Third Practical Assignment – Understanding query evaluation plans

1 Introduction

The practical will involve writing queries in SQL using postgresql, and comparing the performance of different indexing schemes and corresponding evaluation plans.

The data will be a subset of the database used for the movie information database, see http://www.imdb.com..

The relations and datatypes of the schema are:

Actor(id,fname, lname, gender)
Movie(id, name, year)
Directors(id,fname,lname)
Casts(pid,mid,role)
Movie_Directors (did,mid)
Genre(mid,genre)

All id fields are integers, all other fields are character strings.

We will be using a subset of IMDB database. It might not include correct values or even have some missing values due to the character set limitations and textual information stored in the corresponding fields. However, it's a recent replica of the IMDB dataset from their publicly available mirrors (http://www.imdb.com/interfaces). In this practical, missing values will not be a problem in your evaluation.

2 Running the practical

You can run the practical during the lab sessions. But if you prefer, you can also run it on your own – this will also give you more time to experiment, since certain actions you do on the data may take time.

If you run it in the lab, then you just connect to the server by running:
psql -U imdb

The schema is already loaded, but should have no keys or indices – if there are any such keys or indices, you will need to drop them with an SQL DROP INDEX or ALTER TABLE command, in order to start with the expected instance.

If for some reason, you need to reload the database from scratch, this is possible by running the script

```
IMDB_noindex.sql
```

that you can find in the directory /usr/local/practicals/databases.

If you want to run the practical on your own machine, then the steps are:

- get a copy of postgres at *http://www.postgresql.org/download* and install it on your local machine
- create the schema and load the data using the script available on the webpage. When you unzip it, you should have a file

  ```
  IMDB_noindex.sql
  ```

  in some directory on your machine. Run psql from the same directory, and then give the command

  ```
  \i  IMDB_noindex.sql
  ```

  from the psql prompt. This could take upwards of 30 minutes.

3 The task

Regardless of how you get to the database, consider the following query:
QUERY: Find the name(s) of the movie(s) after 2010 in which the actor 'Tom Hanks' played.

1. Write the query in SQL and run using 'EXPLAIN ANALYZE' command:

   An example of the use of this command is:

   EXPLAIN ANALYZE SELECT m.name FROM Movie m

2. Observe the query processing time and the evaluation plan and explain briefly; what does this plan mean?

   Please provide and briefly describe the estimated and actual processing times. You shall also provide the total number of items in your query result.

   In order to do this you will want to read the information about Postgesql evaluation plans available on the link from the course webpage.

3. Create an index on Casts(pid) and run the same query to observe the difference in query processing time. Briefly explain your results.

4. Which indexes are necessary to efficiently process the query? Create them, and show the indices you created, your best time, and the evaluation plan for processing the query.