C3DWV
Collaborative 3D Web Viewer

Installation and Administration Guide

Author(s): Jan Sutter DFKI

Release Date: 04-08-2015
Revision: 1.0

Table of Contents

INTRODUCTION 4

1. SOFTWARE LICENSE 4

2. INSTALLING THE SOFTWARE 4

2.1. System Requirements 4
2.2. Download 4
2.3. Installation and Configuration 4
2.4. Sanity Check 10

3. ADMINISTRATION PROCEDURES 10

Introduction

This document describes the installation and configuration procedure for the Collaborative 3D Web Viewer. Please the user documentation for information on how to use C3DWV.

1. Software License

The Collaborative 3D Web Viewer is Open Source release under the Apach-2.0 License available at http://www.apache.org/licenses/LICENSE-2.0.

2. Installing the Software

2.1. System Requirements

As a client server application C3DWV has two sets of requirements, one for the server running the application server and one for the client running the visualization.
Server software requirements:

- Operating System: Windows 7 or Linux (Debian Derivate)
- Java7 runtime environment either Oracle or OpenJRE

Minimum server hardware requirements:

- Typical multi-core CPU with at leas 1.8 GHz
- 4 GB of RAM

Client Software requirements:

- Recent version of Chrome (>Version 42)

Minimum client hardware requirements:

- Graphics card not blacklisted by the Chrome WebGL renderer. List of blacklisted graphics cards available at https://www.khronos.org/webgl/wiki/BlacklistsAndWhitelists#Chrome

2.2. Download

Source Code available on Github: https://github.com/dfki-asr-fitman/c3dwv.git
Binary Distribution: <link>

2.3. Installation and Configuration

Download the Wildfly JavaEE application server Version 8.2 binary distribution at: http://wildfly.org/downloads/. Extract the downloaded archive to a folder. For the remainder of this guide that folder will be indicated as `YOUR_WILDFLY_ROOT`.

2.3.1. Add Users to Wildfly

You need to add a user to be able to manage the server via the web interface. To do this,

1. Open a console and start the server via the `standalone.[bat,sh]` script in the `bin` folder of `YOUR_WILDFLY_ROOT.` Either start the server in the background or open another console for the next step.
2. Execute `add-user` from the `bin` folder and answer the given questions as follows:
```
        Type: management user
        Name: <admin or another preferred name>
        Password: <some safe password>
        Groups: <leave empty>
        Used to connect to another process: no
```
3. Add another user and answer the given questions as follows:
```
        Type: application user
        Name: stomp-user
        Password: stomp-user-313
        Groups: guest
        Used to connect to another process: no
```

2.3.2. Add Datasource

Datasource is the JavaEE terminology and abstraction over databases and database like systems. To add a datasource for C3DWV open the administration interface of the Wildfly server in your browser. This administration interface is bound to localhost by default. If you cannot access the website on localhost because you do not have a window system installed on the machine you are installing C3DWV, either tunnel port 9990 through ssh, or change the bind-address for the administration interface. This, however, should only be done for the first setup and not for production use. To change the bind-address of the administration interface restart the Wildfly server as follows:

```
standalone.[bin,sh] -bmanagement=0.0.0.0
```

The administration interface should then be available at http://<ip or localhost>:9990. Use the previously created management user to login. Upon login click the configuration tab and choose Datasources under Connector in the tree on the left hand side (see Figure 1). Press “add” to create a new Datasource with the following values:

```
    Name: Compass
    JNDI Name: java:/compass-remote
```

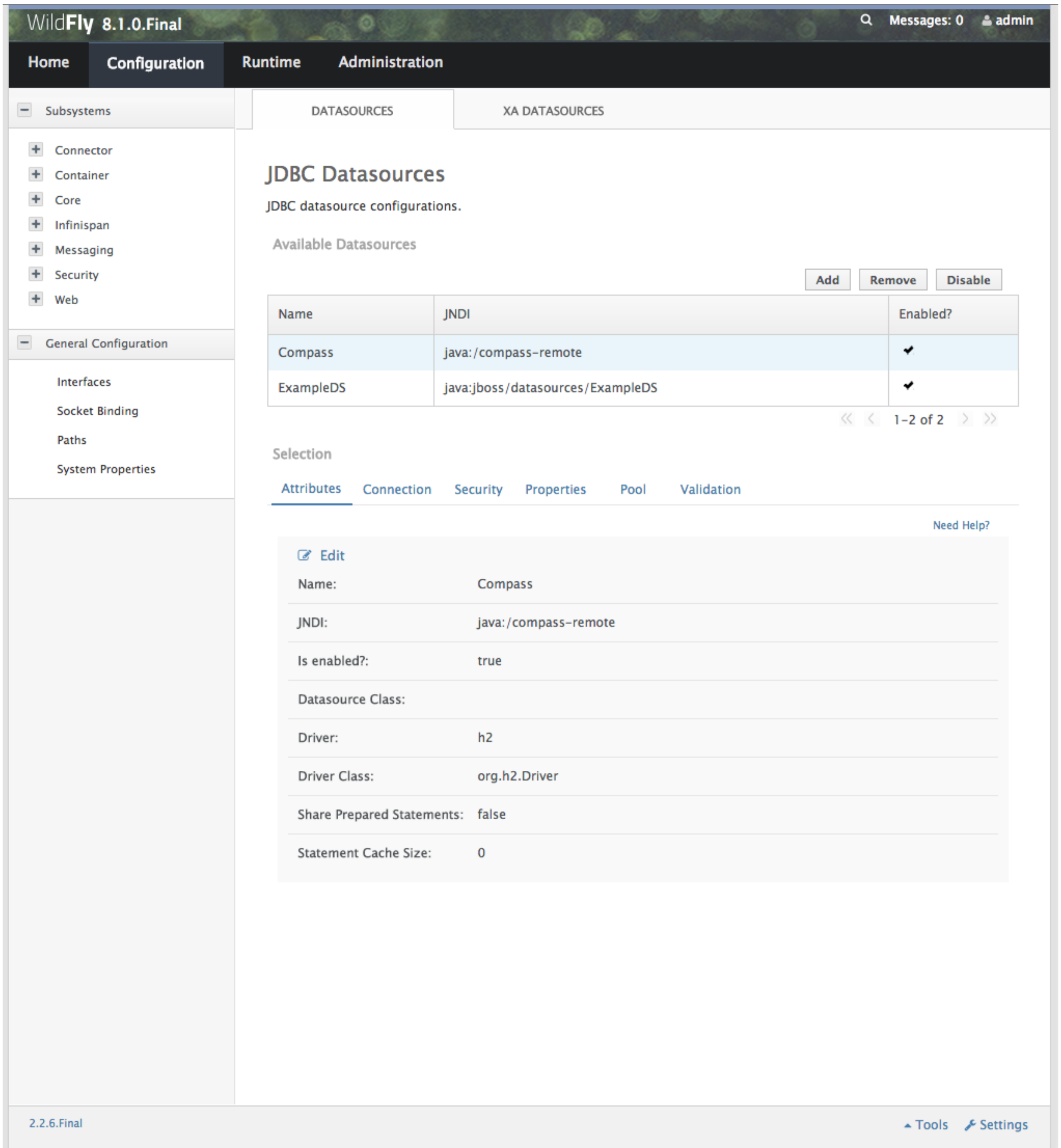
Then press next and choose “h2” as the driver. In the last step insert the following information:

```
    connection-url: jdbc:h2:mem:compass
    Username: sa
    Password: sa
    Security domain: leave empty
```

The above connection-url will create an in-memory database that will reset on each restart of the server. To persist the database in a file use the following:

```
    connection-url: jdbc:h2:<path>/compass
```

If the configuration was successful the Datasource can be enabled.



JDBC Datasources

JDBC datasource configurations.

Available Datasources

Name	JNDI	Enabled?
Compass	java:/compass-remote	✓
ExampleDS	java:jboss/datasources/ExampleDS	✓

Selection

Name:	Compass
JNDI:	java:/compass-remote
Is enabled?:	true
Datasource Class:	
Driver:	h2
Driver Class:	org.h2.Driver
Share Prepared Statements:	false
Statement Cache Size:	0

Figure 1: Configuring a JDBC Datasource.

2.3.3. Messaging System Configuration

C3DWV uses the Java Messaging System for communication between the client application and the server. To setup the respective queues open the standalone.xml at `YOUR_WILDFLY_ROOT/standalone/configuration` inside a text editor. At the extensions section add:

<extension module="org.jboss.as.messaging"/>

Inside the <subsystem xmlns="urn:jboss:domain:ejb3:2.0"> add:

```
     <mdb>
     <resource-adapter-ref
          resource-adapter-name=
          "${ejb.resource-adapter-name:hornetq-ra.rar}"/>
     <bean-instance-pool-ref pool-name="mdb-strict-max-pool"/>
     </mdb>
```

Finally, add a new sybsystem for hornet by adding the following subsystem to the list of subsystems inside the profile:

```
<subsystem xmlns="urn:jboss:domain:messaging:2.0">
     <hornetq-server>
          <journal-file-size>102400</journal-file-size>
          <connectors>
               <http-connector name="http-connector" socket-
binding="http">
                    <param key="http-upgrade-endpoint"
value="http-acceptor"/>
               </http-connector>
               <http-connector name="http-connector-
throughput" socket-binding="http">
                    <param key="http-upgrade-endpoint"
value="http-acceptor-throughput"/>
                    <param key="batch-delay" value="50"/>
               </http-connector>
               <in-vm-connector name="in-vm" server-id="0"/>
               <connector name="netty-connector">
                    <factory-
class>org.hornetq.core.remoting.impl.netty.NettyConnectorFacto
ry</factory-class>
               </connector>
          </connectors>
          <acceptors>
               <http-acceptor http-listener="default"
name="http-acceptor"/>
               <http-acceptor http-listener="default"
name="http-acceptor-throughput">
                    <param key="batch-delay" value="50"/>
                    <param key="direct-deliver"
value="false"/>
               </http-acceptor>
               <in-vm-acceptor name="in-vm" server-id="0"/>
               <acceptor name="stomp-websocket">
```

```
                    <factory-
class>org.hornetq.core.remoting.impl.netty.NettyAcceptorFactor
y</factory-class>
                    <param key="host" value="0.0.0.0"/>
                    <param key="port" value="61614"/>
               </acceptor>
               <acceptor name="stomp">
                    <factory-
class>org.hornetq.core.remoting.impl.netty.NettyAcceptorFactor
y</factory-class>
                    <param key="host" value="0.0.0.0"/>
                    <param key="port" value="61613"/>
               </acceptor>
          </acceptors>
          <security-settings>
               <security-setting match="#">
                    <permission type="send" roles="guest"/>
                    <permission type="consume" roles="guest"/>
                    <permission type="createNonDurableQueue"
roles="guest"/>
                    <permission type="deleteNonDurableQueue"
roles="guest"/>
               </security-setting>
          </security-settings>
          <address-settings>
               <address-setting match="#">
                    <dead-letter-address>jms.queue.DLQ</dead-
letter-address>
                    <expiry-
address>jms.queue.ExpiryQueue</expiry-address>
                    <max-size-bytes>10485760</max-size-bytes>
                    <page-size-bytes>2097152</page-size-bytes>
                    <message-counter-history-day-
limit>10</message-counter-history-day-limit>
               </address-setting>
          </address-settings>
          <jms-connection-factories>
               <connection-factory
name="InVmConnectionFactory">
                    <connectors>
                         <connector-ref connector-name="in-
vm"/>
                    </connectors>
                    <entries>
                         <entry
name="java:/ConnectionFactory"/>
                    </entries>
               </connection-factory>
               <connection-factory
name="RemoteConnectionFactory">
                    <connectors>
                         <connector-ref connector-name="http-
connector"/>
```

```
                    </connectors>
                    <entries>
                         <entry
name="java:jboss/exported/jms/RemoteConnectionFactory"/>
                    </entries>
               </connection-factory>
               <pooled-connection-factory name="hornetq-ra">
                    <transaction mode="xa"/>
                    <connectors>
                         <connector-ref connector-name="in-
vm"/>
                    </connectors>
                    <entries>
                         <entry name="java:/JmsXA"/>
                         <entry
name="java:jboss/DefaultJMSConnectionFactory"/>
                    </entries>
               </pooled-connection-factory>
          </jms-connection-factories>
          <jms-destinations>
               <jms-queue name="ExpiryQueue">
                    <entry
name="java:/jms/queue/ExpiryQueue"/>
               </jms-queue>
               <jms-queue name="DLQ">
                    <entry name="java:/jms/queue/DLQ"/>
               </jms-queue>
               <jms-topic name="compass.projects">
                    <entry
name="java:/jms/topic/compass/projects"/>
               </jms-topic>
               <jms-topic name="compass.scenarios">
                    <entry
name="java:/jms/topic/compass/scenarios"/>
               </jms-topic>
               <jms-topic name="compass.sceneNodes">
                    <entry
name="java:/jms/topic/compass/sceneNodes"/>
               </jms-topic>
               <jms-topic name="compass.prefabSets">
                    <entry
name="java:/jms/topic/compass/prefabSets"/>
               </jms-topic>
               <jms-topic name="compass.sceneNodeComponents">
                    <entry
name="java:/jms/topic/compass/sceneNodeComponents"/>
               </jms-topic>
          </jms-destinations>
     </hornetq-server>
</subsystem>
```

2.4. Sanity Check

After the configuration start the server. If the installation procedure was successful C3DWV should be up and running at http://<ip>:8080/c3dwv.

3. Administration Procedures

C3DWV does not provide special configuration options for administrators. For security and user rights management related configurations of the JavaEE application server please consult the documentation of the Wildfly server at:
https://docs.jboss.org/author/display/WFLY9/Documentation