

# Transformations de modèles : d'un modèle abstrait aux modèles EJB et CCM

© Olivier Caron <sup>1</sup>, Xavier Blanc <sup>2,3</sup>

LIFL<sup>1</sup> - Softeam <sup>2</sup>- LIP <sup>6</sup><sup>3</sup>

# Plan

- ✓ La spécification MDA (Model Driven Architecture) sur les transformations de modèles
- ✓ Le projet RNTL ACCORD
- ✓ Les expériences EJB et CCM
- ✓ Un processus méthodologique de transformations de modèles
- ✓ Conclusion

# La spécification MDA

- ✓ Démarche de développement proposée par l'OMG
- ✓ Définition d'une architecture de spécifications structurée en modèles ([Poole 01, Miller 01]) :

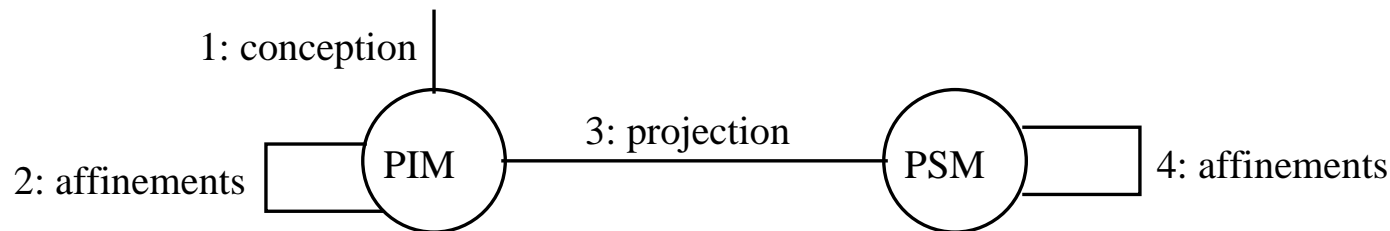


FIG. 1: L'architecture de modèles MDA

- ✓ De nombreuses qualités :
  - ▶ Ciblage vers plusieurs plates-formes technologiques
  - ▶ Standardisation des projections, productivité

# MDA et les transformations de modèles (1/3)

- ✓ La spécification MDA présente différentes approches de transformation de modèles<sup>1</sup> :
  - ▶ L'approche par méta-modèles
  - ▶ L'approche par marquage de modèles
  - ▶ Des approches hybrides
- ✓ La spécification MDA reste très informelle sur ces techniques

---

<sup>1</sup>[http://www.omg.org/mda/mda\\_files/MDA\\_Guide\\_Version1-0.pdf](http://www.omg.org/mda/mda_files/MDA_Guide_Version1-0.pdf)

## MDA et les transformations de modèles (2/3)

✓ L'approche par méta-modèles :

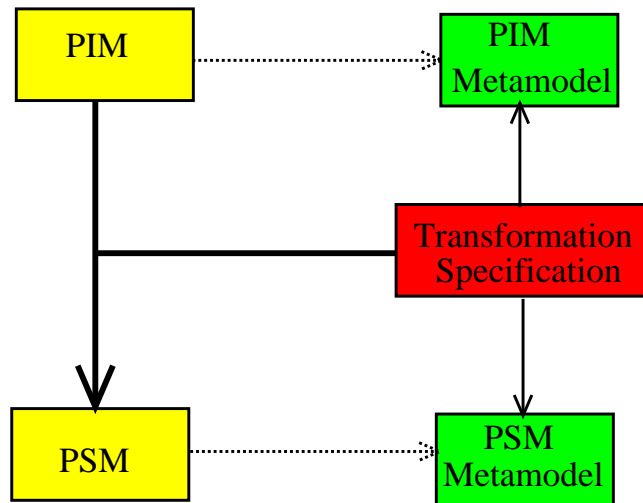


FIG. 2: Transformation MDA

✓ L'approche la plus explorée : ([Lemesle 98, Bézivin 01, Belaunde 02, Peltier 02])

## MDA et les transformations de modèles (3/3)

✓ L'approche par annotations

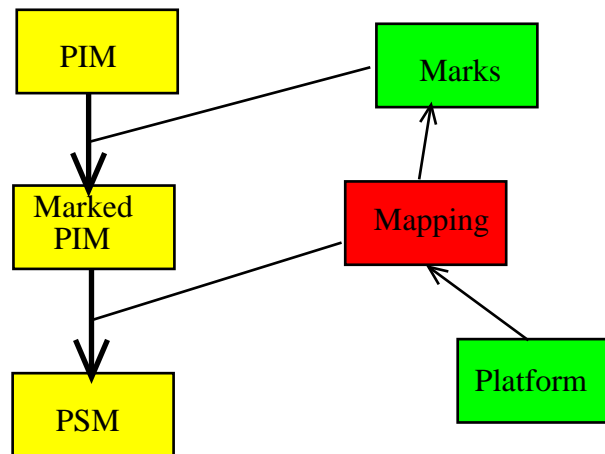


FIG. 3: Transformation MDA

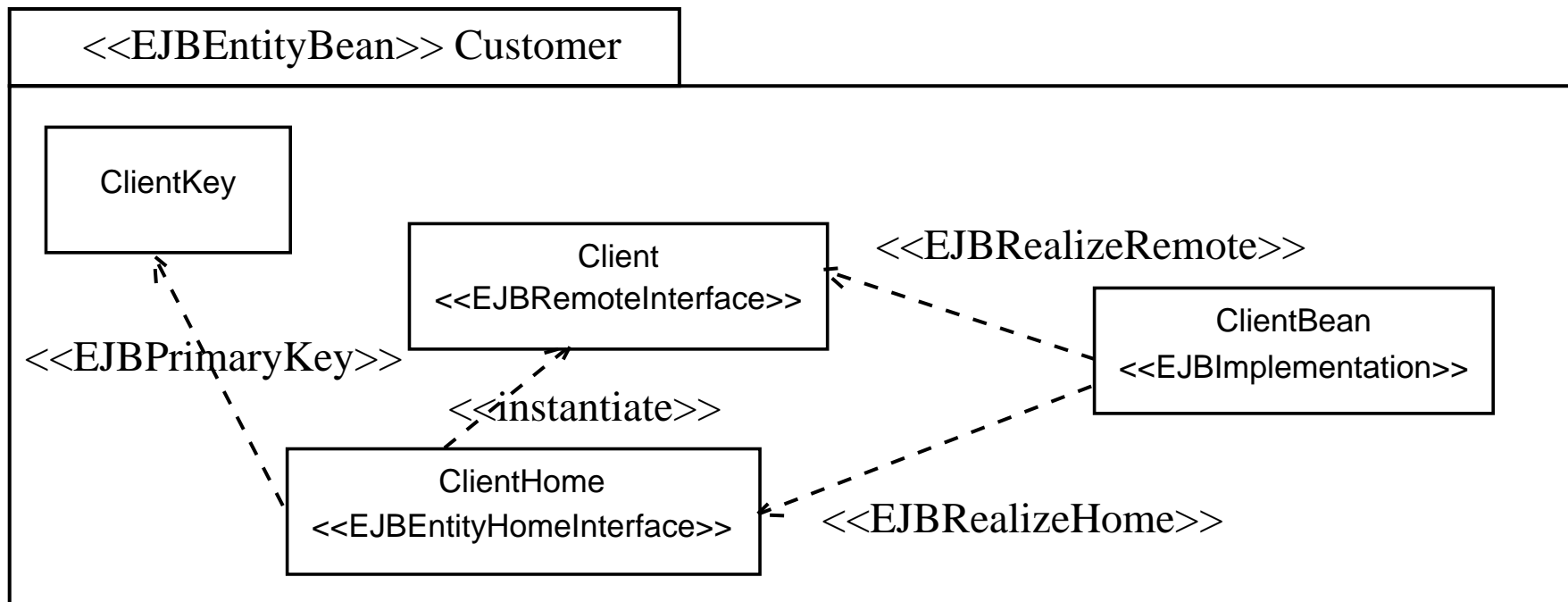
- ✓ Approche très peu explorée, de nombreuses interrogations
- ✓ La technique de profils UML semble particulièrement adaptée

## Qu'est-ce qu'un profil UML ?

- ✓ Mécanisme standard d'extension d'UML
- ✓ Permet de spécialiser un schéma UML selon un domaine particulier
- ✓ **Ensemble** cohérent de stéréotypes, valeur marquées (tagged value) et contraintes
- ✓ Principe général : on n'ajoute pas de méta-classes mais des **annotations** aux méta-classes UML existantes.
- ✓ Des profils standards existent : le profil CORBA (OMG), le profil EJB 1.0 (JCP), . . .

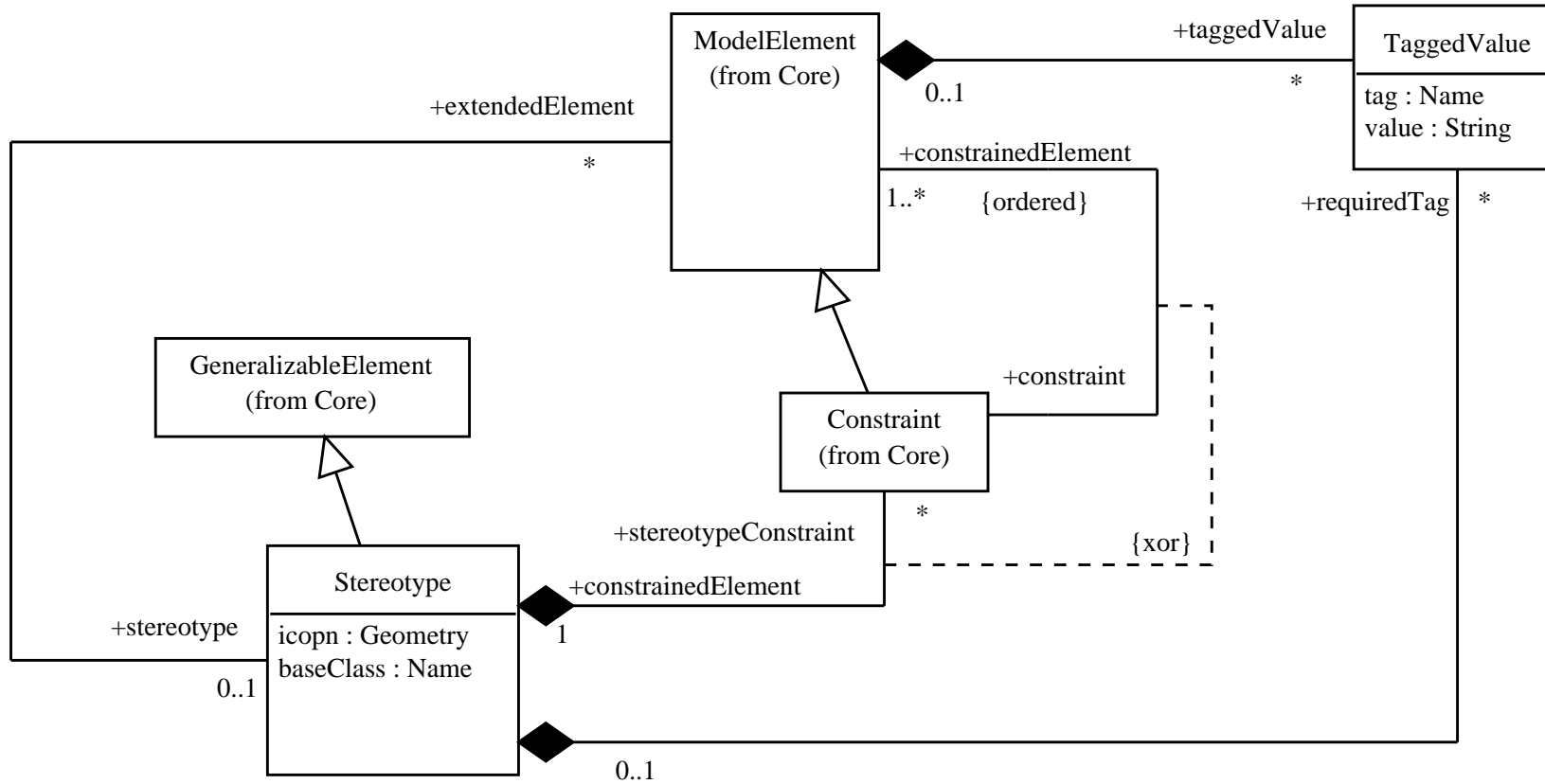
## Exemple de profil UML

✓ Le profil EJB 1.0 :





# Mécanisme d'extension UML



## Le projet RNTL ACCORD

- ✓ ACCORD : Assemblage de Composants par Contrats en environnement Ouvert et Réparti
  1. Définition d'un modèle abstrait de composants avec spécifications de contrats d'assemblage ([JC'2004])
  2. Processus de transformation de modèles : du modèle ACCORD aux modèles EJB et CCM.
  3. Deux cas d'études : France Telecom R&D et EDF R&D
- ✓ Implémentation par le mécanisme de profils UML, prototype UML 2 - Objecteering

# Le modèle abstrait de composants ACCORD

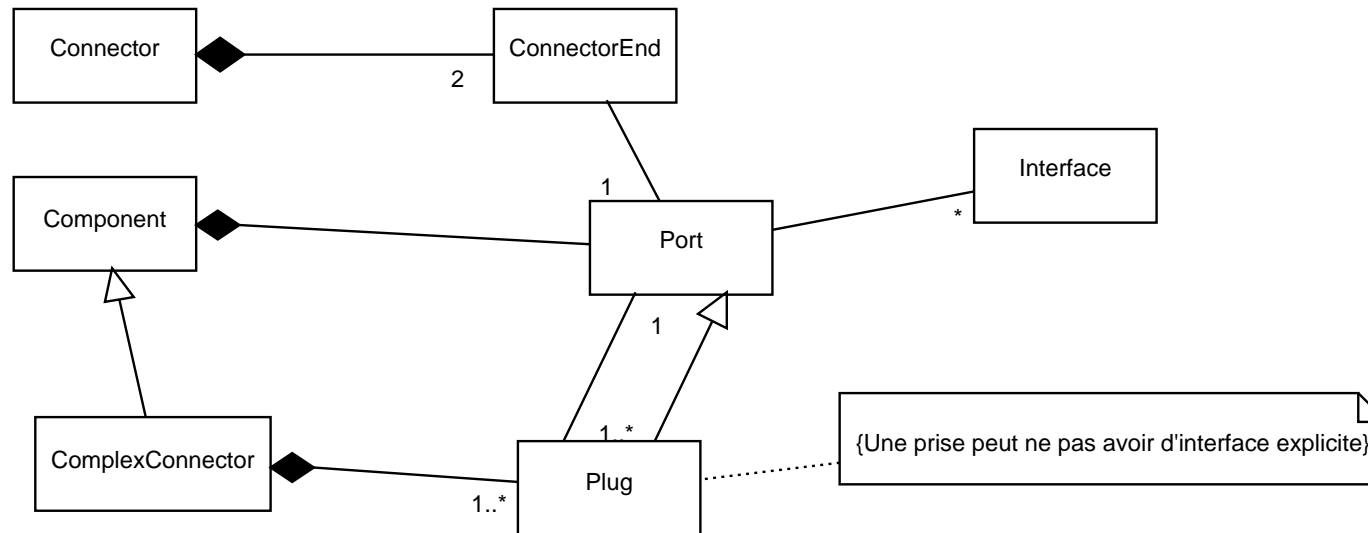
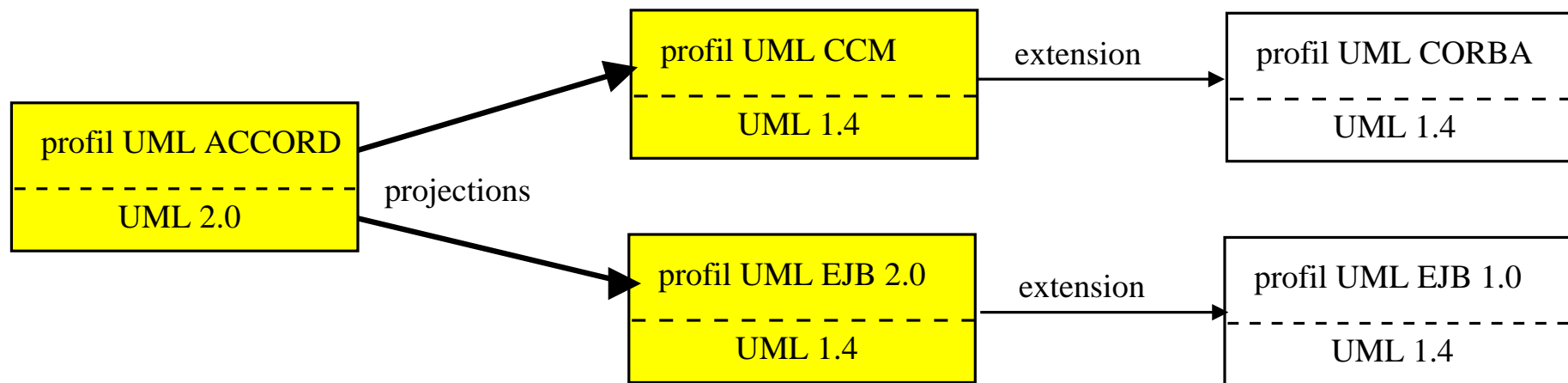


FIG. 4: Le méta-modèle des composants ACCORD

- ✓ Modèle proche d'UML2 (hors contrats), travaux en parallèle
- ✓ Introduction de la notion de connecteur-complexe (JC'2004)

# La plate-forme ACCORD

- ✓ Prototype UML 2 "Profile Builder" d'Objecteering-Softteam
  - ▶ Editeur de profils
  - ▶ Langage de script J : parcours du méta-modèle UML et commandes de création de modèles UML



- ✓ Le profil CCM a été standardisé à l'OMG

# Transformations de modèles

- ✓ Comparatif des concepts ACCORD et PSM (EJB, CCM) :
  1. Un concept ACCORD n'a pas de correspondant
  2. Un concept ACCORD a un unique correspondant direct
  3. Un concept ACCORD a plusieurs correspondants possibles
- ✓ Absence du point 1
- ✓ Prédominance du point 3 :
  - ▶ Pas de correspondance meilleure qu'une autre, dépend de l'application
  - ▶ Nécessité d'introduire de la **flexibilité** dans la projection
- ✓ Mise en place d'un système flexible de **points de choix**

## Projections multiples (1/4)

✓ Exemple modèle abstrait :

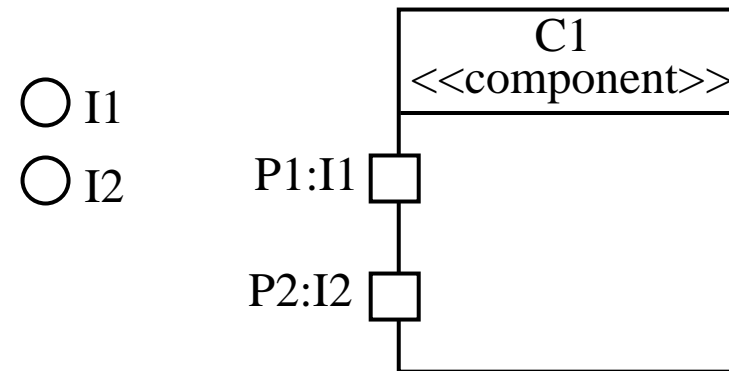


FIG. 5: Un composant ACCORD avec deux ports

## Projections multiples (2/4)

✓ Projections possibles :

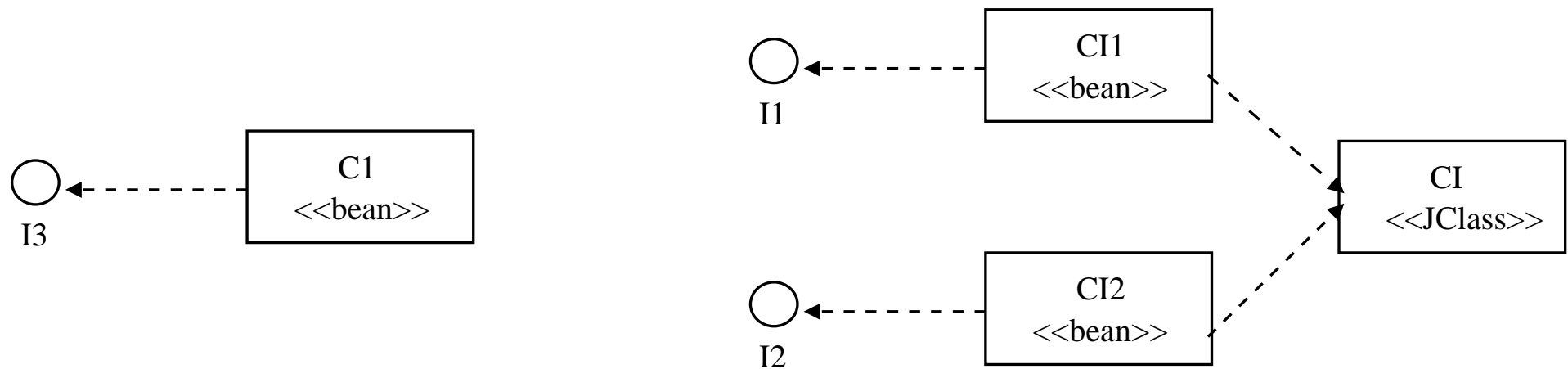


FIG. 6: Solutions envisageables en EJB

✓ Introduction du point de choix "delegate"

## Projections multiples (3/4)

✓ Exemple modèle abstrait :

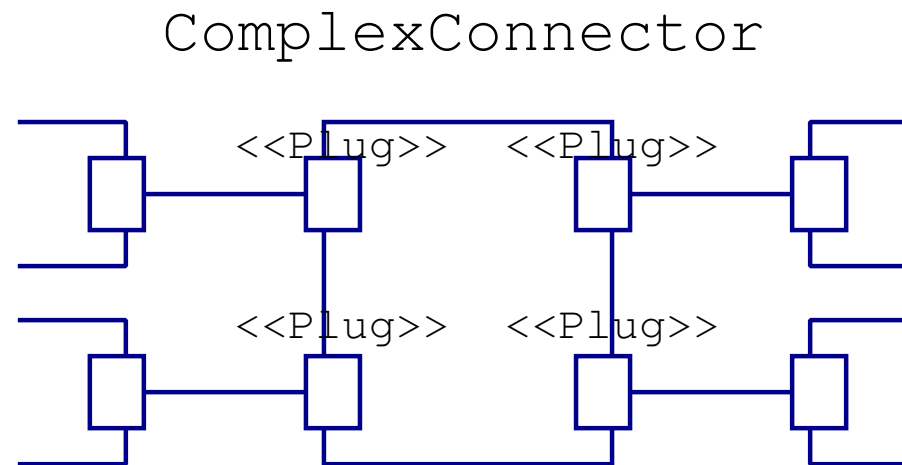


FIG. 7: Un connecteur-complexe ACCORD



## Projections multiples (4/4)

✓ Projections possibles :

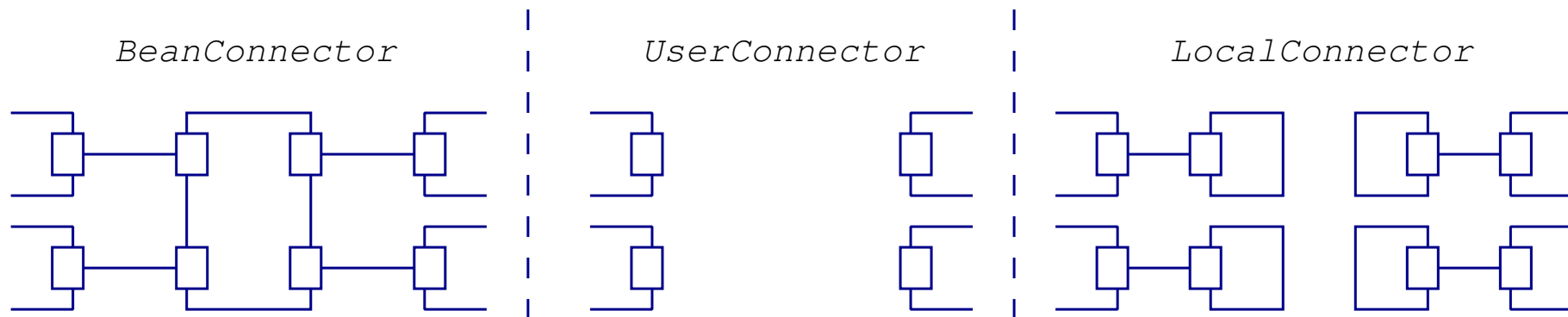


FIG. 8: Solutions envisageables

✓ Introduction des points de choix "BeanConnector", "UserConnector", "LocalConnector"

## Comparatif EJB et CCM

- ✓ Les études de projections EJB et CCM ont été menés en parallèle
- ✓ Apparition de points de choix communs
  - Exemple : traduction des connecteur-complexes
- ✓ Des distinctions dues à chaque modèle de composant :
  - ▶ Exemple CCM : projection du modèle **abstrait** ACCORD vers le modèle **abstrait** CCM
    - Report des spécifications d'implémentation dans le processus
  - ▶ Niveau EJB : prise en compte au niveau de la projection des spécifications d'implantations

## L'architecture de modèles

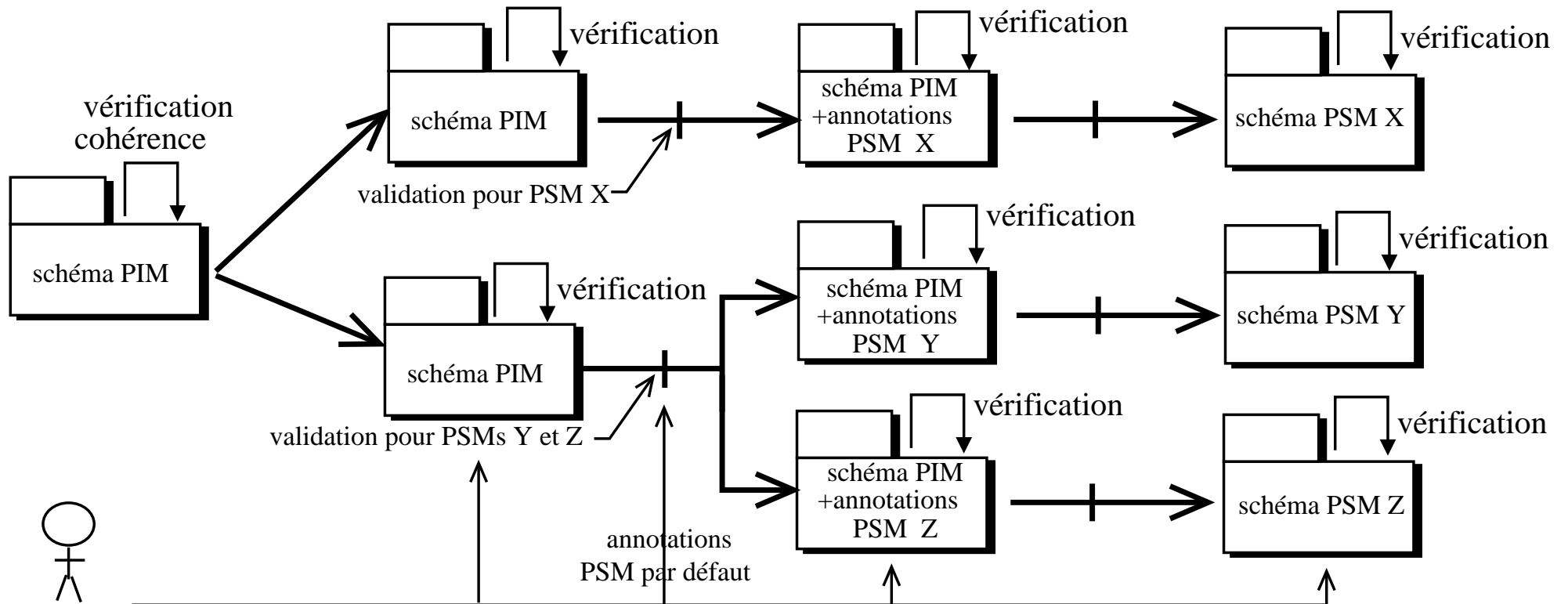


FIG. 9: Architecture unifiée de modèles

## Conclusion (1/2)

- ✓ Retour d'expérience MDA :
  - ▶ Objectifs CCM et EJB atteints
  - ▶ La transformation de modèles est un processus lourd (5000 lignes de script J)
  - ▶ Modules objecteering accessibles sur le site web ACCORD<sup>2</sup>
- ✓ Contribution OMG

---

<sup>2</sup>[http ://www.infres.enst.fr/projets/accord/](http://www.infres.enst.fr/projets/accord/)

## Conclusion (2/2)

- ✓ Une architecture générique de transformation basée sur les profils
- ✓ Meilleure spécification MDA pour les transformations de modèles par marquage :
  - ▶ Meilleure spécification de ces marques :
    - contraintes de pré-transformation
    - points de choix sous forme de valeurs marquées
  - ▶ Définition d'un processus d'introduction de ces marques
    - Guidé par l'architecture de profils
    - Ajout automatique de valeurs pas défaut
    - Possibilité d'adaptation manuelle