University of Pisa

MSc in Computer Engineering

Systems for Strategic Management and Support

LECTURE 17

http://www.iet.unipi.it/m.cimino/pdis/

Mario G.C.A. Cimino

Department of Information Engineering

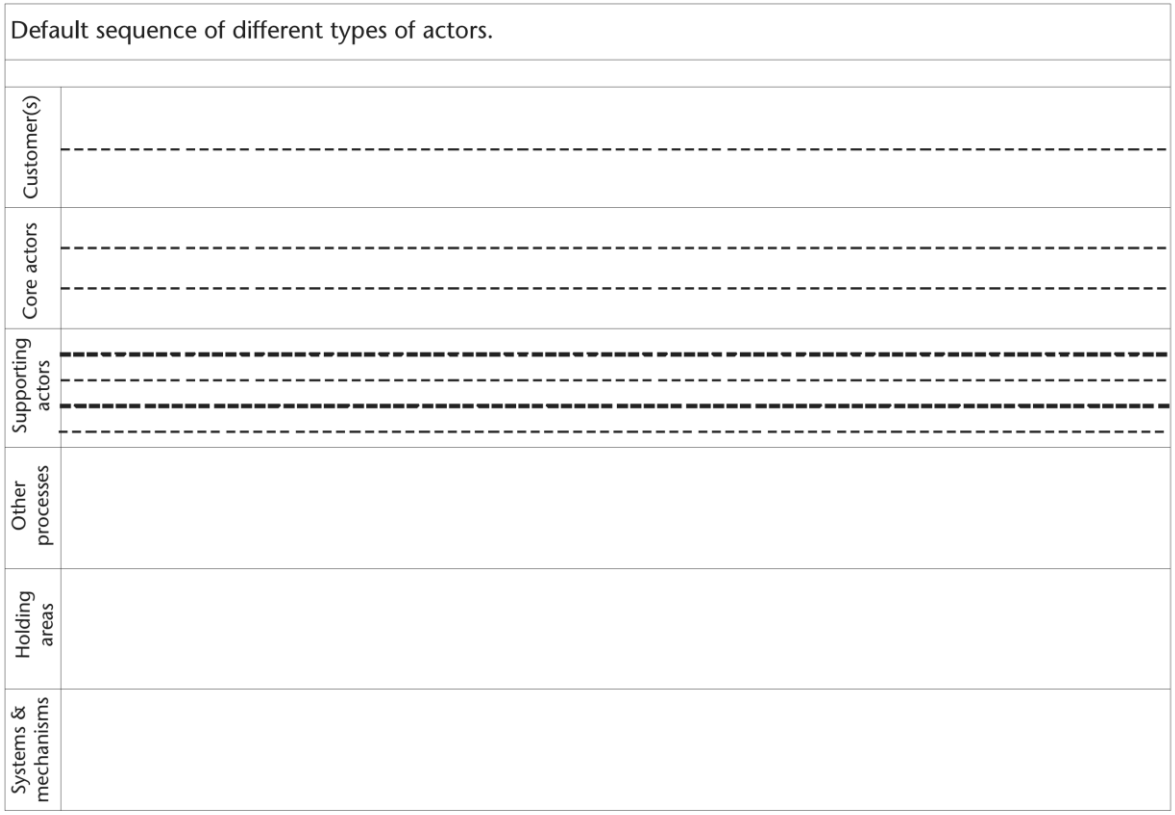
Question to solve: **how to model…**

- Branching?
- Optional steps?
- The role played by systems or mechanisms: when we hand off control of the system, when the system is used to support an activity, but is not given control?
- Interaction with other processes?
- The appropriate level of detail?
- Activities spanning multiple swimlanes (e.g.conversation)?
- Steps that do not happen in a particular order but must all be completed before a subsequent step can begin?
- Steps that interact continuously, or iterate?
- Steps triggered by the clock?
- Steps carried out by an actor with a very small role in the workflow?

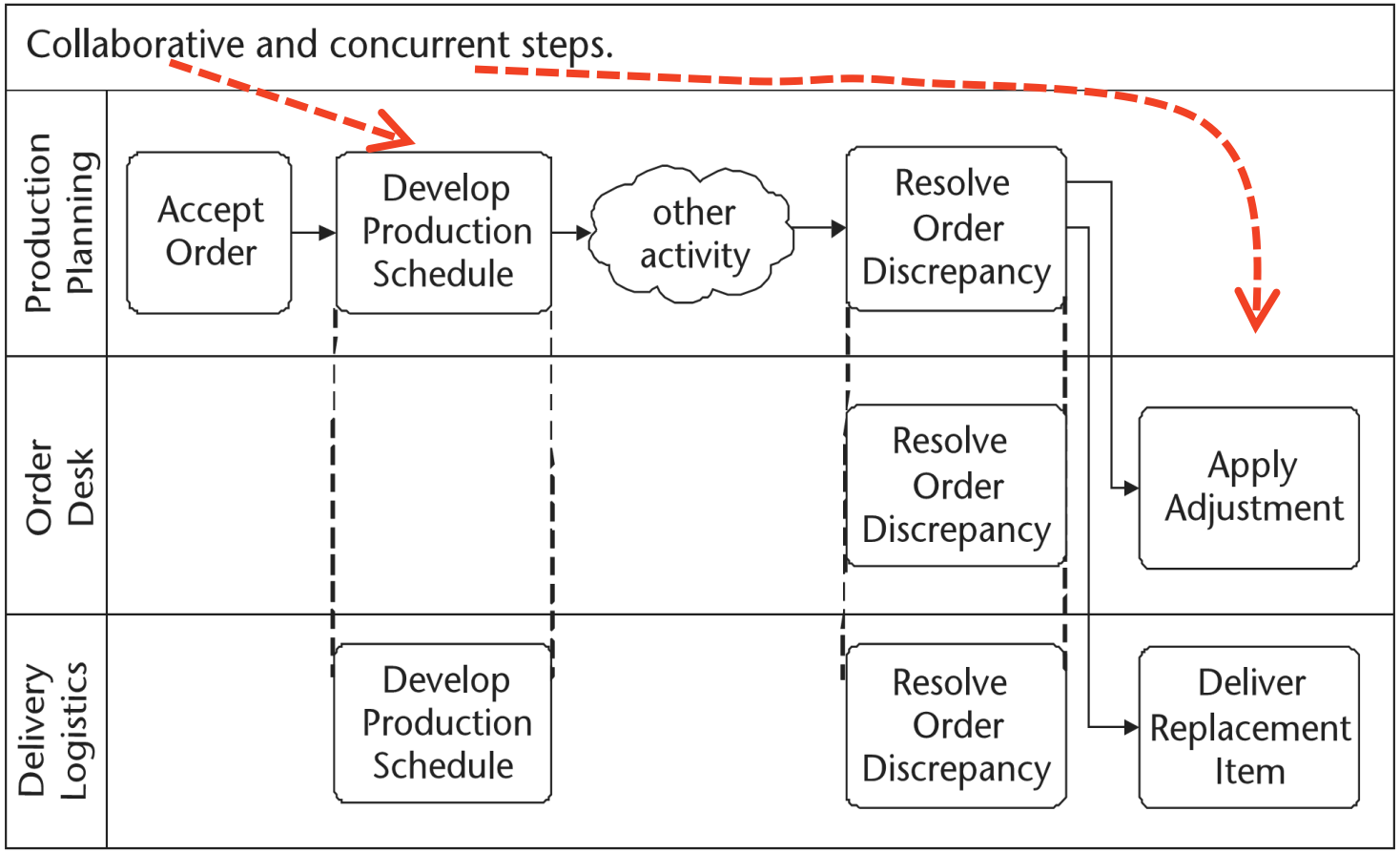
- **Actor** can appear from top to bottom in an order that makes sense: order of appearance (default), starting from the busier actor (to highlight the points of the flow with main participants), according to their physical position (in *Lean* practice, to limit “up-and-down”)

A default order related to the type:

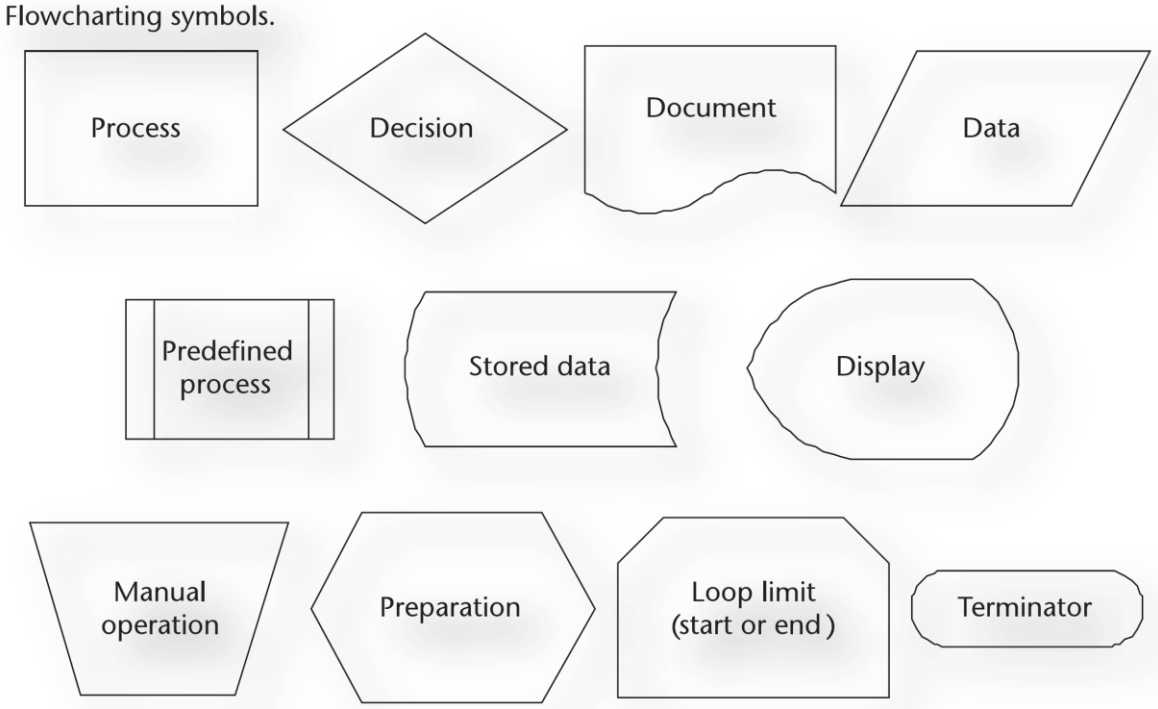
1. Customers
2. Core Actors,
3. Supporting Actors,
4. Other Processes,
5. Holding areas,
6. Systems and mechanisms (machinery, equipment,…)



- Do not confuse collaborative work with concurrent (parallel) work: in a collaborative step, the actors are working together on the same step, while in concurrent work they are working independently on separate tasks.



- Avoid notations used for modeling specialized processes (e.g. software engineering, industrial engineering).
- Swimlane diagram aims at the key aspect of the process (the flow of work): anything else distract attention, add noise, not information.
- Workflow modeling needs the participation of a large range of people: avoid overformalization with IT flowcharts.

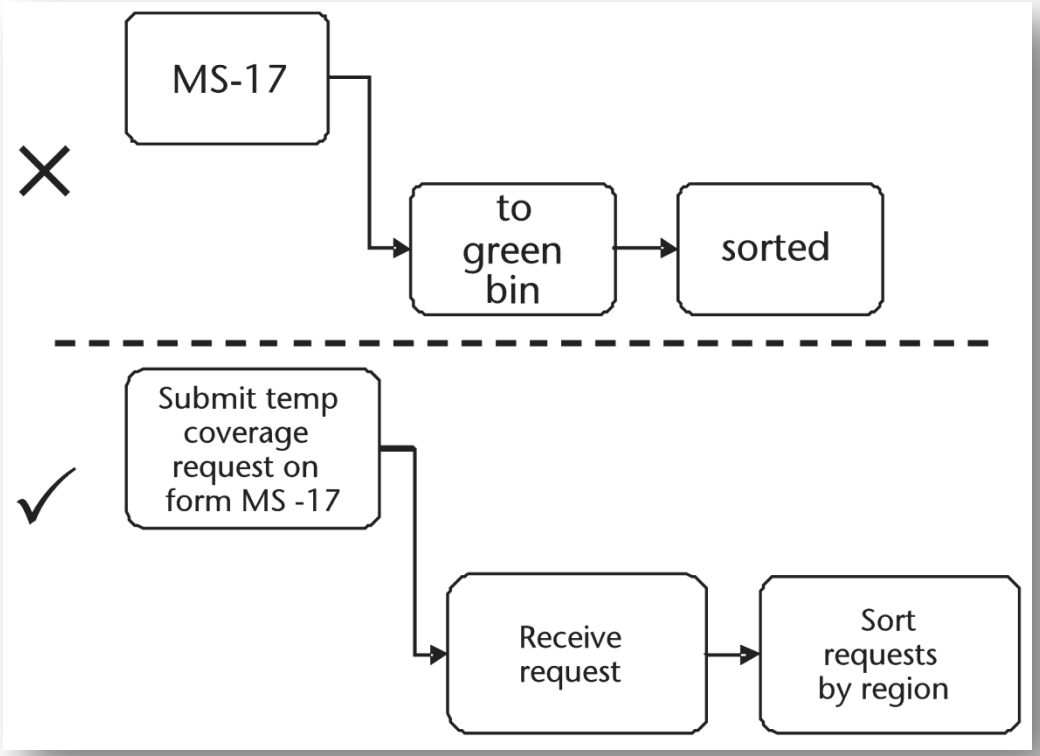


Which steps to include?

- An actor might tell you about many activities is responsible for, but you are modeling a specific process, from trigger to result
- The process usually traces a single work item or "package" of work items, such as a service problem, an order, an engineering upgrade, a building permit request, an item being manufactured, a material requisition, a replacement part.
- Any activity that "holds" one of these work items should be part of the process, whether is adds value or not to the process. It may introduce delay, move the work along, subtract value.

Guidelines for naming process steps

- Avoid cryptic step names. Follow the same guidelines for naming a process (*verb-noun* or *verb-object* format) with additional detail.
- The step name should convey the result achieved by the step, if flip the verb-noun format.



- Basic structure:
  - + Action verb (assign, validate, sort,…)
  - + Optional qualifier (initial, replacement,…)
  - + Noun(s) (service request, payment,…)
  - + Optionally, information on how (by form, by fax,…)

- Remind, the name is **not**:
  - an area or function such as *titling*, *accounting*, *inventory*
  - an event or result such as *claim arrives*, *claim is registered*
  - a state such as *sorted*
  - based on mushy verbs or jargon

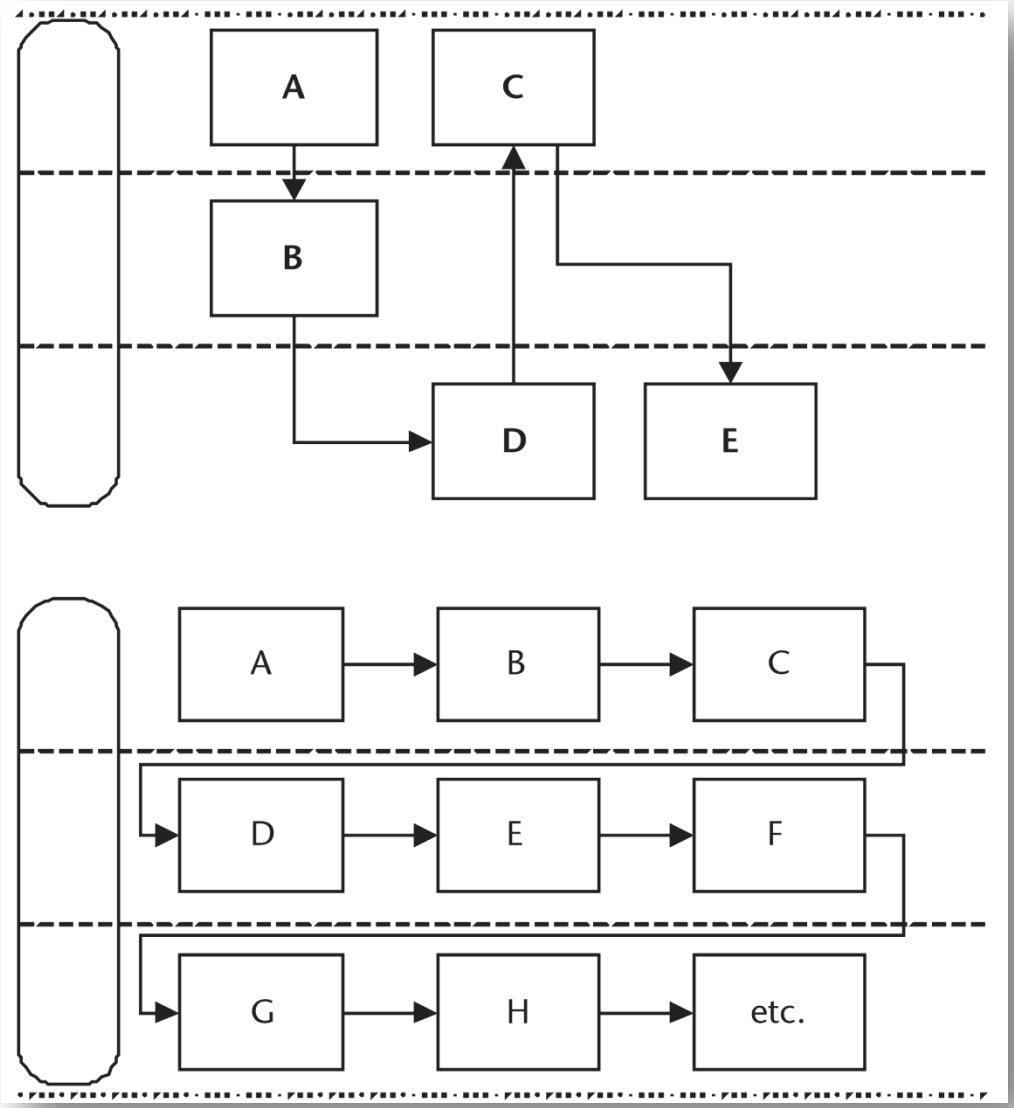
Flow

- Is the passing of work from one step to another. The next step cannot start until the previous step has been completed.
- A handoff is a special kind of flow where the work passes from one actor to another. It is often a place of delay, errors, expenses.

- Avoid the layouts represented Figure:

The objective of a workflow diagram is to graphically show sequence, dependency, time.

The objective is **not** to save paper



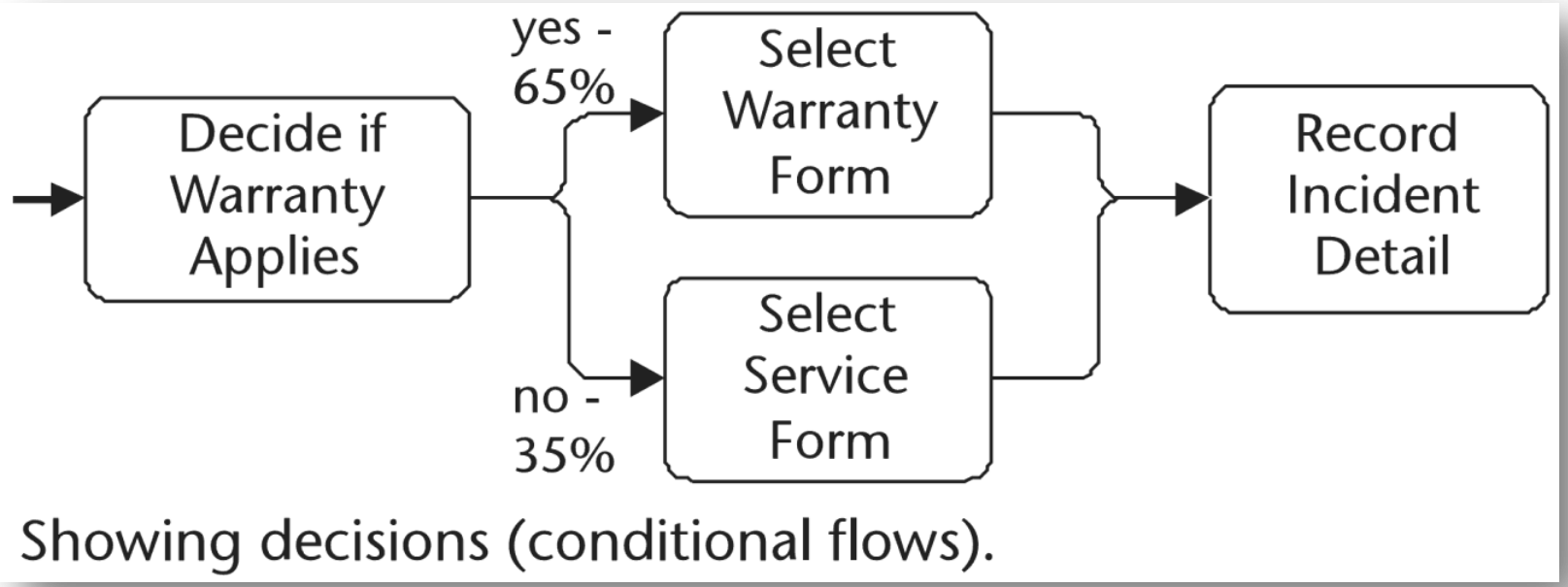
- Distinguish the concepts of *data flow* and *work flow*.
- When data flow is superimposed on a workflow model, the latter is not so evident any more

Data Flow -
data from Provide Quote is used by Issue Invoice

Work Flow -
various steps between Provide Quote and Issue Invoice

Exclusive flows, key points:

- the decision is computed before the diamond, use a dedicated step for this: *decide/determine if…*
- label each branch to indicate which decision outcome wil follow that path, along with statistics if appropriate



Showing decisions (conditional flows).

Managing progressive details:

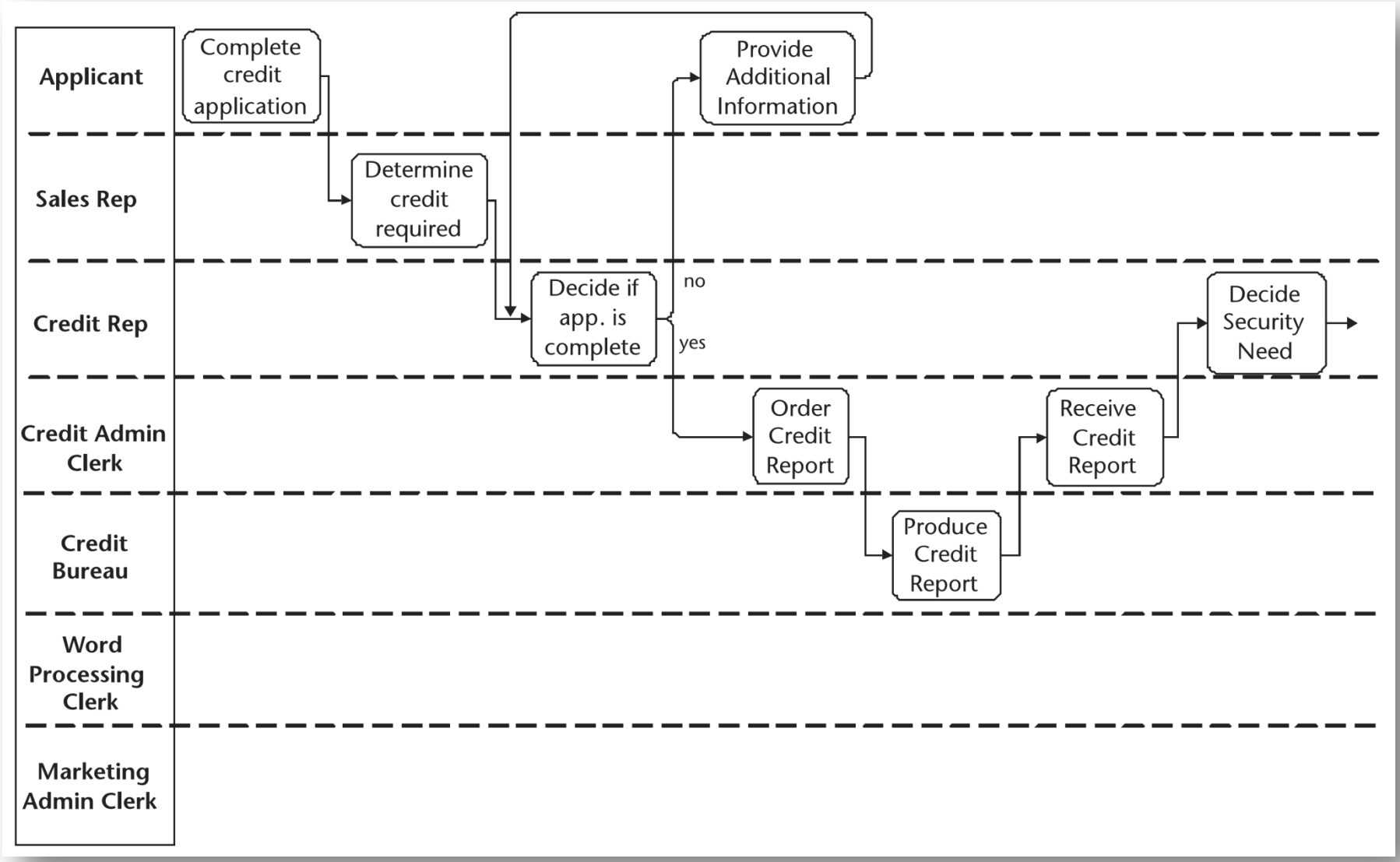
- To avoid facing with unmanageable complexity, do not jump immediately into modeling minute details instead of building layers of progressively detailed models: overall process map → business processes → sub-processes → process step.
- Save your energy by avoiding **the curse of detail**: if you add more and more detail, seemingly unable to stop, your project will be canceled and you will never get to work on the *to-be* process!
- There are three levels of a workflow diagram:

- Level 1 or *handoff-level* diagram (i.e., the flow of work)

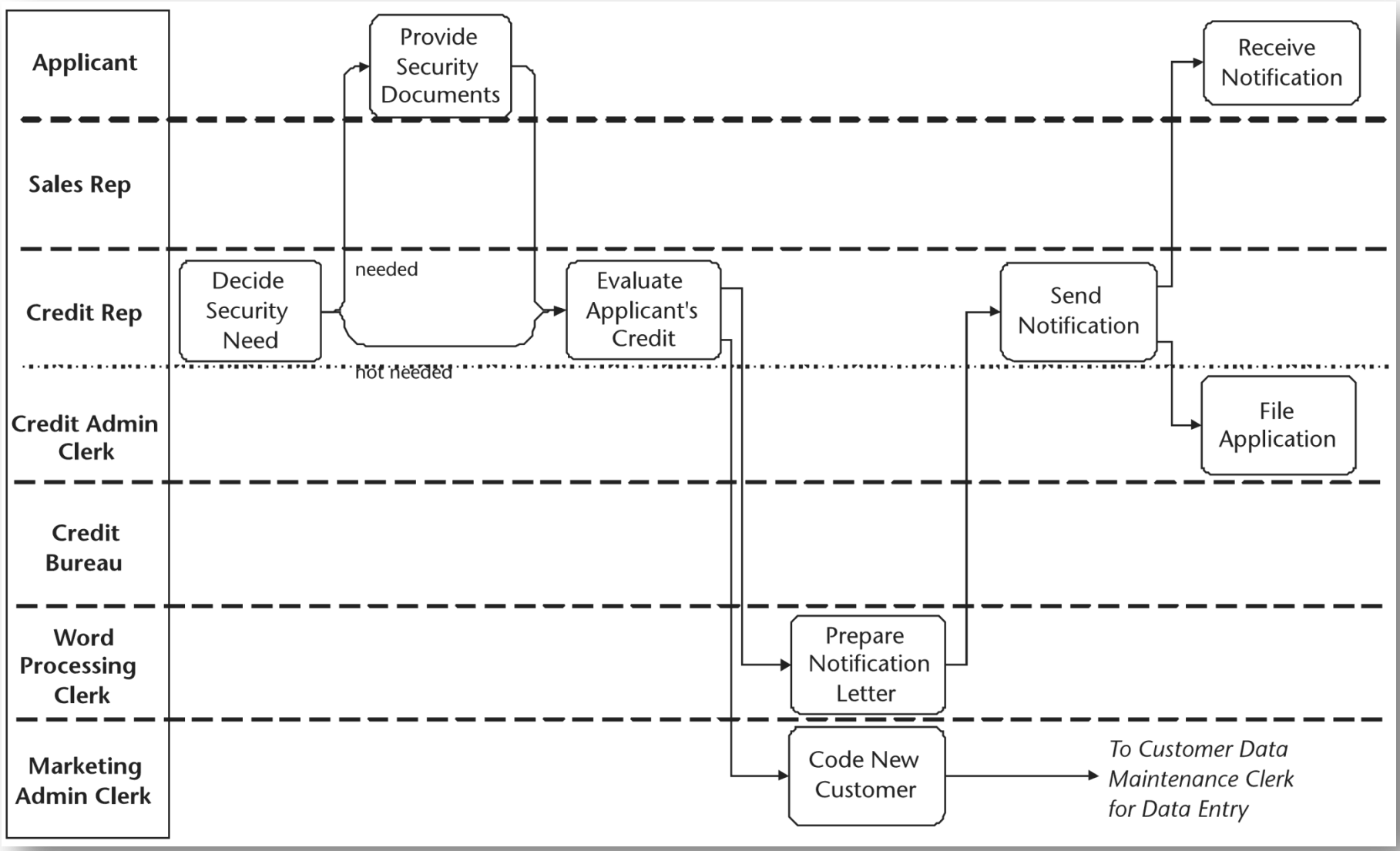
- Level 2 or *service-level* diagram (related to SOA and BPM)

- Level 3 or *task-level* diagram (rarely used)

- An example of workflow model segmented into two sub-diagrams



Workflow Modeling



Handoff diagram:

- It makes the overall structure in which each step summarizes the actor's involvement at a specific time in a process: whenever an actor does a lot or a little of work, draw *one* box and *move on*.
- The visual distinction in terms of relative amount of work becomes apparent with the service diagram
- Simplify steps, not actors. Handoff diagram is important to highlight "delay, errors and expenses". It makes "yo-yoing" immediately evident, in contrast with a detailed diagram.

When to stop modeling

- You can stop modeling the *as-is* process as soon as you are able to understand why the process behaves the way it does

- This can happen at the handoff level, if something like a timing issue, convoluted workflow, or bottleneck is identified as the root cause of performance problem

- Usually we have to proceed to a second-level diagram.