111122

N95-17537

3/942

p. 7

SCOSII: ESA'S NEW GENERATION OF MISSION CONTROL SYSTEMS THE USER'S PERSPECTIVE

P.Kaufeler and M.Pecchioli, Mission Operations Department (MOD), European Space Operations Centre (ESOC), Darmstadt, Germany
I.Shurmer,VEGA Group PLC, Harpenden, UK, (MOD, ESOC, Darmstadt, Germany)

Abstract - In 1974 ESOC decided to develop a reusable Mission Control System infrastructure for ESA's missions operated under its responsibility. This triggered a long and successful product development line, which started with the Multi Mission Support System (MSSS) which entered in service in 1977 and is still being used today by the MARECS and ECS missions; it was followed in 1989 by a second generation of systems known as SCOS-I, which was/is used by the Hipparcos, ERS-1 and EURECA missions and will continue to support all future ESOC controlled missions until approximately 1995. In the meantime the increasing complexity of future missions together with the emergence of new hardware and software technologies have led ESOC to go for the development of a third generation of control systems, SCOSII, which will support their future missions up to at least the middle of the next decade. The objective of the paper is to present the characteristics of the SCOSII system from the perspective of the mission control team; i.e. it will concentrate on the improvements and advances in the performance, functionality and work efficiency of the system.

1. INTRODUCTION

The concepts and functionality of the Mission Control Systems (MCS) which are currently in use in ESOC, i.e. MSSS and SCOS-I, are mainly originating from the mission control requirements of the 1970's which were based on the hardwired spacecraft technology which was the standard at this time. The arrival of a new generation of more complex spacecraft with significant amount of on-board software and increased on-board autonomy, such as EURECA or ERS1, placed much more demanding requirements in terms of functionality and performance on the MCS which, although they could be accommodated (sometimes requiring development of mission specific adds-on), revealed the limits of these systems. Therefore the decision for the development of a new generation of MCS, SCOSII, was taken, with the following main objectives:

- reduce mission adaptation/maintenance costs,
- improve efficiency of mission preparation, execution and evaluation tasks,
- increase operational quality and reliability,
- have a life time of at least 10 years,
- cope with a wide range of different mission type/size/complexity.

which led to the following major design requirements:

- SCOSII must be a full scope generic system.
- It must be a modular and open system, being adaptable and expandable in size, performance and functionality.
- It must operate in a basic hardware and software environment that is vendor independent.
- It must be based on state-of-the-art software technology.
- It must be compatible with the new standards in the space domain such as in particular the CCSDS and related ESA standards for telemetry and telecommand packets, and the standards and guidelines of the ESA Committee for Operations and EGSE Standards (COES).

2. SYSTEM CHARACTERISTICS AND CONCEPTS

The SCOSII system has been conceived as a generic infrastructure platform, providing an exhaustive set of standard functionality constituting the basis for the development of mission dedicated MCSs. As such, a particular instance of a SCOSII based MCS will not offer multi-mission support, but will be able to cope with multi-satellite missions, thus supporting simultaneous control of several satellites of the same family.

2.1 Architecture

The high flexibility and performance requirements placed on SCOSII led to the choice of a decentralized architecture, consisting of a network of Unix workstations in a *'Client-Server'* configuration. Each operational user will interface to the system through a dedicated workstation providing local processing power to cope with the user-interface processing load, and local storage for e.g. hosting of the most recent historical data, while a set of system level services (e.g. interfacing to the ground stations) ensuring overall coordination will be provided by central server processors. The use of such a distributed system will allow the computing power to be tuned to the demand of a particular mission and will also offer advantages in terms of system availability and failure tolerance. A more detailed description of the architecture of SCOSII can be found in References [2] and [5].

2.2 Overview of Functionality & Utilisation

SCOSII is intended to cover the following functions and services:

- ***Mission Planning***, including acceptance, checking and pre-processing of various types of planning requests, generation of a conflict-free *'Plan'*, and derivation of an executable *'Schedule'*.
- ***Monitoring & Commanding (M&C)***, of the spacecraft, the mission support services provided by the ground network (e.g. telemetry and telecommand services of the TT&C stations) and SCOSII itself (i.e. control of user configurable functions). This means that e.g. the same generic M&C functions (e.g. status monitoring, commanding, procedures execution) can be used to handle the spacecraft, ground station services and on-line SCOSII configuration.
- ***Historical Data & Performance Evaluation***, consisting of the storage of all mission data in an on-line manner, the ability to access these data for direct visualisation and/or subsequent processing using powerful data analysis and presentation tools, and the production of corresponding reports.
- ***Mission Database Handling***, consisting of the generation and

maintenance of all the static mission data used, to configure the system for a given mission (e.g. user privileges, display lay-out), and to define the characteristics of the mission (e.g. TMTC processing data, operations procedures, etc...).
-*On-board Software Maintenance*, consisting of the tools to monitor, and modify the content of the on-board memories (i.e. memory images).
- *System Level Tools & Services*, such as state-of-the-art Human Computer Interface (HCI) techniques, user access control mechanism, advanced help facility, etc...

The wide range of functionality provided by the system, and its flexibility and adaptability, will allow SCOSII to be tailored to cover, for a given mission, different *'Roles'*, each being carried out by a specific instance of a SCOSII system. This will of course include its main role of *'Prime'* MCS which will incorporate the full set of functionality required to support the mission, but SCOSII may also be used as *'Mini-backup'* spacecraft M&C system to be located at e.g. a TT&C ground station. Furthermore, the fact that SCOSII is being designed in accordance with the standards and guidelines of COES, will ensure not only its full compatibility with checkout systems, but would allow SCOSII to be used, with minor adaptations, as a checkout system as such.

Having outlined the functionality and roles of the system, we will now address the various user scenarios which SCOSII will have to support. Here again, SCOSII constitutes a major step forward with respect to its predecessors which were only providing very restrictive and rigid centralised user access, in that it will also support various types of remote access scenarios as described below and illustrated in figure 1.

- ***Office based users***, for mission preparation and/or evaluation activities.
- ***Home-based users***, for on-call contingency support.
- ***Engineering support users***, such as spacecraft manufacturer, for anomaly investigation, mission evaluation.
- ***User Operations Control Centres*** (USOC), for the control of given payload(s) on a spacecraft.

2.3 Configurability

Since SCOSII will constitute the basic MCS kernel for a wide range of missions of different type and complexity, the system will have to be highly configurable. One important aspect in this context, is the capability of SCOSII to be descoped, adapting its functionality and hardware to the needs of the mission. For a simple mission, a mini-system running on a single SUN workstation, could be used. Moreover, its portability will allow such a mini-system to run on a PC.

Another issue related to configurability is that the system must be, as much as possible, data-base driven, maximizing the tailoring capabilities and minimizing the need for software modifications. For predecessor systems, this approach was limited to the spacecraft TM/TC processing characteristics, which were fully defined in the spacecraft characteristics database. For SCOSII this concept has been expanded to all functional subsystems, including

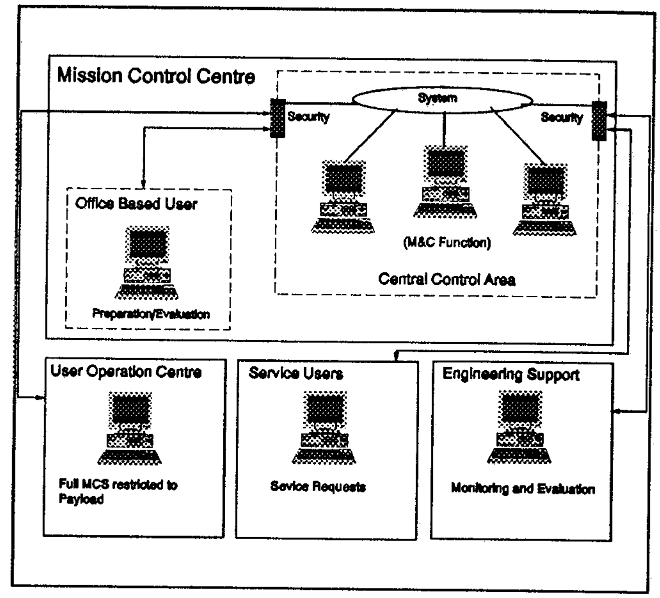

Figure 1: SCOSII User Scenario

data driving the system configuration, thereby providing the user with the capability of defining through the Mission Database the haracteristics of major elements of the system such as:

- HCI layout (e.g. layout of input forms or of displays templates),
- defaults for most of the functions (e.g. which packets are to undergo which types of checks),
- definition of standard named sets of user privileges.

The SCOSII system will therefore consist of a set of generic functions plus a generic default configuration, which can be modified by the user to suit the needs of his mission.

2.4 Performance

As SCOSII is intended to be the basis for MCSs for at least the next ten years, very ambitious performance goals have been adopted. These include concurrent real-time telemetry and telecommand rates of 2Mbps and 4Kbps respectively, display update rates exceeding 10 per second, very short response times to user requests - e.g. from 5 sec for retrieval of data not older than a few weeks to 30 sec for data being several years old -, the above requirements being applicable to utilisation scenario involving up to 50 workstations used simultaneously.

2.5 System Level Tools

In support of its main functions as described above, SCOSII will provide a set of very powerful system level services and tools, the most significant of which are presented below.

2.5.1 Modelling Tool

Previous control systems were based on a low-level view of the spacecraft in that they only considered its telemetry and telecommand components, and thus did not include any information about their link to the higher level components of the spacecraft such as the devices/units, subsystems, etc..., and their interrelationships. This approach was sufficient to handle relatively simple missions, but was not adequate for introducing more advanced functionality and user interfaces which require a more structured and intuitive view of the mission/spacecraft.

A fundamentally different approach was followed for SCOSII. In the SCOSII database the mission will be described as a hierarchical structure of components of operational significance. This is achieved by defining a decomposition following the object-oriented *'whole-part'* relationship, starting with the mission as the highest level component, down to the devices/units hosting the individual measurements and command items at the lowest level.

In addition to this decomposition into what are called *'System Elements'* in the SCOSII jargon, it will be possible to associate with them synthetic information, called *'Operational Modes'* and *'Roles'*. The former represents particular states of operational significance as derived from the state of their constituted parts, while the latter corresponds to their function(s) within their respective mission domain. This will provide a first step towards an advanced modelling capability; initially modelling will be restricted to data routing, power control and redundancy but this will be further extended in future releases of the system to include the full set of standard functions and behaviours of the typical mission components.

Moreover, SCOSII will also provide a library of *'System Elements'* which could be used as building blocks. In order to define e.g. the battery 1 component of mission X, the user would chose the standard battery building block in the SCOSII library; he would, if required, modify it to correspond to the characteristics of the batteries of mission X by specifying its difference to the standard SCOSII battery, and instantiate it to become battery 1 by specifying the links to its constituent telemetry and telecommand items. These modelling capabilities which are illustrated in Figure 2 below and further expanded in Reference [5], will provide significant improvements in the following domains:

- ***Mission Database Definition:*** increased efficiency and quality/consistency, by reducing the information to be specified by the user to a strict minimum and by providing him with a more intuitive view of its mission.
- ***System Configurability and Controllability***: by allowing the user/system to exercise this at mission component level (for navigation through mimic display, to disable functional checks for only a particular mission component, to allocate/restrict functional privileges to e.g. a particular spacecraft subsystem, etc...).
- ***Mission Execution:*** by making use of the modelling data (in particular the *'Roles'*) to predict the status of the mission, thereby supporting the mission planning and commanding functions in assessing the effect of future commands (e.g. to ascertain their safety/feasibility), and the monitoring function by generating the expected mission status as reference for comparison against the status obtained from telemetry. This *'Prediction'* function is a new feature, making use of the *'Mission Model'* to obtain the best estimate of the mission status at any time in the future, based on an initial state and on the knowledge of any planned activities and any foreseen on-board events and actions.

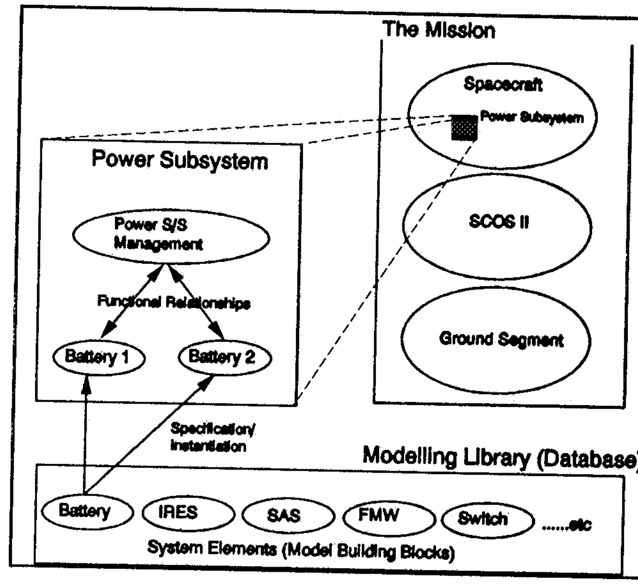

Figure 2: SCOSII Modelling

2.5.2 Operations Language

To allow an efficient definition and maintenance of the mission specific knowledge in the Mission Database, a dedicated SCOSII Operations Language (OL) is required. The OL has been designed to provide users without software design expertise, with a set of mini languages offering the necessary expressive capability to define the knowledge for the more advanced SCOSII functions, such as:

- ***Procedural knowledge***, for the presentation of, navigation through and automatic/semi-automatic execution of procedures to e.g. control the spacecraft, ground station services and SCOSII configuration,
- ***Events & Actions***, to identify from the incoming data, user defined events to be logged and the corresponding actions to be initiated by the system (e.g. an event could be a particular spacecraft anomaly which would initiate a specific set of recovery and diagnosis actions),
- ***Selection Strategies***, to provide the various data selection capabilities that will be required by the user and/or system in support of the different activities/applications (e.g. selection strategies could be applied to restrict a particular function to a subset of the data it would normally be applied to).

Further details about the SCOSII OL can be found in Reference [4].

2.5.3 Mission Database Test Function

This is another new functionality, which will provide an on-line mission database checking capability, using as data sources either real-time or historical telemetry, or data generated by the *'Mission Model'* being driven by a predefined sequence of commands. This local test function will allow to significantly reduce the turn around time for database changes, and to alleviate the need for the lengthy and resource-intensive validation using an external software simulator.

2.6 Human Computer Interface (HCI)

The SCOS II HCI will provide users of all levels of experience, with an intuitive, but reliable and robust interface. The SCOSII HCI will be based on WIMP (Windows, Icon, Mouse and Pull-down menus) technology. SCOS II will support all the traditional display types (e.g. alphanumeric, graphic and mimics displays), however, the users will be given tools which will allow them to combine these different data display techniques to display data in a more flexible and efficient manner. Due to the increase in the diversity and versatility of the HCI with respect to previous systems, particular attention has been paid to the specification of general guidelines concerning display and data representation techniques in order to provide the user with a consistent HCI across all applications.

3. MISSION DATABASE

The scope of the SCOS II Mission Database is much wider than that of the earlier systems, which generally concentrated upon the data required by the Monitoring and Control functions. In addition to the latter, a SCOS II database will contain, e.g. the mission model data, the mission planning/scheduling data, the on-board software memory images, the operations procedures and the Spacecraft Users Manual (SUM), and will also include the system set-up and configuration data (e.g. definition of user privileges).

3.1 Mission Database Structure

The Mission Database will consist of a hierarchical collection of database parts, each with a unique identifier and version number, arranged in a user defined structure (Figure 3). The higher level parts are used purely for organisational purposes, the lowest level parts contain the data and constitute the lowest level entities submitted to version control. For a given mission, the user will have some flexibility of configuring the structure to its particular needs.

3.2 Database Management

There will be three types of Mission Database.

- ***The Operational Database***: A database which is, or has been judged so in the past, capable of supporting real-time operations. SCOS II will support a number of Operational Database versions.
- ***The Active Database***: The Operational Database which currently supports real-time operations; any of the Operational Databases may be selected as the Active Database.
- ***The Draft Database***: A database used as an intermediate step to constructing a new Operational Database. There will be only one Draft Database.

It can be imagined that all the databases are kept within a *'Database Area'* and accessed via the users from a *'Working Area'*. The *'Working Area'* contains a number of user accounts, i.e. *'User orking Areas'*, which will allow multiple user database maintenance. Special mechanisms will be provided in order to ensure this multiple user maintenance is done in an orderly manner; e.g. each user working area will be completely isolated and the system will prevent several users from being able to work on the same database part simultaneously. The database manager will be able to select modified database parts and to integrate them

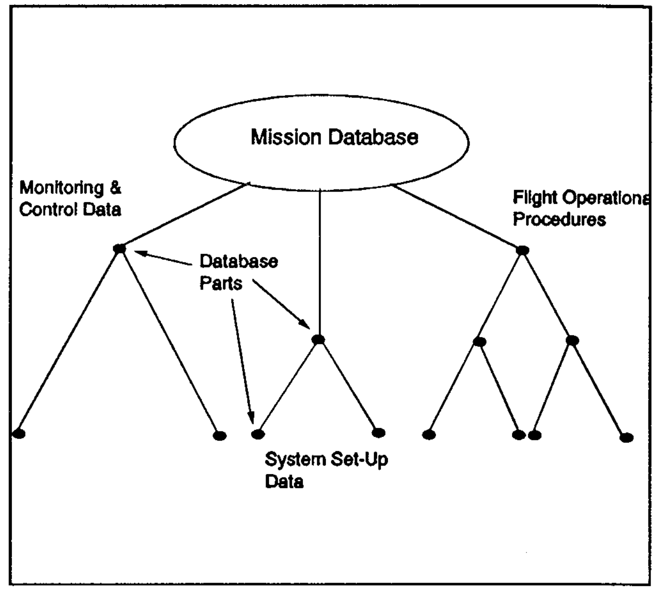

Figure 3: Mission Database Structure

back into an Operational Database, either directly (for on-line changes) or via the draft database (for changes of a higher magnitude). Subsequently, this database can be selected as the Active Database.

Version Control functionality will automatically maintain the version of the Operational or Draft Databases and their constituent parts. In addition Change Control functionality will permit exhaustive recording of all database changes at item level and at the higher levels of the database hierarchy. The SCOSII database management concept, as described above, is illustrated in Figure 4, below.

3.3 Database Maintenance

Mission Databases are mainly constructed from input data that are provided from spacecraft/payload(s) manufacturer(s) or from checkout. Since the data volume may be extremely high (typically several thousands of parameters, just for telemetry) these data are to be provided in an electronic form. SCOSII will be able to import these source databases in various electronic formats (e.g. ORACLE, ASCII), to integrate the contained data items into the SCOSII internal database, and to subsequently handle new versions of the source database (e.g. functions to compare a new source version with previous ones or SCOSII versions).

In addition to the acquisition of the source database, SCOSII will provide the editing capabilities required to handle the data that have been acquired from the source database (for this dedicated functions will be provided to facilitate large scale editing) and to subsequently maintain the data. The data maintenance functions will of course include exhaustive but flexible data consistency checking functionality. Consistency checking will be performed at all levels (e.g. data item, database part and database), however, the user will be able to switch the checking off, an essential feature for the preparation phase, when inconsistencies cannot be avoided.

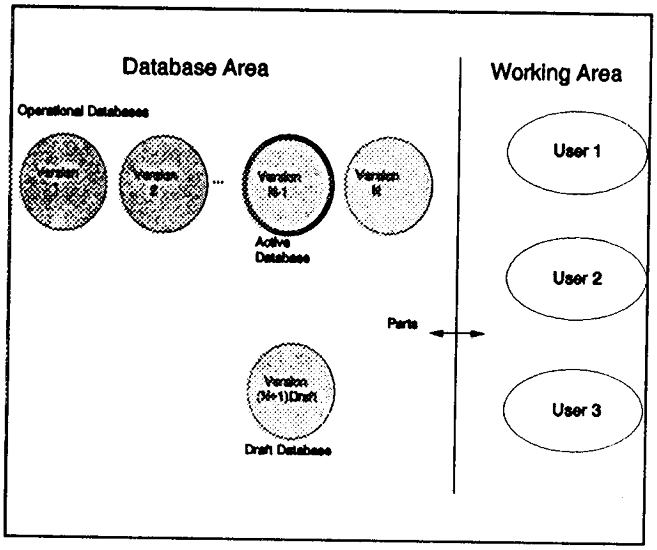

Figure 4: Database Management Environment

4. MONITORING

The following monitoring tools will be provided.

4.1 Monitoring Parameters

Unlike previous systems, there will be several potential sources of monitoring parameters in addition to those that come from the spacecraft, e.g. SCOS II parameters and Ground Segment parameters. Regardless of source, all parameters will be processed by SCOS II in the same manner.

SCOSII will be far more flexible and versatile than previous systems. The users will be given the facilities to view the monitoring parameters in a number of different ways called *'Representations'*. The user will be able to select, in real-time, which *'Representation'* is to be displayed. In particular, SCOS II will support:

- ***The Raw representation**:* the uncalibrated view of the parameter value.
- ***The Engineering representation**:* the calibrated view of the parameter value.
- ***The Functional representation**:* obtained by applying a function to the parameter eg. derivative, integral, mean, max.
- ***The Status representation**:* returning the state of the currently applicable *'Checking'*, eg. nominal (see section 4.3 below).

In addition, SCOS II will support the display of values in different formats (e.g. raw representation in hex, decimal or binary) and the on-line conversion between engineering units.

4.2 Parameter Validity

SCOS will support the concept of monitoring parameter validity, since for a number of possible reasons, the latest parameter value could be meaningless or unreliable. The following factors influencing the validity have been identified.

- ***Power**:* the status of the power supply the parameter is dependent upon.
- ***Data Unit**:* the quality of the data unit within which the parameter was transported.
- ***Data Routing**:* the status of the transmission route taken by the data.
- ***Age**:* the age of the parameter value.
- ***Stability**:* the parameter value may be in a transient state due to commanding activity.
- ***Status**:* any other explicit criteria the user wishes to specify.

SCOS II will check all these factors when assessing a parameter's validity. The resultant validity state of a parameter will be automatically propagated throughout the system affecting other processing where relevant (e.g. synthetic parameters will also be flagged as invalid if they use an invalid parameter) and affecting how the data is displayed to the user.

The user will be able to gain real-time access to the results of each validity component check. Hence, the SCOSII user will be provided with significantly improved validity checking facilities and, when a parameter is flagged as invalid, considerably more information about the reason why.

4.3 Parameter Checking

The objective of checking is twofold. On one hand the system must be able to check whether the operator has not or is not going to place the mission elements under its responsibility (e.g. the spacecraft) in a non-nominal or unsafe state, on the other hand, the system must be able to detect whether these elements are behaving as expected. This led to the following categories of parameter checking being provided by SCOSII.

- ***Operational Status Checks**:* Monitor the on-board status which is required regardless of any commanding activities, to ensure that the spacecraft is left in the correct state after a series of operations.
- ***Operational Constraints Checks**:* Are of the same nature as Operational Status Checks, but stronger. They are rules which should never be violated operationally and as such, should never be disabled. They will contribute to *'Activity' 'Pre-Execution Validation'* (see section 5.3 below).
- ***Behaviour Checks**:* Are based upon the prediction of the on-board status, taking into account the effects of commanding and of predicted events. The checking performed is to ensure that any behaviour exhibited (e.g. change of state after a command) is as expected.

5. COMMANDING

An overview of the envisaged full SCOSII Commanding functionality is given in Figure 5.

5.1 Activities

In order to control the mission, a SCOSII user will be able to initiate the execution of *'Activities'*, where an *'Activity'* is either a Procedure (highest level), a Command Sequence (simplified procedure syntax), or a Command (lowest level). SCOSII treats each of these in exactly the same manner. Each can have execution pre-requisites, each can be monitored through its execution phase and each can be verified. Activities will be initiated manually, or automatically by the system. The long term aim of SCOSII is to have a fully automated Procedure execution capability.

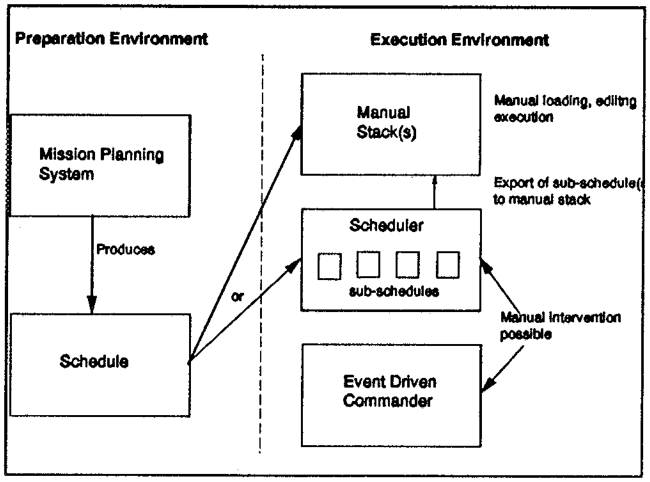

Figure 5: Commanding Overview

5.2 Preparation

This consists in the production of the schedule of activities corresponding to a given time increment, for later submission to the activity execution function. While this will be initially done manually, it will be carried-out, in later releases of the system, by a generic Mission Planning functionality, which will include:

- *Processing of Planning Requests*: This covers the acceptance, checking and pre-processing of planning requests received from external entities e.g. experimenters, external control centres, flight dynamics.
- *Planning/Scheduling Function*: This covers all activities required to generate a conflict-free *'Plan'* and its corresponding *'Schedule'* of activities from the pre-processed planning requests.

5.3 Activity Execution

It will be possible for the user to execute operational *'Activities'* by means of three facilities:

- *The Scheduler*: Pre-prepared *'Schedules'* will be imported from the preparation environment into the *'Scheduler'*. If necessary, the user will also be able to split this imported *'Schedule'* into a number of logical partitions called *'Sub-Schedules'*. Each *'Sub-Schedules'* of executable activities could then be assigned to a different user and/or to a different type of operations (e.g. one sub-schedule could be dedicated to Payload-X), thus delegating execution control. Nominally the *'Scheduler'* will manage the execution of activities automatically, taking into account execution pre-requisites and links between activities, prompting for manual input when required. However the user will always retain the capability of regaining, if required, control over the *'Scheduler'*.
- *The Manual Stack:* The traditional commanding facility, allowing the user to directly control the release of Activities will, of course, also be supported by SCOSII in order to provide the user with fully manual execution capability in the eventuality of critical and/or unplanned operations.
- *The Event Driven Commander:* This is a new SCOSII concept that will give the users the capability of setting up event-action relationships as Event Driven Commanding Routines (EDCRs). EDCRs can periodically monitor for the occurrence of an event that will trigger the execution of a specified set of activities, e.g. can be used for automatic closed loop reaction to on-board anomalies.

All executable activities will have pre-requisites which must be satisfied before they can be released from the SCOSII system. In SCOSII, these are called *'Pre-Execution Validation'* (PEV) checks. These will have three components:

- *Feasibility Checks*: Checking that all necessary resources are available, e.g a transmission route.
- *Safety Checks:* eg. Checking that Unit A if OFF before switching Unit B ON.
- *Dynamic Checks:* Checks which are not related to the activity in isolation, but to the external context of the execution of a specific instance of an activity, e.g. time window and interlock checks.

Activities will then be released by SCOSII when authorised by their respective PEV, based on a *'Release Strategy'* specified at preparation. SCOSII will support both manual and automated release strategy such as "initiate execution X minutes after event Y".

5.4 Activity Execution Monitoring

To enable the user to be aware of the transmission and execution status of any activity that has been released from SCOSII, dedicated verification checks will be performed. For command execution verification, the users will explicitly define verification criteria, using the Operations Language, in the Mission Database. Though, there will be the capability of doing the same for command sequences and procedures, the majority of their checks will be implicitly defined by the checks defined for each command they contain. SCOSII will support the explicit definition of simple or complex multi-staged verification criteria, the latter for those commands which are executed in a number of stages (eg. reception on-board, reception by application, execution stage 1, execution stage 2). For each identified verification stage, SCOSII will automatically compute a verification window based on expected execution times and/or user defined margins/delays.

6. CONTROL OF SCOSII SYSTEM

The M&C functionality will be controllable flexibly. This is particularly important in the case of contingency situation where the normal conditions of applicability of a function may not be valid any more; past systems have been rather rigid in this respect. During operations, the user will have the capability to control the way the functions are applied and to which data they are applied; e.g. one will be able to completely or partly disable parameter validity checks. Standard parameter checking, as described in 4.3 above, will be applicable to the status of the controlled functions, to ensure that they are not left in a non-nominal/undesirable state.

7. MISSION EVALUATION

Sophisticated tools will enable the users to access historical data and then view, analyze them and to produce reports. This functionality will be an integral part of SCOSII, and unlike on previous systems, will be available on-line. The following functions will be provided.

7.1 Historical Data Access

The user will be able to access data and if required to save them for later re-use, either for direct presentation using the standard displays used for real-time monitoring, or for submission to further processing (e.g. detailed analysis). Data access will be supported by a powerful syntax, allowing the user to define expressions, called *'Data Access Strategies'*, which he could save for later re-use, and capable of specifying:

- A time window or multiple time windows.
- References to events eg. 'entry into eclipse'
- Expressions eg.'when A123 > 35 degrees C'
- Data access criteria e.g. all AOCS telemetry

7.2 Viewing Historical Data

SCOSII will provide the user with two viewing modes, *'Replay as Live'* and *'Video Replay'*. *'Replay as Live'* will be dedicated to the technical analysis of the mission data, i.e. it will allow the user to replay historical data and to interact with them as if they were being generated in live, while *'Video Replay'* will be dedicated to operational investigation, i.e. it will allow a user to be confronted with the same data and workstation lay-out as at the time of reception of the data.

In both cases, the user will have complete control over the replay, controlling its start time, the number of workstations it appears on and its speed and direction (eg. fast forward, forward, pause, rewind, fast rewind etc...).

7.3 Historical Data Analysis

The users will be provided with a data analysis package which will have the following functionality at a minimum:

- ***Data Manipulation***, allowing the user to select a subset of the retrieved data for analysis.
- ***Mathematical functions***, e.g sin, cos, tan, log, differentiation, integration.
- ***Statistical Analysis functions***, e.g. mean, standard deviation.
- ***Graphical tools***, allowing the user to produce 2- and 3-D graphs, straight line fits, polynomial fits, bar charts, pie charts etc...

7.4 Report Generation

A great amount of effort is expended by Operational Teams producing reports, many of which are of a routine nature. Therefore SCOSII will, unlike on previous systems, include a report generation function allowing production of text documents in which mission history data can be incorporated. This function will also support an automatic report generation facilities; a user will be able to define report templates, e.g. definition of the contents and structure of a report, and use these to automatically produce reports of data for a user defined time period.

8. CONCLUSION

SCOSII is a major step forward with respect to its predecessor systems, which will put ESA at the forefront of the technology and meet its main goals of minimising mission costs and improving mission preparation efficiency and operational performance.

This is the first time that a systematic and thorough effort has been invested in defining user requirements for a generic infrastructure (as opposed to individual mission systems). This has involved a close cooperation between the users and the developers of the system, and has included exploratory prototyping (as well as technology prototyping).

Release 1 of SCOSII is at an advanced stage of implementation, a preliminary delivery being expected in November 1994. Broadly speaking Release 1 is covering the same range of functionality as the previous infrastructure, with inclusion of the Commanding function (not available in SCOS-I) and with enhanced functionality and more modern human computer interfaces. More advanced functionality will be added in Release 2 (1995-6) and Release 3 (1996-7), including Modelling, Mission Planning, Data Distribution and certain of the more advanced database features. Consolidation of Release 1 functionality will also take place in the later releases. Such an incremental implementation has been chosen in order to minimise technical and schedule risks to the first client missions of the system, HUYGENS, ARTEMIS,and ENVISAT to be launched in 1997-1998.

References

1 Packet Utilisation Standard, ref. ESA PSS-07-101, issue 1, May 1994, to be published.

2 N.Head & J.F.Kaufeler, Evolution of the Agency's Software Infrastructure for Spacecraft and Mission Control, ESA Bulletin no.67, August 1991.

3 M.Jones et al., SCOSII: ESA's New Generation of Mission Control System, ESA Bulletin no.75, August 1993.

4 A.Baldi et al., SCOSII OL: A Dedicated Language for Mission Operations, SpaceOps 94.

5 M.Jones, N.Head et al., SCOSII: ESA's New Generation Control System, SpaceOps 94.