

# Applications orientées données (NSY135)

## 14 – Applications concurrentes

Auteurs: Raphaël Fournier-S'niehotta et Philippe Rigaux  
([philippe.rigaux@cnam.fr](mailto:philippe.rigaux@cnam.fr), [fournier@cnam.fr](mailto:fournier@cnam.fr))

Département d'informatique  
Conservatoire National des Arts & Métiers, Paris, France

## Introduction

- La notion de transaction est classique dans le domaine des bases de données
- Tous les SGBD implantent un **système de contrôle de concurrence**
- Hibernate ne ré-implante pas un protocole de contrôle transactionnel qui serait redondant avec celui du SGBD.
- En première approche, une application Hibernate est donc simplement une application standard qui communique avec la base de données
- Elle est éventuellement en concurrence avec d'autres applications, et lui soumet des transactions.
- Hibernate mérite un développement spécifique pour le traitement de transactions "longues", particulièrement utiles dans le cadre d'une application Web.

# Plan du cours

- 1 Rappels sur la concurrence d'accès

## Rappels sur la concurrence d'accès

- Lire attentivement le chapitre du cours BDD
- Comprendre les différentes caractéristiques de la concurrence d'accès et des niveaux d'isolation
- Faire l'exercice

## Exercice

### Simuler des transactions en ligne de commande

- Étendre la base Films pour pouvoir :
  - ajouter des séances diffusant les films
  - autoriser les internautes à réserver des places pour les séances
- Les tables \*Seance\* et \*Reservation\* existent déjà dans la base. Faites le \*mapping\* JPA pour ces deux entités.
- Pour l'instant, on n'utilise pas Hibernate, mais deux terminaux connectés à MySQL, représentant deux sessions concurrentes.
- La commande "prompt" sous l'interpréteur MySQL permet de caractériser chaque session.

## Exercice

- Reproduisez les situations suivantes :
  - en mode "read uncommitted": lecture sale, menant une des transactions à connaître une mise à jour de l'autre transaction, alors que cette dernière fait un "rollback";
  - en mode "read committed": des mises à jour perdues, résultant en un état de la base incohérent;
  - faire la même exécution concurrente en "repeatable read": peut-on s'attendre à un changement?
  - en mode "serializable": un interblocage.