

保存したデータを取り出す



土屋 健

前章まででデータの保存ができるようになったので、保存したデータの参照機能を作ります。

データ取り出しの仕様

● 取り出したデータにフィルタをかける

前章までで整備した時系列データベース InfluxDB からのデータ取り出しは、Organization と Bucket を指定して行います。第3章で述べたように Organization は1つしか作成しないので、いずれかの Bucket からデータを取り出すのが基本処理となります。Bucket は、データ・サーバではチャンネル ID で指定します。

Bucket からデータを全て取り出すと、大量に出力されるので、出力データを絞るためにフィルタ機能を用意します。以下のフィルタ条件を指定して出力データを絞るようにします。

▶ 1, measurement

通常の使い方として、同種のデータは同じ measurement に集めるので、大抵は measurement で出力を絞ると思います。

リレーショナル・データベース管理システム (RDBMS) でもテーブル単位でデータを扱うのが一般的ですし、Bucket 全体ではなく measurement を範囲とした方がデータを扱いやすいと思います。

▶ 2, source

source は Tag として登録しているので、インデックス利用で高速にフィルタできます。機能としては用意していますが、今回の実験では使っていません。

▶ 3, 期間指定

期間指定は start と stop の2つで行います。日時は YYYYMMDDhhmmss (例 20220901123040) 形式の文字列とします。start, stop の組み合わせで表1のような動作となります。

● 出力データ形式

API から返却するデータは、リスト1のJSON形式とします。この形であれば、検索条件と、どこから

表1 期間指定は start と stop の2つで行う

start	stop	出力範囲
なし	なし	全期間
指定あり	なし	start 指定日時以降の最新までのデータ
なし	指定あり	最過去から stop 指定日時までのデータ
指定あり	指定あり	start 指定日時から stop 指定日時までのデータ

(bucket+measurement) のデータかの判断ができ、全てのデータを取り出すことが可能となります。なお、本章のリストは右上のQRコードから閲覧できます。

InfluxDB からデータを取得するプログラム

リスト1の仕様を満たすデータ取得処理を実装します。まずは、InfluxDB アクセス・ライブラリ (LIB/influxdb.py, リスト2) のデータ取得に必要な以下の処理について説明します。

● get_data() メソッド (105 ~ 170 行目)

InfluxDB に保存されているデータを、条件に従って取得するメソッドです。以下の引数を指定して呼び出します。なお、start, stop, source は省略可能です。

```

bucket_name : 対象となる Bucket
measurement : 対象となる measurement
start       : 開始日時の指定
stop        : 終了日時の指定
source      : 対象となる 'source' Tag の値を指定

```

処理としては、検索条件の文字列を作成して、クエリ API を実行し、結果を整型する流れになります。

112 ~ 115 行目で出力期間の開始日時設定を行います。start の指定に応じて range フィルタの設定文字列の start 指定部分を組み立てます。

117 ~ 120 行目で出力期間の終了日時設定を行います。stop の指定に応じて range フィルタの設定文字列の stop 指定部分を組み立てます。