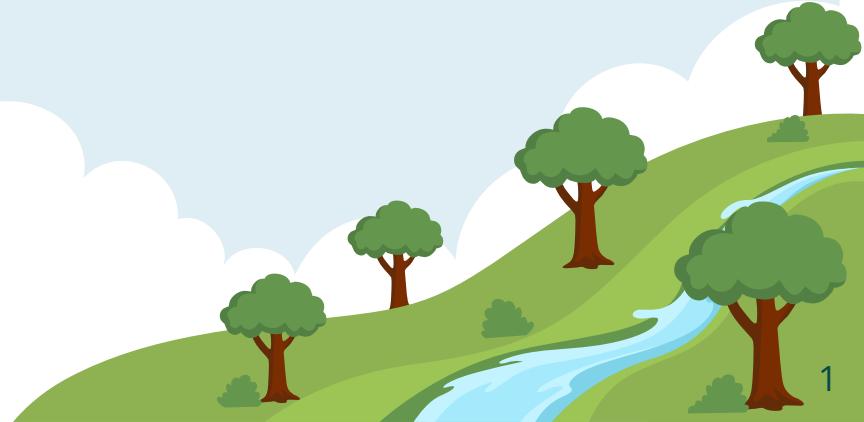


Mycelium 3.0



Le projet Mycélium



Problématique



Arrivée de la ligne B du métro

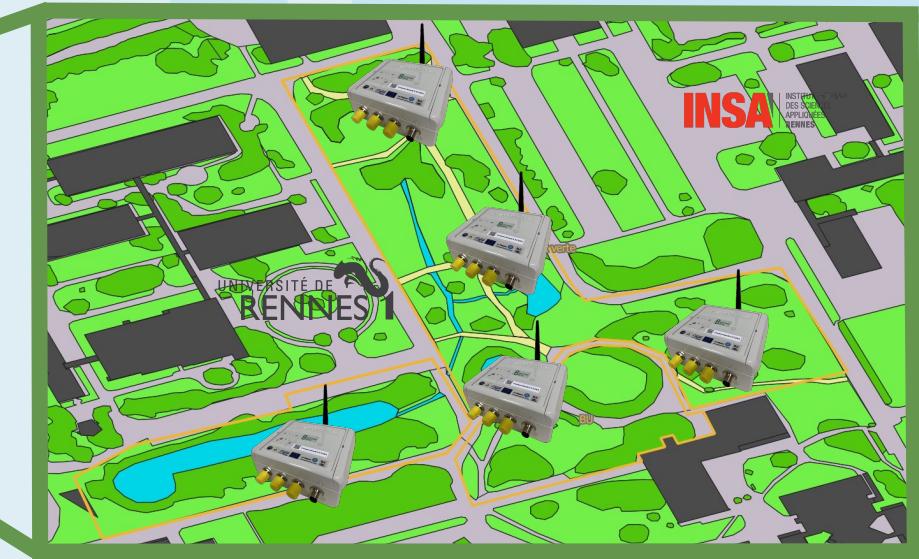


Travaux importants



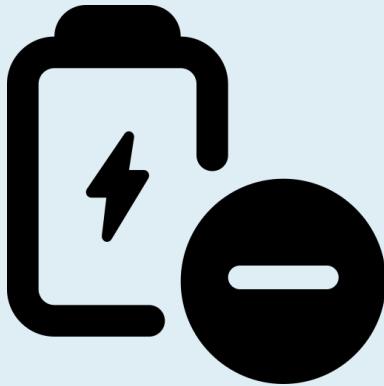
Dégâts sur l'environnement

Renaturation de la Croix Verte

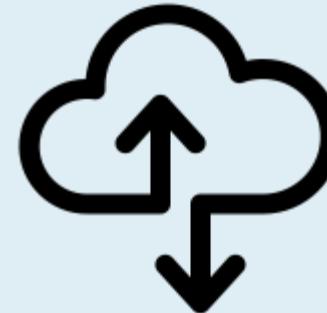


La croix verte à Beaulieu

Contraintes



- Besoin de consommation faible



- Limiter les transferts de données inutiles

Schéma de principe Mycélium

Traitement

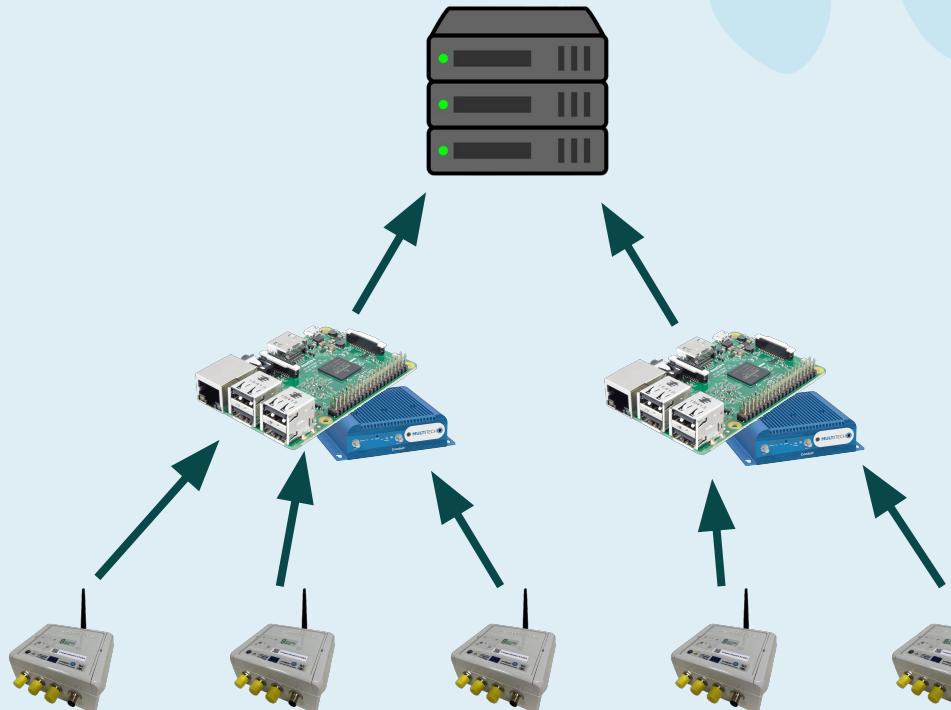
Serveur de l'INSA

Traitement
intermédiaire

Raspberry Pi
Gateway

Capture des
données

Capteurs



L'équipe :

Valentin CHEREL



Maïwenn LE GOASTELLER



Pierre OUVRARD



Pierrick POURCHER



Thomas DERRIEN



Léo LESSIRARD



Arthur MAUGER

Nos encadrants



Nikos PARLAVANTZAS Volodia PAROL-GUARINO

Enseignant-chercheur à
l'INSA Rennes et à l'IRISA

Doctorant à l'INRIA



Julien MOUREAU

Ingénieur d'études en
instrumentation à l'OSUR

Sommaire

01

Objectifs

02

Travail Réalisé

03

Phase de Tests

04

Difficultés

05

Planification

06

Conclusion

01

Objectifs



Projets précédents

Mycelium 1.0 et 2.0



Mise en place de
l'architecture



Choix des
technologies



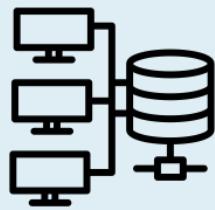
Ajouts de
fonctionnalités



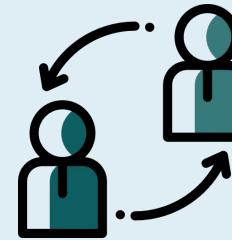
Premiers
essais

Objectifs

Mycelium 3.0



Améliorer l'architecture



Faciliter la prise en main

Amélioration de l'architecture

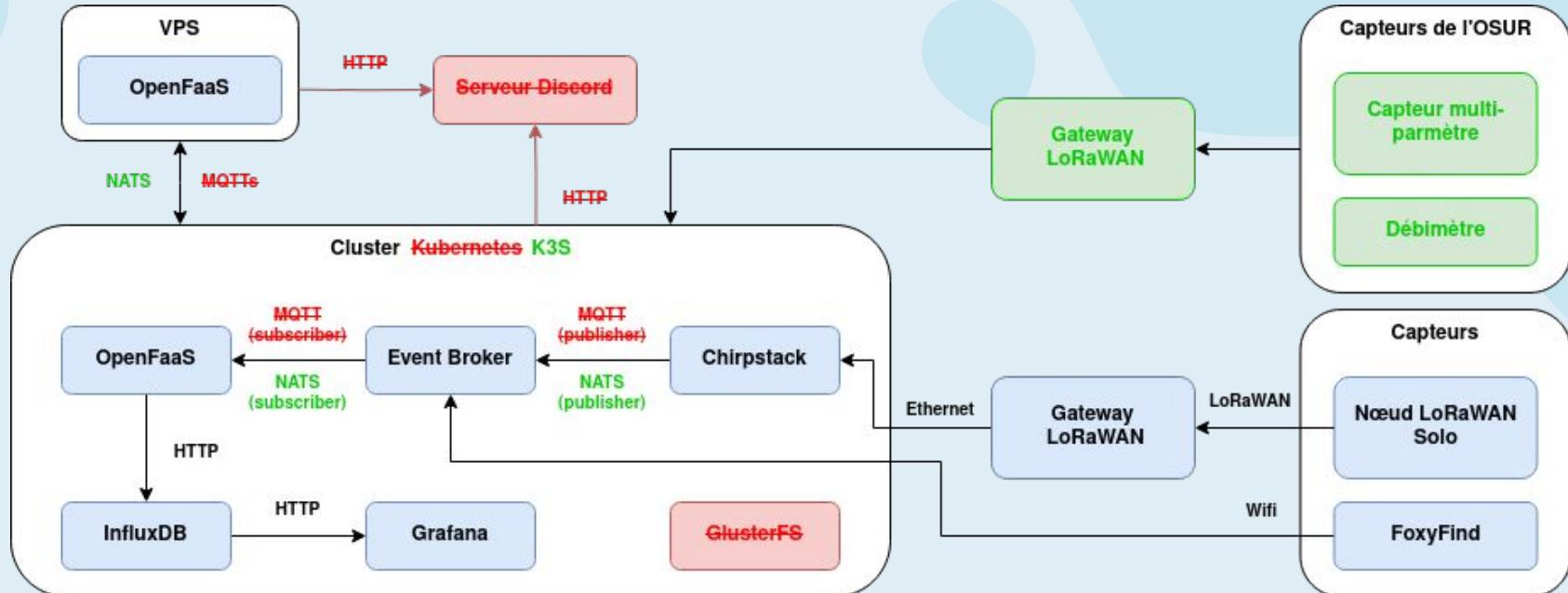
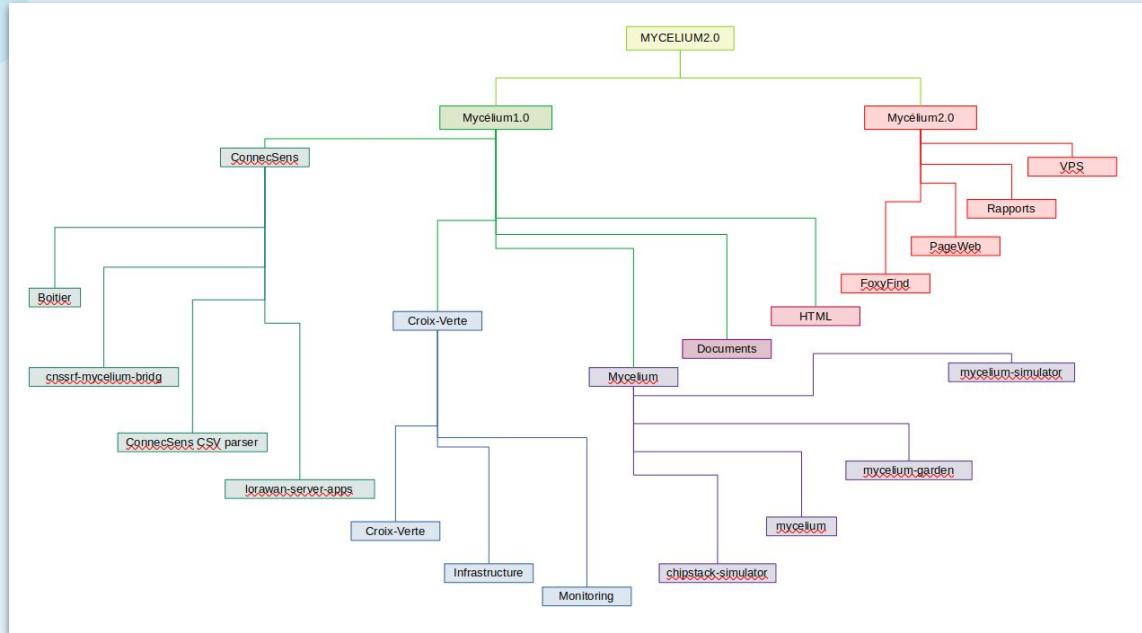


Schéma des changements prévus de l'architecture

→ Objectif : Améliorer l'architecture, remplacer des technologies

Facilitation de la prise en main



→ Objectif : Faciliter de la prise en main

22 dépôts Git

Facilitation de la prise en main



Absence de guides détaillés pour Mycélium 1 et 2



Conséquences : Perte de temps lors de la reprise du projet, pas d'indications sur où trouver les informations

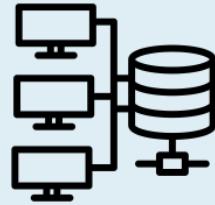
Notre objectif: Permettre une reprise rapide du projet et une documentation détaillée en cas de problème.

02

Travail réalisé



Amélioration de l'architecture

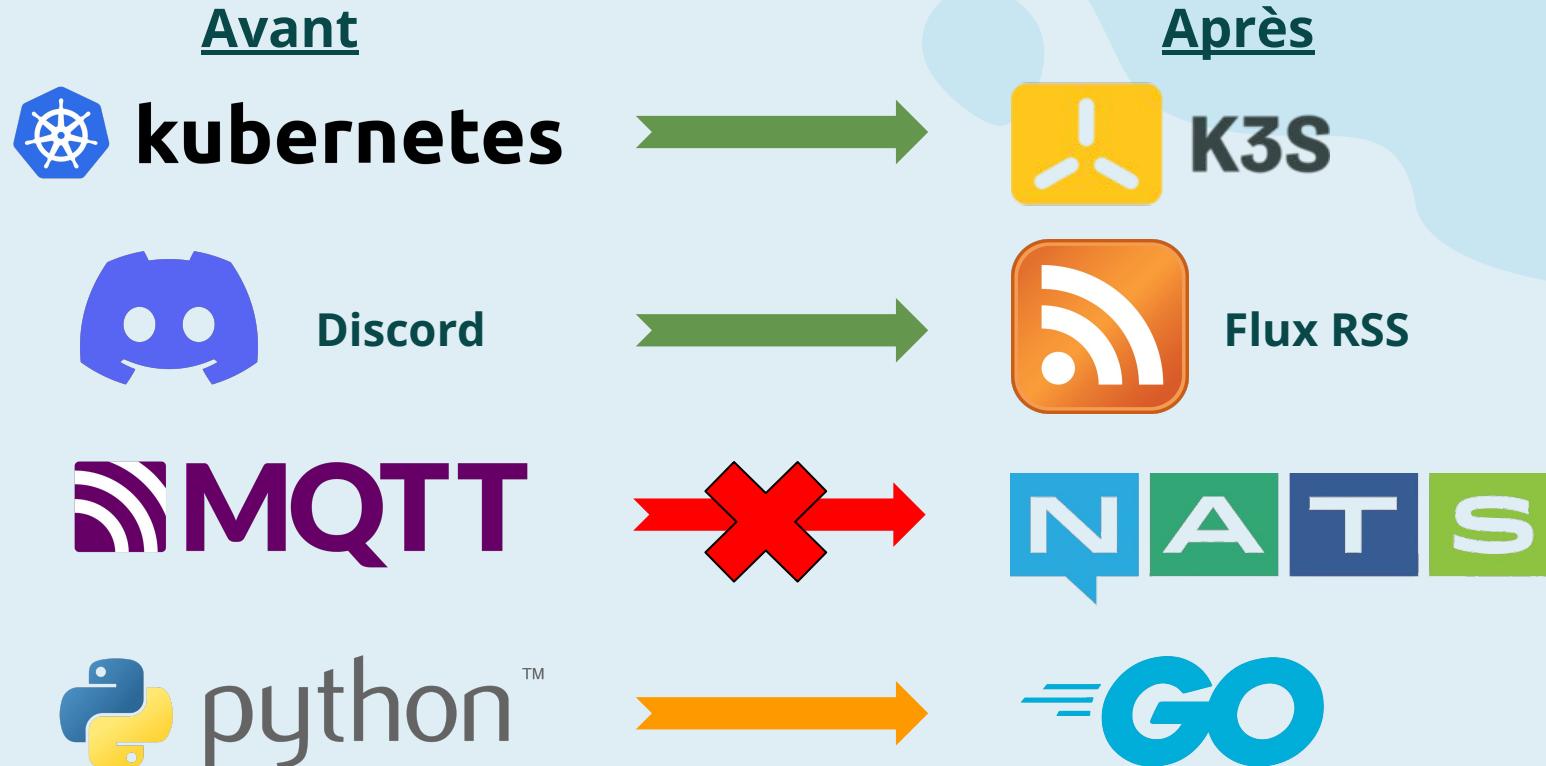


Améliorer l'architecture

Remplacement de technologies

Architecture repensée

Remplacement des technologies



Nouvelle architecture

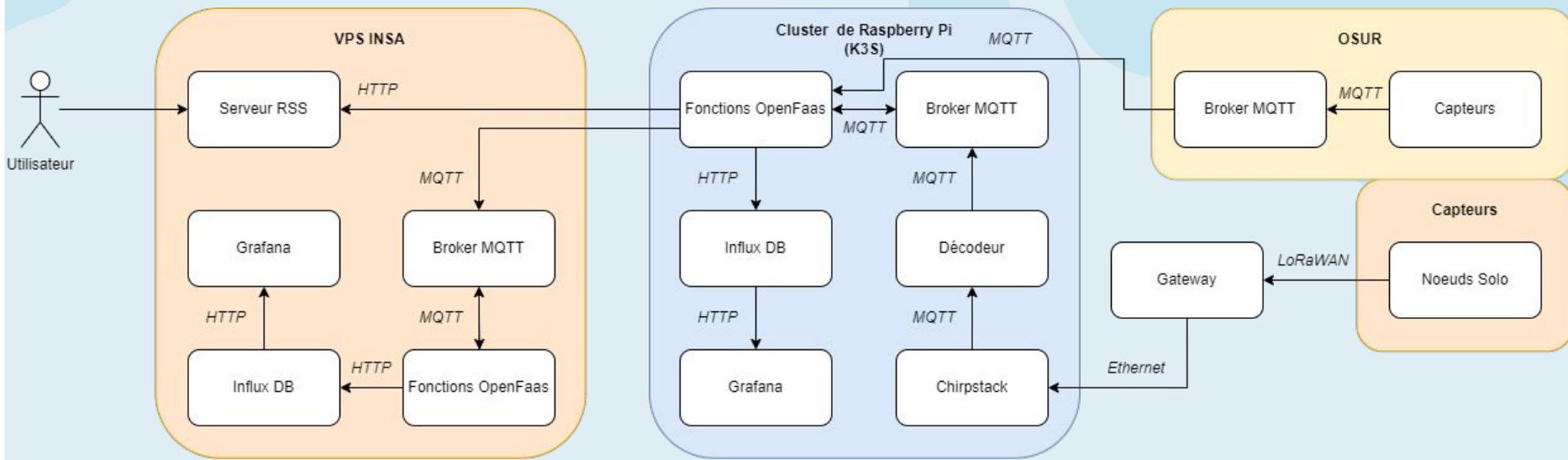
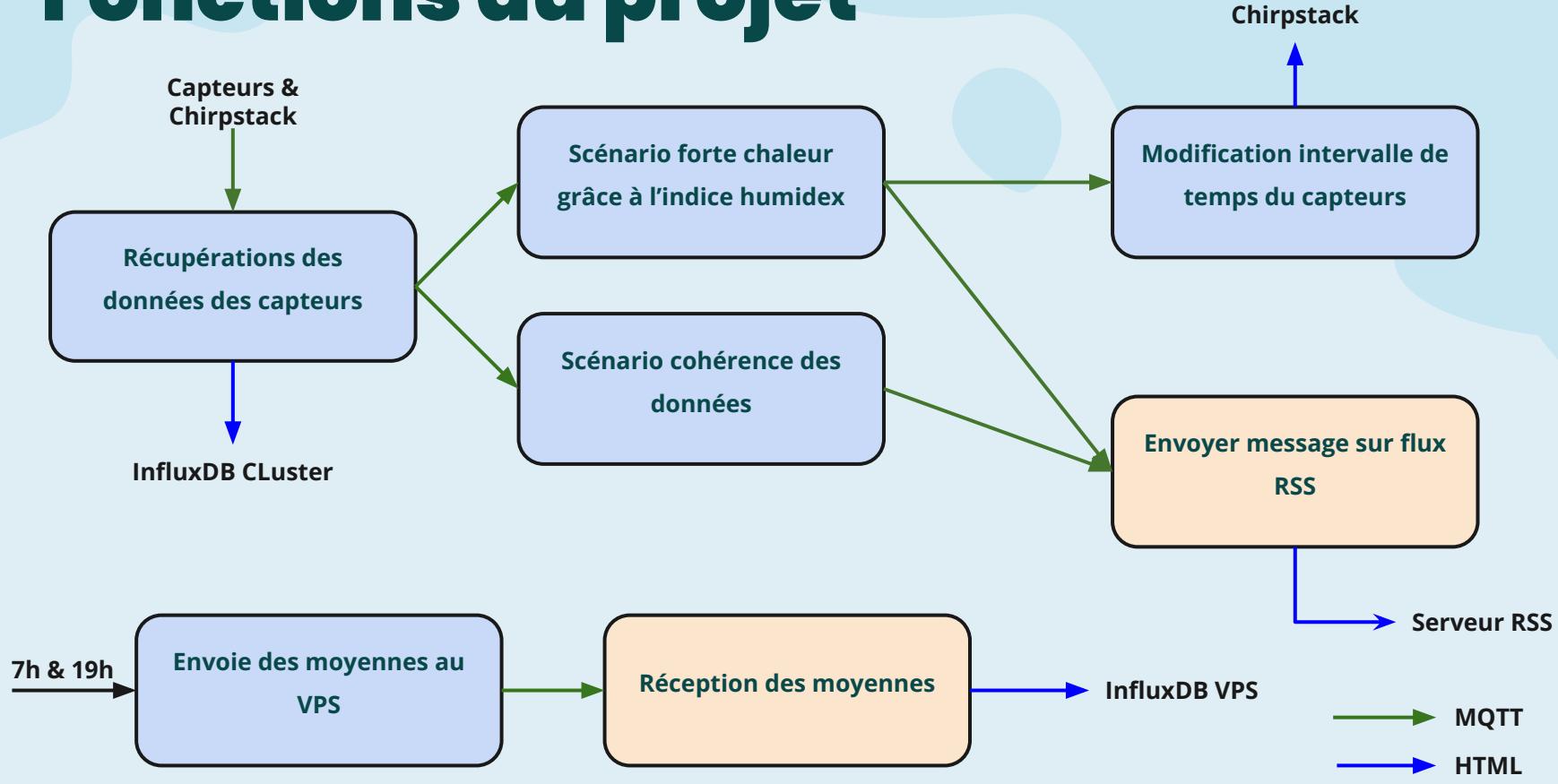
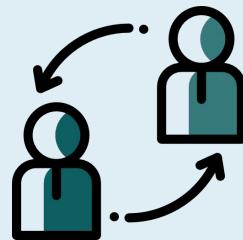


Schéma de l'architecture de Mycélium 3.0

Fonctions du projet



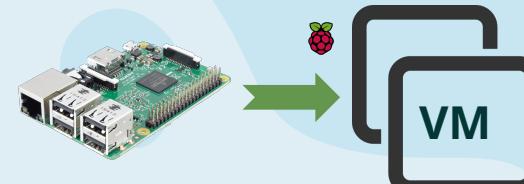
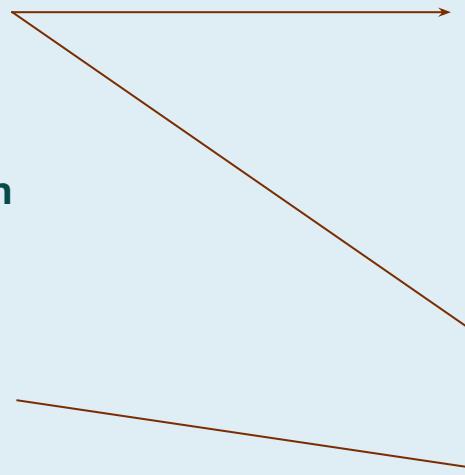
Faciliter la prise en main



Faciliter la prise en main



Faciliter la maintenance



Réalisation d'une machine virtuelle

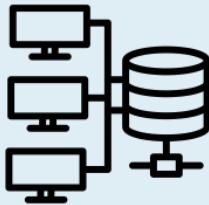


Écriture d'un guide d'utilisation détaillé



Centralisation de la documentation

Au final ?



- Nouvelle Architecture fonctionnelle



- Récupération de données extérieures (OSUR)



- VPS Fonctionnel



- Un scénario complet



- Prise en main et maintenance bien plus facile



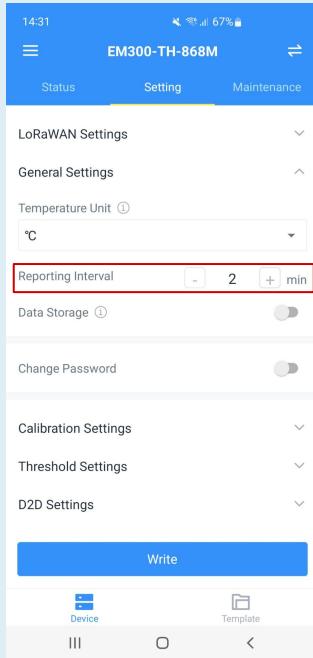
- Interface utilisateur

03

Phase de tests



Fonctionnalités



```
user@mycelium3:~/Mycelium3.0/Fonctions/Fonctions-Extraction$ faas-cli logs chaleur
2024-05-05T09:08:25Z Pour une température de 17.7°C et une humidité de 59.0%, l'humidex est 18.8.
2024-05-05T09:08:25Z Sensation : Sensation de confort
2024-05-05T09:08:25Z Message publié sur le topic notification: {"title":"humidex","description":"Temperature : 18°C ,Humidité : 59% ,Humidex : 19 , Sensation : Sensation de confort"}
2024-05-05T09:08:27Z 2024/05/05 09:08:27 POST / - 200 OK - ContentLength: 0B (2.0085s)
2024-05-05T09:10:25Z Pour une température de 17.8°C et une humidité de 59.0%, l'humidex est 18.9.
2024-05-05T09:10:25Z Sensation : Sensation de confort
2024-05-05T09:10:25Z Message publié sur le topic notification: {"title":"humidex","description":"Temperature : 18°C ,Humidité : 59% ,Humidex : 19 , Sensation : Sensation de confort"}
2024-05-05T09:10:27Z 2024/05/05 09:10:27 POST / - 200 OK - ContentLength: 0B (2.0113s)
```

Traitement des données environnementales reçues avec OpenFaas



```
pierre@po-dell-g3-3500:~$ mosquitto_sub -h 10.133.33.52 -p 1883 -t notification
{"title":"humidex","description":"Temperature : 18°C ,Humidité : 58% ,Humidex : 19 , Sensation : Sensation de confort"}
{"title":"Cohérence données EM300-TH","description":"Valeurs non cohérente ! Vérifiez capteur !"}
{"title":"humidex","description":"Temperature : 18°C ,Humidité : 58% ,Humidex : 19 , Sensation : Sensation de confort"}
 {"title":"Cohérence données EM300-TH","description":"Valeurs non cohérente ! Vérifiez capteur !"}
 {"title":"humidex","description":"Temperature : 18°C ,Humidité : 58% ,Humidex : 19 , Sensation : Sensation de confort"}
 {"title":"Cohérence données EM300-TH","description":"Valeurs non cohérente ! Vérifiez capteur !"}
```

```
xml version='1.0' encoding='UTF-8'?
<rss xmlns:atom="http://www.w3.org/2005/Atom" xmlns:content="http://purl.org/rss/1.0/modules/content/"
version="2.0">
<channel>
<title>Mycelium 3.0</title>
<link>https://projets-info.insa-rennes.fr/projets/2022/Mycélium_2.0/</link>
<description>Flux RSS du projet Mycelium 3.0</description>
<docs>http://www.rssboard.org/rss-specification</docs>
<generator>python-feedgen</generator>
<language>fr</language>
<lastBuildDate>Sat, 04 May 2024 21:27:28 +0000</lastBuildDate>
<pubDate>Sat, 04 May 2024 21:27:28 +0000</pubDate>
<item>
    <title>Cohérence données EM300-TH</title>
    <description>Valeurs non cohérente ! Vérifiez capteur !</description>
    <pubDate>Sun, 05 May 2024 09:04:29 +0000</pubDate>
</item>
<item>
    <title>humidex</title>
    <description>Temperature : 18°C ,Humidité : 59% ,Humidex : 19 , Sensation : Sensation de
confort</description>
    <pubDate>Sun, 05 May 2024 09:04:25 +0000</pubDate>
</item>
<item>
    <title>Données EM500-CO2 OSUR</title>
    <description>Temperature: 14.8°C, Humidity: 82.5%, Pressure : 1007.4hPa, CO2 :
487.0ppm</description>
    <pubDate>Sun, 05 May 2024 09:03:08 +0000</pubDate>
</item>
```

Émission de flux RSS



Stockage des données dans InfluxDB puis visualisation avec Grafana

Guide

4) Commandes utiles :

- Open Faas :

Voir si le cluster a bien démarré :	<code>\$ kubectl get deployments -n openfaas -l "release=openfaas, app=openfaas"</code>
Vérifier si la gateway est bien déployée :	<code>\$ kubectl rollout status -n openfaas deploy/gateway</code>
Voir les que la redirection est bien activée (une ligne avec <i>Running</i> devrait s'afficher) :	<code>\$ jobs</code>
Détails sur un service :	<code>\$ kubectl get service <name-of-service> -n openfaas</code>
Supprimer les pods inutiles :	<code>\$ kubectl get po -A --all-namespaces -o json jq '.items[] select(.status.reason!=null) select(.status.reason contains("Evicted"))' "kubectl delete po \(.metadata.name) -n \(.metadata.namespace)"' xargs -n 1 bash -c</code>
Affiche tous les pods du cluster :	<code>\$ kubectl get pods -A</code>
Informations détaillées sur les noeuds :	<code>\$ kubectl describe nodes</code>
Affiche des détails spécifique au pod XXX :	<code>\$ kubectl describe pod XXX -n openfaas</code>
Liste des déploiements du cluster :	<code>\$ kubectl get deployments -A</code>
Supprime le déploiement XXX :	<code>\$ kubectl delete deployment XXX -n openfaas</code>

I. Guide VM	2
1) Installations :	2
2) Créer une VM :	2
3) Se connecter en SSH à la VM :	2
II. Guide Kubernetes (K3S)	3
1) Installation de K3S sur la main :	3
2) Installation de K3S sur les agents :	3
3) Commandes optionnelles :	3
III. Guide Docker	4
1) Installation de Docker :	4
2) Commandes utiles :	4
IV. Guide Open Faas	5
1) Installations :	5
2) Préparation du cluster :	6
3) Création et modification de fonctions OpenFaas :	6
4) Commandes utiles :	8
5) Exemples de fonctions :	9
V. Guide MQTT	10
1) Création du serveur MQTT :	10
2) Test du serveur MQTT :	10
3) Méthode pour ajouter plusieurs topics :	11
VI. Guide K9s	12
1) Installations :	12
2) Commandes utiles :	12
VII. Guide Influx DB	12
VIII. Guide ChirpStack	14
1) Configurer Chirpstack gateway bridge :	14
2) Ajouter une gateway :	14
3) Ajouter un capteur sur Chirpstack :	15
IX. Guide Gateway	15
1) Connexion Gateway-Cluster :	16
2) Configuration de la gateway Multitech :	16
3) Connection Gateway-VM :	17
4) Configurer la connexion sur la VM :	17
X. Guide Cluster	18
1) Scan adresses cluster :	18
2) Manips tailscale (possiblement pas nécessaires) :	19
XI. Guide VPS	20
XII. Guide Git avec SSH	20
XIII. Commandes à relancer si besoin à chaque lancement de la VM	20

04

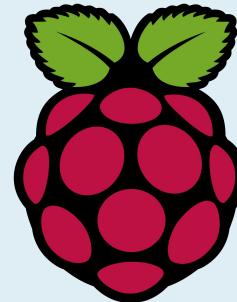
Difficultés



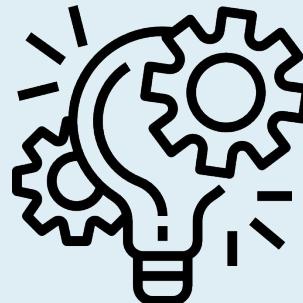
Difficultés



Difficultés
avec le VPS



Limites du
cluster

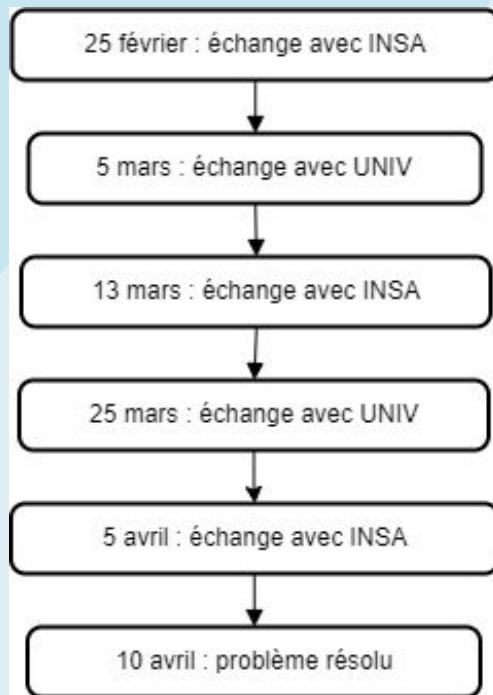


Technologies peu
documentées



Difficultés de
communication
avec les DSIs

Communications avec les DSIs



Évolution des échanges avec les DSIs

- Chaque fois, il faut **plusieurs semaines** pour que le problème soit résolu
- Ces problèmes sont souvent **bloquants** et ralentissent grandement le développement

05

Bilan Planification



Organisation du groupe

2 groupes thématiques flexibles



- Équipe Capteurs / ChirpStack



- Équipe Cluster / VM



Rapports : Cycles en V A green checkmark icon inside a rounded square border.

Partie technique : Agile A red X icon inside a rounded square border.

Organisation du projet

Frise chronologique



Novembre Décembre

Janvier

Février

Mars

Avril

Rapports et organisation du travail (20/11/23 - 29/04/24)

Capture des données (23/11/23 - 11/04/24)

Traitements des données (27/11/23 - 26/03/24)

Communication (19/12/23 - 08/04/24)

Lien avec l'utilisateur (26/02/24 - 01/04/24)

06

Conclusion



Conclusion

Points clés :

- Formation sur des technologies d'avenir
- Projet pluri-disciplinaire complexe
- Collaboration avec l'OSUR

Pour l'avenir:

- Implémentation de nouvelles fonctions
- Amélioration de l'interface utilisateur
- Utilisation du cluster
- Enregistrement automatique de nouveaux capteurs



**Merci pour
votre écoute**



Récapitulatif

Objectifs :

- Simplifier l'architecture
- Faciliter la prise en main
- Faciliter la maintenance

Points clés :

- Formation sur des technologies d'avenir
- Projet pluri-disciplinaire complexe
- Collaboration avec l'OSUR

Difficultés :

- Manipulation des technologies
- Prise en main du projet
- Communication avec la DSJ





Annexes



Frise chronologique Capture des données

Novembre Décembre

Janvier

Février

Mars

Avril

Capture des données (23/11/23 - 11/04/24)

Mise en place de la capture des données (23/11/23 - 01/12/23)

Système d'identification (01/12/23 - 08/12/23)

Système de localisation des capteurs (01/12/23 - 06/12/23)

Simplification de l'ajout de nouveaux capteurs (18/01/24 - 16/02/24)

Rajout du capteur FoxyFind (28/03/24 - 11/04/24)

Frise chronologique Traitement des données

Novembre Décembre

Janvier

Février

Mars

Traitement des données (27/11/23 - 26/03/24)

Mise en place de NATS sur la VM (27/11/23 - 13/12/23)

Développement de fonctions en Go (27/11/23 - 11/12/23)

Mise en place de la BDD (18/12/23 - 17/01/24)

Mise en place d'un flux RSS (18/12/23 - 21/12/23)

Test global de l'architecture sur la VM (22/01/24 - 25/01/24)

Basculer depuis la VM vers le cluster de Raspberry Pi (08/02/24 - 20/02/24)

Test global de l'architecture sur le cluster (20/02/24 - 23/02/24)

Suppression de GlusterFS (25/03/24 - 26/03/24)

Frise chronologique Communication

Décembre

Janvier

Février

Mars

Avril

Communication (19/12/23 - 08/04/24)

Mise en place de la communication entre les capteurs et Chirpstack (19/12/23 - 22/01/24)

Réception des données de Chirpstack sur la VM (22/01/24 - 25/01/24)

Test de la communication (29/01/24 - 01/02/24)

Intégration des données de l'OSUR (01/04/24 - 08/04/24)

Frise chronologique Lien avec l'utilisateur

Février

Lien avec l'utilisateur (26/02/24 - 01/04/24)

Mars

Écriture du guide détaillé (26/02/24 - 05/03/24)

Avril

Interface utilisateur de monitoring (18/03/24 - 01/04/24)

Affichage de statistiques via Jupyter et Grafana (19/03/24 - 28/03/24)

Matériel

5 x



Cluster de
Raspberry Pi



FoxyFind

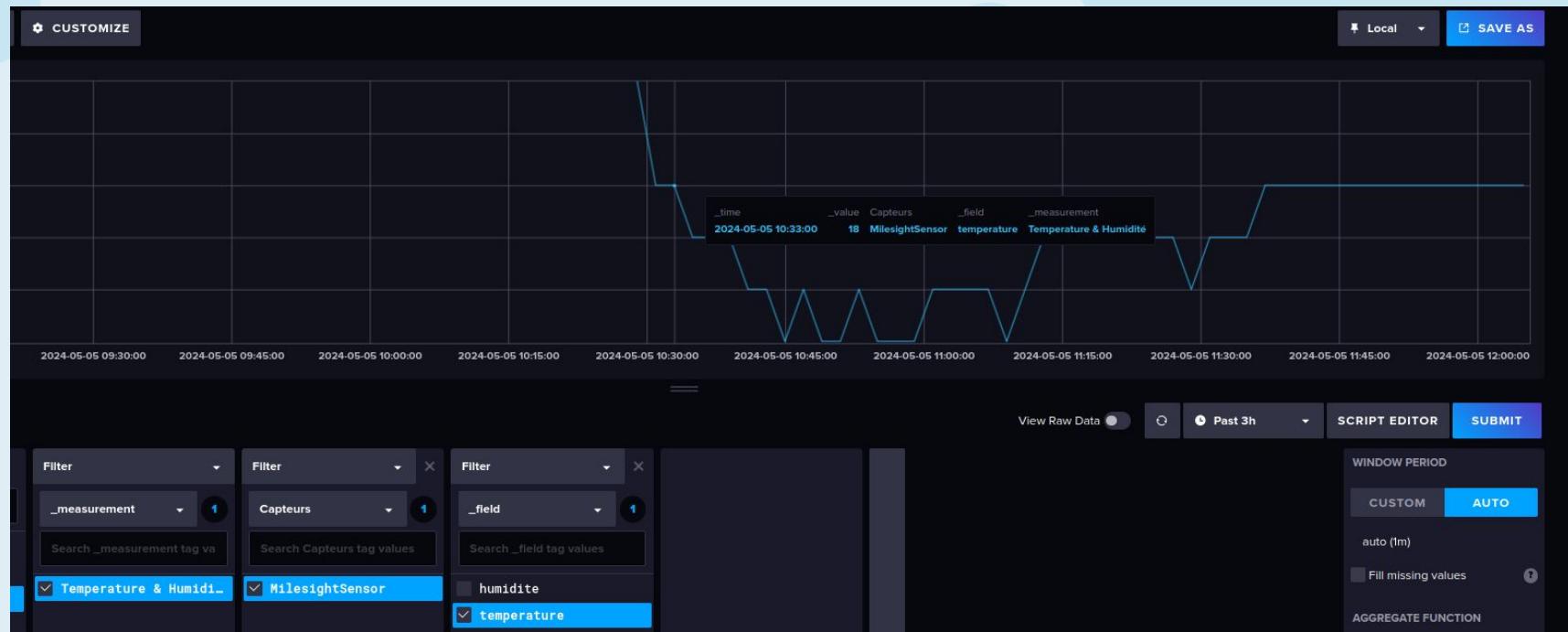


Gateway

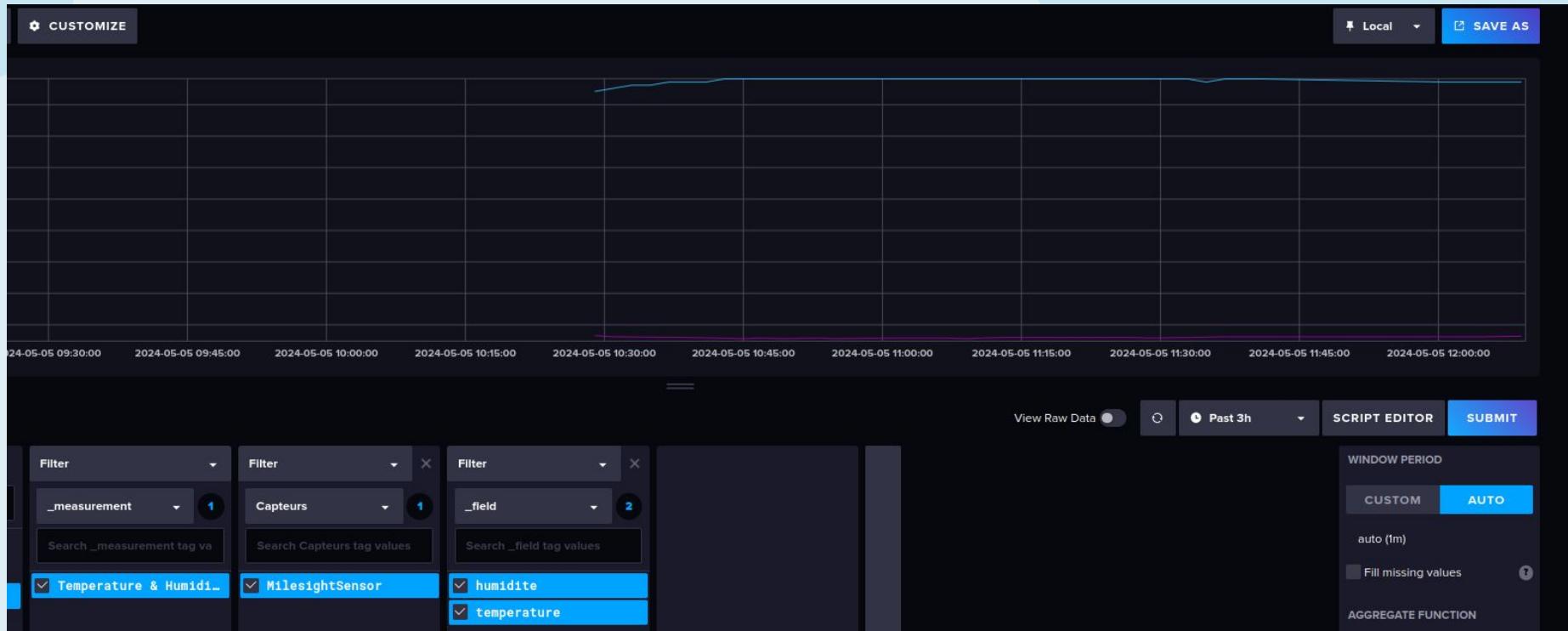


Noeud Solo

InfluxDB



InfluxDB



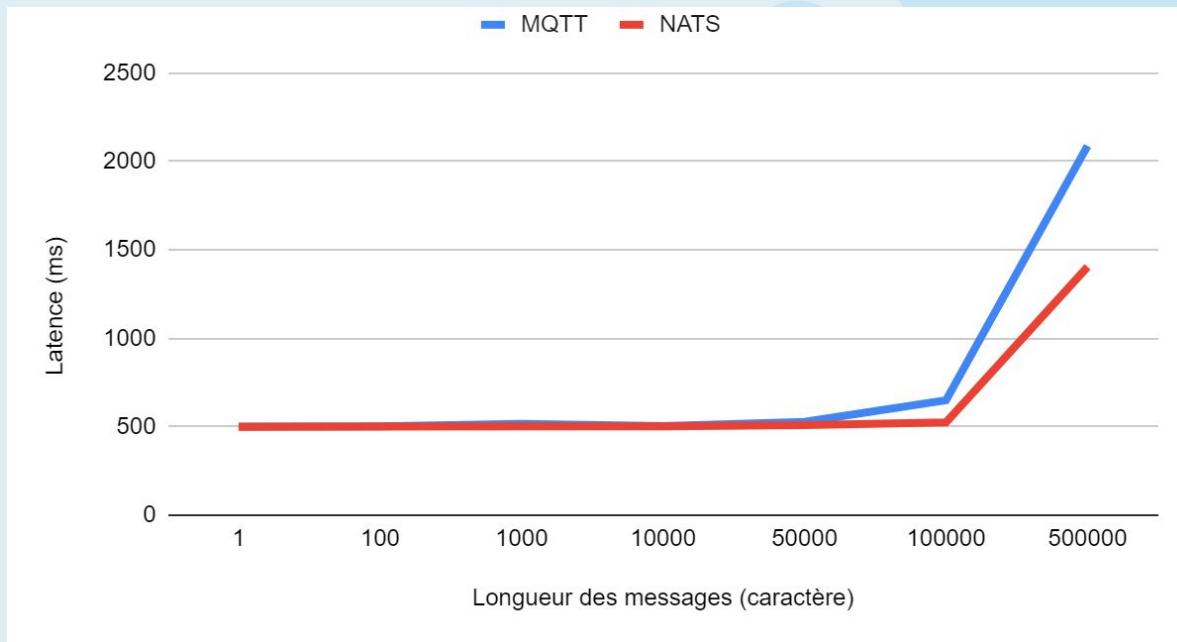
Grafana



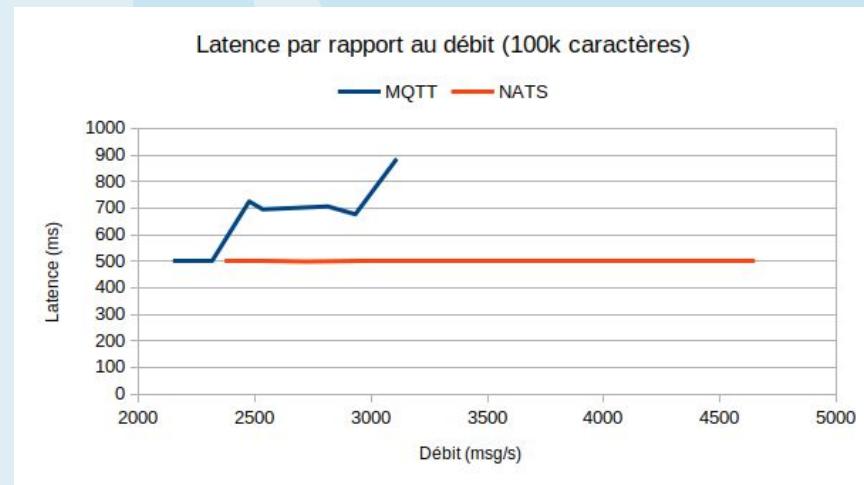
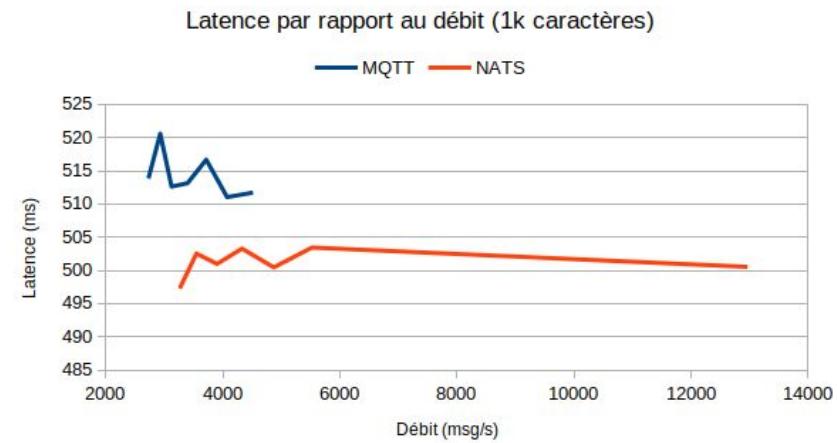
Grafana

```
<?xml version='1.0' encoding='UTF-8'?>
<rss xmlns:atom="http://www.w3.org/2005/Atom" xmlns:content="http://purl.org/rss/1.0/modules/content/" version="2.0">
  <channel>
    <title>Mycelium 3.0</title>
    <link>https://projets.info.insa-rennes.fr/projets/2022/Mycélium 2.0/</link>
    <description>Flux RSS du projet Mycelium 3.0</description>
    <docs>http://www.rssboard.org/Rss-specification</docs>
    <generator>python-feedgen</generator>
    <language>fr</language>
    <lastBuildDate>Sat, 04 May 2024 21:27:28 +0000</lastBuildDate>
    <pubDate>Sat, 04 May 2024 21:27:28 +0000</pubDate>
    <item>
      <title>Cohérence données EM300-TH</title>
      <description>Valeurs non cohérente ! Vérifiez capteur !</description>
      <pubDate>Sun, 05 May 2024 08:44:29 +0000</pubDate>
    </item>
    <item>
      <title>humidex</title>
      <description>Temperature : 18°C ,Humidité : 59% ,Humidex : 19 , Sensation : Sensation de confort</description>
      <pubDate>Sun, 05 May 2024 08:44:25 +0000</pubDate>
    </item>
    <item>
      <title>Données EM500-CO2 OSUR</title>
      <description>Temperature: 14.9°C, Humidity: 80.5%, Pressure : 1007.3hPa, CO2 : 496.0ppm</description>
      <pubDate>Sun, 05 May 2024 08:43:06 +0000</pubDate>
    </item>
    <item>
      <title>Cohérence données EM300-TH</title>
      <description>Valeurs non cohérente ! Vérifiez capteur !</description>
      <pubDate>Sun, 05 May 2024 08:42:29 +0000</pubDate>
    </item>
    <item>
      <title>humidex</title>
      <description>Temperature : 18°C ,Humidité : 59% ,Humidex : 19 , Sensation : Sensation de confort</description>
      <pubDate>Sun, 05 May 2024 08:42:25 +0000</pubDate>
    </item>
    <item>
      <title>Cohérence données EM300-TH</title>
      <description>Valeurs non cohérente ! Vérifiez capteur !</description>
      <pubDate>Sun, 05 May 2024 08:40:29 +0000</pubDate>
    </item>
    <item>
      <title>humidex</title>
      <description>Temperature : 18°C ,Humidité : 58% ,Humidex : 19 , Sensation : Sensation de confort</description>
      <pubDate>Sun, 05 May 2024 08:40:25 +0000</pubDate>
    </item>
```

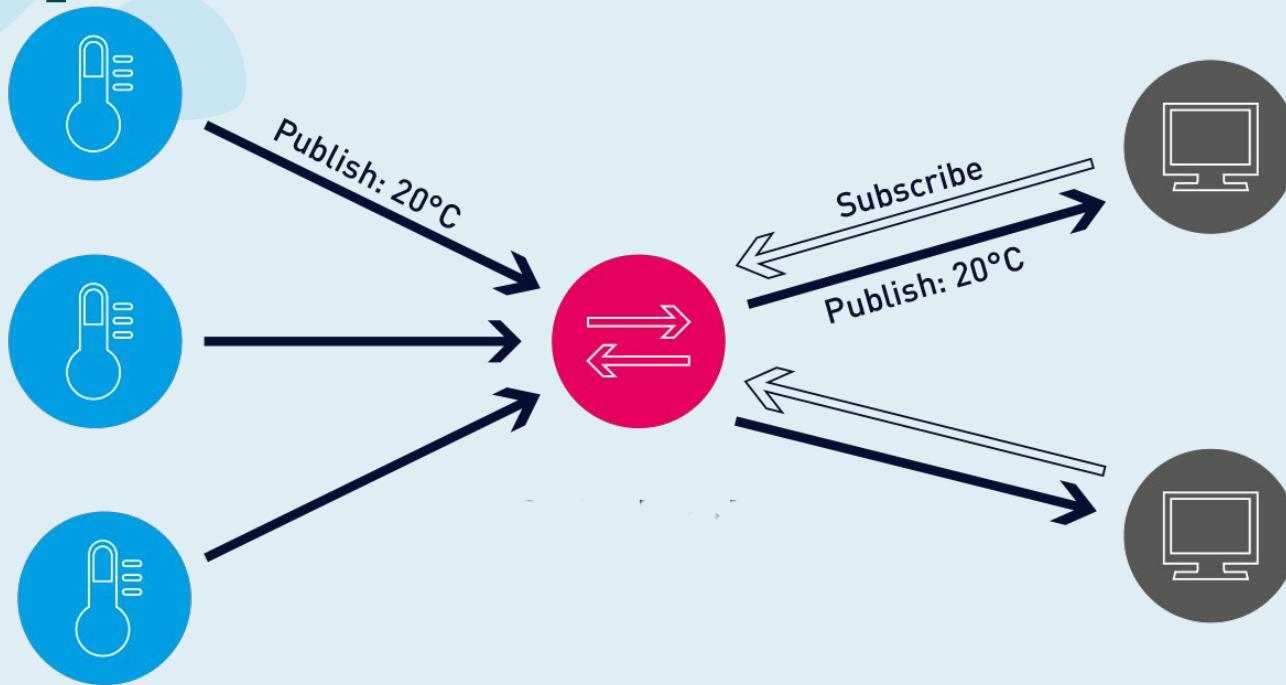
Evaluation performance MQTT-NATS



Evaluation performance MQTT-NATS



Principe MQTT



Principe MQTT

```
import grpc
import math
import os
from chirpstack_api import api # type: ignore
from dotenv import dotenv_values

# Configuration.
config = dotenv_values(".env")
config = {
    **dotenv_values(".env"), # load shared development variables
    **os.environ, # override loaded values with environment variables
}

server = config["SERVER"]
api_token = config["API_TOKEN"]
dev_eui = config["DEV_EUI"]
period_sec = config["PERIOD_SEC"]

hex_period = format(int(period_sec), 'x')
len_payload=math.ceil(len(hex_period) / 2)

# Byte conversion and byte order inversion
hex_period = int(period_sec).to_bytes(len_payload, byteorder='big')
hex_period = hex_period.hex()[2:] + hex_period.hex()[:2]
payload = "ff03" + hex_period
payload=bytes.fromhex(payload)
```

```
if __name__ == "__main__":
    # Connect without using TLS.
    channel = grpc.insecure_channel(server)

    # Device-queue API client.
    client = api.DeviceServiceStub(channel)

    # Define the API key meta-data.
    auth_token = [("authorization", "Bearer %s" % api_token)]

    # Construct request.
    req = api.EnqueueDeviceQueueItemRequest()
    req.queue_item.confirmed = False
    req.queue_item.data = payload
    req.queue_item.dev_eui = dev_eui
    req.queue_item.f_port = 85

    resp = client.Enqueue(req, metadata=auth_token)

    # Print the downlink id
    print(resp.id)
```