View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by NASA Technical Reports Server

FORMALIZING PROCEDURES FOR OPERATIONS AUTOMATION, OPERATOR TRAINING AND SPACECRAFT AUTONOMY

N95- 23720

François Lecouat, Arnaud De Saint Vincent

MATRA MARCONI SPACE
31 rue des Cosmonautes
31077 Toulouse Cedex, France

Tel: +33 61 39 69 39 Fax:+33 62 24 77 80
email: lecouat@isis.matra-espace.fr

INTRODUCTION

Procedures play a key role in the space domain, since most of the activities that require commanding a spacecraft are based on procedures. Procedures permit to keep the spacecraft inside safe limits whatever happens during operations. Another important property of procedures is that they are a convenient support for bringing together various kinds of expertise in a way that facilitates validation: procedures are written in a language that can be understood by most people involved in a space project.

The generation and validation of operations procedures is a key task of mission preparation that is quite complex and costly. This has motivated the development of software applications providing support for procedures preparation. Several applications have been developed at MATRA MARCONI SPACE (MMS) over the last 5 years. They are presented in the first section of this paper. The main idea is that if procedures are represented in a formal language, they can be managed more easily with a computer tool and some automatic verifications can be performed. One difficulty is to define a formal language that is easy to use for operators and operations engineers.

Once formalised procedures have been generated for a spacecraft, they can be used by other tools for many interesting applications including generation of detailed timelines, automatic or semi-automatic procedure execution, and operators training. Such applications developed by MMS are described in this paper.

Moreover, this concept of formal operations procedures can be adapted to on-board procedures for representing the information necessary to increase spacecraft autonomy. This idea has been explored on the AMR mobile robot and is being developed on the IARES project for CNES dedicated to the development of a demonstrator of a planetary exploration mobile robot.

PROCEDURES PREPARATION

The POM tool has been developed by MMS to support the generation and maintenance of satellite ground control procedures, and to facilitate their use during operations thanks to a procedure browser. POM is now used operationally for the procedures of the Telecom 2, HISPASAT and SOHO spacecrafts. Savings that can be credited to POM during the procedure elaboration phase at MMS were estimated at 50%. Another fine result was the increase of procedure quality.

From the experience of the various procedures management tools developed in the last five years (including the POM, EOA and CSS projects [4]), MMS has derived OPSMAKER, a generic tool for procedure elaboration and validation. It has been applied to quite different types of missions, ranging from crew procedures (PREVISE system [5]), ground control centers management procedures (PROCSU system), and - most relevant to the present paper - satellite operation procedures (PROCSAT developed for CNES, to support the preparation and verification of SPOT 4 operation procedures, and OPSAT for MMS telecom satellites operation procedures).

The basic functions provided by OPSMAKER procedures preparation applications are :
- a procedure editor which supports "assisted editing" (e.g: on-line access to system data) for more efficient procedures writing;
- a procedures compiler, which generates an internal, formal representation of the procedures (and, when applicable, detects syntactic errors);
- a procedures formatter, which generates automatically a high-quality document (FOP, Flight Data File);
- a procedures checker, based on qualitative simulation, which provides for a rich set of verifications to speed up procedure development : simple errors are detected early before starting detailed simulations.

Procedures are entered with the editor in a special form with several columns and various fields. The body of a procedure is entered in a formal language that is a normalisation of the natural language usually used in operations. Quick access to system data (e.g. TM, TC, TC blocks, ground system data) is provided as well as various search mechanisms. In PROCSAT and OPSAT, procedures are saved in a relational database enabling fast search functions and safe team work: several instances of the editor can be opened at the same time (client-server architecture).

The use of a formal language for representing procedures in the Editor (operations engineers view) enables the implementation of a procedure compiler that generates an internal representation of the procedure. The formater then generates a command file for a standard desktop publishing tool (e.g. FrameMaker). Data from the database is automatically inserted in the procedures (e.g. verification TM for a TC, list of TCs for a block) to build up the operators view. The procedures can also include additional information (text and graphics). Formalisation of procedures and modelling of actions facilitate team work by guarantying homogeneous procedures manuals. Everybody works at the same level of detail, with the same language. Maintenance of procedures is facilitated since information is never duplicated and powerful search functions are provided. The use of a normalised language and a normalised presentation by the operations team, should secure the execution of operations.

Several verification mechanisms are provided ranging from simple "local" checks on the individual consistency of every statement, up to the "logical" verification of a procedure by simulating the effects of commands and checking operations constraints (e.g. TC and TC groups pre-validation checks). These verification functions work on the basis of information stored in the spacecraft database. Consistency checking of the operational data and the use of these data without possible corruption improves the consistency and quality of procedure manuals.

There are on-going studies at MMS on the adaptation of OPSMAKER to support *integration procedures* . These procedures used in spacecraft integration have a lot of common aspects with operations procedures. Common data structures and tools would significantly increase spacecraft development productivity.

Another part of mission preparation activities is devoted to the preparation of timelines, in particular for LEOP (Launch and Early Orbit Phases) and IOT (In Orbit Test) operations. An additional advantage of formal procedures is the possibility of detailed timeline generation. Once a top level timeline has been created (timed sequence of procedures) it is quite easy to explore the procedure database and print each procedure action, together with its execution time, in a detailed timeline.

PROCEDURES EXECUTION

Requirements for the improvement of operations safety and efficiency motivate the development of advanced tools to provide a real-time support to spacecraft operators during monitoring and control activities.

The Expert Operators' Associate (EOA), developed for ESOC by MMS and CRI is a prototype centered around the concept of assisted procedures execution. The EOA procedure language allows to attach to the procedure some informations which were not present in the "conventional" procedures: goal, context of applicability, and a more complete description of the execution constraints.

EOA main functions include :
- real-time monitoring of spacecraft telemetry and alarm filtering;
- on-line selection of applicable procedures, in particular in case of contingency;
- managing a timeline of procedures;
- supporting the operator for the execution of the procedure (presenting the chosen procedure to the user in both textual and graphical form, and dynamically reflecting on the display the status of execution of the procedure). Automatic execution of procedures is also possible;
- continuously verifying the validity of operations constraints.

A procedure interpreter allows safe procedure interruption and restart as well as concurrent execution of procedures. A reactive architecture ensures that appropriate response is given to user queries and incoming alarms.

The EOA has been interfaced to the ESOC Multi-Satellite Support System (MSSS), and experimented with MARECS spacecraft analysts on the MARECS simulator, and on the MARECS B2 spacecraft where an eclipse operations was executed by EOA in a completely automatic way (in parallel to the operator). This demonstrates the feasibility of a generic mechanism for semi-automated procedures. Moreover a lot of progress has been made in applications such as PROCSAT and OPSAT, to make the procedure language easy to use by operations engineers. This is a very important aspect for the maintainability of procedure and the acceptability of the tool by users.

MMS is now developing a new generation procedure execution tool that is compatible with the OPSMAKER approach for procedure generation. This procedure executor shall be easily connected to existing control centers as an add-on tool. Expected benefits include:
- improved reliability of spacecraft control thanks to pre-recorded procedures, automatic TC uplink verification, greater number of checks (constraints verification), assistance in conditional branching, timely invocation of procedures from schedule...
- improved efficiency of spacecraft control: operators can be relieved from real time monitoring for well tested procedures (e.g. eclipse procedures), fast execution of recovery procedures (e.g. payload switch-off).

With respect to ad-hoc computer programs implementing procedures, this concept shall permit:
- better observability and control
- an interactive execution mode where the operator can be fully in the loop
- a formalism to encode the operations that does not duplicate efforts and that facilitates maintenance.

OPERATOR TRAINING

Operators training in a spacecraft control center is a recurrent activity, which is going to take an increasing importance with the growing complexity and increasing life duration of modern spacecrafts. In this perspective, it appears essential to develop new training environments/tools allowing to make this task easier and less demanding on instructors availability.

This is the objective of the on-going ATIS project, carried out by CISE and MMS for ESA/ESTEC [1]. This system is applied to the case of astronaut training to the operation of a microgravity payload (RAMSES), but is based on widely applicable concepts and mechanisms which are :

- tutoring functions/modes : in these modes, the user can access to and navigate in technical / operational documentations, either in a free manner, or being guided by the system following an initially specified "training objective";
- procedural training functions/modes : in these modes, ATIS is connected to a simulator. The session is started by specifying an initial scenario (possibly a contingency case) ; the user (operator) executes an operational procedure as in "traditional" simulation session, but is constantly monitored by ATIS which in parallel tracks the procedure to be executed. In case of error, the operator is given corrective guidance. Contextual access to relevant informations is also provided.

Such functionalities could be usefully integrated to a Mission Control Center. A key point is that such a tool can reuse a large part of data and knowledge already produced by other tools during mission preparation (in particular formal procedures). Having a unique source of information for training and operations will enforce the representativity of training.

INCREASING AUTONOMY WITH ON-BOARD PROCEDURES

Many space projects can benefit from a greater spacecraft autonomy. This can be achieved by:
- performing well defined operations on-board without ground intervention
- optimizing the use of the communication link, and of ground processing by generating synthetic reports for the ground
- providing on-board anomaly handling mechanisms.

Formal procedures associated to an on-board procedure executer can help to achieve these requirements. A library of data structures representing operations procedures is stored on board. Procedures to be executed are referenced in a master timeline, and the procedure executer starts interpreting each procedure at the appropriate time. This brings many advantages with respect to dedicated on-board software or to simple on-board command sequences:
- convenient representation: a procedure is more expressive than a command sequence (it contains command verifications, branches, constraints).
- cost saving: procedures are directly written by operations engineers in a high level language, not by software developers.
- ease of validation: the control mechanism is decoupled from procedures. When a new procedure is written the control mechanism has not to be validated.
- finer control: progress of a running procedure can be monitored. Execution can be interrupted and resumed. General exception handling mechanisms can be provided.

An alternative to procedure execution for increasing autonomy would be planning. Not only these techniques are quite complex to be implemented on-board, but they may be not very well suited. Two simple facts give an idea why state of the art planners cannot replace procedures. First of all, space operations are often described with constructs not supported by planners, like loops and execution constraints. Second, the goals that underlie operations preparation are not only expressed in term of states, like in most planners, but also in term of behaviour over a period of time as described in [3] (e.g. "diagnose cause of alarm1 and alarm2 before reconfiguration").

The AMR mobile robot project and the on-going IARES project for CNES are two contexts in which MMS explores related ideas.

CONCLUSION

The formalization of operations procedures brings a lot of benefits. It facilitates mission preparation thanks to automated procedure verification and formating tools. It also makes possible new applications for operator training and operations automation.

REFERENCES

[1] A.Bertin, L.Bertotti, F.Buciol, A.De Saint Vincent, P.Caloud, J.Kaas, F.Allard, *An Intelligent Tutoring System for System and Procedural Training of Astronauts*, 4th ESA Workshop on AI and KBS for Space, Noordwijk, The Netherlands, May 1993.

[2] J.M.Darroy, F.Lecouat, J.M.Brenot, A.de Saint Vincent, *OPSWARE : A new generation of software tools for making space operations faster, better and cheaper*, 44th Congress of the International Astronautical Federation, Craz, Austria, Oct. 1993.

[3] M.Georgeff. *Reasoning about procedural knowledge*, Proceedings of the AAAI 85 Conference, 1985.

[4] F.Lecouat, M.Nielsen, K.Grue, *Knowledge-Based Assistance for Generating and Executing Space Operations Procedures*, ESA Workshop on AI and KBS for Space, Noordwijk, The Netherlands, May 1991.

[5] A.De Saint Vincent, F.Lecouat, G.Leonis, F.Allard, *PREVISE: a KBS to Support the Preparation and Verification of Crew Procedures*, 4th ESA Workshop on AI and KBS for Space, Noordwijk, The Netherlands, May 1993.

The AUSTRALIS-1 Control Architecture —
A Behavioural Model for Satellite Autonomy

C. Lindley

THIS PAPER WAS NOT SUBMITTED.