

JBOSS/Wildfly

Java Naming and Directory Interface

What is JBoss/Wildfly ?

- JBoss Application Server (or **JBoss AS**) is a free software/open-source **Java EE**-based application server.
- Not only implements a server that runs on Java, but it actually implements the Java EE part of Java.
- Because it is Java-based, the JBoss application server operates **cross-platform**: usable on any operating system that supports Java.
- JBoss AS was developed by JBoss, now a division of **Red Hat**.

JEE

- JEE provides an API and runtime environment for developing and running enterprise software, including network and web services, and other large-scale, multi-tiered, scalable, reliable, and secure network applications.
- Java EE extends the Java SE providing API for object-relational mapping, distributed and multi-tier architectures, and web services.
- The platform incorporates a design based largely on modular components running on an application server.

Key JEE Components

- **EJB** (Enterprise JavaBeans) define a distributed component system that is the heart of the Java EE specification . This system , in fact, provides the typical features required by enterprise applications , such as scalability, security , data persistence , and more.
- **JNDI** defines a system to identify and list generic resources , such as software components or data sources .
- **JDBC** is an interface for access to any type of data bases.
- **JTA** is a system for distributed transaction support .
- **JPA** is an API for the management of persistent data .
- **JAXP** is an API for handling files in XML format.
- **JMS** (Java Message Service) a system for sending and managing messages.

Installing Wildfly

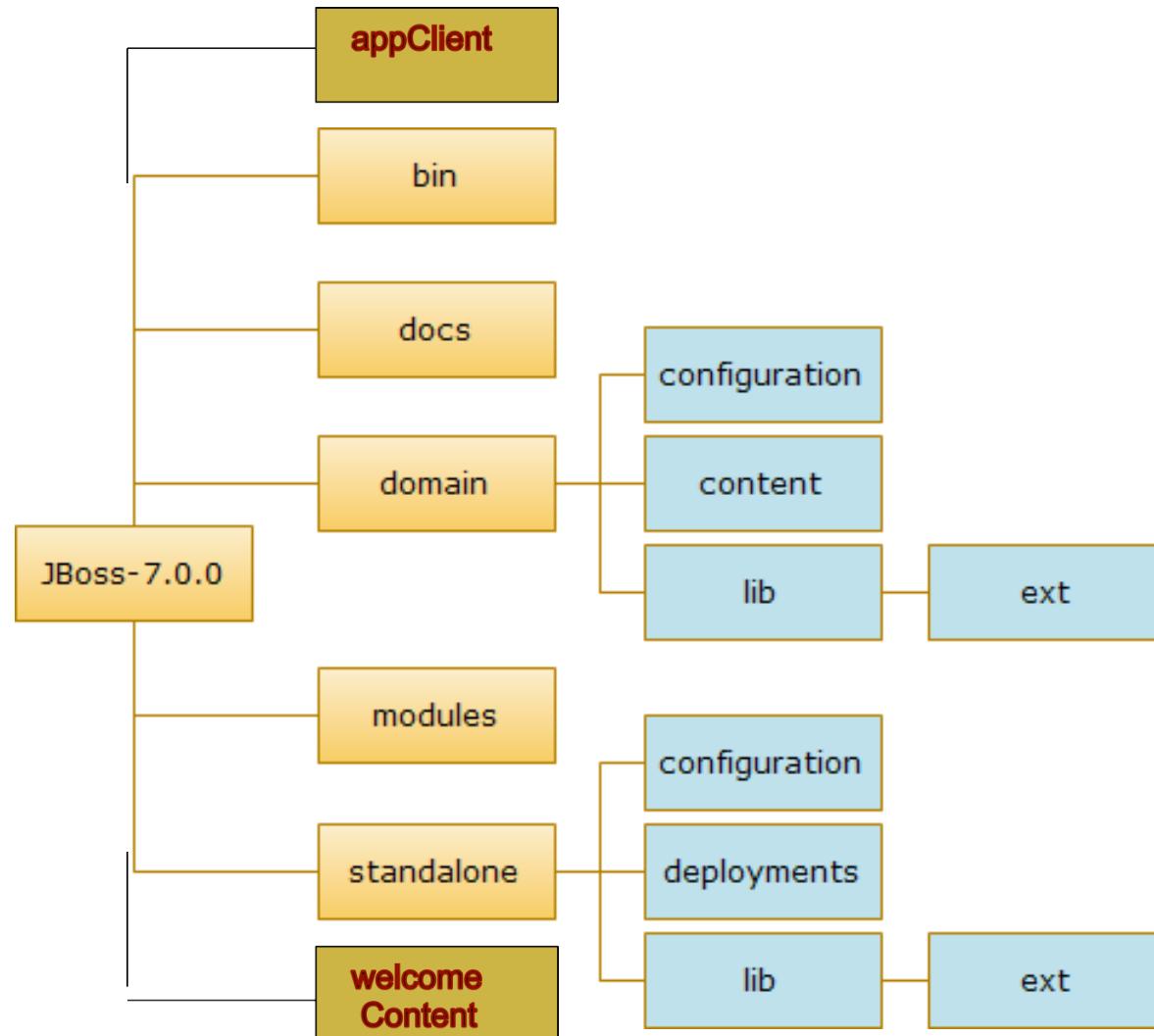
- <http://wildfly.org/downloads/>
- Download 14.01.Final
-

GETTING STARTED GUIDE:

<http://docs.wildfly.org/14/>

- unzip it (where you like: that will be your JBOSS_HOME – define it as a login variable)

Wildfly directory structure



See http://docs.wildfly.org/14/Getting_Started_Guide.html#wildfly---a-quick-tour

Starting and stopping Wildfly

cd into your JBOSS_HOME/bin

UNIX (LINUX-MAC)

- START: ./standalone.sh &
- STOP: ./jboss-cli.sh --connect command=:shutdown

WINDOWS

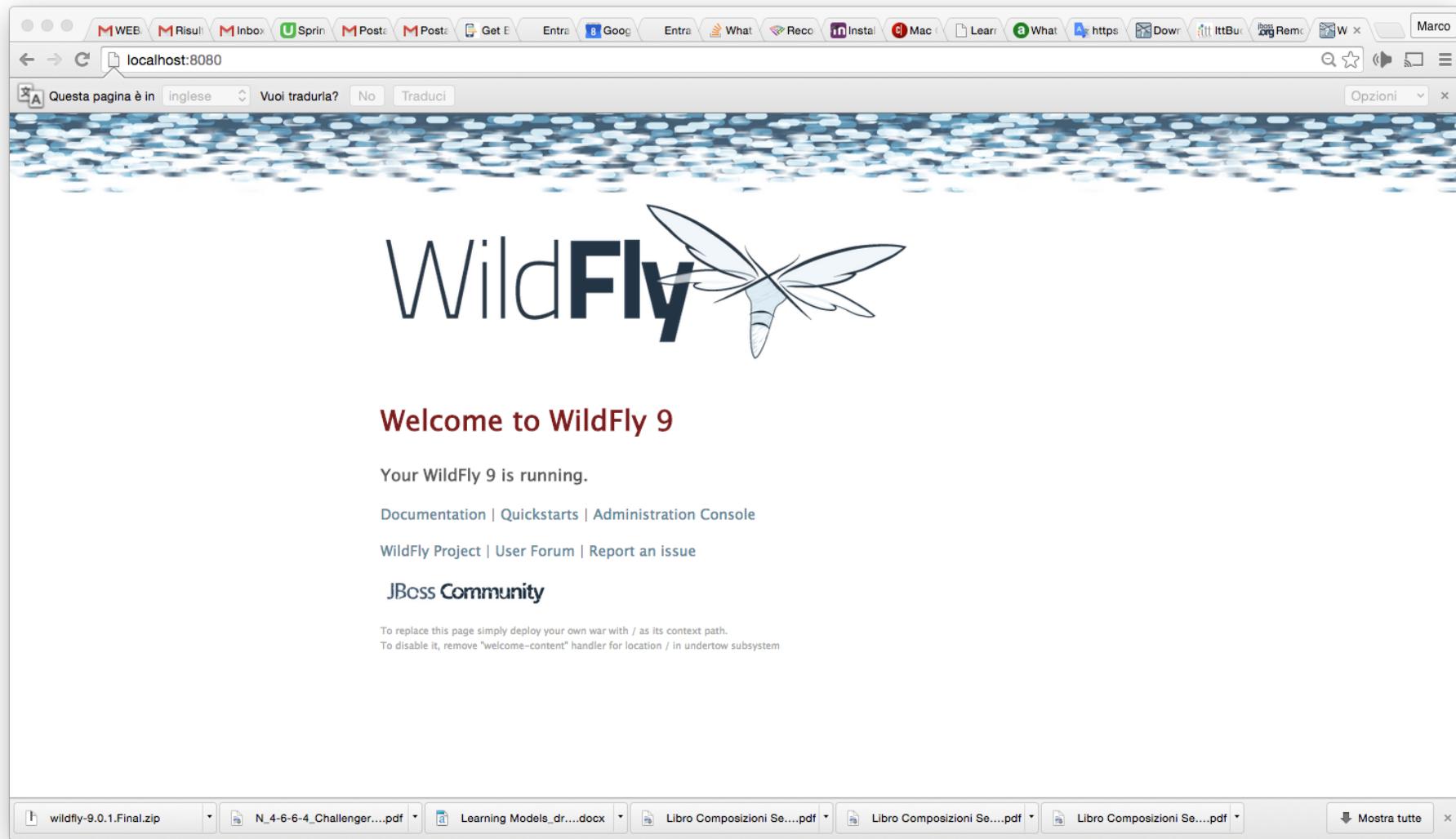
- START: ./standalone.bat
- STOP: ./jboss-cli.bat --connect command=:shutdown

On starting....

- =====
 - JBoss Bootstrap Environment
 - JBOSS_HOME: /Users/ronchet/Downloads/wildfly-9.0.1.Final
 - JAVA: java
 - JAVA_OPTS: -server -XX:+UseCompressedOops -server -XX:+UseCompressedOops -Xms64m -Xmx512m -XX:MaxPermSize=256m -Djava.net.preferIPv4Stack=true -Djboss.modules.system.pkgs=org.jboss.byteman -Djava.awt.headless=true
 - =====
 - Java HotSpot(TM) 64-Bit Server VM warning: ignoring option MaxPermSize=256m; support was removed in 8.0
 - 22:43:16,319 INFO [org.jboss.modules] (main) JBoss Modules version 1.4.3.Final
 - 22:43:16,565 INFO [org.jboss.msc] (main) JBoss MSC version 1.2.6.Final
 - 22:43:16,662 INFO [org.jboss.as] (MSC service thread 1-6) WFLYSRV0049: WildFly Full 9.0.1.Final (WildFly Core 1.0.1.Final) starting...
 - (...)
- 22:43:19,264 INFO [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly Full 9.0.1.Final (WildFly Core 1.0.1.Final) started in 3303ms - Started 205 of 382 services (211 services are lazy, passive or on-demand)



http://localhost:8080



Wildfly: Creating admin users

```
ronchet$ ./add-user.sh
```

What type of user do you wish to add?

- a)Management User (mgmt-users.properties)
 - b)Application User (application-users.properties)
- (a): **a**

Enter the details of the new user to add.

Realm (ManagementRealm) : - *leave this blank-*

Username : **admin**

Password : **pw**

Re-enter Password : **pw**

About to add user 'admin' for realm 'ManagementRealm'

Is this correct yes/no? **yes**

Added user 'admin' to file '/Users/ronchet/Downloads/jboss-as-7.1.1.Final/standalone/configuration/mgmt-users.properties'

Added user 'admin' to file '/Users/ronchet/Downloads/jboss-as-7.1.1.Final/domain/configuration/mgmt-users.properties'

http://localhost:9990

The screenshot shows the WildFly 10.0.0.CR2 management console interface. At the top, there is a navigation bar with tabs: Home, Deployments, Configuration, Runtime, Access Control, and Patching. The Home tab is selected. Below the navigation bar, the page title is "WildFly".

The main content area is divided into several sections:

- Deployments:** Adds and manages deployments. It includes a "Deploy an Application" section with steps 1. Use the 'Add Deployment' wizard to deploy the application and 2. Enable the deployment.
- Configuration:** Configures subsystem settings. It includes a "Create a Datasource" section with steps 1. Select the Datasources subsystem, 2. Add a Non-XA or XA datasource, and 3. Use the 'Create Datasource' wizard to configure the datasource settings.
- Runtime:** Monitors server status. It includes a "Monitor the Server" section with steps 1. Select the server and 2. View log files or JVM usage.
- Access Control:** Manages user and group permissions for management operations. It includes an "Assign User Roles" section with steps 1. Add a new user or group and 2. Assign one or more roles to that user or group.
- Patching:** Manages WildFly patches. It includes an "Apply a Patch" section with steps 1. Download the patch file to the local machine and 2. Use the 'Apply Patch' wizard to select and apply the patch.
- Need Help?** Provides links to general resources and help options. It includes sections for General Resources (WildFly Home, WildFly Documentation, Admin Guide, Model Reference Documentation, Browse Issues, Latest News), Get Help (Access tutorials and quickstarts, User Forums, IRC, Developers Mailing List), and Tools/Settings.

At the bottom of the page, there is a footer with a download link for "wildfly-10.0.0.CR2.zip" and a "Mostra tutte" button.

making WildFly accessible from remote

edit the server descriptor:

- cd into \${JBOSS_HOME}/standalone/configuration/
- edit standalone.xml

toward the end of the file, find the interfaces definitions:

```
<interfaces>
```

```
...
```

```
<interfaces/>
```

ADD A NEW INTERFACE (give it a name: in this example *myInterface*), setting the access of it to any ip.



making WildFly accessible from remote

```
<interfaces>
    <interface name="management">
        <inet-address value="${jboss.bind.address.management:127.0.0.1}">
    </inet-address></interface>
    <interface name="public">
        <inet-address value="${jboss.bind.address:127.0.0.1}">
    </inet-address></interface>
    <interface name="unsecure">
        <inet-address value="${jboss.bind.address.unsecure:127.0.0.1}">
    </inet-address></interface>
    <interface name="myInterface">
        <any-address/>
    </interface>
</interfaces>
```

making JBOSS accessible from remote

find the socket-binding-group tag, which sets the ports required for the given interfaces.

change the default-interface parameter to match the new interface (*myInterface* in ourcase)

was:

```
<socket-binding-group default-interface="default" name="standard-sockets" port-offset="${jboss.socket.binding.port-offset:0}">
```

must become:

```
<socket-binding-group default-interface="myInterface" name="standard-sockets" port-offset="${jboss.socket.binding.port-offset:0}">
```

Save, and restart the server

JBOSS: Creating users

ronchet\$./add-user.sh

What type of user do you wish to add?

- a)Management User (mgmt-users.properties)
- b)Application User (application-users.properties)

(a): **b**

Enter the details of the new user to add.

Realm (ApplicationRealm) : - *leave this blank-*

Username : **user**

Password : **pw**

Re-enter Password : **pw**

What roles do you want this user to belong to? (Please enter a comma separated list, or leave blank for none) : - *leave this blank-*

About to add user 'user' for realm 'ApplicationRealm'

Is this correct yes/no? **yes**

Added user 'user' to file '/Users/ronchet/Downloads/jboss-as-7.1.1.Final/standalone/configuration/application-users.properties'

Added user 'user' to file '/Users/ronchet/Downloads/jboss-as-7.1.1.Final/domain/configuration/application-users.properties'

Adding JNDI bindings

1) locate in your standalone/configuration/standalone.xml the line

```
<subsystem xmlns="urn:jboss:domain:naming:1.1"/>
```

2) replace it with the following section

```
<subsystem xmlns="urn:jboss:domain:naming:1.1">
  <bindings>
    <simple name="java:jboss/exported/jndi/mykey" value="MyJndiValue"/>
    <lookup name="java:jboss/exported/link/mykey" lookup="java:jboss/exported/jndi/
mykey"/>
  </bindings>
</subsystem>
```

3) restart your server

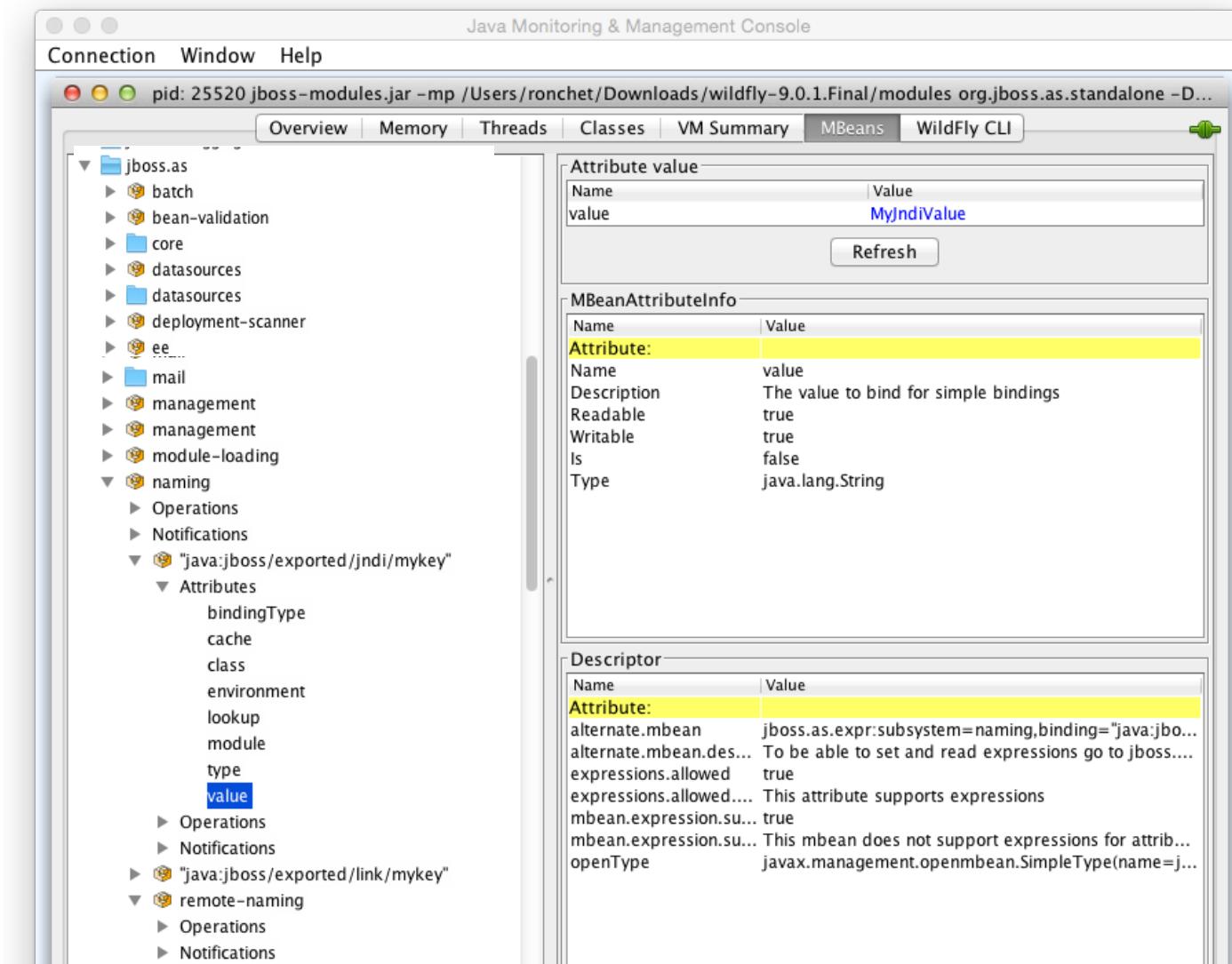
NOTE: the space visible on the client is the one following java:jboss/exported/

On server: **java:jboss/exported/jndi/mykey**

On client: **jndi/mykey**

- 1. start javaconsole (It's in the bin directory)**
- 2. connect with your wildfly running instance**
- 3. inspect the jndi bindings**

Inspecting your server



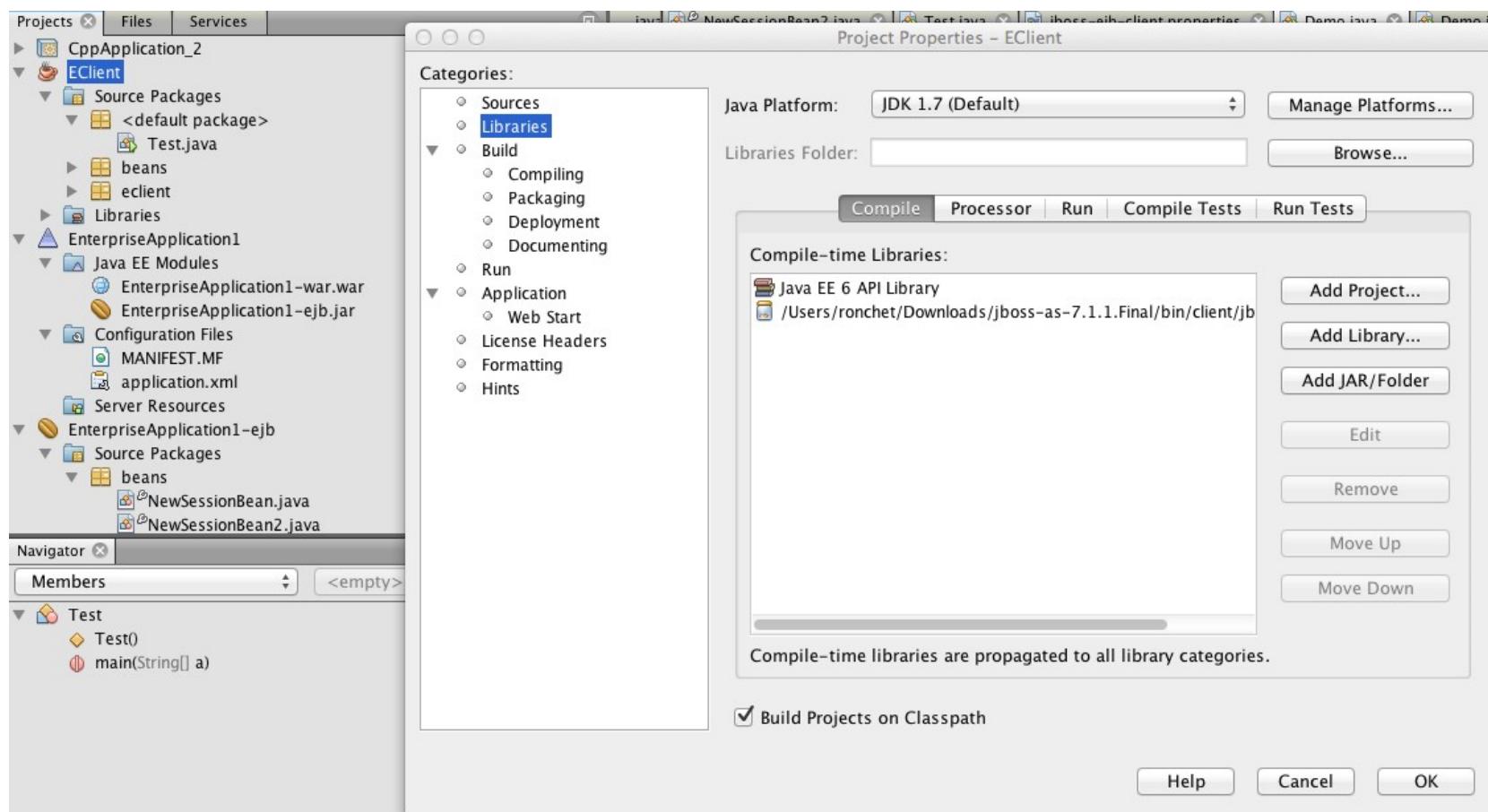
Accessing via code

```
public class JNDIaccess {  
    public static void main(String a[]) throws NamingException {  
        new JNDIaccess();  
    }  
    public JNDIaccess() throws NamingException {  
        Properties jndiProps = new Properties();  
        jndiProps.put(Context.INITIAL_CONTEXT_FACTORY,  
                      "org.jboss.naming.remote.client.InitialContextFactory");  
        jndiProps.put(Context.PROVIDER_URL, "http-remoting://127.0.0.1:8080");  
        // put your username!  
        jndiProps.put(Context.SECURITY_PRINCIPAL, "admin");  
        // put your password!  
        jndiProps.put(Context.SECURITY_CREDENTIALS, "pippo123!");  
        InitialContext initialContext = new InitialContext(jndiProps);  
        Object result = initialContext.lookup("jndi/mykey");  
        System.out.println(result);  
    }  
}
```

see <https://docs.jboss.org/author/display/WFLY8/Remote+EJB+invocations+via+JNDI+-+EJB+client+API+or+remote-naming+project>

Add the libraries!

They're in JBOSS_HOME/bin/client/jboss-client.jar



run:

ott 08, 2015 10:58:08 PM org.xnio.Xnio <clinit>

INFO: XNIO version 3.3.1.Final

OUTPUT

ott 08, 2015 10:58:08 PM org.xnio.nio.NioXnio <clinit>

INFO: XNIO NIO Implementation Version 3.3.1.Final

ott 08, 2015 10:58:08 PM org.jboss.remoting3.EndpointImpl <clinit>

INFO: JBoss Remoting version 4.0.9.Final

MyJndiValue

BUILD SUCCESSFUL (total time: 0 seconds)

Warning

JNDI access to these data on Wildfly is
READ ONLY!

Warning

- JNDI access from code is READ ONLY!, but you can write from CLI
- **\$./jboss-cli.sh connect**

```
[standalone@localhost:9999 /] /subsystem=naming/binding=java\:jboss\exported\demoParam:add(value="Demo configuration value",binding-type=simple)
```

```
{ "outcome" => "success"}  
[standalone@localhost:9999 /] quit
```

All on one line!

Then reconnect with jconsole to see the result

Useful references

- <http://docs.wildfly.org/14/Getting Started Guide.html>
- <http://docs.oracle.com/javase/jndi/tutorial/trailmap.html>
- <https://docs.jboss.org/author/display/AS71/CLI+Recipes>