IBM Informix
Version 11.70

IBM Informix Glossary

SC27-3531-00

IBM Informix
Version 11.70

IBM Informix Glossary

SC27-3531-00

Note
Before using this information and the product it supports, read the information in "Notices" on page B-1.

Contents

Glossary

This glossary includes terms and definitions for IBM® Informix® database software products.

The following cross-references are used in this glossary:

- **See** refers you from a term to a preferred synonym, or from an acronym or abbreviation to the defined full form.
- **See also** refers you to a related or contrasting term.

To view glossaries for other IBM products, go to www.ibm.com/software/globalization/terminology.

Numerics

16-bit code set
See double-byte character set.

8-bit character
A single-byte character that consists of eight bits, which means that the code point is in the range 128 through 255. See also single-byte character.

8-bit clean
An attribute of software that can process character data that contains 8-bit characters. The operating system or the database server reads the eighth bit as part of the code value.

A

access method
A set of routines the database server uses to manipulate and/or access a table, index or other object.

access privilege
An authority that relates to a request for a type of access to data.

ActiveX value object
A Microsoft Common Object Model (COM)-compliant object that contains a client-side copy of an opaque type and its support routines.

aggregate function
A function that optionally accepts arguments and returns a single scalar value that is the result of an evaluation of a set of like values, such as those in a column within a set of one or more rows. See also function, routine.

aggregate support function
One of a group of user-defined functions that the database server uses to calculate a user-defined aggregate.

alias In an SQL query or in a form-specification file, a single-word temporary alternative name that is used in place of a qualified table name (for example, t1as an alias for owner.table_name). Aliases are often used in complex subqueries and are required for a self-join.

American Standard Code for Information Interchange
A standard code used for information exchange among data processing systems, data communication systems, and associated equipment. ASCII uses a coded character set consisting of 7-bit coded characters.

AP

1. See application program.

2. See application process.

applet A program that performs a specific task and is typically portable between operating systems. Often written in Java, applets can be downloaded from the Internet and run in a web browser.

application process
A unit to which resources and locks are allocated. An application process involves the running of one or more programs.

application program
A complete, self-contained program, such as a text editor or a web browser, that performs a specific task for the user, in contrast to system software, such as the operating system kernel, server processes, and program libraries.

arbitrary rule
A rule that defines the limits of an SQL query that uses any relational or logical operators to define expressions. See also range rule.

archive
To copy programs, data, or files to another storage media, usually for long-term storage or security.

argument
A value passed to or returned from a function or procedure at run time.

arithmetic function
A function that returns a value by performing a mathematical operation on one or more arguments.

array A structure that contains an ordered collection of elements of the same data type in which each element can be referenced by its index value or ordinal position in the collection. See also element.

ASCII See American Standard Code for Information Interchange.

asynchronous disk I/O virtual processor
A virtual processor that performs nonlogging disk input/output. See also virtual processor.

attached index
An index that has the same distribution scheme as its table. See also detached index, system index, user index.

audit event
Any database server activity or operation that can potentially access or alter data.

audit file
A file that contains records of audit events and resides in the specified audit directory. Audit files provide an audit trail of information that can be extracted by the database secure auditing facility for analysis.

audit mask
A structure that specifies which audit events should be logged (or excluded from logging) by the database secure auditing facility.

authenticated user
A user whose identify has been verified by the system.

autocommit mode
A mode in which a COMMIT statement is automatically executed after each statement sent to the database server.

B

backup
The process of making a copy of a data file that can be used if the original file is destroyed.

base table
In Informix, within a table hierarchy, a parent table that can be used to create child tables based on the parent's schemas.

base type
See opaque data type.

binary large object
A data type whose value is a sequence of bytes that can range in size from 0 bytes to 2 gigabytes less 1 byte. This sequence does not have an associated code page and character set. BLOBs can contain, for example, image, audio, or video data. See also large object.

bitmap index
A type of index that stores a bitmap for any duplicated key value.

BLOB See binary large object.

blobpage
A unit of disk allocation within a blobspace. The size can vary, depending on the size of the TEXT or BYTE data that the user inserts.

blobspace
A logical collection of blobpages that are used to store TEXT and BYTE data.

Boolean
Characteristic of an expression or variable that can only have a value of true or false.

bootstrap
A small program that starts a computer by loading the operating system and other basic software.

branch node
An index page that contains pointers to a leaf node or other branch nodes. The database server creates a branch node when the root node and subsequent leaf nodes become full. See also node.

branch page
A location on a tree structure that has at least one page below and one page above it. In an R-tree index, branch pages are located in the intermediate levels, between the root page and leaf pages.

B-tree index
An index that is arranged as a balanced hierarchy of pages and that minimizes access time by realigning data keys as items are inserted or deleted.

buffer An area of storage that compensates for the different speeds of data flow or timings of events by temporarily holding a block of data to be processed or written to an I/O device.

buffered disk I/O
Disk I/O that is controlled by the operating system instead of by an application. With buffered disk I/O, the operating system stores data in the kernel portion of memory before periodically writing the data to disk. See also disk I/O, unbuffered disk I/O.

buffered logging
A type of logging that holds transactions in a memory buffer until the buffer is full, regardless of when the transaction is committed or rolled back.

built-in cast
A cast that is built into the database server. A built-in cast performs automatic conversions between different built-in data types.

built-in data type
A data type defined by the database server. See also data type.

built-in function
A predefined, SQL-invoked function that provides some basic arithmetic and other operations, such as cos(), log(), or today().

C

CA See certificate authority.

Cartesian product
The resulting set from performing a multiple-table query without any specified joining conditions among the tables.

cascading delete
The process of deleting rows from a child table when the foreign key is deleted from the parent table. When any rows are deleted from the primary key column of a table, cascading deletes, if enabled, delete identical information from any foreign-key column in a related table.

cast A database object and an operator for converting data from one data type to another. Built-in data types have built-in casts to compatible data types within database server. See also explicit cast, user-defined cast, implicit cast.

cast function
A function that is used to convert instances of a source data type into instances of a different target data type. In general, a cast function has the name of the target data type and has one single argument whose type is the source data type. Its return type is the target data type.

cast support function
A function that is used to implement an implicit or explicit cast by performing the necessary operations for conversion between two data types.

certificate authority
A trusted third-party organization or company that issues the digital certificates. The certificate authority typically verifies the identity of the individuals who are granted the unique certificate. See also Secure Sockets Layer.

character large object
A data type whose value is a sequence of characters (single byte, multibyte, or both) that can range in size from 0 bytes to 2 gigabytes less 1 byte. In general, the CLOB data type is used whenever a character string might exceed the limits of the VARCHAR data type. See also large object.

check constraint
A user-defined constraint that specifies the values that specific columns of a base table can contain. See also constraint.

checkpoint
A point at which the database manager records internal status information in the log; the recovery process uses this information if the subsystem abnormally terminates.

child table
A table that has a referential constraint to a column in a different table; the referenced table is called a parent table. See also parent table.

chunk A data storage location. For example, a raw disk device or a cooked file.

class A description of a set of objects that share the same attributes, operations, methods, relationships, and semantics. A class may use a set of interfaces to specify collections of operations that it provides to its environment.

CLASSPATH
In the execution environment, a variable that specifies the directories in which to look for class and resource files.

client application
An application that users the services of the database services by direct connection or via application servers. See also client/server architecture.

client locale
The locale that a client application uses to perform read and write operations on the client computer. See also locale, server locale.

client/server
Pertaining to the model of interaction in distributed data processing in which a program on one computer sends a request to a program on another computer and awaits a response. The requesting program is called a client; the answering program is called a server.

client/server architecture
A hardware and software design that allows the user interface and database server to reside on separate nodes or platforms on a single computer or over a network. See also server-processing locale, client application.

client/server connection statement
An SQL statement that can connect to a database. These statements include CONNECT, DISCONNECT, and SET CONNECTION.

CLOB See character large object.

clone An identical copy of the data and configuration of a server at a particular point in time.

cloud A network that delivers requested virtual resources as a service.

cloud storage
A storage resource provided by a cloud, or the storage of data on virtual public or private servers in the cloud.

cluster
A group of servers connected by a network and configured in such a way that if the primary server fails, a secondary server takes over.

code set
A set of unambiguous rules that establish a character set and the one-to-one relationship between each character of the set and its bit representation.

collating sequence
The sequence in which the characters are ordered for the purpose of sorting, merging, comparing, and processing indexed data sequentially.

collation
The logical ordering of characters and strings according to defined rules.

collation order
The logical order that character-string values in a database are sorted and indexed by. The ordering is either based on the order of the code set or a locale-specific order.

collection
An instance of a collection data type; a group of elements of the same data type stored in a SET, MULTISET, or LIST data type.

collection cursor
A database cursor that has an IBM Informix ESQL/C collection variable associated with it and provides access to the individual elements of a column whose data type is a collection data type.

collection data type
A complex data type whose instances are groups of elements of the same data type, which can be any opaque data type, distinct data type, built-in data type, collection data type, or row data type. See also complex data type.

collection-derived table
In Informix, a table that can be mapped to a transient table, and its elements can be mapped to rows of the transient table.

collection subquery
A collection subquery enables users to construct a collection expression from a subquery expression.

collection variable
An IBM Informix ESQL/C host variable or SPL variable that holds an entire collection and provides access, through a collection cursor, to the individual elements of the collection.

column distribution value
See data distribution.

column expression
An expression that includes a column name and optionally uses column subscripts to define a column substring.

command file
A system file that contains one or more statements or commands.

commit
To end a unit of work by releasing locks so that the database changes made by that unit of work can be perceived by other processes. This operation makes the data changes permanent.

committed read
In Informix, an isolation level under which a query in a transaction can read only rows that are committed at the moment when the query is requested. The user cannot view rows that were changed as a part of a currently uncommitted transaction. Committed read is available through a database server and set with the SET ISOLATION statement. It is the default level of isolation for databases that are not ANSI compliant. See also uncommitted read, read committed.

commutator function
A Boolean function whose arguments are the reverse of, and evaluates to the same result as another Boolean function. Commutator functions might execute quicker depending on the nature of the query.

compatible data types
Two different data types that can be cast to one another in the database. See also implicit cast.

compile
To translate all or part of a program expressed in a high-level language into a computer program expressed in an intermediate language, an assembly language, or a machine language.

complex data type
A data type that is built from a combination of other data types by using an SQL type constructor and whose components can be accessed through SQL statements. See also collection data type, row data type, data type.

complex qualification
A WHERE clause in a query that uses two or more logical operators on the same column as where the R-tree index is defined.

composite data type
See row data type.

composite index
An index constructed on two or more columns of a table. The order imposed by the composite index varies least frequently on the first-named column and most frequently on the last-named column.

compressed bitmap
An indexing method that identifies records through a fragment identifier and a record identifier.

concatenation operator
The symbol used to join two character data items. The concatenation operator is often represented as two vertical bars (||).

concurrency
The shared use of resources by multiple interactive users or application processes at the same time.

configuration file

1. A file that contains the values of configuration parameters.

2. A file read during database server disk or shared-memory initialization that contains the parameters that specify values for configurable behavior. A database server and its archiving tool use configuration files. See also parameter.

connection
In data communication, an association established between entities for conveying information. See also SQL connection.

Connection Manager
A deamon program that manages and redirects client connection requests based on service level agreements configured by the system administrator.

constraint
A rule that limits the values that can be inserted, deleted, or updated in a table. See also check constraint, primary key, referential constraint, unique constraint, unique key.

constructed data type
A complex data type created with a type constructor.

coordinating server
In a query that spans multiple database servers, the server in which the query is initiated. To respond to the query, the coordinating server starts sessions on the other servers involved in the query. See also subordinate server, distributed query, remote table.

correlated subquery
A subquery that contains a correlated reference.

corrupted index
An index that does not correspond exactly to the data in its table.

cross-server query
See distributed query.

current row
The most recently retrieved row of the active set of a query.

cursor In SQL, an identifier associated with a group of rows or with a collection.

cursor function
A user-defined routine that returns one or more rows of data and therefore requires a cursor to execute. See also function cursor, noncursor function.

cursor manipulation statement
An SQL statement that controls cursors; specifically, the CLOSE, DECLARE, FETCH, FLUSH, OPEN, and PUT statements.

cursor stability
An isolation level that locks any row accessed by a transaction of an application while the cursor is positioned on the row. The lock remains in effect until the next row is fetched or the transaction is terminated. If any data is changed in a row, the lock is held until the change is committed to the database.

D

data access statement
A subset of SQL statements that can be used to grant and revoke permissions and to lock tables.

database administrator
A person who is responsible for the design, development, operation, maintenance, and use of a database.

database extension
A group of built-in database objects and supporting code that extends an Informix database server to manage specialized data or add new features. For example, the *basic text search* database extension enables searching for words and phrases in a document repository stored in a column of a table. See also DataBlade® module.

database management system
A software system that controls the creation, organization, and modification of a database and the access to the data that is stored within it.

database object
An object that a user creates in the database, such as a procedure, trigger, or any other object that can be created by issuing a CREATE statement.

database security administrator
A person who has the authority to implement and maintain label-based access control.

database server
A software program that uses a database manager to provide database services to other software programs or computers.

database server instance
A logical database server environment that consists of configuration files, directories, and a set of authorized users

DataBlade module
A package of database objects and supporting code that extends an Informix database server to manage new kinds of data or add new features. A DataBlade module can include new data types, routines, casts, aggregates, access methods, SQL code, and client code. See also database extension.

Data Definition Language
A language for describing data and its relationships in a database. See also Data Manipulation Language.

data definition statement
A subset of SQL statements that can be used to create, alter, drop, and rename data objects, including databases, tables, views, synonyms, triggers, sequences, and user-defined routines. See also declaration statement.

data dictionary
A set of tables that keep track of the structure of both the database and the inventory of database objects.

data distribution
A mapping of the data values within a column into a set of categories that are equivalent to a histogram or to a frequency distribution. See also query optimizer.

data file
A flat file containing data to be loaded into the database.

data integrity
The condition that exists as long as accidental or intentional destruction, alteration, or loss of data does not occur.

data integrity statement
A subset of SQL statements that can be used to control transactions and audits. Data integrity statements also include statements for repairing and recovering tables.

Data Manipulation Language
A subset of SQL statements that is used to manipulate data. Most applications primarily use DML SQL statements, which are supported by the DB2 Connect™ program. SELECT, INSERT, UPDATE, and DELETE statements are similar across the IBM relational database products. See also Data Definition Language, Structured Query Language.

data manipulation statement
A subset of SQL statements that can be used to query tables, insert into tables, delete from tables, or update tables (select, insert, delete, update).

data object
The data that is stored in an R-tree indexed column of a table and in the R-tree index itself.

data restriction
See constraint.

data type
In Informix, a descriptor that is assigned to each column in a table, function argument, or function return type that indicates the type of data that can be held. See also distinct data type, complex data type, built-in data type, user-defined data type, opaque data type.

DBA See database administrator.

DBA-privileged
A class of SPL routines that only a user with DBA database privileges creates.

DBCS See double-byte character set.

DBMS
See database management system.

dbspace
The logical unit of storage in an Informix database. For example, a table or table fragment is created in a dbspace. See also root dbspace.

DDL See Data Definition Language.

deadlock
Unresolved contention for the use of resources.

decision-support application
An application that provides information that is used for strategic planning, decision-making, and reporting. It typically executes in a batch environment in a sequential scan fashion and returns a large fraction of the rows scanned. Decision-support queries typically scan the entire database.

decision-support query
A query that a decision-support application generates. A decision support query often requires multiple joins, temporary tables, and extensive calculations, and can benefit significantly from PDQ

declaration statement
A programming language statement that describes or defines objects; for example, defining a program variable. See also data definition statement, procedure.

delimited identifier
A sequence of one or more characters enclosed by quotation marks ("").

delimiter
A flag that is formed by a character or a sequence of characters to group or separate items of data by marking the beginning and end of a unit of data. The delimiter is not a part of the flagged unit of data.

deploy
To place files or install software into an operational environment.

deployment assistant
A computer program for packaging a snapshot of an Informix database server instance in preparation for deploying the instance.

deployment utility
A computer program for deploying a snapshot of an Informix database server instance to multiple target computers.

derived table
The set of rows returned by a subquery within a DML statement of SQL. Derived tables can be defined by simple, UNION, nested, or joined subqueries, including OUTER joins.

descriptor
In Informix, a quoted string or variable that identifies an allocated system-descriptor area or an SQLDA structure. See also system-descriptor area, identifier.

detached index
In a table with indexes, an index that uses a fragmentation strategy that is different from the table fragmentation. See also attached index.

diagnostic area
A data structure that stores diagnostic information about an executed SQL statement.

diagnostics table
A special table that contains information about integrity violations that were caused by the rows of a violations table. See also violations table.

digital certificate
An electronic document used to identify an individual, a system, a server, a company, or some other entity, and to associate a public key with the entity. A digital certificate is issued by a certification authority and is digitally signed by that authority.

dirty read
A read request that does not involve any locking mechanism. This means that data can be read that might later be rolled back resulting in an inconsistency between what was read and what is in the database. See also read uncommitted.

disabled mode
The object mode in which a database object is disabled. When a constraint, index, or trigger is in the disabled mode, the database server acts as if the

object does not exist and does not take it into consideration during the execution of data manipulation statements.

disk configuration
The organization of data on a disk.

disk I/O
Fixed-disk input and output. See also buffered disk I/O.

display label
A temporary name for a column or an expression in a query.

distinct data type
In SQL, a data type derived from another data type. A distinct type always has the same internal storage representation as its source type (such as an existing opaque data type, built-in data type, named row type, or distinct data type). It might have the same casts and routines as its source type, however it must have a unique name. See also data type.

distributed query
A query that accesses data from a database other than the current database. See also coordinating server.

Distributed Relational Database Architecture™
The architecture that defines formats and protocols for providing transparent access to remote data. DRDA® defines two types of functions: the application requester function and the application server function.

DLL See dynamic link library.

DML See Data Manipulation Language.

dominant table
A table that has its data preserved over the course of an outer join.

dormant connection
A connection that is suspended and that is not used by SQL statements. See also SQL connection.

double-byte character set
A set of characters in which each character is represented by two bytes. These character sets are commonly used by national languages, such as Japanese and Chinese, that have more symbols than can be represented by a single byte.

DRDA
See Distributed Relational Database Architecture.

duplicate index
An index that allows duplicate values in the indexed column.

dynamic link library
A file containing executable code and data bound to a program at load time or run time, rather than during linking. The code and data in a DLL can be shared by several applications simultaneously. See also shared-object file, library.

dynamic management statement
An SQL statement that describes, executes, and prepares other statements.

dynamic routine-name specification
The execution of a user-defined routine whose name is determined at run time through an SPL variable in the EXECUTE PROCEDURE, EXECUTE ROUTINE, or EXECUTE FUNCTION statement.

dynamic SQL
An SQL statement that is prepared and executed at run time. In dynamic SQL, the SQL statement is contained as a character string in a host variable and is not precompiled.

E

element
In Informix, a member of a collection, such as a LIST, MULTISET, or SET data type. An element can be a value of any built-in data type, opaque data type, distinct data type, named row type, unnamed row type, or collection data type.

enabled mode
The default object mode in which database objects are enabled. When a constraint, index, or trigger is in this mode, the database server recognizes the existence of the object and takes the object into consideration while executing data manipulation statements.

end-user format
The format in which data appears within a client application as literal strings or character variables. End-user formats are useful for data types whose database format is different from the format to which users are accustomed.

end-user routine
A user-defined routine (UDR) that performs a task within an SQL statement that the existing built-in routines do not perform. Examples of tasks include encapsulating multiple SQL statements, creating trigger actions, and restricting who can access database objects.

environment variable
A variable that defines an aspect of the operating environment for a process. For example, environment variables can define the home directory, the command search path, the terminal in use, or the current time zone.

error trapping
See exception handling.

escape character
A character that suppresses or selects a special meaning for one or more characters that follow.

exception
A condition or event that cannot be handled by a normal process.

exception handling
The performance of a specified response to an abnormal condition. Exception handling allows control and information to be passed to an exception handler when an exception occurs.

exclusive access
A condition that permits write access to a single user only.

exclusive lock
A lock that prevents concurrently executing application processes from accessing database data. See also shared lock.

executable file
A file that contains programs or commands that perform operations on actions to be taken.

exemption
In label-based access control, a privilege that causes one LBAC rule of one security policy to be bypassed for a user or a set of users to whom that privilege is granted. See also LBAC credentials.

explicit cast
A specifically invoked cast using either the CAST AS keywords or the cast operator (::). The database server does not automatically invoke an explicit cast to resolve data type conversions. See also implicit cast, cast.

explicit connection
A connection to a database in which both a user ID and password are specified. See also implicit connection.

explicit transaction
A transaction that is initiated by the BEGIN WORK statement. This type of transaction is available only in non-ANSI compliant databases that support logging. See also implicit transaction, singleton implicit transaction.

expression-based fragmentation
A fragmentation process that puts rows that contain specified values in the same fragment.

extendable chunk
A chunk that Informix can automatically extend or that you can manually extend when additional storage space is required for an application.

extended data type
A term used to refer to data types that are not built in; namely complex data types, opaque data types, and distinct data types.

extent An allocation of space, within a container of a table space, to a single database object. This allocation consists of multiple pages.

external function
A function that has its functional logic implemented in a programming language application that is outside the database, in the file system of the database server. The association of the function with the external code application is specified by the EXTERNAL clause in the CREATE FUNCTION statement. See also external routine, function.

externally authenticated user
A user who is authenticated by an external service (such as PAM or Kerberos).

external procedure
A procedure that has its procedural logic implemented in an external programming language application. The association of the procedure with the external application is specified by a CREATE PROCEDURE statement with a LANGUAGE clause that has a value other than SQL and an EXTERNAL clause that implicitly or explicitly specifies the name of the external application. See also external routine, procedure.

external routine
A function, method, or procedure that has its routine logic implemented in a programming language application that is outside the database, in the file system of the database server. The association of the routine with the external code application is specified by the EXTERNAL clause in the CREATE statement for the routine. See also external function, external procedure, routine.

external space
Storage space that a user-defined access method manages rather than the database server. The IN clause of the CREATE TABLE and CREATE INDEX statements can specify the name of an external space instead of a dbspace.

external table
Data organized in table format that is stored externally to the database server.

extspace
A logical name associated with an arbitrary string that signifies the location of external data.

F

fault tolerance
The ability of a system to continue to operate effectively after the failure of a component part. See also high availability.

feature
Part of a product that is either included with the product or can be ordered separately.

fetch An SQL action that positions a cursor on the next row of its result table and assigns the values of that row to host variables.

fetch buffer
A buffer in the application process that the database server uses to send fetched row data (except TEXT and BYTE data) to the application.

field An area into which a particular category of data or control information is entered.

FIFO See first-in first-out.

filter In an SQL query, a set of conditions that impact on how a row or record is selected. The conditional expression in the WHERE clause is a filter that controls the active set of the query.

filtering mode
An object mode of a database object, that causes bad rows to be written to the violations table during DML operations. During DML operations, the database server enforces requirements of a constraint or of a unique index that is in this mode and identifies any rows that would violate the requirement.

first-in first-out
A queuing technique in which the next item to be retrieved is the item that has been in the queue for the longest time. See also last-in first-out.

fixed-point number
A number where the decimal point is fixed at a specific place regardless of the value of the number.

flexible temporary table
An explicit temporary table that Extended Parallel Server automatically fragments using a round-robin distribution scheme.

FLRU See free least-recently used.

forest of trees index
A B-tree index that is divided into smaller subtrees. The subtrees contain

multiple root nodes and leaves. Multiple root nodes can alleviate root node contention because concurrent users can access the index. Subtrees can improve query performance by reducing the number of levels involved in buffer read operations. See also B-tree index.

FOT index
See forest of trees index.

fragment
A portion of a table or index.

fragmentation

1. A data organization scheme in which index or table data is divided across multiple storage objects.

2. A process that enables the user to define groups of rows or index keys within a table according to some algorithm or scheme in order to control where data is stored at a table level.

fragment elimination
The process of applying a filter predicate to the fragmentation strategy of a table or index and removing the fragments that do not apply to the operation.

free least-recently used
A list that tracks free or unmodified pages in a queue.

function
A relationship between a set of input data values and a set of result values that is used to extend and customize SQL or XQuery. Functions are invoked from elements of SQL statements such as the select list or the FROM clause and from XQuery as primary expressions. See also aggregate function, built-in function, external function, user-defined function, routine.

functional index
An index that stores the result of executing a specified function on a table column.

function cursor
A cursor that is associated with an EXECUTE FUNCTION statement, which executes routines that return values. See also cursor function.

function overloading
See routine overloading.

fundamental data type
In databases, a data type which represents the smallest unit that can be specified in a SELECT statement.

G

G11N See globalization.

global catalog
A table that contains a global inventory of enterprise replication configuration information.

globalization
In computing, the provision of a single software solution that has multicultural support, and a user interface and documentation that is available in one or more languages.

global variable
A variable or identifier whose scope of reference is all of the modules in a program. See also local variable, scope of reference.

grid A named set of interconnected replication servers for propagating commands from an authorized server to the rest of the servers in the set.

H

hash rule
A user-defined algorithm that maps each row in a table to a set of hash values and that is used to determine the fragment in which a row is stored.

HDR snapshot
See snapshot.

heterogeneous commit
A protocol governing a group of database servers, of which at least one is a gateway participant. A heterogeneous commit ensures the all-or-nothing basis of distributed transactions in a heterogeneous environment.

hierarchy
The tree-like arrangement of segments in a database, beginning with the root segment and proceeding down to dependent segments.

high availability
The ability of IT services to withstand all outages and continue providing processing capability according to some predefined service level. Covered outages include both planned events, such as maintenance and backups, and unplanned events, such as software failures, hardware failures, power failures, and disasters. See also fault tolerance.

high-availability cluster
See cluster.

hold cursor
A special type of cursor which is not closed at the end of a transaction. The database server still closes all other cursors, and it still releases all locks, but the hold cursor remains open until it is explicitly closed.

host variable
In an application program, a variable that is referred to by embedded SQL statements. Host variables are programming variables in the application program and are the primary mechanism for transmitting data between tables in the database and application program work areas.

I

IANA See Internet Assigned Numbers Authority.

identifier
A sequence of bits or characters that serves as the unqualified name for a database, storage device, or program object. See also descriptor.

implicit cast
A built-in or user-defined cast using either the IMPLICIT keyword or automatically invoked by the database server to perform data-type conversion when faced with non-comparable types (such as an INTEGER compared with a CHAR). See also explicit cast, compatible data types, cast.

implicit connection
A connection that is made to a database without a user ID or password. See also explicit connection.

implicit transaction
A transaction that begins implicitly after the COMMIT WORK or ROLLBACK WORK statement. This is the only type of transaction that ANSI-compliant databases support, but it is also available for other databases that support logging. See also explicit transaction, singleton implicit transaction.

imported restore
A process in which a backup is taken of an instance on one server and restored on another server.

indexed sequential access method
An access method that can be used for either direct or sequential update or retrieval.

index self-join
A type of index scan that is a union of many small index scans, each one with a single unique combination of lead-key columns and filters on non-lead-key columns. The union of small index scans results in an access path that uses only subsets of the full range of a composite index.

initialize
To prepare a system, device, or program for operation; for example, to initialize a diskette.

input parameter
In a prepared SQL statement, a value, represented by a "?" placeholder symbol, that must be provided when the prepared statement is executed. See also parameter.

insert cursor
The position of the cursor marking where new characters will be added when entering text.

instantiation
Creation of a database server instance.

Internet Assigned Numbers Authority
An organization that defines a hierarchy for naming tables and columns and for deriving numerical object identifiers (OIDs). IANA assigns identifiers to companies that use the SNMP protocol.

interquery parallelism
The ability to process multiple queries simultaneously to avoid a performance delay when multiple independent queries access the same table. See also intraquery parallelism.

interrupt key
A key used to cancel or abort a program or to leave a current menu and return to the menu one level above.

intraquery parallelism
The ability to process parts of a single query at the same time using intra-partition parallelism, inter-partition parallelism, or both. See also interquery parallelism.

ISAM See indexed sequential access method.

isolation level
An attribute that defines the degree to which an application process is isolated from other concurrently executing application processes. Isolation levels generally relate to the behavior of an application with respect to locks. See also cursor stability, uncommitted read.

iterator function
A cursor function that returns a data set by successive calls. It can be written in C or Java or can be an SPL routine that includes RETURN WITH RESUME.

J

jagged rows
A query result in which rows differ in the number and type of columns that they contain because the query applies to more than one table in a table hierarchy.

K

kernel The part of an operating system that contains programs for such tasks as input/output, management and control of hardware, and the scheduling of user tasks.

key A column or an ordered collection of columns that is identified in the description of a table, index, or referential constraint. The same column can be part of more than one key.

keystore
In security, a storage object, either a file or a hardware cryptographic card, where identities and private keys are stored, for authentication and encryption purposes. Some keystores also contain trusted, or public, keys.

L

label-based access control
A security mechanism that uses security labels to restrict user access to individual table rows and columns. See also security label, security policy.

large object
A data object whose data type supports the storage and manipulation of more data than most other data types.

last-in first-out
A queuing technique in which the next item to be retrieved is the item most recently placed on the queue. See also first-in first-out.

LBAC See label-based access control.

LBAC credentials
In label-based access control, the set of security labels and exemptions held by a database user. See also exemption.

leaf page
A page that contains pairs of keys and record identifiers and that points to actual data.

least recently used
Pertaining to an algorithm used to identify and make available the cache space that contains the data that was least recently used.

level of isolation
See isolation level.

library
In Informix, a group of precompiled routines designed to perform tasks that are common to a given kind of application. See also dynamic link library, shared-object file.

LIFO See last-in first-out.

light append
An unbuffered, unlogged insert operation. See also raw table.

link In Informix, a way of combining separately compiled program modules into one module that is typically used in an executable program.

LIST data type
A collection data type created with the LIST constructor in which elements are ordered and duplicates are allowed.

load job
The information required to load data into a relational database using the HPL. This information includes the format, map, filter, device array, project, and special options.

LOB See large object.

locale A setting that identifies language or geography and determines formatting conventions such as collation, case conversion, character classification, the language of messages, date and time representation, and numeric representation. See also server-processing locale, client locale.

localized
In national language support, pertaining to the support based on a user's national language, country or region, culture, and character encoding.

localized order
A collation order other than code-set order, if defined for a locale. Only NCHAR and NVARCHAR data values are collated in a localized order. Database objects collate in their creation-time order, if this is not the runtime order.

local loopback
A connection between the client application and database server that uses a network connection even though the client application and the database server are on the same computer.

local variable
A symbol defined in one program module or procedure that can only be used within that program module or procedure. See also scope of reference.

lock A means of preventing uncommitted changes made by one application process from being perceived by another application process and for preventing one application process from updating data that is being accessed by another process. A lock ensures the integrity of data by preventing concurrent users from accessing inconsistent data.

locking granularity
The level and type of information that a lock protects.

lock mode
A representation for the type of access that concurrently running programs can have to a resource that a lock is holding.

loop label
In SPL routines, an SQL identifier whose declaration immediately precedes an iterative statement, and whose name immediately follows the terminating END FOR, END LOOP, or END WHILE keywords. The EXIT label statement can transfer program control to the first executable statement that follows the last statement in the FOR, FOR LOOP, LOOP, WHILE LOOP, or WHILE loop that has the specified loop label.

LRU See least recently used.

M

manifest file
In Informix, an ASCII file, which must not be customized, that is automatically created during installation of a database server instance. The file contains a history of installation activity. See also response file.

mapped user name
In Informix, an operating system identity for users who do not have operating system user accounts. The mapped user name is either mapped to an operating system account or to a default set of properties.

master replicate
An Enterprise Replication replicate that guarantees data integrity by verifying that replicated tables on different servers have consistent column attributes. See also shadow replicate, replicate.

member
In SQL, a component of an opaque data type that can be accessed by an SQL statement by using a user-defined accessor function.

Memory Grant Manager
A database server component that coordinates the use of memory and I/O bandwidth for decision-support queries.

MGM See Memory Grant Manager.

mirroring
The process of writing the same data to multiple disks at the same time. The mirroring of data protects it against data loss within the database or within the recovery log.

MLRU
See modified least-recently used.

modified least-recently used
A list that tracks modified pages in a queue.

multibyte character
A mixture of single-byte characters from a single-byte character set and double-byte characters from a double-byte character set. See also single-byte character.

multicultural support
In computing, the ability of a single software solution to be translatable and to support the cultural conventions of multiple languages and geographic regions. Cultural conventions include the use of various writing

systems, sort orders, different formats for date, time, numbers, and currency, different keyboard layouts, and so on.

multiplexed connection
A single network connection between a database server and a client application that handles multiple database connections from the client.

MULTISET constructor
A type constructor used to create a MULTISET data type.

MULTISET data type
A collection data type created with the MULTISET constructor in which elements are not ordered and duplicates are allowed.

multithreading
A mode of operation in which the operating system can run different parts of a program, called threads, simultaneously.

N

named row data type
A row data type created with the CREATE ROW TYPE statement that has a declared name and inheritance properties and can be used to construct a typed table. See also row data type.

node

1. In the context of an index for a database, an ordered group of key values having a fixed number of elements. A B-tree for example, is a set of nodes that contain keys and pointers arranged in a hierarchy. See also branch node, root node.

2. In hardware, a uniprocessor or symmetric multiprocessor (SMP) computer that is part of a clustered system or a massively parallel processing (MPP) system. See also symmetric multiprocessing system.

noncursor function
A user-defined function that returns a single group of values (one row of data) and therefore does not require a cursor when it is executed. Compare with cursor function. See also cursor function.

nonvariant function
A user-defined function that always returns the same value when passed the same arguments. A nonvariant function must not contain SQL statements. Compare with variant function. See also variant function.

O

object identifier
A hierarchical sequence of numbers that uniquely identifies an object.

object-relational database
A database that adds object-oriented features to a relational database, including support for user-defined data types, user-defined routines, user-defined casts, user-defined access methods, and inheritance.

OID See object identifier.

OLTP See online transaction processing.

online transaction processing
A type of interactive application in which requests submitted by users are

processed as soon as they are received. Results are returned to the requester in a relatively short period of time.

opaque data type
A data type whose inner structure is not visible to the database server. Opaque types that are not built-in need user-defined routines and user-defined operators that work on them. See also data type.

operational table
A logging permanent table that uses light appends for fast update operations. Operational tables do not perform record-by-record logging.

operator-class function
One of the operator-class support functions or operator-class strategy functions that constitute an operator class. For user-defined operator classes, the operator-class functions are user-defined functions.

operator-class strategy function
An operator-class function that can appear as a filter in a query. The query optimizer uses the strategy functions to determine if an index of a particular secondary access method can be used to process the filter.

operator-class support function
An operator-class function that a secondary access method uses to build or search an index.

operator function
An arithmetic function that has a corresponding operator symbol. An operator function processes one to three arguments and returns a value.

owner-privileged
Pertaining to a class of SPL routines that any user can create who has Resource database privileges.

P

packed decimal
A storage format that represents either two decimal digits or a sign and one decimal digit in each byte.

page In Informix, the physical unit of disk storage and basic unit of memory storage that the database server uses to read from and write to a database. Page size is fixed for a particular operating system and platform.

parallel database query
SQL queries made in parallel rather than sequential order. The tasks that a query requires are distributed across several processors. This type of distribution enhances database performance.

parallelizable routine
A routine that can be executed within a parallel database query statement.

parallel processing platform
A parallel-processing platform is a set of independent computers that operate in parallel and communicate over a high-speed network, bus, or interconnect. See also symmetric multiprocessing system.

parameter
A value or reference passed to a function, command, or program that serves as input or controls actions. The value is supplied by a user or by another program or process. See also configuration file, routine signature, input parameter.

parent table
A table that is a parent in at least one referential constraint. See also child table.

parm See parameter.

participant
In Enterprise Replication, the data (database, table, and columns) to replicate and the database servers to which the data replicates.

partition

1. A named fragment of a table or index.

2. A logical division of storage on a fixed disk.

partitioning
See fragmentation.

PDQ See parallel database query.

PDQ priority
A measure of priority that determines the amount of resources that a database server allocates to process a query in parallel. These resources include memory, threads (such as scan threads), and sort space.

phantom row
A row of a table that is initially modified or inserted during a transaction but is subsequently rolled back.

physical log
A log that stores unmodified copies (before-image) of pages in shared memory. The pages in the physical log can be any database server page except a blobspace blobpage.

pointer
A data element or variable that holds the address of a data object or a function.

precision
An attribute of a number that describes the total number of binary or decimal digits.

predicate
An element of a search condition that expresses or implies a comparison operation.

predicate lock
A lock held on index keys that qualifies for a predicate. In a predicate lock, exclusive predicates consist of a single key value, and shared predicates consist of a query rectangle and a scan operation such as inclusion or overlap.

prepared statement
An SQL statement that is generated by the PREPARE statement from a character string or from a variable that contains a character string.

prepare script
A file containing SQL statements that describe the DataBlade module.

preprocessor
A routine that processes source code before the code is compiled, resulting in altered source code.

primary key
In a relational database, a key that uniquely identifies one row of a database table. See also constraint, unique key.

primary-key constraint
A constraint that specifies that each entry in a column or set of columns contains a unique non-null value.

primary server
A database server participating in a high-availability configuration. A primary server permits read and write access from client applications and owns the logical logs that are sent to secondary servers.

primary user name
For authenticated users. the name that is used to determine which system resources the users are authorized to access.

privilege
The right to use or change the contents of a database, table, table fragment, or column.

privileged user
In Informix, a user who was granted one or more administrator roles.

procedure
A routine that does not return any value. See also declaration statement, user-defined procedure.

procedure overloading
See routine overloading.

process
An instance of a program running on a system and the resources that it uses.

projection
A mapping from a set to a subset of it.

promotable lock
A lock that can be changed from a shared lock to an exclusive lock. See also update lock.

protocol
A set of rules controlling the communication and transfer of data between two or more devices or systems in a communication network.

purpose function
One of a set of functions that an access method uses to create, search, and drop indexes, and to insert entries into an index, delete from an index, and so on.

Q

query A request for information from a database that is based on specific conditions: for example, a request for a list of all customers in a customer table whose balances are greater than USD1000.

query optimization information statement
An SQL statement that is used to optimize the performance of queries. These statements include SET EXPLAIN, SET OPTIMIZATION, and UPDATE STATISTICS.

query optimizer
A component of the SQL and XQuery compiler that chooses an access plan for a data manipulation language statement by modeling the execution cost of many alternative access plans and choosing the one with the minimal estimated cost. See also data distribution.

R

range fragmentation
A distribution scheme that distributes data in table fragments that contain a specified key range. This technique can eliminate scans of table fragments that do not contain the required rows, making queries faster.

range rule
A user-defined algorithm for expression-based fragmentation. It defines the boundaries of each fragment in a table using SQL relational and logical operators. Expressions in a range rule can use the following restricted set of operators: >, <, >=, <=, and the logical operator AND. See also arbitrary rule.

raw table
A nonlogged permanent table that uses light appends. See also light append.

read committed
An ANSI compliant level of isolation that the SET TRANSACTION statement can specify, in which a user can view rows that are currently committed at the moment of the query request, but cannot view rows that were changed as part of a currently uncommitted transaction. This is the default isolation level for databases that are not ANSI compliant. See also committed read.

read uncommitted
An ANSI-compliant level of isolation, set with the SET TRANSACTION statement, that does not account for locks. This allows a user to view any existing rows, even rows that can be altered within currently uncommitted transactions. Read uncommitted is the lowest level of isolation (no isolation at all), and is thus the most efficient. See also dirty read.

record The storage representation of a row or other data. See also row.

Record-ID
A four-byte RSAM entity that describes the logical position of the record within a fragment.

recover time objective
In disaster recovery planning, the length of time that a system can be offline.

recovery point objective
In disaster recovery planning, the point at which data is restored to in the event of a disaster.

recovery time objective
In disaster recovery planning, the total time one can allow for their systems to be offline.

referential constraint
The requirement that the nonnull values of a designated foreign key are valid only if they also appear as values of the primary key of the parent

table. The referential constraint is always defined from the perspective of the dependent file. See also constraint.

register
In a database, to store information about user-defined database objects in the system catalog tables of a database.

registration
The process of registering a database extension so that its code is available to use in a particular database.

relational database
A database that can be perceived as a set of tables and manipulated in accordance with the relational model of data.

remainder page
A page that accommodates subsequent bytes of a long data row. If the trailing portion of a data row is less than a full page, it is stored on a remainder page. After the database server creates a remainder page for a long row, it can use the remaining space in the page to store other rows.

remote connection
A communications link between the local system and a system or device located on another network, or at a distant site.

remote standalone secondary server
A remote standalone secondary server participating in a high-availability configuration. RS secondary servers can be geographically distant from the primary server, serving as remote back-up servers in disaster-recovery scenarios. Each RS secondary server maintains a complete copy of the database, with updates transmitted asynchronously from the primary server over secure network connections.

remote table
In a distributed query, a table in a database of a server that is not the local database server. See also subordinate server, coordinating server.

replicate
In Enterprise Replication, pertains to participants and various attributes of how to replicate the data, such as frequency and how to handle any conflicts during replication. See also master replicate, shadow replicate.

replication server
An Informix database server that participates in Enterprise Replication.

reserved pages
The first 12 pages of the initial chunk of the root dbspace. Each reserved page stores specific control and tracking information that the database server uses.

reserved word
A word that has been set aside for special use in the SQL standard.

response file
An ASCII file that can be customized with the setup and configuration data that automates an installation. The setup and configuration data would have to be entered during an interactive install, but with a response file, the installation can proceed without any intervention.

restore point
A point in time when an instance was in a consistent state.

role A job function that identifies the tasks that a user can perform and the resources to which a user has access. A user can be assigned one or more roles.

role separation
A database server installation option that allows different users to perform different administrative tasks.

roll back
To restore data that is changed by an SQL statement to the state at its last commit point.

root dbspace
The initial dbspace that the database server creates. The root dbspace contains reserved pages and internal tables that describe and track all other dbspaces, blobspaces, sbspaces, tblspaces, chunks, and databases. See also dbspace.

root node
A single index page that contains node pointers to branch nodes. The database server allocates the root node when you create an index for an empty table. See also node.

root supertype
The named row data type at the top of a type hierarchy. A root supertype has no supertype above it.

round-robin fragmentation
A type of fragmentation that places rows one after another in fragments, rotating through the series of fragments to distribute the rows evenly.

routine
A set of statements in a program that causes the system to perform an operation or a series of related operations.

routine modifier
A keyword in the WITH clause of a CREATE FUNCTION, CREATE PROCEDURE, ALTER FUNCTION, ALTER PROCEDURE, or ALTER ROUTINE statement that specifies a particular attribute or usage of a user-defined routine.

routine overloading
The process of assigning one name to multiple routines and specify parameters of different data types on which the routines can operate. See also routine resolution, routine signature.

routine resolution
The process that the database server uses to determine which user-defined routine to execute, based on the routine signature. See also routine overloading.

routine signature
The information that the database server uses to uniquely identify a user-defined routine. See also parameter, routine overloading.

row The horizontal component of a table, consisting of a sequence of values, one for each column of the table. See also record.

row data type
A complex data type that contains one or more related data fields, of any data type except IDSSECLABEL, that form a template for a record. The

data in a row type can be stored in a row or column. See also complex data type, named row data type, unnamed row data type.

ROWID
See row identifier.

row identifier
A value that uniquely identifies a row. This value is stored with the row and does not change.

row variable
An IBM Informix ESQL/C host variable or SPL variable that holds an entire row type and provides access to the individual fields of the row.

RPO See recovery point objective.

RS secondary server
See remote standalone secondary server.

RTO See recovery time objective.

R-tree index
An indexing structure that supports spatial data. An R-tree index uses a bounding box, which is a set of coordinates that contains one or more objects and supports spatial data (two-dimensional, three-dimensional, and so on). An object can theoretically belong to more than one bounding box. An R-tree index is useful for searches on multidimensional data.

runtime environment
A set of resources that are used to run a program or process.

runtime error
An error that occurs during program execution.

S

sbspace
A logical storage area that contains one or more chunks that store only BLOB and CLOB data.

scale The number of digits in the fractional part of a number.

schema
A collection of database objects such as tables, views, indexes, or triggers that define a database. A schema provides a logical classification of database objects. See also collection.

scope of reference
The portion of a routine or application program where an identifier can be accessed. Three possible scopes exist: local (applies only in a single statement block), modular (applies throughout a single module), and global (applies throughout the entire program). See also local variable, global variable.

scroll cursor
A cursor that can fetch the next row in a sequence or any of the output rows. See also sequential cursor.

secondary access method
An access method that has routines that access an index. Secondary access methods hasten the speed of data retrieval.

secure auditing facility
A facility of Informix database servers that lets a database system security officer monitor unusual or potentially harmful user activity. Use the onaudit utility to enable auditing of events and to create audit masks. Use the onshowaudit utility to extract audit event data for analysis.

Secure Sockets Layer
A security protocol that provides communication privacy. With SSL, client/server applications can communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery. See also certificate authority.

security label
In label-based access control (LBAC), a database object that can be granted to users and can also be applied to columns and rows in a table to protect the data. Only users who are granted appropriate security labels can access data that is protected by a security label. See also label-based access control, security label component, security policy.

security label component
In label-based access control, a database object that represents one of the criteria that an organization uses to decide who has access to specific data. See also security label, element.

security policy
In label-based access control, a database object that is associated with one or more tables and that defines how LBAC can be used to protect those tables. The security policy defines what security labels can be used, how the security labels are compared to each other, and whether optional behaviors are used. See also label-based access control, security label.

select cursor
A cursor that is associated with a SELECT statement.

selective index
A type of generalized-key index that contains keys for only a subset of a table.

selectivity
The probability that any table row will satisfy a predicate.

selectivity function
A function that calculates the percentage of rows that will be returned by a filter function in the WHERE clause of a query. The optimizer uses selectivity information to determine the fastest way to execute an SQL query.

sequence
A database object that is independent of any one table that automatically generates unique key values based on initial user specifications.

sequential cursor
A cursor that can fetch only the next row in sequence and can read through a table only once each time the cursor is opened. See also scroll cursor.

server locale
The locale that a database server uses when it performs its own read and write operations. The SERVER_LOCALE environment variable can specify a nondefault locale. See also client locale.

server multiplexer group connection
A multiplexed network connection that provides reliable, secure, high-performance communication between servers in a high-availability environment.

server name
The unique name of a database server, assigned by the database server administrator, that an application uses to select a database server.

server number
A unique number between 0 and 255, inclusive, that a database server administrator assigns when a database server is initialized.

server-processing locale
The locale that a database server determines dynamically for a given connection between a client application and a database. See also locale, client/server architecture.

session
A logical or virtual connection between two stations, software programs, or devices on a network that allows the two elements to communicate and exchange data for the duration of the session. See also SQL connection, transaction.

shadow column
A hidden column on replicated tables that contains values supplied by the database server. The database server uses shadow columns to perform internal operations.

shadow replicate
A copy of an existing (primary) Enterprise Replication replicate. Shadow replicates allow Enterprise Replication to manage alter and repair operations on replicated tables. See also master replicate, replicate.

shared lock
A lock that limits concurrently running application processes to read-only operations on database data. See also exclusive lock.

shared memory
An area of memory simultaneously accessible to more than one cooperating process.

shared-object file
A library that is not linked to an application at compile time but instead is loaded into memory by the operating system as needed. Several applications can share access to the loaded shared-object file. See also dynamic link library, library.

silent installation
An installation that does not send messages to the console but instead stores messages and errors in log files. A silent installation can use response files for data input. See also response file, unattended installation.

Simple Network Management Protocol
A set of protocols for monitoring systems and devices in complex networks. Information about managed devices is defined and stored in a Management Information Base (MIB).

single-byte character
A character that uses one byte of storage. Because a single byte can store values in the range of 0 to 255, it can uniquely identify 256 characters.

With these code sets, an application can assume that one character is always stored in one byte. See also multibyte character, 8-bit character.

singleton implicit transaction
A single-statement transaction that does not require either a BEGIN WORK or a COMMIT WORK statement. This type of transaction can occur only in a database that is not ANSI compliant, but that supports transaction logging. See also explicit transaction, implicit transaction.

SLV See statement local variable.

smart large object
A large object that is stored in an sbspace, which has read, write, and seek properties (similar to a UNIX file), is recoverable, obeys transaction isolation modes, and can be retrieved in segments by an application. Smart large objects include BLOB and CLOB data types.

SMI See system-monitoring interface.

SMX connection
See server multiplexer group connection.

snapshot
An image that is an exact copy of the original files or directories from which it was created.

SNMP
See Simple Network Management Protocol.

source file
A file of programming code that is not compiled into machine language. See also data file.

source type

1. The data type from which a DISTINCT type is derived.

2. An existing type that is used to internally represent a distinct type.

SPL See Stored Procedure Language.

SPL function
An SPL routine that returns one or more values. See also stored procedure.

SPL procedure
An SPL routine that does not return a value.

SPL routine
A user-defined routine that is written in Stored Procedure Language (SPL). Its name, parameters, executable format, and other information are stored in the system catalog tables of a database. An SPL routine can be an SPL procedure or an SPL function. See also stored procedure, Stored Procedure Language.

SPL variable
A variable that is declared with the DEFINE statement in an SPL routine. See also stored procedure.

SQL See Structured Query Language.

SQLCA
See SQL communication area.

SQL communication area
A set of variables that provides an application program with information about the execution of its SQL statements, XQuery expressions, or requests from the database manager.

SQL connection
An association between an application process and a local or remote application server or database server. See also connection, session, dormant connection.

SQLDA
See SQL descriptor area.

SQL descriptor area
A set of variables in a structure used in the processing of certain SQL statements that describe input variables, output variables, or the columns of a result table. The SQLDA is intended for dynamic SQL programs. See also system-descriptor area.

SSL See Secure Sockets Layer.

stack operator
An operator that programs use to manipulate values that are on the stack.

statement
An instruction in a program or procedure.

statement block
A unit of SPL program code that performs a particular task and is usually marked by the keywords begin and end. The statement block of an SPL routine is the smallest scope of reference for program variables.

statement label
An SQL identifier that enables the GOTO label statement to transfer program control to the first executable statement that follows the declaration of the specified statement label.

statement local variable
An SPL, C, or Java function that is invoked in the WHERE clause of a query, which can declare one or more statement-local variables that are visible in other parts of the same query, including its subqueries.

static table
A nonlogging, read-only permanent table.

status variable
A program variable that indicates the status of some aspect of program execution. Status variables often store error numbers or act as flags to indicate that an error has occurred.

storage pool
A logical division of storage (directories, cooked files, and raw devices) reserved for automatically expanding existing storage spaces (dbspace, temporary dbspace, sbspace, temporary sbspace, or blobspace).

storage space
A dbspace, blobspace, or sbspace that is used to hold data.

stored procedure
A block of procedural constructs and embedded SQL statements that is stored in a database and that can be called by name. Stored procedures allow an application program to be run in two parts, one on the client and

the other on the server, so that one call can produce several accesses to the database. See also SPL routine, SPL variable, SPL function.

Stored Procedure Language
An Informix extension to SQL that provides flow-control features such as sequencing, branching, and looping. See also SPL routine.

string In programming languages, the form of data used for storing and manipulating text.

Structured Query Language
A standardized language for defining and manipulating data in a relational database. See also Data Manipulation Language.

subordinate server
Any database server in a distributed query that did not initiate the query. See also coordinating server, remote table.

subquery
In SQL, a subselect used within a predicate, for example, a select-statement within the WHERE or HAVING clause of another SQL statement.

subscript
An integer or variable whose value selects a particular element in a table or array.

substring
A part of a character string.

subtable
A typed table that inherits properties (column definitions, constraints, triggers) from a supertable above it in the table hierarchy.

subtree
A tree structure created by arbitrarily denoting a node to be the root node in a tree. A subtree is always part of a whole tree. See also forest of trees index.

subtype
A named row data type that inherits all representation (data fields) and behavior (routines) from a supertype above it in the type hierarchy and can add additional fields and routines. The number of fields in a subtype is always greater than or equal to the number of fields in its supertype.

supertable
A typed table that passes properties (constraints, storage options, triggers) to a subtable beneath it in the table hierarchy.

supertype
A named row data type whose representation (data fields) and behavior (routines) is inherited by a subtype below it in the type hierarchy.

symmetric multiprocessing system
A system composed of multiple computers that are connected to a single high-speed communication subsystem. See also node, parallel processing platform.

synonym
A name that is assigned to a table, view, or sequence and that can be used in place of the original name. A synonym does not replace the original name; instead, it acts as an alias for the table, view, or sequence.

sysadmin database
A server-level database that supports all the logged databases of a database server instance. Its tables store information that the DBSA can use in various administrative operations, including tuning and configuring the database server, monitoring and analyzing resource usage, scheduling recurring maintenance tasks, and logging calls to the SQL Administration API functions.

syscall
See system call.

system call
A call by a program to an operating system subroutine.

system catalog
See data dictionary.

system-defined cast
A pre-defined cast that is known to the database server. Each built-in cast performs automatic conversion between two different built-in data types.

system-descriptor area
A dynamic SQL management structure that is used to store information about database columns or host variables used in dynamic SQL statements. The structure contains an item descriptor for each column; each item descriptor provides information such as the name, data type, length, scale, and precision of the column. See also SQL descriptor area, descriptor.

system index
An index that the database server creates to implement a unique constraint or a referential constraint. A system index is distinct from a user index, which a user creates explicitly. See also attached index, user index.

system-monitoring interface
A collection of tables in the sysmaster database that maintains dynamically updated information about the operation of the database server. The tables are constructed in memory but are not recorded on disk. Users can query the SMI tables with the SELECT statement of SQL.

T

table fragment
Zero or more rows that are grouped and stored in a dbspace that are specified when you create the fragment. A virtual table fragment might reside in an sbspace or an extspace.

table fragmentation
A method of separating a table into potentially balanced fragments to distribute the workload and optimize the efficiency of the database operations. Also known as data partitioning. Table-fragmentation methods (also known as distribution schemes) include expression-based, hybrid, range, round-robin, and system-defined hash.

table hierarchy
A structure representing the relationship between typed tables in which subtables inherit the behavior (constraints, triggers, storage options) from supertables. Subtables can have additional constraint definitions, storage options, and triggers.

table space
A logical collection of extents that are assigned to a table. A table space

contains all the disk space that is allocated to a given table or table fragment and includes pages allocated to data and to indexes, pages that store TEXT or BYTE data in the dbspace, and bitmap pages that track page use within the extents.

target table
The underlying base table that a violations table and diagnostics table are associated with.

TB See terabyte.

tblspace
The part of the table that is located in a single data file. Each tblspace can hold one table fragment.

template
In Informix, a mechanism that is used to set up and deploy Enterprise Replication for a group of tables on one or more servers.

terabyte
For processor storage, real and virtual storage, and channel volume, 2 to the 40th power or 1 099 511 627 776 bytes. For disk storage capacity and communications volume, 1 000 000 000 000 bytes.

text data type
A data type for a simple large object that stores text and can be as large as 2^31 bytes.

thread A stream of computer instructions that is in control of a process. In some operating systems, a thread is the smallest unit of operation in a process. Several threads can run concurrently, performing different jobs. See also user thread.

trace A record of the processing of a computer program or transaction. The information collected from a trace can be used to assess problems and performance.

transaction
An atomic series of SQL statements that make up a logical unit of work. All of the data modifications made during a transaction are either committed together as a unit or rolled back as a unit.

transaction mode
The method by which constraints are checked during transactions.

transient type
An unnamed complex data type that is used to hold query results or is part of a temporary table.

trigger
A database object that is associated with a single base table or view and that defines a rule. The rule consists of a set of SQL statements that runs when an insert, update, or delete database operation occurs on the associated base table or view.

trusted context
A database security object that enables the establishment of a trusted relationship between a database management system and an external entity.

trusted connection
A database connection whose attributes match the attributes of a unique trusted context defined at the database server.

tuple See row.

two-phase commit
A two-step process by which recoverable resources and an external subsystem are committed. During the first step, the database manager subsystems are polled to ensure that they are ready to commit. If all subsystems respond positively, the database manager instructs them to commit.

type constructor
An SQL keyword that indicates to the database server the type of complex data to create (for example, LIST, MULTISET, ROW, SET).

typed collection variable
An ESQL/C collection variable or SPL variable that has a defined collection data type associated with it and can only hold a collection of its defined type. See also untyped collection variable.

typed table
A table in which the data type of each column is defined separately or the types for the columns are based on the attributes of a user-defined structured type.

type hierarchy
A relationship that you define among named row data types in which subtypes inherit representation (data fields) and behavior (routines) from supertypes and can add more fields and routines.

type substitutability
The ability to use an instance of a subtype when an instance of its supertype is expected.

U

UDA See user-defined aggregate.

UDC See user-defined cast.

UDF See user-defined function.

UDR See user-defined routine.

UDS See user-defined statistics.

unattended installation
An installation that does not require any user interaction. See also silent installation.

unbuffered disk I/O
Disk I/O that is controlled directly by the database server instead of the operating system. This direct control helps improve performance and reliability for updates to data. Unbuffered I/O is supported by character-special files on UNIX and by both unbuffered files and the raw disk interface on Windows. See also buffered disk I/O.

uncommitted read
An isolation level under which a query in a transaction can read any rows, even if there are uncommitted changes to those rows made by statements in other transactions. The UR isolation level is upgraded to CS when data is being modified. See also cursor stability, isolation level, committed read.

unique constraint
In Informix, a descriptor that is assigned to each column in a table, function argument, or function return type that indicates the type of data that can be held.

unique index
An index that ensures that no identical key values are stored in a column or a set of columns in a table.

unique key
A key that is constrained so that no two of its values are equal. See also constraint, primary key.

unlock
To release an object or system resource that was previously locked and return it to general availability.

unnamed row data type
A row data type created with the ROW constructor that has no defined name and no inheritance properties. Two unnamed row data types are equivalent if they have the same number of fields and if corresponding fields have the same data type, even if the fields have different names. See also row data type.

untyped collection variable
A generic ESQL/C collection variable or SPL variable that can hold a collection of any collection data type and takes on the data type of the last collection assigned to it. See also typed collection variable.

update lock
In Informix, a promotable lock that is acquired during a SELECT...FOR UPDATE. An update lock behaves like a shared lock until the update actually occurs, and it then becomes an exclusive lock. See also promotable lock.

upgrade
To install a new version or release of a product to replace an earlier version or release of the same product.

UR See uncommitted read.

user-defined aggregate
An aggregate function that is not provided by the database server (is not built in) that includes extensions to built-in aggregates and newly defined aggregates. The database server manages all aggregates.

user-defined cast
A cast that a user creates with the CREATE CAST statement. A user-defined cast typically requires a cast function. A user-defined cast can be an explicit cast or an implicit cast. See also cast.

user-defined data type
A data type that is defined for use in a relational database. One can define opaque data types and distinct data types. See also data type.

user-defined function
A user-defined routine that returns at least one value. You can write a user-defined function in SPL (SPL function) or in an external language that the database server supports (external function).

user-defined procedure
A user-defined routine that does not return a value. You can write a

user-defined procedure in SPL (SPL procedure) or in an external language that the database server supports (external procedure). See also procedure.

user-defined routine
A routine that users write and register in the system catalog tables of a database, and that an SQL statement or another routine can invoke. A user-defined routine is written in SPL or in an external language (external routine) that the database server supports.

user-defined statistics
Information about the opaque data type values in a database that is collected by the UPDATE STATISTICS statement, which calls user-defined functions to calculate the statistics. The optimizer uses these statistics to determine the fastest way to execute an SQL query.

user index
An index that a user creates explicitly with the CREATE INDEX statement. See also attached index, system index.

user mapping tables
System catalog tables that map users to operating system properties that enable access to Informix and control the level of privileges.

user thread
The primary thread that the database server runs to service client applications. A user thread also includes a thread to service requests from the onmode utility, threads for recovery, and page-cleaner threads. See also thread.

V

value object
A self-contained binary object that provides standard interfaces to its callers. Value objects can be used in client applications.

variable
A representation of a changeable value. See also global variable.

variant function
A user-defined function that might return different values when passed the same arguments. A variant function can contain SQL statements. See also nonvariant function.

variant routine
A routine that can return different values when it is invoked with the same arguments.

view A dynamically controlled subset of the columns of one or more database tables. A view can give the programmer control over what information the user sees and manipulates and represents a virtual table that holds the results of a specified SELECT statement.

view folding
A query plan strategy for joined queries in which one or more of the data sources is a view, by which the query optimizer incorporates the view definitions into the main query. For queries in which this technique is possible, view folding can significantly improve performance, as compared to materializing the view as a temporary table.

violations table
A table that holds rows that fail to satisfy the constraints and unique index requirements during data manipulation operations on base tables. See also diagnostics table.

virtual column
A column of information derived from an SQL statement that is not stored in the database.

virtual-column index
A type of generalized-key index that contains keys that are the result of an expression.

virtual processor
A multithreaded process that makes up the database server and is similar to the hardware processors in the computer. It can serve multiple clients and, where necessary, run multiple threads to work in parallel for a single query. See also asynchronous disk I/O virtual processor.

virtual table
A table created to access data in an external file, external DBMS, smart large object, or in the result set of an iterator function in a query. The database server does not manage external data or directly manipulate data within a smart large object. The Virtual-Table Interface allows users to access the external data in a virtual table using SQL DML statements and join the external data with Informix table data.

W

white space
A sequence of one or more characters, such as the blank character, the newline character, or the tab character, that belong to the space character class.

wide character
A character whose range of values can represent distinct codes for all members of the largest extended character set specified among the supporting locales.

Appendix. Accessibility

IBM strives to provide products with usable access for everyone, regardless of age or ability.

Accessibility features for IBM Informix products

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in IBM Informix products. These features support:

- Keyboard-only operation.
- Interfaces that are commonly used by screen readers.
- The attachment of alternative input and output devices.

Tip: The information center and its related publications are accessibility-enabled for the IBM Home Page Reader. You can operate all features by using the keyboard instead of the mouse.

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

Related accessibility information

IBM is committed to making our documentation accessible to persons with disabilities. Our publications are available in HTML format so that they can be accessed with assistive technology such as screen reader software.

You can view the publications in Adobe Portable Document Format (PDF) by using the Adobe Acrobat Reader.

IBM and accessibility

See the *IBM Accessibility Center* at http://www.ibm.com/able for more information about the IBM commitment to accessibility.

Dotted decimal syntax diagrams

The syntax diagrams in our publications are available in dotted decimal format, which is an accessible format that is available only if you are using a screen reader.

In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), the elements can appear on the same line, because they can be considered as a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that your screen reader is set to read punctuation. All syntax elements that have the same dotted decimal number (for example, all syntax elements that have the number 3.1) are mutually exclusive

alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, the word or symbol is preceded by the backslash (\) character. The * symbol can be used next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is read as 3 * FILE. Format 3* FILE indicates that syntax element FILE repeats. Format 3* * FILE indicates that syntax element * FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol that provides information about the syntax elements. For example, the lines 5.1*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, that element is defined elsewhere. The string following the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 means that you should refer to a separate syntax fragment OP1.

The following words and symbols are used next to the dotted decimal numbers:

? Specifies an optional syntax element. A dotted decimal number followed by the ? symbol indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element (for example, 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that syntax elements NOTIFY and UPDATE are optional; that is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

! Specifies a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicates that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the same dotted decimal number can specify a ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the default option for the FILE keyword. In this example, if you include the FILE keyword but do not specify an option, default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In

this example, if the FILE keyword is omitted, default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP only applies to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.

* Specifies a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be repeated. For example, if you hear the line 5.1* data-area, you know that you can include more than one data area or you can include none. If you hear the lines 3*, 3 HOST, and 3 STATE, you know that you can include HOST, STATE, both together, or nothing.

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.
3. The * symbol is equivalent to a loop-back line in a railroad syntax diagram.

+ Specifies a syntax element that must be included one or more times. A dotted decimal number followed by the + symbol indicates that this syntax element must be included one or more times. For example, if you hear the line 6.1+ data-area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. As for the * symbol, you can only repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the * symbol, is equivalent to a loop-back line in a railroad syntax diagram.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy,

modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Printed in USA

SC27-3531-00