Automatic generation of a complete model driven system reference database (SRDB) application

Francesco Sgaramella[1], Emilia Barbagallo[2], Cosimo Bruno[3]

[1] European Space Agency (ESTEC)
Keplerlaan 1
2200AG Noordwijk
The Netherlands
francesco.sgaramella@esa.int

[2] Sapienza Consulting
Kapteynstraat 1
2201BB Noordwijk
The Netherlands
emilia.barbagallo@esa.int

[3] dottCB Consulting s.a.s.
Viale dei Lecci, 16
74015 Martina Franca (TA)
Italy
cosimo.bruno@dottcb-consulting.it

INTRODUCTION

Reliable data population and exchange needed during a space system lifecycle are essential to support the efficiency and effectiveness of the engineering processes, together with all other lifecycle activities and processes.

One of the main data definition and exchange processes consists in the definition and exchange of the System Reference Database (SRDB) – mainly collecting and maintaining all the knowledge related to spacecraft Monitoring and Control (M&C) such as Telemetries (TM) and Telecommands (TC) for allowing:

- On board SW development
- System AIT
- Delivery of data for in-flight operations

The System Reference Database of a given ESA project complies with the project specific conceptual data model, that depends on the nature of the data to be defined and exchanged among stakeholders.

From a data producer viewpoint, the challenge consists in the timely population of the SRDB, fully integrated in the SRDB change control and validation processes.

From a data consumer viewpoint, the challenge consists in the timely integration and extension of the SRDB, fully integrated in the SRDB change control and validation processes.

Applying the data producer/consumer viewpoint to the complete Space System Model (SSM) [3] and given the project specific nature of the SRDB data models that are needed to fulfil the project stakeholders data requirements, one of the main challenges is to verify that the SRDB data delivered by the Space Segment (Prime) to the Ground Segment (Operations) are valid and complete according to the ESA project requirements.

As part of the ESTEC/TEC-SW Database Reference Facility, the SRDB Application Generator Framework has been developed in order to *timely and cost effectively* (i.e. in line with the project schedule and constraints) produce means to assess the SRDB data quality by automatically generating the Space System Information Base (SIB) applications in compliance with the project specific data model requirements.

The SIB application is a fully model driven SRDB application able to conform to a given project specific SRDB model.

THE SYSTEM REFERENCE DATABASE

Along with the space system lifecycle, suppliers deliver "Products" to customers. A product consists of hardware component, software component, or both, together with the associated documentation containing all the knowledge required by the customer during the complete lifecycle of the product.

To facilitate the sharing and reuse of the product knowledge by the customer, the products are organised according to a formal structure that is represented by the space system model (SSM) [3].

The SSM structure is defined to capture the space system knowledge and reflects the structure of the space system itself. The SSM is hierarchically broken down into system elements (SE) mirroring the functional breakdown of the space system.

A system element is a data structure whose properties are the means to capture the space system knowledge.

System elements correspond to the elements resulting from the functional decomposition of the space system. From the highest level downwards, these are progressively: system, subsystem, set, equipment or software product, assembly, part (hardware) or module (software), as depicted in Fig. 1.

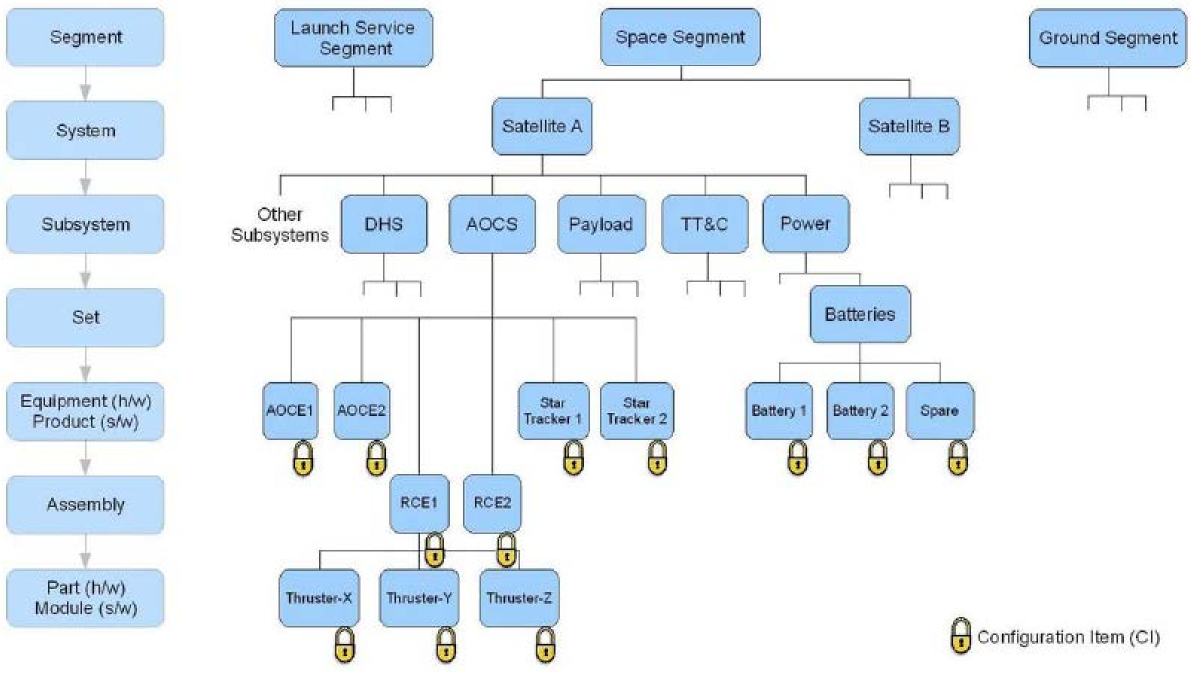

Fig. 1: decomposition of a space system into system elements

The SRDB is the formal and unique data repository where all the SSM knowledge should be centralised. When the SRDB is used to handle the SSM knowledge, all customers and suppliers that contribute to the space system development can effectively and efficiently share the space system knowledge.

Reliable data population and data exchange needed during a space system lifecycle are essential to support the efficiency and effectiveness of the engineering processes, together with all other lifecycle activities and processes.

It is fundamental that all stakeholders share the SSM knowledge at conceptual level, meaning that all stakeholders shall map their concepts to the overall SSM conceptual data model (CDM) in order to achieve the required semantic interoperability at space system level (such as effective and correct data exchange).

Once the semantic interoperability is achieved, the CDM can be used to define each stakeholder view, being sure that the exchanged knowledge is the required one and that all stakeholders needs have been fulfilled.

One example of the knowledge contained in the SRDB consists of the monitoring and control data required to operate the space system and allowing among others:

- On board SW development
- System AIT
- Delivery of data for in-flight operations

Each time a new space mission is developed, the System Reference Database shall comply with the project specific conceptual data model, that depends on the nature of knowledge to be defined and exchanged among stakeholders.

From a data producer viewpoint, the challenge consists in the timely population of the SRDB, fully integrated in the SRDB change control and validation processes. From a data consumer viewpoint, the challenge consists in the timely integration and extension of the SRDB, fully integrated in the SRDB change control and validation processes.

For the purpose to timely, efficiently and effectively support the knowledge life cycle management in the frame of a space system development and operation, an adequate SRDB tool is strongly required.

The requirements of the SRDB tool are quite challenging and include:

- The generation and instantiation of the project-specific database physical schema in compliance with the project/stakeholder-specific SRDB conceptual data model. Once the CDM of the space system has been specified, then each stakeholder, in the frame of the customer-supplier chain, shall have the means to specify and store the according knowledge.
- The import facility aimed to integrate the knowledge data coming from product suppliers in the stakeholder SRDB repository according to the agreed interface control documents (ICDs).
- The stakeholder-specific knowledge data consistency checker, to verify that the SRDB data are defined according to the project specific data integrity requirements such as data population requirements, constraints requirements, naming convention requirements.
- The export facility to export the knowledge data towards product consumers according to the agreed ICDs.
- Validation status awareness, where the validation status of concepts (data objects) is specified as:
    - a concept property reflecting the data validation information of that concept (such as validation status, validation campaign, validation procedure)
    - a dynamic property whose status changes when specific concept attributes are changed.
- Impact analysis, meaning the identification of the impacts that any data change has on the validation status of the existing concepts. This requirement is fundamental when analysis for non-regression test is required. The impact analysis engine should be configurable in order to select the concept properties that actually imply the concept to be flagged as changed. It has to be noted that concepts can be impacted not only through changed concepts but also through impacted concepts. An example of a concept impacted by a change in another one is given by a parameter where the used validity parameter has been changed. An example of a concept impacted by an impact in another one is given by a parameter where the used validity parameter has been impacted by a change in its own validity parameter (and so on in a recursive way through all possible concepts dependencies).
- The identification of the differences between concepts and datasets of two different SRDB versions (or subsets of the SRDB). Differences at concept level between two or more SRDB deliveries are fundamental for the analysis of the impact on already performed or on incoming activities (for example impacts on test procedure and flight procedures).
- The management of SRDB knowledge data through Man Machine Interface (MMI) both in debugging mode (flat table MMI) and in concept MMI mode.

THE CHALLENGE

Given the need to apply the data producer/consumer viewpoint to the complete Space System Model (SSM) and given the project specific nature of the SRDB data models that are needed to fulfil the project stakeholders data requirements, the main challenge is to timely support the space system development as far as the SSM knowledge sharing management is concerned. Examples of SSM knowledge sharing management are:

- The implementation of the complete customer/supplier knowledge data population, data exchange as well as data verification and validation management along with the full space system development.
- The verification that the SRDB data delivered by the Space Segment (Prime) to the Ground Segment (Operations) are valid and complete according to the ESA project requirements.

However, the main critical requirement of the SRDB is driven by the space system data development life cycle that usually is very challenging given the needs of data deliveries among stakeholders since the early space system development stages.

This SRDB availability requirement is made much more challenging if we consider that the foundation of the SRDB is the space system CDM, that requires a considerable effort in terms of analysis and time, therefore implying already a challenge with respect to its availability in the frame of the overall space system schedule.

The SRDB Application is required in order to *timely and cost effectively* (i.e. in line with the project schedule and constraints) provide the means to assess the SRDB data quality in the frame of the customer/supplier chain and in compliance with the project specific conceptual data model requirements.

THE SOLUTION: AUTOMATIC GENERATION OF A COMPLETE MODEL DRIVEN SRDB

It is important to understand that the SRDB requirements are based on the definition of the concepts as defined in the CDM view of a given stakeholder, that however is semantically interoperable with other stakeholders views thanks to the achieved semantic interoperability at space system level [4]. As a matter of fact, requirements such as import, export, MMI, integrity consistency checks, impact analysis for non-regression tests, difference analysis and validation awareness are clearly based on the definition of concepts, concepts roles and concept constraints that are defined in the CDM.

As part of the ESTEC/TEC-SW Database Reference Facility, the SRDB Application Generator Framework has been developed in order to *timely and cost effectively* (i.e. in line with the project schedule and constraints) produce means to assess the SRDB data quality by automatically generating the Space System Information Base (SIB) applications in compliance with the project specific conceptual data model requirements.

The SIB architecture consists of two main components as depicted in Fig. 2: the SIB application framework and the SIB MMI product editor.

The SIB application framework is in charge of handling the mapping between the SRDB CDM and the SRDB physical data model (PDM) to allow management of:

- The generation and instantiation of the project-specific database PDM, including the management of the database kernel engine to handle all involved database objects such as tables, constraints, procedures, triggers. The definition of the PDM is one of the possible PDMs derived from the CDM (actually the PDM is derived from one of the possible logical data models (LDM) derived from the CDM).
- The stakeholder import and export facility.
- The stakeholder-specific knowledge data consistency checker.
- The knowledge data validation status awareness.
- The impact analysis engine.
- The identification of the differences between the concepts of two different SRDB versions.

The SIB MMI product editor component is in charge of handling the “Create, Read, Update and Delete” (CRUD) functions against the SRDB physical data model. The SIB MMI product editor consists of two main components:

- The SRDB flat table MMI, used to access each table of the SRDB PDM, normally used for data investigation and debugging purposes.
- The SRDB concepts MMI, used to access each concept of the SRDB by using a compound view of the concepts through grouping of tables at MMI level. The SRDB concepts MMI allows having a structured access to the SRDB data, meaning closer to the user view.

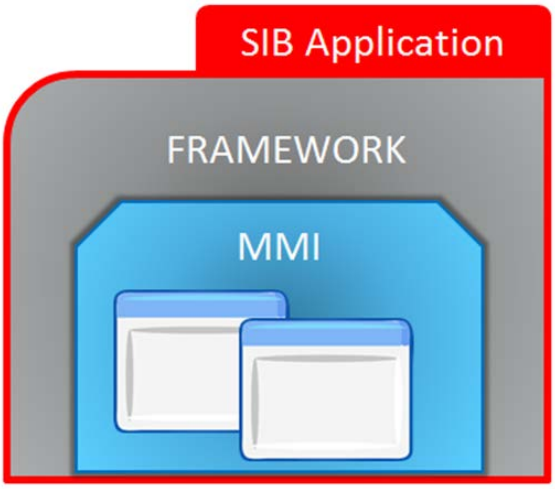

Fig. 2: SIB architecture overview

In order to generate the SIB application for a given purpose, the following models are required:

- SRDB Physical Relational Model, consisting of:
    - The SRDB PDM, used to physically instantiate the SRDB physical database schema. The current implementation of the SIB application is based on an Oracle™ physical database relational schema.
    - Validation Rules Model, consisting of a set of constraints and procedures aimed to implement the relationships among concepts and derivation rules defined in the CDM.
- MMI Model (see Fig. 3), consisting of:
    - The flat MMI view, where the MMI is one to one mapped to the PDM tables
    - The concepts MMI where the full specification related to the mapping between the concepts and the organisation is given.

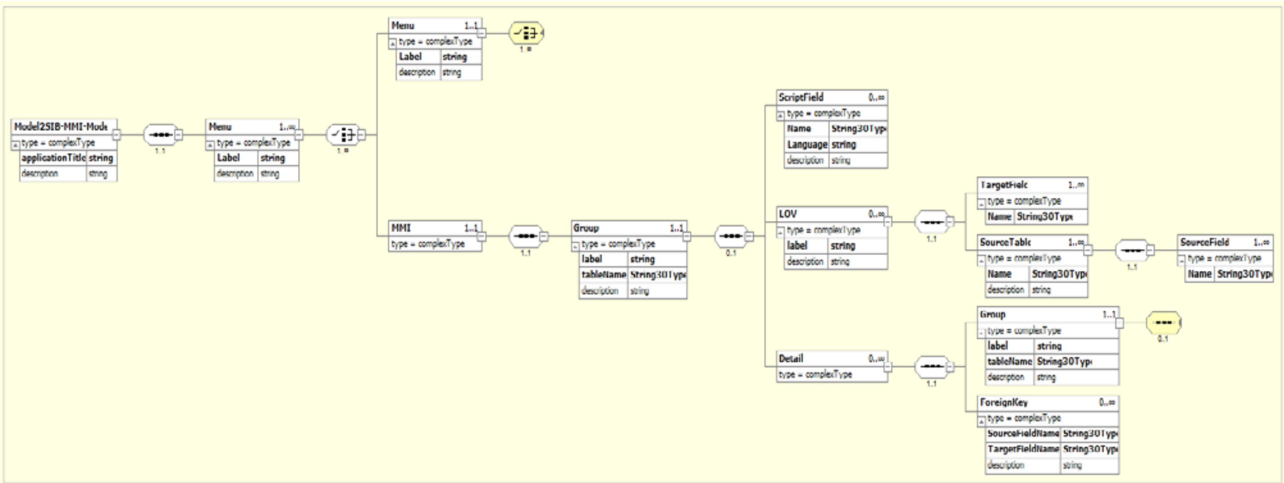

Fig. 3: MMI Model in XSD format

- Configuration Item Model (see Fig. 4), consisting of:
    - “Part of” Model, where the full specification related to the mapping between the concepts and the PDM is given in order to provide the “part of” definition of each concepts. This model addresses the complete definition on how a given concept is built on the underlying PDM in terms of entry point table, tables defining the concept and relationships among them.
    - Impact Analysis Model, where for each table in the frame of a given concept definition, all table attributes are classified in terms of impact criticality to the concept. The impact criticality is defined as the weight that each table field has in the evaluation of the changes a given concept has been

subject to. Moreover, the impact network definition is also given in terms of concepts that are referenced by a given concept to complete its definition. The impact network definition is used to evaluate the impact analysis coming from the concepts updates.

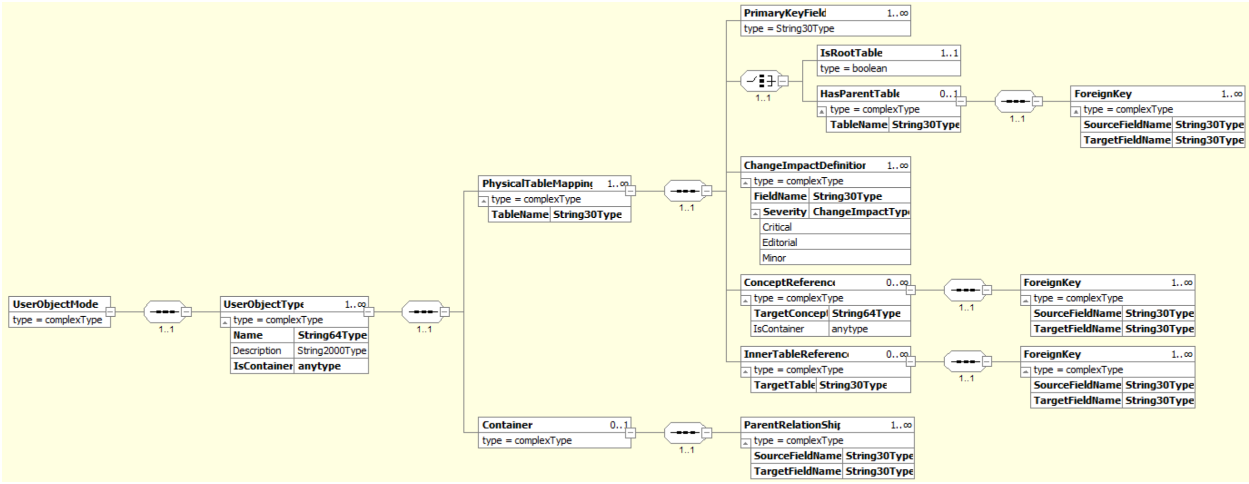

Fig. 4: Configuration Item Model in XSD format

The SIB application generation is performed by using two different model injection approaches, namely:

- Run-time model awareness, where the application software has a different behaviour at run-time depending on the input models specification. The SIB application framework component has been developed by using this approach in order to fulfil the requirements concerning the PDM management and instantiation, the validation rules engine, the different report generator and the impact analysis engine.
- Model driven code generation, meaning that the application code is first generated by using a code generator and then deployed in the application server. This approach has been chosen to provide the SIB MMI product editor component.

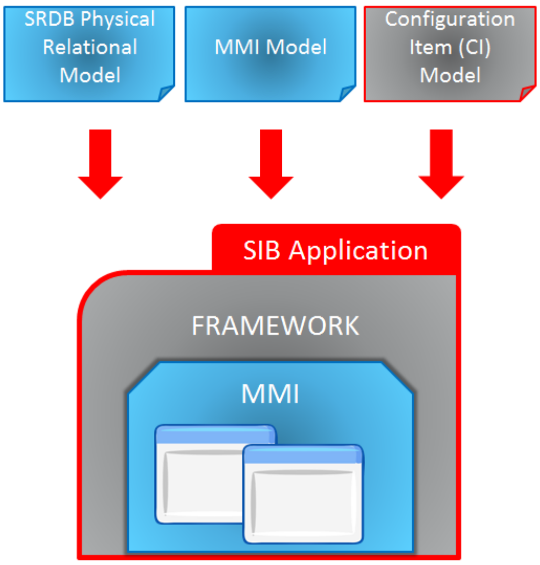

Fig. 5: SIB application generation

The SIB application is deployed on a multiple user's three-tier architecture, where the application server hosts the software and is in charge of satisfying all HTTP requests coming from any number of thin clients.

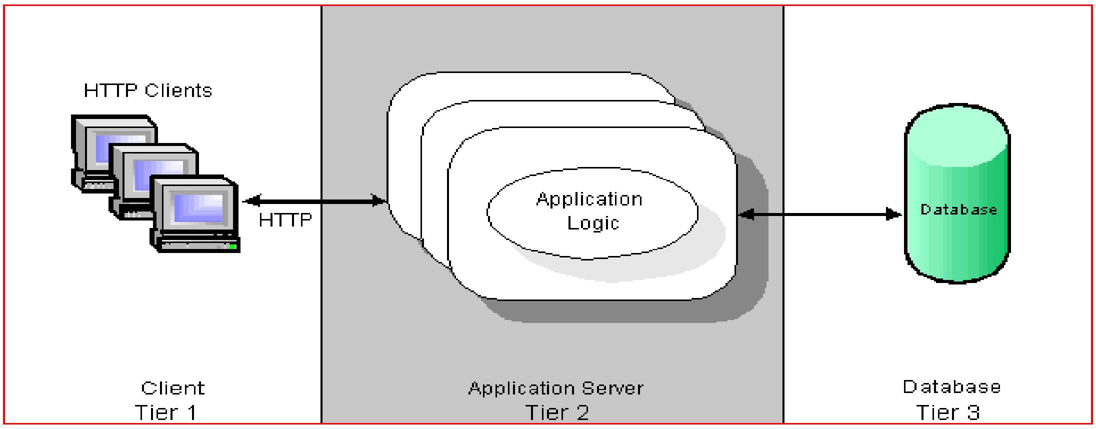

Fig. 6: SIB application three-tier architecture

The three tiers are as follows:

1. Presentation Layer (Client) - The Client of the system is responsible for outputting the MMI to the user. It simply exchanges information with the Application Server.
2. Logic Layer (Application Server) - The server is the 'intelligent' layer as it interacts with the presentation tier (Client) by being responsible for processing requests received by the client. It does the relevant computation and sends information back to the necessary clients, and manipulates data in the content layer, i.e. it updates the database.
3. Content Layer (Database) - The database is the content layer of the system as it is responsible for storing all data that needs to be saved within the SIB database repository. It saves information that it receives from the Application Server, and sends the requested information back to it.

This architectural design ensures that all clients have consistent information, as all of information is centralized through the Application Server, which sends the current system state out to the clients. In essence the clients need only the MMI, that is implemented through the use of standard web browsers. All of the processing is done by the Application Server and by the Database Server.

CONCLUSION

The SIB application is a fully-model driven SRDB application able to conform to a given project specific SRDB conceptual data model. Thanks to the automatic generation of the SIB application, the effort related to the software development is reduced virtually to zero, allowing the software analysts to concentrate their effort to fully specify the CDM and the according SRDB Physical Relational Model, MMI Model and Configuration Item Model. This approach allows the availability of the SIB application in time for the project needs since the early project phases. If we consider that actually the SRDB Physical Relational Model, MMI Model and Configuration Item Model can be automatically derived from the CDM [1], as depicted in Fig. 7, it becomes evident that the challenge to “timely and cost effectively provide the means to assess the SRDB data quality in the frame of the customer/supplier chain and in compliance with the project specific conceptual data model requirements” can be achieved.

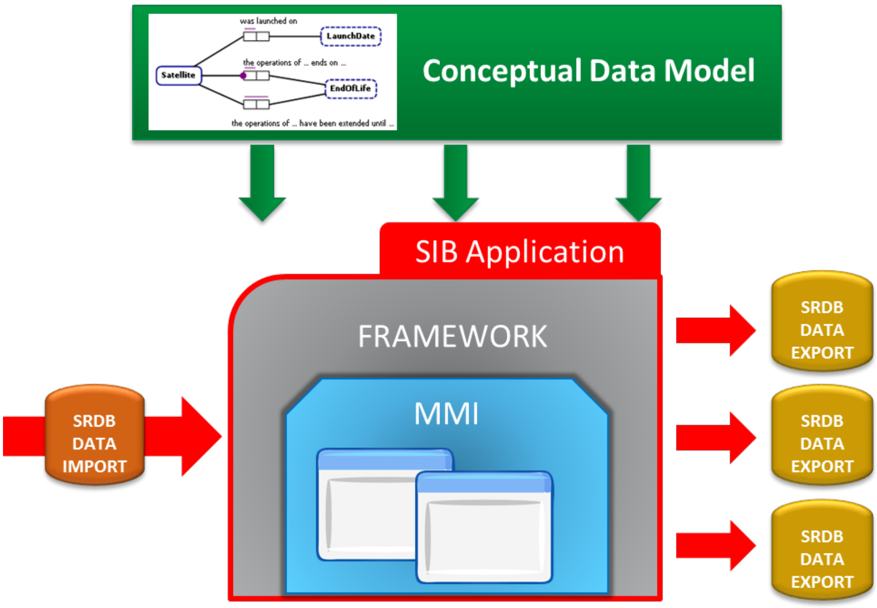

Fig. 7: 100% Model Driven SIB Application

REFERENCES

[1] FAMOUS (Fact based Modelling Unifying System) Framework

[2] ECSS-E-TM-10-23A, Space system data repository, 25 November 2011

[3] ECSS-E-ST-70-31C, Ground system and operations – Monitoring and control data definition, 31 July 2008

[4] Inge Lemmens, Francesco Sgaramella, Serge Valera, “Developing of tooling to support Fact-Oriented Modeling at ESA”, *On the Move 2009 Workshops*.

[5] Sante Candia, Francesco Sgaramella, Gennaro Mele, “PUS services software building block automatic generation for space missions”, *DAta System In Aerospace (DASIA)*, 2008