

**NAME**

CREATE\_CONVERSION – define a new encoding conversion

**SYNOPSIS**

```
CREATE [ DEFAULT ] CONVERSION name
    FOR source_encoding TO dest_encoding FROM function_name
```

**DESCRIPTION**

**CREATE CONVERSION** defines a new conversion between two character set encodings.

Conversions that are marked **DEFAULT** can be used for automatic encoding conversion between client and server. To support that usage, two conversions, from encoding A to B *and* from encoding B to A, must be defined.

To be able to create a conversion, you must have **EXECUTE** privilege on the function and **CREATE** privilege on the destination schema.

**PARAMETERS****DEFAULT**

The **DEFAULT** clause indicates that this conversion is the default for this particular source to destination encoding. There should be only one default encoding in a schema for the encoding pair.

*name*

The name of the conversion. The conversion name can be schema-qualified. If it is not, the conversion is defined in the current schema. The conversion name must be unique within a schema.

*source\_encoding*

The source encoding name.

*dest\_encoding*

The destination encoding name.

*function\_name*

The function used to perform the conversion. The function name can be schema-qualified. If it is not, the function will be looked up in the path.

The function must have the following signature:

```
conv_proc(
    integer, -- source encoding ID
    integer, -- destination encoding ID
    cstring, -- source string (null terminated C string)
    internal, -- destination (fill with a null terminated C string)
    integer, -- source string length
    boolean -- if true, don't throw an error if conversion fails
) RETURNS integer;
```

The return value is the number of source bytes that were successfully converted. If the last argument is false, the function must throw an error on invalid input, and the return value is always equal to the source string length.

**NOTES**

Neither the source nor the destination encoding can be **SQL\_ASCII**, as the server's behavior for cases involving the **SQL\_ASCII** “encoding” is hard-wired.

Use **DROP CONVERSION** to remove user-defined conversions.

The privileges required to create a conversion might be changed in a future release.

**EXAMPLES**

To create a conversion from encoding UTF8 to LATIN1 using **myfunc**:

```
CREATE CONVERSION myconv FOR 'UTF8' TO 'LATIN1' FROM myfunc;
```

**COMPATIBILITY**

**CREATE CONVERSION** is a PostgreSQL extension. There is no **CREATE CONVERSION** statement in the SQL standard, but a **CREATE TRANSLATION** statement that is very similar in purpose and syntax.

**SEE ALSO**

ALTER CONVERSION (**ALTER\_CONVERSION**(7)), CREATE FUNCTION (**CREATE\_FUNCTION**(7)), DROP CONVERSION (**DROP\_CONVERSION**(7))