

CAN data analyses using AI tool

In this article CSS Electronics (Denmark) explains how engineers may leverage a new 'Grafana Assistant' AI tool to explore and visualize large amounts of CAN data.

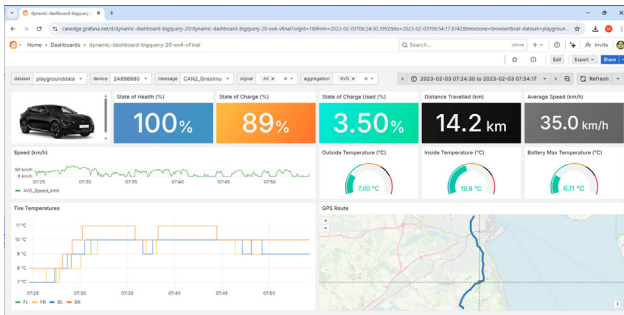


Figure 1: Grafana Assistant vehicle dashboard (Source: CSS Electronics)

Engineers using the products from CSS Electronics face a challenge. They can collect tons of CAN data using the CANedge data logger, but how should they explore/visualize data across several devices, thousands of log files and months/years of data? Most of these users are engineers, but not data scientists. And even if they are, statistical data analysis/visualization can be extremely time consuming.

Why not use ChatGPT?

One solution to this can be to use ChatGPT to help analyze the data. CSS Electronics wrote about it in [this article](#). However, ChatGPT has some practical limitations:

- ◆ Limited to manually uploading small files (e.g., 100-MB CSV)
- ◆ Analysis results are not easy to share
- ◆ Outputs are "static" (e.g., Python-generated plots)

Key features of Grafana Assistant

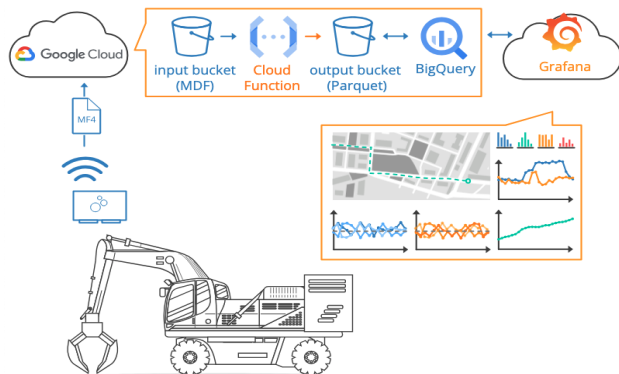


Figure 2: Principle of data acquisition (Source: CSS Electronics)

In practice, CANedge end users typically collect gigabytes (or terabytes) of data to their own Amazon, Google, or Azure cloud server. Here, the CAN data is automatically decoded into Parquet data lakes and can be queried via an open-source dashboard tool called Grafana.

Within Grafana, the new 'Grafana Assistant' is essentially a Large Language Model (LLM) that can access the data and help visualize it. This introduces multiple features:

- ◆ Zero setup: When users have already deployed a Grafana integration, no setup is required and they can start prompting their data immediately.
- ◆ Analyze TBs of data: Unlike ChatGPT, analyses are not limited to e.g., 100 MB of data, the assistant can query an entire data lake out-the-box.
- ◆ Data exploration: Users can now chat their way to powerful data insights and conduct complex diagnostic analyses. No SQL query or coding knowledge required.
- ◆ Dashboard creation: The assistant can create fully customized Grafana dashboards in seconds based on high-level prompts. This drastically reduces the time spent.
- ◆ Retain and share insights: Data insights can be summarized into dashboards, enabling users to navigate the data temporally and share it with their teams.

How to get started



Figure 3: Grafana dynamic dashboard template (Source: CSS Electronics)

For users that have already connected their data lake in Grafana, no further setup is necessary. To start chatting with Grafana Assistant, a user can open the chat window via the left-menu Assistant tab. Alternatively, they can open the chat panel within a dashboard via the icon in the upper right corner.

However, to get the best experience, CSS provides various tips, including the use of a dashboard template (see Figure 3) as the starting point and explicitly informing the assistant about what data source to focus on. In addition, it is useful to provide a specific ‘system prompt’ to the LLM as a ‘rule’ to help it e.g., understand the structure of the CAN data lake structure.

Example use cases

To showcase how the AI assistant can be used, CSS uses their public data pack, which consists of 1 GB of CAN data from a Kia EV6 electric vehicle. Below are some examples of what can be done via the assistant.

Example 1: What data is available?

A good starting point is to get an overview of what data is available. This is a key step in exploring a CAN data lake as users may not necessarily have a clear overview of what devices, CAN frames and CAN signals are available. Further, constructing the SQL queries to extract this information is quite complex.

Prompt:
 My data source contains data from a Kia EV6 electric car, recorded with a CANedge CAN bus data logger with device ID 2A896980. The data includes battery data from the EV and GPS/IMU data from a sensor module. I want to know the following:
 1: What tables and columns are available for this device?
 2: What time period does the data span?

In this case, Grafana Assistant starts analyzing the data via multiple queries, summarizing the results in a tabular form and in the chat. Further, the LLM iterates through multiple queries to understand the data structure. Users can also inspect what queries are used to generate each result, ensuring full transparency. The summary provides a good starting point for further investigation.

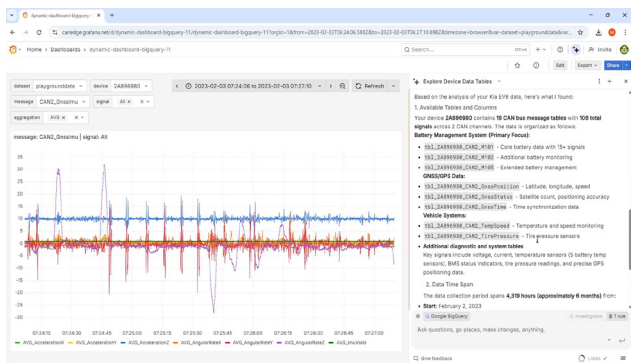


Figure 4: Example 1 data exploration (Source: CSS Electronics)

Example 2: Explore the data ad hoc

In many cases users will want to explore the data directly in the chat window prior to creating actual panels. Here, Grafana Assistant runs the relevant queries and produces the output in-chat as text or as plots. It is also possible to prompt the LLM to create a panel that displays speed over time within a dashboard.

Prompt:
 What is the average speed from the CAN2_GnssSpeed message in July 2023? It's in m/s. Please create a panel in this dashboard to visualize the average speed over time

Calculate Average GNSS Speed July

What is the average speed from the CAN2_GnssSpeed message in July 2023? It's in m/s

I'll help you find the average speed from the CAN2_GnssSpeed message in July 2023. Let me query the BigQuery datasource to get this information.

execute SQL query

The average speed from the CAN2_GnssSpeed message in July 2023 was **19.72 m/s** for device 2A896980. This represents the overall average speed recorded by the GPS system across the entire month of July 2023, indicating consistent moderate speed operation throughout that period.

Figure 5: Ad-hoc data analyses (Source: CSS Electronics)

Example 3: Create dashboard via a high-level prompt

An obvious use case for Grafana Assistant is to create a new dashboard from the scratch. One way to do this is through a minimal-effort prompt, leaving a lot up to the imagination of the AI tool. The result is a functional 12-panel dashboard where all SQL queries are as expected. In particular the LLM proactively identifies available (and relevant) messages/signals.

However, the resulting dashboard is highly inconsistent between re-runs of the same prompt, which is to be expected due to the open-ended nature of the prompt.

Prompt:
 My data source contains data from a Kia EV6 electric car, recorded with a CANedge CAN bus data logger with device ID 2A896980. The data includes battery data from the EV and GPS/IMU data from a sensor module.

Update my dashboard to include panels showing my Kia EV6 battery data and GPS/IMU data.

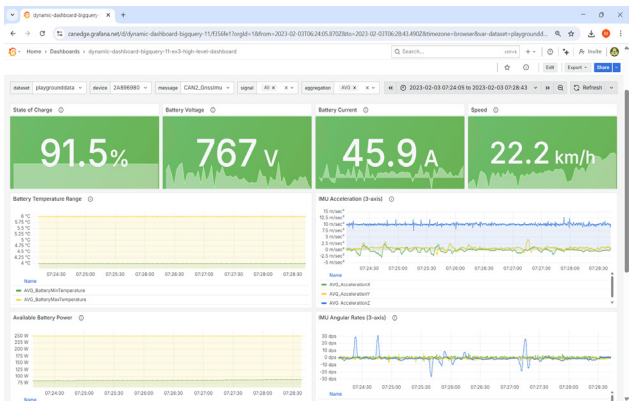


Figure 6: Example 3 dashboard created via a high-level prompt (Source: CSS Electronics)

Example 4: Create a dashboard via a detailed prompt

An alternative approach is to provide a prompt with highly detailed guidance - similar to what users might provide a human assistant if they were to design a Grafana dashboard. The full detailed prompt is available on the CSS website.

In this case, the result looks 'as intended'. And, more importantly, when this prompt is executed ten times (in new conversations each time), the result is a more than 90%-consistent dashboard each time.

It might be argued that writing a prompt like this is also time consuming - but CSS argues that it is still 5-10 times faster than if users were to construct this dashboard from scratch. For example, many of the referenced CAN message/signal names are 'approximate', leaving Grafana Assistant to figure out what the exact table and column names are. Further, some of the panels involve complex queries (e.g., consumed state of charge and delta distance traveled), which would require significant SQL expertise to create. Further, while the prompt is detailed, it specifies no SQL syntax - making it possible to write by any end user. The AI-generated dashboard can be reviewed in the public dashboard playground.

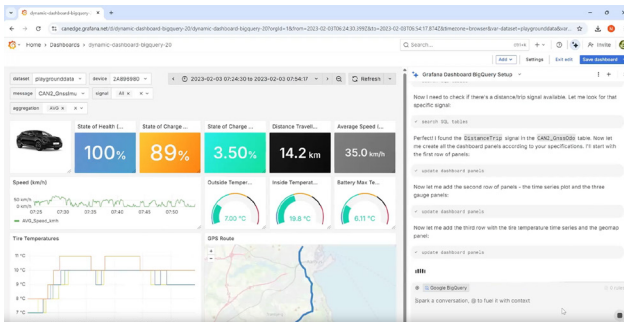


Figure 7: AI-generated dashboard example (Source: CSS Electronics)

Summary thoughts

Based on their extensive testing of the Grafana Assistant AI tool, CSS Electronics concludes that the overall concept is excellent, in particular due to the ease-of-access for non-technical users that have already connected Grafana to their CAN data.

Further, in many cases the AI tool can be a true time saver generating multi-panel customized dashboards from a single prompt and abstracting away the complexity of writing advanced SQL queries. However, the assistant will also sometimes 'get stuck' or hallucinate – making it critical to review the output.

Further, the LLM is able to modify dashboards and execute SQL queries, making it an excellent tool for data visualization and light exploration. However, it is not able to run scripts (in contrast to e.g., ChatGPT), making it less suitable for highly complex multi-step data analysis. It is key to use the right tool for the job. To learn more and see additional showcases, view the [full Grafana Assistant article](#) on the company's website.



Author

Martin Falch
CSS Electronics
contact@csselectronics.com
www.csselectronics.com



CONTROLLERS • GATEWAYS • RELAYS • HMI-SYSTEMS • CONNECTIVITY

- For the **extension** of existing CAN systems or as **standalone** PLCs.
- Free configurability and programmability.
- Supporting **all standard protocols** like CAN 2.0 A/B, CANopen, CAN FD, LIN, SAE J1939, ISOBUS, RS232 or RS485.
- 32-bit products can be used to implement **cybersecurity** requirements in your vehicle, reach out for consultation.



Applics Studio Graphical Programming

- **Simple creation** of control logic by linking several function blocks.
- **Own graphic blocks** can be created with C-code and managed in their own graphical libraries.

