

# Migration vers PostgreSQL

## Contexte :

Migration d'une base de données Oracle 11.2.0.4 vers PostgreSQL 11.

L'outil PgBadger 11.2 est également installé et testé.

<https://github.com/darold/pgbadger/tree/v11.2>

Le mode opératoire sera appliquée pour l'ensemble des bases à migrer Oracle -> Postgres (seul les fichiers de configuration sont susceptibles de changer, pg\_hba.conf, ora2pg.conf).

Serveurs dédié aux migrations :

pupglmig001 pour VAL /PROD

rupglmig001 pour Hors - Prod.

Cas test avec la base source DMDVAL et base cible DMD.

- [1 - Prérequis système](#)
- [2 - Initialiser et démarrer le serveur postgres](#)
- [3 - Configuration](#)
- [4 - Migration](#)
  - [4.1 Exporter le DDL](#)
  - [4.2 Créer les partitions \(facultatif\)](#)
  - [4.3 Extraire les données \(facultatif\)](#)
  - [4.4 Configuration](#)
  - [4.5 Import](#)
  - [4.6 Refresh](#)
- [6 - Problème durant l'import](#)
- [7 - Jira](#)
- [8 - Sources](#)
- [8 - Bibliographie](#)
- [FAQ](#)

## 1 - Prérequis système

cible Postgres:

4 coeurs

32 Go

Source Oracle :

8 CPU

189 Go

serveur de gestion :

8 CPU

32 Go

j => connexion Postgres

J => connexion Oracle

Télécharger ora2pg-20.0

ici : <https://github.com/darold/ora2pg/releases>

et dézipper sous /mnt/dbbcp/PROJ\_ENCOURS/upgr\_virtu/psql\_mig/ora2pg-20.0 .

Installation du client oracle et de la JDK

```
rpm -ivh /mnt/dbbcp/sources_postgres/oracle-instantclient19.8-basic-19.8.0.0-1.x86_64.rpm
rpm -ivh /mnt/dbbcp/sources_postgres/oracle-instantclient19.8-devel-19.8.0.0-1.x86_64.rpm
rpm -ivh /mnt/dbbcp/sources_postgres/oracle-instantclient19.8-sqlplus-19.8.0.0-1.x86_64.rpm
rpm -ivh /mnt/dbbcp/sources_oracle/jdk-8u261-linux-x64.rpm
```

Installation de ora2pg :

```
cd /mnt/dbbcp/PROJ_ENCOURS/upgr_virtu/psql_mig/ora2pg-20.0
perl Makefile.PL
make && make install
```

## Installation de DBD::Oracle

```
cd /mnt/dbbkb/PROJ_ENCOURS/upgr_virtu/psql_mig/DBD-Oracle-1.80
export LD_LIBRARY_PATH=/opt/oracle/instantclient_19_8
export ORACLE_HOME=/opt/oracle/instantclient_19_8
perl Makefile.PL
make
make && make install
```

## Installation de DBD:Pg

```
cd /mnt/dbbkb/PROJ_ENCOURS/upgr_virtu/psql_mig/DBD-Pg-3.14.2
perl Makefile.PL
make
make && make install
```

ora2pg est installé par défaut sous : `/usr/local/bin/ora2pg`  
Le fichier de configuration est sous : `/etc/ora2pg/`

## Installer postgresql 11

```
yum install postgresql11 postgresql92-contrib.x86_64 postgresql92-devel.x86_64
```

```
groupadd --gid 27 postgres
useradd postgres -u 26 -g 27 -b /home
```

### Initialiser les variable suivantes dans le .bash profile

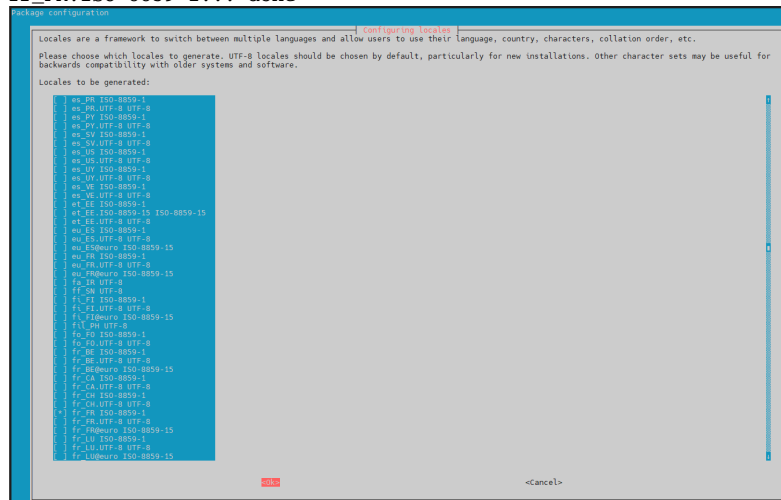
```
export PERL5LIB=/app/ora2pg
export LD_LIBRARY_PATH=/app/oracle/product/11.2.0/lib
export ORACLE_HOME=/app/oracle/product/11.2.0
export LANG=fr_FR.ISO-8859-1
```

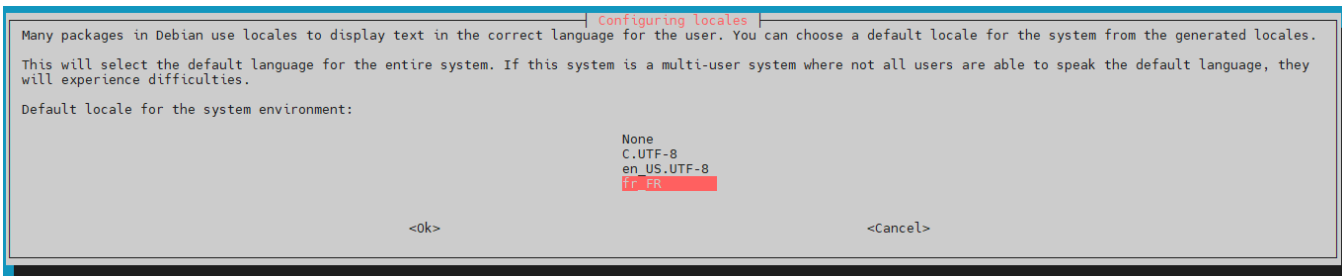
Création des répertoires pour stocker les tablespaces :

```
/pgdata/tbs_<nom_bdd>_<nom_tbs>
```

/pgindex/tbs\_<nom\_bdd>\_<nom\_tbs>

```
Ajouter la locale attendue :
dpkg-reconfigure locales
Generating locales (this might take a while)...
en_US.ISO-8859-1... done
en_US.ISO-8859-15... done
fr_FR.ISO-8859-1... done
```





```
root@dupglc001:~# dpkg-reconfigure locales
Generating locales (this might take a while)...
  en_US.UTF-8... done
  fr_FR.UTF-8... done
Generation complete.
root@dupglc001:~# exit
logout
jjoly@dupglc001:~$ sudo -tu postgres
postgres@dupglc001:~$ pg_ctl initdb -D /pgcluster/data
Les fichiers de ce système de bases de données appartiendront à l'utilisateur postgres.
Le processus serveur doit également lui appartenir.

L'instance sera initialisée avec la locale fr_FR.UTF-8.
L'encodage par défaut des bases de données sera configuré en conséquence
avec LATIN1.
La configuration de la recherche plein texte sera initialisée en français.

Les sommes de contrôle des pages de données sont désactivées.

correction des droits sur le répertoire existant /pgcluster/data... ok
création des sous-répertoires... ok
sélection de la valeur par défaut pour max_connections... 100
sélection de la valeur par défaut pour shared_buffers... 128MB
sélection du fuseau horaire par défaut... Europe/Paris
sélection de l'implémentation de la mémoire partagée dynamique... posix
création des fichiers de configuration... ok
lancement du script bootstrap... ok
exécution de l'initialisation après bootstrap... ok
synchronisation des données sur disque

ATTENTION : active l'authentification trust pour les connexions
locales.
Vous pouvez changer cette configuration en editant le fichier pg_hba.conf
ou en utilisant l'option -A, ou --auth-local et --auth-host au prochain
lancement d'initdb.

Succès. Vous pouvez maintenant lancer le serveur de bases de données en utilisant :

/usr/lib/postgresql/11/bin/pg_ctl -D /pgcluster/data -l fichier_de_trace start
```

sudo apt install postgresql-server-dev-11

Pour les droits :

chown -R postgres:postgres /usr/share/doc/postgresql-server-dev-11

chown -R postgres:postgres /usr/lib/postgresql

chown -R postgres:postgres /usr/share/postgresql

mkdir /usr/share/doc/postgresql-doc-11

chown postgres:postgres /usr/share/doc/postgresql-doc-11

Extensions :

Installer external file :

unzip -d \$HOME /mnt/dbbcp/sources\_postgres/external\_file-master.zip

Installer orafce :

unzip -d \$HOME /mnt/dbbcp/sources\_postgres/orafce-master.zip

## 2 - Initialiser et démarrer le serveur postgres

L'initialisation est adaptée en fonction de la locale qui a été choisie au niveau de l'OS :

pg\_ctl initdb -D /pgcluster/data

Démarrage des processus Postgres :

/usr/bin/pg\_ctl -D /pgcluster/data -l logfile start

postgresql.conf :

listen\_addresses = '\*' # what IP address(es) to listen on;

pg\_hba.conf

host all all <servername>/32 trust

Créer les extensions suivantes :

```
CREATE EXTENSION pg_trgm;
CREATE EXTENSION orafce;
CREATE EXTENSION external_file;
```

```
mba=# \dx
          List of installed extensions
  Name      | Version | Schema      | Description
-----+-----+-----+-----
external_file | 1.0     | external_file | functions to read or write files from postgresql server filesystems
plpgsql     | 1.0     | pg_catalog   | PL/pgSQL procedural language
(2 rows)

mba=# CREATE EXTENSION orafce;
CREATE EXTENSION
mba=# \dx
          List of installed extensions
  Name      | Version | Schema      | Description
-----+-----+-----+-----
external_file | 1.0     | external_file | functions to read or write files from postgresql server filesystems
orafce       | 3.14    | public      | Functions and operators that emulate a subset of functions and packages from the Oracle RDBMS
plpgsql     | 1.0     | pg_catalog   | PL/pgSQL procedural language
(3 rows)

mba=#
```

Vérifier que les schémas suivants sont créés :

```
postgres=# \dn
          List of schemas
  Name      | Owner
-----+-----
dbms_alert  | postgres
dbms_assert | postgres
dbms_output | postgres
dbms_pipe   | postgres
dbms_random | postgres
dbms_utility | postgres
external_file | postgres
oracle      | postgres
plunit      | postgres
plvchr      | postgres
plvdate     | postgres
plvlex      | postgres
plvstr      | postgres
plvsubst    | postgres
public      | postgres
utl_file    | postgres
(16 rows)
```

Calculer les statistiques sur le serveur Oracle avant migration.

### 3 - Configuration

Création d'un projet :

```
ora2pg --project_base /app/migration/ --init_project dmd_mig
```

```
ora2pg -t SHOW_VERSION -c config/ora2pg.conf
ora2pg -t SHOW_REPORT -c config/ora2pg.conf
```

A propos des bases dont l'encodage n'est pas UTF8 :

**Si la base de données Oracle est en FRENCH\_FRANCE.WE8ISO8859P15, il faut définir le NLS\_LANG sur FRENCH\_FRANCE.WE8ISO8859P15 et le CLIENT\_ENCODING sur LATIN1.**

Le schéma dmd\_admin comporte 24 tables.

Pour la migration de dmdval ora2pg créé les fichiers suivants :

```
tables/INDEXES_table.sql
tables/table.sql
tables/FKEYS_table.sql
tables/CONSTRAINTS_table.sql
grants/grant.sql
sequences/sequence.sql
tablespaces/tablespace.sql
tablespaces/TBSP_INDEXES_tablespace.sql
directories/directorie.sql
```

La base postgres pour dmd aura le jeu de caractère suivant :

```
Encoding | Collate
-----+-----
LATIN1   | fr_FR.ISO-8859-1
```

### Création des tablespaces :

Les tablespaces de données doivent respecter cette norme :

/pgdata/tbs\_<nom\_bdd>\_<nom\_tbs>

Les tablespaces d'index doivent respecter cette norme :

/pgindex/tbs\_<nom\_bdd>\_<nom\_tbs>

exemple :

```
CREATE TABLESPACE tbs_dmd_data LOCATION '/pgdata/tbs_dmd_data';  
CREATE TABLESPACE tbs_dmd_indx LOCATION '/pgindex/tbs_dmd_indx';
```

```
GRANT ALL ON TABLESPACE tbs_dmd_data TO dmd_admin;  
GRANT ALL ON TABLESPACE tbs_dmd_indx TO dmd_admin;  
GRANT ALL ON TABLESPACE temp TO dmd_admin;
```

## Optimisation I/O

- Paramètres de durabilité
  - fsync = off
  - full\_page\_writes = off
  - synchronous\_commit = off
  - WAL sur des disques SSD
- Tuning du Kernel
  - vm.dirty\_background\_ratio = 1
  - vm.dirty\_ratio = 2

## PostgreSQL configuration - 1

- shared\_buffers = 10GB
- maintenance\_work\_mem = 2GB
- checkpoint\_segments = 64
- checkpoint\_completion\_target = 0.9
  - 4,5 min => 270s : 1Go/270s = 3,7Mo/s
- wal\_level = minimal
- archive\_mode = off
- wal\_buffers = 32Mo

## PostgreSQL configuration - 2

- Impact de l'autovacuum
  - autovacuum = on => pas de problèmes, INSERT
  - autoanalyze => ne se déclenchera qu'à la fin
- Gestion des traces
  - log\_destination & log\_directory sur la partition système
  - log\_min\_duration\_statement = -1
  - On ne trace que les erreurs

## 4 - Migration

Avant chaque migration

A désactiver sur la target :  
systemctl stop zabbix-agent.service  
service puppet stop

### 4.1 Exporter le DDL

./export\_schema.sh  
Optimisé : ./export\_schema.sh -J 3 -j 8

```

postgres@pupglmig001 testfrmval $ ./export_schema.sh
[=====] 33/33 tables (100.0%) end of scanning.
[=====] 7/7 objects types (100.0%) end of objects auditing.
Running: ora2pg -p -t TABLE -o table.sql -b ./schema/tables -c ./config/ora2pg.conf
[=====] 33/33 tables (100.0%) end of scanning.
[=====] 33/33 tables (100.0%) end of table export.
Running: ora2pg -p -t PACKAGE -o package.sql -b ./schema/packages -c ./config/ora2pg.conf
[=====] 0/0 packages (100.0%) end of output.
Running: ora2pg -p -t VIEW -o view.sql -b ./schema/views -c ./config/ora2pg.conf
[=====] 0/0 views (100.0%) end of output.
Running: ora2pg -p -t GRANT -o grant.sql -b ./schema/grants -c ./config/ora2pg.conf
Running: ora2pg -p -t SEQUENCE -o sequence.sql -b ./schema/sequences -c ./config/ora2pg.conf
[=====] 1/1 sequences (100.0%) end of output.
Running: ora2pg -p -t TRIGGER -o trigger.sql -b ./schema/triggers -c ./config/ora2pg.conf
[=====] 0/0 triggers (100.0%) end of output.
Running: ora2pg -p -t FUNCTION -o function.sql -b ./schema/functions -c ./config/ora2pg.conf
[=====] 0/0 functions (100.0%) end of functions export.
Running: ora2pg -p -t PROCEDURE -o procedure.sql -b ./schema/procedures -c ./config/ora2pg.conf
[=====] 0/0 procedures (100.0%) end of procedures export.
Running: ora2pg -p -t TABLESPACE -o tablespace.sql -b ./schema/tablespaces -c ./config/ora2pg.conf
Running: ora2pg -p -t PARTITION -o partition.sql -b ./schema/partitions -c ./config/ora2pg.conf
[=====] 0/0 partitions (100.0%) end of output.
Running: ora2pg -p -t TYPE -o type.sql -b ./schema/types -c ./config/ora2pg.conf
[=====] 0/0 types (100.0%) end of output.
Running: ora2pg -p -t MVIEW -o mview.sql -b ./schema/mviews -c ./config/ora2pg.conf
[=====] 0/0 materialized views (100.0%) end of output.
Running: ora2pg -p -t DBLINK -o dblink.sql -b ./schema/dblinks -c ./config/ora2pg.conf
[=====] 0/0 dblink (100.0%) end of output.
Running: ora2pg -p -t SYNONYM -o synonym.sql -b ./schema/synonyms -c ./config/ora2pg.conf
[=====] 0/0 synonyms (100.0%) end of output.
Running: ora2pg -p -t DIRECTORY -o directorie.sql -b ./schema/directories -c ./config/ora2pg.conf
[=====] 1/1 directory (100.0%) end of output.
Running: ora2pg -t PACKAGE -o package.sql -b ./sources/packages -c ./config/ora2pg.conf
[=====] 0/0 packages (100.0%) end of output.
Running: ora2pg -t VIEW -o view.sql -b ./sources/views -c ./config/ora2pg.conf
[=====] 0/0 views (100.0%) end of output.
Running: ora2pg -t TRIGGER -o trigger.sql -b ./sources/triggers -c ./config/ora2pg.conf
[=====] 0/0 triggers (100.0%) end of output.
Running: ora2pg -t FUNCTION -o function.sql -b ./sources/functions -c ./config/ora2pg.conf
[=====] 0/0 functions (100.0%) end of functions export.
Running: ora2pg -t PROCEDURE -o procedure.sql -b ./sources/procedures -c ./config/ora2pg.conf
[=====] 0/0 procedures (100.0%) end of procedures export.
Running: ora2pg -t PARTITION -o partition.sql -b ./sources/partitions -c ./config/ora2pg.conf
[=====] 0/0 partitions (100.0%) end of output.
Running: ora2pg -t TYPE -o type.sql -b ./sources/types -c ./config/ora2pg.conf
[=====] 0/0 types (100.0%) end of output.
Running: ora2pg -t MVIEW -o mview.sql -b ./sources/mviews -c ./config/ora2pg.conf
[=====] 0/0 materialized views (100.0%) end of output.

```

## 4.2 Créer les partitions (facultatif)

Si le schéma Oracle contient des tables partitionnées, créer les partitions avec la commande :

```
ora2pg -p -t PARTITION -o partition.sql -b ./schema/partitions -c ./config/ora2pg.conf
```

Exemple pour remplir une table partitionnée : `ora2pg -c config/ora2pg.conf -t COPY --pg_dsn "dbi:Pg:dbname=frm; host=dupglfrm001;port=5432" --pg_user postgres -J 8 -j 3 -L 20000 -a 'TABLE[SERVICE_LOG_ENTRY]'`

## 4.3 Extraire les données (facultatif)

```
ora2pg -t COPY -o data.sql -b ./data -c ./config/ora2pg.conf
```

<!> Calculer les statistiques de la base oracle avant<!>

## 4.4 Configuration

```

schema/tables/INDEXES_table.sql
schema/tables/CONSTRAINTS_table.sql
schema/tables/FKEYS_table.sql
SET DEFAULT_TABLESPACE = tbs_forma_indx;
schema/tablespaces/tablespace.sql

CREATE TABLESPACE tbs_forma_data LOCATION '/pgdata/tbs_forma_data';
CREATE TABLESPACE tbs_forma_indx LOCATION '/pgindx/tbs_forma_indx';
CREATE TABLESPACE temp LOCATION '/pgtmp/pgsql_tmp';

GRANT ALL ON TABLESPACE tbs_forma_data TO forma_admin;
GRANT ALL ON TABLESPACE tbs_forma_indx TO forma_admin;
GRANT ALL ON TABLESPACE temp TO forma_admin;

schema/directories/directorie.sql

CREATE EXTENSION external_file;
CREATE ROLE exp_full_database;
CREATE ROLE imp_full_database;
ALTER DATABASE frm SET search_path="$user",public,forma_admin,external_file;

```

## 4.5 Import

Vérifier : `echo $PERL5LIB (/opt/oracle/ora2pg-20.0)`

```

- régénérer le script de création des séquences ;
ora2pg -p -t SEQUENCE -o sequence.sql -b ./schema/sequences -c ./config/ora2pg.conf

- régénérer le script de création des privilèges ;
ora2pg -p -t GRANT -o grant.sql -b ./schema/grants -c ./config/ora2pg.conf

- ajouter le privilège USAGE dans les commandes du type : GRANT SELECT ON SEQUENCE <schema>.<sequence> TO
<propriétaire>;

```

```
cd /app/migration/dmd_mig
```

```
./import_all.sh -h pupgldmd001 -d dmd -o postgres -D
```

```

Would you like to create the database dmd? [y/N/q] y
createdb -E LATIN1 --owner postgres dmd
Would you like to drop the database dmd before recreate it? [y/N/q] N
Would you like to import TABLE from ./schema/tables/table.sql? [y/N/q] y
psql --single-transaction -U postgres -d dmd -f ./schema/tables/table.sql
Would you like to import SEQUENCE from ./schema/sequences/sequence.sql? [y/N/q] y
psql --single-transaction -U postgres -d dmd -f ./schema/sequences/sequence.sql
Would you like to import DIRECTORY from ./schema/directories/directorie.sql? [y/N/q] N voir https://github.com
/darold/external_file
Would you like to import GRANT from ./schema/grants/grant.sql? [y/N/q] y
psql -U postgres -d dmd -f ./schema/grants/grant.sql
Would you like to import TABLESPACE from ./schema/tablespaces/tablespace.sql? [y/N/q] N
CREATE TABLESPACE tbs_dmd_data_01 LOCATION '/oradata/dmd/TBS_DMD_DATA_01';
CREATE TABLESPACE tbs_dmd_indx_01 LOCATION '/oradata/dmd/TBS_DMD_INDX_01';
CREATE TABLESPACE users LOCATION '/oradata/dmd/USERS';
Would you like to process indexes and constraints before loading data? [y/N/q] N
Would you like to import data from Oracle database directly into PostgreSQL? [y/N/q] y
ora2pg -c config/ora2pg.conf -t COPY --pg_dsn "dbi:Pg:dbname=dmd;host=duorapoc002;port=5432" --pg_user postgres
optimisé :

ora2pg -c config/ora2pg.conf -t COPY --pg_dsn "dbi:Pg:dbname=atd;host=pupglatd001;port=5432" --pg_user postgres -L 40000 -J 8 -j 3 -e
'TABLE[SCHEMA_VERSION]'

ora2pg -c config/ora2pg.conf -t COPY --pg_dsn "dbi:Pg:dbname=atd;host=pupglatd001;port=5432" --pg_user postgres -a 'TABLE
[SCHEMA_VERSION]'

Would you like to import indexes from ./schema/tables/INDEXES_table.sql? [y/N/q] y
psql -U postgres -d dmd -f ./schema/tables/INDEXES_table.sql
Would you like to import constraints from ./schema/tables/CONSTRAINTS_table.sql? [y/N/q] y
psql -U postgres -d dmd -f ./schema/tables/CONSTRAINTS_table.sql
Would you like to import foreign keys from ./schema/tables/FKEYS_table.sql? [y/N/q] y
psql -U postgres -d dmd -f ./schema/tables/FKEYS_table.sql

```

Modifier les mots de passe dans la base postgres



```
ALTER USER postgres PASSWORD 'xxxx';
ALTER USER dmd_admin PASSWORD 'xxxx';
ALTER USER dmd_user PASSWORD 'xxx';
ALTER USER dmd_maj PASSWORD 'xxx';
```

```
dmd=# GRANT USAGE ON SCHEMA dmd_admin TO dmd_admin;
dmd=# GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA dmd_admin TO dmd_admin;
dmd=# grant select,delete,insert,update on all tables in schema dmd_admin to dmd_admin;
```

```
dmd=# GRANT USAGE ON SCHEMA dmd_admin TO dmd_maj;
dmd=# GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA dmd_admin TO dmd_maj;
dmd=# grant select,delete,insert,update on all tables in schema dmd_admin to dmd_maj;
```

```
dmd=# GRANT USAGE ON SCHEMA dmd_admin TO dmd_user;
dmd=# GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA dmd_admin TO dmd_user;
dmd=# grant select,delete,insert,update on all tables in schema dmd_admin to dmd_user;
```

Après la migration :

Créer le rôle zabbix :

```
CREATE ROLE zbx_monitor;
ALTER ROLE zbx_monitor WITH NOSUPERUSER INHERIT NOCREATEROLE NOCREATEDB LOGIN NOREPLICATION NOBYPASSRLS PASSWORD
'<mdp>';
GRANT pg_monitor TO zbx_monitor GRANTED BY postgres;
```

Activer la supervision :

```
# systemctl start zabbix-agent.service
# service puppet start
```

## 4.6 Refresh

```
cp ./schema/tables/INDEXES_table.sql ./schema/tables/old_INDEXES_table.sql
cp ./schema/tables/FKEYS_table.sql ./schema/tables/old_FKEYS_table.sql
cp ./schema/tables/table.sql ./schema/tables/old_table.sql
cp ./schema/tables/CONSTRAINTS_table.sql ./schema/tables/old_CONSTRAINTS_table.sql
cp ./schema/grants/grant.sql ./schema/grants/old_grant.sql
cp ./schema/tablespaces/tablespace.sql ./schema/tablespaces/old_tablespace.sql
cp ./schema/sequences/sequence.sql ./schema/sequences/old_sequence.sql
cp ./schema/directories/directorie.sql ./schema/directories/old_directorie.sql
```

```
ora2pg -p -t TABLE -o table.sql -b ./schema/tables -c ./config/ora2pg.conf
ora2pg -p -t GRANT -o grant.sql -b ./schema/grants -c ./config/ora2pg.conf
ora2pg -p -t TABLESPACE -o tablespace.sql -b ./schema/tablespaces -c ./config/ora2pg.conf
ora2pg -p -t SEQUENCE -o sequence.sql -b ./schema/sequences -c ./config/ora2pg.conf
```

```
MODIFIER tablespace.sql, CONSTRAINTS_table.sql, INDEXES_table.sql
```

```
AJOUTER SET DEFAULT_TABLESPACE = tbs_<bdd>_indx;
```

Récupérer la séquence sur Oracle :

```
SELECT sequence_name, last_number
FROM user_sequences;
```

Mettre à jour la séquence sur Postgres :

```
SELECT setval('<nom_sequence>', <last_number>, true);
```

Exemple de lancement en nohup :

```
nohup ora2pg -t COPY -J 10 -j 20 -L 50000 -c ora2pg.conf -a "<Table Names>" > /log/file 2>&1 &
```

## 6 - Problème durant l'import

```
-> exécuter ora2pg -t COPY -o data.sql -b ./data -c ./config/ora2pg.conf
-> vérifier les types de la table qui pose problème (exemple pour la base atd : ATD_ADMIN.PAYMENT_INFO)
-> voir si la table a des lignes (TRUNCATE bigtable, fattable RESTART IDENTITY;)
-> vider les tables
-> relancer l'import
```

## 7 - Jira

N° Jira	Description
<a href="#">ARCHBO-12323</a> - Détails de la demande en cours d'obtention... <b>ÉTAT</b>	Création d'un script pour compter les lignes migrées
<a href="#">ARCHBO-12245</a> - Détails de la demande en cours d'obtention... <b>ÉTAT</b>	Ajout des bases postgres à la sauvegarde et aux clonages
<a href="#">ARCHBO-12139</a> - Détails de la demande en cours d'obtention... <b>ÉTAT</b>	MAJ des MDP dans KEEPPASS

## 8 - Sources

rpm : G:\IT\PRODUCTION\ARCHITECTURE BACK OFFICE\\_COMMUN\_ARCHITECTURE BACK OFFICE\Espace Commun\Sources

<https://github.com/darold/ora2pg/releases>, ora2pg

## 8 - Bibliographie



<http://ora2pg.darold.net/documentation.html#Requirement>, prérequis d'ora2pg  
<https://metacpan.org/pod/DBD::Oracle> , DBD Oracle  
<https://metacpan.org/release/DBD-Pg> , DBD Pg  
<https://github.com/orafce/orafce> , orafce  
<https://github.com/pgbackrest/pgbackrest> , sauvegarde incrémentale postgres

## FAQ

```
[=====>] 450/444 rows (101.4%) Table ATD_ADMIN.PAYMENT_INFO (450 recs/sec)
[===== ] 1580916/1583068 total rows (99.9%) - (51 sec., avg: 30998 recs/sec).
DBD::Pg::db pg_putcopyend failed: ERREUR: syntaxe en entre invalide pour l'entier : 487.88 30994 recs/sec)
CONTEXT: COPY release_notice, ligne 22, colonne act_amount : 487.88 at /usr/local/share/perl/5.20.2/Ora2Pg.pm line 14497.
FATAL: ERREUR: syntaxe en entre invalide pour l'entier : 487.88
CONTEXT: COPY release_notice, ligne 22, colonne act_amount : 487.88
```

double precision doit être remplacé par DECIMAL(p,s) (ici DECIMAL(15,2))

=> il faut supprimer et recréer a table avec le bon type.

voir <http://www.sqlines.com/oracle-to-postgresql>

1 vidage des tables

```

TRUNCATE
atd_admin.activity ,
atd_admin.activity_log ,
atd_admin.activity_release_notice,
atd_admin.app_config ,
atd_admin.audit_entry ,
atd_admin.audit_event ,
atd_admin.batch_execution ,
atd_admin.contact ,
atd_admin.dossier ,
atd_admin.notice_info ,
atd_admin.parameter ,
atd_admin.partner_logs_entry ,
atd_admin.payment_info ,
atd_admin.release_notice ,
atd_admin.return_file ,
atd_admin.return_file_article_50 ,
atd_admin.return_file_article_59 ,
atd_admin.return_file_article_60 ,
atd_admin.return_item ,
atd_admin.role ,
atd_admin.role_permission ,
atd_admin.schema_version ,
atd_admin.seizure_notice ,
atd_admin.seizure_notice_item ,
atd_admin.seizure_notice_total ,
atd_admin.user_agent RESTART IDENTITY;

```

Corriger le type des colonnes :

Par défaut ora2pg choisit le type BIGINT qui n'est pas adapté pour les colonnes suivantes :

```

ALTER TABLE atd_admin.payment_info ALTER amount TYPE DECIMAL(15,2);
ALTER TABLE atd_admin.payment_info ALTER release_amount TYPE DECIMAL(15,2);
ALTER TABLE atd_admin.partner_logs_entry ALTER response_message TYPE VARCHAR(6000);
ALTER TABLE atd_admin.release_notice ALTER act_amount TYPE DOUBLE PRECISION;
ALTER TABLE atd_admin.return_file_article_50 ALTER act_amount TYPE DOUBLE PRECISION;
ALTER TABLE atd_admin.return_file_article_60 ALTER act_amount TYPE DOUBLE PRECISION;
ALTER TABLE atd_admin.seizure_notice_item ALTER act_amount TYPE DOUBLE PRECISION;

```

Faire l'import des données table par table :

```

postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.RETURN_FILE_ARTICLE_50_data.sql
TRUNCATE TABLE
COPY 151
postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.RETURN_FILE_ARTICLE_59_data.sql
TRUNCATE TABLE
COPY 11
postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.RETURN_FILE_ARTICLE_60_data.sql
TRUNCATE TABLE
COPY 437
postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.RETURN_FILE_data.sql
TRUNCATE TABLE
COPY 599
postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.ROLE_PERMISSION_data.sql
TRUNCATE TABLE
COPY 46
postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.SEIZURE_NOTICE_data.sql
TRUNCATE TABLE
COPY 14
postgres@vupglpoc001:/pgdata/migration/atd_mig/data$ psql atd < ATD_ADMIN.SEIZURE_NOTICE_ITEM_data.sql

```

ETC...

si erreur :

```

Can't locate DBI.pm in @INC (you may need to install the DBI module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.28.1 /usr/local/share
/perl/5.28.1 /usr/lib/x86_64-linux-gnu/perl5/5.28 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/perl/5.28 /usr/share/perl/5.28 /usr/local/lib/site_perl /usr/lib
/x86_64-linux-gnu/perl-base) at /usr/local/share/perl/5.28.1/Ora2Pg.pm line 29.
BEGIN failed--compilation aborted at /usr/local/share/perl/5.28.1/Ora2Pg.pm line 29.
Compilation failed in require at /usr/local/bin/ora2pg line 27.
BEGIN failed--compilation aborted at /usr/local/bin/ora2pg line 27.

```

=> réinstaller les modules perl

DBI, DBD-Oracle et DBD-Pg