

NAME

ALTER FOREIGN DATA WRAPPER – change the definition of a foreign-data wrapper

SYNOPSIS

ALTER FOREIGN DATA WRAPPER *name*

[HANDLER *handler_function* | NO HANDLER]

[VALIDATOR *validator_function* | NO VALIDATOR]

[OPTIONS ([ADD | SET | DROP] *option* ['*value*'] [, ...])]

ALTER FOREIGN DATA WRAPPER *name* OWNER TO { *new_owner* | CURRENT_ROLE | CURRENT_USER | SESSION_USER }

ALTER FOREIGN DATA WRAPPER *name* RENAME TO *new_name*

DESCRIPTION

ALTER FOREIGN DATA WRAPPER changes the definition of a foreign-data wrapper. The first form of the command changes the support functions or the generic options of the foreign-data wrapper (at least one clause is required). The second form changes the owner of the foreign-data wrapper.

Only superusers can alter foreign-data wrappers. Additionally, only superusers can own foreign-data wrappers.

PARAMETERS

name

The name of an existing foreign-data wrapper.

HANDLER *handler_function*

Specifies a new handler function for the foreign-data wrapper.

NO HANDLER

This is used to specify that the foreign-data wrapper should no longer have a handler function.

Note that foreign tables that use a foreign-data wrapper with no handler cannot be accessed.

VALIDATOR *validator_function*

Specifies a new validator function for the foreign-data wrapper.

Note that it is possible that pre-existing options of the foreign-data wrapper, or of dependent servers, user mappings, or foreign tables, are invalid according to the new validator. PostgreSQL does not check for this. It is up to the user to make sure that these options are correct before using the modified foreign-data wrapper. However, any options specified in this **ALTER FOREIGN DATA WRAPPER** command will be checked using the new validator.

NO VALIDATOR

This is used to specify that the foreign-data wrapper should no longer have a validator function.

OPTIONS ([ADD | SET | DROP] *option* ['*value*'] [, ...])

Change options for the foreign-data wrapper. ADD, SET, and DROP specify the action to be performed. ADD is assumed if no operation is explicitly specified. Option names must be unique; names and values are also validated using the foreign data wrapper's validator function, if any.

new_owner

The user name of the new owner of the foreign-data wrapper.

new_name

The new name for the foreign-data wrapper.

EXAMPLES

Change a foreign-data wrapper dbi, add option foo, drop bar:

```
ALTER FOREIGN DATA WRAPPER dbi OPTIONS (ADD foo '1', DROP bar);
```

Change the foreign-data wrapper dbi validator to bob.myvalidator:

```
ALTER FOREIGN DATA WRAPPER dbi VALIDATOR bob.myvalidator;
```

COMPATIBILITY

ALTER FOREIGN DATA WRAPPER conforms to ISO/IEC 9075–9 (SQL/MED), except that the **HANDLER**, **VALIDATOR**, **OWNER TO**, and **RENAME** clauses are extensions.

SEE ALSO

CREATE FOREIGN DATA WRAPPER (**CREATE_FOREIGN_DATA_WRAPPER(7)**), **DROP FOREIGN DATA WRAPPER** (**DROP_FOREIGN_DATA_WRAPPER(7)**)