



## Contexte du projet et objectifs

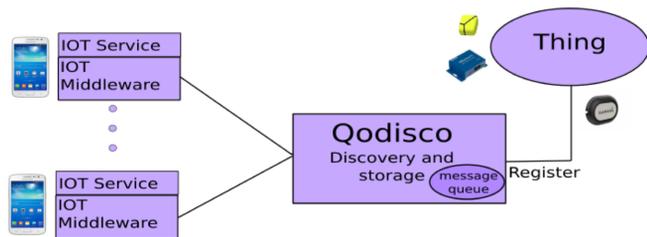
👁 Avec l'internet des objets, de multiples capteurs se retrouvent sur une même zone géographique.

- 👁 Des services veulent utiliser des valeurs instantanées de ces capteurs.
- Comment sélectionner les bonnes données selon des critères précis (localisation, type de capteur ...)
- Comment mettre à jour ces données en temps réel.

### Auteurs

Maxence Vandoolaeghe  
Rui Xiong

Encadrante:  
Chantal Taconet



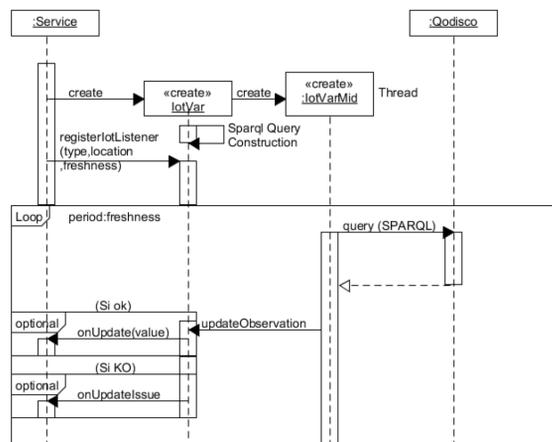
## Proposition

- 👁 Concept de « variable IOT ».
- 👁 Middleware IOTVAR met à jour ces variables
- ➔ Faciliter le développement des services.
- 👁 IOTVAR interagit avec une plateforme IoT pour sélectionner et obtenir les valeurs des variables IoT
- 👁 Plusieurs stratégies envisagées :
  - stratégie synchrone,
  - stratégie asynchrone,
  - stratégie avec filtrage sur le client ou le serveur.

## Résultats

### QoDisco (1) Middleware (stratégie synchrone)

Diagramme de séquence de la stratégie synchrone



### Utilisation de IOTVAR

```

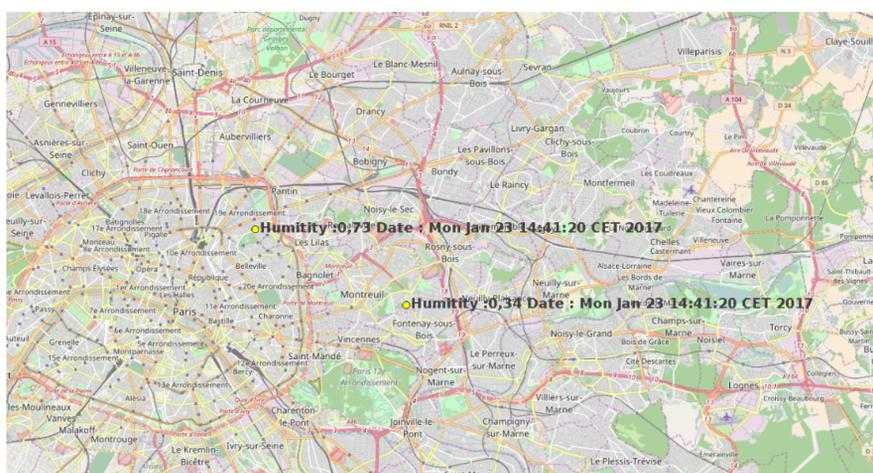
public class IoTHandlerDemo {
    public static void main(String[] args) {
        Display display = new Display();
        IoTVariable t1 = new IoTVariable("Temperature",
            new Location(48.8, 2.3, 10),
            new Freshness(1, TimeUnit.SECONDS));
        t1.registerIoTListener(display);
    }
}
    
```

- 👁 Nombre de lignes de code pour la gestion d'une variable IoT
- Avec IOTVAR : 3 lignes,
- Sans IOTVAR : 300 lignes
- 👁 Le développeur déclare une variable avec un type de capteur et une localisation --> le middleware l'associe ensuite à un capteur réel et se charge de mettre à jour automatiquement les données.



- 👁 Middleware d'interaction (stratégie synchrone) en deux versions
- 1. avec un handler de test (non connecté à une plateforme IOT)
- 2. avec le handler pour la plateforme Qodisco.

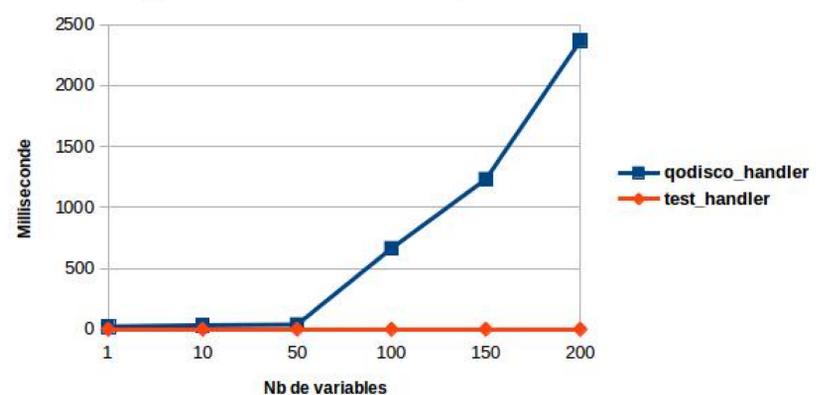
### (2) Démonstrateur



👁 Les performances dépendent de la plateforme IOT, de la fraîcheur attendue, et du nombre de variables.

### (3) Performances (stratégie synchrone)

Fraîcheur : 1sec  
Retard moyen de la fraîcheur avec un pool de 5 threads en 1 minute



## Travaux futurs

- 👁 Utiliser un service de découverte différent (OM2M)
- 👁 Utiliser des stratégies différentes, pour augmenter le nombre de variables supportées et limiter l'utilisation de la batterie
- 👁 Stratégie asynchrone, stratégie asynchrone filtrant les données.

👁 Déclaration de 2 variables IoT, et affichage des valeurs en temps réel avec 6 lignes de code.