The ADVANCED TECHNOLOGY OPERATIONS SYSTEM ATOS

J.-F. Kaufeler[1], H.A. Laue[1], K. Poulter[2], H. Smith[3]

Abstract

Mission control systems supporting new space missions face ever-increasing requirements in terms of functionality, performance, reliability and efficiency. Modern data processing technology is providing the means to meet these requirements in new systems under development.

During the past few years the European Space Operations Centre (ESOC) of the European Space Agency (ESA) has carried out a number of projects to demonstrate the feasibility of using advanced software technology, in particular, knowledge based systems, to support mission operations.

A number of advances which must be achieved before these techniques can be moved towards operational use in future missions, namely, integration of the applications into a single system framework and generalisation of the applications so that they are mission independent.

In order to achieve this goal, ESA has initiated the Advanced Technology Operations System (ATOS) programme, which will develop the infrastructure to support advanced software technology in mission operations, and provide applications modules to initially support: Mission Preparation, Mission Planning, Computer Assisted Operations and Advanced Training.

The first phase of the ATOS programme is tasked with the goal of designing and prototyping the necessary system infrastructure to support the rest of the programme. This paper presents the major components of the ATOS architecture.

This architecture relies on the concept of a Mission Information Base (MIB) as the repository for all information and knowledge which will be used by the advanced application modules in future mission control systems. The MIB is being designed to exploit the latest in database and knowledge representation technology in an open and distributed system.

In conclusion we present the technological and implementation challenges we expect to encounter, as well as the future plans and time scale of the project.

1. Mission Control Systems

1.1. Tasks of Mission Control Systems

The Mission Control System (MCS) is the interface between a spacecraft in orbit and the human user of the spacecraft on ground. It is essential to conduct a space mission and to fulfil the mission objective. These objectives are met by:

- mission planning
- monitoring and control of the spacecraft platform
- planning and execution of flight dynamics manoeuvres
- configuration of platform and payloads
- monitoring and control of payloads
- generation of operation schedules
- derivation of spacecraft engineering and navigational state
- reception, processing, annotation and distribution of payload data (in case of measuring payloads)

The scope of a mission control system can range from a single expert working with a small desktop computer to a string of control centres with multiple spacecraft control processing systems, communication systems and stacks of documentation. Leaving aside the system required to communicate with the spacecraft on one hand, and with other ground facilities or mission users on the other hand, a mission control system has these essential components (Fig.1):

- One or more humans controlling the mission
- A set of information about the mission and its objectives
- A data processing system

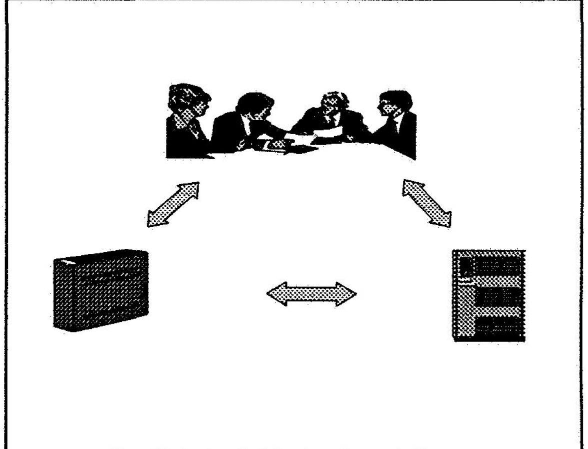

Fig.1 The basic Mission Control System

[1] European Space Operations Centre, Robert-Bosch-Str.5, 6100 Darmstadt, Germany

[2] Logica Cambridge Limited, UK

[3] Logica Space and Communications, UK

In the minimum example quoted above, the mission knowledge is probably completely imbedded in the human expert's mind, and the data processing system performs a simple task of decommutating, displaying and storing telemetry, and transmitting telecommand entered by the operator. This is obviously only feasible in the case of a very simple spacecraft and mission.

The more complex our spacecraft system gets, the more involved becomes the relationship between the mission control elements shown above.

The single human expert is replaced by a team of experts, each dealing with a different aspect of the mission, and each having his own specialist knowledge. In case of round the clock operations, different experts of the same specialisation will work on the same subject, having to exchange information about the state of affairs at shift changes.

The knowledge about the mission becomes too complex and too vast to be held even by this team, and gets stored in paper specifications, plans and procedures, and databases imbedded in the data processing system.

The data processing system will then comprise a large number of inter-related software tasks, implemented on different processors, and connected by a variety of methods, ranging from inter-process communication to access to common data bases to off-line (i.e. via the human user) transfer of data. An example of the latter method, still very much in use in mission operations, is the reading from a printout of the result of a mission planning task, say the time of a particular payload action, and the manual transfer of this result to a command scheduling task.

1.2. Requirements Drivers

The most obvious reason for this ever-increasing 1complexity of mission control systems is that missions become more ambitious and requirements of payload users more demanding. This trend is not just linked to manned missions; in particular in unmanned satellite systems we find:

- operation of multiple payloads at the limit of the resource envelope
- numerous payload modes
- higher payload duty cycles
- shorter mission planning cycles
- direct user commanding of payloads
- shorter payload data turnaround requirements
- higher payload service availability requirements.

Increased mission requirements in turn lead to more complex spacecraft designs, which use on-board resources more efficiently, and are manifest in:

- automatic and autonomous on-board functions
- highly optimised subsystem redundancy concepts
- extensive use of on-board software
- complexity of on-board data handling systems.

The increased complexity of spacecraft directly translates into the need for improved functionality and performance of mission control systems. However, it also means that the scope for on-board malfunctions increases and that different on-board subsystems and operating modes are much more inter-dependent. This requires qualitative increases in the

- understanding of system design
- evaluation of subsystem dependencies
- analysis of operating constraints
- operations safety measures
- contingency handling

in order to operate the mission reliably and with high availability, and to safeguard the high investment in the mission. Most of this increased complexity and functionality of the MCS will have to be absorbed in its data processing element.

However, despite the fact that missions and spacecraft become more complex, the cost of operating them is supposed to stay at a constant level or to even decrease. This of course is most apparent in the commercial space sector, but also operators of long lifetime scientific missions and operational services simultaneously operating of families of spacecraft have to respond to pressures of reducing the share of operations in the overall budget.

Since the most important cost element in mission operations is the manpower needed for preparing and executing a mission, the design of future mission control systems must address the improved use of manpower resources during all mission phases by giving the operations staff the tools enabling them to perform tasks more efficiently.

1.3. Technological Drivers

One of the most striking developments in the Information Technology industry is of course the dramatic increase in the hardware processing power, putting previous mainframe performance at the disposal of workstation users. Linked to the resulting developments in work station technology is a push towards the goal of Open Systems. Open Systems is not a well-defined term, but it will give increased flexibility to the developers of data processing systems in terms of:

- Hardware vendor independence;

- Application inter-operability;
- Applications having well-defined application interfaces so that they can be substituted with a functionally equivalent system without upheaval to other applications which use them;
- Standardisation of application programmer interfaces.

All of these features are important for the development of future MCS; firstly they have implications with respect to the nature of the tools and environments which will be provided by both hardware and software vendors and secondly they have architectural implications for future MCS in their own right.

Perhaps the most significant impact that these developments will have is that an MCS will no longer be a single software entity, but rather will be an integration of proven general purpose tools, software tools and spacecraft operations specific applications.

This is as much a necessity as a possibility since more capable mission control systems will require ever more sophisticated environments and tools to support the operator effectively. However, in order to meet financial constraints at the same time, it is essential that we are able to use commercial applications when appropriate within an MCS and that we are able to reuse software effectively from one MCS to another. An MCS will in future be a collection of integrated applications which provide an overall functionality, rather than a single, indivisible, software entity

In parallel to these developments in processing power and architectures, there have been dramatic improvements in the software discipline in terms of

- knowledge based systems
- advanced man-machine interfaces
- data base management
- software engineering

In particular knowledge based systems have evolved within a few years from a research discipline to revenue-generating systems. In space missions, the natural application for knowledge based systems is in the mission control, where they can help to analyse problems which are too complex (due to the amount of underlying knowledge) for a restricted team of operators to solve within a restricted time scale, or which are of a highly repetitive nature.

Previous ESA research projects have identified the role to be played by knowledge-based systems in future mission control systems. These projects, in the form of prototypes for specific problem domains and specific projects, have successfully demonstrated the usefulness of this technology in mission control applications. The challenge to be addressed now is the full integration of knowledge based systems into an operational MCS.

The latest development in the progressive application of knowledge based systems is the realisation that an important cost factor of such systems lies in the collection and encoding of the knowledge required to exploit them. This has resulted in the recognition of knowledge as a 'corporate resource', and in efforts to improve the re-utilisation and sharing of knowledge between different and diverse knowledge driven data processing applications.

2. ATOS

2.1. Objectives

In order to exploit the available technological advances in data processing technologies, such as knowledge based systems, man-machine interfaces and data base techniques to meet the challenge for improved functionality, performance, reliability and efficiency of future mission control systems, ESA's European Space Operations Centre (ESOC) is pursuing a parallel approach (Fig.2).

ESOC's current multimission software infrastructure, SCOS-A and SCOS-B, which is based on DEXC/VAX midi computers, is planned to be succeeded by SCOS-II (Ref. 4) with a completely open systems approach based on UNIX work stations, allowing configurable, distributed processing and employing object-oriented technology.

At the same time ESOC has initiated the development work advanced mission control system capabilities under the project name ATOS, for Advanced Spacecraft Operations System. ATOS shall complement 'third generation' mission control systems such as SCOS-II as a generalised, integrated data processing platform for advanced mission control and mission support modules, employing state of the art technology with special emphasis on knowledge sharing, automatic reasoning and man-machine interfaces.

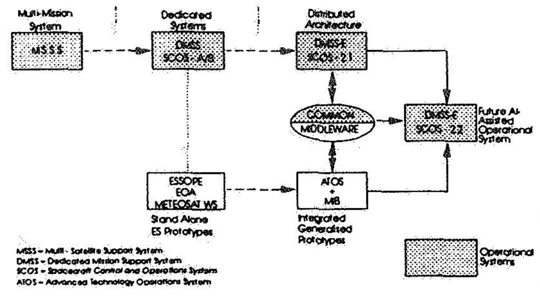

Fig.2 ESOC's MCS Development Approach

ATOS will allow to progressively add advanced control modules for use along with modules of the conventional spacecraft control system so that the advanced control functions can be evaluated under realistic operational conditions, with the conventional functions providing a safe fall-back, until for appropriate applications the advanced modules can fully take over.

Pre-ATOS prototype developments at ESOC have placed emphasis upon the use of tools and technologies in order to fabricate applications and have paid little attention to representational issues. This has led to a situation where the applications that have been built to date cannot be integrated with each other or with the classical elements of an MCS. In some cases even the tools used to build them are no longer available from the suppliers. When the tool goes so does the knowledge present within the system. Often these tools do not formally define the semantics of their knowledge representations. To make matters worse, the inferencing mechanisms often contribute to the semantic content of the knowledge. For all these reasons ATOS is placing emphasis upon the representational aspects of the system rather than the technologies that may be used to build the final architecture.

These overall objectives for ATOS are supplemented by the following generic system requirements:

Mission Independence

Efficiency in the use of manpower resources starts in the development stage for the mission control system for a new mission. Here it is important to be able to base any new development on generic systems which may be adapted to the mission in question with a minimum of additional effort. For ATOS this means the creation of generic intelligent application modules for certain mission control tasks which are problem oriented, not mission oriented.

This requires the removal of all problem and mission oriented imbedded knowledge from the application code, and its storage in parametrised form outside the application so that by changing the external knowledge base alone the application may be set up for the support of a particular mission. It also requires that the representation of such external knowledge must be abstracted from the features of any particular mission.

Task and Knowledge Integration

Increased functional capabilities of the overall system, and improved efficiency in operational exploitation of the system will be achieved by the linking of different intelligent application modules of ATOS. This task integration shall allow different ATOS (and other MCS) applications to share and update external knowledge relevant to their function in a dynamical fashion. It goes without saying that this requires careful strict configuration control measures.

Applications shall also have access to different elements of the mission knowledge, meaning that logical links must be established and maintained between them, allowing the intelligent applications to use these links for enhanced depth of reasoning.

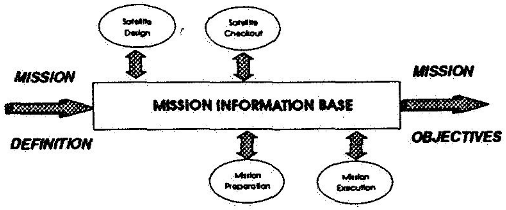

Fig 3 Knowledge Integration

Implementation Independence

Artificial intelligence tools such as will be used in ATOS are undergoing constant evolution, and the tools which may be used in future ATOS modules cannot be predicted now. For this reason, ATOS will be designed as an open system both with respect to representation schemas, paradigms and application tools in order to be able to accommodate future developments without essential changes to the ATOS system concept.

Re-use of Knowledge

The last three requirements can be subsumed under the term re-utilisation of knowledge, to be assured

- between different missions
- between different applications and services
- between different phases and generally in time.

Growth Potential

The system design of ATOS must allow for future growth of the system in the following respects:

- accommodation of future intelligent applications not presently planned for or considered;
- improvements in computer reasoning in support of human operator tasks;
- increases in the volume of knowledge to be processed for a particular application or in mission complexity.

2.2. Architecture

Given the above requirements, the ATOS concept is defined as the combination of:

- A system architecture (including the definition of external interfaces) allowing the overall objectives to be met and guaranteeing the compatibility with the conventional spacecraft control system
- A distributed, but logically unique, repository of mission data and knowledge (i.e. information), called the **Mission Information Base (MIB)**, and the associated tools;
- A set of advanced knowledge based system applications (**Advanced Application Modules - AAM**) interacting with the above knowledge and complying with a common set of internal and man-machine interfaces.

In order to practically realise such an architecture it is interesting to partition the application domains and analyse which information is needed within each domain, and particularly, where these information sets intersect.

Four generic application domains are considered to cover the functionality of mission operations:

- Operations Preparation
- Mission Planning
- Operations Execution
- Training.

The central problem is to identify the information required in support of these domains, and in particular, that information which intersects the domains, in order to permit interworking between the various application modules. For example, planning takes place during the early phases of a mission. Not all preparation and mission design activities will have been completed. There are many interactions between these activities and the system must give support to the users in this regard. Planning, and re-planning, also occurs during the operations phase and must take account of the current state of operations, the spacecraft and its environment. If different applications are to successfully interwork and collaborate in achieving mission goals, they must agree as to which information is shared.

Although it is possible to think of the MIB as a single logically unique entity, it is unlikely that it will be realised as such. Certain partitions of the MIB will be closely concerned with the different application domains. It will be natural, and efficient, for these partitions to be located with the applications that utilise them. However, in order for the applications to interwork it is necessary to adopt some agreements on the form of the shared information. Such an agreement is called as "Ontology". In philosophy, an ontology is a theory about existence. In the artificial intelligence community the term is used to indicate definitional information concerning the "content" of a knowledge base.

Ref.2 defines the term as follows: "The ontology of a system consists of its vocabulary and a set of constraints on the way terms can be combined to model a domain."

2.3. The Operations Ontology

The establishment of an ontology for the domain of spacecraft operations is seen as a crucial component of the ATOS architecture. Without agreement on the form of shared elements of the mission information base there can be no interworking between applications. In this sense an ontology in a KBS application plays a role very similar to that of a database schema in a conventional data system. The crucial difference however is that, since we intend to develop advanced applications which reason about mission information, the form of the ontology needs to be expressed in a way which provides unambiguous semantic intent. Traditional database schemas describe the form of data, but not the content and semantics of the data. Ontology development is at the forefront of knowledge based system development.

The ontology permits ATOS applications to exchange information without loss of semantic content. That is, applications can publish information in a form in which other applications can understand.

The mission information base at any instant comprises the entity of knowledge known to all applications. All applications have access to all knowledge at all times, although each application may (is likely to) preserve certain information in forms suitable for efficient local processing in a representation specifically designed to permit efficient manipulation in the context of one application.

For example, a planning system may require a model of the spacecraft design to be available in a representation which is efficient for the propagation of functional system modes. Such a representation may be entirely inadequate for use in a modelling application used during computer assisted operations. Rather than agree on a standard (compromise) representation, each application should be free to use whatever representation is appropriate, while at the same time being able to publish, and be notified of changes, within the shared knowledge, by virtue of agreement on ontology.

The ATOS architecture is at once efficient, open and practical. It takes account of the practical issues facing knowledge base and knowledge base application implementation, while at the same time creating, through the ontology, a resource able to live on beyond the availability of any particular tools employed in its construction.

The ontology resource, being represented in a machine readable (and translatable) format, shall be able to be utilised in the construction of schemas for various mission information base technologies and in the creation of well-formed coupling interfaces between applications and the distributed access service (Fig.4).

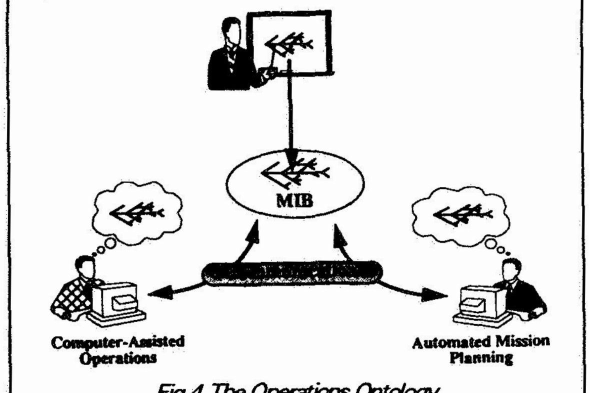

Fig 4 The Operations Ontology

2.4. Mission Information Base

The Mission Information Base as the common repository for all system and mission knowledge has two important integrative aspects:

- It provides continuity and integration throughout the different phases of the mission life cycle:
  - Mission and satellite design
  - Satellite check-out
  - Mission preparation
  - Mission execution
- It also provides integration between the conventional and the advanced technology mission control system elements on one hand, and between the different ATOS application modules on the other hand.

These integration objectives require that the scope of the Mission Information Base contents covers information for all mission phases and for both conventional and advanced mission control modules. In terms of variability of information with time, we consider the following categories:

- Static mission information: fixed information about the spacecraft design, check out results, invariable operations procedures, parameter definitions, static knowledge bases, subsystem models
- Dynamic mission information: current spacecraft configuration, resource data, dynamic knowledge bases, dynamic operations procedures
- Transitory mission data, generated during the mission and immediately processed, distributed and then archived for future reference, i.e. *housekeeping and payload telemetry and* spacecraft activity (telecommand) logs.

Of these categories, the MIB will concentrate on the static and dynamic mission information as the shared mission knowledge. This is however not to be understood as a central repository to which applications read and write. The crucial difference is that in a centralised approach, the only way an application can use its own representation (which we see as an absolute necessity) is for wholesale translation of bulk information into and out of the representation used by the central store. This would be highly inefficient. Rather, we would prefer to partition the information where it is needed and use representations appropriate to the task, only agreeing on the form of shared information.

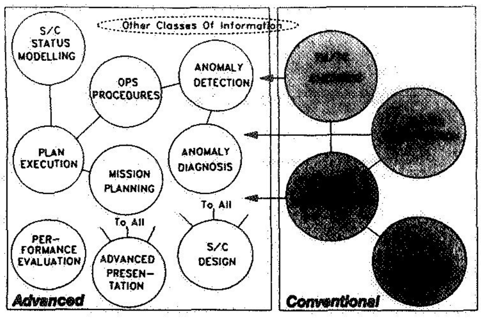

Fig 5 MIB Contents

Research indicates that in collaborative endeavours such as space mission design and operations, where a large number of different disciplines interact loosely, the shared information is significantly less than the whole. Significant functional advantage can be gained by agreeing on the form of this shared information so that it can be freely interchanged.

In ATOS, it is proposed that the medium through which applications will interact will be a local and wide area network "distributed access service". Applications shall attach to the service through access points controlled by "knowledge agents". Knowledge agents shall be autonomous processes able to mediate between applications that employ different knowledge representations and different query and assertion languages.

The set of knowledge agents which comprise the service shall interact using the vocabulary of the shared ontology. The service can therefore be viewed as a "knowledge bus" into which applications can be plugged. An important feature of such an architecture is that shared information is persistently maintained either by the knowledge agents, or by attached "knowledge servers". Such servers can be considered to be applications in their own right.

The Mission Information Base will therefore contain the following building blocks:

- The distributed set of databases covering various information classes and supporting a wide range of different information structures, representational schemas and links.
- Internal management which ensures consistency and configuration control
- A set of tools for data input, browsing, editing and report generation of the MIB
- The Distributed Access Service providing retrieval and update capabilities for the ATOS Application Modules with a common API accessing 'knowledge servers'.

2.5. Knowledge Reuse

By agreeing upon the form and semantic interpretation of spacecraft operations knowledge (i.e. the operations ontology) the possibility to reuse knowledge between missions is made more tractable since generic knowledge, such as generic subsystem models, can be maintained electronically in a form which is independent of the infrastructure which is being used within a particular mission.

This opens up the opportunity for operating organisations to develop libraries of standard reusable knowledge about spacecraft and mission operations which can be used across a variety of different missions irrespective of any technological changes to the MCS infrastructure which may occur.

One of the major tasks facing a mission operations organisation in preparing for a mission is the conversion of information and knowledge supplied by the satellite manufacturer into a form suitable for use within the MCS and its supporting tools. This task is actually the subject of one of the planned ATOS advanced application modules.

The envisaged common ontology of spacecraft operations will enable a mission operations organisation to identify the information and knowledge it requires from the satellite industry irrespective of the specific software employed within the MCS. Furthermore, the ontology can be used to define knowledge capture tools which can be used by the industry to supply the required information and knowledge in an electronically readable format ready for direct incorporation into the Mission Information Base.

3. Planned Implementation and Evolution

The ATOS programme has started in 1991 and initially covers 4 individual projects until 1995. Within the programme, two distinct phases can be identified, overlapping however in time:

- The system and concept phase (ATOS-1)
- The applications development phase (ATOS-2 to ATOS-4), covering the development of pre-operational application modules for advanced mission preparation, automatic mission planning and computer assisted operations.

3.1. System and Concept Phase

The ATOS-I project is concerned with defining the overall ATOS concept and in providing the specification of the mission information base. This involves the following tasks:

- The formal specification of an Ontology of Spacecraft Operations: This must be seen as the "first draft proposal" for such an ontology. It is expected that it shall be further develop, and validated, following ATOS-I and within the other ATOS phases.
- Translation of the ontology into the logical schema of a carefully selected mission information base foundation technology: The ontology shall be studied in ATOS-I to select the most appropriate foundation technologies to support full implementation of ATOS. Having identified candidates, the ontology shall be translated to provide the required database/information base schemas. It is likely that this will involve the integration of relational, object oriented and semantic information models.
- The architectural design of the Distributed Access Service: this shall include the design of knowledge agents able to provide services in support of application inter-operability and of application access to shared information.

The ATOS-1 project is aiming to provide as complete a mission operations ontology as possible, together with a proof of concept demonstrator. In order to be convincing, the demonstration must include the following elements: ontology translation, knowledge level interaction/reasoning between knowledge agents, together with the interface between applications to the knowledge bus, a so-called "knowledge manipulation and query language".

Several prototype elements have been identified and are planned to be integrated. An important candidate prototype consists of an environment in which the ontology can be further developed. This would permit transfer of the ontology to authorised parties within the ATOS programme for the development of specific application modules and, potentially, to allow use of the ontology outside of the ATOS programme in support of the development of advanced applications throughout the industry.

3.2. Applications Development Phase

This phase shall design and develop prototypes of certain advanced application modules (Fig.6). To do this will require access to the ontology developed in ATOS-1, together with tools to support translation of the ontology. Within the ATOS programme there will be a need for co-ordination of development of the ontology. The draft ontology being developed in ATOS-I must be adopted, extended and developed, under proper coordination, by all other ATOS projects if the viability of the approach is to be realised and full benefit brought to the programme. Further details of the measures necessary to achieve this are described in the companion paper (Ref. 3).

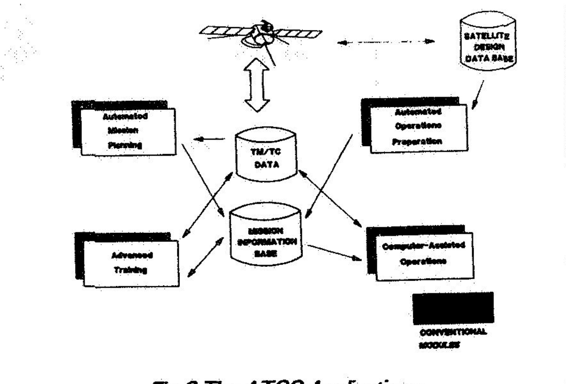
SATELLITE DESIGN DATA BASE
Automated Mission Planning
TM/TC DATA
Automated Operations Preparation
MISSION INFORMATION BASE
Advanced Training
Computer-Assisted Operations
CONVENTIONAL MODULES

Fig 6 The ATOS Applications

Automated Mission Preparation

Presently, the most time and resource consuming task for the operations staff is the preparation of system files and operations documentation prior to their actual use in simulations and operations:

- Generation of Flight Control Procedures from manufacturer's design information, operations handbook(s), subsystem specifications;
- Generation of telemetry and telecommand parameter characteristics files for use by the spacecraft control software, from paper satellite specifications or from a satellite data base also used for checkout purposes;
- Generation of auxiliary spacecraft control information in files or in code, such as mode equations, derived parameter definition, high level command sequences, mission constraints, based on the results of the two activities above, and on additional spacecraft and mission information.

This data is subject to frequent updates during the preparation period and sometimes even after launch, due to design changes, and test and actual operations results. In all these cases, the effects of an update in one item on the rest of the data has to be carefully checked and considered.

The automated operations preparation module will support the operations engineer in this task by automatically transcribing satellite design, build and test data into the appropriate representations and by:

- Reduction of high volume input data from the satellite design and development process, existing in diverse formats, to a defined structure;
- Evaluation of dependencies in the input data and generation of logical links and conditions in the mission information base, using automatic reasoning based on the mission operations ontology;
- Automated Validation of generated Flight Control Procedures for completeness, consistency and operational safety;
- Efficient high level interface with the human operations engineer who remains responsible for conflict resolution and final vetting of the mission information.

Automated Mission Planning

Automated mission planning addresses the automated generation of optimised plans for payload and spacecraft exploitation and configuration, astronaut activity planning and ground facilities scheduling.

The resulting plans shall maximise mission output (activities, products), while minimising the use of depletable or expensive resources (e.g. power, fuel), and satisfying a set of constraints (e.g. safety, deadliness, environment, available resources).

This covers initial mission planning well in advance of actual mission operations as well as near real time re-planning necessary to optimally react to unforeseen changes in the execution of the plan, or in the environment.

Computer-Assisted Operations Execution

Computer-assisted operations address the core mission operations tasks:

- Re-configuration of spacecraft and payload in execution of agreed mission plans or as required by the space environment;
- Manoeuvring of spacecraft attitude and orbit;
- Continuous health monitoring of spacecraft and payload;
- Contingency actions in case of deviations from nominal satellite state and operations.

These tasks shall be covered by a number of intelligent support functions for the mission operator:

- Automatic generation of command sequences for complex Flight Operations Procedures including the necessary feedback from the satellite;
- Automatic telemetry monitoring for comparison with nominal state and trend analysis of critical parameters;
- Automatic direct and indirect anomaly detection and diagnostic support;
- Generation of contingency command sequences based on intelligent failure diagnostics and partially pre-defined procedures.

Adaptive Training

Safe and efficient conduct of mission operations requires comprehensive initial training and repetitive proficiency training of staff throughout the operational lifetime of the mission. This is of particular importance where, due to the overall extent of the operations, tasks become fragmented, and where a long mission duration prevents team continuity. Training can be thought of as a controlled simulation of nominal and contingency operational situations requiring adequate response from the trainee, who is presented with an interface close to the one with a real satellite. The adaptive training module, planned for later implementation within the ATOS environment, shall allow to exercise mission operations using the results of the other ATOS modules adapted to the trainee's tasks, using the knowledge about the mission stored in the MIB.

3.3. Integration with conventional MCS

As stated above, ATOS is being developed primarily to complement ESOC's new mission control system SCOS-II, but the ATOS concept should of course be usable to enhance the capabilities of any reasonably advanced MCS. The foreseen interaction between ATOS and SCOS-II will be a good model of the interaction of other MCS with ATOS.

The fundamental link between the basic MCS and the ATOS applications will be constituted by the MIB, which will be based on the common mission operations ontology. The MIB for a particular mission will logically be considered as a single logical entity, but physically it may be implemented as multiple entities with different storage mechanisms. The MIB can be accessed by messages through the Distributed Access Service passing knowledge requests, which will be served by knowledge servers attached to the MIB or even by intelligent ATOS applications. Such requests may come from ATOS applications or from application of the basic MCS. In fact, in order to avoid overlaps in functions, ATOS applications may make requests through the MIB DAS for basic MCS services (such as telemetry parameter limit checking) to be handled by the basic applications.

Through these mechanisms the scope and functionality of the basic MCS is widened by the advanced ATOS applications, and all applications of this wider MCS are of equal status and can interwork closely through the MIB DAS.

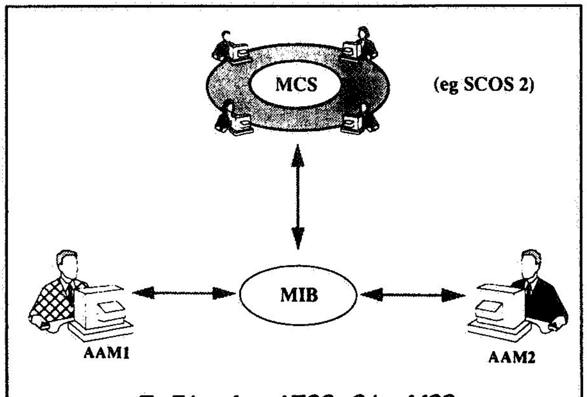

Fig 7 Interface ATOS - Other MCS

4. Conclusion

In as much as it is aiming to advance the state of the art with respect to computer reasoning (intentionally avoiding the term artificial intelligence) in space mission control systems, ATOS must certainly be regarded as an ambitious project. Yet the real ambition lies not so much in the exploitation of the latest data processing technology and knowledge based systems, but in the objective to create a generalised framework which will be open to future advances in the field, and will remain valid beyond the lifetime of currently available algorithms and tools.

The other novel element addressed by ATOS is the concept of operations knowledge sharing and re-use across different dimensions, i.e. lifetime, applications and organisations. The key to this is agreement on the mission operations ontology, which will provide the common understanding on the interpretation of models of the operations domain.

The validity and even necessity of advancing the re-use of knowledge is of course not limited to the problem domain of space mission operations. It has been recognised in many application areas that the high cost of building new information processing systems for the mastering of complex entities requires the re-use of knowledge in the same way that standard software tools are already re-used in present developments.

In the mission operations domain it is probably inevitable that different mission operations organisations, because of differences in their operational requirements and because of different traditions, will continue to develop differing MCS infrastructure tools. However, by agreeing to standardise upon a common ontology of spacecraft operations, in those areas where it is required for the tools to inter-operate, we can establish a practical mechanism for exchanging and sharing knowledge between them.

References

1. M. Curkosky, R. Engelmore, R. Fikes, T. Gruber, M. Genesereth, W. Mark, J. Tenebaum, J. Weber, "PACT: An Experiment in Integrating Concurrent Engineering Systems", Submitted to IEEE Computer January 1993 (Special Issue in Computer Supported Concurrent Engineering).

2. R. Neches et al, "Enabling Technology for Knowledge Sharing, AI Magazine", Fall 1992.

3. K.J. Poulter, H.N. Smith "ATOS-1: Designing the Infrastructure for an Advanced Spacecraft Operations System", Proc. 2nd International Symp. "Ground Data Systems for Space Mission Operations", SPACEOPS '92, 17-20 Nov. 1992, Pasadena.

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