

Meta-Modeling

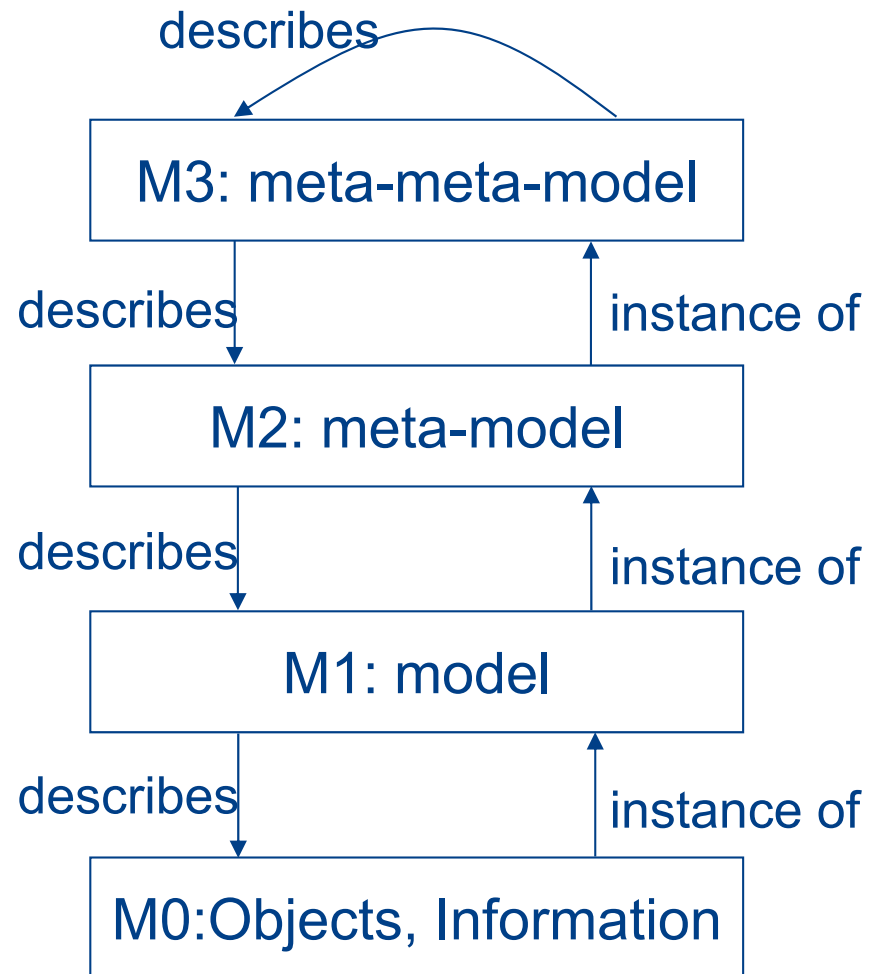
Prof. Dr. Knut Hinkelmann



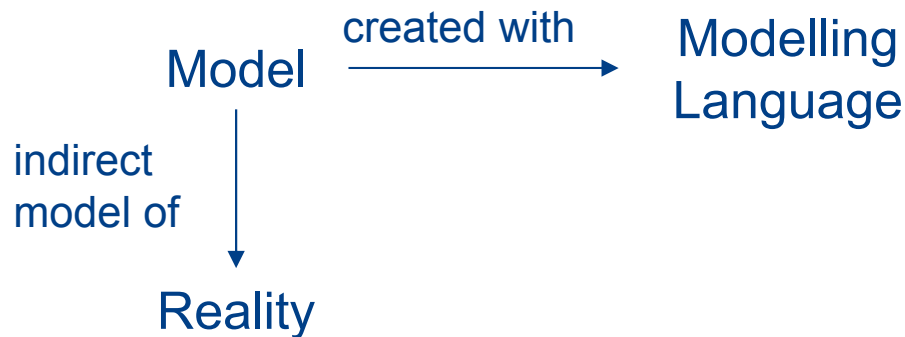
Meta-Modeling

- A meta-model defines the languages from which to form a model.
- A meta-model is a model that defines the language for expressing a model.
- A meta-model specifies the ***abstract syntax*** of a modeling language. It can be understood as the representation of the class of all models expressed in that language
- A ***model*** conforms to a language whose abstract syntax is represented by a ***metamodel***.

The OMG Model Stack

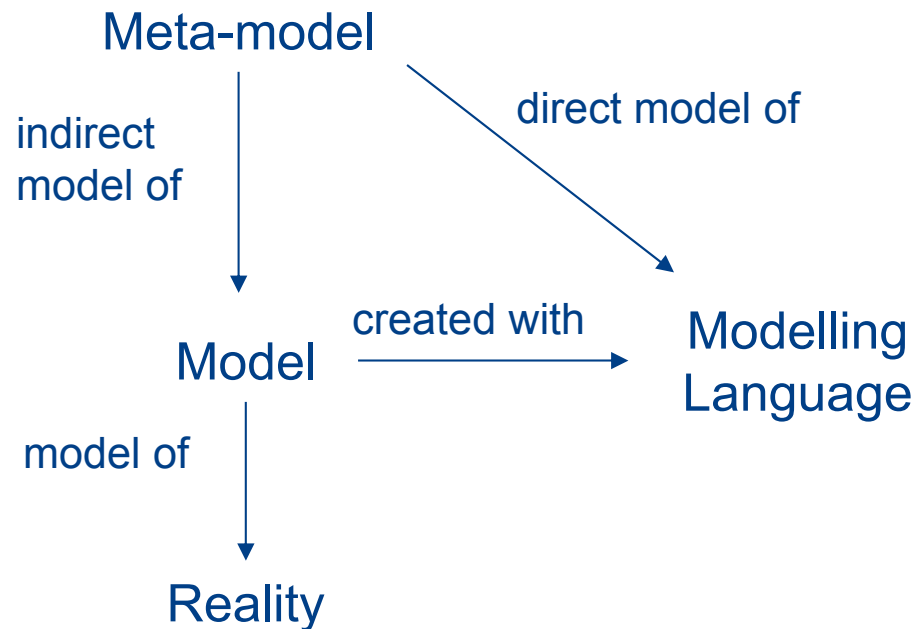


Modelling Language



- 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



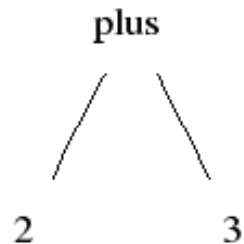
A meta-model defines the modelling language, i.e. the building blocks that can be used to make a model. It defines the

- ◆ object types that can be used to represent a model
 - ◆ relations between object types
 - ◆ attributes of the object types
 - ◆ meaning of the object types
 - ◆ rules to combine object types and relations
- The meta-model is the abstract syntax, the modeling language is the concrete syntax.

Abstract vs. Concrete Syntax

Abstract Syntax

- Deep structure of a language.
- What are the significant parts of the expression?
- Example: a sum expression has its two operand expressions as its significant parts



Concrete Syntax

- Surface level of a language.
- What does the expression look like?

Example: *the same* sum expression can look in different ways:

<code>2 + 3</code>	<code>-- infix</code>
<code>(+ 2 3)</code>	<code>-- prefix</code>
<code>(2 3 +)</code>	<code>-- postfix</code>
<code>bipush 2</code> <code>bipush 3</code> <code>iadd</code>	<code>-- JVM</code>
<code>the sum of 2 and 3</code>	<code>-- English</code>

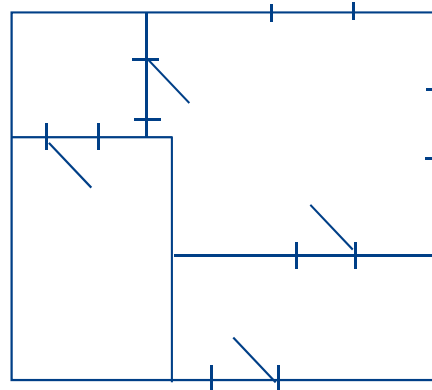
Model and Meta-Model in Architecture

real object



house

model



architect's drawing
(plan)

meta-model
(abstract syntax)

object types:

- wall
- door
- window

rules:

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

modeling language
(concrete syntax)

object types:

— wall

+ / + door

+ — + window

Illustration: Meta-model and Model for Processes

Meta-model:

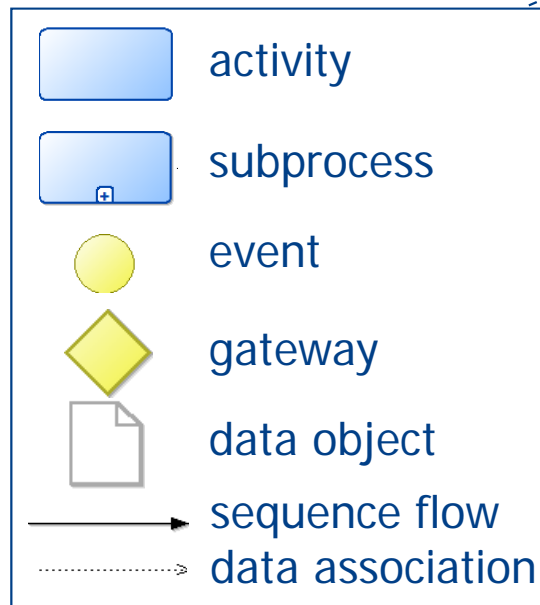
A process model consists of object types for

- «activity» and «subprocess»,
- «events», «gateways»
- «data object»
- «sequence flow» and «data association».

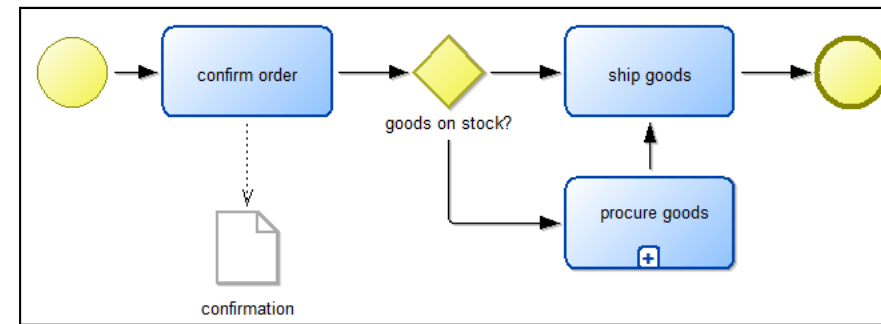
The elements have attributes and there are rules how the elements can be combined.

Modeling Language:

Syntax (appearance) and semantics of meta-model elements



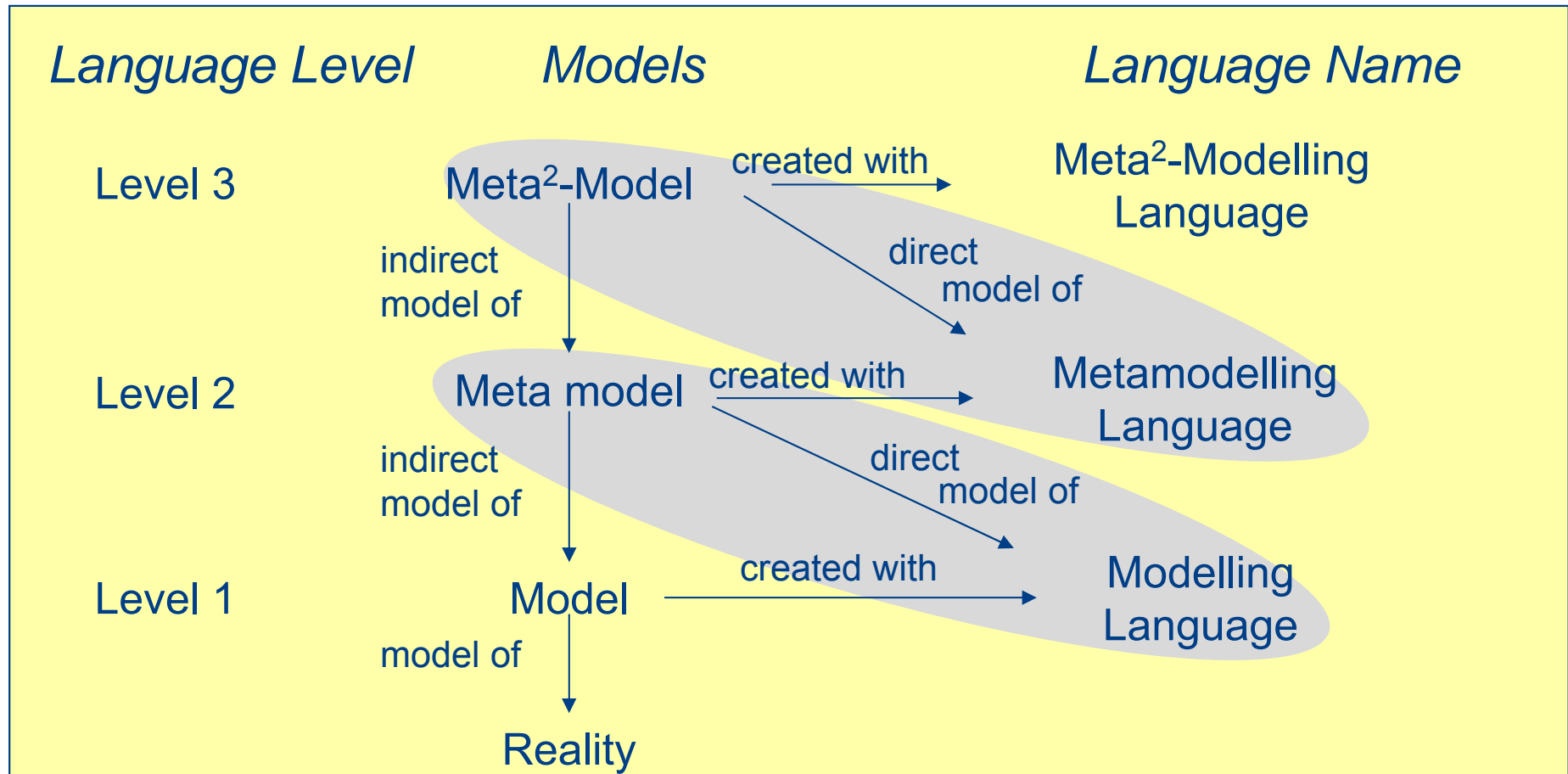
Model:



A model contains instances of the object types defined in the meta-model, according to the concrete syntax of the modeling language. The object „confirm order“ represents a real entity; it is an instance of the object type "activity"

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

Modeling a Meta-Model

- OMG uses UML Class Diagrams for Meta-Modeling
- Example: Business Motivation Meta-Model

