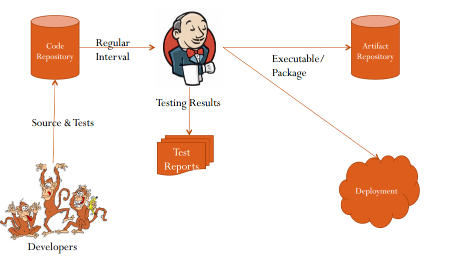
**What is Jenkins?**

Jenkins is a continuous integration and continuous delivery application cross-platform that increases your productivity

**Jenkins use cases**

Use Jenkins to **build and test your software projects continuously** making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to **continuously deliver** your software by providing powerful ways to define your build pipelines and integrating with a large number of testing and deployment technologies

**This is how Jenkins works**



**Installing Jenkins on Unix/Linux**

**System requirement**

Jenkins requires Java7 or above to function. Java8 is recommended. Jenkins requires a fair amount of memory to operate well. Smaller installations should start around 256MB-1GB.

**Unix/Linux Installation**

Jenkins has native integrations with the following OSes:

* [Jenkins/Hudson as Solaris 10 service](http://pauloswald.com/blog/article/29/hudson-solaris-smf-manifest)
* [Jenkins on Ubuntu](https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+on+Ubuntu)
* [Jenkins on Red Hat Distributions](https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+on+Red+Hat+distributions)
* [Jenkins as a Unix daemon](https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+as+a+Unix+daemon)
* ﻿[Jenkins with Docker](https://wiki.jenkins-ci.org/display/JENKINS/Installing+Jenkins+with+Docker)

Alternatively, if you have a servlet container that supports Servlet 2.4/JSP 2.0, such as Glassfish v2, Tomcat 5 (or any later versions), then you can run them as services, and deployjenkins.war as you would any other war file.

And finally, if you are having difficulty installing Jenkins with a container, you can always run Jenkins just by itself: java -jar jenkins.war (which uses Jetty)

Installing Jenkins with [IBM WebSphere](https://wiki.jenkins-ci.org/display/JENKINS/IBM+WebSphere) as container specific

**Warning**  
Running Jenkins on WebSphere 7.x requires WebSphere 7.0.0.7 or above. Jenkins will NOT work on WebSphere versions 7.0.0.0 - 7.0.0.5.

* The following installation was tested with WAS 6.1 ND, WAS 7.0 ND, WAS 8.0 Developer Edition and WAS 8.5 Developer Edition
* Some links in Jenkins are broken in WebSphere, but there is usually either another way to access them or it is intuitive to correct the URL in the address bar. It is not critical problem, thus no time to investigate it further.

# Requirements

* **WAS versions:**
  + Previously mentioned Jenkins versions should all run on WAS 6.1.

With WAS 7.0 (7.0.0.0, 7.0.0.1 and 7.0.0.3), a WebSphere class loader issue will prevent Hudson 1.291 or greater to start and use all its internal components

# Installation procedure

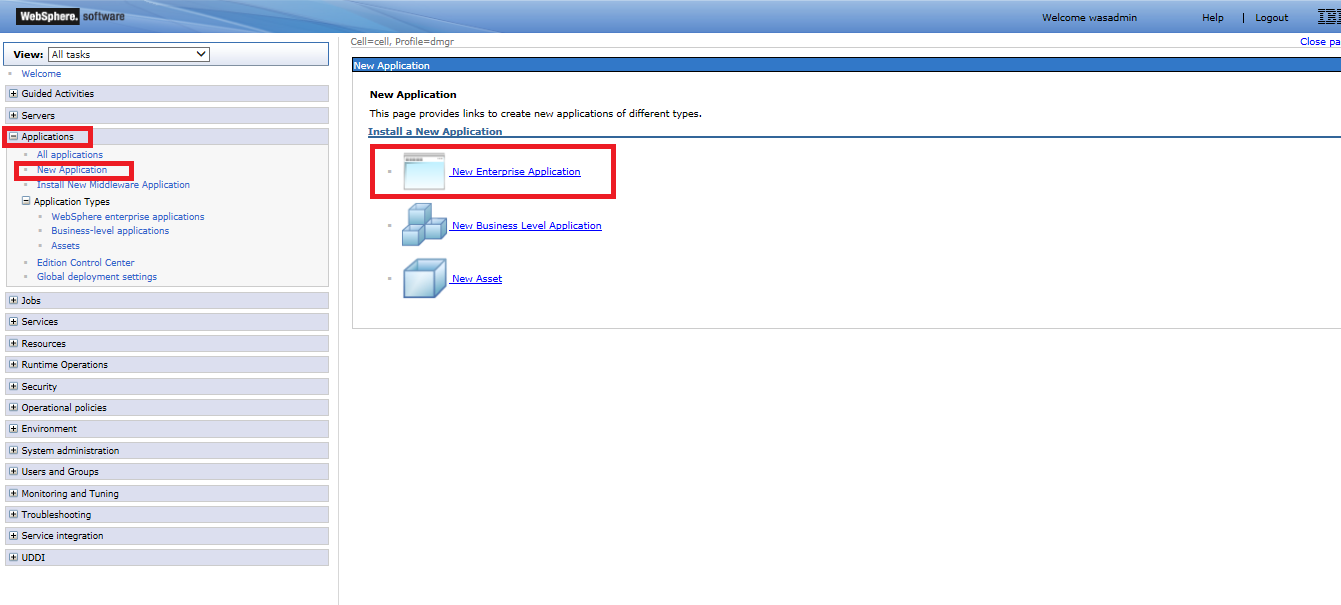
* [Manually through the Admin console](https://wiki.jenkins-ci.org/display/JENKINS/IBM+WebSphere#IBMWebSphere-console);

## Admin console installation

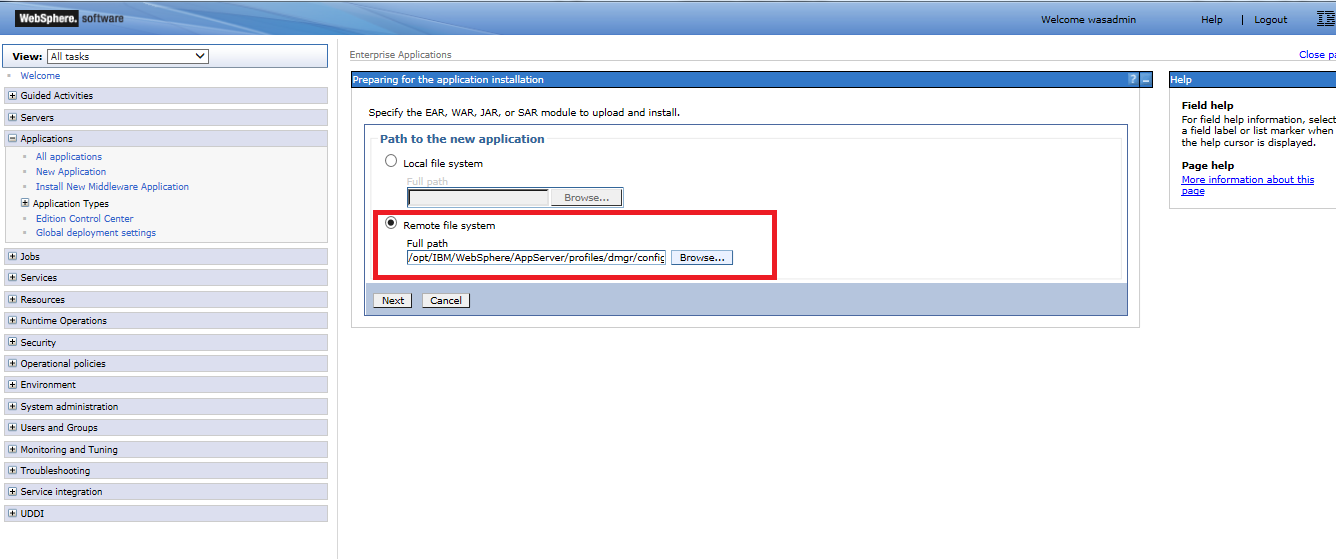
* The following works fine on WAS 6.1, WAS 7.0, WAS 8.0 and WAS 8.5.

The installation instructions are simple as for each WAR Module.

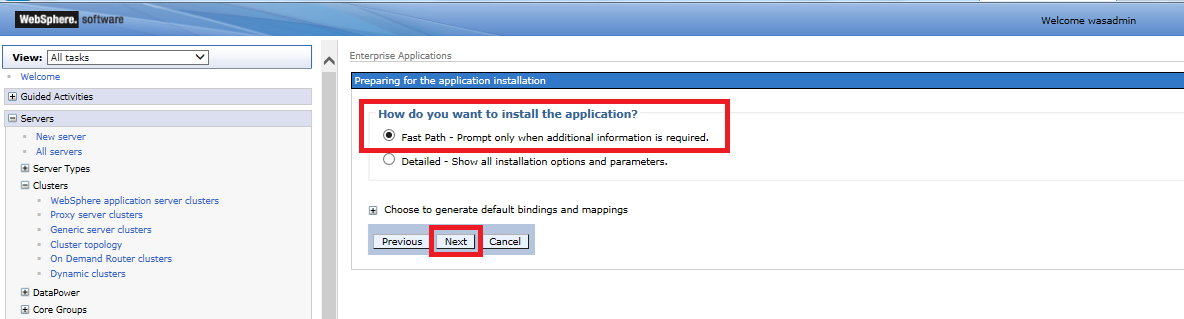
* Start the installation wizard under Applications - Install new application.



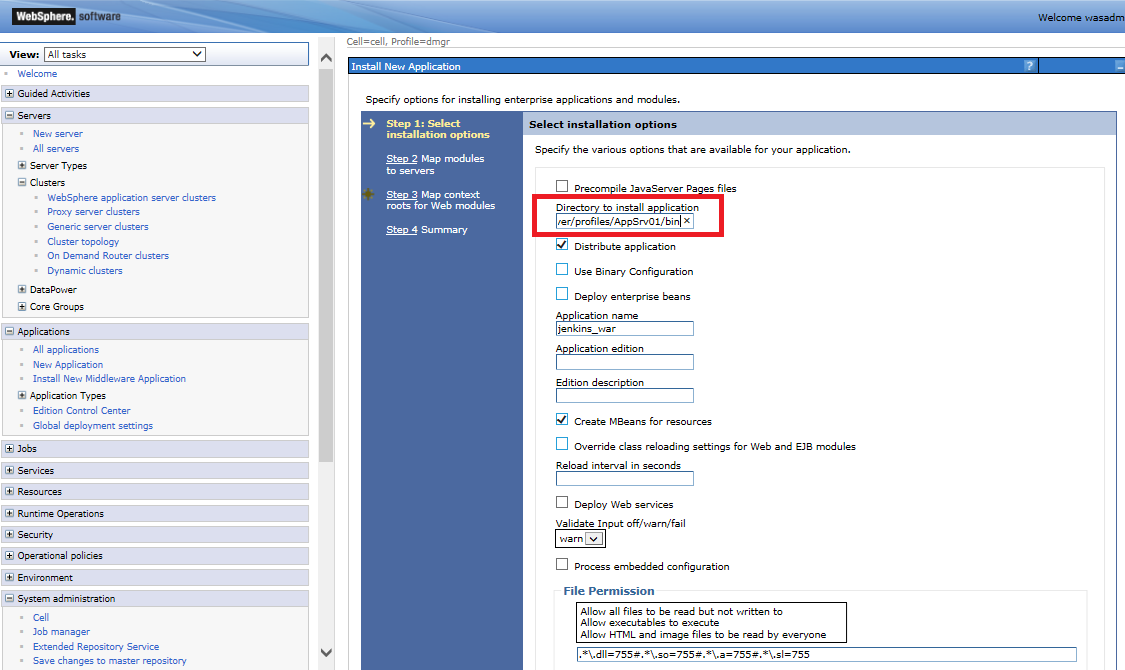
* Select Remote file System and then enter the context root for the application – In this example jenkins.war (the file must be downloaded and placed on a unix directory using a PSFTP connection before this steps)

****

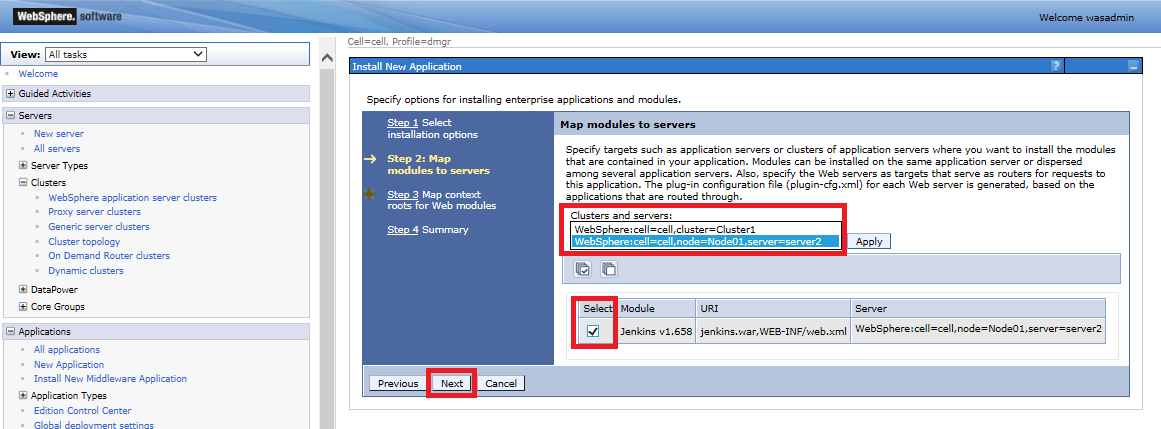
* Select the Fast Path Option



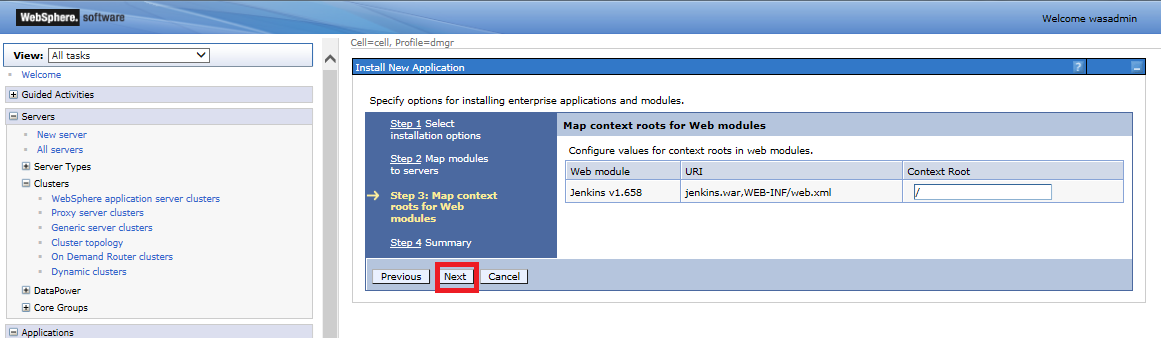
* Specify the Directory to install the application



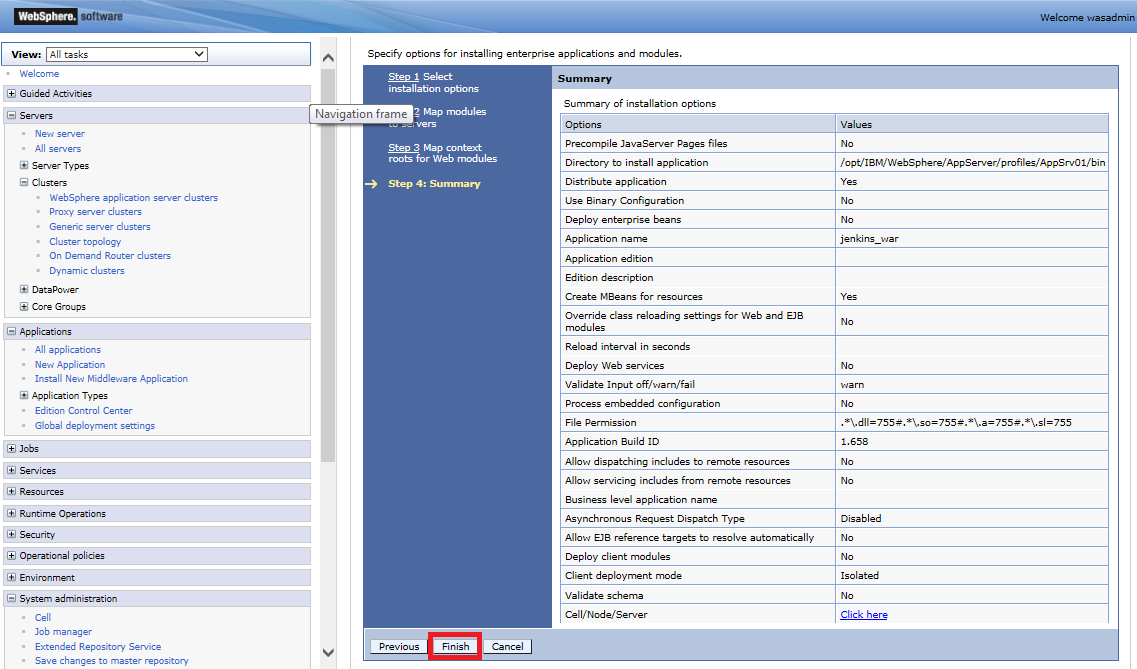
* Select the correct Cluster for the application. In this way the DMGR Knows to where the application Should be deployed (cluster and server)



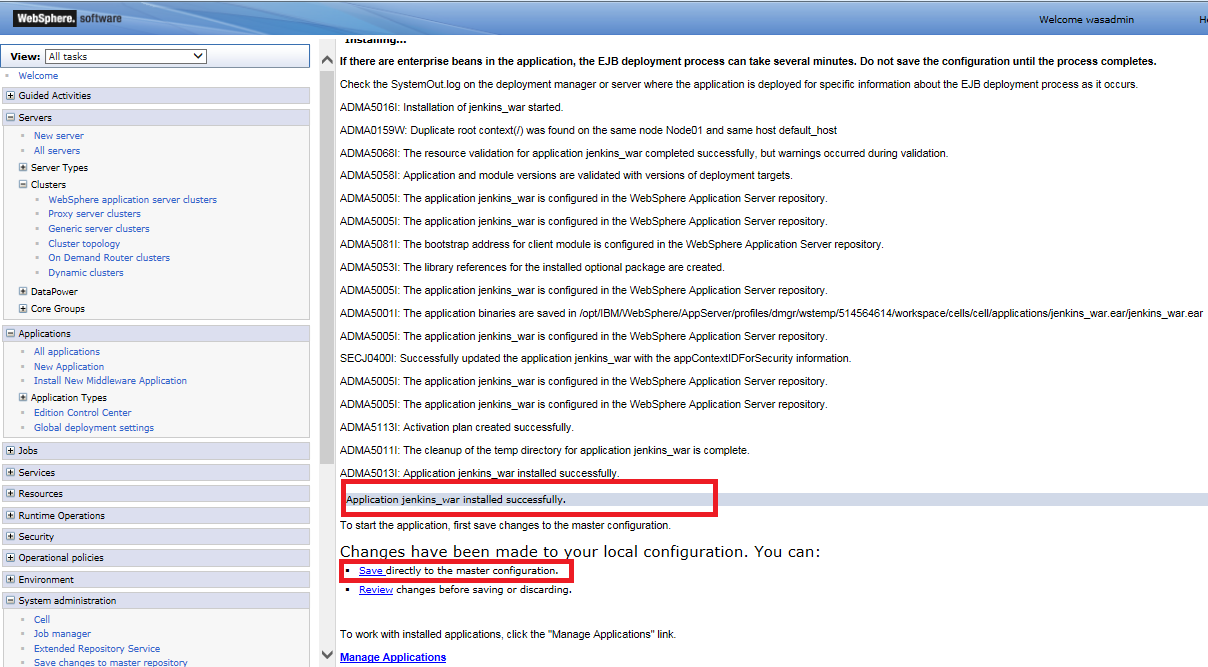
* Specify a Map context root if needed or use default provided and click next button

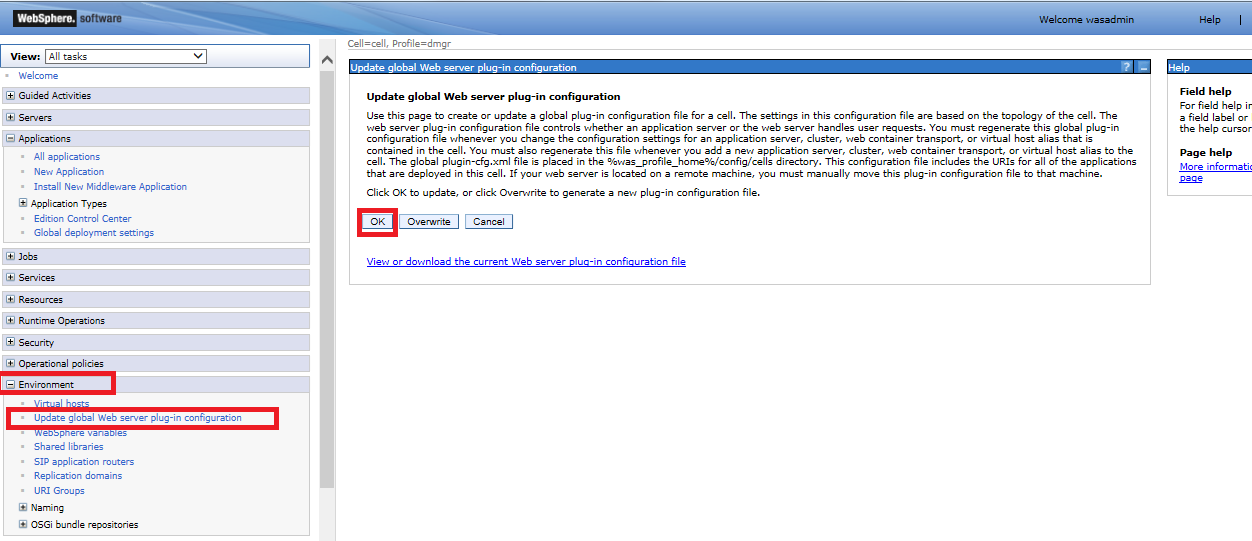


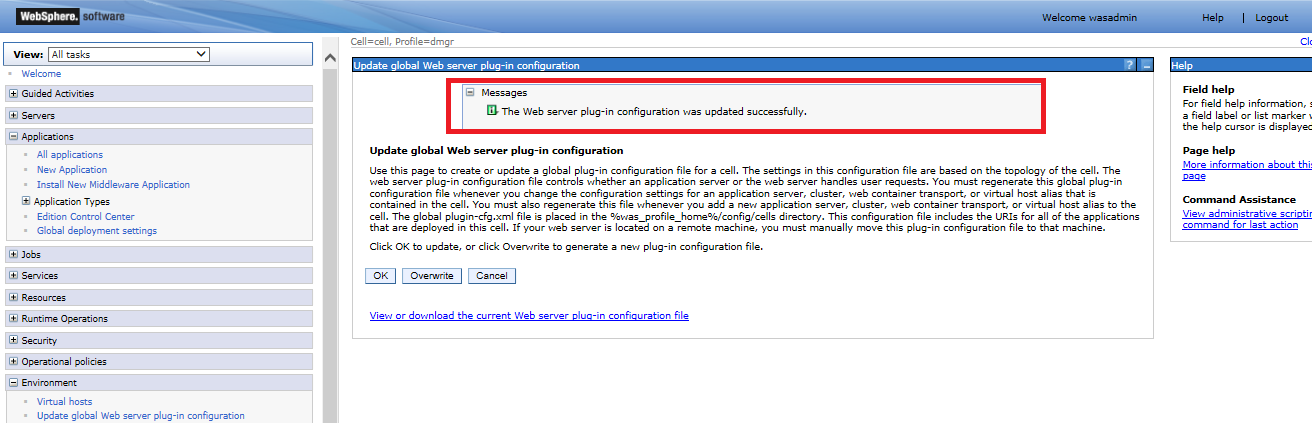
* Once Done you will see a Resume screen and click Finish.

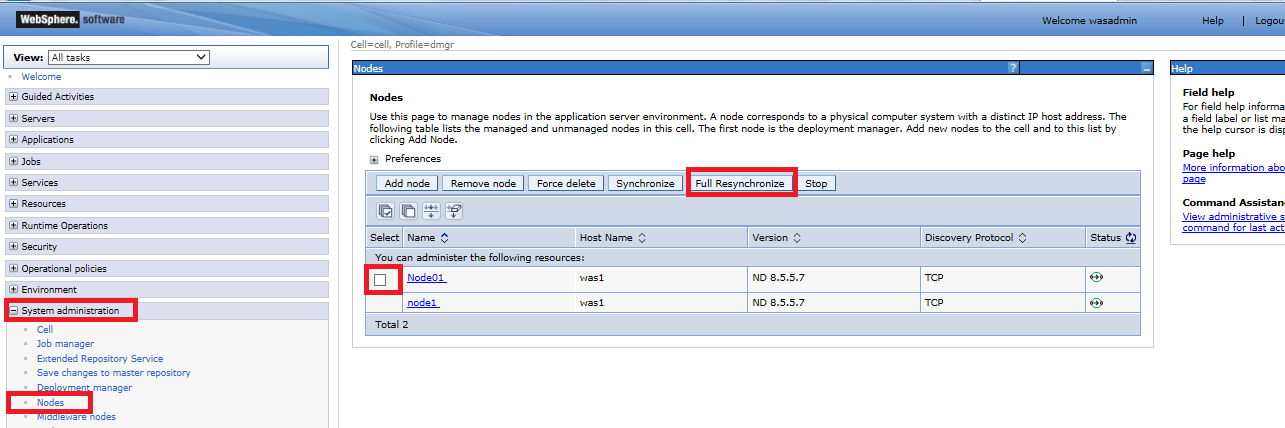


* You will see a progress screen and a message once the application has been installed

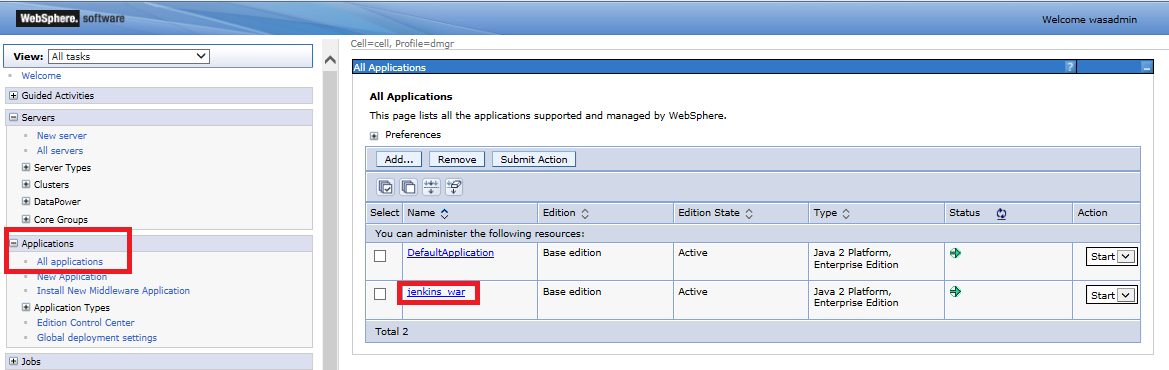


* Once the Application has been deployed with no errors, Regenerate the plugging and Sync the Node (FullResynchronize)****

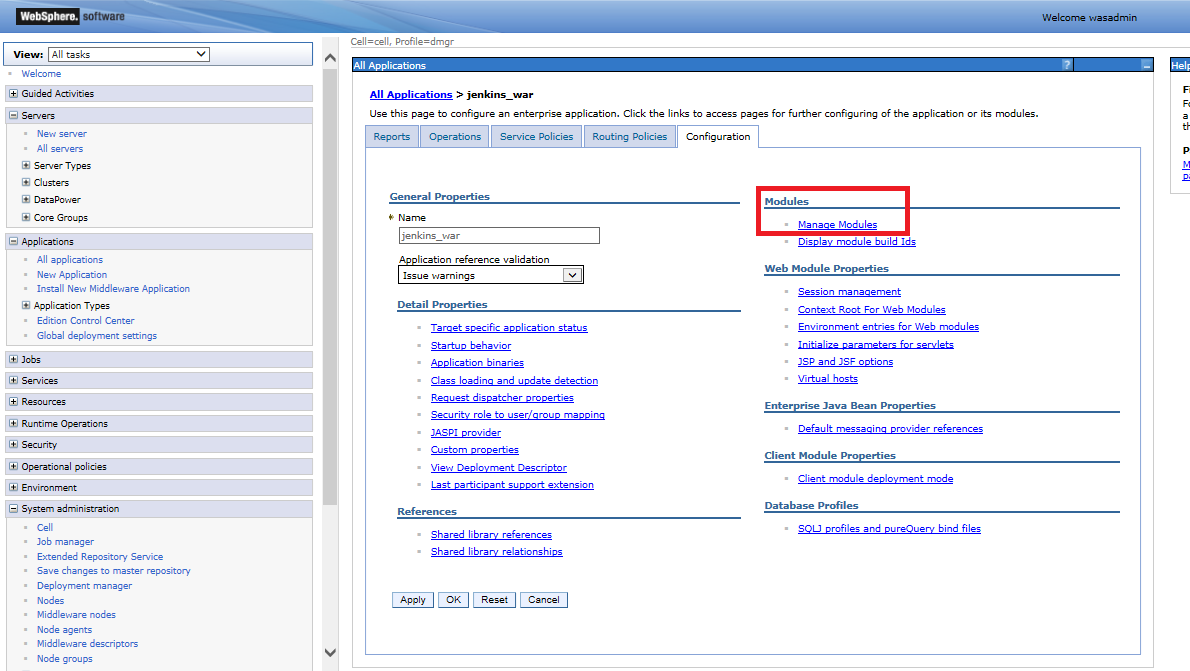
****

****

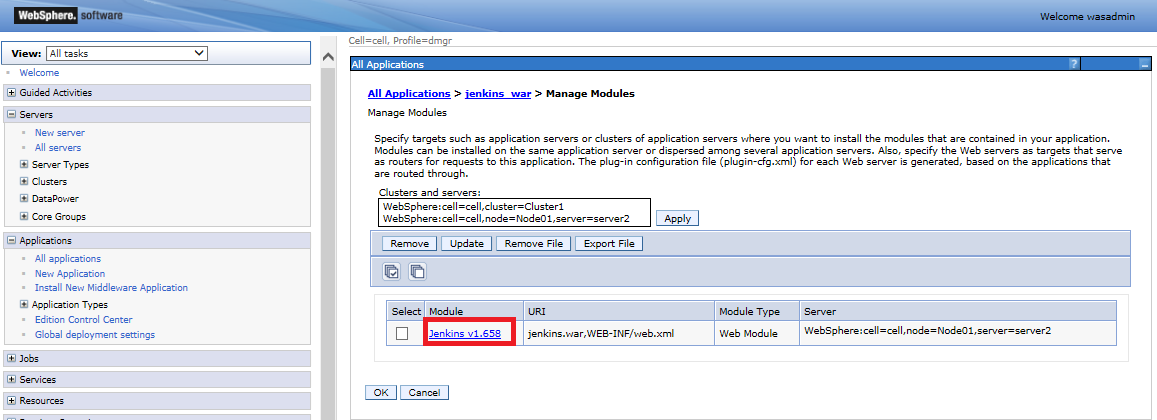
After finishing the wizard click the installed application (jenkins\_war in this example) and perform the following steps to change the classloader mode:

****

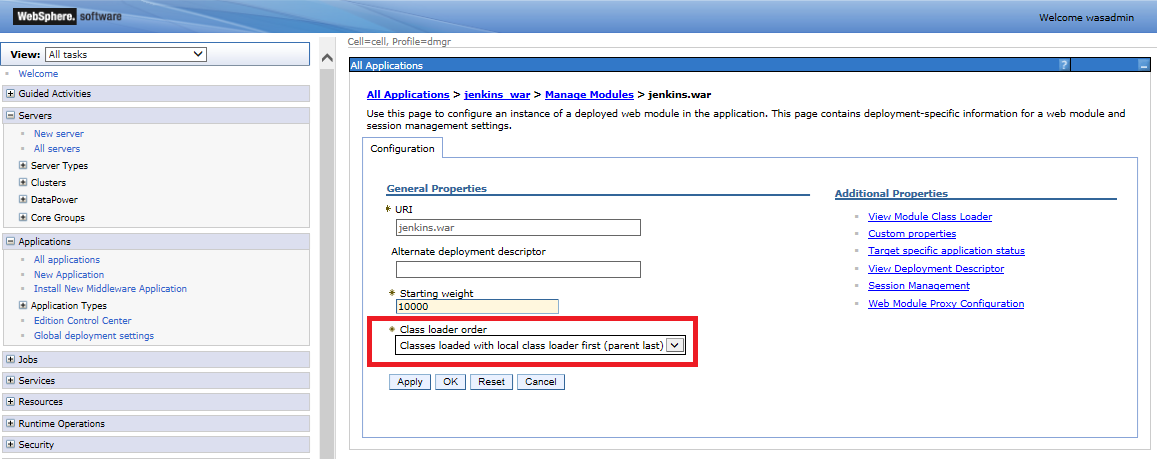
* Click Manage Modules

****

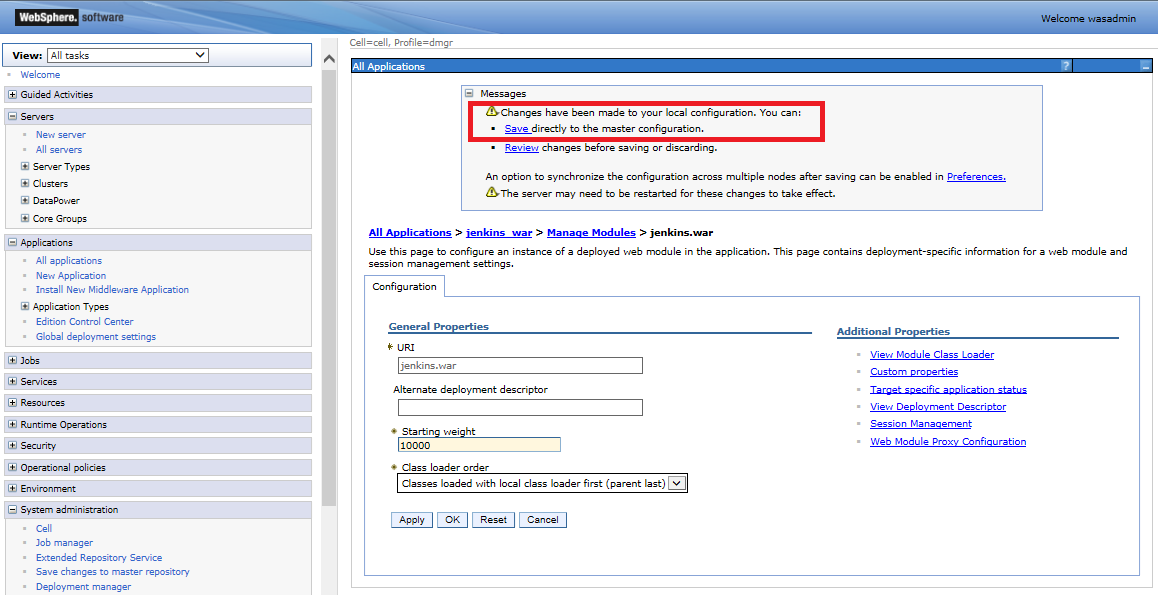
* Click Hudson / Jenkins

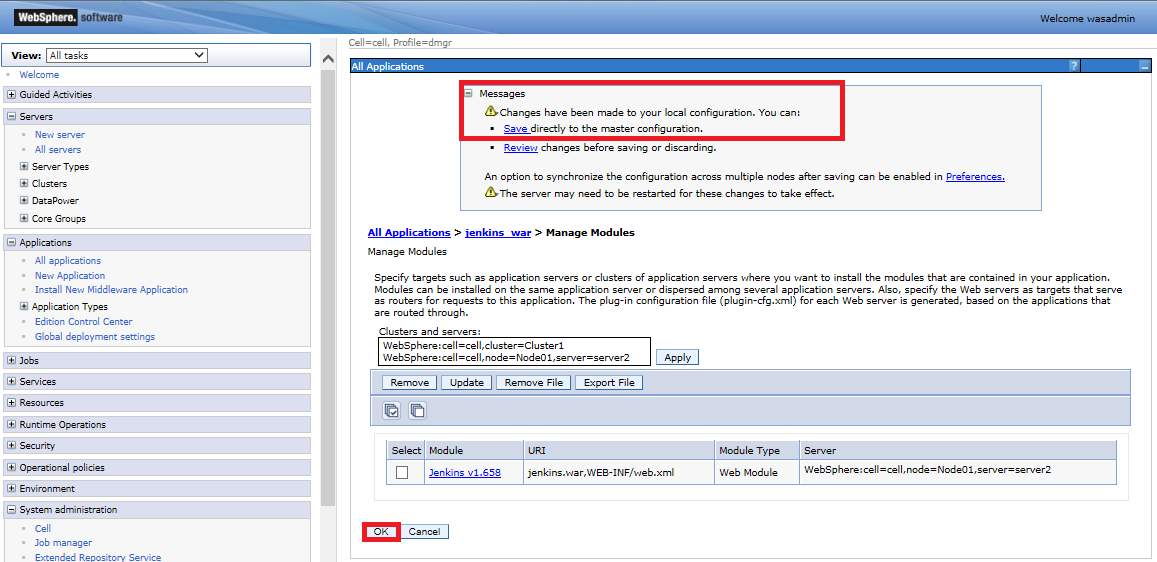
****

* From the "Class loader order" drop-down select "Classes loaded with application class loader first" ("Classes loaded with local class loader first" V8.5)

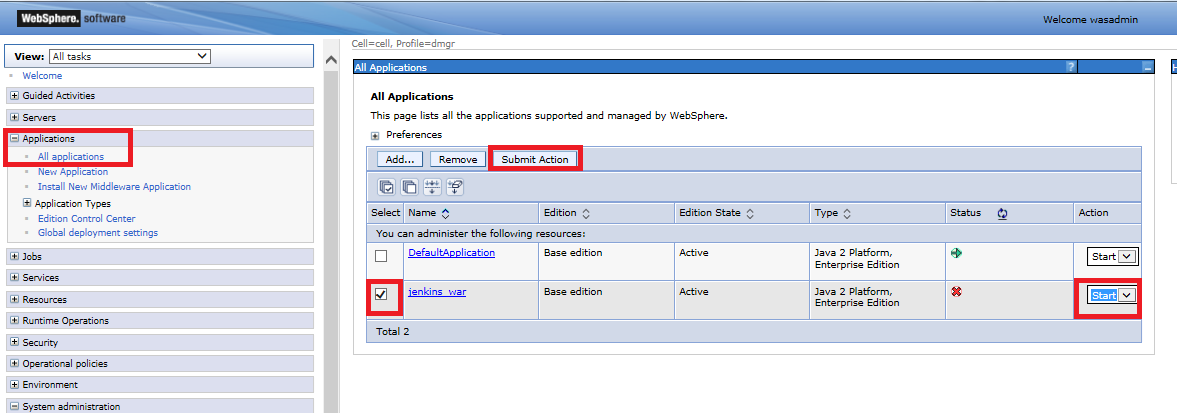
****

* Save the changes

****

****

* Restart the server - this is necessary to apply the changed class loader mode. You need to restart only the server onto which jenkins is installed. (V8.5: no need to restart the server, just start the Jenkins app).

****

Note:

If you try to install Jenkins 1.450 on WAS 8, you may face some exceptions mentioned in [JENKINS-12334](https://issues.jenkins-ci.org/browse/JENKINS-12334). This may occur, if an SLF4J implementation is missing.  
Since log4j is already on the Classpath, you might want to download the slf4j-log4j bridge and copy that JAR into the WEB-INF/lib folder of Jenkins.

**Jenkins Unix/Linux Installation as JVM**

* First of all you need to have a JDK and JRE installed. openjdk-7-jre and openjdk-7-jdk are suggested.
* Download the proper Jenkins.war file version from <https://jenkins.io/index.html>
* Create a Directory to place the war file, i.e:

mkdir -p /opt/jenkins/cache/jenkins/war

* Place the file in the root directory created which in this case will be:

/opt/Jenkins/

* Create a JVM running the following command changing the value for httpListenAddress:

/usr/local/java/jdk1.8.0\_91/bin/java -Dcom.sun.akuma.Daemon=daemonized -Djava.awt.headless=true -DJENKINS\_HOME=/opt/jenkins -jar /opt/jenkins/jenkins.war --logfile=/opt/jenkins/jenkins.log --webroot=/opt/jenkins/cache/jenkins/war --daemon --httpPort=8383 --httpListenAddress=192.168.15.4 --debug=5 --handlerCountMax=100 --handlerCountMaxIdle=20

* Once done you can run Jenkins with the following URL:

<http://LocalHost:Port/jenkins>

Which in this example will be:

<http://192.168.15.4:8383/jenkins>

Note: For this example the last command is running in foreground and command line interface will not be able to use. We need to execute Ctrl + z to stop the actual command and type bg 1 to restart it again in background.

Where #1 represents the number of the job. This can be confirm typing jobs command on command line.