Le modèle OFL au service du métaprogrammeur

Application à Java

Adeline Capouillez, Pierre Crescenzo et Philippe Lahire

mercredi 23 janvier 2002
LMO 2002, Montpellier

Plan

- OFL (*Open Flexible Languages*)
  - Contexte et objectifs
  - Architecture
- Définition de Java en OFL
  - Composants-descriptions de Java
  - Composants-relations de Java
- Bilan

OFL : Contexte

- Langages à objets avec classes
  - Java
  - Eiffel
  - C++
- Génie Logiciel
  - Contrôles prépondérants
  - Approche pragmatique

OFL : Objectifs

- Permettre au programmeur de mieux exprimer ses intentions
- Mettre l'accent sur la spécification des relations interclasses
- Ne pas imposer un nouveau langage ou mécanisme

⇨ Décrire la sémantique opérationnelle des langages à objets avec classes au moyen de composants

⇨ Réifier les classes mais aussi leurs relations

⇨ Simplifier autant que possible la création et l'utilisation des composants de langage

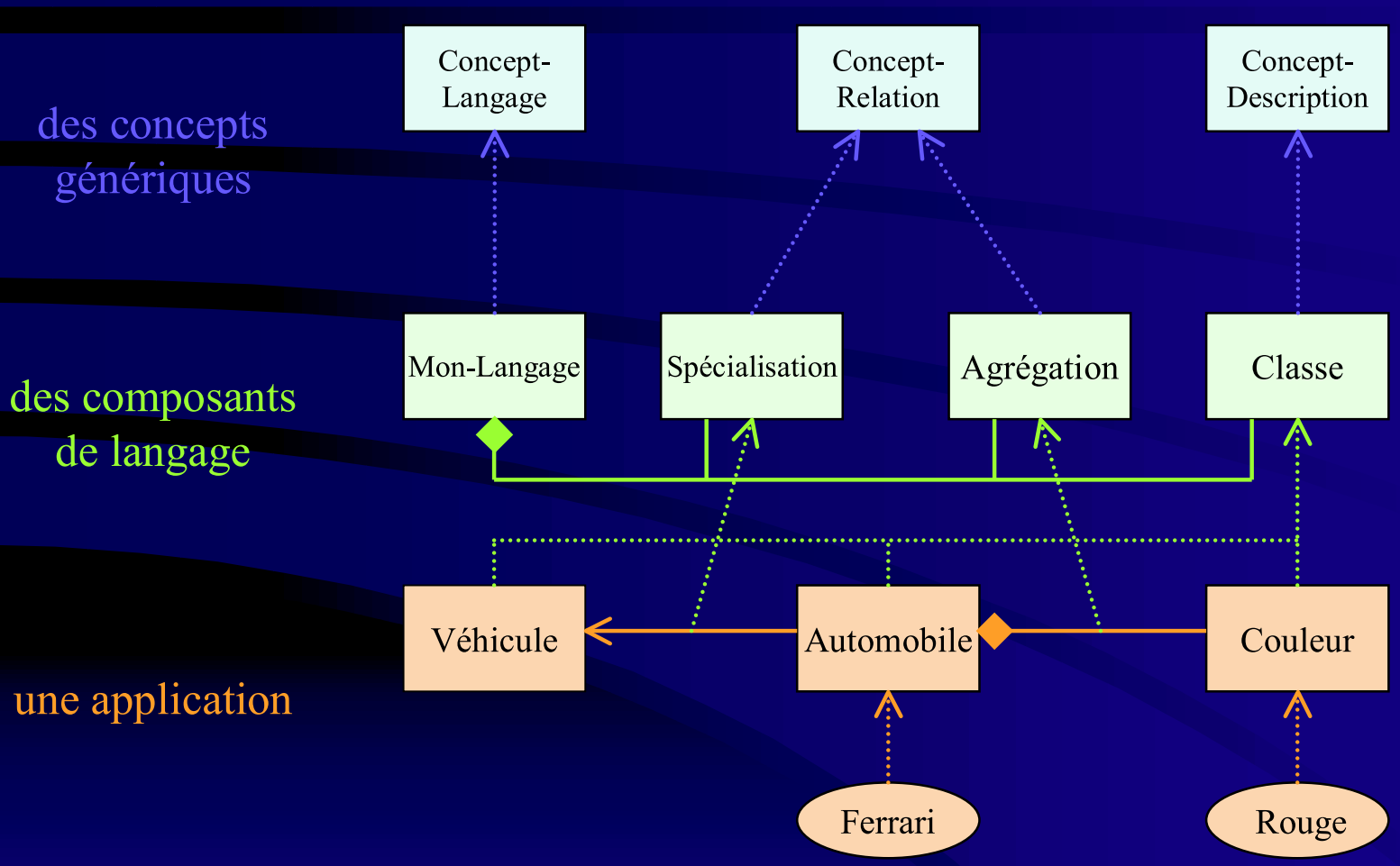
OFL : Paramètres et actions

- Chaque composant est notamment décrit à l'aide d'un ensemble de *paramètres*.
- Paramètre ≈ partie de la sémantique
  - Paramètres de composant-relation : Cardinality, Circularity, ...
  - Paramètres de composant-description : Generator, Overloading, ...
- Les *actions* (lookup, send_message, ...) tiennent compte de la valeur des paramètres pour effectuer la compilation et l'exécution.

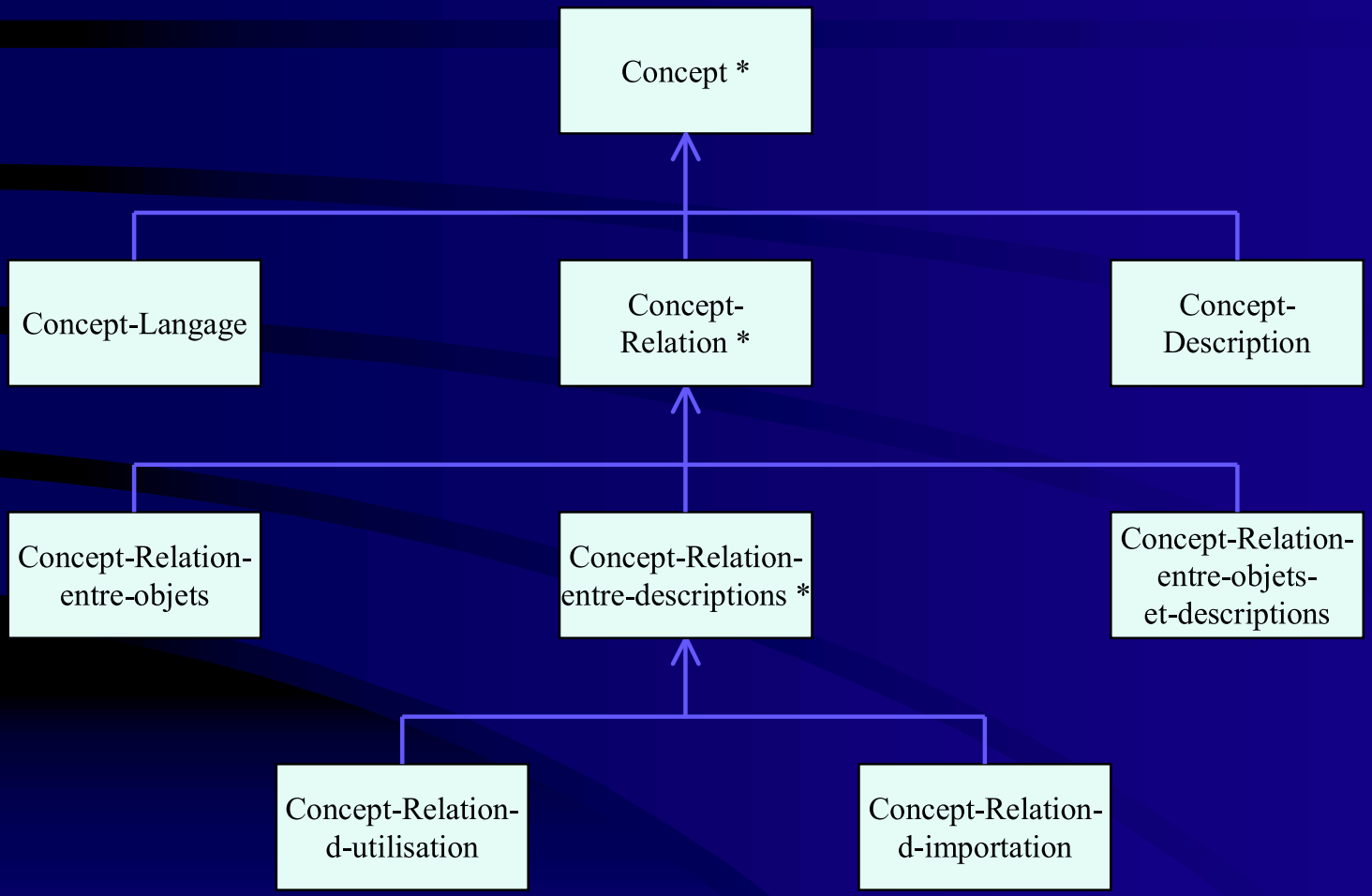
OFL : Composants de base

- Trois éléments essentiels
  - Composant-description
    - ≈ métaclasse paramétrée
  - Composant-relation
    - ≈ métarelation paramétrée
  - Composant-langage
    - ≈ métalangage, composition

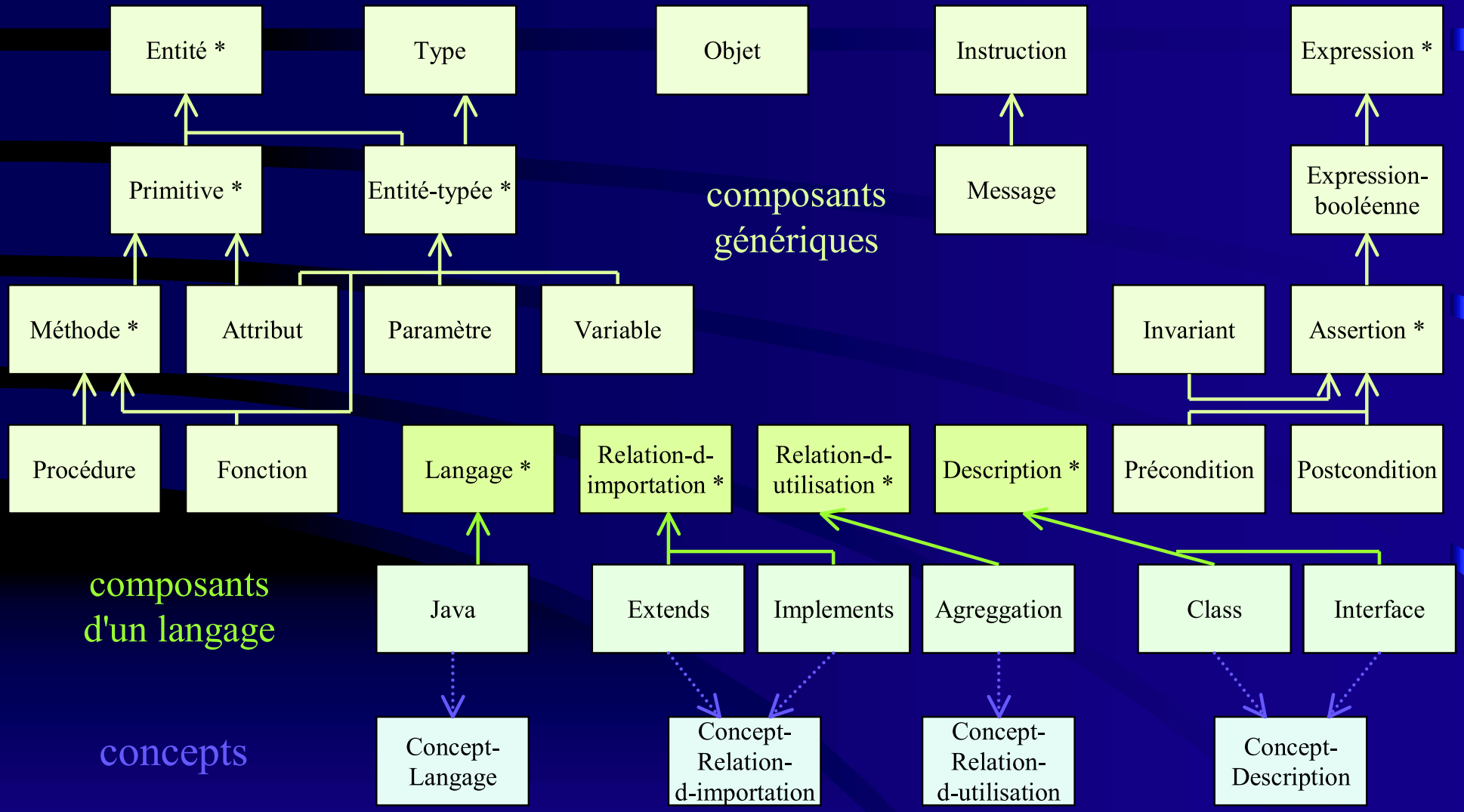
OFL : Architecture



Hiérarchie des concepts



Hiérarchie partielle des composants de base



Définition de Java en OFL

composants-relations
héritage interclasses
héritage interinterfaces
implémentation
concrétisation
agrégation
agrégation de classe
composition
composition de classe

composants-descriptions
classe
classe abstraite
interface
tableau
type primitif
classe membre statique
classe membre
classe locale
classe anonyme
interface membre statique

OFL : les points clés

- La *relation entre description* et non la *description* au centre du modèle
- Système de paramètres et d'actions pour simplifier la tâche du méta-programmeur
- Contrôles assertionnels de cohérence du modèle et de ses usages
- Métaprogrammation et programmation différenciée
- Pas de langage support choisi *a priori*
- Langages décrits sous forme de composants