

PostgreSQL与Oracle的差异对比

原创 Pierre_ 2016-08-21 18:07:30 27644 收藏 13

版权

分类专栏: [Oracle](#) [postgresql](#)

Postgresql 与 Oracle 相关差异对比

通过查阅资料及实验，以下对比了 Postgresql 9.3 版本与 Oracle 11g版本的相关差异。**注意：相关细节仍待考证和完善。**

1. 基本语法差异

1.1. 基本数据类型差异

Oracle	Postgresql
Varchar2	varchar
DATE	date/time/timestamp
null	null
clob	text
blob	bytea
number	smallint/integer/bigint/numeric/real/double precision
不支持boolean，可通过0/1代替	支持 boolean

1.2. 基本函数差异

Item	Oracle	Postgresql
字符串连接符		concat()
'a' null =	'a'	null
trunc(时间)	trunc(date)	date_trunc()
取当前系统时间	SYSDATE	localtimestamp, now()
to_char, to_number, to_date	自动格式转换	需指定格式
decode	√	×
outer join	(+)	left(right) join
GOTO	√	×
procedure	√	× 需改写成function
package	√	× 需改写成function

1.2.1. 游标属性

Oracle	Postgresql
%FOUND	found
%NOTFOUND	not found
%ISOPEN	×
%ROWCOUNT	×

1.2.2. 系统内置函数包

Oracle	Postgresql
DBMS_OUTPUT	×
DBMS_SQL	×
UTIL_FILE	×
UTIL_MAIL	×

1.3. DDL 差异

1.3.1. Sequence语法及使用差异

Item	Oracle	Postgresql
create	<pre>create sequence SEQ_TAB_NAME minvalue 1 maxvalue 9999999999999999 start with 1 increment by 1 cache 20;</pre>	<pre>create sequence seq_tab_name minvalue 1 maxvalue 9223372036854775807 start 1 increment 1 cache 20;</pre>
query	<pre>select seq_tab_name.nextval from dual;</pre>	<pre>select next_val(seq_tab_name) from dual;</pre>

注意：pgsql中的 dual，需自主实现。详见[兼容性设置->虚表dual问题](#) 章节。

1.3.2. constraint 差异

Item	Oracle	Postgresql
constraint	alter table tab_name add constraint pk_tab_name primary key(column_id) using index;	alter table tab_name add constraint pk_tab_name primary key(column_id);

2. 高级语法差异

2.1. 事务差异

Oracle 中，通过 commit/rollback来实现事务提交或回滚。结构类似于：

```
1  begin
2      select ...
3      update ...
4      delete ...
5
6      commit;
7  exception
8      when others then
9          rollback;
10 end;
```

PostgreSQL 实际上把每个 SQL 语句当做在一个事务中执行来看待。如果你没有发出BEGIN命令，那么每个独立的语句都被一个隐含的BEGIN 和 (如果成功的话)COMMIT包围。一组包围在BEGIN和COMMIT 之间的语句有时候被称做事务块。

例如：

```
1  BEGIN;
2  UPDATE accounts SET balance = balance - 100.00
```

```
3      WHERE name = 'Alice';
4  SAVEPOINT my_savepoint;
5  UPDATE accounts SET balance = balance + 100.00
6      WHERE name = 'Bob';
7  -- 呀! 加错钱了, 应该用 Wally 的账号
8  ROLLBACK TO my_savepoint;
9  UPDATE accounts SET balance = balance + 100.00
10     WHERE name = 'Wally';
11  COMMIT;
12
```

在 PL/pgSQL中, 也提供了类似于Oracle 的 Begin、End及ExceptionCode处理机制。他们之间的差异不明显。事实上, PL/SQL (Oracle数据库操作语言) 与PL/pgSQL是高度相似的, 这让procedure在Oracle与 Postgresql之间迁移变得极为便捷。

2.2. 函数继承与重载

Oracle不支持 继承和重载特性, postgresql支持继承和函数重载;
(待完善)

2.3. 类型转换

postgresql 中的类型转换 “::” 符, Oracle 不支持。

2.4. 子查询

子查询, postgresql要求更严格, 必须具有别名才可以;

3. 其他差异

3.1. jdbc差异

Oracle的jdbc连接字符串: `db.url=jdbc:oracle:thin:@192.168.1.1:1521:ORCL`

Postgresql的连接字符串: `db.url=jdbc:postgresql:@192.168.1.1:5432/database`

4. 兼容性设置

4.1. 字符串连接兼容性解决方案

Postgresql中没有concat函数, 且由于 || 无法使用, 需要通过在public schema中创建concat函数来解决。

```
1  --在 public schema中创建 concat 函数
2  create or replace function concat(text, text) returns text as
3  $body$ select coalesce($1,'') || coalesce($2,'')$body$ language 'sql' volatile;
4
5  alter function concat(text, text) owner to postgres;
6
```

4.2. 虚表 dual 问题

Postgresql中没有 dual 虚拟表, 为保证兼容性, 需创建伪视图 (view) 代替:

```
1  create or replace view dual as
2  select NULL::"unknown"
3  where 1=1;
4
5  alter table dual owner to postgres;
6  grant all on table dual to postgres;
7  grant select on table dual to public;
8
```

4.3. 数据分页问题

Oracle中没有 limit, postgresql中没有rownum。需重写相关语句。

```
1  -- Oracle
2  select * from (
3      select * from (
4          select * from t1 order by col1, col2
5      ) where rownum <=50 order by col3, col4
6  ) where rowmun <=20 order by col5, col6;
7
8  -- postgresql
9  select * from (select * from (select * from t1 order by col1, col2) ta order by col3, col4 limit 50) tb
10 order by col5, col6 limit 20;
```

注意: limit必须用于 order by 之后!

扩展阅读

- [从PL/SQL 向PL/pgSQL移植](#)