

AKSEL

CAUBEL

RT2 IOM

ÉTENDRE

AVANT DE COMMENCER :

Durant le S3 nous avons eue un aperçu de l'IOT. Cette aperçu ainsi que mon expérience en entreprise ne me permettent pas encore de répondre réellement à ces compétences. Je vous y répondrai avec plus de précision durant mon S4 grâce à l'expérience LoRa que je devrais équerir.

COMPRENDRE LES ARCHITECTURES ET SPÉCIFICITÉS DES RÉSEAUX DÉDIÉS À L'IOT

//Explication de MQTT et la mise en place de refus des anonymes.

Avec LoRa comment les messages circulent et comment il communique avec au différent information du capteur//

Dans un réseau IoT, on vise à consommer le moins d'énergie possible. Dans ce but il nous faut transmettre le moins de données possibles avec des protocoles qui nécessitent peu de puissance à l'émission mais aussi qui possèdent des trames "utiles" dans le sens où tous les champs qui ne sont pas de la donnée sont réduits au maximum possible.

Pour répondre à cette différentes problématique il existe plusieurs manières de conceptualiser des architectures IoT.

Prenons l'exemple d'une architecture LoRaWAN :

Tout d'abord nous avons le LoRa. Il est la technique de modulation pour envoyer les informations d'un capteur. Les signaux émis sont réceptionnables par quiconque possède une Gateway LoRa. Une fois le signal LoRa capturé vient le LoRaWAN qui est le protocole qui va encapsuler le LoRa pour aller sur l'internet. Ce flux va ensuite être amené à un "Network Server" LoRa (port de fonctionnement 1700 tcp/udp) qui va permettre de décoder la trame grâce aux informations des capteurs. Ce dernier communique avec un "Application Server" qui va par la suite nous permettre de récupérer les informations grâce à plusieurs intégrations de données différentes. Pour ma part j'ai utilisé uniquement MQTT comme intégration de données pour sa simplicité (port de fonctionnement 1883 -> 8883 MQTT). À partir d'ici il faut un espace de stockage pour les données :

Et j'en viens à InfluxDB. Selon moi cette base de données est parfaite pour le stockage de données IoT. Nous possédons dans notre base de données des tables (buckets) avec comme clé primaire le temps, la possibilité de ne pas déclarer à l'avance notre structuration de données pour une flexibilité très importante.

Tout va marcher grâce à des filtres qui selon moi se marient très bien avec MQTT. MQTT fonctionne avec des "topics" afin de faire circuler les données et donc un sujet peut être directement lu dans la structuration d'une base de données Influx. par exemple ici :

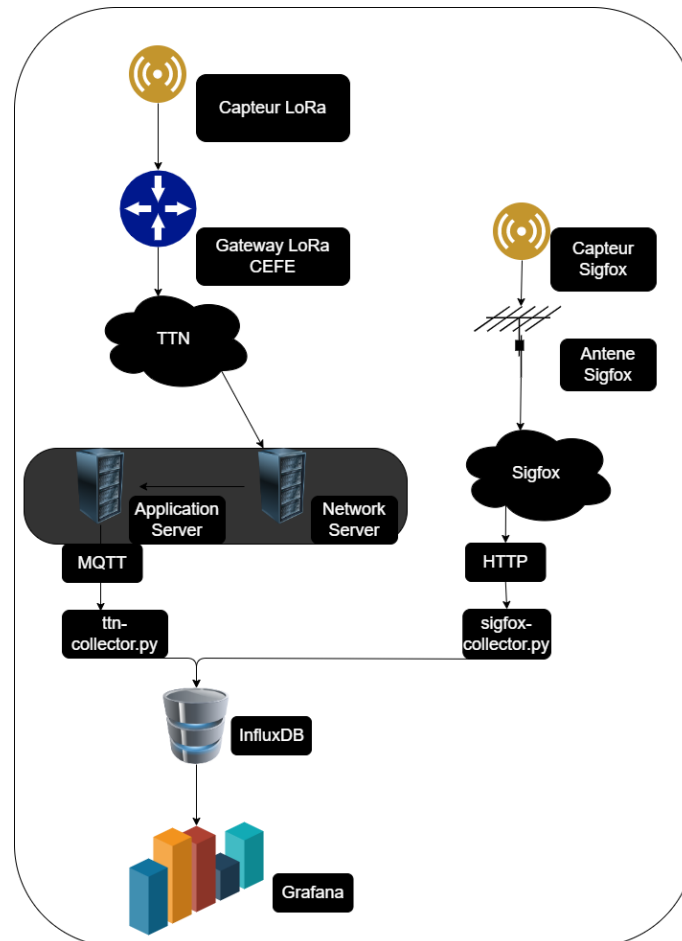
The image shows a screenshot of the InfluxDB query builder interface. It consists of five columns, each with a dropdown menu and a search bar. The columns are:
1. **_measurement**: Dropdown set to '1', search bar 'Search _measurement tag', list with 'esp' and 'sensor' (selected).
2. **_field**: Dropdown set to '1', search bar 'Search _field tag values', list with 'value' (selected).
3. **clientid**: Dropdown set to '1', search bar 'Search clientid tag values', list with '2' (selected).
4. **datatype**: Dropdown set to '1', search bar 'Search datatype tag values', list with 'co2' (selected), 'hum', and 'temp'.
5. **device**: Dropdown set to '1', search bar 'Search device tag values', list with '0090ec0116130c58' (selected) and '0090ec0116170e72'.

Dans cette arborescence InfluxDB servant au stockage de données de capteur de CO2 nous pouvons ressortir le topic MQTT qui contient l'id client, le type de données ainsi que le devEUI de notre capteur.

Pour finir notre architecture; L'interface Homme-Machine.

Il existe plusieurs façons de faire les représentations des données grâce à des graphiques simples avec l'outil Grafana ou bien des bibliothèques JavaScript comme chartJS.

Afin que vous ayez un visuel d'une architecture globale sous forme de schéma, je vous propose un schéma d'une application créée pour répondre à une problématique de bien-être animal dans un poulailler :



TTN étant The Things Network, un réseau LoRa Privée. le CEFE est l'emplacement où nous étions et les "ttn-collector.py" et "sigfox-collector.py" sont deux programmes développés par moi-même dans le but d'envoyer les données collectées par MQTT et par HTTP dans une base de données InfluxDB.

METTRE EN OEUVRE DES SYSTÈMES DE TRANSMISSIONS POUR L'ACCÈS À UN RÉSEAU IOT

- mise en service de la passerelle LoRa de la SAE 3I04 et de la passerelle Dragino LPS8N mise en place

Comme mise en œuvre d'un système de transmissions pour l'accès à un réseau IoT, je vous parlerai de la mise en place de GateWay LoRa.

En terme d'action, ces mises en place ont été très rapides et simples avec très peu de configuration à faire.

L'une des choses les plus importantes à faire est de savoir quels sont les ports à utiliser par notre application. Savoir les protocoles utilisés pour chacun en regardant si c'est du trafic sortant ou entrant. Une fois l'identification faite et les demandes d'ouvertures de port sur notre Firewall faites, nous pouvons commencer notre configuration et voir les résultats.

Lors de mon apprentissage au seins de la menuiserie Carayon, j'ai pu mettre en service deux GateWay LoRa (réf : Dragino LPS8-N).

La configuration de ce type de passerelle est très simple. Le paramétrage se fait directement sur une interface WEB.



Comme on le voit sur cette capture d'écran, la passerelle possède un point d'accès wifi qui nous permettra alors de faire la première configuration.

En se qui concerne la configuration LoRa, il nous suffit de choisir la bande de fréquence (ici 868MHz [Règlementation oblige])

Comme stipulé dans la compétence précédente; En se qui concerne LoRaWAN il faudra le faire pointer un Network Server qui fonctionne nativement sur le port 1700. Voir ci dessous :

Bien évidemment l'adresse de mon Network server est renseigné dans le champs (Server Address) mais pour des raisons de sécurité ce dernier n'apparaîtra pas.

Durant la pose d'un réseau LoRaWAN dans un poulailler j'ai également pu utiliser une passerelle Multi-Tech aillant des champs supplémentaires à renseigner :

Le Duty Circle qui permet de définir chaque combien de temps souhaitons nous que l'on puisse envoyer des informations à nos capteurs (grâce aux normes défini pour l'utilisation LoRa)