

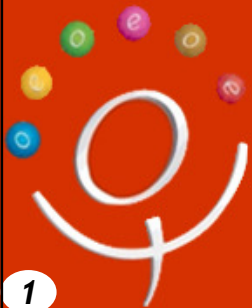
NEPTUNE 2009

MDE, DSL et UML, où en est on ?

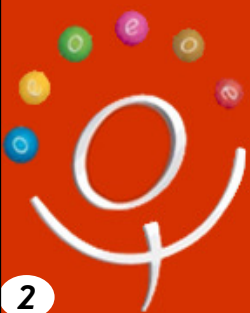
Philippe Desfray

SOFTEAM

www.objecteering.com, www.softeam.fr

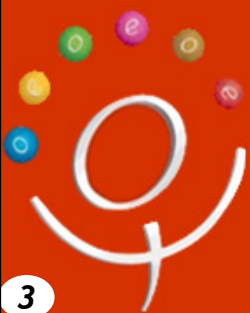


- **Des centaines de modèles différents**
- **Un modèle par pays, domaine : Merise, Coad Yourdon, ER, SASD, IDEF0, ...**
- **Pas ou rarement de métamodèles**
- **Des outils hétérogènes**
 - → fort cloisonnement, faible capitalisation
- **Les débuts de la modélisation par objet (+ 80 méthodes objet) aggravait la situation**
- **→ En pratique, MDA n'était pas envisageable:**
 - Dispersion des investissements
 - Difficultés de diffusion, partage, consolidation
 - Difficulté d'apprentissage

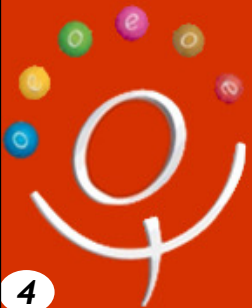




- **Une notation unique et standard :**
 - Connue de tous
 - Pour tous les domaines informatiques
 - Exploitable pour tout niveau de développement (Analyse préliminaire ... code)
 - Utilisée même en dehors du domaine de l'informatique (Organisation d'entreprise, Ingénierie des systèmes, etc.)
- **Une sémantique définie par un méta modèle**
- **Un format d'échange entre ateliers (XMI)**
- **Un mécanisme d'extension pour adapter UML à tout contexte (les profils)**
- **Des profils standard sont disponibles :**
 - cibles techniques, architecture (CORBA, Soc, soaML);
 - SysML (System modeling language) ...



- Les composants (EJB, .NET, etc.)
- Modélisation des architectures
- System engineering
- Les domaines « externes » : XML, EJB, SOAP, .NET, etc.
- Le domaine des réseaux, temps réel
- Les modèles de processus métier (activity diagrams, workflow), connexion EAI, WEB services, etc.
- UML exécutable



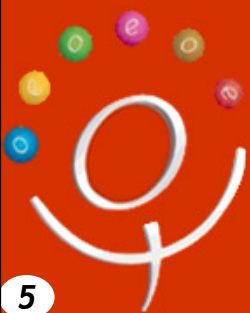
- UML est imparfait:

- Vraie difficulté: assembler plusieurs modèles en un tout cohérent.
- Bizarreries: Instance Specification/Parts,
- Complexités ou spécifications mal construites : ports
- Ambiguïtés sémantiques : Exemple des Use Case/Extension points

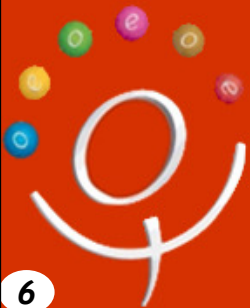
The concept of an “extension location” is intentionally left underspecified because use cases are typically specified in various idiosyncratic formats such as natural language, tables, trees, etc. Therefore, it is not easy to capture its structure accurately or generally by a formal model. The intuition behind the notion of extension location is best explained through the example of a textually described use case: ...

- Mais:

- C'est le standard universellement reconnu
- Il est suffisamment flexible/extensible pour que chacun y puise son mode d'usage



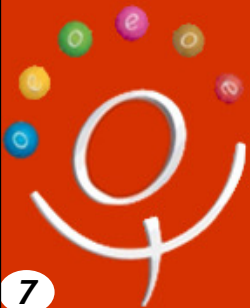
- **Addition des modèles les plus utilisés: E/R, SM, Modèles de traitement, MSC, Composants, ADL, etc.**
- **Addition de plusieurs points de vues: systèmes techniques, SI, embarqué, BPM, etc.**
- **Spécialisation par domaine avec les profiles**
- **→ UML a une très large couverture des capacités de modélisation et pratiques connues**



- **Nécessité de suivre la voie du standard**

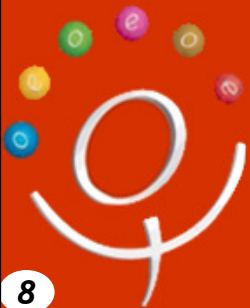
- Il est très onéreux et risqué d'inventer de nouveaux modèles
- Risque d'autisme des outillages spécifiques et des "experts" :
 - ✓ Compréhension non partagée des modèles
 - ✓ Exchange des modèles entre organisations difficile
 - ✓ Outillages spécifiques à maintenir
- Disponibilité d'outils et de paramétrages "sur étagère" : offres commerciales, paramétrages diffusés par des organisations

→ UML et les profiles sont la solution du standard

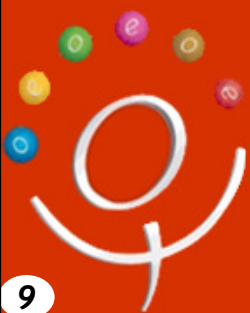


- **Acceptation dans toutes les cultures?**
 - Maitrise d'ouvrage l'adopte difficilement
 - Modélisation des données
 - Modélisation des processus métiers
 - Architecture d'entreprise

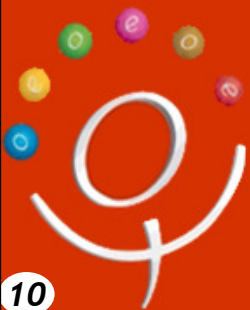
- **Couverture de tous les besoins de modélisation?**
 - Difficultés pour les éléments de cadrage: Objectifs, Dictionnaire, Exigences, Règles métier



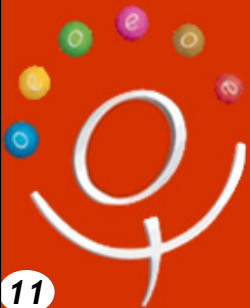
- **UML couvre presque tous les modèles usuels**
 - Sauf par exemple réseaux de Petri, relationnel binaire, qui sont plutôt « confidentiels »
- **L'emploi d'alternatives à UML est essentiellement culturel**
 - Habitudes antérieures à UML
 - Rejet de la complexité UML
 - Population cible non technique
- **UML est plus adopté**
 - Par les populations techniques que métier
 - Par les domaines techniques (telecom, avionique, ...) que tertiaire



- **UML peu accepté en MOA : Graphisme, complexité, jugé trop « technique »**
- **UML pour qui? Compréhension des décideurs, adoption par les différents interlocuteurs**
- **UML répandu en MOE : Analyse, conception technique et développement**
- **UML souvent rejeté par les programmeurs: « perte de temps »**



- **Acceptation des formalismes**
 - Compréhension immédiate
 - Pas ou très peu d'apprentissage dans les parties amont
- **Formalismes adaptés au problème traité**
 - Reposer sur les aspects : Concepts métier; Organisation; Géographie; Matériel; Architecture logique; etc.
- **Des éléments informels en première approche, puis formels dans les détails**
 - Eléments textuels
 - Vue générale d'organisation



Développement logiciel : petite partie des problèmes entreprises

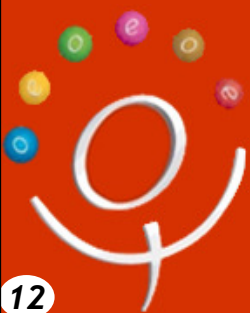
- Stratégie
- Organisation, Processus
- Intégration avec système existant
- ...

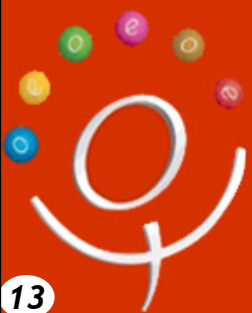
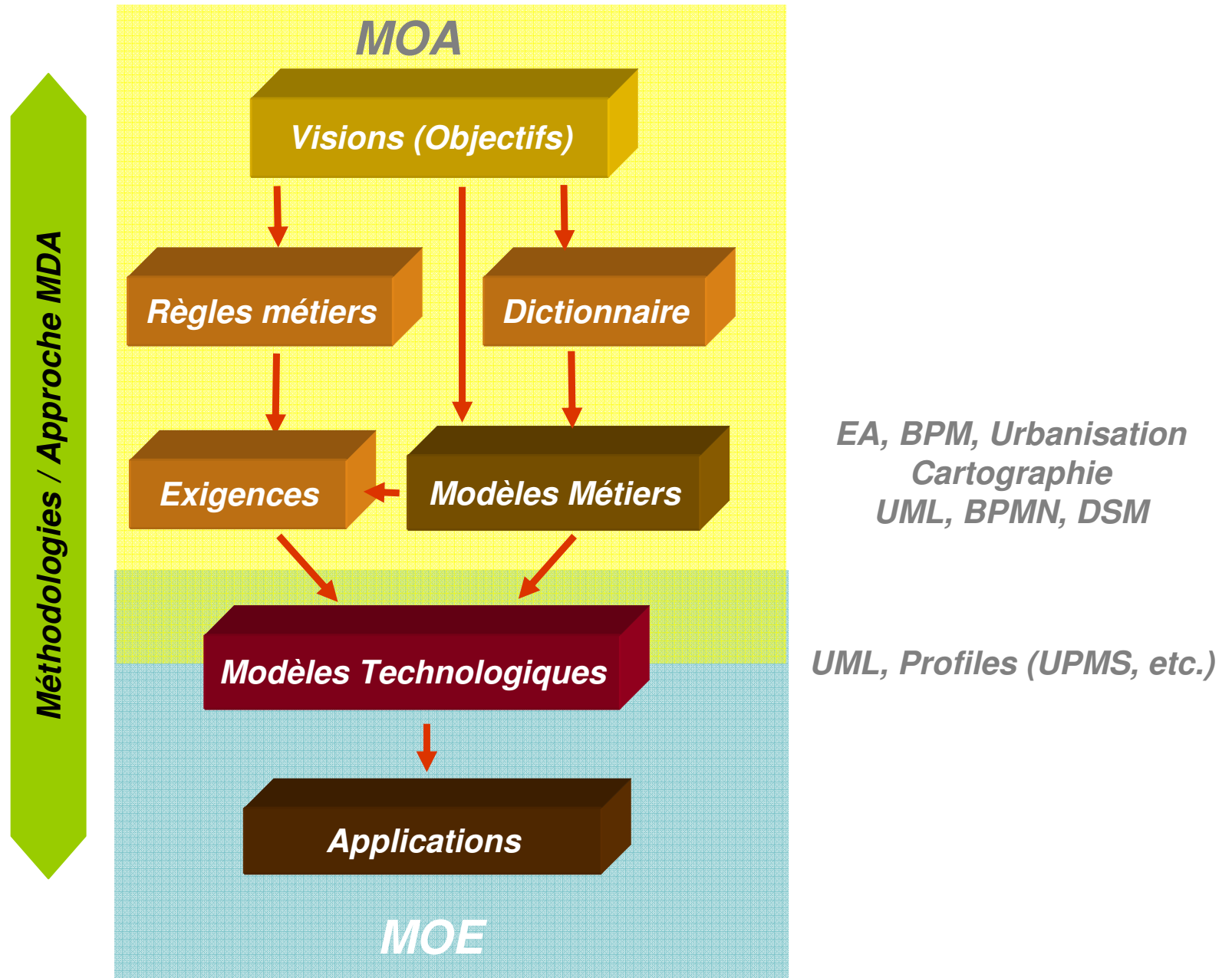
MDA est trop cantonné:

- Logiciel
- Génération code ou architecture technique

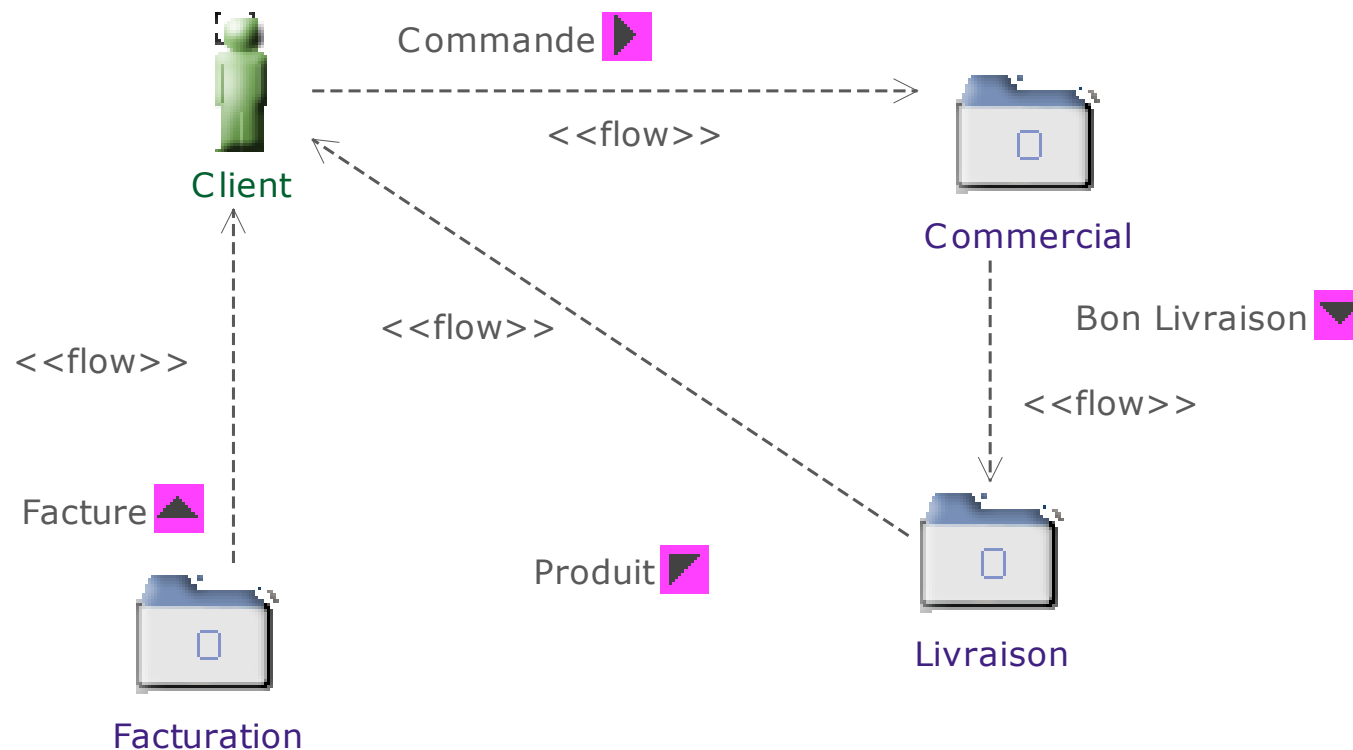
MDA doit être davantage utilisé pour

- Soutenir les méthodologies
- Soutenir la modélisation sur toute la portée de l'entreprise

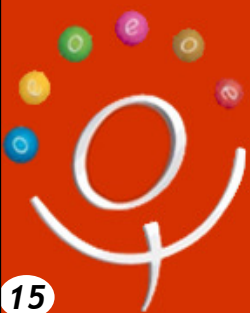




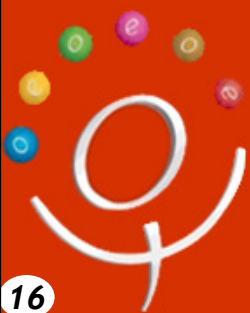
- **Exemple: Modèle d'organisation (approche systémique)**
- **Le modèle d'organisation représente les structures de l'entreprise impliquée dans les processus métier, avec les échanges d'information.**



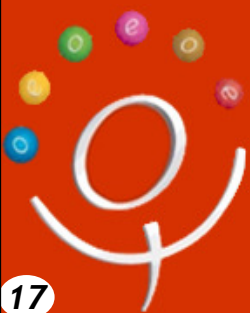
- Les domaines « Enterprise Architecture », « Business Process Modeling » restent séparés de UML
- Des outils dédiés très séparés des outils UML
- Il y a doublon de standards (BPMN, ODM,)
- → Séparation « artificielle » des intervenants MOE et MOA
- Mais il y a des initiatives de Profiles UML pour le métier.



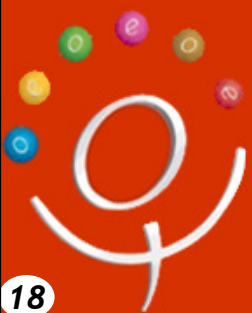
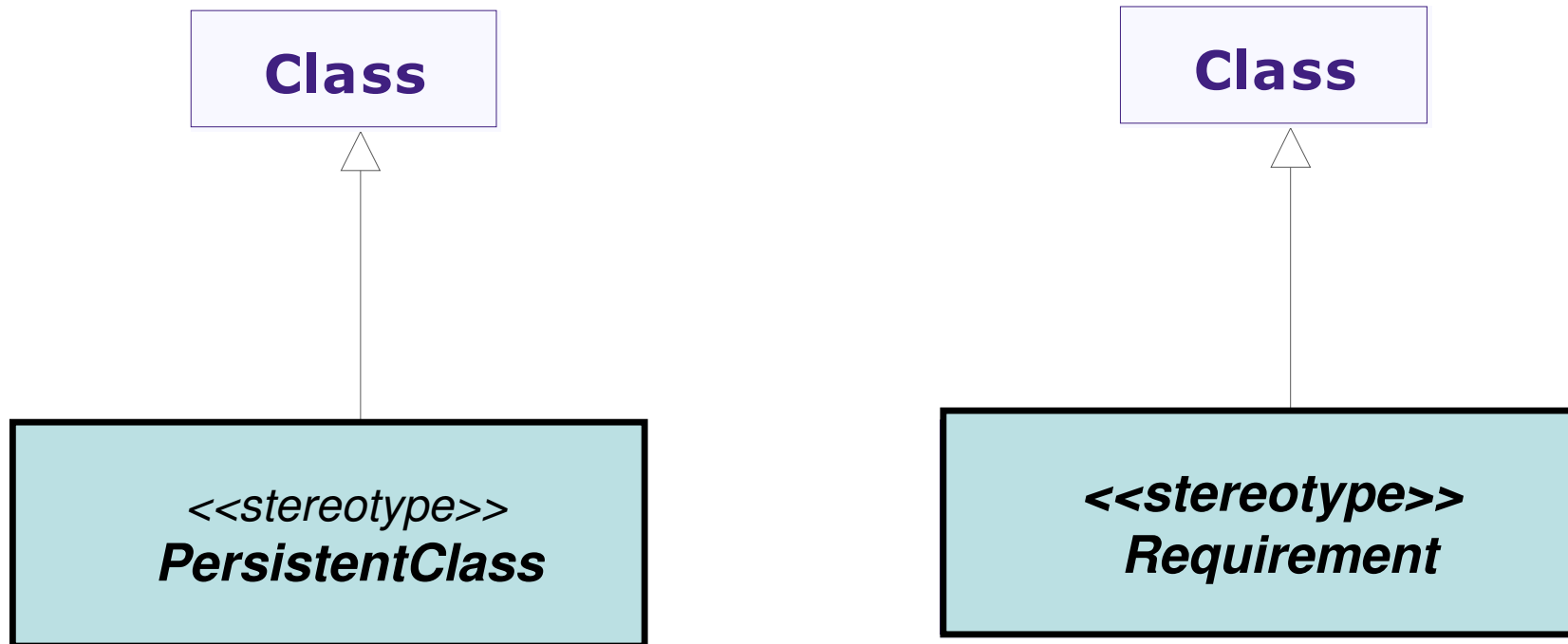
- **Quelle est la différence?**
 - UML + Profile = DSL
- **Large confusion entre:**
 - Le besoin : avoir un modèle dédié à une famille de problèmes
 - L'approche technique : profiler UML ou métamodéliser
- **Pollutions intellectuelles**
 - Guéguerre Profile/Métamodèles
 - Arrières pensées marketing : position Microsoft, et autres acteurs
- **UML n'a pas que des supporters**



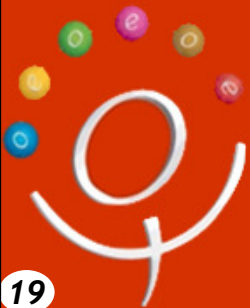
- **C'est presque pareil, mais:**
- **Profiler UML:**
 - - Contraint à réutiliser un existant technique (métamodèle UML)
 - + permet de bénéficier immédiatement des services des outils support
 - + permet d'être appliqué/retiré dynamiquement sur un modèle existant
 - + peut être combiné avec d'autres profiles
 - Permet d'échanger des modèles entre environnements hétérogènes, en gardant l'intersection commune (> UML)
- **Métamodéliser**
 - + permet de cibler n'importe quel domaine (ex: chimie)
 - + n'impose pas de contrainte « sémantique et technique » de réutilisation d'un modèle
 - - risque d'autisme, non standard, outillage « from scratch », apprentissage, ...



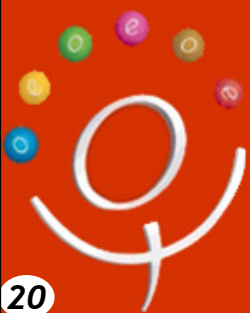
- Dépend de la cohérence sémantique UML/modèle à profiler.



- **Profiles:**
 - Des outils UML avec un bon support des profils
 - Un standard XMI enfin stabilisé
 - Une mise en œuvre des profils très répandue
- **DSL:**
 - Une offre open source prédominante (EMF)
 - Des outils commerciaux matures
- **Des standards pour la métamodélisation, transformation de modèle, expression de règles de cohérence**



- **Microsoft s'investit dans UML – présence OMG**
- **Vers une nouvelle évolution MOF et UML**
 - Rationalisation du métamodèle UML
 - Notion de point de vue et de vue
 - Rapprochement profiles/métamodélisation?
- **Les positions DSL/UML se clarifient**
- **UML et les Profiles poursuivent leur « banalisation »**



- **UML est un standard reconnu et très diffusé**
- **UML doit encore convaincre certain types d'utilisateurs**
- **UML2 contient plusieurs évolutions significatives pour**
 - Beaucoup d'activités (BPM, phases amont, architecture)
 - De nombreux domaines (SOA, system engineering, embarqué)
 - La bonne couverture des langages actuels
- **Un nombre croissant de profils standards sont sur étagère (SysML, SOAML, MARTE)**
- **Les outils UML2 doivent apporter une plus value fonctionnelle et un fort support MDA**
- **Le point d'équilibre n'est pas trouvé (IDE, MDA, langages, Frameworks), mais MDA et UML étendent leurs domaines d'utilisation et leurs capacités**

