18th International Conference on Space Operations, Montreal, Canada, 26 - 30 May 2025.


SpaceOps-2025, ID # 272

# Three (now four) years into operations for a satellite health monitoring system and its evolution.

**Agnese Del Moro[a*], Tobias Lesch[a], Daniel Leidreiter[a], Kathrin Helmsauer[a], Clemens Schefels[a], Tobias Göttfert[a]**

[a] Deutsches Zentrum für Luft- und Raumfahrt, German Space Operations Center (DLR-GSOC), Münchener Straße 20, 82234 Wessling, Germany
* Corresponding Author, agnese.delmoro@dlr.de

**Abstract**

Responding to the needs of satellite operators, a modern data visualisation and health monitoring system has been developed at the DLR - German Space Operations Center (GSOC). The web application for data visualisation and analysis (ViDA), which allows fast and interactive long-term telemetry data exploration, has been used operationally at GSOC since 2021. Since then, it has seen an ever-increasing success, with a larger user base and the adoption by additional space missions. The ViDA ecosystem includes a machine-learning based novelty detection service, called ATHMoS, which performs the automatic detection of abnormal behaviour in telemetry data. ATHMoS has been used operationally since 2022, assisting the satellite operators in their routine subsystem checks.

This paper presents the evolution of the ViDA ecosystem in the past four years of operations, describing how the needs and feedback of the operators have been shaping the development of the tool, from the inclusion of more mission's data, to the addition of new features and capabilities, as well as the improvement of the novelty detection and result interpretability. As ViDA keeps growing and evolving, the roadmap for the future is also showcase
**Keywords:** (ViDA; ATHMoS; spacecraft operations; data visualisation; web application; AI integration)

**Acronyms/Abbreviations**

AI for automation of Satellite Health monitoring and Ground Operations (AISHGO)
Anomaly Report (ARI)
Artificial intelligence (AI)
Automated Telemetry Health Monitoring System (ATHMoS)
Database (DB)
European Space Operations Centre (ESOC)
Fondazione Bruno Kessler (FBK)
German Space Operations Center (GSOC)
Geostationary Earth Orbit (GEO)
High Priority (HP)
*Intrinsic Dimensionality (ID)*
Low Earth Orbit (LEO)
*Low Priority (LP)*
Machine Learning (ML)
Mission Operations Technology (MBT)
Mission Data Access (MiDA)
*Monitoring and Control System (MCS)*
On-board Event (OBE)
Out of Limit (OOL)
*Outlier Probability via Intrinsic Dimensionality (OPVID)*
Recommendation (RI)
Satellite health Monitoring AI-powered suite (SMAI)
Telecommand (TC)
Telemetry (TM)
Trending detection (TD)
Visualisation and Data Analysis (ViDA)
User interface (UI)

## 1. Introduction

The work of satellite operations is becoming more and more challenging, due to the increasing complexity of modern satellites and the number of satellites each operator needs to monitor. Space agencies nowadays are investing a lot of resources in research and development of new methods and tools for automating some of the engineers' tasks by using modern technologies to assist them in their daily work.

With this scope, the Visualisation and Data Analysis (ViDA) tool and framework was developed at the German Space Operations Center (GSOC) at the DLR. ViDA is a web application that allows fast, full-lifetime time series plots of satellite telemetry (TM) for monitoring the spacecraft status. It is a multi-mission tool, which provides highly customisable visualisations of satellite data, analysis workspaces and immediate overviews of the health status of all satellites through dashboards [1], supported by a machine-learning (ML) based novelty detection service, the Automated Telemetry Health Monitoring System (ATHMoS), also developed at GSOC, which monitors the TM daily [2; 3].

Starting off as a prototype, the development of the operational software started in 2019 and it was first released operationally in-house at GSOC in July 2021. In the past 4 years of operations, the close communication with the end users and the identification of new solutions to alleviate their "pain-points" helped shaping ViDA and increasing its usage and success.

In this paper the evolution of the ViDA framework during the past 4 years is presented. In section 2 an overview the software and infrastructure is reported. Section 3 details the user-feedback workflow adopted in the course of the development. Section 4 describes how the software has evolved up to the current status, focusing on the newly implemented features and capabilities. In section 5 the on-going projects and collaborations adding new features to the ViDA framework are presented. To conclude, in section 6 a brief view of the roadmap and new extensions planned for the future are described.

## 2. What is ViDA?

ViDA is an interactive, web-based application designed to efficiently explore satellite TM and various types of data generated by space missions. It serves as more than just a telemetry display tool by incorporating features from business intelligence, data science, and artificial intelligence (AI), tailored for multi-spacecraft operations.

As key functionalities, ViDA provides a multi-mission environment, with the ability to support long-term spacecraft monitoring, offering near-instant plotting of TM data spanning the entire mission duration. The user-friendly and reactive interface allows users to tailor visualisations based on their specific selection of key TM parameters. To support engineers in identifying unusual behaviour or emerging patterns, the ViDA framework integrates the ATHMoS novelty detection service—a ML-based system purpose-built for identifying novel patterns in large-scale satellite telemetry data (Sect. 2.1), and displays its detection results daily in dashboards and via highlighting specific time ranges in time-series plots that help pinpoint deviations and trends in the data. However, for satellite operators, the real challenge lies in understanding the root causes behind unusual TM readings, especially to spot early signs of hardware degradation. Early detection enables faster, more informed responses to potential issues. To aid deeper investigation, ViDA provides flexible workspaces where users can build and save custom visualisations tailored to their analysis needs.

For more in-depth and user-specific investigations, the ViDA framework includes a powerful centralised analysis environment built on JupyterLab and provides an easy-to-use API, known as MiDA (Mission Data Access), which gives direct and secure access to telemetry and other relevant data sources in the DB.

Overall, ViDA is a powerful tool that enhances the way space mission data is visualised and analysed. By integrating advanced visualisation techniques, machine learning capabilities, and analytics, it helps mission teams gain deeper insights and make well-informed decisions more efficiently. Its ability to unify diverse data sources, facilitating inspection and investigation, makes it an essential component of modern space operations, contributing to mission success and operational excellence.

### *2.1 Technical characteristics and infrastructure*

ViDA is an off-line application with a modular and scalable architecture to handle the complex demands of space mission data analysis. It is accessible from any browser in the GSOC office environment via a role-based access control, ensuring data security and compliance to mission requirements. The whole framework around the application includes:

- Web-Based Architecture: ViDA is a web application based on the Angular framework [4], a development platform built on Typescript, providing an interactive and reactive single-page application. This allows access from multiple devices without requiring local installations.

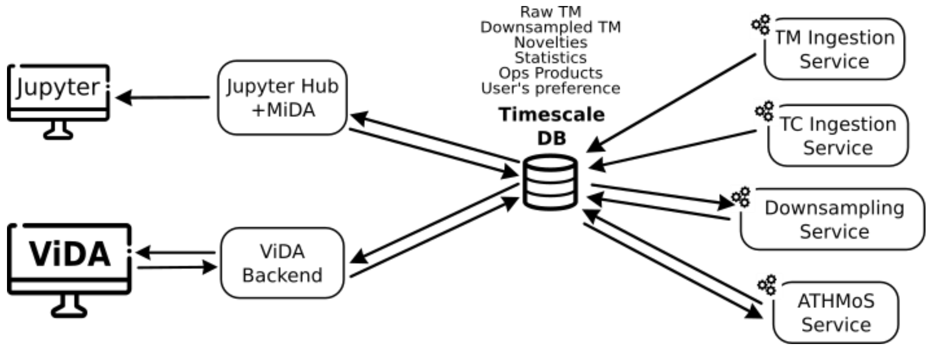


Fig. 1. Schematic view of the ViDA infrastructure, updated from Fig.5 of [1]

- Backend: The backend system is built using Scala [5] and the Akka framework [6], which supports concurrency, asynchronous communication, and scalability via actor-based messaging and is optimised for high-performance data processing. It manages database access for satellite data and user preferences while offloading resource-intensive processing from the frontend. It serves pre-processed data via a GraphQL API, through queries restricted to authenticated users. Authorization is handled using JSON Web Tokens, validated by an internal GSOC server. The backend connects to the multi-mission DB, providing TM data, metadata, anomaly detection results and additional contextual data sources (see Sect. 3.3.1).

- Database: The central data storage of the ViDA framework uses TimescaleDB [7]. As a PostgreSQL extension, it offers full SQL support, very good performance in insert rate and query speed, and two key features: automatic downsampling, which improves query efficiency by aggregating long time-series data, and native compression, reducing disk usage, by 98% in our case, through optimised storage techniques (see Sect. 3.2). TimescaleDB is deployed on a RAID10 setup and has been running smoothly.

- Data services: The ViDA data ingestion service efficiently imports housekeeping TM and operational data into a central relational database. Built using Scala and Akka—like the ViDA backend—the ingestor processes pre-formatted plain text files (time-series tabular structure) and maps them into a normalised schema (one row per parameter per timestamp). Leveraging Akka Streams, it establishes resilient, parallel data pipelines that scale with CPU cores, limited only by disk I/O throughput. Batch inserts via Akka's Slick connector further enhance performance, enabling ingestion at high throughput. The system is deployed as multiple containerised instances, distributed across nodes, with each instance dedicated to a specific mission. Collectively, these instances handle over 700 million rows per day. For example, one mission alone produces ~230 million rows daily, ingested in just 15 minutes using a 4-core, 32 GB RAM setup co-located with the database [1]. New data services, handling new spacecraft data type such as telecommand (TC) logs and TC acknowledgement history, have now been added to the framework, as described in Sect. 3.3.

- AI-based novelty detection service: ATHMoS ([2], [3]) is aimed to enhance the monitoring of spacecraft telemetry data. Its primary function is to detect novel behaviour in TM parameters that may not be identified by traditional out-of-limits (OOL) checks, thereby providing early warnings of potential issues. At the core of ATHMoS is an algorithm that utilizes the Intrinsic Dimensionality (ID e.g., [8], [9]) of a data set for statistical outlier detection. By analysing historical telemetry data, ATHMoS employs a semi-supervised ML methodology to build databases of nominal and anomalous behaviour. When new telemetry data are analysed, the system uses the "Outlier Probability via Intrinsic Dimensionality" (OPVID, [2]) to assign anomaly probabilities, based on the density of the data cluster and the distance of the new data points to those of the nominal cluster. It then classifies the novelties as high priority (HP), low priority (LP) or trending detection (TD), based on their anomaly probability and the number of consecutive time windows that are found to be non-nominal. ATHMoS has been used operationally at GSOC since 2022 [3; 10] and ViDA is the main user interface displaying its results, by providing operators with intuitive visualisations and daily analytics to support decision-making processes. This integration enhances the efficiency and effectiveness of spacecraft operations by enabling proactive monitoring and response to potential anomalies.

- Infrastructure: Since the ViDA framework is designed as a multi-mission, fleet-capable solution rather than a single application, its architecture is therefore reliable and scalable to accommodate growing demand. Moreover, it is intended to support deployment in different configurations based on security or data locality requirements. The services comprising the whole system are running on Linux servers, using virtualisation and containerisation technology. Currently, the Docker containers are running the software and services on a few servers, and regular instances of PostgreSQL databases as the basis for TimescaleDB. The total resources available to the ViDA framework are currently 448 GB RAM, 40 CPUs and 36 TB of disk space. These resources are estimated to be suitable for supporting up to 10 satellites over their full lifespan (10-20 years) with the currently available features.

## 2 User feedback and agile development

Since the initial implementation and throughout the entire development of ViDA, a prominent weight has been given to direct communication with the end-users and their needs. The software has been implemented following a (loose) agile development workflow, which is an iterative and flexible approach that emphasises collaboration, adaptability, and continuous improvement. User feedback is crucial in agile development as it ensures the product meets real-world needs by *i)* identifying usability issues and functional gaps; *ii)* prioritising features based on actual user needs; *iii)* improving product quality through iterative refinements, thus reducing development risks and increasing customer satisfaction and engagement.

For the development of ViDA, the engagement of the users was not limited to the standard "Sprint Reviews" foreseen as part of the agile workflow, but it has been expanded by direct, more informal communication between the ViDA Product Owner, developers and the end users within GSOC, facilitated by the fact that the software is developed and used in-house, as well as by providing, directly in ViDA, a "Feedback" space for the users to write problems, bug reports, suggestions and new ideas. The input from the users is then checked regularly and, based on priority and feasibility, incorporated in the development backlog, therefore helping shaping and evolving the product on real usage and needs.

## 3 Four years of operations: the evolution of ViDA

The first release of ViDA at GSOC, in 2021, was only serving 2 LEO missions – 2 satellites – and provided very limited features, namely the main long-term time series plotting capability for single TM parameters, compared to what the software offers now. Over the past four years the ViDA framework has grown and evolved to include the ATHMoS novelty detection service, made operational in 2022 and analysing the TM of those satellites since then. New plotting capabilities, such as multi-parameter plots and the pre-configured, automatically-generated plots, called "autoplots", were then implemented and made available to the users daily. Subsequent main features have been the user dashboard, providing an immediate overview of all satellite's status by summarising the latest novelties detected by ATHMoS, and the satellite-specific dashboards, listing the novelties in more details. For a full overview of the entire ViDA framework up to 2023, we refer to [1].

Throughout this evolution, the user-base of ViDA has also grown within GSOC, on-boarding more missions over the past years, and therefore managing and serving larger amount of data. Currently, ViDA is supporting 5 different LEO and GEO missions at GSOC, counting a total of 9 satellites. This has led to changes in the database and infrastructure as well. In the following subsections the current status of ViDA is presented, detailing the latest developments of the whole framework.

### *3.2 Database and infrastructure*

The addition of new missions in the ViDA framework involved the ingestion of the historic TM data of each satellite since the beginning of the mission, largely increasing the storage usage and the database load during the ingestion times. This reflected on the performances of the application, which were hindered by the data load and drastically slowed down, impairing the usability of ViDA. Therefore, an upgraded of the DB to a newer TimescaleDB version was performed, which allowed the migration of the continuous aggregate tables, holding the downsampled TM data, into hierarchical compressed continuous aggregates. This change significantly improved the speed of the creation and updating of the aggregates, while also improving the query performance, since the data are compressed. Moreover, the data compression reduced the storage usage, leaving more available resources for new data.

### *3.3 Data ingestion services*

In the past years, the data ingestion service in ViDA has only been catering the TM data ingestion. This service receives pre-processed, off-line housekeeping data in plain text file format and converts them into the data model defined to store the data into the DB.

Progressing along the ViDA roadmap, placing ViDA as a centralised tool for users to access various types of data to add contextual information alongside of TM data, the ingestion of operational products into the DB has started. As a first step, the priority has been given to TC Logs and TC acknowledgment history (TCAH), and a new data service to parse and store these data has been implemented (Fig.1).

#### *3.3.1 TC Log ingestion service*

The TC Log ingestion service takes as input the TC logs and TCAH files generated by the offline data processing system. The service, implemented in Python, parses each TC log file row by row keeping the data of an entire file in memory and writing them to the DB in batches. Subsequently, the TCAH are parsed and matched to the corresponding TCs in the DB, based on the TC sequence counter, process ID and the TC execution time, which should be close to the generation time of the acknowledgment. Once all the matches are gathered, the TC acknowledgements are also ingested into the DB, in a unified table together with the TCs. The data ingestion through this service is very fast and efficient. So far, the full history of TCs for 2 of the satellites served by ViDA (~10 years of data each) was ingested into the DB in only a couple of days.

Regular ingestion of upcoming data is also foreseen (but not yet enabled) and handled by the service by temporary caching the TCAH files to ensure that all the related TC Log files are ingested first. Once the TCs are in the DB, the TC acknowledgments can also be stored.

### *3.4 Extended ViDA features*

The latest releases of the software have included several new features aimed at supporting and facilitating the investigation of the satellite status by the operators. The main additions can be grouped in two main categories: i) adding explainability to the novelties detected by ATHMoS; ii) enrich the TM with contextual information, specifically, the sent and executed TCs.

#### *3.4.1 Explainability of novelty detections*

One of the first steps spacecraft operators typically do for checking the status of a specific sub-system is visualising the TM parameters of interest. To support this inspection in ViDA, the results from ATHMoS are shown, together the related TM in time series plots, highlighting the time ranges when HP novelties or TDs were found (Figs. 2 and 3). However, unless there is a very obvious deviation from the nominal behaviour, it is not always trivial to understand why the algorithms classified some time ranges as not nominal.

To ease and speed up the investigation of these occurrences, users are now provided with a dynamic table listing all the ATHMoS detections, which adapts to the TM displayed in the related plot as the users modify the visualisation (zoom-in and out, add/remove parameters). Moreover, each table row can be expanded showing the statistical features (composing the feature vector calculated by the ATHMoS algorithms) that have contributed the most to identifying that particular time range as non-nominal (Fig. 2).

In addition, ViDA now provides the possibility to add comments to each detected novelty, link it to existing anomaly reports (ARIs) or recommendations (RIs), assign labels (e.g. if the time range corresponds to an Eclipse, Sun/Moon blinding, a maneuver, etc.) and also re-classify the novelty, e.g., in case of a false positive detection (see Fig. 3). The user input is then stored in the DB and visible to all other users to avoid work repetition and increase efficiency. Moreover, the user re-classification of the novelties, after a validation stage, can then be used as input by ATHMoS to improve its models and the future results, minimising the chance of false positive detections.


Fig. 2. Novelty table with feature vector contribution to the novelty detection.


Fig. 3. Anonymised TM time-series plot and associated novelty table, showing assigned labels, comments and ARIs/RIs, as examples.

### *3.4.2 Contextual information*

ViDA has always been designed and intended as a centralised tool, able to access different data sources to provide the users with a panchromatic view of the events happening on a spacecraft and their possible causes and consequences (Sect. 2). To this end, the new TC log ingestion service has been implemented (Sect. 3.3.1), as the first step towards the inclusion of all operational products supporting space operations (OBE history, OOL, etc.). An easy and fast access to such information will have a significant impact on the efficiency of spacecraft operators' work, when investigating problems and potential anomalies that can lead to system failures.

ViDA is now providing a very effective way to visualise these contextual data, currently the TCs, together with TM parameters in time-series plots (Fig. 4), allowing the users to have an immediate view of the cause-effect chain of the executed commands and spot potential problems and unexpected behaviours. Since this type of visualisation does not allow displaying the thousands of commands sent to a spacecraft daily, appropriate filtering and time limitations for displaying TCs are being implemented.

To overcome the limitations of the plotting capabilities in displaying the TC details, an additional table view, listing all the TCs and related information is therefore planned to be implemented in ViDA, as requested by the users. In the near future, these visualisations will extend to all operations products available in the ViDA DB.

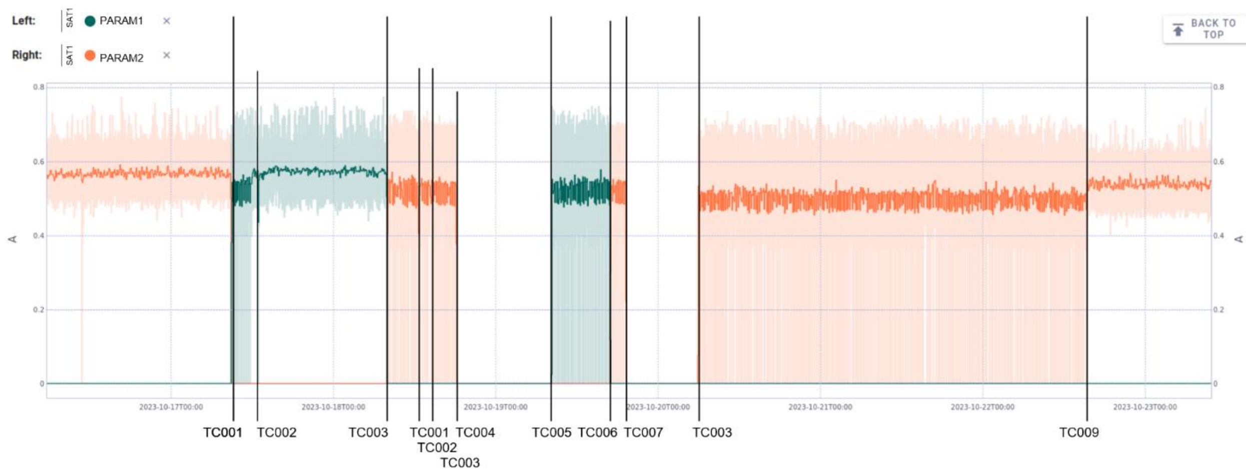


Fig. 4. Prototype example of the visualisation of TCs together with TM parameters in time series plots.

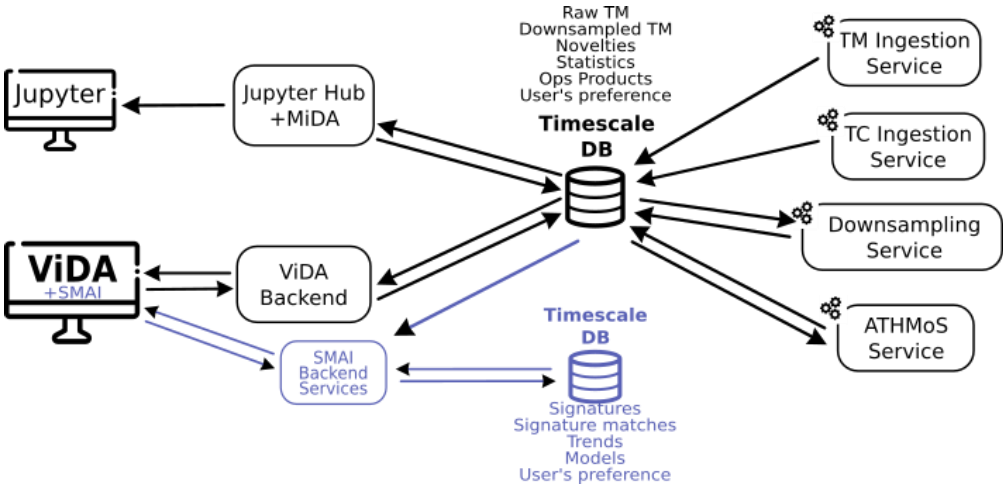


Fig. 5. Schematics of the ViDA infrastructure adapted to integrate the SMAI extension.

## 4 Ongoing projects and add-ons: SMAI

In the past last year, a new project, the "AI for automation of Satellite Health monitoring and Ground Operations" (AISHGO), has been taken on by the MBT team together with other industry partners (*Solenix*, *Starion*, *FBK*, *Airbus*). The aim of the project is developing new AI-based applications and services for space operations, as part of the ESA's Artificial Intelligence for Automation (A2I) roadmap [11]. The resulting products are intended to be generic and, by the end of the project, adopted in operations at the European Space Operations Centre (ESOC) and other non-ESOC environments, namely GSOC, for the satellite operations topics. As part of the project, the two topics related to satellite operations that were investigated and developed are: i) Intelligent telemetry data anomaly detection and ii) Long-term trend analysis and reporting, which resulted in the development of a new tool, called SMAI (Satellite health Monitoring AI-powered suite) [12]. The capabilities offered by SMAI consist of the detection of recurrent spacecraft behaviours, named "signatures", defined by the users as a set of related TM parameters, TCs and/or OBEs, in past and upcoming data. An AI-based pattern detection service has also been implemented as part of the solution, enabling easy recognition of recurring patterns in TM data. The second main functionality is the identification of long-term trends in TM data, and the deviation from these trends, supporting the users in easily assessing the status of the spacecraft sub-systems and reporting of any potential problem.

The project is currently in the operationalisation phase, where the SMAI functionalities are being fully implemented for operational use and the tool is being integrated into ESOC and GSOC environments. Given the affinity of the topics of the AISHGO project with the scopes and roadmap envisioned for ViDA, an obvious choice for the operationalisation of SMAI at GSOC was to integrate the tool as an add-on into ViDA. This choice was made also following the wish of the end-users, both at ESOC and GSOC, of not having to adopt yet another application in their routine work, but preferably having new functionalities added to already adopted applications.

### *4.2 Integration into ViDA*

Since the intention to integrate the final product into already existing tools was expressed early on in the project, the choice of technologies for the implementation of SMAI was made in line with the technologies used in ViDA and in NgWebMUST (at ESOC) [13], to ease the integration in the operationalisation phase. Indeed, the SMAI frontend is implemented using Angular and Typescript, requiring small adaptations on the ViDA side to incorporate the additional features and new stand-alone components.

The SMAI backend is implemented as modular Python services, which are Dockerised and connected via their URLs and exposed ports. An additional element of the tool is a TimescaleDB database, for storing the user-defined signatures, signature matches, trends definitions, models and user's preferences; the DB also runs as a Docker container and connects to the other components in the same manner.

Fig.5 shows a schematic view of how SMAI is planned to be integrated within the ViDA framework infrastructure. The ViDA backend will remain independent from SMAI. On the other hand, the SMAI backend requires the connection to the existing infrastructure, namely, the frontend for displaying the SMAI analysis results and the ViDA DB for retrieving TM data and ops products. To this end, a new data loader service is being implemented to securely serve the missions' data to be analysed by the SMAI backend services, making use of the ViDA user authentication token to preserve the user permissions throughout the whole workflow and avoid unauthorised data access.

## 5 Discussion and future

After having laid a solid foundation for a reliable, scalable and user-friendly application, ViDA has been growing and evolving over the past years, responding of the specific needs of the spacecraft operators. The increased number of served missions and growing user-base testifies the success of this application and the benefits it brings in supporting the engineers in their everyday work.

Despite the new functionalities implemented and under development, the ViDA roadmap has not reached the end yet, as the plans for further improvements and developments are also growing and evolving as time goes by. Given the increased number of services, thus increasing computational demand and complexity in functional orchestration, a migration to a cluster/orchestration solution, such as Kubernetes, is foreseen. An existing Kubernetes cluster set-up has already been instantiated in-house at GSOC, which could easily host the portable components and services comprised in the ViDA framework.

Another important step in the roadmap is making the application and its features more accessible and sharable. The most feasible and immediate solutions that are planned are the capability of exporting plots, autoplots, and workspaces thus facilitating sharing work between users and report writing. With this feature, ViDA will be able to replace the functionality of an old and outdated tool still in use in-house, which could then be discarded, making the ground segment more cohesive and efficient.

A bigger milestone for making ViDA more accessible to the users will be allowing external access to the web UI. As mentioned in the previous sections, ViDA is currently accessible from GSOC premises only. This inevitably limits the time the operators can spend using the tool. This limitation has become more severe in the past years, since the DLR company policy now allows for partial time of working from home, measure implemented after the COVID-19 pandemic and kept since. Users have therefore expressed the wish of accessing ViDA from home (ViDA@Home). This will have significant implications on data security and compliance, which need to be carefully evaluated. Limiting the functionality of the ViDA@Home version and increase restrictions on the user authorisations and data access could be part of the answer. However, a full concept and design for such a solution is still ongoing and under evaluation.

**Acknowledgements**

The authors are thankful to Steffen Zimmermann, Dr Edith Maurer, all members of the MBT-Team and Heavens-Above GmbH for their valuable support and fruitful discussions.

**References**

[1] A. Del Moro, M. Dauth, T. Lesch, C. Schefels, A. Braun, V. Filip, T. Göttfert, A Modern Approach to Visualise Structured and Unstructured Space Missions' Data, 17th International Conference on Space Operations (SpaceOps 2023), Dubai, United Arab Emirates, 2023, 6 - 10 March.
[2] C. O'Meara, L. Schlag, L. Faltenbacher, M. Wickler, ATHMoS: Automated Telemetry Health Monitoring System at GSOC using Outlier Detection and Supervised Machine Learning, 14th International Conference on Space Operations, pp. 2347, 2016.
[3] C. Schefels, L. Schlag, A. Del Moro, K. Halmsauer, T. Lesch, T. Göttfert, Bringing a Machine-Learning Based Novelty Detection Software Tool from Research to Production, 17th International Conference on Space Operations (SpaceOps 2023), Dubai, United Arab Emirates, 2023, 6 - 10 March.
[4] Angular, https://angular.io/, (accessed 04.04.25).
[5] Scala, https://www.scala-lang.org/, (accessed 04.04.25).
[6] Akka, https://akka.io/, (accessed 04.04.25).
[7] Timescale, Inc., TimescaleDB, https://www.timescale.com/, (accessed 04.04.25).
[8] L. Amsaleg, O. Chelly, T. Furon, S. Girard, M. E. Houle, and M. Nett, Estimating Continuous Intrinsic Dimensionality, National Institute of Informatics Technical Report, Tokyo, Japan, March 2014.
[9] M. E. Houle, Dimensionality, Discriminability, Density and Distance Distributions, IEEE 13th International Conference on Data Mining Workshops, December 2013, pp. 468-473.
[10] K. Helmsauer, A. Del Moro, T. Göttfert, C. Schefels, L. Schlag, AI-based Novelty Detection in Space Operations: Three Years of Operational Experience and Progression at GSOC, ID #217, 18th Conference on Space Operations (SpaceOps 2025), Montreal, Canada, 2025, 26 - 31 May.
[11] G. De Canio, J. Eggleston, J. Fauste, A. M. Palowski, M, Spada, Development of an actionable AI roadmap for automating mission operations, 17th International Conference on Space Operations (SpaceOps 2023), Dubai, United Arab Emirates, 2023, 6 - 10 March.

[12] N. Policella, J. Martinez, N. Salor Moral, S. Kotisis, K. Helmsauer, T. Göttfert, M. Cristoforetti, A. Gobbi, F. Hegwein, G. De Canio, Innovating Space Operations with AI: The AISHGO Project, ID #49, 18th Conference on Space Operations (SpaceOps 2025), Montreal, Canada, 2025, 26 - 31 May.

[13] J. Silva, A. Donati, WebMUST Evolution, 14th Conference on Space Operations (SpaceOps 2016), Daejeon, Korea, 2016, 16 - 20 May.