# MAVEN Websphere Plugin

This is a Maven Plugin for WebSphere Applicaton/WebSphere Portal Application development, which providers below goals:

## For WebSphere Applications:

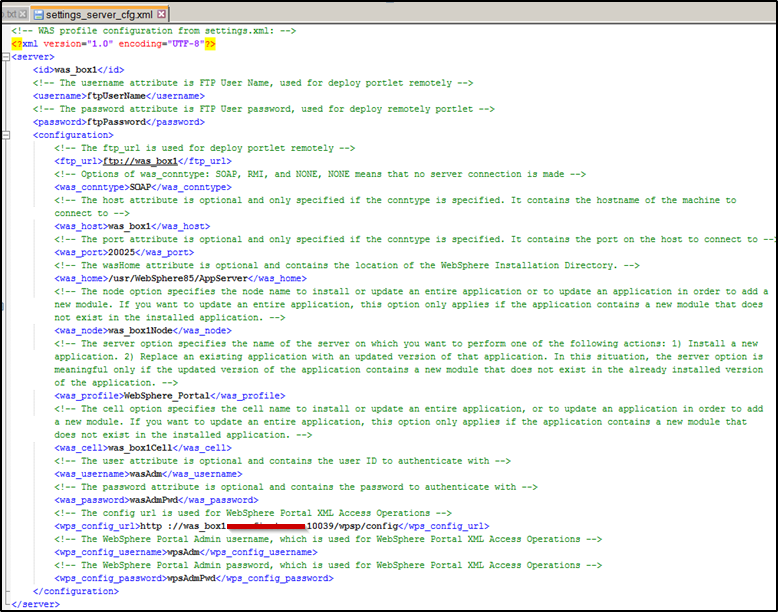
1. The **deploy-ear** task enables you to install a new application into a WebSphere Server or Cell.This task is a wrapper for the AdminApp.install() command of the wsadmin tool. Refer to the wsadmin documentation for information on the valid options available during application installation.   
     
   **Usages:**   
   1. Build and Deploy EAR Application   
   *mvn clean package websphere:deploy-ear -Dserver=was\_box1 -DwasHome=C:\IBM\WebSphere\AppServer*   
   2. Deploy an existing EAR Application   
   *mvn clean websphere:deploy-ear -Dserver=was\_box1 -DwasHome=C:\IBM\WebSphere\AppServer -DinstallableApp=C:\dist\BusinessDomainServices.ear*   
   3. Deploy an EAR Application from Nexus   
   *mvn clean websphere:deploy-ear -Dserver=was\_box1 -DwasHome=C:\IBM\WebSphere\AppServer -DinstallableApp =http://stype-nexus:8081/nexus/content/repositories/releases /com/xyz/BusinessDomainServices /9.34.87/BusinessDomainServices-9.34.87.ear -DappName=BusinessDomainServices*
2. The **deploy-war** task enables you to install a Web Application into a WebSphere Server or Cell.This task is a wrapper for the AdminApp.install() command of the wsadmin tool. Refer to the wsadmin documentation for information on the valid options available during application installation.   
     
     
   **Usages:**   
   1. Build and Deploy a WAR   
   *mvn clean package websphere:deploy-war -Dserver=was\_box1 -DcontextRoot=/wpsp/themes/HomePageTheme -DwasHome=C:/IBM/WebSphere/AppServer*   
   2. Deploy an existing WAR   
   *mvn clean websphere:deploy-war -Dserver=was\_box1 -DcontextRoot=/wpsp/themes/HomePageTheme -DwasHome=C:/IBM/WebSphere/AppServer -DinstallableApp=C:/dist/theme1.war*   
   3. Deploy a WAR from Nexus   
   *mvn clean websphere:deploy-war -Dserver=was\_box1 -DinstallableApp=http ://stype-nexus:8081/nexus/content/repositories/releases /com/xyz/theme1/9.34.87/theme1-9.34.87.war*
3. The **wsdl-to-java** task creates Java classes and deployment descriptor templates from a Web Services Description Language (WSDL) file. This task is a wrapper the com.ibm.websphere.ant.tasks.WSDL2Java, Refer to the IBM documentation for information on the valid options available.   
     
   *<wsdl2javatask url="location of input WSDL document"   
   output="root directory for emitted files"   
   role="J2EE development role"   
   container="j2ee container"   
   genjava="generate java files"/>*   
     
   **Usages:**   
   1. Generate Java classes base by WSDL   
   *mvn clean websphere:wsdl-to-java package -DwasHome=C:/IBM/WebSphere/AppServer -Durl=http://memsvcuser:M3msvcus3r@dveweb1/ secure/mws/services/UserAccess.wsdl -Doutput=${project.basedir}/src/main/generated*

## For WebSphere Portal Applications:

1. The **deploy-portlet** task enables you to install portlet into a WebSphere Portal Server. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Build and deploy a portlet   
   *mvn clean package websphere:deploy-portlet -Dserver=was\_box1*   
   2. Deploy an existing portlet:   
   *mvn clean websphere:deploy-portlet -Dserver=was\_box1 -DinstallableApp=C:/dist/portlet1.war*   
   3. Deploy portlet with customer uniqueNames   
   *mvn clean websphere:deploy-portlet -Dserver=was\_box1 -DinstallableApp=C:/dist/portlet1.war -DuniqueNames=com.xyz.plt1*   
   4. Deploy portlet from Nexus   
   *mvn clean websphere:deploy-portlet -Dserver=was\_box1 -DinstallableApp =http://stype-nexus:8081/nexus/content/repositories/releases /com/xyz/ptl1/9.34.87/ptl1-9.34.87.war*

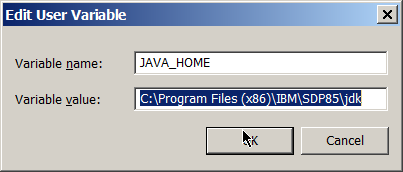
1. The **deploy-multi-portlet** task enables you to install multiable portlets into a WebSphere Portal Server. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Deploy multi portlets:   
   *mvn clean websphere:deploy-multi-portlet -Dserver=was\_box1 -DinstallableApps=C:/dist/portlet1.war,C:/dist/portlet2.war*   
   2. Deploy multi portlets with customer uniqueNames, with comma seperated uniqueNames args:   
   *mvn clean websphere:deploy-multi-portlet -Dserver=was\_box1 -DinstallableApps=C:/dist/portlet1.war,C:/dist/portlet2.war -DuniqueNames=com.xyz.plt1,com.xyz.plt1*   
   3. Deploy multi portlets from Nexus   
   *mvn clean websphere:deploy-multi-portlet -Dserver=was\_box1 -DinstallableApps =http://stype-nexus:8081/nexus/content/repositories/releases /com/xyz/ptl1 /9.34.87/ptl1-9.34.87.war,http://stype-nexus:8081/nexus/content /repositories/releases/com/xyz/ptl2/9.34.87/ptl2-9.34.87.war*
2. The **export-page** task enables you to export the WebSphere Portal Page based on UniqueName. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Export Page   
   *mvn clean websphere:export-page -Dserver=was\_box1 -DuniqueName=com.xyz.page.home1*   
   2. Export Page without Children   
   *mvn clean websphere:export-page -Dserver=was\_box1 -DuniqueName=com.xyz.page.home1 -Drecursion=false*
3. The **export-portlet** task enables you to export the WebSphere portlet(s) based on UniqueName(s). This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Export single portlet   
   *mvn clean websphere:export-portlet -Dserver=was\_box1 -DuniqueName=com.xyz.ptlt1*   
   2. Export multi portlets   
   *mvn clean websphere:export-portlet -Dserver=was\_box1 -DuniqueName=com.xyz.ptlt1,com.xyz.ptlt2*
4. The **export-theme** task enables you to export the WebSphere Theme based on Theme. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Export theme   
   *mvn clean websphere:export-theme -Dserver=was\_box1 -DuniqueName=com.xyz.theme1*
5. The **export-urlmapping** task enables you to export the WebSphere urlmapping based on Uniquename or label. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Export URL Mapping by uniqueName   
   *mvn clean websphere:urlmapping -Dserver=was\_box1 -DuniqueName=com.xyz.urlmapping.friend1*   
   2. Export URL Mapping by label   
   *mvn clean websphere:urlmapping -Dserver=was\_box1 -Dlabel=friend1*
6. The **import** task enables you to import a XMLAccess XML into WebSphere Portal. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Import a XMLAccess XML   
   *mvn clean websphere:import -DimportXML=c:/xmls/pages\_portal1.xml*
7. The **migrate-page** task enables you to migrate XMLAccess Page XML from source WPS server to target WPS server. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Migrate a specified page from was\_box1 to was\_box2   
   *mvn clean websphere:migrate-page -DsourceServer=was\_box1 -DtargetServer=was\_box2 -DuniqueName=com.xyz.page.home1*   
   2. Migrate a specified page from was\_box1 to was\_box2 without child pages   
   *mvn clean websphere:migrate-page -DsourceServer=was\_box1 -DtargetServer=was\_box2 -DuniqueName=com.xyz.page.home1 -Drecursion=false*
8. The **migrate-portlet** task enables you to migrate XMLAccess Portlet XML(s) from source WPS server to target WPS server. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Migrate single portlet from was\_box1 to was\_box2   
   *mvn clean websphere:migrate-portlet -DsourceServer=was\_box1 -DtargetServer=was\_box2 -DuniqueName=com.xyz.ptlt1*   
   2. Migrate multi portlets from was\_box1 to was\_box2   
   *mvn clean websphere:migrate-portlet -DsourceServer=was\_box1 -DtargetServer=was\_box2 -DuniqueNames=com.xyz.ptlt1,com.xyz.ptlt2*
9. The **migrate-theme** task enables you to migrate XMLAccess theme XML from source WPS server to target WPS server. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
     
     
   **Usages:**   
   1. Migrate Theme XML from was\_box1 to was\_box2   
   *mvn clean websphere:migrate-theme -DsourceServer=was\_box1 -DtargetServer=was\_box2 -DuniqueName=com.xyz.theme1*
10. The **migrate-urlmapping** task enables you to migrate XMLAccess Urlmapping XML from source WPS server to target WPS server. This task is a wrapper for the XMLAccess command of the WebSphere Portal. Refer to the WebSphere Portal documentation for more information.   
      
      
    **Usages:**   
    1. Migrate urlmapping from was\_box1 to was\_box2 by uniqueName   
    *mvn clean websphere:migrate-urlmapping -DsourceServer=was\_box1 -DtargetServer=was\_box2 -DuniqueName=com.xyz.urlmapping.friend1*   
    2. Migrate URL Mapping by label   
    *mvn clean websphere:migrate-urlmapping -DsourceServer=was\_box1 -DtargetServer=was\_box2 -Dlabel=friend1*

## Maven Settings.xml configuration:





## Download and Usages:

1. For executing WebSphere Maven Plugin, you need to use the IBM RAD JDK or WebSphere JRE, set the JDK/JRE in environment:  
   

For executing the WebSphere Application Tasks, you need to have IBM WebSphere instance in your local, and create at least one profile under it.