



# Guía de despliegue y configuración SincroHCE

v3.4

Diciembre, 2018  
Equipo de Desarrollo  
Equipo de Arquitectura

## Contenido

Introducción .....	3
Requerimientos previos .....	3
Configuración y despliegue .....	3
Configuración de la base de datos .....	3
Configuración client.cfg.....	4
Paramétrica .....	4
Despliegue.....	5

## Introducción

Este módulo surge como respuesta a la necesidad de contar con un componente genérico que facilite la creación de CDAs, firma de los mismos y generación de metadata asociada, entre otros.

Por lo que permite mantener la información necesaria de pacientes y documentos clínicos asociados a actos clínicos que involucren a dichos pacientes.

Las estructuras contienen los datos necesarios para:

- Crear pacientes en el Índice Nacional de Usuarios de persona (INUS)
- Crear y almacenar en el XDS local de una Institución los CDAs y metadata asociada
- Crear y almacenar en el Registro XDS Nacional la metadata a documentos clínicos de la Institución

## Requerimientos previos

1. Linux y postgresql 9.5
2. Despliegue de XDS (v2.+ o v3.+)
3. AppliancePS
4. truststore y keystore

## Configuración y despliegue

### Configuración de la base de datos

Para crear la base de datos será necesario ejecutar las sentencias que se muestran en el siguiente esquema.

SINCROHCE

- CREATE USER <USER\_SINCROHCE\_DB> WITH PASSWORD '<PASS\_SINCROHCE\_DB>';
- CREATE DATABASE sincrohce OWNER=<USER\_SINCROHCE\_DB>

Luego, ejecutar el script que contiene el esquema (SINCRO\_3.4\_DB\_SCHEMA.sql). Tener en cuenta que cuando se ejecute la creación de tablas se debe realizar bajo el usuario (OWNER)

generados. En caso contrario se puede dar un error al momento de ejecución por problemas de permisos.

## Configuración client.cfg

Dentro de WEB-INF/classes/com/sincrohce es necesario modificar el archivo client.cfg. Los datos a configurar son los que se muestran en la siguiente tabla.

VARIABLES	DATOS
USER_ID	Nombre Usuario BD (Owner)
USER_PASSWORD	Password del Usuario BD (Owner)
DB_URL	jdbc: postgresql://[HOST_BD]: [Puerto]/NombreDB
CS_BLOB_PATH	PublicTempStorage

## Paramétrica

### Application

Esta tabla contiene una entrada por cada aplicación o subsistema que sea gestionado por la misma Institución/Organización. En caso de que no se distinga más de una aplicación o subsistema existirá un solo registro tal como se muestra en la siguiente tabla.

ID	CÓDIGO	MEDIA TYPE
1	<CODIGO_APP>	<COD_MEDIA_TYPE>

### ParameterType

ID	CÓDIGO	DESCRIPCIÓN
1	INST	Paramétrica relacionada a la Institución
2	KEYORG	Keystore que contiene el certificado de organización para firmar el CDA
3	XDSREP	Datos del XDS asociado a la Institución
4	INUS	Datos del AppliancePS asociado a la Institución

### Parameter

PARAMETER TYPE ID	PARAMETER CODE	PARAMETER VALUE	PARAMETER DESC
1	STRUCT	<ID_ESTRUCTURA>	Id Estructura Institución
	NAME	<NOMBRE_INST>	Nombre de la Institución
	REPID	2.16.858.2.<ID_ESTRUCTURA>.71867.	Raíz del OID de Repositorio
	OID	<OID_INST>	OID Institución
2	PATH	<PATH_KEYSTORE>	Ruta del keystore
	PASS	<PASSWORD>	Password del keystore
	ALIAS	<ALIAS>	Alias del certificado en el keystore

3	URL	https://<HOST_LOCAL>:<PUERTO>/XDSBUS/adocumentrepository_services	URL XDS BUS
	WSDL	file:///<PATH>/adocumentrepository_services.xml	WSDL XDS BUS Services
	PORT	adocumentrepositoryPort	Nombre Puerto XDS BUS
4	URL	https://<HOST_APPLIANCEPS>/AppliancePS/agestionpacientes_services	URL AppliancePS
	WSDL	file:///<PATH>/agestionpacientes_services.xml	WSDL AppliancePS
	PORT	agestionpacientesPort	Nombre Puerto AppliancePS

## PersonIdType

Esta tabla contiene los OIDs de tipo de documentos más utilizados por las Instituciones. En caso de manejar otro OID que no se encuentre definido, se puede configurar en PersonIdType.

Además, esta tabla contiene un registro: <COD\_APP>, <OID\_MNR\_APP> por cada identificador de aplicación o subsistema de la Institución. En caso de que no se distinga más de una aplicación o subsistema existirá un solo registro MRN.

ID	CÓDIGO	ROOT	DESCRIPCIÓN	TIPO
1	CI	2.16.858.2.10000675.68909	Cédula de identidad uruguaya	OTR
2	PAS	2.16.858.2.10000675.68912	Pasaporte	OTR
3	CRCI	2.16.858.2.10000675.68944	Credencial cívica	OTR
4	CCOND	2.16.858.2.10000675.69011	Libreta de conducir nacional	OTR
5	AFIPUB	2.16.858.2.10000675.69019	Afiliado público	OTR
6	CJPROF	2.16.858.0.1.0.0	Nro. Caja profesional	OTR
7	<COD_APP>	<OID_MNR_APP>	OID MNR APP	MNR

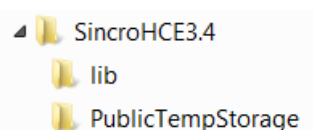
## Despliegue

### Logs

Crear una carpeta donde se almacenarán los logs de ejecución de los procesos del SincroHCE (PATH\_LOG)

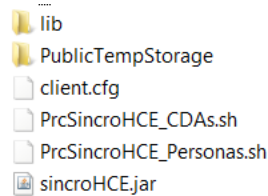
### Módulo

Crear una carpeta SincroHCE donde se va descomprimir SincroHCE.rar (PATH\_SINCROHCE). Debería quedar la siguiente estructura:



## Archivos de ejecución

Bajo SincroHCE3.4 se van a encontrar dos archivos de ejecución:



- *PrcSincroHCE\_Personas.sh*: Envía los datos de persona al INUS, a través del AppliancePS
- *PrcSincroHCE\_CDAs.sh*: Genera y firma documentos para luego enviarlos al XDS local

### PrcSincroHCE\_Personas.sh

```
#!/bin/bash

# Ruta donde se van a almacenar los logs
PATH_LOG=/var/log/SincroHCE

# Ruta donde se encuentran los fuentes y librerías liberadas
PATH_GX=/home/genexus/sincroHCE

# Contraseña del trustore
PASSWORD_TRUSTSTORE=prestador1.123

# Contraseña del keystore
PASSWORD_KEYSTORE=prestador1.123

# Archivo trustStore (ruta+archivo)
FILE_TRUSTSTORE=/opt/jboss/domain/configuration/prestador1-truststore.jks

# Archivo keystore (ruta+archivo)
FILE_KEYSTORE=/opt/jboss/domain/configuration/prestador1-keystore.jks

LOGFILE=$PATH_LOG/prcSincroHCE_PERS_$(date +%Y_%m_%d).log

JAVA_CLASSPATH="lib/*:sincroHCE.jar:"

function EnviaLog {
    FECHA=$(date +%d-%m-%y) ("H:%M:%S") - ""
    echo -e "$FECHA $1 \n" >> $LOGFILE
}

EnviaLog "Inicia el proceso de envio de pesonas";
java -mx512m -cp $JAVA_CLASSPATH -Djavax.net.ssl.trustStore=$FILE_TRUSTSTORE -Djavax.net.ssl.trustStorePassword=$PASSWORD_TRUSTSTORE -Djavax.net.ssl.keyStore=$FILE_KEYSTORE -Djavax.net.ssl.keyStorePassword=$PASSWORD_KEYSTORE com.sincrohce.aenviopacientes >> $LOGFILE 2>&1
EnviaLog "Finalizó el proceso de envio de personas";

exit 0
```

### PrcSincroHCE\_CDAs.sh

```
#!/bin/bash

# Ruta donde se van a almacenar los logs
PATH_LOG=/var/log/SincroHCE

# Ruta donde se encuentran los fuentes y librerías liberadas
PATH_GX=/home/genexus/SincroHCE3.3

# Contraseña del trustore
PASSWORD_TRUSTSTORE=prestador1.123

# Contraseña del keystore
PASSWORD_KEYSTORE=prestador1.123

# Archivo trustStore (ruta+archivo)
FILE_TRUSTSTORE=/opt/jboss/domain/configuration/prestador1-truststore.jks

# Archivo keystore (ruta+archivo)
FILE_KEYSTORE=/opt/jboss/domain/configuration/prestador1-keystore.jks

LOGFILE=$PATH_LOG/prcSincroHCE_CDA_$(date +%Y_%m_%d).log

JAVA_CLASSPATH="lib/*:sincroHCE.jar:"

function EnviaLog {
    FECHA=$(date +%d-%m-%y) ("H:%M:%S") - ""
    echo -e "$FECHA $1 \n" >> $LOGFILE
}

EnviaLog "Inicia el proceso de envio de CDAs";
java -mx512m -cp $JAVA_CLASSPATH -Djavax.net.ssl.trustStore=$FILE_TRUSTSTORE -Djavax.net.ssl.trustStorePassword=$PASSWORD_TRUSTSTORE -Djavax.net.ssl.keyStore=$FILE_KEYSTORE -Djavax.net.ssl.keyStorePassword=$PASSWORD_KEYSTORE com.sincrohce.aenviocda >> $LOGFILE 2>&1
EnviaLog "Finalizó el proceso de envio de CDAs";

exit 0
```