



Introdução

Baseado no artigo <https://www.enterprisedb.com/node/3480> e na documentação https://www.enterprisedb.com/docs/en/6.0/repguide/EDB_Postgres_Replication_Server_Users_Guide.1.38.html

Devemos fazer o procedimento a seguir para que o Odoo consiga gerar chaves primárias e estrangeiras de forma distribuída.

1. Parâmetro de configuração

Criar um parâmetro de configuração no arquivo de configuração do postgres (/etc/postgresql/10/main/postgresql.conf) para identificar o nó.

- a) # Colocar o número do nó, este número deve ser único e não pode ser repetido, variando de (1 até 127)

```
#-----  
# CUSTOMIZED OPTIONS  
#-----  
  
# Add settings for extensions here  
  
odoo.mmr_id = '1'
```

- b) # Recarregar o postgresql: /etc/init.d/postgresql restart
c) # Acessar qualquer banco de dados e validar se a configuração esta funcionando

```
psql template1  
  
template1=# odoo.mmr_id = '1'  
  
current_setting  
-----  
1  
(1 registro)
```



```
SELECT current_setting('odoo.url')
```

2. Criação das funções

```
CREATE OR REPLACE FUNCTION mmr_sequence_nextval(seq_id varchar)
  RETURNS bigint
  LANGUAGE sql
AS
  $function$
    SELECT ((SELECT current_setting('odoo.mmr_id'))::int << 24)::int +
nextval($1::regclass);
  $function$;

CREATE OR REPLACE FUNCTION mmr_sequence_currval(seq_id varchar)
  RETURNS bigint
  LANGUAGE sql
AS
  $function$
    SELECT ((SELECT current_setting('odoo.mmr_id'))::bigint << 24)::int +
currval($1::regclass);
  $function$;

CREATE OR REPLACE FUNCTION mmr_sequence_convert(old_seq_value bigint)
  RETURNS bigint
  LANGUAGE sql
AS
  $function$
    SELECT ((SELECT current_setting('odoo.mmr_id'))::bigint << 24)::bigint + $1;
  $function$;
```

```
SELECT ((SELECT current_setting('odoo.mmr_id'))::int << 24)::bit(32);
```

3. Modificação do core do Odoo

<https://github.com/kmee/OCB/commit/fb2e530c6d283ee5d4cb4bc4f0e633756fdc7a81>

4. Issues

Upgrading database, with DDL changes;