

Licence Informatique

Projet Semestre 5

Le projet neOCampus (<https://www.irit.fr/neocampus/>) est un projet de l'Université Paul Sabatier qui vise notamment à contrôler les consommations énergétiques du campus. L'ensemble des fluides consommés sur le campus est mesuré au travers de capteurs localisés sur le campus. Ces fluides sont la consommation d'eau, d'air comprimé, d'électricité, de gaz et de chauffage. Afin de pouvoir contrôler les consommations énergétiques du campus, il est nécessaire de pouvoir visualiser les différentes données et ce à plusieurs niveaux : bâtiments, étages, pièces, etc.

Le premier objectif de ce projet est de réaliser une interface de visualisation des données de consommation énergétique du campus. La visualisation de ces données devra pouvoir se faire de deux manières. D'une part « en temps réel » : dans ce cas, l'interface de visualisation affiche les données actuelles afin de surveiller les potentiels problèmes de consommation (fuite d'eau, radiateur défaillant, etc.) et, en cas de problèmes détectés, en alerter le personnel chargé de la surveillance. D'autre part, dans un but d'analyse *a posteriori*, l'interface de visualisation pourra reprendre, afficher et comparer des données de consommation sur une période déterminée (une semaine, un mois, une année, etc.).

Cependant, actuellement, peu de bâtiments du campus sont réellement équipés de capteurs permettant de récupérer ces données de consommation en temps réel. Ainsi, pour pouvoir simuler le bon fonctionnement des interfaces de visualisation, le second volet de ce projet est de développer une interface qui permette de simuler le fonctionnement de capteurs en émettant des données identiques à celles qui seraient fournies par des capteurs réels.

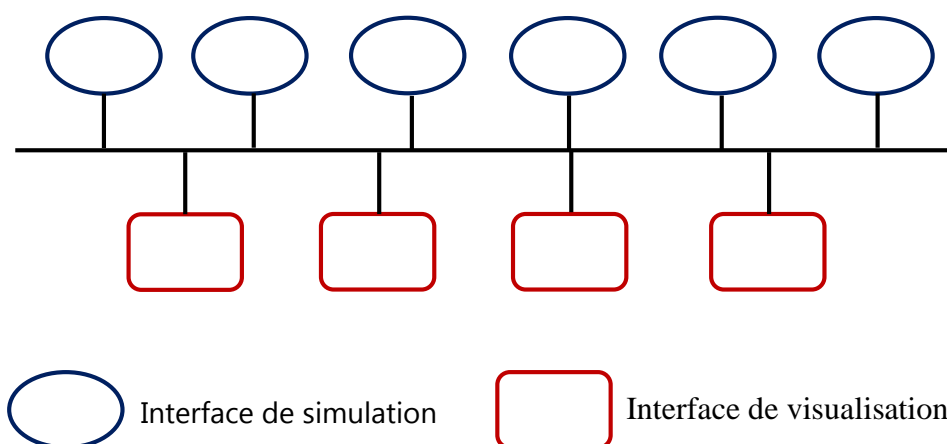


Figure 1 : Exemple de réseau de simulation avec 6 interfaces de simulation

et 4 interfaces de visualisation connectées

Une fois que le campus aura été entièrement équipé de capteurs, chaque agent de l'université travaillant sur la consommation des fluides du campus aura une interface de visualisation. Ce projet doit donc permettre de connecter plusieurs interfaces de simulation (à terme remplacées par des capteurs réels) à plusieurs interfaces de visualisation : c'est-à-dire que chaque interface de visualisation connectée à un réseau peut récupérer l'ensemble des données des différentes interfaces de simulation de ce même réseau. Nous nommerons cet ensemble d'interfaces connectées un réseau de simulation (voir Figure 1).

Cahier des charges

Paramètres commun aux deux interfaces

Les interfaces de visualisation et de simulation devront pouvoir communiquer avec les autres interfaces d'un même réseau de simulation. Par conséquent, il est nécessaire que sur chaque interface, l'utilisateur puisse paramétrer les informations liées à la connexion au réseau de simulation. Cela signifie que les deux interfaces doivent fournir la possibilité de choisir une adresse IP et un port de manière graphique (champ de texte, fenêtre pop-up, liste déroulante, etc.). Vos interfaces doivent aussi fournir des outils graphiques pour permettre la connexion et la déconnexion au réseau.

Interface de simulation

L'interface de simulation vous permettra de simuler le fonctionnement d'un capteur. Ainsi, l'interface permettra à l'utilisateur de renseigner les paramètres du capteur :

- Son identifiant (ex : Radiateur, Prise électrique, etc.)
- Le type de données qu'il mesure (ex : Température, Humidité, Consommation électrique, etc.)
- Sa localisation
 - o Pour un capteur d'extérieur, ses coordonnées GPS
 - o Pour un capteur d'intérieur, le bâtiment, l'étage et la salle dans laquelle il se trouve ainsi que sa position relative (ex : Bâtiment U3, 2^{ème} étage, salle 307, au milieu du plafond)
- L'intervalle des valeurs possibles (ex : la température mesurée va de -10° à 50° sur ce capteur)

Les coordonnées GPS sont au format (Latitude, Longitude) exprimées en degrés décimaux ($-90 < \text{Lat} < 90$; $-180 < \text{Long} < 180$) et sont des chiffres à virgule. La liste des bâtiments, étages et salles sera récupérée dans un fichier de configuration (qui se trouvera dans le répertoire courant de votre programme), et seules les informations enregistrées dans ce fichier pourront être utilisées par l'interface pour remplir la localisation d'un capteur intérieur (par exemple, proposée sous forme de liste déroulante à l'utilisateur). La position relative est un commentaire que peut rajouter l'utilisateur pour aider à localiser plus facilement le capteur dans la pièce.

Ces informations seront entrées au lancement de l'interface de simulation et ne seront plus modifiables. Une fois ces informations rentrées, l'interface de simulation doit permettre :

- De se connecter ou déconnecter du réseau

- De choisir si les données du capteur sont générées aléatoirement (dans l'intervalle choisi) ou si elles sont entrées via l'interface (outil graphique permettant de définir la valeur actuelle du capteur, tout en contraignant le choix de l'utilisateur dans l'intervalle des valeurs possibles)
- De choisir la fréquence d'envoi des données (ex : envoyer la valeur de température toutes les 5 secondes ou toutes les 200 millisecondes, etc.)

Interface de visualisation

En premier lieu, l'interface de visualisation devra présenter une liste exhaustive de tous les capteurs connectés au réseau, et renseigner leur localisation (nom du bâtiment, étage, pièce, coordonnées GPS, etc.). La liste des bâtiments, étages et salles sera récupérée dans un fichier de configuration (qui se trouvera dans le répertoire courant de votre programme). Elle possèdera un identifiant unique afin de l'identifier sur le réseau.

Ces informations seront affichées sous la forme d'un arbre. La racine de l'arbre différenciera les capteurs intérieurs des capteurs extérieurs.

Dans la catégorie des capteurs extérieurs ceux-ci seront ordonnés selon leurs coordonnées en latitude, puis en longitude. Ainsi, un capteur dont la localisation est : -45.2215 ; 0.0000 sera placé avant un capteur dont la localisation est : 12.9876 ; 0.0000. Cet ordre, une fois appliqué sur la latitude, sera appliqué sur la longitude. Ainsi, pour deux capteurs ayant la même latitude, celui dont la longitude est la plus petite sera disposé en premier.

Dans la catégorie des capteurs intérieurs, ceux-ci seront rangés dans les sous-catégories « bâtiment », puis « étage », et pour finir « Salle ». A l'intérieur d'une même salle, ils seront ordonnés par ordre alphabétique de leur identifiant. Ainsi, un capteur nommé « Lampe » sera placé avant un capteur nommé « Radiateur » s'ils se trouvent dans la même salle, au même étage du même bâtiment.

Cela signifie donc que l'arbre aura une profondeur de 2 niveaux côté extérieur (Extérieur > Capteurs rangés par coordonnées) alors qu'il aura une profondeur de 5 côté intérieur (Intérieur > Bâtiment > Etage > Salle > Capteurs rangés par ordre alphabétique).

De plus, lorsqu'un nœud de l'arbre sera sélectionné, un tableau permettra de suivre les mesures de l'ensemble des capteurs présents dans ce nœud de l'arbre. L'ensemble de ces données sera affiché sur l'interface sous la forme d'un tableau où chaque ligne du tableau correspondra à un capteur, reprenant son nom, le type de mesures, sa localisation et la valeur mesurée. L'utilisateur pourra filtrer les données temps réel soit en fonction du type de mesures, soit en fonction de la localisation des capteurs (filtre appliqué au tableau).

De plus, l'utilisateur pourra définir des alertes. Pour cela, il devra définir le type de mesure concerné et les bornes minimum et maximum à ne pas dépasser. Si une de ces bornes est atteinte, la ligne du tableau pour laquelle la valeur dépasse le seuil sera mise en évidence dans le tableau (par exemple, affichée en gras ou surlignée).

D'autre part, l'interface de visualisation devra proposer une page de visualisation de données *a posteriori* : c'est-à-dire, une page sur laquelle l'utilisateur pourra afficher l'ensemble des données au cours du temps pour un capteur en particulier. La représentation de ces données se fera sous la forme d'un graphique (que vous pouvez dessiner ou créer à partir d'une bibliothèque de fonctionnalités externe). Il devra donc être possible de sélectionner un capteur particulier (selon la même arborescence que celle décrite précédemment) pour afficher ses mesures. Les données du capteur seront chargées à partir d'un fichier de données (qui se trouvera dans le répertoire courant de votre programme).

Cela signifie que les données du fichier sont totalement isolées du réseau : il est possible d'avoir un capteur dans le fichier qui ne correspond à aucun capteur connecté sur le réseau à l'instant T. Cela n'empêche en rien de pouvoir visualiser les données antérieures de ce capteur enregistré dans le fichier.

Travail à réaliser

Chaque groupe de TD va réaliser une simulation de consommation énergétique du campus.

Dans chaque groupe de TD, vous devez vous répartir en équipes de 3 ou 4 étudiants. Chaque équipe devra réaliser une interface de simulation ainsi qu'une interface de visualisation. Toutes les interfaces réalisées au sein d'un groupe de TD devront pouvoir se connecter entre elles et constituer un même réseau de simulation. Ainsi chaque équipe pourra tester son interface de visualisation en recevant les données simulées par les interfaces de simulation de toutes les autres équipes du groupe de TD.

Travail à rendre

L'évaluation se fera sous la forme d'une soutenance par groupe de TD début Janvier.

En plus de cette soutenance, chaque équipe devra rendre :

- Un exécutable pour chacune de ces interfaces,
- Le code source de l'ensemble de son projet,
- Un rapport qui décrit le fonctionnement des interfaces, et explique les choix de conception ainsi que les différentes classes développées pour ce projet.