**Framework Software Installation Guide**

**Java v8:**

1. Download JAVA JDK exe file from below URL for windows (x86 or x64).

(<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>)

1. Install Java JDK.
2. In the search field type in – *view advanced system settings*
3. Click on the match on top of the list
4. Set JAVA\_HOME Environment variable
   1. In System Properties dialog, select Advanced tab and clicks on the Environment Variables... button.
   2. Under “System variables” click the “New…” button and enter JAVA\_HOME as “Variable name” and the path to your Java JDK directory (e.g. C:\Program Files\Java\jdk1.8.x\_xxx) under “Variable value”

## Update System PATH

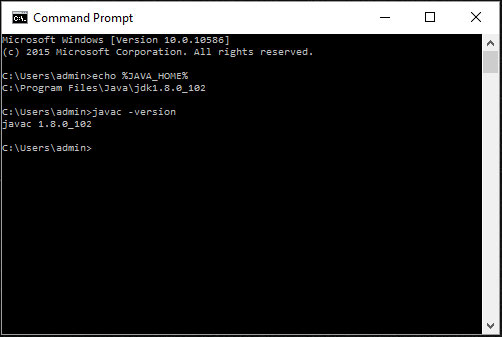
1. In “Environment Variables” window under “System variables” select Path
2. Click on “Edit…”
3. In “Edit environment variable” window click “New”
4. Type in  %JAVA\_HOME%\bin
5. Restart System

## Test your configuration

* 1. Open a new command prompt and type in:

**javac -version**

this will print out the version of the java compiler if the Path variable is set correctly or *“javac is not recognized as an internal or external command…”* otherwise



**Apache Maven v3.6.0**

1. Download binary zip of maven step (i.e. [*apache-maven-3.6.0-bin.zip*](https://archive.apache.org/dist/maven/maven-3/3.6.0/binaries/apache-maven-3.6.0-bin.zip) ) from below URL
   1. (<https://archive.apache.org/dist/maven/maven-3/3.6.0/binaries/>)
2. Extract the maven zip file in C: drive
3. In the search field type in – *view advanced system settings*
4. Click on the match on top of the list
5. Set JAVA\_HOME Environment variable
   1. In System Properties dialog, select Advanced tab and clicks on the Environment Variables... button.
   2. Under “System variables” click the “New…” button and enter MAVEN\_HOME as “Variable name” and the path to your Java JDK directory (e.g. C:\apache-maven-3.x.x) under “Variable value”

## Update System PATH

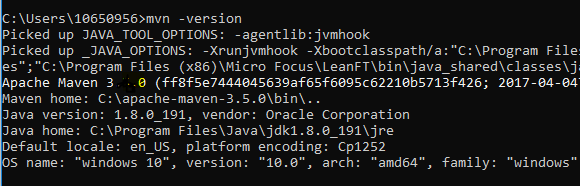
1. In “Environment Variables” window under “System variables” select Path
2. Click on “Edit…”
3. In “Edit environment variable” window click “New”
4. Type in  %MAVEN\_HOME%\bin
5. Restart System

## Test your configuration

* 1. Open a new command prompt and type in:

**mvn -version**

this will print out the version of the maven version info if the Path variable is set correctly or *“mvn is not recognized as an internal or external command…”* otherwise



**Eclipse:**

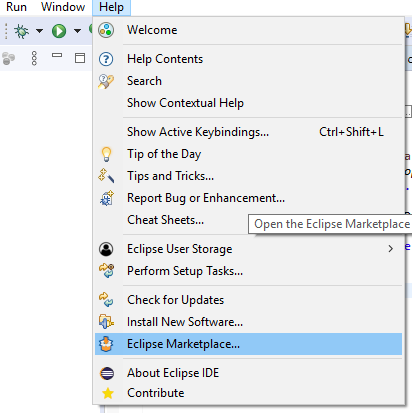
1. Download Eclipse step up from below URL:

( <https://www.eclipse.org/downloads/> )

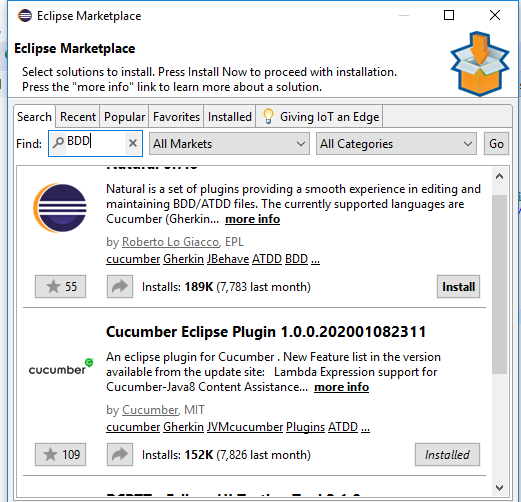
1. Initiate the setup and let it install.

**Cucumber Eclipse Plugin:**

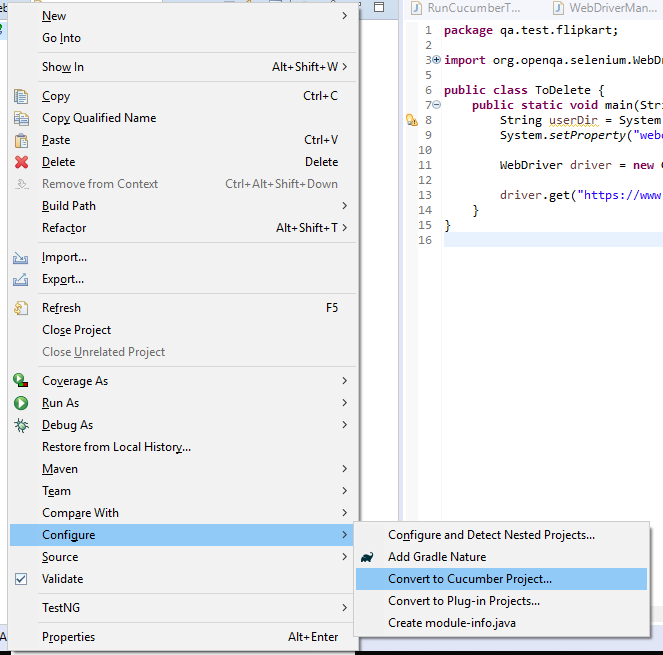
1. Open Eclipse > Help > Eclipse Marketplace…



1. Search for BDD in Market Place

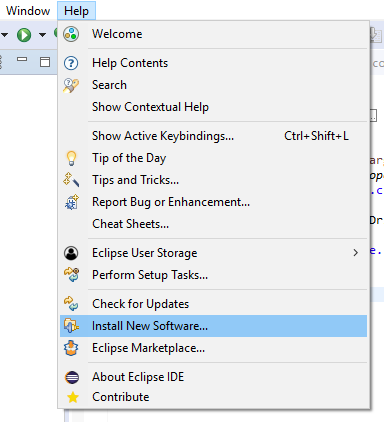


1. Click on Install for Cucumber Eclipse Plugin
2. Restart Eclipse
3. Right click on Project > Configuration > Convert to Cucumber Project…

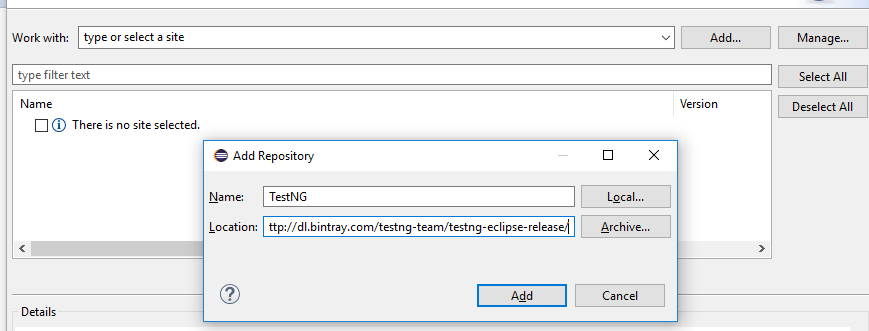


**TestNG:**

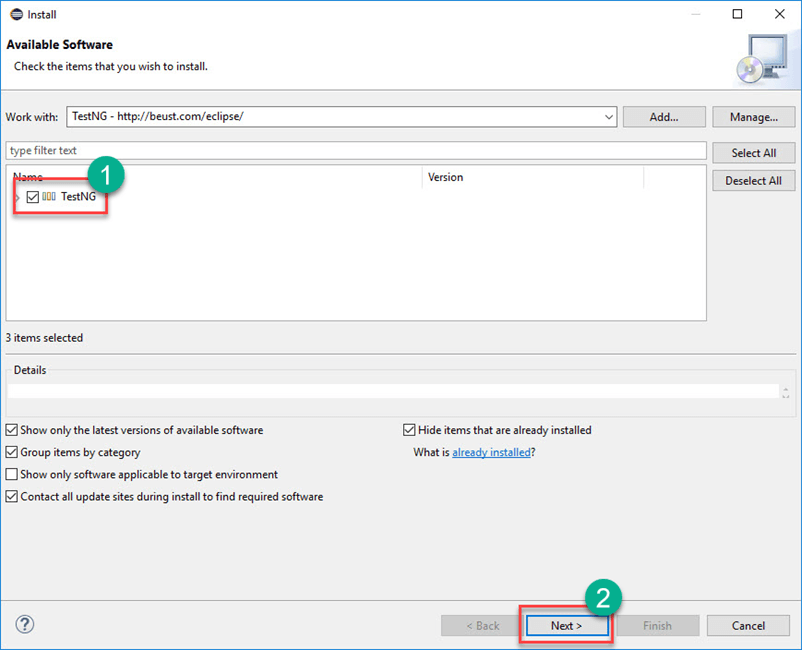
1. Open Eclipse > Install New Software



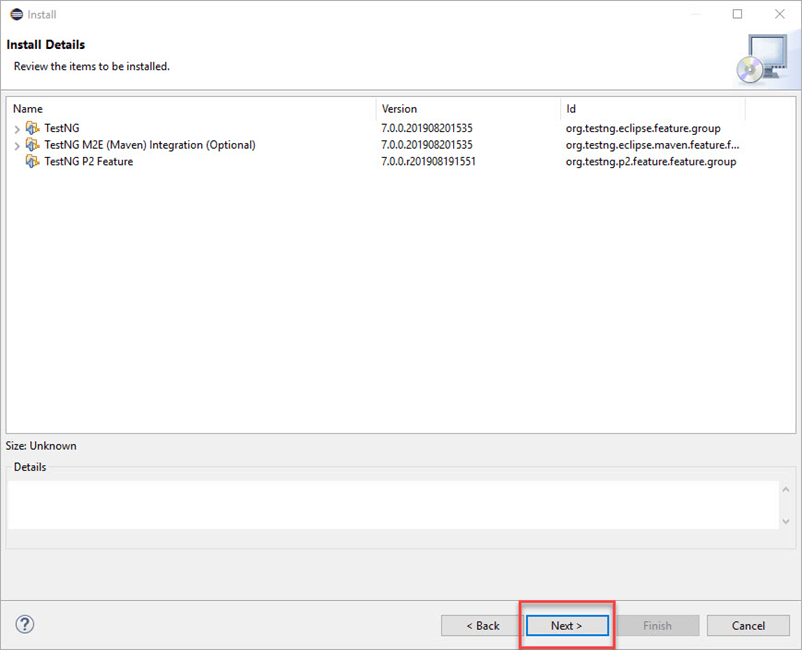
1. Click on Add on in Install dialog
2. Enter TestNG in Name Text Field
3. Enter http://dl.bintray.com/testng-team/testng-eclipse-release/ in the Work with box
4. Click on Add



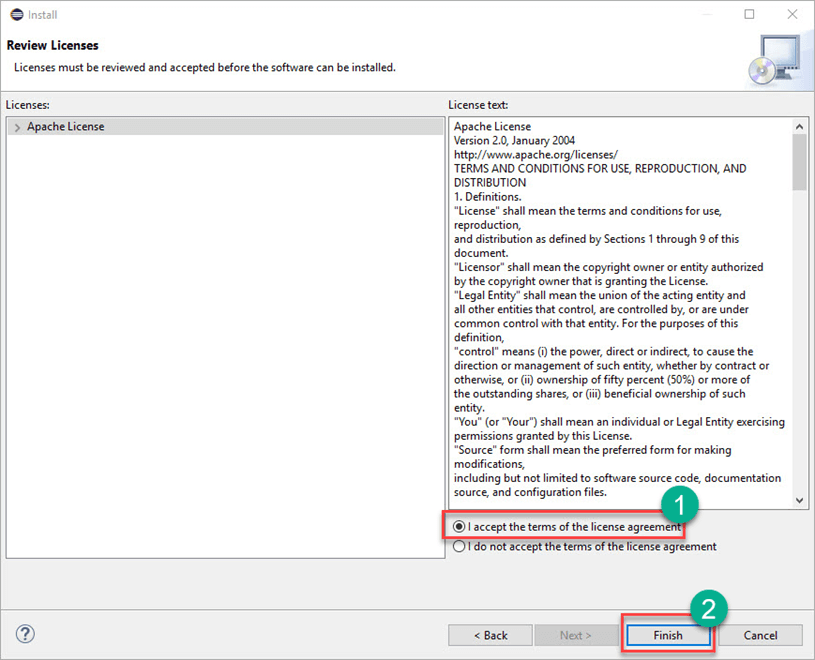
1. Select TestNG and click on Next



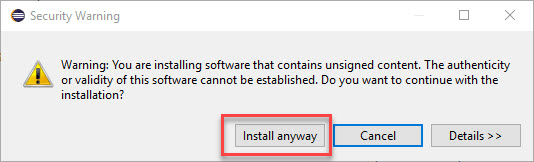
1. It will give you a review of Items to be installed. Click on Next.



1. Select the radio button "I accept the terms of the license agreement" and click on Finish.

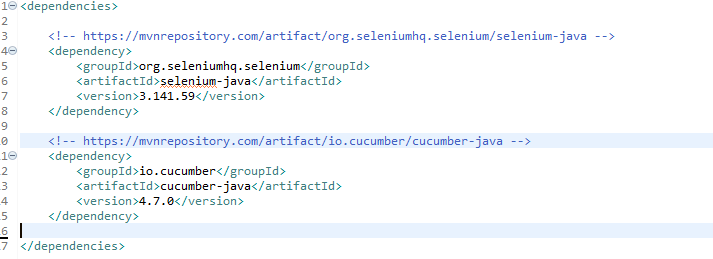


1. If you encounter a Security warning, just click "Install Anyway".

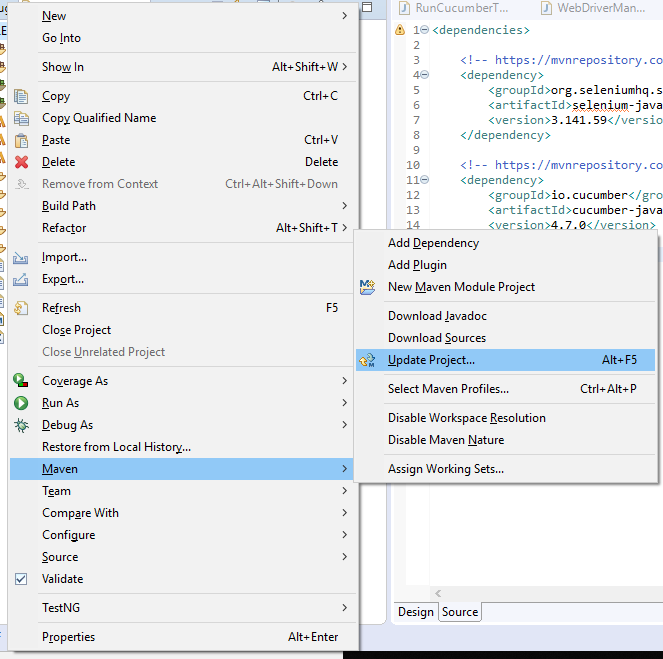


**Maven Dependencies:**

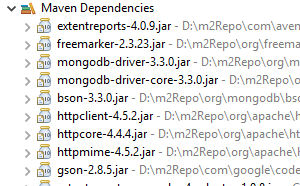
1. Open pom.xml of project and mentions all the dependencies under <dependencies></ dependencies> tag and save the file.



1. Right click on Project > Maven > Update Project…



1. Let the Project build once done dependencies will get added under Maven Dependencies library



**Extent Report Config:**

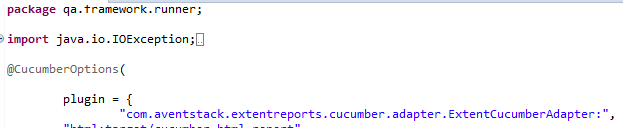
1. Navigate to below url to download ‘extent.properties’ file

( <https://github.com/grasshopper7/cuke4-extent-adapter-report/blob/master/cuke-extent-adapter-report/src/test/resources/extent.properties> )

1. Navigate to below url to download ‘extent-config.xml’

( <https://github.com/grasshopper7/cuke4-extent-adapter-report/blob/master/cuke-extent-adapter-report/src/test/resources/extent-config.xml> )

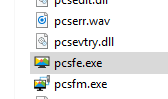
1. Put ‘extent.properties’ and ‘extent-config.xml’ under ‘src/text/resources’ folder only.
2. In Runner file under CucumberOptions mention plugin as "com.aventstack.extentreports.cucumber.adapter.ExtentCucumberAdapter:"

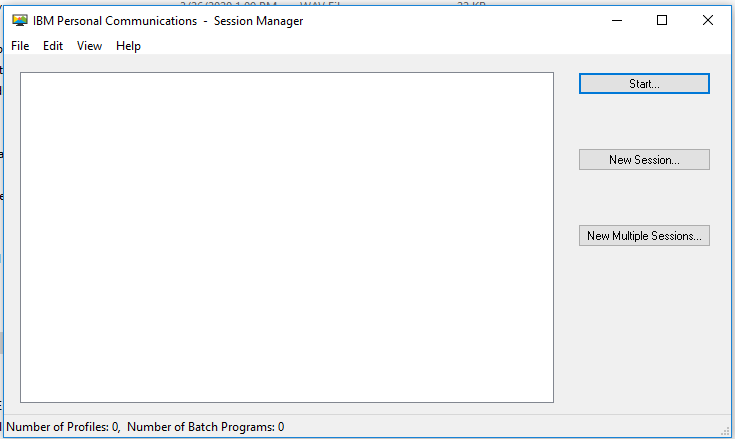


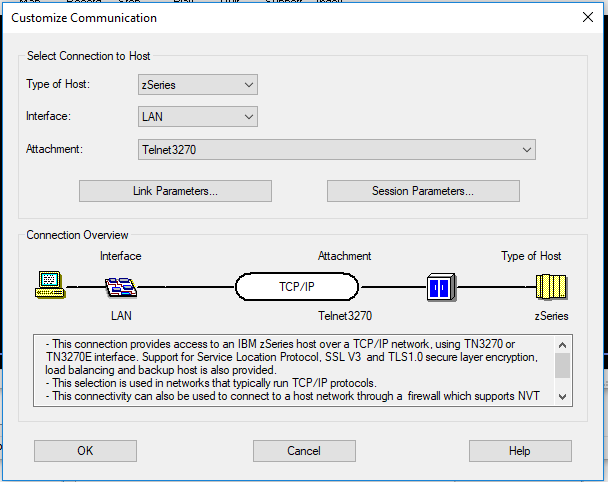
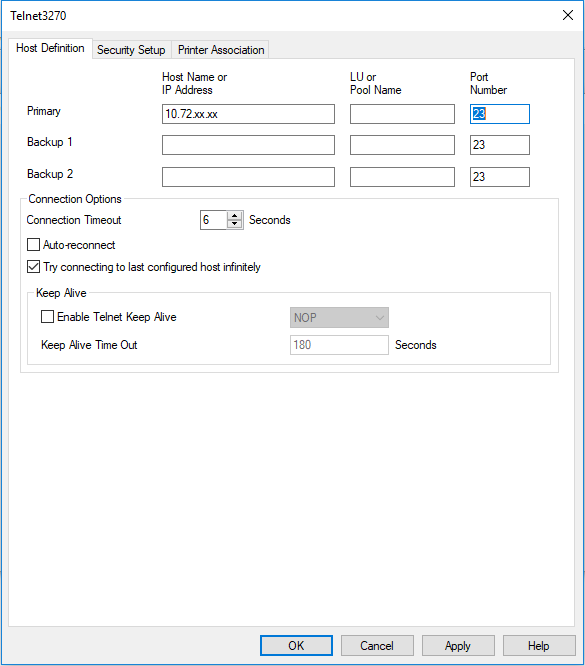
1. Reference: <http://grasshopper.tech/824/>

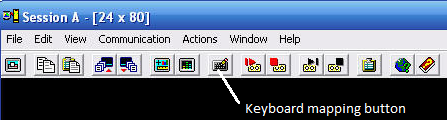
**IBM PCOMM Configuration:**

1. Install **IBM PCOMM v12.0** (restart system if required).
2. Navigate to installation folder (default folder is ‘C:\Program Files\IBM\Personal Communications’) and start IBM PCOMM by double clicking on ‘pcsfe.exe’.





1. Click on ‘New Session’ which will open ‘Customize Communication’ window. 
2. Click on ‘Link Parameters…’ button. Here user need to specify IP address and port to communicate with respective server. (project specific setting)
3. Save the changes made.
4. Click on File > Save, save the session in ‘src/test/resources/mf/launcher’ folder under project. It will be a ‘.ws’ extension file.
5. Click on ‘Keyboard Function’ button shown in below image. ‘Customize Keyboard -US’ window will appear.

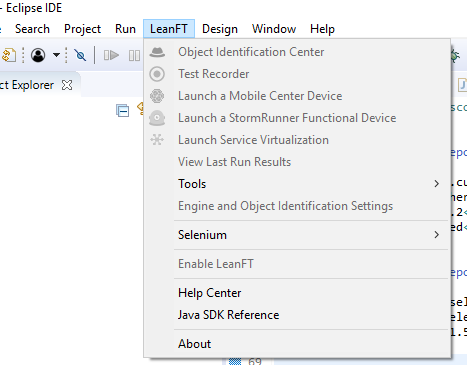


1. Click on File > Save as > in ‘src/test/resources/mf/launcher’ folder under project. It will be a ‘.kmp’ extension file.

**LeanFT Installation and Configuration:**

Take IT personal help to install LeanFT v14.x.

1. **Integration of LeanFT with Eclipse**
2. Navigate to LeanFT droppings folder. For example: ‘C:\Program Files (x86)\Micro Focus\LeanFT\IDE\Eclipse\dropins’ copy all the jars shown in this folder.
3. Navigate to Eclipse installation directory dropins folder. For example: ‘C:\Users\xxxxxx\eclipse\java-2018-12\eclipse\dropins’.
4. Paste all the jars here.
5. Launch Eclipse. User will find ‘LeanFT’ menu tab added. This proves that LeanFT is added integrated to Eclipse.



1. **Installation of LeanFT jar to Maven local Repository (.m2)**
2. Navigate to LeanFT > SDK > Maven folder. For example: ‘C:\Program Files (x86)\Micro Focus\LeanFT\SDK\Maven’
3. Open the above path in Command line and run below maven command for all the jar files in the respective folder.

Syntax:

mvn install:install-file -Dfile="**<jar file name.jar>**" -DgroupId="com.hp.lft" -DartifactId="**<jar file name without version and .jar extension>**" -Dversion="**<version mentions in jar file xx.xx.xx>**" -Dpackaging="jar" -DgeneratePom=true

For Example:

mvn install:install-file -Dfile="com.hp.lft.common-14.52.0.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.common" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.report-14.52.0.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.report" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.report-14.52.0-javadoc.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.report-javadoc" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.reportbuilder-14.52.0.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.reportbuilder" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.sdk-14.52.0.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.sdk" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.unittesting-14.52.0.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.unittesting" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.unittesting-14.52.0-javadoc.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.unittesting-javadoc" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.verifications-14.52.0.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.verifications" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

mvn install:install-file -Dfile="com.hp.lft.verifications-14.52.0-javadoc.jar" -DgroupId="com.hp.lft" -DartifactId="com.hp.lft.verifications-javadoc" -Dversion="14.52.0" -Dpackaging="jar" -DgeneratePom=true

1. Once all the commands are executed successfully verify the same in .m2 folder.
2. **Integrating LeanFT jars to Project**
3. Open your Project in Eclipse
4. Open pom.xml file
5. Create path variable for all the LeanFT jar installed in previous steps

<properties>

<sdk.path>\\192.xx.xx.xx\Users\xxxx\.m2\repository\com\hp\lft\com.hp.lft.sdk\14.52.0\com.hp.lft.sdk-14.52.0.jar</sdk.path>

<report.path>\\192.xx.xx.xx\Users\xxxx\.m2\repository\com\hp\lft\com.hp.lft.sdk\14.52.0\com.hp.lft.report-14.52.0.jar</report.path>

<reportbuilder.path>\\192.xx.xx.xx\Users\xxxx\.m2\repository\com\hp\lft\com.hp.lft.sdk\14.52.0\com.hp.lft.reportbuilder-14.52.0.jar</reportbuilder.path>

<unittesting.path>\\192.xx.xx.xx\Users\xxxx\.m2\repository\com\hp\lft\com.hp.lft.sdk\14.52.0\com.hp.lft.unittesting-14.52.0.jar</unittesting.path>

<verifications.path>\\192.xx.xx.xx\Users\xxxx\.m2\repository\com\hp\lft\com.hp.lft.sdk\14.52.0\com.hp.lft.verifications-14.52.0.jar</verifications.path>

<common.path>\\192.xx.xx.xx\Users\xxxx\.m2\repository\com\hp\lft\com.hp.lft.sdk\14.52.0\com.hp.lft.common-14.52.0.jar</common.path>

</properties>

1. Mention all the jars dependencies under <dependencies> tag as shown below:

<dependency>

<groupId>com.hp.lft</groupId>

<artifactId>com.hp.lft.sdk</artifactId>

<version>14.52.0</version>

<scope>system</scope>

<systemPath>${sdk.path}</systemPath>

</dependency>

<dependency>

<groupId>com.hp.lft</groupId>

<artifactId>com.hp.lft.report</artifactId>

<version>14.52.0</version>

<scope>system</scope>

<systemPath>${report.path}</systemPath>

</dependency>

<dependency>

<groupId>com.hp.lft</groupId>

<artifactId>com.hp.lft.reportbuilder</artifactId>

<version>14.52.0</version>

<scope>system</scope>

<systemPath>${reportbuilder.path}</systemPath>

</dependency>

<dependency>

<groupId>com.hp.lft</groupId>

<artifactId>com.hp.lft.unittesting</artifactId>

<version>14.52.0</version>

<scope>system</scope>

<systemPath>${unittesting.path}</systemPath>

</dependency>

<dependency>

<groupId>com.hp.lft</groupId>

<artifactId>com.hp.lft.verifications</artifactId>

<version>14.52.0</version>

<scope>system</scope>

<systemPath>${verifications.path}</systemPath>

</dependency>

<dependency>

<groupId>com.hp.lft</groupId>

<artifactId>com.hp.lft.common</artifactId>

<version>14.52.0</version>

<scope>system</scope>

<systemPath>${common.path}</systemPath>

</dependency>

1. LeanFT have an issue that SDK.init() does stop running even after execution is finished. To resolve this issue an additional ‘glassfish.tyrus’ dependency is required.

<dependency>

<groupId>org.glassfish.tyrus</groupId>

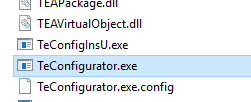
<artifactId>tyrus-container-grizzly-client</artifactId>

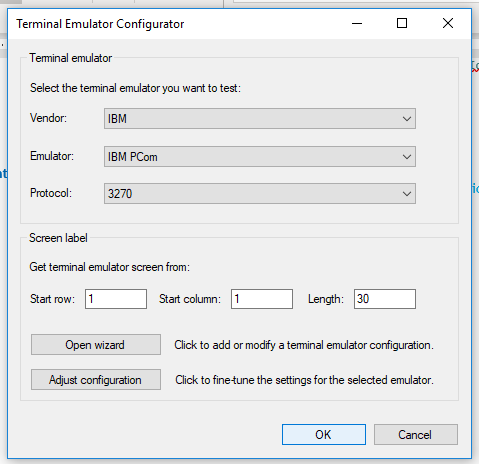
<version>1.13</version>

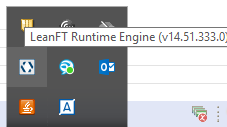
<scope>test</scope>

</dependency>

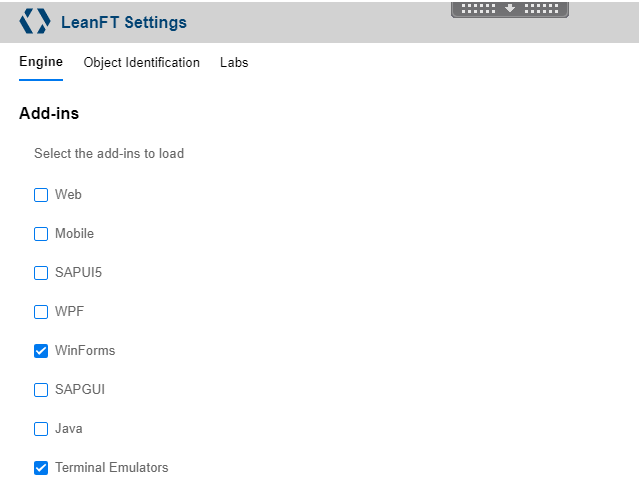
1. Save pom.xml, LeanFT jars will get auto configured in class-path.
2. **Integrating LeanFT with IBM PCOMM**
3. Navigate to LeanFT > bin folder. For example: ‘C:\Program Files (x86)\Micro Focus\LeanFT\bin’.
4. Open terminal configuration file ‘TeConfiguration.exe’. ‘Terminal Emulator Configuration’ window will open.



1. Select ‘Vendor’ as ‘IBM’.
2. Select ‘Emulator’ as ‘IBM Pcom’
3. Select ‘Protocol’ as ‘3270’ 
4. Save the settings.
5. **Starting LeanFT Engine**
6. Run ‘LeanFT Runtime’ from window’s start menu.
7. Once the LeanFT engine is started, click on task bar ‘LeanFT icon’ and select ‘Settings’



1. ‘LeanFT Settings’ window will appear. Select ‘Terminal Emulators’ option and save the changes made.



1. Your LeanFT engine is configured and started.