Web and Databases

- E-commerce sites
  - Products, order, customers
- News sites
  - Subscribers, articles
- Web boards
  - Users, postings
- ... anywhere where a large amount of information needs to be managed safely and efficiently

Database vs. File

- More efficient search
- ACID
  - Atomicity
  - Consistency
  - Isolation
  - Durability

Relational Model

- Proposed by Edgar F. Codd in early 1970’s
- All major DBMS are relational (and the good ones are object-relational)

A Relational DB Example

<table>
<thead>
<tr>
<th>orders</th>
<th>products</th>
</tr>
</thead>
<tbody>
<tr>
<td>CID</td>
<td>CID</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>PID</td>
<td>Description</td>
</tr>
<tr>
<td>1</td>
<td>Intel Pentium 4</td>
</tr>
<tr>
<td>2</td>
<td>Intel Pentium 4</td>
</tr>
<tr>
<td>3</td>
<td>AthlonXP</td>
</tr>
<tr>
<td>4</td>
<td>ASUS</td>
</tr>
<tr>
<td>5</td>
<td>TYAN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>customers</th>
<th>order_details</th>
</tr>
</thead>
<tbody>
<tr>
<td>CID</td>
<td>FName</td>
</tr>
<tr>
<td>1</td>
<td>Chengyu</td>
</tr>
<tr>
<td>2</td>
<td>Steve</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OID</th>
<th>PID</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Terminology

- Database Management System (DBMS)
- Database
- Table, relation
- Attribute, field
  - Type
- Record, tuple, row
- Column
- Schema
**SQL**

- Standard query language of relational databases
- Supported by all major relational databases with some variations

**MySQL**

- Not a good DBMS in the traditional sense
- Very popular in web development
  - Very fast search
  - Full text indexing and search
  - Many small things
    - `drop if exists`
    - `insert into values`
    - `*/` `*/`
    - `...`

**Databases in MySQL**

- MySQL Server
  - tables
  - indexes
  - constraints
  - views
  - ...

- user information
- access privileges

**MySQL on the CS Server**

- Version 5.0.22
- One database per user
  - DB name is the same as the server account user name. E.g. `cs320stu31`
  - Username and password are the same as the ones for the server account
- Connect to the database
  - `mysql -p`

**mysql Command Line Options**

- `mysql [database]`
- `-?`
- `-u username`
  - default: current user
- `-p`
  - required if the password for the account is not empty
- `-h hostname`
  - default: localhost

**Some MySQL Commands**

- Status
  - `status;`
- Help
  - `\h` or `help;`
- Quite MySQL client
  - `\q` or `quit;` or `exit;`
- Change password
  - `set password = password('something');`
  - `set password for 'user'@'host' = password('something');`
More MySQL Commands ...

- Show databases
  - `SHOW DATABASES;
- Use database
  - `USE dbname;
- Show tables
  - `SHOW TABLES;
- Show table schema
  - `DESCRIBE tablename;

... More MySQL Commands

- Run a script
  - `. demo.sql` or `SOURCE demo.sql;
- Run a script at command prompt
  - `mysql < demo.sql

Create a Table

```sql
CREATE TABLE table_name (
  field_name field_type [NOT NULL] [UNIQUE] [DEFAULT value],
  field_name field_type [NOT NULL] [UNIQUE] [DEFAULT value],
  [PRIMARY KEY(field_name, ...)]
);
```

```sql
CREATE TABLE products (
  prod_id char(8) NOT NULL, -- product id
  description text, -- product description
  price decimal(12,2), -- price
  PRIMARY KEY (prod_id)
);
```

Field Types

- **Numerical types**
  - int, float, double, decimal(m,n)
- **String types**
  - char(n), varchar(n)
- **Date and time**
  - date, time, datetime, timestamp
  - 'yyyy-mm-dd hh:mm:ss'

Auto Increment Field

```sql
CREATE TABLE users (
  id int auto_increment PRIMARY KEY,
  username varchar(64) NOT NULL UNIQUE,
  password char(16)
);
```

```sql
INSERT INTO users (username, password) VALUES ('cysun', 'abcd');
INSERT INTO users (username, password) VALUES ('csun', 'xyz');
```

Populate Tables

- **Insert a record**
  - `INSERT INTO orders VALUES (1000, 1, '2004-04-29', '2004-05-01');`
  - `INSERT INTO orders VALUES (1001, 2, '2004-05-01', NULL);`
- **Load a data file**
  - `LOAD DATA LOCAL INFILE 'orders.txt' INTO TABLE orders;`
- **Import a data file (at command prompt)**
  - `mysqlimport -u cs320stu31 -p orders.txt`
  - `\N` for NULL
Search for Records

select field(s) from table(s) where condition(s);

- select description, price from products;
- select * from products;
- select * from products where price < 300;
- select * from products where prod_id = 'cpu-0001';

Pattern Matching

- LIKE, REGEXP
  - % -- any zero or more characters
  - _ -- any single character
  - [abc], [a-z], [0-9] -- range
  - * -- zero or more instances of the preceding character
  - ^ -- beginning of a string
  - $ -- end of a string

- select * from products where description like '%intel%';

Update Records

update table set field=value [, ...] where condition(s);

- update products set price=320 where prod_id = 'cpu-0001';
- update products set price=200, description='Intel Pentium M 1.7GHz' where prod_id = 'cpu-0001';

Delete Records

delete from table where condition(s);

- Examples:
  - delete from orders;
  - delete from orders where order_date < '2003-12-31' and ship_date is not null;

- Drop a database
  - drop database cs320stu31; -- Don't do this!

- Drop a table
  - drop table products;

Schema Design Example ...

Customer, Product, Order

public class Customer {
    int id;
    String lastName;
    String firstName;
    String address;
}

public class Product {
    int id;
    String description;
    double price;
}

... Schema Design Example

public class Order {
    int id;
    Date dateOrdered;
    Date dateShipped;
    Customer customer;
    Map<Product, int> products;
}
Simple Schema Design Rules

<table>
<thead>
<tr>
<th><strong>OO</strong></th>
<th><strong>Relational</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>Table</td>
</tr>
<tr>
<td>Class variables</td>
<td>Attributes</td>
</tr>
<tr>
<td>Java types</td>
<td>SQL types</td>
</tr>
<tr>
<td>References</td>
<td>ID</td>
</tr>
<tr>
<td>Collection</td>
<td>New Table</td>
</tr>
</tbody>
</table>

Exercises

- Read MySQL Reference Manual
- String functions
- Date and time functions