
AI-assisted retrieval and visualization of control-system data from the InfluxDB time-series database at the European XFEL



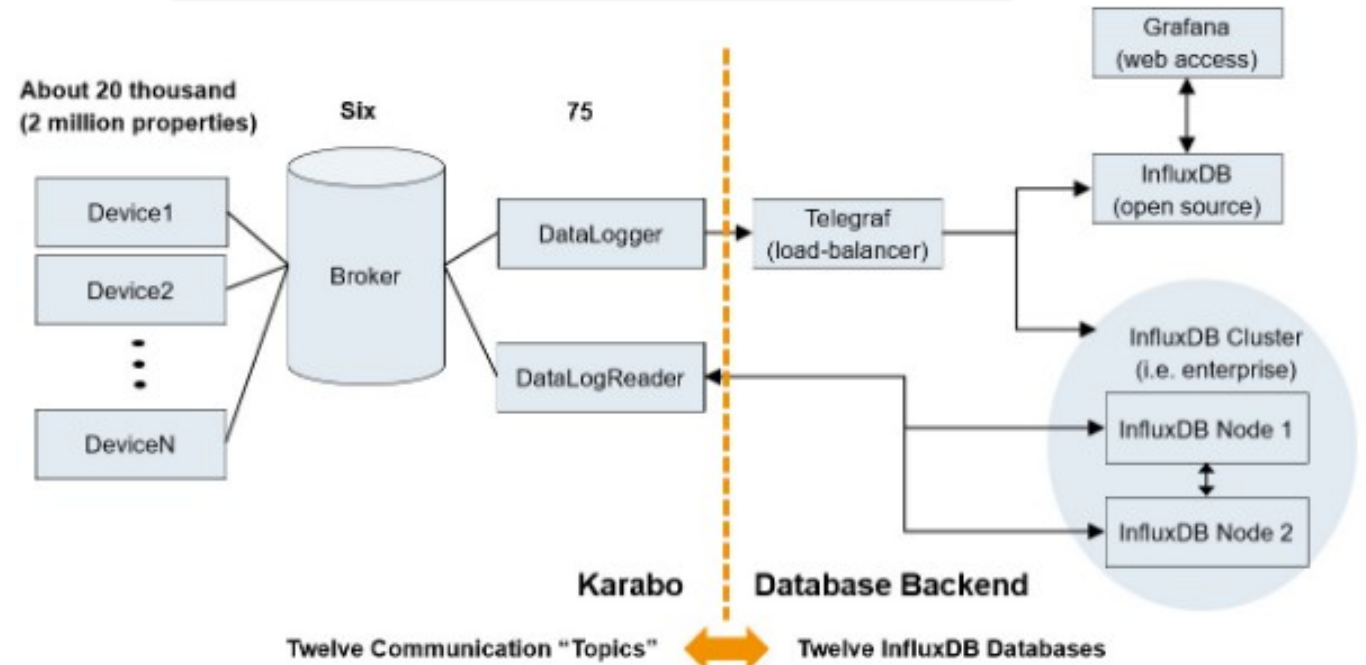
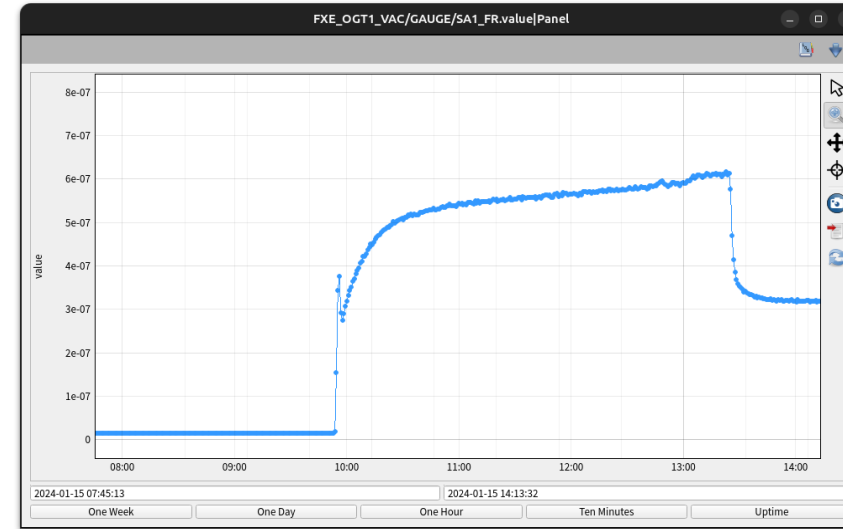
F. Sohn

14.11.2025

8th Round Table on Deep Learning @ DESY

The InfluxDB time-series database

- Continuous logging of control data, such as device settings and low-bandwidth measurements.
- Integrated in Karabo
- Grafana web interface
- InfluxQL (SQL-like) query language
- Facilitates trend analysis, anomaly detection, and performance optimization.



A prototype for an InfluxDB query assistant

- Intended use: Query data from InfluxDB via the Zulip chat tool.
- Allows to integrate data plots easily into the instrument log books.
- No knowledge of Grafana or the Karabo GUI required.
- Automatic data aggregation
- Readiness Level: TRLs 3-4: development

ORG/CTRL/BotTest > Influx query test ✓ ⌵ ⋮ NOV 11



Florian Sohn

15:43

@Influx Query Bot Plot property 'value' of 'FXE_OGT1_VAC/GAUGE/SA1_FR' from 15 Jan 2024, 08:00 to 15 Jan 2024, 14:00.



Influx Query Bot

15:44

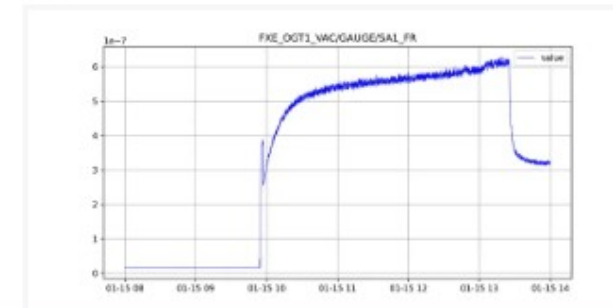
Parameters:

- Device ID: FXE_OGT1_VAC/GAUGE/SA1_FR
- Property: value
- Start time: 2024-01-15T08:00:00+01:00
- End time: 2024-01-15T14:00:00+01:00
- Aggregation: mean, 5s (applied automatically)

Note: the query backend applied automatic aggregation (mean) with a 5s period to limit data points. Aggregation periods smaller than 5s are not allowed; if you need a different aggregation function or coarser/finer period ($\geq 5s$), tell me and I can re-run.

Grafana view: [Open in Grafana](#)

Plot





Explore

FXE

Split

Add to dashboard

<

2024-01-15 08:00:00 to 2024-01-15 14:00:00

>

Run query

A (FXE)

SELECT mean("value-FLOAT") FROM "FXE_OGT1_VAC/GAUGE/SA1_FR" WHERE time >= 1705302000s AND time <= 1705323600s GROUP BY time(5s)

Format as

Time series

Alias by

Naming pattern

+ Add query

Query history

Inspector

Graph

Lines

Bars

Points

Stacked lines

Stacked bars

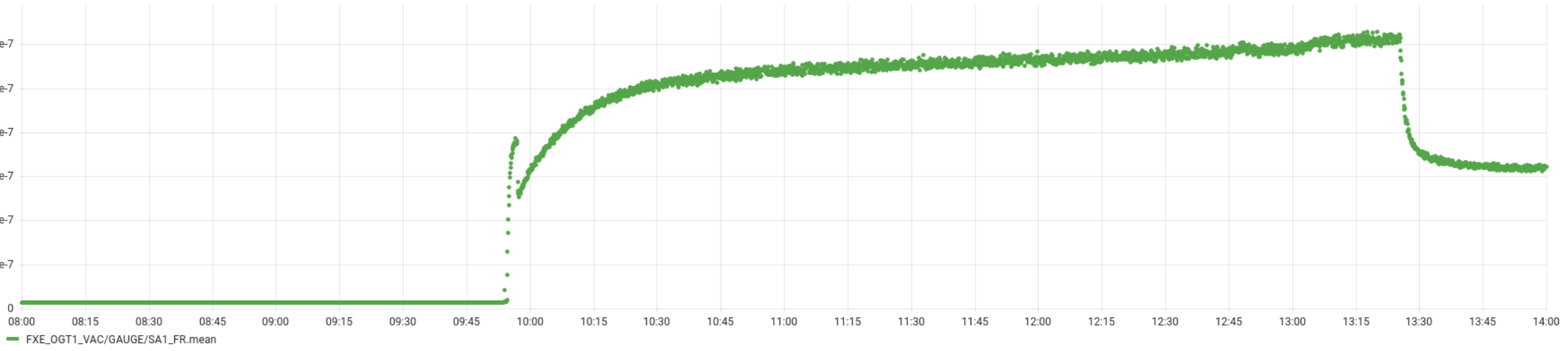
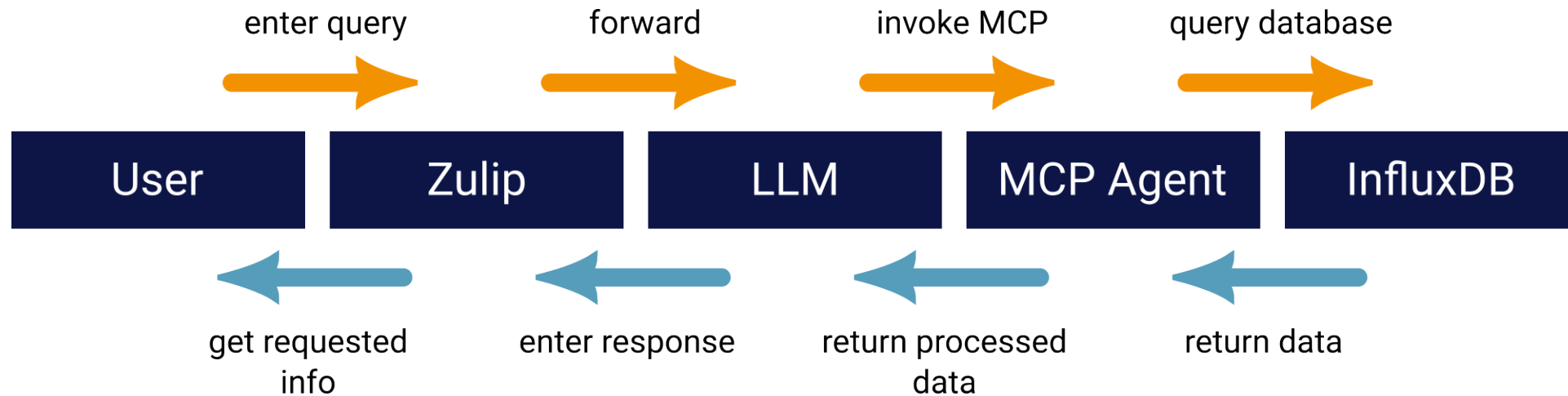


Table	
Time	FXE_OGT1_VAC/GAUGE/SA1_FR.mean
2024-01-15 08:00:00	1.44e-8
2024-01-15 08:00:05	1.46e-8
2024-01-15 08:00:10	1.43e-8

InfluxDB query assistant



■ The LLM extracts query parameters from the user input.

■ The MCP agent queries InfluxDB via a python-based tool.

■ The MCP tool processes and formats the data.

■ The LLM processes the data and generates a response for the user.

Conclusion

- MCP allows to provide the LLM with tools in a generic fashion, allowing for flexible AI assistant implementations.
- Downside of using MCP:
 - Requires more powerful LLMs.
 - Increases computational cost and/or query processing times.
 - Drawbacks might become less relevant over time due to technological evolution.
- Todo: allow more generic database queries
- Nice-to-have: Karabo-agnostic queries, no device IDs and property names.

ORG/CTRL/BotTest > Influx query test ✓ ·)) : NOV 11



Florian Sohn

15:43

@Influx Query Bot Plot property 'value' of 'FXE_OGT1_VAC/GAUGE/SA1_FR' from 15 Jan 2024, 08:00 to 15 Jan 2024, 14:00.



Influx Query Bot

15:44

Parameters:

- Device ID: FXE_OGT1_VAC/GAUGE/SA1_FR
- Property: value
- Start time: 2024-01-15T08:00:00+01:00
- End time: 2024-01-15T14:00:00+01:00
- Aggregation: mean, 5s (applied automatically)

Note: the query backend applied automatic aggregation (mean) with a 5s period to limit data points. Aggregation periods smaller than 5s are not allowed; if you need a different aggregation function or coarser/finer period ($\geq 5s$), tell me and I can re-run.

Grafana view: [Open in Grafana](#)

Plot

