Relational Databases

Relational Data Model:

- introduced by T. Codd (IBM) in 1970.
- simple, mathematical foundations (mathematical relation, set theory)
- implemented in a large number of commercial systems

SQL: standard for commercial RDBMSs

Previous Data Models: Hierarchical Model, Network Model

proposed in the 60s. implemented in 70s, 80s.

Relational Model Concepts

- represent a database as a collection of relations.
- each relation looks like a table of values.
- each row in the table represents a collection of related data values.
- each row in the table represents a fact that corresponds to a real-world entity or relationship
- the table name and the column name help to interpret the meaning of the values in each row.
- all values in a column are of the same data type.

Relational Model Terminology

- a row is called a tuple
- a column name is called an attribute
- a table is called a relation
- the data type of the values that can appear in a column is called a domain

tuples, attributes, relations, domains

A Domain D is a set of atomic values. (atomic means indivisible wrt the relational model)

To specify a domain D, we can specify a data type, from which the data values of the domain can be taken.

`Example:` **Some Domains**

- **Canadian/US phone numbers, set of valid 11-digit phone numbers**
- **Canadian SIN, set of valid SIN**
- **Names: Smith, Lopez ...**
- **Employee_Age: possible ages of employees of a company**
- **Dept names in a University: Computer Science, Mathematics, Chemistry ...**
- **Dept codes in a University: CS, MATH, CHEM ...**

these are logical definitions of domains.

a data type or format is also specified for each domain.

- Canadian/US phone numbers: 1(ddd)ddd-dddd
- Canadian SIN: ddd ddd ddd
- Employee_Age: integer $\in [\mathbf{18}, \mathbf{65}]$

A domain is equipped with a name, data type, format.

Often, additional information is required to interpret the values of the domain (e.g. distance in km or miles)

A relation schema $\mathbf{R}$, denoted by $\mathbf{R}(\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_n})$ is a relation name $\mathbf{R}$, and a list of attributes $\mathbf{R}(\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_n})$.

Each attribute $\mathbf{A_i}$ has a domain $\mathbf{dom}(\mathbf{A_i})$.

A relation schema describes a relation $\mathbf{R}$.

$\mathbf{R}$ is called the name of the relation.

The degree of the relation is the number of attributes $\mathbf{n}$.

STUDENT(Name, SSN, HomePhone, Address, OfficePhone, Age, GPA)

name? attributes? degree? domains of the attributes?

null values represent attributes whose values are unknown or do not exist.

each tuple in the relation STUDENT represents a particular student.

A relation $\mathbf{r}$ of the relation schema $\mathbf{R(A_1, A_2, \ldots, A_n)}$, also denoted by $\mathbf{r(R)}$, is a set of n-tuples $\mathbf{r = \{t_1, t_2, \ldots, t_m\}}$.

Each n-tuple $\mathbf{t}$ is an ordered list of $\mathbf{n}$ values $\mathbf{t = \langle v_1, v_2, \ldots, v_n \rangle}$ where each value $\mathbf{v_i \in dom(A_i)}$ for $\mathbf{i = 1, \ldots, n}$ or has the special value null.

The $\mathbf{i^{th}}$ value in an n-tuple $\mathbf{t}$, corresp. to the attribute $\mathbf{A_i}$, is referred to as $\mathbf{t[A_i]}$.

Alternative definition of a relation

A relation $\mathbf{r(R)}$ is a mathematical relation (subset of the cartesian product) of degree $\mathbf{n}$ on the domains $\mathbf{dom(A_i)}$.

$$\mathbf{r(R) \subset dom(A_1) \times \cdots \times dom(A_n)}$$

The cartesian product contains all possible combinations of values from the domains.

The total number of tuples in the cartesian product (provided finiteness) is

$$\mathbf{|dom(A_1)| \star \cdots \star |dom(A_n)|}$$

The relation contains only those valid tuples that represent a particular state of the real world. As the state of the real world changes, so does the relation.

The relation schema R does not change very often. (except when we need to add a new attribute).

Several attributes can have the same domain.

Attributes specify different roles/interpretations for the same domain. (e.g. phone numbers $\rightarrow$ HomePhone, OfficePhone)

Characteristics of relations

How does a relation differ from a file or a table?

- **Ordering of tuples in a relation**

  Tuples in a relation don't have any particular order.

  In a file or a table, there is a certain order.

  Many logical orders can be specified on a relation.

- **Ordering of values within a tuple**

  An n-tuple is an ordered list of n values.

  The ordering of values in a tuple (attributes in a relation schema) is important.

  The order is irrelevant, as long as the correspondence between attributes and values is known.

  This points to another definition of a relation, making the order of values in a tuple unnecessary.

▷ Values in the tuples

Each value in a tuple is an atomic value.

Composite and multivalued attributes are not allowed.

(1st NF assumption)

Multivalued attributes: represented by extra relations

Composite attributes: represented by their components

Special role of the `null` value (unknown, does not apply to, unavailable)

proposition to have many different types of null values

▷ Interpretation of a relation

relation schema == declaration or assertion.

the STUDENT relation asserts that ...

each tuple in the relation represents a fact
or a particular instance of the assertion.

some relations represent facts about entities/relationships.

the relational model represents facts uniformly as relations

relation schema == predicate (PROLOG, deductive DBs)

values in each tuple, satisfy the predicate.

Relational Model Notation

- relation schema R of degree n, $R(A_1, A_2, \ldots, A_n)$
- an n-tuple of a relation $r(R)$, $t = \langle v_1, v_2, \ldots, v_n \rangle$
- $t[A_i]$ and $t.A_i$ refer to the value v_i in t of attribute A_i
- $t[A_{i_1}, A_{i_2}, \ldots, A_{i_k}]$ and $t.(A_{i_1}, A_{i_2}, \ldots, A_{i_k})$ (where $A_{i_1}, A_{i_2}, \ldots, A_{i_k}$ is a list of k attributes from R) refer to the subtuple of values $t = \langle v_{i_1}, v_{i_2}, \ldots, v_{i_k} \rangle$ in t corresponding to the k attributes
- Q, R, S denote relation names
- q, r, s denote relation states
- t, u, v denote tuples
- the name of a relation schema, also indicates the current set of tuples in the relation (relation state)

- an attribute A can be prefixed by the relation name R in which it belongs using the dot notation $\mathrm{R.A}$

e.g. STUDENT.Name, STUDENT.SIN

- The same name can be used for 2 attributes in different relations

- All attribute names within one relation must be unique.