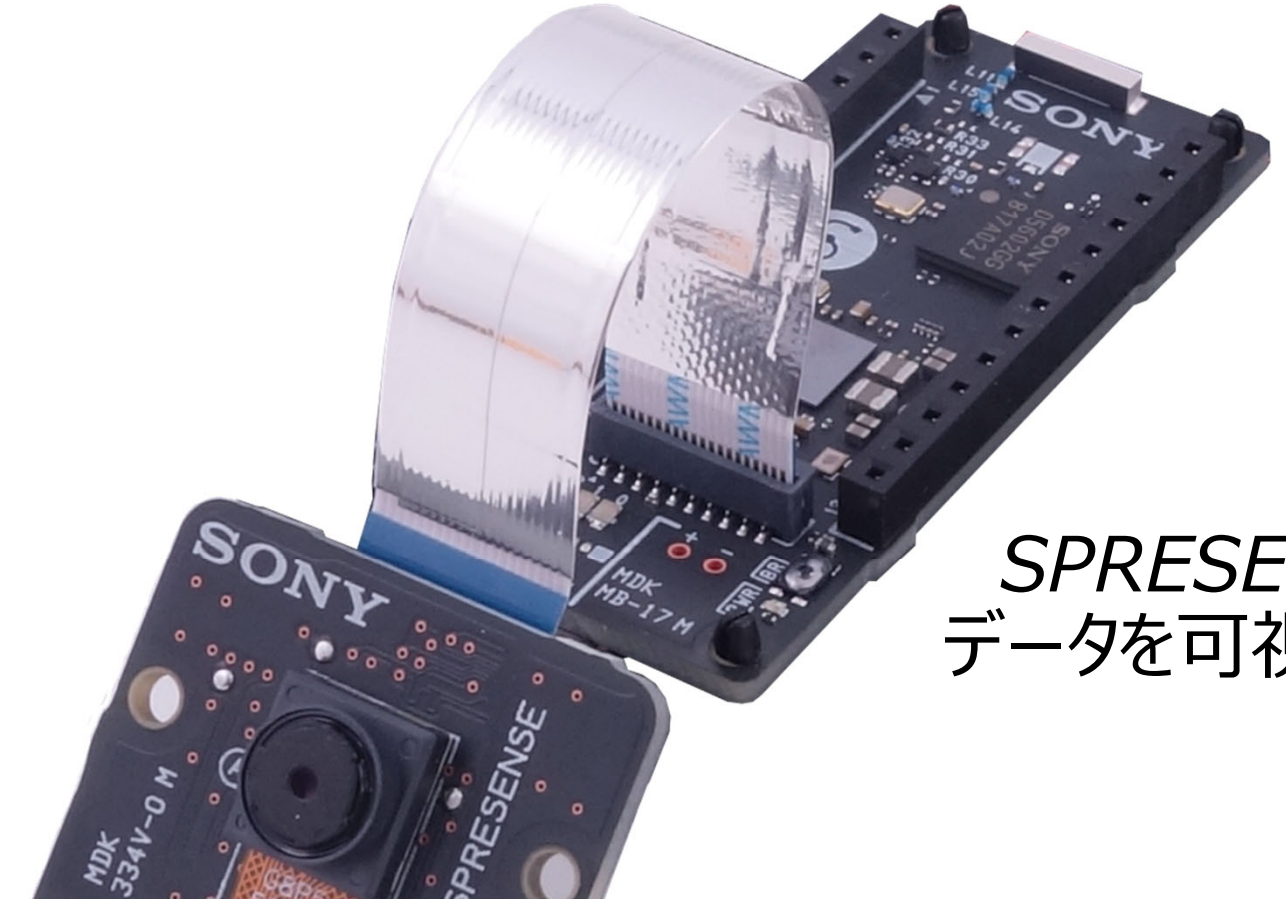


SPRESENSE™ からの
データを可視化してみよう！

Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
  - Ambientを使ってみよう！
  - kintoneを使った可視化をしてみよう！
  - フリーブローカを使ってローカルPCで可視化しよう！
- SpresenseでいろんなLPWAを使ってみよう！

データの可視化の必要性

改めて言うまでもないですが、データは**人が理解する**ことで初めて**価値を持つ**ものであり、目に見えて分かる形にされていないデータは価値がありません。（のちに利用するとしてもその時点で可視化されます。）
データの可視化とは、単なる数値だけでは理解しにくい現象や関係性を**ひと目で分かる形**（グラフ・チャート・表）に表すことを指します。

Spresenseで使える無料での可視化ツール

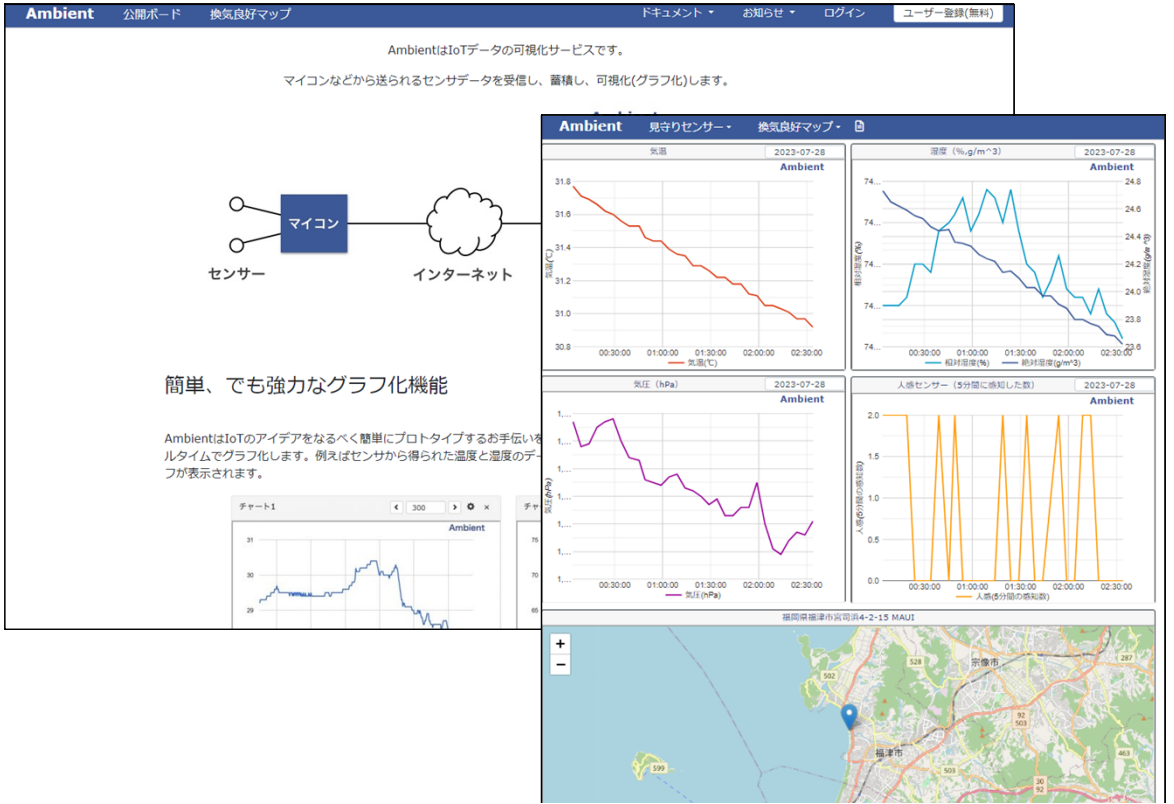
- **Ambient**
- AWS
- Azure
- **kintone**
- ~~Machinist @IIJ~~
- ~~**モノプラットフォーム @SAKURA Internet**~~
- Harvest @SORACOM
- **フリーのMQTTブローカーを使った可視化**

etc...

Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
  - Ambientを使ってみよう！
  - kintoneを使った可視化をしてみよう！
  - フリーブローカを使ってローカルPCで可視化しよう！
- SpresenseでいろんなLPWAを使ってみよう！

Ambientを使ってみよう！



https://ambidata.io/

Ambientは無料で使いやすいツールです。

ロケーション情報も含め手軽に可視化できます。

Ambient 使用の流れ

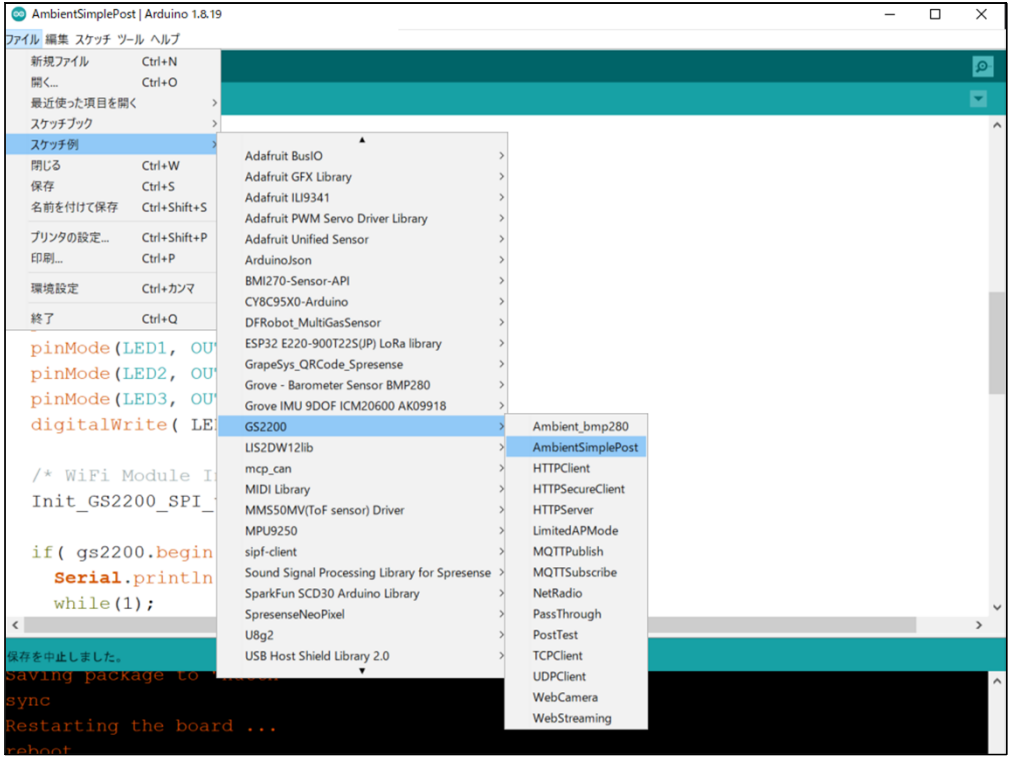
Ambient には、チュートリアルがあり、これを見ながら作業が可能です。
まずは、 **Ambientを使ってみる** を参考に進めます。(https://ambidata.io/docs/gettingstarted/)

Ambientを使う大まかな流れは次のようになります。

1. ユーザー登録(無料)
2. チャネル生成
3. マイコン側プログラミング
4. データー送信
5. 可視化(グラフ化)

Ambientではマイコンからデーターを送ってグラフ化するまでは非常に簡単です。 最低限必要なのはAmbientサイトで「チャネル」を作ることと、 マイコン側のプログラムでチャネルIDとライトキーを指定してデーターを送ることだけです。 チャネルの名前やどんなマイコンがつながっているか、 どんなセンサーのどんなデーターを送るかといった属性情報は後から記述することができますが、 複雑なことは後回しにして、とにかく最初は簡単に始められます。

SpresenseのデータをAmbientに送ってみよう！



Ambient は、Ambient が用意しているキーにJSON形式でデータを入れてHTTPでPOSTすることで可視化しています。
そのため、HTTPでPOSTできれば、どのような通信プラットフォーム（LTE、WiFi、etc...）でも構いません。

今回は、WiFiを使って試してみます。

GS2200ライブラリの

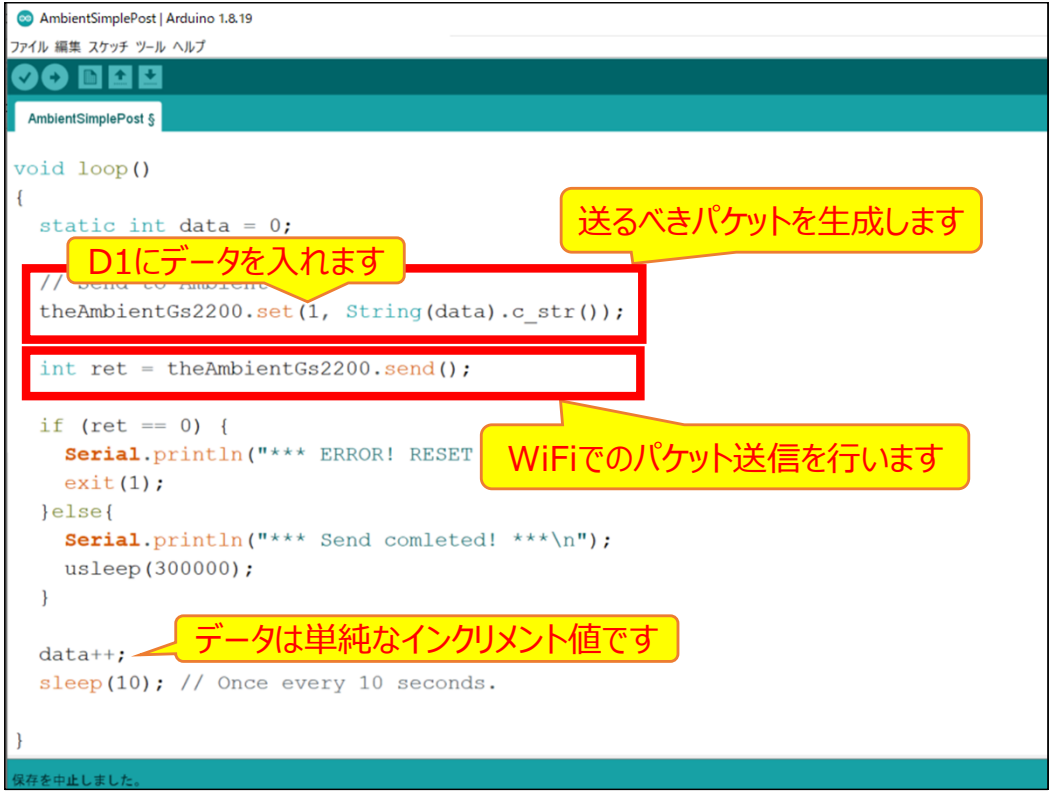
AmbientSimplePost.ino

を使います。

AmbientSimplePostのコード解説

AmbientSimplePostのコード解説

AmbientSimplePostのコード解説



データは、D1～D8まで使えます。
D9、D10は、緯度、経度になります。

入れるべきデータは、

bool set(int field, const char * data);
bool set(int field, double data);
bool set(int field, int data);

文字列、整数、少数が可能です。

うまく動かない時に！

- WiFiのタイプはあっていますか？
- WiFiのアクセスポイントはあっていますか？
- AmbientのチャンネルID、write keyはあっていますか？

LTEでも同様に可能です！

- 太田さん作成のLTE向けAmbientライブラリがあります。
https://github.com/TE-YoshinoriOota/Ambient_SpresenseLTEM

- LTEを使ったCO2監視サンプルがこちらにあります。
CO2_sensing_to_Ambient.ino

ご参考に。

Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
  - Ambientを使ってみよう！
  - kintoneを使った可視化をしてみよう！
  - フリーブローカを使ってローカルPCで可視化しよう！
- SpresenseでいろんなLPWAを使ってみよう！

kintone をつかってデータ収集と可視化をしてみよう！

ご紹介していたキャリアの提供しているIoTプラットフォームですが、最近、低調気味です。

- ~~Machinist @IIJ　：廃止~~
- ~~モノプラットフォーム @SAKURA Internet　：停止~~
- Harvest @SORACOM

その中で、今回は、一般的には業務管理ツールのイメージのある kintone をIoTに向けて使ってみる方法をご紹介します。

kintone の開発ライセンス

APIを使ったアプリの開発を目的とした場合、開発ライセンスが1年間無料で使えます。
（制限はあります。）

お試しであれば、こちらが使えます。

https://cybozu.dev/ja/kintone/developer-license/

kintone でのデータ収集（アプリ開発）

基本的には、kintoneを使ったアプリ開発は、こちらを参照に実施します。

https://kintone.cybozu.co.jp/seminar/ondemand.html

kintone でのアプリ開発

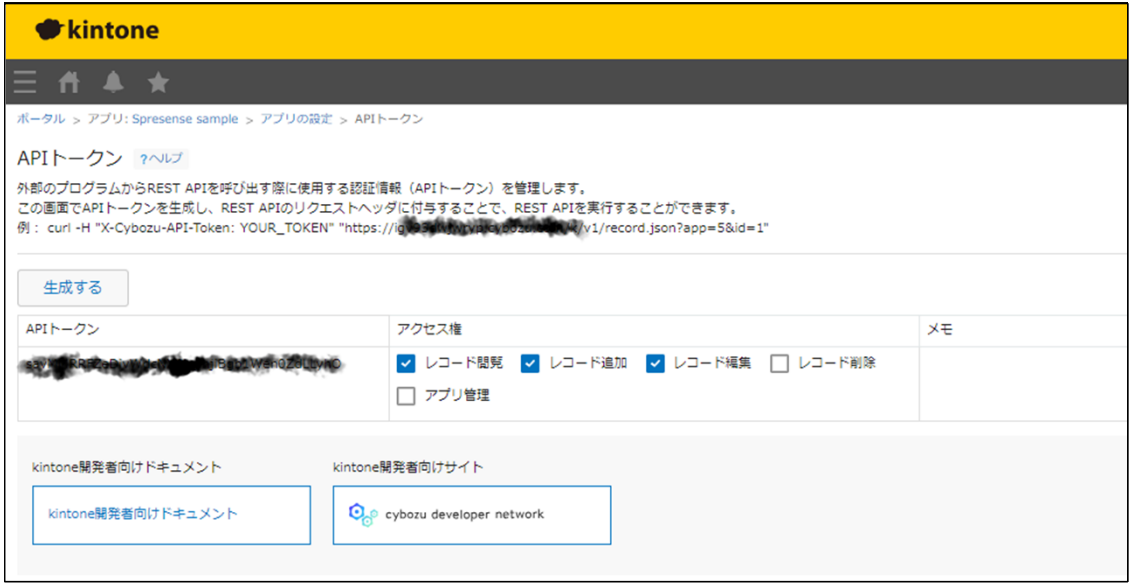
今回のサンプルとしては、単純な数値データフィールドを一つだけ作成します。

Spresenseを使ったkintoneアプリのブログはこちらにあります。

https://kintone-geeks.hatenablog.com/entry/SonySpresenseLTE_to_kintone

kintone のAPIトークンの準備

アプリを作成したら、APIトークンキーを生成します。



このAPIトークンは、クラウドアクセスに使いますが、誰でもアクセス可能になってしまいますので、キーの扱いには気を付けてください。

問題が発生した場合は、そのキーを破棄して、新たにキーを生成することで回避可能です。

SpresenseからLTE経由でデータを送ってみよう！

Kintoneの準備ができましたら、LTE経由でSpresenseからhttpsでデータを送るサンプルを動かしてみましょう。
このサンプルは、こちらにあります。

https://github.com/TomonobuHayakawa/Spresense-Playground/blob/master/ForHandsOn/DataVisualization/forKintone/viaLTE/sample_for_visualization/sample_for_visualization.ino

こちらも単純なインクリメント値を送信するサンプルになります。
1分ごとにデータを送ります。（kintone上での時刻がかぶらないようにしています。）

基本は、LTEの LteHttpSecureClient.ino になります。こちらの使い方に関しては、LTEのハンズオンを参照してください。

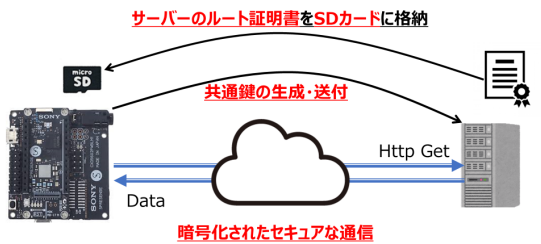
SIMは、MEEQのSIMを使用しています。お使いのSIMに合わせて修正してください。

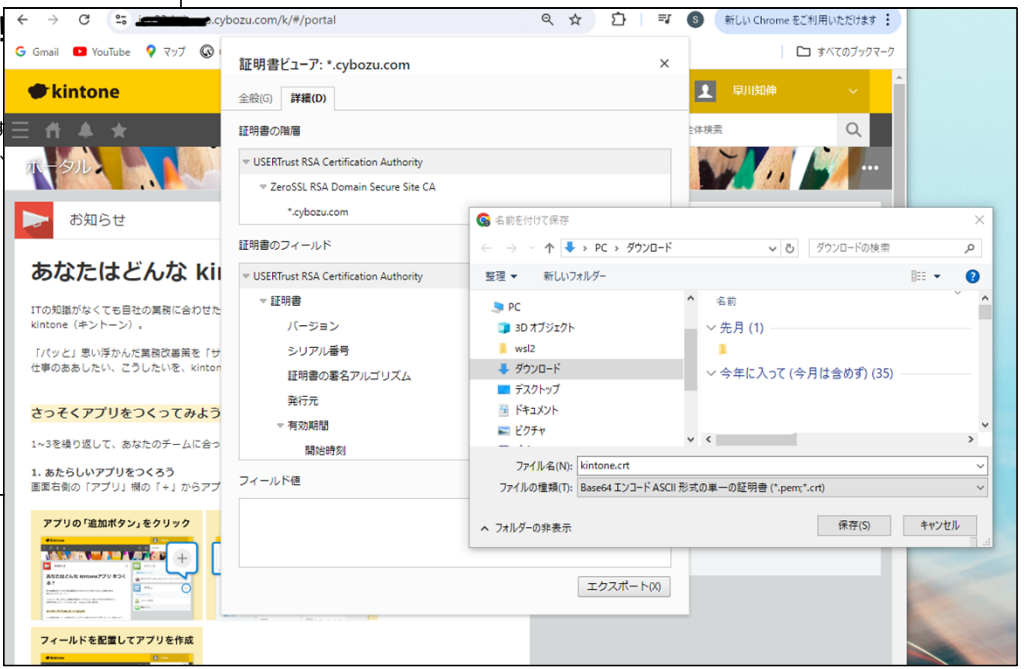
Cybozuサーバの証明書の取得

Kintoneのサーバには、セキュアなアクセスしかできませんので、証明書の取得が必要です。
こちらは、 LteHttpSecureClient.ino の時と同じです。

セキュアなWebサーバからデータを取得してみよう！
（TLSクライアントでの接続）

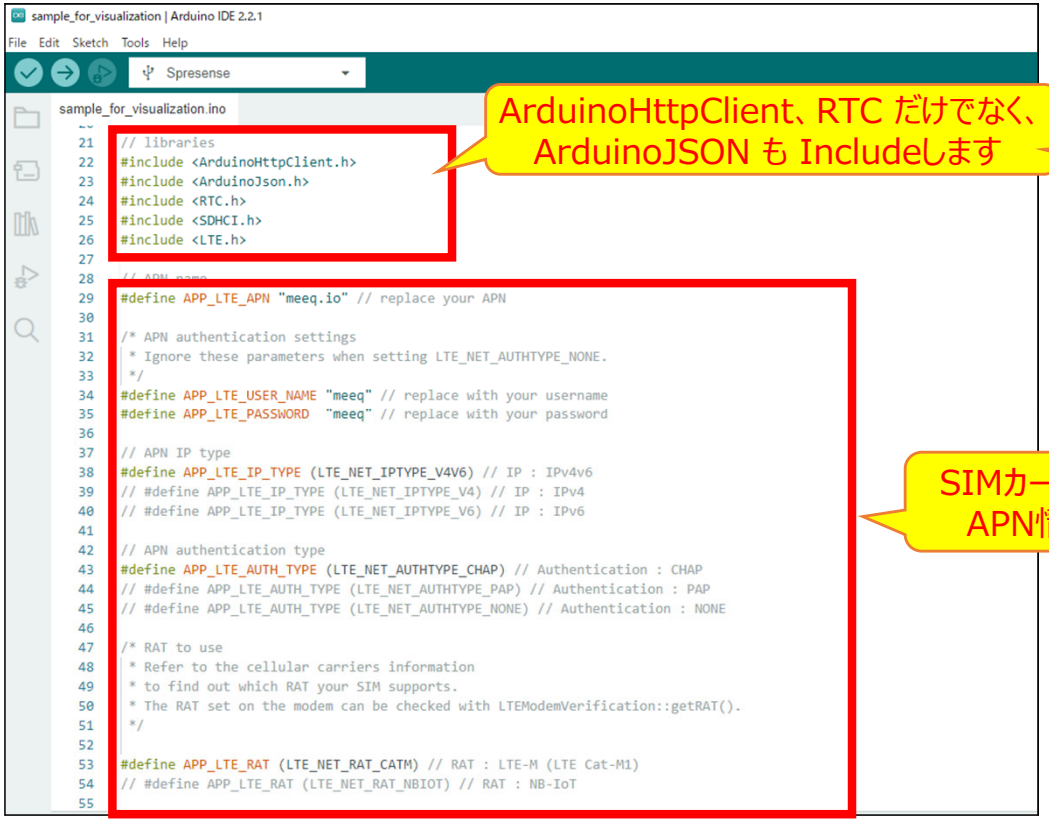
実際のインターネットサービスは、ほとんどの通信が暗号化によりセキュリティーが施されています。
ほとんどのサイトが **https://** で始まっており、**SSL/TLS** を利用した通信データの暗号化、
止、なりすまし防止を行っています。



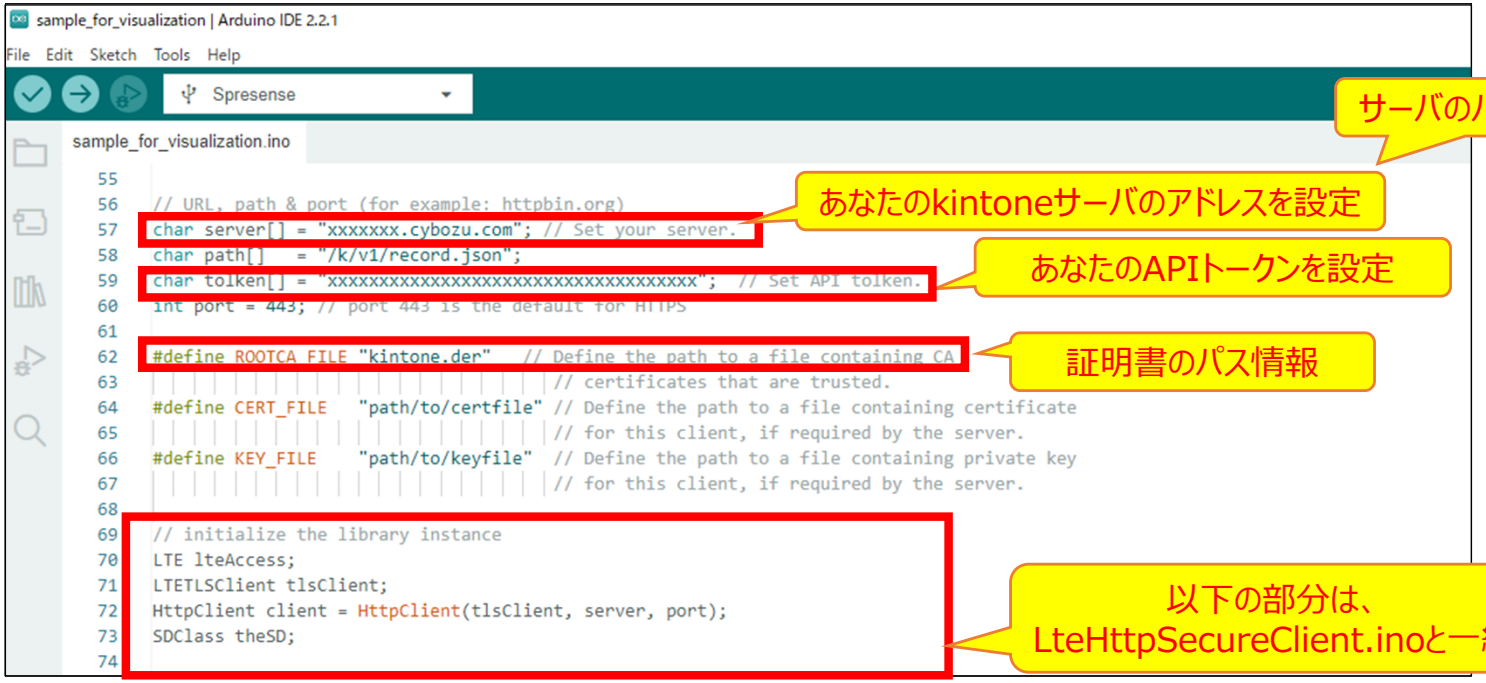


kintone.crt という名前でSDカードの
ルートに保存してください。

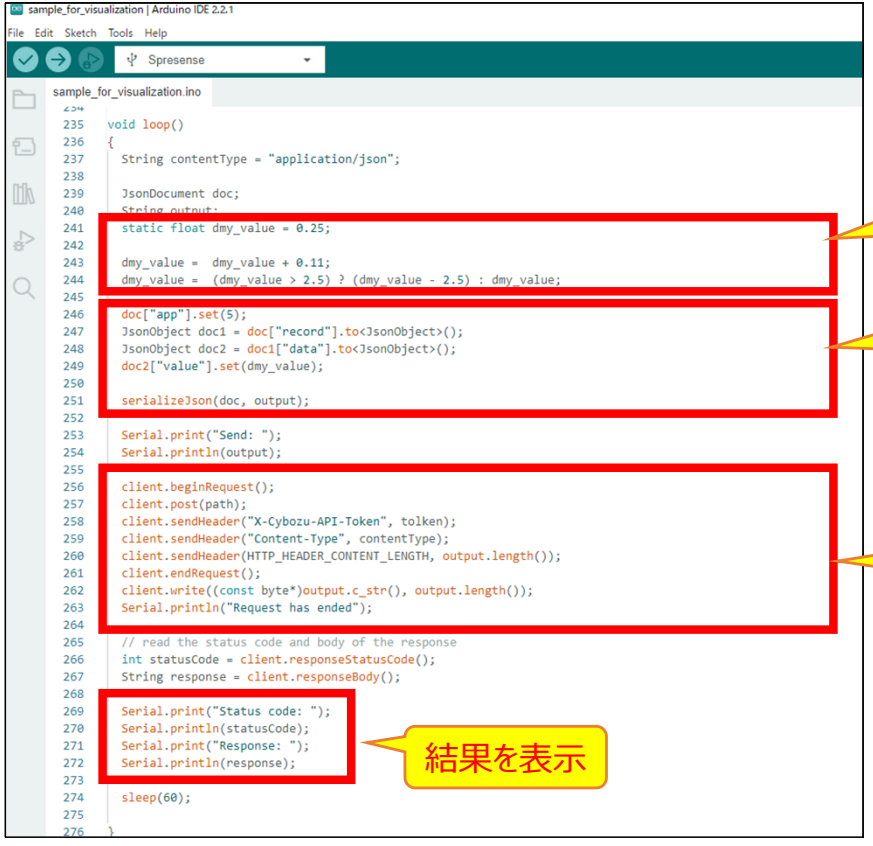
sample_for_visualizationのコード解説



sample_for_visualizationのコード解説

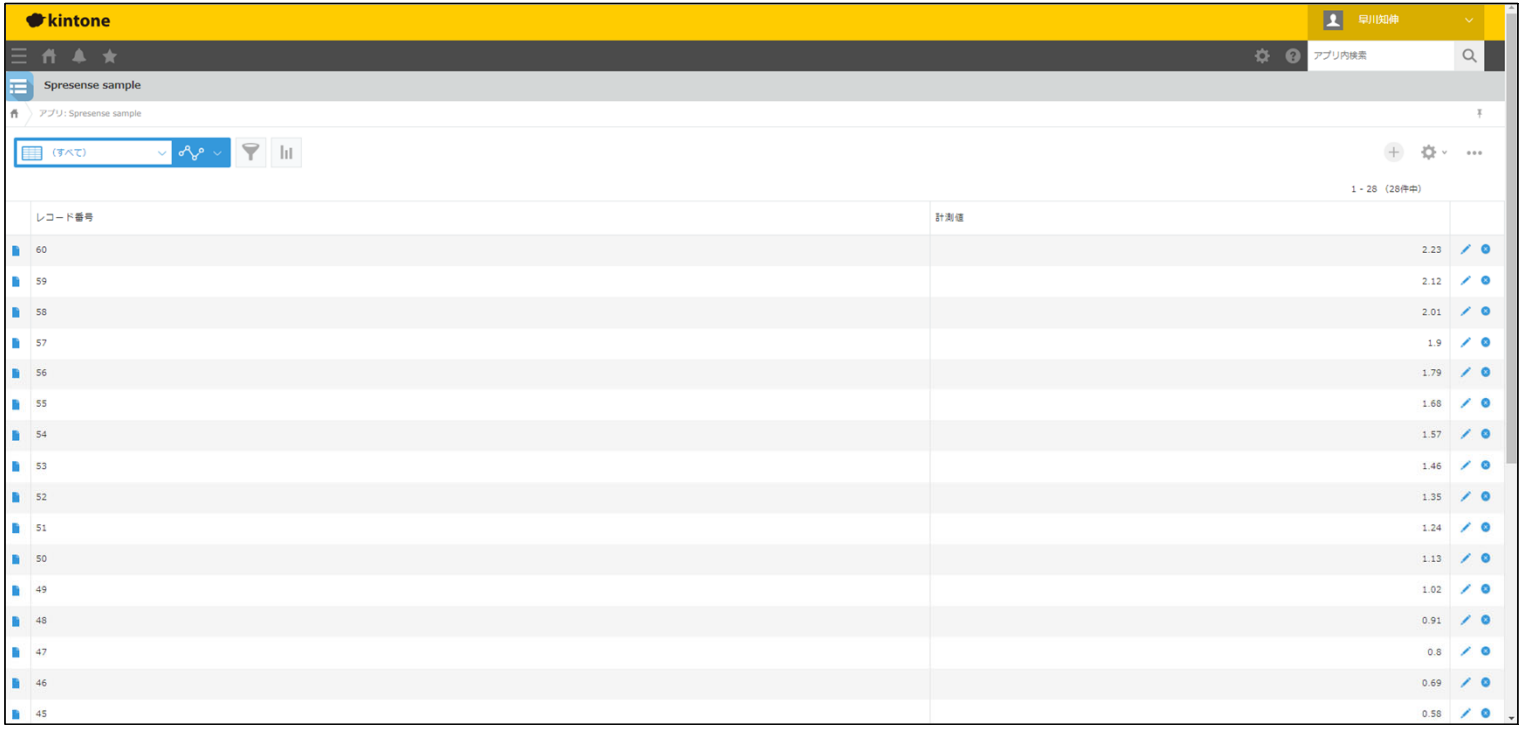


sample_for_visualizationのコード解説



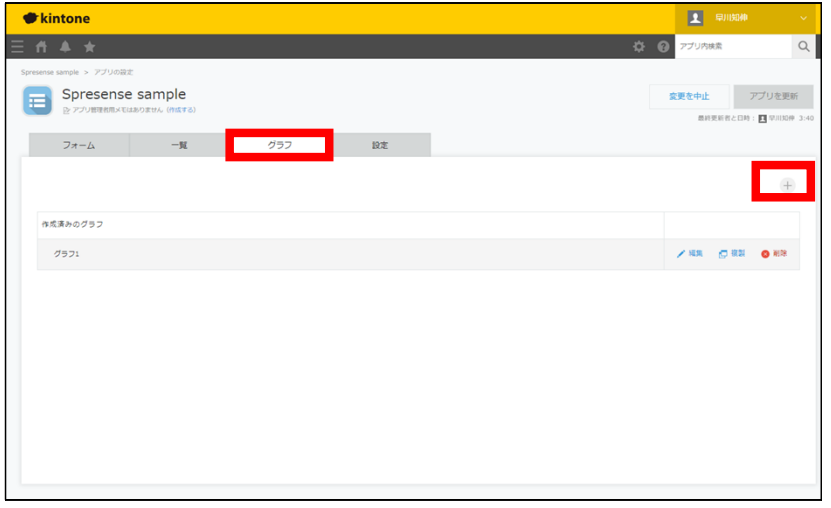
データの収集結果

上手くいくとこのようにレコード情報が追記されていきます。



データのグラフ化

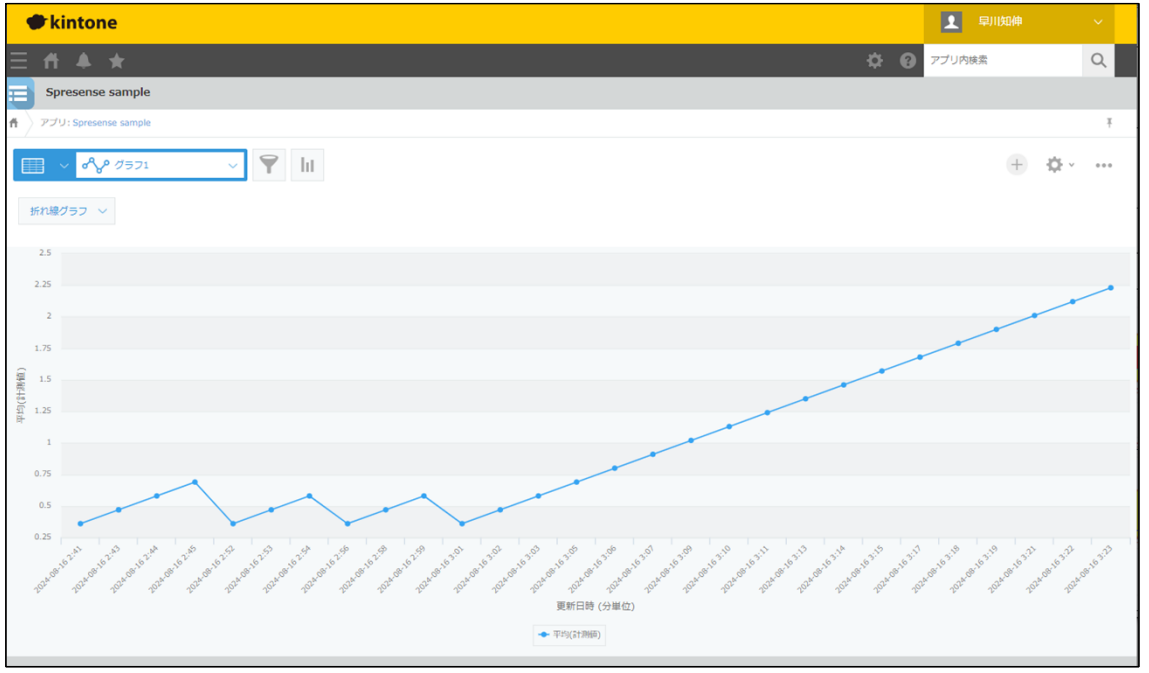
データはグラフなどを利用して可視化し、解析することで意味が出ます。
グラフ化する方法は以下になります。



設定タブの中のグラフタブを指定します。
そこで、グラフを追加します。

データのグラフ化

このような形でグラフ化できました。



kintoneを使うことで、各種の**業務アプリ**との連携も可能になります。

業務のIoT/DX化に大きく貢献できることになります。

Agenda

- Spresense Add-on・拡張ボードご紹介
- Spresense LTEを使ってみよう！
- WiFi Add-onを使ってみよう！
- BLE Add-onを使ってみよう！
- Spresenseからのデータを可視化してみよう！
  - Ambientを使ってみよう！
  - kintoneを使った可視化をしてみよう！
  - フリーブローカを使ってローカルPCで可視化しよう！
- SpresenseでいろんなLPWAを使ってみよう！

フリーブローカを使ってもっと手軽にローカルPCで可視化

- 外部のサービスを使うと、
  - 契約が必要
  - フリー期間が限られている
  - 有償

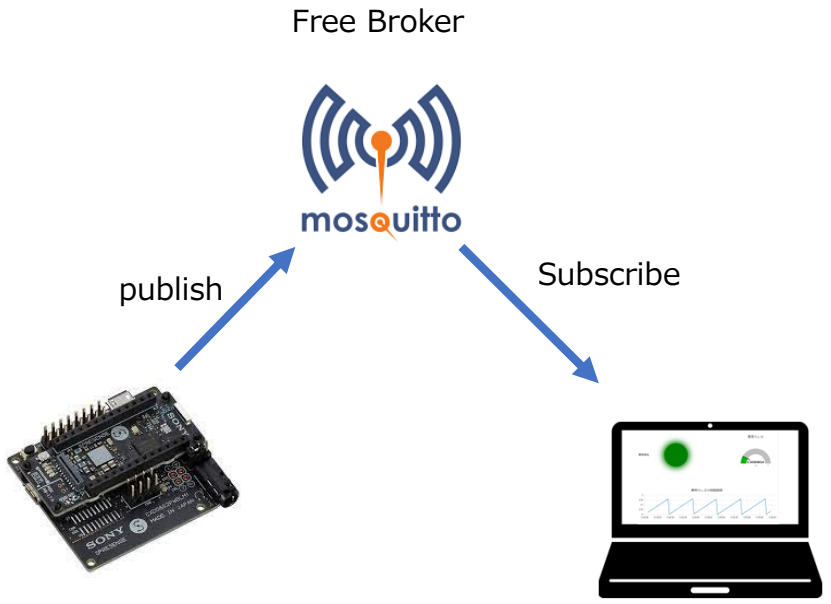
など、制約が発生する。

もっと手軽に見える可したい。

ローカルPCでの

フリーのMQTTブローカーを使った可視化

を説明します。



Node-REDをローカルPCにインストール

Node-REDの公式ページにあるインストールの手順に従って、

Node.js　（18.15.0以降）
npm　　（9.5.0以降）

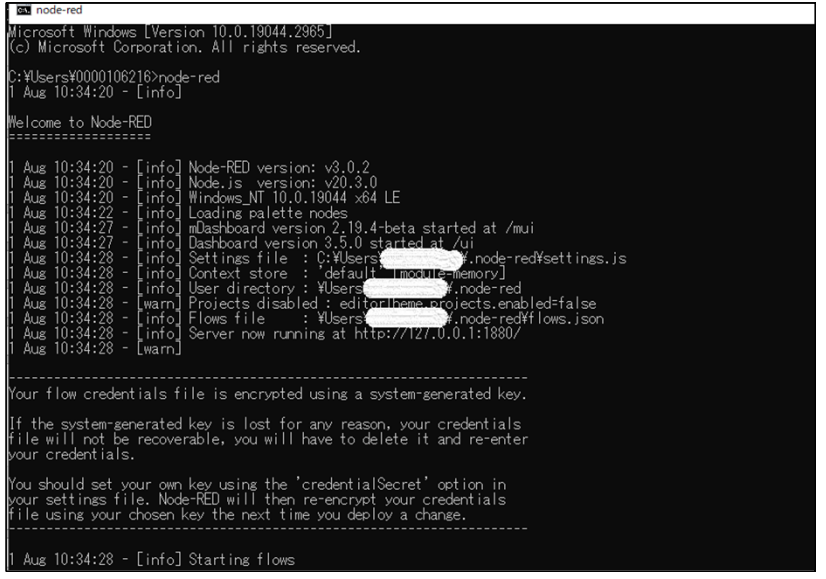
をインストール。

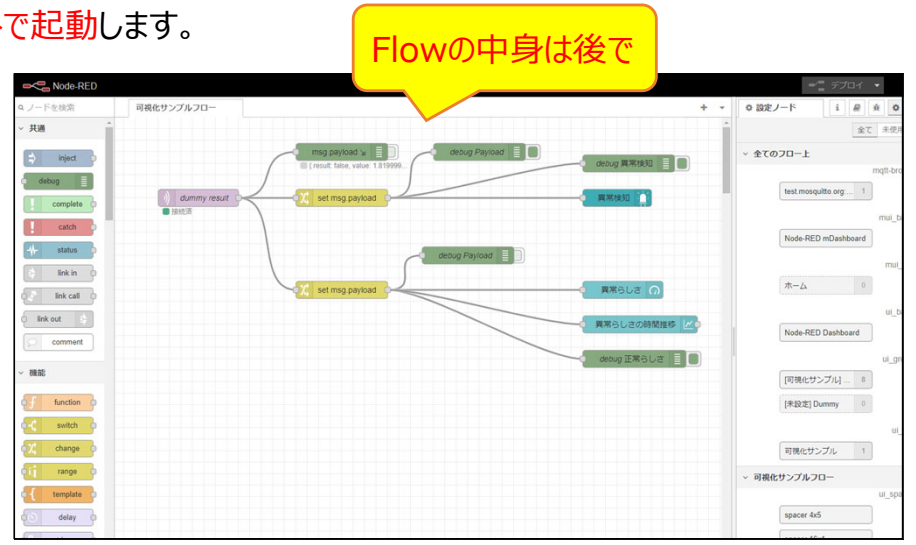
その後、Node-REDをインストールしてください。

https://nodered.jp/docs/getting-started/windows

Node-REDの起動

Node-REDを起動します。
※ただし、なぜかPowerShellでの起動には失敗します。
　コマンドプロンプトでの実行は成功するのでコマンドプロンプトで起動します。



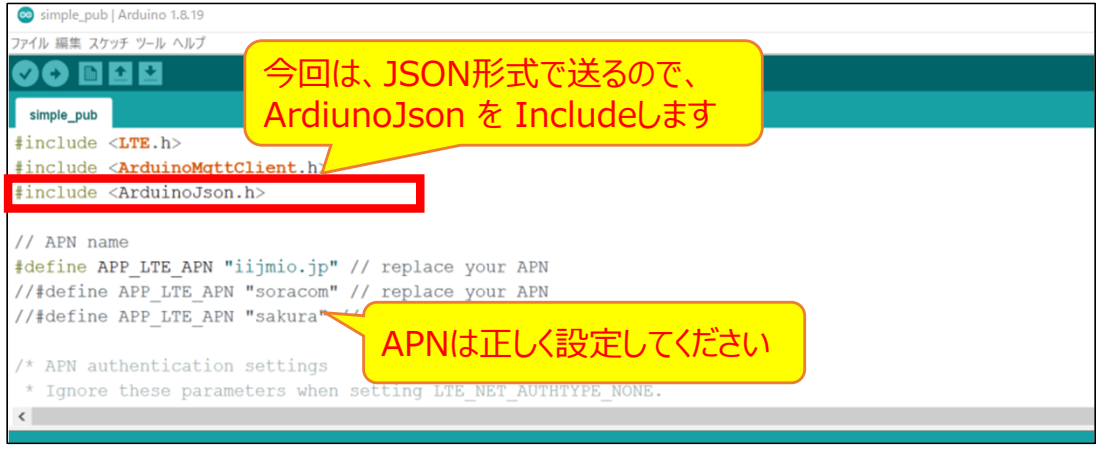


起動すると、

http://localhost:1880

でNode-REDが立ち上がります！

LTE経由でデータを送信するサンプルを作成してみよう！



ArduinoJsonは、Spresense向けのパッチが必要なので、
こちらからダウンロードしてください。

https://github.com/YoshinoTaro/ArduinoJson

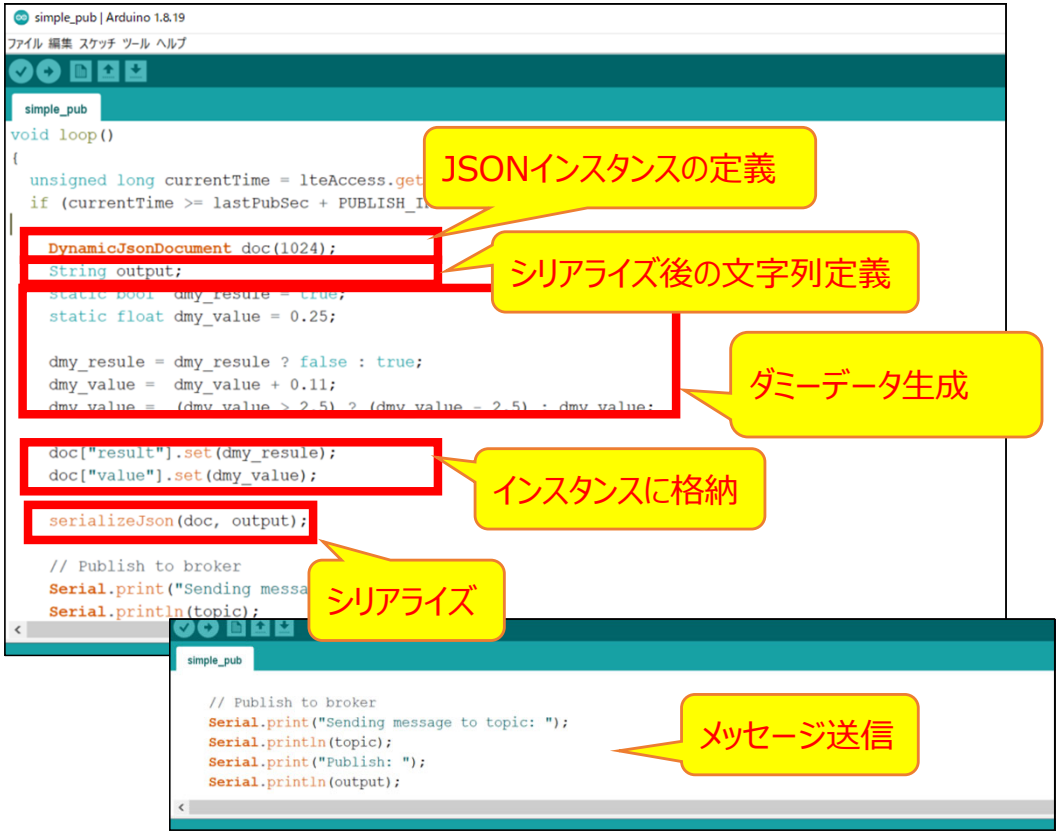
LTEのハンズオンで使った
simple_pub.ino
をベースに変更していきます。

基本的な使い方は、LTEの時と同じです。

ここに作ったものを置きました。

https://github.com/TomonobuHayakawa/Spresense-Playground/blob/master/ForHandsOn/DataVisualization/viaLTE/dummy_publish_for_local_visualization/dummy_publish_for_local_visualization.ino

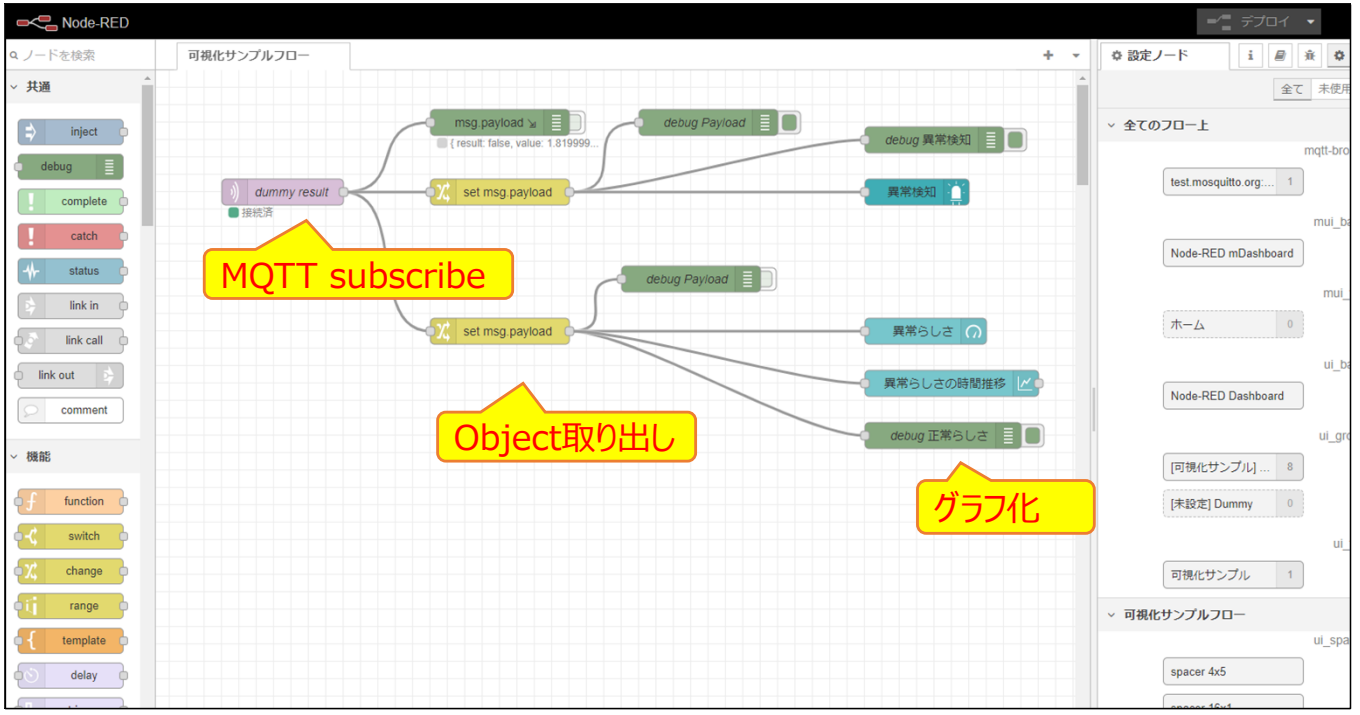
LTE経由でデータを送信するサンプルを作成してみよう！



LTEのハンズオンで使った
simple_pub.ino
をベースに変更していきます。

基本的な使い方は、LTEの時と同じです。

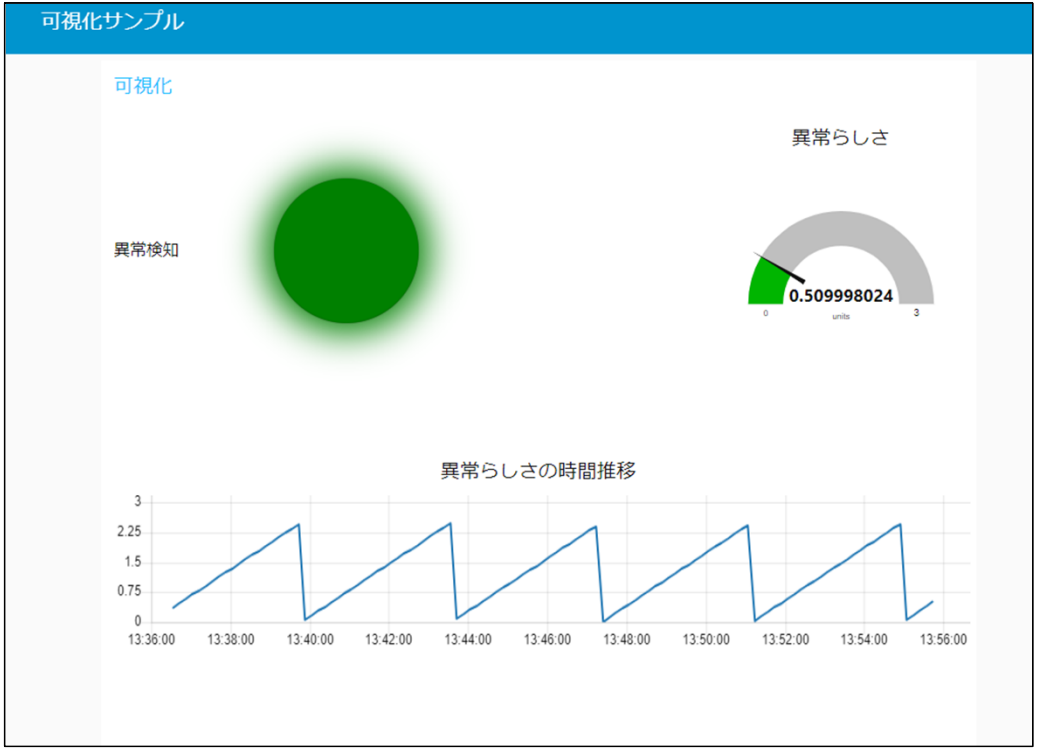
Node-RED flow



今回はJSON
そのものを送って
いるので簡単です。

ここに作ったものを置きました。
https://github.com/TomonobuHayakawa/Spresense-Playground/blob/master/ForHandsOn/DataVisualization/viaLTE/Node-RED/dummy_publish_for_local_visualization.json

可視化結果



http://localhost/ui:1880

で可視化結果が表示されます！

ここで上げたものは様々な可視化ツールの一部です。

ご自分で最適なツールを見つけて、
是非、SpresenseをIoTのPOCに活用してみてください！