Databases

Muchang Bahng

Fall 2024

Contents

This is a course on database languages (SQL), database systems (Postgres, SQL server, Oracle, MongoDB), and data analysis.

1 Relational Databases

Definition 1.1 (Data Model)

A **data model** is a notation for describing data or information, consisting of 3 parts.

1. *Structure of the data.* The physical structure (e.g. arrays are contiguous bytes of memory or hashmaps use hashing). This is higher level than simple data structures.
2. *Operations on the data.* Usually anything that can be programmed, such as **querying** (operations that retrieve information), **modifying** (changing the database), or **adding/deleting**.
3. *Constraints on the data.* Describing what the limitations on the data can be.

The most intuitive way to store data is with a *table*, which is called a relational data model, which is the norm since the 1990s.

Definition 1.2 (Relational Data Model)

A **relational data model** is a data model where its structure consists of

1. **relations**, which are two-dimensional tables.
2. Each relation has a set of **attributes**, or columns, which consists of a name and the data type (e.g. int, float, string, which must be primitive).[a]
3. Each relation contains a set[b] of **tuples** (rows), which each tuple having a value for each attribute of the relation. Duplicate (agreeing on all attributes) tuples are not allowed.

So really, relations are tables, tuples are rows, attributes are columns.

Definition 1.3 (Schema and Instance)

The **schema** of a relational database just describes the form of the database, with the name of the database followed by the attributes and its types.

```
Beer (name string, brewer string)
Serves (bar string, price float)
...
```

The **instance** is the actual table, like the collection of tuples.

SQL (Structured Query Language) is the standard query language supported by most DBMS. It is **declarative**, where the programmer specifies what answers a query should return,but not how the query should be executed. The DBMS picks the best execution strategy based on availability of indices, data/workload characteristics, etc. (i.e. provides physical data independence). It contrasts to a **procedural** or an **operational** language like C++ or Python.

[a] The attribute type cannot be a nonprimitive type, such as a list or a set.

[b] Note that since this is a set, the ordering of the rows doesn't matter , even though the output is always in some order.