

NAME

ALTER_COLLATION – change the definition of a collation

SYNOPSIS

ALTER COLLATION *name* REFRESH VERSION

ALTER COLLATION *name* RENAME TO *new_name*

ALTER COLLATION *name* OWNER TO { *new_owner* | CURRENT_ROLE | CURRENT_USER | SESSION_USER }

ALTER COLLATION *name* SET SCHEMA *new_schema*

DESCRIPTION

ALTER COLLATION changes the definition of a collation.

You must own the collation to use **ALTER COLLATION**. To alter the owner, you must be able to SET ROLE to the new owning role, and that role must have CREATE privilege on the collation's schema. (These restrictions enforce that altering the owner doesn't do anything you couldn't do by dropping and recreating the collation. However, a superuser can alter ownership of any collation anyway.)

PARAMETERS

name

The name (optionally schema-qualified) of an existing collation.

new_name

The new name of the collation.

new_owner

The new owner of the collation.

new_schema

The new schema for the collation.

REFRESH VERSION

Update the collation's version. See Notes below.

NOTES

When a collation object is created, the provider-specific version of the collation is recorded in the system catalog. When the collation is used, the current version is checked against the recorded version, and a warning is issued when there is a mismatch, for example:

WARNING: collation "xx-x-icu" has version mismatch

DETAIL: The collation in the database was created using version 1.2.3.4, but the operating system provides version 2.3.4.5.

HINT: Rebuild all objects affected by this collation and run ALTER COLLATION pg_catalog."xx-x-icu" REFRESH VERSION.

A change in collation definitions can lead to corrupt indexes and other problems because the database system relies on stored objects having a certain sort order. Generally, this should be avoided, but it can happen in legitimate circumstances, such as when upgrading the operating system to a new major version or when using **pg_upgrade** to upgrade to server binaries linked with a newer version of ICU. When this happens, all objects depending on the collation should be rebuilt, for example, using **REINDEX**. When that is done, the collation version can be refreshed using the command ALTER COLLATION ... REFRESH VERSION. This will update the system catalog to record the current collation version and will make the warning go away. Note that this does not actually check whether all affected objects have been rebuilt correctly.

When using collations provided by libc, version information is recorded on systems using the GNU C library (most Linux systems), FreeBSD and Windows. When using collations provided by ICU, the version information is provided by the ICU library and is available on all platforms.

Note

When using the GNU C library for collations, the C library's version is used as a proxy for the collation version. Many Linux distributions change collation definitions only when upgrading the C library, but this approach is imperfect as maintainers are free to back-port newer collation definitions

to older C library releases.

When using Windows for collations, version information is only available for collations defined with BCP 47 language tags such as en-US.

For the database default collation, there is an analogous command ALTER DATABASE ... REFRESH COLLATION VERSION.

The following query can be used to identify all collations in the current database that need to be refreshed and the objects that depend on them:

```
SELECT pg_describe_object(refclassid, refobjid, refobjsubid) AS "Collation",
       pg_describe_object(classid, objid, objsubid) AS "Object"
FROM pg_depend d JOIN pg_collation c
  ON refclassid = 'pg_collation'::regclass AND refobjid = c.oid
WHERE c.collversion <> pg_collation_actual_version(c.oid)
ORDER BY 1, 2;
```

EXAMPLES

To rename the collation de_DE to german:

```
ALTER COLLATION "de_DE" RENAME TO german;
```

To change the owner of the collation en_US to joe:

```
ALTER COLLATION "en_US" OWNER TO joe;
```

COMPATIBILITY

There is no **ALTER COLLATION** statement in the SQL standard.

SEE ALSO

CREATE COLLATION (**CREATE_COLLATION**(7)), DROP COLLATION (**DROP_COLLATION**(7))