**Distributing common framework repository to individual application respositories through maven distribution management features**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Date | Remarks |
| 0.1 | Atul Gogte | 19th Nov 2016 | initial draft version |
|  |  |  |  |

**Step 1 - Adding your machine’s SSH public key to external bitbucket repository**

We have created a **external private** repository on bitbucket.org called **TMOAuto\_Framework\_Repo**. In order to deploy to this repopsitory you must do 2 things

1. Add your machines **SSH Public Key** under **TMOAuto\_Framework\_Repo** in bitbucket.org under username P2248342 through manage account option. (please send me you SSH key file and I will get it added)
2. Add a file named **settings.xml** under your **<user\_home>/.m2** folder as well as **<M2\_HOME>/conf** folder with following content. Replace only the yellow highlighted section with the path of your ssh private key location on your machine.

<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0

https://maven.apache.org/xsd/settings-1.0.0.xsd">

<servers>

<server>

<id>Framework\_Public</id>

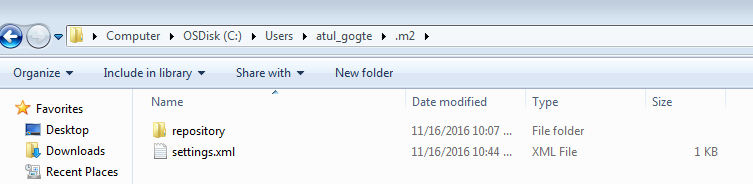
<username>P2248342</username>

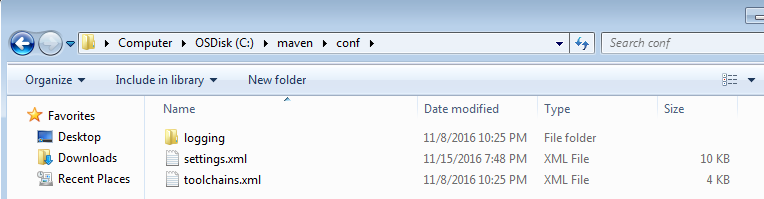
<privateKey>C:/Users/atul\_gogte/.ssh/id\_rsa</privateKey>

</server>

</servers>

</settings>





**Step 2 - Updating Framework pom.xml**

Please add the folling sections in your framework pom.xml

*<!--defining the naming convention for framework repository --****>***

<groupId>com.chtr.tmoauto</groupId>

<artifactId>Framework</artifactId>

<version>1.0</version>

*<!--Please add wagon git plugin to your plugin respository section to push the artifacts into remote repository --****>***

<pluginRepositories>

<pluginRepository>

<id>synergian-repo</id>

<url>https://raw.github.com/synergian/wagon-git/releases</url>

</pluginRepository>

</pluginRepositories>

*<!--We will be using distribution management to define where we want to push the artifact to i.e external repository -->*

<distributionManagement>

<repository>

<id>Framework\_Public</id>

<name>BitBucket Framework Public Repository</name>

<url>git:master://git@bitbucket.org:P2248342/tmoauto\_framework\_repo.git</url>

</repository>

</distributionManagement>

*<!--We will be adding wagon git extension under the build section -->*

<build>

<extensions>

<extension>

<groupId>ar.com.synergian</groupId>

<artifactId>wagon-git</artifactId>

<version>0.2.5</version>

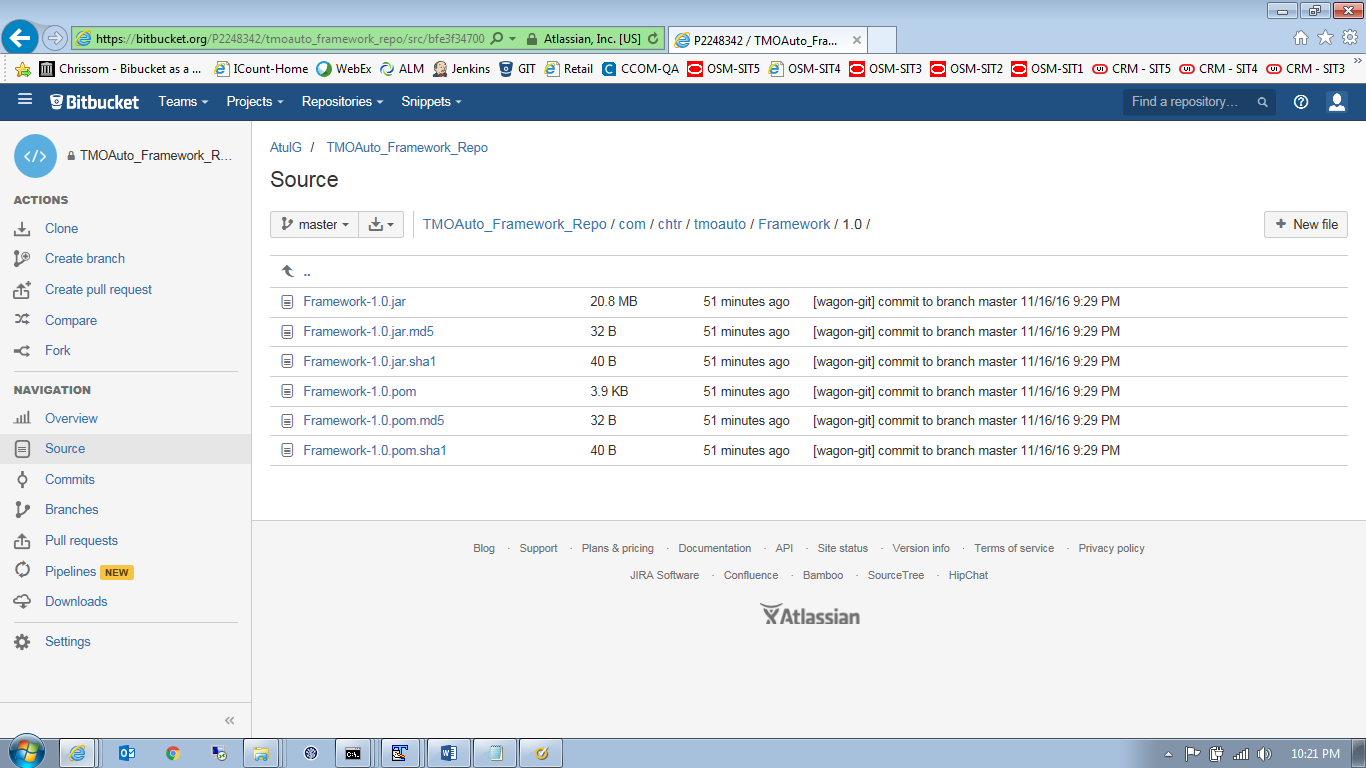
</extension>

</extensions>

</build>

After adding these sections to framework repository pom.xml, you can deploy the framework artifact to remote public repository using **mvn clean deploy** command.

What it does is upload your project artifacts to external repository as shown below.



**Application level pom.xml**

Add following sections in your application level pom.xml allowing your application to refer to third party external repository (along with maven central repopsitory) to access the framework functions.

<repositories>

<!-- This allows application pom to refer to our public framework repository -->

<repository>

<id>Framework\_Public</id>

<name>BitBucket Framework Public Repository</name>

<url>https://bitbucket.org/P2248342/tmoauto\_framework\_repo/raw/master/</url>

<layout>default</layout>

</repository>

<!-- Override and disable Super POM Maven Central repository -->

<repository>

<id>central</id>

<name>Maven Repository Switchboard</name>

<url>http://repo1.maven.org/maven2</url>

<releases>

<enabled>true</enabled>

</releases>

<snapshots>

<enabled>false</enabled>

</snapshots>

</repository>

</repositories>

<!-- Add following dependency to framework functions from public repository -->

<dependency>

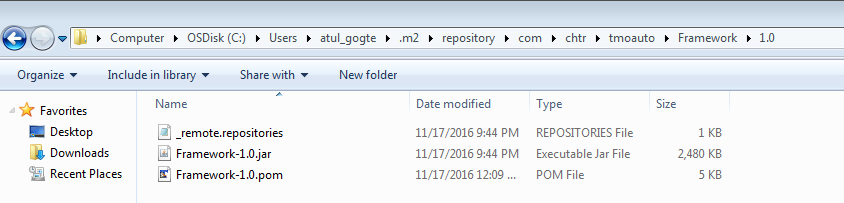
<groupId>com.chtr.tmoauto</groupId>

<artifactId>Framework</artifactId>

<version>2.0</version>

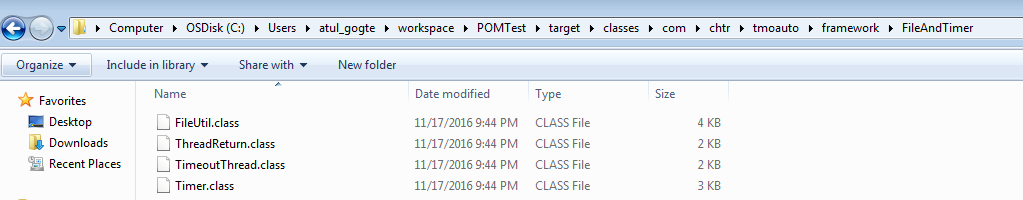
</dependency>

Now run the **mvn clean package** command on your application project. This will **downlaod the framework.jar from external repository to your local .m2 reository**



It will also **unpack the downaloaded framework.jar into your application project’s target/classes folder**. You will see something like below…

[INFO] Unpacking C:\Users\atul\_gogte\.m2\repository\com\chtr\tmoauto\Framework\1.0\Framework-1.0.jar to C:\Users\atul\_gogte\workspace\POMTest\target\classes with includes "" and excludes ""



Now you can reference the common framework functions insite your application project inside your IDE

