

FONTALVO Lilian | NOTICE Première Maquette Projet CPOO

28/10/2021

Diagramme UML

Présentation rapide du corps de projet en diagramme UML avec explications de choix effectués.

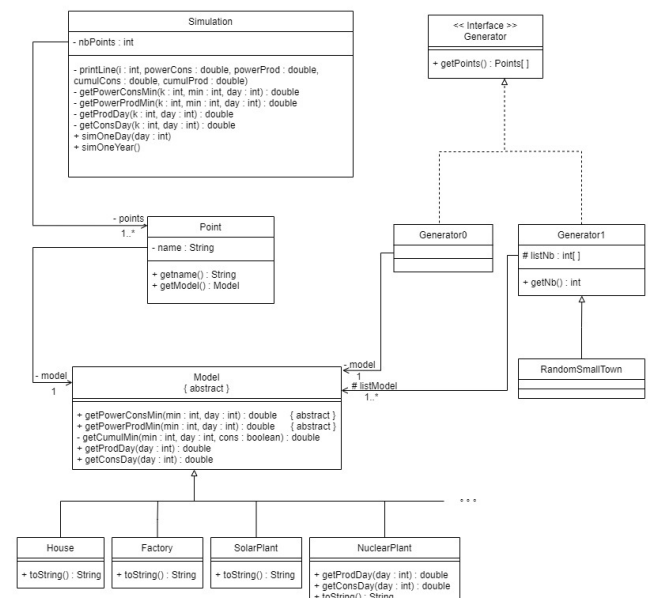
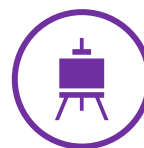


Le diagramme se compose de deux parties quasiment indépendantes : une partie simulation (sur la gauche) et une partie génération de scénario (sur la droite).

La partie simulation comprend au plus bas niveau l'ensemble des classes formant la classe abstraite *Model*. Choisir une classe abstraite paraissait judicieuse dans l'implémentation des fonctions d'énergie quotidienne qui sont soit calculé automatique soit spécifié dans les sous-classes (permettant ainsi un gain de calcul). Elle permet aussi de garantir une adaptation efficace à de futures classes qui pourraient, par exemple, dépendre de la météo, ou d'autres paramètres.

La classe *Point* représente une entité physique de la simulation, prenant comme argument un modèle. Enfin, *Simulation* fournit toutes les méthodes de calcul et d'affichage de la simulation. Il prend comme argument une liste de *Point*.

La deuxième partie permet de faciliter la création de la liste de *Point* nécessaire à la simulation.



« *RandomSmallTown* est une sous-classe de *Generator1* permettant de créer une liste de *Point* en fournissant les modèles qui seront en nombre aléatoire »

Notice d'utilisation

The simpler the better.

Les générateurs sont là pour faciliter l'utilisation de *Simulation*, et ainsi améliorer l'expérience de l'utilisateur. Il y a dans cette première maquette 3 générateurs : *Generator0*, *Generator1* et *RandomSmallTown*.

Generator0 permet de récupérer une liste avec un unique point d'intérêt à l'aide du modèle fourni.

Generator1 permet de récupérer une liste de *Point* en fournissant une liste de *Model* et une liste d'entier représentant leur nombre d'occurrence dans la liste.

RandomSmallTown génère automatiquement et aléatoirement une petite ville (de 1000 à 10000 maisons, de 1 à 5 usines, de 5 à 10 centrale solaire et 0 ou 1 centrale nucléaire).

Pour réaliser une simulation, commencez par créer, dans le fichier *main*, un nouveau *scénario* en initialisant un générateur avec les arguments correspondant ou en donnant directement une liste de *Point*.

Ensuite, créez une nouvelle *simulation* en fournissant au constructeur la liste de points *scenario.getPoints()*.

Pour finir et récupérer le bilan quotidien ou annuel, appliquez respectivement les méthodes *simOneDay(day, graph)* ou *simOneYear(graph)* à *simulation*. Le paramètre *graph* permet d'afficher un graphique de la puissance consommée.

Exécutez le fichier *main*.

Exemple d'utilisation des générateurs :

```
src > core > mains > SimulationMain.java > ...
1  package core.mains;
2
3  import core.generators.*;
4  import core.models.*;
5  import core.simulation.*;
6
7  public class SimulationMain {
8      Run | Debug
9      public static void main(String[] args) {
10
11          // For only one element
12          Generator scenario0 = new Generator0(new NuclearPlant());
13          Simulation sim0 = new Simulation(scenario0.getPoints());
14          sim0.simOneDay(241, false); // jour 241, le 28 août, un jour comme un autre (jour 0 = 1er janvier)
15          sim0.simOneYear(false);
16
17          // For a list of points and their occurrences
18          Model[] listModel = { new House(), new Factory(), new SolarPlant(), new NuclearPlant() };
19          int[] listNb = { 100, 1, 4, 1 };
20          Generator scenario1 = new Generator1(listModel, listNb);
21          Simulation sim1 = new Simulation(scenario1.getPoints());
22          sim1.simOneDay(241, false);
23          sim1.simOneYear(false);
24
25          // For a small randomly generated town
26          Generator scenario2 = new RandomSmallTown();
27          Point[] listPoint = scenario2.getPoints();
28          Simulation sim2 = new Simulation(listPoint);
29          sim2.simOneDay(241, true);
30          sim2.simOneYear(true);
31      }
```



Site web :
www.lms.isae.fr/course/view.php?id=1737



Adresse email:
Lilian.FONTALVO@student.isae-superaero.fr