Process Implementation

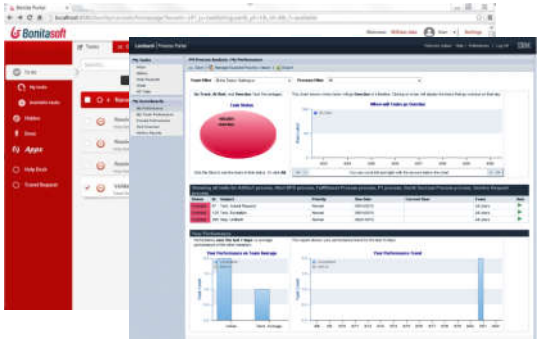
From Conceptual to Executable BPMN Process Models
A Step-by-Step Method

Spring 2021 - MAJU

Nauman H. Ansari

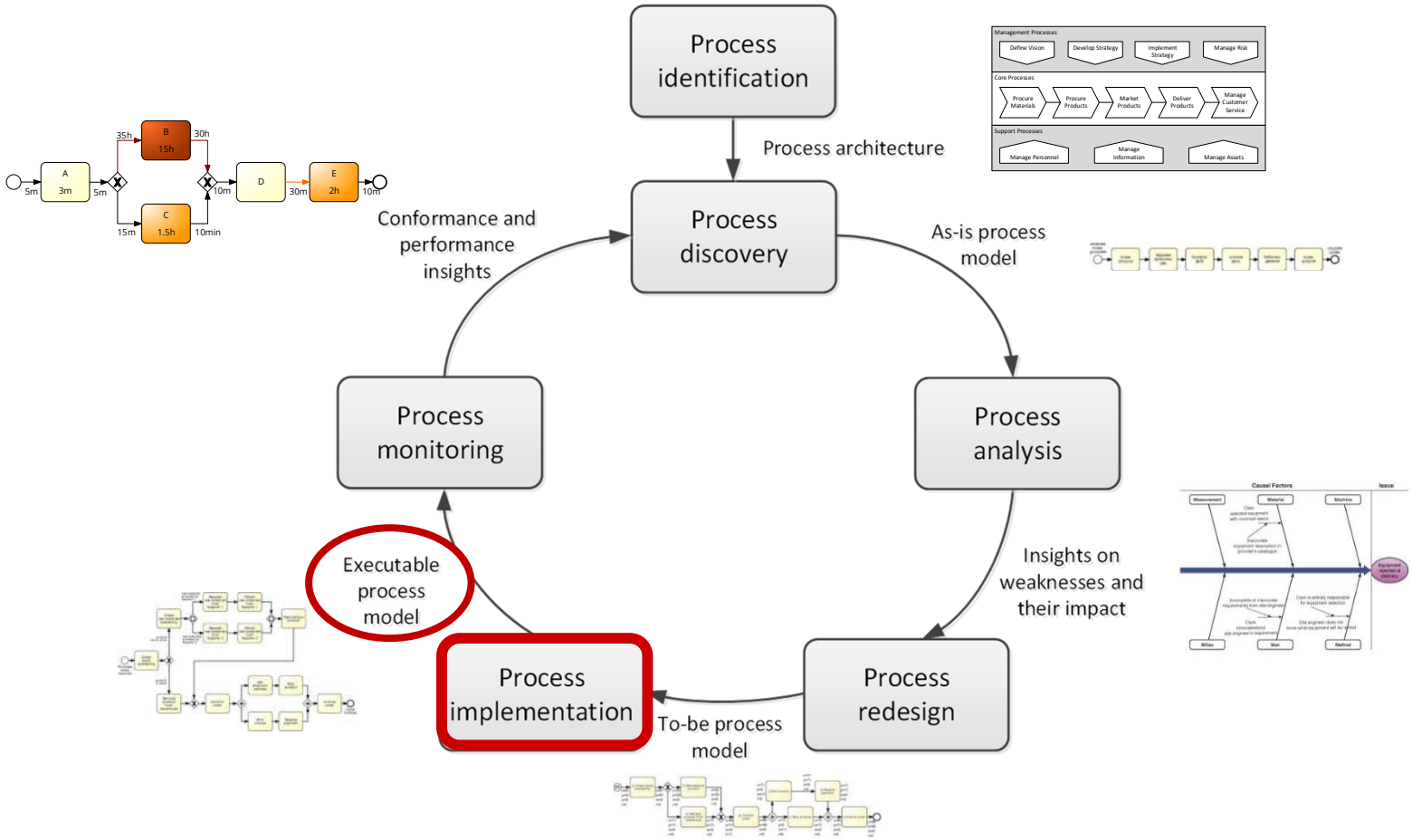
What's this about?

Conceptual process model

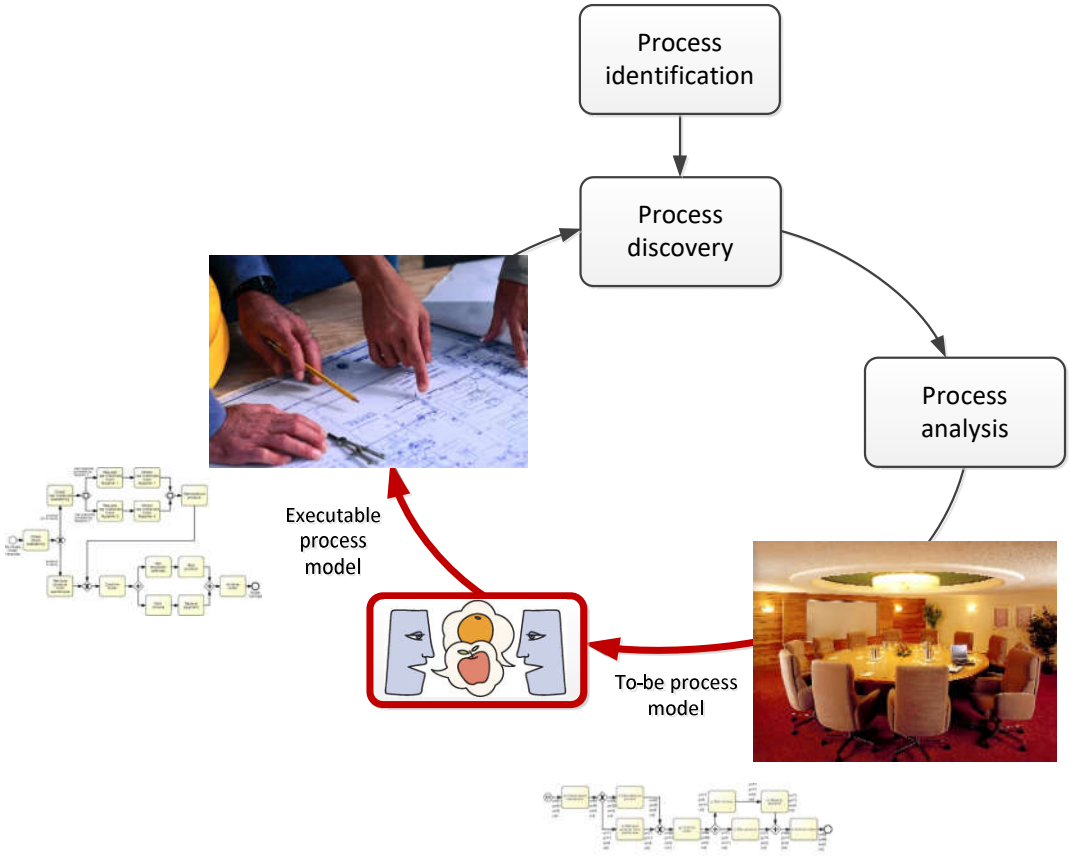


Executable process model

Process Implementation in the BPM Lifecycle



The well-known gap…



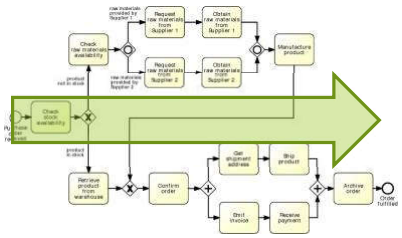
The result: two sides of the story

Conceptual “to-be” process models

- are made by domain experts
- provide a basis for communication amongst relevant stakeholders
- must be understandable
- must be intuitive and may leave room for interpretation
- contain purely a relevant set of process information

Executable process models

- are made by IT experts
- provide input to a process enactment system - BPMS
- must be machine readable
- must be unambiguous and should not contain any uncertainties
- contain further details that are only relevant to implementation

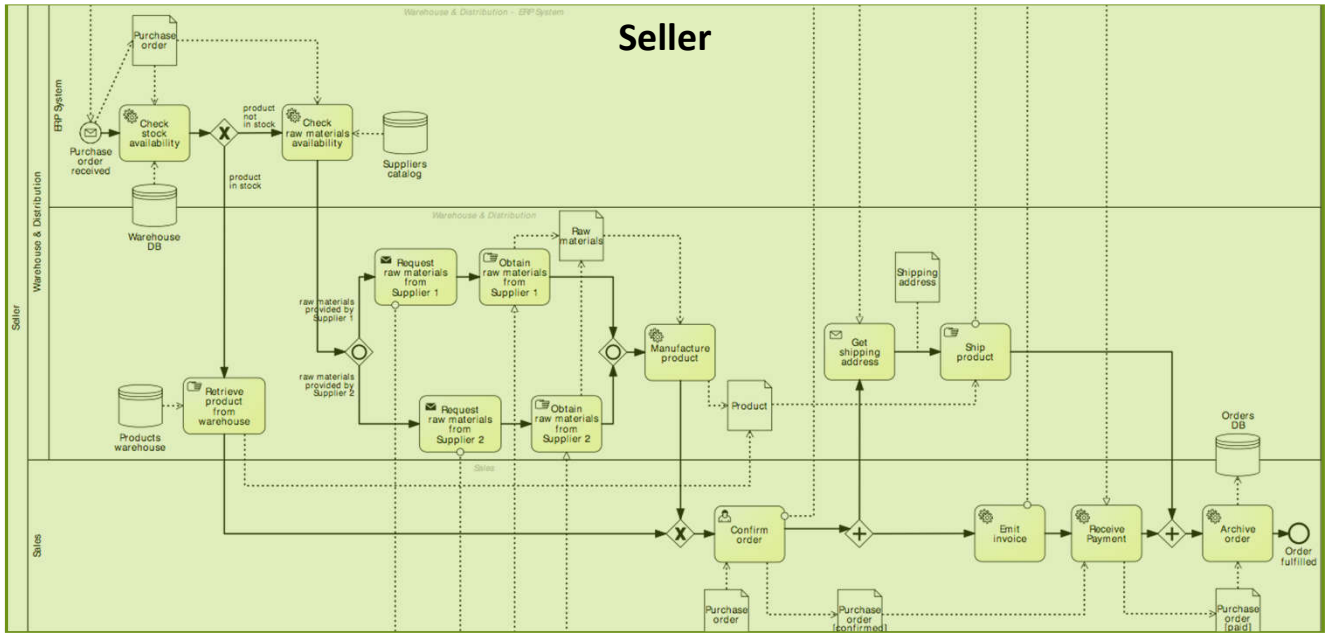


“to-be executed” process model

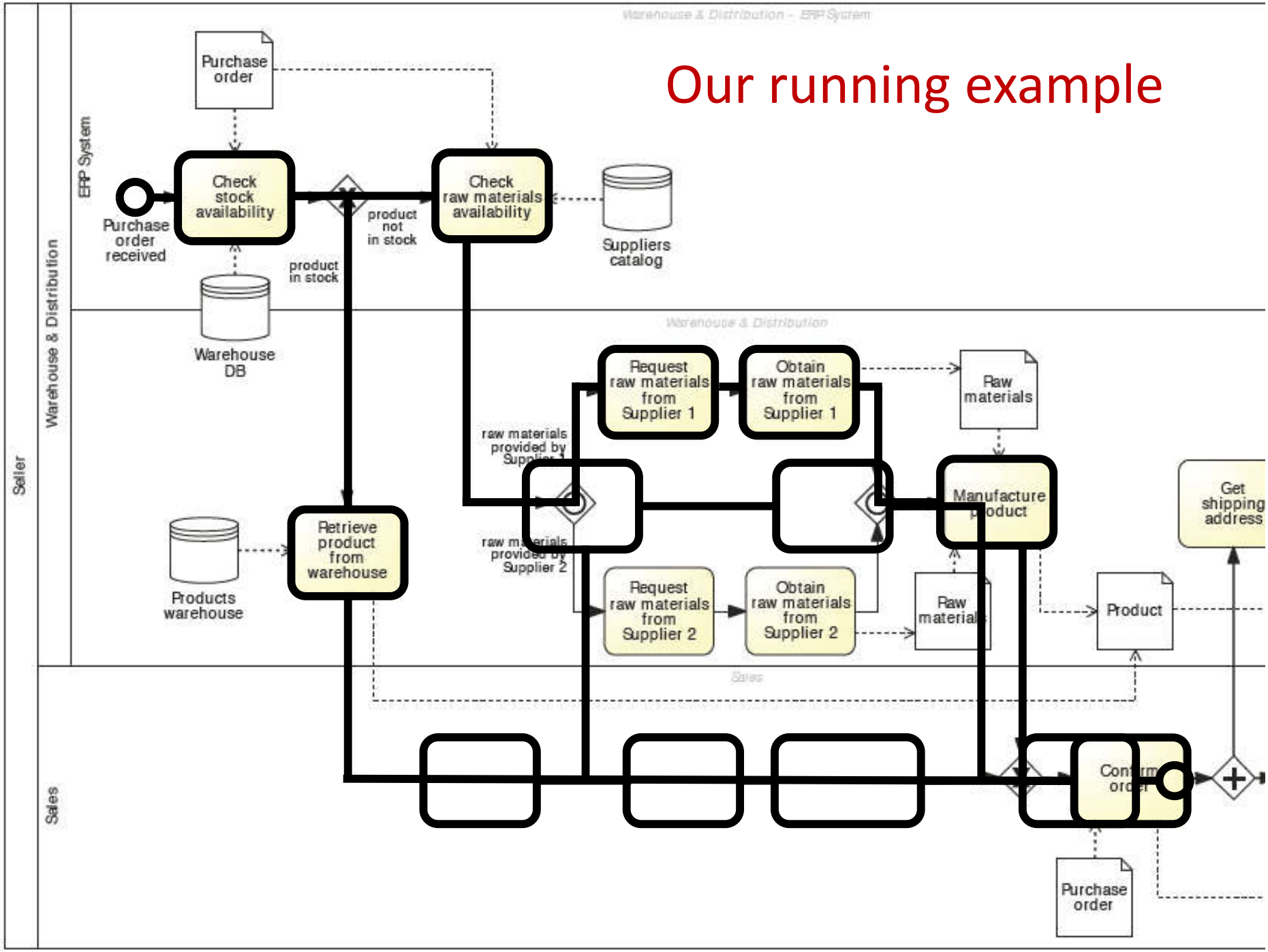
Bridging the gap: one task at a time

- This chapter deals with turning conceptual models into executable models
- Executable models can be used by a process-aware information system to coordinate a business process
- We propose a systematic method for carrying out this transformation, which consists of five steps:
  1. Identify the automation boundaries,
  2. Review manual tasks,
  3. Complete the process model,
  4. Bring the process model to an adequate level of granularity, and
  5. Specify execution properties.
- By following this method a conceptual model incrementally becomes less abstract and more IT-oriented
- As part of this method, two standards complementary to BPMN are used:
  - the Case Management Model and Notation (CMMN), and
  - the Decision Model and Notation (DMN)

Our running example



Our running example



1. Identify the automation boundaries

Principle: not all processes can be automated.

Based on this principle, establish which parts of a process can be coordinated by the BPMS and which parts cannot

-> Start by identifying each task's type:

1

Automated tasks

2

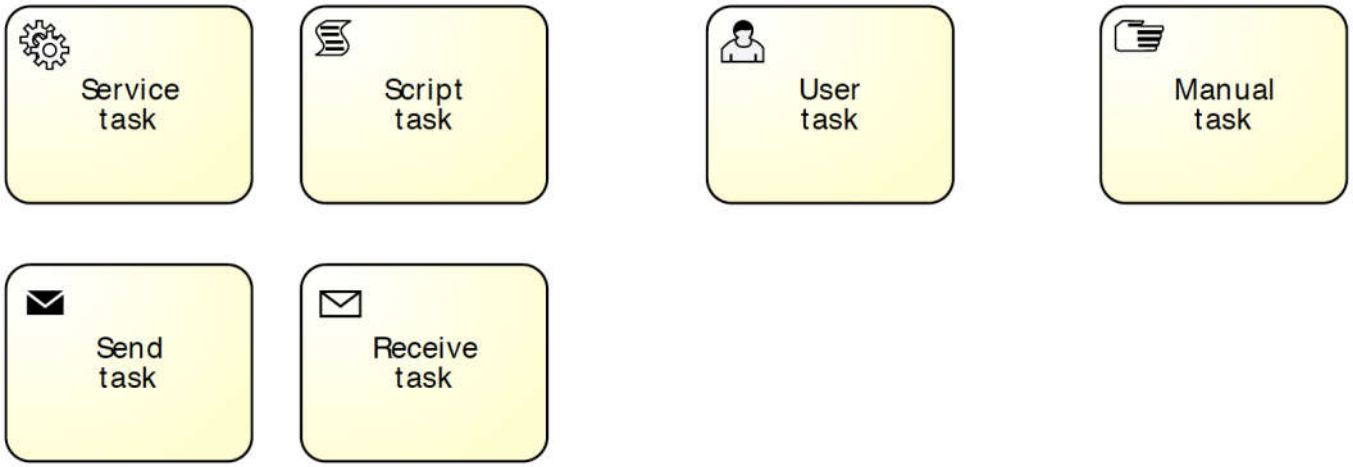
User tasks

3

Manual tasks

1. Automated: are performed by the BPMS itself or by an external service,
2. Manual: are performed by process participants without the aid of any software,
3. User: are performed by a participant with the assistance of the worklist handler of the BPMS or an external task list manager

BPMN Notation: specify task markers



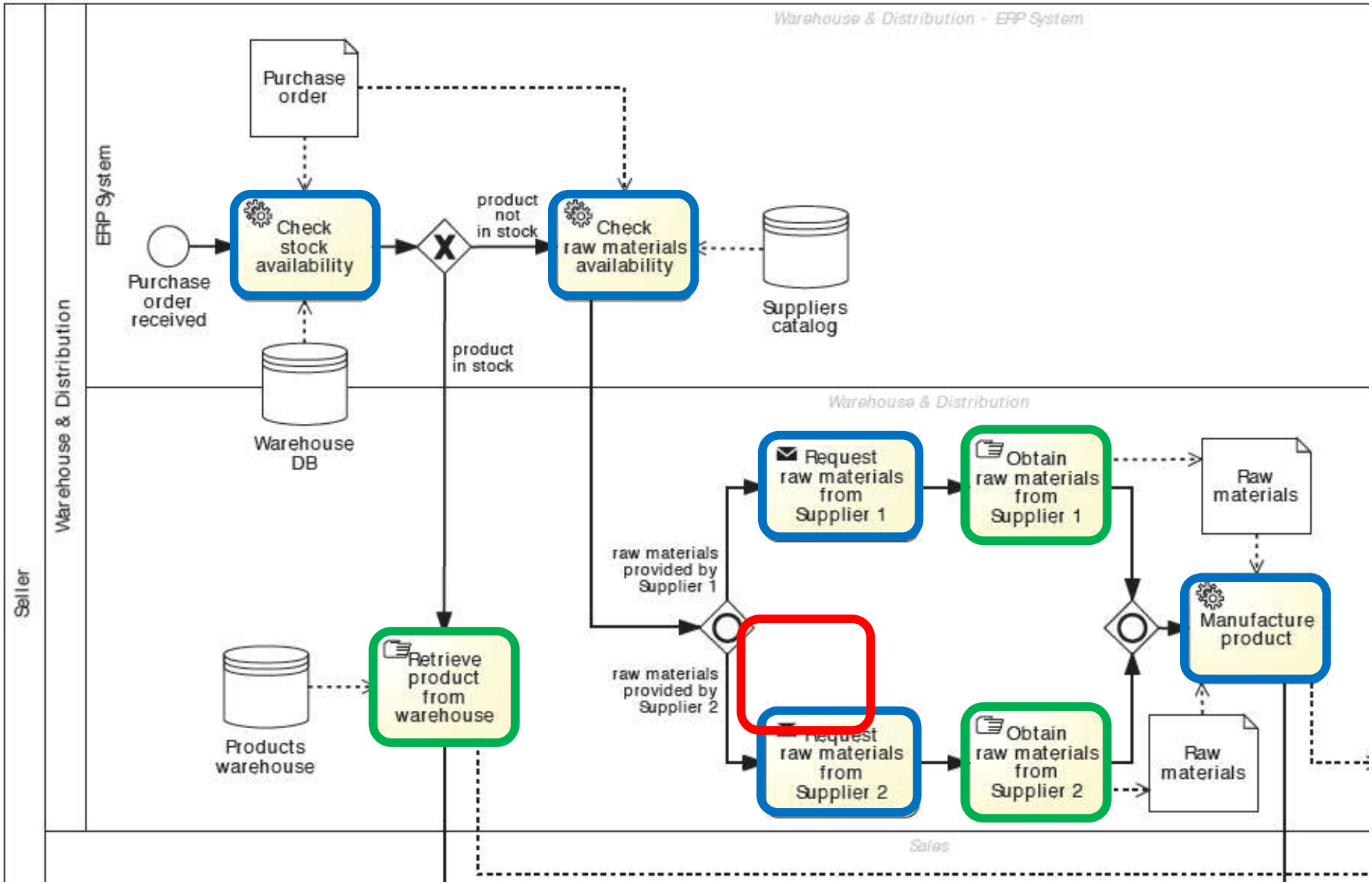
Automated tasks

User task

Manual task

In our example…

automated
user
manual



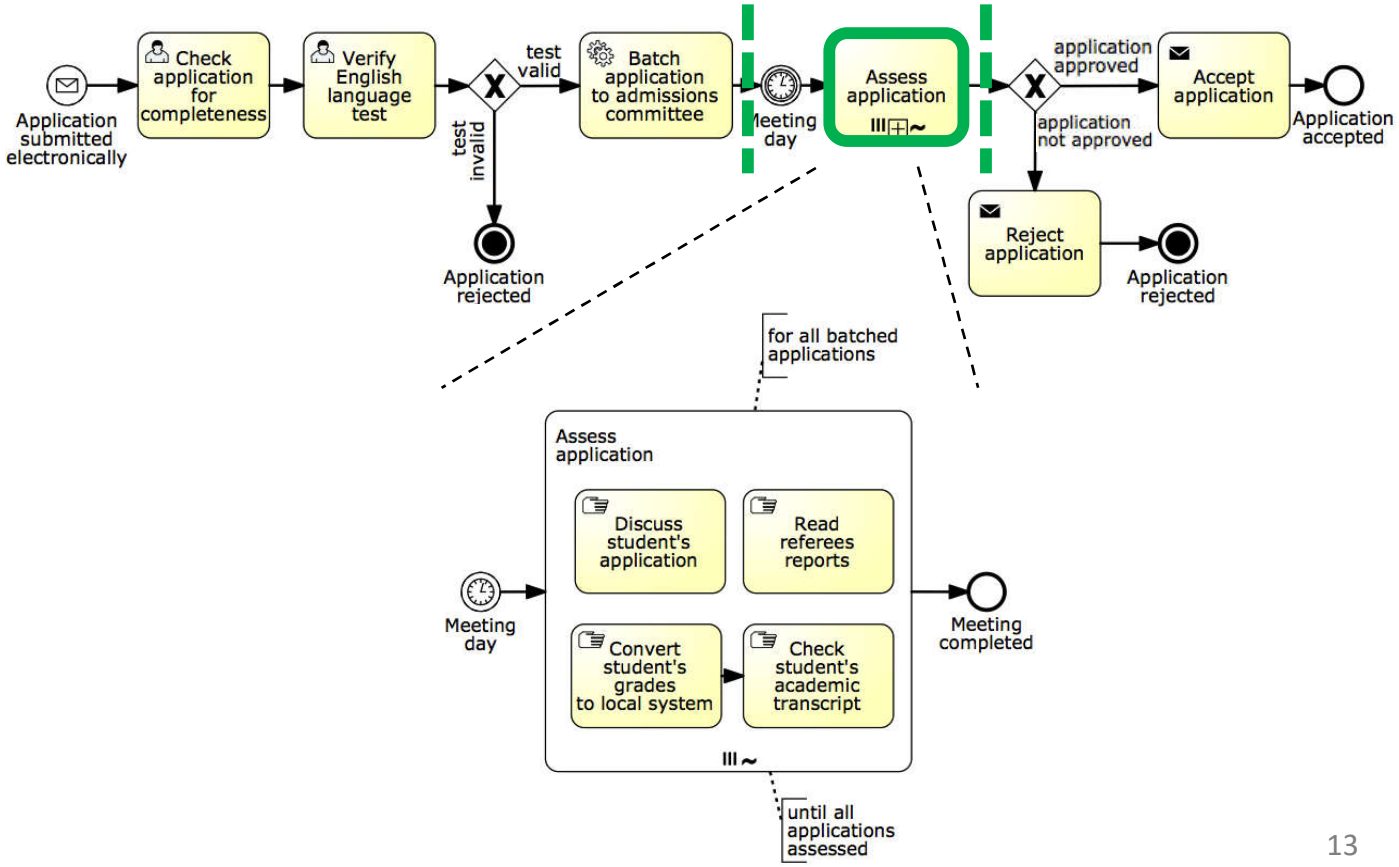
2. Review manual tasks

- Goal: check whether the manual tasks can be linked with the BPMS
- Guiding principle: *if the task cannot be seen by the BPMS, it does not exist*

- Two ways of linking a manual task to a BPMS:
  1. **Implement as User Task**
  2. **Implement as Automated Task**

- Note: There are cases in which it is not convenient to link manual tasks to a BPMS.

Alternative: isolate manual tasks



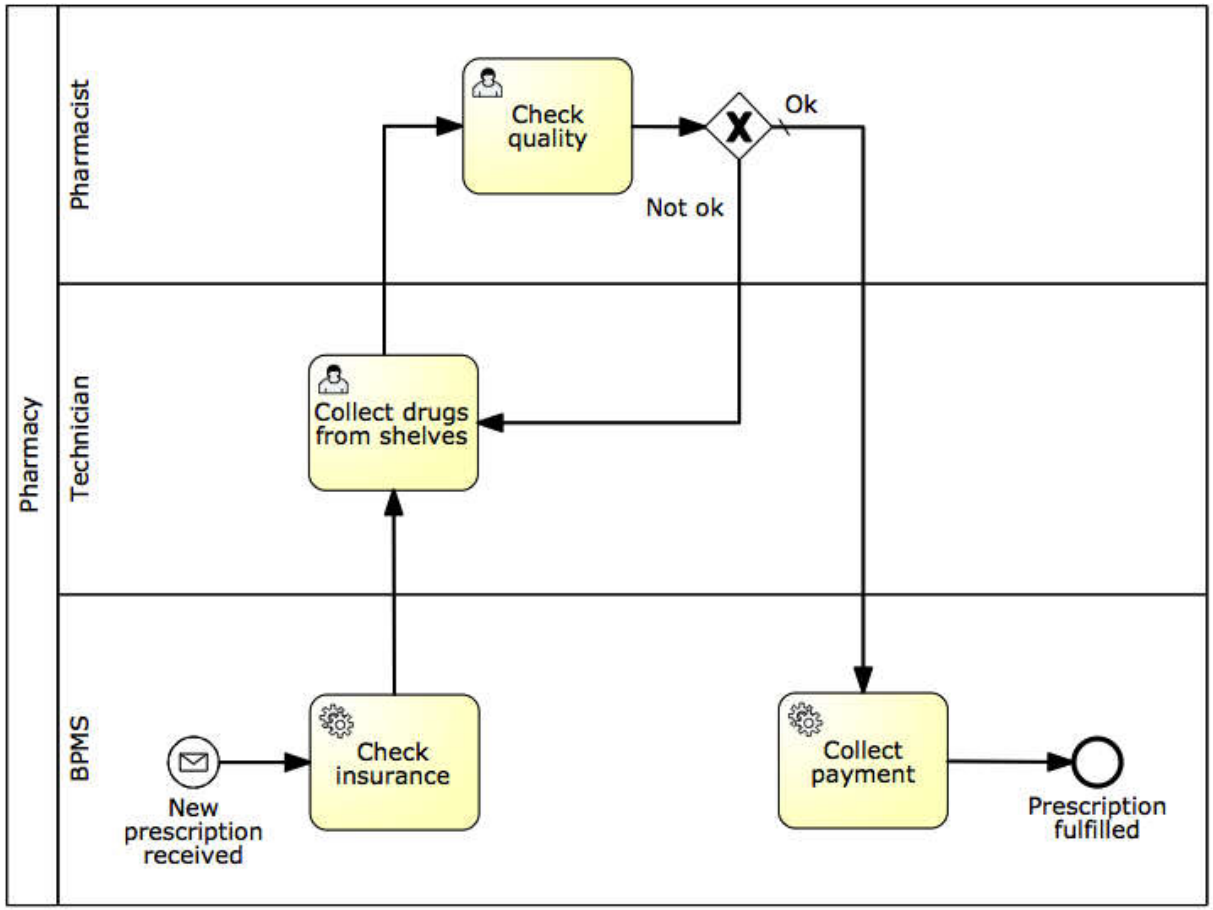
Quiz: let's consider this process fragment

Prescription fulfillment process:

- Once the prescription passes the insurance check, it is assigned to a technician who collects the drugs from the shelves and puts them in a bag with the prescription stapled to it.
- After that, the bag is passed to the pharmacist who double-checks that the prescription has been filled correctly.
- After this quality check, the pharmacist seals the bag and puts it in the pick-up area.
- When a customer arrives to pick up their prescription, a technician retrieves the prescription and asks the customer for their payment.

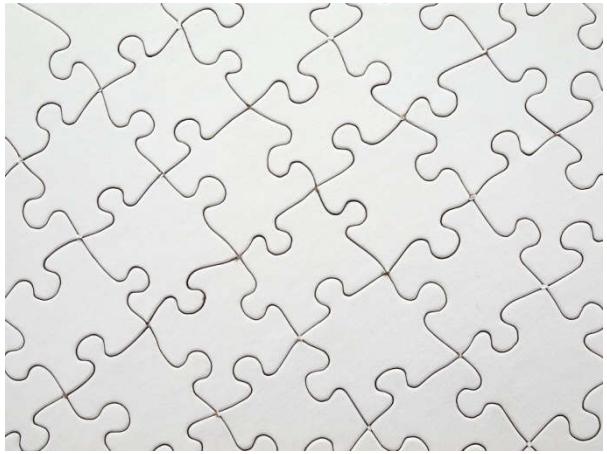
Assume the pharmacy system automates this process. Identify the type of each task and link manual tasks to the system.

Possible solution



3. Complete the process model

Main goal: Establish that the process model is complete



Principle: exceptions are the rule.

-> Add exception handlers

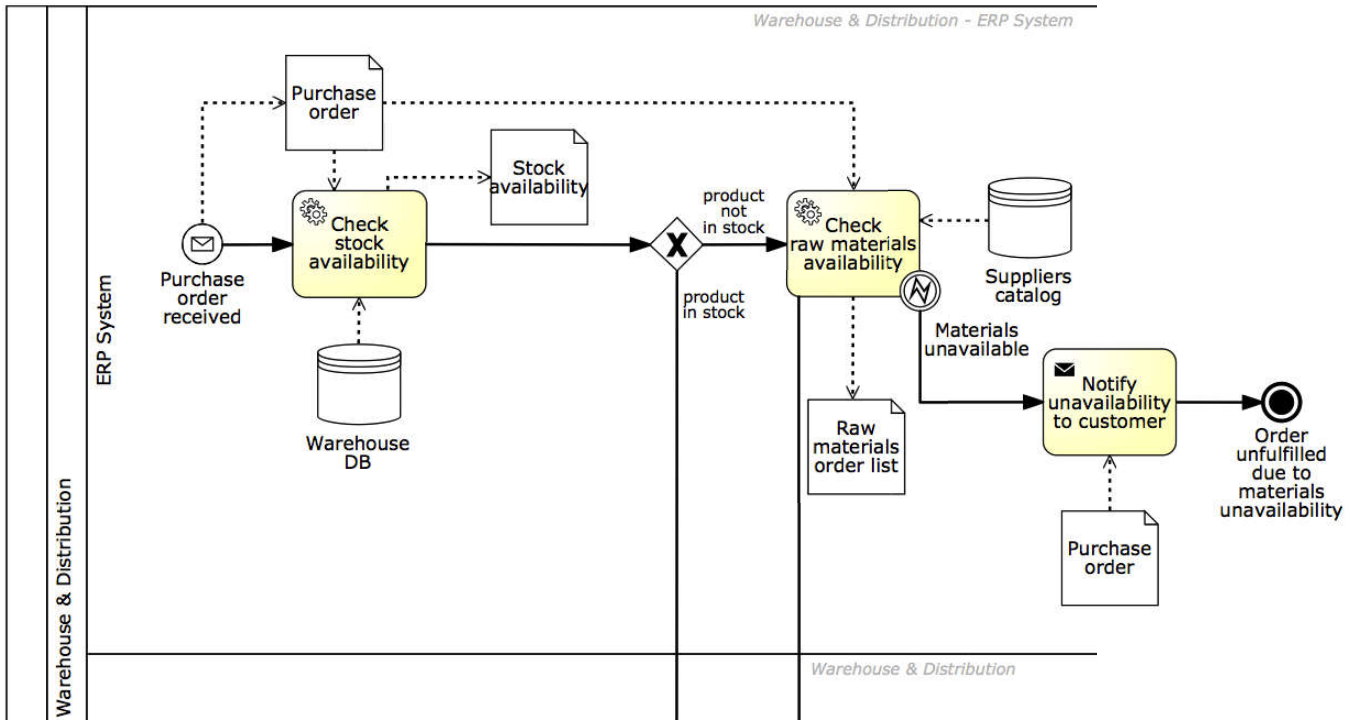
Airline	Flight	Destination	ETD	Gate	Status
	TT8035	ADELAIDE	09:45	3	CANCELLED
	TT5118	SYDNEY	11:50	3	CANCELLED
	TT5504	GOLD COAST	12:00	3	CANCELLED
	TT5106	SYDNEY	14:55	3	CANCELLED
	TT5206	BRISBANE	15:50	3	CANCELLED
	TT5584	ROCKHAMPTON	16:45	3	CANCELLED
	TT5108	SYDNEY	17:05		CANCELLED
	TT5596	PERTH	17:20		CANCELLED
	TT8037	ADELAIDE	17:45	3	CANCELLED
	TT5704	CAIRNS	22:05	3	CANCELLED

It happed for real!

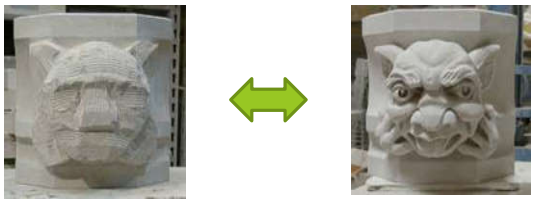
Principle: no data = no decisions, no tasks handover.

-> Specify all <u>electronic</u> business objects

In our example…



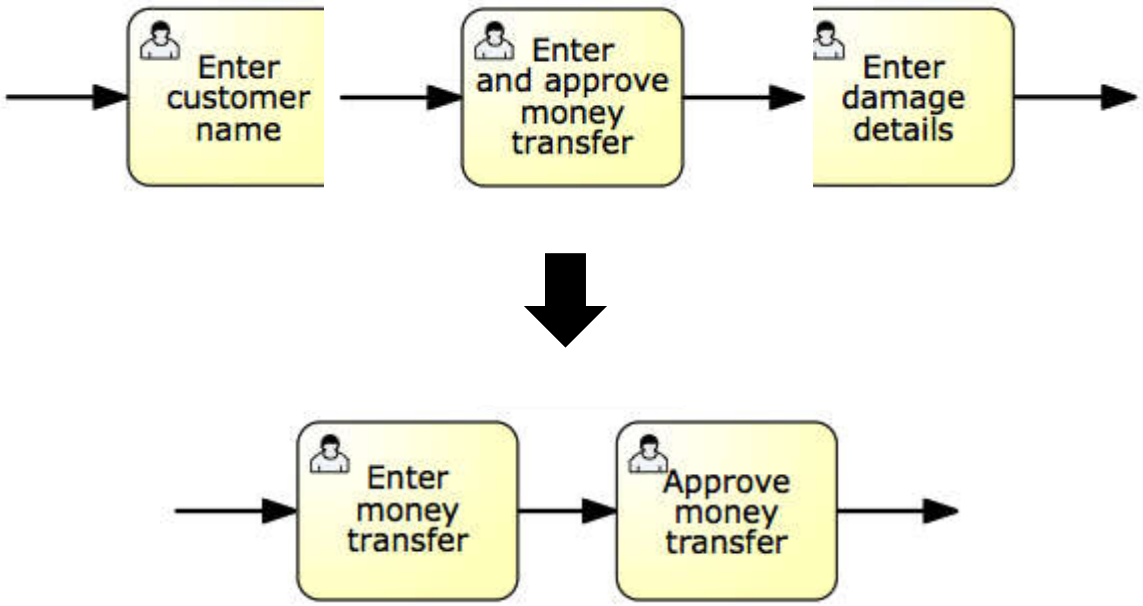
4. Adjust task granularity



Principle: BPMSs add value if they coordinate handovers of work between resources.

-> Aggregate any two consecutive tasks which are too detailed or assigned to the same resource

-> Refine tasks that are too abstract



5. Execution Properties

- To make the model *fully executable*, we need to specify in the last step how each model element is effectively implemented by the BPMS of choice

- The relevant *Execution Properties* are:
  - Variables, messages, signals, errors, and their data types,
  - Data mappings,
  - Service details for service, send and receive tasks, and for message and signal events,
  - Code snippets for script tasks,
  - Participant assignment rules and user interface structure for user tasks,
  - Task, event, and sequence flow expressions, and
  - Other BPMS-specific properties.