**Mule Migration 3.6.2 to 3.8.2 Version (designed by Nikhil Sharma & Ruchi)**

1. In Pom.xml do the following:

* Remove mule-ee-license test dependency
* Remove service-registry-esb property and dependency
* In properties set mule-version to 3.8.2
* Spring Namespace version 3.1.1 Release
* JDKName-JavaSE-1.8
* JDK Version-1.8

1. In EFT-flows-config.xml do the following:

* Add the Service Registry Namespace & Schema**:**

--- xmlns:msr=”<http://statefarm.com/schema/msr>”

--- xsi:schemaLocation="http://statefarm.com/schema/msr http://statefarm.com/schema/msr/msr.xsd”

* Add the following as HTTP is deprecated:

--- xmlns:sfhttp="http://statefarm.com/schema/mule/http"

--- xsi:schemaLocation=[http://statefarm.com/schema/mule/http http://statefarm.com/schema/mule/http/current/sf-mule-http.xsd](http://statefarm.com/schema/mule/http%20http://statefarm.com/schema/mule/http/current/sf-mule-http.xsd)

* **In <flow> tag now <http:inbound-endpoint> is replaced by <sfhttp:listener> and**

**<http:outbound-endpoint> is replaced by <sfhttp:request> as follows**:

<flow name=””>

<sfhttp:listener config-ref="msr-listener-httpConnector" path="testPath" doc:name="HTTP"/>

<msr:resolve-endpoint serviceName="TestService" />

<sfhttp:request config-ref="msr-request-httpConnector" path="" method="POST" doc:name="HTTP">

<sfhttp:success-status-code-validator values="200,500"/>

</sfhttp:request>

<transformer ref="statusTransformer" />

</flow>

1. In mule-config.xml do the following:

* Add in spring bean import for service registry Setup for runtime use:

**<**spring:import resource="classpath:msr-config.xml" />

<spring:import resource="classpath:msr-config/msr-config-https-proxied-config.xml" />

1. In endpoints.xml do the following changes:

* Replace n4 with n5 and v4 with v5:

<ns5:getEndpointsResponse xmlns:ns5="http://v5.service.registrymanager.sf/"></n5>

1. -Dmule.srstatus startup argument not required in 3.8.2
2. In functional-test-config.xml replace http namespace and schema with sfhttp
3. To Setup service registry for runtime follow the steps given in:

<https://techguide.test.statefarm.com/index.php/Mule_ESB_3.8.2_Guide#Customizing_the_Mule_Runtime_for_Local_Development>

### Customizing the Mule Runtime for Local Development

**Alert: This step cannot be skipped when developing for Mule 3.8.2. When updates are announced to the service registry jar, it is highly recommended to update your workstation installation by pulling the latest version of the studio-runtime-updater (see step #2). The current version of the studio-runtime-updater is 3.8.2.1.2**

1. The service registry jar should no longer be set as a compile time dependency in your Mule ESB application. We have decided to start making that dependency part of the Mule ESB Runtime. In the same way that your pom manages Mule dependencies as provided by the Mule Runtime, you will mark the Service Registry jar that you intend to use as provided in your pom. Next you will want to follow the steps to install the service registry into your local runtimes.
2. Log into enterprise Artifactory instance at <http://lt00es000000000.opr.statefarm.org/artifactory>
3. Search for "studio-runtime-updater" and find the version of the Mule ESB Runtime that you are interested in developing for, according to the current version called out in [the Mule Blueprint manifest](https://techguide.test.statefarm.com/index.php/Mule_Blueprint#Version_Table)
4. Click on the results, or hover over and click "download".
5. Move the downloaded POM to {STS HOME DIRECTORY}\sts-3.8.0.RELEASE\plugins\org.mule.tooling.server.3.8.2.ee\_{some version}\mule\lib\user
6. Rename the studio-runtime-updater pom to pom.xml
7. The POM contains a placeholder called "ltsServer". If your .m2/settings.xml file doesn't have a server with credentials defined named according to your LTS server, open the POM and edit it as necessary to provide a reference to your repository and snapshot repository.
8. Ensure STS is not currently running, otherwise the maven clean process that will run during jar update to follow will be unable to exeucte becuase STS is holding on reference to the jars/directory.
9. Open a command prompt in the {STS HOME DIRECTORY}\sts-3.8.0.RELEASE\plugins\org.mule.tooling.server.3.8.2.ee\_{some version}\mule\lib\user directory
10. Run the following command:  
    mvn clean dependency:copy-dependencies -DltsServer=lt00asc0 [the -DltsServer parameter is your LTS artifactory server, the one in the example is for auto]
11. A number of jars should be added, including the version of the service registry referenced in the POM of your project, as well as any known patches that have been applied to the platform Mule servers.
12. Start STS. The JAR's that are loaded should now appear in the "Mule Server 3.8.2 EE" server runtime brought in to your project.