



MANUAL TECNICO
TERRANUM BUSINESS PLATAFORM
Terranum

Wilson Amaya
Desarrollador
Koghi S.A.S.

April 2, 2012

0.1. CONTROL DE CAMBIOS

0.1 CONTROL DE CAMBIOS

Versión	Autor	Descripción	Fecha
1.0	Wilson Favian Amaya	Versión Inicial	April 2, 2012

0.2. PROPOSITO DEL DOCUMENTO

0.2 PROPOSITO DEL DOCUMENTO

Este documento tiene como propósito aclarar aspectos técnicos de la aplicación.

0.3 ALCANCE DEL DOCUMENTO

En este documento se describirán algunas restricciones de la aplicación y se listaran los datos que deben ser ingresados por la base de datos y la forma de ingresarlos a la misma.

Contents

0.1	CONTROL DE CAMBIOS	1
0.2	PROPOSITO DEL DOCUMENTO	2
0.3	ALCANCE DEL DOCUMENTO	3
0.4	Conexión a la base de datos	5
0.5	Cobro Anticipado.	5
0.6	Original - Copia	6
0.7	Ruta de almacenamiento de los archivos SIIGO y los PDF . .	6
0.8	Reporteador	6
0.9	Archivo de Conexión a la Base de Datos	13
0.10	Configuración de Jboss	13
0.11	Detección de Errores	14
0.12	Obtener la Fecha de un Cambio en Base de Datos	14

0.4 Conexión a la base de datos

En este manual se mencionaran Scripts que deben ser ejecutados en la base de datos, para poder ejecutar estos Scripts se debe tener una conexión previa a la base de datos, esta conexión se realiza ejecutando los siguientes comandos en la consola del servidor.

1. su (ingresar la contraseña de root)
2. su postgres
3. psql _U postgres terranvm_db

0.5 Cobro Anticipado.

- Se realizara una liquidación anticipada de N días para las hojas de términos con cobros anticipados, así que se debe tener en cuenta que el sistema adelantará (virtualmente) la fecha a la hora de calcular los conceptos anticipados, por ende si se crean y aprueban facturas, entre el día en que se realiza esta liquidación anticipada y el inicio del nuevo mes, estas tendrán un consecutivo de facturación superior al de las facturas ya generadas, pero serán del mes anterior (ya que no se haría adelanto virtual de fecha) por ende, no se recomienda crear y aprobar facturas del mes actual, si ya se generaron y aprobaron facturas anticipadas del mes entrante, para que no tengan problemas de consecutivos.
 - Para ingresar a la base de datos los días que se adelantara el sistema a la hora de realizar los cobros anticipados, se deben seguir los siguientes pasos.
1. SELECT * FROM days_for_early_payment;(al ejecutar esta consulta veremos los datos actuales de la tabla)
 2. UPDATE days_for_early_payment set days=N where id=?(donde N debe ser reemplazado por el número de días que se adelantara el sistema, y ? debe ser reemplazado por el número del id que le corresponde al mes que se desea actualizar.)

0.6 Original - Copia

- Los datos de las facturas que dicen quien se quedara con el original, quien con la copia1, y quien con copia2 son ingresados por bases de datos usando los siguientes Scripts.

1. UPDATE business_entity set original ='X1';
2. UPDATE business_entity set first_copy ='X2';
3. UPDATE business_entity set second_copy ='X3';

Donde las variables X1, X2, X3 serán reemplazadas por los datos de las entidades que se quedaran con la original, copia1 y copia2 respectivamente.

0.7 Ruta de almacenamiento de los archivos SI-IGO y los PDF

- Para ingresar esta ruta a la base de datos se deberán usar los siguientes Scripts.

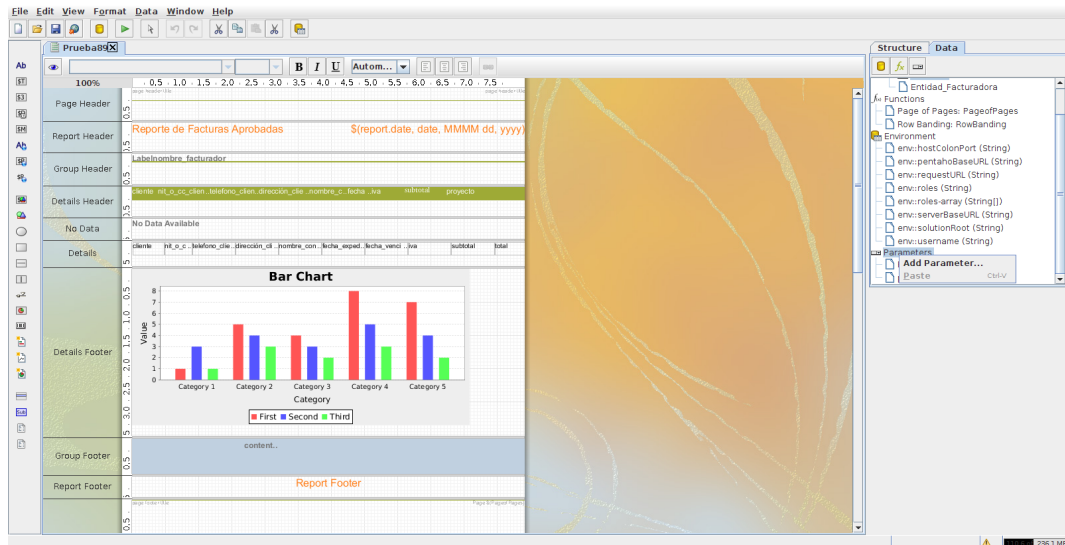
1. UPDATE system_configuration SET value ='X1' WHERE name='Carpeta_PDF';
2. UPDATE system_configuration SET value ='X2' WHERE name='Carpeta_SIIGO';

Donde las variables X1 será reemplazada por la ruta donde se almacenaran los PDF y X2 será reemplazada por la ruta donde se almacenaran las interface SIIGO.

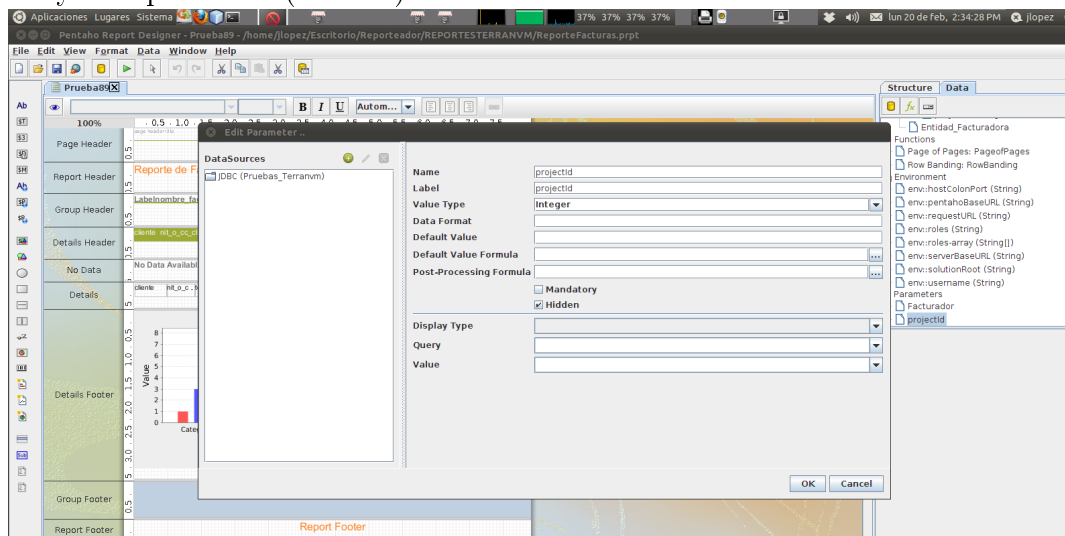
todas las carpetas indicadas en estas rutas deben existir previamente en el servidor, y las rutas no deben terminar en “/”.

0.8 Reporteador

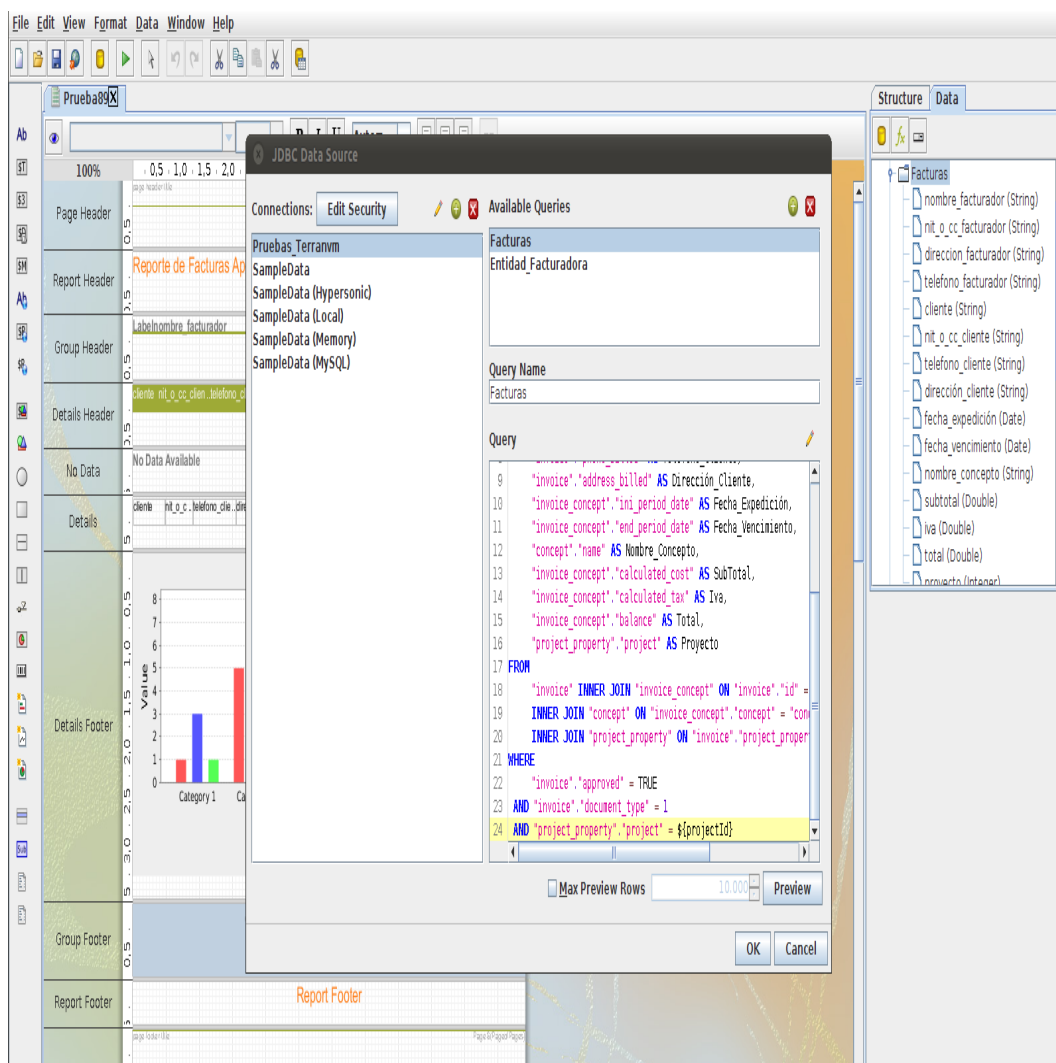
Para poder insertar un reporte a Orbis, es necesario generarlo desde la aplicación Pentaho, una vez hecho esto se debe crear un parametro, para esto se da click en data, en la parte inferior de data esta la opción parameters, se debe dar click derecho sobre esta opción, y add Parameter.



ahora se configurara el parámetro con el nombre projectId, será de tipo entero y de tipo oculto (Hidden)



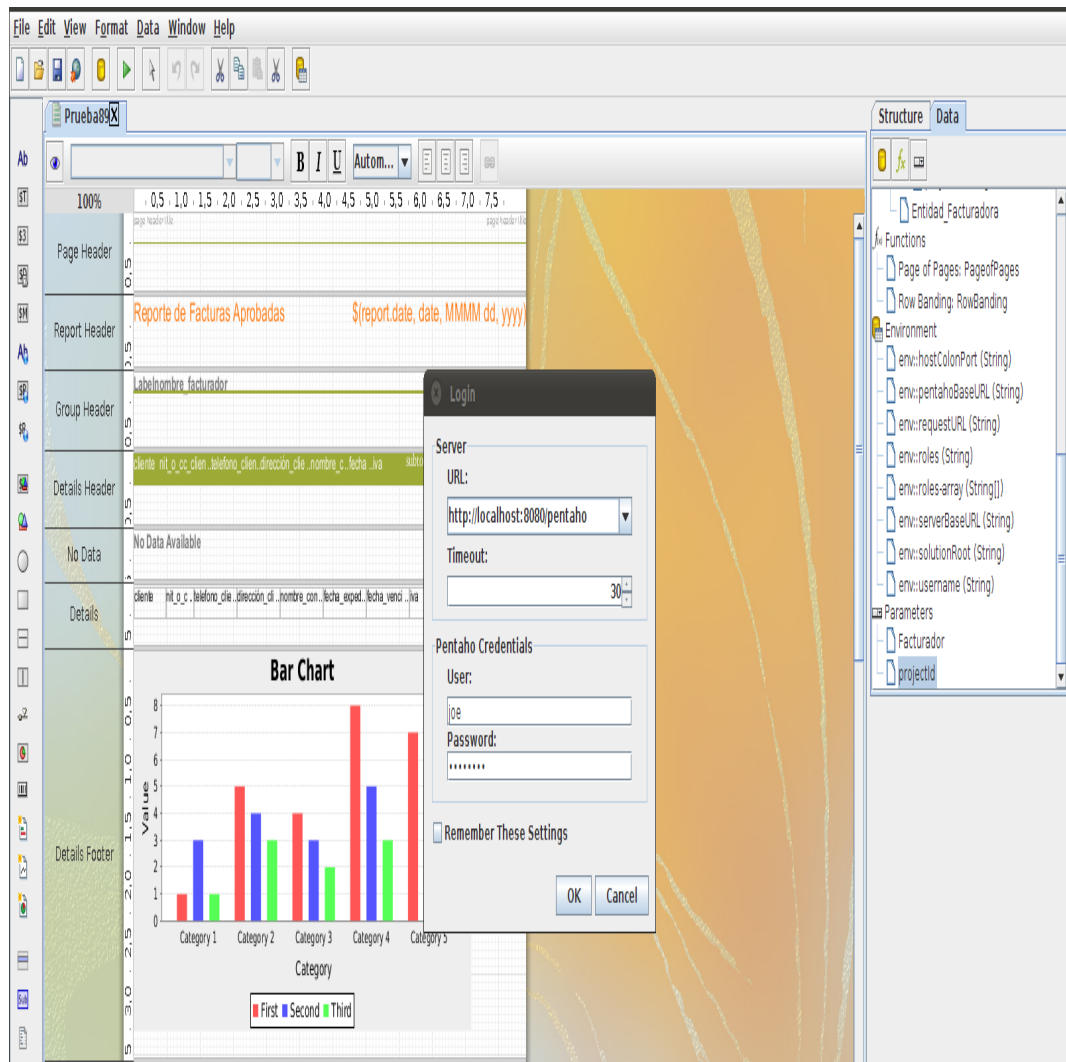
Una vez configurado este parámetro, modificaremos la consulta que realiza Pentaho, para que tome el parámetro creado y lo utilice en sus filtros.



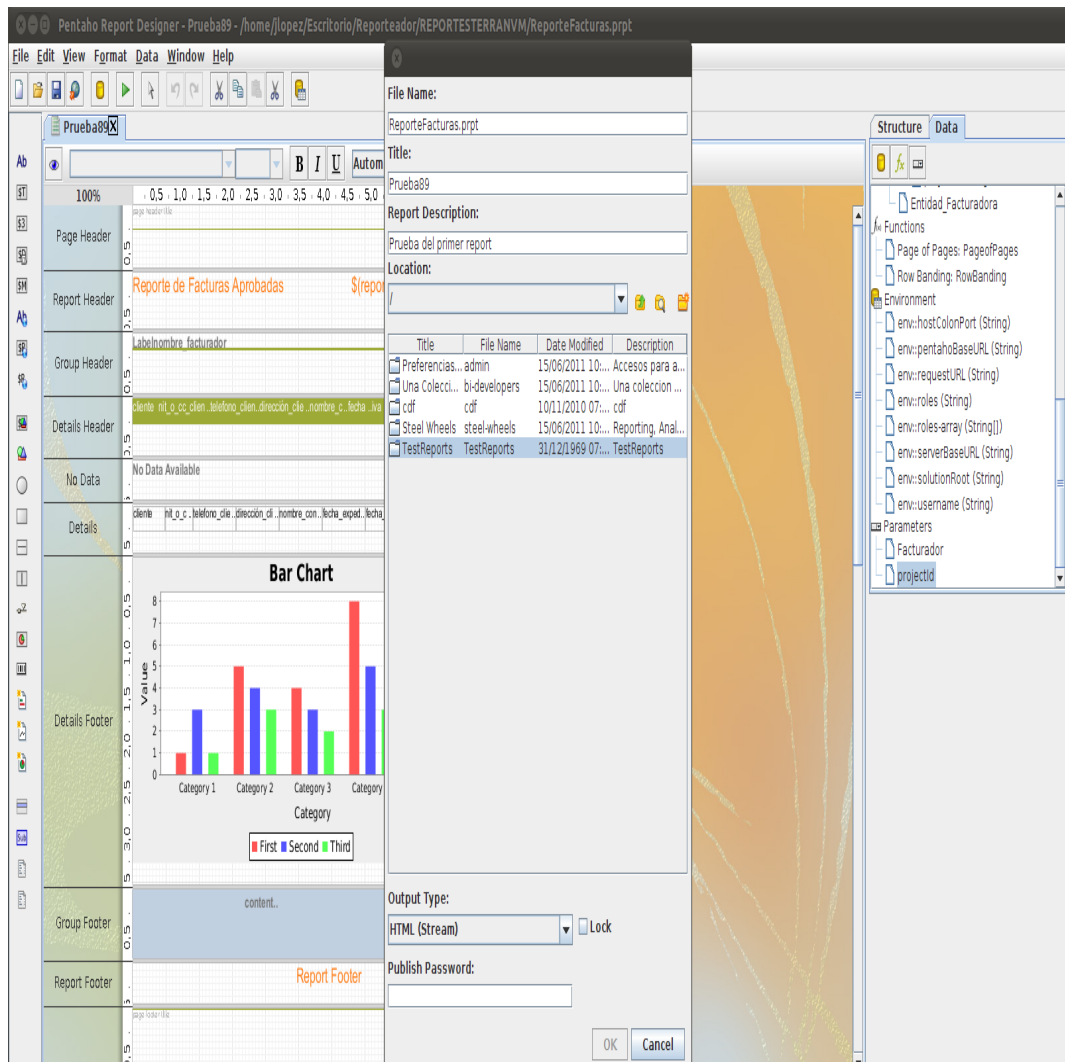
La forma de utilizar este parámetro varía según cada consulta, en este ejemplo, se consultan invoice, invoice_concept y project_property, ya que project_property almacena el id del proyecto, hacemos que solo se consulten los datos de project_property donde el project_property.project sea igual al id del proyecto.

una vez modificada la consulta, se procede a publicar el archivo al servidor de Pentaho.

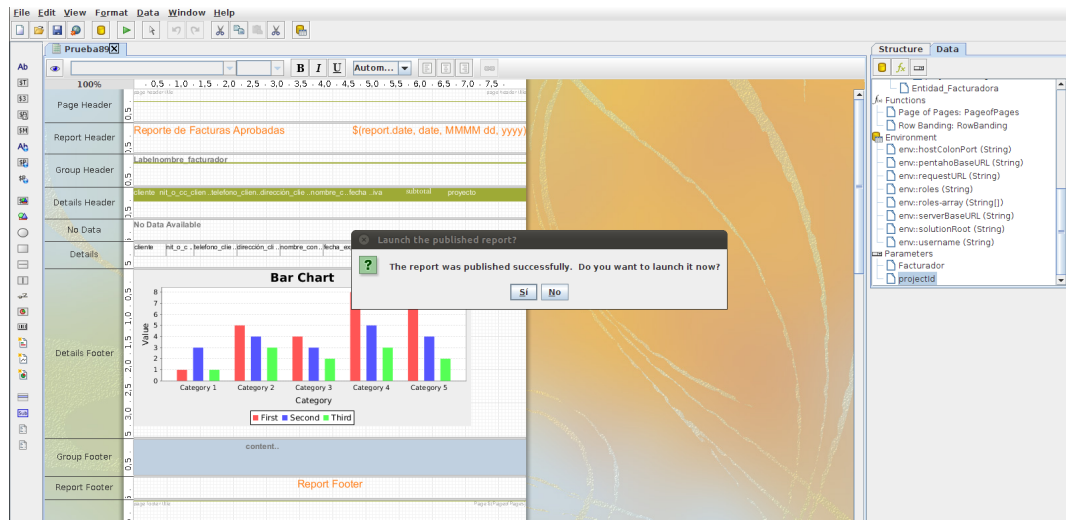
Se puede publicar el reporte dando click en File y luego en Publish.



se ingresa el usuario, contraseña y se da click en ok.



Aquí se selecciona la carpeta en la que se almacenará el reporte, en la parte de abajo piden un password de publicación, este es el password que se configuró en `/ruta_Pentaho/biserver-ce/pentaho-solutions/system/publisher_config.xml` donde `ruta_pentaho` es la ruta donde se encuentra la carpeta del servidor de pentaho, se da click en ok.



se debe mostrar el mensaje anterior.

Report Parameters

Facturador
GENERAL DE EQUIPOS DE COLOMBIA S.A. GECOLSA S.A.

Output Type
HTML (Paginated)

View Report ☒ Auto-Submit

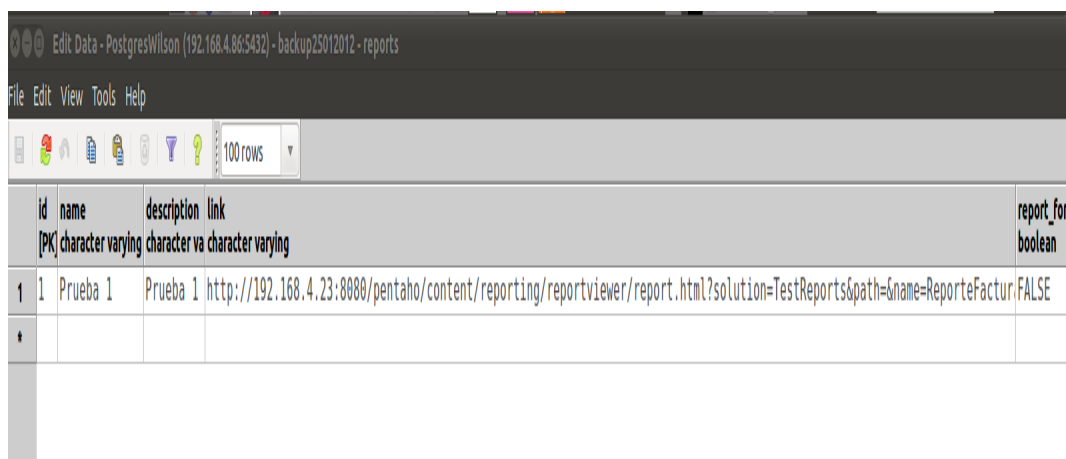
Page 1 of 1

Reporte de Facturas Aprobadas febrero 20, 2012
TERRANUM PRUEBA

cliente	nit_o_cp_cliente	telefono_cliente	direccion_cliente	nombre_cliente	fecha	iva	subtotal20
CLIENTE PRUEBA 1	900800..	5405433	CALLE 70 NO. 4 - 70	PRIMER ANTICIPO HONORARIOS DE ADMINISTRACION DELEGADA	05-08-2011	04-09-2011	0,00000000.. 14.546.244,...
CLIENTE PRUEBA 1	900800..	5405433	CALLE 70 NO. 4 - 70	SEGUNDO ANTICIPO HONORARIOS DE ADMINISTRACION DELEGADA	05-08-2011	04-09-2011	0,00000000.. 10.909.683,...
CLIENTE PRUEBA 2	800700..	9999999	Cll 99 No. 88 - 77	FONDO COMUN DE GASTOS	07-10-2011	06-11-2011	160.000.000.. 1.000.000,0,...
CLIENTE PRUEBA 1	900800..	5405433	CALLE 70 NO. 4 - 70	SEGUNDO ANTICIPO HONORARIOS DE ADMINISTRACION DELEGADA	05-08-2011	04-09-2011	0,00000000.. 9.709.617,8,...

copiamos la url de la página que nos muestra el reporte en el navegador, (para esto seleccionamos la opción de mostrar el reporte publicado)

0.9. ARCHIVO DE CONEXIÓN A LA BASE DE DATOS CONTENTS



	id	name	description	link	report_for
	[PK]	character varying	character va	character varying	boolean
1	1	Prueba 1	Prueba 1	http://192.168.4.23:8080/pentaho/content/reporting/reportviewer/report.html?solution=TestReports&path=&name=ReporteFactur	FALSE
*					

se ingresa en la base de datos de Orbis, y se insertan los siguientes datos en la tabla reports, se ingresa el id, el nombre del reporte, la descripción y la url completa del mismo, por último se debe ingresar true o false, si se ingresa true significa que este reporte no se discrimina por proyecto, de lo contrario se ingresara false.

0.9 Archivo de Conexión a la Base de Datos

El archivo de configuración a la base de datos se llama terranvm-ds.xml, este se encuentra en la carpeta de Jboss, en la ruta server/default/deploy

en este archivo se debe configurar el nombre de la base de datos, el usuario y el password de conexión.

0.10 Configuración de Jboss

Jboss tiene dos archivos de configuración que deben ser editados, el primero modificara la cantidad de memoria Ram que sera asignada a la aplicación,

este archivo se encuentra en la carpeta de Jboss /bin y se llama run.conf, lo abrimos con un editor de texto, y editamos la linea JAVA_OPTS=

Aquí cambiamos los valores de Xms, Xmx, asignándole los valores de memoria mínima y máxima que podrá usar la aplicación,

El segundo Archivo esta dentro de la carpeta de Jboss en la ruta /server/default/deploy/ y se llama transaction-jboss-beans.xml, en este archivo buscamos la linea que contiene la siguiente expresión “<property name=”defaultTimeout” esta expresión indica el tiempo limite de ejecución de una tarea, por ejemplo Cron

Job, así que se debe cambiar el valor numérico de esta línea a una cantidad de segundos superior al tiempo máximo de ejecución de todas las tareas que realiza la aplicación.

0.11 Detección de Errores

Para poder detectar un error en la aplicación es necesario leer los logs que se encuentran almacenados en la carpeta de Jboss dentro de la ruta `/server/default/log/` y el nombre del archivo es `server.log`.

0.12 Obtener la Fecha de un Cambio en Base de Datos

Para obtener la fecha de un cambio en una tabla de la base de datos que este siendo auditada, es necesario crear un procedimiento almacenado como el siguiente: `CREATE OR REPLACE FUNCTION from_unixtime(double precision) RETURNS TIMESTAMP AS ' SELECT to_timestamp($1)::timestamp AS result ' LANGUAGE 'SQL'`; luego de esto es necesario realizar la siguiente consulta: `select FROM_UNIXTIME(revtstmp/1000) from revinfo where rev = ?`; donde el signo de interrogacion debera ser reemplazado por el id al cual corresponde el cambio que se desea consultar. Para conocer cual es valor del id correspondiente a un cambio, se debera dirigir a la tabla auditada de la cual desee conocer la fecha del cambio; dentro de los campos de esta tabla se encuentra el campo `rev`; se debera revizar cual es el numero correspondiente al cambio que quiera consultar y agregar este valor a la consulta ya indicada.