

CS4221 Team Project

No. of students per project team: 4 students

Deadlines:

Proposals due: 5 Feb 2015 (Thurs of week 4)

Project submissions due: 26 Mar 2015 (Thurs of week 10)

Presentations and Demos if any:

April 2, 9, 16 (Thurs of weeks 11, 12, and 13)

Late Penalty for all submissions 3 marks a day

Details can be found on:

<http://www.comp.nus.edu.sg/~lingtw/cs4221/project.html>

Project proposal

- Each project team needs to submit a short proposal (softcopy) which includes the following information by 5 Feb 2015 (Thurs of week 4) to dcsltw@nus.edu.sg for approval:
 1. A **project title**.
 2. Project **team members'** names and matriculation numbers.
 3. **Type of the project**, i.e. implementation, survey and comparison, or preliminary research study.
 4. An **abstract** which briefly describes the project with some references if any.

3 Types of Projects

- **Implementation/Programming**

- The deliverables: a 15 page report (double spacing and 12 point font, hardcopy) + code (in CD-ROM) + PowerPoint presentation slides (softcopy) + a 20 minute presentation and demo.

- **Survey and Comparison**

- The deliverables: a 15 to 20 page report (double spacing and 12 point font, hardcopy) + PowerPoint presentation slides (softcopy) + a 20 minute presentation.

- **Preliminary Research Study**

- The deliverables: a 15 page report (double spacing and 12 point font, hardcopy) + code (if any, in CD-ROM) + PowerPoint presentation slides (softcopy) + a 20 minute presentation.

Note: Project report submission. Please also include your project number in your email Subject, project report, and ppt file.

For implementation projects, please send the code (in CD-ROM) to Ling Tok Wang (room COM2 03-01).

Finding a Project Topic

There are many topics that you may choose, such as:

- **CASE tool** implementation
 - Relational Database Schema Designer using Bernstein's Algorithm
 - Good interface for input and editing FDs
 - Output results of the intermediate steps with brief explanation
 - Find closure of attributes
 - Normal form tester
 - Remove the shortcomings of Bernstein's algorithm
- **Schema translation** between different data models
 - ERD to RDB for database schema design tool
 - ERD editor with good user interface
 - Translating ERD to RDB schema
 - Normal form ERD generator
 - RDB to ERD or OODB

- **Semantics discovery** in databases RDB, XML, etc.
 - Discover ORA-semantics (object, relationship, attribute of object and relationship) in databases
- **Data and schema integration**
- **View updates**
 - How to use database views to update the database
- **Materialized view maintenance**
 - Views are materialized and stored physically
 - What are the impact of database updates to the materialized views
- **XML twig pattern query processing**
- **XML keyword query processing** with/without ORA-semantics
- **RDB keyword query processing** with/without ORA-semantics

Projects suggested by Professor Stephane Bressan

- 1) Implement a pre-processing tool that compiles general SQL **integrity constraints** into **triggers**.
 - The triggers enforce the constraints using propagation or using a generalized assertion verification of the CHECK type. Choose the target system and argue that its Trigger sublanguage has the features needed for a valid implementation.
- 2) Implement a **Universal Relation user interface** with window functions (Ullman Vol. 2 Chapter 17) on top of a commercial DBMS. You may implement System/U language or (preferred) an interactive graphical user interface (QBE-like).

3) Dataset generator. Implement a realistic generic random instance generator (with interactive interface). The generator generates random but very realistic data (names, countries, currencies amounts etc.) given a schema, database and other (real world) constraints and distributions. For example (simple), generate a list of persons: the persons should have realistic Chinese names, realistic Chinese telephone numbers and addresses (that match) and should be located in cities proportionally to the actual population sizes. The more constraints are taken into account, the better.