Introduction to Database Management Systems (DBMS)

Date: 27 July, 2011 Scribe: Amit Awekar [awekar@iitg.ernet.in]

Today's class dealt with three main topics

1. Understanding importance of DBMS
2. Overview of database design process
3. Functions of and relation between various components in a typical DBMS (refer to lecture slides for the diagram)

Data is the most important entity across all facets of computer science. For example, data structures are designed to represent data in convenient manner and computer networks are designed to transfer data quickly and reliably. Many open research issues in computer science also require handling large amounts of data. For example, one has to derive interesting facts from history of stock prices to predict future stock prices and to answer keyword queries search engines such as Google has to search through multi-peta bytes of data. Importance of data is also evident from the fact that success of most of the well-known companies (Google, Yahoo, Microsoft and many others) in computing domain depends on intelligent data management.

We need to build separate DBMS as traditional file systems and operating systems are not enough to handle complex data management tasks such as transactions, concurrent access, and flexible privacy policies. We need separate data query languages as traditional programming languages are too generic and procedural.

Database design process can be viewed as a three step process.
1. Capture real world vague requirements into an objective model (Conceptual modeling)
2. Convert conceptual model into logical model which can be directly implemented on a computer (Logical modeling)
3. Optimize the physical storage of data for better performance (Physical modeling)

A typical DBMS receives queries through various interfaces. Query evaluation engine converts these queries into optimized execution plan, just like a C compiler converts C program into an executable binary file. File access method take care of security and privacy policies. In most cases, all data cannot fit into main memory. Buffer manager selectively fetches data from secondary storage into the main memory. Disk space manager efficiently arranges data in the secondary storage. Concurrency control and recovery modules are mainly responsible for handling transactions.