BPMN: modeling industry processes pt. 2

Gabriele Genovese
genovese@i3s.unice.fr

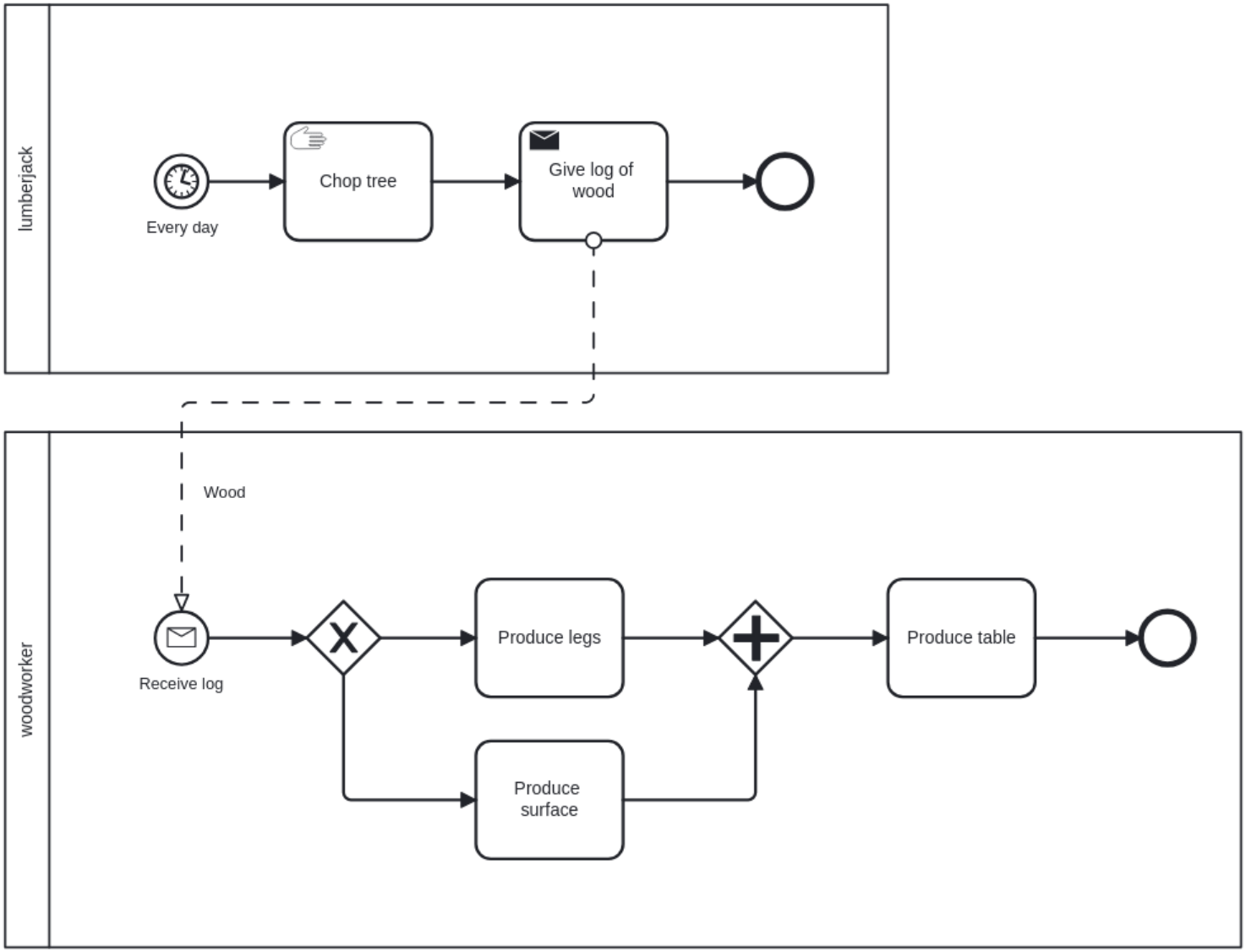
18 Febraury 2026

Exercise

- A woodworker can build a table only after both the legs and the tabletop have been completed.
- Each part requires one log of wood.
- The lumberjack delivers one log of wood to the woodworker every day.

Solution

Solution


lumberjack
Every day
Chop tree
Give log of wood
Wood
woodworker
Receive log
Produce legs
Produce surface
Produce table

What happens if I try to execute it?

What happens if I try to execute it?

Deadlock!

Best practice

- Never mix gateways:
  - if split exclusive and merge parallel: deadlock
  - if split parallel and merge exclusive: token duplication
  - if merge exclusive, split parallel and loop: token duplication
  - if merge parallel: deadlock
- Always use explicit join or split
- Use unique start and ending point

Static analyzer for BPMN

Verification for:

- Synchronization
- Guaranteed termination
- Unique end event execution
- No dead activities

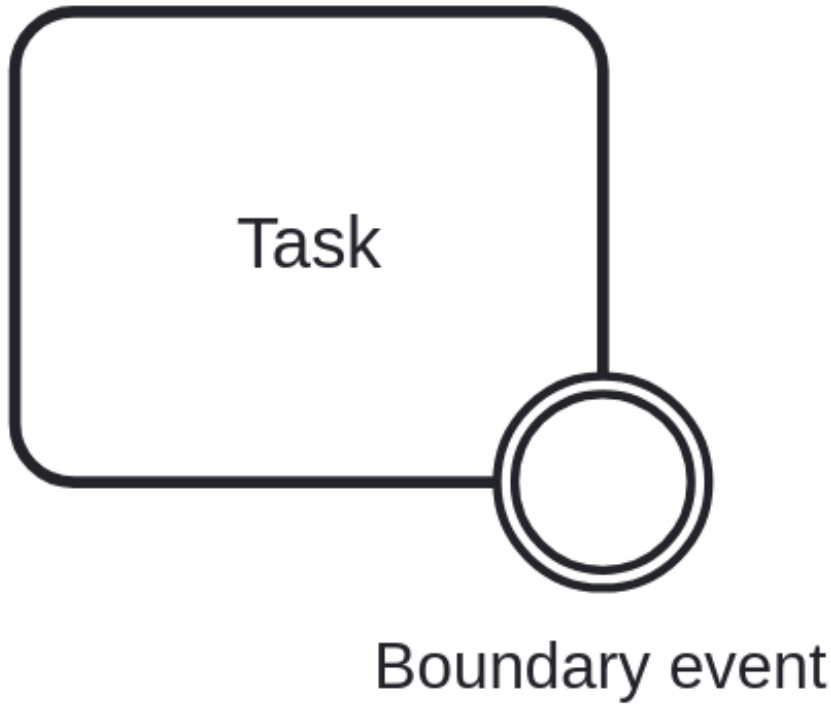
https://timkraeuter.com/bpmn-analyzer-js/

Intermediate and advanced aspects of BPMN

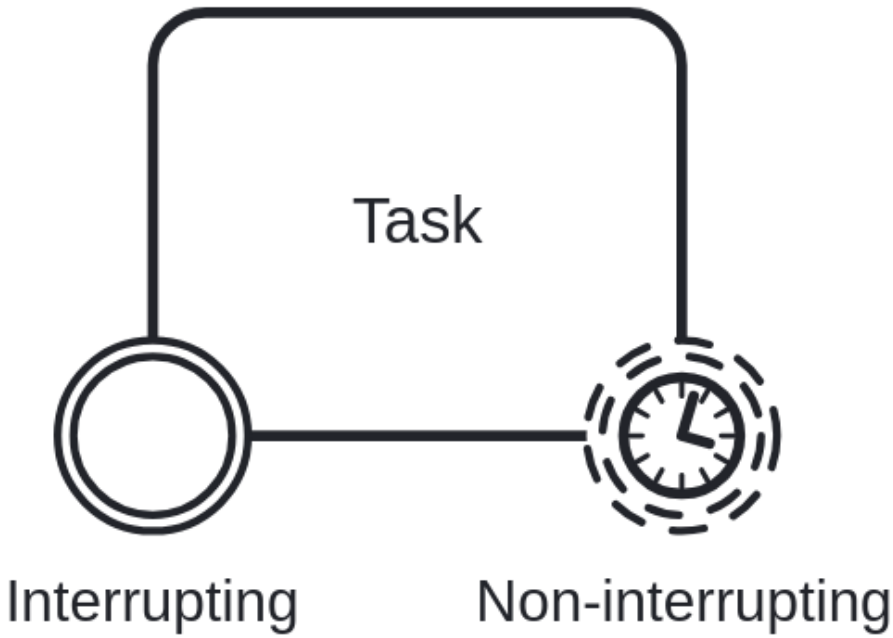
- Boundary events
  - Error
  - Compensation
  - Link
  - Escalation
- Sub-processes
- Objects

Boundary events

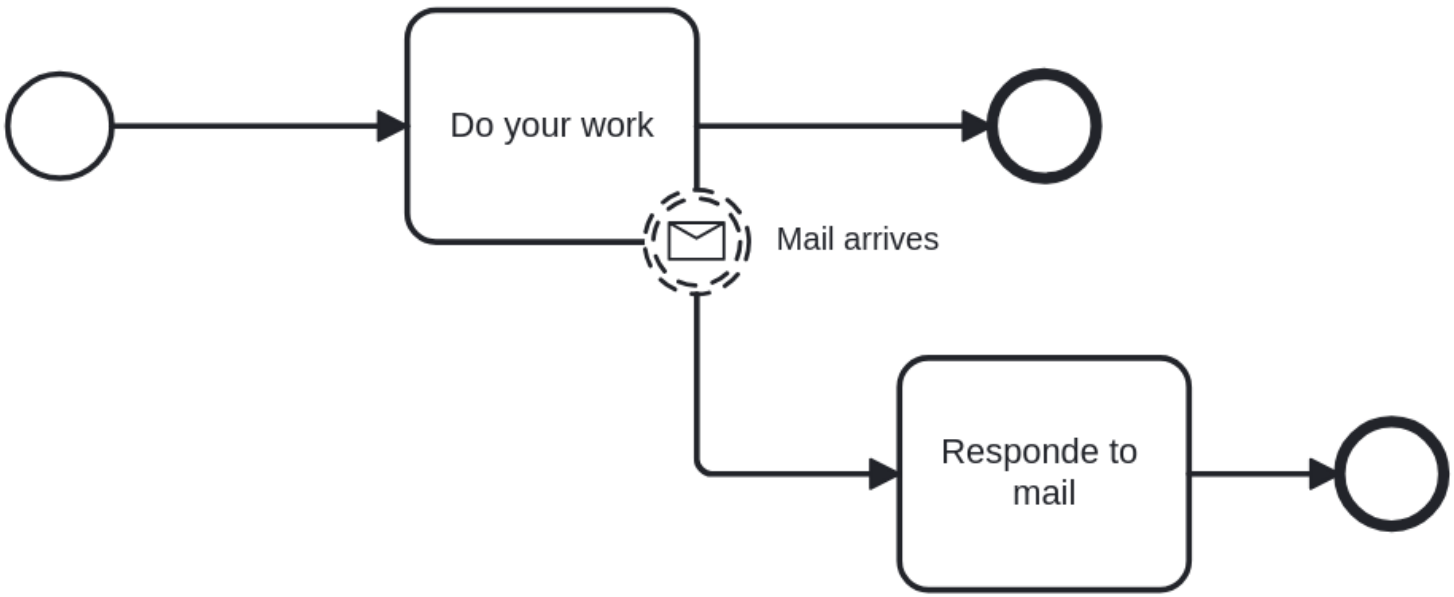
Catch an external event when the activity is active.



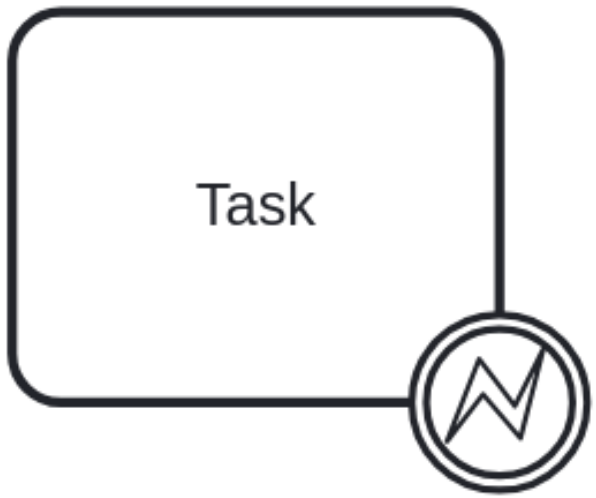
Two types of boundary events



Example



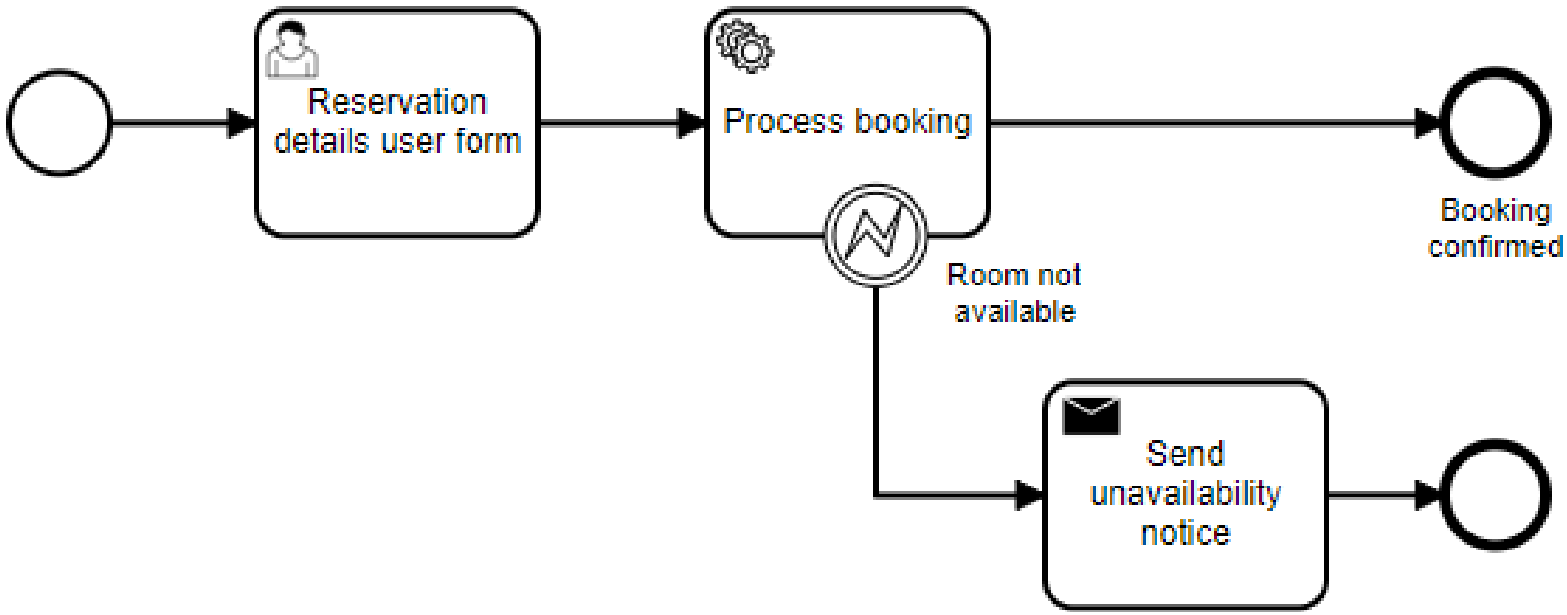
Error events



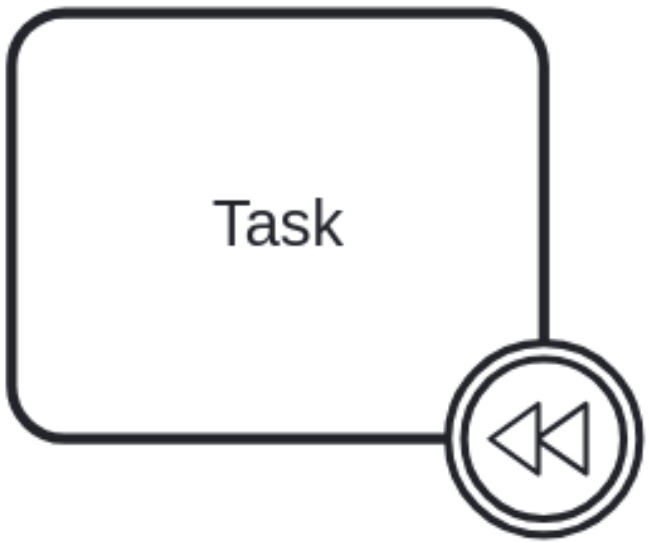
Error boundary event

Error end event

Example


Reservation details user form
Process booking
Booking confirmed
Room not available
Send unavailability notice

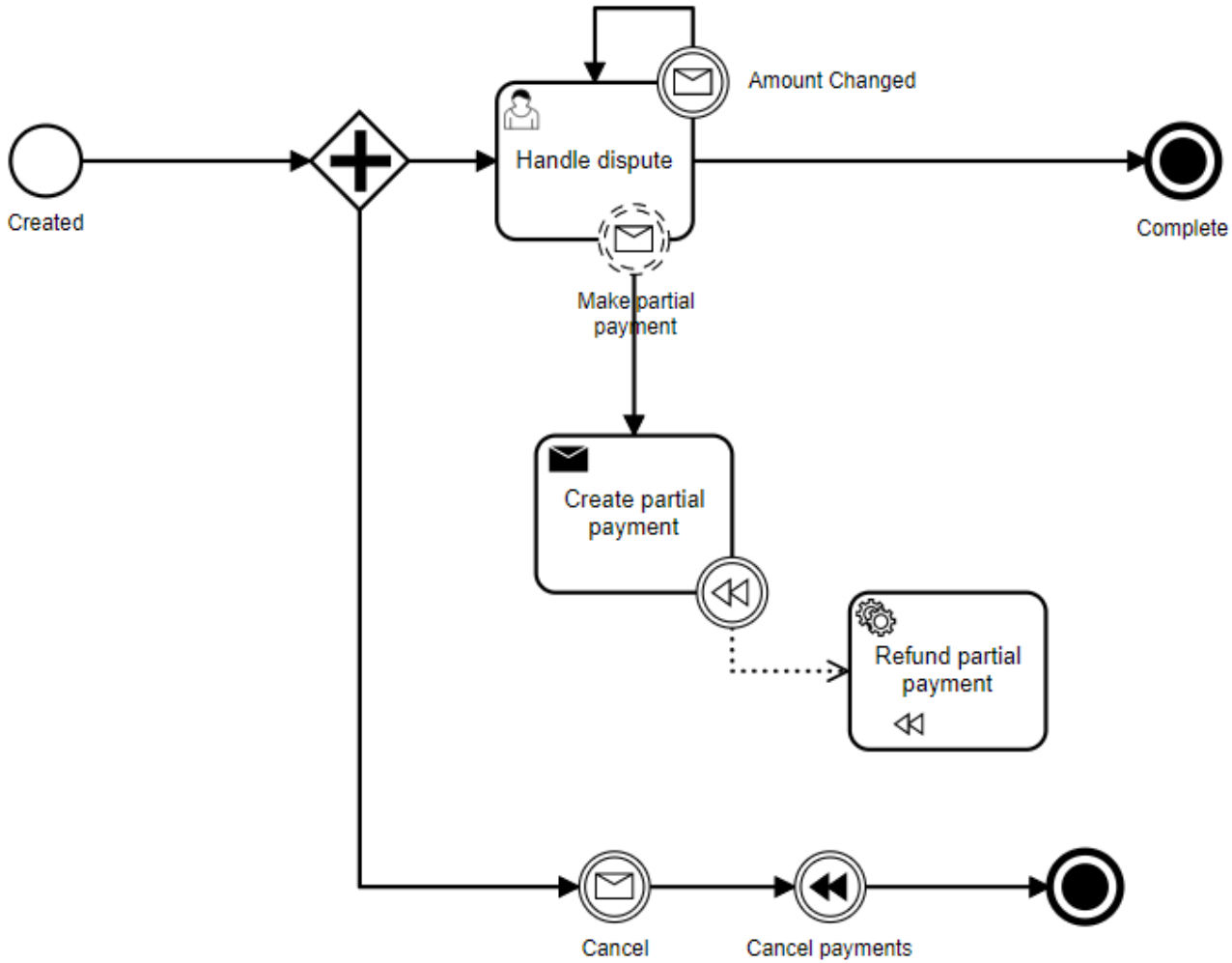
Compensation events



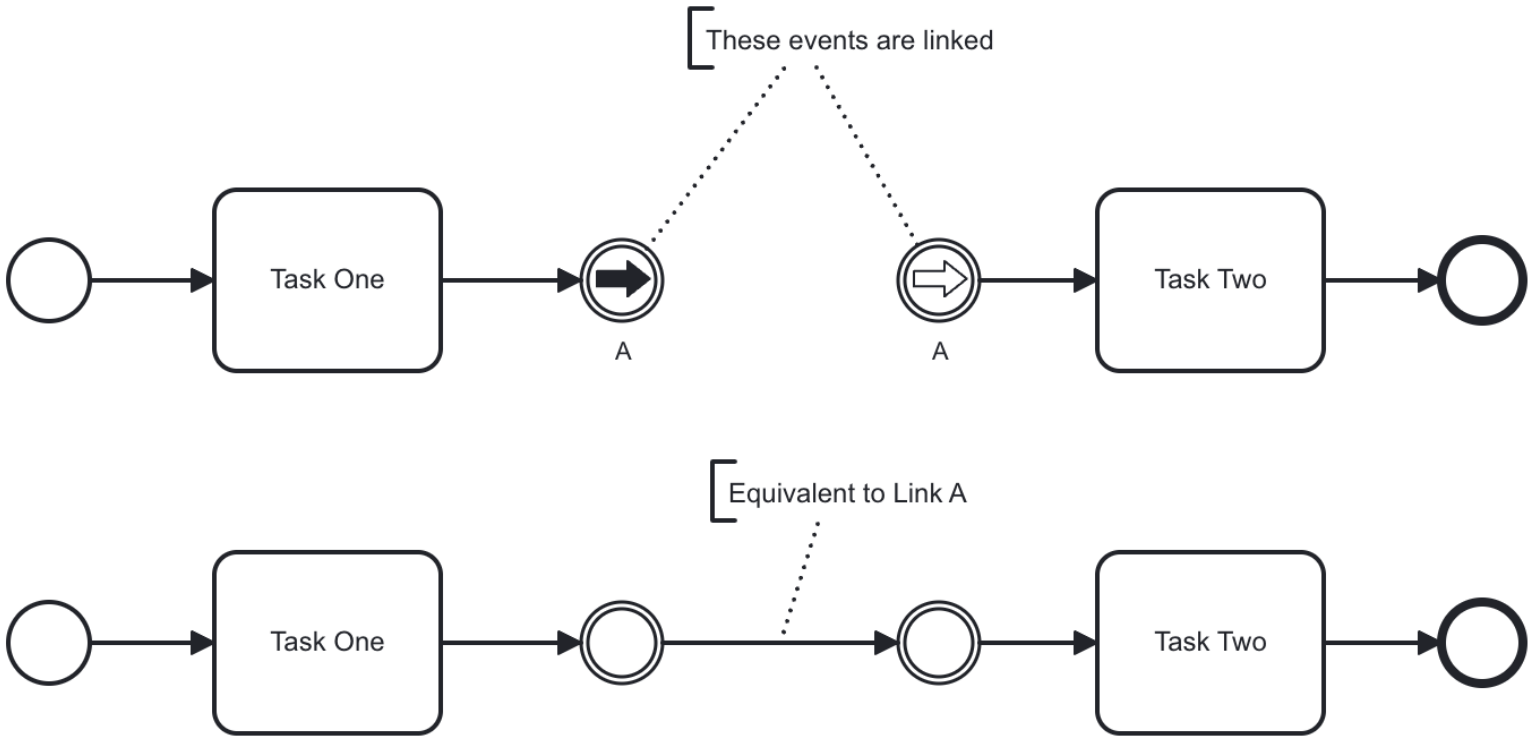
Compensation
boundary event

Compensation
end event

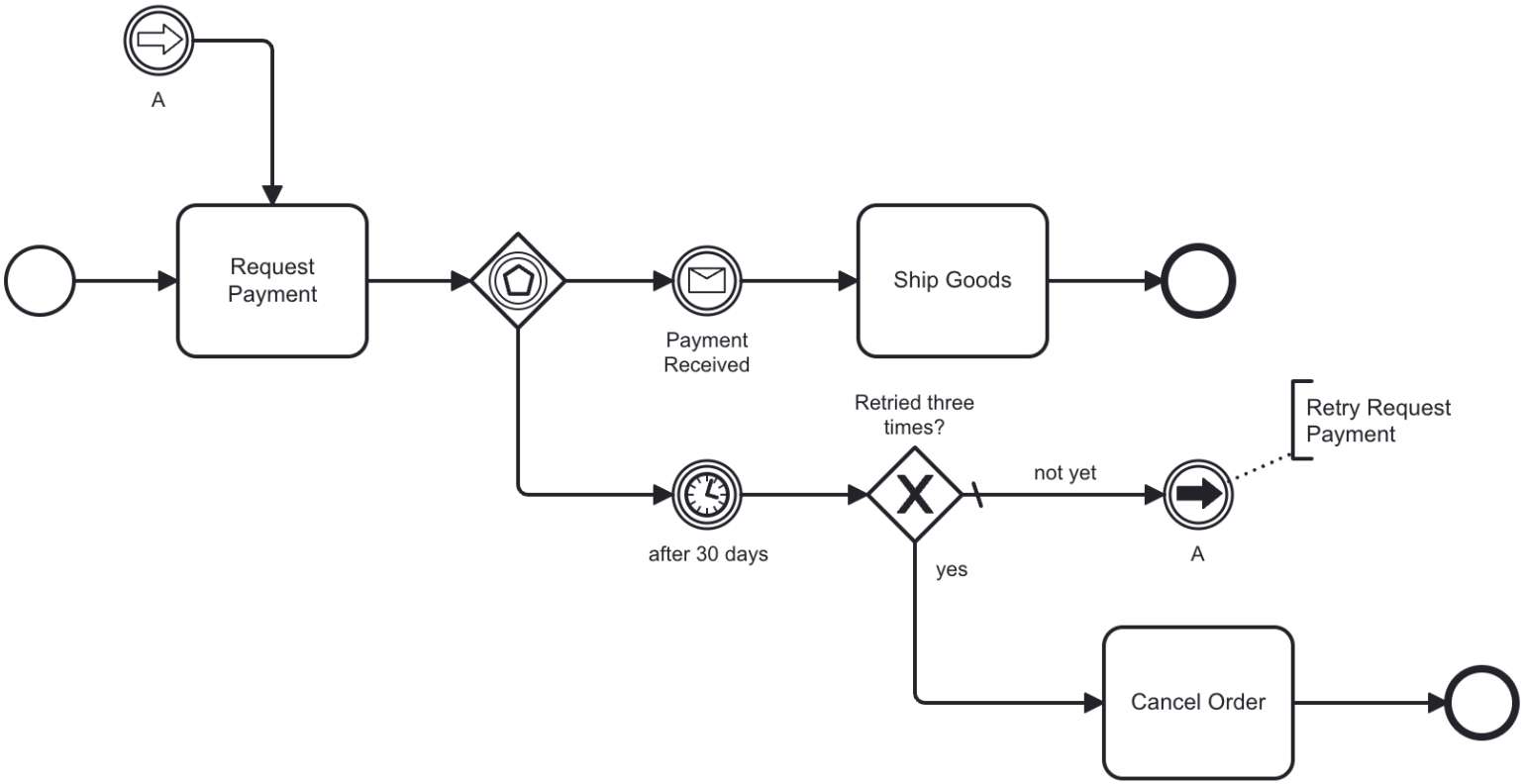
Example


Amount Changed
Handle dispute
Created
Complete
Make partial payment
Create partial payment
Refund partial payment
Cancel
Cancel payments

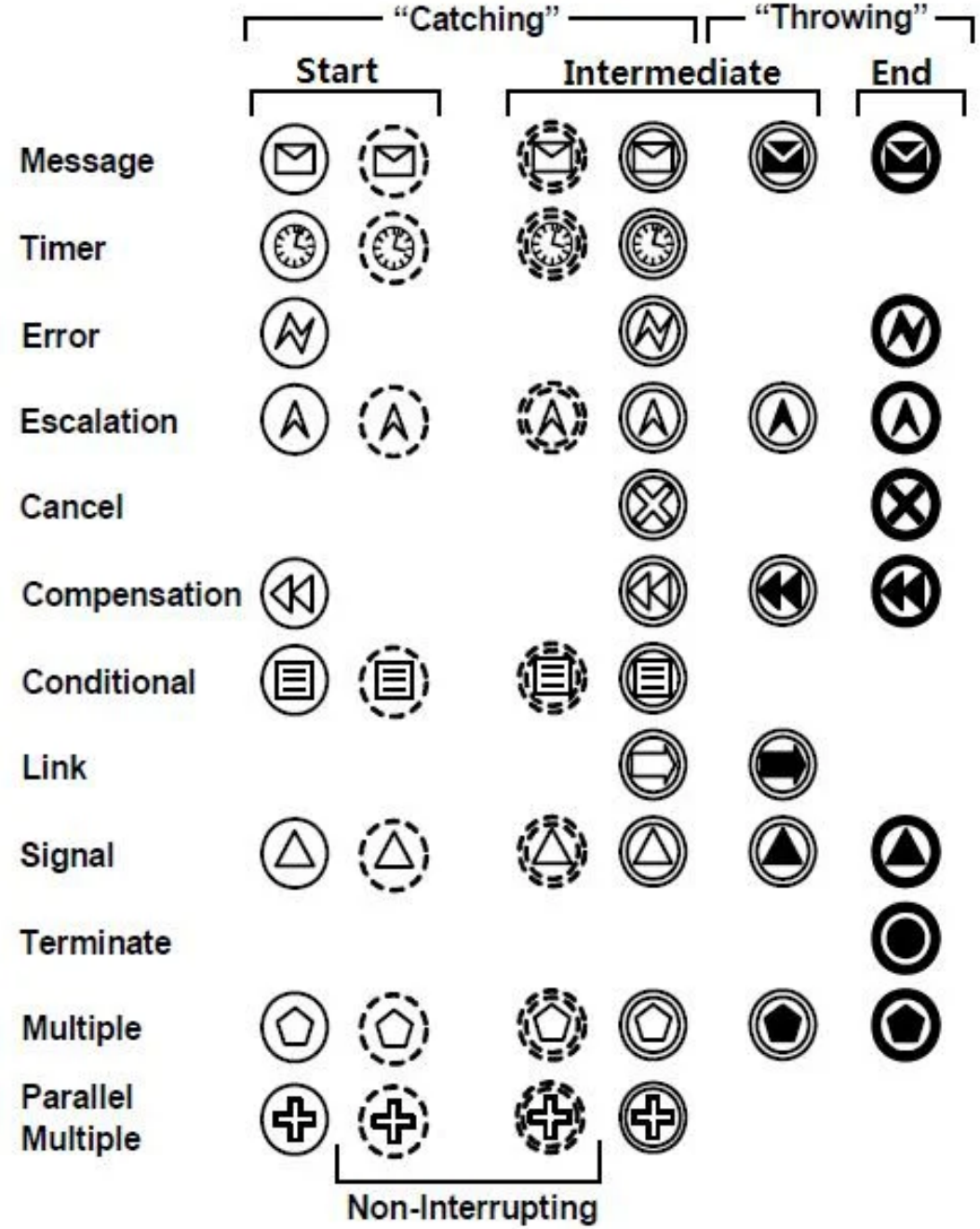
Link events


These events are linked
Task One
A
A
Task Two
Equivalent to Link A
Task One
Task Two

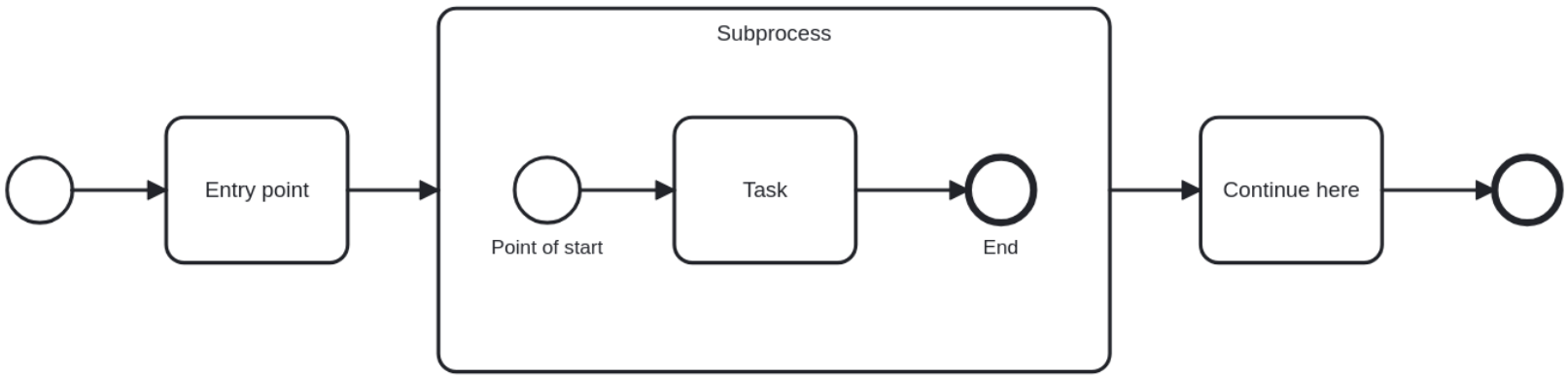
Example


A
Request Payment
Payment Received
Ship Goods
Retried three times?
Retry Request Payment
not yet
after 30 days
A
yes
Cancel Order

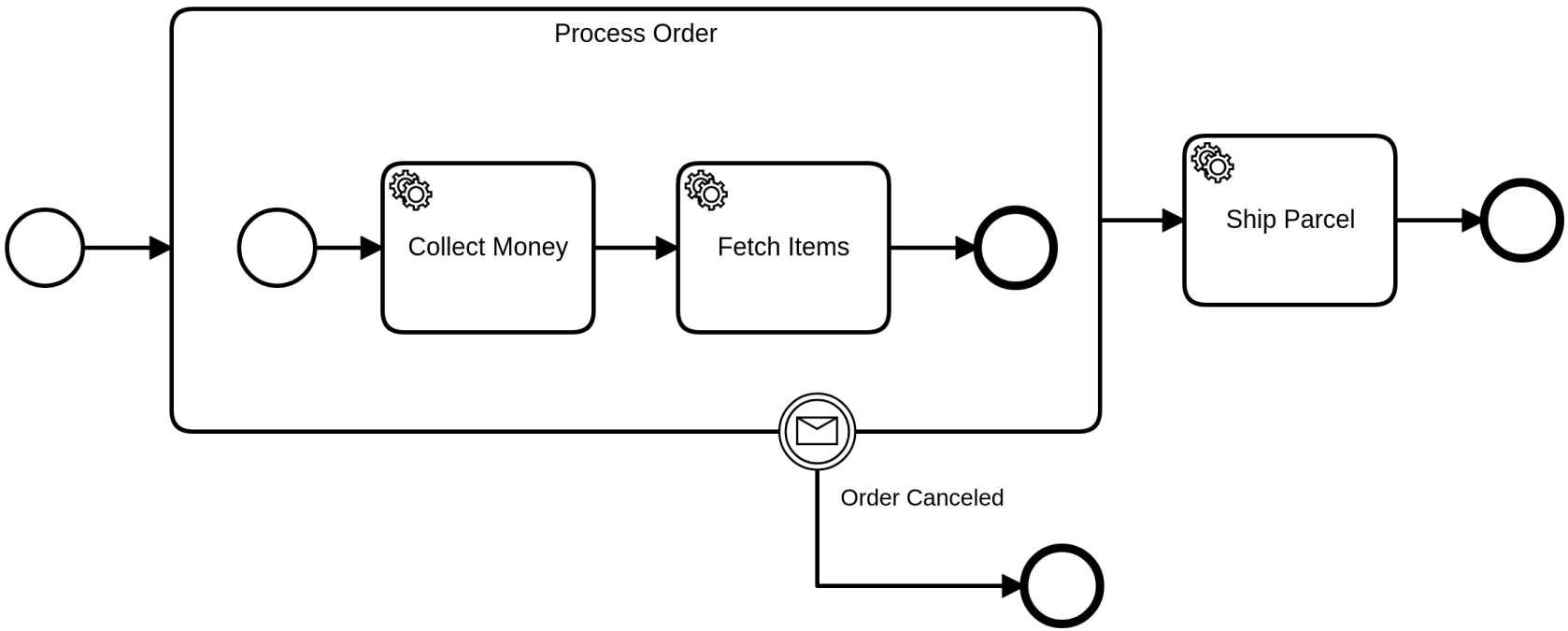
Summary


"Catching"
"Throwing"
Start
Intermediate
End
Message
Timer
Error
Escalation
Cancel
Compensation
Conditional
Link
Signal
Terminate
Multiple
Parallel Multiple
Non-Interrupting

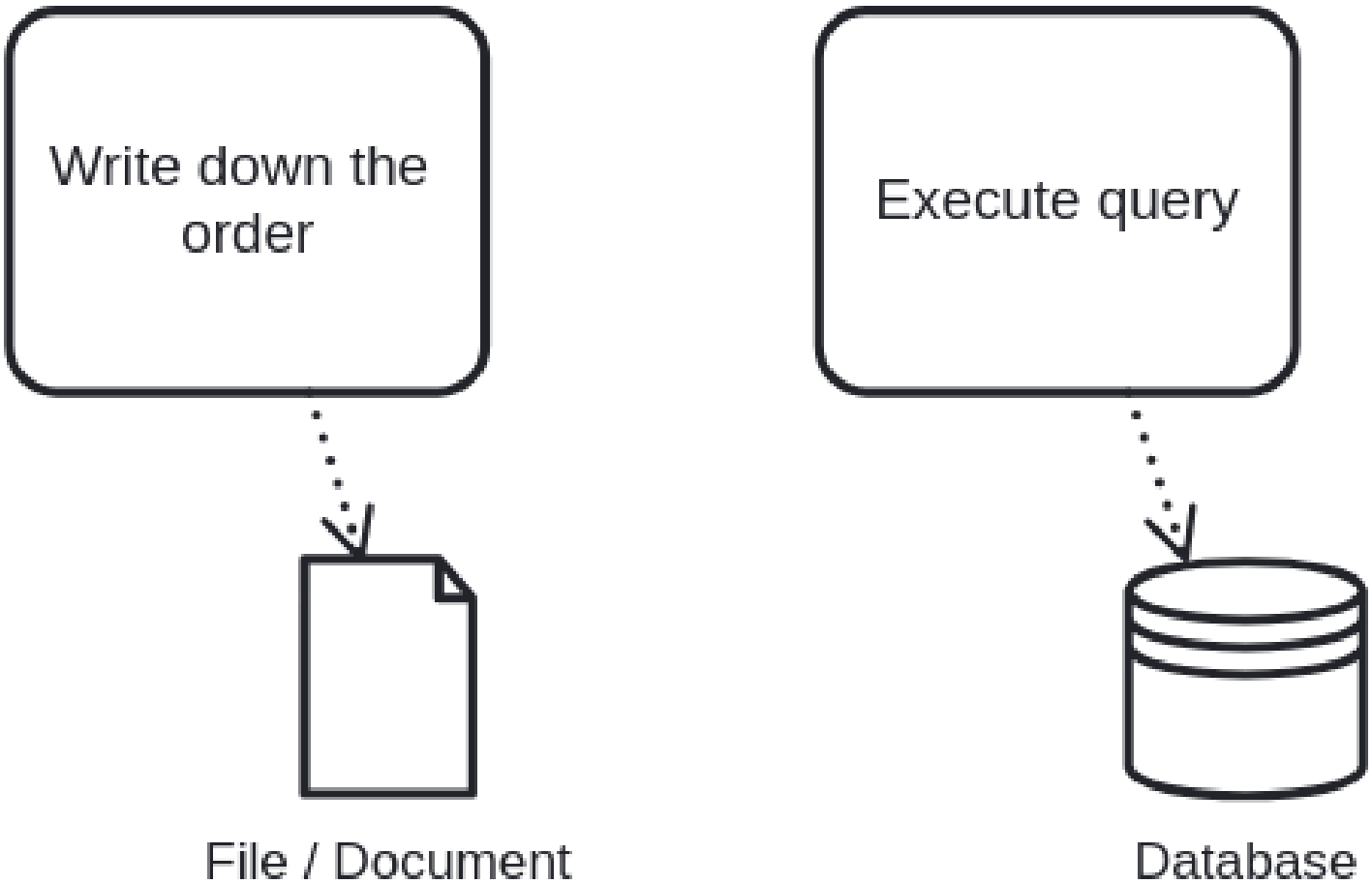
Subprocess


Subprocess
Entry point
Point of start
Task
End
Continue here

Subprocesses can have boundary events



Documenting resources



Exercise: Bank Loan

A customer submits a loan request to a bank. The bank processes the request and evaluates it. During this processing phase, a technical issue may occur that prevents the request from being completed successfully.

If such an error happens, the issue is handled by the bank's IT department. Once the problem has been resolved, the bank retries processing the loan request.

When the request is successfully processed, the bank sends a response to the customer indicating whether the loan has been approved or rejected. The customer then reacts to the outcome. If the loan is approved, the customer receives the money and, after 6 months, may either repay the loan or fail to do so.

If the customer does not repay the loan, a bank employee escalates the situation by notifying their supervisors, who will take care of further actions.

Solution

BPMN Choreography

Description

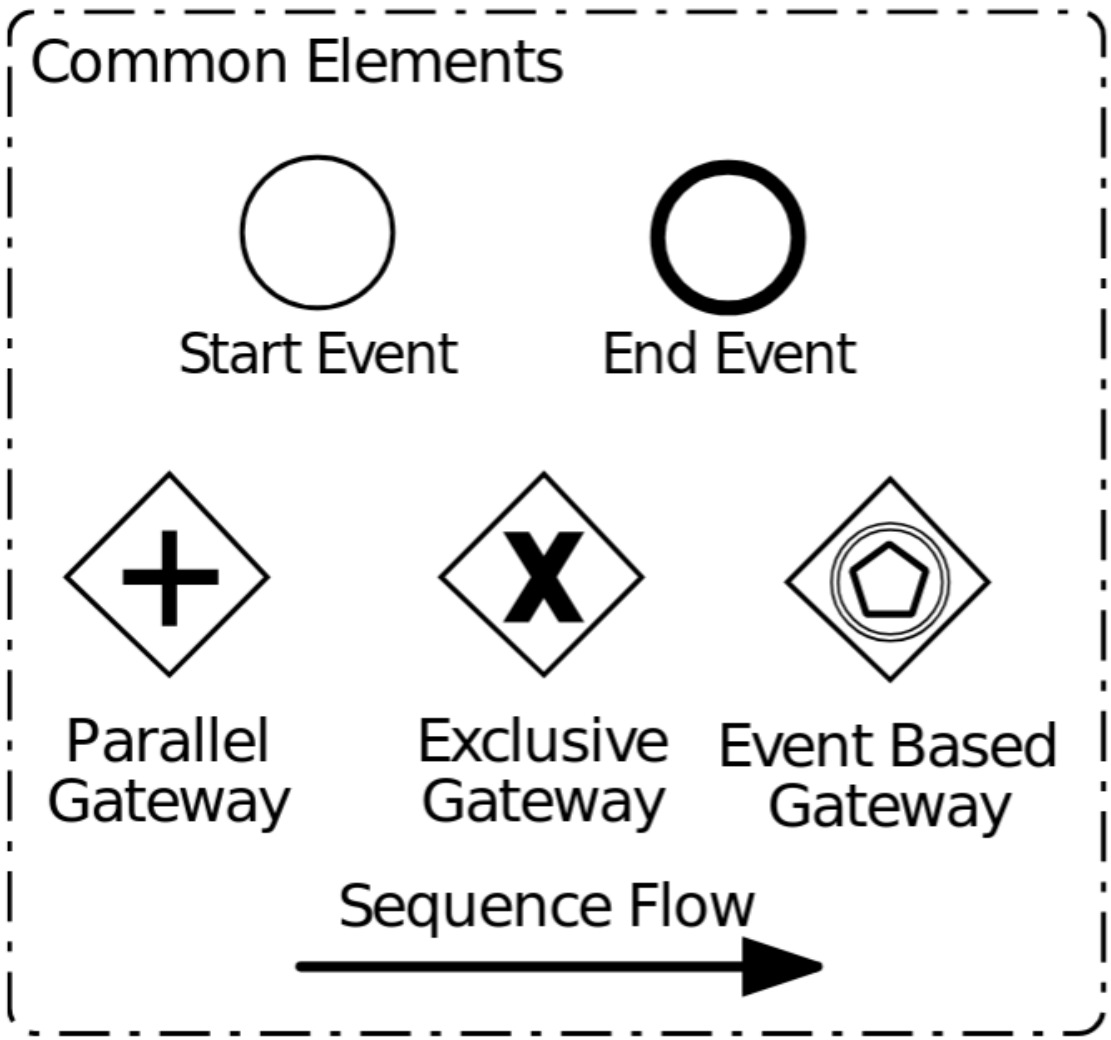
BPMN Choreography:

- Introduced in BPMN version 2.0
- Description of interaction in a Business Process
- Focused on the communication
- (BPMN == orchestration) vs (Chor == distributed)

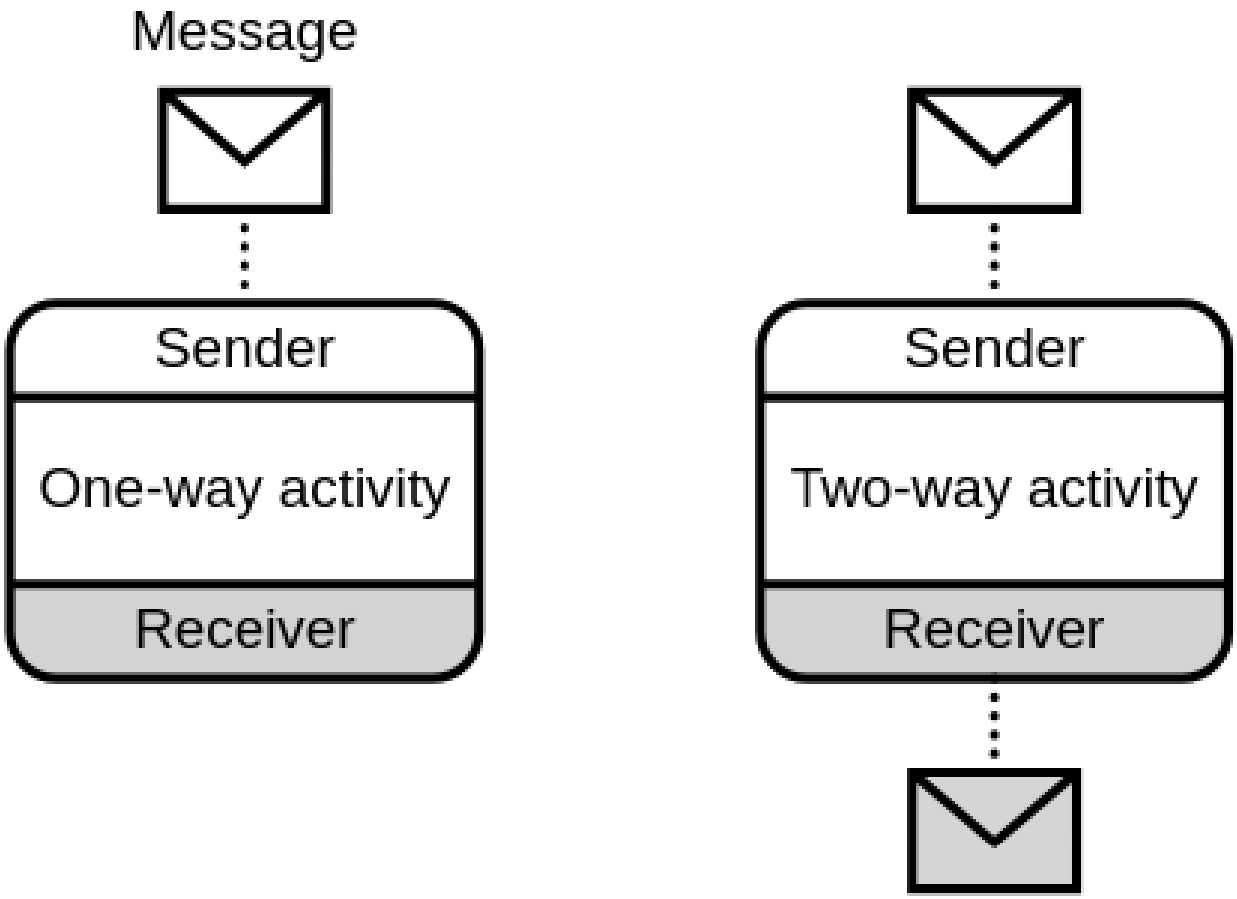
Modeler

https://bpt-lab.org/chor-js-demo/

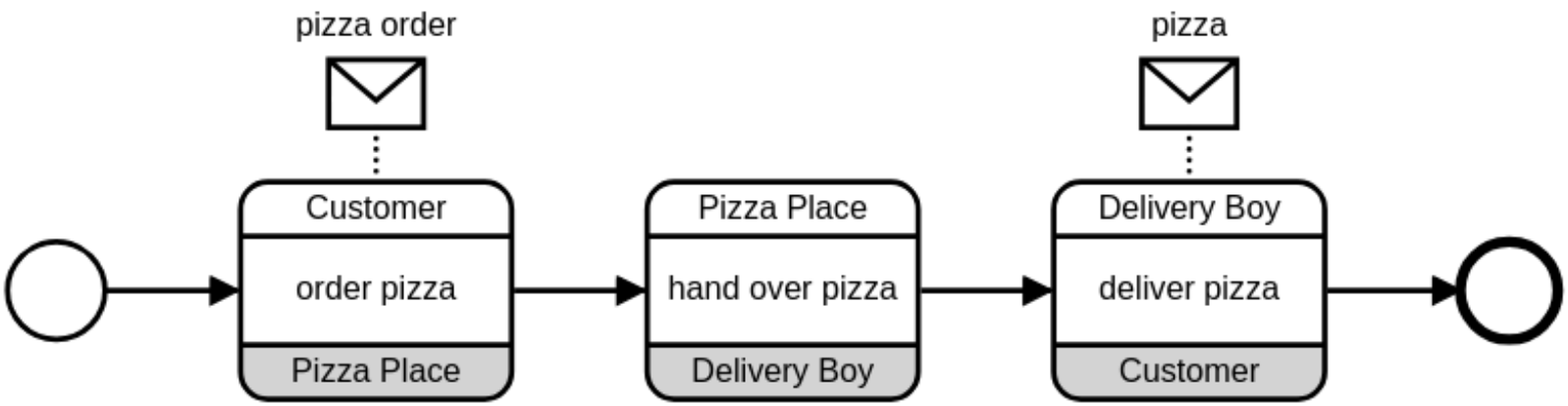
Elements in common with BPMN



Task


Message
Sender
One-way activity
Receiver
Sender
Two-way activity
Receiver

Example


pizza order
pizza
Customer
order pizza
Pizza Place
Pizza Place
hand over pizza
Delivery Boy
Delivery Boy
deliver pizza
Customer

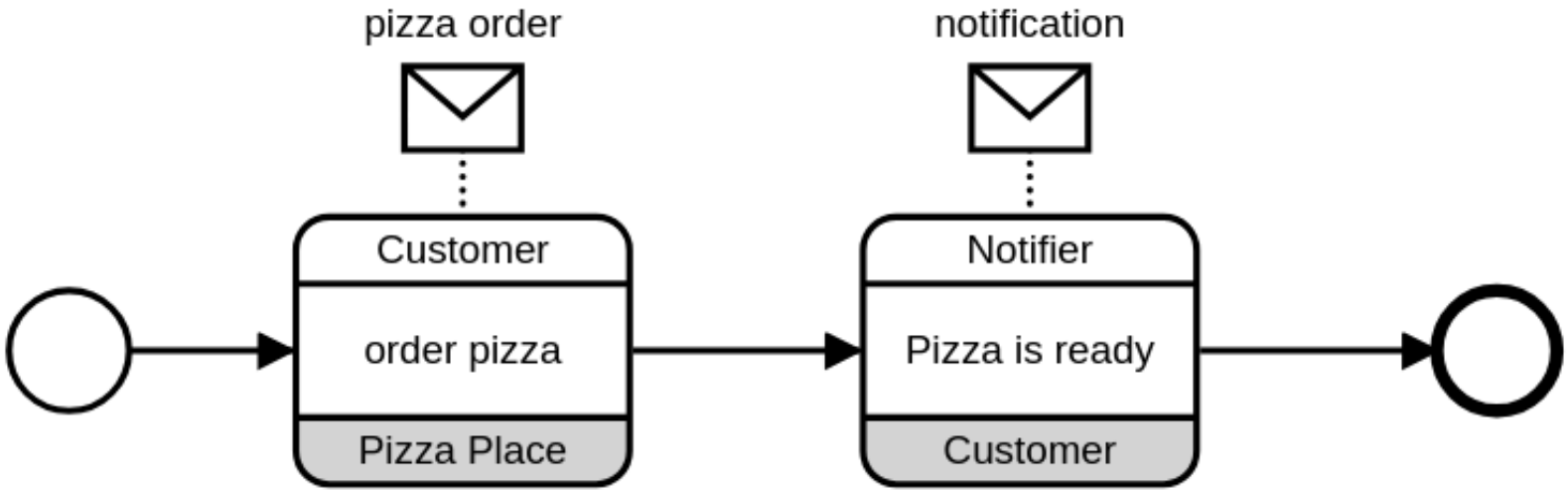
Realizability conditions

Key problem in top-down models for distributed processes:

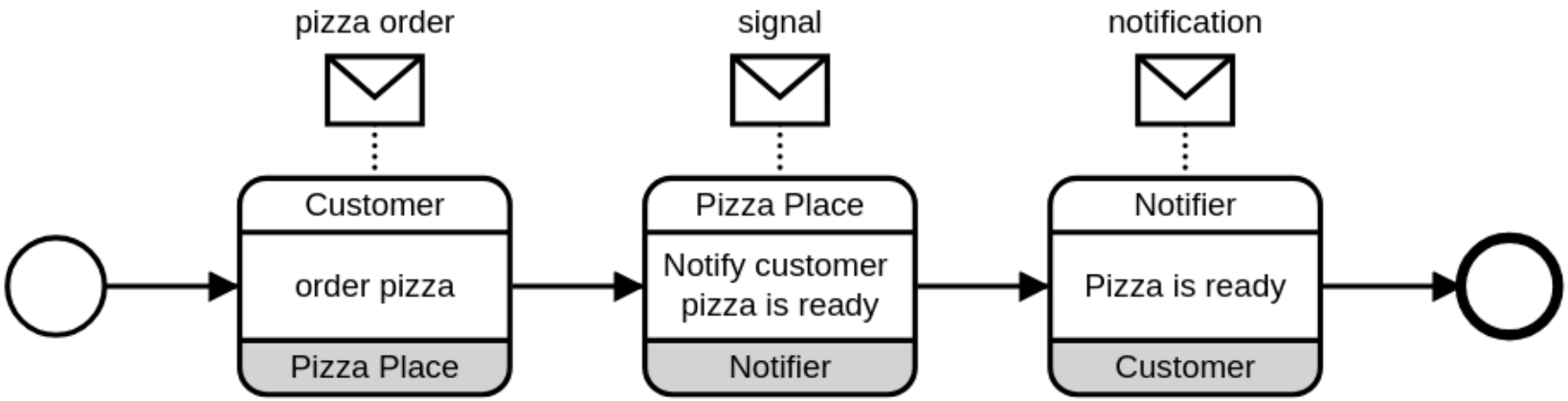
- How does a partecipants know when to send a message?
- Or when to receive a message and not get stuck?

Solution: every task should be initialized by one of the partecipants of the previous communication.

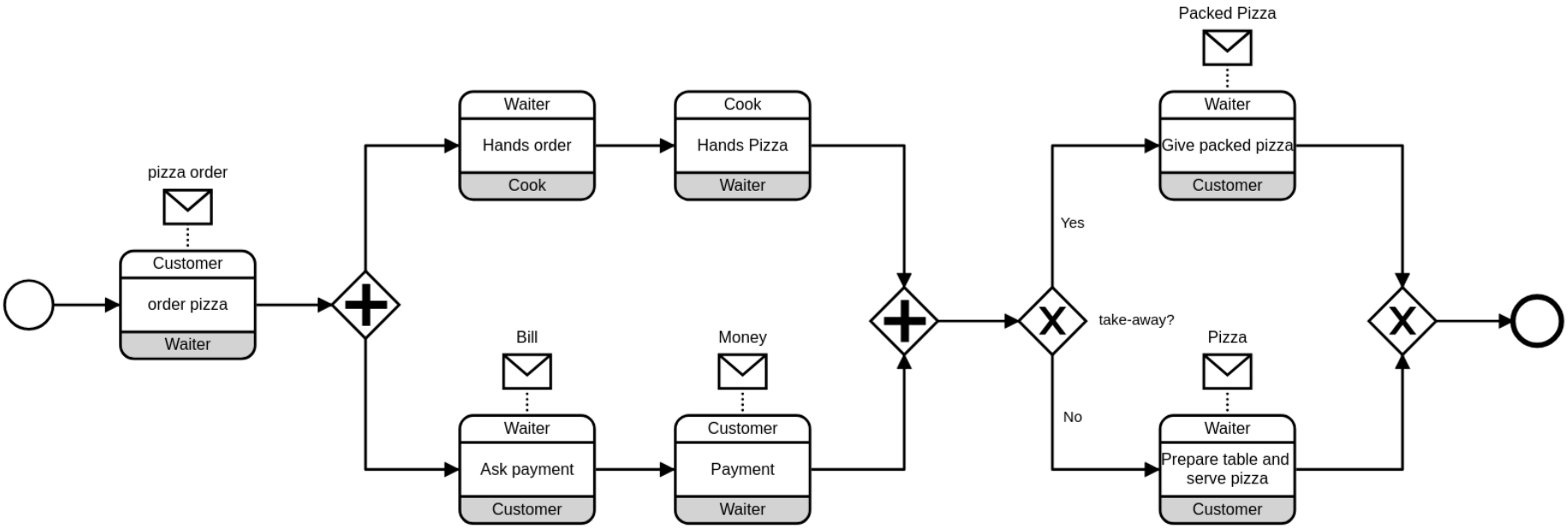
Not-realizable Choreography



Fixed Choreography



How does it work with gateways?



Exercise: Wedding Planner

A WP must coordinates all the bookings for a wedding for a certain date. WP must book the Church and the Restaurant for the same date:

- If both the Church and the Restaurant are available the protocol ends.
- If one of the two is not available, try again with a different date.

Solution

The end! Questions?