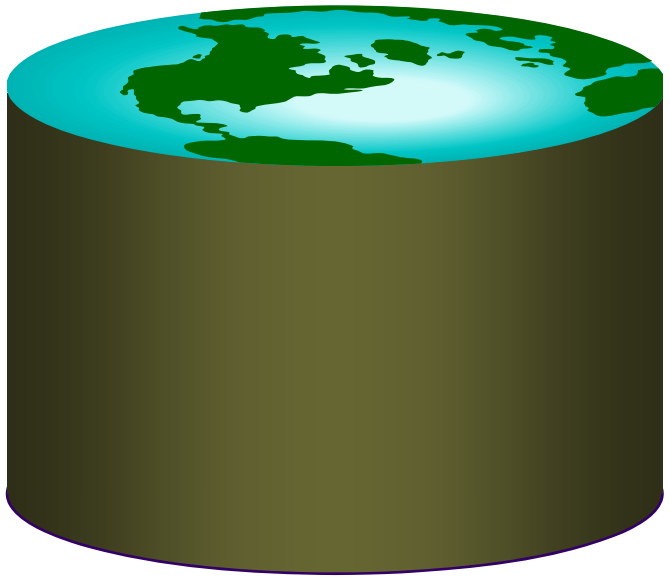
CS 505: Intermediate Topics in Database Systems

Instructor: Jinze Liu

Fall 2008



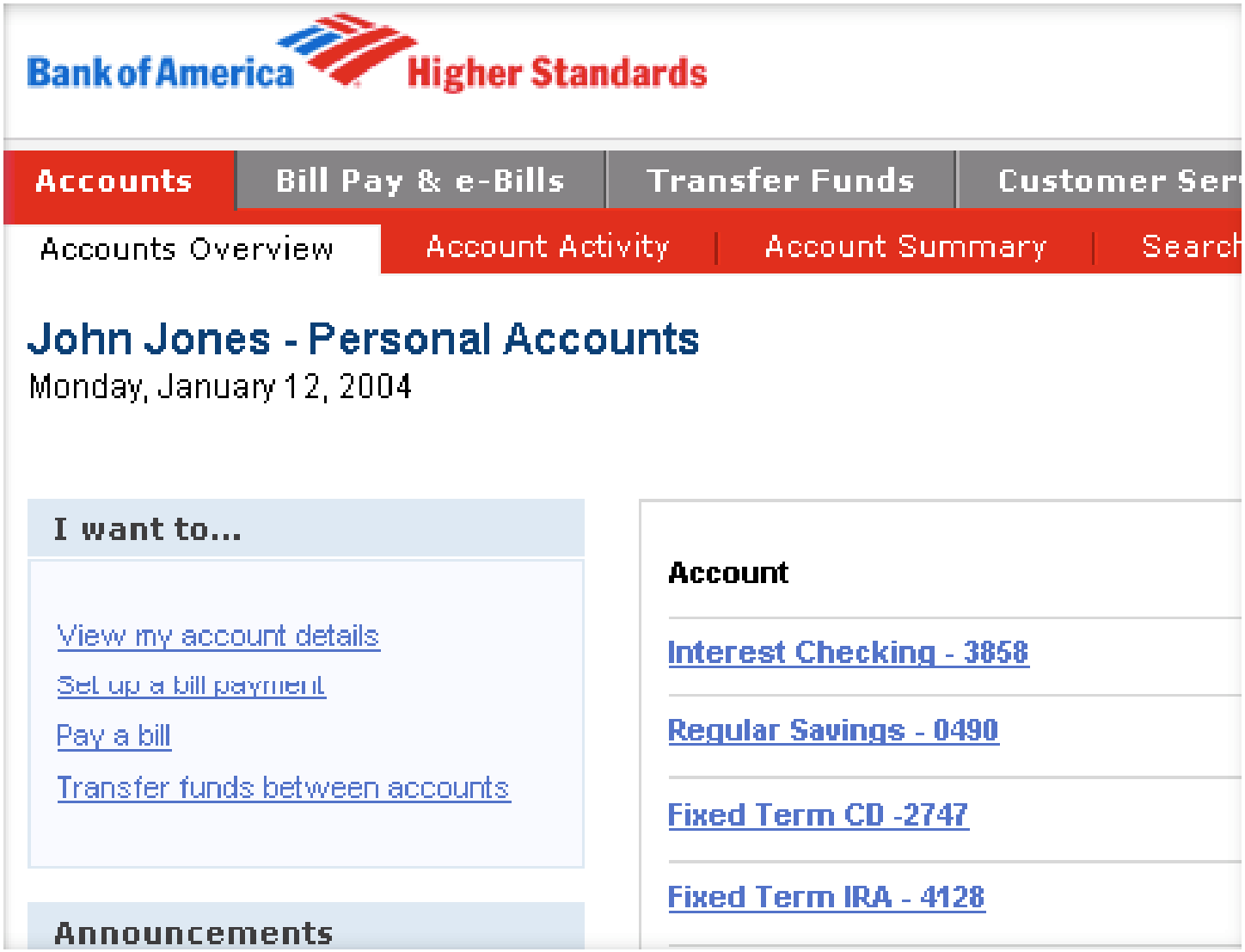
Topics for Today

- What is a database?
- What is a database management system?
- What's the difference between CS405G and CS505?
- Why take a database course?
- How to take the class?
- Preview of class contents

Database Systems?

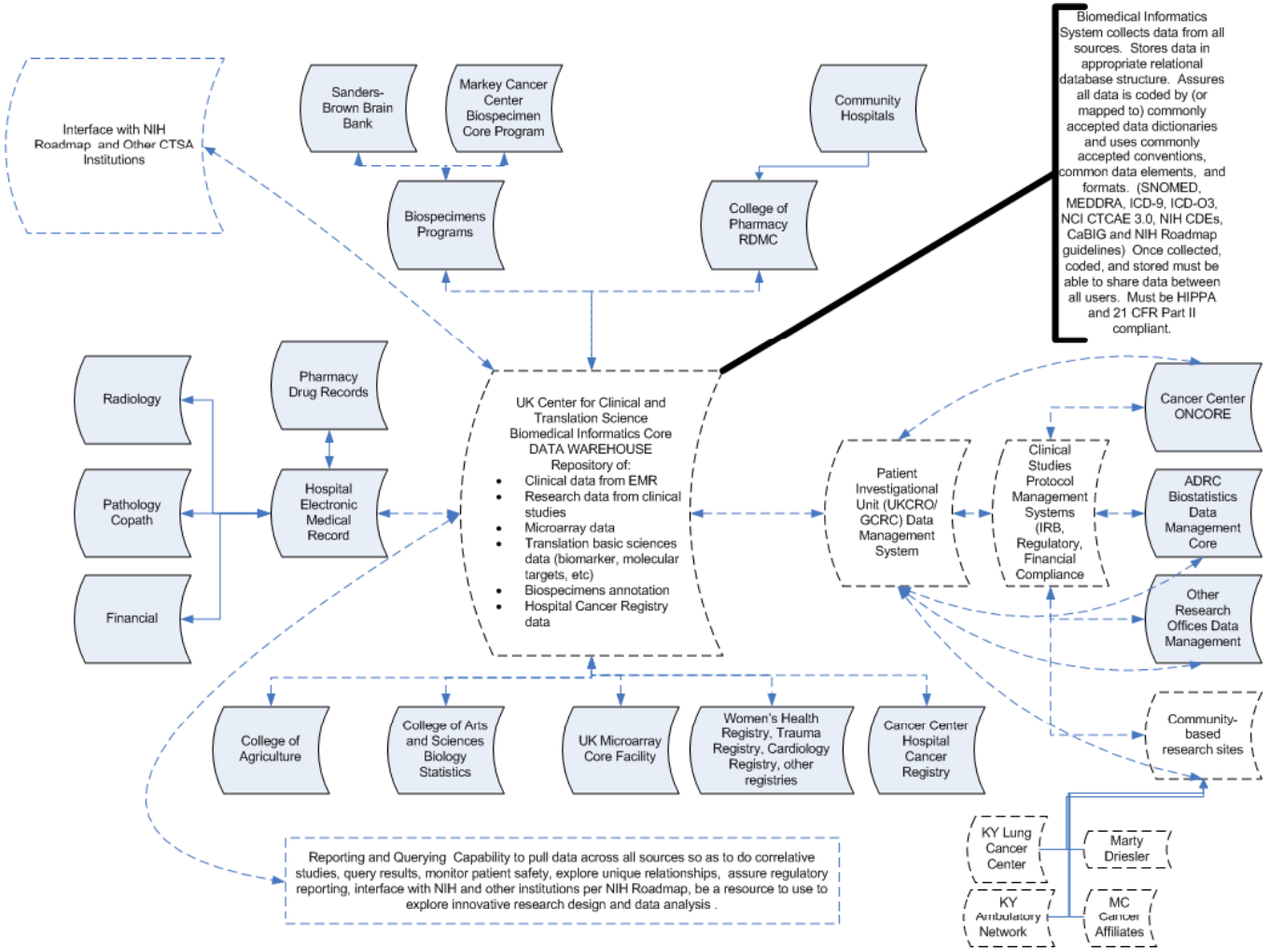
- Name a few!

Database Systems: Bank Systems

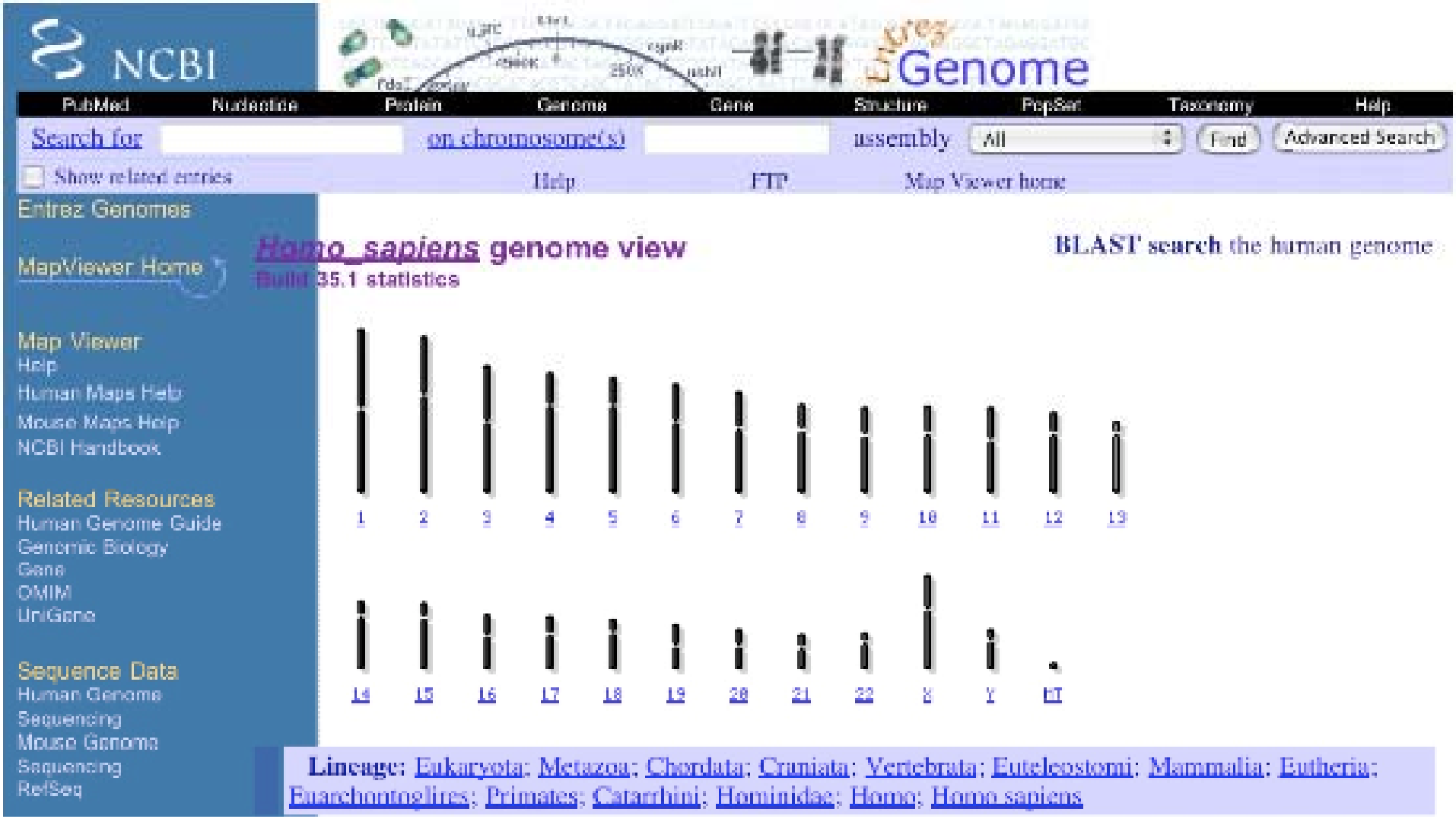


Database Systems - Ecommerce

Database Systems: Clinical Databases



Database Systems: Genome Bank



What is a Database?

- A *database* is an integrated collection of data.
  - Data is a group of facts that can be recorded.

- Typically a database is used to model a real-world "enterprise" (or a *miniworld*)
  - Entities (e.g., *basketball teams*, *games*)
  - Relationships (e.g. *UK's basketball team* beat <you name it> last week)

- Might surprise you how flexible this is
  - Web search:
    - Entities: words, documents
    - Relationships: word in document, document links to document.
  - P2P filesharing:
    - Entities: words, filenames, hosts
    - Relationships: word in filename, file available at host

What is a Database Management System?

- A Database Management System (DBMS) is a collection of programs that enable users to create and maintain databases
  - store, manage, and access data in a database.

- Typically this term is used narrowly
  - Relational databases with transactions
    - E.g. Oracle, DB2, SQL Server
  - Mostly because they predate other large repositories
    - Also because of technical richness
  - When we say DBMS in this class we will usually follow this convention
    - But keep an open mind about applying the ideas!

Main Characteristics of Databases

- Self-describing nature of a database system
  - A DBMS catalog stores the *description* of the database. The description is called meta-data.
- Insulation between programs and data
  - Allows changing data storage structures and operations without having to change the DBMS access
- Data Abstraction
  - Use **data model** to hide storage details and present the users with a *conceptual view* of the database
- Support of multiple views of the data
  - Each user may see a different view of the database, which describes *only* the data of interest to that user.
- Sharing of data and multi-user transaction processing

Databases make these folks happy ...

- End users in *many* fields
  - Business, education, science, …
- DB application programmers
  - Build data entry & analysis tools on top of DBMSs
  - Build web services that run off DBMSs
- Database administrators (DBAs)
  - Design logical/physical schemas
  - Handle security and authorization
  - Data availability, crash recovery
  - Database tuning as needs evolve
- DBMS vendors, programmers
  - Oracle, IBM, MS …

…must understand how a DBMS works

What: Is the WWW a DBMS?

- Fairly sophisticated search available
  - Crawler indexes pages on the web
  - Keyword-based search for pages

- But, currently
  - data is mostly unstructured and untyped
  - search only:
    - can't modify the data
    - can't get summaries, complex combinations of data
  - few guarantees provided for freshness of data, consistency across data items, fault tolerance, …
  - web sites typically have a (relational) DBMS in the background to provide these functions.

What: Is the WWW a DBMS?

- The picture is changing quickly
  - Information Extraction to get structured data from unstructured data
  - New standards e.g., XML, Semantic Web can help data modeling

What makes youtube a success?

- http://mysqldatabaseadministration.blogspot.com/2007/04/youtube-and-mysql.html
- Top reasons for YouTube Scalability includes drinking

What makes youtube a success?

- Top reasons for YouTube database scalability include Python, Memcache and MySQL replication.
  - The fastest query on the database is that is never sent to the database.

- What can go wrong with replication?
  - Replication goes slow?

What makes youtube a success?

- Top reasons for YouTube database scalability include Python, Memcache and MySQL replication.
  - The fastest query on the database is that is never sent to the database.

- What can go wrong with replication?
  - Replication goes slow
  - Inconsistency
  - Unbalanced Load between master and replicas
  - ……

What: Is a File System a DBMS?

- Thought Experiment 1:
  - Yo same file.
  - Yo
  - W

A) You ther E) ???

- Thou
  - –Yo
  - –Th
  - –Which changes survive?

A) All B) None C) All Since Last Save D) ???

Q: How do you write programs over a subsystem when it promises you only "???" ?

A: Very, very carefully!!

OS Support for Data Management

- Data can be stored in RAM
  - This is what every programming language offers!
  - RAM is fast, and random access
  - Isn't this heaven?

- Every OS includes a File System
  - manages *files* on a magnetic disk
  - allows *open*, *read*, *seek*, *close* on a file
  - allows protections to be set on a file
  - drawbacks relative to RAM?

Database Management Systems

- What more could we want than a file system?
  - Simple, efficient *ad hoc*[1] queries
  - concurrency control
  - recovery
  - benefits of good data modeling

- S.M.O.P.[2]? Not really…
  - as we'll see this semester
  - in fact, the OS often gets in the way!

---

[1]ad hoc: formed or used for specific or immediate problems or needs

[2]SMOP: Small Matter Of Programming

Current Commercial Outlook

- A major part of the software industry:
  - Oracle, IBM, Microsoft
  - also Sybase, Informix (now IBM), Teradata
  - smaller players: java-based dbms, devices, OO, …
- Lots of related industries
  - data warehouse, document management, storage, backup, reporting, business intelligence, ERP, CRM, app integration
- Traditional Relational DBMS products dominant and evolving
  - adapted for extensibility (user-defined types), native XML support.
  - Microsoft merger of file system/DB…?

Advantages of a DBMS: a short list

- Controlling redundancy
- Restrict unauthorized access
- Providing persistent storage for program objects
- Providing storage structure for efficient query processing
- Providing backup and crash recovery
- ….
- And many many others that are going to be explored in this class

What database systems will we cover?

- We will try to be broad and touch upon
  - Relational DBMS (e.g. Oracle, SQL Server, DB2, Postgres)
  - "Semi-structured" DB systems (e.g. XML repositories like Xindice)
  - Data mining: transfer data into knowledge!
- Starting point
  - We assume you have used web search engines
  - We assume you know the basics of relational databases
    - Yet they pioneered many of the key ideas
  - So focus will be on relational DBMSs
    - With side-notes on search engines, XML issues

Why take this class?

A. Database systems are at the core of CS
B. They are incredibly important to society
C. The topic is intellectually rich
D. ~~It isn't that much work~~
E. Looks good on your resume

Let's spend a little time on each of these

Why take this class?

A. Database systems are the core of CS

- Shift from computation to information
  - True in corporate computing for years
  - Web, p2p made this clear for personal computing
  - Increasingly true of scientific computing

- Need for DB technology has exploded in the last years
  - Corporate: retail swipe/clickstreams, "customer relationship mgmt", "supply chain mgmt", "data warehouses", etc.
  - Web:not just "documents". Search engines, e-commerce, blogs, wikis, other "web services".
  - Scientific: digital libraries, genomics, satellite imagery, physical sensors, simulation data
  - Personal: Music, photo, & video libraries. Email archives. File contents ("desktop search").

Why take this class?

B. DBs are incredibly important to society

- "Knowledge is power." -- Sir Francis Bacon

- "With great power comes great responsibility." -- SpiderMan's Uncle Ben

Policy-makers should understand technological possibilities.
Informed Technologists needed in public discourse on usage.

Why take this class?

C. The topic is intellectually rich.

- representing information
  - data modeling
- languages and systems for querying data
  - complex queries & query semantics*
  - over massive data sets
- concurrency control for data manipulation
  - controlling concurrent access
  - ensuring transactional semantics
- reliable data storage
  - maintain data semantics even if you pull the plug
- data mining
  - Let your data speak

* semantics: the meaning or relationship of meanings of a sign or set of signs

Why take this class?

~~**D. It isn't that much work.**~~

- Bad news: It is a lot of work.
- Good news: the course is balanced and fun

Why take this class?

E. Looks good on my resume.

- Yes, but why? This is not a course for:
  - Oracle administrators
  - IBM DB2 engine developers
    - Though it's useful for both!
- It is a course for well-educated computer scientists
  - Database system concepts and techniques increasingly used "outside the box"
    - Ask your friends at Microsoft, Yahoo!, Google, Apple, etc.
    - Actually, they may or may not realize it!
  - A rich understanding of these issues is a basic and (un?)fortunately unusual skill.

About the course: Information

- Class web page is at
  - http://protocols.netlab.uky.edu/~liuj/teaching/
  - Syllabus, homework, grading policy, etc. available from class web page
- Textbook
  - Database: the Complete book
  - Can get it from the bookstore
- Jinze's Office Hours:
  - 237 Hardymon building
  - Email: please include CS505G in the subject line for fast response
- Class mailing list
  - Will be used for announcement/clarification of assignments/answering questions

About the Course – Workload

- First Part
  - 3 homework assignments (30%)
  - 1 Programming project (40%)
  - Mid-term Exam (30%)
- Second Part
  - 3 homework assignments (30%)
  - 1 Programming project (40%)
  - Final Exam (30%)

- Cheating policy: zero tolerance
  - We have the technology…

Project

- Programming projects have a practical, hands-on focus:
  - A database for a particular application
    - To be named (let me know your interest!)
  - Two stages
    - Basic database implementation
      - Relational database systems + web interface
      - XML database + web interface
    - Extra features
      - Database security
      - Query speed-up
      - Data mining features
  - Projects are to be done in teams of 2
  - Pick your partner ASAP!