





**IDERA**

Infraestructura de  
Datos Espaciales de la  
República Argentina

**Guía Instalación de PostgreSQL,  
PostGIS, Tomcat, GeoServer y  
GeoNetwork en Centos 7.6**

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

<u>Título</u>	Guía de Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6
<u>Versión</u>	1.2
<u>Autor</u>	Grupo de Trabajo Tecnología y Desarrollo
<u>Fecha</u>	22/05/2019
<u>Estado</u>	Revisión
<u>Publicador</u>	Grupo de Trabajo Tecnología y Desarrollo
<u>Contacto</u>	<a href="mailto:coordinador_id@idera.gob.ar">coordinador_id@idera.gob.ar</a>
<u>Resumen</u>	Este documento es una guía que contiene las instrucciones de instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6 .
<u>Palabras claves</u>	IDERA, sistemas de información geográfica, GIS, software libre, servicios

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

## Objetivo

Este documento tiene como objetivo dar una guía clara del proceso de Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6 para los usuarios finales y los productores de datos.

## Alcance

El presente documento está dirigido a las diversas organizaciones de los ámbitos público, privado, académico, no gubernamental y sociedad civil de la República Argentina que requieran establecer un nodo IDE.

## Guía de Pasos:

### Primera Parte

#### A. Instalo Centos 7.6 con las siguientes opciones:

##### **Servidor web básico:**

*Cliente respaldo*

*Cliente de directorio*

*Agentes de huésped (al ejecutarse en hipervisores)*

*Plataforma Java*

*Cliente de sistema de archivos de red*

*Soporte para PHP*

*Python*

*Herramientas de desarrollo*

*Herramientas de seguridad*


**Luego de la instalación de Linux y en el primer booteo, actualizo el sistema operativo (S.O.):**

```
# yum update -y
```

**Reboteo Linux para que cargue el nuevo S.O. actualizado e instalo:**

```
# yum install -y epel-release
```

```
# yum install -y mc ntfs-3g lnx
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

## B. Instalación de PostgreSQL

### Instalo el repositorio de PostgreSQL (a la fecha, la versión estable era la 11)

```
# yum install https://download.postgresql.org/pub/repos/yum/11/redhat/rhel-7.5-x86_64/pgdg-redhat11-11-2.noarch.rpm -y
```

### Instalo desde el repositorio los siguientes paquetes de Postgres

```
# yum -y install postgresql11 postgresql11-server postgresql11-contrib postgresql11-libs postgresql11-devel
```

### Cambio la clave del usuario de linux postgres (NO el administrador de la base de datos):

```
# passwd postgres
```

### Inicializamos la base de datos por primera vez:

```
# /usr/pgsql-11/bin/postgresql-11-setup initdb
```

### Y levantamos el motor de base de datos:

```
# systemctl start postgresql-11
```

### Listamos las unidades, con seguridad la unidad postgresql-11 estará deshabilitada:

```
# systemctl list-unit-files | grep postgres
```

### Habilitamos la unidad postgresql-11


```
# systemctl enable postgresql-11
```

### Creamos la extensión adminpack:

```
# su - postgres
$ /usr/pgsql-11/bin/psql -p 5432 -c "CREATE EXTENSION adminpack;"
```

### Salimos:

```
$ exit
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

## C. Instalación PostGIS:

```
# yum -y install postgres25_11 postgres25_11-client
```

### Instalamos la librería ogrfdw:

```
# yum -y install ogr_fdw11
```

### Instalamos pgRouting, la extensión para enrutamiento espacial:

```
# yum -y install pgrouting_11
```

### Habilitamos y creamos las extensiones:

```
# su - postgres
$ psql
CREATE DATABASE gistest;
\connect gistest;
CREATE EXTENSION postgis;
CREATE EXTENSION postgis_topology;
CREATE EXTENSION ogr_fdw;
```

### Verificamos que todo esté correctamente instalado:

```
SELECT postgis_full_version();
```

### Si devuelve:

```
postgis_full_version
-----
-----
-----
POSTGIS="2.5.1 r17027" [EXTENSION] PGSQL="110" GEOS="3.7.0-CAPI-1.11.0 673b9939"
PROJ="Rel. 4.9.3, 15 A
ugust 2016" GDAL="GDAL 1.11.4, released 2016/01/25" LIBXML="2.9.1" LIBJSON="0.11"
TOPOLOGY RASTER
(1 fila)
```

### Está todo correcto

### Salimos del psql y retornamos al Shell del usuario root:

```
\q
exit
```

### Creamos el template template\_postgis, que será usado para GeoNetwork:

```
# su - postgres
$ createdb template_postgis
$ psql -d template_postgis -f /usr/pgsql-11/share/contrib/postgis-2.5/postgis.sql
$ psql -d template_postgis -f /usr/pgsql-11/share/contrib/postgis-
2.5/spatial_ref_sys.sql
$ psql -d template_postgis -f /usr/pgsql-11/share/contrib/postgis-
2.5/postgis_comments.sql
$ exit
```


### Probamos la base de datos creada:

```
# su - postgres
$ psql template_postgis
SELECT postgis_FULL_VERSION();
=# exit
```

### Y nuevamente:

```
$ exit
```

**La instalación de PostgreSQL, PostGIS y sus extensiones, ha concluido.**

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	Página 6 de 14

## Segunda Parte:

### A. Creación de la base de datos mapas para utilizarlo en el SIG

Desde el usuario root, modificar el archivo pg\_hba.conf y modificar el valor "ident".

```
# vi /var/lib/pgsql/11/data/pg_hba.conf
```

Ir al final del archivo, comentamos lo que está sobresaltado en amarillo (se recomienda copiar la línea debajo antes, dado que solamente cambiaremos "ident" por "md5") e insertamos las dos líneas que acá remarcamos en rojo y debe quedar de esta forma el final del mismo:

```
# TYPE DATABASE USER ADDRESS METHOD
# "local" is for Unix domain socket connections only
local all all peer
# IPv4 local connections:
# host all all 127.0.0.1/32 ident
host all all 127.0.0.1/32 md5
# IPv6 local connections:
# host all all ::1/128 ident
host all all ::1/128 md5
```

**Reiniciar la base de datos (o el servidor)**

```
# systemctl restart postgresql-11
```

**Creamos la base de datos:**

```
# su - postgres
```

```
# psql
```

*(Ahora puede que solicite la contraseña previamente creada)*

```
CREATE DATABASE mapas ENCODING 'UTF-8';
```

**Visualizamos si está creada como UTF 8:**

```
\list
```

**Nos conectamos a la base:**

```
\c mapas;
```

**Creamos el lenguaje (puede que ya esté creado)**

```
CREATE LANGUAGE plpgsql;
```

**Creamos usuario**


```
CREATE USER admin_mapas WITH SUPERUSER LOGIN PASSWORD 'adminmapas';
```

**Cambiamos el propietario de la base**

```
ALTER DATABASE mapas OWNER TO admin_mapas;
```

**Salimos de la base de datos:**

```
\q
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

**Ejecutar luego estos comandos desde la línea del sistema (nos solicitará la clave de admin\_mapas que para nuestro ejemplo es adminmapas):**

```
$ psql -d mapas -f /usr/pgsql-11/share/contrib/postgis-2.5/postgis.sql -U
admin_mapas -h localhost -W
$ psql -d mapas -f /usr/pgsql-11/share/contrib/postgis-2.5/spatial_ref_sys.sql -U
admin_mapas -h localhost -W
```

**Salimos del shell de postgres**

```
exit
```

**Editamos el archivo postgresql.conf**

```
vi /var/lib/pgsql/11/data/postgresql.conf
```

**Modificamos la línea donde dice:**

```
#listen_addresses = 'localhost'
```

**Descomentamos quitando el “#” al principio de la línea y reemplazamos ‘localhost’ por ‘\*’:**

```
listen_addresses = '*'
```


**Guardamos y salimos.**

**Podemos además dejar el puerto abierto para las futuras conexiones externas (QGIS).**

```
firewall-cmd --permanent --zone=public --add-port=5432/tcp
firewall-cmd --reload
```

**Y reiniciamos la base de datos:**

```
# systemctl restart postgresql-11
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	Página 8 de 14

## B. Instalación de Tomcat:

**Como usuario root, instalamos java:**

```
# yum -y install java
```

**Verificamos la versión de java:**

```
# java -version
```

**Actualizamos el sistema (IMPORTANTE)**

```
# yum update -y
```

**Creamos el grupo tomcat:**

```
# groupadd tomcat
```

**Agregamos el usuario tomcat al grupo tomcat:**

```
# useradd -M -s /bin/nologin -g tomcat -d /opt/tomcat tomcat
```

**Nos cambiamos a otra carpeta auxiliar para descargar el Tomcat:**

```
# cd
# mkdir temporal
# cd temporal
# yum install wget -y
# wget http://www-eu.apache.org/dist/tomcat/tomcat-9/v9.0.17/bin/apache-tomcat-9.0.17.tar.gz
```

**Ejecutamos los siguientes comandos para instalar el tomcat en el directorio /opt**

```
# tar -zxpvf apache-tomcat-9.0.17.tar.gz -C /opt/
# cd /opt/
# mv apache-tomcat-9.0.17 tomcat
# cd /opt/tomcat
# chown tomcat:tomcat /opt/tomcat
# chown -R tomcat:tomcat *
```

**Luego de ejecutados los comandos, la carpeta auxiliar, en este caso /temporal, y su contenido, ya se podrán eliminar**

**Modificamos el archivo /opt/tomcat/conf/tomcat-users.xml y configuramos los usuarios administradores de Tomcat, antes del tag final </tomcat-users>.**


```
# vi /opt/tomcat/conf/tomcat-users.xml
```

**Acá muestro como me quedó la configuración al final, agregamos lo resaltado en amarillo:**

```
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="admintomcat" roles="tomcat"/>
  <user username="both" password="admintomcat" roles="tomcat,role1"/>
  <user username="role1" password="admintomcat" roles="role1"/>
-->
<!-- Usuarios agregados -->
<!-- User manager-tomcat can access only manager section -->
<role rolename="manager-gui"/>
<user username="manager-tomcat" password="admintomcat" roles="manager-gui"/>
<!-- User Admin Who can access manager and admin section both -->
<role rolename="admin-gui"/>
<user username="admin-tomcat" password="admintomcat" roles="admin-gui"/>
</tomcat-users>
```

**Es en este archivo donde se puede modificar el usuario y la password. Se recomienda su modificación, tanto de usuarios como de clave y la eliminación de usuarios innecesarios.**



	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

**Antes de comenzar el servicio Tomcat, primero configuraremos la variable de entorno requerida CATALINA\_HOME usando los siguientes comandos:**

```
# echo "export CATALINA_HOME='/opt/tomcat/'" >> ~/.bashrc
# source ~/.bashrc
```

**Ejecutamos para comprobar el funcionamiento de Tomcat:**

```
cd /opt/tomcat/bin/
./startup.sh
```

**Si muestra lo siguiente:**

```
[root@ideer bin]# ./startup.sh
Using CATALINA_BASE:   /opt/tomcat/
Using CATALINA_HOME:   /opt/tomcat/
Using CATALINA_TMPDIR: /opt/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/tomcat/bin/bootstrap.jar:/opt/tomcat/bin/tomcat-juli.jar
Tomcat started.
```

**¡Es porque todo funciona correctamente!**

**Apagamos el servidor Tomcat**

```
# ./shutdown.sh
```

**Agregamos la siguiente regla en el firewall (habilitamos el puerto 8080):**

```
# firewall-cmd --permanent --zone=public --add-port=8080/tcp
# firewall-cmd --reload
```

**Y para poder administrarlo remotamente, creamos el archivo manager.xml:**

```
# cd /opt/tomcat/conf/Catalina/localhost/
# vim manager.xml
```

**Y agregamos el siguiente contenido (dado que el archivo está vacío):**

```
<Context privileged="true" antiResourceLocking="false"
  docBase="${catalina.home}/webapps/manager">
<Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="^.*$"/>
</Context>
```

**Cambiamos al directorio de instalación de Tomcat:**

```
# cd /opt/tomcat
```

**Modificamos la propiedad de los directorios:**

```
# chgrp -R tomcat /opt/tomcat
# chmod -R g+r conf
# chmod g+x conf
# chown -R tomcat webapps/ work/ temp/ logs/
```


**Instalamos el Unit File (o servicio) del Systemd**

**Creamos el archivo tomcat.service**

```
# vi /etc/systemd/system/tomcat.service
```

**A continuación, copiamos el siguiente script. Las necesidades de asignación de memoria están especificadas en la variable CATALINA\_OPTS:**

```
# Systemd unit file for tomcat
[Unit]
Description=Apache Tomcat Web Application Container
After=syslog.target network.target
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

```
[Service]
Type=forking

Environment=JAVA_HOME=/usr/lib/jvm/jre
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment='CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true
Djava.security.egd=file:/dev/./urandom'

ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/bin/kill -15 $MAINPID

User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always

[Install]
WantedBy=multi-user.target
```

### Guardamos el archivo y seteamos el flag de ejecución.

```
chmod +x /etc/systemd/system/tomcat.service
```

### Recargamos el Systemd:

```
systemctl daemon-reload
```

**ATENCIÓN:** En este punto se recomienda rebootear el equipo y luego continuar.

### Arrancamos el servicio tomcat:

```
systemctl start tomcat
```

### Si todo está bien lo podemos observar con el siguiente comando:


```
systemctl status tomcat
```

### Si entre el mensaje, aparece lo siguiente, todo está bien:

- tomcat.service - Apache Tomcat Web Application Container
  - Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; vendor preset: disabled)
  - Active: **active (running)**

### Habilitamos el servicio (unidad) para que arranque luego del booteo:

```
systemctl enable tomcat
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	Página 11 de 14

## C. Instalación de GeoServer

### Seguimos los siguientes pasos

- Ir al sitio <http://geoserver.org/>
- Descargar la última versión estable (Stable)
- Seleccionar "Web Archive"
- Esta acción permite descargar la última versión estable de GeoServer

### Una vez descargado el archivo debemos editar el siguiente archivo

```
# vi /opt/tomcat/webapps/manager/WEB-INF/web.xml
```

### Y debemos modificar esta parte (pasamos de 50 Mb a 100Mb):

```
<multipart-config>
  <!-- 50MB max -->
  <max-file-size>52428800</max-file-size>
  <max-request-size>52428800</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

### Por:

```
<multipart-config>
  <!-- 100MB max -->
  <max-file-size> 104857600</max-file-size>
  <max-request-size> 104857600</max-request-size>
  <file-size-threshold>0</file-size-threshold>
</multipart-config>
```

### Luego reiniciamos el Tomcat:

```
# systemctl restart tomcat
```

### Descomprimos el archivo descargado, luego nos conectamos al servidor de Tomcat desde:

```
http://ip_del_server:8080
```

Ingresamos como manager-tomcat y password admintomcat (o el usuario y password que hayamos definido en Tomcat) y luego desde el menú de desplegar, seleccionamos el archivo y lo desplegamos (tomará un tiempo desplegarlo).


### Testearlo desde el navegador con:

```
http://ip_del_server:8080/geoserver/web/
```

Usuario: admin

Password: geoserver

**Se requiere cambiar la password del usuario "admin" para evitar problemas de seguridad.**

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

## D. Instalación de GeoNetwork

### Instalamos apache

```
# yum install -y httpd
```

### Iniciamos apache

```
# systemctl start httpd
```

### Probamos si apache está arriba:

```
# lynx localhost:80
```

### Si todo está correcto, habilitamos el demonio para el booteo:

```
# systemctl enable httpd
```

### Y configuramos el firewall:

```
# firewall-cmd --permanent --add-service=http
# firewall-cmd --permanent --zone=public --add-port=80/tcp
# firewall-cmd --reload
```

### Paramos el servicio de Tomcat:

```
# systemctl stop tomcat
```

### Descargamos GeoNetwork del sitio: <http://geonetwork-opensource.org/>

### Seleccionamos el archivo geonetwork.war (Version 3.6.0 al momento de instalar)

### Copiamos el archivo geonetwork.war en el directorio: /opt/tomcat/webapps y le otorgamos propiedad a tomcat

```
# chown tomcat:tomcat /opt/tomcat/webapps/geonetwork.war
```

### Iniciamos el servicio de Tomcat

```
# systemctl start tomcat
```

### Esperamos un rato para que se despliegue el war y apagar el servicio de Tomcat nuevamente. Podemos ver el progreso ejecutando el comando `df -hk` para visualizar como va creciendo el file system y cuando deje de crecer el file system, ejecutamos:

```
# systemctl stop tomcat
# chown tomcat:tomcat -R /opt/tomcat
```

### Lo arrancamos nuevamente:

```
# systemctl start tomcat
```

### Creamos la base de datos para GeoNetwork desde psq

```
# su - postgres
$ psql
```

### Creamos usuario administrador para geonetwork y definimos su clave, en este caso el usuario será "geonetwork" y su clave "admingeonetwork", pero puede darse cualquier nombre, al igual que la modificación de la clave:


```
CREATE USER geonetwork PASSWORD 'admingeonetwork';
```

### Creamos la base de datos que utilizará geonetwork:

```
CREATE DATABASE geonetwork WITH ENCODING='UTF-8' OWNER=geonetwork TEMPLATE template_postgis;
```

### Otorgamos el privilegio de administración de la base "geonetwork" al usuario previamente definido (al que llamamos "geonetwork"):

```
GRANT ALL PRIVILEGES ON DATABASE geonetwork TO geonetwork;
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	

**Ejecutamos los siguientes comandos desde el psql para asegurar poseer las tablas principales de postgis con permisos total:**

```
\c geonetwork
ALTER VIEW geometry_columns OWNER TO geonetwork;
ALTER VIEW geography_columns OWNER TO geonetwork;
ALTER TABLE spatial_ref_sys OWNER TO geonetwork;
\q
```

**Cambiamos la clave de administrador de postgres (nos solicitará dos veces la clave):**

```
$ psql
\password
```

**Salimos del psql**

```
\q
```

**Testeamos la conexión con el usuario geonetwork:**

```
$ psql -U geonetwork -d geonetwork
```

**Si se conecta, está todo correcto, salir con:**

```
\q
```

**y seguir con la instalación, si no se conecta, modificar lo siguiente y volver a reintentar**

**ATENCIÓN: Si genera el siguiente error**

```
psql: FATAL: la autenticación Peer falló para el usuario «geonetwork»
```

**hacemos exit para regresar a la sesión anterior**

**entonces editar el siguiente archivo:**

```
vim /var/lib/pgsql/11/data/pg_hba.conf
```

**modificamos la siguiente línea, cambiando “peer” por “md5” y que quede de esta forma:**

```
local all all md5
```

**Y reiniciar la base de datos:**


```
systemctl restart postgresql-11
```

**reintentamos nuevamente:**

```
# su - postgres
$ psql -U geonetwork -d geonetwork
```

**Si conecta sin error, salimos:**

```
quit
exit
```

	Tipo de documento: Documento técnico	Versión: 1.2
	Grupo de trabajo: Tecnología y Desarrollo	Fecha: 22/05/2019
	<b>Guía Instalación de PostgreSQL, PostGIS, Tomcat, GeoServer y GeoNetwork en Centos 7.6</b>	Página 14 de 14

## E. Configuración de la base de datos en Geonetwork:

Como root (tipeamos `exit` para regresar a la sesión anterior o ingresamos como "root"), modificamos el archivo que configura la conexión:

```
# vim /opt/tomcat/webapps/geonetwork/WEB-INF/config-node/srv.xml
```

Seleccionamos y descomentamos el conector postgres y comentamos el h2 como indica la imagen a continuación (comentar en amarillo y descomentar lo que está en verde):

```
<!-- Uncomment the database configuration you need to use -->
<!--<import resource=" ../config-db/h2.xml"/> -->
<!--<import resource=" ../config-db/jndi-postgres-postgis.xml"/> -->
<!--<import resource=" ../config-db/oracle.xml"/>-->
<!--<import resource=" ../config-db/mysql.xml"/> -->
<!--<import resource=" ../config-db/db2.xml"/> -->
<import resource=" ../config-db/postgres.xml"/>
<!--<import resource=" ../config-db/sqlserver.xml"/> -->
<!--<import resource=" ../config-db/postgres-postgis.xml"/> -->
```

Ahora modificamos el conector, modificando el siguiente archivo:

```
# vim /opt/tomcat/webapps/geonetwork/WEB-INF/config-db/jdbc.properties
```

Ingresamos los datos de conexión de la base:

```
jdbc.username=geonetwork
jdbc.password=admingeonetwork
jdbc.database=geonetwork
jdbc.host=localhost
jdbc.port=5432
jdbc.basic.removeAbandoned=true
jdbc.basic.removeAbandonedTimeout=120
jdbc.basic.logAbandoned=true
jdbc.basic.maxActive=33
jdbc.basic.maxIdle=${jdbc.basic.maxActive}
jdbc.basic.initialSize=${jdbc.basic.maxActive}
jdbc.basic.maxWait=200
jdbc.basic.testOnBorrow=true
jdbc.basic.timeBetweenEvictionRunsMillis=10000
jdbc.basic.minEvictableIdleTimeMillis=1800000
jdbc.basic.testWhileIdle=true
jdbc.basic.numTestsPerEvictionRun=3
jdbc.basic.poolPreparedStatements=true
jdbc.basic.maxOpenPreparedStatements=1200
jdbc.basic.validationQuery=SELECT 1
jdbc.basic.defaultReadOnly=false
jdbc.basic.defaultAutoCommit=false
```

Reiniciamos Tomcat:

```
# systemctl restart tomcat
```

Opcional: Modificamos la variable `JAVA_OPTS`, editamos el archivo o lo creamos:

```
# vi /etc/default/tomcat
```

Agregamos lo siguiente:

```
JAVA_OPTS="-Djava.awt.headless=true -Dfile.encoding=UTF-8 -server -Xms1024m -Xmx4084m -XX:NewSize=512m -XX:MaxNewSize=1024m -XX:PermSize=512m -XX:MaxPermSize=1024m -XX:+DisableExplicitGC"
```

**Nota:** se realiza el seteo en las variables de acuerdo a la memoria del equipo disponible, en este ejemplo la memoria es de 4 Gb de RAM (-Xmx4084m), donde Xmx es el máximo de la memoria disponible por la máquina virtual, también es importante la correcta configuración de la memoria Perm para que posibilite una carga rápida del sistema.