Outline

- Announcements
- Databases (cont’d)
- Algorithms and Protocols
- Student Presentations
- Akamai
Announcements I

- Database Assignment due 12/2 (submit electronically)
- Business paper – due 12/2 (last day of instruction)

Announcements II

- Student Presentations next week?
  - ??
- Reading:
  - Chapter 10 of Messerschmitt (Reader 1)
  - American Airline Case Study (Reader 2)
  - Chapter 1 on Networking
- 2nd Database tutorial
  - Friday, Dec. 2, 3:00 p.m., BE109
Student Presentations

- Rachel Karagianes - Artificial Skin
- Eleonor Concepción - Galaxy Hotel System

The Relational Model

<table>
<thead>
<tr>
<th>EMPLOYERS</th>
<th>DEPARTMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPL_ID</td>
<td>DEPT_ID</td>
</tr>
<tr>
<td>EMPL_NAME</td>
<td>DEPARTMENT</td>
</tr>
<tr>
<td>EMPL_POSITION</td>
<td>ADDRESS</td>
</tr>
<tr>
<td>EMPLOYER</td>
<td>DEPARTMENT</td>
</tr>
<tr>
<td>100 Alice Manager 1</td>
<td>IT San Jose</td>
</tr>
<tr>
<td>101 Bob Programmer 1</td>
<td>2 Finance New York</td>
</tr>
<tr>
<td>102 Chris Manager 2</td>
<td>1 IT San Jose</td>
</tr>
<tr>
<td>103 David Accountant 2</td>
<td>2 Finance New York</td>
</tr>
</tbody>
</table>
Database Operations

**EMPLOYERS**

<table>
<thead>
<tr>
<th>EMPL_ID</th>
<th>EMPL_NAME</th>
<th>EMPL_POSITION</th>
<th>DEPT_ID</th>
<th>DEPT. ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Alice</td>
<td>Manager</td>
<td>1</td>
<td>San Jose</td>
</tr>
<tr>
<td>101</td>
<td>Bob</td>
<td>Programmer</td>
<td>1</td>
<td>San Jose</td>
</tr>
<tr>
<td>102</td>
<td>Chris</td>
<td>Manager</td>
<td>2</td>
<td>New York</td>
</tr>
<tr>
<td>103</td>
<td>David</td>
<td>Accountant</td>
<td>2</td>
<td>New York</td>
</tr>
</tbody>
</table>

**DEPARTMENTS**

<table>
<thead>
<tr>
<th>DEPT_ID</th>
<th>DEPARTMENT</th>
<th>DEPT. ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IT</td>
<td>San Jose</td>
</tr>
<tr>
<td>2</td>
<td>Finance</td>
<td>New York</td>
</tr>
</tbody>
</table>

**JOIN**

**EMPL_DEPT**

<table>
<thead>
<tr>
<th>EMPL_ID</th>
<th>EMPL_NAME</th>
<th>EMPL_POSITION</th>
<th>DEPT. ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Alice</td>
<td>Manager</td>
<td>IT</td>
</tr>
<tr>
<td>101</td>
<td>Bob</td>
<td>Programmer</td>
<td>IT</td>
</tr>
<tr>
<td>102</td>
<td>Chris</td>
<td>Manager</td>
<td>Finance</td>
</tr>
<tr>
<td>103</td>
<td>David</td>
<td>Accountant</td>
<td>Finance</td>
</tr>
</tbody>
</table>
Application Logic and Tables

Databases & OLTP

- Click to add text
Recall - Two capabilities

Aggregation: accessing multiple databases

Sharing: two or more applications accessing the same databases

Example - Travel Agency

Travelocity.com

CheapTickets.com

Hotels
Cars
Airtickets

What can go wrong?
Example - Travel Agency

A resource might be unavailable

Example - Travel Agency

Two applications might try to access & update the same resource concurrently
An application or a host might crash before the completion of the transaction.

A customer’s transaction should be completed in its entirety, or aborted.
Transaction Processing

“The coordination of multiple resources and the shared access to common resources in a systematic and consistent way”

Examples?
- Financial applications (stock market, ATMs)
- Reservations (travel, theatre)
- Manufacturing (inventory, purchasing, billing)
- Etc...

Online Transaction Processing
(OLTP)

Transaction Processing for networked applications

4 Important Properties of transactions: ACID
- Atomicity
- Consistency
- Isolation
- Durability
The ACID properties

- **Atomicity**
  - All transaction components should either complete together (commit) or abort
  - E.g. All reservations (airline, hotel, car) should be grouped as a single transaction that either commits, or aborts
- **Consistency**
  - A transaction must leave the system in a consistent state at the end of the transaction, or else abort
  - E.g. Either a consistent set of reservations has been made, or none
- **Isolation**
  - Concurrent transactions are allowed only if they don’t interfere with each other
  - Two travel agents can concurrently access the same database if the reservations are for different dates/places
- **Durability**
  - A transaction leaves the resources in a permanent state after it commits

Structure of a Transaction
OLTP

- Simplifies application development

- Enables protection and integrity of mission-critical data in a transparent way
  - for the end user
  - for the application developer

Monopoly players protocol
Application and infrastructure

The application defines its own application-level protocols

Internally, the network uses protocols to implement the services it provides

Example:

HEADQUARTERS

Airline Intranet

Airline Dataserver

HHC Server

Wireless Link

HHC
Layered Protocols Example

Example: HTTP (Hyper Text Transfer Protocol)

1. User activates URL
2. HTTP request
3. Browser displays document (if HTML) or invokes “helper application”
4. HTTP response (embedded document)
Example

Path from [start position] to [end position] is (R,D,D,D,R,R,R,R)

Is (R,D,D,D,R,R,R,R) an address? No! -- not an address, because it depends on starting point

Example

Address of [end position] is (6,5)

Route from [start position] can be inferred
**Program**

- Precise description of an algorithm in a formal language that is called programming language
- Actions are applied to data

**Formulation in a language**

- **Natural language**
  - No strict syntactic rules
  - Great density and semantic capability
- **Formal language**
  - Strict syntax and semantics
- **Programming language**
  - Formal language in which computations can be described
  - Executable by an electronic computer
Can we solve all problems?

Collatz Conjecture (Ulam):

\[
\text{while } x \neq 1 \text{ do} \\
\text{if (x is even) then } x = x/2 \\
\text{else } x = 3x + 1
\]

Example:

\[
7 \rightarrow 22 \rightarrow 11 \rightarrow 34 \rightarrow 17 \rightarrow 52 \rightarrow 26 \rightarrow 13 \rightarrow 40 \rightarrow 20 \rightarrow 10 \rightarrow 5 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1
\]

Given any arbitrary number \( x \), will the program terminate?

Open problem!

Translation of programs

Source Code

\((\text{in a programming language})\)

\[\downarrow\]

Compiler

\[\downarrow\]

Input \rightarrow Executable program \rightarrow Output

\((\text{machine language})\)
Quiz 4  (total 10 pts)

- What is SQL?

- How long (in bits) is an IP address?

- Akamai is famous for what?