University of Applied Sciences Northwestern Switzerland
School of Business

Master of Science
Business Information Systems

Modeling and Meta-Modeling

Models and Modeling

Modeling

Describing and Representing all relevant aspects of a domain in a defined language.

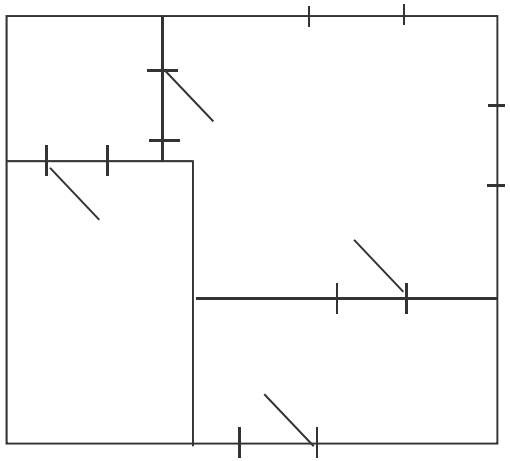
Result of modelling is a model - an exemplary reproduction of reality.

Model

A reproduction of the part of reality which contains the essential aspects to be investigated.

Model and Real Object in Architecture

modell (plan)



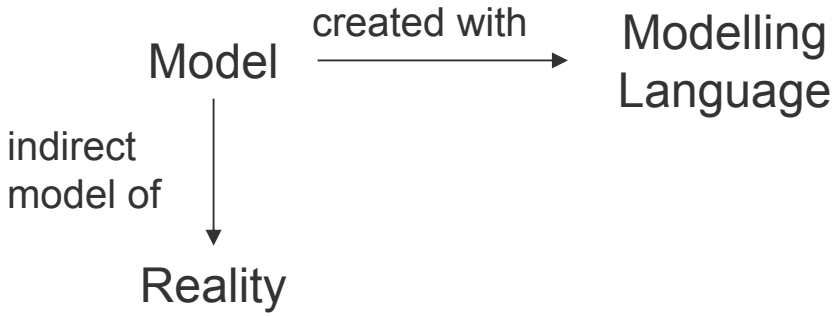
real object

Rationale for Modelling

- Models provide abstractions of a physical system that allow engineers to reason about that system by ignoring extraneous details while focusing on relevant ones.
- All forms of engineering rely on models to understand complex, real-world systems.
- Models are used in many ways:
  - predict system qualities
  - reason about specific properties when aspects of the system are changed
  - communicate key system characteristics to various stakeholders

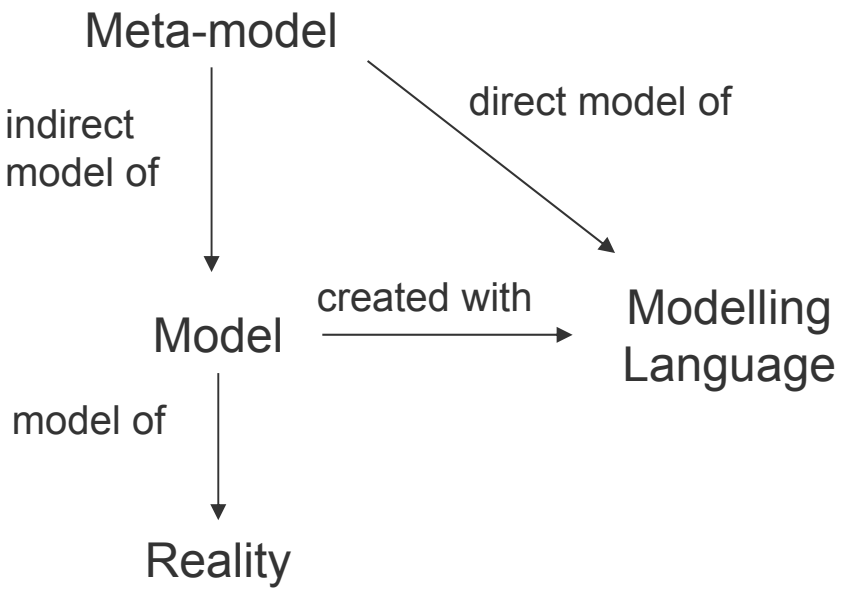
(Brown 2004)

Modelling Language



- A modelling "language" specifies the building blocks (elements) from which a model can be made.
- There can be different types of modelling languages, depending on the kind of model
  - graphical model
  - textual description
  - mathematical model
  - conceptual model
  - physical model

Meta-model



A meta-model defines the modelling language, i.e. the building blocks that can be used to make a model. It defines the

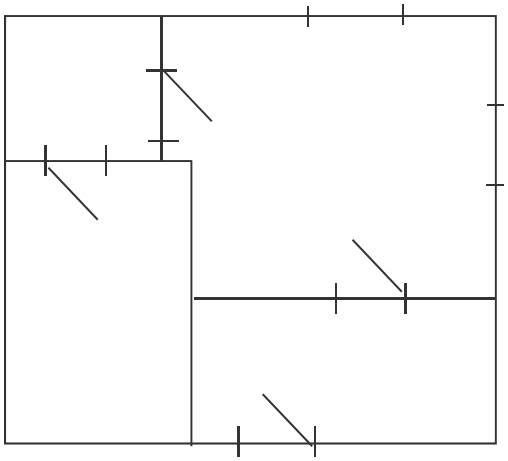
- object types that can be used to represent a model
- relations between object types
- attribtues of the object types
- meaning of the object types
- rules to combine object types and relations

Model and Meta-Model in Architecture

real object

house

model

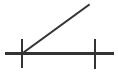


architect's drawing (plan)

meta-model
(modelling language)

object types:

wall



door

window

rules:

- a door is adjacent to a wall on both sides
- Windows are on outer walls.

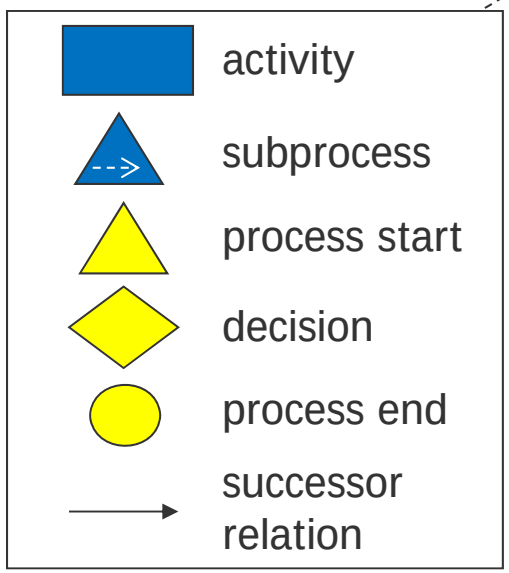
Illustration: Meta-model and Model of Flow Diagrams

Meta-model:

A flow diagram consists of object types for active elements «activity» and «subprocess», control elements «process start», «decision» and «process end» and the relation type «successor».
The elements have attributes and there are rules how the elements can be combined.

Modeling Language:

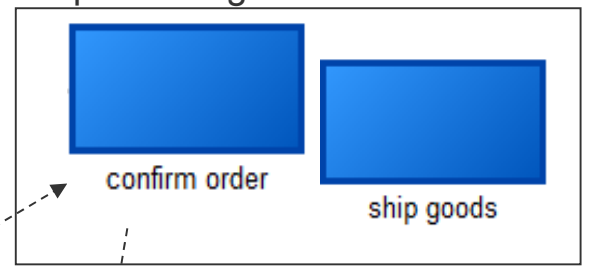
Syntax (appearance) and semantics of meta-model elements



Objects:

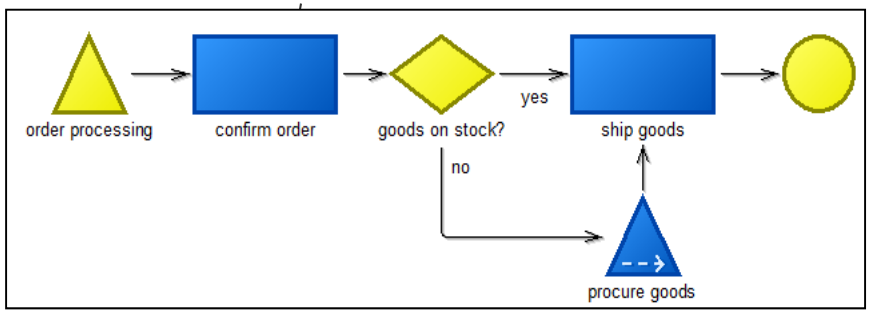
Instances of object types representing real entities

Object „confirm order“ is an instance of the object type "activity"



Instance of "confirm order" in a particular model

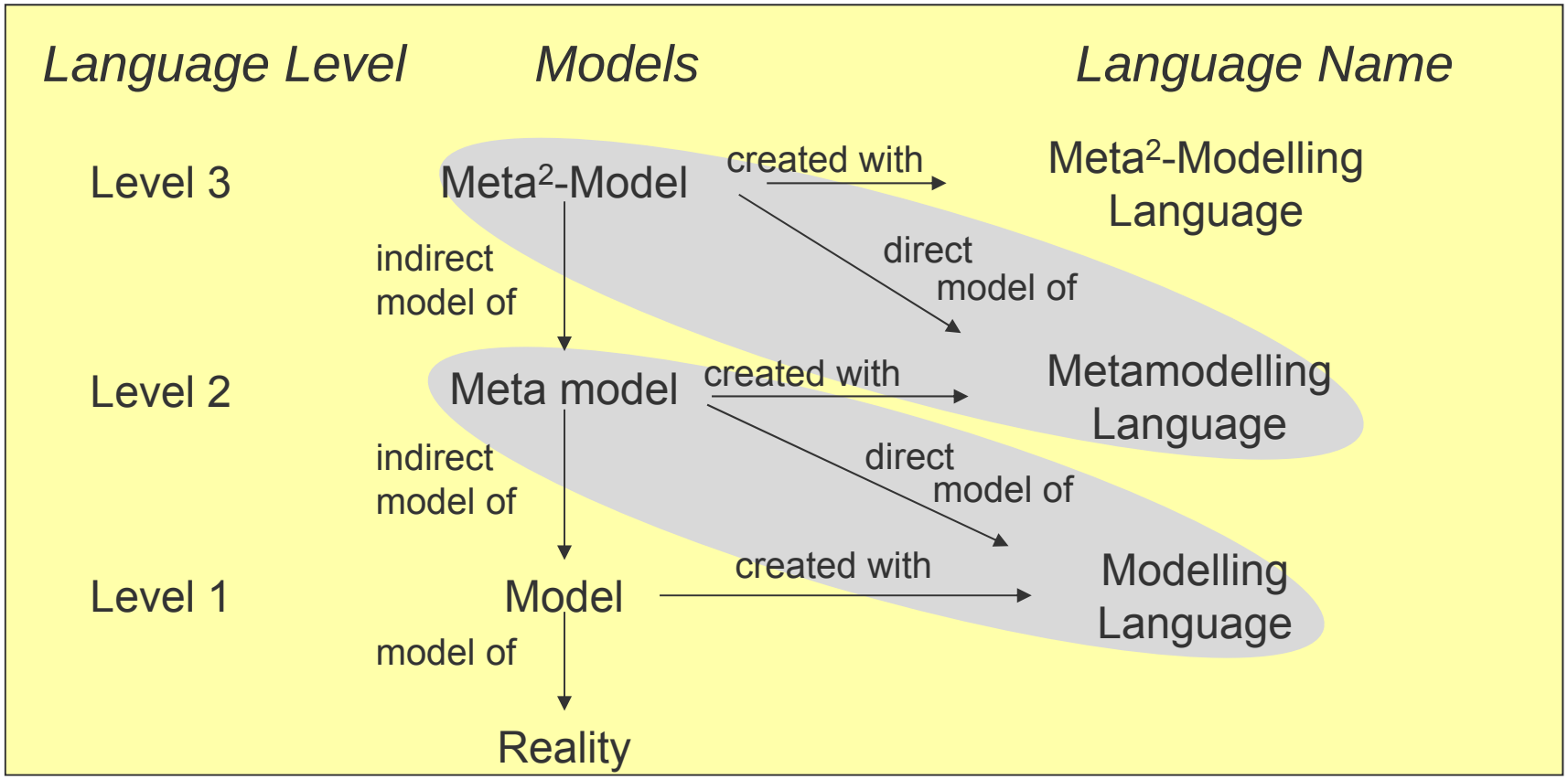
Model:



combination of objects representing a process

Meta Model Hierarchy

The meta-model must again be described in some language, which has to be specified in a meta-model

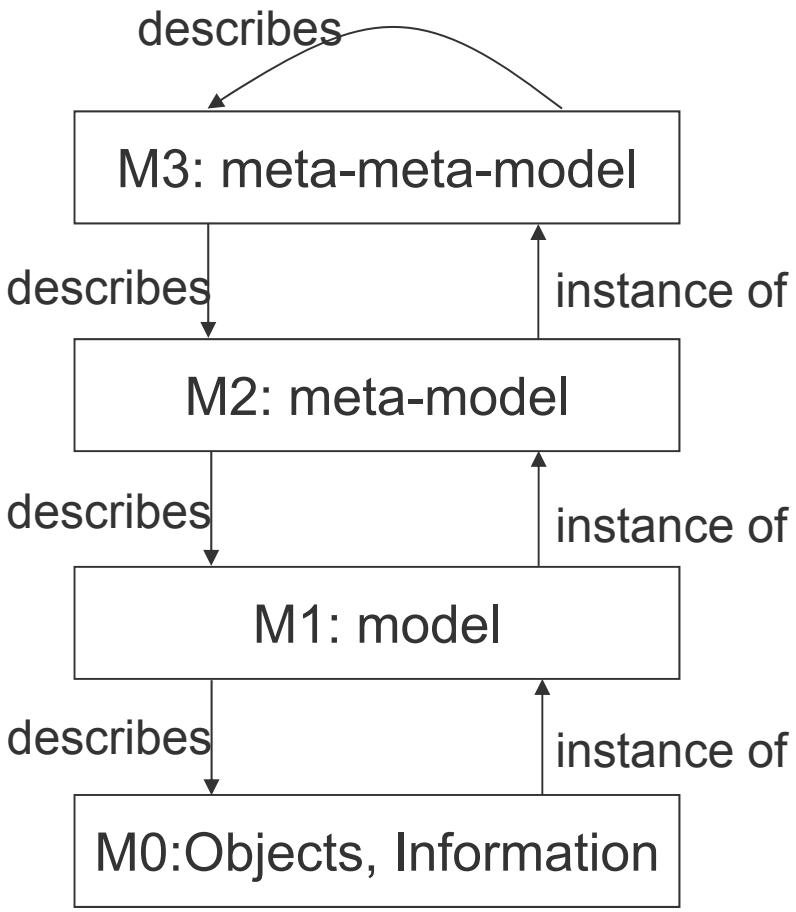


Often the meta-model and the modeling language are unified and not distinguished.

MOF – Meta Object Facility

- The Meta Object Facility (MOF) is an OMG meta-modeling standard.
- MOF is itself a *meta-meta-model*, a specification describing how one may build meta-models.
- MOF is closely based on Unified Modeling Language (UML):
  - Meta-models are represented with class diagrams of UML
- MOF defines the theoretical underpinnings of the XML Metadata Interchange (XMI)
  - XMI is a standard syntax for the Exchange of Models

The OMG Model Stack



The Meta Object Facility (MOF) distinguishes four levels:

- M0 is the basic data, the lifeblood of the business
  - the customer name "Peter Miller", the price "$291.70".
- M1 is the metadata: schemas and interfaces describing the structure of the data.
  - a table customer with a name column
- M2 is the meta-model, or the "IT language" - specifying the concepts of the modelling language
  - "A relational database has tables, each table has zero or more columns".
- M3 is the MOF specification itself, which allows us to draw the boxes-and-arrows of UML