



iBit

Illes Balears
innovació
tecnològica

Instal·lació de SISTRA

Guia Ràpida d'Instal·lació de SISTRA





Informació general del document.

Descripció.

Títol: Guia Ràpida d'Instal·lació de SISTRA
Estat: Esborrany/Aprovat
Versió: 1.0
Autor/s: Antoni Nadal
Creat: 23/11/09
Modificat 21/12/11
Fitxer: Guia_Rapida_de_Instalacio_de_Sistra.odt

Històric de modificacions.

Comentari:	Autor/s:	Data:
-------------------	-----------------	--------------

Font documental.



Index de contingut

1.-Introducció.....	3
2.-Configurar JBOSS.....	4
2.1.-Llibreries.....	4
2.2.-Fitxers de Configuració.....	4
2.3.-Copia de binaris.....	7
3.-Gestió de BBDD.....	7
4.-Configurar SISTRA.....	11
4.1.-Configurar IP's.....	11
4.2.-Plantilla de Justificants per Defecte.....	11
4.3.-Revisar Documentació addicional.....	13
5.-Configuracions Externes.....	13
5.1.-Configurar OpenOffice.....	13
5.1.1.- Instal·lar OpenOffice.....	13
5.1.2.- Arrancar OpenOffice com a Servei.....	14
5.1.3.- Script per Arrancar OpenOffice com a Servei.....	14
6.-ANNEX I: Compilar SISTRA des de subversion de sourceforge.....	15
7.-ANNEX II: Script que automatitza la cridada als scripts sql.....	16

1.- Introducció

Aquest document explicar com a partir d'un binari SISTRA o una compilació del codi de SISTRA, es pot instal·lar sobre un servidor jboss i el sistema gestor de bbdd.

(A) COMPILAT: S'han de seguir les instruccions de l'annex I d'aquest document.

(B) DESCARREGAR BINARI: Accedint al projecte sistra de sourceforge en podem



descarregar el binari corresponents. Per exemple accedir via web a la següent adreça <http://sourceforge.net/projects/sistra/files/sistra-1.1/sistra-1.1.1/bin/> i descarregar el fitxer `sistra-1.1.1_postgresql_jaas.zip` o `sistra-1.1.1_oracle_jaas.zip`, segons la BBDD que tinguem. Després descomprimir el zip en l'arrel del nostre home.

Aquí s'enumeren la llista d'accions per posar en marxa un sistema SISTRA:

2.- Configurar JBOSS

Per millorar els scripts farem ús d'una variable d'entorn per apuntar al nostre servidor JBoss:

```
$ export JBOSS=/usr/local/jboss-3.2.8.SP1
```

2.1.- Llibries

1.- Copiar commons-codec-1.3.jar a /usr/local/jboss-3.2.8.SP1/server/default/lib.

```
$ sudo cp ~/sistra/lib/commons-codec-1.3.jar $JBOSS/server/default/lib
```

2.- Actualitzar arxius de xerces:

```
$ sudo cp ~/sistra/doc/resources/xerces2.7.1/* $JBOSS/lib/endorsed/
```

3.- Actualització llibreria SAA

```
$ sudo rm $JBOSS/server/default/lib/jboss-saaj.jar
$ sudo cp ~/sistra/lib/saaj-api.jar $JBOSS/server/default/lib/
$ sudo cp ~/sistra/lib/saaj-impl.jar $JBOSS/server/default/lib/
```

4.- Baixar el fitxer ojdbc14.jar

Accedir a http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc_10201.html i descarregar el fitxer `ojdbc.jar` i copiar-ho a `~/sistra/lib` i dins `/usr/local/jboss-3.2.8.SP1/server/default/lib/`.

5.- Controlador JDBC per postgresql



Si estem emprant com a SGBD PostgreSQL llavors descarregar el fitxer de la següent adreça <http://jdbc.postgresql.org/download/postgresql-8.4-703.jdbc3.jar> i copiar-ho dins el directori de llibreries del JBOSS: \$JBOSS/server/default/lib/.

2.2.- Fitxers de Configuració

1.- Editar el fitxer \$JBOSS/server/default/conf/jboss-service.xml:

Modificar la línia afegint "deploysistra/":

ACTUAL	MODIFICAT
<pre><attribute name="URLs"> deploy/ </attribute></pre>	<pre><attribute name="URLs"> deploy/, deploysistra/ </attribute></pre>

I crear el nou directori deploysistra:

```
$ sudo mkdir -p $JBOSS/server/default/deploysistra
```

2.- Indicar la ubicació dels fitxers de propietats. Crear un nou fitxer anomenat sistra-properties-service.xml dins deploysistra (/usr/local/jboss-3.2.8.SP1/server/default/deploysistra) amb el següent contingut:

```
<?xml version="1.0" encoding="UTF-8"?>
<server>
<mbean code="org.jboss.varia.property.SystemPropertiesService"
name="jboss:type=Service,name=BootProperties">
<attribute name="Properties">
<!-- En /config_sistra/sistra estaran los archivos de configuracion -->
ad.path.properties=/config-sistra/
</attribute>
</mbean>
</server>
```

Executar les següents comandes:

```
$ sudo mkdir /config-sistra
$ sudo mkdir /config-sistra/sistra
$ sudo cp ~/sistra/doc/resources/config/sistra/* /config-sistra/sistra
```

3.- Afegir DISPLAY en Sistemes unix. En \$JBOSS/bin/run.sh afegir les següent línies:

```
DISPLAY=:0.0
export DISPLAY
```



4.- Afegir la següent entrada al final del fitxer \$JBASS/server/default/deploy/jms/jbossmq-destinations-service.xml (abans de </server>)

```
<mbean code="org.jboss.mq.server.jmx.Queue"
name="jboss.mq.destination:service=Queue,name=AvisadorBTE">
<depends optional-attribute-
name="DestinationManager">jboss.mq:service=DestinationManager</depends>
<!-- redelivery delay 30 [s] -->
<attribute name="RedeliveryDelay">30000</attribute>
<!-- unlimited redelivery -->
<attribute name="RedeliveryLimit">3</attribute>
</mbean>
```

5.- Configurar Correu

(a) Copiar fitxer de service:

```
$sudo cp ~/sistra/doc/resources/bbdd/datasources/mobtratel-mailTest-
service.xml $JBASS/server/default/deploy/sistra/
```

(b) Editar el fitxer \$JBASS/server/default/deploy/sistra/mobtratel-mailTest-service.xml i configurar-ho:

```
<!DOCTYPE server>
<server>
  <mbean code="org.jboss.mail.MailService"
    name="jboss:service=MobtratelMailTest">
    <attribute name="JNDIName">java:/es.caib.mobtratel.mailTest</attribute>
    <attribute name="User">anadal@ibit.org</attribute>
    <attribute name="Password">contrasenya</attribute>
    <attribute name="Configuration">
      <configuration>
        <property name="mail.transport.protocol" value="smtp"/>
        <property name="mail.smtp.host" value="smtp.mailserver.es"/>
        <property name="mail.from" value="cuental@mailserver.es"/>
        <property name="mail.debug" value="false"/>
        <property name="mail.smtp.auth" value="true"/>
      </configuration>
    </attribute>
  </mbean>
</server>
```

6.- Descomentar entrada dins el fitxer \$JBASS/server/default/conf/jboss-service.xml

ACTUAL	MODIFICAT
<pre><mbean code="org.jboss.tm.XidFactory" name="jboss:service=XidFactory"> <!--attribute</pre>	<pre><mbean code="org.jboss.tm.XidFactory" name="jboss:service=XidFactory"> <attribute</pre>



```
name="Pad">true</attribute-->
</mbean>
```

```
name="Pad">true</attribute>
</mbean>
```

7.- Configurar accés dels usuaris. Afegir al final del fitxer \$JBASS/server/default/conf/login-config.xml abans de "</policy>" la següent entrada:

```
<application-policy name = "seycon">
<authentication>
<login-module code = "es.caib.mock.loginModule.MockCertificateLoginModule"
flag = "sufficient">
<module-option name="roleTothom">tothom</module-option>
</login-module>
<login-module code = "es.caib.mock.loginModule.MockDatabaseLoginModule"
flag = "sufficient">
<module-option name="unauthenticatedIdentity">nobody</module-option>
<module-option name = "dsJndiName">java:/es.caib.mock.loginModule.db</module-option>
<module-option name = "principalsQuery">SELECT USU_PASS,USU_NOM,USU_NIF FROM
SC_WL_USUARI WHERE USU_CODI = ?</module-option>
<module-option name = "rolesQuery">SELECT UGR_CODGRU, 'Roles' FROM SC_WL_USUGRU
WHERE UGR_CODUSU = ?</module-option>
</login-module>
</authentication>
</application-policy>
```

8.- Modificar el Tomcat per habilitar el single sign on entre les aplicacions. Per això hi ha que descomentar el valve que l'implementa el fitxer \$JBASS/server/default/deploy/jbossweb-tomcat50.sar/server.xml

```
<Valve className="org.apache.catalina.authenticator.SingleSignOn" debug="0"/>
```

2.3.- Copia de binaris

1.- Hi ha que col·locar el jar loginModuleMOCK.jar en \$JBASS/server/default/lib. A més en aquest directori també es requereixen les llibreries de bouncy castle pel maneig de certificats (bcmail.jar, bcprov.jar i bctsp.jar). Podem trobar aquestes llibreries en el directori del projecte: ~/sistra/pluginsMOCK/lib. Són necessàries si s'empra el modul de login de test i el plugin de firma de test.

```
$ sudo cp ~/sistra/pluginsMOCK/product/lib/loginModuleMOCK.jar $JBASS/server/default/lib
$ sudo cp ~/sistra/lib/bouncy/bcmail.jar $JBASS/server/default/lib
$ sudo cp ~/sistra/lib/bouncy/bcprov.jar $JBASS/server/default/lib
$ sudo cp ~/sistra/lib/bouncy/bctsp.jar $JBASS/server/default/lib
```

2.- Copiar ears a deploysistra

```
$ sudo cp ~/sistra/product/ear/1-sistra.ear
```



```
$JBASS/server/default/deployastra/  
$ sudo cp ~/sistra/pluginsMOCK/product/ear/2-modul-pluginsMOCK.ear  
$JBASS/server/default/deployastra/
```

2.4.- DataSources

Els datasources definits a ~/sistra/doc/resources/bbdd/datasources/ s'han de copiar a /usr/local/jboss-3.2.8.SP1/server/default/deployastra/:

- audita-ds.xml
- loginMOCK-ds.xml
- redose-ds.xml
- bantel-ds.xml
- mobtratel-ds.xml
- sistra-ds.xml
- form-ds.xml
- zonaper-ds.xml

```
$sudo cp ~/sistra/scripts/datasources/ [bbdd]/*.xml  
$JBASS/server/default/deployastra/
```

on [bbdd] pot ser oracle o postgresql.

3.- Gestió de BBDD

3.- Crear usuaris i BBDD:

3.1.- Connectar-se a la BBDD

```
$ sudo bash  
$ su postgres  
$ psql -U postgres
```

3.2.- Per cada BBDD crearem un usuari:

```
CREATE USER "audita" WITH ENCRYPTED PASSWORD 'audita' NOCREATEUSER;  
CREATE USER "loginMock" WITH ENCRYPTED PASSWORD 'loginMock' NOCREATEUSER;  
CREATE USER "redose" WITH ENCRYPTED PASSWORD 'redose' NOCREATEUSER;
```




```
CREATE USER "bantel" WITH ENCRYPTED PASSWORD 'bantel' NOCREATEUSER;
CREATE USER "mobtratel" WITH ENCRYPTED PASSWORD 'mobtratel' NOCREATEUSER;
CREATE USER "sistra" WITH ENCRYPTED PASSWORD 'sistra' NOCREATEUSER;
CREATE USER "form" WITH ENCRYPTED PASSWORD 'form' NOCREATEUSER;
CREATE USER "zonaper" WITH ENCRYPTED PASSWORD 'zonaper' NOCREATEUSER;
```

3.3.- Crear la BBDD

```
CREATE DATABASE "audita" WITH OWNER=audita;
CREATE DATABASE "loginMock" WITH OWNER="loginMock";
CREATE DATABASE "redose" WITH OWNER=redose;
CREATE DATABASE "bantel" WITH OWNER=bantel;
CREATE DATABASE "mobtratel" WITH OWNER=mobtratel;
CREATE DATABASE "sistra" WITH OWNER=sistra;
CREATE DATABASE "form" WITH OWNER=form;
CREATE DATABASE "zonaper" WITH OWNER=zonaper;
```

3.4.- Sortir

\q

4.- Permisos d'accés a la BBDD per part de l'usuari i d'altres host

4.1.- S'ha d'editar el fitxer /etc/postgresql/8.4/main/pg_hba.conf i afegir les següents línies

```
## ===== audita =====
local audita          audita          password
host    audita        audita          www.xxx.yyy.zzz/24 trust
## ===== loginMock =====
local loginMock       loginMock       password
host    loginMock     loginMock       www.xxx.yyy.zzz/24 trust
## ===== redose =====
local redose          redose          password
host    redose        redose          www.xxx.yyy.zzz/24 trust
## ===== bantel =====
local bantel          bantel          password
host    bantel        bantel          www.xxx.yyy.zzz/24 trust
## ===== mobtratel =====
local mobtratel       mobtratel       password
host    mobtratel     mobtratel       www.xxx.yyy.zzz/24 trust
## ===== sistra =====
local sistra          sistra          password
host    sistra        sistra          www.xxx.yyy.zzz/24 trust
## ===== form =====
local form            form            password
host    form          form            www.xxx.yyy.zzz/24 trust
## ===== zonaper =====
local zonaper         zonaper         password
host    zonaper       zonaper         www.xxx.yyy.zzz/24 trust
```

On posa `www.xxx.yyy.zzz` l'hem de substituir per els tres primers números de la nostra IP.



Per exemple si la nostra IP es 192.168.121.132 llavors on posa `www.xxx.yyy.zzz` hem d'escriure 192.168.121.132. Podem obtenir la nostra IP escrivint `ipconfig` en sistemes Windows o `ifconfig` en sistemes linux.

4.2.- Reiniciem el postgres per a que els canvis s'apliquin:

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

4.3.- Si no teniu accés des d'ordinadors diferents a localhost, seguiu les recomanacions el que es diu en la següent pàgina:

<http://www.cyberciti.biz/tips/postgres-allow-remote-access-tcp-connection.html>

5.- Crear BBDD i configurar-la (Per cadascun dels usuaris i BBDD)

5.1.- Connectar-se al servidor de BBDD. Si la BBDD esta en el mateix servidor executar

```
$ psql -h localhost -p 5432 -U [USER] -W -d [BBDD]
```

i si està en un altra servidor llavors executar

```
$ psql -h www.xxx.yyy.zzz -p 5432 -U [USER] -W -d [BBDD]
```

5.2.- Donam permisos al usuari:

```
GRANT ALL PRIVILEGES ON DATABASE "[USER]" TO [BBDD];  
GRANT ALL PRIVILEGES ON SCHEMA PUBLIC TO [USER];
```

5.3.- Importam l'estructura i dades de taules per a la BBDD en concret

```
\i ~/sistra/scripts/bbdd/x.y/[sgdb]/create/sistra_[BBDD]_create_schema.sql  
\i ~/sistra/scripts/bbdd/x.y/[sgdb]/create/sistra_[BBDD]_create_data.sql
```

On **[sgdb]** pot ser oracle o postgresql i **[BBDD]** és alguna de les Bases de Dades descrites anteriorment: audita, form, ...

5.4.- Sortir:

```
\q
```

Per exemple per "redose" i per una bbdd postgresql d'un sistra 1.1.x, la llista de comandes a executar és:

```
$ psql -h localhost -p 5432 -U redose -W -d redose
```



```
GRANT ALL PRIVILEGES ON DATABASE "redose" TO redose;
GRANT ALL PRIVILEGES ON SCHEMA PUBLIC TO redose;
\i
~/sistra/doc/resources/bbdd/scripts/bbdd/1.1/postgresql/create/sistra_redose_create_schema.sql
\i ~/sistra/scripts/bbdd/1.1/postgresql/create/sistra_redose_create_data.sql
\q
```

Fer el mateix per la resta d'scripts.

6.- Donar permisos totals a l'usuari admin.

Per defecte, els usuaris es troben dins la BBDD loginMock en la taula SC_WL_USUARI. En aquesta s'haurà creat un usuari administrador amb username "admin" i contrasenya "admin", però que per defecte no pot accedir a tots els contextos existents en sistra.

El fitxer ~/sistra/doc/resources/bbddscripts/permisos_totals_per_admin.sql conté les seqüències sql per donar a l'usuari admin permisos per accedir a qualsevol lloc de sistra, per això només s'ha d'executar el fitxer anterior o les sentències del seu interior.

7.- Crear seqüències. El numero que apareix és l'any, per la qual cosa arribat l'any 2016 s'hauran d'afegir noves seqüències pels anys següents (STR_SEQE17, STR_SEQE18, ...)

7.1.- En la BD de Sistra:

```
CREATE SEQUENCE STR_SEQE12 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQE13 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQE14 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQE15 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQE16 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;

CREATE SEQUENCE STR_SEQP12 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQP13 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQP14 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQP15 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE STR_SEQP16 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
```

7.2.- En la BD de Bantel:

```
CREATE SEQUENCE BTE_SEQE12 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE BTE_SEQE13 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE BTE_SEQE14 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE BTE_SEQE15 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
CREATE SEQUENCE BTE_SEQE16 INCREMENT 1 MINVALUE 1 MAXVALUE 9223372036854775807;
```

8.- Modificar postgresql.conf



Configurar arxiu postgresql.conf i modificar entrada "# max_prepared_transactions = 0" per "max_prepared_transactions = 20".

3.1.1.- Funcions d'Oracle per PostgreSQL

A causa que algunes parts de SISTRA estan implementades internament fent ús de funcions específiques d'Oracle en les consultes SQL, cal instal·lar dins del PostgreSQL un paquet anomenat orafce format per les funcions SQL més importants d'Oracle implementades per PostgreSQL.

Podeu descarregar el paquet i veure com s'instal·la en <http://orafce.projects.postgresql.org/>.

A continuació es descriuen les instruccions per la versió 3.0.3 (per noves versions llegir l'apartat de instal·lació de la wiki d'aquest producte):

1) Executar les següents comandes :

```
$ wget http://pgfoundry.org/frs/download.php/2908/orafce-3.0.3.tar.gz
$ tar xvfz orafce-3.0.3.tar.gz
```

2) Abans de compilar necessitem instal·lar algunes eines de desenvolupament. Per això executarem:

```
$ sudo apt-get install libpq-dev
$ sudo apt-get install postgresql-server-dev-8.4
$ sudo apt-get install bison
$ sudo apt-get install flex
```

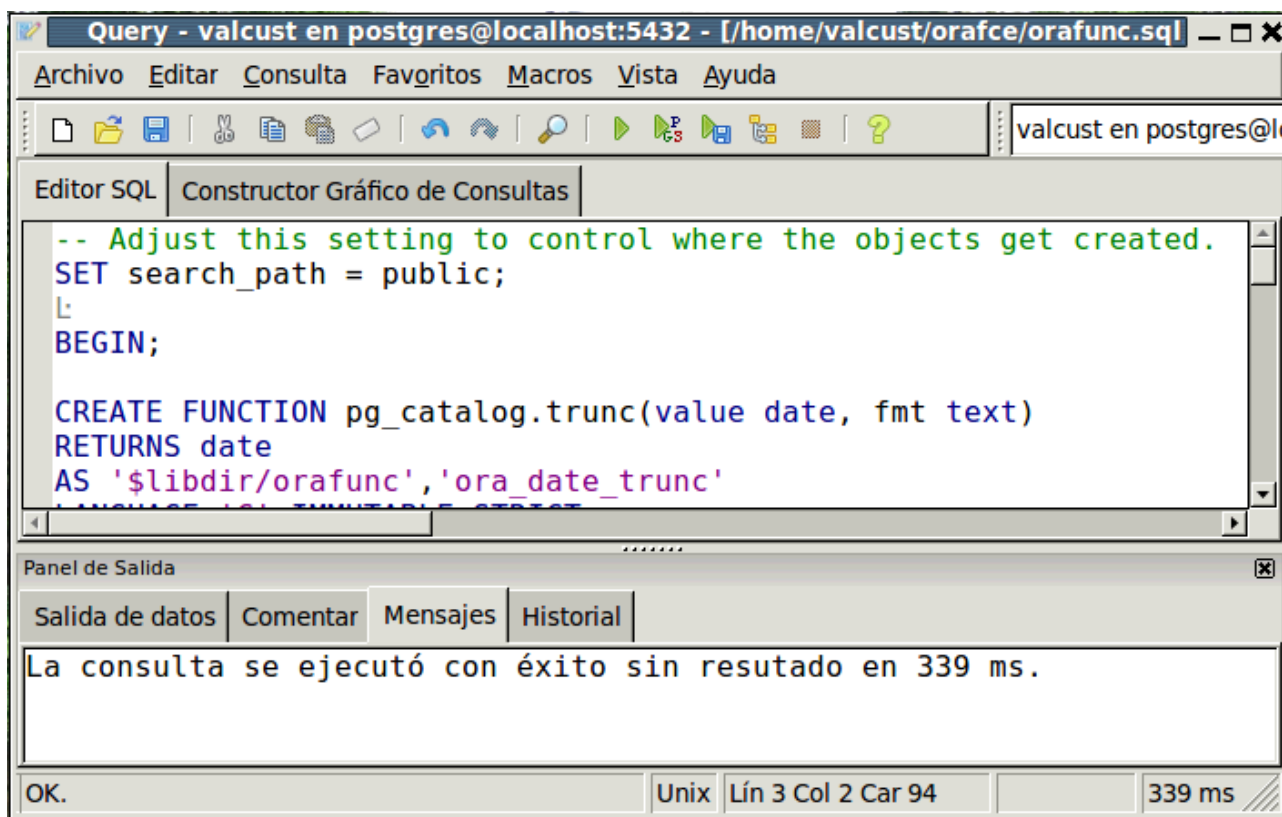
3) Anar al directori orafce (cd orafce) i executar la següent comanda

```
$ make USE_PGXS=1
$ sudo make USE_PGXS=1 install
```

4) Reiniciar postgresql:

```
$ sudo /etc/init.d/postgresql-8.4 restart
```

5) Obrir una connexió a la BBDD emprant "pgAdmin III" amb l'usuari postgres i accedint a la BBDD "valcust". Obrir l'editor SQL i carregar el fitxer ~/orafce/orafunc.sql. Pitar el botó verd per executar les sentències.



- 6) Per comprovar que tot ha anat bé, anau al menú “Archivo” - “Nueva Ventana” de pgAdmin III, escriviu la següent sentència i executau-la:

```
select next_day(current_date, 'saturday');
```

4.- Configurar SISTRA

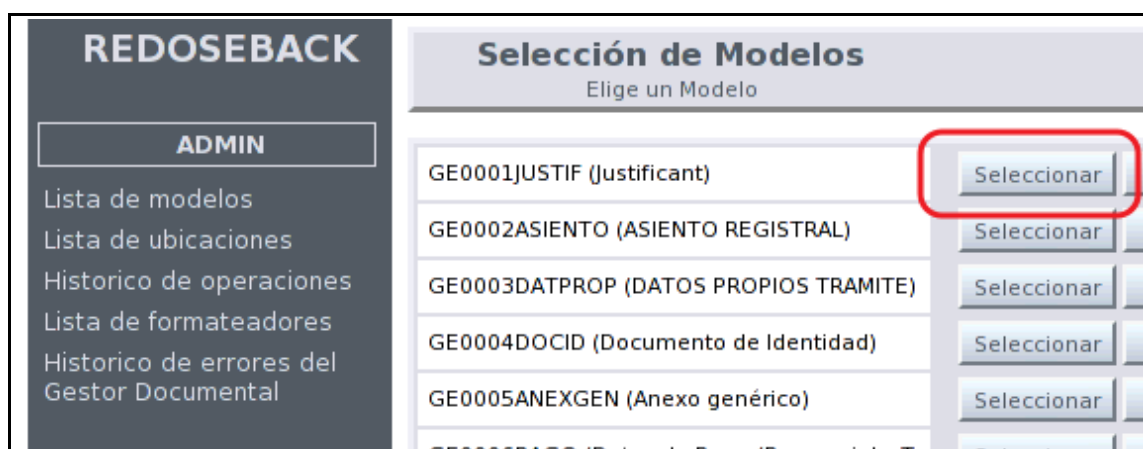
4.1.- Configurar IP's

Obrir el document `/config-sistra/sistra/global.properties` i reemplaçar totes les entrades “localhost” per l'adreça o IP pública del JBOSS.

4.2.- Plantilla de Justificants per Defecte

Al finalitzar cada instància de tràmit, es crea un document que és un justificant del tràmit que l'usuari acaba de realitzar. La configuració bàsica per defecte per generar aquest document es cerca dins un Model anomenat "GE0001JUSTIF (Justificant)" de RedoseBack. Per defecte ve preconfigurat amb un fitxer de propietats (cambiar.txt) que no és vàlid per la qual cosa s'ha d'inicialitzar de la següent forma:

(a) Accedir al RedoseBack ([http://\[SERVER\]:8080/redoseback](http://[SERVER]:8080/redoseback)) i seleccionar el model GE0001JUSTIF



(b) Seleccionar la versió 1 i després seleccionar "Plantillas Asociadas". Ens apareixerà una pantalla com la següent:



REDOSEBACK

ADMIN

- ca de modelos
- ca de ubicaciones
- torico de operaciones
- ca de formateadores
- torico de errores del
- stor Documental

Modificación de Plantillas

Datos de la Plantilla

Tipo	PDF
Formateador	Formateador específico para Justificante
Defecto	<input checked="" type="checkbox"/>
Barcode verificador	<input checked="" type="checkbox"/>
Sello (Pre) Registro/Envio	<input type="checkbox"/>

Catalan

Spanish

Fichero

cambiar.txt

Borrar

Nuevo fichero

Browse...

Fichero asociado a la plantilla para el idioma:

- PDF para formateadores basados en plantillas PDF
- Archivo jrxml o archivo zip para formateador basado en JasperReport
- Otro tipo de archivo dependiendo del formateador específico

Modificar

Reiniciar

Cancelar

- (c) Dins el directori scripts\plantilles\ del repositori de codi font de sistra trobarem dos fitxers (justificante-messages_es.properties i justificante-messages_ca.properties) que haurem d'actualitzar pitjant allà on posa "browser" tant de la pipella Spanish com Catalan respectivament.



REDOSEBACK

ADMIN

- Lista de modelos
- Lista de ubicaciones
- Historico de operaciones
- Lista de formateadores
- Historico de errores del Gestor Documental

Modificación de Plantillas

Datos de la Plantilla

Tipo	PDF
Formateador	Formateador específico para Justificante ▼
Defecto	<input checked="" type="checkbox"/>
Barcode verificador	<input checked="" type="checkbox"/>
Sello (Pre) Registro/Envío	<input type="checkbox"/>

Catalan Spanish

Fichero	justificante-messages_es.properties	Borrar
Nuevo fichero	<input type="text"/>	Browse...

Fichero asociado a la plantilla para el idioma:

- PDF para formateadores basados en plantillas PDF
- Archivo jrxml o archivo zip para formateador basado en JasperReport
- Otro tipo de archivo dependiendo del formateador específico

Modificar
Reiniciar
Cancelar

4.3.- Revisar Documentació adicional

<< PENDENT: Plugins, Connectors, Gestors de Documentació, ... >>

5.- Configuracions Externes

5.1.- Configurar OpenOffice

5.1.1.- Instal·lar OpenOffice

Executar la següent comanda per instal·lar OpenOffice en un sistema Linux:

```
# apt-get install openoffice.org
```




5.1.2.- Arrancar OpenOffice com a Servei

Crearem el fitxer “openoffice” en el directori /etc/init.d amb el contingut que es descriu en el punt “5.1.3.-Script per Arrancar OpenOffice com a Servei”. Per fer que el sistema executi cada vegada que arranca aquest script s'ha d'executar la següent comanda dins del directori /etc/init.d/:

```
$ sudo chmod 777 /etc/init.d/openoffice  
$ chkconfig -add openoffice
```

5.1.3.- Script per Arrancar OpenOffice com a Servei



```
#!/bin/bash
### BEGIN INIT INFO
### END INIT INFO
# openoffice.org headless server script
#
# chkconfig: 2345 80 30
# description: headless openoffice server script
# processname: openoffice
#
# Author: Vic Vijayakumar
# Modified by Federico Ch. Tomaszczik
#
OOo_HOME=/usr/bin
SOFFICE_PATH=$OOo_HOME/soffice
PIDFILE=/var/run/openoffice-server.pid
set -e
case "$1" in
start)
if [ -f $PIDFILE ]; then
echo "OpenOffice headless server has already started."
sleep 5
exit
fi
echo "Starting OpenOffice headless server"
$SOFFICE_PATH -headless -nologo -nofirststartwizard
-accept="socket,host=127.0.0.1,port=8100;urp" & > /dev/null 2>&1
touch $PIDFILE
;;
stop)
if [ -f $PIDFILE ]; then
echo "Stopping OpenOffice headless server."
killall -9 soffice && killall -9 soffice.bin
rm -f $PIDFILE
exit
fi
echo "Openoffice headless server is not running."
exit
;;
*)
echo "Usage: $0 {start|stop}"
exit 1
esac
exit 0
```



6.- ANNEX I: Compilar SISTRA des de subversion de sourceforge

Aquest manual explica com compilar l'aplicació SISTRA a partir del repositori de subversion de sourceforge. Es requereix java i ant. La compilació es realitzarà en un directori sistra del home de l'usuari (~/.sistra/).

- (1) Estant en el home de l'usuari fer check out executant la següent comanda:

```
$svn co https://sisra.svn.sourceforge.net/svnroot/sistra/sistra-core/trunk sistra
```

Podem baixar la branca principal (trunk) o una branca qualsevol (branch/INDRA-103-110)

- (2) Baixar el fitxer ojdbc14.jar de http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc_10201.html i copiar-ho a ~/.sistra/lib i dins /usr/local/jboss-3.2.8.SP1/server/default/lib/.

- (3) Editar fitxer ~/.sistra/config.properties i seleccionar el SGBD que volem per hibernate: (Descomentar l'opció desitjada). Per postgresql quedaria com:

```
# Configuracion Hibernate
# ---- PostgreSQL
hibernate.dialect=net.sf.hibernate.dialect.PostgreSQLDialect
hibernate.query.substitutions=

# ----- Oracle
# hibernate.dialect=net.sf.hibernate.dialect.Oracle9Dialect
# hibernate.query.substitutions=true 1, false 0
```

- (4) Anar a ~/.sistra/ i compilar executant la següent comanda:

```
$ ant
```

- (5) Anar a ~/.sistra/pluginsMOCK i executar:

```
$ant
```