

[http://powergres.sra.co.jp/s/ja/tech/plus/experience/vol4/oracle\\_migration2.php](http://powergres.sra.co.jp/s/ja/tech/plus/experience/vol4/oracle_migration2.php)  
<http://blog.livedoor.jp/loopus/archives/50311640.html>

## 準備

### Perl

perl を使える状態にしておく

### Ora2pg

#### ダウンロード

<http://pgfoundry.org/projects/ora2pg>  
から Ora2pg をダウンロードするして解凍する

#### コンパイル? ( 必要ないかも )

解凍したディレクトリで

```
perl Makefile.PL
```

を実行する。

```
Ora2Pg.pm
```

が出来上がればOK .

#### 移行用のスクリプトを作成

```
#!/usr/bin/perl

BEGIN {
    $ENV{ORACLE_HOME}='C:\oracle\exe\app\oracle\product\10.2.0\server';
    $ENV{NLS_LANG} = 'JAPANESE_JAPAN.JA16SJISTILDE';
}

use strict;
use Ora2Pg;

my $schema=new Ora2Pg (
    datasource=>'dbi:Oracle:host=localhost;sid=xe;port=1521',
    user=>'system',
    password=>'manager',
    schema=>'orauser',
    type=>'TABLE',
    debug=>1
);

$schema->export_schema('C:\temp\schemata.sql');
exit(0);
```

見たいな感じのを作る。

```
$ENV{ORACLE_HOME}
```

Oracle をインストールしたディレクトリ。Linux だと  
\$ENV{ORACLE\_HOME}=/home/oracle/app/oracle/product/10.1.0/db\_1; とか。

\$ENV{NLS\_LANG}

Oracle の文字コード

datasource

Oracle に接続するための情報

user

Oracle へ接続するためのユーザー。 dba\_users などにアクセス出来るユーザ。

password

上記ユーザのパスワード

schema

移行対象のスキーマ

type

|            |  |
|------------|--|
| TABLE      | Extract all tables with indexes, primary keys, unique keys,foreign keys and check constraints. |
| VIEW       | Extract only views.  |
| GRANT      | Extract roles converted to Pg groups, users and grants on all objects.                         |
| SEQUENCE   | Extract all sequence and their last position.  |
| TABLESPACE | Extract storage space, need PostgreSQL >= v8.  |
| TRIGGER    | Extract triggers defined following actions.  |
| FUNCTION   | Extract functions.   |
| PROCEDURES | Extract procedures.  |
| PACKAGE    | Extract packages and package bodies.   |
| DATA       | Extract datas as INSERT statement.   |
| COPY       | Extract datas as COPY statement.   |
| PARTITION  | Extract range and list Oracle partitioning.  |
| TYPE       | Extract user defined Oracle type.  |

```
export_schema('パス');
```

スキーマの出力先。 /home/oracle/data\_for\_migrate/schema.sql とか。

## 移行作業

### 1 . Oracle からエクスポート

準備作業で作成した移行用スクリプトを実行する。

### 2 . 「5C 問題」の対応 ( Oracle の文字コードが SJIS の場合 )

Oracle の文字コードが SJIS の場合、5C 問題の関係で出力ファイルに不要な「\」が出力される。

```

use Encode;
my $file = shift;
open(IN, $file) or die "Can't open $file: $!\n";
while(<IN>){
    $tmp = $_;
    # $tmp = s/\\\\\\\\/\\/g;
    $tmp = s/([%x81-%x9f%x0-%xfc]%%)%%/¥1/g;
    Encode::from_to($tmp, "shiftjis", "utf8" );
    print $tmp;
}
close(IN);

```

こんなスクリプトを書いて、convert.pl として保存する。

1 . で出力したファイルを data.sql とした場合

```
perl convert.pl data.sql > data_utf8.sql
```

と実行して、不要な「\」を取り除いて UTF8 に変換。

### 3 . Postgresql ヘインポート

```
psql -h ホスト名 -U ユーザー -f インポートファイル データベース名
```

または、psql 内で

```
psql => ¥i インポートファイル
```

でインポート。

#### orafce

<http://orafce.projects.postgresql.org/index-ja.html>

orafce の目的は Oracle Database で使用できる関数と互換性のある関数を PostgreSQL に実装することです。

個人的な感想としては意外と。

#### インストール

[http://pgfoundry.org/frs/?group\\_id=1000113](http://pgfoundry.org/frs/?group_id=1000113)

から rpm をダウンロード。

```
rpm -ihv orafce-xxxxxx.rpm
```

でインストール。終わったら

```
psql -U postgres testdb < orafunc.sql
```

でスクリプト実行。

orafunc.sql は、/usr/share/pgsql/contrib あたりにはあるはず。

無ければ、find で検索。