

# FDW(Foreign Data Wrapper)

## 機能

dblink 以外に FDW という機能が PostgreSQL 9.1 から使えるようになった。  
外部データベースのテーブルを PostgreSQL のテーブルのように扱えるようにする機能。

## FDW で Oracle と接続する

### 機能

PostgreSQL 9.3 の oracle\_fdw から INSERT、UPDATE、DELETE がサポートされた。  
それ以前は、SELECT のみ。

### インストール

#### ダウンロード

[http://laurenz.github.io/oracle\\_fdw/](http://laurenz.github.io/oracle_fdw/)  
からダウンロード

Oracle Client も必要なので、Oracle Client Instant をダウンロードしてインストールしておく。

#### コンパイル

```
sudo yum install postgresql-devel
tar zxvf oracle_fdw-ORACLE_FDW_1_2_0.tar.gz
cd oracle_fdw-ORACLE_FDW_1_2_0
make
sudo make install
```

pg\_config が見つからない場合は

```
find / -name pg_config
```

で探して、ディレクトリにパスを通す。

```
export PATH=${PATH}:/usr/pgsql-9.4/bin
```

など。

#### 起動スクリプト修正

PostgreSQL のプロセスに Oracle ライブラリのパスを教える必要がある。

```
sudo vim /etc/sysconfig/postgresql
```

```
LD_LIBRARY_PATH=/usr/lib/oracle/12.1/client64/lib/
```

postgresql 起動スクリプトを作成する。

本来の起動スクリプトを読み込んでからカスタマイズを書く。

Environment で直接指定してもいいけど、環境変数ファイルを指定しておく。

```
sudo vim /etc/systemd/system/postgresql.service
```

```
.include /lib/systemd/system/postgresql.service  
[Service]  
EnvironmentFile=/etc/sysconfig/postgresql
```

systemd でない場合は、postgres ユーザの .bashrc に

```
export LD_LIBRARY_PATH=/usr/lib/oracle/10.2.0.4/client64/lib/
```

としてもよい。

### 拡張読み込み

postgresql 内に oracle\_fdw の拡張を読み込む。

```
psql -U postgres  
CREATE EXTENSION oracle_fdw;
```

もし、ここでエラーが出る場合は、SELinux まわりかライブラリのパス指定にミスがある可能性がある。

### 外部テーブルの定義

リモートデータベース名は任意の名前。

#### リモートデータベースを作成する

```
create server リモートデータベース名 foreign data wrapper oracle_fdw options (dbserver  
'//xxx.xxx.xxx.xxx:1521/XE');
```

#### ユーザマッピングの作成

```
create user mapping for postgresql のユーザ名 server リモートデータベース名 options(user 'oracle の  
ユーザ', password 'oracle のパスワード');
```

#### 外部テーブルの作成

```
create foreign table テーブル名 (  
    a date,  
    b date,  
    c text,  
    d text) server リモートデータベース名 options(schema 'oracle のスキーマ', table 'oracle の  
テーブル');
```

テーブル名は大文字、小文字を識別するので注意

#### スキーマごとのインポート

テーブルを一つ一つマッピングせずに、スキーマをまるごとリンクできる。

(9.5 以降)

```
IMPORT FOREIGN SCHEMA リモートスキーマ名 FROM SERVER リモートデータベース名 INTO ローカルスキーマ名;
```