

# Different types of BPMN objects



# Object types

- Many classes of BPMN objects have types and additional attributes
  - E.g. Start Events have triggers, Tasks have types etc.
- Sometimes they impact the visualization of the object
- Not mandatory, but often helpful
  - Communication
  - Analysis
  - Automation

# Object types

- We will cover most commonly used types
  - Start events
  - Tasks
  - Sub-processes
  - End events
- There are many more, which are covered in the Advanced BPMN part of the course

# What starts a process

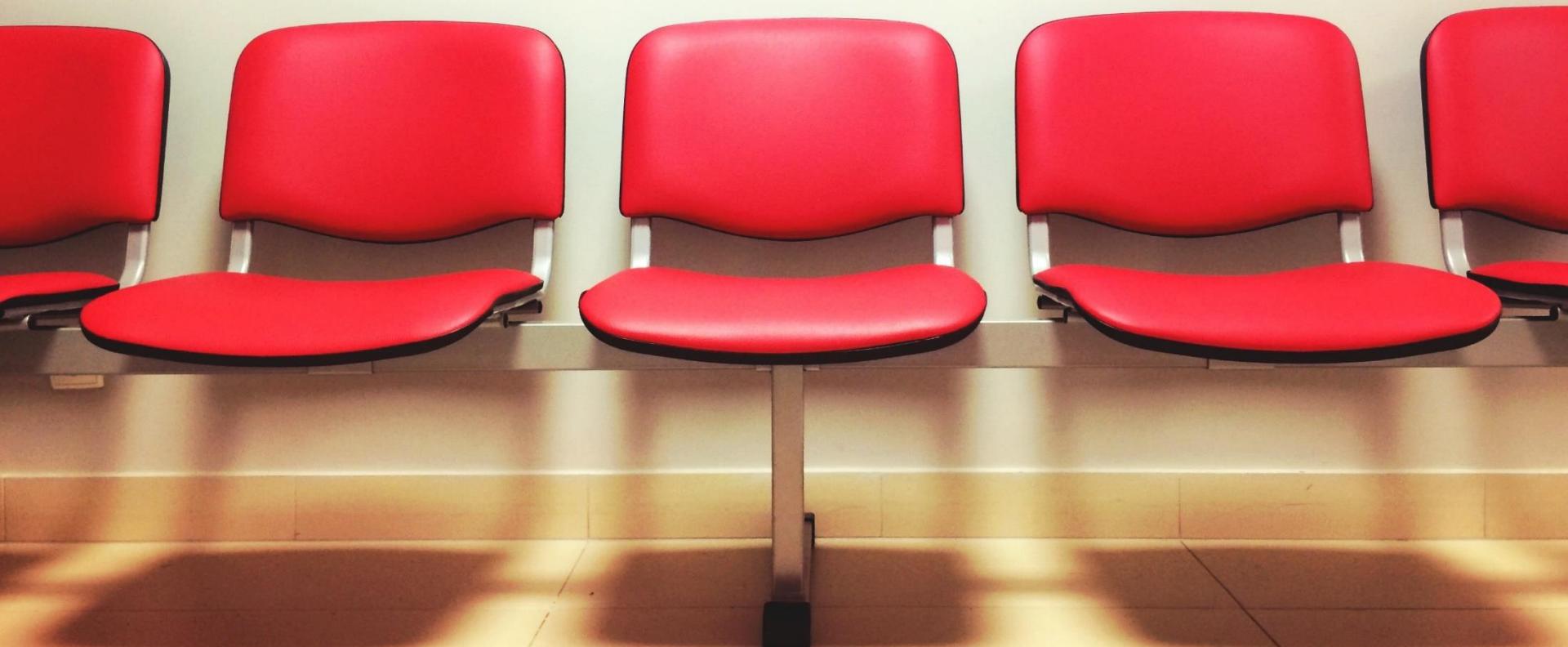


# Object types

- **Start Event**
  - **Start Event** – circle with a single thin border
  - Shows what needs to happen, so that a process can start. This is NOT what you do, but what triggers process to start
- **Commonly used types (triggers)**
  - Message
  - Timer
  - Conditional



# Start events are always passive



# Do names matter?

*What's in a name?*

*That which we call*

*a rose*

*By any other name  
would smell as*

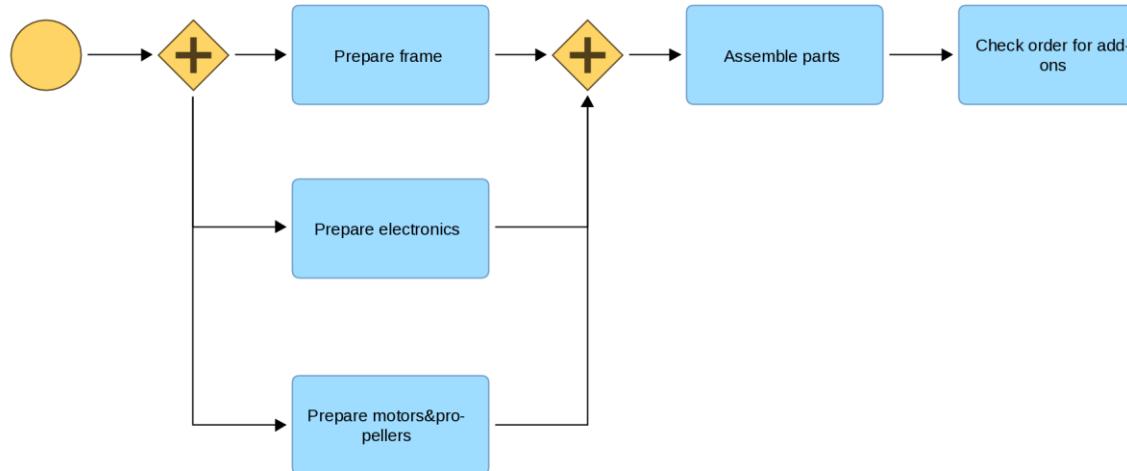
*sweet*

ROMEO AND JULIET, WILLIAM  
SHAKESPEARE



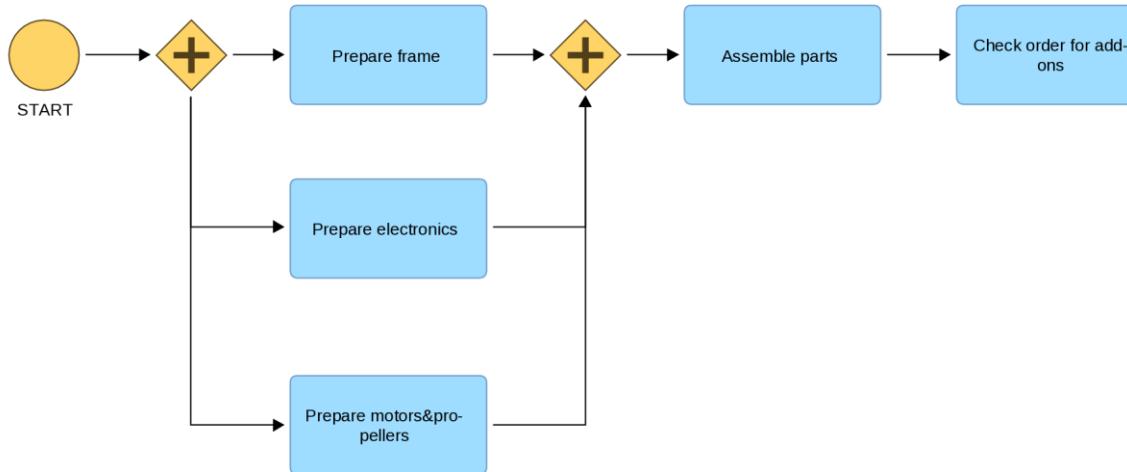
# Names

- Names are not mandatory
- But...



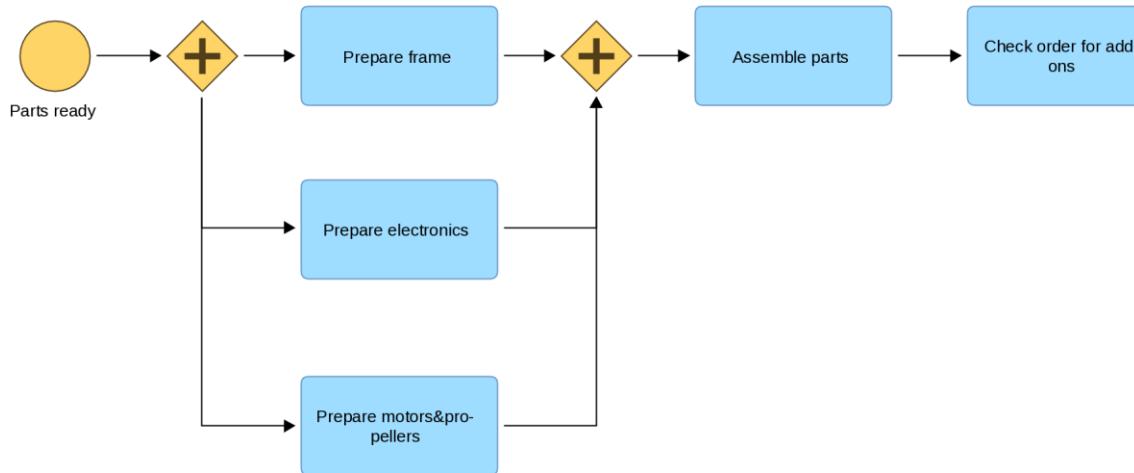
# Names

- Names are not mandatory
- But...



# Names

- Names are not mandatory
- But a meaningful name helps a lot

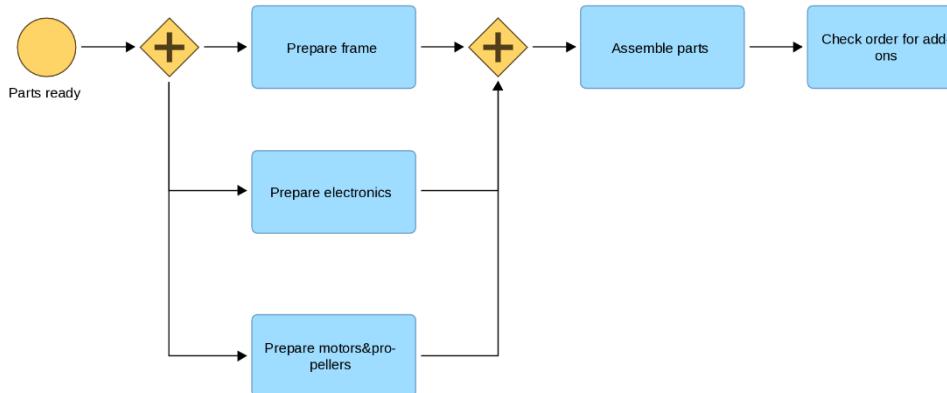


# Start events – common types



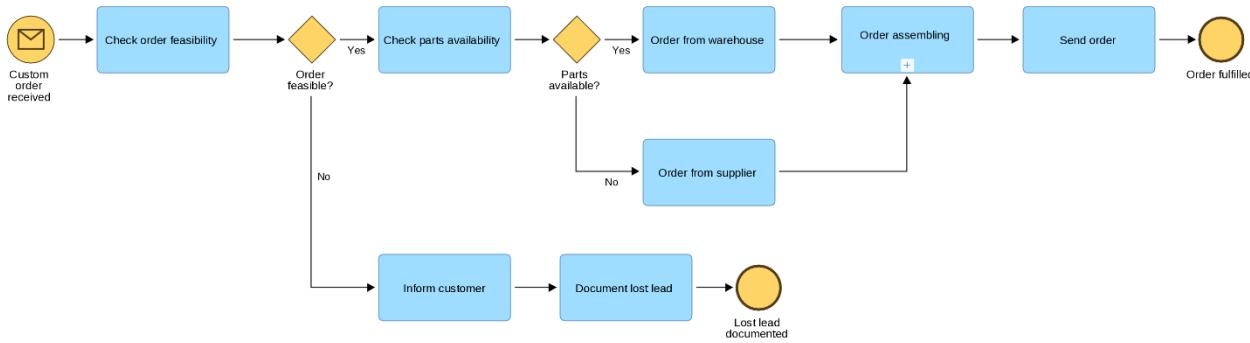
# Plain start event

- Start event with no trigger
  - Default option
  - Process starts e.g. when employee decides it is needed
  - Necessary in some cases (e.g. for Subprocesses)



# Message start event

- Triggered by a Message
  - Process is triggered by e-mail, phone call, talk, ...
  - Directed communication (specific sender and recipient) received **from the outside of the process** (e.g. from a customer, partner company, ...)
  - Envelope marker
  - Naming convention: “MESSAGE NAME received” e.g. Order received, Request for quote received, ...



# Timer start event

- Triggered by time
  - Process starting periodically (e.g. every Friday at 6pm or first working day of a quarter)
  - Clock marker
  - Naming convention: Time/date reference (NOT technical)



# Conditional start event

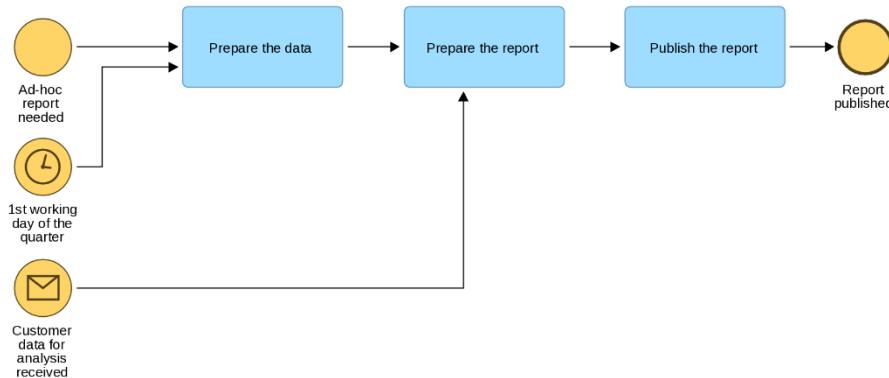
- Triggered by condition

- Process starts when certain condition (external to the process) becomes true
- Additional tokens starting new instances are not created unless the condition state changes from true to false and again (no flooding)
- Lined paper marker
- Naming convention: name of the condition in normal language e.g. Stock level below minimum, S&P 500 drops over 5%, Oven temperature over 475°F



# Many starts?

- How to show many possible start events?
  - Usually it is a good idea to have one start event
  - Sometimes you may need to show that many things can start your process (apart from one plain start event)

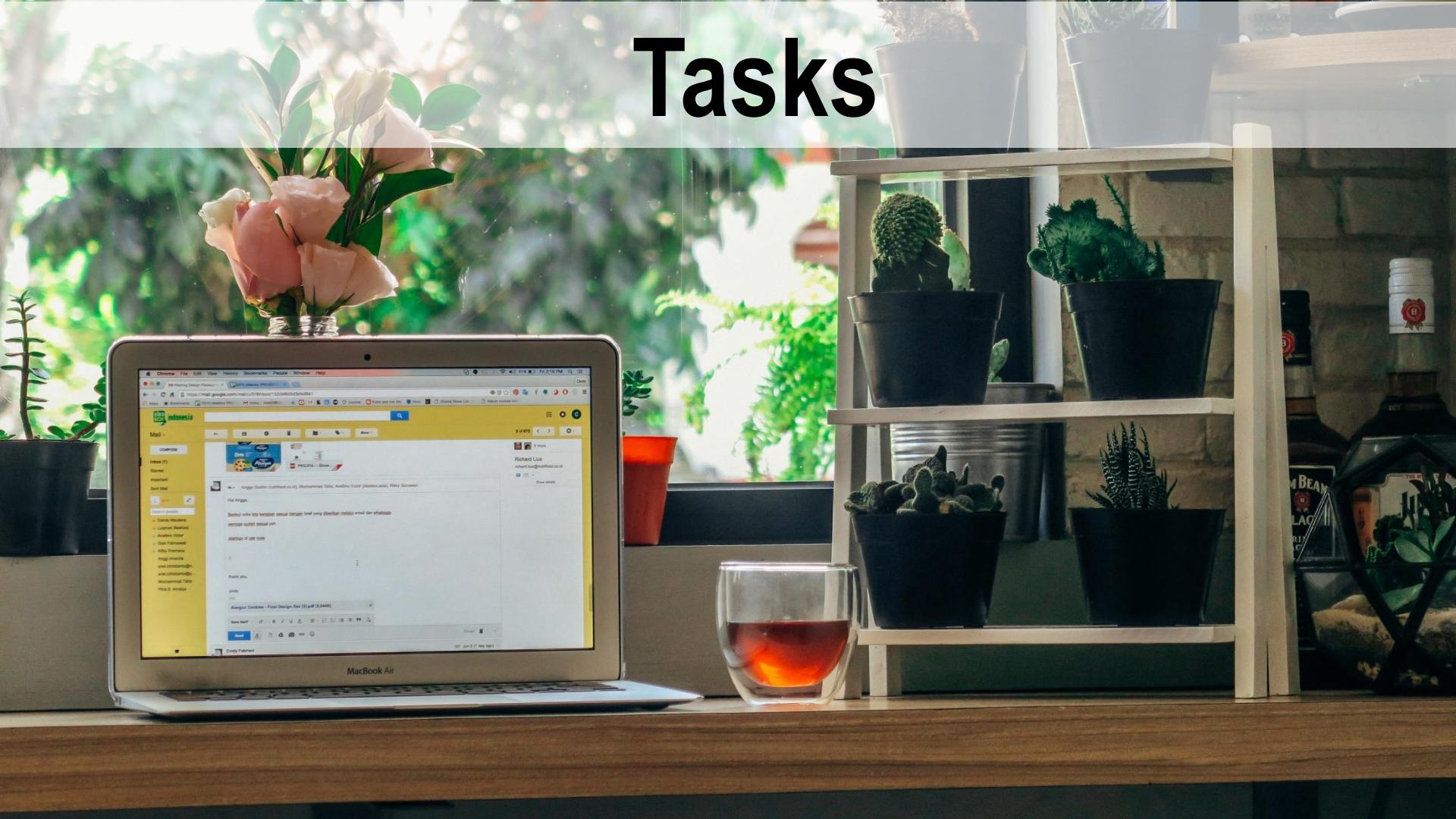


- It is also possible to use a Multiple start event (pentagon marker) or Parallel Multiple start event (light plus marker)

# What needs to be done

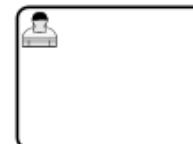


# Tasks



# Task types

- **Task**
  - Rounded rectangle
  - Simple process step (“atomic”)
- **Commonly used types**
  - User (employee performs it in an IT system)
  - Service (IT system does it automatically)
  - Manual (employee performs it outside the IT system)



# Manual task

- **Performed by employee**
  - “A Manual Task is a Task that is expected to be performed **without the aid of any business process execution engine or any application**. An example of this could be a telephone technician installing a telephone at a customer location.” (BPMN Spec)
  - Someone needs to do it, but without coordination of IT system
  - Hand marker



Check order feasibility



Inform customer

# Manual task



# Manual task



# User task

- Performed by employee in IT system
  - “A User Task is a typical “workflow” Task where a human performer performs the Task with the assistance of a software application and is scheduled through a task list manager of some sort.” (BPMN Spec)
  - IT system assigns task to user, user does it (usually using some form), IT system assigns next task and so on
  - Human figure/avatar marker



Order from sup-  
plier



Document lost  
lead

Create a filter +

Created



Add Comment +

My Tasks (3)



Filter Tasks

3



### Order from supplier

Custom drone order handling

Demo Demo

Created 6 hours ago

50

### Assign Reviewer

Review Invoice

Demo Demo

Created 6 hours ago

50

Invoice Amount:

Invoice Number:

10.99

▼ PSACE-5342

### Assign Reviewer

Review Invoice

Demo Demo

Created 6 hours ago

50

Invoice Amount:

Invoice Number:

10.99

▼ PSACE-5342

## Order from supplier

Custom drone order handling

Set follow-up date

Set due date

Add groups

Demo Demo

Form

History

Diagram

Description

Provide ID of the part that needs to be ordered

Provide a name of the part

Number of parts to be ordered

Select a supplier for the part

Specify a desired delivery date

Was this part previously ordered?

Save

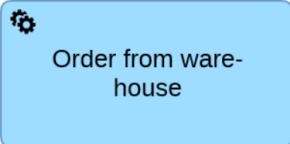
Complete

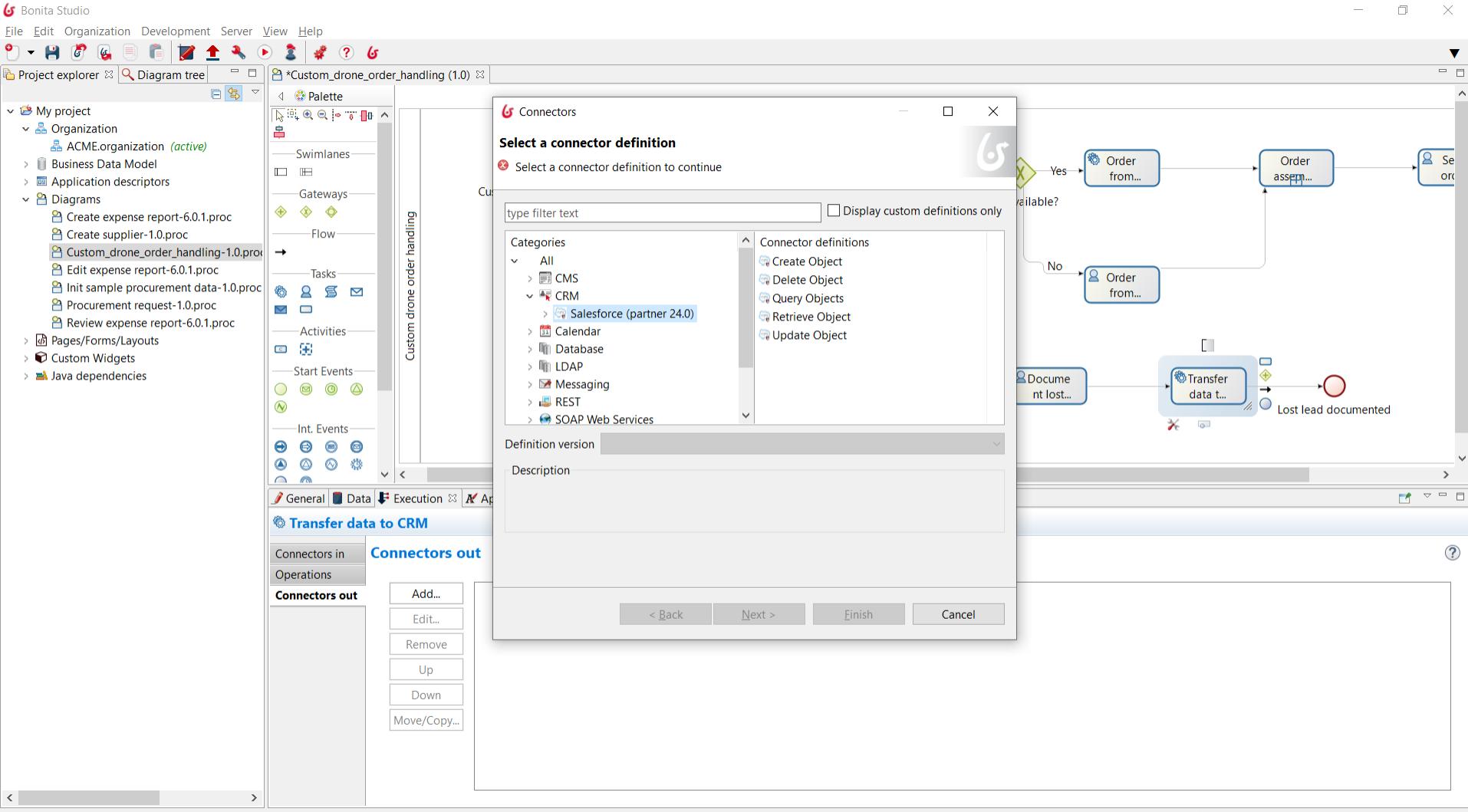
# User task



# Service task

- Performed automatically by an IT system
  - “A Service Task is a Task that uses some sort of service, which could be a Web service or an automated application.” (BPMN Spec)
  - Process automation engine calls some external system (e.g. ERP, CRM) or other application – no involvement from user
  - Cogwheels marker





# Service task

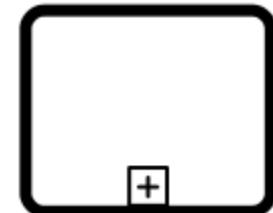
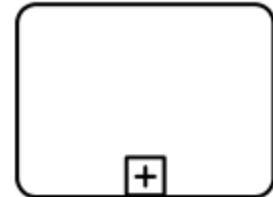


# Sub-Processes



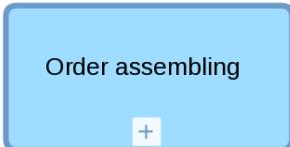
# Sub-Process types

- **Sub-Process**
  - Rounded rectangle with + marker (unless it's expanded)
  - Complex process step
  - Possibly calls some more detailed diagram
- **Commonly used types**
  - By default Sub-Process hides complexity
  - Lower level diagram is a “child”
  - Call Activity
  - We reuse some standard procedure e.g. Procurement



# Call Activity

- **Calling some standard reusable process**
  - This external process can be done outside the lane (different team)
  - Usually we need some standard for providing inputs and handling outputs (think about Procurement in your company)
  - Thick border



# Who ya gonna call?



# How does it end



# Object types

- **End Event**
  - Circle with a single thick border
  - Shows the result of a process e.g. Order fulfilled
  - Consumes token(s)
- **Commonly used types**
  - Message
  - Terminate



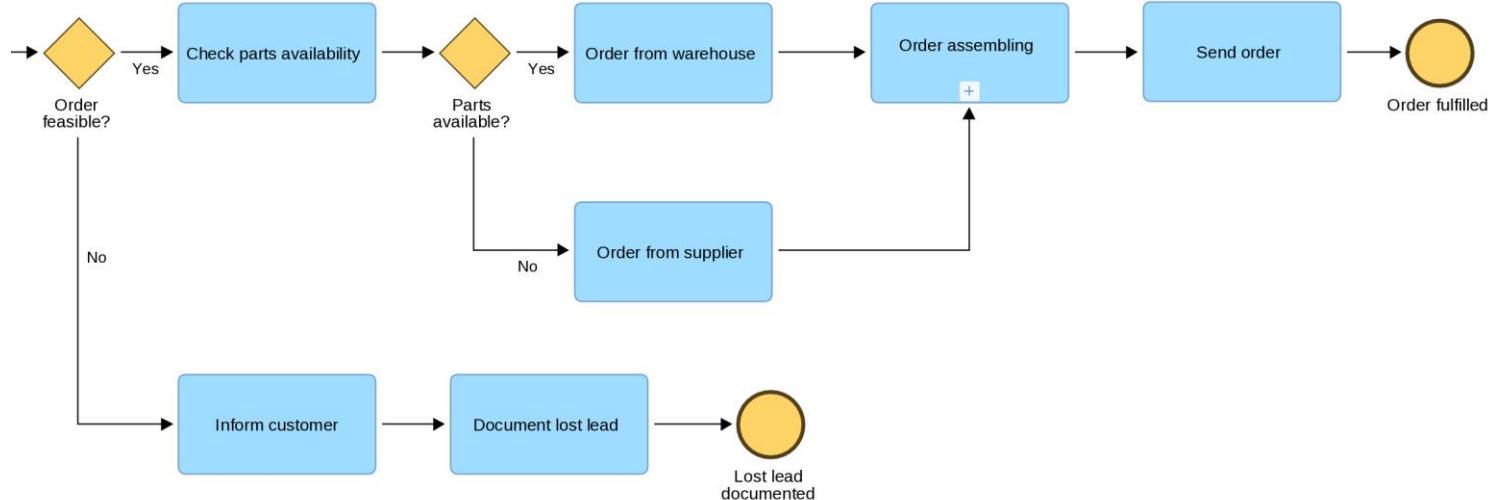
# End events are always active



# Plain end event

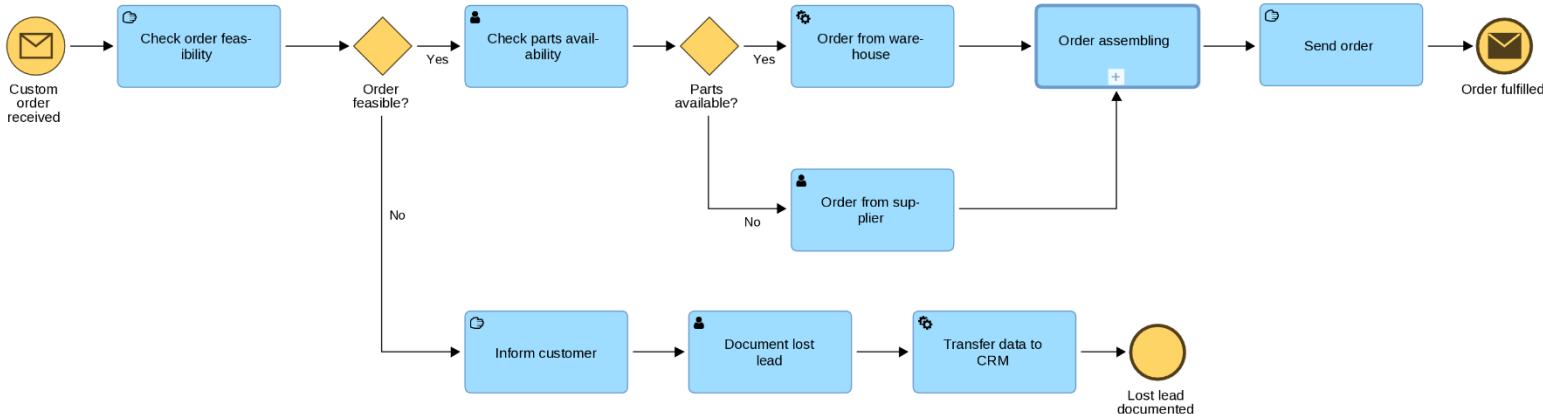
- End event showing result of a process

- Default option
- Many end events are common (end of happy path vs exceptions)
- When token reaches end event it is consumed



# Message end event

- Reaching this end event sends a message
  - When token reaches end event it is consumed and a message is sent (to the outside of a process)
  - This implies automated process
  - Envelope marker with dark fill

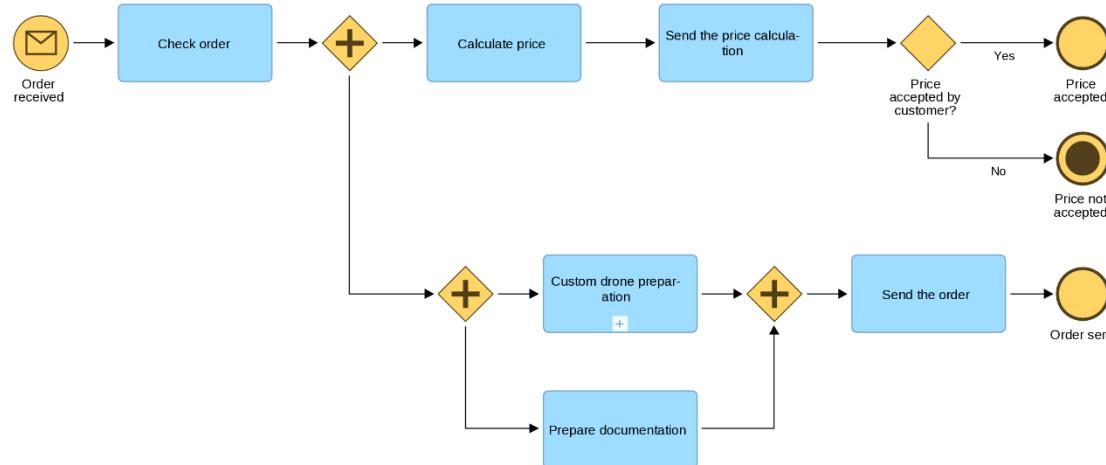


# “Swan song”



# Terminate end event

- Reaching this end event finishes all Activities in a process
  - All other tokens from this process instance are “killed”
  - Dark dot marker



# Hasta la vista, baby



T-800 by Hannaford / Flickr (CC BY-SA 2.0)