ECE 650
Systems Programming & Engineering

Spring 2018

PostgreSQL Database and C++ Interface

Also some midterm review

Tyler Bletsch

Duke University

Slides are adapted from Brian Rogers (Duke)

PostgreSQL

- Also called Postgres
- Open source relational database system
- Based on SQL
- Features:
  - ACID compliant (i.e. the properties we want for **transactions**)
  - Supports foreign keys, joins, views
  - Many useful built-in data types
  - Interfaces for C/C++, Java, Python, Ruby, …
  - Sophisticated query optimizer
- Other common SQL alternatives
  - MySQL/MariaDB (free, older)
  - Microsoft SQL server (paid)
  - Oracle (paid, expensive)
  - SQLite (free, fast, single-user)

Postgres Installation

- Open-source; available for Linux, MacOS, Windows, ...
- I'll show steps for Linux; tested using Ubuntu VM image
  - https://vm-manage.oit.duke.edu/vm_manage
- Install Postgres:

```
sudo apt-get install postgresql
sudo apt-get install postgresql-contrib
```

- Install C++ API:

```
sudo apt-get install libpqxx-dev
```

Other Setup

- By default, installation creates a user 'postgres'
- Connect to postgres server and set up password

  ```
  sudo su - postgres
  psql
  ALTER USER postgres with encrypted password 'abc123';
  ```

  - Then execute command '\q' to leave postgres
  - Then 'exit' to exit from 'postgres' user back to your default user ID
- Find file pg_hba.conf in your system and edit as follows:

  ```
  sudo vim /etc/postgresql/9.5/main/pg_hba.conf
  ```

  - Change this line:

  ```
  local   all       postgres                      peer
  ```

  - To:

  ```
  local   all       postgres                      md5
  ```

This allows passwords on local (UNIX FIFO based) connections as opposed to just looking at UNIX username. You can skip this if you always connect explicitly over TCP by giving “-h 127.0.0.1” when connecting.

- Restart postgres:

  ```
  sudo service postgresql restart
  ```

Create a Database

- Start Postgres shell

  psql -U <userid>

  e.g. psql -U postgres
- Create database

  CREATE DATABASE testdb;
- Connect to database:

  \l   # to list available databases

  \c testdb # to connect to 'testdb' database

- Connect to database when running shell

  psql -U <userid> <database>

  e.g. psql -U postgres testdb
- Can run a file of SQL commands by adding -f, e.g.:

  psql -U postgres testdb -f mycommands.sql

<u>Shell warning</u>

You need to end commands with a semicolon.

If you don’t, it will assume you’re entering a multiline command and quietly give you another prompt.

Database Operations

- Create tables

```
CREATE TABLE COMPANY(
  ID INT PRIMARY KEY     NOT NULL,
  NAME           TEXT    NOT NULL,
  AGE            INT     NOT NULL,
  ADDRESS        CHAR(50),
  SALARY         REAL,
  JOIN_DATE      DATE
);
```

- List info about tables in a database

```
\d                  # Show overview of all tables
\d company          # Show details of 'company' table
```

Database Operations

- Insert rows into a table

```
INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY,JOIN_DATE) VALUES
   (1, 'Paul', 32, 'California', 20000.00 ,'2001-07-13');
INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,JOIN_DATE) VALUES (2, 'Allen',
   25, 'Texas', '2007-12-13');
INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY,JOIN_DATE) VALUES
   (3, 'Teddy', 23, 'Norway', 20000.00, DEFAULT );
INSERT INTO COMPANY (ID,NAME,AGE,ADDRESS,SALARY,JOIN_DATE) VALUES
   (4, 'Mark', 25, 'Richmond ', 65000.00, '2007-12-13' ), (5, 'David', 27, 'Texas',
   85000.00 , '2007-12-13');
```

- Query a database

```
SELECT * from COMPANY;
SELECT ID, NAME, SALARY FROM COMPANY;
```

Midterm review

Midterm Topics (1)

These slides represent a summary of the course to date. For specific study tips on the midterm, see the link on the course site.

- Concurrency & Synchronization
  - Process vs. Thread
  - Concurrent Programming
  - Race conditions, mutual exclusion, synchronization
- IPC
  - Shared memory vs. Message passing
  - Mmap for shared memory across different processes
  - UNIX Fifos and Pipes for messaging
- Networking Introduction
  - Network structure
  - Circuit vs. packet switching
  - Network stack & Layering (OSI & TCP/IP models)

Midterm Topics (2)

These slides represent a summary of the course to date. For specific study tips on the midterm, see the link on the course site.

- Link Layer
  - Framing (how to divide bit streams into frames)
  - Error detection & error correction
  - Link layer protocols (stop & wait, sliding window)
- Network Layer
  - Connectionless vs. connection-oriented service
  - Routing concepts and routing algorithms
  - Count-to-infinity problem
- Transport Layer
  - Sockets
  - Flow control and sequence numbers

Midterm Topics (3)

These slides represent a summary of the course to date. For specific study tips on the midterm, see the link on the course site.

- Relational databases
  - Relation schema, Relations, domains, constraints
  - Relational algebra operations
- SQL
  - SQL terminology
  - SQL query operations & options; how to retrieve data
- Database transactions
  - Database model for transactions
  - Motivation for concurrency control (3 problems)
  - System log
  - ACID properties of transactions
  - Serializability