



Some Thoughts about the Realization of Modelling Methods

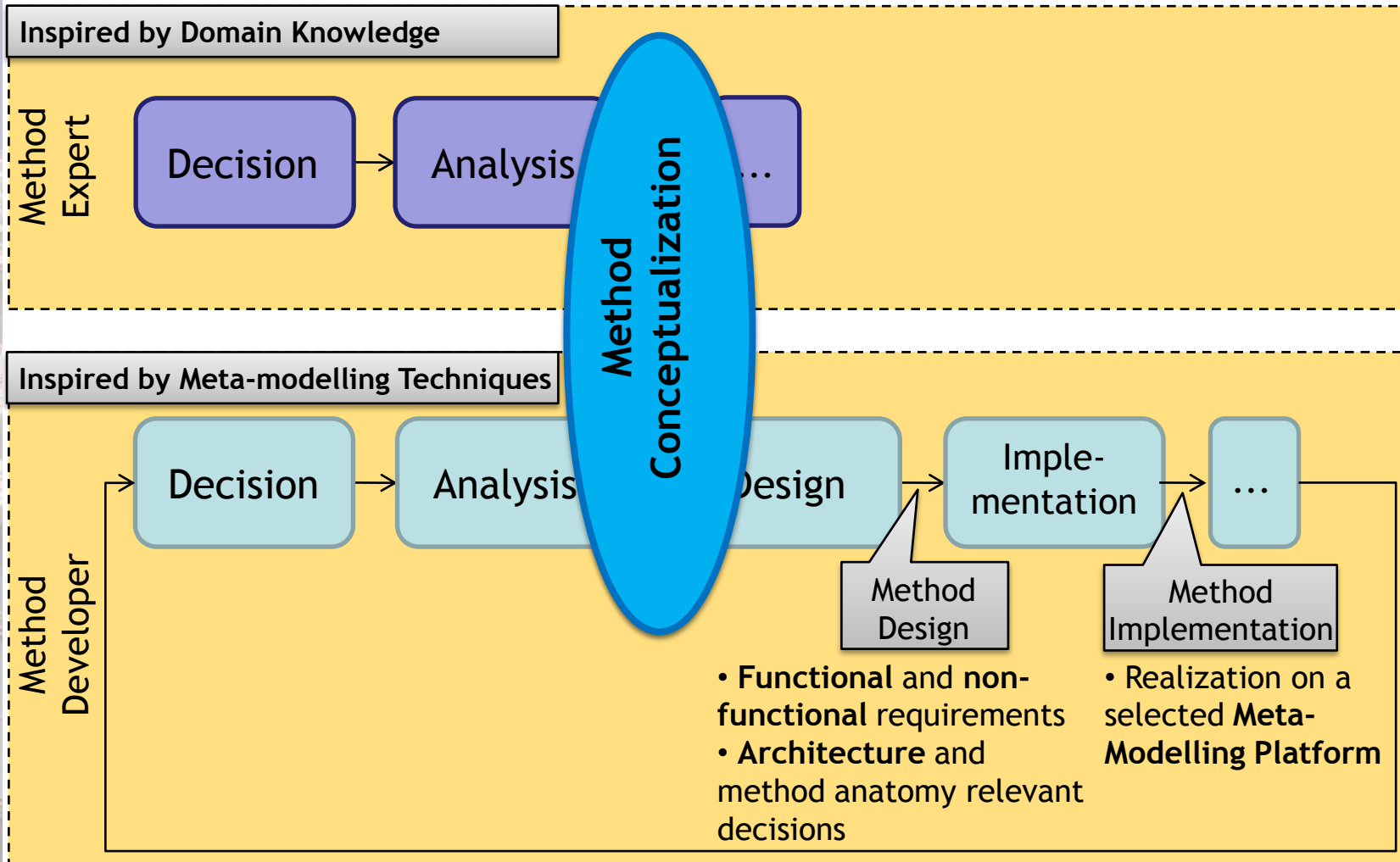
Alexander Bergmayr, Robert Woitsch



Observation

- When the realization of a modelling method is expected to result in an **application software**, a domain expert's (i.e., method expert) viewpoint need to be “augmented” with the viewpoint of a software developer (i.e., method developer)
- Typically, a method expert **rarely considers** design, implementation or deployment relevant artefacts when “conceptualizing” a modelling method
- A method developer on the other hand is usually **not an expert** in the domain that is addressed by a certain modelling method

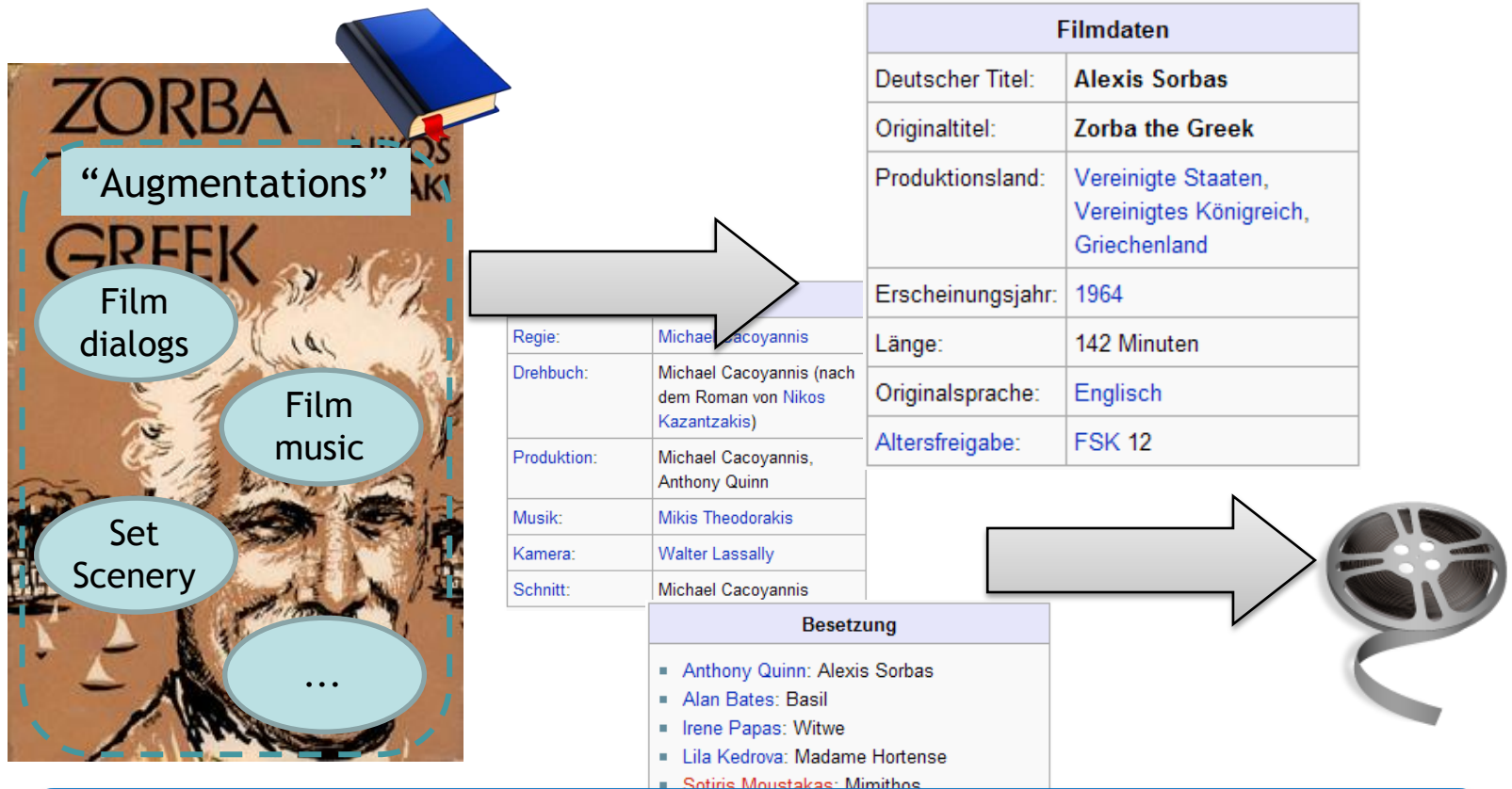
Realizing a Modelling Method (Simplified view)



Method Conceptualization

- Capturing of fundamental **concepts**, **relationships** in between and **properties** adhering to them, usually obtained through the analysis of a selected domain
- Descriptions of such conceptualizations varies depending on the **addressed audience** (different expectations)
 - End User, Modeler
 - Developer
- From a development perspective, a method conceptualization needs to be **formal enough** to enable developer continue along the life-cycle
 - A **model** of the method (language) that facilitates a **coherent view** on the core concepts involved

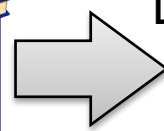
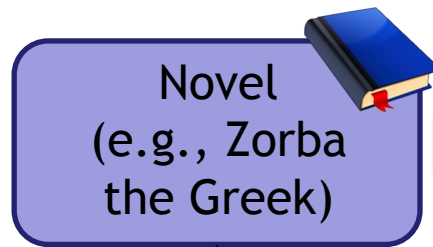
The “Book to Movie” Metaphor ...



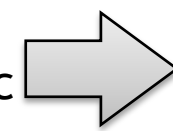
We learned that the method descriptions from method experts need typically to be augmented with models that satisfy the expectations of method developer

... Continue the Metaphor

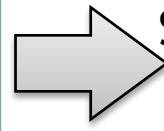
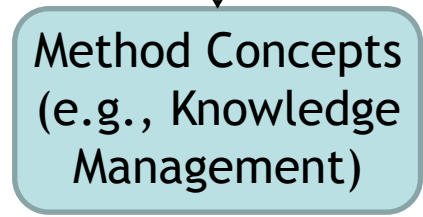
“Book to Movie”
Metaphor



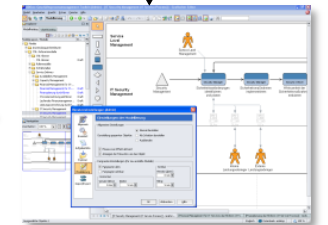
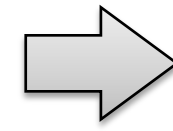
Light
Set scenery
...
Film music
Film dialogs



Modelling Method
Development

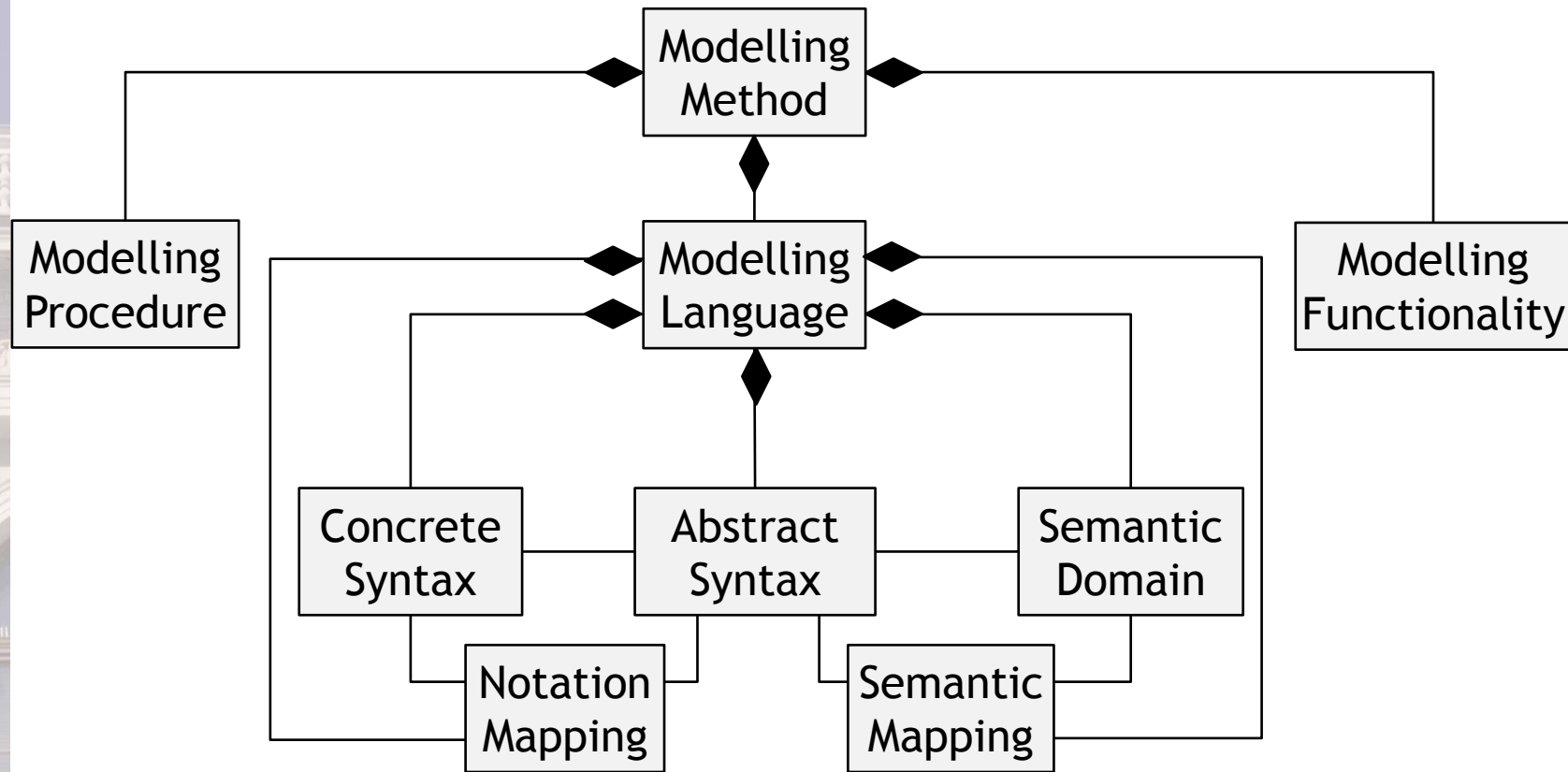


Abstract
Syntax
Concrete
Syntax
Model ...
Manipulation
(e.g., in terms of
dedicated
functionality)



Tool-supported
Modelling Method

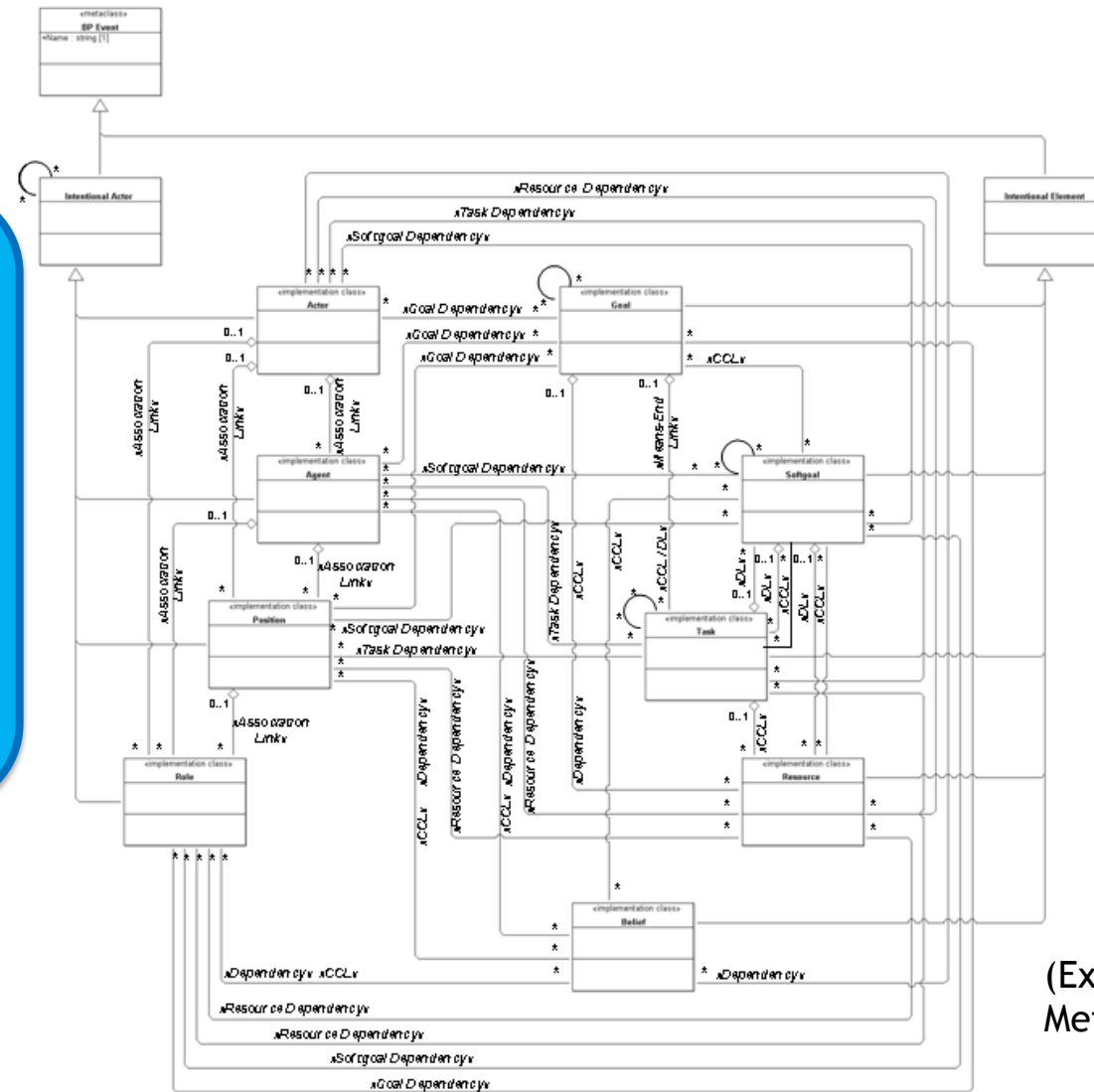
Ingredients for Modelling Methods (Simplified Version)



In accordance to Karagiannis, D., Kühn, H.: „Metamodelling Platforms“. In Bauknecht, K., Min Tjoa, A., Quirchmayer, G. (Eds.): Proceedings of the Third International Conference EC-Web 2002 - Dexa 2002, Aix-en-Provence, France, September 2002, LNCS 2455, Springer, Berlin/Heidelberg, p. 182 ff.

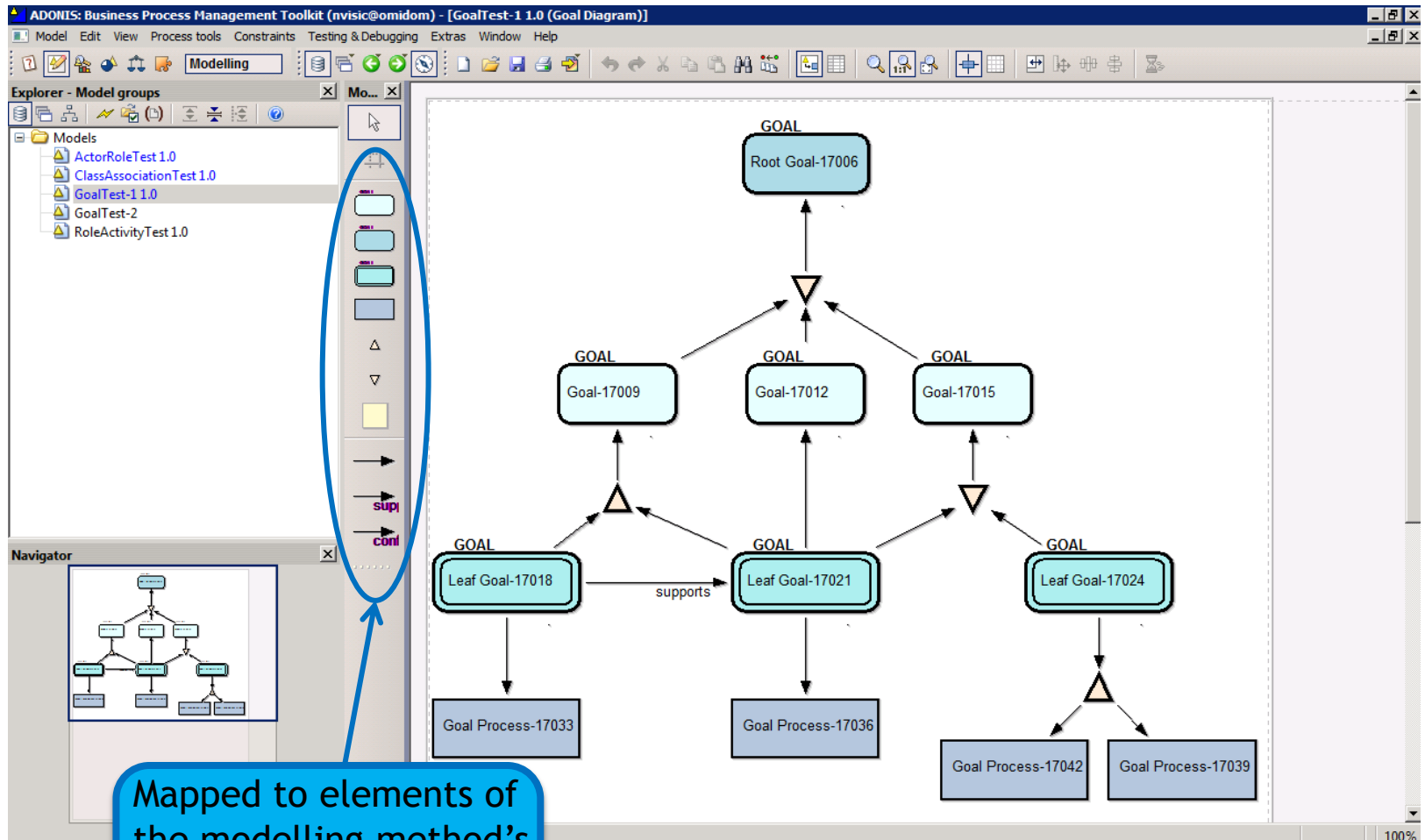
Modelling Language: Abstract Syntax in terms of a Metamodel

- Selection of an appropriate language or formalism to describe the Metamodel
- Description of method concepts in terms of elements from the selected language or formalisms (i.e., a mapping to their notation and meaning)



(Excerpt from iStar Metamodel)

Modelling Language: Concrete Syntax for Tool Realisation



Modelling Language: Semantics for Syntactic Elements

- Aspects of a modelling language that **cannot** be described with mechanisms for syntax definitions are pushed into the semantics area¹
- Operational Semantics
 - The basic interest is on the “execution” of models based on an abstract machine
 - Eg., Interpreter for Petri-Nets or Statecharts
- Denotational Semantics
 - The denotation is expressed through a mapping of syntactic constructs to constructs of a commonly accepted domain that is assumed to be well understood
 - Eg., Control-Flow of BPEL denoted in terms of Petri-Nets

1) cf., David Schmidt, Denotational Semantics: A Methodology for Language Development, 1986 10

Research Questions

- What are appropriate (modelling) languages for describing the components of a modelling method, i.e., a language specific to the domain of modelling method development?
- What are specifics of a development life cycle for modelling methods (in contrast to typical SDLC) that should be communicated to a “modelling community”?
- What are required improvements of current approaches, guidelines or practices and existing technologies for the development of modelling methods?



Thank you for
your attention!