Spring, 2005 CIS 650

Implementing Data Management Systems

Term Project

For your semester project in CIS 650, you are expected to work on implementing a component or feature in a database or data management system. This may generally involve one of several options:

1. Taking an idea – either something implemented in one of the papers we read, or an extension to some idea in one of these papers – and coding and evaluating it as a self-contained unit. An example would be to build a B+ Tree manager, an XPath matcher, a distributed implementation of eddies, a query reformulation algorithm, etc.

2. Extending an existing system with some new capability that is either novel or fairly complex to implement. An example would be to take the Tukwila data integration system and add support for local storage; or to take MySQL and add support for bitmapped indices or dynamic programming optimizer.

3. Building a simple "proof of concept" of a system, with one or more novel aspects. An example would be a P2P query processing engine along the lines of PIER, a distributed CVS system, an efficient search engine (i.e., more efficient than building over an RDBMS), an update propagation system, etc.

In addition to the implementation, you are expected to write a 5-10 page paper about the project. This paper should be in the format of a research paper, and describe the problem you're attempting to solve, your goals and methodologies, the basic structure of your solution, an evaluation of your project, and some ideas for future development. The project and report will be due by the time of our officially scheduled final exam.

Here are a few ideas for projects that have a bit of research novelty but also (so far as I can judge) should be possible to implement in a semester:

1. Adding a System-R (or Starburst or Volcano) query optimizer to MySQL. See http://delivery.acm.org/10.1145/590000/583898/p49-badia.pdf?key1=583898&key2=4998637011&coll=GUIDE&dl=GUIDE&CFID=37836996&CFTOKEN=13638324 for some details on a group's experiences in trying to add features to MySQL. This is probably a 2-person project.

2. A P2P query engine built over Pastry, Chord, CAN, or some other distributed hash table system. See the PIER paper for some ideas of how this is done.

3. A distributed Eddies implementation. The eddies paper provides a queue-based approach to query scheduling. Consider how this might fit into a distributed context. See also http://db.cs.pitt.edu/courses/cs3550/04-2/papers/wisc-eddies.vldb03.pdf.

4. A simple schema matching tool based on the approach of LSD.

5. Adding support to the Tukwila distributed query engine for local storage using the BerkeleyDB storage libraries. The main challenge here is implementing an index scan operator and adding support into the query optimizer.

6. Building an efficient, larger-than-memory indexer and information retrieval engine for local storage, patterned along the model of Google or Yahoo Desktop Search.

7. Building P2P distributed catalog and maintenance features for the Piazza peer-to-peer data management system, which provides support for mappings between arbitrary schemas and data sources. This project might pair up nicely with someone else working on idea #2.