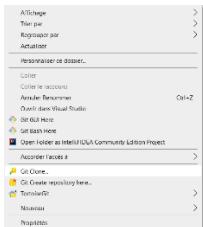


1/Installation

The source code is available on the UNIGE gitlab repository: to retrieve it, run a gitclone command on the following repository address:

<https://github.com/philippe-glass/energy/tree/master>



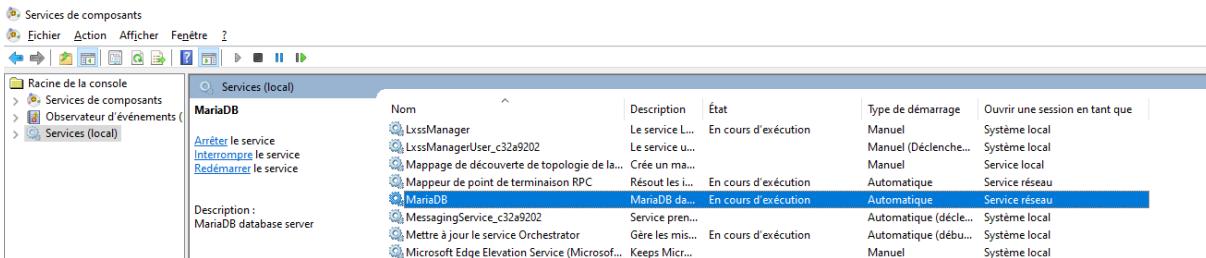
1.2/Coordination platform database

Agents use a relational database to store event data, node state, and number of observations that generate Markov chain transition matrices.

1.2.1/Server installation

- Download the last version of MariaDB installer from: <https://mariadb.org/download/>
- Launch the installer file (for example for 10.6.4 version: mariadb-10.6.4-winx64.msi)
- During the installation process, MariaDB can be configured as a Windows service started automatically.

(windows service)

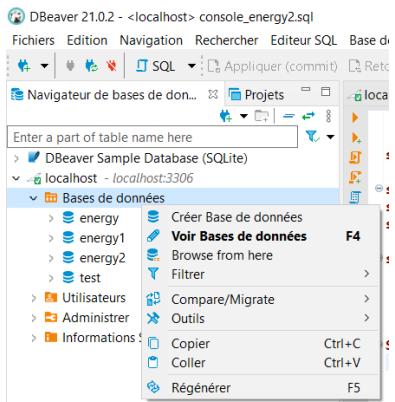


1.2.2/Client GUI installation

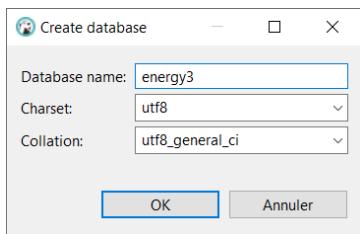
- Download the last version of DBEaver installer from: <https://dbeaver.io/download/>
- Install DBEaver (for example: dbeaver-ce-21.0.2-x86_64-setup.exe)

1.2.3/Database creation

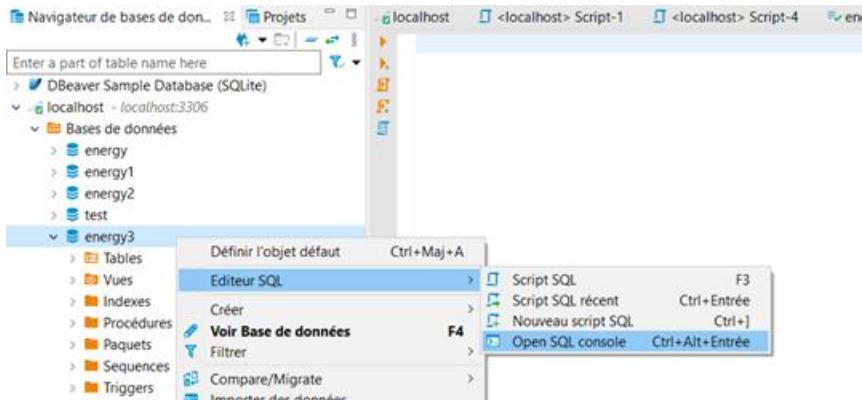
- Launch DBeaver
- On the left frame, select "localhost:3306" server, right click and select the menu to create a database



- Enter the database named (for example "energy"). The database name will have to set in the SpringBoot configuration file of SAPERE.
- keep the default charset and collation (utf8 / utf8_general_ci)

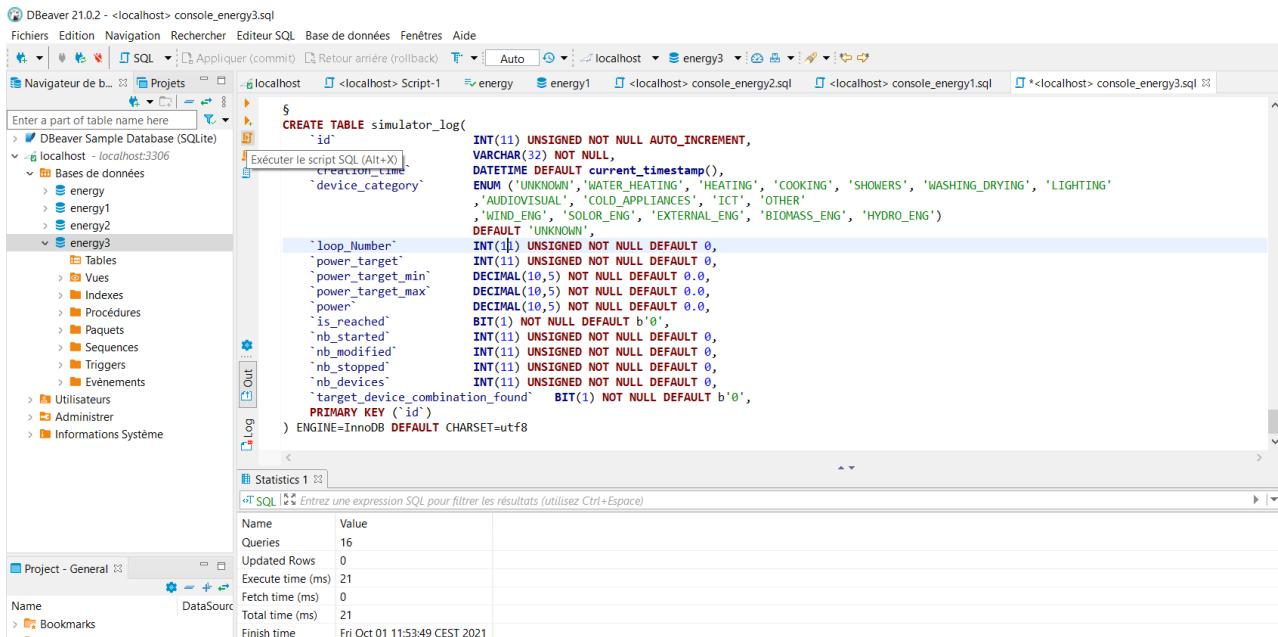


- Open a new SQL console: right click on the new database, select “SQL Editor”-> “Open SQL console”



This console can be saved into a local file. This file is linked to the chosen database and contains the entered SQL requests. You can choose to store several console files linked to the same database.

- On this new database, launch the following scripts which are in the Mariadb sub-folder. For each script, copy entire the content to the console and execute it by using “Execute SQL script” button
 - o 01_create_tables.sql: create the needed tables in the new database
 - o 02_procedures.sql: create the needed SQL function and store procedures
 - o 03_ref_data.sql: populate the referential data
 - o 04_assign_rights.sql: assign rights on this basis to the “learning_agent” account which is used by the different agents:

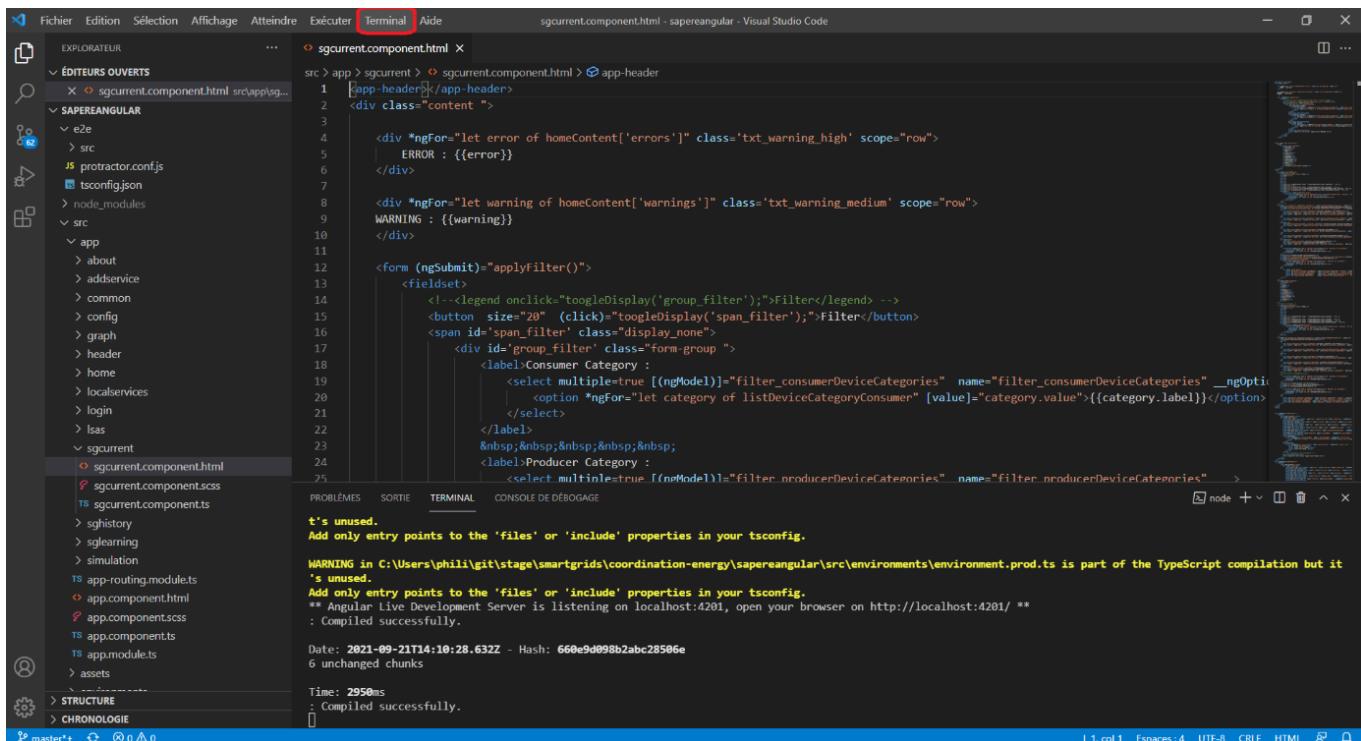


1.3/Web application:

1.3.1/ Installation of Visual Studio code

Visual Studio code is used as development environment for the web application.

- Download Visual Studio code from <https://code.visualstudio.com/download>
- Install Visual Studio Code
- Open the sapereangular folder from Visual Studio Code
- Launch a terminal from visual studio code ("Terminal" menu)



- On the new command window, go to the sapereangular directory (use WMing64 console or Linux commands)
- Launch the installation command: `npm install --save-dev @angular-devkit/build-angular`

1.3.2/ Install NodeJS

Download NodeJS V16 installer on <https://nodejs.org/en/blog/release/v16.20.0>

Launch the installer

Choose “add to PATH” option on the screen “Custom Setup”.

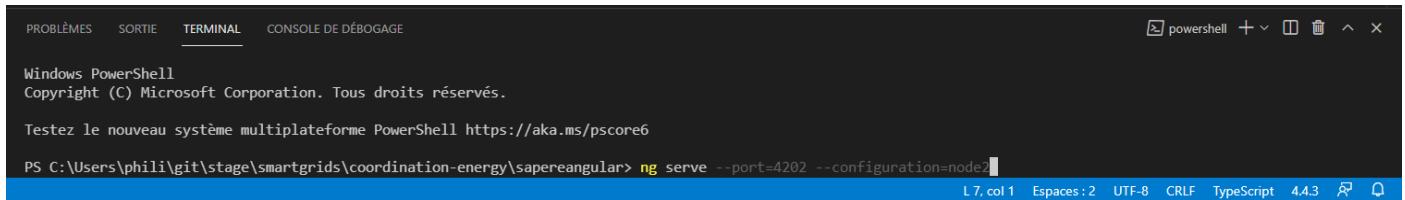
Launch npm install -g @angular/cli

1.3.2/ Starting the server

- Launch a terminal session
- move to sapereangular directory
- launch the starting command: `ng serve --port=<port_number> --configuration=<config_name>`

<port_number> is the port number and <config_name> is the configuration defined in the file named “environment.<config_name>.ts” and located in “environments” subdirector.

Example:



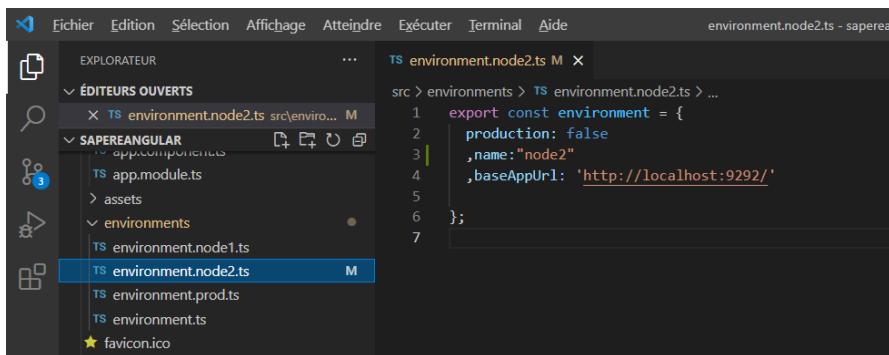
```
PROBLÈMES SORTIE TERMINAL CONSOLE DE DÉBOGAGE
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Users\phili\git\stage\smartgrids\coordination-energy\sapereangular> ng serve --port=4202 --configuration=node2
L 7, col 1   Espaces : 2   UTF-8   CRLF   TypeScript 4.4.3
```

By default,

- port number is 4200
- used configuration is the default configuration which is defined in “environments/environment.ts” file
“node2” configuration is chosen (as in the example above), the server uses the configuration defined in “environment.node2.ts” file.



The “baseAppUrl” field defines the address of the SAPERE coordination platform service located at the same node.

1.3.3/Adding a new configuration

If you need to add another node environment, you can add a new environment configuration

- enter the new configuration in a new file named “environments/environment.<config_name>.ts”.
- update angular.json file:
 - o insert the new configuration in “configurations” block:

add a “fileReplacements” item attached to the new configuration

Example, for “node3” configuration:

```
        },
        "node3": {
          "fileReplacements": [
            {
              "replace": "src/environments/environment.ts",
              "with": "src/environments/environment.node3.ts"
            }
          ]
        }
      }
```

- insert the new configuration in “serve” -> “configuration” block

```
"serve": {
  "builder": "@angular-devkit/build-angular:dev-server",
  "options": {
    "browserTarget": "angular:build"
  },
  "configurations": {
    "production": {
      "browserTarget": "angular:build:production"
    },
    "node1": {
      "browserTarget": "angular:build:node1"
    },
    "node2": {
      "browserTarget": "angular:build:node2"
    },
    "node3": {
      "browserTarget": "angular:build:node3"
    }
  }
}
```

- insert the new configuration in “e2e” -> “configuration” block:

```
"e2e": {
  "builder": "@angular-devkit/build-angular:protractor",
  "options": {
    "protractorConfig": "e2e/protractor.conf.js",
    "devServerTarget": "angular:serve"
  },
  "configurations": {
    "production": {
      "devServerTarget": "angular:serve:production"
    },
    "node1": {
      "devServerTarget": "angular:serve:node1"
    },
    "node2": {
      "devServerTarget": "angular:serve:node2"
    },
    "node3": {
      "devServerTarget": "angular:serve:node3"
    }
  }
},
```

1.4/ SAPERE coordination platform service

1.4.1/ Installation of Eclipse

The installer can be downloaded from:

<https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse-ide-java-developers>

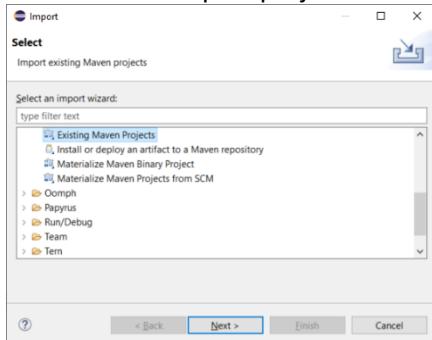
Also, you need to install java if you haven't already.

The last version can be downloaded from:

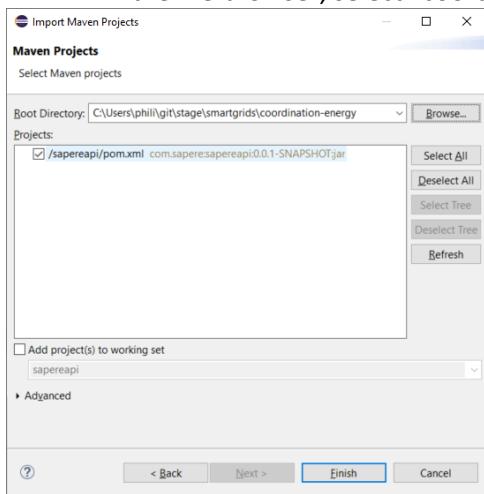
<https://www.oracle.com/java/technologies/downloads/>

1.4.2/Import of the SAPERE Spring-boot project

- File > Import project > Existing maven project



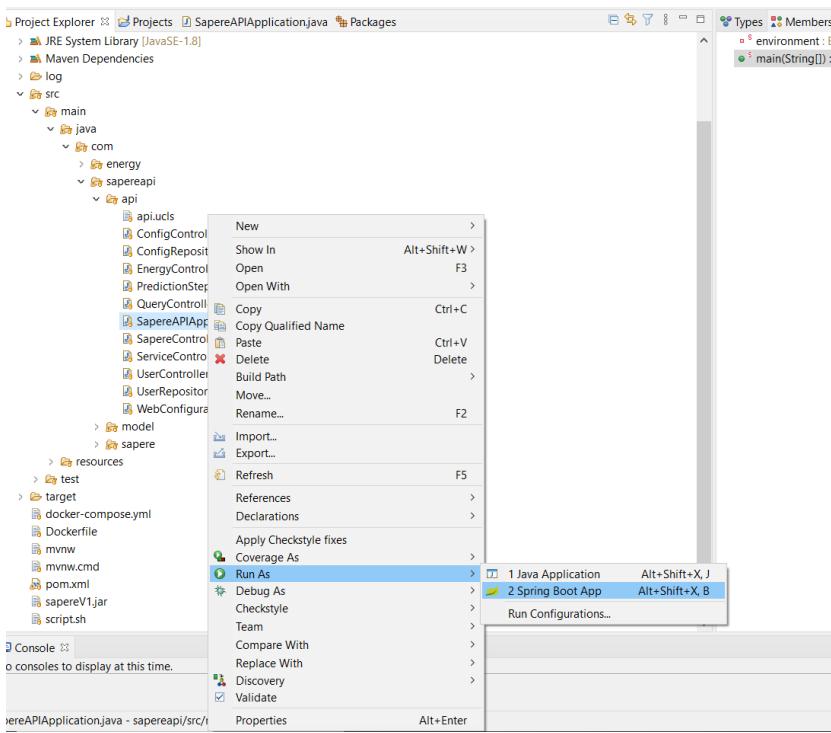
- In the file browser, select “coordination-energy” directory



- Select sapereapi/pom.xml and click on “Finish”

1.4.2/Starting the SAPERE Spring boot application

- Select the SapereAPIApplication class
- Right click on the class name and select “Run AS” -> “Spring boot app”



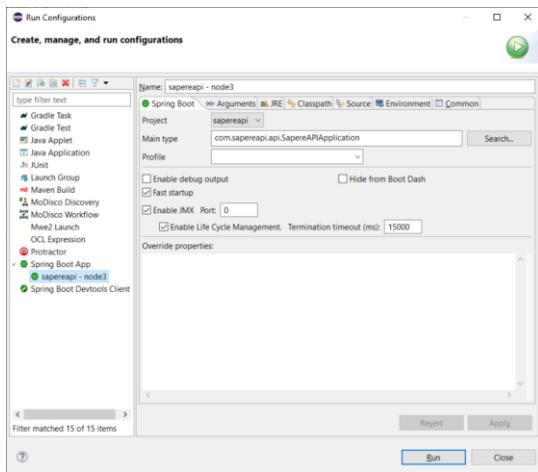
The default configuration information is in the "application.properties" file. The default server port which is set to 9090

```

server.port=9090
# =====#
# DATABASE
# =====#
#spring.jpa.hibernate.ddl-auto:update
#spring.datasource.url=jdbc:mysql://10.194.69.15:3306/sapere?serverTimezone=UTC&useLegacyDatetimeCode=false
#spring.datasource.username=root
#spring.datasource.password=root99
spring.main.allow-bean-definition-overriding=true
#mongodb
#spring.data.mongodb.host=localhost
#spring.data.mongodb.port=27017
#spring.data.mongodb.database=sapere
#spring.data.mongodb.uri=mongodb://localhost:27017/sapere
spring.data.mongodb.uri=mongodb://localhost:27017/sapere
    
```

You can change the running configuration:

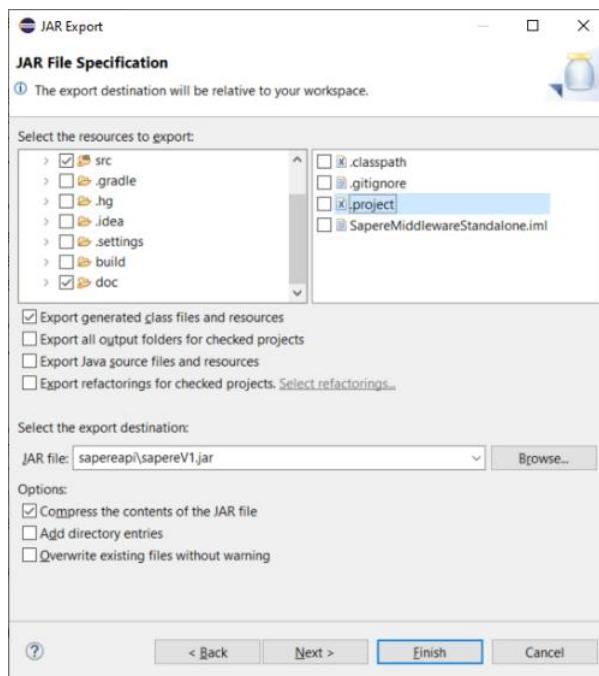
- Select the menu run -> run configuration
- Update the configuration name and the argument "Arguments". The argument corresponds to the configuration name ("node3" for "application-node3.properties")

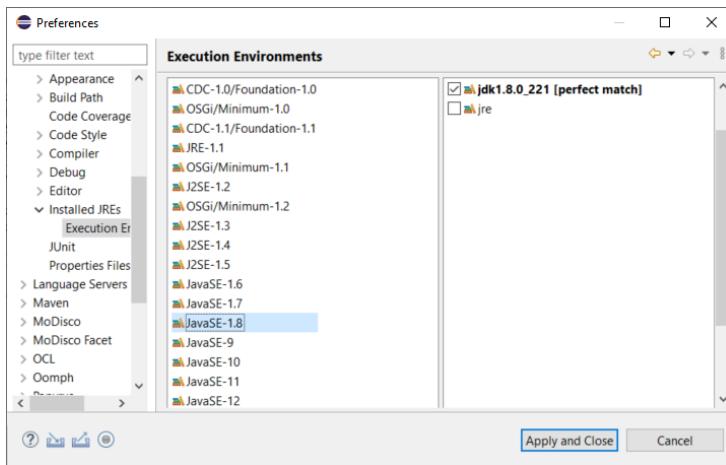


1.4.3/Regeneration of SapereV1.jar

SapereV1.jar is the core library of SAPERE

- From the Eclipse project SapereMiddleWareStandalone, export the jar: "right click" -> "export" -> "jar file"
- on the right frame, deselect ".classpath", ".gitignore", ".project" and "SapereMiddlewareStandalone.iml" files because they are not necessary for the generation of this jar.
- on the left frame, deselect the "build" directory because it is not necessary for the generation of this jar.





- Make sure the jar is properly updated in the sapearpi project
- Go to the command line in the sapereapi project
- Launch :

`mvn install:install-file -Dfile=sapereV1.jar -DgroupId=com.sapere -DartifactId=sapere -Dversion=1.0 -Dpackaging=jar`

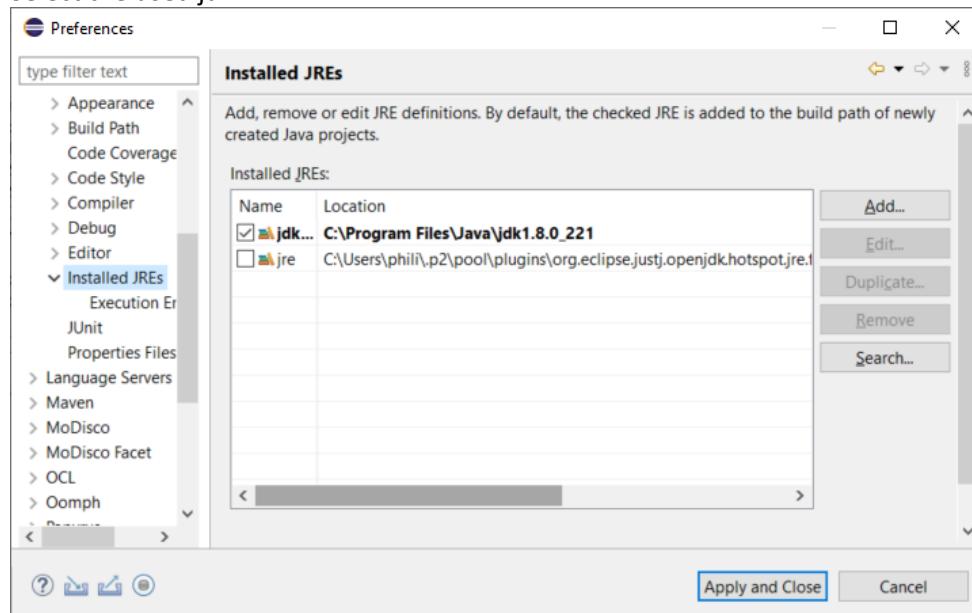
- Launch : mvn clean package
- Launch the maven task "update project" from the eclipse project sapereapi

1.4.4/JDK Problem resolution:

"[ERROR] No compiler is provided in this environment. Perhaps you are running on a JRE rather than a JDK" in the maven command "mvn clean package"

In Windows->Preference->Installed JREs

- Select the used jdk



- idem on Execution Environment -> JavaSE-1.8

- Make sure that the system environment variables point to the JDK

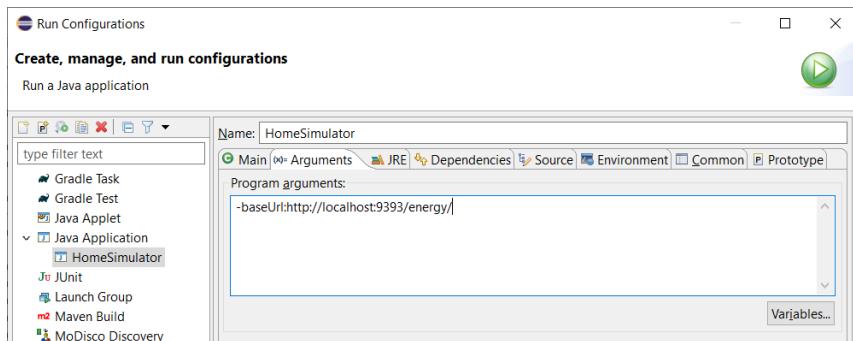
Path: must contains '...\\Java\\jdk1.8.0_221\\bin'

JAVA_HOME: must contains '...\\Java\\jdk1.8.0_221'

1.5/Running the home simulator

Select run->run configuration->

- Enter the main class com.energy.test.HomeSimulator
 - o Add the coordination platform service address in the parameters:
baseUrl:http://localhost:<server_port>/energy/



- Click on “apply” to save the configuration
- Click on Run

2/ Test with 2 node environments

Node	Start web application	Start test simulator	Coordination platofrm service start-up
Node1	ng serve --port=4201 --configuration=node1	com.energy.test.HomeSimulator -baseUrl:http://localhost:9191/energy/	com.sapereapi.api.SapereAPIApplication node1
Node2	ng serve --port=4202 --configuration=node2	com.energy.test.HomeSimulator -baseUrl:http://localhost:9292/energy/	com.sapereapi.api.SapereAPIApplication node2