Project: Part 3
Cassandra

1 Assignment

The aim of the project is to design and develop a Cassandra keyspace, starting from requirements specified in terms of a given relational schema.

1.1 Requirement analysis

1. Consider the domain represented by the relational schema `DBDiagram.pdf`.
2. Provide a workload, including at least 5 queries, containing one query for each category below:

   Q1 One query involving only one entity.

   Q2 One query involving at least two entities and at least one join.

   Q3 One query involving at least three entities and at least two joins.

1.2 Logical design

Propose a Cassandra logical schema for the reference domain, in terms of a set of column-families. The schema should:

- contain at least three column-families;
- contain at least one dynamic column family (i.e., a column family containing wide rows with a dynamic schema);

1.3 Implementation

1. Create a Cassandra keyspace and schema according to the result of the design performed in Step 1.3 (Cassandra is available in the docker container, see Exercize 1 for details)

2. Load data into your Cassandra keyspace. To this aim, you can: (i) start from the content of the relational database (available as a set of csv files in directory `DBData`); (ii) transform such data, generating one csv file for each column-family in your schema; (iii) use the CQL `COPY` command to copy file content into the related column-family.

   For what concerns step (ii), you may rely on Trifacta `www.trifacta.com`, just uploading the csv files.

3. Provide a CQL statement for each query in the proposed workload.

1.4 Physical organization*

1. Get information about how many files (`SSTables`) are generated when storing your Cassandra keyspace and how rows are stored in them (e.g., which order is used for each column-family storage).

2. For each query in your workload, determine whether it is executed either by reading only files from main memory (`memtables`) or accessing disk (`SSTable`).

2 Rules for project development and delivery

- Only students which have developed Project Part 1 and Part 2 can develop and deliver a solution for Project Part3. **All other students should contact instructors for a personal project assignment.**

- The project, Part 3, can be developed by groups of up to two persons.

- Each student should upload on AulaWeb, a single zip file of name CognomeN (where N is the initial name letter). The file should contain all the developed code (script for data transformation, CQL statements) and a pdf file with the required documentation for each of the request above.

  The code must include: (i) script for data transformation (if designed); (ii) generated csv files; (iii) all CQL commands for creating keyspace and schema, loading data, specifying queries in the workload. The code must be adequately structured in directories and instructions on how to use the codezip file must also be included.

The documentation must include:

- all the considered assumptions;
- proposed workload, specifying which queries satisfy the proposed constraints;
- logical schema, adequately commented and motivated; in particular for each column family you should specify whether it is a static column family (i.e., a column family containing skinny rows with a specified schema) or a dynamic one (i.e., a column family containing wide rows with a dynamic schema);
- description and motivations of the approaches followed for each assignment (1.1, 1.2, 1.3, 1.4);
- a rough estimate of the effort (in terms, e.g., of number of hours) devoted to the project overall, and on the various parts.

Uploaded files which do not conform to the previous rules, will not be considered.

- Assignment 1.4 is not mandatory but it will lead to an extra bonus.
- Students developing the project in teams can sustain the oral exam on different dates.
- The zip file should be uploaded at least one week before the oral exam (to be scheduled during the written exam or by email).
- A rating among {A+, A, B, C, D} will be assigned to Project Part 3, depending on the quality of the provided solution and documentation.