

Comenzando con Pentaho

Prólogo

Antes de comenzar este segundo tutorial sobre Pentaho quisiera pedir disculpas por la demora pero el tiempo me queda corto, ahora si..... continuemos con Pentaho; habíamos dejado el tutorial anterior con la instalación de Pentaho en un Fedora 7 (revisen el tutorial anterior ya que mencionamos detalles importantes acerca de la instalación), en este tutorial trataremos los siguientes temas:

- Acceso a Pentaho desde un cliente.
- Conexión a nuestra base de datos MySQL
- Nuestro Primer Reporte

1. Acceso a Pentaho desde un cliente

Aquellos que instalaron Pentaho con el tutorial anterior habrán notado que desde el servidor donde instalación Pentaho no había ningún problema para acceder; es decir bastaba escribir en nuestro Browser <http://localhost:8080> y ya nos mostraba la suite de Pentaho; sin embargo aquellos que querían acceder desde otra computadora a la suite de Pentaho se llevaban con la sorpresa de que no se podía acceder, pues el problema radica en que JBOOS no esta configurado para ser accedido desde otra computadora, así que a configurarlo.

Para aquellos que no han entendido lo que trate de decirles líneas arriba, aquí les dejo unas imágenes para que se den cuenta del problema (ya que tarde o temprano tendrán que pasar por este problema).

En la figura 1 se muestra la suite de Pentaho corriente sobre Fedora 7 que se encuentra en el VmWare, aquí no hay ningún problema todo aparentemente esta correcto.

En la figura2 les muestro lo que ocurre cuando trato de acceder a la suite de Pentaho desde mi Windows 2003 Server, pues.... No puedo acceder.

Espero con esto haya quedado claro lo que tratamos de solucionar.

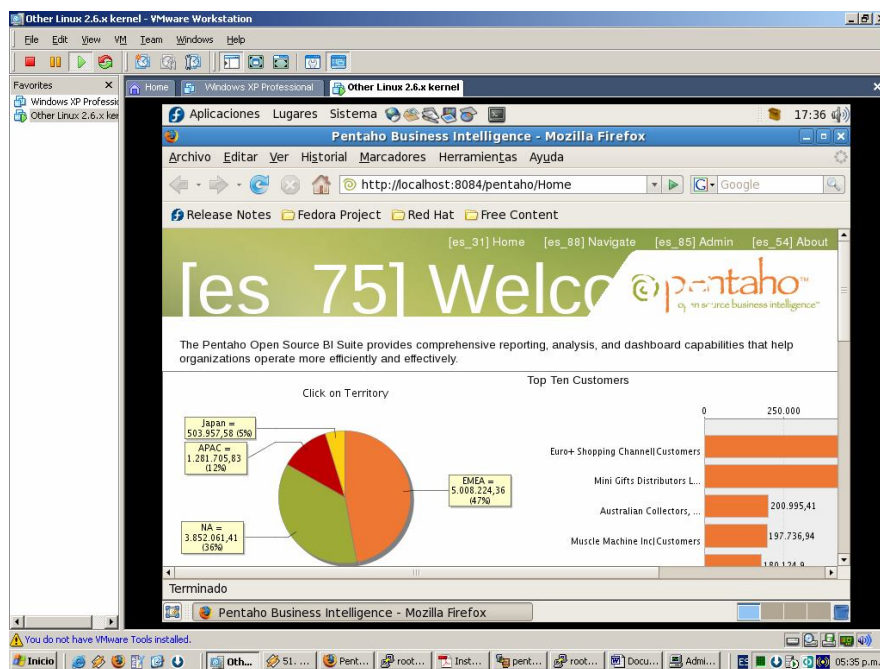


Figura1.- Pentaho en Fedora 7

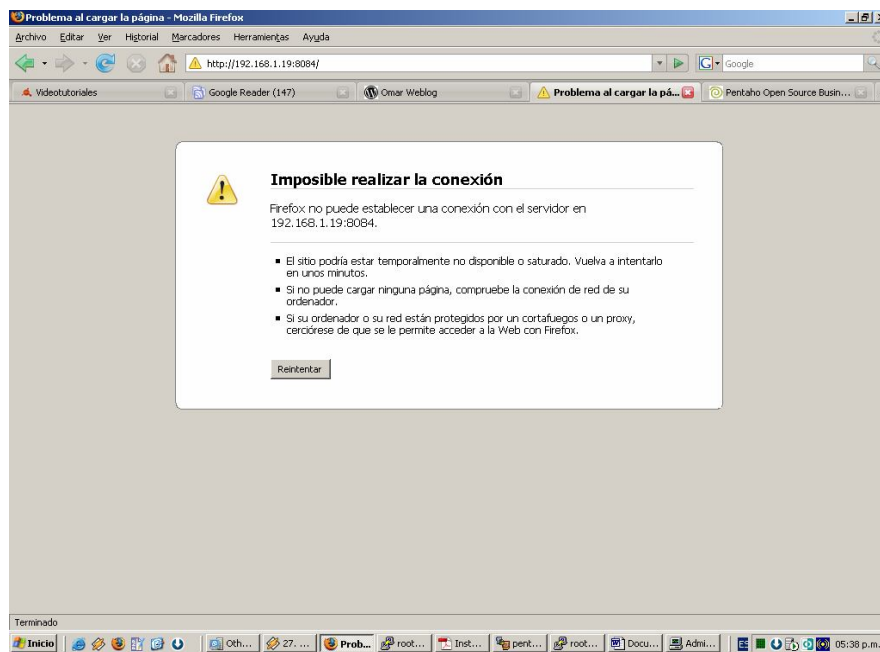


Figura2.- Intentando acceder a Pentaho desde Windows (Cliente)

Para solucionar esto debemos configurar el archivo web.xml que se encuentra en la siguiente ruta:

`%PCI%/jboss/server/default/deploy/pentaho.war/WEB-INF`

Nota: Si han instalado siguiendo exactamente el tutorial anterior %PCI% debe ser reemplazado por /opt y para los que lo están haciendo en Windows deben reemplazarlo por c:/Archivos de Programa/Pentaho

Dentro de la ruta mostrada existe un archivo llamado web.xml, este archivo en la linea 384 (aprx) debe mostrar lo siguiente:

```
<param-value>pentaho-solutions</param-value>
</context-param>

<context-param>
  <param-name>base-url</param-name>
  <param-value>http://localhost:8084/pentaho</param-value>
</context-param>

<context-param>
  <param-name>locale-language</param-name>
  <!-- for example 'fr' for French, 'de' for German -->
  <param-value></param-value>
```

`<param-value>http://localhost:8080/Pentaho</param-value>`

esta etiqueta debemos modificarla y en vez de localhost, debemos colocar el ip de la maquina donde esta instalado Pentaho, en mi caso quedaria de esta manera:

`<param-value>http://192.168.1.19:8080/Pentaho</param-value>`

Después de haber grabado los cambios debemos reiniciar Pentaho:

<code>/opt/pentaho-opensource-1.2.1/stop-pentaho.sh</code>	Parando Pentaho
<code>/opt/pentaho-opensource-1.2.1/start-pentaho.sh</code>	Iniciando Pentaho

Después de 3 minutos aproximadamente Pentaho habrá iniciado y ahora si podremos acceder desde cualquier host a la suite de Pentaho, la figura3 muestra como podemos acceder a Pentaho desde Windows.



Figura3.- Pentaho desde Windows.

2. Conexión a nuestra base de datos MySQL

Requerimientos

- Descargar el conector de MySQL: *mysql-connector-java-3.1.12-bin.jar*, aunque en la versión que estamos usando ya viene incorporada.
- Tener instalado el servidor de base de datos Mysql

Primer Paso

Debemos crear un JNDI para nuestra conexión a MySQL, para esto debemos crear un archivo con el siguiente nombre (el nombre puede variar según ustedes gusten): *conexion-ds.xml*, donde *conexion* es el nombre de mi JNDI.

vi opt/pentaho-opensource/jboss/server/default/deploy/conection-ds.xml, y escribimos lo siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<datasources>
```

```
<local-tx-datasource>
```

```
<jndi-name>conexion</jndi-name>
```

```
<connection-url>jdbc:mysql://localhost/pentaho</connection-url>
```

```
<driver-class>com.mysql.jdbc.Driver</driver-class>
```

```
<user-name>user_pentaho</user-name>
```

```
<password>password</password>
```

```
</local-tx-datasource>
```

```
</datasources>
```

Nombre del JNDI

Nombre de la base de datos a la q accederemos con Pentaho

Usuario con privilegios para acceder a la base de datos declarada arriba.

Password del usuario declarado arriba

Segundo Paso

Agregar las siguientes líneas al archivo web.xml con el comando vi:

vi /opt/pentaho-opensource-1.2.1/jboss/server/default/deploy/pentaho.war/WEB-INF/web.xml

Agregamos al final y antes de la última etiqueta:

```
<resource-ref>
  <description>Mi primera conexion</description>
  <res-ref-name>jdbc/conexion</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
</resource-ref>
```

Nombre del JNDI
que estamos usando:
conexion

Tercer Paso

Agregar del mismo modo que en los anteriores casos las siguientes líneas al archivo jboss-web.xml usando el comando vi:

vi /opt/pentaho-opensource-1.2.1/jboss/server/default/deploy/pentaho.war/WEB-INF/jboss-web.xml

```
<resource-ref>
  <res-ref-name>jdbc/conexion</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <jndi-name>java:/conexion</jndi-name>
</resource-ref>
```

Nombre del JNDI que
hemos declarado

Cuarto Paso

Verificar que el conector de mysql se encuentre en la siguiente ruta:

```
[root@KALYPSO deploy]# ls /opt/pentaho-opensource-1.2.1/jboss/server/default/lib | grep mysql
mysql-connector-java-5.0.5-bin.jar
```

Pues..... ya esta, eso es todo!! Ya tenemos configurado Pentaho para conectarse a nuestra base de datos MySQL, con esto haremos nuestro primer reporte (lo mas simple que se pueda solo para verificar que la conexión ya esta hecha).

3. Nuestro Primer Reporte

Requerimientos

Tener instalado MySQL y crear lo siguiente:

- Crear una base de datos llamada: pentaho
- Crear un usuario (con privilegios para esa base de datos): pentaho_user

Nota: No voy a explicar como crear la base de datos y el usuario porque extenderia mucho el tutorial, pero.... creo que la manera mas facil es usando PHPMYADMIN.

Primer paso

Después de que hemos creado el usuario y la base de datos, creamos el archivo para nuestro primer reporte, de la siguiente manera:

```
root@KALYPSO lib# vi /opt/pentaho-opensource-1.2.1/pentaho-solutions/samples/datasources/conection.xaction
```

y agregamos lo siguiente:

```
<?xml version="1.0" encoding="UTF-8"?>
<action-sequence>
  <name>conection.xaction</name>
  <version>1</version>
  <title>Reporte</title>
  <loggin-level>debug</loggin-level>
  <inputs/>
  <outputs>
    <rule-result>
      <type>list</type>
    </rule-result>
  </outputs>
  <resources/>
  <actions>
    <action-definition>
      <action-outputs>
        <rule-result type="list"/>
      </action-outputs>
    <component-name>org.pentaho.component.SQLLookupRule</component-name>
    <action-type>rule</action-type>

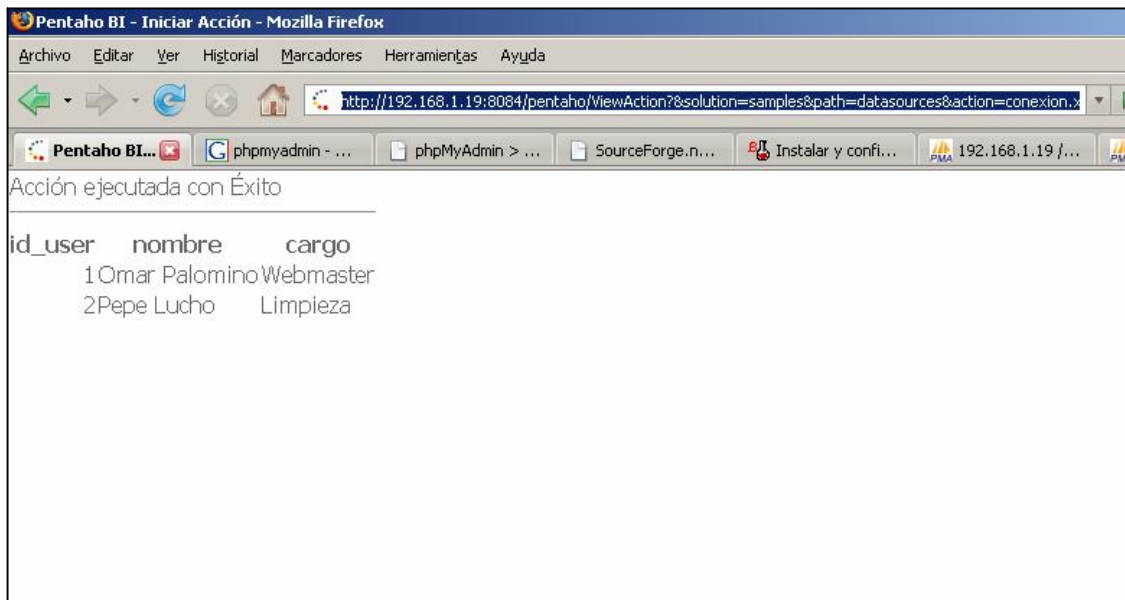
    <component-definition>
      <jndi>conexion</jndi>
      <query><![CDATA[select id_user, nombre, cargo from usuarios]]></query>
    </component-definition>
  </action-definition>
</actions>
</action-sequence>
```

Segundo Paso

Solo nos queda reiniciar Pentaho y probar el reporte:

<http://IP:puerto/pentaho/ViewAction?&solution=samples&path=datasources&action=conection.xaction>

y nos debe aparecer algo como esto:



Es obvio que es muy simple, pero lo que en verdad importaba con esto es la declaración del JNDI, ya que para los próximos tutoriales usaremos el mismo JNDI para hacer reportes con el Wizard de reportes con el que cuenta Pentaho, de la misma manera para poder crear los cubos usaremos el JNDI.

Nota:

Tengan mucho cuidado al momento de agregar todos los cambios que hemos hecho ya que con cualquier etiqueta mal cerrada, nos mostrara un mensaje de error.

..... para terminar:

No creo ser la única persona que está implementando pentaho, ya que en foro de pentaho en español muchas personas hablan de sus proyectos con esta herramienta, creo que estas personas deberían también hacer tutoriales para que todos los que queremos implementar de forma profesional Pentaho podamos ayudarnos mutuamente.

Y no se olviden de dejar sus comentarios en el Blog.