CS 166: Lab7 Assignment: Indexing

May 16, 2019

The purpose of the following assignment is to explore how indexes can be used to improve the performance of SQL queries, the different types of indexes supported by Postgres, and the issues involved around indexes and the optimizer.

Having read chapter 8 in the textbook will help you understand many aspects of this lab.

For this assignment you will need to download and unzip lab7.zip from iLearn. Make sure that you are unzipping the archive in your */tmp/$USER/data* directory File *create_tables.sql* contains table schema definition and allows you to create tables and load data into them.

Write the following SQL queries and measure their execution time **both with and without** using indexes:

1. Count how many parts in NYC have more than 70 parts on_hand
2. Count how many total parts on_hand, in both NYC and SFO, are Red
3. List all the suppliers that have more total on_hand parts in NYC than they do in SFO.
4. List all suppliers that supply parts in NYC that aren't supplied by anyone in SFO.
5. Update all of the NYC on_hand values to on_hand - 10.
6. Delete all parts from NYC which have less than 30 parts on_hand.

Some notes about the data model and the questions:

- part_number is the primary key for each part table. But it is <u>not</u> unique across both tables.
- If a part has the same number in NYC and SFO it is the same part, regardless of color, etc.
- If I say, e.g. "Red parts", I mean color_name = "Red" not color = 0.

- Different suppliers may supply the same part in NYC and SFO.

If you want to measure execution time of individual query you need to execute "\timing" command in interactive psql terminal. I have provided you with a *measure.sh* script, which measures this time for you. Execution time can vary depending on the multiple factors, hence be sure that you report **average** time across several executions.

Write all your queries in *queries.sql* file. File *create_indexes.sql* should contain create statements for all the indexes you decide are best for the queries you have written.

Notice that the last two questions are changing the data. To make the experiment reasonable, make sure that your queries run on the same data, with and without indexes.

You should experiment with B-tree indexes, built on different columns. Use the following syntax for create index statement:

```
CREATE INDEX index_name
ON table_name
[USING BTREE]
(index_col_name)
```

You can find the Postgres documentation on indexes at
http://www.postgresql.org/docs/8.1/static/indexes.html

Requirement

You should have the database ready before running measure.sh script:

1. start the database:

   ```
   source [PATH]/startPSQL.sh
   ```

2. Create the database:

   ```
   source [PATH]/createPostgreDB.sh
   ```

If everything is fine, you should be able to connect to the database successfully:

```
psql -h localhost -p $PGPORT $USER"_DB"
```

Copy the lab7 files to your data directory:

```
cp [PATH]/lab7/* /tmp/$USER/[data-folder]
```

you should be able to see the files in your data directory.

measure.sh

The script fist creates the tables:

```
psql -h localhost -p $PGPORT $USER"_DB" <
/tmp/$USER/[data-folder]/create_tables.sql
```

you should be able to list the tables, and see the data in your database after this. Next, it runs the indexes script file (you should modify it based on the assignment description).

```
psql -h localhost -p $PGPORT $USER"_DB" <
/tmp/adava003/myDB/data/create_indexes.sql
```

You should have the indexes in the database now, connect to the database and run the following to see the indexes:

```
select * from pg_indexes where schemaname = 'public';
```