

Quick Installation Guide for Layman: DSpace 6.x on Linux (CentOS 7)

January, 2019

Yatrik Patel

INFLIBNET Centre
yatrik@inflibnet.ac.in

Swapnil Patel

INFLIBNET Centre
swapnil@inflibnet.ac.in

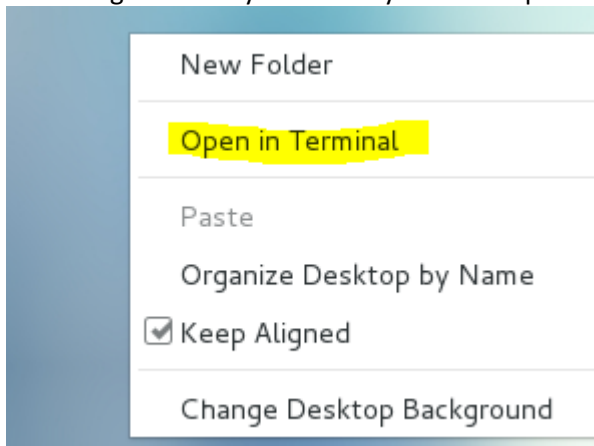
Disclaimer: This guide has been adapted from documents available at <https://wiki.duraspace.org/display/DSDOC6x/> with an intension to provide set of instructions to participants of DSpace workshop at INFLIBNET Centre, as well as novice users, any feedback to author will be appreciated.

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 (JDK)	http://www.oracle.com/technetwork/java/javase/downloads/index.html (Choose JAVA SE 8uXXX, JDK)
Apache Tomcat	http://tomcat.apache.org/download-90.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	https://github.com/DSpace/DSpace/releases/tag/dspace-6.2

This files are available and kept in folder /opt/source, To reach to that folder Do the right click any where on your 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, perform list command

```
[root@localhost source]# ls -al
total 327848
drwxr-xr-x. 2 root root      218 Sep 12  2018 .
drwxr-xr-x. 8 root root     4096 Sep 12  2018 ..
-rw-r--r--. 1 root root  5818302 Sep 12  2018 apache-ant-1.9.11-bin.tar.gz
-rw-r--r--. 1 root root  8799579 Sep 12  2018 apache-maven-3.5.3-bin.tar.gz
-rw-r--r--. 1 root root  9417189 Sep 12  2018 apache-tomcat-8.0.50.tar.gz
-rw-r--r--. 1 root root   8421275 Sep 12  2018 dspace-6.2-src-release.zip
-rw-r--r--. 1 root root 160182448 Sep 12  2018 jdk-8u77-linux-x64.rpm
-rw-r--r--. 1 root root 143059307 Sep 12  2018 postgresql-9.6.8-1-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-8u25-linux-x64.rpm .
```

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-8u25-linux-x64.rpm
```

run the installer,

```
[root@localhost opt]# rpm -ivh jdk-8u25-linux-x64.rpm
Preparing...                               ##### [100%]
Updating / installing...
 1:jdk1.8.0_25-2000:1.8.0_25-fcs            ##### [100%]
Unpacking JAR files...
   rt.jar...
   jsse.jar...
   charsets.jar...
   tools.jar...
   localedata.jar...
   jfxrt.jar...
[root@localhost opt]#
```

and follow the instructions

Now Check, Which Java you are running by

```
[root@centos ~]# java -version
java version "1.7.0_51"
OpenJDK Runtime Environment (rhel-2.4.4.1.el6_5-x86_64 u51-b02)
OpenJDK 64-Bit Server VM (build 24.45-b08, mixed mode)
[root@centos ~]# █
```

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/jdk1.8.0_25/ █
```

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 jdk1.8.0_25]# alternatives --install /usr/bin/java java /usr/java/jdk1.8.0_25/bin/java 2 █
```

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

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

 Selection    Command
-----
*+ 1          /usr/lib/jvm/java-1.7.0-openjdk-1.7.0.51-2.4.5.5.el7.x86_64/jre/bin/java
   2          /usr/java/jdk1.8.0_25/bin/java

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

Again, verify the version.

```
[root@localhost opt]# java -version
java version "1.8.0_25"
Java(TM) SE Runtime Environment (build 1.8.0_25-b17)
Java HotSpot(TM) 64-Bit Server VM (build 25.25-b02, mixed mode)
[root@localhost opt]# █
```

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-3.2.5-bin.tar.gz .
```

3

Untar and inflate

```
[root@localhost opt]# tar -xvf apache-maven-3.2.5-bin.tar.gz
```

You will have maven installed in /opt/apache-maven-3.2.5

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

Copy to /opt and inflate by following command sequence

```
[root@localhost opt]# cp source/apache-ant-1.9.4-bin.tar.gz .  
[root@localhost opt]# tar -xvf apache-ant-1.9.4-bin.tar.gz
```

It will get installed in /opt/apache-ant-1.9.4

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

```
[root@localhost opt]# cp source/apache-tomcat-8.0.15.tar.gz .  
[root@localhost opt]# tar -xvf apache-tomcat-8.0.15.tar.gz
```

It will get installed in /opt/apache-tomcat-8.0.15

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/jdk1.8.0_77  
ANT_HOME=/opt/apache-ant-1.9.11  
CATALINA_HOME=/opt/apache-tomcat-8.0.50  
MAVEN_HOME=/opt/apache-maven-3.5.3  
  
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 which 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/jdk1.8.0_25
[root@localhost profile.d]# echo $CATALINA_HOME
/opt/apache-tomcat-8.0.15
[root@localhost profile.d]# ant -version
Apache Ant(TM) version 1.9.4 compiled on April 29 2014
[root@localhost profile.d]# mvn -version
Apache Maven 3.2.5 (12a6b3acb947671f09b81f49094c53f426d8cea1; 2014-12-14T22:59:23+05:30)
Maven home: /opt/apache-maven-3.2.5
Java version: 1.8.0_25, vendor: Oracle Corporation
Java home: /usr/java/jdk1.8.0_25/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "linux", version: "3.10.0-123.el7.x86_64", arch: "amd64", family: "unix"
```

Output of command sequence will first **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-8.0.15
Using CATALINA_HOME:   /opt/apache-tomcat-8.0.15
Using CATALINA_TMPDIR: /opt/apache-tomcat-8.0.15/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_25
Using CLASSPATH:       /opt/apache-tomcat-8.0.15/bin/bootstrap.jar:/opt/apache-tomcat-8.0.15/bin/tomcat-juli.jar
Tomcat started.
[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 shut down your tomcat instance issue following command

```
[root@localhost bin]# ./shutdown.sh
Using CATALINA_BASE:   /opt/apache-tomcat-8.0.15
Using CATALINA_HOME:   /opt/apache-tomcat-8.0.15
Using CATALINA_TMPDIR: /opt/apache-tomcat-8.0.15/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_25
Using CLASSPATH:       /opt/apache-tomcat-8.0.15/bin/bootstrap.jar:/opt/apache-tomcat-8.0.15/bin/tomcat-juli.jar
```

5

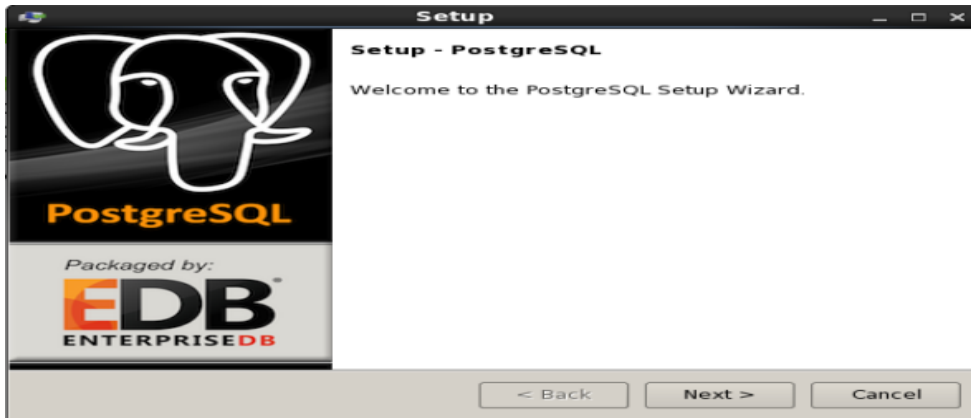
We will attempt to install PostgreSQL (if your operating system doesn't contain it by default)

Copy installer file to /opt

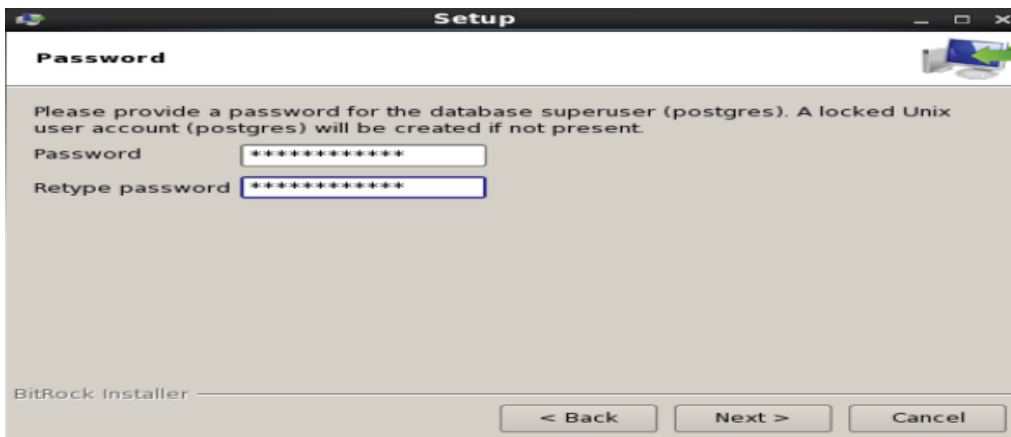
Make it executable and execute. !

```
[root@localhost ~]# cd /opt/
[root@localhost opt]# cp source/postgresql-9.4.0-1-linux-x64.run .
[root@localhost opt]# chmod +x postgresql-9.4.0-1-linux-x64.run
[root@localhost opt]# ./postgresql-9.4.0-1-linux-x64.run
```

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.

6



[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/9.4/data/
[root@localhost data]# vim pg_hba.conf
```

Add the line "host<tab> dspace<tab> dspace<tab>127.0.0.1/32<tab> 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
# IPv6 local connections:
```

Please **restart postgres** service

```
[root@localhost data]# service postgresql-9.4 restart
Redirecting to /bin/systemctl restart postgresql-9.4.service
[root@localhost data]# systemctl restart postgresql-9.4.service
[root@localhost data]# █
```

OR

```
[root@centos data]# service postgresql-9.3 restart
Restarting PostgreSQL 9.3:
waiting for server to shut down.... done
server stopped
waiting for server to start.... done
server started
PostgreSQL 9.3 restarted successfully
[root@centos data]# █
```

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/9.4/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 to **create an operating system user** named “dspace”

```
[root@localhost data]# 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: enter password for dspace database
Enter it again: re-enter password for dspace database
Password: enter password for postgres user
[root@localhost data]# █
```

To create database

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

Go to PG Admin and create extension in ‘dspace’ database as mentioned below



The screenshot shows a window titled 'Query Editor' with a 'Query History' tab. The main area contains a single line of SQL code: `1 create extension pgcrypto;`

Now we will attempt **actual installation**.

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

```
[root@localhost /]# cd /
[root@localhost /]# cp /opt/source/dspace-6.2-src-release.zip .
```

∞

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

```
[root@localhost ~]# unzip dspace-6.2-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-6.2-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 -al” command.

```
drwxr-xr-x. 16 dspace dspace    201 Jun 14 11:06 dspace  
drwxr-xr-x. 15 dspace dspace   4096 Sep  7  2017 dspace-6.2-src-release
```

Now **take control as dspace user and go to source folder (/dspace-6.x-src-release)**

```
[root@localhost ~]# su dspace  
[dspace@localhost ~]$ cd /dspace-6.2-src-release/
```

Edit dspace.cfg in config directory to tell dspace, about configuration settings.

```
[dspace@localhost ~]$ cd /dspace-6.2-src-release/dspace/config/  
[dspace@localhost config]$ vim dspace.cfg
```

Change at least following things

dspace.name = DSpace at INFLIBNET Workshop

db.username = dspace

db.password = dspace (or whatever you have chosen)

and save your file.

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

```
[dspace@localhost config]$ cd /dspace-6.2-src-release/dspace  
[dspace@localhost dspace]$ mvn package
```

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
[INFO] Scanning for projects...
[INFO] -----
[INFO] Reactor Build Order:
[INFO]
[INFO] DSpace Addon Modules
[INFO] DSpace Kernel :: Additions and Local Customizations
[INFO] DSpace XML-UI (Manakin) :: Local Customizations
[INFO] DSpace LNI :: Local Customizations
[INFO] DSpace JSP-UI :: Local Customizations
[INFO] DSpace REST (JAX RS 1 - Jersey) :: Local Customizations
[INFO] DSpace SWORD :: Local Customizations
[INFO] DSpace SWORD v2 :: Local Customizations
[INFO] DSpace SOLR :: Local Customizations
```

It should begin the downloading,

Wait for confirmation....!

```
[INFO] DSpace SWORD :: Local Customizations ..... SUCCESS [ 11.443 s]
[INFO] DSpace SWORD v2 :: Local Customizations ..... SUCCESS [46:37 min]
[INFO] DSpace SOLR :: Local Customizations ..... SUCCESS [10:28 min]
[INFO] DSpace OAI 2.0 :: Local Customizations ..... SUCCESS [01:37 min]
[INFO] DSpace Assembly and Configuration ..... SUCCESS [02:12 min]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:19 h
[INFO] Finished at: 2015-01-16T17:37:12+05:30
[INFO] Final Memory: 29M/107M
[INFO] -----
[dSPACE@localhost dSPACE]$ cd dSPACE-4.2-src-release/
```

Now, go to installer directory,

```
[dSPACE@localhost /]$ cd /dSPACE-6.2-src-release/dSPACE/target/dSPACE-installer/
```

issue "ant fresh_install" command.

```
[dSPACE@localhost dSPACE-installer]$ ant fresh_install
```

wait for most "awaited" message...!

```
BUILD SUCCESSFUL
Total time: 1 minute 8 seconds
```

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

```
[dSPACE@localhost /]$ cd /dSPACE/bin/
[dSPACE@localhost bin]$ ./dSPACE create-administrator
Creating an initial administrator account
E-mail address: yatrik@inflibnet.ac.in
First name: Yatrik
Last name: Patel
Password will not display on screen.
Password:
Again to confirm:
Is the above data correct? (y or n): y
Administrator account created
```

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 bin]$ vim /opt/apache-tomcat-8.0.15/conf/Catalina/localhost/jspui.xml
```

add the lines as shown below, and save

```
<?xml version='1.0'?>
<Context
  docBase="/dspace/webapps/jspui"
  debug="0"
  reloadable="true"
  cachingAllowed="false"
  allowLinking="true"/>
```

Do the same thing for “solr”

```
[dspace@localhost bin]$ vim /opt/apache-tomcat-8.0.15/conf/Catalina/localhost/solr.xml
```

```
<?xml version='1.0'?>
<Context
  docBase="/dspace/webapps/solr"
  debug="0"
  reloadable="true"
  cachingAllowed="false"
  allowLinking="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**

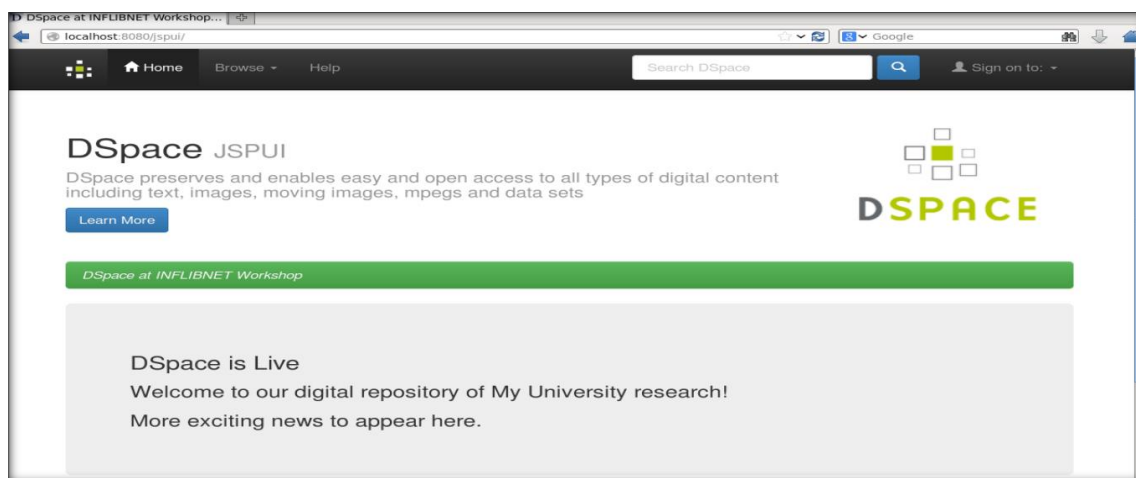
```
[root@localhost opt]# chown dspace.dspace apache-tomcat-8.0.15/ -R
```

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

```
[dspace@localhost opt]$ cd $CATALINA_HOME/bin
[dspace@localhost bin]$ ./startup.sh
Using CATALINA_BASE:   /opt/apache-tomcat-8.0.15
Using CATALINA_HOME:   /opt/apache-tomcat-8.0.15
Using CATALINA_TMPDIR: /opt/apache-tomcat-8.0.15/temp
Using JRE_HOME:        /usr/java/jdk1.8.0_25
Using CLASSPATH:       /opt/apache-tomcat-8.0.15/bin/bootstrap.jar:/opt/apache-tomcat-8.0.15/bin/
tomcat-juli.jar
Tomcat started.
[dspace@localhost bin]$
```

11

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.