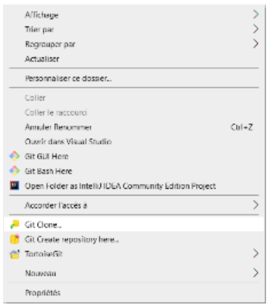
1/Installation

The source code is available on the GitHub repository: to retrieve it, run a gitclone command on the following repository address:
https://github.com/philippe-glass/energy/tree/master



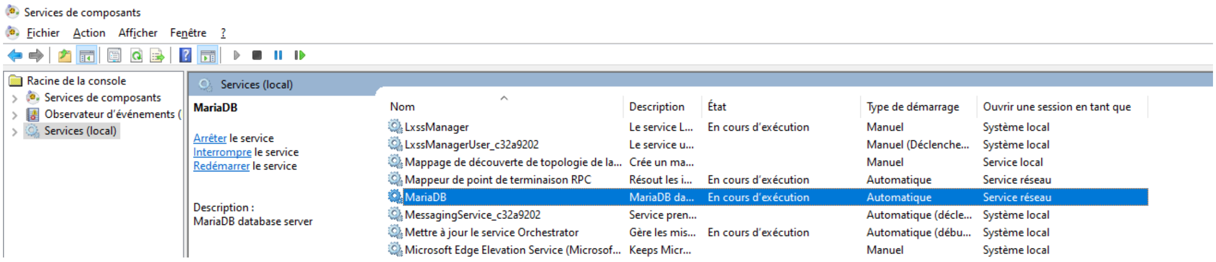
1.2/Coordination platform database

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 le 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)



MariaDB

Arrêter le service
Interrompre le service
Redémarrer le service

Description :
MariaDB database server

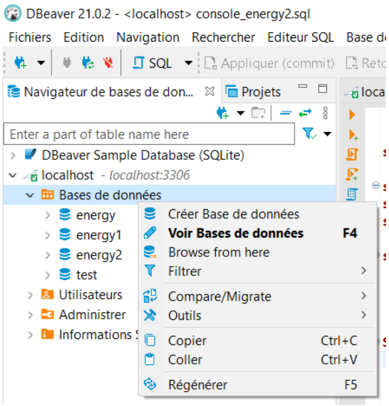
Nom	Description	État	Type de démarrage	Ouvrir une session en tant que
LxssManager	Le service L...	En cours d'exécution	Manuel	Système local
LxssManagerUser_c32a9202	Le service u...		Manuel (Déclenche...	Système local
Mappage de découverte de topologie de la...	Crée un ma...		Manuel	Service local
Mappeur de point de terminaison RPC	Résout les i...	En cours d'exécution	Automatique	Service réseau
MariaDB	MariaDB da...	En cours d'exécution	Automatique	Service réseau
MessagingService_c32a9202	Service pren...		Automatique (décle...	Système local
Mettre à jour le service Orchestrator	Gère les mis...	En cours d'exécution	Automatique (débu...	Système local
Microsoft Edge Elevation Service (Microsof...	Keeps Micr...		Manuel	Système local

1.2.2/Client GUI installation

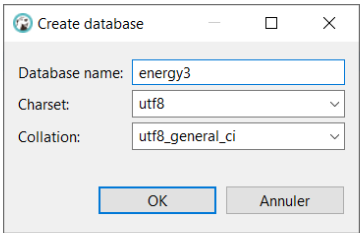
- Download the last version of DBEaver installer from: https://dbeaver.io/download/
- Install DBEaver (for example: dbeaver-ce-21.0.2-x86_64-setup.exe

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)



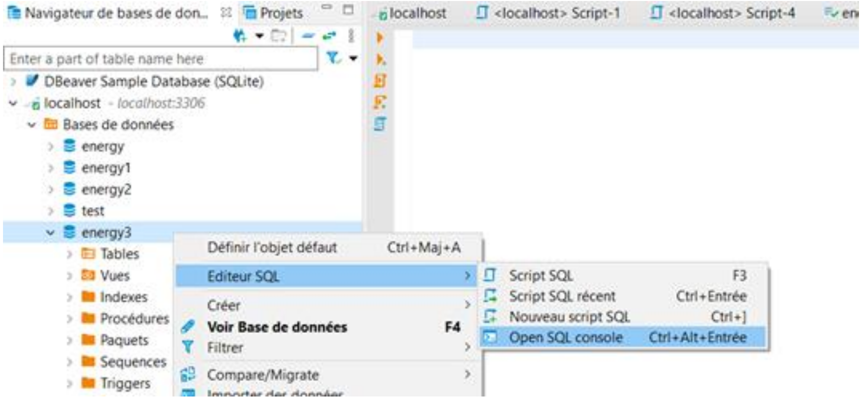
Create database

Database name: energy3

Charset: utf8

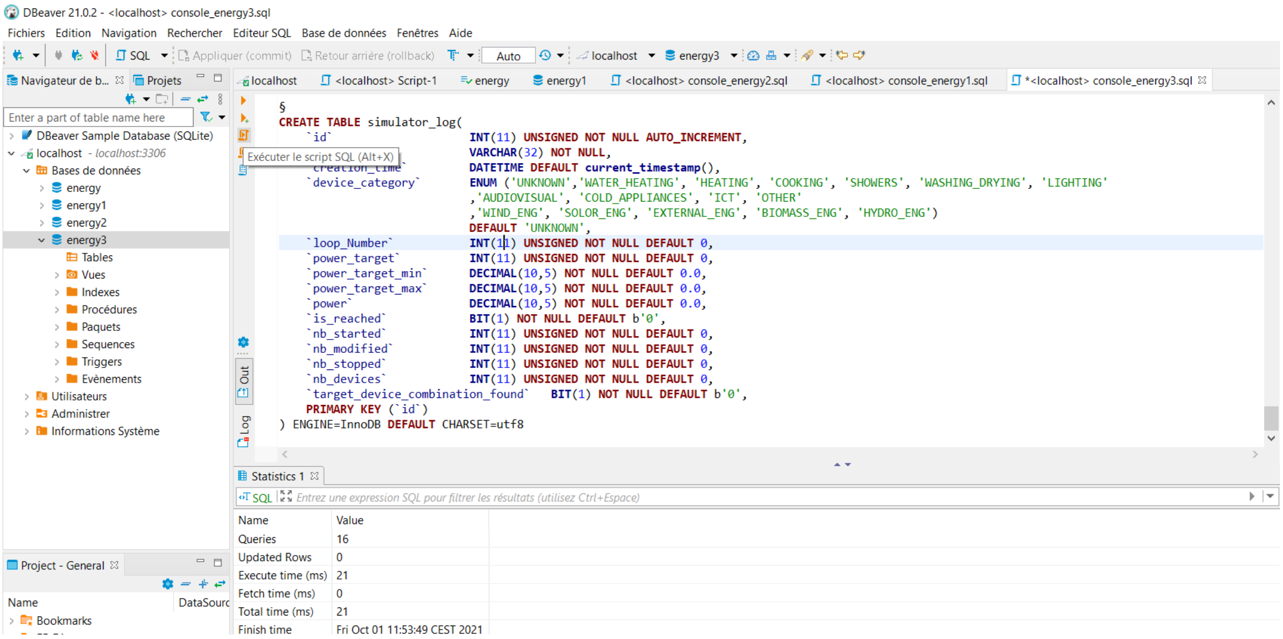
Collation: utf8_general_ci

- Open a new SQL console: right clock 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. Yon 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
    - 01_create_tables.sql: create the needed tables in the new database
    - 02_procedures.sql: create the needed SQL function and store procedures
    - 03_ref_data.sql: populate the referential data
    - 04_assign_rights.sql: assign rights on this basis to the “learning_agent” account which is used by the different agents:



DBeaver 21.0.2 - <localhost> console_energy3.sql
Fichiers Edition Navigation Rechercher Editeur SQL Base de données Fenêtres Aide
SQL Appliquer (commit) Retour arrière (rollback) Auto localhost energy3
Navigateur de b... Projets
localhost <localhost> Script-1 energy energy1 <localhost> console_energy2.sql <localhost> console_energy1.sql *<localhost> console_energy3.sql
Enter a part of table name here
DBeaver Sample Database (SQLite)
localhost - localhost:3306
Bases de données
energy
energy1
energy2
energy3
Tables
Vues
Indexes
Procédures
Paquets
Sequences
Triggers
Evènements
Utilisateurs
Administrer
Informations Système
Exécuter le script SQL (Alt+X)
Out
Log
Statistics 1
SQL
Entrez une expression SQL pour filtrer les résultats (utilisez Ctrl+Espace)
Project - General
Name
DataSourc
Bookmarks

```
§
CREATE TABLE simulator_log(
    `id`                        INT(11) UNSIGNED NOT NULL AUTO_INCREMENT,
                                VARCHAR(32) NOT NULL,
    `creation_time`             DATETIME DEFAULT current_timestamp(),
    `device_category`           ENUM ('UNKNOWN','WATER_HEATING', 'HEATING', 'COOKING', 'SHOWERS', 'WASHING_DRYING', 'LIGHTING'
                                ,'AUDIOVISUAL', 'COLD_APPLIANCES', 'ICT', 'OTHER'
                                ,'WIND_ENG', 'SOLOR_ENG', 'EXTERNAL_ENG', 'BIOMASS_ENG', 'HYDRO_ENG')
                                DEFAULT 'UNKNOWN',
    `loop_Number`               INT(11) UNSIGNED NOT NULL DEFAULT 0,
    `power_target`              INT(11) UNSIGNED NOT NULL DEFAULT 0,
    `power_target_min`          DECIMAL(10,5) NOT NULL DEFAULT 0.0,
    `power_target_max`          DECIMAL(10,5) NOT NULL DEFAULT 0.0,
    `power`                     DECIMAL(10,5) NOT NULL DEFAULT 0.0,
    `is_reached`                BIT(1) NOT NULL DEFAULT b'0',
    `nb_started`                INT(11) UNSIGNED NOT NULL DEFAULT 0,
    `nb_modified`               INT(11) UNSIGNED NOT NULL DEFAULT 0,
    `nb_stopped`                INT(11) UNSIGNED NOT NULL DEFAULT 0,
    `nb_devices`                INT(11) UNSIGNED NOT NULL DEFAULT 0,
    `target_device_combination_found`   BIT(1) NOT NULL DEFAULT b'0',
    PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8
```

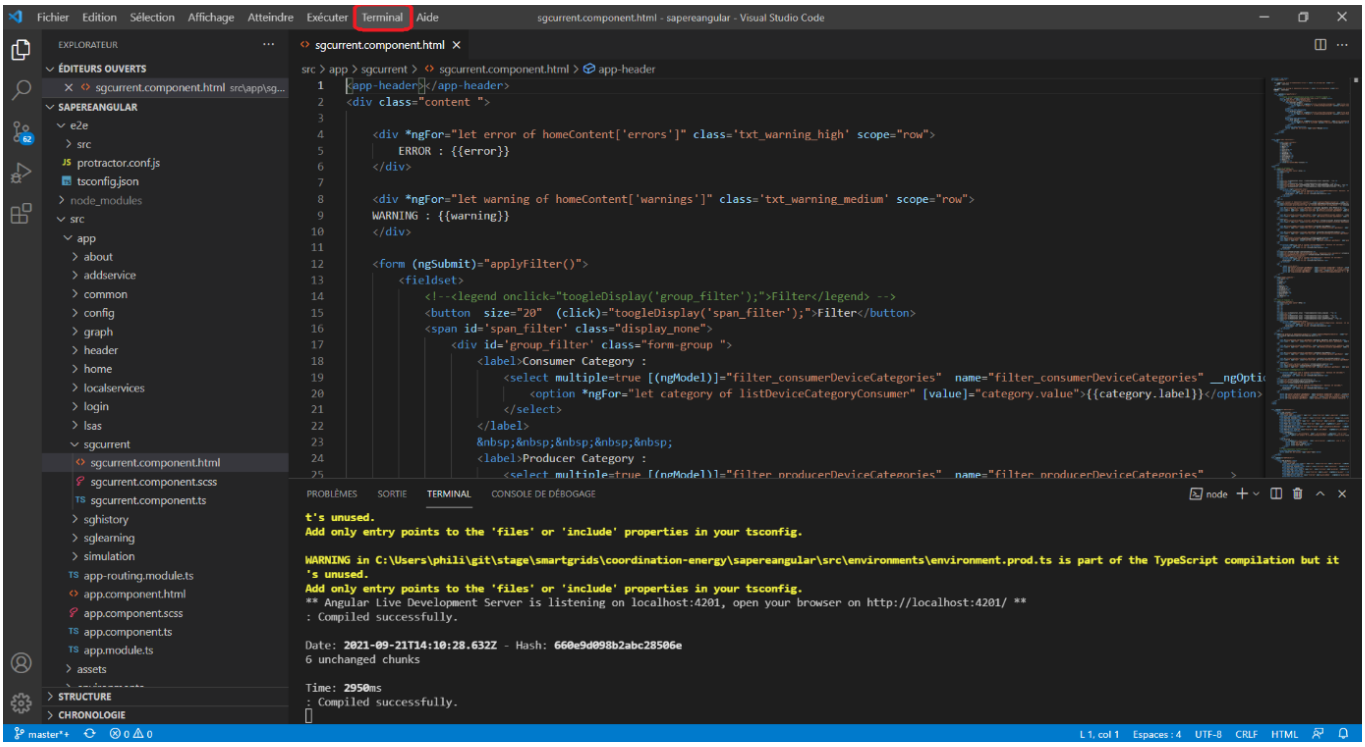
Name	Value
Queries	16
Updated Rows	0
Execute time (ms)	21
Fetch time (ms)	0
Total time (ms)	21
Finish time	Fri Oct 01 11:53:49 CEST 2021

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)



```
<app-header></app-header>
<div class="content ">

    <div *ngFor="let error of homeContent['errors']" class='txt_warning_high' scope="row">
        ERROR : {{error}}
    </div>

    <div *ngFor="let warning of homeContent['warnings']" class='txt_warning_medium' scope="row">
    WARNING : {{warning}}
    </div>

    <form (ngSubmit)="applyFilter()">
        <fieldset>
            <!--<legend onclick="toogleDisplay('group_filter');">Filter</legend> -->
            <button  size="20"  (click)="toogleDisplay('span_filter');">Filter</button>
            <span id='span_filter' class="display_none">
                <div id='group_filter' class="form-group ">
                    <label>Consumer Category :
                        <select multiple=true [(ngModel)]="filter_consumerDeviceCategories"  name="filter_consumerDeviceCategories" __ngOptio
                            <option *ngFor="let category of listDeviceCategoryConsumer" [value]="category.value">{{category.label}}</option>
                        </select>
                    </label>
                    &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
                    <label>Producer Category :
                        <select multiple=true [(ngModel)]="filter_producerDeviceCategories"  name="filter_producerDeviceCategories"    >
```

```
t's unused.
Add only entry points to the 'files' or 'include' properties in your tsconfig.

WARNING in C:\Users\phili\git\stage\smartgrids\coordination-energy\sapereangular\src\environments\environment.prod.ts is part of the TypeScript compilation but it
's unused.
Add only entry points to the 'files' or 'include' properties in your tsconfig.
** Angular Live Development Server is listening on localhost:4201, open your browser on http://localhost:4201/ **
: Compiled successfully.

Date: 2021-09-21T14:10:28.632Z - Hash: 660e9d098b2abc28506e
6 unchanged chunks

Time: 2950ms
: Compiled successfully.
```

- 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”.

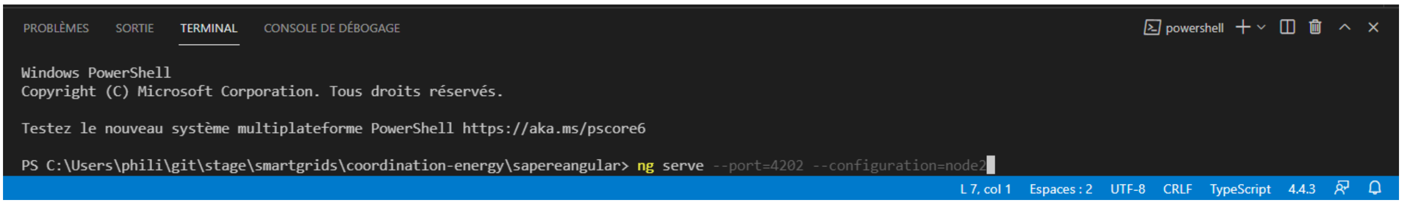
Launch npm install -g @angular/cli

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.

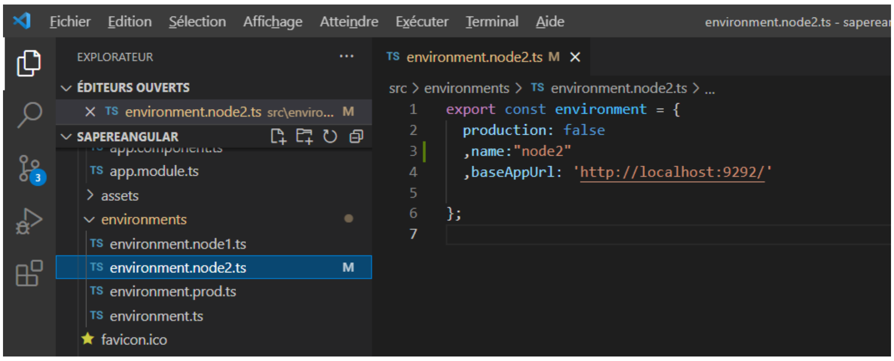
Example:



```
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
```

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.



```
export const environment = {
  production: false
  ,name:"node2"
  ,baseAppUrl: 'http://localhost:9292/'

};

```

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

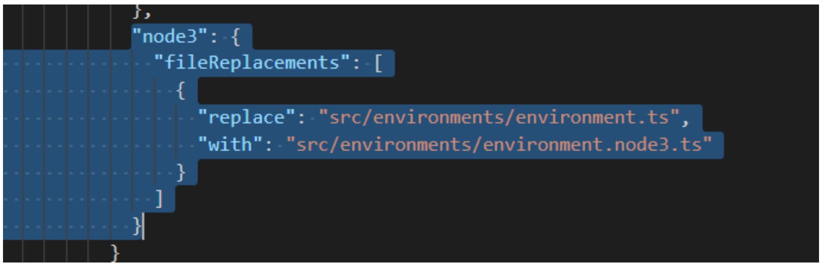
1.3.3/Adding a new configuration

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:
  - insert the new configuration in “configurations” block:

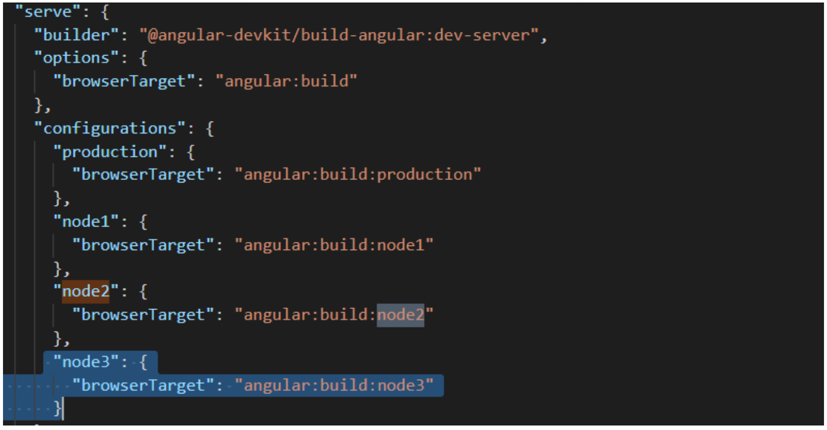
add a “fileReplacements” item attached to the new configuration

Example, for “node3” configuration:



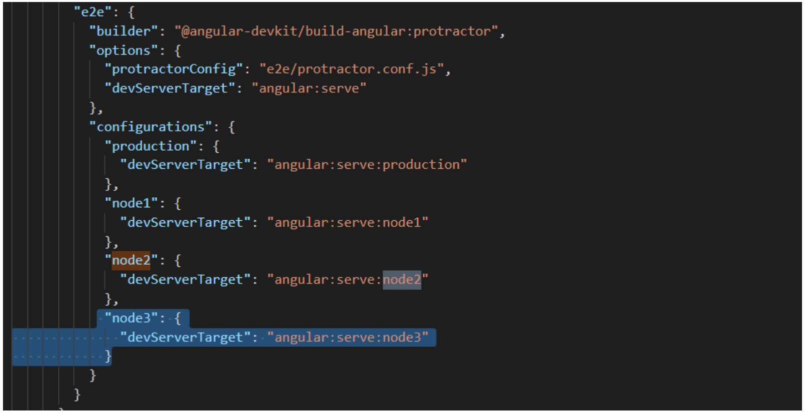
```
},
"node3": {
  "fileReplacements": [
    {
      "replace": "src/environments/environment.ts",
      "with": "src/environments/environment.node3.ts"
    }
  ]
}
}
```

- 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

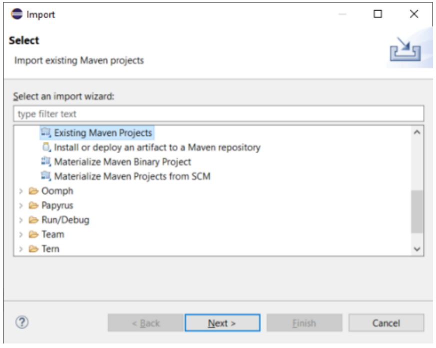
1.4.1/ Installation of Eclipse

The installer can be downloaded from:
https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse-ide-java-developers

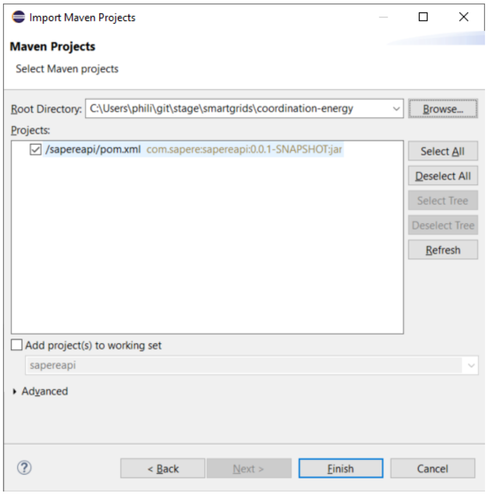
Also, you need to install java if you haven't already.
The last version can be downloaded from:
https://www.oracle.com/java/technologies/downloads/

1.4.2/Import of the SAPERE Spring-boot project

- File > Import project > Existing maven project


Import
Select
Import existing Maven projects
Select an import wizard:
type filter text
Existing Maven Projects
Install or deploy an artifact to a Maven repository
Materialize Maven Binary Project
Materialize Maven Projects from SCM
Oomph
Papyrus
Run/Debug
Team
Tern
< Back
Next >
Finish
Cancel

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



Root Directory: C:\Users\phili\git\stage\smartgrids\coordination-energy

Projects:

☑ /sapereapi/pom.xml com.sapere:sapereapi:0.0.1-SNAPSHOT:jar

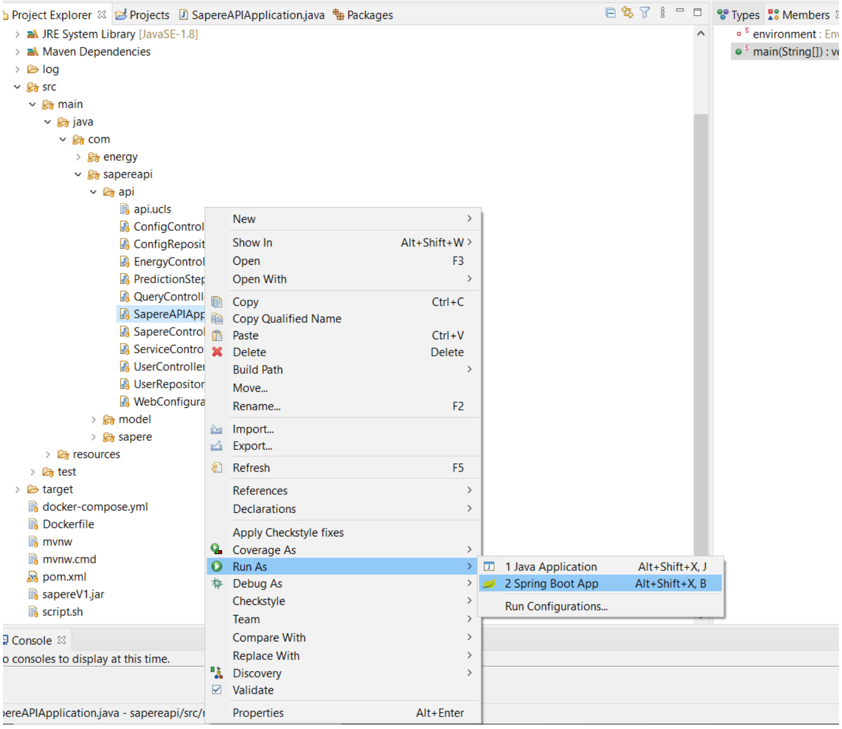
□ Add project(s) to working set

sapereapi

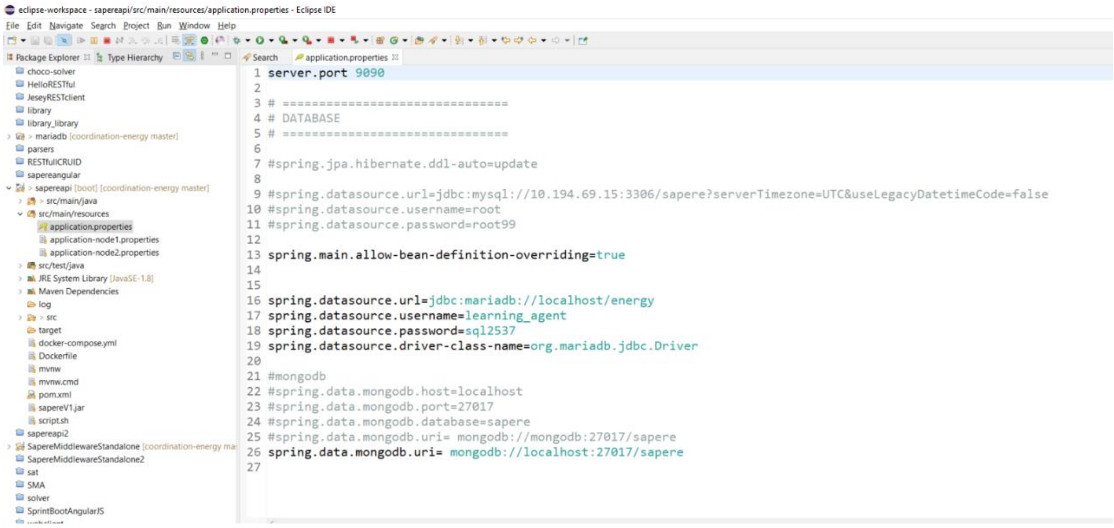
- Select sapereapi/pom.xml and click on “Finish”

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**”



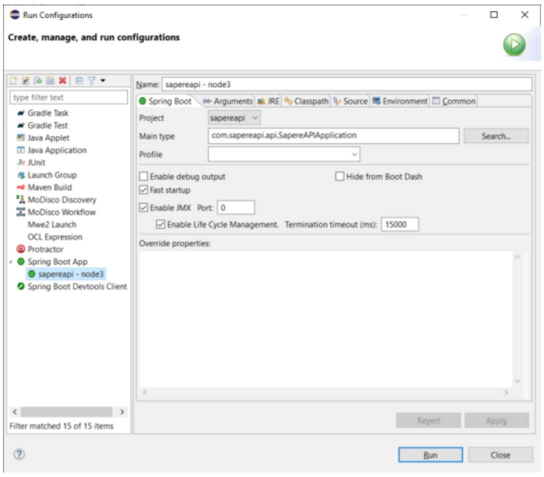
The default configuration information is in the "application.properties" file. The default server port which is set to 9090


eclipse-workspace - sapereapi/src/main/resources/application.properties - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Package Explorer
Type Hierarchy
Search
application.properties
choco-solver
HelloRESTful
JeseyRESTclient
library
library_library
mariadb [coordination-energy master]
parsers
RESTfulCRUD
sapereangular
sapereapi [boot] [coordination-energy master]
src/main/java
src/main/resources
application.properties
application-node1.properties
application-node2.properties
src/test/java
JRE System Library [JavaSE-1.8]
Maven Dependencies
log
src
target
docker-compose.yml
Dockerfile
mvnw
mvnw.cmd
pom.xml
sapereV1.jar
script.sh
sapereapi2
SapereMiddlewareStandalone [coordination-energy ma
SapereMiddlewareStandalone2
sat
SMA
solver
SprintBootAngularJS

```
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


spring.datasource.url=jdbc:mariadb://localhost/energy
spring.datasource.username=learning_agent
spring.datasource.password=sql2537
spring.datasource.driver-class-name=org.mariadb.jdbc.Driver

#mongodb
#spring.data.mongodb.host=localhost
#spring.data.mongodb.port=27017
#spring.data.mongodb.database=sapere
#spring.data.mongodb.uri= mongodb://mongodb: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”)



Name: sapereapi - node3

Project: sapereapi

Main type: com.sapereapi.api.SapereAPIApplication

Profile:

☐ Enable debug output ☐ Hide from Boot Dash

☑ Fast startup

☑ Enable JMX Port: 0

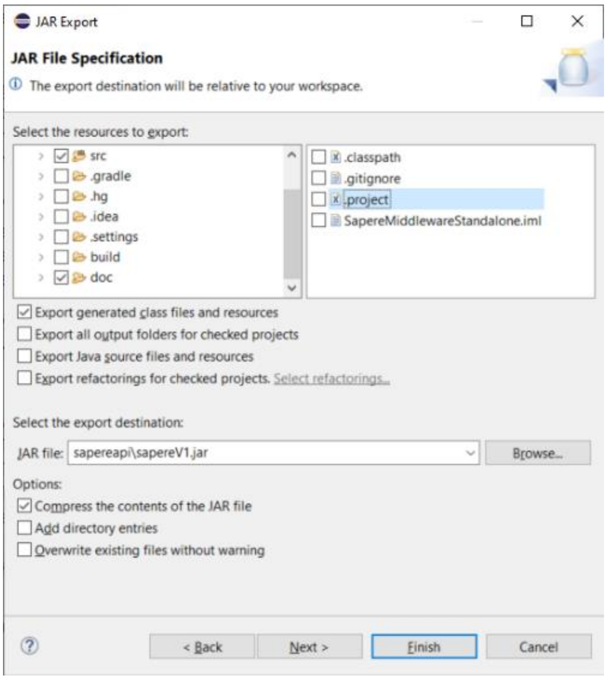
☑ Enable Life Cycle Management. Termination timeout (ms): 15000

Override 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.
-



Select the resources to export:

☑ src
☐ .gradle
☐ .hg
☐ .idea
☐ .settings
☐ build
☑ doc

☐ .classpath
☐ .gitignore
☐ .project
☐ SapereMiddlewareStandalone.iml

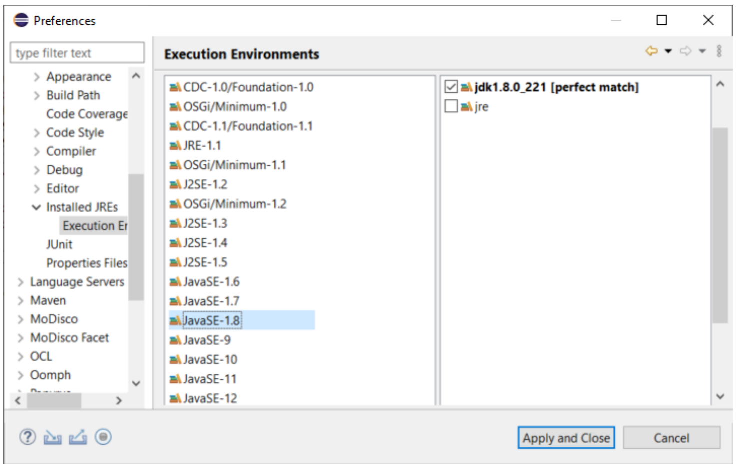
☑ Export generated class files and resources
☐ Export all output folders for checked projects
☐ Export Java source files and resources
☐ Export refactorings for checked projects. Select refactorings...

Select the export destination:

JAR file: sapereapi\sapereV1.jar

Options:

☑ Compress the contents of the JAR file
☐ Add directory entries
☐ Overwrite existing files without warning



Execution Environments

CDC-1.0/Foundation-1.0
OSGi/Minimum-1.0
CDC-1.1/Foundation-1.1
JRE-1.1
OSGi/Minimum-1.1
J2SE-1.2
OSGi/Minimum-1.2
J2SE-1.3
J2SE-1.4
J2SE-1.5
JavaSE-1.6
JavaSE-1.7
JavaSE-1.8
JavaSE-9
JavaSE-10
JavaSE-11
JavaSE-12

☑ jdk1.8.0_221 [perfect match]
□ jre

- 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
```

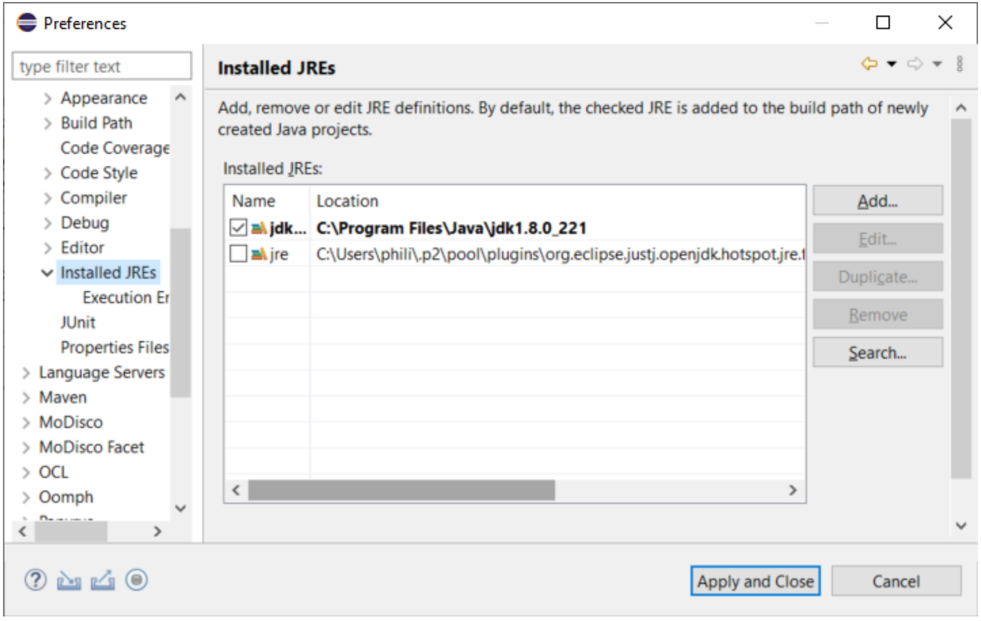
- Launch : mvn clean package
- Launch the maven task "update project" from the eclipse project sapereapi

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



Installed JREs

Add, remove or edit JRE definitions. By default, the checked JRE is added to the build path of newly created Java projects.

Installed JREs:

Name	Location
☑ jdk...	C:\Program Files\Java\jdk1.8.0_221
□ jre	C:\Users\phili\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f

- idem on Execution Environment -> JavaSE-1.8

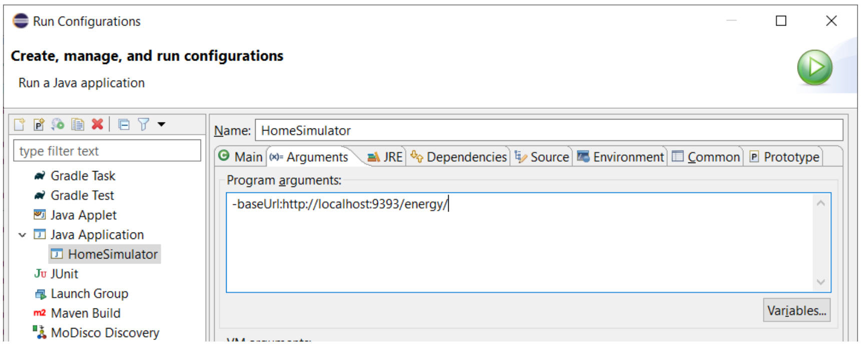
- Make sure that the system environment variables point to the JDK

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
    - ○ Add the coordination platform service address in the parameters:
      baseUrl:http://localhost:<server_port>/energy/



Run Configurations

Create, manage, and run configurations

Run a Java application

Name: HomeSimulator

Program arguments:

```
-baseUrl:http://localhost:9393/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