A

access plan—A set of instructions, generated at application compilation time, that is created and managed by a DBMS. The access plan predetermines the way an application's query will access the database at run time.

active data dictionary—A data dictionary that is automatically updated by the database management system every time the database is accessed, thereby keeping its information current. See also *data dictionary*.

ActiveX—Microsoft's alternative to Java. A specification for writing programs that will run inside the Microsoft client browser (Internet Explorer). Oriented mainly to Windows applications, it is not portable. It adds "controls" such as drop-down windows and calendars to Web pages.

ActiveX Data Objects (ADO)—A Microsoft object framework that provides a high-level application-oriented interface to interact with OLE-DB, DAO, and RDO. ADO provides a unified interface to access data from any programming language that uses the underlying OLE-DB objects.

ad hoc query—A "spur-of-the-moment" question.

ADO.NET—The data access component of Microsoft's .NET application development framework. The Microsoft .NET framework is a component-based platform for developing distributed, heterogeneous, and interoperable applications aimed at manipulating any type of data over any network under any operating system and programming language.

Aggregation – This is where larger entity can be composed of smaller entities. Aggregation, along with **composition** is used to represent "has_a" or "part_of" relationships between entities.

alias—An alternative name given to a column or table in any SQL statement.

ALTER—The SQL command used to make changes to table structure. Followed by a keyword (ADD or MODIFY), it adds a column or changes column characteristics.

American National Standards Institute (ANSI)—The group that accepted the DBTG recommendations and augmented database standards in 1975 through its SPARC committee.

AND—The SQL logical operator used to link multiple conditional expressions in a WHERE or HAVING clause. It requires that all conditional expressions evaluate to true.

anonymous PL/SQL block—A PL/SQL block that has not been given a specific name.

application processor—See *transaction processor (TP)*.

application programming interface (API)—Software through which programmers interact with middleware. Allows the use of generic SQL code, thereby allowing client processes to be database server-independent.

association - an association is a relationship between entities.

association class - An association class is used to represent a *:* relationship in UML class diagrams.

atomic attribute—An attribute that cannot be further subdivided to produce meaningful components. For example, a person's last name attribute cannot be meaningfully subdivided into other name components; therefore, the last name attribute is atomic.

atomicity—See *atomic transaction property*.

atomic transaction property—A property of transactions that states that all parts of a transaction must be treated as a single logical unit of work in which all operations must be completed (committed) to produce a consistent database.

attribute—A characteristic of an entity or object. An attribute has a name and a data type.

attribute domain—See *domain*.

attribute hierarchy—Provides a top-down data organization that is used for two main purposes: aggregation and drill-down/roll-up data analysis.

audit log—A database management system security feature that automatically records a brief description of the database operations performed by all users.

authentication—The process through which a DBMS verifies that only registered users are able to access the database.

authorization management—All activities that are dedicated to enforcing the security, privacy, and integrity of the data in a database.

automatic query optimization—A method by which a DBMS takes care of finding the most efficient access path for the execution of a query.

AVG—A SQL aggregate function that outputs the mean average for the specified column or expression.

B

b-tree - A B-tree consists of a hierarchy of nodes which contain a set of pointers that link the nodes of the B-tree together. They are used to maintain an ordered set of indexes or data to allow efficient operations to select, delete and insert data.

back-end CASE tools—A computer-aided software tool that has been classified as "back-end" because it provides support for the coding and implementation phases of the SDLC. In comparison, front-end case tools provide support for the planning, analysis, and design phases.

base table—The table on which a view is based.

batch update routine—A routine that pools transactions into a single "batch" to update a master table in a single operation.

binary lock—A lock that has only two states: *locked* (1) and *unlocked* (0). If a data item is locked by a transaction, no other transaction can use that data item. See also *lock*.

binary relationship—An ER term used to describe an association (relationship) between two entities. Example: LECTURER teaches COURSE.
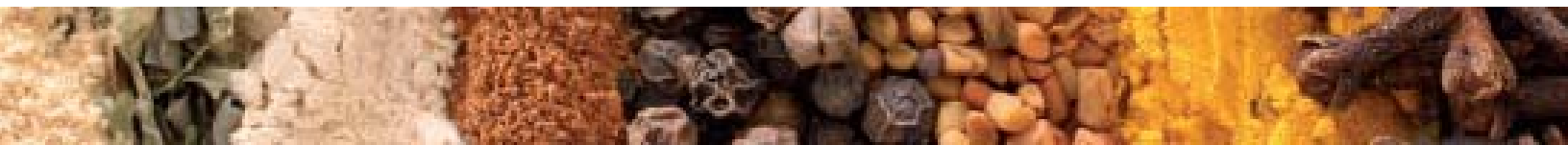
bitmap index - A popular type of index, based on a two-dimensional array, which is often used on multi-dimensional data held in data warehouses.

Boolean algebra—A branch of mathematics that deals with the use of the logical operators OR, AND, and NOT.

bottom-up design—A design philosophy that begins by identifying individual design components and then aggregates those components into larger units. In database design, it is a process that begins by defining attributes and then groups those attributes into entities. Compare to *top-down design*.

boundaries—The external limits to which any proposed system is subjected. These include budget, personnel, and existing hardware and software.

Boyce-Codd normal form (BCNF)—A special form of third normal form (3NF) in which every determinant is a candidate key. A table that is in BCNF must be in 3NF. See also *determinant*.

bridge entity—See *composite entity*.

buffer—See *buffer cache*.

buffer cache—A shared, reserved memory area that stores the most recently accessed data blocks in RAM. Also called *data cache*. Used to take advantage of a computer's fast primary memory compared to the slower secondary memory, thereby minimizing the number of input/output (I/O) operations between the primary and secondary memories.

business rule—Narrative descriptions of a policy, procedure, or principle within an organization. Examples: A pilot cannot be on duty for more than 10 hours during a 24-hour period. A lecturer may teach up to four classes during any one semester.

bus topology—Network topology requiring that all computers be connected to a main network cable. Usually implemented using coaxial cable; it bears the disadvantage that a single lost computer can result in network segment breakdown.

C

Call Level Interface (CLI)—A standard developed by the SQL Access Group for database access.

candidate key—See *key*.

cardinality—Assigns a specific value to connectivity. Expresses the range (minimum to maximum) of allowed entity occurrences associated with a single occurrence of the related entity.

CARTESIAN PRODUCT - The CARTESIAN PRODUCT is usually written as $R_1 \; X \; R_2$ with the new resulting relation R_3 containing all the attributes which are present in R_1 and R_2 along with all the possible combinations of tuples from both R_1 and R_2.

cascading order sequence—Refers to a nested ordering sequence for a set of rows. For example, a list in which all last names are alphabetically ordered and, within the last names, all first names are ordered represents a cascading sequence.

CASE—See *computer-assisted software engineering (CASE)*.

centralized database—A database located at a single site.
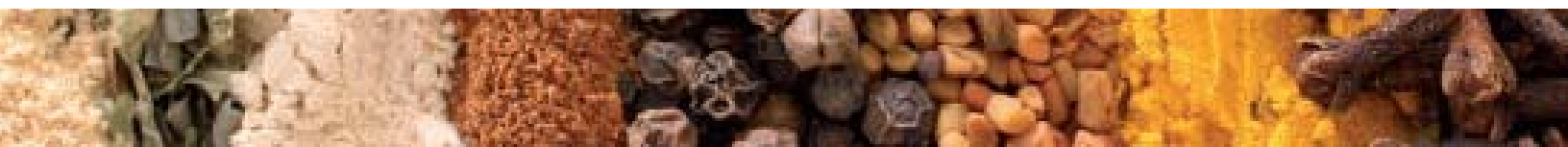
centralized design—A process in which a single conceptual design is modeled to match an organization's database requirements. Typically used when a data component consists of a relatively small number of objects and procedures. Compare to *decentralized design*.

checkpoint—In transaction management, an operation in which the database management system writes all of its updated buffers to disk.

Chen model—See *entity relationship (ER) model*.

class—A collection of like objects with shared structure (attributes) and behavior (methods). A class encapsulates an object's data representation and a method's implementation. Classes are organized in a class hierarchy.

class diagram – The class diagram is a component of the **Unified Modeling Language (UML)** which can be used to produce entity relationship models.

class hierarchy—The organization of classes in a hierarchical tree where each "parent" class is a *superclass* and each "child" class is a *subclass*. See also *inheritance.*

client/server architecture—Refers to the arrangement of hardware and software components to form a system composed of clients, servers, and middleware. The client/server architecture features a user of resources, or a client, and a provider of resources, or a server.

client-side extensions—These extensions add functionality to a Web browser. Although available in various forms, the most commonly encountered extensions are plug-ins, Java, JavaScript, ActiveX, and VBScript.

closure—A property of relational operators that permits the use of relational algebra operators on existing tables (relations) to produce new relations.

cluster - Usually tables which share common columns are used in the same queries and hence they are physically clustered together to increase the speed of data retrieval. The cluster key is a field, or set of fields, that the clustered tables have in common which is identified usually through the table join. The cluster key is determined when the cluster is created.

cohesivity—The strength of the relationships between a module's components. Module cohesivity must be high.

COMMIT—The SQL command that permanently saves data changes to a database.
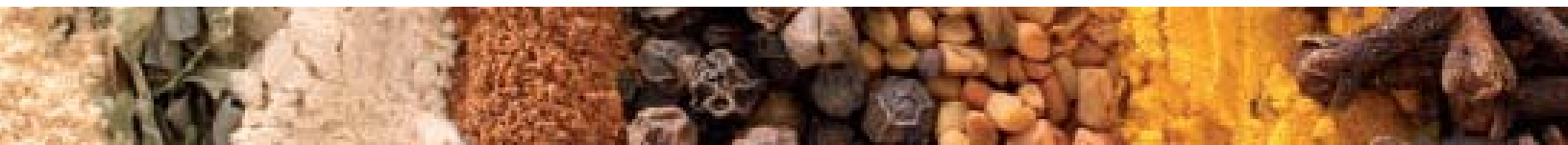
Common Gateway Interface (CGI)—A Web server interface standard that uses script files to perform specific functions based on a client's parameters.

completeness constraint—A constraint that specifies whether each entity supertype occurrence must also be a member of at least one subtype. The completeness constraint can be partial or total. Partial completeness means that not every supertype occurrence is a member of a subtype; that is, there may be some supertype occurrences that are not members of any subtype. Total completeness means that every supertype occurrence must be a member of at least one subtype.

composite attribute—An attribute that can be further subdivided to yield additional attributes. For example, a phone number (615-898-2368) may be divided into an area code (615), an exchange number (898), and a four-digit code (2368). Compare to *simple attribute*.

composite entity—An entity designed to transform an *:* relationship into two 1:* relationships. The composite entity's primary key comprises at least the primary keys of the entities that it connects. Also known as a *bridge entity*. See also *linking table.*

composite key—A multiple-attribute key.

composition – A special case of aggregation is known as composition. This is a much stronger relationship than aggregation as when the parent entity instance is deleted; all child entity instances are automatically deleted. Composition, along with **aggregation** is used to represent "has_a" or "part_of" relationships between entities.
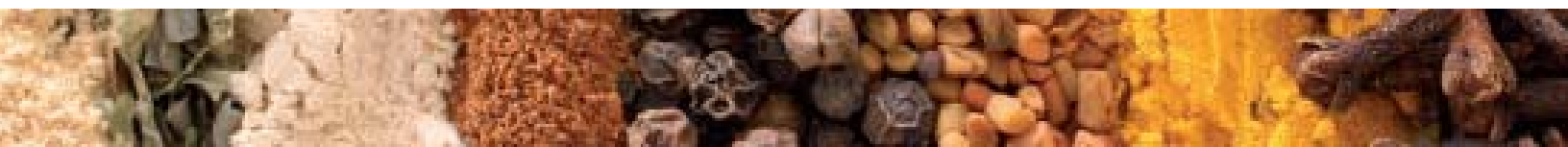
composite usage map – a simplified version of an ERD used to show data volume and data usage statistics.

computer-assisted software engineering (CASE)—Tools used to automate part or all of the Systems Development Life Cycle.

conceptual design—A process that uses data modeling techniques to create a model of a database structure that represents the real-world objects in the most realistic way possible. Both software- and hardware-independent.

conceptual model—The output of the conceptual design process. The conceptual model provides a global view of an entire database. Describes the main data objects, avoiding details.

conceptual schema—A representation of the conceptual model, usually expressed graphically. See also *conceptual model.*

concurrency control—A DBMS feature that is used to coordinate the simultaneous execution of transactions in a multiprocessing database system while preserving data integrity.

concurrent backup—A backup that takes place while one or more users are working on a database.

Conference on Data Systems Languages (CODASYL)—A group originally formed to help standardize COBOL; its DBTG subgroup helped to develop database standards in the early 1970s.

connectivity—Describes the classification of the relationship between entities. Classifications include 1:1, 1:*, and *:*.

consistency—A database condition in which all data integrity constraints are satisfied. To ensure consistency of a database, every transaction must begin with the database in a known consistent state. If the database is not in a consistent state, the transaction will yield an inconsistent database that violates its integrity and business rules.

consistent database state—A database state in which all data integrity constraints are satisfied.

constraint—A restriction placed on data. Constraints are normally expressed in the form of rules. Example: "A student's average must be between 0.00 and 100." Constraints are important because they help to ensure data integrity.

coordinator—The transaction processor (TP) node that coordinates the execution of a two-phase COMMIT in a DDBMS. See also *data processor (DP), transaction processor (TP),* and *two-phase commit protocol*.

correlated subquery—A subquery that executes once for each row in the outer query.

cost-based optimizer—A query optimizer technique that uses an algorithm based on statistics about the objects being accessed, that is, number of rows, indexes available, indexes sparsity, and so on.

COUNT—A SQL aggregate function that outputs the number of rows containing not null values for a given column or expression, sometimes used in conjunction with the DISTINCT clause.

CREATE INDEX—A SQL command that creates indexes on the basis of any selected attribute or attributes.

CREATE TABLE—A SQL command used to create a table's structures, using the characteristics and attributes given.

CREATE VIEW—A SQL command that creates a logical, "virtual" table that can be treated as a "real" table.

cross join—A join that performs a relational product (also known as the Cartesian product) of two tables.

Crow's Foot notation—A representation of the entity relationship diagram using a three-pronged symbol to represent the "many" sides of the relationship.

cube cache—In multidimensional OLAP, refers to the shared, reserved memory area where data cubes are held. Using the cube cache assists in speeding up data access.

cursor—A special construct used in procedural SQL to hold the data rows returned by a SQL query. A cursor may be thought of as a reserved area of memory in which the output of the query is stored, like an array holding columns and rows. Cursors are held in a reserved memory area in the DBMS server, not in the client computer.

D

data—Raw facts, that is, facts that have not yet been processed to reveal their meaning to the end user.

Data Access Objects (DAO)—An object-oriented API (application programming interface) used to access MS Access, MS FoxPro, and dBase databases (using the Jet data engine) from Visual Basic programs. DAO provides an optimized interface that exposes the functionality of the Jet data engine (on which MS Access database is based) to programmers. The DAO interface can also be used to access other relational style data sources.

data administrator (DA)—The person responsible for managing the entire data resource, whether computerized or not. The DA has broader authority and responsibility than the database administrator (DBA). Also known as an *information resource manager (IRM)*.

data allocation—In a distributed DBMS, describes the process of deciding where to locate data fragments.

data anomaly—A data abnormality that exists when inconsistent changes to a database have been made. For example, an employee moves, but the address change is corrected in only one file and not across all files in the database.

database—A shared, integrated computer structure that houses a collection of related data. A database contains two types of data: end-user data (raw facts) and metadata. The metadata consist of data about data, that is, the data characteristics and relationships.

database administrator (DBA)—The person responsible for planning, organizing, controlling, and monitoring the centralized and shared corporate database. The DBA is the general manager of the database administration department.

database design—The process that yields the description of the database structure. The database design process determines the database components. Database design is the second phase of the database life cycle.

database development—A term used to describe the process of database design and implementation.

database fragments—Subsets of a distributed database. Although the fragments may be stored at different sites within a computer network, the set of all fragments is treated as a single database. See also *horizontal fragmentation* and *vertical fragmentation.*

database instance—In an Oracle DBMS, refers to the collection of processes and data structures used to manage a specific database.

database-level lock—A type of lock that restricts database access to only the owner of the lock. It allows only one user at a time to access the database. Successful for batch processes, but unsuitable for online multiuser DBMSs.

database life cycle (DBLC)—The history of a database within an information system. Divided into six phases: initial study, design, implementation and loading, testing and evaluation, operation and maintenance, and evolution.

database management system (DBMS)—Refers to the collection of programs that manages the database structure and controls access to the data stored in the database.

database middleware—Database connectivity software through which application programs connect and communicate with data repositories.

database object—Any object in a database, such as a table, a view, an index, a stored procedure, or a trigger.

database performance tuning—A set of activities and procedures designed to reduce the response time of a database system, that is, to ensure that an end-user query is processed by the DBMS in the minimum amount of time.

database recovery—The process of restoring a database to a previous consistent state.

database request—The equivalent of a single SQL statement in an application program or a transaction.

database security officer (DSO)—Responsible for the security, integrity, backup, and recovery of the database.

database system—An organization of components that defines and regulates the collection, storage, management, and use of data in a database environment.

database task group (DBTG)—A CODASYL committee that helped develop database standards in the early 1970s. See also *Conference on Data Systems Languages (CODASYL)*.

data cache—A shared, reserved memory area that stores the most recently accessed data blocks in RAM. Also called *buffer cache*.

data cube—Refers to the multidimensional data structure used to store and manipulate data in a multidimensional DBMS. The location of each data value in the data cube is based on the x-, y-, and z-axes of the cube. Data cubes are static (must be created before they are used), so they cannot be created by an ad hoc query.

data definition language (DDL)—The language that allows a database administrator to define the database structure, schema, and subschema.

data dependence—A data condition in which the data representation and manipulation are dependent on the physical data storage characteristics.

data dictionary—A DBMS component that stores metadata—data about data. Thus, the data dictionary contains the data definition as well as its characteristics and relationships. A data dictionary may also include data that are external to the DBMS. Also known as an *information resource dictionary.* See also *active data dictionary, metadata,* and *passive data dictionary*.

Data Encryption Standard (**DES**)—The most widely used standard for private-key encryption. DES is used by the U.S. government.

data extraction—A process used to extract and validate data taken from an operational database and external data sources prior to their placement in a data warehouse.

datafile—See *data files*.

data files—A named physical storage space that stores a database's data. It can reside in a different directory on a hard disk or on one or more different hard disks. All data in a database are stored in data files. A typical enterprise database is normally composed of several data files. A data file can contain rows from one table, or it can contain rows from many different tables.

data filtering—See *data extraction*.

data fragmentation—A characteristic of a DDBMS that allows a single object to be broken into two or more segments or fragments. The object might be a user's database, a system database, or a table. Each fragment can be stored at any site over a computer network.

data inconsistency—A condition in which different versions of the same data yield different (inconsistent) results.

data independence—A condition that exists when data access is unaffected by changes in the physical data storage characteristics.

data integrity—A condition in which given data always yield the same result. Data integrity is mandatory in any database. In a relational database, refers to the characteristic that allows a DBMS to maintain entity and referential integrity.

data management—A process that focuses on data collection, storage, and retrieval. Common data management functions include addition, deletion, modification, and listing.

data manager (DM)—See *data processing (DP) manager*.

data manipulation language (DML)—The language (set of commands) that allows an end user to manipulate the data in the database (SELECT, INSERT, UPDATE, DELETE, COMMIT, and ROLLBACK).

data mart—A small, single-subject data warehouse subset that provides decision support to a small group of people.

data mining—A process that employs automated tools to analyze data in a data warehouse and other sources and to proactively identify possible relationships and anomalies.

data model—A representation, usually graphic, of a complex "real-world" data structure. Data models are used in the database design phase of the database life cycle.

data processing (DP) manager—A DP specialist who evolved into a department supervisor. Roles include managing the technical and human resources, supervising the senior programmers, and troubleshooting the program. Also known as a *data manager (DP)*.

data processing (DP) specialist—A now obsolete position formed in the conversion from manual filing systems to computer filing systems; once filled by an employee who created and programmed the necessary file structures, wrote the software that managed the data in those structures, and designed the application programs that produced reports from the file data.

data processor (DP)—The software component residing on a computer that stores and retrieves data through a DDBMS. The DP is responsible for managing the local data in the computer and coordinating access to that data. See also *transaction processor (TP)*.

data redundancy—A condition that exists when a data environment contains redundant (unnecessarily duplicated) data.

data replication—The storage of duplicated database fragments at multiple sites on a DDBMS. Duplication of the fragments is transparent to the end user. Used to provide fault tolerance and performance enhancements.

DataSet—In ADO.NET, refers to a disconnected memory-resident representation of the database. That is, the DataSet contains tables, columns, rows, relationships, and constraints.

data source name (DSN)—Identifies and defines an ODBC data source.

data sparsity—A column distribution of values or the number of different values a column could have.

data store—The component of the decision support system that acts as a database for storage of business data and business model data. The data in the data store has already been extracted and filtered from the external and operational data and will be stored for access by the end-user query tool for the business data model.

data warehouse—Bill Inmon, the acknowledged "father of the data warehouse," defines the term as "an integrated, subject-oriented, time-variant, nonvolatile collection of data that provides support for decision making."

DBMS performance tuning—Refers to the activities required to ensure that clients' requests are responded to in the fastest way possible, while making optimum use of existing resources.

deadlock—A condition that exists when two or more transactions wait indefinitely for each other to release the lock on a previously locked data item. Also called *deadly embrace.* See also *lock*.

deadly embrace—See *deadlock*.

decentralized design—A process in which conceptual design is used to model subsets of an organization's database requirements. After verification of the views, processes, and constraints, the subsets are then aggregated into a complete design. Such modular designs are typical of complex systems in which the data component consists of a relatively large number of objects and procedures. Compare to *centralized design*.

decision support system (DSS)—An arrangement of computerized tools used to assist managerial decision making within a business.

deferred update—When transaction operations do not immediately update a physical database. Also called *deferred write*.

deferred write—See *deferred update*.

DELETE—A SQL command that allows specific data rows to be deleted from a table.

denormalization—A process by which a table is changed from a higher level normal form to a lower level normal form. Usually done to increase processing speed. Potentially yields data anomalies.

dependency diagram—A representation of all data dependencies within a table.

derived attribute—An attribute that does not physically exist within the entity and is derived via an algorithm. Example: Age = current date – birth date.

description of operations—A document that provides a precise, detailed, up-to-date, and thoroughly reviewed description of the activities that define an organization's operating environment.
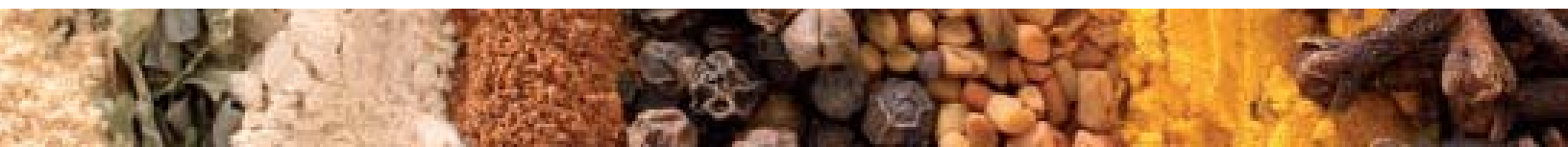
design trap—Occurs when a relationship is improperly or incompletely identified and, therefore, is represented in a way that is not consistent with the real world. The most common design trap is known as a *fan trap.*

desktop database—A single-user database that runs on a personal computer.

determinant—Any attribute in a specific row whose value directly determines other values in that row. See also *Boyce-Codd normal form (BCNF)*.

determination—The role of a key. In the context of a database table, the statement "A determines B" indicates that knowing the value of attribute A means that (determine) the value of attribute B can be looked up (determined).

DIFFERENCE - The DIFFERENCE operator denoted as $Relation_1$ - $Relation_2$, returns all tuples in one relation that are not found in the other relation; that is, it subtracts one relation from the other. The two relations must be union-compatible.

differential backup—A level of database backup in which only the last modifications to the database (when compared with a previous full backup copy) are copied.

dimensions—Qualifying characteristics that provide additional perspectives to a given fact.

dimension tables—In a data warehouse, used to search, filter, or classify facts within a star schema. The fact table is in a one-to-many relationship with dimension tables.
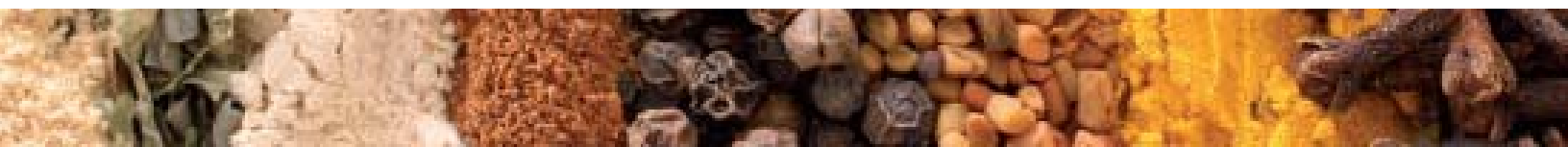
disaster management—The set of DBA activities dedicated to securing data availability following a physical disaster or a database integrity failure.

disjoint subtype (nonoverlapping subtype)—A unique and nonoverlapping subtype entity set.

DISTINCT—A SQL clause designed to produce a list of only those values that are different from one another.

distributed database—A logically related database that is stored over two or more physically independent sites.

distributed database management system (DDBMS)—A DBMS that supports a database distributed across several different sites; governs the storage and processing of logically related data over interconnected computer systems in which both data and processing functions are distributed among several sites.

distributed data catalog (DDC)—A data dictionary that contains the description (fragment names, locations) of a distributed database. Also known as a *distributed data dictionary (DDD)*.

distributed data dictionary (DDD)—See *distributed data catalog*.

distributed global schema—The database schema description of a distributed database as seen by the database administrator.

distributed processing—The activity of sharing (dividing) the logical processing of a database over two or more sites connected by a network.

distributed request—A database request that allows a single SQL statement to access data in several remote DPs in a distributed database.

distributed transaction—A database transaction that accesses data in several remote DPs in a distributed database.

distribution transparency—A DDBMS feature that allows a distributed database to be treated as though it were a single logical database.

DIVIDE - Selects tuples in *Relation$_1$* that match every row in *Relation$_2$*, denoted as by *Relation$_1$ ÷ Relation$_2$*

document type definition (DTD)—A file with a .dtd filename extension that describes XML elements; in effect, a DTD file provides the description of a document's composition and defines the syntax rules or valid tags for each type of XML document.

domain—Used to organize and describe an attribute's set of possible values.

DO-UNDO-REDO protocol—Used by a DP to roll back and/or roll forward transactions with the help of a system's transaction log entries.

drill down—To decompose data into more atomic components, that is, data at lower levels of aggregation. Used primarily in a decision support system to focus on specific geographic areas, business types, and so on. See also *roll up*.

DROP—A SQL command used to delete database objects such as tables, views, indexes, and users.

durability—The transaction property indicating the permanence of a database's consistent state. Transactions that have been completed will not be lost in the event of a system failure if the database has proper durability.

dynamic-link libraries (DLLs)—Shared code libraries that are treated as part of the operating system or server process so they can be dynamically invoked at run time.

dynamic query optimization—Refers to the process of determining the SQL access strategy at run time, using the most up-to-date information about the database. Contrast with *static query optimization*.

dynamic SQL—A term used to describe an environment in which the SQL statement is not known in advance, but instead is generated at run time. In a dynamic SQL environment, a program can generate the SQL statements at run time that are required to respond to ad hoc queries.

dynamic statistical generation mode—The distributed database management system will automatically evaluate and update the statistical information after each access.

dynamic-link libraries (DLLs)—Shared code modules that are treated as part of the operating system or server process so they can be dynamically invoked at run time.

E

EER diagram (EERD)—Refers to the entity-relationship diagram resulting from the application of extended entity relationship concepts that provide additional semantic content in the ER model.

embedded SQL—A term used to refer to SQL statements that are contained within an application programming language such as COBOL, C++, ASP and Java.

encryption – a method used to transform the data in the message into an encrypted message through the use of an encryption algorithm and an encryption key(s).

end-user presentation tool—Used by a data analyst to organize and present selected data compiled by the end-user query tool.

end-user query tool—Used by a data analyst to create the queries that access the specific desired information from the data store.

enterprise database—The overall company data representation, which provides support for present and expected future needs.

entity—Something about which someone wants to store data; typically a person, a place, a thing, a concept, or an event. See also *attribute*.

entity cluster—A "virtual" entity type used to represent multiple entities and relationships in the ERD. An entity cluster is formed by combining multiple interrelated entities into a single abstract entity object. An entity cluster is considered "virtual" or "abstract" in the sense that it is not actually an entity in the final ERD.

entity instance—A term used in ER modeling to refer to a specific table row. Also known as an *entity occurrence*.

entity integrity—The property of a relational table that guarantees that each entity has a unique value in a primary key and that there are no null values in the primary key.

entity occurrence—See *entity instance*.

entity relationship diagram (ERD)—A diagram that depicts an entity relationship model's entities, attributes, and relations.

entity relationship (ER) model—A data model developed by P. Chen in 1975. It describes relationships (1:1, 1:*, and *:*) among entities at the conceptual level with the help of ER diagrams.

entity set—In a relational model, refers to a grouping of related entities.

entity subtype—In a generalization/specialization hierarchy, refers to a subset of an entity supertype where the entity supertype contains the common characteristics and the entity subtypes contain the unique characteristics of each entity subtype.

entity supertype—In a generalization/specialization hierarchy, refers to a generic entity type that contains the common characteristics of entity subtypes.

equijoin—A join operator that links tables based on an equality condition that compares specified columns of the tables.

exclusive lock—A lock that is reserved by a transaction. An exclusive lock is issued when a transaction requests permission to write (update) a data item and no locks are previously held on that data item by any other transaction. An exclusive lock does not allow any other transactions to access the database. See also *shared lock.*

existence-dependent—A property of an entity whose existence depends on one or more other entities. In an existence-dependent environment, the existence-independent table must be created and loaded first because the existence-dependent key cannot reference a table that does not yet exist.

existence-independent—An entity that can exist apart from one or more related entities. It must be created first when referencing an existence-dependent table to it.

explicit cursor—In procedural SQL, a cursor created to hold the output of a SQL statement that may return two or more rows (but could return zero rows or only one row).

extended entity relationship (EER) model—Sometimes referred to as the enhanced entity relationship model; the result of adding more semantic constructs (entity supertypes, entity subtypes, and entity clustering) to the original entity relationship (ER) model**.**

extended relational data model (ERDM)—A model that includes the object-oriented model's best features in an inherently simpler relational database structural environment.

extends—In a DBMS environment, refers to the data files' ability to automatically expand in size, using predefined increments.

Extensible Markup Language (XML)—Derived from SGML, this is a metalanguage used to represent and manipulate data elements. Unlike other markup languages, XML permits the manipulation of a document's data elements.

external model—The application programmer's view of the data environment. Given its business-unit focus, an external model works with a data subset of the global database schema.

external schema—The specific representation of an external view, that is, the end user's view of the data environment.

F

facts—Measurements (values) that represent a specific business aspect or activity. For example, sales figures are numeric measurements that represent product and/or service sales. Facts commonly used in business data analysis are units, costs, prices, and revenues.

fact table—In a data warehouse, refers to the star schema center table containing facts that are linked and classified through their common dimensions. A fact table is in a one-to-many relationship with each associated dimension table.

failure transparency—A DDBMS feature that allows continuous operation of a DDBMS, even in the event of a failure in one of the nodes of the network.

fan trap—A design trap that occurs when one entity is in two 1:* relationships to other entities, thus producing an association among the other entities that is not expressed in the model.

field—A character or group of characters (alphabetic or numeric) that defines a characteristic of a person, place, or thing. For example, a person's Social Security number, address, phone number, and bank balance all constitute fields.

field-level lock—Allows concurrent transactions to access the same row as long as they require the use of different fields (attributes) within that row. Yields the most flexible multiuser data access but requires a high level of computer overhead.

file—A named collection of related records.

file organization - Techniques for physically arranging relations onto secondary storage.

first normal form (1NF)—The first stage in the normalization process. It describes a relation depicted in tabular format, with no repeating groups and with a primary key identified. All nonkey attributes in the relation are dependent on the primary key.

flags—Special codes implemented by designers to trigger a required response, to alert end users to specified conditions, or to encode values. Flags may be used to prevent nulls by bringing attention to the absence of a value in a table.

foreign key—See *key*.

fourth-generation language (4GL)—A computer language that allows a user to specify what must be done without specifying how it must be done. The most prominent 4GL in the database environment is SQL.

fourth normal form (4NF)—A table is in 4NF when it is in 3NF and contains no multiple independent sets of multivalued dependencies.

fragmentation transparency—A DDBMS feature that allows a system to treat a distributed database as a single database even though the database is divided into two or more fragments.

front-end CASE tools—A computer-aided software tool that has been classified as "front end" because it provides support for the planning, analysis, and design phases of the SDLC. In comparison, back-end case tools provide support for the coding and implementation phases.

full backup (database dump)—A complete copy of an entire database saved and periodically updated in a separate memory location. Ensures a full recovery of all data in the event of a physical disaster or a database integrity failure.

full functional dependence—A condition in which an attribute is functionally dependent on a composite key but not on any subset of that composite key.

fully heterogeneous distributed database system (fully heterogeneous DDBMS)—Integrates different types of database management systems (hierarchical, network, and relational) over a network. It supports different database management systems that may even support different data models running under different computer systems, such as mainframes, minicomputers, and microcomputers. See also *heterogeneous DDBMS* and *homogeneous DDBMS*.

fully replicated database—Stores multiple copies of each database fragment at multiple sites. See also *partially replicated database.*

functional dependence—Within a relation R, an attribute B is functionally dependent on an attribute A if and only if a given value of the attribute A determines exactly one value of the attribute B. The relationship "B is dependent on A" is equivalent to "A determines B" and is written as A->B.

---

granularity—Refers to the level of detail represented by the values stored in a table's row. Data stored at their lowest level of granularity are said to be *atomic data*.

GROUP BY—A SQL clause used to create frequency distributions when combined with any of the aggregate functions in a SELECT statement.

<p align="center">H</p>

hardware independence—Means that a model does not depend on the hardware used in the implementation of the model. Therefore, changes in the hardware will have no effect on the database design at the conceptual level.

hash file – A file which is hashed on one or more fields.

HAVING—A particularly useful extension of the GROUP BY feature. The HAVING clause is applied to the output of a GROUP BY operation to restrict the selected rows.
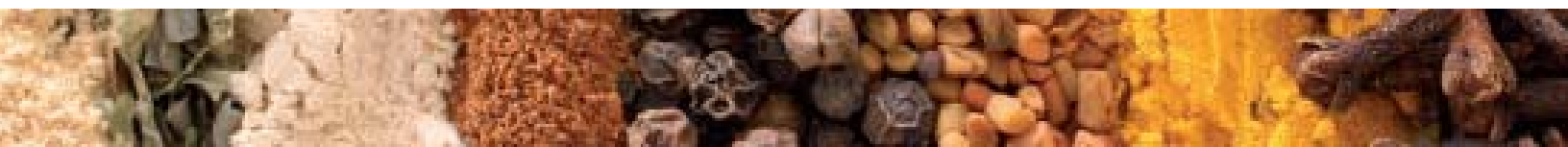
heap file – A file which contains randomly ordered records.

heterogeneity transparency—A DDBMS feature that allows a system to integrate several different centralized DBMSs into one logical DDBMS.

heterogeneous DDBMS—Integrates different types of centralized database management systems over a network. See also *fully heterogeneous distributed database system (fully heterogeneous DDBMS)* and *homogeneous DDBMS*.

hierarchical model—No longer a major player in the current database market; important to know, however, because the basic concepts and characteristics form the basis for subsequent database development. This model is based on an "upside-down" tree structure in which each record is called a segment. The top record is the root segment. Each segment has a 1:* relationship to the segment directly below it.

homogeneous DDBMS—Integrates only one particular type of centralized database management system over a network. See also *heterogeneous DDBMS* and *fully heterogeneous distributed database system (fully heterogeneous DDBMS)*.

homonym—Indicates the use of the same name to label different attributes; generally should be avoided. Some relational software automatically checks for homonyms and either alerts the user to their existence or automatically makes the appropriate adjustments. See also *synonym*.

horizontal fragmentation—The distributed database design process that breaks up a table into subsets consisting of unique rows. See also *database fragments* and *vertical fragmentation*.

host language—A term used to describe any language that contains embedded SQL statements.

I

identifiers—The ERM uses identifiers to uniquely identify each entity instance. In the relational model, such identifiers are mapped to primary keys in tables.

identifying relationship—A relationship that exists when the related entities are existence-dependent. Also called a *strong relationship* or *strong identifying relationship* because the dependent entity's primary key contains the primary key of the parent entity.

immediate update—When a database is immediately updated by transaction operations during the transaction's execution, even before the transaction reaches its commit point.

implicit cursor—A cursor that is automatically created in procedural SQL when the SQL statement returns only one value.

inconsistent retrievals—A concurrency control problem that arises when a transaction calculating summary (aggregate) functions over a set of data—while other transactions are updating the data—yields erroneous results.

incremental backup—A process that makes a backup only of data that has changed in the database since the last backup (incremental or full).

index—An ordered array composed of index key values and row ID values (pointers). Indexes are generally used to speed up and facilitate data retrieval. Also known as an *index key.*

index key—See *index*.

index selectivity—A measure of how likely an index will be used in query processing.

information—The result of processing raw data to reveal its meaning. Information consists of transformed data and facilitates decision making.

information engineering (IE)—A methodology that translates a company's strategic goals into data and applications that will help the company achieve its goals.

information resource dictionary—See *data dictionary*.

information resource manager (IRM)—See *data administrator (DA)*.

information system (IS)—A system that provides for data collection, storage, and retrieval; facilitates the transformation of data into information and the management of both data and information. An information system is composed of hardware, software (DMBS and applications), the database(s), people, and procedures.

information systems architecture (ISA)—The output of the information engineering (IE) process that serves as the basis for planning, developing, and controlling future information systems. (IE allows for the translation of a company's strategic goals into the data and applications that will help the company achieve those goals. IE focuses on the description of the corporate data instead of the processes.)

information systems (IS) department—An evolution of the data-processing department when responsibilities were broadened to include service and production functions.

inheritance—In the object-oriented data model, the ability of an object to inherit the data structure and methods of the classes above it in the class hierarchy. See also *class hierarchy*.

inner join—A join operation in which only rows that meet a given criteria are selected. The join criteria can be an equality condition (natural join or equijoin) or an inequality condition (theta join). Inner join is the most commonly used type of join. Contrast with *outer join.*

inner query—A query that is embedded (or nested) inside another query. Also known as a *nested query* or a *subquery*.

input/output (IO) request—A low-level (read or write) data access operation to/from computer devices (such as memory, hard disks, video, and printers).
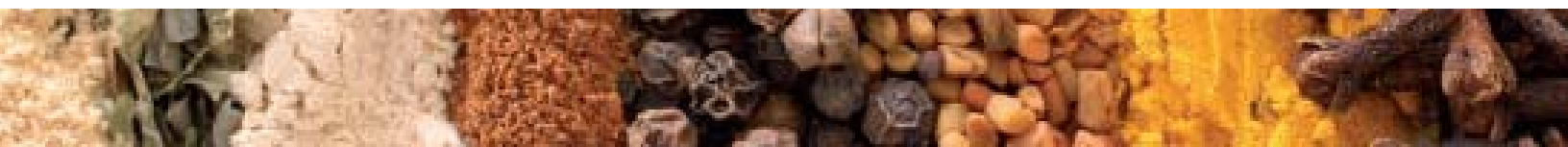
INSERT—A SQL command that allows the insertion of data rows into a table, one row at a time or multiple rows at a time, using a subquery.

integrity constraints – a set of pre-defined rules that the data must adhere to – see *constraint*.

internal model—In database modeling, refers to a level of data abstraction that adapts the conceptual model to a specific DBMS model for implementation.

internal schema—Depicts a specific representation of an internal model, using the database constructs supported by the chosen database. (The internal model is the representation of a database as "seen" by the DBMS. In other words, the internal model requires a designer to match the conceptual model's characteristics and constraints to those of the selected implementation model.)

INTERSECT - The INTERSECT operator denoted as $Relation_1 \cap Relation_2$**,** returns only the tuples that appear in both relations. The relations must be union-compatible.

islands of information—A term used in the old-style file system environment to refer to independent, often duplicated, and inconsistent data pools created and managed by different organizational departments.

isolation—A property of a database transaction that guarantees that a data item used by one transaction is not available to other transactions until the first transaction ends.

iterative process—A process based on repetition of steps and procedures.

J

Java—An object-oriented programming language developed by Sun Microsystems that runs on top of the Web browser software. Java applications are compiled and stored in the Web server. Java's main advantage is its ability to let application developers develop their applications once and run them in many environments.

JavaScript—A scripting language (one that enables the running of a series of commands or macros) developed by Netscape that allows Web authors to design interactive Web sites. JavaScript code is embedded in Web pages. This JavaScript is downloaded with the page and is activated when a specific event takes place, such as a mouse click on an object.

join columns—A term used to refer to the columns that join two tables. The join columns generally share similar values.

join index - The JOIN index is used mainly in data warehousing and applies to columns from two or more tables whose values come from the same domain. It is often referred to as a bitmap join index and it is a way of saving space by reducing the volume of data that must be joined.

K

key—An entity identifier based on the concept of functional dependence; may be classified as follows: *Superkey:* An attribute (or combination of attributes) that uniquely identifies each entity in a table. *Candidate key:* A minimal superkey, that is, one that does not contain a subset of attributes that is itself a superkey. *Primary key (PK):* A candidate key selected as a unique entity identifier. *Secondary key:* A key that is used strictly for data retrieval purposes. For example, a customer is not likely to know his or her customer number (primary key), but the combination of last name, first name, middle initial, and telephone number is likely to make a match to the appropriate table row. *Foreign key:* An attribute (or combination of attributes) in one table whose values must match the primary key in another table or whose values must be null.

key attribute(s)—The attribute(s) that form(s) a primary key. See also *prime attribute*.
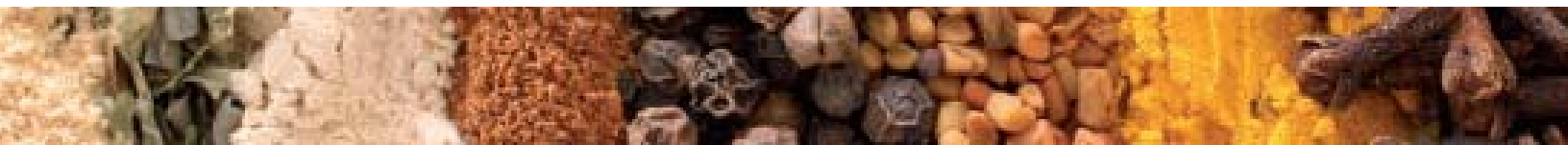
knowledge—The body of information and facts about a specific subject. Knowledge implies familiarity, awareness, and understanding of information as it applies to an environment. A key characteristic of knowledge is that "new" knowledge can be derived from "old" knowledge.

L

left outer join—In a pair of tables to be joined, a left outer join yields all of the rows in the left table, including those that have no matching values in the other table. For example, a left outer join of Customer with Agent will yield all of the Customer rows, including the ones that do not have a matching Agent row. See also *outer join* and *right outer join*.

linking table—In the relational model, a table that implements a M:M relationship. See also *composite entity*.

local mapping transparency—A property of a DDBMS in which access to the data requires the end user to know both the name and the location of the fragments in order to access the database. See also *location transparency.*

location transparency—The property of a DDBMS in which access to the data requires that only the name of the database fragments be known. (Fragment locations need not be known.) See also *local mapping transparency*.

lock—A device that is employed to guarantee unique use of a data item to a particular transaction operation, thereby preventing other transactions from using that data item. A transaction requires a lock prior to data access, and that lock is released (unlocked) after the operation's execution to enable other transactions to lock the data item for their use.

lock granularity—Indicates the level of lock use. Locking can take place at the following levels: database, table, page, row, and field (attribute).

lock manager—A DBMS component that is responsible for assigning and releasing locks.

logical data format—The way in which a human being views data.
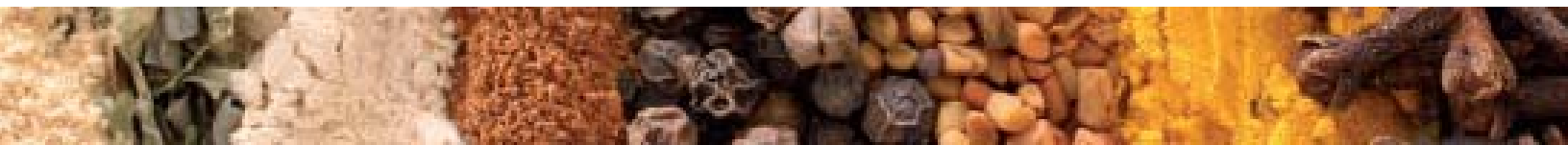
logical design—A stage in the design phase that matches the conceptual design to the requirements of the selected DBMS and is, therefore, software-dependent. It is used to translate the conceptual design into the internal model for a selected database management system, such as DB2, SQL Server, Oracle, IMS, Informix, Access, and Ingress.

logical independence—A condition that exists when the internal model can be changed without affecting the conceptual model. (The internal model is hardware-independent because it is unaffected by the choice of computer on which the software is installed. Therefore, a change in storage devices or even a change in operating systems will not affect the internal model.)

lost updates—A concurrency control problem in which data updates are lost during the concurrent execution of transactions.

M

mandatory participation—A term used to describe a relationship in which one entity occurrence must have a corresponding occurrence in another entity. Example: EMPLOYEE works in DIVISION. (A person cannot be an employee if he or she is not assigned to a company's division.)

manual query optimization—An operation mode that requires the end user or programmer to define the access path for the execution of a query.

manual statistical generation mode—One mode of generating statistical data access information used for query optimization. In this mode, the DBA must periodically run a routine to generate the data access statistics; for example, running the RUNSTAT command in an IBM DB2 database.

many-to-many (*:*) relationships—One of three types of relationships (associations among two or more entities) in which one occurrence of an entity is associated with many occurrences of a related entity and one occurrence of the related entity is associated with many occurrences of the first entity.

materialized view—A dynamic table that not only contains the SQL query command to generate the rows, but also stores the actual rows. The materialized view is created the first time the query is run and the summary rows are stored in the table. The materialized view rows are automatically updated when the base tables are updated.

MAX—A SQL aggregate function that yields the maximum attribute value encountered in a given column.

metadata—Data about data, that is, data concerning data characteristics and relationships. See also *data dictionary*.

method—In the object-oriented data model, a named set of instructions to perform an action. Methods represent real-world actions. Methods are invoked through messages.

metrics—In a data warehouse, numeric facts that measure something of interest to the end user.

MIN—A SQL aggregate function that yields the minimum attribute value encountered in a given column.

minimal data rule—Defined as "All that is needed is there, and all that is there is needed." In other words, all data elements required by database transactions must be defined in the model, and all data elements defined in the model must be used by at least one database transaction.

mixed fragmentation—Regarding data fragmentation, refers to a combination of horizontal and vertical strategies, meaning a table may be divided into several rows, each row having a subset of the attributes (columns).

module—(1) A design segment that can be implemented as an autonomous unit, sometimes linked to produce a system. (2) An information system component that handles a specific function, such as inventory, orders, or payroll.
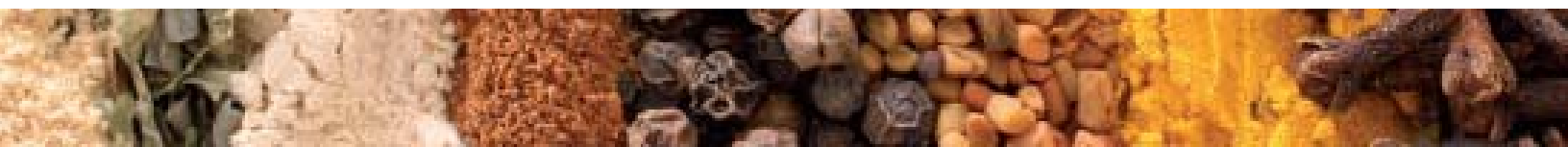
module coupling—A description of the extent to which modules are independent of one another.

monotonicity—Ensures that time stamp values always increase. (The time stamping approach to scheduling concurrent transactions assigns a global, unique time stamp to each transaction. The time stamp value produces an explicit order in which transactions are submitted to the DBMS.)

multidimensional database management system (MDBMS)—A database management system that uses proprietary techniques to store data in matrixlike arrays of *n*-dimensions, known as cubes.

multidimensional online analytical processing (MOLAP)—Extends online analytical processing functionality to multidimensional database management systems.

multiple-site processing, multiple-site data (MPMD)—A scenario describing a fully distributed database management system with support for multiple DPs and transaction processors at multiple sites.

multiple-site processing, single-site data (MPSD)—A scenario in which multiple processes run on different computers sharing a single data repository.

multiuser database—A database that supports multiple concurrent users.

multiplicity - refers to the number of instances of one entity that are associated with one instance of a related entity**.**

multivalued attribute—An attribute that can have many values for a single entity occurrence. For example, an EMP_DEGREE attribute might store the string "BSc, MBA, PHD" to indicate three different degrees held.

mutual consistency rule—A data replication rule requiring that all copies of data fragments be identical.

mutual exclusive rule—A condition in which only one transaction at a time can own an exclusive lock on the same object.


N

natural join—A relational operation that links tables by selecting only the rows with common values in their common attribute(s).

natural key (natural identifier)—A real-world, generally accepted identifier used to identify real-world objects. As its name implies, a natural key is familiar to end users and forms part of their day-to-day business vocabulary.

network model—A data model standard created by the CODASYL Data Base Task Group in the late 1960s. It represented data as a collection of record types and relationships as predefined sets with an owner record type and a member record type in a 1:* relationship.

network schema—In a network data base model, defines the conceptual organization of the entire database as viewed by the database administrator. The schema includes a definition of the database name, the record type for each record, and the components that make up those records.

network subschema—In a network database model, defines the portion of the database "seen" by the application programs that actually produce the desired information from the data contained within the database. The existence of subschema definitions allows all application programs simply to invoke the subschema required to access the appropriate data.
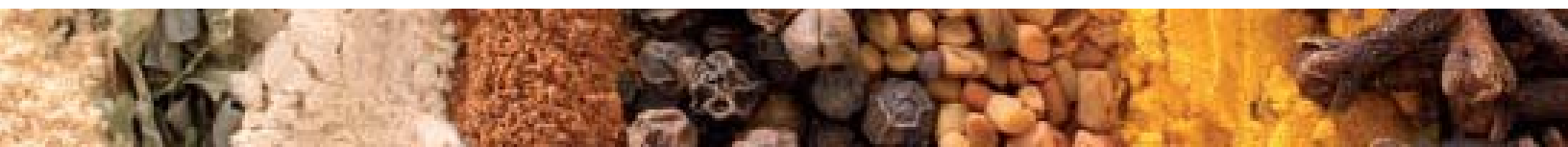
non-identifying relationship—A relationship that occurs when the primary key of the dependent (many side) entity does not contain the primary key of the related parent entity. Also known as a *weak relationship*.

nonkey attribute—See *nonprime attribute*.

nonprime attribute—An attribute that is not part of a key.

normalization—A process that assigns attributes to entities in such a way that data redundancies are reduced or eliminated.

NOT—A SQL logical operator that negates a given predicate.

null—In SQL, refers to the absence of an attribute value. Note: A null is not a blank.

O

object—An abstract representation of a real-world entity that has a unique identity, embedded properties, and the ability to interact with other objects and with itself.

Object Linking and Embedding for Database (OLE-DB)—Based on Microsoft's Component Object Model (COM), OLE-DB is database middleware that adds object-oriented functionality for accessing relational and nonrelational data. OLE-DB was the first part of Microsoft's strategy to provide a unified object-oriented framework for the development of next-generation applications.

object-oriented database management system (OODBMS)—Data management software used to manage data found within an object-oriented database model.

object-oriented data model (OODM)—A data model whose basic modeling structure is an object.

object-oriented programming (OOP)—An alternative to conventional programming methods based on object-oriented concepts. It reduces programming time and lines of code and increases programmers' productivity.

object/relational database management system (O/RDBMS)—A DBMS based on the extended relational model (ERDM.) The ERDM, championed by many relational database researchers, constitutes the relational model's response to the OODM. This model includes many of the object-oriented model's best features within an inherently simpler relational database structural environment.
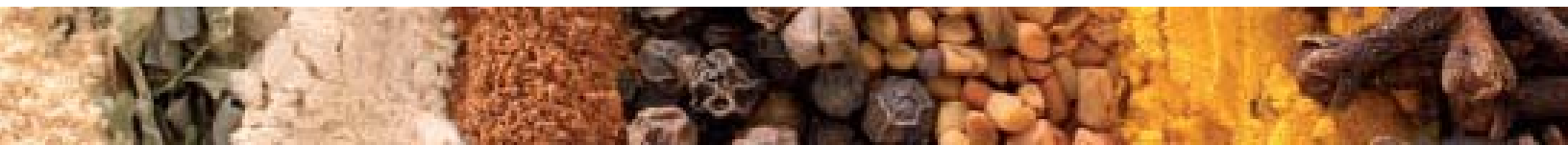
one-to-many (1:*) relationship—One of three types of relationships (associations among two or more entities) that are used by data models. In a 1:* relationship, one entity instance is associated with many instances of the related entity.

one-to-one (1:1) relationship—One of three types of relationships (associations among two or more entities) that are used by data models. In a 1:1 relationship, one entity instance is associated with only one instance of the related entity.

online analytical processing (OLAP)—Decision support system (DSS) tools that use multidimensional data analysis techniques. OLAP creates an advanced data analysis environment that supports decision making, business modeling, and operations research activities.

Open Database Connectivity (ODBC)—Database middleware developed by Microsoft to provide a database access API to Windows applications.

operational database—A database that is designed primarily to support a company's day-to-day operations. Also known as a *transactional database* or *production database.*

optimizer hints—Special instructions for the query optimizer that are embedded inside the SQL command text.

optional participation—Means that one entity occurrence does not require a corresponding entity occurrence in a particular relationship.

OR—The SQL logical operator used to link multiple conditional expressions in a WHERE or HAVING clause. It requires that only one of the conditional expressions be true.

ORDER BY—A SQL clause useful for ordering the output of a SELECT query (for example, in ascending or descending order).

outer join—A relational-algebra JOIN operation that produces a table in which all unmatched pairs are retained; unmatched values in the related table are left null. Contrast with *inner join.* See also *left outer join* and *right outer join.*

overlapping subtypes—Subtypes that contain a subset of the supertype entity set, but each entity instance (row) of the supertype can appear in more than one subtype.
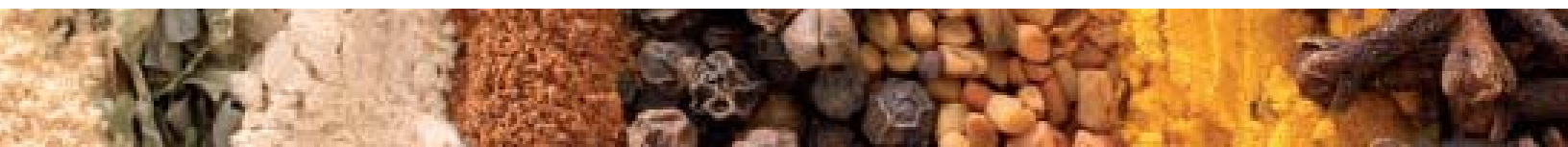
P

page-level lock—In this type of lock, the database management system will lock an entire diskpage, or section of a disk. A diskpage can contain data for one or more rows and from one or more tables.

partial completeness—In a generalization hierarchy, means that not every supertype occurrence is a member of a subtype; that is, there may be some supertype occurrences that are not members of any subtype.

partial dependency—In normalization, a condition in which an attribute is dependent on only a portion (subset) of the primary key.

partially replicated database—A distributed database in which copies of only some database fragments are stored at multiple sites. See also *fully replicated database*.

participants—An ER term used to label the entities that participate in a relationship. Example: LECTURER teaches CLASS. (The *teaches* relationship is based on the participants LECTURER and CLASS.)

partitioned data allocation strategy—In a DDBMS, the data allocation strategy in which the database is divided into fragments that are stored at several sites.

partitioning—The process of splitting a table into subsets of rows or columns.

passive data dictionary—A DBMS data dictionary that requires an end-user-initiated command to update its data access statistics. See also *data dictionary.*

performance transparency—A DDBMS feature that allows a system to perform as though it were a centralized DBMS (no degradation of response times).

performance tuning—Activities that make a database perform more efficiently in terms of storage and access speed.

periodicity—Usually expressed as current year only, previous years, or all years; provides information about the time span of data stored in a table.

persistent stored module (PSM)—A block of code (containing standard SQL statements and procedural extensions) that is stored and executed at the DBMS server.

personalization—Customization of a Web page for individual users.
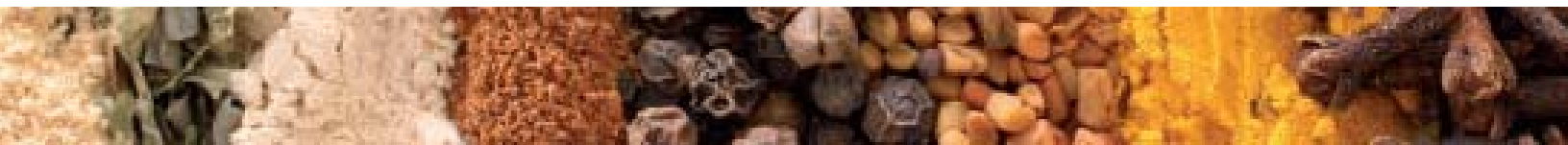
physical data format—The way in which a computer "sees" (stores) data.

physical design—A stage of database design that maps the data storage and access characteristics of a database. Since these characteristics are a function of the types of devices supported by the hardware, the data access methods supported by the system (and the selected DBMS) physical design is both hardware- and software-dependent. See also *physical model*.

physical independence—A condition that exists when the physical model can be changed without affecting the internal model.

physical model—A model in which the physical characteristics (location, path, and format) are described for the data. Both hardware- and software-dependent. See also *physical design*.

plug-in—In the World Wide Web (WWW), a client-side, external application that is automatically invoked by the browser when it is needed to manage specific types of data.

policies—General statements of direction that are used to manage company operations through the communication and support of the organization's objectives.
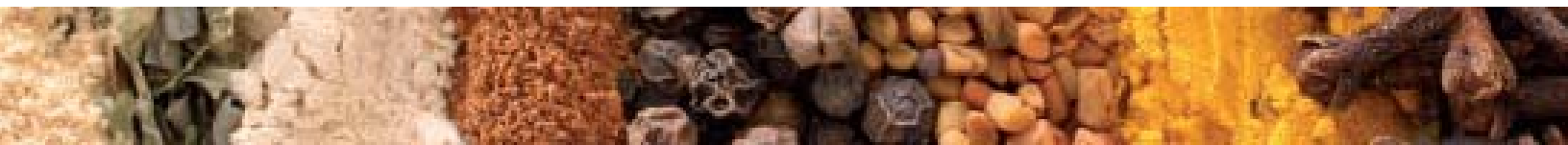
predicate logic—Used extensively in mathematics, provides a framework in which an assertion (statement of fact) can be verified as either true or false. For example, suppose that a student with a student ID of 12345678 is named Melissa Sanduski. That assertion can easily be demonstrated to be true or false.

primary index - These indexes are placed on unique fields such as the primary key. They are used to locate a specific record pointed to by the index. A file can have at most one primary index but can have several secondary indexes.

primary key (PK)—In the relational model, an identifier composed of one or more attributes that uniquely identifies a row. See also *key*.

prime attribute—A key attribute, that is, an attribute that is part of a key or is the whole key. See also *key attribute*.

privacy—Control of data usage dealing with the rights of individuals and organizations to determine the "who, what, when, where, and how" of data access.

procedural SQL (PL/SQL)—A type of SQL that allows the use of procedural code and SQL statements that are stored in a database as a single callable object that can be invoked by name.

procedure cache—A shared, reserved memory area that stores the most recently executed SQL statements or PL/SQL procedures (including triggers and functions). Also called *SQL cache*.

procedures—Series of steps to be followed during the performance of a given activity or process.

production database—The main database designed to keep track of the day-to-day operations of a company. See also *transactional database*.

profile—In Oracle, a named collection of settings that controls how much of the database resource a given user can use.

PROJECTION - The PROJECT operator, denoted by $\prod$ returns all values for selected attributes.

Q

query—A question or task asked by an end user of a database in the form of SQL code. A specific request for data manipulation issued by the end user or the application to the DBMS.

query language—A nonprocedural language that is used by a DBMS to manipulate its data. An example of a query language is SQL.

query optimizer—A DBMS process that analyzes SQL queries and finds the most efficient way to access the data. The query optimizer generates the access or execution plan for the query.

query result set—The collection of data rows that are returned by a query.

R

RAID—An acronym that means Redundant Array of Independent Disks. RAID is used to provide balance between performance and fault tolerance. RAID systems use multiple disks to create virtual disks (storage volumes) formed by several individual disks. RAID systems provide performance improvement and fault tolerance.

record—A collection of related (logically connected) fields.

recursive query—A nested query that joins a table to itself. For example, a recursive query joins the EMPLOYEE table to itself.

recursive relationship—A relationship that is found within a single entity type. For example, an EMPLOYEE is married to an EMPLOYEE or a PART is a component of another PART.

redundant transaction logs—Most database management systems keep several copies of the transaction log to ensure that the physical failure of a disk will not impair the DBMS's ability to recover data.

referential integrity—A condition by which a dependent table's foreign key must have either a null entry or a matching entry in the related table. Even though an attribute may not have a *corresponding* attribute, it is impossible to have an invalid entry.

relational algebra—A set of mathematical principles that form the basis of the manipulation of relational table contents; composed of eight main functions: SELECT, PROJECT, JOIN, INTERSECT, UNION, DIFFERENCE, PRODUCT, and DIVIDE.

relational algebra expression – used to create and manipulate relations (tables) in a database.

relational schema - a textual representation of the database tables where each table is described by its name followed by the list of its attributes in parentheses.

relational database management system (RDBMS)—A collection of programs that manages a relational database. The RDBMS software translates a user's logical requests (queries) into commands that physically locate and retrieve the requested data. A good RDBMS also creates and maintains a data dictionary (system catalog) to help provide data security, data integrity, concurrent access, easy access, and system administration to the data in the database through a query language (SQL) and application programs.

relational model—Developed by E. F. Codd (of IBM) in 1970, it represents a major breakthrough for users and designers because of its conceptual simplicity. The relational model, based on mathematical set theory, represents data as independent relations. Each relation (table) is conceptually represented as a matrix of intersecting rows and columns. The relations are related to each other through the sharing of common entity characteristics (values in columns).
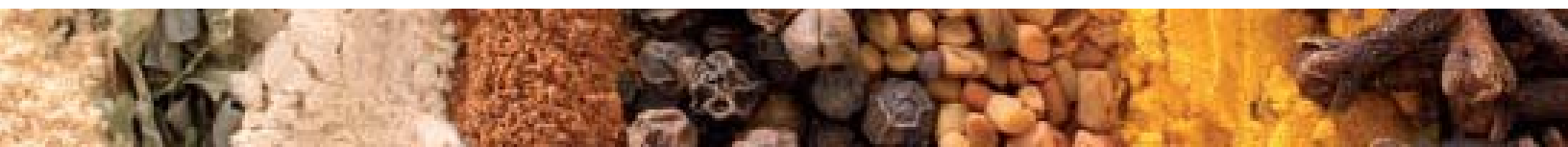
relational online analytical processing (ROLAP)—Provides online analytical processing functionality by using relational databases and familiar relational query tools to store and analyze multidimensional data.

relational schema—The description of the organization of a relational database as seen by the database administrator.

relations—In a relational database model, represent entity sets. Relations are implemented as tables. Relations (tables) are related to each other through the sharing of a common entity characteristic (value in a column).

relationship—An association between entities.

relationship degree—Indicates the number of entities or participants associated with a relationship. A relationship degree can be unary, binary, ternary, or higher level.

Remote Data Objects (RDO)—A higher-level object-oriented application interface used to access remote database servers. RDO uses the lower-level DAO and ODBC for direct access to databases. RDO was optimized to deal with server-based databases such as MS SQL Server, Oracle, and DB2.

remote request—A DDBMS feature that allows a single SQL statement to access data in a single remote DP. See also *remote transaction*.

remote transaction—A DDBMS feature that allows a transaction (formed by several requests) to access data in a single remote DP. See also *remote request*.

repeating groups—In a relation, a characteristic describing a group of multiple entries of the same type that exist for a single key attribute occurrence. For example, a car can have multiple colors (top, interior, bottom, trim, and so on).

replicated data allocation—A data allocation strategy by which copies of one or more database fragments are stored at several sites.

replica transparency—Refers to the DDBMS's ability to hide the existence of multiple copies of data from the user.

replication—The process of creating and managing duplicate versions of a database. Used to place copies in different locations and to improve access time and fault tolerance.

reserved words—Words that are used by a system and that cannot be used for any other purpose. For example, in Oracle SQL , the word INITIAL cannot be used to name tables or columns.

RESTRICT – see the **SELECT** relational algebra operator

right outer join—In a pair of tables to be joined, a right outer join yields all of the rows in the right table, including the ones with no matching values in the other table. For example, a right outer join of CUSTOMER with AGENT will yield all of the agent rows, including the ones that do not have a matching CUSTOMER row. See also *left outer join* and *outer join*.

role (UML) - In UML the ends of a relationship between two classes (entities) are referred to as its "roles". In an UML class diagram, roles are shown instead of the association (relationship) name.

role —In Oracle, a named collection of database access privileges that authorize a user to connect to a database and use the database system resources.

ROLLBACK—A SQL command that restores the database table contents to their original condition (the condition that existed after the last COMMIT statement).

roll up—In SQL, an OLAP extension used with the GROUP BY clause to aggregate data by different dimensions. (Rolling up the data is the exact opposite of drilling down the data.) See also *drill down*.

row-level lock—A comparatively less restrictive database lock where the DBMS allows concurrent transactions to access different rows of the same table, even when the rows are located on the same page.

row-level trigger—A trigger that is executed once for each row affected by the triggering SQL statement. A row-level trigger requires the use of the s FOR EACH ROW keywords in the trigger declaration.

rule-based optimizer—A query optimization mode based on the rule-based query optimization algorithm.

rule-based query optimization algorithm—A query optimization technique that uses a set of preset rules and points to determine the best approach to executing a query.

rules of precedence—Basic algebraic rules that specify the order in which operations are performed, such as conditions within parentheses being executed first. For example, in the equation $2 + (3 * 5)$, the multiplication portion is calculated first, making the correct answer 17.
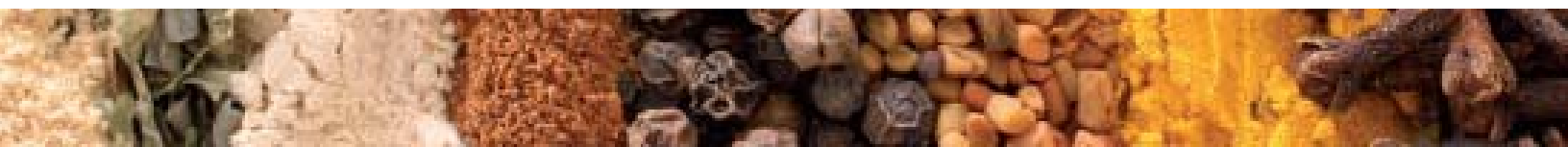
S

scheduler—The DBMS component that is responsible for establishing the order in which concurrent transaction operations are executed. The scheduler *interleaves* the execution of database operations in a specific order (sequence) to ensure *serializability*.

schema—A logical grouping of database objects (tables, indexes, views, queries, etc.) that are related to each other. Usually, a schema belongs to a single user or application.

scope—That part of a system that defines the extent of the design, according to operational requirements.

script—A programming language that is not compiled, but rather is interpreted and executed at run time.

search services—Business-enabling Web services that allow Web sites to perform searches on their contents.

secondary index - These indexes can be placed on any field in the file that is unordered.

secondary key—A key that is used strictly for data retrieval purposes. For example, a customer is not likely to know his or her customer number (primary key), but the combination of last name, first name, middle initial, and telephone number is likely to make a match to the appropriate table row. See also *key.*

second normal form (2NF)—The second stage in the normalization process in which a relation is in 1NF and there are no partial dependencies (dependencies in only part of the primary key).

security—(1) Protecting data against accidental or intentional use by unauthorized users. (2) Services designed to ensure safety and privacy of data through encryption, digital certificates, SSL, S-HTTP, firewalls, and proxy servers.

SELECT—A SQL command that yields the values of all rows or a subset of rows in a table. The SELECT statement is used to retrieve data from tables.

selection - The relational operator SELECT (also known as RESTRICT), can be used to list all of the row values, or it can return only those row values that match a specified criterion. Selection is denoted by σ_θ

semantic data model—The first of a series of data models that more closely represented the real world, modeling both data and their relationships in a single structure known as an object. The SDM, published in 1981, was developed by M. Hammer and D. McLeod.

sequential file - A sequential file is where the records are stored in a sequence based on the value of one or more fields which is often the primary key.

serializability—A transaction property that ensures that the selected order of transaction operations creates a final database state that would have been produced if the transactions had been executed in a serial fashion.

server-side extension—A program that interacts directly with the server process to handle specific types of requests. They add significant functionality to Web servers and to intranets.

set theory—A mathematical science component that deals with sets, or groups of things, and is used as the basis for data manipulation in the relational model.

shared lock—A lock that is issued when a transaction requests permission to read data from a database and no exclusive locks are held on that data by another transaction. A shared lock allows other read-only transactions to access the database. See also *exclusive lock*.

simple attribute—An attribute that cannot be subdivided into meaningful components. Compare to *composite attribute*.

single-site processing, single-site data (SPSD)—A scenario in which all processing is done on a single CPU or host computer (mainframe, minicomputer, or PC) and all data are stored on the host computer's local disk.

single-user DBMS—A database management system classification that depicts a DBMS that supports only one user at a time.
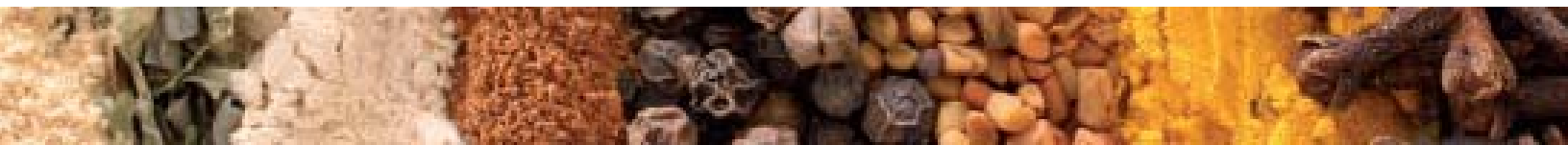
single-valued attribute—An attribute that can have only one value.

slice and dice—Multidimensional jargon meaning the ability to cut slices off of the data cube to perform a more detailed analysis.

snowflake schema—A type of star schema in which the dimension tables can have their own dimension tables. The snowflake schema is usually the result of normalizing dimension tables.

software independence—A property of any model or application that does not depend on the software used to implement it.

sparsity—In multidimensional data analysis, a measurement of the density of the data held in the data cube.

specialization hierarchy—A hierarchy that is based on the top-down process of identifying lower-level, more specific entity subtypes from a higher-level entity supertype. Specialization is based on grouping unique characteristics and relationships of the subtypes.

SQL cache—A shared, reserved memory area that stores the most recently executed SQL statements or PL/SQL procedures (including triggers and functions). Also called *procedure cache*.

SQL performance tuning—Activities oriented toward generating a SQL query that returns the correct answer in the least amount of time, using the minimum amount of resources at the server end.

standards—A detailed and specific set of instructions that describes the minimum requirements for a given activity. Standards are used to evaluate the quality of the output.

star schema—A data modeling technique used to map multidimensional decision support data into a relational database. The star schema represents data, using a central table known as a fact table, in a 1:* relationship with one or more dimension tables.

stateless system—Describes the fact that at any given time, a Web server does not know the status of any of the clients communicating with it. The Web does not reserve memory to maintain an open communications "state" between the client and the server.
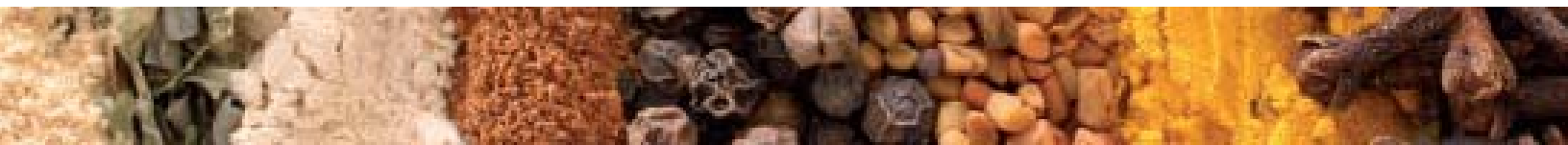
statement-level trigger—A SQL trigger that is assumed if the FOR EACH ROW keywords are omitted. This type of trigger is executed once, before or after the triggering statement completes, and is the default case.

static query optimization—A query optimization mode in which the access path to a database is predetermined at compilation time. Contrast with *dynamic query optimization.*

static SQL—A style of embedded SQL in which the SQL statements do not change while the application is running.

statistically based query optimization algorithm—A query optimization technique that uses statistical information about a database. These statistics are then used by the DBMS to determine the best access strategy.

stored function—A named group of procedural and SQL statements that returns a value, indicated by a RETURN statement in its program code.

stored procedure—(1) A named collection of procedural and SQL statements. (2) Business logic stored on a server in the form of SQL code or some other DBMS-specific procedural language.

strong (identifying) relationship—When two entities are existence-dependent; from a database design perspective, this exists whenever the primary key of the related entity contains the primary key of the parent entity.

structural dependence—A data characteristic that exists when a change in the database schema affects data access, thus requiring changes in all access programs.

structural independence—A data characteristic that exists when changes in the database schema do not affect data access.

Structured Query Language (SQL)—A powerful and flexible relational database language composed of commands that enable users to create database and table structures, perform various types of data manipulation and data administration, and query the database to extract useful information.

subordinate—In a DDBMS, a DP node that participates in a distributed transaction, using the two-phase COMMIT protocol.

subquery—A query that is embedded (or nested) inside another query. Also known as a *nested query* or an *inner query*.

subtype (entity set)—An entity (set) that contains the unique characteristics (attributes) of an entity whose general characteristics are found in another, more broadly defined entity known as a supertype. In a generalization hierarchy, it is any entity that is found below a parent entity. Example: The subtype PILOT of the supertype EMPLOYEE.

subtype discriminator—The attribute in the supertype entity that determines to which entity subtype each supertype occurrence is related.

SUM—A SQL aggregate function that yields the sum of all values for a given column or expression.

superkey—See *key*.

supertype (entity set)—An entity (set) that contains the general (commonly shared) characteristics of an entity (see *subtype*). If the entity set can include characteristics that are not common to all entities within the set, the supertype becomes the parent to one or more subtypes in a generalization hierarchy.

surrogate key—A system-assigned primary key, generally numeric and auto-incremented.

synonym—The use of different names to identify the same object, such as an entity, an attribute, or a relationship; should generally be avoided. See also *homonym*.

system catalog—A detailed system data dictionary that describes all objects in a database.

systems administrator (SYSADM)—The person responsible for coordinating the activities of the data processing function.

systems analysis—The process that establishes the need for and the extent of an information system.

systems development—The process of creating an information system.

Systems Development Life Cycle (SDLC)—The cycle that traces the history (life cycle) of an information system. The SDLC provides the big picture within which the database design and application development can be mapped out and evaluated.

T

table—A (conceptual) matrix composed of intersecting rows (entities) and columns (attributes) that represents an entity set in the relational model. Also called relation.

table-level lock—A locking scheme that allows only one transaction at a time to access a table. A table-level lock locks an entire table, preventing access to any row by transaction T2 while transaction T1 is using the table.

table space—In a DBMS, a logical storage space used to group related data. Also known as *file group*.

tag—In markup languages such as HTML and XML, a command inserted in a document to specify how the document should be formatted. Tags are used in server-side markup languages and interpreted by a Web browser for presenting data.

ternary relationship—An ER term used to describe an association (relationship) between three entities. Example: A CONTRIBUTOR contributes money to a FUND from which a RECIPIENT receives money.
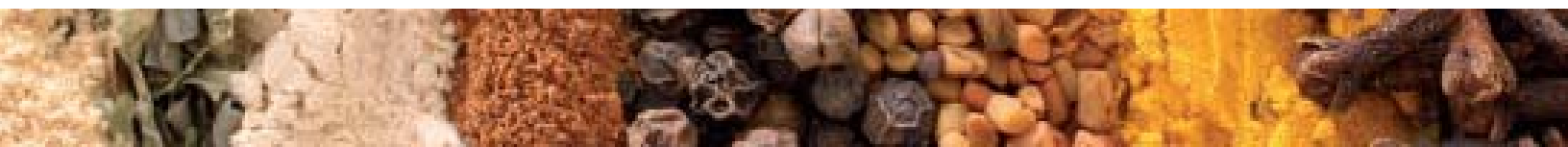
theta join—A join operator that links tables, using an inequality comparison operator (<, >, <=, >=) in the join condition.

third-generation language (3GL)—A language that requires a programmer to specify both what must be done and how it must be done. Examples include COBOL, BASIC, and FORTRAN.

third normal form (3NF)—A table is in 3NF when it is in 2NF and no nonkey attribute is functionally dependent on another nonkey attribute; that is, it cannot include transitive dependencies.

time stamping—In transaction management, a technique used in scheduling concurrent transactions that assigns a global unique time stamp to each transaction.

time-variant data—Data whose values are a function of time. For example, time variant data can be seen at work when the history of all administrative appointments (date of appointment and date of termination) are tracked.

top-down design—A design philosophy that begins by defining the main (macro) structures of a system and then moves to define the smaller units within those structures. In database design, it is a process that first identifies entities and then defines the attributes within the entities. Compare to *bottom-up design*.

total completeness—In a generalization/specialization hierarchy, a condition in which every supertype occurrence must be a member of at least one subtype.

transaction—A sequence of database operations (one or more database requests) that accesses the database. A transaction is a logical unit of work; that is, it must be *entirely* completed or aborted—no intermediate ending states are accepted. All transactions must have the following properties: (1) *Atomicity* requires that, unless all operations (parts) of a transaction are completed, the transaction be aborted. A transaction is treated as a single, indivisible logical unit of work. (2) *Consistency* indicates the permanence of the database consistent state. Once a transaction is completed, the database reaches a consistent state. (3) *Isolation* assures that the data used during the execution of a transaction cannot be used by a second transaction until the first one is completed. (4) *Durability* assures that once transaction changes are done, they cannot be undone or lost, even in the event of a system failure. (5) *Serializability* ensures that the selected order of transaction operations creates a final database state that would have been produced if the transactions had been executed in a serial fashion.
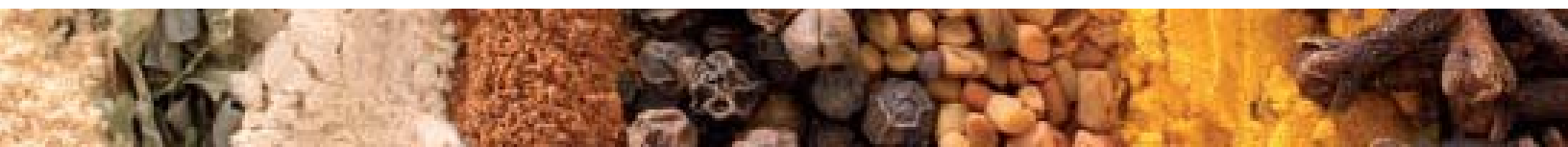
transactional database—A database designed to keep track of the day-to-day transactions of an organization. See also *production database*.

transaction log—A feature used by the DBMS to keep track of all transaction operations that update the database. The information stored in this log is used by the DBMS for recovery purposes.

transaction log backup—Backs up only the transaction log operations that are not reflected in a previous backup copy of the database.

transaction manager (TM)—See *transaction processor (TP)*.

transaction processor (TP)—In a DDBMS, the software component on each computer that requests data. The TP is responsible for the execution and coordination of all databases issued by a local application that access data on any DP. Also called *transaction manager (TM)*. See also *data processor (DP)*.

transaction transparency—A DDBMS property that ensures that database transactions will maintain the distributed database's integrity and consistency. They ensure that a transaction will be completed only when all database sites involved in the transaction complete their part of the transaction.

transaction usage map – see *composite usage map*.

transitive dependency—A condition in which an attribute is dependent on another attribute that is not part of the primary key.

trigger—A procedural SQL code that is automatically invoked by the relational database management system upon the occurrence of a data manipulation event.

tuple—In the relational model, a table row.

two-phase commit protocol—In a DDBMS, an algorithm used to ensure atomicity of transactions and database consistency as well as integrity in distributed transactions.

two-phase locking—A set of rules that governs the way transactions acquire and relinquish locks. Two-phase locking guarantees serializability, but it does not prevent deadlocks. The two-phase locking protocol is divided into two phases: (1) A *growing phase* occurs when the transaction acquires all of the locks that it needs without unlocking any *existing* data locks. Once all locks have been acquired, the transaction is in its *locked* point. (2) A *shrinking phase* occurs when the transaction releases all locks and cannot obtain a new lock.

U

unary relationship—An ER term used to describe an association *within* an entity. Example: A COURSE is a prerequisite to another COURSE.

uncommitted data—When trying to achieve concurrency control, uncommitted data causes data integrity and consistency problems. It occurs when two transactions are executed concurrently and the first transaction is rolled back after the second transaction has already accessed the uncommitted data, thus violating the isolation property of transactions.

Unified Modeling Language (UML) - A language that describes a set of diagrams and symbols that can be used to model a system graphically.

UNION - The UNION set operator combines all tuples from two relations, excluding duplicate tuples. The relations must have the same attribute characteristics (the columns and domains must be identical) to be used in the UNION.

union-compatible—Two or more tables are union-compatible when they share the same column names and the columns have compatible data types or domains.

unique fragment—In a DDBMS, a condition indicating that each row is unique, regardless of which fragment it is located in.

unique index—An index in which the index key can have only one pointer value (row) associated with it.

uniqueness—In concurrency control, a property of time stamping that ensures that no equal time stamp values can exist.
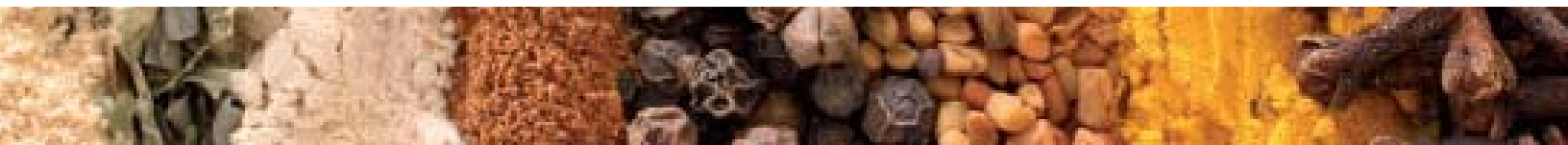
Universal Data Access (UDA)—Within the Microsoft application framework, a collection of technologies used to access any type of data source and to manage the data through a common interface.

unreplicated database—A distributed database in which each database fragment is stored at a single site.

updatable view—A view that can be used to update attributes in base tables that are used in the view.

UPDATE—A SQL command that allows attribute values to be changed in one or more rows of a table.

user—In a system, a uniquely identifiable object that allows a given person or process to log on to the database.

user-views – used to define the roles of different types of users.

V

VBScript—A client-side extension in the form of a Microsoft language product used to extend a browser's functionality; derived from Visual Basic.

vertical fragmentation—In distributed database design, the process that breaks up a table into fragments consisting of a subset of columns from the original table. Fragments must share a common primary key. See also *database fragments* and *horizontal fragmentation*.

very large databases (VLDBs)—As the name implies, databases that contain huge amounts of data—gigabyte, terabyte, and petabytes ranges are not unusual.

view—A virtual table based on a SELECT query.

W

wait/die—A concurrency control scheme in which the older transaction waits and the younger transaction is rolled back and rescheduled.

weak entity—An entity that displays existence dependence and inherits the primary key of its parent entity. Example: A DEPENDENT requires the existence of an EMPLOYEE.

weak relationship—A relationship that exists when the PK of the related entity does not contain a PK component of the parent entity. Also known as a *non-identifying relationship.*

Web application server—A middleware application that expands the functionality of Web servers by linking them to a wide range of services, such as databases, directory systems, and search engines.

Web-to-database middleware—A database server-side extension program that retrieves data from databases and passes it on to the Web server, which sends it to the client's browser for display purposes.

wildcard character—A symbol that can be used as a general substitute for one or more characters.

workgroup database—A multiuser database that supports a relatively small number of users (usually fewer than 50) or that is used for a specific department in an organization.

wound/wait—A concurrency control scheme in which the older transaction rolls back the younger transaction and reschedules it.

write-ahead-log protocol—In concurrency control, a process that ensures that transaction logs are always written to permanent storage before any database data are actually updated. Also called write-ahead protocol.

write-ahead protocol—See *write-ahead-log protocol*.

write-through—In concurrency control, a process that ensures that a database is immediately updated by transaction operations during the transaction's execution, even before the transaction reaches its commit point.

X

XML schema—An advanced data definition language that is used to describe the structure (elements, data types, relationship types, ranges, and default values) of XML data documents. One of the main advantages of an XML schema is that it more closely maps to database terminology and features. For example, an XML schema will be able to define common database types such as date, integer or decimal, minimum and maximum values, list of valid values, and required elements. Using the XML schema, a company could validate the data for values that may be out of range, incorrect dates, valid values, and so on.

XML schema definition (XSD)—A file containing the description of an XML document.

XSL (Extensible Style Language)—A specification used to define the rules by which XML data are formatted and displayed. The XSL specification is divided into two parts: Extensible Style Language Transformations (XSLT) and XSL style sheets.

XSL style sheets—Similar to presentation templates, define the presentation rules applied to XML elements. The XSL style sheet describes the formatting options to apply to XML elements when they are displayed on a browser, cellular phone display, PDA screen, and so on.

XSLT (Extensible Style Language Transformations)—A term that describes the general mechanism used to extract and process data from one XML document and to enable its transformation within another document.