

1

Métamodélisation en UML

Du modèle au métamodèle

Syntaxes abstraites et concrètes

D'après Eric Cariou

2

Rappel : les 4 niveaux de l'OMG

- L'OMG définit 4 niveaux de modélisation
 - M0 : système réel, système modélisé
 - M1 : modèle du système réel défini dans un certain langage
 - M2 : méta-modèle définissant ce langage
 - M3 : méta-méta-modèle définissant le méta-modèle
- Le niveau M3 est le MOF
 - Dernier niveau, il est méta-circulaire : il peut se définir lui même
- Le MOF est – pour l'OMG – le méta-méta-modèle unique servant de base à la définition de tous les métamodèles

Méta-modélisation UML

3

Dans UML, on retrouve également les 4 niveaux, mais avec le niveau M3 définissable en UML directement à la place du MOF

Exemple de système réel à modéliser (niveau M0)

- Une pièce possède 4 murs, 2 fenêtres et une porte
- Un mur possède une porte ou une fenêtre mais pas les 2 à la fois
- Deux actions sont associées à une porte ou une fenêtre :
 - ouvrir et fermer
- Si on ouvre une porte ou une fenêtre fermée, elle devient ouverte
- Si on ferme une porte ou une fenêtre ouverte, elle devient fermée

Cours de Cariou

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

Métamodélisation UML

4

Pour modéliser ce système, il faut définir 2 diagrammes UML : niveau M1

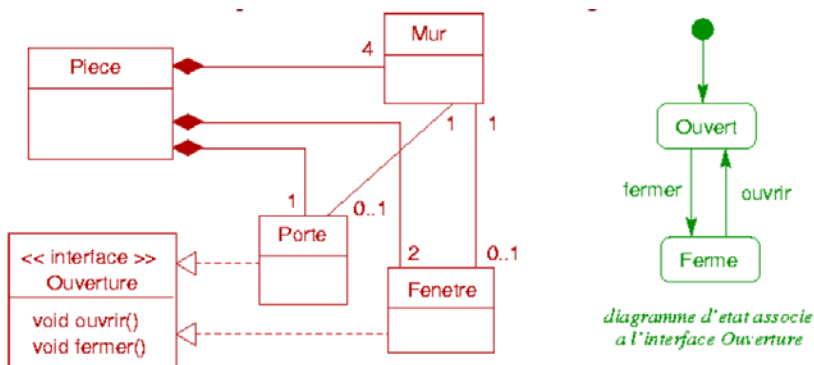
- Un diagramme de classe pour représenter les relations entre les éléments (portes, murs, pièce)
- Un diagramme d'état pour spécifier le comportement d'une porte ou d'une fenêtre (ouverte, fermée)
- On peut abstraire le comportement des portes et des fenêtres en spécifiant les opérations d'ouverture et fermeture dans une interface
 - Le diagramme d'état est associé à cette interface
- Il faut également ajouter des contraintes OCL pour préciser les contraintes entre les éléments d'une pièce

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

M1 : Spécification du Système

5



context Mur inv: fenetre -> union(porte) -> size() <= 1
 -- un mur a soit une fenêtre soit une porte (soit rien)

context Piece inv:

mur.fenetre -> size() = 2 -- 2 murs de la pièce ont une fenêtre

mur.porte -> size() = 1 -- 1 mur de la pièce a une porte

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

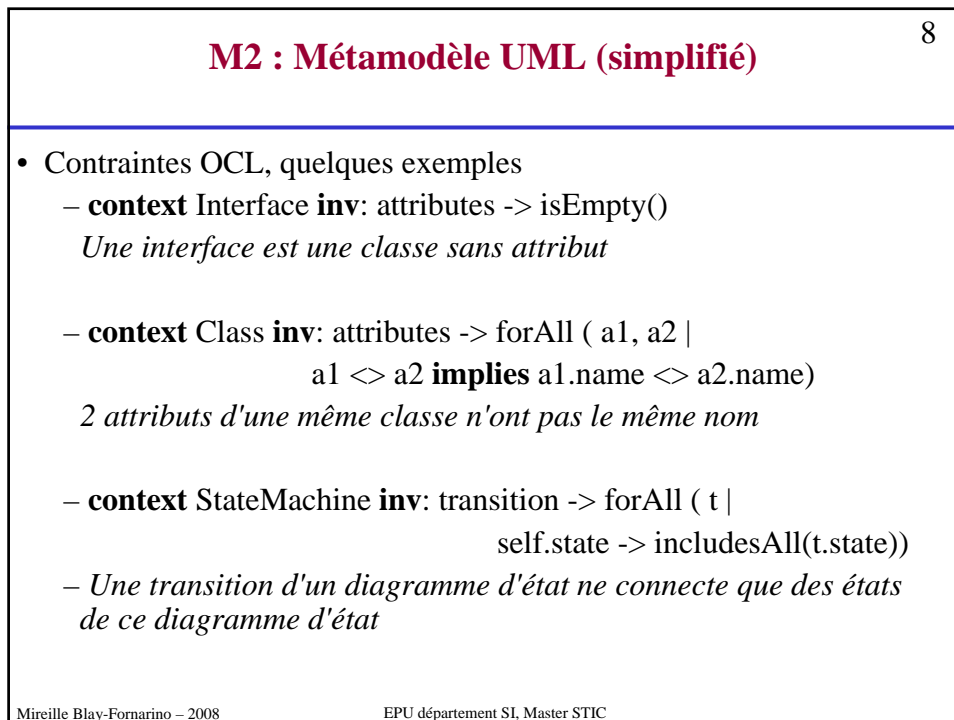
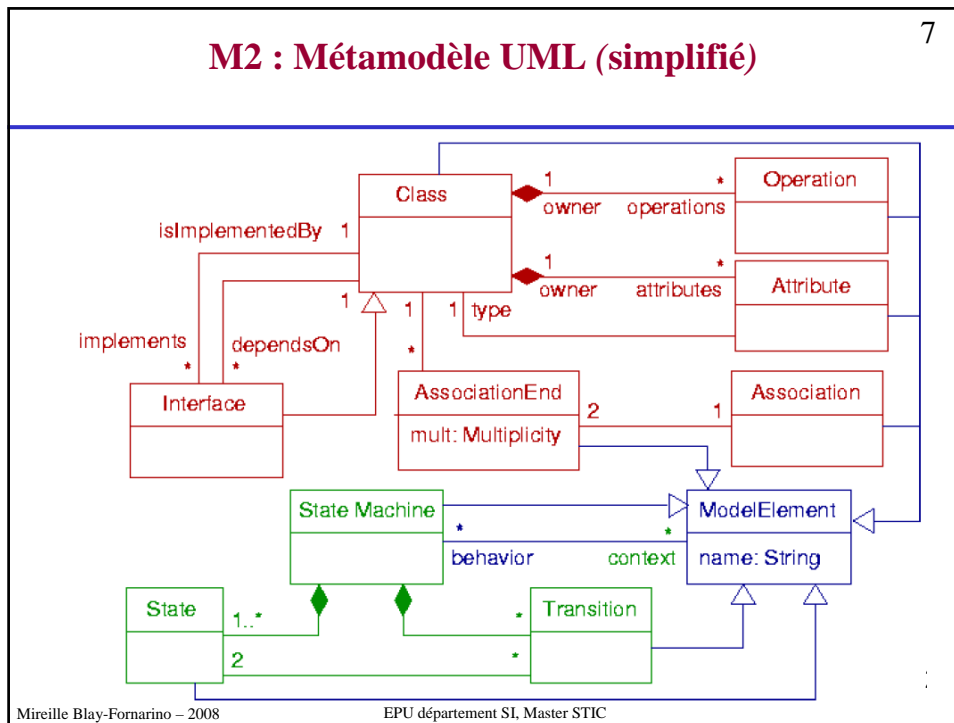
Validité d'un modèle UML?

6

- Les 2 diagrammes de ce modèle de niveau M1 sont des diagrammes UML valides, parce qu'ils sont conformes au métamodèle UML.
- Les contraintes sur les éléments des diagrammes UML et leurs relations sont définies
 - Dans le métamodèle UML : niveau M2
 - Un diagramme UML (de classes, d'états ...) doit être conforme au métamodèle UML
- Métamodèle UML
 - Diagramme de classes spécifiant la structure de tous types de diagrammes UML
 - Avec contraintes OCL pour spécification plus précise

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC



Métamodélisation UML

11

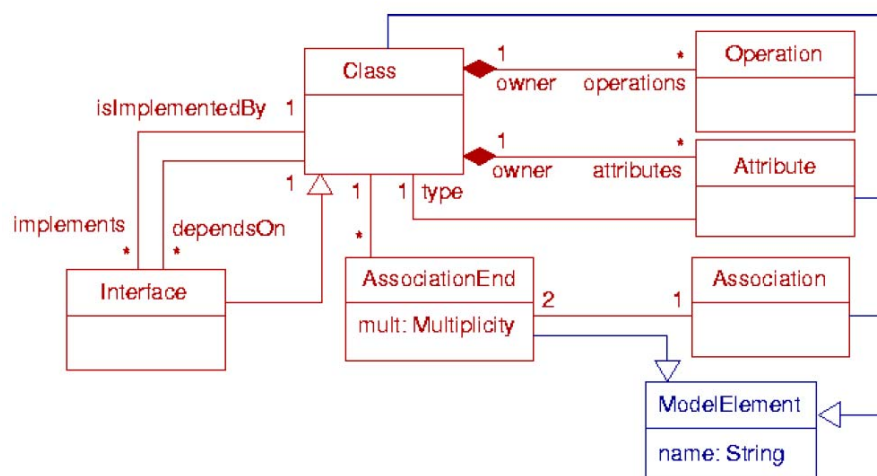
- Qu'est ce que le métamodèle UML ?
 - Un diagramme de classe UML (avec contraintes OCL)
- Comment spécifier les contraintes d'un diagramme de classe UML ?
 - Via le métamodèle UML
 - Ou plus précisément : via la partie du métamodèle UML spécifiant les diagrammes de classes
- Le **métamodèle UML** doit aussi être précisément défini
 - Il doit être conforme à un métamodèle
 - C'est le méta-métamodèle UML
 - Niveau M3
- Méta-métamodèle UML = copie partielle du métamodèle UML

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

M3 : Méta-métamodèle UML (simplifié)

12



Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

Métamodélisation UML

13

- **Méta-métamodèle UML** doit aussi être clairement défini
- Il doit être conforme à un métamodèle
- Qu'est ce que le méta-métamodèle UML ?
 - Un diagramme de classe UML
- Le méta-métamodèle UML peut donc se définir lui même
 - Méta-circulaire
 - Pas besoin de niveau méta supplémentaire

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

Hierarchie de modélisation

14

Niveau M3

conforme à

Niveau M2

conforme à

Niveau M1

conforme à

Niveau M0

conforme à



20

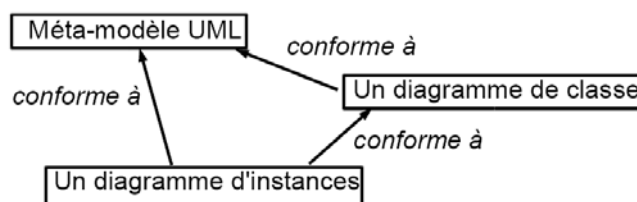
Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

Diagramme d'instances en UML

15

- Un diagramme d'instances est particulier car
 - Doit être conforme au métamodèle UML
 - Qui définit la structure générale des diagrammes d'instances
 - Doit aussi être conforme à un diagramme de classe
 - Diagramme de classe est un métamodèle
 - Qui doit être conforme également au métamodèle UML



Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

Spécification d'un diagramme d'instance en UML

16

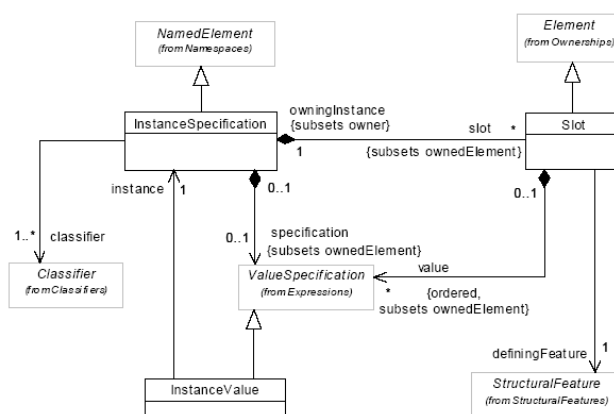


Figure 35, p. 55, infrastructure 2.0

« The defining feature of each slot is a structural feature (directly or inherited) of a classifier of the instance specification. »
 slot->forAll(s | classifier->exists(c | c.allFeatures()->includes(s.definingFeature)))

Mireille Blay-Fornarino – 2008

EPU département SI, Master STIC

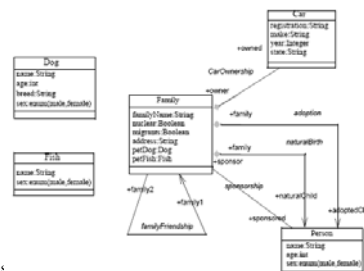
SyntaxeS

17

- Un langage est défini par un métamodèle
- Un langage possède une syntaxe
 - Précise comment représenter chaque élément d'un modèle
 - Elément d'un modèle = instance d'un métaélément
- **Syntaxe textuelle**
 - Ensemble de mots-clés et de mots respectant des contraintes défini selon des règles précises : Notions de syntaxe et de grammaire dans les langages

Family "The McDonalds" {
address: "7 Main Street"
familyFriends: "The Smiths"
petFish: female Fish "Wanda";
petDog: "Spike"
CarOwnership: "755-BDL" {
state: QLD
make: "Mitsubishi Magna"
year: 1992

← HUTN
 Human-Usable
 Textual Notation



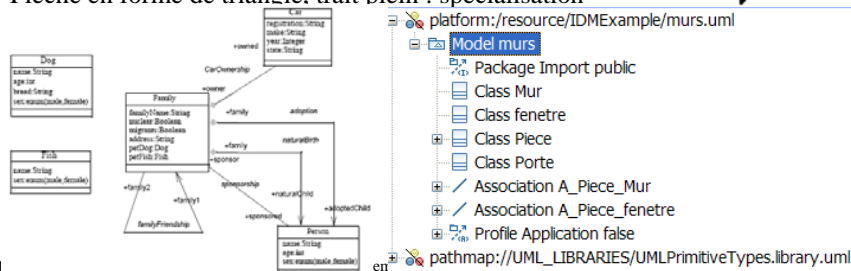
Mireille Blay-Fornarino – 2008

EPU département SI, Ma

Syntaxe graphique

18


- Notation graphique, chaque élément a une forme graphique particulière
- Exemple : associations entre classes/interfaces sur les diagrammes de classe UML
 - Trait normal : association —————
 - Flèche, trait pointillé : dépendance - - - - ->
 - Flèche en forme de triangle, trait en pointillé : implémentation - - - - ->
 - Flèche en forme de triangle, trait plein : spécialisation —————>




Mireille Bl

Syntaxe abstraite/concrète

19



- Abstraite
 - Les éléments et leurs relations sans une notation spécialisée
 - Correspond à ce qui est défini au niveau du métamodèle, à des instances de méta-éléments
- Concrète
 - Syntaxe graphique ou textuelle définie pour un type de modèle
 - Plusieurs syntaxes concrètes possibles pour une même syntaxe abstraite
- Un modèle peut être défini via n'importe quelle syntaxe
 - L'abstraite
 - Une des concrètes
- MOF : langage pour définir des métamodèles
 - Pas de syntaxe concrète définie

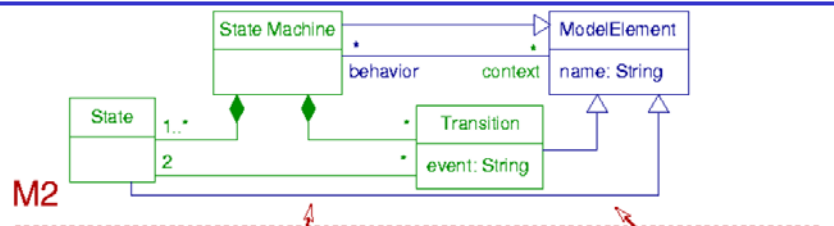


Family: "The McDonalds" (
 address: "7 Main Street"
 familyFriends: "The Smiths"
 petFish: female Fish "Wanda"
 petDog: "Spike"
 CarOwnership: "753-BDL")


Mireille Blay-Fornarino – 2008 EPU département SI, Master STIC

Syntaxe : exemple diagramme état

20



M2



M1

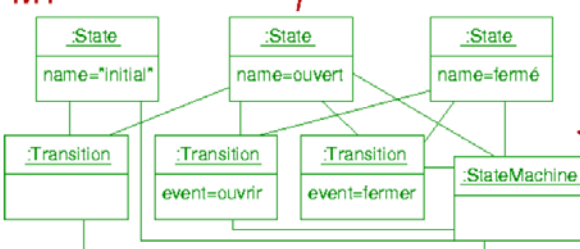
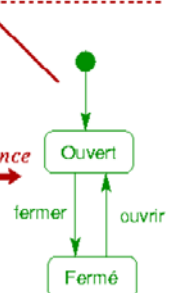


Diagramme défini via syntaxe abstraite

conforme à

conforme à

équivalence



Syntaxe concrète

Mireille Blay-Fornarino – 2008 EPU département SI, Master STIC