Creating a new eSDRT development environment on Windows 2018-07

# Section 1: The following steps are for non-Java developers:

## Copy a clean virtual machine (VM) image, Dev\_Win7\_EN.zip, to your base machine (Windows likely), and unzip it. Remember where you store it.

## Open the VM using VMWare:

* 1. Start up VMware.
  2. Click the “File” menu item.
  3. Click “Open…”.
  4. In the “Open” popup window, navigate to the place where the clean VM image is stored, select it and open it.
  5. From VMWare, click the VM name and then click “Edit virtual machine settings”, the “Virtual Machine Settings” popup window is displayed.
  6. In the “Hardware” tab, alter Memory size (recommend 8 gig if possible), and alter Processors if possible (recommend at least 4).
  7. In the “Options” tab, rename the VM to your favorite name, such as “ESDRT\_VM\_WIN7”.
  8. Clicks “Shared Folders”, select “Always enabled” and clicks on “Add…” to open the “Add Shared Folder Wizard”. Follow the steps and add your designated folder to this virtual machine. **NOTE**: Check off “Map as a network drive in Windows guests”. This shared folder will be used to transfer files between your VM and your host machine.
  9. Click “OK” to save changes.

## Power on the VM. When you see the Windows logon screen, enter “hcuser” in the username field to log on to Windows (no password is needed).

## Update Internet Explorer to version 11 if necessary.

## Install SQL Developer 4.2.0:

* 1. Download SQL Developer from <http://www.oracle.com/>.

**NOTE**: You can download a package with either JDK 8 included or not.

* 1. Extract files to an appropriate location and execute sqldeveloper.exe from that location.
  2. After the installation is completed, you can create connections if you want.

## Install Firefox:

* 1. Download Firefox from <https://www.mozilla.org/en-CA/>.

**NOTE**: As of 2018-07-25, the Firefox version is 61.

* 1. Click the downloaded file, and follow the instructions to install Firefox.

## Install Smart CVS 7.1.9:

* 1. Download and install Smart CVS from <https://www.syntevo.com/smartcvs/>.

## Check out the ESDRT project:

* 1. Start up SmartCVS.
  2. Click the “Project” menu item.
  3. Click “Repository Profiles”.
  4. In the Repository Profiles popup window, click “Add” and follow the steps to add the Repository profile. The resulted profile should be something like “:pserver:rwang@cvs:/srv/cvs”.
  5. Back to the SmartCVS main window, click the “Project” menu item and click “Check Out…”, the “Check Out Project” popup window is displayed. Steps to check out a project are listed on the left panel.
  6. In the first step “Repository”, select the repository that you want. Click “Next” to go to the next step.
  7. In the “Modules” step, you need to select the module you want to check out. Click “DevRepository” -> “FNIHB” -> “000022EHRTJJ” -> “ESDRT”. Click “Next” go to the next step. Click the “Modules” tab and select “ESDRT”.
  8. In the “Target Directory” step, you need to select a local folder where the checked out project will be stored. Check off “Check out into alternative path (instead of module path/name) and enter “ESDRT” into the Alternative Path field. Click “Next” after you make the selection.
  9. In the “Checkout Options” step, keep the default and click “Next” to go to the next step.
  10. In the “Project Settings” step, make no changes and click “Next” to go to the next step.
  11. In the “Confirmation” step, click “Finished”.

## Install Java 1.8:

* 1. Download jdk-8u181-windows-x64.exe from <http://www.oracle.com/>.
  2. Click the downloaded file, and follow the instructions to install Java.
  3. Add the Java\_Home variable to the Windows environment and point it to the Java location, e.g. “C:\Program Files\Java\jdk1.8.0\_181”.
  4. Update the path variable by appending the Java bin folder, “%JAVA\_HOME%\bin”.
  5. Verify the Java version by running “Java -version” from the command line.

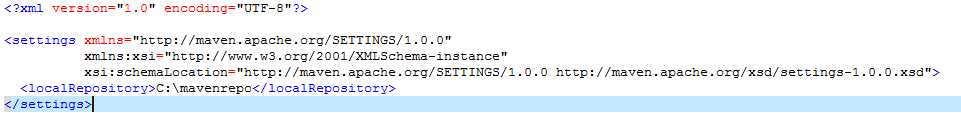
## Install Tomcat 8:

* 1. Download apache-tomcat-8.0.50-windows-x64.zip from <http://tomcat.apache.org/>.
  2. Extract files to an appropriate location and note this location.
  3. Add the Catalina\_Home variable to the Windows environment and point it to the Tomcat location, e.g. “C:\apache-tomcat-8.0.50”.

## Install Maven 3.5.4:

* 1. Download apache-maven-3.5.4-bin.zip from <https://maven.apache.org/>.
  2. Extract files to an appropriate location and note this location.
  3. Add both the M2\_HOME and MAVEN\_HOME variables to the Windows environment and point it to the Maven location, e.g. “C:\apache-tomcat-8.0.50”.
  4. Update the path variable by appending the Maven bin folder, “%M2\_HOME%\bin”.
  5. Verify the Maven version by running “mvn -version”.

**NOTE**: Maven settings are specified in the settings.xml file and this file can be found in two locations: “${maven\_home}/conf/” and “{user\_home}/.m2”. Be sure to check the contents in both files because if you inherit the VM from somebody else, there may be some existing settings that you don’t want to use. For example, Health Canada only wants you to download libraries from an internal repository and we don’t want this. Just use the default settings that are downloaded from the website. By default, Maven downloads all libraries from <https://repo.maven.apache.org/maven2/>. This is what we want. You can also specify your own favorite location for the local repository. The following is my settings.xml file:



# Section 2: Build and Run the ESDRT Application:

**NOTE**: The requirements for working with the ESDRT project are Java 8 (or later), CVS, Maven and Tomcat 8 (or later). Eclipse is not required. You can do a full compile and build a WAR file just from the command line.

1. From the command line, go to the location that stores the ESDRT project by executing something like the following:

cd c:\workspace\ESDRT

1. Download maven jars to local maven repository, and also non-maven jars must be installed to the repository. To do this, use Maven initialize:

mvn initialize

1. Build the project:

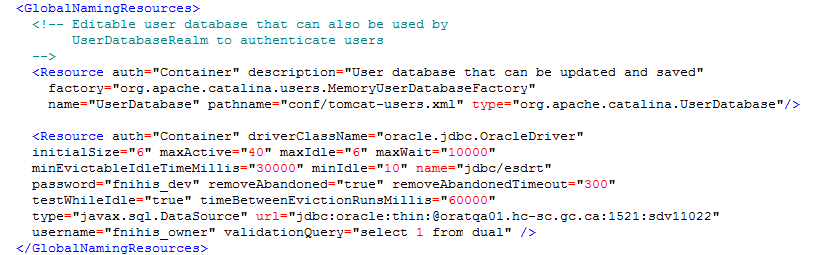
mvn clean install

After the project is successfully built, the war file ESDRT.war will be found in the target folder of the ESDRT project.

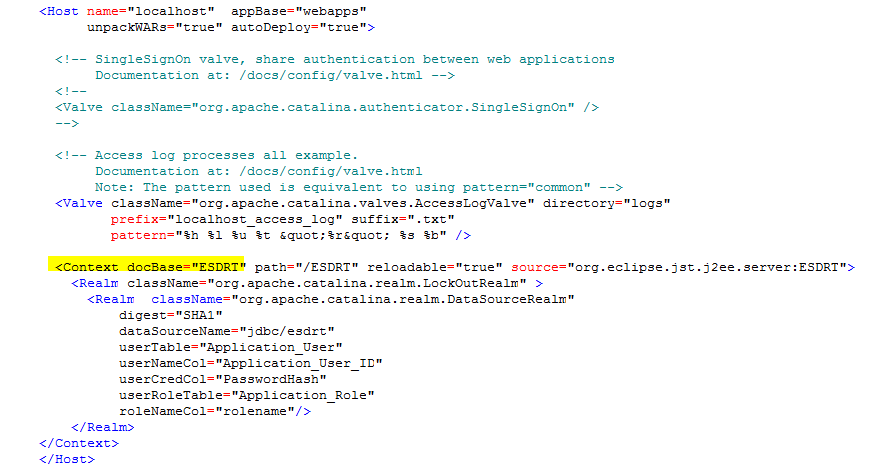
1. Configure Tomcat:

The datasource used for the system must be defined in the Tomcat server.xml file (found in the conf folder of your local Tomcat installation), as must the configuration for the form based authentication mechanism. The basic steps required to a stock server.xml file are:

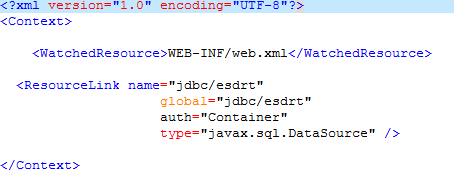
1. Add a datasource in the “GlobalNamingResources” section:



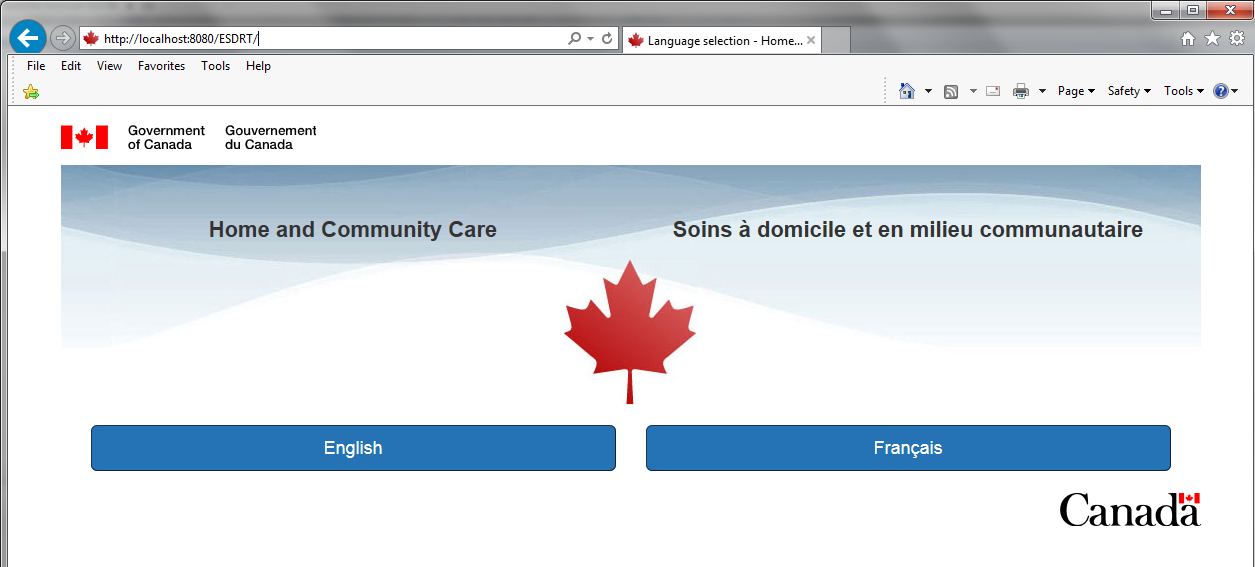
1. After ESDRT has been added as a web application, it should be listed in the server.xml file as a “context”. Expand the context with a security realm. If no context section is found, create one:



1. Update the context.xml file (found in the conf folder of your local Tomcat location) with the following contents:



1. Copy ojdbc6-11.2.0.3.0.jar from the NonMavenJars folder of the ESDRT project to the Tomcat’s lib folder. Note that this step may not be necessary.
2. Copy the ESDRT.war file to the webapps folder of the Tomcat installation.
3. Go to the bin folder of the Tomcat installation, run startup.bat from the command line. Tomcat should start up without errors.
4. Open a browser and enter <http://localhost:8080/ESDRT/> in the address bar, you should see the ESDRT home page:



# Section 3: Java developers can continue to install Eclipse and run ESDRT within it:

## Install Eclipse Oxygen:

* 1. Download the Eclipse-Oxygen (eclipse-inst-win64.exe, currently version 4.7.3a) from <https://www.eclipse.org/oxygen/>.
  2. Navigate to where you downloaded Eclipse and execute eclipse-inst-win64.exe.
  3. Double-click “Eclipse IDE for Java EE Developers” in the eclipse installer.
  4. Follow the steps to install the Eclipse.

## Install Eclipse Plug-ins:

1. Install the CVS Plug-in:
   * 1. In Eclipse, click “Help” -> “Install New Software...”.
     2. In the "Work with:" field, click the drop-down menu, and select the "--All Available Sites--" option.
     3. From the list check "Eclipse CVS Client".
     4. Select “Next” twice, accept the license, and click “Finish”.
2. Create a CVS repository:
   * 1. In Eclipse, click “Windows” -> “Perspective” -> “Open Perspective”, and select “CVS Repository Exploring”.
     2. Right click the blank space in the “CVS Repository” view, and select “New” and then “Repository Locations…”.
     3. In the “Add CVS Repository” popup window, enter necessary information, and click “Finish”. The new repository will be something like “:pserver:yourname@cvs:/srv/cvs”.
     4. Explore the new repository and look for the ESDRT project by click “HEAD” -> “eHRTTj” -> “ESDRT”.
3. Install the Groovy Plug-in:
4. In Eclipse, click “Help” -> “Install New Software...”.
5. For Groovy support in Eclipse, use the following as a new software source and install all components. This may need to be changed in the future when the plug-in hits general availability. In the "Work with:" field, type in “http://dist.springsource.org/release/GRECLIPSE/e4.7/”, and hit the enter key.

**NOTE**: This version is for Eclipse Oxygen, version 4.7. Make sure to match the plugin to your version of Eclipse.

1. Check everything in the list.
2. Select “Next” twice, accept the license, and click “Finish”.
3. Install the Jaspersoft Studio Plugin:
   * 1. In Eclipse, click “Help”.
     2. Click on “Eclipse Marketplace”.
     3. Search for “Jaspersoft Studio”.
     4. Click “install” to install the available version, currently version 6.6.0.
     5. Accept terms.
     6. If you receive a warning about unsigned software, accept the software.
     7. A restart of eclipse will likely be needed

## Prepare the ESDRT project for Eclipse:

From the command line, go to the ESDRT root folder and execute the following command:

mvn eclipse:eclipse

This Maven command will generate Eclipse project configuration files.

## Open the ESDRT project in Eclipse:

* 1. From the “File” menu, choose “Import…”.
  2. In the “Import” popup window, click “Maven” and select “Existing Maven Projects”, and click “Next”.
  3. Browse to your ESDRT root folder and select it.
  4. Click “Finish”.
  5. When done, if there any errors, leave them and we’ll deal with them later.

## Setup Tomcat in Eclipse:

* 1. From main Eclipse screen, select on the “Servers” tab in the window at the bottom of your Eclipse window.
  2. Click on link to add a new server
  3. In the server type of server drop down, choose the appropriate Apache Tomcat version (probably Tomcat v8.0 Server).
  4. Click “Next”.
  5. Browse to the Tomcat location that you downloaded and extracted.

**NOTE**: My recommendation is to have a separate Tomcat folder from Step 8 of Section 1 and this server is only for Eclipse.

Choose the jdk folder as your JRE.

* 1. Click Finish
  2. From this same “Servers” tab, double click the newly created Tomcat server.
  3. Click “Open Launch configuration” in “Overview” window.
  4. Click the “Classpath” tab.
  5. Click “User Entries”.
  6. Click the “Add External Jars” button.
  7. Browse to find the ojdbc6-11.2.0.3.0.jar (probably in the “NonMavenJars” folder of the ESDRT project or wherever you downloaded if you download this file yourself), click “Open”.
  8. Click “OK” to close “Edit launch configuration properties” window.
  9. Back to the “Overview” window, select “Use Tomcat installation (takes control of Tomcat installation)”.
  10. Close “Overview” window after saving the changes.

## Configure Tomcat in Eclipse:

* 1. From the Project Explorer view, expand “Servers” and then “Tomcat v8.0 Server at localhost-config”.
  2. Double click the context.xml file to open it.
  3. Add a “ResourceLink” section according to the directions in Section 2.
  4. Save the context.xml file.
  5. Double click the server.xml file to open it.
  6. Add a datasource in the “GlobalNamingResources” section and some “Realm” information in the “Context” section according to the directions in Section 2.

## Configure the ESDRT project in Eclipse:

* 1. From the Project Explorer view, right click ESDRT and choose properties.
  2. Click “Deployment Assembly”.
  3. Click “Add” again from the “Deployment Assembly” window.
  4. Double click “Folder”.
  5. Select “target/ESDRT/WEB-INF/classes/reports” (or “target/classes/reports”) and click “Finish”.
  6. In “Deployment Assembly” window, double click on the “Deploy Path” value for the new folder you added and change it from “/” to “WEB-INF/classes/reports”.
  7. Click “Apply and Close”.

## Review any errors in Eclipse:

* 1. Maven errors:
     1. “Plugin execution not covered by lifecycle configuration: org.apache.maven.plugins:maven-install-plugin:2.4:install-file (execution: InstallFonts, phase: initialize)”: This is the famous issue with M2Eclipse. Since a documentation says “To get the Maven execution from within Eclipse to work you don’t have to do anything.”, go to “Windows” -> “Preferences” -> “Maven” -> “Errors/Warnings”, and change “Error” to “Warning” in option: “Plugin execution not converted by lifecycle configuration”.
     2. sdfsdf
  2. Groovy compiler errors:
     1. Go to “Windows” -> “Preferences” -> “Groovy” -> “Compiler”, switch to 2.4.
     2. In the Project Explorer view, right click the ESDRT project, select “Properties” -> “Groovy Compiler”, and select 2.4 for “Groovy compiler level for this project”.
     3. A restart of Eclipse will likely be needed,
  3. Errors with HTML and JSP can probably be ignored.
  4. Maven dependency errors:
     1. In the Project Explorer view, right click the ESDRT project, select “Properties” -> “Deployment Assembly”, examine the source list and ensure that there are no duplicate libraries to assemble to the same deployment path.

## Start up Tomcat in Eclipse:

* 1. From the “Servers” tab, right click on the server you setup and choose “Add & Remove”.
  2. Add “ESDRT” to the server and click “Finish”.

**NOTE**: If you have an error message telling you that a context already exists, please go to the context.xml file and delete the one created in this step.

## Run the ESDRT application in Eclipse:

* 1. From the “Servers” tab, click the “Restart server” button.
  2. Open a browser and enter <http://localhost:8080/ESDRT/> in the address bar, you should see the ESDRT home page.

## Rebuild with Maven:

* 1. If you make some changes to the Java code, the JasperReports templates or other code, and you want to rebuild the application, right click the ESDRT project, select “Maven Build”, enter “clean install” in the “Goals” field and click “Run”.
  2. If the build is successful, follow the directions in Step 10 and run the application.

**NOTE**: If for whatever reasons, you could not see the changes in your new compiled code, try right clicking the Tomcat server in the “Servers” tab and select “Clean…” or “Publish” or both. **It seems that for now, every time we make changes to the JasperReports templates and do a Maven clean install, we need to manually publish to the Tomcat server. I don’t have time to figure out why!**

# Section 4: Other Notes:

* Gradle (from groovy language) may be used in future for configuration/build.
* Groovy is used here (cotlin (sp?) may be used later).
* Maven build Creates ESDRT.WAR (this is what gets copied to tomcat server).
* Ftp ESDRT.WAR to SSC delivery location for deployment.