Outlines

- 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></td>
<td>DEPT_NAME</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
</tr>
<tr>
<td>101</td>
<td>1</td>
</tr>
<tr>
<td>102</td>
<td>2</td>
</tr>
<tr>
<td>103</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DEPT_ID</th>
<th>DEPARTMENT</th>
<th>DEPT_ADDR</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>

Primary Keys
Database Operations

<table>
<thead>
<tr>
<th>EMP_ID</th>
<th>EMP_NAME</th>
<th>EMPLOYERPOSITION</th>
<th>DEPT_ID</th>
<th>DEPT</th>
<th>DEPT_ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>Alice</td>
<td>Manager</td>
<td>1</td>
<td>IT</td>
<td>San Jose</td>
</tr>
<tr>
<td>101</td>
<td>Bob</td>
<td>Programmer</td>
<td>1</td>
<td>IT</td>
<td>San Jose</td>
</tr>
<tr>
<td>102</td>
<td>Chris</td>
<td>Manager</td>
<td>2</td>
<td>Finance</td>
<td>New York</td>
</tr>
<tr>
<td>103</td>
<td>David</td>
<td>Accountant</td>
<td>2</td>
<td>Finance</td>
<td>New York</td>
</tr>
</tbody>
</table>

Application Logic and Tables

Databases & OLTP

Recall - Two capabilities

Example - Travel Agency

What can go wrong?
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.

**Application and infrastructure**

The application defines its own application-level protocols.

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

**Example:**

- Wireless Link
- HHC server
- HEADQUARTERS
- Airline Intranet
- Airline Data server
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):

```plaintext
while x!=1 do
  if (x is even) then x=x/2
  else x=3*x+1
Example:
7 → 22 → 11 → 34 → 17 → 52 → 26 → 13 → 40 → 20 → 10 → 5
16 → 8 → 4 → 2 → 1
Given any arbitrary number x, will the program terminate?
Open problem!
```

**Translation of programs**

Source Code
(in a programming language)

↓

Compiler


Input → Executable program → Output
(machine language)

---

**Quiz 4  (total 10 pts)**

- What is SQL?

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

- Akamai is famous for what?