

Noyaux Multi-cœurs et Virtualisation

Gaël Thomas

gael.thomas@lip6.fr

Université Pierre et Marie Curie
Master Informatique
M2 – Spécialité SAR

Introduction

Nouveaux appareils

- ✓ Informatique embarquée
 - PDAs, téléphone, voiture, satellites, avions, trains, cartes à puce...
 - ☞ Ressources contraintes
 - ☞ Besoins spécifiques : sécurité, économie d'énergie...
- ✓ Architectures multi-cœurs
 - ☞ Augmentation de la vitesse en augmentant le parallélisme
 - ☞ Gestion du placement, des caches

Nouveaux comportements

- ✓ Informatique ubiquitaire
 - ☞ Environnement matériel changeant et hétérogène
- ✓ Informatique nomade
 - ☞ Bureau portable, masquage de la localisation

*Besoin
d'extensibilité*

Introduction

Nouveaux appareils

- ✓ Informatique embarquée
 - PDAs, téléphone, voiture, satellites, avions, trains, cartes à puce...
 - ☞ Ressources contraintes
 - ☞ Besoins spécifiques : sécurité, économie d'énergie...
- ✓ Architectures multi-cœurs
 - ☞ Augmentation de la vitesse en augmentant le parallélisme
 - ☞ Gestion du placement, des caches

Nouveaux comportements

- ✓ Informatique ubiquitaire
 - ☞ Environnement matériel changeant et hétérogène
- ✓ Informatique nomade
 - ☞ Bureau portable, masquage de la localisation

Introduction

Nouveaux appareils

- ✓ Informatique embarquée
 - PDAs, téléphone, voiture, satellites, avions, trains, cartes à puce...
 - ☞ Ressources contraintes
 - ☞ Besoins spécifiques : sécurité, économie d'énergie...
- ✓ Architectures multi-cœurs
 - ☞ Augmentation de la vitesse en augmentant le parallélisme
 - ☞ Gestion du placement, des caches

Nouveaux comportements

- ✓ Informatique ubiquitaire
 - ☞ Environnement matériel changeant et hétérogène
- ✓ Informatique nomade
 - ☞ Bureau portable, masquage de la localisation

*Besoin
de virtualisation*

Introduction

Nouveaux appareils

- ✓ Informatique embarquée
 - PDAs, téléphone, voiture, satellites, avions, trains, cartes à puce...
 - ☞ Ressources contraintes
 - ☞ Besoins spécifiques : sécurité, économie d'énergie...
 - ✓ Architectures multi-cœurs
 - ☞ Augmentation de la vitesse en augmentant le parallélisme
 - ☞ Gestion du placement, des caches
- Besoin de concurrence*

Nouveaux comportements

- ✓ Informatique ubiquitaire
 - ☞ Environnement matériel changeant et hétérogène
- ✓ Informatique nomade
 - ☞ Bureau portable, masquage de la localisation

21/09/10

Architectures à Composants

5

Plan du cours

Étude de la virtualisation (2 semaines)

- ✓ Les moniteurs de machines virtuelles
- ✓ Les machines virtuelles applicatives
- ✓ Les ramasse-miettes

Étude de l'extensibilité dans le noyau Linux (2,5 semaines)

- ✓ Architecture du noyau Linux
- ✓ Construction de modules

Étude des multi-cœurs (2,5 semaines)

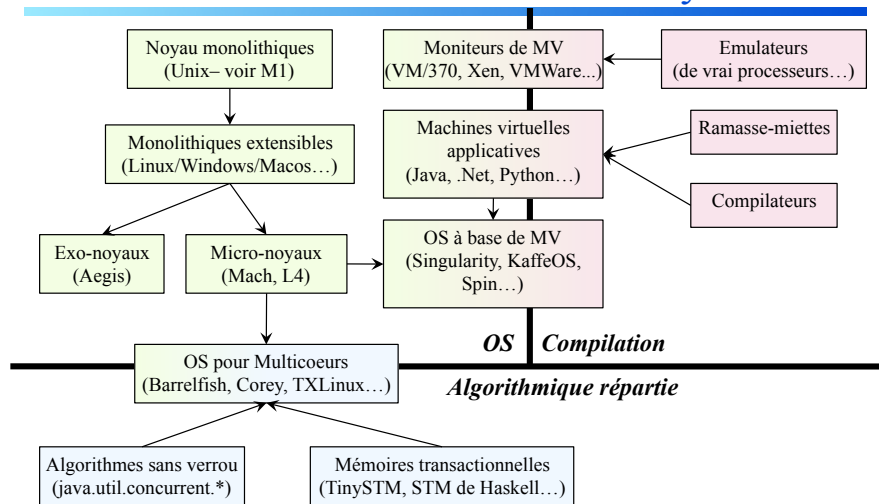
- ✓ Architectures et contraintes des multi-cœurs
- ✓ Concurrency en mémoire partagée
- ✓ Algorithmes classiques sans verrous
- ✓ Mémoires transactionnelles logicielles

21/09/10

Architectures à Composants

6

Vue d'ensemble de l'évolution des systèmes

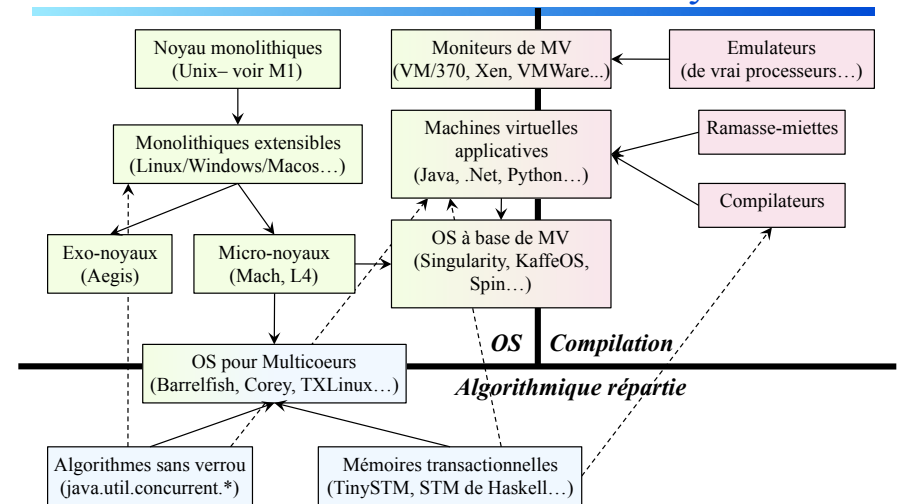


21/09/10

Architectures à Composants

7

Vue d'ensemble de l'évolution des systèmes



21/09/10

Architectures à Composants

8

Modalité d'examens

Un examen final sur 70%

Une note de contrôle continu sur des mini-projets 30%

- ✓ Réalisation d'un module Linux

Bibliographie

Machines virtuelles et ramasse-miettes

- ✓ Garbage Collection: algorithms for automatic dynamic memory management. R. Jones, R. Lins. Wiley (1996).
- ✓ Virtualisation logicielle : de la machine réelle à la machine virtuelle abstraite. B. Folliot, G. Thomas. *Techniques de l'Ingénieur*. Hermès (2009)

Noyau Linux

- ✓ Le noyau linux. D. P. Bovet, M. Casata, J. Cornavi Oreilly (2001)

Programmation multi-coeurs

- ✓ The Art Of Multiprocessor Programming. M. Herlihy, N. Shavit. Elsevier (2008)

Divers

- ✓ Bibliographie du module Noyau (M1)
- ✓ Conférences et revues en système : SOSP, OSDI, ASPLOS, USENIX, EUROSYS, HOTOS, VEE, CACM, JACM, Computing Surveys, TOPLAS, SP&E, JPDC