

UV Infrastructures d'intégration

TP 2020

Golang / MQTT / HTTP / NoSQL

Objectif du TP

Développer un système de collecte et de restitution de données météo des aéroports (température, vitesse du vent, pression atmosphérique)

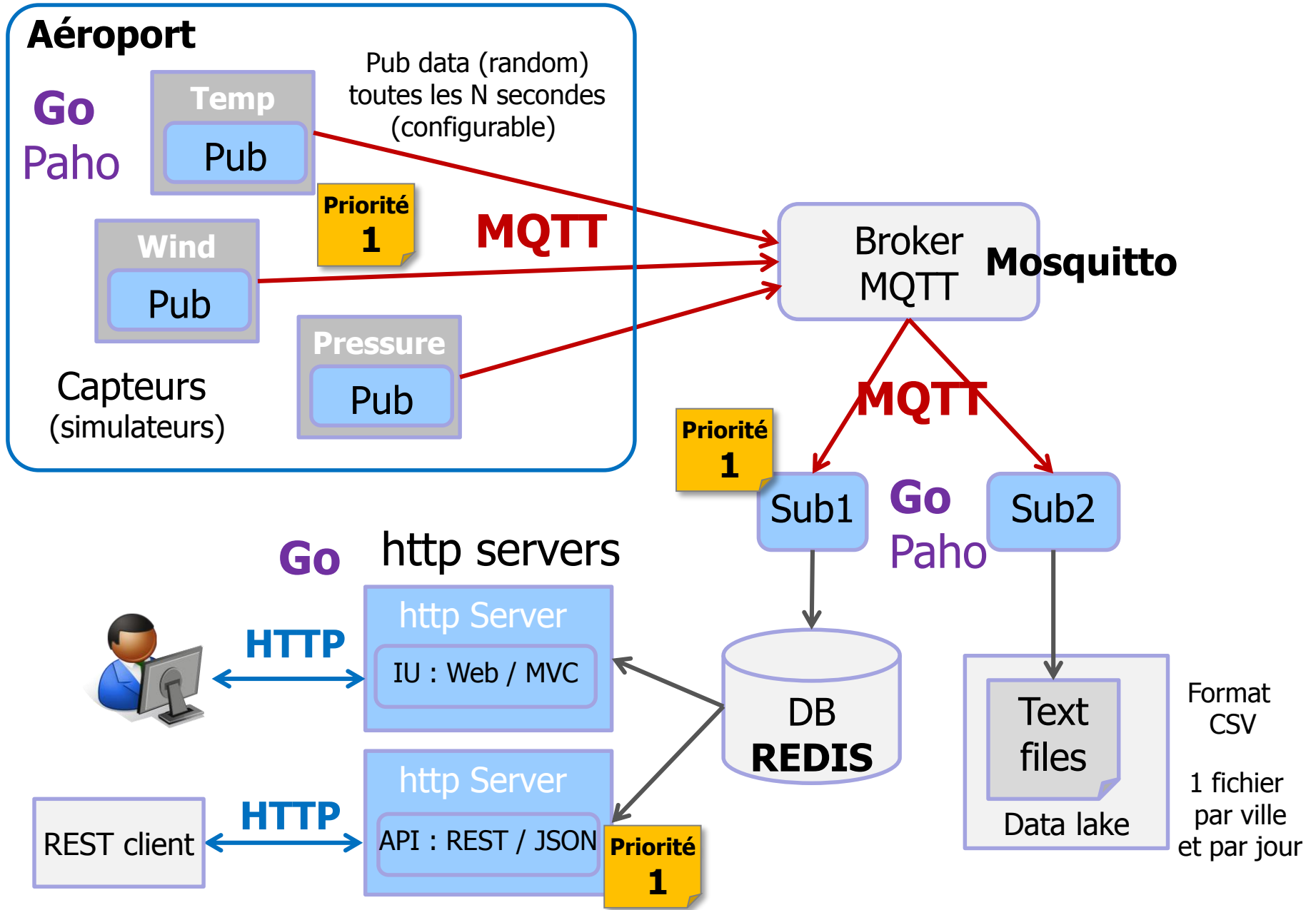
Tous les composants logiciels sont développés en **Go**.

Protocoles utilisés : **MQTT** & **HTTP**

Database NoSQL Key-Value : **REDIS**

Broker MQTT : **Moquitto**

Vue d'ensemble



Capteurs

Données envoyées par les capteurs :

- Id du capteur (entier)
- Id de l'aéroport (code « IATA » sur 3 caractères)
- Nature de la mesure (Temperature, Atmospheric pressure, Wind speed)
- Valeur de la mesure (numérique)
- Date et heure de la mesure (timestamp : YYYY-MM-DD-hh-mm-ss)

Fréquence des émissions de données :

Toutes les 10 secondes

Configuration d'un capteur (fichier de config) :

- Adresse du broker MQTT
- Port du broker MQTT
- Niveau de QoS
- ID client

Codes IATA : https://fr.wikipedia.org/wiki/Liste_des_codes_AITA_des_a%C3%A9roports/A

Base de données et fichiers

Base « NoSQL » : REDIS (ou autre base NoSQL)

Doit conserver toutes les données émises par les capteurs

Fichiers plats (.csv)

1 fichier par aéroport, par jour et par type de mesure
(ex : « NTE-2019-10-25-Temp.csv »)

API REST

Fournir une description « **Open API** » / « Swagger » de l'API

API de consultation (« GET ») :

- Liste des mesures d'un certain type entre 2 bornes de temps (date + heure)
- Moyenne des mesures des 3 types pour une journée

Format des données renvoyées : **JSON**