Mars Express Mission Planning: An example of successful software reuse

Thierry Bru
European Space Operation Centre (ESOC)
Robert Bosch Strasse 5, D-64293 Darmstadt, Germany
Fax +49 (0) 6151 902067, E-mail: thierry.bru@esa.int

Erhard Rabenau
NOVA Space Associates Ltd
11, Kingsmead Square, Bath BA1 2AB England
Tel: +49-6151-902989, E-mail: erhard.rabenau@esa.int

Marc Niézette, Ian Shaw
Anite Systems GmbH
Robert Bosch Strasse 7,
D-64293 Darmstadt, Germany
E-mails : marc.niezette@anite.com, ian.shaw@anite.com

Abstract. Mission Planning is an area for which no official reusable infrastructure exists. This contrasts with Mission Control Systems and simulators where reusable kernels have been used for many years and are in some cases in their third generation. However there have been successful examples of reuse of mission planning systems and this paper describes the most recent one. It shows the evolution of the MPS starting from the first study targeting a "Generic Mission Planning Facilities", which began in 1995, to more sophisticated systems like Envisat, which is a very complex mission from the planning point of view, or Mars Express, which is the most recent one. The resulting "MPS Kernel" will be detailed. All along, the major motivation for this evolutions were costs reduction imposed by short timescales available for the developments. The goal has been achieved whilst keeping a high level of quality in all aspects of the missions. Technical choices were made in order to reduce the licensing costs, and the design of the software did target the reuse of modules between missions. Furthermore, extending the initial idea of a generic tool, the configurability of the MPS has been significantly enhanced in the sense that numerous possible changes coming from the users community, can be integrated to the software just by editing configuration files, without changing the source code itself. This paper will finally shows the different possibilities of expansion of the actual kernel and the different studies proposed on the subject. Lessons may be drawn for the goal of a reusable MPS infrastructure.

1. INTRODUCTION

This paper is organised as follow:

- Overview of the Mars Express mission with experiments.
- Summary of the main objectives of the MPS.
- Historic of the Mission Planning Systems at ESOC.
- Presentation of the lessons learnt from Envisat MPS
- Introduction to the concept of the Mars Express MPS.
- Technological aspect of the software implementation.
- Architecture of the system and users' tools.
- Presentation of the future possible developments.

2. MARS EXPRESS MISSION OVERVIEW

2.1 Description

The Mars Express mission comprises nowadays one spacecraft (S/C) and its instruments, orbiting Mars to perform scientific observations and measurements. The main objective of the mission is to search for sub-surface water. The science goals are manifold and challenging for a low-budget mission. They include instruments and experiments to conduct:

- Global high-resolution photogeology
- Global spatial high-resolution mineralogical mapping of the Martian surface
- Global atmospheric circulation and composition
- Subsurface structure mapping down to permafrost
- Measurement of surface-atmosphere interaction
- Measurement of atmosphere-interplanetary medium interaction.

The spacecraft is powered by solar arrays in combination with a Lithium-ion battery pack. Data communication is via a single high-gain antenna in X-band, with the data rate varying between 28 kbps and 228 kbps. Solar conjunctions and oppositions contribute to telecommunication outage periods. All telemetry and science data is stored in a Solid-State Mass Memory (SSMM) device. The S/C is three-axis stabilized by reaction wheels. Thrusters are provided for orbit and attitude maintenance in certain S/C modes. The S/C needs to be pointed to Earth for communications while instrument operations require pointing to nadir or inertial attitudes, e.g. in support of star occultation observations.

The ground segment comprises the Mission Operations Centre (MOC) at ESOC and the new ESA deep space ground station at New Norcia, Australia, which is remotely controlled from ESOC. The MOC performs the traditional functions of monitor and control, data transfer, orbit prediction, determination and maintenance, attitude maneuvers, payload operations and mission planning. Science operations and mission planning are supported initially by a Lander Operations Control Centre at Leicester University, UK, and a Payload Operations Service (POS) managed by Rutherford Appleton Laboratory (RAL), UK.
The nominal mission duration is one Martian year (687 Earth days) with an extension option of another Martian year. The S/C is flying in a highly inclined elliptical orbit with a period of about 7.5 hours. This orbit has been designed to achieve a global coverage of Mars.

2.2 Needs for Planning

The main objectives of the Mars Express Mission Planning System (MEX MPS) were initially as follow:

- To check that external requests for observations, generated by Principal Investigators, and communicated (via the Science Operations Centre) in the form of Payload Operations Requests, do not exceed the resources (e.g. power, data) available to the spacecraft.
- To process the Lander (Beagle 2) operations according to requests for operations generated by the Lander Operations Centre in the form of Lander Operations Requests (Obsolete).
- To plan the Mars Express Lander Communications Package according to the requests for operations in the form of Lander Contact Requests.
- To plan the Mars Express spacecraft operations according to the requests for operations generated by the Flight Control Team (FCT) in the form of Spacecraft Operations Requests
- To plan Mars Express Flight Dynamics Operations according to the requests for operations generated by the Flight Dynamics Team in the form of Flight Dynamics Requests
- To provide the Detailed Mission Operations Plan (DMOP) to inform all concerned parties of the Mars Express operations planned in response to their requests. In addition, a subset of planned operations will be reported via the Detailed Science Operations Plan (DSOP)
- To generate the command schedule for both the spacecraft and the ground stations in accordance with the generated planning
- To generate timeline event information (“Operations Lists”) for use by the operators of the MOC, in accordance with the plan.
- To provide the Restituted DMOP (RDMOP) to inform all request originators of the operations actually executed on-board the spacecraft with respect to the planned platform and payload operations (i.e. was planned event executed as expected, delayed, or failed to execute for some reason).

The number of operations requests both for payload, S/C subsystem and ground station operations is estimated to be in the order of 300 per week. Thus, the amount of tasks to be performed daily requires an automated system that supports operators in their tasks. A decision was therefore taken to implement a Mission Planning System at ESOC.

3. MISSION PLANNING SYSTEMS

3.1 Historic

Since the middle of the 90’s, ESOC has initiated several study projects with the purpose of assessing the “Proof of Concept” of a Generic Mission Planning Facility (GMPF) . Several prototypes have been produced: Advanced Technology Operations System (ATOS)-3 in 1997 has shown the limitation in term of functionalities by the maximalisation of the use of commercial tools, and ATOS-4 in 1998. The latest was a model-based mission control system based on the use of a generic all-purpose model of the mission shared by all component of the Mission Control System (planning, diagnosis, monitoring and procedure execution). It has been integrated into the ERS-2 mission control system, configured to handle a fraction of the ERS-2 mission. The main difficulty there was to adapt the planning technology to the rules that needed to be applied.
The resulting idea from these experiences led to the concept of a general approach for modeling the planning domain, in a representation of the internal modeling close to the user’s representation of the planning domain and problem. This was the start of a general approach to modeling and planning, extending as such the approach used by the GMPF study. The idea of a “generic” kernel based on Object Oriented techniques was born.
The cost reduction motivation was of course at the centre of the development of the kernel. Generic doesn’t mean that the kernel could be reused without modifications at all. Still some adaptations and extensions are necessary to match new mission’s requirements, but the central components of the model should be equally valid, regardless of the mission. Nevertheless, development based on “delta” specifications and such a generic kernel, shorten considerably the realization of a basic running system. Furthermore it reduces the efforts needed for producing the documentation, since only “deltas” are necessary. This help the development team as well as the team who will use the system operationally since it becomes easier to learn about the new system if you have been using the previous one.
The concept of mission planning kernel has been first implemented with the Envisat mission. The description of the kernel itself will be presented, later on in this paper, within the context of the Mars Express MPS, this system being the most mature one.

But first let us see what were the lessons learnt from Envisat.

3.2 Lessons learnt from Envisat

The Envisat Mission Planning system has been put in operation at ESOC since the beginning of March 2002. The principles leading the development of the software have been validated through the operational usage of the system. The lessons learned from the system development and the early operations are summarized hereafter.

- The module-based approach to mission planning is well suited for this kind of application and facilitates the re-use of the Kernel.
- The development of the rules in C++ provides a high-level of flexibility to the developers.
- The Fully configurable scheduling layer allows the mapping of activities and parameters to actual commands or commands sequences.
- The Input file parsers are decoupled from the file handlers enhancing the flexibility of the system in term of commercial product's dependencies.
- The successful re-use of the ERS Graphical Interface layouts has greatly helped to the familiarization with the new system.

The key factors having influenced the design are:

- Costs drivers
- Planning requirements are specified as rules, which identify a situation and an action to be taken when the situation occurs on the plan.
- The planning of some instruments is actually performed by external routines provided by the customer, which have to be integrated in the planning software.
- The requirements on the interaction between the user and the system lead to planning being divided into intermediate steps, which can be selected individually by the user.

3.3 The kernel concept

The development of the Envisat MPS is based on a Mission Planning Kernel providing a set of C++ libraries of re-usable components that covers the main areas of the Mission Planning domain. MEX MPS has taken over this successful concept of re-usable components, which are described in this chapter. They are split in four main groups:

- Input components, used to import data into the MPS.
- Output components, formatting the MPS outputs, such as schedules and reports.
- Planning components, combined to create the core of the planning and scheduling applications.
- MMI components, used to compose the graphical user interface.

The central element of this structure are the planning components, which are subdivided into two groups:

- Static components stored in the MPS database.
- Dynamic components generated during the planning process.

The static components are:

- Missions elements like the platform, the instrument or the environmental elements (e.g. visibility, eclipse, etc.).
- planning rules associated to each elements of the system.
- possible states of the systems and their combinations (e.g. on/off transitions and associated conditions)
- Resources available (storage capacity, link budget, etc.) and their behaviour (consumable, reusable, etc.)
- Commands sequences allowed as extracted from the mission database.

The dynamic elements are:

- Service requests, which are created from the planning input files by the planning process.
- Plans, which are created by the planning process.
- Activities, which are created by the planning process.
- Occurrences of environment element states, which are created from the planning input files by the planning process.
- Command or command sequence calls, which are created by the scheduling process.

3.4 MEX Mission Planning System

Mars Express has inherited from the Envisat MPS, and the status of the software is presented in the following lines. Core changes have been introduced in the kernel, resulting in technical improvements of the software. They are presented hereafter.

3.4.1 New ANSI standards C++ implementation of each individual rules

The proposed use of Rules to support the planning mechanism of the Mars Express MPS aims to make the planning tool more adaptable with respect to changing operational rules and constraints, or to new information concerning device characteristics, etc. The idea of a generic rule-based planning system has been studied in the past - it seems attractive, elegant, and highly configurable. However, implementing a generic rule language to implement the planning steps would require a significant effort, especially effort in the generalisation of concepts, which was not required for the final goal of the project, i.e. the design and implementation of the Mars Express MPS. Nevertheless, the idea has been kept for future developments.

3.4.2 Rule execution mechanism

The reason for using C++ (as already mentioned) is to avoid the development of a dedicated operational language for rules, while ensuring a minimum query/action structure in the way planning steps are implemented.

- Rules are assigned to modules: Typically, a given module will oversee the execution of a single rule, therefore, modules correspond to the lowest level of resolution at which the User can manipulate the Rule Application sequence, i.e. the "atomic" unit of rule application, for purposes of selecting / deselecting.
- Dependencies may be declared between Modules: Within the configuration database, these dependencies influence the rule application sequence. A "Hard" dependency between 'A' and 'B' forces the User to apply module 'A' prior to the application of module 'B'. A "Soft" dependency between the same two modules will ensure that 'B' follows 'A' is both are to be applied – however, sensible results can be obtained from module 'B' in either case.
- Condition and action evaluations are implemented as simple C++ methods ('apply' methods attached to specific condition and action classes).

3.4.3 Object oriented database

A database is needed to store the Static and Dynamic components. The following criteria have driven the choice of a DBMS to store these objects:

- Minimising the mismatch between manipulation and programming language: One of the main criticisms of relational database programming is the mismatch between the data manipulation language (DML), normally SQL, and the application programming language, typically some general purpose language such as C. Relational database applications have an mismatch, in that database access via the query language is table-based while application programming is individual value-based. Extra code and intellectual hurdles are required to translate between the two. A benefit of Object Oriented Database (OODBMS) is that the application programming language and the DML are the same. The design of the Mars Express MPS being an object-oriented design, the choice of an OODBMS to store these objects would therefore minimise the work needed to interface the MPS and the DBMS.
- Transactions and concurrency model: Several independent physical processes will concurrently access objects in the database (e.g. planning and scheduling share plans). A transaction mechanism would ensure consistent access to the database, although this aspect is here less critical than in other systems, in a sense that an acceptable solution for data locking could be implemented in the MPS itself.
- Versioning capability: A database versioning capability is essential, especially for static elements of the Mission Planning Database.
- Persistency model: Persistence may be based on an object's class, meaning that all objects of a given class are persistent. Each object of a persistent class is automatically made persistent. An alternative model is that persistence is a unique characteristic of each object (i.e., it is orthogonal to class). Under this model, an object's persistence is normally specified when it is created. This solution would be preferred, as it would provide more flexibility for the use of transient objects.

The use of an object oriented data storage mechanism was appropriate. Furthermore, the toolkit chosen for implementing the persistent objects needed by the MEMPS (POST++), does provide an adequate basis upon which to build the required functionality. POST++ is a freeware, which is good for the costm and it had shown improved performance upon the previous commercial database used on Envisat. The kernel software interface to the database has been adapted to accept easily other DBMS.

3.4.4 XML (Extended Mark-up Language) for the configuration of the database

A shortcoming encountered with the Envisat configuration database population was that it not being possible to modify the default settings of database objects without software support (to modify the initialisation code). Under the new scheme, all objects entering the configuration database are held in XML files, these files to be ingested at the database population stage. In this way, it is possible to configure all aspects of the mission (payload, platform, resource models, rules) at the configuration level, no code change being necessary. The freeware 'Xerces' from the Apache XML Project, providing a world-class XML parsing and generation, has been selected for Mars Expresss MPS.

3.4.5 MIB (Mission Information Base) import

The Service subsystem of the kernel has been updated to allow for input of command sequences directly from the MIB. The MPS retains (within its own configuration database) a snapshot of this Mars Express Operational Database to populate the "Service" aspects of the configuration database, such that only valid command sequences are allowed as requests.

3.4.6 User tailoring MPS output

The kernel version in place for the Envisat development allowed for a number of the outputs to be specified to a large extent by the user, utilising "templates" which are picked up at run-time, combined with planning data, and the resultant output being written to the associated files. This mechanism has been extended as follows:

- The majority of MPS output file types are now defined in this fashion. The user can amend absolutely all aspects of these file types.
- The templates to define the output files are now formatted using XML (as are all configuration files within the MPS) – differing formats are no longer employed.

3.4.7 Move to 64 bits Operating System

The new platform for MEX MPS is Solaris 8. The move to a 64 bits application has allowed increasing the addressing range available for the planning system.

3.5 Tools(Architecture)

The overall Mars Express MPS is composed of seven physical processes, which are presented hereafter. By using these tools, the mission planner will be able to generate a mission timeline (see figure 1), which can be edited for modification.

3.5.1 The Planning Input process

The Planning Input process regularly polls the directory in which data files for Mars Express Mission Planning are received. The received files may be from any external source. On receipt of a file it is checked for valid name and syntax. If a file fails syntax check, a warning message is given, the file is moved to an invalid files area and a file syntax check report is produced. A file that passes syntax checking is moved to a Holding area. The Planning Input process exists only in the operational environment.

3.5.2 The Master Operations process

The Master Operations process provides a range of facilities enabling the viewing, deleting, printing and copying of files in the Holding or Master or History or Working or Invalid File areas, and the viewing of syntax check error reports. The main function of the Master Operations process allows files to be selectively added to the Master area from the Holding area.

3.5.3 The Manual Requests editor process

The Manual Requests Generation process provides the facility to create Manual Requests (MR) for all data types using a selected input file as a template. After creating a Manual Request file in the Working Area, the Manual Requests Generation process provides facilities to edit the file. An Instrument Unavailability file can also be generated with the editing facility in the Manual Requests Generation process. Each Manual Request file will be assigned a unique MR number. When an MR is selected for transfer to the Master Area a syntax check will be performed.

3.5.4 The Plan Engine process

The functionality of the Plan Engine process covers all aspects of the handling of the Plan, the working unit of the Mission Planning System. The lifetime of a Plan can be considered over several distinct phases:

- Creation of the Skeleton Plan
- Building up the unplanned plan
- Application of the planning Rules (can be done in several stages)
- Un-planning of the Plan
- Committing the Plan into the Master Plan

3.5.5 The Scheduler process

The Scheduler process provides the second major function of the MPS, i.e. the scheduling function. This process converts activities from a plan, or from a section of the master plan, to command or command sequence calls meaningful to the recipient mission control functions.

3.5.6 The RDMOP editor process

The RDMOP Editor provides an interface for the generation of an RDMOP file. The generation such a file will require processing in a number of stages that are driven by the user from the editor interface. An option will be provided to create a file in an RDMOP working area by merging DMOP files over a specified orbit range. An option shall enable a TC (Telecommand) Delta file to be generated from a comparison of time corrected data in a TC Schedule file and a TC History file. A user interface option will provide a display of a TC Delta file and an editor to enable an RDMOP file to be manually updated.

3.5.7 The Mission Planning Database editor process

The Mission Planning Database Editor provides an interface to insert, delete, or update items of Configuration Database.

4. FUTURE DEVELOPMENT

Expansion of the Kernel

From the mission control infrastructure evolution plan 2004-2009, it is already specified that the definition of control procedures imported from the off-line flight operation plan, as well as the generated schedule increment, will comply with the PLUTO standard [5].

Going in the same direction, a High-level Query language could also be developed to express the constraints that mission planners want fullfilled by the generated plan.

Taking optimisation techniques into consideration would certainly help the mission planner to find solutions to conflicting schedules. Science and orbit planning constraints should be integrated into this scheme.

Grounds segment planning being limited and in a way decoupled from the spacecraft activities planning, a more elaborated coordination of the resources would certainly lead to an overall more coherent planning.

5. CONCLUSION

The MEX MPS as it is now, will be reused for the Venus Express MPS. Both missions have a lot in common, starting by the design of spacecrafts, going to the numerous similarities between the missions. At this point of the development of the Venus Express MPS, only the operational constraints distinguish the two MPS. The is the equivalent to a delta development, mainly oriented in configuration changes.

The success of the MEX MPS reuse is due to the long history of the mission planning system kernel, which has been developped over several missions at ESOC, with the strong will from the designers, to extract the essential features each time. The choice of object oriented technology has allowed to design a system, which is following the user representation, characterised by the modularity of its reusable components. To mentioned also is the tendency to release the pressure of commercial products. This has an impact on the design of the system, which should be able to adapt itself to any free product following the conventional standards of the industry.

6. REFERENCES

[1] E.Rabenau, M.Denis, P.Jayaraman, *Implementation of a mission planning system for an interplanetary mission*, SPACEOPS 2002, Houston, Texas, October 2002.

[2] M.Niézette, I.Shaw, *Planning how to cut the cost of mission planning*, RCSGSO 2003.

[3] F.Delhaise, M.Niézette, I.Shaw, *A new world for the generic mission planning kernel*, SPACEOPS 2002, Houston, Texas, October 2002.

[4] B.Santos, I.Shaw, M.Niézette, *Mission planning for Europe's latest earth observer,* SPACEOPS 2000, Toulouse, June 2000.

[5] PLUTO: ECSS-E-70-32 *Ground systems and operations - Procedure definition language*.