

Codd's 12 rules

Codd's twelve rules^[1] are a set of thirteen rules (numbered zero to twelve) proposed by Edgar F. Codd, a pioneer of the relational model for databases, designed to define what is required from a database management system in order for it to be considered *relational*, i.e., a relational database management system (RDBMS).^{[2][3]} They are sometimes jokingly referred to as "Codd's Twelve Commandments".

Contents

Details

Rules

See also

References

Further reading

Details

Codd produced these rules as part of a personal campaign to prevent the vision of the original relational database from being diluted, as database vendors scrambled in the early 1980s to repackage existing products with a relational veneer. Rule 12 was particularly designed to counter such a positioning.

Rules

Rule 0: The *foundation rule*:

For any system that is advertised as, or claimed to be, a relational data base management system, that system must be able to manage data bases entirely through its relational capabilities.

Rule 1: The *information rule*:

All information in a relational data base is represented explicitly at the logical level and in exactly one way – by values in tables.

Rule 2: The *guaranteed access rule*:

Each and every datum (atomic value) in a relational data base is guaranteed to be logically accessible by resorting to a combination of table name, primary key value and column name.

Rule 3: *Systematic treatment of null values*:

Null values (distinct from the empty character string or a string of blank characters and distinct from zero or any other number) are supported in fully relational DBMS for representing missing information and inapplicable information in a systematic way, independent of data type.

Rule 4: *Dynamic online catalog based on the relational model*:

The data base description is represented at the logical level in the same way as ordinary data, so that authorized users can apply the same relational language to its interrogation as they apply

to the regular data.

Rule 5: The *comprehensive data sublanguage rule*:

A relational system may support several languages and various modes of terminal use (for example, the fill-in-the-blanks mode). However, there must be at least one language whose statements are expressible, per some well-defined syntax, as character strings and that is comprehensive in supporting all of the following items:

1. Data definition.
2. View definition.
3. Data manipulation (interactive and by program).
4. Integrity constraints.
5. Authorization.
6. Transaction boundaries (begin, commit and rollback).

Rule 6: The *view updating rule*:

All views that are theoretically updatable are also updatable by the system.

Rule 7: *Possible for high-level insert, update, and delete*:

The capability of handling a base relation or a derived relation as a single operand applies not only to the retrieval of data but also to the insertion, update and deletion of data.

Rule 8: *Physical data independence*:

Application programs and terminal activities remain logically unimpaired whenever any changes are made in either storage representations or access methods.

Rule 9: *Logical data independence*:

Application programs and terminal activities remain logically unimpaired when information-preserving changes of any kind that theoretically permit unimpairment are made to the base tables.

Rule 10: *Integrity independence*:

Integrity constraints specific to a particular relational data base must be definable in the relational data sublanguage and storable in the catalog, not in the application programs.

Rule 11: *Distribution independence*:

The end-user must not be able to see that the data is distributed over various locations. Users should always get the impression that the data is located at one site only.

Rule 12: The *nonsubversion rule*:

If a relational system has a low-level (single-record-at-a-time) language, that low level cannot be used to subvert or bypass the integrity rules and constraints expressed in the higher level relational language (multiple-records-at-a-time).

See also

- [IBM System R](#)

References

1. "[Codd's 12 Rules](#)". University of Derby. 2015-09-15. Retrieved September 6, 2018.
2. Codd, Edgar Frank (14 October 1985), "Is Your DBMS Really Relational?", *ComputerWorld*.
3. Codd, Edgar Frank (21 October 1985), "Does Your DBMS Run By the Rules", *ComputerWorld*.

Further reading

- Codd, Edgar F. (1990). *The relational model for database management: Version 2*. Addison-Wesley. ISBN 9780201141924.
- Harrington, Jan L. (2002). "Codd's Rules". *Relational Database Design Clearly Explained*. The Morgan Kaufmann Series in Data Management Systems (2nd ed.). Morgan Kaufmann. ISBN 9781558608207.
- Krishna, S. (1992). "Criteria for Evaluating Relational Database Systems". *Introduction to Database and Knowledge-Base Systems*. Computer Science. **28**. World Scientific. pp. 91 et seq. ISBN 9789810206192.