

Systemes concurrents – Conclusion

Philippe Quéinnec

14 septembre 2012



Programmation parallèle

- souvent utile
- parfois indispensable
- fragile et complexe
- souvent difficile
- amusant

Deux aspects

- le parallélisme
- la synchronisation



Approches traditionnelles

- création explicite d'activités
- synchronisation explicite
- mécanismes classiques (verrou d'exclusion mutuelle, sémaphore, moniteur)
- raisonnement connues
- schémas classiques (producteurs/consommateurs, lecteurs/rédacteurs)

Exemples : Java Thread, C POSIX Threads



Approches modernes

- création implicite d'activités
- synchronisation implicite
- schémas classiques (fork-join)
- encore peu développées
- ne résolvent pas tous les problèmes

Exemple : OpenMP



Approches d'avenir

- création implicite et explicite d'activités
- synchronisation implicite
- mémoire transactionnelle
- bibliothèque de structures de données non bloquantes
- à voir dans 5 ou 10 ans. . .

