9/5/2022

Définition

**Time Series Data**. Il s’agit d’une séquence de points de données collectés à intervalles réguliers sur une période de temps.(donnée horodatée)

(source https://www.lebigdata.fr/time-series-database-definition)

**Installation/Découverte de influxDB et de l’outil grafana**

Architecture

* + Si la mesure n’existe pas lors de l’insertion, la mesure est créée.
  + Des clés de balises (tag keys) peuvent être définies dans une ligne.
  + Une ou plusieurs clés de champs (field keys) sont définies dans une ligne.
  + Série = combinaison mesure/tag key.

measurement, tag key1=value1, tag key2=value2 [,...]

* + Protocole ligne InfluxDB : format texte pour écrire des points dans InfluxDB.

<measurement>[,<tag\_key>=<tag\_value>[,<tag\_key>=<tag\_value>]] <field\_key>=<field\_value>[,<field\_key>=<field\_value>] [<timestamp>]

* + InfluxDB stocke les données dans des groupes de fragments (shard groups). Les groupes de fragments sont gouvernés par la politique de rétention (infinie ou non) et stockent les données par intervalles de temps appelés durée de fragment (shard duration) (défini par l’utilsiateur).

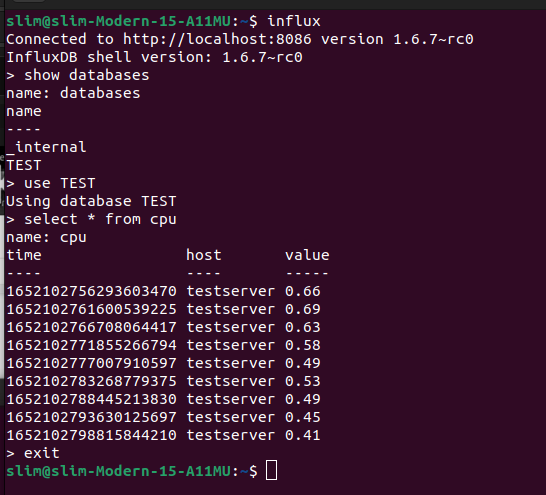
moteur "In Memory Index"

* + Chaque base de données a ses propres fichiers WAL (Write Ahead Log) et TSM.
  + Les segments WAL stockent les blocs compressés des écritures et suppressions.
  + Les fichiers TSM stockent les données compressées des séries en colonnes (columnar format).
  + Le cache est une représentation en mémoire (in-memory) des données stockées dans le WAL. Il est interrogé à l’exécution et fusionné avec les données stockées dans les fichiers TSM.
  + L’index In-Memory est un index partagé par les fragments qui fournit un accès rapide aux mesures, balises et séries.

(source https://www.sqlpac.com/fr/documents/influxdb-v1.7-architecture-installation-configuration-utilisation.html)

### Installation de influxdb : faite (+ test effectué : stocker le nombre moyen de processus prets à exécuter dans une table d’une database influx)

Voici les données de la base de données TEST (de la table cpu)



Les données sont obtenues et enregistrées via ce script: (permettant de voir à chaque 5 secondes le nombre moyen des processus en attente à étre exécutés)

while true; do curl -i -XPOST 'http://localhost:8086/write?db=TEST' --data-binary "cpu,host=testserver value=`cat /proc/loadavg | cut -f1 -d' '`"; sleep 5; done

Conversion de la consommation

Les propositions à retenir:

1 unité (consommation pour une ampoule LED) = 9 W

1 unité (de consommation annuelle d’un smartphone) = 3,65 kWh

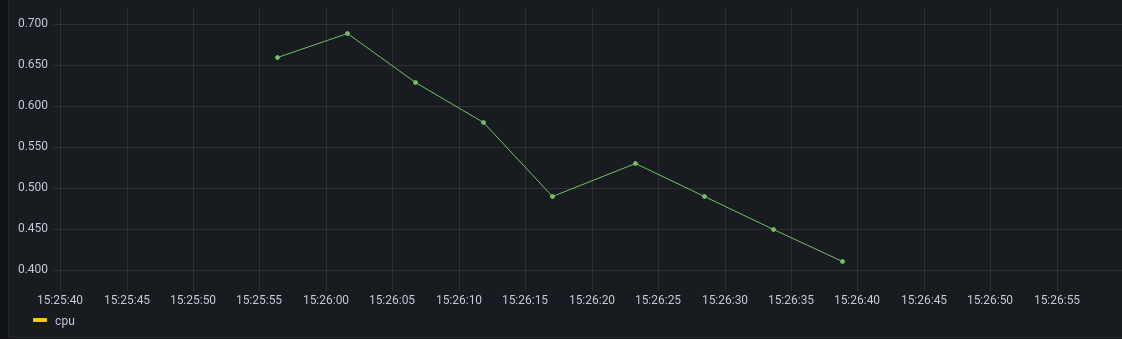
1 unité (de consommation annuelle pour un Réfrigérateur classique) = 100 et 200 kWh

1 unité (consommation électrique moyenne d’un foyer français) = 390 kWh/mois

(Le choix dépendra des valeurs des mesures (via scaphandre/powerJoular/powerapi))

Installation de grafana: faite (+ test sur la database précédente)

Voici le graphique du test:



Demain : voir le fonctionnement/tester scaphandre/powerJoular/powerapi