Business Process Modelling with BPMN

Monique Snoeck
Management Information Systems Group
FEB KU Leuven
monique.snoeck@kuleuven.be

SADL

Agenda

- What are Business Processes ?
- What is Business Process Management ?
- Business Process Modelling Notation
  - Basic notation
  - Extra concepts

SME SPIRE

Business Processes & BP Management

Monique Snoeck
Management Information Systems Group
FEB KU Leuven
monique.snoeck@kuleuven.be

SADL

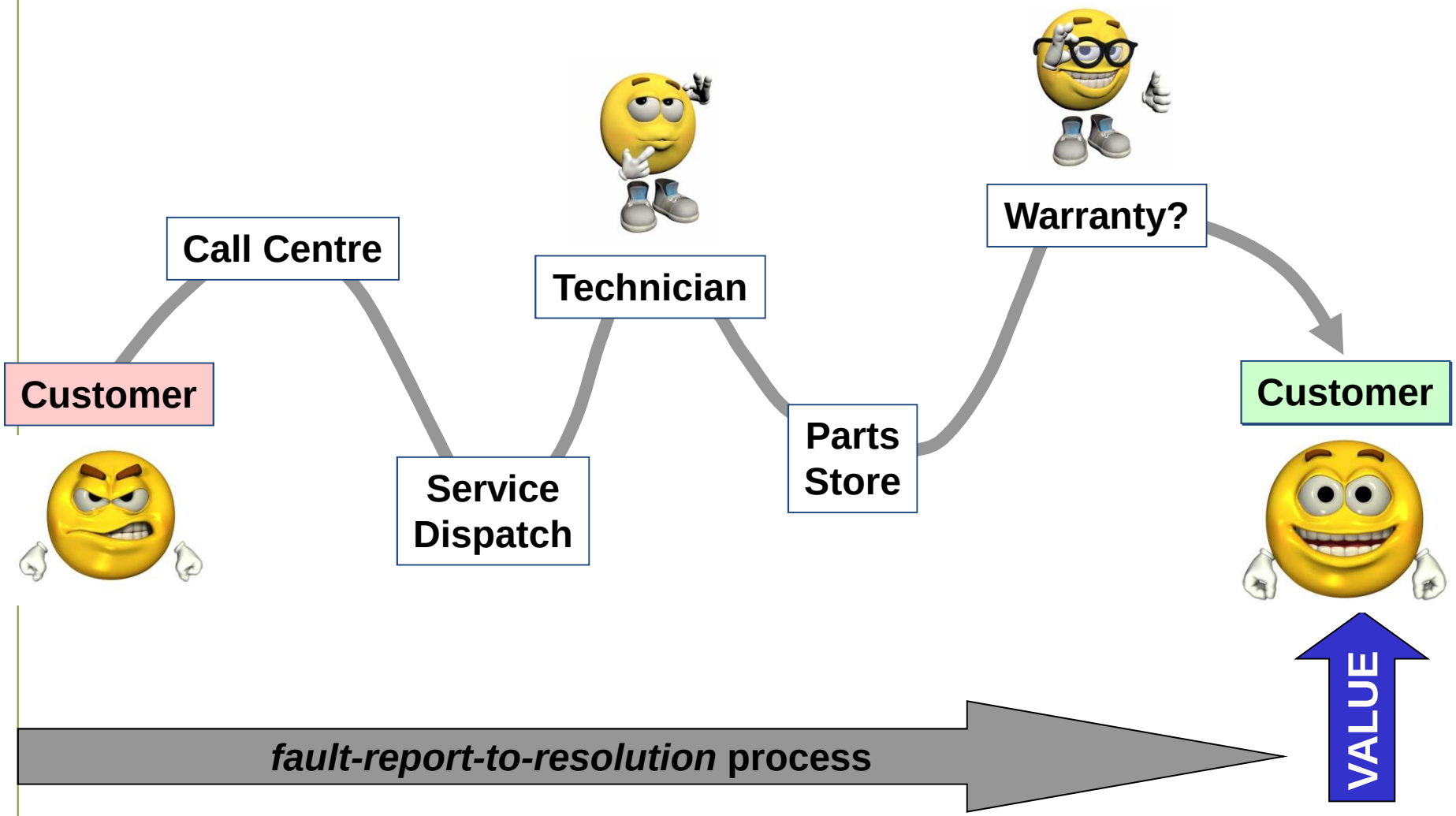
What is a (Business) Process?

Collection of related events, activities and decisions, that involve a number of actors and resources, and that collectively lead to an outcome that is of value to an organization or its customers.

Examples:

- Order-to-Cash
- Quote-to-order
- Procure-to-Pay
- Application-to-Approval
- Claim-to-Settlement
- Fault-to-Resolution (Issue-to-Resolution)

“My washing machine won’t work!”



Processes and Outcomes

- Every process leads to one or several outcomes, positive or negative
  - Positive outcomes deliver value
  - Negative outcomes reduce value
- Fault-to-resolution process
  - Fault repaired without technician intervention
  - Fault repaired with minor technician intervention
  - Fault repaired and fully covered by warranty
  - Fault repaired and partly covered by warranty
  - Fault repaired but not covered by warranty
  - Fault not repaired (customer withdrew request)

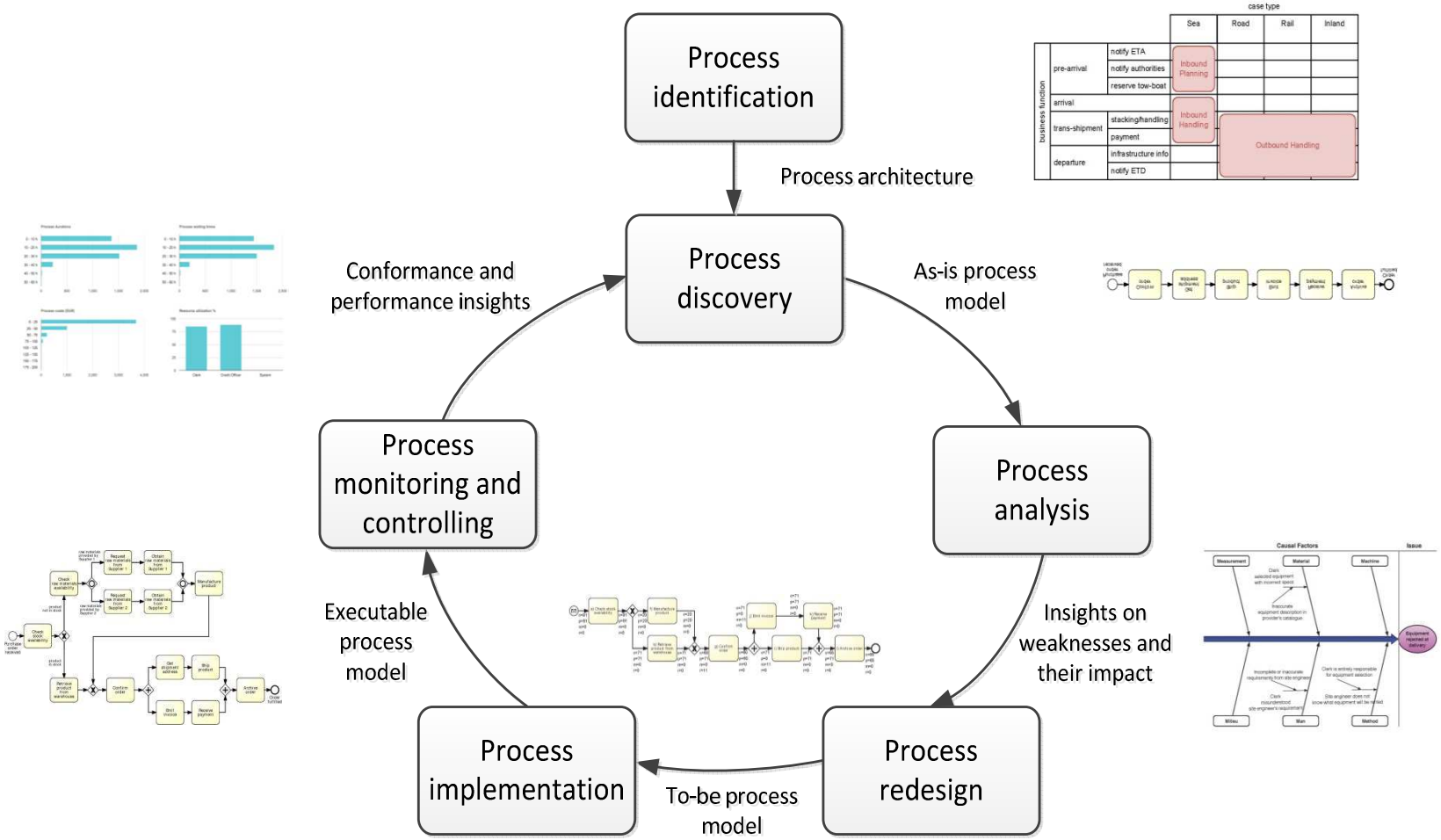
Business Process Management

Body of principles, methods and tools to design, analyze, execute and monitor business processes, with the ultimate goal of improving them.

Dumas, La Rosa, Mendling, Reijers, 2013

Business Process Management

...design, analyze, execute and monitor business processes...

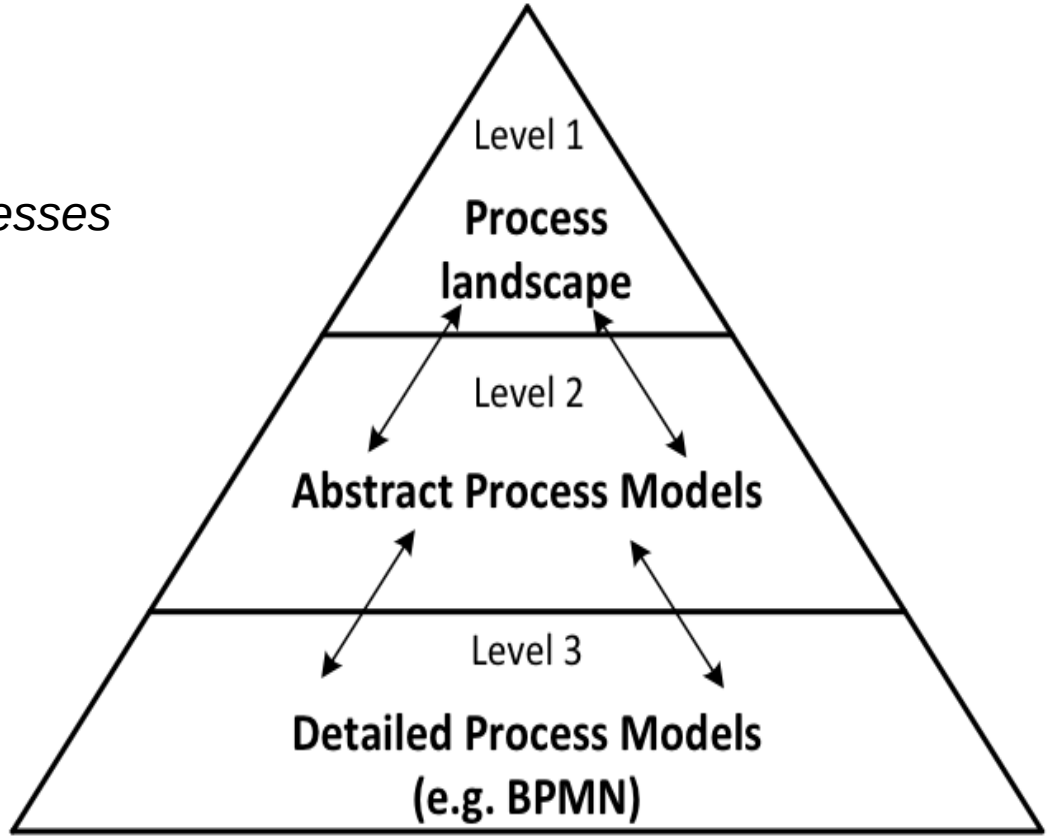


Process Identification

Core processes
Support processes
Management processes

Quote handling
Product delivery
Invoice handling

Detailed quote handling process



Phase 1: Performance Measure Identification and Selection

Cost

- Cost per execution
- Resource utilization
- Waste

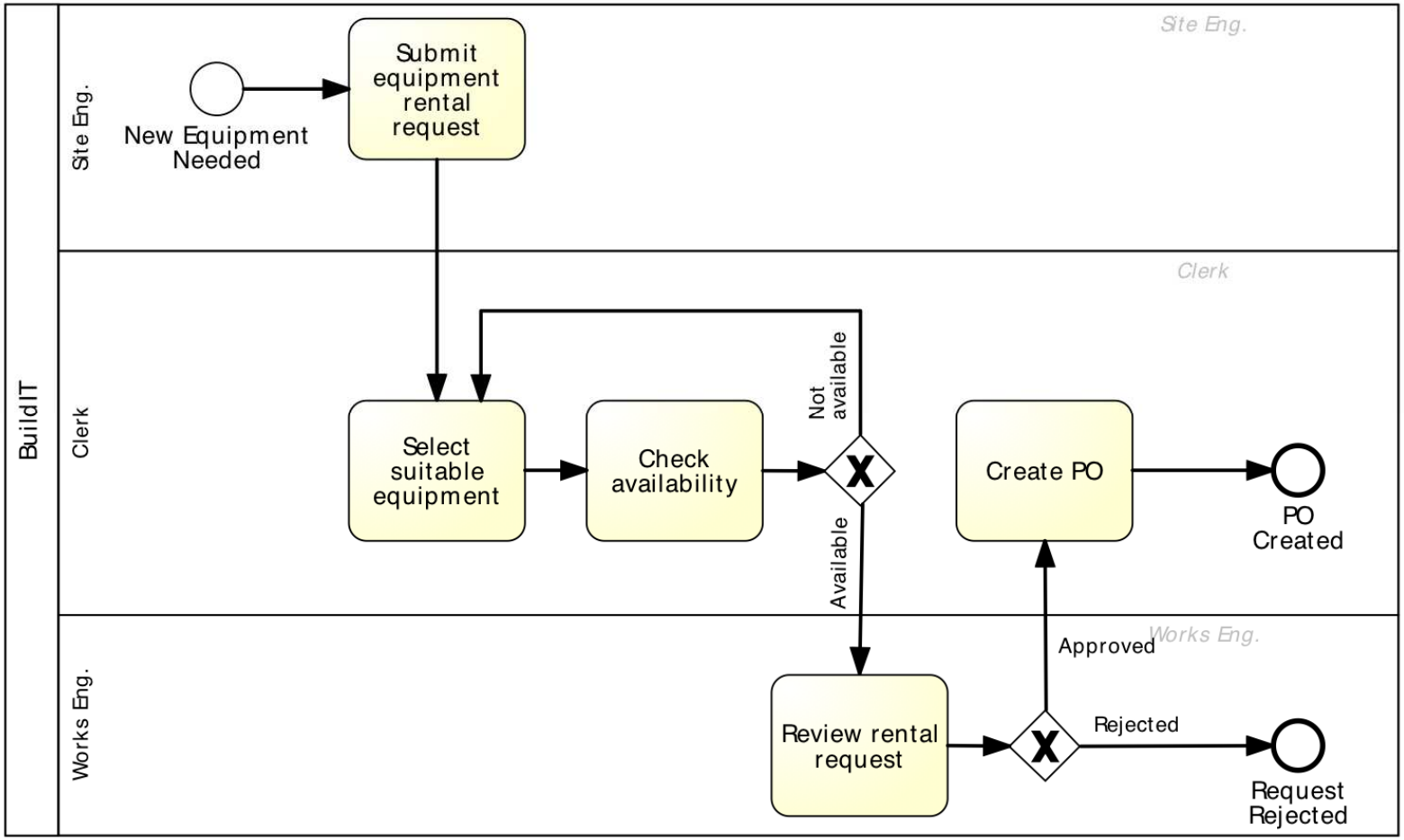
Time

- Cycle time
- Waiting time
- Non-value-adding time

Quality

- Error rates
- SLA violations
- Customer feedback

Phase 2: Process Discovery



Phase 3: Analysis

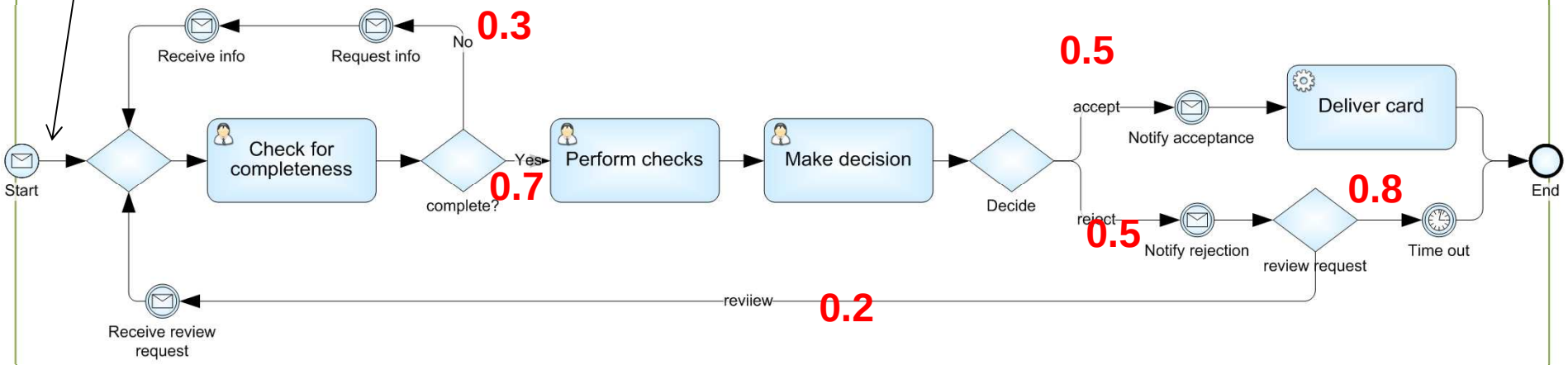
- Model-based
- Qualitative analysis
  - Root-cause analysis
  - PICK charts
  - Issue register
- Quantitative Analysis
  - Flow analysis
  - Queuing analysis
  - Process simulation

Example: Issue Register

Issue No.	Short Description	Issue Explanation	Broad Consequence	Assumptions	Impact
2	Information regarding units does not match	Units in Relocation system do not match information provided by ...	Wrongly calculated entitlements cause manual calculation...	5% of cases go to the wrong queue, 5 minutes to sort queue and redirect. 5% recalculating on average 10 minutes per calculation.	28,000x0.05x15 = 21,000 minutes 350 hours/7.5 47 hrs 9.5 working days
5	Protected/ Mandatory data entry fields	Not all fields in data entry forms are relevant but mandatory. So "fuzzy" information is entered	Resource intensive, incorrect data. Cases in Clarify need to physically be closed.	5% of cases taking 2 minutes to locate and close. 5% of relocations requiring entry that is not needed taking 30 minutes each.	28,000x0.05x32 = 44,800 minutes 477 hours/7.5 99.5 hrs 20 working days
11	Information on posting orders	Time consuming to sort through posting orders to identify relocations....	MBR does not get Info pack therefore cannot process move. More information could be provided which could be used later in process ...	Only 1/3 rd of postings and CIPC's are entitled to relocation. 28000 relocations then sorting through 84000 postings. 3 to 4 minutes on average to sort through each.	84,000x3.5 = 294,000 min/60/7.5 = 653 days /250 working days in year. 2.61 FTE

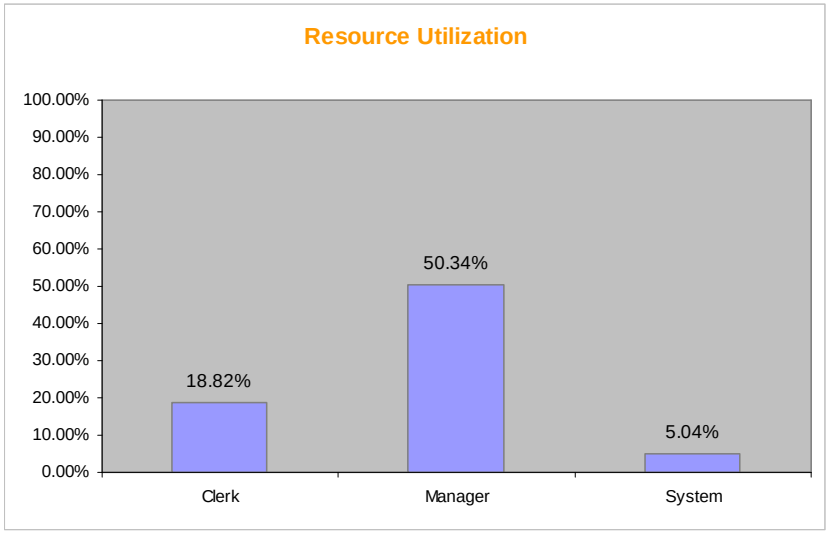
Example: Simulation + What-If Analysis

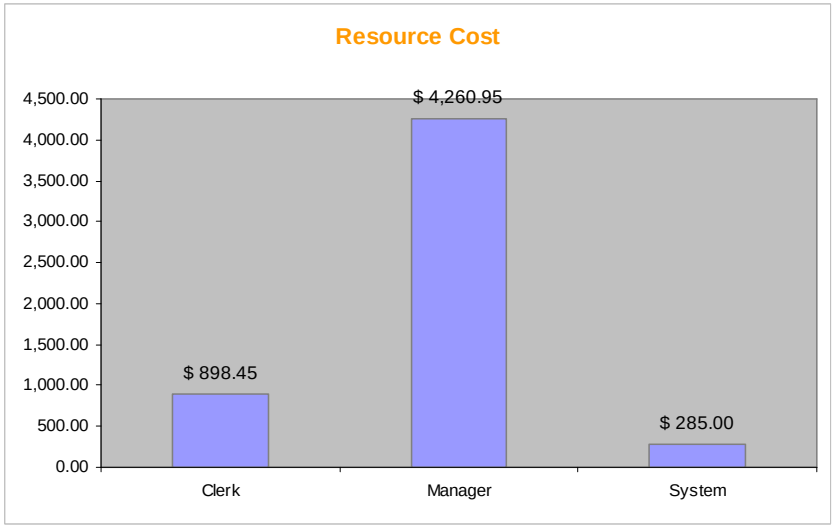
10 applications per hour
Poisson arrival process
(negative exponential)

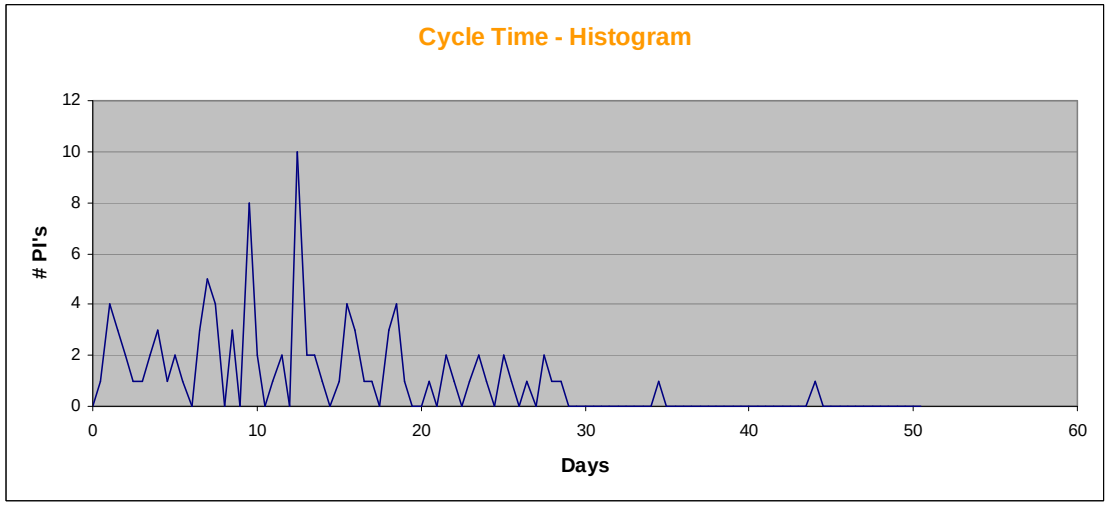


Task	Role	Execution Time (mean, dev.)	
Receive application	system	0	0
Check completeness	Clerk	30 mins	10 mins
Perform checks	Clerk	2 hours	1 hour
Request info	system	1 min	0
…	…	…	…

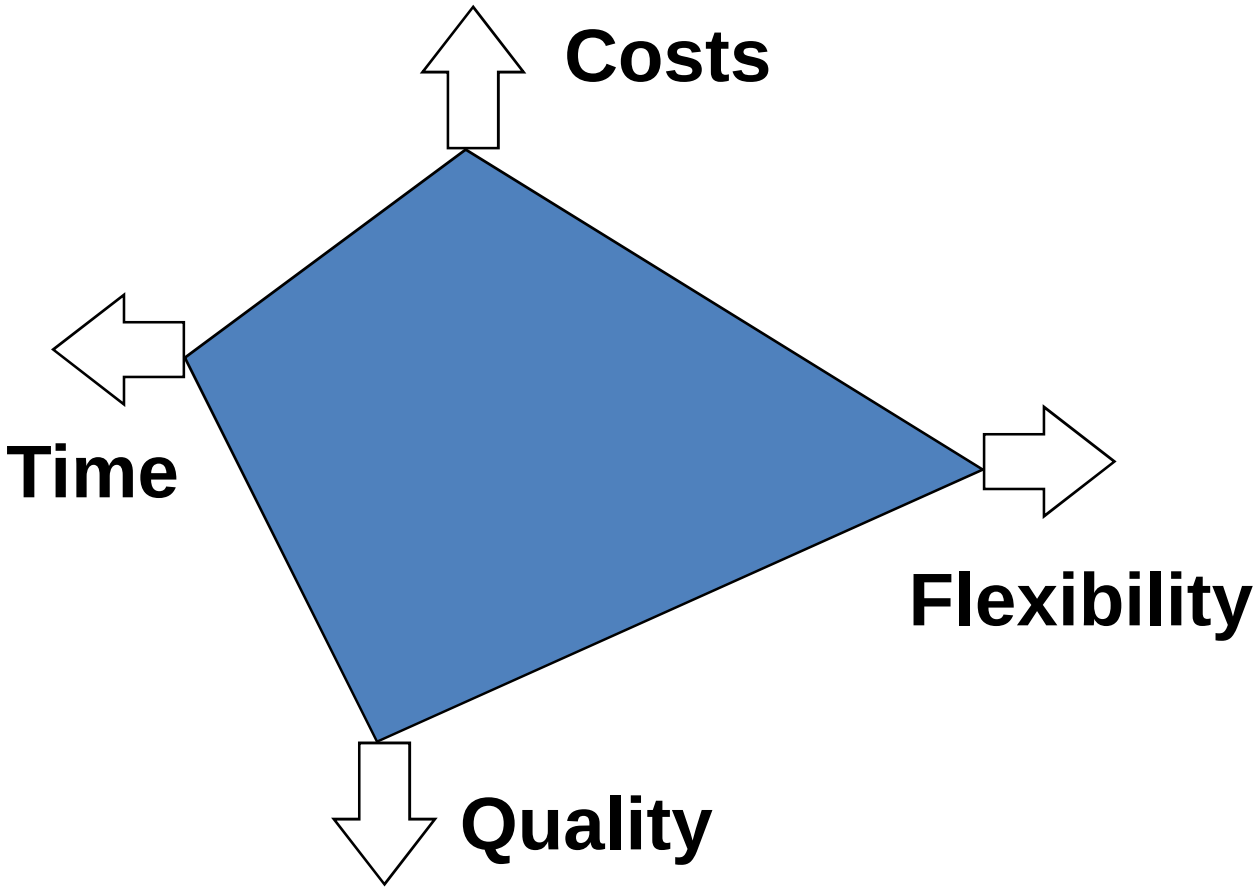
Example: Simulation output (KPIs)





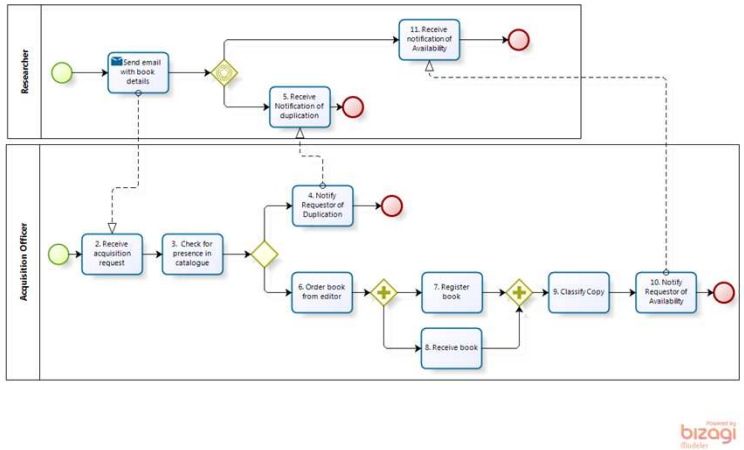


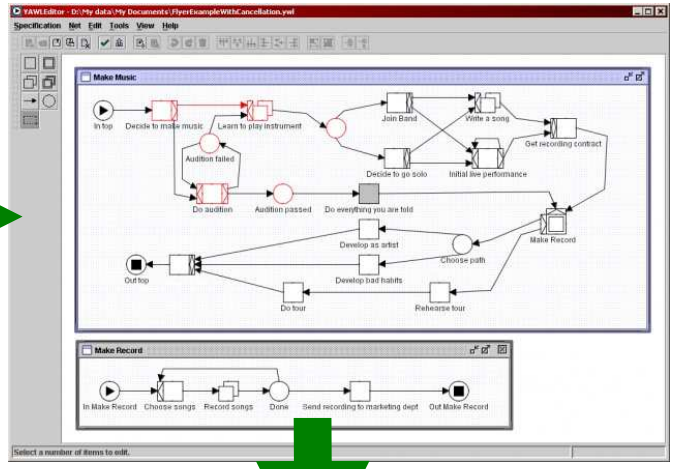
Phase 4: Process Re-Design

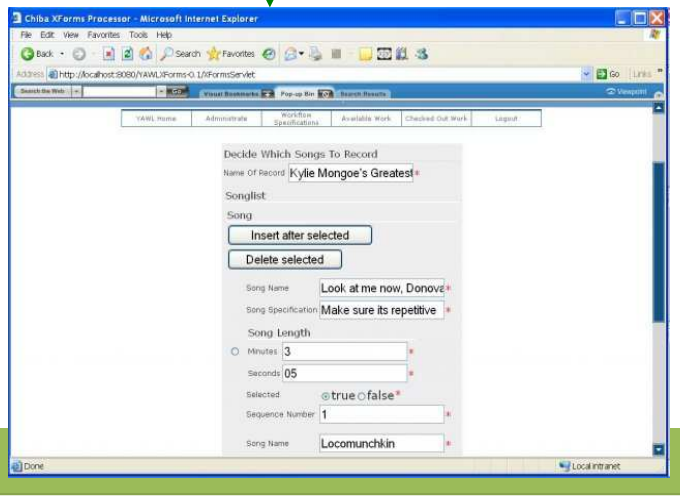


Phases 5: Business Process Implementation

- = When technology Kicks in…







Phase 6: BP Monitoring & Control

- Business Activity Monitoring (BAM)
  - The aggregation, analysis, and presentation of real time information about activities or processes inside organizations.
    - Customer satisfaction, Product or service quality
    - Percent of late deliveries, Number of errors Number of minutes per order, Cost per order
    - Percent of time devoted to rework , Number of people or resources involved in a process
    - Time to complete case
  - Is a driver for business process improvement!

Figure 1: Rich thin-client GUI-PeopleSoft real-time BAM dashboard

SME SPIRE

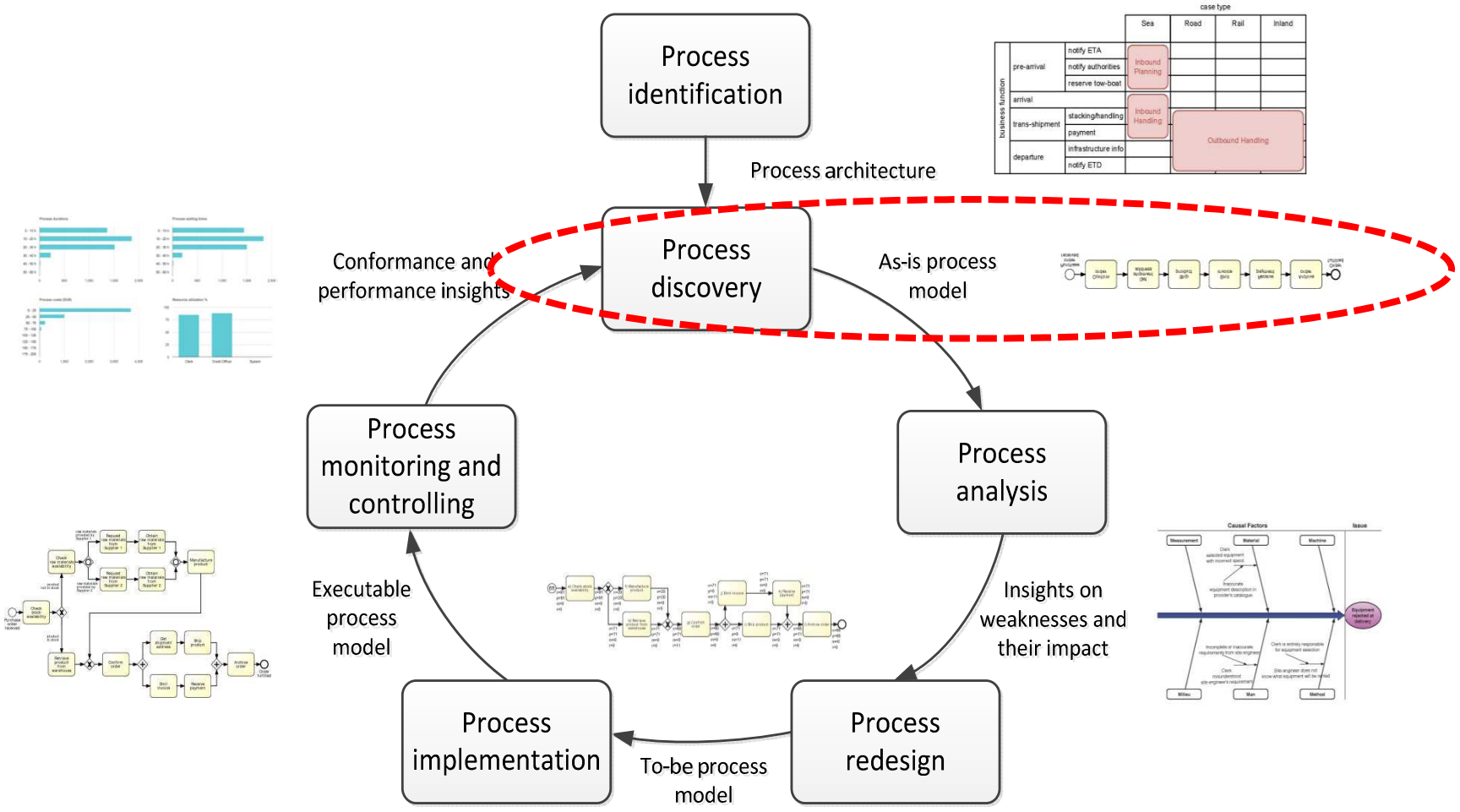
Process Modelling basics

Monique Snoeck
Management Information Systems Group
FEB KU Leuven
monique.snoeck@kuleuven.be

SADL

Business Process Management

...design, analyze, execute and monitor business processes...



Why Process Modelling?

- Business processes are increasingly valued as essential assets of an organisation
- This significance demands dedicated management of processes
- We need ways to extract processes out of the organisational complexity in order to discuss, analyse, improve and automate them

Why Process Modelling…

- …discuss, analyse, improve and automate processes
- ➔ Modelling is at the basis of BP Maturity and Improvement

- Drawing a BP diagram
  - requires to understand how we work today (as-is)
    - enables us to analyse how we work
  - enables us to define how we want to work in the future (as-is)
    - enables us to simulate/analyse our planned reforms
  - enables us to communicate how we work / should work
  - enables us to develop automated support
    - enables us to do business activity monitoring
    - enables us to check/analyse our (real) performance
    - enables us to suggest improvements
  - ….

The Core Elements of a Process

- Activities
  - (e.g. ‘enter sales order’)
  - time-consuming, resource-demanding
  - state-changing

- Events
  - (e.g. ‘sales order has been entered’)
  - represent conditions / circumstances
  - atomic, instantaneous

The Core Elements of a Process

- Business Objects (or Data)
  - the organizational artifacts that undergo state changes
  - physical or electronic information
  - examples:
    - sales order, digital object, consulting proposal
- Actors (or Resources)
  - the entities performing process activities and generating events
  - human and systems
  - examples:
    - financial officer, warehouse clerk
    - ERP, CRM, SAP, application X…

Further Potential Elements in a Process

- Objectives, Goals
  - link to strategy
- Risks
  - for risk-profiling the process
- Policies, Rules
  - for checking process compliance
- Knowledge
  - to depict expertise required
- …

Modelling of Agents: RACI

- Agents: RASCI
  - R =Responsible Who is responsible for executing the activity?
  - A =Accountable Who is accountable and needs to report ?
  - (S =Supportive Who can give support ?)
  - C =Consulted Who needs to be consulted ?
  - I = Informed Who needs to be informed ?

- Example: granting a dispensation for a course
  - R = clerk in Student office
  - A = Program Director
  - S = Legal councelor
  - C = teacher of course
  - I = teacher of course, student

SIPOC

- Origin = Six Sigma
  - acronym for Suppliers - Inputs - Process - Outputs – Customers
  - high-level picture of the process that depicts how the given process is servicing the customer

- Suppliers
  - provide inputs to the process.
- Inputs
  - define the material, service and/or information that are used by the process to produce the outputs.
- Process
  - is a defined sequence of activities, usually adds value to inputs to produce outputs for the customers.
- Outputs
  - the products, services, and/or information that are valuable to the customers.

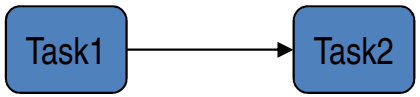
- Customers
  - are the users of the outputs produced by the process.

SIPOC for Grant Dispensation

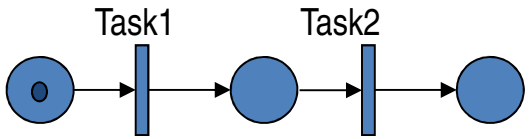
Supplier	Input	Process	Output	Customer
Student Teacher Legal Advisor	Course descriptions of followed course + course for which disp. is asked Legal regulations Archive of prior decisions	Steps undertaken to take decision e.g. 1. Lookup archive 2. If found, communicate decision 3. Request advise teacher 4. Request advise PD 5. Inform student 6. Add decision to archive	Decision on Dispensation	Student

Modelling perspectives

- Prescriptive (Focus on control flow!)
  - Tasks (BMPN)



  - Tasks and States (Petri Nets)

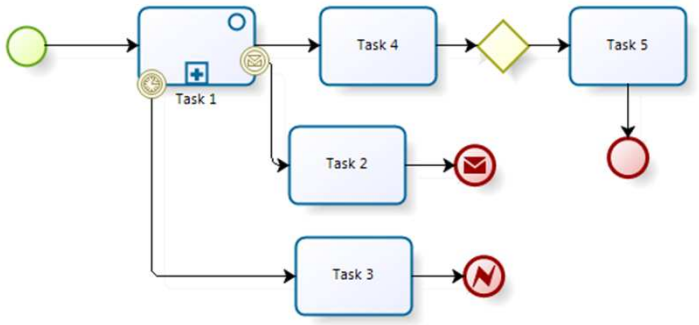


- Good for simple and structured processes!

- Descriptive
  - Organisational focus
    - RACI - SIPOC
  - Business Rules
    - Autorisation, input validation, sequence constraints, policy, …
    - Preconditions
      - When can an activity start ?
      - ship when paid
    - Postconditions
      - ensures result
      - when shipped: shipment registered
- Easier for complex processes

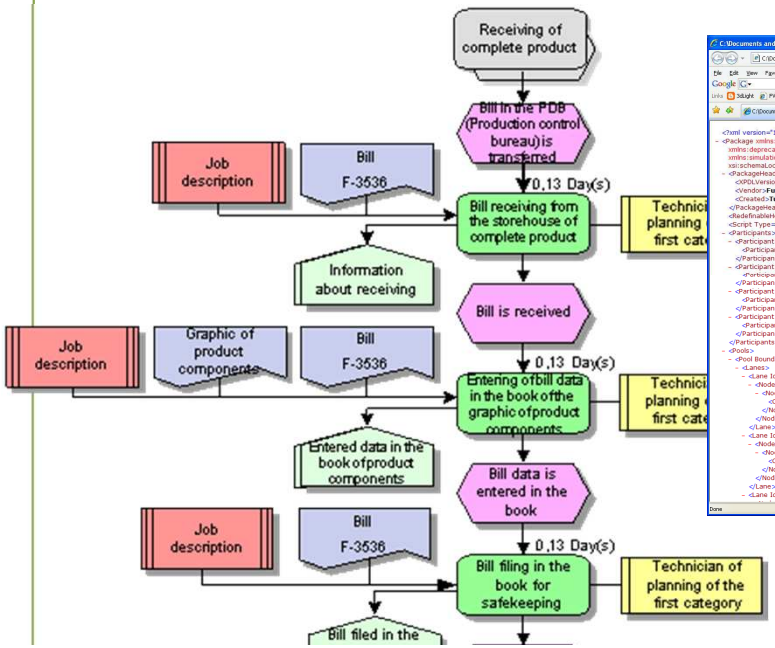
BP Control flow modelling techniques

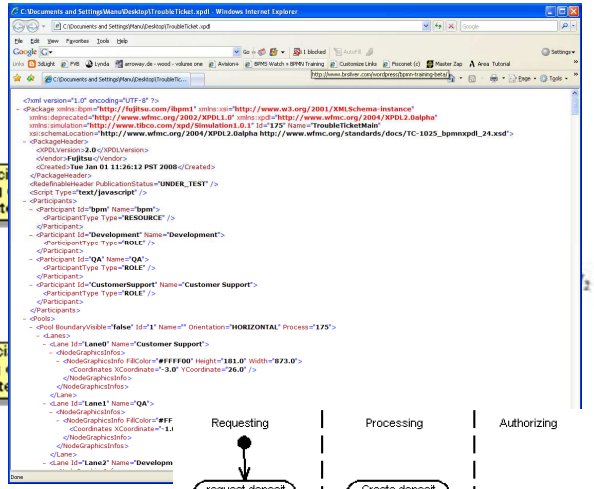
- BPMN – Business Process Modeling Notation

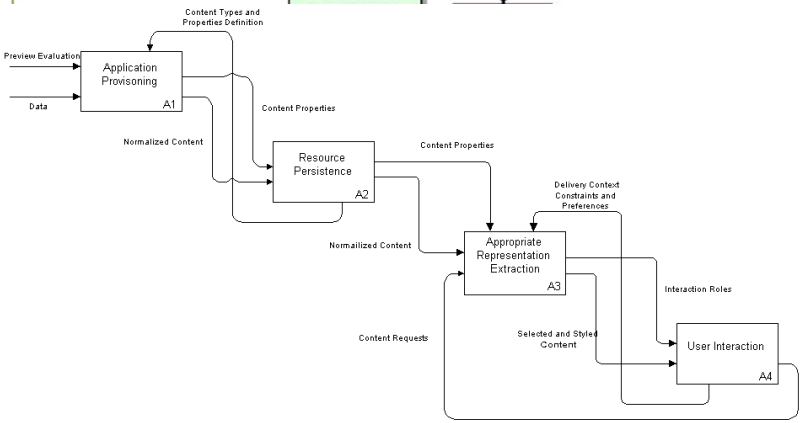


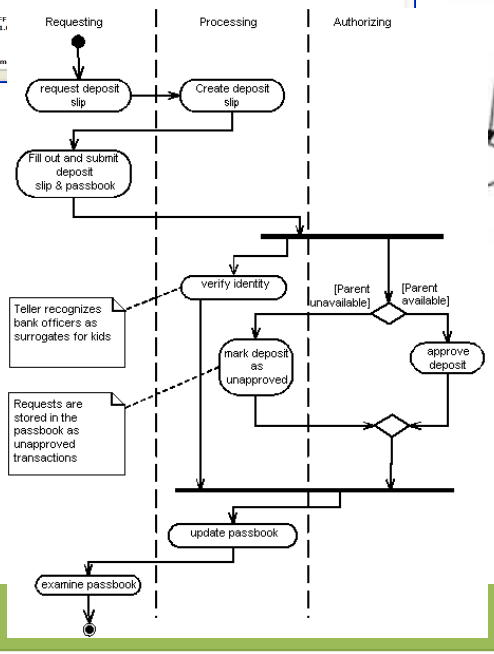
- Petri Net Theory
- UML – Activity Diagrams
- EPC – Event Driven Process Chains
- XPDL – XML Process Definition Language
- IDEF – ICAM Definition Language
- BPDM – Business Process Definition Meta-model
- EPML – EPC Markup Language
- Flowcharts
- BPEL – Business Process Execution Language
- …

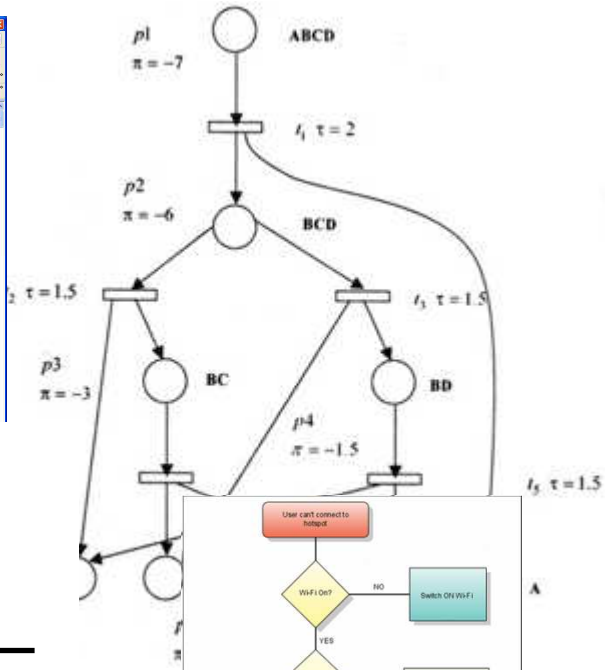
Some other Control Flow oriented notations

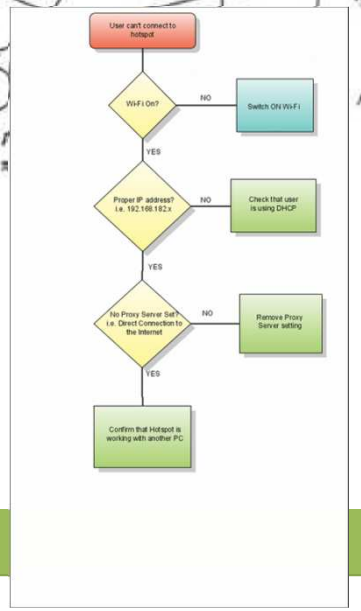












SME SPIRE

BPMN basics

Monique Snoeck
Management Information Systems Group
FEB KU Leuven
monique.snoeck@kuleuven.be

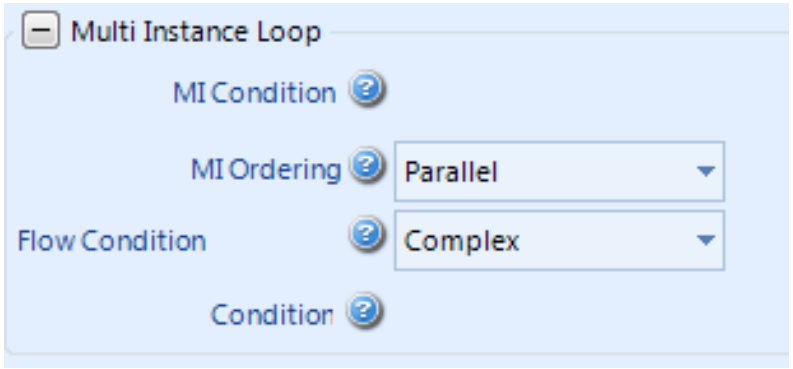
SADL

What is BPMN

- BPMN 2.0 introduction:
  - "This specification represents the amalgamation of best practices within the business modeling community to define the notation and semantics of **Collaboration** diagrams, **Process** diagrams, and **Choreography** diagrams. The intent of **BPMN** is to standardize a business process modeling notation in the face of many different modeling notations and viewpoints. In doing so, **BPMN** will provide a simple means of communicating process information to other business users, process implementers, customers, and suppliers."
- Three basic types of sub-models within an end-to-end **BPMN** model:
  - **Processes** (Orchestration), including:
    - *Private Non-executable* (*internal*) **Business Processes**
    - *Private Executable* (internal) **Business Processes**
    - *Public* **Processes**
  - **Choreographies**
  - **Collaborations**, which may include **Processes** and/or **Choreographies**
    - A view of **Conversations**

What is BPMN?

- "Not a Pure Graphical notation"
  - Consists of 3 basic shapes (+ some extensions)
    - And a total of ±52 graphical elements
  - BPMN 2.0 has a formally defined execution semantics
    - …and can be mapped to BPEL
    - BPMN 2.0 is an XML language for executable process design
  - Able to handle complexity (sub-processes, expand, collapse)
  - + many invisible attributes



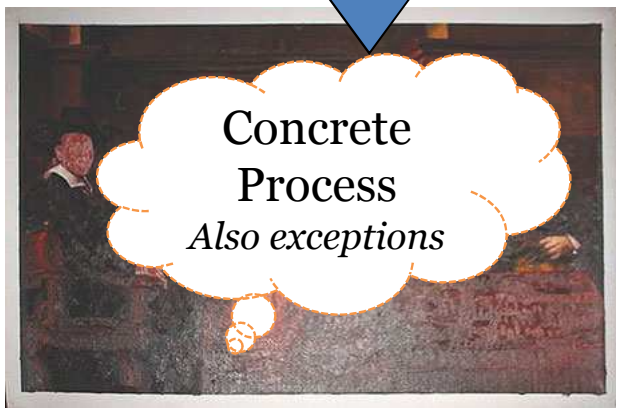
Approach: work iteratively

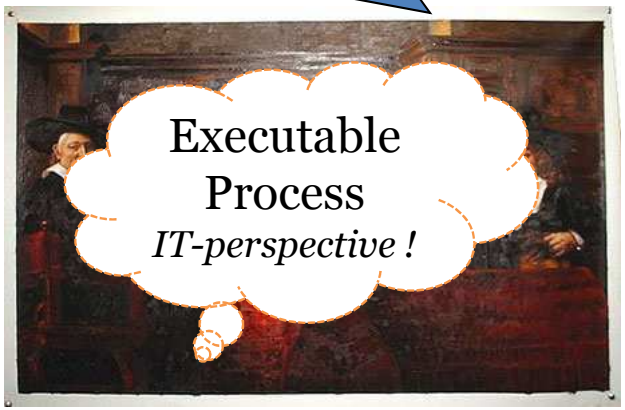
PLAN

Iteration 1

Iteration 2

Iteration N



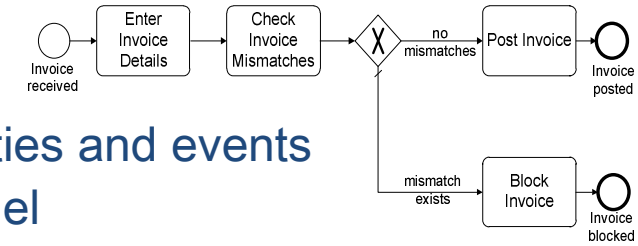


Different types of BPMN Models

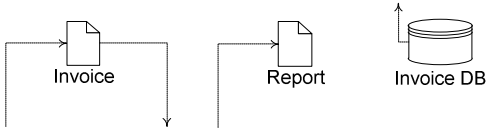
- Level 1: Descriptive BPMN
  - Limited set of symbols
  - Hierarchical, top-down modelling
  - Focus on understanding and handling complex real-world processes
  - Focus: Business
- Level 2 : analytical BPMN
  - Full palette of symbols, incorporates Events, Exception Handling
  - Emphasises consistency and technical meaning of shapes
  - Focus: shared meaning between Business & IT ➔ refines level 1 process until sufficient detail is achieved to be useful for IT.
- Level 3: executable BPMN
  - XML language underneath BPMN ➔ *out of scope for this course*

BPMN combines perspectives

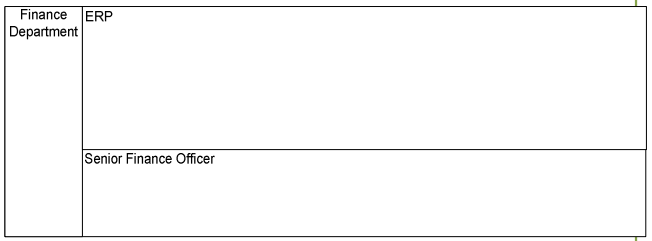
- Control Flow Perspective
  - “what needs to be done and when”
  - predecessor/successor relationship among activities and events
  - the central information depicted in a process model



- Data Perspective
  - “what do we need to work on”
  - input/output data to activities
  - complements the control flow

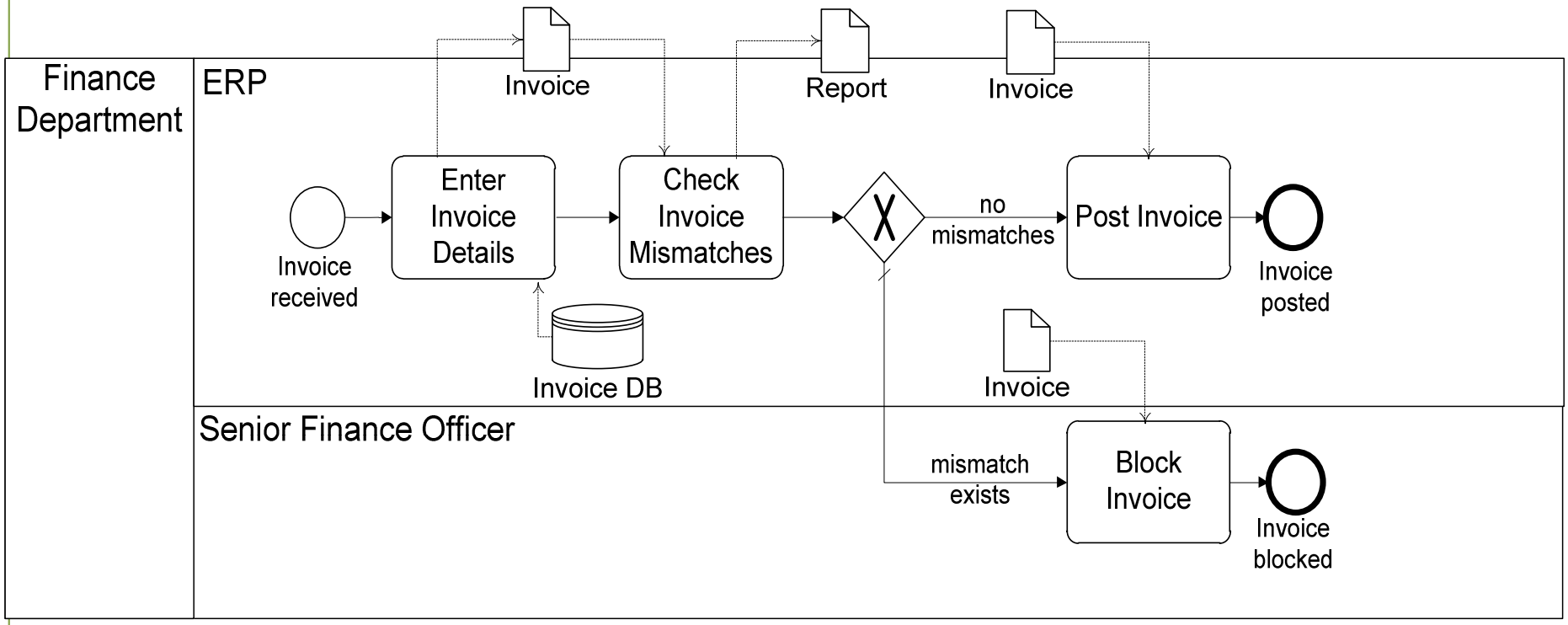


- Resource Perspective
  - “who’s doing the work”
  - human participants and systems that perform control flow activities and generate events
  - complements the control flow

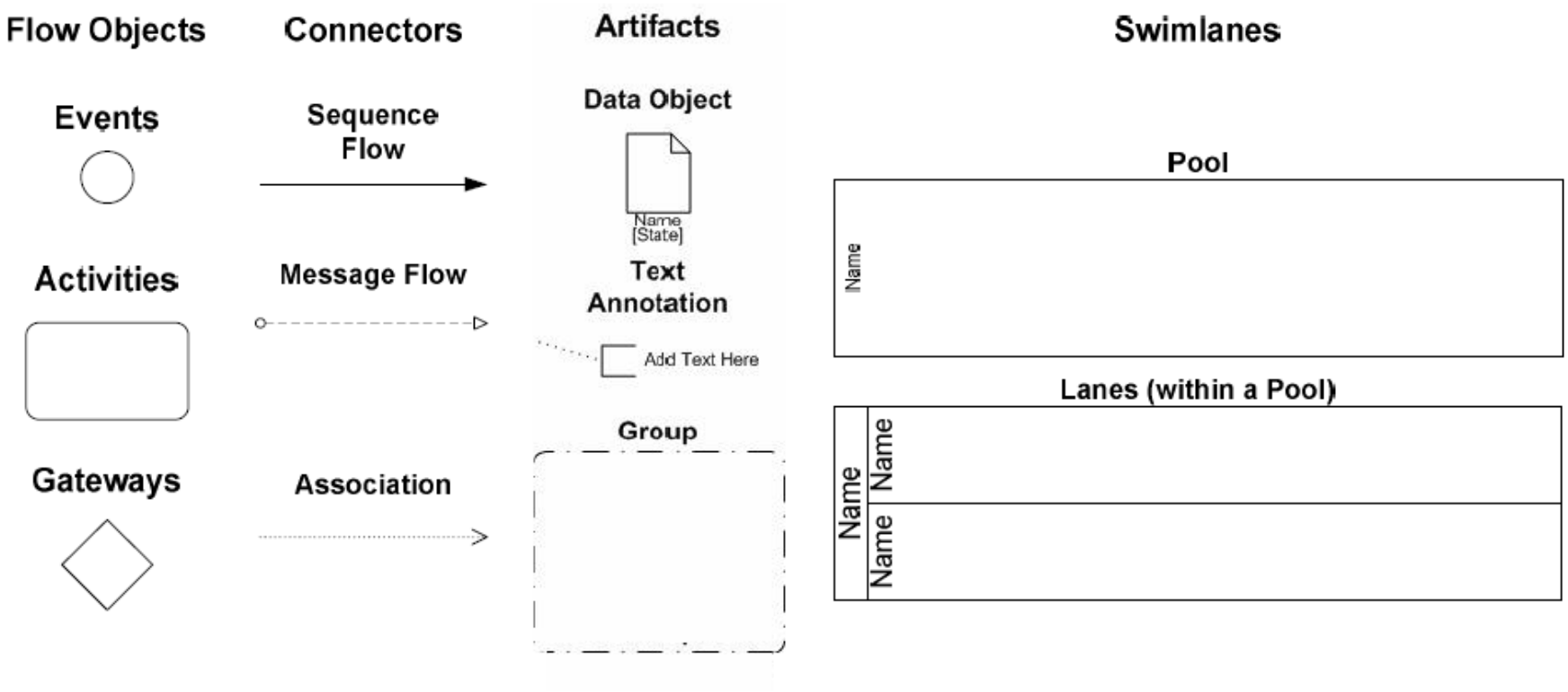


BPMN combines perspectives

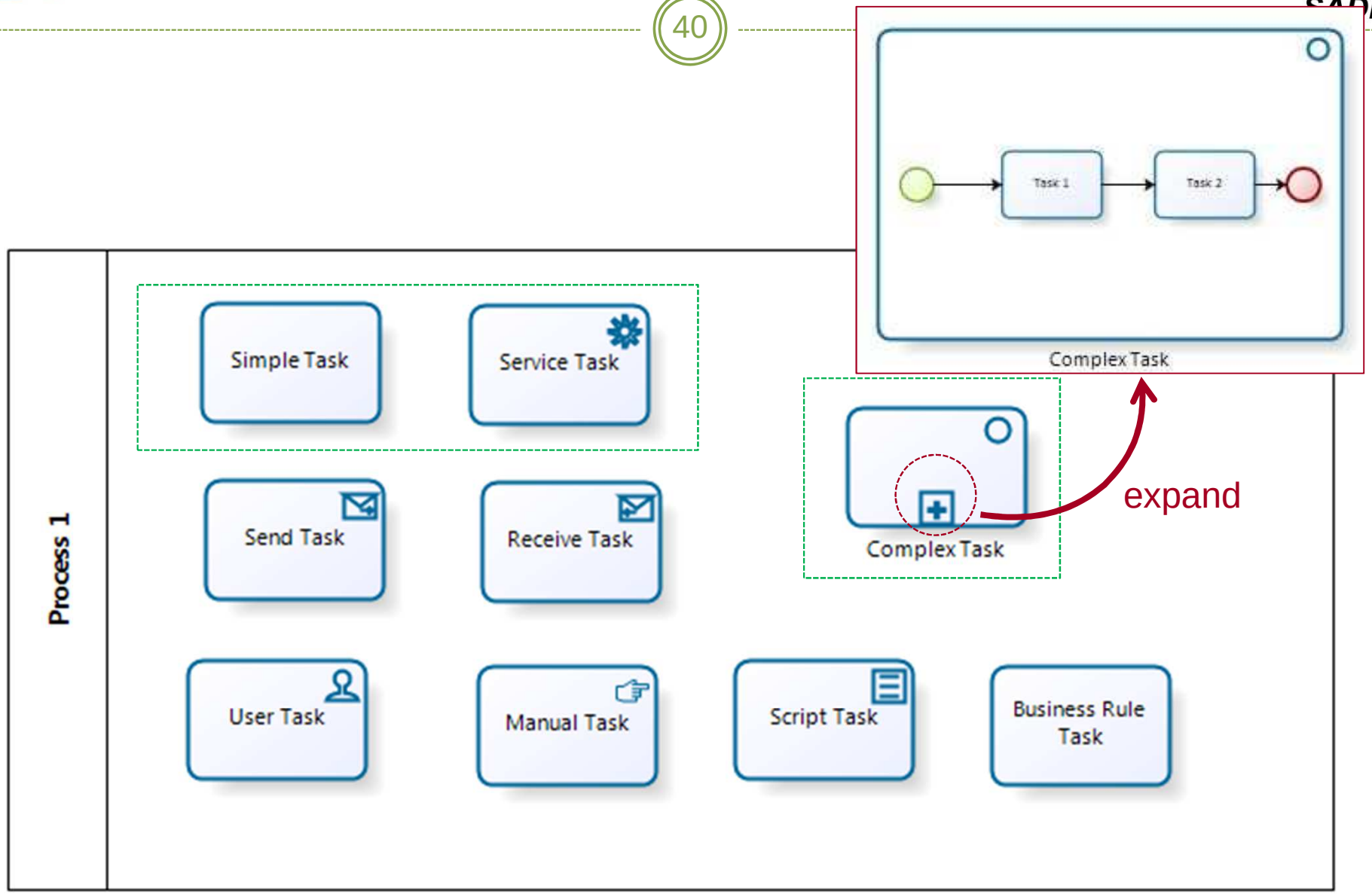
1. What needs be done and when? - *Control flow*
2. What do we need to work on? – *Data*
3. Who's doing the work? - *Resources (human & systems)*



BPMN diagram elements



BPMN Activities



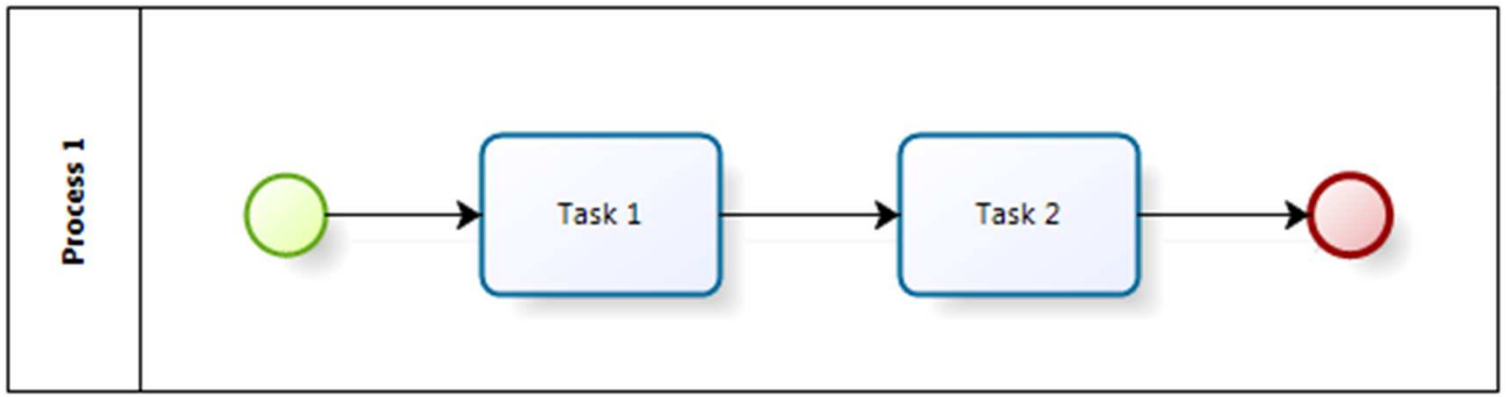
BPMN subprocesses

- Visualise end-to-end process
- Enable top-down modelling
- Clarify governance boundaries
- Scope event handling

- CALL Activity
  - Subprocess with a thick border
  - “independent” subprocess
  - = reusable or “global” task
  - (↔ subprocess embedded and only know in encompassing superprocess).

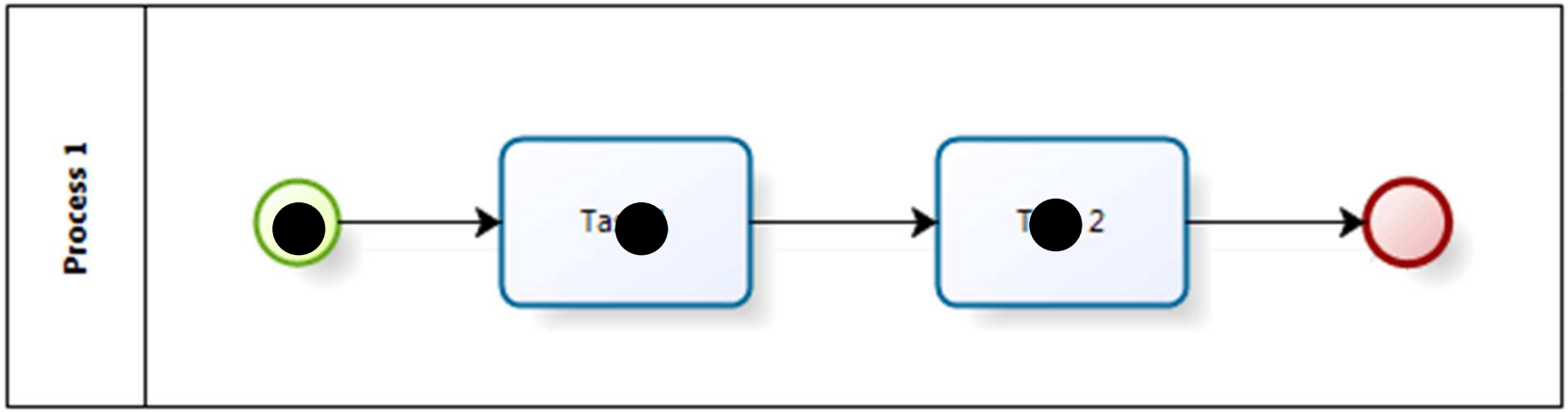
Task sequencing

- Sequence flow: 



Understanding process execution

- “Token” symbolises a process instance ●

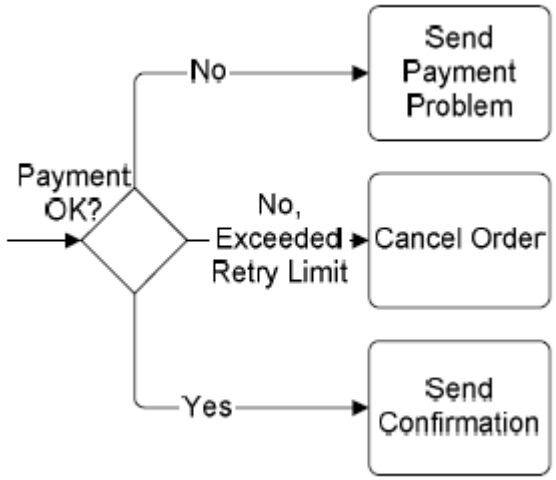


BPMN gateways

- Diamond represents place where control is needed
- Different types:
  - (exclusive) Data-based OR 
  - Event-based OR 
  - AND 
- Used for
  - Splitting
  - Merging

BPMN data based OR gateways (split)

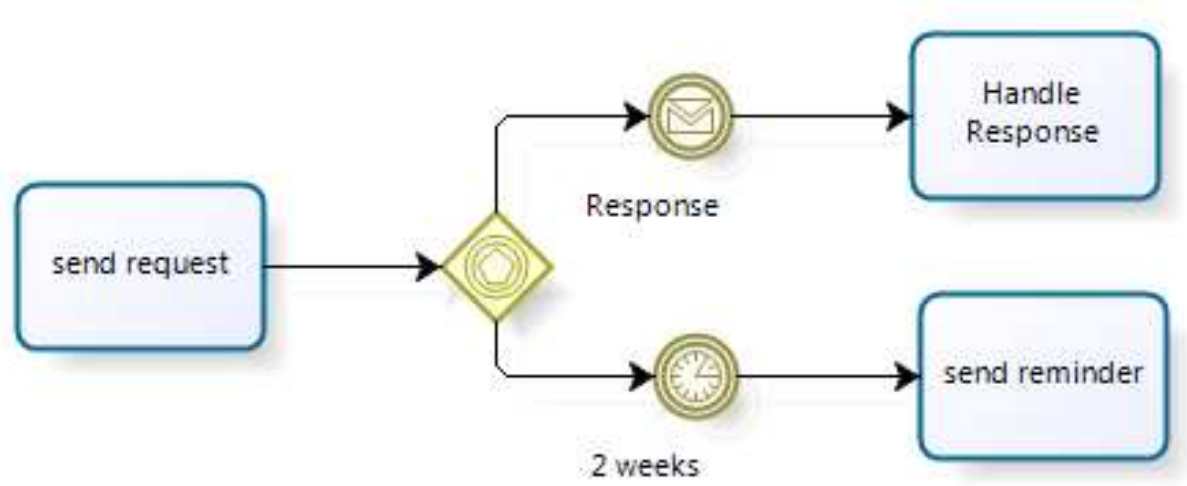
- Splitting the flow in 2 or more alternative paths
- only 1 outgoing path is chosen
- Choice is based on data known to the process (conditions)



Data based

BPMN Event-based OR gateways (split)

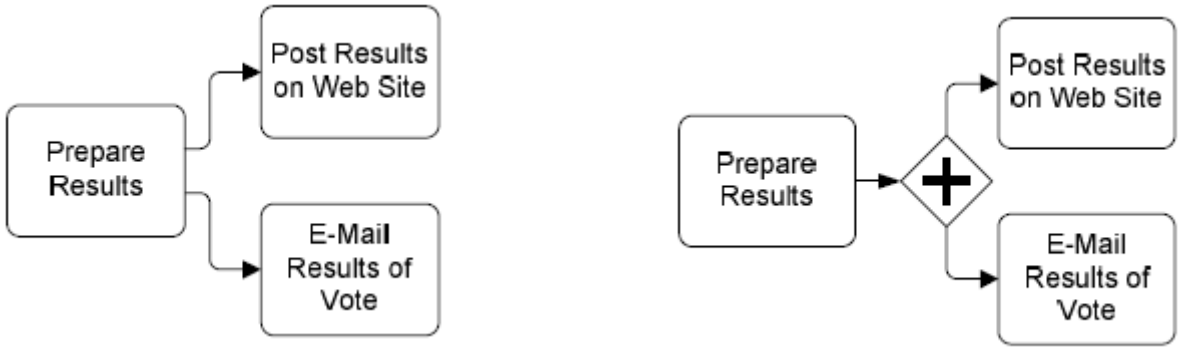
- Splitting in 2 or more alternative paths
- only 1 outgoing path is chosen
- Choice is based on event that happens next



Event-based

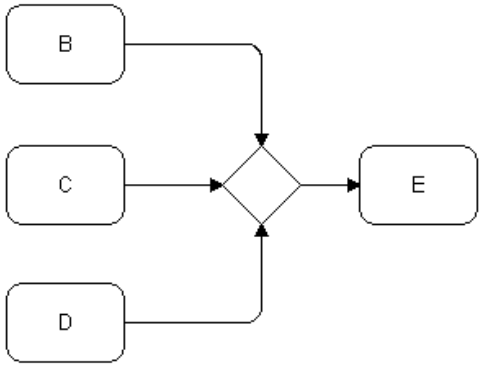
BPMN parallel gateways (split)

- multiple outgoing parallel paths

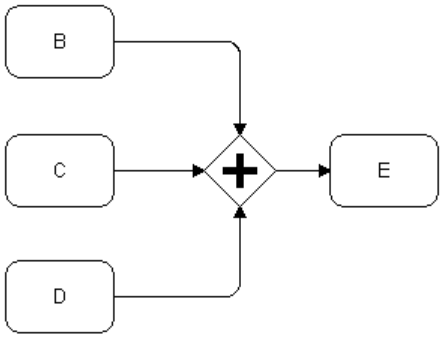


Merging paths: XOR versus AND

- XOR:
  - Merging exclusive incoming alternative paths, without gateway
  - XOR join behaves as a multi-merge: triggered once for each incoming token
  - you can also omit the gateway if you wish so

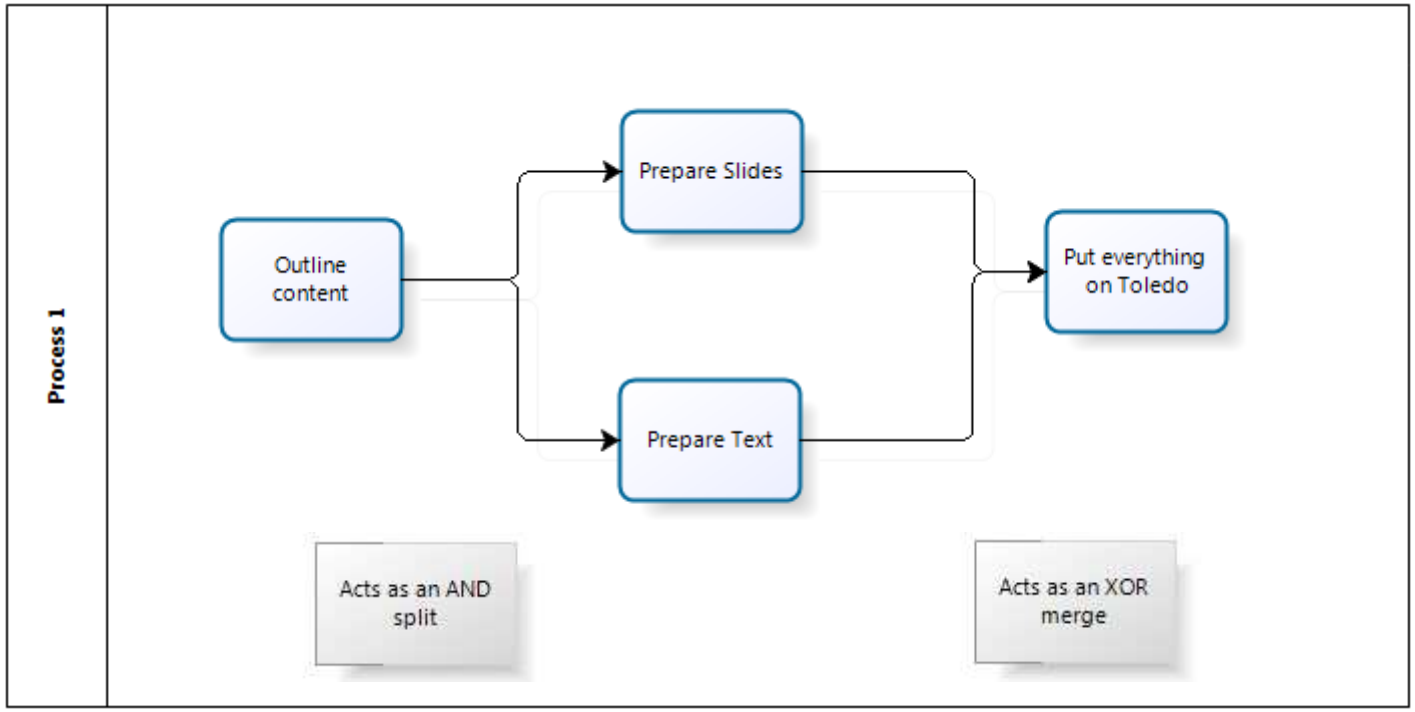


- AND:
  - Merging parallel incoming alternative paths
  - Waits until all incoming paths have finished
  - AND gateway join enforces synchronisation



Always use gateways !

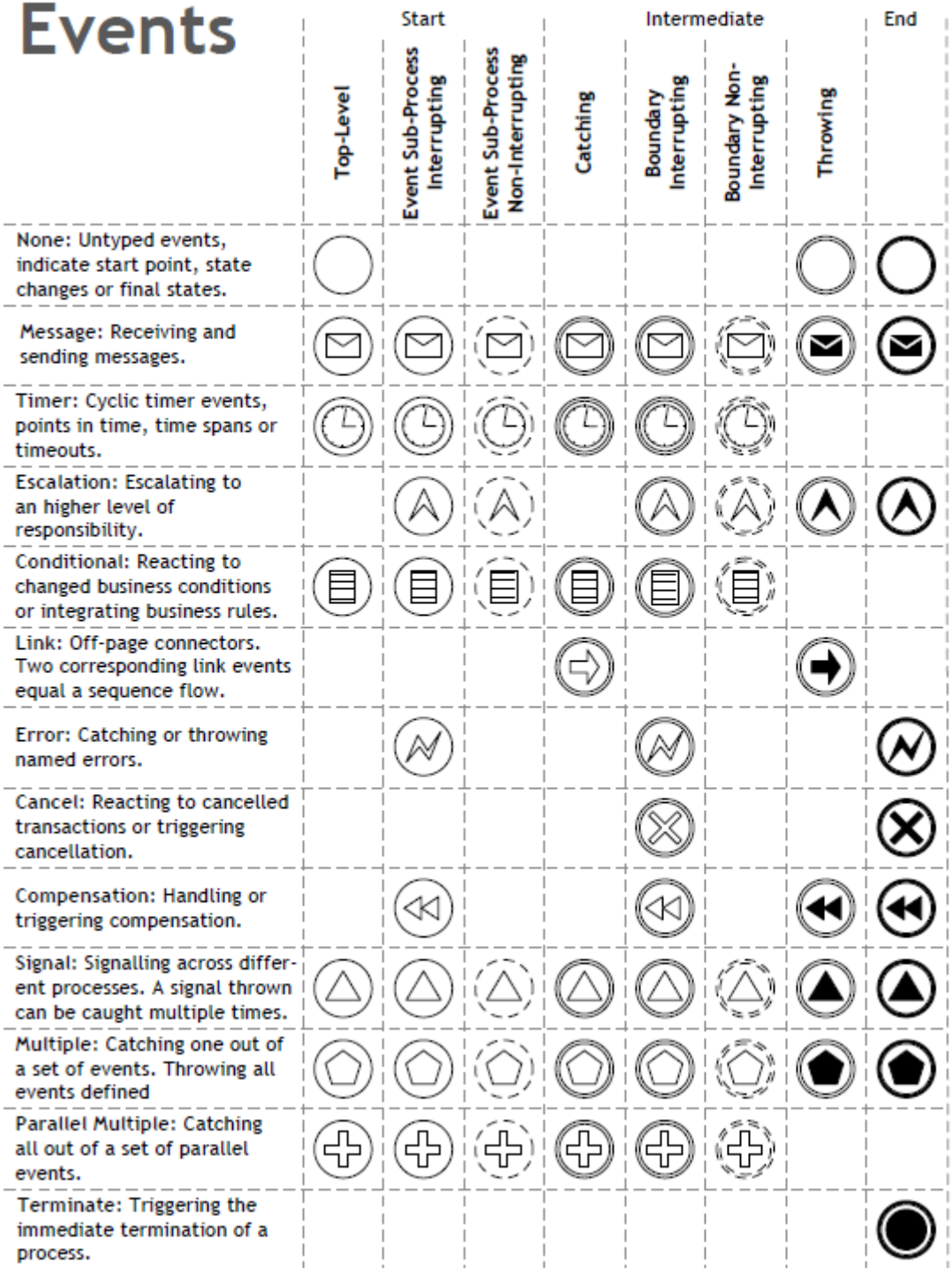
- How many times will you put docs on Toledo ?



- Using no gateways makes diagrams confusing …

BPMN Events

Events



BPMN Event

- Event – circle
  - Indication that “something has happened”
  - They affect (start, pause, end) the flow of the process
  - The process may send and receive events
    - Receive or send messages, mail, order, etc…
  - 3 types: based on when they affect the flow
    - Start event: indicates where the process starts
    - End event: indicates where the process ends
    - Intermediate event: between start and end event
  - Many, many, many types (±63)

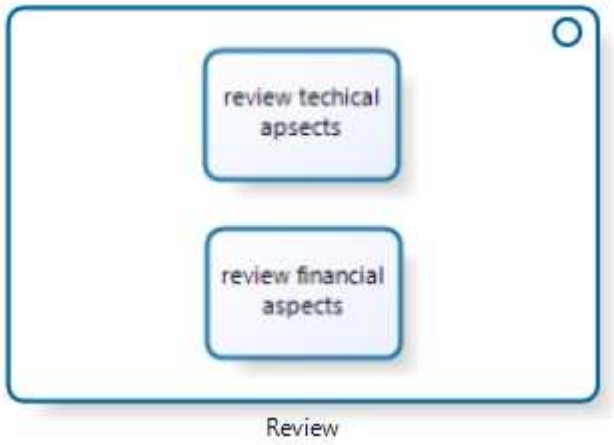
BPMN Event

- Start event:
  - Thin circle
  - Indicates where the process starts
  - Usually just one but not required

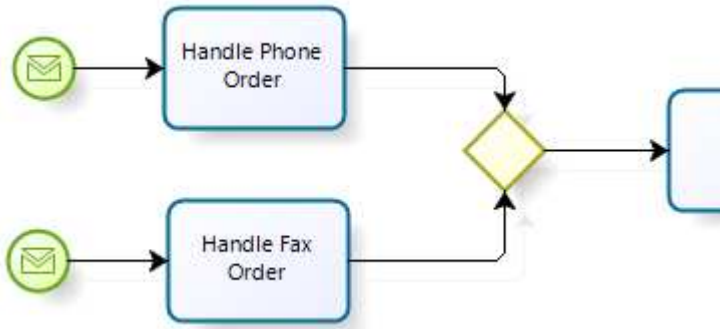
- Trigger = what causes the event
  - any = unspecified
  - message = external signal (a message) directed to this specific process
  - timer = scheduled process
  - conditional = watched data condition
  - signal =external signal, broadcasted to *any* listening process
  - multiple = any of multiple signals (any of above)

How many Start events ?

- no start event ➔ no end event
  - any activity, intermediate event or gateway with no incoming arrow is enabled and can start.



- More than one start event ➔ e.g. used for channel-dependent start



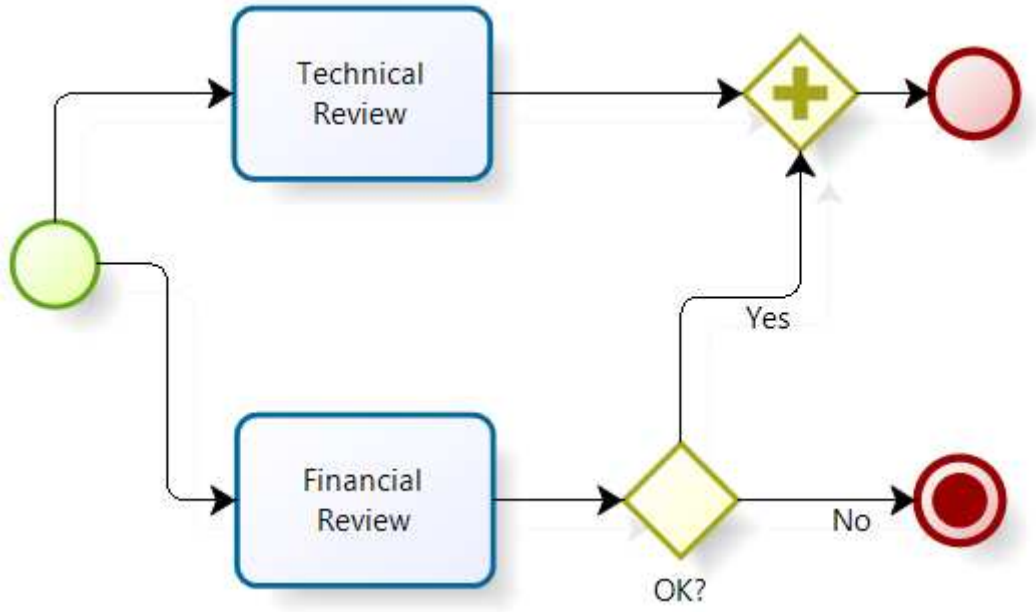
BPMN Event

- End event:
  - Thick circle
  - Indicates where the process ends

  - Multiple end events are possible ➔ Good practice to use more than one for each distinct end state
  - Each parallel path must reach an end for the (sub)process to complete normally
  - Termination, error, cancel complete process abnormally ➔ = immediately, even if other parallel paths are nog completed yet
  - May specify a result signal ➔ throws a result
    - message = sends a message to an external participant
    - signal = broadcasts a result signal, not addressed to a particular process

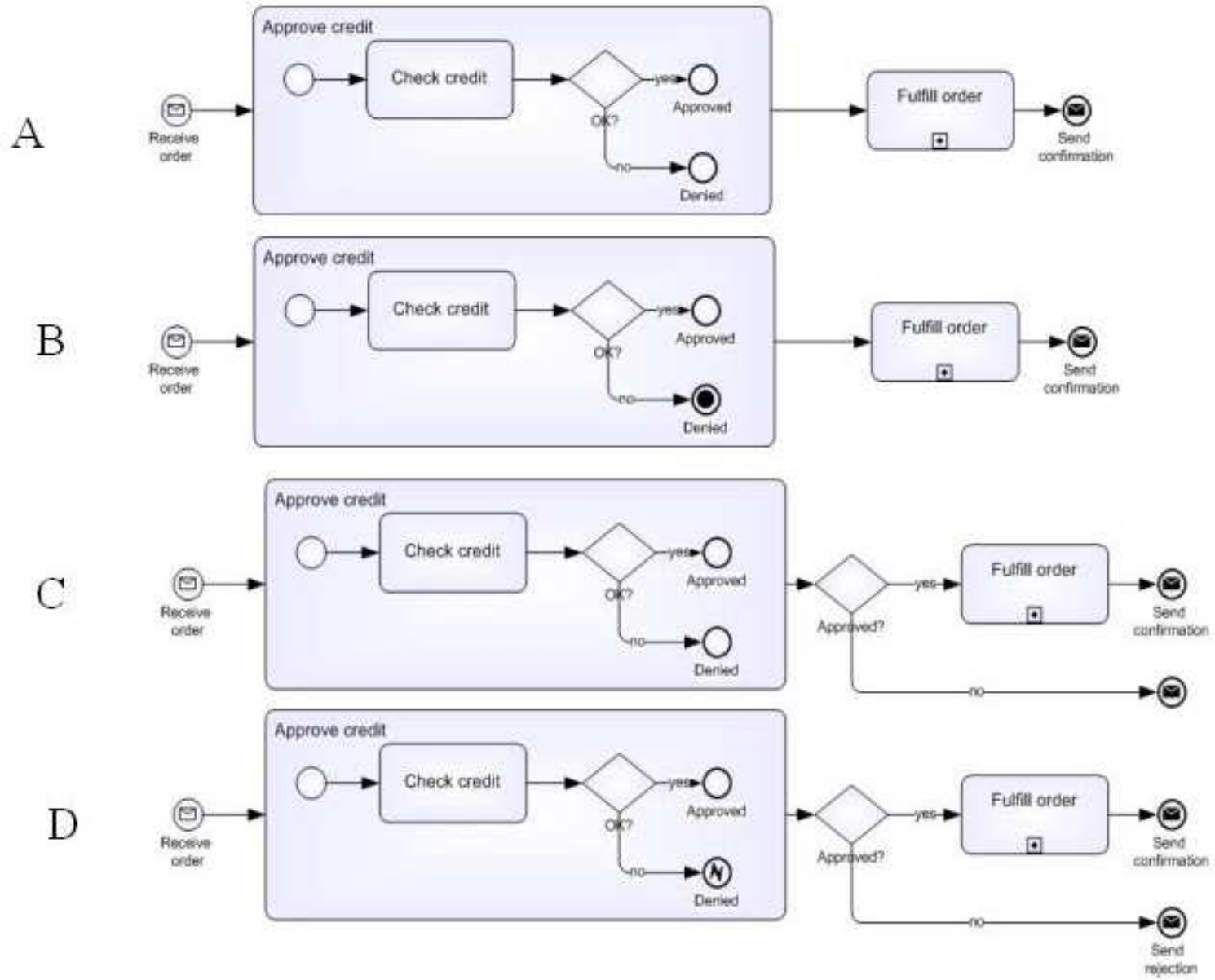
End Events

- Terminate 
  - terminates process level, even if parallel paths are still active



1. Exception handling. Which diagram is correct?

(source: Bruce Silver)

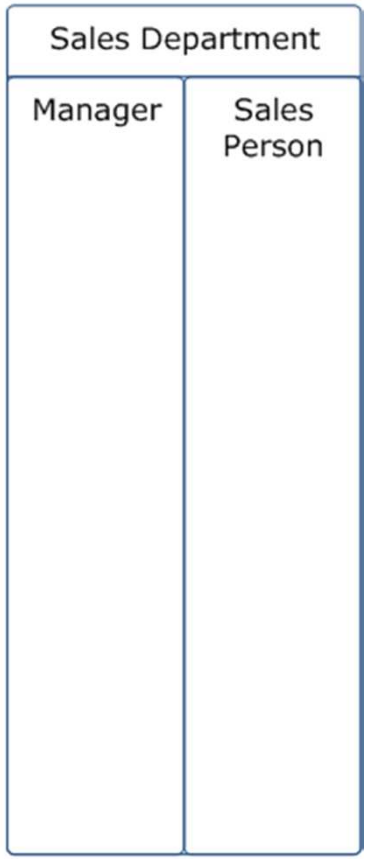


Pools

- Represents a participant in the process
  - Business Entity: company, department
  - Business Role: buyer, seller, etc.
- … or represents a business process
  - Complaint handling, or order process
- Well-defined semantics in BPMN
  - Sequence flow not allowed to cross the boundary of a pool.
  - The pool is a container for the sequence flow between activities.

Lanes

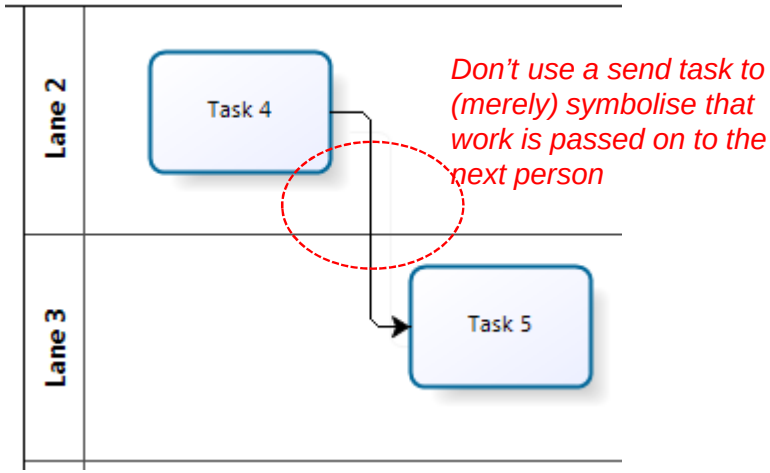
- Pools may be subdivided into lanes
  - Business Units: departments
  - Business Role: buyer, seller, etc.



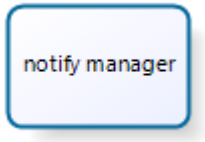
- Can be nested
  - Company – Departments – Roles
  - Laneset applies to a specific process level
  - You can replicate lane set in parent and child level diagrams

Communication inside pools

- Communication between lanes
  - Task 5 will be created as a work item in its peformers worklist



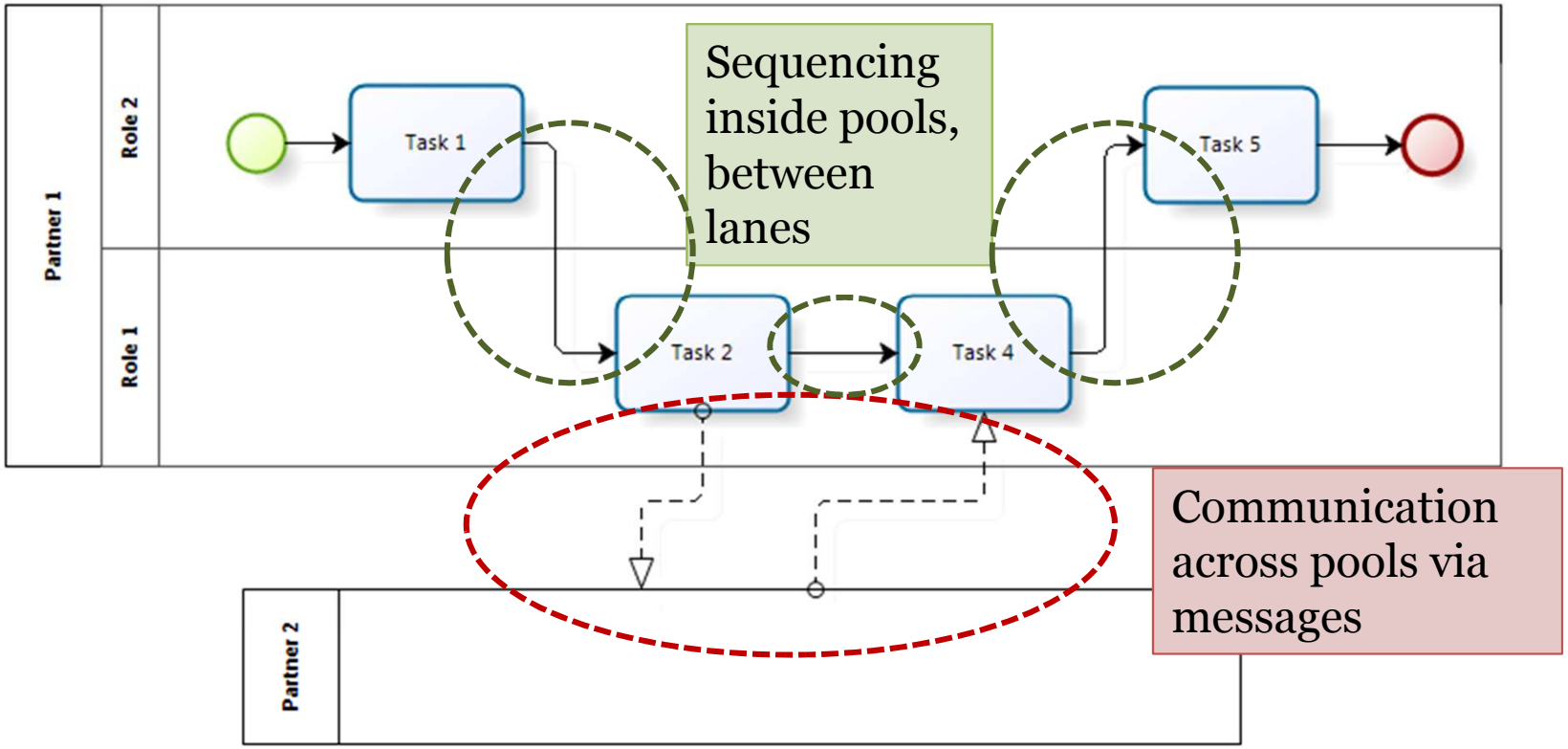
  - Use a notification task to symbolise action of notification for people without worklist or outside scope of process.



Communication across pools

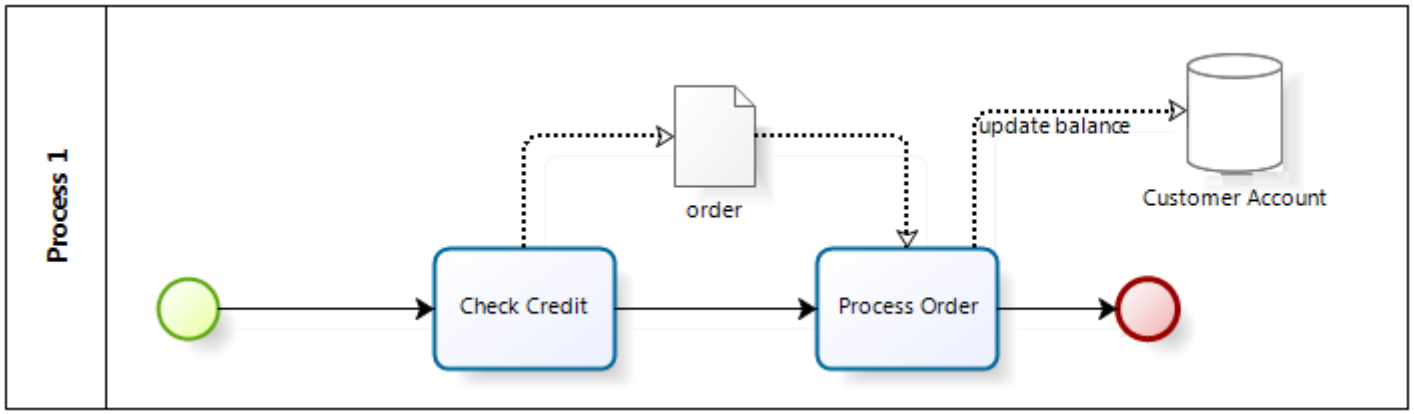
- Pools represent different partners in the process
- Usually act as black box towards each other
- Communication by means of messages

Communication

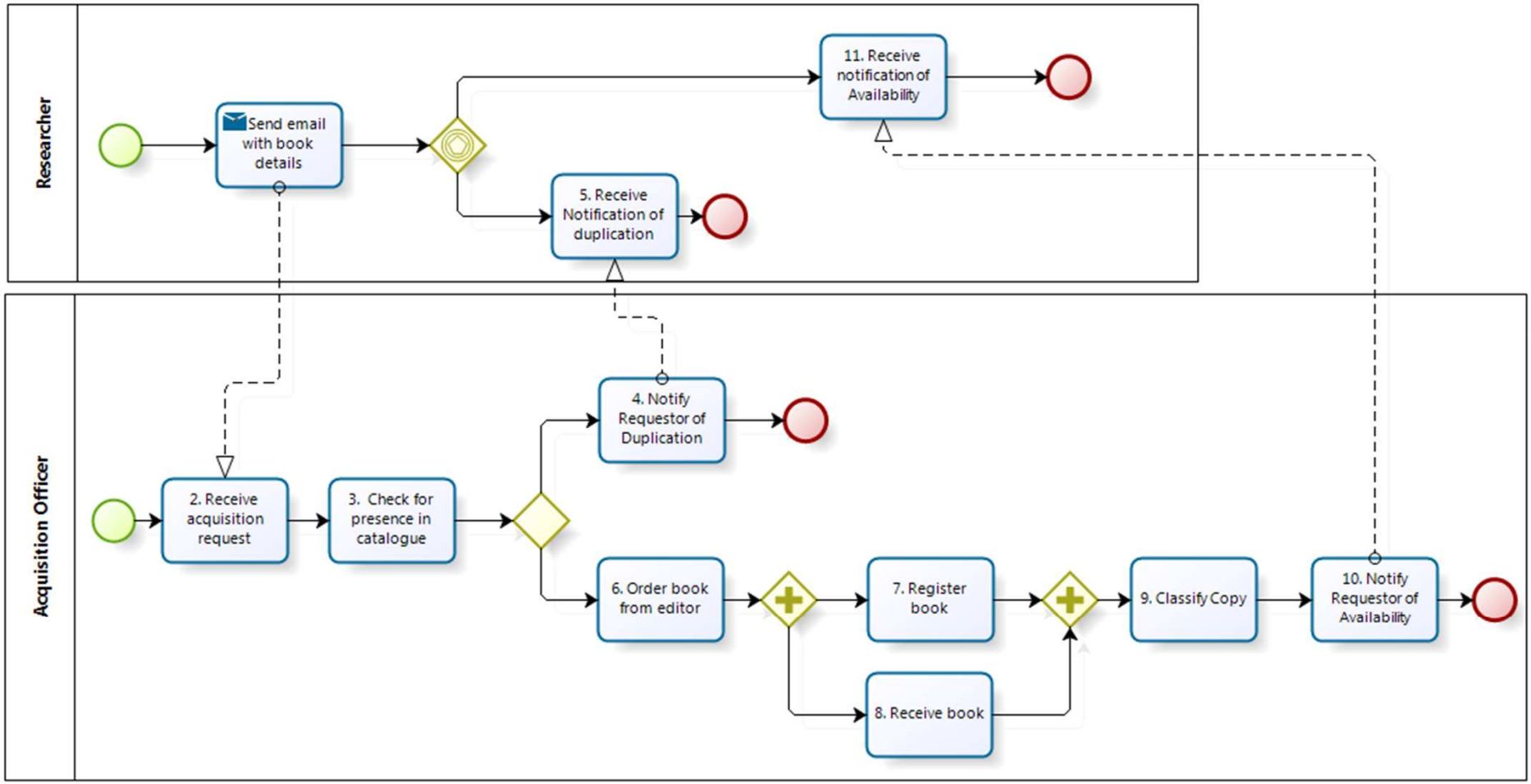


Dat Object and Data Store

- Data object
  - = local variable
- Data Store
  - = persistent data

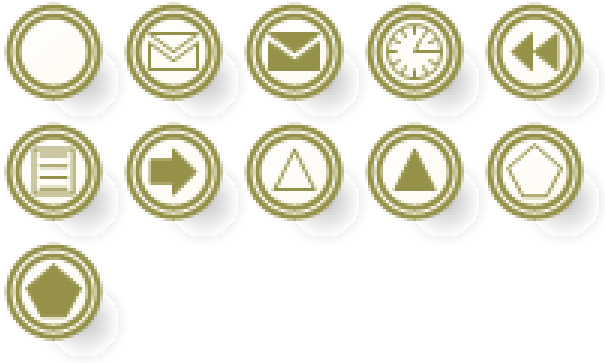


Example


Researcher
Send email with book details
11. Receive notification of Availability
5. Receive Notification of duplication
Acquisition Officer
2. Receive acquisition request
3. Check for presence in catalogue
4. Notify Requestor of Duplication
6. Order book from editor
7. Register book
8. Receive book
9. Classify Copy
10. Notify Requestor of Availability

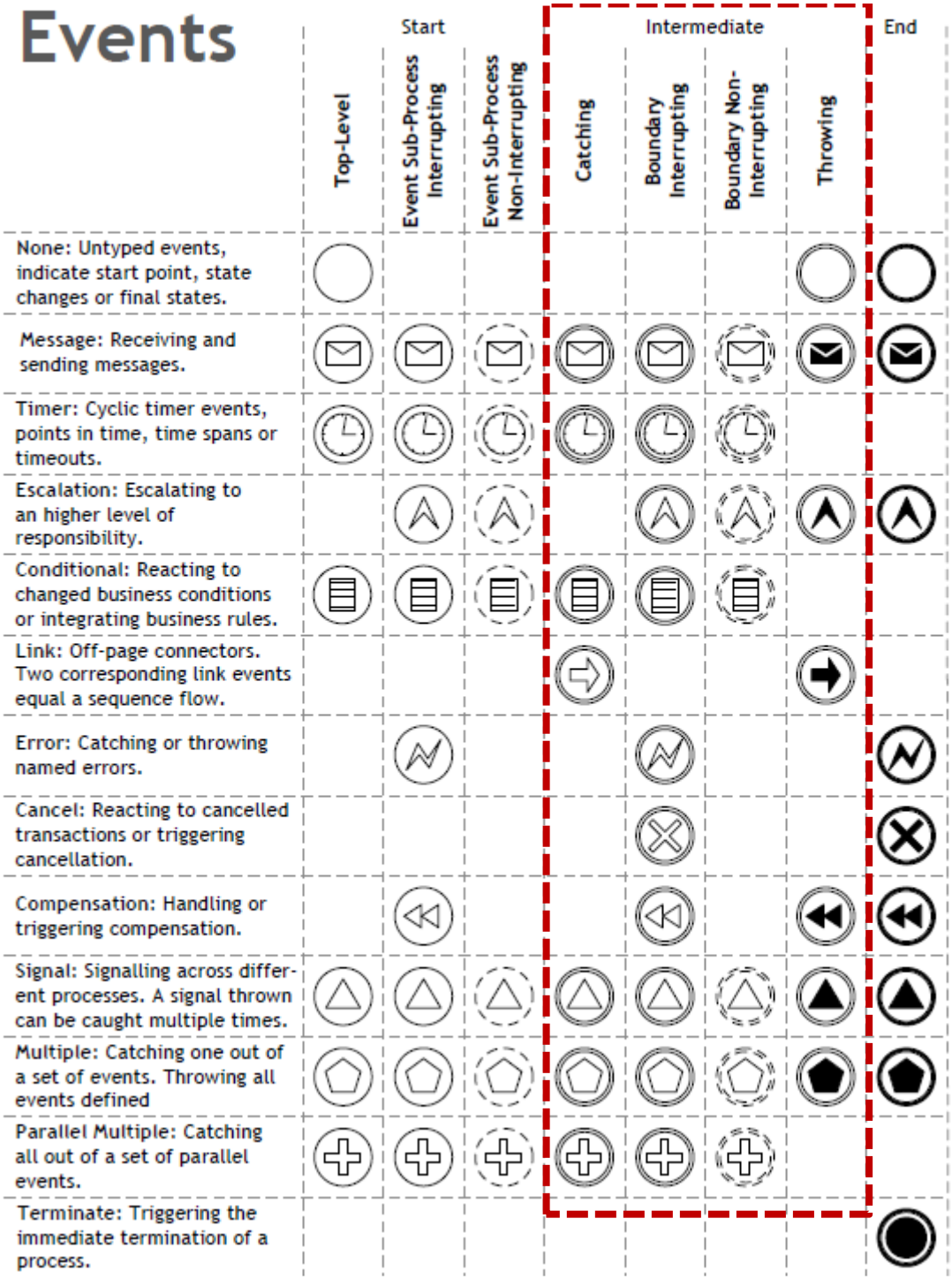
BPMN Intermediate event

- Important and powerful aspect of BPMN 2.0
- Semantics depend on the placement, the border style and the content of the event in the diagram
  - Placement
    - in sequence flow
    - Boundary = Attached to an activity
  - Boundary Event Border style
    - full = interrupting
    - dashed = non-interrupting
  - Filled or empty symbol
    - filled = throwing
    - empty = catching
      - all boundary events are of catching type



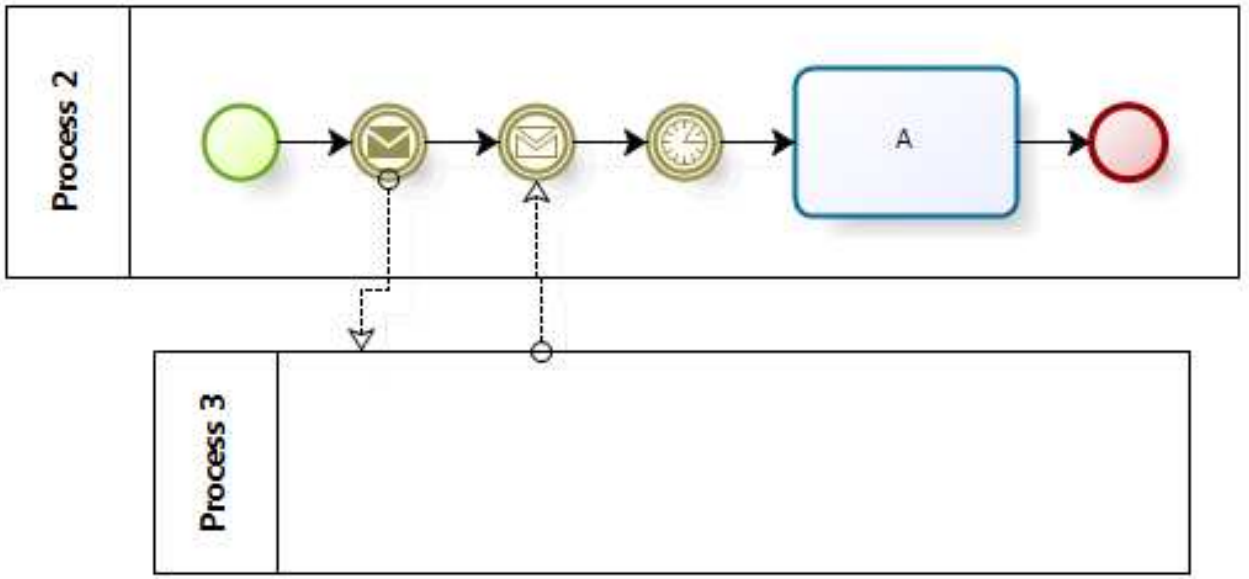
BPMN Events

Events



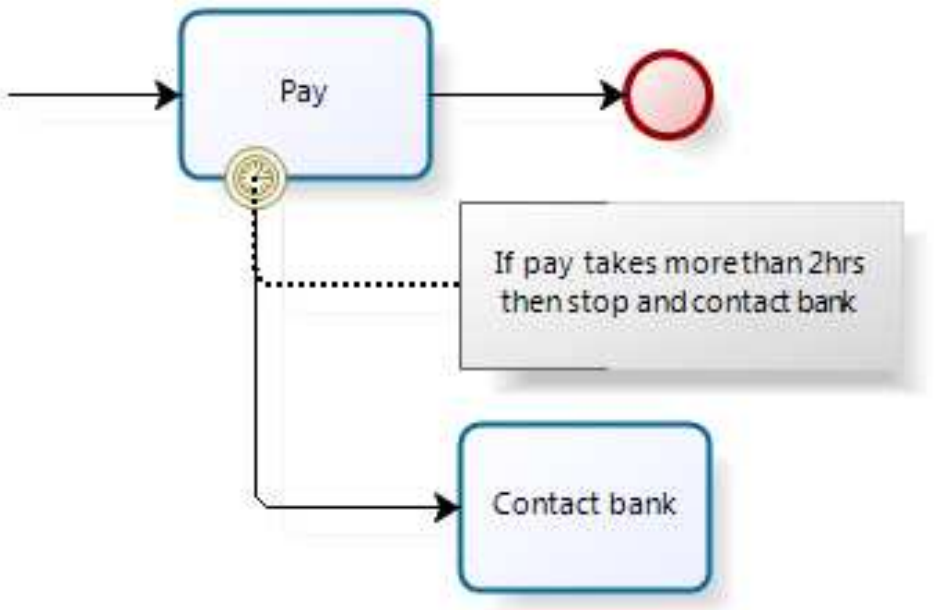
BPMN Intermediate event

- Used in a regular sequence flow
  - Throwing = process *generates* the trigger signal
    - Not possible for e.g. Timer (you can catch a time, not throw it) or error (error terminates the process ➔ cannot be intermediate)
  - Catching = process *waits* for the trigger signal and then proceeds
    - Not possible for e.g. error (you can only catch an error on the boundary of a task/subprocess, not "out of the blue")



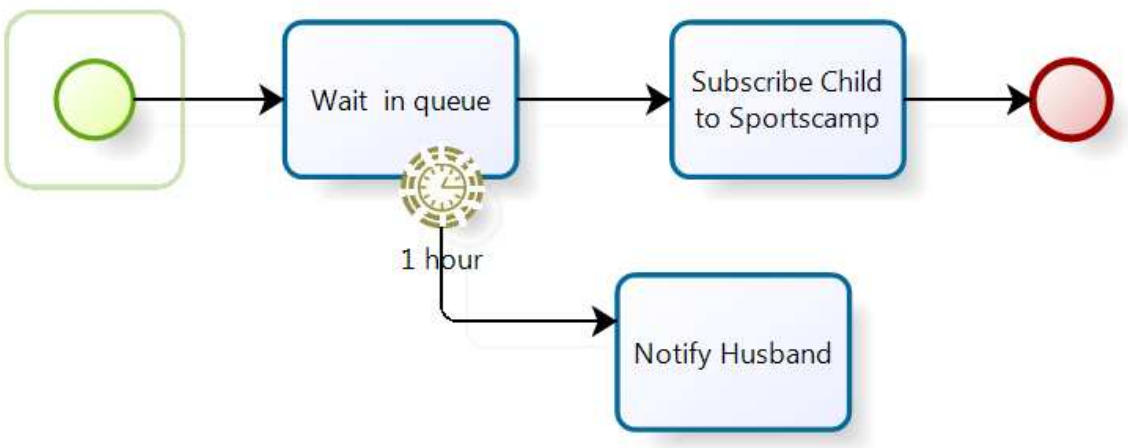
BPMN Intermediate event (interrupting)

- Attached to an activity (task and sub-process)
  - On occurrence of the event (if full line, interrupt the activity and) follow the exception flow
  - If event doesn't occur then follow normal flow
  - The activity defines the scope of the event
    - Timer starts when activity starts and event is ignored when activity finishes



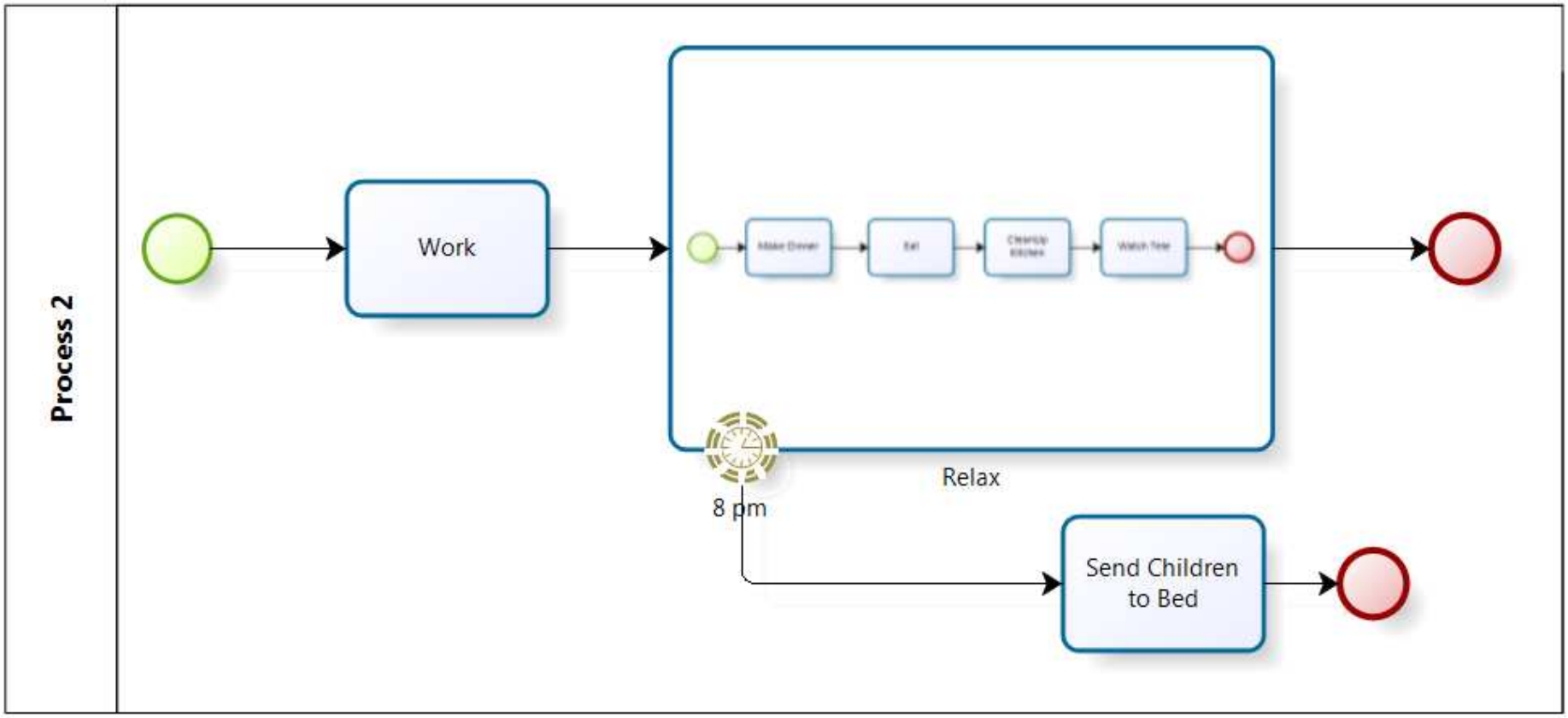
BPMN Intermediate event (non-interrupting)

- Attached to an activity (task and sub-process)
  - On occurrence of the event (if full line, interrupt the activity and) follow the exception flow
  - In any case follow normal flow
  - The activity defines the scope of the event
    - Timer starts when activity starts and event is ignored when activity finishes



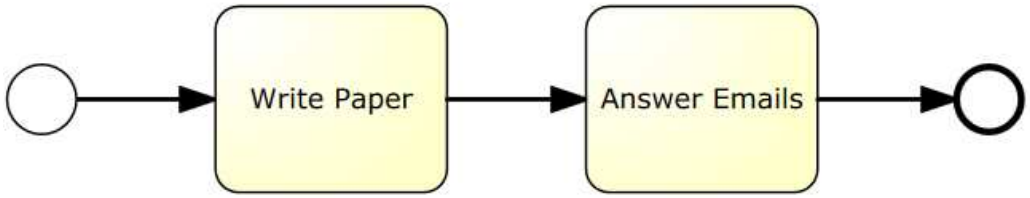
Timed Interval

- Send my children to bed at 20h, whatever I'm doing at that moment
  - Use a subprocess

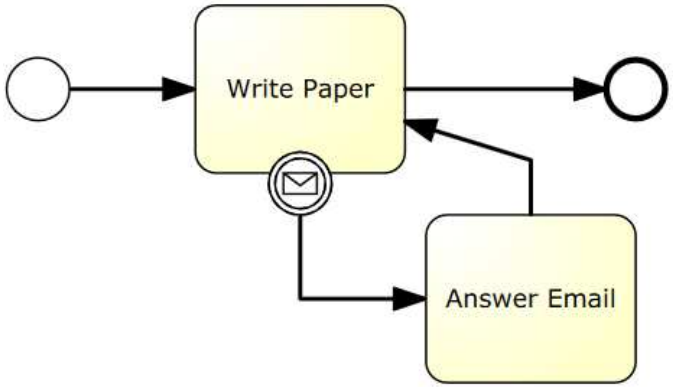


Example

- I write a paper, and when my paper is finished I answer my emails



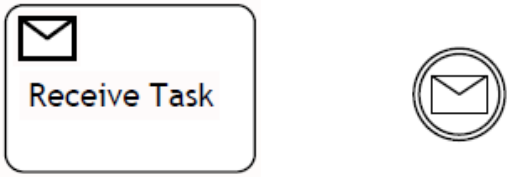
- I write a paper but in the mean time answer all my emails



Communication & Events

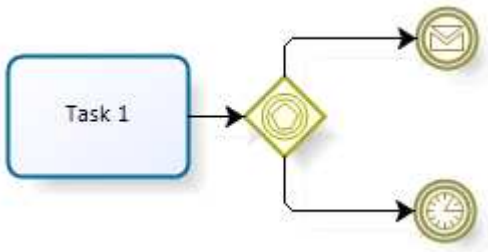
- Sending a message

- Waiting for a message



- =/ "catching" a message (boundary event):

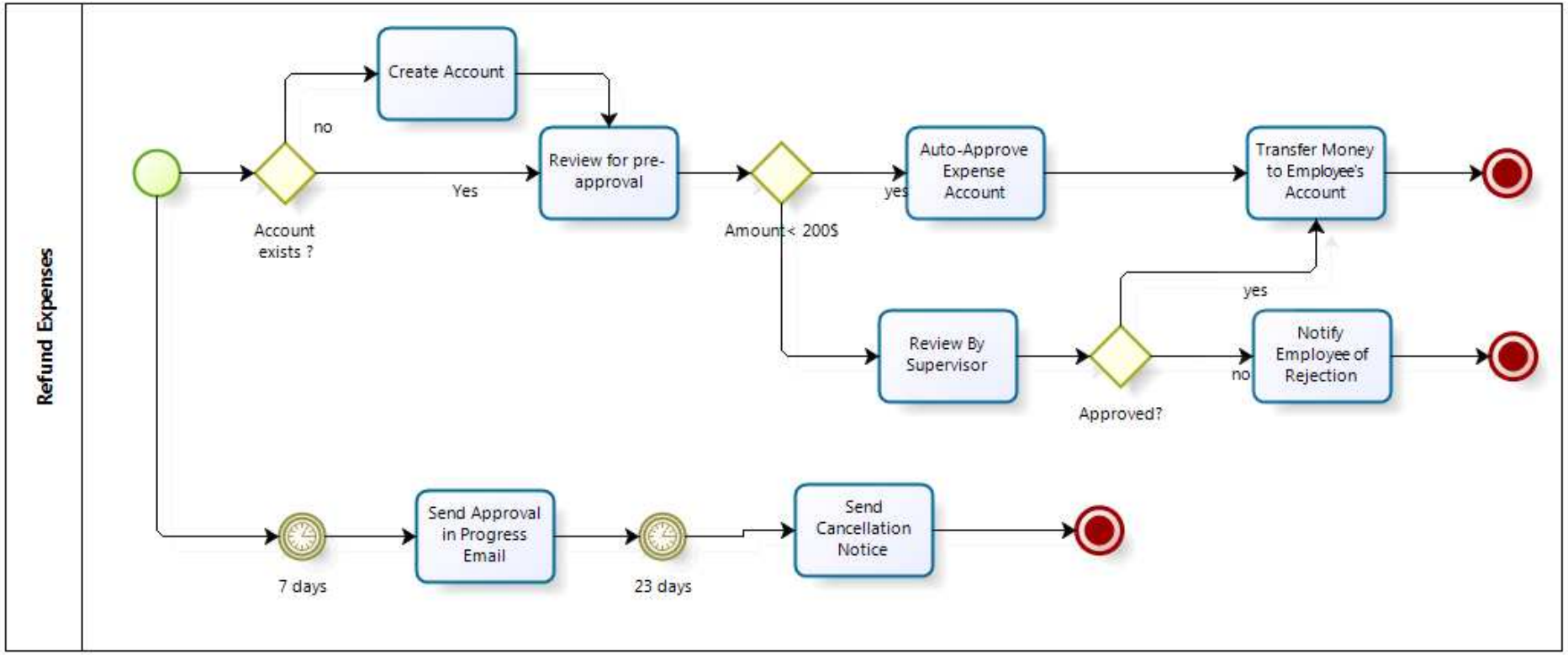
- Use event-based gateway to model time-out:

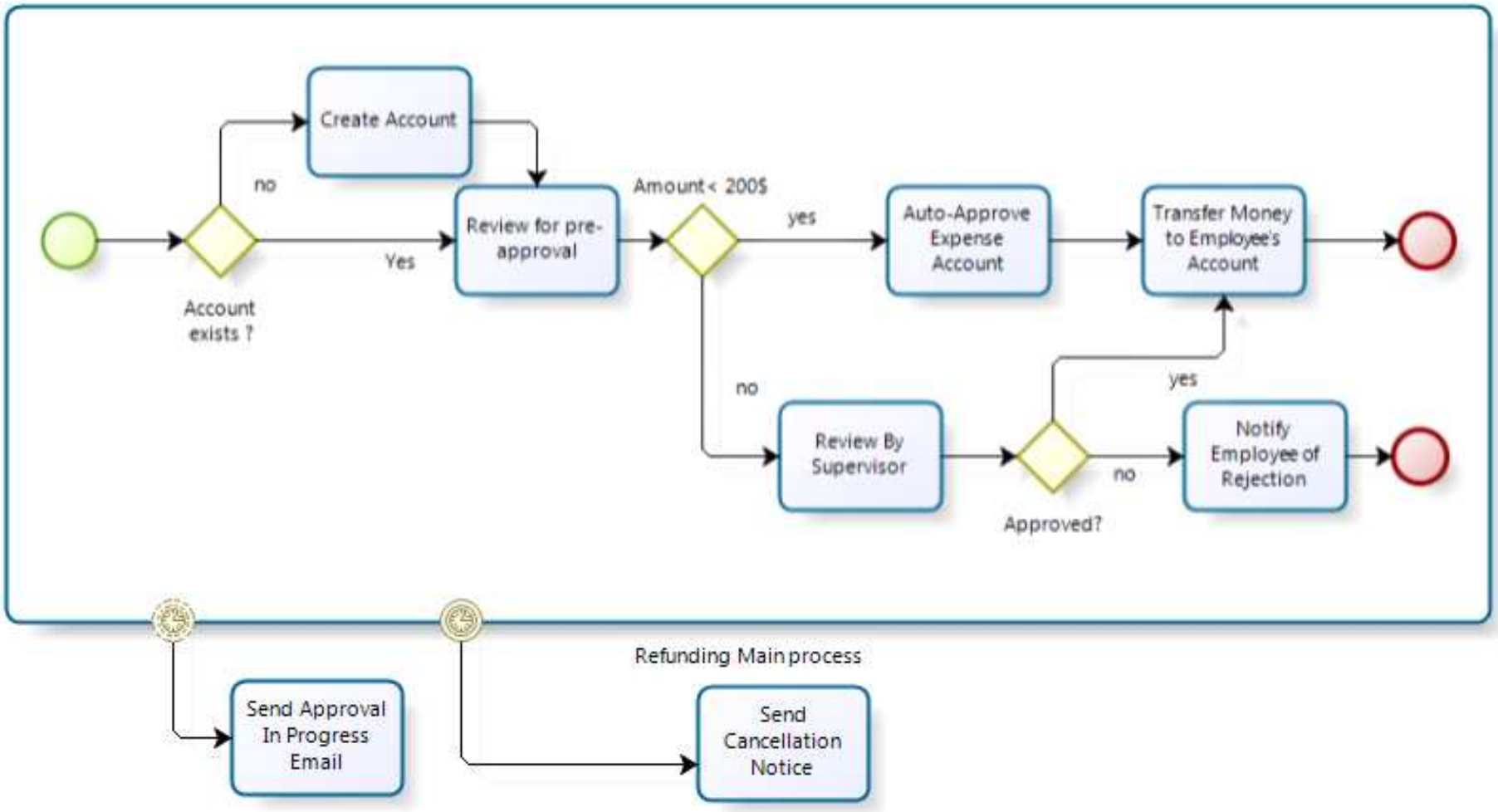


Example : Reimbursement of expenses

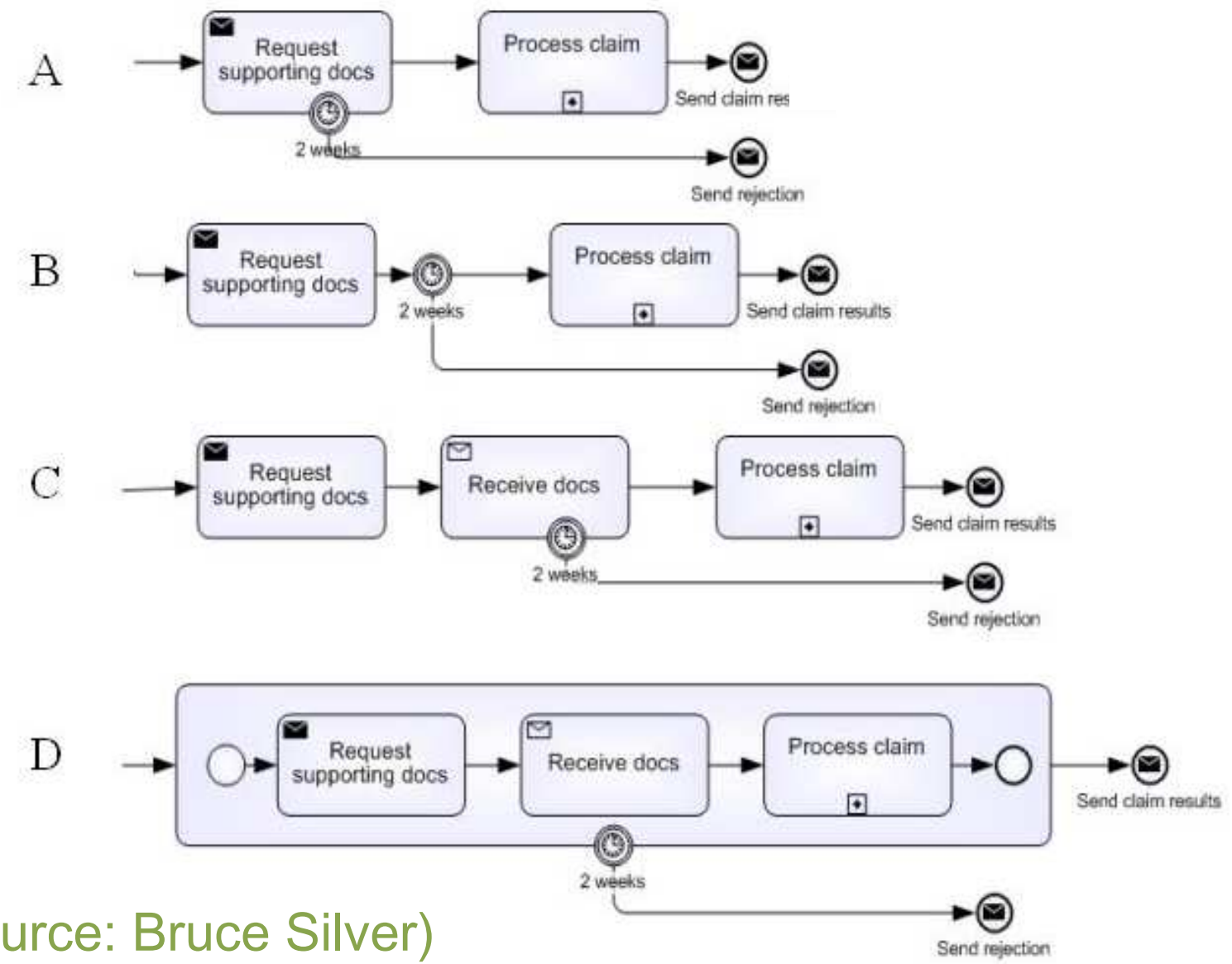
- Draw process for reimbursement of expenses incurred by employees (e.g. buying office supplies)
- Process information:
  - After the Expense Report is received, a new account must be created if the employee does not already have one
  - The report is then reviewed for automatic approval
    - Amounts 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 final 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

Process Model



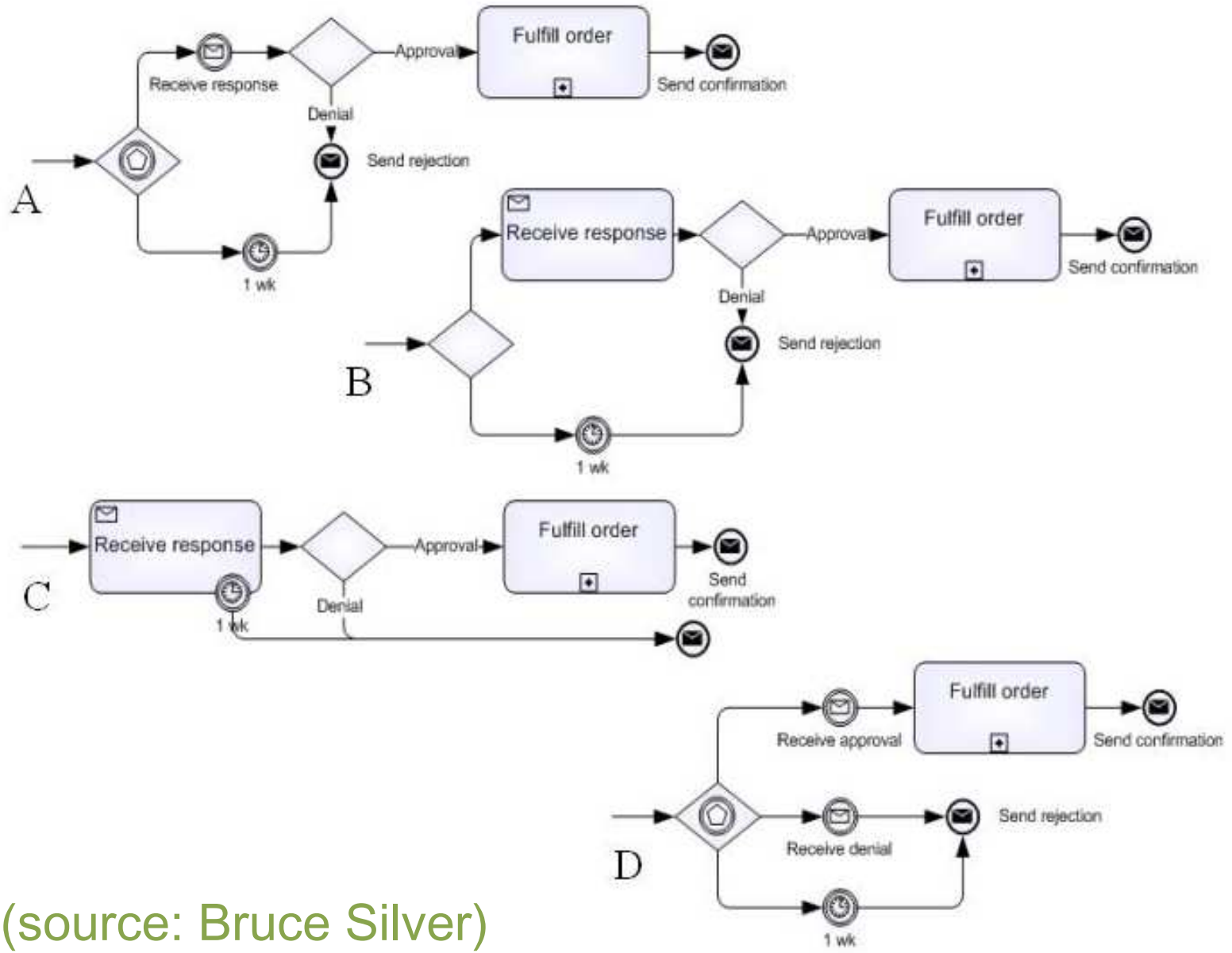

Create Account
no
Amount < 200$
Review for pre-approval
yes
Auto-Approve Expense Account
Transfer Money to Employee's Account
Yes
Account exists ?
no
Review By Supervisor
yes
Notify Employee of Rejection
no
Approved?
Refunding Main process
Send Approval In Progress Email
Send Cancellation Notice

3. Wait for requested document or timeout. Which diagram is correct? (10 points)



(source: Bruce Silver)

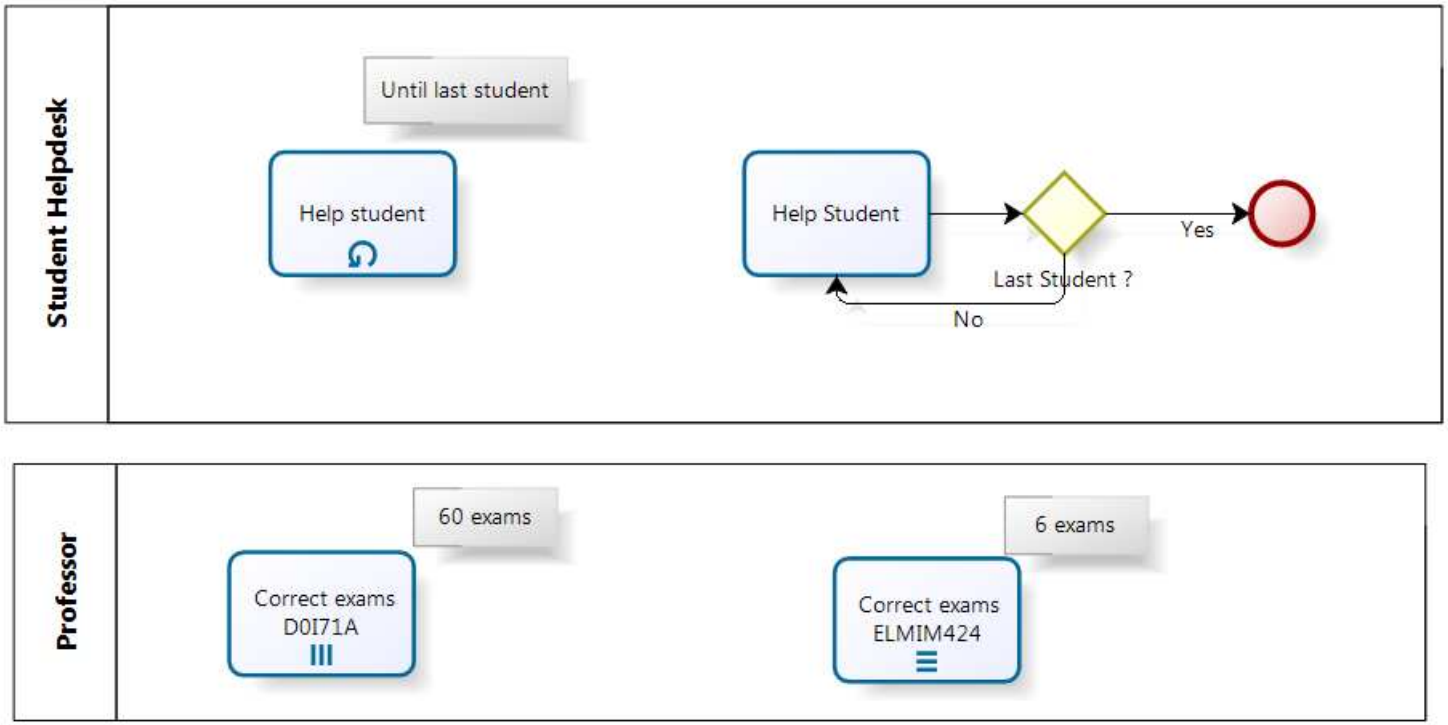
5. Alternative responses with timeout. Three diagrams are correct; which one is incorrect?



(source: Bruce Silver)

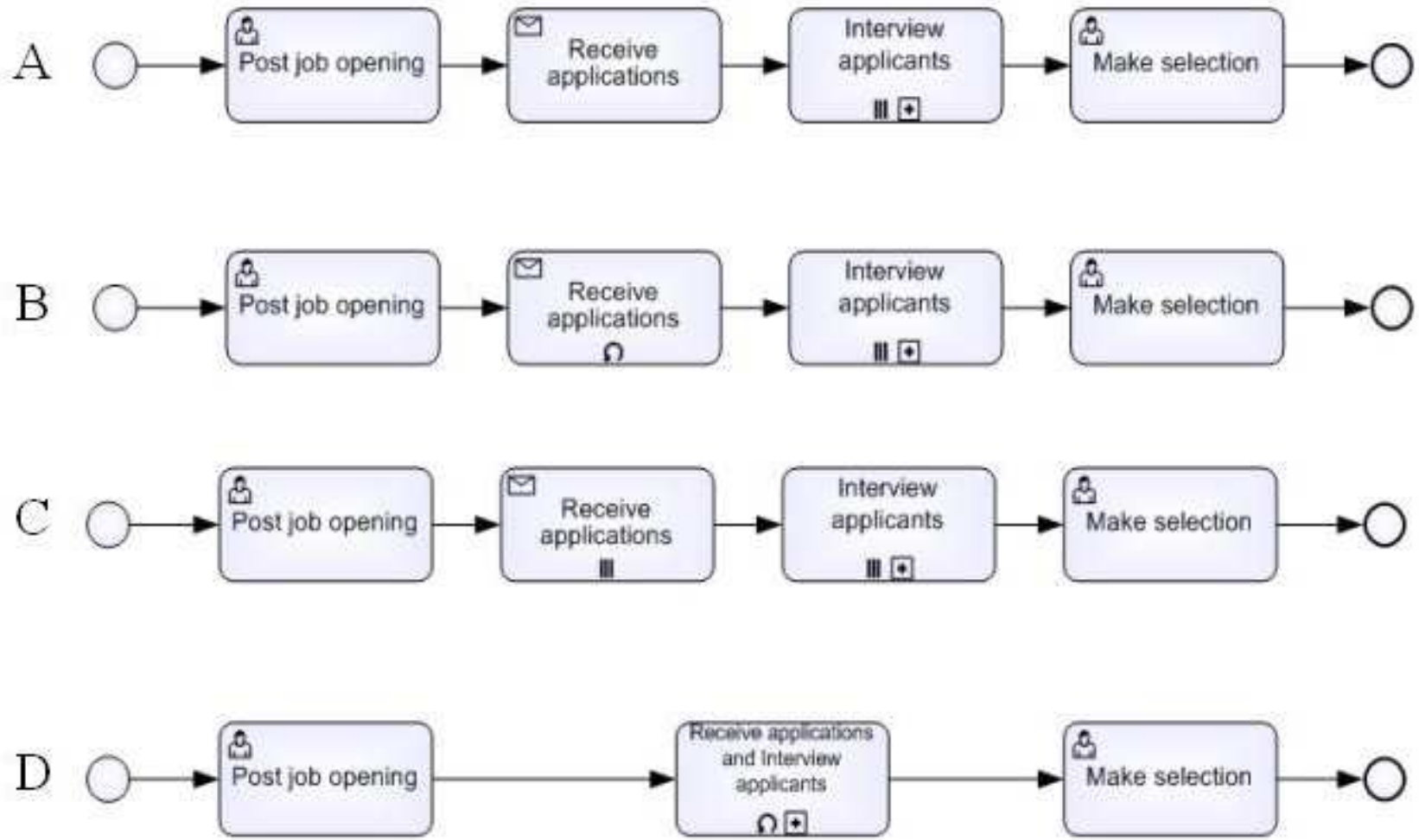
Iteration

- Loop activity = Do ... While ...
  - same as an activity followed by a loop-back gateway
- Multi-instance = For ...
  - in parallel (|||) or sequential (≡)



2. Loop and MI activities. Which diagram is best? (10 points)

(source: Bruce Silver)



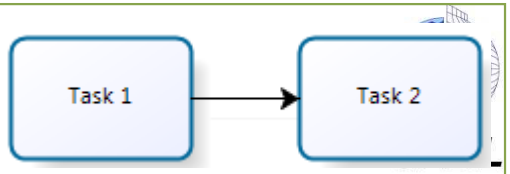
SME/SPIRE

BPMN: Extra considerations Advanced concepts

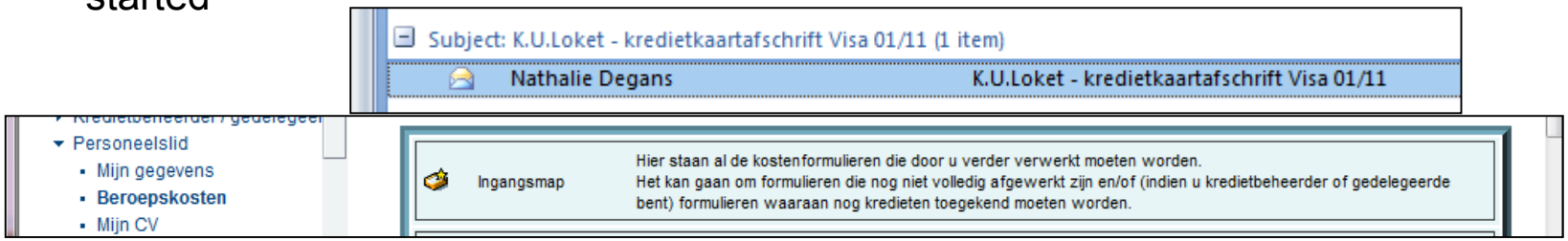
Monique Snoeck
Management Information Systems Group
FEB KU Leuven
monique.snoeck@kuleuven.be

SADL

Activities: Starting

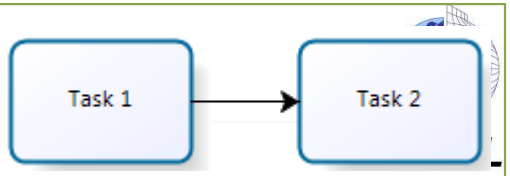


- When the first activity completes, the second one starts immediately
  - SOA: A service is immediately available upon request & accessible to each authorized requester
  - BPMN Service Task: the service provider is assumed to begin performing the business function immediately upon invocation
  - BPMN User task:
    - A Task becomes available on a work list, and/or is e.g. notified to the user by mail
    - Even though performer has not begun to work, technically, the activity has been started

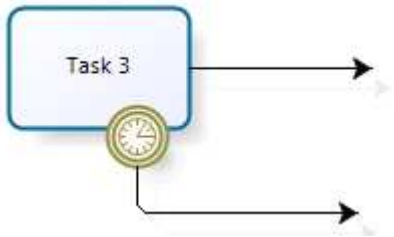


    - Example: At 11:55 I tell my daughter to empty the dishwasher. She finishes reading her book chapter, starts emptying the dishwasher at 12:10 and finishes by 12:15. How long did the task "empty dishwasher" take ?

Activities: Completing



- Normal Completion
  - the thing that triggers the flow out of an activity.
  - A subprocess is completed when all its parallel paths have reached an end
- Abnormal completion
  - Interruption of an activity or a subprocess by various exceptional conditions
  - Internal state of an activity is invisible to the model reader
  - ➔ use Events attached to boundary of task to model abnormal completion
  - Exception flow (out of boundary event) happens immediately, normal flow does not get executed.

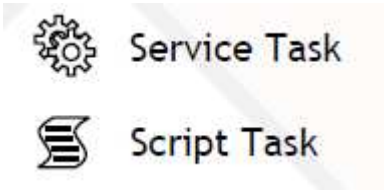


- NOTE !
  - Events & Gateways have no performers. These are control logic elements.
  - These are assumed to execute immediately & instantaneously (no time lapse)

Decisions & Rules

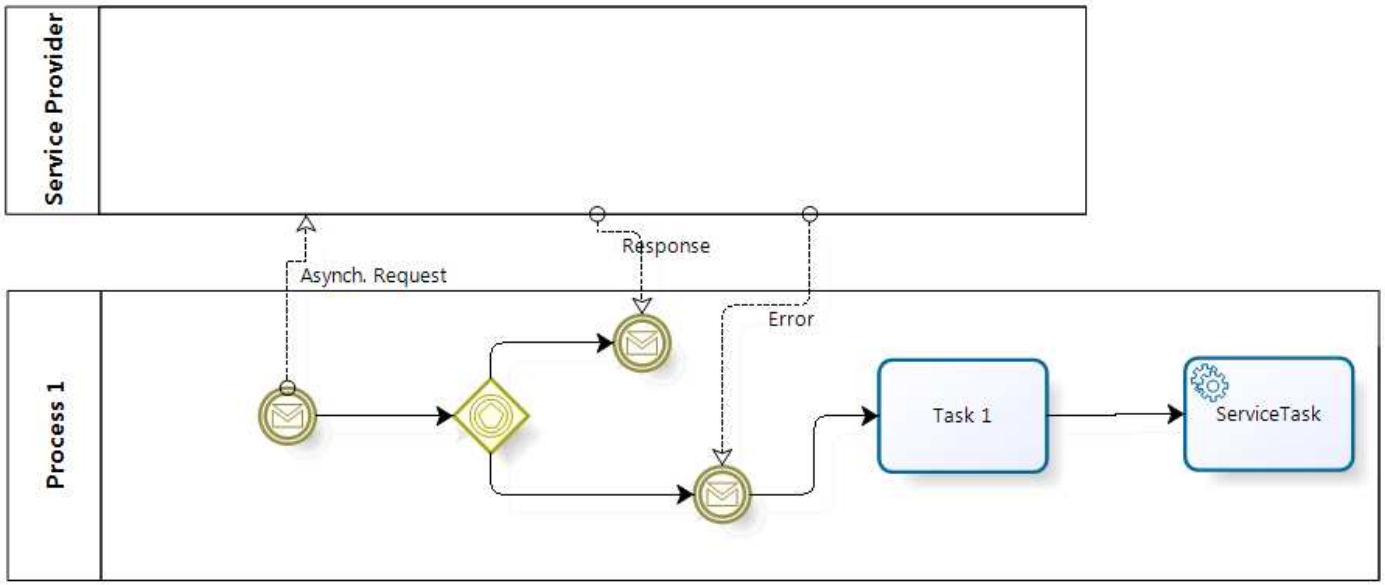
- Processes & Rules are complementary.
- No clear guidelines on how to combine the two paradigms together however.
- Little support for rules in BPMN
- Business rules should be defined independently of the process in which they are used
- BPMN
  - gateways = "routing rules", process specific
  - Business Rule Task
    - = decision service at a particular point in the process
    - Gateway after the task can be used to route further process depending on outcome
  - Conditional Event: signals when a particular condition has become true (e.g. queue too long)

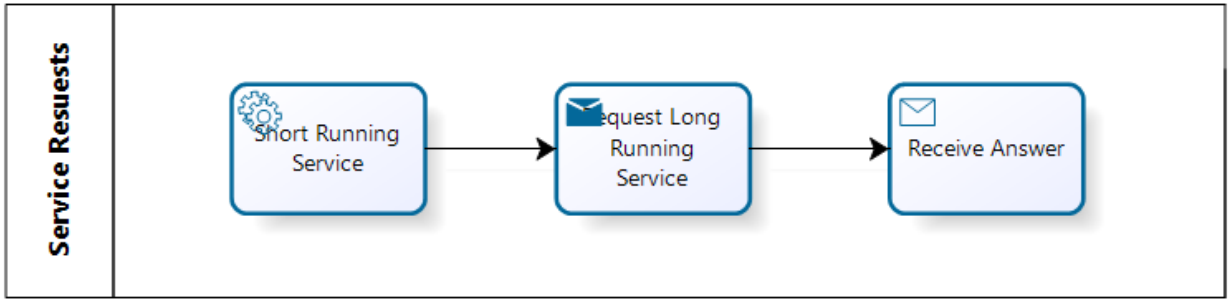
Automated Tasks



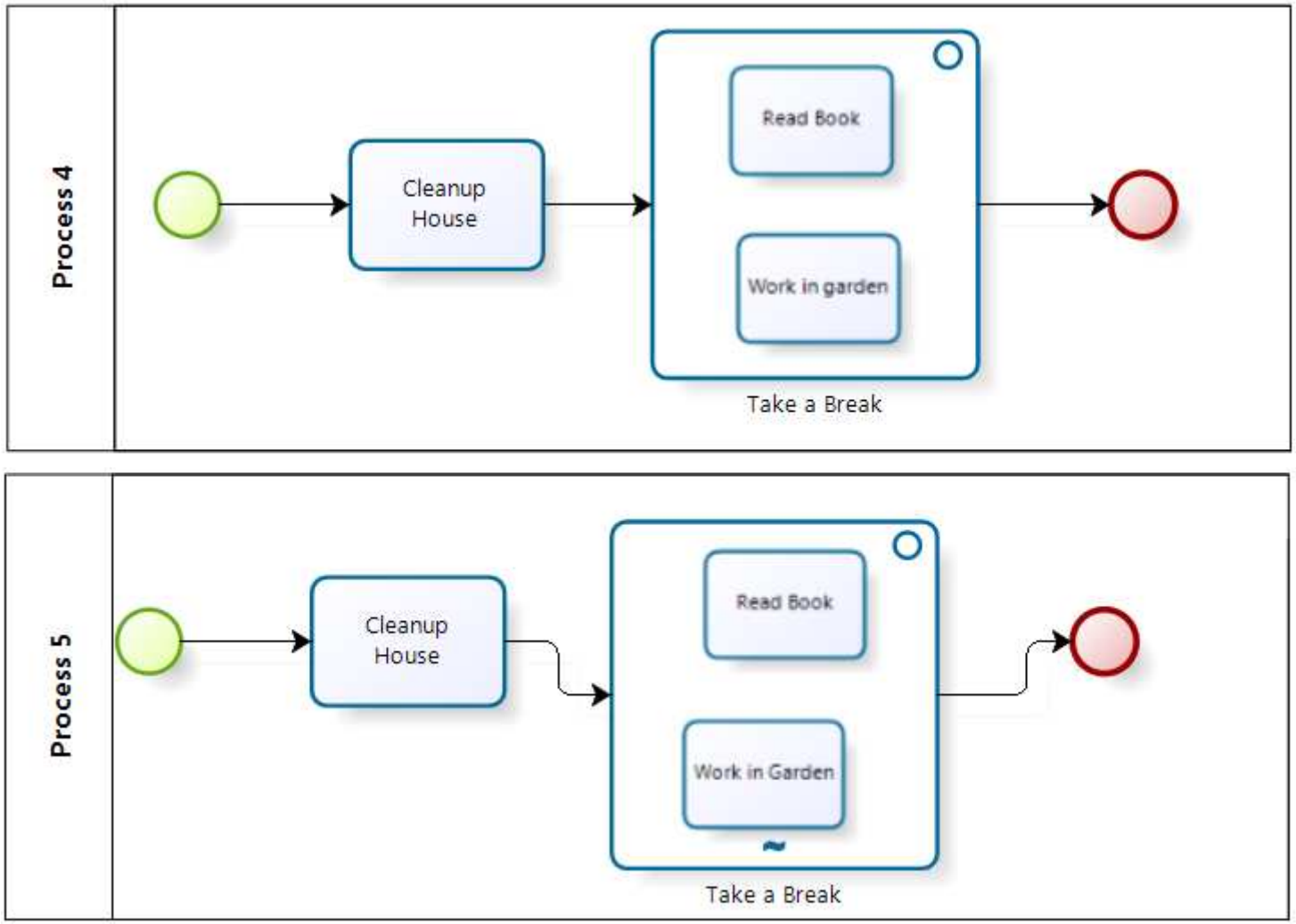
- Script Task = script executed on the process engine itself
- Service Task = Automated task executed by something other than process engine
  - Recommendation to use it for synchronous call to a service, i.e. process will wait until service has completed and sends a reply
  - ➔ not to use for long-running processes
- Asynchronous Service Request
  - model service as Black Box Pool

Synchronous versus Asynchronous Service Request



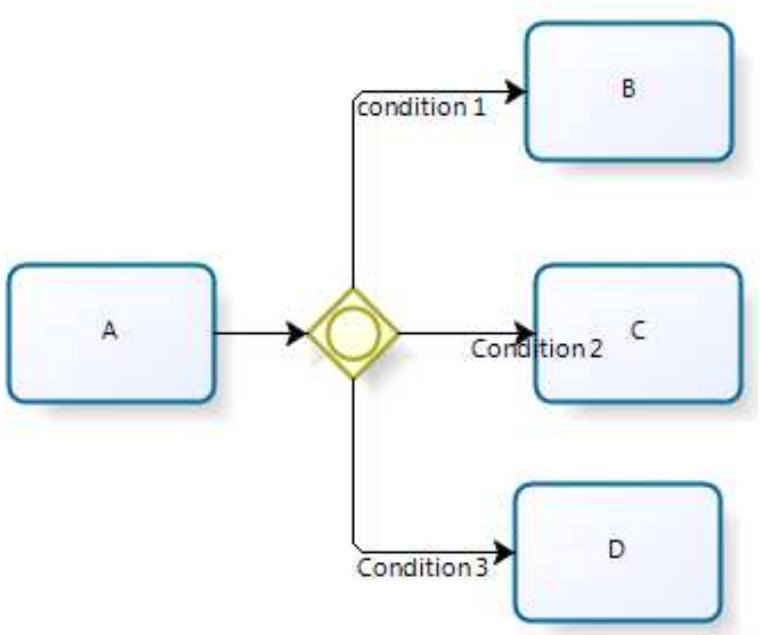


BPMN Subprocess (legacy)



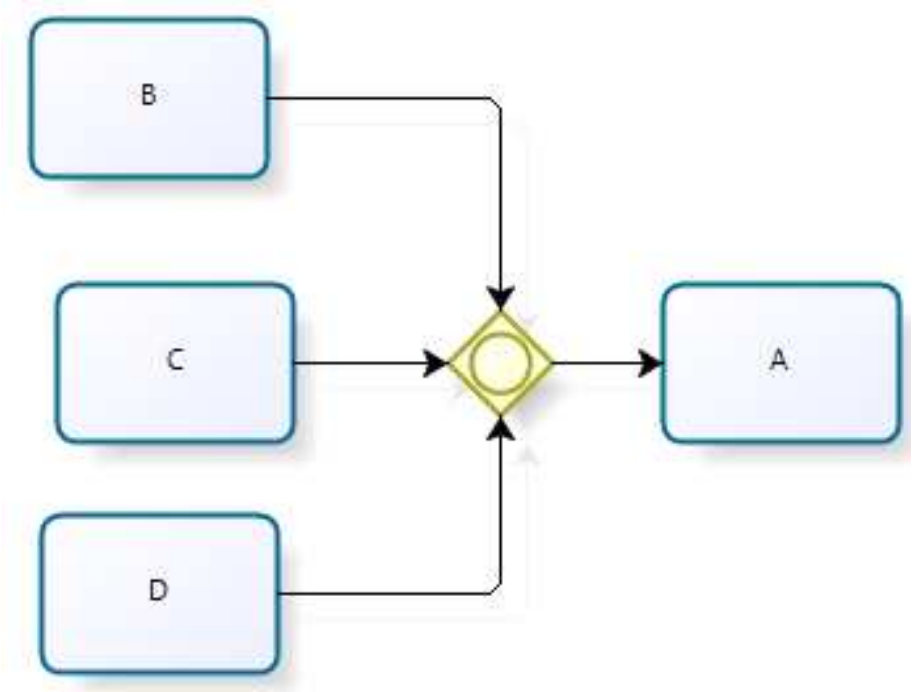
Inclusive OR (Split)

- OR gateway with "O" inside signifies an *inclusive OR*
  - Multiple outgoing paths can be chosen
  - requires a condition on each of its gates



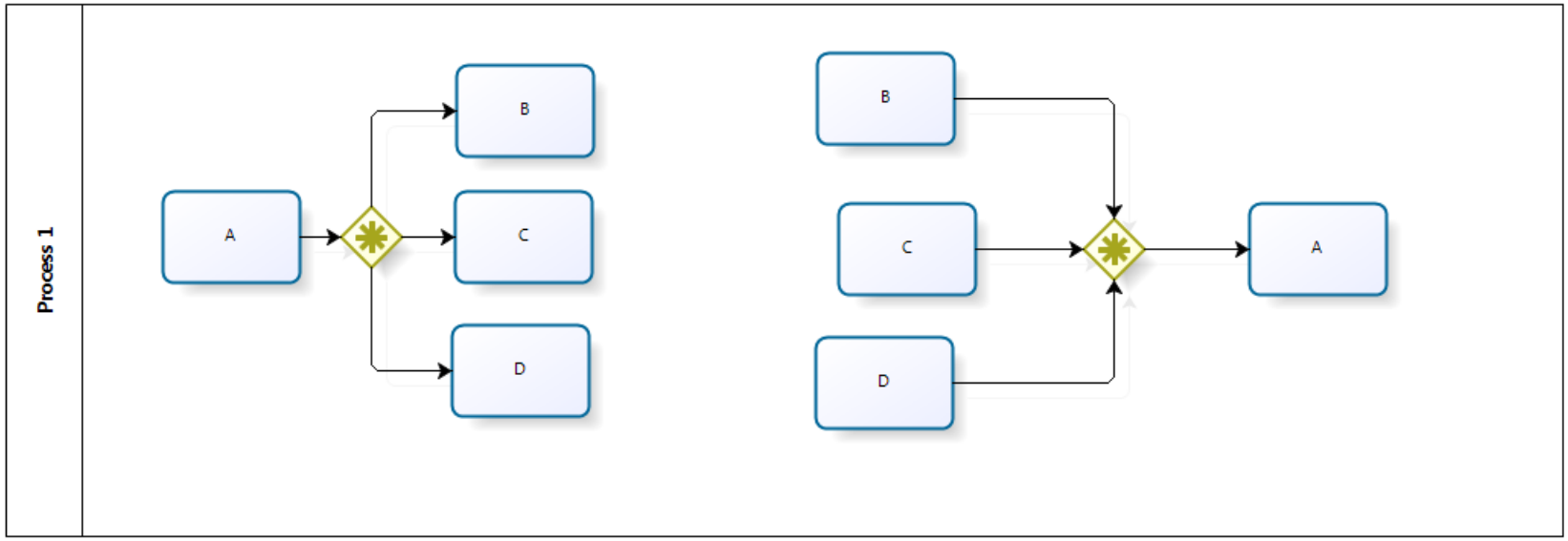
Inclusive OR (merging paths)

- Conditional multi-merge:
  - Waits for some incoming paths, not all
  - Behaves like an AND, but ignores some incoming flows

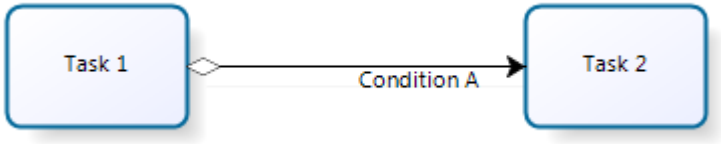


Split & Join Complex gateway

- Requires a textual annotation to define its behavior

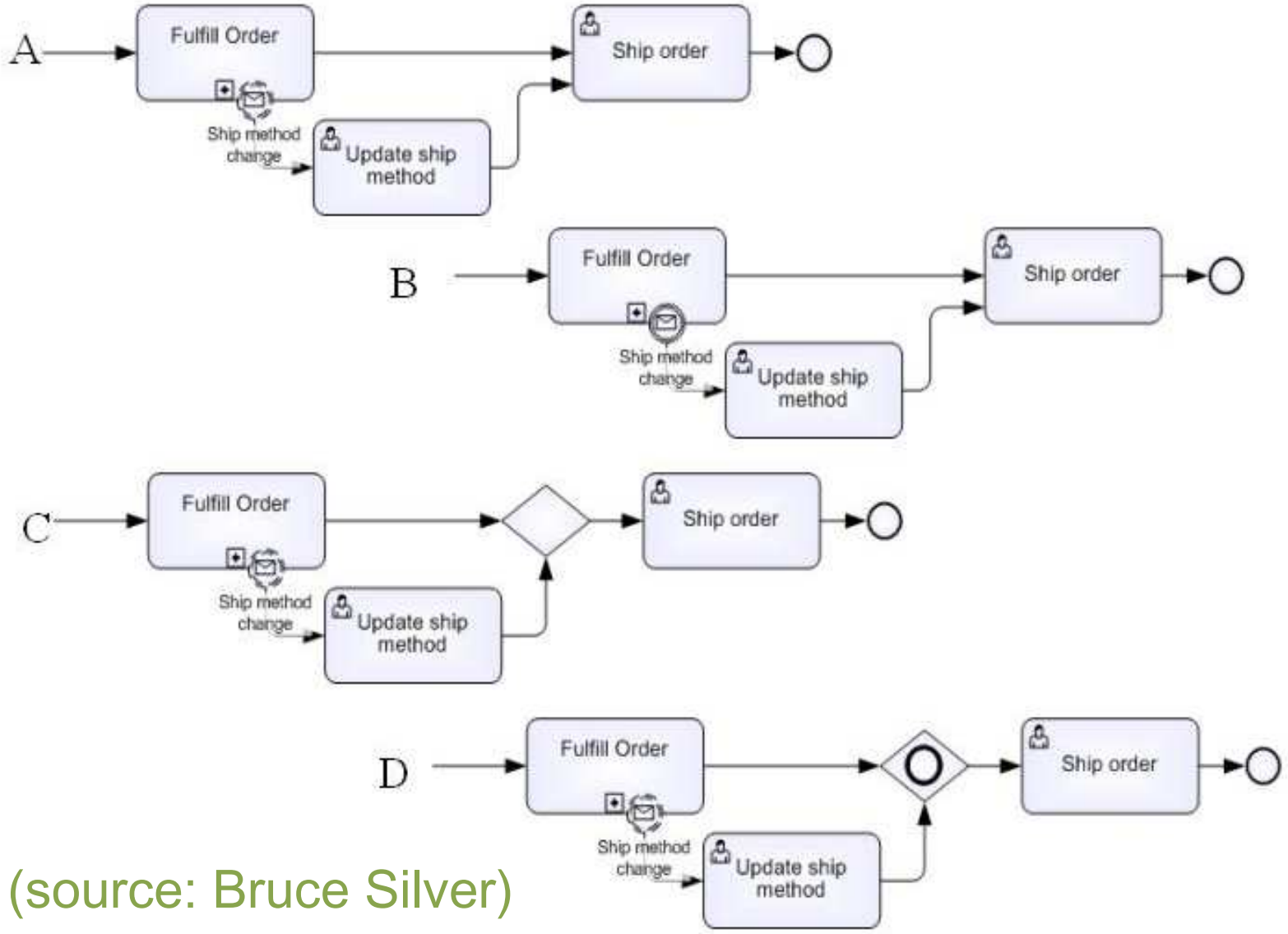


Conditional Sequence flow



- Can be used to simulate Inclusive OR gateway
- Can be used to simulate Exclusive OR gateway ...
  - ... bad practice ...

9. Order change. During fulfillment, the ship method is changed. Which diagram is correct?



(source: Bruce Silver)

Throwing, Catching and (Non-)interrupting events

Events

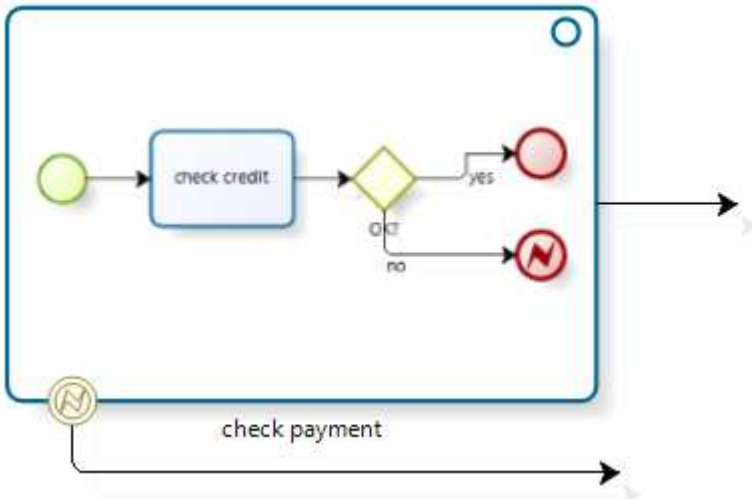
	Start			Intermediate				End
	Top-Level	Event Sub-Process Interrupting	Event Sub-Process Non-Interrupting	Catching	Boundary Interrupting	Boundary Non-Interrupting	Throwing	
Message: Receiving and sending messages.								
Escalation: Escalating to an higher level of responsibility.								
Error: Catching or throwing named errors.								
Cancel: Reacting to cancelled transactions or triggering cancellation.								
Signal: Signalling across different processes. A signal thrown can be caught multiple times.								

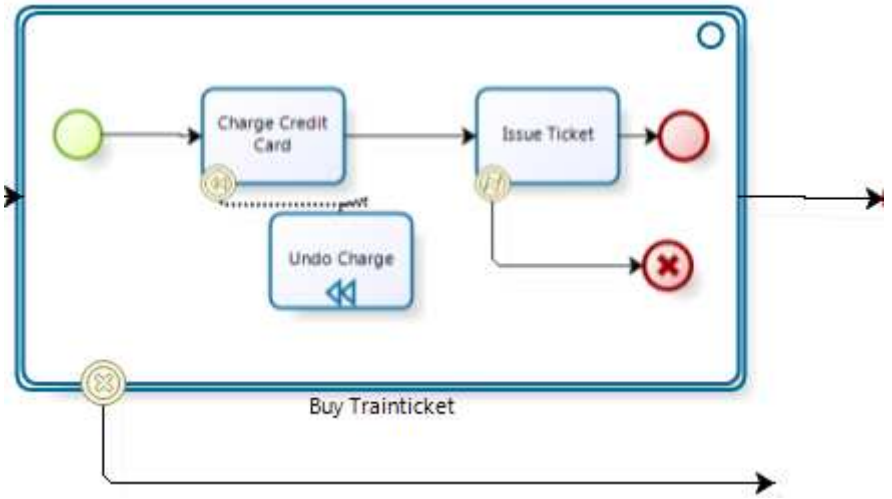
Throwing & Catching and Signals

- Message throw-catch only works between pools
- Error and Cancel throw-catch only send from an end-event of a subprocess to the boundary of that subprocess
  - Error/cancel is always terminating the subprocess
  - An error/cancel cannot be caught by a start event or a event-based gateway
- Escalation throw-catch can be thrown as intermediate event or as end event to the boundary of the same subprocess

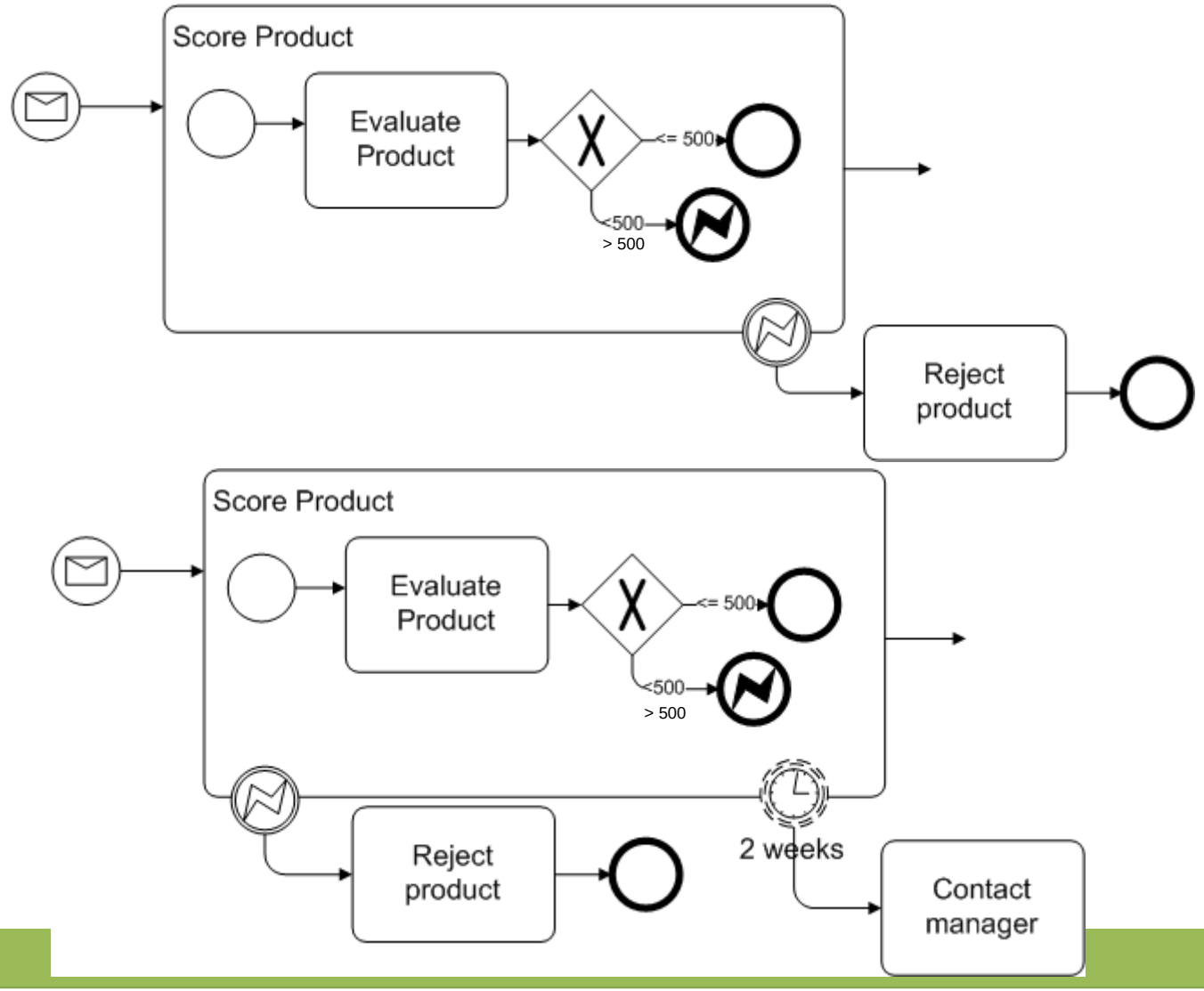
Special End Events

- Error
  - immediately ends process level
  - propagates error to parent level
  - result is caught by intermediate event attachted to boundary
- Escalation
  - similar, but non-interrupting
- Cancel
  - similar, interrupting
  - only used in transactions



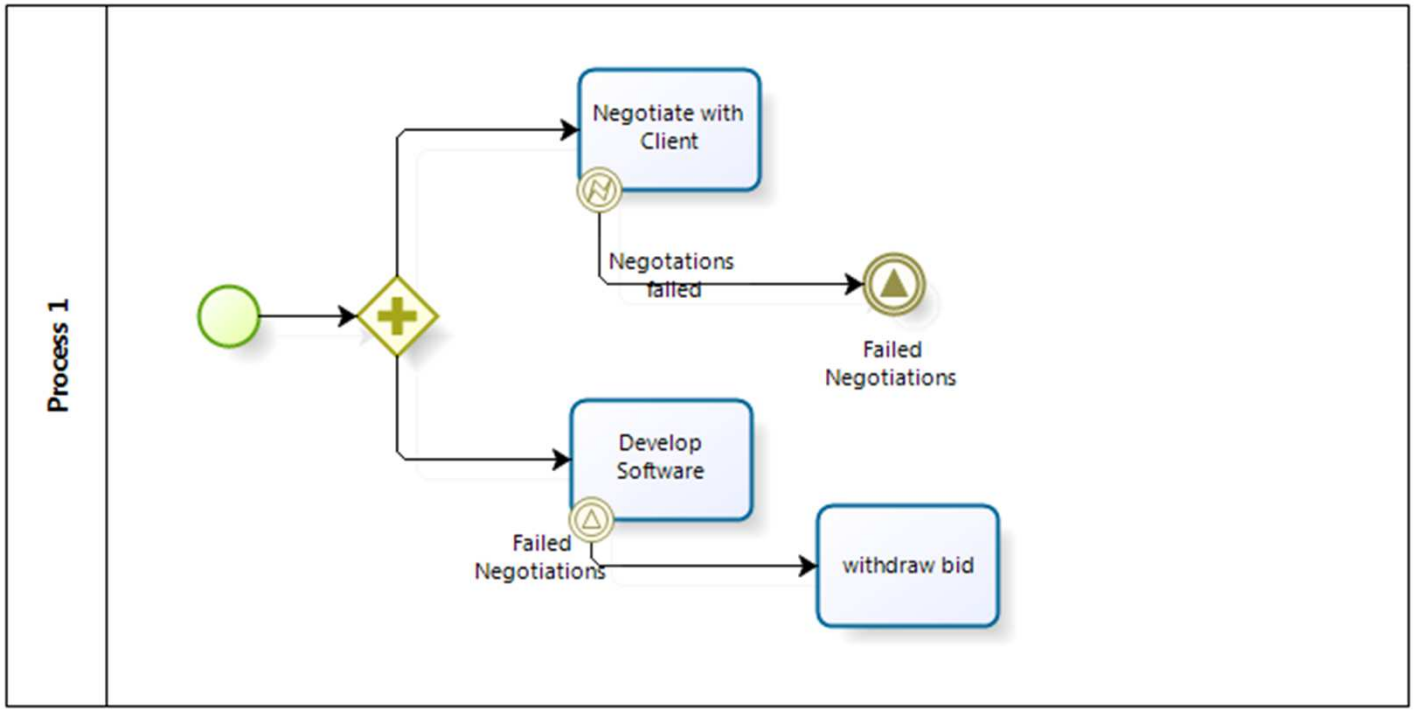


Throwing, Catching and Non-interrupting events



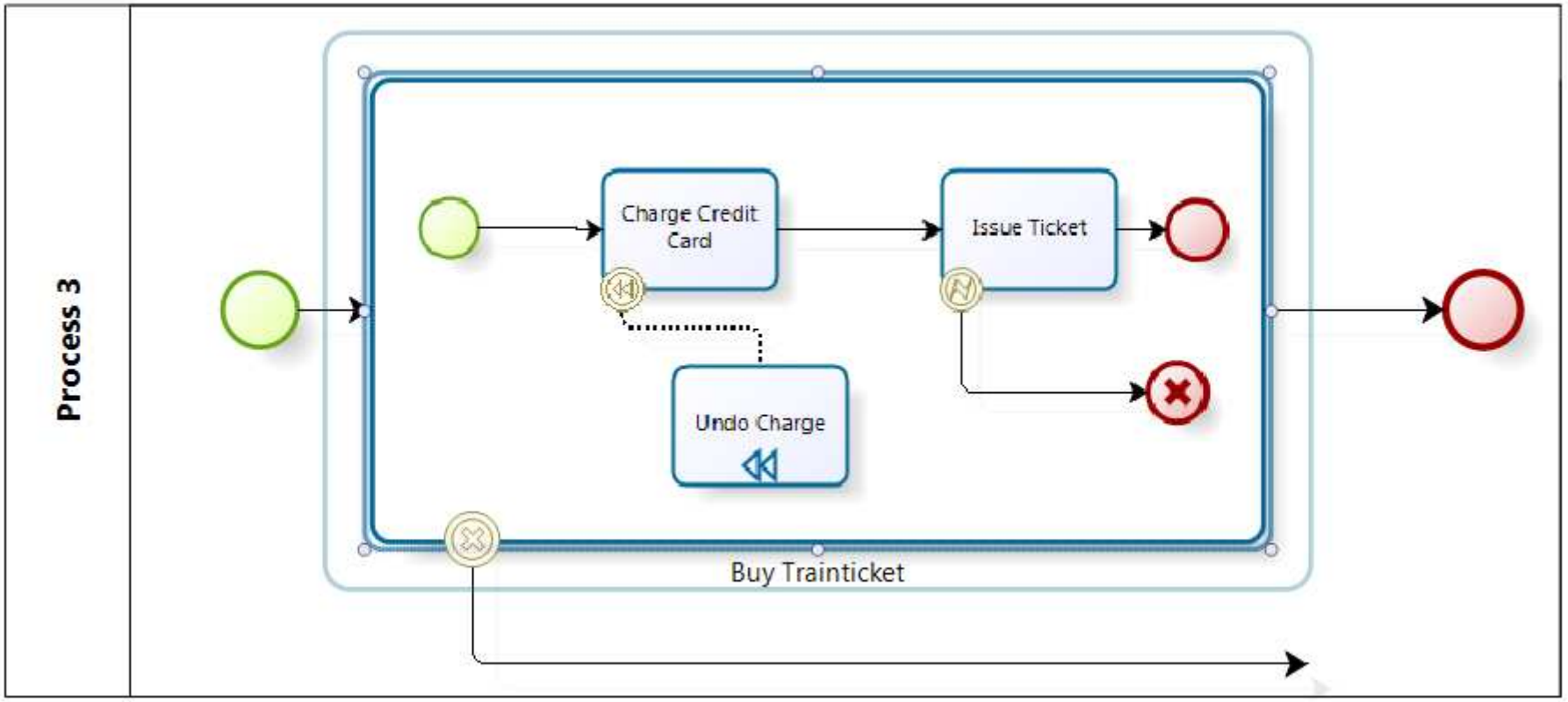
Throwing & Catching and Signals

- Signal can communicate within a pool and across pools.
  - Because it is not directed at a particular pool, throw-catch events are not linked graphically but by using matching labels.



Transactions

- Modelled using error, cancel and compensation events



Method & Style

- Step 1: Define process scope
- Step 2: Create Top Level Diagram for Happy Path
- Step 3: Add top level exception paths
- Step 4: expand subprocesses to show details
- Step 5: Add intermediate message flows to External Pools (optional)

Method & Style

- Step 1: Define process scope
  - What constitutes the start ?
  - What determines when the process ends ?
  - What does an instance of the process represent ?
  - Who is the customer ?
  - Are there different ways of ending the process ?

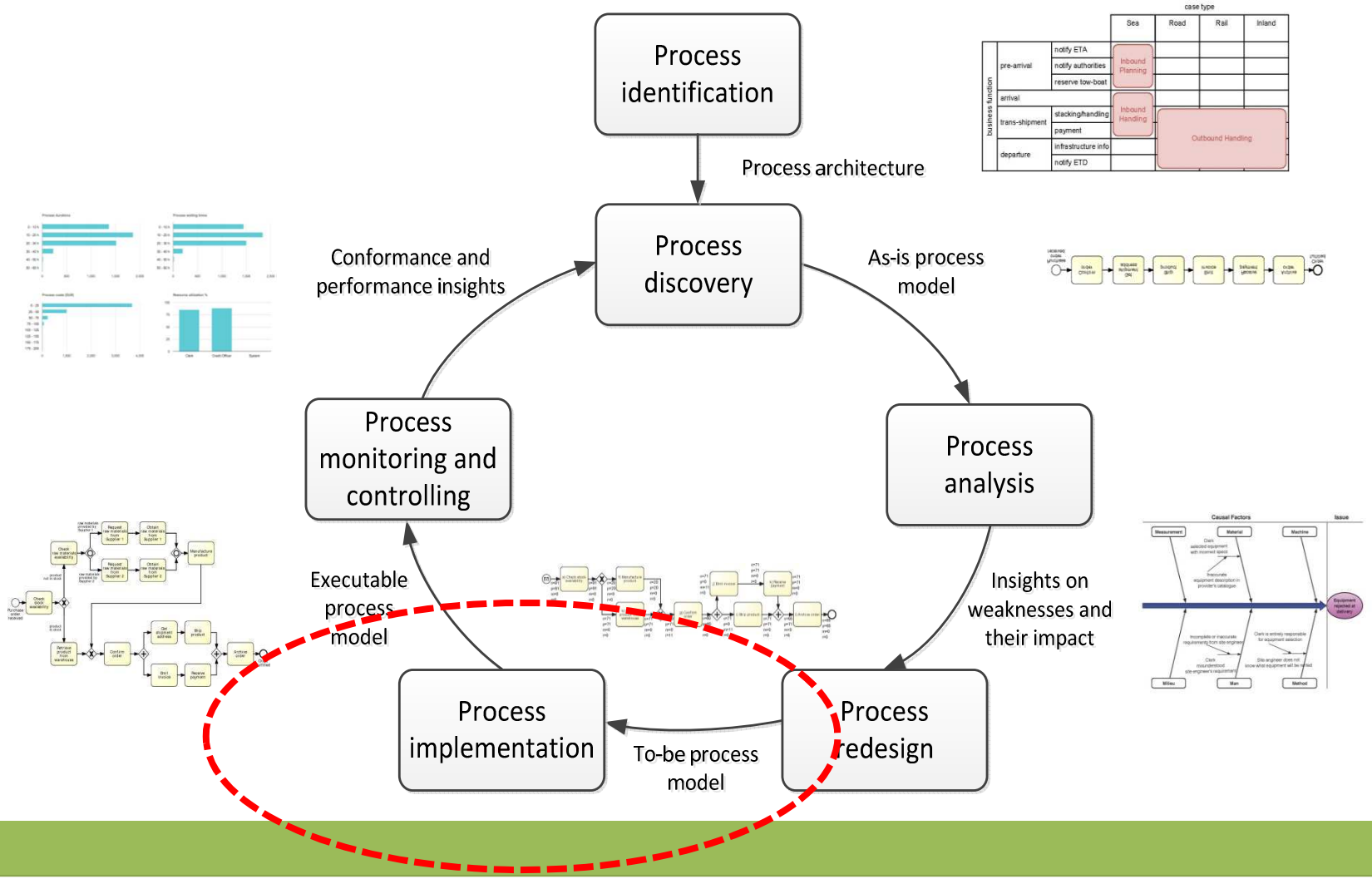
- Step 2: Create Top Level Diagram for Happy Path
  - Create a High level map: enumerate important steps
  - Organise them into a process model
    - Add Pools
    - Add lanes to Pools
    - Add Happy Path start and end-event to Process Pool
    - Add major steps in Happy Path
    - Reconnect concurrent and conditional steps

Method & Style

- Step 3: Add top level exception paths
  - Identify exception end states
  - Insert End Event for each Distinct end state
  - Insert Gateway to Define Exception Paths
- Step 4: expand subprocesses to show details
- Step 5: Add intermediate message flows to External Pools (optional)
  - Add black box pools for external participants
  - Add top-level message flows
  - Add message flows to child-level diagrams

Business Process Management

...design, analyze, execute and monitor business processes...



Business Process Enactment

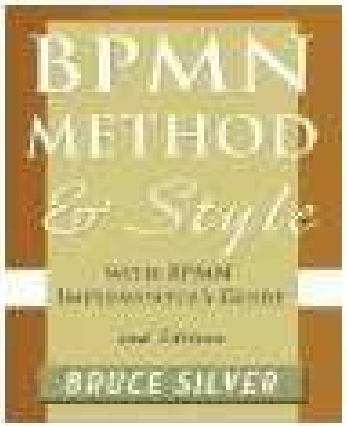
- BP Management Systems can manage tasks across many executing actors
- BPMS has connectors with supporing functional applications and can call service to invoke business logic in the applications
- BPMS provides facilities for acitivity management and performance evaluation
- Bizagi demo video
  - promo, but good & simple illustration of BPMS capbilities
  - http://player.vimeo.com/video/67221811?autoplay=1

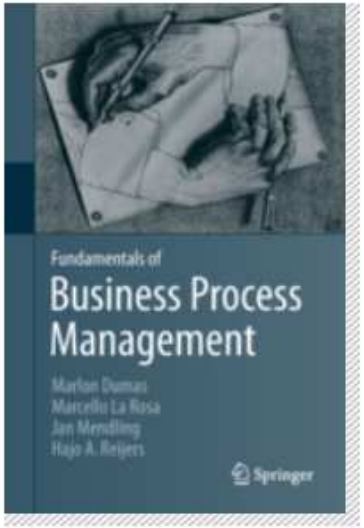
BPM Tools overview

- http://en.wikipedia.org/wiki/Comparison_of_Business_Process_Modeling_Notation_tools

Readings

BPMN Method & Style Bruce Silver, 1st or 2nd edition





Fundamentals of Business Process Management

Dumas, M., **La Rosa**, M., **Mendling**, J., **Reijers**, H.A.

2013, XXVII, 399 p. 161 illus., 7 illus. in color.

Free Preview

Available Formats:

eBook

(gross) price for Belgium

51,16 €

Add to cart

ISBN 978-3-642-33143-5

SME SPIRE

Thank you for your attention !

Questions ?
monique.snoeck@kuleuven.be

SADL