

COL362 Assignment-1

Part 1

In this part, we had to make a database using Postgres SQL Language which had three main tables (entities) :-

1. Student - The names and the unique identifiers of the students.
2. Course - The names and unique identifiers of the courses.
3. Teacher - The names and unique identifiers of the teachers.

These can have any string identifier and these identifiers are the primary keys for these tables.

There is another table for section which is a weak entity and its existence depends on the course. It can only have the values - 'A', 'B', 'C', 'D' for each course. Deleting a course deletes all the sections associated to it too.

There are three tables corresponding to the relations :-

1. registers - Every student registers for courses defined by the student_id and course_id
2. teaches - Every course is taught by a teacher defined by the teacher_id and course_id
3. has - Every course can have upto 4 sections defined by the course_id and section_id

Any updations or deletions in the main tables have to be transmitted to all the other tables, hence, the use of "on update cascade on delete cascade".

"not null" function has been used to specify that some of these cant have null values.

"references" function creates the link between the main tables and the tables corresponding to relations.

Part2

Order of time taken for inserting 500,000 tuples in 'registers' table :

using JDBC > INSERT statement for each tuple > bulk load

Tuples were inserted in 'registers' table of the given design and not after removing constraints.

JDBC

JDBC was used statement by statement which takes more time than a single command.

In JDBC, each statement is passed from java to postgres which also takes some time.

Then each statement has to be parsed before it can be executed.

INSERT Statements

In insert statements each line has to be parsed which takes time.

There are more commands(500,000) to be executed as compared to bulk load which takes more time.

Bulk Load

In bulk load is less flexible and data is not parsed it is directly transferred from file to table.

It is done in only one command which reduces time a lot.

It has significantly less overhead for large data loads and is optimized for loading large numbers of rows.

Used COPY to load all the rows in one command, instead of using a series of INSERT commands.

Conclusion:

Bulk load is very fast as compared to other methods for adding large amount of data in the tables. And bulk load should be used for inserting large amount of data in the database.

Assignment done by:

Prakhar Kumar(2015CS10667)

Ankesh Gupta(2015CS10435)

Divgian Singh Sidhu(2015CS10226)