# Database Tuning, ITU, Spring 2007

Rasmus Pagh

May 1, 2007

## Project – 5th part and full report

**Deadline: May 23, 15.00 Danish time. Hand in at the exam office in *three* copies.**

## Purpose

The purpose of the final part of the project is to allow students to pursue their own interests, within the scope of the project. While the "core" project is supposed to be done by all group members together, extensions may be pursued by any subset of the group (e.g., by single group members), based on differences in interests or ambition.

The full report documents each group's work on the project, and the ability to communicate clearly on complicated technical matters. It also documents the ability to independently formulate a small research project (a possible thesis project) within the subject area. In addition, it forms the basis for the examination and grade.

## Suggested topics

Below you can find some inspiration for topics to address in the last part of the project. Feel free to modify or extend the suggestions, or come up with your own.

**Sampling.** In the course you have seen several uses of sampling: Query result size estimation, "on-line aggregation", and producing early join results. Use the `SAMPLE` operator of Oracle to experiment with sampling. Can sampling be used to improve, in some sense, any of the queries studied in the project? How do sampling-based estimates of query result sizes compare with the estimates made by the Oracle query optimizer, and with the actual sizes? Is it efficient to maintain a materialized view that is a sample of a relation?

**Column-based databases.** Some people believe that DBMSs that store relations by column, rather than by row, have a great future. One instantiation of this idea is the "TransRelational™" data model. The latest edition of Chris Date's textbook on database systems contains an enthusiastic overview of this way of organizing relational data. Read Date's overview (available upon request) and criticize it. Another possibility is to emulate a column-based database in a conventional row-based database by splitting up a relation with $k$ attributes into $k$ relations with (rownum,attribute) pairs.

**OLAP.** Come up with some OLAP-like queries in the context of the project, and try to implement them such that response time is small. You may use materialized views and "on-line aggregation". While bitmap indexes are not explicitly supported in Oracle XE, it is possible to use a normal (B-tree) index to achieve a similar effect. The idea is to build a materialized view where each tuple contains a small part of the bitmap. To access the bitmap, use the `bitand(x,y)` function which takes two binary integers, and returns the integer resulting from a bit-wise conjunction of the bit strings.

**Spatial indexing.** Try out some spatial indexing technique for the "nearest point" query on the geographical data in the VXL database. At least two indexing techniques can be tried, by reduction to a one-dimensional query: Space-filling curves (Z-ordering is the easiest) and grid files.

**Star joins.** Read and understand the algorithm in [PP06, sec. 3.2.1], and implement it *as a sequence of SQL operations*. Can you devise relations and data such that this method beats the standard join algorithm used by the DBMS?

## The full report

The report should be written as if the reader is a student who has followed the lectures of DBT, but not worked on the project. If it seems relevant, the report may summarize parts of the material covered by the DBT lectures, but it is the *application* of this knowledge, with accompanying analysis and arguments, that is the main focus.

The full report should contain:

- An introductory section describing the relevant prerequisites (courses and projects) of each group member.

- The same information as in answers to the four deliverables. You may, and probably should, revise this material according to the feedback received. It is perfectly fine to arrange the material differently than according to deliverable number and question number.

- You are encouraged to extend your discussion to aspects that seem relevant for the VXL case, but not touched upon by the deliverables. For example, if there are cases where acceptable performance could not be achieved, you may discuss what could be done to alleviate this, including restrictions on what (or when) queries are performed, hardware enhancements, etc. If you need more specific knowledge than what is available, make reasonable assumptions.

- An experience report: What aspects of the work were harder (or easier) than expected? Was this because of special restrictions (or features) of the DBMS software, hardware limitations, or other constraints? Was there any information that you would have liked to have at the beginning of the project? Etc.

- Documentation of the programming performed: Printouts of Java programs, DDL statements, and other relevant files. These should be easily understandable in the context of the report (e.g., through a combination of in-code comments and overview sections in the report).

- A proposal for at least one thesis project related to performance of databases. You are supposed to independently formulate a small "research problem" that would be interesting to pursue, based on the knowledge you have obtained in DBT. Also, you should outline a possible methodology using which the problem could be addressed. You are *not* expected to investigate the literature on the subject to find out if the problem has been addressed before.

If some section of the report (and the programming work behind it) was done by a proper subset of the group members, this should be indicated below the section heading. At the exam, you will be expected to be able to discuss any section on the "core" part of the project, whereas a section on an extension will not be discussed with students who did not contribute to it.

Of course, if you have used (even for inspiration only) any papers or books when doing the report, these should be appropriately cited. You are strongly encouraged to upload the report into my.itu. However, remember that you must hand in paper copies to the exam office before the deadline.