CREATE STOGROUP statement

Last Updated: 2023-01-13

8.10.2

The CREATE STOGROUP statement defines a new storage group within the database, assigns storage paths to the storage group, and records the storage group definition and attributes in the catalog.

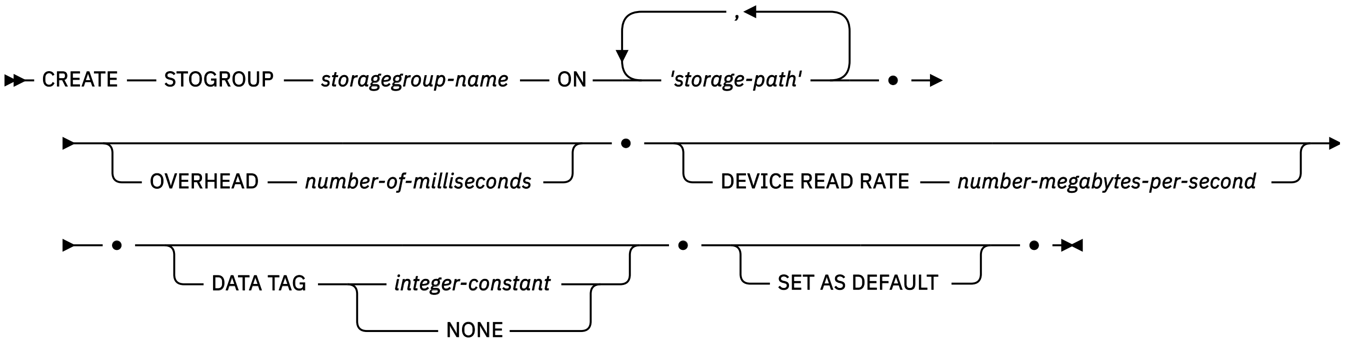
Invocation

This statement can be embedded in an application program or issued interactively. It is an executable statement that can be dynamically prepared only if DYNAMICRULES run behavior is in effect for the package (SQLSTATE 42509).

Authorization

The privileges that are held by the authorization ID of the statement must include SYSCTRL or SYSADM authority.

Syntax



Description

storagegroup-name

Names the storage group. This is a one-part name. It is an SQL identifier (either ordinary or delimited). The *storagegroup-name* must not identify a storage group that exists at the current server (SQLSTATE 42710). The *storagegroup-name* must not begin with the characters 'SYS' (SQLSTATE 42939).

ON

Specifies storage paths to be added for the named storage group. For partitioned database environments, the same storage paths are defined on all database partitions unless database partition expressions are used.

storage-path

A string constant that specifies containers the location where automatic storage table spaces are to be created. The format of the string depends on the operating system, as illustrated in the following table:

Operating system	Format of storage path string
Linux® AIX®	An absolute path
Windows	The letter name of a drive

The string can include database partition expressions to specify database partition number information in the storage path.

The maximum length of a path is 175 characters (SQLSTATE 54036).

A storage path that is added must be valid according to the naming rules for paths, and must be accessible (SQLSTATE 57019). Similarly, in a partitioned database environment, the storage path must exist and be accessible on every database partition (SQLSTATE 57019).

OVERHEAD *number-of-milliseconds*

Specifies the I/O controller usage and disk seek and latency time. This value is used to determine the cost of I/O during query optimization. The value of *number-of-milliseconds* is any numeric literal (integer, decimal, or floating point). If this value is not the same for all storage paths, set the value to a numeric literal that represents the average for all storage paths that belong to the storage group.
If the OVERHEAD clause is not specified, the OVERHEAD is set to 6.725 milliseconds.

DEVICE READ RATE *number-megabytes-per-second*

Specifies the device specification for the read transfer rate in megabytes per second. This value is used to determine the cost of I/O during query optimization. The value of *number-megabytes-per-second* is any numeric literal (integer, decimal, or floating point). If this value is not the same for all storage paths, set the value to a numeric literal that represents the average for all storage paths that belong to the storage group.
If the DEVICE READ RATE clause is not specified, the DEVICE READ RATE is set to the built-in default of 100 megabytes per second.

DATA TAG *`integer-constant`* or DATA TAG NONE

Specifies a tag for the data for table spaces that use this storage group unless explicitly overridden by the table space definition. This value can be used as part of a WLM configuration in a work class definition or referenced within a threshold definition. For more information, see the CREATE WORK CLASS SET and CREATE THRESHOLD statements.

integer-constant

Valid values for *integer-constant* are integers 1 - 9.

NONE

If NONE is specified, there is no data tag.

SET AS DEFAULT

Specifies the storage group that is created is designated as the default storage group. If no default storage group exists, the first one created is designated the default even if this clause is not specified. Since there can only be one storage group that is designated as the default storage group, specifying this clause removes the default attribute from the existing default storage group. Specifying a new default storage group has no effect to the storage group used by existing table spaces.

Rules

- The CREATE STOGROUP statement cannot be run while a database partition server is being added (SQLSTATE 55071).
- A storage group can have up to 128 defined storage paths (SQLSTATE 5U009).
- A database instance can have up to 256 defined storage groups (SQLSTATE 54035).

Notes

- ***Calculation of free space:*** When free space is calculated for a storage path on a database partition, the database manager checks for the existence of the following directories or mount points within the storage path. The database manager uses the first one that is found.

  ```
  <storage path>/<instance name>/NODE####/<database name>
  <storage path>/<instance name>/NODE####
  <storage path>/<instance name>
  <storage path>
  ```

  Where:

  - `<storage path>` is a storage path that is associated with the database.
  - `<instance name>` is the instance under which the database resides.
  - `NODE####` corresponds to the database partition number (for example, NODE0000 or NODE0001).
  - `<database name>` is the name of the database.

- ***Isolating multiple database partitions under one storage path:*** The file systems can be mounted at a point beneath the storage path, and the database manager recognizes that the actual amount of free space available for table space containers might not be the same amount that is associated with the storage path directory itself.

Consider an example in which two logical database partitions exist on one physical computer, and a single storage path exists (`/dbdata`). Each database partition uses this storage path, but you might want to isolate the data from each partition within its own file system. In this case, a separate file system can be created for each partition and it can be mounted at `/dbdata/<instance>/NODE####`. When creating containers on the storage path and determining free space, the database manager does not retrieve free space information for `/dbdata`, but instead retrieves it for the corresponding `/dbdata/<instance>/NODE####` directory.

- ***Multiple storage paths:*** A storage path can be added to different storage groups, or to the same storage group multiple times.
- ***Similar media characteristics:*** Ensure that the storage paths added to a storage group share similar media characteristics. If the media characteristics are dissimilar, specify a value that represents an average for OVERHEAD and DEVICE READ RATE.

Examples

1. Create a storage group that is named HIGHEND with two paths under the /db directory (`/db/filesystem1` and `/db/filesystem2`) which are attached to Solid State Disks.

```
CREATE STOGROUP HIGHEND ON '/db/filesystem1', '/db/filesystem2'
   OVERHEAD 0.75 DEVICE READ RATE 500
```

8.10.2

2. Create a storage group that is named MIDRANGE with two drives D and E and designate it as the default storage group.

```
CREATE STOGROUP MIDRANGE ON 'D:\', 'E:\' SET AS DEFAULT
```

3. Create a storage group that is named MIDRANGE with two paths under the /db directory, and designate it as the default storage group.

```
CREATE STOGROUP MIDRANGE ON '/db/filesystem1', '/db/filesystem2' SET AS DEFAULT
```

Related concepts

→ Using database partition expressions

Related reference

→ ADMIN_GET_STORAGE_PATHS table function - retrieve automatic storage path information

→ ALTER STOGROUP statement

→ CREATE THRESHOLD statement

→ CREATE WORK CLASS SET statement

→ MON_GET_CONTAINER table function - Get table space container metrics

→ MON_GET_TABLESPACE table function - Get table space metrics

→ RENAME STOGROUP statement