

Quick Installation Guide for Layman: DSpace 1.8 on Linux (CentOS 6)

Yatrik Patel
INFLIBNET Centre
yatrik@inflibnet.ac.in

Preparing your System for DSpace(Pre-requisites).

As per standard installation guide you will need Java(JDK), Apache Tomcat, Apache Maven, Apache Ant, Postgresql and DSpace source.

You can download the same from following URLS (Look for appropriate versions 32/64 bits as per your OS)

Java:<http://www.oracle.com/technetwork/java/javase/downloads/index.html> (Choose JAVA SE 6 Update XX as JAVA SE 7 is not recommended by DSpace)

Apache Tomcat : <http://tomcat.apache.org/download-70.cgi>

Apache Maven : <http://maven.apache.org/download.html>

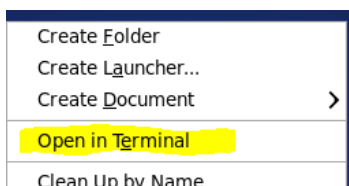
Apache Ant : <http://ant.apache.org/bindownload.cgi>

Postgresql : <http://www.enterprisedb.com/products-services-training/pgdownload>

DSpace: <http://sourceforge.net/projects/dspace/files/DSpace%20Stable/1.8.1/dspace-1.8.1-src-release.zip/download>

This files are available and kept in folder /opt/source

To reach to that folder “Right Click” on your Linux desktop and Choose “Open in Terminal”



In terminal window go to /opt/source directory

```
[root@localhost Desktop]# cd /opt/source
```

To see what all sources we have perform list command

```
[root@localhost source]# ls -al
total 152340
drwxr-xr-x. 2 root root    4096 Feb 10 17:15 .
drwxr-xr-x. 8 root root    4096 Feb 10 17:00 ..
-rw-----. 1 root root  8324124 Feb  9 15:54 apache-ant-1.8.2-bin.tar.gz
-rw-r--r--. 1 root root  4873043 Feb  9 15:21 apache-maven-3.0.4-bin.tar.gz
-rw-r--r--. 1 root root  7527434 Feb  9 15:24 apache-tomcat-7.0.25.tar.gz
-rw-r--r--. 1 root root   7379506 Feb  9 17:03 dspace-1.8.1-src-release.zip
-rw-----. 1 root root  81053867 Feb  9 17:53 jdk-6u30-linux-x64-rpm.bin
-rw-r--r--. 1 root root  46808180 Feb 10 17:15 postgresql-9.0.6-1-linux-x64.run
```

It is recommended to keep all source file intact, while installing we will copy each file to /opt directory and perform further steps.

First of all prepare your system for Java

Copy jdk installer file to /opt

```
[root@localhost source]# cd /opt
[root@localhost opt]# cp source/jdk-6u30-linux-x64-rpm.bin .
```

Now we need to give "execute" permission to the source file (by default it is having only read permission)

```
[root@localhost opt]# chmod a+x jdk-6u30-linux-x64-rpm.bin
```

run the installer,

```
[root@localhost opt]# ./jdk-6u30-linux-x64-rpm.bin
Unpacking...
Checksumming...
Extracting...
UnZipSFX 5.50 of 17 February 2002, by Info-ZIP (Zip-Bugs@lists.wku.edu).
  inflating: jdk-6u30-linux-amd64.rpm
  inflating: sun-javadb-common-10.6.2-1.1.i386.rpm
  inflating: sun-javadb-core-10.6.2-1.1.i386.rpm
  inflating: sun-javadb-client-10.6.2-1.1.i386.rpm
  inflating: sun-javadb-demo-10.6.2-1.1.i386.rpm
  inflating: sun-javadb-docs-10.6.2-1.1.i386.rpm
  inflating: sun-javadb-javadoc-10.6.2-1.1.i386.rpm
Preparing...
 1:jdk          ##### [100%]
```

and follow the instructions

```
http://java.sun.com/javase/registration/JDKRegistrationPrivacy.html
Press Enter to continue.....
```

Now Check, Which Java you are running by

```
[root@localhost opt]# java -version
java version "1.6.0_20"
OpenJDK Runtime Environment (IcedTea6 1.9.7) (rhel-1.39.1.9.7.el6-x86_64)
OpenJDK 64-Bit Server VM (build 19.0-b09, mixed mode)
```

Is it the Java which you installed ? Naaah..!, We have not installed OpenJDK, we attempted JAVA SE. **Where is JDK you have installed?**

It will be installed as default at

```
/usr/java/jdk1.6.0_30
```

You may need to **change Java configuration file and explicitly tell your system**, from where java should be executed, to do this install your flavor of java as alternatives.

```
[root@localhost jdk1.6.0_30]# alternatives --install /usr/bin/java java /usr/java/jdk1.6.0_30/bin/java 2
```

And, tell your system, from where it should execute java..

```
[root@localhost jdk1.6.0_30]# alternatives --config java
There are 2 programs which provide 'java'.

  Selection    Command
-----
*+ 1           /usr/lib/jvm/jre-1.6.0-openjdk.x86_64/bin/java
   2           /usr/java/jdk1.6.0_30/bin/java

Enter to keep the current selection[+], or type selection number: 2
```

Again, verify the version.

```
[root@localhost jdk1.6.0_30]# java -version
java version "1.6.0_30"
Java(TM) SE Runtime Environment (build 1.6.0_30-b12)
Java HotSpot(TM) 64-Bit Server VM (build 20.5-b03, mixed mode)
```

We have the java, the way we wanted..

Now its time to install Apache siblings i.e. Apache Maven, Apache Ant and Apache Tomcat.

To **Install Maven**, proceed with following step.

Copy to /opt

```
[root@localhost opt]# cd /opt
[root@localhost opt]# cp source/apache-maven-3.0.4-bin.tar.gz .
[root@localhost opt]#
```

Untar and inflate

```
[root@localhost opt]# tar -xvf apache-maven-3.0.4-bin.tar.gz
```

You will have maven installed in /opt/apache-maven-3.0.4

To **install Apache Ant**, Same as above.

Copy to /opt and inflate by following command sequence

```
[root@localhost opt]# cp source/apache-ant-1.8.2-bin.tar.gz .
[root@localhost opt]# tar -xvf apache-ant-1.8.2-bin.tar.gz
```

It will get installed in /opt/apache-ant-1.8.2

Same process will be applied for **Installing Apache Tomcat**.

```
[root@localhost opt]# cp source/apache-tomcat-7.0.25.tar.gz .  
[root@localhost opt]# tar -xvf apache-tomcat-7.0.25.tar.gz
```

It will get installed in /opt/apache-tomcat-7.0.25

Now it is time to tell your linux, that where it can find Maven, Ant and Tomcat. To do this, you will need to set environment of your linux system by putting **appropriate environment variables** (and PATHs).

Generally when a user logs in, environment variables are set from various places. That includes /etc/profile (for all users). /etc/profile.d/ is a good place to put your application specific setups. To do this go to

```
[root@localhost opt]# cd /etc/profile.d
```

Create a new file called java.sh (or any other file with “.sh” extension)

```
[root@localhost profile.d]# vim java.sh
```

and place the following lines in that file (double check the paths you are entering)

```
#!/bin/bash  
JAVA_HOME=/usr/java/jdk1.6.0_30  
ANT_HOME=/opt/apache-ant-1.8.2  
CATALINA_HOME=/opt/apache-tomcat-7.0.25  
MAVEN_HOME=/opt/apache-maven-3.0.4  
  
PATH=$JAVA_HOME/bin:$ANT_HOME/bin:$MAVEN_HOME/bin:$PATH  
export PATH JAVA_HOME ANT_HOME MAVEN_HOME  
export CLASSPATH=.
```

exit the file with <esc>: wq! (Write and Quit)

```
:wq!
```

Give execute the permission to the file you have created

```
[root@localhost profile.d]# chmod +x java.sh
```

Give the effect to the OS by executing following command

```
[root@localhost profile.d]# source java.sh
```

Doing above process (i.e. putting shell script in /etc/profile.d) will also ensure that every time system boots up/user logs in, environment variable will be set.

Now, its time to check what we did.

```
[root@localhost profile.d]# echo $JAVA_HOME
/usr/java/jdk1.6.0_30
[root@localhost profile.d]# echo $CATALINA_HOME
/opt/apache-tomcat-7.0.25
[root@localhost profile.d]# ant -version
Apache Ant(TM) version 1.8.2 compiled on December 20 2010
[root@localhost profile.d]# mvn -version
Apache Maven 3.0.4 (r1232337; 2012-01-17 14:14:56+0530)
Maven home: /opt/apache-maven-3.0.4
Java version: 1.6.0_30, vendor: Sun Microsystems Inc.
Java home: /usr/java/jdk1.6.0_30/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "2.6.32-131.0.15.el6.x86_64", arch: "amd64", family: "unix"
```

Output of command sequence will , first it will **check for JAVA_HOME**(which is necessary to run tomcat and some java applications), Then **CATALINA_HOME** will tell the system where your tomcat is, Next is to verify **Ant** version followed by **Maven** version, its installation path and which java it takes.

You can start default Tomcat server by executing following command sequence

```
[root@localhost profile.d]# cd $CATALINA_HOME/bin
[root@localhost bin]# ./startup.sh
Using CATALINA_BASE:   /opt/apache-tomcat-7.0.25
Using CATALINA_HOME:   /opt/apache-tomcat-7.0.25
Using CATALINA_TMPDIR: /opt/apache-tomcat-7.0.25/temp
Using JRE_HOME:        /usr/java/jdk1.6.0_30
Using CLASSPATH:       /opt/apache-tomcat-7.0.25/bin/bootstrap.jar:/opt/apache-tomcat-7.0.25/bin/tomcat-juli.jar
[root@localhost bin]#
```

You can check whether your tomcat is running or not by putting <http://localhost:8080/> in address bar of your browser



To shutdown your tomcat instance issue following command

```
[root@localhost bin]# cd $CATALINA_HOME/bin
[root@localhost bin]# ./shutdown.sh
Using CATALINA_BASE:   /opt/apache-tomcat-7.0.25
Using CATALINA_HOME:   /opt/apache-tomcat-7.0.25
Using CATALINA_TMPDIR: /opt/apache-tomcat-7.0.25/temp
Using JRE_HOME:        /usr/java/jdk1.6.0_30
Using CLASSPATH:       /opt/apache-tomcat-7.0.25/bin/bootstrap.jar:/opt/apache-tomcat-7.0.25/bin/tomcat-juli.jar
```

We will attempt to install Postgresql (if your operating system doesn't contain it by default) Copy installer file to /opt

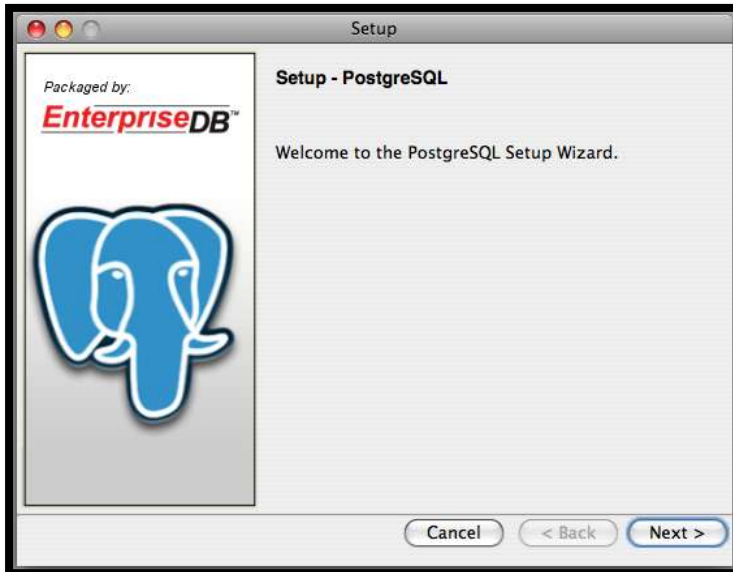
```
[root@localhost opt]# cp source/postgresql-9.0.6-1-linux-x64.run .
```

Make it executable and execute. !

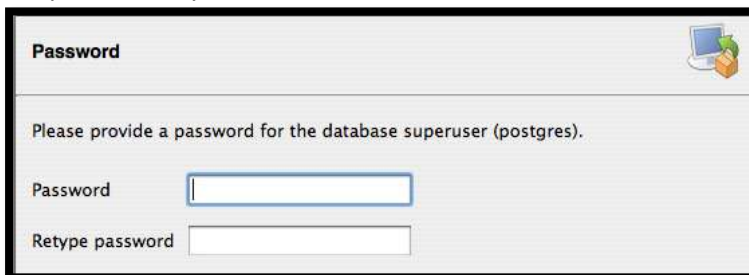
```
[root@localhost opt]# chmod +x postgresql-9.0.6-1-linux-x64.run
[root@localhost opt]# ./postgresql-9.0.6-1-linux-x64.run
```

5

You will get a GUI based installer

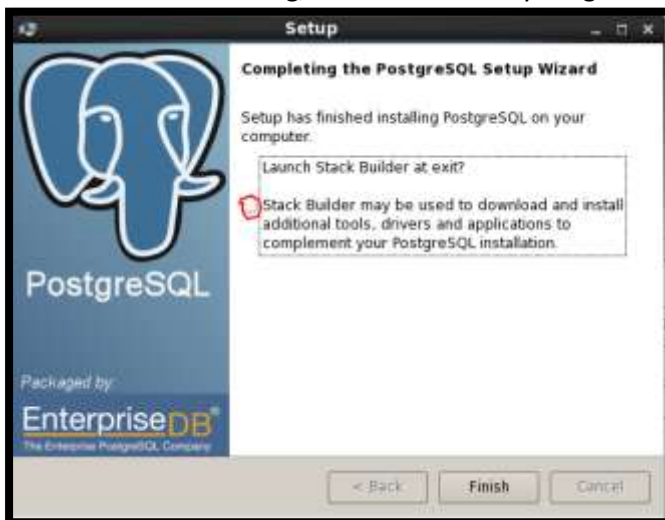


Please follow the interactive installation instructions for installation and data path (**Its better to note it down somewhere so we can find it later**), while proceeding you will be prompted to enter password for postgres account(Don't forget to remember !!), this will be Superuser for your database.



Here, an account with username 'postgres' will be created in your OS with the password supplied by you. Proceed further and enter port and other information asked by installer (better to keep as it is !)

Once installation is completed you will be prompted with following screen. Uncheck the "Stack Builder...." Thing, we don't need anything additional at this point of time.



[Another Option to install Postgresql would be (This requires internet connection), Postgres installation files will be downloaded from online repository and will get installed to your OS. To do this #yum install postgres*]

Now we will tune postgres to suite our requirement for DSpace.

To do this first we need to edit host configuration file to allow access to dspace user to dspace database without hassle on localhost , perform the following command sequence

```
[root@localhost opt]# cd /opt/PostgreSQL/9.0/data/  
[root@localhost data]# vim pg_hba.conf
```

Add the line "host dspace dspace 127.0.0.1/32 md5 " as shown below,

```
# IPv4 local connections:  
host dspace dspace 127.0.0.1/32 md5  
host all all 127.0.0.1/32 md5
```

Please restart postgres service

```
[root@localhost data]# service postgresql-9.0 restart  
Restarting PostgreSQL 9.0:  
waiting for server to shut down.... done  
server stopped  
waiting for server to start.... done  
server started  
PostgreSQL 9.0 restarted successfully
```

It would be also appropriate to add postgres to your PATH environment variable so you can execute various postgres commands, to do this , recall the file we have created earlier "java.sh" , open the same in editor

```
[root@localhost data]# vim /etc/profile.d/java.sh
```

and edit PATH as following

```
PATH=/opt/PostgreSQL/9.0/bin:$JAVA_HOME/bin:$ANT_HOME/bin:$MAVEN_HOME/bin:$PATH
```

save java.sh and give effect to your system environment by

```
[root@localhost data]# source /etc/profile.d/java.sh
```

DSpace Installation:

The first thing we need to do at this stage is is to create an operating system user named "dspace"

```
[root@localhost Desktop]# useradd -m dspace
```

Now we will attempt to create "dspace" user for database and a database named "dspace" owned by "dspace" user,

To create user

```
[root@localhost data]# createuser -U postgres -d -A -P dspace  
Enter password for new role: Please put a dspace password  
Enter it again: Please put a dspace password  
Shall the new role be allowed to create more new roles? (y/n) y  
Password: Here you have to enter your postgres password
```

To create database

```
[root@localhost data]# createdb -U dspace -E UNICODE dspace
Password: Enter dspace password
```

Now we will attempt **actual installation**.

Copy your dspace file to your root (/) folder (you may choose any other folder too..)

```
[root@localhost data]# cd /
[root@localhost /]# cp /opt/source/dspace-1.8.1-src-release.zip .
```

As the file is in zipped format, inflate the same .

```
[root@localhost /]# unzip dspace-1.8.1-src-release.zip
```

We will also required to create **one more directory at root** , named **“dspace”** , which will **contain actual DSpace** application files.(or your dspace installation).

```
[root@localhost /]# mkdir dspace
```

Now we need to **change the ownership of both the directories** as they will get operated by **dspace user**. The present permissions are with root.

```
[root@localhost /]# chown dspace.dspace /dspace -R
[root@localhost /]# chown dspace.dspace /dspace-1.8.1-src-release -R
```

Now dspace user will have full permission for above mentioned directories and its sub directories you can verify the same by using “ls” command.

```
drwx-----.  3 root  root    4096 Feb  8 22:41 .dbus
drwxr-xr-x.  20 root  root    3860 Feb 10 16:49 dev
drwxr-xr-x.   2 dspace dspace  4096 Feb 10 17:52 dspace
drwxr-xr-x.  13 dspace dspace  4096 Dec 15 10:25 dspace-1.8.1-src-release
```

Now **take control as dspace user and go to config folder** of downloaded dspace source.

```
[root@localhost /]# su dspace
[dspace@localhost /]$ cd /dspace-1.8.1-src-release/dspace/config/
```

Edit dspace.cfg to tell dspace, about configuration settings.

```
[dspace@localhost config]$ vim dspace.cfg
```

Change at least following things

```
dspace.url = ${dspace.baseUrl}/jspui
dspace.name = DSpace at INFLIBNET Workshop
db.username = dspace
db.password = dspace (or whatever you have chosen)
mail.server = <will tell you in lab>
mail.from.address : your@email.com
feedback.recipient: your@email.com
mail.admin: your@email.com
```

and save your file.

Its time for compilation now, to do this go to dspace folder.

```
[dspace@localhost config]$ cd /dspace-1.8.1-src-release/dspace
```

begin the compilation by (Please ensure that your system is connected to internet otherwise, things will not work as you want)

```
[dspace@localhost dspace]$ mvn package
```

It should begin the downloading

```
dspace@localhost dspace]$ mvn package
[INFO] Scanning for projects...
Downloading: http://repo.maven.apache.org/maven2/org/dspace/dspace-pom/12/dspace-pom-12.pom
Downloaded: http://repo.maven.apache.org/maven2/org/dspace/dspace-pom/12/dspace-pom-12.pom (11 KB at 5.3 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/sonatype/oss/oss-parent/7/oss-parent-7.pom
Downloaded: http://repo.maven.apache.org/maven2/org/sonatype/oss/oss-parent/7/oss-parent-7.pom (5 KB at 6.8 KB/sec)
[INFO] -----
[INFO] Reactor Build Order:
[INFO]
[INFO] DSpace Addon Modules
[INFO] DSpace XML-UI (Manakin) :: Web Application
[INFO] DSpace LNI :: Web Application
[INFO] DSpace OAI :: Web Application
[INFO] DSpace JSP-UI :: Web Application
[INFO] DSpace SWORD :: Web Application
[INFO] DSpace SWORDv2 :: Web Application
[INFO] DSpace SOLR :: Web Application
[INFO] DSpace Assembly and Configuration
```

Wait for confirmation....!

```
[INFO] Copying files to /dspace-1.8.1-src-release/dspace/target/dspace-1.8.1-build
[INFO] -----
[INFO] Reactor Summary:
[INFO]
[INFO] DSpace Addon Modules ..... SUCCESS [58.732s]
[INFO] DSpace XML-UI (Manakin) :: Web Application ..... SUCCESS [6:48.896s]
[INFO] DSpace LNI :: Web Application ..... SUCCESS [23.489s]
[INFO] DSpace OAI :: Web Application ..... SUCCESS [10.637s]
[INFO] DSpace JSP-UI :: Web Application ..... SUCCESS [24.420s]
[INFO] DSpace SWORD :: Web Application ..... SUCCESS [4.149s]
[INFO] DSpace SWORDv2 :: Web Application ..... SUCCESS [56.286s]
[INFO] DSpace SOLR :: Web Application ..... SUCCESS [1:01.850s]
[INFO] DSpace Assembly and Configuration ..... SUCCESS [1:17.409s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 12:09.512s
[INFO] Finished at: Fri Feb 10 18:28:37 IST 2012
[INFO] Final Memory: 17M/66M
[INFO] -----
[dspace@localhost dspace]$
```

Now, go to build directory,

```
[dspace@localhost dspace]$ cd /dspace-1.8.1-src-release/dspace/target/dspace-1.8.1-build/
```

issue "ant fresh_install" command.

```
[dspace@localhost dspace-1.8.1-build]$ ant fresh_install
```

wait for most "awaited" message...!

```
BUILD SUCCESSFUL
Total time: 2 minutes 39 seconds
[dspace@localhost dspace-1.8.1-build]$
```

We need to proceed for **administrator account for dspace**.

```
[dspace@localhost dspace-1.8.1-build]$ /dspace/bin/dspace create-administrator
Creating an initial administrator account
E-mail address: yatrik@inflibnet.ac.in
First name: Yatrik
Last name: Patel
WARNING: Password will appear on-screen.
Password: whyishouldtell
Again to confirm: whyishouldtell
Is the above data correct? (y or n): y
Administrator account created
[dspace@localhost dspace-1.8.1-build]$
```

Now its **time to tell tomcat where your dspace is**, there are several approaches to do this, we will create use "context path", **get back as root, go to your tomcat's** installations config folder and create a new file jspui.xml as following (**Ensure that your tomcat is stopped! You know how to stop...**)

```
[dspace@localhost dspace-1.8.1-build]$ exit
exit
[root@localhost /]# cd /opt/apache-tomcat-7.0.25/conf/Catalina/localhost/
[root@localhost localhost]# vim jspui.xml
```

add the lines as shown below.. and save

```
<Context path="/jspui"
  docBase="/dspace/webapps/jspui"
  debug="0"
  reloadable="true"
  cachingAllowed="false"
  crosscontext="true" />
```

[You can use same approach for other interfaces like xmlui,oai etc.]

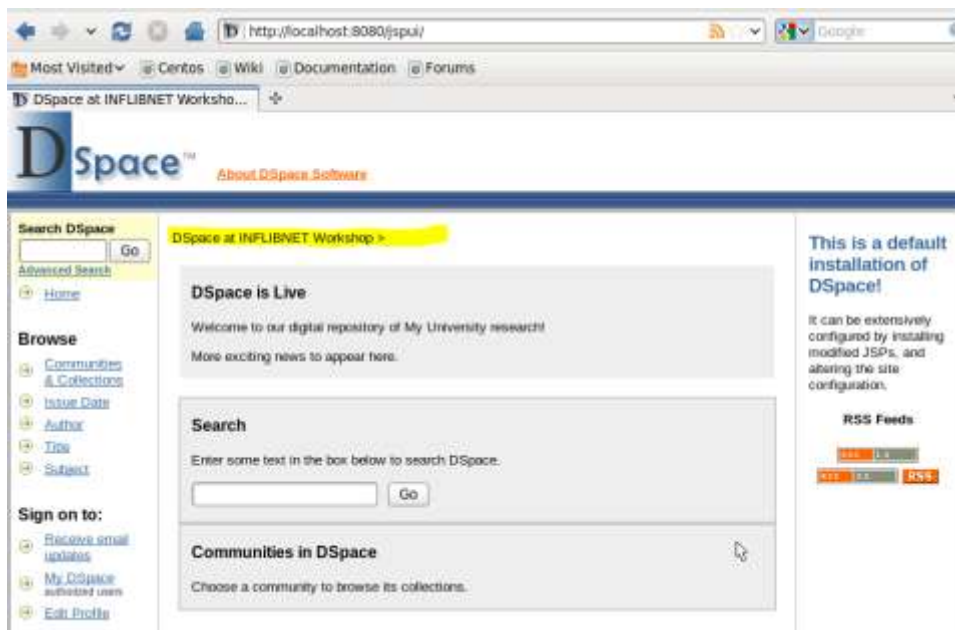
Now it's time to **change permission of tomcat owner**, so the **tomcat can be executed by dspace user**

```
[root@localhost localhost]# cd /opt
[root@localhost opt]# chown dspace.dspace apache-tomcat-7.0.25/ -R
```

Again **take control as dspace, and start tomcat**

```
[root@localhost opt]# su dspace
[dspace@localhost opt]$ cd $CATALINA_HOME/bin
[dspace@localhost bin]$ ./startup.sh
```

Finally The moment of truth



As per theory of “karma” every action produces an equal and opposite reaction. Every time we think or do something, we create a cause, which in time will bear its corresponding effects. So, if you don’t get desired results or effect, re-think about your “karma” of installing dspace, try to correct your “karma” and you will get the result.