

ONTOLOGIES BASED COMMUNICATIONS
through MODEL DRIVEN TOOLS :
FEASIBILITY of the MDA
APPROACH in UCE projects

A.F. Cutting-Decelle

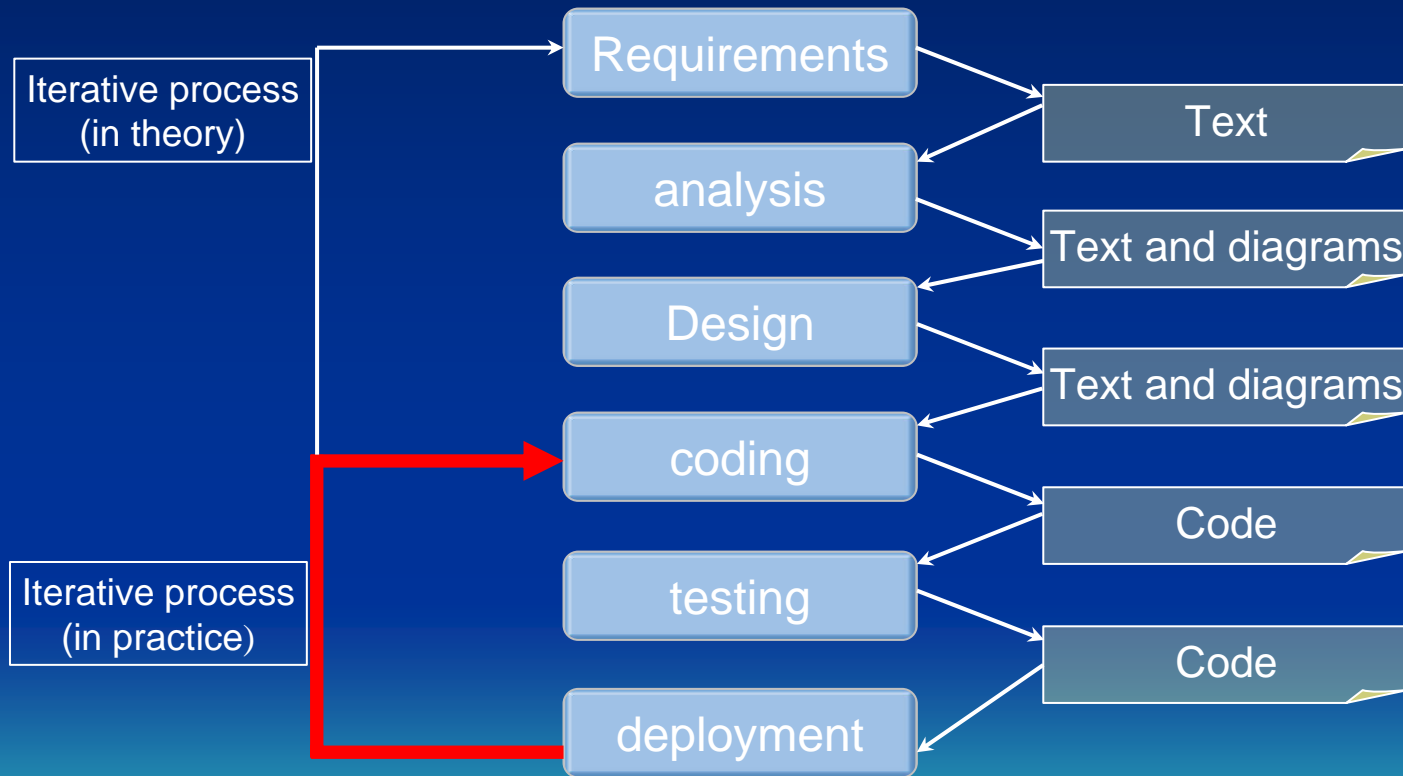
*Industrial Engineering Research Lab, Ecole
Centrale Paris, Chatenay Malabry, F*

- Introduction
- The MDA approach
- The MDD interoperability framework
- Application of MDA techniques to the management of ontologies in UCE projects
- Issues - perspectives

Introduction

- ❑ The productivity problem
- ❑ The portability problem
- ❑ The interoperability problem
- ❑ The documentation problem

The Productivity Problem :



The Portability Problem :

- ❑ Increase of new technologies
- ❑ Need to follow these new technologies
 - Technology required by customers
 - Can solve real problems (XML)
 - Only new technologies supported by tool vendors



Software tools need to interoperate

The Interoperability Problem :

- ❑ Existing software need to communicate with new ones
- ❑ Increase of modular tools
- ❑ Modules built according to specific technologies



Increase of need for interoperability

The Documentation Problem :

□ Documentation :

- often considered as a by-product
- time consuming
- slows down the development process

□ Result :

- need to generate documentation directly from the source code



The MDA approach

Written by :

**REYES GRANGEL (Universitat Jaume I),
JEAN-PIERRE BOUREY and
RAMZI BEN SALEM (Ecole Centrale Lille)**

- ❑ The MDA framework
- ❑ The MDA development life cycle
- ❑ The MDA benefits
- ❑ Inside the MDA Framework
- ❑ Model transformation

Object Management Group



❑ Is a consortium

- Created in April 1989
- By 11 companies (3Com, American Airline, Canon, HP, Philips, Sun, Unisys, ...)

❑ Over than 850 members

- Contributors (Boeing, Borland Software Corporation, Sun Microsystems, W3 Consortium...)
- Platforms (BEA Systems , Fujitsu, Hewlett-Packard , Nokia , SAP AG)
- Universities (INRIA , EURESCOM ...)

Object Management Group



□ Main goals

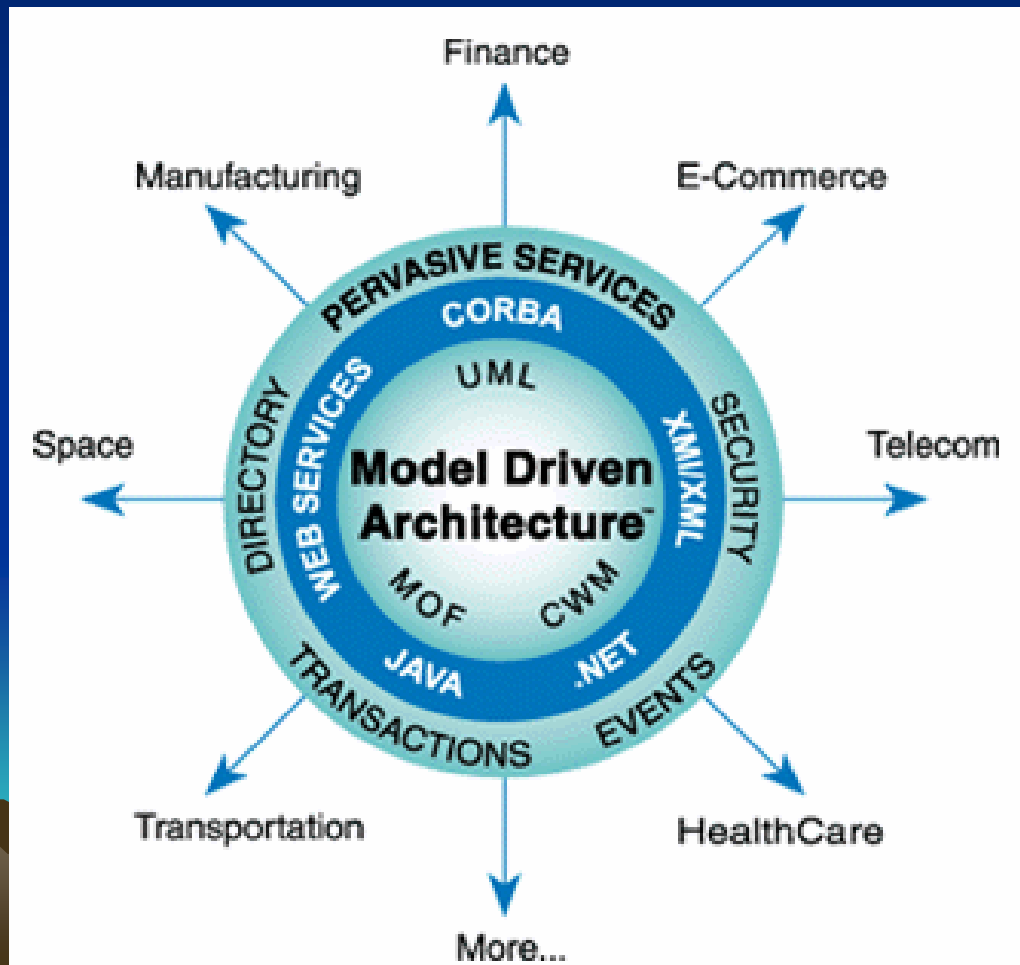
- Setting standards in the area of distributed object computing

□ OMG's specifications

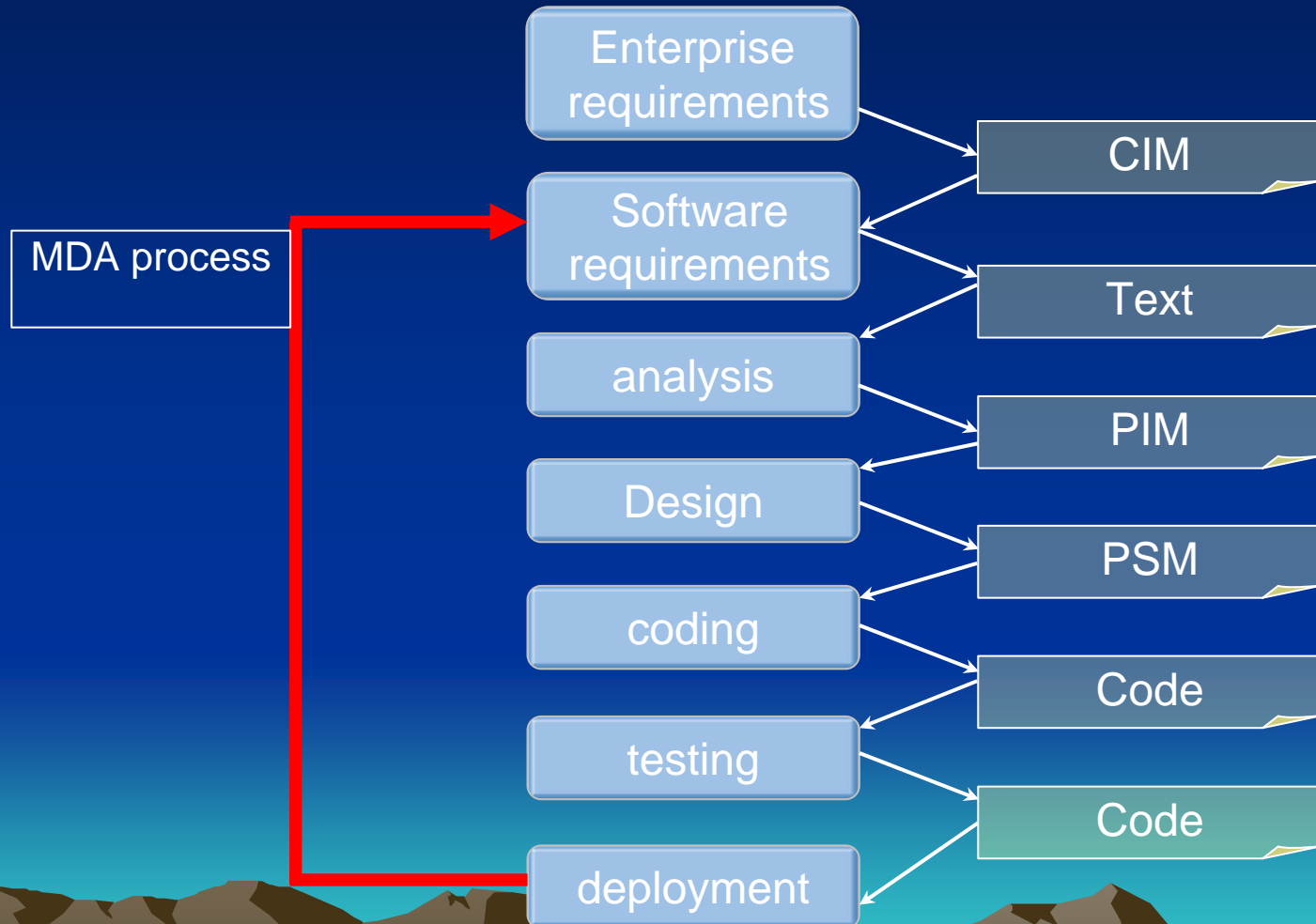
- UML
- CORBA/IIOP
- MDA
- CWM,...

MDA (Model Driven Architecture)

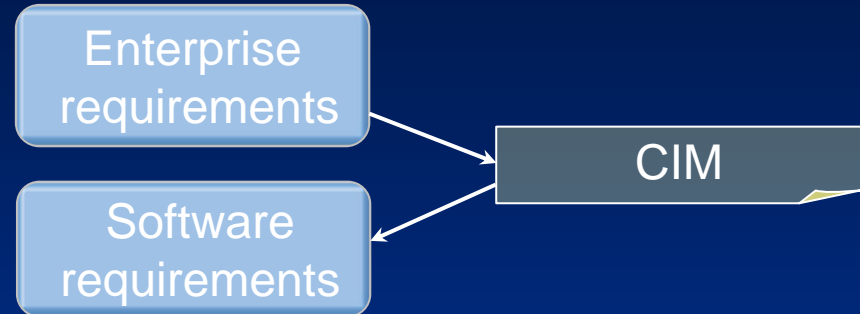
Attempt to solve the previous problems



The MDA development life cycle

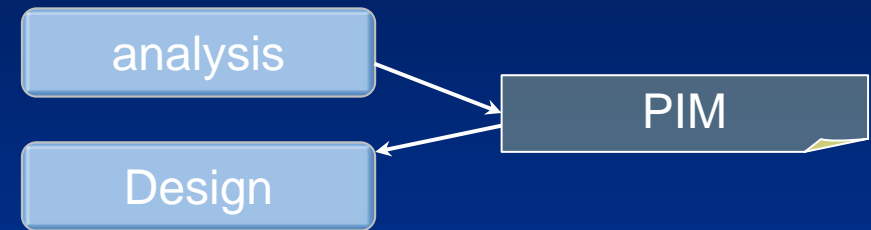


Computation Independent Model (CIM)



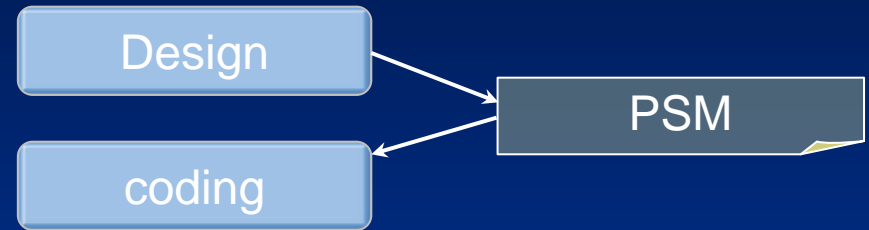
- ❑ Specifies business processes, stakeholders, departments, dependencies between processes, ...
- ❑ Represents the system requirements
- ❑ Does not show details of the structure
- ❑ Made of two subdivisions:
 - Business Model
 - Business Requirements for Systems.

Platform Independent Model (PIM)



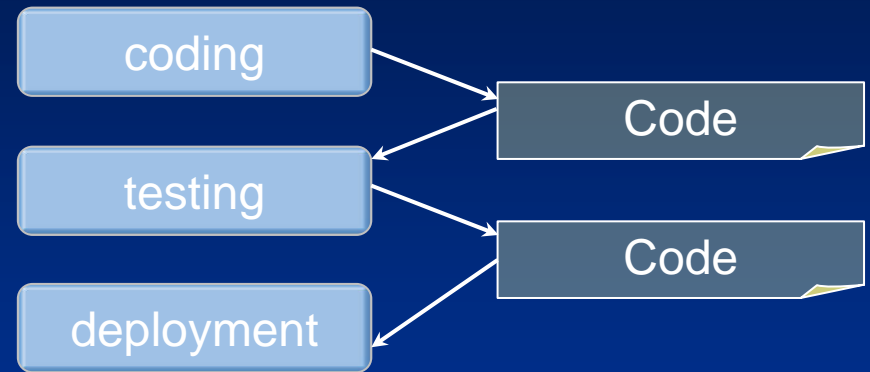
- ❑ Shows a high level of abstraction
- ❑ Independent of any implementation technology
- ❑ Describes the software system

Platform Specific Model (PSM)



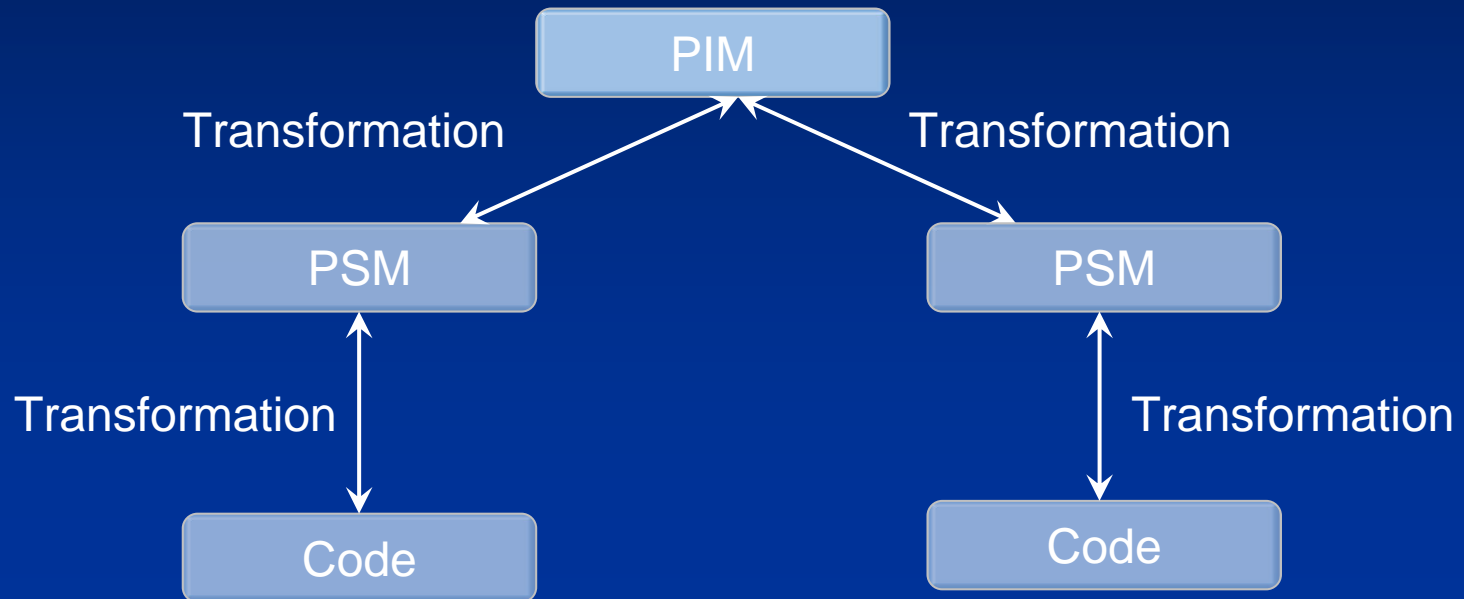
- ❑ Describes the system in one specific implementation technology
- ❑ A PIM is transformed into one or more PSMs
- ❑ For each specific technology platform a separate PSM is generated
- ❑ Example:
 - Relational Database PSM for (column, foreign key...)
 - EJB PSM (entity bean, home interface, session bean...)

Code



- ❑ Represents the final step of the development process
- ❑ Automatically generated

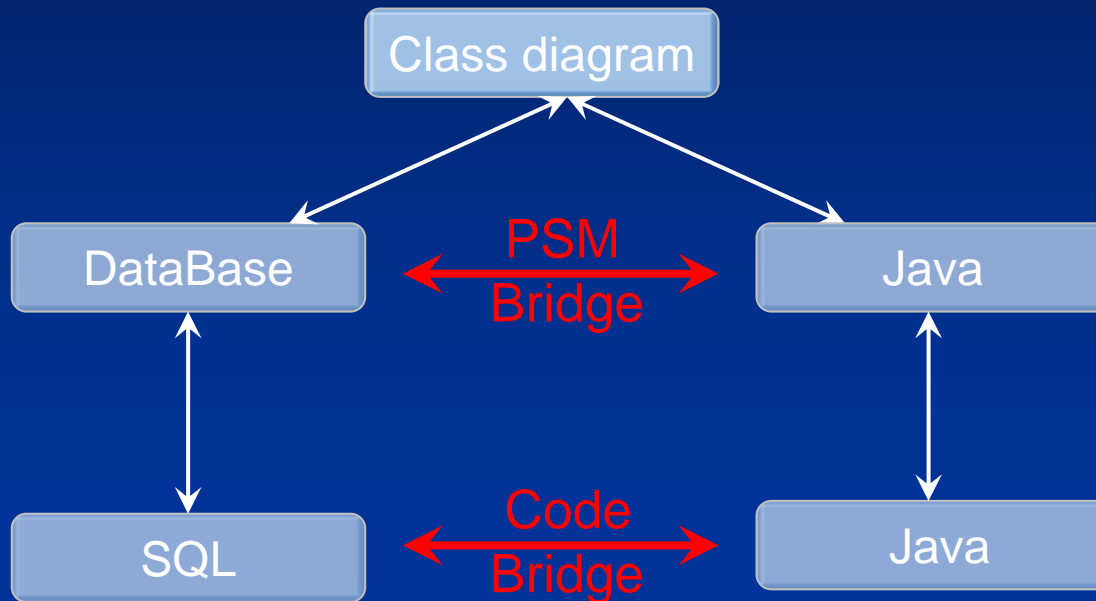
The MDA transformation steps



MDA benefits

- ❑ Interoperability
- ❑ Portability
- ❑ Productivity
- ❑ Documentation

The interoperability benefit



Productivity gains

- ❑ Minimize the development time
- ❑ Focused on PIM development
- ❑ Improved functionalities
- ❑ Reduced development time

Portability benefits

- ❑ A PIM can be transformed into multiple PSMs
- ❑ A PIM describes a portable specification

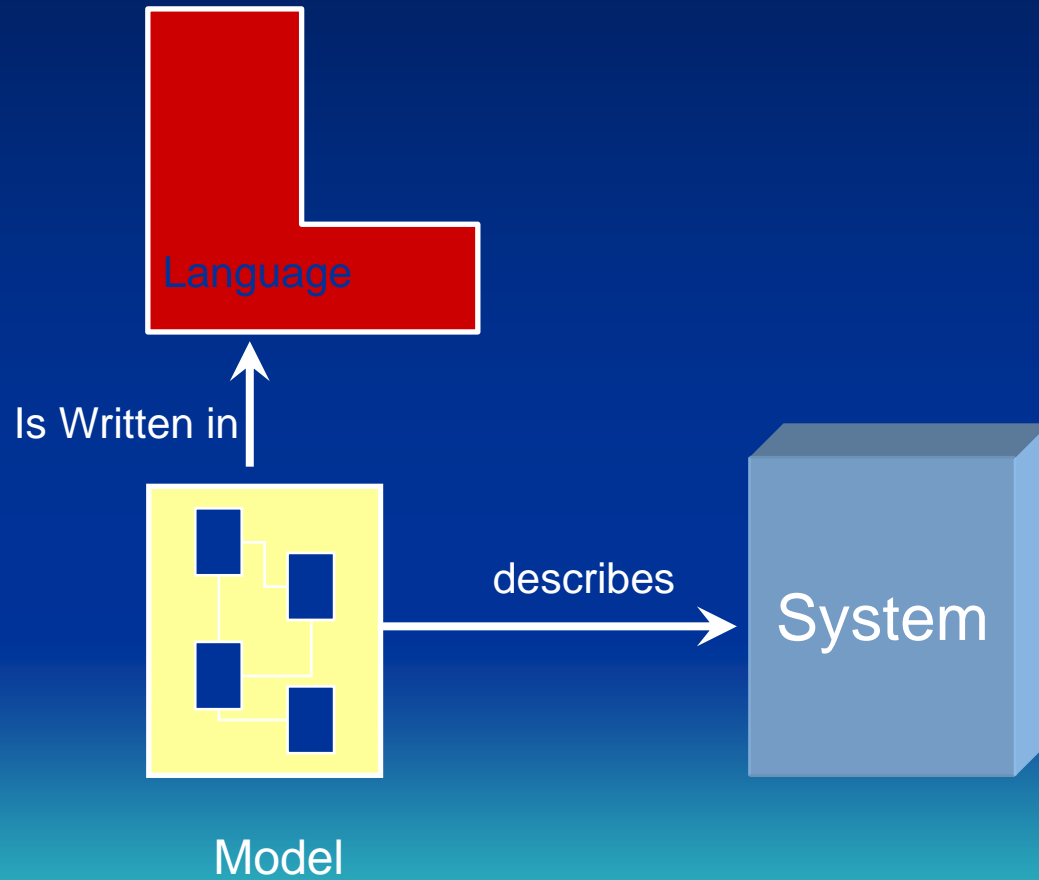
Maintenance and documentation benefits

- ❑ Documentation always up-to-date
- ❑ The PIM provides the high level documentation
- ❑ Consistency checking between a high level documentation and the code

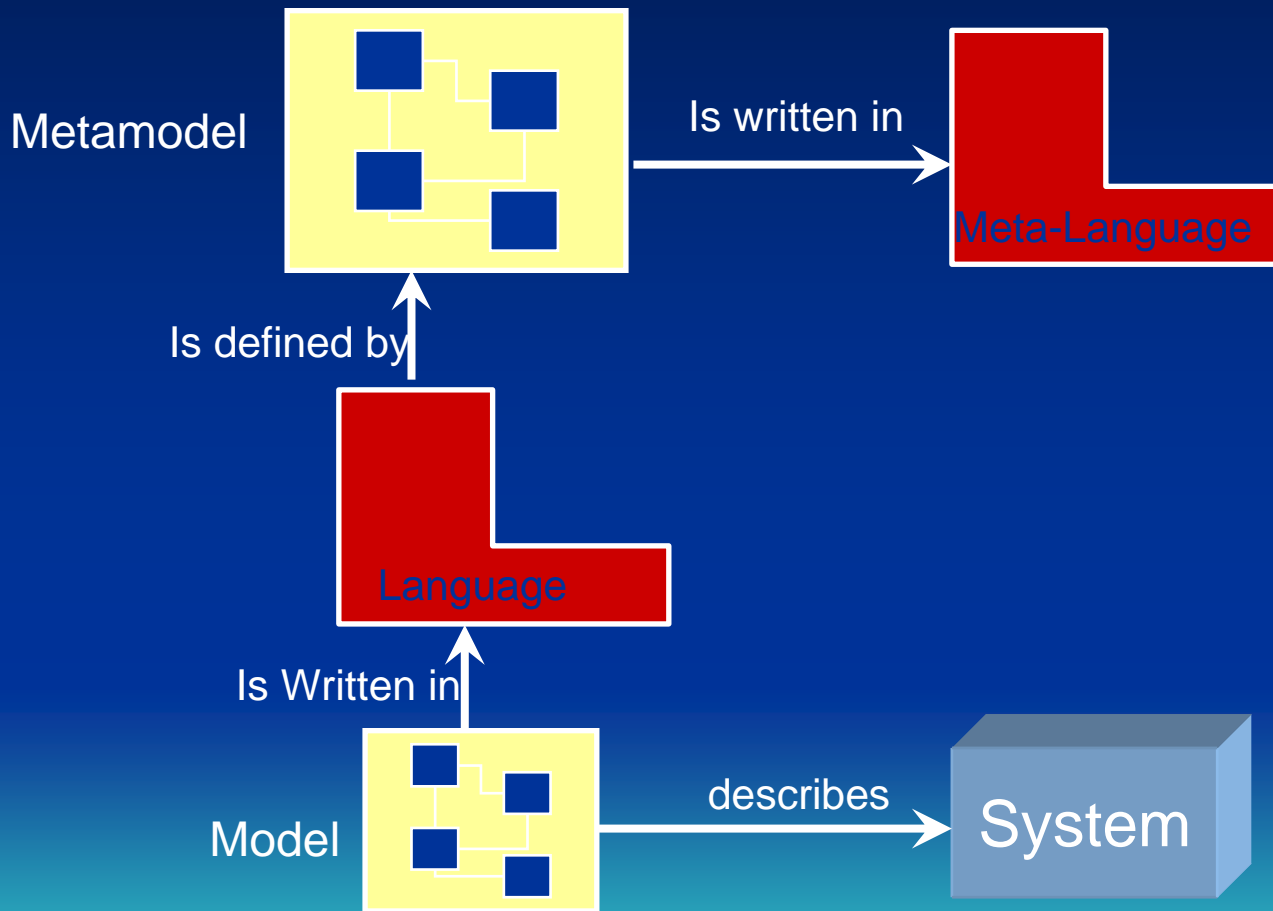
Inside the MDA Framework :

Modelling and metamodelling

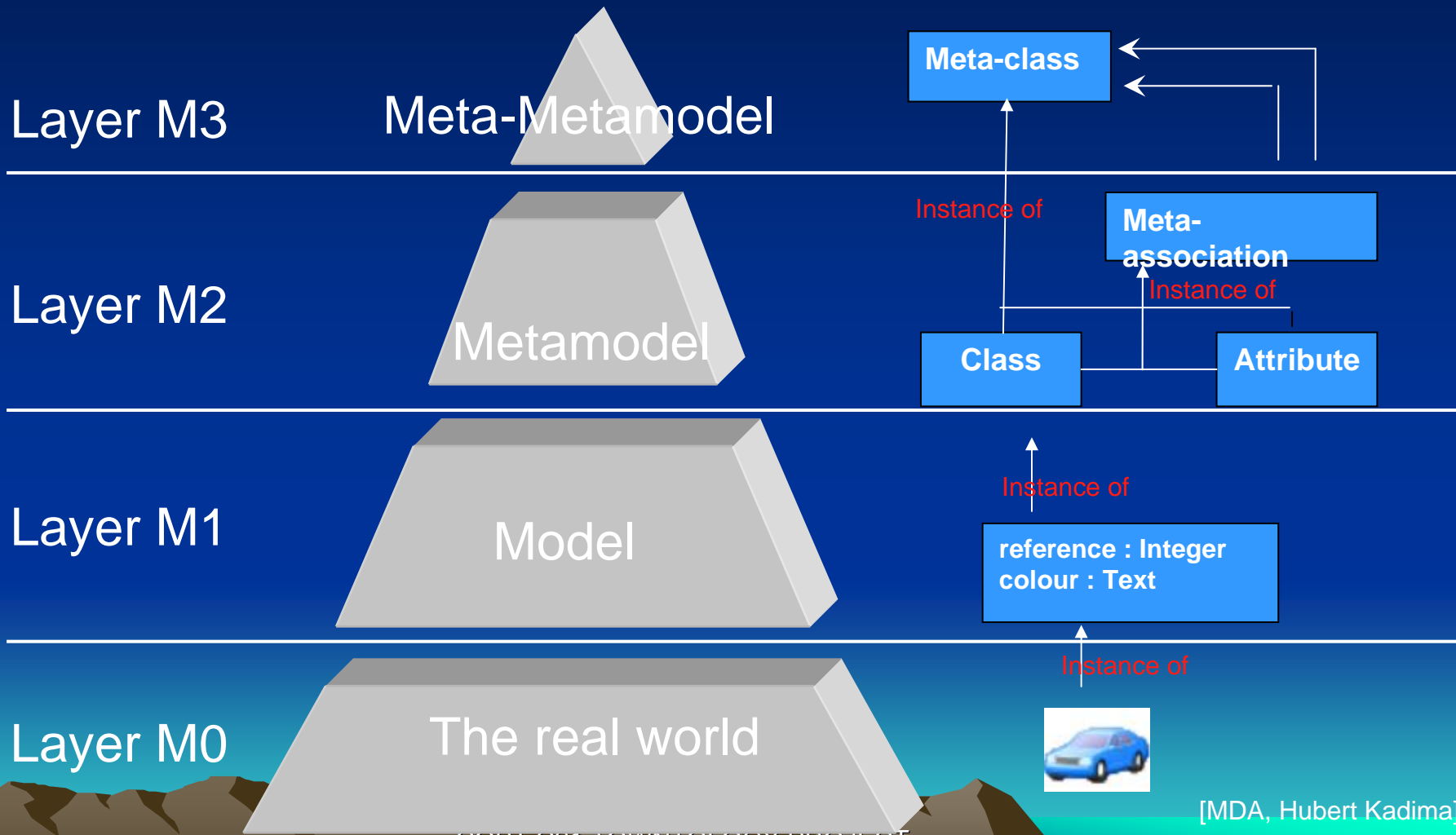
Model



Metamodel



The OMG's metamodeling architecture



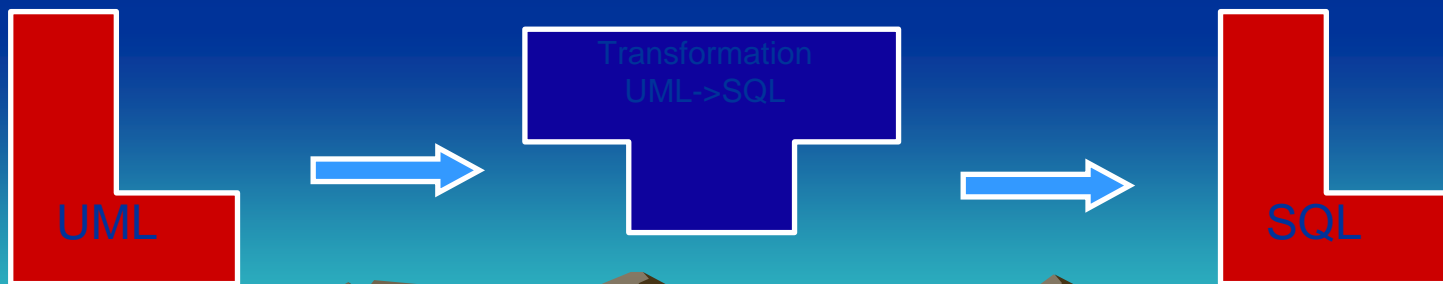
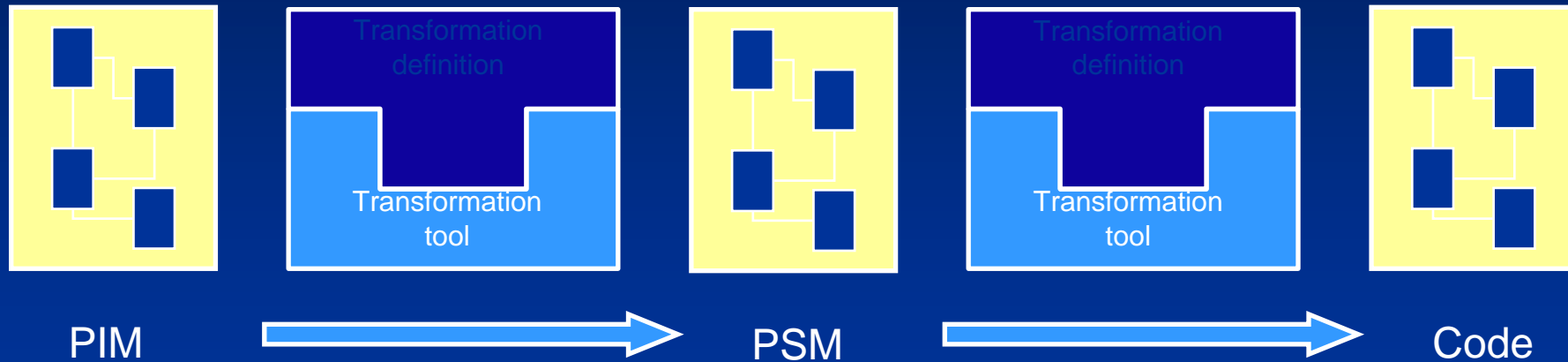
[MDA, Hubert Kadima]

Inside the MDA Framework: Transformations

Some definitions

- A **transformation** is the automatic generation of the target model from a source model, according to a transformation definition
- A **transformation definition** is a set of transformation rules that together describe how a model in the source language can be transformed into a model in the target language
- A **transformation rule** is a description of how one or more constructs in the source language can be transformed into one or more constructs in the target model

Role of the transformation tool



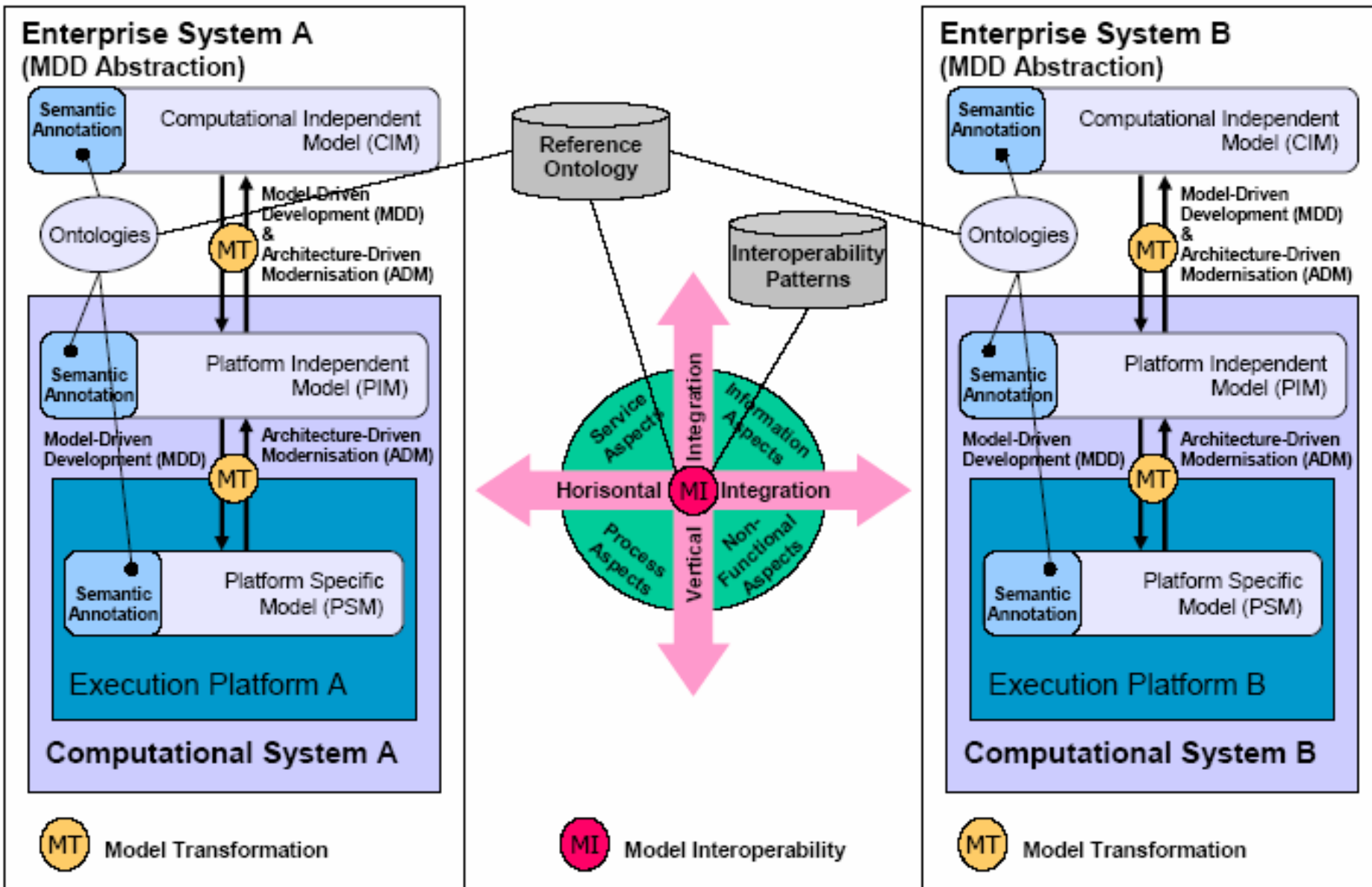
Requirements for a transformation rule :

- ❑ Knowledge of the source language reference
- ❑ Knowledge of the target language reference
- ❑ A set of source language model elements
- ❑ A set of target language model elements
- ❑ The source language condition
- ❑ The target language condition
- ❑ A set of mapping rules



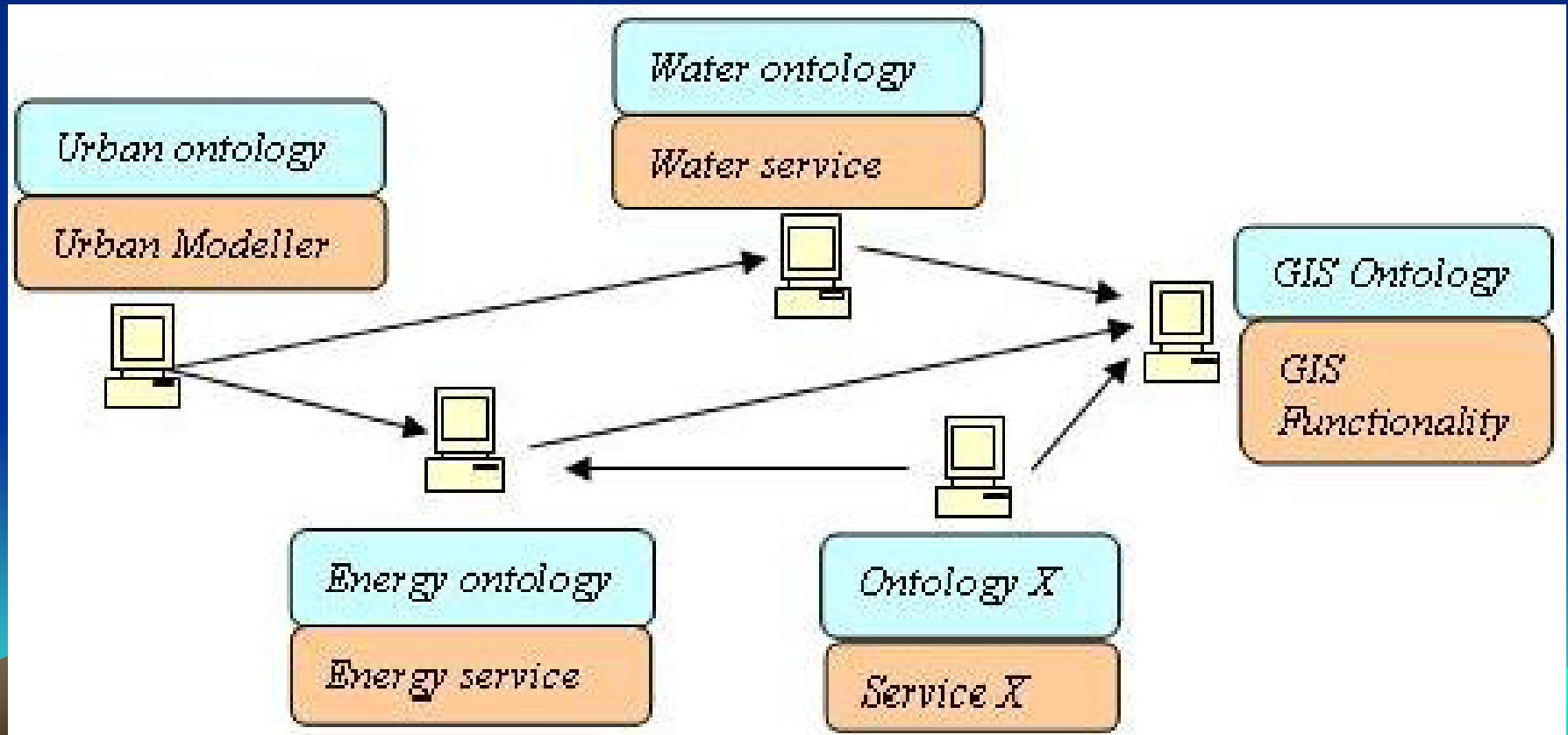
NEED OF ONTOLOGIES

MDD Interoperability framework

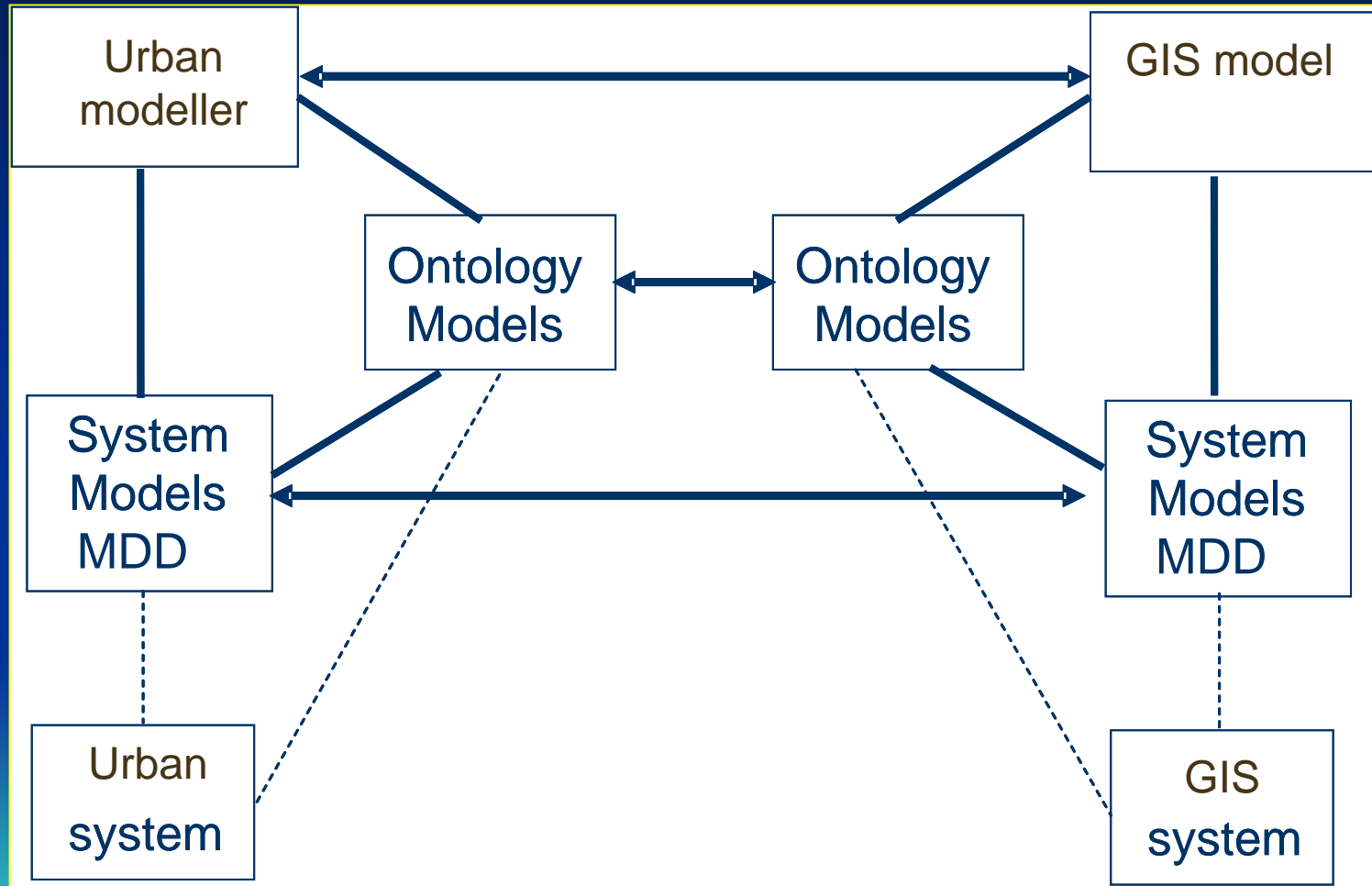


Application of MDA Techniques : Management of ontologies in UE projects

Example of a generic urban development application using several ontology-based software tools :

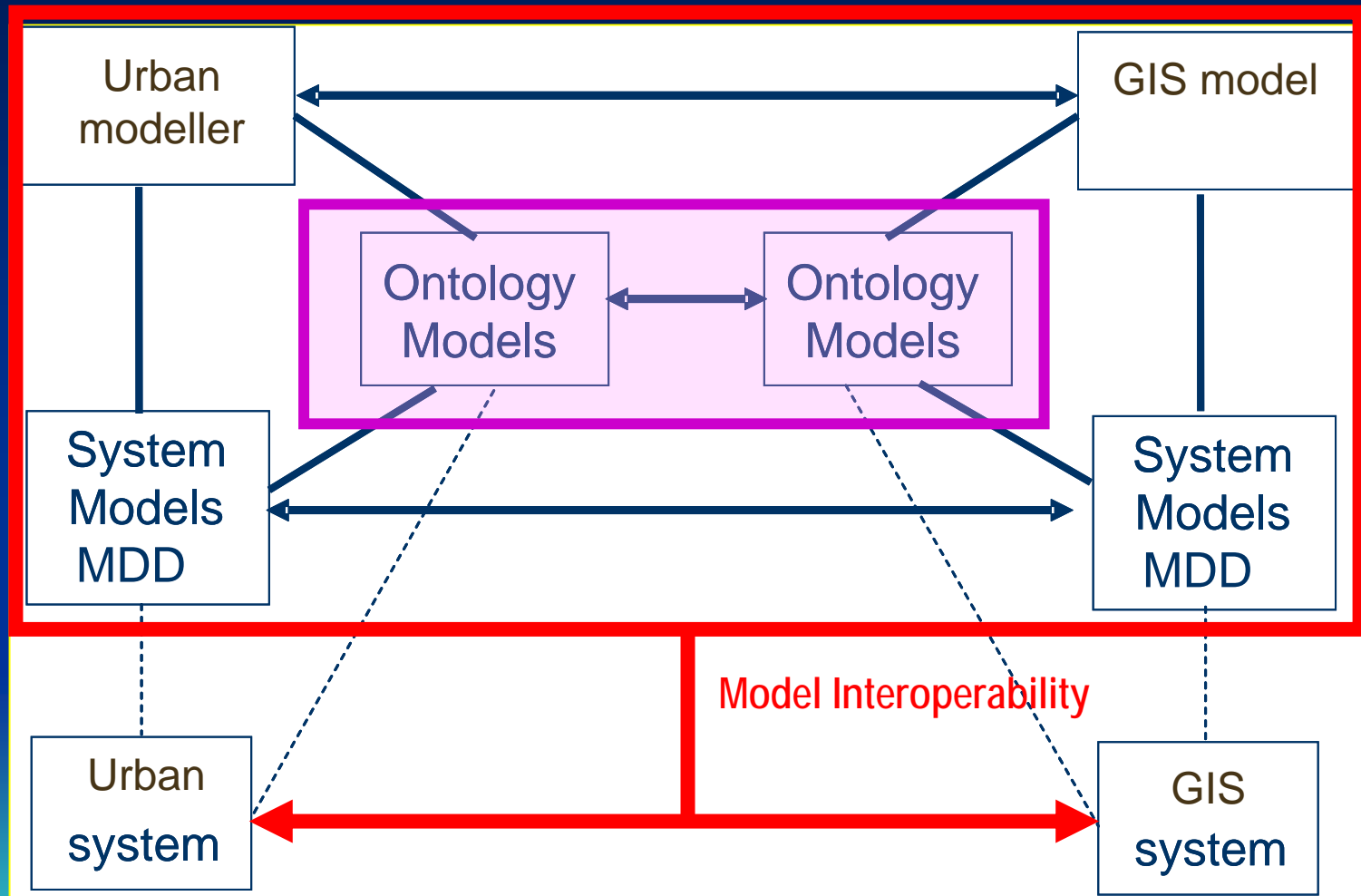


Model transformations based on ontologies :



Model transformations based on ontologies :

Model interoperability



Issues- Perspectives

Specificity of the sector :

- ❑ Important coordination activities in UCE projects among numerous software tools : need of sharing a common knowledge about the project
- ❑ Not all the activities are computerised : need of a common knowledge understandable by humans
- ❑ Numerous informal relations among the partners : an important part of the knowledge is not formalised : important implicit knowledge
- ❑ Important variations between countries – or else between regions



Importance of model driven approaches
Role of ontologies for model transformations

Model Driven Architecture

Developed by **PARTNER**
for **INTEROP Project**
www.interop-noe.org

Authors:

Reyes Grangel grangel@lsi.uji.es
Institution: Universitat Jaume I
Country: Spain

Jean-Pierre Bourey Jean-Pierre.Bourey@ec-lille.fr
Ramzi Ben Salem bensalemramzi@yahoo.fr
Institution: Ecole Centrale Lille
Country: France

History:

Describe here the tutorial history (past projects, etc.)

Published:

Disclaimer and Copyright Notice: Permission is granted without fee for personal or educational (non-profit) use. In other cases contact with INTEROP Project Coordinator interop_coordinator@listes.u-bordeaux1.fr for use conditions. Some of the figures presented in this tutorial are freely inspired by others reported in referenced works/sources. For such figures copyright and all rights therein are maintained by the original authors or by other copyright holders. It is understood that all persons copying these figures will adhere to the terms and constraints invoked by each copyright holder.

QUESTIONS ?