EVSS Developer Setup (without portal)

This document contains the steps to setup development tools common across all non-portal projects.

Install Subversion Client

Subversion is used on EVSS for version control of all source code and other technical artifacts. Typically you will use the SVN capabilities of your IDE (i.e. Eclipse). If you want though you may find it useful to install the SVN command line client or a GUI client.

Command Line Client

a. Download SilkSVN , the recommended SVN client for EVSS, from https://sliksvn.com/download/. Make sure to download the correct version for your operating system (32-bit vs 64-bit). If you are unsure use the 32-bit version.

GUI Client

TortoiseSVN is the recommended GUI SVN client for EVSS. It can be downloaded from http://tortoisesvn.net/downloads.html. Make sure to download the correct version for your operating system (32-bit vs 64-bit). If you are unsure use the 32-bit version.

Install Java

1. Download the latest version of Java Development Kit (JDK) 8 from Oracle.
   1. a. Make sure to download the appropriate version (32bit vs. 64bit) for your operating system. If in doubt, use the 32-bit version.

 b. NOTE: Our software deploys to servers that run JDK 6. We use JDK 8 locally so we can run the latest Eclipse, but the

Maven poms specify our JDK source/target are 6 thus the local JDK will still compile down to the 6 level.

2. After install, set your JAVA\_HOME environment variable to your JDK installation directory and alter your PATH to append **;%JAVA\_HOME%\bin**

3. Open a command line terminal and run the following command to verify your installation

java -version

Output should look something like:

java version <VERSION\_DISPLAYED\_HERE>

Install Maven

1. It is recommend you install version 3.0.4 as this has been used and works. You can download from:  http://maven.apache.org/download.html
2. It is recommended that you install this version of Maven to **C:\va\_wss\_dev\apache-maven-3.0.4**
3. After install, set your MAVEN\_HOME environment variable to **C:\va\_wss\_dev\apache-maven-3.0.4** and alter your PATH to  append **;%MAVEN\_HOME%\bin**
4. Open a command line terminal and run the following command to verify your installation  mvn -version

Checkout source code from SVN

TBA.

Build with Maven

Use a local maven repo if an external repo is unavailable. A repo and sample settings file is available.

1. Copy the entire maven repository to a local folder: ***C:\dev\software\EVSS\m2Repo***

2. Run maven from the command line to create the basic *.m2* structure in the local user directory (e.g. *C:\Users\vtanase\.m2*).

a. Probably you can skip this step and just create it manually.

b. Or you can import the java projects into eclipse, as maven projects (*File >> Import >> Maven >> Existing Projects*)

c. Right click on a project and select: *Run As >> Maven clean*.

d. This will create the .m2 folder structure under your local NT profile.

4. Copy *settings.xml* from *m2Repo* to your local *.m2* folder.

 5. Instruct maven to use the local repository to build and resolve the dependencies (i.e. basically replace the nexus folder with the local repositories -

*C:\dev\software\EVSS\m2Repo*).

 a. Go to your local user folder, under the *.m2* sub folder and edit *settings.xml*.

1. replace: ***existing repo URLs***, with:

ii. **file://C:\dev\software\EVSS\m2Repo\repository**

1. Run a build from the command line. If you are using a local maven repo, the build for wss-common-services may fail due to a missing jar. Download the missing version and install it in your local repo using: mvn install:install-file

Install Ant

1. Download the latest version of Ant from https://ant.apache.org/bindownload.cgi
2. Unzip the installation into the recommended location of **C:\va\_wss\_dev\apache-ant**, or choose your own location
3. After install, set your PATH environment variable to append **;C:\va\_wss\_dev\apache-ant\bin** (or your custom installation  location)
4. Openacommandlineterminalandrunthefollowingcommandtoverifyyourinstallation  ant -version  Output should look something like:   Apache Ant(TM) version 1.9.4 compiled on April 29 2014

Install Virtual Machine Manager

VirtualBox is the recommended Virtual Machine manager for EVSS since it is available free of charge on all platforms. To install it on your machine follow the instructions below.

1. Dowload the latest version of VirtualBox from https://www.virtualbox.org/wiki/Downloads. Make sure to download the appropriate version for your operating system.
2. Follow the instructions on the installer for your specific OS.

Setup Local Database

The EVSS project uses Oracle for its database provider. Since running a full blow oracle server on your machine would consume a lot of resources we use a lighter-weight OracleXE instance running on a VM instance.

1. Download the latest VM

 2. Import the VM into the Virtual Box software

3. From the menu at the top of VirtualBox select Machines->Add and select the VirtualBox Machine Definition file. 4.

Install / Configure IDE

**Eclipse Mars (4.5)**

1. Install the Mars version (4.5.1) of Eclipse. Choose the version **Eclipse IDE for Java EE Developers** from http://www.eclipse.org/downloads/. **Note:** Make sure that both the Eclipse and JDK are of the same n-bit version (i.e 64-bit version or the 32-bit version)

a. To support multiple side-by-side Eclipse installations, it is recommended that you unzip Eclipse to the directory **C:\va\_wss\_dev** with the version number intact. For example, **C:\va\_wss\_dev\eclipse-jee- mars** or **C:\eclipse-jee-mars**

* 1. After installing (unzipping) Eclipse start it by running the eclipse.exe found in Eclipse install dir (i.e. **C:\va\_wss\_dev\eclipse-jee-mars\eclipse**)
  2. At the prompt, specify a path for your workspace. The recommended workspace for wss trunk development is **C:\va\_wss\_dev\workspaces\wss-all-trunks**

1. Make sure that the Java setting points to a JDK and not a JRE:

a. In Eclipse, go to **Window** > **Preferences** > **Java** > **Installed JREs**b. The checked JRE **Location** column should point to a JDK folder. If this is not the case, point to a local JDK installation  (i.e. the JDK that ships with WLS) and click ok.

1. OPTIONAL: Install Spring Tool Suite (from within Eclipse search for this via the Help - Eclipse Marketplace option)
2. Install Maven Integration for Eclipse (m2e)
3. Install Maven Integration for WTP (m2e-wtp) (from within Eclipse search for this via the Help - Eclipse Marketplace option)
4. Install the Subversive plugin (from within Eclipse search for this via the Help-EclipseMarketplace option)

 a. Make sure the m2e subversive plugin also is installed. After installing Subversive: in the Eclipse Project Explorer choose "Import - Import - Maven - Check out Maven projects from SCM." In the popup window in the bottom right click the link to "Find more SCM connectors from m2e Marketplace." In that popup locate the m2e subversive plugin, possibly near  the bottom.

1. After install, go to Window – Preferences – Team – SVN and change SVN Interface to SVNKit (you may get an error  which you can ignore prior/during doing this because JNI may not be available)
2. Go to Window --> Open Perspective --> SVN Repository Exploring

d. Add a new repo with URL=TBA