User Guide

MQTT_LiveGraph

A tool for graphical visualization of MQTT data
Developed by Jens Krogsgaard · 2025

MQTT_LiveGraph – User Guide

25-11-2025 – Jens Krogsgaard, Farum - Denmark

Table of Contents

1. Overview

MQTT_LiveGraph is a Windows application for graphical display of MQTT data in the form of line charts.
The program is ideal for monitoring, analysis, and documentation of sensor values, model railway data, transformer loads, energy consumption, and more.

Key features:
• Real-time display of MQTT topics as line graphs
• Flexible formula editor for calculating new values based on multiple MQTT topics
• Work with multiple datasets, each containing its own graphs
• Support for one or two Y-axes
• Calculation of minimum, maximum, and average values
• Language can be switched between Danish, English, and German

I developed the program in the autumn of 2025 as a hobby project together with ChatGPT. Others are welcome to use it — but naturally at their own risk.
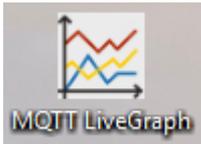Unfortunately, I can't promise lifetime support 🙄

2. Installation and configuration

2.1.   Installation

The installation is performed in these few steps:

1. Run the file **setup_MQTT_LiveGraph.exe**.

2. Follow the installation wizard.

3. After installation, a desktop shortcut is created automatically.

2.2.   Choose Language

MQTT_LiveGraph supports these three languages:
• Danish
• English
• German

The language can be changed via **Settings → Language**.
The user interface updates immediately.

2.3. MQTT - Connection

Now you need to connect to your MQTT broker — either locally or in the cloud:


- **Server:** IP address or hostname of the MQTT broker

- **Port:** Standard is 1883

- **Username / Password:** Optional – only if the connection requires it.

- **Use TLS / Certificate Verification**: TLS is usually not required for local MQTT installations. For cloud services such as HiveMQ, TLS is often mandatory.

- **Test connection**: Test if the connection is ok.

- **Sample-interval (s):** Number of seconds before new data is read from the MQTT broker.
- **UI-refresh (ms):**   How often the screen graphics are updated (in milliseconds).
- **Buffer-length (Min.):** Number of minutes of data that are stored and displayed in the chart buffer.
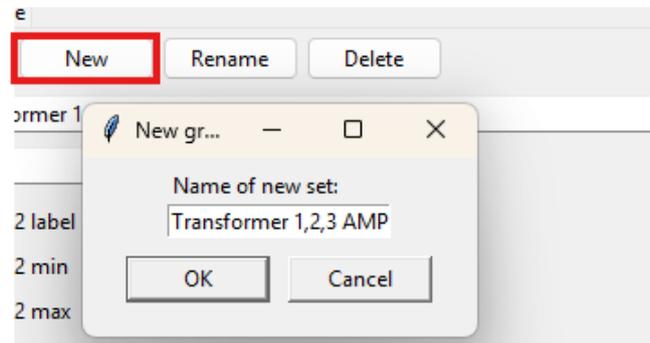
3. Settings - Charts
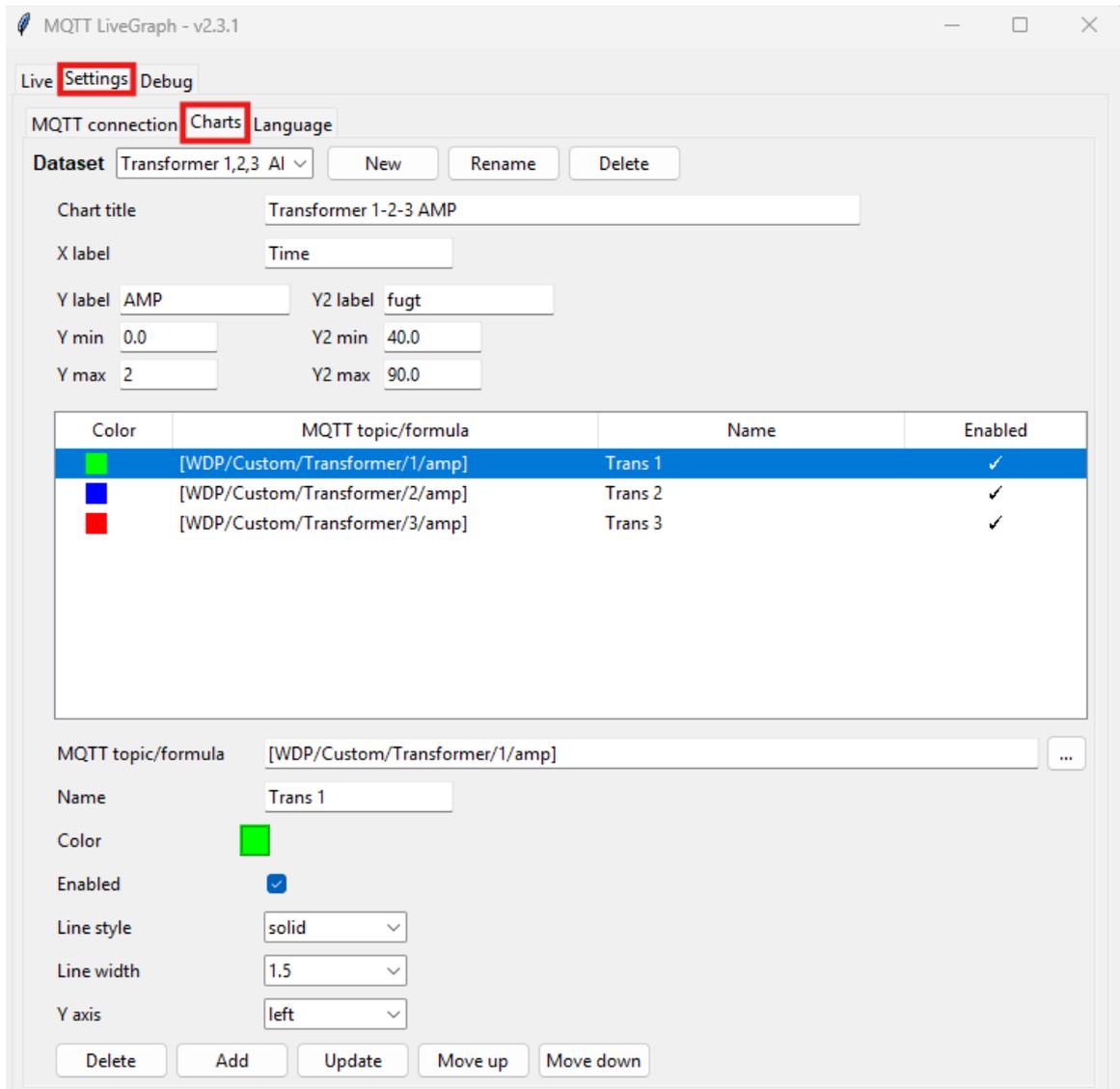
Let's say you want to create an overview of the amperage values for transformers 1, 2, and 3. You may already have these data available in your MQTT broker:
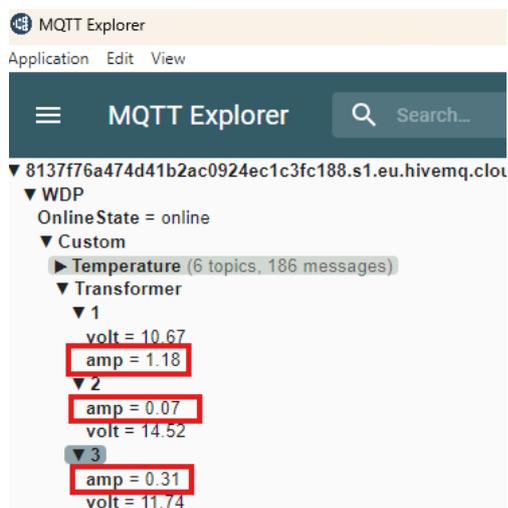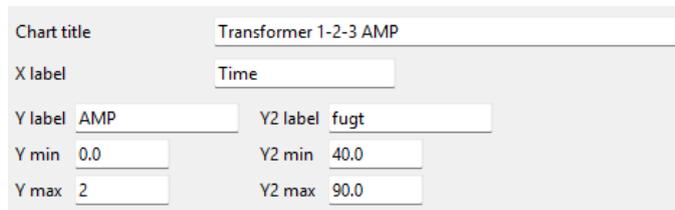
Click the **"New"** button and enter a name for your new dataset.
The dataset is created as a copy of the most recently active dataset.
You can now modify it or add new series.

Edit your settings here — you can choose to use one or two Y-axes, each with its own scale.

Now let's create the three time series. The basic principle is very simple:

You enter the series settings in the lower section and then click **"Update"**.
In the overview above, you can then see the time series you have created.

When specifying an MQTT topic, it must always be placed in square brackets — for example:
[WDP/Custom/Transformer/3/amp].

In addition, you can choose the color, line style, and name — and of course decide whether the series should be active or not.
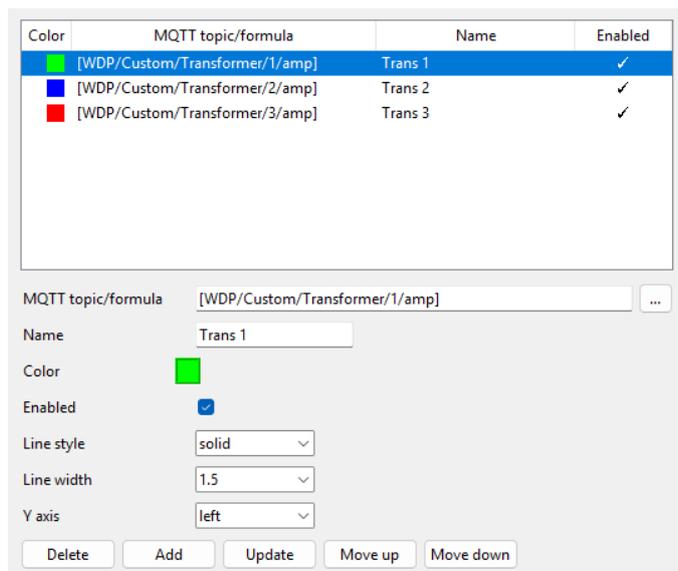
You can also enter a formula here. You can use the four basic arithmetic operations (+ − * /) as well as parentheses and constants. This is useful, for example, if you want to convert values to watts or combine the power from several transformers.

Click the button to edit the formula. [...]

Edit formula   ×

Enter formula.
Topics are written as [topic_name].
Example:
[topic_1_amp] * [topic_1_volt] +
[topic_2_amp] * [topic_2_volt] +
[topic_3_amp] * [topic_3_volt]

```
([WDP/Custom/Transformer/1/amp]  *  [WDP/Custom/Transformer/1/volt])  +
([WDP/Custom/Transformer/2/amp]  *  [WDP/Custom/Transformer/2/volt])  +
([WDP/Custom/Transformer/3/amp]  *  [WDP/Custom/Transformer/3/volt])  +
([WDP/Custom/Transformer/5/amp]  *  [WDP/Custom/Transformer/5/volt])  +
([WDP/Custom/Transformer/6/amp]  *  [WDP/Custom/Transformer/6/volt])  +
([WDP/Custom/Transformer/7/amp]  *  [WDP/Custom/Transformer/7/volt])  +
([WDP/Custom/Transformer/8/amp]  *  [WDP/Custom/Transformer/8/volt])
```

Annullér   OK

Calculation of the total power (Watts) for multiple transformers

Edit formula   ×

Enter formula.
Topics are written as [topic_name].
Example:
[topic_1_amp] * [topic_1_volt] +
[topic_2_amp] * [topic_2_volt] +
[topic_3_amp] * [topic_3_volt]

```
([WDP/Custom/Transformer/1/amp]  *  [WDP/Custom/Transformer/1/volt])  *  100 / 18
```
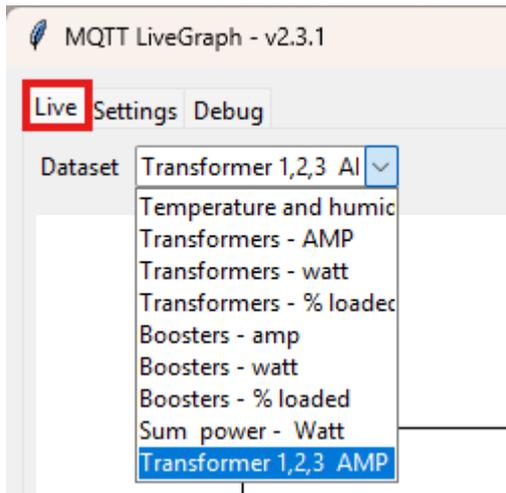
Annullér   OK

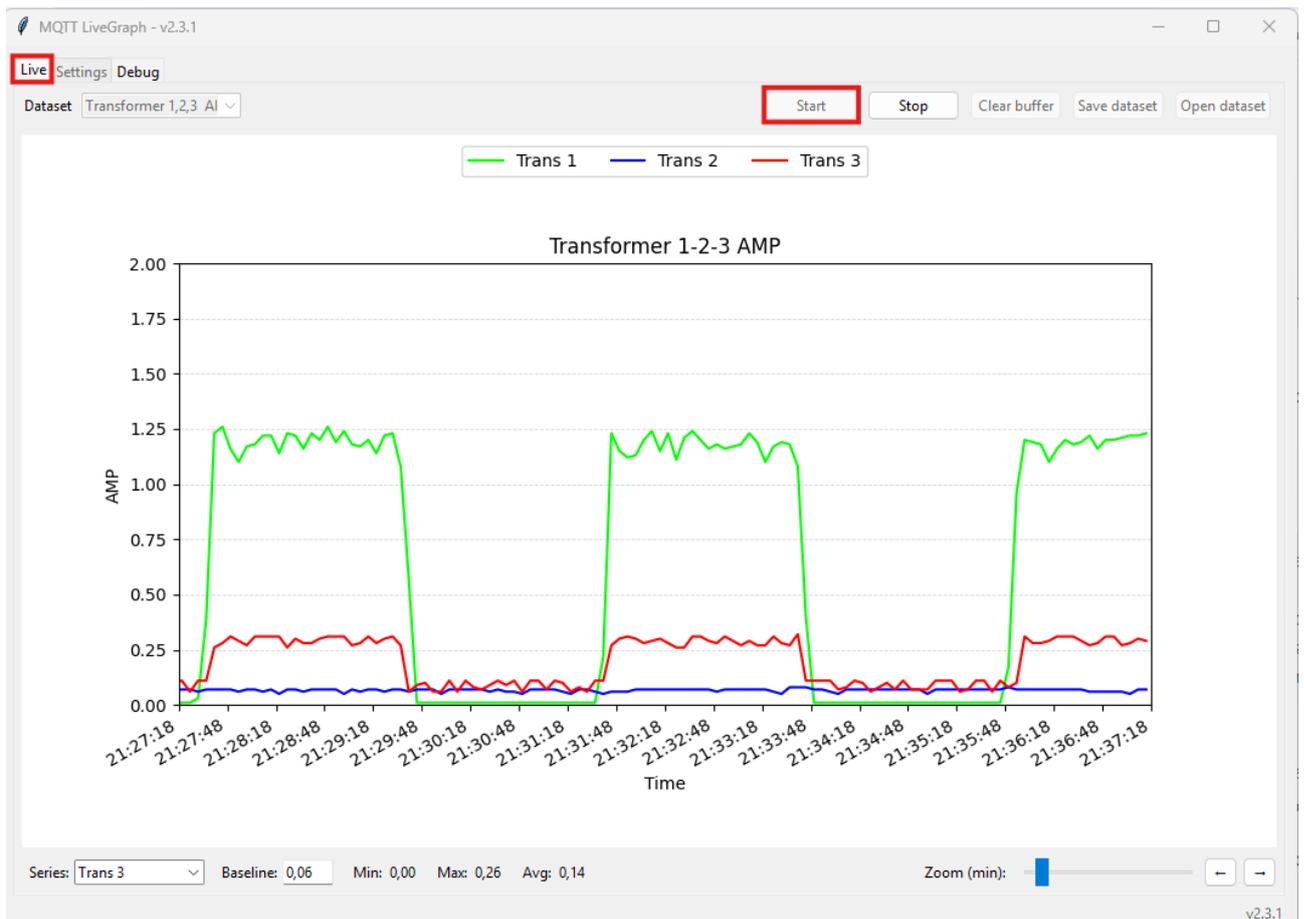Calculation of the transformer load as a percentage of its maximum capacity.

Move up   Move down

With these buttons, you can change the order of the rows in the table view of MQTT topics/formulas.

4. Live View

Start by selecting a dataset — for example the one we just created in the previous chapter.

Click the **"Start"** button, and the data will now be collected live.
In the example, you can see that transformers 1 and 3 are particularly loaded every other minute — this is because they are part of my day/night control system, where day and night each last two minutes.
At night, these transformers are, for example, further loaded by house lighting and similar loads.

Here you can select a graph and view the minimum, maximum, and average values for the chosen time period.

We can also specify a baseline offset. This is useful, for example, if we want to find out how much the consumption of transformer 3 increases at night.
In my model railway — *Kælderkøbing* — night typically lasts 2 minutes, and the same goes for daytime.
The constant consumption is around 0.06 AMP — we can use this as a baseline value to determine how much transformer 3 uses for LED lighting and similar loads during the night.
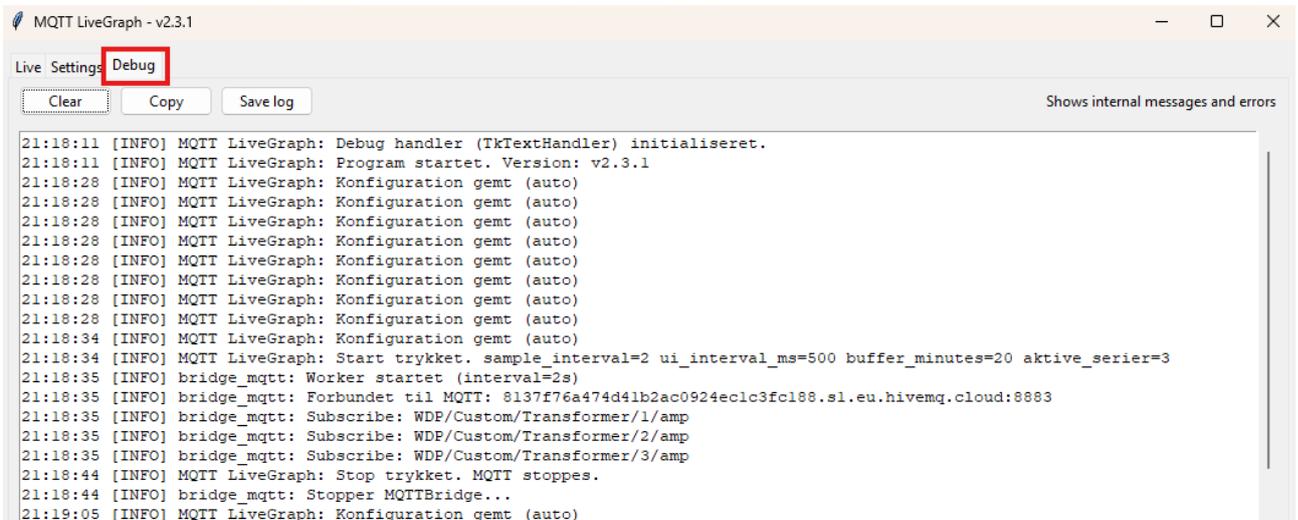

With this feature, you can zoom in and out on the X-axis — and move the entire graph to the left or right.
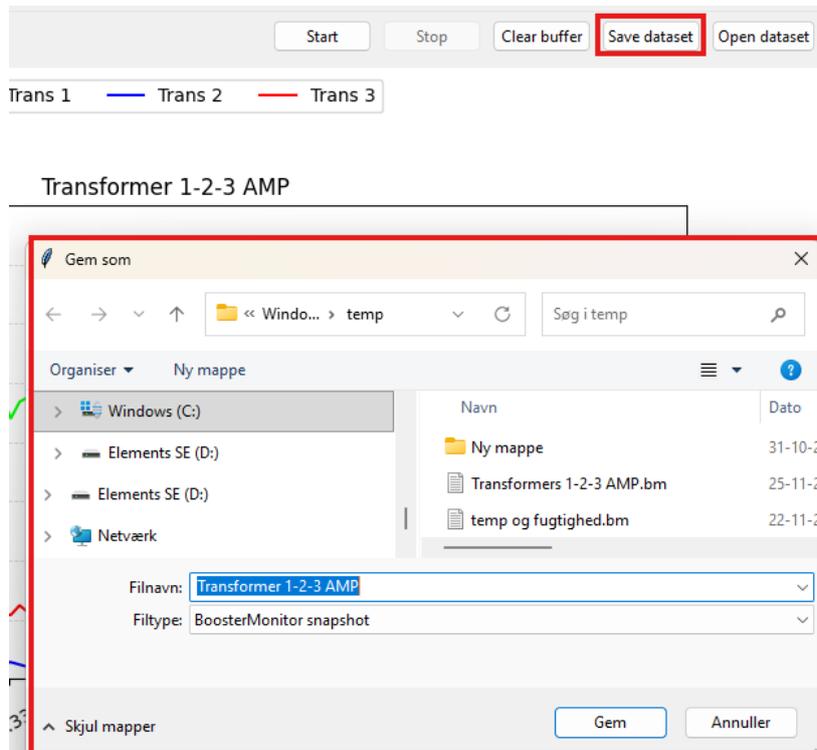
You can also save a dataset so you can reuse it later for analysis and similar purposes.
Before this is possible, however, you must first stop the data collection.

Likewise, you can open a previously saved dataset again ...

5. Debug

This window is primarily used to monitor the system and display useful information if an error occurs.