



Praxeme, the sense of action
An initiative for a public method

Semantic Modeling & Data Management

MDM Project Outcome
5/04/05

✉ info@praxeme.org
☎ +33 (0) 6 77 62 31 75

<http://www.praxeme.org>

Référence :
MDM/SI B-15

Version : 5/04/05



Objectives

- **Explain MDM Project approach**
 - Why build a Semantic Model before Logical Model
 - How to exploit the Semantic Model
- **Learn from the experience**
 - Difficulties encountered
 - Best practices
- **Draw up the project outcome**
 - Benefits of Semantic Modelling
 - Benefits for the IS



- **The Approach**
 - The theoretical basis
 - Process
- **Semantic Modelling**
 - Principles & objectives
 - Illustration
 - Benefits
- **Semantic Model Transformation**
 - Logical Data Model
 - Service Oriented Architecture
- **The Plan**
 - The story of the Project
 - Continuing on post-MDM project



Agenda

| Topic | Time |
|--------------------------------------|----------------------|
| The Approach | 17pm – 17:15 |
| Semantic Modelling | 17:15 – 17:30 |
| Semantic Model Transformation | 17:30 – 17:45 |
| The Plan | 17:45 – 18pm |



1

Contents of Part 1

- **"Aspects" of the Enterprise System**
 - Theoretical basis to the approach
- **Principles & objectives of Semantic Modelling**



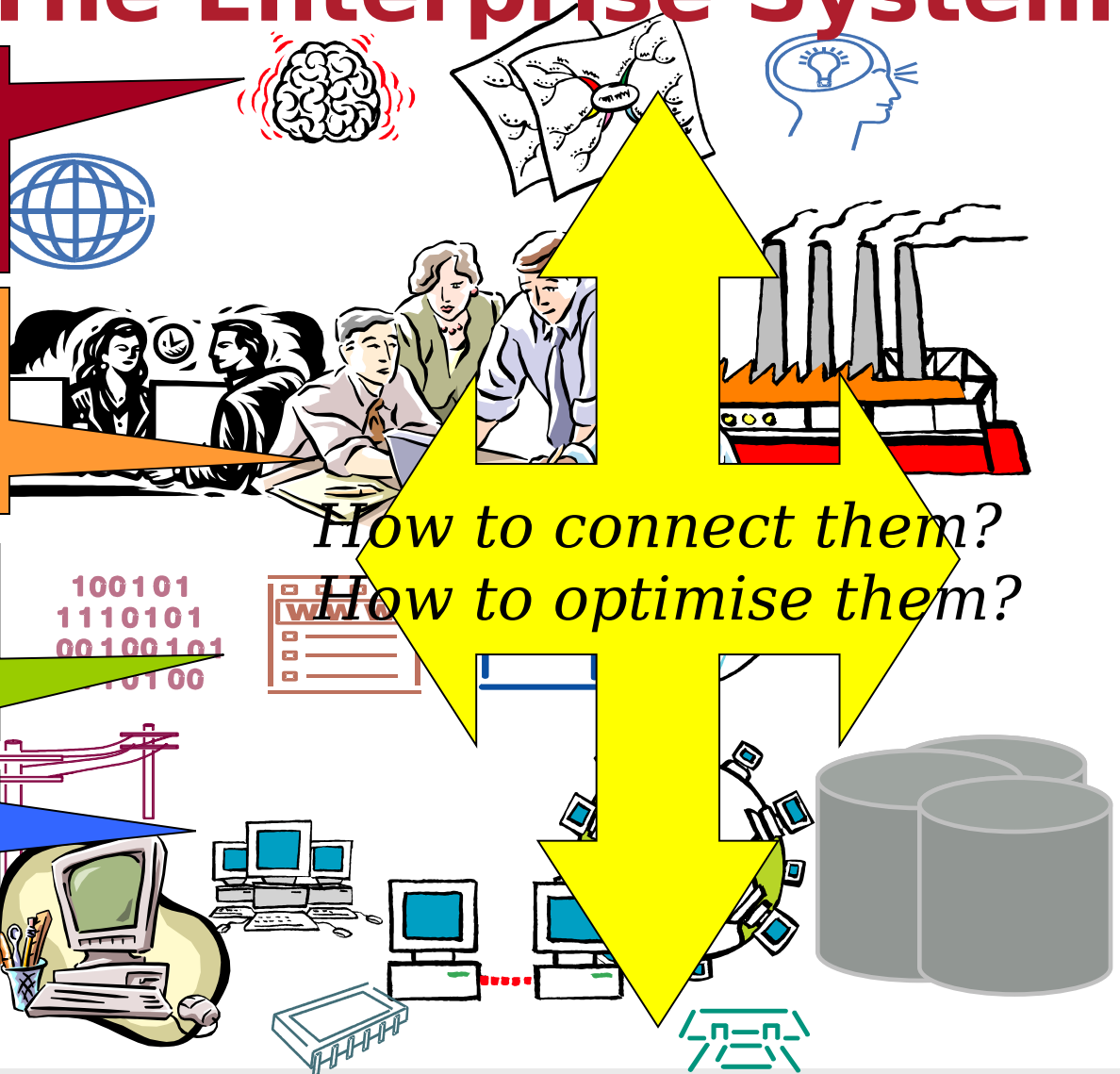
The Enterprise System

Knowledge: the core business, notions...

Activity: business processes...

Software Tools

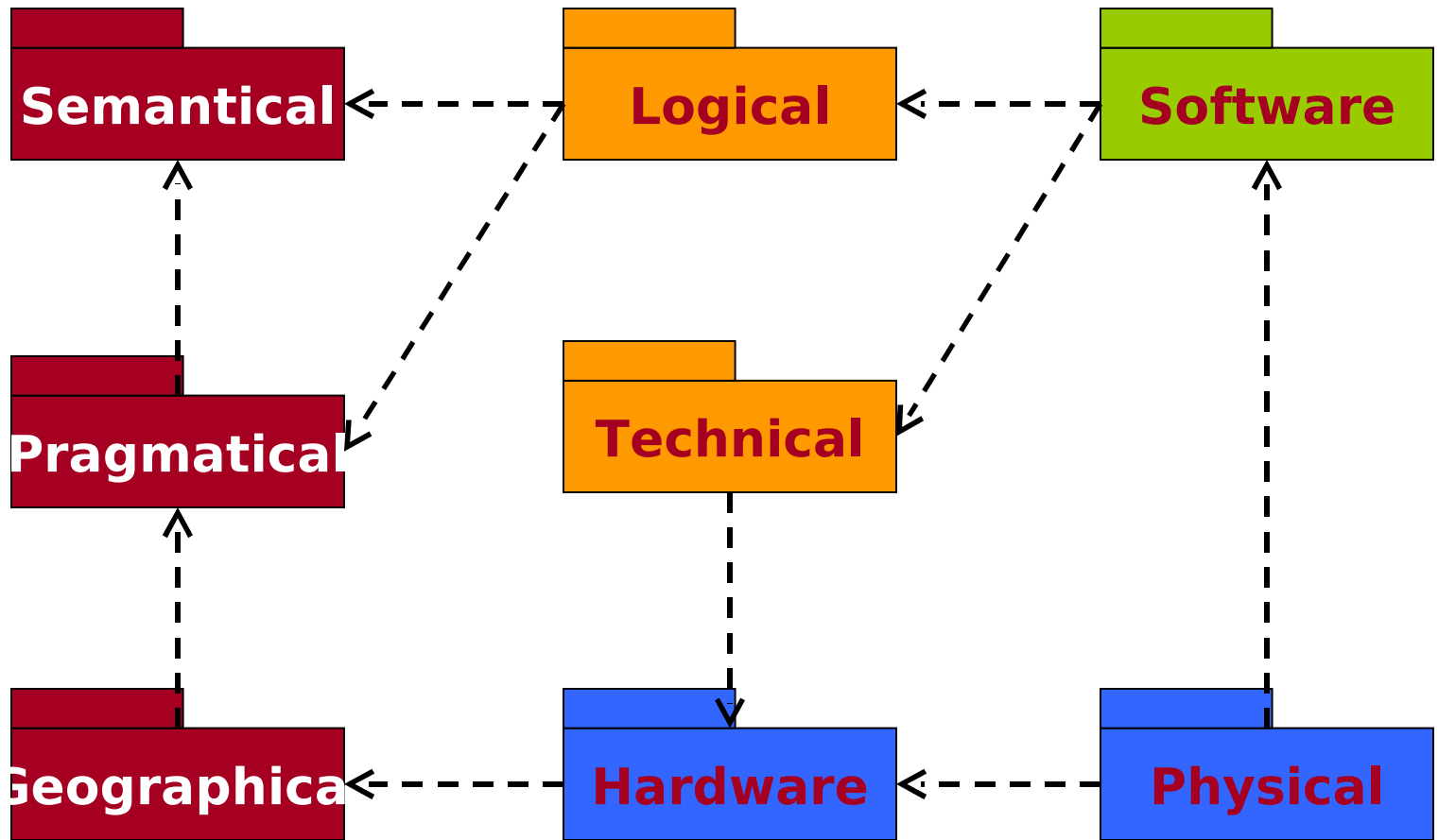
Equipment





"Aspects" of the Enterprise System

The Approach

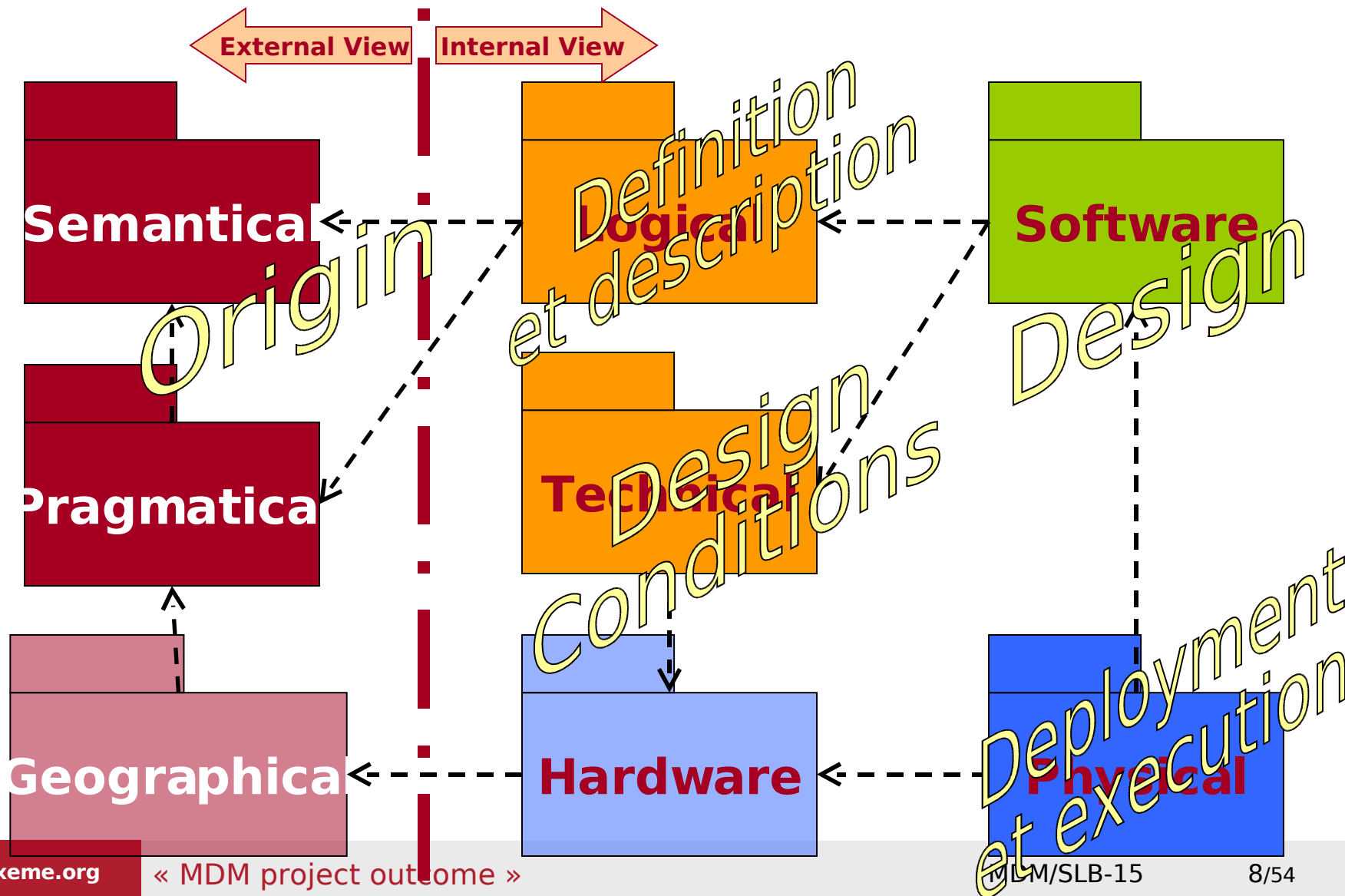


The Enterprise System Topology



The Service transformation

The Approach Chain





- **A multiple aspect approach**
- **The MDM project respects the requirements:**
 - Provide signification to the data
 - Convergence on a data perspective
- **To do this, the approach is based upon the following principles:**
 - Abstraction
 - Conceptual view
 - Signification
 - Define the modelling elements as rigourously as possible
 - Structure
 - Improve the quality of the system: encapsulation, limited coupling...
 - Prepare for a Service Oriented Approach



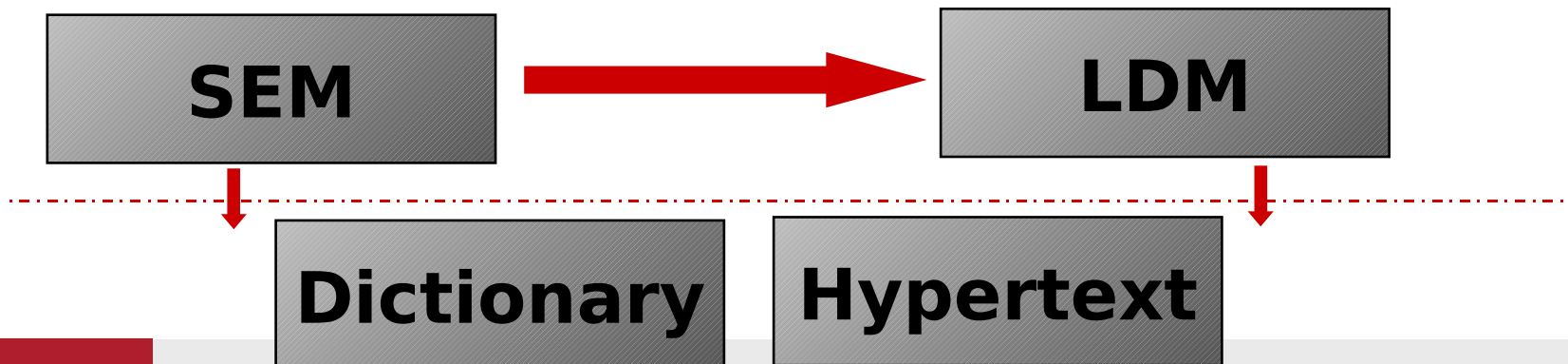
1. The final deliverable is the Logical Data Model

- Scope: needs of the CORSO project
- LDM: Relational Model

2. An intermediate deliverable: the Semantic Model

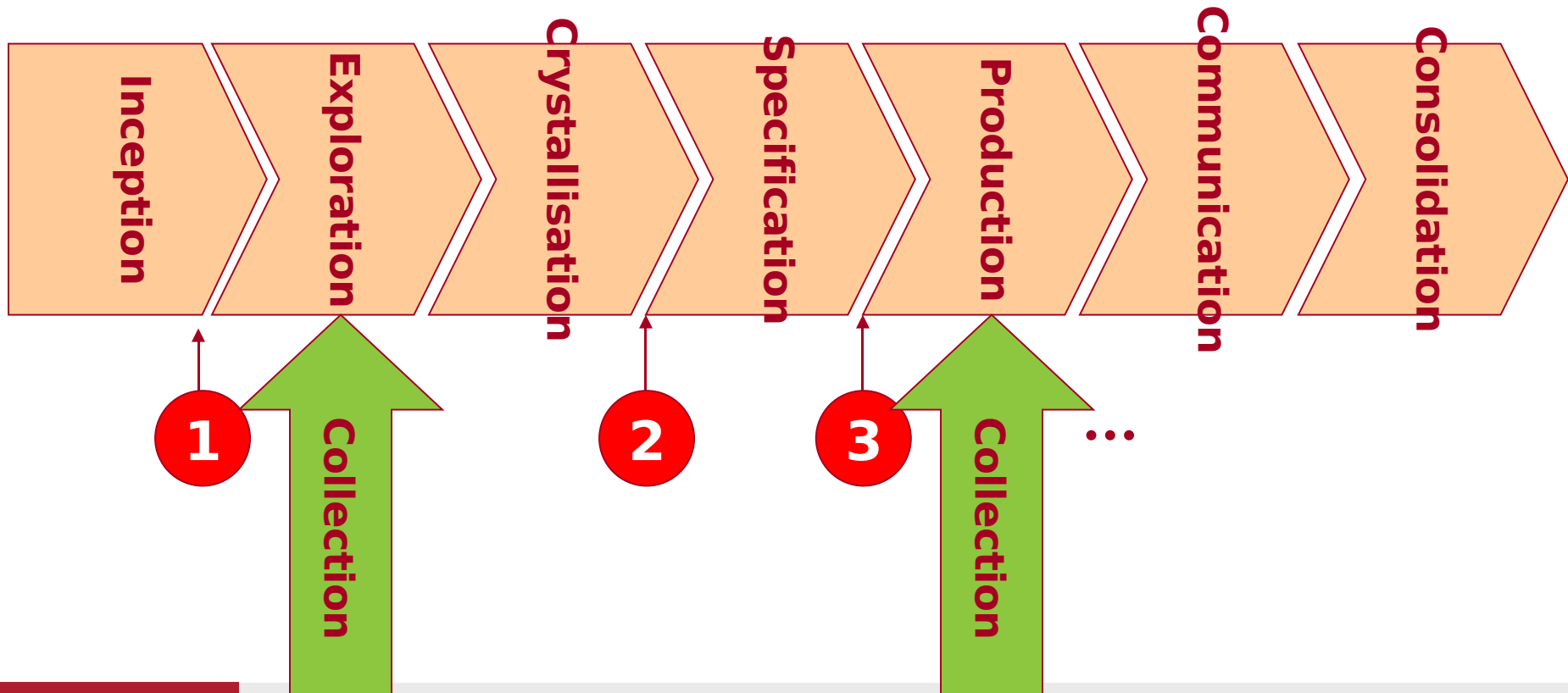
3. Other derived products

- To facilitate the use of the models



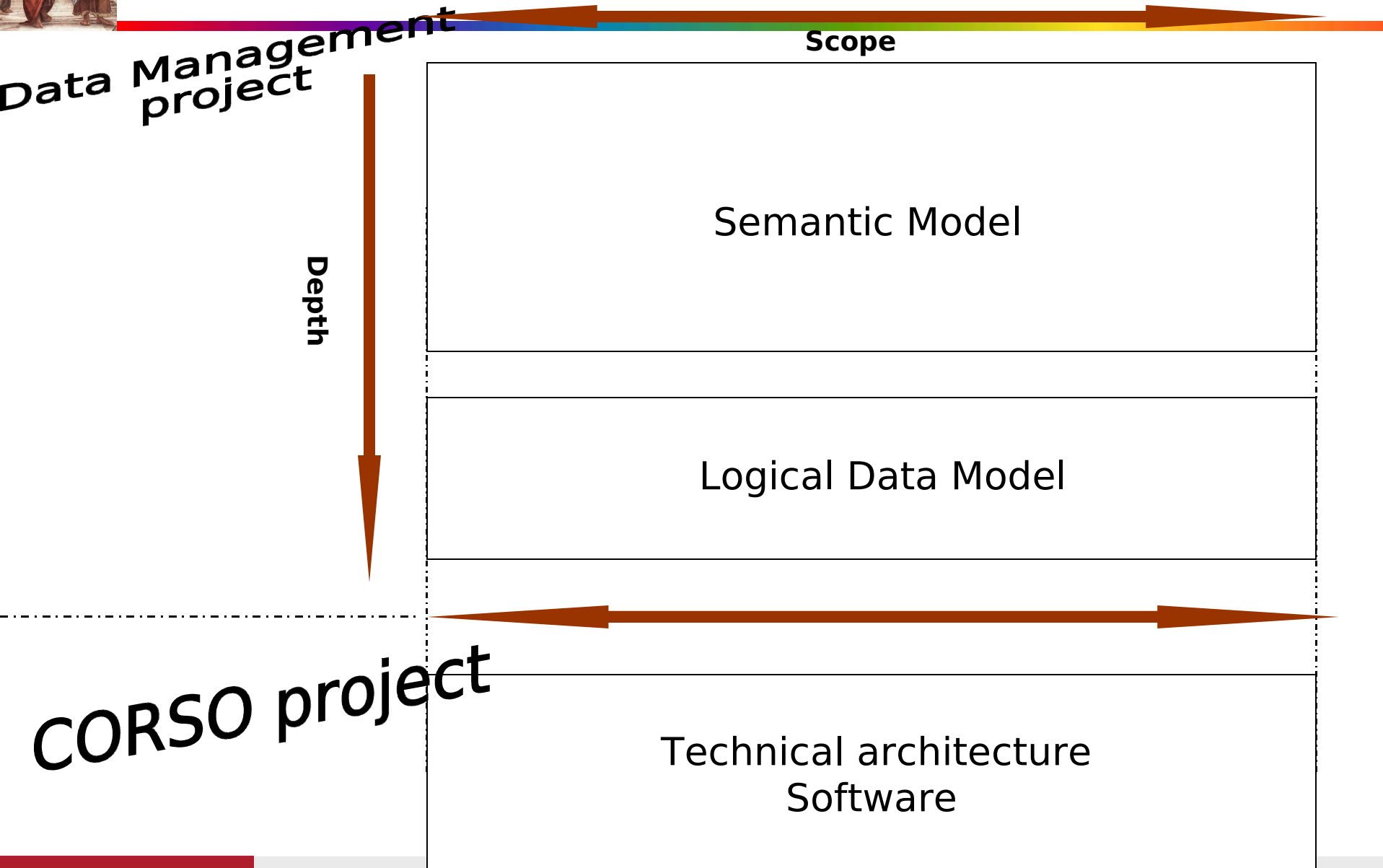


▪ "Innovation" Approach





The Approach





The Approach

Data Management project

Scope

Semantic Model

Depth

Exploration phase

Production phase

Crystallisation

Logical Data Model

Communication phase

CORSO project

Technical architecture
Software



The Approach

Exploration phase

Exploration Phase Unilog

Modelling the semantics
from use-cases
description

Draft Sem :Mdl

Exploration Phase Celesio

Selecting the relevant
information from the
legacy

Logical data
description

Production Phase

Matching the Sem
Mdl
and the existing data

Enhanced Sem :Mdl

Comm. Phase

Deriving the LD Mdl
from the Sem Mdl

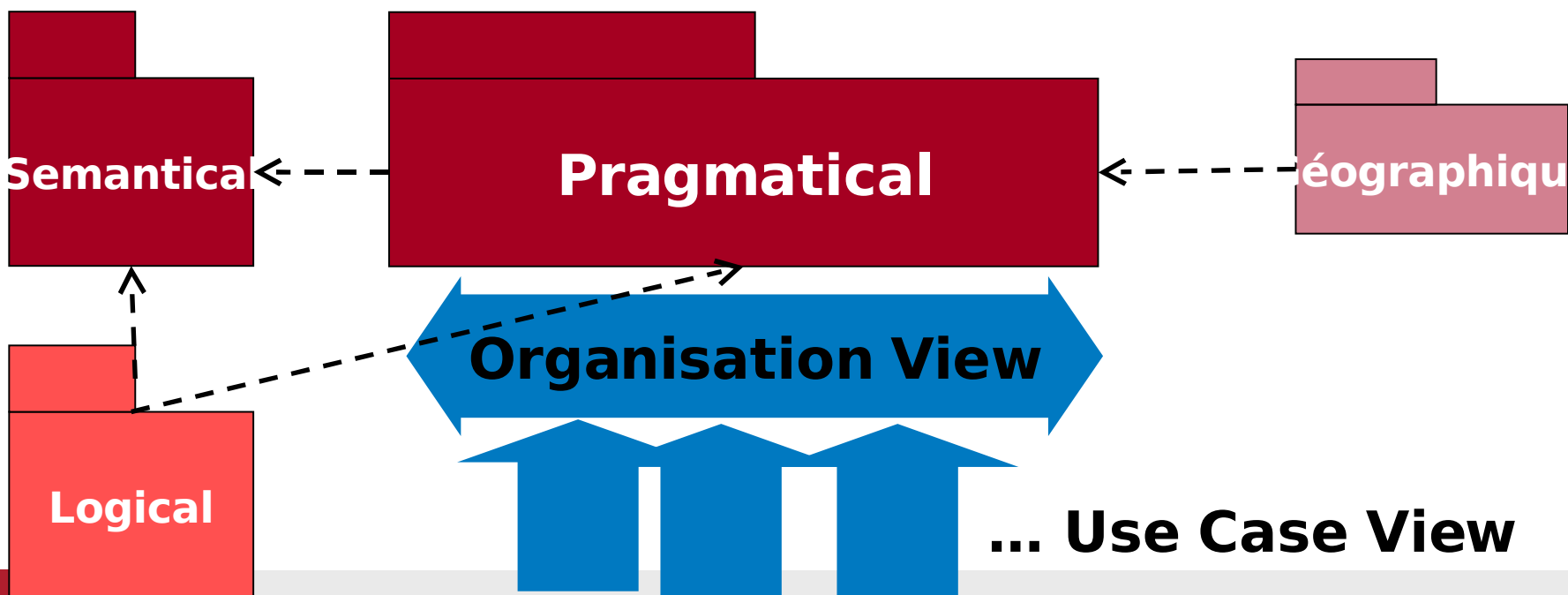
LD :Mdl



Semantic Model and Use Case

The Approach View

- One of the representation inside the « Pragmatical » aspect
 - Functional approach
 - Local point of view





Method deliverables

- **User's Guide to Semantic Model**
 - MDS03-UserGuide.doc
- **Templates for the collected documentation**



- **Principles and objectives of semantic modelling**
- **Difference between classical and semantic modelling**
- **Illustration**
 - Genericity
 - Transfer order



Principles & Objectives

Semantic Modelling

- **Format**

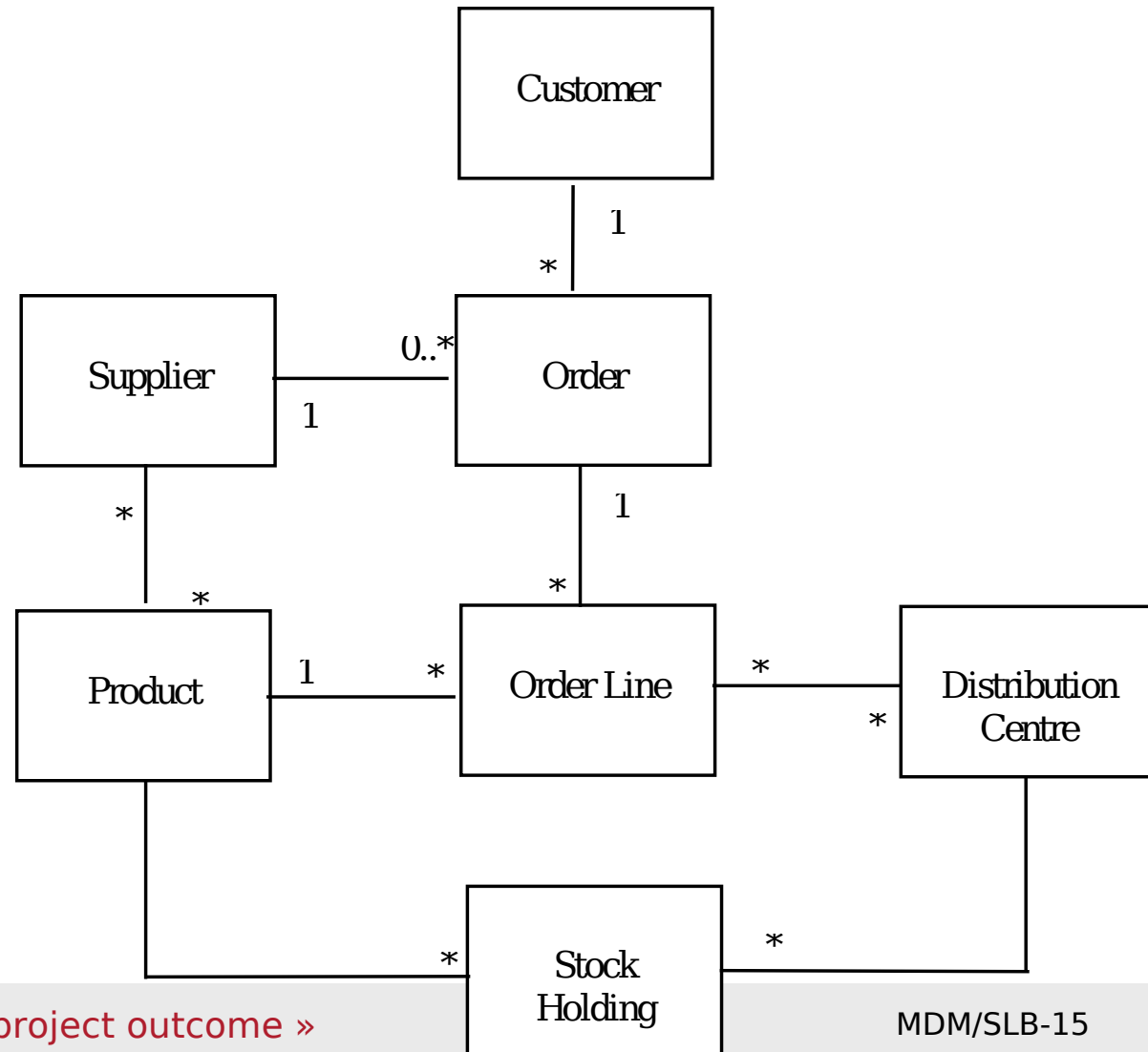
- UML language: class diagram, state diagram & collaboration diagram
- Data Dictionary
 - Synonyms, keywords, definitions and comments

- **High level of abstraction**

- Improves expressiveness
- Provides a basis for modelling for follow-on projects
- Data oriented (with an object-oriented approach)
 - Reminder: It is not be a “complete” model (few operations and constraints will be captured)

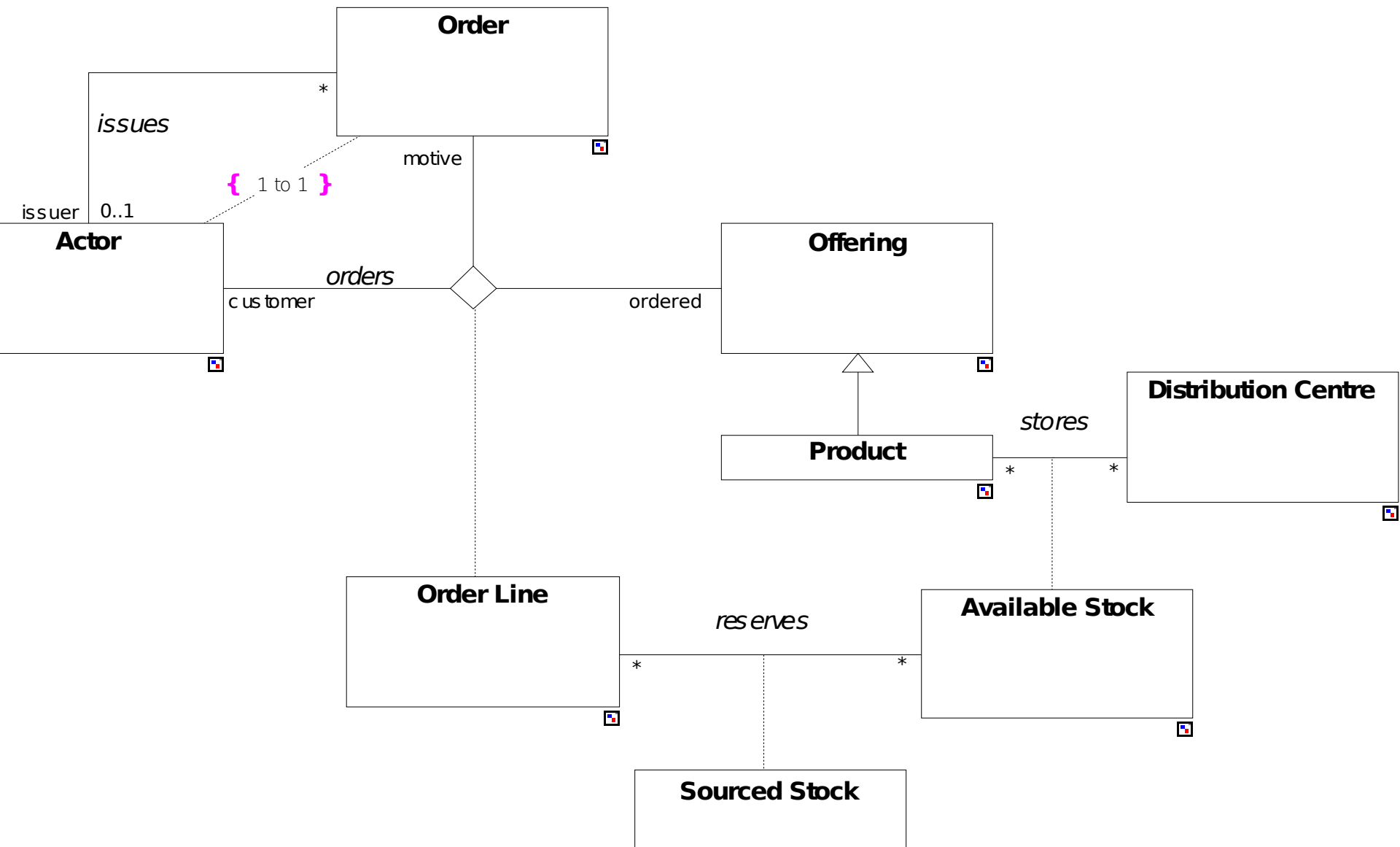


From classical data model... Semantic Modelling





...towards semantic modelling





- **The generic concept « Offering »**
 - Allow to order services as well as product by the same process
 - Avoid structural redundancy
 - Factorizing the common properties
- **The generic concept of actor**
 - Customer, Supplier, Entity, Person...
 - A given actor can act in many ways
 - Both as customer and supplier
 - As a worker...
 - A supplier or a DC can order products



Genericity (continued)

Transfer order

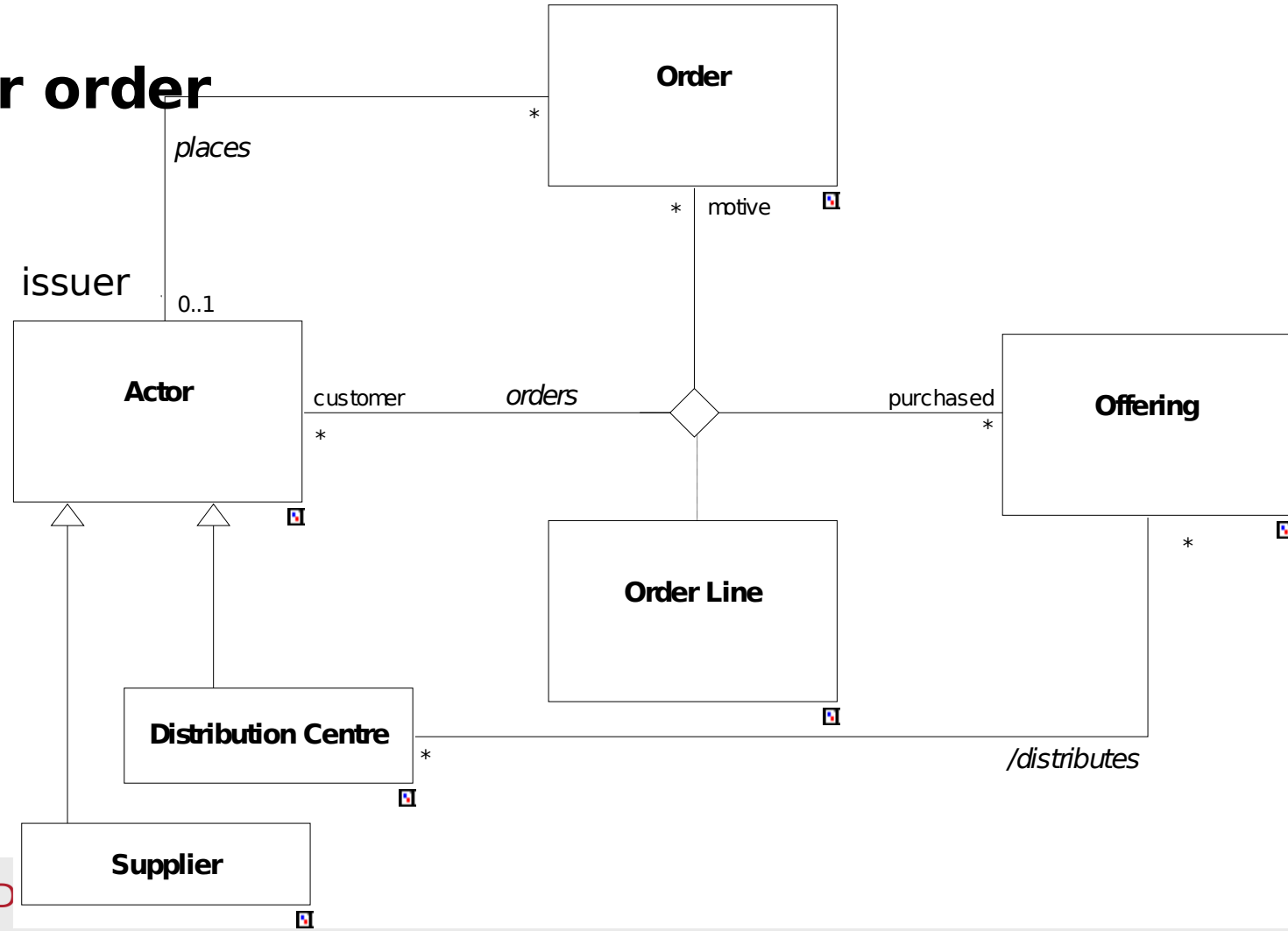
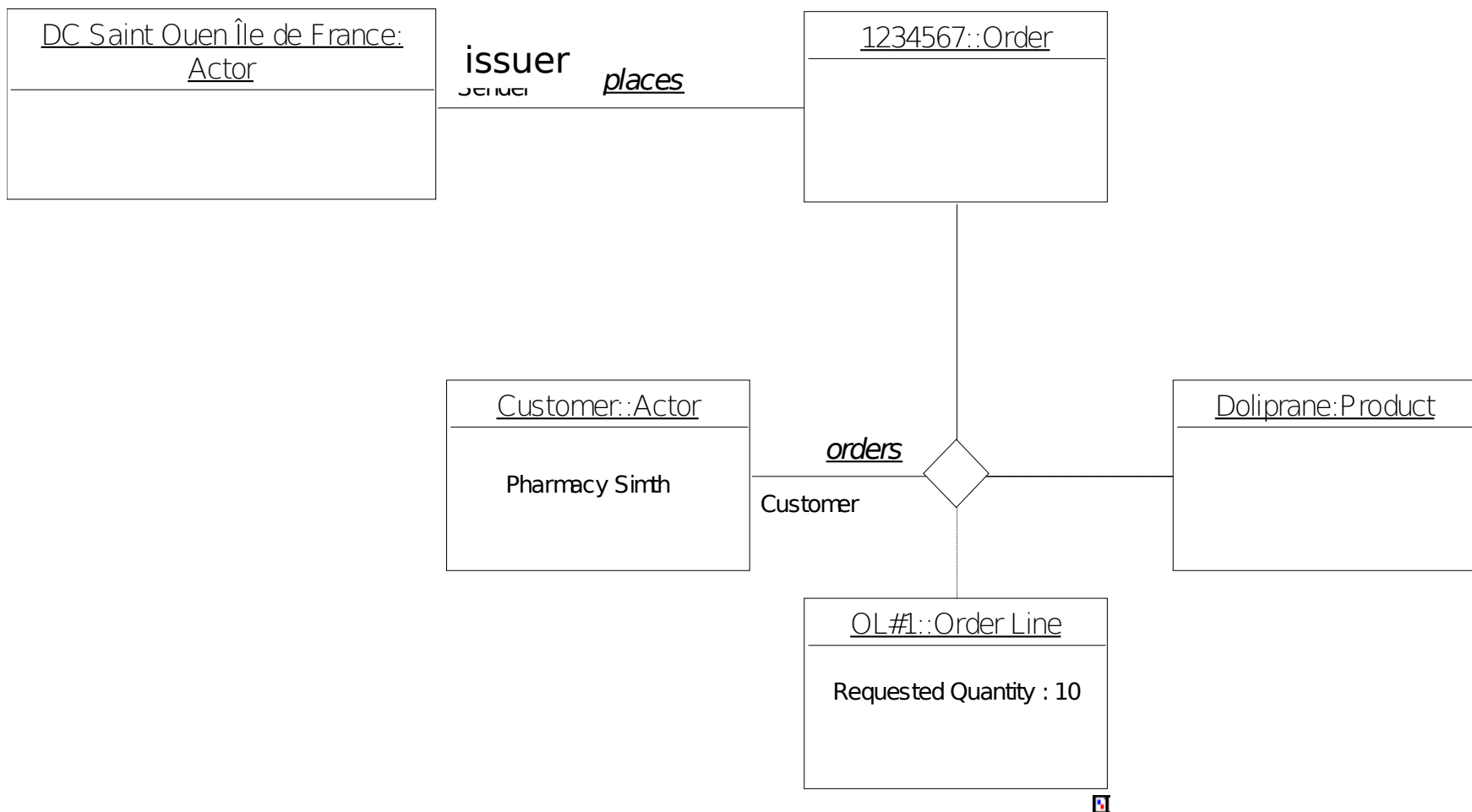




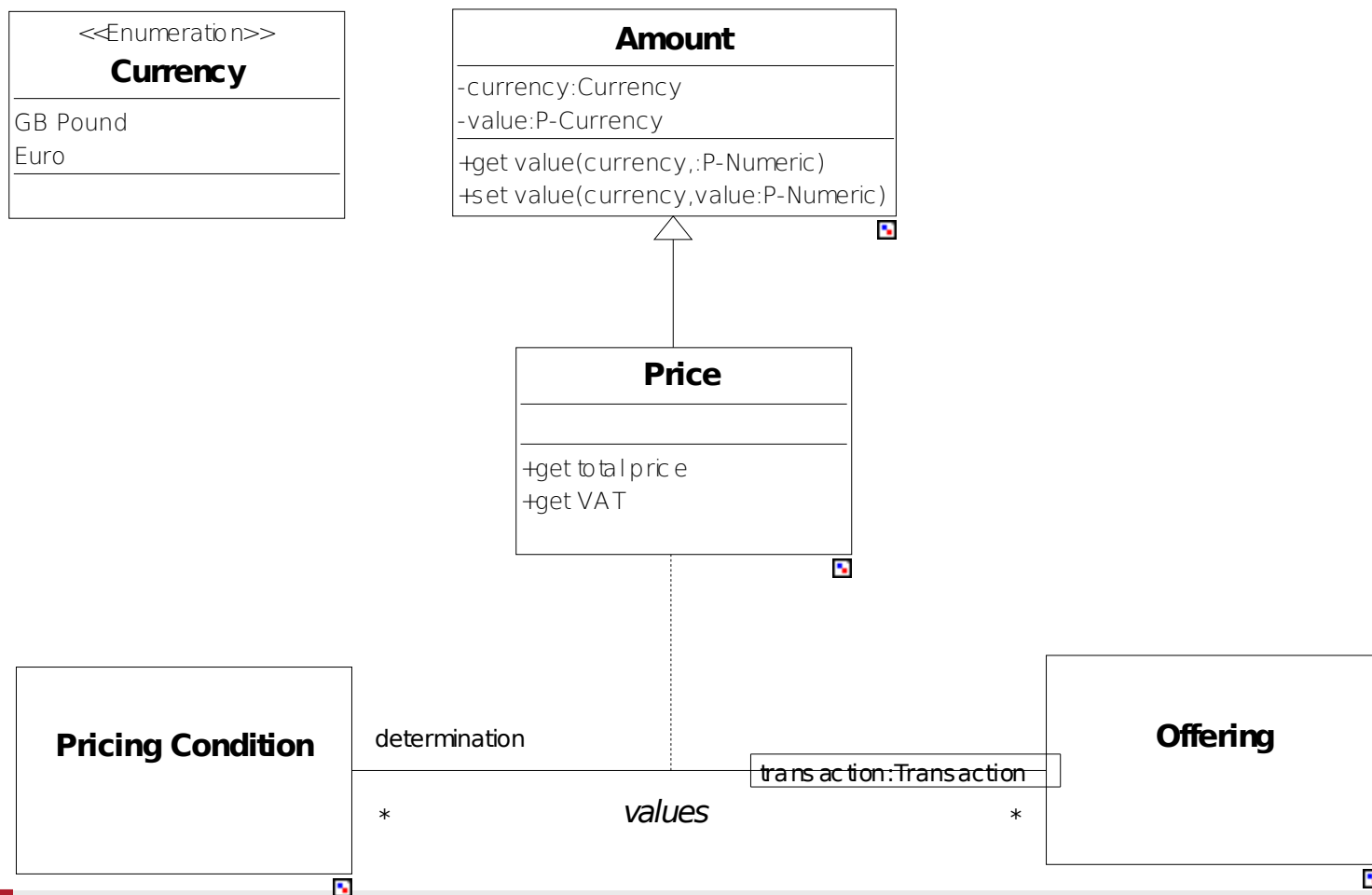
Illustration of Semantic Modelling

instance for the scenario to the UML diagram
c'est la "statique"
le commentaire déroule le scénario...
(si on préfère, possible de compléter par un schéma dynamique - hors projet)





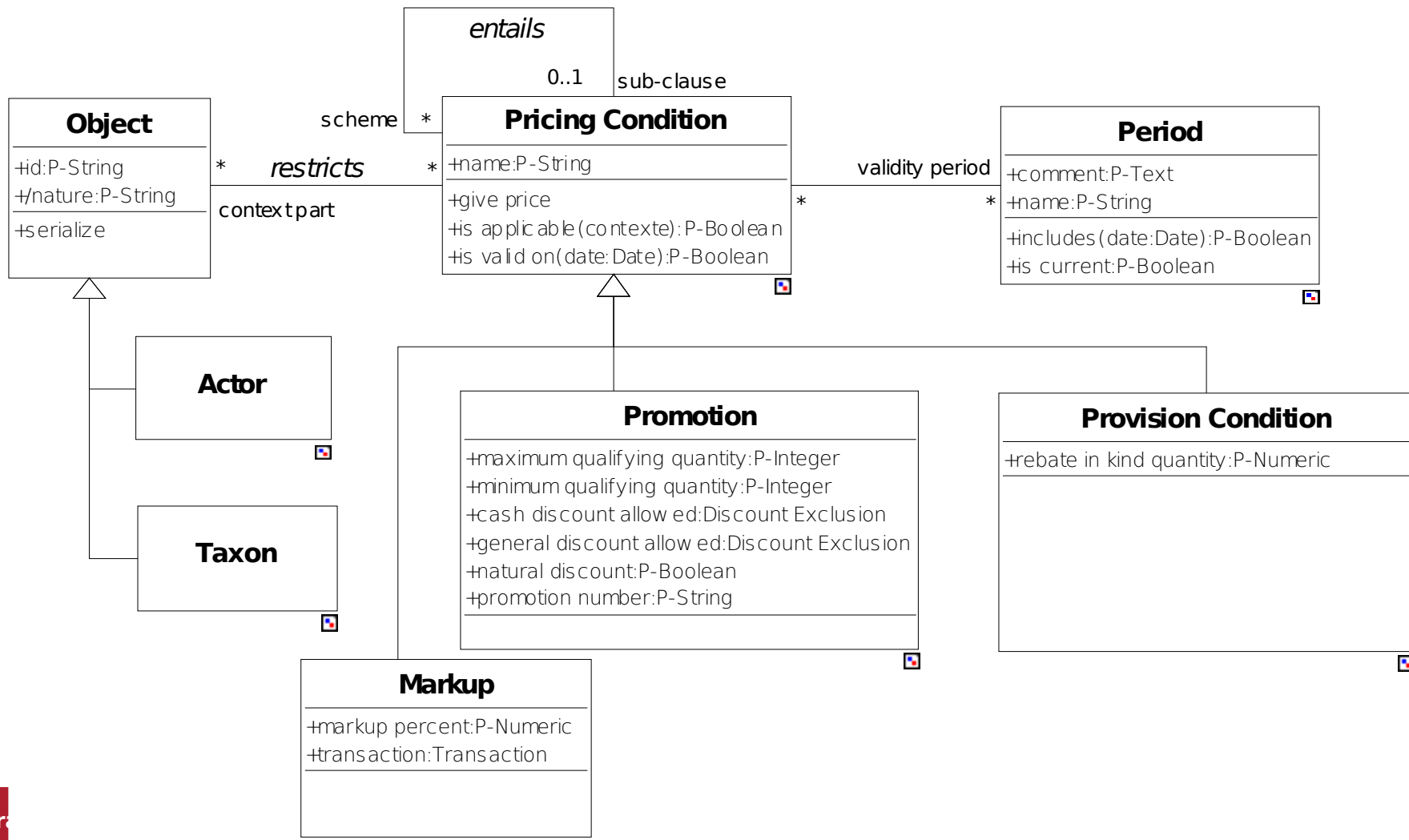
Pricing: the only way to record a price





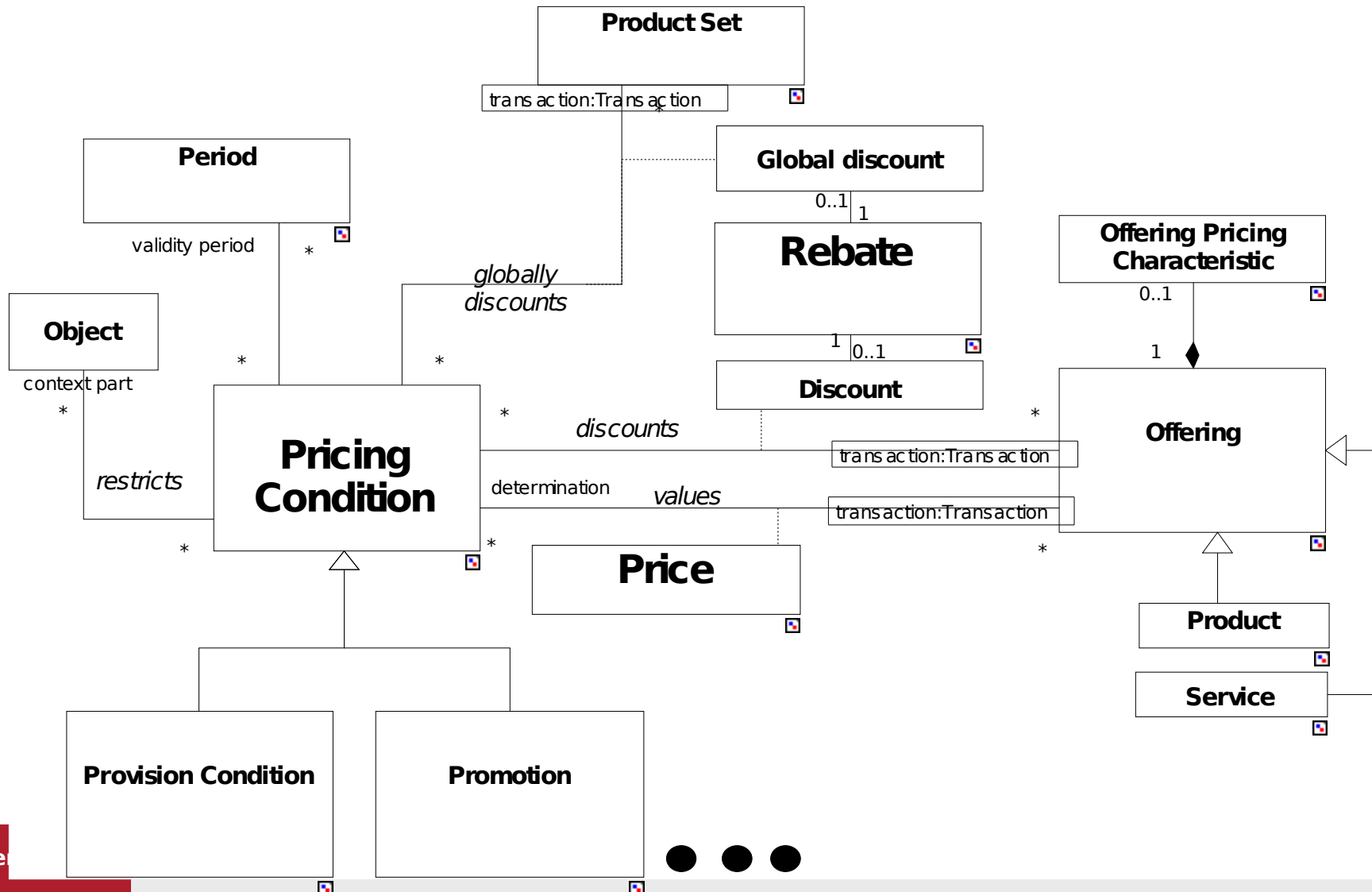
How to express pricing conditions

Semantic Modelin





Pricing: detailed solution





- **Goal of semantic modelling**
 - Capture the very meaning of the business
 - Help the convergence
- **Characteristics of semantic models**
 - Expressiveness
 - Genericity
 - Adaptability
- **Object-oriented approach *versus* functional approach**
 - The case of pricing



Highlights on the approach

Semantic Modelling

- **Object-oriented philosophy**
 - *Versus* data-oriented approach
 - A condition for restoring the meaning
 - The model shows some operations as sample
 - The state machine expresses the dynamic
- **The quest for genericity**
 - Provide the more of services with the less of terms
- **Separation of concerns**
 - The “abstraction levels”
 - Get rid off the technical concerns (for a while)
 - Focus on the essential core
- **Modelling rules**



- **Standard notation**
 - UML (*Unified Modelling Language*)
- **The diagrams we used**
 - Class diagram
 - Object diagram
 - State diagram
- **Pre-modelling**
 - The data dictionary
 - The traceability
 - Reference to the collected documents





- **The logical aspect and the service oriented architecture**
 - Principles and methods
 - Services and data
 - Layers
 - Transformation
- **The by-products of the semantic model**
 - The Logical Data Model
 - Other possible transformations





Service Oriented Architecture

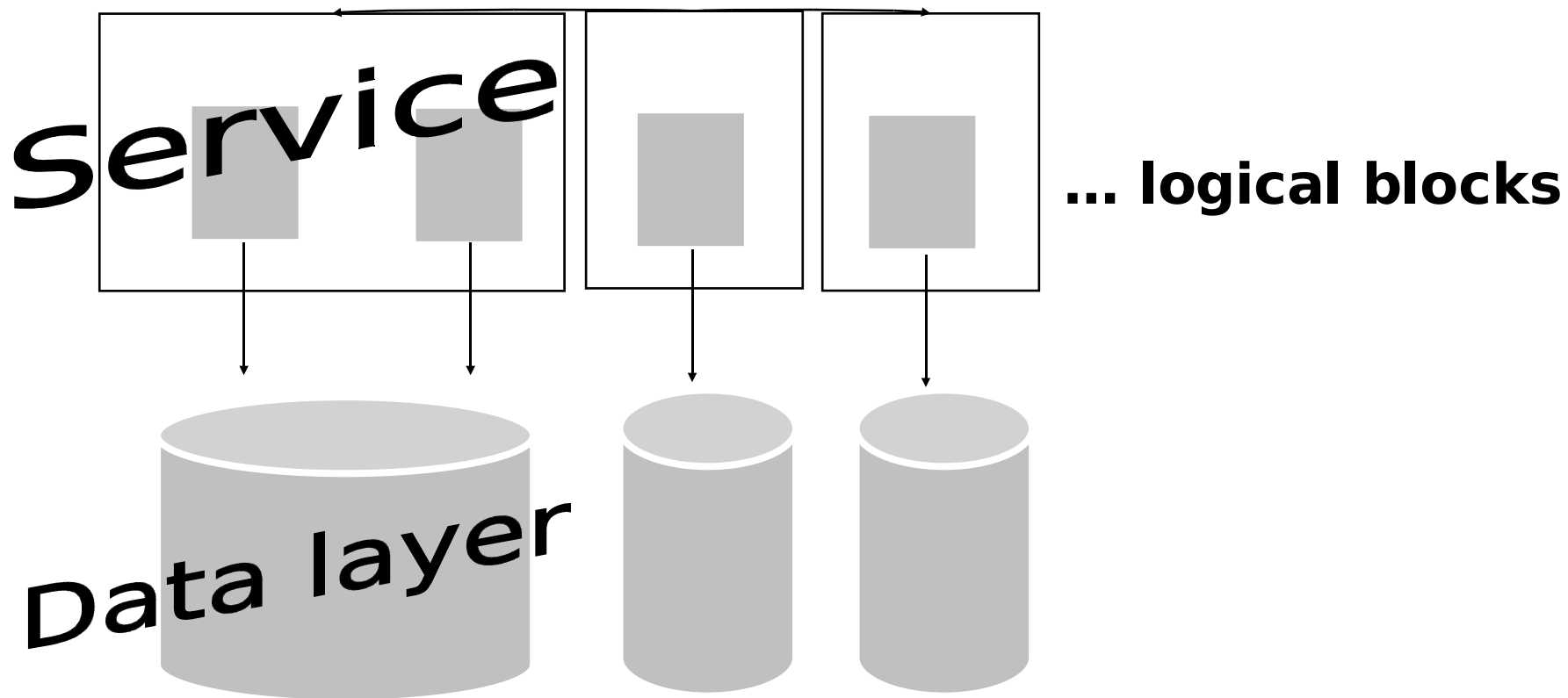
- 1. The Service layer masks the Data layer**
- 2. The Services are ordered in the logical architecture**
- 3. The Services are ordered in the logical architecture**
- 4. Conception of Services follows on from Urbanisation decisions**
- 5. Conception of Services follows on from Urbanisation decisions**
- 6. Services and data derive from model of the upper abstract level**
- 7. Services and data derive from model of the upper abstract level**
- 8. The transformation from an aspect to an other**
- 9. The transformation from an aspect to an other**

principles



Semantic Model Transformation

Encapsulating



The service layer hides the data layer



- **The Services are ordered in the logical architecture**
 - Several levels of composition

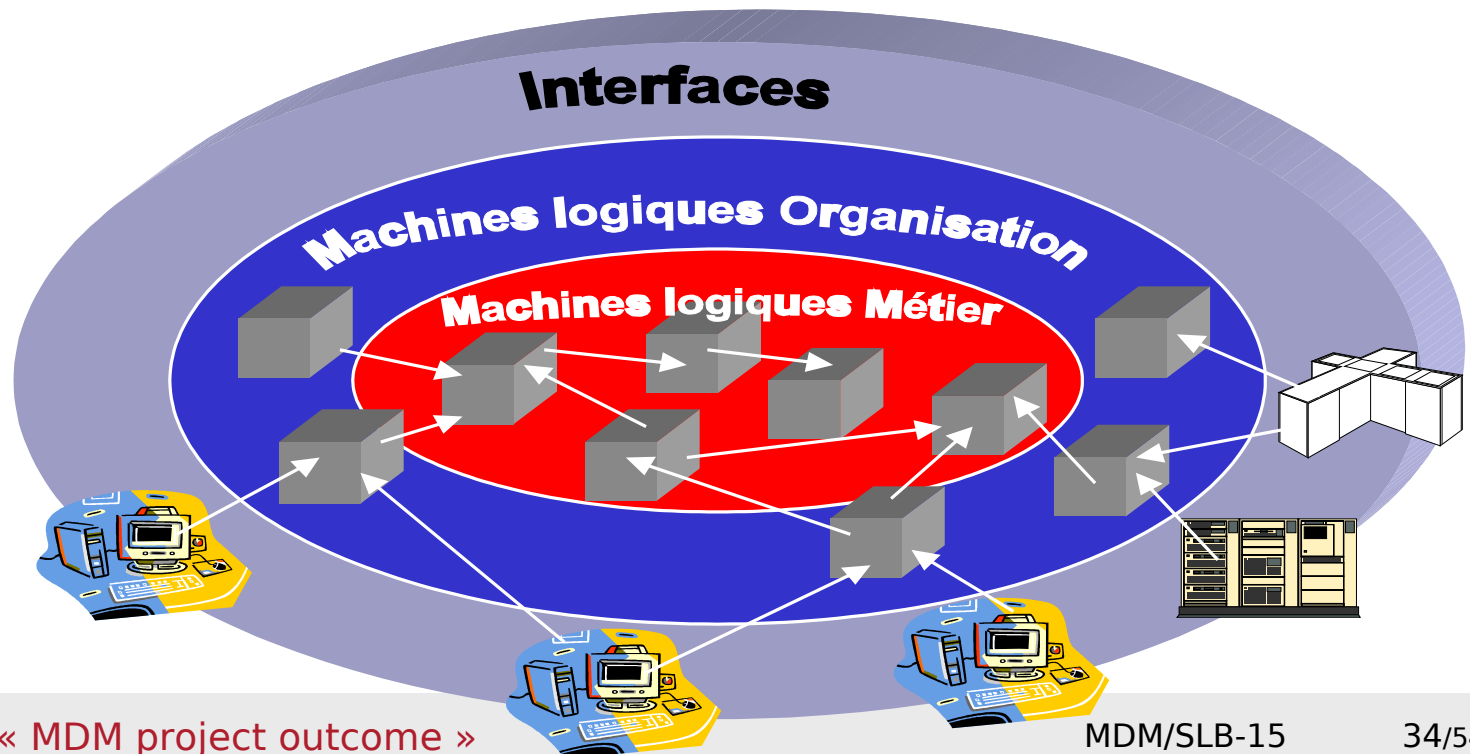




Semantic Model Transformation

Layering the System

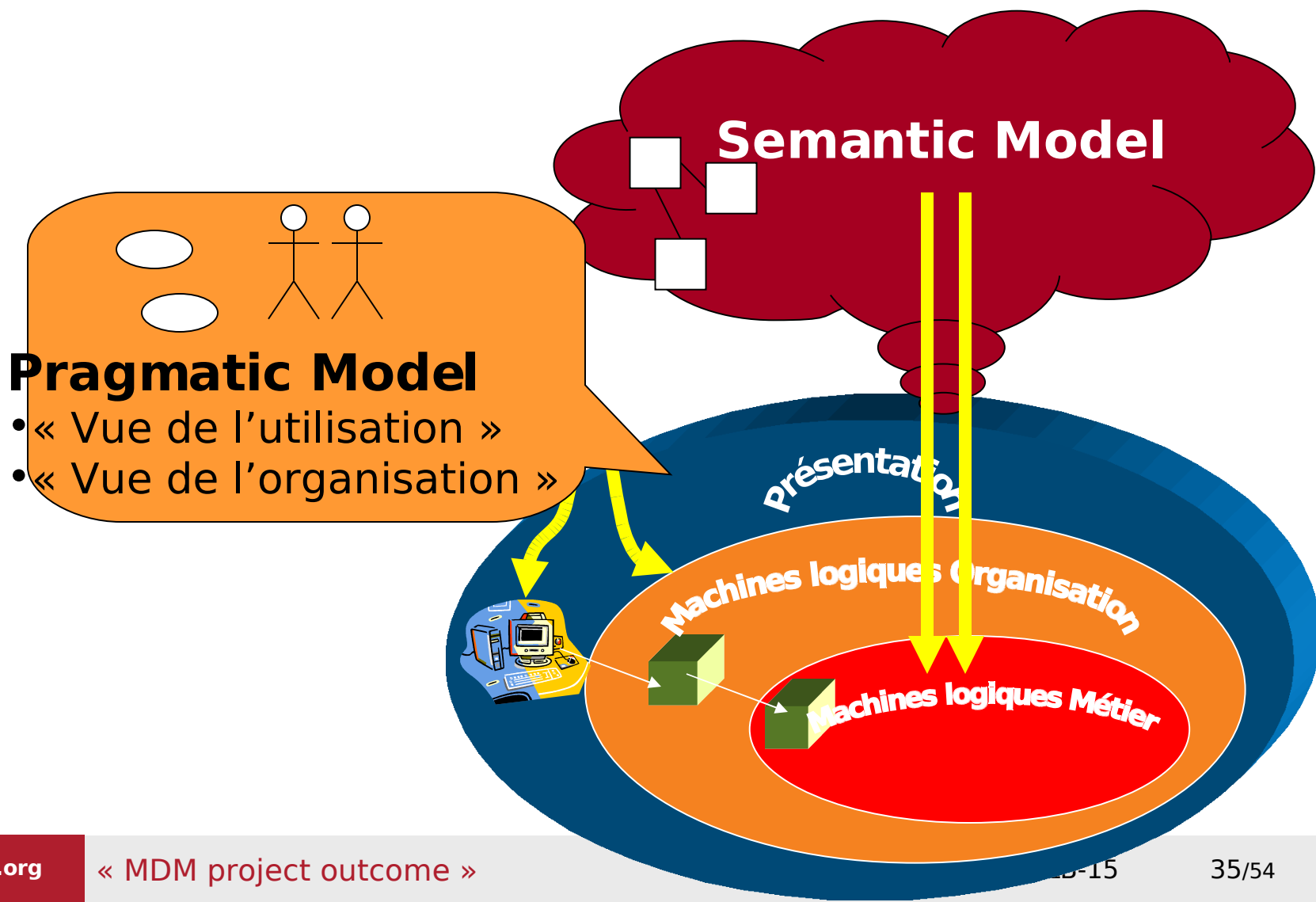
- **At a logical level**
 - Not to be confused with the technical architecture model





Semantic Model Transformation

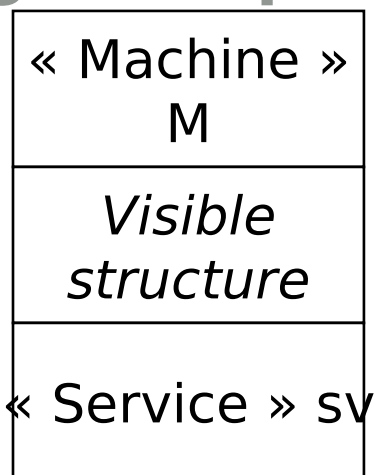
Origin of Logical Services





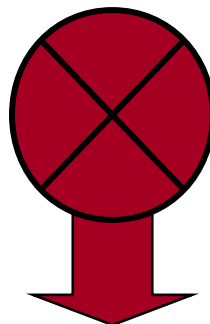
Semantic Model Transformation Chain (I)

Logical Aspect

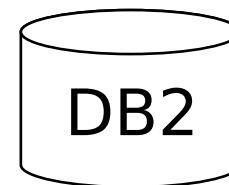
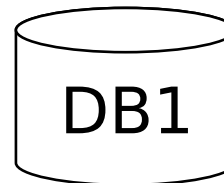
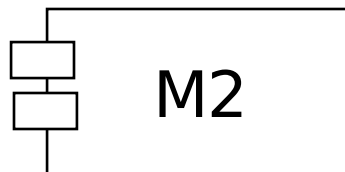
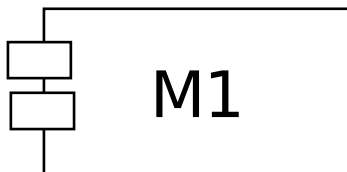


Technical Aspect

- A1: Java, Oracle...
 - A2: RPG, DB2...
 - Etc.
- + development rules



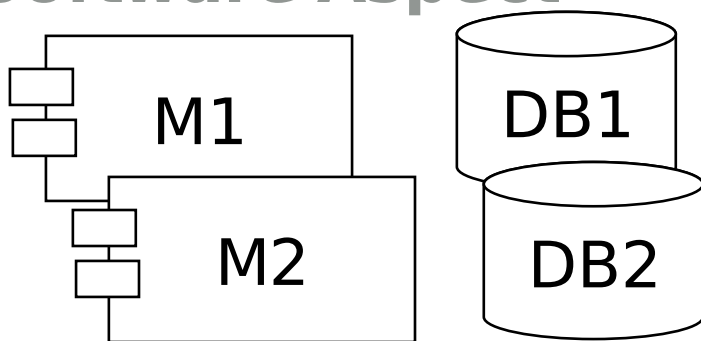
Software Aspect



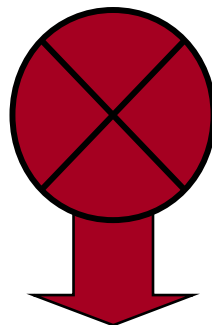
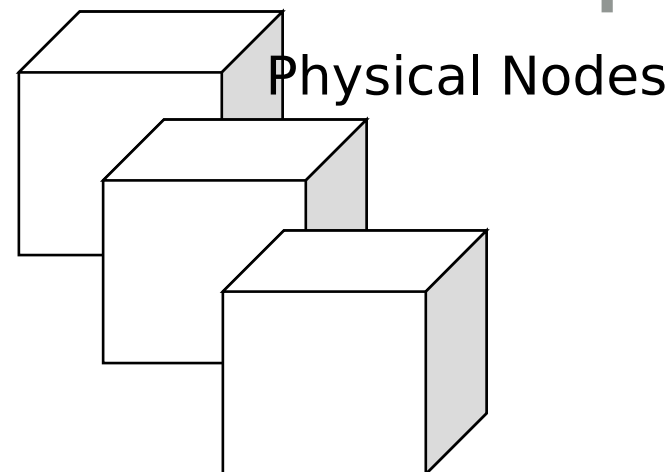


Semantic Model Transformation Chain (2)

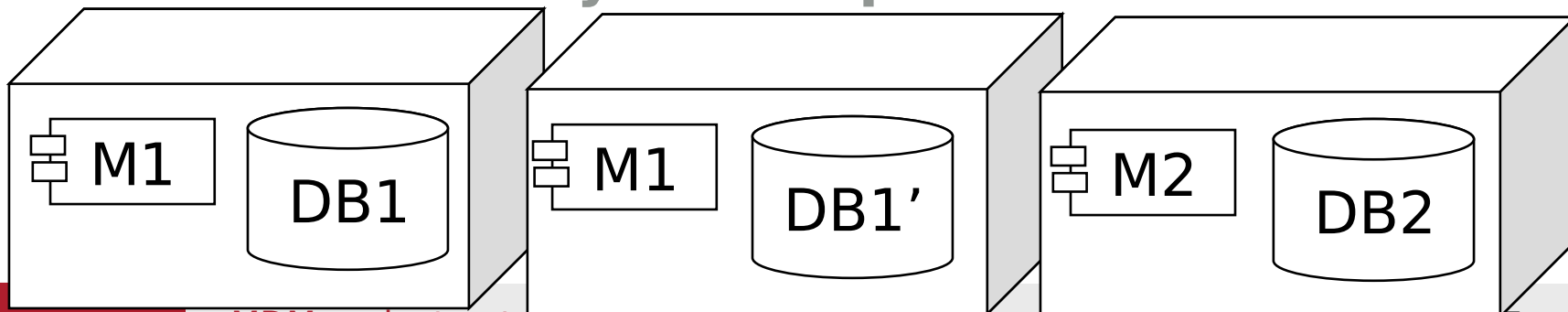
Software Aspect



Hardware Aspect



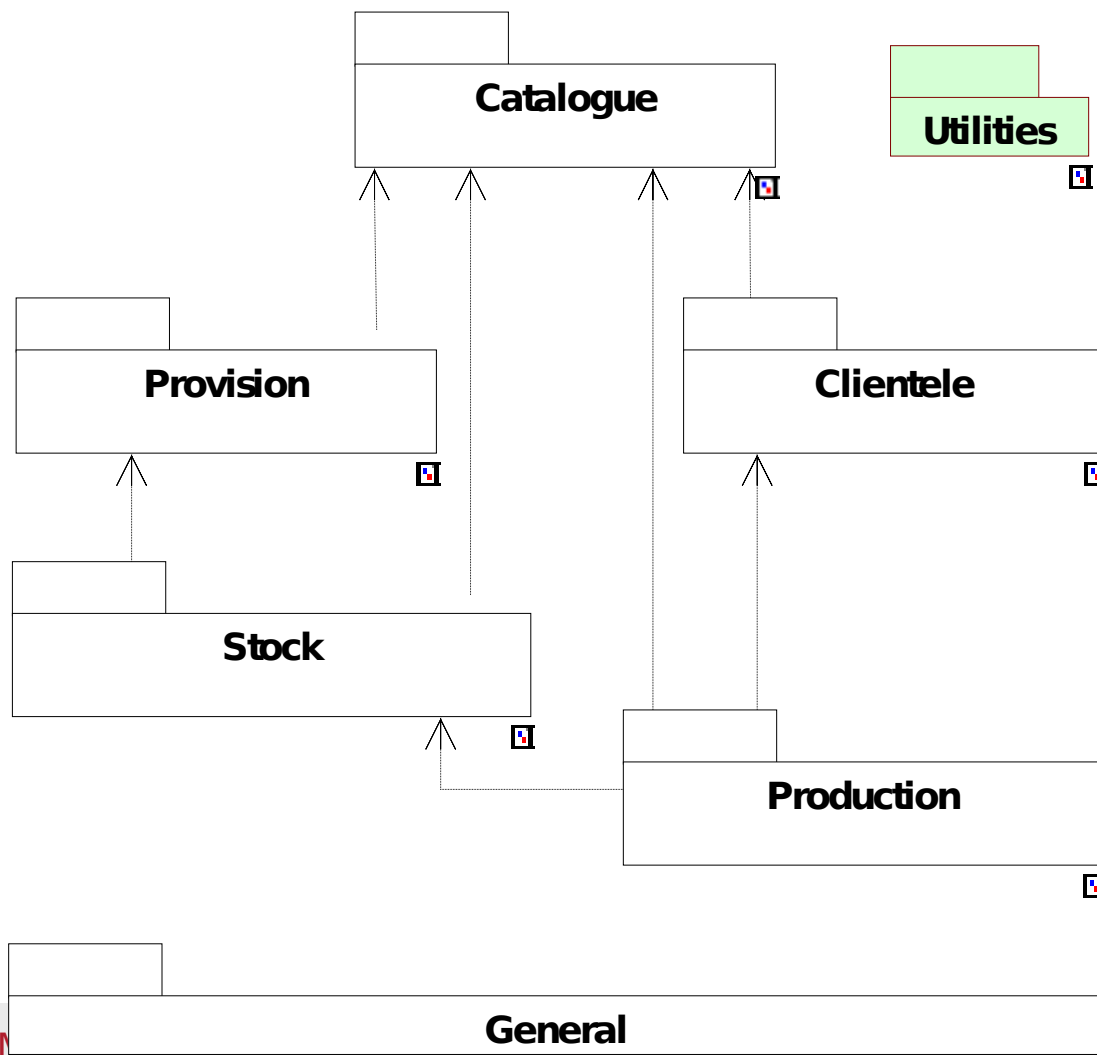
Physical Aspect





Semantic Model Transformation

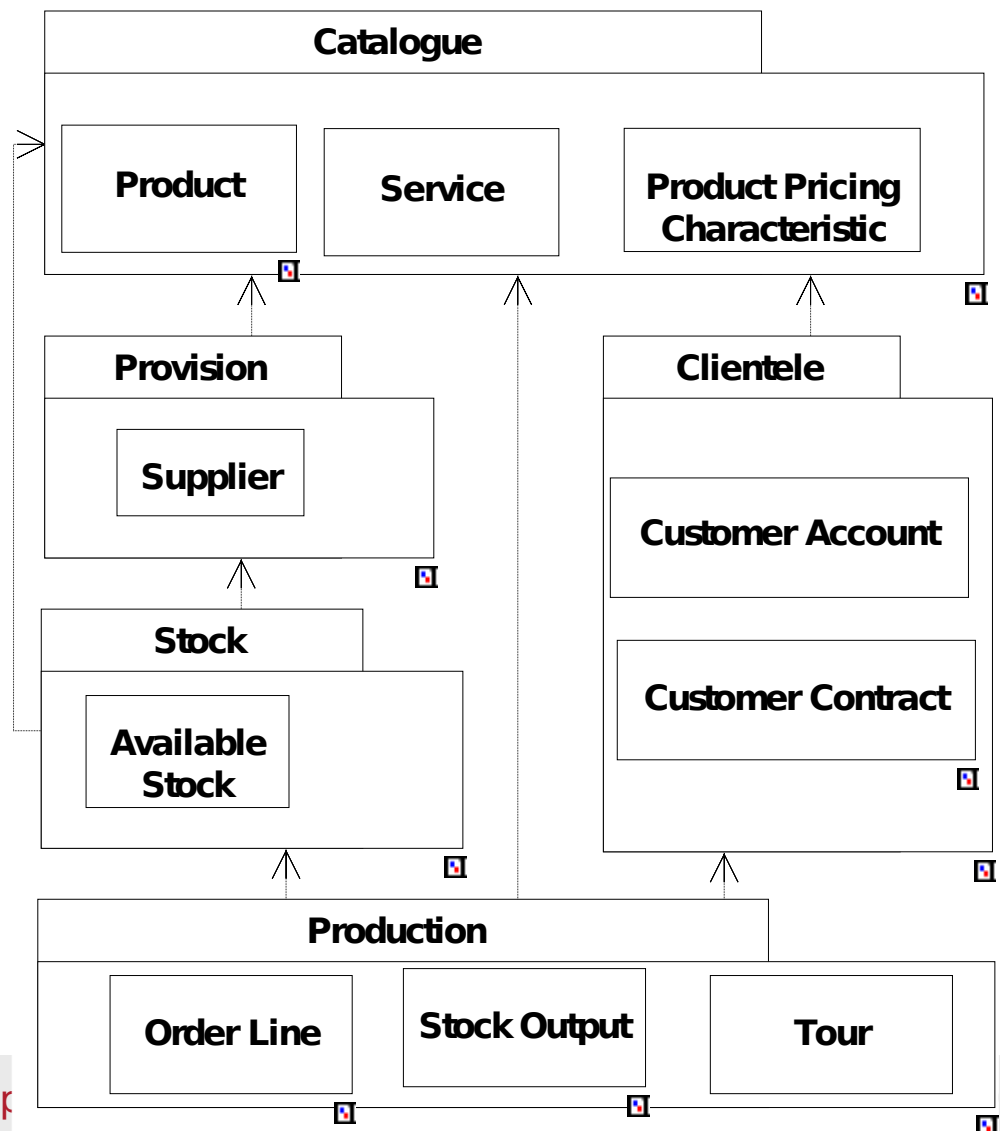
Logical Architecture Graph





Logical Architecture Graph

Semantic Model Transformation (detailed)





Semantic Model Transformation Subjects for the Logical Model

- **Logical / Technical discussions**
 - Examples
 - Transaction management
 - Error management
 - Event management
- **Regarding the modelling of the Logical Model**
 - Object identification
 - Keeping history traces
 - Data reconciling



Semantic Model Transformation Object Identification (1/3)

- **The problem**

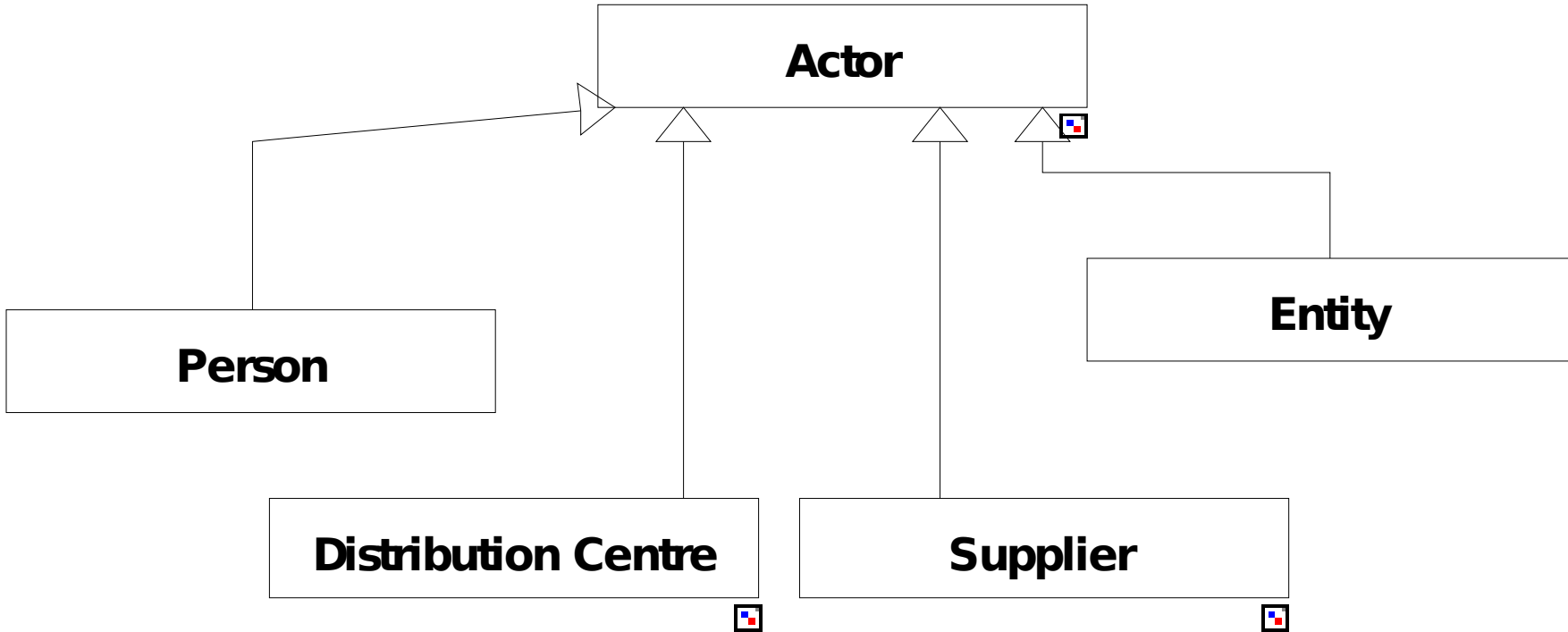
- A unique system for all countries
- A unique system for the classes inside the same inheritance tree



Semantic Model Transformation (2/3)

Object Identification (2/3)

- An example





▪ **The solution**

- A universal system for object identification
 - Valid for all countries
- This is a question for the logical design
 - Unless it receives a technical answer



Semantic Model Transformation

Keeping history traces (1/2)

- **On many concepts**
 - Creation date, update date and so on
 - User responsible for the action

Order Acquisition

```
+acquisition start time:P-Time  
+acquisition end time:P-Time  
+employee id:P-Numeric  
+employee name:P-String  
+employee short name:P-String  
+employee title:P-String  
+terminal id:P-String
```





Semantic Model Transformation Keeping history traces (2/2)

- **It is not relevant in the semantic model**
 - Risk of overloading the core model
- **Another issue for the logical design**



■ The problem

- Existing data in different structures
- Must converge to the future and unified system
 - Either feeding the future system
 - Or providing it with information as requested



- **Two options**
 - Conversion
 - Transforming data, migrating to the new database
 - One-shot, on a project-time basis
 - Reconciliation
 - Thanks to EAI possibilities
 - On a run-time basis
- **This is part of the logical/technical discussion**
 - It can be decided by parts
 - It must be set in the urbanisation roadmap



4

Contents of Part 4

- **Milestones**
- **Collected documents**
- **Continuing on post-MDM project**
 - Consolidation
 - Possible works to be done



| | |
|--------------------------------|---|
| Availability of Modelling Tool | Beginning of January |
| Document collection | Deadline: 7/01/05 Collection stopped: 7/02/05 Difficulties respecting perimeter |
| 1st Workshop | 14/01/05 |
| Data dictionary delivery | 2/02/05 |
| v2 Semantic Model delivery | 11/02/05 |
| 2nd Workshop | 16-17/02/05 |
| v3 Semantic Model delivery | 22/02/05 |
| Logical Data Model delivery | 7/03/05 |



■ Collected documentation towards Semantic Model

Fields

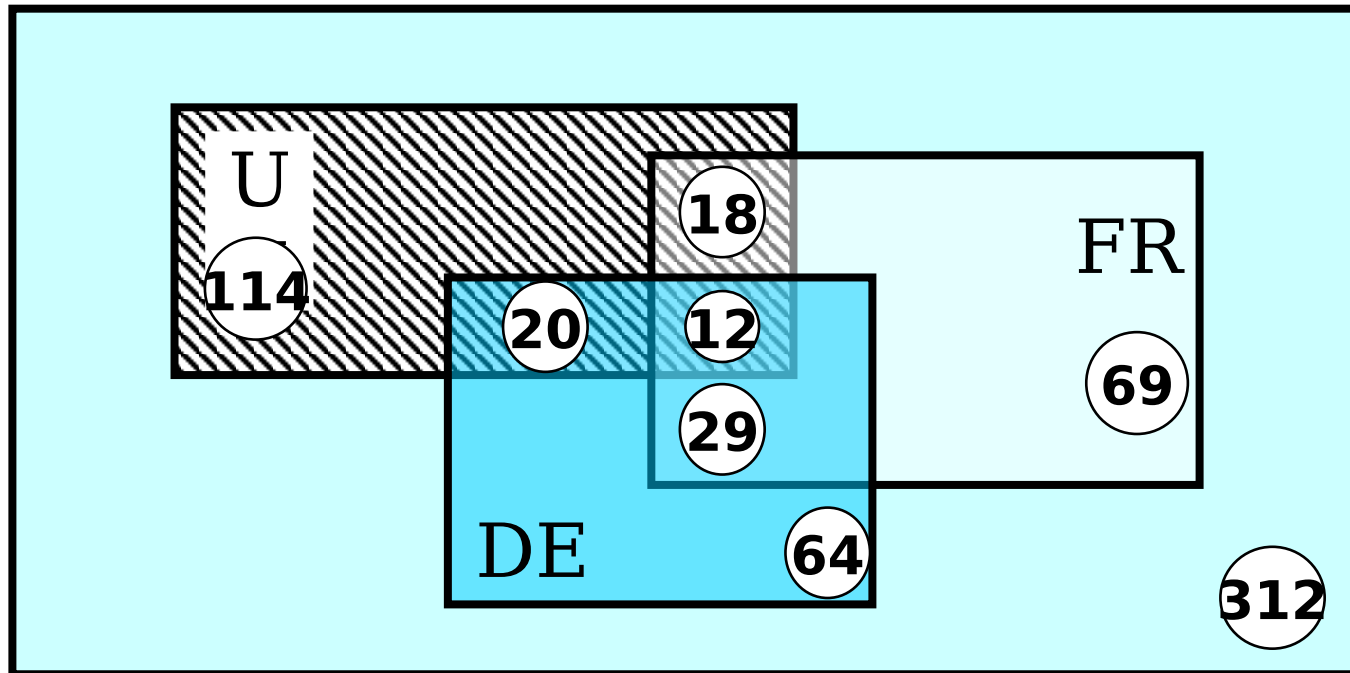
| | | |
|--------------|-------|-----|
| Out of Scope | 940 | 45% |
| Issue | 178 | 9% |
| No Link | 498 | 24% |
| Link | 473 | 23% |
| | <hr/> | |
| | 2089 | |

- The accuracy of the collected documents has been poor
 - See '[metric.xls](#)' file for details



Convergence Metrics The Plan

- The Semantic Model attributes

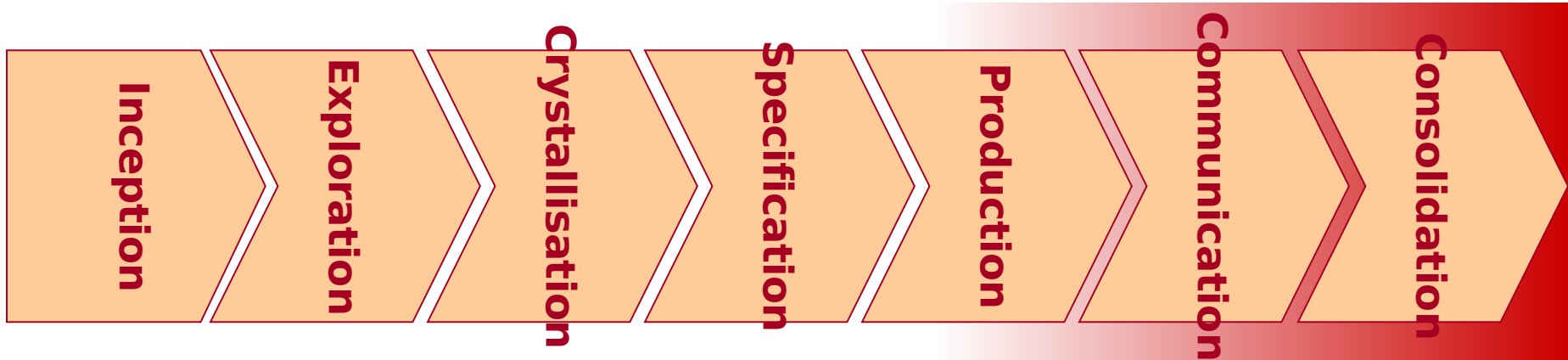




Continuing on post-MDM

The Plan project

- **Explaining and maintaining the semantic model**
 - The MDM project ends with the delivery of the logical data model
 - It can carry on with the « Consolidation phase »
 - Change process on SEM and LDM





- **On the semantic model**
 - Complete the State machines
 - States & Transitions
 - Identification of all the Operations
 - Before deriving the services
 - Complete & improve upon comments
 - More explanations
- **From the use case description**
 - Bridge the Use Case View and the SEM
 - By means of connections to the relevant modelling elements
 - By means of new diagrams
 - Structure the Use Case View
 - By eliminating redundancy



- **Outcome of the MDM project**
 - Benefits of Semantic Modelling
 - Convergence
 - Simplification
 - A start for the business repository
 - Benefits for the IS
 - A source for deriving the services in SOA
- **Planning, means and budget**
 - Kept on track
 - Absolute need for a UML tool
- **Difficulties**
 - Tools
 - Cultural change
 - State of the input documentation
 - Scope



Praxeme, the sense of action
An initiative for a public method

Semantic Modeling & Data Management

MDM Project Outcome
5/04/05

✉ info@praxeme.org

☎ +33 (0) 6 77 62 31 75

🌐 <http://www.praxeme.org>

Référence :
MDM/SI B-15

Version : 5/04/05



Objectives

- **Explain MDM Project approach**
 - Why build a Semantic Model before Logical Model
 - How to exploit the Semantic Model
- **Learn from the experience**
 - Difficulties encountered
 - Best practices
- **Draw up the project outcome**
 - Benefits of Semantic Modelling
 - Benefits for the IS

Benefits other than the Logical Data Model.



Contents

- **The Approach**
 - The theoretical basis
 - Process
- **Semantic Modelling**
 - Principles & objectives
 - Illustration
 - Benefits
- **Semantic Model Transformation**
 - Logical Data Model
 - Service Oriented Architecture
- **The Plan**
 - The story of the Project
 - Continuing on post-MDM project



Agenda

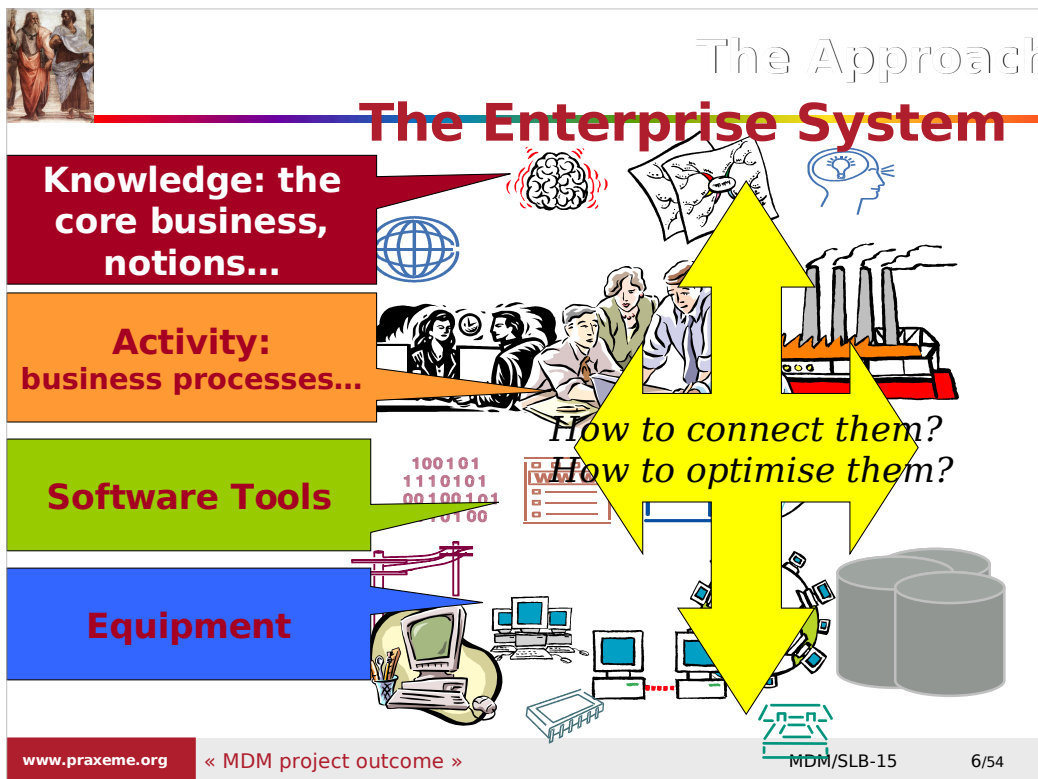
| Topic | Time |
|-------------------------------|----------------------|
| The Approach | 17pm - 17:15 |
| Semantic Modelling | 17:15 - 17:30 |
| Semantic Model Transformation | 17:30 - 17:45 |
| The Plan | 17:45 - 18pm |



1

Contents of Part 1

- **"Aspects" of the Enterprise System**
 - Theoretical basis to the approach
- **Principles & objectives of Semantic Modelling**



Le Système Entreprise

Les éditeurs d'outils d'optimisation ou de mesure d'infrastructure façonnent un discours qui cherche à « remonter » leur offre vers l'aspect processus. En filigrane, le destinataire doit comprendre qu'optimiser l'infrastructure c'est optimiser les processus !

Cette attitude, aux motivations évidentes, conduit à relier les différents plans de la réalité des entreprises.

Ce schéma vise le même objectif.

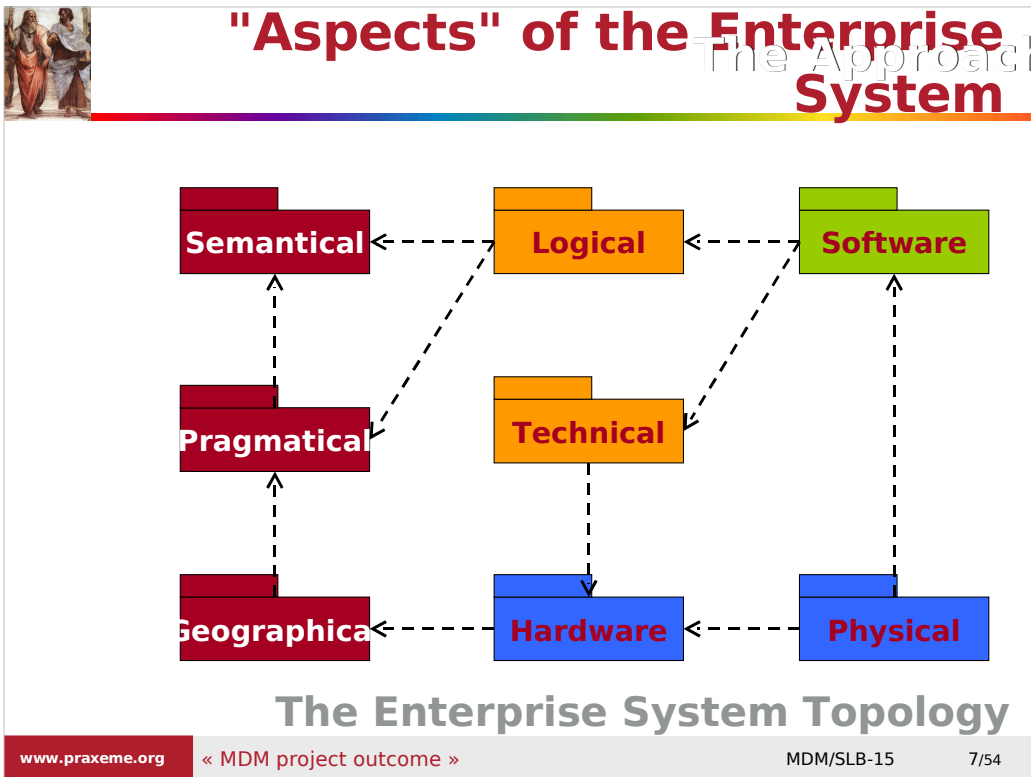
Il constitue une 1ère approche, intuitive du « Système Entreprise », sur lequel nous voulons agir.

Une 2ème approche, plus systématique, est formulée par la Topologie du Système entreprise (voir plus loin).

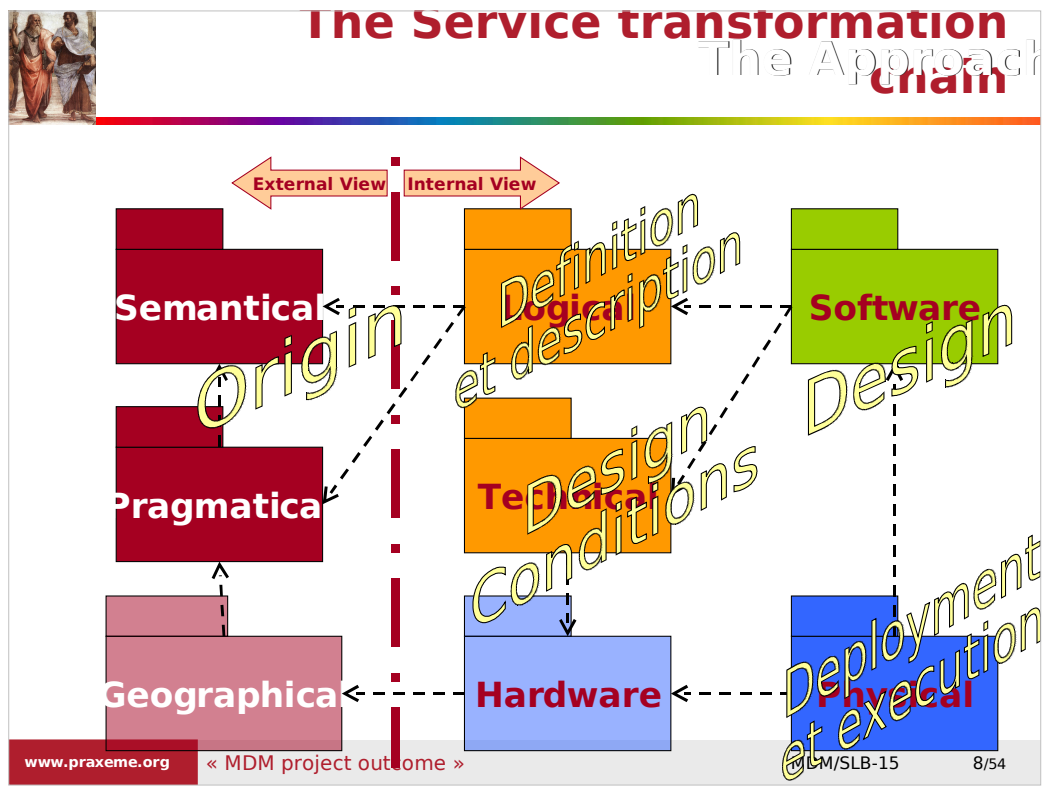
Dans une 3ème approche, l'analyse détaillée des aspects permet de distribuer les objets et notions sur les aspects.

Le souci est de relier les différents éléments d'information et les décisions, à travers tous les aspects : de la connaissance métier à l'infrastructure matérielle, en passant par les processus et la localisation.

Cette clarification des aspects constituant le Système entreprise est un préalable à l'action. On ne saurait optimiser sans connaître l'objet entreprise dans toutes ses dimensions.



Animation de la topologie : point de départ = « Aspect Logiciel »





The MDM context

- **A multiple aspect approach**
- **The MDM project respects the requirements:**
 - Provide signification to the data
 - Convergence on a data perspective
- **To do this, the approach is based upon the following principles:**
 - Abstraction
 - Conceptual view
 - Signification
 - Define the modelling elements as rigorously as possible
 - Structure
 - Improve the quality of the system: encapsulation, limited coupling...
 - Prepare for a Service Oriented Approach



Approach Products

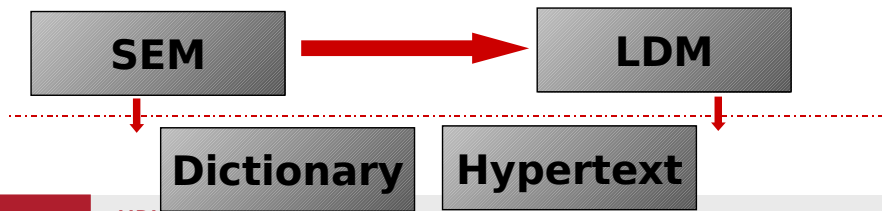
1. The final deliverable is the Logical Data Model

- Scope: needs of the CORSO project
- LDM: Relational Model

2. An intermediate deliverable: the Semantic Model

3. Other derived products

- To facilitate the use of the models



Le produit final est le Modèle Logique des Données du domaine étudié

Critère de délimitation : besoins du projet CORSO

MLD : format entités-relations (modèle relationnel)

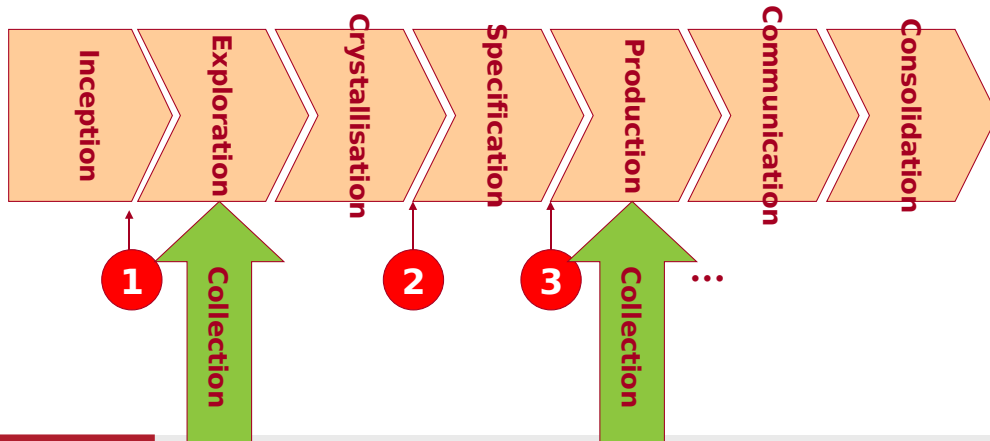
Sa production passe par un produit intermédiaire : le Modèle Sémantique

Des produits dérivés augmentent l'exploitabilité



The Process

▪ "Innovation" Approach



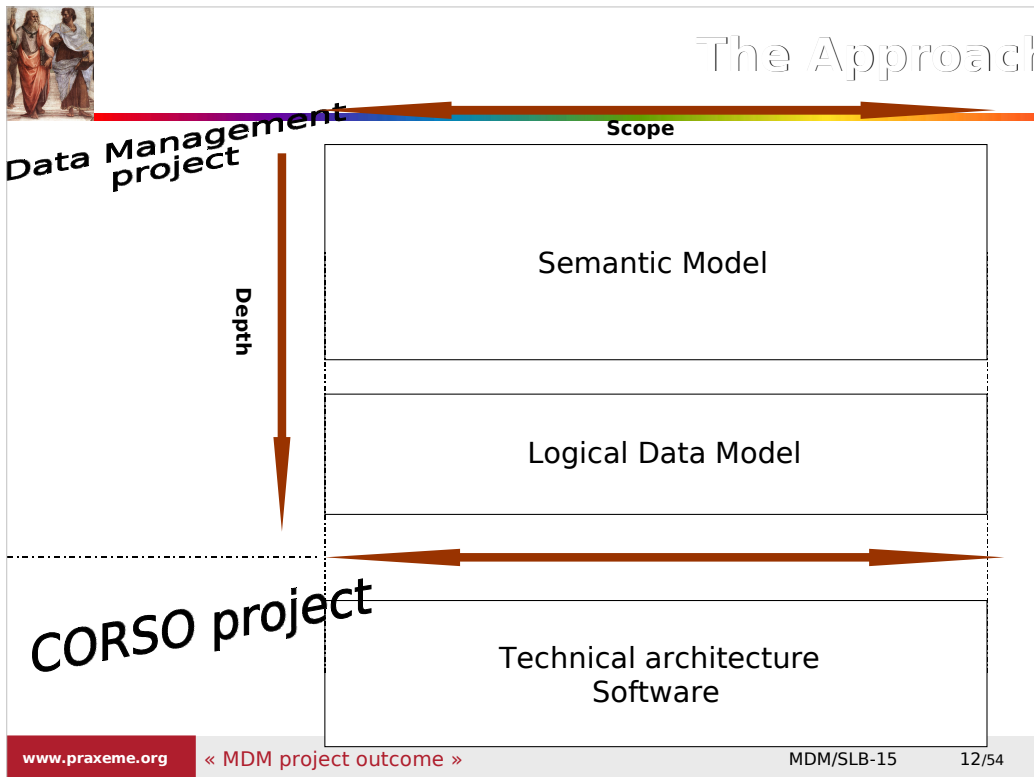
La démarche « Innovation »

Pour tenir compte du contexte

Réduction progressive de l'incertitude

Phasage fondé sur la logique de l'action

Plutôt que celle du produit



1st message:

The scope of the Data Management project is strictly the scope of the Corso project.

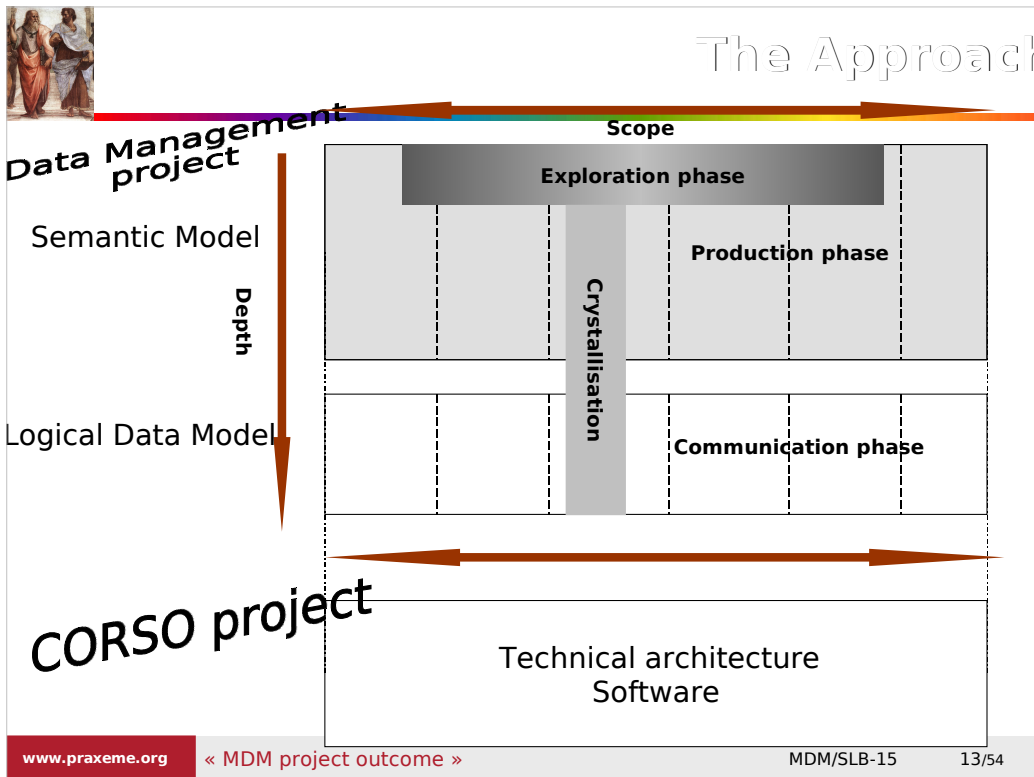
That's the relevance criteria for the information to be collected.

2nd message :

The building of the data model will proceed in two steps.

Firstly, an intermediate product will capture the meaning and constraints related to the business concepts. That is the Semantic Model.

Secondly, the Semantic Model will be transformed into e Logical Data Model.



Message 1:

The studied field is broken down into domains of objects. The structure will be discussed later.

Normally, the logical model matches the semantic structure: that's an assurance of quality and easy evolution.

Message 2:

During the Exploration phase, we intend to cover a great part of the field, let us say 80%, but with a tiny depth (20%).

The Crystallisation phase is meant to elaborate and demonstrate the detailed operational process and the kind of products you can expect. So, like a prototype, the coverage is narrow but digs deeply.

The whole field will be covered during the Production phase.

The Semantic model will result of this work, delivered by domain along the phase, then consolidated before delivery.

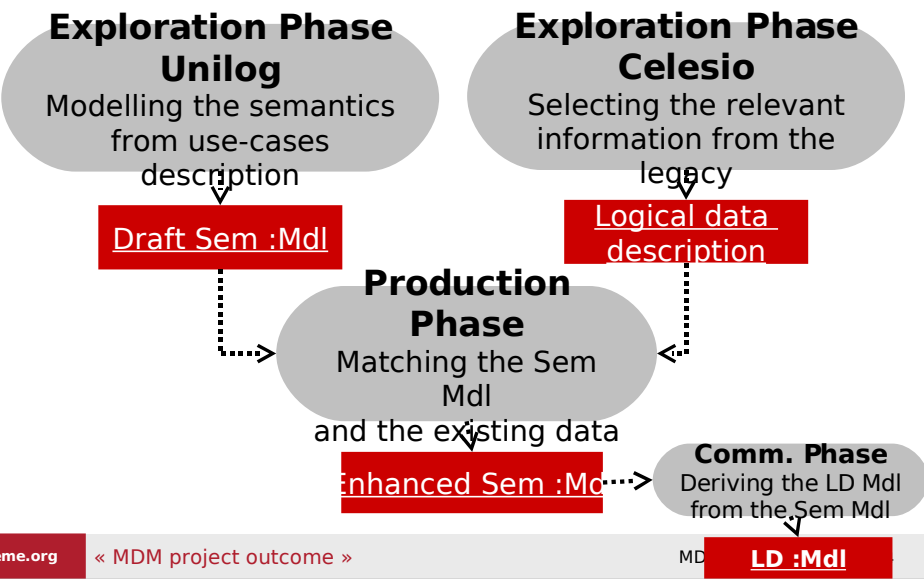
Message 3:

The Logical Data Model will derive from the Semantic Model. It will be produced by the Communication phase, as other derived products.



The Approach

Exploration Phase



Semantic Model and Use Case View
The Approach View

- **One of the representation inside the « Pragmatical » aspect**
 - Functional approach
 - Local point of view

The diagram illustrates the following components and relationships:

- Semantical** (dark red box) and **Pragmatical** (dark red box) are connected by a dashed double-headed arrow.
- Logical** (red box) is connected to **Semantical** by a solid arrow pointing up.
- Pragmatical** is connected to **Organisation View** (blue arrow) by a dashed arrow pointing down.
- Organisation View** is supported by three blue upward-pointing arrows.
- Organisation View** is connected to **... Use Case View** (text) by a dashed arrow pointing right.
- ... Use Case View** is connected to **Pragmatical** by a dashed arrow pointing left.

www.praxeme.org « MDM project outcome » MDM/SLB-15 15/54

Functional not object-oriented
Local not global



Method deliverables

- **User's Guide to Semantic Model**
 - MDS03-UserGuide.doc
- **Templates for the collected documentation**

**2****Semantic Modelling**
Contents of Part 2

- **Principles and objectives of semantic modelling**
- **Difference between classical and semantic modelling**
- **Illustration**
 - Genericity
 - Transfer order

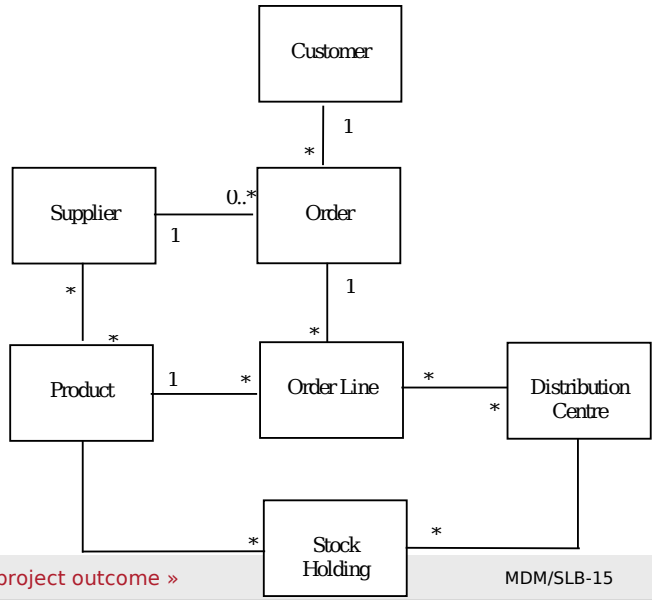


Principles & Objectives

- **Format**
 - UML language: class diagram, state diagram & collaboration diagram
 - Data Dictionary
 - Synonyms, keywords, definitions and comments
- **High level of abstraction**
 - Improves expressiveness
 - Provides a basis for modelling for follow-on projects
 - Data oriented (with an object-oriented approach)
 - Reminder: It is not be a “complete” model (few operations and constraints will be captured)

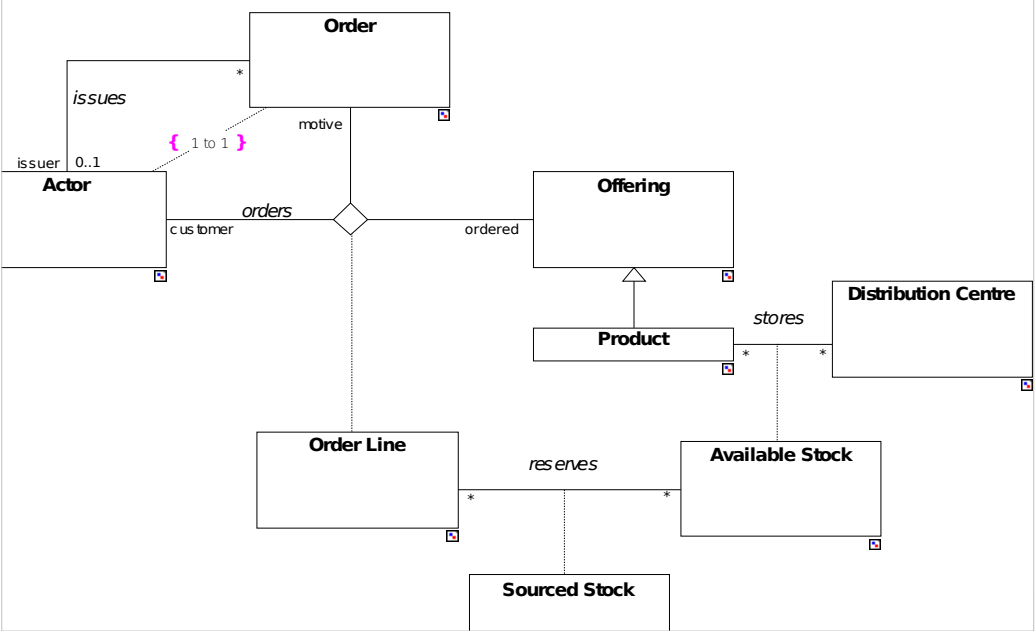


From classical data models... Semantic Modelling





...towards Semantic Modelling





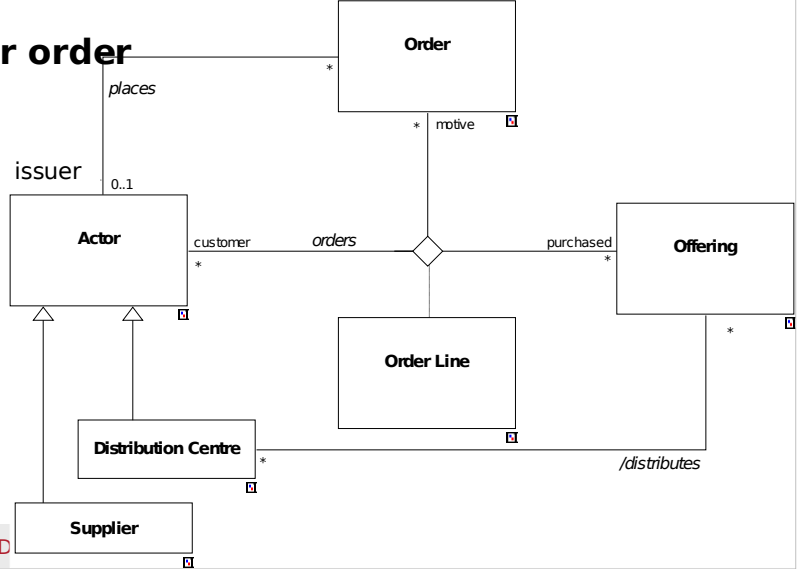
Semantic Modelling Generativity

- **The generic concept « Offering »**
 - Allow to order services as well as product by the same process
 - Avoid structural redundancy
 - Factorizing the common properties
- **The generic concept of actor**
 - Customer, Supplier, Entity, Person...
 - A given actor can act in many ways
 - Both as customer and supplier
 - As a worker...
 - A supplier or a DC can order products



Genericity (continued)

Transfer order



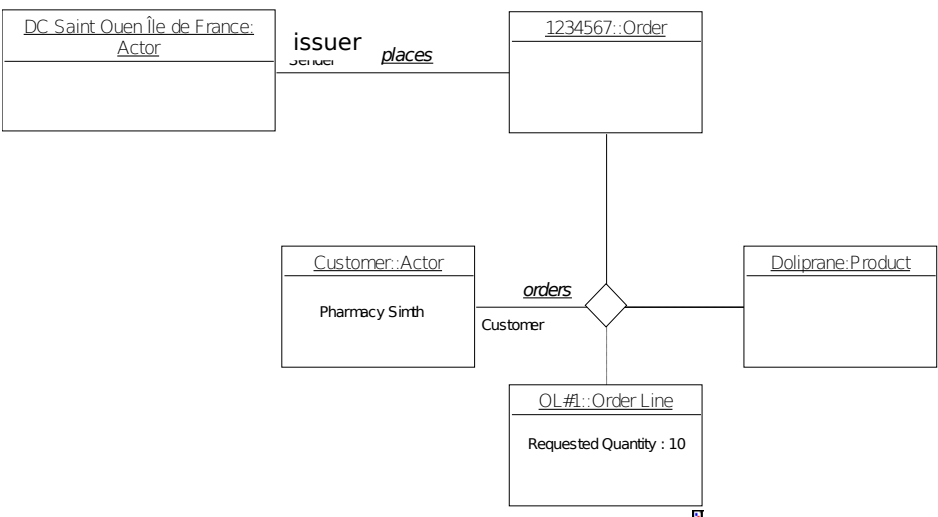
www.praxeme.org

<< MD



Illustration of Semantic Model

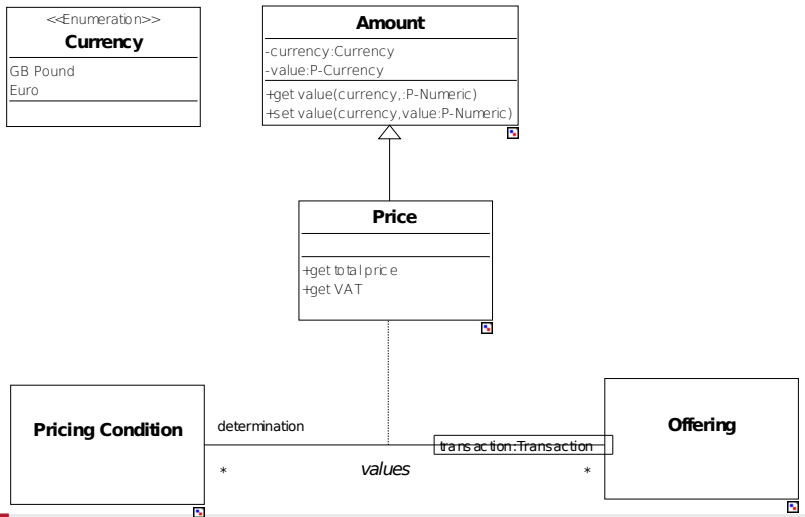
instanciation d'un scénario de dialogue
 c'est la "statique"
 le commentaire déroule le scénario...
 (si on préfère, possible de compléter par un schéma dynamique - hors projet)

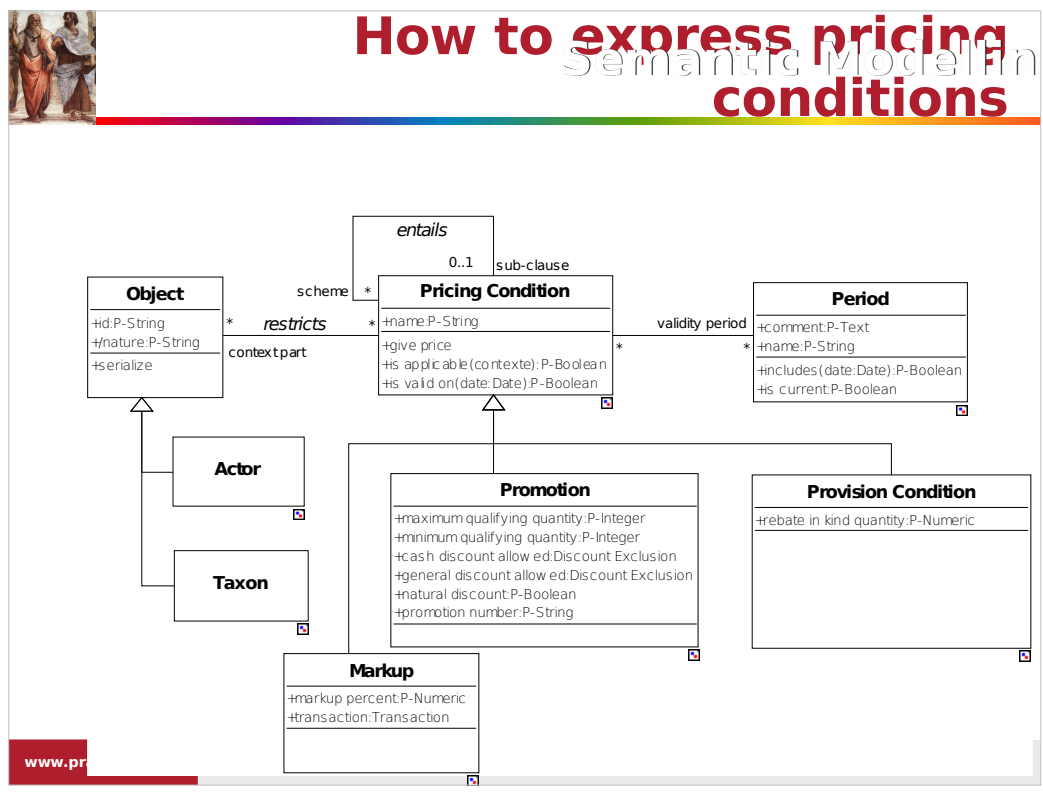




Pricing: the only way to record a price

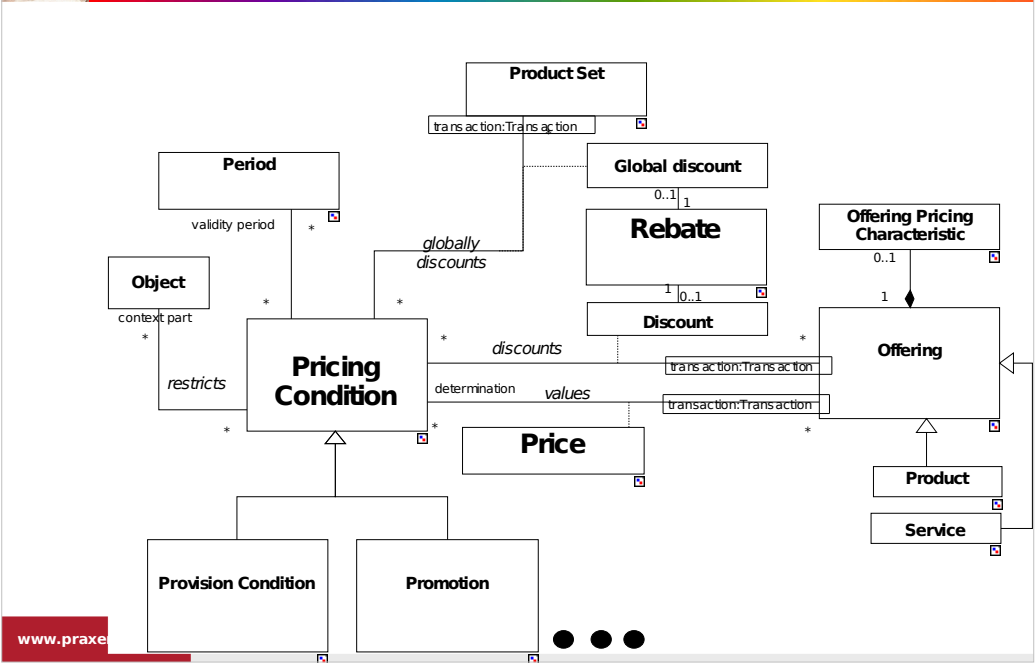
Semantic Modelling







Pricing: detailed solution



www.praxel.com



Semantic Modelling Benefits

- **Goal of semantic modelling**
 - Capture the very meaning of the business
 - Help the convergence
- **Characteristics of semantic models**
 - Expressiveness
 - Genericity
 - Adaptability
- **Object-oriented approach *versus* functional approach**
 - The case of pricing

Case of pricing : plutôt que de développer une fonction “gestion du catalogue” (prix de vente) puis une fonction pour l’approvisionnement (prix à l’achat)

→ Même solution (du moins pour le coeur du système)



Highlights of the approach

- **Object-oriented philosophy**
 - *Versus* data-oriented approach
 - A condition for restoring the meaning
 - The model shows some operations as sample
 - The state machine expresses the dynamic
- **The quest for genericity**
 - Provide the more of services with the less of terms
- **Separation of concerns**
 - The “abstraction levels”
 - Get rid off the technical concerns (for a while)
 - Focus on the essential core
- **Modelling rules**

Rendre le maximum de services avec le minimum de concepts

Se concentrer sur l'essentiel pour exprimer le métier



Semantic Modelling

- **Standard notation**
 - UML (*Unified Modelling Language*)

- **The diagrams we used**
 - Class diagram
 - Object diagram
 - State diagram

- **Pre-modelling**
 - The data dictionary
 - The traceability
 - Reference to the collected documents





3 Semantic Model Transformation

Contents of Part 3

- **The logical aspect and the service oriented architecture**
 - Principles and methods
 - Services and data
 - Layers
 - Transformation
- **The by-products of the semantic model**
 - The Logical Data Model
 - Other possible transformations





Semantic Model Transformati

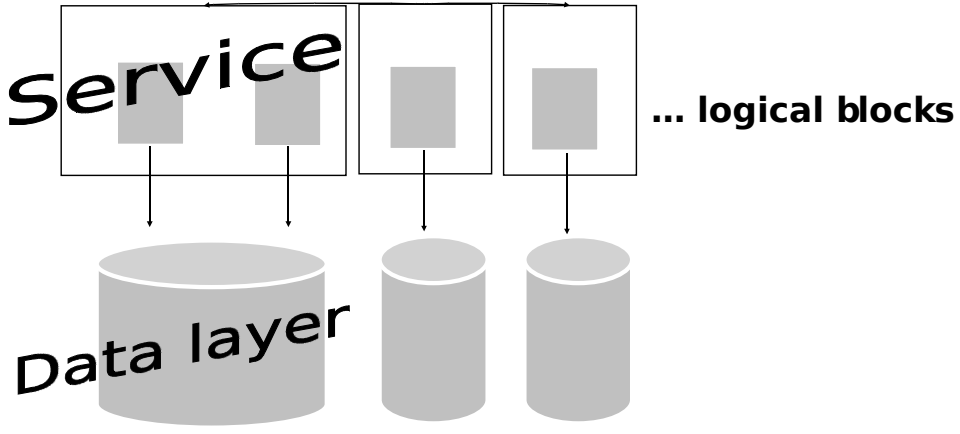
Service Oriented Architecture principles

- 1. The Service layer masks the Data layer**
- 3. The Services are ordered in the logical architecture**
- 5. Conception of Services follows on from Urbanisation decisions**
- 7. Services and data derive from model of the upper abstract level**
- 9. The transformation from an aspect to an other**



Semantic Model Transformation

Encapsulating



The service layer hides the data layer



Semantic Model **Structuring**

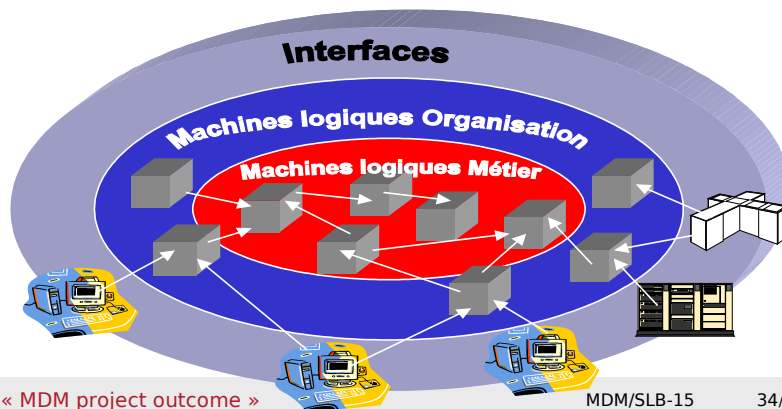
- **The Services are ordered in the logical architecture**
 - Several levels of composition





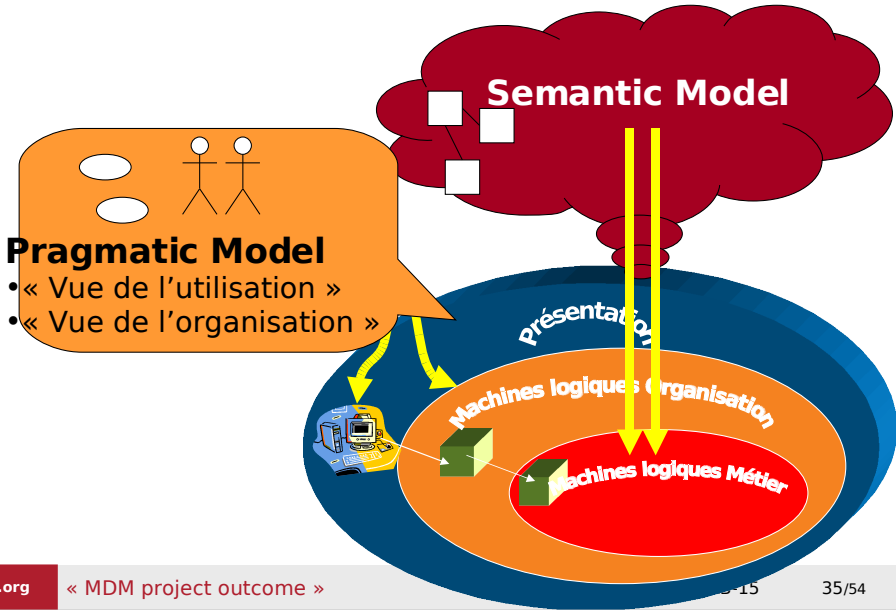
Semantic Model Transformation Layering the System

- **At a logical level**
 - Not to be confused with the technical architecture model



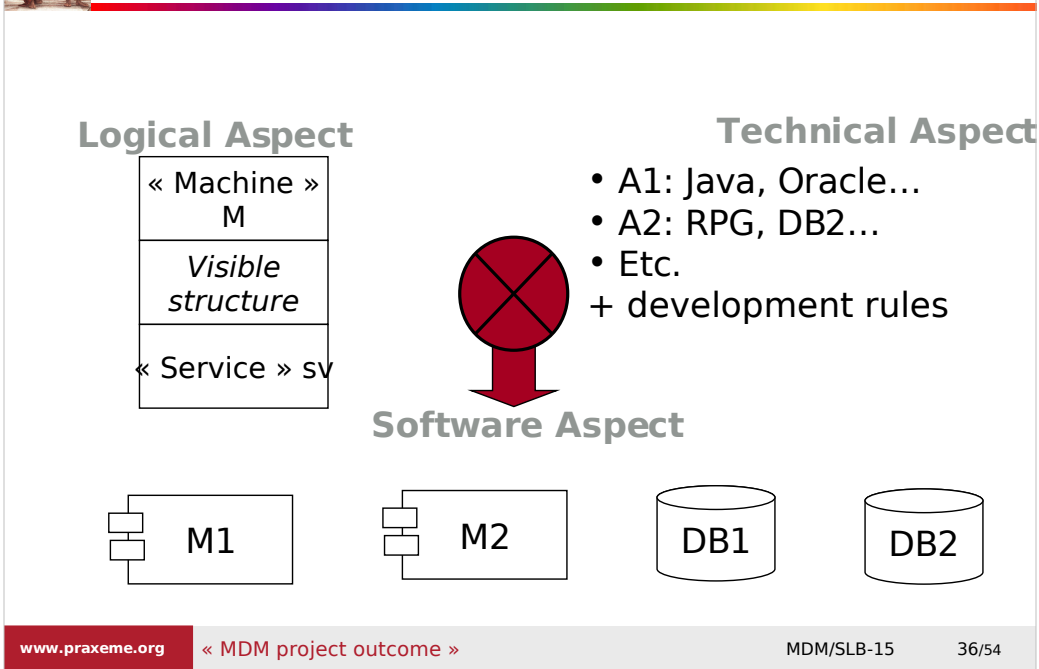


Semantic Model Transformation Origin of Logical Services





Semantic Model Transformation Chain (2)



Règles de développement :

Une ML → une classe Java ou un programme AS/400.

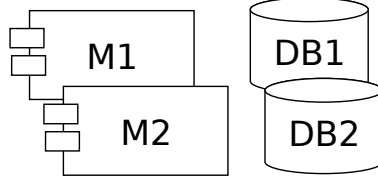
Un sv → une méthode... + référencement dans le Broker



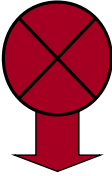
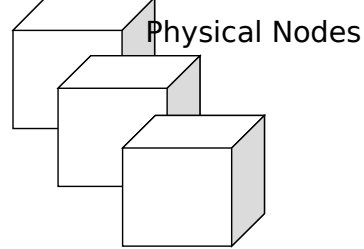
Semantic Model Transformation

Transformation Chain (2)

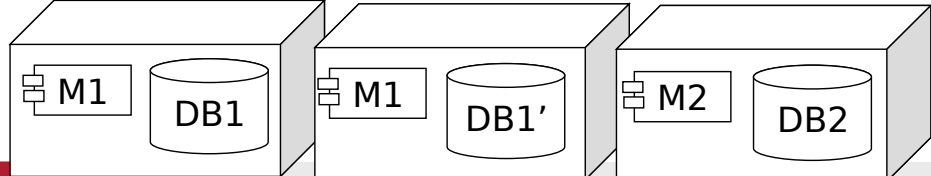
Software Aspect



Hardware Aspect



Physical Aspect

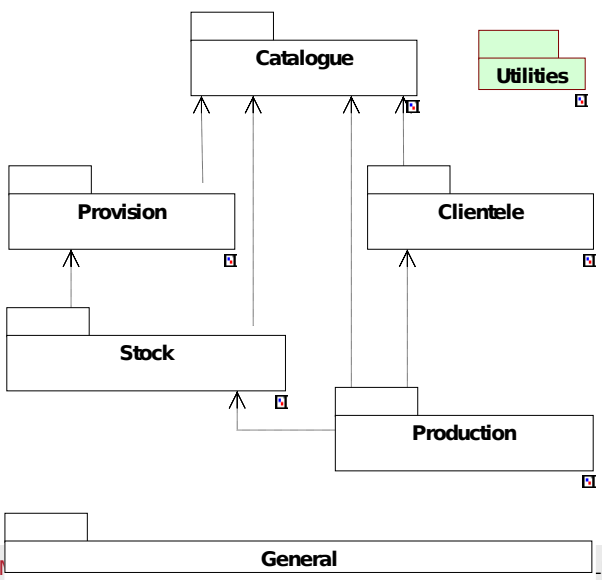




Semantic Model Transformation

Logical Architecture Graph

■ Cliquez pour ajouter un plan



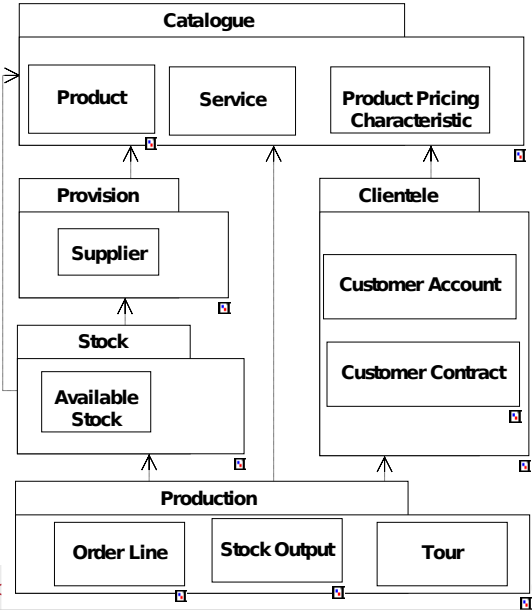


Logical Architecture Graph

Semantic Model Transformati

(detailed)

▪ Cliquez pour ajouter un plan





Subjects for the Logical Model

- **Logical / Technical discussions**
 - Examples
 - Transaction management
 - Error management
 - Event management
- **Regarding the modelling of the Logical Model**
 - Object identification
 - Keeping history traces
 - Data reconciling



Semantic Model Transformation Subject Identification (1/3)

▪ The problem

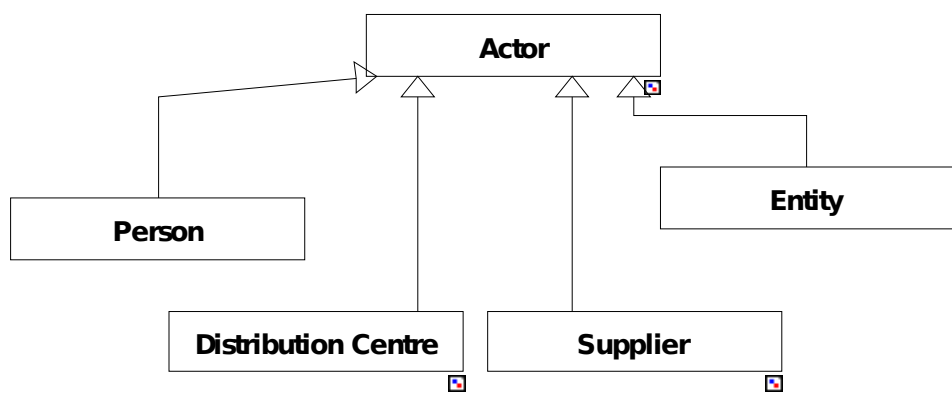
- A unique system for all countries
- A unique system for the classes inside the same inheritance tree



Semantic Model Transformation (2/3)

Subject Identification (2/3)

▪ **An example**





Semantic Model Transformation Object Identification (B, B)

▪ **The solution**

- A universal system for object identification
 - Valid for all countries
- This is a question for the logical design
 - Unless it receives a technical answer



Semantic Model Transformations Keeping history traces (1/2)

- **On many concepts**
 - Creation date, update date and so on
 - User responsible for the action

Order Acquisition

```
+acquisition start time:P-Time  
+acquisition end time:P-Time  
+employee id:P-Numeric  
+employee name:P-String  
+employee short name:P-String  
+employee title:P-String  
+terminal id:P-String
```



Semantic Model Transformation Keeping history traces (2/2)

- **It is not relevant in the semantic model**
 - Risk of overloading the core model
- **Another issue for the logical design**



Semantic Model Transformation Data Redundancy (1/2)

▪ The problem

- Existing data in different structures
- Must converge to the future and unified system
 - Either feeding the future system
 - Or providing it with information as requested



Semantic Model Transformation Data Reconciliation (2/2)


- **Two options**
 - Conversion
 - Transforming data, migrating to the new database
 - One-shot, on a project-time basis
 - Reconciliation
 - Thanks to EAI possibilities
 - On a run-time basis
- **This is part of the logical/technical discussion**
 - It can be decided by parts
 - It must be set in the urbanisation roadmap



4

Contents of Part 4

- **Milestones**
- **Collected documents**
- **Continuing on post-MDM project**
 - Consolidation
 - Possible works to be done



| | |
|--------------------------------|---|
| Availability of Modelling Tool | Beginning of January |
| Document collection | Deadline: 7/01/05 Collection stopped: 7/02/05 Difficulties respecting perimeter |
| 1st Workshop | 14/01/05 |
| Data dictionary delivery | 2/02/05 |
| v2 Semantic Model delivery | 11/02/05 |
| 2nd Workshop | 16-17/02/05 |
| v3 Semantic Model delivery | 22/02/05 |
| Logical Data Model delivery | 7/03/05 |

www.praxeme.org « MDM project outcome » MDM/SLB-15 49/54

WS1: making aware of the approach and information needs



Collected documents

- Collected documentation towards Semantic Model

Fields

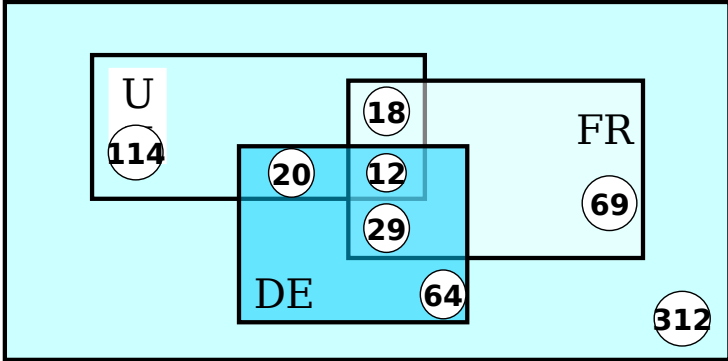
| | | |
|--------------|------------|-----|
| Out of Scope | 940 | 45% |
| Issue | 178 | 9% |
| No Link | 498 | 24% |
| Link | <u>473</u> | 23% |
| | 2089 | |


- The accuracy of the collected documents has been poor
 - See '[metric.xls](#)' file for details



Convergence Metrics

- **The Semantic Model attributes**

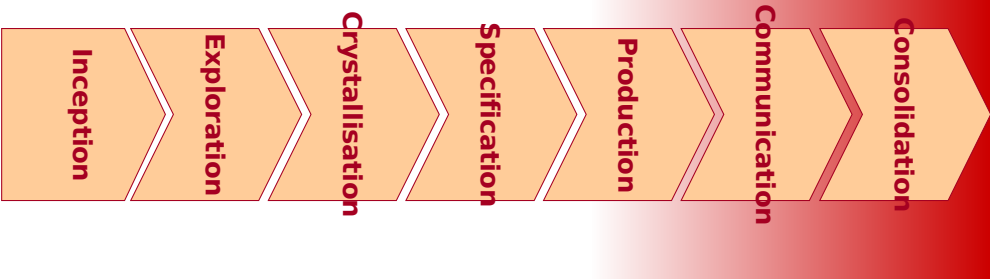




Continuing on post-MDM project

The Plan

- **Explaining and maintaining the semantic model**
 - The MDM project ends with the delivery of the logical data model
 - It can carry on with the « Consolidation phase »
 - Change process on SEM and LDM



www.praxeme.org « MDM project outcome » MDM/SLB-15 52/54

Acting as the custodians of the semantic model (with JPP)
proposal



Possible Works

- **On the semantic model**
 - Complete the State machines
 - States & Transitions
 - Identification of all the Operations
 - Before deriving the services
 - Complete & improve upon comments
 - More explanations
- **From the use case description**
 - Bridge the Use Case View and the SEM
 - By means of connections to the relevant modelling elements
 - By means of new diagrams
 - Structure the Use Case View
 - By eliminating redundancy



Conclusion

- **Outcome of the MDM project**
 - Benefits of Semantic Modelling
 - Convergence
 - Simplification
 - A start for the business repository
 - Benefits for the IS
 - A source for deriving the services in SOA
- **Planning, means and budget**
 - Kept on track
 - Absolute need for a UML tool
- **Difficulties**
 - Tools
 - Cultural change
 - State of the input documentation
 - Scope