Fourth International *i Workshop,
Hammamet, 7th-8th of June 2010**

*i** on ADOxx®: A Case Study

… an Open Models Project!

Authors:
Margit Schwab, Dimitris Karagiannis, Alexander Bergmayr

universität wien
Faculty of Computer Science

DKE

Agenda

- **The Open Model Initiative**
  http://www.openmodels.at
- **What is ADO*xx*®**
- **The *i** Method on ADO*xx*®**
- **Recent Work**

The Open Models Initiative - OMI

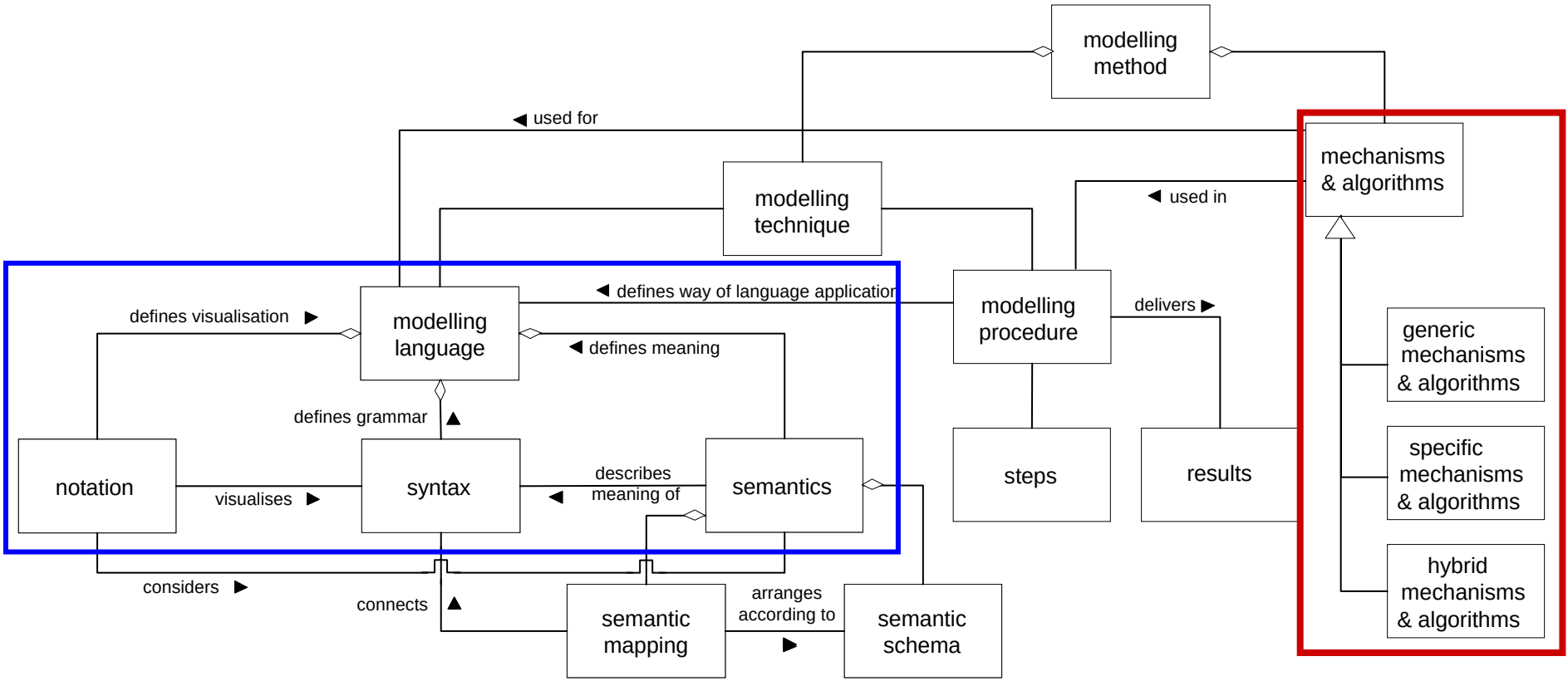
Vision "Models for everyone"*

The Open Source Community for Modellers where models are operational knowledge!

- ***Archetype: Open Source Community***
  - Open Source is a development method for software with features like:
    - Free availability of the source code
    - Free convertibility of the source code
    - Free copying of source code and software
    - Collaborative development
  - Similar communities for software developers: SourceForge, Eclipse

* In this context conceptual models are meant as mathematical graphs („Open Model Graph");

Starting Point for Meta-Modelling / the ‘Conceptualization’: The Used Meta-Modelling Framework



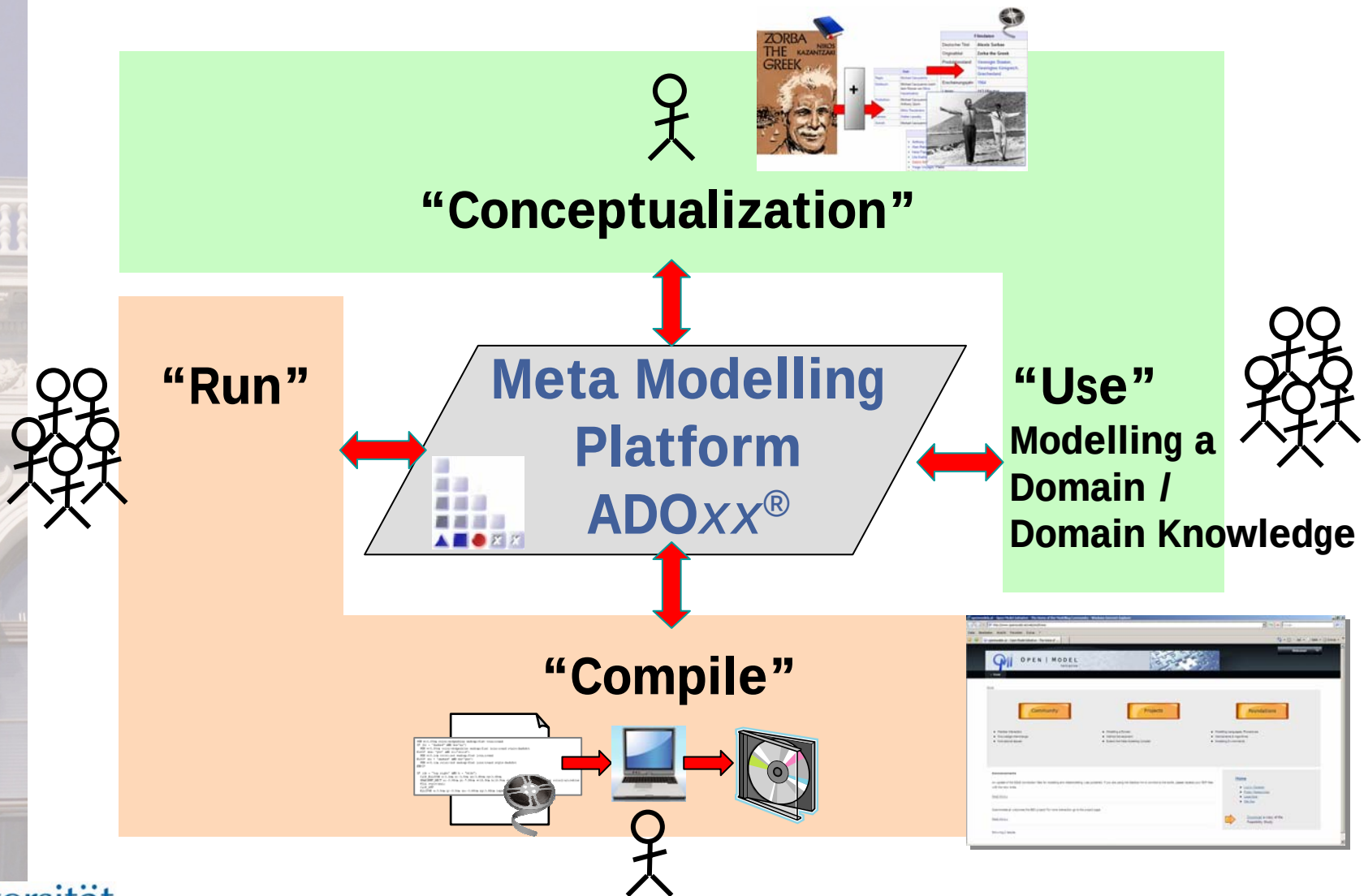
Karagiannis, D., Kühn, H.: „Metamodelling Platforms“. In Bauknecht, K., Min Tjoa, A., Quirchmayer, G. (Eds.): Proceedings of the Third International Conference EC-Web 2002 – Dexa 2002, Aix-en-Provence, France, September 2002, LNCS 2455, Springer, Berlin/Heidelberg, p. 182 ff.

The ‘Processes’ in OMI

Method Engineer(s)

Method Engineering Phases	Conceptualization	Implementation	Deployment
Output & Tool	**Output: Metamodel Specification** Wiki, "Pen and Paper", Scientific Publication, Standards	**Output: Modelling Tool Prototype** ADOxx® Development Environment	**Output: Delployment of Modelling Tool & Services** ADOweb®
Infrastructure	**Personal Computing Environment**	**Option 1: Remote application environment** using Windows Server2008 RemoteApps, CITRIX, etc. **Option 2: ADOxx® LAN/WAN Deployment**	**Web Deployment within ADOweb® Service Framework** Apache Tomcat, Modelrepository DB, Modelling Services

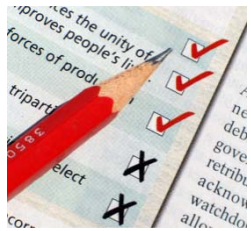
The Open Models Initiative - OMI



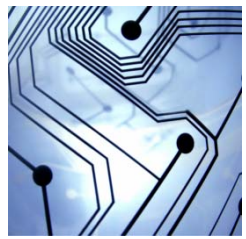
The Open Models Initiative - OMI

OPEN | MODEL Initiative

Open Model Community



Open Model Projects



Open Model Foundations

bmvit

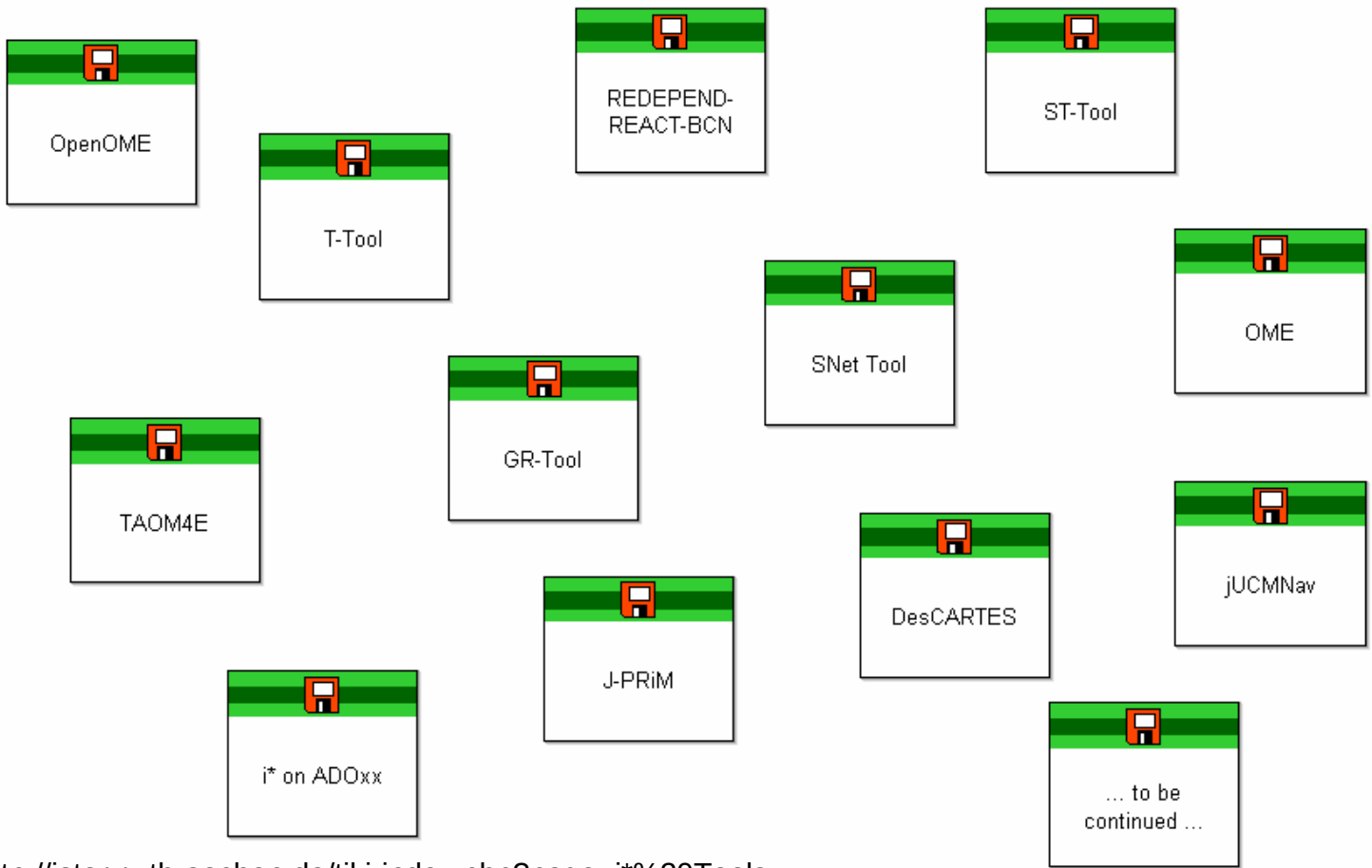
Promoted by the Federal Ministry for Transport, Innovation and Technology

universität wien

Faculty of Computer Science

DKE

Diversity of *i** Tools (selected)



http://istar.rwth-aachen.de/tiki-index.php?page=i*%20Tools

Agenda

- **The Open Model Initiative**
  http://www.openmodels.at
- **What is ADO*xx*®**
- **The *i** Method on ADO*xx*®**
- **Recent Work**

Metamodelling Platforms
Criteria

What are the criteria?

... which make a modelling platform to a metamodelling platform?

Essential Functional Requirements

- flexible metamodelling capabilities instead of fixed metamodels,
- -> the metamodel can be freely defined,
- -> easy adaptation of the metamodel to the problem under consideration, and
- -> further extensible.

[Kühn, Karagiannis, "Metamodelling Platforms", 2002]

Other Requirements

- Powerful graphical editor
- DB based
- Repository
- Multi-user
- Web-enabled
- ...

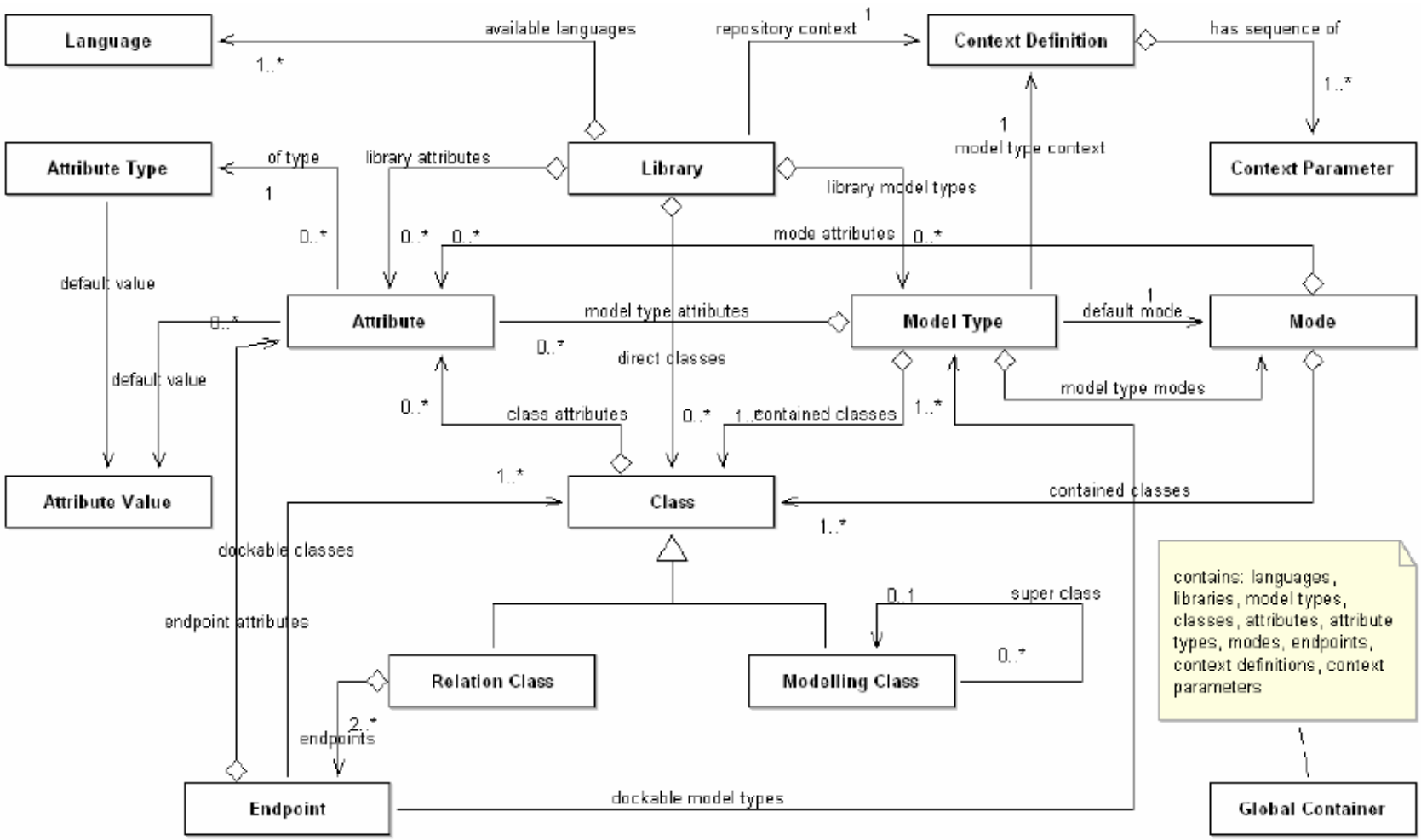
Metamodelling Platforms

Overview – Selected Platforms

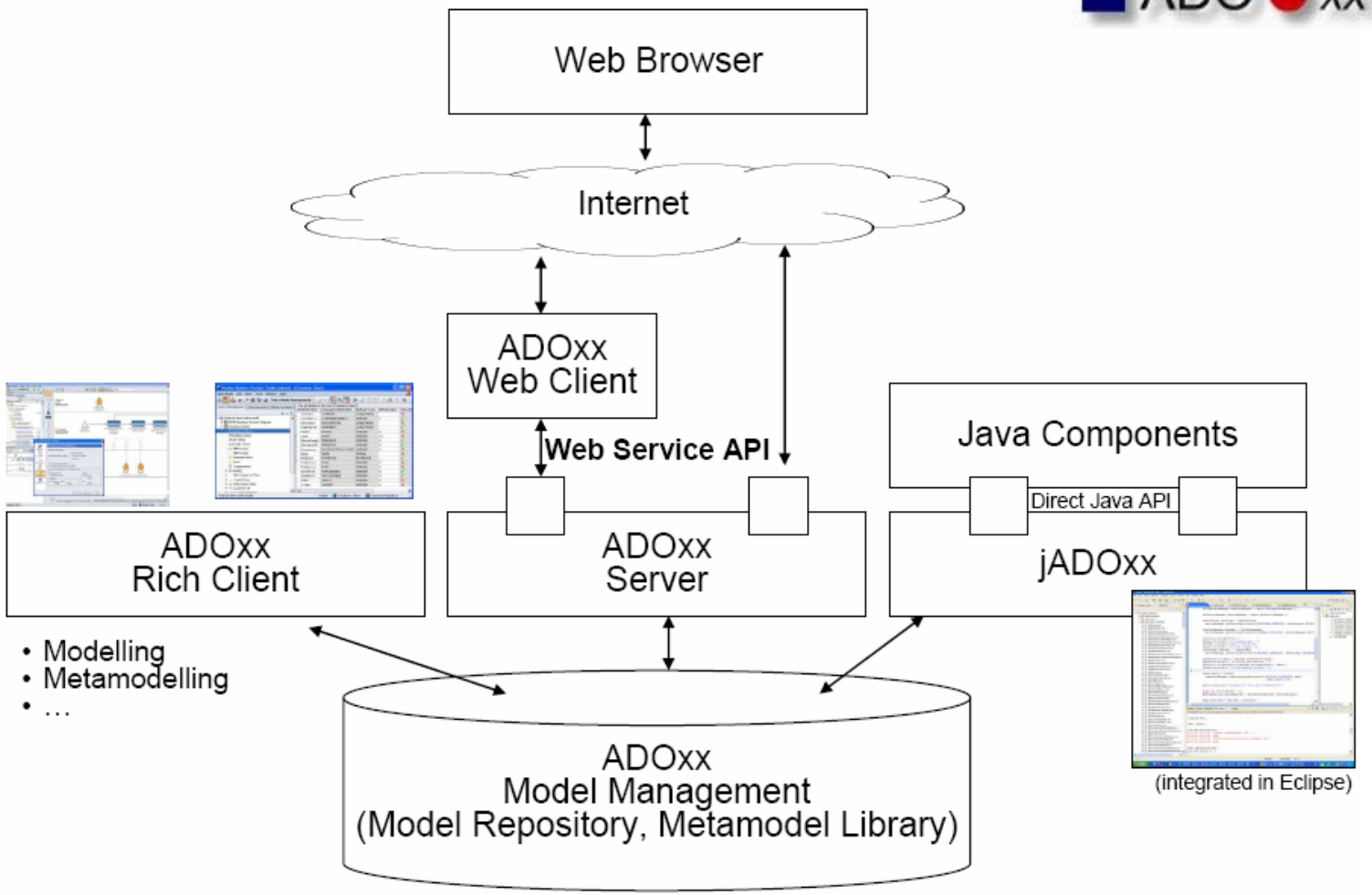
Platform Name	Enterprise	Environment
ADOxx	BOC ITC, AT	**Conceptual and Domain Specific Modelling** ADOxx is a metamodelling-based development andconfiguration environment to create domain-specific modelling tools. http://www.openmodels.at/c/document_library/get_file?p_l_id=65121&folderId=65129&name=DLFE-2505.pdf
Eclipse GMF	Eclipse Foundation Inc., CA	**Software Development** http://www.eclipse.org/org/foundation/contact.php
Epiwork Computational Modeling Platform	EPIWORK, FP7 EU Project	* based on GLEaM, a discrete stochastic epidemic computational model based on a meta-population approach, * in which the world is defined in geographical census areas connected in a network of interactions by human travel fluxes, * corresponding to transportation infrastructures and mobility patterns. [http://www.epiwork.eu/resources/wp4-computational-modeling-platform/]
INNOVATOR	MID Ltd., DE	A modelling platform for Business Process Management, Object and Structured Softwareanalysis, Object Oriented Design and Data Modelling. In particular applicable for the development of methodologically correct and efficient Model-Driven Architecture scenarios as the platform strictly follows OMG standards and integrates UML 2 profiles for language extentions. http://www.mid.de/Modellierungsplattform-Innovat.innovator.0.html
MetaEdit+	MetaCase, FI	**Domain Specific Modelling** * radically improve development productivity and quality by generating full code directly from models, * firstly for the design the modeling language with MetaEdit+ Workbench and then, * other developers model with the language in MetaEdit+ Modeler. http://www.metacase.com/products.html
MLDesigner/ SatLab	ML Design Technologies, US	* is an integrated platform for modelling and analyzing the architecture, function and performance of high level system designs, * primary domains include Discrete Event, Dynamic Data Flow, and Synchronous Data Flow, * simulation environment for the predictability, productivity, quality of the entire development process and eventual product/system integration. http://www.mldesigner.com/mldesigner/
Oslo	Microsoft, US	**Metadata -based Software Development** * "M" Language Specification is the authoritative source for "M" grammar and syntax, * it contains detailed information about all aspects of the language, * is a language for defining domain models and domain-specific languages (textual DSLs). http://msdn.microsoft.com/library/dd285282
Semantion	Semantic Inc., CA	* provides a solution that replaces monolithic systems with a distributed virtual platform for modeling of any type of system and process, * web-based information management, and * deployment, distributed execution, and analysis of business processes. http://www.semantion.com/products.html
…		

2. 'Conceptualization' for the ADOxx® Platform

Extract of the ADOxx® Meta-Metamodel



The ADOxx® Metamodelling Platform
Deployment Architecture

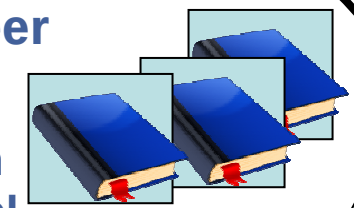


Agenda

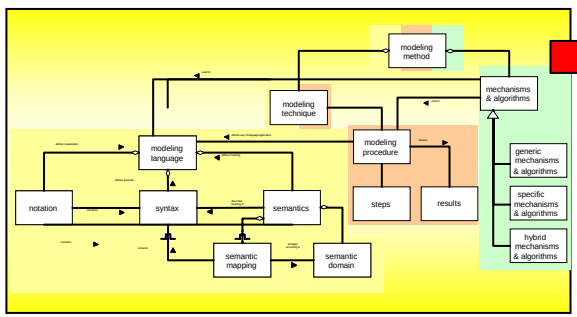
- **The Open Model Initiative**
  http://www.openmodels.at
- **What is ADO*xx*®**
- **The *i** Method on ADO*xx*®**
- **Recent Work**

Method Conceptualization

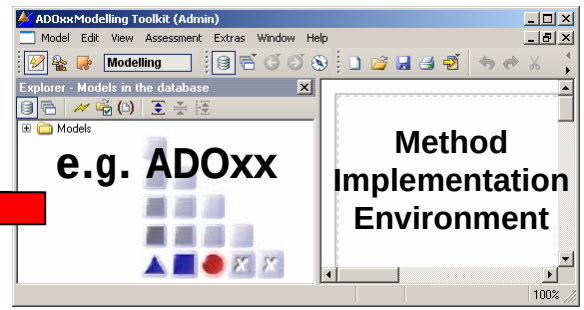
Method Engineer



Method description
on a theoretical level



Conzeptualization of the Method



Validation

“Conceptual Modelling“

Domain User Environment
End User Tool

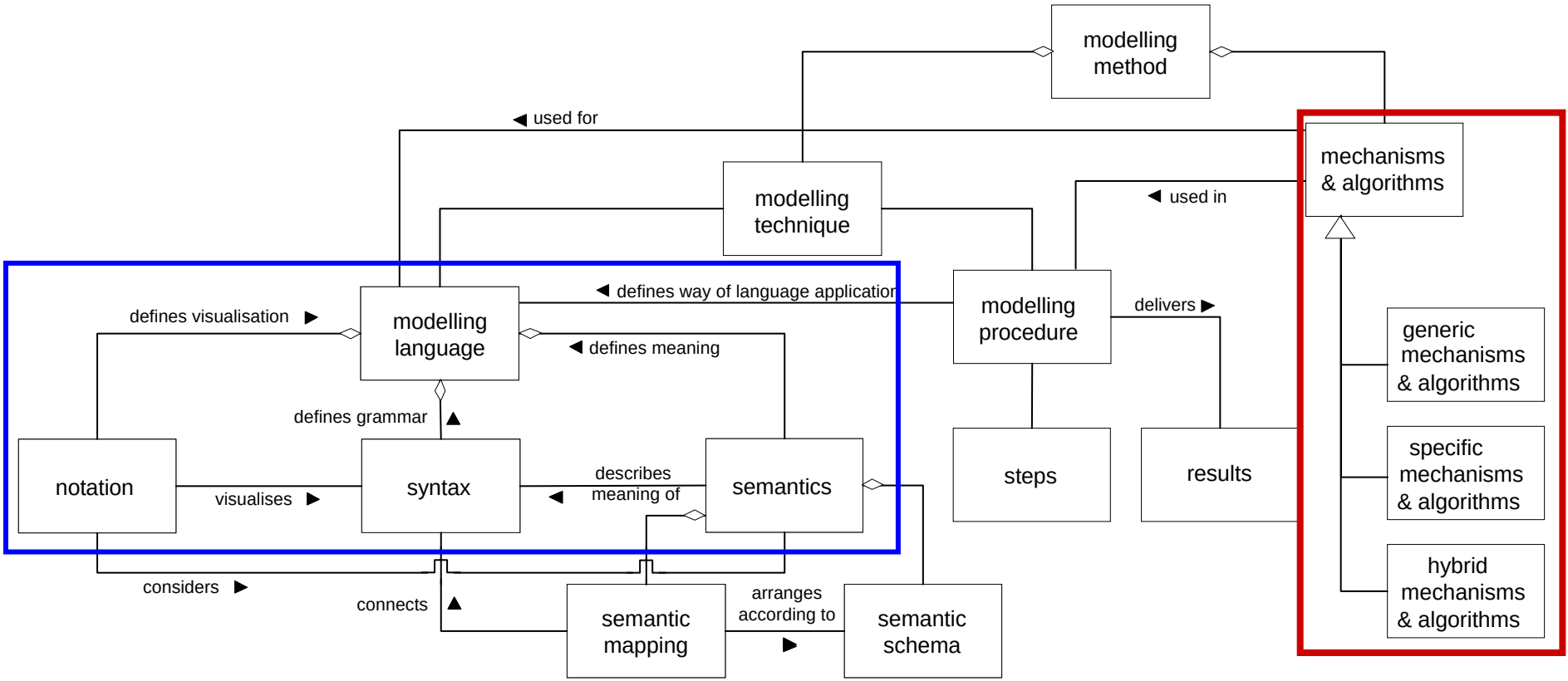
‘Conceptualization’ of the *i** Method

1. Analysing & Studying the *i** Method

What is the *i** Method?

- Method which has been developed to show social relationships for their analysis and design
- In particular helpful to understand complex relationships among actors with strategic intent
- It includes human and IT resources
- Does not: aim to map and design the execution of certain steps in a certain temporal dimension
- Developed 1995 by associate Professor Eric Yu as a PhD Thesis and Professor John Mylopoulos as supervisor, Faculty of Information, University of Toronto

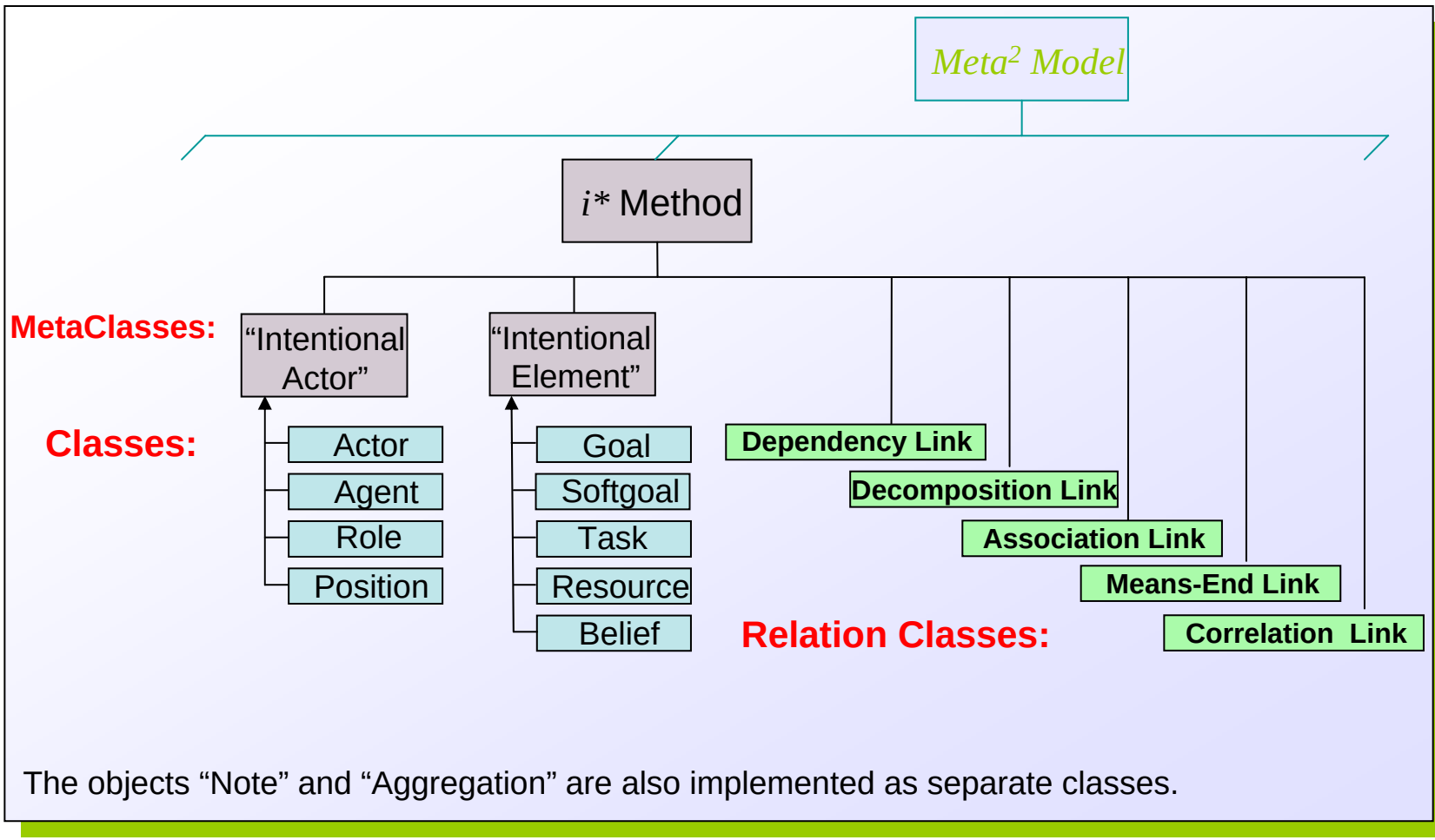
Starting Point for Meta-Modelling / the ‘Conceptualization’: The Used Meta-Modelling Framework



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.

2. 'Conceptualization' for the ADOxx® Platform

Required Classes and Relations of *i**

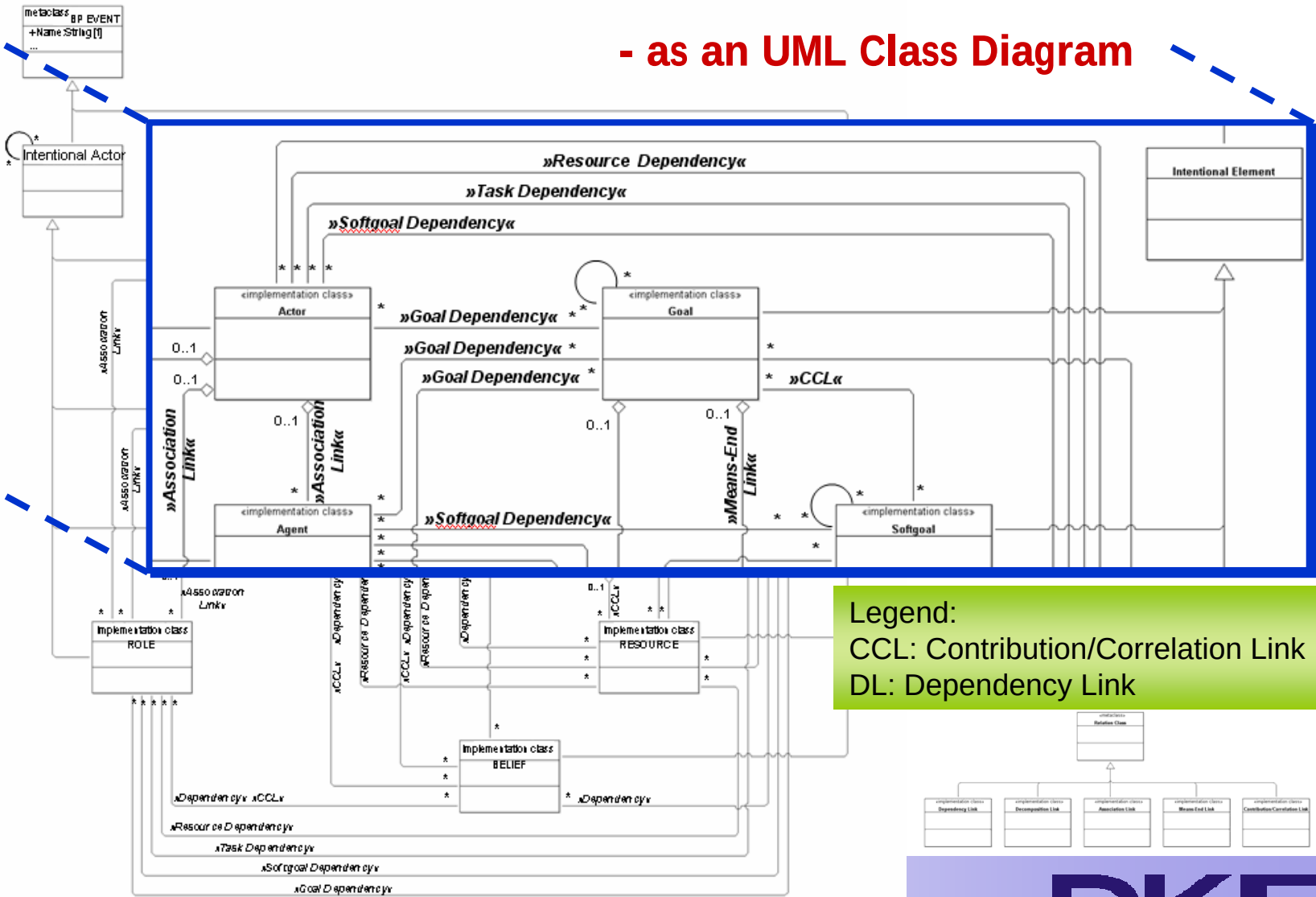


The objects "Note" and "Aggregation" are also implemented as separate classes.

2. ‘Conceptualization’ for the ADO*xx*® Platform

Required Classes and Relations of *i**

- as an UML Class Diagram



Legend:
CCL: Contribution/Correlation Link
DL: Dependency Link

univer wien
Faculty of Computer Science

DKE

2. ‘Conceptualization’ for the ADOxx® Platform

Ascertainment of Notation

Notation -> Graphical representation of objects/relations

Depending on the definition of the respective class – here ***Actor …***


```
GRAPHREP layer:0 sizing:keep-aspect-ratio
  AVAL set-default: "without" b: "Boundary"
  AVAL set-default: "down right" rb: "Representation of boundary"
  AVAL i: "Order"
  AVAL set-default:"x" p:"Referenced actor"
  AVAL sub:"Referenced actor"

    PEN w:0.05cm color:dodgerblue endcap:flat join:round
    IF (bl = "dashed" AND ka="no")
      PEN w:0.05cm color:dodgerblue endcap:flat join:round style:dashdot
    ELSIF (ka= "yes" AND bl="solid")
      PEN w:0.1cm color:red endcap:flat join:round
    ELSIF (bl = "dashed" AND ka="yes")
      PEN w:0.1cm color:red endcap:flat join:round style:dashdot
    ENDIF

    IF (rb = "top right" AND b = "with")
      CLIP_ELLIPSE x:3.0cm y:-3.0cm rx:3.88cm ry:3.88cm
      GRADIENT_RECT x:-3.88cm y:-7.88cm w:10.8cm h:10.8cm style:downdiag color1:white
color2:aliceblue
      FILL style:null
      CLIP_OFF
      ELLIPSE x:3.0cm y:-3.0cm rx:-3.88cm ry:3.88cm layer:0 sizing:keep-aspect-ratio

    ELSIF (rb = "down right" AND b = "with")
      CLIP_ELLIPSE x:3.0cm y:3.0cm rx:3.88cm ry:3.88cm
      GRADIENT_RECT x:-3.88cm y:-3.88cm w:10.8cm h:10.8cm style:downdiag color1:white
color2:aliceblue
      FILL style:null
      CLIP_OFF
      ELLIPSE x:3.0cm y:3.0cm rx:-3.88cm ry:3.88cm layer:0 sizing:keep-aspect-ratio
    ENDIF
...
```

PEN w:0.05cm
color:dodgerblue
endcap:flat join:round

… in ADOxx® the **Notation** of the **Class** is realized in the **“Graphrep“.**

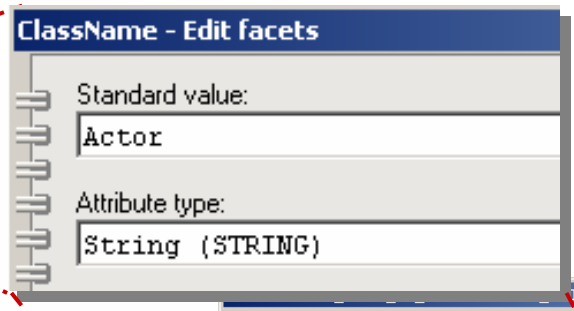
2. 'Conceptualization' for the ADOxx® Platform

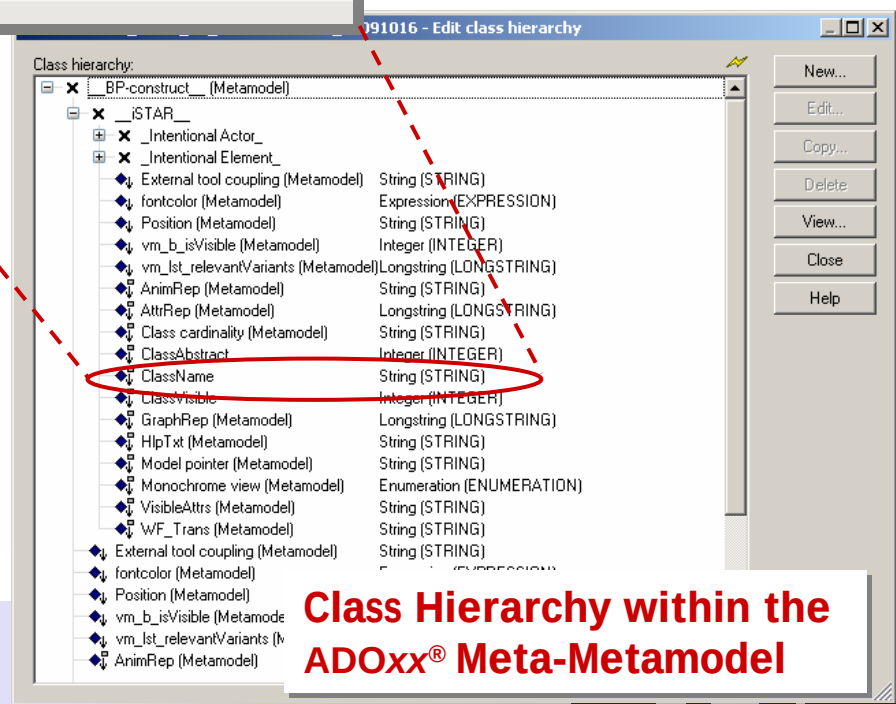
Ascertainment of Syntax

Syntax -> Object and relation definition

… the **Syntax** of the **Class** is realized depending on **definition** of the **method** and the given Syntax of the ADOxx® Meta-Metamodel in the **"Attrep".**

```
NOTEBOOK
CHAPTER "General"
ATTR "Name"
ATTR "Order"
ATTR "Boundary"
ATTR "Representation of boundary"
ATTR "Boundary lines"
CHAPTER "Description"
ATTR "Description"
ATTR "Comment"
ATTR "Representation of name"
ATTR "Font size"
CHAPTER "Further Details - Benefits"
ATTR "Referenced actor"
ATTR "Display name and reference"
ATTR "Key actor"
ATTR "Main skills and competence"
ATTR "Responsibility" lines:5
CHAPTER "Furhter Details - Constraints"
ATTR "Constraints" lines:5
ATTR "Costs"
GROUP "Availability"
ATTR "Available from"
ATTR "Available till"
ENDGROUP
```



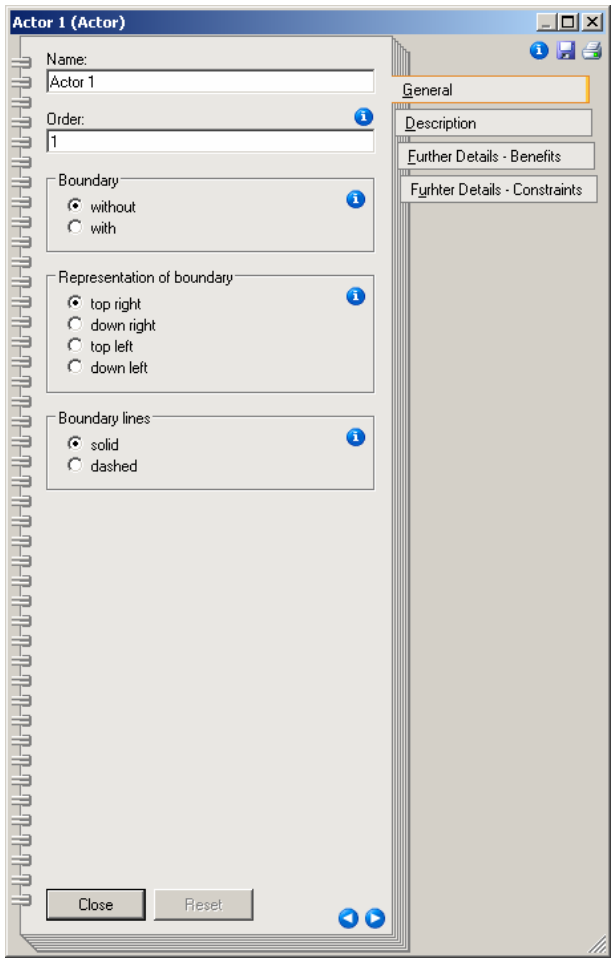


Class Hierarchy within the ADOxx® Meta-Metamodel

2. ‘Conceptualization’ for the ADOxx® Platform

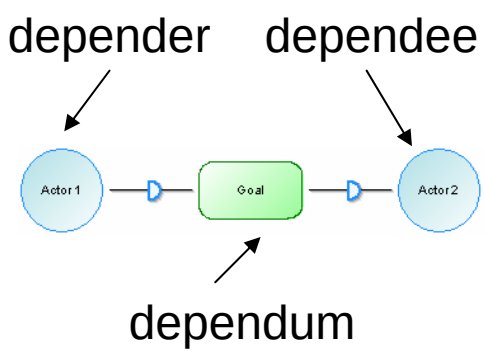
Ascertainment of Semantics

Semantics -> Object and relation characteristics definition

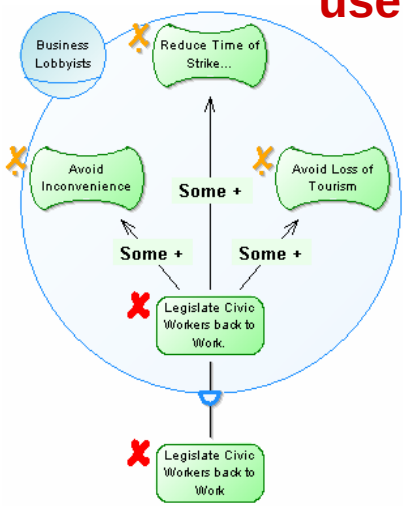


… the **Semantics** of the **Class** are expressed by the **values of the defined Class Attributes** and by the respective **use** depending on the rules as determined by the **method developer**.

‘rules’



‘use’



Horkoff, J.; Schwab, M.; June 2009

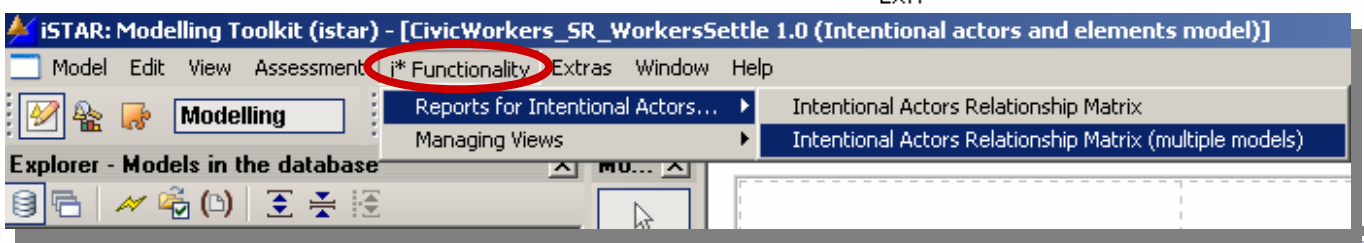
‘Conceptualization’ for the ADOxx® Platform

Optional: Ascertainment of Mechanisms & Algorithms

- **AdoScript** is the macro language of the ADOxx® platform
- it is procedural

- Allows **easy access** to almost all functionalities of the ADOxx® platform, like
  - New menus
  - Model-specific cardinality checks
  - Realization of new interfaces
  - Specific program calls
  - etc.

```
#----------------------------------------
#
# ADOxx(R) iSTAR
#
# University of Vienna, DKE, 2009
#
#----------------------------------------
SETL idlist_modelids:("")

IF (oneModelOnly = 1)
{
#--------------------------------
#--Check if model is loaded
#--------------------------------

 SEND "GET_ACTIVE_MODEL" to:"Modeling" answer:modelid
 IF (modelid = "")
 {
  CC "AdoScript" INFOBOX (g_str_ui_IAR_matrix_info1)
  EXIT
 }
#--------------------------------
#--Check if active model is of type Strategic Dependency Model
#--------------------------------

 SETL id_model:(VAL modelid)
 CC "Core" GET_MODEL_INFO modelid:(id_model)

 IF (modeltype != mod_type8)

 {
  CC "AdoScript" INFOBOX (g_str_ui_IAR_matrix_info2)
  EXIT
```



Agenda

- **The Open Model Initiative**
  http://www.openmodels.at
- **What is ADO*xx*®**
- **The *i** Method on ADO*xx*®**
- **Recent Work**

‘Conceptualization’ for the ADO*xx*® Platform

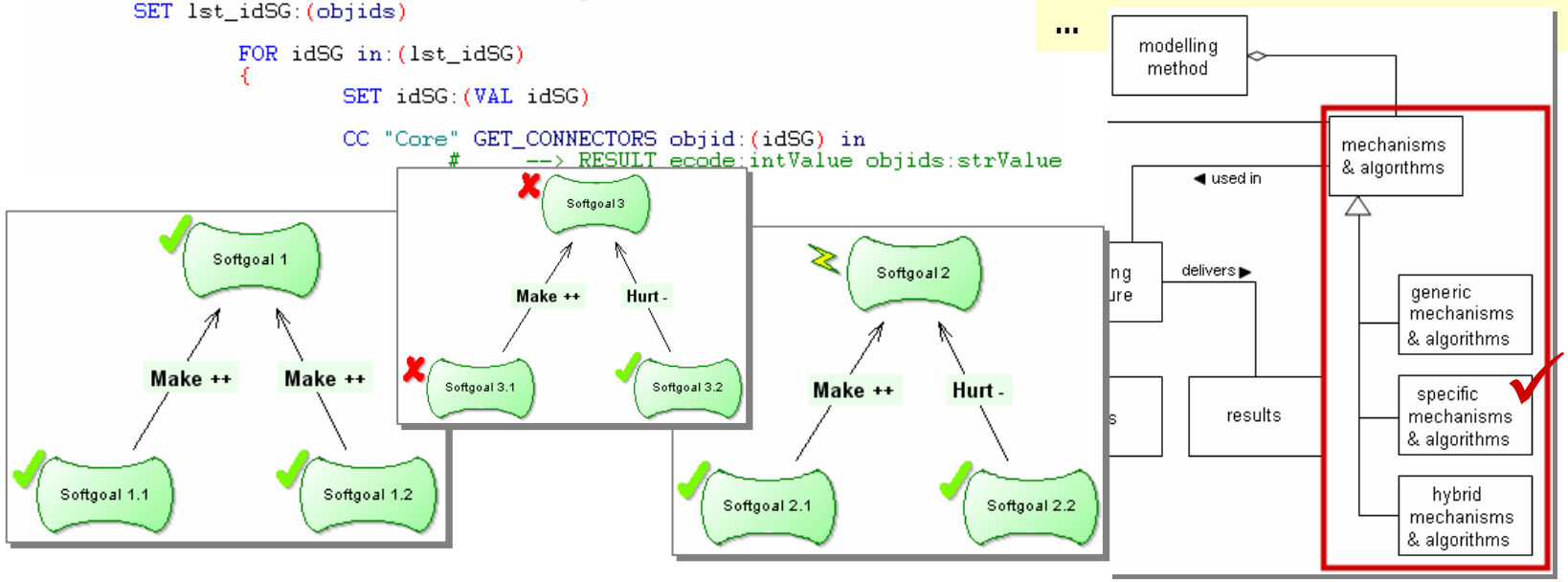
Resent Work: Algorithm for Analyzing Interdependency Graphs

Implementing the rules for the evaluation of interdependency graphs by means of AdoScript.

```
CC "Core" GET_MODEL_INFO modelid:(int_modid)
        #     --> RESULT ecode:intValue  modelname:strValue

CC "Core" GET_CLASS_ID classname:(g_Softgoal)
        #     --> RESULT ecode:intValue classid:intValue isrel:intValue
        SET idClassSG:(classid)

CC "Core" GET_ALL_OBJS_OF_CLASSID modelid:(int_modid) classid:(idClassSG)
        #     --> RESULT ecode:intValue objids:list
        SET lst_idSG:(objids)

                FOR idSG in:(lst_idSG)
                {
                        SET idSG:(VAL idSG)

                        CC "Core" GET_CONNECTORS objid:(idSG) in
                                #     --> RESULT ecode:intValue objids:strValue
```

Applicable Propagation Rules

W+ equals W+
W- equals W-
W+ or W+ results in ?
W- and any value results in ?
...



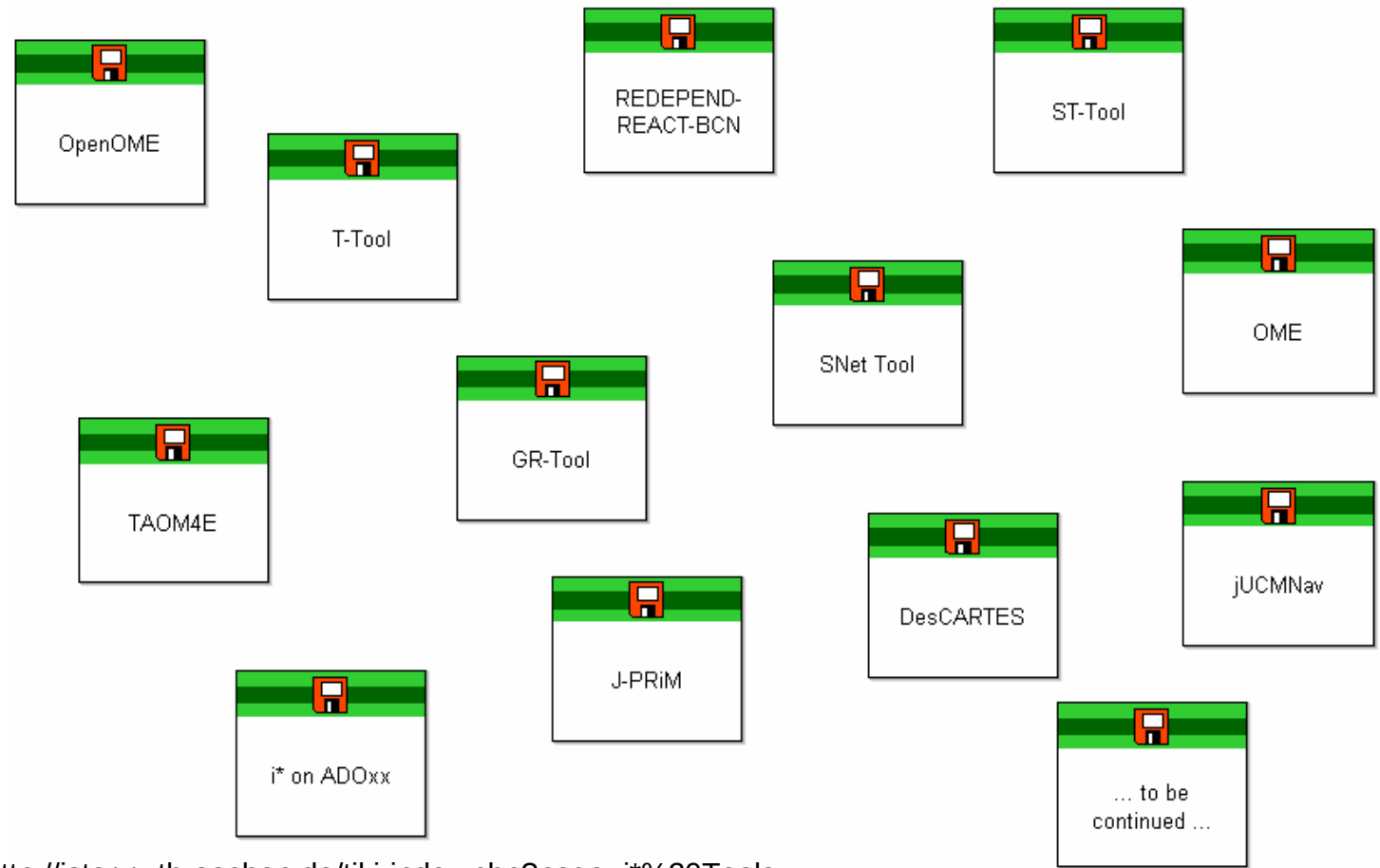
Examples for “automatic“ label propagation

[Chung, Nixon, Yu, Mylopoulos, „Non-Functional Requirements in SE, p76 / p79]

Recent Work: The *i** Model Transformation with iSTARML

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Cooperation Project

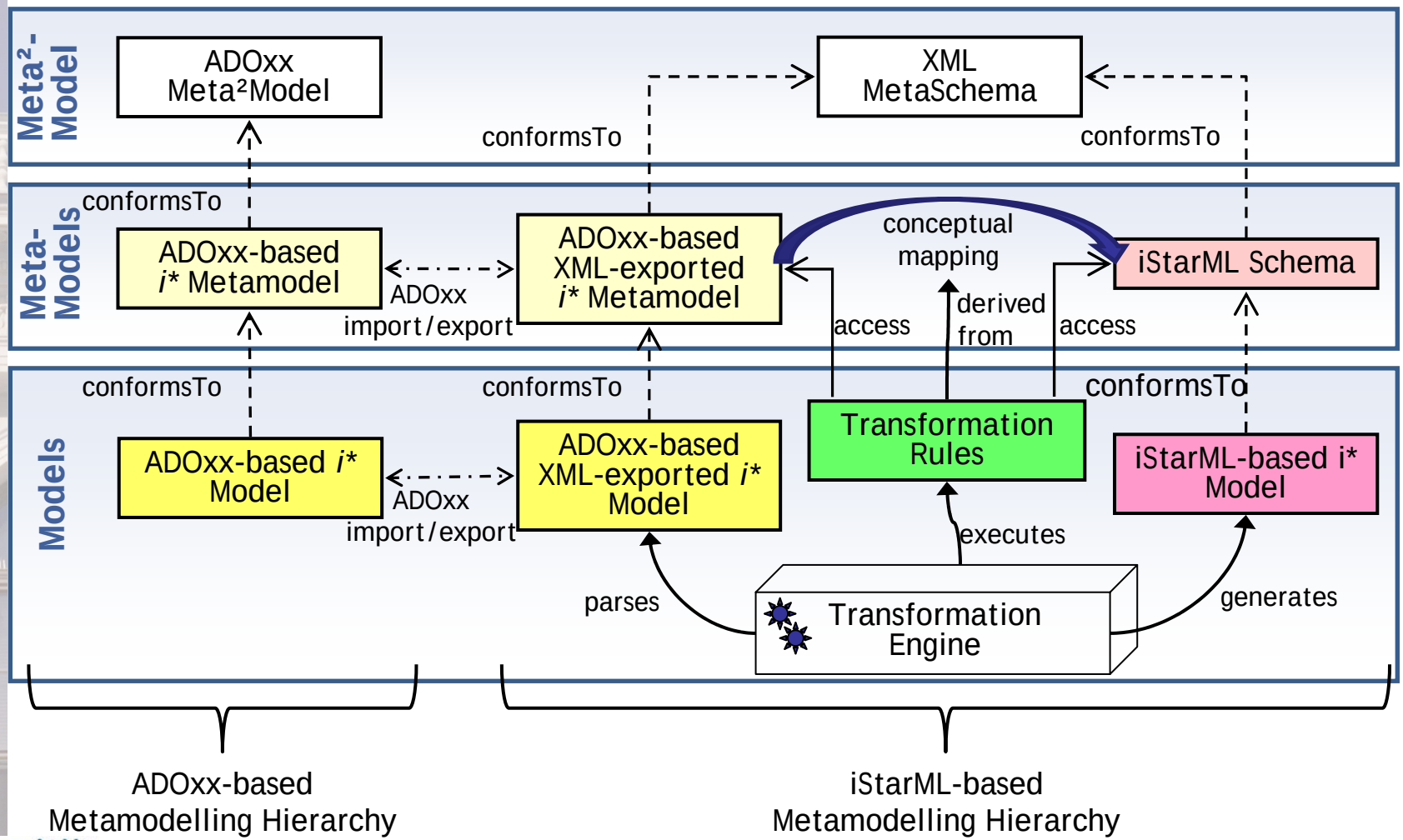


http://istar.rwth-aachen.de/tiki-index.php?page=i*%20Tools

Recent Work: The *i** Model Transformation with iSTARML

Cooperation Project with UPC BNC

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH



universität wien

Faculty of Computer Science

DKE

Recent Work: The *i** Model Transformation with iSTARML Different Modelling Language Definition Formalisms

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Cooperation Project

- **One may observe that different metamodelling platforms rely on different formalisms utilized for the definition of modelling methods**
  - These formalisms, i.e., a platforms meta-metamodel is usually built into them
- **Addressing the definition of a modelling language's abstract syntax most platforms provide** different formalisms **to achieve this task**
  - The utilization of different formalisms hampers the **exchange of models,** e.g. concrete *i** model, corresponding to a modelling language, like *i**, realized with them
  - The problem is that a language's model, i.e., metamodel can be interpreted by a platform only if the formalism used to express the metamodel is known

Recent Work: The *i** Model Transformation with iSTARML Different Modelling Language Definition Formalisms

Model Processing by applying appropriate Mechanisms & Algorithms

- Representation of models in machine-interpretable form by the implementation in a metamodelling platform
- Doing this, appropriate formalisms not only for the representation but also for the processing of the models are necessary
- Formalisms providing the description of the latter are essential to express "platform functionality" dealing for example with:
  - Model Analysis
  - Model Simulation
  - Model Integration
  - Model Comparison
  - Model Transformation
  - **Model Exchange**
  - ...

XML-Technologies to realize the transformation rules

```
<template match="INSTANCE[@class='Actor'] |
                 INSTANCE[@class='Agent'] |
                 INSTANCE[@class='Role'] |
                 INSTANCE[@class='Position']">
  <element name="actor">
    <attribute name="id"><value-of select="@id" />
    <attribute name="type"><value-of select="@class" />
    <attribute name="name"><value-of select="@name" />
    // dependency links between actors
    // layout (e.g. xpos, ypos, with, height, etc.)
  </element>
</template>
```

✓ *XSL – Extensible Stylesheet Language*

✓ *Java-based XSL-Engine*

... iSTARML in ADOxx!

& UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

Cooperation Project

universität wien

Faculty of Computer Science

DKE

Thank you for your attention!

For any further information or logon details to the Open Models Initiative Platform, please contact

Margit.Schwab@dke.univie.ac.at