UNIVERSITE LIBRE DE BRUXELLES
Ecole Polytechnique

September 2013

INFO-H-417 : Database System Architecture Course Information

The objectives in brief

In contrast to a typical introductory course in database systems where one learns to design and query relational databases, the goal of this course is to get a fundamental insight into the implementation aspects of database systems. In particular, we take a look under the hood of relational database management systems, with a focus on query and transaction processing.

With respect to query processing, we study the whole workflow of how a typical relational database management system optimizes and executes SQL queries. By having an in-depth understanding of the query-optimization-and-execution pipeline, one becomes more proficient in administering DBMSs, and hand-optimizing SQL queries for fast execution.

With respect to transaction processing we study how a typical relational database management systems ensures recovery from errors and controls concurrent access to the data.

Contacts

– Lecturer : Stijn Vansummeren (UB4.125, `stijn.vansummeren@ulb.ac.be`)
– Assistant : Stefan Eppe (UB4.131, `stefan.eppe@ulb.ac.be`)
– Course Web Page : `http://cs.ulb.ac.be/public/teaching/infoh417`

Schedule

Please check the course online calendar available on the course website for room information.

Date	Time		Content
Fri 20 Sep	10h–13h	Lect. 1	Course introduction Reminder of the operations of the Relational Algebra Translation of SQL into the Relational Algebra
Tue 24 Sep	10h–13h	Ex. 1 Lect. 2	Exercises on SQL-to-Relational-Algebra translation Logical Query Optimization
Fri 4 Oct	10h–13h	Ex. 2 R.A. 1	Exercises on Logical Query Optimization Reading assignment : Physical Data Organization
Fri 11 Oct	10h–13h	Lect. 3	Index Structures : Classical index structures + BTrees
Fri 18 Oct	10h–12h	Lect. 4	Index Structures : Hashing Introduction to DBMS architecture + Project assignment
	13h-15h	Ex. 3	Computer lab on Physical data organization
Fri 25 Oct	10h–12h	Lect 5	Multidimensional index structures
	13h-15h	Ex .4	Computer lab on Physical data organization
Fri 8 Nov	10h–13h	Lect 6	Physical Operators
Fri 15 Nov	10h–13h	Ex 5. Lect 7	Exercises on phyiscal operators Cost-based plan selection
Fri 22 Nov	10h–13h	Ex. 6 Demo 1 Lect. 8	Exercises on cost-based plan selection Demo : cost-based plan selection in practice Coping with system failures
Fri 29 Nov	10h–13h	Ex . 7 Lect. 9	Integrated exercises Transaction processing : concurrency control
Fri 6 Dec	10h–13h	Ex. 8 Demo 2 Q&A	Exercises on system failures & concurrency control Demo 2 : Transaction processing in practice Question & Answer session
Fri 13 Dec	10h–13h	Q&A Ex. 9	Question & answer session Overview exercises

Detailed Objectives

Upon successful completion of this course, the student should master the following competences :

1. Translating a given SQL expression into the Relational Algebra
2. Improving a relational algebra expression by, where possible, removing redundant joins in select-project-join subexpressions
3. Improving a relational algebra expression by, where possible, (a) replacing cartesian products by joins ; and (b) pushing selections and projections
4. Describing and being able to implement traditional secondary-memory index structures (BTrees, Hashing)
5. Being able to describe and demonstrate the shortcomings of traditional index structures with respect to multi-dimensional search keys. In addition, explaining the studied multi-dimensional indexes by means of an example
6. Describing the most important implementation algorithms (one-pass, sorting, hashing, index) for each of the relational algebra operators, as well as judging the cost of each operator, and knowing their limitations of applicability
7. Given a logical query plan and given base statistics about the size and distributions of the database relations, constructing a heuristically optimal physical query plan, by estimating the sizes of the intermediate results and correspondingly comparing the possible implementations. When joins can be reordered, choosing the order with the least cost.
8. Solving exercises on logging like the ones in sections 17.2.6, 17.3.5 and 17.4.4
9. Solving exercises on concurrency control like exercises 18.2.4, 18.2.5, 18.8.1, 18.8.2, and 18.9.1
10. Solving exercises on recoverability like exercises 19.1.1, 19.1.2 and 19.1.3
11. Being able to reconstruct the studied proofs

Method of evaluation

The project work contributes 6/20 points to the overall score, and the written exam contributes the remaining 14/20 points. Participation in both the project work and the written exam are mandatory requirements for passing the course.