

Comment réaliser l'alignement de modèles métier et applicatif ?

PEPIN Jonathan - décembre 2013
Doctorant Sodifrance/AeLos

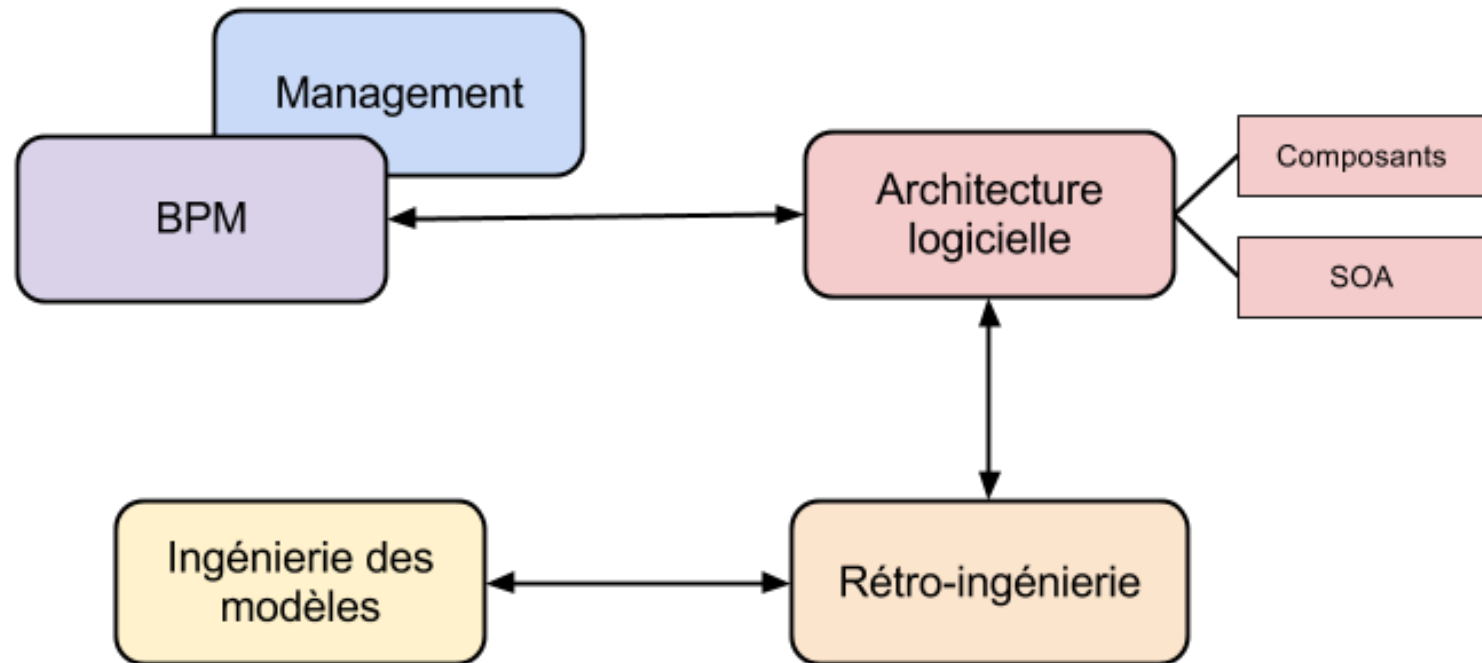
Problématique

- Mettre en correspondance le monde de la gestion du SI et l'informatique
- Aligner la stratégie de l'entreprise avec le parc applicatif
- Gérer une masse d'information importante
- Automatiser/assister l'alignement
- S'abstraire de toute technologie ou représentation

Contexte

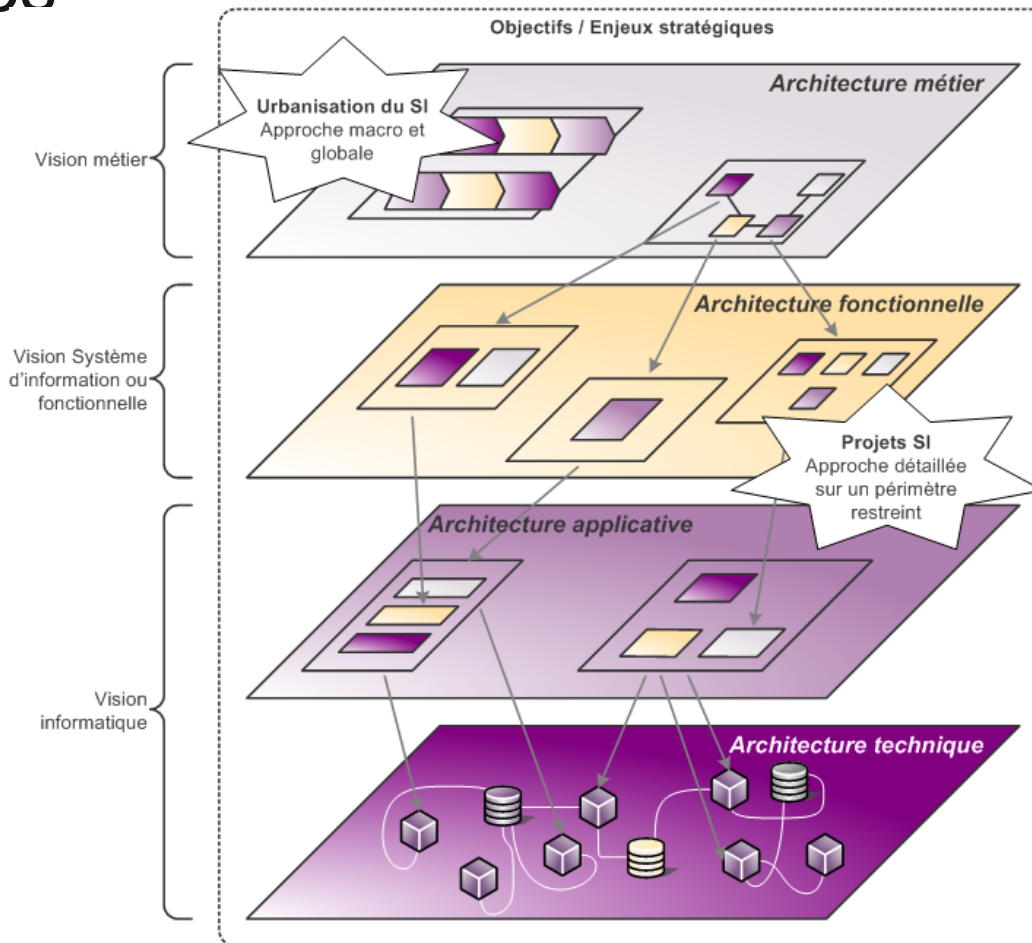
- Entreprise = **Systeme d'information**
- Définition
« Les organismes publics et les entreprises privées possèdent un ensemble organisé de ressources matérielles, logicielles et humaines appelé système d'information. » De Courcy R.

Domaines



Approche existante

- Décomposition du SI en points de vue / couches



Approche existante

1. **Vue stratégique**

Facteurs internes et externes de l'entreprise qui va orienter son activité

2. **Vue métier**

Représentation des processus et activités : BPM.

3. **Vue fonctionnelle**

Classification en blocs des fonctionnalités offertes par le métier, les dépendances sont structurelles : zone, ilot, quartier.

4. **Vue applicative**

Composants logiciels et services.

5. **Vue infrastructure/technique**

Représentation des dépendances technologiques : bases de données, serveurs, conteneurs, réseau, etc.

Approche existante

- Urbanisation / Architecture d'entreprise
 - vision francophone
 - urbanisation (métaphore de la cité)*
 - Longépé / club urba (fin 1990)
 - vision anglosaxonne
 - Enterprise Architecture*
 - Zachmann (1980's IBM), puis TOGAF (1990's)
 - SOA / SoEA
 - apparu en 2001

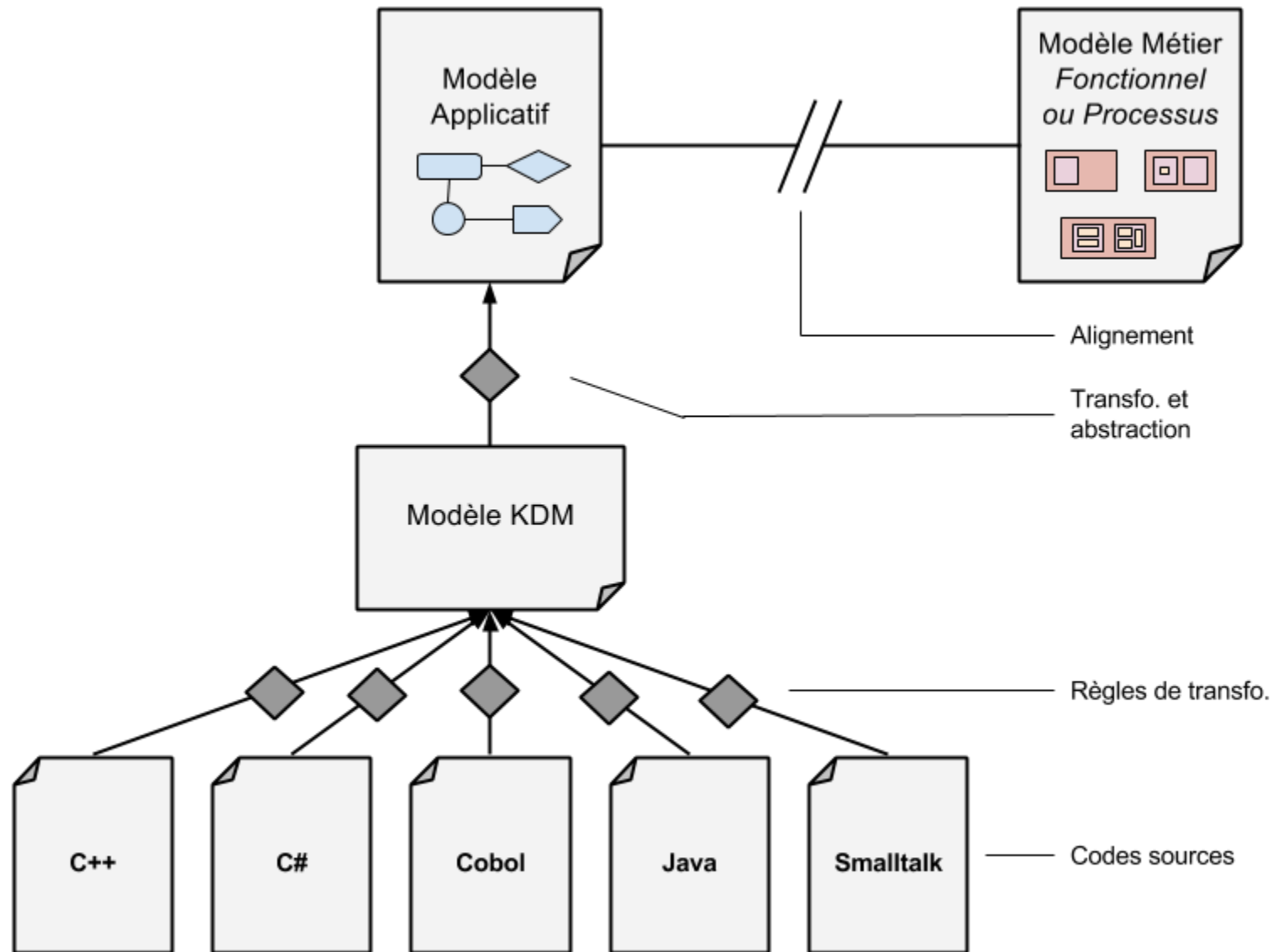
Approche existante

- But de la démarche d'urbanisation :
Faire évoluer le SI, aligner
- Etapes clefs :
 - cartographie de l'existant et cible
 - définition des règles d'urbanisme
 - plan d'urbanisation directeur
 - pilotage, arbitrage, accompagnement
- L'idée : 1 couche = 1 modèle = 1 méta-modèle
spécifiant les concepts

Notre démarche

- Les idées :
 - 1 couche = 1 modèle = 1 méta-modèle spécifiant les concepts
 - Alignement par mise en relation de la couche métier et applicative (formellement)
 - Déduire la couche applicative du code source (problématique d'automatiser au maximum)
 - Mesurer l'alignement et obtenir des indicateurs

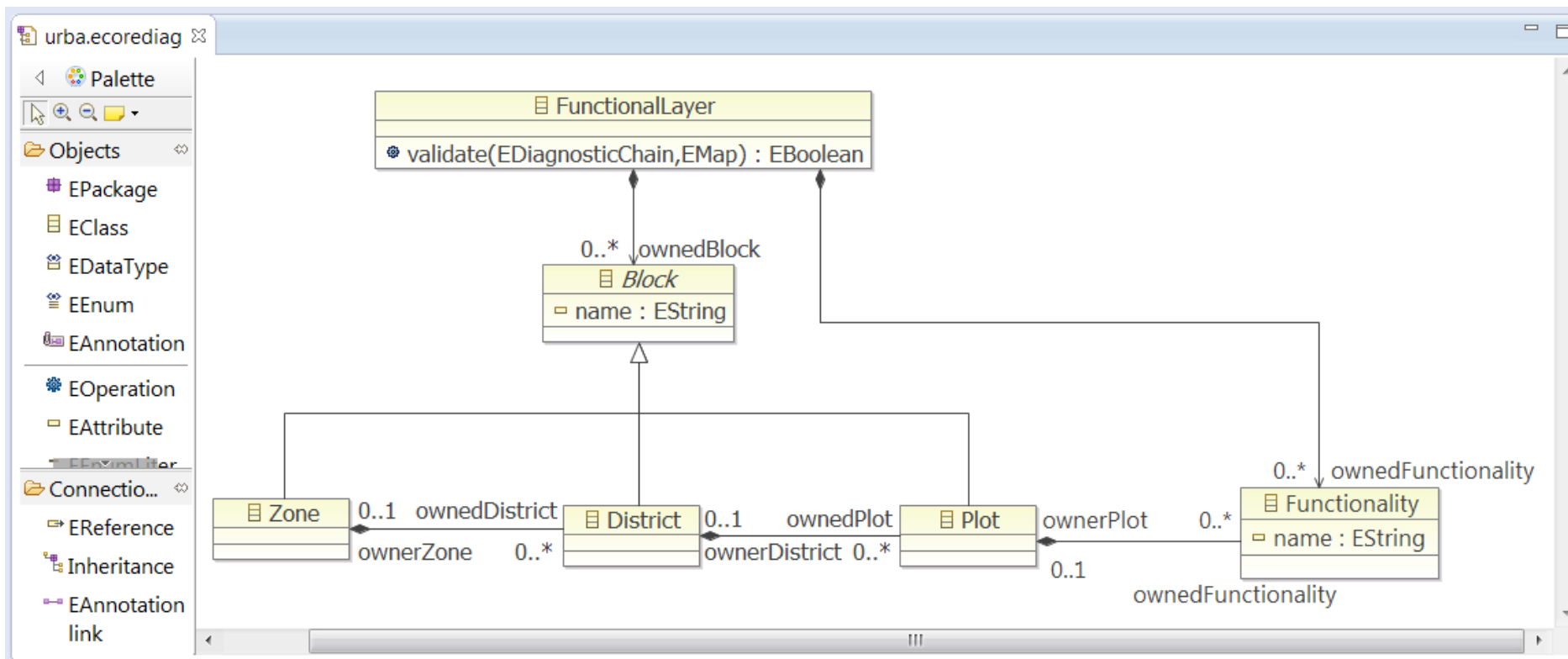
Notre démarche



- Utilisation d'Eclipse EMF
 - modélisation en MOF
 - mécanisme de validation OCL
 - génération d'une implémentation en Java
 - persistance en XMI
(possible une BD relationnelle via un ORM)
 - création d'éditeur rapide sous forme de plugin Eclipse
 - création de diagramme via GMF

MDE

- Exemple de méta-modèle



MDE

- Exemple de règle de validation OCL

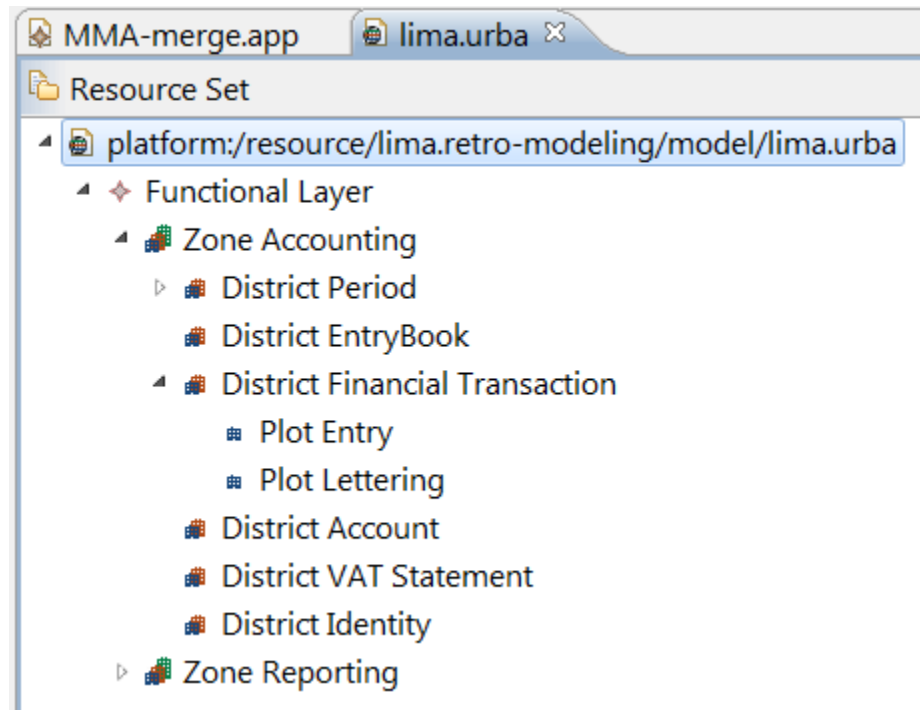
- ▷ triggeredBy : Function
- ▷ triggers : Function
- ▲ DataObject -> Element
 - ▷ GenModel
 - ▷ Ecore
 - ▲ OCL
 - dataObjectExistInServiceAndFunction -> self.accessedByFunction.realizes.accesses -> exists(s | s = self)
 - Element
 - ▷ accessedByFunction : Function
 - ▷ accessedByService : Service
- ▷ Element

Problems Javadoc Declaration Console Synchronize Error Log Call Hierarchy Vue CPD Debug Properties

Property	Value
Key	dataObjectExistInServiceAndFunction
Value	self.accessedByFunction.realizes.accesses -> exists(s s = self)

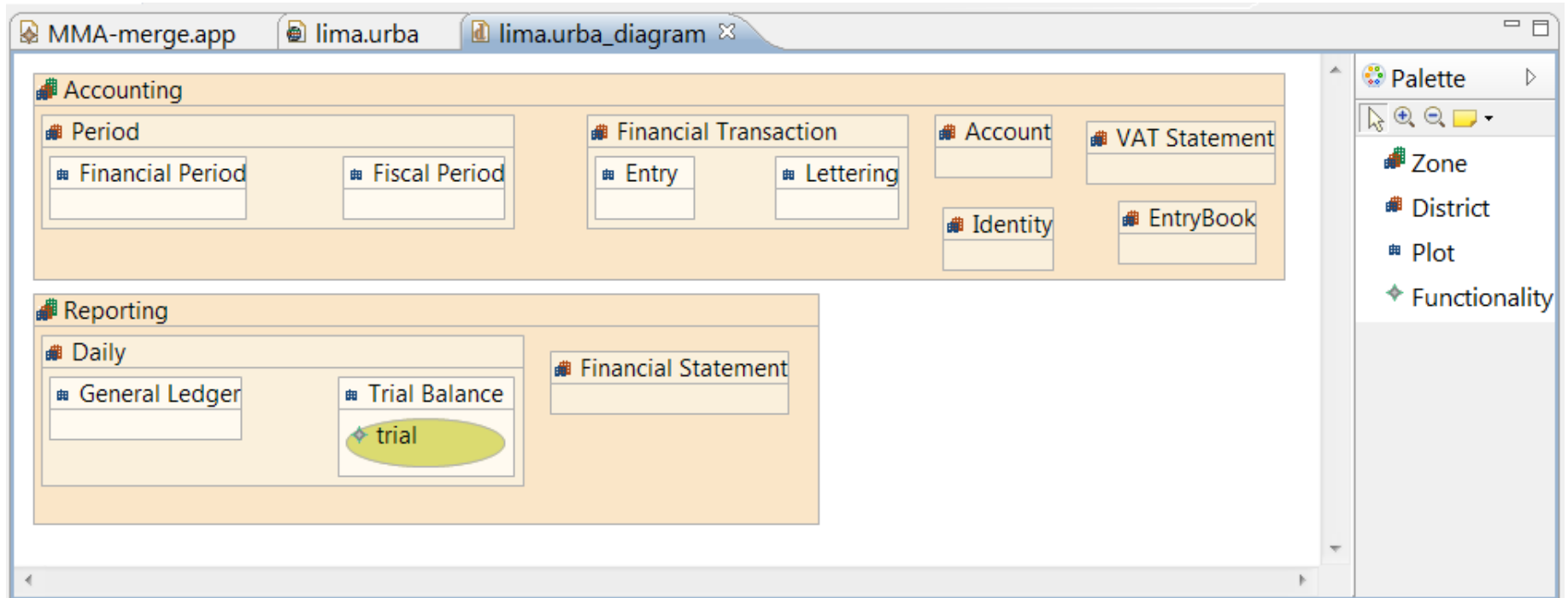
MDE

- Exemple d'éditeur

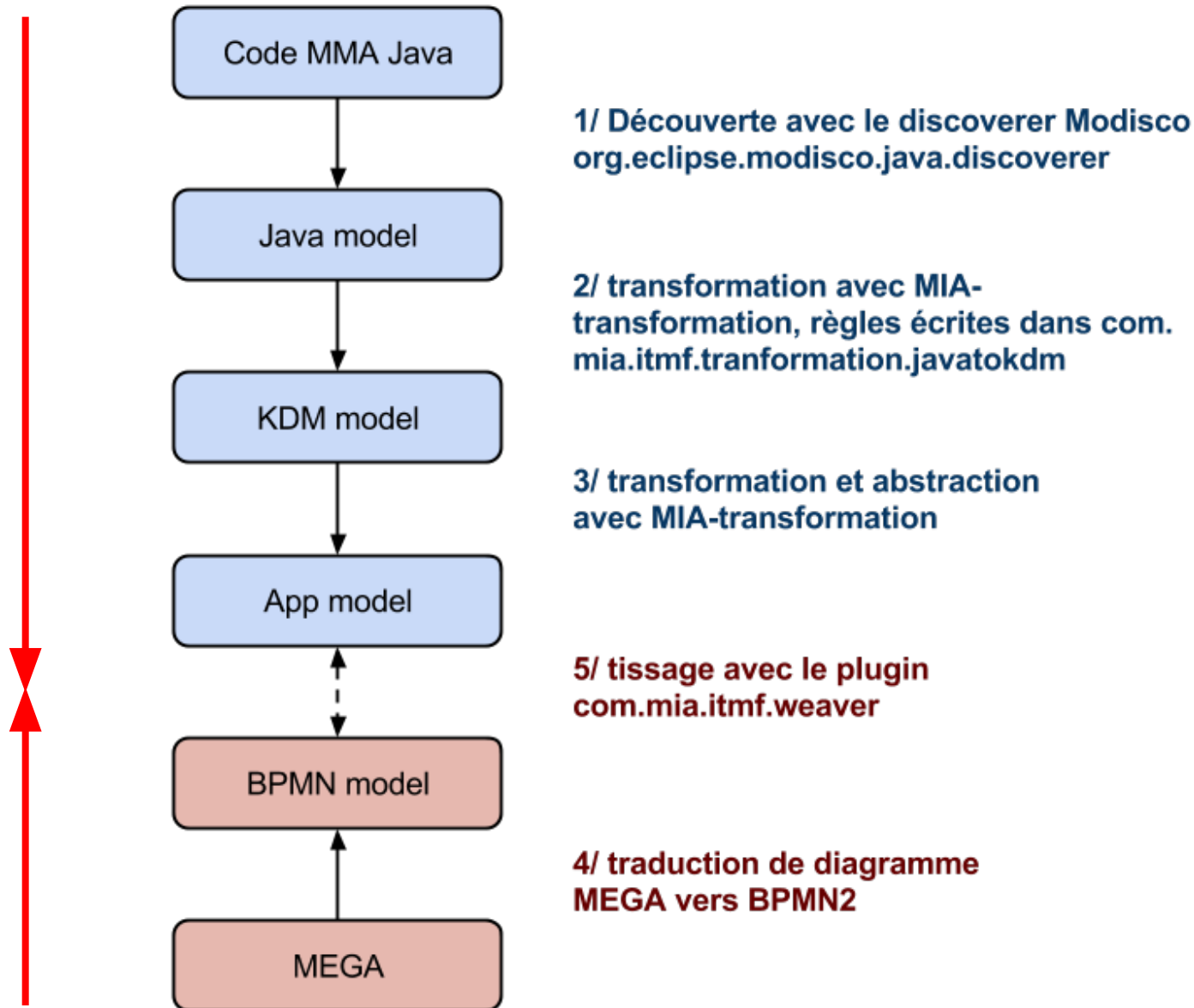


MDE

- Exemple d'éditeur de diagrammes



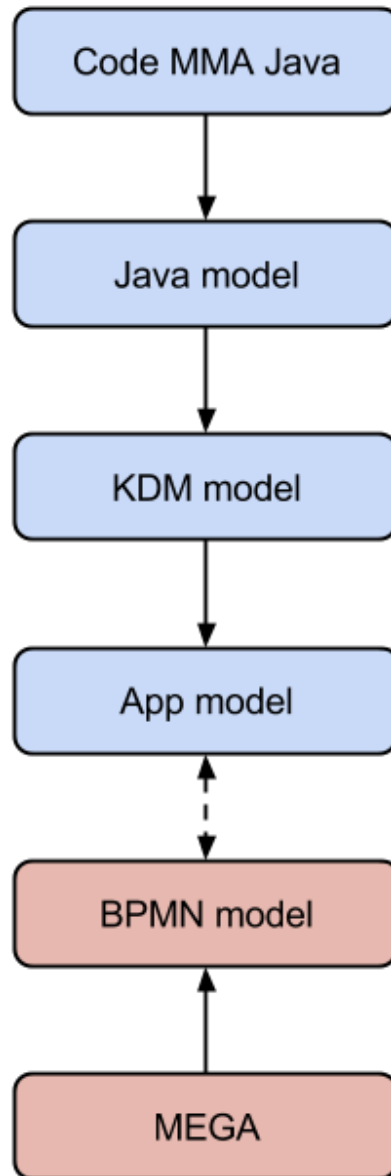
Expérimentation - cas MMA



Expérimentation - cas MMA

- Application DEFFIS
 - Gestion des contrats d'assurance
- Code applicatif très important
 - 33 400 classes Java
 - 3 400 000 lignes de code
 - 792 JSP
- Une cartographie existe et écrite en MEGA

Expérimentation - cas MMA



1/ Découverte avec le discoverer Modisco
org.eclipse.modisco.java.discoverer

2/ transformation avec MIA-
transformation, règles écrites dans com.
mia.itmf.tranformation.javatokdm

3/ transformation et abstraction
avec MIA-transformation

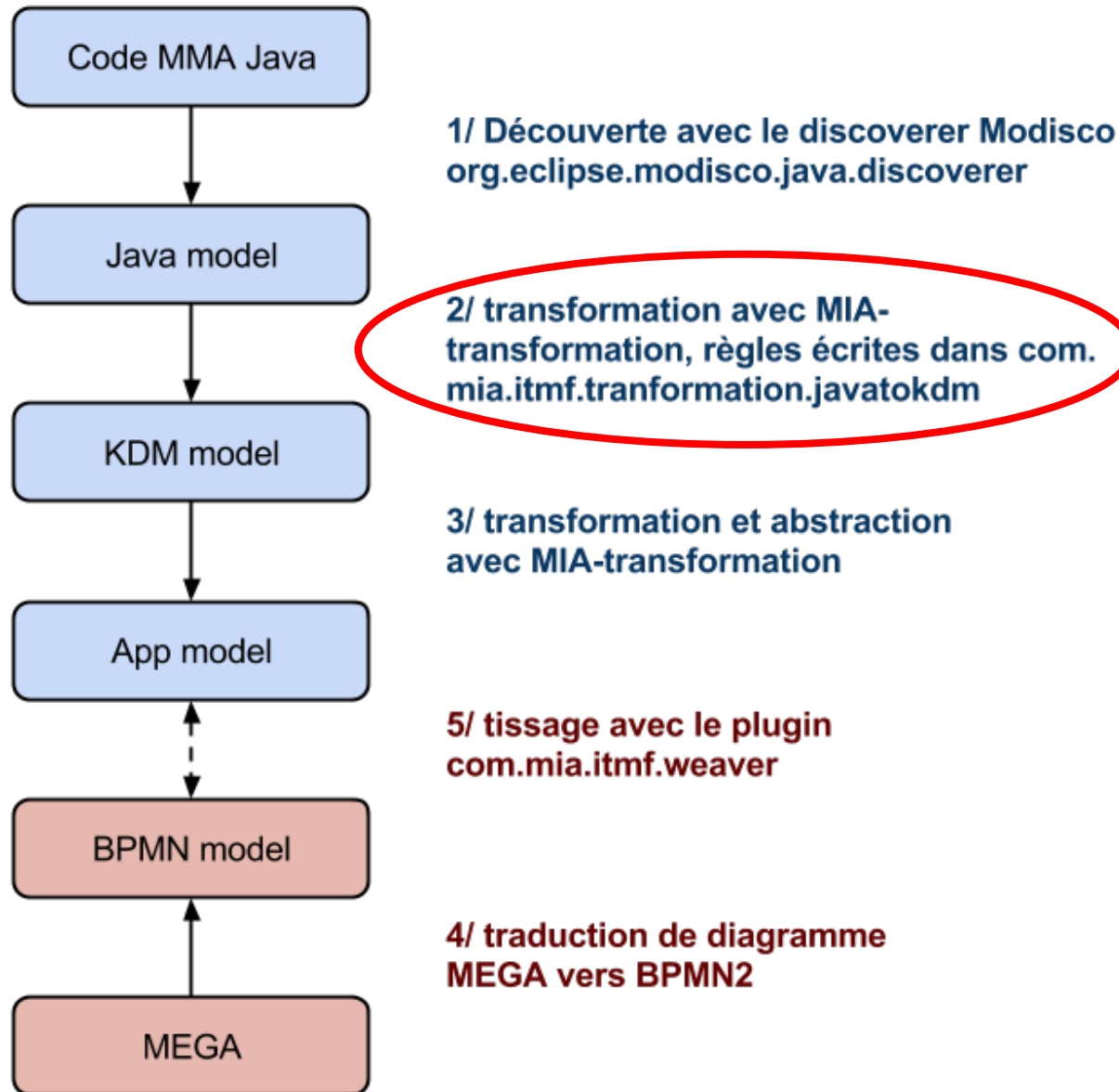
5/ tissage avec le plugin
com.mia.itmf.weaver

4/ traduction de diagramme
MEGA vers BPMN2

Etape 1 - Analyse du code source

- Découverte du code source et stockage dans un modèle PSM sous la forme d'un XMI
- Utilisation de l'outil open source Modisco
 - Initié par AtlanMod : Inria, Mines Nantes et LINA
 - Supporté par Mia-Software (R&D Sodifrance)
- et de passer de JDT vers Java Model
 - JDT : socle de développement Java d'Eclipse (compilation, formattage, debug, ...)

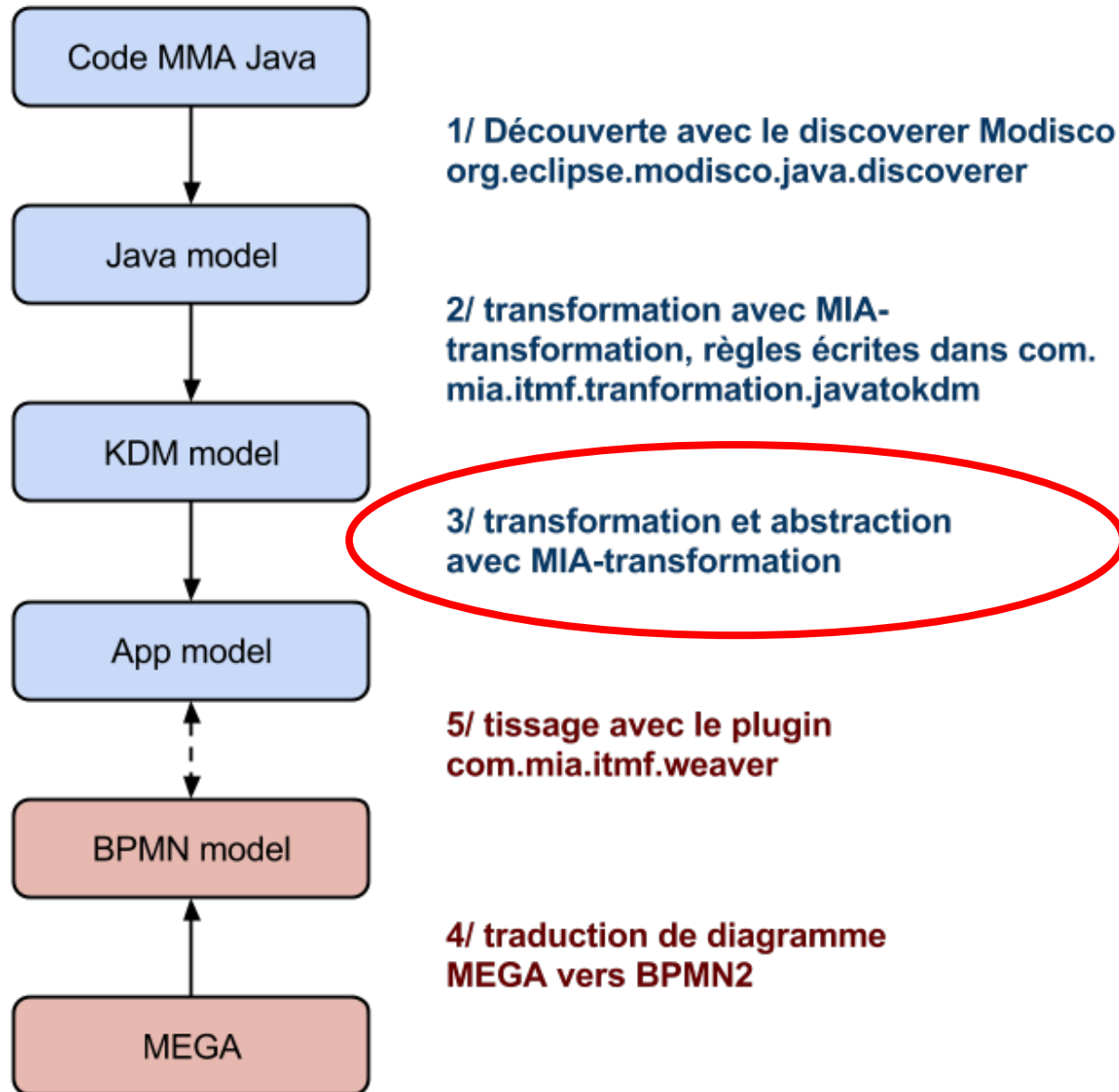
Expérimentation - cas MMA



Etape 2 - Transformation

- Utilisation du modèle pivot PIM :
KDM (standard OMG)
- Modisco propose un discoverer Java vers KDM avec une transformation écrite en ATL
 - ne passe l'échelle, ne fonctionne que sur des petits modèles, pas de connaissance pour le support
- Réécriture d'une transformation avec l'outil Mia-Transformation
 - Transformation générique, utilisable pour n'importe quelle application Java

Expérimentation - cas MMA

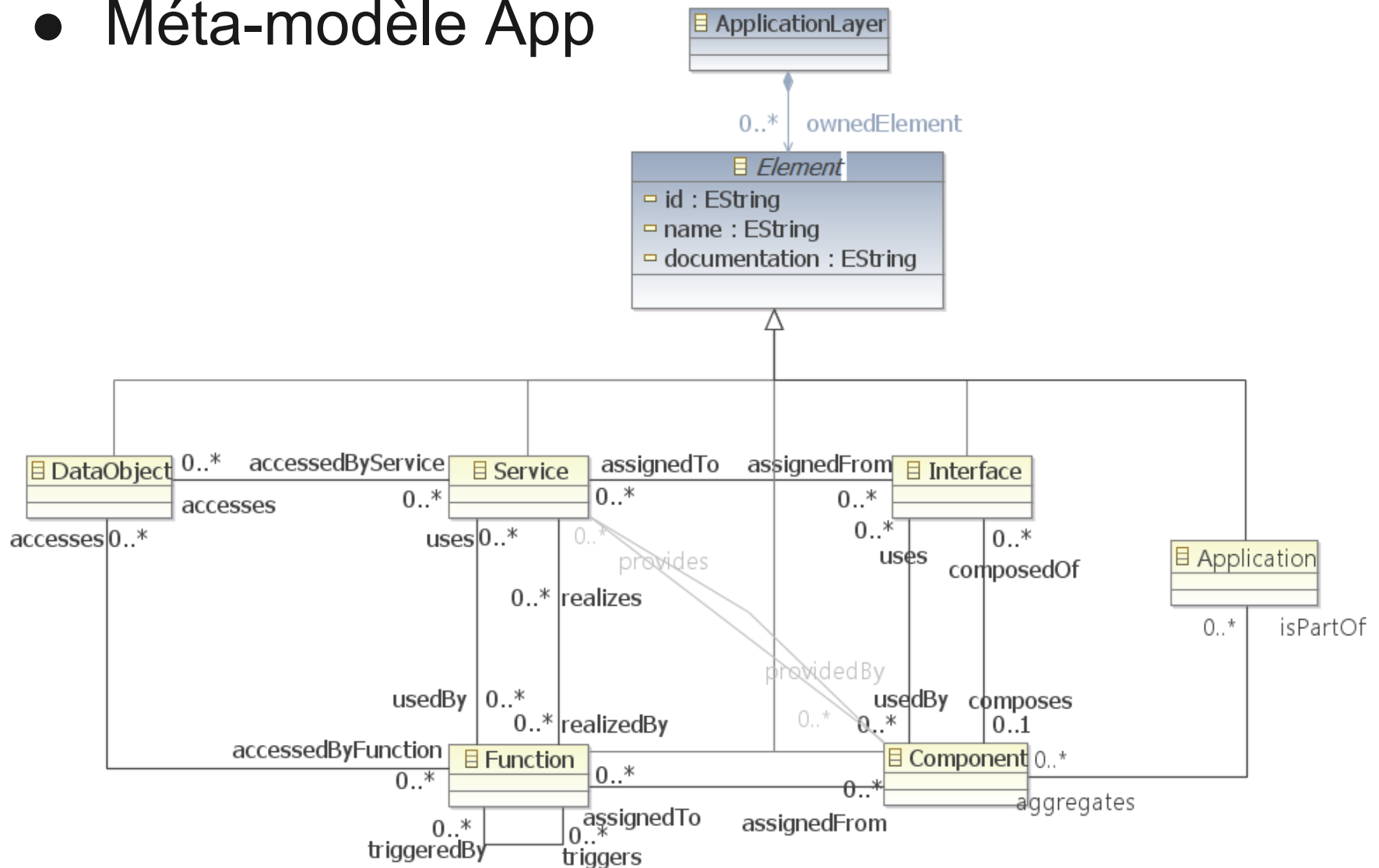


Etape 3 - Abstraction

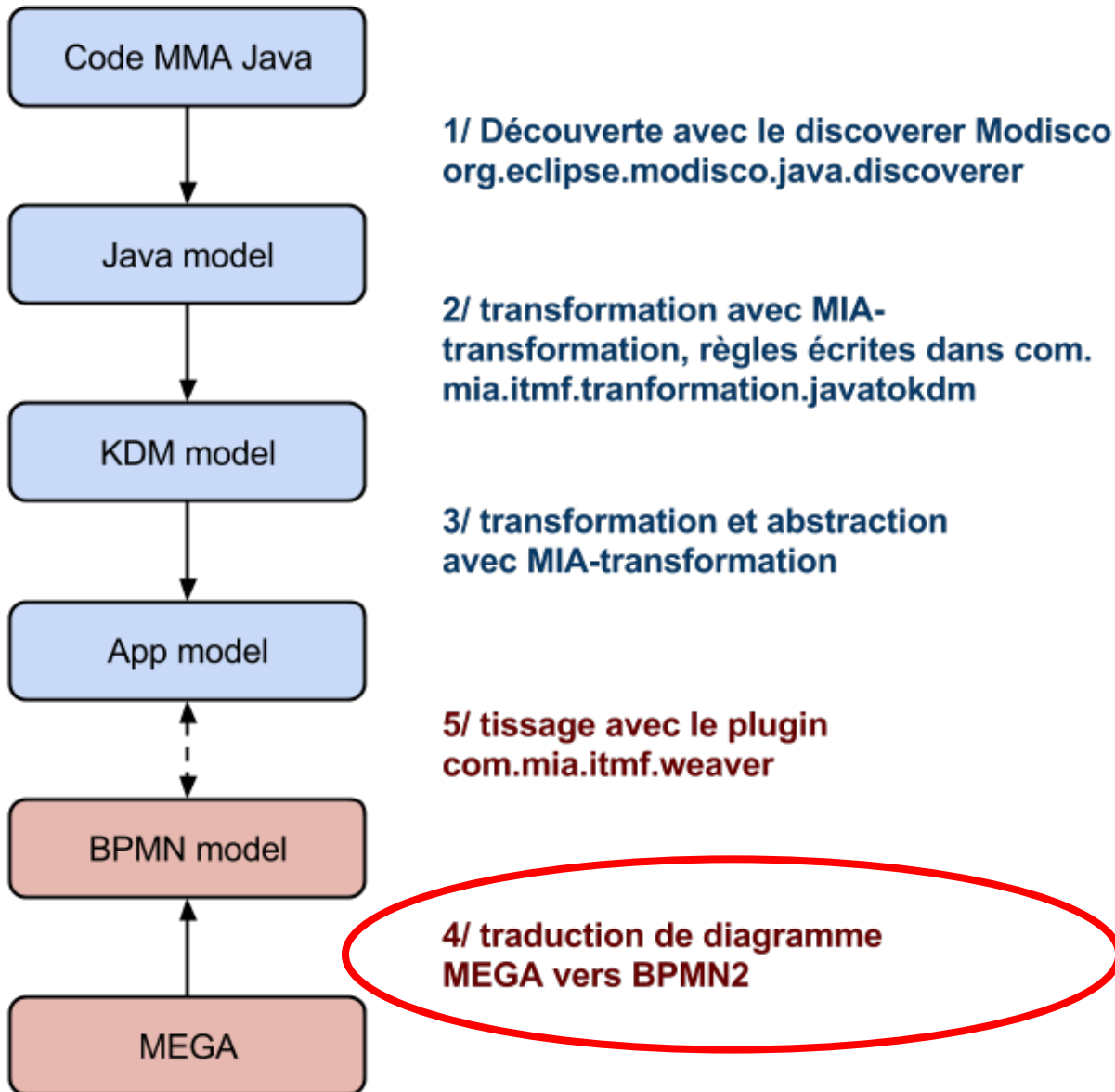
- Transformation et abstraction de KDM vers un modèle applicatif “maison” appelé App
- Écriture d'une transformation avec l'outil Mia-Transformation
 - Transformation générique au niveau de la traduction des concepts de KDM vers App, mais transformation spécifique sur la reconnaissance des concepts : Composant, Interface, DataObject.

Etape 3 - Abstraction

- Méta-modèle App



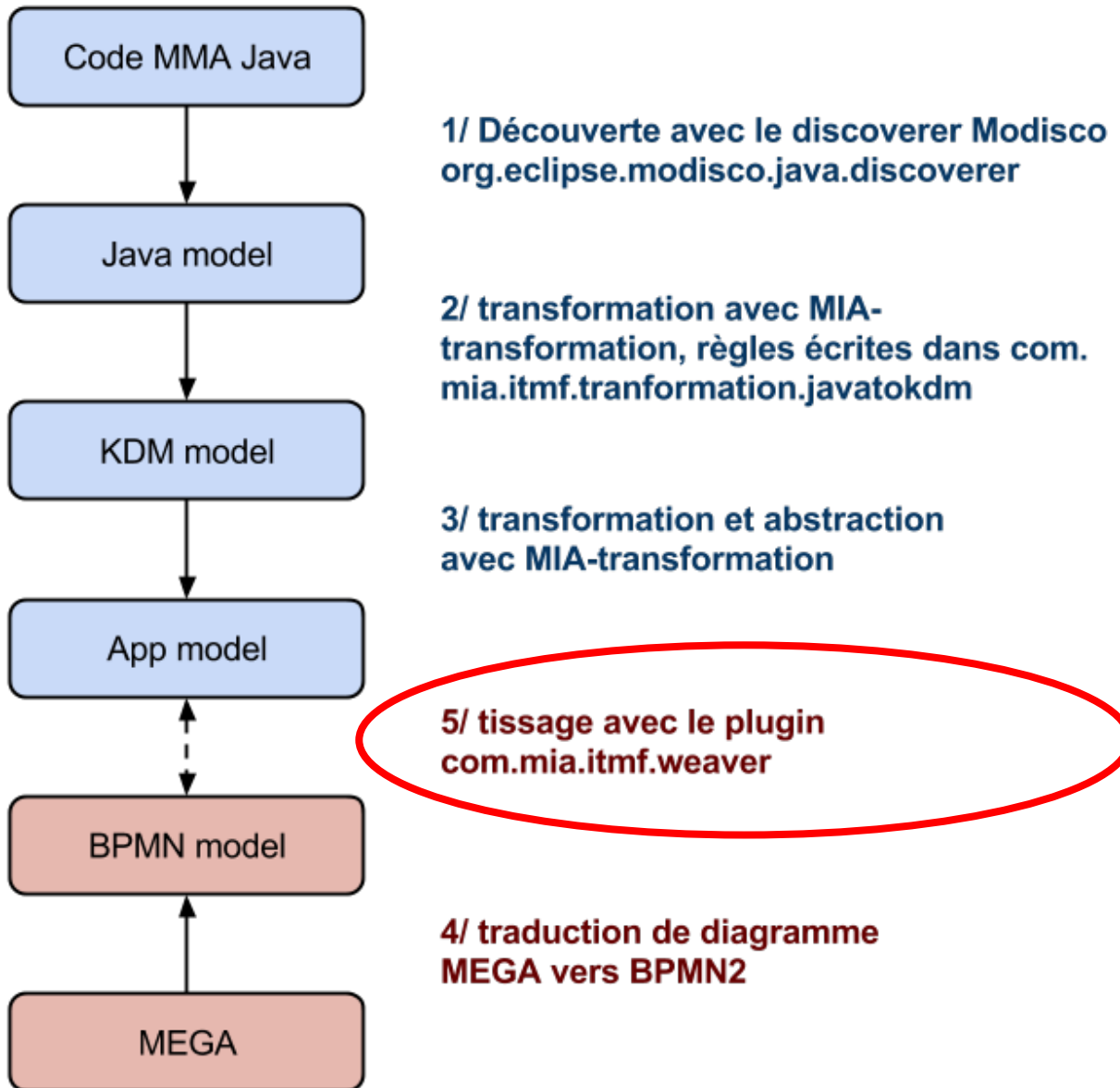
Expérimentation - cas MMA



Etape 4 - Traduction

- Traduction de modèles métiers vers BPMN
- Exploitation de modèles métier d'origine d'un référentiel MEGA exporté en HTML
- Le format XMG source n'est pas disponible, transformation théorique possible, pour le moment test en manuel

Expérimentation - cas MMA

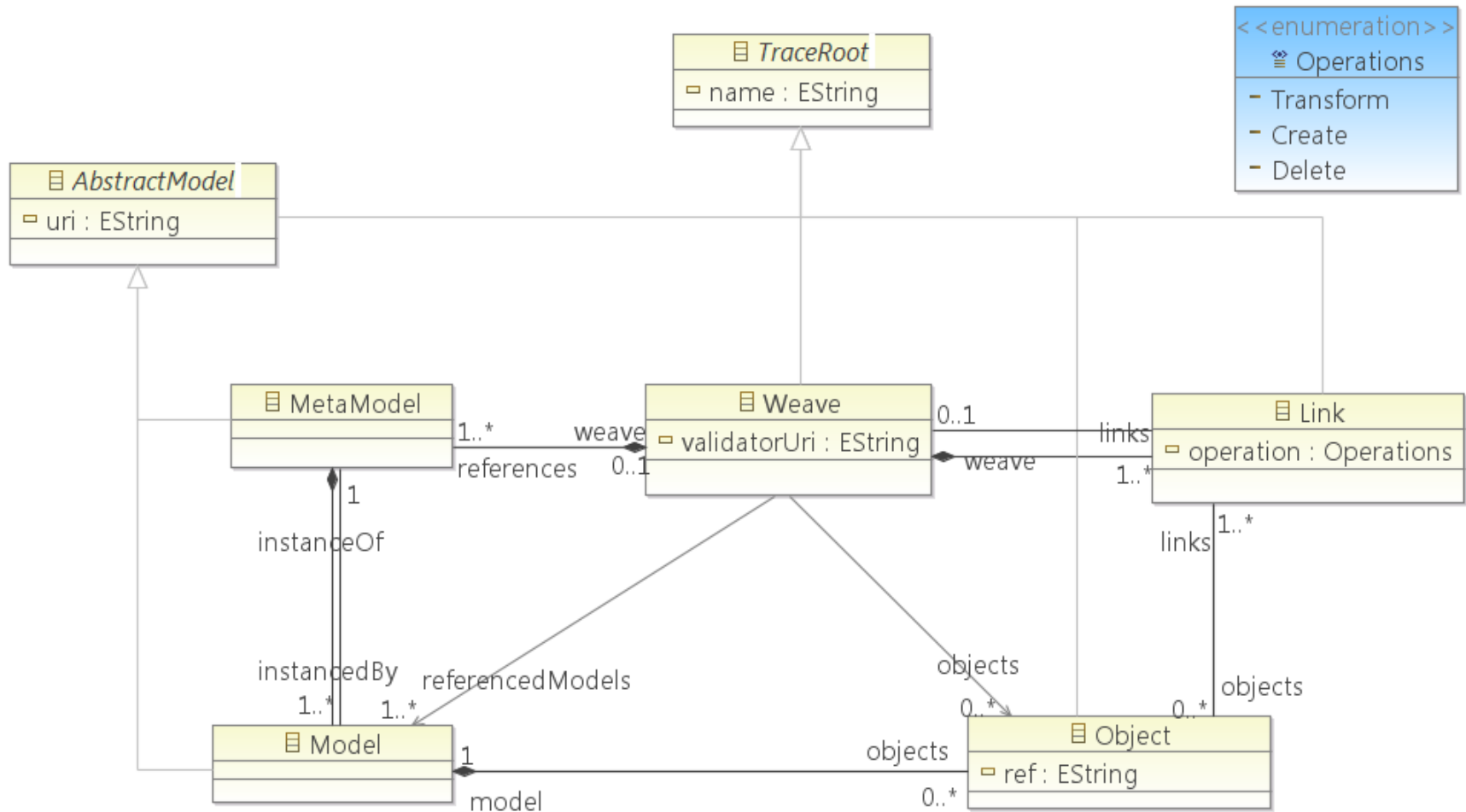


Tissage

- Recherche d'une méthode de tissage non intrusif
- Analyse des techniques existantes :
 - AMW (ATLAS Model Weaver)
 - ATLAS Group, INRIA & LINA
 - Virtual EMF
 - AtlanMod, Mia-Software
- Proposition d'un plugin maison plus souple avec restriction optionnelle par tissage au niveau méta-model.

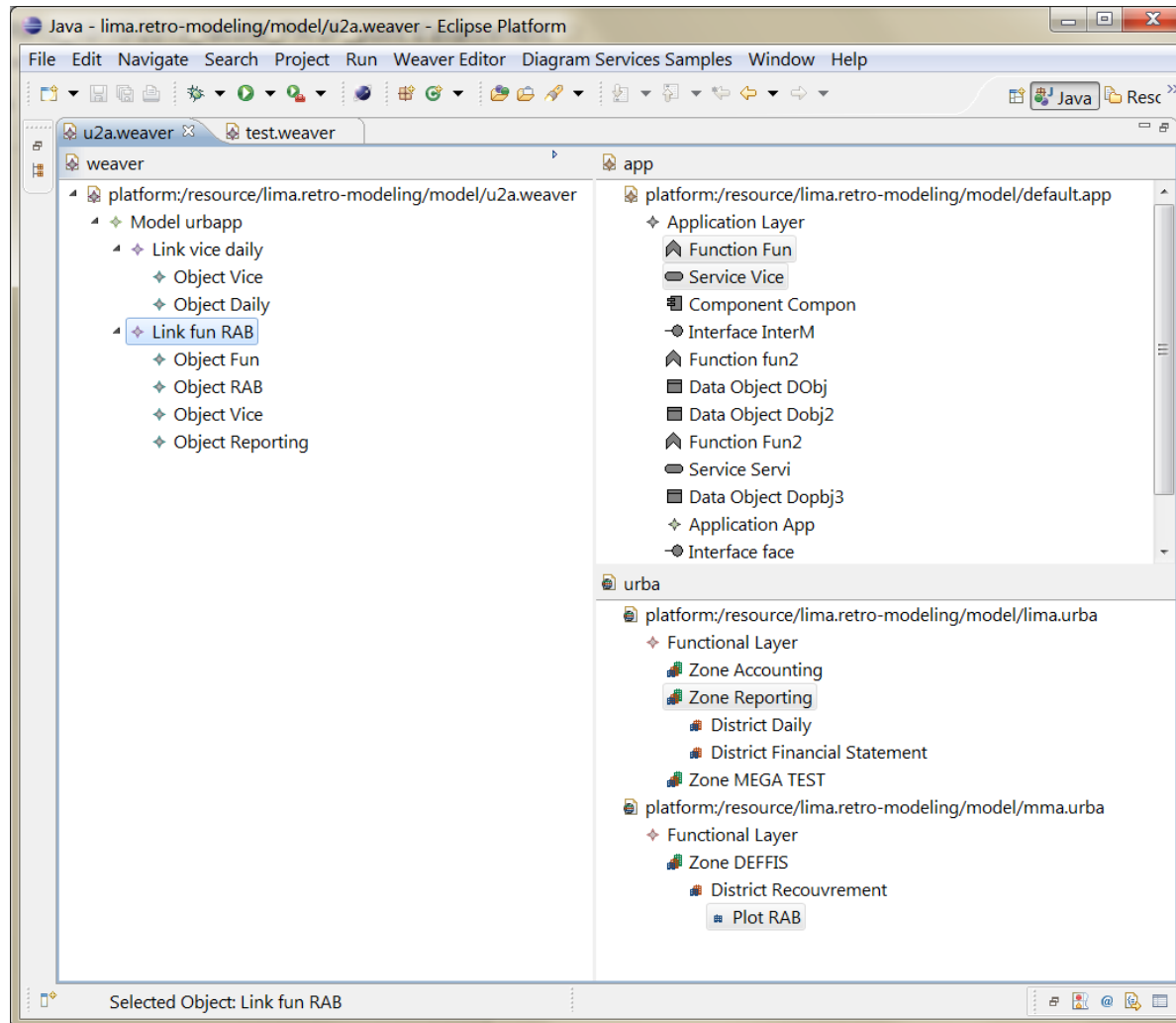
Tissage

- Le méta-modèle



Tissage

- Éditeur de modèles de tissage

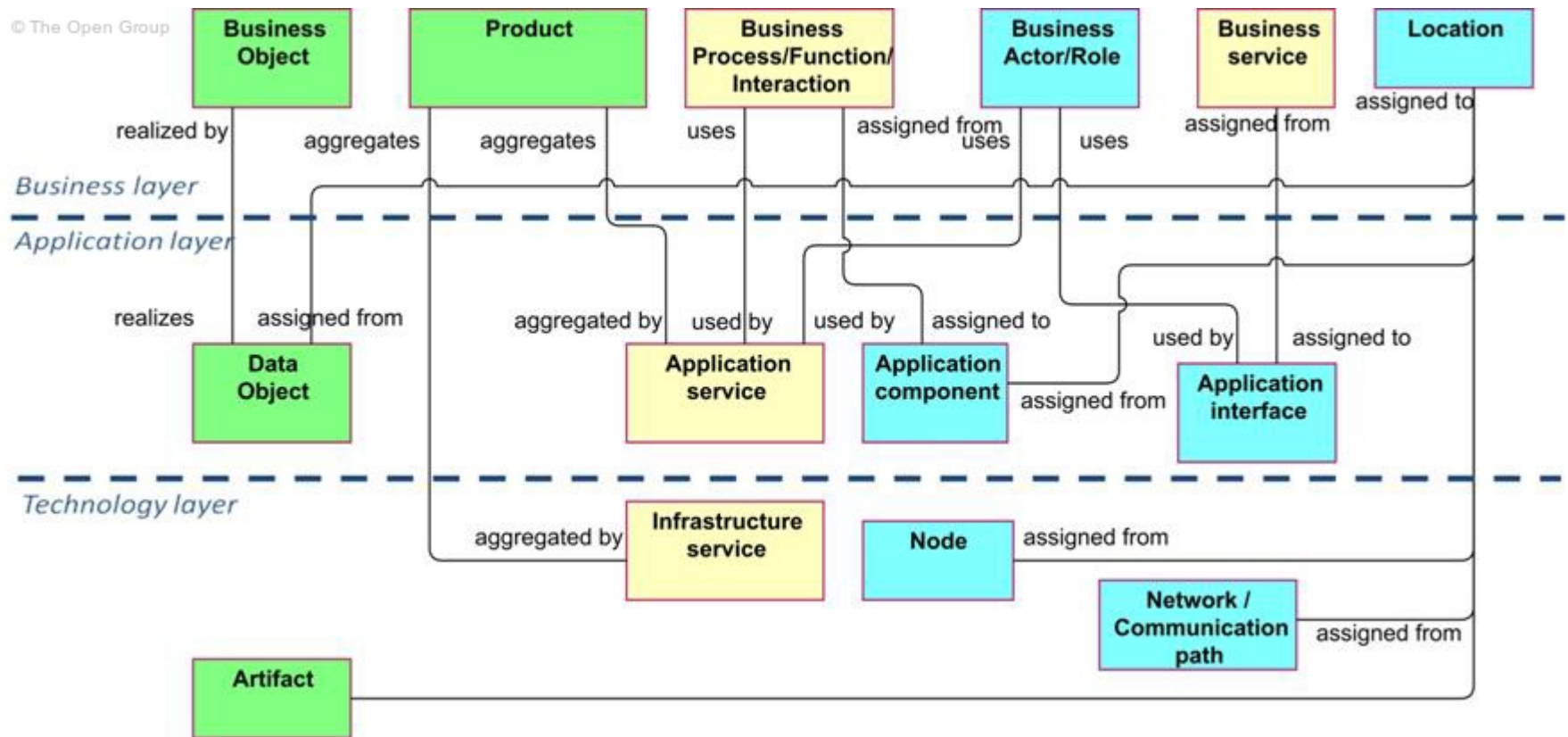


Liens entre méta-modèles

- Etat de l'art
 - Cadre Commun d'Urbanisation du Système d'Information de l'Etat, 10/2012
 - Direction interministérielle des systèmes d'information et de communication de l'Etat
 - ArchiMate 2.1, 12/2013
 - Open Group / TOGAF

Liens entre méta-modèles

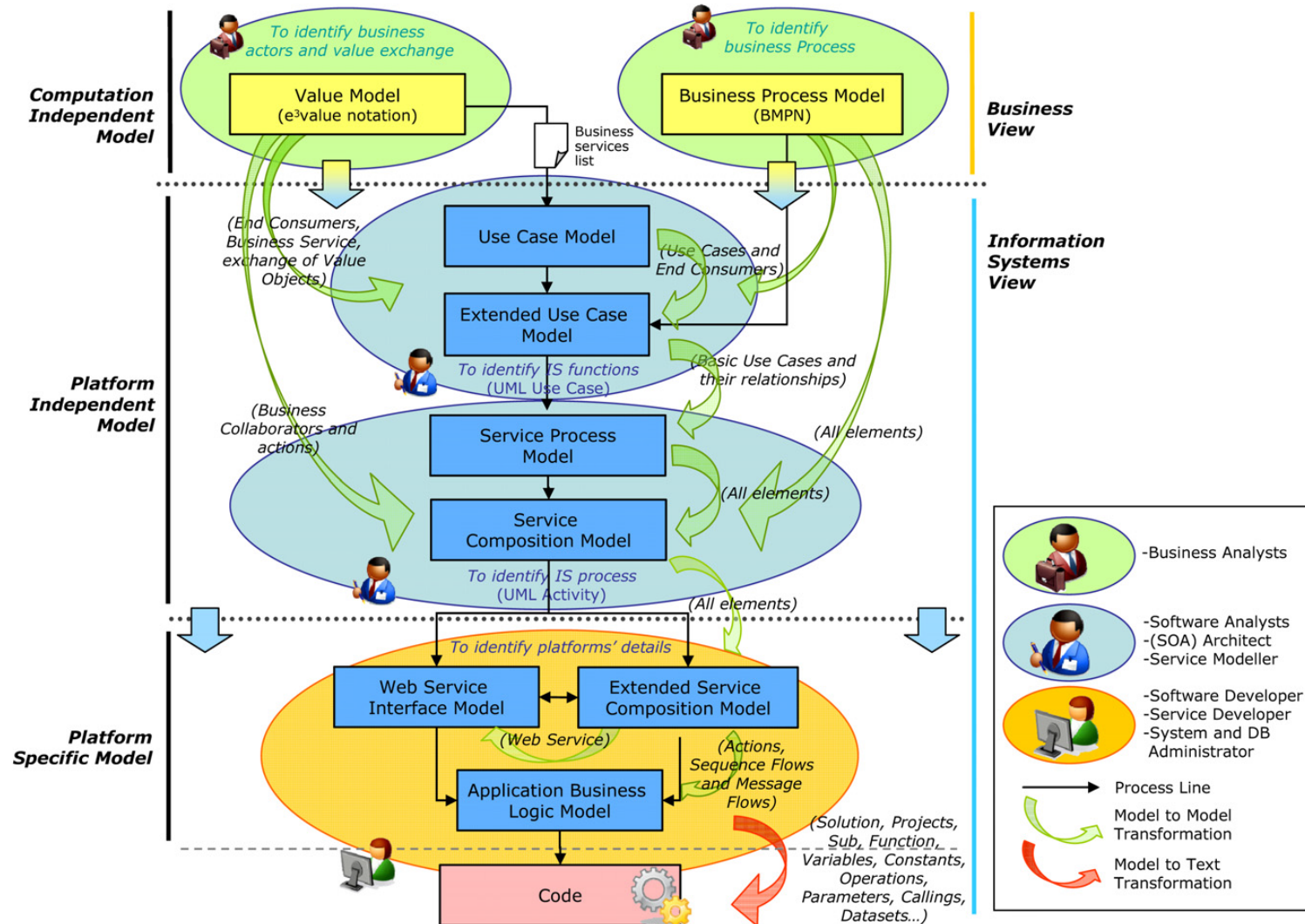
- Extrait méta-modèle ArchiMate



Comparaison avec un article

- *Applying CIM-to-PIM model transformations for the service-oriented development of information systems*
Esperanza Marcos, Elsevier, 2010
 - Utilisation de MDE pour générer une application SOA à partir de modèles processus
 - S'appuie sur EMF, ATL, GMF

Comparaison avec un article



Comparaison avec un article

- **Points négatifs**
 - BPMN vers SOA, source et cible sont homogènes
 - Vocabulaire technique orienté Model-Driven touchant un public de type architecte logiciel
- **Points intéressants**
 - concepts mis en relation
 - détail sur les règles de transformation
 - technologies utilisés présentes des similarités

Analyse et mesure

- Une fois les modèles tissés que faire ?
 - -> prochaine étape de la thèse
- Quantifier et qualifier l'alignement
 - critères ?
 - métriques ?
- Type d'analyse
 - vérification / validation
 - couverture