ATENA – adjusting open test exchange standard to the space domain

Workshop on Simulation and EGSE for Space Programmes (SESP)
28-30 March 2017

ESA-ESTEC, Noordwijk, The Netherlands

Łukasz Kwieciński(1), Oskar Maria Baksalary(1, 2), Jakob Livschitz(3), Joanna Modławska(1, 2), Piotr Obmiński(1), Szymon Panfil(1), Rafał Renk(1, 2), Grzegorz Wojciechowski(1)

(1)ITTI Sp. z o.o.
ul. Rubież 46, 61-612 Poznań, Poland
Email: lkwiecinski@itti.com.pl

(2)Adam Mickiewicz University, Faculty of Physics
ul. Umultowska 85, 61-614 Poznań, Poland
Email: OBaksalary@gmail.com

(3)ESA-ESTEC
Keplerlaan 1, 2201 AZ Noordwijk, The Netherlands
Email: Jakob.Livschitz@esa.int

ABSTRACT

Due to a diversity of activities carried out by ESA, as well as a multitude of entities collaborating with the Agency, there is a strong need to standardize as many issues as possible. The ATENA project – *Adjusting open Test Exchange staNdard to the spAce domain* – addresses this fact with respect to specification and execution of diagnostic test sequences, and proposes a joint and uniform approach to this area of the Agency activities. The cornerstone of the ATENA system is the OTX – Open Test sequence eXchange standard (ISO 13209), which was originally designed for the needs of an automotive domain, but later proved to be profitably applicable also in other branches of industry. The overall goal of ATENA is to adjust OTX to the requirements and peculiarities of the space domain, so that the resulting space-tailored, OTX-based procedure format will allow for Space System Model (defined within ECSS-E-ST-70-31C). The worked out solution is to satisfy also the requirements specified by the ECSS-E-ST-70-32C standard. Within the project software solutions necessary to demonstrate usability of the outcomes of ATENA are to be developed as well.

INTRODUCTION

It is very important for the entities involved in specification and execution of diagnostic test sequences, particularly those consisting of a few branches (possibly scattered over different locations) and/or having an extensive collaboration with other companies or institutions, to base their work on a proper common approach. Without a well-designed and uniform standard, the specification and execution of the test sequences will likely require higher time expenditure than necessary and lead to a superfluous increase of costs. Moreover, one can expect that a lack of the standardised attitude in preparation of test sequences may create overlapping and inconsistences of work between different entities involved in the process of developing new equipment. In addition, without one joint trustworthy standard an exchange of the sequences between collaborating groups, branches, or entities may be troublesome (e.g. due to different software versions) and may lead to a superfluous increase of error susceptibility.

The importance of implementation of a joint and uniform approach to the specification and execution of diagnostic test sequences was recognized in the automotive domain, where a dedicated standard, namely OTX – Open Test sequence eXchange (ISO 13209), was developed and commonly introduced. Although, designed to satisfy the needs of a single area, the standard proved to be open-ended providing possibilities to be exploited also in other domains. From the perspective of its potential reusability, a crucial feature of OTX is that its high flexibility is accompanied by its relatively low complexity.

The ATENA project – *Adjusting open Test Exchange staNdard to the spAce domain* – was launched to tailor OTX standard to the needs of the space domain. The overall goal of ATENA is to provide ESA with an OTX solution capable of building and executing test sequences of space systems. Since ATENA builds upon a standard which has already been extensively exploited in the automotive domain (but also in other industry areas) where it has proved to be extremely useful and reliable, it is anticipated that the outcomes of the project will meet all requirements of the space domain, providing full compatibility with space standards applied for the management, engineering and product assurance in space projects and applications, e.g. [1]. The results of ATENA can be looked at as an alternative solution to currently used standards, like [2].

OPEN TEST SEQUENCE EXCHANGE STANDARD (OTX)

OTX is an international standard for a description of diagnostic test sequences. The standard determines an open and standardized, tester-independent, XML-based data exchange format (along with a sequence language) for formal description and documentation of executable sequences. The OTX documentation consists of three parts specifying the main purpose of the standard with corresponding data model, requirements and mechanisms.

The first part [3] '*General information and use cases*' (called OTX Overview) describes the purpose of the standard on a high level, together with a list of use cases which the standard aims to fulfill. General considerations regarding integration of OTX with existing standards (ODX – Open diagnostic data exchange, MVCI – Modular vehicle communication interface, UDS – Unified diagnostic services, etc.), achieving long-term availability of test sequences, and other related issues are presented there. The use cases included in this part define multiple aspects, i.e. documentation and specification, exchange and reusability, extensibility, localization and runtime execution.

The second part [4] '*Core data model specification and requirements*' (called OTX Core) describes the main requirements and principles of the core OTX data model as well as validation and extension approach. OTX Core data model consists of the OTX features and mechanisms that are necessary to build basic test sequences. OTX Core data model may be extended following the extension mechanism based on extension interface. This aspect is crucial for adjusting OTX to the space domain purposes.

The third part [5] '*Standard extensions and requirements*' (called OTX Extensions) delivers a set of additional features, defined in compliance with the extension mechanisms specified in OTX Core. OTX Extensions consist of OTX features, which provide additional functionalities to the diagnostic system. The most commonly used OTX Extensions are automotive industry-specific, like DiagCom, DiagDataBrowsing, but there are also more general extensions provided, for instance DateTime, HMI, Math, Measure. Selected OTX Extensions might be useful in a space system use cases as well.

OTX data model

The OTX Core data model is a complex structure, which includes a great number of various features. The standard defines each data model aspect in a detailed way including syntax, semantics, possible exceptions, validation rules, examples, figures, etc. The data model definition is provided within UML (Unified Modelling Language) and XSD (XML Schema Definition) that ensures consistency of description. Fig. 1 presents the high level design of OTX Core data model.

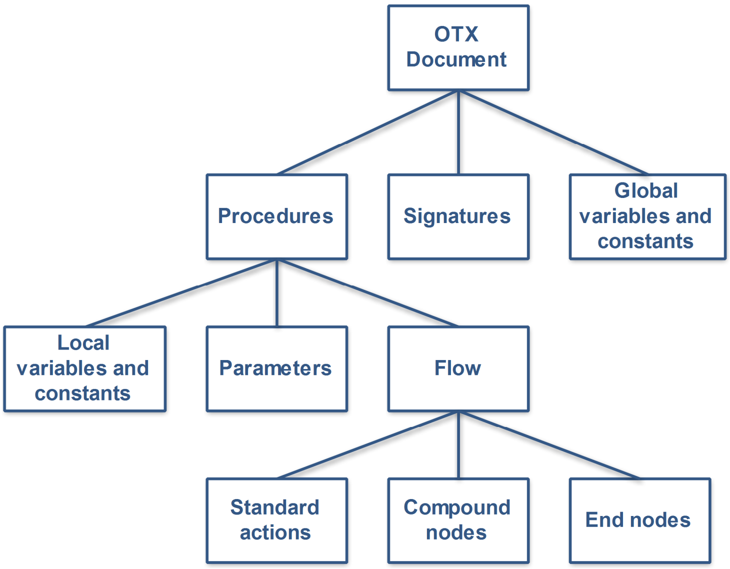

Fig. 1 Overall view of OTX Core data model

OTX Core data model is represented in a hierarchical way. The main part of every OTX document is a root element including basic information, like name, version, meta-data, imports, etc. as well as global scope information, e.g. declarations of variables or constants. Within an OTX document there may be declared procedures (representing a part of a test sequence) or signatures (equivalent to programming interface), which are an essential part of OTX.

Procedure, being a NamedAndSpecified type within OTX data model, is represented by basic attributes and three elements – specification, metaData and realisation. The specification includes a textual description of a procedure, its purpose and further details. metaData is a container of various data elements, each being a key-value pair, which can be used to more extensively describe a NamedAndSpecified element. Those data elements can be simple strings, embedded XML documents, href links to pictures or videos, html, CDATA, etc. Specification and metaData elements do not alter the processing logic of a node they describe. The realisation of a procedure defines the parameters (in-, out- or inout-) and locally used constants/variables of a specific OTX data type. Besides, the realisation defines a logical flow of executable nodes.

The flow consists of nodes which are interpreted in a sequential way by an OTX runtime, developed accordingly to the OTX standard. The following types of nodes are distinguished: standard actions, compound nodes (e.g. Group, Loop, Branch, Parallel) and end nodes (e.g. Return, Continue, Break, Throw).

Advantages of using OTX

The OTX standard enjoys several features which, even though developed to meet requirements of the automotive industry, seem to be well matching to the needs of the space domain. Thus, the standard will likely prove to be an appreciated competitor to the corresponding solutions used within ESA at present.

Availability of human- and machine-readable levels within OTX test sequence

An essential advantage of OTX is the fact that the specification, metaData and implementation of test sequences are combined in a single OTX document. The specification and metaData can provide a comprehensive description of a test sequence as well as of each element declared within it. It allows people without technical knowledge to understand a purpose of any test sequence (human-readable), but also assists building high level test sequences and adding implementation data to them. The implementation of any single step of a test sequence describes technically how it should be performed (machine-readable). There are three stages of the test sequence development. The first one is a specification stage describing at linguistic level the overall sequence logic. Then, an implementation stage applies to the situation when isolate steps of a test sequence are executable. Finally, a realisation stage corresponds to a fully executable test sequence, i.e. where there are not any specification-only parts in the sequence.

Referring to the defined OTX development stages, a typical test sequence development process can be described by the following steps. First, during the specification stage, future test sequences are created in a preliminary form even though not all sequence details are known. The sequences are specified as free text, stored in a target XML format, and can be exchanged. Then, the free-text sequences are filled with instructions which can be processed by an OTX runtime. Actually, the script can be executed already in this partially implemented stage. Once this stage is completed, all test steps defined in the specification stage are extended with executable instructions. The completed script is afterwards implemented and can be executed in the realization stage.

To conclude, such a scheme is characterized by high efficiency of the creation process (in terms of time and costs) and reduction of possibility of errors occurrence, since capability to store both human-readable and machine-readable formats within OTX files allows maintaining both concept and implementation logic synchronously minimizing the threat of inconsistencies.

Extensibility of OTX data model

An extensibility of the OTX data model enables building rich libraries of OTX Extensions easy-adaptable to the needs of the domain of interest. Extending the OTX data model is possible due to a clear extension mechanism which simplifies adding new data types, terms, variables and actions/signature realisations. This way, any missing OTX features that might be needed in a new area can be developed. As long as the OTX principles and requirements are followed, it is not a problem to assure consistency with the OTX Core as well as with other OTX extensions developed in parallel for other domains.

Reusability of OTX test sequences

OTX test sequences are usually dedicated to particular systems, but a part of them might be defined at universal level to increase their reusability within other testing phases. One of the mechanisms that makes it possible is a signature. The signature is a similar feature to an interface used in object programming languages (e.g. Java). It provides a possibility to define general input and output parameters of a procedure without defining the implementation body (i.e. without concrete procedure realisation part). Then it is possible to provide more than one procedure able to realise a given signature within the whole test sequence. Selection of a valid procedure for the signature implementation purpose is made by other validity mechanisms. This way, test sequences as internally linked sets of procedures and signatures, may be partially used as an input to other test sequences performed on a different System Under Test (SUT).

Error detection while creating OTX test sequences

OTX standard includes a strictly defined set of stand-alone OTX checker applications which ensure OTX document (XML file) correctness at two levels. The first level concerns OTX document correctness with respect to constraints imposed by the OTX Schema and the second one includes all cases that cannot be verified at the XSD level. The second level validation concerns semantic issues (e.g. reference validity, proper calls defined within procedures, data interpretation, etc.). For a selected OTX feature the set of dedicated checker rules defined by severity level and OTX runtime behavioural commands encompasses every possible semantic error. These features of OTX ensure error-safe environment in terms of building test procedures.

Focusing on know-how, rather than on programming expertise

Since OTX is a XML based standard devoted to test procedures, any specific programming expertise is no longer an indicator of successful test preparation and common understanding of test sequence flow by various contributors. The standard is relatively simple to understand and implement. A test designer is expected to have know-how of the domain of interest when using an OTX tool (e.g. dedicated editor ensuring the syntax and semantics correctness). Since every human-mistake can be automatically detected within preparation and execution stages of the testing process, the test sequences creation process supported by OTX is less error-prone than through preparing scripts of the procedural test sequences.

Ensuring manufacturer independence in terms of software tools design

There are multiple competitive tools on the market utilizing OTX standard for car diagnostics. Since OTX is an ISO standard, the only differences between fully OTX compliant tools are on the side of performance and interface, which makes them interchangeable and helps to avoid becoming vendor-locked. The same is true also for software tools expandability, e.g. providing a possibility to use new OTX extensions through already built OTX test sequences.

Long-term availability of OTX test sequences

There is no expiration date on the OTX test sequences and therefore they can be reused and shared at any time, that being attributed an ISO specification guarantees long term validity of the standard

SCOPE OF SPACE DOMAIN STANDARDIZATION

Space domain standardization is ensured by a great number of standards corresponding to various areas, phases and systems, provided mainly by ECSS (The European Cooperation for Space Standardization). The scope of the standardization essential from the point of view of adjusting OTX to the space domain covers space engineering branch within ground systems and operations discipline (E-70).

The main model that needs a brief introduction, since it plays a crucial role in ATENA, is Space System Model (SSM) specified within [1]. SSM reflects a functional decomposition of the space system and is represented as a tree structure describing a space system with its system elements (SE) that can be expanded by activities, events, reporting data or other sub-systems (i.e. another related system elements). The possible objects defined within SSM are:

- System Elements (SE) that are results of a functional decomposition of a space system defined in ECSS-E-00 and ECSS-E-70;
- Activities (A) that provide monitoring and control functions. They are associated with SE referring to procedures, telecommands and any function provided by the EMCS (Electrical Monitoring and Control System) implemented within the EGSE (Electrical Ground Support Equipment) or any other mission control system;
- Reporting Data (RD) associated with SE that comprises parameters;
- Events (E) associated with SE and RD representing occurrences of a set of conditions that can arise.

SSM in the respect to adjusting OTX to the space domain can play a role of an interface between a SUT and a dedicated OTX test sequence.

ADJUSTING OTX TO SPACE DOMAIN

In order to enable building, exchanging, running and maintaining SSM based test sequences, a development of space-oriented OTX runtime environment is necessary. A significant part of this task is devoted to design and implementation of the space domain extensions. An approach to design work follows the extension mechanism defined in OTX Core and already used to deliver existing OTX Extensions of the standard in the automotive industry. Such an approach allows to achieve a common language for exchanging ESA diagnostic test sequences.

The ATENA is aimed to perform an integration between ECSS-E-70 and OTX in the way presented on Fig. 2.

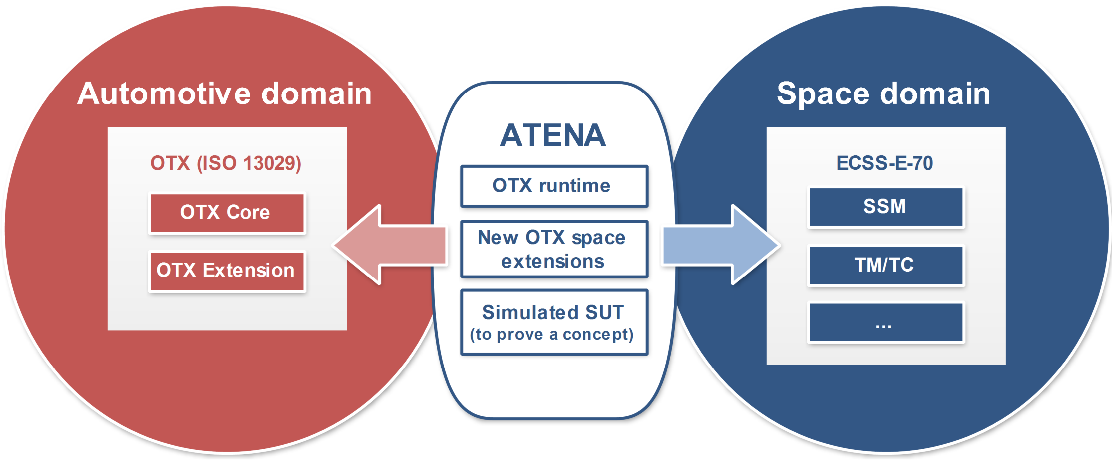

Fig. 2 Concept of adjusting OTX to the space domain

OTX runtime and library of relevant and new space extensions (as the main part of the ATENA system) constitute an integration point for both OTX and SSM. A simulated SUT plays a role of a real system (e.g. spacecraft, ground station) communicated via dedicated channel equivalent to TM/TC interfaces to give a proof of proposed concept.

Integration points between OTX and SSM

A model consisting of two SSM levels is the base while elaborating the SSM-oriented extensions to OTX. These levels are SSM Declaration (general specification of the SSM objects, e.g. activities names and identifiers) and SSM Implementation (a set of values and technical description of the interfaces/communication channels, e.g. specific XTCE – XML Telemetric & Command Exchange – data).

For the ATENA project purposes SSM is represented as a XML file. As depicted on Fig. 3, for every ATENA space system two SSM representations are provided, namely SSM Declaration and SSM Implementation, along with their respective XSD files.

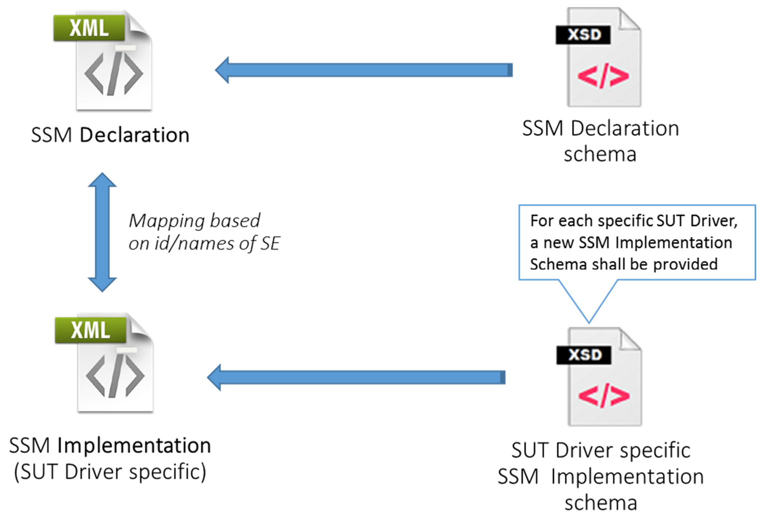

Fig. 3 SSM files describing a space system

The space-dedicated OTX solution is to be adjusted to the declarative part of the SSM model and is not supposed to be connected with real TM/TC interfaces directly. Each space system to be verified and validated with OTX test sequences is to be accessible through a dedicated driver responsible for translating SSM Declaration to SSM Implementation corresponding to TM/TC understandable by SCOE (Special Check-Out Equipment). An example of such a translation at SSM level performed by REST (REpresentational State Transfer) based driver is presented on Fig. 4.

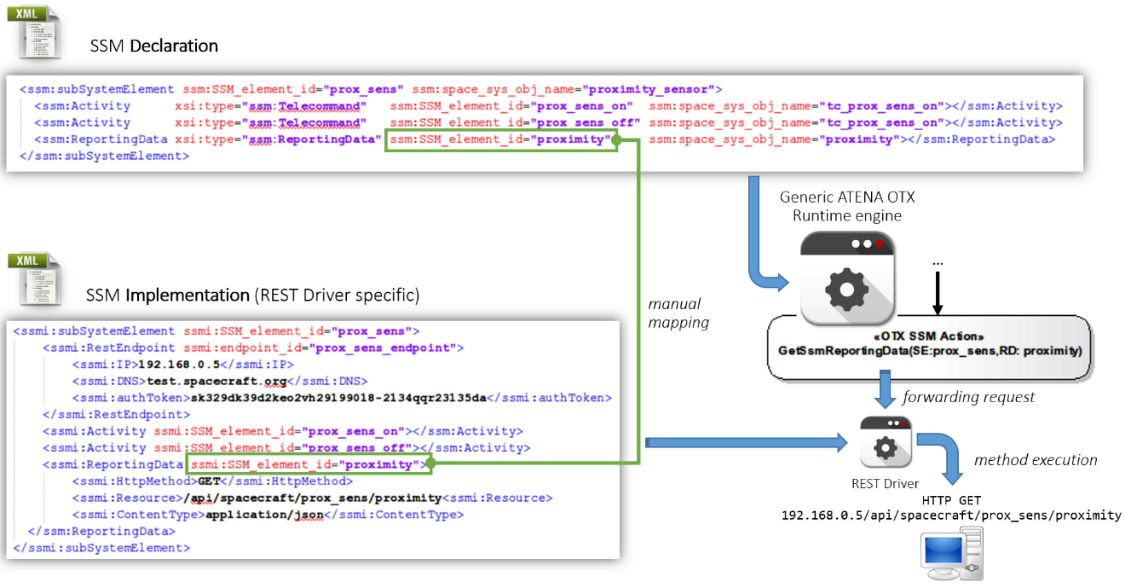

Fig. 4 SSM/OTX data flow example

In this use case a specific action execution request based on SSM Declaration is forwarded from the OTX runtime engine to the dedicated RESTful driver. Thanks to mapping provided between SSM Declaration and SSM Implementation, the driver can identify which REST method needs to be invoked in order to execute the proper TC.

Products of the ATENA project

The final ATENA system is understood as an operational system independent toolkit, including three main components, i.e. OTX Editor for creating OTX procedures, OTX Engine for running OTX procedures and SUT Driver for assuring communication with SUT. Each of these components takes part in the process of building and executing OTX test sequences. Dependencies and data flow between the components are presented on Fig. 5.

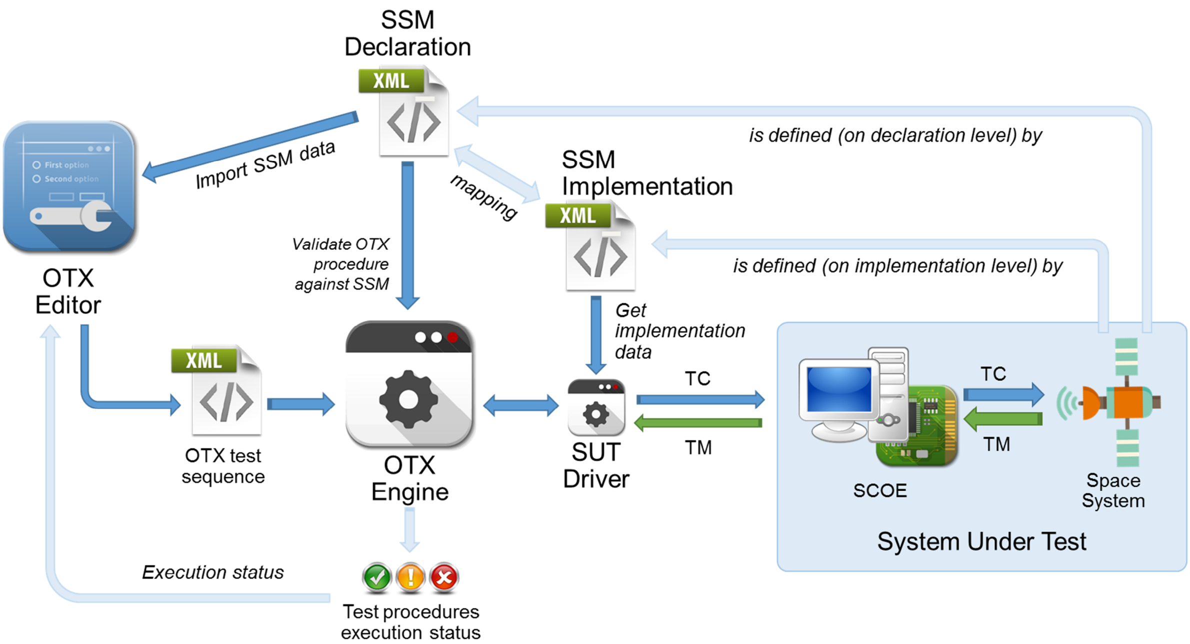

Fig. 5 Products of the ATENA system

The presented operation flow starts with the OTX Editor, which allows a user to build OTX test sequence (in form of OTX documents list) compliant with the SSM Declaration file. When OTX test sequence is ready, its execution can be initiated by the OTX Engine. At first, the OTX Engine validates OTX test sequence's content with SSM Declaration to avoid data incorrectness and then starts the execution. Each time a procedure step referring to SSM Declaration is interpreted by the OTX Engine, the proper activity call is send to the SUT Driver. In turn, the SUT Driver identifies the received activity call and maps it to SSM Implementation to call SUT by using dedicated technology and (defined in SSM Implementation).

The flow goes analogically the other way around. At first, SUT Driver gets TM from SUT using a dedicated technology. Then, the SUT Driver maps received message to 'technology-free' context (based on SSM Declaration) and sends the proper results to OTX Engine. The OTX Engine interprets received message and communicates the updates execution status to a user (e.g. via the OTX Editor dedicated console).

CONCLUSIONS

It is believed that the results of the ATENA project will make a valuable contribution to the space domain. OTX standard extended by space-dedicated extensions is a promising solution to improve the workflow concerned with the test sequences preparation, performance and maintenance. In this respect, a large number of added values are foreseen to be brought to ESA routines. OTX approach brings transparency to what is required from the vendors in terms of testing procedures within the space domain, which shall ease specification of system requirements. It shall also shorten the process of preparation of the test sequences and smoothen their further expanding and exchanging within Agency and among cooperating entities. Moreover, reusability of the existing OTX test sequences shall impact the effectiveness of future testing tasks. Finally, the possibility of working simultaneously on a single OTX test sequence file (or a set of files) at two levels – human-readable and machine-readable – shall enhance the chances of achieving a common understanding between all engaged stakeholders.

ATENA is to assure a coherent integration between OTX space-oriented extensions (at both design and implementation level) with SSM (representing the model of SUT). Supplementing the integration with a high quality OTX compliant runtime and an editor prototype (both to be delivered as the project outcomes), shall lead to the overall system which will likely become a starting point for a significant upgrades in the several important areas of the space domain.

REFERENCES

1. ECSS-E-ST-70-31C, "Ground systems and operations – Monitoring and control data definition", 31 July 2008.
2. ECSS-E-ST-70-32C, "Test and operations procedure language", 31 July 2008.
3. ISO 13209-1, "Road vehicles – Open Test sequence eXchange format (OTX) – Part 1: General information and use cases", 2011-11-01.
4. ISO 13209-2, "Road vehicles – Open Test sequence eXchange format (OTX) – Part 2: Core data model specification and requirements", 2012-08-15.
5. ISO 13209-3, "Road vehicles – Open Test sequence eXchange format (OTX) – Part 3: Standard extensions and requirements", 2012-08-15.