Homework 5

SQL using mSQL

Out November 20, Due November 29

1 mSQL

mSQL is a lightweight database engine designed to provide fast access to stored data with low memory requirements. Although mSQL only supports a subset of SQL (no views, sub-queries etc.), everything it supports is in accordance with the ANSI SQL specification. The mSQL package includes the database engine, a terminal "monitor" program, a database administration program, a schema viewer, and a C language API. The API and the database engine have been designed to work in a client/server environment over a TCP/IP network. You have been provided with a user's manual and additional information is available from the CMSC661 web page.

The mSQL system has been installed on the SGIs in the `cs.umbc.edu` domain and the server should always be running on `retriever.cs.umbc.edu`. If it does not appear to be running, please send me mail right away. We have also installed the client programs on the `gl.umbc.edu` domain. These can access the server running on retriever if you use the `-h retriever.cs.umbc.edu` argument when you invoke them.

We have created a database for you to use with your userid as its name – try running the `relshow` command to verify that it is there. Only you have read or write privileges for this database. If you are unable to access it, please let us know as soon as possible (e.g., send email to finin@umbc.edu). You should use this database to represent all of the tables in this assignment. If you are interested in looking at the source code for mSQL or in accessing the library functions, see `/usr/cs/src/msql/` on any of the SGI's in the CS domain (e.g., retriever.cs.umbc.edu).

2 The suppliers, parts and shipments database

Create the *suppliers-parts-projects* tables in your database. You can extract the data from either the rdb version (spj.rdb) or the Prolog version (spj.pl) which were used in previous homeworks. Write SQL queries to answer the following queries against the *suppliers-parts-projects* tables you created. If a query can not be done with mSQL say so and explain what limitations prevent you from doing it in mSQL.

1. Get the supplier names for all suppliers who supply project j1.
2. Find triples of suppliers, projects and parts such that all three are located in the same city.
3. Get the names of parts supplied by a supplier in London.
4. Get project numbers for projects using at least one part available from supplier s1.
5. Get the supplier names of suppliers who supply no parts.

6. Get project numbers for projects not supplied with any red parts by any London suppliers.
7. Get supplier numbers for suppliers who supply the same part to all projects.
8. Get part numbers for parts that are supplied either by a London supplier or to a London project.
9. What is the average weight of a screw?
10. Which part is the heaviest?
11. Show the names of suppliers who supply all projects.

3 Evaluating mSQL

Write a short but comprehensive evaluation of mSQL. By short, we mean three or four pages of single-spaced formatted text. Feel free to write more if you are motivated. If you do a good job, you could submit it for publication in any of several database magazines or newsletters. By comprehensive, we mean that you should address at least the following issues: what features of SQL are and are not supported, which unsupported SQL features (if any) are serious limitations, how good is the interactive query interface, what API libraries are available and how good are they, what features of a serious relational DBMS are missing, how portable is the mSQL package, is it easy to install and maintain, what are alternatives to mSQL and how does it compare to them.

Whenever you write something, you should have in mind who your audience is. Please add a note which describes who you are writing this for (and don't say Professor Finin). For example, you might say "I am writing this as a short article for a technical magazine aimed at practicing database professional" or "This was written as a short review to be submitted to the ACM SigMOD newsletter and is aimed at academics who might consider using mSQL for instruction".

You should base your evaluation on a number sources, including: your own experience using mSQL, an examination of the software package (available on `retriever.cs.umbc.edu` in `/usr/cs/src/msql`), information on SQL standards such as *SQL Standards Home Page* at `http://www.jcc.com/sql_stnd.html` and the NIST page on SQL at `http://waltz.ncsl.nist.gov/~len/sql_info.html` and information on other "free" database systems, such as the *ACM SIGMOD Index of publicly available database software* at `http://bunny.cs.uiuc.edu/databaseSoftware/` and the *Catalog of Free Database Systems* at `http://www.cis.ohio-state.edu/hypertext/faq/usenet/free-databases/faq.html`. Many of these URLs are accessible via the CMSC661 page.