

Database Systems, CSCI 4380-01
Homework # 8
Due Monday May 9, 2016 at 11:59:59 PM

Introduction.

In this homework, you are allowed to work in groups of at most 2. If you do, please write the name of your group members in your solution file. Turn in a single submission (PDF or text) per group. This is a mini-homework, worth about 50 points (half of a usual homework).

This homework is about database tuning. You are given your own database of IMDB (as in Homework #4) and a number of queries (mostly from answers to Homework #4, but includes some updates). Suppose your workload is given by these queries and each one is equally likely. You are trying to reduce the cost of total runtime. You are asked to find at least 5 indices to create or query changes that reduce the total cost (each worth 10 points). You must have at least 3 indices, not all changes can be query changes. Please read on how to accomplish this.

- Make sure you do not have any indices that were not given to you in the IMDB file. You should have an index for each primary index. That is all.
- Run `analyze` to make sure you have generated all the statistics first.
- Download the file called `hw8queries.sql`. Do not run the queries in this file as they costly, but find their estimated runtime.

For each query `Q`, run `explain Q` and look at the query plan. Simply record the time to first and all tuples as well as the total cost in your homework solution. For example, I see the following first line for `explain query Q3`:

```
HashAggregate (cost=23046.10..178174.70 rows=4708 width=24)
```

which means the time to get the first tuple is 23046.10 and all the tuples is 178174.70. We mostly care about the second number for this homework, but you should report on the performance change in both values.

Your numbers may be different depending on which server you execute this. As long as you use the same server for all parts, it is fine. Simply report on the two numbers for each query and their total.

- Now, find different indices that may help each query. Best thing is to find indices that will help multiple queries. For each index you create, look at the query plan and check that it is using your index. Has the cost reduced? If so, keep it.

You can use any type of index: clustered or unclustered. If you don't know what that means, review the lecture notes for Database Tuning.

Remember to recreate the statistics after creating an index to make sure that they are up to date (though latest version of Postgresql does not seem to need this step), by running `analyze`.

- When you have found which indices are useful, put in your homework solution the index creation commands and any query plan that uses the newly created indices fully.

- Alternatively, you can rewrite the queries that reduce the cost (hopefully without changing their result, so be careful). If so, put your new query into homework solution as well.
- Finally, put the cost of each query after all changes and mark total savings in your homeworks.

You are asked to find at least 5 indices or query changes but you can create more. We will announce the best homework times before the finals! I was able to get savings of 50% and 22% for first and all answers (going down for all from total 265K to 56K).

Deliverables

In short, we expect a single text file (PDF is not needed, but also fine) with the following information:

Homework #8:

Team members: Sibel Adali (adalis@rpi.edu)

Query costs before tuning

(note which server it ran on, e.g. my own Macbook Air, Early 2015):

	First	All
Q1	1	2
Q2	3	4

Total	4	6

Indices created:

```
create index abc ...
create index def ...
```

Query plans effected (an index may impact more than one query):

Q1:

New query plan

Q2:

New query plan

Query costs after tuning:

	First	All
Q1	1	2
Q2	2	2

Total	3	4

Savings:

First: $3/4. = 0.75$

All: $4/6. = 0.66$