

The background features a dark blue gradient with a starry space pattern. On the left side, there are several technical diagrams, including circular gauges with numerical scales (140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260) and various circular arrows indicating flow or rotation.

IDÉE D'UN CLOUD POUR RÉCEPTIONNER LES DONNÉES D'IOT

GROUPE DE TRAVAIL IOT

15/12/2021

NAISSANCE DE L'IDEE

- Discussion entre les membres du GT réunissant
 - Electroniciens
 - Développeurs
 - ASRs
 - Instrumentalistes
- Besoin d'une chaine de traitement des infos remontées par les objets

NAISSANCE DE L'IDEE (2)

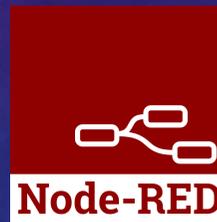
- Des guides d'installations existent
 - <https://funprojects.blog/2020/02/01/influxdb-with-node-red/>
 - <https://medium.com/@nicoletsylvain/how-to-monitor-a-thermal-sensor-with-a-raspberry-pi-node-red-influxdb-and-grafana-b4fa44155502>
- Des services existent
 - <https://fred.sensetecnic.com/>
 - <https://www.stackhero.io>
 - Azure et AWS via une installation manuelle

NAISSANCE DE L'IDEE (3)

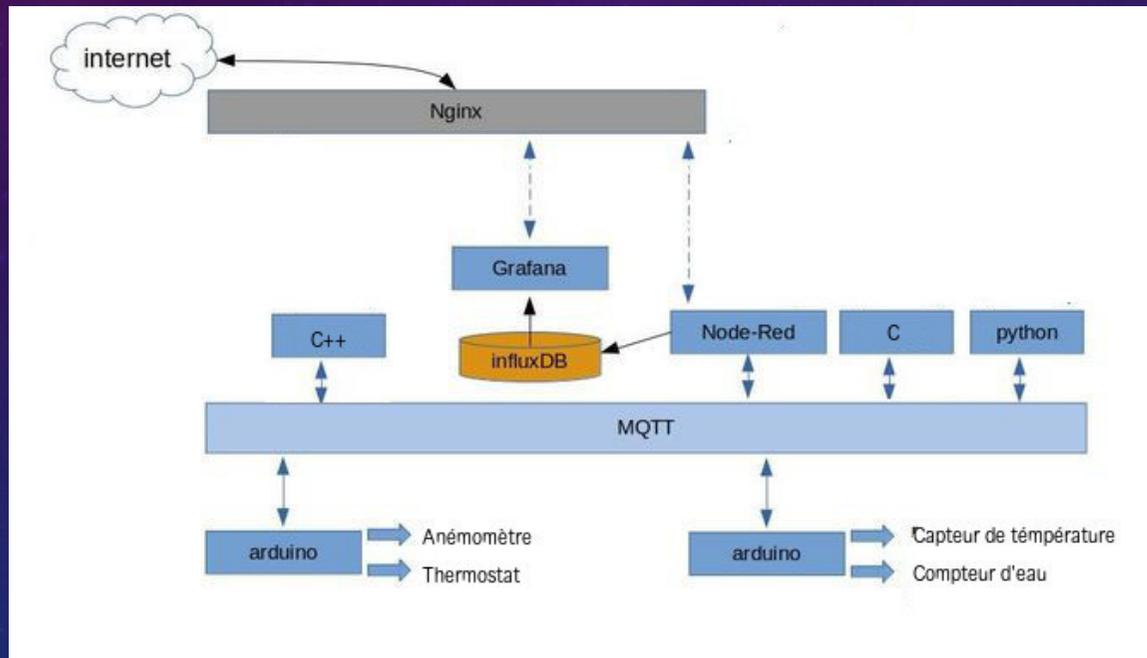
- Coût
- Contrôle des données
- Nécessite plus ou moins de compétences
- Pérénnité

IDENTIFICATION DU BESOIN

- Node Red
- MQTT
- InfluxDB
- Grafana
- Home Assistant



PRINCIPE DE FONCTIONNEMENT



POC : PREMIERE HYPTOHESE

- Hébergement sans container dans une VM
 - Installation de chaque application
 - Configuration de chaque application
- Fonctionnement ok

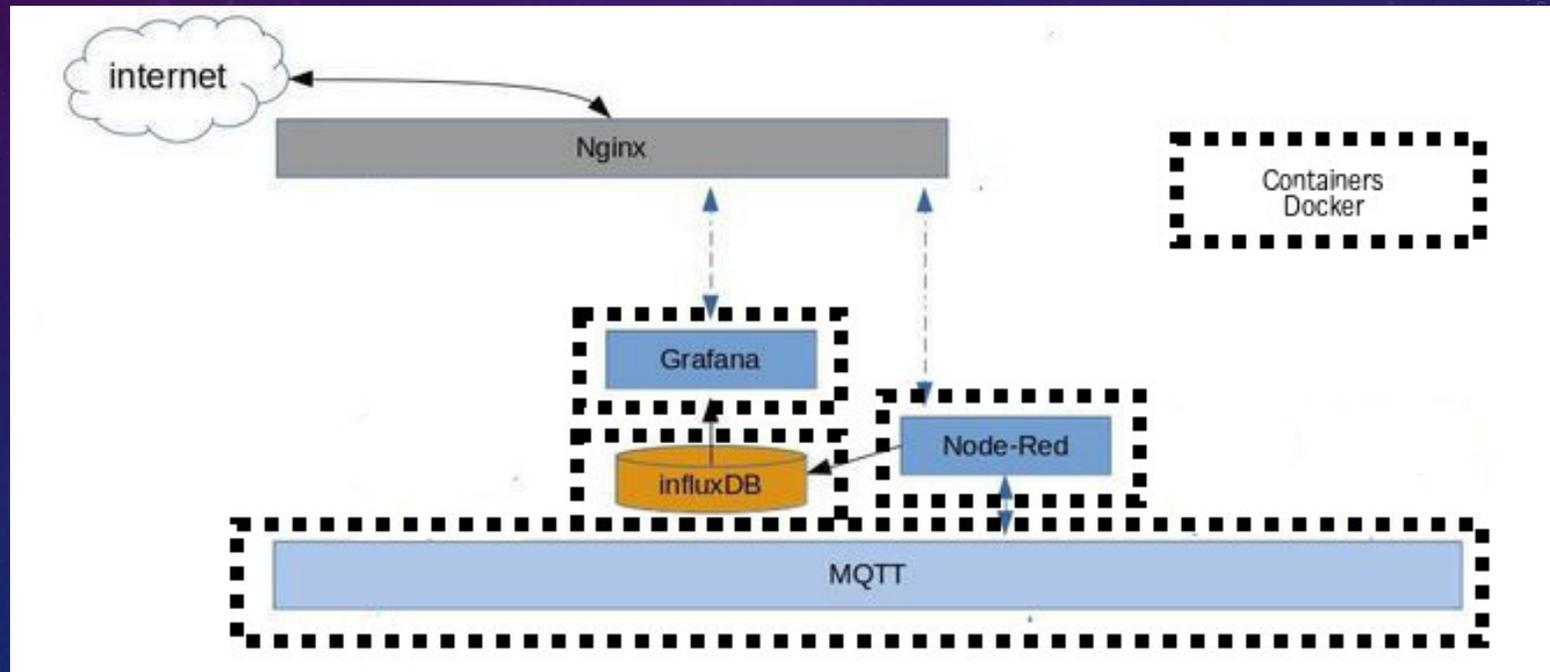
RÉSULTATS

Hébergement hors container n'est pas adapté à influxDB/Grafana et ne passe pas l'échelle

- Fonctionnel pour un seul projet/personnes
- InfluxDB/Grafana ne permettent pas de gérer la séparation entre utilisateur dans la version open source
=> Nécessite l'achat de la version Entreprise

POC : DEUXIEME HYPTOHESE

- Utilisation de Docker



ETAT ACTUEL

- Work in progress, POC
- Peu de test => Fonctionnement ok
- Un ensemble de container Docker clonable par projet
- Séparation des projets claires



QUESTIONS OUVERTES

- Ajout de homeassistant
- Contour du service
- Hébergement
- Sécurité
 - Usage de docker
 - Beaucoup de plugins
- Ressources allouées

QUESTIONS



REDNODE

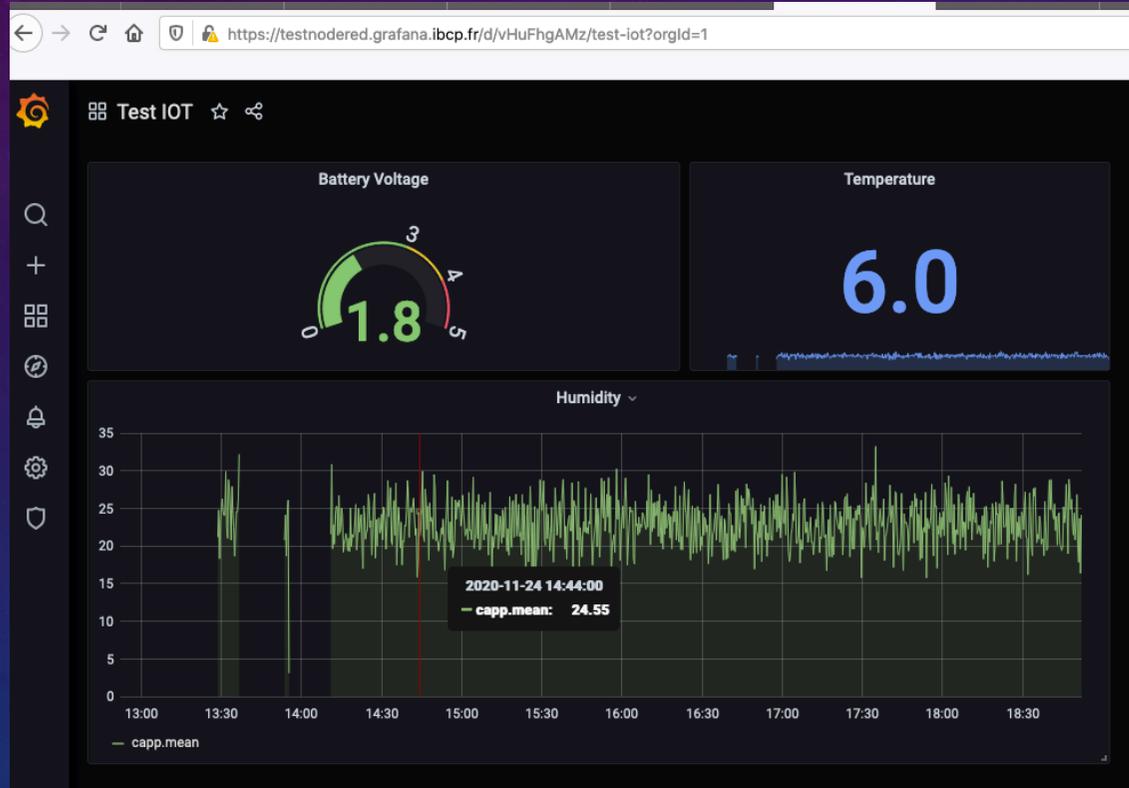
The screenshot displays the RedNode interface, which is used for creating and managing IoT flows. The main workspace shows a flow diagram with several nodes: a 'timestamp v' node, two 'testnod' nodes (one connected), a 'msg.payload' node, and three 'function' nodes. One function node is connected to the 'msg.payload' node and outputs to an InfluxDB node with the URL '[v1.8-flux] http://influxdb_001:8086 cap'.

The 'Edit function node' window is open, showing the following code:

```
1 - msg.payload = {
2 // You bucket
3 // Optional (it can be defined in the node credentials settings)
4 bucket: 'myBucket',
5
6 // Precision of timestamp
7 // Optional
8 // Can be `ns` (nanoseconds),
9 // `us` (microseconds),
10 // `ms` (milliseconds),
11 // `s` (seconds).
12 // The default is `ns`
13 precision: 'ns',
14
15 // Data to send to InfluxDB
16 // Can be an array of objects or only one object
17
18 measurement: 'testIOT',
19 deviceId: 'gyh43',
20 hardwareVersion: '1.0.2',
21 softwareVersion: '2.5.1',
22 location: 'factory-1',
23 temperature: 12*Math.random(),
24 humidity: 46*Math.random(),
25 vibrations: 18*Math.random(),
26 batteryVoltage: 3.6*Math.random(),
27 timestamp: Date.now()
28
29 - };
30
31 return msg;
```

The debug console on the right shows a series of log entries, each containing a timestamp, a node ID, and a message payload. The messages are strings representing the data being sent to InfluxDB, such as: "1606223511850", "1606223512852", "1606223513853", "1606223514853", "1606223515854", "1606223516856", "1606223517856", "1606223518857", "1606223519858", and "1606223520859".

GRAFANA



INFLUXDB (CHRONOGRAF)

