Figure 1. The XMM spacecraft

The XMM Simulator
- The Technical Challenges

H. Côme
Simulation Section, ESA Directorate of Technical and Operational Support, ESOC, Germany

M. Irvine
Vega PLC, Welwyn Garden City, United Kingdom

The XMM satellite will be launched in August 1999 and a simulator has been developed to test and validate the supporting ground segment. Due to the very tight schedule and the high fidelity of the modelling required, this development effort has provided numerous unique challenges. This article details these challenges and the approaches that have been taken in meeting them, as well as the simulator's architecture and current status.

The XMM project

ESA's X-ray Multi-Mirror (XMM) observatory mission involves both a space segment and a ground segment. The former consists of the XMM satellite itself and the Ariane-5 launcher that will be used to put it into orbit. The ground segment consists of the Control Centres (MOC: Mission Operation Centre and SOC: Science Operations Centre), the ground facilities, the ground stations and the Science Survey Centre (SSC).

The XMM satellite (Fig. 1) is an observatory mission that will operate in the soft X-ray portion of the electromagnetic spectrum. The heart of the mission is therefore the X-ray telescope, which consists of three large mirror modules and associated focal-plane instrumentation, held together by the telescope central tube. Three scientific instruments have been selected, each of which is being developed by a separate multinational team. Each instrument team is headed by a Principal Investigator (PI) who has overall responsibility vis-a-vis ESA for the delivery of the working hardware and associated services. The three instruments that have been selected for flight are: the European Photon Imaging Camera (EPIC-MOS, EPIC-pn), the Reflection Grating Spectrometer (RGS) and the Optical Monitor (OM).

The XMM ground segment consists of the elements shown in Figure 2. The MOC will be located at ESOC in Darmstadt (D), the SOC at Villafranca near Madrid (E), and the SSC in Leicester (UK). Contact with the satellite will be maintained via the ESA ground stations in Kourou (French Guiana), Perth (W. Australia) and Villafranca during the mission's Launch and Early Orbit Phase (LEOP), and via the Kourou and Perth stations during the routine mission phase.

The MOC is responsible for mission operations planning and scheduling, execution of the schedule, satellite safety and recovery from satellite anomalies, maintenance of spacecraft-platform onboard software, and instrument-anomaly recovery in real-time liaison with the SOC. The SOC, in turn, is responsible for all science operations, including observation planning and monitoring, observation-proposal handling, payload software maintenance, and data distribution to the scientific community.

XMM simulator requirements

An XMM MOC simulator (MOCSIM) is required to test and validate the MOC data systems. It will be used to:

- test and validate the XMM Mission Control System (XMCS) and the Flight Dynamics System (FDS)
- validate the XMM MOC operational database
- test and validate the flight-operations procedures (both nominal and contingency) and the operations time line
- validate the procedures to support System Validation Tests (SVTs) and end-to-end tests
- train the Flight Operations Team.

The MOC simulator is also the data source for the simulation programme, which involves the complete mission-operations team both for the LEOP and routine mission phases.

Similarly, an XMM SOC simulator (SOCSIM) is required to test and validate the SOC data systems. It will be used to:

- test and validate the XMM Science Control System (XSCS)
- validate the XMM SOC operational database
- test and validate the instrument operation procedures (both nominal and contingency)
- validate the procedure to support System Validation Tests (SVTs) and end-to-end tests
- train the Science Operations Team.

The SOC simulator will be the data source for the SOC simulation programme.

The high-level requirements for MOCSIM are:
- accurate simulation of the XMM platform
- software emulation of CDMU (Control and Data Management Unit) and ACC (Attitude Control Computer) onboard processors to allow the actual onboard software to be executed at binary-code level
- functional modelling of the XMM instruments, with accurate modelling of the housekeeping instrument telemetry and dummy modelling of the science telemetry
- identical interface to the MOC as the actual ground-station equipment for telecommand uplinking (Telecommand Encoder: TCE) and telemetry downlinking (Telemetry Processor: TMP4)
- possibility to introduce simulated failures or special effects in all platform subsystems, and to a limited extent in the instruments, during the execution of a simulation run.

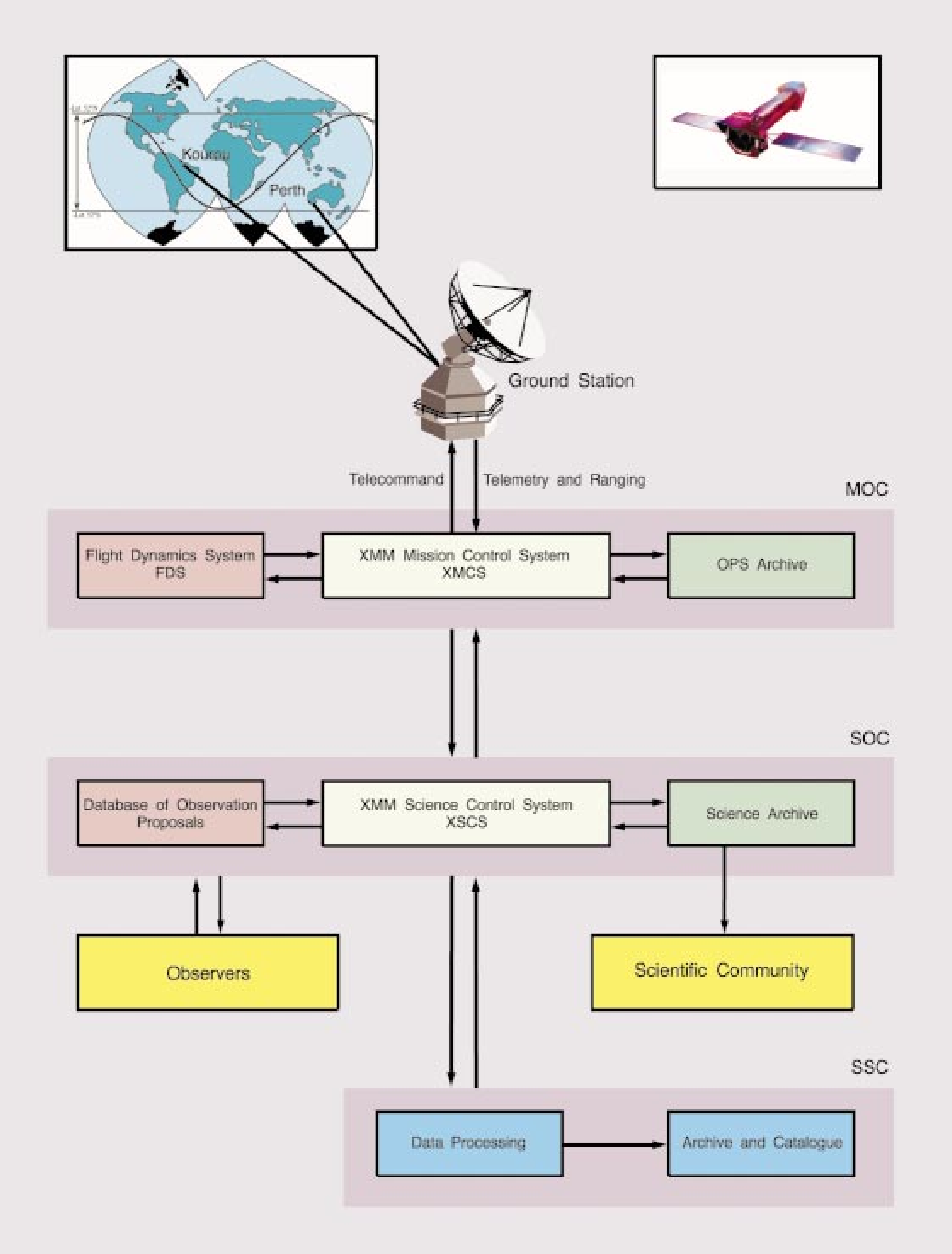

Figure 2. The XMM ground segment

The high-level requirements for SOCSIM are:

- accurate functional modelling of OBDH for all related instrument functions
- accurate modelling of instruments
- software emulation of the various instrument-controller processors to allow actual instrument software to be executed (only one instrument is required to be emulated at a time)
- accurate functional modelling of the instrument software of the processors that are not emulated
- playback of science data recorded with the actual instruments
- possibility to define at run-time which instrument is emulated
- possibility to introduce failures or special effects in all instruments.

Both MOCSIM and SOCSIM are real-time simulators, based on SIMSAT (Software Infrastructure for the Modelling of SATellite). SIMSAT is a standard ESOC infrastructure which provides the core functionality of any simulator: real-time kernel, event scheduling, public data management, logging, commanding, graphical user interface, modelling of such ground-station equipment as telecommand encoders and telemetry processors. Both simulators also make maximum use of standard ESOC generic models and have been developed in Ada to run on DEC-Alpha workstations.

The MOCSIM and SOCSIM modelling requirements are summarised in Table 1 on a subsystem-by-subsystem basis.

As can be seen from Table 1, the requirements for the two simulators are different, although similarity in some of the requirements suggests that a certain amount of synergy between MOCSIM and SOCSIM could be applied. Unfortunately, only 2.5 years were available for the complete development effort, from the review of user requirements to delivery of the

Table 1. MOCSIM and SOCSIM subsystem requirements

Subsystem	**MOC modelling**	**SOC modelling**
Radio Frequency	Accurate	Fixed (TM and TC flow only)
OBDH Bus	Accurate	Accurate
CDMU Processor and Software	Emulated	Functional
CDMU Hardware	Accurate	Realistic (to allow TC/TM flow)
ACC Processor and Software	Emulated	Not required
ACC Hardware	Accurate	Fixed
Attitude Sensors (Sun Sensors, Gyros)	Accurate	Not required
Actuators (Reaction Wheel, Thrusters)	Accurate	Not required
Star Tracker	Accurate	Not required
Orbit and Environment	Accurate	Not required
Dynamics	Accurate	Not required
Power Generation and Distribution	Accurate for platform and instruments	Fixed for platform, Realistic for instruments
Thermal	Accurate for platform and instruments	Fixed for platform, Realistic for instruments
RGS	Accurate for HK Fixed for science	Accurate for HK Accurate for science Science data file playback Emulated and functional IC
EPIC-MOS	Accurate for HK Fixed for science	Accurate for HK Accurate for science Science data file playback Emulated and functional IC
EPIC-pn	Accurate for HK Fixed for science	Accurate for HK Accurate for science Science data file playback Emulated and functional IC
OM	Accurate for HK Fixed for science	Accurate for HK Accurate for science Science data file playback Emulated and functional IC
EPIC Radiation Monitor	Accurate for HK Fixed for science	Accurate for HK Accurate for science Science data file playback
TCE	Realistic	Realistic
TMP4	Realistic	Realistic

Note: The following terminology is used: Accurate: modelling to a declared tolerance, Realistic: modelling such that parameter trends can be observed, Fixed: no dynamic modelling is applied.

complete system. Such a short time frame did not allow elements developed for one simulator (e.g. CDMU modelling of MOCSIM) to be reused as a baseline for the development of the other. It was therefore decided to merge the two sets of requirements and to develop a single simulator, the XMM simulator, which would meet both MOC and SOC requirements.

The implementation of the XMM simulator, as summarised in Table 2, reflects the merged MOCSIM and SOCSIM User Requirements. This table indicates the level of modelling (functional or software emulation) and whether or not a generic model is used.

The XMM simulator has been developed for ESOC by an industrial consortium led by Vega PLC, with Vitrociset as subcontractor. The user requirements and software requirements have been defined by ESA, with the involvement of the consortium during the software requirements phase. The architectural and detailed design phases have been performed by Vega/Vitrociset under firm fixed-price conditions, based on the XMM Simulator Software Requirements Document (SRD). The SIMSAT TCE and TMP4 ground models were developed in parallel by an industrial consortium led by TERMA, with SSL as subcontractor.

Challenges and solutions

The developers of the XMM simulator were confronted with two major challenges: the high fidelity of the modelling needed and the very tight schedule, which together have been the main drivers for the design and development approach adopted.

Table 2. XMM simulator subsystems

Subsystem	Modelling	Generic Model
Radio Frequency	Functional	No
OBDH Bus	Functional	Yes
CDMU Processor and Software	Emulated	Yes
CDMU Hardware	Functional	Yes
ACC Processor and Software	Emulated	Yes
ACC Hardware	Functional	No
Attitude Sensors (Sun Sensors, Gyros)	Functional	Yes
Actuators (Reaction Wheel, Thrusters)	Functional	Yes
Star Tracker	Functional	No
Orbit and Environment	Functional	Yes
Dynamics	Functional	Yes
Power Generation and Distribution	Functional	Yes
Thermal	Functional	No
RGS	Functional and Emulated	Yes
EPIC-MOS	Functional and Emulated	Yes
EPIC-pn	Functional and Emulated	Yes
OM	Functional and Emulated	Yes
EPIC Radiation Monitor	Functional	
TCE	Functional	Yes
TMP4	Functional	Yes

The high-fidelity modelling requirement resulted in numerous technical challenges, in particular:
- simulator design as close as possible to the satellite and instrument design
- high number of software emulators and its impact on the computer resources
- large number of possible configurations.

As for any major software development, the underlying software infrastructure and development environment were also key factors. Ultimately, all of these challenges had to be addressed and solutions found which would ensure the fidelity of the modelling without drastically impacting schedule and budget.

Schedule

The development schedule for the XMM simulator allows just 2.5 years from User Requirements Review to delivery of the complete simulator, which is a very tight schedule indeed for such a large software project. It is made even tougher by the need to incorporate spacecraft design changes arising during the development.

Incremental deliveries of the simulator have to be released at Launch–11 months (launch baselined for 2 August 1999), Launch–10 months, Launch–8 months and Launch–5 months to allow time for the validation of procedures and the simulation campaign. There is little flexibility within the schedule to allow for satellite design changes, major problems, etc.

The XMM simulator schedule (Fig. 3) is dependant on external information being provided to the development team as Customer Furnished Items (CFIs). A set of CFIs was identified for each simulator subsystem, with due dates 2 weeks prior to the start of work on the subsystem itself. This allowed the satellite design to be followed accurately and guaranteed that the most up-to-date data (documentation, database or software) would be used.

A major CFI for the simulator development is the Prime-Contractor-delivered Satellite Database (SDB). It is common to most of the work packages since it defines the telecommand and telemetry formats and contents. It is also used to generate the configuration files for the On-Board Data Handling (OBDH) model. Like the rest of the satellite design, the SDB is continually

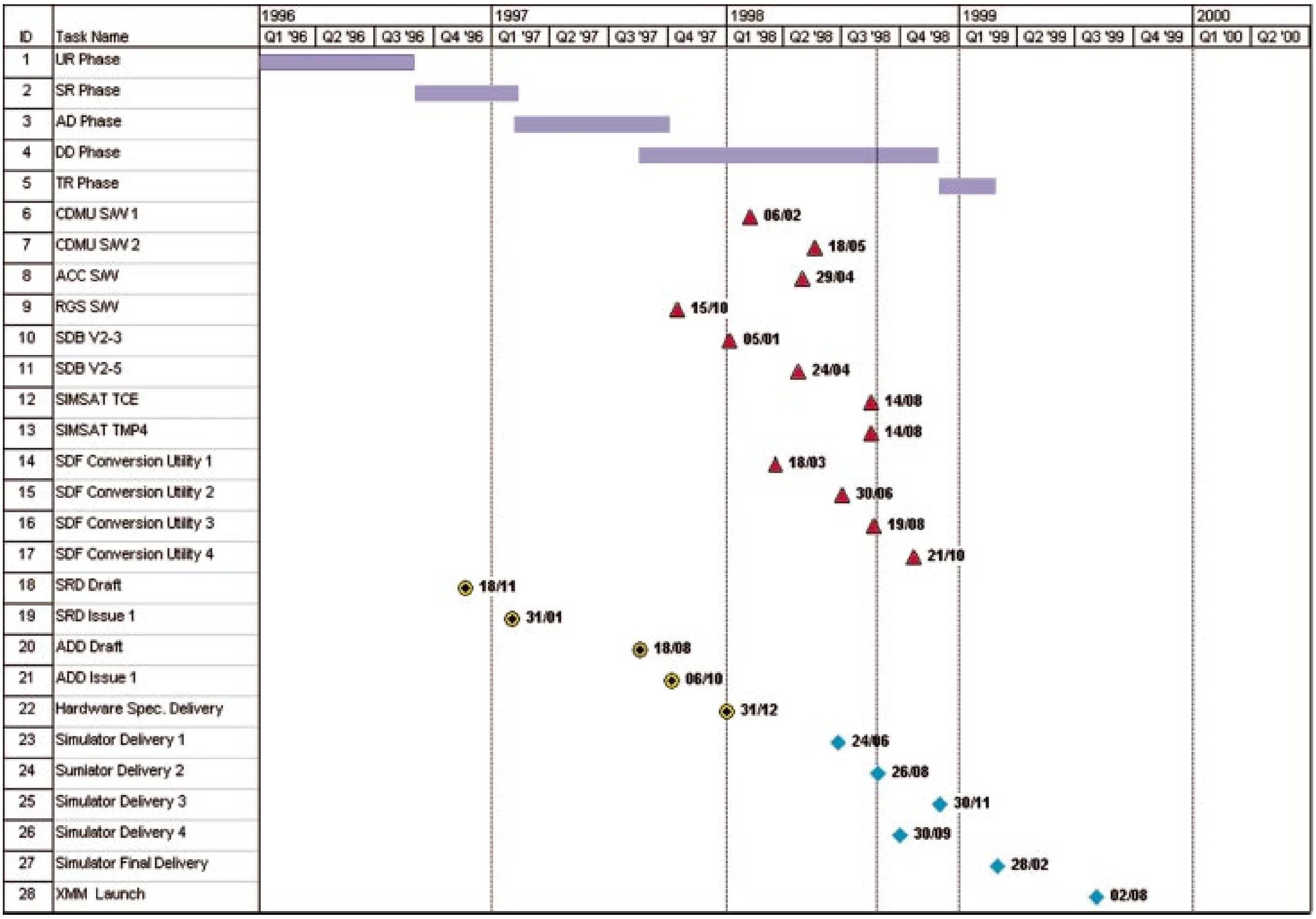

Figure 3. Schedule for the XMM simulator's development

changing, and will continue to do so up to and beyond the launch. The design approach, as presented later, has allowed the impact of these SDB changes on the design of the XMM simulator to be minimised.

Following the spacecraft design

Despite the fact that the XMM simulator was to be built under a firm fixed-price contract, its development had to be performed following a 'design-to-design' policy whereby the simulator had to shadow the 'evolving design' of the real XMM satellite. This meant no fixed baseline, which further increased the complexity of the development from both the technical and managerial viewpoints. The design baseline has been managed via a controlled set of satellite design information sources provided by ESOC as carefully selected and monitored CFIs. The Simulation Section at ESOC has been the focal point for the flow of information between the XMM Project and the XMM simulator development team. In order for the design-to-design policy to work, this information flow had to be maintained throughout the simulator and XMM satellite development efforts.

Payload modelling

The accurate modelling required for the XMM payload is a good example of the challenges implied by the above policy. The very complex XMM instruments are being developed by scientific institutes, under extreme time pressure. Consequently, immediate access to design information has not always been possible. The simulator development team has therefore had to make assumptions concerning the designs of the different instruments. Because of the latter's uniqueness, these assumptions have had to be carefully validated to ensure that the simulator design remained in line with that of the instruments. As it was not possible to overly divert the instrument Principal Investigators from their own critical development work, special Workshops were organised for each instrument, bringing together the scientists, XMM project staff and the simulator development team.

Science data files

The baseline for the XMM simulator was to play back science data recorded from the actual instruments. Early in 1997, however, it was recognised that science data covering all operational modes would not be available from ground testing. An alternative source of data had therefore to be identified. The Science Simulator (SciSIM) developed by ESA Space Science Department seemed a good candidate, although the data (Observation Data File - ODF) that it generated would need to be modified for compatibility with the architecture and design baseline foreseen for the XMM simulator. An offline utility (XMM Science Data File Conversion Utility) was developed to convert ODF into Pre-IC data files usable by the XMM simulator. This task had to be performed in parallel with the development of the XMM simulator, and was defined to fit in with the various XMM simulator deliveries. The XMM simulator development team actively participated in the review and acceptance of the Science Data File conversion utility.

Emulation and computer resources

The XMM simulator has to cope with three software emulations of an MA31750 processor implementing the MIL1750A instruction-set standard. Software emulations had already been used in previous ESOC simulators, but XMM was the first simulator to require three emulators running in parallel (CDMU, ACC and one of the instruments). In any simulator, the most demanding computer resource by far is the emulator. Early indications based on previous projects showed that running three emulators could be feasible on a high-end DEC Alpha mono-processor workstation. Since there was a small risk involved with the approach, a detailed performance analysis was made during the architectural design phase.

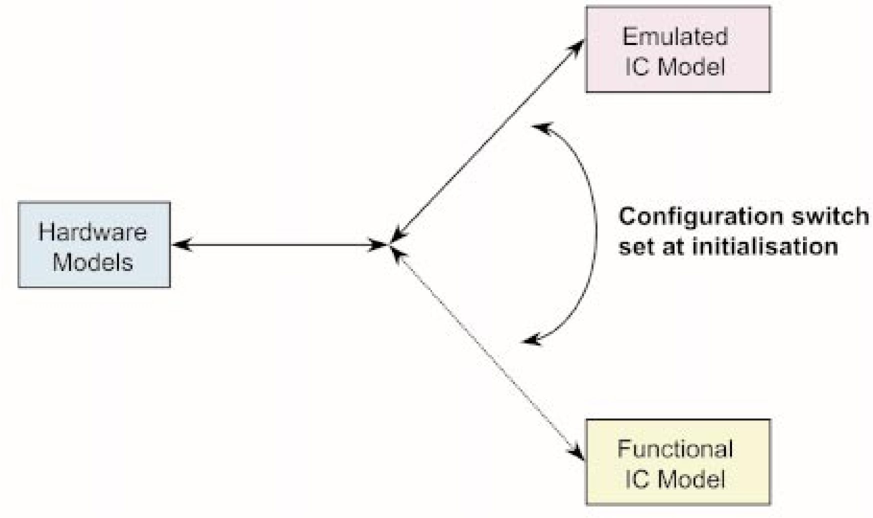

Figure 4. Instrument configuration selection

That performance analysis revealed that the MA31750 processor used by XMM was significantly more powerful and therefore more demanding in terms of computer resources than the 1750 and 31750 processors used on previous satellites. Computer resource projections were made for the different XMM onboard software items to be emulated, taking into account their characteristics (processors, clock speed, wait states), to obtain the projected CPU requirement of the XMM simulator. This showed that running three emulators would be extremely marginal on a mono-processor workstation. All of ESOC's earlier simulators had been targeted to mono-processor workstations and SIMSAT had never been used or tested in a multi-processor/multi-thread environment. An analysis carried out to identify potential problems, and to validate the core of the simulator architecture, concluded that a multi-thread/multi-processor architecture was indeed feasible and would not require any modifications to SIMSAT. The choice was therefore made to adopt this architecture and to run the XMM simulator on a DEC Alpha dual-processor workstation (DEC AXP 1200).

Simulator configurations

The simulator configurations define whether a functional or an emulated model is to be configured for each of the instruments. The XMM simulator has seven possible configurations, which can be conveniently selected using the SIMSAT MMI (man/machine interface). A configuration 'switch' in the instrument models is then used to select either the functional or emulated processor model (Fig. 4).

New ground and generic models

The XMM simulator is the first user of the new SIMSAT TCE and TMP4 ground models developed in parallel with the simulator. These are the first SIMSAT ground models to implement the ESA telemetry and telecommand packet standard. Due to their size and complexity, their delivery date was close to that of the XMM simulator, leaving very little time for integration and end-to-end testing. Fortunately, with SIMSAT the interface between spacecraft models and ground models is standardised. Moreover, the XMM development team was involved in the requirements reviews for these models, which helped to avoid incorrect assumptions being made on either side.

Architecture of the XMM simulator

The general architecture of the XMM simulator is shown in Figures 5 and 6. Its main characteristics are as follows:

- The breakdown into simulator subsystems reflects the satellite subsystems, for example RF, OBDH, power, thermal, etc. Interfaces between subsystems in the simulator then become identifiable against the actual satellite design, interface documents, etc.
- The interfaces between subsystems are minimised to aid design and ultimately simplify integration.
- Each subsystem can be viewed as a 'module'with an OBDH bus connection point, power network connection points, and thermal connection points. This is an

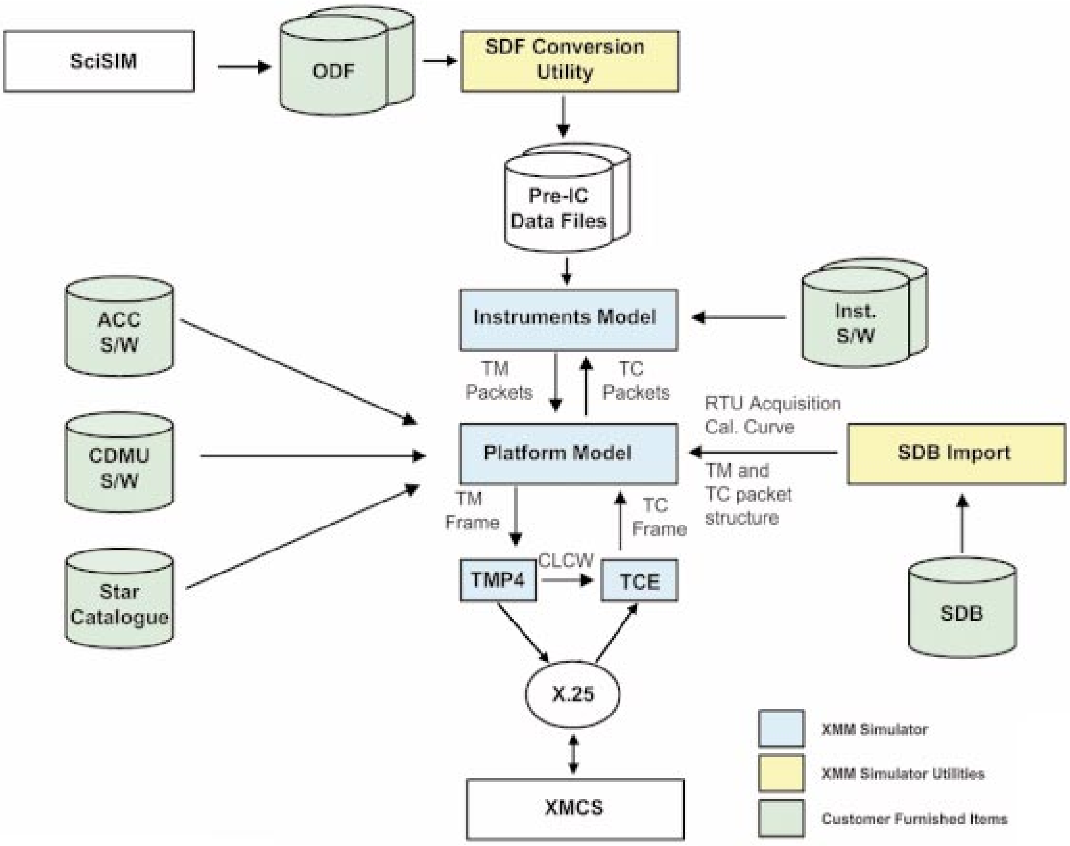

Figure 5. XMM simulator interfaces

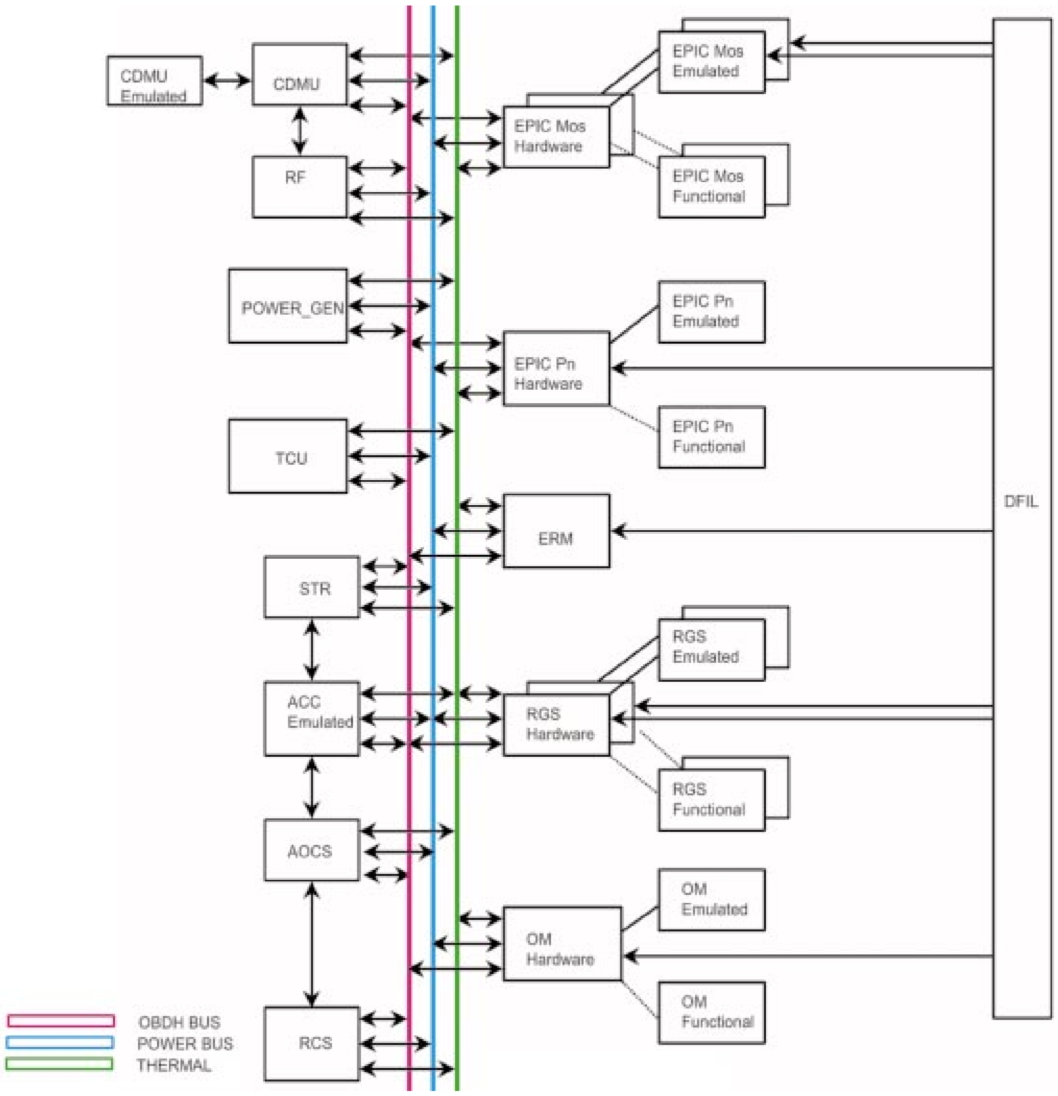

Figure 6. XMM simulator architecture

important feature since it allows for incremental deliveries of the simulator, each delivery adding new subsystems and functionality. Integrating a new subsystem into the simulator within the defined architecture ‘only’ involves connecting it to the OBDH bus, power network and thermal network.

- The instrument models are configurable as emulated or functional models. For commonality and efficiency reasons, both emulated and functional models use the same memory models, and access the hardware models through the same I/O interface models.
- The OBDH Remote Terminal Unit (RTU) model is initialised from files auto-generated from the Satellite Database (SDB). This file sets up the RTU commands, acquisitions and calibration curves used during analogue-to-digital conversions. This is an important feature since it minimises the impact of SDB changes on the OBDH models. Most changes in the contents of the SDB with respect to OBDH bus traffic only require the re-generation of the configuration files and then start up of the simulator. This has obvious time benefits not only during simulator development, but also for simulator maintenance.
- The XMM simulator has a telemetry decoder facility, which processes telemetry packets produced by the CDMU system. Analogue telemetry parameters are displayed in both engineering and raw format. The telemetry-decoder facility allows simulator operation without the XMM Control System, which is extremely useful during the simulator’s system and acceptance testing. Like the OBDH RTU model, it uses configuration files auto-generated from the SDB, with the same benefits.

Current status

The first XMM simulator delivery was made at the end of June 1998, and the second at the end of August. The latter, which at the time of writing is still under acceptance testing, consists of a full platform model, including emulation of both the CDMU and ACC software. It currently runs on the target workstation in a mono-processor configuration. The third delivery, which will add one instrument, namely RGS, and implement the multi-processor architecture, is under final coding and testing and will be released at the end of September. The complete simulator will be released at the end of November 1998. A final delivery, incorporating corrections to any errors found during the user acceptance of the simulator, will take place at the end of February 1999.

Conclusion

The XMM simulator is an ambitious project due to the combination of the high fidelity of the modelling required and the very tight implementation schedule. To ensure that its design would closely mirror that of the satellite, the Customer Furnished Items that have been used in the simulator’s development have been selected and monitored with considerable care. The simulator’s architecture has been designed to take full advantage of the DEC Alpha multi-processor workstation in order to support the high number of configurations required. Maximum use was made of the ESOC infrastructure and expertise in order to cope with the strict schedule and budget constraints, and in seeking appropriate solutions to each of the challenges that would satisfy all parties. In the event, the very good co-operation between all parties involved - the XMM Project, Science and Operations Teams, the ESOC Simulation Section, and XMM Simulator Development Team - has been one of the key elements in this development project’s success. esa