Migrating from GSOC's SCOS derivate GECCOS to a distributed EGS-CC operations environment based on CCSDS MO/MAL

Michael P. Geyer[(1)], Armin Braun[(1)], Stefan Gärtner[(1)], Philipp Hamacher[(1)], Leonard Schlag[(1)], Anne-Katrin Schroeder-Lanz[(1)], Christian Stangl[(1)], Nico Trebbin[(2)]

[(1)]Deutsches Zentrum für Luft- und Raumfahrt
Münchener Straße 20, 82234 Weßling, Germany
Michael.Geyer@dlr.de, Armin.Braun@dlr.de,
Stefan.Gaertner@dlr.de, Philipp.Hamacher@dlr.de,
Leonard.Schlag@dlr.de, Anne-Katrin.Schroeder-Lanz@dlr.de,
Christian.Stangl@dlr.de

[(2)]LSE Space GmbH
Argelsrieder Feld 22, 82234 Weßling, Germany
Nico.Trebbin@dlr.de

INTRODUCTION

With the German Space Operation Center's (GSOC), a department of the German Aerospace Center (DLR), ongoing concept study, we will verify a migration path to a distributed, service oriented CCSDS Mission Operations (MO) environment based on Message Abstraction Layer (MAL) [8,9] and the European Ground Segment Common Core (EGS-CC) [7]. As EGS-CC will intrinsically support the exchange of monitoring and control (M&C) models, procedures and other data between satellite AIT and mission operations, it will allow GSOC to consolidate the path which has been started by exchanging SCOS MIB databases and procedure input with satellite manufacturers. We will establish a prototype CCSDS/MO service architecture where standardised service interfaces for telemetry and command are available for internal subsystems as well as for external partners.

Though GSOC has been continuously improving and extending its telemetry and commanding system GECCOS (GSOC Enhanced Command- and Control System for Operating Spacecrafts, [6]), a SCOS derivate, to support new mission needs, the end of the SCOS age not only at DLR is coming nearer. Maintaining a system with source code and an architecture based on concepts of the 90ies will be more and more painful. GSOC's GECCOS has been extended with features supporting automation and procedures; however the new European solution EGS-CC can now be seen on the horizon. It is expected to support modern operation concepts and simplify data exchange between manufacturers and operators. Future M&C systems will presumably be based on CCSDS MO service oriented architectures

The new M&C test system will be set up right inside the productive environment, including GSOC's standard security mechanisms from user login to access controlled doors, to be as close to the operational environment as possible. As a first proof of concept, we will establish a test system based on GECCOS to demonstrate the feasibility of using MO services and interfaces inside GSOC's secure operational network structure. We will ensure that the new concept can cope with our performance and security requirements. In a second step, we will allow some test system services like telemetry and command to connect to clients located in GSOC's less secure office environment. We intend to cooperate with GSOC-external partners to set up a complete distributed test environment, so that an attached simulator or engineering model may be operated from a partner's AIT site outside GSOC. Obstacles to be overcome are internal and external network boundaries guarded by inevitable firewalls and general security concerns.

In a final step, we will exchange GSOC's commanding and telemetry processing kernel GECCOS by the new common core EGS-CC. With this system setup, we will verify the suitability of EGS-CC for GSOC's environment, including also a basic communication test with a flying mission, based on a simplified, reduced M&C model database (Fig. 1).

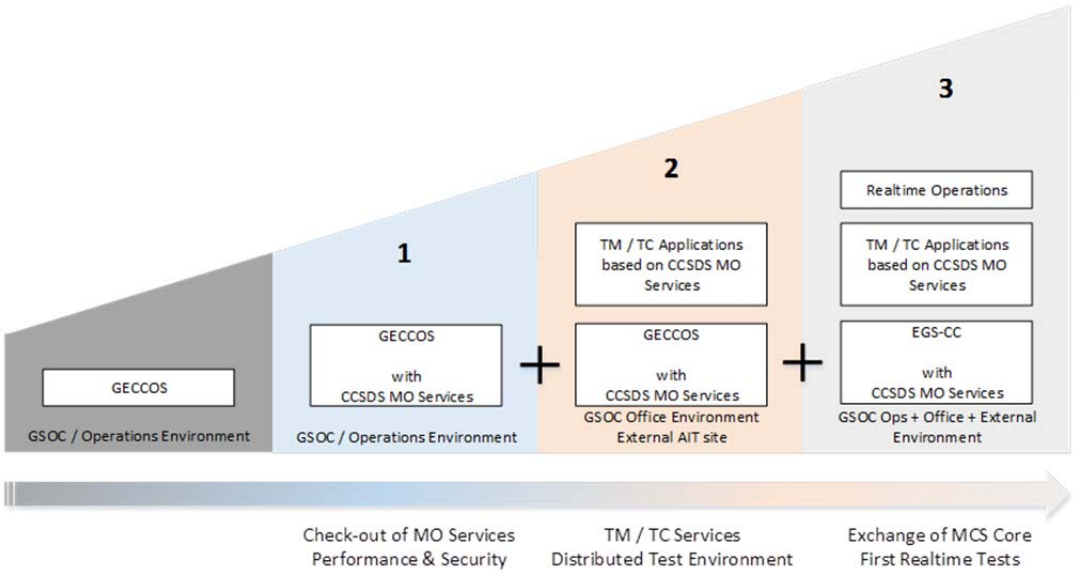

Fig.1. Evolution stages from GSOC's actual operations environment to a distributed system.

This M&C test system will then be the basis for cooperation with spacecraft manufacturers and their AIT environment for M&C database exchange and performing e.g. end-to-end system validation tests or integrated simulations.

In this paper, we start with the requirements on a modern flight operation systems (FOS), compare this to the existing GSOC solution and show a possible migration path to a renewed multi-mission environment.

CUSTOMER DEMANDS ON A SPACERAFT CONTROL CENTER BY AIT AND OPERATIONS

GSOC develops and operates FOS in close collaboration with DLR internal and external partners. Future demands for spacecraft operation missions, going beyond the actual ongoing projects, have been gathered from several sources:

- Experiences from requirement engineering, system implementation and integration phases
- Experiences established during longtime nominal and off-nominal operations
- Experiences from cooperation with spacecraft manufacturers during assembly, integration and testing (AIT).

Key Features of a Modern FOS

The customer demands and the system scenarios as described in the foregoing section lead to the following basic requirements on a multi-mission FOS deployed at GSOC and other sites:

- **Modular:** It shall be possible to set up a FOS independent of where the individual functions and processes are executed.

- **Scalable:** Modularity implies that the FOS architecture allows scaling and support of non-demanding standard missions as well as complex, prototype-like missions; however, it is clear that for complex use cases, the individual services may have to be adapted.

- **Service Oriented:** The internal and external interfaces have to be generalised as far as possible, making use of accepted European and world-wide standards, ECSS and CCSDS, respectively. It has to be considered that proprietary interfaces have to be reflected and supported in order to guarantee a safe transition phase.

- **Multi-Mission:** The systems shall be capable of being integrated into GSOC's multi-mission environment. This presumes that services are defined mission independent.

- **Flexible:** It shall be possible to alter system setups after mission start, like moving payload commanding from GSOC to an external partner, with adequate revalidation effort, simplifying the support for different mission phases and use cases.

- **Distributed:** It shall be possible to perform distinct operations tasks, including monitoring and control, from locations inside and outside the GSOC internal operations environment.

Distributed Operations

A distributed operations system setup not only includes forwarding of real-time or offline telemetry from GSOC to external partners or, vice versa, tele-commands from external partners to GSOC, but also remote access to various FOS application services: The customer may operate parts of the spacecraft, like the payload or certain experiments by himself, while relying on GSOC to safely perform standard BUS operations. Another example might be the access to a smart database interface, where the user dynamically defines which information he needs, in contrast to specifying all the needed telemetry parameters as part of a fixed interface definition. Operation engineers may be located in dedicated control rooms with direct access to the FOS, they may be located in GSOC offices, or they may be located at the external partner's site. Thus, not only the system itself is distributed, also the operations concept has to match the intended distribution of tasks. In consequence, this means that this opens the door for a wide variety of system setup possibilities (see Fig. 2):

- Users on-site at GSOC operations environment may access services using GSOC client applications or third party applications integrated into the GSOC environment.

- Users located in a different network section, like the GSOC office environment, or users at external sites may access services using web browsers or client applications provided by GSOC or by themselves. A wide variety of setup scenarios will be possible, ranging from single client software applications up to the integration of a full FOS environment.

The key factor is that all client applications use the same interface technology and defined service descriptions, simplifying the exchange of and access by the actual application software implementations. We decided to make use of the CCSDS MO standard, where GSOC already took over an active part [1]. Recent releases of standard documents, e.g. the M&C service description [10], and projects like French National Space Agency's (CNES) implementation of a control center based on CCSDS MO [4] makes us confident what this will be the ground segment design of the future.

Possible example application scenarios are:

- Support the satellite manufacturer by providing a central checkout system (CCS), either directly at customer's site or by remote access to a CCS facility at GSOC.
- The satellite manufacturer establishes the mission telemetry and tele-command database, flight operations procedures (FOP) and display definitions making use of GSOC tools installed at the customer's site. This validated operational data may then be re-used for operations at GSOC and also at possible external sites.
- The customer performs the payload monitoring and control, creating tele-commands or procedures and using the GSOC service for uploading, while GSOC operates the satellite BUS (which assumes that such an operations scenario is supported by the satellite itself, i.e. that it is possible to operate the payload independently from the BUS).
- The end customer accesses a GSOC archive interface to get offline telemetry.
- The end customer processes (parts) of the real-time and / or offline telemetry, and may also provide obtained products to his own customer.
- The end customer accesses flight dynamics services or mission planning services to support operations performed at his site.

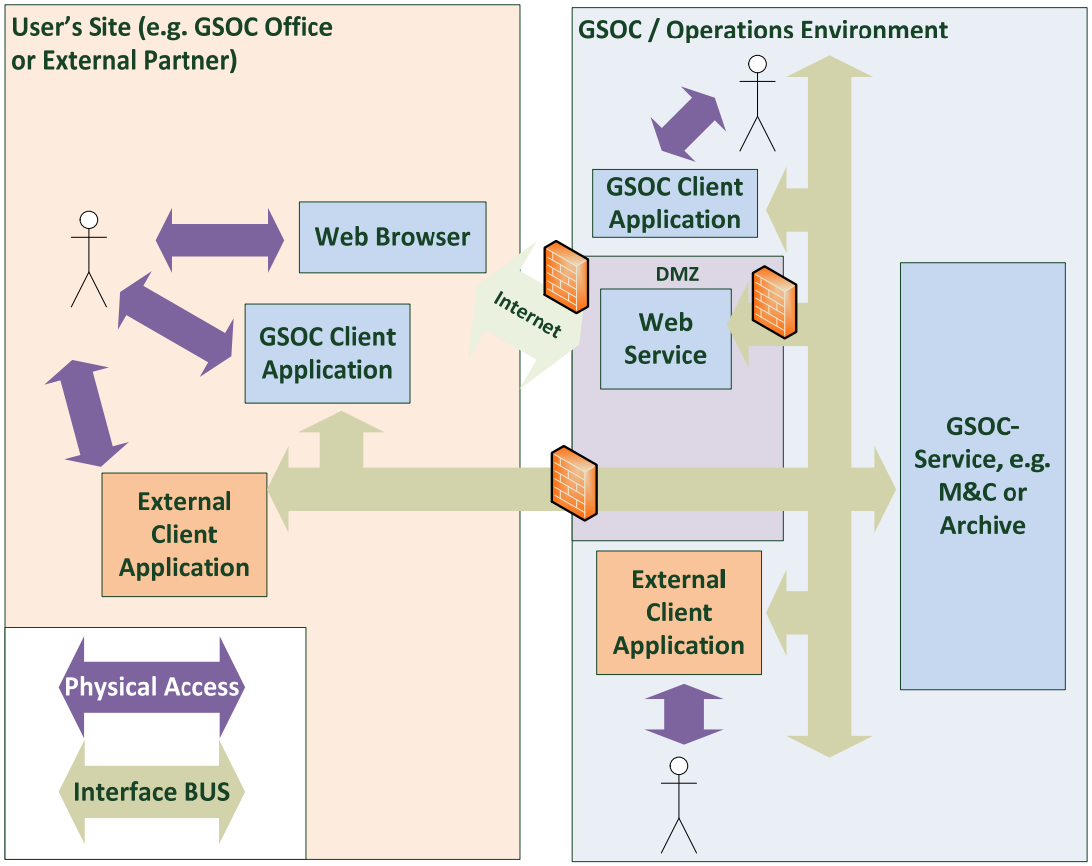

Fig.2. A combined view of various distributed operation applications. In case of external partner sites, the gray interface BUS may use e.g. the internet for connecting the user's site and the operations environment.

- The customer wants to provide his own backup FOS or “control room”.

TOWARDS A SMART AIT & FOS SERVICE STRUCTURE

The GSOC FOS design is driven by the network architecture, which is basically divided into a highly secure operations network and an office environment, with less strict security rules.

Typically for an operations center, the following use cases have to be covered:

- **Spacecraft AIT support:** Connections to simulators or engineering models at the spacecraft manufacturer’s site.
- **FOS implementation and system validation:** Connections to local simulators and simulators or engineering models at the spacecraft manufacturer’s site are needed for flight procedure creation and validation. Simulations for flight operations team training purposes have to be supported.
- **Launch and early orbit phase (LEOP) and commissioning phase support:** The system is used to perform operations with the spacecraft in orbit. Typically, during this phase parts of the FOS that cannot be tested without the spacecraft in orbit have to be validated. This kind of operations has a large demand on the size of facilities, due to the number of involved people. Switching from LEOP to commissioning phase typically means that the satellite team is reduced – however, remote access to the FOS by the spacecraft manufacturer’s team may be needed for certain planned or unplanned activities.
- **Routine Operations:** The routine flight support team may have other needs w.r.t. to the FOS, e.g. deeper involvement of mission planning and the cutomer(s). Also payload operators at external sites might be integrated. Needs for data distribution and information to a distributed ground segment arise.

Features and Limitations of the actual Setup

From the viewpoint of distributed operations, the actual FOS mission setups at GSOC already support distributed operations with several functionalities:

- Processed real-time telemetry can be forwarded unidirectional to external engineers using the telemetry client software Satmon@Home (developed by Heavens-Above for GSOC, [3])
- Ground and spacecraft activities may be initiated by using GSOC’s internet portal Opsweb, providing, amongst other features, online anomaly reporting, recommendations for activities and documentation. However, this is a manual process where in the end ground personnel has to ingest commands into the system: The operator picks up the prepared flight procedures, loads and uplinks them via the MCS.
- Import of telemetry from an external control center is performed in the technical demonstration payloads (TDP) project, demonstrating LEO to GEO optical communications links [5].
- With the European Data Relay System (EDRS), which provides a high-speed communication service between ground stations and LEO satellites, GSOC performs commanding of a payload with a remote partner control center [2].

On the other hand, though supporting mission operations in a reliable, safe and secure way, the GSOC FOS setup has limitations:

- The individual applications are, in the end, standalone applications which are integrated into a heterogeneous environment using various kinds of logical and technical interface technologies
- Data and archives are distributed over various networks and have to be synchronized; data exchange is mostly file based with limited performance.
- External access to offline historic telemetry data is limited. E.g. for handling spacecraft anomalies during commission and nominal operations phases, a simple and performant interface for the spacecraft support team, not on site anymore, is needed.

The challenge is to design a distributed FOS, that is homogenous from both the technical as well as the user viewpoint, and which supports the key requirements modularity, service-oriented, scalability and flexibility and maintains the mission adequate security level.

The Vision: A distributed, service oriented Architecture

Fig. 3 shows the vision of a distributed, service oriented architecture, supporting satellite AIT, ground segment validation as well as LEOP, commissioning and routine phase operations. At all sites, wherever test or routine operations are performed, it is ensured that the same validated client software uses the same validated server applications – reducing the overall validation effort. Some possible use-cases are:

- The satellite manufacturer may use his own or GSOC client tools (e.g. for display, procedure creation and execution, flight dynamics or software upload purposes) local at his site to develop and validate flight procedures and M&C databases. The services are located at GSOC.
- A routine operations setup where the flight operations engineers may monitor and control the system from various locations. A splitting of tasks, e.g. performing payload monitoring and operations at a customer's site, is possible. Services like mission planning or flight dynamics may be included.
- Support of spacecraft anomalies: Even if routine operations is done only at one location, spacecraft anomalies often need to be examined by a satellite manufacturer's support team. At any time, the FOS may be flexibly configured such that appropriate clients may be installed externally.
- Operations hand-over from a LEOP/commissioning control center to a distinct routine control center: If both control center's architecture is based on the CCSDS-MO MAL standard, the need for doubling operations software will be reduced; the customer may decide which building blocks or services of the FOS shall be re-used – without changes.
- EGS-CC ready: as soon as the new European command kernel EGS-CC supporting MO is available, the MOS GECCOS/SCOS3.1 may be replaced.

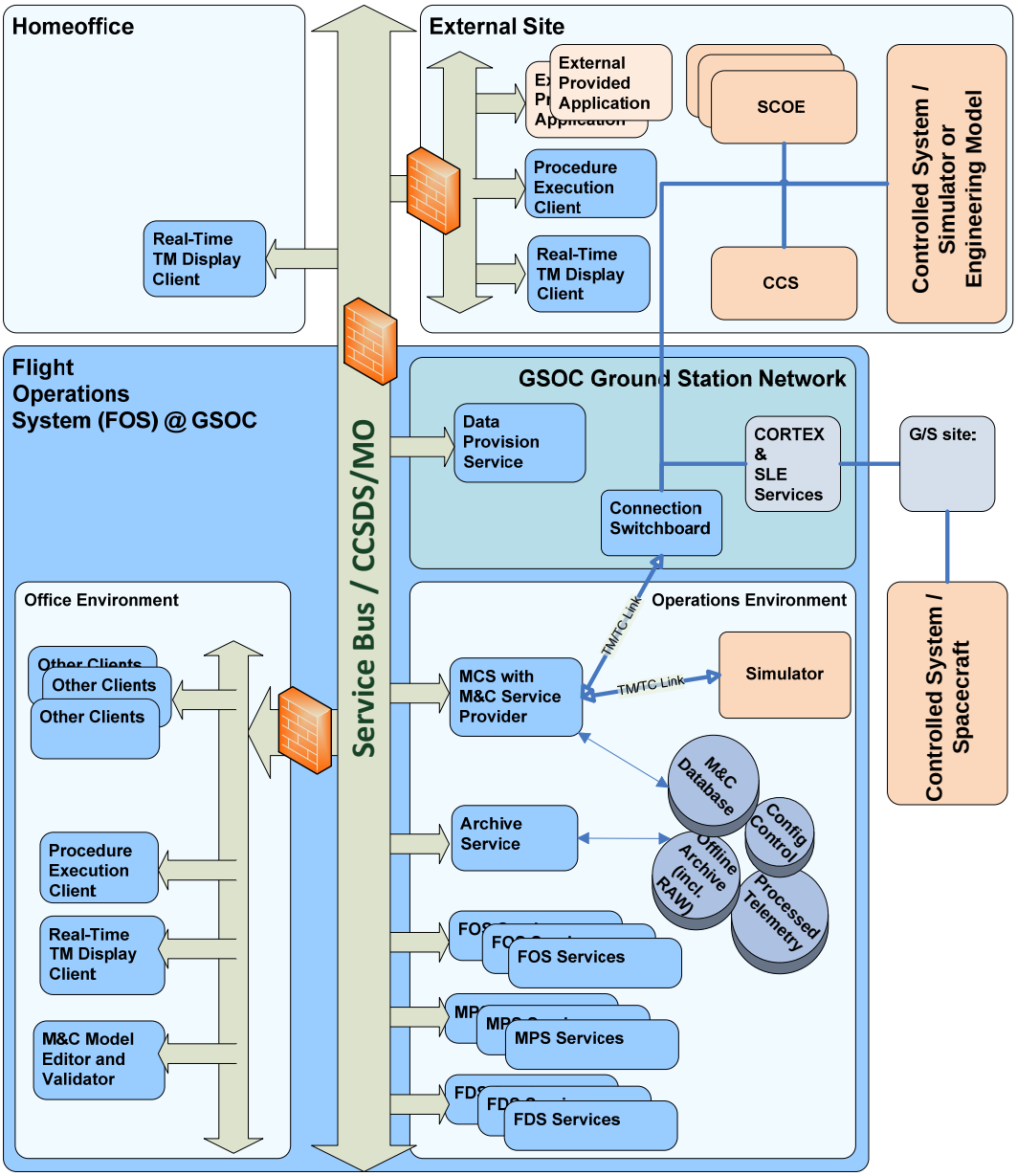

Fig.3. Sketch of a distributed, CCSDS-MO based FOS. The setup may be used for AIT support as well as LEOP, commissioning and nominal operations.

The basis of this service oriented architecture is the CCSDS MO Services standard, making use of the Message Abstraction Layer (MAL). This makes need for a way to traverse network boundaries, which typically are secured by one or more firewalls, in a defined and controlled way. For security and safety reason users will be allowed to receive only a subset of information, and to perform only a defined subset of activities. In consequence, there is the need of filtering at the interface boundaries, including adequate logging on different levels.

Backbone of such a homogenous distributed FOS is a message bus architecture that interconnects all services and their users. Exchanged messages follow the format and interaction patterns provided by the Messaging Abstraction Layer

MAL set forth in CCSDS standards. Besides being standardized and thus being readily usable by external partners without implementing proprietary interfaces, building upon the MAL also has several advantages for internal use:

- **Technology independency:** The message exchange bus system is merely a conceptual bus that in fact maps to many different technologies. Depending on context some technologies are more suitable than others. However, service providers and users should not care about these details. They simply use the bus and can rely on message transport using the most appropriate technology. For example, an internal message transport system is realized using a broker protocol, but external connections are provided using plain TCP/IP. A bridge connects these technologies transparently for any service provider and user. By decoupling the actual technology from the higher-level interfaces to the bus (i.e. the MAL messages and interaction patterns), boundaries to the in-house infrastructure group are well defined. This enables them to provide the bus technologies as infrastructure and allows their continuous improvement without making continuous adaptations to the service provider and user applications necessary.

- **Security:** Because all service data transport is mediated by MAL messages, their particular message structure can be exploited when implementing security. Although CCSDS MO does not prescribe any particular security model, it enables secure operations by a number of means:

    - Transport-level security can be employed for confidentiality. This is completely transparent to the MAL because securing the communication channels happens on the lower technology level. Broker-based systems typically allow secure communications out of the box.

    - Authentication Id: Each MAL message has to be equipped with an authentication field. This field may provide a signature, ensuring information integrity and authorisation, e.g. by issuing public-private key-pairs to providers and consumers.

    - Filtering based on any other MAL message header field: Authorisation of users can be enforced by filtering messages based on a number of message header fields, e.g. the domain field or session type field. Filters can be adjusted in a fine granularity down to the level of single operation invocations. However, by ensuring a proper domain tree for all missions configuration of the filters can be kept simple.

- **Well-defined service interfaces:** On top of technology independency that is made possible by the MAL-defined messages and interaction patterns, users and providers generally do not need to be aware of the concrete messages exchanged nor their exchange patterns. They simply want to invoke a certain operation of some service, possibly with parameters, and might expect some result now or later. Therefore the MAL also provides a language that enables such a service description. This description can be represented as well-defined XML making the scope and boundaries of services unambiguous. Furthermore, concrete technical interfaces and documentation can be derived automatically from this description. All standardized MO services like M&C come with such a service description and user services are typically also described in the same way

- **Self-organising, robust FOS:** Up to now, setting up interfaces between all operations systems is a tedious and error-prone manual process. Moreover, different mission phases might require a different set of interfaces, e.g. connections to the flight model during AIT are different than during routine operations. Thus, constant manual changes to the interfaces are necessary. By using MO, service providers announce their availability to a central directory, which is used by consumers to look them up. Apart from the bus and directory infrastructure no more manually created interfaces are needed. Even failover to redundant systems or short-term increase of demand can be handled this way.

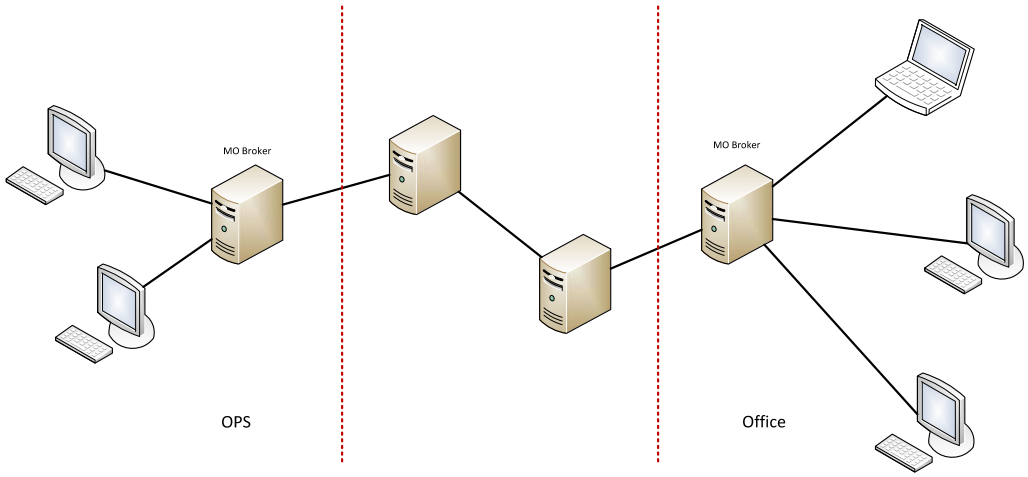

Fig.4. Connecting MO services and clients regardless of network boundaries and firewalls (dashed lines).

Implementation of MO services at GSOC is at the prototyping stage currently. The scope of implementation is a subset of M&C services and an in-house automation service for providing time-tagged execution of commands, for which no standardized service exists yet. In a first working setup an M&C provider (which will be a bridge to GECCOS later) has been connected to our combined procedure preparation and execution tool on the same network using a message broker technology. Any point-to-point technology like TCP/IP was ruled out due to our network setup and security implications (Fig. 4). The broker technology currently in use is message queue telemetry transport (MQTT, [11]), which is a lightweight protocol with origins in the Internet-of-things community. Service providers and consumers connect to a central broker instead of directly to each other as would be the case with TCP/IP. As already mentioned, this is transparent from the application point-of-view. In fact, for testing purposes the transport layer was exchanged with several other technologies without changing the applications. It remains to be seen which concrete message broker protocol best fulfils our requirements. Some of the faced challenges:

- **Network topology:** The strict separation between operational, office and external networks was the driver for a broker-based bus architecture. Brokers are deployed in each network with a defined interface boundary between them. Messages can be transported across the boundaries using broker federation or broker bridging – depending on the concrete broker implementation. Currently RabbitMQ and mosquitto are evaluated which both provide capability of interconnecting several brokers.

- **Security:** Due to very few interfaces between the networks the attack surface is kept to a minimum and connections can be severed quickly if malicious network traffic (e.g. a DDoS attack) is detected. All services vital for a mission are deployed in the operational network and remain functional while some convenient, but not mission-critical consumer applications might be interrupted. Most message broker implementations provide the capability of message interception and filtering, thus allowing implementation of security measures like authorization centrally on the broker instead of in each service provider. Messages can also be logged for real-time or offline analysis, providing another level of accountability.

- **Performance:** In contrast to direct connections performance of a broker-based system can be worse. No dedicated performance tests have been carried out so far but no performance problems could be observed during testing either. Should performance issues arise, the situation can be mitigated by operating the broker as a cluster. Cluster operation is supported by many broker implementations, thus reducing load on each single cluster node. The typical broker implementation also provides extensive instrumentation that can be connected to already

existing monitoring tools. On the other hand, broker-based solutions can provide a native implementation of MAL's Publish-Subscribe pattern, reducing strain on the service provider. Some performance problems are completely independent of transport technology and even exist in a traditional non-service oriented system, but are much more difficult to tackle there: Service providers can be proxied if they cannot keep up with consumer requests (this would regularly be the case if the service provider is located in space). More service instances can be spun up on demand, e.g. if there is increased demand during LEOP in contrast to routine operations.

ACTUAL STATUS

As a prerequisite, we put the prototype FOS as near as possible to GSOC's operational environment. The new system will be set up in GSOC's "Operations LAN", which is decoupled via various firewalls from the office environment or from demilitarized zones (DMZ) and thus from external access. Besides network separation, this also means that we are working inside the productive environment and have to respect all operational rules including GSOC's standard security mechanisms from user login to access controlled doors.The prototype will, in its final setup, provide a complete FOS starting from clients and ending with a connection to a simulator or an engineering model of an actual satellite mission.

Already performed milestones are:

- Decoupling the GUI from GECCOS/SCOS 3.1 to allow a complete separation of the user interface from the underlying SCOS kernel; including support for flight procedures and automation. This will later on ease the integration of the EGS-CC kernel.
- Setup of a realistic test environment (virtual machines) into GSOC operations network, including user setup.
- Prototype of a messaging bus in a single network for connecting CCSDS/MO services via MAL.
- Realization of a M&C service provider located on the external interface layer of the GECCOS/SCOS M&C system
- Start of integration effort EGS-CC / test setups

In the next steps, our goals will be:

- Finalizing the messaging setup for further distribution of M&C services also into distinct network, e.g. the GSOC office environment and an external partner
- We will have to reflect that not all CCSDS MO/MAL services are fully defined yet.
- Demonstrating the feasibility of direct real-time commanding of an engineering model using clients outside GSOC, located at the site of an external partner. With this setup, besides AIT, also operational scenarios like parallel commanding from different sites can be tested.
- In one of the final steps, the GSOC GECCOS / SCOS3.1. kernel will be exchanged by EGS-CC using its MO adapter. This will test the modularity and flexibility as well as verify the interplay of GSOC proprietary applications with the EGS-CC. As a prerequisite, the used TM/TC database has to be converted into EGS-CC's M&C model. In consequence, we want to confirm the feasibility of exchanging spacecraft data, like procedures, M&C models and display definitions from the AIT site to the operational world.

REFERENCES

[1] S.A. Gärtner, J. H. Hartung, M.Wendler, “Implementation of CCSDS Mission Operations Services at the German Space Operations Center”., American Institute of Aeronautics and Astronautics. SpaceOps, May 2014

[2] T. Beck, M. Schmidhuber, J.C. Scharringhausen, "Automation of Complex Operational Scenarios - Providing 24/7 Inter-Satellite Links with EDRS", American Institute of Aeronautics and Astronautics. SpaceOps, May 2016

[3] Heavens-Above, München, http://www.heavens-above.com/

[4] N. Champsavoir and J.-M. Georger, “ISIS MCS: A High-Performance Mission Control System
Based On CCSDS Mission Operations Standards”, American Institute of Aeronautics and Astronautics. SpaceOps, May 2014

[5] E. Benzi, I. Shurmer, N. Policella, D. Troendle, M. Lutzer, Sv. Kuhlmann, M. James, "Optical Inter-Satellite Communication: the Alphasat and Sentinel-1A in-orbit experience", American Institute of Aeronautics and Astronautics. SpaceOps, May 2016

[6] C. Stangl, B. Lotko, M.P. Geyer, M. Oswald, A. Braun; “GECCOS - the new Monitoring and Control System at DLR-GSOC for Space Operations, based on SCOS-2000”, American Institute of Aeronautics and Astronautics. SpaceOps, May 2016

[7] A. Walsh, J.M.Carranza, W. Bothmer, P.-Y. Schmerber, J. Rueting, P. Parmentier, P. Chiroli, M.-C.Charmeau, M. Geyer, "The European Ground Systems - Common Core (EGS-CC) Initiative", , American Institute of Aeronautics and Astronautics. SpaceOps, May 2012

[8] CCSDS, “Mission Operations Services Concept,” Dec. 2010, Informational Report, Green Book, 520.0-G-3, http://public.ccsds.org/publications/archive/520x0g3.pdf

[9] CCSDS, “Mission Operations Message Abstraction Layer,” March 2013, Recommended Standard, Blue Book, 521.0-B-2, http://public.ccsds.org/publications/archive/521x0b2.pdf

[10] CCSDS, “Mission Operations Monitor & Control Services,” Sept. 2016, Draft Recommended Standard, Red Book, 522.1-R-3

[11] MQTT Version 3.1.1 Plus Errata 01. Edited by Andrew Banks and Rahul Gupta. 10 December 2015. OASIS Standard Incorporating Approved Errata 01. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/errata01/os/mqtt-v3.1.1-errata01-os-complete.html. Latest version: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html