

Prototype Cayenne-NodeRED-InfluxDB-Grafana

I. Objectif:

Suite à la réussite d'envoyer les données à "Cayenne", on établit le prototype de communication entre Cayenne-NodeRED-InfluxDB-Grafana. Tous ces communications, on peut construire via NodeRED-un outil de développement basé sur les flux pour la programmation visuelle développé à l'origine par IBM pour connecter des périphériques matériels, des API et des services en ligne dans le cadre de l'Internet des objets.

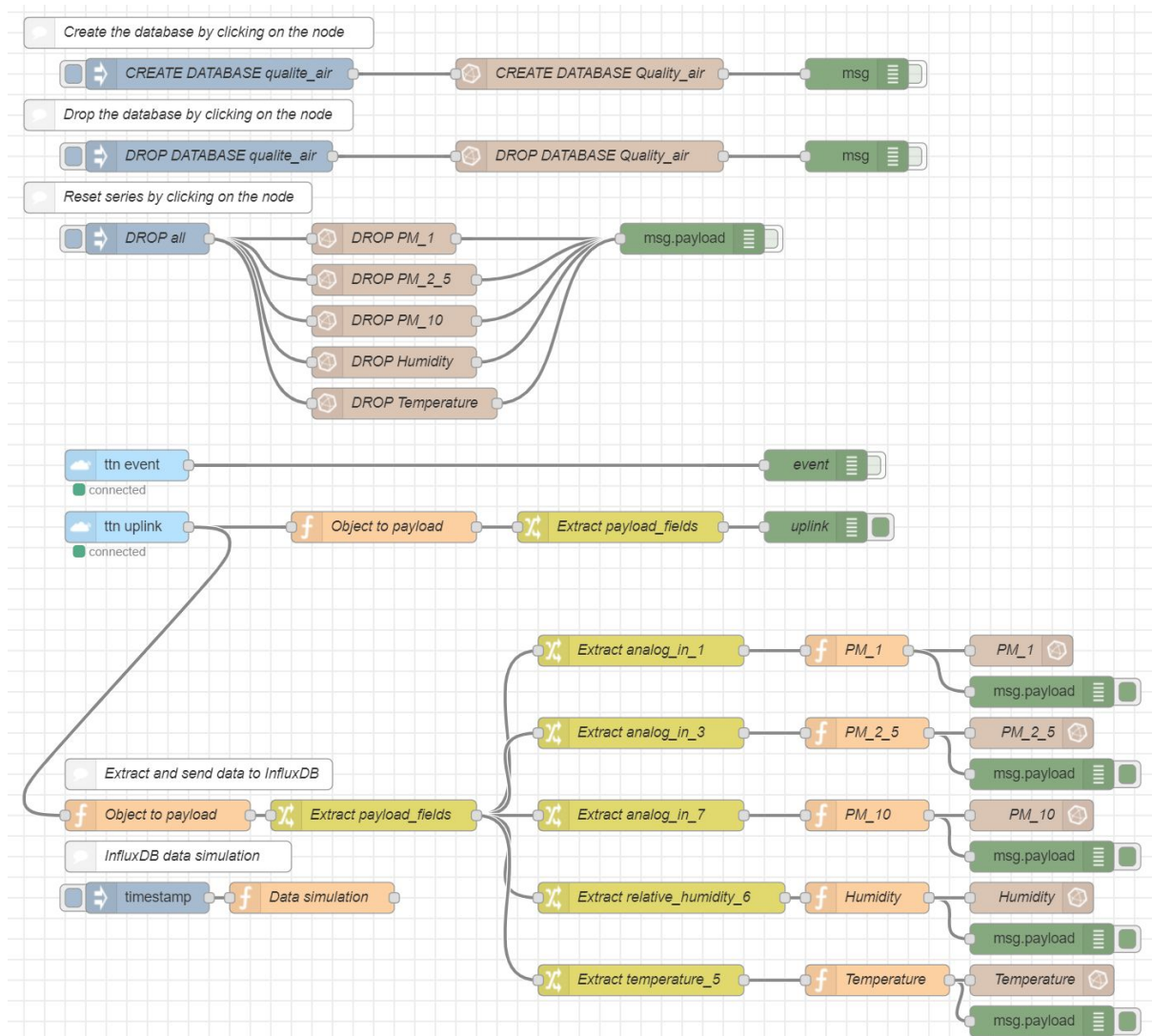


Figure I.1: Le flux complété

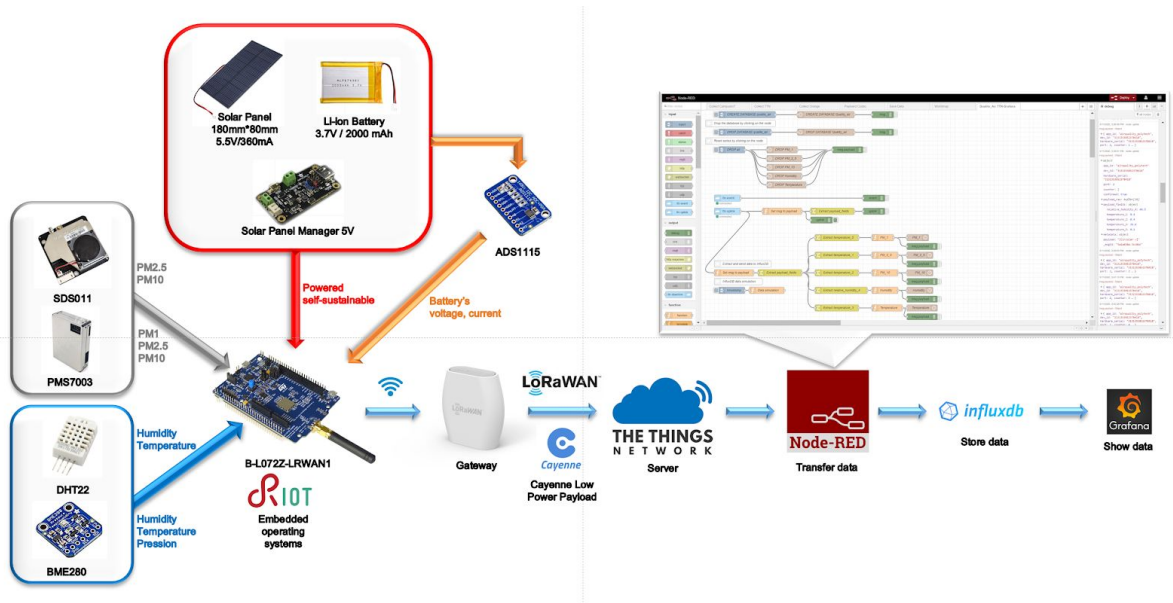


Figure 1.2: Le prototype de communication

II. La communication entre Cayenne et NodeRED:

Sur NodeRED, on utilise les noeuds “ttn”:

- ttn event: Un nœud pour recevoir les événements des appareils sur The Things Network.
- ttn uplink: Un nœud pour recevoir les messages de liaison montante des appareils sur The Things Network.
- ttn downlink: Un nœud pour envoyer un message de liaison descendante à un appareil sur The Things Network.

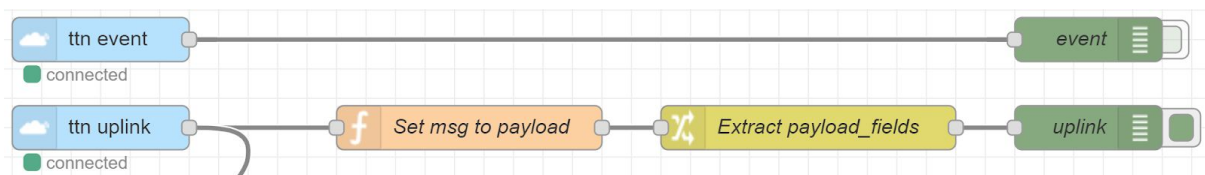


Figure II.1: Les noeuds de TTN

Configuration:

- **App ID:** airquality_polytech
- **Access Key:**
ttn-account-v2.mXRfK4rBu8-YdEVpAGzfu8BWizPFDAG8Z0hL6iETwSw
- **Discovery Address:** discovery.thethingsnetwork.org:1900
- **Device ID:** 3131353852378418

Sur le niveau de projet, on n’a que besoin d’utiliser le noeud “ttn uplink” pour récupérer les données.



Figure II.2: Le message reçu par le noeuds TTN

III. La communication entre NodeRED et InfluxDB:

On peut créer la base des données InfluxDB sur NodeRED en ajoutant les noeuds spécifiques qui permettent d'effectuer des requêtes de base sur une base de données de séries chronologiques influxdb. En configurant sur ces noeuds, on peut choisir entre créer ou annuler la base des données désirées.

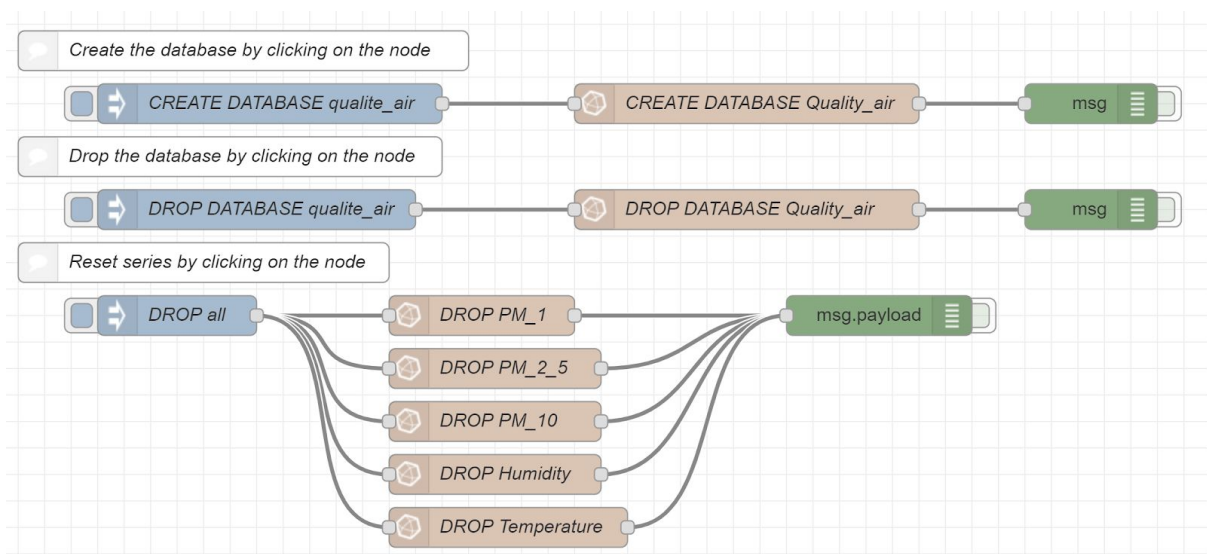


Figure III.1: Les noeuds d'InfluxDB

Configuration:

- **Host:** influxdb
- **Port:** 8086
- **Database:** Quality_air
- **Query:**
 - CREATE DATABASE "Quality_air" WITH DURATION 60d
REPLICATION 1 SHARD DURATION 1h NAME "sixty_days"

- DROP DATABASE "Quality_air"
- DROP MEASUREMENT PM_1

En cliquant sur les noeuds “CREATE DATABASE” et “DROP DATABASE”, on va recevoir les messages correspondants ci-dessous. Avec la même fonction, le noeud “DROP” permet de redémarrer les séries des données dans la base des données.

```
3/9/2020, 2:48:29 PM node: 9d58288f.035cb8
msg : Object
▼ object
  _msgid: "851c954b.6c08e8"
  topic: ""
  payload: array[0]

3/9/2020, 2:48:33 PM node: ff56dcde.3947f
msg : Object
▼ object
  _msgid: "ca29c51d.7a08b8"
  topic: ""
  payload: array[0]
```

Figure III.2: Les messages reçu quand on crée une base de données d’InfluxDB

Parce que le message qu’on reçoit est sous la forme “msg: Object” donc on a besoin d’utiliser le noeud de fonction pour changer le nom de message “Set msg to payload” et de là, on peut extraire les données qu’on veut grâce au noeud de changement “Extract payload_fields”. Une fois que le message est extrait, on reçoit le résultat comme ci-dessous:

```

3/9/2020, 12:02:33 PM  node: uplink
msg.payload : Object
▼ object
  analog_in_1: 0.57
  analog_in_3: 0.65
  relative_humidity_6: 37
  temperature_5: 22.3

3/9/2020, 12:02:34 PM  node: cb27168d.1f16a8
msg.payload : Object
▼ object
  PM_2_5: 0.57

3/9/2020, 12:02:35 PM  node: 33f7ca4b.c4e3d6
msg.payload : Object
▼ object
  PM_10: 0.65

3/9/2020, 12:02:36 PM  node: f02f8b73.cd10d8
msg.payload : Object
▼ object
  Humidity: 37

3/9/2020, 12:02:37 PM  node: 8c5fd2e5.17ae5
msg.payload : Object
▼ object
  Temperature: 22.3

```

Figure III.3: Les messages envoyés à la base de données d’InfluxDB

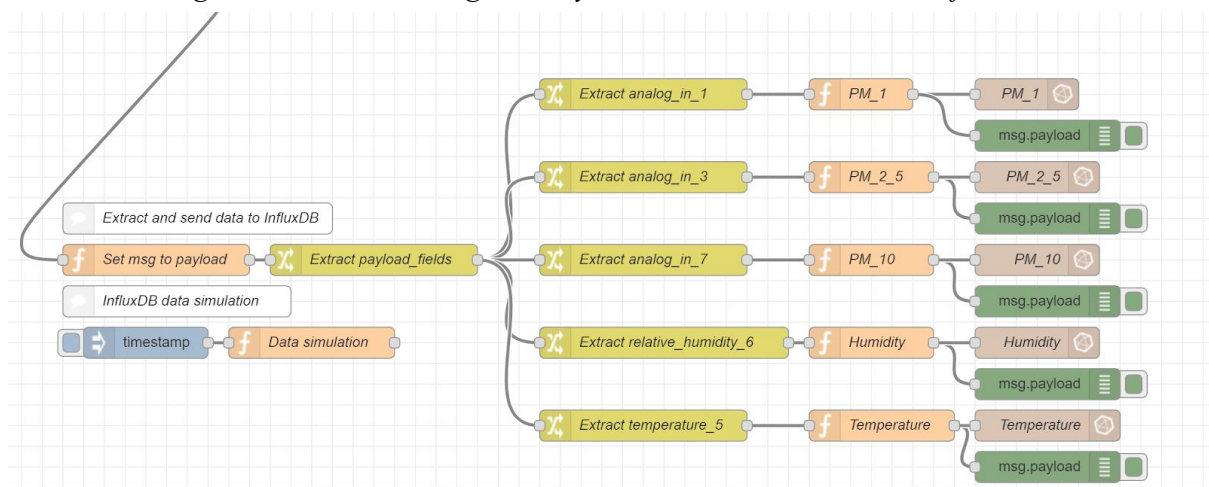


Figure III.4: Les noeuds qui envoient les messages à la base de données d’InfluxDB

Configuration:

- **Measurement:** PM_1_QA

Après, on extrait la valeur de chaque catégorie, et on construit pour chaque catégorie un message sous la forme d’“Object”. Ensuite, ces messages sont envoyés et sauvegardés sur la base des données “InfluxDB”, dans ce cas, la base des données est installé sur server d’<https://air-quality.iot.imag.fr/>.

IV. La communication entre InfluxDB et Grafana:

Sur Grafana, on ne doit pas installer Grafana sur le PC personnel car c'est déjà installé sur le server. En effet, on ne doit que créer un "Data Sources" pour récupérer les données dans la source de données InfluxDB et les afficher sur "Dashboard".

Ici, on crée une source de données "InfluxDB" qui contient une base de données "Quality_air". Une fois qu'on clique sur "Save and Test" et reçoit deux affichages "Database upload" et "Data source is working" qui sont bien validées, la base des données est activée.

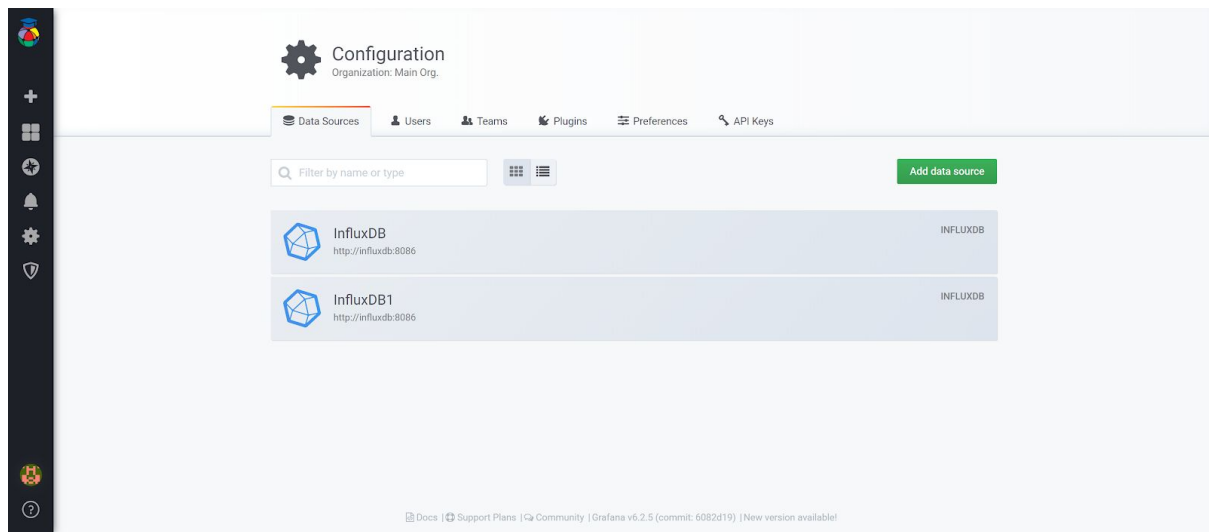


Figure IV.1: Les sources des données sur Grafana

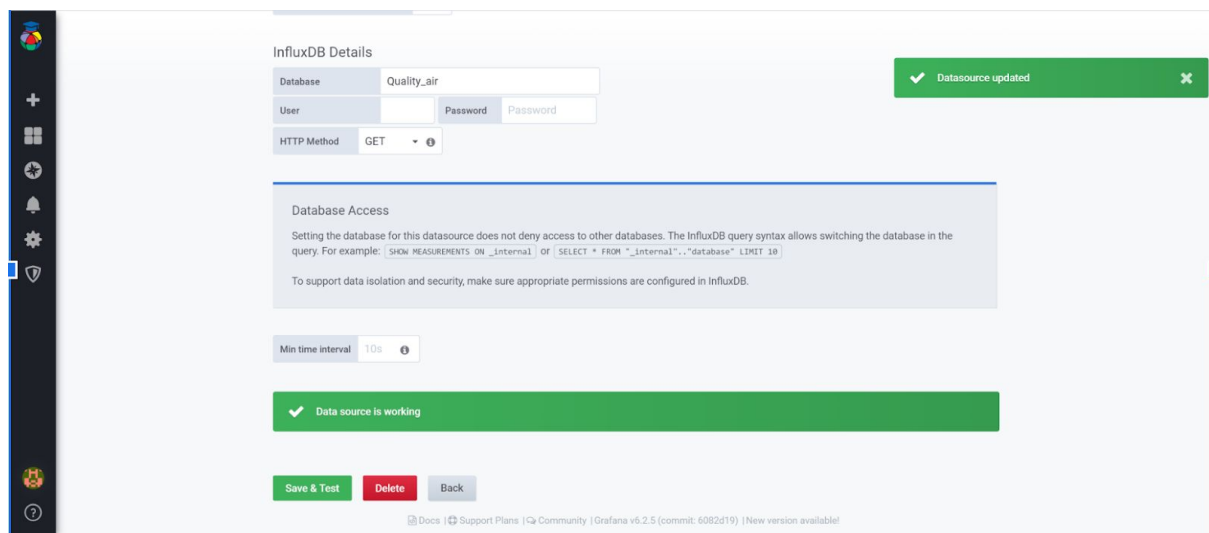


Figure IV.2: La création d'une source des données

Enfin, on crée un "Dashboard" pour afficher les données.

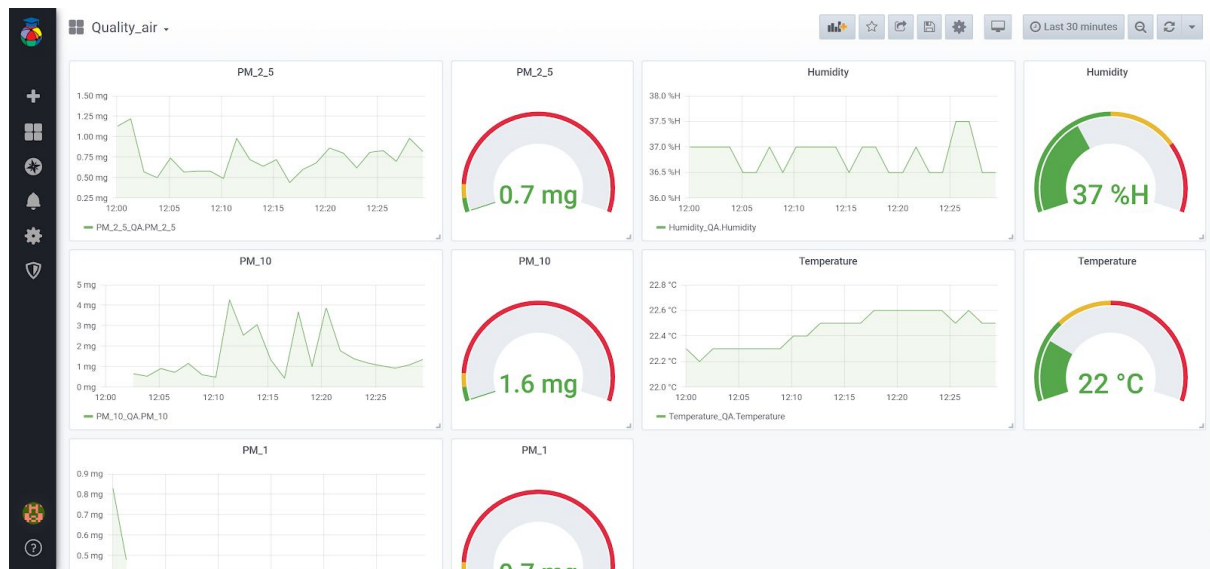


Figure IV.3: L'affichage de Grafana

V. Access:

Le dashboard Grafana qu'on utilise pour présenter les données de la station et de celles de Polytech (MCF88):

[https://air-quality.iot.imag.fr/](https://air-quality.iot.imag.fr/admin)
admin
E7534gx7Hp2TW679N34264

Pour personnaliser le flow Nodered en intégrant TTN:

[https://air-quality-n.iot.imag.fr/](https://air-quality-n.iot.imag.fr/admin)
admin
f723aAA33emV6jv