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

SpaceOps-2025, ID # 340

Don't Fear the DevOps – We Already do it

Tristan Edwards[a], Alessandra Nanni[b], Daniel Lee[c]

[a] *Infrastructure Operations Team Leader, EUMETSAT, Germany, tristan.edwards@eumetsat.int*
[b] *Ground Segment and Solution Coordinator, EUMETSAT, Germany, alessandra.nanni@eumetsat.int*
[c] *Data Access Systems Operations Manager, EUMETSAT, Germany, daniel.lee@eumetsat.int*

Abstract

The System Engineering V-model is common in the space industry. It works well for activities with large deliverables that require perfection in stable final configurations. It is less suited for long-running systems, such as EUMETSAT's multi-mission facilities, whose lifecycles require continual evolution without significant downtimes. These facilities are some of our most critical assets, as they serve all missions. It must be possible to adapt their configuration daily (e.g. add data flows, monitor new missions, remove outdated items) and restore it to previous states. This makes them unsuited for cumbersome and slow approaches to change. In theory, DevOps provides the best of both worlds: Speed and agility while limiting risk.

EUMETSAT operations uses DevOps approaches for multi-mission elements, made possible by our team structure. Working in small batches, using fast feedback and monitoring, and promoting a generative culture allow teams responsible for EUMETSAT's monitoring and data access systems to harness the benefits of DevOps – completing work quickly to maximise total value over time, while minimising risk produced by changes. Operations and maintenance teams identify, document, implement, and deploy changes in multi-mission systems, whose lifetimes extend beyond those of individual missions. While adopting these methods instinctively, we recognised the need for a structured approach to adopting DevOps.

In two pathfinders we identified what to preserve from our legacy practices, what to eliminate, what to accept because of legacy constraints, and what to create when developing new ground segments and promoting a cultural shift through awareness and training. This paper explores what DevOps means in these pilots, and how that aligns with EUMETSAT system engineering.

Frequent changes increase traceability and lower individual risk but produce overhead. They can also reduce overall system maturity. This is managed using representative environments and automated testing. This shifts the focus of deployment engineers from manual work to pipeline development and maintenance. In the case of the multi-mission data dissemination system, this means automated deployments to representative validation and operational environments can occur daily, with human approval but without manual work, using GitLab, Nexus, and Docker.

In addition to the pilots, work on an overarching approach is ongoing in the development department to bridge silos using provisioning platforms, tools, format standardization and process simplifications. This could be reused by operations, providing end-to-end DevOps optimization for ground segment systems. When completed, this will be a key accelerator in the deployment and maintenance of ground segments for new missions – and across the organisation.

In this paper, we show how DevOps balances time-critical needs without compromising system safety. In addition to DevOps cultural practices, in EUMETSAT operations continuous integration and deployment technologies are critical technological enablers. They give teams a single platform to make and review changes traceably, and package and deploy software and configuration themselves. They make changes reproducible so that rollbacks are straightforward and create space for additional augmentation of our deployment pipeline, e.g. using automated tests. We explain how we marry this approach with established change control procedures to ensure compliance, safety, and speed simultaneously.

Keywords: EUMETSAT, system engineering, DevOps, CI/CD, multi-mission operations.

1. Acronyms/Abbreviations

CAM	Competence Area Manager
CDF	Common DevSecOps Framework
CI/CD	Continuous Integration/Continuous Delivery
DaaS	Data as a Service
DWG	DevSecOps Working Group
EUMETSAT	European Organisation for the Exploitation of Meteorological Satellites
EO	Earth Observations
FAMB	Functional Area Management Board
GSB	Ground System Board
GSCG	Ground System Coordi
IaaS	Infrastructure as a Service
IaC	Infrastructure as a Code
ICSI	Information Centric System Infrastructure
IJPS	Initial Joint Polar System
MCC	Mission Control Centre
MTEP	Mid-Term Evolution Plan
PaaS	Platform as a Service
UVN	Ultra-violet/Visible/Near-Infrared instrument

1. Introduction to EUMETSAT

EUMETSAT is an intergovernmental organisation based in Darmstadt (Germany), responsible for the exploitation of Europe's meteorological satellites. EUMETSAT operates a system of meteorological satellites that observe the atmosphere, ocean and land surfaces – 24 hours a day, 365 days a year. This data is then supplied to the National Meteorological Services of the organisation's Member and Cooperating States in Europe, as well as other users worldwide.

The satellites currently operated solely by EUMETSAT include:

- Geostationary satellite Meteosat-12 (Meteosat Third Generation) launched in December 2022. It provides imaging services supporting nowcasting applications.
- Geostationary satellites Meteosat -10, and -11 over Europe and Africa, Meteosat-9 over the Indian Ocean. This corresponds to the Meteosat Second Generation.
- Metop polar-orbiting satellites (Metop-B and Metop-C) as part of the Initial Joint Polar System (IJPS) shared with the US National Oceanic and Atmospheric Administration (NOAA).
- Micheal Freilich Sentinel-6 satellite providing global sea surface height observations for climate monitoring and ocean and seasonal forecasts. Jason-3 is also part of the Jason mission (an international partnership between EUMETSAT, CNES, NOAA, NASA and the European Union via the Copernicus programme), even though it is not operated by EUMETSAT.
- Sentinel-3 satellites (S3A and S3B) collecting observations of global ocean colour, sea surface temperature and sea surface height.

One of the main objectives of EUMETSAT is also to create synergies with other operators of Earth observation satellites. Currently, EUMETSAT cooperates with other agencies including in Europe, China, India, Japan, South Korea and the United States, benefiting from the sharing of data from many other satellites.

The data and products from both EUMETSAT and third-party organisations are vital to weather forecasting and make a significant contribution to the monitoring of the environment and climate change. They aid meteorologists in identifying and monitoring the climate change or the development of potentially dangerous weather situations that affect air travel, shipping, road traffic, farming, constructions and many other critical industries.

EUMETSAT's planned missions and launches will continue to grow during the next 20 years. This increasing number of satellites and, consequently, amount of data that needs to be handled generate a big impact on the structure of the organisation. The addition of missions requires not only scaling many of the existing systems, but also redesigning those that become inefficient or simply unable of handling the new load. 2025 is in particular a busy year for EUMETSAT, with the planned launches of Meteosat Third Generation Sounding Mission, Sentinel-6B and Metop Second Generation.

The EUMETSAT Mission Control Centre (MCC) is based at its headquarters in Darmstadt (Germany) and is divided into two control rooms, one for the geostationary (GEO) missions and the other for the Low Earth Orbit (LEO) missions. Shift teams of satellite and ground segment controllers work 24 hours a day, supported by teams of on-call operators and maintenance engineers.

The MCC is the part of the overall ground segment responsible for the safe operation of all satellites. It provides monitoring and control functions for the spacecraft and antennas, but also monitoring of science data (L0, L1, etc.) and supporting infrastructure. It also provides reporting functions in order to notify the user community in case of expected or unexpected events, display the status of each mission in a real-near time manner, or generate KPI reports.

2. DevOps in the EUMETSAT Context

A main focus of this paper is the EUMETSAT team structure and how this can fit a DevOps style of working, but first it is important to define what we define by DevOps in this context.

Study of [1] thru [4] broadly describes DevOps as a cultural understanding in a business or organization to remove the siloes of software development and operations and integrate them into a more singular function where both teams are involved in the entire application lifecycle. This may also include quality assurance and security. The nature of DevOps is not one that is necessarily black and white - some people can take it to mean implicitly the use of fast Continuous Integration / Continuous Delivery (CI/CD), and there may even be some cynicism that this can invariably reduce the amount of quality gates and increases risk. Indeed, this is a struggle found at EUMETSAT too, where concepts of maintaining a strong baseline with only bi-annual releases compete with more modern continuous development.

It helps in this case to understand the maintained system in more granularity. One may assume the core mission control system software component is a relatively stable entity, which requires detailed review and explicit authorization to change. It is natural in this instance to want to apply a monolithic approach to change management here. On the other hand, a large number of changes that EUMETSAT makes to operational systems have to be fast by nature of our business. As a primary hub for sharing weather and climate data worldwide, changes to our dissemination (and monitoring thereof) of operational products and services is a near daily business, and we often have to react fast to problems and changes by third parties. In this case, a bi-annual release clearly would not work, and we rely on a leaner system to apply change while ensuring quality control and configuration management is maintained. In the latter case we are evolving as an organisation towards the use of CI/CD to fulfil the need.

Even if EUMETSAT does not currently follow a tight "as-designed" DevOps structure where development and operations teams are a singular entity, the management structure and matrix working style does facilitate this style of working. The organisation of EUMESTAT for the context of this discussion is structured as follows:

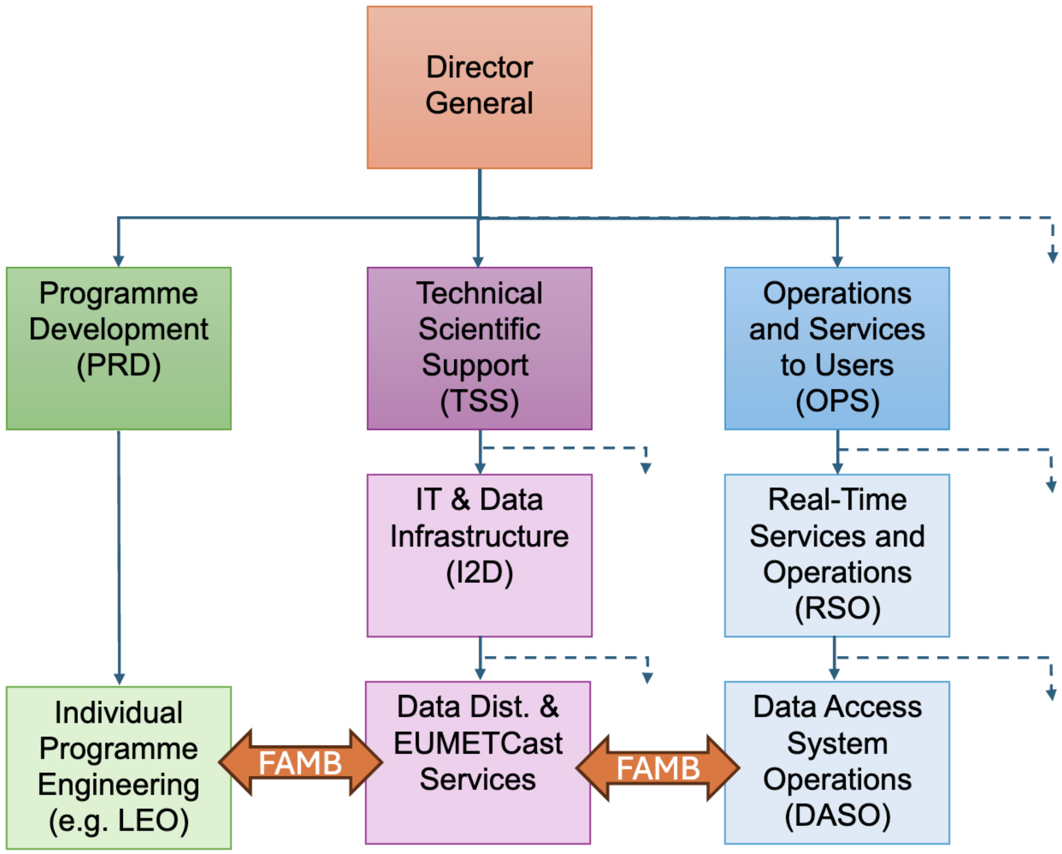

Figure 1: Organisation Composition of EUMETSAT Technical Divisions and Example of Matrix

In the organisational structure, three core departments – Programme Development (PRD), Technical Support (TSS) and Operations (OPS) provide the technical competencies for respectively developing, implementing and operating satellite and ground segment systems. While it could be considered that OPS and PRD are consumers of technical competences provided by TSS, the relationship between them is considered more as stakeholders in the system.

The structure is best described with an example of one of our operational competence areas – that shown in Fig. 2. as Data Access. Broadly speaking, these are systems and facilities that provide EUMETSAT's users with near real time and offline data. The two teams in OPS and TSS have their distinct competencies and roles but also act as a singular entity in the frame of the "Functional Area Management Board" (FAMB). The FAMB consists of co-chairs which are the Competence Area Managers (CAM) from the respective TSS and OPS teams.

The FAMB ensures a consistent oversight between implementation and operations teams at all stages of a development lifecycle. This lifecycle begins with a development roadmap presented on an annual basis to a board of management from all departments, which then forms part of a Mid-Term Evolution Plan (MTEP) reviewed by the same body. The roadmap and MTEP is executed by formal projects or by Engineering Change Proposals requiring approval from the FAMB co-chairs. PRD also have a stake in the FAMB process, to ensure that any development of ground segment solutions fits in with future satellite development.

One important aspect of the EUMETSAT strategy is the wide scale deployment of Multi-Mission Systems and Services (MMS), which include our example of Data Access Services. By declaring these services to future programmes, efficiency is found through re-use of systems rather than re-design.

In following a standard software development the cornerstone, the default model used is still the engineering V-model as typically found in large Satellite Operations Agencies. In our example, the DASO team may provide user requirements to the TSS Data Services team who develop the system requirements and design, followed by the implantation either directly or via a contractor. Once verified the TSS team hands over the software to the OPS team for validation. While this would imply the antithesis of a DevOps concept, there is also a heavy emphasis on collaboration (a EUMETSAT core value) between the TSS and OPS teams in forums of the FAMB to ensure a continuous feedback on developed systems.

An additional cornerstone of the EUMETSAT organisational matrix is the concept of Software Unit (SU) vs Data Unit (DU). The SU is the core software for a system or service, and is typically the responsibility of the TSS team to maintain. A standard requirement of any proprietary or third party in use however is that the DU or configuration of the software must be a separate Configurable Item (CI) which has a different maintenance cycle. The key concept here is the ownership of the DU, which is typically by the operations team. With this, they are not only the customers of services but provide the day to day configuration of them. This allows rapid response to changes, such as to our operational products as described above. In this area, the governance and implementation of a development for a simple configuration team (such as an additional product for dissemination) can be done in one team, and here we find opportunity for leading towards a more CI/CD based cycle as described in the following sections.

Overall, it can be considered that we would rank EUMETSAT currently in the "managed" stage of maturity in considering the overall level of adoption within the organisation, with a desire in the coming months to move into the "defined" stage

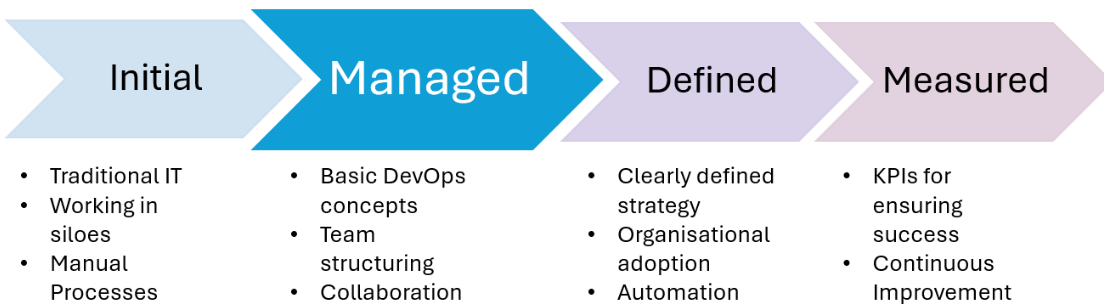

Figure 2: DevOps Maturity Index and EUMETSAT positioning*

3. DevOps Pilots

3.1. The technology platform: Information-Centric Service Infrastructure

EUMETSAT has strategically embarked on the development of an Information-Centric Service Infrastructure (ICSI) as the foundation of its transition toward cloud-based service delivery. This initiative marks a pivotal shift, as it is the first deployment of a private computing cloud within our organization, enabling us to gain essential operational expertise before expanding into cross-agency public cloud services. The decision to invest in an internal private cloud infrastructure aligns with our long-term vision of progressively integrating public cloud capabilities, while ensuring a secure, controlled, and scalable environment for our mission-critical operations.

The ICSI platform is designed to standardize and optimize data access services, serving as a key enabler for EUMETSAT's evolving Big Data Services. This approach empowers users by providing direct access to data, leveraging self-service capabilities, and fostering a highly automated, resilient, and performance-driven environment..

The core objectives of the ICSI private cloud are:

---

* Mohamed, S. I. (2015). "DevOps Shifting Software Engineering Strategy—Value Based Perspective." *International Journal of Computer Engineering*, Vol. 17, No. 2, pp. 179–187.

- *Develop internal expertise*: Establish a robust knowledge base within EUMETSAT before transitioning to a hybrid or multi-cloud strategy that involves external cloud providers.
- *Enable data-driven operations*: Support the delivery of high-volume, high-velocity data services, including online data access, hosted processing, and Earth Observation (EO) data transformations.
- *Promote automation and self-service*: Reduce operational complexity through automated provisioning, orchestration, and containerized environments, ensuring a developer-friendly, scalable infrastructure.
- *Support long-term cloud adoption*: Lay the groundwork for future interoperability with cross-agency and EWC EUMETSAT Weather Cloud initiative, aligning with EUMETSAT's Data Services Roadmap.

To achieve these goals, the ICSI platform integrates modern cloud-native technologies, leveraging Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Data-as-a-Service (DaaS) models. These capabilities are crucial in ensuring that users can access, process, and derive insights from vast datasets efficiently and securely.

By deploying ICSI as an internal capability, EUMETSAT is positioning itself at the forefront of cloud-enabled operations, ensuring that our infrastructure is adaptable and resilient, while providing a scalable foundation for the next generation of space-based data services.

3.2. Leveraging our internal cloud: EUMETSAT's DevOps pilot outside operations

In 2016, EUMETSAT embarked on pathfinding to improve our data services to users. Initially, this involved a number of pathfinders, which then evolved into pilots, and these pilots eventually transitioned into operations. Because we were building on a green field, this gave us a lot more options than we would have normally had in operations, which are built to support spacecraft and work on timelines appropriate for that kind of work in mind.

The pathfinders were built in our on-premises cloud environment, the ICSI. One of them, the Data Tailor, which had originally been conceived as a simple toolbox, had grown in complexity because it was meant to transform a high number of data formats into a high number of potential target formats, with multiple other transformations in the middle. It was our all-purpose data preparation toolbox to make workflows compatible with whatever pipeline users wanted to use on the data downstream.

As we transitioned from Pathfinder into Pilot, we realized that the burden of testing all the use cases that we supported was very high. Faced with the question of investing in our technical debt or spending an ever-higher portion of our time on automatable work, we invested heavily in automated testing. This was done while we were rebuilding the software to be used as a web service rather than a stand-alone software program that got deployed to a user's computer. Leveraging the services available in ICSI—most importantly GitLab and Ansible—we developed a deployment pipeline that applied all tests and on success deployed cleanly to a UAT environment. If the deployment was approved in UAT, it could be rolled out very rapidly to the production environment with the click of a button.

This meant that we could do multiple deployments per day, which allowed us to use a monorepo, monobranch development approach where the developers got fast feedback on what they had built. They could also respond to problems found by QA or Infosec very quickly, whilst the integration effort was kept to a minimum. On top of that, the use of tags in git, which are immutable and protected by GitLab, made it possible to always keep track of the latest approved version, even if it wasn't in UAT at the time. A practice emerged of deploying to production twice a day because it was easy to do and it let users who were experimenting with the pilot service see the updates as soon as they came out. This production service was what was integrated with the other pilot services that were being built in parallel—a web data viewer and a data store.

Contrast this with our normal development pipeline, which would have had a large number of configurable items that are work in progress because deployments took place rarely. Most systems at the time were putting out just a few releases a year. The fastest rate at which a deployment to operations could be made was two weeks from finishing the code—a week advance notice was required to schedule the deployment to our validation environment, and following that another week to deploy to operations. And this only if no problems were found. These often needed to be fixed forward, and then re-integrated with the development branch, which would have moved on quite a way from the one that was deployed to operations. The result was that, because mutable subversion tags were tracked as the deployment assets and deployments were not automated, mistakes were sometimes made, nonetheless. Furthermore, the learning cycles were very loose and oftentimes code bases were out of sync, so that when problems were discovered in operations a lot of rework was required, and a lot of detective work to figure out what was needed. By comparison to this two deployments per day are extremely fast.

This all culminated in our first big demonstration to senior management. At that point, we had built something that we were really proud of. We provided a demo that showed an end-to-end workflow from discovering interesting data in the view service, then finding where the raw data behind the visualization could be found in the data store, before tailoring it into a format that fit on a phone and downloading it—all online. The team was super proud. But one of the executives in the room, while impressed, made a sad face and said, "It took a long time to load."

It did indeed. The reason was the slow parser we were using, which meant that it took about 30 seconds for the settings to load. This had not been a significant issue, however we had already experimented with migrating to a new parser, which would be a drop-in replacement. This had not yet been implemented but was completed later so when the team heard the feedback and pushed it to UAT. Hundreds of tests were executed, plus manual verification, and a few hours later we had it in production, to the delight of the executive.

It is important to point out however that all of this was possible at the time because we were not yet in operations, so we had the team and the technology platform without the constraints of business as usual.

4. Generic DevOps Approach and Optimisation

4.1. Leveraging our internal cloud: EUMETSAT's DevOps pilot outside operations

In operations, implementing pure DevOps practices is more challenging due to the significant manual involvement required for rolling out changes, as outlined in previous sections. Another issue has been how we interpret our processes as an organization. Historically, our stance has been that any changes must go through our entire change management process, introducing weeks of lead time from when code is ready to ship.

To address these challenges, the Real-Time Data Services team implemented a process that fits within our current procedures to the best possible extent. This approach has given us a greater degree of control along with increased speed. We have held ourselves back by not applying this process to underlying software, considering the changes we are introducing as "flight configuration" rather than full deployments. However they are full deployments, and by using DevOps techniques combined with these technologies, we achieve a greater degree of speed and control – at the same time.

We strived to streamline software changes by fostering collaboration, embracing DevOps principles, and leveraging the capabilities of GitLab. The software change process was slow and heavily reliant on many asynchronous, manual steps, causing waits for the engineers involved. Moreover, the need to collaborate with external companies using GitLab revealed inefficiencies in integrating updates with SVN. As engineers from different divisions with diverse responsibilities, we realized we shared a common issue, which also presented an interesting opportunity.

We migrated our projects to GitLab and integrated CI/CD pipelines, overcoming the inefficiencies of the SVN-based workflow. Frequent, often informal meetings were the backbone of the communication after an initial meeting with the management of all teams involved that got us on the same page for how we would move forward. The change was achieved using GitLab for version control, GitLab Runners for automation, Ansible for rollouts, and existing CM processes for manual deployment authorization.

This transformation had significant impacts. Tasks that previously took hours now take minutes, improving responsiveness and the rate at which we can learn. Collaboration became easier with streamlined workflows and teamwork with external partners. Resource optimization was achieved by reducing reliance on librarians for packaging and deployment tasks, freeing them for higher-value activities. Innovations in process and tools were introduced, with CI/CD pipelines automating packaging, testing, and deployment while ensuring compliance.

The success of this initiative required cross-domain collaboration, with engineers and software librarians sharing their expertise across different domains, fostering effective teamwork and problem-solving. The point was not just to execute, but to scale our work continuously. A shared vision and understanding of the benefits—efficiency, automation, and improved quality of life—aligned everyone's efforts and motivations. Consistent management support and trust, built over years of collaboration, provided the foundation for a smooth transition. Everyone involved embraced a culture of learning, iteration, and knowledge-sharing, ensuring each step brought lasting improvements.

This serves as a template for how things can work at EUMETSAT within the existing processes and structures. Of course, we can improve further, and we intend to do so, as will be described in the next section.

4.2. Driving Organizational Change Through DevSecOps Governance

As EUMETSAT advances in its DevOps maturity journey, the organization is transitioning from an ad-hoc adoption of DevOps practices towards a structured, governance-driven model that aligns with its long-term cloud-based service strategy. The introduction of Big Data Services into operations and the adoption of DevSecOps pilots in Programme Preparations are serving as catalysts for this transformation. These initiatives underscore the need for enterprise-wide DevSecOps governance, ensuring that modernization efforts are scalable, consistent, and aligned with operational and programmatic objectives.

Recognizing the need to evolve beyond fragmented DevOps implementations, EUMETSAT is establishing a DevSecOps Working Group (DWG) and a Common DevSecOps Framework (CDF) to consolidate process automation, infrastructure-as-code (IaC), and CI/CD adoption across teams. This move is a critical step in progressing from the "Managed" stage towards the "Defined" stage in EUMETSAT's DevOps Maturity Model, reinforcing a cohesive, scalable approach to software development and deployment.

It is hoped the structured adoption of DevSecOps principles leads to fundamental shifts in EUMETSAT's organizational landscape, affecting four key areas:

1. *Alignment with Organizational Objectives (OO)*
    - The Ground System Board (GSB) now plays a central role in endorsing DevSecOps-based recommendations that impact cross-functional areas.
    - The Functional Area Management Boards (FAMBs) integrate DevSecOps recommendations into the broader technical roadmap, ensuring cross-departmental alignment.
    - The goal is to shifted from isolated development pipelines to an enterprise-wide Common DevSecOps Framework (CDF) that fosters automation, integration, and security standardization.
2. Creation of the DevSecOps (DWG) Working Group.
    - A dedicated DevSecOps Working Group is being established as a functional, independent entity to develop and refine technical recommendations.
    - The DWG operates autonomously, generating guidelines, frameworks, and best practices while ensuring they remain free from external influence or bias.
    - The DWG interacts with FAMBs for cross-functional impact assessment before recommendations are submitted for final endorsement by the GSCG/GSB.
3. Expansion of Expertise and Targeted Recruitment
    - Recognizing the need for specialized skills, new expert roles have been created within the DevSecOps framework, focusing on: CI/CD pipeline development, Infrastructure as Code (IaC), Automated Testing and Security, Monitoring & Performance Optimization
    - The recruitment strategy has evolved to ensure EUMETSAT acquires the necessary DevSecOps engineering expertise, particularly for continuous integration, verification, validation, and security automation.
4. Implementation of Pilot Prototypes in Programme Development
    - DevSecOps principles are being tested in pilot projects in Programme Development of Processors (MTG UVN and Sterna new Programme) and in Projects phases for setting up the platform, having the DevSecOps team work together with the Platform engineers. The pilots aim in demonstrating how automated deployments and CI/CD pipelines could accelerate deployment cycles also for Programme Development.

Lessons learned on DevOps from both Operations and Programme Pilots are influencing the shift from ad-hoc DevOps adoption to an enterprise-wide strategic implementation

4.3. The DevOps Governance Model: Ensuring Sustainable Adoption

To safeguard the scalability and consistency of DevSecOps adoption, a three-tier governance model is being proposed:

1. *Tier 3 - DevOps Working Group (DWG)*
   - Develops frameworks, automation strategies, and technical best practices.
   - Ensures alignment with evolving cloud-native methodologies.
2. *Tier 2 - Functional Area Management Board (FAMB)*
   - Conducts cross-functional impact assessments.
   - Aligns DevOps strategies with system engineering and operational policies.
3. *Tier 1 - Ground System Board (GSCG/GSB)*
   - Provides strategic oversight and endorses enterprise-wide DevSecOps adoption.

This governance structure facilitates a structured, scalable DevSecOps transformation, ensuring that automation, security, and integration efforts are systematically aligned with EUMETSAT's cloud and modernization strategy.

The governance model aims in establishing technical independence while integrating DevSecOps recommendations into organizational decision-making.

4.4. Key Challenges and Next Steps

As EUMETSAT moves towards a more mature DevSecOps model, several key questions remain central to the ongoing transformation:

1. *How Can DevOps Improve the Transition Between Programme Development and Operations*?
   - Historically, Programme Development and Operations have functioned as distinct phases, leading to:
     - Integration bottlenecks
     - Manual and Re-engineering rework
     - Silos teamwork and knowledge transfer inefficiencies
   - Embedding CI/CD pipelines and Infrastructure as Code (IaC) from early development stages into our mission non-critical environment will help automate system validation and reduce deployment risks before scaling out into mission critical environments used for production by Operations.
2. *How Can DevOps Enhance Multi-Mission Operations Stability, Maintainability and Scalability at the same time*?
   - EUMETSAT operates a multi-mission ground systems where real-time operational stability is critical, yet the new needs, obsolescence and evolution are key for their maintainability and scalability.
   - Automated testing, continuous monitoring, and configuration management integrated in a common DevSecOps framework and new processes aiming to reduce time and costs on customization capabilities for programmes and enhance maintenance responsiveness for operations, enabling team collaboration to react quickly on changes maintaining security and compliance. However, this requires a significant up-front investment.

6. Conclusions

This paper has outlined EUMETSAT's current DevOps maturity level, highlighting progress, challenges, and future directions. While EUMETSAT is still in the "Managed" stage of DevOps maturity, the steps taken—such as the establishment of the DevOps Working Group, the Common DevOps Framework, and structured governance mechanisms—mark significant progress toward a more structured, scalable, and secure DevOps approach.

Key takeaways from EUMETSAT's DevOps journey include:

- DevOps is not a single tool or process but rather a cultural and structural shift that requires governance, automation, and cross-functional collaboration.
- The formation of the DWG and the structured governance model have been pivotal in ensuring organization-wide consistency in DevOps adoption.
- Automation, CI/CD, and Infrastructure as Code (IaC) are becoming core enablers of multi-mission operational efficiency.
- DevOps pilots in Programme Development are bridging the gap between development and operations, reducing deployment risks and accelerating system validation cycles.

-

Looking ahead, EUMETSAT's next phase in DevOps maturity involves:

- Scaling CI/CD adoption to additional operational and programmatic domains.
- Strengthening automation and monitoring to enhance real-time operational resilience.
- Further integrating DevOps into cross-agency cloud services, ensuring seamless interoperability across evolving cloud-based infrastructure.

While EUMETSAT has made significant strides, the journey towards a fully integrated, monitored DevOps model is ongoing. The lessons learned and the governance framework now in place will serve as a foundation for accelerating modernization, ensuring agility, and maintaining mission-critical stability.

While we would rank ourselves as only in a relatively low maturity stage regarding embracing DevOps as a concept, it is important to acknowledge the steps already made, as we believe the team culture the singular most important facet. To this end EUMETSAT is benefitted by having an organisational structure that lends itself naturally towards a DevOps style of working. Departments and teams that have clear and separate responsibilities for developing, maintaining and operating functions of the ground segment and multi mission systems come together under a matrix organisation, facilitated by processes such as the FAMB. With this, operations are closely embedded in the roadmaps, decision making and development of systems.

At a more granular scale, we structure ourselves such that singular operations teams are solely responsible for small, repetitive and fast paced changes. This helps enable fast turnaround; however a full CI/CD pipeline would assist even further in enabling such change. While we already employ DevOps enabling tools and software such as GitLab and Ansible, there is still work to be done at both a technical and managerial level to fully integrate such changes into such pipelines and ensure compatibility with our change scheduling and configuration management processes. We continue to emphasise that a shift towards DevSecOps should not be read as a reduction in the quality of changes and appropriate checks. Indeed, having a lean process for standard and low risk changes can mean a refocus and better scrutiny on changes out of the ordinary.

The pathfinders we have identified in this paper have been a useful input for a wider management level discussion on how we can embed more DevSecOps in our business logic within a defined framework and governance model, and a DevSecOps working group consisting of stakeholders from across the organisation to assist with it. The transition therefore starts to build up pace, with targeted recruitments and contractual expertise, targeted organisational objectives, pilot activities and alignment with organisational objectives.

While the concept of "not fearing DevOps because we already do it" is deliberately provocative, there is a genuine belief this is the case, at least in terms of how the organisational structure lends itself and the initial steps taken. As this paper reflects on however DevOps is also not black and white, but a path to follow under a robust governance. It is clear that we are still at the beginning of this path, however we are keen to share our experiences with others so far and gain their insight into what challenges they face, such that we can continue to develop and move towards a mature DevSecOps model.

References

[1] Yevgeniy Brikman, Fundamentals of Devops and Software Delivery: A Hands-On Guide to Deploying and Managing Software in Production, O'Reilly, 2025

[2] Gene Kim, Jez Humble, Patrick Debois, John Willis, and Nicole Forsgrenm, The DevOps Handbook, Second Edition: How to Create World-Class Agility, Reliability, & Security in Technology Organizations, IT Revolution, 2021

[3] Nicole Forsgrenm, Gene Kim, and Jez Humble, Accelerate: The Science of Lean Software and Devops: Building and Scaling High Performing Technology Organizations, Trade Select, 2018

[4] Smith, Bradley, DevOps for the desperate: a hands-on survival guide, No Starch Press, 2022