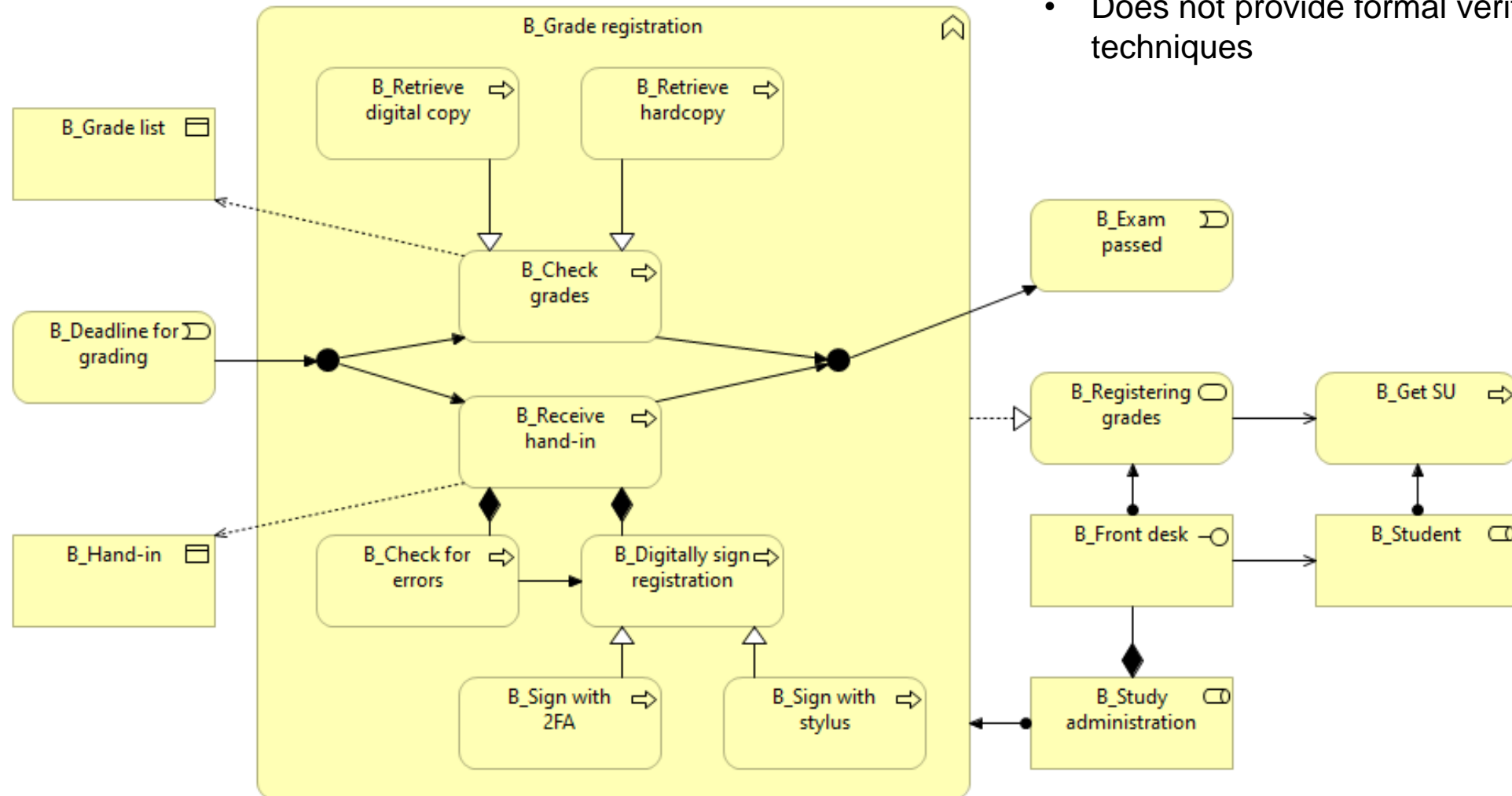


02291 System Integration

Introduction to BPMN

© Giovanni Meroni

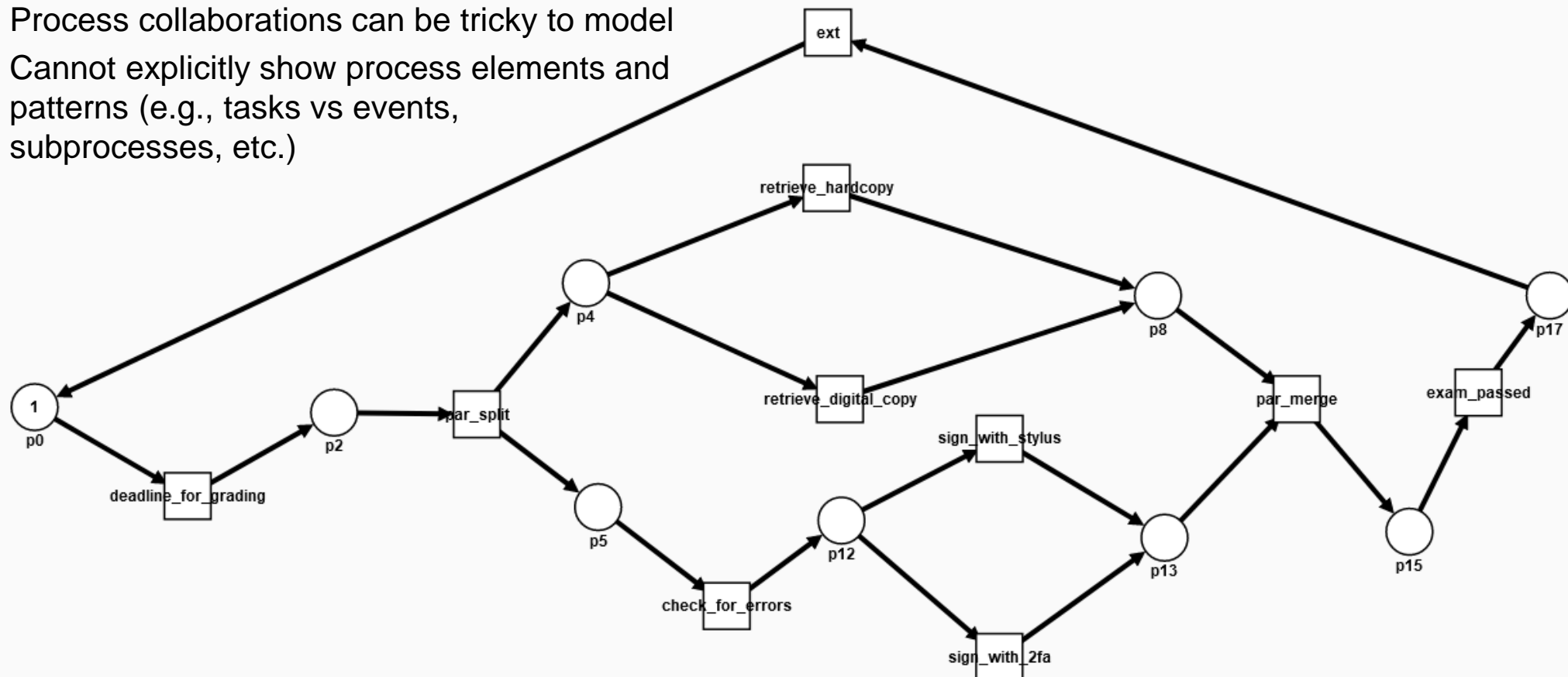
Behavioral models in ArchiMate



- Does not provide formal verification techniques

Behavioral models in Petri Nets

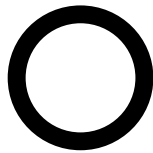
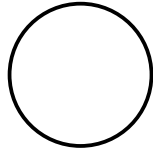
- Difficult to understand by non-experts
- Process collaborations can be tricky to model
- Cannot explicitly show process elements and patterns (e.g., tasks vs events, subprocesses, etc.)



Introducing BPMN

- Business Process Modeling Notation
- A visual language to model business processes
- Can be understood by both computer scientists and non-computer scientists (Business Analyst, Process Designer, ...)
- Support both the analysis/definition of processes, and the execution through a BPMS (Business Process Management System)
- Supports Orchestration: Workflows, internal processes of an organization, private processes
- Supports Collaboration: Multi-party processes, B2B processes

Main elements



- Start Event
 - There must always be at least one
 - Indicates the start of the process
- End Event
 - There must always be at least one
 - Indicates the end of the process
- Activity
 - It represents a unit of work to be done
 - It can be atomic (task) or indicate a subprocess (more on this later)
- Control flow (also named execution flow)

Example 1

To get a refund, each user has to collect a specific form from the secretariat on the ground floor, fill it in, and finally hand it in to the payroll office on the third floor.

Example 1

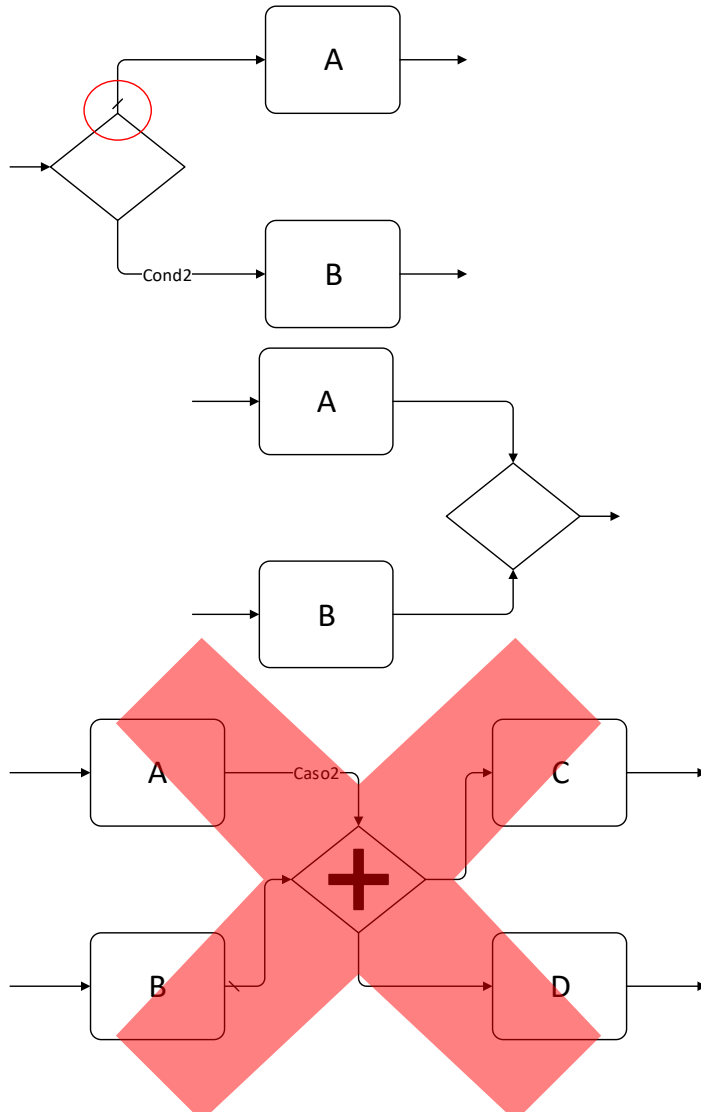
To get a refund, each user has to **collect** a specific **form** from the secretariat on the ground floor, **fill it in**, and finally **hand it in** to the payroll office on the third floor.



Gateways

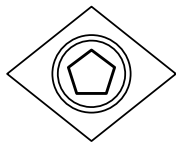
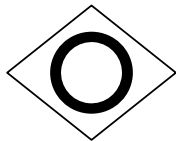
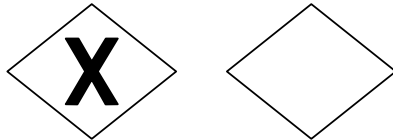
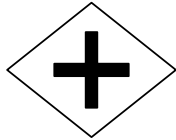
- An activity doesn't always need to be done
- Some activities may be mutually exclusive
- To improve efficiency, it would be desirable to carry out some activities in parallel
- Some activities may need to be repeated until a condition is met
- Gateways allow you to model these behaviors

Gateways



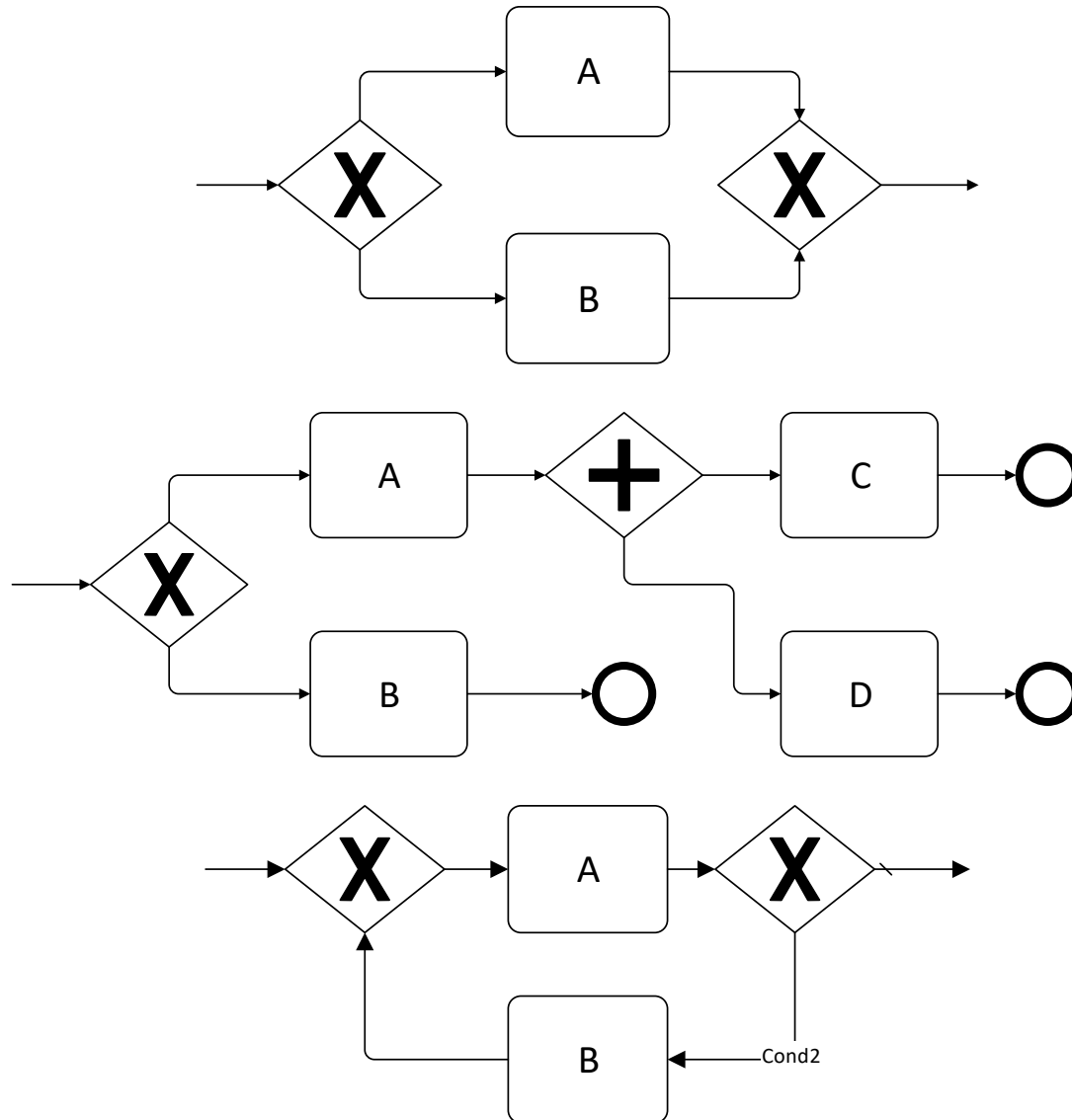
- Gateway in Split Mode
 - Splits the execution flow into multiple branches
 - The condition for a branch to be taken is specified on the flow
 - The default flow (if any) is marked with a dash
- Gateway in Merge Mode
 - Merges multiple execution flows
- Gateway in Hybrid Mode (deprecated)
 - Merges multiple execution flows and immediately splits them again
 - Ambiguous, use a gateway in merge mode followed by a gateway in split mode instead

Gateways



- Parallel gateway (AND)
 - In split mode, all branches will be taken
 - In merge mode, all branches being merged must complete
- Exclusive gateway (XOR)
 - In split mode, only one branch will be taken
 - In merge mode, only one branch being merged must complete
- Inclusive gateway (OR)
 - In split mode, at least one branch will be taken
 - In merge mode, all **active** branches being merged must complete
- Event-based Exclusive gateway (XOR)
 - We'll discuss this later when we talk about the events

Gateways

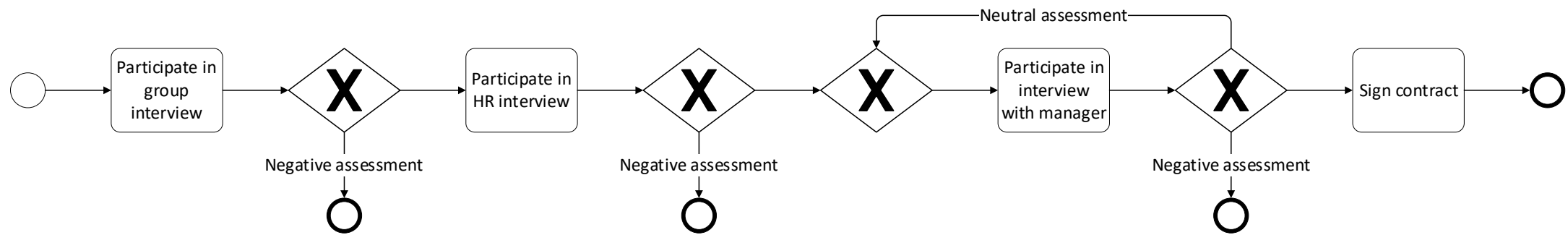


- Branches created by a gateway should be merged with a gateway of the same type
 - This prevents deadlocks or activities executed multiple times
 - When using an inclusive gateway, you must do so
- All branches must be merged or eventually end with an end event
- Loops are modeled using Exclusive gateways

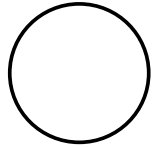
Example 2

The recruitment process of a company is structured as follows: first, a candidate participates in a group interview, which is meant to assess his/her problem-solving skills. If the assessment is negative, the process ends. Otherwise, the candidate participates in an individual interview with the HR staff, which is meant to assess his/her soft skills. If the assessment is negative, the process ends. Otherwise, the candidate participates in an interview with a manager to assess his/her technical skills. If the assessment is positive, the candidate signs the employment contract, and the process ends. If the outcome is negative, the process ends. If the outcome is neutral, the candidate participates in a new interview with another manager.

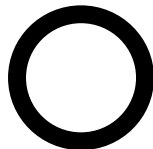
Example 2



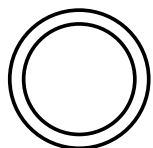
Events



- Start Event
 - There must always be at least one
 - Indicates the start of the process
 - Cannot have incoming control flows

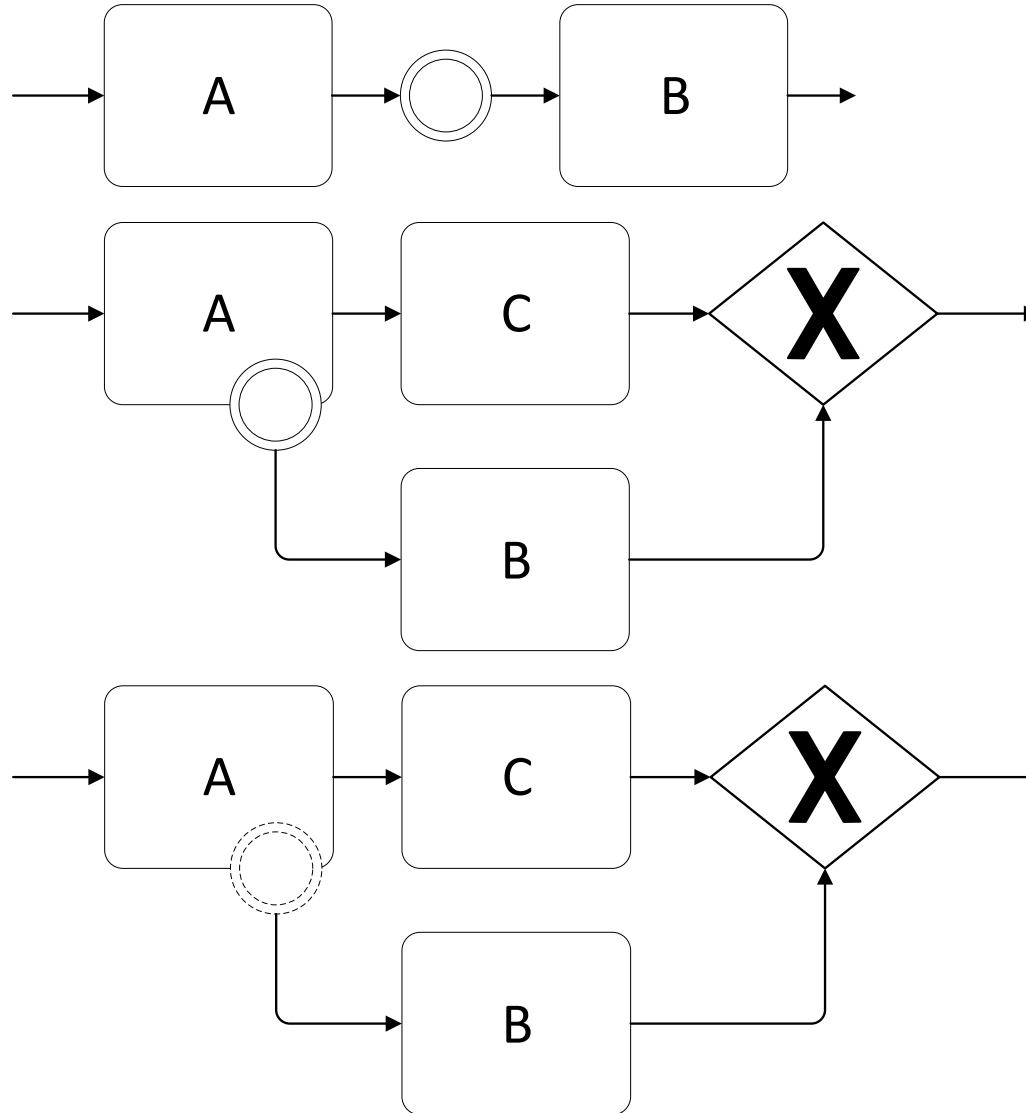


- End Event
 - There must always be at least one
 - Indicates the end of the process
 - Cannot have outgoing control flows



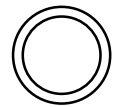
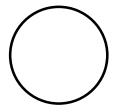
- Intermediate event
 - Indicates the occurrence of an event during the execution of the process
 - Can be applied to an activity or to a control flow

Events



- Applied to a control flow
 - Catch event: stops the flow until the event occurs
 - Throw event: causes the event to occur
- Applied to an activity (interrupting)
 - If the event occurs while the activity is running, interrupts the activity
- Applied to an activity (non-interrupting)
 - If the event occurs while the activity is running, triggers the outgoing flow from the event
 - The activity continues to run in parallel with the flow triggered by the event

Events



- Generic
 - Doesn't specify the event



- Message
 - Sender and recipient are known



- Signal
 - Recipient is unknown



- Timer
 - A date/time or interval

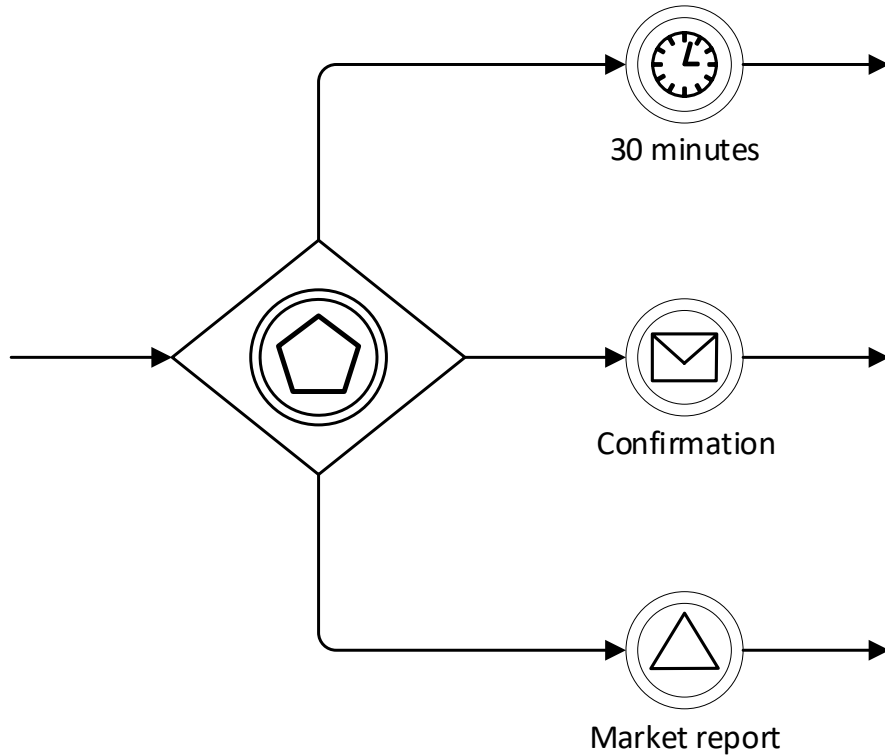


- Condition
 - One condition is valid



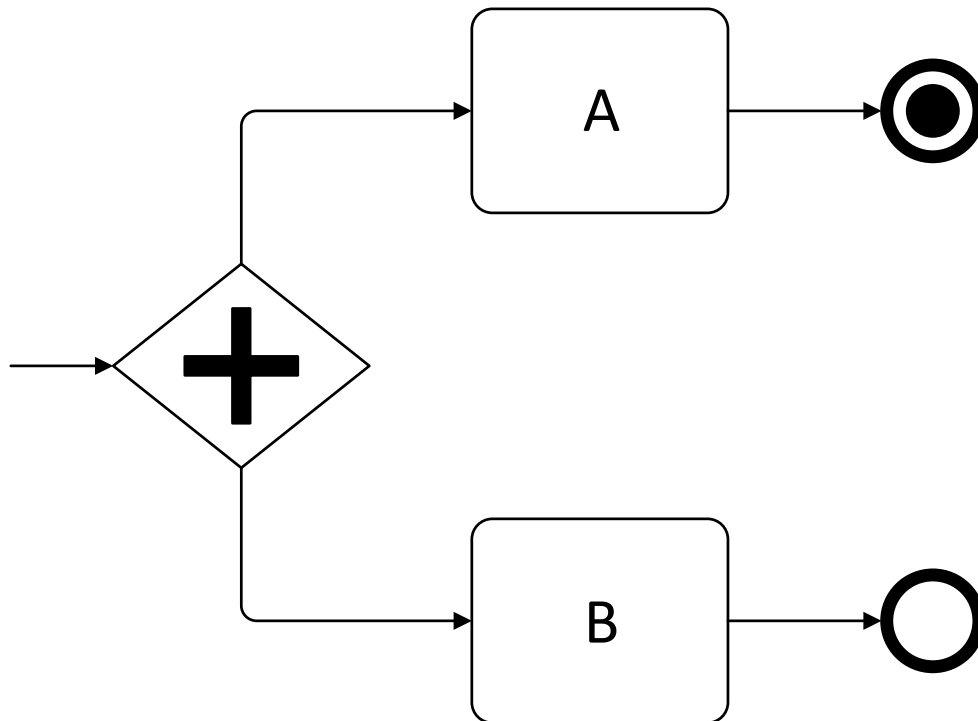
- Termination
 - Terminates the whole process

Event-based Exclusive Gateway



- Similar behavior to the Exclusive gateway
- The active branch is determined by the event
- If more than one event occurs, only the one that happened first is considered

Events

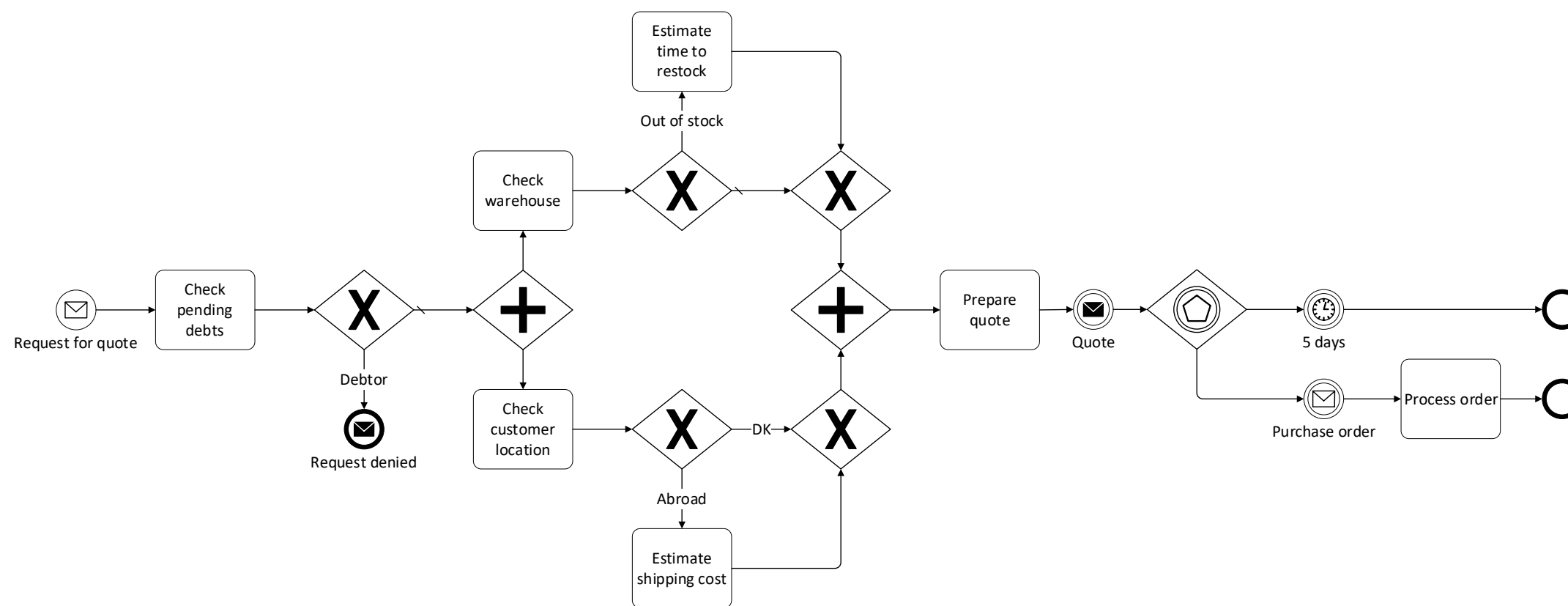


- End events generate an event
- The Terminate event terminates all active execution flows
- All running activities are terminated

Example 3

A supplier of electronic equipment receives a request for a quote from a customer. First, it checks if the customer has no pending debts. If this is the case, the request is rejected, and the customer is notified. Otherwise, the supplier checks the availability of the equipment being requested. If it is not in stock, it estimates the time to restock it. In the meantime, if the customer is located outside Denmark, the supplier also estimates the cost to ship the order. When all these steps are completed, the supplier sends the proposal to the customer and waits for a purchase order. When the order arrives, the supplier processes the order, and the process ends. If no response is received within 5 days since the quote was sent, the process ends.

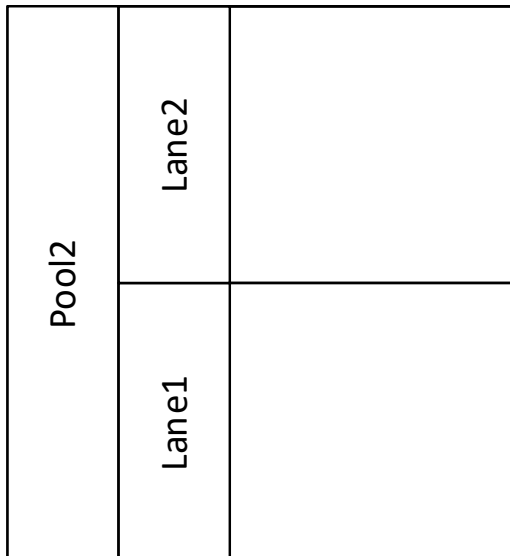
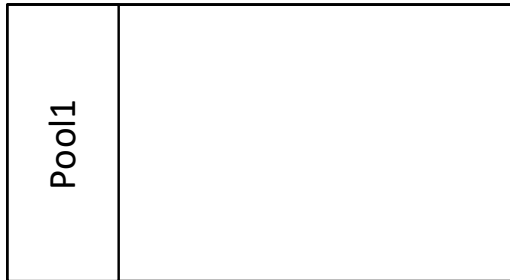
Example 3



Process collaboration

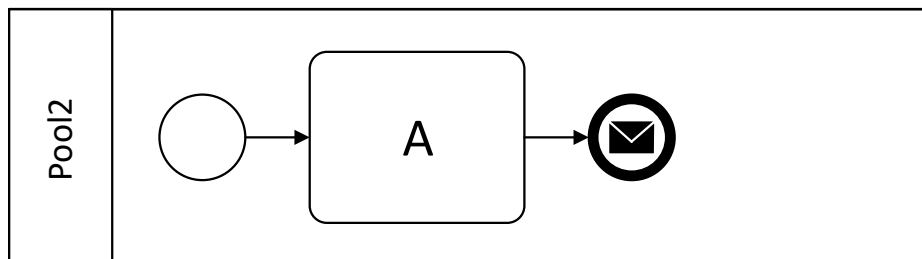
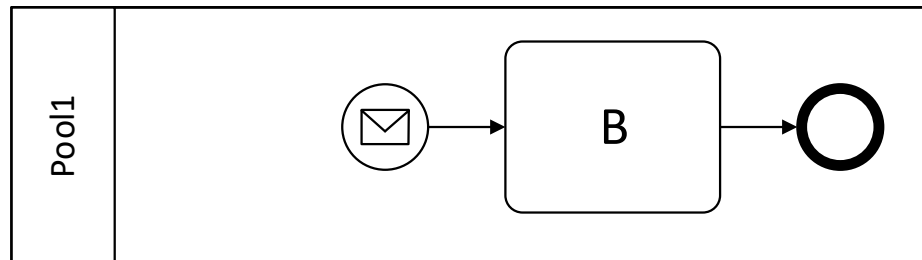
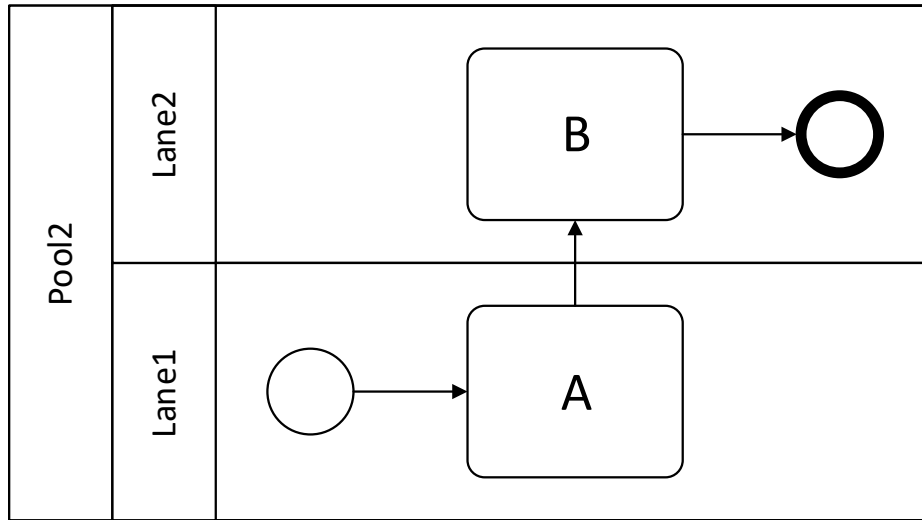
- So far, we have been working on modeling processes belonging to a single organization
 - Process Orchestration
- BPMN also allows you to model interactions between processes belonging to distinct business entities
 - Cross-process collaboration

Pools and lanes



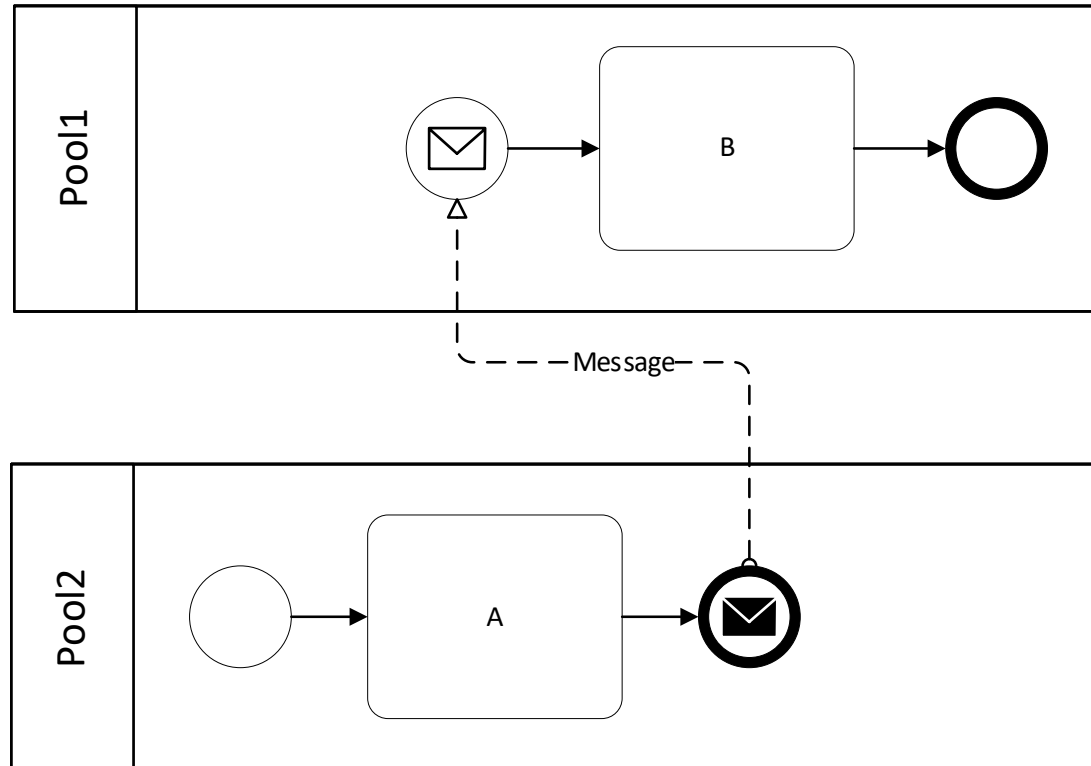
- Pool: Represents a separate business entity
 - A pool can be treated as a black-box or its processes made explicit (only if known)
- Lane: Represents a partition of a pool
 - Typically, a role or division (e.g. administration)
 - It only makes sense inside a Pool

Pools and lanes



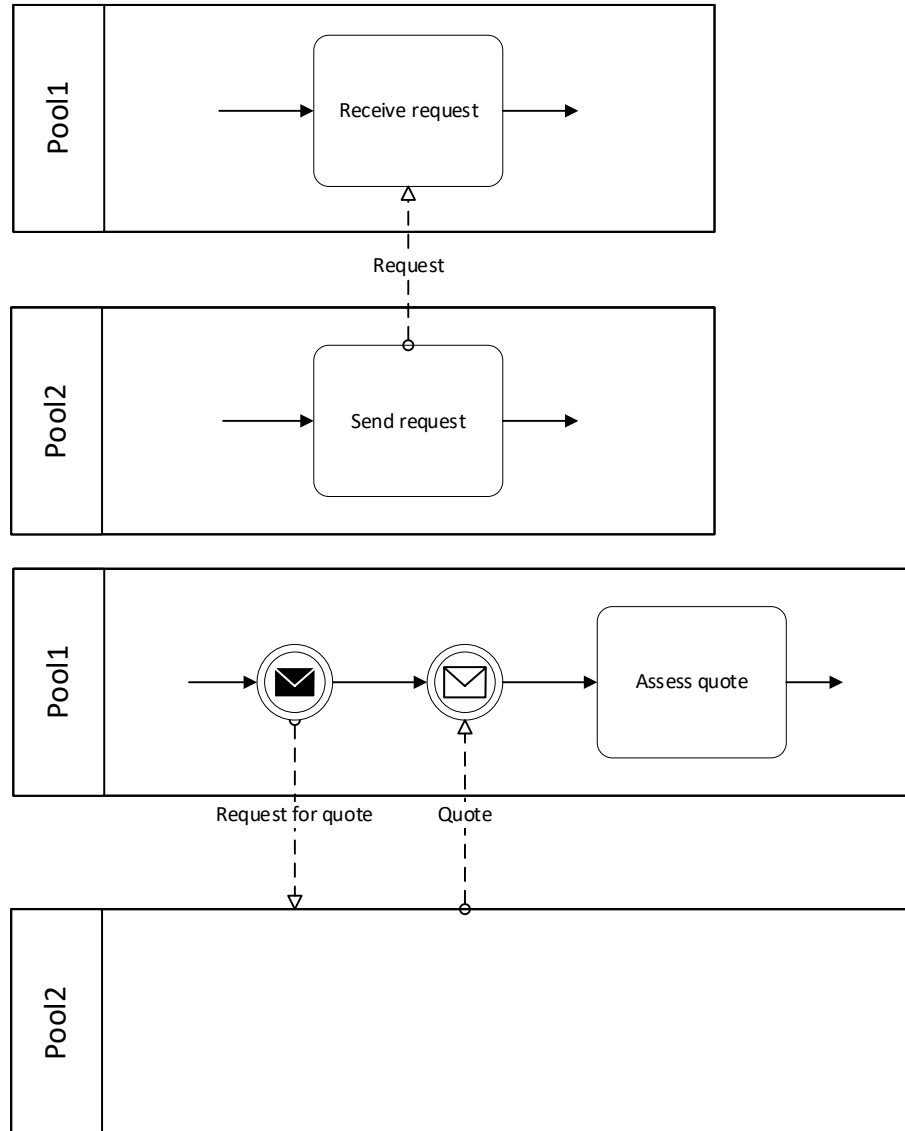
- Within a pool, coordination takes place through the control flow
- Between separate pools, communication takes place only through events
- It is **FORBIDDEN** to use the control flow outside of a pool

Pools and lanes



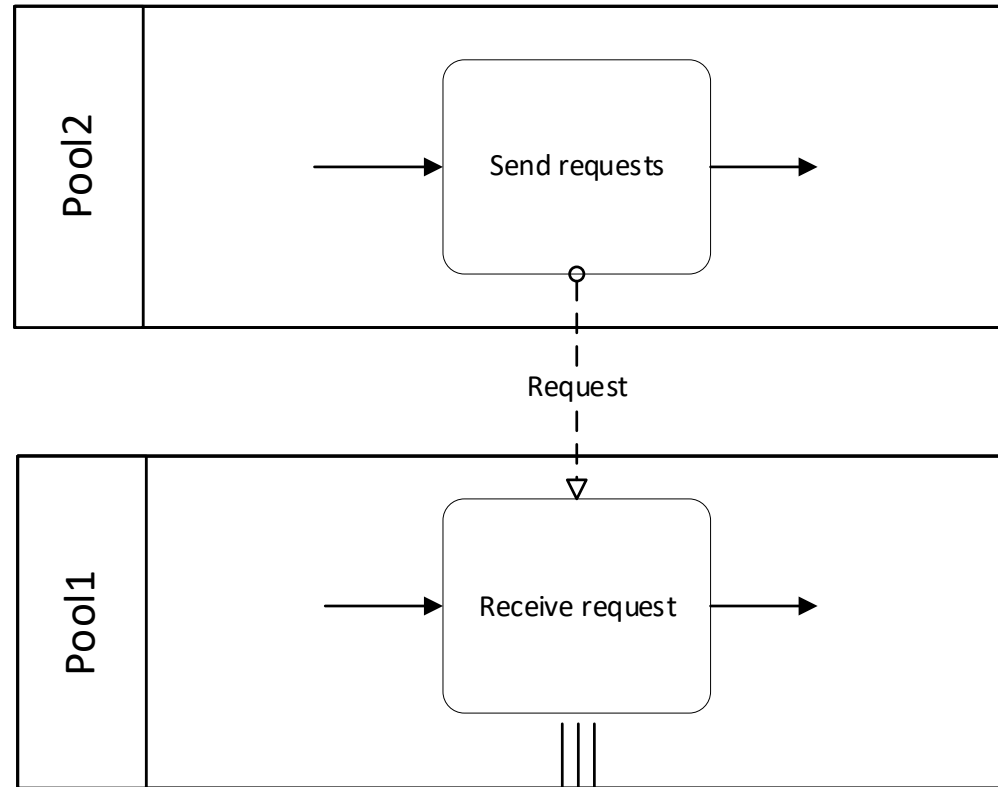
- It is possible to make explicit the coordination between processes by representing the message flow
- You can't use message flows within a pool
 - Coordination within a pool is done exclusively through the control flow

Message flow



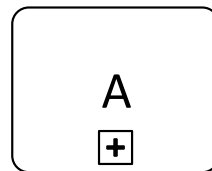
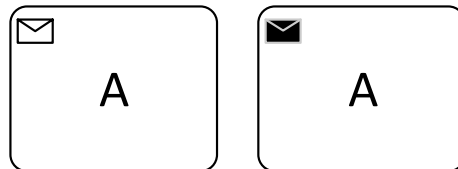
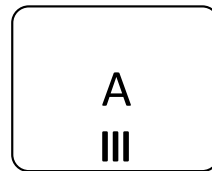
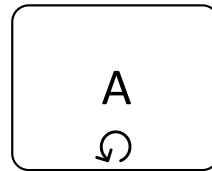
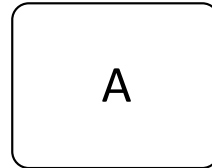
- Message flow can be used only with message events
- If the message flow is explicit, you can replace intermediate events with activities
- If the process inside a pool is not known, the message flow is connected to the pool

Multi-instance pools



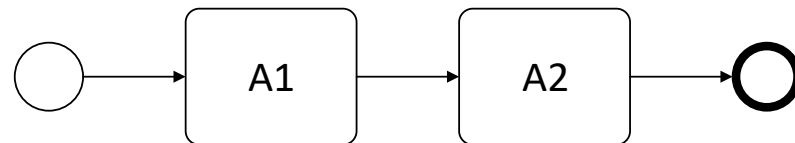
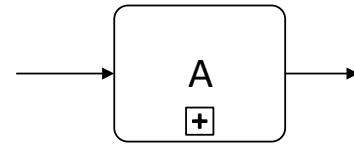
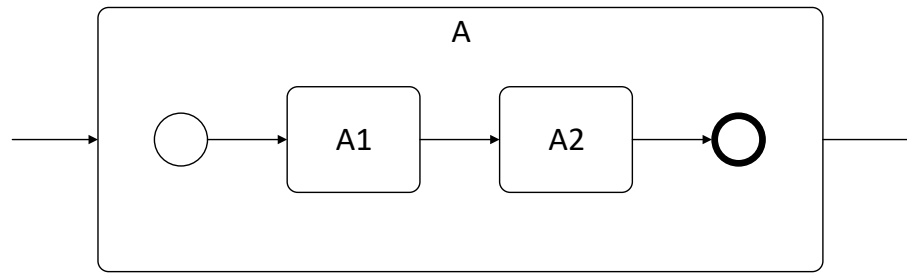
- Multiple business entities, all following the same process

Activities



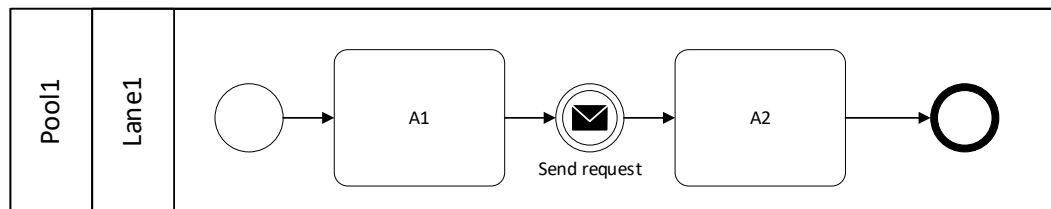
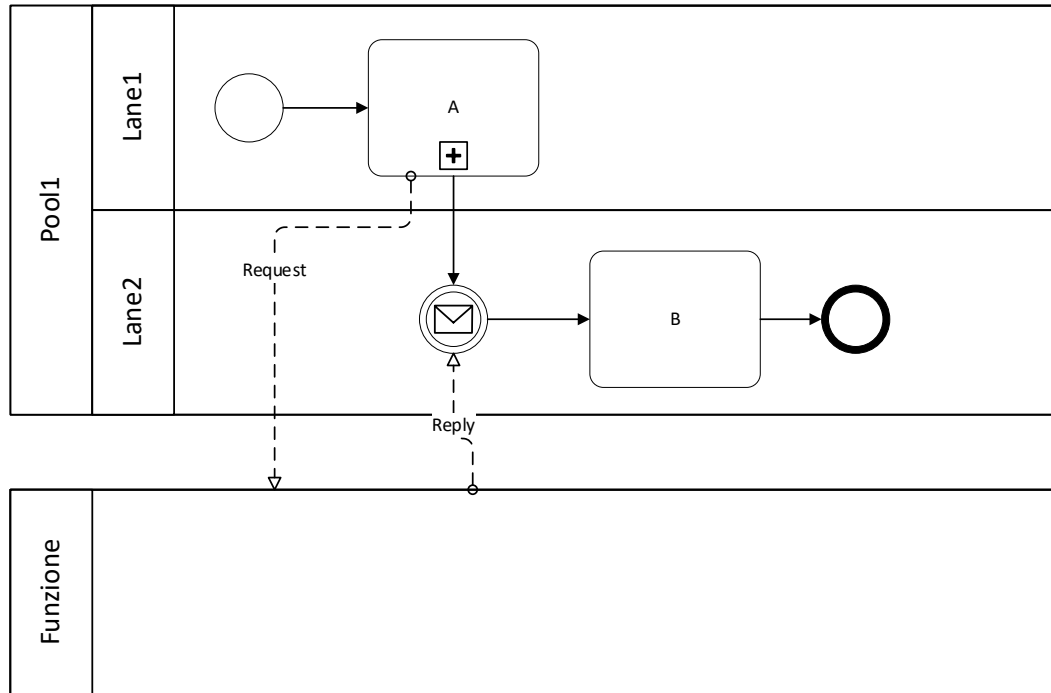
- Task
 - Atomic Operation Performed Only Once
- Cyclical activity
 - Atomic operation performed repeatedly until a condition is true
- Parallel multi-instance activity
 - Multiple operations of the same type carried out in parallel
- Receive/Send Tasks
 - Allows sending/receiving a message (similar to a Message event)
- Sub-process
 - A non-atomic operation that can also be represented as a process

Subprocesses



- It is possible to make a subprocess explicit by representing it directly inside the activity of the parent process where it is executed
- You can also represent it separately in another diagram

Subprocesses

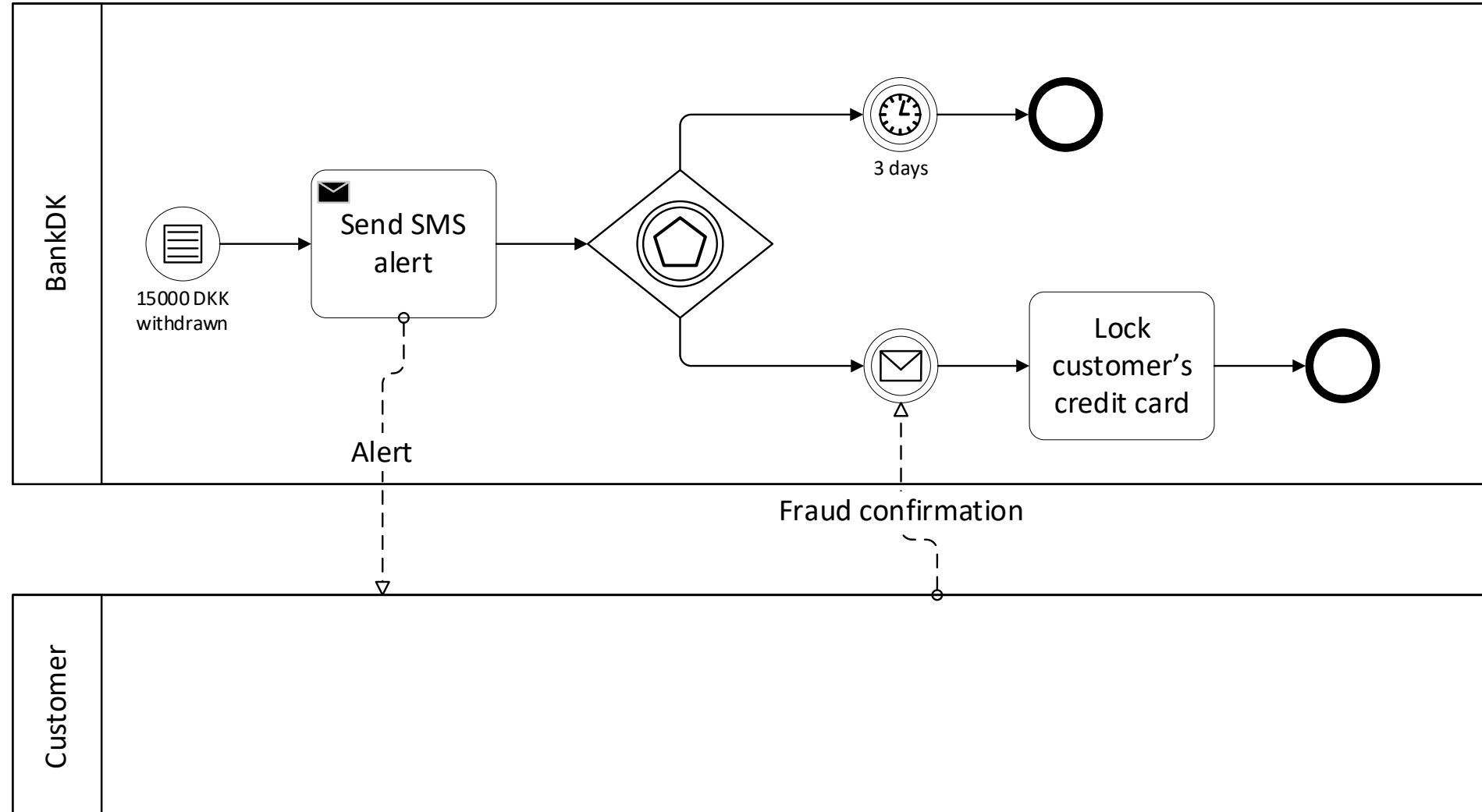


- A subprocess can contain only the pool and lane of the parent process
- Collaboration between actors cannot be represented within a sub-process

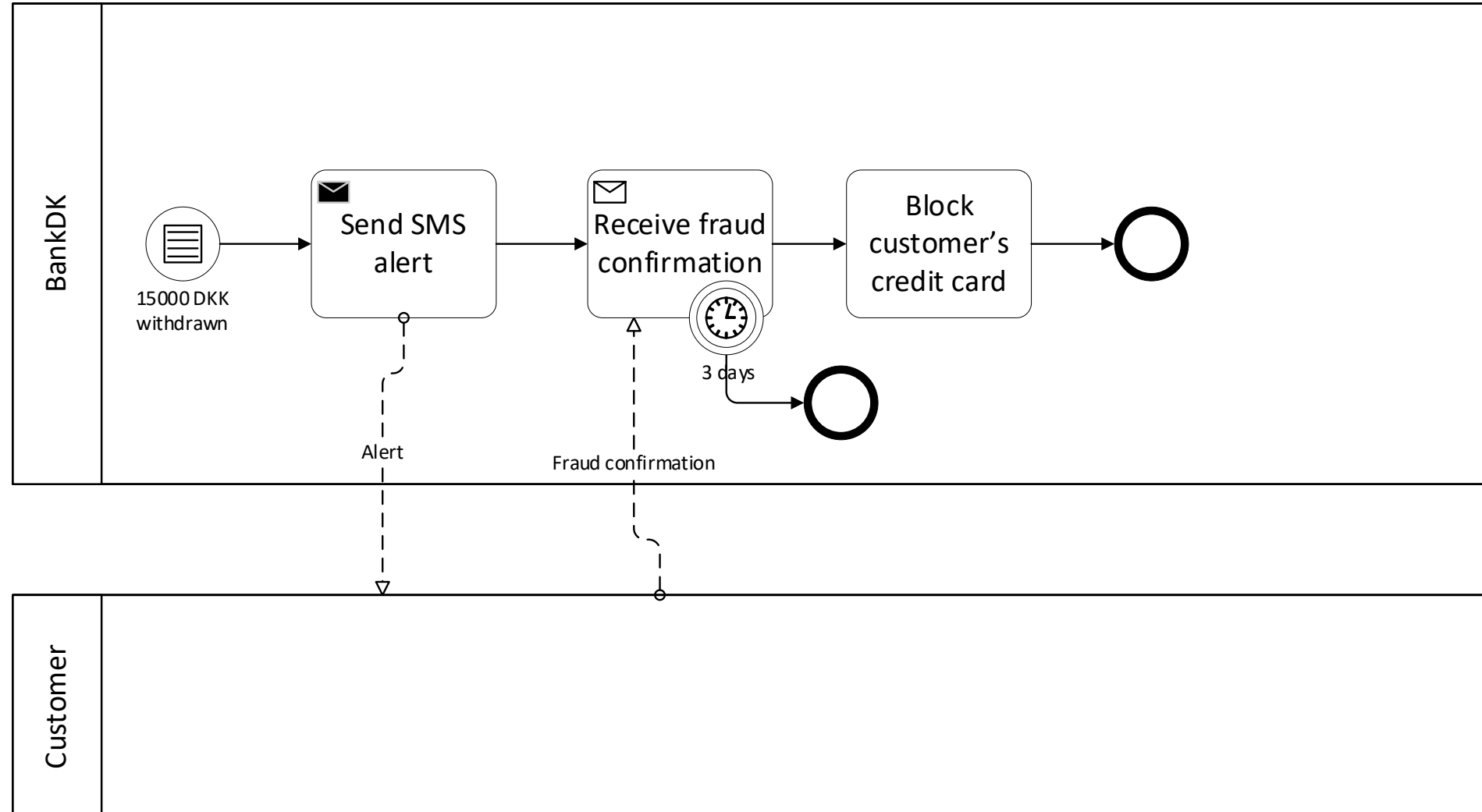
Example 4

The fraud detection process of BankDK is structured as follows. If withdrawals made with a customer's credit card at ATMs exceed 15000 DKK, BankDK sends an SMS alert to the customer's mobile phone. If the customer replies to that alert with a message containing the text FRAUD, BankDK blocks the customer's credit card. If no message is received after 3 days since the SMS was sent, the process ends.

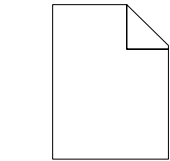
Example 4



Example 4 (alternative solution)



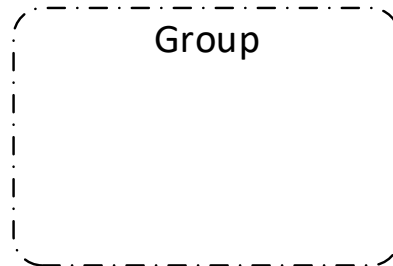
Artifacts



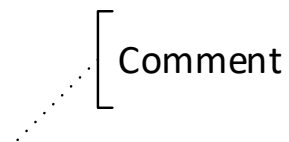
Data Object



Data store



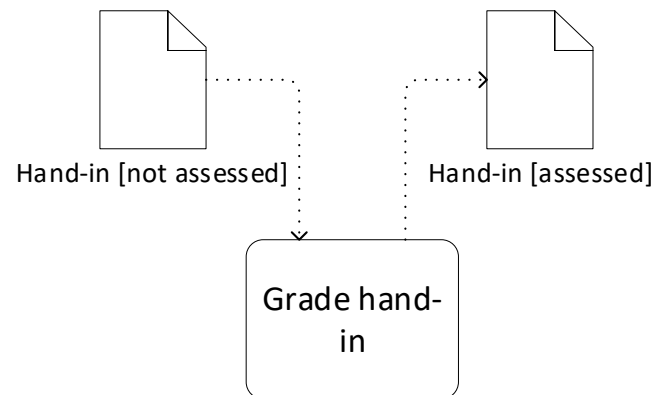
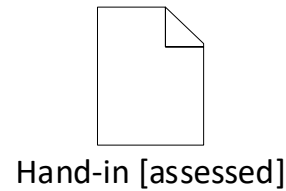
Group



Comment

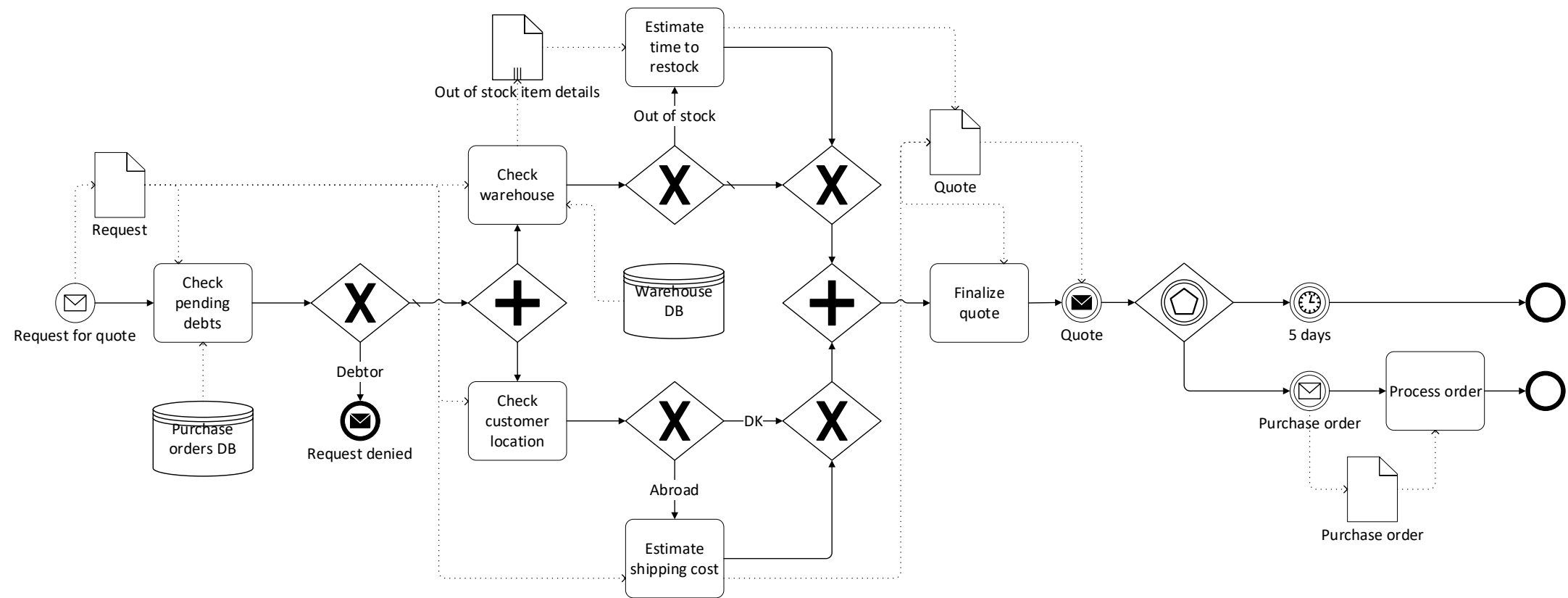
- Data object
 - Shows the physical or virtual objects used by the process
- Data store
 - Shows where the data reside
- Group
 - Groups related process elements
 - Has no effect on the execution of the process
- Annotation
 - Adds a comment to a linked element

Data objects



- Data collection
 - Aggregates multiple data objects of the same type
- Data state
 - Specifies the state of a data object
- Data association
 - Links a data object to an activity
 - Can be input, output, or both

Example 3 enriched



Study material

- Books and articles:
 - Dumas et al. - Fundamentals of Business Process Management (2nd Edition)
 - Available at: <https://link.springer.com/book/10.1007/978-3-662-56509-4>
 - Chapter 3 (not 3.6) and 4
 - (alternatively) Weske - Business Process Management
 - Available at: <https://link.springer.com/book/10.1007/978-3-662-59432-2>
 - Chapter 4.7
- Modeling tools:
 - bpmn.io: <https://demo.bpmn.io/>
 - (alternatively) Camunda Modeler: <https://camunda.com/download/modeler/>
 - (alternatively) SAP Signavio: <https://academic.signavio.com/p/login>

Exercises

Please answer all exercises to demonstrate your skills.
Solutions will be available at 11:45

Exercise 1 – MagiMug

MagiMug is a manufacturer of promotional mugs, which are sold to companies, universities, hotels, etc. In particular, MagiMug buys white ceramic mugs from a wholesaler and screen prints on them a picture provided by the customer.

When the number of cups in stock is less than 10000 units, MagiMug's administration prepares a purchase order and sends it to its wholesaler. Once the order has been received, the wholesaler contacts a shipper. Then, while waiting for the shipper to reach its premises, the wholesaler prepares a container to be shipped to MagiMug. If, while preparing the container, part of the load gets damaged, the activity is stopped, and the extent of the damage is estimated. At the same time, the damaged load is removed from the container. Once these two activities are complete, the container preparation is repeated. Once the container is ready and the shipper has reached the wholesaler's premises, the container is given to the shipper, who attaches it to his/her truck, and delivers it to MagiMug. Once the container is received, MagiMug's warehouse workers check the integrity of the load. If everything is in order, this is reported to the shipper, who in turn notifies the wholesaler of the success of the activity, and the process ends. If, on the other hand, part of the load is damaged, the shipper takes the container back to the wholesaler and the process ends."

Exercise 2 – AlText

AlText is a publisher of printed academic textbooks and scientific journals.

When the paper rolls that AlText has in stock fall below 500 rolls, a request for a quote for 2000 rolls is prepared and sent to one of its supplier. The supplier, upon reception of the request, verifies if it has enough paper rolls in its warehouses. If the supplier does not have enough rolls, it terminates its process by informing AlText. AlText then sends the request for a quote again to another supplier. If a supplier has enough rolls available, it prepares a quote, which is sent to AlText. If, after assessing the quote, AlText is satisfied with it, AlText notifies the supplier that the quote has been accepted, the supplier registers the order, and the process ends on both sides. If AlText is not satisfied with the quote, it sends the request for a quote again to another supplier. If a supplier does not receive the notification within 10 days since the quote was sent, it ends the process.

Exercise 3 - CompGears

CompGears, a consumer electronics company, wants to model its helpdesk's technical support process.

The process begins when a customer contacts the company's helpdesk with a technical support request, communicating his/her identity. First, the helpdesk operator checks whether the customer is already present in the system and, if not, (s)he registers the customer. Then, the helpdesk operator asks the customer to provide the serial number of the product. Once this information is received, the helpdesk operator checks if that product is still under warranty. If not, the helpdesk operator ends the process by notifying the customer that the product is out of warranty. Otherwise, (s)he notifies the user that the request has been accepted and forwards it to an operator from the company's in-house technical department.

First, the technical operator asks the customer to describe the problem. Once this information is received, (s)he examines the symptoms and then looks for a possible resolution action, which is provided to the customer. The customer then verifies whether that resolution action solves his problem and communicates the outcome to the technical operator. If the resolution action is successful, the process ends on both sides. If not, the technical operator searches again for a resolution action. This is repeated until the problem is resolved. If the technical operator is busy for more than 30 minutes on a specific support request, the process ends by inviting the user to repeat the entire procedure again.