SDB NEXT – a step to virtual satellite

Pierre-Yves Schmerber, Anthony Mollier

Thales Alenia Space France

5 allée des Gabians, 06150 Cannes-La-Bocca FRANCE

Email : pierre-yves.schmerber@thalesaleniaspace.com

anthony.mollier@thalesaleniaspace.com

MODEL BASED ENGINEERING LEADS TO VIRTUAL SATELLITE MODEL

Model based engineering is spreading in all domains of the space industry. For space system design engineering, Thales Alenia Space is using the Arcadia model based method from Thales [1]. The method allows to keep consistency between the successive models used in support of the system operational analysis, the system functional and non-functional needs identification, the system logical architecture, and the system physical architecture with its product breakdown as illustrated in Fig.1. The Arcadia method, supported by the Capella open source software [2], also provides the capacity to synchronize sub-system design models with the system models, encouraging to use the same model based engineering method to design the different sub-systems. This allows to grow consistent design models for all the domains of the satellite engineering going from the system down to the different subsystems and further down to equipment and software. These models are the backbone of the virtual satellite where the information contained in the models is sufficient to prepare the space system operation, to prepare the validation tests, to build representative simulators, and to develop on board software before having the actual satellite hardware.

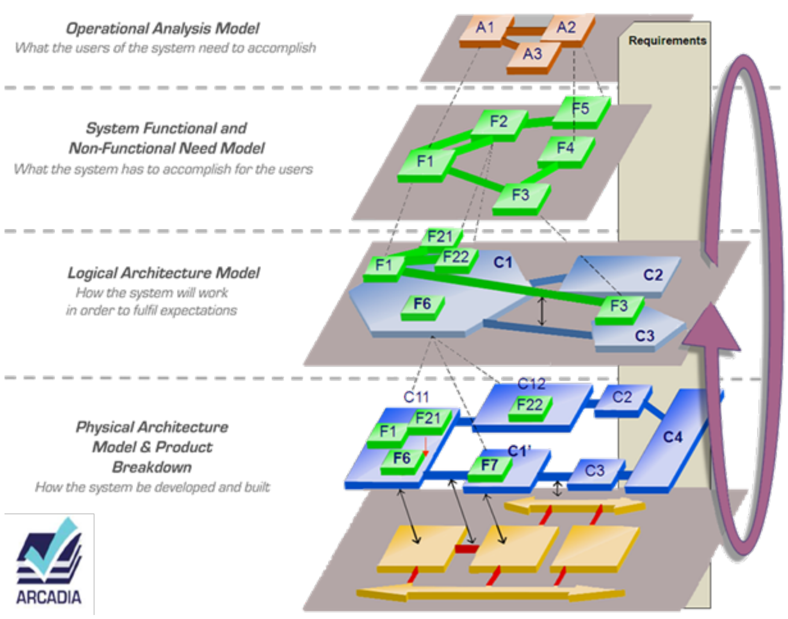

Fig.1. Model Based System Engineering with ARCADIA

However, in complement to the model based design engineering, the space system development involves additional work by all contributing stakeholders, supported by domain specific tools and models. These concurrent processes of development produce detailed technical data, that need to be shared among stakeholders to keep the system consistency. The Satellite Reference Data Base is the repository where these technical data are kept in configuration so that all stakeholders share a common reference during the satellite development and deployment lifecycle. This technical data progressively growing in maturity gives consistency to the virtual satellite and progressively improve its accuracy.

SDB NEXT SOLUTION TO HOST VIRTUAL SATELLITE DATA

The Satellite Reference Data Base tools started with the exchange of data related to the telemetry and telecommand of the satellite so that every stakeholder involved in the satellite development share a common data reference without having to recreate the data from a common paper specification every time they need it. However with the growing adoption of model based technique, the need to exchange reference technical data has increased, and existing tools have to evolve to follow the improvement of the processes followed within the different satellite domains. The Satellite Reference Data Base tools have thus to manage smoothly the arrival of new stakeholders and the growth of the referenced data concepts. In addition, the Satellite Reference Data Base is also expected to organize the technical data so that it reflects the system elements used by the design models, in order to improve traceability and reuse of technical data within a satellite product line.

To answer these new needs, Thales Alenia Space is preparing a new Satellite Reference Data Base named SDB NEXT. As represented in Fig.2, SDB NEXT is made of a kernel with the data repository and services for consistency checking, configuration control, change management, access rights management, and data maturity follow-up. This kernel is designed to manage data in building blocks, and to be extended by a meta-model that describes the technical data content along with consistency rules. This decomposition allows to isolate the capacity to evolve the referenced data concepts from the general purpose kernel. The data building blocks managed by the kernel reflect the system elements of the engineering design model, in order to keep traceability and to take into account reuse in the frame of satellite product lines. The SDB NEXT kernel is also extended by bridges that perform data transformation to adapt to the stakeholder’s specific formats used to exchange technical data, and by editors that provide stakeholder adapted man machine interface to the technical data content.

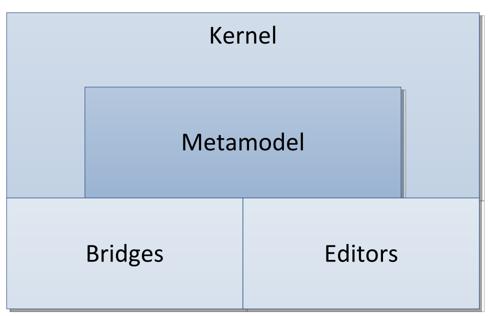

Fig.2. High level components breakdown of SDB NEXT

DATA ORGANIZATION FOLLOWS ECSS STANDARDS

Data organization in building blocks follow the space system decomposition described in the ECSS 10-23 technical memorandum, where the space system is seen as a hierarchy of system elements. Each system element can be seen under different engineering perspective, so the building block can host data coming from different engineering domains, that each have their specific model. In order to ease sharing of technical data, the kernel maintains in the building block all shared data with a common metamodel. Bridges and editors contribute to the restitution under different engineering perspectives of the building block data. The monitoring and control perspective is the traditional domain of Satellite Reference Data Base, and its data model has been standardized within the European Ground System Command Control (EGS-CC) initiative. Thus the current metamodel used in SDB NEXT conforms to the EGS-CC conceptual data model.

The process for Satellite Reference Data Base population starts from the engineering design models, that provide the backbone data organization in system elements, and already identifies the main activities, events and reporting data based on the operation concepts. On board software definition further complement the engineering models with the actual services and parameter definitions, and with mission specific on board software configuration and on board monitoring and recovery strategy. Hardware definition complement the engineering models with actual equipment interface details and actual harness connections between equipment that makes the addressing scheme of each hardware resource. On top of these models, the data handling defines the content of the telemetry and telecommand packets, while the concurrent preparation of Avionic Test Bench, Assembly, Integration and Test and Operation complement the model with EGSE and Ground Control System building blocks, and with procedures and ground monitoring definitions.

CONFIGURATION BRANCHES

Along the industrial process, the space system data evolves, and need to be properly managed in configuration. The space system data evolves along configuration branches that correspond to different level of definition for the virtual satellite as illustrated on Fig.3.

One such configuration named AsDesigned reflects the spacecraft as designed for a mission, with its evolution in time. It corresponds to the design model used as the reference design of one or several satellites in the same program. In the case of a constellation, it corresponds to the generic model.

Another configuration named AsBuilt reflects the spacecraft as built for a specific flight model or program specimen, that will have a separate evolution from the spacecraft mission design, without preventing some synchronization points in time. There are usually many AsBuilt configurations in a program, that are created from branching from the AsDesigned configuration at some point in time.

In addition, for each test campaign there is a need to manage a separate configuration reflecting the system under test rather than the target flight model, and to be able to correct technical data in this dedicated configuration to reflect test experience, with the intention to decide at the end of the test campaign the data that needs to be propagated back to the other configurations. Such configuration are named AsIs and are usually created from branching from an AsBuilt configuration, for the duration of their dedicated test campaign.

Furthermore when considering a satellite product line such as NEOSAT, additional configuration branches are needed to keep a catalog of product variants that hold the standard definitions of the product line. This allows to reflect the reuse of product line design in the reuse of technical data on the mission specific design branch. It also allows to update the product line technical data with the return of experience coming from specific missions.

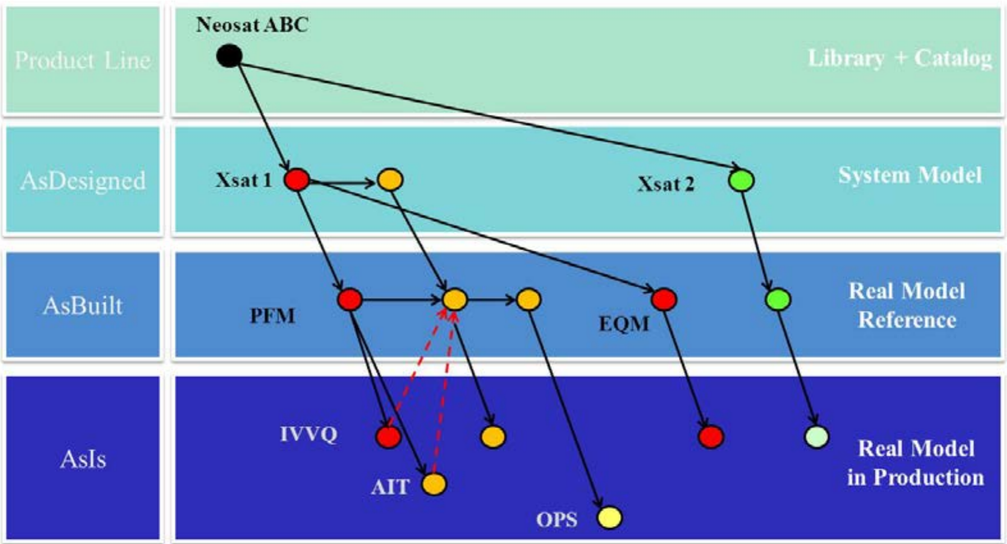

Fig.3. Satellite Configuration Branch Management

SDB NEXT is able to manage all these separate configuration branches, while easing propagation of specific data modifications between the configuration branches when decided. Each configuration branch is managed in configuration, with the identification of reference versions, and of a working version that corresponds to the data currently under preparation and not yet frozen in a reference version.

SDB NEXT data repository relies on GIT configuration management functions that have demonstrated their efficiency in the software development context, especially for the flexibility of its tag and branch management support. SDB NEXT smallest configuration item is the building block, meaning that technical data evolutions are traced to building block evolutions. In addition, to provide project wide visibility on the technical data evolutions, the SDB NEXT solution relies on the change management process hosted in the company change management tool.

EVOLUTIVITY

We have seen previously that a challenge that has to be solved by modern Satellite Reference Database is the capacity for evolution to follow the evolution of the stakeholders processes that may adopt new tools, and define new data concepts that need to be kept in referenced and exchanged with other stakeholders. Typically it has been identified that the SDB NEXT metamodel will evolve over the timeframe of a satellite product line.

So one of the most important architecture driver of SDB NEXT product is the capability to offer a system database solution which facilitates reuse of building blocks while allowing metamodel evolutions at minimal cost. The SDB NEXT kernel is designed so that metamodel enhancements will not require a complete reshape of kernel, editors and bridges concepts. The metamodel implementation is based on Ecore models from the Eclipse Modeling Framework, that allows to define the technical data concepts hosted in the building block documents.

SDB NEXT design address the evolutivity challenge with different solutions:

- A first aspect is in its capacity to provide services independent of the data content of the documents managed within the building blocks, so that it is possible to manage different documents with an impact limited to the bridges to extract and fill the shared data.

- A second aspect is its capacity to manage simultaneously product lines and space systems compliant with different data model versions, so that the transition of product line and mission specific data could occur progressively when needed.
- A third aspect is in the use of the Eclipse Modeling Framework that provide solutions to manage the impact of data model evolutions.

The Eclipse Modeling Framework allows to complete the meta-models description of the technical data with consistency checking rules. Then it provides code generators that accelerate the development of dedicated bridges and data editors, especially when modifications of the meta-model have to be propagated in the code. Tools are also available to migrate the documents by analyzing the changes in their meta-model versions. This tooling allows to take into account the evolutivity challenge with minimal effort.

As a conclusion even if the first focus of SDB is the technical data that contribute to the command and control aspects, this is only a first step and SDB NEXT is developed with the objective of being extended to become the Reference Satellite Data Base for all engineering domains that participate to the virtual satellite definition.

REFERENCES

[1] ARCADIA introduction on Polarsys site https://polarsys.org/capella/arcadia.html
[2] Capella home on Polarsys site https://polarsys.org/capella/index.html