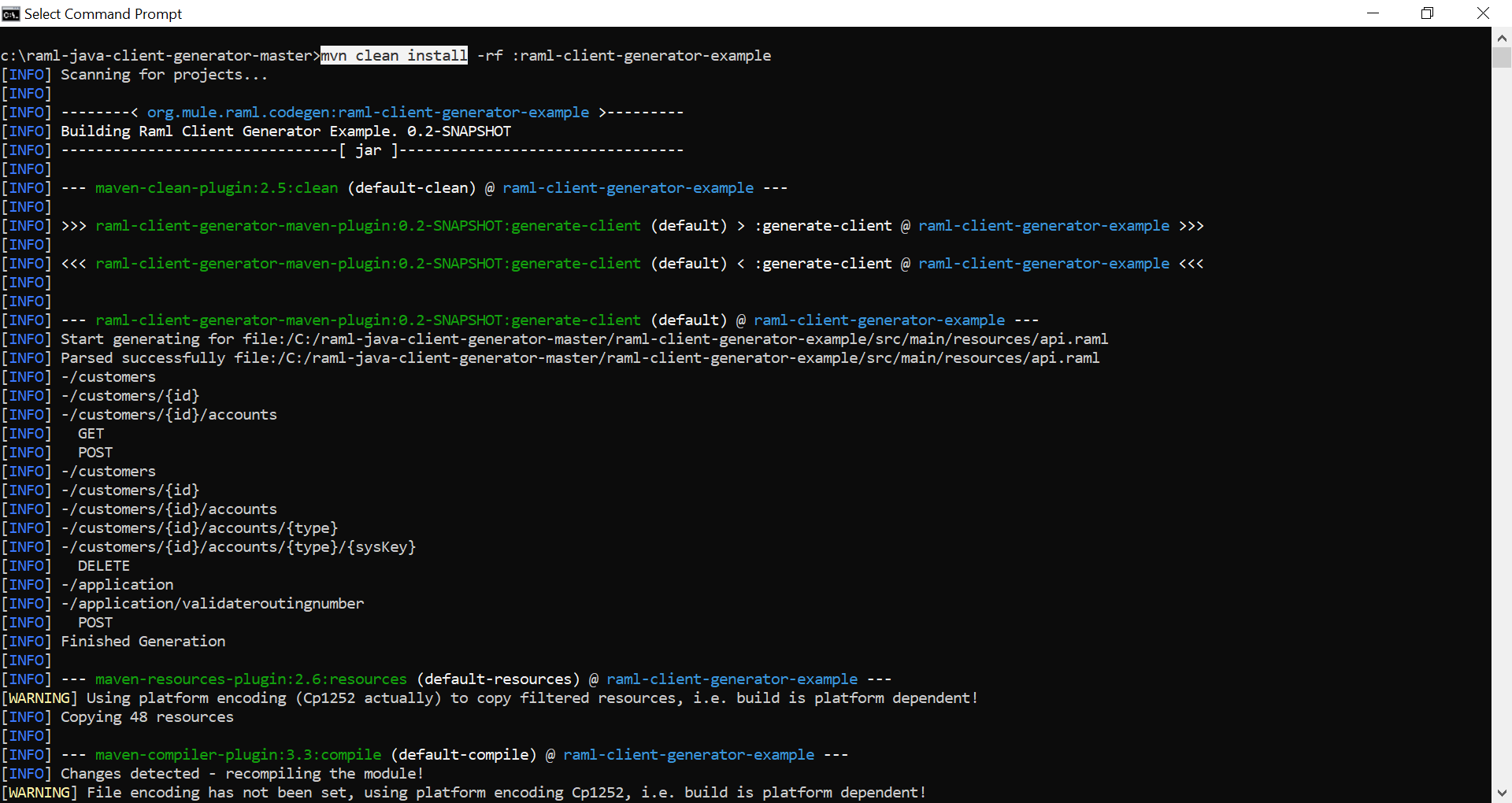
**RAML Java Client Generator**

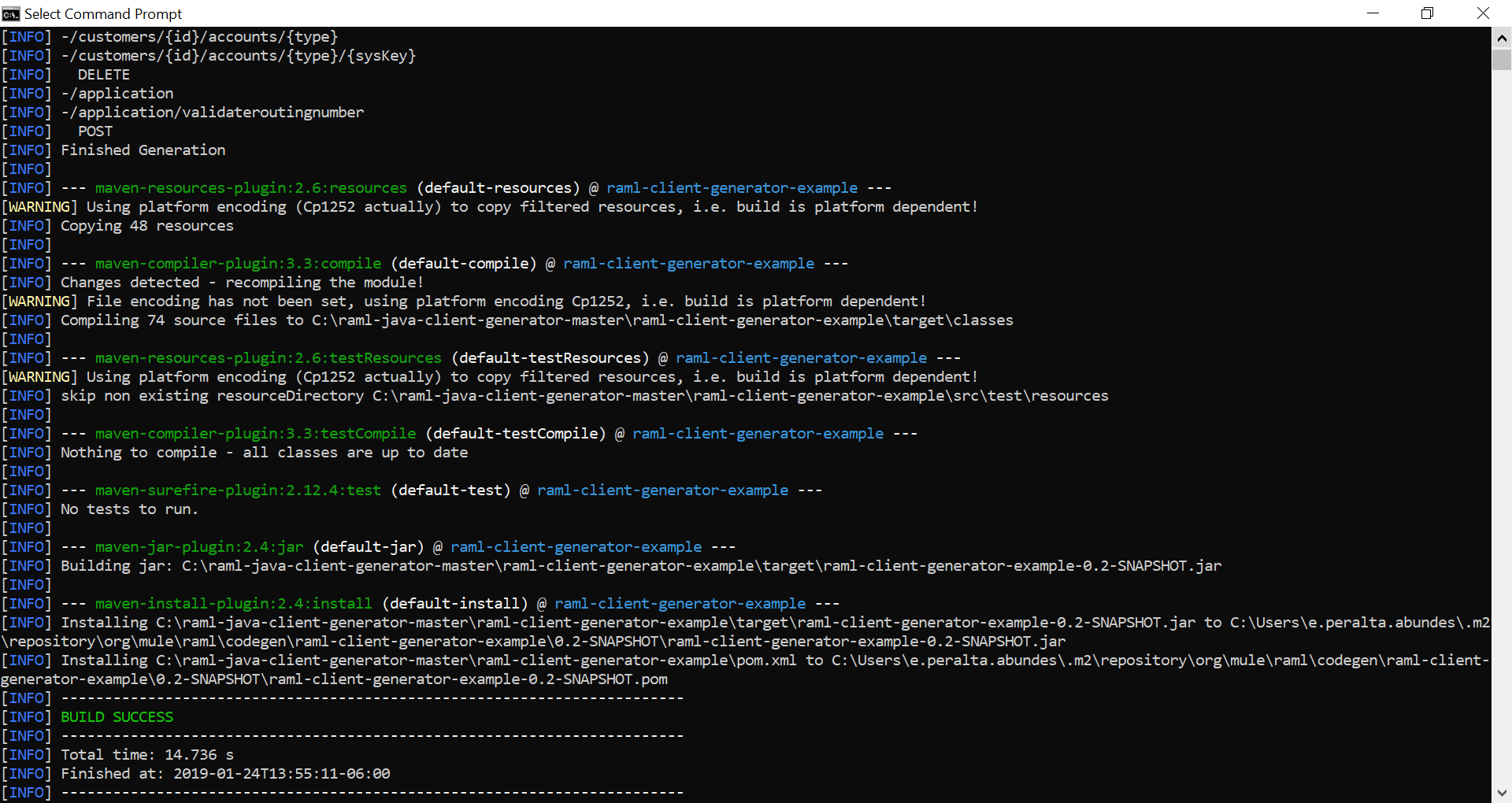
Requisites:

* Java SDK 1.8
* Maven
* eclipse
* Get the ALIP API RAML project from Mulesoft Platform Exchange
  + Including the api.raml file, examples and schemas

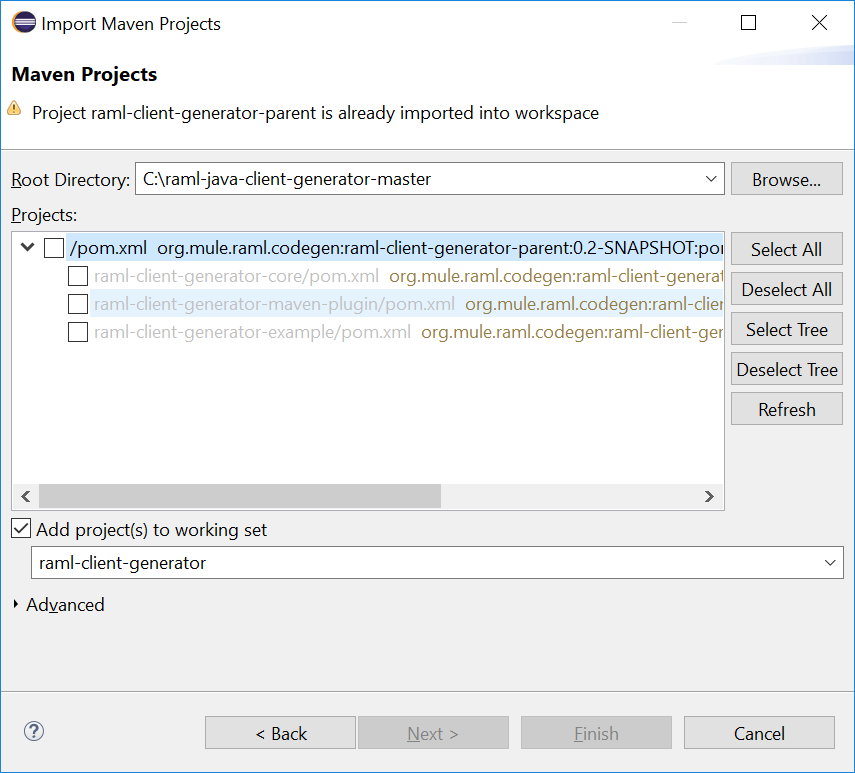
Steps:

1. Download the raml-java-client-generator-master.zip file from
   1. <https://github.com/mulesoft-labs/raml-java-client-generator>
2. Unzip the raml-java-client-generator-master.zip file
3. Open File Explorer and go to project folder ‘resources’
   1. C:\raml-java-client-generator-master\raml-client-generator-example\src\main\resources
   2. Replace the api.raml with your own file from alip
      1. Must be named as ‘api.raml’
   3. Include the ‘examples’ folder
   4. Include the ‘schemas’ folder
4. Open a windows command prompt
5. Run maven build from master
   1. c:\raml-java-client-generator-master>mvn clean install

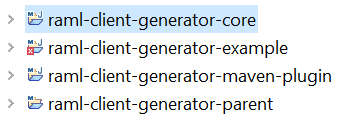




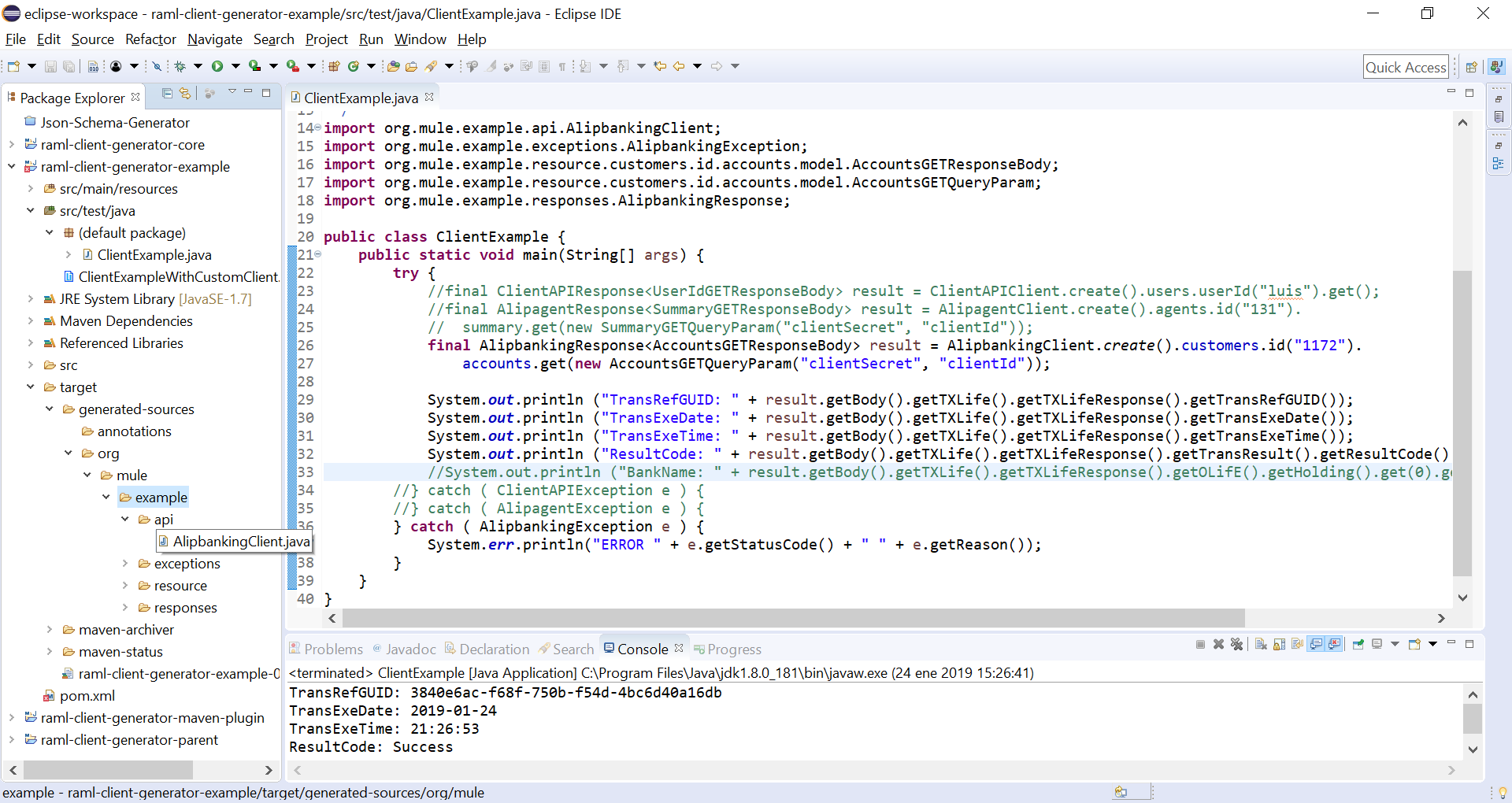
1. Confirm the maven build success
   1. in case of errors, verify and fix the api.raml, examples or schemas issues
2. Import the master project into eclipse as maven
   1. Import Existing Maven Projects
   2. Root directory: C:\raml-java-client-generator-master
   3. Select the pom.xml file from master



* 1. Verify the imported projects: core, example, maven-plugin and parent



1. Verify the java client files generated from your api.raml
   1. C:\raml-java-client-generator-master\raml-client-generator-example\target\generated-sources\org\mule\example
      1. api
      2. exceptions
      3. resource
      4. responses
2. Update the ClientExample.java to use the new generated classes to make the service call
   1. import the new java classes
   2. use the client to make a service call
   3. print the response to verify
3. Run the ClientExample.java as a java application



**Troubleshoots**

**Property Order -** JsonPropertyOrder was wrong generated

First, removed the generated jar file for raml-cient-generator-example.

Second, included the target/generated-sources in the build path.

Third, updated the Party.java @JsonPropertyOrder as corresponded.

So, now the ClientExample.java is resolving/using  the Party.java class updated manually,

instead of resolving/using the jar classes.

-------------------------------------------------------------------------------------------------------------------------

**MODEL classes** created during the RAML Client generator build are wrongly created which change the sequence to the tags e.g.  (TransExeDate and TransExeTime)

The schema file contains the **TransExeDate** and **TransExeTime**, both defined as required string.

The example file doesn´t contains  **TransExeDate** ,**TransExeTime,** so I am assuming the example file is wrong/incomplete, please confirm and fix.

The model java class generated for TXLifeRequest contains both fieds:

    @JsonProperty("TransExeDate")

    private String transExeDate;

    @JsonProperty("TransExeTime")

    private String transExeTime;

but only the get/set methods were created for **TransExeDate**

the get/set methos for **TransExeTime** were not generated,

so I think this is an issue with the raml java client generator tool,

we can manually include the missing get/set methods and try.

---------------------------------------------------------------------------------------------------------------------------------

Mismatches example /schema files on Newbusiness API definition

* Replaced occurrences from “#text” to “$” in examples/schemas
* Updated pre/postSubmit examples – missing schemas
* Commented out query params client\_id, client\_secret, because they are not supported by the web adapter service
* Included missing “TXLife” tags as required in examples/schemas

{

"$schema": "http://json-schema.org/draft-03/schema#",

"definitions": {},

"id": "http://example.com/example.json",

"properties": {

"TXLife": {

"id": "/properties/TXLife",

"properties": {

**DELETE method** doesn’t supports body request.

There is an issue with the RAML java client generated for delete method:

the DELETE method on client is not supporting the body request

Exception in thread "main" java.lang.IllegalStateException: Entity must be null for http method DELETE.

**Reference**:

<https://stackoverflow.com/questions/25229880/how-to-send-enclose-data-in-delete-request-in-jersey-client>

*From the Jersey client API, the* [*SyncInvoker*](http://docs.oracle.com/javaee/7/api/javax/ws/rs/client/SyncInvoker.html) *class doesn't support a delete method with entity body as its argument. So, I can only use either POST or PUT to send the data to the server like the following (but not for DELETE):*