

Bases de données

Cours L3 Informatique

Pierre Senellart Leonid Libkin



15 janvier 2020

Bases de données

- Systèmes de gestion de bases de données (SGBD):
composants présents dans un grand nombre d'applications

Bases de données

- Systèmes de gestion de bases de données (SGBD): composants présents dans un grand nombre d'applications
- Industrie gigantesque (Oracle, Microsoft, IBM, etc.)

Bases de données

- Systèmes de gestion de bases de données (SGBD): composants présents dans un grand nombre d'applications
- Industrie gigantesque (Oracle, Microsoft, IBM, etc.)
- Raison principale de ce succès: déléguer au SGBD tout l'aspect stockage, intégrité des données, efficacité de l'accès aux données, reprise sur panne, concurrence, pour se focaliser sur la logique de l'application

Bases de données

- Systèmes de gestion de bases de données (SGBD): composants présents dans un grand nombre d'applications
- Industrie gigantesque (Oracle, Microsoft, IBM, etc.)
- Raison principale de ce succès: déléguer au SGBD tout l'aspect stockage, intégrité des données, efficacité de l'accès aux données, reprise sur panne, concurrence, pour se focaliser sur la logique de l'application
- À la base: théorème de Codd (années 1970) exprimant l'équivalence entre formalismes logiques (accès aux données de manière déclarative) et algèbre (opérateurs implémentables d'accès aux données)

Bases de données

- Systèmes de gestion de bases de données (**SGBD**): composants présents dans un grand nombre d'applications
- Industrie **gigantesque** (Oracle, Microsoft, IBM, etc.)
- Raison principale de ce **succès**: déléguer au SGBD tout l'aspect stockage, intégrité des données, efficacité de l'accès aux données, reprise sur panne, concurrence, pour se focaliser sur la logique de l'application
- **À la base**: **théorème de Codd** (années 1970) exprimant l'équivalence entre formalismes logiques (accès aux données de manière déclarative) et algèbre (opérateurs implémentables d'accès aux données)
- **Positionnement de ce cours**: entre **pratique** (SQL, indexation, transactions...) et **théorie** (logique, complexité algorithmique, théorie des modèles finis...)

Programme du cours (1/2)

- 12/02 Introduction à la gestion de données, le modèle relationnel, l'algèbre relationnelle, SQL
- 26/02 Conception de schémas, normalisation
- 04/03 Vues virtuelles et vues matérialisées, maintenance de vues, mises à jour de vues
- 11/03 Aspects logiques : calcul relationnel, indépendance du domaine, théorème de Codd ([Leonid Libkin](#))
- 18/03 Récursion, Datalog, langages à point fixe, logique du second-ordre ([Leonid Libkin](#))
- 25/03 Complexité computationnelle des langages de requêtes ([Leonid Libkin](#))

Programme du cours (2/2)

- 01/04 Analyse statique: minimisation de requêtes, requêtes acycliques, réécriture de requêtes
- 22/04 Contraintes sur les données : dépendances fonctionnelles, dépendances d'inclusion, poursuite
- 29/04 Optimisation de requêtes : génération de plans, modèles de coût
- 06/05 Indexation et stockage
- 13/05 Transactions et gestion de concurrence: sérialisabilité, verrouillage à deux phases, estampillage
- 20/05 Applications Web et sécurité
- 27/05 Examen

Organisation pratique

- Cours commun avec l'ENS Paris-Saclay, à Ulm
- Mercredi de 8h30 à 10h15: cours en salle U/V
- Mercredi de 10h30 à 12h15: TD/TP (en salles Info3 et Info4) avec Nathan Grosshans et Michaël Thomazo
- Évaluation:
 - Par moitié sur les TP/TD (rendu demandé)
 - Par moitié par un examen

Bibliographie



Serge Abiteboul, Richard Hull, and Victor Vianu.
Foundations of Databases.
Addison-Wesley, 1995.



Michael Benedikt and Pierre Senellart.
Databases.
In Edward K. Blum and Alfred V. Aho, editors, *Computer Science. The Hardware, Software and Heart of It*, pages 169–229. Springer-Verlag, 2012.



Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom.
Database Systems: The Complete Book.
Pearson, 2008.