StorHouse Glossary

StorHouse/SM, RM, and RFS

StorHouse Release 5.6
Publication Number
007-6317-001

November 19, 2013

Publication Number: 007-6317-001

Contents

Contents

Welcome

Welcome to the *StorHouse Glossary*. This document contains definitions for the terms used in the documentation for StorHouse/SM, StorHouse/RM, and StorHouse/RFS.

Purpose of this Document

The purpose of this document is to define the terms used in the StorHouse product documentation in one central location, thereby making it easier for readers to find the information they need.

Intended Audience

The *StorHouse Glossary* is for all StorHouse users.

Document Organization

This document contains one chapter, which defines the glossary terms.

C H A P T E R 1

List of Terms

The StorHouse product documentation uses the following terms.

$$BUFFER

The file set name assigned to the performance buffer.

absolute revision number

See revision number.

absolute time

A format for expressing time in StorHouse Command Language commands. Absolute time is a StorHouse calendar and clock time.

Format: day-month-year:hour:minute:second

Example: 12-FEB-2001:03:00:00

absolute version number

See version number.

access group

See file access group.

access mode

One of four modes you use to open a StorHouse file:

- READ – Enables you to read records in SEQUENTIAL, RECORD, KEYED, and KEYSEQUENTIAL files.
- WRITE – Enables you to write records to a SEQUENTIAL file.
- APPEND – Enables you to write new records to RECORD, KEYED, and KEYSEQUENTIAL files.
- UPDATE – Enables you to read, change, and delete records in RECORD, KEYED, and KEYSEQUENTIAL files.

access privileges

The privileges that enable a StorHouse account to bypass various security checks and to perform specific StorHouse functions. For example, the StorHouse SQLADMIN access privilege gives an account DBA access to all StorHouse databases.

Access Time Factor (ATF)

An attribute that works with the StorHouse migrate function to keep data most likely to be accessed in the StorHouse performance buffer while maintaining a supply of free space. StorHouse migrates files with higher ATF values from the performance buffer to their destination file sets before files with lower ATF values.

accessor

The component of a robotic arm that holds optical disk or tape cartridges. Also called *picker* and *gripper*.

account, StorHouse

A collection of administrative data used to monitor and control the use of StorHouse. Each account has its own set of access and command privileges. An account can be dedicated to one user or shared by multiple users. For example, a system administrator can have one account, and system operators can share another account.

account identification code (aid), StorHouse

An identifier or name for a StorHouse account. This 1- to 12-character account ID can consist of letters, numbers, a dollar sign ($), or an underscore (_). You can delimit account IDs that do not follow these conventions.

active database

A database that has one or more sessions currently connected.

active load, StorHouse/RM

A load that is in progress (currently running), successfully completed but not confirmed, or incomplete (failed) with errors.

active release, StorHouse/RM

The StorHouse/RM software release that is current when the StorHouse software is started.

active set

See result set.

administration log

A system-generated log containing hardware status, error information, and general data used by a SGI® customer support representative to analyze system operations.

aggregate function

An SQL operation that derives its result from a collection of values across one or more rows of a database table. For example, the aggregate function AVG computes the average of a set of numbers. See also function and scalar function.

aid

See account identification code.

alias

An alternate name that refers to a table or a view in an SQL statement. An alias is valid only for the life of a query.

Alternative to Microfiche/Microfilm Online (AMMO-II)™

SGI report distribution and management software. AMMO-II is a menu-driven, online viewing package that enables you to access reports residing on host DASD and on StorHouse.

AMMO-II™

See Alternative to Microfiche/Microfilm Online.

answer set

See result set.

API

See application program interface.

application program interface (API)

A programming language interface between a system control program or licensed program and a user program. Also called *application programming interface*. The StorHouse Callable Interface is an example of an application program interface.

archive process, StorHouse

To copy files from a primary StorHouse volume set to an archive volume set, typically in preparation for exporting the files from StorHouse. The primary copy remains intact. The Command Language ARCHIVE command enables a system administrator to create an archive copy on an archive volume set.

archive copy

A copy of a primary StorHouse file. Archive copies are stored on one or more archive volume sets in the StorHouse archive directory.

archive directory, StorHouse

The directory that contains information about archive copies of primary files.

argument

A component of an SQL statement that completes or modifies the meaning of the keyword it follows. *See also* keyword.

arithmetic operator

An operator that performs arithmetic operations on column values. StorHouse SQL supports five arithmetic operators: + (addition), - (subtraction), * (multiplication), / (division), and unary + and -. *See also* unary operator.

array

A group of data items, or elements, assigned to one variable name. Arrays enable you to process a collection of elements with one SQL statement.

ASCII transfer type, StorHouse/RM

A mode of transfer in which data is passed as a stream of ASCII text records that end with ASCII carriage return/line feed (CRLF) pairs. You can specify the transfer type when loading data with the SGI FTP Data Loader or unloading data with the SGI FTP Data Unloader. Contrast BINARY transfer type. See *also* transfer type.

ATF

See Access Time Factor.

atomic

A state where all operations in a transaction are either committed or rolled back. *See also* commit and roll back.

authorization manager, StorHouse/RM

The StorHouse/RM process that checks granted privileges on database components.

auto-join

A type of join that combines a table with itself. Also called *self-join*.

automatic metadata recovery process, StorHouse/RM

The StorHouse/RM software that restores metadata when a StorHouse engine abnormally terminates during COMMIT or metadata DELETE, INSERT, or UPDATE processing as well as when there's a power loss or an operating system crash.

back end communications manager, StorHouse/RM

The StorHouse/RM server software that unmarshals SQL statements and marshals (arranges) result set data.

back up, StorHouse

To copy all new file extents from the StorHouse performance buffer to their primary file sets. The Command Language BACKUP command enables a system administrator to back up new file extents. *See also* write-back.

backup/nobackup attribute

The StorHouse file attribute that indicates whether you can back up and archive a file. The system parameter BACKUP is the default for new files. The Command Language SET FILE command enables you to change this attribute for file versions.

backup copy, StorHouse

A copy of the primary file. You can use a backup copy to recover a corrupt or destroyed primary copy. Backup copies are stored on one or more backup volume sets.

backup directory, StorHouse

The directory that contains information about backup copies of primary files.

bar code label

An external label on a tape cartridge.

bar code reader

A tape library device, mounted on the robotic arm that automatically reads bar code labels on tape volumes. Bar code readers enable the system to identify a tape volume without having to load it into a drive. *See* slot.

basic predicate

A type of predicate that compares two values with a relational operator. If any expression in the predicate evaluates to null, the result is unknown.bin

blank volume

An uninitialized volume, meaning StorHouse has not written an internal volume label on the volume. (Note that blank tape volumes are prelabeled with an external bar code.) StorHouse requests the operator to load a blank volume into the exchange station when the number of volumes in any free pool volume set falls below its minimum value. *Compare* empty volume.

binary large object

See BLOB.

BINARY transfer type, StorHouse/RM

A transfer mode where data is passed intact from a data file. You can specify the transfer type when loading data with the SGI FTP Data Loader or unloading data with the SGI FTP Data Unloader. Contrast ASCII transfer type. See also transfer type.

bind descriptor

See input SQLDA.

blank

See whitespace.

BLOB

A variable-length array of bytes up to 2 gigabytes (GB) in size. Examples of BLOBs are audio and video clips, photos, and graphics. See also large object.

boolean operator

An operator that relates one or more true/false values and produces a single true/false value. StorHouse supports three boolean operators: AND, NOT, and OR. Also called *logical operator*.

byte

The representation of a character.

byte order

The sequence where a host stores binary values in memory. Some hosts store them with the most significant byte (msb) first and the least significant byte (lsb) last. Others store them in reverse order (lsb to msb). When loading data with the SGI FTP Data Loader, you can specify the native values key to identify the byte order of your client host. The SGI MVS Data Loader utility automatically uses the appropriate byte order for MVS hosts (msb to lsb). *See also* native values key.

C-style comment

Comment text bracketed by the characters /* and */. You can use C-style comments in C language statements, static SQL, or SQL that is dynamically prepared. Contrast SQL-style comment.

cache

High-speed buffer storage that contains frequently accessed instructions and data. Its purpose is to reduce access time.

cache directory, StorHouse/RFS

The location where StorHouse/RFS places data that it reads from StorHouse.

callable Interface

The StorHouse application program interface (API) that enables user application programs to access StorHouse. Through the Callable Interface, you can perform file functions, transfer data, use selected Command Language commands, and receive StorHouse responses.

Call Home

An error reporting procedure that automatically identifies error conditions at customer sites, reports them to the SGI Support Center at headquarters, and notifies SGI customer support personnel, as appropriate.

Cartesian product

A result set from a join operation that omits a restrictive clause. In the Cartesian product of a set of tables, each row is the row in the first table, concatenated with a row in the second table, concatenated with a row in the third table, and so on.

cartridge

See optical disk cartridge or tape cartridge.

catalog, StorHouse

To add file and file set information for uncataloged volumes on a volume set to the StorHouse catalog. The Command Language CATALOG VSET command enables your system administrator to catalog volume sets. *Contrast* uncatalog.

catalog, StorHouse/RM

See system table.

CCSID

Coded Character Set Identifier. An identifier for describing input data and control statements used by the SGI MVS Data Loader utility and the SGI FTP Data Loader. StorHouse supports the following CCSIDs: 500 (EBCDIC code page), 819 (ISO 8859-1 code page), and 850 (PC code page). The default CCSID is 500 for MVS and 819 for FTP. See also character set.

cell

The intersection of a column and a row in a table. Also called *data value*.

character large object

See CLOB.

character literal

A type of constant that specifies a character or character string consisting of letters, numbers, blanks, or special characters enclosed in quotes. *See also* literal.

character set

A finite group of characters defined for a specific purpose. When you load data, you specify the character set of the input data. StorHouse supports EBCDIC (CCSID 500), ISO 8859-1 (CCSID 819), and PC (CCSID 850) character sets as well as the following ORACLE® character set names: WE8EBCDIC500, WE8PC850, and WE8ISO8859P1. See also CCSID.

checkfile utility, StorHouse/RFS

A StorHouse/RFS utility that determines the location of a CRC error

checkpoint, StorHouse

A StorHouse Interactive Interface function that takes a snapshot of system files and saves them to a StorHouse volume set on magnetic tape. Your system administrator can initiate this function by issuing the StorHouse Command Language CHECKPOINT command. In the unlikely event that one or more system files becomes corrupted such that they cannot be recovered from shadowed copies, they can be recovered from checkpointed copies.

Also, a Callable Interface file function that allows you to create restart points during VRAM™ append operations. If an abort prevents the successful completion of a full append, you can restart the append at a previous checkpoint location by specifying the desired checkpoint number when reopening the file.

checkpoint, StorHouse/RM

A restart point. When loading from MVS, the client data loader takes a checkpoint after writing a certain number of megabytes to the temporary VRAM™ file. When loading from MVS or with FTP, the server data loader takes a checkpoint for each data extent when it reaches the maximum size.

checkpoint file, StorHouse

A file containing a snapshot of StorHouse system files at a given point in time.

checkpoint dataset or file, StorHouse/RM

The dataset or file containing the checkpoint value that's used to recover from an aborted or failed run.

checkpoint recovery, StorHouse

The recovery of directory information for level L (and level S) files and other system file information from special checkpoint files that contain a snapshot of system files at the time the most recent CHECKPOINT command was issued.

clause, StorHouse/RM

A component of an SQL statement that enables you to select and manipulate a subset of information in your database. Clauses begin with a keyword followed by arguments. *See also* keyword and argument.

cleaning cartridge (tape)

A special cartridge used to clean drive heads in a tape library device.

client data loader, MVS

The SGI MVS Data Loader utility component that resides on your host. The client data loader prepares your data for loading and writes the data stream to a temporary file on StorHouse. See also SGI MVS Data Loader utility. Contrast server data loader, MVS.

client FTP tool

The FTP program on your host computer that sends server-level FTP commands and transfers files to the StorHouse FTP server during data loading and unloading.

client-server

A type of network architecture. A client is an agent, typically an operating system or software that enables authorized users to access services and associated resources on the network. A server provides a specific service to users. Examples of servers are archive servers, file servers, print servers, and fax servers.

CLOB

A type of character data up to 2 GB in size. *See also* large object.

closed cursor

A cursor that is no longer associated with the active set of rows returned by a query. A closed cursor, however, remains associated with the SELECT statement. *Contrast* open cursor.

coalesce operation

See merge operation.

collection

A group of user files stored in a single file. A collection residing on local or network storage accessible to StorHouse/RFS is called a local collection. A collection written to StorHouse is called a StorHouse collection. A collection also consists of corresponding file locator data. For a local collection, file locator data is stored in a separate file in a collection directory. For a StorHouse collection, file locator data is stored in a table array and within the StorHouse collection. *See also* local collection and StorHouse collection.

collection definition

A group of parameters that define collection options. Each collection set has a collection definition. You create collection definitions with StorHouse/CCi.

collection directory

A location where StorHouse/RFS accumulates file locator data for a local collection.

collection set

A group of StorHouse collections. For example, if an e-mail collection is written to StorHouse once a day, then after a month, the e-mail collection set would consist of 30 or 31 StorHouse collections, each consisting of data and associated file locator data.

collector

The StorHouse/RFS server component that checks a collection directory for files to collect and accumulates those files into collections. You define a collector by creating a collector definition. You can define multiple collectors. *See also* collector definition.

collector definition

A profile that defines each collector, including where a collector looks for files to collect. You create collector definitions in a StorHouse/RFS profile using StorHouse/CCi.

column

An attribute or characteristic of a table. A data field in an input data record that you load into a column of a user table. *Contrast* field. *See also* data field.

column definition

The column name, data type, and any default value or LOB storage options assigned to a column on a CREATE TABLE statement. See also table definition.

Command Language

See StorHouse Command Language.

command privileges

The privileges that determine which Command Language commands and parameters and Callable Interface functions an account can use.

commit

To accept and make permanent the changes that a transaction made to a database. *Contrast* roll back.

comparison operator

A component of an SQL statement that enables you to perform comparisons on values. StorHouse SQL supports six comparison operators: = (equal), <> (not equal), < (less than), > (greater than), <= (less than or equal to), and >= (greater than or equal to).

complex predicate

A type of predicate (often containing boolean (logical) operators) that relates one or more predicates and produces a single true, false, or unknown value. *See also* boolean operator.

compliant media

In relation to StorHouse, media such as WORM optical, WORM tape, or disk that incorporates intrinsic control of file (or object) retention (for example, EMC Centera and SnapLock from NetApp). *Compare* write once, read many (WORM).

compound index

An index constructed on multiple columns of a table. Also called *composite index* or *concatenated index*.

concurrency

The shared use of resources by more than one application process at the same time.

confirm, StorHouse/RM

To end a load. You must explicitly confirm a load when loading with FTP. The SGI

MVS Data Loader utility automatically confirms a successful load. Confirming a load updates the metadata and removes all restart information stored in checkpoint files.

connection string

A fully qualified database name specified on a CONNECT statement to indicate a remote database. Also called *connect string*.

contiguous allocation

A user-assigned file set storage allocation attribute that instructs StorHouse to allocate storage on one entire volume side or entire volume sides with any remaining allocation placed on one volume side. *Contrast* noncontiguous allocation.

continuation field, StorHouse/RM

A string of characters, hexadecimal digits, or blanks you can identify to combine physical records into a logical record. When loading data, you can specify a continuation field with the CONTINUEIF clause of the LOAD DATA statement.

control file, StorHouse/RM

When loading data with the SGI FTP Data Loader, the file on your client computer that contains control statements like LOAD DATA, LOAD INDEX, MERGE, or StorHouse SQL statements. A control file with a LOAD DATA statement can also contain data records.

control panel

The set of indicator lights and switches (for example, READY, SLOT FULL, DOOR LOCKED, INPUT ERROR, and RESET) located on the front of a StorHouse library device. The operator uses the control panel to monitor the status of library device activity.

control statements

When loading data with the SGI MVS Data Loader utility, the control statements (LOAD and SMDEF) you create to define load parameters and specify StorHouse information. For the SGI FTP Data Loader, control statements are the LOAD DATA, LOAD INDEX, MERGE, and any SQL statements.

copy phase, load

The transfer of a load data stream to a temporary VRAM file on StorHouse. *Compare* load phase.

copyin utility, StorHouse/RM

The metadata conversion program that updates metadata for a new StorHouse/RM release. *Compare* copyout utility.

copyout utility, StorHouse/RM

The metadata conversion program that makes a copy of the current metadata and performs the required conversions. *Compare* copyin utility.

creator, StorHouse/RM

The StorHouse account ID of the user who creates a database user component. The creator owns the component unless he or she assigns ownership to another account ID. See also owner.

current connection

The database where you are currently executing SQL.

current row

The most recently retrieved row of the active set and the row to which the cursor is pointing.

cursor

A named control structure used by an application program to point to the current row in the active set and retrieve rows from that set.

cycle time

The time period, specified in number of days, that controls which side of a volume StorHouse uses to allocate file space. When the specified number of days has elapsed since the last (most recent) file space was allocated on side A, StorHouse deactivates side A and activates side B. See also deactivation time.

damaged file

A file that is missing one or more extents; that is, a truncated or partial file.

DASD

See direct access storage device.

data conversion, StorHouse/RM

The process that changes (if necessary) the data type from one representation to another, for instance, an input data type to a database data type when loading data, or a database data type to a host language data type when submitting a query with ESQL, or a database data type to an unloader data type when unloading data.

data definition language (DDL)

A category of SQL. DDL statements create and drop database components, alter user tablespaces, and grant and revoke privileges.

data delimiter record, StorHouse/RM

A record in a load data stream that separates the data records from other record types. Data records begin immediately after the data delimiter record. *See also* data stream.

data dictionary

See system table.

data extent

A type of file extent that contains data records. *See also* extent.

data feeder

A program on your host that supplies the input data for a load.

data field

A column or field in an input data record. *See also* column and field.

data loader

A program that loads data into StorHouse user tables and creates index entries for user table indexes. See SGI MVS Data Loader utility or SGI FTP Data Loader.

data read/write device (DRD)

A device that reads and writes data on removable media. An optical disk drive is an example of a DRD.

data source

A combination of a database system, the operating system it uses, and any network software required to access it.

data manipulation language (DML)

A category of SQL. DML statements query data and insert, update, and delete rows in system tables.

data source

A server or a database in a federated system.

data stream, StorHouse/RM

Information packaged during a load. Also called load stream. A data stream consists of four record types: environment definition, LOAD DATA and other SQL statements, data delimiter record, and user data.

data type

A classification that denotes the characteristics and length of data values in a column (for example, CHAR(5)). *See also* database data type, host language data type, loader data type, unloader data type, and native data type.

data value

See cell.

database administrator, StorHouse

The person responsible for security, operation, and tuning of a StorHouse database. *Compare* system administrator.

Database Administrator (DBA) privilege

The highest database privilege that can be assigned to a StorHouse account. An account with DBA privilege can create database user components, access and maintain system tables, grant and revoke privileges, and select data from any table in a specific database.

database component privilege

A StorHouse privilege that determines an account's access to specific database components. Database component privileges include ALL, DELETE, INDEX, INSERT, SELECT, and UPDATE.

database container directory

The directory in the UNIX file system that contains all StorHouse database directories. *See also* system tablespace.

database data type

A StorHouse data type used in a column definition in a CREATE TABLE statement.

database default user tablespace

The user tablespace assigned to a user table when a StorHouse account does not specify a user tablespace on the CREATE TABLE statement and does not have an account default user tablespace. The database default user tablespace is the one assigned to PUBLIC in the SYSSMUSERS system table. See also user tablespace and account default user tablespace.

database directory

See system tablespace.

database gateway

A type of software that translates an application program's database requests from one relational database system protocol to another. Each gateway has a server component and a client component. StorHouse supports the ODBC gateway.

database name

The identifier that names a StorHouse database. A database name must start with a letter, cannot exceed 32 characters, is case sensitive, and can consist of any combination of letters (a–z or A–Z), numbers (0–9), and underscore (_).

database privilege

A privilege that controls the functions a StorHouse account can perform in a specific StorHouse database. Database privileges include DBA, RESOURCE, and

SCAN. See also Database Administrator (DBA) privilege, RESOURCE privilege, and SCAN privilege.

database system components

The structures that StorHouse/RM uses to manage a database. These components include system tables, system table indexes, system table logs, system table index logs, and the system tablespace. *Contrast* database user components.

database user components

The structures that a StorHouse account works with to store, retrieve, and manage data. These components include user tables, columns, views, synonyms, indexes, and user tablespaces. *Contrast* database system components.

database system files

The UNIX files that contain system tables, system table indexes, system table logs, and system table index logs. *Contrast* database user files.

database user files

The StorHouse files that contain table data, index entries, and LOB data. *Contrast* database system files.

date literal

A constant that specifies a month (MM), day (DD), and year (YYYY) string in a particular format.

DDL

See data definition language.

DEACTIVATED state

A state indicating that additional storage allocations are not allowed on the volume side. StorHouse can read files on a deactivated volume side.

deactivated volume

A volume with both sides classified as deactivated.

deactivation time

The time period, specified in number of days, that controls when StorHouse deactivates a volume side. StorHouse deactivates the volume side when the specified number of days has elapsed since the first file space allocation.

decimal literal

A constant that specifies a decimal number as a signed or unsigned number with a maximum of 31 digits that may include a decimal point.

declarative statement

An SQL statement that does not generate any calls to StorHouse/RM routines.

Declare Section

A required ESQL program component where you declare host language and indicator variables.

dedicated device

A disk or tape drive that supports only one file read or write operation at a time.

default definition

A default value to be loaded into a column if no value is supplied. You specify a default definition on the CREATE TABLE statement.

default environment

The default values associated with a StorHouse account and activated at StorHouse signon. Defaults include access group name, volume set name, file set name, and access rights.

default subspace

The lowest-numbered subspace that allows a component type. If you do not explicitly select subspaces or request rotation during a load, the SGI data loader selects the default subspace.

default user tablespace

See account default user tablespace or database default user tablespace.

deferred index

An index created after a table is loaded. You use a SGI data loader to create index entries for existing segments. *See also* index load operation.

deferred dynamic system parameter

A StorHouse system parameter that the system administrator can change during system operation. The changes do not take effect until the system is restarted. CHKP_ACCOUNT is an example of a deferred dynamic system parameter. *See also* dynamic system parameter and static system parameter.

delete

To mark StorHouse files as deleted in the primary, backup, or archive directories. You must mark files as deleted before you can remove them from StorHouse. You can "undelete" files that have not yet been removed. The Command Language DELETE command enables you to mark files as deleted. *See also* remove.

delete directory

The directory that contains information about files that are marked for deletion. The delete directory holds an entry for a file marked for deletion until you remove the file.

deleted space

The bytes used by files that have been deleted and removed from a file set or from a file set partition.

delimited data, StorHouse/RM

Data records with delimiters. You can load delimited data. When unloading data, you can delimit the result data. See *also* enclosed data and terminated data.

delimited SQL identifier

An alias, cursor, or database component name enclosed in double quotes (") because it contains a space or special character, is case sensitive, starts with a character other than a letter, or is an SQL reserved word. See also SQL identifier.

delimiter

A marker for separating data fields in an input record or a result record. A termination delimiter follows a data field, and an enclosure delimiter surrounds a data field.

delta time

A format for expressing time in Command Language commands. Delta time indicates the time before or after an event. The format for delta time is: Ddays-hours:minutes:seconds.

descriptor flags

A subset of information recorded by StorHouse for a file version. Descriptor flags indicate whether the file version:

- Was archived
- Was backed up
- Is being cataloged
- Is named
- Has a copy/transfer pending
- Has NOBACKUP attribute value
- Is hardware disabled
- Is software disabled

The Command Language SHOW FILE /FULL command enables you to display the values of descriptor flags.

destination file set

The file set you specify when transferring a file to StorHouse. Also called *primary file set*. StorHouse uses your account's default file set if you do not specify one. The destination file set contains the primary copy of the file.

destination volume set

The volume set you specify when transferring a file to StorHouse. Also called *primary volume set*. StorHouse uses your account's default volume set if you do not specify one. The destination volume set contains the file set with the primary copy of the file.

detail data

The most granular level of information that an enterprise can collect at a single point of service. Examples of detail data are call detail records, clickstream data, e-mail messages, and point-of-sale information.

device

A component of StorHouse identified by a device identification code. Devices consist of the following:

- Units – magnetic disk drives, library devices, and shelf storage
- Subunits – accessors, drives, exchange station, and slots.

device identification code (did)

A code that identifies a device. The format of the did is:

{level} {unit number} {subunit type} {subunit number}

An example is L00D01, where L is the level (library device), 00 is the unit number (first library device), D is the subunit type (optical disk or tape drive), and 01 is the subunit number (the second optical disk or tape drive).

device scan recovery

A procedure that recovers and then catalogs directory information for StorHouse files, extents, volume sets, and file sets on a specific level F device.

DF extent

A type of file extent that contains information necessary to retrieve data. For StorHouse/RM segment files, there is one DF extent for each data extent. *See also* extent.

did

See device identification code.

direct access storage device (DASD)

A device that provides immediate access to stored data.

direct connectivity/direct connect

A means of directly attaching a host to StorHouse using bus and tag cables. *Contrast* network connectivity.

directory

A catalog that identifies, locates, and maintains statistics for files stored in StorHouse. StorHouse has four directories: primary, backup, archive, and delete.

directory extraction

The process of copying information from the StorHouse PRIMARY, BACKUP, and ARCHIVE directories to directory extraction files. The EXTRACT DIRECTORY command extracts StorHouse directory information from the directories and stores the information in directory extraction files. This information can be used to restore StorHouse directories should the need arise. *See also* directory extraction files.

directory extraction files

The files that are created by the EXTRACT DIRECTORY command. One extraction control file and multiple extraction VTOC files are created each time the command is issued. See extraction control file and extraction VTOC file.

directory recovery

A procedure for recovering one or more directory files. Directory recovery is used in the unlikely event that normal, or shadowed, recovery cannot restore corrupted directory files.

- For removable volumes (levels L and S), you can recover directory information through checkpoint recovery, directory recovery from extraction files, or directory recovery from physical volumes.
- For level F Centera volumes, you can recover directory information through device scan recovery.

See checkpoint recovery, directory recovery from extraction files, directory recovery from physical volumes, and device scan recovery.

directory recovery from extraction files

A procedure that uses special files containing information extracted periodically from the directories of one or more StorHouse systems to recover the directories of those systems.

directory recovery from physical volumes

A procedure that uses the volume table of contents (VTOCs) on removable volumes in the system to recover StorHouse directory information.

disabled file version

A file version that StorHouse cannot read. StorHouse will not attempt to read a disabled file version until you issue the Command Language ENABLE command.

DISABLED state

A state indicating that a volume side is missing, physically damaged, or has excessive I/O errors.

disabled volume

A volume with at least one side classified as disabled.

discard file, StorHouse/RM

A StorHouse VRAM file that contains records discarded during a load operation. *See also* discarded record.

discarded record, StorHouse/RM

A logical record that did not meet any condition (WHEN clause) specified for a data load operation. StorHouse/RM collects discarded records in a discard file on StorHouse. See also discard file.

Disk File Transfer Interface

The SGI StorHouse interface that provides file transfer functions for user files on host storage devices.

DML

See data manipulation language.

DRD

See data read/write device.

drive

See optical disk drive or tape drive.

duplex

A feature that allows StorHouse to read the backup or archive (duplex) copy of a file extent when the primary copy is unavailable or for load balancing. *See also* load balancing.

durable

A state in which all operations in a transaction are permanent (in other words, committed). *See also* commit.

dynamic SQL statement

An SQL statement that is prepared and executed at runtime. The SQL statement can change several times during the execution of the application program. *Contrast* static SQL statement.

dynamic system parameter

A StorHouse system parameter that the system administrator can change during system operation. Changes to dynamic system parameters take effect immediately. The system changes some dynamic parameters automatically to reflect the current system status. Access Time Factor (ATF) is an example of a dynamic system parameter. *See also* deferred dynamic system parameter and static system parameter.

E-notation

A floating-point literal expressed as two numbers separated by E, for example, 10E132.

EDC

See Error Detection Code.

embedded SQL statement

See static SQL statement.

empty volume

An initialized volume that does not contain file data or file labels. *Compare* blank volume.

enabled volume

A volume with all sides enabled, or available for use.

enclosed data

Data fields preceded and followed by an enclosure delimiter. *See also* delimiter.

engine

See StorHouse engine.

entry/exit slot

See exchange station.

environment definition

The load data stream record that contains nvk (affects interpretation of binary multibyte numerics), ccsid (character data), record type, and db_ref (affects interpretation of loader control information). The StorHouse FTP server and the SGI MVS Data Loader utility generate this record based on parameters you provide for a load as well as internally supplied data.

environment variable

A UNIX parameter that defines an aspect of your working environment, such as your home directory or frequently used directory path.

equi-join

A type of join that uses predicates specifying equalities to join a column from one table with a column from another table.

erasable

Rewritable media. *Contrast* write-once-read-many.

Error Detection Code (EDC)

The method used to detect data transfer, software, and system file errors. The host generates and stores error detection codes at the end of each data frame prior to transferring that data frame to StorHouse. The host checks the error detection codes of each data frame during transfer from StorHouse.

ESCON

Enterprise System Connection. IBM's fiber optic-based channel connection technology. When connecting to an MVS®/ESA™ host, StorHouse can be defined as an ESCON channel-to-channel adapter (SCTC).

ESQL

Embedded Structured Query Language. An interface that enables you to embed SQL in a C or C++ program.

ESQL construct

An SQL statement prefixed by EXEC SQL and terminated by a semi-colon (;). The prefix, statement, and delimiter constitute the construct.

ESQL precompiler

The precompiler used to build database applications in the StorHouse SQL development environment.

exabyte

A unit of measurement equivalent to approximately 1,000,000 terabytes or 1,000 petabytes.

exchange station

The entry and exit point for volumes in a library device. Also called *mail slot* and *entry/exit slot.* The operator loads and unloads removable optical or tape volumes at the exchange station.

exclusive lock

A lock that reserves a table for updating only. Also called *write lock*. This lock prohibits a table from being shared. One engine can have an exclusive lock on a table. All other lock requests (shared and exclusive) for the table are queued. *Contrast* shared lock

executable statement

An SQL statement that generates StorHouse calls and return codes. Executable statements execute instructions on a specified database at runtime.

EXECUTE STH_LOAD command

The StorHouse Command Language command that starts the loading process in StorHouse. The SGI MVS Data Loader utility and the SGI FTP Data Loader generate and submit this command for you.

execution manager

The StorHouse/RM process that runs the constructed execution tree built by the optimizer.

execution tree

An internal representation of a query in a form that an engine can process. The StorHouse/RM explain facility enables you to create a representation of the execution tree in the form of relational tables. You can query these tables to view the execution strategies selected by the optimizer. *See also* explain facility and node.

expiration time

The number of days that must elapse after the last (most recent) file space was allocated on a volume side before StorHouse marks the side as deactivated and expired.

EXPIRED state

A state indicating that a volume side is expired and the files on that side are ready to be removed from the system. Additional storage allocations are not allowed on an expired volume side. However, StorHouse can read data on an expired volume side.

expired volume

A volume with both sides classified as expired.

explain facility

The StorHouse/RM feature that enables you to examine the strategy, or execution plan, implemented by the StorHouse/RM optimizer for a given query. The explain facility displays the execution plan in a set of explain tables. *See also* explain tables.

explain tables

A set of relational tables, located in a system tablespace, containing the execution plan for a query. You use StorHouse SQL statements—CREATE EXPLAIN TABLES, EXPLAIN PLAN, DROP EXPLAIN TABLES—to manage the explain tables. *See also* explain facility.

explicit lock

A lock you place on a file version to prevent other account users from reading, writing, and deleting the file version until you unlock it. An explicit lock locks a file version in the specified directory (primary, archive, or backup). *Contrast* implicit lock.

export

To remove an uncataloged volume, volumes, or volume set from a StorHouse system. StorHouse does not manage exported volumes and volume sets. The Command Language EXPORT command enables your system administrator to export uncataloged volumes and volume sets. *Contrast* import.

expression

An operand or a collection of operands and operators that yields a single value.

extended recovery

The restoration of a StorHouse system to normal operation in the unlikely event of a failure. For example, the loss of directory information on magnetic disk would require extended recovery if both the primary and shadowed copies are corrupted. *See also* directory recovery.

extension

An extra option or feature supported by StorHouse SQL in addition to those features specified in the ANSI SQL standard. See also Structured Query Language (SQL).

extent

A collection of file data that StorHouse treats as a unit. Each extent has an extent sequence number and must fit on a single volume side. StorHouse supports the following extent types:

- Data – contains user data records. There is one data section per revision or checkpoint.

- Definitions – contains information that defines a revision of a file version and maps relative record numbers to specific extents containing user data records. For VRAM files, there is one definitions section per revision or checkpoint. For STORHOUSE files, there is one definitions extent per file.

- Change – contains updated user data records for one or more file data sections. A change section contains only those records updated for a single revision.

- Key Data Base – contains access key indexes that map key values to relative record numbers.

- Map - contains the high-level index for STORHOUSE hash and value index files. There is one map extent for each hash index file and one map extent for each value index file.

- Update – contains all updated user data records for a data section for all revisions of a file.

extent sequence number

The identification number assigned to each file extent when that extent is added to a file version. StorHouse assigns sequence number zero to the first extent of a file version and increments the sequence number by one for each extent that is added.

extent set

In append operations (mode=APPEND), a collection of extents consisting of a data extent, a DF extent, and for KEYED files, a K extent. Each extent set of a file must be no larger than 2 GB or a single volume side, whichever is smaller.

external key

A named data field, located in a special key record that is associated with a separate data record. You use external keys to locate and read specific data records in a KEYED file. *Contrast* internal key.

extraction, StorHouse/RM

A type of query that returns all rows for one or more columns in a user table. This type of query results in a full table scan or a full segment select. *See also* full table scan and full segment select and *contrast* select.

extraction control file, StorHouse

A file created by the EXTRACT DIRECTORY command that identifies groups, volume sets, file sets, volumes, and all extraction VTOC files associated with a StorHouse directory extraction. Each directory extraction creates one extraction control file.

extraction VTOC file, StorHouse

A file created by the EXTRACT DIRECTORY command for each volume side for which file information is extracted. The information on an extraction VTOC file is similar to the information in a volume table of contents (VTOC) on a volume. Each directory extraction can generate multiple extraction VTOC files.

Extractor, StorHouse/RM

A StorHouse feature that provides fast-path processing for some queries that result in full table scans and full segment selects.

federated system

A database management system (DBMS) that supports applications and users who submit SQL statements that reference two or more DBMSs or databases in a single statement (for example, a join between tables in different databases). The IBM DB2 UDB product with StorHouse as a data source is an example of a federated system.

federator

The software component in a federated system that coordinates SQL processing. For instance, a federator accepts and parses a query from a client, breaks the query into smaller queries, submits the queries to the appropriate data sources, receives the results from the data sources, assembles the results into a single result set, and returns the result set to the client. DB2 UDB is both a federator and a database system.

fid

See file identifier

Field, StorHouse/RM

A named portion of a record to be used in a condition, for instance, to indicate which records to load. Fields are not loaded into user tables. *Contrast* column. *See also* data field.

field terminator, StorHouse/RM

A delimiter that follows a data field in an input record or a result record.

file

The named collection of logically related data. StorHouse manages files in the performance buffer, in library devices, and on shelves. Different file versions with the same file name are considered different files. *See also* file version.

file extent

See extent.

file version, StorHouse/RFS

A file that has been collected with the same file name and path as a previously archived file. StorHouse/RFS creates file versions only for files without retention requirements. StorHouse/RFS adds a negative numeric suffix (-1, -2, and so on) to a file version name in the virtual directory, for example status.txt(-1).

file locator data, StorHouse/RFS

Information about each file in a collection. StorHouse/RFS obtains file locator data for each file collected, stores the data in the StorHouse collection and in a StorHouse table, and uses the data to locate files.

file access group

A set of named files. Each file in a StorHouse system is a member of one file access group. A named file is uniquely identified by a file access group name, a file name, and a version number.

file access group name

A 1- to 8-character name assigned to a file access group. The file access group name is unique within a StorHouse system.

file extent

See extent.

file identifier (fid)

A unique identifier assigned to each file when it is created. A file identifier consists of two numbers: a system identifier (sid) and a file number (fno). All files created in StorHouse have the same system identifier. The file number makes the file identifier unique within a StorHouse system.

File Information Display Utility (LSMFI)

Three StorHouse Callable Interface functions that retrieve information about StorHouse user files. You can use the information to produce quantitative reports of file usage and to determine which files you can export or take offline. The information retrieved by the File Information Display Utility is similar to the information displayed by the Command Language SHOW FILE command.

file label

An identifier containing StorHouse directory information for a file and pointers to the file's data extent on a volume.

file LIMIT

The attribute that sets the number of file versions that StorHouse will retain. The maximum number of primary file versions StorHouse can retain is 32,768.

file migration

The removal of files from the performance buffer or the movement of files from one volume set to another. StorHouse automatically migrates file extents off the performance buffer based on extent access history, extent size, and Access Time Factor (ATF). Your system administrator can manually migrate files from one volume set to another by issuing the StorHouse Command Language MIGRATE /BY_VSET command.

file name

A 1- to 56-character name assigned to a file. File names are unique within a file access group.

file number (fno)

One of two numbers that make up a file identifier (fid). Because all files created in a single StorHouse system have the same system identifier, the file number makes the file identifier unique within that system.

file organization

The way StorHouse stores records in a file. File organization can be SEQUENTIAL, RECORD, KEYED, KEYSEQUENTIAL, and STORHOUSE.

file reference variable

A host variable used for transferring a LOB value from StorHouse to a client file.

file retention attribute

File characteristic that determines the retention period of a file. *See* retention period.

file revision

The result of a file version when you create, update, and append to it. You update the contents of a file version by opening it; changing, deleting, or adding records; and closing it. Thus, a file version can have multiple revisions. The maximum number of file revisions that a file version can have is 65,535. *See also* file version, revision number, and version number.

file set

A collection of files within a volume set. Files are stored on file sets.

file set LIMIT

The attribute that sets the maximum size of a file set. The LIMIT must be greater than or equal to the total size (in other words, the storage capacity) of all volumes or the volume set, and it must be an integer multiple of the physical volume's maximum size. StorHouse will not extend the size of a file set beyond its LIMIT.

file set name

A 1- to 8-character name that identifies a file set. A file set name is unique within a volume set.

file set partition

See partition.

file set retention attribute

File set characteristic that determines the retention period for files in the file set. If a file has an unspecified retention attribute at create time or /FORCE_RETENTION is in effect on the file set, the file inherits its retention attribute from its file set.

file system, StorHouse

The StorHouse/SM software that manages StorHouse files.

file system, UNIX

The UNIX operating system software that manages StorHouse/RM metadata and range indexes.

file version

A new copy of a file that you can revise and manage independently of the original version. A file version is identified by a file access group name, file name, and relative version number. You create a new file version each time you transfer a file from a host to StorHouse. StorHouse can maintain a maximum of 32,768 versions of a file for one file name in a given file access group. *See also* file revision, revision number, and version number.

file version, StorHouse/RFS

A file that has been collected with the same file name and path as a previously archived file. StorHouse/RFS creates file versions only for files without retention requirements. StorHouse/RFS adds a negative numeric suffix (-1, -2, and so on) to a file version name in the virtual directory, for example, status.txt(-1).

fixed content, object-based disk

Media such as EMC Centera™ Content Addressed Storage. StorHouse supports this media type in the level F user disk layer in the storage hierarchy.

fixed-length record

A data record that has the same length as other data records in a data file. The last byte in the data file must exactly finish a fixed-length record. *Contrast* variable-length record and text record.

floating-point literal

A constant that specifies a floating-point number as two numbers separated by an E (for example, 15E1 or 2.3E5). Both numbers can include a sign, but only the first number can include a decimal point. The maximum number of digits allowed in the first number is 17. The maximum number of digits allowed in the second number is 2.

format string

The mask values that indicate the format StorHouse/RM should use to return date and time values.

fno

See file number.

frame

A unit of data or control information that can be transferred between processors and storage devices without reformatting. A frame must be contained entirely within a single file extent.

free pool indicator

The second, third, and fourth characters of a level F free pool volume set name.

free pool volume set

The volume set of empty volumes (per recording type) in a library device or level F device. Volumes in the free pool volume set are used for future allocation to volume sets. *See also* empty volume.

front end communications manager

The StorHouse/RM client software that arranges (marshals) prepared SQL in communication packets and unmarshals result set data.

FSET

See file set.

FTP

File Transfer Protocol. An established standard for transferring files between two computers connected by a network. SGI provides the SGI FTP Data Loader for bulk data loading through FTP.

FTP command, StorHouse/RM

An instruction you provide to run the SGI FTP Data Loader and Unloader. You supply user-level FTP commands to your client FTP tool. Your client FTP tool sends server-level FTP commands to the StorHouse FTP server. For instance, put is a user-level FTP command and STOR is the corresponding server-level FTP command.

FTP session

When loading or unloading, the period of time or interaction that begins when you log into the StorHouse FTP server and ends when you log off. During an FTP session, you can run one or multiple loads and unloads.

full segment select

A type of query that returns one or more entire segments with the use of a range index. A full segment select may qualify for extractor processing.

full table scan

To read all rows in a table without the use of an index. A query that results in a full table scan may qualify for extractor processing. A StorHouse account requires the SCAN database privilege to perform a full table scan.

function

A named operation in an SQL statement, followed by one or more expressions. StorHouse supports two types of functions: aggregate and scalar. *See also* expression, aggregate function, and scalar function.

gateway

See database gateway.

GB

See gigabyte.

general space

The storage space that StorHouse allocates to SEQUENTIAL files and to the sections of VRAM files that are not associated with updates.

gigabyte (GB)

A unit of measurement equivalent to approximately 1,000 megabytes.

grantee

The account ID that is granted a database or database component privilege. *Contrast* grantor.

grantor

The account ID that grants a database or database component privilege to another account ID. *Contrast* grantee.

gripper

See accessor.

group

See file access group.

hash index

An index that uses a proprietary SGI algorithm to locate individual table rows based on individual index values. StorHouse typically uses hash indexes to search for a single value or for a small range of specific values.

hash index file

A STORHOUSE-type file that contains entries for a hash index. *See also* STORHOUSE file type.

hexadecimal literal

A string of one or more hexits, enclosed in quotation marks and prefaced by an uppercase or lowercase X. Hexadecimal literals represent binary data. You can use hexadecimal literals (for example, X '01020ABC') in StorHouse SQL WHERE clauses and SELECT lists. See *also* hexit.

hexadecimal character (hexit)

A member of the character set A-F, 0-9, and a-f. These are the valid characters for writing base-16 (hexadecimal) values.

HOLD attribute

A volume or volume set attribute that helps determine the order that volumes will be migrated from a library device to shelf storage. Valid HOLD attribute values are HOLD or NOHOLD. For any library device, StorHouse migrates volumes with a HOLD value of NOHOLD before migrating volumes with a HOLD value of HOLD.

host

The computer that contains your application programs.

host data file

See data file or input dataset.

host interface types

The software interfaces that enable a host to access StorHouse; for example, Callable, Interactive, and Disk File Transfer.

host language data type

A data type that is native to the host language. For example, long is a C language data type.

host language variable

An application variable that can be referenced by host language statements and embedded SQL statements. *See also* input host variable and output host variable.

hybrid IN join method

A type of join method that the optimizer may use when a query is an equi-join and the inner table has a value index on the join column or for inner-join and outer-join operations. *Compare* nested loop join method. *See also* join method.

identifier

See SQL identifier.

implicit lock

A temporary lock that StorHouse places on a file during update, append, copy, or various directory operations. StorHouse unlocks the file after the operation has completed. *Contrast* explicit lock.

import

To add a previously uncataloged volume or volume set to a StorHouse system. The Command Language IMPORT command enables your system administrator to import volumes and volume sets. *Contrast* export.

inconsistent database

The state of a database when a single atomic operation that updates one or more system tables fails before completing and committing all updates.

index

A database component used to locate data stored in tables. StorHouse supports three types of indexes for user tables: hash, range, and value. *See also* hash index, range index, and value index.

index file

A StorHouse file that contains index entries for user tables. There's one index file for each hash index and each value index for each segment in a user table. *See also* hash index file, value index file, and STORHOUSE file type.

index ID

The numeric identifier assigned to each index in a database. Index IDs are unique within a database.

index load operation

A type of load that creates index entries of deferred indexes for existing segments. You run an index load with a SGI data loader. *See also* deferred index.

indicator variable

An optional variable used to detect null or truncated values in an application program.

inner-join

A type of join that combines matching rows from specified tables. The unmatched rows are omitted from the result table. *Contrast* outer-join.

INITIALIZED state

A state indicating that a volume label record has been written on the volume side.

initialized volume

A volume that contains a volume label record (on both sides of a two-sided volume). *See also* volume label.

in-line LOB

A LOB data value stored in a row with the other table data. You can store a LOB data value in-line when the value and the row do not exceed 32 KB. Contrast out-of-line LOB.

input data file, StorHouse/RM

A file that contains the data records you're loading into a StorHouse user table. This file can reside on your client computer or on StorHouse. LOB values can also reside in separate LOB data files. Also called input dataset. Contrast result file. See also LOB data file.

input data record, StorHouse/RM

A record in an input dataset or data file. An input data record typically corresponds to a row in the user table. *Contrast* result data record.

input dataset, StorHouse/RM

The file that contains the input data records you are loading into StorHouse user tables. This file can reside on your host or on StorHouse. Also called *data file* or *input data file*.

input host variable

A host variable used to pass data to your program. Input host variables are generally specified in the WHERE clause of a SELECT statement. Contrast output host variable.

input SQLDA

A storage area that holds information about input host variables in a dynamic SQL statement. Also known as bind descriptor. Contrast output SQLDA.

integer literal

A constant that specifies a binary integer as a signed or unsigned number with a maximum of 10 significant digits and no decimal point. The range for integer literals is +2147483647 to -2147483648.

Interactive Interface

The StorHouse host interface that enables you to enter Command Language commands and to receive StorHouse responses.

interface

The hardware or software that links systems, programs, and devices.

interior node

A node, in an execution tree, that has both a parent and one or more children. *See also* node, leaf node, root node, and execution tree.

internal key

A named data field, located within a data record, that you use to locate and read specific data records in a KEYED file. *Contrast* external key.

invalidated segment

A segment that has been replaced or made obsolete through a SGI data loader merge, or coalesce, operation.

isolation directory

A directory called RFS_ISOLATED containing user files that fail a verification check.

ISQL

Interactive SQL. StorHouse/CCi provides an ISQL tool for submitting SQL statements to StorHouse.

join

A database operation (SELECT) that combines rows from one or more tables or views. StorHouse supports outer-joins, inner-joins, equi-joins and self-joins. *See also* outer-join, inner-join, equi-join, and self-join.

join method

The series of steps that the optimizer performs to join tables. StorHouse/RM supports two join methods: nested loop and hybrid IN. See also nested loop join method and hybrid IN join method.

journal chain

Multiple journal files—current, cycled, and archived—that are linked together and replayed when a database is restored. The journal replay utility can replay an entire journal chain or a partial chain. *See also* redo journal.

journal file

See redo journal.

journal file, StorHouse/RFS

A StorHouse/RFS file that contains all information about file activity from file creation until the collection containing the file is ready to be written to StorHouse.

jukebox

See library device.

key

A named data field used to access records in KEYED and KEYSEQUENTIAL files.

KEYED file organization

A RECORD file organization that enables you to access records by key values. You can define up to 31 keys for a KEYED file. See also RECORD file organization.

Keyed Record Access (KRA)

The StorHouse Virtual Record Access Manager (VRAM) software that enables you to access records by key values. KRA supports KEYED and KEYSEQUENTIAL files. Compare Relative Record Access (RRA).

KEYSEQUENTIAL file organization

A KEYED file organization with the following restrictions:

- You can define only one key.
- You cannot change the key value when updating a record.
- You cannot enter duplicate key values.
- You must write the records to the file in ascending key value order.
- StorHouse does not create a key database extent for the file.

See also KEYED file organization.

keyword, StorHouse/RM

An SQL statement component with a predefined meaning that is reserved for special tasks (for example, WHERE). Arguments complete or modify the meaning of the keyword. See also argument.

KRA

See Keyed Record Access.

LAN

See local area network.

large object (LOB)

A sequence of bytes with a length up to 2 GB. A LOB may be a binary large object (BLOB) or a character large object (CLOB). *See also* BLOB and CLOB.

layer

A hierarchy of storage within a storage level. For example, storage level L (library device) can contain a layer of write-once-read-many (WORM) media, a layer of erasable optical media, and a layer of tape media.

LD

See library device.

leading blanks

One or more blank characters at the beginning of a data field. A SGI data loader does not trim blanks that are part of a data field that’s enclosed by enclosure delimiters. It may trim leading blanks from a character-type data field when optional enclosure delimiters are not present.

leaf node

A node, in an execution tree, that represents operations on base tables. A leaf node has a parent, but no children. *See also* node, interior node, root node, and execution tree.

left outer-join

A type of join that returns all of the matched rows of the tables, and for unmatched rows, returns all rows from the outer or left table with null values for the inner or right table. *See also* outer-join. *Contrast* inner-join.

level

See storage level.

library device (LD)

A StorHouse device that uses robotics to transfer optical disk or tape cartridges between the drives, slots, and exchange station. A single StorHouse system can support multiple library devices. A library device contains the following components:

- Two or more *drives*
- Multiple *slots* for storing cartridges
- A transport mechanism, or *robotic arm*, that moves cartridges between the slots, drives, and exchange station. The part of the robotic arm that holds a cartridge is called an *accessor*. Library devices have one or two accessors.

Tape library devices may also have *bar code readers* mounted on the robotic arm. Bar code readers automatically read bar code labels on tape volumes.

- An *exchange station* for loading and unloading cartridges

- A microprocessor-based *controller* that manages the robotic arm and drives.

library device attribute

The volume set attribute that indicates the device identification code of the library device to use when a free pool volume is required. When a volume set requires more storage space for files, StorHouse takes a volume from the specified device's free pool and allocates it to the volume set. Some library devices support multiple recording types and have a free pool for each recording type.

LIMIT attribute

See file LIMIT, file set LIMIT, or volume set LIMIT.

literal

A string or numeric constant that denotes a specific value. Literals can be classified as string, numeric (exact and approximate), and binary. *See also* character literal, date literal, decimal literal, floating-point literal, hexadecimal literal, integer literal, time literal, and timestamp literal.

load balancing

The act of distributing processing more evenly across libraries to improve system performance. StorHouse practices load balancing when the duplex feature is enabled and primary and backup or archive (duplex) file copies reside in different online library devices. The system administrator can select the degree of similarity between the primary and duplex media to be used in load balancing operations. *See also* duplex.

LOAD DATA statement, StorHouse/RM

An SQL-like statement that describes the data to be loaded into one or more user tables in a StorHouse database. *Contrast* UNLOAD statement.

load file, StorHouse/RFS

A file containing file locator data for a local collection.

load ID, StorHouse/RM

An identifier that uniquely names each load. A load ID is case sensitive and can consist of up to 40 alphanumeric characters.

load phase, StorHouse/RM

The portion of a load when the server data loader builds any indexes and writes segments on StorHouse. *Compare* copy phase, load.

load stream

See data stream.

loader data type

A data type that describes the data fields (columns) being loaded into a database. You provide this data type in a LOAD DATA statement.

LOB

See large object.

LOB data file

A file on a client computer or a remote system that contains one LOB data value to be loaded into a user table on StorHouse.

LOB locator

See locator variable.

LOB record

A BLOB or CLOB data field that consists of one or more physical records in the input data file. A LOB record starts with a 64-bit length field followed by the data. LOB records are automatically considered part of the logical record, that is, no CONCATENATE or CONTINUEIF clause is needed and, if present, does not apply to these records.

LOB result file

A file on a client computer or a remote system that contains one LOB data value unloaded from a StorHouse user table.

LOB subsegment file

A StorHouse file that contains out-of-line LOB values. Multiple LOB columns in a user table can share a LOB subsegment file. *See also* out-of-line LOB and STORHOUSE file type.

LOB subsegment ID

A numeric identifier for each LOB subsegment file in a segment.

local area network (LAN)

A single physical communications link that connects multiple workstations, enabling them to share hardware, software, and data.

local collection

A group of user files stored in one rename directory and related file locator data stored in a collection directory. StorHouse/RFS deletes a local collection when the user-defined maximum collection space is exceeded, but never before successfully writing the collection to StorHouse. *See also* collection. *Contrast* StorHouse collection.

local database

A database that resides on your host system. *Contrast* remote database.

local search, StorHouse/RFS

A search for a file in a staging area or a rename directory. StorHouse/RFS starts a StorHouse search if it cannot locate the file locally. *Contrast* StorHouse search.

local table, StorHouse/RFS

An in-memory table containing the file locator data that StorHouse/RFS checks during a local search. *Contrast* StorHouse table.

locator variable

A host variable that represents a single LOB value and enables an application to manipulate that value on StorHouse with various SQL functions.

logical operator

An operator that relates one or more true/false values and produces a single true/false value. StorHouse supports three logical operators: AND, NOT, and OR. Also called *boolean operator*.

logical record in input data, StorHouse/RM

A record that is assembled from one or more physical records. Logical records can contain both fields and columns. *Contrast* physical record in input data.

logical volume

The portion of a physical volume (such as a volume side) that can be accessed while it is mounted in a drive and is treated as a mountable unit of storage.

LSMFI

See File Information Display Utility.

MAGDISK

The name of the volume set that contains the StorHouse performance buffer.

magnetic disk drives

The drives located in the magnetic disk unit chassis in the StorHouse processor cabinet. Magnetic disk drives write and read data on fixed, magnetic disk volumes.

magnetic disk unit (MDU) chassis

The chassis that holds the magnetic disk drives in the StorHouse processor cabinet.

manual migration

The movement of file from their resident file set to the performance buffer for faster access. Files are queued for upward migration using the StorHouse Command Language STAGE command. *See* upward migration.

metadata file, StorHouse/RFS

A StorHouse/RFS file that contains data about collected files, including file locator information and changes to previously collected files.

MAID (Massive Array of Idle Disks)

A storage system comprising a very large array of disk drives where a majority of the drives are powered off. MAID is designed to reduce energy consumption, maintain performance, and increase storage density.

mail slot

See exchange station.

map extent

A type of file extent that contains the high-level index that StorHouse reads first when doing index lookups. Hash index files and value index files consist of data, DF, and map extents. *See also* extent.

merge operation, StorHouse/RM

A type of load that consolidates existing segments (based on selection criteria) into one segment or a smaller number of segments. Also called *coalesce operation*. You use a SGI data loader to merge segments.

metadata

The data that StorHouse/RM uses to manage a database. *See also* system table and system table index.

metadata backup file, StorHouse/RM

A file created by the metadata backup utility. A metadata backup file contains a copy of system tables, system table indexes, system table logs, and system table index logs. If necessary, you can use metadata backup files to restore metadata.

metadata backup utility, StorHouse/RM

The software that backs up the system components in a database directory. This utility creates metadata backup files.

metadata conversion utility, StorHouse/RM

The software that copies and reformats existing metadata so that a new StorHouse/RM software release can use it. *See also* copyout utility and copyin utility.

metadata entry

A row in a system table.

metadata recovery

See automatic metadata recovery process or metadata restore utility.

metadata restore utility

The software that recovers one or more database directories with metadata backup files. This utility locks the metadata while copying the metadata backup files to the database directories.

MDU

See magnetic disk unit chassis.

media life span

The estimated time period that media can be used. Also called *shelf life*. For example, erasable optical media has an estimated life span in excess of 20 years.

media type

The first two characters of the StorHouse volume identification code. Media type identifies the type of drive on which the volume can be formatted and processed (for example, MA for magnetic disk).

medium

A type of storage volume, such a tape cartridge.

megabyte

A unit of measurement equivalent to approximately 1,000,000 bytes.

memo

Volume or volume set information that an authorized user can include in the StorHouse system files. For example, a user can note a volume's contents or location by using the /MEMO parameter modifier in the MOVE VOLUME, MOVE VSET, SET VOLUME, or SET VSET commands. The SHOW VOLUME and SHOW VSET commands display the /MEMO comment.

MF

See migration factor.

migration

The process of moving data from one storage level to another or between library devices within storage level L. For example, data can be migrated from storage level F (magnetic disk) to storage level L (library device). *See also* file migration and volume migration.

migration factor (MF)

A 32-bit integer that is derived from a file's Access Time Factor (ATF), size, and access history. StorHouse maintains a migration factor for each extent in the performance buffer. Extents with the lowest migration factor values are migrated off the performance buffer first.

mirror system, StorHouse/RFS

A secondary StorHouse system with the same user data, and file locator data as the primary StorHouse system. StorHouse/RFS accesses data on the mirror StorHouse system should the primary system become unavailable.

mode

See access mode.

modifier

The part of a Command Language command line that limits or enhances the scope of a command or parameter. You specify a modifier after the command or parameter it modifies, and you precede the modifier with a slash (/). Using MONITOR /ALL as an example, MONITOR is the Command Language command and /ALL is the command modifier.

mount count

The number of times a removable volume has been used (mounted and dismounted).

mount limit

The maximum number of mounts, recommended by the media manufacturer and initially set at installation, for each media type and recording type combination configured in the system.

.netrc file

A file (on the home directory of your client computer) that contains macros for automating FTP command entry.

native data type, StorHouse/RM

A binary data type. StorHouse supports the following native data types: BINARY, BYTE, DECIMAL, DOUBLE, FLOAT, INTEGER, NUMERIC, RAW, SMALLINT, VARBINARY, VARBYTE, and VARCHAR. When loading data with the SGI FTP Data Loader, different hosts interpret binary data differently, so you may have to identify a native values key on the FTP put command. See also native values key.

native file format backup, StorHouse

The process of backing up files in their native format, which eliminates the need to restore files from tape to disk prior to access.

native values key

A value that identifies the client hardware type used to interpret byte order, floating point representation, and INT length of binary data. You can specify a native values key when loading data with the SGI FTP Data Loader.

NEEDS_CLEANING state

A state indicating that the volume side should be cleaned.

nested loop join method

A join method that the optimizer may use when the number of qualifying rows in the outer table is small, when the predicate is not an equals-type, or when the join column of the inner table does not have an index. *Compare* hybrid IN join method. *See also* join method.

network connectivity

A means of attaching multiple hosts or workstations to the StorHouse system through a network. *Contrast* direct connectivity/direct connect.

nickname

In a federated system, the representation in a DB2 catalog for a remote table or view controlled by a particular server. You can create nicknames for StorHouse tables and views through the DB2 CREATE NICKNAME statement.

node, StorHouse/RM

A discreet step in the execution of the query. You can display the nodes in an execution plan by using the StorHouse/RM explain facility. A node may be a root node, an interior node, or a leaf node. *See also* explain facility, root node, interior node, and leaf node.

noncontiguous allocation

A user-assigned file set storage allocation attribute that instructs StorHouse to allocate the first available free space to the file set to obtain the desired size. This space may be on one or more volumes. *Contrast* contiguous allocation.

NOT NULL

A column constraint that indicates you must provide a value for the column during a load operation for a user table or during an insert or update operation for a system table.

null value

A value that indicates unknown, not applicable, or missing information.

object identifier (OID)

The entry in a table row that identifies an out-of-line LOB. The OID contains the LOB length (in bytes), the LOB subsegment ID, and the starting frame number and offset within the LOB subsegment file. StorHouse/RM uses OIDs to access LOB values in LOB subsegment files. See also out-of-line LOB.

ODBC

Microsoft Open Database Connectivity.

ODLD

Optical Disk Library Device. *See* library device.

offline volume

A volume that you manage manually because StorHouse catalogs no longer contain information about that volume. An exported volume is an example of an offline volume. *Contrast* online volume.

OID

See object identifier.

online volume

A volume managed by StorHouse and listed in StorHouse catalogs. Online volumes reside in storage levels F (magnetic), L (library device), and S (shelf). Contrast offline volume.

open cursor

A cursor that is associated with the result set returned by a query. *Contrast* closed cursor.

operator (SQL)

A component of an SQL statement that enables you to manipulate table data by performing such tasks as comparisons and arithmetic operations on column values. There are four main types of operators: arithmetic, comparison, logical, and set. *See also* arithmetic operator, comparison operator, logical operator, and set operator.

optical disk cartridge

A type of removable optical media used to store data. Also called *platter* and *volume*. Optical disk cartridges contain an optically sensitive metallic layer between two glass or plastic disks.

optical disk drive

A subunit of a library device that reads and writes data on removable optical disk cartridges. Each library device has at least two optical disk drives.

optical disk volume

See optical disk cartridge.

optimizer

A component of the StorHouse engine that receives analyzed SQL statements from the parser and develops the most efficient plans to execute the statements. *See also* parser.

out-of-line LOB

A large binary or character object that's stored in a LOB subsegment file. *Contrast* in-line LOB.

outer-join

A type of join that returns all of the matched rows of the tables, and for unmatched rows, returns all rows from one table with null values for the other table. StorHouse supports left outer-joins. *See also* left outer-join. *Contrast* inner-join.

output host variable

A host variable that passes data and status information to a program. You specify output host variables on the INTO clause of a SELECT or FETCH statement and on the VALUES INTO statement. C*ontrast* input host variable.

output SQLDA

A storage area that holds information about the output host variables in a dynamic SQL statement. Also known as *select descriptor*. *Contrast* input SQLDA.

overhead space

The unusable space in a StorHouse volume set. Overhead space is not allocated to file sets.

owner

A StorHouse account ID that creates or is assigned ownership of a database component. *See also* creator.

package

A set of stored SQL statements that have been bound statically and are available for processing.

page manager

The StorHouse process that translates retrieval requests into calls to StorHouse/SM data retrieval services.

parameter

See system parameter.

parser

A component of the StorHouse engine that analyzes an SQL request for syntax and semantics, and then breaks it into basic elements for the optimizer. *See also* optimizer.

partial file

A file version that is missing one or more extents before the last extent.

partition

The part of a file set on a specific volume side.

passthru

In a federated system, a mechanism by which DB2 allows a client to issue queries and other SQL statements directly to a data source. With the

StorHouse/UDB Link, StorHouse is the data source.

password

A unique set of alphanumeric characters used to protect systems and files from unauthorized use. You enter passwords to sign on to StorHouse, use an account, access a file access group, and access a file.

performance buffer

The portion of level F storage (magnetic disk) that is used as a staging area for performance copies of files waiting to be backed up to their primary file sets. The performance buffer is a file set named $$BUFFER in the volume set named MAGDISK. The performance buffer is used as cache. Compare and contrast staging and caching.

performance copy

A file or file extent that is stored temporarily in the performance buffer.

PERM_LOCKED state

A state indicating that the volume side is permanently writelocked due to the condition of the media. StorHouse can read data on a perm_locked volume side.

perm_locked volume

A volume with both sides classified as perm_locked.

petabyte

A unit of measurement equivalent to approximately 1,000,000 gigabytes or 1,000 terabytes.

physical record in input data

One input data record (or line) in an input dataset or data file. *Contrast* logical record in input data.

physical volume

A unit of media where data can be recorded and read. *See* volume. *Contrast* logical volume and virtual volume.

picker

See accessor.

PIT instance

A StorHouse/RFS server that presents a specific point-in-time historical view of StorHouse data. You create a PIT instance using StorHouse/CCi. *See also* StorHouse point-in-time recovery.

platter

See optical disk cartridge.

precision

The total number of digits in a number defined as a DECIMAL or NUMERIC data type. *Contrast* scale.

predicate

An element of a search condition that implies or expresses a comparison operation. Predicates reduce the number of rows returned by a query. StorHouse supports three types of predicates: basic, complex, and quantified. *See also* basic predicate, complex predicate, and quantified predicate.

prepared statement

An SQL statement that has been parsed for syntax errors and assigned an identifier. You can execute a prepared statement as often as necessary within the same transaction.

primary copy

The file copy that is available to an account user for normal access. Primary file copies are stored on primary file sets.

primary directory

The directory that contains information about primary copies of files.

primary file set

The file set that contains the primary copy of a file. Also called destination file set. Primary file sets are located on primary volume sets.

primary volume set

The volume set that contains primary file sets. Also called destination volume set.

privileges

The authority to perform a certain type of function or to bypass a certain type of security in the system. Accounts can have access, command, database, and database component privileges. *See also* access privilege, command privilege, database privilege, and database component privilege.

processor cabinet

The StorHouse hardware component that typically contains the StorHouse processor, a modem, power switches, and the magnetic disk unit chassis and/or RAID units. The master console terminal typically sits on the processor cabinet.

project

A table operation that selects all rows for one or more columns in a table. *Contrast* select.

PUBLIC

All accounts that can access StorHouse. Granting database component privileges to PUBLIC is the same as granting them to all accounts that can access StorHouse.

pushdown

In a federated system, the act of moving processing into a particular data source.

quantified predicate

A type of predicate that compares a value to a collection of values preceded by the keyword ALL, ANY, or SOME.

RAID (Redundant Array of Independent Disks)

A category of disk drives that employ two or more drives in combination for fault tolerance and performance. RAID storage can be used in the level F layer of the StorHouse storage hierarchy.

range index

A type of StorHouse index that contains the highest and lowest values contained in each segment. Range indexes are stored in a set of system tables.

read lock

See shared locked.

read-only optical device

An optical drive that StorHouse places in a read-only state due to a failure. A system operator can manually place an optical drive into a read-only state using the SET DEVICE command.

record, StorHouse

A unit of user data or file control information.

record, StorHouse/RM

The storage representation of a row or other data. Also called *row*.

RECORD file organization

A file organization where records are stored in the order they are written and where updated records are stored separately from the initial record. You can read and update records in sequential or random order, and you can append new records to the end of a RECORD file. You cannot insert records between existing records.

record segment

The part of a record contained in a single frame. Each record segment has a header that defines the segment.

record terminator

A delimiter that marks the end of an input record or a result record.

recording type

The part of the volume identification code that identifies the mode of recording used to format a volume. Examples:

- A – single-sided, single-density, sector reusable (magnetic)
- B – double-sided, single-density, non-erasable (optical)
- C – single-sided, double-density, sector reusable (magnetic)

- D – double-sided, quadruple-density, erasable (optical).

Recovery, StorHouse/RM

See automatic metadata recovery process or metadata restore utility.

recovery log file

A file that lists any system table logs and system table index logs containing undo records. A recovery process checks this log during metadata recovery.

redo journal, StorHouse/RM

A file that contains data used to re-create metadata. The following committed transactions are captured in a redo journal: INSERT ROW, DELETE ROW, UPDATE ROW, DROP INDEX, DROP TABLE, CREATE INDEX, and CREATE TABLE. You use redo journaling utilities to manage the redo journal. Also called *journal file*. *See also* redo journaling utilities and journal chain.

redo journaling utilities

A set of database recovery utilities for capturing, storing, and restoring transactions that affect metadata. The utilities enable you to cycle a redo journal (close the current one and start a new one), archive a redo journal to StorHouse, audit a redo journal, purge a redo journal from disk, and replay a redo journal to restore changes. *See also* redo journal.

Relative Record Access (RRA)

The StorHouse Virtual Record Access Manager (VRAM) software that enables you to access records by relative record number. RRA supports RECORD files. *Compare* Keyed Record Access.

relative record number

A number assigned to records in RECORD files. StorHouse assigns relative record number 1 to the first record written, 2 to the second record written, and so on. StorHouse assigns the same relative record number to the update record.

relative revision number

See revision number.

relative version number

See version number.

remote database

A database that resides on a system other than your host system. For example, a StorHouse database is considered a remote database in a DB2 application running on an MVS host. Contrast local database.

removable medium or volume

A volume that can be dismounted from its drive. Examples of removable volumes are optical disk cartridges and magnetic tapes.

remove

To remove files from the StorHouse delete directory. The Command Language REMOVE FILE command enables your system administrator to remove files that are marked for deletion. You cannot access files that have been removed from the delete directory. *See also* delete.

rename directory

A directory that contains renamed user files as a local collection. StorHouse/RFS creates rename directories automatically. The files in one rename directory and their associated file locator data in a collection directory compose a local collection. StorHouse/RFS removes local collection data from a rename directory when the maximum collection space value in the StorHouse/RFS configuration file is exceeded, but never before successfully writing the collection to StorHouse.

replaced segment

A segment that has been invalidated. Users cannot access files in replaced segments.

replay checkpoint

A point (or journal file) in a journal chain where the journal replay utility stops restoring transactions. With replay checkpointing, you perform a partial replay, restoring only those transactions up to a checkpoint. *See also* journal chain and redo journaling utilities.

replicated file

A file that has at least one copy on another StorHouse system, including on the performance buffer.

replication class

A named set of information about the destination StorHouse system that will contain replicas of files on a source StorHouse system. A replication class consists of the following components: replication class name, disabled flag, network device information, network system name and link name used to connect to the destination StorHouse, and the names of the volume set and file set that will contain file replicas.

reserved word

See SQL reserved word.

residence, file version

The location (volume set and file set) of a file version in a StorHouse system.

residence, volume

The location (storage level F, L, or S) of a volume in a StorHouse system.

residence, volume set

The location of all member volumes in a volume set. Removable volumes can be on different storage levels and different devices; thus, a volume set can reside on more than one storage level and device at the same time.

RESOURCE privilege

A StorHouse database privilege that enables an account to create tables, synonyms, views, and indexes and to select information from those tables, synonyms, and views.

restrictive clause

The clause that limits the number of rows returned by a query. For example, WHERE and HAVING are restrictive clauses on a SELECT statement.

result data record

A record in a result file. *Contrast* input data record.

result file

A file on a client computer that contains data unloaded from a user table on StorHouse. *Contrast* input data file.

result set

The set of rows returned from a SELECT statement. Also called *answer set* and *result table*.

result table

See result set.

retained file

A file that has a non-zero retention period that has not expired.

retention period

The time span that a file may not be deleted from StorHouse. A file retention period can be a specified number of days, ZERO to indicate no retention, or FOREVER, to indicate that StorHouse will not allow the file to be deleted. The maximum retention period is 65,000 days, or approximately 178 years.

RETIRE state

A state indicating that the volume side should no longer be used in the system due to the condition of the media. Additional storage allocations are not allowed on a retired volume side. StorHouse may not be able to read data on a retired volume side.

retired volume

A volume with both sides classified in the RETIRE state.

retriever

The StorHouse/RFS server component that performs local searches and StorHouse searches when a user or an application searches for files or requests to open a file.

return code

A four-digit numeric value associated with a symbolic name for a StorHouse message. The return code indicates the success or failure of a specific function.

revision number

The number assigned to a file revision. The system assigns revision number 1 to a file *version* when it is created in StorHouse and increments the number by one each time the contents of the file version are changed. Revision numbers can be relative (0 through -65,534) or absolute (1 through 65,535). The most recent file revision has relative revision number 0, and the previous file revisions have relative revision numbers -1, -2, -3, up to -65,534. *See also* file revision, file version, and version number.

rfsmaint utility

A StorHouse/RFS utility used for compacting and deleting StorHouse collections.

rfsrestore utility

A StorHouse/RFS utility used in conjunction with other StorHouse/RM utilities for recovering file locator tables in a StorHouse database.

robotic arm

The transport mechanism in a library device. The robotic arm moves optical disk or tape cartridges among the slots, drives, and exchange station. The part of the robotic arm that holds the cartridge is called the *accessor*. A robotic arm can have one or two accessors depending on the type and model of the library device. Some tape library devices have *bar code readers* mounted on the robotic arm.

roll back

To reverse changes made by a transaction to a database. *Contrast* commit.

root node

The entry point into an execution tree. *See also* node, interior node, leaf node, and execution tree.

row

A collection of columns. Also called *record*.

RRA

See Relative Record Access.

safety directory, StorHouse/RFS

A directory that contains duplicate entries of user file changes (such as file deletes, renames, and security changes). You specify the directory path in a StorHouse/RFS profile. StorHouse/RFS deletes the data in the safety directory after successfully writing the collection to StorHouse.

scalar function

An SQL operation that produces a single value from another value. For example, the scalar function ABS computes the absolute value of a specified expression. *See also* function and aggregate function.

scale

The number of digits to the right of the decimal point in a number defined as a DECIMAL or NUMERIC data type. *Contrast* precision.

scan

See full table scan.

SCAN privilege

A StorHouse database privilege that enables an account to read all rows in any user table (in other words, perform a full table scan) on which it has SELECT privilege. An account with SQLADMIN or DBA privilege can grant SCAN privilege to other accounts.

security table, StorHouse/RFS

A table in a StorHouse database that contains security information for Windows systems that use extended security, such as security descriptors.

segment

A set of StorHouse database user files. Each user table consists of one or more segments. Each segment consists of a table data file, one index file for each hash index and each value index, and one or more LOB subsegment files. Segments consist of extents. *See also* extent, table data file, index file, and LOB subsegment file.

segment delete utility

The software that identifies invalidated segments, removes the segment files (table data file, index files, LOB subsegments), and removes the associated entries in system tables. Use this facility to mass-remove invalidated segments and their associated objects from a StorHouse database.

segment file

A StorHouse file that contains table data, hash index entries, value index entries, or LOB data. Each segment consists of a set of segment files.

segment ID

A numeric identifier for each segment in a user table.

segment tag

A name assigned to a segment with the SEGMENT clause on a LOAD DATA or MERGE statement. The default segment tag is the load ID.

select

A table operation that selects all columns from one or more rows in a table. *Contrast* project. Also known as a *selection*, which contrasts with *extraction*.

select descriptor

See output SQLDA.

select list

Column names and other expressions specified in the SELECT clause.

self-join

A type of join that joins a table with itself. Also called *auto-join*.

SEQUENTIAL file organization

A file organization where records are stored in the order they are written. You cannot randomly access records in a SEQUENTIAL file; you must access records from the beginning of the file to the end.

serializable

The ANSI/ISO transaction isolation level that guarantees the highest read consistency and data integrity in a database. StorHouse supports this isolation level.

server data loader, MVS

The SGI MVS Data Loader utility software component that loads data into StorHouse user tables. The server data loader runs on StorHouse. *See also* SGI

MVS Data Loader utility. *Contrast* client data loader, MVS.

session

The period of time or the interaction that begins when you sign on to StorHouse and ends when you sign off from StorHouse.

set

To change attributes, parameters, passwords, and so on for the system, accounts, files, file sets, file access groups, volumes, volume sets, and user IDs. The Command Language SET commands enable you to change these items (if you have the proper privilege).

set operator

An operator you use to combine separate SQL queries in different ways. StorHouse supports the UNION and UNION ALL set operators.

severity code

A code that identifies the severity level of operator messages. StorHouse severity codes include: A (Action), E (Error), F (Fatal error), I (Information), S (Success), and W (Warning).

SGI Data Loader

A SGI utility that allows you to bulk load data from your host computer (the client) into StorHouse/RM user tables on the StorHouse server. SGI offers two data loaders: one for MVS and one for FTP.

SGI FTP Data Loader

The SGI software that bulk loads data from a UNIX or other FTP-enabled host computer into user tables on StorHouse.

SGI FTP Data Unloader

The SGI software that unloads data from StorHouse to a sequential file on your client, or to a VRAM file on StorHouse, or to another program. You can also unload LOB data to separate LOB files on your client computer or on a remote host. You use your standard client FTP software to unload data.

SGI MVS Data Loader utility

The SGI software that bulk loads data from an MVS host computer into user tables on StorHouse.

SGI OpticalVSAM™

The SGI software component that provides VSAM-based MVS applications transparent access to data stored on StorHouse.

SGI OVSAM™

See SGI OpticalVSAM.

shadowing

The process of maintaining a duplicate copy of critical system files, such as directory files, on an alternate magnetic disk or RAID.

shared device

A drive that can support multiple file reads and one write concurrently.

shared lock

A lock that reserves a table for reading only. Also called *read lock*. Multiple engines can have a shared lock on the same table. This lock prevents a table from being dropped. *Contrast* exclusive lock.

shelf life

See media life span.

shelf storage

The managed storage of optical or tape volumes located on shelves or in storage outside of the StorHouse system. Shelf storage is storage level S.

short record

An input data record that is missing a data field described in a field specification.

shortword

A 2-byte integer (SMALLINT) preceding each variable-length data record. This shortword contains the length of each data record. The shortword is not included in the length of the data record.

sid

See system identifier.

side

See volume side.

sign off

To end a StorHouse session and disconnect from the StorHouse system.

sign on

To connect to the StorHouse system and start a StorHouse session.

simple query

A type of query processed by the extractor. A simple query must conform to a specific set of rules and results in a full table scan.

single-column index

An index created on one column in a table.

site identifier

A 1- to 14-character code assigned to a site during installation. The system parameter SITE_ID contains the value of the site identifier.

slot

The area within a library device where volumes are stored or "parked" when not in use. Also called *bin*.

SNA interface

The client and server hardware and software necessary to communicate using LU6.2 with the DRDA gateway.

soft drop

A feature that enables you to recover from an errant table drop. When you drop a user table, you can later undrop or purge it.

software disabled

The state of a VRAM file version when you open it in APPEND or UPDATE mode. After you successfully close the file version, StorHouse marks it as software enabled. StorHouse leaves an open file in the software disabled state whenever:

- You close the file with the abort flag set.
- The application program terminates.
- The network disconnects.
- The host crashes.
- StorHouse crashes.
- StorHouse forces the file to close as a result of an internal error.

space

See whitespace.

special register

A storage area for specific information that can be referenced in SQL statements. StorHouse supports three special registers: USER, SYSDATE, and SYSTIME.

SQL-style comment

Comment text that starts with two hyphens (--) and terminates by the end of the line. You use SQL-style comments in static SQL statements. *Contrast* C-style comment.

SQL identifier

A basic syntactical unit that provides a name for a database component, alias, or cursor. An SQL identifier must start with a letter and can be followed by a–z, A–Z, 0–9, or an underscore (_). Identifiers in StorHouse SQL can have a maximum length of 32 characters. *See also* delimited SQL identifier.

SQL reserved word

A keyword that has been predefined to SQL to initiate specific processing tasks within the SQL environment. You should not use reserved words as database component names, but if you do, you must delimit the names.

SQL statement

A StorHouse SQL instruction used to manage and access a StorHouse database.

SQL statement manager

The StorHouse/RM software that tracks SQL statements through all processing steps.

SQL status code

A numeric code that StorHouse/RM generates in response to StorHouse SQL statements.

SQLADMIN privilege

A StorHouse access privilege that gives an account DBA privilege in all StorHouse databases.

SQLCA

SQL Communications Area. A storage area that holds information about the execution of an application's most recently executed SQL statement.

SQLCOMMAND privilege

A StorHouse command privilege that enables an account to load data (invoke a SGI data loader to submit the StorHouse Command Language EXECUTE STH_LOAD command).

SQLDA

SQL Descriptor Area. A storage area that holds information about the input or output host variables in a dynamic SQL statement. *See also* input SQLDA and output SQLDA.

SQLEXECUTE privilege

A StorHouse command privilege that enables an account to submit SQL statements.

stage

The act of queuing a transfer of a specified file (extent) to the performance buffer.

staging area

The location where StorHouse/RFS writes physical files for a user or application during the collection process. A staging area consists of a staging directory and a user directory. You assign a collector to a staging directory and user directory. A collector removes files from the staging area and adds them to a local collection. *See also* user directory and staging directory.

staging attribute, StorHouse

The file attribute that indicates whether you can stage (copy) a file from Level L storage to the performance buffer for faster access. The Command Language CREATE FSET and SET FSET commands enable you to change this attribute for files.

staging directory, StorHouse/RFS

A directory in a staging area where StorHouse/RFS looks to collect directory-level security information. A staging directory may have one or more user directories. A user or application must have permission to access a staging directory. *See also* user directory and staging area.

static SQL statement

An SQL statement that is embedded in an application program, prefixed by the keywords EXEC SQL, and terminated by a semi-colon. Once prepared, a static SQL statement does not change. *Contrast* dynamic SQL statement.

static system parameter

A system parameter that cannot be changed while the system is operating. LOG_MAX (which indicates the number of user logs to maintain) is an example of a static system parameter. *See also* deferred dynamic system parameter and dynamic system parameter.

status code

See SQL status code.

STH

The default StorHouse file access group for segment files.

sthname

The name of the StorHouse system to which you connect when you load or unload StorHouse data.

storage definition, StorHouse/RFS

A profile that defines where file locator data is stored on StorHouse. The StorHouse/RFS configuration file contains storage definitions.

storage level

One of three types of storage managed by StorHouse:

- Level F – fixed storage of unremovable volumes (magnetic disk)
- Level L – library storage with robotic access to removable volumes (library device)
- Level S – shelf storage of removable volumes (shelves).

StorHouse®

An intelligent storage virtualization and data management platform that can archive, retrieve, and back up massive amounts of relational and file-based information using an automatically managed pool of traditional and alternative storage devices. Virtualized devices from different vendors can include high-performance disk, commodity SATA disk, and highly efficient tape in automated libraries.

StorHouse provides an intelligent storage virtualization and data management layer that complements existing IT and storage infrastructures. It enables organizations to leverage existing investments in storage technology by matching storage resources to storage requirements. With StorHouse, administrators can transparently introduce new storage, retire old storage, and migrate data between diverse storage devices – all while maintaining 100% uptime and accessibility. *See also* StorHouse/RM, StorHouse/SM, StorHouse/CCi, and StorHouse/RFS.

StorHouse API

The StorHouse software that enables user applications to access StorHouse, perform file functions, and transfer data.

StorHouse collection

A group of user files stored as a single file on StorHouse. *Contrast* local collection.

StorHouse Command Language

The set of commands that make up the StorHouse Interactive Interface. You issue Command Language commands to communicate with, monitor, and control StorHouse. For example, you use Command Language to create, modify, delete, and show information for files; transfer files between a host and StorHouse; migrate files between StorHouse storage levels; back up files; and so on.

StorHouse communications manager

The StorHouse software that manages socket-level communications between StorHouse client and server systems, including ESCON channel support.

StorHouse database, StorHouse/RFS

A relational database that contains StorHouse tables used for storing file locator data, collection metadata, security entries, aliases, and statistics for StorHouse/RFS.

StorHouse engine, StorHouse/RM

The main StorHouse/RM process. A StorHouse engine parses, optimizes, and executes SQL statements. During a load, a StorHouse engine executes non-LOAD statements in a control file, obtains necessary metadata for a load, verifies StorHouse privileges for loading, and updates metadata at the end of a load.

STORHOUSE file organization

A file organization created and used exclusively by StorHouse/RM to support relational database capabilities on the StorHouse system. STORHOUSE files have standard frame and record structures, but users cannot access STORHOUSE files directly. Users must submit Structured Query Language (SQL) statements to StorHouse/RM to load and access database data. The StorHouse/RM software determines the contents of STORHOUSE files and file records.

StorHouse for EMC Centera

A full-featured StorHouse system that always uses Centera as the user disk layer in the StorHouse storage hierarchy.

StorHouse FTP server

The SGI software component that acts as a valid FTP server. It accepts FTP commands from your client FTP tool, generates and transfers data streams to StorHouse, and replies to your FTP commands.

StorHouse index

A database component that contains indexing data for a StorHouse table.

StorHouse point-in-time recovery

A StorHouse feature that enables StorHouse/RFS administrators to configure and allow end-user access to views of a StorHouse storage environment in the exact state that existed on a specified date and time. Using these point-in-time system views, end-users can access historical files through the StorHouse/RFS system the same way they access current files.

StorHouse search, StorHouse/RFS

A search that queries a StorHouse table and accesses data in StorHouse collections. *Contrast* local search.

StorHouse table, StorHouse/RFS

The StorHouse database component containing file locator data and collection metadata for each collected file, as well as statistics, aliases, and security entries. You create StorHouse tables with the StorHouse/RFS utility called tblgen. (StorHouse/RFS automatically creates file locator tables after you create a collections table with the tblgen utility.) *Contrast* local table. *See also* table array.

StorHouse subsystem

A standard IBM MVS subsystem that transfers data, control information, and status between your application programs and StorHouse. The StorHouse subsystem must be active before you can request StorHouse services from the subsystem's host computer.

StorHouse/CCI

A web-based system and database administration interface for managing StorHouse systems. StorHouse/CCi provides all the features administrators need to control, track, and report on StorHouse systems regardless of their physical location.

StorHouse/ODBC driver

The SGI software that enables a Microsoft Windows or UNIX tool or application that supports the ODBC call library to use a StorHouse database as a data source.

StorHouse/Relational File System (RFS)

The StorHouse file system interface. StorHouse/RFS enables applications to store and access a virtually unlimited number of files on StorHouse. Users and/or applications request files archived through StorHouse/RFS using standard file I/O methods supported by their computer's operating system (for example, NFS, NTFS, and CIFS).

StorHouse/RFS audit log

A log that contains records written for a given file whenever a specific system event occurs. Administrators can configure the StorHouse/RFS software to generate the audit log as a text file or in database format. For the text file format, StorHouse/RFS writes a local log only and starts a new one every day. For the database format, StorHouse/RFS initially writes log records to a text file and subsequently bulk loads them into a user table in the audit log database at a user-specified interval.

StorHouse/RFS configuration file

The file that contains the operating parameters for a StorHouse/RFS server. You can maintain this file using the StorHouse/RFS profile feature in StorHouse/CCi.

StorHouse/RFS duplexing

A feature where StorHouse/RFS writes the same data to two StorHouse systems for disaster protection and data availability.

StorHouse/RFS file system driver

The interface between the StorHouse/RFS server and the user application or operating system. This driver provides the virtual file system that enables users or applications to read and write files through StorHouse/RFS.

StorHouse/RFS profile

A StorHouse/CCi tool for managing the properties, or operating parameters, of a

StorHouse/RFS server. In other words, it is the graphical representation of the sections and definitions in the StorHouse/RFS configuration file. After profile creation, SGI deploys, or assigns, the profile to a StorHouse/RFS server according to your site specifications. Once deployed, a profile resides on your StorHouse/RFS server in a text file format.

StorHouse/RFS server

The set of programs—StorHouse/RFS file system driver, StorHouse/RFS collectors, and StorHouse/RFS retriever— that make up StorHouse/RFS.

StorHouse/RM

The SGI RDBMS component that works in conjunction with StorHouse/SM to specifically administer the storage, access, and movement of relational data. StorHouse/RM provides row-level SQL access to high volumes of detail data on any layer in the StorHouse storage hierarchy, including tape. SQL access is available from different platforms through a variety of industry-standard protocols. *See also* StorHouse and StorHouse/SM.

StorHouse/RM API

The client software that passes prepared SQL statements to the front end communications manager.

StorHouse/SM

The StorHouse storage management software component for controlling a hierarchy of storage devices, including cache, RAID, ATA disk, MAID, erasable and write-once-read-many (WORM) optical disk jukeboxes, and high-performance and high-capacity erasable and WORM tape in automated libraries. StorHouse/SM is also responsible for automating critical system management tasks like data migration, backup, and recovery.

StorHouse/UDB Link

The SGI wrapper software that implements the connection between the DB2 UDB and StorHouse databases. This software gives DB2 users near- transparent access to data on StorHouse.

Structured Query Language (SQL)

A standardized language for defining and manipulating data in a relational database. StorHouse SQL is compatible with the ANSI SQL-99 standard and includes extensions to support additional capabilities.

subsegment ID

See LOB subsegment ID.

subquery

A SELECT statement within the WHERE or HAVING clause of another SQL statement.

subspace

A part of a user tablespace that defines storage parameters for a specific component type—table data, hash index, value index, LOB data—or for all component types. *See also* default subspace.

subspace number

A number you assign to each subspace when you create a user tablespace. Subspace numbers range from 0 to 2,147,483,647.

subunit

A device within a unit. For example, a library device is a unit and the accessors, drives, exchange station, and slots are subunits within the library device.

surface

See volume side.

surface partition

See partition.

surface set

The file set partitions that reside on the same volume side.

symbolic name

An alphabetic code in a StorHouse message. Symbolic names are listed and explained in the StorHouse *Messages and Codes Manual*.

synonym

An alternate name you can assign to a user table, view, or synonym for easy identification.

SYSADM account

A StorHouse account with the privileges to administer all StorHouse databases. SYSADM has all StorHouse/SM command privileges (ALLPRIVILEGE) and all StorHouse/RM database privileges—DBA, RESOURCE, and SCAN—in all databases.

syscreate utility

The program that creates StorHouse databases.

system administrator

The person responsible for the security, operation, and tuning of the StorHouse system.

system definition, StorHouse/RFS

A profile that defines where a StorHouse collection is stored. A system definition contains the StorHouse server name, DNS name, StorHouse file access group name, StorHouse storage resource names, as well as the account and encrypted password used to log in to StorHouse to write collections. You create system definitions in a StorHouse/RFS profile using StorHouse/CCi.

system file

A StorHouse internal file that resides on magnetic disk and contains programs, directories, account data, or other data used to control the system.

system identifier (sid)

A unique 1- to 6-character code assigned by SGI to each StorHouse system. The system parameter SYSTEM_ID contains the value of the system identifier. The system identifier is part of the file identifier. All files within a StorHouse system have the same system identifier.

system parameter

The information that influences how StorHouse manages its resources. Your system administrator uses system parameters in Command Language command lines to set specified system-wide defaults. *See also* deferred dynamic system parameter, dynamic system parameter, and static system parameter.

system table

A database table that contains definition and tracking information about the database. StorHouse/RM automatically creates system tables for each new database and updates system tables when you create and drop database user components and when you load data. Range indexes are stored in system tables. *See also* metadata.

system table index

A type of index that StorHouse/RM automatically creates for some system tables when a database is created.

system table index log

A system component that is used to recover changes to system table indexes. Before StorHouse/RM updates a system table index, it first copies a before image (or undo record) to the system table index log.

system table log

A system component that is used to recover changes to system tables. Before StorHouse/RM updates a system table, it first copies a before image (or undo record) to the system table log.

system tablespace

Logically, a database component that stores metadata in a database. Physically, the database directory under which all database system component files reside. Each StorHouse database has a system tablespace.

table array, StorHouse/RFS

A set of tables that StorHouse/RFS uses to store file locator information for one or more collection sets. After you run the tblgen utility to create a collections table, StorHouse/RFS creates an initial set of 32 tables for file locator data and expands the array as necessary up to 255 sets of 32 tables.

table ID

A numeric identifier for each table in a database. StorHouse/RM assigns a table ID to a user table when you issue a CREATE TABLE statement. StorHouse/RM assigns table IDs to system tables when a database is created.

table data file

A StorHouse file that contains user table data. Each segment contains one table data file.

table definition

The table name, column definitions, and user tablespace assignment supplied on a CREATE TABLE statement. See also column definition.

tablespace

See system tablespace, user tablespace, or temporary tablespace.

tape cartridge

A type of removable media used to store data.

tape drive

A subunit of a library device that reads and writes data on removable tape cartridges.

tape volume

See tape cartridge.

tblgen utility

A StorHouse/RFS utility that creates the following tables in a StorHouse database: aliases, statistics, collections, and security. For Windows installations, you also use this utility to set the number of directory levels for extended security.

temporary tablespace

The area on RAID where StorHouse/RM temporarily stores result sets and builds and sorts indexes.

terabyte

A unit of measurement equivalent to approximately 1,000,000 megabytes or 1,000 gigabytes.

terminated data

Data fields followed by a termination delimiter. *See also* delimiter.

text record

A record that contains text only and ends with the appropriate end-of-line character, such as a UNIX newline (LF) or an ASCII carriage return/line feed (CRLF) pair. Contrast fixed-length record and variable-length record.

time literal

A constant that specifies hours (HH), minutes (MM), seconds (SS), and optionally milliseconds (CCC).

timestamp literal

A constant that specifies a date and time combination.

trailing blanks

One or more blank characters at the end of a data field. A SGI data loader does not trim trailing blanks from character-type data fields, VARCHAR data fields, or enclosed data fields (within enclosure delimiters).

Transaction

One or more SQL statements that are treated as a single unit of work. If successful and if all its changes are accepted, a transaction can be committed (made permanent). If unsuccessful, a transaction can be rolled back (canceled). *See also* atomic, commit, durable, and roll back.

transfer type

A mode of transfer between your client FTP tool and the StorHouse FTP server. A transfer type defines how data and end-of-line characters are interpreted. Two transfer types are ASCII and BINARY. See also ASCII transfer type and BINARY transfer type.

transport mechanism

See robotic arm.

truncated file

A file whose last known extent does not have the last extent flag. The Command Language SHOW FILE /EXTENT command enables you to display the last extent flag as an extent_status value. See also extent.

Trusted Edge®

Intelligent migration management software that simplifies moving forever-read data to a StorHouse repository for storage, retrieval, backup, and data protection throughout the content lifecycle. The product copies/moves files from local or NAS storage to a location managed by StorHouse/RFS according to administrator-specified policies. StorHouse/RFS subsequently groups the files and transfers them in collection format to any StorHouse-managed media, including tape.

Trusted Edge Enterprise Edition

An optional Trusted Edge component that includes all standard Trusted Edge features plus the ability to manage multiple Trusted Edge servers from a single graphical user interface. The customer license agreement determines the number of servers that can be managed through this interface.

Trusted Edge migration

Copying/moving files from local or NAS storage to a location managed by StorHouse/RFS according to administrator-specified policies.

tuple

A row in a table.

tuple ID

A row ID.

UF

See usage factor.

uid

See user identification code.

uncatalog

To remove directory information for files on volumes and volume sets stored on storage levels L (library device) and S (shelves). The Command Language UNCATALOG VOLUME and UNCATALOG VSET commands enable your system administrator to uncatalog volumes and volume sets. *Contrast* catalog.

unary operator

An arithmetic operator that reverses the sign of a positive numeric value operand (-, unary minus) or does not change the operand (+, unary plus). *See also* arithmetic operator.

undo record

A "before image" of a system table or system table index, copied to a system table log or system table index log. The automatic metadata recovery process applies undo records during metadata recovery to roll back incomplete transactions.

unload

To select data from a StorHouse user table(s) and copy it to a sequential file on a host or pipe it to a program. You can also unload LOB data to separate LOB files on a client or remote system.

UNLOAD statement

The SQL-like control statement that specifies the character set of result data, defines the format of result data, specifies a character to append to the end of result records, describes each data field in result records, and contains the query that selects the StorHouse data to unload. *Contrast* LOAD DATA statement.

unloader data type

A data type that describes the result data fields being unloaded from a database. You provide this data type in the UNLOAD statement.

unprotected database

The state of a database when access and further updates to inconsistent metadata is allowed.

upward migration

The process of moving files from their resident file set to the performance buffer for faster access. Files are queued for upward migration using the StorHouse Command Language STAGE command. (*See* manual migration.)

usage factor (UF)

The portion of the migration factor that indicates the number of times a file has been accessed, when those accesses occurred, and the size of the file.

user components

See database user components.

user directory

A directory in a staging area. You assign a collector to a staging directory and to a specific user directory. StorHouse/RFS automatically creates a folder or directory in the virtual file system for each user directory. *See also* staging area and staging directory.

user mapping

In a federated system, the correspondence between a user’s DB2 identity, or authorization ID, and the identify used when communicating with a data source, for instance, a StorHouse account ID.

user file

A file you transfer from a host to the StorHouse system.

user identification code (uid)

A number (0-9999) used by StorHouse to identify a session.

user log

A user-accessible StorHouse log file that contains statistical information to help you manage the system. Statistical information includes startup time, shutdown time, signons, function counts, security violations, and file operations.

user table

A database component that contains columns and rows of user data.

user table name

An SQL identifier that you provide on the CREATE TABLE statement to name a user table.

user tablespace

A logical database component that defines how to store segment files on StorHouse. A user tablespace consists of one or more subspaces. *See also* subspace.

user tablespace ID

A numeric identifier that StorHouse/RM assigns to a user tablespace when you issue a CREATE TABLE SPACE statement. Tablespace ID 0 is reserved for the system tablespace and tablespace ID 1 is reserved for the temporary tablespace.

user tablespace name

An SQL identifier that you provide on the CREATE TABLE SPACE statement to name a user tablespace.

value index

A list of all values in one or more columns of a given user table. StorHouse/RM typically uses a value index to search for a range of values or for a specific value.

value index file

The value index entries that are stored in STORHOUSE-type files on StorHouse.

Value Realization Modeler

A tool that enables customer prospects to determine the immediate cost benefits of a StorHouse® solution during the pre-purchase evaluation/discovery process.

variable-length record

A record with a length that varies from other records in the file. Each record is preceded by a shortword (2-byte integer) that indicates the length of the data record. *Contrast* fixed-length record, text record, and LOB record.

version

See file version.

version number, StorHouse

A number assigned to a file version. The system assigns version number 1 to a file when it is created in StorHouse and increments the number by one each time a version is added or deleted. Version numbers can be relative (0 through -32,767) or absolute (1 through 32,768). The most recent file version has relative version number 0, and the previous file versions have relative version numbers – 1, -2, -3, up to -32,767. *See also* file revision, file version, and revision number.

vid

See volume identification code.

view

An alternative representation of data from one or more tables. A view can include some or all of the columns contained in tables on which the view is based.

virtual file system, StorHouse/RFS

The shared drive or mount point on the platform running the StorHouse/RFS server. The virtual file system is the user or application interface to StorHouse/RFS. To an application, the virtual file system, which looks like any other available drive or disk on the network, is where users or applications write and open files.

Virtual Record Access Manager (VRAM™)

The SGI software component that controls access to data at the record level, updates files logically, ensures index integrity, and manages all file versions.

virtual volume

A named set of data that is mounted in a drive and accessed like a logical volume. However, unlike a logical volume, a virtual volume does not correspond to a component that can be physically removed from a library. MAID is an example of a type of storage device that contains only virtual volumes.

volume

A unit of media where data can be recorded and read. Optical volumes are also called *platters* and *optical disk cartridges*. Tape volumes are also called *tape cartridges*. *See* physical volume.

volume identification code (vid)

A code that identifies a volume by media type, recording type, volume label, and volume side.

volume label

A code assigned by StorHouse to a volume during initialization. Labels for optical volumes contain 1 to 16 characters. Labels for tape volumes contain 1 to 6 characters. The volume label is part of the volume identification code, making each volume identification code unique for a media type and recording type.

A system administrator can define the method StorHouse uses to generate volume labels. If desired, the administrator can set up StorHouse to generate custom volume labels.

volume migration

The movement of volumes from one storage level to another or between library devices within storage level L. Volume migration is also the movement of a blank volume into a library device.

Your operator can manually migrate a volume or volume set using the Command Language MOVE VOLUME and MOVE VSET commands. Your system administrator can manually migrate a blank volume into a library device using the Command Language MIGRATE command. *See also* blank volume and file migration.

volume set

A collection of one or more physical or virtual volumes that StorHouse treats as a unit. Every removable volume in a StorHouse system is a member of a volume set.

volume set LIMIT

The attribute that sets the maximum size of a volume set. The LIMIT must be an integer multiple of a physical volume’s maximum size. StorHouse will not extend the size of a volume set beyond the LIMIT.

volume side

The physical surface (A or B) of a volume.

Volume Storage Allocation and Control (VSAC™)

The StorHouse software components that manage storage allocation according to attributes that you define.

volume surface

See volume side.

volume table of contents (VTOC)

A directory of the files stored on a volume. The VTOC contains the file labels for each file extent on a volume. Each volume has a VTOC.

VRAM

See Virtual Record Access Manager.

VROM™

SGI client/server software for accessing StorHouse data from Windows applications.

VSAC

See Volume Storage Allocation and Control.

VSET

See volume set.

VTF

See Vulnerability Time Factor.

VTOC

See volume table of contents.

Vulnerability Time Factor (VTF)

The file attribute that indicates when StorHouse should copy new extents of a file version from the performance buffer to their primary file set. VTF values are as follows:

- DIRECT – Bypass the performance buffer and write the file version directly to the primary file set.

- NEXT – Write the file version to the performance buffer and then copy it to the primary file set the next time the system performs the Command Language BACKUP command.
- NOW – Write the file version to the performance buffer and then copy it to the primary file set immediately.

Web-AMMO™

SGI software that enables AMMO-II users to retrieve and view all AMMO-II reports over the Internet or an intranet using a standard web browser.

whitespace

A blank, space, or tab in character data. *See also* leading blanks and trailing blanks.

wild card character

An asterisk (or user-defined character) used in place of data fields or partial data fields in some Command Language commands to indicate that all values or any matching values can be substituted for the wild card. The Command Language SET USER /WILDCARD command enables you to change the wild card character.

WITH GRANT OPTION

The SQL GRANT statement component that enables the specified accounts to grant their privileges or a subset of their privileges to other accounts.

WORM

See write-once-read-many.

wrapper

The software that implements a connection in a federated system. The StorHouse/UDB Link is the SGI-supplied wrapper software to implement the connection between DB2 UDB and StorHouse.

write-back

An operation that copies new file extents from the performance buffer to their primary file sets. Write-back ensures that the primary copy of a new or revised file version is complete. Write-back occurs when your system administrator or the MIGRATE command issues the Command Language BACKUP command.

WRITELOCKED state

A state indicating that additional storage allocations are not allowed on the volume side unless it is unwritelocked. StorHouse can read file extents that reside on a writelocked volume side.

writelocked volume

A volume with both sides classified as writelocked.

write once, read many (WORM)

Technology that supports the writing of data to a medium one time and the reading of data from that medium many times. *Contrast* erasable. *Compare* compliant media.

zettabyte

A unit of storage equal to 1,000 exabytes or one billion terabytes.