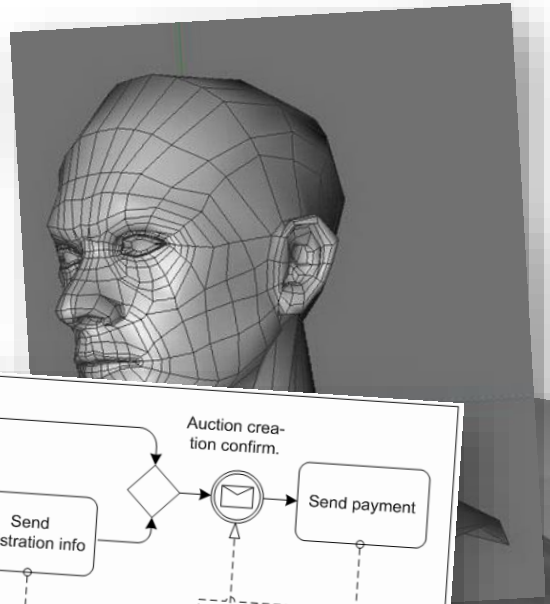
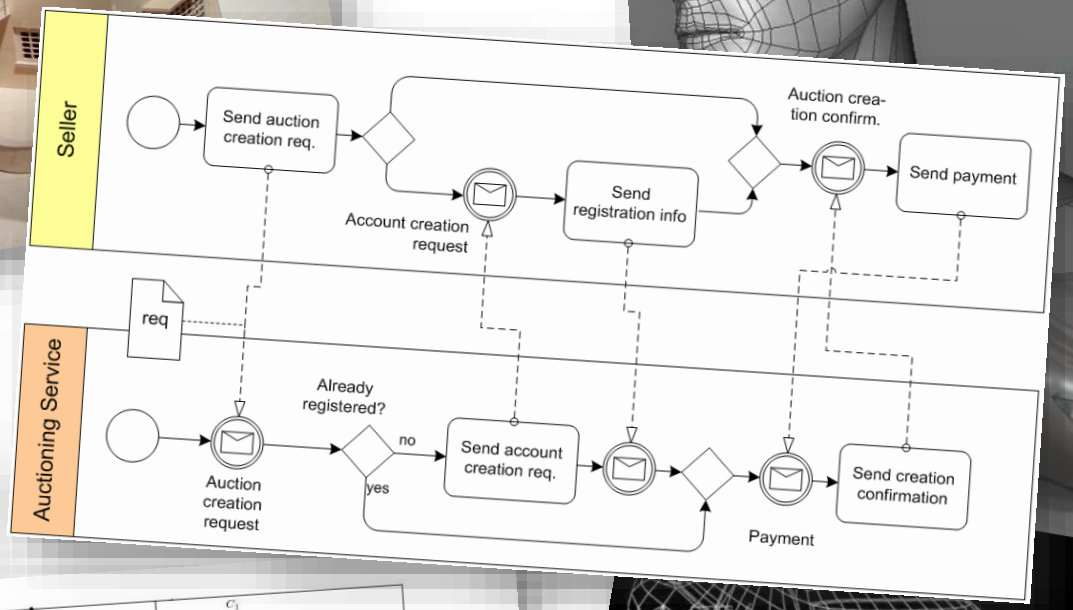
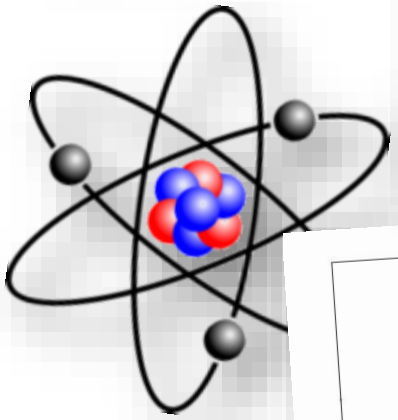
# Why models and modeling?

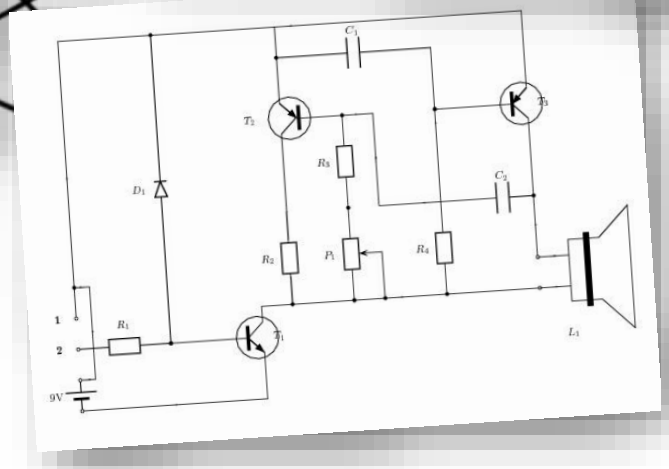
- Real life phenomena are **complex** thus difficult to describe just using text
- Models can be used to **understand** them
- Models are also tools for **planning**

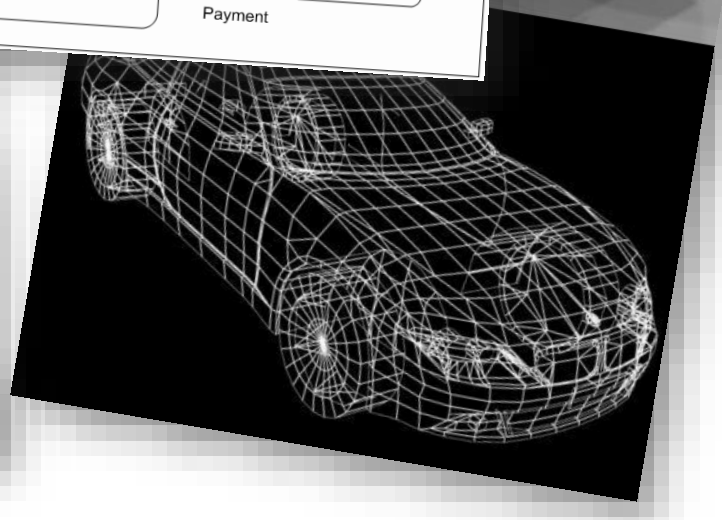
→ Modeling has to be **goal driven**




Seller
Send auction creation req.
Account creation request
Send registration info
Auction crea-tion confirm.
Send payment
req
Auctioning Service
Auction creation request
Already registered?
no
yes
Send account creation req.
Payment
Send creation confirmation







# Model characteristics

What do all these models have in common?

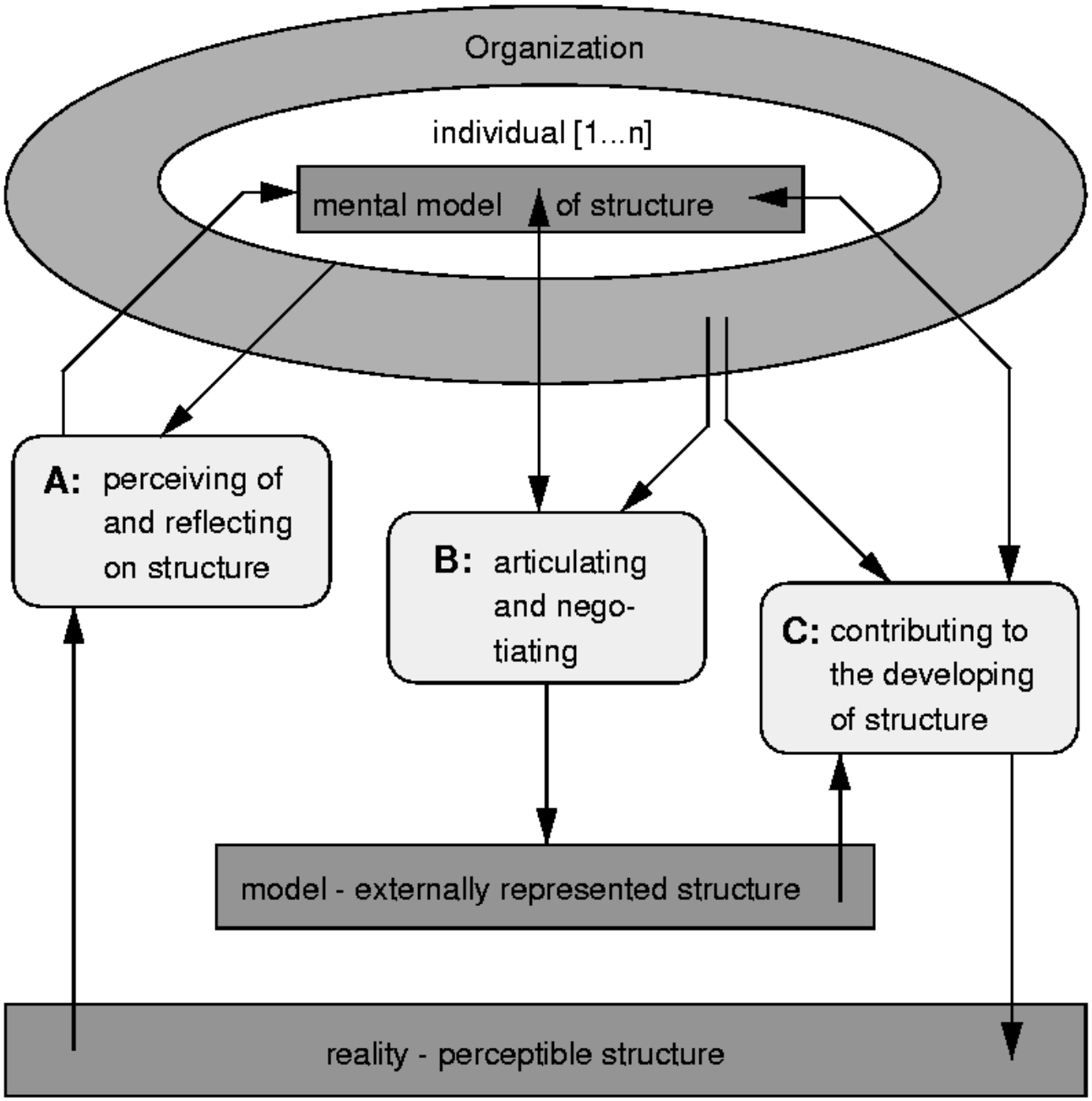
- **Representation** of real life phenomena
- Do not depict all but only **relevant aspects**
- **Replace** certain subjects for a certain amount of time in a limited manner
  (Stachowiak, 1973)

→ Models are usually only useful a for a **certain period of time**

# Model characteristics

What do all these models have in common?

- Do not only represent but may also be used to **form reality**.
- Models are **not self-explanatory**. Additional **context** is required to understand them.
- Models are the **result of a construction process** done by a modeler.


Organization
individual [1...n]
mental model
of structure
A: perceiving of and reflecting on structure
B: articulating and negotiating
C: contributing to the developing of structure
model - externally represented structure
reality - perceptible structure

# Coordination Theory

Malone and Crowston (1990)

- **Processes:** Means for Coordination
- Which are the **elements** of Processes / Coordination?
  - Actors
  - Activities
  - Interdependencies (between resources)
- Elements of all modeling notations
- Need to coordinate: Development of **common notation**

# Coordination Theory

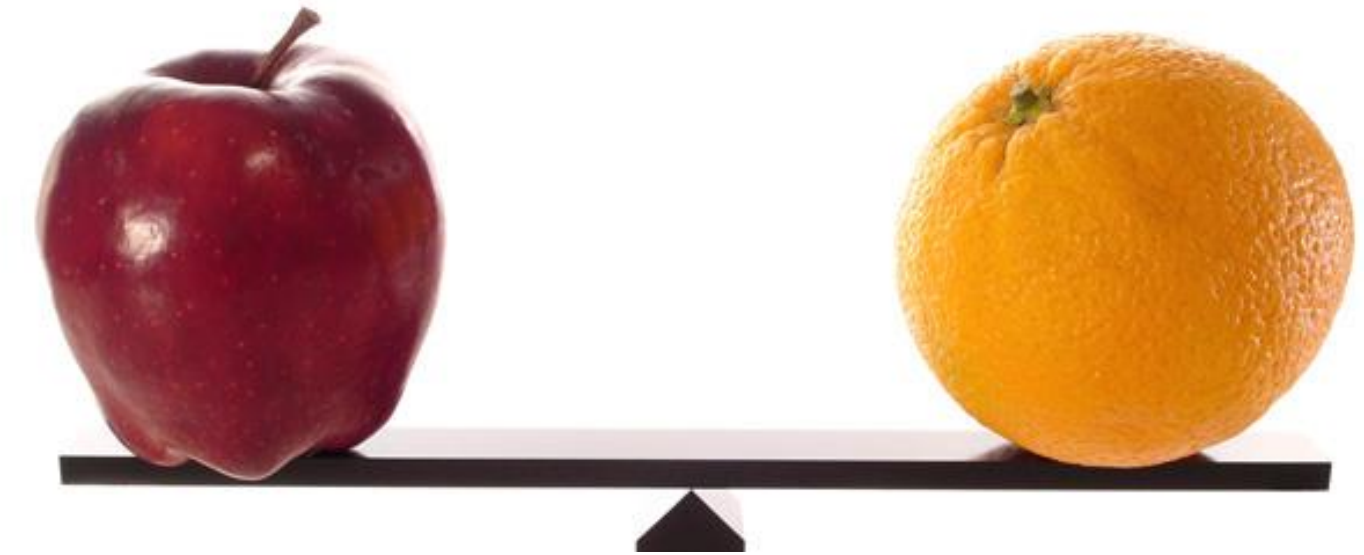
Malone and Crowston (1990)

- **Definition:**
  - Relating goals, activities and actors to each other
    **AND**
  - Managing interdependencies

- **Interdependencies between activities** can be analysed by common / shared entities (objects)

# Interdependencies between activities: Results

- Activities may have **similar** or **conflicting** results



# Interdependencies between activities: Results

- Activities may have **similar** or **conflicting** results

| Kind of interdependency | Coordination mechanism |
| --- | --- |
| • Similar result | • Identify and stop redundant work: Merge activities or choose one |
| • Overlapping results | • Negotiate common results |
| • Conflicting results | • Choice of one activity / task |

# Interdependencies between activities: Input

- Activities may need the **same input**

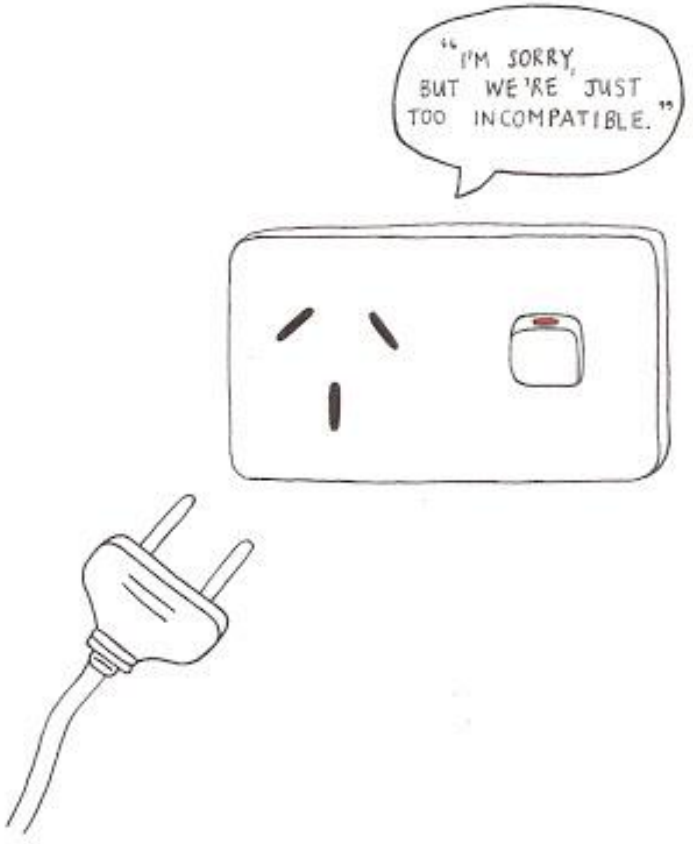
# Interdependencies between activities: Input

- Activities may need the **same input**

| Kind of interdependency | Coordination mechanism |
| --- | --- |
| • Sharable resource | • - |
| • Non-sharable resource | • Make conflicts transparent<br>• Plan use of resources |
| • Non-reusable resource | • Choose one activity |

# Interdependencies between activities: Input

- The **result** of one activity can be the **input** for another activity



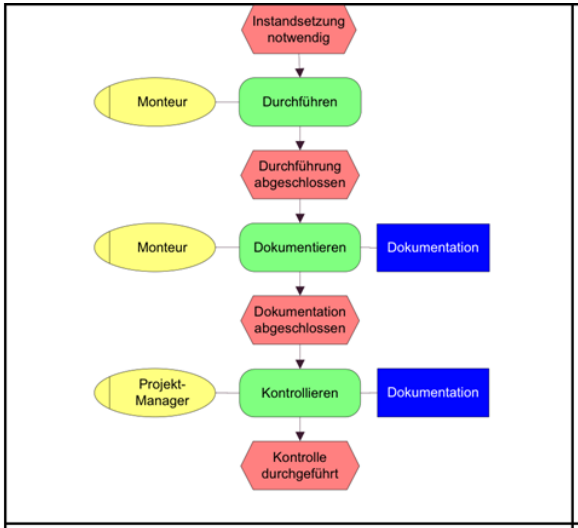
# Interdependencies between activities: Input

- The **result** of one activity can be the **input** for another activity

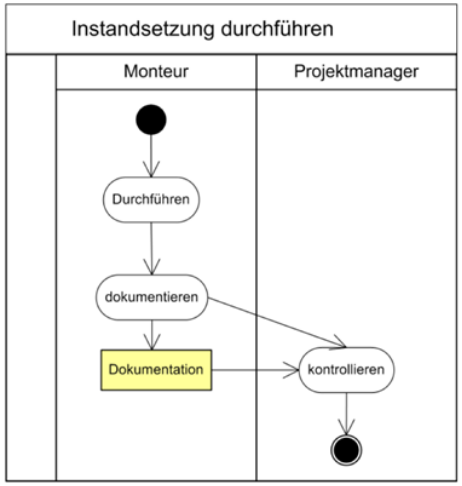
| Kind of interdependency | Coordination mechanism |
| --- | --- |
| • Compatible | • Fix sequence<br>• Ensure usability of output<br>• Control resource flow |
| • Incompatible | • Re-structuring of activities<br>• Adding conflict-resolving activity |

# Modeling notations

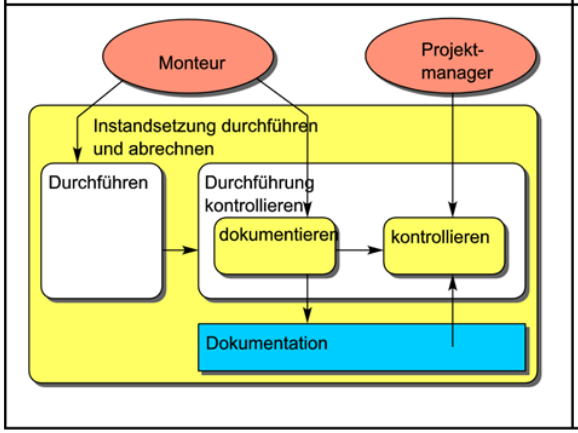
EPC



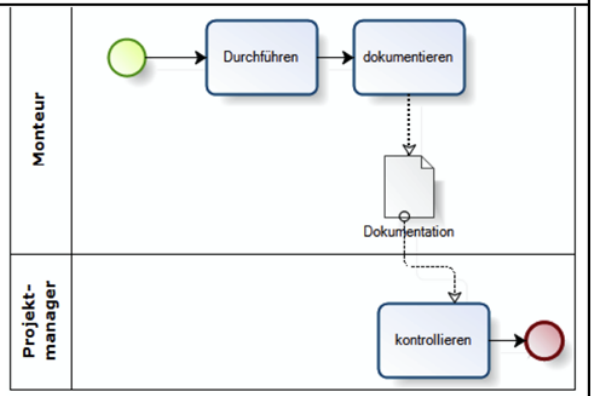

UML




See Me

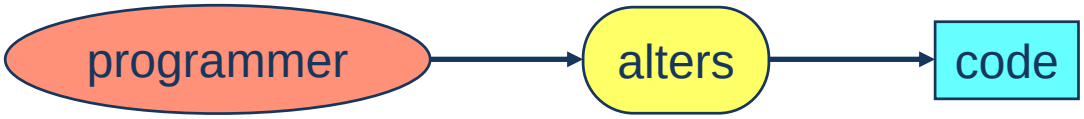



BPMN



# Specifics of visual modeling notations

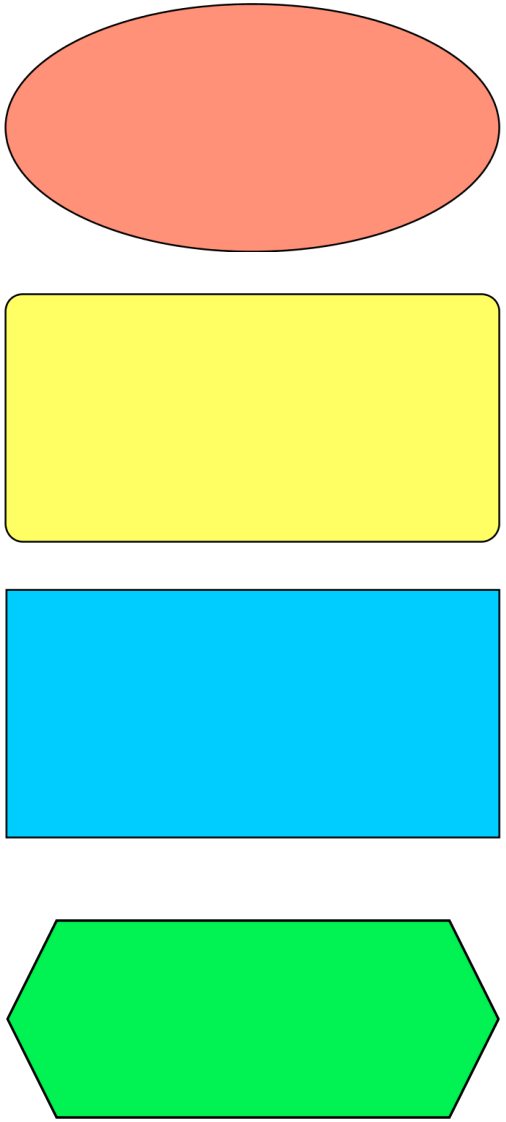
- Consist of **elements** (visual shapes) and **relations** (connections between them)
- A **syntax** describes how these elements may be interconnected with each other
- A **semantic** describes their meaning in relation to real life phenomena




Some modeling notations contain **multiple diagram types** (e.g. UML)

# Elements of modeling notations

- Organizational structure
  - Roles (e.g. engineer, project manager)
  - Rights and responsibilities
- Functions
  - Workflow
  - Executed by roles or systems
  - Same granularity as other functions
- Information
  - Data and containers (artifacts)
- Resources
  - In- and Out-put for (Sub-)processes
- Control
  - Branches and conditions



# Typical challenges of modeling

- **Creating a model:** How to start, how to go on?
- **Appropriate abstraction:** What to include, what to leave out in a model?
- **Representing real life:** How to translate real life phenomena into modeling notation?
- **Choosing modeling notation and tools:** What is appropriate for whom / for which purpose?

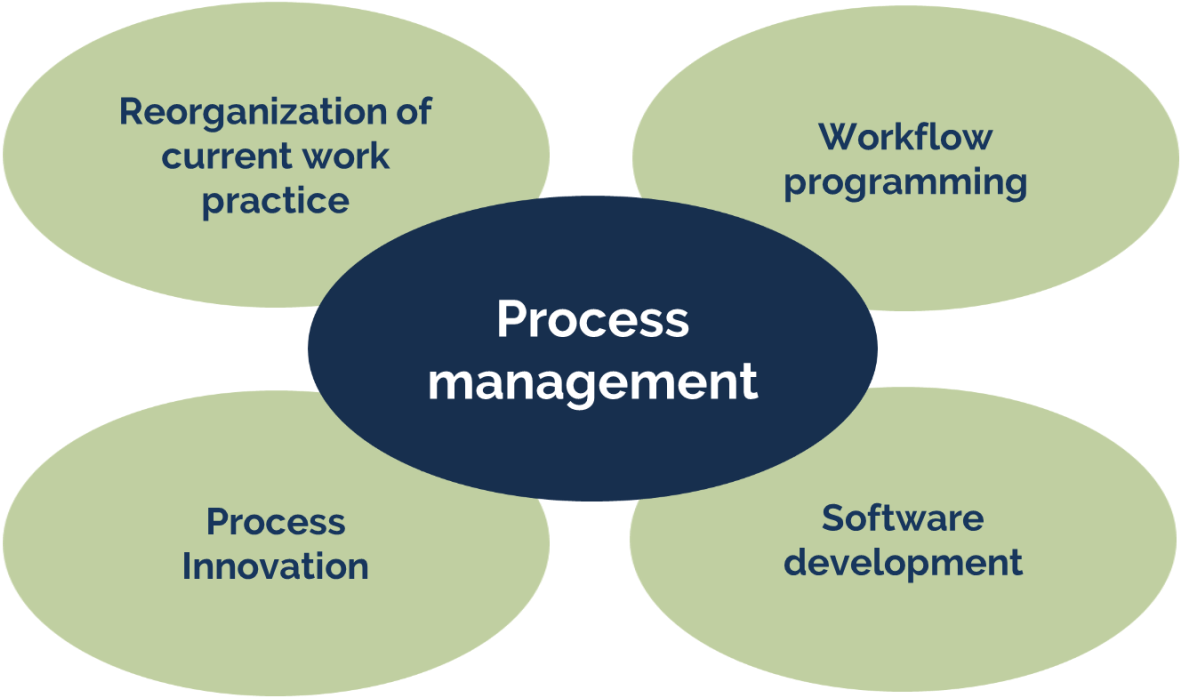
# Modeling: How to do it?

Freund and Götzer (2008)

- Decide which **elements / perspectives of modeling** should be used
- Gather information about **process** and its **context**
- Use **reference models**
- Choose a suitable modeling notation
- Analyse processes
  - Existing processes
  - Requirements for new processes

# Choice of a modeling notation

- **Goal** of a modeling project




There is **no modeling notation** that fits everywhere

# Choice of a modeling notation

## Additional criteria

- **Purpose / goal**
  - Development, analysis, communication, ...
- **Target group(s)**
  - End user, technicians, management, ...
- **Experience** with notations / notations in use
- Extent of notation / aspects to be modeled
- Phase of process management
  - Documentation, analysis, formalization

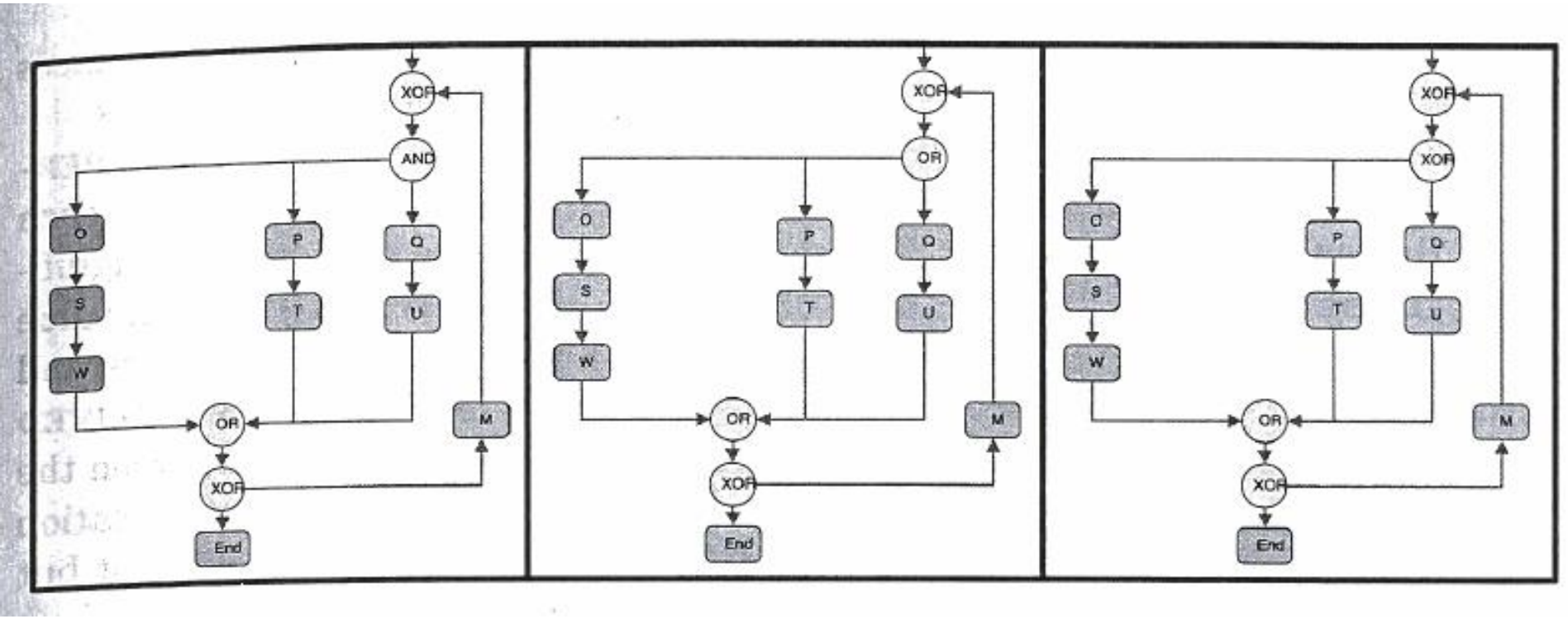
# Choice of a modeling notation

Freund and Götzer (2008)

- **Syntactical** correctness / quality
  - Is there a meta model for verification? Is it required?
- **Semantic** correctness / quality
  - Can all relevant aspects be represented?
- Economic **efficiency**
  - Effort of modeling, maintenance, usage?
- **Clarity**
  - Can the model be understood? By whom?

# Understanding models (or not…)

Reijers et al. (2007)




- If T is executed for a case, can U be executed for the same case?
- Can T, M and O all be executed for the same case?

# Choice of a modeling notation

## Degrees of abstraction and complexity

- **Simple** metaphors
  - e.g. activities and sequences
  - Coarse analyses, communication, overview
- **Formal** models
  - From simple to more complex
  - Analysis and design of processes
- **Programmatical** models
  - Technical orientation
  - Automation