SYBASE®

**Sybase® Adaptive Server™ Enterprise
Glossary**

Adaptive
Server™

Adaptive Server Glossary

A

access

The use of a select, insert, update, or delete command on a table or view.

access method

The method used to find the data rows needed to satisfy a query. Access methods can be serial or parallel. Serial access methods include: **table scan**, **nonclustered index** access, **clustered index** access. A parallel access method is either a **partition-based access method** or a **hash-based access method**. Partition-based access methods include **parallel partition scan** and **parallel clustered index partition scan**. Hash-based access methods include **parallel hash-based table scan** and **parallel nonclustered index hash-based scan**.

Adaptive Server

The server in the Sybase Client/Server architecture (called SQL Server prior to the Adaptive Server 11.5 release). Adaptive Server manages multiple databases and multiple users, keeps track of the actual location of data on disks, maintains mapping of logical data description to physical data storage, and maintains data and procedure caches in memory. See also **segment**.

Adaptive Server engine

See **engine**.

address lock

A type of lock applied to certain resources such as network buffers or internal structures. Address locks are also used to lock index pages.

aggregate function

A function that generates summary values that appear as new columns in the query results. The aggregate functions available in Transact-SQL are: average (avg), maximum (max), minimum (min), sum (sum), and count of the number of items (count).

alias

A pseudonym that allows an Adaptive Server user to be known in a database as another user.

allocation page

The first page of an allocation unit, which tracks the use of all pages in the allocation unit.

allocation unit

A logical unit of 1/2 MB, or 256 2K pages. The disk init command initializes a new database file for Adaptive Server and divides it into allocation units.

Application Program Interface (API)

A subroutine that allows client applications to interface with SQL Server. Also known as a **library**.

application-building tool

See **development tool**.

argument

A value supplied to a function or procedure that is required to evaluate the function.

arithmetic expression

An expression that contains only numeric operands and returns a single numeric value. In Transact-SQL, the operands can be of any Adaptive Server numeric datatype. They can be functions, variables, parameters, or they can be other arithmetic expressions. Also called a **numeric expression**.

arithmetic operators

Symbols that enable you to create **arithmetic expression**s in SQL statements. Addition (+), subtraction (-), division (/), and multiplication (*) can be used with numeric columns. Modulo (%) can be used with *int*, *smallint*, and *tinyint* columns only. See also **comparison operator**.

audit trail

Audit records stored in the *sybsecurity* database.

auditing

The act of recording security-related system activity that can be used to detect penetration of the system and misuse of system resources.

automatic recovery

A process that runs every time Adaptive Server is restarted. The process ensures that all transactions that completed before Adaptive Server stopped are brought forward and all incomplete transactions are rolled back.

B

B-tree

Short for balanced tree, or binary tree. Adaptive Server uses B-tree indexing. All leaf pages in a B-tree are the same distance from the root page of the index. B-trees provide consistent and predictable performance, good sequential and random record retrieval, and a flat tree structure.

backup

A copy of a database or transaction log, used to recover from a media failure.

base date

January 1, 1900; the date supplied by Adaptive Server when a user does not specify a value for a date column.

base tables

The permanent tables on which a view is based. Also called underlying tables.

batch

One or more Transact-SQL statements terminated by an end-of-batch signal, which submits them to Adaptive Server for processing.

Boolean expression

An expression that evaluates to TRUE (1), or FALSE (0). Boolean expressions are often used in control of flow statements, such as **if** or **while** conditions.

buffer

A unit of storage in a memory pool. A single **buffer cache** can have pools configured for different I/O sizes, or buffer sizes. All buffers in a pool are the same size. If a pool is configured for 16K I/O, all buffers are 16K, holding eight data pages. Buffers are treated as a unit; all data pages in a buffer are read, written, or flushed from cache simultaneously.

buffer cache

An area of memory within Adaptive Server that contains the in-memory images of database pages, as well as the data structures required to manage the pages. Each cache is given a unique name that is used for configuration purposes. By default, Adaptive Server has a single cache named "default data cache." Caches configured by users are called "user defined caches." Buffer caches are also referred to as data caches and named caches.

buffer pool

An area of memory within a **buffer cache** that contains a set of buffers linked together on an MRU/LRU (most recently used/least recently used) list.

buffer reuse strategy

Reading pages into the data cache at the LRU end of the cache chain, so that the same buffer is available for reuse immediately. This strategy keeps select commands that require large numbers of page reads from flushing other data from the cache.

built-in functions

A wide variety of functions that take one or more **arguments** and return results. The built-in functions include **aggregate functions**, **mathematical functions**, **system functions**, **string functions**, **text and image functions**, **date functions**, and **datatype conversion functions**.

bulk copy

The process of copying data in and out of databases. In Adaptive Server this operation is performed with the `bcp` utility.

business rule

A rule that restricts input and updates based on real-world requirements. For example, a *sales* database can be required to forbid the sale of any items for which there is insufficient stock. Adaptive Server can check each row before it is added to the *sales_orders* table and enforce the requirement using the business rule on the *quantity* column.

C

cache

See **buffer cache**.

cache hit ratio

For many processes, Adaptive Server uses an in-memory cache. The cache hit ratio is the percentage of times a needed page or result was found in the cache. For data pages, the cache hit ratio is the percentage of page requests that are serviced by the data cache compared to requests that require disk I/O.

candidate key

A **primary key** or **unique constraint** column. A table can have multiple candidate keys.

Cartesian product

All the possible combinations of the rows from each of the tables specified in a join. The number of rows in the Cartesian product is equal to the number of rows in the first table times the number of rows in the second table. Once the Cartesian product is formed, the rows that do not satisfy the join conditions are eliminated.

cascading delete

A delete operation that affects related data in other tables.

catalog stored procedure

A type of **system procedure** that returns data from the **system table**s in tabular format.

chained transaction mode

Determines whether or not Adaptive Server automatically starts a new transaction on the next data retrieval or data modification statement. When chained transaction mode is set on outside a transaction, the next data retrieval or data modification statement begins a new transaction. This mode is ANSI compliant. It ensures that every SQL data retrieval and data modification statement occurs inside a transaction. Chained transaction mode may be incompatible with existing Transact-SQL programs. Chained transaction mode is off by default. Applications that require ANSI SQL (such as the Embedded SQL precompiler) should automatically set the chained option on at the beginning of each session.

character expression

An expression that returns a single character-type value. It can include literals, concatenation operators, functions, and column identifiers.

character set

A set of specific (usually standardized) characters with an encoding scheme that uniquely defines each character. ASCII and ISO 8859-1 (Latin 1) are two common character sets.

character set conversion

Changing the encoding scheme of a set of characters on the way into or out of Adaptive Server. Conversion is used when Adaptive Server and a client communicating with it use different character sets. For example, if Adaptive Server uses ISO 8859-1 and a client uses Code Page 850, character set conversion must be turned on so that both server and client interpret the data passing back and forth in the same way.

check constraint

A constraint placed on the `check` command that limits the values users can insert into a column of a table. A `check` constraint specifies the *search_condition* that values must pass before being inserted into the table.

checkpoint

The point at which all data pages that have been changed are guaranteed to have been written to the database device.

CIS

See **Component Integration Services (CIS)**.

classification

A hierarchical level of security; for example, Top Secret, which has a higher classification value than Secret.

clause

A set of **keywords** and options that tailor a Transact-SQL command to meet a particular need. Also called a keyword phrase.

client

The user's side of a client/server arrangement; can refer to the software making the calls to the server or to the machine running the client software.

client cursor

A cursor declared through Open Client calls or Embedded SQL. Open Client keeps track of the rows returned from Adaptive Server and buffers them for the application. Updates and deletes to the result set of client cursors can be done only through Open Client calls.

client/server architecture

A computer system architecture in which clients request a service and a server provides that service. Each machine can then specialize in the tasks it is best suited for.

client task

A thread spawned to service a client request.

clustered index

An index in which the physical order and the logical (indexed) order is the same. The leaf level of a clustered index represents the data pages themselves. A table can have only one clustered index.

code set

See **character set**.

collating sequence

See **sort order**.

column

A data value that describes one characteristic of an **entity**. A column contains an individual data item within a row or record. Also called a **field**.

column-level constraint

Limit the values of a specified column. Place column-level constraints after the column name and datatype in the create table statement, before the delimiting comma.

command

An instruction that specifies an operation to be performed by the computer. Each command or SQL statement begins with a keyword, such as insert, that names the basic operation performed. Many SQL commands have one or more **keyword phrases**, or **clauses**, that tailor the command to meet a particular need.

command permission

A **permissions** that applies to commands. See also **object permission**.

command terminator

The end-of-batch signal that sends the batch to Adaptive Server for processing.

comparison operator

A symbol used to compare one value to another in a query. Comparison operators include equal to (=) greater than (>), less than (<), greater than or equal to (>=), less than or equal to (<=), not equal to (!=), not greater than (!>), and not less than (!<). See also **arithmetic operators**.

compartment

One of a set of non-hierarchical values used with classifications to make up a sensitivity label. Compartments usually represent topics or work groups.

compatible datatypes

Datatypes that are automatically converted for implicit or explicit comparison.

compiled object

Any object that requires entries in the *sysprocedures* table, including check constraints, defaults, rules, stored procedures, triggers, and views. These objects are described by **source text**. Adaptive Server uses compiled objects to contain vital information about each database and to help you access and manipulate data.

Component Integration Services (CIS)

Component Integration Services is a feature that extends Adaptive Server capabilities and provides enhanced interoperability. It is the core interoperability feature of OmniConnect. Component Integration Services allows Adaptive Server and OmniConnect to present a uniform view of enterprise data to client applications and provides location transparency to enterprise-wide data sources.

composite indexes

Indexes that involve more than one column. Use composite indexes when two or more columns are best searched as a unit because of their logical relationship.

composite key

An index key that includes two or more columns; for example, *authors(au_lname, au_fname)*.

concatenation

Combining expressions to form longer expressions. The expressions can include any combination of binary or character strings or column names.

concurrency

Concurrent execution (multiprocessing) of independent and possibly competing processes or transactions.

consumer process

In parallel sorts, consumer processes simultaneously sort a discrete range data received from producer processes. Consumer processes hand the sorted data off to a **coordinating process**

constant expression

An expression that returns the same value each time the expression is used. In Transact-SQL syntax statements, a constant expression does not include variables or column identifiers.

constraint

A rule applied to a database object that ensures that all entries in the database object to which it applies satisfy a particular condition. For example, a column may have a constraint requiring that all values in the column be unique.

context-sensitive protection

Protection that provides certain permissions or privileges depending on the identity of the user. This type of protection can be provided using views and the user_id built-in function.

control page

A reserved database page that stores information about the last page of a **partition**.

control-break report

A report or data display that breaks data into groups and generates summary information for each break. The breaks control the generation of summary data.

control-of-flow language

Transact-SQL's programming-like constructs (such as if, else, while, and goto) that control the flow of execution of Transact-SQL statements.

conversion

See **character set conversion**.

coordinating process

In parallel sorts, the coordinating process merges the results of the consumer processes into a single result set. For queries, the result set is the final, sorted data. For a create index statement, the coordinating process merges the sub-indexes into one, final index.

correlated subquery

A **subquery** that cannot be evaluated independently, but depends on the outer query for its results. Also called a **repeating subquery** because the subquery is executed once for each row that might be selected by the outer query. See also **nested queries**.

correlation names

Distinguish the different roles a particular table plays in a query, especially a correlated query or **self-join**. Assign correlation names in the from clause and specify the correlation name after the table name:

```
select au1.au_fname, au2.au_fname
from authors au1, authors au2
where au1.zip = au2.zip
```

counter

A measurable performance item that can be reported by the Windows NT Performance Monitor. This is generic Windows NT terminology for describing a mechanism for producing statistical performance information. Adaptive Server maintains a special set of counters to measure and report on Adaptive Server objects or events on the Windows NT platform.

covered query

See **index covering**.

covering

See **index covering**.

cursor

A named select statement that retrieves one or more rows from a given table, and allows you to modify or delete the rows individually. Cursors consist of two parts: the **cursor result set** and the **cursor position**.

cursor position

Indicates the current row of the cursor. You can explicitly reference that row using statements designed to support cursors, such as delete and update. Change the current cursor position through fetch, which moves the current cursor position one or more rows down the **cursor result set**.

cursor result set

The set of rows resulting from the execution of the select statement associated with the cursor.

cursor scan

The process of generating a **cursor result set**.

cursor scope

The context in which the **cursor** is used. A cursor's existence depends on its scope: within a particular user session, within a stored procedure, or within a trigger.

cursor stability

A **locking level** or **isolation level** in which Adaptive Server has a shared **lock** on the base table pages that contain a current cursor row. The page remains locked until the cursor is no longer positioned on the page (as a result of fetches). If the base table has an index, the corresponding index pages also have shared locks.

D

data cache

See **buffer cache**

data definition

The process of setting up databases and creating database objects such as tables, indexes, rules, defaults, procedures, triggers, and views. See also **source text**.

data dictionary

The system tables that contain a description of each **database object** and how it is structured.

data integrity

The correctness and completeness of data within a database.

data modification

Adding, deleting, or changing information in the database with the insert, delete, and update commands.

data retrieval

The act of requesting data from the database and receiving the results. See also **query**.

data warehouse

A subject-oriented information store specifically designed for decision support and analytical processing. A data warehouse contains large amounts of information used by managers, strategic marketers, and merchandisers to help them understand and predict market trends and make crucial business decisions. The information in a data warehouse is not usually modified by the people who use it. See also **database**.

database

See **relational database**.

database administration

The tasks involved in maintaining, designing, implementing changes to, tuning, and expanding a database. See also **system administration**.

database device

A **device** dedicated to the storage of the objects that make up databases.

database file

See **table**.

database integrity

A characteristic of a database evidenced by the database being both valid and complete. Database integrity consists of two complementary components: **validity**, which guarantees that all false information is excluded from the database, and **completeness**, which guarantees that all true information is included in the database. See also **integrity constraints**.

database object

One of the components of a database: table, column, view, index, procedure, trigger, default, or rule. See also **object**.

Database Object Owner

A user who creates a **database object**.

Database Owner

The creator of a database. The Database Owner has control over all the database objects in that database. The login name for the Database Owner is "dbo."

datatype

Specifies what kind of information each column will hold and how the data will be stored. Datatypes include *char*, *int*, *money*, and so on. Users can construct their own datatypes based on the Adaptive Server system datatypes.

datatype conversion function

A function that is used to convert expressions of one datatype into another datatype, whenever these conversions are not performed automatically by Adaptive Server.

datatype hierarchy

The hierarchy that determines the results of computations using values of different datatypes.

date function

> A function that displays information about dates and times, or manipulates date or time values. The date functions include getdate, datename, datepart, datediff, and dateadd.

date part

> Parts of a date, such as day, month, or year, recognized by the Transact-SQL date functions.

DBMS

> Database Management System. The Adaptive Server DBMS component performs query processing and transaction management for the current task. For example, the DBMS parses, compiles, and executes SQL statements and returns any results.

"dbo" account

> See **Database Owner**.

deadlock

> A deadlock occurs when two or more user processes each have a lock on a separate page or table and each wants to acquire a lock on the other process's page or table. The transaction with the least accumulated CPU time is killed and all of its work is rolled back.

Decision Support System (DSS)

> A database system that supports queries involving large amounts of data. Commonly used to make business decisions. DSS queries typically access entire tables or large portions of tables, involve joins between many tables, and return summaries of large result sets. DSS applications are often run on a scheduled basis; for example, to produce the same report at the close of every business day. See also **data warehouse**.

default

> The option chosen by the system when no other option is specified.

default clause

> Specifies the default value for a column in the create table statement.

default database

> The database that users connect to when they log in.

default language

For a user, the language that displays that user's prompts and messages, set with **sp_modifylogin** or the **language** option of the set command. For Adaptive Server, the language used to display prompts and messages for all users unless a user chooses a different language.

deferred update

An update operation that takes place in two steps. First, the log records for deleting existing entries and inserting new entries are written to the log, but only the delete changes to the data pages and indexes take place. In the second step, the log pages are rescanned, and the insert operations are performed on the data pages and indexes. See also **direct update**.

definition statement

See **source text**.

degree of parallelism

The number of worker processes that the optimizer chooses to execute the query in parallel. The degree of parallelism depends on both the upper limit of parallelism for the query and the level of parallelism suggested by the optimizer.

delete/insert direct update

A type of **direct update** operation. The row is deleted from its original location, and inserted at a new location.

delimited identifiers

Object names enclosed in double quotes that avoid certain restrictions on object names.

demand lock

A demand lock prevents any more shared locks from being set on a table or data page. Any new shared lock request has to wait for the demand lock request to finish.

density

The average fraction of all the rows in an index that have the same key value. Density is 1 if all of the data values are the same and $1/N$ if every data value is unique.

dependent

Data is logically dependent on other data when master data in one table must be kept synchronized with detail data in another table in order to protect the logical consistency of the database.

detail

Data that logically depends on data in another table. For example, in the *pubs2* database, the *salesdetail* table is a detail table. Each order in the *sales* table can have many corresponding entries in *salesdetail*. Each item in *salesdetail* is meaningless without a corresponding entry in the *sales* table.

development tool

Software such as PowerBuilder that helps you build specialized GUI applications for accessing SQL Server databases.

device

Any piece of disk (such as a **partition**) or a file in the file system used to store databases and their objects. See also **database device**.

device I/O

The action of reading to or writing from a **database device**.

direct update

An update operation that takes place in a single step, that is, the log records are written and the data and index pages are changed. Direct updates can be performed in three ways: **in-place update**, **on-page update**, and **delete/insert direct update**. See also **deferred update**.

dirty read

Occurs when one transaction modifies a row, and then a second transaction reads that row before the first transaction commits the change. If the first transaction rolls back the change, the information read by the second transaction becomes invalid.

discretionary access controls (DAC)

Restrict your access to objects based on your identity and/or your group membership. The controls are discretionary in the sense that a user with a certain access permission is capable of passing that permission onto any other user (such as with the grant command). See also **permissions**.

disk allocation pieces

Disk allocation pieces are the groups of allocation units from which Adaptive Server constructs a new database file. The minimum size for a disk allocation piece is one **allocation unit**.

disk initialization

The process of preparing a **database device** or file for Adaptive Server use. After the device is initialized, it can be used for storing databases and database objects. The command used to initialize a database device is disk init.

disk mirror

A duplicate of a Adaptive Server database device. All writes to the device being mirrored are copied to a separate physical device, making the second device an exact copy of the device being mirrored. If one of the devices fails, the other contains an up-to-date copy of all transactions. The command disk mirror starts the disk mirroring process.

display precision

The number of significant binary digits offered by the default display format for *real* and *float* values. Internally, *real* and *float* values are stored with a precision less than or equal to that of the platform-specific datatypes on which they are built. For display purposes, Sybase *real* values have 9 digits of precision; Sybase *float* values, 17.

DLL

Dynamic Link Library. Software used by Microsoft Windows and IBM OS/2 to provide services to applications.

driver

A Sybase library that provides an interface to an external service provider.

dump

The action of making a backup of an entire database, including the data and the **transaction log**, which you accomplish with the dump database command. Also, the data that results from this action. See also **load**.

dump striping

Interleaving of dump data across several dump volumes.

dump volume

A single tape, partition, or file used for a database or transaction dump. A dump can span many volumes, or many dumps can be made to a single tape volume.

dynamic dump

A dump made while the database is active.

dynamic index

A worktable built by Adaptive Server for the resolution of queries using or. As each qualifying row is retrieved, its row ID is stored in the worktable. The worktable is sorted to remove duplicates, and the row IDs are joined back to the table to return the values.

E

engine

An instance of the Adaptive Server executable that can communicate with other Adaptive Server engines in shared memory. An Adaptive Server running on a uniprocessor machine always has one engine, engine 0. An Adaptive Server running on a multiprocessor machine can have one or more engines. Also called **server engine**.

enterprise data

Data that exists anywhere in a networked system. Enterprise data can be stored on Sybase servers or as heterogeneous data.

entity

A database or a database object that can be identified by a unique ID and that is backed by database pages. Examples of entities are the database *pubs2*, the log for database *pubs2*, the clustered index for table *titles* in database *pubs2*, and the table *authors* in database *pubs2*. Identifying entities is the first step in the **logical design** of a database.

equijoin

A **join** based on equality.

error handling

Techniques available to Transact-SQL programmers on which to base code and display errors and error messages.

error log

A file that stores severe **error messages** and the results of the startup and recovery of databases.

error message

>A message issued by Adaptive Server, usually to the user's terminal, when Adaptive Server detects an error condition.

error state number

>The number attached to a Adaptive Server error message that allows unique identification of the line of Adaptive Server code at which the error was raised.

evaluated configuration

>The configuration of SQL Server that was evaluated at the C2 security level by the NSA (National Security Agency) in 1996 on the HP 9000 HP-UX BLS, 9.09+ platform. Certain features of SQL Server, such as remote procedures and direct updates to system tables, were excluded from the evaluated configuration. For a complete list of features excluded from the evaluated configuration, see Appendix A in the *SQL Server Installation and Configuration Guide for HP 9000 HP-UX BLS, 9.09+*.

exclusive lock

>A type of lock acquired on an object during a write operation. It does not allow other transactions to acquire exclusive, updated, or shared locks on the object. Exclusive locks can be obtained on a table or page.

execute cursor

>A cursor that is a subset of client cursors whose result set is defined by a stored procedure that has a single select statement. The stored procedure can use parameters. The values of the parameters are sent through Open Client calls.

existence join

>A type of join performed in place of certain subqueries. Instead of the usual nested iteration through a table that returns all matching values, an existence join returns TRUE when it finds the first value and stops processing. If no matching value is found, it returns FALSE.

expression

>A computation, column data, a built-in function, or a subquery that returns values.

Extended Stored Procedure (ESP)

>A mechanism for calling external procedural language functions from within Adaptive Server. Users invoke an extended stored procedure using the same syntax as they use for a regular stored procedure, but instead of executing Transact-SQL statements, the ESP executes procedural language code, which is compiled in a dynamic link library (DLL) or shared library.

extent

Whenever a table or index requires space, Adaptive Server allocates a block of eight 2K pages, called an extent, to the object.

external login

An alternate login name and password used when a Component Integration Services client logs into a remote server. It is created using sp_addexternlogin.

F

fatal errors

Errors with severity levels of 19 and above. They terminate the user's work session, so that it is necessary to log in again.

fetch

A fetch moves the current cursor position down the cursor result set. Also called a cursor fetch.

field

See **column**.

FIPS flagger

An Adaptive Server option activated with the set command. When the Federal Information Processing Standards (FIPS) flagger is active, Adaptive Server returns a warning message when you use Transact-SQL extensions to entry level SQL92. FIPS recognizes SQL89 as the base standard.

foreign key

A key column in a table that logically depends on a **primary key** column in another table. Also, a column (or combination of columns) whose values are required to match a primary key in some other table.

format file

A file created while using bcp to copy data out from a table in a Adaptive Server database to an operating system file. The format file contains information on how the data being copied out is formatted and can be used to copy the data back into a Adaptive Server table or to perform additional copy outs.

fragment

When you allocate only a portion of the space on a device with create or alter database, that portion is called a fragment.

freelock list

On a multi-engine Adaptive Server, each engine has its own freelock list (a list of locks available to satisfy lock requests). If the engine freelock list runs out of locks, Adaptive Server moves locks from the global freelock list to the engine freelock list.

free-space threshold

A user-specified threshold that specifies the amount of space on a segment, and the action to be taken when the amount of space available on that segment is less than the specified space.

functions

See **built-in functions**.

G

gateway

Intermediate software that provides language translation, datatype conversion, and protocol conversion between the client and the data source.

Generic Access Module

A customer-built Open Server application used as a gateway. The application must conform to the interface described in the *OmniSQL Server Generic Access Module Reference Manual.* Customers are no longer encouraged to build applications to the generic access specification. The *sds* specification is preferred.

global variable

A system-defined **variable** that Adaptive Server updates on an ongoing basis. For example, *@@error* contains the last error number generated by the system. See also **local variable**.

group

A uniquely named set of users assigned a set of permissions for the objects and operations within a database.

grouped aggregate

See **vector aggregate**.

guest

A user name in the *sysusers* table of the *model* database, which enables a user with a valid Adaptive Server login to use databases created from *model,* with limited privileges.

H

Halloween problem

An anomaly associated with cursor updates, whereby a row seems to appear twice in the result set. This happens when the index key is updated by the client and the updated index row moves farther down in the result set.

hard fault

A persistent corruption of Adaptive Server found during a `dbcc checkstorage` operation. Hard faults cannot be correct by restarting Adaptive Server. See also **soft fault**.

hash-based access method

A category of parallel access method in which multiple worker processes work on a single chain of index or data pages. This category includes **parallel hash-based table scan** and **parallel nonclustered index hash-based scan**.

heap table

A table without a **clustered index**, where data is stored as a heap. The data rows are not stored in any particular order, and new data is always inserted on the last page of a page chain. A typical heap table is an unpartitioned table without a clustered index, but a partitioned table without a clustered index is also classified as a heap table.

heterogeneous data

Any data from non-Sybase data sources such as Oracle, Informix, or DB2.

hexadecimal string

A hexadecimal-encoded binary string that begins with the prefix 0x and can include the digits 0 through 9 and the uppercase and lowercase letters A through F. The interpretation of hexadecimal strings is platform specific. For some systems, the first byte after the prefix is the most significant; for others, the last byte. For example, the string 0x0100 is interpreted as 1 on some systems and as 256 on others.

I

identifier

A string of characters used to identify a database object, such as a table name or column name.

IDENTITY column

A column that contains system-generated values that uniquely identify each row in a table. IDENTITY columns store unique numbers, such as invoice numbers or employee numbers, that are generated automatically by Adaptive Server. The value of the IDENTITY column uniquely identifies each row in a table.

image function

See **text and image function**.

implicit conversions

Datatype conversions that Adaptive Server automatically performs to compare datatypes.

index

A database object that consists of key values from the data tables and pointers to the pages that contain those values. Indexes speed up access to data rows by pointing Adaptive Server to the location of a table column's data on disk.

index covering

A data access condition where the leaf-level pages of a nonclustered index contain the data needed to satisfy a query. The index must contain all columns in the select list as well as the columns in the query clauses, if any. The server can satisfy the query using only the **leaf level** of the index. When an index covers a query, the server does not access the data pages.

index selectivity

The ratio of duplicate key values in an index. An index is selective when it lets the optimizer pinpoint a single row, such as a search for a unique key. An index on nonunique entries is less selective. An index on values such as "M" or "F" (for male or female) is extremely nonselective.

initial response time

The time required to return the first result row of a query to a user. For some queries, initial response time can be very brief, even though time to return the full result set can take much longer.

initialization

See **disk initialization**.

inner query

See **subquery**.

in-place update

A type of **direct update** operation. An in-place update does not cause data rows to move on the data page. Compare to **on-page update** and **delete/insert direct update**.

int

A signed 32-bit integer value.

integrity constraints

A model to describe the **database integrity** in the create table statement.

integrity rules

Rules that describe how data will be kept accurate and consistent in the relational model. See also **database integrity**.

intent lock

A type of table-level lock. A shared intent lock indicates that there is a shared lock on a page of that table. It prevents other transactions from acquiring an exclusive table-level lock. An exclusive intent lock indicates that there is an exclusive or update lock on a page of that table. It prevents other transactions from acquiring an exclusive or shared table-level lock.

I/O

See **device I/O**.

isolation level

Specifies the kinds of actions that are not permitted while the current transactions execute; also called "locking level." The ANSI standard defines four levels of isolation for SQL transactions. Level 0 prevents other transactions from changing data already modified by an uncommitted transaction. Level 1 prevents **dirty reads**. Level 2 (not supported by Adaptive Server) also prevents **non-repeatable reads**. Level 3 prevents both types of reads and **phantoms**; it is equivalent to doing all queries with holdlock. The user controls the isolation level with the set option transaction isolation level or with the at isolation clause of select or readtext. The default is level 1.

J

join

A basic operation in a relational system that links the rows in two or more tables by comparing the values in specified columns.

join selectivity

> An estimate of the number of rows from a particular table that will join with a row from another table. If index statistics are available for the join column, Adaptive Server bases the join selectivity on the **density** of the index (the average number of duplicate rows). If no statistics are available, the selectivity is $1/N$, where N is the number of rows in the smaller table.

K

kernel

> A module within Adaptive Server that acts as the interface between Adaptive Server and the operating system. The kernel manages tasks associated with Adaptive Server's clients. For example, it tracks the state of the task, which is necessary in a multithreaded environment because of context switching.

key

> A field used to identify a record, often used as the **index** field for a table.

key value

> Any value that is indexed.

keyword

> A word or phrase that is reserved for exclusive use by Transact-SQL. Also known as a **reserved word**.

keyword phrase

> See **clause**.

L

language cursor

> A cursor declared in SQL without using Open Client. As with Adaptive Server cursors, Open Client is completely unaware of the cursors and the results are sent back to the client in the same format as a normal select.

Language Module

> A set of files, including localization files, that provide alternate language, sort order, and character sets for Adaptive Server.

last-chance threshold

A default threshold in Adaptive Server that suspends or kills user processes if the transaction log has run out of room. This threshold leaves just enough space for the de-allocation records for the log itself. The last-chance threshold always calls a procedure named sp_thresholdaction. This procedure is not supplied by Sybase. It must be written by the System Administrator.

leaf level

The level of an index at which all key values appear in order. For Adaptive Server clustered indexes, the leaf level and the data level are the same. For nonclustered indexes, the last index level above the data level is the leaf level, since key values for all of the data rows appear there in sorted order.

legacy data

Data from older, perhaps out-of-date, data sources or from data sources that are no longer supported by current standards.

libraries

See **Application Program Interface (API)**.

lightweight process

See **worker process**.

livelock

A request for an **exclusive lock** that is repeatedly denied because a series of overlapping **shared locks** keeps interfering. Adaptive Server detects the situation after three denials and grants a demand lock to the update transaction, queueing further shared lock requests after the demand lock.

load

To restore data stored in a backup created during a **dump**.

localization files

Files that contain translated error messages, character set and sort order definitions, and utilities for converting Adaptive Server's character set into the appropriate character set for a particular terminal.

local variable

A user-defined **variable** defined with a declare statement. See also **global variable**.

location transparency

A feature unique to Component Integration Services that makes remote data appear as if it were local. Users do not need to know where the data resides in order to access it.

lock

A concurrency control mechanism that protects the integrity of data and transaction results in a multiuser environment. Adaptive Server applies page or table locks to prevent two users from attempting to change the same data at the same time, and to prevent processes that are selecting data from reading data that is in the process of being changed.

lock promotion threshold

The number of page locks allowed in a table before Adaptive Server attempts to issue a table lock. If the table lock is successful, Adaptive Server releases the page locks.

locking

The process of restricting access to resources in a multi-user environment to maintain security and prevent concurrent access problems. Adaptive Server automatically applies locks to tables or pages.

locking level

See **isolation level**.

logical design

A design in which you define the tables, relations, and keys of a relational database. See also **physical design**.

logical expression

An expression that evaluates to TRUE (1), FALSE (0) or UNKNOWN (NULL). Logical expressions are often used in control of flow statements, such as if or while conditions.

logical key

The primary, foreign, or common key definitions in a database design that define the relationship between tables in the database. Logical keys are not necessarily the same as the **physical key**s (the keys used to create indexes) on the table.

logical operators

The operators and, or, and not. All three can be used in where clauses. The operator and joins two or more conditions and returns results when all of the conditions are true; or connects two or more conditions and returns results when any of the conditions is true.

logical read

The process of accessing a data or index page already in memory to satisfy a query. See also **physical read**.

login

The name a user uses to log into Adaptive Server. A login is valid if Adaptive Server has an entry for that user in the system table *syslogins*.

look-ahead set

The set of pages for a particular operation to be fetched by asynchronous prefetch. Each Adaptive Server operation that uses asynchronous prefetch builds a look-ahead set based on the known or expected set of pages that will be needed by the operation in the near future.

LRU buffer reuse strategy

A caching strategy for replacing the least-recently used buffers in the data cache. A clean data page is taken from the LRU end of the data cache to store a page read from disk. The new page is placed on the data cache's page chain at the MRU end of the cache, so that it stays in memory.

M

MAPI

Messaging Application Programming Interface. An e-mail application programming interface developed by Microsoft. MAPI support for Adaptive Server integrates Adaptive Server with MAPI to provide an interface for sending and receiving data by e-mail rather than through the traditional client/server connection-based facility.

master database

The system database that controls the user databases and the operation of Adaptive Server as a whole. Known as *master*, it keeps track of such things as user accounts, ongoing processes, and system error messages.

master table

> A table that contains data on which data in another table logically depends. For example, in the *pubs2* database, the *sales* table is a master table. The *salesdetail* table holds detail data that depends on the master data in *sales*. The detail table usually has a foreign key that joins to the primary key of the master table.

master-detail relationship

> A relationship between sets of data where one set of data logically depends on the other. For example, in the *pubs2* database, the *sales* table and *salesdetail* table have a master-detail relationship. See **detail** and **master table**.

matching index scan

> A scan using a nonclustered index when the query has a where clause (search argument) on a set of columns, and the columns form a **prefix subset** of keys on the index. The index is used to position the search at the first matching key, and then scanned forward for additional matches on the specified index key columns. The scan stops at the first row that does not match. Matching index scans are quite fast and efficient. Compare to **nonmatching index scan**.

mathematical function

> A function that returns values commonly needed for operations on mathematical data.

media failure

> A media failure occurs when the information on a medium (typically a hard disk drive) becomes unusable.

message number

> The number that uniquely identifies an error message.

metadata

> Data about data. Metadata is stored in local proxy tables by Component Integration Services. The metadata stored by Component Integration Services represents schemas with information about remote tables.

metadata cache

> A reserved area of memory used for tracking information on indexes, objects, or databases. You can configure the size of the metadata caches based on the number of **metadata descriptors** used by indexes, objects, or databases.

metadata descriptor

A memory data structure that represents the state of an index, an object, or a database while it is in use or cached between uses.

MIME

Multipurpose Internet Mail Extentions. A format for exchanging complex messages through the Internet. Web browsers and servers use MIME types to describe the messages transmitted from the server to the browser.

mirror

See **disk mirror**.

model database

A template for new user databases. The installation process creates *model* when Adaptive Server is installed. Each time the create database command is issued, Adaptive Server makes a copy of *model* and extends it to the size requested, if necessary.

modulo

An **arithmetic operator** represented by the percent (%) sign that gives the integer remainder after a division operation on two integers. For example, 21 % 9 = 3 because 21 divided by 9 equals 2 with a remainder of 3.

MRU replacement strategy

A caching strategy for table scans and nonclustered index scans. The optimizer chooses this strategy when it determines that the pages need to be accessed only once for a particular query. Instead of adding all of the pages to the MRU/LRU chain, the pages are immediately flushed as soon as the query finishes examining them, and the next page for the query is read into the buffer.

multibyte character set

A **character set** that includes characters encoded using more than one byte. EUC JIS and Shift-JIS are examples of character sets that include several types of characters represented by multiple bytes in a Japanese language environment.

multiprocessing

Multiple processes that share memory or use some type of synchronized method for passing messages between them are performing multiprocessing.

N

named cache

See **buffer cache**.

natural join

A **join** in which the values of the columns being joined are compared on the basis of equality, and all the columns in the tables are included in the results, except that only one of each pair of joined columns is included.

nested queries

select statements that contain one or more subqueries.

nested select statements

See **nested queries**.

nonclustered index

An **index** that stores key values and pointers to data. The **leaf level** points to data pages rather than containing the data itself.

nonmatching index scan

A scan using a nonclustered index when the search arguments do not form a **prefix subset** of the index key columns, although they match some parts of the **composite key**. The scan is performed using the index from the lowest key value to the highest key value, searching for the matches specified in the query. This type of scan is performed on nonclustered indexes when all columns for a table referenced in the query are included in the index. Although cheaper than a table scan, a non-matching index scan is more expensive than a **matching index scan**.

normalization rules

The standard rules of database design in a relational database management system.

not-equal join

A join on the basis of inequality.

null

A value given to a parameter or option that has no explicitly assigned value. NULL is not equivalent to zero, or to blank. A value of NULL is not considered to be greater than, less than, or equal to any other value, including another value of NULL.

numeric expression

See **arithmetic expression**.

O

object

A passive entity that contains or receives information, and that cannot change the information it contains. In Adaptive Server, objects include rows, tables, databases, stored procedures, triggers, defaults, and views. See also **database object**.

Object Allocation Map (OAM)

One or more pages within a table or an index on a table that contain pointers to the allocation pages for each **allocation unit** where the object uses space.

object permission

A **permissions** that regulates the use of certain commands (**data modification** commands, plus select, truncate table and execute commands) to specific tables, views or columns. See also **command permission**.

ODBC

Open Database Connectivity. The ODBC interface, defined by Microsoft Corporation, is a standard interface to database management systems in the Windows and Windows NT environments.

online transaction processing (OLTP)

Real-time access to and manipulation of data. Commonly used for business applications, for example, a bank customer's transactions at an automatic teller machine. It is characterized by a large number of users that need fast online access to small result sets.

on-page update

A type of **direct update** operation, performed when the length of the data row changes. The changed data row remains on the same data page, but other rows on the page may move. Contrast to **in-place update** and **delete/insert direct update**.

operating system

A group of programs that translates your commands to the computer, so that you can perform such tasks as creating files, running programs, and printing documents.

operating system file

A collection of data named and recognized by the **operating system**. Adaptive Server data is not stored in operating system files, but can be exported to operating system files by using the **bulk copy** operation.

Operator

The role required to allow the ability to back up and restore databases on a server-wide basis. See also **roles**.

operators

Symbols that act on two values to produce a third. See also **comparison operator**, **logical operators**, and **arithmetic operators**.

optimizer

Adaptive Server code that analyzes queries and database objects and selects the appropriate **query plan**. The Adaptive Server optimizer is a cost-based optimizer. It estimates the cost of each permutation of table accesses in terms of CPU cost and I/O cost.

OR strategy

An **optimizer** strategy for resolving queries using **or** and queries using **in** (*values list*). Indexes are used to retrieve and qualify data rows from a table. The row IDs are stored in a worktable. When all rows have been retrieved, the worktable is sorted to remove duplicates, and the row IDs are used to retrieve the data from the table.

outer join

A **join** in which both matching and nonmatching rows are returned. The operators *= and =* are used to indicate that all the rows in the first or second tables should be returned, regardless of whether or not there is a match on the join column.

outer query

Another name for the principal **query** in a statement containing a **subquery**.

overflow page

A data page for a table with a nonunique clustered index, which contains only rows that have duplicate keys. The key value is the same as the last key on the previous page in the chain. There is no index page pointing directly to an overflow page.

ownership chain

Dependencies of views on other views and/or tables, and of procedures on other procedures, views, and/or tables.

P

page

A 2K block of data that is the minimal unit that can be read from or written to disk.

page chain

See **partition**.

page lock

A page lock locks an entire data or index page. Adaptive Server uses page locks as frequently as possible to reduce the contention for data among users and to improve concurrency. A page lock is less restrictive than a table lock.

page split

An action performed by Adaptive Server when new data or index rows need to be added to a **page**, and there is not enough room for the new rows. Usually, the data on the existing page is split approximately evenly between the newly allocated page and the existing page.

page stealing

The allocation of a new last page for a partition from a **device** or **extent** that was not originally assigned to the partition. This action is performed as needed by Adaptive Server.

parallel clustered index partition scan

A **partition-based access method** in which multiple worker processes simultaneously scan data pages in a partitioned table when the clustered index key matches a search argument. One worker process is assigned to each partition. A worker process may also traverse the clustered index first to find the first data page in the partition that satisfies the search argument.

parallel hash-based table scan

A **hash-based access method** in which multiple worker processes scan a single chain of data pages in a table at the same time and apply an internal hash function to each page ID. The hash function determines which worker process reads the rows in current page. Typically, the number of worker processes is no more than two or three because it takes only two to three worker processes to fully utilize the I/O of a given physical device.

parallel nonclustered index hash-based scan

A **hash-based access method** in which multiple worker processes simultaneously scan level 0 index pages and apply a hash function to the data page IDs or the key values in each index page. The hash function determines which worker process scans the data rows referenced by the data page ID or key value.

parallel partition scan

A **partition-based access method** in which multiple worker processes simultaneously scan partitions in a partitioned table. The number of worker processes is equal to the number of partitions.

parallel processing

The simultaneous execution of Adaptive Server tasks or subtasks. Multiple processes share memory or use some type of synchronized method for passing messages between them.

parallel sort

A technique that employs multiple worker processes to sort data in parallel, either in response to a create index command or to a query that requires an internal sort. A single process partitions the input data into discrete ranges; multiple processes simultaneously sort each individual range of data rows and create subindexes; then a single process merges the sorted ranges and indexes into one sorted index or result set.

parameter

An **argument** to a **stored procedure** or **system procedure**.

partition

A linked chain of database pages that stores a **database object**. You can divide a table into multiple partitions that reside on separate physical devices to improve the performance of concurrent inserts and to facilitate parallel queries and sorts.

partition-based access method

A parallel **access method** that uses two or more worker processes to access separate partitions of a table. This category includes **parallel partition scan** and **parallel clustered index partition scan**.

passthrough mode

A mode that allows clients to communicate with remote servers in native syntax. The Transact-SQL parser and compiler are bypassed in this mode, and each language "batch" received from the user is passed directly to the server to which the user is connected.

performance

The speed with which Adaptive Server processes queries and returns results. Performance is affected by several factors, including indexes on tables, use of raw partitions compared to files, and segments.

permissions

The authority to perform certain actions on certain database objects or to run certain commands.

phantoms

Occur when one transaction reads a set of rows that satisfy a search condition, and then a second transaction modifies the data (through an insert, delete, update, and so on). If the first transaction repeats the read with the same search conditions, it obtains a different set of rows.

physical design

Mapping the **logical design** to the Transact-SQL data definition commands that actually create the databases on the server.

physical key

A column name, or set of column names, used in a create index statement to define an index on a table. A physical key on a table is not necessarily the same as the **logical key**.

physical read

A disk I/O to access a data, index, or log page. Adaptive Server estimates physical reads and logical reads when optimizing queries. See also **logical read**.

point query

A **query** that restricts results to a single specific value, usually using the form "where *column_value = search_argument*".

precision

The maximum number of decimal digits that can be stored by *numeric* and *decimal* datatypes. The precision includes all digits, both to the right and to the left of the decimal point.

predicate

A clause that is based on a **comparison operator**. It does not apply to and, or, or not.

prefetch

> The process of performing multipage I/O's on a table, nonclustered index, or the transaction log. For logs, the server can fetch up to 256 pages, for nonlog tables and indexes, the server can fetch up to 8 pages.

prefix subset

> Used to refer to keys in a composite index. Search values form a prefix subset when leading columns of the index are specified. For an index on columns A, B, and C, these are prefix subsets: A, AB, ABC. These are not: AC, B, BC, C. See also **matching index scan** and **non-matching index scan**.

primary key

> The **column** or columns whose values uniquely identify a **row** in a table.

primary key constraint

> A **primary key** constraint is a unique constraint that does not permit null values for the component key columns. There can only be one primary key constraint per table. The primary key constraint creates a unique index on the specified columns to enforce this data integrity.

privilege

> See **permissions**.

process

> An execution environment scheduled onto physical CPUs by the operating system.

process affinity

> A process in which a certain Adaptive Server task runs only on a certain engine or a certain engine runs only on a certain CPU.

producer process

> In a **parallel sort**, producer processes read data simultaneously from the input table, determine the range to which each data value belongs, and distribute data values to consumer processes associated with the proper ranges.

projection

> A subset of the columns in a table. It is one of the basic **query** operations in a relational system. See also **selection** and **view**.

proxy authorization

The ability to impersonate another user in the server. A System Security Officer can grant proxy authorization to a user. Proxy authorization allows administrators to check permissions for a particular user or to perform maintenance on a user's database objects. Application servers can log into the server and execute procedures and commands on behalf of several users.

proxy table

A local table that has been mapped to a table on a remote server. The proxy table contains metadata and is used to access the remote table as if it were a local table.

Q

qualified

The name of a database object can be qualified, or preceded by, the name of the database and the object owner.

query

A SQL statement or group of SQL statements that access and/or manipulate data in a database. See also **data retrieval**.

query plan

The ordered set of steps required to carry out a **query**, complete with the access methods chosen for each table.

query tree

An internal tree structure that represents the user's **query**. A large portion of query processing and compilation is built around the shape and structure of this internal data structure. For stored procedures, views, triggers, rules and defaults these tree structures are stored in the *sysprocedures* table on disk, and read back from disk when the procedure or view is executed. See also **source text**.

R

range query

A **query** that requests data within a specific range of values. These include greater than and less than queries, queries using between, and some queries using like.

RDBMS

See **relational database management system (RDBMS)**.

read access

Permission to read an object (for example, to select rows from a table). See also **permissions**.

record

See **row**.

recovery

The process of rebuilding one or more databases from database and log **dumps**. See also **automatic recovery**.

referential integrity

The rules governing data consistency, specifically the relationships among the **primary keys** and **foreign keys** of different tables. Adaptive Server addresses referential integrity with user-defined **triggers** and with **referential integrity constraints**.

referential integrity constraint

A requirement that data inserted into a "referencing" table, the table that defines the constraint, must have matching values in a "referenced" table. You cannot delete rows or update column values from a referenced table that match values in a referencing table. Also, you cannot drop the referenced table until the referencing table is dropped or the referential integrity constraint is removed.

reformatting strategy

A strategy used by Adaptive Server to resolve join queries on large tables that have no useful index. Adaptive Server builds a temporary clustered index on the join columns of the inner table, and uses this index to retrieve the rows. Adaptive Server estimates the cost of this strategy and the cost of the alternative—a table scan—and chooses the least expensive method.

relational database

A set of related data tables and other **database objects** that are organized and presented to serve a specific purpose. See also **data warehouse**.

relational database management system (RDBMS)

A system for storing and retrieving data from two-dimensional tables in which the use of SQL is standard.

relational expression

A type of Boolean or logical expression of the form:

arith_expression relational_operator arith_expression

In Transact-SQL, a relational expression can return TRUE, FALSE, or UNKNOWN. The results can evaluate to UNKNOWN if one or both of the expressions evaluates to NULL.

relational operator

An operator that compares two operands and yields a truth value, such as "5 < 7" (TRUE), "ABC" = "ABCD" (FALSE) or "@value > NULL" (UNKNOWN).

relationship

A description of how entities are related. A basic step in **logical design** of a database is to identify the relationships between the entities you have identified. See also **entity**.

remote login

A login to a remote server.

remote procedure call (RPC)

A **stored procedure** that is executed on an Adaptive Server other than the server the user is logged into.

repeating subquery

See **correlated subquery**.

replication

For databases, a process by which the changes to data in one database (including creation, updating, and deletion of records) are also applied to the corresponding records in other database.

reserved word

See **keyword**.

response time

The time it takes for a single task, such as sending a Transact-SQL query to Adaptive Server, to complete. See also **initial response time**.

restriction

See **selection**.

return status

A value that indicates that the procedure completed successfully or indicates the reason for failure.

RID

See **row ID**.

roles

Titles recognized by Adaptive Server that provide individual accountability for users performing system administration and security-related tasks in Adaptive Server. The **System Administrator**, **System Security Officer**, and **Operator** roles can be granted to individual server login accounts. In addition to these system roles, a System Security Officer can create user-defined roles, such as "financial analyst" and "salary administrator".

rollback transaction

A Transact-SQL **statement** used with a **user-defined transaction**, before a commit transaction has been received, that cancels the transaction and undoes any changes that were made to the database.

row

A set of related **column**s that describes a specific **entity**. Also called a **record**.

row aggregate function

Functions (sum, avg, min, max, and count) that generate a new **row** for summary data when used with compute in a select **statement**.

row ID

A unique, internal identifier for a data **row**. The row ID, or RID, is a combination of the data page number and the row number on the page.

RPC

See **remote procedure call (RPC)**.

rule

A specification that controls what data can be entered in a particular **column** or in a column of a particular **user-defined datatype**.

run value

The value of the configuration parameter currently in use.

S

"sa" login account

> See **System Administrator**.

SARG

> See **search argument**.

savepoint

> A marker that the user puts inside a **user-defined transaction**. The user can later use the rollback transaction command with the savepoint name to cancel any commands up to the savepoint, or commit transaction to actually complete the commands. See also **transaction** and **rollback transaction**.

scalar aggregate

> An **aggregate function** that produces a single value from a select statement that does not include a group by clause. This is true whether the aggregate function is operating on all the rows in a table or on a subset of rows defined by a where clause. See also **vector aggregate**.

scale

> The maximum number of digits that can be stored to the right of the decimal point by a *numeric* or *decimal* datatype. The scale must be less than or equal to the **precision**.

scan descriptor

> An internal data structure that manages the scan of tables being queried, particularly tables that have references to other tables.

schema

> A collection of objects associated with a particular schema name and **schema authorization identifier**. The objects can be tables, views, domains, constraints, assertions, privileges, and so on. A schema is created by a create schema statement.

schema authorization identifier

> All the objects are said to be owned by or to have been created by the associated schema authorization identifier for the schema.

scope

> The context in which a feature, such as a **cursor** or a **global variable**, is used. In Adaptive Server, the scope can be an individual user session, a **stored procedure**, or a **trigger**.

search argument

A predicate in a query's where clause that can be used to locate rows via an index. Also referred to as SARG.

segment

A named subset of database devices available to a particular database. It is a label that points to one or more database devices. Segments can be used to control the placement of tables and indexes on specific database devices.

select list

The **columns** specified in the main clause of a select statement. In a **dependent** view, the target list must be maintained in all underlying views if the dependent view is to remain valid.

selection

A subset of the **rows** in a table. Also called a restriction, it is one of the basic **query** operations in a relational system. See also **producer process** and **view**.

selectivity

See **index selectivity** and **join selectivity**.

self-join

A **join** used for comparing values within a **column** of a **table**. Since this operation involves a join of a table with itself, you must give the table two temporary names, or **correlation names**, which are then used to qualify the column names in the rest of the query.

semaphore

A semaphore is a simple internal locking mechanism that prevents a second task from accessing the data structure currently in use. Adaptive Server uses semaphores to protect transaction logs, user log caches, and I/O devices. Like a **spinlock**, a semaphore is relevant only in SMP environments.

server class

An interface specification defining remote access for use with Component Integration Services. For each server class there is a separate access method that handles interaction between the server and remote servers of the same class. Each server class has a set of unique characteristics that System Administrators and programmers use to configure the server for remote data access. The characteristics are:

– Mapping between local tables (proxy tables) and external objects

– Datatype conversions specific to the class or access method

– Special considerations for each class

– Restrictions, if any, on Transact-SQL statements applied to the class

server cursor

A **cursor** declared inside a **stored procedure**. The client executing the stored procedure is not aware of the presence of these cursors. Results returned to the client for a fetch appear exactly the same as the results from a normal select.

server engine

See **engine**.

server user ID

The ID number by which a user is known to Adaptive Server.

severity level number

The severity of an error condition.

shared lock

A type of lock acquired on an object for a read operation. It does not allow other transactions to acquire exclusive locks on the object but allows them to acquire shared locks. Shared locks can be obtained on a table or page.

soft fault

A corruption in Adaptive Server that is usually not persistent. Most soft faults result from temporary inconsistencies in the target database caused by users updating the target database during the dbcc checkstorage operation or by dbcc checkstorage encountering Data Definition Language (DDL) commands. These faults are not repeated when you run dbcc checkstorage a second time. See also **hard fault**.

sort order

A specification used by Adaptive Server to determine the order in which to sort character data. Also called **collating sequence**.

source text

SQL statements that define a **compiled object**. Source text is stored in the *text* column of the *syscomments* table. It is used during upgrade processes; therefore, it must not be deleted. Source text can be encrypted using sp_hidetext.

spinlock

A simple internal locking mechanism that prevents a process from accessing the resource currently used by another process. All processes trying to access the resource must wait (or "spin") until the lock is released. Spinlocks protect internal data structures such as a data cache.

SQL

See **Structured Query Language (SQL)**

SQL Server

See **Adaptive Server**.

statement

One or more complete Transact-SQL **commands**. A statement usually consists of a **keyword**, for example, select, followed by clauses such as from or where.

statement block

A series of Transact-SQL **statements** enclosed between the **keywords** begin and end so that they are treated as a unit.

stored procedure

A collection of Transact-SQL **statements** and optional control-of-flow statements stored under a name. See also **system procedure**.

string function

A function that operates on strings of characters or binary data. substring and charindex are Transact-SQL string functions.

Structured Query Language (SQL)

The language used to communicate with a relational database and that is the subject of standards set by several standards bodies.

subject

A server process acting on behalf of a user. In Adaptive Server, subjects include users, and **stored procedure**s and trusted **trigger**s while executing.

subquery

A select **statement** that is nested inside another select, insert, update, or delete statement or inside another subquery.

suspect page

A page that recovery has marked as suspect because of corruption. Suspect pages are normally inaccessible to users unless they have been forced online by special procedures.

symmetric multiprocessing (SMP)

An environment in which multiple CPUs simultaneously process client tasks. Any CPU can execute any task.

system administration

An assortment of tasks including but not limited to managing Adaptive Server's physical storage, creating and backing up databases, creating user accounts, granting permissions, and running diagnostic and repair functions. See also **database administration**.

System Administrator

A user in charge of Adaptive Server system administration, including managing disk storage, granting and revoking the System Administrator role, and creating new databases. The "sa" account, a single **login**, is created when Adaptive Server is installed. This login is configured with both the System Administrator and **System Security Officer** roles. To increase individual accountability, use the "sa" account to assign **roles** to individual logins.

system database

A database provided Sybase, for example, *master*, *tempdb*, *model*, and *sybsystemprocs*.

system function

A function that returns special information from the database, particularly from the **system table**s.

system procedure

A **stored procedure** residing in *sybsystemprocs*, which is provided by Sybase. System procedures provide shortcuts for retrieving information from the **system tables**, or mechanisms for accomplishing database administration and other tasks that involve updating system tables.

system procedure tables

Tables in the *master* database that the **system procedure**s use to convert internal system values (for example, status bits) into human-readable format.

System Security Officer

A user who controls security-related operations in Adaptive Server, including auditing, locking and unlocking login accounts, creating user-defined roles, and password management. See also **System Administrator**.

system table

One of the data dictionary tables. System tables keep track of information about the Adaptive Server as a whole and about each user database. The *master* database contains some system tables that are not in user databases.

T

table

A collection of **row**s that have associated **column**s. The logical equivalent of a database file.

table scan

A method of accessing a table by reading every row in the table. Table scans are used when there are no conditions (where clauses) on a query, when no index exists on the clauses named in the query, or when the Adaptive Server optimizer determines that an index should not be used because it is more expensive than a table scan. See also **access method**.

table-level constraint

A constraint that limits values on more than one column of a table. Enter table-level constraints as separate comma-delimited clauses in the create statement. You must declare constraints that operate on more than one column as table-level constraints.

target database

The database that you want to check with dbcc.

task

A unit of execution scheduled by the kernel to fulfill a request for service.

temporary database

A database that provides a storage area for temporary tables and other temporary working storage needs (for example, intermediate results of group by and order by). In Adaptive Server the temporary database is *tempdb.*

text and image function

A function that operates on *text* and *image* data. The text and image functions include patindex, textptr, and textvalid.

text chain

A special data structure used to store text and image values for a table. Data rows store pointers to the location of the text or image value in the text chain.

theta join

A **join** that uses **comparison operators** as the join condition.

thread

See **worker process**.

thread of execution

See **worker process**.

threshold

The estimate of the number of log pages required to back up the transaction log, and the action to be taken when the amount of space falls below that value.

throughput

The volume of work completed in a given time period. It is usually measured in transactions per second (TPS).

time slice

The amount of time a task is allowed to run on a Adaptive Server engine before voluntarily yielding the engine to another task. To see the value on your server use:

```
sp_configure time slice
```

Transact-SQL

The SQL dialect used in Sybase SQL Server.

transaction

A group of Transact-SQL statements that is treated as a single unit of work. Either all statements in the group are executed or no statements are executed. The tables queried during the transaction are locked until the transaction is complete.

transaction log

A **system table** (*syslogs*) in which all changes to the database are recorded.

trigger

A special form of **stored procedure** that goes into effect when a user gives a change command such as insert, delete, or update to a specified table or column. Triggers are often used to enforce **referential integrity**.

trigger actions

The action for which a trigger is specified.

trigger conditions

The conditions that cause a trigger to take effect.

trigger table

The table to which a trigger is attached.

trigger test tables

When a data modification affects a key column, triggers compare the new column values to related keys by using temporary work tables called trigger test tables.

trusted user

A user who is authorized to operate over a range of sensitivity levels in a session. The user logs in by specifying a range of levels over which he or she would like to operate, and can change his or her current read and write levels during a session.

U

unchained transaction mode

The transaction mode that requires explicit begin transaction statements paired with commit transaction or rollback transaction statements to complete a transaction. See also **chained transaction mode**.

underlying tables

See **base tables**.

ungrouped aggregate

See **scalar aggregate**.

unified login

The capability for users to be pre-authenticated by a security mechanism before logging into the server. This capability enables a user to log into several servers without having to supply a login name and password for every connection.

unique constraint

A constraint requiring that all non-null values in the specified columns must be unique. No two rows in the table are allowed to have the same value in the specified column. The **unique** constraint creates a unique index on the specified columns to enforce this data integrity.

unique indexes

Indexes that do not permit any two rows in the specified columns to have the same value. Adaptive Server checks for duplicate values when you create the index (if data already exists) and each time data is added.

update

An addition to, deletion from, or change in data, involving the **insert**, **delete**, **truncate table**, or **update** statements.

update in place

See **in-place update**.

update lock

A type of lock acquired on an object during the scanning phase of an update or delete operation. It does not allow other transactions to acquire update or exclusive locks on the object but allows them to acquire shared locks. Update locks can be obtained only on a page.

user database

A database created by a user. See also **system database**.

user ID

The ID number by which a user is known in a specific database. See also **server user ID**.

user-defined cache

See **buffer cache**.

user-defined datatype

A definition, created by the user, of the type of data (the datatype) contained in a column. The user defines these **datatypes** in terms of the existing system datatypes. Rules and defaults can be bound to user-defined datatypes, but not to system datatypes.

user-defined transaction

See **transaction**.

V

variable

An **entity** that is assigned a value. Adaptive Server has two kinds of variables, **local variable** and **global variables**.

vector aggregate

A value that results from using an aggregate function with a group by clause. See also **scalar aggregate**.

view

A named select **statement** that is stored in the database as an object. It allows users to see a subset of rows or columns from one or more tables. See also **producer process** and **selection**.

view resolution

In queries that involve a **view**, view resolution is the process of verifying the validity of database objects in the **query** and combining the query and the stored definition of the view.

Virtual Server Architecture (VSA)

A database architecture designed to make efficient use of hardware and operating system resources. It is Sybase's implementation of SMP processing. VSA is based on the use of Adaptive Server engines to achieve fully symmetric processing and Sybase's own database kernel—a multithreaded operating system that handles typical operating system functions, including scheduling and dispatching tasks.

W

wash area

An area of a **buffer pool** near the LRU end of the MRU/LRU page chain. After pages enter the wash area, Adaptive Server initiates an asynchronous write on the pages. The wash area provides clean buffers at the LRU for any query that needs to perform a disk I/O.

wildcard character

A special character used with the Transact-SQL like keyword that can stand for one or any number of characters in pattern matching.

worker process

An Adaptive Server subtask spawned by an Adaptive Server task in a parallel query processing environment. Together with other worker processes, executes parts of a query simultaneously to reduce response time. Worker processes use only a small amount of the operating system resources that a process uses. They do not need any operating system resources after they are launched. They can share memory space with each other. Also called a thread or lightweight process.

worktable

See **dynamic index**.

write access

Permission to write an object, for example, to update a row or to add a row to a table.

write-ahead log

A log, such as the transaction log, that Adaptive Server automatically writes to when a user issues a statement that would modify the database. After all changes for the statement have been recorded in the log, they are written to an in-cache copy of the data page.

X

XP Server

An Open Server™ process that executes extended stored procedures on behalf of Adaptive Server. XP Server and Adaptive Server run on the same machine and communicate with each other through **remote procedure call**s.