17th International Conference on Space Operations, Dubai, United Arab Emirates, 6 - 10 March 2023.

SpaceOps-2023, ID # 679

Data Systems and Infrastructure of the ESA Space Safety programme – latest developments, overview, and outlook

Dominik Marszk[a*], Johannes Klug[a], Gianpiero Di Girolamo[a], Elmar Brendel[b], Kamill Panitzek[b], Chakshu Baweja[c]

[a] *European Space Operations Centre (ESOC), European Space Agency (ESA), Darmstadt, Germany,*
email: firstname.lastname@esa.int
[b] *CGI Deutschland B.V. & Co. KG, Darmstadt, Germany*
email: firstname.lastname@cgi.com
[c] *RHEA Systems GmbH, Darmstadt, Germany*
email: c.baweja@rheagroup.com
* Corresponding Author

Abstract

The Space Safety Programme (S2P) is one of the most rapidly growing ESA programmes. It began in 2009 and currently encompasses 4 segments, with over 50 active software projects, 4 flying space instruments, and 2 space missions in preparation.

In this paper, we describe a unique view of the Data Systems and Infrastructure teams on the ESA S2P. We discuss the paradigm-shifts and industrial best practices introduced to answer the need to develop, manage, deploy, and operate tens of heterogeneous projects. These consist of multiple systems built as hybrids of highly mission-specialised components on one side, and common internet technologies on the another. In the context of the most recent developments like rapidly increasing number of scientific instruments or steeply growing criticality of our systems, we outline how our teams are constantly evolving their approach and mindset, applying lessons learnt in an agile manner.

The paper first outlines the ESA S2P in terms of its overall structure, needs, and interfaces between its stakeholders, then moves on to detail the data centre infrastructure in terms of its design, environments, networking, and underlying hardware stacks. The last major point details the software engineering solutions and practices employed by our team, with a particular focus on unified software project requirements baseline, enforcing a strict branching and versioning model, and a complete containerisation of the environments. Finally, the paper discusses the most important lessons learnt and presents short- and long-term plans of the key evolution points on our roadmap.

Keywords: S2P, Data Systems, Cloud, Infrastructure, DevOps

Acronyms/Abbreviations

CI/CD – Continuous Integration and Deployment
DS – Data Systems
DC – Data Centre
ESEC – European Space Security and Education Centre
ESOC – European Space Operations Centre
ESRIN – European Space Research Institute / ESA Centre for Earth Observations
NEO – Near Earth Objects
NEOCC – Near Earth Objects Coordination Centre
PD – Planetary Defence
S2P – ESA Space Safety Programme
SD – Space Debris
SWE – Space Weather

1. Background

The European Space Agency (ESA) Space Safety Programme (S2P) is a comprehensive program aimed at ensuring the safety of European spacecraft, astronauts, and ground infrastructure. It launched in 2009, establishing a core organisational structure over the years.

The mandate to manage, coordinate, and support the Programme's activities has been given to the ESA European Space Operations Centre (ESOC) in Darmstadt, Germany. However, the number and geographical distribution of its stakeholders drives a very heterogeneous and complex system. Most notably so when it comes to the location of the data centres, variety of software and hardware vendors, ownership of the system elements, number of space and ground assets, areas of expertise held by the contributors, and finally the level of support required and requested by the projects.

The main driving factors behind such a structure are both technical and economical. Most notably, sensor networks work most efficiently only if they are distributed spatially across the entire globe, as well as into space. Furthermore, projects as big and complex require a lot of funding, which comes from the entirety of the Agency's 22 Member States, 4 Associate Member States, 3 Cooperating States, Canada, and the European Commission. Finally, the wide range of expertise needed to implement particular parts of the Programme is often grouped in small focus groups around Europe.

1.1. Notable programme projects and periods

The S2P development and operations planning happens in incremental phases called periods, which are tied to ESA Ministerial Council meetings happening every 2-3 years on which each contributing state makes a financial commitment to selected parts of the ESA programmes.

The programme has so gone through 3 preparatory periods P1-P3 over the first 10 years, which consisted of procurement and development of hundreds of research, preparatory, and preoperational projects. Since 2019, S2P became one of the fundamental ESA activities and entered operational phase, with the first such period marked "S1". S2P is currently entering Period 2, subsequently marked "S2". The main mandate of these new phases is evolving and maintaining previously built systems into reliable and operational portfolio. Furthermore, only in recent years the programme has secured its first space missions – Hera, driven by the Planetary Defence, and Vigil, driven by the Space Weather, and ADRIOS, driven by the Clean Space & Space Debris[1].

1.2. Detailed programme structure

The programme structure break-down has been outlined in the Fig. 1. Elements not directly relevant to technical interfaces and the implementation of the systems have been mostly hidden from it. Focus was put on the elements supported by our teams. Most importantly - Space Weather, Planetary Defence and Space Debris.

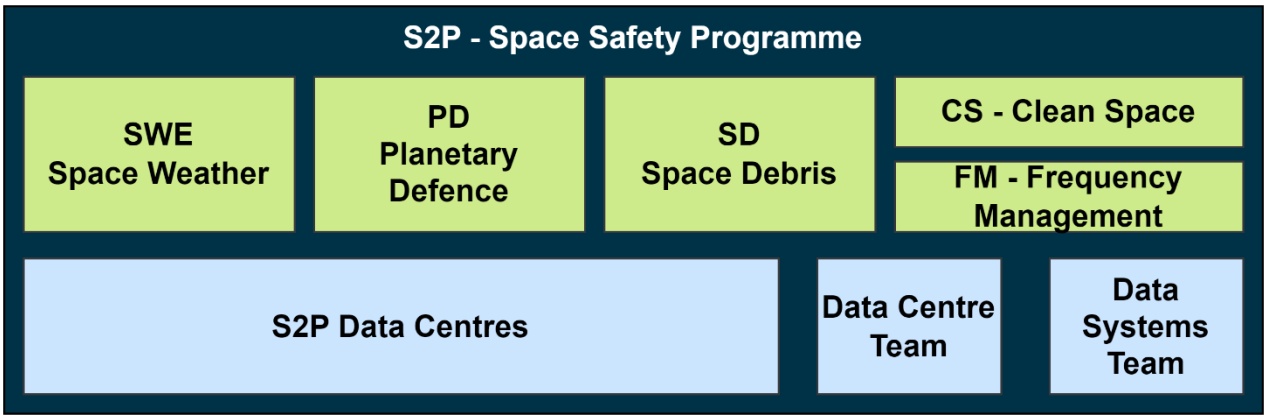

Fig. 1 - S2P high level structure break-down

1.2.1. Core Segment – Space Weather

The goal of the Space Weather (SWE) segment is to understand and mitigate the effects of space weather on human and robotic missions in space. This includes monitoring and predicting the space weather conditions and developing technologies and procedures to protect spacecraft and astronauts from the potentially harmful effects of solar flares, magnetic storms, and other space weather events. To achieve this, ESA works closely with national and international organizations to coordinate space weather research, develop predictive models, and provide operational space weather services. This segment is crucial for the safety of long-duration missions and the sustainability of space activities. The ESA SWE team ensures the accuracy of data by utilizing various simulation models. They collaborate with institutes and industrial partners to develop and validate these models, and then provide them to the scientific community through

ESA federated tools managed by the Data Systems and Data Centre teams. The extensive data sets generated by SWE products are frequently utilized by other S2P segments and the wider ESA community. SWE also maintains and verifies atmospheric models used by orbit propagators in flight dynamics and space debris teams for atmospheric re-entry.

Space Weather segment co-funds the Space Weather Coordination Centre (SSCC) which is tasked with operating and coordinating the Space Weather Service Network (SWESNET) – a federation of Expert Service Centres and Expert Groups which provide an ever-growing number of products and tools [2]. ESA significantly contributes to the federation by subsidising its sub-components, hosting core infrastructure like the Space Weather Portal and the Single Sign-On service, as well as providing products and analysis tools, like the Ionosphere Monitoring Facility (IONMON), the Ionospheric Scintillation Monitoring (ISM), and the Data Browsing and Analysis Tool (SWE Data).

1.2.2. Core Segment – Space Debris

The objective of the Space Debris (SD) segment is to address the increasing threat posed by the proliferation of debris in Earth orbit. Space debris consists of man-made objects such as spent rocket stages, defunct satellites, and fragments from collisions or explosions. These objects can pose a significant hazard to operational spacecraft, potentially damaging or destroying them, and contributing to the growing amount of space debris in orbit.

To mitigate this threat, the Space Debris segment is focused on improving our understanding of the environment in orbit, characterizing the amount and types of debris, and developing technologies and techniques for tracking and avoiding debris. The Space Debris segment is essential for the long-term viability of space activities and the safety of human and robotic missions. Furthermore, threats of re-entering objects are analysed and the risk on ground is assessed. A major contribution is also the development of guidelines and strategies to minimize the future creation of space debris. Operators of functional and newly launched spacecraft are required to comply to this rule set to ensure the access and usability of the earth's orbital environment in the years to come. These rules include methods for the passivation of satellites at their end of life, the definition of "graveyard" orbits as well as active and passive de-orbiting.

A further important activity of the Space Debris Office is the development and international contribution to a Space Traffic Management framework. The goal here is to create unified and accepted principles for the coordination between operators of spacecrafts. It shall allow the exchange of information about orbital data and the conjunction risks between the individual objects. In the more and more probable event of a conjunction, the actions and manoeuvres to be executed need to be defined for each involved party. Also, methods for the automation of collision avoidance are under research due to the predicted dramatic increate of new satellite - especially due to mega-constellations.

1.2.3. Core Segment – Planetary Defence

The Planetary Defence (PD) segment is dedicated to protecting our planet from the impact of near-earth objects (NEOs), such as asteroids and comets. The goal of this segment is to detect, track, and characterize potentially hazardous NEOs, and to develop the capability to mitigate the impact threat if required [3].

To achieve this, the Planetary Defence segment involves multiple activities such as the development of ground-based and space-based observational systems to detect and track NEOs, the characterization of their physical and orbital properties, and the assessment of the impact risk they pose to our planet. A major contribution is the development of the highly specialized Flyeye telescopes with their 16 cameras each. They are designed to perform survey observations and will significantly improve the detection of new asteroids and increase the prewarning time for earth impactors.

The program also supports the development of technologies and techniques for deflection and disruption of hazardous objects, such as the use of kinetic impactors or gravity tractors, as well as the investigation of the physical and chemical properties of NEOs, which will be essential for any future mitigation efforts.

Planetary Defence funds the Near Earth Object Coordination Centre (NEOCC) which is tasked with overseeing the operational part of the segment, like the NEO Portal, observation processing pipelines, as well as a variety of tools like the NEO Risk List, the Orbit Determination and Impact Monitoring Software (Aegis), and the NEO Population Generator (NEOPOP).

1.2.4. Core Segment – Frequency Management

The Frequency Management (FM) segment of the ESA Space Safety Programme is concerned with the allocation and use of radio frequencies for space activities. The goal of this segment is to ensure the safe and efficient use of the radio frequency spectrum for both civil and military applications in space.

To achieve this, the Frequency Management segment involves activities such as the coordination and management of frequency assignments for space missions, the provision of technical support and advice to the space industry, and the participation in international standardization efforts.

1.2.5. Core Segment – Clean Space

The Clean Space segment of the ESA Space Safety Programme is focused on ensuring the responsible and sustainable use of outer space. The goal of this segment is to minimize the environmental impact of space activities and to promote the use of environmentally friendly technologies and practices in the space industry.

This segment is spear-heading both technical and political efforts to ensure future missions are built and disposed with minimal footprint on both space and earth environments. This involves concepts like Design for Demise, which is targeted to maximise the chance of a complete spacecraft disintegration upon re-entry as well as promoting sustainable designs that avoid use of rare or toxic materials for building and propelling spacecrafts.

The flagship project of this segment is Active Debris Removal and In Orbit Servicing (ADRIOS).

1.2.6. Core Element – Data Systems Team

The Data Systems (DS) Team is responsible for procurement, development, deployment, and subsequent operations of the software systems that each of the segments is built upon. The team offers expert consultancy to each software project built within the segment, ensuring that it is produced adhering to established best practices and processes, and that the resulting product can be easily integrated into our environment to ultimately release it into production. The DS team is the primary driver and the maintenance team for the data acquisition, incorporation, and processing chains from the core segments [4].

1.2.7. Core Element – Data Centre Team

The Data Centre (DC) Team is responsible for the underlying infrastructure the software data systems built on top of it. The DC Team's core competency involves development, deployment, maintenance and operations of the data centres, field equipment, networking, and virtualisation infrastructure, as well as operations of some production and monitoring software components.

The DC and DS Teams are jointly responsible for a long-term system-level vision and implementation of the entire programme infrastructure and software.

1.2.8. Core Element – Data Centres

S2P Data Centres (DCs) are managed by our team and embedded within the data centres of different sites of the Agency. There are 3 major S2P centres in 3 locations – ESOC, ESEC and ESRIN. These make up a private cloud environment, managed by the DC and used by the DS teams.

The global network of data centres and networking infrastructure also provides a backend for a number of field equipment. One such example is the backend of NEO telescope stations. The reliability and robustness of this infrastructure is topmost priority to provide uninterrupted access to several projects and services. Regular testing, updating of systems and maintenance is performed by our team.

2. Applied solutions

The data systems of S2P are very heterogeneous [5]. These differences are based in a multitude of reasons, some of them being:

- Maturity and timescale – some of the software has been developed as early as in the 90s and are in maintenance for decades, while others are still in design phase. In addition, some of our systems are designed, developed, and deployed within weeks,
- Development teams – rarely different systems are built by the same people, some might have been developed internally, and others by vastly different companies,
- Technology stacks – some projects can be almost entirely written in Fortran, while others can feature Python or C++17,
- Hardware stacks – many of our systems are entirely software-centred, but some involve operating real hardware like a telescope, or processing data and controlling a real instrument on ground or in space,
- Deployment location – plenty of systems are developed by 3rd party and deployed there completely independently, just needing sporadic consultancy, others might need support when integrating into our federated API, while some are deployed and operated entirely by us. Deployments within the ESA DC are always approved by DS and DC teams, to maintain harmony of the systems and mitigate any spontaneous user errors.

Moreover, we currently have over 200 repositories in total, with that number growing every month. The number of repositories by segment is shown in Table 1.

Table 1. S2P repositories supported by the Data Systems & Infrastructure teams per segment (February 2023)

Segment	Repository Count
PD	104
SD	52
SWE	39
DS/DC	10
Total	**205**

In the following sections we outlined a few key qualities, processes, and technologies we identified as critical to ensure a level of success and quality to our projects.

2.1. Harmonisation

Technical coordination of tens of projects as diverse as described above, with some in maintenance and operations, and others still in development when having a team of about 10 people requires a harmonised approach. Below are some high-level rules we learned to apply to every development, ongoing or new. All the recommendations discussed below are described in the S2P Data Systems Development Guidelines document we distribute and apply to all the projects. Thanks to that, the interfaces between each team member and their currently supported projects are quite similar, thus allowing for a quick allocation and reallocation of effort wherever and whenever it is needed.

2.2. Agile, CI/CD and DevOps

Despite S2P and ESA processes required to be ECSS-compliant, we try to keep our software production process lightweight, by increasing the number and frequency of iterations, reducing the number of reviews, and making the process software-centric instead of documentation-centric. Many of these practices and optimisations are described in the ECSS Agile software development handbook [6].

Furthermore, we absolutely enforce the use of Gitlab CI/CD, providing to all our projects the ESA in-house Gitlab infrastructure, which ensures that not only anyone developing or reviewing the project has immediate feedback on its health, thanks to the continuous integration, but also anyone responsible for operations can easily deploy the project thanks to the continuous delivery. Not only that – by having well defined means of building and deploying the software, it is ensured that this is not a knowledge that has to be passed on, or is hidden within the documentation.

Finally, all of the above enables us to build and foster the DevOps culture, which ensures that not only the loop is tightly closed between development, operations, and user teams, but also elaborates the needs of every individual team, allowing to quickly converge on optimal solutions [7, 8]. Our most mature DevOps workflow is built around the Planetary Defence operations, where the NEOCC team takes an active part in all of the software processes.

2.3. Environment split

Each of our products is always deployed into at least 2 environments. Usually these include:

- Integration – IRE – where new patches from various teams can be easily tested and integrated internally, without impacting any of the production versions,
- Pre-Operational – Pre-OPE – i.e., staging, where we stage the releases and can make them available to our partners before its promotion to production,
- Operational – OPE – i.e., production, providing products that are accessible publicly.

Each of the environments can be easily duplicated whenever needed. Typically projects also include a Development environment configuration, which is most rich in tools for testing, simulation, and debugging and can be spawned locally by the development teams [9].

2.4. Technologies

All of these solutions are heavily dependent on a set of core technologies that our team rely on daily. Below is a non-exhaustive list of all kinds of technologies, stacks, and standards:

- Virtualisation – all our data centres rely on VMware,
- Containerisation – all our applications are now containerised using Docker,
- Orchestration – At the VM and infrastructure level, the systems are orchestrated with an Ansible AWX, applications are orchestrated using either Docker Compose or Docker Swarm,
- Code versioning and CI/CD – we rely virtually exclusively on git, Gitlab, and Gitlab CI/CD,

- Artifactory – our build artifacts and produced containers are stored either within Gitlab CI/CD or a Sonatype Nexus instance,
- Single Sign-On (SSO) – we have several SSO deployments across different domains, relying primarily on OpenAM, Keycloak, and Oauth 2.0,
- Tracking – most of our projects are relying on JIRA, with a few using Gitlab or other issue tracking systems,
- Monitoring – our system and application monitoring tools are focused around feeding a central, sharded ELK stack,
- Programming languages – Bash, C, C++, Fortran, Java, Python are all very widely used in our repositories,

2.5. Lessons learnt

Some of our past system-level developments were aimed to create very scalable, enterprise-grade solutions where the size of our user base, and the funding alike did not fully justify the complexity and the maintenance cost coming with that scalability as well as the technology and architecture chosen. This resulted in an impeded functionality, difficult maintenance, and thus a low buy-in, poor adoption, which in turn perpetuated a growing technical debt across our software stack. In future designs great care must be taken when trading off growing system-level complexity for less harmonisation and less scalability.

Many examples of a "maximum commitment" to the microservices failing are available across the internet [10], indicating that there is a balance to be found between maximum sharding the application codebase and design, and keeping some of the system parts monolithic. We expect this balance to shift over time, prompting larger harmonisation activities overarching previously heterogenous, organic components. The main drivers for this shift anticipated by us are: maintenance teams growing in size and independency, and the need to scale parts of the system selectively.

3. Recent developments

Out of the applied solutions outlined in the previous chapter, many has evolved organically for years, whilst others are fairly new and introduced ground-breaking changes.

Starting in late 2019, a number of core internal services were introduced by the infrastructure team to get ahead of the rapidly growing programme. Notably, we started using a Sonatype Nexus instance, an internal central credential management LDAP, and an Ansible AWX. The Nexus was critical to migrate a more complex projects into CI/CD pipelines, while LDAP and AWX ease management of an ever-expanding list of assets.

Around 2020, migration of every existing project in maintenance onto Gitlab CI/CD was concluded. Previously the build systems were anything but harmonised, and deployments were mostly done manually. This caused a very steep learning curve to get the software running from scratch. Not only that – a high cost of deployment from scratch makes the teams especially risk averse when it comes to potentially breaking changes in the system, which hinders agile and quick paced evolution.

Since 2022, using the programme rename from SSA to S2P as an opportunity to upgrade and prepare for scaling, our team took the responsibility for the entire DNS zone suffix *s2p.esa.int* [11]. This allowed us to start centrally managing domains and their respective SSL certificates, which in turns enables quicker provision of new services both internally and to the public. We have also prototyped several application technologies like Apache Airflow that would facilitate evolution of our SWE and PD systems towards an integrated Data Hub and pipelines processing.

4. State of art, outlook, and future evolution

In this chapter we describe the challenges and evolution we expect to be facing in the following years. As the programme growth is guaranteed, we must plan and prepare for the increased volume and size of the projects. This will naturally impact a quantitative resources planning, but has to be followed with a qualitative capabilities planning, evolving our infrastructure, processes, and the offering.

4.1. DevSecOps

While application of DevOps culture and processes bring clear benefits already, the road does not end there. We want to evolve towards a DevSecOps approach, recently exploding in popularity due to the steeply growing concern and awareness of security in digital systems. There are certain low-hanging fruits and parts we can introduce incrementally, while others have to come in steps. For instance:

- Each of the containers should be regularly scanned for security vulnerabilities – this could be performed as a 3rd party tool connecting to the container repository and running the scans in the background, notifying us in case of problems found, which is a very transparent and incremental change,

- All of the software pipelines should run fairly a new, patched software stack – modification of some older build pipelines can be not a trivial task and has to be carefully managed and prioritised, according to the software age, attack surface and sensitivity of the stored, accessed, and processed data.

4.2. Hybrid cloud evolution and cloud native solutions

Current software deployments always rely on a set of containers, expected to produce a self-contained, well-defined application. Many modern applications are built around cloud-native technologies, by directly utilising an API specific to data storage, processing, and transformation capabilities of a cloud vendor API. This allows the applications to very easily auto-scale. Some of our computational and storage needs could utilise these abilities when processing large batches of input data, special requests, or just needing a specialised computational hardware (e.g. GPUs). ESA in-house cloud is now growing new features of hybridization with Azure and AWS to address this very use case [12].

Furthermore, not every software copes well when containerised on a small scale. For instance, it is not recommended to containerise database environments like PostgreSQL for high-availability and redundancy in production environments. Here again commercial cloud solutions like Database-as-a-Service (DbaaS) or self-hosted multi-tenant clusters could provide us with higher quality backend for less costs.

We are also recurrently evaluating at Kubernetes (K8s), which is often understood as the pinnacle of capabilities when it comes to deployment orchestration, scalability, and management. So far the complexity of maintaining a self-hosted K8s cluster deemed to outweigh the benefits, mostly due to the small scale of our application stacks. This is likely to change in the future, driven by the increased need for heavily scalable and redundant workloads, as well as anticipated growing availability of 3rd party cloud providers like Azure or AWS, where we could off-load some of our hosting needs.

4.3. Bolstering redundancy and high availability

Most of our current production environments typically do not span across multiple sites. Alongside with the above cloud solutions, we could strengthen our deployments availability by utilising at least two independent datacentre locations for each of our production environments. Our preliminary tests of configuring multi-shard Docker Swarm clusters across the sites together with a Round-Robin DNS are quite promising. Here, use of 3rd party cloud solutions like Content Delivery Network (CDN), DNS, and Load Balancers could once again bridge the gap between a self-hosted cloud, and high availability ensured by the global providers.

4.4. Data Hub

As most of our software relies on storing and processing the data, we envision being able to serve that software with a set of common APIs, all pointing at a common data exchange, instantiated per S2P segment. Such an approach would allow to centralise and deduplicate tens of data stores deployed across our infrastructure, letting us put all focus on keeping up a single, highly reliable, homogeneous data store and exchange. This further adds to the usability of the data, providing the users with parallel data accessibility from different sources, which they can use to compare and develop complex scientific models [13].

Such an approach brings obvious benefits of preventing an explosion of interfaces all across the system, whenever data has to be exchanged between two components. However, it has to be carefully engineered as to not introduce extra bottlenecks or unacceptable limitations onto the applications.

4.5. Extract, Transform, Load (ETL) pipelines

ETL pipelines are a basis of aggregating, cleaning, and subsequently feeding data into a Data Warehouse, or a Data Hub. Example of a basic ETL involves Payload Ground Processor, which aggregates L0 data, filters and transforms it, subsequently producing L1 data sets and products which could be loaded onto a Data Hub. Further processing could be done either manually by the operators, to create handpicked and tailored data sets available to other users. A proper ETL framework would be built on top of Data Hub APIs and provide the users with the building blocks necessary for prototyping and then automating new pipelines, without having to worry about the complexity or the load generated.

4.6. "One stop" Data Centre offering

All the aforementioned solutions are meant to ultimately add up to a portfolio of services that we could offer to all of our stakeholders and external users either as cloud-hosted SaaS or self-hosted high quality software images. Note that this constitutes a paradigm shift, where the teams are meant to walk away from supporting a heterogeneous set of unlinked applications, but rather focus on a core set of consistent services, developed, and tailored for each of the main user groups (i.e., core segments), on top of which meaningful applications can be developed. There is still quite a technology and development gap that must be bridged over the coming years before this vision can be fulfilled.

An example envisioned SWE Data Centre infrastructure, with focus on Payload Processing and data federation with SWE Expert Centres Network is shown on Fig. 2.

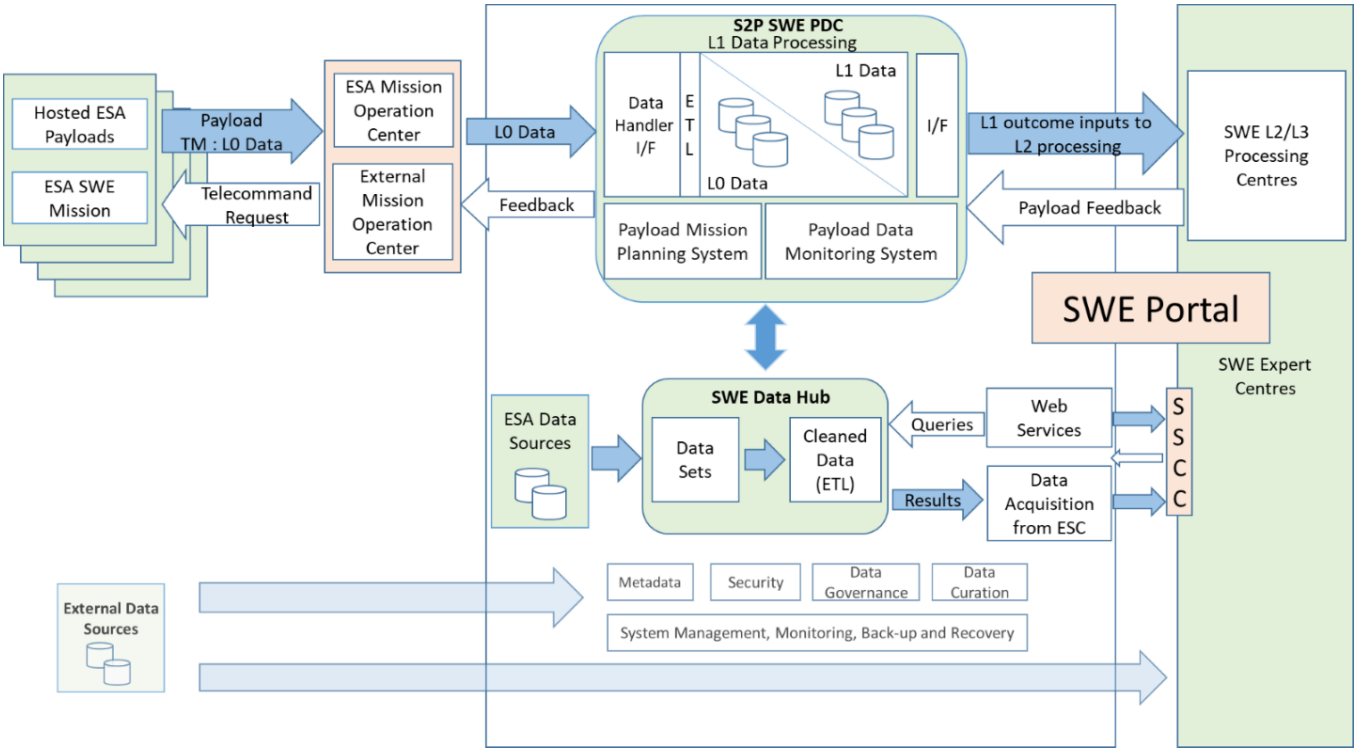

Fig. 2 Future Space Weather Data Centre diagram

5. Conclusions and Discussion

The S2P Data Centre & Systems is the most rapidly growing software and infrastructure segment across the entire Agency, with its expansion rate and needs akin to a prosperous start-up. This has pushed us towards the adoption of an increasing number of successful practices and solutions known from commercial sectors.

With the maturing technology which can support service-oriented data processing architectures, alongside a quickly growing user base and application portfolio, we believe it is time to take a refocused approach on a top-down system design, starting at the level of each of the S2P segment, eventually looking into harmonising common services between different pillars.

The S2P programme is largely data-oriented, and thus ultimately driven by software systems. We believe it should keep adopting modern practices involved in creating, maintaining, and operating the software. The fact that our programme is still young and many of the solutions are developed from ground up can be used to further concentrate on the above goal, allowing to quickly introduce innovative and novel approaches that would normally take decades to adopt in an industry that is as conservative as ours. We hope to keep using mature agile system engineering practices and tools that allow for continuous improvement [14]. That will require to resist completely giving in to the prevalent mindset of “If it ain’t broke, don’t fix it” that, when applied literally, ends up hindering the innovation and favours growing a technical debt.

References

[1] European Space Agency, “Space Safety, The story so far,” 2019. [Online]. Available: https://www.esa.int/Space_Safety/The_story_so_far.

[2] European Space Agency, Space Weather System Requirements Document, SSA-SWE-RS-RD-0001 (09-07-2013)

[3] European Space Agency, PDC 2021 – ESA’s Planetary Defence NEO Coordination Centre DevOps model based Operations

[4] ESA-SSA-DS-SP-0001. (2018). SSA Data System Development Guidelines and Specifications

[5] J. K. A. L. R. S. M. Gianpiero Di Girolamo, "SSA Data Centers and Data Systems: Evolution towards a hybrid architecture," in SpaceOps Conferences, Marseille, France, 2018.
[6] ECSS, E-HB-40-01A - Agile software development handbook, 2020.
[7] J. Humble, and D. Farley, Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment
[8] L. Chen, ―Continuous Delivery: Huge Benefits, but Challenges Too, IEEE Software, vol. 32, no. 2, pp. 50-54, 2015
[9] The Centre for Research on Engineering Software Technologies, Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices
[10] Twilio Segment, "Goodbye Microservices," 2018. [Online]. Available: https://segment.com/blog/goodbye-microservices/. [Accessed 2023].
[11] European Space Agency, Services for Spacecraft Operations support within the ESA Space Situational Awareness Space Weather Service Network
[12] European Space Agency, Ground Systems Engineering (GSE), Evolution towards hybrid system architectures
[13] European Space Agency, Data Management (DM), Heterogeneous data source management in esa space safety programme
[14] E. Laukkanen, J. Itkonen, and C. Lassenius, ―Problems, causes and solutions when adopting continuous delivery—A systematic literature review,‖ Information and Software Technology, vol. 82, pp. 55-79, 2017