

# Organización, consulta e tratamento da información

Anterior: [Administración básica e utilización de sistemas ERP-CMR](#)

Seguinte: [Implantación de sistemas ERP/CRM](#)

## Sumario

- 1 Configuración inicial
- 2 Acceso á base de datos
- 3 Deseño de consultas
- 4 Creación de vistas na BD
- 5 Creación de vistas en OpenERP
- 6 Edición de informes con LibreOffice
- 7 Exportar datos en fomato CSV

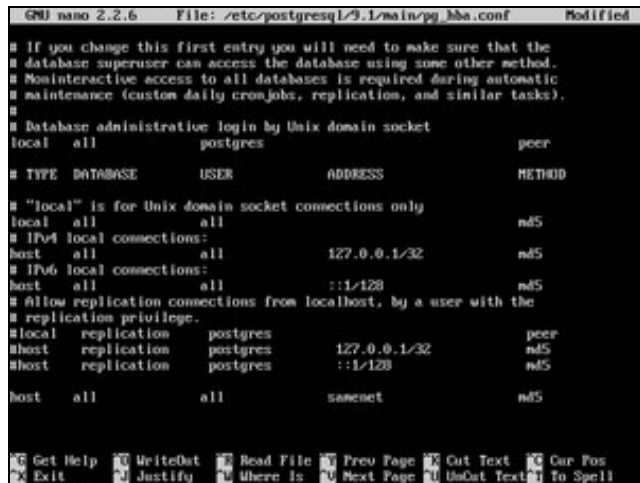
## Configuración inicial

Punto de partida: [Instalación de OpenERP 7.0 en Ubuntu Server](#)

Vamos acceder ao servidor Postgres do Ubuntu Server utilizando a aplicación [PgAdmin](#) instalada en outro equipo. Para permitir este acceso remoto é preciso realizar os pasos descritos no apartado [configuración inicial de OpenERP 6.1](#) cun par de cambios:

- Configuramos Postgresql editando *pg\_hba.conf*. Obrigamos a que os usuarios da BD se autenticuen con clave encriptada: Na liña que comeza por "local all all" trocamos a última palabra por "md5". No final do ficheiro inserimos unha nova liña para permitir as conexións desde outro equipo da subrede:

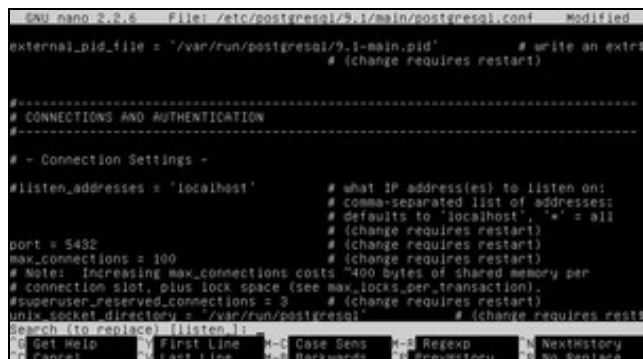
```
host    all    all    samenet    md5
```



```
GNU nano 2.2.6 File: /etc/postgresql/9.1/main/pg_hba.conf Modified
# If you change this first entry you will need to make sure that the
# database superuser can access the database using some other method.
# Noninteractive access to all databases is required during automatic
# maintenance (custom daily cronjobs, replication, and similar tasks).
#
# Database administrative login by Unix domain socket
local   all             postgres                                peer
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local   all             all                                md5
# IPv4 local connections:
host    all             all             127.0.0.1/32          md5
# IPv6 local connections:
host    all             all             ::1/128              md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
#local   replication    postgres                                peer
#host    replication    postgres            127.0.0.1/32          md5
#host    replication    postgres            ::1/128              md5
host    all             all             samenet              md5

Get Help  WriteOut  Read File  Prev Page  Cut Text   Cur Pos
Exit      Justify   Where Is   Next Page  UnOut Text To Spell
```

- Modificamos en *postgresql.conf* a liña "listen\_addresses", eliminando # (símbolo de comentario) e substituíndo *localhost* por un asterisco:



```
GNU nano 2.2.6 File: /etc/postgresql/9.1/main/postgresql.conf Modified
external_pid_file = '/var/run/postgresql/9.1-main.pid' # write an extra
# (change requires restart)

#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

#listen_addresses = 'localhost' # what IP address(es) to listen on:
#                               # comma-separated list of addresses:
#                               # defaults to 'localhost', '*' = all
#                               # (change requires restart)
port = 5432 # (change requires restart)
max_connections = 100 # (change requires restart)
# Note: Increasing max_connections costs ~400 bytes of shared memory per
# connection slot, plus lock space (see max_locks_per_transaction).
#superuser_reserved_connections = 3 # (change requires restart)
unix_socket_directory = '/var/run/postgresql' # (change requires restart)

Search (to replace) (listen):
Get Help  First Line  Case Sens  Regexp  NextHistory
Exit      Last Line   Backwards  PrevHistory  No Replace
```

Antes

```
SAO nano 2.2.6- File: /etc/postgresql/9.1/main/postgresql.conf Modified
external_pid_file = '/var/run/postgresql/9.1-main.pid'           # write an extra
                                                                # (change requires restart)

#-----
# CONNECTIONS AND AUTHENTICATION
#-----

# - Connection Settings -

listen_addresses = '*'                                          # what IP address(es) to listen on:
                                                                # comma-separated list of addresses;
                                                                # defaults to 'localhost', '*' = all
                                                                # (change requires restart)
port = 5432                                                      # (change requires restart)
max_connections = 100                                           # (change requires restart)
# Note: Increasing max_connections costs ~400 bytes of shared memory per
# connection slot, plus lock space (see max_locks_per_transaction).
#superuser_reserved_connections = 3                             # (change requires restart)
unix_socket_directory = '/var/run/postgresql'                   # (change requires restart)

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where To Next Page UnCut Text To Spell
```

Depois

- Reiniciamos Postgresql:

```
sudo /etc/init.d/postgresql restart
```

- Creamos un usuario en Postgres con permiso para crear bases de datos. No noso exemplo chámase *varela*, con clave *abc123*:

```
sudo su postgres
createuser varela --pwprompt
...(Responder as preguntas: Pode crear BDs, o resto non.)
exit
```

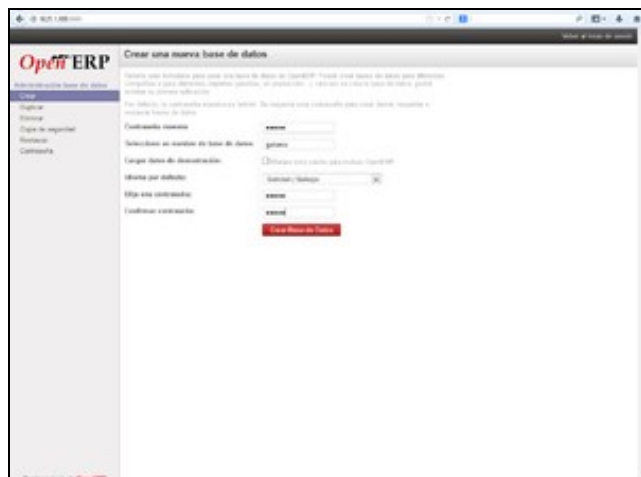
- Configuramos o servidor OpenERP para indicarlle o nome do usuario e clave creados no paso anterior:

```
sudo gedit /etc/openerp/openerp-server.conf
```

- Reiniciamos o servidor OpenERP co seguinte comando:

```
sudo /etc/init.d/openerp restart
```

- Verificar que podemos acceder desde calquera equipo da subrede utilizando un browser, indicando a IP do servidor e o porto 8069



## Acceso á base de datos

Instalamos **pgAdmin** nun equipo da subrede e realizamos os seguintes pasos:

- Accedemos a Odoo e facemos clic en "Xestionar Bases de Datos".



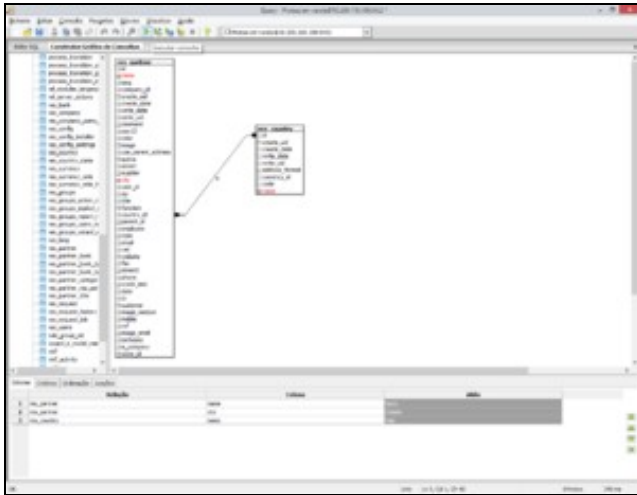
- Creamos unha base de datos de probas. Marcamos a opción "Cargar datos de demostración".



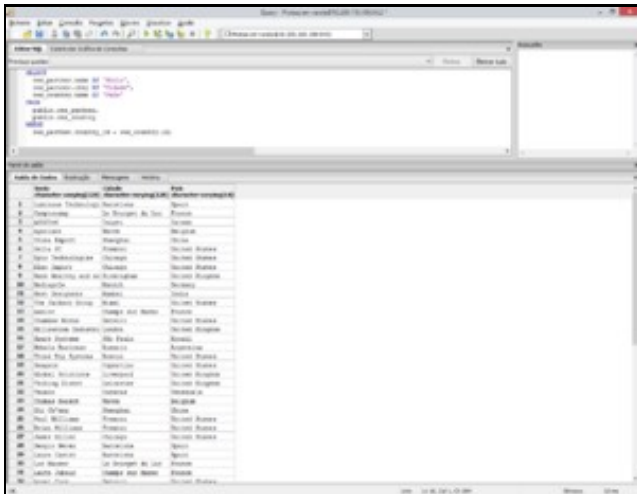
- Iniciamos pgAdmin e conectamos co servidor Postgres



- Por exemplo, queremos saber o nome dos socios e de que cidade e país son. Primeiro facemos clic nas táboas *res\_partner* e *res\_country*. Despois arrastramos a clave foránea *country\_id* para relacionala co campo *id* da táboa *res\_country*. Escollemos para visualizar os campos *name* e *city* e escribimos os alias correspondentes. Finalmente prememos en *Executar SQL*.



- Podemos ver o código SQL xerado e o resultado da consulta.



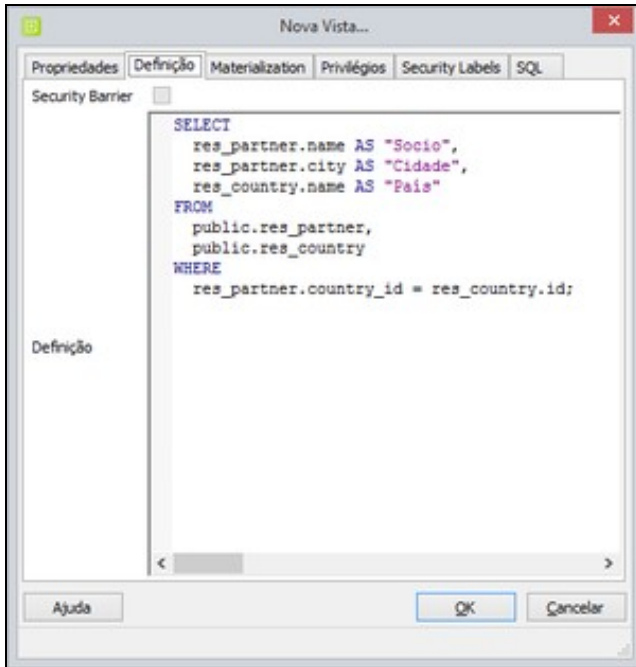
## Creación de vistas na BD

Vamos crear unha vista co código SQL da consulta anterior.

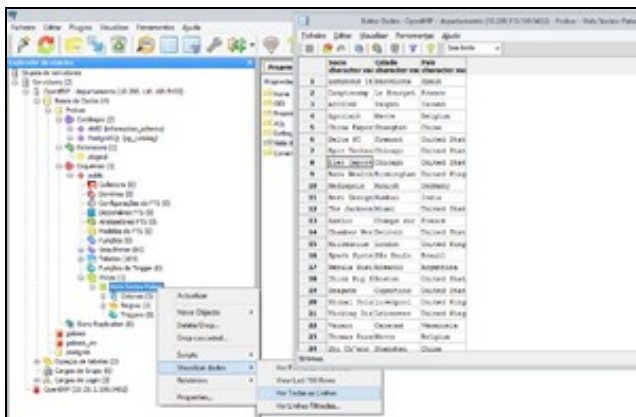
- Na pantalla inicial procuramos o apartado *Vistas* da base de datos *Probas*. Creamos unha nova vista chamada *Vista Socios-Países*.



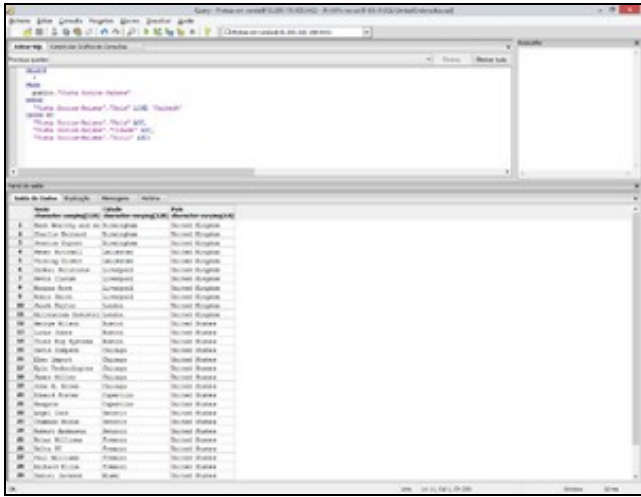
- No separador *Definição*, pegamos o código SQL da consulta.



- Agora podemos visualizar os dados da vista e trabalhar com ela como se se tratase dunha táboa.



- Por exemplo, vamos realizar unha segunda consulta cos datos da vista, que amose o resultado ordenado por país, despois por cidade e por nome. Podemos filtrar para que só vexamos os datos dos socios dos países que comezan por "United".



## Creación de vistas en OpenERP

Agora que xa sabemos utilizar pgAdmin nunha BD de probas, vamos a realizar unha práctica completa, seguindo o fío da páxina [Administración básica e utilización de sistemas ERP-CMR](#).

En OpenERP, para a empresa *Galaxia Software*, queremos crear un novo menú cunha ligazón para visualizar os datos dos socios (clientes, provedores, usuarios e a propia empresa).

En primeiro lugar creamos unha vista en OpenERP que conecte coa táboa *res.partner* seleccionando os campos que nos interese amosar.

Coa conta de administrador/a, facemos clic en *Configuración, Técnico, Interfaz de Usuario, Vistas, Crear..* Escribimos o nome do obxecto *res.partner* e o nome da vista *res.partner.vistaArbore*. Mediante código XML especificamos os campos que queremos visualizar.



Aproveitamos para duplicar a vista, editar a copia, e transformala nunha vista de formulario.







Crear: m2o

Nombre de acción:

Objeto:

Objeto origen:

Uno de la acción:

Tipo de acción:

Ventana destino:

Configuración general Seguridad

Vistas

Tipo de vista:

Modo de vista:

Ref. vista:

Ref. vista independiente:

Filtros

Valor de dominio:

Valor de contexto:

Límite:

Auto-refrescar:

Auto independiente: ☒

Filtro:

Guardar Descartar

Inicio Galería de Portal Noticias Contacto Publicaciones Recursos Más...

Galaxy SW

Inicio

Ver Series Avance

Ver Series Avance


Dataset VistaS... / KiteSw

Editar Crear Imprimir Más

8 / 11

Nombre	Proveedor	Clase	Salamanca
11		Clase	11

[Menú](#)
[Galaxia de](#)
[Portal](#)
[Ventas](#)
[Contabilidad](#)
[Purchases](#)
[Atención](#)
[Más...](#)



[Abrir Vista Socios Arbore](#)

[Inicio](#)

[Ver Socios Arbores](#)

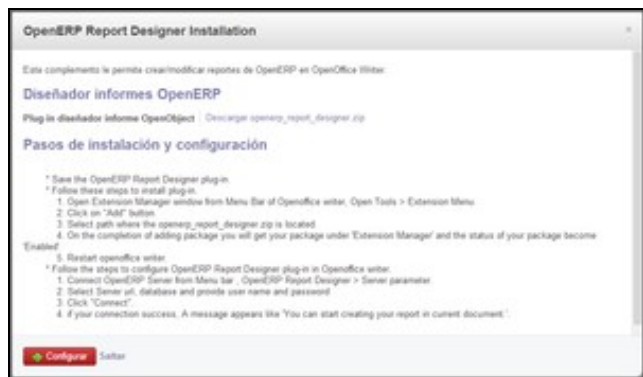
[Ver Socios Formulero](#)

Nombre	Ciudad	Proveedor	Vista Formulero
Administradora			
Ana Toranzo Guevara			
Anonymous			
emitted=682705238aa4914810db990734135a9			
Galaxia Software	Santiago de Compostela		
Kid@se			True
Manuel Martín Penuela			
MatCR	Viana do Castelo	True	
MatCR2	Ourense	True	
Sergios	Santiago de Compostela		True
Template User			

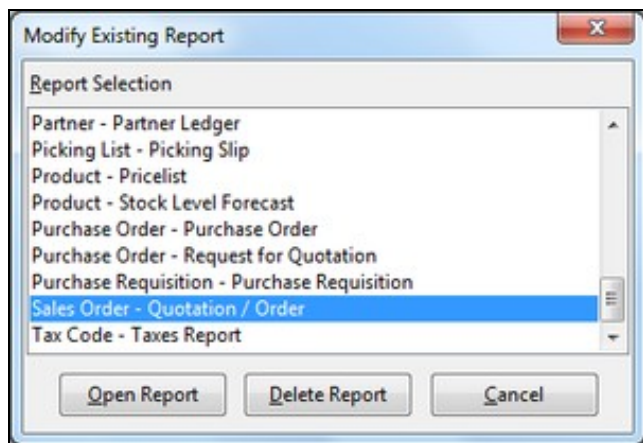
É importante non confundirmos as vistas da BD coas vistas de OpenERP.

## Edición de informes con LibreOffice

Instalamos o módulo *OpenOffice Report Designer*. Tamén **hai que descargar o plugin** en formato ZIP para Writer e seguir as instrucións de configuración.



- Iniciamos Writer.
- En *Ferramentas, Xestor de extensións, Engadir*, escollemos o ficheiro *base\_report\_designer\_installer.zip* previamente descargado. Se non ten extensión *.zip*, poñémoslla nós.
- Reiniciamos Writer. Aparece unha nova barra de ferramentas e un novo menú.
- Executamos *Server parameters* e conectámonos co servidor OpenERP utilizando a conta *admin*.
- Executamos *Modify Existing Report* e escollemos o informe de orde de vendas.



- Neste exemplo traducimos *Enderezo de envío* e *Enderezo de facturación*, pero pódense realizar outras accións como inserir campos.



- 

Os datos de calquera vista pódense exportar en formato CSV, de forma que serán facilmente procesables por outras aplicacións.

[illegible][illegible]

--Vence