An approach for industrialising software systems for ESOC Operations

James Eggleston

21/02/2023

Agenda

ESOC's Use Case and Constraints

- How we used the traditional method
- Its successes, its failures
- → the ESOC Problem Statement: how to empower industry, but remain in business

A paradigm shift: partnerships through software communities

- ESA Public and Community licenses

Another paradigm shift: partnerships through end-to-end services

- Enabling a coherent DevOps based approach across ESOC

The ESOC Use Case

Keeping satellites healthy and delivering quality mission products for as long as possible

Varied Missions

- Routine operations 24/7 for unique Earth Observation, Astronomy & Interplanetary missions
- Including LEOPs, fly-bys, landings, …
- Also, for 3rd parties

Multiple missions, launches, events

- Multiple (currently 20) flying satellites, lifetimes 5-20 years after launch and counting
- Multiple missions in development, launching years in 1-4 years future

ESA mandate is to support ESA Member States industry

ESOC missions are operated from ESOC premises (today)

ESOC missions are facing increasing security requirements

ESOC Software Strategy since 2000

Avoid custom-made mission solutions, in order to avoid…

- individual maintenance contracts
- increased dependency count, higher rate of churn
- → ultimately increased maintenance and operations costs for ESOC overall

Avoid vendor lock-in, in order to avoid…

- issues from vendor sale, business difficulties
- enforced updates v expensive dedicated support
- (unplanned) product termination

(re-)Use of a common infrastructure for IT and mission ground segment software

- IT layers are 100% re-used across missions, using commodity hardware and services
- Commodity OS and commercial solutions, based on open source
- Bespoke software layers are owned by ESA – “operational software” clause
  - 90-100% common across missions and ground stations
  - Make software available to industry for their re-use in commercial applications

Successes …

➔ lowered costs ➔ lowered risks ➔ increased speed

Over 15 years, product suites such as

MICONYS (featuring SCOS-2000) for mission control systems and

SIMULUS (based on SIMSAT) for simulation, and

libraries implementing CCSDS standards (e.g. SLE, CFDP)

Were

Migrated to (at least) major 5 IT and OS baselines

Used for launch preparation and mission operations for dozens of missions

Licensed hundreds of times for commercial, academic and institutional use

… and Problems

Only driver for code evolution was ESOC

- Yearly pressure on budgets for maintenance, sustaining and evolution efforts
- Growing software footprint
- → lower funds available per system

Systems are re-usable, but not designed for re-use - ready to deploy systems, but for a vanilla use-case…

- Tailoring of systems often required deep knowledge of system internals and design history…
- Changes to systems – especially to older systems - (often) needed system interface changes
- Repatriation of changes to master codebase difficult, poor value for money
- → Missions often had branched solutions

Mission and infrastructure development under different control, with different concerns

- silos form based on local concerns, i.e. software first v mission first
- → Common software is only the first step, common software processes needed as well

ESOC is the only driver…

Paradigm Shift #1 – Software Communities

Use of the Operation Software clause is unusual in ESA

- Normally industry retains ownership of intellectual property

ESA being the sole owner presents the same problems to industry and partners that ESA want to avoid!

- Industry are dependent on ESOC releasing products, and updates in a regular manner
- Industry could not rely on irregular and lengthy agency process for rapid business needs
- → Industry will not invest in building a product that they cannot sell outside ESOC
- ➔ Innovation is controlled by ESOC investment limits
- ➔ ESOC products not commercially viable for industry

➔ ESOC Ground Data Systems Infrastructure decided to move to a business model supporting partnerships through software communities

- Approval of ESA Software Licenses provided an alternative path to…
  - Innovation and supply of industrial products based on ESOC de-facto standard solutions
  - Avoid vendor lock-in

The ESA Software Licenses

- ESA defined 2 collaboration software licenses:
  - **European Space Agency Software Public License (open source)**
  - **European Space Agency Software Community License (restricted to ESA MS territories)**
- Each license comes in three flavours: permissive, weak copyleft, strong copyleft
- Cover ESA Convention requirements, esp.:
  - Arbitration clause
  - Jurisdiction of law

The ESA Community Software Licenses

ESA Software Community License (ESCL) version 2.4 flavours:

- **Permissive**

Can: Use, Modify, Distribute, Sublicense (under *any* license)
Cannot: Hold Liable, Use Trademark
Must: Include Copyright, Include Notices

- **Strong Copyleft**

Can: Use, Modify, Distribute (under *same* license)
Cannot: Sublicense, Hold Liable, Use Trademark
Must: Include Copyright, Include Notices, *Disclose Source*

- **Weak Copyleft**

→ Similar conditions to *Strong Copyleft* for **Covered Code** (i.e. project)
→ Similar conditions to *Permissive* for **External Code** (e.g. combination)

⇨ In all cases: Use is restricted to ESA Member States territories
⇨ ESCL Weak Copyleft is so far the most commonly used license in ESA
⇨ **License texts are available at** **https://essr.esa.int/license/list**
⇨ **Explanations on the license are additionally available at**
⇨ https://essr.esa.int/license/european-space-agency-public-license-esa-pl-commentary-v2-3

ESA Community License and Communities

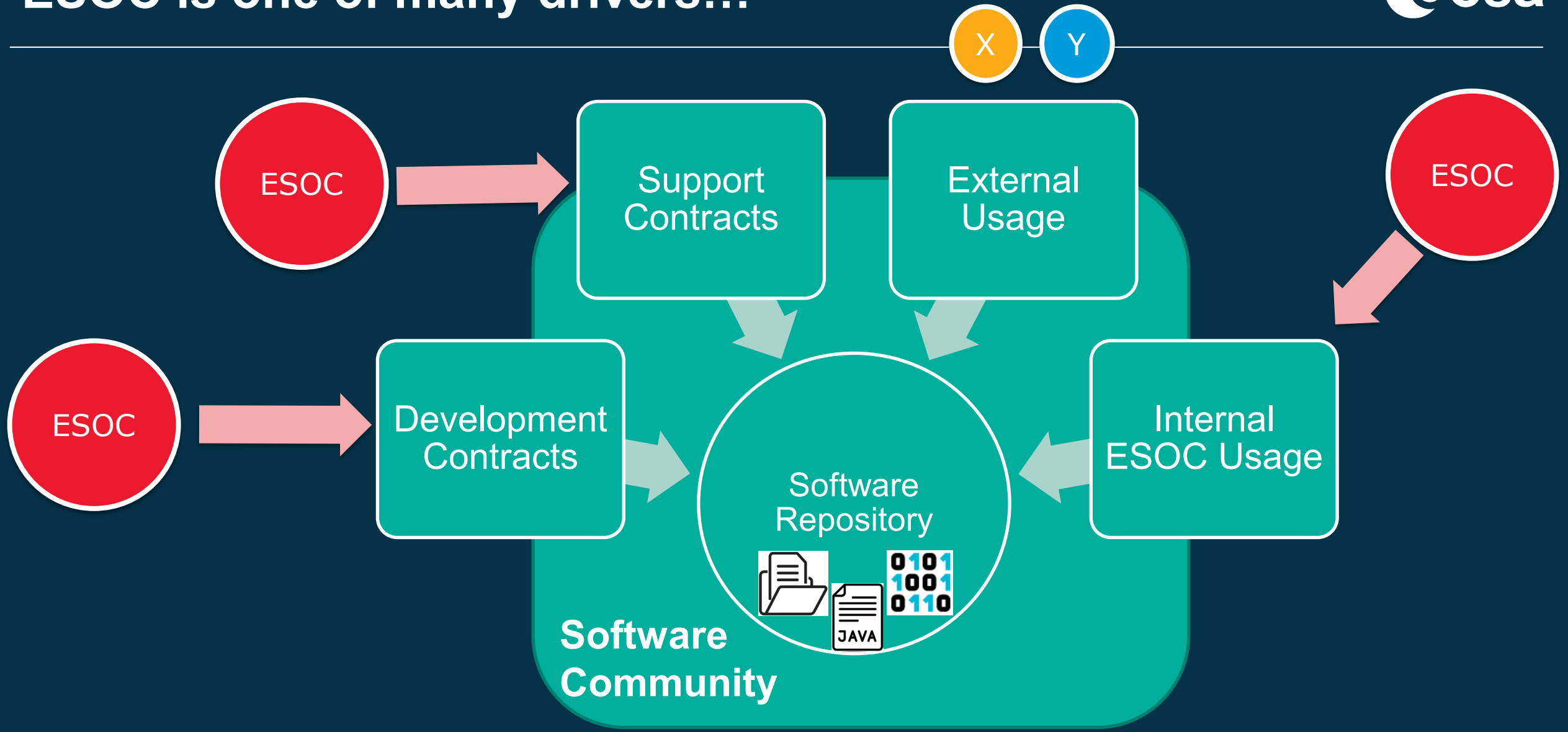
- ESA Software Community License – ESCL (Weak type)
  - The license includes a copyleft clause, where copyleft exceptions allow for extensions to be distributed under a different license (including proprietary licenses)
  - License terms allow for direct re-distribution of the software within the ESA Member States
  - When distributing the software, its source code and modifications must be made available to the provided party under the same ESA Community License – However, extensions can be closed-source under a proprietary license
- ESCL enables collaborative community approaches : Opening new business opportunities for the industry
  - Enable creation of partnerships to co-design and evolve the (base) community software
  - Help businesses to more rapidly develop a proprietary commercial product around an innovation
  - Allow addition of added-values and new innovations with business freedom provided by the ESCL license
- Various user cases possible. For example :
  1. ESA Operational Software, released under Community License, extended by a business to create an enhanced version of that software, itself released as a commercial product to the market
  2. Collaborative development with multiple companies/entities contributions, resulting in a shared "*open sourced*" software base
     - Community effect with optional participation of ESA
  3. Dual Licensing : Author keeps all IPR (non-exclusive) but also releases a Community version of the software

ESA means of software distribution

gitlab.esa.int	gitlab.space-codev.org	github.com/esa	essr.esa.int
“Inner Source” Community limited to ESA staffs and contractors (under active contract)	“Open Source” Community limited to ESA Member States	Open Source Community World Wide	No community interactions
Access associated to ESA Contract Restricted Communities	ESA Software Community Licenses	ESA Software Public Licenses	Any license

CoDeV - https://www.space-codev.org

ESOC is one of many drivers…



Benefits of Communities

For ESOC

- Supports Mission Operations data system software in Europe **centre of excellence**
- Eases and promotes the use of ESA **investments**
- **Simplified** software distribution and licensing scheme
  - e.g. exchange of Customer Furnished Items and Item Made Available between ESA projects and companies
    - **Eased collaboration** on software development between ESA and others
- Incentivises the existing industry and ESA **to be better**
  - Documentation, code quality, software design, builds working all the time, all now visible!
- **Potential** to lower the total cost of ownership for ESOC systems

For Industry & Partners

- **Simplified software distribution** and licensing
  - Capitalise **quickly** on new business opportunities
  - Allows (Inter-)National Organisations to **exchange**
- **Increased access** to reusable components and working software
  - All versions of the software are available
- **New business opportunities**
  - **Build additions or improvements** to address shortcomings
  - **Commercial productisation** of the systems
  - Be able to offer company **consulting services** (to ESA, or for commercial opportunities)
  - Improve team **skills** and company **reputation**

Paradigm Shift #2 – end-to-end services

Common Software, used by all missions and users is not enough

Need common software, used by all missions and users, in the same way

→ Common Processes

The EGOS-MG project …

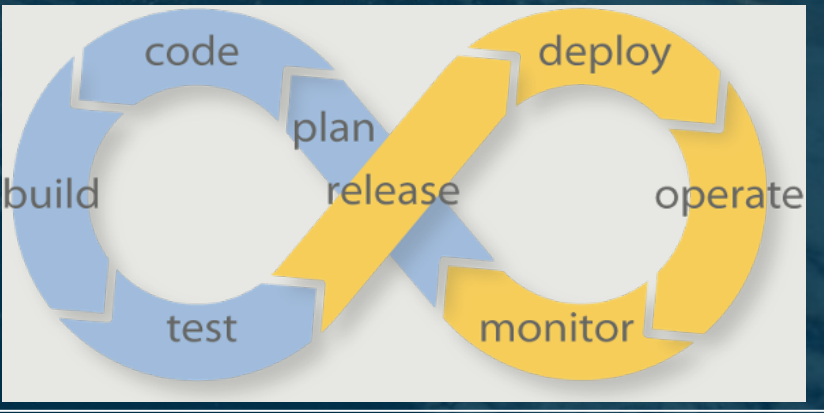
create a multi-mission infrastructure, converting the ESOC deployment model away from mission specific

Adoption of industry standard DevSecOps for systems and for flight procedures

To empower rapid roll out of validated, systems to multiple mission on a regular basis

Allow industry innovation for end user, roll out/back new versions flexibly to specific groups

EGOS MG does not evolve end user functionality



EGOS-MG: An integrated System of Multi-Mission Applications and Services

Application layer: Generic Applications supporting the complete Operations lifecycle for all missions

Prepare mission operations
- Space segment data definitions
- Operations Procedures

Validate mission operations
- Operations Procedures
- Operations Timelines
- Planning Rules
- High-fidelity simulations

Plan mission operations
- Ground operations
- Platform operations
- 'Payload operations

Execute mission operations
- Manual operations
- Automated operations
- Real-time and deferred data processing

Manage operational data
- Evaluate mission
- Generate reports
- Disseminate data

Service layer: Application level Multi-mission Services supporting the complete Mission lifecycle

Set-up mission
- Manage team
- Customise system and allocate resources
- Develop mission specific artefacts

Validate mission support
- Verify mission customisation
- Execute reference validation scenarios in the mission context

Support mission
- System engineering
- Incident management
- Maintenance

Infrastructure layer: Integrated Multi-mission Environments supporting the complete System lifecycle

Development Environment
- Multi-mission Infrastructure
- Generic Applications
- Mission Systems

AIV Environment
- System validation
- Missions operational validation

Operations environment
- Run-time infrastructure
- Multi-mission systems
- Mission Systems

EGOS-MG Targets Commercial Services

EGOS MG promotes a model where services can be tendered to provide specific services

Successful bidders will provide a specific service for all missions for N years

- Industry has a strong commercial incentive to invest in creating a compelling service
- ESOC avoids a vendor lock-in thanks to Community License approach
- ESOC is free to ask other industry to contribute to the Community, e.g. evolution, study, …

Unifying Processes around each product allow other services to monitor and interact with the product

Why impose a specific DevOps approach?

ESOC needs to rely on long-term viable processes

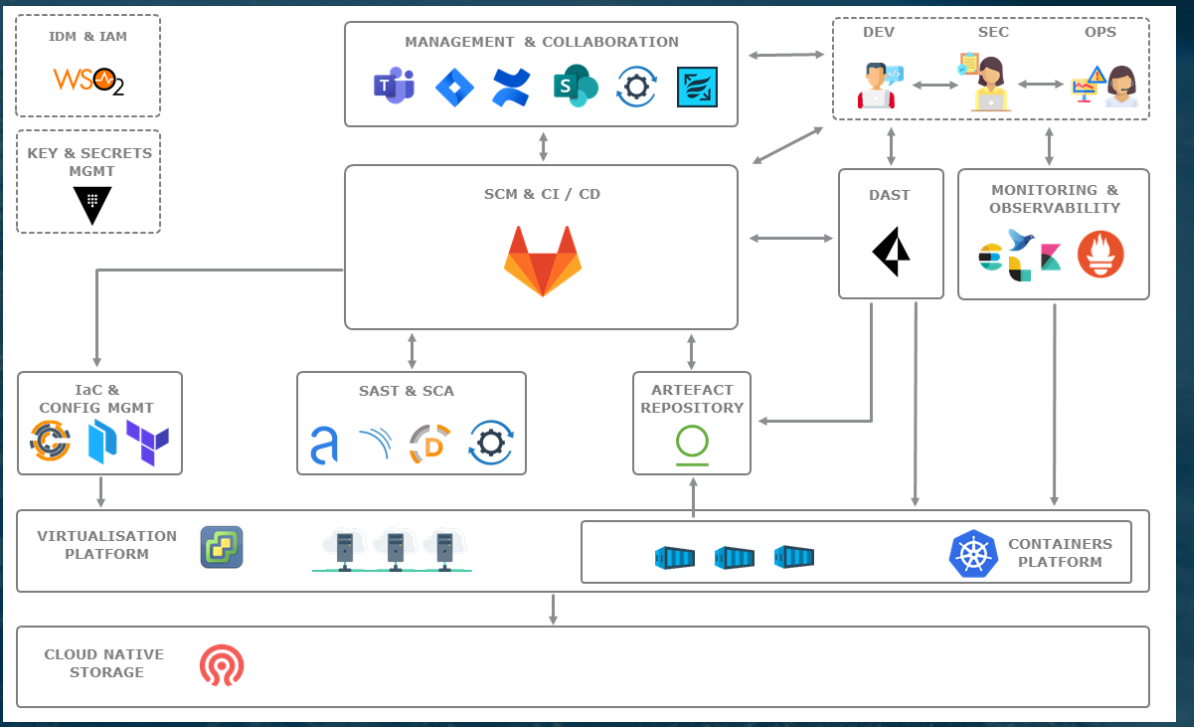
ESOC has strong Operational needs

→ Standard tools and approaches for

- Deployment
- Configuration
- Resource Monitoring
- Alerting
- Security
- IT support

➔ efficiently operate the all systems for all missions and phases

➔ EGOS MG will define the reference solution that systems must follow



Use of External Commercial Solutions in EGOS-MG

Implies the components chosen can ‘fit’ into the ESOC Ground Data System

architecture → same software interface

processes → same or compatible DevOps processes

Software can be obtained from the community → likely the result will be based on the community product

Processes can be obtained from ESOC, and are relatively standard

Where standards exist, this is feasible

e.g. use of CCSDS Space Link Extension (SLE) to communicate between MCS and Spacecraft, via commercial ground stations

No standard? Can be a quick return to vendor or product lock-in, or an ESOC only product, or a product that is unique to operate

Paradigm Shift #3 – what more is needed?

Enable operability of different ground data system components

1) by standardization e.g. CCSDS MOIMS

2) by full Model Based System View

→ Would allow competing products for the same ‘slot’ in a mission’s functional stack, allowing different missions to choose different tools for the same function

→ Would allow competing products for the same ‘slot’ in a full ground segment

Is this a good idea for training users and admins? Perhaps only if remote services are being integrated?

Enable distribution of different ground data system components to multiple remote locations

Ground segment functions purchased from vendors, integrated and executed across multiple locations

Considering all necessary security, operational and performance constraints

Conclusions

How to take advantage of rapid innovations and commoditization of commercial space services?

How to position your agency to effectively acquire ground services rather than developing and purchasing ground systems?

How to take advantage of industry's willingness to invest in new ideas and take business and technological risks?

ESOC is embracing an open community model for all its ground data systems. This will

- increase access, familiarization, reuse of generic ESOC systems
- empower industry to create commercial solutions based on and around ESOC systems
- increase compatibility with commercial offerings

ESOC is moving toward a service-based approach for all ground data system operations. This will

- allow innovation by industry to improve the end-user experience
- Increase innovation in ESOC operations

To enable use of fully commercial products on a mission by mission or function by function basis, further steps are needed,

- defining standards for the ground data segment, e.g. CCSDS MOIMS
- defining processes for meeting needs of spacecraft operations