

Ora2pg 使用总结

| | |
|-------------------------------|---|
| Ora2pg 使用总结..... | 1 |
| 1 Ora2pg 特性..... | 1 |
| 2 Ora2pg 支持的导出对象..... | 2 |
| 3 Ora2pg 数据类型转换对照..... | 2 |
| 4 Ora2pg 安装..... | 3 |
| 4.1 安装需要的驱动包..... | 3 |
| 4.2 下载并安装 Oracle 客户端..... | 3 |
| 4.3 安装 DBI..... | 4 |
| 4.4 安装 DBD-Oracle 客户端..... | 4 |
| 4.4.1 在当前用户配置一下环境变量..... | 4 |
| 4.4.2 安装 DBD-Oracle..... | 4 |
| 4.5 安装 Ora2pg 客户端..... | 4 |
| 4.6 ora2pg 参数说明..... | 5 |
| 5 Ora2pg 使用案例..... | 6 |
| 5.1 ora2pg 数据导入到 pg 案例..... | 6 |
| 5.1.1 编写配置案例..... | 6 |
| 5.1.2 使用 ora2pg 把数据下载到本地..... | 7 |
| 5.1.3 查看文件的大小与行数..... | 8 |
| 5.1.4 把数据导入到 postgres 中..... | 8 |
| 5.1.5 校验 pg 中数据的准确性..... | 8 |
| 6 把 PG 数据加载到 GP 中..... | 9 |
| 6.1 把 postgres 的数据下载到磁盘..... | 9 |
| 6.2 把磁盘上的数据加载到 GP 的数据库中..... | 9 |
| 6.3 在 GP 中修改表的分布键..... | 9 |

1 Ora2pg 特性

- 1、导出整个数据库模式（表、视图、序列、索引），以及唯一性，主键、外键和检查约束。
- 2、导出用户和组的授权/权限。
- 3、导出筛选的表（通过制定表明）。
- 4、导出 Oracle 模式到一个 PostgreSQL（7.3 以后）模式中。
- 5、导出预定义函数、触发器、程序、包和包体。
- 6、导出范围和列表分区。
- 7、导出所有的数据或跟随一个 WHERE 子句。
- 8、充分支持 Oracle BLOB 对象作为 PG 的 BYTEA。
- 9、导出 Oracle 视图作为 PG 表。
- 10、导出定义的 Oracle 用户格式。

- 11、提供关于转换 PLSQL 码为 PLPGSQL 的基本帮助（仍然需要手工完成）。
- 12、可在任何平台上工作。
- 13、Ora2Pg 尽力将 Oracle 数据库转换到 PostgreSQL 中，但是仍需一部分的手动工作。Oracle 特定的 PL/SQL 代码生成函数、过程 和触发器时必须进行审查，以便匹配 PostgreSQL 的语法

2 Ora2pg 支持的导出对象

这是允许导出的不同的格式，默认是 TABLE:

| | |
|------------|-------------------------------------|
| table | 提取所有包括索引、主键、唯一键、外键和检查约束的表。 |
| view | 提取视图。 |
| grant | 提取在所有对象中转换为 pg 组、用户和权限的用户。 |
| sequence | 提取所有的序列以及上一个位置。 |
| tablespace | 提取表空间。 |
| trigger | 提取通过动作触发的被指定的触发器。 |
| function | 提取函数。 |
| proceduers | 提取存储过程。 |
| package | 提取包和包主体。 |
| data | 提取数据，生成 insert 语句。 |
| copy | 提取数据，生成 copy 语句。 |
| partition | 提取范围和列表分区。 |
| type | 提取 oracle 用户自定义的格式。（以下两条是 10.0 新加的） |
| fdw | 提取外部数据封装表 |
| partition | 提取作为快照刷新视图所建立的视图 |

3 Ora2pg 数据类型转换对照

| oracle 类型 | postgresql 类型 |
|-----------|---------------|
| date | timestamp |
| long | text |
| long raw | bytea |
| clob | text |
| nclob | text |
| blob | bytea |
| bfile | bytea |
| raw | bytea |
| rowid | oid |

| | |
|--------------------------------|--------------------------|
| float | double precision |
| dec | decimal |
| decimal | decimal |
| double precision | double precision |
| int | integer |
| integer | integer |
| real | real |
| smallint | smallint |
| binary_float | double precision |
| binary_double | double precision |
| timestamp | timestamp |
| xmltype | xml |
| binary_integer | integer |
| pls_integer | integer |
| timestamp with time zone | timestamp with time zone |
| timestamp with local time zone | timestamp with time zone |

4 Ora2pg 安装

4.1 安装需要的驱动包

```
yum install -y perf cpan
```

4.2 下载并安装 Oracle 客户端

oracle 客户端下载地址(下载 basic/sqlplus/devel/jdbc 后缀为 rpm 文件即可)

<https://www.oracle.com/database/technologies/instant-client/downloads.html>

<https://www.oracle.com/database/technologies/instant-client/linux-x86-64-downloads.html>

安装下载的软件包

```
scp -r rpm -ivh oracle-instantclient19.3-basic-19.3.0.0.0-1.x86_64.rpm
```

```
scp -r rpm -ivh oracle-instantclient19.3-devel-19.3.0.0.0-1.x86_64.rpm
```

```
scp -r rpm -ivh oracle-instantclient19.3-sqlplus-19.3.0.0.0-1.x86_64.rpm
```

测试客户端

sqlplus64 username/password@ip:port/sid 或 service_name

4.3 安装 DBI

```
cpan install -y DBI
```

4.4 安装 DBD-Oracle 客户端

4.4.1 在当前用户配置一下环境变量

```
export ORACLE_HOME=/usr/lib/oracle/19.3/client64
export PATH=$ORACLE_HOME/bin:$PATH
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
```

19.3 安装的 oracle 客户端的版本号

4.4.2 安装 DBD-Oracle

```
wget http://search.cpan.org/CPAN/authors/id/P/PY/PYTHIAN/DBD-Oracle-1.74.tar.gz
tar -zxvf DBD-Oracle-1.74.tar.gz
```

```
cd DBD-Oracle-1.74
```

```
perl Makefile.PL -I
make && make test
make install
```

4.5 安装 Ora2pg 客户端

```
wget https://github.com/darold/ora2pg/archive/v18.2.tar.gz
```

```
tar -zxvf v18.2.tar.gz
```

```
cd ora2pg-18.2/
```

```
perl Makefile.PL
make && make install
```

4.6 ora2pg 参数说明

```
# ora2pg --help
```

```
Usage: ora2pg [-dhpqv --estimate_cost --dump_as_html] [--option value]
```

- a | --allow str : 指定允许导出的对象列表，使用逗号分隔。也可以与 SHOW_COLUMN 选项一起使用。
- b | --basedir dir: 设置默认的导出目录，用于存储导出结果。
- c | --conf file : 设置非默认的配置文件，默认配置文件为 /etc/ora2pg/ora2pg.conf。
- d | --debug : 使用调试模式，输出更多详细信息。
- D | --data_type STR: 通过命令行设置数据类型转换。
- e | --exclude str: 指定导出时排除的对象列表，使用逗号分隔。也可以与 SHOW_COLUMN 选项一起使用。
- h | --help : 显示帮助信息。
- g | --grant_object type: 导出指定类型的对象上的授权信息，取值参见 GRANT_OBJECT 配置项。
- i | --input file: 指定要导入的 Oracle PL/SQL 代码文件，导入文件时不需要连接到 Oracle 数据库。
- j | --jobs num : 设置用于发送数据到 PostgreSQL 的并发进程数量。
- J | --copies num: 设置用于从 Oracle 导出数据的并发连接数量。
- l | --log file : 设置日志文件，默认为 stdout。
- L | --limit num : 导出数据时，每次写入磁盘之前在内存中缓冲的记录数量，默认值为 10000。
- m | --mysql : 导出 MySQL 数据库。
- n | --namespace schema: 设置需要导出的 Oracle 模式。
- N | --pg_schema schema: 设置 PostgreSQL 中的搜索路径 search_path。
- o | --out file : 设置导出的 SQL 文件的存储路径。默认值为当前目录下的 output.sql 文件。
- p | --plsql : 启用 PLSQL 代码到 PLPGSQL 代码的转换。
- P | --parallel num: 同时导出多个表，设置并发数量。
- q | --quiet : 不显示进度条。
- s | --source DSN: 设置 Oracle DBI 数据源。
- t | --type export: 设置导出类型。该参数将会覆盖配置文件中的导出类型 (TYPE)。
- T | --temp_dir DIR: 为多个同时运行的 ora2pg 脚本指定不同的临时存储目录。
- u | --user name : 设置连接 Oracle 数据库连接的用户名。也可以使用 ORA2PG_USER 环境变量。
- v | --version : 显示 Ora2Pg 版本信息并退出。
- w | --password pwd: 设置连接 Oracle 数据库的用户密码。也可以使用 ORA2PG_PASSWD 环境变量。

`--forceowner` : 导入数据时, 强制 ora2pg 将导入 PostgreSQL 的表和序列的拥有者设置为连接 Oracle 数据库时的用户。如果设置为指定的用户名, 所有导入的对象属于该用户。默认情况下, 对象的拥有者为连接 Pg 数据库的用户。

`--nls_lang code`: 设置 Oracle 客户端的 NLS_LANG 编码。

`--client_encoding code`: 设置 PostgreSQL 客户端编码。

`--view_as_table str`: 将视图导出为表, 多个视图使用逗号分隔。

`--estimate_cost` : 在 SHOW_REPORT 结果中输出迁移成本评估信息。

`--cost_unit_value minutes`: 成本评估单位, 使用分钟数表示。默认值为 5 分钟, 表示一个 PostgreSQL 专家迁移所需的时间。如果是第一次迁移, 可以设置为 10 分钟。

`--dump_as_html` : 生成 HTML 格式的迁移报告, 只能与 SHOW_REPORT 选项一起使用。默认的报告是一个简单的文本文件。

`--dump_as_csv` : 与上个参数相同, 但是生成 CSV 格式的报告。

`--dump_as_sheet` : 生成迁移评估时, 为每个数据库生成一行 CSV 记录。

`--init_project NAME`: 创建一个 ora2pg 项目目录结构。项目的顶级目录位于根目录之下。

`--project_base DIR`: 定义 ora2pg 项目的根目录, 默认为当前目录。

`--print_header` : 与 `--dump_as_sheet` 一起使用, 输出 CSV 标题信息。

`--human_days_limit num`: 设置迁移评估级别从 B 升到 C 所需的人工日数量。默认值为 5 人工日。

`--audit_user LIST` : 设置查询 DBA_AUDIT_TRAIL 表时需要过滤的用户名, 多个用户使用逗号分隔。该参数只能用于 SHOW_REPORT 和 QUERY 导出类型。

`--pg_dsn DSN` : 设置在线导入时的 PostgreSQL 数据源。

`--pg_user name` : 设置连接 PostgreSQL 的用户名。

`--pg_pwd password` : 设置连接 PostgreSQL 的用户密码。

`--count_rows` : 在 TEST 方式下执行真实的数据行数统计。

`--no_header` : 在导出文件中不添加 Ora2Pg 头部信息。

`--oracle_speed` : 用于测试 Oracle 发送数据的速度。不会真的处理或者写入数据。

`--ora2pg_speed` : 用于测试 Ora2Pg 发送转换后的数据的速度。不会写入任何数据。

5 Ora2pg 使用案例

5.1 ora2pg 数据导入到 pg 案例

5.1.1 编写配置案例

```
# vim ora2pg.conf
```

```
# oracle client 安装地址
```

```
ORACLE_HOME /usr/lib/oracle/19.3/client64
```

```
# oracle connent info
```

```
ORACLE_DSN dbi:Oracle:host=ipaddress;sid=orcl;port=1521
```

ORACLE_USER username
ORACLE_PWD password

oracle schema
SCHEMA schemaname

oracle type
TYPE TABLE COPY DATA

输出的文件名字
OUTPUT output.sql

输出的路径(本地路径)
OUTPUT_DIR ./

Ipaddress : 链接 oracle 的 IP 地址
username : 链接 oracle 的用户名
password : 链接 oracle 的密码
schemaname : 链接 oracle 的 schema 信息

5.1.2 使用 ora2pg 把数据下载到本地

time ora2pg -c ora2pg.conf -a **tablename**

```
[=====>] 1/1 tables (100.0%) end of scanning.  
[>] 0/1 tables (0.0%) end of scanning.  
[>] 0/1 tables (0.0%) end of scanning.  
[=====>] 1/1 tables (100.0%) end of table export.  
[=====>] 541243/537749 rows (100.6%) Table tablename (7959  
recs/sec)  
[=====>] 541243/537749 total rows (100.6%) - (68 sec., avg: 7959  
recs/sec).  
[=====>] 537749/537749 rows (100.0%) on total estimated data (68 sec.,  
avg: 7908 recs/sec)  
[=====>] 541243/537749 rows (100.6%) Table tablename (8591 recs/sec)  
[=====>] 1082486/537749 total rows (201.3%) - (63 sec., avg: 17182  
recs/sec).  
[=====>] 537749/537749 rows (100.0%) on total estimated data (63 sec.,  
avg: 8535 recs/sec)  
  
real 2m26.185s  
user 2m9.606s  
sys 0m1.153s
```

tablename: 单表的名字

5.1.3 查看文件的大小与行数

```
# du -sh output.sql
79M output.sql

# wc -l output.sql
542914 output.sql
```

5.1.4 把数据导入到 postgres 中

```
# time psql -U postgres -d databasename -h 192.168.***.** -p 5432 -f output.sql
SET
SET
CREATE TABLE
CREATE INDEX
ALTER TABLE
SET
SET
BEGIN
ALTER SEQUENCE
ALTER SEQUENCE
COMMIT

real 0m10.196s
user 0m0.215s
sys 0m0.202s
```

databasename : 数据库的名字

5.1.5 校验 pg 中数据的准确性

```
select count(*) from "public".tablename;
-- 541243
```


tablename : 表的名字

6 把 PG 数据加载到 GP 中

6.1 把 postgres 的数据下载到磁盘

```
psql -d chinadaas -h 192.168.***.** -p 5432 -U postgres -c "copy tablename to 'filepath' WITH  
DELIMITER AS E'\u0001' NULL as 'null string' "
```

6.2 把磁盘上的数据加载到 GP 的数据库

```
psql -d staging -h 192.168.***.** -p 5432 -U gpadmin -c "copy tablename from 'filepath' WITH  
DELIMITER AS E'\u0001' NULL as 'null string' "
```

6.3 在 GP 中修改表的分布键

```
alter table tablename set with(REORGANIZE=true)  
distributed by(filename);
```