EJERCICIO Nº 8

Relational database model

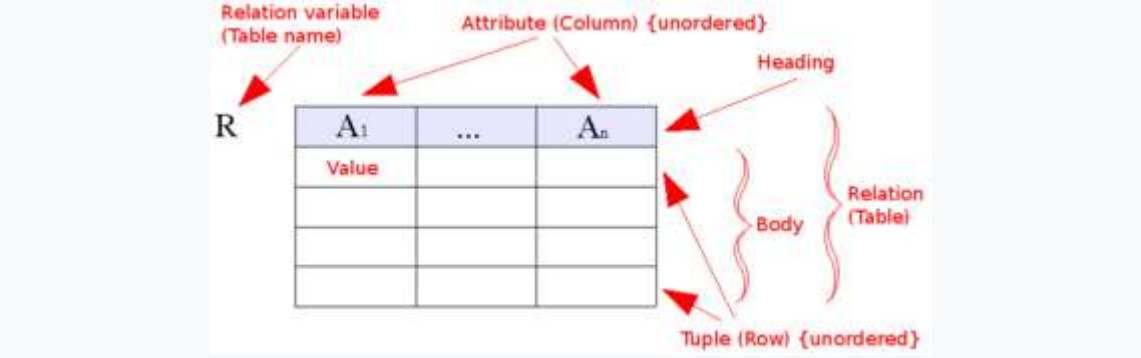
The **relational model** (**RM**) for database management is an approach to managing data using a structure and language consistent with first-order predicate logic, first described in 1969 by English computer scientist Edgar F. Codd, where all data is represented in terms of tuples, grouped into relations. A database organized in terms of the relational model is a relational database.

The purpose of the relational model is to provide a declarative method for specifying data and queries: users directly state what information the database contains and what information they want from it, and let the database management system software take care of describing data structures for storing the data and retrieval procedures for answering queries.

Most relational databases use the SQL data definition and query language; these systems implement what can be regarded as an engineering approximation to the relational model. A *table* in an SQL database schema corresponds to a predicate variable; the contents of a table to a relation; key constraints, other constraints, and SQL queries correspond to predicates. However, SQL databases deviate from the relational model in many details, and Codd fiercely argued against deviations that compromise the original principles.

Overview

The relational model's central idea is to describe a database as a collection of predicates over a finite set of predicate variables, describing constraints on the possible values and combinations of values. The content of the database at any given time is a finite (logical) model of the database, i.e. a set of relations, one per predicate variable, such that all predicates are satisfied. A request for information from the database (a database query) is also a predicate.



Relational model concepts.

Alternatives

Other models include the hierarchical model and network model. Some systems using these older architectures are still in use today in data centers with high data volume needs, or where existing systems are so complex

and abstract that it would be cost-prohibitive to migrate to systems employing the relational model. Also of note are newer object-oriented databases.

Implementation

There have been several attempts to produce a true implementation of the relational database model as originally defined by Codd and explained by Date, Darwen and others, but none have popular successes so far. As of October 2015, Rel is one of the more recent attempts to do this.

The relational model was the first database model to be described in formal mathematical terms. Hierarchical and network databases existed before relational databases, but their specifications were relatively informal. After the relational model was defined, there were many attempts to compare and contrast the different models, and this led to the emergence of more rigorous descriptions of the earlier models; though the procedural nature of the data manipulation interfaces for hierarchical and network databases limited the scope for formalization.

Structural database analytics employing relational modality protocols frequently employ data sequence differentials to maintain hierarchical architecture designations with incorporation of new input. These systems are functionally similar in concept to alternative relay algorithms, which form the foundation of cloud database infrastructure.

History

The relational model was developed by Edgar F. Codd as a general model of data, and subsequently promoted by Chris Date and Hugh Darwen among others. In their 1995 The Third Manifesto, Date and Darwen try to demonstrate how the relational model can accommodate certain "desired" object-oriented features.

Topics

The relational model of data permits the database designer to create a consistent, logical representation of information. Consistency is achieved by including declared constraints in the database design, which is usually referred to as the logical schema. The theory includes a process of database normalization whereby a design with certain desirable properties can be selected from a set of logically equivalent alternatives. The access plans and other implementation and operation details are handled by the DBMS engine, and are not reflected in the logical model. This contrasts with common practice for SQL DBMSs in which performance tuning often requires changes to the logical model.

The basic relational building block is the domain or data type, usually abbreviated nowadays to type. A tuple is an unordered set of attribute values. An attribute is an unordered pair of attribute name and type name. An attribute value is a specific valid value for the type of the attribute. This can be either a scalar value or a more complex type.

A relation consists of a heading and a body. A heading is a set of attributes. A body (of an n-ary relation) is a set of n-tuples. The heading of the relation is also the heading of each of its tuples.

A relation is defined as a set of n-tuples. In both mathematics and the relational database model, a set is an unordered collection of unique, non-duplicated items, although some DBMSs impose an order to their data. In mathematics, a tuple has an order, and allows for duplication. E. F. Codd originally defined tuples using this mathematical definition. Later, it was one of E. F. Codd's great insights that using attribute names instead of an ordering would be more convenient (in general) in a computer language based on relations. This insight is still being used today. Though the concept has changed, the name "tuple" has not. An immediate and important consequence of this distinguishing feature is that in the relational model the Cartesian product becomes commutative.

A table is an accepted visual representation of a relation; a tuple is similar to the concept of a row.

A relvar is a named variable of some specific relation type, to which at all times some relation of that type is assigned, though the relation may contain zero tuples.

The basic principle of the relational model is the Information Principle: all information is represented by data values in relations. In accordance with this Principle, a relational database is a set of relvars and the result of every query is presented as a relation.

The consistency of a relational database is enforced, not by rules built into the applications that use it, but rather by constraints, declared as part of the logical schema and enforced by the DBMS for all applications. In general, constraints are expressed using relational comparison operators, of which just one, "is subset of" (⊆), is theoretically sufficient [citation needed]. In practice, several useful shorthands are expected to be available, of which the most important are candidate key (really, superkey) and foreign key constraints.