**Term list: Informatics session 2, Databases, v. 1.0.1**

ACID

See transaction.

Ad hoc query or report

A new database query to address a particular problem or question that is usually run just once or a few times against an existing database. Because the database already exists, the query must be fit to the database rather than the database being designed to support the query. Analytic queries are often ad hoc.

Analytic database

A database designed to support data exploration and analysis in which queries are not easily predictable. These databases usually have a flexible structure that is meant to support a variety of ad hoc query types rather than being optimized for particular repetative queries. Data are often moved from transactional databases (see Transactional database) to analytic databases when exploratory analysis is needed.

Back end

In a client-server database system, the back end is the database management software that does the data storage and processing. It runs remotely on a central server computer, receiving commands and returning data over the network to client computers.

BLOB

“Binary large object.” A large volume data type for database fields that traditionally is used for unstructured text narratives or images.

Cardinality

Numerical relationships between entities and data elements in a database, such as one to many, many to one, and many to many. For example, a patient may have many clinical encounters, but an encounter will have only one patient. A patient may have many physicians and a physician may have many patients (many to many). The need to support these data relationships partially defines the structure of databases.

Client-server

A software and hardware system design in which multiple user computers (clients, front end) communicate over a network with a centralized data store (back end) on a server computer. The software on the clients provides tools to work with the data, and the server software managing the data store communicates with the clients over the network to receive requests for data and respond to them.

Codd

E.F. Codd was a computer scientist who developed a set of 13 requirements that define relational databases. They are numbered 0 to 12 and so are sometimes referred to as “Codd’s 12 laws” even though there are 13. Consider it a baker’s dozen, or lagniappe.

CRUD

The basic database operations: create, retrieve, update, and delete. Also often uttered by people as they try to develop or work with databases.

Data mart or data cube

Data organized for analytic query (ie, an analytic database). Marts and cubes are generally topic- or question-oriented and smaller than data warehouses, and may be created as extracts of data warehouses. The term “cube” is jargon that implies a dimensional design with a central fact table and surrounding dimension tables, similar to the design of a data warehouse.

Data integrity

Accuracy, completeness, and consistency of a data set. In some settings, data integrity may also refer to the security of a data set, how resistent it is to inappropriate changes introduced inadvertantly, by software error, or by tampering.

Data dictionary

The list of data elements in a database. Usually includes definitions and required data types, and for relational databases it will generally indicate the table that contains the data element.

Data warehouse

An analytic database designed to integrate large amounts of data and perform correlations and other analytics operations efficiently. DW schemas are often star shaped (star schema), in which there is a central “fact table” that links a set of surrounding “dimension” tables and allows the dimensions to be correlated with each other. In a snowflake schema, the dimension tables are surrounded by subdimension tables and can act as mini-fact-tables themselves. Because of their design, data warehouses process analytical queries rapidly. But loading them requires updating the fact table and many dimension tables for each fact, and this can be time consuming. Therefore DW are often loaded periodically (eg, once a week or month) overnight. The design does not support efficiently daily operations of businesses where rapid updating in real time is required and thus transactional databases optimized for routine updates/queries are typically used for business operations. Data is then transferred periodically from one or more transactional systems to a data warehouse for integration and analysis.

Database

 A set of data elements organized into fields and records for efficient data storage and retrieval.

Database management system

A computer program (or set of related programs) that allows users and other systems to work with databases while maintaining the integrity of the data.

Database, Flat file

A single table data set, similar to a spreadsheet. Simple, but does not handle one to many and many to many relationships well. Subject to wasted space and redundancy. Flat file databases differ from spreadsheets in that each column (data element) must be a single data type such as integer, floating point number, or text string (this is true for databases in general).

Database, Hierarchical

A data set organized into a tree structure that supports one to many relationships. Can manage large data sets and is fast for adding and looking up data along its defined branching paths. Not efficient for data lookups that are not along the defined paths. Subject to redundancy.

Database, Network

A data set with custom-designed links between data elements. Supports one to many and many to many relationships, can handle large data sets, and can be fast. But must be custom designed for a purpose and is demanding to build and manage when data relationships are complex. Best for large data sets with simple relationships. Some are called “NoSQL” databases because they don’t use SQL, the standard query language of relational databases.

Database, Object-oriented

A database design often used for large, complex data types such as long documents or images.

Database, Relational

A database design consisting of multiple linked tables. Tables are related to each other by keys (unique row identifiers) that may be shared between tables. Because repeating data elements can be moved to their own tables, relational databases can substantially reduced redundancy and increase flexibility compared with flat file or hierarchicial databases, and they can support one-to-many, many-to-one, and many-to-many relationships. Relational databases also support a common query language, Structured Query Language (SQL) and may be transactional. There are a number of other characteristics of true relational databases, as defined by E.F. Codd.

Dimension table

An outlying table in a data warehouse with a star schema. Entries in dimension tables are connected with items in the central fact table. If the fact table contained clinical encounters, for example, the dimension tables would include the specifics of those encounters, such as patient, physician, orders, lab results, etc. This structure allows efficient database queries that correlate the dimensions with each other.

Discrete data

Data that is well defined and controlled. Typically a specific numerical type or a member of a defined set of allowable textual values. Contrast with uncontrolled free text that can take any textual value.

Entity

Objects that database records represent or are collected about. For example, an EHR might have patients, physicians, encounters, and lab tests as entities among many others. In relational databases, tables represent entities and each row (record) represents an instance of the entity.

ER diagram

Entity-Relation diagram. A diagram of the schema of a relational database in which the tables are represented by boxes that contain a list of the table’s column names and their data types. The tables are connected by lines that represent relations between the tables (ie, the shared keys).

Fact table

The central organizing table in a data warehouse star schema. The fact table represents the event or entity that the data warehouse tracks, such as clinical encounters or retail purchases, and is connected to outlying dimension tables containing the specific data describing those events or entities.

Field

An individual data element in a database. These are usually basic data types such as numbers or short text but may sometimes be long unstructured data contained in BLOBs. In a relational database, the fields are the individual data elements (columns) in a table row.

Free text

Uncontrolled textual data that can take any textual value.

Front end

In a client-server database system, the front end is the client software running on a computer that usually is used by people to communicate over a network with database management software running on a central server.

Index

In typical database usage, a listing of the values of a data element from a set of records that can be searched quickly to find associated records. The values in the index are linked to the records containing them to allow retrieval. For example, an index of ICD10 diagnosis codes in patient visit encounters could be built that allowed rapid retrieval of all patient records with a particular diagnosis code. Indexes are sorted to allow rapid and efficient searches that do not require all elements of the index to be tested for a match to the search term.

Key, foreign

A data element (column) in a relational database table that is the primary key in another table. Foreign keys establish a data relation between the two tables.

Key, primary

A data element (column) in a relational database table that uniquely identifies each row in the table.

Metadata

Data about data. This may include provenance (e.g., where the data came from and how it has been handled) or context (e.g., information about how data was obtained). What constitutes primary data and what is metadata is often a matter of perspective and can be related to the question being asked of the data set.

MUMPS

Massachusetts General Hospital Utility Multi-Programming System. MUMPS is a database programming language and management system developed for medical information systems in the 1960’s at Mass General Hospital. It is fast, efficient, reliable, and widely used in healthcare and finance. MUMPS databases are essentially hierarchical in form, but MUMPS has developed over the years to have features that mitigate some of the limitations of hierarchical databases. The portions of Epic that support daily operations (ie, high speed, real time update and retrieval of small amounts of data) are written in MUMPS.

Normal form

The degree to which a relational database schema eliminates redundancy and data dependencies. The highest normal form has no data redundancy and all data elements are independent of each other. For a variety of practical reasons, including updating and query speed, most relational databases are reasonably but not completely normalized.

Overloading

The practice of combining several data elements into a single field in a database. “Chaining” refers to concatenating several text data elements to yield an overloaded text string for storage in one field. The disadvantage to this is that the individual data elements cannot be searched or managed individually by the database management system. Instead, the whole string must be retrieved and then broken down by a separate program to evaluate the pieces. There should be careful consideration of the benefits vs. the disadvantages of overloading before using it.

Query

A database search. Database management systems provide query capabilities as part of their routine operations. Sophisticated indexing and searching methods allow efficient searching of numerical and character (text) data types. Free text searching is possible but requires specialized methods. Relational databases have a standardized query language, SQL. Other types of databases have query languages or tools that are more or less specific to the form of database and/or developer.

RDF triple store

A database that stores Resource Description Framework (RDF) triples. RDF triples are made up of three data elements representing two concepts (a subject and object) and the relationship between those concepts (a predicate). Triple stores are usually implemented as network databases where the nodes represent subjects and objects, and the connecting lines represent predicates. They are often used in working with ontologies and using ontologies for reasoning.

Record

A set of data elements (fields) that are meaningful together. A paper form designed for a particular purpose would be a record once it’s filled out. The individual blanks in the form would be fields. In databases, the data elements that describe a specific entity contained in the database (ie, a particular patient, clinical encounter, or lab test) are records. In a relational database, a row in a table is considered a record.

Record tables

Database tables in LIS and APLIS that contain operational data. Operational data includes orders, patient and QC results, specimen tracking events, etc. This is LIS jargon and refers to patient and operational data as distinct from configuration data stored in maintenance tables, so don’t confuse it with the general database concept of a record. The use of “table” in this context is general and doesn’t necessarily refer to a relational database table (though it could).

Relation

A relationship between two tables in a relational database established by sharing a data element (column) that is a primary key in one table and a foreign key in the other table.

Report

A database query result, usually formatted for convenient use by people. Reports are often tabular (rows and columns) but may also be a set of records with a more complex layout.

Retrieval

The ability of a database to return a set of records matching search terms.

SaaS

Software as a Service. A business arrangement whereby a user or organization purchases network access to software and hardware that is operated and managed by a vendor in the vendor’s data center. Communications may be encrypted for security purposes (ie, a virtual private network).

Search

 See Query.

Schema

The structure of a database. In a hierarchical database, the schema shows the data elements and the tree structure that connects them. In a relational database, the schema shows the tables/columns and relations in the database, often as an ER diagram. In a network database, the schema shows the data elements and their interconnections.

Sort

Putting a set of data elements or records into a defined sequence or order. For records, sorting is usually based on a defined data element. Databases use a variety of efficient sorting algorithms depending on the data being sorted and the goal of the sort.

SQL

Structured Query Language. The relatively standardized query and management language of relational databases. SQL can perform all the CRUD operations of a relational database, can create and delete databases and tables, and can manage users and user access.

Table

The core data storage structure in a relational database. Tables have rows containing records and columns containing the individual data elements of each record (fields). The data type of each column (field) is defined and fixed, eg, columns must contain all integers or strings, not a mixture of both. One column of each table contains a unique identifier for rows in that table, called the primary key. Tables may also have columns that contain foreign keys, which are primary keys from other tables. Tables are searched and sorted based on the values contained in the columns.

Thin client

In a client-server system, a thin client is a generic software “shell” on a client machine that communicates with a server to receive a user display or downloads software that runs on the client machine to create the user display (the “front end” of the client-server system). Once the display is initiated, it can communicate over the network with a database back end to carry out data management, processing, and display tasks for the client with limited or no data processing on the client. The advantage of a thin client is that client machines automatically receive the most current front end of the client server system each time they log in, so maintenance related to installing software upgrades and fixing configuration problems is reduced. Many modern systems use a Web browser as a thin client.

Transaction

A transaction is a complete database operation, such as an update, a deletion, or a query. In relational databases these operations may involve multiple tables and actions. If all actions are not completed (eg, because of a hardware failure or power outage at just the wrong time) or if several operations collide (eg, try to change the same data elements without control), the operation or the database itself could be left in an incomplete or erroneous state. To avoid this, transactional databases have the capability to verify that an operation has been completed successfully before it is “committed” or made part of the database. If the full operation does not complete successfully, the database will “roll back” to the starting point of the operation and indicate that the transaction failed (and could be re-attempted from the start if desired). Transactions are said to have ACID properties. This means that to viewers external to the database transactions appear to be a single action (Atomic), they result in accurate database changes (Consistent), they do not interfere with each other even at high data processing rates (Independent), and that once completed they are preserved (Durable).

Transactional database

Typically, a database that supports routine business operations efficiently and reliably using ACID transaction properties (see Transaction). Transactional databases are optimized to manage high volumes of small, pre-defined updates and queries. Because they are highly optimized for specific operations, transactional databases are usually less efficient at supporting ad hoc queries for analytics, and thus data is often moved from transactional to analytic databases to support analysis (see analytic database).

Tree structure

The basic schema of a hierarchical database in which each data element is connected to 0 or more underlying data elements (one-to-many relationships). Hierarchical databases are usually shown as inverted trees, with the trunk on top and branches extending out below (or think of roots if you like). Tree structures are very fast for update and retrieval of data along the defined pathways (eg, all the lab tests from last week’s encounter), but tend to be slow for other kinds of query (eg, all elevated troponins across all patients) and have difficulty representing many-to-one and many-to-many relationships.



© 2023 James H. Harrison, Jr. This glossary is made available under a [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/). The contents may be used without restriction, modified, and redistributed if they are attributed and any changes are indicated.