Formation Flying EMCS based on the ESTEC EMCS Reference Facility

Massimiliano Mazza[(1)], Guy Fokou[(2)]

[(1)]*Vitrociset Belgium, permanent establishment in the Netherlands*
'sGravendijkseweg, 53 – 2201CZ Noordwijk (NL)
Email: m.mazza@vitrociset.be

[(2)]*Vitrociset Belgium S.p.r.l.,*
Rue Devant les Hêtres 2 – B-6890 Transinne. Belgium
Email: g.fokou@vitrociset.be

1 INTRODUCTION

Scientists have just begun to understand the full potential of space vehicle formation flying. In the last few years, this technology has gone from a space oddity - and a high risk one at that - to a concept fully embraced by earth and space scientists around the world. Simultaneous measurements by multiple formation flying vehicles can provide substantial benefits. Distributed spacecraft technologies enable higher resolution imagery and interferometry, robust and redundant fault-tolerant spacecraft system architectures, and complex networks dispersed over clusters of satellites in space.

Obviously formation flying does not mean just launching multiple satellites. Not even operating constellations like for navigation (even though these can benefit from newer capabilities in the operations front-ends).

A formation flying mission can be seen like a team-work between the involved spacecrafts, where the team is the actor and the single unit is not in the conditions to carry out the meant achievements. More specifically Formation Flying missions are where more than one spacecraft coordinate to conduct a mission whose relative states are rigidly fixed, measured and controlled autonomously in closed loop.

Like in acrobatic aeronautical teams the formation flying relies exactly in a perfectly coordinated attitude, guidance and navigation of the involved bodies. The formation is therefore meant to behave like a single, distributed, object.

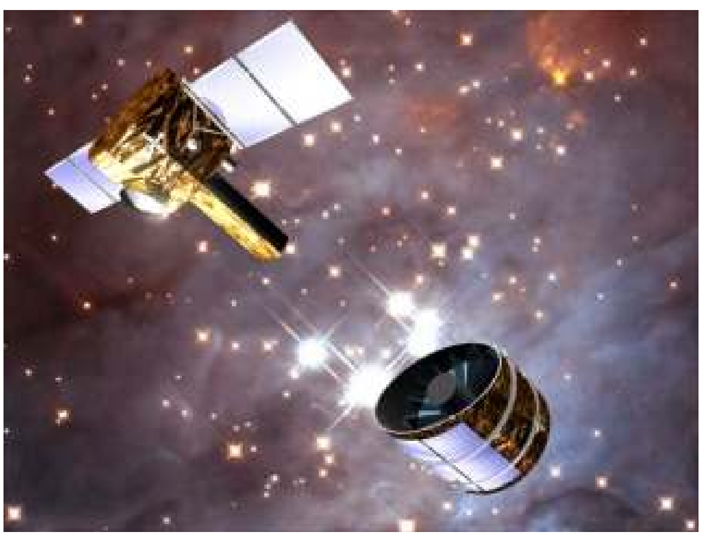

Fig. 1: artist impression of the XEUS mission

Formation flying can be achieved with different levels of automation, obviously increasing with the improvements of the technology and the possibility to implement more advanced algorithms. As much as in space this should reflect on the ground, either in the operations either in the ground support equipments.

This paper illustrates a formation flying EMCS architecture that is based on reuse of proven technologies able to support present scenarios and showing the capability to provide much more.

It does not want to provide an exhaustive description of the formation-flying philosophy and applied technologies, therefore it will not go in details on this subject.

2 THE CHALLENGE

The Formation Flying ground data systems most peculiar requirement is supporting the concept of a "Virtual Spacecraft". This means driving all involved spacecrafts as if they were only one through a set of predefined coordinated operations. This feature is reflected throughout preparation and execution subsystems for testing and operations. This family of applications provides a fairly new frontier of development for the tools and the subsystems involved in the definition, implementation and operation of the formation flying missions.

The challenge we want to face is achieve all the involved features by the customisation of proven and reliable applications developed around the prototype architecture of the ESTEC EMCS Reference Facility.

Despite the main interest of these new developments is focused on improving Mission Control Systems infrastructures, thanks to the duality embedded in the ERF, the immediate side effect of this approach materializes in EGSE applications with the comparable features almost free of charge.

2.1 The ESTEC EMCS Reference Facility

The idea behind the ESTEC EMCS Reference Facility is to implement a building blocks philosophy on Ground Data Systems facilities by exploiting the commonalities between EGSE and MCS applications together with the implementation of the ECSS standards E-70-31 (for what is concerning the Space System Model, SSM) and E-70-32 (through the PLUTO scripting language).

Its architecture is a customisation of the standard ESOC MCS (SCOS 2000 r3.1), with extensions to support an EGSE protocol (to manage SCOEs) and to be driven by an ECSS compliant scripting language. By this means the Database ICD, the way TC are formatted and created, the way the TM is decommutated and monitored are de facto reused from the MCS. Therefore the core of the ESTEC ERF can be deployed as a CCS as well as a MCS for a given mission.

This approach provides many advantages like:

1. reuse of components and applications in 2 separate scopes of the project with cost savings
2. preliminary validation of the MCS suite through use and troubleshoot of the CCS
3. validation of the very same Mission DB

The scheme of the EMCS concept is depicted in the Fig. 2 courtesy of ESA [1].

The main components of the architecture are:

- SCOS 2000 r3.1 with EGSE extension: provides TM/TC engine, monitoring and archiving facilities, TM display
- ASE (Automated Schedule Execution): (in the figure referred generically as Test & Operation Procedure Environment) provides PLUTO procedures and activities schedules preparation and execution environment in EGSE applications or Mission Automation features with operations schedules execution in MCS deployments (MATIS); embeds a front-end to the runtime SSM.
- EGSE Router: provides TM/TC/EV routing to/from SCOEs, BBE…
- The SSM database infrastructure: hosts the runtime MIB + SSM

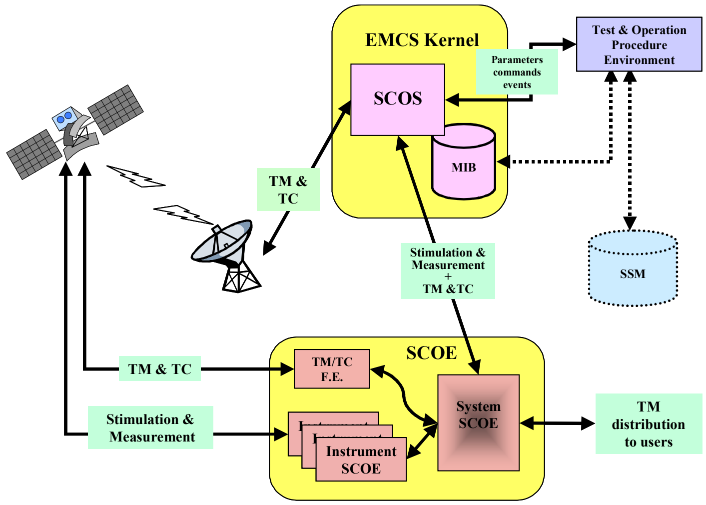

Fig. 2: EMCS Reference Facility architecture

2.2 Formation Flying EMCS based on the ERF

The ERF systems currently deployed and available with stable and proven releases have been designed for the command and control of a single platform. Therefore, as could be easily expected, this new challenge has been addressed with the development of new featured releases both for MCS (SCOS 2000 r4 and r5) and for EMCS (new ERF based on SCOS 2000 r5). Unfortunately space software needs several years to achieve maturity, due mostly to the extremely limited number of users and most of all of use cases. Reuse in space business is therefore considered good practice and is often encouraged or, better, required. Based on these considerations we decided to design an architecture based on a customisation of the actual ERF redirecting the multi-domain feature of the ASE developed for the VEGA EGSE to allow connections of multiple independent SCOS 2000 instances.

ASE can be described as a PLUTO framework with SSM support. The language itself does not impose any limitations in the context of multi-satellite operations; neither does the interpreter, since it was designed absolutely generic and independent from the kind of unit to be driven / tested. The SSM provides on top the perfect formalism for the representation of any kind of structured environment without limitations, embedding as well the native mapping of SCOS 2000 MIB to be extended with minor changes to multiple instantiation. This means that models of more spacecrafts can be mapped onto different branches of the very same Space System Model. This provides the advantage to have all of the involved information and activities available at the same time to the same procedures / schedules / displays, therefore getting the capability to command and control in a coordinated way the combined spacecraft operations (or stimulate and monitor all the hardware under test in CCS). The very same features could be then migrated to applications like MATIS (Mission Automation framework of the EGOS suite based on ASE).

The idea of the framework relies on two main concepts: activities and reporting data. These can be seen like methods and properties of an object oriented environment where this layer is meant to provide the abstraction and the homogeneity and the implementation is hidden to the user. In our case ASE implements them through drivers. In general this means that a PLUTO procedure or a schedule will see as activity every action that can be implemented through a driver within ASE, and will treat all of them in the same way independently of their implementation. Therefore as an example a PLUTO procedure could open a document editor, command an antenna positioning and send a TC through a second antenna with a sequence of three lines of code in the same PLUTO procedure.

When ASE introduced the multi-domain support this was in response to the requirement of providing the possibility to support multiple SCOEs of the same type in the VEGA EGSE. This means that the core of the application is already embedding this support and that a customisation should be implemented to apply this feature to the SCOS 2000 drivers.

The architecture we designed is shown in Fig. 3.

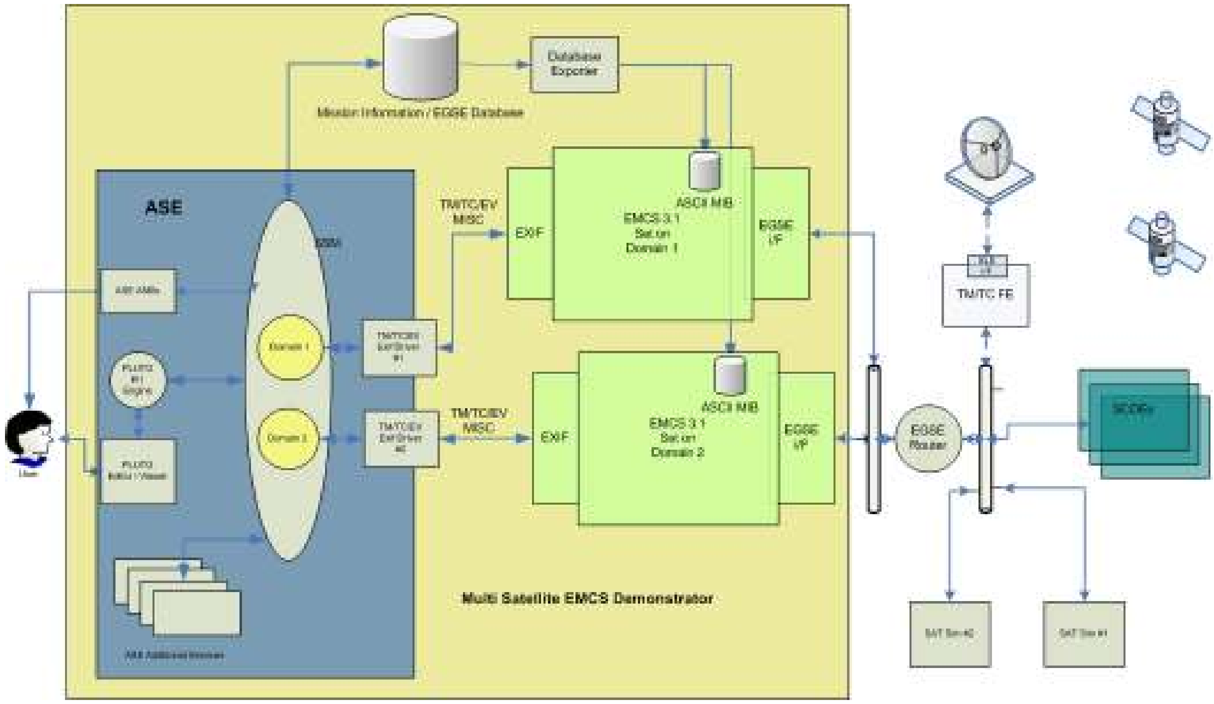

Fig. 3: Multi-domain EMCS architecture

Using multiple instances of SCOS 2000 implies having multiple MIB (one per domain). This will prevent issues in the duplication of SPID (the unique identifier for TM packets in s2k) avoiding the need to overload and compromise PUS compliance in the system. ASE on the other side is parameter based and is not affected by this kind of issues. It is indeed in ASE that data from the different platforms will meet for correlation and manipulation.

The EGSE Router of the ERF is the component dedicated to routing commands and TM correctly as instructed by the information stored in the header of the EGSE wrapper.

The advantage of this architecture relies in focusing the multi platform handling in the component with the highest automation capability, leaving hardworking tasks (like monitoring, calibration and archiving) to the SCOS 2000 servers. This way the power of the PLUTO scripting language could be used to improve the level of automation of the ground facility through watchdogs and scheduled tasks.

Such a multi-platform capable EMCS would be in the conditions to manage formation flying in two most common scenarios.

2.2.1 Master-Slave Formation Flying

In this case (proper F/F) the formation is handled by the master S/C being the only one commanded from ground. In these conditions ASE – SCOS would command the master only and monitor the two at the same time.

Upon reception of a formation TC the master S/C communicates to the slave the action to be taken in synchronisation and takes care of monitoring the execution of the coordinated motion / operation through a closed loop control.

During nominal operations only the TC chain for the master would be used while independent TM reception and S/C monitoring could still be performed on ground independently. Having a convergence of the reporting data in ASE would enable advanced monitoring capabilities (implemented by PLUTO code on parameters from more spacecrafts). In this context the Watchdog activities implemented in the frame of the VEGA EGSE CCS would apply consistently.

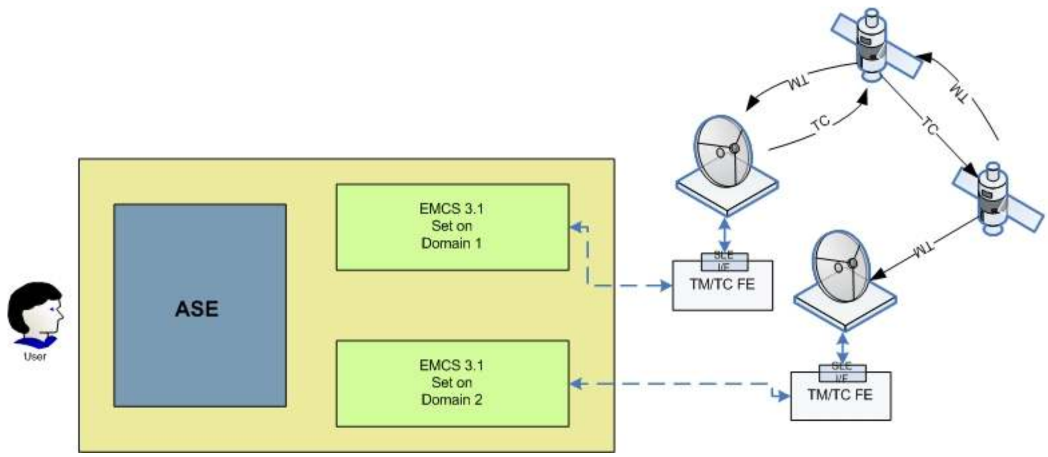

Fig. 4: Communications in master-slave formation flying

In this configuration the FF EMCS would still be in condition either to operate independently the satellites till the achievement of the formation either to perform independent testing during commissioning.

Independent satellite commanding capability is anyway an inevitable fallback for contingency situations and maintenance operations.

An EGSE designed for the formation flying could allow testing the closure of the virtual spacecraft command and control loop in a meaningfully representative way.

2.2.2 Ground driven Formation Flying

This is the condition (better known as swarm flying) where the formation and the coordination mechanisms are managed by ground. This situation applies when the timing in the closure of the loop in the mutual attitude and guidance of the spacecrafts could be less stringent for what is regarding the safety of the operations (higher distance between spacecrafts, looser tolerance in the mutual position). In this case, with the presented architecture, ASE would be in the component charge to manage the coordination of the operations. The advanced logic that is possible to implement within the PLUTO language, together with the possibility to run multiple activities in parallel is able to realise complex algorithms. In this case ASE would close the loop independently on each spacecraft.

This situation is equivalent to the transient phase of the previous case before the formation is acquired.

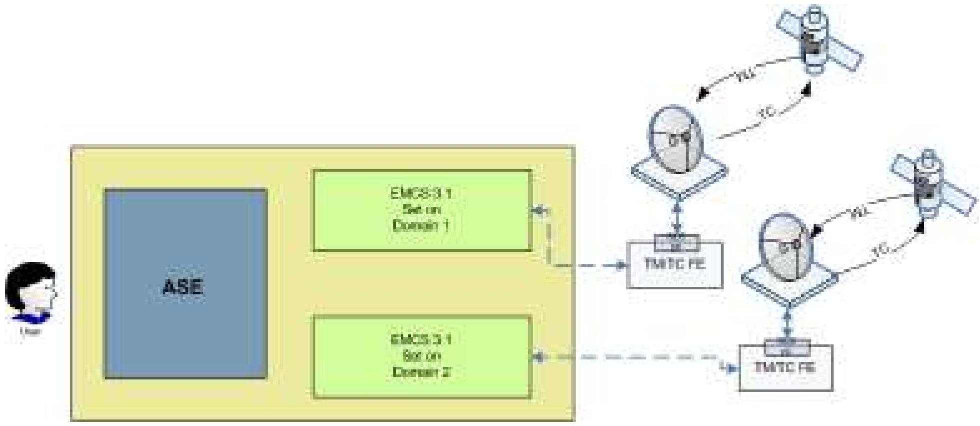

Fig. 5: communications in swarm type of formation flying

2.2.3 Applicability and scope of the presented solution

As shown either one or the other usages have absolutely no impact on the architecture of the EMCS and differences rely only on the Spacecraft capabilities and the MIB population.

It is fairly evident how all of above could be functional to in-flight operations. It may be less how this could apply to testing and validation. Investigation and search did not bring results while looking for past experience on specific EGSE developments for swarm or formation flying projects. Nevertheless the possibility to have this capability would clearly open the path to new possibilities in multi-satellites testing campaigns and a highly improved accuracy of SVT campaigns versus in flight operations.

This reuse is based on the suite of applications from the final version of the VEGA-EGSE CCS architecture. Due to the challenging performances achieved in this project, running on hardware less performing of what actually available on the market, we do not envisage any issue in the ability of the presented architecture to withstand the expected workload in the cases discussed above.

3 CONCLUSIONS AND FURTHER WORK

Based on thorough analysis and on previous experiences we believe in the possibility to implement an EMCS able to satisfy all requirements related to formation flying missions and swarms through reuse and customisation of proven and performing components.

The compliance of the requirements proposed for MCS would automatically reflect into unforeseen features with a high potential in improving integration testing and verification with an overall reduction in costs induced by the reuse of the same component between EGSE and MCS. The second induced advantage, as embedded in the concept of the EMCS ERF, would be in benefiting from a completely homogeneous handling of commanding and reporting between testing and operations with the immediate possibility to transfer and exchange back and forth experiences between AIV and FOS experts (critical during commissioning and contingencies handling).

The presented solution is compliant to ECSS-E-70-31 and ECSS-E-70-32 as inherited from the ESTEC ERF and does not foresee the risk of abusing of the PUS standard boundaries to allow the mapping of TM/TC specifications for the various satellites plus the virtual spacecraft on a single MIB. Therefore it is proposed as an implementation highly devoted to standardisation and to the follow-up of the roadmap traced by the Agency and the ECSS committee on these subjects.

A GSTP activity is about to be started to implement an EMCS based on the architecture proposed in the present document. The use-case will be taken as an input for the complementation of the requirements for the next release of

ASE (ASE 5). As a first output of the expected work order will be implemented a demonstrator of the capabilities and indented features to propose to upcoming projects (mostly Proba 3, which inspired this architecture).

As side effect of the GSTP activity a lesson learned exercise will be carried out to migrate the pertinent features to the MATIS branch of the ASE family.

References

[1] S. Valera, "EMCS-SUM", Issue 4, Rev 0, pag. 9, *unpublished*, march 2004.

[2] F.H. Bauer, K. Hartman et al. "Enabling Spacecraft Formation Flying through Spaceborne GPS and Enhanced Automation Technologies", *Proceedings from ION-GPS Conference*, Sept 1999.

[3] TAS Proba3 Team, "PROBA 3 Formation Flying Demonstration Mission Phase A – Summary Report", *unpublished,* Sept. 2007