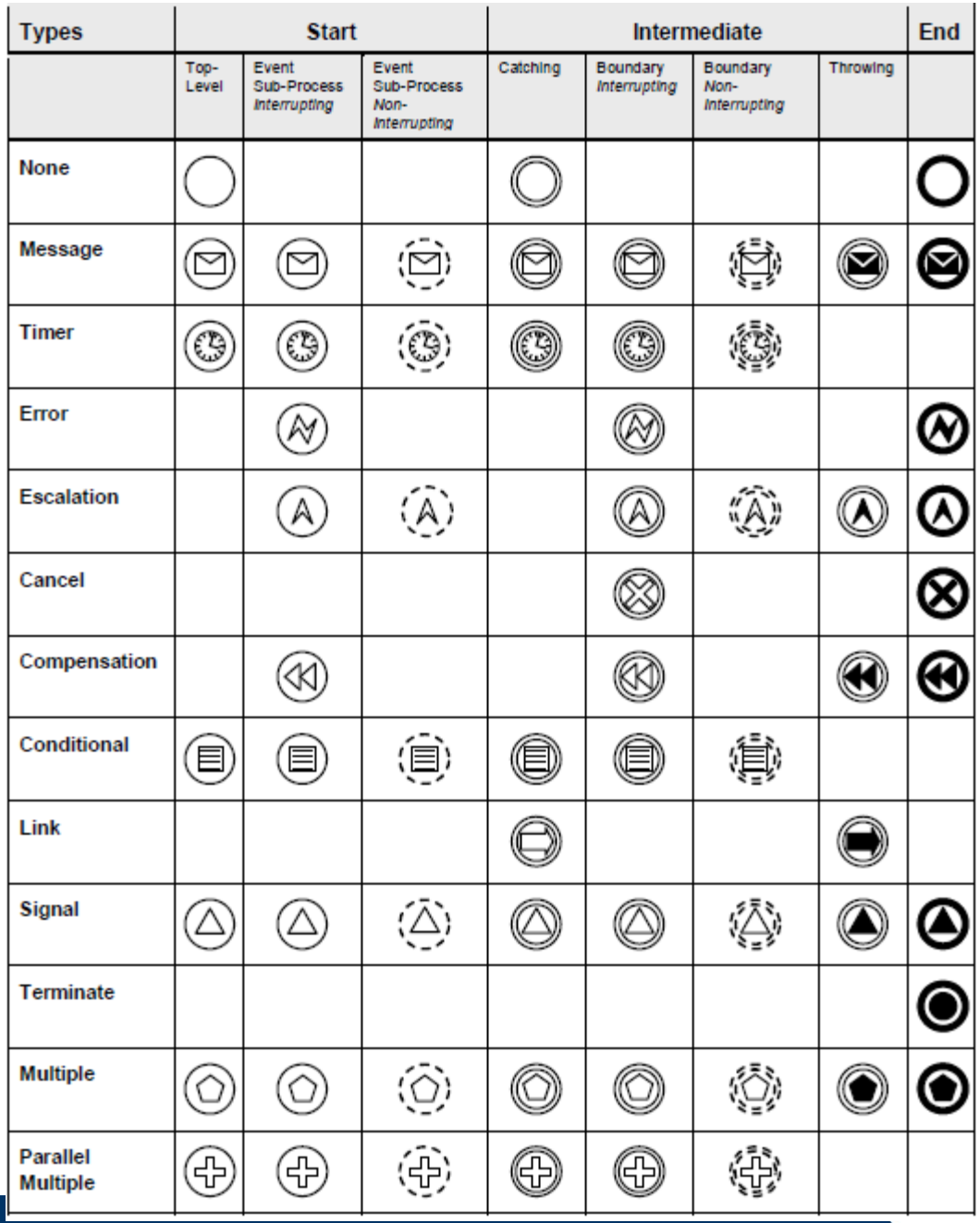
Non-interrupting Event Sub-process (collapsed)



Interrupting Event Sub-process



BPMN 2.0 Event Gallery



Types	Start			Intermediate				End
	Top-Level	Event Sub-Process *Interrupting*	Event Sub-Process *Non-Interrupting*	Catching	Boundary *Interrupting*	Boundary *Non-Interrupting*	Throwing	
None								
Message								
Timer								
Error								
Escalation								
Cancel								
Compensation								
Conditional								
Link								
Signal								
Terminate								
Multiple								
Parallel Multiple								