



# *Ingénierie dirigée par les modèles* *Model Driven Architecture*

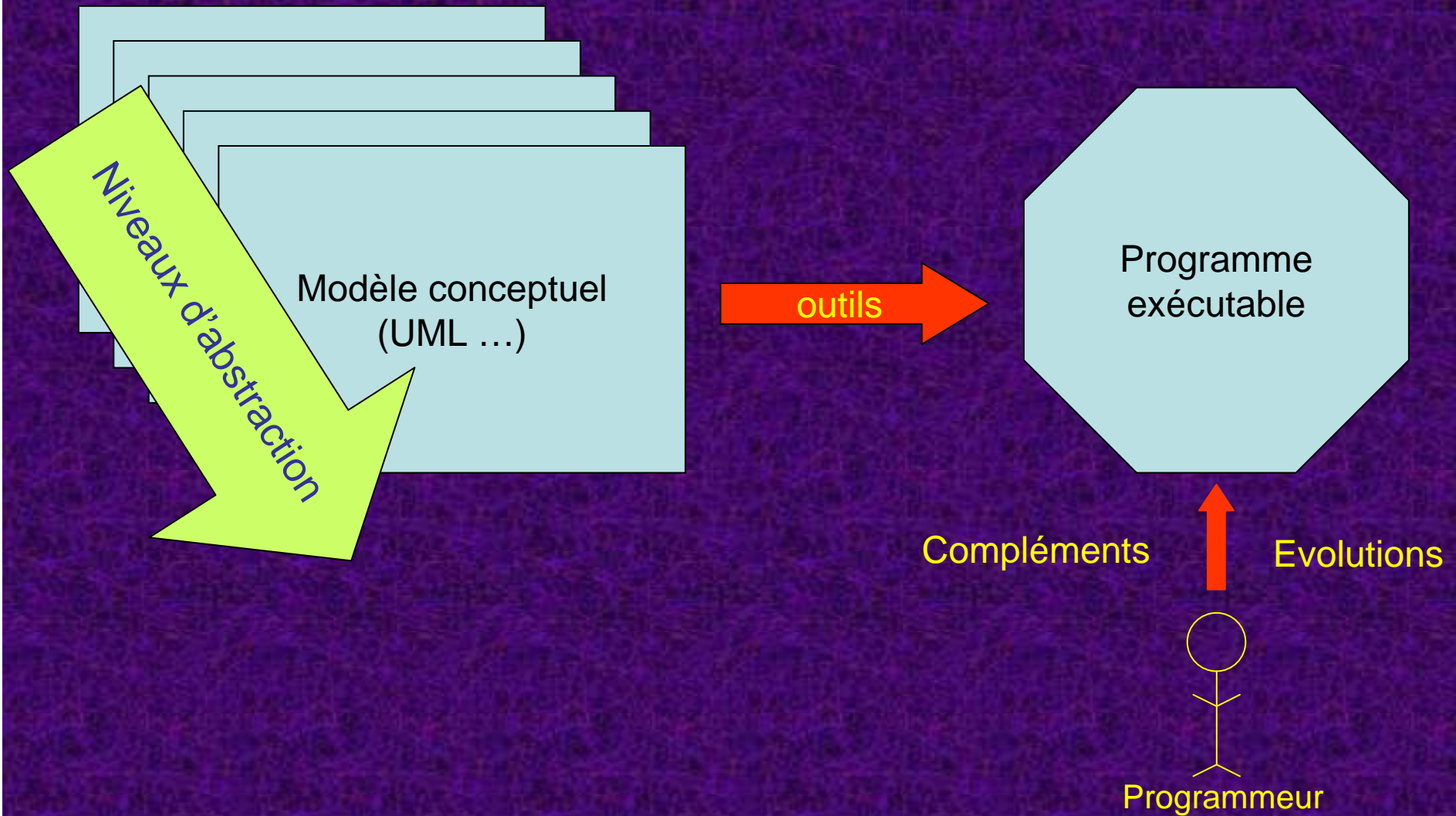
Laurent Pérochon

INRA

URH

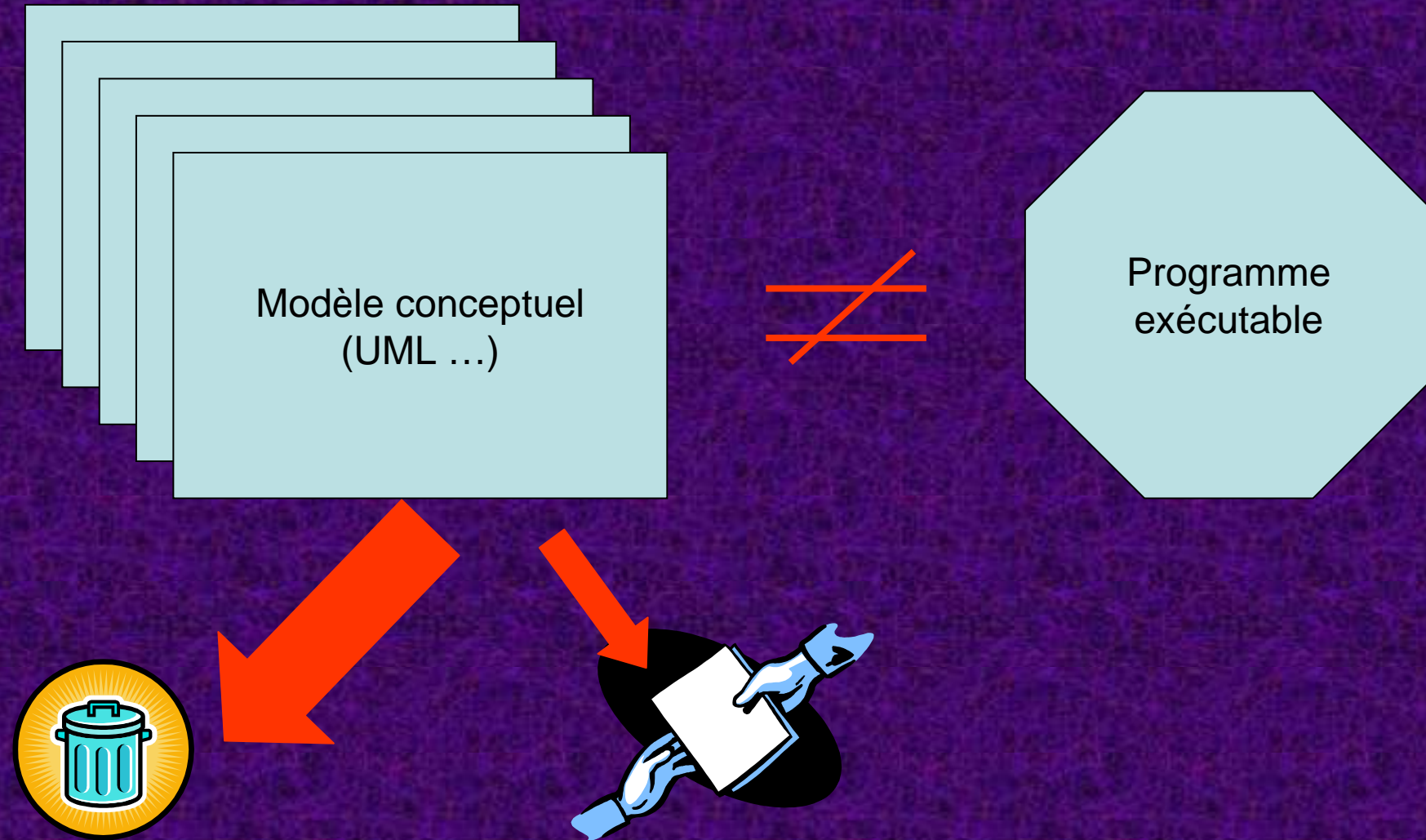
63122 Saint Genès Champanelle  
[perochon@clermont.inra.fr](mailto:perochon@clermont.inra.fr)

# La conception d'un programme classique



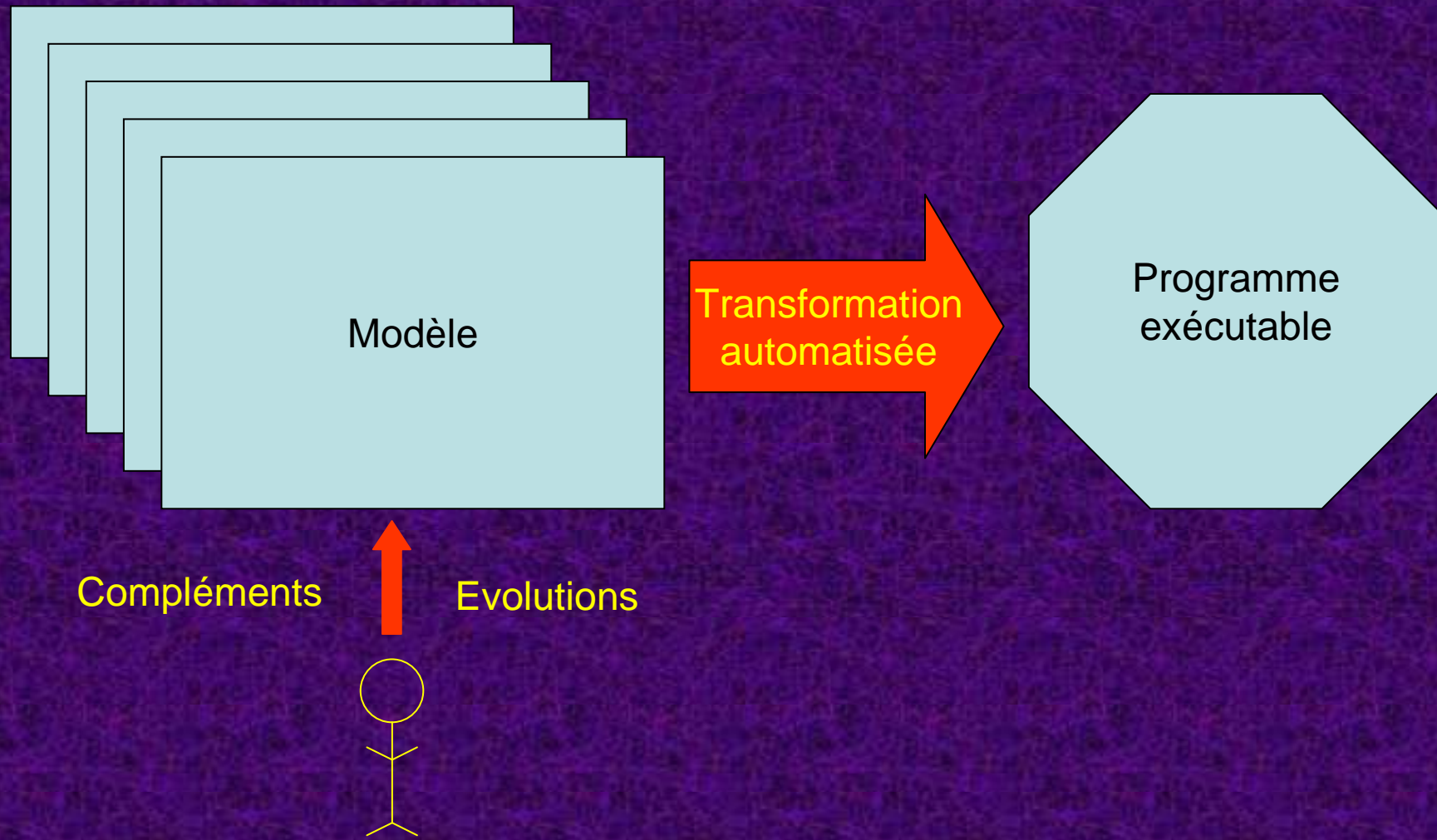


# Constat



# L'Ingénierie Dirigée par les Modèles (Model Driven ...)

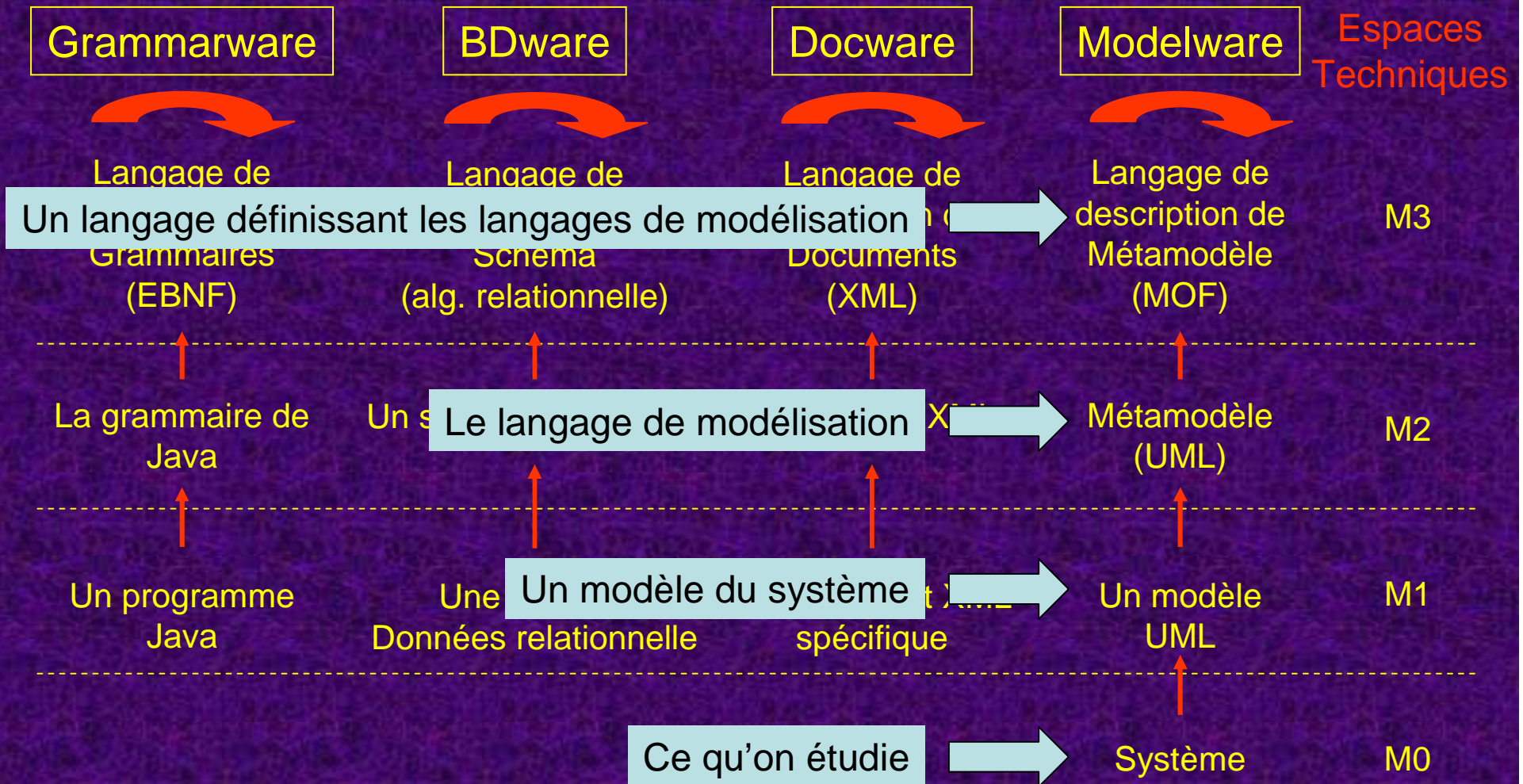
## Objectifs





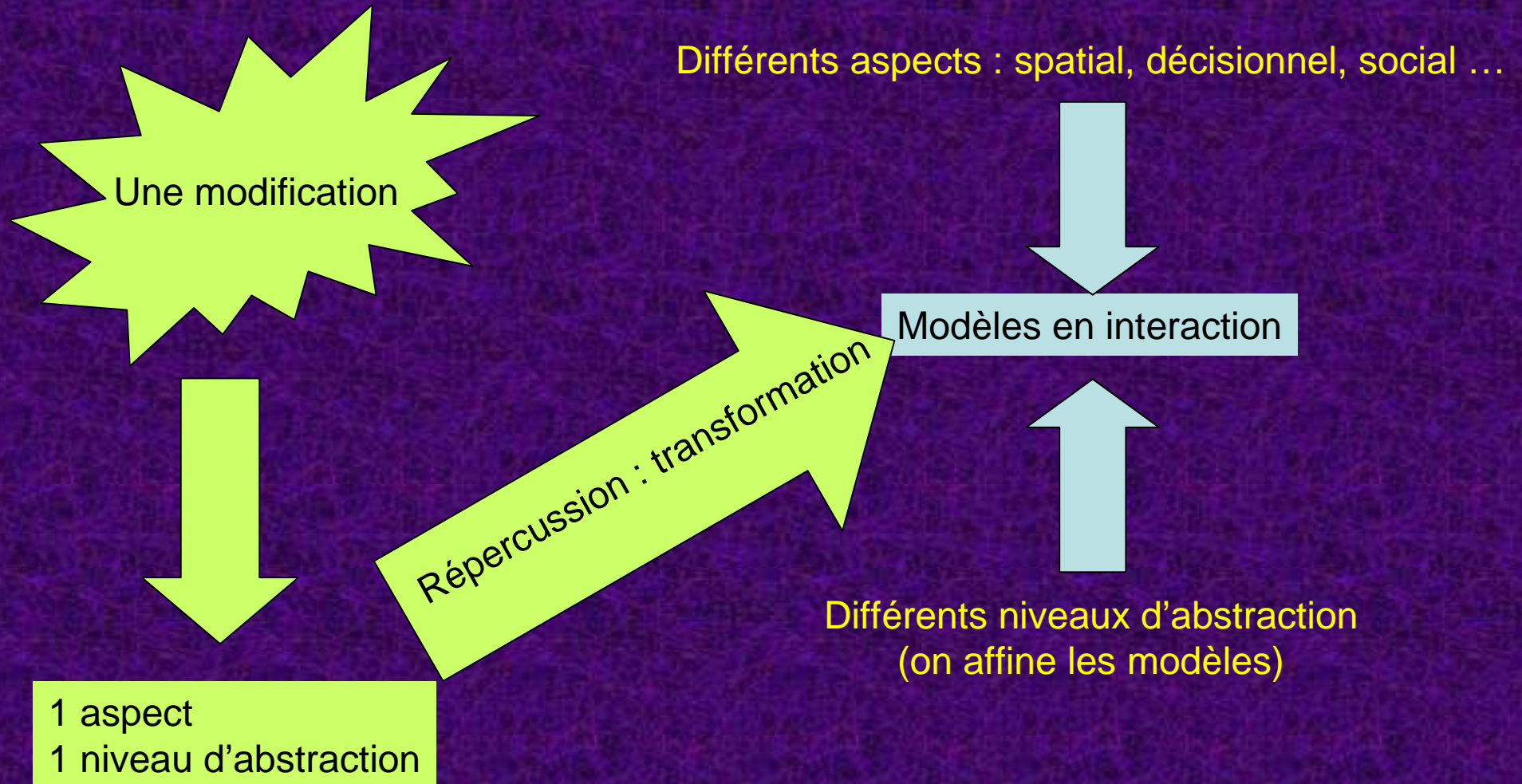
# L'Ingénierie Dirigée par les Modèles (IDM)

## Hiérarchie pyramidale / espaces techniques



# L'Ingénierie Dirigée par les Modèles (IDM)

## La manipulation des modèles : la transformation





# *L'Ingénierie Dirigée par les Modèles (IDM)*

## *C'est quoi ?*

- Intégration des techniques précédentes
- vise à couvrir tout le cycle de développement du logiciel
- On manipule des modèles
- On automatise les transformations entre différents modèles (aspects , niveaux d'abstraction)

L'élément de base n'est plus l'objet : c'est le modèle

*(Favre et al., 2006)*

Laurent Pérochon, programme ENVOL2008, Annecy 19 au 24 octobre 2008

# L'Ingénierie Dirigée par les Modèles (IDM)

## Les architectures basées sur l'IDM

Microsoft

Software factories

Domain Specific Langage

IBM

Eclipse Modeling Framework

Ecore (Essentiel MOF)

Object  
Management  
Group  
(OMG) UML

Model Driven Architecture  
(MDA)

Meta Object Facility (MOF)

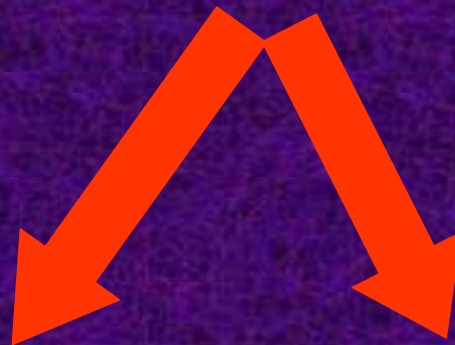


# *Model Driven Architecture (MDA)*

*L'ingénierie dirigée par les modèles vue par l'OMG*

Une architecture, mais pas une méthode de modélisation

Vision lancée en 2000, toujours en cours de développement



Des parties ont été créées

D'autres restent à développer

# Model Driven Architecture (MDA)

Modèle des exigences  
Computation Independent Model  
CIM

## Plateforme

Fournit des fonctionnalités,  
pas besoin de les programmer

Java, PHP, .Net ...

Modèle métier  
Platform Independent Model  
PIM

## Idées de base

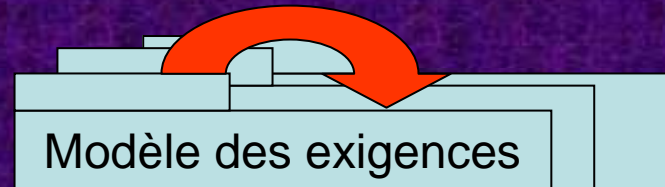
- Le métier évolue moins vite que la technique
- On doit pouvoir utiliser plusieurs plateformes, changer de plateforme
- Le PIM reste le PSM change

Modèle technique  
Plateforme Specific Model  
PSM

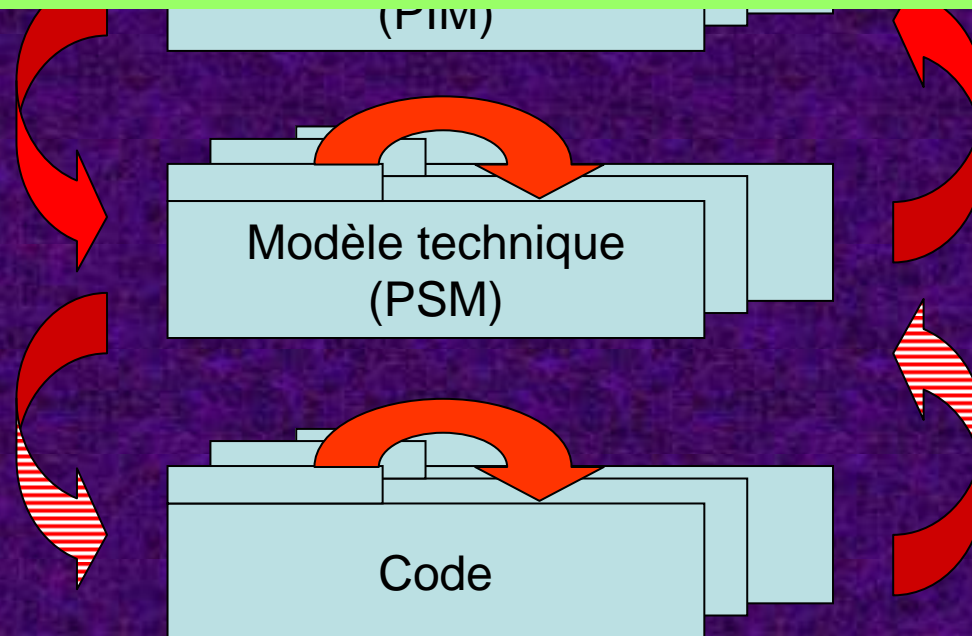
Code



## *Model Driven Architecture (MDA)*



On veut modéliser ces transformations  
pour faciliter les futurs évolutions



## *Model Driven Architecture (MDA)*

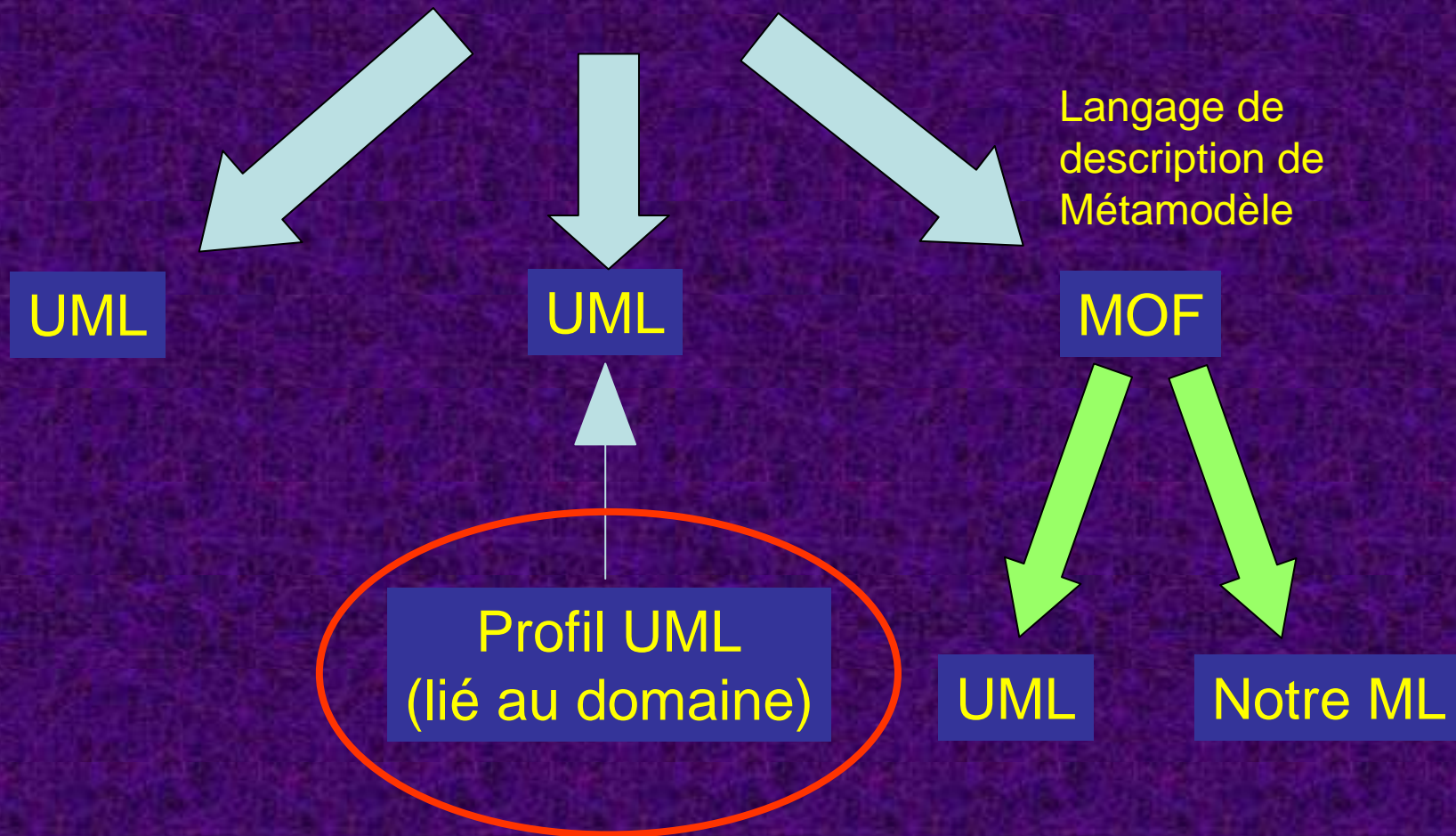
Comment représenter les modèles (CIM PIM PSM) ?

Comment représenter les transformations ?



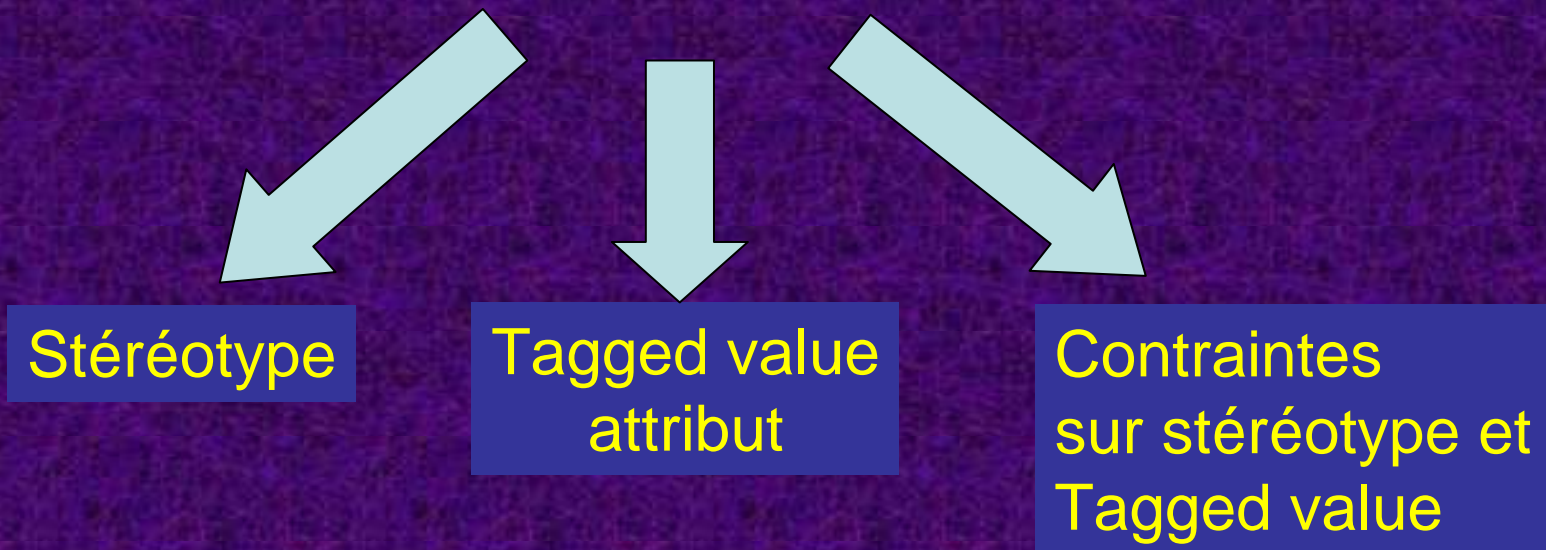
# Model Driven Architecture (MDA)

## Représenter les modèles



# *Model Driven Architecture (MDA)*

## Profils UML

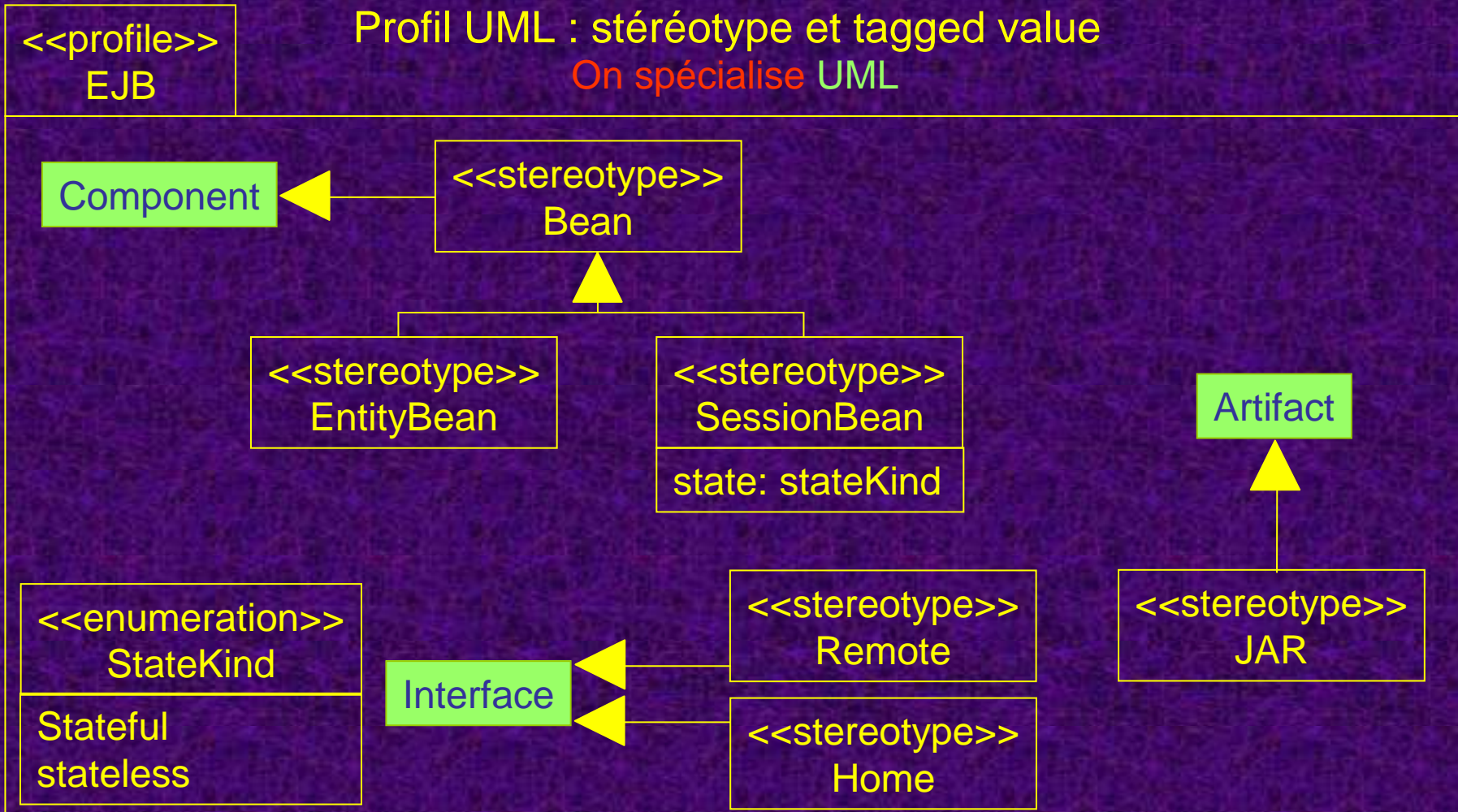




# Model Driven Architecture (MDA)

Profil UML : stéréotype et tagged value

On spécialise UML

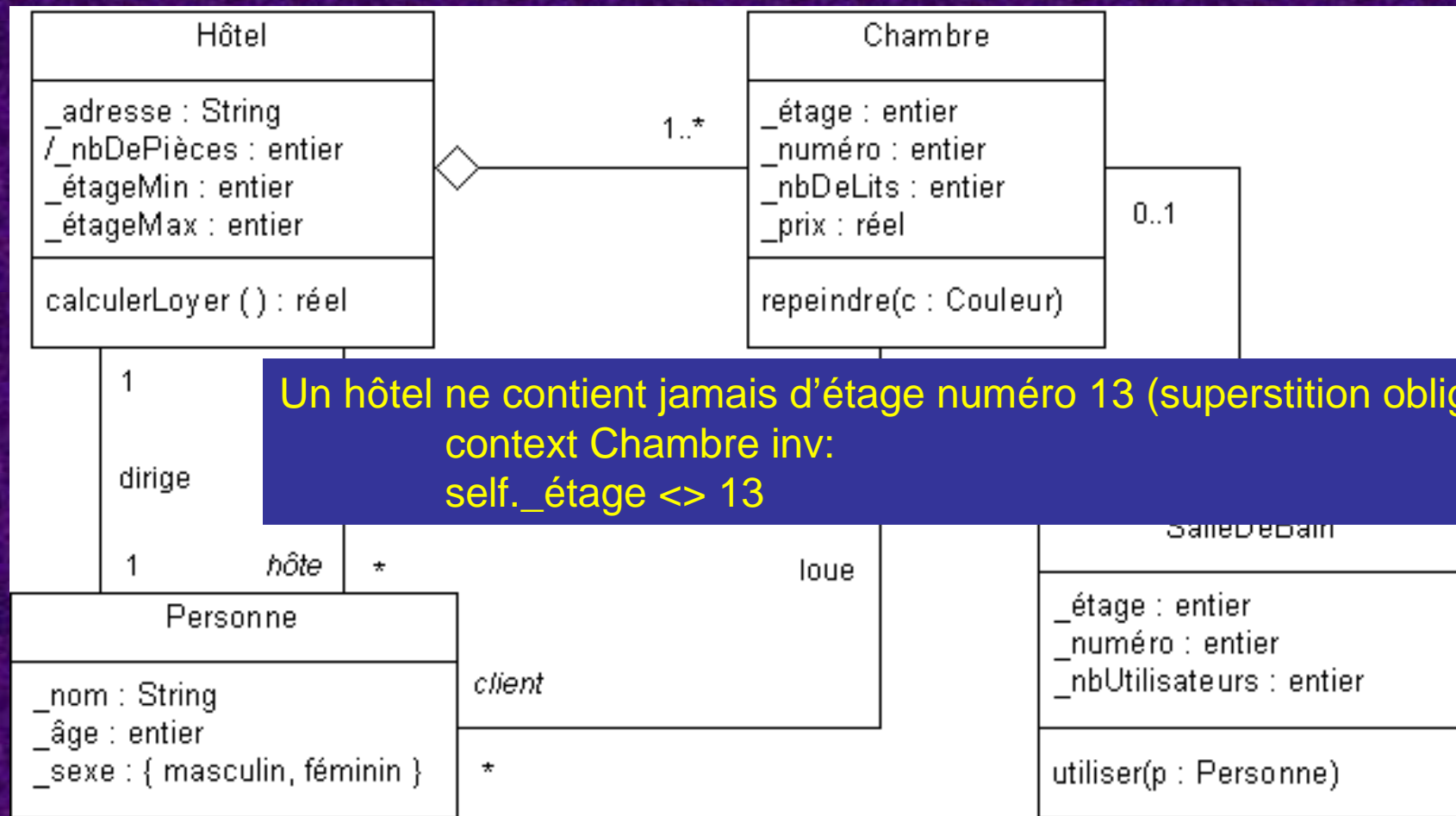


(T. Stahl et M. Völter, 2006)

Le nor context Chambre inv:

client->size <= \_nbDeLits or  
(client->size = \_nbDeLits + 1 and  
client->exists(p : Personne | p.\_age < 4))

le lits  
ans ne  
moins



Un hôtel ne contient jamais d'étage numéro 13 (superstition oblige).  
context Chambre inv:  
self.\_étage <> 13

(Olivier Caron : <http://www.polytech-lille.fr/~ocaron/>)

Laurent Pérochon, programme ENVOL2008, Annecy 19 au 24 octobre 2008



# *Model Driven Architecture (MDA)*

Profil UML : Modéliser les opérations

Object **Constraint** Langage (OCL)

Il ne permet pas des modifications dans le modèle

Exemple : augmenter le nombre de chambre de 1

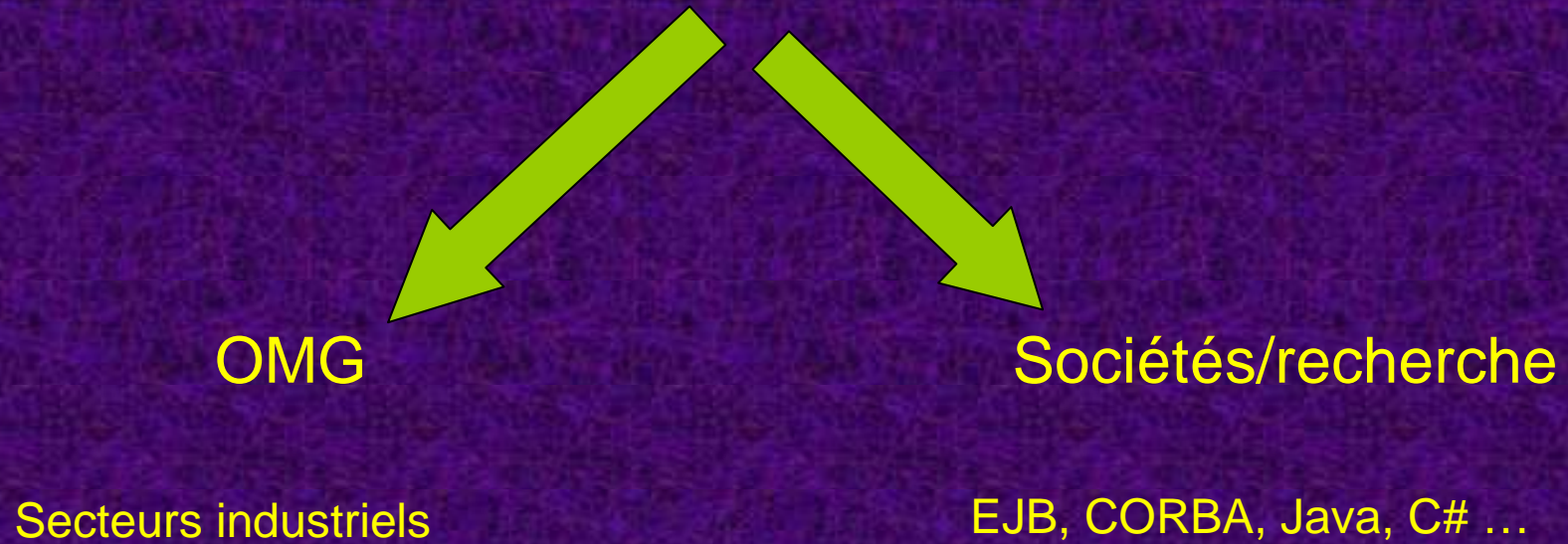
Action Semantic (AS). Standard OMG

Spécifier les actions, opération qui fait changer l'état du modèle

Pas de représentation textuelle : difficulté

# *Model Driven Architecture (MDA)*

Profils UML disponibles





# Model Driven Architecture (MDA)

## Bilan Profil UML

Description sous forme de diagramme des classes : ok

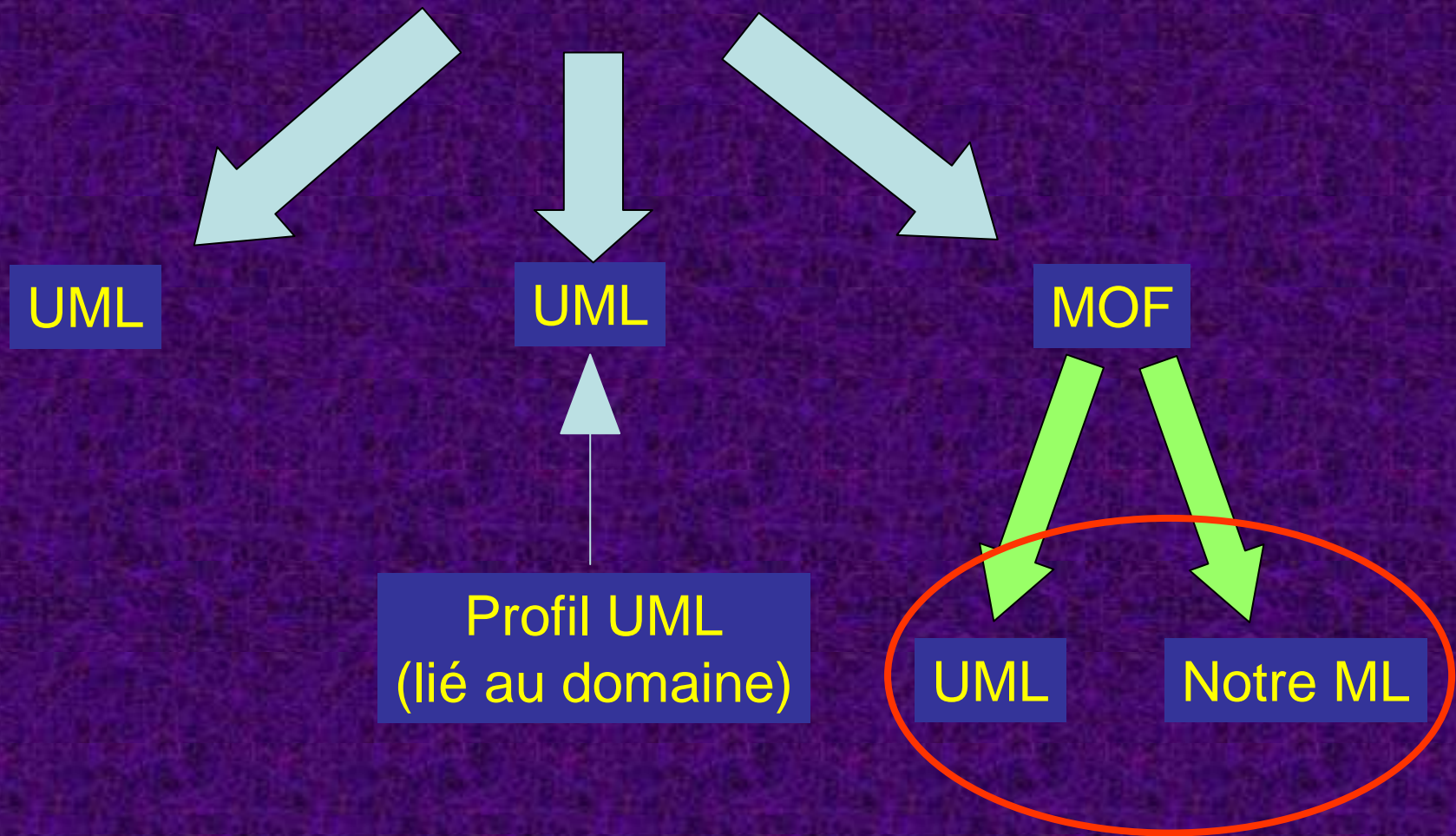
Object Constraint Language c'est bien mais ....

La majorité des outils ne supportent pas ou trop partiellement OCL durant une transformation (*Cabot et Teniente 2006*)

Action Semantic  
A voir dans le futur

# Model Driven Architecture (MDA)

## Représenter les modèles

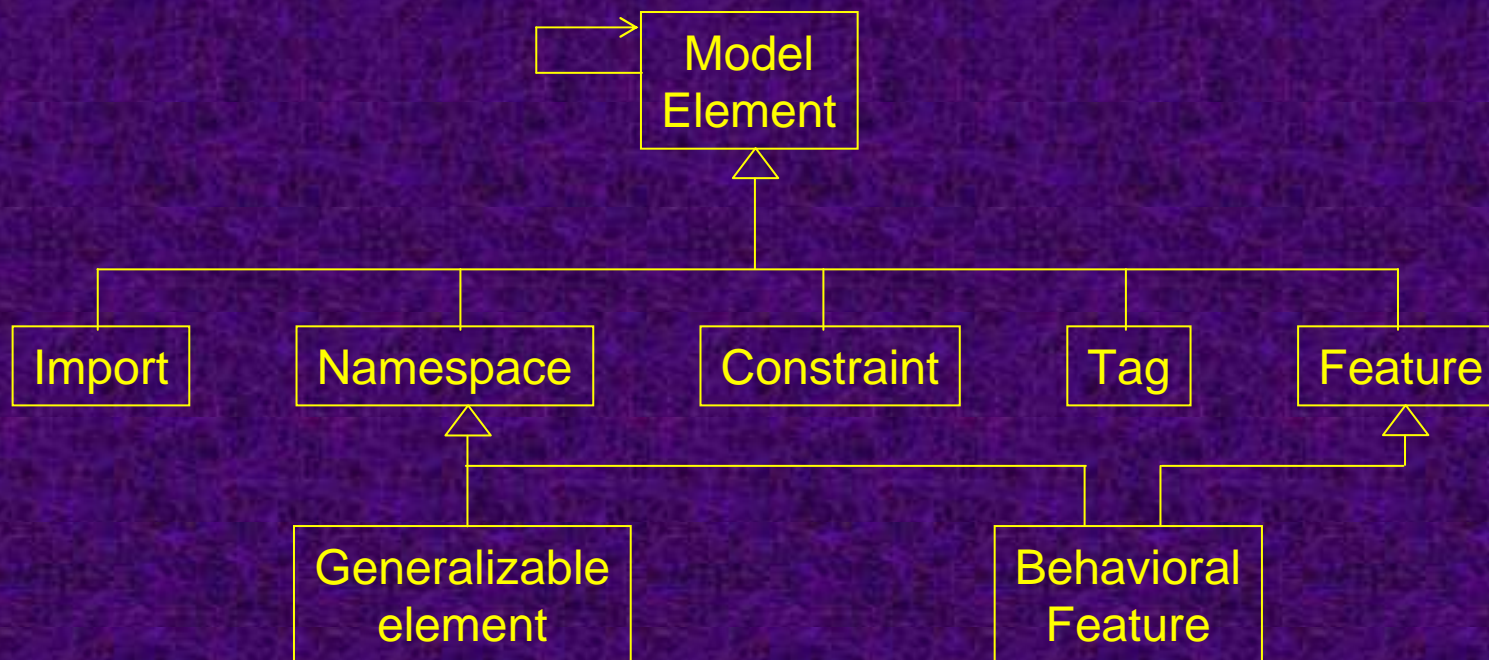




# Model Driven Architecture (MDA)

## Meta Object Facility

Le MOF est un langage de définition de métamodèles  
Il définit UML, pourquoi ne pas définir directement notre métamodèle du domaine ?

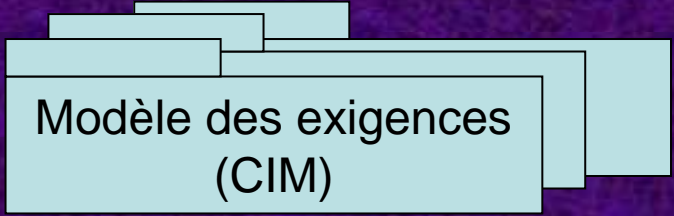


# Model Driven Architecture (MDA)

Représenter les modèles

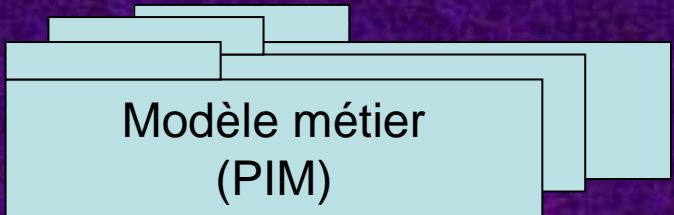
## Architecture

Cas d'utilisation



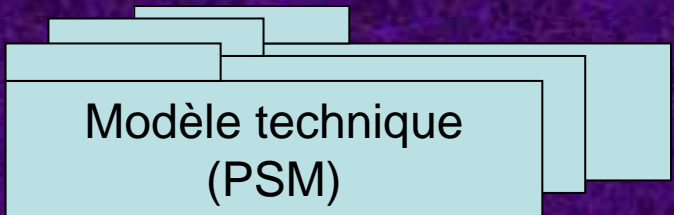
Modèle des exigences  
(CIM)

UML, Profils UML  
ou  
à partir du MOF



Modèle métier  
(PIM)

Principalement  
Profils UML



Modèle technique  
(PSM)



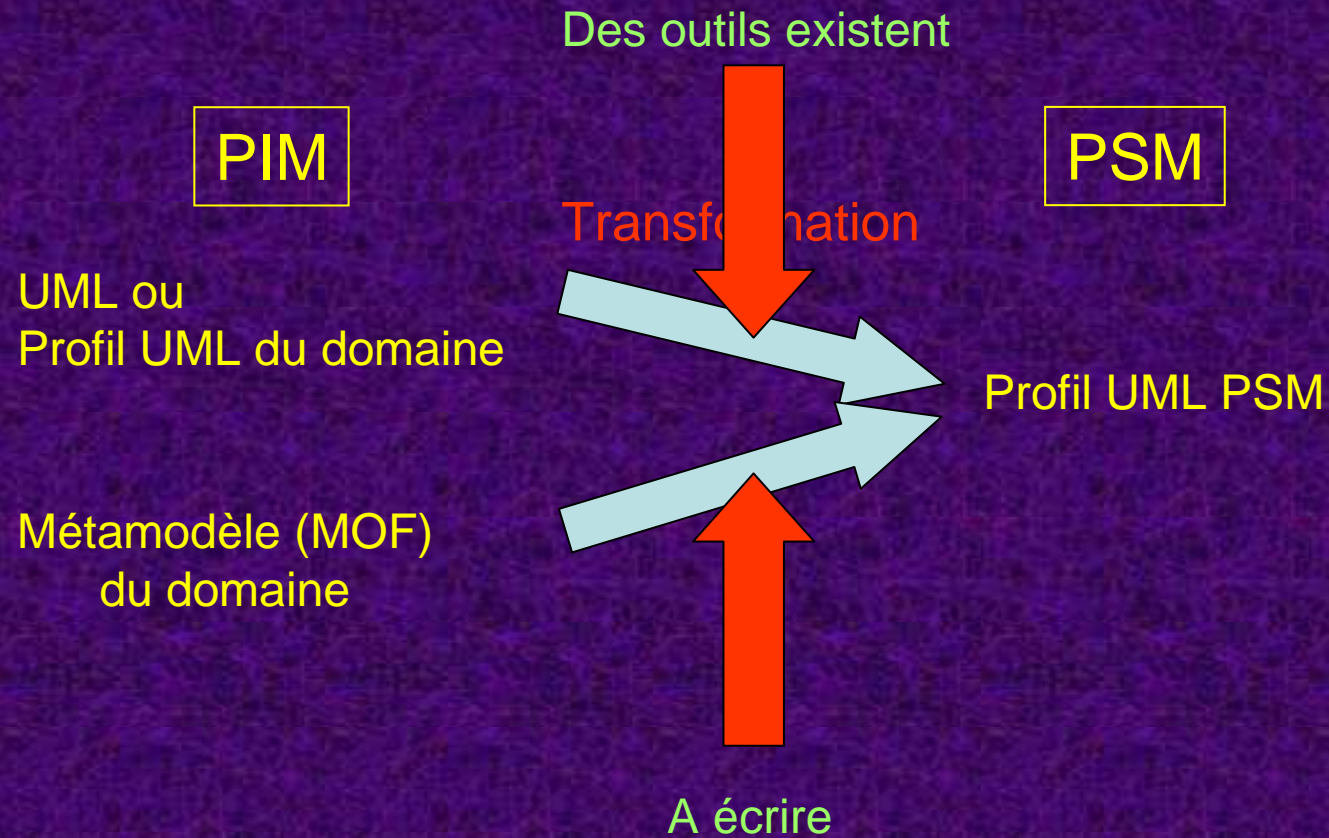
## *Model Driven Architecture (MDA)*

Comment représenter les modèles (CIM PIM PSM) ?

Comment représenter les transformations ?

# Model Driven Architecture (MDA)

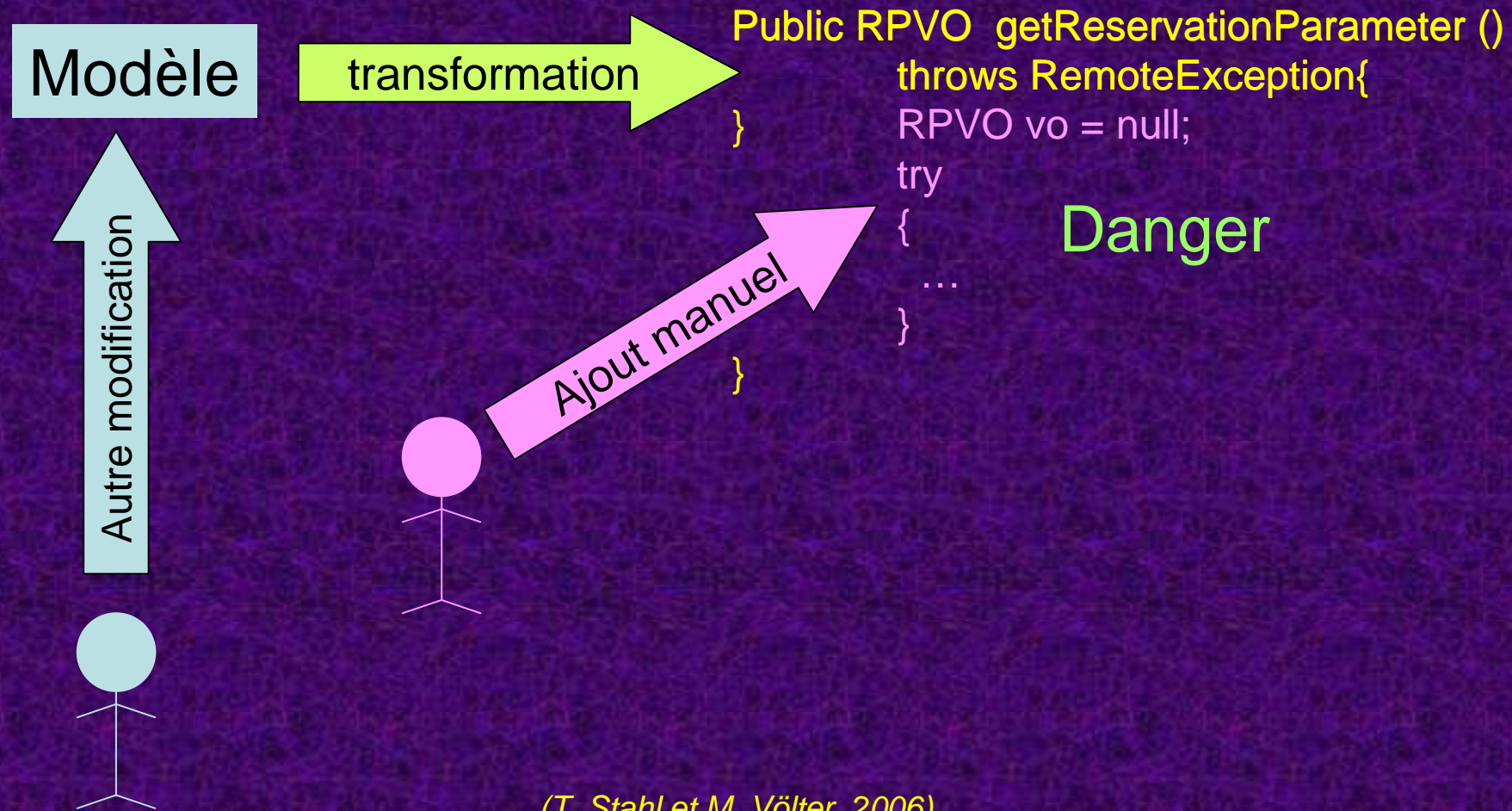
## Les transformations





# Model Driven Architecture (MDA)

Les transformations ne feront pas tout automatiquement : danger



## *Model Driven Architecture (MDA)*

Les transformations ne feront pas tout automatiquement

```
Public RPVO getReservationParameter ()
    throws RemoteException{
    // PROTECTED REGION ID(12) START
    RPVO vo = null;
    try
    {
        ...
    }
    // PROTECTED REGION END
}
```

*(T. Stahl et M. Völter, 2006)*



# *Model Driven Architecture (MDA)*

## les transformations

### Comment décrire les règles de transformation ?

- Approche par programmation
  - Java MetaData Interchange (JMI),
  - Eclipse Metamodel Framework (EMF)
  - MOF 2.0 to IDL (MOF vers Interface Definition Language)
- Approche par template
  - (paramètres du modèle template)
- Approche par modélisation. On modélise la transformation. Query View Transformation (MOF 2.0 QVT) : métamodèle des modèles de transformation.

# Model Driven Architecture (MDA)

## Exemple de transformation avec RubyTL

Module <module-name> do

```
rule <rule-name> do
  from <source-metaclass>
  to {target-metaclass}
```

Nom de la règle

Défini source et cible

```
  filter do |source_element|
    <expression>
  end
```

Filtre les éléments de la source

```
  mapping do |<source_element>, {target_element}|
    {bindings}
  end #de mapping
```

Décrit précisément les transformations

```
end #de rule
```

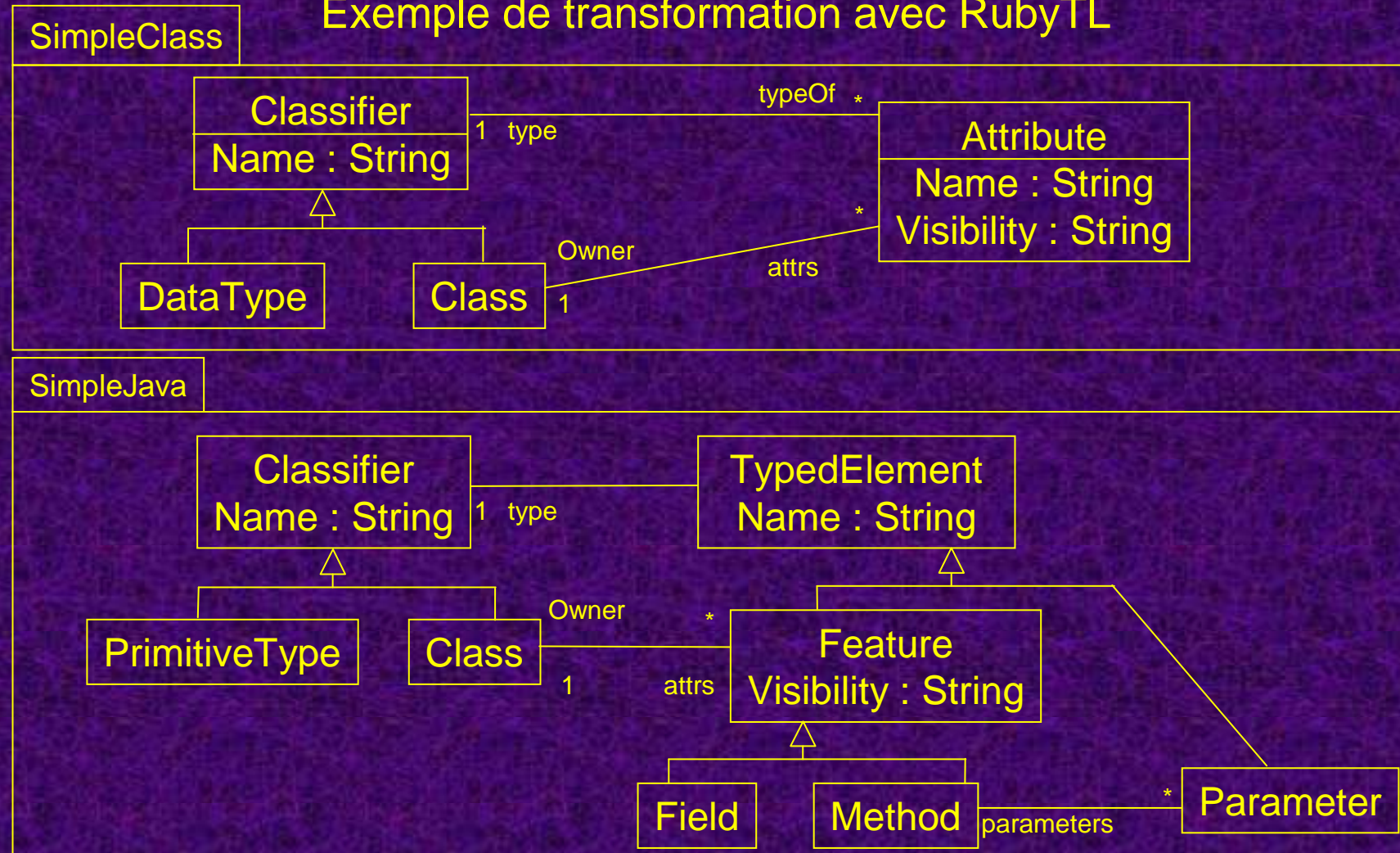
```
End #de Module
```

*(Cuadrado et al. 2006)*



# Model Driven Architecture (MDA)

## Exemple de transformation avec RubyTL

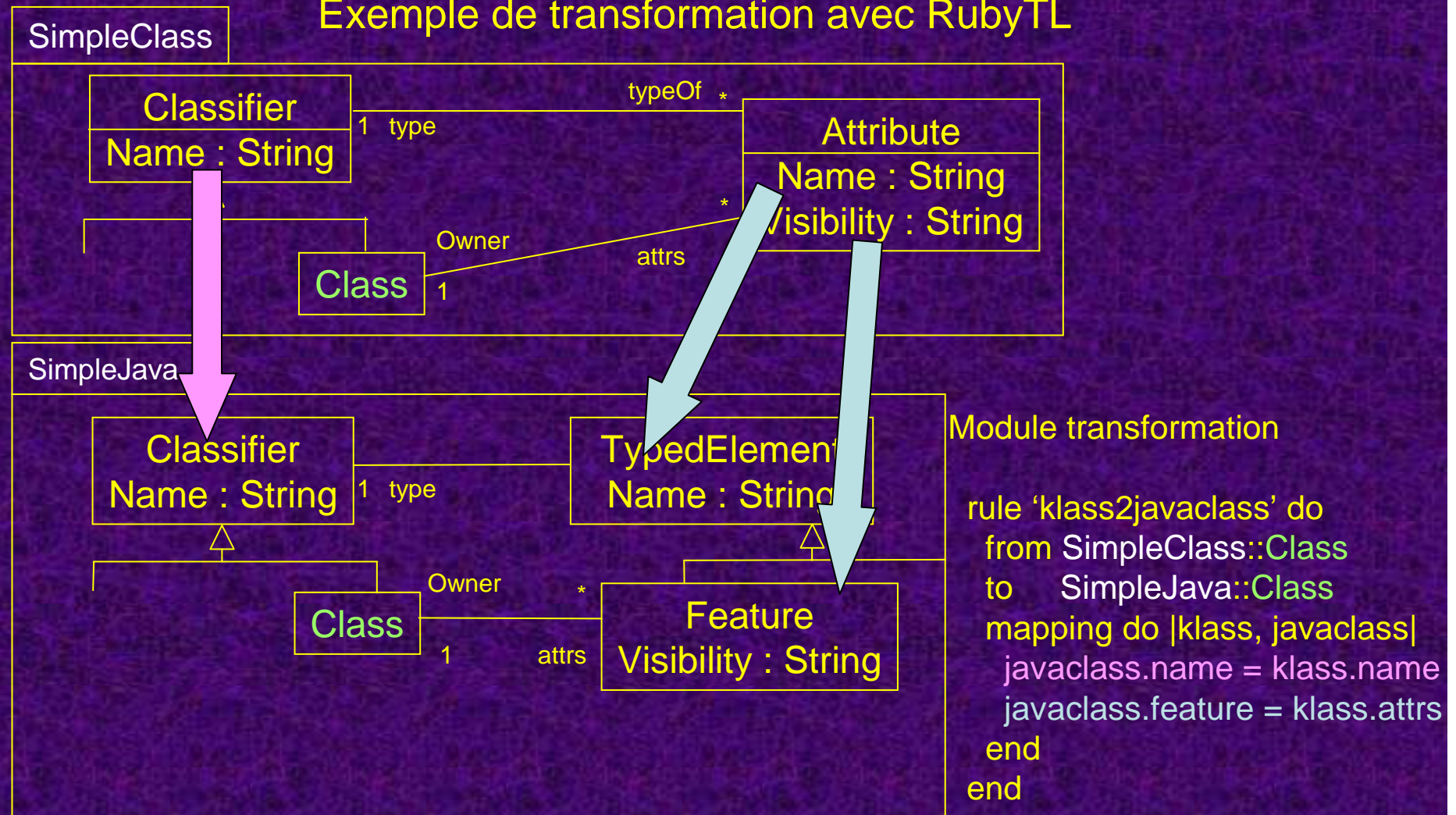


(Sánchez Cuadrado et al. 2006)



# Model Driven Architecture (MDA)

## Exemple de transformation avec RubyTL



(Sánchez Cuadrado et al. 2006)



# *Model Driven Architecture (MDA)*

## les transformations

Le problème récurrent :

Plusieurs règles peuvent pouvoir être appliquées en même temps (ordre ?)

Deux règles peuvent s'imbriquer

*(Sánchez Cuadrado et al. 2006)*

Laurent Pérochon, programme ENVOL2008, Annecy 19 au 24 octobre 2008

# *Model Driven Architecture (MDA)*

## Persistence et échange de modèles

enregister      Transférer entre applications



# *Model Driven Architecture (MDA)*

## Persistance et échange de modèles : XMI

XMI (XML Metadata Interchange) permet de représenter un modèle sous forme de document XML. (Pas les diagrammes).

On complète avec DI (UML Diagram Interchange) pour les outils gérant les modèles, sinon on utilise SVG (écrit en XML) pour les outils purement graphique.

XLST (eXtensible Stylesheet Language Transformation) permet de spécifier des transformations de documents XML (donc éventuellement de modèles). Transformation entre XMI, UML/DI vers SVG

## Problème

Actuellement XMI ne permet pas l'échange de modèle entre outils car il repose sur différentes versions de UML DI, XMI



# *Model Driven Architecture (MDA)*

Persistance et échange de modèles: alternatives à XMI

OMG :

MOF 2.0 to IDL

Java Community Process (JCP) SUN :

Java Metadata Interface (JMI)

IBM :

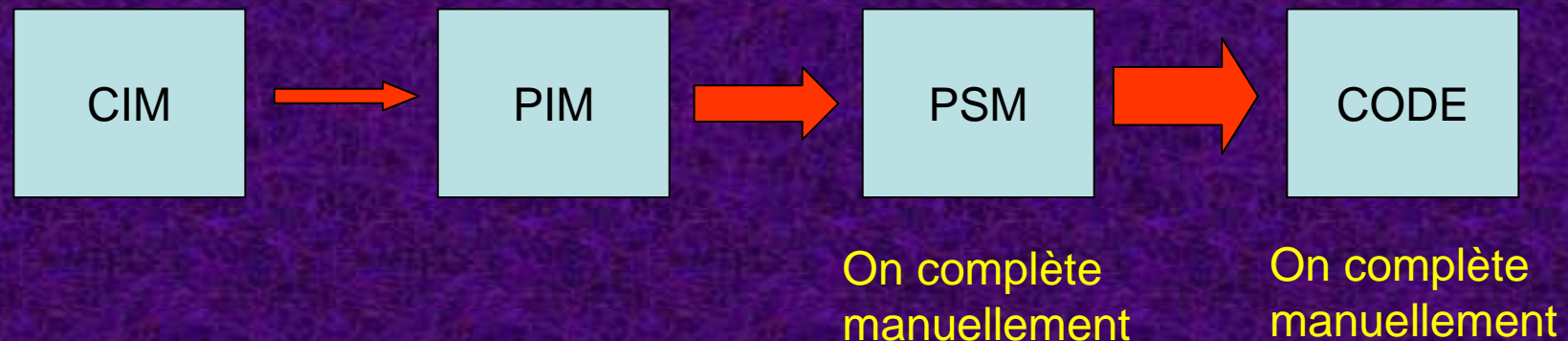
Framework Eclipse Modeling Framework (EMF)



# Model Driven Architecture (MDA)

## En bref

- Séparation infos métiers du technique : pérennité des savoir-faire
- Automatisation de modèles, productivité meilleur
- MDA est une approche à partir de CIM PIM et PSM, mais n'est pas une méthode
- Prise en compte des plateformes dans les PSM et les transformations PIM PSM



- Stockage des modèles pas encore vraiment standardisé
- transformations à améliorer

# *Model Driven Architecture (MDA)*

## Les outils

### Freeware

AndroMDA ([andromda.org](http://andromda.org))

Eclipse/Acceleo (Obeo)

Eclipse / Open Architecture Ware ([openArchitectureWare.org](http://openArchitectureWare.org))

Eclipse/ Generic Eclipse Modeling System (Vanderbilt University)

Eclipse / Xscarecrows 4 MDA

### Commercial

MDA Modeler (Softteam)

Eclipse/Rational Software Modeler (IBM)



# *Model Driven Architecture (MDA)*

Bilan (mitigé)

MDA or not MDA ?

**OUI**

C'est une architecture qui repose sur des standards  
Il y a de nombreuses recherches, de nombreux outils  
Le gain potentiel semble intéressant

**NON**

Tout n'est pas encore standardisé, voir même développé  
On doit compléter avec des approches/outils/langages non standardisés ou  
propriétaires (paradoxe avec le soucis d'indépendance aux plateformes)

**DONC ?**

Cela semble être la voie du future proche : Il faut donc en tenir compte.  
La séparation du domaine et de la technique est à réaliser. L'utilisation  
d'UML ou des profils UML semble aussi être un bon choix.



## *Bibliographie*

- Cette présentation est basée principalement sur le livre :  
Blanc X.. MDA en action. Ingénierie guidée par les modèles. Eyrolles. 2005. 269 pages.
- Pour un aperçu général sur l'ingénierie dirigée par les modèles :  
Favre J.M., Estublier J., Blay-Fornarino M. L'ingénierie dirigée par les modèles. Au-delà du MDA.. Lavoisier. 2006. 227 pages.
- Pour une vision pratique de l'IDM :  
Stahl T., Völter M. Model-Driven Software Development. Wiley. 2005. 428 pages.
- Pour une vision rapide :  
Revue IEEE Computer Society, Vol 39, N°2, 2006.
- Un profil UML en biologie ?  
Roux-Rouquié M., Caritey N., Gaubert L., Le Grand B., Soto M. Metamodel and modeling language : towards an unified modelling language (UML) profile for systems biology. SCI05, Orlando, Florida, USA, 2005. Disponible à [www.mcxapc.org/docs/ateliers/0506roux1.pdf](http://www.mcxapc.org/docs/ateliers/0506roux1.pdf). 2005



- Métamodélisation et ontologies :

Gašević D., Djurić D., Devedžić V. Model Driven Architecture and Ontology development. Springer. 2006. 311 pages

- Divers :

Cabot J., Teniente E. Constraint Support in MDA tools : a survey. Model Driven Architecture-Foundations and Applications. ECMDA-FA, Bilbao, Spain. 2006. Springer. Pp. 256-267.

Mellor S.J., Clark A.N., Futagami T. introduction: Model-Driven development. IEEE Software, 2003. vol 20, no 5, pp 14-18.

Roux-Rouquié M., Schuch da Rosa D. Ten Top Reasons for systems biology to get into Model-Driven Engineering. Proceedings of the 2006 international workshop on Global integrated model management. Metamodels and semantics. Pp. 55-58. 2006

Sánchez Cuadrado J., García Molina J., Menárguez Tortosa M. RubyTL: A practical, extensible Transformation Language. Model Driven Architecture-Foundations and Applications. ECMDA-FA, Bilbao, Spain 2006. Springer. Pp. 158-172.

- Sites internet :

[www.omg.org](http://www.omg.org)

[www.eclipse.org](http://www.eclipse.org)

[www.developpez.com](http://www.developpez.com)

exemple simple : [cedric-brun.developpez.com/mda/acceleo/acceleogmf/](http://cedric-brun.developpez.com/mda/acceleo/acceleogmf/)

Laurent Pérochon, programme ENVOL2008, Annecy 19 au 24 octobre 2008