



Summer Student Report

PID Metric Monitoring

Author: Haya Noor

Supervisor: Rafael Figueiras dos Santos, Edward Ireland-Jones

Department: BE-ICS-ECS

Summer 2025

August 29, 2025

Abstract

I was selected for a 12-week CERN Summer Student program. During this time, I contributed to the BE-ICS-ECS group, working on the PID monitoring project for CERN's Cooling and Ventilation (CV) systems. My work focused on developing a data pipeline to extract, process, analyze and visualise PID performance, with an emphasis on identifying PIDs that behaves unexpectedly and those operating outside of defined thresholds. This enabled systematic performance monitoring and visualization, supporting the reliable operation of CERN's CV infrastructure.

The system extracts data points (PIDs) from the NXCALS database using `pytimber` and Spark, focusing initially on CV-related PIDs with the potential to scale to a larger set of PIDs. The pipeline leverages hierarchical paths (e.g., `/CV/LHC/P6/COOL_SF6/CPC_Controller%`) together with suffix-based filtering (e.g., `.MV`, `.OUTOVST`, `.ACTSP`) to simplify PID extraction and automate the collection of relevant signals. This approach has made PID data retrieval considerably easier and more systematic.

Once extracted, the pipeline performs several transformations: cleaning, timestamp deduplication, and forward/backward filling of missing values. The processed data is stored in Parquet format and subsequently analyzed to compute PID-specific performance metrics, such as the standard deviation of MV–ACTSP differences and OUTOVST values. The resulting statistics are persisted in CERN's database on demand (DBOD) PostgreSQL database, thresholds are maintained for monitoring purposes, and the outcomes are visualized in Grafana dashboards.

This workflow enables reliable PID performance tracking, simplifies PID extraction, and provides a foundation for extended monitoring across CERN's infrastructure.

1 Introduction

2 Introduction

Previous work has already explored automatic PID performance monitoring in the context of the LHC cryogenic system [1]. That study demonstrated the benefits of systematically identifying poorly tuned controllers and highlighted the importance of scalable monitoring approaches across thousands of regulation loops. Inspired by that methodology, the current project applies similar principles to the Cooling and Ventilation (CV) systems, while also building a more flexible data pipeline and visualization framework in Grafana.

Previously, the CV department relied on manually navigating through WinCC OA or TIMBER to locate relevant PIDs. There were no established metrics to evaluate PID performance, and all analysis was performed manually. Identifying problem areas was cumbersome and time-consuming, and only historical data was available, without contextual performance indicators.

My contribution was to automate this process. I developed a pipeline to extract PIDs using hierarchical paths, and implemented metrics to analyze PID performance over time. The project now provides a history of PIDs relative to these metrics. Currently, users can view the performance of PIDs in relation to selected metrics via a dropdown menu, simplifying monitoring and analysis. In the Grafana dashboard, users can also explore the performance of each PID separately, as well as compare multiple PIDs in relation to each other through time-series visualizations. Additionally, a dedicated table highlights PIDs operating outside of the specified minimum and maximum thresholds, allowing for quick identification of underperforming controllers.

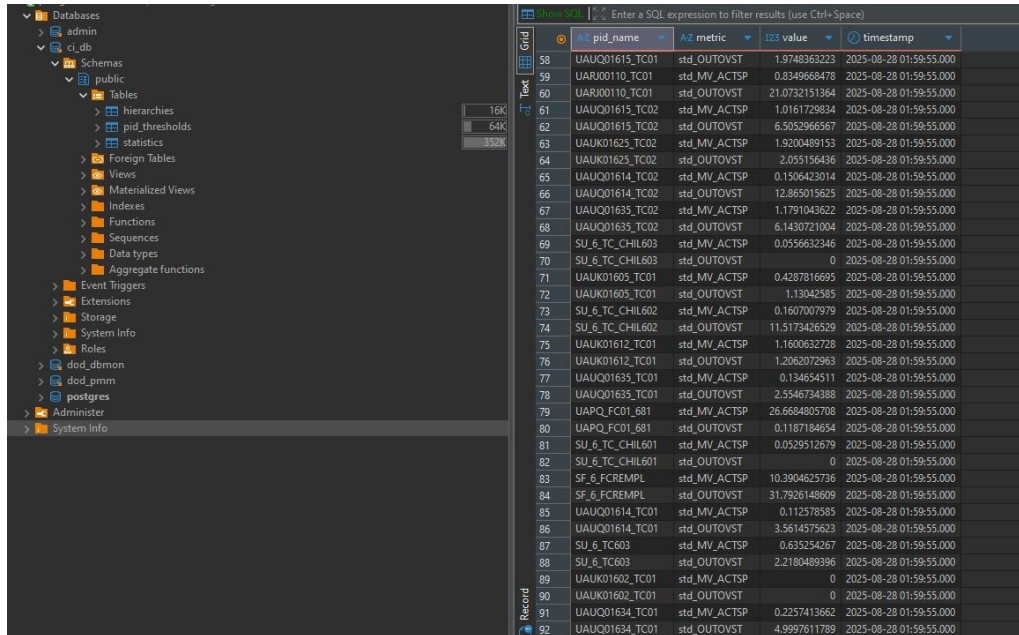
3 Project Workflow

The workflow of the PID monitoring project is summarized in Figure 3. It begins with the main execution script, which extracts variables from the NXCALS database using `pytimber`. The variables of interest are filtered to retain only signals relevant for PID analysis, such as `.MV`, `.OUTOVST`, and `.ACTSP`. The data then undergoes a cleaning stage, where columns are renamed, duplicate timestamps are removed, and missing values are

filled. The processed data is stored as dataframes for further analysis.

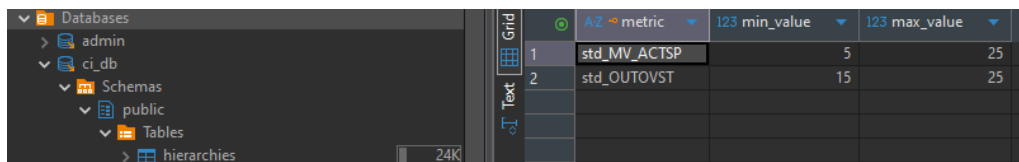
Subsequently, the data is analyzed to compute basic statistics describing PID performance. In parallel, threshold values are defined for monitoring purposes. Both the computed statistics and thresholds are written to the DBOD PostgreSQL database, enabling systematic storage and retrieval. Finally, the results are visualized through Grafana dashboards, allowing users to monitor PID performance interactively and identify under-performing controllers.

Examples of the processed data stored in the DBOD PostgreSQL database are shown in Figures 1 and 2. This table illustrates how PID identifiers, metrics, and threshold values are systematically organized for monitoring and retrieval.



	pid_name	AZ metric	I23 value	timestamp
58	UAUQ01615_TC01	std_OUTOVST	1.9748363223	2025-08-28 01:59:55.000
59	UARI00110_TC01	std_MV_ACTSP	0.8349668478	2025-08-28 01:59:55.000
60	UARI00110_TC01	std_OUTOVST	21.0732151364	2025-08-28 01:59:55.000
61	UAUQ01615_TC02	std_MV_ACTSP	1.0161729834	2025-08-28 01:59:55.000
62	UAUQ01615_TC02	std_OUTOVST	6.5052966567	2025-08-28 01:59:55.000
63	UAUK01625_TC02	std_MV_ACTSP	1.9200489153	2025-08-28 01:59:55.000
64	UAUK01625_TC02	std_OUTOVST	2.055156436	2025-08-28 01:59:55.000
65	UAUQ01614_TC02	std_MV_ACTSP	0.1506423014	2025-08-28 01:59:55.000
66	UAUQ01614_TC02	std_OUTOVST	12.865015625	2025-08-28 01:59:55.000
67	UAUQ01635_TC02	std_MV_ACTSP	1.1791043622	2025-08-28 01:59:55.000
68	UAUQ01635_TC02	std_OUTOVST	6.1430721004	2025-08-28 01:59:55.000
69	SU_6_TC_CHIL603	std_MV_ACTSP	0.0556632346	2025-08-28 01:59:55.000
70	SU_6_TC_CHIL603	std_OUTOVST	0	2025-08-28 01:59:55.000
71	UAUK01605_TC01	std_MV_ACTSP	0.4287816695	2025-08-28 01:59:55.000
72	UAUK01605_TC01	std_OUTOVST	1.13042585	2025-08-28 01:59:55.000
73	SU_6_TC_CHIL602	std_MV_ACTSP	0.1607007979	2025-08-28 01:59:55.000
74	SU_6_TC_CHIL602	std_OUTOVST	11.5173426529	2025-08-28 01:59:55.000
75	UAUK01612_TC01	std_MV_ACTSP	1.1600632728	2025-08-28 01:59:55.000
76	UAUK01612_TC01	std_OUTOVST	1.2062072963	2025-08-28 01:59:55.000
77	UAUQ01635_TC01	std_MV_ACTSP	0.134654511	2025-08-28 01:59:55.000
78	UAUQ01635_TC01	std_OUTOVST	2.5546734388	2025-08-28 01:59:55.000
79	UAPO_FC01_681	std_MV_ACTSP	26.6684805708	2025-08-28 01:59:55.000
80	UAPO_FC01_681	std_OUTOVST	0.1187184654	2025-08-28 01:59:55.000
81	SU_6_TC_CHIL601	std_MV_ACTSP	0.0529512679	2025-08-28 01:59:55.000
82	SU_6_TC_CHIL601	std_OUTOVST	0	2025-08-28 01:59:55.000
83	SF_6_FCREMPL	std_MV_ACTSP	10.3904625736	2025-08-28 01:59:55.000
84	SF_6_FCREMPL	std_OUTOVST	31.7826148609	2025-08-28 01:59:55.000
85	UAUQ01614_TC01	std_MV_ACTSP	0.112578385	2025-08-28 01:59:55.000
86	UAUQ01614_TC01	std_OUTOVST	3.5614575623	2025-08-28 01:59:55.000
87	SU_6_TC603	std_MV_ACTSP	0.635254267	2025-08-28 01:59:55.000
88	SU_6_TC603	std_OUTOVST	2.2180489396	2025-08-28 01:59:55.000
89	UAUK01602_TC01	std_MV_ACTSP	0	2025-08-28 01:59:55.000
90	UAUK01602_TC01	std_OUTOVST	0	2025-08-28 01:59:55.000
91	UAUQ01634_TC01	std_MV_ACTSP	0.2257413662	2025-08-28 01:59:55.000
92	UAUQ01634_TC01	std_OUTOVST	4.9997811789	2025-08-28 01:59:55.000

Figure 1: Example of processed PID data stored in the DBOD PostgreSQL database.



	metric	I23 min_value	I23 max_value
1	std_MV_ACTSP	5	25
2	std_OUTOVST	15	25

Figure 2: Example of thresholds for PIDs stored in the DBOD PostgreSQL database.

The following flowchart illustrates the overall workflow of the PID monitoring pipeline:

- The pipeline is divided into two main parts:
 - Data Extraction Pipeline
 - Data Analysis Pipeline

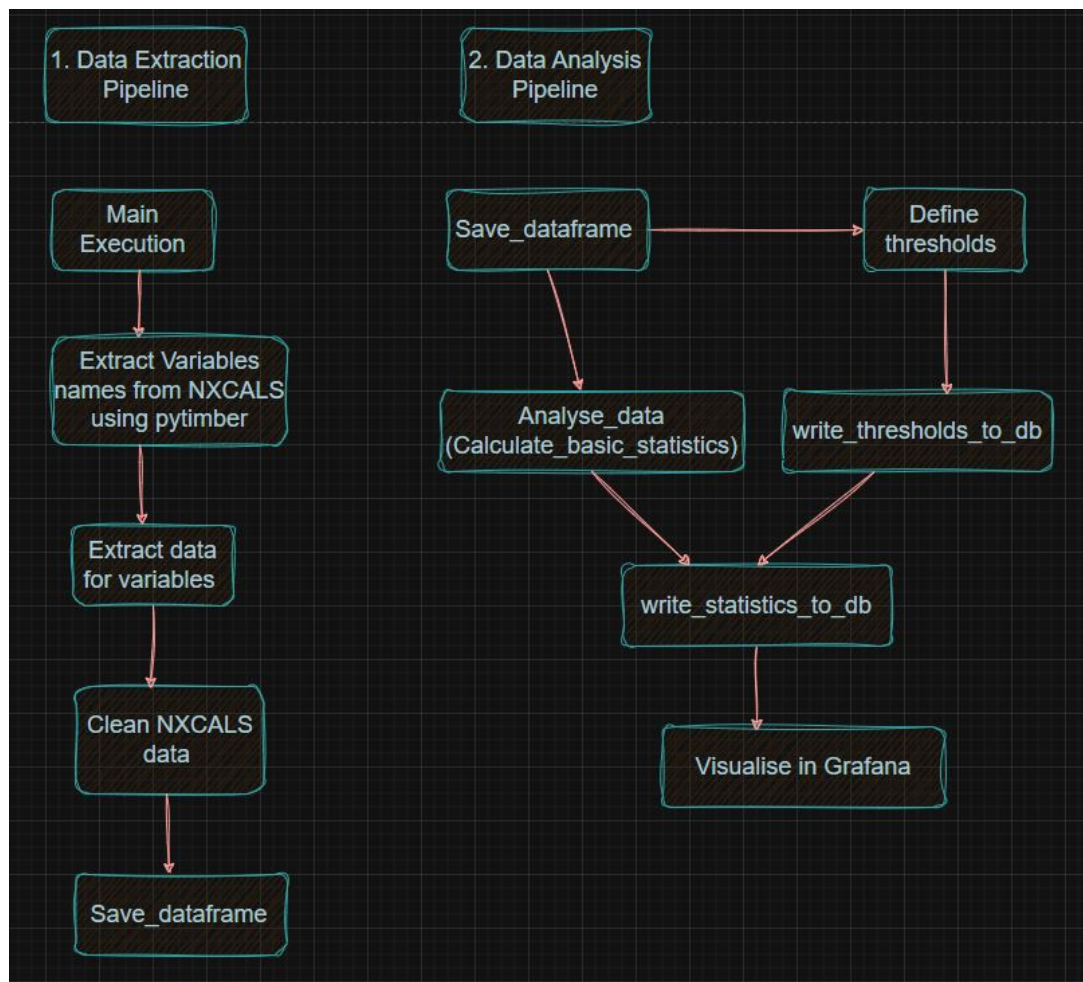


Figure 3: Code flow of the PID monitoring pipeline.

The project workflow can be summarized in three main stages as shown in Figure 4:

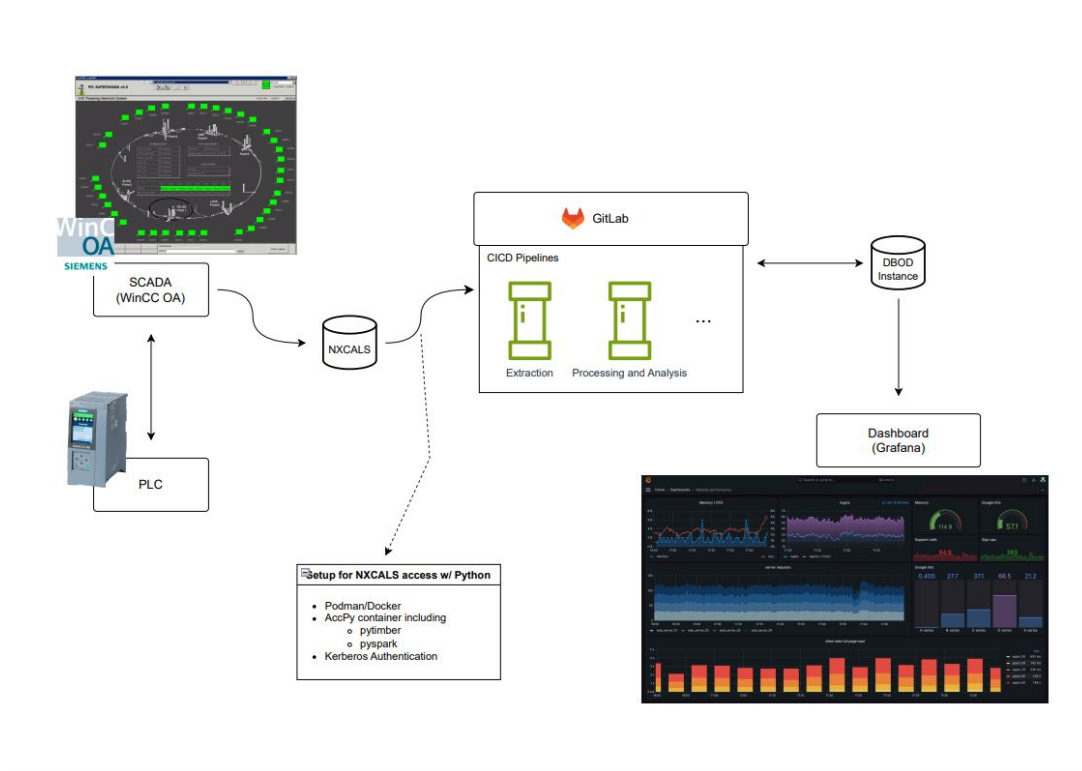


Figure 4: Project workflow of the PID monitoring pipeline.

4 Results

The monitoring pipeline provides clear visibility into PID performance through tabular and graphical visualizations. Figure 5 shows an example of out-of-range PIDs, where calculated statistics are compared against pre-defined thresholds. The table highlights PIDs that are either above or below their expected operating range, allowing users to quickly identify potential problems.

pids

All ▾

metric

std_OUTOVST ▾

Out-of-range-pids

pid_name ▾	metric ▾	value ▾	status ▾	min_value ▾	max_value ▾	timestamp ↑ ▾
UAUT01630_TC01	std_OUTOVST	0	below	15	25	22/08/2025, 18:18:59.000
UAUT01630_TC02	std_OUTOVST	0	below	15	25	22/08/2025, 18:18:59.000
UAUT01631_TC01	std_OUTOVST	1.21	below	15	25	22/08/2025, 18:18:59.000
UAUT01631_TC02	std_OUTOVST	3.86	below	15	25	22/08/2025, 18:18:59.000
UAVP_TC01_620	std_OUTOVST	0	below	15	25	22/08/2025, 18:18:59.000
UAVP_TC02_620	std_OUTOVST	46.2	above	15	25	22/08/2025, 18:18:59.000

Figure 5: Example of out-of-range PIDs detected by the monitoring pipeline.

In addition to the tabular output, time-series plots provide a clearer view of PID behavior over time. Figure 6 illustrates the evolution of PID statistics across multiple controllers, making it easier to spot anomalies, long-term drifts, or sudden deviations. These diagrams reflect PID behavior dynamically, updating every day to display the most recent performance. Users can interact with the visualization to see up-to-date PID behavior and compare it relative to different performance metrics. Currently, the system tracks two key metrics: the standard deviation of `OUTOVST` (`std(OUTOVST)`) and the standard deviation of the difference between measured value and setpoint (`std(MV-ACTSP)`). In the future, additional metrics can be incorporated to provide a more comprehensive view of PID performance.

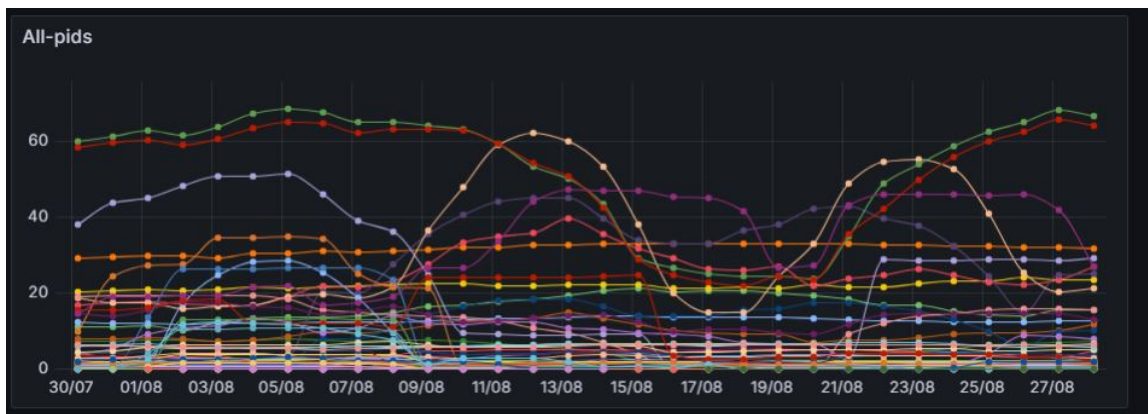


Figure 6: Time-series visualization of PID statistics across different controllers.

Finally, a diagram of a single PID controller is shown in Figure 7 to illustrate the basic setup of one monitored PID. This diagram also updates dynamically and allows

users to see the real-time behavior of the PID.



Figure 7: Single PID controller diagram.

5 Limitations

During the development of the monitoring system, several limitations were identified:

5.1 Grafana Visualization Constraints

Grafana has a significant limitation where the min and max thresholds in visualizations cannot be variables and must be hard-coded. This constraint meant that threshold values had to be specified as data points and plotted directly on the graph rather than being configured as axis bounds. This workaround was necessary to visualize threshold boundaries effectively but represents a suboptimal solution.

5.2 Missing Data Issues

Several controllers exhibited missing data values that impacted the analysis. For example, the UAUK01603_TC02 controller was missing ACTSP (Actual Setpoint) values, as shown in Figure 8. This missing data prevented the calculation of certain metrics and limited the completeness of the performance analysis for these controllers.



Figure 8: Example of missing ACTSP value for UAUK01603_TC02 controller.

6 Future Work

In future work, the plan is to extend the system by including historical data alongside the performance metrics, providing a more complete view of PID behavior over time. Currently, the pipeline queries, analyzes, and visualizes around 216 variables, all of which correspond to CV (Cooling and Ventilation) data points. In the future, this scope will be broadened to include additional data points beyond CV, enabling wider applicability of the monitoring framework. More performance metrics could also be added, and the visualization layer further improved by incorporating graphs that are more relevant for specific use cases. Furthermore, the Grafana interface will leverage hierarchies, allowing users to select PIDs directly from a structured dropdown menu, removing the need to manually navigate through multiple hierarchies.

Additional future enhancements include:

- Connecting to the WinCC OA Oracle database to visualize the data points of the controllers themselves alongside the metrics
- Expanding to include more controllers from different systems
- Adding new performance metrics for comprehensive analysis
- Enhancing Grafana features with hierarchical selection/filtering capabilities

- Implementing better handling of missing data and anomalous controller behavior

References

References

- [1] B. Bradu, E. Blanco Vinuela, R. Martí, and F. Tilaro, "Automatic PID Performance Monitoring Applied to LHC Cryogenics," *Proc. 16th Int. Conf. on Accelerator and Large Experimental Control Systems (ICALEPCS)*, Barcelona, Spain, pp. 1017–1021, 2017. doi: 10.18429/JACoW-ICALEPCS2017-WEAPL02