

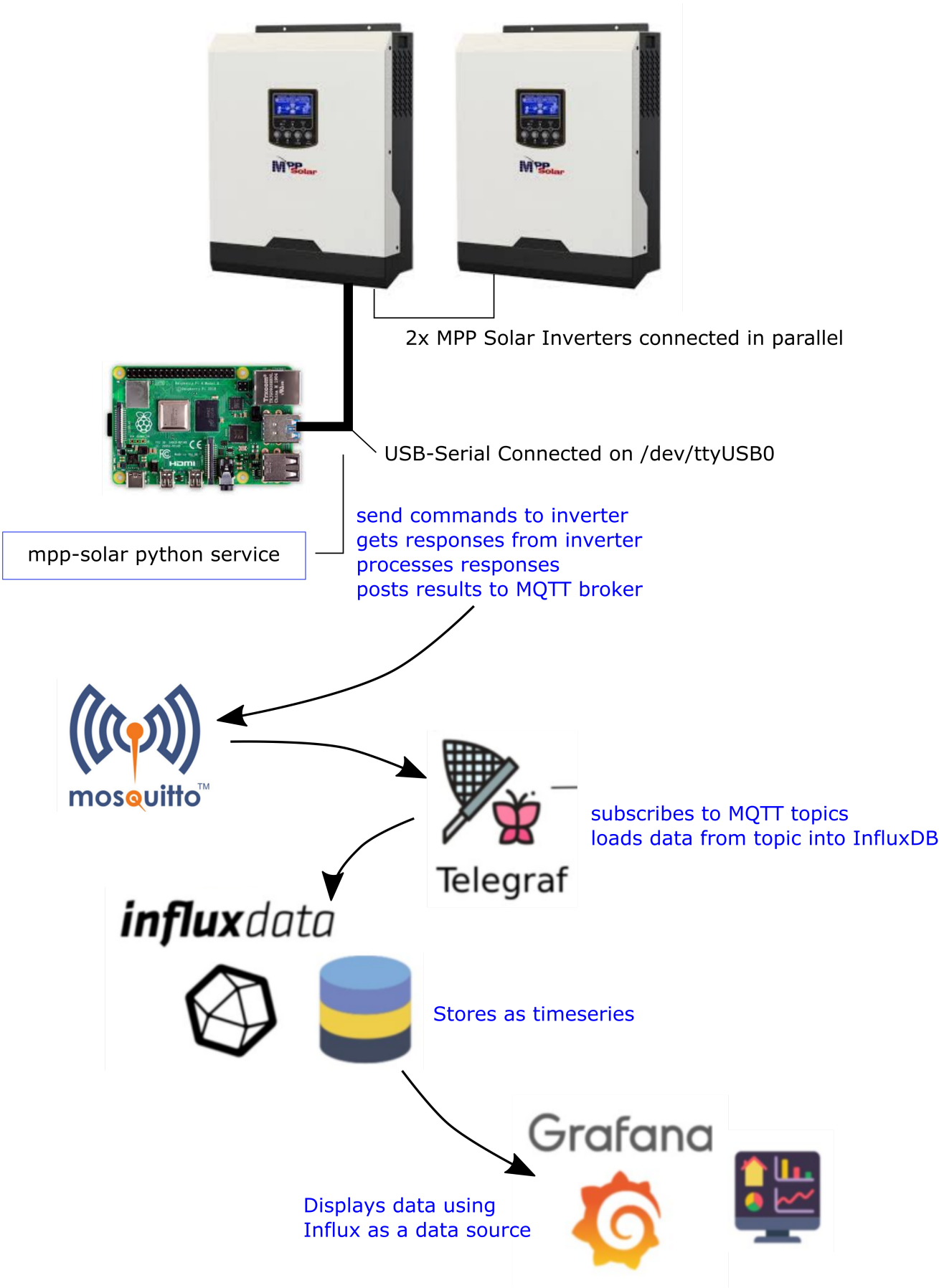
# Dev Guide (Incomplete version)

---

This document is a short guide for developers using current prediction service infrastructure. This document version is incomplete, as the prediction service is in development.

## Overview

The infrastructure for the prediction service consists of solar inverter that generates analytics related to solar power production and consumption. The inverter is a physical device connected directly to solar panels and an RPi - the data collector. The RPi collects inverter data, decodes and sends it via an MQTT connection to the AWS server. The AWS server has deployed several services on it. The first service from data flow perspective is MQTT broker. MQTT broker receives the data from the RPi and stores it in the internal broker queue. The data is read by an intermediate service - Telegraf, that sends it to the InfluxDB database. InfluxDB is a time series data storage, used frequently in IoT systems. Here the data is stored for later processing by other services, including Prediction service (currently not integrated in the infrastructure) and Analytics & Monitorings service - Grafana



Development for specific service requires some configurations, described below

## RPI and data collector service

Data collection is performed on an RPi connected serially to the inverter. The service responsible for data collection is called mpp-solar service and is a [github repository](#). The server is run as a docker container with configuration available in [project github](#).

*Note:* For RPi and mpp-solar development, currently it is needed connection to local inverter network.

To communicate with the RPi from an external device, one can use SSH connection

```
ssh microlab@192.168.1.100
```

some copying to the RPI can be done using [scp](#) tool.

*Another note:* The commands above are meant for Linux users. Windows users can use [wsl](#) or Windows-specific commands to connect via ssh

## AWS ECS Server development

For developing the AWS ECS instance, there are required credentials for connecting and working with the server. For the credentials and any other questions related to server development, please contact the Project Owner.

The first thing to do when starting the development on the server part, make sure the server is up.

```
ping 18.197.198.110
```

If the server is responding, the next thing to do is to connect

```
ssh -i ~/.ssh/YourKey.pem ec2-user@18.197.198.110
```

To send some files from external devices to the server, use [scp](#)

```
scp -i ~/.ssh/YourKey.pem "$YourFile" ec2-user@18.197.198.110:~
```

To send data from external devices to the server, make sure network security groups permit that if, if not, contact the Product Owner.

## Monitoring and Analytics

To monitor the inverter data is used [Grafana](#)