Simulators around BASILES

S. Salas Solano(1), A. Strzepek(1), P. Landrodie(1), J. Marigo(1), F. Manon(1), N. Rousse(1), M.F. Del Castillo(1)
R. Atori (2), F. Quartier(2), S. Deschamps(2), P. Verhoyen(2), M. Joubert(2), H.T. Pham(2)

(1) Centre National d'Etudes Spatiales (CNES)
18, Avenue Edouard Belin
31401 Toulouse Cedex 4 – France
silvia.salas@cnes.fr

(2) Spacebel
Ildefons Vandammestraat 7
Hoeilaart Office Center
1560 Hoeilaart – Belgium
rachid.atori@spacebel.be

ABSTRACT

Simulators have become essential for space systems; from design to operations phases many different simulators are used. However, the requirements of each of these simulators are very distinct. In order to centralize CNES (Centre National d'Études Spatiales) knowledge on simulation topics and to reduce costs, efforts have been focused on the development of one single platform for all sorts of simulators and test beds; this platform is BASILES (Base d'Applications pour SImulateurs et Logiciels d'Étude de Systèmes complexes). This paper will describe the different use cases of BASILES, it will be focused on the challenges to overcome and the solutions adopted overviewing different space programs, inside CNES but also outside.

INTRODUCTION

Simulation is a key element in the validation of space programs. It is present in all phases of the lifecycle of a project. For this reason, the Centre National d'Etudes Spatiales (CNES), and in particular the simulation department, has since several years now the goal of promoting model and simulation reuse among space programs and among the different simulators that are created during the lifecycle of a project. To reach this aim, a common simulation platform named BASILES (Base d'Applications pour SImulateurs et Logiciels d'Étude de Systèmes complexes) has been implemented. More than a simple software tool, BASILES provides a methodology and a standard for CNES simulation.

Nowadays BASILES has achieved a good level of maturity; numerous kinds of simulators have been developed based on this platform. Main applications on CNES side are the *Study Simulators* for prototyping and algorithms definition as well as the *Training Operations and Maintenance Simulator* (TOMS) for operations training and ground segment qualification. Furthermore, it has been used outside CNES in other simulation contexts that are detailed here after.

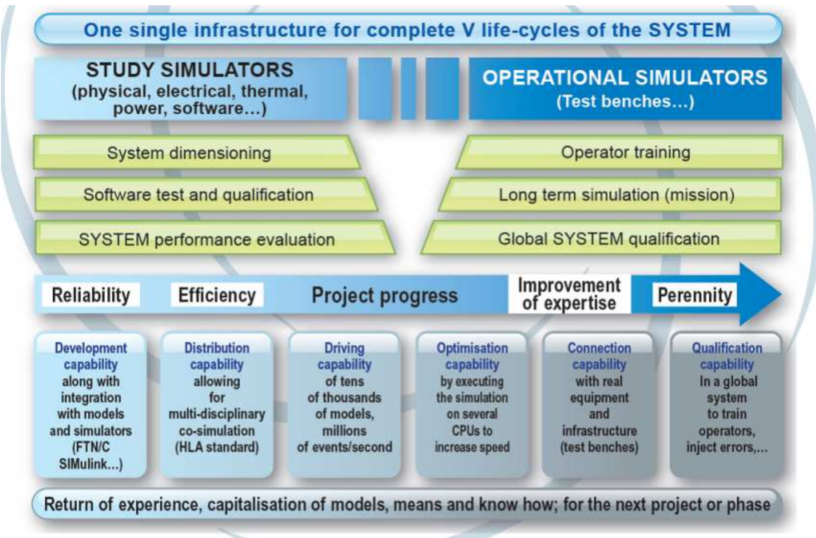

Figure 1: BASILES main use cases

BASILES

BASILES is an open software *Simulation Framework* which allows representing complex systems within discrete event simulation. It comprises:

- A runtime environment (*BASILES kernel*) in charge of time and schedule handling, logger service, integrators, processor emulator management, distributed simulation treatment, etc.
- A *Toolkit* to help in the design, configuration, execution and results analysis of simulators.
- Other specific features such as *HLA (High Level Architecture) Service* for simulation distribution providing the possibility of models communicating when executed in different BASILES kernels or *Simulink wrapper* in order to reuse in BASILES models developed through Matlab/Simulink.

BASILES core is in C/C++, the scripting language for test execution is Tcl and MMIs are in tk. The modelling framework allows developing models in different computer languages. The operating system is Linux; however it is possible to use it from other operating systems using a virtual machine.

BASILES infrastructure offers a "standard" for the simulation development and execution. It defines the type of interfaces, the way in which models can be connected between them and with the kernel, the different services available in the kernel, the mode in which the simulator can be configured, etc. From now on, it will be referred as BASILES standard even if it is not an official standard. The Toolkit mentioned above is based on this standard.

From the very beginning, one of the goals was the capitalisation and the reuse. In fact, BASILES is based on previous CNES simulation infrastructures (mainly SINUS, PRESTO and MACSIM), assembling and improving features. Based on previous feedback, BASILES was conceived in such a way that it could meet the requirements of two kinds of user's profiles: *novice user* and *advanced user*. For the first category, it was important to free them from programming language and BASILES software specificities. Thus, all functionalities are accessible by MMI and by formatted ascii files and there are code auto-generation facilities hiding interfaces with BASILES kernel (the interfaces between the models and the kernel are generated by BASILES toolkit during the compilation); models developed in such a way are called *Elementary Model*. However, a pattern must be respected which limits the software operations. In the other hand, for the advanced users, almost everything is possible; the user can go to a lower level in the architecture to control the scheduling or other main services of the simulation. He has also a scripting language in order to interact with the simulation. However he shall manage all interfaces with BASILES kernel. This kind of model is called *Specialized Model*. Even so, the user can start building an elementary model and after "specialize" it.

Another objective was to focus on the core of the simulation activities, externalising functionalities that could be potentially shared with other disciplines. BASILES would just provide the interfaces to be able to interact with these external tools. This is the case of VTS (Visualization Tool for Space Data in 2 and 3-Dimensions), whose first beginnings were part of BASILES, as well as PrestoPlot, a tool to display and manipulate graphs from sets of data in 2-Dimensions. Both tools are at present widely used in multiple contexts and for BASILES projects they provide an invaluable help to examine the simulation in real or differed time.

In addition to a simulation framework, BASILES is also a *model library* in order to promote model reuse from one project to another (even inside the same project) and to share and perpetuate CNES expertise. This library includes models in Fortran (for historic reasons), C, C++ and Java. All of them are executable in BASILES kernel.

BASILES AND SMP2

As it has been said before, BASILES has its own standard for the simulator development and execution. This standard differs from the SMP2 standard (Simulation Model Portability, version 1.2). However, in order to be able to reuse models developed by different partners, a software component has been developed in BASILES to provide adaptation of the SMP2 interfaces to the BASILES interfaces. The role of this component is to hide SMP2 specifics from the BASILES infrastructure. This component is called *SMP Service*.

SMP Service is a BASILES run time software component in charge of:

- Mapping equivalent concepts between BASILES and SMP2.
- Provision of SMP2 infrastructure services which are not available in native BASILES (e.g. Event Manager).
- Handling of SMP2 artefacts used to load and initialise a simulation (catalogue, assemblies, etc).

The SMP Service architecture overview is illustrated here below:

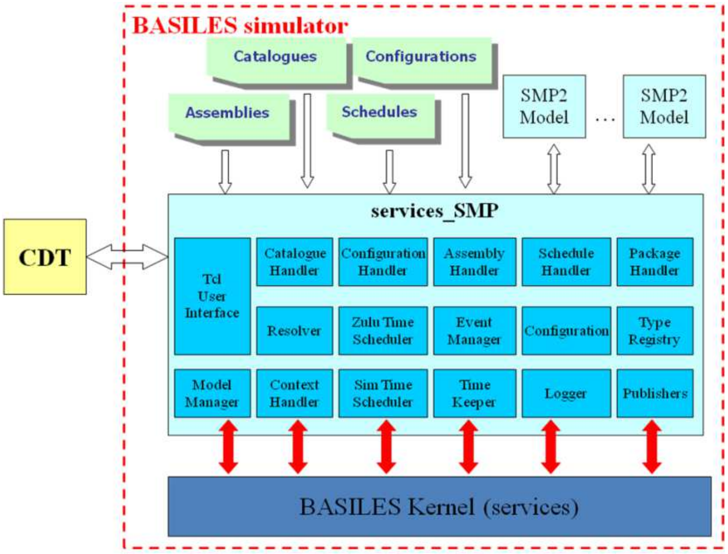

Figure 2: SMP Service in BASILES

BASILES does not include tools to support the design of SMP2 models or the generation of any of SMP2 artefacts. For this purpose, CNES must use ESA tools like UMF (Universal Modelling Framework).

SMP service is compliant with SMP2. Additionally, it also supports two extensions of the protocol:

- Support of the E-TM-40-07 Configuration file.
- Automatic data propagation.

The second point is really important when working with field links, otherwise the simulation run time configuration becomes too much heavy and almost unrealizable. The objective is to automatically propagate the value of an output field to the connected input fields. This will allow, for instance, insuring that a receiver model can have the latest value of a field when triggered via event or interface links.

To cope with the automatic data propagation, BASILES uses a modified Model Development Kit, which is called *MdkDP*. It implements the classic SMP2 interfaces and an SMP2 interfaces extension, which we call ExtDP. This extension provides the needed interfaces to support automatic data propagation. It is the intention of the CNES to submit and present this extension for consideration in the SMP ECSS WG. The principle of this MdkDP consists in making all SMP2 model fields C++ objects typed by C++ classes (instead of simple fields of type Smp::Float64, Smp::Int32, as in SMP2). The class will encapsulate all functionalities that allow an automatic data propagation of the field when updated (storage of connected input fields and updating notification).

Mixed simulations with SMP2 models and BASILES native models are possible with some limitations. In fact, BASILES models may be present and running in parallel with SMP2 models but connections between both are not yet feasible.

BASILES AND ISIS

ISIS (Initiative for Space Innovative Standards) is a set of specification documents established by CNES, Airbus DS, and Thales Alenia Space, with the goal of rationalizing space housekeeping services, from on-board equipment to in-orbit operations, and with the ultimate goal of decreasing overall mission costs. Unlike previous product lines, ISIS objective is not to create a new product, but to ease the convergence of existing industrial product lines towards something more reusable, and fitted to both CNES and prime contractors customers' needs (institutional and export).

From a simulation point of view, ISIS defines user requirements for the TOMS, these requirements are in line with propositions made by SSRA (Spacecraft Simulation Reference Architecture) concerning simulator architecture and the conformance with SMP2 standard amended by the automatic dataflow propagation is imposed. Indeed, ISIS "completes" SMP2. SMP2 defines the kind of computer interfaces between models, but to really exchange models is necessary to go further and to specify the "functional" interface. For this purpose, two simulator specifications have been carried out around the TOMS (see [1] and [2] for more details). The first specification is focused on the following kind of requirements: functional, architecture and design, operational and management. The second document is an interface specification. Its purpose is to allow exchanges of simulation models, models data and reference runs between the different partners via well-defined interfaces. Following interfaces are covered:

- System Interfaces: electrical interfaces between the OBC model and the equipment models.
- External Interfaces: between the spacecraft simulator and the external world (TM/TC).

- Physical models interfaces: spacecraft environment and dynamics model interfaces.
- Central solver interfaces.
- Models and Models Data exchange Interfaces: specification of the exchange of models and associated data (including reference runs).

Considering the existing inter-model communication mechanisms in SMP2, two different approaches have been defined for the System Interfaces:

- Use of Event and/or Field Links (see [3] for more details about this SMP2 mechanism). This approach is mainly used in the simulators developed by Thales Alenia Space for CNES; their simulation platform is more adapted to this kind of interface.
- Use of Interface Links (see [3] for more details about this SMP2 mechanism). This approach is mainly used in the simulators developed by Airbus DS for CNES; their simulation platform is more adapted to this kind of interface.

In practice, a good level of convergence has been achieved with the partners for the System Interfaces. The rest of points are more subject to discussion and must be customized for each new mission.

Taking into account the importance and convergence on System Interfaces, there are works in-progress to enlarge BASILES model library to include models for each kind of System Interface as well as example models to trigger and consume each interface.

SIMULATORS AROUND BASILES INSIDE CNES

As it has been said before, main BASILES application inside CNES are the Study Simulators and the TOMS.

In the analysis and design phase, BASILES has been used to implement mathematical algorithms in specific domains such as Attitude Orbit Control System, Electrical, Thermal, etc. In this context, the models have been developed by the own subsystem experts and the process is really iterative. The challenge was to provide to the architects a user-friendly tool. Another goal was to reuse these models in later project phases.

Different demonstrators have also been created to analyse “new” concepts or components like the satellite on-board autonomy or the distributed simulation for satellites in formation flight.

BASILES has also been used with hardware in the loop, for example to validate on-board software.

However, main BASILES application remains the TOMS; used for the ground system qualification and operations. Some of these simulators are SMP2 based and others are BASILES based. The SMP2 approach is required when working with external partners.

Simulators Based on BASILES Standard

Most of CNES Study Simulators are developed using native BASILES. In fact, we master it better than SMP2 standard and we consider it easier to implement, above all thanks to the notion of Elementary Model. This level of model is generally sufficient for Study Simulators and sometimes also for TOMS. In addition, our entire Toolkit is oriented to this standard, therefore there are much more helping tools. To develop complying with SMP2, external tools not mastered internally must be used.

In addition, our model library is basically composed of BASILES native models. It is a normal choice to try to reuse them to reduce costs and development time. Each new simulator contributes to enlarge BASILES model library; on one hand, improving existing environment models and on the other hand adding new specific models. The existing environment and orbit models are shared by most of our missions.

Next sub-sections describe in more detail some of the applications based on BASILES standard.

CSO AOCS simulator

CSO program is a French military observation satellite for image acquisition and processing. From the B phase, it is important to have an environment, dynamics and AOCS equipments models as much representative and accurate as possible.

For AOCS study activities, it is essential to carry out temporal simulation of spacecraft in order to analyse, validate and improve the algorithms of on-board attitude estimation and attitude control. These activities allow studying and analysing the performances of a space mission. Simulators have to be very accurate at dynamics and orbit level. To answer to these requirements, the AOCS engineers need to have a simulator with the following features:

- Model development/modification must be simple and quick (basic computing knowledge shall be required).
- User-friendly tools.
- Execution as fast as possible.
- Hundreds of runs with sweeping variables.
- Results analysis in differed time.
- Reusable models (study simulators tend to be developed in an incremental and iterative way starting with few and relatively basic models).

As BASILES framework answers to all these requests in terms of model development, simulation at high frequency (128 Hz) and representativeness and as BASILES would be used for the CSO TOMS, it was a natural choice to take BASILES. CSO AOCS simulator is based on native BASILES and is exclusively composed of Elementary Models.

It is important to highlight that this simulator was developed by the AOCS department; the simulation department was just in support in case of problems using BASILES. At the beginning there was a tighter collaboration between both departments but the AOCS team quickly took over BASILES functionalities and they became completely autonomous.

Thanks to their feedback some improvements were made in the feature in charge of the generation/management of hundreds of tests sweeping a group of specific variables to turn it completely operational. The batch option relative to the simulator execution was also included by their request.

MEDON and SNOB

In the context of NOSYCA (New Operational SYstem for the Control of Aerostats) a hybrid simulator and a full numerical simulator have been developed to train the operators and to validate the new balloon system.

MEDON is the hybrid simulator used for the on-board software validation as well as for the ground segment qualification in NOSYCA program. It is the first TOMS using BASILES as well as the first BASILES application with hardware in the loop. Concerning simulation, NOSYCA project was looking for:

- A high level of representativeness with respect to the real system.
- Means to validate the real on-board software.
- Means to qualify the ground segment.
- Implementation of specific situations and/or failures hard or impossible to observe in nominal configuration or without damaging equipment.

The figure below shows the NOSYCA system and the MEDON architecture.

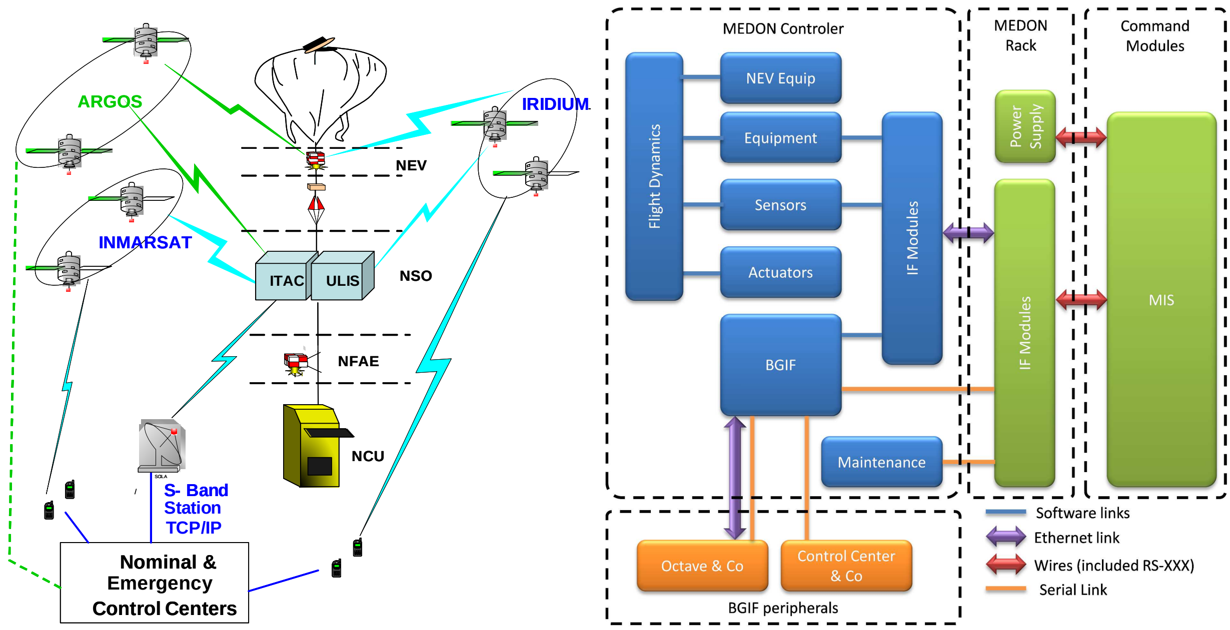

Figure 3: NOSYCA system and MEDON architecture

MEDON is composed of:

- The balloon flight dynamics model which takes into account the wind profiles and the environment.
- The on-board equipments numerical models (actuators, sensors and communication means), representative from a functional point of view as well as regarding their communication with the OBC.

- MIS hardware (particular configuration of the OBC) and real on-board software.
- A FrontEnd in charge of the communication between numerical and hardware components.

MEDON numerical models implement BASILES standard; all of them are Elementary Models which has allowed their development in record time. MEDON is executed from BASILES framework generic MMI, none specificity has been necessary.

Taking into account that it was the first operational application, some improvements were necessary concerning interfaces with other systems. This is the case of the Calibration Service in order to manage polynomials and tabs files. Some amelioration must still be done for the interface with the project data base, the contexts management remains a little bit arduous. It must be emphasised that MEDON is operated by NOSYCA project team; the simulation department just provide support in case of problems. Therefore, the functionalities must be intuitive and user-friendly. For this last point, there is work in-progress to improve GUI synoptics, main interface for the operators to inject failures.

Nevertheless main difficulties have come from the hardware component. Some bugs have been fixed concerning the context saving/restoration as well as PPS time synchronisation. However some improvements might still be performed if one day equipment with hard real time constraints needs to be connected to a BASILES simulator. In fact the time is guaranteed with respect to a fix reference which can be configured; this approximation could be not sufficient in some cases. There are solutions like an external controller or clock which will be more suitable to manage the time in a more accurate way.

In a second time, NOSYCA project realised that it would be useful to have a simulator that is easy to transport and to deploy in order to train the operators and to perform operational qualification tests during the balloon campaigns; these campaigns take place all over the world. In this context, the best solution to free them from hardware constraints was to develop a full numerical simulator; this simulator is SNOB. SNOB has reused as much of models from MEDON as possible, however some equipment models have had to be redeveloped as they have not conceived initially to this kind of simulation. It would have been necessary to separate the functional part from the system interface part, in order to share the functional part between MEDON and SNOB; the system interfaces remain specific to each kind of simulator. Additionally, in SNOB the real on-board software has been replaced by a functional software developed as Elementary Models. SNOB has progressively replaced MEDON for operational qualification; nevertheless MEDON is irreplaceable for the validation of the real on-board software.

The NOSYCA team is really satisfied of MEDON and SNOB behaviour. The best recognition is that a new project for another kind of balloon is on-going and will use the same approach as NOSYCA for simulation concerns. A new full numerical simulation and a new hybrid simulator will be developed based on the BASILES platform reusing as much of models as possible from MEDON and SNOB.

ARGOS

ARGOS program is a satellite-based system which collects, processes and disseminates environmental data from fixed and mobile platforms worldwide. Several ARGOS instrument payloads are already deployed on various low orbit satellites such as NOAA, METOP and SARAL. Its wide and increasing use generates the need for additional and higher performance of ARGOS instruments. A new generation of the ARGOS instrument and beacons is under development. By consequence, a new simulator was needed to validate the new designs and to ensure the global system performance.

The main challenge for this kind of simulator was to achieve a good level of performance simulating more than 40000 objects (e.g. 7 satellites, 10 ground stations, 40000 beacons). Overall simulation performance needs to exceed fifty times real-time to allow the execution of test scenarios spanning several days in a reasonable time. In some cases, nearly half a million events per second need be processed.

The starting point for this simulator was native BASILES using Elementary and Specialized Models. However some improvements have been necessary at scheduler level to cope with the great number of events. Before ARGOS simulator, BASILES scheduler was based on a "list" of events. Events were ordered by date, next by priority and finally by posting order. However, this architecture does not allow achieving the performance imposed by ARGOS project where at any given moment there are more than 10000 events in the scheduler; the events themselves are low time-consuming, however as there are a lot of events, it is the time passed to insert/look for them which becomes penalizing. To overcome this challenge a new scheduler has been conceived. This new scheduler is based on a "map" instead of a list. In the case of a map, a tree is built, events with same date and priority are ranged in the same branch, and each leaf is an event; this new architecture reduces the time to insert a new event when there are already a lot of them in the scheduler. The map structure is suitable for simulators like ARGOS (more than 10000 events at a given instant). However it is not optimal for simulators with a few tens of events in the scheduler at a given moment; in this context the map scheduler will degrade the performances for the event insertion/research (this is the case for most of our Study Simulators as well as for the TOMS). For this last situation, the scheduler based on a list remains more adequate.

Consequently, both kinds of schedulers are at present available in BASILES platform, the user can choose between them in the simulator configuration; by default, it is the scheduler based on a list which will be used.

The new scheduler has implied a lot of modifications at Monitoring Service level. We have taken advantage of this situation to rebuild the Monitoring and make it more modular. The Monitoring Service has one main scheduler and it can contain several secondary schedulers. It can also include external schedulers such as the ones of processor emulators. In addition, the Monitoring Service comprises complementary modules in charge of event logging, event statistical analysis …

In addition to improvements at scheduler level, such a simulator needs also special tools to create the thousands of models, to follow and control the execution of the test scenario and to collect the test results in human readable forms, such as maps, histograms, statistics ... Specific tools and GUI have been developed for this purpose. BASILES MMI was too much generic.

Simulators Based on SMP2 Standard

Most of CNES TOMS are ISIS compliant and thus SMP2 compliant. The TOMS is part of the ground segment and it is under CNES responsibility. The majority of TOMS models are delivered by the satellite contractor. In fact the prime contractors also need numerical models for the SVF (Software Validation Facilities) and the AIT (Assembly, Integration and Test) facilities which are under their responsibility; the CNES choice is to integrate these same models in the TOMS. Therefore, the numerical models must fulfil the requirements of all these kinds of simulators, TOMS usually implies additional error injection cases. The integration of the TOMS also includes some CNES models developed internally, such as physical models (dynamics and environment) and models in interface with the ground segment. For the integration, CNES is supported by an integrator contractor (Spacebel at present). The SVF and the AIT are performed in the satellite contractor’s simulation platform, while the TOMS is executed in BASILES runtime environment. This means that the only way to exchange models is to agree with a standard independent of the simulation platform; consequently, ISIS and SMP2 have a main role in this context.

CSO TOMS

The CSO TOMS is the first simulator at CNES based on ISIS and SMP2. Airbus DS is the prime satellite contractor of CSO. In this context, they have delivered standalone SMP2 models to CNES (OBC, power, actuators and sensor models). They have developed the models using their simulation platform SimTG and they have validated them in their platform but also in BASILES infrastructure. These models have been integrated by CNES and Spacebel team into the TOMS.

Taking into account that it was the first operational experience at CNES based on SMP2, some updates have been necessary at BASILES and SimTG level with respect to the implementation of SMP2.

One of the particularities of the models delivered by Airbus DS is that only the top level models are integrated using SMP2 mechanisms; the sub-models used dedicated means based on SimTG. In addition, the top models always use the SMP2 software interface when there is an asynchronous behaviour to be simulated (vs SMP2 event and field approach). The fact that only top level models use SMP2 mechanisms has some advantages and inconveniences. On one hand, this approach is probably more efficient in terms of initialisation time and runtime performance, it also reduces the complexity at configuration level (a part of the assembly is hard-coded in the sub-model code). On the other hand, this method reduces the visibility of sub-models when using other infrastructure than SimTG (e.g. connections at sub-model level are not published). To mitigate this limitation, Airbus DS delivers complementary SMP2 services so as to be able to overload a variable at sub-model level or to connect an external model to a port at sub-model level.

The CSO experience feedback shows that the model exchange is really efficient thanks to ISIS and SMP2. However there are some limitations/differences concerning the services at infrastructure level in order to interact with the models. One additional effort is the cross validation of models between different infrastructures due to the different test languages used (e.g. Java for SimTG and Tcl for BASILES). The last point to be highlighted is that the configuration of assembly and schedules files remains really laborious.

Myriade Evolutions

Myriade Evolutions program aims at developing and qualifying a platform for the period 2015-2025 for satellites in the range of 400 kg. The project is carried out in partnership between CNES, Airbus DS, and Thales Alenia Space. It is the next generation of the Myriade product line (platform for microsatellites). This new platform will be available for scientific, defence and industrial applications. Just like the first generation, low cost, adaptability and short planning are the main drivers of the program. The principal improvements are the increase of capability in terms of mass, power and data flow but also in terms of reliability (0.5 after 7 years) and agility (30deg in 30s). Moreover, Myriade Evolutions will be compliant with ISIS.

Such a platform requires the development of ten new units that will complete the own product lines of the three partners. Nine of these ten new units imply a new simulation model (S-Band, X-Band, Reaction Wheel, Propulsion, GNSS, Mass Memory, PCDU, Battery and Solar Arrays); these new models will potentially be used in the SVF and the TOMS of the different missions. As working with external partners, the strategy is the use of ISIS and SMP2 standard; the development will be made using ESA tools and the validation will be carried out in BASILES on CNES side; the validation in our partners' platform will be carried out by them during the SVF of the missions based on Myriade Evolutions. The specifications are co-written by the partnership and the models are developed by Spacebel Belgium in the frame of a French-Belgium agreement. This time, the whole model will be SMP2; this approach has the benefit of being really generic, the models as well as the sub-models are completely accessible in any platform compliant with SMP2. However we must keep attentive to performances and configurability awkwardness.

MERLIN TOMS

MERLIN (MEthane Remote sensing Lidar missioN) is a joint French-German cooperation, dedicated to the methane monitoring at a global scale. As it is the first Myriade Evolutions application, the in-flight qualification of the new platform will be demonstrated throughout this mission.

The payload unit and its numerical model are provided by Airbus DS Germany. The platform and the numerical platform models are delivered by Airbus DS France. Models are ISIS compliant and developed using SimTG (except for the models coming from Myriade Evolutions); the validation is carried out in SimTG and also in BASILES. Concerning the TOMS integration, the context is similar to CSO except by the fact that there are three different model suppliers; however it is not the case for the SVF conducted by Airbus DS where "external" models coming from Myriade Evolutions will probably also be used (in past programs the SVF was completely carried out with internal models to Airbus DS). In this situation models previously validated in BASILES will be later integrated and validated in SimTG (in CSO the exchange was always in the other way around). This new application case will surely imply some new updates in BASILES and SimTG with respect to SMP2, but it will also show the level of maturity of both platforms regarding SMP2.

Other TOMS

Among the different activities in-progress, there will be a new TOMS at CNES for which Thales Alenia Space is developing models with their simulation platform K2 that will be later encapsulated into an SMP2 model for validation in BASILES. This will be the first operational application at CNES using models developed with K2 and compliant with SMP2. The models will also be ISIS compliant following the event & field link approach with the automatic data propagation (not in SMP2 at present). This last point is really relevant, without the automatic data propagation the simulator configuration becomes almost impossible.

In addition, in this same simulator, there will also be models coming from Airbus DS. Originally, Airbus DS and Thales Alenia Space have different approaches for SMP2 mechanisms (see previous section BASILES and ISIS). In this frame, SimTG will be updated to include a wrapper to adapt the software interface into events and fields (to be confirmed).

SIMULATORS AROUND BASILES OUTSIDE CNES

Outside CNES, BASILES is used for SVF in the MTG (Meteosat Third Generation) project, and soon for PROBA 3. To cope with the requirements of such a simulator, BASILES functionalities have been extended; this implies for the example the support of multiple gdb OBSW debuggers that can set breakpoints while all the surrounding simulator facilities such as test languages, quick views, introspection, TM/TC interfaces…, remain active. These simulators are SMP2 based and they are developed by Spacebel.

Additionally, the French General Delegation for Armament (DGA) has also used a prototype based on BASILES standard for helping in the configuration and sizing of their test devices.

Furthermore, there are on-going investigations for the use of BASILES in other application fields such as FDIR (Fault Detection, Isolation and Recovery) and reliability simulation, road traffic simulation and traffic jam management. In fact, BASILES approach is not specific to aerospace and it can be reused in other domains.

MTG SVF

For the realisation of the MTG SVF, OHB has chosen to use the SMP2 modelling standard and BASILES as the underlying simulation infrastructure.

For an effective development cost and in order to fulfil the planning constraints, it has been split into two parts:

- Simulation Framework (SF): an infrastructure that includes the simulation infrastructure (i.e. BASILES) and related simulation services and models (e.g. CCS I/F models). In addition, the SF contains the MTG SMU model, an SMP2 model which is provided by Terma and is integrated to run in BASILES by Spacebel.
- Equipment Models: SMP2 models, to be developed separately. They are meant to integrate together with the Simulation Framework (i.e. the models will in fine run on BASILES within the SF). The equipment models are developed based on the SSRA approach (an Equipment model has an Operational submodel, a Functional submodel and System IF Layer submodel).

Below is an overview of the MTG-SVF design architecture:

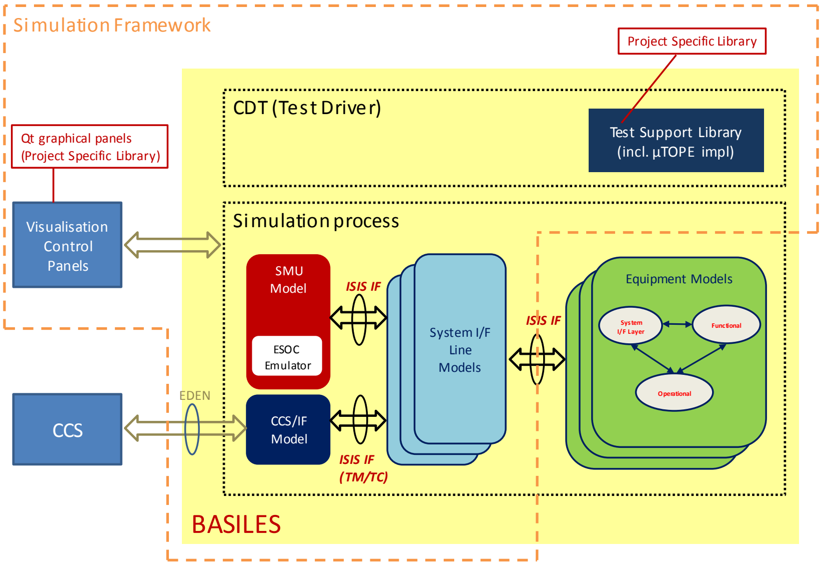

Figure 4: MTG-SVF architecture

It is to be highlighted that the connections between the OBC model (here the SMU) & the Line Models & the Equipment Models are following the ISIS standard for System Interfaces.

Strong points of using BASILES:

- Simple and lightweight simulation kernel.
- Rich user interface for test driving, management and results exploitation.
- Rich set of simulation user services, e.g. sampling, surveillance, plotting (real-time and offline) simulation variables, quick views, powerful error injection mechanisms…
- Fine integration of the emulator and the simulation schedule, which allows synchronising the emulator time line and the simulation time line (100% timing accuracy for handling the telecommands in the OBC model).
- BASILES script language is Tcl, which is designed for tests and which is the same language as in SCOS (SpaceCraft Operating System) based CCS systems. This allows reusing the CCS test scripts in the SVF simulator by providing a µTOPE implementation on the SVF.
- Non-intrusive & symbolic debugging with an external tool (e.g. GDB) & map file.

Lessons learned and points to improve:

- The µTOPE implementation using SIMPACK library is not sufficient to support all SCOS MIB (Mission Information Base) features used in MTG.
- The µTOPE implementation within BASILES can further support for real CCS µTOPE features (i.e. not all options are supported).

CEAO

The Design Test Computer-Assisted (CEAO) project of the DGA aims at offering a software tool which facilitates the definition and the pre-configuration of the test devices; the objective is to determine if a given configuration of the test devices is adequate for a specific real mission (e.g. determining the number of necessary radars).

A first prototype of the CEAO tool was carried out by DGA based on their own simulation tools and on the planet exploration open software World Wind (NASA property). However this prototype did not conform to the users' major requirements of the future CEAO tool.

DGA started looking for new solutions and BASILES appeared as a possible candidate. Most of the requirements were the same as for CNES Study Simulator; the major difference was the interface with World Wind instead of VTS.

In order to confirm this new approach, a second prototype was requested based on BASILES. Spacebel was in charge of the development of this prototype. The solution consisted of:

- Conceiving a GUI, dedicated to CEAO, to enable an easy use of the BASILES concepts.
- Keeping the World Wind application and completing it (test configuration and real-time visualization).
- Defining a general controller to manage all the elements of the CEAO tool at the same time.
- Developing BASILES Elementary Models for each element constituting the simulation.

As for ARGOS it was important to have specific GUI; BASILES one was too much generic.

During the development of the project, there were no major difficulties; the only one to be highlighted was the difference in the vocabulary between BASILES concepts and DGA ones which implied to establish a correspondence between both. The prototype demonstrated that BASILES fully meets the simulation needs of the CEAO project with our turnkey solution.

CONCLUSION

At present, BASILES is successfully used in many space programs, inside CNES but also outside. There are BASILES simulators in all project phases. However its main application remains the TOMS; study simulators tend to use Simulink more and more.

Concerning SMP2 and ISIS, they have a main role in our simulation activities, it has been demonstrated that they provide an inestimable added value. However, next working group around SMP2 should consider the different limitations addressed in this paper, above all for the automatic data propagation. In complement to SMP2, the additional standardisation effort relative to ISIS and SSRA is completely necessary to ensure an efficient model exchange. We should continue working along this line. For example, with respect to ISIS, some progress can be done in the area of the physical and central solver interfaces to really specify and fix all of them.

For the future, there is a working group around BASILES to define the next generation of the infrastructure (BASILES NG). Among the objectives, it is important to keep current modularity and multiple functionalities with the good level of performances of the kernel; however an upgrade of the technology is imposed (computer languages…) and a deeper integration between native BASILES and SMP2 is necessary, including the design phase. In addition, there are some R&D actions to try to improve the simulation performance in order to cope with the quantity and frequency increase of on-board CPUs.

ACKNOWLEDGMENTS

The author thanks whole simulation department at CNES and Spacebel for their active contributions to this work as well as BASILES users for their invaluable feedback.

REFERENCES

[1] B. Delatte, N. Rousse, "ISIS Training Operation and Maintenance Simulator (TOMS) Generic Customer Needs Specification", ISIS-SIM- TS-304-CNES version 3, September 2013.
[2] ISIS AIV Project Team, "ISIS Training Operation and Maintenance (TOMS) Interface Specification", ISIS-SIM-IF-305-CNES version 5, September 2013.
[3] "SMP 2.0 Handbook", EGOS-SIM-GEN-TN-0099 version 1.2, October 2005.