Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

JOINT MILITARY SPACE OPERATIONS SIMULATION AS A SERVICE

Erdal Cayirci

Department of Computer Science
University of Stavanger
P.O. Box 8600 Forus
Stavanger, 4036, NORWAY

Hakan Karapinar

Training & Simulation Systems Department
HAVELSAN
2120 Cad No:39 Çankaya
Ankara, 06510, TURKEY

Lutfu Ozcakir

Training & Simulation Systems Department
HAVELSAN
2120 Cad No:39 Çankaya
Ankara, 06510, TURKEY

ABSTRACT

Our Training and Experimentation Cloud Architecture, namely the hTEC, is applied to joint military space operations simulation. The hTEC follows the recommendations on the modelling and simulation as a service (MSaaS) by NATO Science and Technology Organization. The space mission areas and their characteristics are investigated and the requirements are analyzed. The hTEC services are designed such that these requirements are fulfilled. Each service addresses the minimum set of functions that may be needed by a military space operations mission area, and can be run independently, federated as a composed service or linked into a software application. The designed joint military space operations simulation architecture is implemented in a testbed called the extended BSigma.

1 INTRODUCTION

NATO Modelling and Simulation Group (NMSG) anticipates that future military capabilities, including training, mission planning and decision making will be provided through increased use of modelling and simulation (M&S). However, there are currently two main barriers, namely the perceived cost and time taken to compose and develop simulation systems. Furthermore, limited credibility resulting from unknown validity and ad-hoc processes continue to be a serious problem (NATO MSG-136 2017).

M&S products are highly valuable to NATO and military organizations, and it is essential that M&S products, data and processes are conveniently accessible to a large number of users as often as possible. Therefore a new "M&S ecosystem" is required where M&S products can be more readily identified and accessed by a large number of users to meet their specific requirements (NATO MSG-136 2017). For these reasons, Modelling and Simulation as a Service (MSaaS) (Cayirci 2013; Siegfried, Berg, Cramp and Huiskamp 2014) approach has being adapted for the future M&S architecture that supports the military users in NATO and many Nations. The M&S Research Task Group (RTG) MSG-136 ("Modelling and Simulation (M&S) as a Service (MSaaS) – Rapid deployment of interoperable and credible simulation environments") has completed a three year collaborative research, developed a prototype for demonstration and prepared an operational concept document (NATO MSG-136 2017).

Parallel to the research by MSG-136, we develop our own training and experimentation cloud (hTEC), (Cayirci, Karapinar and Ozcakir 2016) where all the principles and recommendations by MSG-136 are followed. The hTEC Architecture has a highly scalable, layered, distributed and service oriented design that supports interoperability, service discovery and composability. It provides standardized services and simple interfaces to access them, and supports the adoption of the previously developed services for the M&S as a service (MSaaS) ecosystem. We currently implement hTEC as an international testbed called BSigma, where we develop our joint military space operations related simulation services.

Space capabilities are significant force multipliers when integrated into joint operations. Such integration requires a common and clear understanding by Joint Force Commanders (JFCs) and space operators on how space capabilities contribute to joint operations and how they should be integrated into joint operations. In NATO and especially coalition operations, this is challenging due to the fact that space capabilities are allocated by a limited number of nations. Hence, joint military space operations simulation (JMSOS) is a critical requirement especially for collective training and exercises in the international environments, such as NATO. Nevertheless, space operations simulation services are very limited and do not suffice fulfilling the requirements in international computer assisted exercises (CAX). Therefore, we selected services for JMSOS as the first to implement on BSigma.

The main contribution of this paper is to show a way to decompose JMSOS into hTEC services following the recommendations by MSG-136 (NATO MSG-136 2017). To the best of our knowledge, ours is the first attempt to achieve this. JMSOS services can be composed into a joint theater level or global simulation by using hTEC interfaces as explained in this paper or can join to a federation as an HLA federate (HLA 1516-2010). The preliminary results from BSigma proves the concept. We plan to join the follow on MSG-136 RTG with our JMSOS services to improve and demonstrate its effectiveness.

In Section 2, we introduce the hTEC architecture. The requirements for JMSOS are elaborated on in Section 3. The main contribution of this paper, which is the hTEC service layout for JMSOS services, is in Section 4. The BSigma testbed for JMSOS is explained also in the same section. The paper is concluded in Section 5, where our future plans for the hTEC JMSOS services are briefly mentioned.

2 THE HTEC ARCHITECTURE

In Figure 1, the hTEC layers and their mapping to cloud service models including MSaaS (Cayirci 2013: Zehe et al. 2015; Taylor et al. 2015) is illustrated. The bottom layer in hTEC is a platform as a service layer (PaaS). All the details related to the infrastructure and platforms are autonomously taken care by the PaaS layer. The service layer runs on top of the PaaS layer. The models in this layer manage and process the data related to the synthetic environment by using the services from the PaaS layer. The service layer provides models as services (MaaS) (Cayirci 2013), including the database management functions. The users can manipulate the synthetic environments by using the services provided by the service layer. Please note that the security service is a sublayer within the service layer.

The service composition layer can compose a service mashup from the models provided by the service layer. It can be mapped to modelling as a service in cloud service models with a difference. Modelling as a service can be used to create new atomic or composed models (Cayirci 2013). In hTEC, the service composition layer (Cayirci 2013b) is not used for creating new atomic models but models composed of the services provided by the service layer. Please note that a service can be composed of a single service from service layer. When service composition is complete, a composed model, or in other words a simulation application (i.e., software) is compiled. Therefore, the layers below the red line in Figure 1 are before the compilation of a simulation application, and the layers above the red line provide the run time services.

The session layer in hTEC runs the models composed by the service composition. Therefore, it is equivalent to the simulation as a service model (Cayirci 2013). It enables users to run multiple instances of the composed services and federating them by using various interoperability technologies such as high level architecture (HLA) (IEEE 1516-2010). Each instance runs with its own image of the synthetic environment, therefore the master copy of the synthetic environment is preserved for the usage of the others as long as needed. The instance management service also provides the users with the capability to run each of these instances as different types of simulations such as time stepped, continuous, static or dynamic.

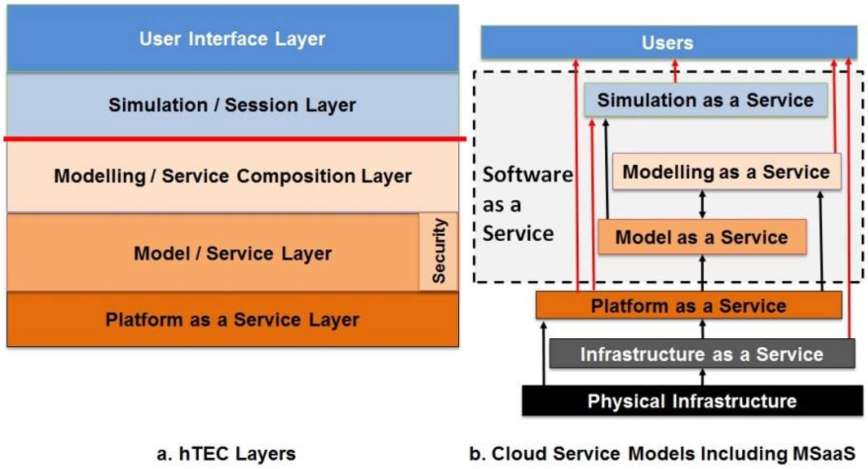

Figure 1: hTEC and MSaaS.

During service composition, the parts of the services that need to be run in the front end due to stringent end to end delay constraints are determined. The part of a MaaS with stringent delay constraints is called as the cerebellum function of the service (Cayirci, Karapinar and Ozcakir 2015). Cerebellum functions are migrated to the machines close enough to the front end (i.e., the machines that satisfy the delay constraints) by the session layer.

In Figure 2, the examples for the services in each hTEC layer are illustrated. hTEC is designed as a distributed architecture. Therefore, there may be thousands of services available around the world when it is implemented as a public cloud. The hTEC architecture can also be used in a private cloud model where hundreds of services are available. Hence, service discovery and service composition is the first challenge.

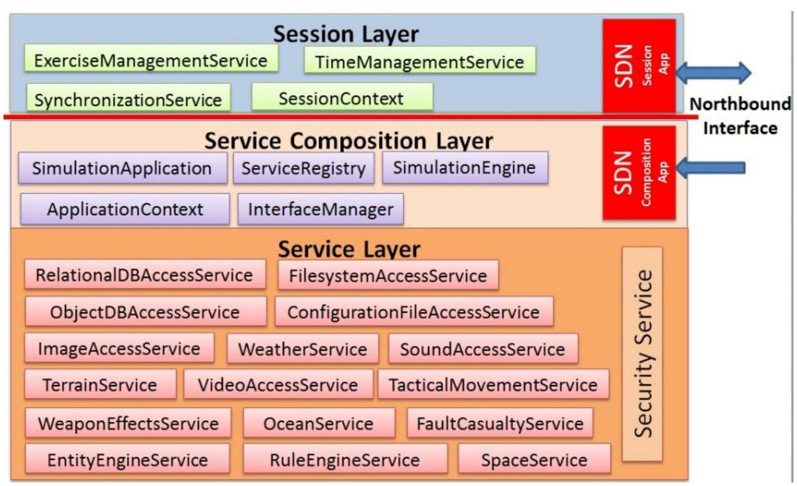

Figure 2: Examples for the services in hTEC.

Service composition is a hard but solvable problem (Cayirci 2013b). In hTEC, the service composition does not have a time constraint because the services are wired into a single application during compilation, and then run in a machine in the cloud that satisfies the quality of service (QoS) requirements. As long as a standard approach is followed to define the services and to interface with them, service discovery is also a trivial task. There are already many standardized and scalable directory (X.500 2016) and service discovery (Helal 2002) mechanisms that can be used for this purpose.

The interfaces of the hTEC services are shown in Table 1, which has two parts: the meta data and the interface for the service. The meta data are the detailed and machine readable description of the service. It includes the information, such as, the service type, the fidelity, the resolution and the service model. The notation and the values for this information have to be standardized for interoperability. For BSigma purposes, we use a proprietary standard which is flexible. Please note that the first two fields in our structure are about the standard followed by the interface and its version. Therefore, hTEC allows multiple standards. However, for composing services that follow different standards, there will be a need for standard conversion before service composition.

Table 1: The Meta Data and Interface for a Service.

Type	Name	Remarks
Meta Data about the Service	The standard	The standard followed for the description of the service
	The version of the standard	The version of the standard followed
	The name of the service	The name of the service
	The service type	The type of the service (from the list in the standard)
	The resolution	The level of resolution (from the list in the standard)
	QoS Parameters	Values for the QoS parameters (from the list in the standard)
	The fidelity	The level of fidelity (from the list in the standard)
	The service description	The details and important remarks about the service
	The version of the service	The version of this particular service
	The date	The date that this version is released
	The developer	The details of the developer
	The service model	Modelling as a service, model as a service, payment model, etc.
	The URL	The link for the service
	The cerebellum function	Null if none, the offset if the service has a cerebellum function
	The delay constraint	The distribution and statistics for the delay constraint
Interface	Return Type	Type and range of the return value by the service
	Input Parameters	The input parameter list including the type and range
	Output Parameters	The output parameter list including the type and range

The interface has three kinds of parameters similar to the structure of the subroutine calls in many programming languages: Please note that the name of the service is already among the meta data. That name is used for calling the service from inside the composed service. Apart from the name, the other fields in the interface are the return value and range, input parameter list including their types and ranges, and finally the output parameter list including their types and ranges.

Another challenge for hTEC is due to the propagation delay between the back end (i.e., the data center where the composed service runs) and front end (i.e., the machine used for interacting with the system). This becomes critical, especially when interactive audio-visual systems are used. The cerebellum function is for solving this issue. The Cerebellum function includes the part of an MSaaS which is time sensitive in responding the user commands (i.e., inputs). Please note that the delay in responding to user commands by the simulation has to be the same as the delay in response to user commands by the real system. For example, if the delay in the real system d_r is between 90 and 100 msec, the delay in the virtual system needs to be within the same 90-100 msec window. As visualized in Figure 3, our scheme is based on the idea that the maximum delay between the user interface and cerebellum function d_{max} must be shorter than the lower bound of the real life system delay $r_{\alpha min}$ according to a given confidence level α. Hence, the delay can be managed such that negative training is avoided and immersion is maintained. The maximum delay d_{max} includes not only the propagation delay p_{max} introduced by the physical distance between two ends of a communications link but also computational delays c_{max} due to processes, such as encryption, decryption, routing, service federating, etc. We treat $d_{\alpha max}$ as a random variable, and make our computations based on the upper bound according to the given confidence level α.

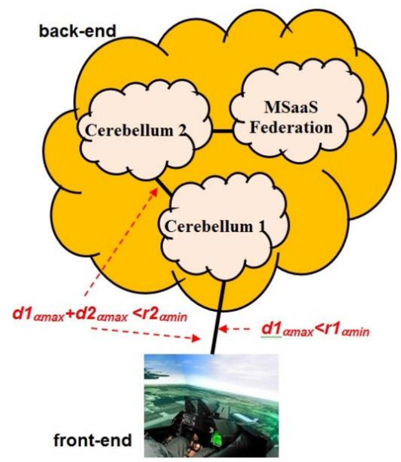

Figure 3: hTEC Cerebellum Function (d_1 is the simulation delay between the user and Cerebellum Function 1 and r_1 is the associated real system delay. d_2 is the simulation delay between the user and Cerebellum Function 2 and r_2 is the associated real system delay).

When the services are designed, the designer should design the time sensitive part of the service as decomposable (i.e., can be separated from the rest of the service). Hence, the entire service and data does not need to be migrated closer to the front end but only the time sensitive part of the service. For example, the part of an interactive visualization service (IVS) that fetches the terrain data and weather conditions and creates three dimensional virtual environments can be designed separately from the part that makes the projections based on the user commands. The later part, which is time sensitive, becomes the cerebellum function for IVS. Please note again that this is only a simplified example to clarify the meaning of the cerebellum function.

In some cases, not only the cerebellum function of a service, but all of the service must be treated as a cerebellum function depending on the configuration of a composed service. If an input of Service s_a uses another Service s_b, which has a part that needs to be treated within the cerebellum function, s_a as a complete service has to be within the cerebellum function. Moreover, a cerebellum function may also have a nested structure, which means that the inputs of a cerebellum function may be coming from another cerebellum function. Therefore, the location of a cerebellum function is selected such that the conditions in Equations 1 and 2 are met.

$$dn_{\alpha \max} = \sum_{k=1}^{n} u(pk_{\alpha \max}) + u(ck_{\alpha \max}). \quad (1)$$

$$dn_{\alpha max} < u(rn_{\alpha max}). \quad (2)$$

where $n-1$ is the number of cerebellum functions preceding the cerebellum function n in the nested structure. Please see (Cayirci, Karapinar, Ozcakir 2015) for the detailed description of the cerebellum function.

The cerebellum function can also provide better security for military MSaaS. Although the environmental data and specifications of military equipment, such as maximum speed and altitude that a military aircraft can reach are unclassified, the turn rates and similar data about the aircraft may be classified. Since the effects like turn rates are time sensitive and therefore will be typically treated by a cerebellum function in IVS, the cerebellum function approach may become useful also for dealing with the security related challenges of MSaaS because it stays in the front end.

In hTEC, two services (i.e., one in the service composition and one in the application layer) are introduced for software defined networking (SDN) (Hu, Hao and Bao 2014), namely the SDN composition and SDN session services. Both of these services are applications to provide northbound interfaces for the control layer in SDN as shown in Figure 4. Please note that the interfaces between the control layer and applications are called as the northbound interfaces in software defined networking (SDN), and SDN Composition and Session Applications in Figure 2 are the hTEC services for SDN. The SDN composition application provides the service to retrieve the data about the network, such as the average delays between the nodes (i.e., hosts, switches and routers). These data are used for designing an SDN and determining the cerebellum functions and their locations during service composition. The SDN session application interacts with the SDN control layer to create and manage the designed SDN during the execution of the simulation.

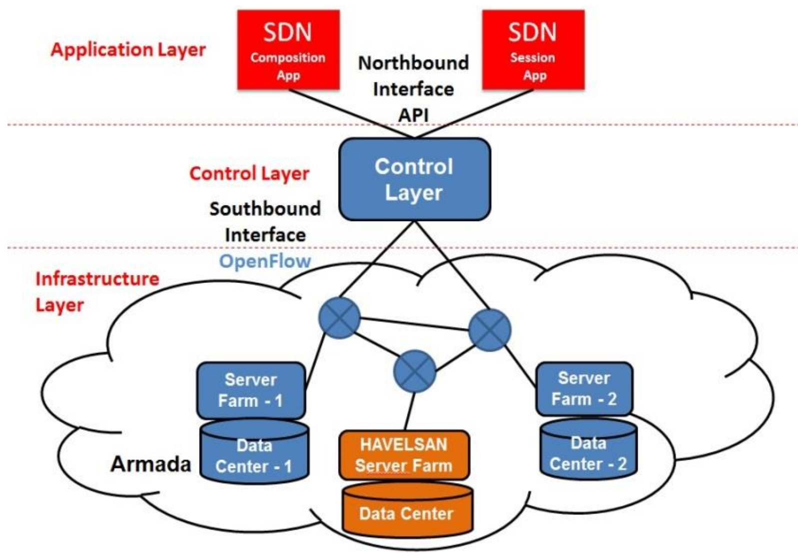

Figure 4. Software Defined Networking (SDN) for hTEC.

3 JMSOS REQUIREMENTS

Space capabilities are complex systems made up of various components including satellite production, checkout and storage facilities, launch facilities, user terminals, ground stations, manned or unmanned spacecrafts, payloads (e.g., sensors) and communication links (Rainey, Davis 2004). Many space capabilities require multiples of these components, such as tens of spacecrafts and ground stations. The components of a space capability are typically procured from different commercial or governmental organizations. Recently, the vendors for space capabilities including space lift are more and more frequently commercial companies.

Space technologies, capabilities and their components used to be controlled only by few nations. They are now available and affordable not only to state but also non-state entities. Moreover, it is not necessary to own all its components for having the space capability. It is possible to access the services by the space capabilities that the others own. Therefore, space is not a safe and secure place for sophisticated intelligence, surveillance, reconnaissance (ISR), communications and navigation technologies for few nations anymore, but a challenging and integrated part of the joint operations especially when defending against hybrid threats (Cayirci, Bruzzone, Longo and Gunneriusson 2016).

Many terrestrial systems critical to military operations, such as navigation and communications, depend on space systems, although it is sometimes not easy to recognize this dependence. Therefore, military has to have processes:

- To determine the space capability requirements
- To contribute the development of new space technologies applicable for military
- To use the available space capabilities optimally
- To defend the components of the space capabilities
- To prevent the belligerents/adversaries from using space capabilities effectively

To realize the global advantages provided by space forces, all space capabilities and the means to protect them, should be integrated into all kinds of military planning including defense, advance and response planning especially in strategic and operational levels. For this, the commanders and their staff *must understand the applications of space capabilities, have access to space-based support sufficient to accomplish their missions, use space systems to the degree needed for completing required tasks expeditiously, and make recommendations to deny or limit an adversary's access to space and use of space systems* (NATO 2009).

Therefore, JMSOS needs to be designed with a large scope which includes:

- Support to planning and operations
- Support to capability development and experimentation
- Support to education, individual training, collective training, exercises and wargames

Our JSMOS services in hTEC are designed with this scope in mind, nevertheless, at our initial step in BSigma, we focus on the JMSOS requirements for operational level exercises and wargames.

There is a need for the simulation support to the exercises/wargames in four space mission areas: space control, space force enhancement, space support, space force application (NATO 2009):

- Space control operations (SCO) are conducted to attain and maintain the space superiority which involves the counter measures against the adversaries' space capabilities. These measures include actions by air, land, maritime, space and special forces. SCO requires space situational awareness about space related conditions including space weather, constraints, capabilities and activities in, from, toward and through space. The details like orbits, payloads, frequencies are all of interest. SCO can be offensive or defensive. Offensive operations can be against not only the assets in space but also ground facilities and stations. Electronic control measures, such as, jamming an uplink or downlink are among the offensive SCO. Available combat models can be adapted to simulate a large subset of offensive and defensive SCO.
- Space force enhancement operations (SFEO) are to support the warfighter and to enhance the battlespace awareness. *There are five force enhancement functions: ISR; integrated tactical warning and attack assessment; environmental monitoring; communications; and position, velocity, time, and navigation* (NATO 2009). The combat models available for operational level exercises in NATO provide functions to simulate the results of a subset of SFEO. However, they are far from being sufficient.
- Space Support Operations (SSO) include space lift, satellite operations, reconstitution of space forces. Space lift delivers satellites, payloads and material to space. Satellite operations are conducted to maneuver, configure and sustain on-orbit forces and to activate on-orbit spares. Finally, reconstitution operations are for replenishing space forces when the existing forces degrade due to various reasons. SSO is seldom practiced in operational level exercises.
- Space Force Application Operations (SFAO) carried out by the weapon systems operating in or through space against terrestrial based targets. SFAO includes ballistic missile defense (BMD), theater ballistic missile defense (TBMD) and force projection. Please note that TBMD and BMD can be conducted also by means other than SFAO. This mission area is not practiced very often in operational level exercises because there is not any known asset available in space for this purpose.

When simulation services for these space mission areas are developed, the following characteristics special to space operations have to be taken into consideration (NATO 2009) (Rainey, Davis 2004):

- Global access and persistence: Satellites can fly over any location on earth, and stay on orbit for extended period of time. However, except for geostationary satellites, they stay over a location on Earth only a limited time.
- Coverage and propagation delay in communications. As the orbit altitude gets higher, the coverage area gets larger, however, the propagation delay in communications also gets longer. The return trip time for an electromagnetic signal from earth to a geostationary satellite is around 500 msec.
- Design life: Most satellites cannot be maintained or repaired. They also can have limited fuel on board to maintain the orbit or making changes in the orbit. Therefore, the lifetime of satellites is limited.
- Older technology: Although software defined technologies are changing this, the technology in a satellite is not the latest but the technology available before the launching day.
- Increasing affordability: New technologies introduce smaller and smaller satellites, such as, micro, nano, pico and femto satellites. More sophisticated satellites can be produced in less sophisticated production facilities and lifted into space more easily and less costly.
- Predictable orbits: Satellite orbits are predictable.
- Vulnerability: Ground to satellite links are susceptible to electronic counter measures and ground facilities and stations can be attacked.
- Resource considerations: Replacing or replenishing space forces need long lead times.
- Legal considerations: Numerous national and international laws have to be considered during planning.
- Space treaties: Although, there is currently no treaty that forbids the deployment of weapons other than weapons of mass destruction in space, many space related treaties introduce constraints to the military use of space.

4 JMSOS SERVICES IN HTEC AND BSIGMA ARCHITECTURE

The hTEC JMSOS has currently nine services in the service layer as shown in Table 2, where the relations between these services and space mission areas are also clarified. In the same table, the meta data and interface descriptions are given. Please note that the meta data fields that apply to every service in the table are written only once in the common to all services row. Please see Table 2 also for the short description of each service.

One of these nine hTEC JMSOS services, namely Space ISR, is a composed service, which means that it includes another service (i.e., Space Sensors Service). When Space ISR Service is used, the functionalities of Space Sensors Service is also automatically included into the service. We plan to include similar composed services into hTEC JSMOS service layer later. In the meantime, users can compose other composed services or in other words, service federations in service composition layer.

The criteria that we used when deciding about the hTEC JMSOS Services in the service layer are as follows:

- Requirement: Every service should address a subset of tasks for the space mission areas and characteristics summarized in Section 3. Moreover, all the space mission areas (i.e., requirements) have to be covered by the hTEC JMSOS Services.
- Dependence: An hTEC JMSOS service in the service layer should not depend on another service. Every service can be run alone when needed, although they can be federated with any other service in the hTEC JMSOS Service Layer.

- Minimality: The functions that may need to be used alone, should be organized as a service. This reduces both financial and computational (i.e., memory and computational power) costs.

Table 2: The JMSOS Services in the hTEC Service Layer.

Service Name	Field Name	Field Value	Mission Area
Common (fields with the same value for all the services in the table)	Standard	NATO MSG-136 Recommendations	All
	Standard version	V0.5	
	Service type	Constructive	
	QoS Parameters	To be determined	
	Service version	0.1	
	Date	01 March 2017	
	Developer	HAVELSAN	
	Service model	Model as a Service, Free	
	URL	To be determined	
	Delay constraint	To be determined	
	Return Type	Returns 1 if successfully terminated, the exception code otherwise.	
	Input Parameters	The handle for the Federation Object Model and Agreement (FOMA) when HLA is used, the scenario file handle otherwise	
Space craft & Orbits	Resolution	Entity	SCO and SSO
	Fidelity	High	
	Description	Simulates the status of spacecraft/satellite, payload, and orbit throughout the lifetime including production, launch and disposal.	
	Cerebellum	The Offset	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages	
Space Weather	Resolution	Vector	SCO and SSO
	Fidelity	High	
	Description	Generates realistic space weather based on the scenario file and user commands.	
	Cerebellum	Null	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	
Weapon Effects on Space Assets	Resolution	Entity	SCO
	Fidelity	Medium	
	Description	Simulates the effects of ordnance dropped at the terrestrial components of space capabilities. Note that the service computes the effects on the overall space capability.	
	Cerebellum	Null	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	
Space Electronic Warfare	Resolution	Entity	SCO and SFAO
	Fidelity	High	
	Description	Simulates the electronic warfare conducted against space capabilities or by space capabilities.	
	Cerebellum	Null	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	

Space Sensors	Resolution	Entity	SFEO
	Fidelity	Medium	
	Description	Simulates detection, recognition and identification by space sensors.	
	Cerebellum	Null	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	
Space ISR	Resolution	Aggregate	SFEO
	Fidelity	Medium	
	Description	Runs together with Space Sensors to generate ISR reports, still images and videos formatted according to NATO STANAGs 4609, 4633, 4658, 4676, 4545, 4559 and 5516.	
	Cerebellum	Null	
	Output	The socket details for the message passing and the messages formatted according to NATO STANAGs.	
Space Communications	Resolution	Entity	SFEO
	Fidelity	Medium	
	Description	Simulates the communication links between the ground stations and the satellites as well as in the space networks It can be used together with Space Electronic Warfare.	
	Cerebellum	Null	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	
GPS	Resolution	Entity	SFEO
	Fidelity	High	
	Description	Simulates the space capabilities for Global Positioning Systems.	
	Cerebellum	Null	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	
Space Weapon Effects	Resolution	Entity	SFAO
	Fidelity	High	
	Description	Simulates the effects of space based weapons.	
	Cerebellum	The offset	
	Output	If HLA is not used, the socket details for the XML message passing and XML formatted messages.	

In the service composition layer, the first hTEC JMSOS application, (i.e., JMSOS 1.0) is composed of Spacecraft and Orbit (ORB), Space Weather (WET) and Space Sensors (SEN) services. JMSOS 1.0 is not federated with SDN applications. That is planned for the later versions together with some other hTEC JMSOS services.

The service composition layer is not only for composing services but also serving the composed services, which are software applications composed of the hTEC JMSOS Services. These applications may consist of a single service. The main difference is that a service in the service layer needs to be linked to a software before it is compiled and run. In the other hand, an application made available by the service composition layer is a software as a service ready to be run. Any of these applications by the service composition layer can be federated with the other simulation systems by using distributed simulation technologies, such as HLA.

JMSOS 1.0 is designed as a proof of concept and demonstration application to be run on the extended BSigma Testbed which is depicted in Figure 5. That is called as extended testbed because JMSOS 1.0 is not only a federation of software as a service (SaaS) federates, namely ORB, WET and SEN but also a federation of platform as a service (PaaS) federates, i.e., virtual machines provided by Center for IP-based

Service Innovation(CIPSI) Datacenter in Stavanger, Norway and by the BSigma Datacenter in Ankara, Turkey. In the extended BSigma, a user in Ankara, Izmir or Stavanger, can run JMSOS 1.0 on a virtual desktop provided by the BSigma or CIPSI datacenter. Our experiments will continue. In the next step, we plan to federate JMSOS 1.0 running on the extended BSigma with the other services in NATO MSG-136 MSaaS environment, and demonstrate during in I/ITSEC and/or the NATO Coalition Warrior Interoperability Exploration, Experimentation, Examination Exercise (CWIX).

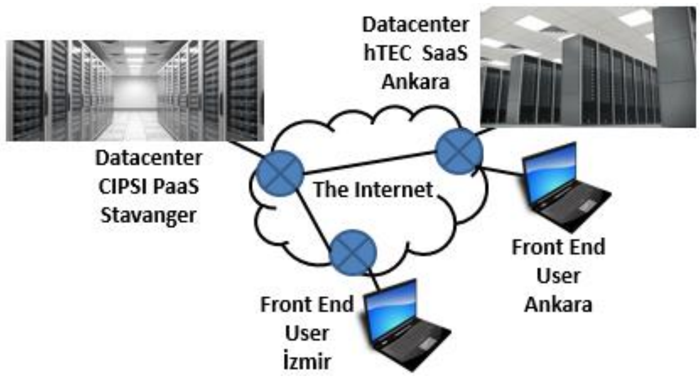

Figure 5. BSigma Architecture.

5 CONCLUSION

MSaaS is considered as a solution to major barriers in applying M&S to military capabilities, such as the perceived cost and time taken to compose and develop simulation systems, limited credibility resulting from unknown validity and ad-hoc processes. Since 2013, NATO has investigated MSaaS and developed recommendations and best practices for its employment for the M&S support in NATO and the Nations. Our training and experimentation cloud architecture, the hTEC, follows these recommendations.

Space capabilities are important for joint military operations, especially when they are expeditionary. For the successful application of the space capabilities, they need to be integrated into joint operations, which requires extensive testing and training. This is possible when the M&S support is available. The hTEC JMSOS services and architecture are developed to address this requirement. An application, namely JMSOS 1.0, is also implemented by using the hTEC JMSOS architecture and services, and the experiments are run on a testbed called the extended BSigma.

We plan to connect the extended BSigma to the international MSaaS networks, to federate the services in the hTEC JMSOS with the others, and to demonstrate during the international events, such as, I/ITSEC and CWIX.

ACKNOWLEDGMENTS

This work is a part of the project titled as "Service Oriented Cloud Based Simulator/Simulation Development and Modelling Infrastructure" funded by TUBITAK 1511-BIT-BBIL-2015-2 program.

REFERENCES

Cayirci, E. 2013. "Modelling and Simulation as a Service: A Survey". *In Proceedings of the 2013 Winter Simulation Conference, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl.* Washington DC, USA.

Cayirci E. 2013b. "Configuration Schemes for Modelling and Simulation as a Service Federations". *Simulation Transactions of the Society for Modelling and Simulation International, Vol. 89, Issue 11, pp. 1388 – 1399.*

Cayirci, E., Karapinar, H., & Ozcakir, L. 2015. “Cerebellum Function for MSaaS”. *EMSS 2015*.
Cayirci, E., Karapinar, H., & Ozcakir, L. 2016. “hTEC: A Layered MsaaS Architecture for Training and Experimentation Cloud”. I/ITSEC 2016.
Cayirci E., A. Bruzzone, F. Longo and H Gunneriusson, 2016. “A Model to Describe Hybrid Conflict Environments”. I3M.
Helal, S. 2002. “Standards for Service Discovery and Delivery”. *IEEE Pervasive Computing, Vol. 1, Issue 3, pp. 95-100*.
Hu, F., Q. Hao, and K. Bao. 2014. “A Survey on Software Defined Network and OpenFlow: From Concept to Implementation”. *IEEE Communications Surveys and Tutorials, Vol, 16, Iss 4, pp. 2181-2206*.
IEEE 1516-2010. 2010. *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-- Framework and Rules*.
NATO MSG 136. 2017. *Operational Concept Document for the Allied Framework for M&S as a Service*. Draft v0.5.0. NATO Science and Technology Organization Technical Report STO-TR-MSG-136-OCD.
NATO. 2009. *Allied Joint Doctrine for Air and Space Operations*. NATO AJP-3.3(A).
Rainey, L.B., Davis, P.K. 2004. *Space Modeling and Simulation: Roles and Applications Throughout the System Life Cycle.* AIAA. ISBN: 1884989152, 9781884989155.
Siegfried, R., T. Berg, A. Cramp, and W. Huiskamp. 2014. “M&S as a Service: Expectations and challenges”. *SISO 2014 Fall Simulation Interoperability Workshop, Paper 14F-SIW-040, Orlando, USA*.
Taylor, S.J.E., A. Khan, K. L Morse, A. Tolk, L. Yilmaz, J. Zander, and P. J Mosterman. 2015. “Grand challenges for modeling and simulation: simulation everywhere from cyberinfrastructure to clouds to citizens”. *Simulation Transactions of the Society for Modelling and Simulation International, Vol. 91, Issue 7, pp. 648-665.*
X.500. 2016. Retrieved April 2016, from http://www.x500standard.com/
Zehe, D., W. Cai, A. Knoll, and H. Aydt. 2015. “Tutorial on a Modelling and Simulation Cloud Service”. *In Proceedings of the 2015 Winter Simulation Conference, edited by L.Yılmaz, W.K.V. Chan, I. Moon, T.M.K. Roeder, C. Macal, and M.D. Rossetti. Huttington Beach, USA.*

AUTHOR BIOGRAPHIES

ERDAL ÇAYIRCI is the CEO of Dataunitor AS and a Professor of the Department of Electrical Engineering and Computer Science at the University of Stavanger, Norway. He holds a Ph.D. in computer engineering. His research interests include military simulation, cloud computing, data science, mobile communications and sensor networks. His email is erdal.cayirci@uis.no.

HAKAN KARAPINAR is the Programs Section Head in the Training and Simulation Systems Department at HAVELSAN A.Ş., Turkey. He holds an M.Sc. in electrical engineering. His research interests include military simulation and training systems and cloud computing. His email is hakank@havelsan.com.tr.

LÜTFÜ ÖZÇAKIR is the Vice President at HAVELSAN A.Ş., Turkey. He holds an M.Sc. in electrical engineering. His research interests include military simulation and training systems and cloud computing.