Evaluation of the Mars Express Planning and Scheduling Approach Over the Lifetime of the Mission

Erhard Rabenau[(1)], Michel Denis[(2)]

[(1)] NOVA Space Associates Ltd
Bath, BA1 2AB, UK
Erhard.Rabenau@esa.int
[(2)] European Space Operations Centre
Robert-Bosch-Strasse 5, D-64293 Darmstadt, Germany
Michel.Denis@esa.int

Abstract

For ESA missions like Mars Express (MEX) which are 'firsts' in many aspects (e.g. first European interplanetary mission, first mission developed on low budget but along a short and accelerated development schedule), the definition of mission planning and scheduling (P&S) requirements depends very much on the experience of the team, the existence of 'legacy' systems and the distribution of knowledge within the organisation. Changes to the operations concept imposed by anomalies of spacecraft subsystems, failures and other unforeseen factors, may impact the P&S approach at any time during the mission. Initially, a P&S system will typically be developed and budgeted to support the nominal mission. Automation will not play a driving role because of cost constraints, development uncertainties and the learning curve on new science missions is very steep. As the mission progresses and more and more knowledge is collected, or because of anomalies and changes to the operations concept, the engineers may develop tools in order to ease the workload and automate error-prone operations. These tools may range from simple visualisation to complex planning applications. For Mars Express a suite of tools was developed over the years (MPS++) that automated the planning of all aspects of communications. Tools based on Artificial Intelligence (AI) engines were developed and integrated into the operational process to support the uplink and downlink planning. A problem inherent in these in-house developed tools is that they are poorly documented, written in varying programming languages, lacking in configuration-control and, most importantly, cannot be easily maintained by anybody other than the programmer. The complex mission scenario of Mars Express always requires new configurations based on the changing environment for which no test cases exist. The only solution is to update the tools. For a long-duration mission, or a mission with several extensions, such maintenance tasks become a problem, if the engineers / developers are assigned to new functions outside or even inside the project. As new missions prepare for operations, they can generally benefit from improvements and increased functionality of existing P&S systems. On the other hand, new requirements may be implemented in existing systems that provide increased functionality. The latter feature may be employed by flying missions to move from the use of dedicated tools to generic ones that are part of the infrastructure development effort of the Agency. This paper focuses on a) the effort expended by the Mars Express mission planning team to integrate the functionality of the MPS++ tool suite in the "corporate" Mission Planning System making use of the Language for Mission Planning (LMP) functionality added to the mission planning kernel, b) the advantages of such a move and how future missions may benefit from this and c) the changes of ground station planning and scheduling and how the concept employed by Mars Express paves the way for a standarized P&S approach for ESA deep space station planning. The P&S concept adopted by MEX may be referred to as 'automation with a human touch', meaning automate what is simple and repetitive but never take the human out of the loop, such that operations remain safe, meaningful and under permanent control of clearly empowered personnel. Automation can be implemented in the P&S processes on ground, but 'lights-out' operations are not suited for complex and demanding missions like MEX.

Nomenclature

AI	=	Artificial Intelligence
AOS	=	Acquisition of Signal
DSN	=	NASA Deep Space Network
ECS	=	Event and Communications Skeleton
EMS	=	ESTRACK Management System
ESOC	=	European Space Operations Center
ESTRACK	=	ESA Tracking Network
LMP	=	Language for Mission Planning
LOS	=	Loss of Signal
MCS	=	Mission Control System
MDAF	=	MTL Detailed Agenda File
MEF	=	Master Event File
MEX	=	Mars Express
MEXAR	=	MEX Scheduling Architecture for Downlink

MISS = Mars Express Integrated Station Scheduler
MPS = Mission Planning System
MPS++ = Enhanced Mission Planning Software Suite
MTL = Master Timeline
P&S = Planning and Scheduling
RAXEM = MEX Scheduling Architecture for Uplink
SOE = Sequence of Events
VEX = Venus Express

Introduction

MARS Express (MEX) is first ‘flexi’ mission of the ESA planetary science programme which postulated a ‘smaller, faster, cheaper, better’ approach to space missions. The MEX mission was implemented on a low budget and accelerated development schedule. Even though the spacecraft is small, it doesn’t imply less complexity. Mars Express exceeded its initial life time of one Martian year and is currently in its 4th extension phase which will extend the mission to 2014. The planning and scheduling (P&S) concept has evolved over the lifetime of the mission to meet the changing requirements of a long duration mission which initially were not expected or simply not known. Being the first implies that there is little knowledge within the organisation to derive ‘lessons learnt’ from and, generally, there is no ‘legacy’ to draw on. The learning curve was therefore very steep. As a result, MEX has provided an extensive knowledge base and paved the way for future deep space missions. In the following, the P&S approach and changes to the concept over time will be illustrated.

The Evolution of the Operations and Mission Planning Concepts

Initially, the operations concept was based on science observations around the pericentre of each orbit. Uplink and downlink communications were to take place during the remaining part of the orbit. A single ground station was dedicated for the communications traffic. Eclipse seasons were not to impact science operations. After launch, several factors surfaced that required the revision of the operations concept in order to ensure the safety of the spacecraft and thus the continuation of the mission, in a way, the downside of the ‘faster, cheaper, better’ approach.: a) a power anomaly permanently reduced the available solar array power which had severe impact on the way the spacecraft was to be operated; b) because of performance limitations in the mass memory unit, the concept of downlinking the accumulated data had to be adapted from a fully automated priority-based system on-board to a sequenced approach planned on ground; c) while initially only the single deep space station of the ESA Tracking Network (ESTRACK) available at the time was scheduled as the baseline for communications, further ground stations had to be added to improve the data downlink situation as the mission progressed (the second ESTRACK station, as well as 34 and 70 meter antennas of the NASA Deep Space Network).

The change of the operations concept drove the approach to mission planning. Processes that were initially performed manually had to be software-supported and/or automated to be able to cope with the requirements. While there was hardly any budget for software development, but the requirements kept changing because of unexpected complexity and, more importantly, the high expectations and tremendous initial success of the science mission, software tools were required to be speedily available without going through a formal and time-consuming software development process. Subsequently, tools were developed by the mission planning team itself to meet the necessity of the moment driven by changing and new requirements. This was mostly achieved by adding functionality to an existing system (in a modular fashion whenever possible and observing a standard file-based interface with the Mission Planning System). In this development approach, testing and debugging cannot always be de-coupled from the ongoing operations. Errors will occur, as it is very difficult and in any case very costly, in a complex mission scenario like the one for MEX, to catch the requirements and appropriate test cases exhaustively.
We are very aware of the downsides of developing tools within the team: they generally do not adhere to software standards, tend to be poorly documented and are written in the ‘favourite’ program language of the engineer responsible for the development. For missions that are extended several times (they effectively become long-duration missions), maintenance of the software may become more difficult because of normal personnel fluctuations within the team, and if loss of experience and knowledge have not been addressed and taken into account.

Mission Planning System (MPS)

Initially, only the MPS was developed to perform all tasks related to mission planning. The specifications of the MPS were driven by the experience of the team (no one at ESA had flown a planetary mission before) and the limited budget available. The planning and scheduling approach by the MPS was mostly driven by input files generated by external applications according to standard interface specifications. Over time the MPS has evolved from dedicated releases for MEX and VEX to a mission planning kernel common to both missions. Furthermore, as ‘inheritances’ feed into the system (MPS being a ‘corporate’ application), new functionalities and improvements have become available. Examples are a) the Language for Mission Planning (LMP), an output from a study, which has been incorporated in the MPS kernel and b) an improved graphical user interface that provides online graphical display of timelines and resource plots which replaces the rather inflexible graphical solution provided with the original release.

Enhanced Mission Planning Software Suite (MPS++)

After launch it soon became apparent that the required input files for the MPS could not continue to be created manually or by the use of simple tools. Especially the handling of the communications with the satellite (uplink, downlink, transmitter switching) and the required ground stations configuration turned out to be very complex and time-consuming. A software application was written by a member of the mission planning team to address these issues. The application, referred to as MPS++ (developed in IDL), grew over time into a very complex software suite that was able to a) process the station application files, determine antenna priorities, remove any station overlaps and generate the so-called Event Communication Skeleton (ECS), b) calculate the transmitter switching pattern, taking the ECS, spacecraft maintenance and science pointing requirements into account, generate a transmitter operations request and a list of available downlink and uplink windows, c) generate the required operations requests for radio science, d) derive the downlink bitrate for a given station track based on a bit rate file (the bit rate file contains different bit rates for different antenna types and sizes), e) generate the Master Event File (MEF) (list of all events that are relevant for planning from the various input files and calculated output events, i.e. station visibilities, station usage, orbital events from the Flight Dynamics Event file, pointing requirements from the pointing timeline, bit rate). The concept to generate the majority of the required input files for planning and scheduling from the MEF proved to be a highly effective way for planning and scheduling. This implied that whenever changes were required, they had to be made at the high-level products, thus easing configuration control as all changes at high level would automatically be reflected at the lower levels. The downside of the tool was that new requirements usually had to be implemented at code level. The workflow based on MPS++ is illustrated in Fig 1.

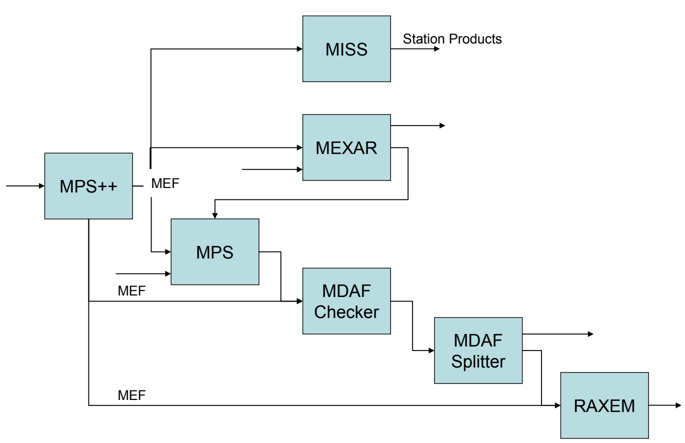

Figure 1: Mission Planning Workflow before MPS2010. It shows the MPS++ as the generator of the MEF and input products for the MPS and supporting applications.

MPS2010

With the inclusion of the generic LMP functionality in the MPS kernel, it was decided in 2010 to translate and integrate the complete MPS++ functionality into configuration files of the MPS, internally referred to as MPS2010. Requirement changes now do not require code changes but can be performed at the 'configuration' level (knowledge of LMP is of course required – see code example in Fig. 2). Because of the paradigm change (all relevant events for P&S now reside on the 'plan'), the MEF has become an output file (generated from XML templates) used for the baseline event collection for the short-term planning and visualisation purposes. The downlink and uplink windows are generated from the plan using the same template mechanism. As the porting effort mostly involved generating LMP code in XML, the cost and effort were deemed acceptable especially in light of the mission extension to 2014 and the subsequent maintainability of the software. The porting was mainly performed by the MPS developer because of limited LMP knowledge in the team (also the MPS developer can provide the maintainability of the system by means of the operational software maintenance contract). The test effort should not be under-estimated, but since many test cases, based on historical data, are available, the operationalisation of the system proceeded smoothly without encountering too many obstacles. Also, because of the limited LMP functionality, not all possible requirements could be fulfilled, therefore simplifications of the requirements had to be performed in some cases. The mission planning workflow, as depicted in Fig 3, has improved: there are fewer interfaces, fewer input files, better controlability, overall saving of time in the planning process.

```
fact(?id1,"MEF", "SWXE", ?st, ?et)
^ owlt(?et, +1, ?oet)
^ ?owltDur <- ?oet - ?et
^ parameter(?id1, "StationId", ?station)
^ parameter(?id1, "dsn", ?dsn)
^ parameter(?id1, "support_mode", ?mode)
-> activity(?id2, "MEF", "TCON", ?oet, ?oet)
-> parameter(?id2, "owlt", ?owltDur )
-> parameter(?id2, "StationId", ?station)
-> parameter(?id2, "dsn", ?dsn)
-> parameter(?id2, "support_mode", ?mode)
-> activity(?id3, "MEF Events", "TCSS", ?oet, ?oet)
-> parameter(?id3, "owlt", ?owltDur )
```

Figure 2: Sample LMP Code

Artificial Intelligence Tools

Other tools developed to improve the planning are two artificial intelligence-based (AI) applications, the first of

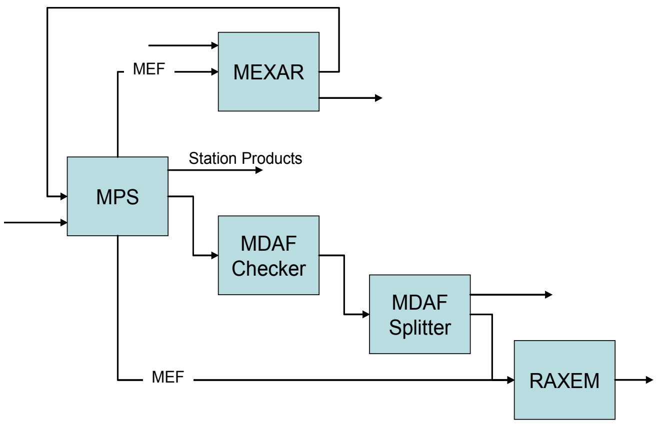

Figure 3: Mission Planning Workflow Using MPS2010. MPS++ functionality has been integrated in the MPS, as well as the generation of the station products. The flow of the generation of the operational products remains unchanged.

such tools in the operations domain at ESOC. MEXAR has successfully been employed by MEX since 2005 to plan the downlink of the science and housekeeping data because of the mass memory problem (see reference 3). The tool uses the downlink windows definition file, a product generated by MPS++/MPS2010 as part of the MEF. The tool not only reduced the downlink planning time by half but also used the downlink channel more efficiently, thus increasing the down-linkable amount of science data.

The other AI-based tool is RAXEM, based on MEXAR, in use since 2007 (see reference 1). The tool uses the uplink windows definition file, a product also generated by MPS++/MPS2010. The tool greatly improved the uplink planning process by a factor of 10. Whatever the complexity and/or scarcity of the station passes, RAXEM consistently provides an uplink horizon of several weeks compared to the 'manual' method which could only achieve a few days.

Ground Station Planning and Scheduling

The scheduling of the ground stations has been the most problem-ridden interface throughout the mission so far. The complexity of the station operations, the handling of occultations and conjunctions and the different approaches to scheduling for ESTRACK and DSN stations very often required manual changes to the station products before delivery. The 2nd generation tool developed by the mission planning team (in JAVA), and in service for several years now, is the Mars Express Integrated Station Scheduler (MISS). It is solely based on the events in the MEF (generated by MPS++ and recently by MPS2010). MISS was programmed to generate the sequence of events file (SOE) for ESTRACK, the ground station jobs for the automatic station control of the ESTRACK stations, the KEYS file for DSN and a radio science experiment list. The automatic station control has since been removed because the direct interface with the ground station computer was too complex and error-prone – in a way because the interface with MPS was set 'at the wrong place'. In order to simplify and standardize the interface for the ESTRACK stations, only the sequence of events file is required, which indirectly drives the ground station configuration performed by the station-dedicated planning and scheduling tool EMS (ESTRACK Management System). The SOE contains all relevant events, like begin of track, end of track, AOS, LOS, transmitter configuration, radio science operations. With the operationalisation of MPS2010 it has been decided to also integrate the functionality of MISS in the MPS. The generation of the station products can be performed from the 'plan' using the proven XML template mechanism (see above). The MEX approach to station planning and scheduling has been adopted as the standard interface for all planetary and deep space missions at ESOC.

Supporting Applications

Other tools were developed by the Flight Control and mission planning teams to support operations. These include

a) a state model checker (developed in PERL) of the telecommand files (MDAFs) before being sent to the Mission Control System (MCS) for uplink to the spacecraft. The MDAFChecker tool was inherited from VEX and upgraded to include the cross-checking of MEF events in the state models. The use of this tool provides an automated way of consistency checking, ('horizontally' between the sequences commanded to the spacecraft subsystems and 'vertically' with regard to what was planned – by checking against the MEF) thus reducing the manual checking effort significantly.
b) a splitting tool (developed in JAVA) that allows the cutting of the MDAFs based on rules defined for the boundary conditions (e.g. TX off at the boundary during eclipse seasons, spacecraft in earth pointing mode, all instruments off) and by type and size. It incorporates the MDAF checker and an automatic transfer of the MDAFs to the MCS. Thanks to this tool the scheduling process could be reduced from several hours to less than half an hour.
c) tools to perform checking of the consistency of the ground station products
d) tools for visualisation of the MEF and downlink plan.

These tools are currently not considered for integration / translation into the MPS because (i) they are outside the planning domain and (ii) it is seen to be of prime importance for quality and safety of the complex operations for MEX that some level of checking is provided by fully independent software (and personnel).

Conclusion

The tools used in mission planning show a trend to more and more automated processes, especially, but not only, for simple and repetitive ones. However, the ‘human in the loop’ remains a firm commitment, so that operations remain safe, meaningful and under permanent control of empowered personnel. The ‘lights-out’ operations idea is not considered feasible, as automation on-board is prohibitively expensive, requires complex planning during development of the spacecraft, but in the end does not provide the flexibility required for a complex and demanding mission like MEX.

In the course of the lifetime of the mission, the mission planning team chose to implement software applications external to the MPS to ease and automate the work load as much as possible. With the extension of the mission beyond the original life time, the maintenance of the tools became difficult. With the availability of added features in the ‘corporate’ systems, i.e. Mission Planning System, it has been possible to mitigate the maintenance problem by integrating as much as possible of the functionality provided by the external tools in the MPS. As the translation effort could be covered by the maintenance budget of the MPS, the translation cost could be kept low. Future missions can certainly benefit from the MEX approach, as the software and configuration files are readily available as ‘legacy’. The experience gained by the MEX mission planning team is the baseline of a knowledge base at ESOC that can be utilised by other deep space missions. For the generation of ground station products, the MEX approach has been approved as the standard interface.

It must be highlighted that all this effort could be done within the envelope allocated for the whole maintenance of the MEX MPS, as low as about a man-week per month (also covering bug fixing, etc.), which shows that a) the ESOC MPS infrastructure is both very solid (very few bugs) and very flexible (built-in ‘user language’) and b) that the ad-hoc operational planning tools used for years by the MEX team were a good investment, as they provided exact requirements and reference prototypes for other missions.

References

[1]Rabenau, E., et al. “The RAXEM Tool on Mars Express - Uplink Planning Optimisation and Scheduling Using AI Constraint Resolution”, SpaceOps 2008. Heidelberg.

[2] Rabenau, E., et al. “Mission Planning Experience Gained from the Mars Express Mission”, SpaceOps 2006. Rome.

[3]Rabenau, E., et al. “Planning Science Data Return of Mars Express with Support of Artificial Intelligence”, SpaceOps 2006. Rome

[4]Schulster, J., et al. “From Mission Concept to Mars Orbit: Exploiting Operations Concept Flexibilities on Mars Express”, SpaceOps 2006. Rome

[5]Rabenau, E., et al. “ Mars Express Mission Planning – Expanding the Flight Box in Flight ”, SpaceOps 2010. Huntsville